Design Document - IoT Project

Group:
Aron Wussler
Luca Moroldo
Luca Tasso

Introduction

In this document we are going to present the implementation of a system that monitors the temperature, air quality and pressure of one or more rooms in a domestic environment and displays the data on a dashboard.

Design overview

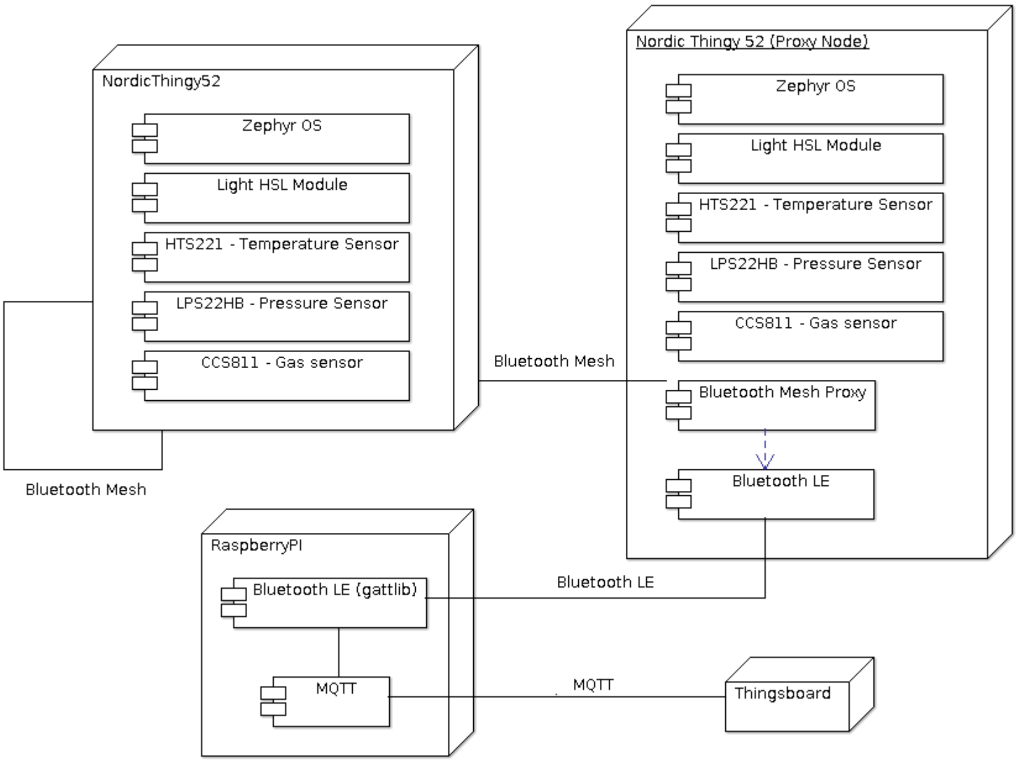
Data acquisition is performed by a network of small devices called **Nordic Thingy 52 (NT52)**. They are equipped with different sensors and actuators, but for the sake of our system we require only three of them: the temperature and humidity sensor (HTS221), the pressure sensor (LPS22HB) and the gas sensor (CCS811).
Every NT52:

- periodically polls its sensors and communicates their data through a Bluetooth Mesh network;
- receives and relays the messages of its peers in the BluetoothMesh Network;
- is identified by a unique address;
- has a group address that associates the device to a particular room of the environment.

One among the NT52 devices also acts as a **proxy node** that shares the messages of the mesh network with a **Raspberry Pi** through a Bluetooth Low Energy (LE) communication. The content of the messages is extracted and published through the MQTT protocol as topics.

An instance of **Thingsboard** running on a remote host subscribes to the topics offered by the Raspberry PI and retrieves the sensor data with MQTT. Thingsboard then collects this data and displays it on a dashboard, organizing it in graphs that shows how temperature, humidity and pressure changes over time.

