ZigBee-based climate measurement system for thermal comfort in traffic busses

ZigBee-baserat mätsystem för termisk komfort i bussar

TALIB ALMOSAWI

GEORGE KARANTA

ZigBee-based climate measurement system for thermal comfort in traffic busses

ZigBee-baserat mätsystem för termisk komfort i bussar

Talib Almosawi
George Karanta

A thesis presented for the degree of electrical engineering

Abstract

This thesis describes the work process two (WP2) of a project led by the traffic administration in Stockholm to achieve the passengers' thermal comfort in traffic buses. The purpose of this thesis is to design and implement a measurement system comprised in eight measurement position in a bus. Each position is able to transmit data using a wireless communication protocol, with a central unit collecting all the transmitted data taking into consideration the accuracy of the measurements and filtering out the incorrect data.

In order to accomplish this, a literature study is conducted investigating different measurement methods for different parameters, different communication protocols were analysed as well to determine an appropriate communication protocol suitable for climate measurement in a bus environment with passengers present on board.

The output of the HVAC system is also required to help the traffic administration demonstrate the results of adjusting the temperature in the bus to reach the thermal comfort and how it led to saving energy.

The system incorporates ZigBee , ESP32 microcontroller in the sender positions and a Raspberry pi operating as a central unit to collect data and store it in a file system providing an accurate data that will help the traffic administration achieve their goals.

Unfortunately, there were some difficulties accessing CAN system data to provide the output of the HVAC system.

Keywords: Measurement system, wireless personal area network, ZigBee, radio communication, Raspberry pi, microcontroller

Sammanfattning

Detta examensarbete beskriver arbetsprocess två i ett projekt som leds av trafikförvaltningen i Stockholm för att uppnå termiska komfort för passagerare i busstrafik. Syftet med detta examensarbete är att utforma och implementera ett mätsystem bestående av åtta positioner i en buss.Varje position kan överföra data med hjälp av ett trådlöst kommunikationsprotokoll, en central enhet samlar in all data med hänsyn till mätningens noggrannhet och filtrering av felaktig data.

För att uppnå detta utfördes en litteraturstudie för att undersöka mätningsmetoder för de olika parametrar. Olika kommunikationsprotokoll undersöktes för att bestämma en lämplig kommunikationsprotkoll för ett mätsystem som mäter klimat i bussar med passagerare ombord.

Uteffekten av HVAC-systemet krävs också för att hjälpa trafikförvaltningen att bestämma fördelarna med att justera temperaturen i bussen för att nå den termiska komforten och hur den ledde till att spara energi.

Systemet inkorporerar kommunikaitonsprotokollet ZigBee, ESP32-mikrokontroller i avsändarpositionerna och en Raspberry pi som fungerade som en central enhet för att samla in data och lagra det i ett filsystem med korrekta data som hjälper trafikförvaltningen att uppnå sina mål.

Svårigheter uppstod med att få tillgång till CAN-system data för uteffekten av HVAC-systemet.

Nyckelord: Mätsystem, wireless personal area network, ZigBee, radiokommunikation, Raspberry pi, mikrokontroller

Acknowledgments

Contents

List of Abbreviations

UITP Union International des Transports Public (French for International Union of Public Transport)

HVAC Heat, Ventilation and Air Conditioning (Systems that are used to move air between indoor and outdoor areas)

CAN Controller Area Network (Standard designed to allow microcontrollers and devices to communicate with each other's without a host computer)

PAN Personal Area Network (Computer network for interconnecting electronic devices)

WPAN Wireless Personal Area Network (Computer network for interconnecting electronic devices in which the connection is wireless)

6LoWPAN IPv6 over Low Power Wireless Personal Area Network (Wireless Personal Area Network protocol)

IETF Internet Engineering Task Force (Open international community of network designers, operators, vendors, and researchers)

BLE Bluetooth Low Energy (Wireless Personal Area Network protocol)

SIG Special Interest Group (A community within a larger organization with a shared interest)

FHSS Frequency Hopping Spread Spectrum (Method of transmitting radio signals by rapidly changing the carrier frequency)

DSSS Direct Sequence Spread Spectrum (Technique used to reduce overall signal interference)

MCU Micro Controller Unit (An intelligent semiconductor IC that consists of a processor unit, memory modules, communication interfaces and peripherals)

GPIO General-Purpose Input/Output (Uncommitted digital signal pin on an integrated circuit or electronic circuit board)

API Application Programming Interface (software that allows two applications to talk to each other)

PHY Physical layer (Electronic circuit required to implement physical layer functions of the OSI model)

MAC Medium Access Control layer (Controls the hardware responsible for interaction with the wired, optical or wireless transmission medium)

NWK Network layer (Ensures the proper operation of the MAC layer and provides an interface to the application layer)

APS Application Support layer (understands applications and filter packets)

AF Framework layer (Allows individual software components to communicate)

ZDO ZigBee Device Object (Application running on endpoint in every ZigBee device)

CSMA/CA Carrier-Sense Multiple Access Collision Avoidance (network multiple access method in which carrier sensing is used, but nodes attempt to avoid collisions by beginning transmission only after the channel is sensed to be idle)

PTC Positive Temperature Coefficient (Materials that experience an increase in electrical resistance when their temperature is raised)

NTC Negative Temperature Coefficient (Materials that experience a decrease in electrical resistance when their temperature is raised)

AC Compressor Air-Conditioner Compressor (Power unit of the air-conditioning system)

OBD-II On-Board Diagnostics II (Cable that provides access to data from the engine control unit)

CSV file comma-separated values file (Plain text file that contains a list of data)

OS Operating System (System software that manages computer hardware and software)

ADC Analog to Digital Converter (System that converts an analog signal)

RSSI Received Signal Strength Indicator (Measurement of the power present in a received radio signal)

1 Introduction

1.1 Background

The International Association of Public Transport (UITP) started the task of determining how to evaluate passenger's thermal comfort [29]. The project's main goal was to measure parameters related to human's climatic conditions such as temperature, humidity, wind speed, Black bulb temperature and CO2 emissions.
The results of these measurements will help determine temperature values for buses used in public transport. Adjusting the temperature to not exceed nor go below the thermal comfort for passengers, and at the same time reduce the power consumption of elements used for the heating and ventilating process.
Thermal comfort is described as the satisfaction with the surrounding climatic situation in a specific environment, it can be reached by studying and adjusting some climate related factors as the ones to be measured in this thesis [5].
The output effect needed for the heating, Ventilation and Air Conditioning (HVAC) system is to be measured as well to provide the opportunity to compare the output effect before and after temperature adjustments.

This thesis represents the second stage of a big working process led by the Swedish traffic administration in Stockholm (Trafikförvaltning). The Swedish traffic administration are responsible for this work and will implement it on Arriva's traffic buses, the aim is to apply UITP's Bus Committee thermal comfort project in Sweden.

The study conducted by the traffic administration consists of several work processes, this thesis covers work process two (WP2). The main purpose of these two processes is designing and implementing a measurement system that collects data from sensors in addition to preforming a field test and collecting its results [29]. The output power of the compressor that drives the HVAC system are also desired by the traffic administration to be able to calculate the HVAC system's output power. This information

can be collected via Controller Area Network (CAN system).

The measured data will be used by the traffic administration in the upcoming working processes to study electricity consumption and thermal comfort with different weather conditions.

1.2 Problem definition

The design of the measurement system will be divided into subgroups that are all connected wirelessly to a central unit to avoid long cables in the bus's passage; especially when testing with passengers on-board.
The temperature, black bulb temperature, relative humidity, CO2 emissions and wind velocity are the required parameters and are to be measured with a good precision every ten seconds, in addition to calculating the average of every parameter every minute. A validation of the results accuracy using a special instrument provided by the traffic administration. HVAC system output data is to be calculated as well.

To fulfill these requirements some main problems have been defined:

- Which wireless system protocols can be used in order to accomplish the desired requirements specification?

- Design of measurement methods, accuracy, precision and need for filtration.

- How to organize and divide sensor data in frames to be sent wirelessly and establish a wireless server that receives these data frames with the least amount of packet loss?

- How to recreate the received segments at the server side and present the sent data in right order?

- Verification of results obtained from different sensors in different positions.

- How is the output power of HVAC system can be measured?

- How to send requests using the CAN system data frames via On-Board Diagnostics II cable?

1.3   Purpose

The purpose of this thesis is to design and incorporate a proof-of-concept wireless climate measuring system in buses by using specialized sensors and WPAN technology. The system provides temperature data for monitoring and analysis in order to achieve thermal comfort for passengers and reduce energy consumption in buses despite different weather conditions.

1.4   Goal

The goal of this thesis is to design a wireless network of sensors connected to a central unit for processing and presenting sensor data.
This goal can be met by achieving the following:

- Choosing the most suitable measuring technology for the bus environment.

- Design an MCU-based wireless system consisting of 8 groups to measure the requested parameters.

- Filtering the inaccurate data to avoid incorrect measurements.

- Analyzing the expected interference that might affect the wireless communication.

- Packing data in correct format and sending to the server.

- Devise a plan to unpack and correctly map the sent sensor data at the receiver side.

- Investigate how to get the power output from the compressor that drives the HVAC system.

1.5   Delimitations

The bus used for this study is not an electric-bus and thus its HVAC system is not depending on electricity completely. Cooling and additional heating are electricity powered by a compressor and thus can be measured.
The CAN system can only be tested with a wired connection directly to the bus, but due to the current Covid-19 situation, the bus field tests are too limited which might cause insufficient testing.

2 Background and theory

2.1 WPAN protocols

Wireless Personal Area networks (WPAN) are used to connect devices on a network with a unique ID. There are several protocols for WPAN that have different use cases, the following are among the most used:

6LoWPAN
IPv6 Low Power Wireless Personal Area Networks (6LoWPAN) is an open standard protocol defined by Internet Engineering Task Force (IETF). It communicates over IEEE 802.15.4 standard. 6LoWPAN is distinguished for providing a layer between the physical layer and network layer for handling IPv6, this 6LoWPAN layer provides the opportunity to transmit IPv6 packets over IEEE 802.15.4 link [41]. 6LoWPAN operates on 2.4 GHz frequency band, it supports a data rate of 250 kbps with a range of 10-100 meters, mesh networking is possible on 6LoWPAN networks [16].

ZigBee
ZigBee is a communication protocol based on IEEE'S 802.15.4 standard to create wireless personal area networks developed by the ZigBee alliance; a non profit mutual benefit corporation under the laws of the state of California [45].
The alliance was formed for the purpose of bringing exposure to connected devices by promoting open standards, provide an environment for members to discuss enhancement of specifications, promote the advantages of connected devices, offer protection to consumers and businesses, maintain and improve relationships with educational, research and government institutions and stimulate competition to increase development [45]. ZigBee is characterised for it's low power consumption, low cost and low data rates. ZigBee operates on the 2.4 GHz, 915 MHz and 868 MHz frequency bands for the data rates of 250, 40 and 20 kbps respectively [16][2].

BLE

Bluetooth Low Energy (BLE) is a protocol developed by the Bluetooth Special Interest Group (SIG), a low energy variant of classic Bluetooth that based on the IEEE 802.15.1 standard. It operates on the 2.4 GHz frequency, which is similar to ZigBee and 6LoWpan. BLE can transmit in up to 40 channels, while classic Bluetooth could transmit in up to 79 [34]. BLE has a range of 15-30 meters with a data rate of 1 Mbps [16].

Each of the aforementioned technologies in this chapter offers certain advantages in different use cases. BLE offers a higher data rate at 1 Mbps but has shorter range at 15-30 meters compared to ZigBee and 6LowPAN's range at 10-100 meters [16]. These protocols support different networking topologies like mesh, star and trees. Network routing schemes are designed to ensure power conservation, and low latency through guaranteed time slots [2].

2.1.1   ZigBee configuration

ZigBee device types

A ZigBee device can be configured into one of three configurations: Coordinator, router and end device.
A PAN may only have one coordinator configured at a time, it works as a manager of the network; maintaining the connection to devices on the the network and buffering data for sleeping end devices. Each PAN has a unique PAN ID given to the coordinator, a coordinator is always on in the network.
There is no limitation to the number of devices that can be configured as router on the network. A router can extend the range of the network by receiving and sending data to accessible devices that the coordinator has otherwise no access to. The router cannot sleep.
The number of devices configured as end devices does not have a limitation as well. End devices cannot allow other devices to join the network nor extend the range of the network, they can be set to sleep mode using different methods and in a range of intervals [35].

