A seminar report on

HOME APPLIANCER’S CONTROLLED BY BLUETOOTH TECHNOLOGY USING ARDUINO UNO

Submitted in partial fulfillment of the requirement for the award of the degree of

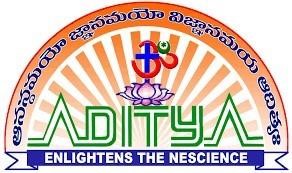
BACHELOR OF TECHNOLOGY

In

# ELECTRICAL AND ELECTRONICS ENGINEERING

By

M. L. R. Tulasi 21P31A0223



Department of Electrical and Electronics Engineering

# ADITYA COLLEGE OF ENGINEERING AND TECHNOLOGY

(Affiliated to JNTUK, Kakinada & Approved by AICTE, New Delhi)

Aditya Nagar, ADB Road, Surampalem – 533437

2023

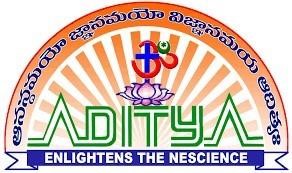
CERTIFICATE

Department of Electrical and Electronics Engineering

### ADITYA COLLEGE OF ENGINEERING AND TECHNOLOGY

(Affiliated to JNTUK, Kakinada & Approved by AICTE, New Delhi)

Aditya Nagar, ADB Road, Surampalem – 533437



This is certify that the seminar report entitled “HOME APPLIANCERS COTROLLED BY BLUETOOTH TECHNOLOGY USING ARDUINO UNO” being submitted by M.L.R Tulasi 21P31A0223 in the partial fulfillment of the requirement for the award of the Degree of Bachelor of Technology in ELECTRICAL AND ELECTRONICS ENGINEERING from Jawaharlal Nehru Technological University Kakinada. It is a record of the bonafide work carried out by him at Aditya college of Engineering.

Head of the Department

Mr.M Rajesh(Ph.D)

Assoc. Prof, Depaertment of EEE

INDEX

**S.NO. CHAPTER NAME PAGE NO**

**ABSTRACT**

**1 Introduction**

**2 Required components &description**

**2.1 Components required**

2.1.1 Arduino Uno

2.1.2 Channel relay (5v)

2.1.3 Bluetooth module (HC05)

2.1.4 Power supply

2.1.5 Load (BULB 220V)

2.1.6 Connecting wires

2.1.7 Vero board

2.1.8 Smart phone (Bluetooth enable)

**2.2 Description of the components**

**3 Connection Diagrams**

3.1 Block diagram

3.2 circuit diagram

**4 design and implementation**

4.1 Dumping of code

4.2 Controlling from application

**5 Conclusion**

**Abstract**

The purpose of this project is to control the home appliances by using Bluetooth technology using Arduino Uno. In every house there are several electrical appliances are using but sometimes the electrical load cannot be turned off so lot of electrical energy can be wasted and, in some houses, there are aged people so they cannot go to wall switches to turn on or turn off at that situation this project will helpful. As technology is advancing so house is also getting smarter modern houses house are gradually shifting from conventional switches to centralized control system a Bluetooth module is interfaced to the Arduino Uno board at the receiver end while on the transmitter side controlling from the application on the cell phone sends the command to the receiver where the loads are connected. The proposing outcomes of this project is controlling of home appliance’s with In the Bluetooth range.

**Introduction**

Nowadays, we have remote controls for our television sets and other electronic systems, which have made our lives really easy. Have you ever wondered about home automation which would give the facility of controlling tube lights, fans and other electrical appliances at home using a remote control? Off-course, yes! But, are the available options cost-effective? If the answer is No, we have found a solution to it. We have come up with a new system called Arduino based home automation using Bluetooth. This system is super-cost effective and can give the user, the ability to control any electronic device without even spending for a remote control. This project helps the user to control all the electronic devices using his/her Smartphone. Time is a very valuable thing. Everybody wants to save time as much as they can. New technologies are being introduced to save our time. To save people’s time we are introducing Home Automation system using Bluetooth. With the help of this system, you can control your home appliances from your mobile phone. You can turn on/off your home appliances within the range of Bluetooth.

