

/*

MECH 307 Group Project Arduino Code
Fall 2014
Group 31

-Code integrates weather sensors (Thermistor, BMP183 Barometric Pressure/Altitude Sensor, and Wind Speed Sensor (Anemometer)) with LCD displays, LED lighting, and buttons.

*/

```
// Defined thermistor analog pin-----
#define THERMISTORPIN A1
// resistance at 25 degrees C
#define THERMISTORNOMINAL 10000
// temp. for nominal resistance (almost always 25 C)
#define TEMPERATURENOMINAL 25
// how many samples to take and average, more takes longer
// but is more 'smooth'
#define NUMSAMPLES 5
// The beta coefficient of the thermistor (usually 3000-4000)
#define BCOEFFICIENT 3950
// the value of the 'other' resistor
#define SERIESRESISTOR 10000
//Thermistor
int samples[NUMSAMPLES];

//Define Anemometer pin and initialization-----
#define ANEMOMETER A4
//const int ANEMOMETER = A5;0
int sensorValue = 0;
float sensorVoltage = 0.0;
float windSpeed = 0.0;

// Initialize BMP183 Sensor-----
#include <Adafruit_Sensor.h>
#include <Adafruit_BMP183.h>
#define BMP183_CLK 3
#define BMP183_SDO 5 // AKA MISO
#define BMP183_SDI 4 // AKA MOSI
#define BMP183_CS 2
Adafruit_BMP183 bmp = Adafruit_BMP183(BMP183_CLK, BMP183_SDO, BMP183_SDI,
BMP183_CS);

//Define LCD pins and include LCD arduino library-----
#include <LiquidCrystal.h>
//          BS E D4 D5 D6 D7
LiquidCrystal lcd1(7, 8, 9, 10, 11, 12);
LiquidCrystal lcd2(7, 6, 9, 10, 11, 12);

// Include SPI Library-----
```

```
#include <SPI.h>
```

```
//Define Button Pins-----
```

```
int ButtonUnit = 13;  
int ButtonLightShow = 20;  
int ButtonSOS = 19;  
int ButtonSound = 18;
```

```
// Initialize Piezo Buzzer-----
```

```
int SPKR = 21;  
int val = 0; // variable to store the read value  
int tempPin = 0;
```

```
//Define LED pins -----
```

```
int led22 = 22;  
int led23 = 23;  
int led24 = 24;  
int led25 = 25;  
int led26 = 26;  
int led27 = 27;  
int led28 = 28;  
int led29 = 29;  
int led30 = 30;  
int led31 = 31;  
int led32 = 32;  
int led33 = 33;  
int led34 = 34;  
int led35 = 35;  
int led37 = 37;  
int led38 = 38;  
int led39 = 39;  
int led40 = 40;  
int led41 = 41;  
int led42 = 42;  
int led43 = 43;  
int led44 = 44;  
int led45 = 45;  
int led46 = 46;  
int led47 = 47;  
int led48 = 48;  
int led49 = 49;  
int led50 = 50;  
int led51 = 51;  
int led52 = 52;  
int led53 = 53;
```

```
// MAIN SETUP for program-----
```

```
// Only ran once*****
```

```

void setup(void) {

// Begins serial data transmission
Serial.begin(9600);
analogReference(EXTERNAL);

//Configure Piezo Buzzer to act as output
pinMode(SPKR, OUTPUT);

//Introduction LCD Message-----
lcd1.begin(20, 4);
lcd2.begin(20, 4);
lcd2.setCursor(0, 0);
lcd2.print("Hello, Im P.E.W.E.");
for (int i=0; i<500; i++) { // generate a 1KHz tone for 1/2 second
  digitalWrite(SPKR, HIGH);
  delayMicroseconds(4000);
  digitalWrite(SPKR, LOW);
}
lcd2.clear();
lcd2.setCursor(0, 0);
lcd2.print("I can make weather");
lcd2.setCursor(0, 1);
lcd2.print("recommendations");
lcd2.setCursor(0, 2);
lcd2.print("based on current");
lcd2.setCursor(0, 3);
lcd2.print("conditions.");
delay(4000);
lcd2.clear();

// Begin Pressure Sensor
