VOICE BASED HOME AUTOMATION

A Project report submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE ENGINEERING

Submitted by

P.NITEESH NAIDU	**-**	**317126510045**
P.ABHINAYREDDY	**-**	**317126510048**
P.VIVEK	**-**	**317126510058**
T.SATYANARAYANA	**-**	**318126510L05**

Under the guidance of

CH.K.RUPESH KUMAR
(ASSISTANT PROFESSOR)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, Bheemili Mandal, Visakhapatnam Dist. (A.P)

2020 – 2021

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

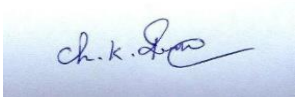
(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, Bheemili Mandal, Visakhapatnam Dist. (A.P)

2017-2021

BONAFIDE CERTIFICATE

This is to certify that this project report **"VOICE BASED HOME AUTOMATION"** is the bonafide work of **P.Niteesh Naidu (317126510045), P.Abhinay Reddy (317126510048), P.Vivek (317126510058),T.SatyaNarayana(318126510L05) of IV/IV CSE** carried out the project work under my supervision.



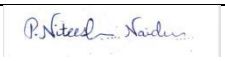
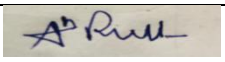
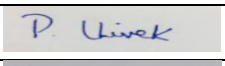
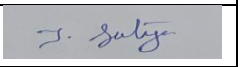
CH.K.RUPESH KUMAR
(ASSISTANT PROFESSOR)
COMPUTER SCIENCE & ENGINEERING
ANITS

Dr.R.SIVA RANJINI
(HEAD OF THE DEPARTMENT)
COMPUTER SCIENCE & ENGINEERING
ANITS

DECLARATION

We, **P.Niteesh Naidu (317126510045), P.Abhinayreddy (317126510048), P.Vivek (317126510058), T.SatyaNarayana (318126510L05),** of final semester B.Tech.in the department of Computer Science and Engineering from ANITS, Visakhapatnam, hereby declare that the project work entitled "**Voice Based Home Automation**" is carried out by us and submitted in partial fulfilment of the requirements for the award of Bachelor of Technology in Computer Science and Engineering, Anil Neerukonda Institute of Technology & Sciences(A) during the academic year 2017-2021 and has not been submitted to any other university for the award of any kind of degree.

Below are the E- Signatures of the team members:

S.NO	REGISTER NO:	NAME	DIGITAL SIGN
1	317126510045	P.Niteesh Naidu	
2	317126510048	P.Abhinayreddy	
3	317126510058	P.Vivek	
4	318126510L05	T.SatyaNarayana	

ACKNOWLEDGMENT

An endeavour over a long period can be successful with the advice and support of many well-wishers. We take this opportunity to express our gratitude and appreciation to all of them.

We owe our tributes to Dr.R.Sivaranjani, Head of the Department, Computer Science & Engineering, ANITS, for her valuable support and guidance during the period of projectimplementation.

We wish to express our sincere thanks and gratitude to our project guide Mr.Ch.K.Rupesh Kumar Assistant Professor, Computer Science and Engineering, ANITS,for the stimulating discussions, in analysing problems associated with our project work and for guiding us throughout the project. Project meetings were highly informative.

We express our warm and sincere thanks for the encouragement, untiring guidance and the confidence they had shown in us. We are immensely indebted for their valuable guidance throughout our project.

We also thank all the staff members of the CSE department for their valuable advice.We also thank supporting staff for providing resources as and when required.

P.NITEESH NAIDU - 317126510045

P.ABHINAYREDDY - 317126510048

P.VIVEK - 317126510058

T.SATYANARAYANA - 318126510L05

TABLE OF CONTENTS

TABLE OF FIGURES

Abstract

Automation is a trending topic in the 21st century making it play an important role in our daily lives.The main attraction of any automated system is reducing human labour,effort,time and errors due to human negligence.With the development of modem technology ,smart phones have become a necessity for every person on this planet.Applications are being developed on android systems that are useful to us in various ways.Another upcoming technology is natural language processing which enables us to command and control things with our voice.Combining all of these,our paper presents a micro controller based voice controlled home automation system using smart phones.Such a system will enable users to have control over every appliance in his/her home with their voice.All that the user needs is an android smartphone ,which is present in almost everybody's hand nowadays,and a control circuit.When the first computers came around,achieving the level of sophistication so as to narrate commands using voice to a machine was only realised in science fiction.However with tremendous breakthrough in the field,we are at the precipice of truly using voice to interface with devices.

Problem Statement

The goal of this project is to create a model that will be able to recognize and determine the hmnan voice and operate devices according to the instructions given by the person . Though the goal is to create a model which can act with a bluetooth module, it can be extended by using wifi module too . The major goal of the proposed system is understanding human language and making the device act according to the instructions given by the person .

1. INTRODUCTION

1.1. OBJECTIVE:

The concept of Home Autmnation is gaining popularity as it helps in reducing human effort and errors and thus increasing the efficiency.With the help of Home Automation system,we can control different appliances like lights,fans,TV,AC e.t.c..,

Additionally ,a home automation system can also provide other features like security,alarms,emergency systems etc. can be integrated.

1.2. PROJECT OVERVIEW:

There are many types of Home automation systems like Bluetooth controlled,Intemet Controlled,Remote Controlled(IR Remote) etc. Each type has its own advantages and disadvantages.

In this project,we have designed a voice activated home automation system,where different appliances are controlled by sending a voice command.

The voice activated home Automation project is implemented using arduino UNO,Bluetooth,and a smartphone.

We are showing a voice control home automation system to control appliances with their own voice command.

COMPONENTS :

To accomplish this project using

- Arduino UNO
- HC-05 Bluetooth module
- 12 v 4 channel relay board
- Arduino IDE software

Automation plays a key role in human life. Home automation allows us to control household electrical appliances like light, door, fan, AC etc. It also provides home security and emergency system to be activated. Home automation not only refers to reduce human efforts but also energy efficiency and time saving. The main objective of home automation and security is to help handicapped and old aged people who will enable them to control home appliances and alert them in critical situations.

This project put forwards the implementation of home automation and security system using Arduino microprocessor and Android smartphone. Home appliances are connected to the microprocessor and communication is established between the Arduino and Android mobile device or tablet via Bluetooth module. We would develop an authentication to the system for authorized person to access home appliances. The device with low cost and scalable to less modification to the core is much important. It presents the design and implementation of automation system that can monitor and control home appliances via android phone or tablet

Voice controlled wireless smart home system has been presented for elderly and disabled people.

The concept of controlling home appliances using human voice is interesting. The proposed system has two main components, they are

voice recognition system, and wireless system. This system to control home appliances uses a voice controlled android application. By the increasing use of PC (personal computers), internet, mobile phone and wireless technology, it makes it easy for a user to remotely access and control the appliances.

A lot of research has been done and many solutions have been proposed to remotely access the appliances. Some of them used internet, wireless technology to communicate and control home appliances, others used Bluetooth and GSM technology for controlling the home appliances.

The main aim of our system is to build a perfect companion for someone to be at home. Generally, home automation research targeted many needs like applications that provide the luxury smart requirements while some threw light on the special needs for elderly and disabled etc. our system is a computer based system that can accept voice to direct commands and process them. The system provides us switching any device ON/OFF.

1.3. LITERATURE SURVEY :

1.3.1. BLUETOOTH BASED HOME AUTOMATION:

- In Bluetooth based home automation systems the home appliances are connected to the Arduino BT board at input output ports using relay.
- The program of the Arduino BT board is based on the high level interactive C language of microcontrollers.
- The connection is made via Bluetooth. The password protection is provided so only authorized users are allowed to access the appliances.
- The Bluetooth connection is established between Arduino BT board and phone for wireless communication.
- In this system the python script is used and it can install on any of the Symbian OS environments, it is portable.
- One circuit is designed and implemented for receiving the feedback from the phone, which indicates the status of the device.

1.3.2. WIFI BASED HOME AUTOMATION:

- Wi-Fi based home automation systems mainly consist of three modules, the server, the hardware interface module, and the software package. The figure shows the system model layout.
- Wi-Fi technology is used by servers, and hardware Interface modules to communicate with each other. The same technology is used to login to the server web based application.
- The server is connected to the internet, so remote users can access server web based applications through the internet using a compatible web browser.
- Software of the latest home automation system is split to server application software, and Microcontroller (Arduino) firmware.
- The Arduino software, built using C language, using IDE comes with themicrocontroller itself.

- Arduino software is culpable for gathering events from connected sensors, then applies action to actuators and pre-programed in the server. Another job is to report the and record the history in the server DB.

- The server application software package for the proposed home automation system, is a web based application built using asp.net.

- The server application software can be accessed from an internal network or from the internet if the server has real IP on the internet using any internet navigator that supports asp.net technology.

- Server application software is culpable for maintaining the whole home automation system, setup, configuration. Server uses a database to keep log of home automation system components, we choose to use XML files to save system log.

1.4. SYSTEM ARCHITECTURE:

Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). 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. It contains everything needed to support the microcontroller, simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

"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. (IDE = integrated development environment)

The input voltage (7 - 12 V) to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through the Vin pin, or, if supplying voltage via the power jack, access it directly through Vin pin.

1.4.1. ARCHITECTURE:

Here Architecture is of Arduino or precisely the IC of Arduino (ATmega328p). The ATmega328/P is a low-power CMOS 8-bit microcontroller based on the **AVR®** enhanced **RISC** (reduced instruction set computer) architecture.

In Order to maximize performance and parallelism, the AVR uses **Harvard** architecture - with separate memories and buses for program and data. Instructions inthe program memory are executed with a single level of pipelining.
The clock is controlled by an external **16MHz Crystal Oscillator.**

1.4.2. The basic working of CPU of ATmega328:-

1. The data is uploaded in serial via the port (being uploaded from the computer'sArduino IDE). The data is decoded and then the instructions are sent to **instruction register** and it decodes the instructions on the same clock pulse.

2. On the next clock pulse the next set of instructions are loaded in the instructionregister.

3. **In general purpose registers** : the registers are of 8-bit but there are 3 16-bit registers also.

a. **8-bit** registers are used to store data for normal calculations and results.

b. **16-bit registers** are used to store data from the timer counter in 2 different registers. Eg. X-low & X-high. They are fast, and are used to store specific hardware functions.

4. **EEPROM** stores data permanently even if the power is cut out. Programming inside an EEPROM is slow.

5. **Interrupt Unit** checks whether there is an interrupt for the execution of instruction to be executed in ISR (Interrupt Service Routine).

6. **Serial Peripheral Interface (SPI)** is an interface bus commonly used to send data between microcontrollers and small peripherals such as Camera, Display, SD cards, etc. It uses separate clock and data lines, along with a select line to choose the device you wish to talk to.

7. **Watchdog** timer is used to detect and recover from MCU malfunctioning.

8. **Analog comparator** compares the input values on the positive and negative pin, when the value of the positive pin is higher the output is set.

9. **Status and control** is used to control the flow of execution of commands by checking other blocks inside the CPU at regular intervals.

10. **ALU** (Arithmetic and Logical unit)The high performance AYR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations b/w general purpose registers are executed. The ALU operations are divided into 3 main categories - arithmetic, logical and bit-function.

11. **1/0 pins** The digital inputs and outputs (digital I/O) on the Arduino are what allow you to connect the Arduino sensors, actuators, and other ICs. Leaming how to use them will allow you to use the Arduino to do some really useful things, such as

12.reading switch inputs, lighting indicators, and controlling relay outputs.

2. Components Description

2.1. ARDUINO UNO

Arduino is an open source computer hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical and digital world. The project's products are distributed as open-source hardware and software, which are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially in preassembled form, or as do-it-yourself (DIY) kits. Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards or Breadboards (shields) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers are typically programmed using a dialect of features from the programming languages C and C++. In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) based on the Processing language project. The Arduino project started in 2003 as a program for students at the Interaction Design Institute Ivrea in Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats, and motion detectors.

The name Arduino comes from a bar in Ivrea, Italy, where some of the founders of the project used to meet. The bar was named after Arduin of Ivrea, who was the margrave of the March of Ivrea and King of Italy from 1002 to 1014.

