ANKARA YILDIRIM BEYAZIT UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

A SECURE CLUSTER-BASED WIRELESS SENSOR NETWORKS APPLICATIONS

PhD. Thesis by

ASMAA SALIH HAMMOODI

DEPARTMENT OF COMPUTER ENGINEERING

May, 2019

ANKARA

A SECURE CLUSTER-BASED WIRELESS SENSOR NETWORKS APPLICATIONS

A Thesis Submitted to

The Graduate School of Natural and Applied Sciences of Ankara Yıldırım Beyazıt University

In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Computer Engineering, Department of Computer Engineering

By

Asmaa Salih Hammoodi

May 2019

ANKARA

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"A SECURE CLUSTER-BASED WIRELESS SENSOR NETWORKS APPLICATIONS "** completed by **ASMAA SALIH HAMMOODI** under the supervision of **PROF. DR. FATIH V. ÇELEBI** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Fatih V. ÇELEBİ

Supervisor

Prof. Dr. Şahin Emrah	Prof. Dr. Suat Ozdemir
Jury Member	Jury Member
Assist.. Prof. Dr. Ozkan Kilic	Assist. Prof. Dr. Shafqat Rehman
Jury Member	Jury Member

Prof. Dr. Ergün ERASLAN

Director

Graduate School of Natural and Applied Sciences

ETHIAL DECLARATION

I hereby declare that, in this thesis, which has been prepared in accordance with the Thesis Writing Manual of Graduate School of Natural and Applied Sciences,

- All data, information and documents are obtained in the framework of academic and ethical rules,
- All information, documents and assessments are presented in accordance with scientific ethics and morals,
- All the materials that have been utilized are fully cited and referenced,
- No change has been made on the utilized materials,
- All the works presented are original,

And in any contrary case of above statements, I accept to renounce all my legal rights.

Date: **Signature:** **……..**

Name & Surname:…….

ACKNOWLEDGMENTS

In the Name of ALLAH, I would like to give thanks first and foremost to God, who has been my strength all my life. The Most Beneficent, the Most Merciful; I owe my deepest gratitude to ALLAH for His blessings, enabling me to complete this dissertation.

I would like to take some time to thank all the people without whom this project would never have been possible. Although there is only my name on the cover, many people have contributed to the research in their own way, and for that, I want to give them special thanks.

I am grateful to my supervisor Prof. Dr. Fatih V. Çelebi for his tremendous support and motivation during my study. His immense knowledge and precious recommendations constituted the milestones of this study. His guidance assisted me all the time of my research and while writing this thesis.

I also want to take a moment to thank my other committee members, Prof. Dr. Suat Özdemir and Dr. Shafqat Rehman. Thank you for investing time and providing interesting and valuable feedback. I feel proud and honoured that you have accepted a place on my committee.

Finally, a special thanks to my family. Words cannot express how grateful I am to my mother. I would also like to thank my husband. To my beloved children, I would like to express my thanks for all the sacrifices that you have made on my behalf.

MAY 2019 **ASMAA SALIH HAMMOODI**

A SECURE CLUSTER-BASED WIRELESS SENSOR NETWORKS APPLICATIONS

ABSTRACT

Recently, advances in wireless communication technologies and wireless sensor networks (WSNs) have become important in the field of computer and electrical engineering. Generally, WSNs include several actuators controlled by sensors to accomplish certain tasks. Consequently, WSNs have gained increasing importance in a variety of military and civilian applications, wherein sensors must interact with each other in highly dynamic scenarios. This challenge has motivated the research community to explore novel applications.

Two of the most important challenges concern the WSN. The first is due to the clustering mechanisms with hierarchical structures that are used to reduce energy consumption in sensor networks and enhance network performance, while the second issue involves securing the nodes using clusters. This work focuses on two essential platforms adopted in WSN applications. The first, the hardware platform, characterizes the WSN cluster nodes by specifying automatically encrypted nodes during routing. The second, the software platform, is a simulation that includes two features: one feature is identical to the hardware platform to enhance and compare results, while the second feature implements security by applying three encryption algorithms to secure cluster nodes.

Encryption was used in this study, and the security of the WSN clusters was tested with two security systems: Caesar cipher is classified under monoalphabetic and substitution, RC4 is stream cipher, and AES is the block cipher, RC4 and AES classified under symmetric secret key. The first goal was achieved with a private protocol that depended on identity-based cryptography via the hardware platform. The goal was to acquire a secure cluster node in a traditional and rapid manner by creating a private protocol, which is routed through the cluster using three ciphers, where each node was automatically encrypted during the transmission of a packet via a hop-by-hop approach.

