

## A Speech Activated Control System for Infrared Appliances

Ajaegbu Chigozirim<sup>1\*</sup>, Adetunji Oluwatofunmi<sup>1</sup>, Vivian O. Nwaocha<sup>2</sup>, Ndunagu Juliana<sup>2</sup>

<sup>1</sup> Department of Computer Science, Babcock University, Ogun State, Ilishan-Remo 121103, Nigeria

<sup>2</sup> Department of Computer Science and Information Technology, National Open University Abuja, Jabi 900211, Nigeria

Corresponding Author Email: [ajaegbuc@babcock.edu.ng](mailto:ajaegbuc@babcock.edu.ng)



<https://doi.org/10.18280/jesa.530113>

### ABSTRACT

**Received:** 5 November 2019

**Accepted:** 14 December 2019

#### **Keywords:**

*speech recognition, infrared signal, control systems, effectors, sensor, controller*

Infrared (IR) technology is still prominent in modern day although an argument can be made of its lack of importance given the availability of more secure and less hazardous means of control in modern day. Nevertheless, it still possesses applications in modern day control systems. Older model infrared appliances cannot benefit from modern day speech control as they do not possess an interface to receive and process speech input and do not have the ability to connect to a local area or wireless network to fully benefit from this technology. A prototype model was adopted with limited functionalities, in order to enable quick implementation and facilitate the analysis of the working of the project. Also, some programming languages such as Arduino, JAVA and MIT App Inventor were adopted to enable for proper interfacing with some hardware components such as Arduino Nano, Bluetooth Module (HC-05) and IR Transmitter. The result of the study showed a control system providing support for older model infrared appliances. Also, a good environment for installation and adequate maintenance practices is demand for optimum workability. In conclusion, the system is user-friendly and requires little to no orientation to understand its workings as the intended appliance is supported.

## 1. INTRODUCTION

Home automation today incorporates the latest technology ranging from IoT to speech recognition to simple motion sensors. However, a majority of these systems require the devices to exist on the same wireless network. Older model devices do not possess the components necessary to connect to and communicate with these networks. Thus, users of older model devices cannot benefit from the ease of using these new technology.

Speech control is a field in Artificial Intelligence which although appears to be a new field to the world is not new to the world of Computer Science. It has been in existence since the 1950's which was first seen in a system "Audrey" the very first known speech recognition system. This system was built in Bells Laboratories in 1952 by Davis, Biddulph, and Balashek [1]. Audrey had the ability to recognize numbers from zero to nine and could only understand this coming from a familiar voice [1].

The evolution of technology in Artificial Intelligence creates opportunity for even greater control over these devices. Functionalities such as speech recognition and Natural Language Processing have made it possible for users to interface with systems by just speaking to it. Speech control has since advanced from the methods used for voice control and synthesis since the 1950's. Now in our daily lives we see applications of voice recognition and control from our soft copy dictionaries to the iPhone Siri.

In recent years, Speech control has been implemented in IoT devices. IoT devices have been popularized over the last few years from application in complex systems such as cars to the simpler ones like blenders. These systems ensure control of

these devices can be controlled even when human agents aren't available.

IoT has applications across broad spectrum of industries, most especially smart homes. Home appliances such as the TV, AC, and decoder can now be controlled remotely.

Although IoT has a variety of applications in modern day and makes controlling of appliances and thus life easier, only appliances capable of accessing and connecting to a wireless network can fully benefit from it. This poses a major drawback as there is little to no support for older model appliances without a dedicated network interface card cannot benefit from this technology and by extension, current speech recognition technology. Hence, this study aims to develop a speech activated controlled system for IR devices.

## 2. LITERATURE REVIEW

In our daily activities, we get to use infrared appliances and gadgets that are being controlled by remote controls. With the advancements in speech recognition technology and home automation, compatibility between older model IR devices and Newer technologies should be of paramount concern in a society bolstering ease of access to services.

In the classroom environment, there are various IR devices such as Air Conditioners, Smart Boards, Projectors. These devices have dedicated remote controls, this Implies having to deal with several remote controls and constant replacement and usage of batteries. Thus, a central control system is needed to combat this limitation. This project aims at creating a single Voice controlled remote for multiple infrared devices with the use of mobile phones.

## 2.1 Review of related works

Pandaya [2] looked at an automated home controlled appliances using voice, routing from an Android application. The main aim is to create a system capable of allowing users to control appliances using their voice. In order to activate the switching mechanism of the system, the study adopted an Arduino Mega board. In essence, their study adopted the use of hardware and software to facilitate the functionality of the system.

Saiskumar [3] looked at the voice recognition technological advancement as well as its invention with focus on using MatLab programming in analysing the different processes involved in speaking technology. The aim of their study is to design a system that will function within the operation of voice recognition system and wireless system in order to present a smart home for the elderly and the disabled. Their study further benchmarked the captured speech with samples of previously trained words in the database and accuracy level of 75% and above were recorded.

Microcontrollers lab [4] implemented a project aimed at creating a system and application that would be easy to use for different individual of any age bracket. Some of the components that aided the design as as follows: Arduino Uno , HC-05 Bluetooth Module, 4 Channel Relay Modules, 16x2 LCD and AC bulbs with holders . From their work, an AMR\_Voice application accepts spoken commands, the Bluetooth receives the command through Arduino which also, writes to the serial monitor.

