[INTERNET OF THINGS]

Practical-10

-:AIM:-
Raspberry pi basic Programming with Ultrasonic Sensor.

Submitted By: Dharmay Sureja

Enrollment No:17012011056

GANPAT UNIVERSITY

U. V. Patel College of Engineering

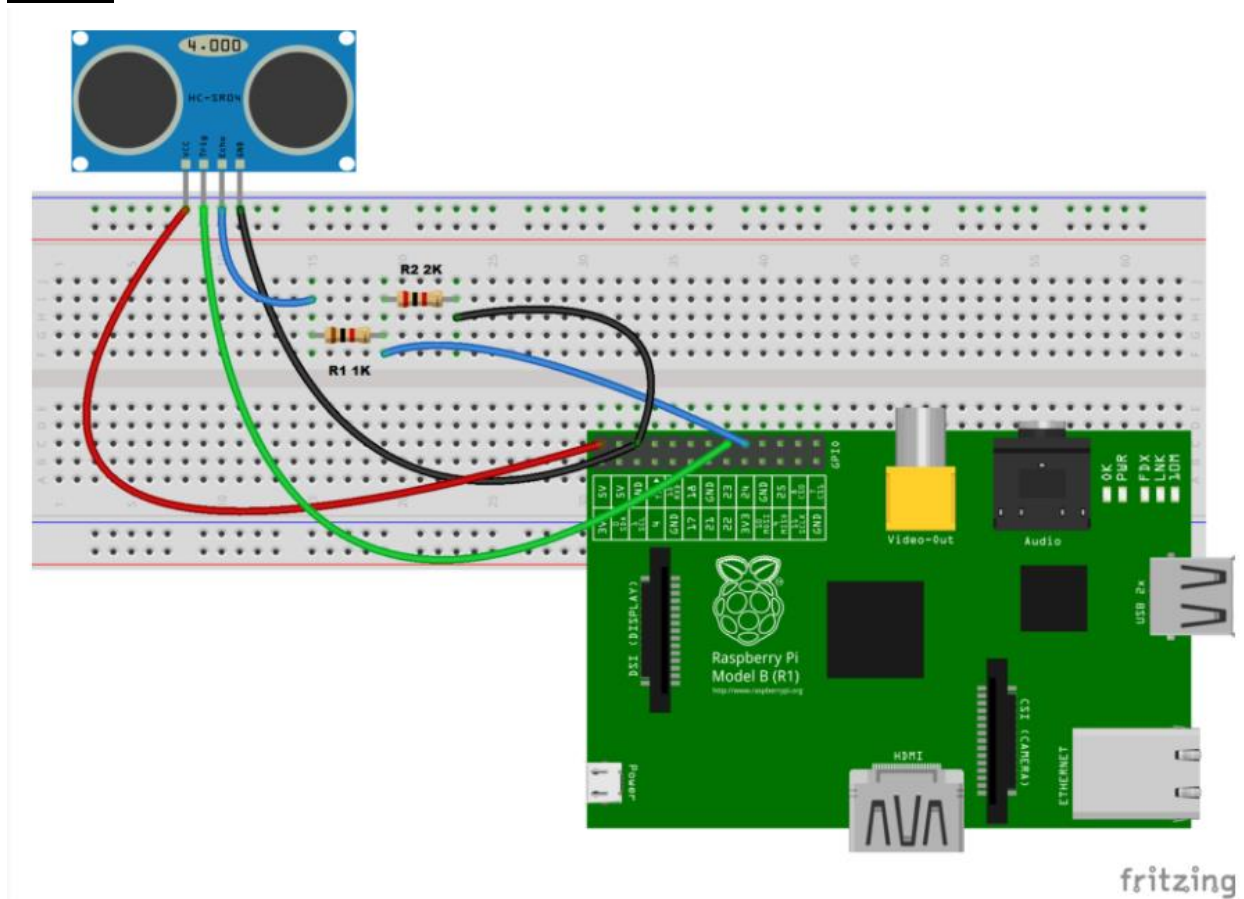**Computer Engineering Department**

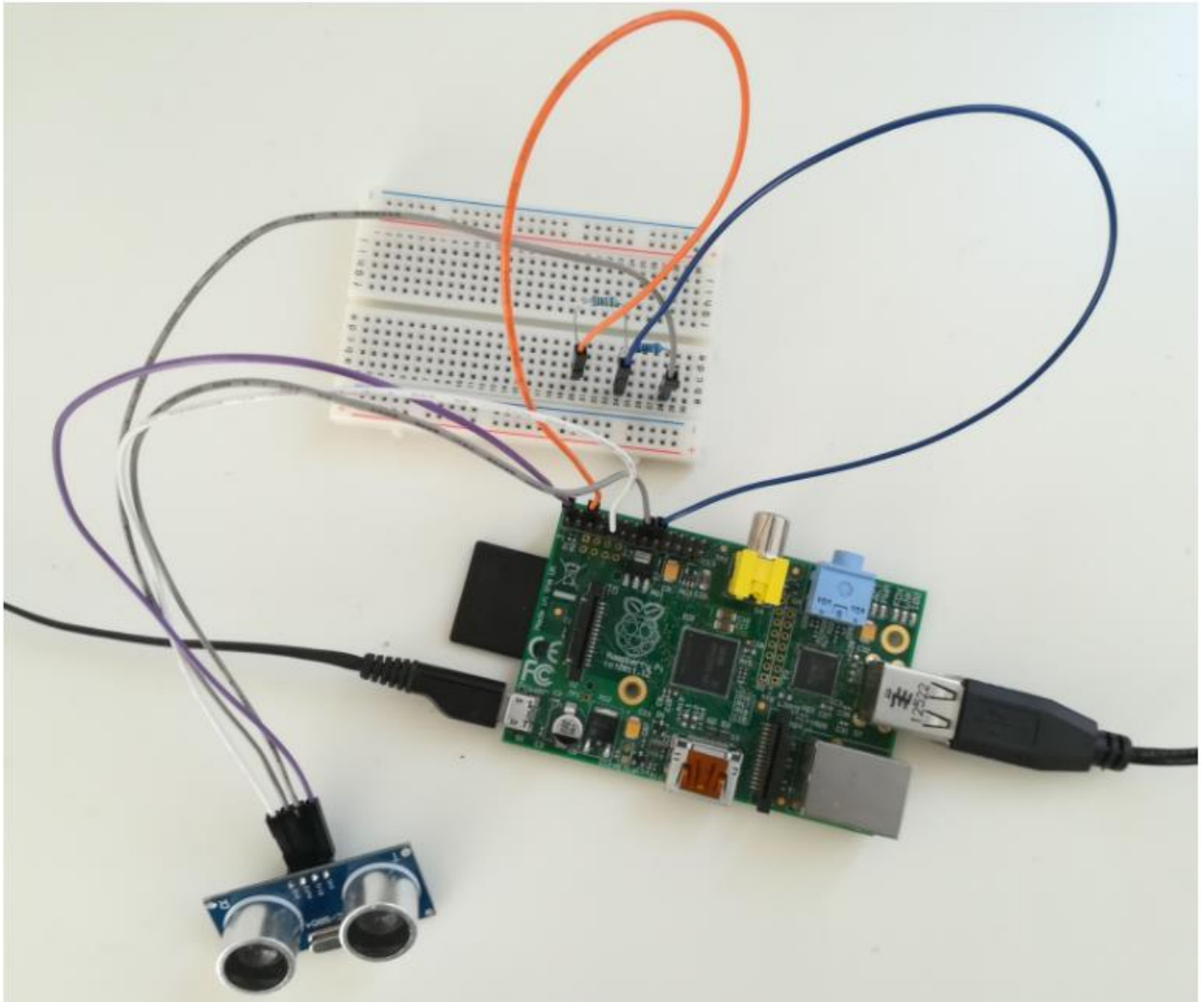# AIM:- Raspbian pi basic programming with Ultrasonic Sensors.

