OCTOBER 2020 M.Sc. in Electrical and Electronics Engineering MIDHAT HUSSEIN IBRAHIM

REPUBLIC OF TURKEY
GAZIANTEP UNIVERSITY
GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES

SUPERVISORY MONITORING AND CONTROL OF TEMPERATURE AND HUMIDITY USING A WIRELESS SENSOR NETWORK

M.Sc. THESIS
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

BY
MIDHAT HUSSEIN IBRAHIM
OCTOBER 2020

SUPERVISORY MONITORING AND CONTROL OF TEMPERATURE AND HUMIDITY USING A WIRELESS SENSOR NETWORK

M.Sc. Thesis

in

Electrical and Electronics Engineering

Gaziantep University

Supervisor

Assoc. Prof. Dr. Tolgay KARA

by

Midhat Hussein IBRAHIM

October 2020

© 2020 [Midhat Hussein IBRAHIM]

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Midhat Hussein IBRAHIM

ABSTRACT

SUPERVISORY MONITORING AND CONTROL OF TEMPERATURE AND HUMIDITY USING A WIRELESS SENSOR NETWORK

IBRAHIM, Midhat Hussein
M.Sc. in Electrical and Electronics Engineering
Supervisor: Assoc. Prof. Dr. Tolgay KARA
October 2020
61 pages

The wireless sensor networks (WSNs) are contemporary techniques that can provide several options to users by controlling and monitoring devices according to reception data from numerous sensors. The main goal of this research is to implement and test supervisory control of temperature and humidity values in a WSN configuration. This thesis presents a simulation model and test results of the WSN and control system in computer environment as well as the experimental results and control implementation in real time. The experimental system consists of a DHT11 sensor, ATmega-328 microcontroller and software for control of the variables. In this research ZigBee protocol is preferred for its suitability for long-distance and speed to transmit information with good performance to implement WSN. A Proportional Integral Derivative (PID) control law is designed and tuned for temperature and humidity control loops in the WSN. Performance of the designed WSN and control system is tested via computer simulations and real time tests, and results reveal satisfactory performance proving applicability of the proposed system in real applications.

Keywords: Wireless Sensor Network (WSN), PID Controller, ZigBee, Temperature Control, Humidity Control.

ÖZET

KABLOSUZ SENSÖR AĞI KULLANILARAK NEM VE SICAKLIĞIN DENETLEYİCİ KONTROLÜ

IBRAHIM, Midhat Hussein
Yüksek Lisans Tezi, Elektrik-Elektronik Mühendisligi
Danışman: Doç. Dr. Tolgay KARA
Ekim 2020
61 sayfa

Kablosuz algılayıcı ağları (WSN'ler), çok sayıda algılayıcıdan gelen alım verilerine göre cihazları kontrol ederek ve izleyerek kullanıcılara çeşitli seçenekler sunabilen çağdaş tekniklerdir. Bu araştırmanın ana amacı sıcaklık ve nem değerlerinin gözetimsel denetimini WSN yapısı ile uygulamak ve test etmektir. Bu tezde WSN ve denetim sisteminin bilgisayar benzetim modeli ve test sonuçlarının yanı sıra deenysel sonuçlar ve denetim uygulamasının gerçek zamanlı sonuçları sunulmuştur. Deneysel sistem DHT11 algılayıcısı, Atmega-328 mikrodenetleyici ve değişkenlerin denetimi için yazılımdan oluşmaktadır. Bu çalışmada WSN uygulaması için uygun olan uzak mesafe ve iletim hızı başarımları nedeniyle ZigBee protokolü tercih edilmiştir. Oransal İntegral Türevsel (PID) denetim kuralı tasarlanmış ve WSN'deki sıcaklık ve nem denetimi çevrimleri için ayarlanmıştır. Tasarlanan WSN ve denetim sistemi, bilgisayar benzetimi ve gerçek zamanlı testlerle test edilmiş ve sonuçların, önerilen sistemin gerçek uygulamalara uygunluğunu kanıtlayan tatmin edici başarım ortaya koyduğu gösterilmiştir.

Anahtar Kelimeler: Kablosuz Algılayıcı Ağı (WSN), PID Denetleyici, ZigBee, Sıcaklık Denetimi, Nem Denetimi.

ACKNOWLEDGEMENTS

In the name of Allah, the Merciful. I would like to start by thanking God for giving me this opportunity to fulfil my dream of getting a master's degree. I would like to express my sincere gratitude to my supervisor, Assoc. Prof. Dr. Tolgay Kara, and to express my gratitude to him for allowing me to obtain a master's degree under his supervision and providing me with ongoing support during my research. His tremendous knowledge and his constant endeavour to keep me on track and feel satisfied with the research was very helpful during the time of research and writing this letter.

And I wish to extend my sincere thanks to Prof. Dr Ergun ERÇELEBİ, Head of our Department, for approving our project work with great interest.

I couldn't forget to thank my teachers throughout my stay in Gaziantep, Prof. Dr. Arif NACAROĞLU, Asst. Prof. Dr. Serkan ÖZBAY, Prof. Dr. Suvac UÇKUN.

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

CHAPTER ONE
INTRODUCTION

1.1 General

Wireless networks sensors (WNSs) are becoming one of the crucial aspects in numerous scientific and industrial fields, thanks to their advantages of simple design with high flexibility, easy installation and low cost [1]. An 802.15.4 IEEE communication standard was developed specifically for low-rate wireless personal area networks (LR-WPANs) with a focus on WSNs [2]. A Wireless Sensor Network (WSN) consists of a large number of smart sensors that enable them to sense various natural or physical phenomena as desired, and there have been great developments in the field of sensors, as they can wirelessly communicate between them and the base station. These sensors measure the factors to be measured and collected from the target area and send them wirelessly to the monitoring station where those factors such as temperature are observed. [1][2]. Because of the challenges that impose the environment on life or the industrial sector, you must find technology that helps or overcome these imposed challenges. Therefore, there is a need to invent wireless sensor networks that help in obtaining data in real-time for the concerned environment. These sensors must be linked in a single network to facilitate the process of data collection and placed in the base station. In which the received data is analyzed and error is determined. WSN can be used as a platform to collect data and study the behaviour of environmental data [3]. In the recent period, the great interest in WSNs has increased due to their evidence of importance in scientific and practical applications [6] [7]. WSNs consist of a group of sensors in nodes linked to actuators and a data transmitter and receivers such as XBee used in our research to perform distributed sensing and representation tasks [8]. The network collects the information and it is communicated by the computer system network environment which is called the base station.

The main station takes appropriate reactions and decisions according to the information that the network has collected and sent to the station and then the operators work according to the directions to amend and take appropriate measures for the environment [7]. This process enables sensitivity to and control of the environmen Besides, there are many places, applications and situations that require the use of WSNs, such as in monitoring the environment, or monitoring production and quality, in addition to more of those problems in which it is preferable to use wireless networks [8]. For this reason, WSNs have been used that are more suitable for long distances, less costly and more efficient than using wires, and good reliability in collecting data from the environment, especially in hard to reach places and controlled variables like temperature and humidity. As stated before, the primary purposes of WSNs are to be able to collect and process information with reasonable accuracy and high effectiveness in an independent automated manner or by the user. In other words, it provides a strong dual form between distribution and computer as well as communication. As mentioned above, the importance of these networks and the increase in the interest of researchers and companies interested in advancement and technology are highlighted. However, not everything looks good with this technology [3]. The importance of WSN is increasing rapidly and noticeably due to its ease, low relative cost, simplicity of use, and the possibility of placing it in places far from the user, which increases its importance in many important applications, especially military, health and agriculture monitoring. Therefore, this work proposed the combination between the XBee and Microchip ATmega328P microcontroller to design a smart WSN node. The XBee Series 2 with ZigBee protocols are used as a wireless transmitter module. The XBee modules are based on the IEEE802.15.4 standard, which works as a wireless personal area network (WPAN) communication bridge. The XBee Series 2 modules are programmed in application programming interface (API) mode with escaping to form a star topography WPAN with one ZigBee Coordinator node and multiple ZigBee end devices (ED) nodes. Each ZigBee ED node consists of an XBee Series 2 radio frequency (RF) communication module loaded with End Device firmware via XCTU, mounted on an Arduino Uno v3 with atmega-328 microcontroller via a proprietary XBee shield. The Arduino Uno serves as a signal processor from two sensors; DHT11 Temperature and Humidity sensor and actuators.

1.2 Thesis Main Purposes

The aims of this thesis are making data collection in WSN further efficient and more convenient to users. There are many studies focused on the controller design of the wireless network sensor. Therefore, in this work the Proportional Integral (PID) controller is used as the system controller for controlling the response of the actuator speed, the desired value of controller independent of the speed value of the actuator. The desired signal is dependent on the feedback signal that is transmitted from the client node and received at the coordinator node.

1.3 Specific Objectives

i. Creating a cloud storage unit to collect data and monitor its diversity and include real-time temperature and humidity measurements.
ii. Building a WSN based on the atmega328 microcontroller and the XBee device within the ZigBee protocol, which provides ease in distributing nodes and executing tasks.
iii. Design and implementation of a WSN with complete control of temperature and humidity both indoors and outdoors.
iv. Availability of information and handling of delays in transferring data from the node to the Coordinator or base station.

1.4 Research Contribution

Wireless sensor network has many applications, it is being a very advanced and renewable system that fits for environmental monitoring. Researchers interested in WSN are starting to use this technology to monitor and develop their small industrial enterprises. In the current search, it has been shown that it can also be used as a universal controller in some environmental conditions. This work designs and builds an XBee and Arduino based WSN as a client to a cloud database server. In current work during development, the Arduino is used as a client node. The Arduino Uno with atmega328 microcontroller was chosen for its ease of programming and its low energy consumption compared to other microcontrollers. The installation of the wireless network also does not take much time.

1.5 Thesis Outline

This thesis is planned as the following organization. In the second chapter, a comprehensive review of WSNs is presented and the applications of WSN is well explained. It also describes the work requirements and tools for a secure WSN, followed by a list of how these tools are connected and how they respond. Chapter 3 provides basic knowledge regarding the Digi's XBee ZigBee product that is used in the implemented design. Additionally, the explanation of the practical way to set up the WSN are given, and also the details of the materials, how they are used, and how they can be monitoring the target environment and controlling the temperature and humidity are explained. Chapter 4 provides a detailed description of the implementation design of the WSN and how sensors system establishes relationships between variables and results. The numerical data appears by curves to indicate the amount of sensitivity and monitor the system efficiency, the discussion of those results and relationships. Lastly, Chapter 5 concludes the thesis.

CHAPTER TWO
LITERATURE REVIEW

2.1 Introduction

In this chapter, the end-to-end components of nodes communication systems are introduced, a brief review of WSN topologies, ZigBee communication protocol and their connection protocols are presented. Finally, Arduino Uno board based on the ATmega328P microcontroller and some digital sensors are discussed.

2.2 Wireless Sensor Networks

A Basically, wireless sensor networks consist of the base station (usually called a "gateway"), and this base station can communicate with many other sensor nodes using wireless communication by radio. The nodes placed in the specified goals sense the variables to be monitored, then send them directly to the gate or the base station. Or, if necessary, other sensor nodes (routers) are used to redirect the information from those nodes to the main station, and then the sent information is presented to the system by communicating with the base station. [4]. WSN has developed significantly and widely in recent years, which has increased its importance to researchers, industries and engineers specialized in creating these networks and the possibility of applying them to military, civil and agricultural facilities. This technology was developed to enable monitoring environmental and industrial problems and obtaining data for those targeted places safely, quickly and remotely [5]. By creating nodes in the targeted places, where these nodes are distinguished by connecting to the basic base wirelessly and creating a global network on sensing and processing data according to the user

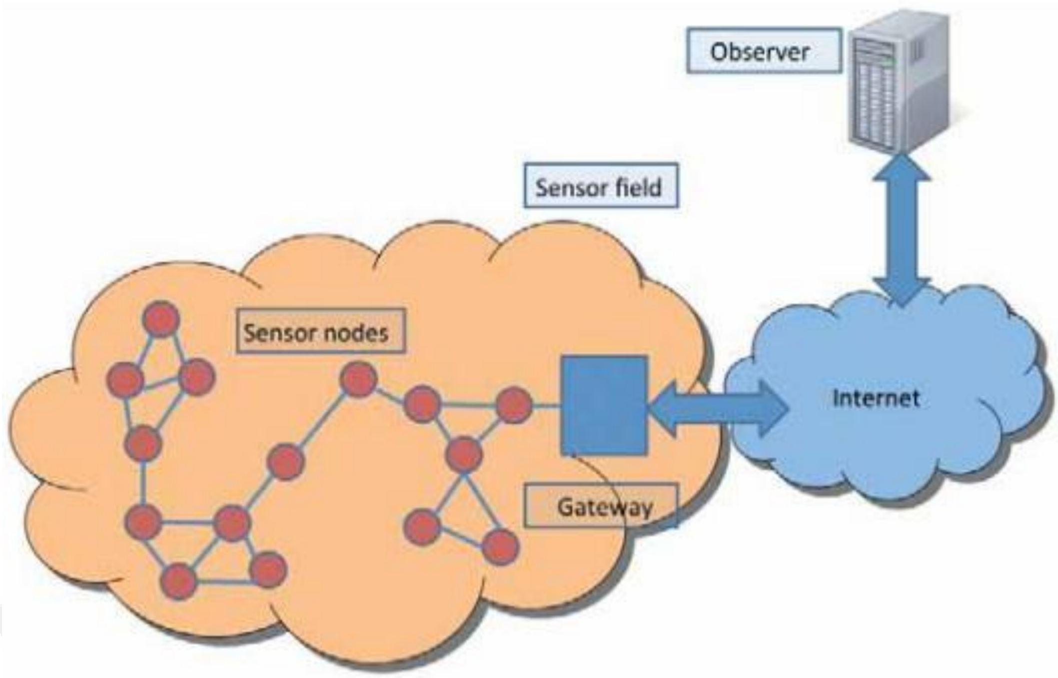

Figure 2.1 Wireless sensor network [5].

WSN has important applications such as remote environmental monitoring, sensing signals such as humidity and temperature [6], and target tracking, especially in last years with the help of sensors that are smaller, cheaper, and intelligent. The technology of WSNs evolves very fast gratitude to the advancement of digital electronic technology and wireless communication technology. WSNs are collected from a large number of sensor nodes that are deployed spatially through an indoor or outdoor environment. These sensor nodes can be used to detect the surrounding conditions such as gas, humidity, light, temperature, sound, and so on. It means that there are various application areas for WSNs [7].

2.3 Wireless Communication

2.3.1 Communication Protocol

in general, we have four standards for Wireless Personal Area Network (WPAN) to data transmission,ZigBee, Wi-Fi, Bluetooth and Ultra Wide Band(UWB). While each standard has specific characteristics used for it. in my research I used ZigBee protocol due based on (IEEE 802.15.4) because specifically develop for WSN, then it has low power for signal transmission, the low rate data and can transmit data for a long-distance [8]. and is very suitable for implement and construction wireless sensor networks. [9].

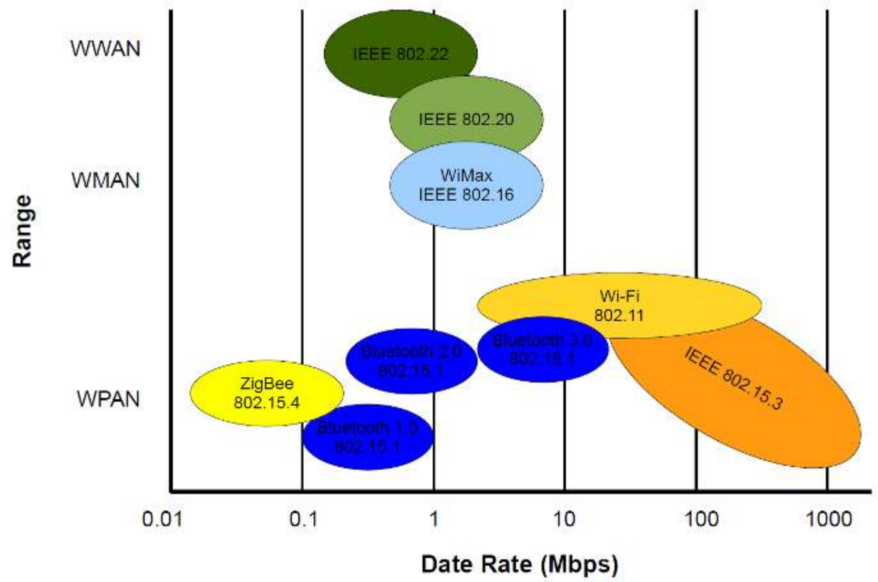

Figure 2.2 Range vs data Rate for various IEEE standards.imagecourtesy of arthur stachowicz [10].

While the 802.15.4 protocols specify the MAC and physical layer, ZigBee protocols handle application layer and above which generally deal with network configuration, security, and application profile [11].

