

# Remote Control of Home Appliances via Bluetooth and Android Smart Phones

[<sup>1</sup>] Roda Vishal, [<sup>2</sup>] B.Rakesh, [<sup>3</sup>] E.Jyothi, [<sup>4</sup>] R.Ranjith

[<sup>1</sup>] Asst. professor,

Balaji Institute of Technology and Science, Warangal, Telangana

**Abstract:** -- With everyone being on the move in a fast paced World, technologies have been increasing rapidly. This Work is regarding a student-designed project allowing users to be able to control multiple appliances remotely from the Single mobile device. This project involves the use of Bluetooth Communication and the Arduino Uno Rev 3 Microcontrollers. The whole idea is to design an app on an Android cell phone to control home appliance remotely such as lights and fans Using AC power. Although there are commercially available products on the market that implement the control of multiple Applications with a single device, this project is a teaching Point for students to build their own communication networks, create Android phone apps, and practice electrical operation of circuits.

## I. INTRODUCTION

The Enabling Technologies Laboratory Student Design Program provides Wayne State University's undergraduate engineering students with the opportunity to design and Create prototypes, custom designed devices, software and services to aid persons, especially for those with disabilities. In This paper, we shall present a student design work. With many Electronic devices adopting wireless technologies, everyday Tasks are becoming easier to perform from a single device. Turning on lights while you are at office or monitoring your Homes thermostat, for example, no longer requires you to perform them from a fixed location. These tasks can be achieved by using Bluetooth systems [1] where Bluetooth Technology is used to send data over short distance. Many Mobile devices such as cell phones and tablets have this Function already packaged with the device. This paper will present a design of using an Arduino microcontroller and an HC-05 Bluetooth module to implement remote functionality of various outputs. The outcome of this work is a system that Utilizes an Android phone app to turn on/off a fan (120V AC) And an LED strip (9V DC). The purpose of this project is to gain a better understanding on how Bluetooth communication operates, more in depth, in terms of sending and Receiving serial data between multiple devices. This project will implement design, circuitry, and programming, graphical user interface (G.U.I), and project management skills in order to construct a system that allows the user to control multiple Outputs remotely using Bluetooth technology.

- 1) Implementation of serial communication between an Arduino and a Bluetooth module,
- 2) Creation of an Android phone app that can be used to Communicate to Bluetooth module,
- 3) Construction of Arduino programs that will receive data from the Bluetooth module and perform tasks based on data on serial bus,
- 4) Test of phone app in conjunction with the Bluetooth module and Arduino, and
- 5) display of status of outputs on phone app and debugging Of software of any unwanted actions / outcomes.

## II. METHODOLOGY

### A. Communication of Various Hardware Boards

The first step in creating a system that utilizes Bluetooth technology is to ensure that the selected hardware boards could communicate with each other. To confirm communication among the Arduino board, Android phone, and Bluetooth module, a simple circuit was constructed to see if an LED could be turned on and off using the phone app. Fig. 1 is the wiring of the Arduino with an HC-05 module. In order to monitor the serial bus, within the Arduino IDE, the serial bus can be displayed using the command Ctrl + Shift + M. Once the serial bus is displayed, traffic over the Tx and Rx pins on the Arduino can be monitored. For this test, text strings LED: ON and LED: OFF were sent from the phone to the Bluetooth module then sent over to the Arduino. The Android phone app was created using the tools from MIT App inventor 2, which is a free online tool. Successful communication was confirmed between the phone app and Arduino. Bluetooth module HC-05 Receive Pin operates at 3.3 Volts, which is why there is a voltage Divider implemented.

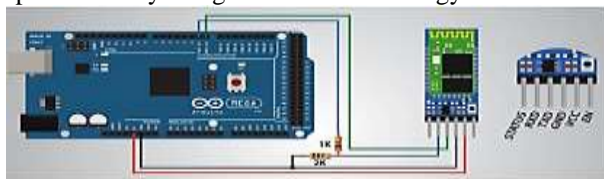


Fig. 1. Arduino board and HC-05 wiring (Courtesy of [2])

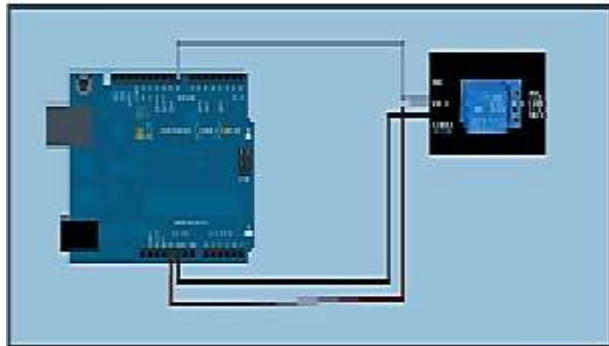


Fig. 2. Wiring a relay to the Arduino (Courtesy of YS Tech Share).