Eventually, the system provided a rapid, efficient, flexible, secure, and authoritative infrastructure for a WSN, the offer on-demand employed for clusters of node sensors throughout domains and applications.

Keyword: WSN applications, cluster, AES algorithm, Caesar cipher, CR4 algorithm.

GRUP TABANLI KABLOSUZ SENSÖR AĞLARINDA GÜVENLİK ve MAHREMİYET

ÖZ

Kablosuz iletişim teknolojileri ve kablosuz sensör ağlarındaki gelişmeler son zamanlarda bilgisayar ve elektrik mühendisliği alanlarında önem kazanmıştır. Kablosuz sensör ağları genellikle belli başlı görevleri yerine getirmek için, alıcılarla kontrol edilen birçok aktüatör içerir. Sonuç olarak, kablosuz sensör ağları, alıcıların birbiriyle yüksek dinamik koşullarda etkileşim içinde olmasını gerektiren çeşitli askeri ve sivil uygulamalarda giderek önem kazanmaya başlamıştır. Bu gereklilik daha yeni uygulamalar keşfetme konusunda araştırma çevrelerini teşvik etmiştir.

Konuyla ilgili zorluklardan en önemli ikisi kablosuz sensör ağlarıyla ilgilidir. Birincisi, alıcı ağlarda enerji tüketimini azaltmak ve ağ gücünü en iyi düzeye çıkartmak için daha yüksek yapılarla olan kümeleme mekanizmasından kaynaklanmaktayken ikinci mesele devrelerin kümeleme yöntemiyle daha güvenli hale getirilmesidir. Bu çalışmanın ana hedefi kablosuz sensör ağlarındaki uygulamalarda kullanılan iki önemli platformdur. Birinci platform, yani donamım platformu, yönlendirme esnasında otomatik olarak şifrelenen devreleri belirleyerek kablosuz sensör ağlarındaki küme devrelerini tanımlar. İkincisi ise yazılım platformudur ve bu platform iki özellik içeren bir simülasyondur. Bu özelliklerden biri sonuçları geliştirmek ve karşılaştırmak amacıyla donanım platformuyla birebir aynıdır fakat ikinci özellik küme devrelerinin güvenliğini sağlamak için, 3 adet kodlama algoritması uygulayarak güvenliği sağlar. Bu çalışmada kodlama kullanılmış ve kablosuz sensör ağlarının güvenliği iki adet güvenlik sistemi kullanılarak test edilmiştir. Sezar şifreleme yöntemi tek alfabeli olarak ve yer değiştirme metodu kapsamında sınıflandırılır. RC4 akış şifreleme ve AES ise bir blok şifreleme algoritmasıdır. RC4 ile AES simetrik gizli anahtar olarak gruplandırılmıştır. İlk hedefe, donanım platformu vasıtasıyla kimlik denetimli şifrelemeye dayanan özel bir protokol ile ulaşıldı. Amaç, özel bir protokol yaratarak geleneksel ve hızlı bir şekilde güvenli kümeleme devresi oluşturmaktı. Bu protokol, veri akışı esnasında adım adım ilerleyen bir yaklaşım vasıtasıyla her devrenin otomatik olarak şifrelendiği üç adet şifreleme yöntemi kullanılarak küme içinde yönlendirilmiştir. Sonuç olarak, sistem

kablosuz sensör ağları için hızlı, etkili, değişken, güvenilir ve yetkili bir altyapı sağlamıştır ve talep üzerine nüfuz alanı ve uygulamalar boyunca devre sensörlerinden oluşan kümeler kullanılmıştır.

Anahtar kelimeler: WSN uygulama, AES algoritma, CR4 algoritma, sezar şifresi.

