EEET2370: Wireless Sensor Networks

Project Report

Dale Giancono: S3422566, Chris Benhard Armanda: S3626565

School of Engineering
RMIT University

November 6, 2018

Abstract

This report details the design and implementation of a Wireless Sensor Network. The network consists of 3 ZigBee devices programmed to collect meteorological data. The purpose of the project is to explore how ZigBee protocol can be applied as a network of sensors to collect data. In the scenario presented in this report, a simple ZigBee-compliant network is designed to simulate a weather station, where sensor nodes and clusters are to collect temperature data and send them as packets to a central Coordinator device. ZigBee's flaws were also discussed, and how the disadvantage were shown during testing of the network.

1 Application Description and Analysis

1.1 Problem Description

Figure 1: Sarajevo's Pollution Problem[1]

Sarajevo, the capital of Bosnia and Herzegovina, suffers high levels of air pollution in the winter months. These levels of pollution have been partly attributed to the phenomenon of temperature inversion. During extended periods of high pressure in winter months, solar radiation reaches the ground, warming it up. At night, the lack of cloud cover means the ground loses heat rapidly and the air in contact with the ground becomes colder. The warmer air rises and acts as a lid trapping the colder air close to the ground. Pollution, including that from heating systems, road traffic, and industry is also trapped. The area closest to the ground becomes more and more polluted. This continues until the prevailing meteorological conditions change.[2]

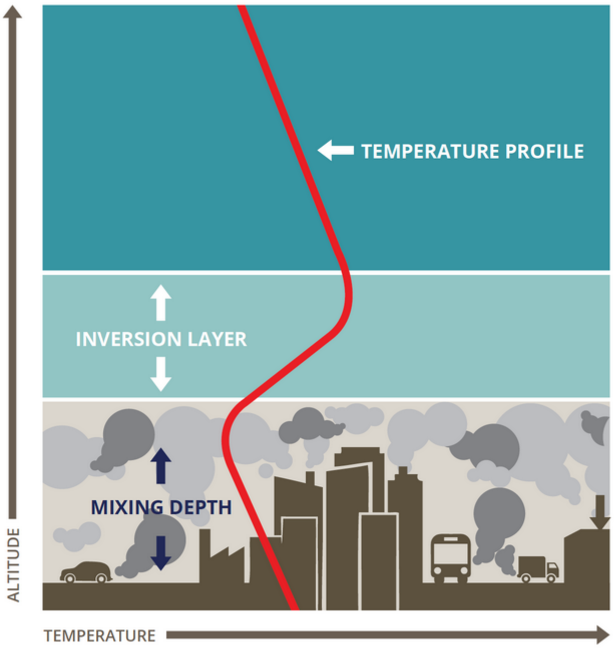

Figure 2: Temperature Inversion

The air flow impeding valley like topography of Sarajevo combined with the typical climate of the region promotes temperature inversion to occur regularly in Sarajevo during the winter months. The heating systems utilised Sarajevo at the time used fuels known for causing heavy pollution such as wood, lignite coal (brown coal), coke, and fuel oil. This air pollution in the atmosphere would be exacerbated by instances of temperature inversion, ensuring the pollution would linger within the city for extended periods of time.

1.2 Solution Description

A temperature inversion early warning system is proposed as a potential solution to Sarajevo's air pollution problems. By creating a meshed network of miniature weather stations, it is possible to measure meteorological parameters such as temperature, humidity, and wind speed at low altitude positions within the valley like topography of Sarajevo and high altitude positions that area above the city. By correlating this data, it should be relatively easy to calculate weather a temperature inversion is about to occur, and take measures to reduce pollution output within the city. These measures may include the reduction of traffic, the banning of wood and coal burning fires, and restrictions of industry located within the city.