**B. Confirmation of Operation between Arduino and Relays**

With communication now confirmed, constructing a circuit to drive a high-power device is our next task. At early stages of the project, we had the Arduino send a digital HI signal (5 Volts) out of Pin 7 to illuminate an LED. This operation was successful and was a breakthrough milestone of project development, as the logic was proven to work correctly. After functionality of controlling an LED on and off was established, the next task is to have the Arduino drive a relay. The use of a relay allows the ability to electrically isolate the Arduino from the output circuitry, thus adding protection to the microcontroller. More importantly, the relay can enable the Arduino to drive high-power devices. The used relays have fly back diodes incorporated into the design to help prevent unwanted arcing [3]. Driving the relay uses the same logic as controlling an LED. This particular relay module requires the user to send digital LO in order to energize the relay coil. Fig. 2 shows the wiring of the Arduino board to a relay. The first test with the relay was to drive a Blue LED light strip that was powered by a separate power supply. The test was successful and the circuitry to control an output with a separate power supply was confirmed to operate correctly. Now that the relay control logic was confirmed to operate correctly, the next task was to build an outlet that would utilize 120VAC. This custom outlet would allow the user to have remote functionality over an application that plugs into a standard wall outlet. There are some limitations with this because the application has to have fixed physical switches to be left in the ON position for this to work. A fan was used to test the custom outlet, and a video will be demonstrated for successful operation of controlling a fan and the LED light strip.

**C. The MIT App Inventor Interface**

MIT App Inventor is an innovative programming platform for app creation using visual, drag-and-drop building blocks [4,5]. The simple graphical interface is so friendly that even an

inexperienced novice can be able to create a basic, fully functional app within an hour or less. Detailed introduction can be found in [4,5], and Fig. 3 is an interface created for this project using MIT App Inventor 2.



Fig. 3. Android app Bluetooth device

**III Program code**

```
void loop()
{
  if (wizbt.available())
  {
    digitalWrite(statusled,HIGH);
    delay(100);
    digitalWrite(statusled,LOW);
    BluetoothData=wizbt.read();

    switch(BluetoothData) {
    // RELAY - 1
    case 'B' :
      digitalWrite(r1,HIGH);
      EEPROM.write(ad1,1);
      break;
    case 'A' :
      digitalWrite(r1,LOW);
      EEPROM.write(ad1,0);
      break;
    // RELAY - 2
    case 'Y' :
      digitalWrite(r2,HIGH);
      EEPROM.write(ad2,1);
      break;
    case 'X' :
      digitalWrite(r2,LOW);
      EEPROM.write(ad2,0);
      break;
    // RELAY - 3
```

```

case 'Q' :
    digitalWrite(r3,HIGH);
    EEPROM.write(ad3,1);
    break;
case 'P' :
    digitalWrite(r3,LOW);
    EEPROM.write(ad3,0);
    break;
// RELAY - 4
case 'N' :
    digitalWrite(r4,HIGH);
    EEPROM.write(ad4,1);
    break;
case 'M' :
    digitalWrite(r4,LOW);
    EEPROM.write(ad4,0);
    break;
// MASTER SWITCH
case 'V' :
    digitalWrite(r1,HIGH);
    EEPROM.write(ad1,1);
    digitalWrite(r2,HIGH);
    EEPROM.write(ad2,1);
    digitalWrite(r3,HIGH);
    EEPROM.write(ad3,1);
    digitalWrite(r4,HIGH);
    EEPROM.write(ad4,1);
    break;
case 'U' :
    digitalWrite(r1,LOW);
    EEPROM.write(ad1,0);
    digitalWrite(r2,LOW);
    EEPROM.write(ad2,0);
    digitalWrite(r3,LOW);
    EEPROM.write(ad3,0);
    digitalWrite(r4,LOW);
    EEPROM.write(ad4,0);
    break;
default :
    for(int i=0;i<5;i++)
    {
        digitalWrite(statusled,HIGH);
        delay(100);
        digitalWrite(statusled,LOW);
    }
    }
    }
    delay(100);
    }
    
```

**COMPLETED PROTOTYPE**



**III. CONCLUSIONS AND DISCUSSION**

This project helped understand the communication methods of devices that utilize Bluetooth technology along with the functionality they provide. After completing this project, students have a better understanding on how to construct their own network of devices and control with a custom made Phone app. The MIT App Inventor 2 makes creating Android apps easy. This project also greatly emphasized the practice of electrical isolation and driving of high power appliances using relays. User interface design is an interesting challenge throughout this project, taking into consideration many factors such as a clean layout as well as to fool proofing the software to make sure that no unwanted situations could happen. We would like to make future improvement such as auto connection, i.e. when the phone comes into range of the Bluetooth module, the devices would automatically pair themselves together. Packaging would also be an area to improve, implementing the circuitry onto a printed circuit board would greatly reduce space and make the project easier to implement into a household environment. Overall, this project was a good introduction to Bluetooth communication along with phone app creation.

**REFERENCES**

[1] "Bluetooth." [https:// en. wikipedia. Org /wiki /Bluetooth](https://en.wikipedia.org/wiki/Bluetooth). Accessed on 22 Nov. 2016.

[2] D. Nedelkovski, "Arduino and HC-05 Bluetooth Module Tutorial: Android Smartphone & Laptop Control." YouTube.com, accessed on 29 Feb. 2016.

[3] "Flyback Diode." [https://en. wikipedia.org/wiki/Flyback-diode](https://en.wikipedia.org/wiki/Flyback-diode). Accessed on 29 Sept. 2016.

[4] D. Nedelkovski, "How To Build Custom Android App for Your Arduino Project Using MIT App Inventor." YouTube.com, accessed on 30 Mar. 2016.

[5] [http :// appinventor. mit. Edu /explore/ about- us.html](http://appinventor.mit.edu/explore/about-us.html). Accessed on Sept2016.372