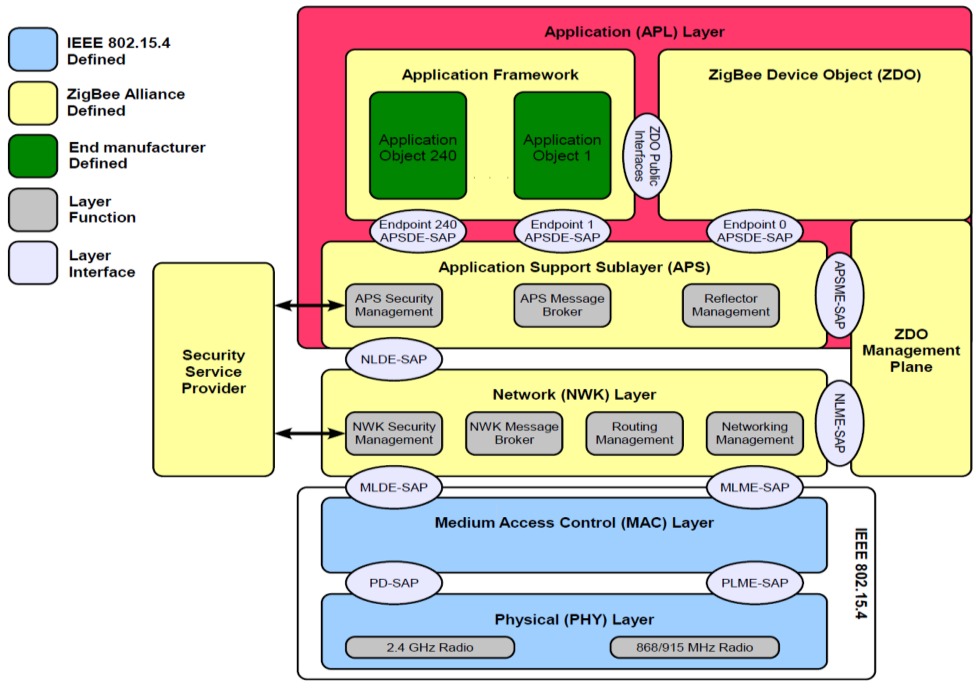

Figure 2.3 Protocol layers for ZigBee stack. Image courtesy of arthur stachowicz [10].

XBee Series 1 and Series 2 modules operate on the 802.15.4 and ZigBee standards respectively with small variations for specific uses. Due to its intended development for low rate sensor networks, XBee series 2 with ZigBee protocols were used as a communication bridge [11].

2.3.2 ZigBee Protocol

The ZigBee stack consists of several layers, including physical (PHY) layer and Medium Access Control (MAC) which is a sub-layer subjected to the IEEE 802.15.4 standard, application support sublayer (APS) layer and ZigBee device object (ZDO) layer. Table 2.1 shows the ZigBee stack configuration file.

Table 2.1 ZigBee layers.

ZigBee Layer	Description
MAC	Manages the process of data transfer between nodes (node to node). The MAC includes services such as transmission retry, acknowledgement management, and collision avoidance technologies (CSMA-CA).
PHY	Physical layer operations(PHY) for the ZigBee its define to receive channel rejection, Output power, number of channels, sensitivity and transmission rate specification. most PHY of ZigBee work at 250 kbps data rate and 2.4GHz ISM.
Network	This layer determines the ability to routing data packets within the radio frequency and specify multiple devices. To transfer data from source to destination
ZDO	Application layer that provides device and service discovery features and advanced network management capabilities.
APS	Application layer that defines various addressing objects including profiles, clusters, and endpoints.

ZigBee utilizes direct-sequence spread spectrum modulation and operates on a fixed channel. The 802.15.4 PHY defines 16 operating channels in the 2.4 GHz frequency band. The graph Figure 2.4 shows ZigBee RF channels, note that normally ZigBee RF channels are counted from 11, up to 26 in the 2.4 GHz band. Because these features found in ZigBee protocol we used it for Communication protocol XBee devices use is the ZigBee protocol. ZigBee protocol is based on IEEE 802.15.4 add some layers. IEEE 802.15.4 certified layers Standard is Physical and MAC, which is the two ones

Important at the moment of choosing the type of network to be implemented. The special layers provided by ZigBee are the Network and Application layer.

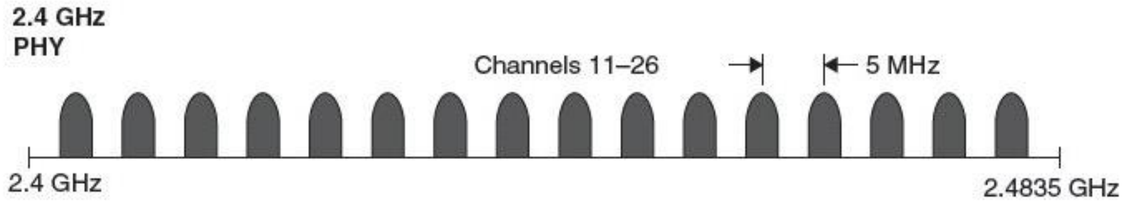

Figure 2.4 ZigBee channels [23].

2.4 WSN Technologies

In addition to the development that has taken place in WSN technology, many changes have occurred in the shape, distribution and organization of these networks. Several technologies have been introduced to connect WSN. Each of these technologies has different conditions, restrictions, and features to be installed. Here are some of the most popular technologies used in WSN [30].

2.4.1 Peer to Peer Technology

This technology is dedicated as it fully supports bar communication between two nodes and transmits information in both directions, and it can communicate with any other node located within the node's contact area [12] (see Figure 2.5).

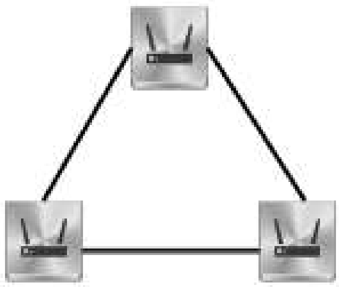

Figure 2.5 Peer-to-Peer technology.

2.4.2 Star Technology

This technology is based on its creation on the formation of a star, which means that there is a central axis through which all the remaining branches are connected, and communication between the branches through its central axis is not allowed. The information is sent from one branch along with the address of the other branch to which the information will be sent (Figure 2.6).

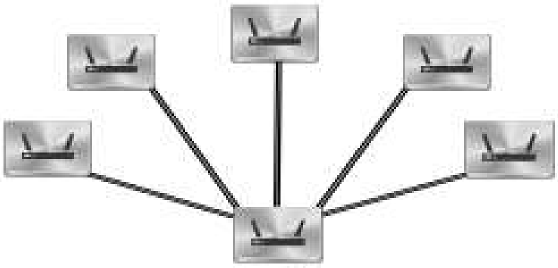

Figure 2.6 Star topology.

The branches that usually represent the nodes are often powered by batteries because they are placed in remote locations as for the main axis, it is connected directly by electricity and placed at the main monitoring base station [4].

2.4.3 **Tree Technology**

This technology is usually called tree technology, and it is a hybrid technique consisting of peer-to-peer technology with star technology linked together. The central hub connects with the root node with its link to the remaining nodes, and with this link it makes this technology look like a tree. When linked to it, one must connect to the central hub and send the signal to the root. So, when one wants to add a new node to such a network, it must be connected to the main hub shown in (Figure 2.7).

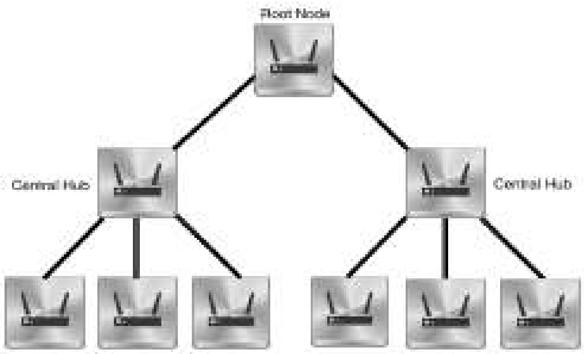

Figure 2.7 Tree technology.

2.4.4 **Mesh Technology**

In mesh technology each node can communicate with each other directly through end to end by hopping without need to communicate the hub or base station, this feature makes mesh grid suitable for configuration sensor networks. Nevertheless, this technology is considered as the most complex and expensive among all (Figure 2.8).

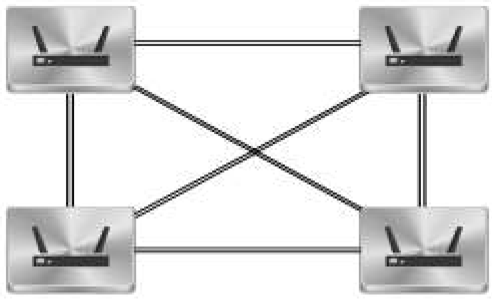

Figure 2.8 Mesh technology.

2.5 Applications of WSNs

The WSNs are often used for monitoring and tracking functions [21]. The tracking function, in which WSN is often used, is for military purposes and searches for violations that occur within the boundaries covered by WSN and can be used for traffic inspection, monitoring road movements, monitoring violations and violations of the base station, among others. Such applications [22]. Besides, it enables increased efficiency and production by monitoring the manufacturing method according to specific commercial specifications. As mentioned above, WSN has great potential in terms of monitoring and control. WSN can be used in the agricultural sector to monitor greenhouses by light intensity, humidity and temperature. It can also be used for military purposes and to monitor the possibility of chemical or biological attacks, which is an essential part of the field of security surveillance. Nowadays, WSN is used in all scientific, industrial and even residential domains, forest fire detection, and vegetation tracking and monitoringWireless sensor networks assist in many military applications such as border control and the risk of infiltration through them, in addition to monitoring ammunition equipment, chemical attack detection and detection of biological and nuclear materials, in addition to security monitoring of the environment and searching for accidents that occur within the target environment for monitoring. Agriculture, as it is used to monitor fertilization, control the amount of irrigation of the crop and monitor its vital state. In addition, it is used in medical applications as it enables the doctor to monitor the patient's condition, monitor his vital activities continuously and accurately, discover emergency situations and warn them, and control the administration of drugs as well. Among the most practical applications of sensor networks are those used in homes for temperature monitoring, intrusion control, lighting control and fire alarms. Also, WSNs are used in many industries and monitor workflow to obtain the best results and detect faults during work [13] [14].in addition to many other applications that are used from these networks [6].

2.5.1 Description of the Climatic Control Problem

The environmental and climatic factors have a significant impact on the growth and production of agricultural crops, as well as the amount of fertilizer and water was given to the plant has a great impact on crop productivity. From this point of view, a greenhouse must be formed that provides ideal climatic and environmental conditions for agriculture, because it will represent a closed environment that can be controlled, from the amount of irrigation and the amount of fertilization needed to obtain the optimum crop yield [15]. Due to environmental and climatic variables, it has become necessary to form networks to monitor climate changes and to create automatic irrigation and fertilization systems in order to enable us to control and control these variables. In view of these requirements for the formation of the greenhouse, it became necessary to establish wireless sensor networks in order to transmit information from different areas within the field to the monitoring centre in order to control the irrigation or fertilization systems or even the intensity of lighting or control that is automatically according to the requirements needed by agriculturalcrops [16] [17].

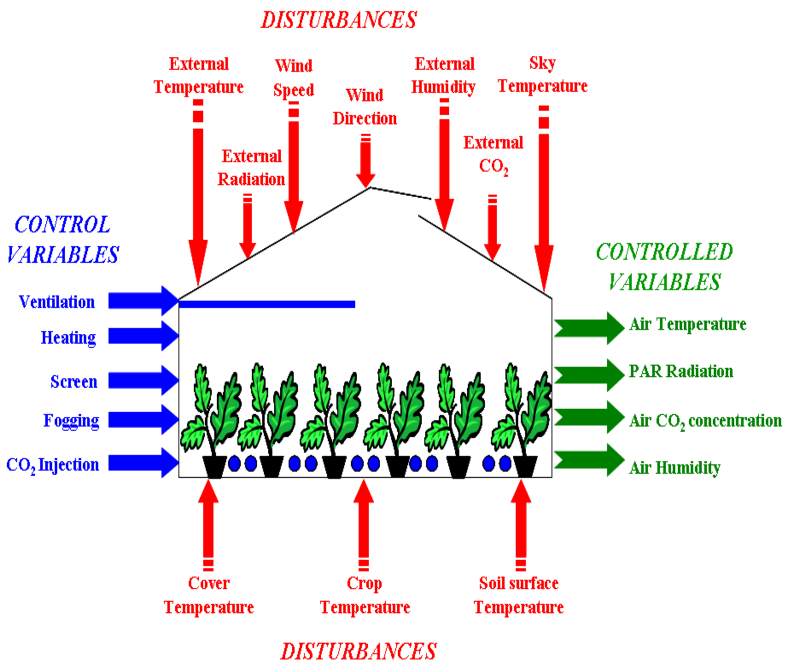

Figure 2.4 Climatic control variables [15].

The dynamic behaviour of the greenhouse microclimate is a combination of physical processes involving energy transfer (radiation and heat) and mass balance (water vapour fluxes and CO_2 concentration) [15]. These processes depend on the outdoor environmental conditions, the structure of the greenhouse, type, and state of the crop, and the effect of the control actuators [18].

2.6 ZigBee- Operation Modes

ZigBee is a protocol that enables XBee devices to operate in various local serial communication modes. The operation and communication of XBee devices with the host is done via the serial interface. XBee devices can operate in two different modes: transparent application (transparent mode) and application programming interface (API mode). and can configure these modes by the XCTU software program.

2.6.1 Application Transparent AT-Mode

In this mode, the radio can receive data and send it at the same time, so it can be known as "transparency". All serial data is sent and received wirelessly within the remote XBee unit destination, all at the same time. And when the other XBee receives data then it sends them over the serial port as it received them. AT-Mode includes restricted functionality, but XBee can be used directly in AT-Mode. It is further detailed on AT-Mode in [19].

2.6.2 Application Programming Interface API-Mode

The API is the second run mode for XBee and an alternative to the AT mode. In the API, the protocol that transfers the data is defined, and the data is delivered in the form of packages called (API frameworks). The API mode gives the user more flexibility to create more comprehensive sensor networks and is more suitable for building a WSN, due to some characteristics such as ease of collecting data from multiple devices, remote device control, or home automation tasks. More details on this operating mode are shown in [20]. The API defines the application, location of sending data, responding to commands, and the status of the module and its receipt from the unit by the serial interface. This feature provides many services.such as

1- XBee module can configure itself.

2- Sending and managing data to various parties.

3- The possibility of rearranging remote units within the network.

4- Displaying the packages for each success and failure operation in sending and receiving data.

5- Distinguishing and identifying packages received from sources.

Based on the Application Programming (AP) parameter value, the radio module can operate in one of two modes: API (AP=1) or API escaped (AP=2) operating mode. Due to the services of API in data management and remote configuration, this operation mode is used as a way for host device communicates with an XBee module in the proposed design of a wireless sensing network.

2.7 **Place of this Work in Literature**

In this thesis, we design and build an XBee and Arduino-based WSN as a cloud database server client. The project uses XBee series 2 together with ZigBee protocols, which are based on the IEEE802.15.4 standard, as the connection bridge for the WPAN. XBee Series 2 modules were programmed to communicate in API mode with an escape to form a WPAN star topography with one ZigBee Coordinator node and multiple ZigBee End devices. Each ZigBee End consists of an XBee Series 2 Radio Frequency (RF) communication unit loaded with the End Device firmware via XCTU, installed on the Arduino Uno v3 via a special XBee shield. The Arduino Uno acts as a signal processor for two local sensors: the DHT11 temperature and humidity sensor and the actuators. And the addition of a control system to the WSN and application kinds of control as like proportional and integral control (PI control). dependent in my work on some literature review as we have seen above how can configure the network and what is preferred to implement as like this network then what are applications for it, and it could be extended for a large network. The nature of the work WSN is the formation of networks of sensors that are placed in different places and are interconnected and communicated wirelessly, and each point that is placed in the specific environment is called a node. Sometimes it is difficult to reach the node, so these nodes are connected wirelessly with the coordinator. Where the coordinator collects data from the nodes spread in the environment and publicizes them to monitor the variables and the target environment. Where an identification number is placed for each transmitter present in the contract, and the authentication process for transferring data between the node coordinator is done in [21]. And two nodes are used, each node containing the XBee device, they are configured within the DiGi platform or XCTU program for developing and configuring XBee devices, as XBees devices are configured to communicate with each other. Through this mechanism, wireless sensor networks are configured, but implementing large networks is much more difficult than configuring only two devices connected because the network identifies each device via its own address for communication with the coordinator. WSN has become very important in the twenty-first century [22]. It has attracted attention and attention in the world through its ability to solve many different problems and in many industrial, military and civil applications, and the most important thing that we will focus on in our research is how to create a network to monitor and control temperature and humidity and apply it in different circumstances and monitor the performance of this network from Transfer data in response to the station's environmental variables and control them via the triggers within the node. Interestingly, however, WSN has many limitations and impediments to its work in monitoring the environment. Some of the main challenges that WSN faces are failure to discover the sensor node and the limited data storage space in the nodes in addition to the storage space and data analysis may encounter problems with the coordinator [23].

CHAPTER THREE
PROPOSED SYSTEM DESIGN

3.1 Introduction

In this chapter, the main parts of the proposed WSN are described, and the input/output signals for client and coordinator nodes are briefly explained. Furthermore, a review on digital sensors that have been used in the developed design is presented. Finally, the simulation response for the Proportional Integral Derivative (PID) controller and gain tuning method is briefly described.