API mode

In order for data to be transmitted through the serial interface, the Xbee module has to be configured to either Transparent mode or Application Programming Interface(API) mode, with transparent mode being the default configuration for all xbee

devices [36]. Data sent and received in Transparent mode is queued with other data on the serial interface without additional data to mark the source of the sender, this means the data received cannot be identified or linked to a sender. API mode is more complex than transparent mode but provides many advantages like the ability to send to several devices, provide sender address, transmission details and overall more advanced features [37].

Data packets in API mode are called frames, consisting of five fields; start delimiter, length, frame type, data and checksum, the frame format can be seen in figure 2.1.

Start delimiter	Length		Frame data								Checksum
			Frame type	Data							
1	2	3	4	5	6	7	8	9	...	n	n+1
0x7E	MSB	LSB	API frame type	Frame-type-specific data							Single byte

Figure 2.1: Xbee data frame [38]

2.1.2   ZigBee topologies

Star topology

A star topology is a network configuration used for smaller networks, it consists of one coordinator which is the highest level node in the network. Other nodes on the network are configured as end devices or routers, although the routers are not used to relay data from other nodes and are operating like an end device without the sleep functionality. The children nodes relay data directly to the coordinator. A disadvantage with a star topology is the that in case the the coordinator stops working all the traffic in the network is stopped [9]. An example for a star topology can be seen in figure 2.2.
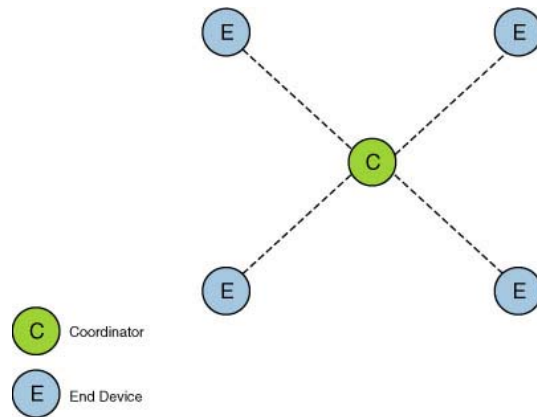
Figure 2.2: Star Networking in ZigBee [4]

Tree topology

A tree topology is not as simple as the the star topology. The nodes in the network have three different levels unlike the two levels in the star topology. The highest level node in the network is the coordinator. Other nodes can be configured as end devices or routers. The routers in the network have the ability to relay data sent from end devices to the router either directly or indirectly through another router that connects directly to the coordinator as seen in the example in figure 2.3. This network topology has the ability to extend the range of the network and get access to nodes that the root node could otherwise not have access to [9].

Figure 2.3: Tree Networking in ZigBee [4]

Mesh networking

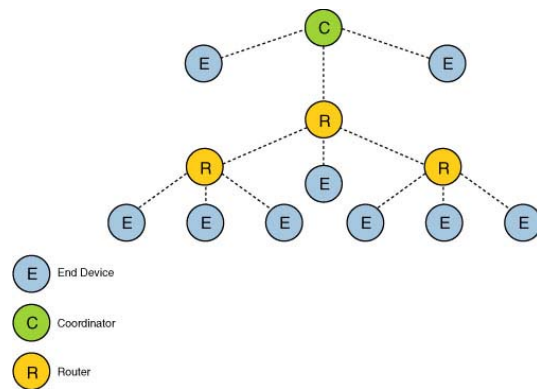Mesh topology is the most complex networking topology used for ZigBee. It has all three nodes types; coordinator, router and end device as shown in figure 2.4. The main feature of this topology is having multiple paths to the destination. The ability to take different routes to the source provides the network self healing capabilities in case if an error occurs on a certain node on the path to the destination. The ability to find a new path to the destination is called route discovery. A mesh network has lower latency and a multi-hop options. The main disadvantages in mesh networks are high cost, storage for routing tables and inability to create super frames [9].
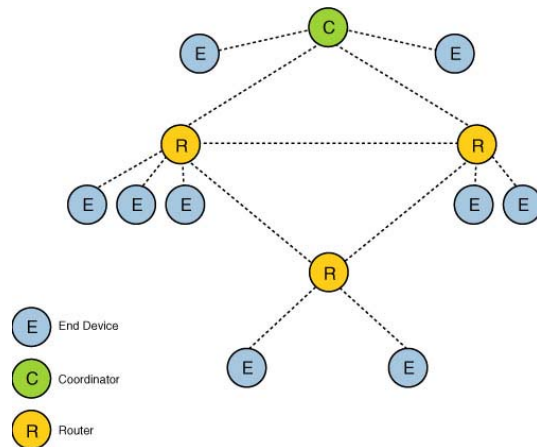
Figure 2.4: Mesh Networking in ZigBee [4]

2.1.3   ZigBee Architecture

The ZigBee protocol consists of five layers; Physical layer (PHY), Medium Access Control layer (MAC), Network layer (NWK), Application Support layer (APS) and the Application Framework layer (AF). PHY and MAC are based on the IEEE 802.25.4 standard, the upper layers are part of the ZigBee specification, furthermore a cross-layer called ZigBee Device Object (ZDO) is in the specification. The ZigBee specification is similar to the IP specification, the main difference lies in the ZigBee specification's inability to have different types of PHY and MAC layers as it is exclusive to IEEE's 802.25.4 PHY and MAC [6].

Each layer provides different functions to the network, PHY handles data transmission on the channel while MAC manages medium access between neighbouring radios on the network. NWK provides the network with the ability for different routing op-

8

tions for data and APS manages the data between NWK layer and APS layer. AF handles distribution of data to applications and ZDO layer provides services to and interactions with the lower layers [39].

Application Layer (APL)

| Application Framework | Zigbee Device Object | Application Support Sub-layer |

Network Layer (NWK)

| Security Management | Message Broker | Routing Management | Network Management |

Defined in the ZigBee Specification

Medium Access Control (MAC)

| beacon mode Slotted CSMA/CA GTS allocation | non-beacon mode Non slotted CSMA/CA |

Physical Layer (PHY)

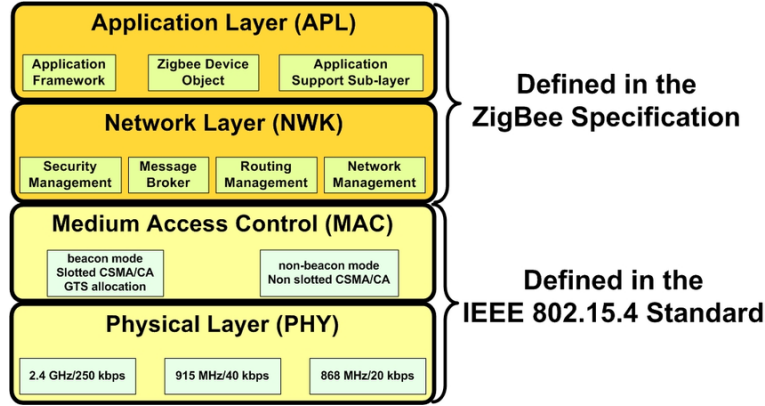| 2.4 GHz/250 kbps | 915 MHz/40 kbps | 868 MHz/20 kbps |

Defined in the IEEE 802.15.4 Standard

Figure 2.5: ZigBee Architecture Layers [39]

In order to to avoid collision a method called Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) is used. This access methods senses for traffic on the nodes before sending, it does not send until the path is clear [6].

2.1.4   2.4 GHz interference

ZigBee, Bluetooth and WiFi are on the 2.4 GHz frequency band [16], this means that interference can happen because the protocols are sharing the same medium, this interference can cause packet loss or corruption. Bluetooth overcomes interference by incorporating Frequency Hopping Spread Spectrum (FHSS); this method divides the frequency band into 79 or 40 channels depending on Bluetooth standard and hops rapidly between channels. WiFi and ZigBee both use Direct Sequence Spread Spectrum (DSSS) [33]. ZigBee operates on channel 1, 6 and 11. Channel 1 and its side of bands overlaps with channels 11-14 on ZigBee, while channel 6 overlaps with 15-19 and channel 11 overlaps with 20-24, leaving channel 25 and 26 without interference [17].

2.2 Measurement system

2.2.1 Temperature

Temperature can be measured using thermistor, which is a thermometer that uses resistance to calculate temperature.

This kind of thermometers has two types; Negative Temperature Coefficient (NTC) and Positive Temperature Coefficient (PTC). The resistance of an NTC thermistor decreases as the temperature increases. While in PTC thermistor when the temperature increases the resistance increases as well [13].

Thermistors can be analog to digital (ADC) sensors that need to be calibrated and converted from Kelvin to Celsius to give a good approximation of the temperature or I2C digital sensors that are already calibrated in Celsius, and gives the temperature value by calculating the temperature output signals from the SDA (data line) pin using this formula [23]:

$$\boxed{T\,(C) = (temperature\ output\ signal/2^{20}) * 200 - 50} \tag{2.1}$$

2.2.2 Relative humidity

Humidity parameter can be measured by identifying the changes of various components such as resistance and capacitance to detect the amount of water in air. Humidity is measured with the percentage of relative humidity (%rH) [11].

Relative humidity sensor types can be equivalent to the temperature sensor types. An ADC sensor that needs to be calibrated and I2C digital sensors that are already calibrated and uses the relative humidity signal received from the SDA pin to calculate the relative humidity using this formula [23]:

$$\boxed{rH(\%) = (relative\ humidity\ signal/2^{20}) * 100} \tag{2.2}$$

2.2.3 Equivalent CO2

Carbon dioxide can be obtained by heating a surface for a couple of hundreds Celsius degrees, this technique is not effective and can be used only in special environments. Another technology called micro hot-plate was used in many sorts of gas sensors, this technology uses a suspended heating structure to isolate the sensor from the surroundings to help the sensor detect the desired gas. This method avoids the excessive heating and power consumption [15].

2.2.4 Wind velocity

Wind velocity or wind speed can be obtained using sensors operating with different principles. The three main principles to calculate wind speed are wind cup, ultrasonic and hot-line (hot-wire) principle.

The wind cup depends on the wind causing the cup to spin at different rates. The spinning speed of the wind cup is equivalent to the wind velocity [22], as shown in figure 2.6.
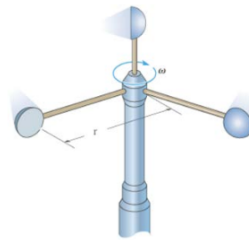

Figure 2.6: Wind cup principle

The relationship between the wind speed and spinning speed can be described using the following formula [22]:

$$\boxed{\begin{aligned} v &= kn + b \\ n &= M/t \end{aligned}}$$

(2.3)

$$\boxed{\begin{aligned} k\,and\,b &= constants \\ n &= cup\,rotation\,rate \\ v &= wind\,speed \\ M/t &= voltage\,pulse\,every\,second \end{aligned}}$$

While anemometers using ultrasonic principle can measure wind velocity by measuring the Time of Flight (ToF) which is also the transmission time needed for signals to go from the sender to the receiver. This principle helps to determine the wind speed and direction but has some flaws when it comes to locations where the wind speed is continually changing [27].

The "Hot-Wire" principle means heating an object to reach a specific temperature and then measuring the amount of electrical power needed to keep the object on the

same temperature while the wind shifts, the heat applied on the object can then be calculated to estimate the wind speed. The devices using this technique are considered extremely vulnerable to low wind speeds [21]. Hot wire principle can be shown in figure 2.8, while a sensor using this principle is shown in figure 2.7 and point (R4) represent the sensitive element.
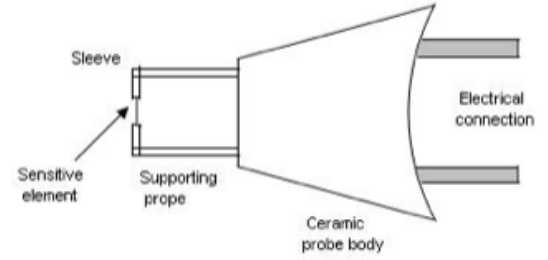

Figure 2.7: Wind Sensor Rev. C. [44]

Figure 2.8: Illustration of the hot wire technique

2.2.5 Black bulb temperature

Black bulb or black globe temperature is the evaluating of heat stress on a specific area. This parameter can be determined theoretically using this formula [14]:

$$T_{WBG} = 0.7\,T_{nw} * 0.2\,T_g * 0.1\,T_a \tag{2.4}$$

$$
\begin{aligned}
T_{WBG} &= wet\,bulb\,globe\,temperature \\
T_{nw} &= natural\,wet\,bulb\,temperature \\
T_g &= black\,globe\,temperature \\
T_a &= dry\,bulb\,temperature
\end{aligned}
$$

Alternatively, a black globe thermometer can be used to get this parameter. The sensor consists of a temperature sensor placed in the center of a half circular shaped

12

black plastic, as shown in figure 2.9. The sensor collects the radiation from various heat sources and surfaces to give an estimated value of the black globe temperature on those sources [18].