Experiments

1. **Measure the distance of an obstacle using ultrasonic sensor.**

Components used : Breadboard , 2.2K Ohm Resistor, 1K Ohm Resistor , Raspberry pi , SD Card & Adapter , Jumper Wire , Ultrasonic Sensor , USB Cable

Circuit:

```
~/raspberrypi$ ls
Adafruit_DHT.py  distance.py
~/raspberrypi$ python distance.py
Measured Distance = 302.8 cm
 Measured Distance = 303.1 cm
 Measured Distance = 303.5 cm
 Measured Distance = 303.2 cm
 Measured Distance = 300.9 cm
 Measured Distance = 304.0 cm
 Measured Distance = 302.1 cm
 Measured Distance = 301.9 cm
 Measured Distance = 301.4 cm
 Measured Distance = 303.3 cm
 Measured Distance = 301.9 cm
 Measured Distance = 303.0 cm
```

Code : `distance.py`

```python
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

GPIO_TRIGGER = 23
GPIO_ECHO = 24

GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
GPIO.setup(GPIO_ECHO, GPIO.IN)

def distance():
    GPIO.output(GPIO_TRIGGER, True)

    # set Trigger after 0.01ms to LOW
    time.sleep(0.00001)
    GPIO.output(GPIO_TRIGGER, False)

    StartTime = time.time()
    StopTime = time.time()

    while GPIO.input(GPIO_ECHO) == 0:
        StartTime = time.time()

    # save time of arrival
    while GPIO.input(GPIO_ECHO) == 1:
        StopTime = time.time()

    # time difference between start and arrival
    TimeElapsed = StopTime - StartTime
    # multiply with the sonic speed (34300 cm/s)
    # and divide by 2, because there and back
    distance = (TimeElapsed * 34300) / 2

    return distance

if __name__ == '__main__':
```

```
try:
    while True:
        dist = distance()
        print ("Measured Distance = %.1f cm" % dist)
        time.sleep(1)

    # Reset by pressing CTRL + C
except KeyboardInterrupt:
    print("Measurement stopped by User")
    GPIO.cleanup()
```