Hardware deployment and communication



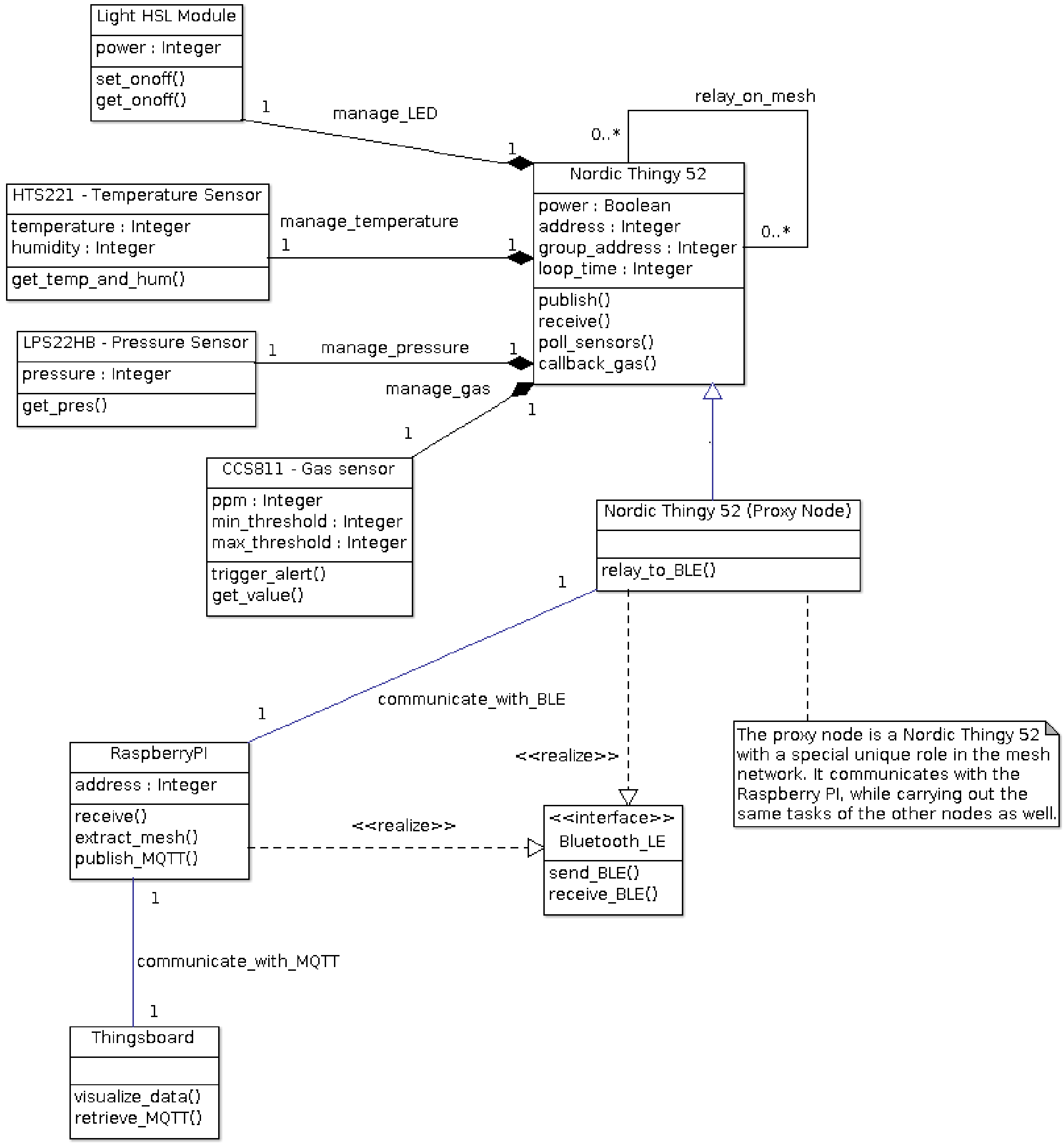
The image depicts deployment and component UML diagram of the system.

