Networks Project Report

By:

Yousef Dessouki	*7001816*	*T02*
Mostafa Moussa	*7001355*	*T02*
Zyad El Halawany	*7000626*	*T09*
AbdelRahman Mohab	*7003985*	*T04*
Abdelrahman Dabour	*7036950*	*T-4*
Mohamed Sameh	*7005440*	*T-9*
Mohamed Hatem	*7015916*	*T-10*
Amr Mansy	*7004871*	*T-10*

Why is the MQTT protocol important?

The MQTT protocol has become a standard for IoT data transmission because it delivers the following benefits:

Lightweight and efficient

MQTT implementation on the IoT device requires minimal resources, so it can even be used on small microcontrollers. For example, a minimal MQTT control message can be as little as two data bytes. MQTT message headers are also small so that you can optimize network bandwidth.

Scalable

MQTT implementation requires a minimal amount of code that consumes very little power in operations. The protocol also has built-in features to support communication with a large number of IoT devices. Hence, you can implement the MQTT protocol to connect with millions of these devices.

Reliable

Many IoT devices connect over unreliable cellular networks with low bandwidth and high latency. MQTT has built-in features that reduce the time the IoT device takes to reconnect with the cloud. It also defines three different quality-of-service levels to ensure reliability for IoT use cases— at most once (0), at least once (1), and exactly once (2).

Secure

MQTT makes it easy for developers to encrypt messages and authenticate devices and users using modern authentication protocols, such as OAuth, TLS1.3, Customer Managed Certificates, and more.

Well-supported

Several languages like Python have extensive support for MQTT protocol implementation. Hence, developers can quickly implement it with minimal coding in any type of application.

In order to measure distance in this project we will use an ultrasonic sensor.Ultrasonic sensors emit a chirp usually between 23 kHz and 40 kHz, much higher than the typical audible range of human hearing at 20 kHz, hence the term ultrasonic .According to the datasheet of the manufacturer, this module can measure distance ranging from 2 cm to 254 cm.We will use 2 sensors from this module.One for the rear and one for the front side.Primary objective is to measure the distance in front and backward of the sensor and display that to the user.

Ultrasonic sensors emit short, high-frequency sound pulses at regular intervals. These propagate in the air at the velocity of sound. If they strike an object, then they are reflected back as echo signals to the sensor, which itself computes the distance to the target based on the time-span between emitting the signal and receiving the echo . The principle of ultrasonic rangefinders is to measure the time it takes the signal sent by a transmitter and propagated back to the receiver . Ultrasonic sensors work by emitting sound waves at a frequency too high for humans to hear. They then wait for the sound to be reflected back, calculating distance based on the time required. This is similar to how radar measures the time it takes a radio wave to return after hitting an object .

Pin Name	**Description**
Vcc	The Vcc pin powers the sensor, typically with +5V
Trigger	Trigger pin is an Input pin. This pin has to be kept high for 10us to initialize measurement by sending US wave.
Echo	Echo pin is an Output pin. This pin goes high for a period of time which will be equal to the time taken for the US wave to return back to the sensor.
Ground	This pin is connected to the Ground of the system.

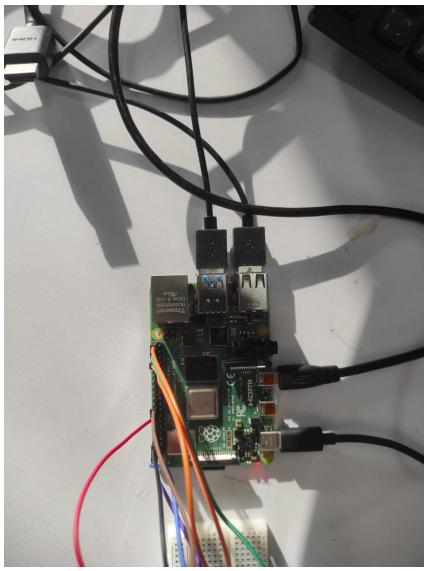
Methodology

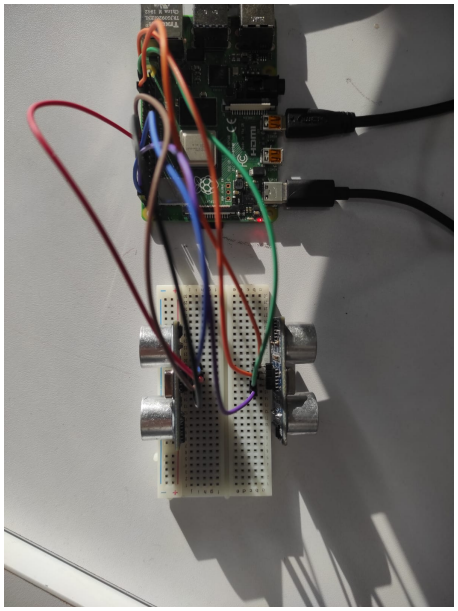
1. connect to node-red using the command node-red start or node-red –safe
2. Connect all the ultrasonic sensors to the raspberry pi using breadboard and female-female wires.
3. Write the function for the ultrasonic sensor.
4. Write the function for the javascript function that converts the string to int.
5. Program the MQTT broker to the public server.
6. Add the debug nodes.
7. Add execute nodes with the file path using sudo python filepath
8. Add the dashboard nodes.
9. Get the IP address of the raspberry pi using ipconfig.
10. On the separate computer write the ip address plus the port number 1880 .