3.2 Types of Nodes

The nodes can be classified into two types of source nodes and sink nodes (gateway). Source nodes are used to collect the information and also work as a routing node. Sink nodes are the nodes gathering the information from all source nodes. They are also called a base station, and they usually use a computer with Windows x86 operation system and connecting by USB v2.0. Sensor nodes can be considered the central part of the WSN. They are responsible for the process of data collection and re-transmission to the base station. The sensing field can be defined as the area to be monitored in which the sensor nodes are located. And in my research it consists of three nodes, one makes it as a coordinator, as for the remaining two, they were installed as end nodes to be placed in the place to be monitored, the network was installed and a mesh topology was used, as this technology allows bi-directional communication between the parts of the network, which enables the process of information transfer and communication between the nodes and nodes with the coordinator.

3.3 Components of Proposed of WSN Material

In this section, the main components of developed WSN are explained. The sensor node or client node consists of Arduino Uno microcontroller, sensors, actuator BLDC-FAN and XBee Series 2 module.

Whilst the Coordinator node is including of XBee-Shield supports XBee ZB module with USB v2.0 and computer as a data logger. The following subsections provide further details on the parts of the proposed WSN.

3.3.1 Sensors

The main job of the sensor is to sense the physical, quantitative and physical factors and convert them into analogue or digital signals depending on the type of sensor. The sensor can be used for more than one purpose as it is used in our research, for temperature and humidity the digital sensor DHT11is used. Distributing sensors or nodes on a large scale, outdoors or indoors, as they are called sensor networks, enables these networks to monitor the environment, machines and fulfil their purposes. These nodes are linked by a wireless connection and collect data from the nodes also wirelessly, allowing ease and possibility of using them in a large scale without much losses, in addition to the simplicity of installation and small size. The process is to control environmental phenomena such as climate in the following manner. The sensor function is used to collect information from physical and environmental features, while the operators must interact with commands to control the situation.

For this reason, wireless sensing networks have been used that are more suitable for long distances, less costly and more efficient than using wires, and good reliability in collecting data from the environment like temperature and humidity, especially in hard to reach places. The primary purpose of forming these networks is to make the monitoring of the required environment or area independently and the process of obtaining and processing information faster, simpler and more flexible, by publishing these nodes in the required places, and that these nodes provide a mixture between consulting, control, communication and computing, so many researchers and companies develop this technology and increase its efficiency due to its importance in the future and the present. The importance of WSNs increases day by day because it provides ease of installation, meagre cost and the possibility of using it even by the general public. It does not include specialists and researchers, and its use provides ease and control from remote places. This can be used in several fields and applications such as military, health and agriculture monitoring.

3.3.2 DHT11 Temperature-Humidity Sensor

DHT11, which is a digital sensor used to sense temperature and humidity, contains inside a humidity sensor and a thermistor to measure the humidity in the air, and it contains three pins coming out of it to be connected with the Arduino, where it sends a digital signal from the data pin to the controller and needs accurate time to be read and obtained. The data is correctly and accurately measured.

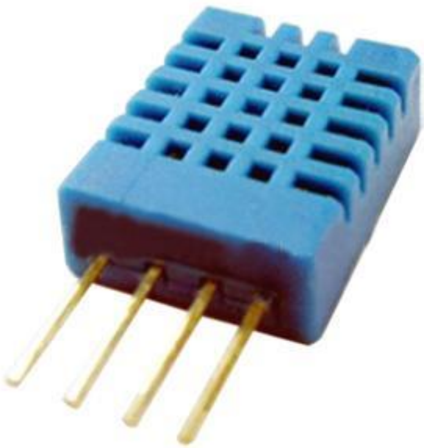

Figure 3.1 DHT11 Sensor.

The sensor is calibrated accurately in the laboratory, and the information is stored in it utilizing a temporary OTP memory, through which the internal signals of the sensor are controlled. It senses a circular area around 20 meters and the dual-sensing feature of humidity and temperature, which makes it suitable for many applications, especially those used in agricultural applications. With the calibration of humidity and temperature, it gives an accurate reading of the surrounding atmosphere in digital form. serial communication or data sensing transfer from DHT 11 sensor to Arduino by digital data and should be writing the special library for DHT 11 in code to make Arduino able to read data coming from a sensor.

Table 3.1 DHT11sensor specification

Item	Measurement Range	Humidity Accuracy	Temperature Accuracy	Resolution	Package	Power supply
DHT11	20-90%RH 0-50 ℃	± 5%RH	± 2℃	1	4 Pin Single Row	5V

3.3.3 ATmega328P Microcontroller

Arduino UNO is a board established with ATmega328P microcontroller developed by Atmel company. The Arduino UNO consists of 14 digital pins and 6 analogue pins starting from A0 to A5. These pins can be work inputs or outputs. Arduino board and uses 6 of the 14 digital pins as pulse width modulation (PWM). It also contains a USB port to facilitate the process of connecting it to the computer, and a power outlet if we want to place it in a location far from the computer. A 9-volt battery can be used, a 16 MHz crystal oscillator, a reset button and the control zeroing, and the Arduino can be connected to the computer by cable [24]. The program is downloaded to it from the computer. The details of specifications for Arduino is provided in Table 3.2.

Figure 3.2 Arduino uno [24].

Table 3.2 Arduino specification [25]

Component	Type/Count
Operating Voltage	5V
Microcontroller	ATmega328
Voltage I\P	6-20V
DC I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
I/P	7-12V Voltage
Clock Speed	16 MHz
EEPROM	1K byte
SRAM	2 K byte.
Flash Memory	32 Kbyte.
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6

In the Arduino, according to the characteristics mentioned in Table 3.2, it works on the voltage of 5 volts as well, and all the 14 digital pins can be used as output and input, and each pin provides a maximum reception of 40 mA and an internal pull resistance ranging from 20-50 kilo-ohms. Special work, meaning that the pins A0 and A1 are used for serial data RX, TX, and usually the pins A3, A4 are used for interrupting. It contains the grounding pins as well as the 5-volt source that is used when connecting the sensor to it. Any digital pins can be used as a 5-volt voltage source.

3.3.4 XBee S2C Serial and XCTU Communication

XCTU is open-source software for configuring ZigBee devices to a selected firmware and network protocol. The software allows ZigBee devices to transmit or receive data either in 16 bit or 64 bit. The formats of transfer data are in hexadecimal. The network area configuration handles the main configuration of the XBee to ensure either one-way or full-duplex communication. The firmware update was used to choose the XBee type and communication type while the console serial ports were used to construct packets, transmit packets and view the packets' communication on both devices. The XBee Serial 2 (S2C) has data broadcast rates, ranging from 20 to 900 kbps with a radio frequency bands of 2.4 GHz, 900 MHz and 868 MHz Two XBee S2s were configured using XCTU software to enable data communication from RX source to TX destination.

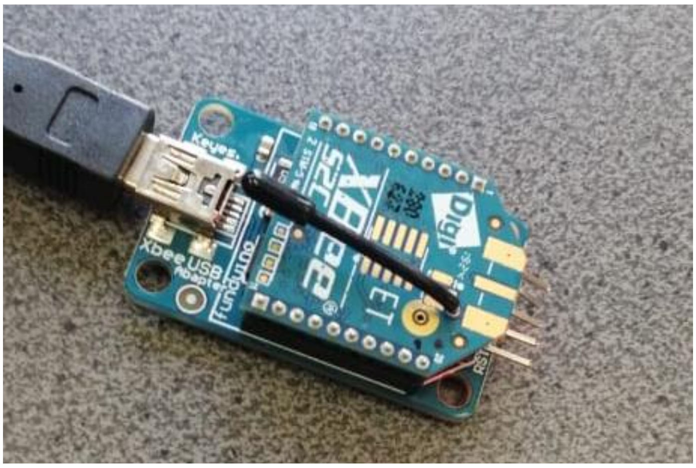

Figure 3.3 XBee S2C adapter kit.

3.3.5 Actuator Brushless DC motor (BLDCM)

The foremost characteristic of Dc fans is supplied by a fixed dc voltage source or battery, the standardised design voltages for DC-fans are 5V,12V, 24V, and 48V.

Nowadays, the popular brushed Dc motors have been replaced by brushless dc (BLDC) motors which are mostly utilised for dc fans. In contrast to the conventional permanent magnetic dc motors that including brushes and commutator, the BLDCs did not need any accessories to connect between rotor and Dc source. The stator supply by dc source whereas the rotor includes four magnets installed on sides in a cross pattern to create a rotor of the motor. The BLDC -motors are grouped according to rotor control technique into; sensor BLDC motors and sensor-less BLDC motors the name since of that motors included a hall effect sensor used for detecting the position of the rotor magnet. Whereas the sensor-less technique of BLDC motors, the rotor back EMF is employed to detect the position of rotor magnetic. The sensor-less BLDC-motors are extensively used small BLDC fans. The picture in Figure 3.4 shows the BLDC fans with 12 V voltage standard.

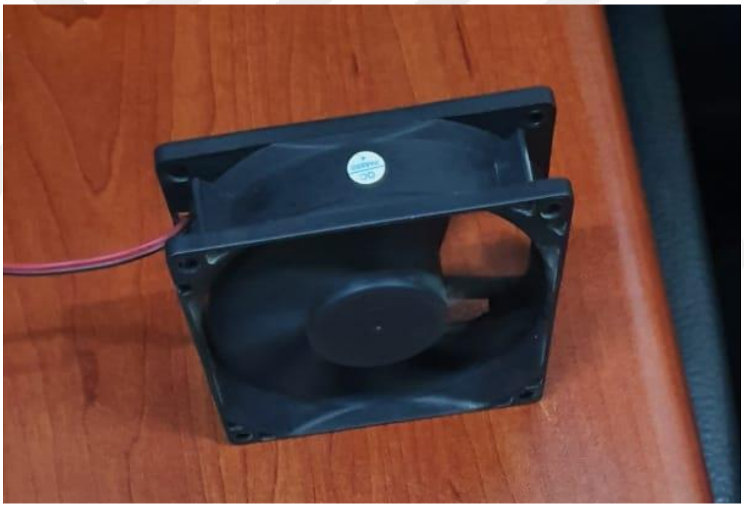

Figure 3.4 BLDC fan 12V.

3.3.6 Breadboard

In our research, the breadboard was used as a construction base in developing electronic circuits for DHT11 temperature-humidity sensor and fan. The breadboard functions as a host for jumper cable connections. The breadboard used in this research was solderless, made of plastic and perforated with several holes. Small tin-plated bronze clips are located under the holes, which provided contact points to attach electronic pieces to create a circuit. The breadboard was used to extend all the functions of the circuit diagram to the Arduino using the jumper cables. All the three pins in the DHT11 temperature and humidity sensor are connected to the Arduino UNO's similar pin functions. The GVS in the DHT11 sensor were bridged and extended the function to the Arduino board [26]. We have implemented a circuit on a breadboard to connect the DC fan with Arduino for control of speed as we have shown in Figure 3.5.

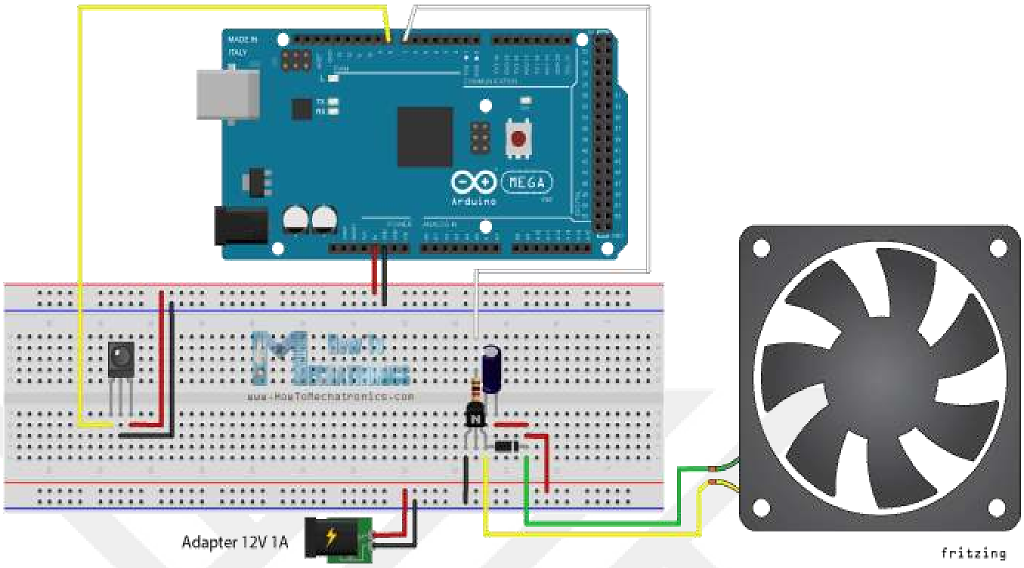

Figure 3.5 Circuit arduino and dc fan with breadboard.

3.3.7 Jumper Cables/ Wire

The jumper cables are components used with the breadboard to generate a functioning circuit. The exposed ends of the cable/wires were used to connect to the breadboard. And similarly, three jumper cables were extended to the Arduino for connection to make the full circuit and enable the DC Fan and DHT11 Sensor to transmit data.

3.4 Arduino IDE

As we know that the Arduino is an open-source, that is, it can be programmed according to the designer's desire to carry out the tasks written within the code (usually each electrical panel is referred to as a microcontroller for modification and programming). Integrated Development Environment (IDE) provides an Arduino development environment, which can be in within the IDE, the development and writing of programs and highlighting the functions for the Arduino is supported and updated by the Arduino company to match the processor on the Arduino board and include the IDE system or page in the Arduino program on the computer the ability to write the code inside it and upload it to the Arduino board via the USB port. The code is written in C or C ++, and all the input, output, and calculation processes for any process inside this page are defined in the C language. This page is linked to several libraries called "Wiring", and a project is created according to the designer's desire, easily, flexibly, and without complication like other requirements.

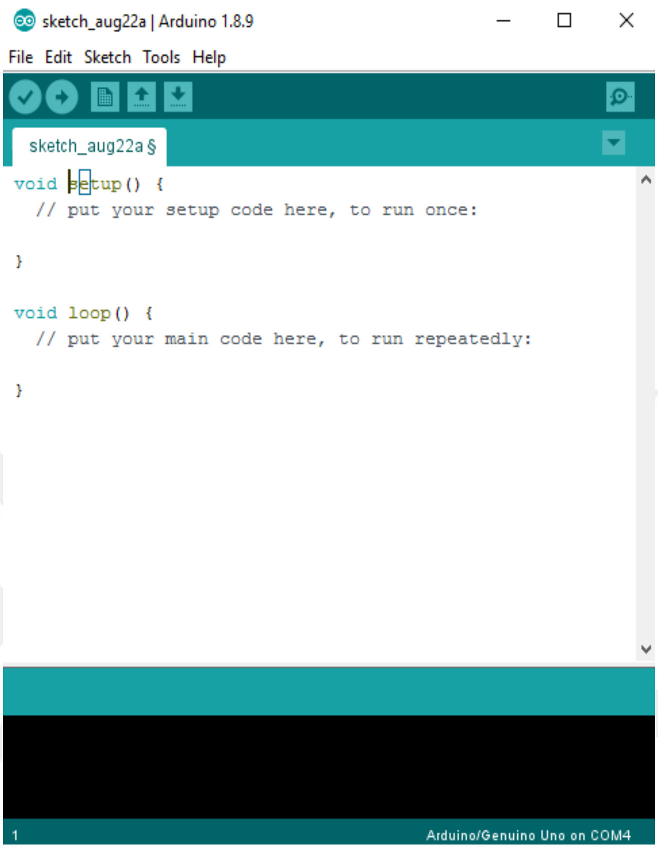

Figure 3.6 IDE arduino.

3.5 Wireless Sensor And actuator Networks for Control Applications

In this section, the architecture of wireless sensor and actuator networks (WANs) describes which represent the primary aspect for building monitor and control applications of wireless sensor network (WSN). Additionally, the main difficulties facing the design of WSANs will also be explained concisely. In general, the WSAN comprise of three parts; base stations, actuators, sensors. The functions of sensors and actuators have been earlier described in this chapter, whereas base stations are usually responsible for managing and monitoring the overall network vie communications with sensors and actuators. In the design of WSANs that will be presented in this thesis, the base station is comprised of the Arduino platform, XBee S2 module with Arduino Xbee shield, and DHT11 sensor, which basic temperature and humidity sensor.

According to whether there is a appear controller entity in the network, two WSN architectures for controlling applications can be classified, as exhibited in Figure 3.7a and Figure 3.7b respectively.

In [1], these two architectures are called automated architecture and semi-automated architecture. In the first architecture shown in (Figure 3.7(a)), there is no clear controller entity in WSN. In this case, the controller is embedded in the actuator, and the control algorithm is used to decide which actions should be performed on the physical system that should be executed on the actuator node. The data collected by the sensor will be directly transmitted to the corresponding actuator through single-hop or multi-hop communication. Then, the actuator processes all input data and performs appropriate operations by executing the pre-designed control algorithm. From a control point of view, the actuator node not only acts as an actuator but also acts as a controller in the control loop. From a high-level perspective, wireless communication via WSAN only involves the transmission of sensed data from the sensor to the actuator. The control command does not need to undergo any wireless transmission, because the controller and the actuator are logically integrated, as shown in Figure 3.7(b) [8].

