



Bluetooth Based Home Automation

NIKITA WANJALE¹, REJOY MATHEWS², BLOSSOM MENDES³, MANALI NAVALE⁴

¹Dept of Instrumentation and Control, Vishwakarma Institute of Technology, Pune, India, E-mail: nikitawanjale@gmail.com.

²Dept of Instrumentation and Control, Vishwakarma Institute of Technology, Pune, India, E-mail: ejoymathews32@gmail.com.

³Dept of Instrumentation and Control, Vishwakarma Institute of Technology, Pune, India, E-mail: blossomendes@hotmail.com.

⁴Dept of Instrumentation and Control, Vishwakarma Institute of Technology, Pune, India, E-mail: manalinavale@rediffmail.com.

Abstract: Bluetooth based Home Automation is no more in the nascent phase. Home Automation refers to controlling of electronic appliances in the house using one or more masters and any communication medium like Bluetooth, RS 232, Ethernet, etc. Choice of communication medium is very crucial when it comes to home automation. Factors such as range, security, data rate, cost and accessibility need to be considered while choosing a medium. Nowadays Bluetooth is available in almost all the phones, making it a perfect medium for wireless communication between a mobile phone and a central controller. This paper describes use of Bluetooth communication to automate various appliances in the house through a mobile phone. Any controller can be used for this purpose. In this paper, Arduino Uno is used for its ease of programming.

Keywords: Android Application, Arduino Uno R3 Microcontroller, Bluetooth Communication, Home Automation.

I. INTRODUCTION

Home automation has been booming in the current years. In this paper, we have shown that various electronics appliances are controlled using Bluetooth one in the house. Having central control is one of the main features of home automation. Establishing communication between the operator interface hardware and the central controller has been a topic of debate based on the various parameters that go into deciding a particular communication protocol. Being wireless, Bluetooth provides for an effective communication protocol.

II. SOFTWARE IMPLEMENTATION AND DESIGN

A. Arduino code logic and algorithm

The Microcontroller searches for Serial data to be available on the Tx pin of the BTBee Pro shield. Serially transmitted data format is defined as follows:

*variable1|variable2|variable3# (1)

Inbuilt function Serial.parseInt(); is used to parse the incoming serial data(i.e. the data before the pipe(|) and store it in a variable. The parsing is done thrice to store the 3 parts of the serially transmitted data in three different variables. The variable "variable1" is used to give commands to the Arduino to perform particular function (e.g. Digital Write, Analog Write, and Curtain Control). The variable "variable2" is used to define the pin number to which a particular load is connected. For example if the light has to be turned on or off and the light circuit gets input from pin number 13 of the Arduino then the value of variable variable2 will be 13. Variable "variable3" is used to give the value to be outputted onto the microcontroller pin selected by variable2.

Fan control: If the variable1 = 10 (for digital control), and variable2 = 12, then the fan control function code is implemented. The fan is switched using relay which gets its input from the 12th pin of the Arduino Microcontroller. Variable3 decides whether the fan must be turned ON or OFF.

Light control: If the variable1 = 10(for digital control}, and variable2 = 13, then the light control function code is implemented. The light is switched using relay which gets its input from the 13th pin of the Arduino Microcontroller. Variable3 decides whether the curtain must be turned ON or OFF.

Curtain Control: If the variable1 = 15 the curtain control function code is implemented. The curtains are rotated in either direction using a motor driver circuit, which gets its input from the 7th and 10th pins of the Arduino Microcontroller. The value of Variable3 decides whether the curtain must be opened or closed.

Digital Control: If the variable1 = 10 (for digital control), and variable2 = 11, then the LED strip digital control function code is implemented. The LED strip is either switched on or off depending on the value of variable3.

LED Strip: LED strip control function may work in 3 different ways.

Analog Control: If the variable1 = 11 (for analog control), and variable2 = 11, then the LED strip analog control function code is implemented. The LED strip intensity is

changed using Pulse Width Modulation (PWM) with the help of an n-channel MOSFET. The variable3 in this case is the analog value required.