Sen et al. [5] designed an intelligent voice controlled home automation system using smartphones. The aim of their work was to prompt control over every home appliances with the voice. The system employed the use of key components such as: Android based phone, Bluetooth module, Arduino Uno and Relay boards. Their work concluded with a result of an automated system capable integrating the operation of every home appliances in a single point without any intervention of manual effort.

Chaitanya et al. [6] designed an Internet of Things based smart IR device using a Wi-Fi module CC3200, which forms the essential part of the design. The aim of their work is to design an automated IR device capable of controlling all home IR enable devices. In order to achieve their objective, a learning and operational module were employed accordingly. Their work concluded with a low cost and low power implementable system.

Parameshchhari et al. [7] proposed an automated smart home control system with an imbedded voice command control system, which enabled users to control the operation of home electrical appliances. The main objective of their study, is to create an automated system that will provide a more conformable environment for individuals more especially the elderly and disabled. Their system adopted a Digital Signal Processor and an XBee transceiver in order to activate the voice command and elimination of large amount of wiring respectively. The work concluded with an actualization of a cost effective system the will make life more comfortable for users.

The block flow diagram of the proposed architecture is as shown Figure 1.

The flow diagram for the voice recognition program is given in Figure 2.

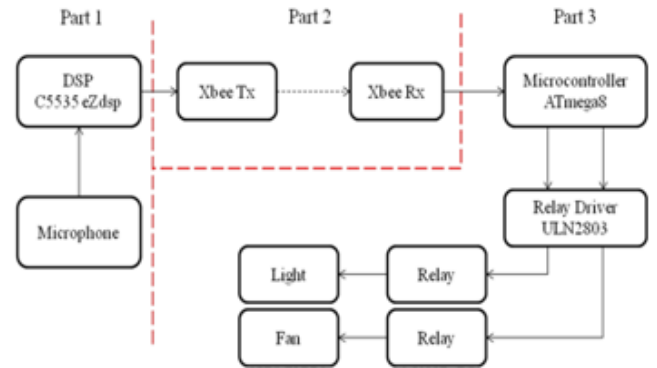


Figure 1. Proposed block diagram of the smart home control system [7]

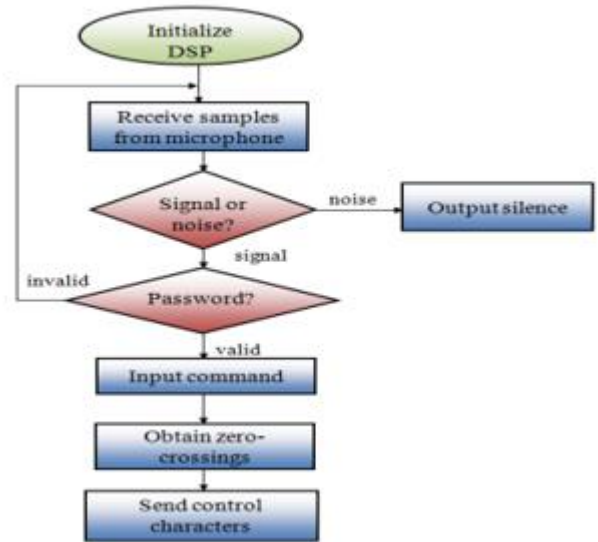


Figure 2. Flow diagram for voice recognition program [7]

Eltayeb et al. [8] proposed a Voice recognition system for controlling a smart home with the intention of fostering its implementation in an existing home environment. Their aim, is to expatiate more on the method of testing the operation and technicality of the voice control functionality in a smart home. Their work concluded with the design of an automated system capable of controlling all home electrical appliances with a satisfactory level of experimental result when implemented.

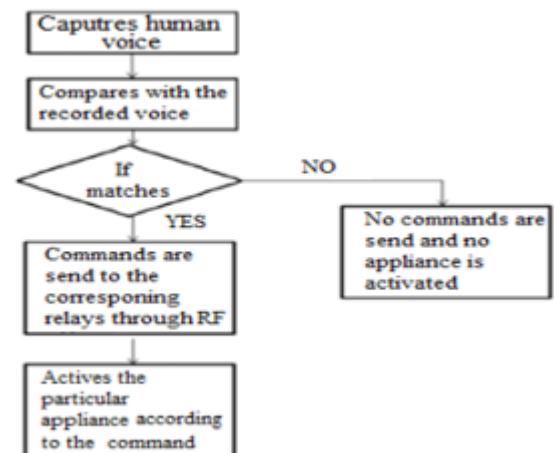


Figure 3. Flow diagram of the voice recognition based home smart control system [8]

The flow chart illustrating the sequence of functions of the Voice Recognition Based Smart Home Control System is given Figure 3.

Mahmud et al. [9] proposed a system that will make use of natural voice technology to activated electronic devices of home appliances hence, receiving voice commands and then behave accordingly to control home appliances. A software was implemented to recognize every voice command of every person. From the voice command, a corresponding control signal is generated to be passed through the parallel port to the control circuit to control the designated appliance.

