Voice Recognized and Password Protected Locker with Arduino

Ruchika Kumari
Department of ECE, GGSIP University
Maharaja Surajmal Institute of Technology Delhi, India

Abstract—The objective of paper is to develop the speech recognition system using interfacing of different modules such as VR Module, LCD and Keypad Matrix with an Arduino Microcontroller. It also describes how the speech recognition followed by interpretations of the algorithm is used to rotate motors of a security locker accurately. Voice Recognition module V2 is used to extract the characteristics from the input speech signal with respect to particular words uttered by the owner of the locker. Then Arduino is used to identify the speaker by matching the keywords and granting access to the secure system.

Keywords—Arduino, LCD, Locker, VR Module

I. INTRODUCTION

Lockers security is one of the areas where advancements in technologies have not been utilized to the fullest potential. In material handling in banks or in colleges or access systems, optimized security is mandatory. For example in a jewelry shop, even today very basic practices are followed that can be made lot better using technologies like voice recognition which is not only easy to access but also secure at the same time.

In this project we take up a 2 level safety system and propose a fault proof way for enhancing the security in lockers. In the existing locker system, all individuals require the locker number and a key. These can be easily accessed or duplicated. If any individual has these then they can easily access the locker because there is no other verification or authentication involved. However, a combination of similar speech with 3 or more exact command added with a password would be quite a bit impossible to hack.

Here, the locker is provided with a simple and low cost embedded system that controls the lock to the locker instead of key. This system comprises of a Liquid crystal display mounted on the locker itself. Arduino is connected to the motors of the locker that in turn rotates, provided that the speech matches and password is correctly entered by the customer.

Additionally, a single project can access multiple lockers and store the database for more than a dozen lockers. A particular locker will open up from an array of lockers corresponding to the matching keywords which the Arduino receives via the module.

II. VR MODULE

This Voice Recognition Module V3 is a compact and easy-control speaking recognition board. It supports up to 80 voice commands in all. Maximum 7 voice commands could work at the same time. Any sound could be trained as command. Users need to train the module first before let it recognizing any voice command. [1]

A. Working Principle

HM2007 is a single chip CMOS voice recognition module. It is an on-chip analog front end large-scale integrated circuit with voice analysis, speech recognition and voice recognition system control processes. It can be operated in two modes:

Manual mode: In manual mode of operation, the voice recognition module HM2007 is used to build simple recognition system by connecting components such as keypad, SRAM of 8 Kbyte memory and other components. If power is on, then the HM2007 starts an initialization process and, if the WAIT pin is L, then the HM2007 checks external memory: 8 Kbyte SRAM – whether it is perfect or not. But, if the WAIT is H, then the memory check process will be skipped by the HM2007. After this initialization process, the HM2007 moves into recognition mode. In the recognition mode, if the WAIT pin is H, then the HM2007 is ready to accept the voice and begins recognition process. If the WAIT pin is L, then the HM2007 doesn’t accept the voice that is to be recognized. Then training new pattern by clearing trained pattern are the two operations performed using function keys: TRN and CLR. All the patterns in the HM2007 can be cleared by entering number key 99 and pressing CLR. The pin configuration of HM2007 is shown in Figure 1.

CPU control mode: In the CPU control mode of operation, several functions such as RECOG, TRAIN, RESULT, UPLOAD, DOWNLOAD, RESET functions of the voice recognition module HM2007 are provided in this mode. Similar to a manual mode of operation, in this mode also power on, recognition, training, resulting, uploading, downloading and reset operations are performed based on different criteria. [2]
B. Parameters
- Voltage: 4.5-5.5V
- Current: <40mA
- Digital Interface: 5V TTL level UART interface
- Analog Interface: 3.5mm mono-channel microphone connector + microphone pin interface
- Size: 30mm x 47.5mm

C. Features
- Support maximum 80 voice commands, with each voice command 1500ms (one or two words speaking)
- Maximum 7 voice commands effective at same time
- Arduino library is supplied
- Easy Control: UART/GPIO
- User-control General Pin Output Terminology

D. Hardware and Software Preparation
1. Connect your Voice Recognition V3 Module with Arduino, By Default:
   - Arduino 5V → 5V
   - 2 → TX
   - 3 → RX
   - GND → GND
2. Download VoiceRecognitionV3 library.(download zip file or use git clone https://github.com/elechouse/VoiceRecognitionV3.git command)
3. If using zip file, extract VoiceRecognitionV3.zip to ArduinoSketch\libraries folder, or if you use git clone command copy VoiceRecognitionV3 to ArduinoSketch\libraries.

III. ALGORITHM

A. Flowchart
   Figure 2 explains the algorithm of the project in the form of a flowchart. It explains the flow of the content and considers the different cases which can be encountered.

B. Working
   With reference to the Figure 3 the working of the project can be illustrated as follows:
   1. The voice input, “Wake,” from mike is detected by the Voice Recognition module.
   2. Arduino interprets and then allows the LCD to display the message “Welcome, *name 1*”.
   3. If the phrase “Wake” doesn’t matches the predefined code, then system remains in sleep mode.
4. However, if the phrase matches, then the person follows up with the voice command “Grant”.
5. If the phrase doesn’t match the code word, then the Arduino notifies through interfaced device and request the user to “speak again” with correct phrase.
6. If matched, The LCD displays the message “Enter your password”.
7. The user enters the seven digit numeric password on the keypad matrix.
8. If the password is correct, the message “Access granted” is displayed on the LCD, Following this, the Door of the locker opens.
9. Else, if the password is incorrect, it prompts “Invalid password”.
10. If the incorrect counter goes to three then the security alarm will trip for some time and system will go to sleep mode.
11. Also, the user can go back to the home screen by saying, “Close”.
12. The display on the LCD prompts the message, “Thank you, *name 1*”.
13. If the user wants to change the password, it can be changed during the active mode by saying “Change”.
14. The LCD displays, “Enter current password”.
15. The password is to be entered using numeric keypad.
16. The LCD displays “Enter new password”.
17. The new password is to be entered using numeric keypad.
18. Then message prompts on LCD,”Password Changed”.