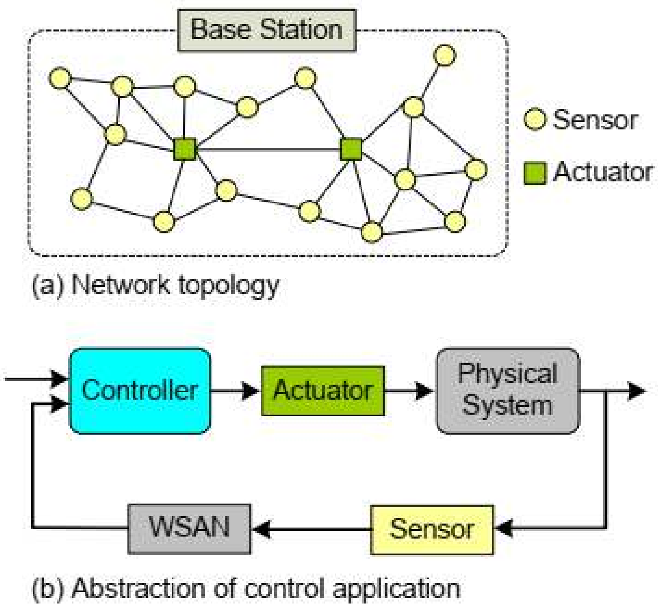

Figure 3.7 WSN architecture with non-explicit controllers.

Figure 3.7(a) shows the second type of architecture, in which one or more controller entities are present in WSAN. The controller entity may be a functional module embedded in a base station or an embedded node with sufficient computing and communication capabilities. Through this architecture, the sensor sends the collected data to the controller entity. Then, the controller entity executes certain control algorithms to generate control commands and send them to the actuator. Finally, the actuator acts. In this case, both sensor data and control commands need to be wirelessly transmitted in a single-hop or multi-hop manner. Figure 3.7(b) [8] depicts a high-level view of the application of this architecture.

3.6 Wireless Sensor Network Controller and Structure

The main contribution of the work is the implementation of a WSN for monitoring and controlling indoor and outdoor temperature and humidity, where the system will be practically established and simulated. The results will be compared with the response of the target environmental environment under the existing control. Initially, a simulation system will be created on computing software and control applications, including PID and application of this control to our process. Each node consists of a microcontroller, XBee, and a sensor. Arduino will be the exact controller used in this research that interacts with ZigBee by a shield. Arduino setup program is supported by C language. Next, we connect the circuit and type the code to control PID then inject it into Arduino. After that, we monitor the target environment to maintain the temperature and humidity in it, where the receipt of the data is received from the node, and then it is processed according to the required category. Then it is sent again to the node where that information or instructions are received and compared with the data in the microprocessor, and according to the accounts or the percentage of the lung that exists between the data calculated from the environment with the data received, the processor acts and operates the necessary action, whether for cooling or heating, according to the instructions.

3.6.1 PID Controller

Proportional integral derivative controller (PID controller or three-term controller) is a feedback control loop mechanism, which has been widely used in industrial control systems and various other applications that require continuous modulation control. The PID controller continuously calculates the error value e(t) as the difference between the expected set value (SP) and the measured process variable (PV) and corrects the term based on proportional, integral and derivative (respectively Expressed as P, I and

D), hence the name. The typical diagram for the PID algorithms is illustrated in Figure 3.8

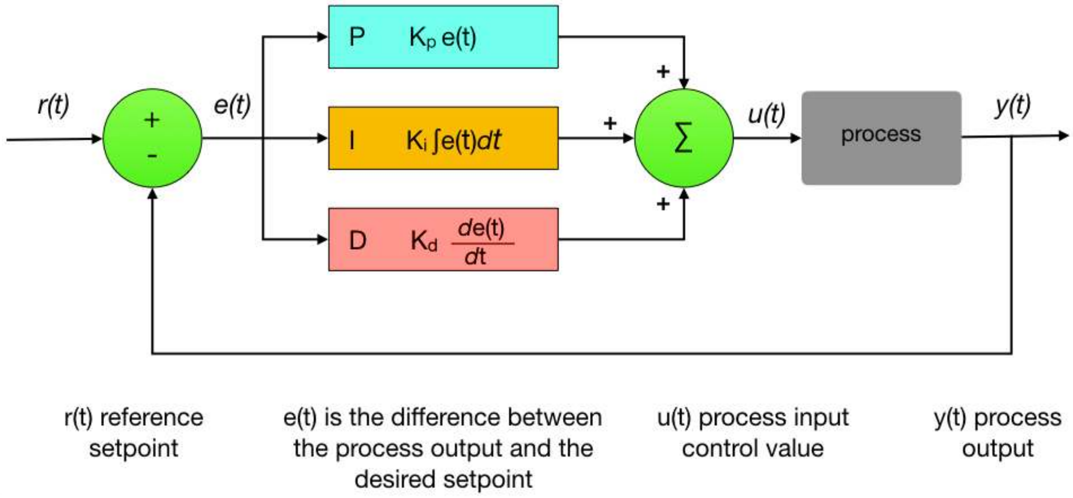

Figure 3.8 PID controller with a feedback loop.

The use of the PID controller in the feedback loop, which is in a variable measuring point (PV), the PID control system tries to make it match the required output when designing and proportional to the setpoint (SP). Any physical value that can be measured by the sensor can be applied to the PID controller, as the PID measures the error which is the difference between the value of the process variable (PV) and the value (SP) in each control loop in the system. The PID system is like a machine to calculate the error in the whole closed-loop, and from this calculation, the resulting error ratio will give the required signal to the operator.

3.6.2 **PID tuning by Reaction curve method**

The Ziegler and Nichols (ZN) in 1942 conducted numerous experiments and suggested rules for estimating the PID gains Kp, Ki, Kd based on the transient step-response of a plant. It applies to plant with neither integrators nor dominant complex-conjugate poles, whose unit-step response resembles an S-shaped curve with no overshoot. This S-shaped curve is called the reaction curve, which is appeared in Figure 3.10. Two constants can represent the S-shaped curve, delay dead time L, τ is time constant. The S-shaped curve constants can be computed by drawing a tangent line at the inflection point of the curve and finding the intersections of the tangent line with the time axis and the steady-state level line [27][28].

In this thesis, Process-Reaction Curve PRC methods are used, where the ZN method has the following tuning rule for a PID controller.

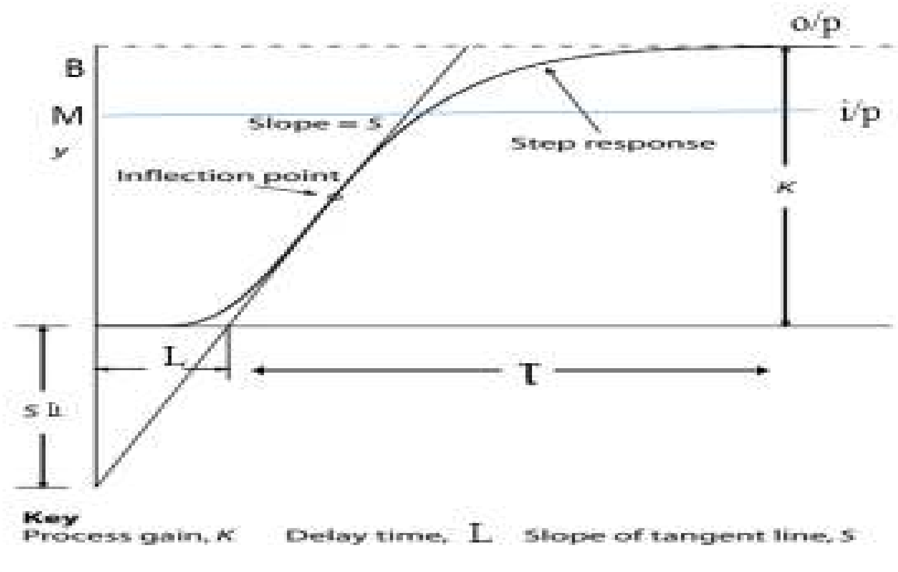

Figure 3.9 Process reaction curve of a system.

Mathematical operations are performed on the parameters obtained from the reaction curve to find the control constants by the straight line with the tangent equation. This method is based on practical experiments. The curve is drawn from the practical laboratory experiments, and the applications are based on the calculation and need the experience to find constants and formation of equations. Steps to find constants from the curve are as follows.

- A step-change of magnitude M is applied as input
- The output of the plant in response to the step input is observed.
- The inflection point is where the slope is max, and then it returns to decrease.

The reaction curve is approximate by the first-order response curve with a time delay as,

$$G_p(s) = \frac{k\, e^{-L s}}{1 + \tau s} \tag{3.6}$$

where K is static gain, τ is time constant, and L is dead time.

The reaction curve for the proposed plant, which is drawn from the practical laboratory experiments is depicted in Figure 3.13 (see Appendix A).

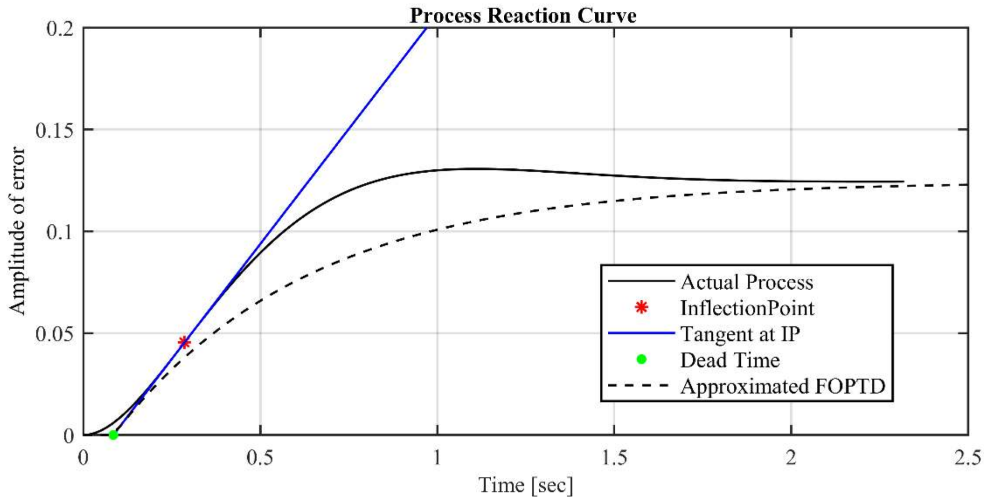

Figure 3.10 Parameters for process reaction curve of a system and the approximate parameters K =0.12 $\boldsymbol{\tau}$ =0.55 and dead time $\mathbf{L} = \mathbf{0.084}$.

3.7 Simulation Software

In this section, the design software for our system in a computing software program and circuit simulation program is discussed, the simulation is utilised to satisfy the requirements needed in real experiments.

3.7.1 Computing Software Program

A computing software is a mathematical program, and it is used in this study to extract some of the constants that are used in the control as it writes code in the simulation window that contains the transmitted function that is appropriate for the fan motor function. This function is deduced according to the characteristics of the functional motor from the amount of magnetic loss and torque, by determining types of variables. Then we configure a closed-loop system that consists of the reference point, control system and plant where we need to control with feedback from the transducer (sensor). Then we collect reading of sensor measurement, compare with reference, the value of the difference between them is an error for the closed-loop system.

The computing software simulation is run and PID control parameters are adjusted using the estimation Transfer Function (TF) for the plant. The block diagram of a system with a PID control block is as in Figure 3.11.

The block diagram to represent the plant in simulation software and adjust the parameters The system is then adopted and two control systems are formed from input and output variables as shown in a drawing in the next chapter, where the control is a parallel control of temperature and humidity and the separation between the variables is in the form of a fixed key and the ratio of the conversion between them is fixed and according to the choice. To switch between a humidity and temperature controller, as an example, a reference temperature will be where a system will stay inside to find a good system response, create a closed-loop, and compute all variables in the computing software program. As well as forming equations to solve the second-degree system utilising the scheme and the equations shown below. We calculate the error at the steady-state in the next chapter.

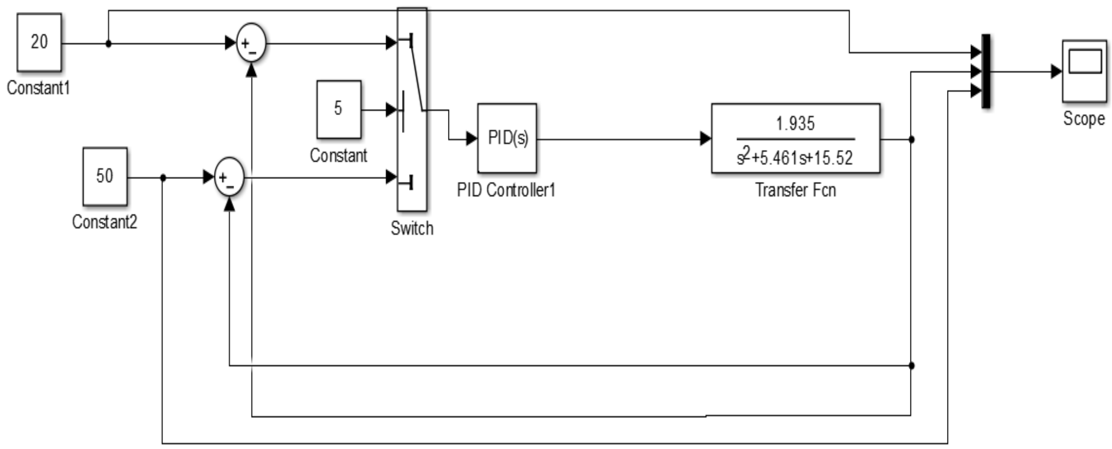

Figure 3.11 System model plant control.

Table 3.4 Performed system parameters

Parameter	**Value**
Rise time	0.142 sec
Settling time	0.817 sec
Overshoot (%)	7.61 %
Steady-state value	2.05

3.7.2 **XCTU Software Program.**

XCTU is free software from Digi. It allows setting up, adjusting and reading the XBee device, such as identifying the device as a coordinator or a router and can connect and identify nodes on it, collect and receive data, specify the destination address. It can be considered a terminal node or Coordinator, define a pin, apply configuration changes and create an operating channel on it. It has the following options**:**

- Provide range testing tools.

- View the Received Signal Strength Index (RSSI).
- It is including an integrated terminal window.
- View ASCII characters and hexadecimal characters at the same time in the device window.
- Automatic detection of unit type.
- Configure packets test data for trial transmission.
- Easy to configure and set up.
- Receive data in Hexa an ASCII mode.

3.7.2.1 Communication Test

The implementation of the WSN nodes detailed in the last of this thesis (see in Appendix A) that comprised over 40 lines of Micro C-Code.

The difficulties encountered in WSN for environmental application in the past that led to this research as regards online real-time node failure detection, user-friendly data representation and using the cloud as a secondary storage server to store large accumulated data that may exceed PC (BS) storage capacity have been solved in this project.

There was a clear-cut defined communication protocol to transmit data (TX) and Receive data (RX) encoded in the API Mode function set. To do this test, two methods are performed:

- XBee to XBee: Its wireless communication process between two devices. A simple communication session between two ZigBee devices as Valudi calls it [42].
- A communication session from Arduino to Arduino: It is a simple communication process that takes place between two Arduino devices, where the transmission is tested by an LED, where one is identified as a coordinator and the other is a router and the message is sent from the router to the Coordinator by XBee devices, who sends messages by serial.

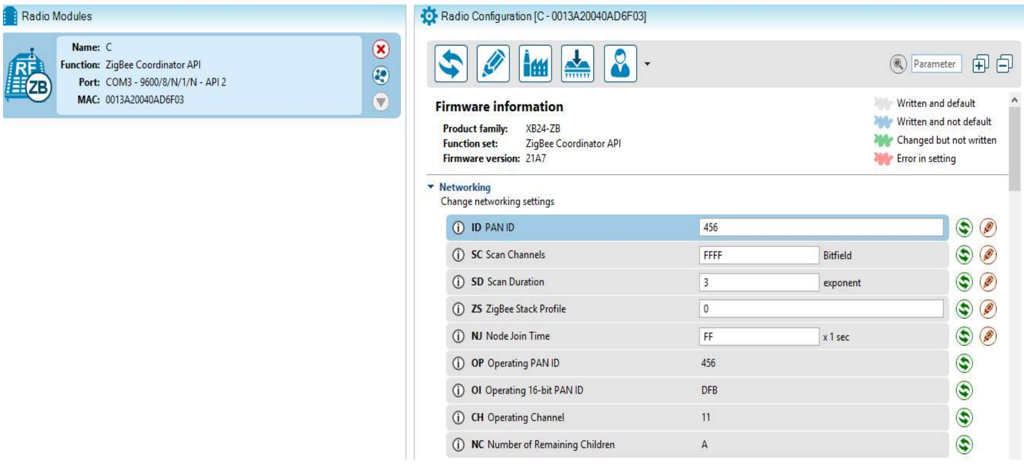

Figure 3.12 Coordinator node configuration.