The voice command is the initiator in the system for all further controls and operations. A particular voice commands corresponds to a predefined control action. The block illustration of the system is shown as Figure 4:

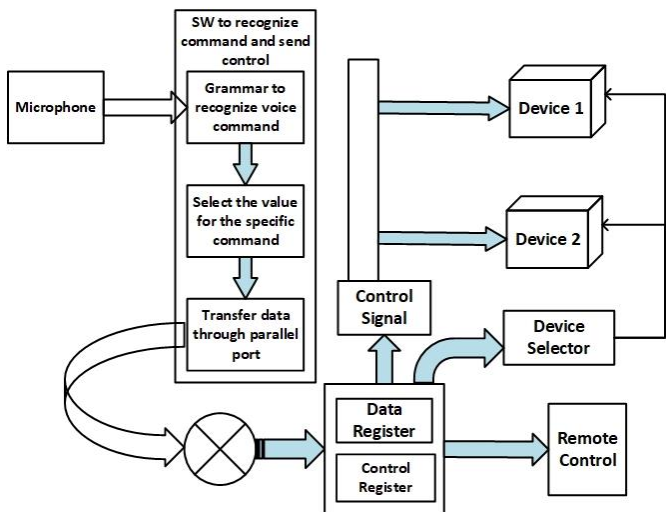


Figure 4. Block diagram of voice activated electronic device control system [9]

The system is divided into two major functions: Detection of the voice and Control of the device. The first part receives voice commands in the computer through a microphone while the second part receives the converted command and controls the appliance accordingly. C# was used to recognize voice commands as well as passing data to port.

Chien and Tai [10] envisioned "The Information Home Appliance Control System-A Bluetooth Universal type remote controller". The idea of their work is to resolve the integration problem of home remote controllers from different products, while providing an integrated control interface. The researchers, employed the features Bluetooth communication technology in order to develop a multi-purpose remote control system. The Bluetooth module was adopted in place of the wireless communication module found in regular interior phone set. Their study showed that the adoption of PCM encoding along with the transmission in digital mode, gave a clearer and stable output signal. The Block flow of the proposed system is illustrated in Figure 5.

Ayres and Nolan [11] proposed a voice activated command and control framework for speech recognition over WiFi. The work employed a Java controlled Lego Mindstorm robot as prototype device with the consideration of three different scenario configurations. Their study also designed a control grammar that was used to map to the corresponding methods in the Lejos code. The researchers, concluded that the developed framework can be easily extended and adapted to suit a variety of devices or applications.

Marimuthu et al. [12] proposed a voice controlled energy management system, that will allow individuals to control house hold electronics using wireless device. Their study reported achieving recording and monitoring of total power consumption with transmission of the data to the personal computer.

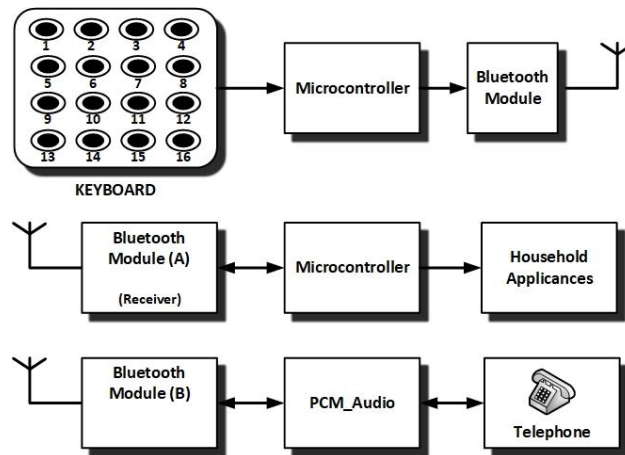


Figure 5. Block flow diagram of the information home appliance control system - A bluetooth universal type remote controller [10]

Gundogdu et al. [13] looked at Developing and modeling of voice control system for prosthetic robot arm in medical systems. In their study, they observed that their technique achieved about 11% efficiency in voice recognition compared with similar techniques in literature.

Ambalkar et al. [14] worked on a review on smart home technology using voice recognition. Their study established that the voice controlled home automated technology is very helpful for the aged and physically handicapped people.

Gunasekhar et al. [15] proposed an IoT based smart energy management system. Their work established that less attention has been paid to automating the appliance control towards conserving energy as at the publication of their work. The researchers were able to develop a prototype system that achieved their aim.

In view of these, the question still remains: *what happens to the older infrared signal controlled appliances without the ability to take advantage of this evolving technology?* Is obvious that older model infrared appliances cannot benefit from modern day speech control as they do not possess an interface to receive and process speech input and do not have the ability to connect to a local area or wireless network to fully benefit from this technology. Hence, the need for a voice controlled system for IR devices with a mobile interface.