Party Mode: If the variable1 = 14, variable2 = 11, then the LED strip is controlled to create various lighting effects such as fast blink, slow blink, fade in fade out. Here, variable3 is used to choose the lighting effect required.

Status Query: A person can also know the status of an Arduino pin through Bluetooth. If variable1 = 13, status query function code is implemented. Variable2 is set as the pin number whose value is required. In this code, variable3 does not play any role. The Arduino board reads the status of the pin enquired and sends 0 or 1 to the phone as a response. If '0' is received, the status is indicated as 'OFF'. If '1' is received, the status is indicated as 'ON'.

B. Android Application

Fig.1.(a) shows a phone application which is used to communicate with the Arduino board. The cell phone application is developed using MIT AppInventor. The major

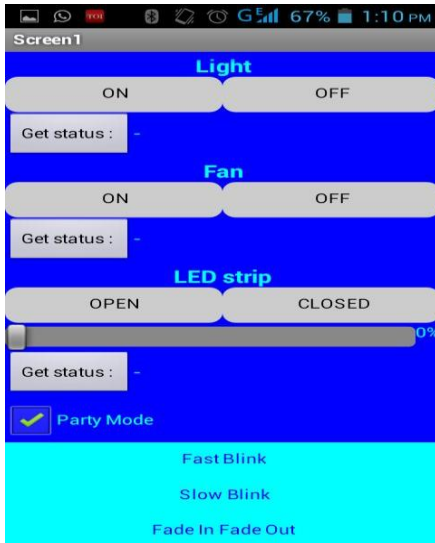


Fig.1.(a) Application interface.



Fig.1.(b) List of paired bluetooth devices.

advantage of using this open-source platform instead of java is that it is extremely user friendly and easy to program. Since this project is a prototype for a real automated house, the software needs to be easy to understand and modify. Fig.1.(b) shows the list of paired bluetooth devices which is displayed in the application enabling a user to choose a Bluetooth device to connect to. The Bluetooth module needs to be paired manually before using the application. A person can choose the address of the Bluetooth module in order to connect with it and then communicate with the Arduino board. Note that only one phone can be connected to the bluetooth module at once.