2. Send SMS if Object distance is less then 10 cm (using Twillo API)

Code : send_sms.py

```
import RPi.GPIO as GPIO
import time
from twillo.rest import Client

account_sid = "AC92b0aeee716d654f31561ac68bec0f12"

auth_token = "95aa6dee3decb6a6c9fb224e4d7c4916"

GPIO.setmode(GPIO.BCM)

GPIO_TRIGGER = 23
GPIO_ECHO = 24

GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
GPIO.setup(GPIO_ECHO, GPIO.IN)

def distance():
    GPIO.output(GPIO_TRIGGER, True)

    # set Trigger after 0.01ms to LOW
    time.sleep(0.00001)
    GPIO.output(GPIO_TRIGGER, False)

    StartTime = time.time()
    StopTime = time.time()
```

```python
    while GPIO.input(GPIO_ECHO) == 0:
        StartTime = time.time()

    # save time of arrival
    while GPIO.input(GPIO_ECHO) == 1:
        StopTime = time.time()

    # time difference between start and arrival
    TimeElapsed = StopTime - StartTime
    # multiply with the sonic speed (34300 cm/s)
    # and divide by 2, because there and back
    distance = (TimeElapsed * 34300) / 2

    return distance

if __name__ == '__main__':
    try:
        while True:
            dist = distance()
            print ("Measured Distance = %.1f cm" % dist)
            time.sleep(1)
            if dist < 10:
                client =  Client(account_sid,auth_token)
                message = client.message.create(
                to = "+917845612390"
                from = "575701"
                body = "Hello from GNU's Rasp_PI")
                print(message.sid)
                quit(0)

        # Reset by pressing CTRL + C
    except KeyboardInterrupt:
        print("Measurement stopped by User")
        GPIO.cleanup()
```
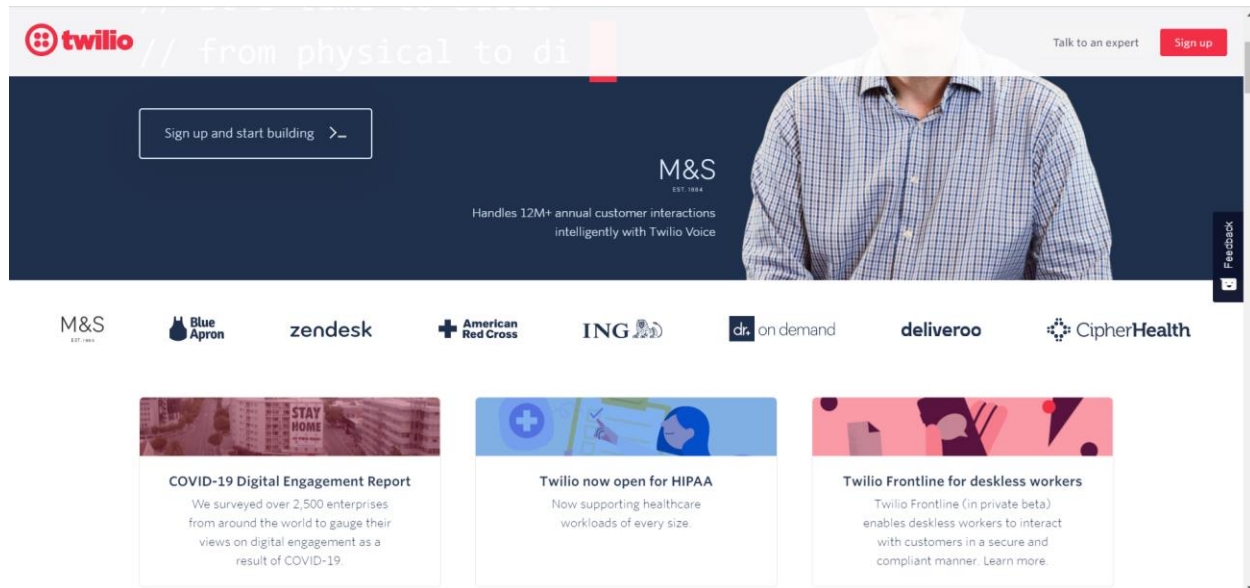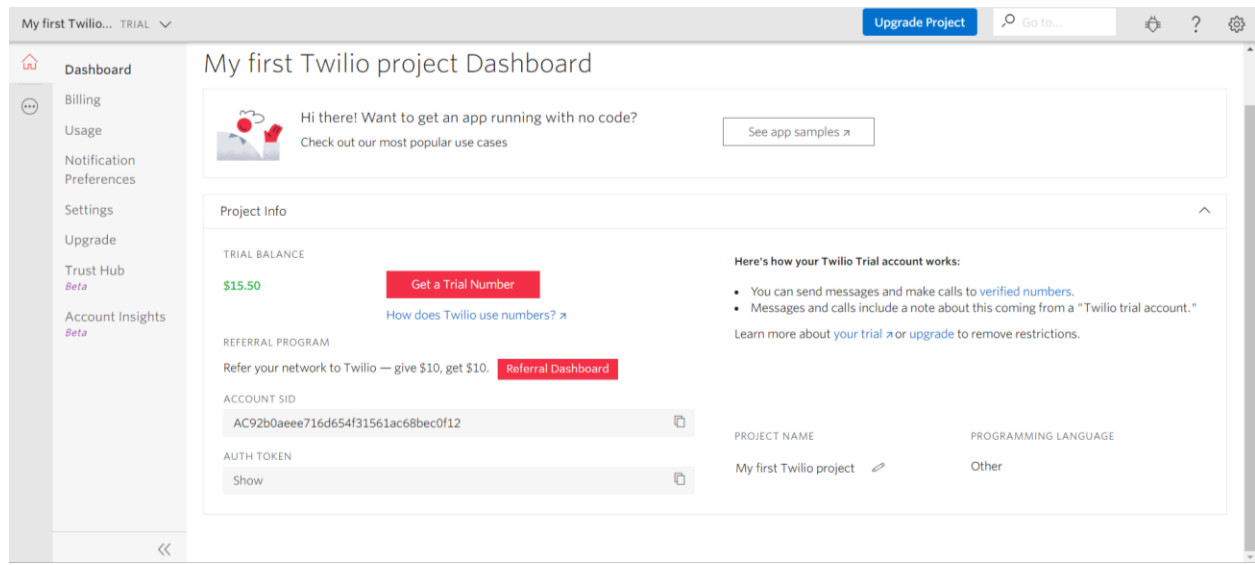
1. Open www.twillo.com

2. Register with your mail and mobile Number .

3. Login www.twillo.com with your email address

4. Note the Authentication ID and Authentication Token.

5. Get twillo phone number (In order to make calls or send Messages through thw twillo API , you need to get a Twillo Phone Number .