Backup Sensor Project

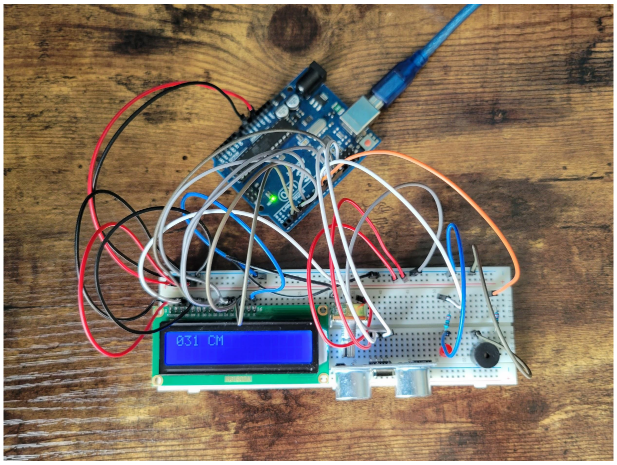
Objective:

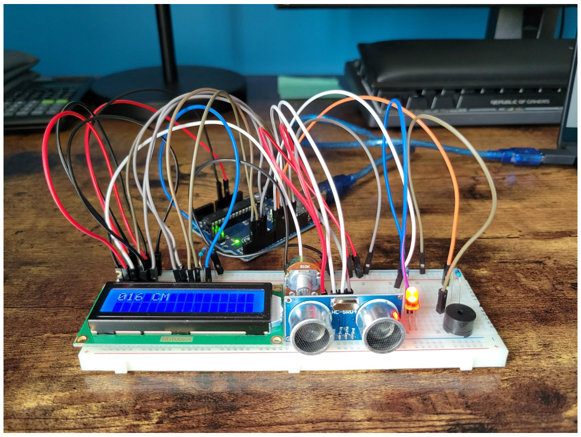
Build a circuit and program the circuit to display the distance between the circuit (sensor) and the incoming object. Additional features are a LED that will blink more frequently when closer and closer to the incoming object. Also a piezo speaker will beep more frequently as the object gets closer too.

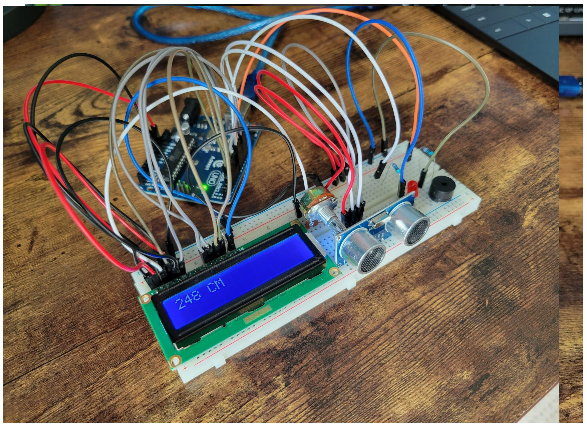
Connections:

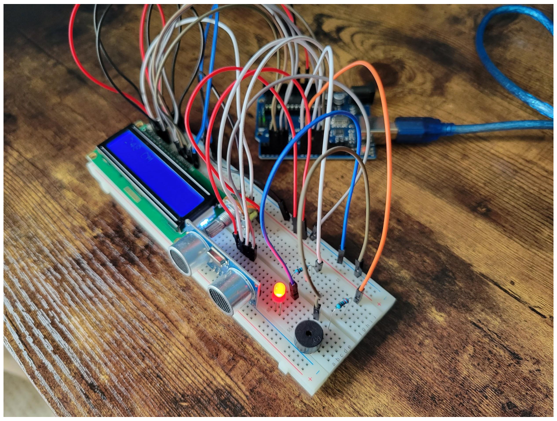
1. Set up VCC and GND to the breadboard.
2. LCD connections
   1. GND → VSS
   2. VCC → VDD
   3. POT WIPER → VO
   4. RS → PIN 12 (ARDUINO UNO)
   5. RW → GND
   6. E → PIN 11
   7. D4 → PIN 5
   8. D5 → PIN 4
   9. D6 → PIN 3
   10. D7 → PIN 2
   11. A → 220 OHM RESISTOR → VCC
   12. K → GND
3. POT connections
   1.WIPER as stated in Step 2
   2. Left to VCC, right to GND
4. Ultrasonic Distance Sensor connections
   1. VCC to VCC and GND to GND
   2. Trig(ger) → PIN 10
   3. Echo → PIN 13
5. Connect LED as usual (Use 220 ohm as it is the lowest resistor available)
6. Connect Piezo as usual (Use 220 ohm as it is the lowest resistor available)