**Applications:**

The most common applications of home automation are

1. lighting control
2. HVAC (Heating, Ventilation, and Air Conditioning)
3. kitchen appliances
4. security systems

**Advantages:**

1. Managing all of your home devices from one place. The convenience factor here is enormous
2. Flexibility for new devices and appliances
3. Maximizing home security
4. Remote control of home functions
5. Increased energy efficiency
6. Improved appliance functionality
7. Home management insights

**Disadvantages:**

1. It can lose connection in certain conditions. It has low bandwidth as compared to Wi-Fi. It allows only short range communication between devices. Security is a very key aspect as it +can be hacked.
2. Maintenance and repair issues

**3. Required components &description**

**3.1 Components required**

3.1.1 Arduino Uno

3.1.2 Channel relay (5v)

3.1.3 Bluetooth module (HC05)

3.1.4 Power supply

3.1.5 Load (BULB 220V)

3.1.6 Connecting wires

3.1.7 Vero board

3.1.8 Smart phone (Bluetooth enable)

**3.2 Description of the components**

**Arduino Uno:**

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 tool chains, the Arduino project provides an integrated development environment (IDE) based on the Processing language project.



Arduino uno

**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: 40ma

DC Current for 3.3V Pin: 50ma

Flash Memory: 32 KB of which 0.5 KB used by boot loader

SRAM: 2 KB (ATmega328)

EEPROM: 1 KB (ATmega328)

Clock Speed: 16 MHz

**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 tool chain, also included with the IDE distribution. The Arduino IDE employs the program argued 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.

**Sketch**

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. ion. Arduino Software (IDE) pre-1.0 saved sketches with the extension. pde.

A minimal Arduino C/C++ program consists 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.

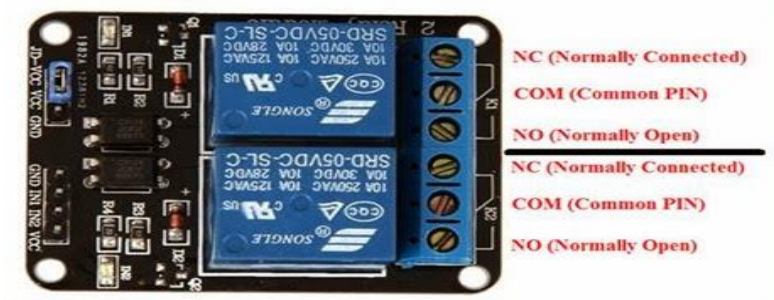
loop (): After setup () has been called, function loop () is executed repeatedly in the main program. It controls the board until the board is powered off or is reset.

**Channel relay:**

We can control high voltage electronic devices using relays. A Relay is actually a switch which is electrically operated by an electromagnet. The electromagnet is activated with a low voltage, for example 5 volts from a microcontroller and it pulls a contact to make or break a high voltage circuit

As an example, for this Arduino Relay Tutorial, we will use the HL-52S 2 channel relay module, which has 2 relays with rating of 10A @ 250 and 125 V AC and 10A @ 30 and 28 V DC. The high voltage output connector has 3 pins, the middle one is the common pin and as we can see from the markings one of the two other pins is for normally open connection and the other one for normally closed connection.

On the other side of the module, we have these 2 sets of pins. The first one has 4 pins, a Ground and a VCC pin for powering the module and 2 input pins In1 and In2. The second set of pins has 3 pins with a jumper between the JDVcc and the Vcc pin. With a configuration like this the electromagnet of the relay is directly powered from the Arduino Board and if something goes wrong with the relay the microcontroller could get damaged.

****

Relay module

**Bluetooth module (HC05):**

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 Blue core 04‐ External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature).

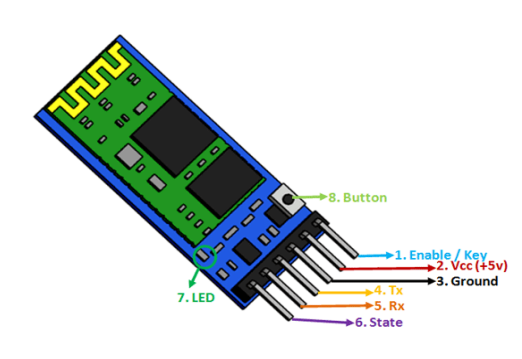
Bluetooth Module HC-05

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.

****

Bluetooth module

**Pin description of Bluetooth module:**

****

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 on and communication also takes place.