The Nordic Thingy 52 devices run Zephyr OS, which manages the sensors and the Bluetooth Mesh communication. The message exchanges are encrypted with a network key, shared to all the devices by a provisioning procedure.

The proxy node is the same as the other NT52, but it also enables a service that encapsulates the mesh traffic with the Bluetooth LE protocol headers and relays them to the Raspberry Pi, which interprets them using the gattlib library. The communication between the proxy node and Raspberry Pi is also encrypted.

Thingsboard is installed on a remote cloud instance and retrieves the data from the mesh by subscribing to the topics offered with the MQTT protocol by the Raspberry Pi. The connection established through MQTT is protected and encrypted by a password.

Components functionality overview



The picture depicts the class UML diagram of the system. It summarizes the functionalities that every component must implement.

- Light HSL module
  - turn on/off LED light
  - return status of the LED when asked
- Sensors
  - communicate temperature/humidity/air quality when requested by the nordic thingy 52

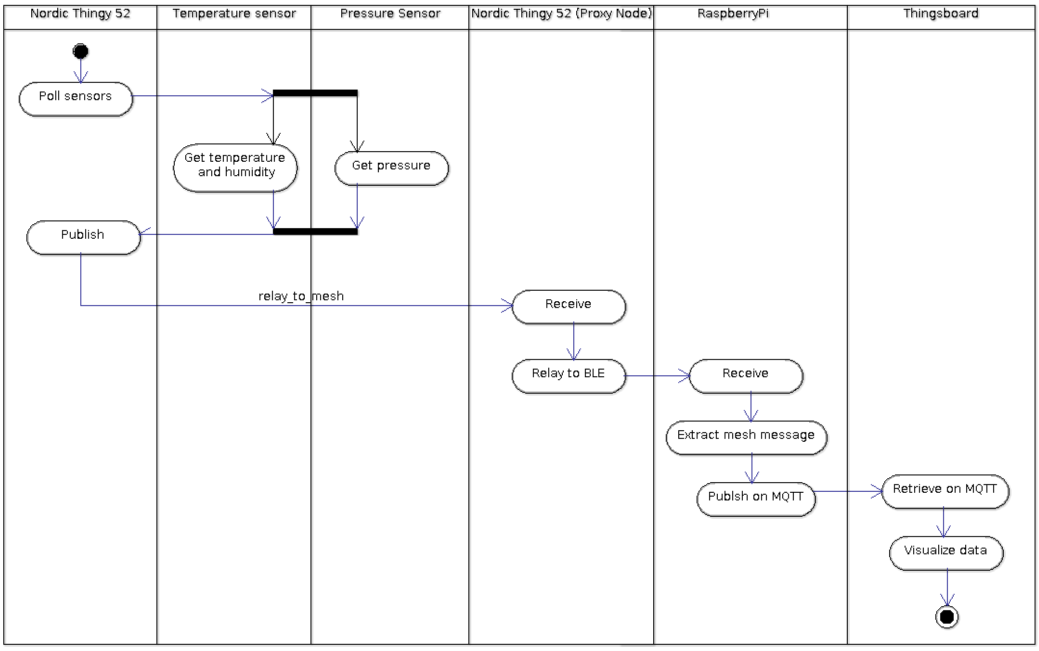
    - alert when gas indicator exceed certain thresholds
- Nordic Thingy 52
    - retrieve and publish data from its own sensors at regular intervals (loop_time)
    - send instructions to its sensors
    - receive and relay messages from other nodes
    - (Proxy node) relay mesh messages through bluetooth le to the raspberry pi
- Raspberry Pi
    - receive message from mesh
    - publish mesh data through MQTT
- Thingsboard
    - visualize data
    - retrieve updated data through MQTT

