VELLORE INSTITUTE OF TECHNOLOGY, VANDALUR-KELAMBEKKAM ROAD, CHENNAI, TAMIL NADU-48

BLUETOOTH BASED HOME AUTOMATION SYSTEM

A PROJECT BASED LEARNING REPORT
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY
(Computer Science)

SUBMITTED FOR

EMBEDDED SYSTEMS

SUBMITTED BY

Name of Student(s)	Registration No.
1. VISHAL CHAUHAN	13BCE1171
2. HARIT PATHAK	13BCE1049
3. YASHKIRATH	13BCE1177

SUPERVISED BY and SUBMITTED TO
Prof. SRIDHAR R

)

TABLE OF CONTENTS

S.No.	Title	Pg. No.
1.	Abstract	
2.	Introduction	
3.	Flowchart	
4.	Literature Survey i. Triac BT136 ii. Optocoupler MOC3021 iii. Bluetooth Module HC-05 iv. The Android Application	
5.	Circuit Diagram	
6.	Arduino Coding	
7.	Hardware Interfacing	
8.	Conclusion & Future Scope	

ABSTRACT

Suppose you're ill or having a nap at home. The weather outside has gone from bad to worse and you feel the chill inside your home. And as often as always the switchboard seems to be a mile away. So because you're ill (or just lazy) you just choose to let things be. But wait what if things would be different?

Welcome to the concept of home automation and control. Automated systems are today beginning to replace humans in a plethora of jobs from high precision wielding to robotic mass assembly.

Then why not in our daily lives?

The concept we put forward here in this project is one with immense future scope and business prospects. We've all seen Tony Stark spit out voice command to his omnipresent super-computer companion Jarvis to carry out multiple tasks at the same time. Although at present, our knowledge limits our horizons, our project will allow its user to seamlessly control all electrical appliances through a simple and intuitive mobile application that we have natively built for the Android OS but can be ported to other popular platforms like IOS and Windows Phone with minimal effort.

Needless to say, our app also has the ability to take voice input with the help of Google-Speech-to-Text Engine or Siri (on IOS) or Cortana(on Windows) and there by perform the same functions as the graphical interface.

Hop on-board as we walk you through the steps to make a Wireless Home Automation System

.

INTRODUCTION

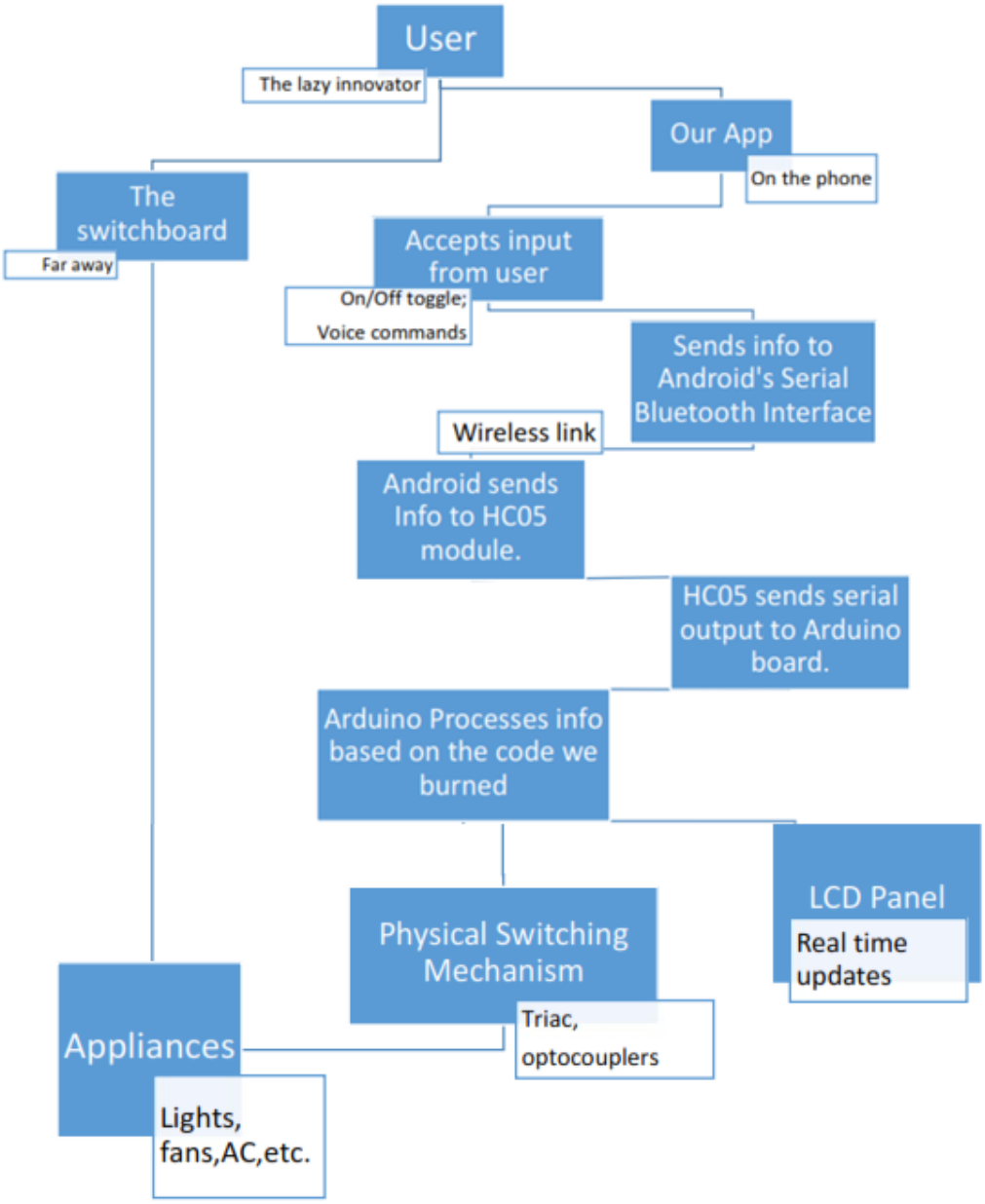
In this project we implemented a prototype system with the help of a prefabricated and commercially available Microcontroller Unit. With our knowledge expanding day-by-day, we hope to design and implement (and possibly even mass produce) a bolt on-and-use kind of device that we envision will someday replace chunky old switchboards.