**VCC:**

Supply Voltage 3.3V to 5V

**GND;**

Ground

**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 another 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

**4. Connection Diagrams**

**4.1 Block diagram**

Relay module

Arduino uno

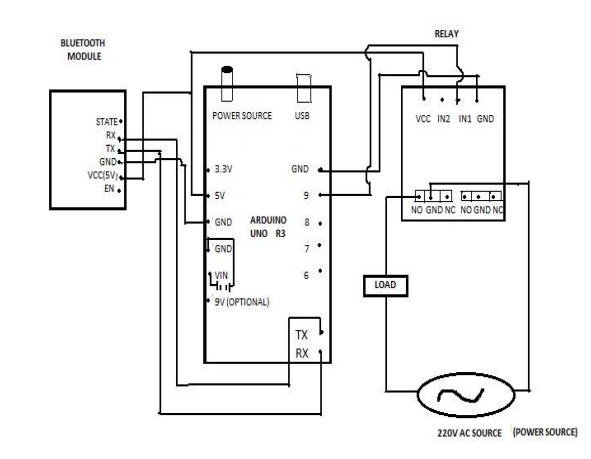
Bluetooth module

Load

Main power source(220v)

9v Power source

**4.2 Circuit diagram:**

****

This is the circuit diagram by using this we have to connect all the components and after the home automation using Bluetooth technology is ready then we have to connect the home appliances to this circuit and the load will operate by this technology by using an android mobile

**5 Design and implementation**

**5.1 Dumping of code:**

By using the ide software, we are dump the code into the Arduino uno board if the load will increase the code will also increase based on the load switching the coding will change so accordingly the working will done.

/\*

Relay IN1 connected to Pinout 2 Arduino

Relay IN2 connected to Pinout 3 Arduino

Relay IN3 connected to Pinout 4 Arduino

Relay IN4 connected to Pinout 5 Arduino

--->you can connect to relay module 4 channel

Serial data sending from Arduino Bluetooth Relay 4CH.apk data

'1'-'4' toon is Relay CH 1-4

data 'A'-'D' to off is Relay CH 1-4

data '9' to on ALL CH 1-4

data 'I' to off ALL CH 1-4

\*/

#include<SoftwareSerial.h>

Software Serial my Serial (10, 11); //Pin10 RX, Pin 11 TX connected to--> Bluetooth TX, RX

#define relay1 2

#define relay2 3

#define relay3 4

#define relay4 5

char Val;

void setup () {

pin Mode (relay1, OUTPUT);

pin Mode (relay2, OUTPUT);

pin Mode (relay3, OUTPUT);

pin Mode (relay4, OUTPUT);

digital Write (relay1, HIGH);

digital Write (relay2, HIGH);

digital Write (relay3, HIGH);

digital Write (relay4, HIGH);

mySerial.egin (9600);

Serial. Begin (9600);

}

void loop () {

//check data serial from Bluetooth android App

If (mySerial.available () >0) {

Val = mySerial.read ();

Serial.println(val);

}

//relay is on

If (Val == ‘1’) {

digital Write (relay1, LOW);}

else if (Val == '2’) {

digital Write (relay2, LOW);}

else if (Val == '3’) {

digital Write (relay3, LOW);}

else if (Val == '4’) {

digital Write (relay4, LOW);}

//relay all on

else if (Val == '9’) {

digital Write (relay1, LOW);

digital Write (relay2, LOW);

digital Write (relay3, LOW);

digital Write (relay4, LOW);}

//relay is off else

if (Val == 'A’) {

digital Write (relay1, HIGH);}

else if (Val == 'B’) {

digital Write (relay2, HIGH);}

else if (Val == 'C’) {

digital Write (relay3, HIGH);}

else if (Val == 'D’) {

digital Write (relay4, HIGH);}

//relay all off

else if (Val == 'I’) {

digital Write (relay1, HIGH);

digital Write (relay2, HIGH);