C. Programming of the Android Application

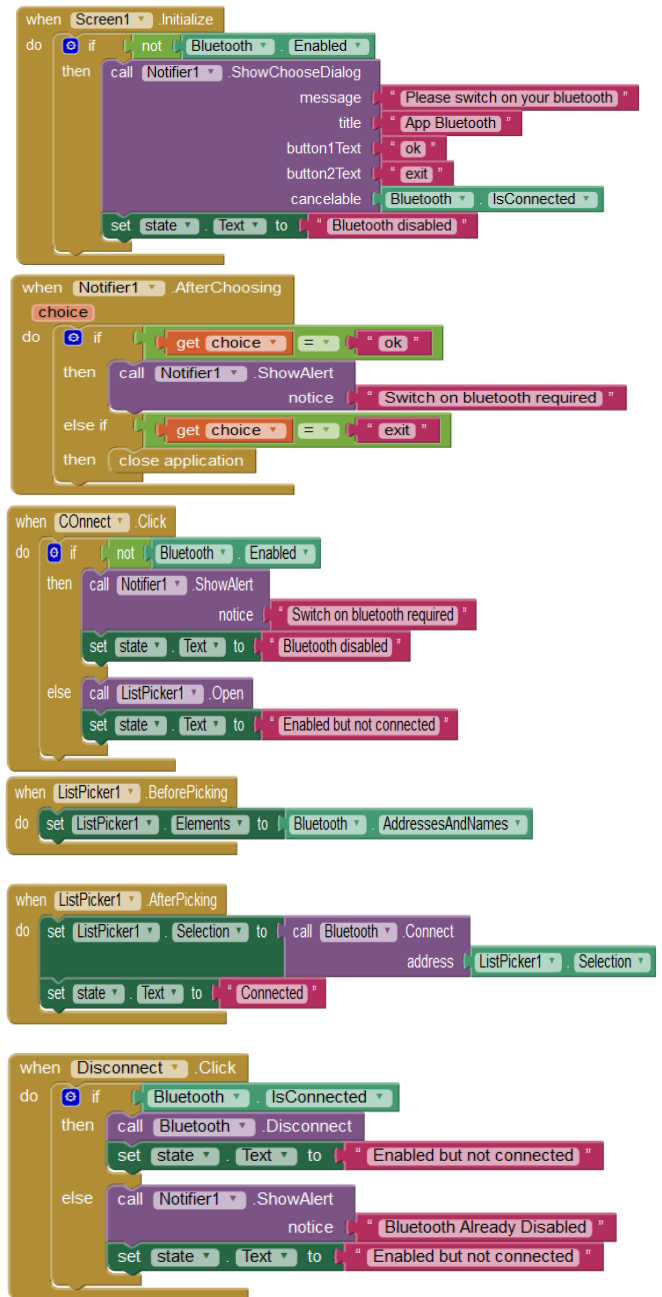


Fig.2. Bluetooth Connection.

Bluetooth Based Home Automation

Fig.2 shows the block programming for establishing a connection with the Bluetooth module. When application screen is initialized, it checks whether Bluetooth is enabled. If Bluetooth is not enabled, the screen will pop up a message saying 'Please switch on your Bluetooth' and will give two options either OK or exit. If exit is selected, the application will close otherwise if ok is selected; the user can switch on the Bluetooth and come back to the application. The status bar will display 'Bluetooth disabled'. When 'Connect' is clicked and Bluetooth is enabled, it will open a list picker with names and addresses of all paired Bluetooth devices. The user has to pick the Bluetooth address of the Bluetooth module. After picking the name, Bluetooth module will be connected to the user's phone. The status will change to 'Connected'. When 'Disconnect' is clicked, the Bluetooth device will be disconnected and the status will change to 'Enabled but not connected'. Also, if the device is not connected to any other Bluetooth device, it will pop up a message saying 'Bluetooth already disabled'.

Once the connection is established, a person can press access the controls provided by the application as follows:

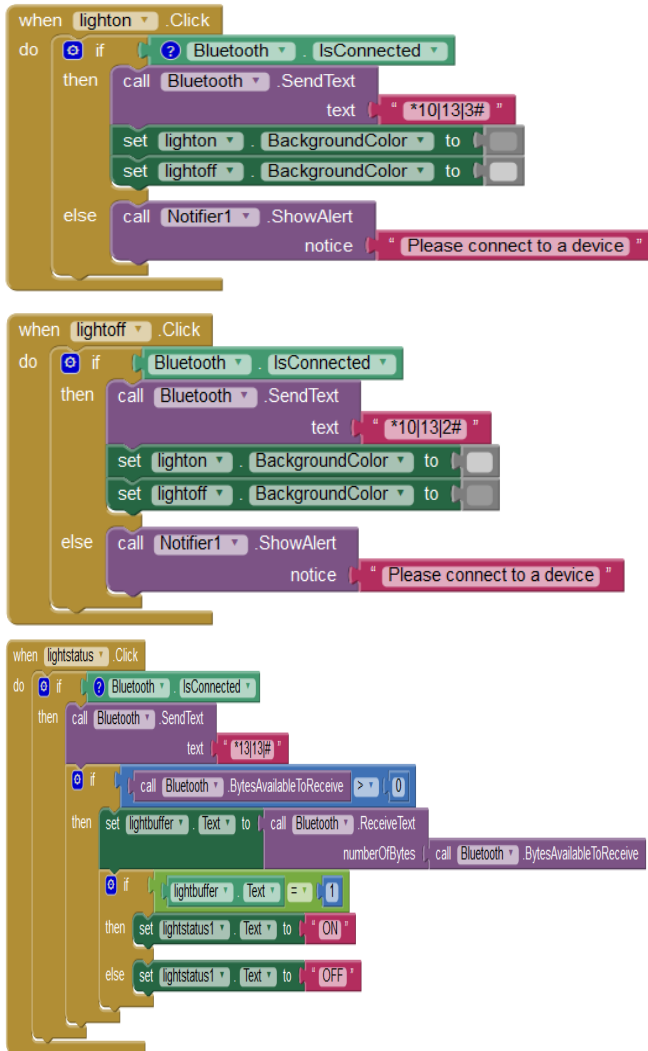


Fig.3. Case study of light.

Fig.3 is a case study of how a light bulb is controlled through the phone application. If light on is clicked, Bluetooth will send *10|13|3# to the Arduino board. Also background color of the button will change to a darker shade of gray. If Bluetooth is not connected, a pop up message will suggest connecting to a Bluetooth device. If light off is clicked, Bluetooth will send *10|13|2# to the Arduino board. Also background color of the button will change to a lighter shade of gray. If Bluetooth is not connected, a pop up message will suggest connecting to a Bluetooth device. If light status is clicked, Bluetooth will send *13|13|# to the Arduino board. It will also check is any bytes are available to be received. If the received data is '1', the status will indicate to be 'ON' otherwise it will indicate to be 'OFF'. Fan, led and other digital circuits (if any) will work in the same way.

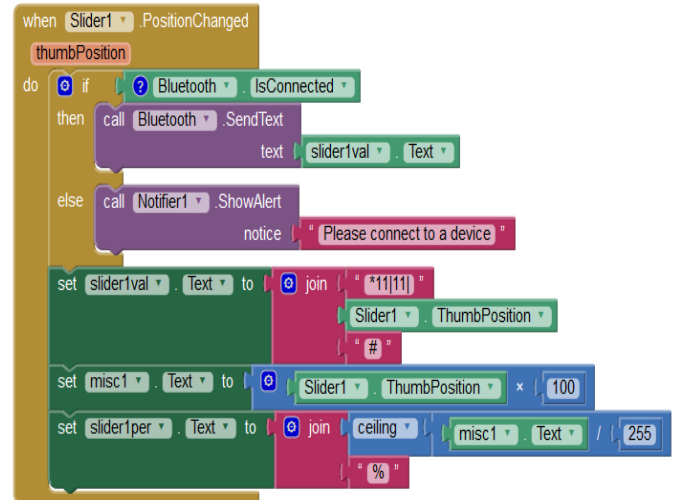


Fig.4. LED strip analog control.

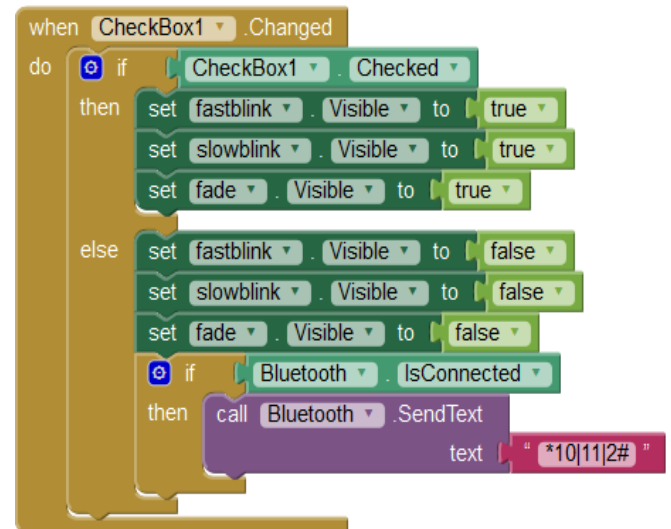


Fig.5. Party mode checkbox.