bmp.begin();

// Configure buttons to act as inputs
//Writes the buttons as high or low

pinMode(ButtonUnit, INPUT);
digitalWrite(ButtonUnit, LOW);

pinMode(ButtonLightShow, INPUT);
digitalWrite(ButtonLightShow, LOW);

pinMode(ButtonSOS, INPUT);
digitalWrite(ButtonSOS, HIGH);

pinMode(ButtonSound, INPUT);
digitalWrite(ButtonSound, HIGH);

//Configures LED's to act as outputs
pinMode(led22, OUTPUT);

```

```

pinMode(led23, OUTPUT);
pinMode(led24, OUTPUT);
pinMode(led25, OUTPUT);
pinMode(led26, OUTPUT);
pinMode(led27, OUTPUT);
pinMode(led28, OUTPUT);
pinMode(led29, OUTPUT);
pinMode(led30, OUTPUT);
pinMode(led31, OUTPUT);
pinMode(led32, OUTPUT);
pinMode(led33, OUTPUT);
pinMode(led34, OUTPUT);
pinMode(led35, OUTPUT);
pinMode(led37, OUTPUT);
pinMode(led38, OUTPUT);
pinMode(led39, OUTPUT);
pinMode(led40, OUTPUT);
pinMode(led41, OUTPUT);
pinMode(led42, OUTPUT);
pinMode(led43, OUTPUT);
pinMode(led44, OUTPUT);
pinMode(led45, OUTPUT);
pinMode(led46, OUTPUT);
pinMode(led47, OUTPUT);
pinMode(led48, OUTPUT);
pinMode(led49, OUTPUT);
pinMode(led50, OUTPUT);
pinMode(led51, OUTPUT);
pinMode(led52, OUTPUT);
pinMode(led53, OUTPUT);
}

//End of main setup-----

//MAIN LOOP-----
//Loops consecutively, allowing program to change and respond-----

void loop(void) {
  uint8_t i;
  float average; //Set as floating point type, a number with a decimal point

  //Anemometer analog value read
  sensorValue = analogRead(ANEMOMETER); //reads value from analog pin
  sensorVoltage = sensorValue * .004882814; // Convert from 0...1024 to 0...5v
  windSpeed = 20*((sensorVoltage)-0.5858); //Converts voltage to meters per second
  Serial.print("Sensor Value: ");
  Serial.print(sensorValue);
  Serial.println("\t");
  Serial.print("Sensor Voltage: ");

```

```

Serial.println(sensorVoltage);
delay(100);

//Thermistor analog read-----
// take N samples in a row, with a slight delay
for (i=0; i< NUMSAMPLES; i++) {
  samples[i] = analogRead(THERMISTORPIN); //reads value from analog pin
  delay(10);

// average all the samples out
average = 0;
for (i=0; i< NUMSAMPLES; i++) {
  average += samples[i];
}
average /= NUMSAMPLES;

// Prints data to serial port to compare with LCD values
Serial.print("Average analog reading ");
Serial.println(average);

// convert the value to resistance
average = 1023 / average - 1;
average = SERIESRESISTOR / average;
Serial.print("Thermistor resistance ");
Serial.println(average);

//Converts voltage to temperature via the Steinhart equation
float steinhartC;
steinhartC = average / THERMISTORNOMINAL; // (R/Ro)
steinhartC = log(steinhartC); // ln(R/Ro)
steinhartC /= BCOEFFICIENT; // 1/B * ln(R/Ro)
steinhartC += 1.0 / (TEMPERATURENOMINAL + 273.15); // + (1/To)
steinhartC = 1.0 / steinhartC; // Invert
steinhartC -= 273.15; // convert to C

float steinhartF;
steinhartF = steinhartC*9.0/5.0 + 32.0; // convert to F

float BaroPSI;
BaroPSI = (bmp.getPressure())*0.000145037738; // Converts pascals to PSI

float windMPH;
windMPH = (windSpeed)*2.23694; // Converts meters per second to MPH

float altFT;
float seaLevelPressure = 1013.25; //Assigns sea level pressure for reference
