

CODE FOR SHOE SENSOR

```
// Pin definitions
const int trigPin = A0;
const int echoPin = A1;
const int buzzerPin = A2; // Make sure your board allows digital write on A2

long duration;
int distance;

void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(buzzerPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // Send pulse
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Read echo
  duration = pulseIn(echoPin, HIGH);

  // Convert to cm
  distance = duration * 0.034 / 2;
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  // Buzzer alert
  if(distance < 20){
    digitalWrite(buzzerPin, HIGH);
  } else {
    digitalWrite(buzzerPin, LOW);
  }

  delay(100);
}
```

EXPLANATION FOR CODE

Step 1: Pin Definitions

trigPin (A0): Sends a pulse to trigger the ultrasonic sensor.

echoPin (A1): Listens for the returning pulse after reflecting off an object.

buzzerPin (A2): Controls a buzzer (turns on/off depending on distance).

Step 2: setup() Function

Sets trigPin and buzzerPin as outputs, echoPin as input.

Initializes Serial communication for debug output.

Step 3: loop() Function

A. Sending the Ultrasonic Pulse

Set trigPin LOW then HIGH (for 10 microseconds), then LOW again.

This sequence sends out a short ultrasonic pulse from the sensor.

B. Reading the Echo

duration = pulseIn(echoPin, HIGH); waits for the echo pin to go HIGH, and measures how long it stays HIGH. This represents the time for the reflected pulse to return.

C. Calculating Distance

$\text{distance} = \text{duration} \times 0.0342$

The speed of sound is about 0.034 cm per microsecond. Dividing by 2 accounts for the pulse traveling to the object and back.

D. Buzzer Alert

If $\text{distance} < 20$, the buzzer is turned ON (`digitalWrite(buzzerPin, HIGH);`). Otherwise, the buzzer is OFF.

E. Serial Output and Loop Delay

The measured distance is printed to serial (could be viewed in the Serial Monitor).

Delay of 100 milliseconds before the next measurement.

Key Concepts Checked:

Pin modes: Why inputs and outputs matter; set-up in `setup()`.

Pulse timing: Measuring how long it takes for a sound pulse to return lets us calculate distance.

Condition check: Using `if` to turn the buzzer on if something is close.

Serial debugging: Helps you see the actual readings in real time.

SCREENSHOT OF THE STIMULATION

The screenshot displays the Tinkercad web interface for a project named "Fantastic Jofo". The circuit includes an Arduino Uno, an HC-SR04 ultrasonic sensor, and a buzzer. The sensor is connected to the Arduino's digital pins (trigPin and echoPin) and its VCC and GND pins. The buzzer is connected to the Arduino's digital pins (buzzerPin and GND). The sensor is emitting a green cone representing its range, with a label "7.7in / 19.4cm".

The code on the right is as follows:

```
1 // Constant pin definitions
2 const int trigPin = A1;
3 const int echoPin = A2; // Make sure your board allows digital pins that are
4 // configured as I/O: Arduino Uno and Pro Mini == A0-A5, Arduino Mega ==
5 // 2-5, Arduino DUE == 0-5
6 long duration;
7 int distance;
8
9 void setup() {
10   pinMode(trigPin, OUTPUT);
11   pinMode(echoPin, INPUT);
12   pinMode(buzzerPin, OUTPUT);
13   Serial.begin(9600);
14 }
15
16 void loop() {
17   // Send pulse
18   digitalWrite(trigPin, LOW);
19   delayMicroseconds(2);
20   digitalWrite(trigPin, HIGH);
21   delayMicroseconds(10);
22   digitalWrite(trigPin, LOW);
23
24   // Read echo
25   duration = pulseIn(echoPin, HIGH);
26
27   // Convert to cm
28   distance = duration * 0.034 / 2;
29   Serial.print("Distance: ");
30   Serial.print(distance);
31   Serial.println(" cm");
32
33   // Buzzer alert
34   if(distance < 20){
35     digitalWrite(buzzerPin, HIGH);
36   } else {
37     digitalWrite(buzzerPin, LOW);
38   }
39
40   delay(100);
41 }
```

The Serial Monitor shows the following output:

```
Distance: 19 cm
Distance: 19 cm
Distance: 19 cm
Distance: 19 cm
Distance: 19 cm
Distance: 19 cm
Distance: 19 cm
```