In the next step, the signal strength was tested, in which the router was placed at a certain point and gradually moved away with the Coordinator from the router node. We measure the distance between them to determine if it was able to secure a good 250 meters' connection, and the router was directed to send data from it to the Coordinator, receive it and re-send the router messages such as resetting the temperature, giving signals to the control unit, which in turn moved the triggers according to the instructions. Another device to form a complete network and the communication process between them is added. Each router can be identified by MAC address to determine which node will be changed, as shown in Figure.3.13

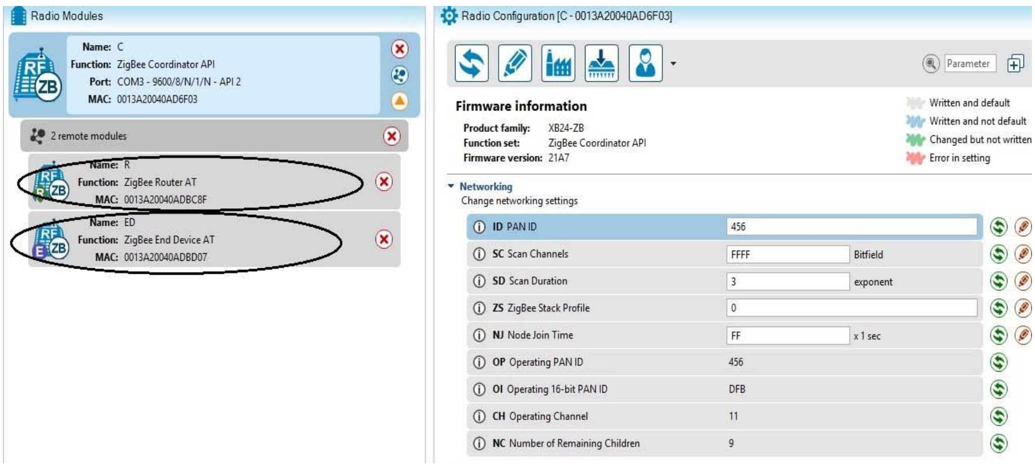

Figure 3.13 Router and end device connected to the network with the coordinator.

3.7.3 Circuit Simulation

The circuit simulation program is an electronic circuit design software which includes design, configuration schematic diagram PCB planning modules (printed circuit board). The PCB design tools are individually viewed among the PCB layout design tools. At the same time, selects component in the PCB layout, appearing in PCB design which it helps during the soldering of components in PCB. The circuit simulation program is also useful in simulating the circuits containing the microcontrollers where we can simulate the circuit by uploading the hex code to the microcontroller, whereas Multisim cannot do this. Netlist file (Written in Verilog Code) if needed for your users can also be extracted from the tools option in this software.

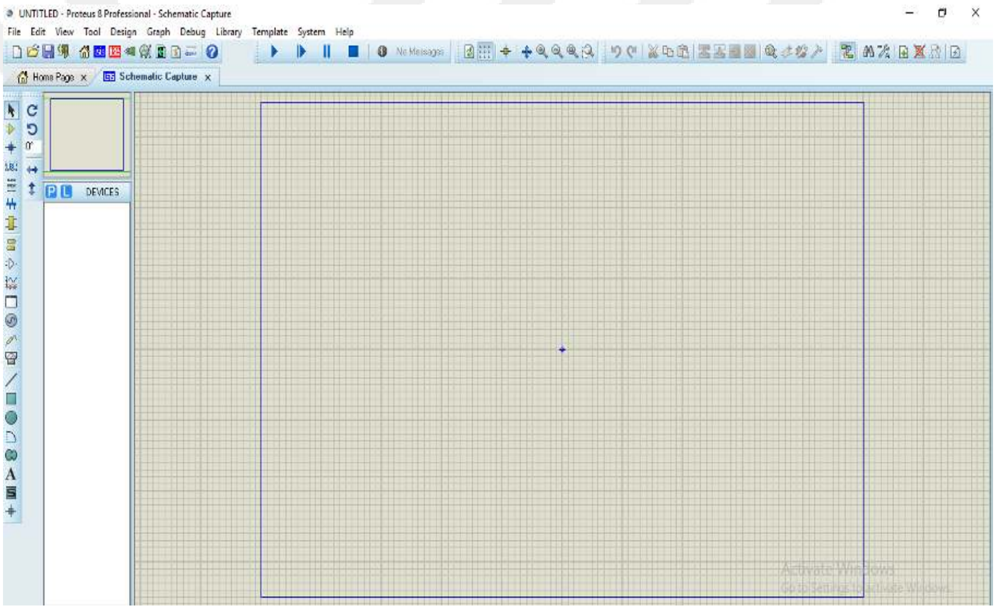

Figure 3.14 Circuit simulation program, software bag ground.

We use the Circuit simulation program to achieve the results taken from the computing software before applying it to the real-time. This is done by drawing the circuit on the program and using the Arduino with a temperature and humidity sensor and XBee in addition to a DC motor that represents the fan for cooling and the heating heater as shown in Figure 3.15. We also connect a terminal screen to display the temperature and humidity and the amount of PWM and the extent of its change with the change of temperature or humidity. The oscillations show the waveform coming from the Arduino to the motors (fan and heater). The pre-written code is loaded on the Arduino platform that includes PID programming and determining the value of the constants in advance and is injected as a hex-file into the microprocessor, after which the program is run to see the results, as displayed in the next chapter.

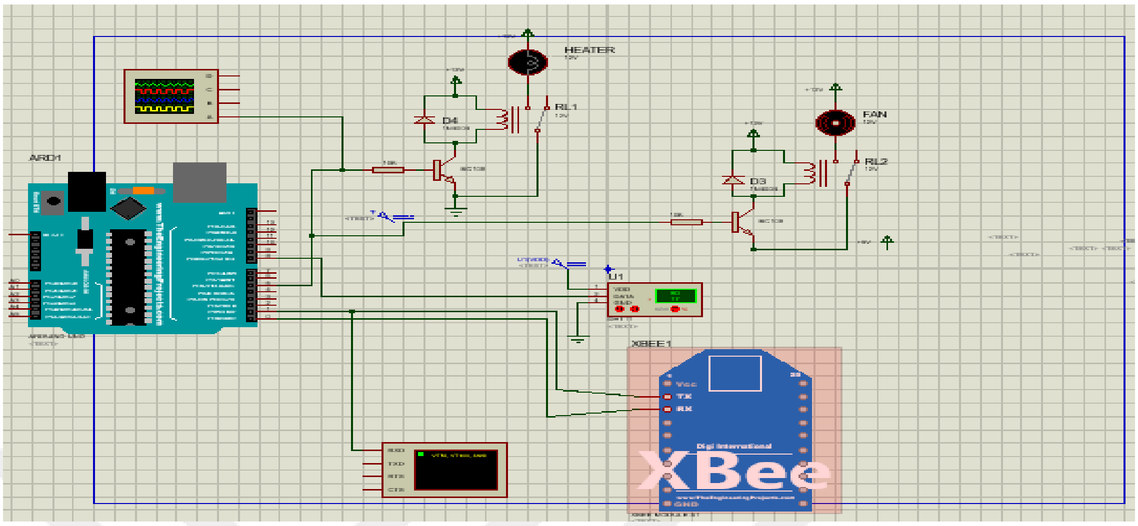

Figure 3.15 Circuit design in the circuit simulation program.

The advantage of this simulation is that any change to the model settings can be applied directly to the model, and the model can be checked automatically before generating the code. The configuration of WSNs is easy to install, compact, and low-cost, which makes them essential in industrial and agricultural applications because they monitor and control the required areas. It allows the simulation process to be more flexible in changing the values of the control constants according to the requirements of the target environment easily and create them according to the desire of the network, whether internal or external. It is also characterized by the ease of installing these networks in the program and the flexibility to change many parameters.

In our research, a WSN was built to monitor temperature and humidity, both inside and outside. The goal is to sense the surrounding environment at the exact location and send it to the base station as shown in Figure 3.16, where data is collected, analyzed, and plotted in the form of a graph or curve showing the change of temperature and humidity.

3.8 DHT11 Connection with Arduino

This library has been downloaded from the official serial link of DHT to allow the DHT11 temperature humidity sensor to connect to the Arduino program by opening the libraries option within the Arduino program where the sensor library is downloaded, and then the Arduino private pins are defined that the sensor is connected to the place where it needs. In our research, a WSN was built to monitor temperature and humidity, both inside and outside. The goal is to sense the surrounding environment at the exact location and send it to the base station as shown in Figure 3.16, where data is collected, analyses, and plotted in the form of a graph or curve showing the change of temperature and humidity.

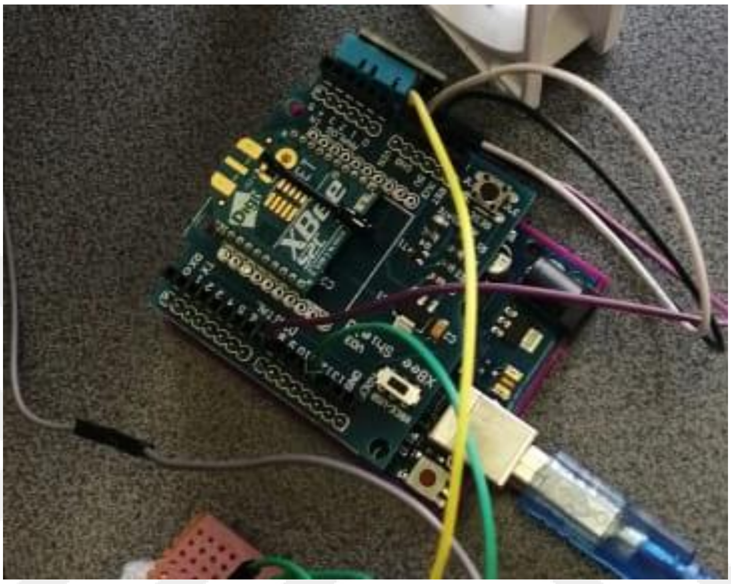

Figure 3.16 Connect DHT11 sensor to arduino in node.

The micro C code is written for real-time testing the ability of DHT 11 Sensor, and the measured results appear in Figure 3.17.

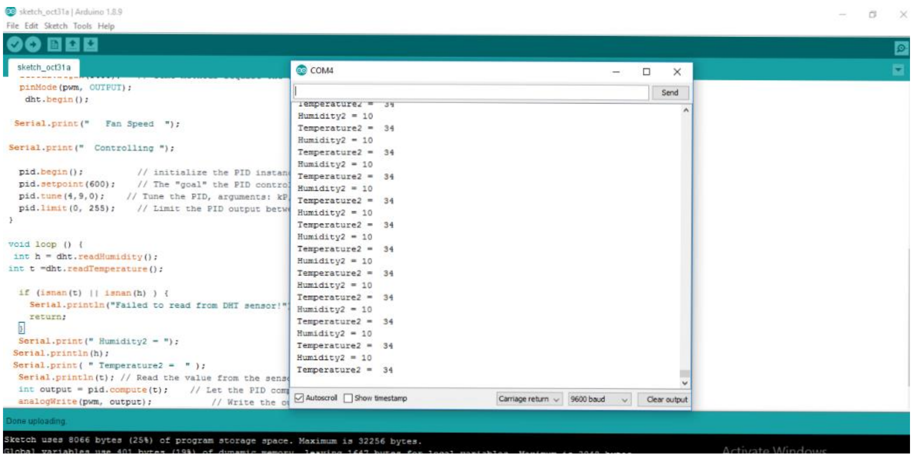

Figure 3.17 Software IDE arduino test with DHT11 sensor.

The image in Figure 3.17 shows the real-time measurement of the sensor and how the Arduino private program is configured and the readings change simultaneously with the environment changes that surround the sensor from the change of moisture or heat and the appearance of those readings. The serial monitor for the Arduino program shows results in the form of Digital numbers, the user can directly record these readings and monitor any changes that occur in the system.

3.9 Arduino UNO and XBee Node

This stage deals with the implementation of authentication. The administrator sends a new key through the control panel and frame configuration, linking Arduino with ZigBee via his own shield and attaching the shield over Arduino and connecting it to this after operating the XCTU program, which later explains the image in Figure 3.18 explaining the process of linking Arduino to XBee.

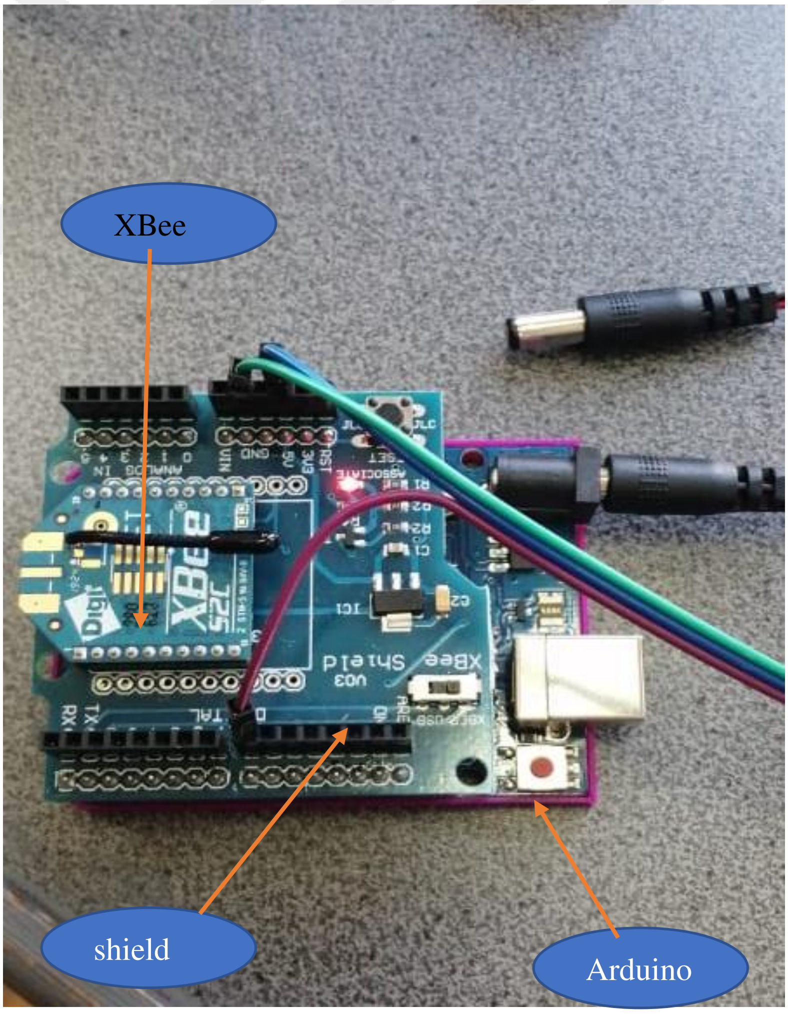

Figure 3.18 Arduino with xbee shield construct.

Figure 3.18 shows how to connect the shield with Arduino and how to install it to connect XBee to it. Since it is possible to transfer between the Arduino system and the XBee system via a key in the shield, which means that when you want to inject Arduino into the program or code, it must be converted into USB mode and code injection. As for the process of sending data from the node to the supervisor, it must be returned to the XBee mode, while the work is done easily and smoothly. Then, the node is placed in the specified place to be monitored, which is attached to the microcontrollers of the fan and the heater to work on the regularity of the target environment as required by the designer.

This step covers establishing authentication. The administrator sends the new code through the dashboard and configures the framework. Then, the Arduino will process the incoming data using the serial message and encode it by the code RC4 and re-send it to the node-set to monitor and measure the temperature. The message destination must be determined by writing the MAC address inside the frame sent from the Coordinator to the node to choose the appropriate sensor node. That message, the node decodes it with a key stored inside it in advance, and after decoding the Arduino will read the sensor message and compare the data coming with the existing and read by the sensor and by analyzing the error and using the control system added to the code applied to the controller, it will instruct the actuators as read. The data from a change in temperature or humidity is done with this control in the target environment, it will take corresponding measures. After collecting the required data from the target location to monitor its temperature and humidity, the node will use the new key to return the data. After encryption, this new encrypted message will be sent to the main monitoring station. Then, we analyse the data arriving at the station and process it according to the required data. The process of collecting the results is carried out in the XCTU program. The XBee we used to be to be called XBee 802.15.4 or 2 series to distinguish another XBee unit 2 series, where the two XBee modules are almost the same [43].

3.10 Configuration of Nodes

3.10.1 Client Node Configuration

The proposed design of client node consists of Arduino platform as a microcontroller, ZigBee via won shield, an actuator with deriving circuit and DHT11 sensor used for sensing of humidity and temperatures from climate around the node. The digital pin of Arduino connects to the drive circuit, which consists of transistor and voltage divided resistors. The Arduino pin with PWM technique is used to drive the transistor gate, which makes it work as a soft switch. The drive module is responsible for controlling the speed of 12V DC fan or 7w hitter. The configuration of the client node is illustrated in Figure 3.19.