Sensors data acquisition

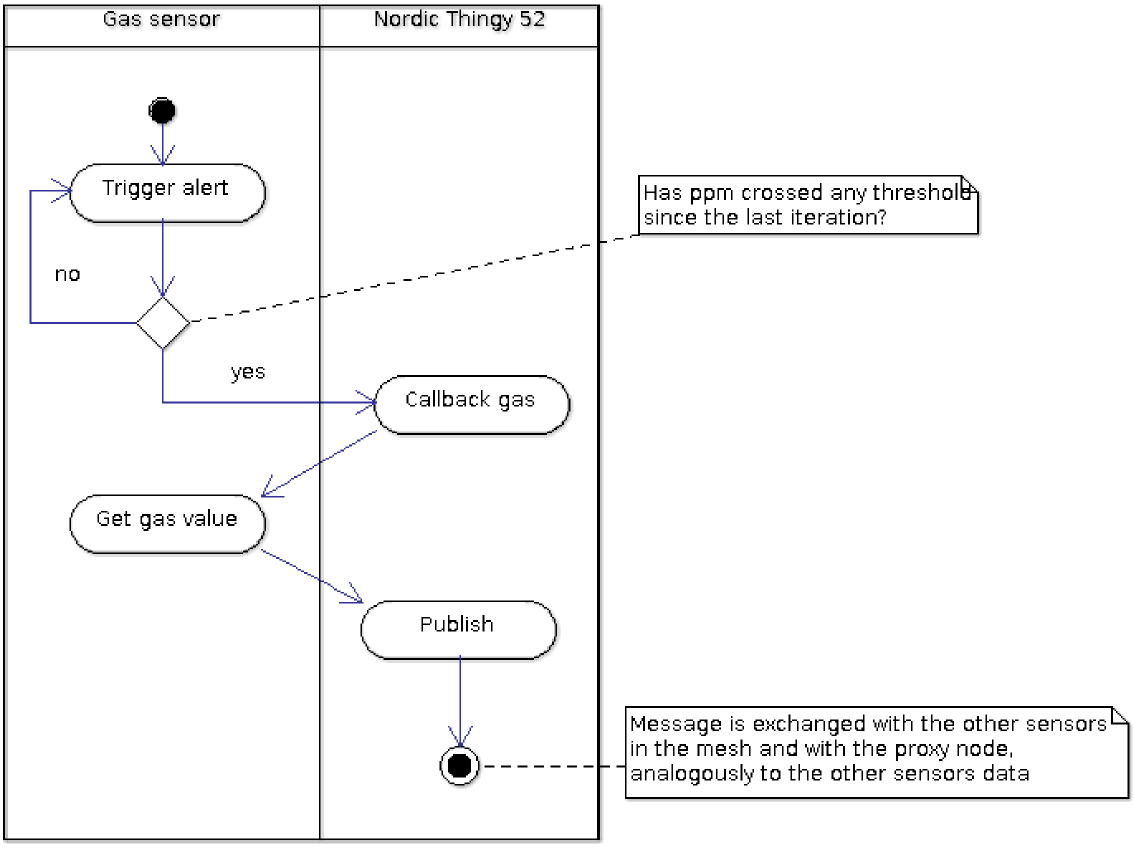
The first image shows the sequence of function calls when the NT52 polls its sensor and what each component does to relay the information up until it is displayed on Thingsboard.

It is worth noting that, after publishing on the mesh network, the NT52 waits for a certain amount of time (loop_time) before polling its sensors again.

Reception of any message immediately triggers a relay.



The second picture shows how data acquisition for the gas sensor differs from that of the other sensors. Instead of being polled on a regular basis, it sends an alert to the NT52 when the gas level exceeds a predefined minimum or maximum threshold. The NT52 requests directly to the sensor the gas level, which is returned and then published like all the other sensors data.


Gas sensor
Nordic Thingy 52
Trigger alert
no
yes
Callback gas
Get gas value
Publish
Has ppm crossed any threshold since the last iteration?
Message is exchanged with the other sensors in the mesh and with the proxy node, analogously to the other sensors data

User Interface

No user interaction beyond the visualization of the sensor data on Thingsboard’s dashboard is planned at this point.