Bills of materials

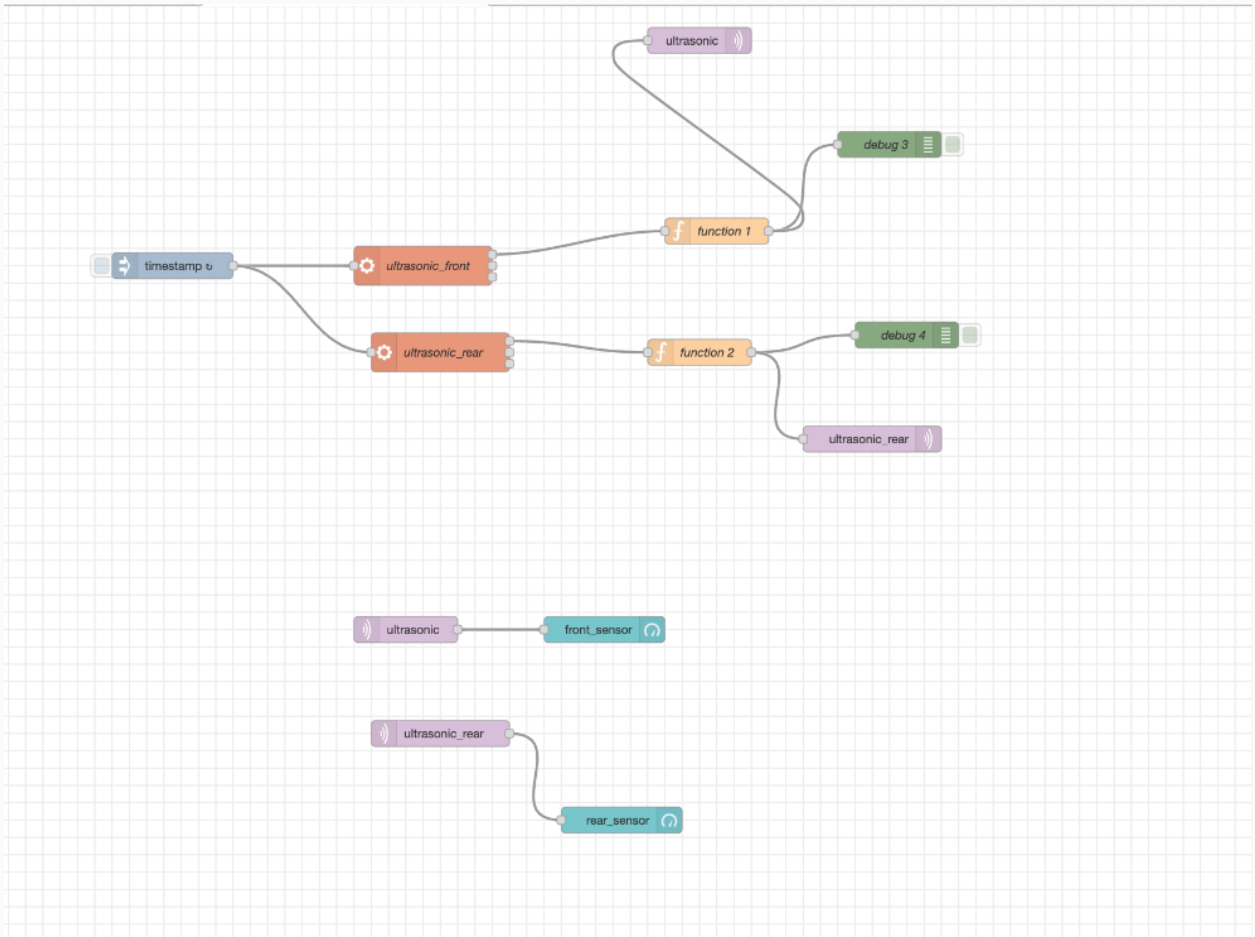
Equipment	quantity	price
Ultrasonic sensor	2	200
Jumper wires	20	40
SD card	1	300
Wires	1	400
Breadboard	1	30

Connection





Node-red


ultrasonic
debug 3
function 1
timestamp
ultrasonic_front
ultrasonic_rear
function 2
debug 4
ultrasonic_rear
ultrasonic
front_sensor
ultrasonic_rear
rear_sensor

Python function for the ultrasonic sensor:

```
import RPi.GPIO as GPIO

import time

GPIO.setmode(GPIO.BCM)

TRIG = 23

ECHO = 24

GPIO.setup(TRIG,GPIO.OUT)

GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG, False)

time.sleep(2)

GPIO.output(TRIG, True)

time.sleep(0.00001)

GPIO.output(TRIG, False)

while GPIO.input(ECHO)==0:

  pulse_start = time.time()

while GPIO.input(ECHO)==1:

  pulse_end = time.time()
```

```
pulse_duration = pulse_end - pulse_start

distance = pulse_duration * 17150

distance = round(distance, 1)

print (distance)

GPIO.cleanup()
```

Javascript function

```
msg.payload = Number(msg.payload); // Convert the payload to a number
if(msg.payload>100) // Check if the payload is greater than 100
{
   msg.payload=100; // If the payload is greater than 100, set it to 100
   return msg; // Return the message
}
else
{
   return msg; // If the payload is less than or equal to 100, return the
message as is
}
```