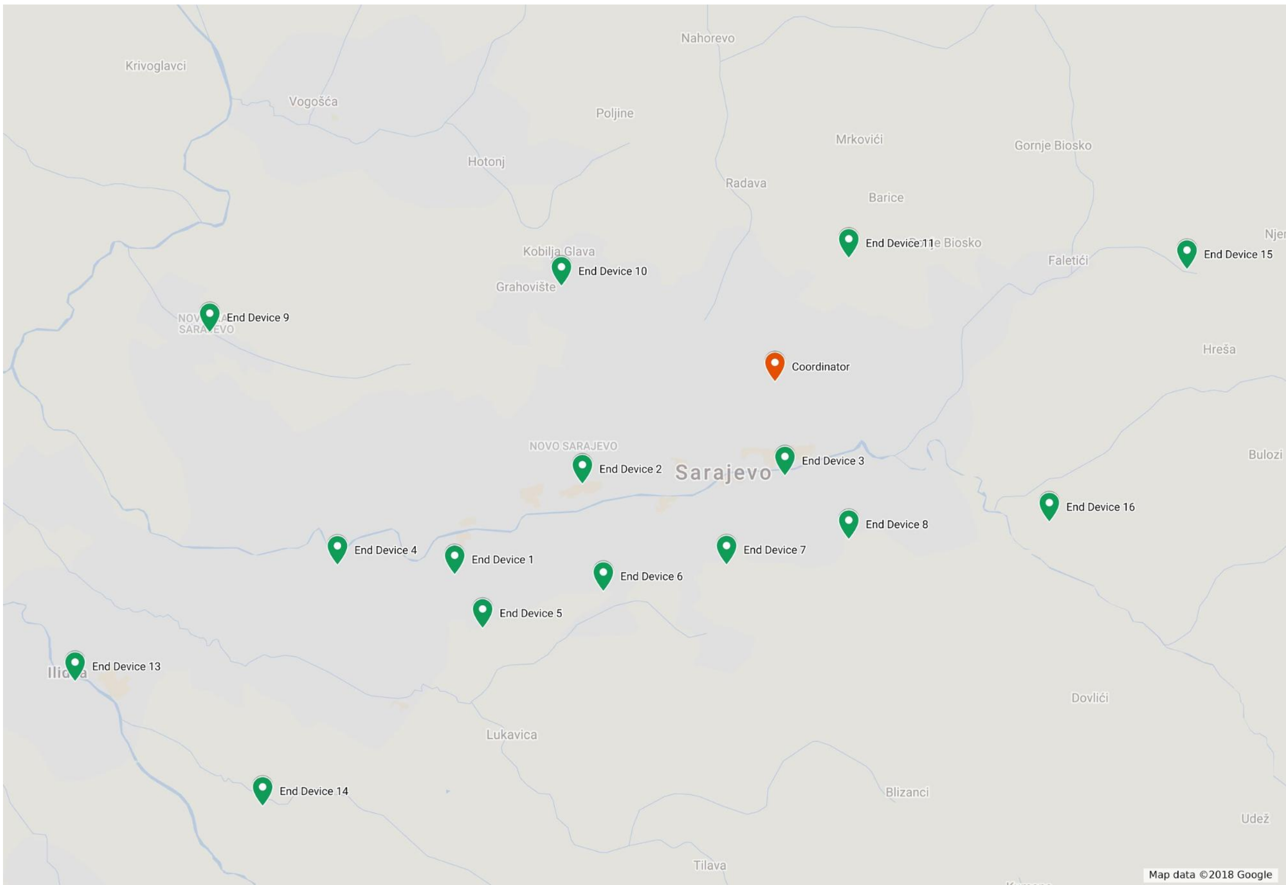

Figure 3: Proposed Weather Station Locations in Sarajevo

The proposed design aims to be autonomous, utilising no existing infrastructure due to its unreliable or non-existent nature within the areas of interest in and around Sarajevo. Each end device and router will come with a small battery and solar panel. The weather stations would behave as end devices, transmitting packets of data through routers back to a central coordinator. This coordinator could then transmit the data to a cloud based system using a gateway. Due to the large distances involved between each weather station end device, many router nodes will be placed around Sarajevo to allow the transmission of data from the end devices to the coordinator.