2.1.1. Features of the Arduino

UNO: Microcontroller: ATmega328

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: 50 mA

Flash Memory: 32 KB of which 0.5 KB used by bootloader

SRAM: 2 KB (ATmega328)

EEPROM: 1 KB (ATmega328)

Clock Speed: 16 MHz

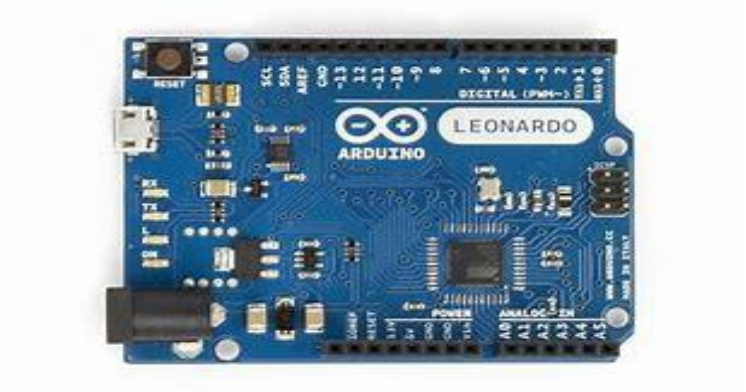


Fig 2.1. Arduino uno

2.1.2. ARDUINO HARDWARE PART:-

Arduino is open-source hardware. The hardware reference designs are distributed under a Creative Commons Attribution Share-Alike 2.5 license and are available on the Arduino website. Layout and production files for some versions of the hardware are also available. Although the hardware and software designs are freely available under copyleft licenses, the developers have requested the name Arduinoto be exclusive to the official product and not be used for derived works without permission. The official policy document on use of the Arduino name emphasizes that the project is open to incorporating work by others into the official product. Several Arduino-compatible products commercially released have avoided the project name by using various names ending in -duino. Most Arduino boards consist of an Atmel 8-bit AVR microcontroller (ATmega8, ATmega168, ATmega328, ATmega1280, ATmega2560) with varying amounts of flash memory, pins, and features. The 32-bit Arduino Due, based on the Atmel SAM3X8E was introduced in 2012. The boards use single or double-row pins or female headers that facilitate connections for programming and incorporation into other circuits. These may connect with add-on modules termed shields. Multiple and possibly stacked shields may be individually addressable via an I²C serial bus. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator or ceramic resonator. Some designs, such as the LilyPad, run at 8 MHz and dispense with the onboard voltage regulator due to specific form-factor restrictions. Arduino microcontrollers are pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory. The default bootloader of the Arduino UNO is the optiboot bootloader. Boards are loaded with program code via a serial connection to another computer. Some serial Arduino boards contain a level shifter circuit to convert between RS232 logic levels and transistor–transistor logic(TTL) level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods. When used with traditional microcontroller tools, instead of the Arduino IDE, standard AVR in-system programming (ISP) programming is used.

The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The Diecimila, Duemilanove, and current Uno provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.1-inch (2.54 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Board and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards. Many Arduino-compatible and Arduino-derived boards exist. Some are functionally equivalent to an Arduino and can be used interchangeably. Many enhance the basic Arduino by adding output drivers, often for use in school-level education, to simplify making buggies and small robots. Others are electrically equivalent but change the form factor, sometimes retaining compatibility with shields, sometimes not. Some variants use different processors, of varying compatibility.

2.1.3. ARDUINO SOFTWARE PART:-

IDE The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version

2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

A program written with the Arduino IDE is called a sketch. [58] Sketches are saved on the development computer as text files with the file extension .ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension .pde.

A minimal Arduino C/C++ program consist of only two functions:

setup(): This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch.

loop(): After setup() has been called, function loop() is executed repeatedly in the main program. It controls the board until the board is powered off or is reset.

Blink example

Most Arduino boards contain a light-emitting diode (LED) and a load resistor connected between pin 13 and ground, which is a convenient feature for many tests and program functions. A typical program for a beginning Arduino programmer blinks a LED repeatedly. This program uses the functions pinMode(), digitalWrite(), and delay(), which are provided by the internal libraries included in the IDE environment. This program is usually loaded into a new Arduino board by the manufacturer.

2.2. RELAY:

A relay is an electrically operated switch. Many relays use an electromagnet to mechanically operate a switch, but other operating principles are also used, such as solid-state relays. Relays are used where it is necessary to control a circuit by a separate low-power signal, or where several circuits must be controlled by one signal. The first relays were used in long distance telegraph circuits as amplifiers: they repeated the signal coming in from one circuit and re-transmitted it on another circuit. Relays were used extensively in telephone exchanges and early computers to perform logical operations. A type of relay that can handle the high power required to directly control an electric motor or other loads is called a contactor. Solid-state relayscontrol power circuits with no moving parts, instead using a semiconductor device to perform switching. Relays with calibrated operating characteristics and sometimes multiple operating coils are used to protect electrical circuits from overload or faults; in modern electric power systems these functions are performed by digital instruments still called "protective relays". Magnetic latching relays require one pulse of coil power to move their contacts in one direction, and another, redirected pulse to move them back. Repeated pulses from the same input have no effect. Magnetic latching relays are useful in applications where interrupted power should not be able to transition the contacts. Magnetic latching relays can have either single or dual coils. On a single coil device, the relay will operate in one direction when power is applied with one polarity, and will reset when the polarity is reversed. On a dual coil device, when polarized voltage is applied to the reset coil the contacts will transition. AC controlled magnetic latch relays have single coils that employ steering diodes to differentiate between operate and reset commands.

A type of relay that can handle the high power required to directly control an electric motor or other loads is called a contactor. Solid-state relayscontrol power circuits with no moving parts, instead using a semiconductor device to perform switching. Relays with calibrated operating characteristics and sometimes multiple operating coils are used to protect electrical circuits from overload or faults; in modern electric power systems these functions are performed by digital instruments still called "protective relays". The Arduino Relay module allows a wide range of microcontroller such as Arduino, AVR ,PIC, ARM with digital outputs to control larger loads and devices like AC or DC Motors, electromagnets, solenoids, and incandescent light bulbs. This module is designed to be integrated with 2 relays that it is capable of control 2 relays.The relay shield use one QIANJI JQC-3F high-quality relay with rated load 7A/240VAC,10A/125VAC,10A/28VDC.The relay output state is individually indicated by a light-emitting diode.

2.2.1. Types:-

1)Coaxial relay

Where radio transmitters and receivers share one antenna, often a coaxial relay is used as a TR (transmit-receive) relay, which switches the antenna from the receiver to the transmitter. This protects the receiver from the high power of the transmitter. Such relays are often used in transceivers which combine transmitter and receiver in one unit. The relay contacts are designed not to reflect any radio frequency power back toward the source, and to provide very high isolation between receiver and transmitter terminals. The characteristic impedance of the relay is matched to the transmission line impedance of the system, for example, 50 ohms.

2)Contactor

A contactor is a heavy-duty relay with higher current ratings, used for switching electric motors and lighting loads. Continuous current ratings for common contactors range from 10 amps to several hundred amps. High-current contacts are made with alloys containing silver. The unavoidable arcing causes the contacts to oxidize; however, silver oxide is still a good conductor. Contactors with overload protection devices are often used to start motors.

3)Force-guided contacts relay

A 'force-guided contacts relay' has relay contacts that are mechanically linked together, so that when the relay coil is energized or de-energized, all of the linked contacts move together. If one set of contacts in the relay becomes immobilized, no other contact of the same relay will be able to move. The function of force-guided contacts is to enable the safety circuit to check the status of the relay. Force-guided contacts are also known as "positive-guided contacts", "captive contacts", "locked contacts", "mechanically linked contacts", or "safety relays". These safety relays have to follow design rules and manufacturing rules that are defined in one main machinery standard EN 50205 : Relays with forcibly guided (mechanically linked) contacts. These rules for the safety design are the one that are defined in type B standards such as EN 13849-2 as Basic safety principles and Well-tried safety principles for machinery that applies to all machines. Force-guided contacts by themselves can not guarantee that all contacts are in the same state, however they do guarantee, subject to no gross mechanical fault, that no contacts are in opposite states.
Otherwise, a relay with several normally open (NO) contacts may stick when energised, with some contacts closed and others still slightly open, due to mechanical tolerances. Similarly, a relay with several normally closed (NC) contacts may stick to the unenergised position, so that when energised, the circuit through one set of contacts is broken, with a marginal gap, while the other remains closed. By introducing both NO and NC contacts, or more commonly, changeover contacts, on the same relay, it then becomes possible to guarantee that if any NC contact is closed, all NO contacts are open, and conversely, if any NO contact is closed, all NC contacts are open. It is not possible to reliably ensure that any particular contact is closed, except by potentially intrusive and safety-degrading sensing of its circuit conditions, however in safety systems it is usually the NO state that is most important, and as explained above, this is reliably verifiable by detecting the closure of a contact of opposite sense. Force-guided contact relays are made with different main contact sets, either NO, NC or changeover, and one or more auxiliary contact sets, often of reduced current or voltage rating, used for the monitoring system. Contacts may be all NO, all NC, changeover, or a mixture of these, for the monitoring contacts, so that the safety system designer can select the correct configuration for the particular application. Safety relays are used

as part of an engineered safety system.

4)Latching relay

A latching relay (also called "impulse", "bistable", "keep", or "stay" relays) maintains either contact position indefinitely without power applied to the coil. The advantage is that one coil consumes power only for an instant while the relay is being switched, and the relay contacts retain this setting across a power outage. A latching relay allows remote control of building lighting without the hum that may be produced from a continuously (AC) energized coil. In one mechanism, two opposing coils with an over-center spring or permanent magnet hold the contacts in position after the coil is de-energized. A pulse to one coil turns the relay on and a pulse to the opposite coil turns the relay off. This type is widely used where control is from simple switches or single-ended outputs of a control system, and such relays are found in avionics and numerous industrial applications. Another latching type has a remanent core that retains the contacts in the operated position by the remanent magnetism in the core. This type requires a current pulse of opposite polarity to release the contacts. A variation uses a permanent magnet that produces part of the force required to close the contact; the coil supplies sufficient force to move the contact open or closed by aiding or opposing the field of the permanent magnet. A polarity controlled relay needs changeover switches or an H bridge drive circuit to control it. The relay may be less expensive than other types, but this is partly offset by the increased costs in the external circuit. In another type, a ratchet relay has a ratchet mechanism that holds the contacts closed after the coil is momentarily energized. A second impulse, in the same or a separate coil, releases the contacts. This type may be found in certain cars, for headlamp dipping and other functions where alternating operation on each switch actuation is needed. A stepping relay is a specialized kind of multi-way latching relay designed for early automatic telephone exchanges.An earth leakage circuit breaker includes a specialized latching relay. Very early computers often stored bits in a magnetically latching relay, such as ferreed or the later remreed in the 1ESS switch. Some early computers used ordinary relays as a kind of latch—they store bits in ordinary wire spring relays or reed relays by feeding an output wire back as an input, resulting in a feedback loop or sequential circuit. Such an electrically latching relay requires continuous power to maintain state, unlike magnetically latching relays or mechanically racheting relays. In computer memories, latching relays and other relays were replaced by delay line memory, which in turn was replaced by a series of ever-faster and ever-smaller memory technologies.

5)Machine tool relay

A machine tool relay is a type standardized for industrial control of machine tools, transfer machines, and other sequential control. They are characterized by a large number of contacts (sometimes extendable in the field) which are easily converted from normally open to normally closed status, easily replaceable coils, and a form factor that allows compactly installing many relays in a control panel.
Although such relays once were the backbone of automation in such industries as automobile assembly, the programmable logic controller (PLC) mostly displaced the machine tool relay from sequential control applications. A relay allows circuits to be switched by electrical equipment: for example, a timer circuit with a relay could switch power at a preset time. For many years relays were the standard method of controlling industrial electronic systems. A number of relays could be used together to carry out complex functions (relay logic). The principle of relay logic is based on relays which energize and de-energize associated contacts. Relay logic is the predecessor of ladder logic, which is commonly used in programmable logic controllers.

6)Mercury relay

A mercury relay is a relay that uses mercury as the switching element. They are used where contact erosion would be a problem for conventional relay contacts. Owing to environmental considerations about significant amount of mercury used and modern alternatives, they are now comparatively uncommon.

