#include<ESP8266WiFi.h>

#include<FirebaseArduino.h>

#include<ArduinoJson.h>

unsigned long signal\_len,t1,t2; //time for which button is pressed

#define inputPin D3 //input pin for push button

#define ledPin D2 //output pin for LED

#define buzzerpin D1 //output pin for buzzer

String code = ""; //string in which one alphabet is stored

int i = 0;

int code\_strength = 0; //for differentiating letters(4) and numbers(5).

#define FIREBASE\_HOST "postman-99991.firebaseio.com" // the project name address from firebase id

#define FIREBASE\_AUTH "ODx9GP58daGUiyYiHYIwq1nhSJjvYb6RxlVkzbYU" // the secret key generated from firebase

#define WIFI\_SSID "Abhijeet" //WiFi Name

#define WIFI\_PASSWORD "123456789" //WiFi Password

String text = ""; //string in which actual message is stored

char readio() //To read digital signal as dot or dash

{

if (signal\_len < 600 && signal\_len > 50)

{

return '.'; //if button press less than 0.6sec, it is a dot

}

else if (signal\_len > 600)

{

return '-'; //if button press more than 0.6sec, it is a dash

}

}

void convertor() //Used to convert morse code to its corresponding letter

{

code\_strength = code.length();

Serial.print("code mass: ");

Serial.println(code\_strength);

static String letters[] = {".-", "-...", "-.-.", "-..", ".", "..-.", "--.", "....", "..", ".---", "-.-", ".-..", "--", "-.", "---", ".--.", "--.-",

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

};

static String numbers[] = {"-----",".----","..---","...--","....-",".....","-....","--...","---..","----.","E" };

int i = 0;

if (code\_strength <= 4) //if code strength is less than or equal to 4 its is a letter

{

while (letters[i] != "E") //loop for comparing input code with letters array

{

if (letters[i] == code)

{

text += char('A' + i);

Serial.println(text);

Firebase.setString("Message",text); //To send text message to firebase

delay(500);

break;

}

i++;

}

if (letters[i] == "E")

{

Serial.println("<Wrong input>"); //if input code doesn't match any letter, error

text = "";

}

code = ""; //reset code to blank string

}

else if(code\_strength == 5)

{ //if code strength is 5 it is a number.

while (numbers[i] != "E") //loop for comparing input code with letters array

{

if (numbers[i] == code)

{

text += char('0' + i);

Serial.println(text);

Firebase.setString("Message",text); //To send text message to firebase

delay(500);

break;

}

i++;

}

if (numbers[i] == "E")

{

Serial.println("<Wrong input>"); //if input code doesn't match any letter, error

text = "";

}

code = ""; //reset code to blank string

}

else{

Serial.println("<Wrong input>"); //if input code doesn't match any letter, error

text = "";

code = "";

}

}

void setup() {

Serial.begin(115200);

pinMode(inputPin, INPUT\_PULLUP); //internal pullup resistor is used to simplify the circuit

pinMode(ledPin,OUTPUT);

pinMode(buzzerpin,OUTPUT);

WiFi.begin(WIFI\_SSID,WIFI\_PASSWORD);

Serial.print("Connecting to ");

Serial.print(WIFI\_SSID);

while(WiFi.status()!= WL\_CONNECTED)

{

Serial.print(".");

delay(500);

}

Serial.println();

Serial.print("Connected to ");

Serial.println(WIFI\_SSID);

Serial.print("IP ADDRESS is : ");

Serial.println(WiFi.localIP());

Firebase.begin(FIREBASE\_HOST, FIREBASE\_AUTH); // connect to firebase

Firebase.setString("Message",""); //send initial string of status

}

void loop()

{

NextDotDash:

while (digitalRead(inputPin) == HIGH) {ESP.wdtFeed();} //To feed watchdog timer before it bites

t1 = millis(); //time at button press

digitalWrite(ledPin, HIGH); //LED on while button pressed

tone(buzzerpin,2500); //buzz at particular frequency

while (digitalRead(inputPin) == LOW) {ESP.wdtFeed();}

t2 = millis(); //time at button release

digitalWrite(ledPin, LOW); //LED off on button release

noTone(buzzerpin); //To stop buzz effect

signal\_len = t2 - t1; //time for which button is pressed

if (signal\_len > 50) //to account for switch debouncing

{

code += readio(); //function to read dot or dash

}

while ((millis() - t2) < 500) //if time between button press greater than 0.5sec, skip loop and go to next alphabet

{

if (digitalRead(inputPin) == LOW)

{

goto NextDotDash;

}

}

convertor(); //function to decipher code into alphabet

}