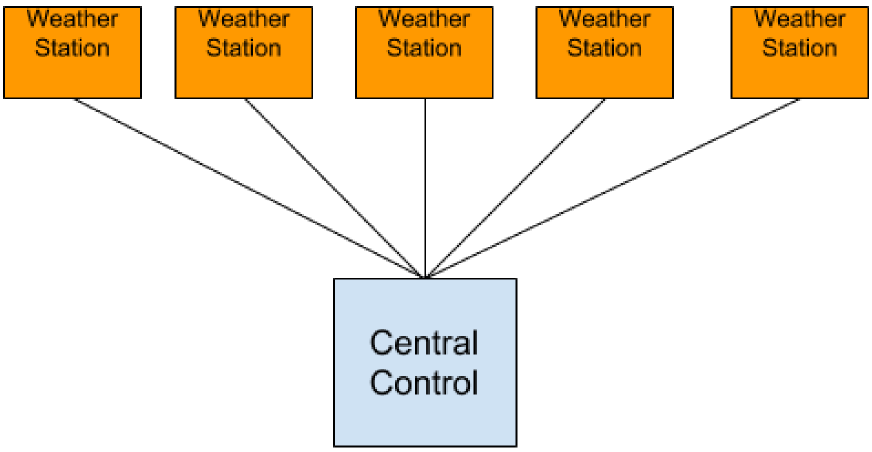

Figure 4: Simple topology of Weather Stations

2 WSN Design and Implementation

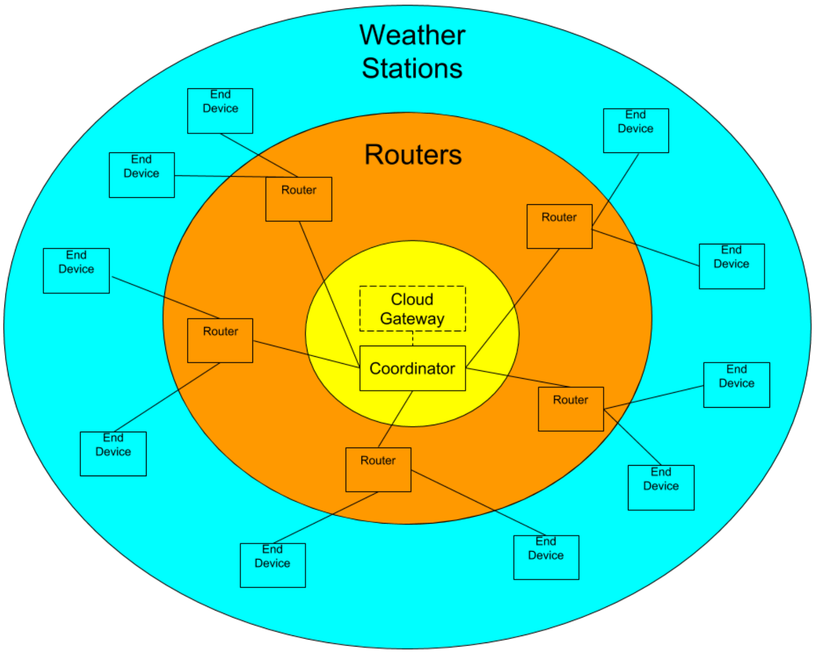

Figure 5: ZigBee network topology

2.1 ZigBee Technical Implementation

Three Texas Instruments SmartRF boards were utilised to create a ZigBee-compliant wireless sensor network (WSN). In this project, two SmartRF05 Evaluation Board and one SmartRF05 Battery Board, each attached with an antenna Evaluation Module are programmed to create a network of weather station, regularly collecting data of temperatures in the surroundings where each sensor is located. The boards are equipped with Texas Instruments CC2530 microcontroller, a modified 8051 microcontroller designed for ZigBee and IEEE 802.15.4 standards application.