7)Mercury-wetted relay

A mercury-wetted reed relay is a form of reed relay in which the contacts are wetted with mercury. Such relays are used to switch low-voltage signals (one volt or less) where the mercury reduces the contact resistance and associated voltage drop, for low-current signals where surface contamination may make for a poor contact, or for high-speed applications where the mercury eliminates contact bounce. Mercury wetted relays are position-sensitive and must be mounted according to the manufacturer's specifications to work properly. Because of the toxicity and expense of liquid mercury, these relays are now rarely used. The mercury-wetted relay has one particular advantage, in that the contact closure appears to be virtually instantaneous, as the mercury globules on each contact coalesce. The current rise time through the contacts is generally considered to be a few picoseconds, however in a practical circuit it will be limited by the inductance of the contacts and wiring. It was quite common, before the restrictions on the use of mercury, to use a mercury-wetted relay in the laboratory as a convenient means of generating fast rise time pulses, however although the rise time may be picoseconds, the exact timing of the event is, like all other types of relay, subject to considerable jitter, possibly milliseconds, due to mechanical imperfections. The same coalescence process causes another effect, which is a nuisance in some applications. The contact resistance is not stable immediately after contact closure, and drifts, mostly downwards, for several seconds after closure, the change perhaps being 0.5 ohm.

8)Multi-voltage relays

Multi-voltage relays are devices designed to work for wide voltage ranges such as 24 to 240 VAC and VDC and wide frequency ranges such as 0 to 300 Hz. They are indicated for use in installations that do not have stable supply voltages.

9)Overload protection relay

Electric motors need overcurrent protection to prevent damage from over-loading the motor, or to protect against short circuits in connecting cables or internal faults in the motor windings. The overload sensing devices are a form of heat operated relay where a coil heats a bimetallic strip, or where a solder pot melts, releasing a spring to operate auxiliary contacts. These auxiliary contacts are in series with the coil. If the overload senses excess current in the load, the coil is de-energized. This thermal protection operates relatively slowly allowing the motor to draw higher starting currents before the protection relay will trip. Where the overload relay is exposed to the same ambient temperature as the motor, a useful though crude compensation for motor ambient temperature is provided. The other common overload protection system uses an electromagnet coil in series with the motor circuit that directly operates contacts. This is similar to a control relay but requires a rather high fault current to operate the contacts. To prevent short over current spikes from causing nuisance triggering the armature movement is damped with a dashpot. The thermal and magnetic overload detections are typically used together in a motor protection relay.
Electronic overload protection relays measure motor current and can estimate motor winding temperature using a "thermal model" of the motor armature system that can be set to provide more accurate motor protection. Some motor protection relays include temperature detector inputs for direct measurement from a thermocouple or resistance thermometersensor embedded in the winding

10) Polarized relay

A polarized relay places the armature between the poles of a permanent magnet to increase sensitivity. Polarized relays were used in middle 20th Century telephone exchanges to detect faint pulses and correcttelegraphic distortion.

11) Reed relay

A reed relay is a reed switch enclosed in a solenoid. The switch has a set of contacts inside an evacuated or inert gas-filled glass tube which protects the contacts against atmospheric corrosion; the contacts are made of magnetic material that makes them move under the influence of the field of the enclosing solenoid or an external magnet. Reed relays can switch faster than larger relays and require very little power from the control circuit. However, they have relatively low switching current and voltage ratings. Though rare, the reeds can become magnetized over time, which makes them stick 'on' even when no current is present; changing the orientation of the reeds with respect to the solenoid's magnetic field can resolve this problem. Sealed contacts with mercury-wetted contacts have longer operating lives and less contact chatter than any other kind of relay.

12) Safety relays

Safety relays are devices which generally implement safety functions. In the event of a hazard, the task of such a safety function is to use appropriate measures to reduce the existing risk to an acceptable level.

13) Solid-state contactor

A solid-state contactor is a heavy-duty solid state relay, including the necessary heat sink, used where frequent on-off cycles are required, such as with electric heaters, small electric motors, and lighting loads. There are no moving parts to wear out and there is no contact bounce due to vibration. They are activated by AC control signals or DC control signals from programmable logic controllers (PLCs), PCs, transistor-transistor logic (TTL) sources, or other microprocessor and microcontroller controls.

14)Solid-state relay

A solid-state relay (SSR) is a solid state electronic component that provides a function similar to an electromechanical relay but does not have any moving components, increasing long-term reliability. A solid-state relay uses a thyristor, TRIAC or other solid-state switching device, activated by the control signal, to switch the controlled load, instead of a solenoid. An optocoupler (a light-emitting diode (LED) coupled with a photo transistor) can be used to isolate control and controlled circuits.

15)Static relay

A static relay consists of electronic circuitry to emulate all those characteristics which are achieved by moving parts in an electro-magnetic relay.

2.2.2. APPLICATIONS OF RELAY:-

Relays are used wherever it is necessary to control a high power or high voltage circuit with a low power circuit, especially when galvanic isolation is desirable. The first application of relays was in long telegraph lines, where the weak signal received at an intermediate station could control a contact, regenerating the signal for further transmission. High-voltage or high-current devices can be controlled with small, low voltage wiring and pilots switches. Operators can be isolated from the high voltage circuit. Low power devices such as microprocessors can drive relays to control electrical loads beyond their direct drive capability. In an automobile, a starter relay allows the high current of the cranking motor to be controlled with small wiring and contacts in the ignition key.

Electromechanical switching systems including Strowger and Crossbar telephone exchanges made extensive use of relays in ancillary control circuits. The Relay Automatic Telephone Company also manufactured telephone exchanges based solely on relay switching techniques designed by Gotthilf Ansgarius Betulander. The first public relay based telephone exchange in the UK was installed in Fleetwood on 15 July 1922 and remained in service until 1959.

The use of relays for the logical control of complex switching systems like telephone exchanges was studied by Claude Shannon, who formalized the application of Boolean algebra to relay circuit design in A Symbolic Analysis of Relay and Switching Circuits. Relays can perform the basic operations of Boolean combinatorial logic. For example, the boolean AND function is realised by connecting normally open relay contacts in series, the OR function by connecting normally open contacts in parallel. Inversion of a logical input can be done with a normally closed contact. Relays were used for control of automated systems for machine tools and production lines. The Ladder programming language is often used for designing relay logic networks.

Early electro-mechanical computers such as the ARRA, Harvard Mark II, Zuse Z2, and Zuse Z3 used relays for logic and working registers. However, electronic devices proved faster and easier to use.

Because relays are much more resistant than semiconductors to nuclear radiation, they are widely used in safety-critical logic, such as the control panels of radioactive waste-handling machinery.
Electromechanical protective relays are used to detect overload and other faults on electrical lines by opening and closing circuit breakers.

2.3. BLUETOOTH MODULE(HC-05 Bluetooth Module)

2.3.1. HC-05 Specification:

Bluetooth protocal: Bluetooth Specification v2.0+EDR

Frequency: 2.4GHz ISM band

Modulation: GFSK(Gaussian Frequency Shift Keying)

Emission power: ≤4dBm, Class 2

Sensitivity: ≤-84dBm at 0.1% BER

Speed: Asynchronous: 2.1Mbps(Max) / 160 kbps,

Synchronous: 1Mbps/1Mbps

Security: Authentication and encryption

Profiles: Bluetooth serial port

Power supply: +3.3VDC 50mA

Working temperature: -20 ~ +75Centigrade

Dimension: 26.9mm x 13mm x 2.2 mm

2.3.2. Overview

HC-05 module 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 (Enhanced Data Rate)3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04- External single chip Rluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature).

Bluetooth Module HC-05 The Bluetooth module HC-05 is a MASTER/SLAVE module.By default the factory setting is SLAVE.The Role of the module (Master or Slave) can be configured only by AT COMMANDS.The slave modules cannot initiate a connection to another Bluetooth device, but can accept connections.Master module can initiate a connection to other devices.The user can use it simply for a serial port replacement to establish connection between MCU and GPS, PC to your embedded project, etc.

Fig 2.2. BLUETOOTH HC-05

2.3.3. Pin Description:-

The HC-05 Bluetooth Module has 6pins. They are as follows:

ENABLE: When enable is pulled LOW, the module is disabled which means the module will not turn on and it fails to communicate.When enable is left open or connected to 3.3V, the module is enabled i.e the module remains onand communication also takes place.

Vcc: Supply Voltage 3.3V to 5V

GND: Ground pin
TXD & RXD: These two pins acts as an UART interface for communication

STATE: It acts as a status indicator.When the module is not connected to paired with any other bluetooth device,signal goes Low.At this low state,the led flashes continuously which denotes that the module is not paired with other device.When this module is connected to/paired with any other bluetooth device,the signal goes High.At this high state,the led blinks with a constant delay say for example 2s delay which indicates that the module is paired.

BUTTON SWITCH: This is used to switch the module into AT command mode.To enable AT command mode,press the button switch for a second.With the help of AT commands,the user can change the parameters of this module but only when the module is not paired with any other BT device.If 39 the module is connected to any other bluetooth device, it starts to communicate with that device and fails to work in AT command mode.

HC-05 Default Settings:-

Default Bluetooth Name: “HC-05”

Default Password: 1234 or 0000

Default Communication: Slave

Default Mode: Data Mode

Data Mode Baud Rate: 9600, 8, N, 1

Command Mode Baud Rate: 38400, 8, N, 1

Default firmware: LINVOR

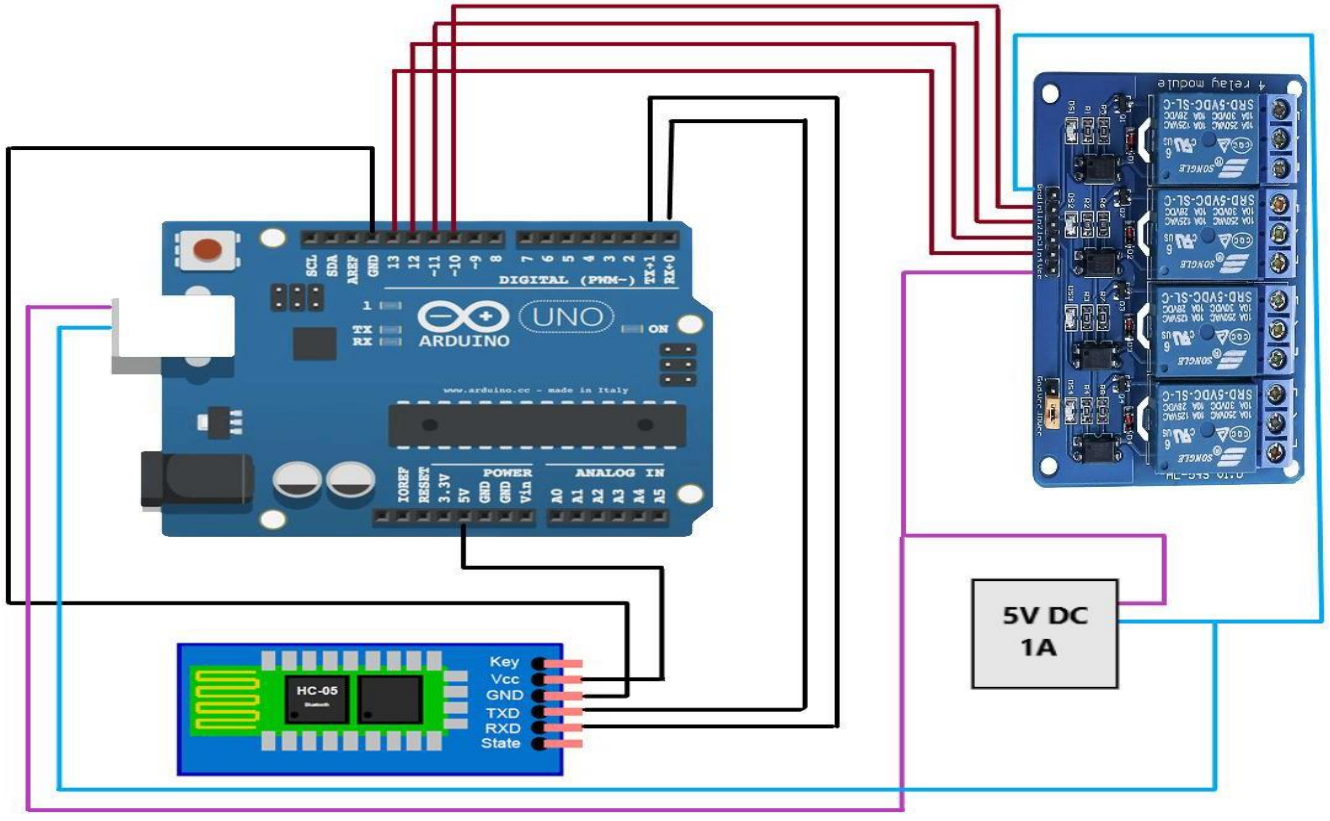


Fig 2.3.

BLOCK DIAGRAM OF HOME AUTOMATION USING BLUETOOTH MODULE

3. Modules

3.1. Bluetooth module:

- A Bluetooth module is interfaced to the Arduino board at the receiver end while on the transmitter end, a GUI application on the cell phone sends ON/OFF commands to the receiver where loads are connected. By touching the specified location on the GUI, the loads can be turned ON/OFF remotely through this technology.