### 3. SYSTEM DESIGN AND ANALYSIS

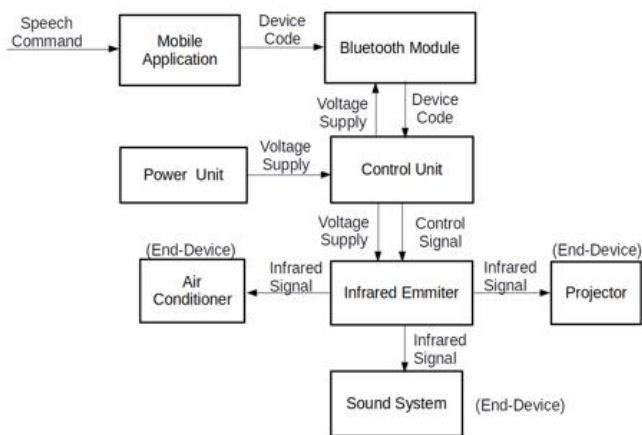
This section outlines the design and analysis of this Voice Control System for Infrared Devices and sheds more light on how the Voice Control module communicates with the Infrared emitter module to control the devices. The basic unit of the system are given below and shown in Figure 6.

- Power Supply Unit
- Bluetooth Serial Module
- Infrared Emitter
- Control Unit



## Arduino Nano Microcontroller

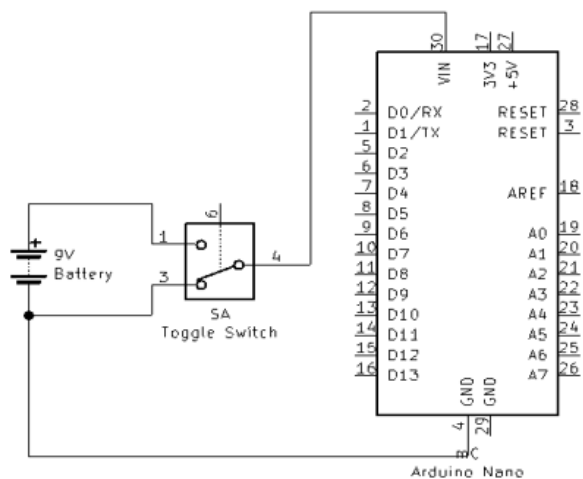
- Android Application
  1. MIT App Inventor
  2. Android Speech-to-Text
  3. Tiny DB
  4. Tiny Web DB



**Figure 6.** Block flow diagram of the speech activated control system for infrared appliances (Source: Authors, 2019)

### 3.1 Power supply unit

All stages in the system require a +5volts DC supply except the Android application which is supported on the mobile device. This is achieved by connecting a 9volts DC battery to the  $V_{in}$  and GND pins of the microcontroller. A switch toggle is added to enable switching between an “OFF” and “ON” state. A schematic diagram of the power supply unit is shown in Figure 7:



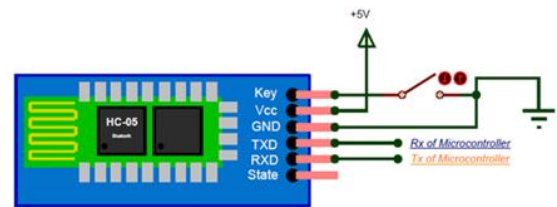
**Figure 7.** Schematic diagram of the power supply unit connection (Source: Authors, 2019)

### 3.2 Bluetooth serial module

The function of this module in this work, is to enable the transfer of data between the device and the accompanying android application. The HC-05 Bluetooth module was used for this device and this was adopted because of its capability for wireless serial connection setup. It is a complete package of V2.0+EDR (Enhanced Data Rate) 3Mbps modulation with 2.4Ghz radio transceiver. It operates with a default command

mode Baud rate of 38400 and data mode Baud rate of 9600 with support for 19200, 38400, 57600 and so on. It has an operating current of 30mA and voltage in the range of +4Volts to +7Volts.

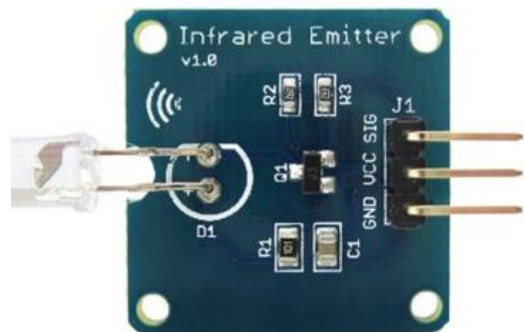
The HC-05 has a maximum connection range of 100m. The HC-05 employs a full-duplex connectivity which implies data can be transmitted and received concurrently. Figure 8 shows a 2-D model of the HC-05 Bluetooth module:



**Figure 8.** 2-D diagram of hc-05 bluetooth module (Source: Components101.com)

### 3.3 Infrared emitter

The function of this component is to blast the controlling signal(s) to the various Infrared devices from the system. The controlling code is sent from the android application to the device, decoded and then blasted to the air by the Infrared emitter. The Infrared emitter used in the device is the Open-Smart Infrared emitter which is based on the HL-A838 integrated infrared receiver module. It operates at a frequency of 38Khz. It has a supply voltage in the range of +3.3Volts to +5.5Volts (see Figure 9 for the diagram of Infrared emitter).