The Evaluation Boards are equipped with Analog-to-Digital Converter (ADC) potentiometer. The potentiometer is used to simulate the fluctuation of temperature. The boards are programmed using IAR Embedded Workbench 3.0 and was based on the latest Zstack v3.0. Zstack provides the ZigBee library files to be uploaded along with the written program to the three boards.

As an exercise to utilise and learn how the three roles of a ZigBee device function, the two Evaluation Boards were programmed to be Coordinator and End Device, whereas the Battery Board played the Router role. Since the Evaluation Board has more input/output peripherals attached to it than the Battery Board, more sensors can theoretically be implemented on the former. The Evaluation Board also have a Serial port which can be connected to a PC.

However, Routers also plays an important part should the network grows when more devices associated itself into the network. ZigBee devices have a typical range of 15 to 20 meters to transmit packet to one another. Having a router in the network can therefore extend the distance and physical size covered by the network.

2.2 Hardware

The WSN was designed to follow a meshed topology, consisting of three ZigBee devices. The three devices in the topology act as a Coordinator, Router, and End Device. The full topology of the WSN is shown in Figure 6.

In practice, it is possible that the Router is not activated in the WSN. Therefore, all packets generated and sent by the End Device will be received directly by the Coordinator, without the need to pass through the Router first. In the implementation of this network, only the End Devices would send its recorded meteorological data

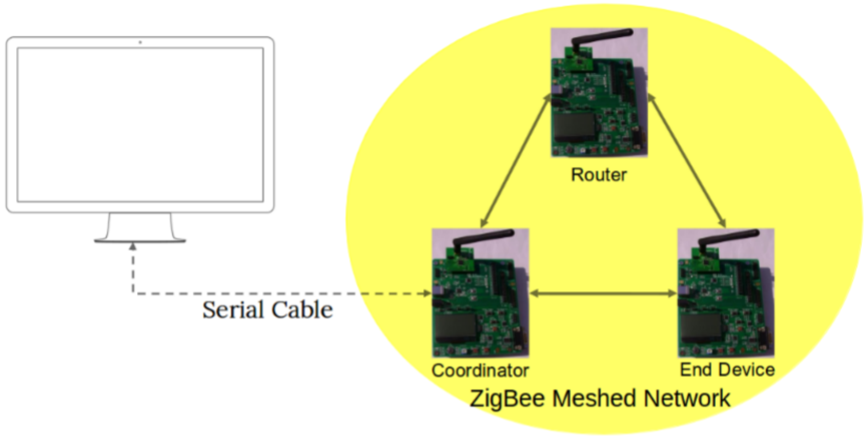

Figure 6: Topology of the Network for the project

to the Coordinator. Since ZigBee networks adhere to ALOHA protocol where packets are completely dropped when colliding with other packets in the same radio channel, this design will reduce occurrence of collisions.

The Coordinator is connected to a computer by using a Serial cable. The Coordinator's serial port will forward all packets it received to the computer, where a Graphic User Interface (GUI) presents the collected data in a more user-friendly way. The GUI, designed with the software Qt, helps the users to monitor which nodes in the network are active. When the user issues the SENSOR_REQUEST from the GUI in its console, the Coordinator sends a request packet to all active End Devices. The End Devices would reply each with its own ID and its temperature reading.