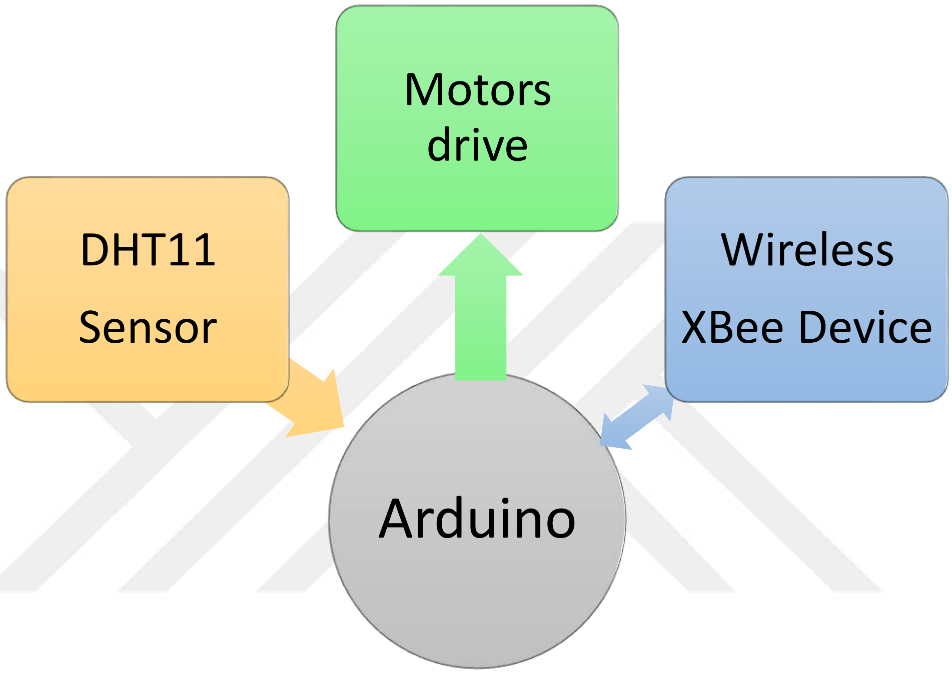

Figure 3.19 Block digrame for client node {arduino +DHT11+drive +wireless}.

The circuit configuration in the nodes consists of a fan and heater as well as a motor to influence humidity. The circuit is configured so that the fan works as in Figure 3.19 where you connect the transistor so that the signal amplifier works to control the speed of the fan or heater coming from Arduino. Also, we have a diode to protect the transistor and the circuit from high voltages that may damage the transistor. Then we connect the node as Figure 3.20.

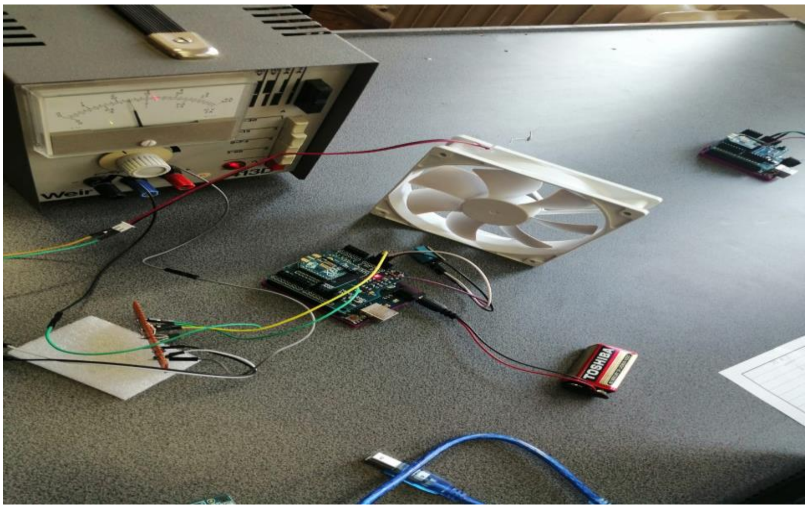

Figure 3.20 Client node configuration

Figure 3.20 shows a node configuration and how we can implement a node to connect the equipment as explained below. After that, all elements of the nodes were threaded through the voltage source by following the user guide for the model to feed the fan and operate it. We connect the 9-volt battery with Arduino to operate it and connect one of the digital pins from Arduino to model fan and connect another heater or humidity control actuator to Arduino pins. Finally, an Arduino + DHT11+ XBee board is then ready for deployment in the field.

After preliminary steps for the hardware configuration of the system is completed, it is time to put everything together and make it work. Because parameters to be measured are humidity and temperature, we integrate the humidity and temperature sensors to the node, which were chosen according to preliminary research. For wirelessly control of temperature and humidity, a DHT11 digital sensor that provides good accuracy in measuring temperature and humidity together in a digital rather than analogue way is preferred. The voltage is directly proportional to the percentage of temperature and humidity and it does not require any external calibration. In Arduino, it is about a simple mathematical application and the process of writing a special code that will read or display on a computer or in a node and send it in XBee format. The data received from the node is calculated and analyses according to the required values, where the error rate is calculated and compensated as we will see in the next paragraph. Arduino performs a schematic diagram of the process described above and sequentially transmits the value of temperature and humidity to XBee Radio. XBee Radio, the router or end device will send this information to the moderator which is used for receiving and processing data correctly and saving it in a text file.

Then we deploy the nodes within the target area taking care that the node does not exceed the permeable Coordinator or the area that XBee can cover, which is sufficient and a large area to measure the final results for implementation.

3.10.2 Coordinator Node Configuration

The Coordinator or router node is used for wireless data collecting. This node consists of a ZigBee device (XBee) with its explorer shield and a computer that includes the XCTU software. The block diagram of the proposed coordinator node is given in Figure 3.21, which will be explained more clearly and in-depth in the next chapter**.**

Figure 3.21 Block diagram for coordinator node (wireless receiver (xbee)+xbee explorer + PC).

CHAPTER FOUR
RESULTS AND DISSECTION

4.1 Introduction

In this chapter, the system operation with the final configuration for the WSN that has been installed and designed are discussed, and various functions of WSN based on the PID controller is also explained. Later, the data collected in different cases is explained. Also, different real-time experiment scenarios are applied to the presented WSN system. The complete diagram for the developed system is shown in Figure 4.1.

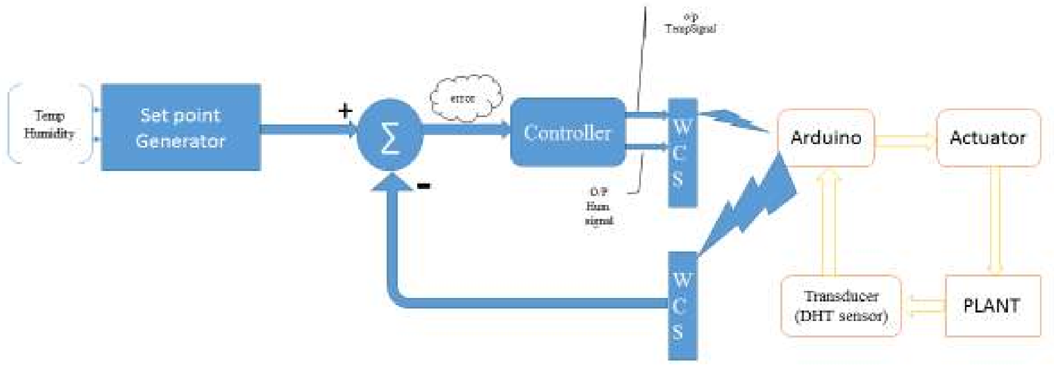

Figure 4.1 Diagram of the implementation system of WSN and control.

The full design of the developed WSN based on the PID -controller consists of a wireless module using for signals transmission. In the wireless received mode, the receiving signals from the coordinator node are used as a setpoint signal for the closed-loop system. While in the wireless transmitted mode, the recorded data for temperatures and humidity at the sensing node is transmitted to coordinator point.

4.2 Real-time Studying Cases and System Initialization

In this section, the real-time testing of the developed WSN under different conditions are briefly described.

4.2.1 Base Station (Coordinator)

The base station or (Coordinator) plays a significant role in WNS. Coordinator node represents the centre of data acquisition in our proposed system.

This station is comprised of Explorer; XBee Shield supports XBee ZB Module. The Coordinator is shown in Figure 4.2.

The WSN collects sensor readings from end devices via the XBee ZB module and using advanced packet protocols via an Ethernet unit for transferring data. The Ethernet unit works as the central node to plug the WSN with the local proxy. In general, the function of the base station is divided into two aspects: first is the process for distinguishing client nodes and looking for a secure connection between points.

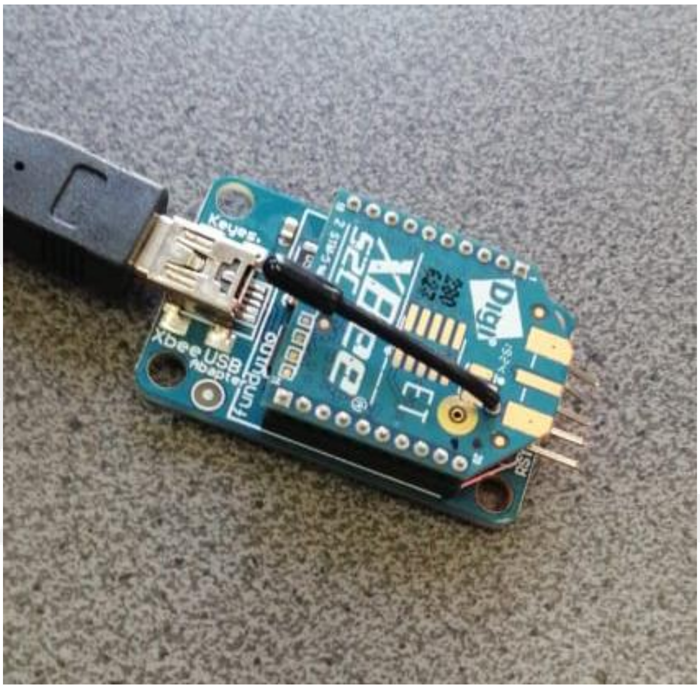

Figure 4.2 Base station xbee connected to PC.

To the configuration of the XBee Shield supports XBee ZB Module, the computer with the Windows operating system is used, and the coordinator node connects to a computer via a USB connection port. After the connection is confirmed, the screen will display the connection of those nodes on the XCTU design program, as shown in Figure 4.3. In the XCTU program, the property of each node can be identified, and the program gives flexibility in testing the connection and computing number of times to live bits per second.

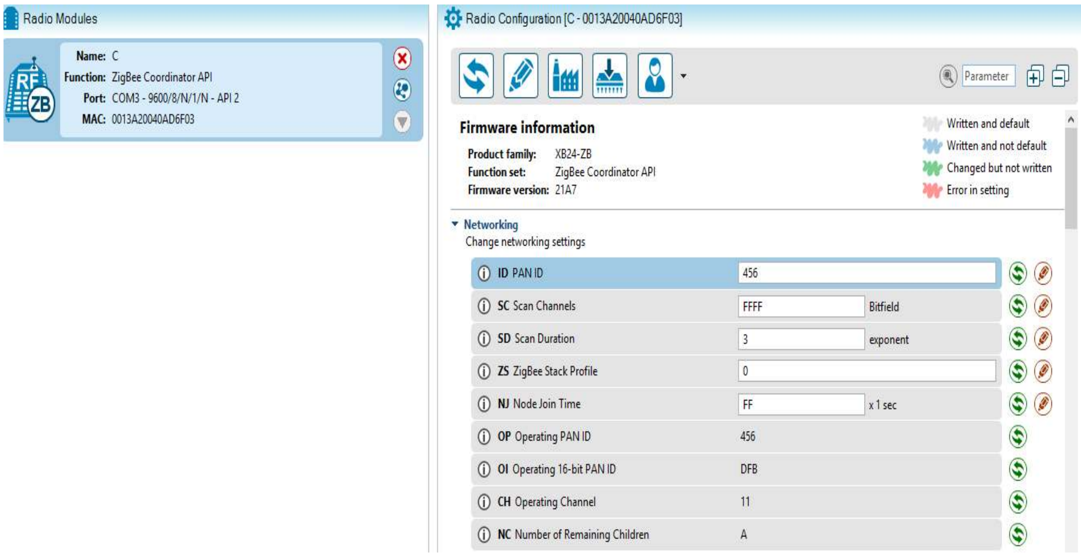

Figure 4.3 Coordinator node configuration.

After the data collection process from the sensor nodes is completed by using a wireless XBee protocol. The collected data are storing in excel sheets to easy draw and analyses.

4.2.2 Client Station (Node)

The client station or sensing node represents a data collection station for sensors before transmitted to the coordinator station. The client station consists of Arduino Uno, Zig Bee module, and DHT11. The full connection of the client node under laboratory testing is exhibited in Figure 4.4.

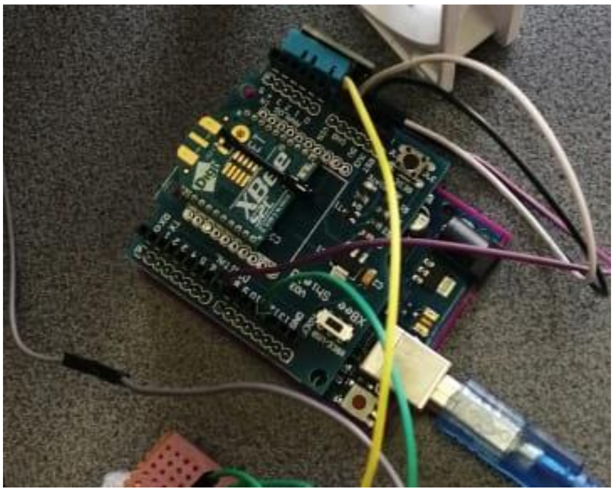

Figure 4.4 Client station under laboratory testing.

4.3 The effect of Distance Variation on the Developed WSN

To able be understand the wireless sensing system, the effect of distance variation on the developed WSN must be investigated. The important thing of the distance variation is to validate the feature of mobility for sensing node and to determine the time delay for data transfer between nodes. Thus, different distance is considered for client node whilst the coordinator node set to be fixed, the results of effect for different distance on transfer packet is shown in Figure 4.5. It can be noted from Figure 4.5 that the connection between nodes can be verified by watching and recording the rate of change on direct information. Also, the sensor node responsible for sending the temperature and humidity values must be defined. Otherwise, there will be pending, and its need to sending a command by the coordinator, and the data reception process will not happen due to lack of synchronisation in the connection and data transmission between nodes[29].

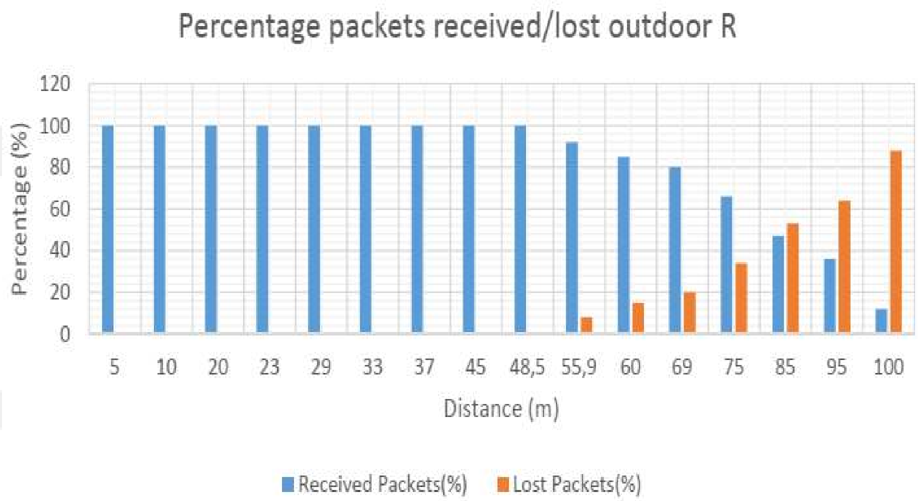

Figure 4.5 Different range test between coordinator station and the client station.

4.3.1 The Sensing Results for Humidity and Temperature at Node

As previously explained, computing software has been used to find PID- controller gains by building an equivalent mathematical model for the real-time circuit. The mathematical model includes transfer function for an actuator which are fan-powered plant or heater. The feedback signal is measurement values for temperature or humidity. Figure 4.5 shows the simulation block diagram of the closed-loop PID controller for WSN client node.

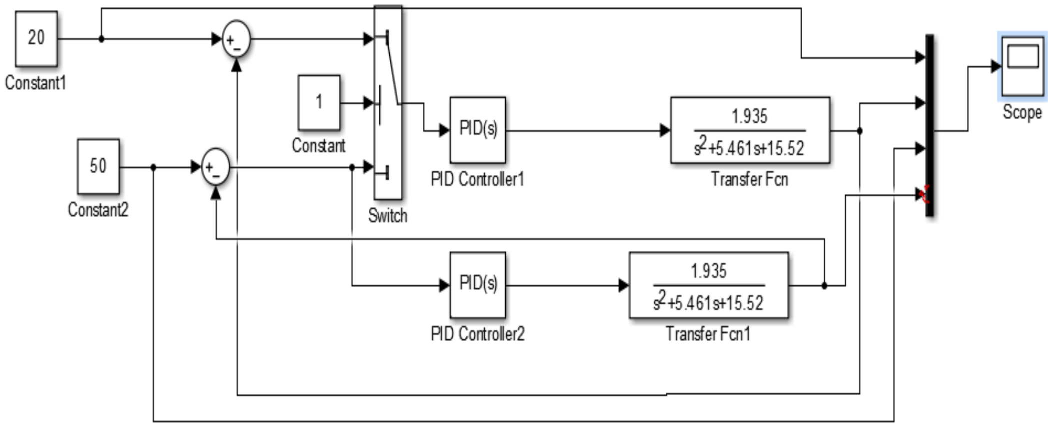

Figure 4.5 Building block diagram of a system in computing software.

The results for PID-controller tuning appears in Figure 4.6 It can be noted in the figure of system response, the control state of the system, how many seconds it takes to reach a stable state and calculate the error rate. The solid line of the system response indicates the response of the system after the PID auto-tuning. In contrast, the dotted line indicates to system response before applying auto-tuning for the PID controller gains. Also, in Figure 4.6, PID controller parameters are determined, the value of the maximum overshoot 7%, which is being an acceptable value. Moreover, in figure 4.6, the settling time seems to be a reasonable value, which is in the range of 1.2 seconds.