**Figure 9.** Image of the infrared emitter (Source: DX.com)

### 3.4 Control unit

This unit processes the data received from the android application, processes it, and sends it to the IR transmitter to blast the signal. It is also tasked with regulating temperature if an AC unit is configured with the system. It consists of an Arduino Nano microcontroller with an ATMEGA328 IC with a 32KB non-volatile Flash program memory that is parallel programmable.

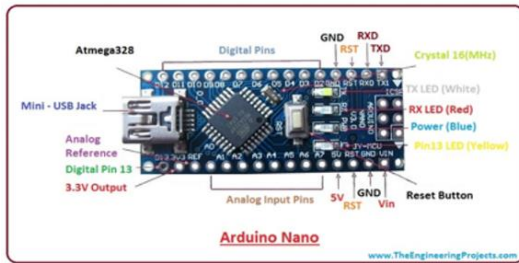
The work employed the use of a **microcontroller chip** and this is to enable ease in programming and re-programming the basic functionalities expected of the system.

The basic features of the Arduino Nano as adopted in this work are listed below and shown in Figures 10 a,b and c:

- Operating Voltage: +5Volts
- Input Voltage (limit): +6V - +12V
- Flash Memory: 32KB (ATmega328) of which 2KB is

used by the bootloader

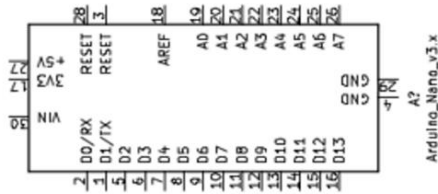
- EEPROM: 1KB (Atmega328)
- SPAM: 2KB (Atmega328)
- Digital I/O ports: 14 of which 6 provide PWM output
- DC Current per I/O port: 40mA
- Analogue Input pins: 8
- Clock Speed: 16MHz



(a)



(b)

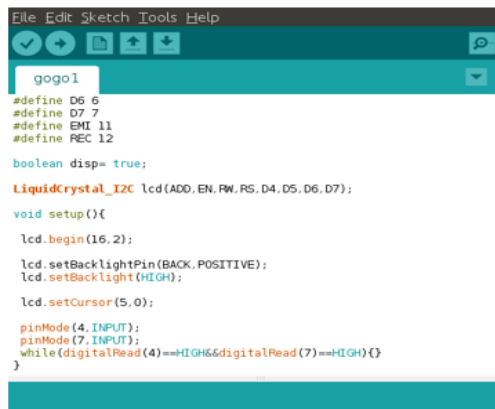


(c)

**Figure 10.** (a.) Diagram of the arduino nano, (b.) Schematic diagram of the atmega328p, (c.) Schematic diagram of the arduino nano (Source: Engineeringproject.com)

### 3.5 Using the microcontroller

The Arduino Nano microcontroller is relatively straightforward to setup and use. Provided the Arduino desktop IDE is available, a simple Arduino code can be written, compiled and uploaded to the board from the IDE. (See Figure 11). The Arduino Nano has a  $V_{in}$  pin for connecting a voltage source to power the board.



**Figure 11.** Diagram of the arduino IDE (Source: Authors, 2019)

### 3.6 Writing the control program

The Arduino programming language is based on the C++ programming language, so it is very easy to comprehend. A program written with the Arduino IDE is called a sketch and is saved on the development computer with a **.ino** or **.pde** extension. The structure of the program entails two main parts:

- **Setup():** This function is called once and is used to initialize variables and I/O pin modes.
- **Loop():** This contains the main body of the sketch. It includes possible function calls and other functionalities the sketch might have.

A simple Arduino sketch for blinking an LED is shown below:

```

#define LED 13
void setup(){
  pinMode(LED,OUTPUT);
}
void loop(){
  digitalWrite(LED,HIGH);
  delay(1000);
  digitalWrite(LED,LOW);
  delay(1000);
}
    
```

#### 3.6.1 Programming the microcontroller

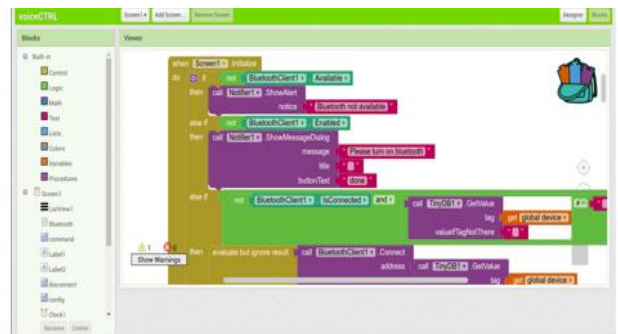
The Arduino Nano sketch is uploaded to the board by connecting it to the serial (USB) port on the development computer. This is done using a type A Male USB to a mini B Male USB cable. The sketch is compiled by the IDE and then uploaded to the board.

### 3.7 Android application

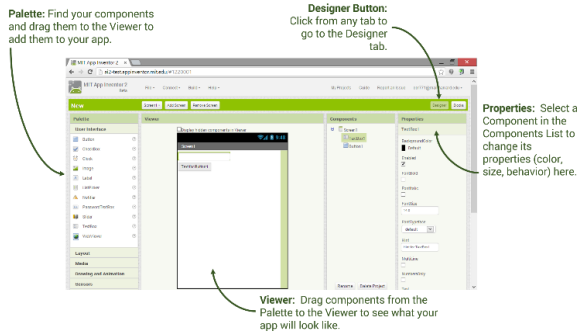
The Android application is tasked with obtaining vocal command from the user and then sending the relevant data to the control unit through the Bluetooth module. The application is built using the **MIT App Inventor** software.