List of Parts :

1. Arduino Uno MCU
2. Bluetooth Serial Module HC-05
3. Resistors
4. LCD
5. Triac and Optocoupler
6. LCD display panel
7. Potentiometer
8. Breadboard
9. Wires.

The microcontroller is the heart and brain of the system. We program the required commands using the software available on www.arduino.cc and burn it onto the MCU. The Arduino and the mobile app communicate via a serial Bluetooth link. The HC05 Bluetooth module does the job of accepting the input from the phone's Bluetooth and transferring it to the Arduino. The LCD display provides real time updates of the changes at the switching end.

FLOWCHART


User
The lazy innovator
Our App
On the phone
The switchboard
Far away
Accepts input from user
On/Off toggle;
Voice commands
Sends info to Android's Serial Bluetooth Interface
Wireless link
Android sends Info to HC05 module.
HC05 sends serial output to Arduino board.
Arduino Processes info based on the code we burned
LCD Panel
Real time updates
Physical Switching Mechanism
Triac, optocouplers
Appliances
Lights, fans,AC,etc.

LITERATURE SURVEY

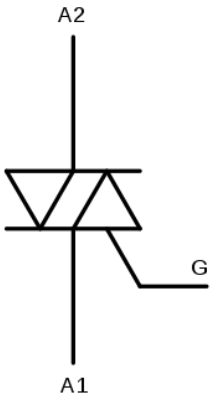
Triac – **BT136**

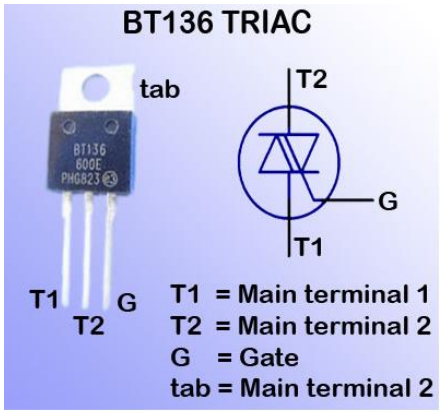
Basic Description:

TRIAC, from **triode for alternating current**, is a generic trademark for a three terminal electronic component that conducts current in either direction when triggered. Its formal name is, **bidirectional triode thyristor** or **bilateral triode thyristor**. A thyristor is analogous to a relay in that a small voltage and current can control a much larger voltage and current. The illustration on the right shows the circuit symbol for a TRIAC where A1 is Anode 1, A2 is Anode 2, and G is Gate. Anode 1 and Anode 2 are normally termed Main Terminal 1 (MT1) and Main Terminal 2 (MT2) respectively.