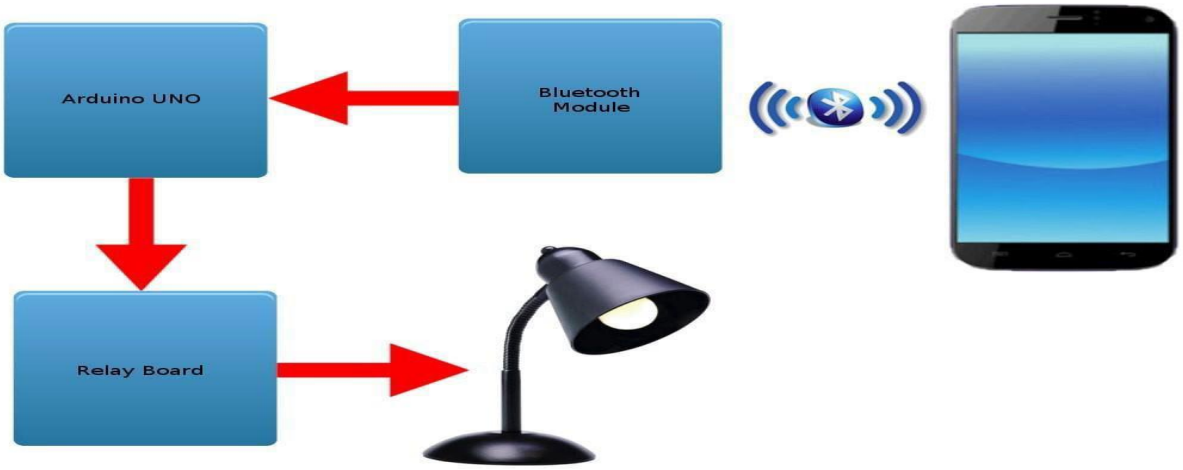


Fig 3.1. Bluetooth module

3.2. Voice module:

- Uses android mobiles internal voice recognition to pass voice commands to yourrobot
  Pairs with Bluetooth Serial Modules and sends in the recognized voice as astring
  for example if you say Hello the android phone will return a sting *Hello# to your bluetooth module *and # indicate the start and stop bits
  Can Be used with any micro controller which can handle strings
  Examples Platforms : Arduino , ARM , PICAXE , MSP430 , 8051 based and many other processors and controllers.

4. CONNECTING THE REQUIRED CONNECTIONS

4.1. ARDUINO

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.

4.2. BLUETOOTH

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.

4.3. RELAY

Relay is basically an electromagnetic switchwhich can be turn on and off by an applying the voltage across its contacts.

• In this project used a 5V 4-channel relay.

4.4. 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.

4.5. 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.

5. WORKING PRINCIPAL

Using the above components we implement our system. The core component of this system is the Arduino Uno which has a microcontroller i.e Atmega 328. Atmega 328 has a32KB flash, it is needed to burn a boot loader and download arduino sketches. The boot loader is programmed under ISP program controller.

An adapter of 12V output power supply is used as an input to the voice controlled arduino system. Relays are connected to the output pins of Arduino Uno, these are used as switches to the loads.

Android is a mobile operating system based on Linux kernel and currently developed by Google. We prefer android platform because of its huge market globally and it is easy to use user interface. The voice recognizer which is an inbuilt feature of android phones is used to build an application which the user can operate to automate the appliances at his house. For wireless communication system a Bluetooth module HC-05 is used as a remote which is connected to the control unit for sensing the signals sent by the android voice application.

The microcontroller device with the Bluetooth module and relay circuit needs to be connected to the switch board. Then we need to launch the android based application – "ANDROID MEETS ROBOT" on our smart phone. Through the application we can instruct the microcontroller to switch ON/OFF an appliance. After getting the instruction through the Bluetooth module, the microcontroller gives the signal to the relay board.

The application first searches for the Bluetooth device. If it is available then it launches the voice recognizer. It reads the voice and converts the audio signal into string. It provides a value for each appliance which will be fed to the microcontroller device. The microcontroller uses the port in serial mode. After reading the data it decodes the input value and sends a signal to the parallel port through which the relay circuit will be activated.

In this project, a simple Voice Activated Home Automation system is designed. Voice commands are used to control different appliances. We will now see the working of the project. All the connections are made as per the circuit diagram above.

After making the necessary connections, we have to switch on the power supply to the circuit. Now, we need to pair the Phone's Bluetooth to the HC – 05 Bluetooth Module. Before that, we have to install the App mentioned above in the phone. The home screen of the app looks some thing like this.

Next step is to connect the phone with the Bluetooth module. For this, choose the option "Connect Robot" and select the appropriate Bluetooth Device. If the devices aren't paired earlier, we need to pair them now using the Pin of the HC – 05 Bluetooth Module.

Next step is to connect the phone with the Bluetooth module. For this, choose the option "Connect Robot" and select the appropriate Bluetooth Device. If the devices aren't paired earlier, we need to pair them now using the Pin of the HC – 05 Bluetooth Module.

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".

USE CASE DIAGRAM

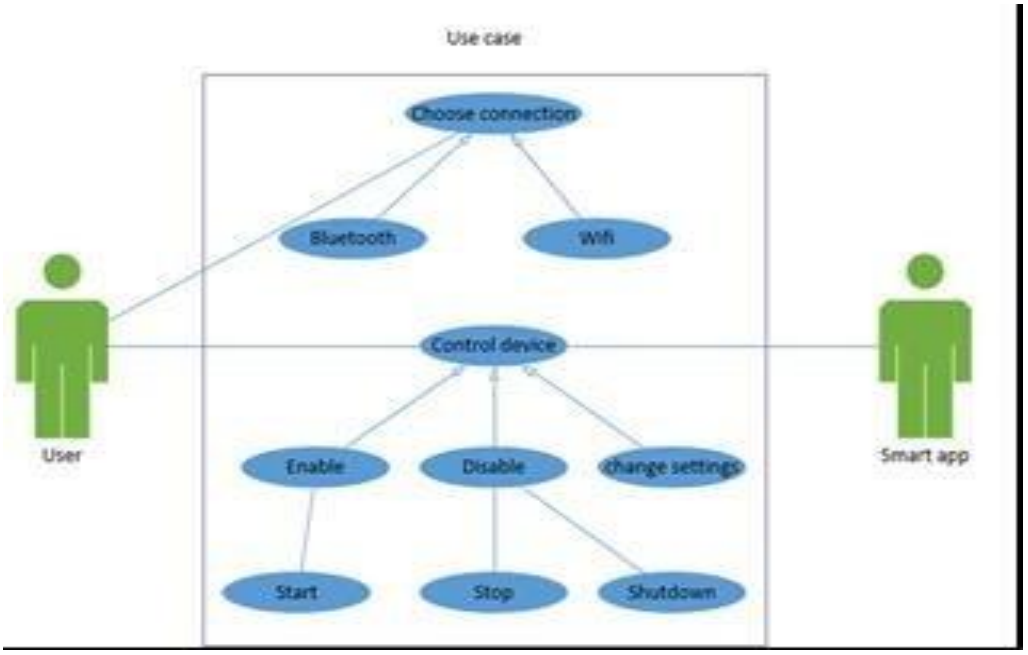
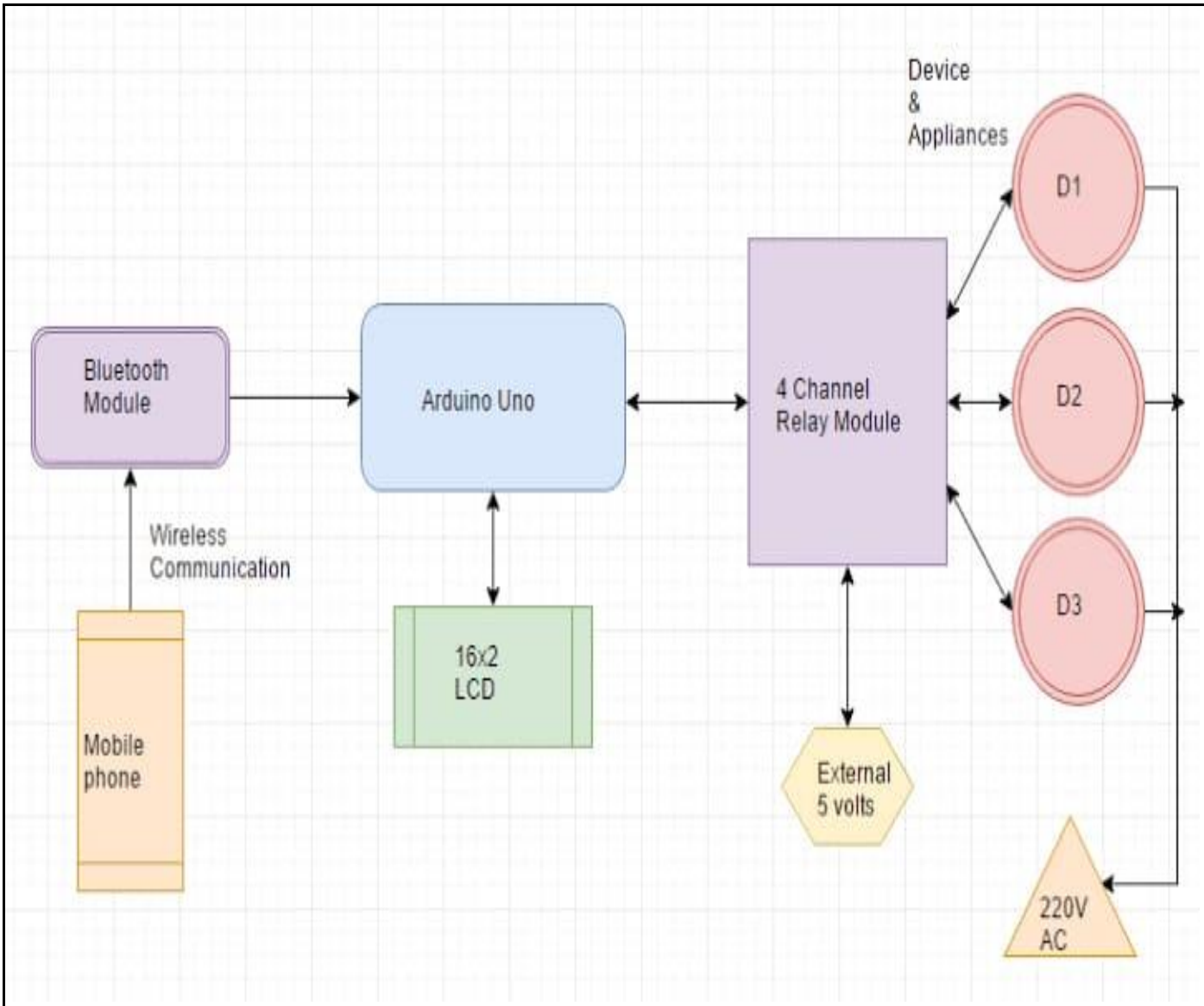
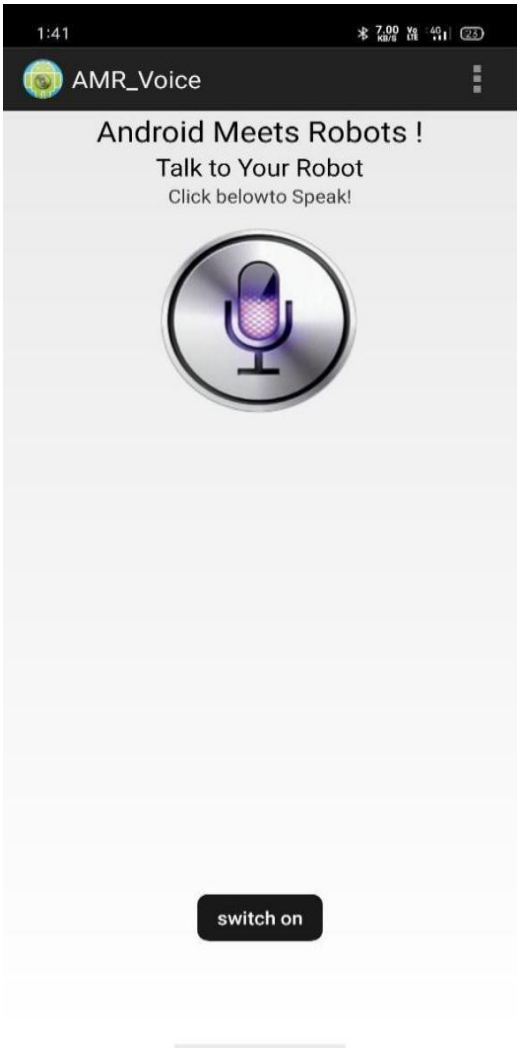


Fig 5.1.