altFT = (bmp.getAltitude(seaLevelPressure))*3.28084; //Converts to ambient altitude

//Configures buttons to read values as high or low

```

```
int ButtonStateUnit = digitalRead(ButtonUnit);
int ButtonStateLightShow = digitalRead(ButtonLightShow);
int ButtonStateSOS = digitalRead(ButtonSOS);
int ButtonStateSound = digitalRead(ButtonSound);
```

```
// LCD1 PARAMETER OUTPUT
```

```
METRIC-----
```

```
-
if (ButtonStateUnit == LOW){ //If Unit button is pushed
```

```
    // lcd1.clear();
    lcd1.setCursor(0, 0);
    lcd1.print("Temp: ");
    lcd1.print(steinhartC);
    lcd1.print(" C");
    //Anemometer-----
    lcd1.setCursor(0, 4);
    lcd1.print("Wind: ");
    lcd1.print(windSpeed);
    lcd1.print(" m/s");
```

```
    //BMP183-----
```

```
    lcd1.setCursor(0, 1);
    lcd1.print("Pressure: ");
    lcd1.print( bmp.getPressure() );
    lcd1.print(" Pa");
```

```
    float seaLevelPressure = 1013.25;
    lcd1.setCursor(0, 2);
    lcd1.print("Altitude: ");
    lcd1.print(bmp.getAltitude(seaLevelPressure));
    lcd1.print(" m");
```

```
    //LCD2 WEATHER LOGIC METRIC -----
```

```
    //High temp alert
    lcd2.clear();
    if (steinhartC > 32.2){
        lcd2.setCursor(0, 1);
        lcd2.print("It's hot out here! ");
        lcd2.setCursor(0, 2);
        lcd2.print("Let's go to the pool ");
        digitalWrite(led43, HIGH);
```

```
    // Piezo buzzer alert
    for (int i=0; i<500; i++) { // generate a 1KHz tone for 1/2 second
        digitalWrite(SPKR, HIGH);
        delayMicroseconds(500);
        digitalWrite(SPKR, LOW);
        delayMicroseconds(500);
    }
```

```

}
//Low temp alert
if (steinhartC < 0.0){
  lcd2.setCursor(0, 1);
  lcd2.print("Below Freezing      ");
  lcd2.setCursor(0, 2);
  lcd2.print("Wear your long johns  ");
  digitalWrite(led42, HIGH);
}
// High wind alert
if( windSpeed > 15.6464){
  lcd2.setCursor(0, 1);
  lcd2.print("Its windy outside    ");
  lcd2.setCursor(0, 2);
  lcd2.print("Go fly a kite!      ");
  digitalWrite(led40, HIGH);
}
//Wind chill alert
if ((steinhartC < 4.4) && (windSpeed > 5.36448)){
  lcd2.setCursor(0,3);
  lcd2.print("Wind Chill Warning  ");
}
//God weather alert
if ((steinhartC > 18.3) && (steinhartC < 29.4) && (windSpeed < 2.2352)){
  lcd2.setCursor(0,1);
  lcd2.print("The weather is nice.");
  lcd2.setCursor(0,2);
  lcd2.print("Go have fun!      ");
}
else{
  lcd2.setCursor(0,1);
  lcd2.print("");
  lcd2.setCursor(0,2);
  lcd2.print("");
}
//Storm Alert
if (( bmp.getPressure() < 82737.08736) && (windSpeed > 3.12928 )){
  lcd2.setCursor(0,1);
  lcd2.print("A storm is brewin'!");
  digitalWrite(led38, HIGH);
}
//High altitude alert
if ( bmp.getAltitude(seaLevelPressure) > 1502.664){
  lcd2.setCursor(0,0);
  lcd2.print("Wow!!                ");
  lcd2.setCursor(0,1);
  lcd2.print("Its great up here! ");
  lcd2.setCursor(0,2);
  lcd2.print("Enjoy the view!    ");
  digitalWrite(led41, HIGH);
}
else{

```

```

    digitalWrite(led41, LOW);
}
//Various screen resets -----
if (steinhartC < 32.2){
    digitalWrite(led43, LOW);

}
if (steinhartC > 0.0){
    digitalWrite(led42, LOW);

}
if( windSpeed < 6.7056){