#### 3.7.1 MIT App inventor

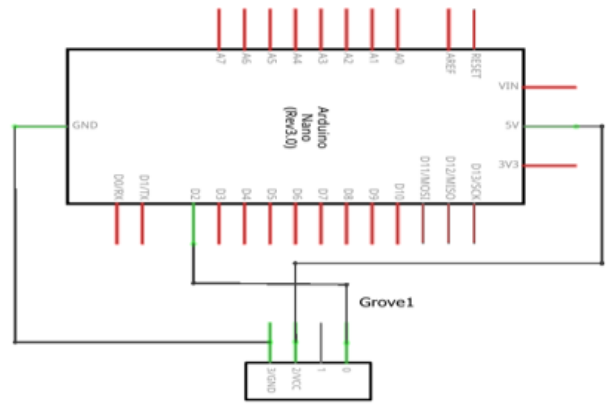
MIT App Inventor is an open-source web application originally provided by Google but now maintained by the Massachusetts Institute of Technology (MIT). It allows users to create software applications for the Android operating system (see Figure 12 for adapted MIT App Inventor). It is easy to use because it employs a drag-and-drop model for design and development (see Figure 13 for design view of adapted MIT App. Inventor). It also has an emulator feature for live testing of applications [16].



**Figure 12.** Block view of MIT App inventor (Source: Authors, 2019)



**Figure 13.** Design view of MIT APP inventor (Source: Authors, 2019)



**Figure 14.** Circuit diagram of the infrared learner (Source: Authors, 2019)

### 3.7.2 Android speech-to-text

The Android operating system comes with a built-in speech-to-text feature through which a user can provide a speech input and it can be converted to plain text. This is incorporated in the application to receive the voice command from the user which can then be sent to the control unit.

### 3.7.3 Tiny DB

Tiny DB is a persistent data store for the application, the data stored here will be available each time the application is run. Data items are stored under tags. To store a data item, a tag is specified under which it will be stored. To retrieve stored data, its corresponding tag just has to be provided. This component is used to store the infrared code and corresponding voice command as data and tag respectively to facilitate the functioning of the application.

### 3.7.4 Tiny web DB

Tiny Web DB is an App Inventor component that allows you to store data persistently in a database on the web. This component is used to store Infrared code for all the possible devices that can be configured for control on the application. The data is retrieved from the Tiny Web DB during configuration and stored in the Tiny DB after configuration.

## 4. SYSTEM IMPLEMENTATION AND TESTING

This section gives a precise documentation of the implementation of the speech activated control system. This section comprises development tools, algorithms and challenges encountered during the development process of the system.

### 4.1 Development tools

#### 4.1.1 Hardware

- **Connecting cables:** These are used to connect the various components together in the circuit.
- **Bread-board:** This formed the base upon which the circuit was built. It is a solder-less board and multiple circuits can be placed on it.
- **Wire strippers:** These were used to remove the outer covering of wires to enable better connection.
- **Android device:** This was used to test the application during construction.
- **Infrared Learner:** This tool is used to receive and decode infrared signals. It was used to learn the infrared control signals for the various supported infrared appliances. See Figure 14 for infrared learner circuit diagram.

#### 4.1.2 Software

- **Arduino IDE:** The Arduino IDE is necessary to program the microcontroller to perform the prescribed tasks.
- **MIT App Inventor:** This is used to create the Android application that receives the speech command and sends the corresponding code over Bluetooth.
- **Libraries:**
  - StandardCplusplus:** This library provides support for C++ data structure that are not supported by the Arduino programming language.
  - Vector:** data structure inherited from StandardCplusplus. Used to temporarily store the infrared signal.
  - IRLibSendBase:** This library is needed for initialization of the workings of the infrared emitter.
  - IRLib\_HashRaw:** This is responsible for facilitating the sending of the infrared signal through the infrared emitter.

## 4.2 System requirements

#### 4.2.1 Hardware requirements

- **Android device:** The Android device must have Bluetooth.
- **Memory:** 11MB storage space or higher is required for installation and optimal functioning.

#### 4.2.2 Software requirements

- **Android:** An android version of at least 4.4
- **Internet access**

## 5. SYSTEM IMPLEMENTATION

The Speech activated control system comprises two modules – An Android application and A physical control unit. The Android application is run on an Android OS supported device and requires just a standard installation of its .apk file.

All components other than the microcontroller require 5V electricity to function. The power supply unit provides 9V to power the microcontroller which in turn provides 5V to power all other components.

### 5.1 System functionality

A user configures the devices to be controlled by choosing the device type and manufacturer and then assigning ON and OFF commands to activate those functionalities. After configuration, the infrared code corresponding to the device

type and manufacturer chosen is retrieved from the Tiny WebDB and stored locally on the applications TinyDB. The infrared code is stored as a string but has to be transmitted as an integer array. When power is given to the physical control unit, a Bluetooth connection is established between the Android device and the control unit by choosing the device from the list of available Bluetooth devices on the Android device.