CONTENTS

ACRONYMS

SNC	Sensor Network and Configuration
WSN	Wireless Sensor Networks
MEMS	Micro Electro-Mechanical System
SEP	Stable Election Protocol
P-SEP	Prolong-SEP
CH	Cluster Head
CC	Cooperative Communication
ACACP	A region Scopes Mindful Clustering Protocol
ADCC	Appropriate Duty Cycle Control
CDS	Connected Dominating Set
PEGASIS	Power-Efficient Gathering in Sensor Information Systems
QoS	Quality-of-Service
ADC	Analog-to-Digital Converter
COTS	Commercial-Off-The-Shelf
OSI	Open System Interconnection
LLC	Logical Link Control
MAC	Media Access Control
TDMA	Time-Division Multiple Access
RFID	Radio Frequency Identification
RF	Radio Frequency
BLE	Bluetooth Low Energy
LEACH	low-Energy Adaptive Clustering Hierarchy
TEEN	Threshold-Sensitive Energy-Efficient Sensor Network
APTEEN	Adaptive Threshold-Sensitive Energy-Efficient Sensor Network
NTP	Network Time Protocol
DoS	Denial of Service
MAC	Message Authentication Code
ACQUIRE	Active Query Forwarding In Sensor Networks
RR	Rumour Routing
GAF	Geographic Adaptive Fidelity
GEAR	Geographical and Energy Aware Routing
SPIN	Sensor Protocols for Information via Negotiation
AC	Address-Centric Protocol
DC	Data-Centric Protocol

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Recent advances in micro-electromechanical systems (MEMS) technology, digital electronics, and wireless communications have focused on the design and development of low cost, low power, multifunctional sensor nodes that are small, and communicate wirelessly over short distances [1] [2]. A wireless sensor network (WSN) is constructed from many smart devices called nodes, which are spatially distributed to perform a global, application-oriented task. The primary component of the network is the sensor, which is essential for monitoring physical variables such as temperature, humidity, the presence (or absence) of something, sound, intensity, vibration, pressure, motion, and pollutants, among others, in different locations [1]. The ever-increasing capabilities of these small sensor nodes, which include data processing, sensing, and communication, enable the realization of WSNs based on collaboration between many sensor nodes. The essential design considerations for a standard sensor network are energy efficiency, memory, computational speed, and bandwidth. A smart device consists of a radio transmitter, an energy source, and a microcontroller. A central computer is sometimes included to manage the network. Regardless of the task, a sensor network essentially performs sensing, computation, and communication by employing software, hardware, and algorithms [1]. Wireless communication and electronics have made it possible to organize multi-hop WSNs for various self-configurable applications, allowing an ad-hoc network to organize itself [2] [3].

Consequently, the general description of a WSN is a collection of many self-powered, small sensing nodes that gather information or detect special actions and connect wirelessly with the objective of relaying their processed data to a base station (BS) [4].

The main goals of a WSN are as follows:

- Estimate physical measures for a given area
- Detect specific events, and estimate parameters for the detected event
- Categorize detected objects

- Track objects [1]

Figure 1.1 Wireless sensor.

We designed and implemented first platform real tested bed (hardware platform), including eight WSN nodes, where each node was identified with a unique name. Consequently, a cluster of these nodes was created via handshake communication to link each node with its nearest-neighbour node. The goal was to build a secure node cluster in a traditional and rapid manner by creating a private protocol that routed through the cluster. A Caesar cipher was used for encryption at each node during hop-by-hop packet transmission. The aim of the secure authentication technique was to avoid the threat of node capture attacks in an easy and flexible manner. The second platform is a simulation platform created to measure the important parameters to give the opportunity for comparing them with the practical side, where the software allowing us to measure the results accurately as well there is no possibility of error. Simulations were run to calculate the time required to perform various tasks using the two platforms. We found that the time required to perform certain tasks decreased when a greater number of chains were used. However, the number of chains should be kept low to prevent data from aggregating due to overlap. The objective of the secure authentication technique was to identify security threats from node capture attacks (i.e., capture of a sensor node through a physical attack) [5].

1.1 Motivation and Related Work

Unfortunately, constructing a routing protocol for a WSN is not an easy task because of the complexities inherent in this type of network, as well as the parameters and properties that need to be considered. Many nodes with sensing and wireless