    digitalWrite(led40, LOW);
}
if (bmp.getPressure() > 82737.08736){
    digitalWrite(led38, LOW);
}
//-----LED-TEMP LIGHTS-C-----

if (steinhartC < 21.1){
    digitalWrite(led44, HIGH);
    digitalWrite(led45, LOW);
    digitalWrite(led46, LOW);
    digitalWrite(led47, LOW);
    digitalWrite(led48, LOW);
}

if ((steinhartC > 21.2) && (steinhartC < 23.9)) {
    digitalWrite(led44, HIGH);
    digitalWrite(led45, HIGH);
    digitalWrite(led46, LOW);
    digitalWrite(led47, LOW);
    digitalWrite(led48, LOW);
}

if ((steinhartC > 24.0) && (steinhartC < 26.7)) {
    digitalWrite(led44, HIGH);
    digitalWrite(led45, HIGH);
    digitalWrite(led46, HIGH);
    digitalWrite(led47, LOW);
    digitalWrite(led48, LOW);
}

if ((steinhartC > 26.8) && (steinhartC < 30.5)) {
    digitalWrite(led44, HIGH);
    digitalWrite(led45, HIGH);
    digitalWrite(led46, HIGH);
    digitalWrite(led47, HIGH);
    digitalWrite(led48, LOW);
}
}

```



```
if ((steinhartC > 30.6) && (steinhartC < 31.1)) {  
    digitalWrite(led44, HIGH);  
    digitalWrite(led45, HIGH);  
    digitalWrite(led46, HIGH);  
    digitalWrite(led47, HIGH);  
    digitalWrite(led48, HIGH);  
}
```

```
//---LED-WIND---M/S-----
```

```
if( windSpeed < 0.89 ){  
    digitalWrite(led49, LOW);  
    digitalWrite(led50, LOW);  
    digitalWrite(led51, LOW);  
    digitalWrite(led52, LOW);  
    digitalWrite(led53, LOW);  
}
```

```
if( windSpeed > 2.24 ){  
    digitalWrite(led49, HIGH);  
    digitalWrite(led50, LOW);  
    digitalWrite(led51, LOW);  
    digitalWrite(led52, LOW);  
    digitalWrite(led53, LOW);  
}
```

```
if( windSpeed > 3.13 ){  
    digitalWrite(led49, HIGH);  
    digitalWrite(led50, HIGH);  
    digitalWrite(led51, LOW);  
    digitalWrite(led52, LOW);  
    digitalWrite(led53, LOW);  
}
```

```
if( windSpeed > 4.47){  
    digitalWrite(led49, HIGH);  
    digitalWrite(led50, HIGH);  
    digitalWrite(led51, HIGH);  
    digitalWrite(led52, LOW);  
    digitalWrite(led53, LOW);  
}
```

```
if( windSpeed > 5.36){  
    digitalWrite(led49, HIGH);  
    digitalWrite(led50, HIGH);  
    digitalWrite(led51, HIGH);  
    digitalWrite(led52, HIGH);  
    digitalWrite(led53, LOW);  
}
```

```
if( windSpeed > 6.71){  
    digitalWrite(led49, HIGH);  
    digitalWrite(led50, HIGH);  
    digitalWrite(led51, HIGH);  
    digitalWrite(led52, HIGH);  
    digitalWrite(led53, HIGH);  
}
```

```

}
// LCD1 PARAMETER OUTPUT
ENGLISH-----
----
if (ButtonStateUnit == HIGH){ //If unit button not pushed

    //lcd1.clear();
    lcd1.setCursor(0, 0);
    lcd1.print("Temp: ");
    lcd1.print(steinhartF);
    lcd1.print(" F");
    //Anemometer-----
    lcd1.setCursor(0, 4);
    lcd1.print("Wind: ");
    lcd1.print(windMPH);
    lcd1.print(" mph");

    //BMP183-----

    lcd1.setCursor(0, 1);
    lcd1.print("Pressure: ");
    lcd1.print( BaroPSI );
    lcd1.print(" PSI");

    float seaLevelPressure = 1013.25;
    lcd1.setCursor(0, 2);
    lcd1.print("Altitude: ");
    lcd1.print(altFT);
    lcd1.print(" ft");

    //LCD2 Weather Logic English -----

    //High temp alert
    lcd2.clear();