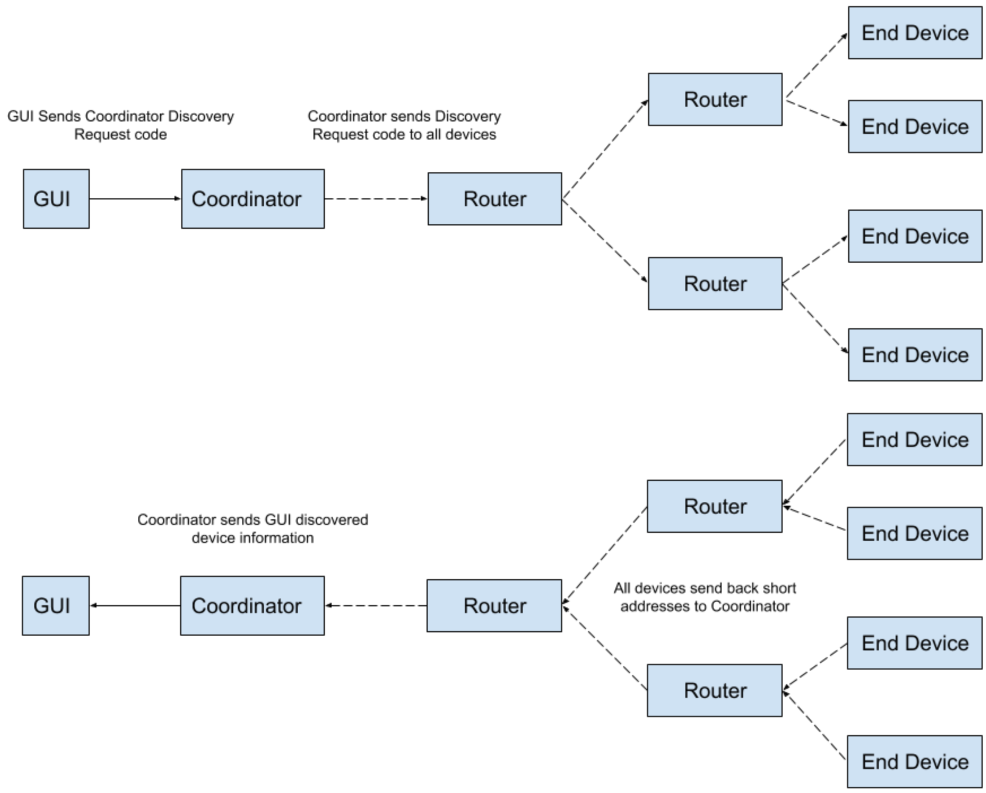

Figure 7: Discovery Request and Response Data Flow

Since ZigBee devices are designed to easily associate and disassociate itself within a network, the network is theoretically scalable. A node can toggle itself in and out of the network, therefore joining or leaving the network depending on the needs of the user. To identify which nodes are active, a user can issue the command DISCOVERY_REQUEST from the GUI's console. The Coordinator would send a request packet to all devices to identify themselves by replying with each its own ID.

2.3 Software

A piece of software was developed to request and read sensor values and node identifying information from the ZigBee network. It functions by allowing the user to connect to the coordinator utilising serial uart communication. Once the user has connected to a coordinator, they can send requests to the network. The two functioning requests are the sensor request, and discovery request.

The sensor request sends out a packet to all devices on the network requesting sensor values. Once received by end devices, a reply packet is sent containing the sensor values. These values are automatically displayed in the software.

The discovery request sends out a packet to all devices on the network requesting addresses and other identifying information. Once received by any device, a reply packet is sent containing identifying data. These values are automatically displayed in the software.

The Figure 8 below shows a screen shot of the the GUI while issuing the discovery request to the network.