Device
&
Appliances
D1
D2
D3
Bluetooth
Module
Arduino Uno
4 Channel
Relay Module
Wireless
Communication
16x2
LCD
Mobile
phone
External
5 volts
220V
AC

6. Sample Input and Outputs



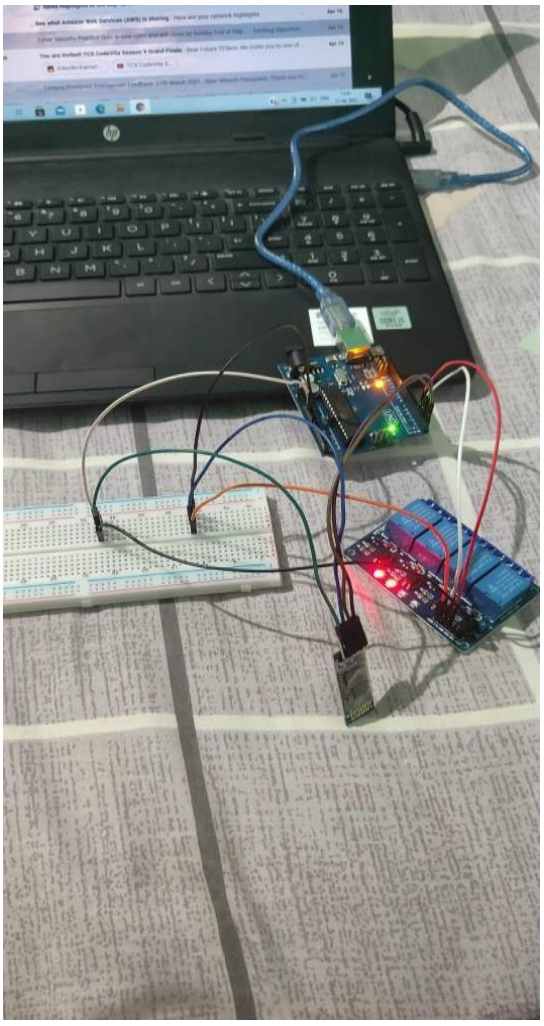
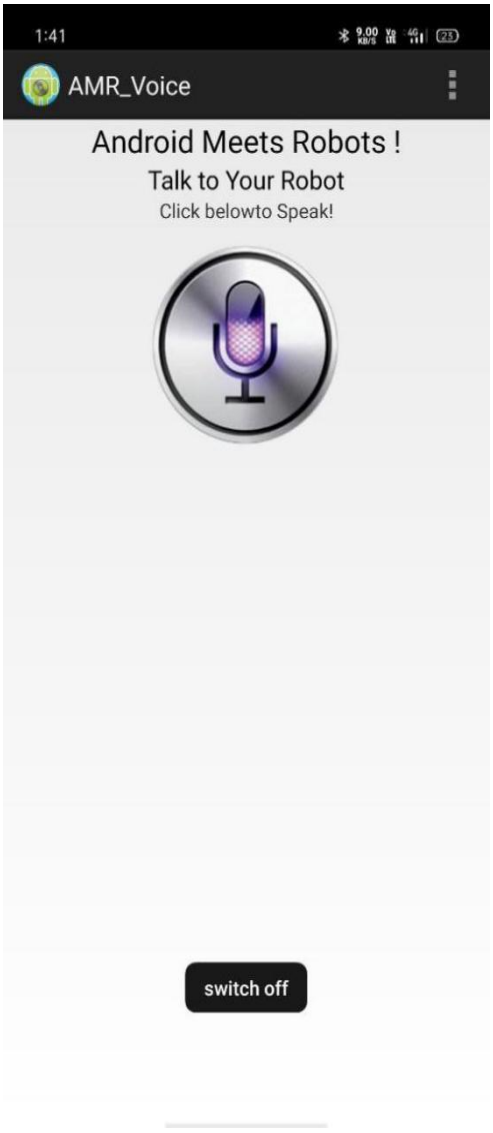
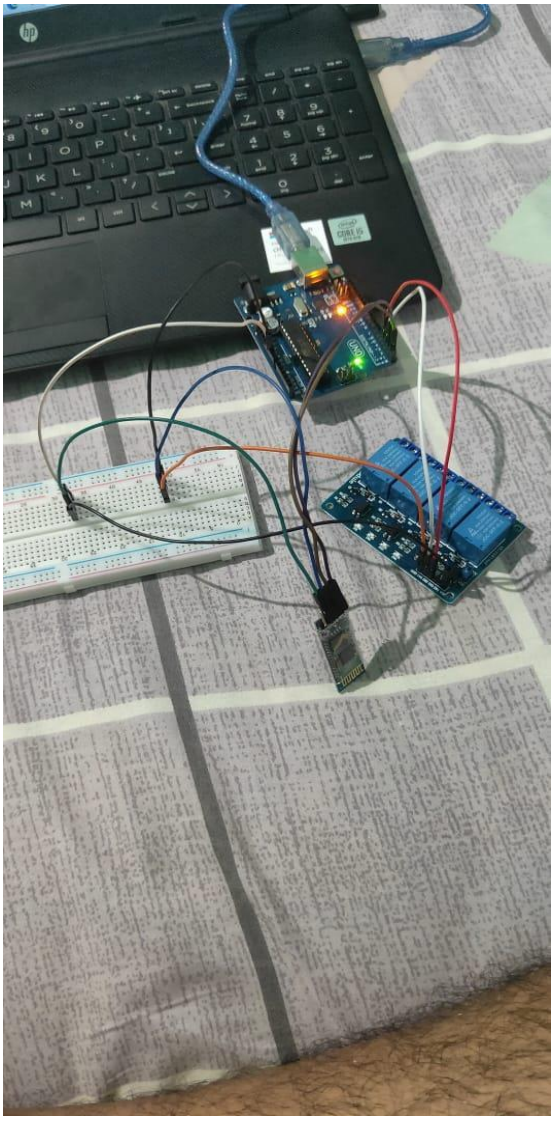


Fig 6.1. Switch on

Fig 6.2. Switch off





Android Meets Robots!

Talk to Your Robot

Click belowto Speak!

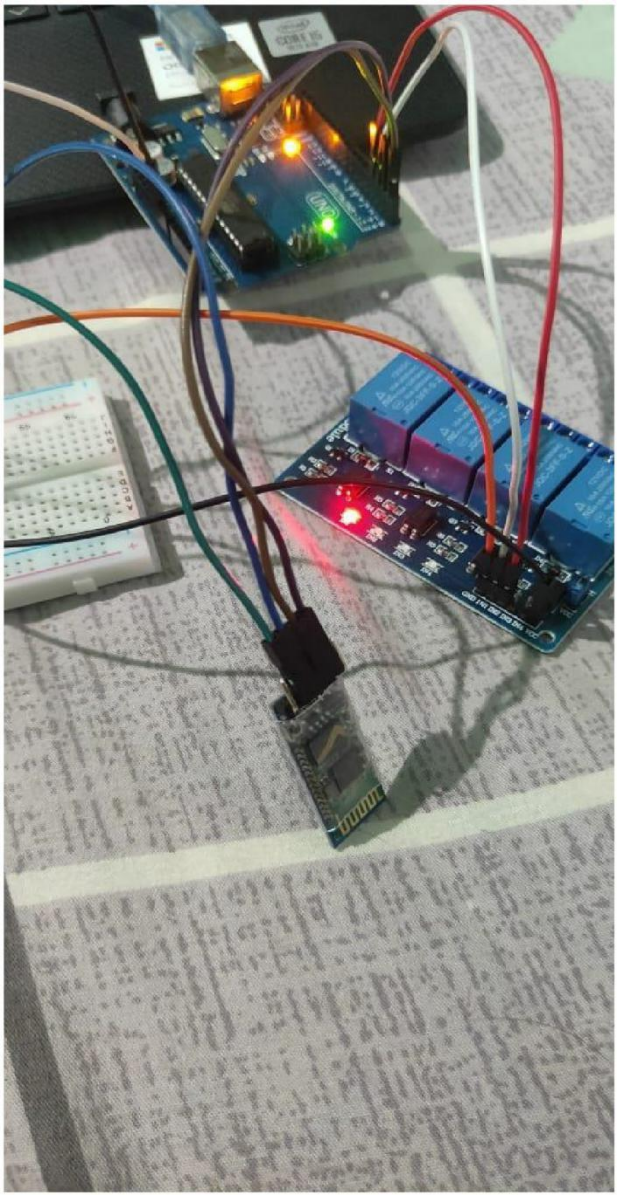


Fig 6.3. light on

7. ALGORITHMS AND PROGRAMMING

7.1. ARDUINO-

```
String voice;
#define relay1 2
#define relay2 3
void setup()
{
  Serial.begin(9600);
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  digitalWrite(relay1, LOW);
  digitalWrite(relay2, LOW);
}
void loop()
{
  while(Serial.available())
  {
    delay(10);
    char c = Serial.read();if (c
    == '#'){
      break;
    }
    voice += c;
  }
    if (voice.length() >0)
    {
      Serial.println(voice); if(voice
      == "*switch on"){
      switchon();
      }
      else if(voice == "*switch off"){
        switchoff();
      }
                                          //You can replace 'lamp on' with anything you
want...same applies to otherselse
      if(voice == "*fan on")
      {


        digitalWrite(relay1, HIGH);
      }
      else if(voice == "*fan off"){
        digitalWrite(relay1, LOW);
      }
      else if(voice == "*tube on"){
        digitalWrite(relay2, HIGH);
      }
```

```
 else if(voice == "*tube off"){
 digitalWrite(relay2, LOW);
       }
       voice="";
     }
}
void switchon()
  digitalWrite(relay1, HIGH);
  digitalWrite(relay2, HIGH);
}
void switchoff()
{
  digitalWrite(relay1, LOW);
  digitalWrite(relay2, LOW);
}

/*

   void..... ()
   {
     digitalWrite(..., LOW/HIGH);
     digitalWrite(..., LOW/HIGH);
   }
```