    if (steinhartF > 90.0){
        lcd2.setCursor(0, 1);
        lcd2.print(" It's hot out here! ");
        lcd2.setCursor(0, 2);
        lcd2.print("Let's go to the pool ");
        digitalWrite(led43, HIGH);
        for (int i=0; i<500; i++) { // generate a 1KHz tone for 1/2 second
            digitalWrite(SPKR, HIGH);
            delayMicroseconds(500);
            digitalWrite(SPKR, LOW);
            delayMicroseconds(500);
        }
    }
    //Low Temp alert
    if (steinhartF < 32.0){
        lcd2.setCursor(0, 1);
        lcd2.print("Below Freezing ");
        lcd2.setCursor(0, 2);

```

```

    lcd2.print("Wear your long johns");
    digitalWrite(led42, HIGH);
}
//High wind alert
if( windMPH > 35.0){
    lcd2.setCursor(0, 1);
    lcd2.print("Its windy outside  ");
    lcd2.setCursor(0, 2);
    lcd2.print("Go fly a kite!      ");
    digitalWrite(led40, HIGH);
    for (int i=0; i<500; i++) { // generate a 1KHz tone for 1/2 second
        digitalWrite(SPKR, HIGH);
        delayMicroseconds(500);
        digitalWrite(SPKR, LOW);
        delayMicroseconds(500);
    }
}
//Wind chill alert
if ((steinhartF < 40.0) && (windMPH > 12.0)){
    lcd2.setCursor(0,3);
    lcd2.print("Wind Chill Warning  ");
}
//Good Weather alert
if ((steinhartF > 65.0) && (steinhartF < 85.0) && (windMPH < 5.0)){
    lcd2.setCursor(0,1);
    lcd2.print("The weather is nice.");
    lcd2.setCursor(0,2);
    lcd2.print("Go have fun!      ");
}
else{
    lcd2.setCursor(0,1);
    lcd2.print("");
    lcd2.setCursor(0,2);
    lcd2.print("");
}
//Storm alert
if (( BaroPSI < 12.0) && (windMPH > 7.0 )){
    lcd2.setCursor(0,1);
    lcd2.print("A storm is brewin'!  ");
    digitalWrite(led38, HIGH);
}
//High altitude alert
if ( altFT > 4930.0){
    lcd2.setCursor(0,0);
    lcd2.print("Wow!!                ");
    lcd2.setCursor(0,1);
    lcd2.print("Its great up here!  ");
    lcd2.setCursor(0,2);
    lcd2.print("Enjoy the view!     ");
    digitalWrite(led41, HIGH);
}
else{

```

```

    digitalWrite(led41, LOW);
}

//Various screen resets
if (steinhartF < 85.0){
    digitalWrite(led43, LOW);

}
if (steinhartF > 32.0){
    digitalWrite(led42, LOW);

}
if( windMPH < 15.0){
    digitalWrite(led40, LOW);
}
if (BaroPSI > 12.0){
    digitalWrite(led38, LOW);
}
//-----LED----TEMP-F-----

if (steinhartF < 70.0){
    digitalWrite(led44, HIGH);
    digitalWrite(led45, LOW);
    digitalWrite(led46, LOW);
    digitalWrite(led47, LOW);
    digitalWrite(led48, LOW);
}

if ((steinhartF > 70.0) && (steinhartF < 75.0)) {
    digitalWrite(led44, HIGH);
    digitalWrite(led45, HIGH);
    digitalWrite(led46, LOW);
    digitalWrite(led47, LOW);
    digitalWrite(led48, LOW);
}

if ((steinhartF > 75.0) && (steinhartF < 80)) {
    digitalWrite(led44, HIGH);
    digitalWrite(led45, HIGH);
    digitalWrite(led46, HIGH);
    digitalWrite(led47, LOW);
    digitalWrite(led48, LOW);
}

if ((steinhartF > 85.0) && (steinhartF < 87)) {
    digitalWrite(led44, HIGH);
    digitalWrite(led45, HIGH);
    digitalWrite(led46, HIGH);
    digitalWrite(led47, HIGH);
    digitalWrite(led48, LOW);
}

```

```

if ((steinhartF > 88.0) && (steinhartF < 90)) {