The user presses the button on the Android application to receive the speech command. When the speech command is received, it is converted to text and compared against tags in the TinyDB. If the speech command is found in the TinyDB, its corresponding infrared code is converted into a list and each element is sent over Bluetooth as a 2-Byte number to the control unit terminating in 1000.

At every instance, the control unit checks if data is available in the Serial buffer and temporarily store all data received. As the infrared code is sent as a 2-Byte number, every second data received is multiplied by 256 and added to the preceding data before being stored in a temporary array. If the data received is 1000, the control unit blasts the accumulated infrared code through the infrared emitter at a frequency of 35 KHz. All temporary data structures are then unset.

The complete circuit diagram of the control unit is shown in Figure 15.

The complete circuit employed a toggle switch connected directly to the Arduino Nano microcontroller, which serves as the heart of the developed system, with a total number of 13 pins divided into 14 digital input/output pins and 8 analog input/output pins. The Silver Mate Bluetooth is to the 9v battery for its power supply and to the infrared emitter to establish control. All resistor values are selected with respect to the required expected input power supply.

Figure 16 shows the main page of the android application with four buttons. The connect button allows the user to establish a bluetooth connections with ready devices, the configure button redirects the user to a page so as to select a device type and corresponding manufacturer, the command button redirects the user to the command configuration page where the user sets the on and off command for a particular device.

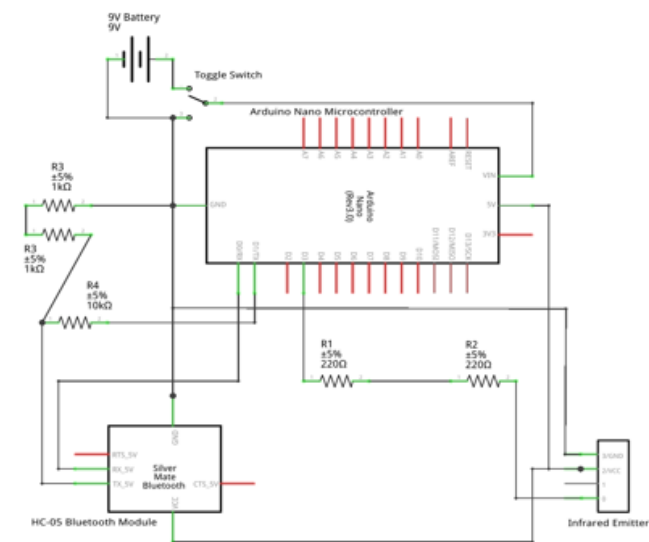


Figure 15. Complete circuit diagram (Source: Authors, 2019)

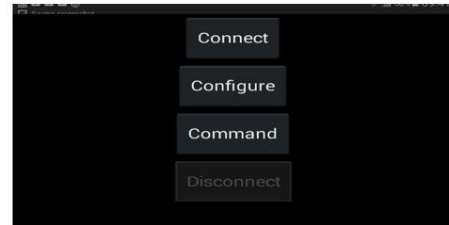


Figure 16. Main page of android application (Source: Authors, 2019)

Figure 17 shows the device configuration page based on the redirection from the main page. This page provides the user the opportunity to select a device type as well as the corresponding manufacturer.

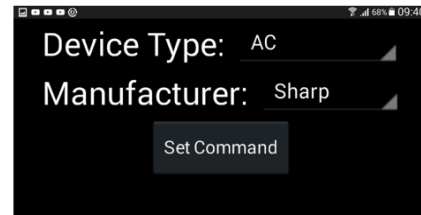


Figure 17. Device configuration page of android application (Source: Authors, 2019)

Figure 18 shows the command configuration page that allows the user to configure the on and off command for each device.

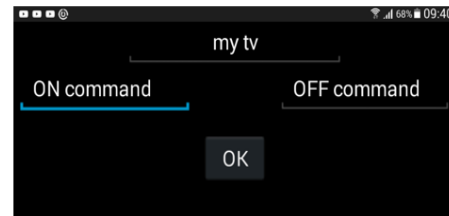


Figure 18. Command configuration page of android application (Source: Project, 2019)

## 6. TESTING AND ANALYSIS

Both modules were tested during development. During construction of the system, the following faults in each module were encountered and corrective measures taken shall be explained further below:

### 6.1 Android applications

- Bluetooth disconnection on screen transition
- Format to transfer stored infrared code

#### Control Unit

- Malfunctioning of the infrared emitter

#### 6.1.1 Bluetooth disconnection on screen transition

The Bluetooth connection between the Android device and the control unit kept breaking when the application transitioned from one screen to another. The problem arose because the connection had been refreshed on every screen. This was solved through the implementation of virtual screens

so all the application's activities occurred on only one screen.

#### 6.1.2 Format to transfer stored infrared code

The infrared code is stored as a string in the TinyDB of the application while the format in which it is to be emitted is an integer array, a problem arose on how to transmit the code. The initial method was to send the entire code as text and format at the control unit but the data proved to be too large for the microcontroller flash memory and the Bluetooth serial buffer. The problem was solved by transferring each element in the array sequentially at 10µs intervals as 2-Byte numbers.