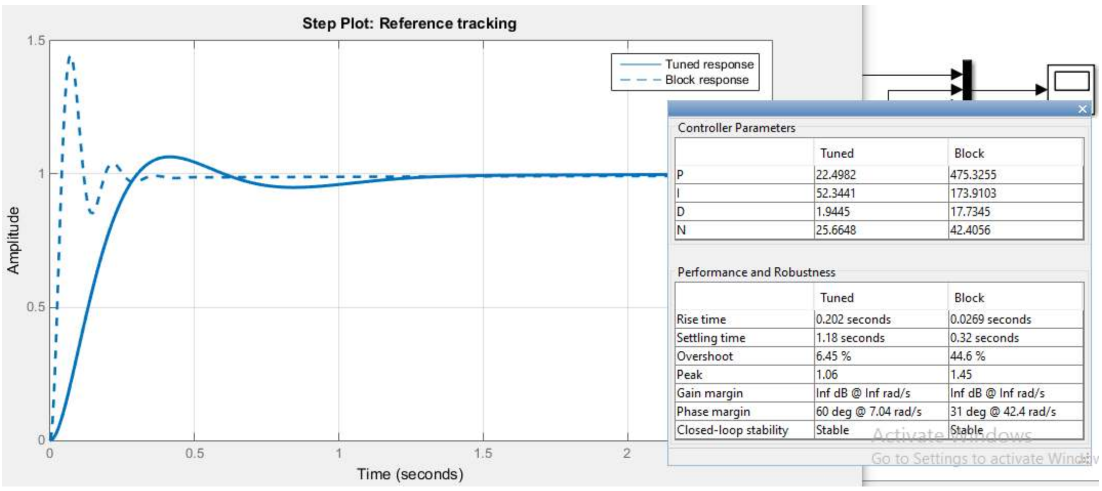

Figure 4.6 PID control response in computing software.

Figure 4.7 shows the system response for temperature and humidity. The constant setpoint is utilised for temperature and humidity, which are equal to 20^{c} and 50, respectively. Also, in figure 4.7, the red colour indicates plant response at humidity setpoint 50, and the response curve takes a long time to reach a stability region. Whilst the curve at the bottom of figure 4.7 indicates system response when the temperature is utilised as a set point. It appeared that the temperature response is slightly stable than humidity response curve. Moreover, the humidity curve takes time to reach stability region.

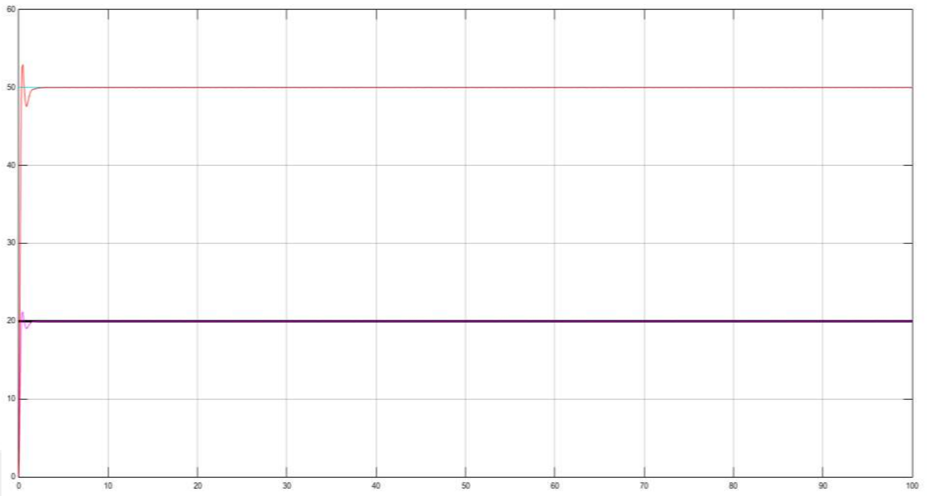

Figure 4.7 PID result response for the system.

4.4 Circuit Simulation Program Results

In the circuit simulation part, the computer simulation program is used to design and draw the electronic circuit. The electronic circuit is comprised of Arduino Uno is a board based on the ATmega328 microcontroller DHT11 sensors, and accouter drive circuit (see in chapter 3 and Figure 3.15). The computer electronic circuit simulator gives the ability for injected the hex format of microcontroller code. The Pulse Width Modulation (PWM) technique is utilised to control the actuator the essential principle of PWM is described in Figure 4.8. The duration or period is the inverse of the PWM frequency. The call range of analogue Write () is 0-255, so analogue Write (255) requires 100% duty cycle (always on), while analogue Write (127) requires 50% duty cycle (half of the duty) ratio or (on half the time).

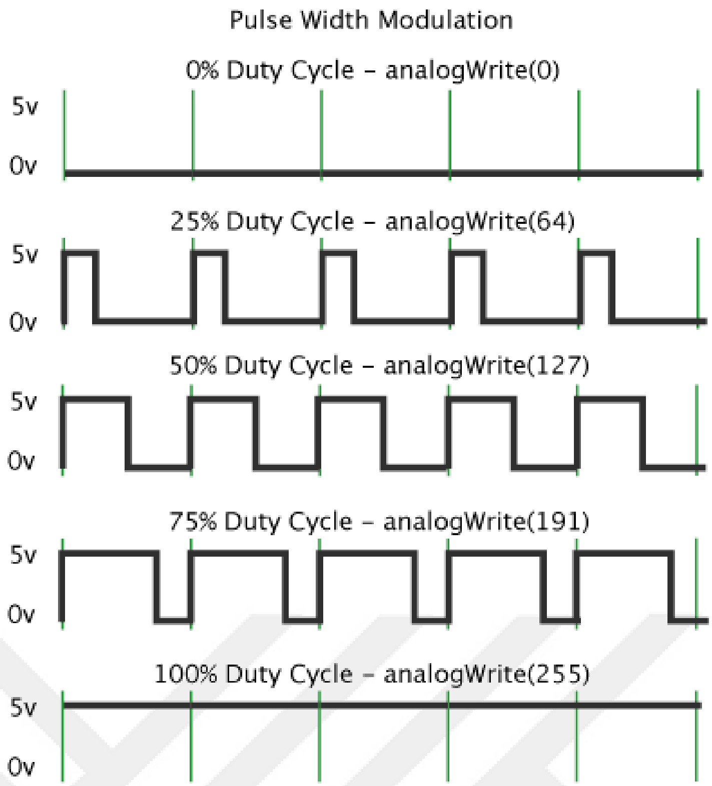

Figure 4.8 Pulse width modulation PWM and duty cycle [30].

Figure 4.9 shows the implements PWM which generate a train of pulses. The digital oscilloscope showed a PWM as number adding that to display sensor reading and setpoint, and there can calculate an error from these readings.

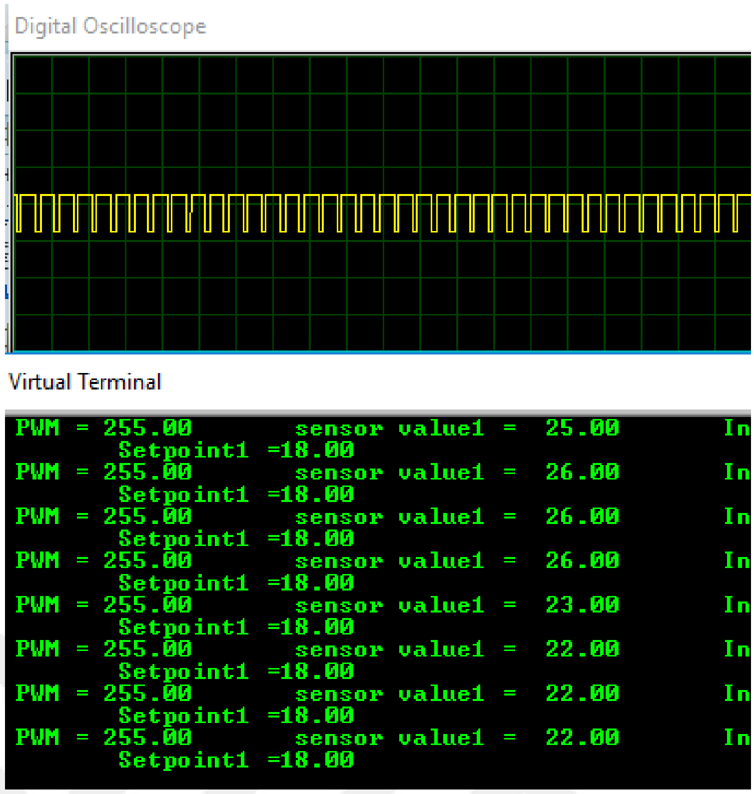

Figure 4.9 PWM for actuator drive circuit in the simulation program.

4.5 Wireless Sensor Data Collection

One of the major uses in this field by the ZigBee system that uses the stander is the collection of data by WSN that has been implemented using the Arduino code found in the node. Where he collects data by connecting the sensor with the Arduino and sending it to the base station through the XBee, where the system was used inside and outside and seeing the difference between the measurements and the amount of error and how the ratio of control to the motors connected to it and how they respond to the influence of temperature and humidity. Moreover, the following figure shows how to receive and display the received data in the PC station.

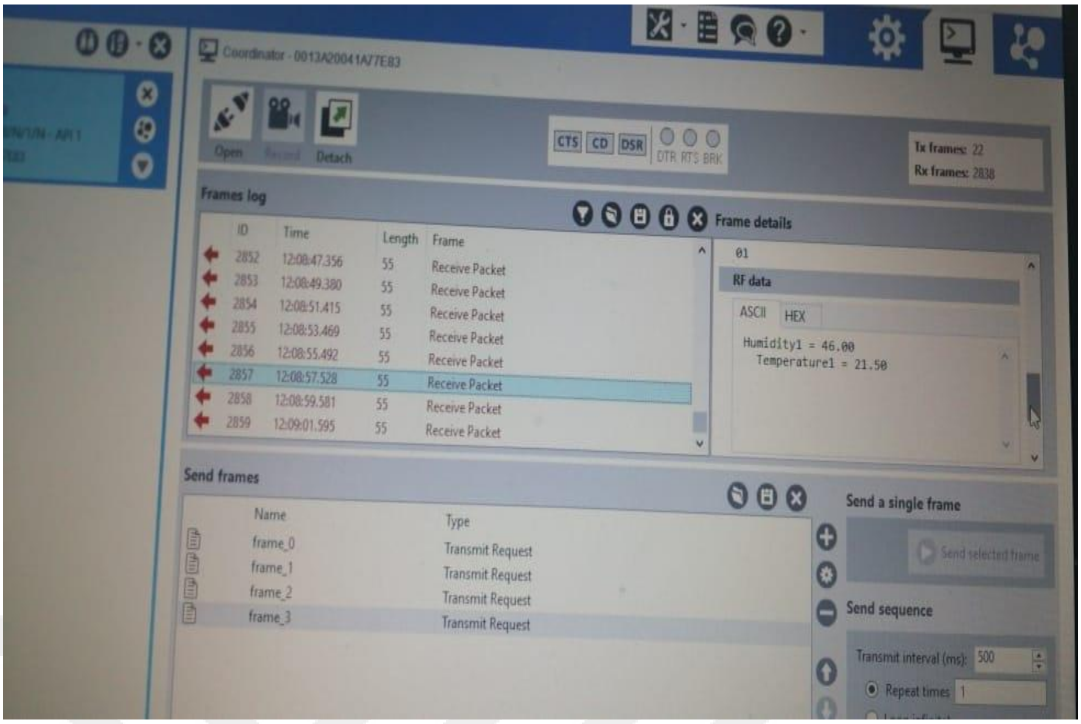

Figure 4.10 Data collection in base station.

As the figure above also shows a picture of the computer and shows the packet of received and transmitted data and time for each transmission and reception process, the symbol appears for each exchange of information between the node and the station, and the other side of the screen shows the values given for the received temperature and humidity and the information for each complete frame. As these values monitor the target area to control their temperature and humidity, At the bottom of the screen, the number of frames used to send special commands to control the operation and shutdown of any node as needed and the designer desires are complete. That is, each frame carries specific prompts and instructions and also contains the XBee MAC address, and through this address, it will determine which node responds without any change. For the entire network, this gives more flexibility to the WSNs to control and control the required environment. For this reason, these networks have become very important in many applications, especially in agriculture and industrial areas, as they provide accurate, safe and fast information to monitor the workflow from the control and control station without the need to move within the farm or factory or any other unit that wants to monitor and control the machinery or its production.

4.6 Results of Sensing Data

After collecting data from sensors nodes, we implemented a table to display the values of temperature and humidity. Table 4.1 shows the reading of humidity and temperature form nodes over there was no clear distinction between the two coordination points and we because decided to adopt a tape display graph data. The adoption of a bar graph was because it gives a better display that is easy to use and gives a greater understanding of changes in temperature and humidity [23].

Table 4.1 Reading humidity and temperature form nodes

Number of nodes	Reference temp	Humidity	Temperature	Time (min)
Node 1 (outdoor)	20	60	25	1
Node 2 (indoor)	18	56	22	5
Node 1 (outdoor)	20	56	23	10
Node 2 (indoor)	18	53	21	15
Node 1 (outdoor)	20	53	22	20
Node 2 (indoor)	18	52	19	25
Node 1 (outdoor)	20	50	20	30
Node 2 (indoor)	18	50	18	35
Node 1 (outdoor)	20	50	20	40
Node 2 (indoor)	18	49	18	45

According to the data received, we will calculate to calculate the error and send the corrections to the node to make it the best job. In the above table, displaying the reading contract in the coordinator, one of which was indoor and another that was outdoors to show the different measurements for each one and the efficiency system stable for all conditions, whether indoor or outdoor, would preserve the environment at the reference point. The error is calculated from the following equation in which the error ratio exists for each data collection process and the corresponding time and it shows the variation of the values in the above table, the system works well, that is, it settles on the designed degrees and remains on them for temperature and humidity, but it takes time because the system rises in an open environment and the actuators that work to keep the system a little small compared to the surrounding environment.

$$\mathrm{E(t)} = R_p(t) - R_f(t) \tag{4.1}$$

Where $R_p(t)$ is setpoint of temperature and humidity, $R_f(t)$ is feedback signal which coming from sensor and $E(t)$ is error value versus time.

Therefore, the calculation of the error in the above equation will not be very accurate due to the proportions of the variance between the variables and the influence time is relatively large between each reading and this shows that the error percentage is large in the beginning and decreases with each reader to reach zero and this is achieved by adding the PID control to the system which is Try to put down the error to zero as soon as possible and bring it to stability.

4.7 **Result Analysis and Drawing Chart**

The last page is DATA. This page provides remote access to the data saved in the sensor data and draws it in the form of charts, for the system to appear and the changes in the target environment to control its temperature and humidity. At the beginning of this section, information is noted about the data received from the node. For example, the number of rows currently saved in the database. I can find sensor node data available along with the start and end dates[29].

This makes it easy to analyse and understand information. The data received is used and drawn on a chart representing temperature and humidity change over time. For each node separately, whether placed outside or inside, show the error rate between the measurements and the system's response speed, and show the effect of disturbances on the system's response speed. In the following figure, the graph shows the data received from the node placed outside and clearly shows that it takes a long time to reach the reference point, both for temperature and humidity, due to the large turbulence that occurs on the system, which delay the arrival to the state of stability quickly and get rid of the error and reached it to zero.

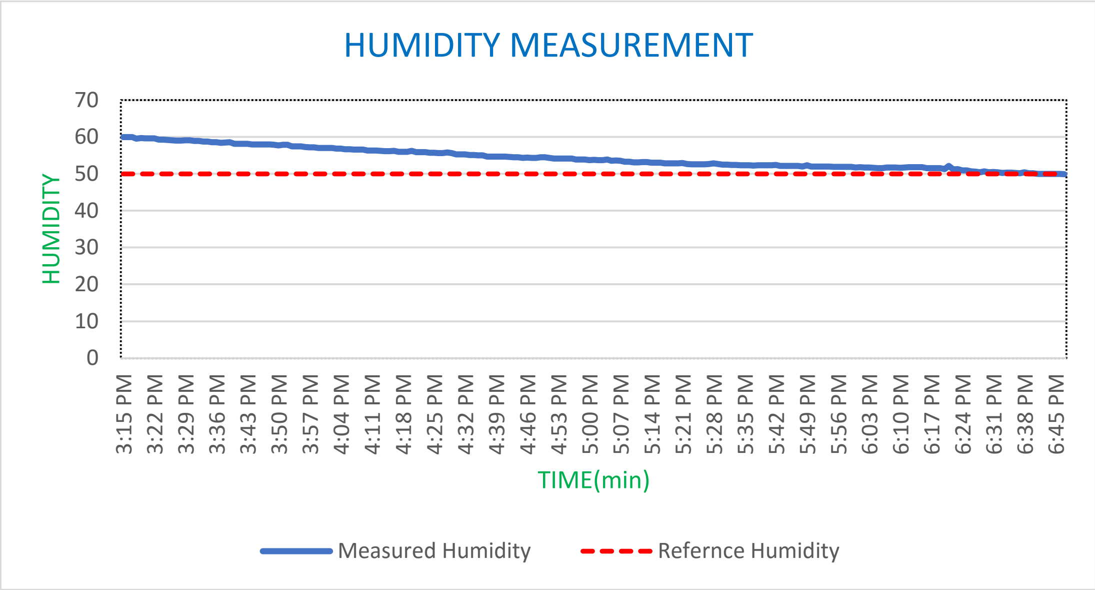

Figure 4.12 Outdoor real measurement humidity.