IV. CODE
The functionality loop of Arduino, ‘void loop’ is an infinite loop. The project has the following code which runs for the secure system.

```c
void loop()
{
  label_12 : ret = myVR.recognize(buf, 50);
  if(ret>0)
  {
    if (buf[1] == wakeup) // wake up
    {
      while(1)
      {
        lcd.setCursor(0,0);
        lcd.print("WELCOME, USER!!!");
        label_1 : ret = myVR.recognize(buf, 50);
        if (buf[1] == grantaccess ) // grant me access
        {
          label_2 : while(1)
          {
            key = password();
            if(key == 6 || key == 7)
            {
              delay(1000);
              goto label5;
            }
          }
        }
        label5 : if (key == 6)//true
        {
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("ACCESS GRANTED");
          delay(1000);
          digitalWrite(pin1, HIGH);
          digitalWrite(pin2, LOW);
          delay(50);
          digitalWrite(pin1, LOW);
          digitalWrite(pin2, LOW);
          delay(1000);
          digitalWrite(pin1, HIGH);
          digitalWrite(pin2, LOW);
          delay(50);
          digitalWrite(pin1, LOW);
          digitalWrite(pin2, LOW);
          delay(1000);
          digitalWrite(pin1, HIGH);
          digitalWrite(pin2, LOW);
          delay(50);
          digitalWrite(pin1, LOW);
          digitalWrite(pin2, LOW);
          delay(1000);
          digitalWrite(pin1, HIGH);
          digitalWrite(pin2, LOW);
          delay(50);
          digitalWrite(pin1, LOW);
          digitalWrite(pin2, LOW);
          delay(1000);
          while(1)
          {
            label_3 : ret = myVR.recognize(buf, 50);
            if (buf[1] == closelock) // close my locker
            {
              lcd.clear();
              lcd.setCursor(0,0);
              lcd.print("Thank you user");
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, HIGH);
              delay(60);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(60);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, HIGH);
              delay(60);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, HIGH);
              delay(60);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, HIGH);
              delay(60);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, HIGH);
              delay(60);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
              digitalWrite(pin1, LOW);
              digitalWrite(pin2, LOW);
              delay(1000);
          }
          if (buf[1] != closelock) //any thing else
          {
            delay(2000);
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("SAY 'close' OR");
            lcd.setCursor(0,1);
            lcd.print("DO MANUALLY");
            lcd.setCursor(0,0);
            lcd.print("Thank you user");
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, HIGH);
            delay(60);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(60);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, HIGH);
            delay(60);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, HIGH);
            delay(60);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, HIGH);
            delay(60);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, HIGH);
            delay(60);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, HIGH);
            delay(60);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
            digitalWrite(pin1, LOW);
            digitalWrite(pin2, LOW);
            delay(1000);
          }
        }
      }
    }
  }
}
```

if (Attempt > 2) {
    digitalWrite(pin3, HIGH);//alarm on
    delay(10000);
    digitalWrite(pin3, LOW);//alarm off
    lcd.clear();
    ret = 0;
    Attempt = 0;
    goto label_12;
} else
    goto label_2;
}

if (buf[1] != grantaccess)//any thing else
    { grant++;
        if (grant>3)
            goto label_1 ;
    }
}
printVR(buf);  // voice recognized][4]

VI. ADVANTAGES

- Robust application and keyless security system.
- Here, as we are using 3 step verification, security is beefed up to 3 times.
- Mistakes at any level, will considered hostile. Password can be changed by the owner.
- If password is entered incorrectly three times, then the alarm is tripped.
- It is a user-friendly interactive system.
- Hacking the software is difficult.

VII. APPLICATIONS

- Bank security system
- Jewelry shop
- Personal lockers
- College sports room

VII. CURRENT AND FUTURE USES

Currently speech recognition is used in many fields like Voice Recognition System for the Visually Impaired [5] highlights the Mg Sys Visi system that has the capability of access to World Wide Web by browsing in the Internet, checking, sending and receiving email, searching in the Internet, and listening to the content of the search only by giving a voice command to the system. In addition, the system is built with a translator that has the functionality to convert html codes to voice; voice to Braille and then to text again. This system comprises of five modules namely: Automatic Speech Recognition (ASR), Text-to-Speech (TTS), Search engine, Print (Text-Braille) and Translator (Text-to-Braille and Braille-to -Text) module, was originally designed and developed for the visually impaired learners, can be used for other users of specially needs like the elderly, and the physically impaired learners. Speech Recognition in Radiology Information System. The Radiology report is the fundamental means by which radiologists communicate with clinicians and patients. The traditional method of generating reports is time consuming and expensive. Recent advances in computer hardware and software technology have improved Speech Recognition systems used for radiology reporting. [6] Integration of Robust Voice Recognition and Navigation System on Mobile Robot [7] and there are many other fields in which speech recognition can be used.

VIII. CONCLUSION

In the speaker identification phase, VR techniques is used. This techniques provides more efficient speaker identification system. The coding of all the techniques mentioned above has been done using Arduino IDE. This project can be installed for several locker system.

REFERENCES