    digitalWrite(led44, HIGH);
    digitalWrite(led45, HIGH);
    digitalWrite(led46, HIGH);
    digitalWrite(led47, HIGH);
    digitalWrite(led48, HIGH);
}
//---LED-WIND---MPH-----
if( windMPH < 2.0 ){
    digitalWrite(led49, LOW);
    digitalWrite(led50, LOW);
    digitalWrite(led51, LOW);
    digitalWrite(led52, LOW);
    digitalWrite(led53, LOW);
}
if( windMPH > 5.0 ){
    digitalWrite(led49, HIGH);
    digitalWrite(led50, LOW);
    digitalWrite(led51, LOW);
    digitalWrite(led52, LOW);
    digitalWrite(led53, LOW);
}
if( windMPH > 7.0 ){
    digitalWrite(led49, HIGH);
    digitalWrite(led50, HIGH);
    digitalWrite(led51, LOW);
    digitalWrite(led52, LOW);
    digitalWrite(led53, LOW);
}
if( windMPH > 10.0){
    digitalWrite(led49, HIGH);
    digitalWrite(led50, HIGH);
    digitalWrite(led51, HIGH);
    digitalWrite(led52, LOW);
    digitalWrite(led53, LOW);
}
if( windMPH > 12.0){
    digitalWrite(led49, HIGH);
    digitalWrite(led50, HIGH);
    digitalWrite(led51, HIGH);
    digitalWrite(led52, HIGH);
    digitalWrite(led53, LOW);
}
if( windMPH > 15.0){
    digitalWrite(led49, HIGH);
    digitalWrite(led50, HIGH);
    digitalWrite(led51, HIGH);
    digitalWrite(led52, HIGH);
    digitalWrite(led53, HIGH);
}
}
}

```

```
// LED wind Array Chaser pattern
if( windMPH > 3.0 ){
  digitalWrite(led29, HIGH);
  delay(50);
  digitalWrite(led34, HIGH);
  digitalWrite(led29, LOW);
  delay(50);
  digitalWrite(led35, HIGH);
  digitalWrite(led34, LOW);
  delay(50);
  digitalWrite(led33, HIGH);
  digitalWrite(led35, LOW);
  delay(50);
  digitalWrite(led32, HIGH);
  digitalWrite(led33, LOW);
  delay(50);
  digitalWrite(led31, HIGH);
  digitalWrite(led32, LOW);
  delay(50);
  digitalWrite(led30, HIGH);
  digitalWrite(led31, LOW);
  delay(50);
  digitalWrite(led25, HIGH);
  digitalWrite(led30, LOW);
  delay(50);
  digitalWrite(led28, HIGH);
  digitalWrite(led25, LOW);
  delay(50);
  digitalWrite(led27, HIGH);
  digitalWrite(led28, LOW);
  delay(50);
  digitalWrite(led24, HIGH);
  digitalWrite(led27, LOW);
  delay(50);
  digitalWrite(led26, HIGH);
  digitalWrite(led24, LOW);
  delay(50);
  digitalWrite(led22, HIGH);
  digitalWrite(led26, LOW);
  delay(50);
  digitalWrite(led23, HIGH);
  digitalWrite(led22, LOW);
  delay(50);
  digitalWrite(led23, LOW);
}
}
```

```
//-----LED-LIGHT-ARRAY-----  
-----
```

```
if(ButtonStateLightShow == LOW){
```

```
    // TURN FRONT LIGHTS OFF-----
```

```
    digitalWrite(led38, LOW);  
    digitalWrite(led39, LOW);  
    digitalWrite(led41, LOW);  
    digitalWrite(led42, LOW);  
    digitalWrite(led44, LOW);  
    digitalWrite(led45, LOW);  
    digitalWrite(led46, LOW);  
    digitalWrite(led47, LOW);  
    digitalWrite(led48, LOW);  
    digitalWrite(led49, LOW);  
    digitalWrite(led50, LOW);  
    digitalWrite(led51, LOW);  
    digitalWrite(led52, LOW);  
    digitalWrite(led53, LOW);  
    delay(50);
```

```
    // LIGHT ARRAY-----
```

```
    digitalWrite(led29, HIGH);  
    digitalWrite(led23, HIGH);  
    delay(100);  
    digitalWrite(led29, LOW);  
    digitalWrite(led23, LOW);  
    digitalWrite(led34, HIGH);  