communication capabilities drive an unlimited number of applications, thus, necessary measures must be taken due to the energy and task execution time required, and proper algorithms must be chosen for use in clustering, especially considering the tasks being deployed in areas that could benefit from WSNs. A modified stable election protocol (SEP), called prolong-SEP (P-SEP) was proposed in the literature to prolong the stability period of fog-supported sensor networks through balanced energy utilization [6]. The P-SEP ensures that nodes are uniformly distributed, which prolongs the stability period of the system and ensures new cluster head (CH) selection policies, especially prior to nodal failure. P-SEP considers two levels of node heterogeneities: advanced and normal nodes. The proposed approach was evaluated by varying most of the network parameters on a simulation platform and comparing them with other current cluster-based routing protocols. Another study [7] focused on cooperative communication (CC) with WSNs that used cluster node protocols. It was shown that arrangements of network terminals could manage the issues that influence the effective activity of machine-to-machine (M2M) communication systems. Similarly, another proposal was presented [8] in which the major objective was to select an ideal number of active sensors (considering lingering energy and the cover set) and to maintain uptime. Moreover, A regional scope-oriented clustering protocol (ACACP The Advisory Committee for Aviation Consumer Protection) technique with improved energy utilization was proposed. Another study proposed an appropriate duty cycle control (ADCC) schema that minimized connection delays and complexities to improve energy efficiency in connected dominating set (CDS)-based WSNs [9].

1.2 Strategy Formulation and Block Diagrams

Our study includes many techniques for implementing this project. Two possible strategies are summarized in hardware and software block diagrams. The first block diagram describes wireless communication between 8 WSN nodes to design one chain by means of handshaking, as shown in Figure 1.2. Figure 1.3 shows the PEGASIS algorithm with two chains and eight WSN nodes with a BS.

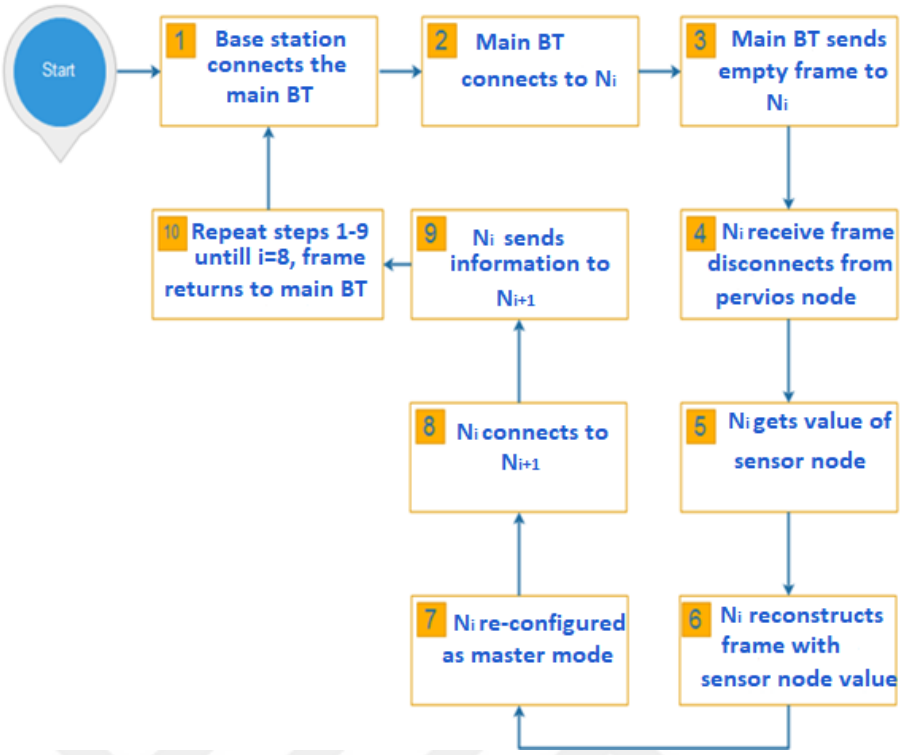

Figure 1.2 Handshake communication on hardware platform.