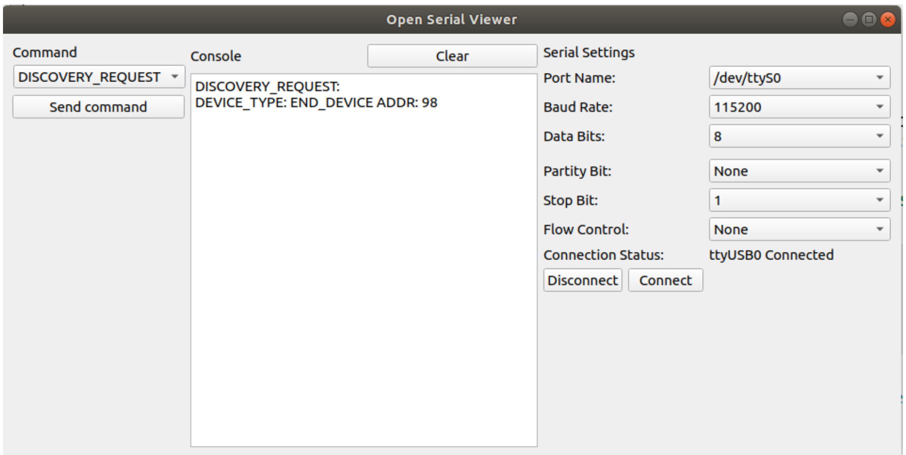

Figure 8: Response to DISCOVERY_REQUEST in the GUI

A screen-shot of the GUI when sending the SENSOR_REQUEST is shown in the Figure 9 below, with its response. For the prototype implementation, the sensor value is tied to the potentiometer that is attached to ADC7 on the TI development boards.

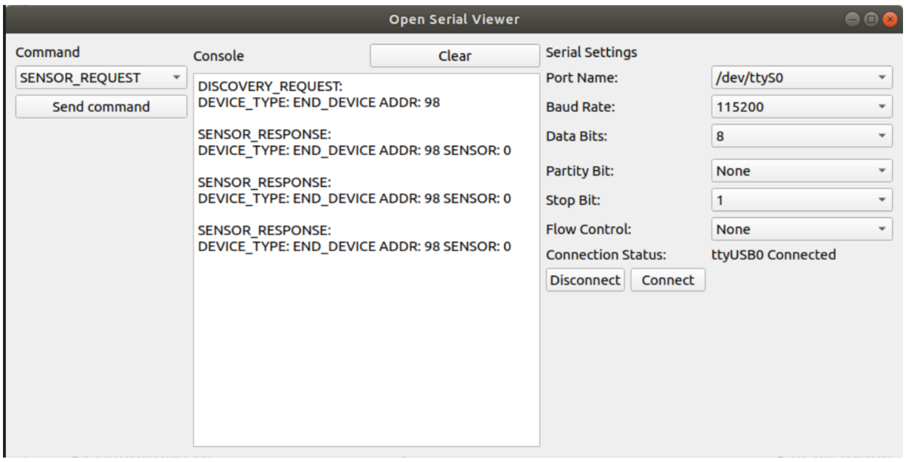

Figure 9: Response to SENSOR_REQUEST in the GUI

3 Visualising the Data

By storing the data that is obtained from sensor requests, it is possible to create a visual representation over time of the sensor values. The following figure shows an example of this. Multiple senor values are displayed at the same time on the line plot. The user is able to switch the specific type of measurement that is currently visible.

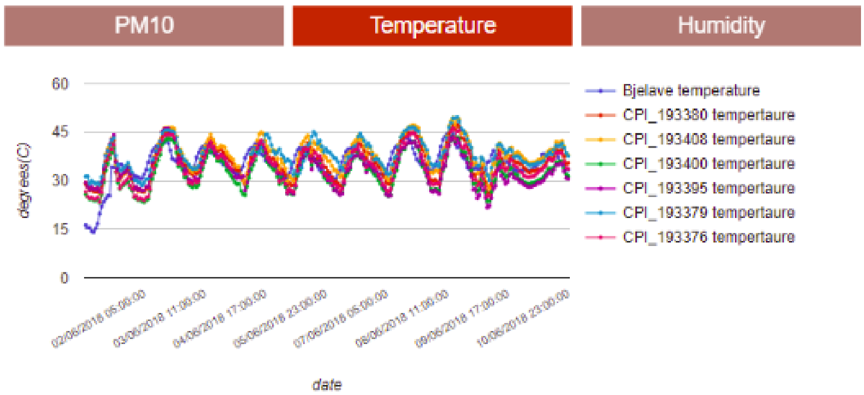

Figure 10: An example of a visualisation of the sensor request data.

4 Testing and Results

Although the three devices are designed to interact seamlessly within a fully meshed network, it is noted that the End Device would automatically connect itself to the Router when the latter is activated, and stop sending packets directly to the Coordinator. Due to this behaviour, it is noted that in a small ZigBee topology, having no Routers may speed up the connections between Coordinator and End Devices. The absence of Routers will reduce the time taken for End Devices to send packets to Coordinator, since there will be one less node for the packet to flow through, and therefore minimising the chance of a dropped packet.