Figure 2.9: Black bulb temperature sensor [32]

2.3 HVAC system

HVAC stands for Heating, Ventilation, and Air Conditioning. HVAC system refers to the systems that are used for moving the air from one area to another along with heating or cooling [19].
These systems are used in various places such as residential buildings, offices and different vehicle types like trucks, cars and buses. Despite being mainly known for cooling and heating, the HVAC systems are also used to maintain humidity and prevent moisture formation through the ventilation unit. A bus's HVAC system is shown in figure 2.10 below.
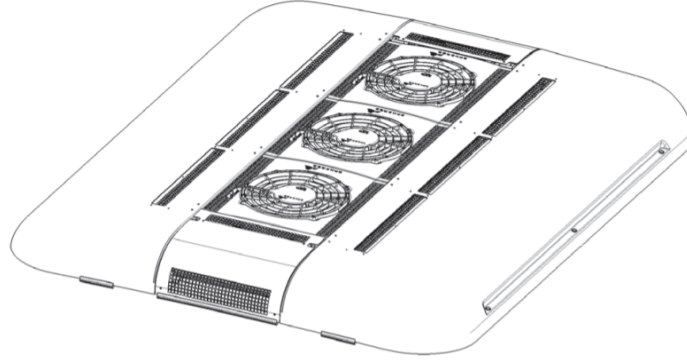
Figure 2.10: 3D graphic of the HVAC system [42]

HVAC systems comes in several variations but all of them uses the same concept that is based on fresh air, the differences are in the sources of energy as some of the system are based on electricity, or uses electricity represented in an AC compressor in some of its features while other features runs on the bus's engine or a water tank [19].

MAN Lion's city A45 bus; that was offered for the project by Arriva, has Valeo (Spheros) REVO air-conditioning system that is driven by Bock AC compressor. The compressor has a large speed range that can provide a huge cooling capacity and can reach a maximum cooling output 32 kW [40].

The output power of the HVAC system can be measured by calculating the output power of the AC compressor by retrieving related parameters from CAN system (output pressure, suction pressure, compressor speed and refrigerant temperature) and using Bock stationary applications website [43].

2.3.1   CAN System

Controller Area Network (CAN) is a serial network technology developed initially for the automotive industry. However, it later proved to be able to serve the needs of networks in other industries such as industrial and home automation.

CAN is reachable using On-Board Diagnostics II (OBD2) connectors and it provides fast real-time communication with a 1 Mega bit/s baud rate [7].

CAN system basically uses messages or data frames in its process. The user should send a data frame (Request) that contains the ID of the desired information and receive another frame (Response) that carries the answer [1].

Data frame

CAN system is a broadcast. It listens to all the transmissions happening in the devices connected to its node, it is considered a multi-master, message broadcast system [1].
There are four different types of messages (frames) that can be send to the CAN Bus:

- Data Frame

- Remote Frame

- Error Frame

- Overload Frame

All these types of frames have the al most the same architecture but with two different types of identifier one is the standard form with 11-bit identifier and the other one is the extended form with 29-bit identifier as shown in figure 2.11, eleven of them indicates to the unique ID of the requested data [1].
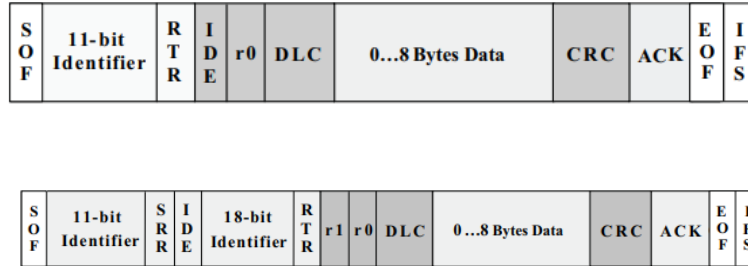

Figure 2.11: CAN bus data identifier [1]

2.4 Related works

A brief inspection of related studies was made in this section. As this thesis studies solutions to construct a wireless connected measurement system and will for that focus on systems using WPAN protocols, different weather measurement systems, thermal comfort measurement systems and HVAC related works.

2.4.1 UITP measurement system in Berlin

"Modeling and Experimental Investigation of Thermal Comfort and Energy Consumption in a Battery Electric Bus" [30] is a study that has been previously done at Berlin institute of technology. In this thesis a measurement system was implemented in a 12 meter electric bus in order to investigate the influence of HVAC-system and the energy consumption.

Data was saved locally on memory cards for each microcontroller and no central unit is used to manage data received from sensors on an external device. HVAC system output data was obtained by using a third party application that was already implemented by the company owning the bus.

2.4.2 Computer-aided measurement system

In the study "A Computer-Aided Modeling and Measurement System for Environmental Thermal Comfort Sensing" [10], a measurement system is designed and implemented using the predicted mean vote (PMV) model. The system is used to present indoor measurements and designed in a box shape where it can be located in different positions to give accurate data of temperature, relative humidity, CO_2 emissions, wind velocity and black globe temperature.

These values are sent into acquisition instruments after converting them to voltages and currents with signal conditioning, afterwards the data is sent to the computer aided PMV for further analysis. The data is saved on a server where users with access are able to view the results.

2.4.3 Wireless Green house management

The study "Wireless sensing and control for precision Green house management" [8] focuses on the usage of ZigBee protocol to send weather related parameters, such as; temperature and relative humidity to a host computer that displays these data in Labview called application.

2.4.4 Low-cost weather system

The paper "Low Cost IoT enabled Weather Station" [28] studies the analysing of measured temperature, relative humidity and wind speed data using Thingspeak server to display the information in graphical designs.

3 Methodology

This chapter will explain the methods used to fulfill the requirements and reach the thesis goal described in the first chapter. A literature study to the available methods and their regulations was preformed to help decide the most suitable technique to implement on this project.

3.1 Literature study

The aim of the literature study is to find and evaluate all the possible information about the WPAN communication protocols, measurement system parameters different techniques and technologies beside the HVAC system's output. The research started with a study about the environment in the bus and the conditions that this system will be deployed in. Along with choosing the best suited electronics for the measurement together with the radio modules will be used to transmit data and investigating their data sheets to have a better acknowledgement about all the part's communication protocol and their operating power.

3.2 Work plan

After the completion of the literature study, a plan was made to help reach the goals and achieve the purpose mentioned in the Introduction chapter. The plan consists of eleven steps:

1. Test every component individually with a simple independent code, observe the results to further develop, enhance or calibrate the code.

2. Test components in groups to check for possible interaction/interference between devices and eliminate errors.

3. Testing radio modules individually, followed by detection tests between two radio modules.

4. Establish a connection between modules and send data in transparent mode.

5. Establish a connection between modules and send data in API mode between two radio modules.

6. Designing a suitable system to the bus's environment, taking into consideration the presence of passengers.

7. Analyze disturbances that can cause packet loss in the chosen communication protocol.

8. Construct a plan to handle framing of data at the sender side and deconstructing the data at the Raspberry pi.

9. Store the deconstructed data in CSV files format.

10. Field testing of the measuring system.

11. CAN system testing using one of Arriva's buses.

3.3 System design

The system consists of 8 different groups located in eight different positions on the bus as shown in Figure 3.1.

Figure 3.1: Measuring positions in a MAN's Lion City A45 bus [29]

Every position consists of various components to measure the desired parameters, while both positions and components order were set by UITP in their previous stage of this project as seen in table 3.2. The microcontroller of every group is planned to be located in a box alongside the power bank and the breadboard containing all the wires. A plastic cable channel will be used to help arrange the components in the desired place as well as hide the cables to make it more suitable for the functioning traffic buses. The sensors will be located outside the cable channel to assure a good measuring with low disturbances and interference.

Table 1: Distribution of components in each position [29]

Position	Air temperature	Air humidity	Air velocity	Globe temperature	Seat temperature	CO2 concentration
1	Yes (4)	Yes (1)	Yes (1)	Yes (2)	No	Yes (1)
2	Yes (4)	No	No	Yes (2)	No	No
3	Yes (4)	No	No	Yes (2)	No	No
4	Yes (4)	Yes (1)	Yes (1)	Yes (2)	Yes (2)	Yes (1)
5	Yes (4)	Yes (1)	Yes (1)	Yes (2)	Yes (2)	No
6	Yes (4)	No	No	Yes (2)	No	No
7	Yes (4)	Yes (1)	Yes (1)	Yes (2)	No	Yes (1)
8	Yes (4)	No	No	Yes (2)	No	No

3.4 Communication

ZigBee is chosen over BLE due to networking capabilities, 6LoWPAN provides many features similar to ZigBee in addition for transmitting IPv6 packets, nevertheless IPv6 is not required due to transmitting data over to a central device for processing. ZigBee is more commercialised and used more than 6LoWPAN at the current time.

To test that radio modules can send and receive data, a connection between two radio modules were established. Both modules are configured using XCTU tool on

windows, the first radio module is configured as coordinator, while the other is configured as router. When the configuration is done the coordinator is connected again to the computer, while the router is connected to a microcontroller via RX and TX pins without any code flashed to the MCU. A scan function is then used with the help of XCTU on the coordinator to detect any devices on the network that has the same PAN ID.

A detection of the router via XCTU indicates that both radio modules are on the same network.

After successfully having both radio modules on the same network, transmitting data is tested, both radio modules are initially configured in a mode called transparent mode, in this mode only one device can communicate with the coordinator, this mode is mainly used for testing or if the network only have two radio modules. The communication is successfully established by writing serial data on the Arduino IDE monitor and successfully receiving it by the coordinator on XCTU monitor. Managing more radio modules than two require API mode, which will be discussed in chapter 4.

In order to avoid interference caused by WiFi due to overlapping in the 2.45 GHz frequency band, The channels that XBee modules may access; can be configured in XCTU. Changing the scan channel parameter for the coordinator decides what channels the network may scan in order to use, scan channel can be changed for end devices and routers as well to decide the channels to scan searching for the coordinator. Forcing the module to use only channel 5 and 6 avoids interference from WiFi due to it operating only on 1, 6 or 11. The scan channel value is set to 0xC000, 1100 0000 0000 in binary, where each corresponds to a channel starting from channel 11 from LSB.

3.5   Hardware

Eight ESP32 microcontrollers were used, one for every position. This type of MCU was chosen because of its tiny size that will make it easy to hide in the bus's environment and its ability to program and control both analog and digital components.[31]. A Raspberry Pi Model 3B accompanied with PICAN2 board and an OBD2 cable will be used at the front side of the bus beside the CAN BUS system socket, it will serve both as a receiver and organizer for all the data coming from the eight positions and CAN BUS data collector to get the required parameters to calculate the HVAC system output.

For the radio modules, nine ZigBee (XBee) modules were used. These modules uses UART serial communication and have wide range (10-100 meter) for both indoor and outdoor communications accompanied with 2.4 GHz frequency [24]. They are considered as transceivers so they can be programmed to act as a transmitter or a receiver. In this project, eight senders for the measuring positions and one receiver connected to the Raspberry pi were needed.

