INTERFACING ULTRASONIC WITH RPI

# Aim

Interfacing an Ultrasonic Sensor with an RPi to create a distance calculator that displays the distance from an object in centimeters.

# Material Required

\* Ultrasonic Sensor  
\* Wires  
\* Raspberry Pi 4

# Code Snippet

```python  
import RPi.GPIO as GPIO  
import time  
  
# Define GPIO pins for the ultrasonic sensor  
TRIG = 23  
ECHO = 24  
  
# Set up GPIO pins  
GPIO.setmode(GPIO.BCM)  
GPIO.setup(TRIG, GPIO.OUT)  
GPIO.setup(ECHO, GPIO.IN)  
  
def distance():  
 # Send ultrasonic pulse  
 GPIO.output(TRIG, True)  
 time.sleep(0.00001)  
 GPIO.output(TRIG, False)  
  
 # Record time of echo  
 pulse\_start = time.time()  
 while GPIO.input(ECHO) == 0:  
 pulse\_start = time.time()  
 while GPIO.input(ECHO) == 1:  
 pulse\_end = time.time()  
  
 # Calculate distance  
 pulse\_duration = pulse\_end - pulse\_start  
 distance = pulse\_duration \* 17150  
 distance = round(distance, 2)  
  
 return distance  
  
# Main loop  
try:  
 while True:  
 dist = distance()  
 print ("Measured Distance = %.1f cm" % dist)  
 time.sleep(0.1)  
  
except KeyboardInterrupt:  
 print("Measurement stopped by User")  
 GPIO.cleanup()  
```

# Working

\* The ultrasonic sensor emits a sound wave and measures the time it takes for the wave to return after bouncing off an object.  
\* The Raspberry Pi receives the echo signal from the sensor and calculates the distance based on the time of flight.   
\* The calculated distance is then displayed on the terminal or any other output device.  
\* The code repeatedly measures the distance and updates the output, providing real-time distance readings.

# Procedure

1. \*\*Connect the Ultrasonic Sensor to the Raspberry Pi:\*\*  
 \* Connect the VCC pin of the sensor to the 5V pin on the Raspberry Pi.  
 \* Connect the GND pin of the sensor to the GND pin on the Raspberry Pi.  
 \* Connect the TRIG pin of the sensor to a GPIO pin on the Raspberry Pi (e.g., GPIO 23).  
 \* Connect the ECHO pin of the sensor to another GPIO pin on the Raspberry Pi (e.g., GPIO 24).  
  
2. \*\*Install the Required Libraries:\*\*  
 \* Open a terminal on your Raspberry Pi and install the RPi.GPIO library using the command: `sudo apt-get install python3-rpi.gpio`  
  
3. \*\*Save the Code:\*\*  
 \* Create a new Python file (e.g., distance\_calculator.py) and copy the code provided into it.  
  
4. \*\*Run the Code:\*\*  
 \* Save the Python file and run it using the command: `python3 distance\_calculator.py`  
  
5. \*\*Test the Distance Measurement:\*\*  
 \* Point the ultrasonic sensor towards an object and observe the distance readings displayed on the terminal.  
 \* Move the object closer or farther to see the distance readings change.

# Result

The distance calculator successfully measured the distance to objects placed in front of the ultrasonic sensor. The readings were displayed in real-time on the terminal, accurately reflecting the distance changes as the objects were moved closer or farther away.

# Learning

Students learned about the principles of ultrasonic sensing, how to interface sensors with the Raspberry Pi, and how to program the Pi to process sensor data. They gained practical experience in setting up and configuring GPIO pins, working with libraries like RPi.GPIO, and developing Python code for data acquisition and processing.