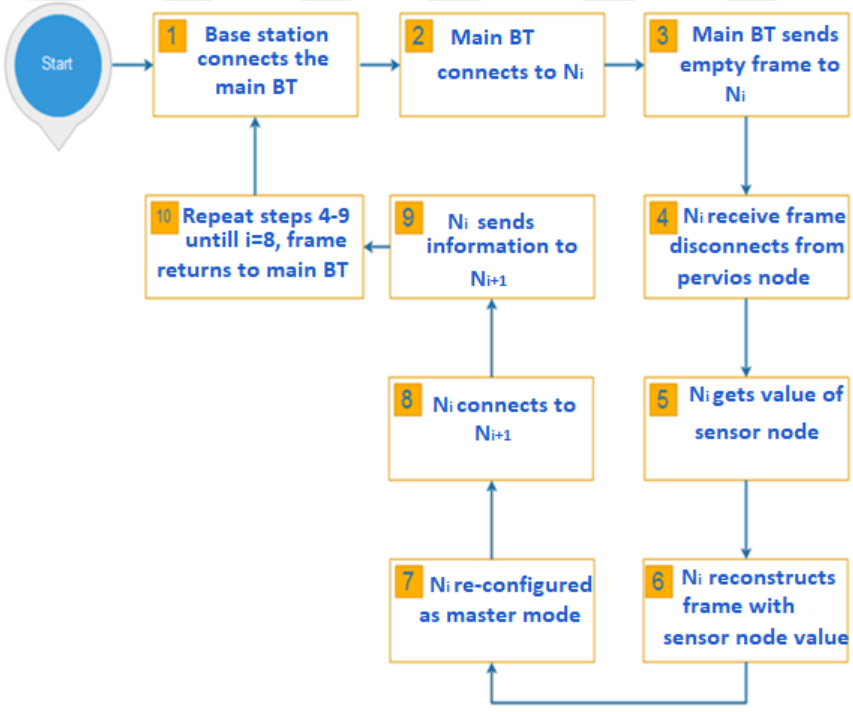

Figure 1.3 PEGASIS algorithm implemented on Software Platform

1.3 Enquiries and Research Questions

- A review of related research and studies on algorithms used in WSN nodes raises the following questions:
- How does the proposed protocol improve on the current approach?
- The answer is in our structure; the use of the Caesar cipher in this protocol provides a rapid approach and easy design on a hardware platform. How are algorithms in current and future WSN nodes used?
- We first focus on PEGASIS chain implementation in cluster WSN nodes that communicate with each other via handshaking in Bluetooth master/slave modes. Moreover, we examine the effects of increasing the number of chains. Secondly, we improve the security with three encrypted algorithms. We also examine the effects of increasing the number of chains. What is the best practical method for validating the model?
- The best method undertaken in the current work is to track two platforms, one of which is validated during design. The other is evaluated based on measurement, while taking into consideration the relationships between constituents, finding the optimum parameters values, and comparing these optimized parameters with real parameters, practically and theoretically.
- What is the problem with current approaches?
- This work presents the problems associated with different modelling techniques for software and hardware platforms. These problems should be considered when obtaining a suitable model, while accounting for modelling error. Furthermore, if we use additional nodes in the prototype designed with eight WSN nodes, we would need additional memory for execution, but this is possible in application.
- Does the application provide enough security for cluster nodes?
- The desired architecture must be flexible enough for use in diverse WSN applications. Additionally, the model was tested using brute-force attempts to break the code, we found that needs unlimited time for breaking codes.

1.4 Research Objectives and Scope

Although the questions and answers listed above were reached using block diagrams, in this thesis, the following tasks are discussed:

- Proposal and use of a protocol model that applies to any type of sensor network application;
- Use of encryption algorithms during routing;
- Proposal of a model that can address problems related to clustering, reliability, cryptography, and data aggregation;
- Offering the best approach for creating an encryption protocol through WSN node clusters in the overall model.

1.5 Justification

The proposed model will be evaluated and justified with the algorithms in this research as follows:

i. Examining and extracting the parameters (the energy required for transmitting/receiving and task execution time) via modern algorithms to achieve better results in WSNs through hardware and software approaches.

ii. The accuracy of the model will be tested by assigning the same values of the previous parameters for each WSN node to those in the hardware and simulator.

iii. After the model is evaluated, it can be used to develop methods to reduce the time required to transmit packets through the cluster. In addition, it can be used to develop algorithms to secure a WSN cluster. Other encryption algorithms could be used in the model in the future.

1.6 Thesis Contribution

The critical issue facing the spread of WSNs is the delivery of energy. In the current work, the final aim is to develop an advanced structure for wireless sensor techniques as follows:

- Clustering WSN nodes, the components of a clustered WSN is; Sensor node , Clusters (Clusters are the hierarchical units for WSNs), Cluster Head (CHs are the leader of a cluster), Base station (The BS provides the communication link between the sensor network and the end-user) and End user(The data in a sensor network can be used for a wide-range of applications where handshaking is used to communicate between neighbouring nodes).
- Hardware platform implementation, Creating an encrypted routing protocol in sensor nodes.
- Optimizing parameters using the simulation platform to obtain accurate results so that we can compare the results through the algorithms executed to find the best algorithm in terms of the parameters.