AHT20 Adafruit sensor is used because of its versatile usages, it can collect air temperature, seat temperature and humidity data. The sensor also has a good accuracy at $\pm0.3°$ Celsius for the typical temperature between 18° Celsius and 58° Celsius, and a ±2 percent accuracy on relative humidity values in a typical 20-80 percent relative humidity (%rH) [23], as shown in figure 3.2.
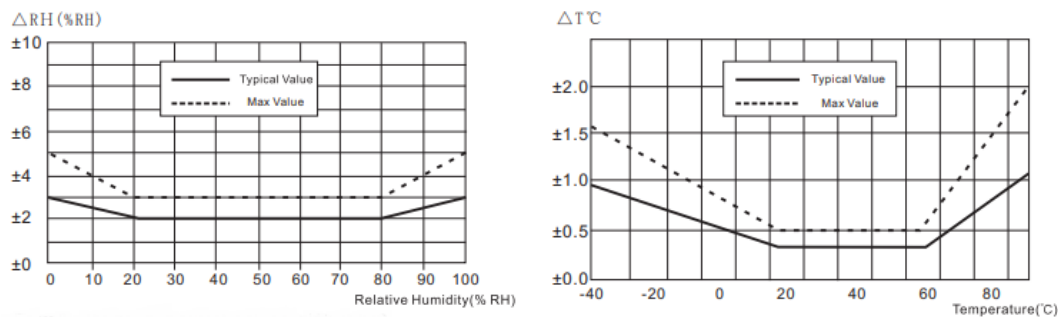

Figure 3.2: Sensor's Accuracy for Humidity and Temperature parameters [23]

CCS811 Adafruit is used to collect Carbon dioxide (CO_2) data due to its low-power consumption and micro hot-plate technology [20], which makes it suitable for power bank driven system that will measure in a bus environment.

A black bulb Temperature analog sensor from Titan Products was chosen due to lack of resources to calculate the black globe manually and because of its good accuracy at $\pm0.2°$ Celsius [32].

Wind sensor Rev. C. will be used to obtain the wind velocity because of its "Hot-Wire" principle [44], as other wind speed sensors rely on fans that does not suit the bus environment or ultrasonic principle that have flaws when the wind speed changes continually.

Both ESP32 MCUs and the Raspberry pi are driven by a 10200 mAh power bank each.

Figure 3.3 shows a block diagram for group 4, which contains the most amount of sensors.
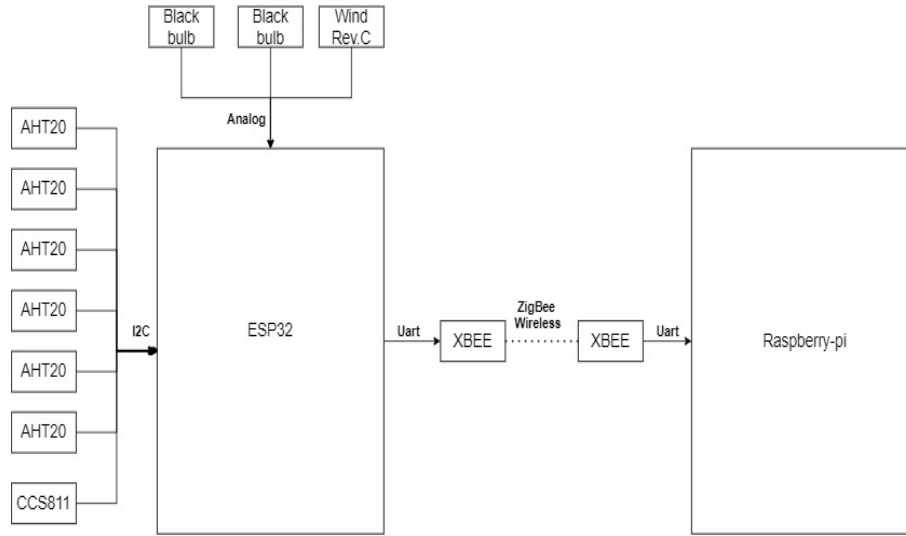
Figure 3.3: Block diagram of group 4

3.6 Software

ESP32 microcontrollers have an official framework called ESP-IDF, they also provide support for Arduino framework on their boards. The MCUs used are programmed using Arduino framework. Arduino has various libraries for the different sensors for this project. Arduino IDE was mainly chosen for convenience and ease of use as well as the variety of libraries available.

The Raspberry pi was programmed in Python mainly using terminal, although a text editor is sometimes used to facilitate changes in code, mainly a text editor called Geany that exists in Raspberry pi OS by default.

Libraries used for receiver program are for receiving ZigBee frames, besides the receiver program other libraries for the CAN are utilized.

In order to test, configure and monitor ZigBee communication, a tool created by the developers of the radio modules called XCTU are also utilized. XCTU provides the opportunity to manage and change parameters for the radio modules, configure the type of device (coordinator, router, end device) and set sleep intervals, etc.

3.7 Parameter data

Arduino IDE was used to program and collect the data from different components and send them to the transmitter device as described previously in this chapter. AHT20 sensor that measure the temperature and humidity using its library package and the command (aht.getEvent(humidity, temp);), both temperature in Celsius (C) and humidity in relative Humidity percentage (rH) can be retrieved as floats. Air Quality sensor CCS811 that measures carbon dioxide emissions has a library package as well to measure the CO2 emission in parts of million (ppm). The measurements are reachable using (ccs.readData();) command as an integer and it starts with 400 ppm, which is the average concentration of carbon dioxide in air. Both black bulb temperature and air velocity are analog to digital conversion (ADC) type of sensors and need to be calibrated correctly in order to get the best estimation. All the collected parameters will then be sent by the radio module (XBee) to the central unit using a payload and xbee.send command.

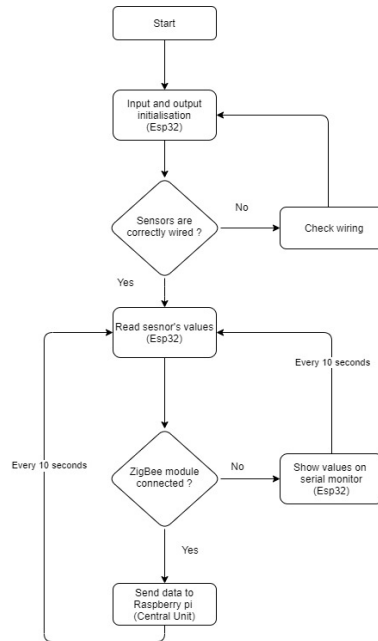Retrieving and sending data can be represented with the following flow chart (figure 3.4):

Figure 3.4: System flow chart

4 Testing and implementation

According to the work plan established in Chapter 3 Methodology, tests were made to all the different components individually and in groups to detect errors and find a suitable solution to reach the optimum design of the system.

4.1 Sensor calibration

Tests for all the sensors were made individually. A reliable instrument called Testo 435 (figure 4.1), was obtained from the traffic administration to assure a good accuracy of the system's measurements.

Figure 4.1: Nordtec Testo 435 device

Nordtec Testo 435 is a device that measures temperature, humidity, carbon dioxide and velocity. It can be used also to determine the black bulb temperature by placing the sensor in a closed room with no windows or surface that will reflect any heat or radiations on it, in that condition it will return a normal room temperature that can be controlled using the Nordtec instrument.

Both the wind and black bulb sensors were analog sensors and in that case they needed to be calibrated to give the correct measurements.

The wind sensor uses the analog pins to read the raw voltage and the temperature data to measure the air velocity in mile per hour (mph) which can be then multiplied with 0.44704 to get the velocity in meter per second (m/s). It can be easily calibrated as well by placing the sensor in a calm environment and adjusting a specific variable named (ZeroWindAdjustment) until it gives a zero value.

The black bulb temperature was acquired by calculating the average of five sampled readings from the thermometer placed in the half circular black piece on top of it and dividing it to the resistance value of 10000 ohm connected to it serially on the breadboard. The data can be calibrated by adjusting the B value, the vector that defines the thermistor's resistive value at its base point. Results were displayed in Celsius (C).

4.2 Inaccurate data

After several tests on the chosen sensors, a problem has been observed with the digital sensor AHT20 that obtain temperature and relative humidity. When placing an object on the sensor front side of the temperature sensor, it gave false values of up to 100 °C, as shown in figure 4.2.
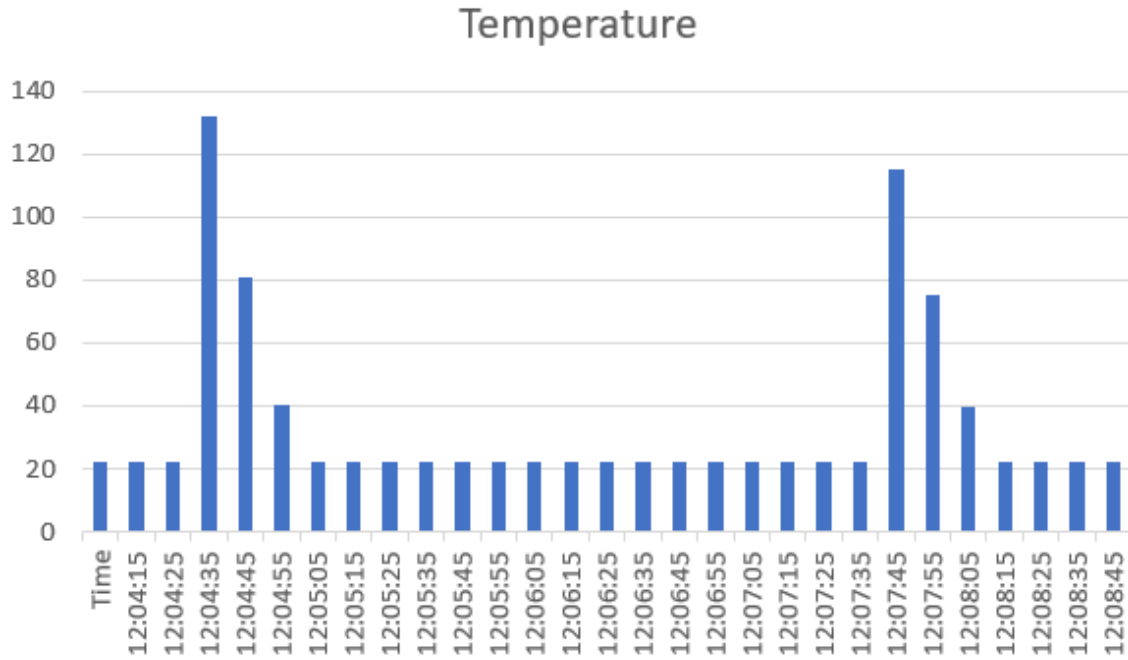
Figure 4.2: Column chart of sampled temperature in a specific time

The previous figure shows the temperature measurements with a sampling frequency of ten seconds between the readings, as required by the traffic administration. An object was placed on the sensor twice; at the beginning and the end, and it took the sensor three whole readings equivalent to thirty seconds to recover and give accurate values again.

4.3 Filtration

Filtration process is important based on the previous test data that will affect the average calculated every minute, a filtration mechanism was used in that case. This mechanism will check the obtained temperature on the sender side and decide if this data needs to be filtered out or not before sending it.
The acceptable degree interval set on this mechanism was between 10 °C and 30 °C. A filter used on the previous test will give the results shown in figure 4.3 below.
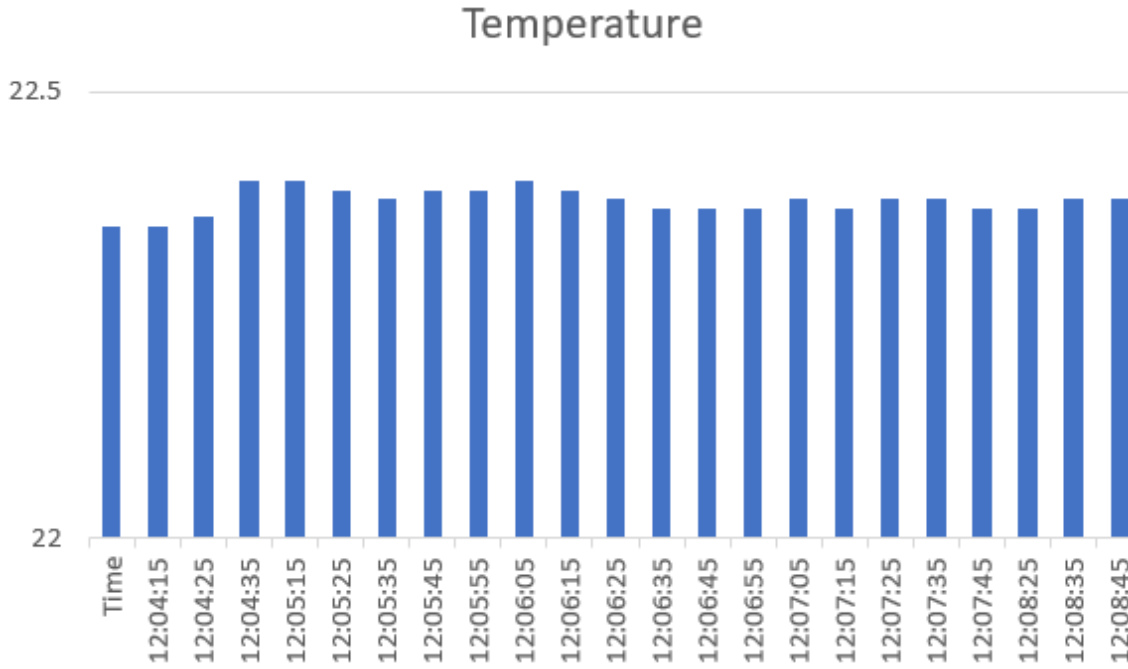
Figure 4.3: Column chart of filtered sampled temperature

4.4 Group integration

Following several individual tests, the groups were assembled as predefined in table 3.2 in methodology. The wiring schematics in Appendix A shows how the sensors were wired and assembled in this project.