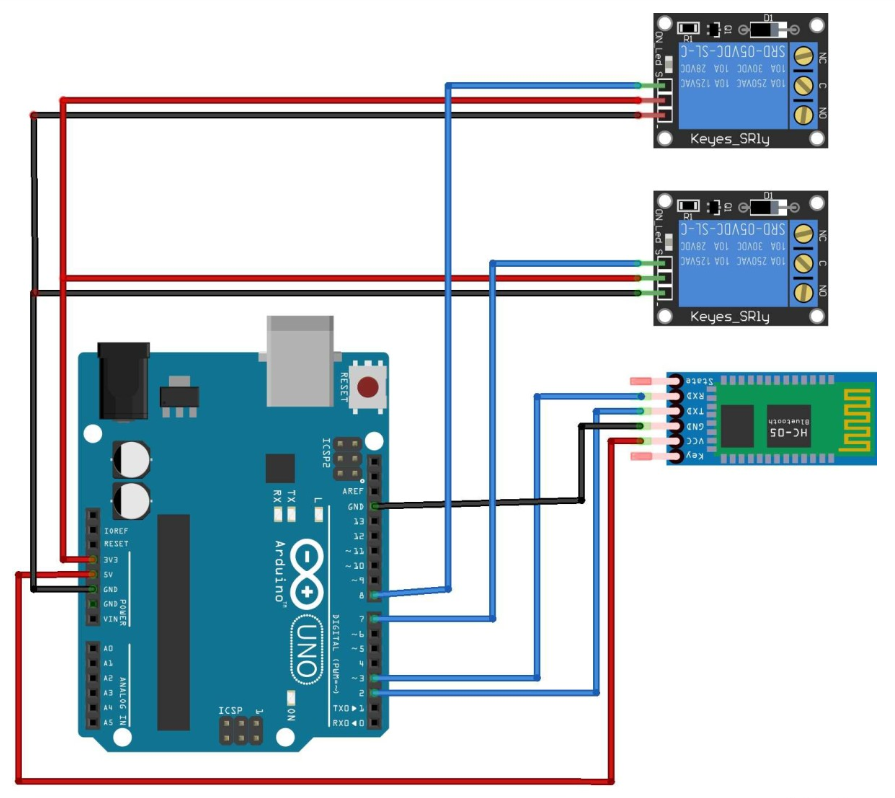
digital Write (relay3, HIGH);

digital Write (relay4, HIGH);}

}

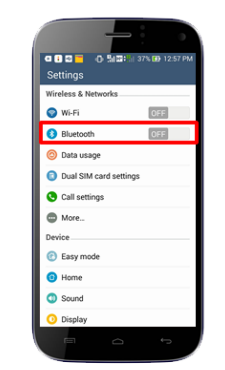
**5.2 Controlling from application:**

After design and installing the code to the Arduino we have to controlling the load from the application in android mobile so for that we have to connect the Bluetooth after the application interface will be

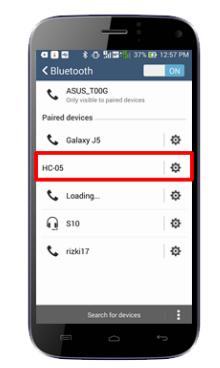


Design of the circuit

* Initially we have to turn on Bluetooth
* Turn on your Bluetooth module from your Arduino board
* Open[settings] menu on your android device and turn on Bluetooth



* Open [Bluetooth settings], search Bluetooth module device HC-05…etc and pair with default password



* Open Bluetooth app control for Arduino
* Select Bluetooth device
* Click to connect button
* Try click button to TURN ON/TURN OFF on your lamp with Bluetooth communication’s



**6.Conclusion**

The system as the name indicates, ‘Home automation’ makes the system more flexible and provides attractive user interface compared to other home automation systems. In this system we integrate mobile devices into home automation systems. A novel architecture for a home automation system is proposed using the relatively new communication technologies. The system consists of mainly three components is a BLUETOOTH module, Arduino microcontroller and relay circuits. WIFI is used as the communication channel between android phone and the Arduino microcontroller. We hide the complexity of the notions involved in the home automation system by including them into a simple, but comprehensive set of related concepts. This simplification is needed to fit as much of the functionality on the limited space offered by a mobile device’s display. This paper proposes a low cost, secure, ubiquitously accessible, auto-configurable, remotely controlled solution. The approach discussed in the paper is novel and has achieved the target to control home appliances remotely using the Wi-Fi technology to connects system parts, satisfying user needs and requirements. Wi-Fi technology capable solution has proved to be controlled remotely, provide home security and is cost-effective as compared to the previously existing systems. Hence we can conclude that the required goals and objectives of home automation system have been achieved. The system design and architecture were discussed, and prototype presents the basic level of home appliance control and remote monitoring has been implemented. Finally, the proposed system is better from the scalability and flexibility point of view than the commercially available home automation systems.