

SPEECH RECOGNITION

ARDUINO PROJECT



**Submitted by:**

Rochak Gupta (11808323)

Uttam Kumar Trivedi (11808579)

Diwanshu Chandra (11808583)

**Speech Encryptor**

Arduino Project

**Introduction**

This project is based on a Cyber Security concept of encryption, there are various algorithms for encrypting the data. Encryptions refers to modifying the content in such a way that no other person can understand it.

The project encrypts your speech with the help of your mobile device, by using an application. It fetches the content from your device using a Bluetooth module and encrypts the data using ROT-13 and Morse code.

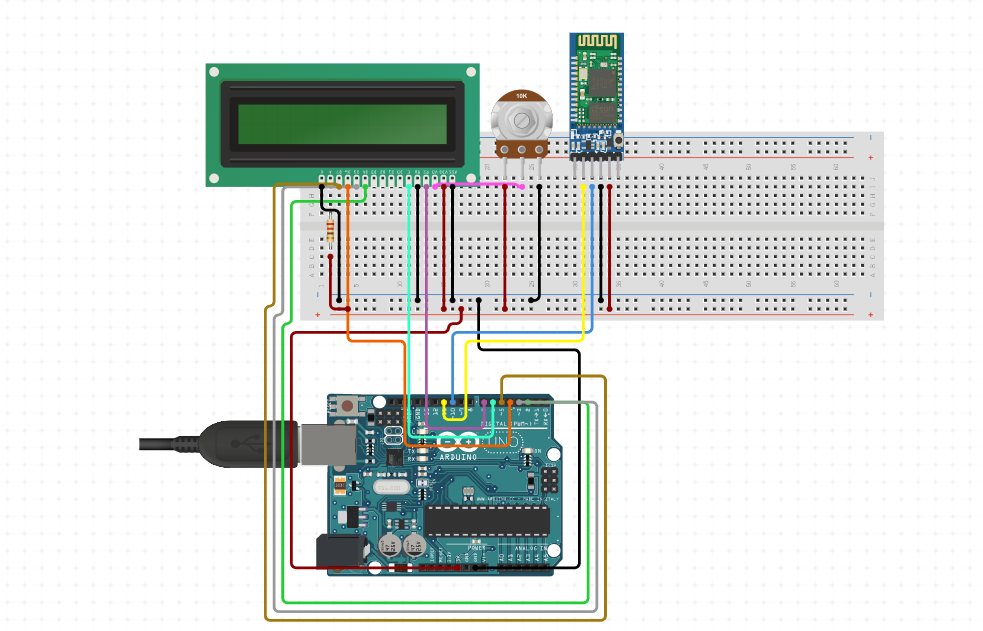
The result can also be maintained in your pc using a software called CoolTerm, which fetches the output from the Arduino through the port it is connected. The software is available in the following link.

**Link :-** <http://freeware.the-meiers.org/>

**Components**

* Arduino UNO
* HC-05 (Bluetooth Module)
* LCD (16x2)
* Resistors or Potential Meter
* Software :-
  + - Arduino Voice Recognition
    - CoolTerm

**Connections**



**Connections**

|  |  |
| --- | --- |
| LCD Connection | Arduino |
| VSS | GND |
| VDD | VCC |
| V0 | Potential meter |
| RS | 12 |
| RW | GND |
| E | 11 |
| D4 | 2 |
| D5 | 3 |
| D6 | 4 |
| D7 | 5 |
| A | VCC |
| K | GND |

|  |  |
| --- | --- |
| HC-05 | Arduino |
| VCC | VCC |
| GND | GND |
| TXD | RXD |
| RXD | TXD |

**CODE**

#include <LiquidCrystal.h>

String voice;

String old;

String encrypt="";

String encrypt2="";

String morse[26]={".-","-...","-.-.","-..",".","..-.","--.","....","..",

".---","-.-",".-..","--","-.","---",".--.","--.-",".-.","...","-","..-",

"...-",".--","-..-","-.--","--.."};

char ch;

const int rs = 12, en = 11, d4 = 2, d5 = 3, d6 = 4, d7 = 5;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {

Serial.begin(9600);

lcd.begin(16, 2);

}

void loop() {

while (Serial.available())

{

delay(10);

char c = Serial.read();

if (c == '#') {break;}

voice += c;

}

if (voice.length() > 1) {

Serial.println("\n"+voice);

//ROT-13 Encryption

for(int i=0;i<voice.length();i++)

{

int num=voice[i];

if(num>=65 && num<=90)

{

num=num%65;

num=num+13+65;

if(num>90)

{

int rem=num%91;

num=65+rem;

}

ch=num;

}

if(num>=97 && num<=122)

{

num=num%97;

num=num+13+97;

if(num>122)

{

int rem=num%123;

num=97+rem;

}

ch=num;

}

if(num==32)

{

ch=num;

}

if(encrypt.length()<voice.length())

{

encrypt=encrypt+ch;

}

}

//Morse Code Encryption

for(int i=0;i<voice.length();i++)

{

int num2=voice[i];

if(num2>=65 && num2<=90)

{

num2=num2%65;

encrypt2+=morse[num2];

encrypt2+=" ";

}

if(num2>=97 && num2<=122)

{

num2=num2%97;

encrypt2+=morse[num2];

encrypt2+=" ";

}

if(num2==32)

{

encrypt2+=" ";

}

}

Serial.println(encrypt);

Serial.println(encrypt2);

}

Serial.print(" ");

lcd.setCursor(0,0);

lcd.print(encrypt2);

lcd.setCursor(0,1);

lcd.print(encrypt);

for (int positionCounter = 0; positionCounter < 3; positionCounter++) {

lcd.scrollDisplayLeft();

}

delay(1000);

voice="";

encrypt="";

encrypt2="";

}

**SCOPE OF PROJECT**

The project is still on initial stage, but it has a potential of encrypting long conversations as well.

With more modifications, we can provide a voice encrypting tool which can encrypt a long meeting in a very precise way for privacy and security.

If you want to send the meeting details to someone, you don’t need to type everything again, as it will create a file of the conversation in encrypted format.

As not any other user will be able to understand it.