As shown in fig. 4, for LED strip, we need analog control along with the digital control. A slider is used for this purpose. Whenever the slider position is changed, a value will be stored in a hidden variable 'slider1ThumbPosition'. If

Bluetooth is connected, it will send an appended text consisting of '11|11', value of slider1ThumbPosition,'#'. A percentage value is also calculated and displayed on the screen. Fig. 5 shows how party mode option visibility is changed with a checkbox. If the checkbox is selected/ticked, all the options for the party modes are enabled, i.e. fast blink, slow blink and fade (in this case). Else the options do not appear on the screen and are disabled. This is done because party mode may not always be wanted by the user. He/She can enable/disable it according to his/her requirement.

III. HARDWARE IMPLEMENTATION AND DESIGN

A. Arduino

The Arduino UNO microcontroller board is used as our central controller. An XBee Shield is mounted on the microcontroller board. The XBee Shield simplifies the interfacing between the Microcontroller board and the BTBee Pro module.

B. Automated House Model

Fig.6 shows the house model used for prototype development. The project model is made up of acrylic sheets, which have dimensions of 50X50X30 cm with a hinged roof. There is a compartment for washroom with dimensions of 15X15cms. It consists of the proximity and the LDR circuits. On the hinged roof lie the smoke detector, the temperature sensor and the buzzer. The roof also has the LED strip, which creates different party moods in the house. The Curtain is placed to one end of the house. There is a level detector which is placed inside the water tank, and it is kept inside the house model. All the other circuitry including the arduino are placed on a wooden plank outside the house model and wired to their respective output devices.



Fig.6. Automated house model.

C. Relay Circuit

A relay is an electronic switch, which is used to switch a high power device using a low power signal. In this home automation application, an Arduino pin cannot directly be used to switch a relay due to current limitations of Arduino. Therefore, we take help of a bipolar junction transistor. The base of the transistor is connected to the Arduino pin. The collector of the transistor is connected to the negative pin of

the relay coil and the emitter is connected to the ground. The load such as bulb/fan is connected to the higher voltage side of the relay. When we give a 5V supply through the base of the transistor, the collector gets the ground and the transistor's switching action takes place. Thus, the current through the Arduino pin acts merely as an enable pin.

D. Washroom light automation

Nowadays, energy conservation, even on small scale, is very important. As we all know, we often forget to switch off the washroom lights when we come out. The logic behind this circuit is to switch on the lights only when required. When a person is present in the washroom and it is dark outside, the washroom light needs to be switched on. The light is not needed in the day time regardless of whether a person is present in the washroom or not. Automation of this light can help us prevent energy wastage. Proximity sensor and a Light Dependent Resistor (LDR) are used to implement this logic.

TSOP1738: is a very popular and commonly used infrared sensor. It will detect only infrared signals that are on & off in a frequency near to 38 KHz. Its output will be low when the prescribed infrared signals fall on it and vice-versa.

LDR: offers resistance in response to the ambient light. The resistance decreases as the intensity of incident light increases, and vice versa.

In the washroom compartment of the project model, when a person enters the washroom, the proximity sensor detects the presence of the person and the LDR detects if it is day or night. Thus if it is night time, the bulb turns on when a person enters the washroom. Similarly for day time, the bulb remains off as there is no requirement for any extra lighting.

E. Water Level Detection

Similar to washroom lights, we often keep the bathtub water running and go out to do our work till it fills up. The water level detection circuit will fire a notification alarm when the bathtub level is full and can also give a notification on the phone application. There are various level sensors available in the market for this purpose but since our project is a prototype, we have used a basic level detection principle. This circuit works on the conductivity principle in which water being the conductive medium switches the transistor to complete the circuit. Fig. 7 shows the water level detection circuit. This circuitry consists of five probes, placed horizontally at different height to indicate different levels of the tank. The probes are made of simple electric wires. There are five LEDs to indicate the level of the tank. As the water reaches to the level of each probe, the base draws current at 5V, which forward-biases the base-emitter and switches the transistor on. Thus, the LEDs turn on when water reaches up to the level of these probes. This happens when the bottom most probes gets the positive voltage supply of +5V and the other probes are connected to ground. The tank is full when the level of water reaches the topmost probe. An alert message is sent to the phone indicating the level of water in

Bluetooth Based Home Automation

the tank. We can acknowledge the message obtained on the phone.