The figure above shows a curved graph of the measured real-time humidity values and provides a delay range for reaching the reference humidity. The reason for this is that the system suffers from many disturbances as well as the ratio of triggers associated with the node with changes in the atmosphere and the error rate is calculated for each reading and therefore the average error in the system is 4.276056338, and this is for measurement outside. Due to these major disturbances, it takes longer to reach the reference point than in the room or building, as shown in the following figure.

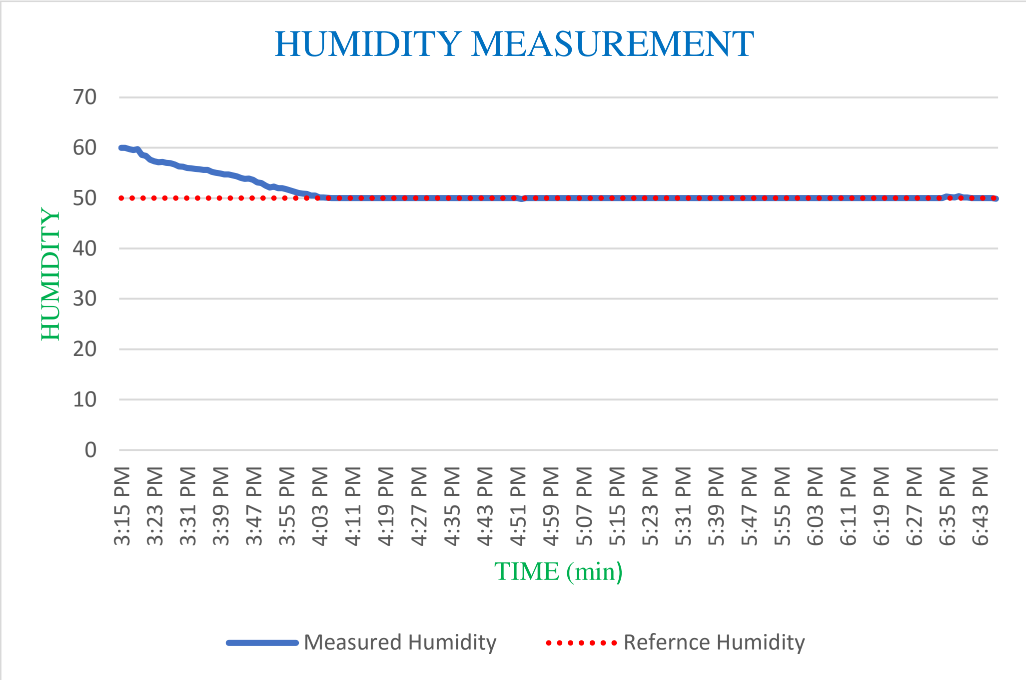

Figure 4.13 Indoor real measurement humidity.

As for this curve, it also shows the values of the degree of humidity measured in the nodes that are inside the room and are at the same time with the measurements of the node in which is located outside. The difference is clear because the work in this knot is much faster to reach the reference point, the degree of humidity. After all, the disturbances, in which it is much less, therefore we see that the access to the degree required by the designer is captured and that the average error also will be much less than the previous one, where The average error is equal to 1.117183099.

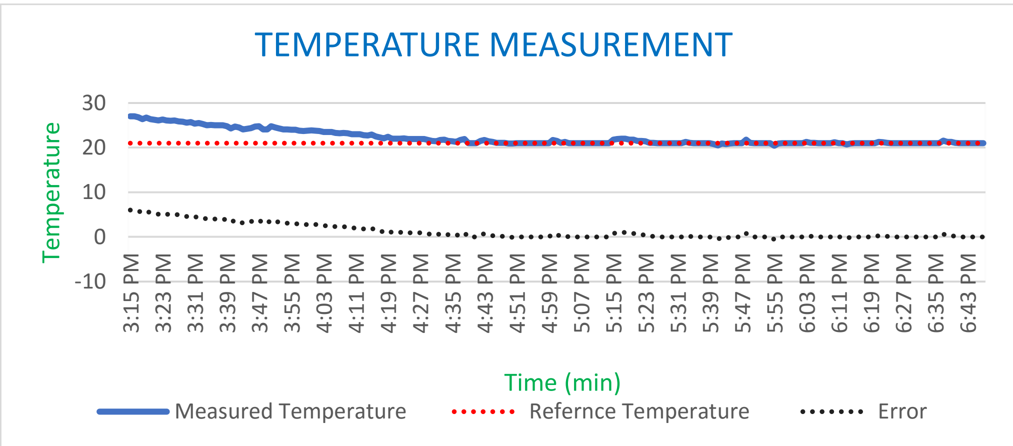

Figure 4.14 Outdoor real measurement temperature.

above chart show temperature measurement in real-time. and appear a difference between measurement humidity and temperature due to the temperature is reached to reference point as quickly because of the difference between the temp reading and reference is less than in the humidity and it reference so that the error for each reading will be less at the total the average of the error will be less for humidity. but we can notice the changes in reading and is unstable due to the sensitive change of temperature. and the average error for reading outdoor is 1.194835681.

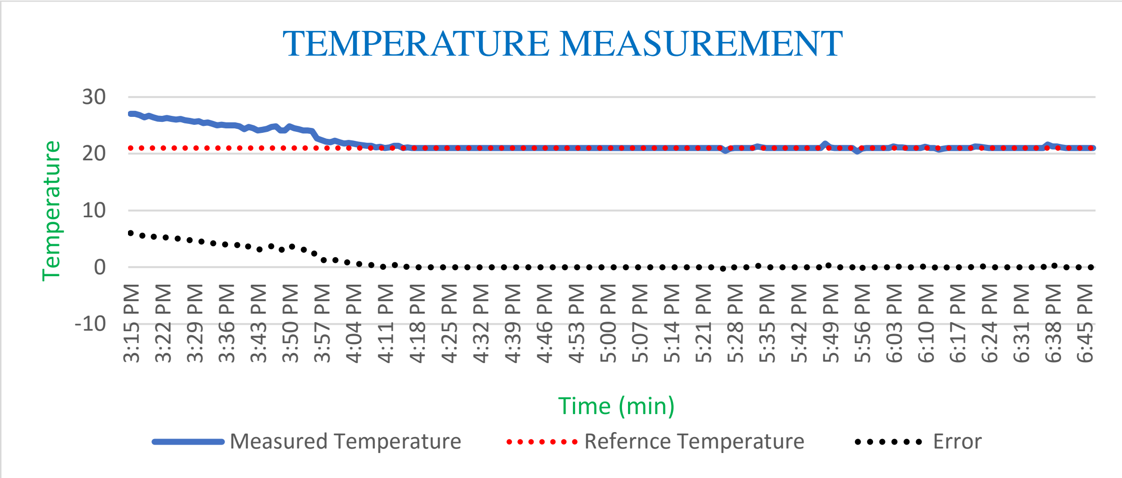

Figure 4.15 Indoor real measurement temperature.

As for this curve, the measured temperature values also appear in the nodes inside the chamber and are at the same time as the node measurements that are outside. The difference is clear because the work in this knot is much faster to reach the reference point, and the temperature is because the much lower disturbances so we see that reaching the degree required by the designer is captured and that the average error will also be much lower than the previous error, Where the mean error is 0.89342723.

CHAPTER FIVE
CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this thesis, a WSN based on the ZigBee protocol has been created. The presented WSN is utilized to monitor environmental humidity and temperature values. The most striking results that emerged to the system under different real-time testing scenarios and the results have been presented in chapter four. The results in chapter four manifested that the system efficiency had significantly improved after the addition of the PID controller to the client node. The main conclusions of the thesis are listed as follows.

1- The wireless client node with PID-controller provides high flexibility.

2- The main objective of this thesis was to make activities of data collection in WNS easier and convenient from the user’s requirements.

3- The proposed system had given full control for humidity, temperature, and the PID-controller in client nodes had provided additional stability for an actuator.

4- The WSN based on the mesh network that had presented in this thesis had included some features in the easy development and flexibility control.

REFERENCES

[1] Vilanova, R., Katebi, M. R., N. Wahab. (2011). N-removal on Wastewater Treatment Plants: A Process Control Approach, *J. Water Resour. Prot.*, vol. no. **3**, 1–11,.

[2] Poslad, S. (2011). *Ubiquitous computing: smart devices, environments and interactions*. John Wiley & Sons,.

[3] Farooq , M. O., Aziz , S., Dogar, A. B. (2010). State of the Art in Wireless Sensor Networks Operating Systems: a Survey in *International Conference on* Future Generation Information Technolog*y*, p. 616–631.

[4] Rodr, J. E., iguez Pallares. (2015). Wireless Sensor Network Implementation with Arduino and Xbee. Universitat Politècnica de Catalunya.

[5] “http://www.iec.ch/whitepaper/pdf/iecWP-internetofthings-LR-en.pdf. (2020).

[6] Souissi, R. , Ben-Ammar, M. (2014). An Intelligent Wireless Sensor Network Temperature Acquisition System With an FPGA. *Wirel. Sens. Netw.*, vol. **6**, no. 01, p. 1.

[7] Mosleh, M. F., Fayadh, R. A., Hamid, S. A. (2017). Design and Implementation of Wireless Sensor Network Based on MATLAB Interfaced with Arduino. no. August.

[8] Lee, J.-S., (2006). Performance Evaluation of IEEE 802.15. 4 for Low-Rate Wireless Personal Area Networks. *IEEE Trans. Consum. Electron.*, vol. **52**, no. 3, pp. 742–749.

[9] Arampatzis, T., ygeros, J. L., Manesis, S. (2005). A Survey of Applications of Wireless Sensors and Wireless Sensor Networks. in *Proceedings of the 2005* IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control., pp. 719–724.

[10] Oliver, J. (2019). Wireless SensorNetworks.Technology, Protocols and Applications. *Hilos Tensados*, vol. **1**, no., pp. 1–476.

[11] Cassero, S. (2016). Design and Analysis of Arduino, Raspberry Pi, and Xbee Based Wireless Sensor Networks. UC Santa Barbara.

[12] Sharma , S., Kumar , D., Kishore K. (2013). A Review on Topologies and Node Architecture. in International Journal of Computer Sciences AND Engineering, , vol. **1**, no. 2, pp. 19–25.

[13] Poellabauer, C. ,W. D. (2010). Fundamentals of Wireless Sensor Networks: Theory and Practice. *John Wiley Sons*,.

[14] Kiran Maraiya, N. G., Kamal Kant. (2011). Application based study on wireless sensor network," *Int. J. Comput. Appl.*, vol. **21**, pp. 9–15,.

[15] Pawlowski, A., Guzman J., Rodr F. iguez, M. Berenguel, J. Sánchez, and S. Dormido. (2009). Simulation of Greenhouse Climate Monitoring and Control with Wireless Sensor Network and Event-Based Control. *Sensors*, vol. **9**, no. 1, pp. 232–252.

[16] Rodríguez, F. (2002). Modeling and Hierarchical Control of Greenhouse Crop Production.

[17] Rodr, F.iguez., Guzmán J. L., Berenguel M., Arahal M. R., (2008) .Adaptive Hierarchical Control of Greenhouse Crop Production. *Int. J. Adapt. Control Signal Process.*, vol. **22**, no. 2, pp. 180–197.

[18] Bot, G. P. A. (2008). Greenhouse Climate: From Physical Processes to a Dynamic Model," Landbouwhogeschool te Wageningen, **2008**.

[19] Kit X. Z. M. Radio Frequency (RF) Modules.

[20] Faludi R., (2010). Building Wireless Sensor Networks: with ZigBee, XBee, Arduino, and Processing. *O'Reilly Media, Inc*.

[21] Souza , M., Arajo, P. J. P. (2012). Relay Methods and Process Reaction Curves: Practical Applications. in *Introduction to PID Controllers - Theory, Tuning and Application to Frontier Areas*, InTech.

[22] Dunfan, Y. Application of Wireless Sensor Networks in Environmental Monitoring.

[23] Odo, C. M. Wireless Sensor Networks for Environmental Monitoring Applications. Thesis.

[24] S. Management and N. Hülsemann. (2015). Wireless Sensor Network Implementation with Arduino and Xbee.

[25] Arduino. [Online] Available: http://www.arduino.cc/No Title.

[26] Goleman, A., daniel, boyatzis., Richard., Mckee and Perdana. Performance Analysis of XBee ZB Module Based Wireless Sensor Networks. *J. Chem. Inf. Model.*, vol. **53**, no. 9, pp. 1689–1699.

[27] Chen, C. L., (1991). A Closed-Loop Reaction-Curve Method for Controller Tuning. Chem. Eng. Commun., vol. **104**, no. 1–3, pp. 87–100.

[28] Huang, H.-P., Jeng, J.-C. (2005). Process Reaction Curve and Relay Methods Identification and PID Tuning," *PID Control*, no. Imc, pp. 297–337.

[29] Mohammad, S. (2014). A Low-Cost Wireless Sensor Network System Using Raspberry Pi and Arduino for Environmental Monitoring Applications.

[30] No Titlehttps://www.arduino.cc/en/uploads/Tutorial/pwm.gif.

APPENDIX
ARDUINO CODE

```
#include <PID_v1.h>
double Setpoint1, Input1, Output;  //not there in original sketch
//Define Variables we'll be connecting to double Setpoint, Input, Output;
//Define the aggressive and conservative Tuning Parameters
double aggKp = 32.4109201693411, aggKi = 60.8507933243211, aggKd =3.6487543055277;
double consKp = 75.170782563329923, consKi = 100.6192905016656, consKd =0.000909685940962655;
//double consKp=2, consKi=5, consKd=1;
#include "DHT.h"

#define DHTPIN 8
#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

//Specify the links and initial tuning parameters
PID myPID(&Input1, &Output, &Setpoint1, consKp, consKi, consKd,DIRECT );
void setup()
{
  Serial.begin(9600);
  //initialize the variables we're linked to
  // Input = map(analogRead(0), 0, 1023, 255, 0);

  //int val = analogRead(0);
  //val = map(val, 0, 1023, 0, 255);
  // analogWrite(9, val);
```

```
  dht.begin();
  Setpoint1 = 23.80;
  myPID.SetOutputLimits(0, 255); //MIN,MAX
  //turn the PID on
  myPID.SetMode(AUTOMATIC);  // MANUAL
  myPID.SetControllerDirection( REVERSE);
  // DIRECT
}
void loop()
{

 float sensorValue1 =dht.readTemperature();
  Input1 = sensorValue1;

  double gap = abs(Setpoint1 - Input1); //distance away from setpoint
  if (gap < 10)
  { //we're close to setpoint, use conservative tuning parameters
    myPID.SetTunings(consKp, consKi, consKd);
  }
  else
  {
    //we're far from setpoint, use aggressive tuning parameters
    myPID.SetTunings(aggKp, aggKi, aggKd);
  }

  myPID.Compute();

analogWrite(7,Output);
analogWrite(5,Output);
delay(20);
```

```
 Serial.print(" PWM2 = ");
 Serial.print(Output);
 Serial.print("      sensor value2 =  ");
 Serial.print(sensorValue1);
 Serial.print("      Input1 =");
 Serial.println(Input1);
 Serial.print("       Setpoint2 =");
  Serial.println( Setpoint1);
 delay(2000);
}
```

MATLAB CODE
REACTION CURVE

```
% step 1
clear all;close all;
clc;
%Define Process Transfer Function
GP = tf(1.935,[1 5.481 15.52]);
[y,t] = step(GP); %Obtain step input response
plot(t,y,'k');title('Process Reaction Curve ');
slope = gradient(y,t); %Calculate the slop at every point
%Find out Point of Inflection coordinates
[tslope,idx] = max(slope);
tIP = t(idx); %time
yIP = y(idx); % error
fprintf('\n Inflection coordinate- x = %d ------------\n',tIP);
fprintf('\n Inflection coordinate- y = %d ------------\n',yIP);
%Estimate tangent to the process reaction curve at Inflection point
yTangentLine = tslope*(t-tIP) + yIP;
%Plot Inflection point and Tangent line on process reaction curve
hold on
plot(tIP,yIP,'r*');
plot(t,yTangentLine,'b');
%ylim([0 1]);
%Estimate FOPTD parameters
L = tIP-(yIP/tslope);
Tau = y(end)/tslope;
```

```
K = y(end)/1;
fprintf('\n--------------- L = %d ------------\n',L);
fprintf('\n--------------- Tau = %d ------------\n',Tau);
fprintf('\n--------------- K = %d ------------\n',K);
plot(L,0,'g.','MarkerSize',15) %Show the delay time in plot
%Define Approximate transfer function and plot the step response
Gm = tf(K,[Tau 1],'InputDelay',L);
[z,t] = step(Gm);
plot(t,z,'k--')
hold off
grid on
xlabel('Time [sec]');
ylabel('Amplitude of error');
%axis([0 3 0 .5]);
legend('Actual Process','InflectionPoint',...
    'Tangent at IP','Dead Time','Approximated FOPTD','Location','best');
```