Images





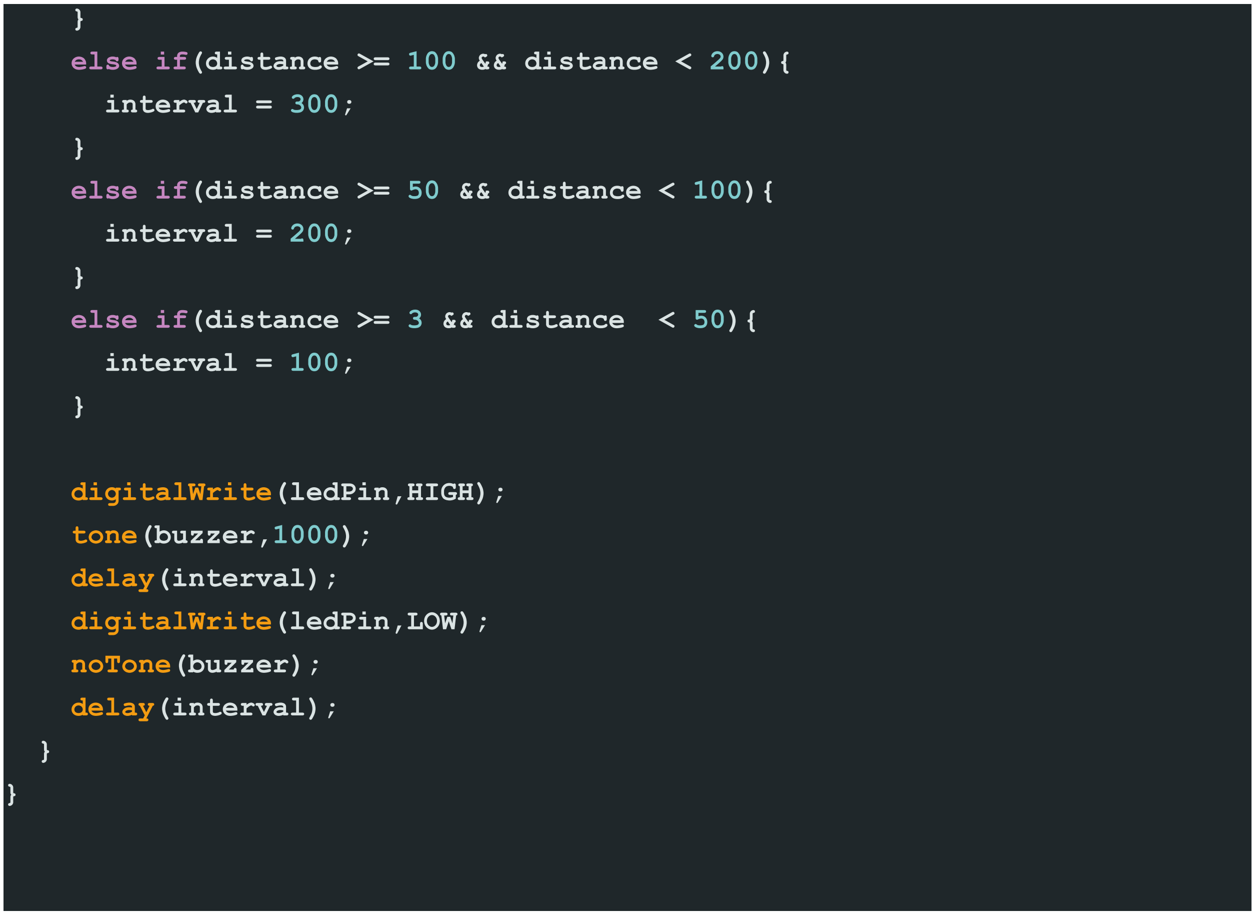




Code

```
//====================== Initialization Declarations
======================

//== Ultrasonic Distance Sensor ==
#define trigPin 10
#define echoPin 13
float duration; //Duration: duration of signal we get back
int distance;
unsigned long interval = 0;

//== LCD ==
#include <LiquidCrystal.h>
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

//== LED ==
int ledPin = 8;

//==Piezo==
const int buzzer = 9;

//== Other Included Libraries ==
#include <math.h>

//====================== Setup
============================================
void setup(){
  Serial.begin(9600);

  pinMode(ledPin,OUTPUT);
  pinMode(buzzer, OUTPUT);

  lcd.begin(16, 2);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin,INPUT);
}
```

```
// ============= Loop ====================
void loop(){
  digitalWrite(trigPin,LOW);
  delayMicroseconds(2);          //Sent for 2 microseconds. Doesn't matter
how long really
  digitalWrite(trigPin,HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin,LOW);

  duration = pulseIn(echoPin,HIGH);         // Search for the pulse sent
out
  distance = int((duration/2)*0.0343);      // Find distance with speed of
sound and this constant.

  if(distance >= 400 || distance <= 2){     // Distance sensor only works
for distances between 400cm and 2 cm
    lcd.clear();
    lcd.print("Out of range");
  }
  else{
    lcd.clear();

    if(distance < 10){
      lcd.print("00" + String(distance) + " CM");
    }
    else if (distance >= 10 && distance <= 99){
      lcd.print("0" + String(distance) + " CM");
    }
    else{
      lcd.print(String(distance) + " CM");
    }

    //=================

    if (distance >= 300 && distance < 400){
      interval = 500;
    }
    else if (distance >= 200 && distance < 300){
      interval = 400;
```



```
    }
    else if(distance >= 100 && distance < 200){
      interval = 300;
    }
    else if(distance >= 50 && distance < 100){
      interval = 200;
    }
    else if(distance >= 3 && distance  < 50){
      interval = 100;
    }

    digitalWrite(ledPin,HIGH);
    tone(buzzer,1000);
    delay(interval);
    digitalWrite(ledPin,LOW);
    noTone(buzzer);
    delay(interval);
  }
}
```