A resistor was used between ground pin and the 5V pin of the ESP32 microcontroller, to prevent the power bank from entering the standby mode and the need of turning it on manually. The used resistor was an 82 ohm with a thick silicon film to hinder it from heating inside the box.

In three of the eight groups (1, 4 and 7), a transistor was used as a switch to turn on and off both CCS811 CO2 sensor and AHT20 temperature and humidity sensor to avoid any impact they might have on each others results. The used transistors were PNP type of transistors, they can be used as switches by connecting the load to ground in the sensor and the PNP transistor switches the power to it. To turn on the sensor, the Base terminal is connected to ground.

4.5   Data transmission

4.5.1   Transmitting data

In order to send data via API mode, a ZigBee frame with a fixed payload size is created. The destination address is set to 0X0000000000000000 (default coordinator address). Among the other fields, the frame type is set to ID 0x10 which is a Transmit Request that can be used to transmit wireless data to a specified destination. The checksum field is calculated by taking the sum of the frame data bytes and then subtracting the resulting 8 lower bits by 0xFF, formulas 4.1, 4.2 and 4.3 depicts the calculation of a checksum.

$$
\begin{aligned}
&FT + FID + 64ADD + 16ADD + RF + BR + OPT = x \\
&x \; AND \; 0xFF = y \\
&0xFF - y = ch
\end{aligned}
\tag{4.1}
$$

$$
\begin{aligned}
&FT = Frame\,type \\
&FID = Frame\,ID \\
&64ADD = 64\,bit\,address \\
&16ADD = 16\,bit\,address \\
&RF = RF\,data \\
&BR = broadcast\,radius \\
&OPT = options \\
&ch = checksum
\end{aligned}
$$

When data is received by a sensor it is converted to chars and then parsed into the payload of the frame. After successfully parsing a value, each value is accompanied with a ”,” character to mark the end of a value in the stream of ASCII characters in the payload.
The sensor values in the transmitter side are distinguished by order, which means that a all the values sent using the payload stream has to be unloaded by the same order at the receiver's side.
The required delays for fetching the data are incorporated at the transmitter side as well, after successfully getting all the sensor values and loading them, a delay of ten seconds is added in order to fetch the data in ten seconds intervals.

4.5.2 Receiving data

The Raspberry pi receives data from the coordinator via the serial port, to verify that the data received is correct, the checksum is verified by adding all the bytes excluding the start delimiter and length fields, if the calculated data results in FF at the last two digits rightmost of the value it means that the data is correctly received and no data is lost.

As aforementioned the data received has to be unloaded with the same order it was loaded into the payload. The payload string is read and the values are separated with ",". A new class is created to represent a sensor at each position in a group, the sensor class contains a list that will contain values in the declared sensors position as well as a name and list of the mean values that are to be calculated in one minute intervals.

Fifty nine unique objects of the datatype "Sensor" are made representing all the sensors in the system. After distributing the data the results are written to sixteen csv files where eight files represent each group containing all the data with timestamps in ten second intervals, the last eight files represent the same groups and sensors but instead displays the averages of the ten second intervals also with timestamps.

4.6 Communication tests

4.6.1 XBee range test

This test was done using XCTU software and by measuring the Received Signal Strength Indicator (RSSI) which describes the amount of energy expended during packet receipt [12]. RSSI can be measured using this formula [26]:

$$P_r = P_t + G_t + G_r - 20log_{10}R - log_{10}f + log_{10}(\frac{c}{4\pi})$$ (4.2)

$$
\begin{aligned}
&Pr = Power\,at\,the\,receiving\,antenna \\
&P_t = Power\,at\,the\,transmitting\,antenna \\
&G_t = Transmitter\,gain \\
&G_r = Receiver\,gain \\
&R = Distance \\
&f = Frequency \\
&c = light\,speed
\end{aligned}
$$

Calculating Pr will give the RSSI value [26], while Pt is a given +5 decibel milliwatts (dBm) and both Gt, Gr values are 2.1 decibel relative to isotrope (dBi), the module has a 2.4 GHz frequency as reported by its data sheet [24]. A calculated RSSI by using the stated values and a two meter proposed distance will give -36.86 dBm.

Doing some indoor experiments, a slight change in the RSSI value was noticed when moving the modules towards each other or far from each other. Placing physical obstacles had affected the RSSI value as well. According to the used XBee module's data sheet, the receiver's sensitivity value is around -100 dBm [24].

This test was implemented on an environment designed to resemble the one on the bus; with two meter distance between the tested modules, with several obstacles present between the modules, to simulate interference two Bluetooth connected devices were present in addition to WiFi connection with phones, as interference can cause packet loss when communication protocols are operating on the same frequency band as discussed in Background and theory (Chapter 2)

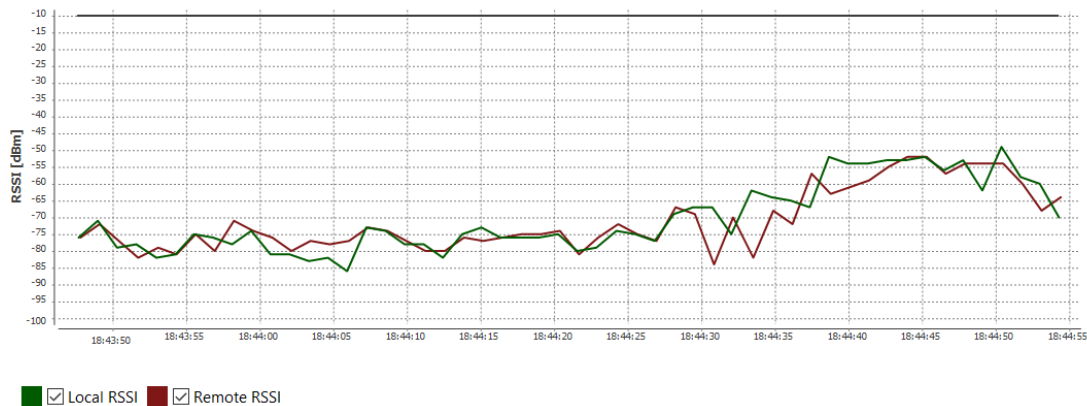The results can be seen in figures 4.4 and 4.5.

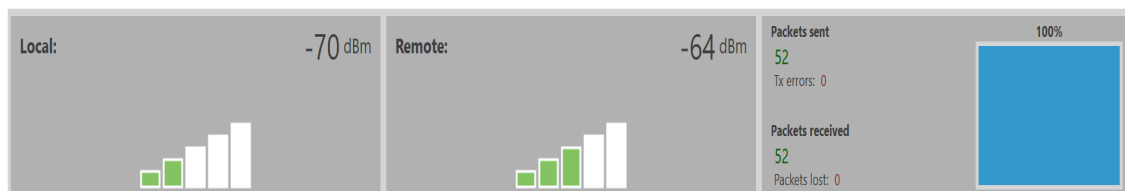Figure 4.4: XCTU test for signal strength in a specific time

Figure 4.5: XCTU test for packet delivery and signal strength

The test result done on XCTU gave 0% packet loss and an RSSI value of -64 on the remote device and -70 dBm on the local coordinator that is connected to XCTU. The graph shows how the signal strength shifted between -50 and -80 dBm under a specific amount of time before settling on the displayed result. Despite the fact that it is lower than the theoretical measured value, this RSSI value is greater than the module's receiver sensitivity which is good and can guarantee a fine percentage of received packets even in further distances it will help avoiding any packet loss [25].

4.6.2   XBee throughput test

The used XBee radio modules operates on a 2.4 GHz frequency which means it should have a 250 kbps throughput according to the 802.15.4 specifications [2]. This throughput value is hard to reach due to many factors that effect the communication's throughput. These factors are represented by the multi-hop paths and the payload packet length [3].

This test was done using XCTU software and only two XBee radio modules, as no multiple hops will be used on the network later on, only a router to coordinator packet sending test was done. The results can be seen in figure 4.6 and showed a stable 8 kbps throughput value under a specific amount of time. A much lower than the typical 250 kbps set by the 802.15.4 specifications but a normal result due to the mentioned factors and the delays caused by the Carrier-sense multiple access with collision avoidance (CSMA/CA) protocol [3].
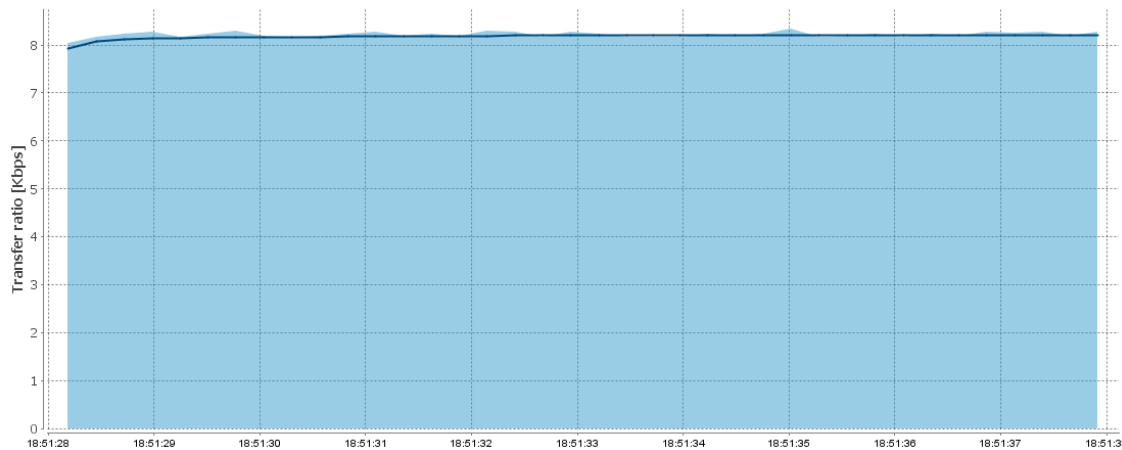

Figure 4.6: XCTU Throughput test between two XBee modules

4.6.3   HVAC system output test

As previously stated in Chapter 2 Background and theory, HVAC system can be calculated using the AC compressor parameters (output pressure, suction pressure, compressor speed and refrigerant temperature) which can be retrieved using "CAN system request" mechanism to send a request and receive an answer containing the desired data.

CAN system request

The PiCAN2 board for the Raspberry pi can be used either with an OBD-II cable or a CAN cable. In order to use it with OBD-II the highlighted Bridges in Figure have to be soldiered together first, the PiCAN2 hat can be installed on the Raspberry pi after soldiering. In order to use the CAN interface certain parameters must be adjusted in the /boot/config.txt file on the Raspberry pi, the SPI interface has to be activated in addition to other parameters as follows:

dtparamdtparam=spi=on
dtoverlay=mcp2515-can0,oscillator=16000000,interrupt=25
dtoverlay=spi-bcm2835-overlay

The datarate is changed as using the command:

/sbin/ip link set can0 up type can bitrate 500000

50000 is the default bitrate for CAN-BUS as well as the CAN-BUS used in this vehicle. The following command is used to listen to all data on the CAN interface:

./candump can0

Another command is also used to send messages to addresses for example, a certain parameter request, using the command:

./cansend

5 Results

This chapter outlines the results of the system's design, the performance of the radio modules and the values obtained from the sensors operating on the chosen technologies. All these factors contributed to the final fully functioning measurement system. This chapter will also highlight the attempts to calculate the HVAC system output by collecting information from CAN system.