7.2. BLUETOOTH CONNECTIVITY CODE:

```
import android.bluetooth.BluetoothClass;
import com.google.android.things.bluetooth.BluetoothClassFactory;
import com.google.android.things.bluetooth.BluetoothConfigManager;
...

BluetoothConfigManager manager = BluetoothConfigManager.getInstance();
// Report the local Bluetooth device class as a speaker
BluetoothClass deviceClass = BluetoothClassFactory.build(
    BluetoothClass.Service.AUDIO,
    BluetoothClass.Device.AUDIO_VIDEO_LOUDSPEAKER);

manager.setBluetoothClass(deviceClass);
import com.google.android.things.bluetooth.BluetoothConfigManager;
...

  BluetoothConfigManager manager = BluetoothConfigManager.getInstance();
// Report full input/output capability for this device
manager.setIoCapability(BluetoothConfigManager.IO_CAPABILITY_IO);
import com.google.android.things.bluetooth.BluetoothProfileManager;
import com.google.android.things.bluetooth.BluetoothProfile;
...

BluetoothProfileManager manager = BluetoothProfileManager.getInstance();
List<Integer> enabledProfiles = manager.getEnabledProfiles();
if (!enabledProfiles.contains(BluetoothProfile.A2DP_SINK)) {
Log.d(TAG, "Enabling A2DP sink mode.");
List<Integer> toDisable = Arrays.asList(BluetoothProfile.A2DP);
List<Integer> toEnable = Arrays.asList(
    BluetoothProfile.A2DP_SINK,
    BluetoothProfile.AVRCP_CONTROLLER);

manager.enableAndDisableProfiles(toEnable, toDisable);
}
import android.bluetooth.BluetoothDevice;
import com.google.android.things.bluetooth.BluetoothConnectionManager;
import com.google.android.things.bluetooth.BluetoothPairingCallback;
import com.google.android.things.bluetooth.PairingParams;
...

public class PairingActivity extends Activity {

BluetoothConnectionManager bluetoothConnectionManager;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    bluetoothConnectionManager = BluetoothConnectionManager.getInstance();
    bluetoothConnectionManager.registerPairingCallback(bluetoothPairingCallback);
}
```

```
  @Override
  protected void onDestroy() {
     super.onDestroy();
     bluetoothConnectionManager.unregisterPairingCallback(bluetoothPairingCallback);
  }

  private void startPairing(BluetoothDevice remoteDevice) {
     bluetoothConnectionManager.initiatePairing(remoteDevice);
  }

  private BluetoothPairingCallback bluetoothPairingCallback = new BluetoothPairingCallback() {

     @Override
     public void onPairingInitiated(BluetoothDevice bluetoothDevice,
           PairingParams pairingParams) {
        // Handle incoming pairing request or confirmation of outgoing pairing request
        handlePairingRequest(bluetoothDevice, pairingParams);
     }

     @Override
     public void onPaired(BluetoothDevice bluetoothDevice) {
        // Device pairing complete
     }

     @Override
     public void onUnpaired(BluetoothDevice bluetoothDevice) {
        // Device unpaired
     }

     @Override
     public void onPairingError(BluetoothDevice bluetoothDevice,
           BluetoothPairingCallback.PairingError pairingError) {
        // Something went wrong!
     }
  };
}
private void handlePairingRequest(BluetoothDevice bluetoothDevice, PairingParams pairingParams) {
  switch (pairingParams.getPairingType()) {
     case PairingParams.PAIRING_VARIANT_DISPLAY_PIN:
     case PairingParams.PAIRING_VARIANT_DISPLAY_PASSKEY:
        // Display the required PIN to the user
        Log.d(TAG, "Display Passkey - " + pairingParams.getPairingPin());
        break;
     case PairingParams.PAIRING_VARIANT_PIN:
     case PairingParams.PAIRING_VARIANT_PIN_16_DIGITS:
        // Obtain PIN from the user
        String pin = ...;
        // Pass the result to complete pairing
        bluetoothConnectionManager.finishPairing(bluetoothDevice, pin);
        break;
     case PairingParams.PAIRING_VARIANT_CONSENT:
     case PairingParams.PAIRING_VARIANT_PASSKEY_CONFIRMATION:
        // Show confirmation of pairing to the user
```

```
        ...
        // Complete the pairing process
        bluetoothConnectionManager.finishPairing(bluetoothDevice);
        break;
    }
}
import android.bluetooth.BluetoothDevice;
import com.google.android.things.bluetooth.BluetoothConnectionManager;
import com.google.android.things.bluetooth.BluetoothConnectionCallback;
import com.google.android.things.bluetooth.BluetoothProfile;
import com.google.android.things.bluetooth.ConnectionParams;
...

public class ConnectActivity extends Activity {

BluetoothConnectionManager bluetoothConnectionManager;


@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    bluetoothConnectionManager = BluetoothConnectionManager.getInstance();
    bluetoothConnectionManager.registerConnectionCallback(bluetoothConnectionCallback);
}

@Override
protected void onDestroy() {
    super.onDestroy();
    bluetoothConnectionManager.unregisterConnectionCallback(bluetoothConnectionCallback);
}

private void connectToA2dp(BluetoothDevice bluetoothDevice) {
    bluetoothConnectionManager.connect(bluetoothDevice, BluetoothProfile.A2DP_SINK);
}

// Set up callbacks for the profile connection process.
private final BluetoothConnectionCallback bluetoothConnectionCallback = new
BluetoothConnectionCallback() {
    @Override
    public void onConnectionRequested(BluetoothDevice bluetoothDevice, ConnectionParams
connectionParams) {
        // Handle incoming connection request
        handleConnectionRequest(bluetoothDevice, connectionParams);
    }

    @Override
    public void onConnectionRequestCancelled(BluetoothDevice bluetoothDevice, int requestType) {
        // Request cancelled
    }

    @Override
    public void onConnected(BluetoothDevice bluetoothDevice, int profile) {
        // Connection completed successfully
```

```
    }

    @Override
    public void onDisconnected(BluetoothDevice bluetoothDevice, int profile) {
       // Remote device disconnected
    }
  };
}
   private void handleConnectionRequest(BluetoothDevice bluetoothDevice, ConnectionParams
  connectionParams) {
  // Determine whether to accept the connection request
  boolean accept = false;
  if (connectionParams.getRequestType() ==
  ConnectionParams.REQUEST_TYPE_PROFILE_CONNECTION) {
    accept = true;
  }

  // Pass that result on to the BluetoothConnectionManager
  bluetoothConnectionManager.confirmOrDenyConnection(bluetoothDevice, connectionParams, accept);
  }
```

8. PERFORMANCE MEASURES

1.Flexibility: A home automation system needs to adapt to different users and their requirements. It needs to be able to recognize voice commands from all users irrespective of their age, gender, accent and pitch. It must also be able to work with appliances the user already owns, so that he doesn't have to purchase additional appliances. The system must have multiple interfaces so that the user can choose whichever is most convenient to him. Interfaces can include a simple voice command interface, a mobile application to control devices or a central hub with all these options.

2.Robustness: A home automation system needs to be robust i.e. it should work even under less than ideal conditions. The voice recognition module must be able to recognize commands even when the user is at a distance from the microphone [6]. Ambient noise and background disturbances must have minimal impact on performance. Suppose the microphone is placed in a room where both, a noisy fan and a television are turned on and the user wishes to give a command. It wouldn't be viable to expect the user to mute the television every time he wants to give a command, therefore a good home automation system must take all these factors into consideration and find a way around them.

3.Security: Modern home automation are also used to lock and unlock the doors in a house. It is obvious that if the system isn't designed with a focus on security this could be a major threat to its owners. Also, if the system works on a wireless network, it is imperative to ensure that the channel is secure, before any transmission is sent. A system that is built from the ground up with emphasis on safety of the client is preferable. This metric will also consider systems that provide additional features like intruder detection, smart alerts, secure locks, etc.

4.Cost: A major factor that decides how successful a home automation system is the cost the user has to incur. A system may have all the latest bells and whistles, but if the user cannot afford it he simply will not buy it. Cost includes but the initial cost of the product as well as the cost of installation and long term maintenance. An automation hub may appear to be cheap but when you consider the cost of all the additional appliances you need to buy along with the hub, the cost adds up. Also, if the system is wired, the cost of labour that is needed for installation needs to be kept in mind.

5. Response Time: Everyone today is used to instantaneous results, therefore in order for a home automation system to be truly useful it must respond to inputs in the shortest time possible. Some algorithms give superior results for command recognition but the time they take for computation is not ideal. An ideal algorithm would be one which strikes a perfect balance between computational power and response time.

9. REFERENCES

1. https://www.geeksforgeeks.org/what-is-arduino /
2. https://www.arduino.cc/en/Tutoria1/HomePage
3. https://create.arduino.cc/projecthub/mayooghgirish/arduino-bluetooth-basic-t utorial-d8b737
4. https://www.c-sharpcomer.com/article/turning-led-off-and-on-through-voice-re cognition/
5. https://www.c-sharpcomer.com/article/tum ing-led-off-and-on-through-voice-re cognition/
6. https://ijarcce.com /upload /2017/march- l 7/IJARCCE%20173.pdf
7. Voice Activated Home Automation,Published:March 7,2017,by anusha.
8. Voice Controlled Home Automation System for the Elderly or Disabled People, Published by:Aqueel-ur-rehman.
9. Voice control Home Automation published by:Md iqbal. https://www.instructables.com/Voice-Control-Home-Automation/
10. Iot Based Home Automation by Tejas baraskar Apr 2019 DOI: 10.22214/ijraset.2019.4363

https://www.researchgate.net/publication/335695672_IOT_based_Home_Automation

BASE PAPER

ISSN No:-2456-2165

IOT Based Home Automation Using Bluetooth with Security Enhancement

Anand Kishore Azad , M.Tech
Dept. of Electrical Electronics and Communication
Gandhi Institute of Technology (Gitam)
Vishakhapatnam, India

Abstract:- The world is moving very fast towards automation. Day by day People have less time to handle any work so adapt automation is simple way to handle any device or machine will work to our desire. This paper presents how to implement the effective Home Automation System with IOT and device. These frameworks are provide help to satisfy the requirements of the aged and also physical disabled at homes. In addition, the concept of home automation system can improve the traditional living standing of homes. The elemental system uses a wireless Bluetooth device provides a wireless access to Smart phones. The system style doesn't take away the present electrical switches and provides a safe management over the electrical switches with low voltage usage technique. This technique is meant to regulate electrical devices all over the house with simple putting in it, simple use and price effective style and implementation.

Keywords:- *IOT, Bluetooth, IDE.*

I. INTRODUCTION

Home automation systems area is one with advancement to the mechanical processes whereby human efforts area required with the machinery and instruments to control various operation in homes. It involves automatic operation of home appliances with totally different technologies and controlled over any of the devices like desktops, laptops smart phones or tablets. Home automation system makes the operations of various home appliances a lot of convenient and also saves energy. With the energy saving conception, home automation or building automation and smart homes makes life very straight forward today. It involves automatic operations of all electrical or electronic devices in homes or perhaps remotely through wireless communication like Internet of Things (IOT), Wi-Fi and Bluetooth is a system of connected physical objects that is accessible through. The 'Thing' in IOT may be someone with a monitor, i.e. objects that are assigned an information science address and have the power to gather and transfer knowledge over a network without manual help or intervention. Android software system is technology among the leading and most popular most well-liked systems in smart phones. Smart phones affordability increases day to day because of their sizes , technology enhancement with different movability. Android applications put in smart phones are also will be update every moment. The operator should use the screen of the phone to regulate the house appliances. This project is an android application that possesses the potential to regulate any kind of electrical appliances providing full remote access from smart phone using Bluetooth. Bluetooth technology is Wireless radio transmissions in an exceedingly short distance providing a necessary technology to make convenience, intelligence and controllability. This generates the personal space networks in the home surroundings, wherever of these appliances are interconnected to each other employing with a single controller.

II. PROBLEM IDENTIFICATION

When we design a home automation system, we have to consider the fact that the user should be able to connect to that Bluetooth module from any device he would wish to connect. Also he should able to change the host from one device to other device and that module should work accordingly. On displaying any error or fault, it should have the ability to diagnose it and the system should start working immediately when an instruction is given to improve the nature of wireless technology. A field programmable gate array (FPGA) board is used here to provide high security to our system but although any mobile user can connect with this system, this also lead to connect any unrecognized person with the home, using Bluetooth software in their smart phones.

III. WORKING PRINCIPLE

This system contains the different microprocessor based electronics devices like Bluetooth module, Arduino Uno, Relay drivers, step down transformer as power supply and some software applications those are as follows:

- The Bluetooth module is used to connect the mobile and Arduino Uno for the operations of appliances at any interval by the help of Bluetooth signal.
- For the interfacing and process the signals the Arduino Uno is used here.
- For the ON and OFF operations of the appliances the relay drivers used as electromagnetic switch.
- For the different loads individual relays is used either at same driver or at different drivers for each relays.
- The operations and the system conditions are displayed on the smart phones or on an alpha-numeric display.

ISSN No:-2456-2165

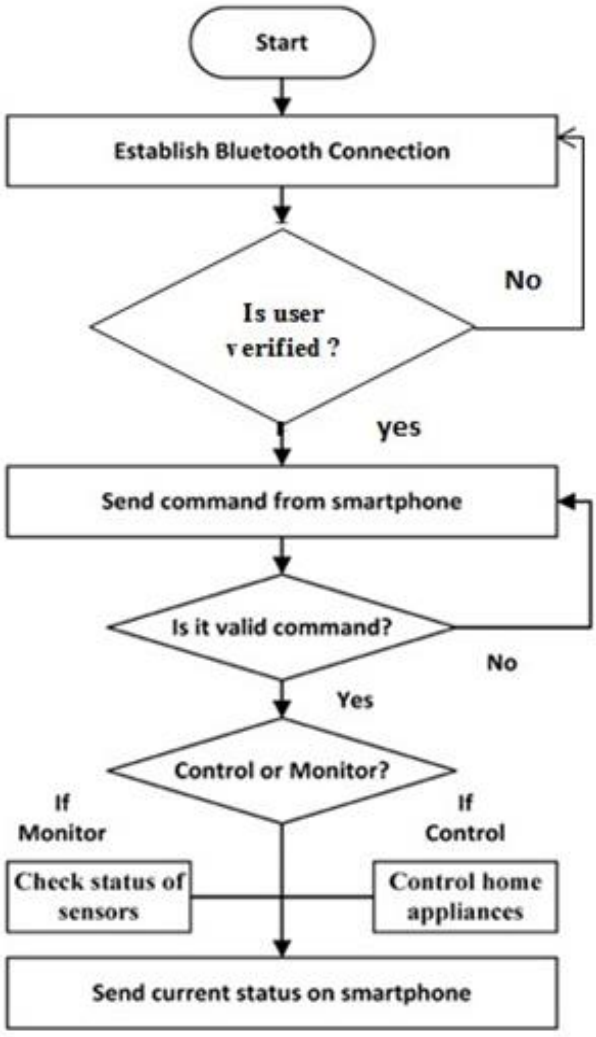


Fig 1:- Flow Chart of the Project

IV. HARDWARE DESIGN

The project system contains 3 main Hardware elements that is smart phone, Arduino Uno board and Bluetooth module.

