Ultrasonic Distance Measurement using Arduino

1. Introduction

This project utilizes an ultrasonic sensor to measure the distance of nearby objects using sound waves. The measured distance is displayed on the Serial Monitor in real-time, making it useful for basic proximity detection systems.

2. Key Components

- Arduino Uno (or any compatible board)
- HC-SR04 Ultrasonic Sensor
- Jumper wires
- Breadboard (optional)
- USB cable (for powering and uploading code to Arduino)

3. Working Principle

1. Trigger Signal Sent: The Arduino sets the TRIG pin HIGH for 10 microseconds to initiate a burst of sound waves from the sensor.

2. Echo Received: The sensor listens for the echo on the ECHO pin. The Arduino measures the time taken for the echo to return using the pulseIn() function.

3. Time to Distance Conversion: The time taken for the echo is multiplied by the speed of sound (0.034 cm/µs) and divided by 2 (since the signal travels to the object and back) to calculate the distance.

4. Displaying the Distance: The calculated distance is printed to the Serial Monitor, updating every loop cycle.

4. Circuit Overview

- TRIG pin of sensor -> Digital Pin 10 on Arduino
- ECHO pin of sensor -> Digital Pin 9 on Arduino
- VCC of sensor -> 5V on Arduino
- GND of sensor -> GND on Arduino

5. Code

```
/*Code Written by-
Fuad Hasan
BME, KUET
*/
```

```
//Ultrasonic Sensor

//Variable and pins declared and initialised.
int trigPin = 10; //TRIG connected to pin 10 of Arduino
int echoPin = 9; // ECHO connected to pin 9 of Arduino
long time;
int distance;

void setup()
  //this loop repeats only once
{
  pinMode(10, OUTPUT); //TRIG pin set as output
  pinMode(9, INPUT); //ECHO pin set as input
  Serial.begin(9600); //begin communication
}

void loop()
  //this loop repeats continuoously
{
  digitalWrite(10, LOW); //TRIG pin set low(cleared)
  delayMicroseconds(2); //delay of 2 microseconds given

  digitalWrite(10, HIGH); //TRIG pin set high (signal transmitted)
  delayMicroseconds(10); //delay of 10 microseconds given
  digitalWrite(10, LOW); //TRIG pin set as low again

  //calculateing the distance:
  time = pulseIn(9, HIGH); //to calculate time of flight
  distance = time*0.034/2; //to calculate distance of object

  //printing the distance:
  Serial.print("Distance: ");
  Serial.println(distance);
}
```

6. Code Explanation

Variable Declaration

- `trigPin` and `echoPin` hold the digital pin numbers.

- `time` stores the duration of the echo.

- `distance` stores the calculated distance.

Setup Function

- Configures pin 10 as output (TRIG) and pin 9 as input (ECHO).

- Begins serial communication at 9600 bps for debugging/output.

Loop Function

- Clears the TRIG pin to ensure a clean signal.

- Sends a 10-microsecond HIGH pulse to generate the ultrasonic wave.

- Waits for the ECHO pin to go HIGH and measures the time it stays HIGH (the round trip time of the sound wave).

- Converts the time into distance using the speed of sound in air (~0.034 cm/µs), and divides by 2 for the round trip.

- Outputs the calculated distance to the Serial Monitor.

7. Conclusion

This simple yet effective Arduino project demonstrates how ultrasonic sensors can be used for non-contact distance measurement. It serves as a foundational concept for obstacle avoidance, level sensing, and other automation systems.