During the testing of the whole system, a Linux Ubuntu laptop was used. While the Coordinator is connected to the laptop, the GUI listens to its /dev/ttyUSB0 port. In Linux machines, the /dev/ttyUSB0 is the port where data from Serial cables are directed. The GUI then processes the data packets into bytes with one stop bit. This configuration led to a clean, smoother data reading on the GUI's console screen.

The Figure 11 shows the three ZigBee devices being tested. The device on the left behaves as a Coordinator sending request packet to active devices, with the "Rsp Sent" on the LCD screen to notify the user that the packet has been sent.

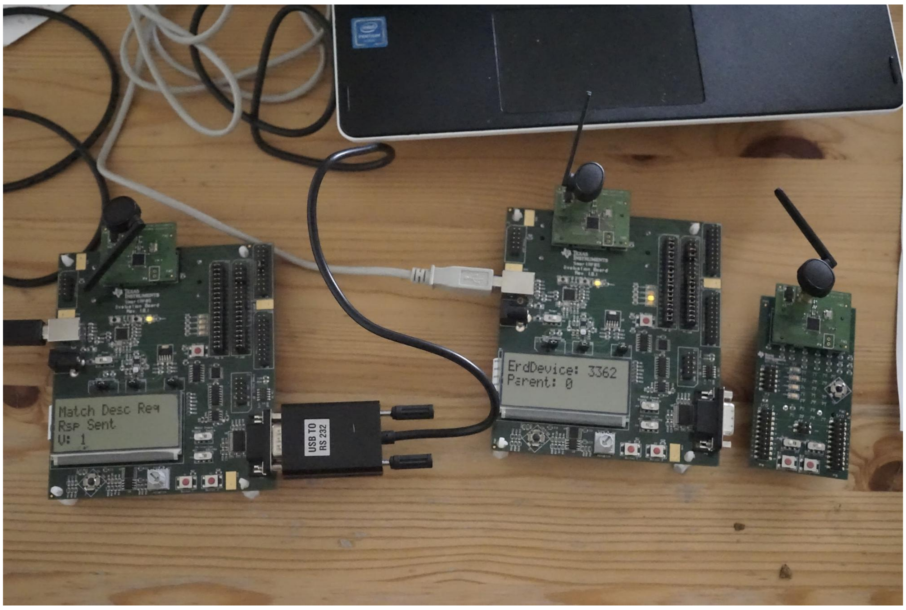

Figure 11: ZigBee devices during testing

Since ZigBee protocol is heavily based on Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) method, inefficiency in sending and receiving packets were noted. CSMA/CA protocol in the ZigBee devices would listen to the channel where it is sending packets and ensure the channel is empty before starting to transmit the packet in its entirety. Since ZigBee devices, especially in the scope of this project, can only transmit packet over the Industry, Scientific and Medical (ISM) radio bands (902 MHz to 928 MHz in Australia), the network would compete for empty channels with other networks present in the surrounding. This problem was apparent during testing in the laboratory, where the network would fail to send and receive packets when other ZigBee devices configured by other teams were present.

5 Conclusion

Although ZigBee is a mature technology facing competition with newer protocols and technologies such as Low Range Wide Area Network (LoRaWAN), it is still a reliable way to collect data using sensors and send packets to extract them. ZigBee's characteristic of self-healing a connection is useful to re-establish a lost connection to recently inactive sensor node, or to self-associate a new End Device. This characteristic benefits the user, since it minimises the power being used by preventing to send request packets to inactive nodes. The sensor network is also scalable, as devices can be easily introduced into the network with little amount of manual configuration by the network administrator is required, when the demand to increase the network coverage size grows.

References

[1] D. Giancono, "Sarajevo historic air pollution and air pollution sources data," Jun 2018.

[2] European Environment Agency, "Temperature inversion traps pollution at ground level," Nov 2016.