TRIACs are a subset of thyristors and are related to silicon controlled rectifiers (SCRs). However, unlike SCRs, which are unidirectional devices and only conduct current in one direction, TRIACs are bidirectional and conduct current in both directions. Another difference is that SCRs can only be triggered by a positive current at their gate, but, in general, TRIACs can be triggered by either a positive or negative current at their gate, although some special types cannot be triggered by one of the combinations. To create a triggering current for an SCR a positive voltage has to be applied to the gate but for a TRIAC either a positive or negative voltage can be applied to the gate. In all three cases the voltage and current are with respect to MT1. Once triggered, SCRs and thyristors continue to conduct, even if the gate current ceases, until the main current drops below a certain level called the holding current.

Terminal Layout:





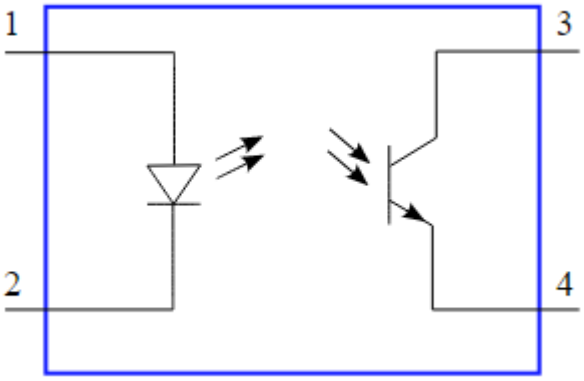
Opto-Coupler – **MOC3021**

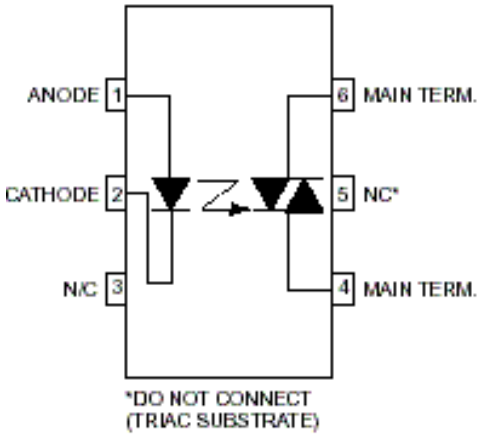
Basic Description:

In electronics, an **opto-isolator**, also called an **optocoupler**, **photocoupler**, or **optical isolator**, is a component that transfers electrical signals between two isolated circuits by using light. Opto-isolators prevent high voltages from affecting the system receiving the signal. Commercially available opto-isolators withstand input-to-output voltages up to 10 kVvoltage transients with speeds up to 10 kV/μs.

A common type of opto-isolator consists of an LED and a phototransistor in the same opaque package. Other types of source-sensor combinations include LED-photodiode, LED-LASCR, and lamp-photoresistor pairs. Usually opto-isolators transfer digital (on-off) signals, but some techniques allow them to be used with analog signals.

Terminal Layout:





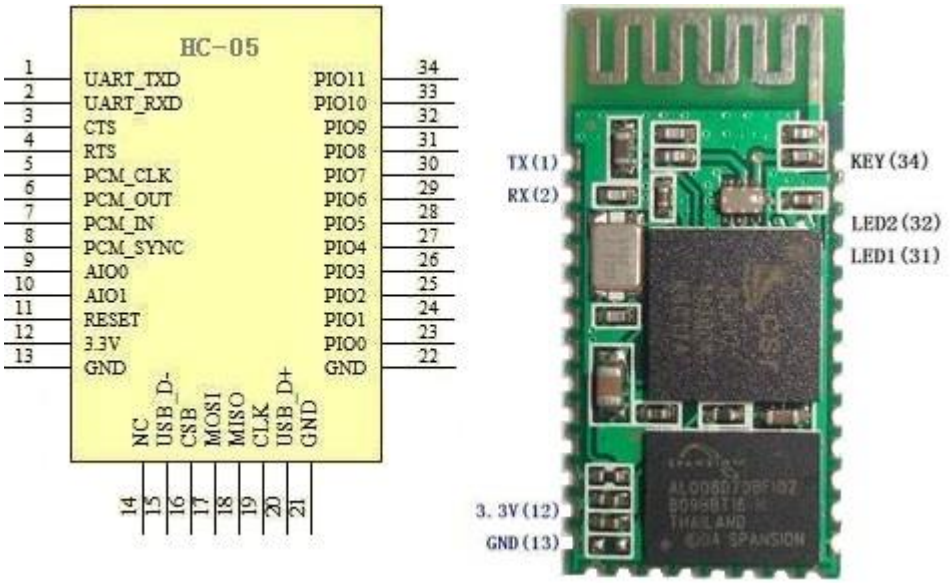
Bluetooth Module – **HC05**

Basic Description:

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

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 Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

Terminal Layout:



The **INTERSTELLAR** App

The App Details:

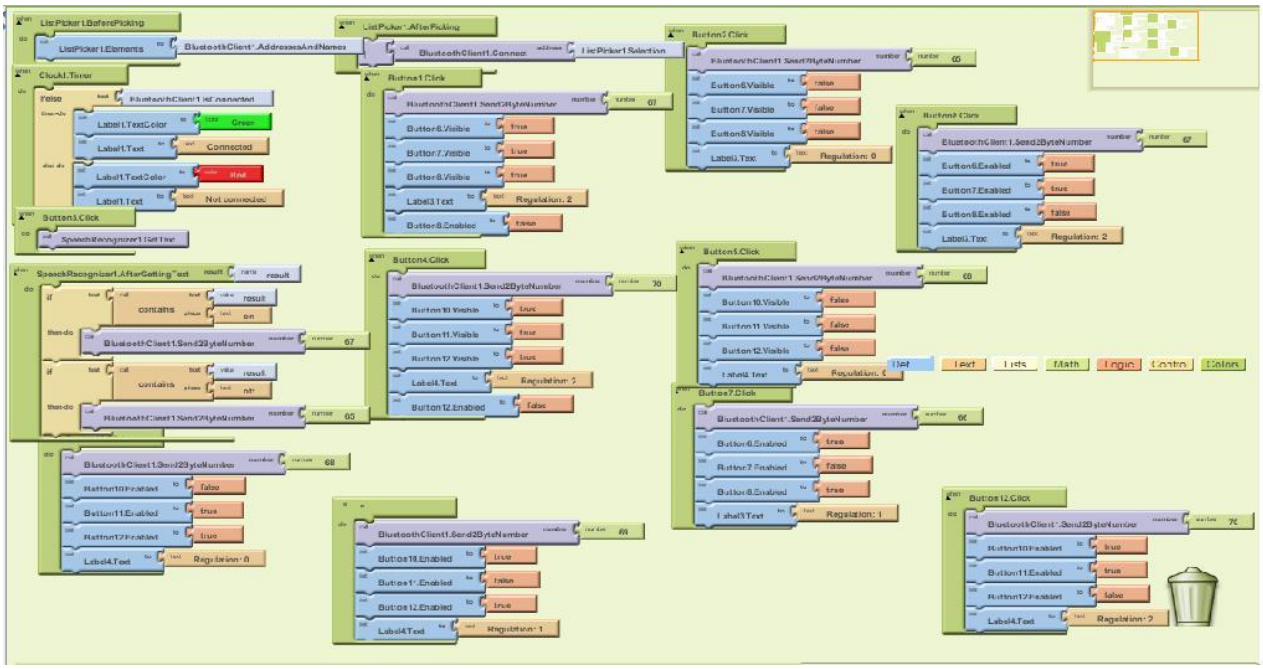
The Android app was made using an open source software called MIT- App Inventor. It's an online tool that allows students with minimal experience in Java to program a wide variety (though not all) of applications.

The interface consists mainly of buttons. There are two switching buttons for BULB. As the device has to first connect to the HC05 Bluetooth module to send any sort of data, we thought it would be handy if we could connect the module through the app itself. Hence the first button leads to the Bluetooth connection interface through the app. The regulation buttons are programmed to pop up only when the BULS's switched on.