Smartphone is employed to speak with Arduino board employing a smartphone application and Bluetooth Technology. During this analysis work Bluetooth module HC 06 and Arduino Uno used for hardware implementation.

A. Arduino Uno

Arduino Uno is microcontroller based open source hardware board in which ATmega328P microprocessor used. Here 14 pins are used as digital input or output and 6 pins uses as analog inputs which controls the switches or sensors and control multiple outputs and it operates on 5 to 9 volts DC supply and at 16 MHz quartz frequency clock . It has 32 KB flash memory, 2 Kb SRAM and 1 KB EEPROM.

B. Bluetooth Module HC-06

This device is employed for connection of Arduino Uno and smartphones. HC-06 could be a slave device and it will operates at 3 to 6 volts DC supply. It has 6 pins that is : State, RXD, TXD, GND, VCC and EN. For the communication of the Arduino and the Bluetooth device the RXD pin of HC 06 is connected to the Arduino TXD at pin 1 and the TXD of HC 06 connect to the RXD of the Arduino at pin 0.

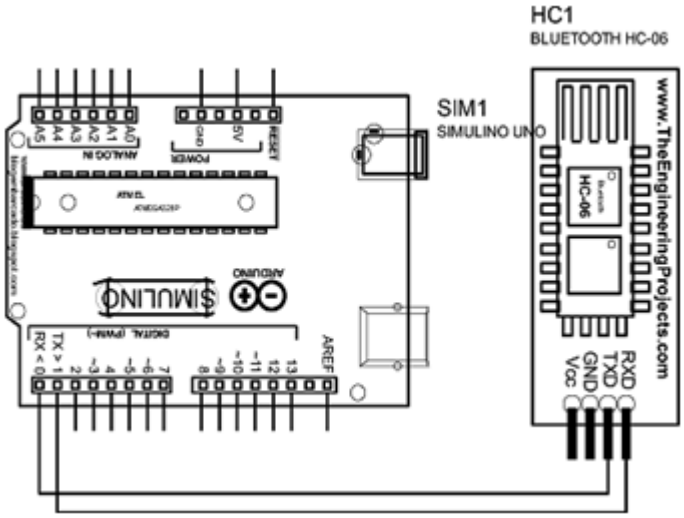


Fig 2:- Connection of Arduino with Bluetooth Module

C. Relay

The proper operation of the project requires an efficient, reliable and fast acting which basically consist of the relays and microcontroller devices. Relay is an electromagnetic device that is use to distinguish two different circuits electrically and connect the magnetically. usually for the interface of the relays electronic circuits associate with it that works on low voltages that works to trigger the devices at high voltages. for an example, a relay will use the 5V DC supply to trigger the 230V AC mains circuit or also higher voltages. A relay switch is operate through an armature which is attracted to an electromagnet or through plunger drawn into the solenoid and this is the input part of the relay that generate force when a low voltage applied to that .This voltage is termed as operative voltage. There are different sizes and configuration of the relays available in market according to operating voltage they are of 5V, 12V, 24V etc. The output section there are 3 terminal present (normally open (NO), normally closed (NC) and common point (COM) that connects or disconnect automatically and it consists of contact switch. once the operating voltage is applied in the relay coil it gets energized and therefore the COM changes contact to NO from NC. There are different relay configurations are in the market like SPST (single pole single throw), SPDT (single pole double throw) and DPDT (double pole double throw) that have completely different range of transmutation contacts. By mistreatment correct combination of contactors, the electric circuit is switched on and off.

➢ *Relay Board Module*

- *Description*

8 Channel Relay Board could be a easy and convenient to interface eight relays at the time for shift application in our project. Input voltage level support is common for the all relays on the board. Simple interface with Microcontrollers based mostly comes under analog circuits.

Fig 3:- 8 Channel Relay Module

- *Description of the module.*

These module offers 12 VDC at 336 mA with 8 SPDT relays output that is used for individually for numerous loads. Relay are of 5VDC/230 VAC with triggering level voltage of 3 to 5 VDC .The individual trigger pin available for all relays along with power and triggering LED indications. For Power supply screw terminal connector available, The total dimension of the board is 152 mm * 60 mm with 4 mounting holes.

D. Power Supply

Although the whole project is work on the 12 volts DC supply ,that is supplied either by 12 volts battery or by 12 volts AC to DC adapter which is consist of the step down transformer ,and the bridge rectifier linked with the 4.7 microfarad capacitor. . The Arduino board also powered via the USB connection using PC or Laptop or with an external power supply source. The adapter can be connected by plugging a 2.1mm plug into the board's power jack. The board can operate on an external DC supply of 6 to 15 volts. If supplied with less than 7V, then during operation the 5V pin of board may supply less than 5 volts and made the board may be unstable. If using more than 12V, the voltage regulator used in the board may overheat and damage the board. So the recommended range is 7 to 12 volts DC. To give the full 24 hours supply to the system mostly they are connected with power inverter used at home for the uninterrupted power supply.

V. SOFTWARE ARCHITECTURE

A. Arduino IDE

IDE stands for Integrated Development Environment, entire programming for projected system is finished in Arduino IDE tool. For serial communication between Arduino board and smartphone the Baud rate of 9600 bits per second used. Arduino IDE command " Serial.available( ) " is employed to receive data directly from smartphone and "Serial.println" command is used to transmit information directly from Arduino board to smartphone.

```
if(string == "A") % condition check
{
void LightOn()
{
digitalWrite(Light, HIGH);% Turn ON the light
}
}
if(string =="B")% condition check
{
void LightOff()
{
digitalWrite(Light, LOW); %Turn OFF the light
}
```

Fig 4:- Codes for Switching of Appliances

B. Bluetooth Terminal Application

A smart phone application named Blueterm could be a easy mechanical man app which will create dominant the pins of Arduino from associate degree mechanical man phone wirelessly attainable. Blueterm employs a straight forward mechanical man computer program to regulate Arduino digital and PWM pins, send text commands from phones through Bluetooth module to Arduino and receive knowledge from Arduino over Bluetooth serial module. Terminal emulator name VT-100 for communicating with any serial device using a bluetooth serial adapter and emulate serial communication over Bluetooth by RFCOMM/SPP protocol.

Fig 5:- User Interface Bluetooth Terminal Application

VI. PROBLEM SOLUTION

After connecting with the system firstly the system ask for the password that is either pin or passcode that is digitally verified by the system ,that the password enter in the system is correct or not thereafter the system give the full monitor and control access to the connected user. The one can be done by easily includes the some line of codes in software used for programming of arduino uno. It increase the security of the entire system and also stop the unrecognized user to connect with the system.

ISSN No:-2456-2165

```
void checkPassword()
{
if (password.evaluate())
{
digitalWrite(greenLED, HIGH);
lcd.print("User verified Access Granted");
}
Else
{
digitalWrite(redLED, HIGH);
lcd.print("Access Denied");
}
```

Fig 6:- Codes for Password Implement

VII. RESULTS AND CONCLUSION

The main aim of this paper was to propose the solution for the problem concerned with security purposes in the home automation basically related to software change and that is related to the Arduino IDE software that we use in this project that is necessary for the check for the connected user is recognized one or not to enhance the security purposes.

Fig 7:- Final Connection of the Project

In this paper, the architecture of low cost and flexible home Automation system using Arduino microcontroller based on the bluetooth wireless system is proposed and implemented. We use Arduino because this is easy to understand & its coding is very easy. By implementing this type of system we can also ensure the energy conservation that is waste every day in the huge homes and bungalows. By help of this system we can increase the efficiency of the appliances and also we can have the complete control over the home appliances from a long distance. This will Increase the comfortability of human being and it will reduce the Human efforts. The Proposed system is analyzed and tested several times within the range of 20 meters and it achieved 100% accuracy.

VIII. FUTURE WORK

This Proposed system is able to operate and control the appliances within short range only within 20m , for future research work it is recommended to increase the range and interface more sensors and also interface with the Google assistant system for enhance the security apart from this project, it should be a low cost and user friendly system. Moreover the home automation system can also be interfaced with biomedical (EMG) signals and It will be beneficial for physically challenged people, they will be able to control the appliances using their muscle's movement only.

REFERENCES

[1]. Amar Pawar, Rahul Sharan , Rahul Patil, and Sandip Chavan , Home Automation using Bluetooth and IOT, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 6, Issue 2, February 2018

[2]. Mummaka Sai Srinath1 , Manepalli Nanda Kishore And M.D. Anto Praveena, Interactive Home Automation System With Google Assistant, International Journal of Pure and Applied Mathematics, Volume 119 No. 12 2018, 14083-14086

[3]. Muhammad Asadullah , Smart Home Automation System Using Bluetooth Technology, (ICIEECT), 978-1-5090-3310-2/ 17/$3\.00 ©2017

[4]. J. Kavitha1, O. Naveen, P. Manoj Kumar, K. Subba Rao, Bluetooth Based Home Automation Using Arduino And Android Application, International Journal For Research In Applied Science & Engineering Technology (Ijraset) Issn: 2321-9653; Ic Value: 45.98; Sj Impact Factor: 6.887 Volume 6 Issue Iii, March 2018

[5]. Aayush Agarwal, Anshul Sharma, Asim Saket Samad, S Babeetha, Home Automation System Using Google Assistant, Vol-4 Issue-2 2018 Ijariie-Issn(O)-2395-4396

[6]. Manish Prakash Gupta, Google Assistant Controlled Home Automation, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 05 Issue: 05 | May-2018

[7]. M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[8]. Ramu M,Home Automation System Using IR Sensors. Internation Journal of Electrical and Electronics Engineering, Google scholar, 2015, 4, 11-16.

PAPER PUBLICATION

INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 7, Issue 3 - V7I3-1702)
Available online at: https://www.ijariit.com

Voice based home automation

Abhinayreddy
abhinayreddy.17.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

Niteesh Naidu
niteesh.17.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

Satyanarayana
tsatyanarayana.l18.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

Vivek
vivekpulidindi.17.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

Ch. K. Rupesh Kumar
rupeshkumar.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

ABSTRACT

Automation is a trending topic in the 21st century making it play an important role in our daily lives. The main attraction of any automated system is reducing human labor, effort, time and errors due to human negligence. With the development of modem technology ,smart phones have become a necessity for every person on this planet. Applications are being developed on android systems that are useful to us in various ways. Another upcoming technology is natural language processing which enables us to command and control things with our voice. Combining all of these, ourpaper presents a micro controller-based voice-controlled home automation system using smart phones. Such a system will enable users to have control over every appliance in his/her home with their voice. All that the user needs are an android smartphone ,which is present in almost everybody's hand nowadays, and a control circuit. 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 breakthrough in the field, we are at the precipice of truly using voice to interface with devices.

Keywords— *Voice Recognition, Home Automation, Arduino, Bluetooth,12v 4channel Relay, Connecting Wires*

1. PROBLEM STATEMENT

The goal of this project is to create a model that will be able to recognize and determine the human voice and operate devices according to the instructions given by the person . Though the goal is to create a model which can act with a Bluetooth module, it can be extended by using Wi-Fi module too . The major goal of the proposed system is understanding human language and making the device act according to the instructions given by the person.

2. INTRODUCTION

2.1 Objective

The concept of Home Automation is gaining popularity as it helps in reducing human effort and errors and thus increasing the efficiency. With the helpof Home Automation system, we can control different appliances like lights, fans, TV, AC etc., Additionally, a home automation system can also provide other features likesecurity, alarms, emergency systems etc. can be integrated.

2.2 Project Overview

There are many types of home automation systems like Bluetooth controlled, Internet Controlled, Remote Controlled(IR Remote) etc. Each type hasits own advantages and disadvantages. In this project, we have designed a voice activated home automation system, where different appliances are controlled by sending a voice command. The voice activated home Automation project is implemented using Arduino UNO, Bluetooth, and a smartphone. We are showing a voice control home automation system to controlappliances with their own voice command.