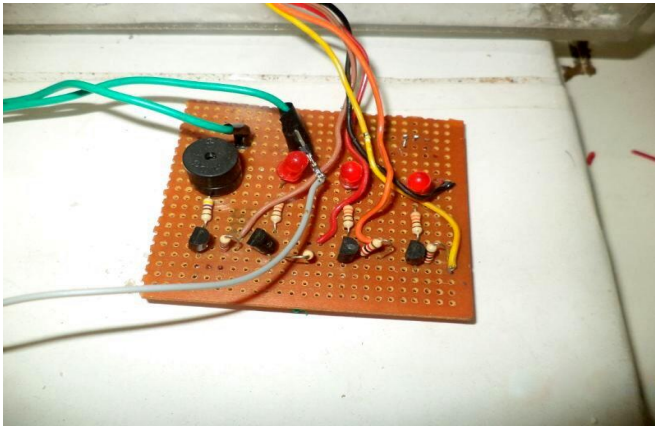


Fig. 7. Water level detection circuit.

F. LED Strip using Pulse Width Modulation

The LED strip is used to change the ambience of the house by adjusting the light intensity. There are two terminals at the end of the strip, positive and negative. The positive is given to the 12V supply and the negative is connected to the drain of the n-channel MOSFET FQP45N03L. The circuit is based on pulse width modulation technique. The Arduino generates analog voltage levels from 0-5V in 256 steps. The MOSFET FQP45N03L regulates the power supplied to the LED Strip based on the voltage at the GATE of the MOSFET. The more the power supplied, higher will be the intensity of the LED Strip.

G. DC Power Supply

A bridge rectifier made by four diodes is used to rectify the signal and convert AC to DC. After the signal is rectified, we get a rippled signal. The filtering capacitor reduces the

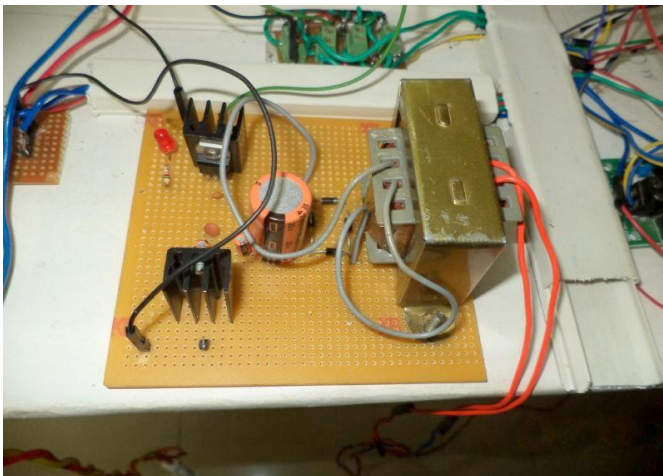


Fig.8. DC power supply.

ripple by storing the charge and discharging it to give a continuous and smooth signal. A heat sink is used to avoid overheating of the ICs. The 12 Vdc power supply is used to drive the motors (used in curtain slider circuit), LED strips in

the house model. It uses a 3-pin voltage regulator IC LM7812 to obtain constant 12V dc supply and is ideal for current requirement of 1 amp. Further, the 5 Vdc supply is used to give supply to all the sensors in the house as well as to switch relays for high power circuits in the house model. The 5V power supply is obtained using the IC LM7805 to step down 12V dc further down to 5 Vdc. These ICs belong to the IC LM78XX series. The LM78XX series is a positive voltage regulator IC for various voltage requirements. Fig. 8 shows the DC power supply built for this application.