5.1 Evaluating the communication protocol

No packet loss has occurred using a simulated communication version of the measurement system, ZigBee was able to transmit data without failure when channels 25 and 26 were used. Actual communication between different transmitters and the central receiving unit was also successful, resulting no packet loss and correct data transmitting. The communication was observed in 2 hour intervals in different occasions and all packets were received with correct interval timing of ten seconds.

5.2 Evaluating the measured data

The measured data were evaluated to ensure the system's good accuracy and reliability to be used in the upcoming stage of the traffic administration's project. These evaluations were obtained by using the Nordtec instrument, that were used in the tests made when assembling the sensor groups.

Temperature

When placing the device in the same position and same distance from the ground with an AHT20 sensor both gave a temperature value of around 22.3° Celsius on the

device and 22.27° Celsius on the AHT20 sensor.

Humidity

Same test were made for the humidity value and both results were around 34.7 and 35 %rH.

Carbon dioxide

The carbon dioxide emissions in the CCS811 sensor gave almost the same result of 560 ppm on the Nordtec device while 556 ppm on the sensor. Nordtec device was a little bit faster in detecting changes on the CO2 ppm while the used sensor was able to give the same result but after a tiny amount of time.

Air velocity

Comparing the sensor's results with the results obtained from Testo 435 device after applying a small amount of wind gave a good accuracy on both the instrument with a 1.33 m/s and the sensor with a 1.30 m/s.

Black bulb temperature

All sixteen black bulb sensors where individually calibrated using the method previously stated in Testing and implementation (Chapter 4). When compared with the used reference (Testo 435), the sensors gave the exact same temperature as the reference, taking into consideration the absence of any surface reflections that might affect the sensors result.

5.3   Final system

Every group's unit consists of a plastic box containing the power bank, breadboard and the microcontroller (ESP32) as shown in figure 5.1. While figure 5.2 displays how a 1.7 meter cable duct made his way through the box's lid with small holes in 4 positions (0.1 m, 0.6 m, 1.1 m, 1.7 m) for the sensors to collect data with no physical barrier.
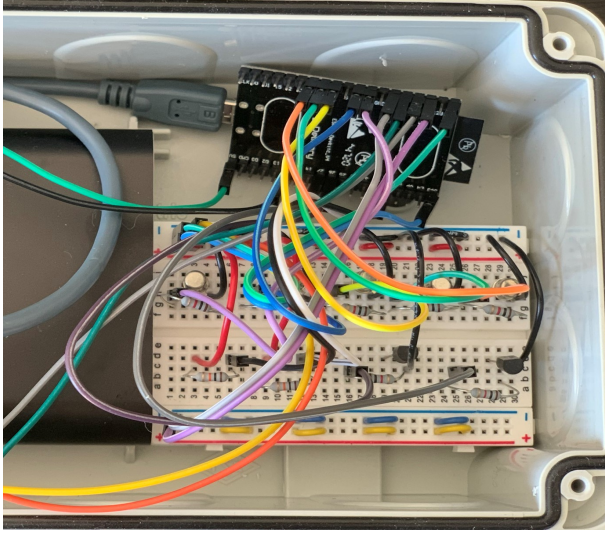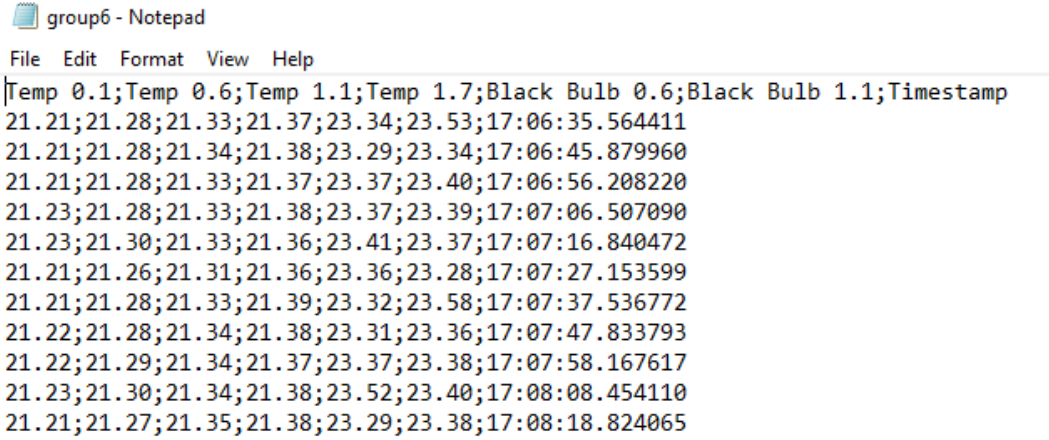
Figure 5.1: Inside a Group's unit box

Figure 5.2: All finished 8 groups

The sensor data in the system is successfully collected from each sensor in each group in a star topology, the collected data is converted into its correct format and then gets sent via ZigBee transmitters alongside a comma "," in ten second intervals. The Raspberry pi is successfully received all the the segments, unpacked the payload and stripped the data from the commas "," and converted the string into the correct corresponding sensor data according the sending order at the sender's MCU. The average was calculated for every parameter every minute, meaning an average calculated every six measurements.

Filtering mechanism was successfully used to remove unreasonable temperature values by eliminating the whole packet sent containing the incorrect data.

The data is parsed into a file for each group after each successful transmission, while the average is calculated every minute and parsed into a file as well. Figure 5.3 and 5.4 show the results of data collected from group 6 with temperature and black bulb temperature and different heights with with time a stamp, 5.3 shows the results with separators for a csv file format, while 5.4 shows the results in an excel file.

Figure 5.3: Collected data in .csv file on notepad

	A	B	C	D	E	F	G
1	Temp 0.1	Temp 0.6	Temp 1.1	Temp 1.7	Black Bulb 0.6	Black Bulb 1.1	Timestamp
2	21.21	21.28	21.33	21.37	23.34	23.53	17:06:35.564411
3	21.21	21.28	21.34	21.38	23.29	23.34	17:06:45.879960
4	21.21	21.28	21.33	21.37	23.37	23.40	17:06:56.208220
5	21.23	21.28	21.33	21.38	23.37	23.39	17:07:06.507090
6	21.23	21.30	21.33	21.36	23.41	23.37	17:07:16.840472
7	21.21	21.26	21.31	21.36	23.36	23.28	17:07:27.153599
8	21.21	21.28	21.33	21.39	23.32	23.58	17:07:37.536772
9	21.22	21.28	21.34	21.38	23.31	23.36	17:07:47.833793
10	21.22	21.29	21.34	21.37	23.37	23.38	17:07:58.167617
11	21.23	21.30	21.34	21.38	23.52	23.40	17:08:08.454110
12	21.21	21.27	21.35	21.38	23.29	23.38	17:08:18.824065
13							

Figure 5.4: Collected data in .csv file on Excel

No information was retrieved from the CAN system of the bus using the OBD-II cable, the command for listening on all data on the CAN interface **./candump** as well as **./cansend** were unable to retrieve any data at all. Resulting a difficulty to calculate the HVAC system output with the absence of the required parameters that the CAN system deliver.

In figure 5.5 and 5.6 can group 1 and 6's placement be seen according to their measuring positions in a MAN´S Lion City 45 bus as in figure 3.1 in Methodology.

Figure 5.5: System's placement in
the bus Position A

Figure 5.6: System's placement in
the bus Position B

6 Analysis and discussion

The measurement system results will be analysed in this chapter, different economical, sustainability aspects and alternative solutions will be discussed.

6.1 Analysing results

The signal strength at the coordinator was good in the different ranges from groups to the coordinator, it can be further enhanced by changing the placement of the coordinator somewhere in the middle of the bus as the Raspberry pi is not connected to the CAN system located nearby the bus driver due to not being able to fetch HVAC system output.

Avoiding the overlapping channels with WiFi by using channels 25 and 26 is a method to avoid interference, although this method might cause a problem if several ZigBee networks are present and are operating 25 and 26 as well, that would cause interference between them, which means that replacing ZigBee for another protocol in these situations for a protocol that operates on another frequency band would be a better solution.

The design of the system was stable, but the cable channel made altering the box very hard, making any changes in the wiring inside the box requires two people, one to hold the cable duct and keep the box stable while the other made modifications to the circuit. With the cable channel the device was taking space because the box could not for example be placed under the seat. Another point was that because of the visibility, groups can draw the attention of passengers and curiosity can cause unwanted prodding with the sensors, if a temperature sensor was held for a couple of seconds the sensor would not give accurate results.
The filtration mechanism was used for the previous cause and gave a good results by avoiding the inaccurate temperature data to evade any false average calculations.

The interval of the acceptable temperature was set by assessing; with the help of the traffic administration consultant and Arriva, the minimum and maximum expected temperature in the traffic buses after taking into account the different seasons, climate situations and outside temperature. The only disadvantage of this mechanism was that the whole packet containing the incorrect value was excluded and not send; carrying the other parameters data as well.

The sensors give stable results in a calm environment, bigger changes occur when the door is opened, the wind sensor gives higher results to the gust of wind, same occurrence happens with the temperature sensors but on a lower scale. The CO2 sensor show higher results when more people are present.

The results displayed in csv files are organized correctly, the only downside is having them placed in sixteen files, if not placed in sixteen files the files would be cluttered due to the sheer amount of data collected.

6.2 Encountered problems

Certain problems appeared during the test processes and affected the progress of the project and the system's functionality. These problems required an alternative way of thinking, especially when it comes to the problem with AHT20 and CCS811 sensors. Transistors were used as switches to turn on and off the sensors one at a time. Both sensors use I2C communication protocol and this led to incorrect readings for the CO2 parameter, which gave only one value of 65535 that means FFFF in hexadecimal.

Another problem was with the power bank's standby mode and was solved by adding resistor between 5V and ground pins in the ESP32 board. The ESP32 microcontroller draws small amount of power for the power bank to detect, thus, when nothing is drawing energy and go directly to deep sleep or standby mode.

Specific pins on the ESP32 microcontroller caused some problems as well. Pins 6-11 are connected to the integrated SPI flash and cannot be used as GPIOs, while the predefined Rx and Tx pins used for UART connection are also used by the board for the debugging process, thus, new UART connection pins were defined on pins 16 and 17. This caused a shortage on the number of unused pins and might cause a future problem in case of adding an extra component to some stacked groups.

6.3   Alternative solutions

This section presents alternative solutions for some problems, these methods are not deemed necessary and incorporating them can be debated.

Using more complicated circuitry caused by the problems that come with CO2 sensors can be avoided by using different CO2 sensors, for example, analog CO2 sensors.

A database is an alternative to csv files but there are aspects to be considered when choosing between a database and a file system. The data required for this measurement system is only viewed by the user, no further interactions are happening with the data inside the program. The program is only performing read and write operations and these operations are faster in a file system. A database is more complex than a file system and data can be indexed and accessed in a better way but in the end it all comes down to requirements and efficiency.

To avoid the power problem encountered in the previous section, a different power bank that does not have a standby mode can be used, the standby feature (in this case a pitfall) is incorporated in newer power banks, older power banks usually do not have standby mode.

6.4   Future developments

The ZigBee modules have 2.00 mm pins unlike the default size of 2.54 mm used for electronics, this means that female cables could not be used due to the distance and wires had to be soldered on the pins. Soldered wires can eventually break when used extensively and are not pleasant to work with when doing many adjustments and configurations on the modules, an adapter with 2 mm headers can bought for this module and gives access to all the pins available on the device, this adapter makes using and configuring the modules easy and flexible that it can be deemed almost essential.

Filtering mechanism can be developed to not exclude the entire packet holding all the other correct values. Another way is to implement a mechanism on the receiver side and program it to exclude only the wrong temperature value and replace it with blank space beside decreasing the measurements count so it will not affect the average calculation at that intended time.

The design can be altered for ease of use, key suggestions that can be offered are replacing the cable duct for an electricity hiding rubber cable. The rubber cable would offer flexibility for positioning and the box can be placed under the seats, while it still draws attention it would be easier to alter and the whole system does not have to be visible.