2.3 Components

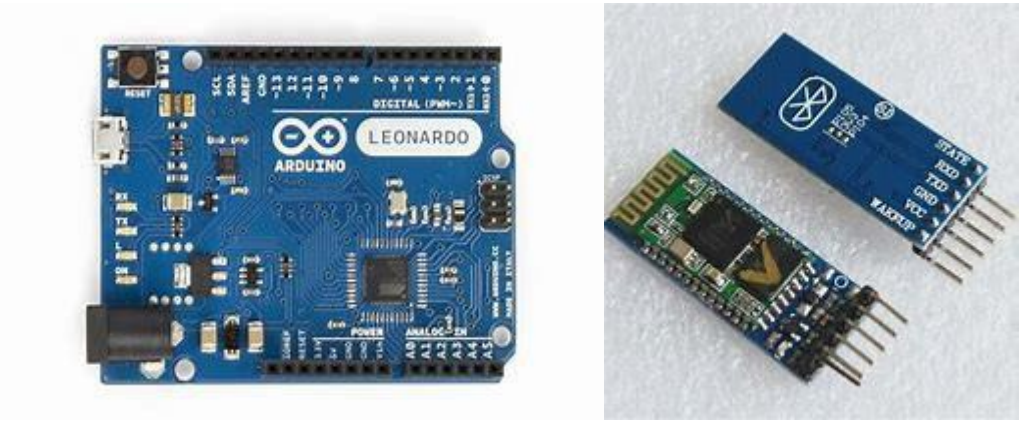
To accomplish this project using

- Arduino UNO
- HC-05 Bluetooth module
- 12 v 4 channel relay board
- Arduino IDE software

3. LITERATURE SURVEY

3.1 Bluetooth based home automation

- In Bluetooth based home automation systems the home appliances areconnected to the Arduino BT board at input output ports using relay.
- The program of the Arduino BT board is based on the high-levelinteractive C language of microcontrollers.
- The connection is made via Bluetooth. The password protection is provided so only authorized users are allowed to access the appliances.
- The Bluetooth connection is established between Arduino BT board andphone for wireless communication.
- In this system the python script is used and it can install on any of theSymbian OS environments, it is portable.
- One circuit is designed and implemented for receiving the feedback fromthe phone, which indicates the status of the device.



4. ARCHITECTURE

Here Architecture is of Arduino or precisely the IC of Arduino (ATmega328p). TheATmega328/P is a low-power CMOS 8-bit microcontroller based on the AVR®enhanced RISC (reduced instruction set computer) architecture. In Order to maximize performance and parallelism, the AVR uses Harvard architecture - with separate memories and buses for program and data. Instructions inthe program memory are executed with a single level of pipelining. The clock is controlled by an external 16MHz Crystal Oscillator.

The basic working of CPU of ATmega328:

(a) The data is uploaded in serial via the port (being uploaded from the computer's Arduino IDE). The data is decoded and then the instructions are sent to instruction register and it decodes the instructions on the same clock pulse.

(b) On the next clock pulse the next set of instructions are loaded in the instructionregister.

(c) In general purpose registers : the registers are of 8-bit but there are 3 16-bitregisters also.

- 8-bit registers are used to store data for normal calculations and results.
- 16-bit registers are used to store data from the timer counter in 2 different registers. Eg. X-low & X-high. They are fast, and are used to store specific hardware functions.

(d) EEPROM stores data permanently even if the power is cut out. Programminginside an EEPROM is slow.

(e) Interrupt Unit checks whether there is an interrupt for the execution of instruction to be executed in ISR (Interrupt Service Routine).

(f) Serial Peripheral Interface (SPI) is an interface bus commonly used to send databetween microcontrollers and small peripherals such as Camera, Display, SD cards, etc. It uses separate clock and data lines, along with a select line to choose the device you wish to talk to.

(g) Watchdog timer is used to detect and recover from MCU malfunctioning.

(h) Analog comparator compares the input values on the positive and negative pin,when the value of the positive pin is higher the output is set.

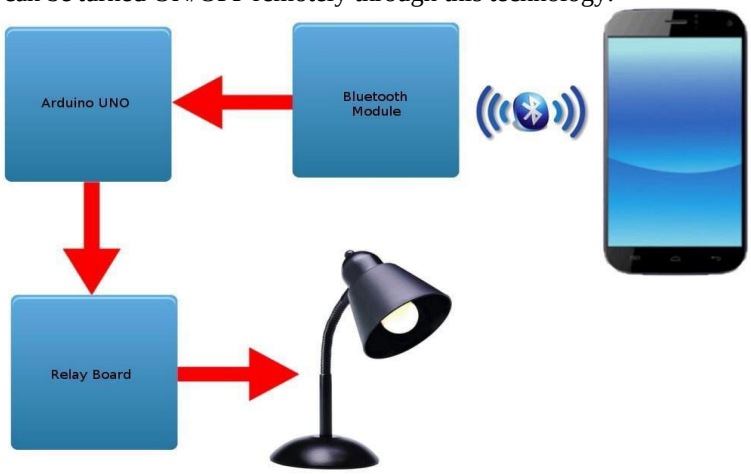
(i) Status and control are used to control the flow of execution of commands bychecking other blocks inside the CPU at regular intervals.

(j) ALU (Arithmetic and Logical unit): The high-performance AYR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations b/w general purpose registers are executed. The ALU operations are divided into 3 main categories - arithmetic, logical and bit-function.

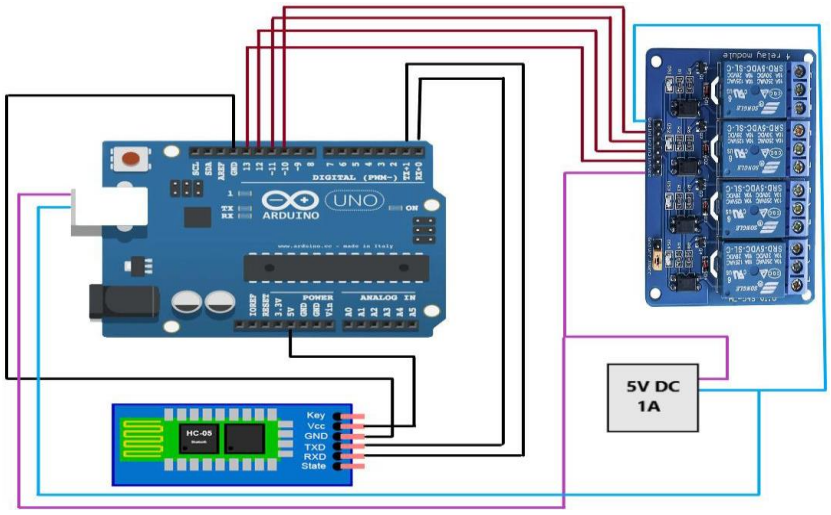
(k) 1/0 pins: The digital inputs and outputs (digital I/O) on the Arduino are what allow you to connect the Arduino sensors, actuators, and other ICs. Leaming how to use them will allow you to use the Arduino to do some really useful things, such as

4.1 Modules

4.1.1 Bluetooth module: A Bluetooth module is interfaced to the Arduino board at the receiver end while on the transmitter end, a GUI application on the cell phone sends ON/OFF commands tothe receiver where loads are connected. By touching the specified location on the GUI, the loads can be turned ON/OFF remotely through this technology.



4.1.2 Voice module: Uses android mobiles internal voice recognition to pass voice commands to yourrobot. Pairs with Bluetooth Serial Modules and sends in the recognized voice as astring. For example, if you say Hello the android phone will return a sting *Hello# to yourBluetooth module *and # indicate the start and stop bits Can Be used with any micro controller which can handle strings. Examples Platforms : Arduino , ARM , PICAXE , MSP430, 8051 based andmany other processors and controllers.



Block diagram of Home Automation using Bluetooth Module

5. ALGORITHMS AND PROGRAMMING

5.1 Arduino

```
String voice; #define relay1 2
#define relay2 3void setup()
{
Serial.begin(9600); pinMode(relay1, OUTPUT); pinMode(relay2, OUTPUT); digitalWrite(relay1, LOW); digitalWrite(relay2, LOW);
}
void loop()
{
while(Serial.available())
{
delay(10);
char c = Serial.read();if (c == '#'){
break;
}
voice += c;
}
if (voice.length() >0)
{
Serial.println(voice); if(voice == "*switch on"){switchon();
}
else if(voice == "*switch off"){switchoff();
```

```
 }

want...same applies to otherselse if(voice == "*fan on")
{

//You can replace 'lamp on' with anything you

digitalWrite(relay1, HIGH);
}
else if(voice == "*fan off"){ digitalWrite(relay1, LOW);
}
else if(voice == "*tube on"){digitalWrite(relay2, HIGH);
}

    else if(voice == "*tube off"){digitalWrite(relay2, LOW);
}
voice="";
}
}
void switchon() digitalWrite(relay1, HIGH);digitalWrite(relay2, HIGH);
}
void switchoff()
{
digitalWrite(relay1, LOW);digitalWrite(relay2, LOW);
}

/*

void..... ()
{
digitalWrite(..., LOW/HIGH);digitalWrite(..., LOW/HIGH);
}
```

6. CONCLUSION

In this project we have successfully implemented voice-controlled home automation system controlling relays using Arduino with Bluetooth module HC-05. This project can be used for controlling '4' number of input controls i.e., by extending number of relays. Our implemented module is more reliable and flexible in order to control any loads and the coverage area for wireless control is 10 meters. Hence this project can be useful for a real time voice-controlled home automation. Thus, Arduino based voice-controlled home appliances proves to be a better remote-controlled operation on home appliances using Bluetooth module HC-05. This project can be extended for many automation applications such as industrial automation, automotive, military, healthcare, transportation and so on. Further the coverage area can also be increased by the use of GSM modules.

7. REFERENCES

[1] https://www.geeksforgeeks.org/what-is-arduino /
[2] https://www.arduino.cc/en/Tutoria1/HomePage
[3] https://create.arduino.cc/projecthub/mayooghgirish/arduino-bluetooth-basic-t utorial-d8b737
[4] https://www.c-sharpcomer.com/article/turning-led-off-and-on-through-voice-re cognition/
[5] https://www.c-sharpcomer.com/article/tum ing-led-off-and-on-through-voice-recognition/
[6] https://ijarcce.com /upload /2017/march- l 7/IJARCCE%20173.pdf

INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

COPYRIGHT TRANSFER FORM

PaperID	V7I3-1702		
Title	Voice based home automation		
Author	**Contact**	**Designation**	**Organisation**
Abhinayreddy	abhinayreddy.17.cse@anits.edu.in 9121381008	Student	Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh
Niteesh Naidu	niteesh.17.cse@anits.edu.in 9502980837	Student	Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh
Satyanarayana	tsatyanarayana.l18.cse@anits.edu.in 8977169990	Student	Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh
Vivek	vivekpulidindi.17.cse@anits.edu.in 6301113405	Student	Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh
Ch. K. Rupesh Kumar	rupeshkumar.cse@anits.edu.in 9248771315	Assistant Professor	Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

1. I/We hereby declare that the material being presented by me/us in this paper is our original work, and does not contain or include material taken from other copyrighted sources. Wherever such material has been included, it has been clearly indented or/and identified by quotation marks and due and proper acknowledgements given by citing the source at appropriate places.
2. I/We hereby assign(s) to IJARIIT (International Journal of Advance Research, Ideas and Innovations in Technology) all rights under copyright that may exist in and to:
    1. the above Work, including any revised or expanded derivative works submitted to the IJARIIT by the me/us based on the Work; and
    2. any associated written or multimedia components or other enhancements accompanying the Work.
3. The paper, the final version of which I/We enclose, is not substantially the same as any that I/we have already published elsewhere.
4. I/We have not sent the paper or any paper substantially the same as the enclosed one, for publication anywhere else.
5. The submitted/enclosed paper is thoroughly proof read by me and in conformity with the instructions to authors communicated to me (as per IJARIIT submission guidelines).
6. I/We agree(s) to indemnify and hold harmless IJARIIT from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
7. In the event the above work is accepted and published by IJARIIT and consequently withdrawn by the author(s), the foregoing copyright transfer shall become null and void and all materials embodying the Work submitted to IJARIIT will be destroyed.
8. I/We, in consideration for his/her/their participation in the Journal, hereby grants the IJARIIT the unlimited, worldwide, irrevocable permission to use, distribute, publish, license, exhibit, record, digitize, broadcast, reproduce and archive, in any format or medium, whether now known or hereafter developed.
9. For authenticity, validity and originality of the research paper the author/authors will be totally responsible.

International Journal of Advance Research, Ideas and Innovations in Technology

Monday, Jun 07, 2021 (03:32:57)