H. Fire and Alarm system

Smoke Detector: The MQ2 Gas and smoke sensor detects the concentration of combustible gases in air. The readings are obtained as analog values of voltages.

Temperature Indicator (LM 35): A temperature indicator (LM 35) is a temperature sensor, which senses the temperature of the surroundings. The output voltage varies by 10mV in response to every degree Celsius rise and fall in temperature.

Buzzer: Buzzer is connected to an Arduino pin and is programmed to go off when smoke as well as high temperature is detected.

Temperature sensor is used along with smoke because sometimes if a person is smoking, a fire alarm can go off causing unnecessary chaos. As this circuit requires both the conditions to be satisfied, it makes sure that alarm is fired if and only if there is an actual case of fire. When the temperature sensor and the smoke sensor reach a certain threshold, the alarm/buzzer starts to ring. A message is sent to the phone when the smoke alarm/buzzer starts to ring. This will enable faster evacuation.

I. Curtain slider using motor driver circuit

The mechanism of curtain slider used in this project is pretty unique. An obstruction is placed between the 1st and the 2nd curtain ring which moves along with a thread attached to the two motors on two ends. As the obstruction moves to and fro, it drags the curtain with it. Thus, the curtain slides. The serial command for curtain control is CMD_CURTAIN. Once the controller enters the curtain control loop the controller checks the value of the pin_value. If pin_value is PIN_HIGH the curtain will open and if the pin_value is PIN_LOW the curtain will close. The controller gives command to two pins which control the direction of the motor. The direction of the motor decides whether the curtain will open or close. The time for which the motor is operational is decided by the delay that is given in the microcontroller code. The motor driver IC drives the motor which draws current at 12 volts DC. The motor driver takes input from the microcontroller and based on the values at input 1 and input 2 of the motor driver circuit the motor rotates in the clockwise or anti clockwise direction. If the value at input1 is 5 volts and if the value at input2 is 0 volts the motor rotates in the clockwise direction and if the value at

input1 is 0 volts and if the value at input2 is 5 volts the motor rotates in the anti-clockwise direction.



Fig.9. Curtain slider.

IV. CONCLUSION AND FUTURE SCOPE

The Bluetooth based Home automation system with an automated home model is built. Our prime objective is to assist handicapped/old aged people. This project gives basic idea of how to control various home appliances and provide a security using Android phone/tab. This project is based on Android and Arduino platform. So the overall implementation cost is very cheap and it is affordable by a common person. Looking at the current scenario we have chosen Android platform so that most of the people can get benefit. The design consists of Android phone with home automation application, a Bluetooth module and Arduino UNO board. User can interact with the android phone and send control signal to the Arduino UNO which in turn will control other embedded devices/sensors. We have discussed a simple prototype in this paper but in future it can be expanded to many other areas. The application can be developed for platforms other than android. SMS facility can be added for remote communication.

V. ACKNOWLEDGMENT

We would like to thank Prof. J. A. Gaikwad for his enormous support and guidance throughout. We would also like to express our deep gratitude to Prof. P. V. Pol and head of department Prof. P. M. Kanjalkar for providing us the technical knowledge that we needed to complete our project successfully.

VI. REFERENCES

- [1] Deepali Javale, Mohd. Mohsin, Shreerang Nandanwar, Mayur Shingate, "Home Automation and Security System Using Android ADK", International Journal of Electronics Communication and Computer Technology (IJECCCT), Volume 3, Issue 2 (March 2013).
- [2] Zeeshan Ahmed, Mujtaba Ali, Saman Majeed, "Implementing Computerized and Digitally Mobile Home Automation System towards Electric Appliance Control and

Security System", nt. J. Emerg. Sci., Volume 1, Issue 3, 487-503.

[3] Android official website. <http://developer.android.com/>.

[4] Appinventor official website. Appinventor.mit.edu.

[5] ArduDroid: A simple 2-way Bluetooth-based android controller for Arduino. <http://www.techbitar.com/>.