    digitalWrite(led22, HIGH);  
    delay(100);  
    digitalWrite(led34, LOW);  
    digitalWrite(led22, LOW);  
    digitalWrite(led35, HIGH);  
    digitalWrite(led26, HIGH);  
    delay(100);  
    digitalWrite(led35, LOW);  
    digitalWrite(led26, LOW);  
    digitalWrite(led33, HIGH);  
    digitalWrite(led24, HIGH);  
    delay(100);  
    digitalWrite(led33, LOW);  
    digitalWrite(led24, LOW);  
    digitalWrite(led32, HIGH);  
    digitalWrite(led27, HIGH);  
    delay(100);  
    digitalWrite(led32, LOW);  
    digitalWrite(led27, LOW);  
    digitalWrite(led31, HIGH);  
    digitalWrite(led28, HIGH);  
    delay(100);
```

```
digitalWrite(led31, LOW);
digitalWrite(led28, LOW);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
delay(100);
digitalWrite(led25, LOW);
delay(150);
digitalWrite(led30, LOW);
delay(150);
```

```
// TEMP/WIND LIGHTS-----
```

```
digitalWrite(led53, HIGH);
digitalWrite(led48, HIGH);
delay(50);
digitalWrite(led53, LOW);
digitalWrite(led48, LOW);
digitalWrite(led52, HIGH);
digitalWrite(led47, HIGH);
delay(50);
digitalWrite(led52, LOW);
digitalWrite(led47, LOW);
digitalWrite(led51, HIGH);
digitalWrite(led46, HIGH);
delay(50);
```

```
digitalWrite(led51, LOW);
digitalWrite(led46, LOW);
digitalWrite(led50, HIGH);
digitalWrite(led45, HIGH);
delay(50);
```

```
digitalWrite(led50, LOW);
digitalWrite(led45, LOW);
digitalWrite(led44, HIGH);
digitalWrite(led49, HIGH);
delay(50);
digitalWrite(led44, LOW);
digitalWrite(led49, LOW);
delay(50);
```

```
// LOGIC LIGHTS -----
```

```
digitalWrite(led38, HIGH);
delay(100);
digitalWrite(led38, LOW);
digitalWrite(led40, HIGH);
delay(100);
digitalWrite(led40, LOW);
digitalWrite(led41, HIGH);
delay(100);
digitalWrite(led41, LOW);
digitalWrite(led42, HIGH);
```



```
delay(100);
digitalWrite(led42, LOW);
digitalWrite(led43, HIGH);
delay(100);
digitalWrite(led43, LOW);
digitalWrite(led39, HIGH);
delay(100);
digitalWrite(led39, LOW);
delay(100);
digitalWrite(led42, LOW);
//-----
digitalWrite(led39, HIGH);
delay(100);
digitalWrite(led39, LOW);
digitalWrite(led43, HIGH);
delay(100);
digitalWrite(led43, LOW);
digitalWrite(led42, HIGH);
delay(100);
digitalWrite(led42, LOW);
digitalWrite(led41, HIGH);
delay(100);
digitalWrite(led41, LOW);
digitalWrite(led40, HIGH);
delay(100);
digitalWrite(led40, LOW);
digitalWrite(led38, HIGH);
delay(100);
digitalWrite(led38, LOW);
delay(100);
//-----

// TEMP/WIND LIGHTS-----
digitalWrite(led49, HIGH);
digitalWrite(led44, HIGH);
delay(50);
digitalWrite(led49, LOW);
digitalWrite(led44, LOW);
digitalWrite(led50, HIGH);
digitalWrite(led45, HIGH);
delay(50);
digitalWrite(led50, LOW);
digitalWrite(led45, LOW);
digitalWrite(led51, HIGH);
digitalWrite(led46, HIGH);
delay(50);

digitalWrite(led51, LOW);
digitalWrite(led46, LOW);
digitalWrite(led52, HIGH);
digitalWrite(led47, HIGH);
delay(50);
```

```
digitalWrite(led52, LOW);
digitalWrite(led47, LOW);
digitalWrite(led53, HIGH);
digitalWrite(led48, HIGH);
delay(50);
digitalWrite(led53, LOW);
digitalWrite(led48, LOW);
delay(50);
//-----
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
