**Iot** **phase 3 project**



Problem statement : Start building the IoT sensor system and Raspberry Pi integration.

Configure IoT sensors (e.g., ultrasonic sensors) to detect parking space occupancy.

Write Python scripts on Raspberry Pi to collect data from sensors and send it to the cloud or mobile app server.

Submitted

By

N.Viswanath

S.Sanjai

A.Ragunath

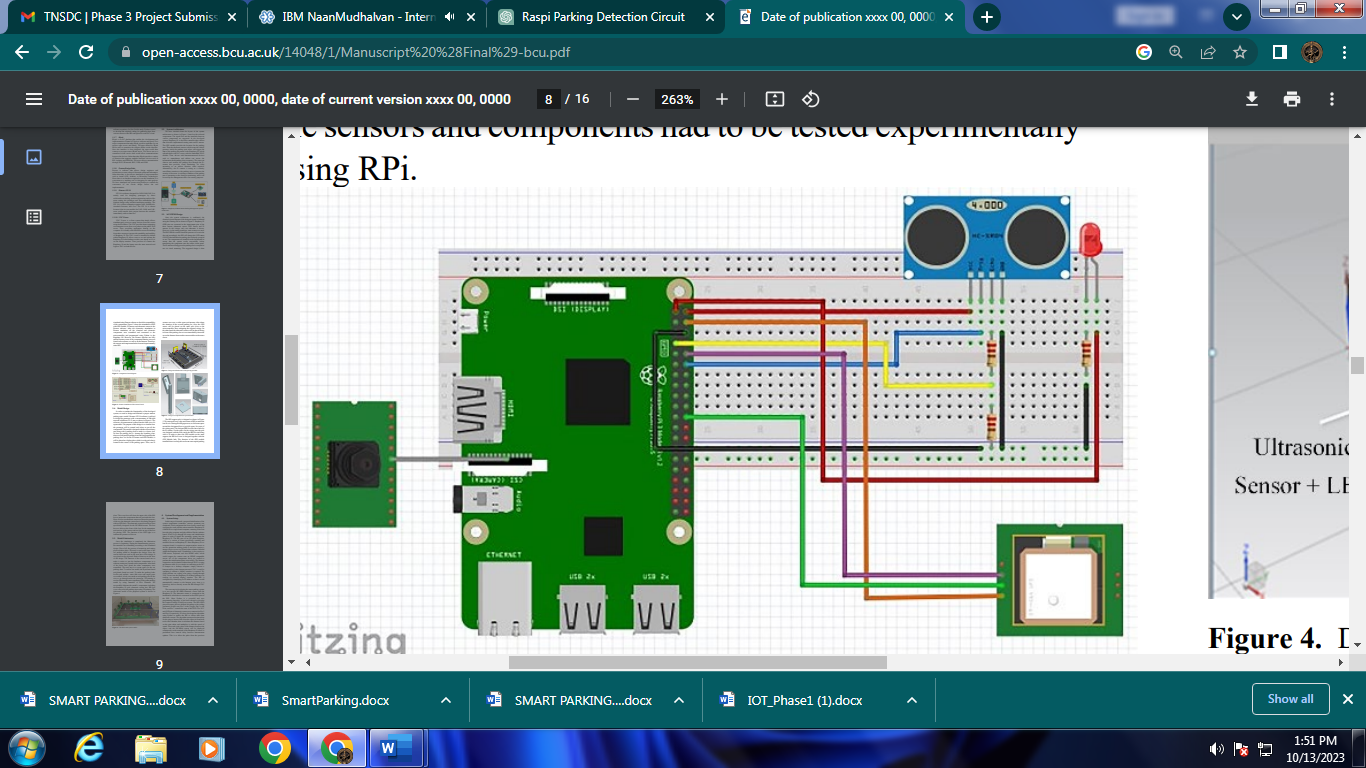
E.Periyadharshan

U.Karthikeyan

E.

**Pin connections:**

* **Ultrasonic Sensor**:
  + VCC (Power): Connect to 5V (Pin 2 or Pin 4) on the Raspberry Pi.
  + Trig (Trigger): Connect to a GPIO pin (e.g., GPIO17, Pin 11 on Raspberry Pi).
  + Echo: Connect to another GPIO pin (e.g., GPIO18, Pin 12 on Raspberry Pi).
  + GND (Ground): Connect to Ground (Pin 6, Pin 9, or Pin 14) on the Raspberry Pi.



Python script for parking detection using an ultrasonic sensor. This script measures the distance and detects if a car is in a parking space based on a distance threshold:

**import RPi.GPIO as GPIO**

**import time**

**# Set GPIO mode and pins**

**GPIO.setmode(GPIO.BCM)**

**TRIG\_PIN = 17**

**ECHO\_PIN = 18**

**GPIO.setup(TRIG\_PIN, GPIO.OUT)**

**GPIO.setup(ECHO\_PIN, GPIO.IN)**

**def distance():**

**GPIO.output(TRIG\_PIN, True)**

**time.sleep(0.00001)**

**GPIO.output(TRIG\_PIN, False)**

**while GPIO.input(ECHO\_PIN) == 0:**

**pulse\_start = time.time()**

**while GPIO.input(ECHO\_PIN) == 1:**

**pulse\_end = time.time()**

**pulse\_duration = pulse\_end - pulse\_start**

**return pulse\_duration \* 17150**

**try:**

**while True:**

**dist = distance()**

**print(f"Distance: {dist} cm")**

**# Adjust this threshold as needed for your parking space**

**threshold = 30 # Change this distance as needed**

**if dist < threshold:**

**print("Parking space occupied")**

**else:**

**print("Parking space available")**

**time.sleep(1)**

**except KeyboardInterrupt:**

**GPIO.cleanup()**