6.5   Economical and sustainable aspects

The most significant economic benefit is reducing the usage and the power of the HVAC system by adjusting the temperature to suit the outside climate and the passengers' thermal comfort by taking in consideration the season and the amount of clothes they are wearing.
Modifying the system will help reduce the fuel usage in normal buses and electricity consumption in the fully electric buses in the future. This can be observed by the traffic administration who are leading this project [29] in the upcoming stages by reading the fuel consumption and AC compressor's output data from a third party device called Green Box that can be found in the busses nowadays.

This system depends on power banks to operate; which is impractical from a sustainable point of view, nine power banks were used and they all need to be charged using electricity. Despite the fact that all the groups do not consume a huge amount of energy, due to the low power used components and that was proved by many tests done under the project's period. A fully charged power bank can run a whole group for more than eight hours.
On the other hand, comparing the market prices of the used reference (Nordtec Testo instrument) with the system build in this project, shows a clear difference as this measurement system as whole costs approximately the equivalent of three Testo 435 devices that can only used in three positions. The other advantage of this system is that it will be used to collect results for a small number of tests in the upcoming year. However, these results will serve a good cause in saving energy.

7 Conclusion

It can be concluded that the system made was able to collect accurate data from sensor groups in different positions and transmit it; with a sampling frequency of ten seconds as previously required by the traffic administration, using a WPAN technology. The sent data is divided and stored in a file system on the receiver central unit.

Theoretical study and the evaluation of measurement systems constructed formerly led to the chosen communication protocol and the right sensor technologies to create a wireless measurement system suitable for a traffic bus environment.

Literature studies were made to determine the required parameters to calculate the HVAC system output effect. These parameters can be obtained using CAN system which the system did not manage to collect due to lack of support and tests because of the current COVID-19 situation as previously stated. The company owning the bus has a previously installed box that collects CAN system information and display it as easy-to-read data. It is possible that this box prevents any external devices to collect the required data, but there was no expert in this field to confirm nor deny this theory.

Cloud service was not used because it would not provide any direct advantages to the system as the collected parameters will not be reviewed and controlled interactively, instead, it will be studied later by the traffic administration.

The chosen sensors gave accurate results and were successfully implemented in this system to meet their purpose, while ZigBee WPAN protocol was also preferred after a study made on different WPAN technologies which also served its purpose without packet losses.

The system constructed in this thesis can be used by the traffic administration in their upcoming work process to collect climate related data.

Following different criteria and studies, other solutions can be found to the construct of a wireless measurement system. Some developments can also be made to the design of the current system, to make it more flexible under transport and to be able to install it in more precise positions to collect data.

References

[1] Steve Corrigan. "Introduction to the Controller Area Network (CAN)". In: *Texas Instruments* 1 (2002), pp. 2–6.

[2] Sinem Coleri Ergen. "ZigBee/IEEE 802.15.4 summary". In: *University of California, Berkley* (2004), pp. 2–5.

[3] Gianluigi Ferrari et al. "Wireless Sensor Networks: Performance Analysis in Indoor Scenarios". In: *EURASIP Journal on Wireless Communications and Networking* 2007 (Dec. 2007), pp. 41–41. DOI: 10.1155/2007/81864.

[4] Ata Elahi and Adam Gschwender. *ZigBee Wireless Sensor and Control Network*. Pearson Education, 2009. ISBN: 978-0137134854.

[5] Noël Djongyang, René Tchinda, and Donatien Njomo. "Thermal comfort: A review paper". In: *Renewable and sustainable energy reviews* 14.9 (2010), pp. 2626–2640.

[6] Jean-Philippe Vasseur and Adam Dunkels. "Chapter 19 - Non-IP Smart Object Technologies". In: *Interconnecting Smart Objects with IP*. Ed. by Jean-Philippe Vasseur and Adam Dunkels. Boston: Morgan Kaufmann, 2010, pp. 295–302. ISBN: 978-0-12-375165-2. DOI: https://doi.org/10.1016/B978-0-12-375165-2.00019-3. URL: https://www.sciencedirect.com/science/article/pii/B9780123751652000193.

[7] L. Thoroe H. Hilpert and M. Schumann. "Real-Time Data Collection for Product Carbon Footprints in Transportation Processes Based on OBD2 and Smartphones". In: (2011), pp. 4–5. DOI: 10.1109/HICSS.2011.356.

[8] C. Akshay et al. "Wireless sensing and control for precision Green house management". In: *2012 Sixth International Conference on Sensing Technology (ICST)*. 2012, pp. 52–56. DOI: 10.1109/ICSensT.2012.6461735.

[9] M. Hammoshi and Awny Sayed. "An analysis for a cluster tree Zigbee topology". In: 64 (June 2014), pp. 674–680.

[10] Qingyuan Zhu et al. "A Computer-Aided Modeling and Measurement System for Environmental Thermal Comfort Sensing". In: *IEEE Transactions on Instrumentation and Measurement* 64.2 (2015), pp. 478–486. DOI: `10.1109/TIM.2014.2345922`.

[11] Shawkat Ali et al. "All-printed humidity sensor based on graphene/methyl-red composite with high sensitivity". In: *Carbon* 105 (2016), pp. 23–32. ISSN: 0008-6223. DOI: `https://doi.org/10.1016/j.carbon.2016.04.013`. URL: `https://www.sciencedirect.com/science/article/pii/S0008622316302743`.

[12] H.H.R. Sherazi et al. "ZigBee's Received Signal Strength and Latency Evaluation under Varying Environments". en. In: *Journal of Computer Networks and Communications* Article ID 9409402 (2016), p. 8. DOI: `10.1155/2016/9409402`. URL: `https://doi.org/10.1155/2016/9409402`.

[13] Obrad Aleksic and Pantelija M Nikolić. "Recent advances in ntc thick film thermistor properties and applications". In: *Facta Universitatis, Series: Electronics and Energetics* 30.3 (2017), pp. 267–284.

[14] Roohalah Hajizadeh et al. "Offering a model for estimating black globe temperature according to meteorological measurements". In: *Meteorological applications* 24.2 (2017), pp. 303–307.

[15] Tatsuya Iwata et al. "CO2 Sensing Characteristics of a La2O3/SnO2 Stacked Structure with Micromachined Hotplates". In: *Sensors* 17.9 (2017). ISSN: 1424-8220. DOI: `10.3390/s17092156`. URL: `https://www.mdpi.com/1424-8220/17/9/2156`.

[16] Shadi Al-Sarawi et al. "Internet of Things (IoT) communication protocols: Review". In: *2017 8th International Conference on Information Technology (ICIT)*. 2017, pp. 685–690. DOI: `10.1109/ICITECH.2017.8079928`.

[17] Xiao Wang and Kun Yang. "A Real-Life Experimental Investigation of Cross Interference between WiFi and ZigBee in Indoor Environment". In: *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 2017, pp. 598–603. DOI: `10.1109/iThings-GreenCom-CPSCom-SmartData.2017.94`.

[18] Concettina Marino et al. "Outdoor Mean Radiant Temperature Estimation: Is the Black-Globe Thermometer Method a Feasible Course of Action?" In: *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I CPS Europe)*. 2018, pp. 1–7. DOI: `10.1109/EEEIC.2018.8493714`.

[19] Shaimaa Seyam. "HVAC System". In: InTech Open, 2018. Chap. 4.

[20] *CCS811 Datasheet*. Version 1.06. ams AG. Feb. 2019.

[21] Zhen Li et al. "Parametric Simulation of 'Hot-wire'-based Gas Velocity Sensor Using Optical Fiber". In: *2019 Photonics Electromagnetics Research Symposium - Fall (PIERS - Fall)*. 2019, pp. 735–740. DOI: `10.1109/PIERS-Fall48861.2019.9021417`.

[22] Taochuang Zhao et al. "A self-powered counter/timer based on a clock pointer-like frequency-tunable triboelectric nanogenerator for wind speed detecting". In: *Nano Energy* 65 (2019), p. 104025. ISSN: 2211-2855. DOI: `https://doi.org/10.1016/j.nanoen.2019.104025`. URL: `https://www.sciencedirect.com/science/article/pii/S2211285519307323`.

[23] *AHT20 Product manuals*. Version 1.1. guangzhou aosong electronics co. ltd. Apr. 2020.

[24] *DIGI XBEE S2C 802.15.4 RF MODULES Datasheet*. Digi. 2020.

[25] Beom Jin Chung Dong Sik Kim and Young Mo Chung. "Analysis of AMI Communication Methods in Various Field Environments". en. In: *energies MDPI* (2020). URL: `https://www.mdpi.com/1996-1073/13/19/5185/pdf`.

[26] Ritesh Ghimire. "Study on the Link Sensitivity of ZigBee". en. In: *Metropolia University of Applied Sciences* (2020). URL: `https://core.ac.uk/download/pdf/323461539.pdf`.

[27] Jia-jia Jiang et al. "An accurate ultrasonic wind speed and direction measuring method by combining time-difference and phase-difference measurement using coded pulses combination". In: *Applied Acoustics* 159 (2020), p. 107093. ISSN: 0003-682X. DOI: `https://doi.org/10.1016/j.apacoust.2019.107093`. URL: `https://www.sciencedirect.com/science/article/pii/S0003682X19309880`.

[28] Dushyant Kumar Singh, Himani Jerath, and P. Raja. "Low Cost IoT Enabled Weather Station". In: *2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM)*. 2020, pp. 31–37. DOI: `10.1109/ICCAKM46823.2020.9051454`.

[29] UITP. "MEASUREMENT: THERMAL COMFORT OF PASSENGERS". In: *UITP Bus* (2020). URL: `https://www.uitp.org`.

[30] Francesco Cigarini et al. "Modeling and Experimental Investigation of Thermal Comfort and Energy Consumption in a Battery Electric Bus". In: *World Electric Vehicle Journal* 12 (Jan. 2021), p. 7. DOI: `10.3390/wevj12010007`.

[31] *ESP32 WROOM 32E ESP32 WROOM 32UE Datasheet*. Version 1.2. Espressif Systems. 2021.

[32] *Black Bulb Temperature Sensor Datasheet*. Titan Products.

[33] Bluetooth. *FHSS*. URL: `https://www.bluetooth.com/blog/how-bluetooth-technology-uses-adaptive-frequency-hopping-to-overcome-packet-interference/`.

[34] *Bluetooth radio versions*. URL: `https://www.bluetooth.com/learn-about-bluetooth/radio-versions/`.

[35] *Device types*. URL: `https://www.digi.com/resources/documentation/Digidocs/90002002/Concepts/c_device_types.htm?TocPath=Zigbee%20networks%7CZigbee%20networking%20concepts%7C_____1`. (accessed: 20.03.2021).

[36] Digi. *API mode in detail*. URL: `https://www.digi.com/resources/documentation/Digidocs/90001942-13/concepts/c_api_mode_detailed.htm?TocPath=XBee%20API%20mode%7C_____1`. (accessed: 03.5.2021).

[37] Digi. *API mode in detail*. URL: `https://www.digi.com/resources/documentation/Digidocs/90001942-13/concepts/c_xbee_comparing_at_api_modes.htm?TocPath=How%20XBee%20devices%20work%7CSerial%20communication%7C_____2`. (accessed: 03.5.2021).

[38] Digi. *APIframe*. URL: `https://www.digi.com/resources/documentation/Digidocs/90001942-13/concepts/c_api_frame_structure.htm?tocpath=XBee%20API%20mode%7C_____2`. (accessed: 03.5.2021).

[39] Digi. *Zigbee stack layers*. URL: `https://www.digi.com/resources/documentation/Digidocs/90002002/Content/Reference/r_zb_stack.html`. (accessed: 28.2.2021).

[40] Guchen. *Bock FK40 Vehicle Compressor for Bus Air-Conditioning*. URL: `https://www.guchen.com/compressor/bock-fk40-series-vehicle-compressor.html`. (accessed: 25.03.2021).

[41] Jonas Olsson. "6LoWPAN demystified". In: ().

[42] Valeo. *Rooftop Unit REVO*. URL: `https://www.valeo-thermalbus.com/eu_en/Products/Air-Conditioning/Buses-over-12-m/REVO`. (accessed: 23.03.2021).

[43] Bock VAP. *Bock stationary applications*. URL: `https://vap.bock.de/`. (accessed: 10.04.2021).