delay(100);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led31, HIGH);
digitalWrite(led28, HIGH);
delay(100);
digitalWrite(led31, LOW);
digitalWrite(led28, LOW);
digitalWrite(led32, HIGH);
digitalWrite(led27, HIGH);
delay(100);
digitalWrite(led32, LOW);
digitalWrite(led27, LOW);
digitalWrite(led33, HIGH);
digitalWrite(led24, HIGH);
delay(100);
digitalWrite(led33, LOW);
digitalWrite(led24, LOW);
digitalWrite(led35, HIGH);
digitalWrite(led26, HIGH);
delay(100);
digitalWrite(led35, LOW);
digitalWrite(led26, LOW);
digitalWrite(led34, HIGH);
digitalWrite(led22, HIGH);
delay(100);
digitalWrite(led34, LOW);
digitalWrite(led22, LOW);
digitalWrite(led29, HIGH);
digitalWrite(led23, HIGH);
delay(100);
digitalWrite(led29, LOW);
digitalWrite(led23, LOW);
delay(150);
```

```
}
```

```
///-----SOS-----
```

if (ButtonStateSOS == LOW){

```
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(250);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(250);
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(250);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
```

```
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(250);
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(250);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(750);
```

```
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
```

```
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(750);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(250);
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(750);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(250);
```

```
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(750);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(750);
```

```
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(250);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
```

```
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(250);
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(250);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(250);
digitalWrite(led29, HIGH);
digitalWrite(led34, HIGH);
digitalWrite(led35, HIGH);
digitalWrite(led33, HIGH);
digitalWrite(led32, HIGH);
digitalWrite(led31, HIGH);
digitalWrite(led30, HIGH);
digitalWrite(led25, HIGH);
digitalWrite(led28, HIGH);
digitalWrite(led27, HIGH);
digitalWrite(led24, HIGH);
```

```
digitalWrite(led26, HIGH);
digitalWrite(led22, HIGH);
digitalWrite(led23, HIGH);
delay(250);
digitalWrite(led29, LOW);
digitalWrite(led34, LOW);
digitalWrite(led35, LOW);
digitalWrite(led33, LOW);
digitalWrite(led32, LOW);
digitalWrite(led31, LOW);
digitalWrite(led30, LOW);
digitalWrite(led25, LOW);
digitalWrite(led28, LOW);
digitalWrite(led27, LOW);
digitalWrite(led24, LOW);
digitalWrite(led26, LOW);
digitalWrite(led22, LOW);
digitalWrite(led23, LOW);
delay(1750);
}
```

```
//-----SOUND-----
```

```
-----
if (ButtonStateSound == LOW){
  for (int i=0; i<500; i++) { // generate a 1KHz tone for 1/2 second
    digitalWrite(SPKR, HIGH);
    delayMicroseconds(500);
    digitalWrite(SPKR, LOW);
    delayMicroseconds(500);
  }
}
else{
  digitalWrite(SPKR, LOW);
}
}
//-----END-OF-CODE-----HAVE A GREAT
DAY-----
```