#### 6.1.3 Malfunctioning of infrared emitter

The infrared emitter was not emitting the infrared code in the first implementation. After analyzing the flow of data at each point in the hardware code, the problem identified was the method of encoding the raw signal the infrared remote Arduino library that was being used. The problem was solved by finding and using a simpler and better infrared remote Arduino library.

## 7. CONCLUSION

The popularity of IoT over the last two decades has made operating numerous devices from a central location easier and more seamless.

Earlier infrared controlled appliances cannot benefit from this technology as they do not possess an interface to connect to the network which is necessary for IoT. Hence, the researchers developed a control system to provide support for older model infrared appliances by providing an interface by which, various existing IR devices and systems could be controlled using current voice processing technology. This could be seen as an advantage of this system over existing reported studies. However, the developed system does not accommodate all devices and hence, the design couldn't factor in all devices and brands due to more of technical know-how and complexity of their encoding algorithm.

Also, it is important to know that for the system to work effectively, a good environment where it can be centrally installed and adequate maintenance practices will need to be provided. The system therefore offers a user-friendly platform and requires little to no orientation to understand its workings as long as the intended appliance is supported.

## 8. RECOMMENDATION

Although the projects provide support for a plethora of devices, there is still a large set of devices and brands it does not which could not be implemented due to time constraints, technical know-how and complexity of their infrared encoding algorithm. The system could be expanded to provide support for more devices. A variety of more control features could be implemented to make the system more dynamic.

## REFERENCES

[1] Kikel, C. (2019). A brief history of voice recognition technology. Total Voice Technology. <https://www.totalvoicetech.com/a-brief-history-of-voice-recognition-technology/>, accessed on Jan. 15,

2020.

[2] Pandaya, K. (2017). Voice controlled home automation system. Electronicforu.com. <http://electronicsforu.com/electronics-project/voice-controlled-home-automation-system>, accessed on Jan. 15, 2020.

[3] Saiskumar, A. (2016). Voice controlled smart home. International Journal of Emerging Technology and Advanced Engineering, 5(1): 2250-2459.

[4] Microcontrollers lab. (2017). Voice controlled Home Automation using adruino. Retrieved from <http://microcontrollerslab.com/voice-controlled-home-automation>, accessed on Jan. 15, 2020.

[5] Sen S., Chakrabarty, S., Toshniwal, R., Bhaumik, A. (2015). Design of an intelligent voice controlled home automation system. International Journal of Computer Applications, 121(15): 39-42. <https://doi.org/10.5120/21619-4904>

[6] Chaitanya, B.V.S., Sekhar, T.C., Ramseh, N.V.K. (2016). IoT based smart IR device using CC32000. Indian Journal of Science and Technology, 9(16). <https://doi.org/10.17485/ijst/2016/v9i16/87057>

[7] Parameshachari, B.D., Gopy, S.K., Hurry, G., Gopaul, T.T. (2013). A study on smart home control system through speech. International Journal of Computer Applications, 69(19): 30-39. <https://doi.org/10.5120/12080-8244>.

[8] Ali, A.T., Eltayeb, E.B., Abusail, E.A.A. (2017). Voice recognition based smart home control system. International Journal of Engineering Inventions, 6(4): 1-5.

[9] Mahmud, K., Joarder, M.M.A., Roy, A.K. (2015). Research article voice activated electronic devices control system for home appliances. Scholars Journal of Engineering and Technology, 3(1B): 66-70.

[10] Chien, J.R.C., Tai, C.C. (2004). The information home appliance control system a bluetooth universal type remote controller. In IEEE International Conference on Networking, Sensing and Control, IEEE, 1: 399-400.

[11] Ayres, T., Nolan, B. (2005). Voice activated command and control with speech recognition over WiFi. Science of Computer Programming, 59(1-2): 109-126. <https://doi.org/10.1016/j.scico.2005.07.007>

[12] Marimuthu, R., Balamurugan, S., Darshit, H.P., Sivanantham, S. (2016). Voice controlled energy management system. Asian Journal of Applied Sciences, 10(1): 25-31. <https://doi.org/10.3923/ajaps.2017.25.31>

[13] Gundogdu, K., Bayrakdar, S., Yucedag, I. (2018). Developing and modeling of voice control system for prosthetic robot arm in medical systems. Journal of King Saud University – Computer and Information Sciences, 30(2). <https://doi.org/10.1016/j.jksuci.2017.04.005>

[14] Ambalkar, R.R., Pagrut, H.P., Satputale, S.S., Kedar, D.A. (2017). A review paper on smart home using voice recognition. 2nd National Conference Recent Innovations in Science and Engineering, 5(9): 79-82.

[15] Gunasekhar, V.A., Lavanya, Sankaranarayanan, S. (2017). IoT based smart energy management system. International Journal of Applied Engineering Research, 12(16): 5455-5462.

[16] MIT App Inventor (2012). Getting started with MIT App Inventor. <https://appinventor.mit.edu/explore/get-started>, accessed on Jan. 15, 2020.