[44]   *Wind Sensor Rev. C Datasheet.* Modern Device.

[45]   *Zigbee alliance Adopter Member Agreement*, p. 2. URL: `https://zigbeealliance.org/wp-content/uploads/2020/05/Zigbee-Alliance-Adopter-Member-Agreement.pdf`.
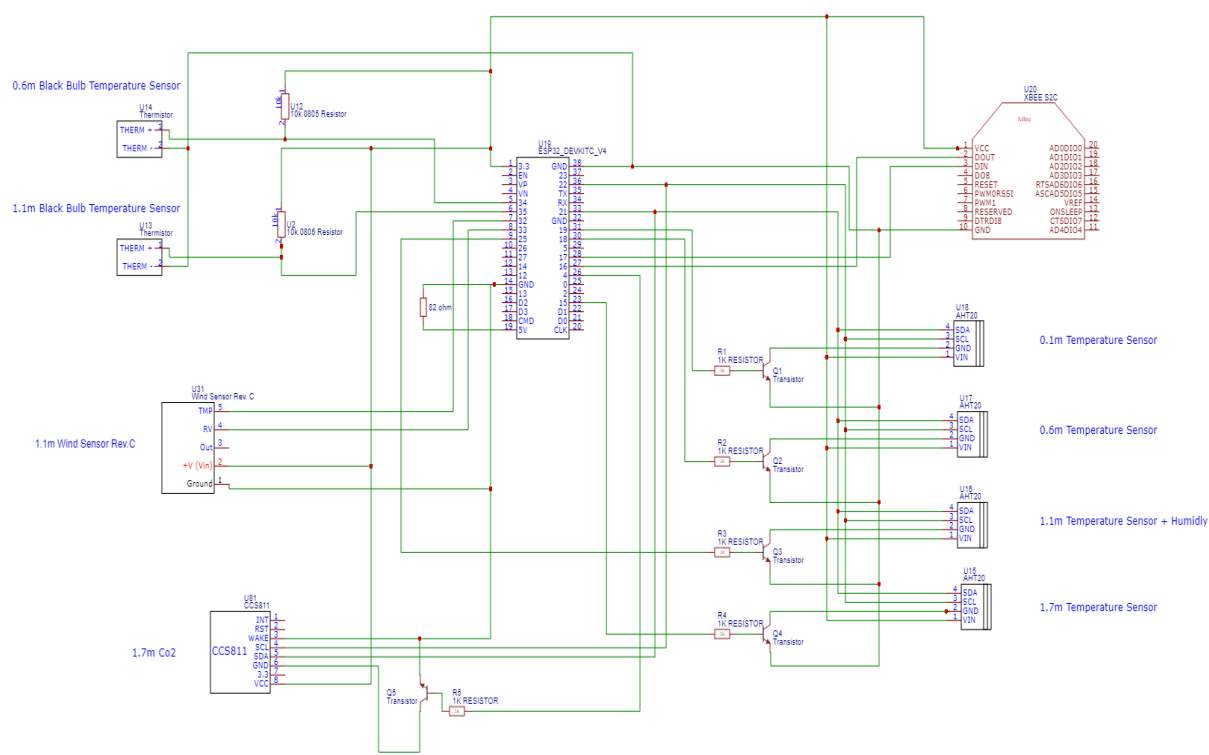
Appendix A

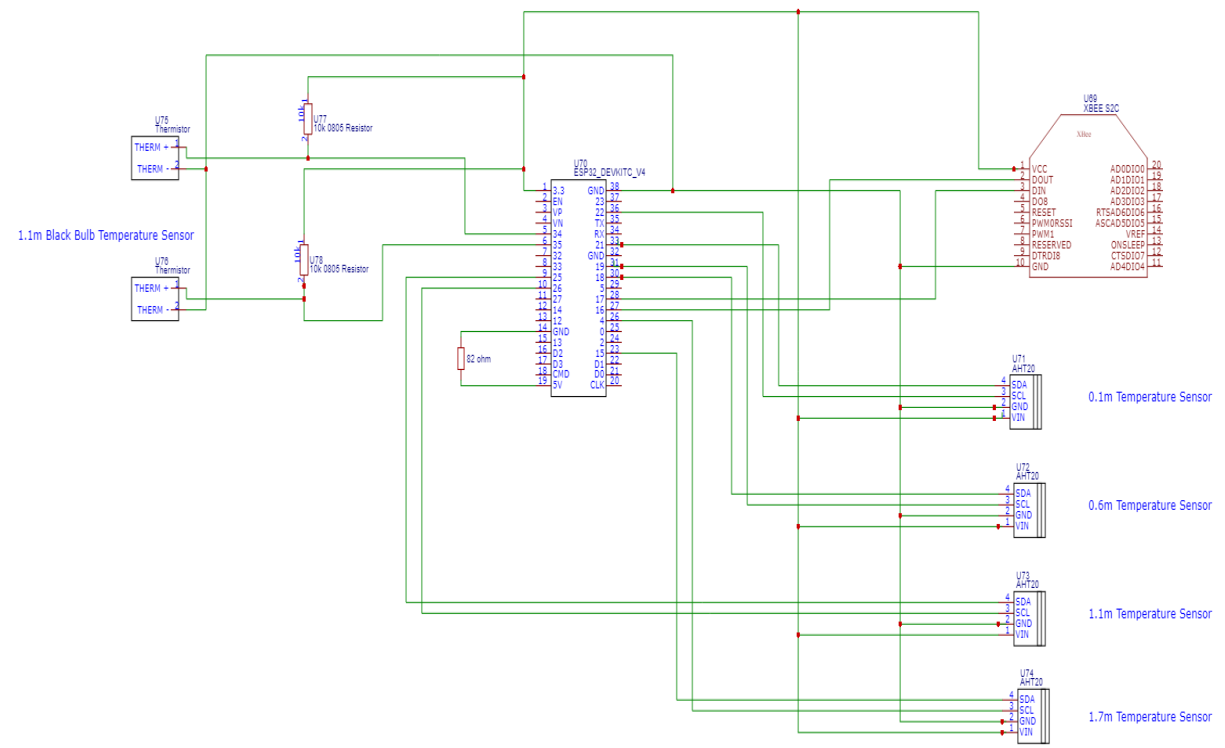## Wiring Schematics

Figure 1: Schematic Design for Group 1 and 7
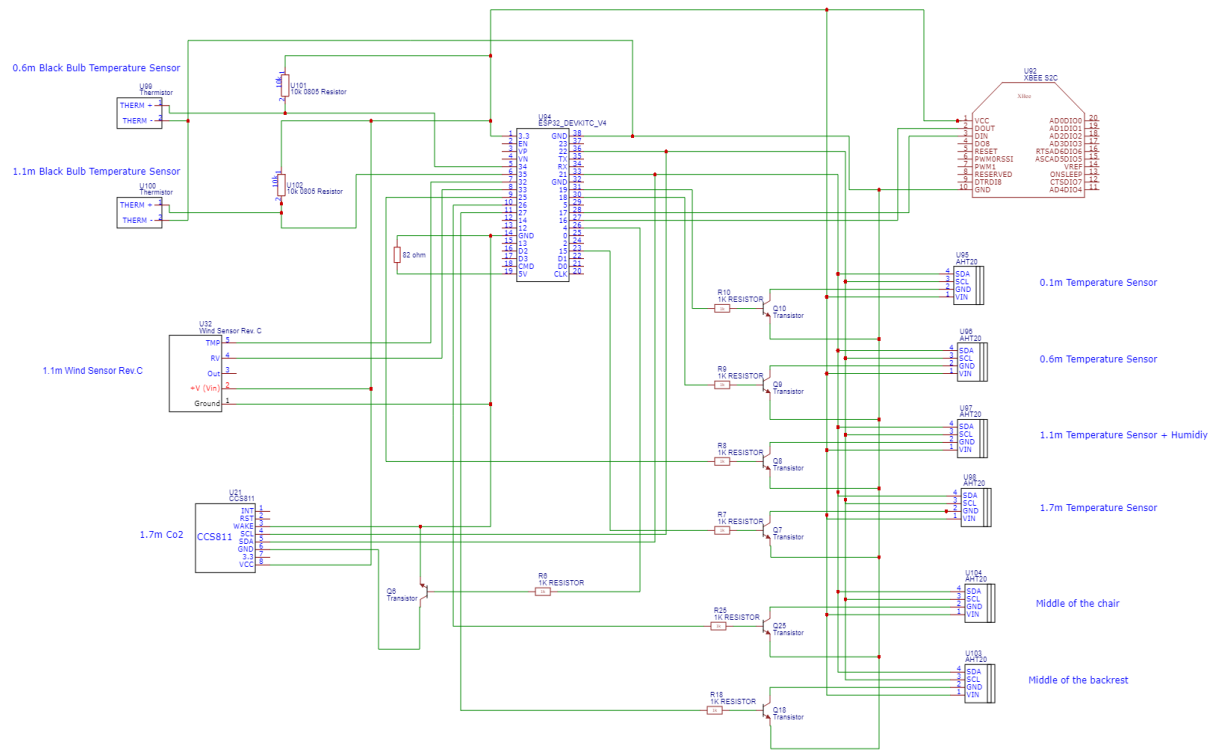
Figure 2: Schematic Design for Group 2, 3, 6 and 7

Figure 3: Schematic Design for Group 4

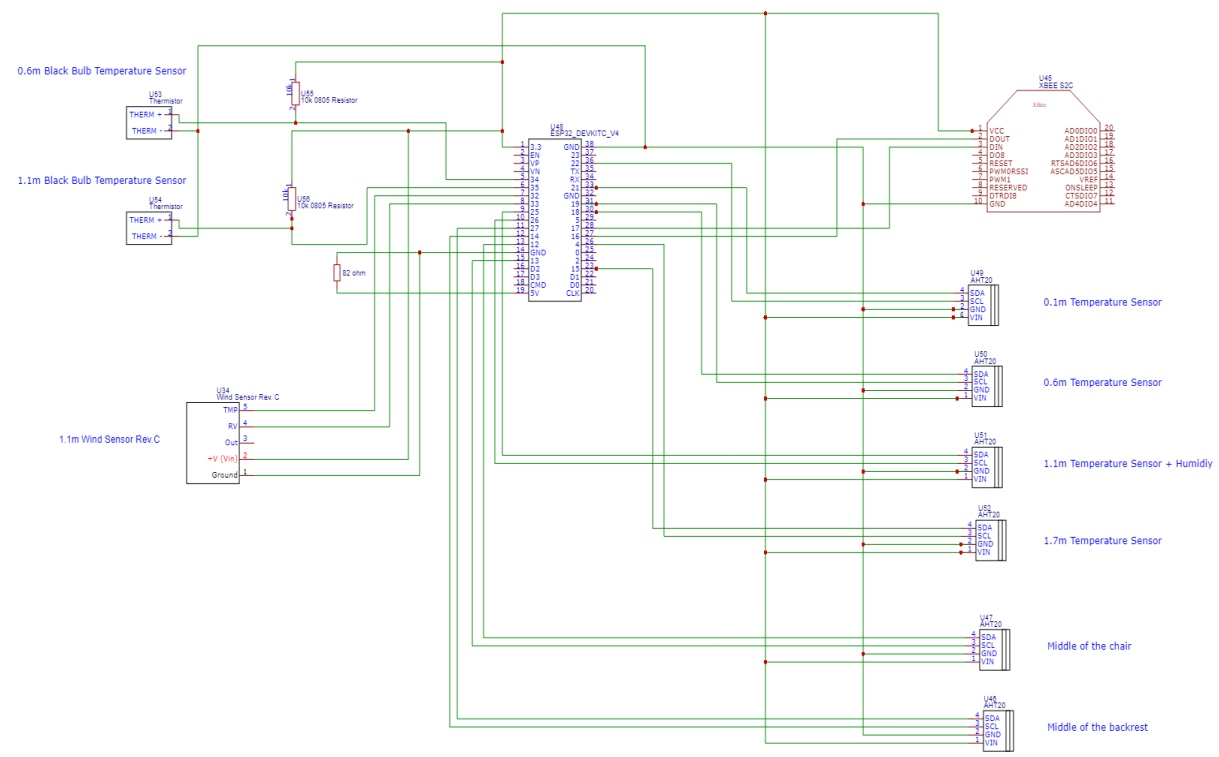Figure 4: Schematic Design for Group 5