The App Logic :

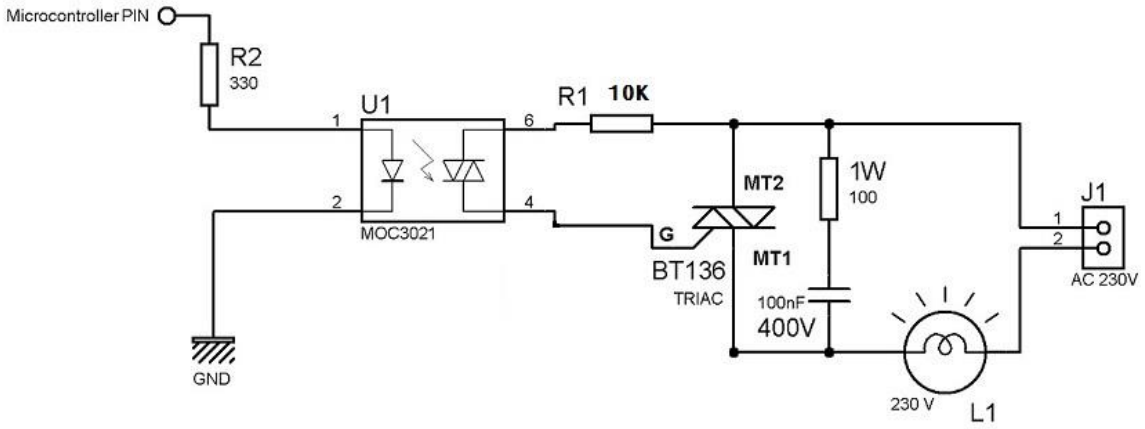
http://beta.appinventor.mit.edu

The logic of the app in MIT-AppInventor

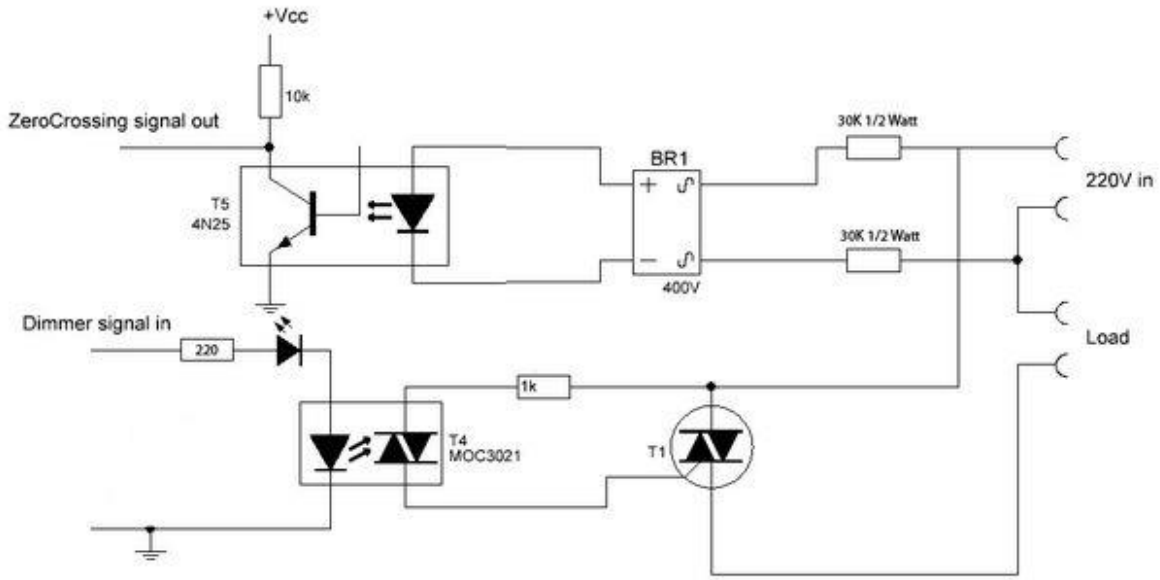


EXISTING CIRCUIT AND OUR PROPOSED CIRCUIT

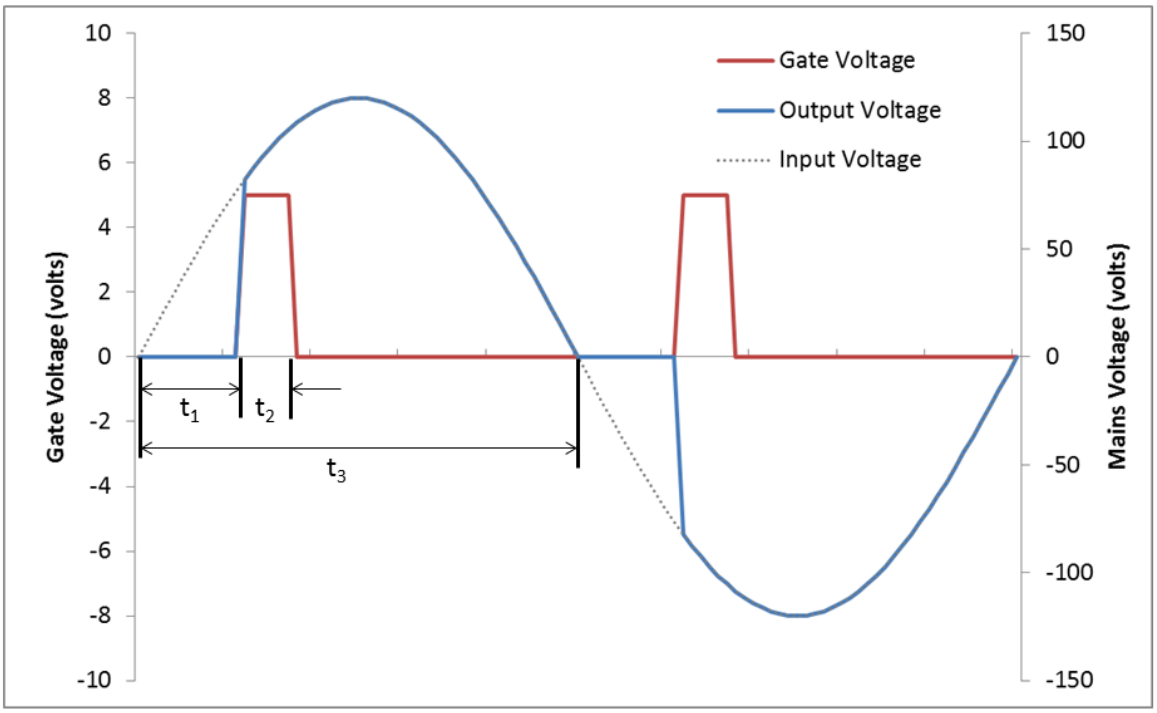
Existing Circuit



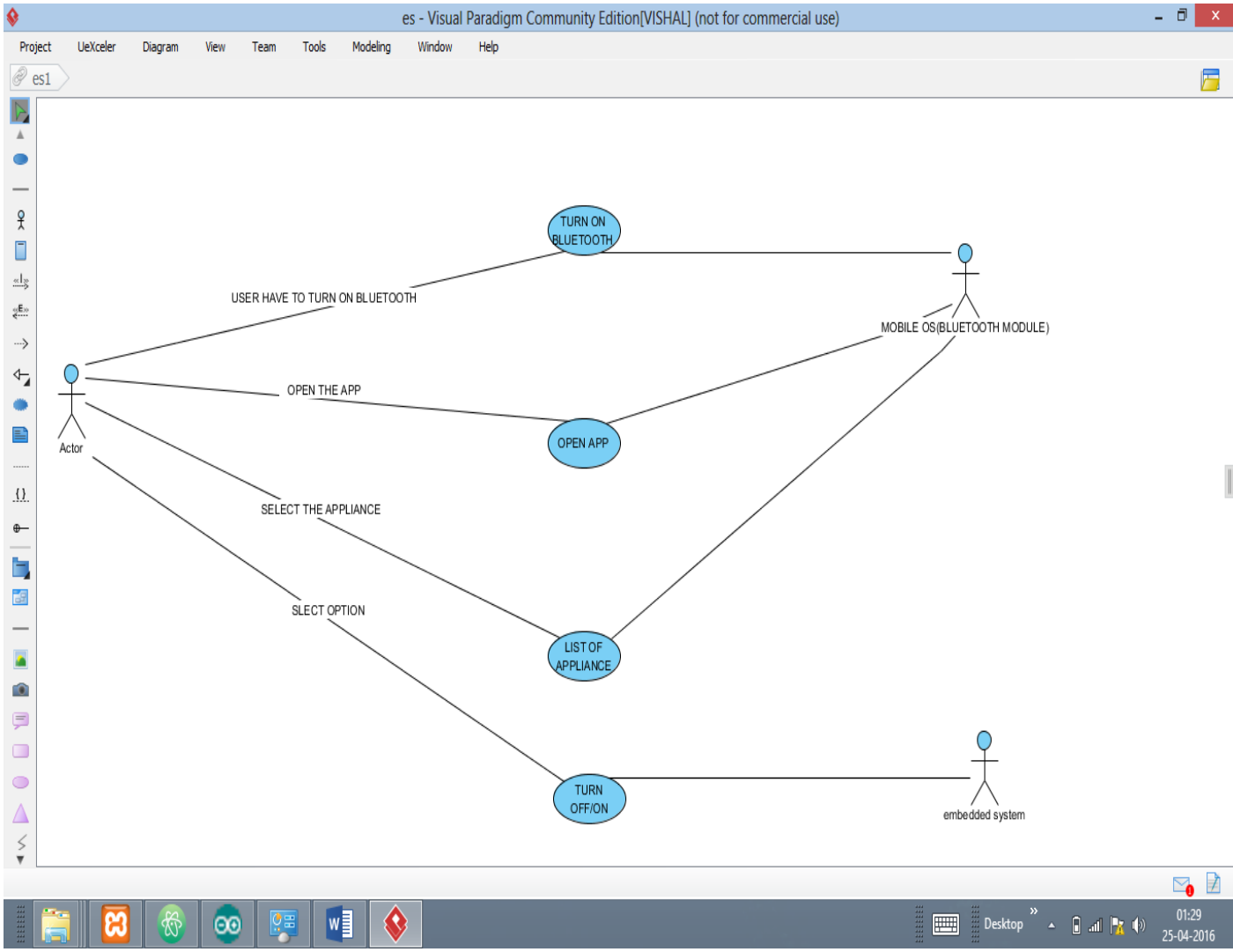
Our Proposed Circuit



SIMULATION OF OUR PROPOSED CIRCUIT



USE CASE DIAGRAM



ARDUINO CODING

```
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>


//SoftwareSerial BTSerial(0,1); // RX | TX

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
int ledp = 6;
//char state = '0';
//int flag = 0;
void setup() {
  pinMode(ledp, OUTPUT);// put your setup code here, to run once:
  Serial.begin(9600);
  Serial.print("We are ready          :\n");
    lcd.begin(16, 2);
  lcd.print("INTERSTELLAR            ");
  lcd.setCursor(0,1);
  lcd.print("TECHNOLOGIES              ");
  delay(3500);
  lcd.setCursor(0,0);
  lcd.print("PRESENTS                ");
  lcd.setCursor(0,1);
  lcd.print("                  ");
```

