AN-020 IoT Radar Sensor with WiFi Interface

The recently released OPS243 radar sensor with WiFi/Bluetooth wireless connectivity is a complete IoT (Internet of Things) solution for traffic monitoring. All that is needed to access remote data is power for the sensor and a WiFi network to transport the data to the cloud for visualization, processing, or storage. The sensor is pre-configured to seamlessly link into a WiFi network and start sending data. This application note describes the use of the OPS243 wireless interface and steps to create a complete IoT traffic monitor solution.

Network Architecture

The WiFi/Bluetooth interface on the OPS243 was designed to allow simple connection to a WiFi network and start viewing the sensor data. To accomplish this, a new OmniPreSense Radar Sensor app for smartphones and tablets is available on the Google Playstore here. The Android app (iOS to follow) is utilized to make an initial connection to the OPS243 and configure it for the WiFi network of choice.

The overall network architecture is configured as shown in Figure 1. The OPS243 uses the local WiFi network to pass data to an MQTT (Message Queuing Telemetry Transport) Broker in the cloud. The MQTT Broker has been pre-configured by OmniPreSense to receive data from the sensor (some limits to data traffic may apply). Data from the sensor is published to the MQTT Broker as it becomes available in a manner similar to the sensor connected to a PC or embedded processor board over USB or UART. The same API format is used for all data communicated.

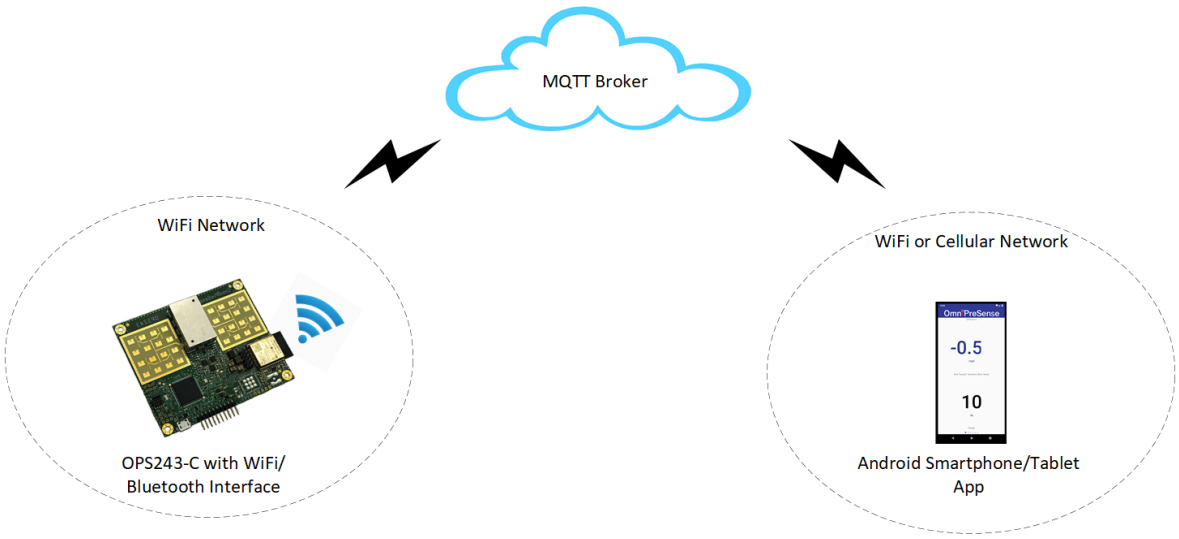

Figure 1. OPS243 Wireless Network Architecture

The app configures itself to be a subscriber to the data published by the sensor/MQTT Broker. This allows it to receive all the data with minimal delay for visualization. In a similar manner, API commands can be sent back to the sensor via the same publish/subscribe setup via the MQTT Broker.

WiFi Network Connection

The steps to connect the OPS243 into the WiFi network are simple and straight forward. The key steps are:

- Open WiFi Radar Sensor app
- Connect smartphone/tablet with OPS243 over Bluetooth
- Scan and choose WiFi network
    - Provide password if protected
- Receive data

The Bluetooth connection is only used for the initial connection and WiFi provisioning. Once connected into the WiFi network, all data is sent over the WiFi network from the sensor. The user of the app does not have to be within Bluetooth range of the sensor but can be located anywhere with a network connection (cellular, WiFi) and still receive sensor data.

The following describes the connection sequence for the Radar Sensor app and the OPS243 WiFi enabled sensor.

1. First, make sure the OPS243 is powered up via USB or other connection.
2. Upon opening the app the first time, several introductory screens will provide an overview of the apps features. At the end of these the app will provide the 5th screen for network connectivity (Figure *2*).

Figure 2. WiFi Radar Sensor App Bluetooth Pairing

3. The user should press the Scan Bluetooth button to scan for available Bluetooth devices. The radar sensor will be labeled OPS_RADAR_xxxx. Select the OPS243 sensor the Bluetooth connection will be completed.
4. The next step is to select and connect to the WiFi network. Press the Scan WiFi button and a list of available WiFi networks within range of the radar sensor will appear.
5. Select the desired WiFi network and if required, enter the password.

At the end of step 5, the OPS243 will be connected to the WiFi network of choice and start streaming data to the cloud. The data can be visualized on either the 1st or 2nd screen of the app (Figure 3).

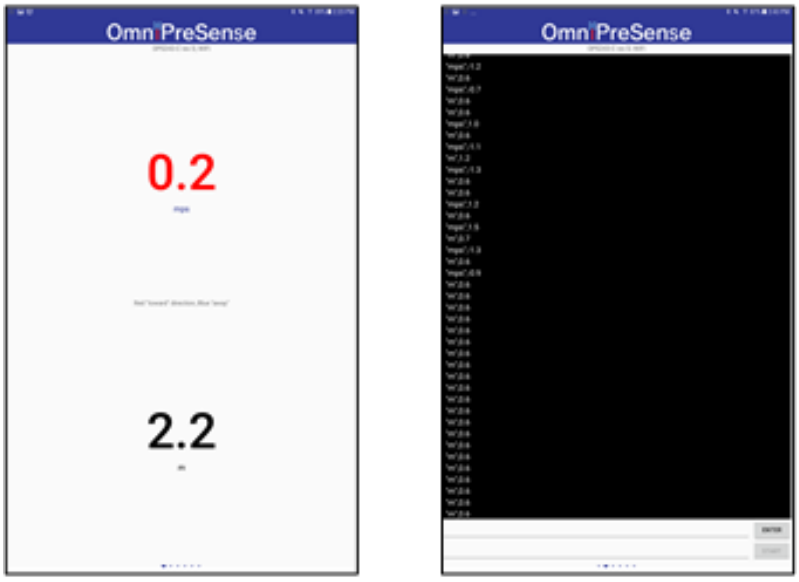

Figure 3. WiFi Radar Sensor App Data Visualization

The 1st screen provides the strongest signal speed and range information while the 2nd screen provides more detailed flow of data. The 2nd screen mimics a command terminal program like Putty and Teraterm and allows for entry of API commands at the bottom. Entering a command will be transferred back to the sensor which will adjust its output or processing accordingly.

Revision History

Version	Date	Description
A	February 2, 2021	Initial release.