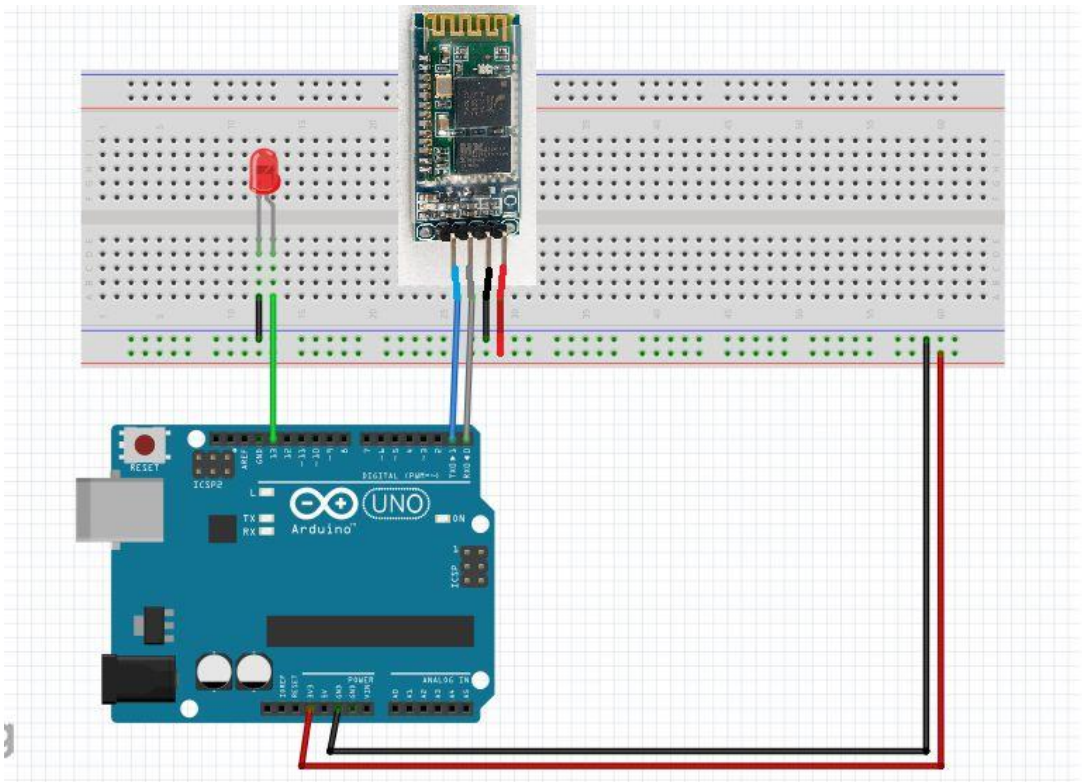
```
  delay(3000);
  lcd.setCursor(0,0);
  lcd.print("REGULATION of");
  lcd.setCursor(0,1);
  lcd.print("HOME APPLIANCES");
  delay(4500);
}

void loop() {

  /*analogWrite(ledp, 0);
  delayMicroseconds();
  analogWrite(ledp, 255);
  delayMicroseconds(87000);
  */




    if( Serial.available() > 0)       // if data is available to read
  {
     //Serial.write(Serial.read());
     char state = Serial.read();
```

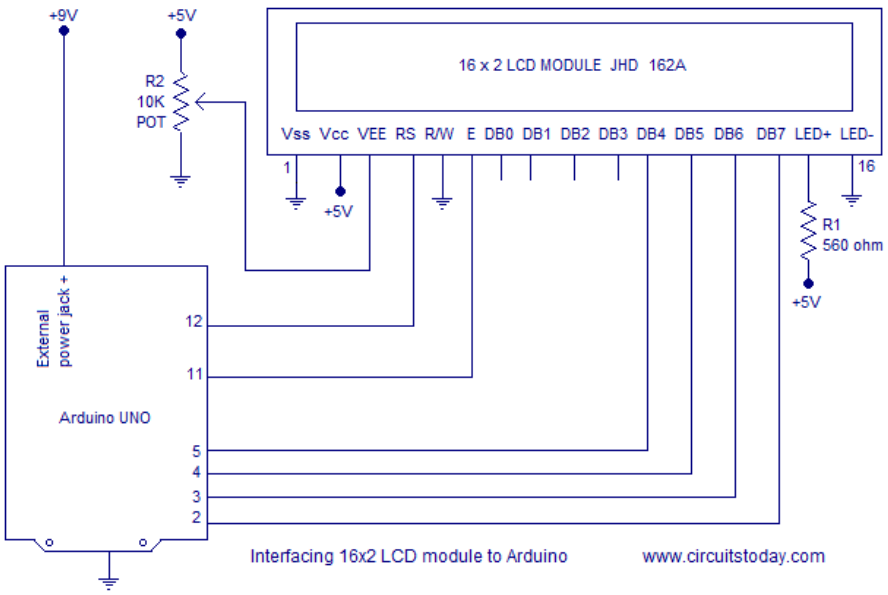


```
    delay(300);
```

```
  Serial.println("you have a connection");
  Serial.println(state);

  if( state == 'A' )              // if '65' was received led 13 is switched off
  {
    analogWrite(ledp,0);    // turn Off pin 13 off
    delay(1000);                 // waits for a second
    Serial.println("light is off");
  }

  if( state == 'F' )              // if '1' was received led 13 is switched on
  {
    analogWrite(ledp, 255);  // turn ON pin 13 on
    delay(1000);                 // waits for a second
    Serial.println("light is on");
  }
 }// put your main code here, to run repeatedly:

}
```

HARDWARE DEVELOPING





CONCLUSION & FUTURE SCOPE

As discussed before, this is a project with immense future scope. We see it as the first step to building intelligent and automated environments around our daily lives. In the future, we plan on implementing the entire circuit on a PCB board, add equipment's for inductive and capacitive loads. We also aim to enhance the app for better control.