1.7 Thesis Structure

Chapter 2 briefly summarizes the background of WSN characteristics and requirements, sensor architecture, and WSN applications. Chapter 3 describes state-of-the-art hardware and software architectures used in major platforms. Chapter 4 presents simulation results with measurement tables of energies and time, which are used for comparisons for hardware and software in terms of application and implementation, as well to optimize system security. Chapter 5 summarizes the main conclusions and suggests avenues for further research.

CHAPTER 2

WSN BACKGROUND AND LITERATURE REVIEW

WSNs are a focal subject in computer and electronic engineering. A WSN contains an enormous number of sensor nodes that must communicate, process data, and gather data from sensors to perform specific tasks. They should be able to detect events, which is obviously an outstanding characteristic in many WSN applications.

The unique characteristic of WSNs is that they contain numerous small nodes that are organized randomly or according to a statistical distribution, depending on the topography of the region in which they are deployed. A sensor node has resource constraints, such as limited signal processing capabilities, low power supply, limited communication, computational functionality, and a small amount of memory. Hence, it can sense only limited information about its location. However, a collection of sensor nodes can produce superior results when they cooperate. Two major advantages of WSNs are their low deployment cost and their independence in querying [10].

In addition, WSNs can be used for continuous information sensing and storing, and the provision of content delivery services. WSNs have attracted a great deal of attention recently thanks to their broad civilian and military applications [11-19].

Figure 2.1 Different sensors and their applications.

WSNs use media allocation within many ad-hoc networks, and they do not provide inclusive correspondence to avert traffic collisions [20]. Regardless of their major functions, a sensor network essentially performs three major tasks: communicating, computation, and sensing; this procedure is executed using three main mechanisms: algorithms, software, and hardware, respectively [1].

2.1 WSN Requirements

The operating requirements and capabilities of WSNs are as follows:

- Self-organization capability
- Cooperative signal processing
- Querying ability
- Limited lifetime
- Large network size, which makes it difficult to control
- Minimized faults
- Dynamic topologies and severe environmental conditions
- Quality of service (QoS)

- Data redundancy
- Large-scale distribution and ad hoc architecture
- Security
- Utilization of several sensors, where real sensor networks are designed based on the following criteria, depending on the communication protocols:
- Low energy consumption: Communication between the sink and the sensors must require low energy because this connection process consumes the most energy. This is one requirement that must be met if the lifetime of the sensors is to be increased.
- Multi-hop connectivity: Ordinarily, sensors avoid direct communication with the sink, because the energy consumed is commensurate with the square of the transmission distance. It is preferable to use other devices as communication hops.
- Scalability: Communication protocols must be authenticated to establish and maintain connectivity among sensors. In addition, protocols must be flexible when the network grows.
- Reliability: data transmission must be reliable in terms reducing the number of lost packets. [19,21-26].

2.2 WSN Characteristics

Deployment of WSNs is increasing rapidly but still faces many limitations, particularly related to limited battery life. In WSN applications, the energy consumed when transmitting a message is double the energy consumed when receiving the same message [21].

Consequently, WSNs must be characterized accurately. The wireless network may include a mobile communications network, a Bluetooth (BT) network, a wireless LAN, an ad hoc network, and so on. Like an ad hoc network, a sensor network has many characteristics, such as switching character, mobility, and limited battery power. The significant characteristics of WSNs are described as follows [27]:

- Communication capabilities: The sensor network's communication bandwidth is variable and narrow, and the limit of its transmitter distance ranges from 10^1 to 10^2 m. Moreover, it is limited in processing speed and storage capacity. It is difficult to keep a WSN running because the sensors are easily influenced by environmental factors such as buildings, mountains, storms, rain, and lighting. Therefore, any software or hardware for a WSN must be sufficiently strong and fault-tolerant.
- A WSN is specified as being self-configured, and its infrastructure is minimal in a network used for physical or environmental monitoring
- A WSN node contains a power supply, sensors, computing devices, and transceivers. Nodes communicate with each other using radio signals.

This requires numerous adjustments in network divisions and network topology. Regardless of the network divisions, portable nodes are useful for transporting data, as the nodes can be moved physically. Consequently, one of the typical issues in a WSN is correspondence failure.

2.3 Architecture of WSNs

WSN architecture contains many sensor nodes that are small, inexpensive, and use limited energy.

The sensor is the core of a WSN. A general hardware architecture for a WSN is shown in Figure 2.2. The design is partitioned into four sections: sensing unit, processing unit, communication unit, and power supply unit. [28].

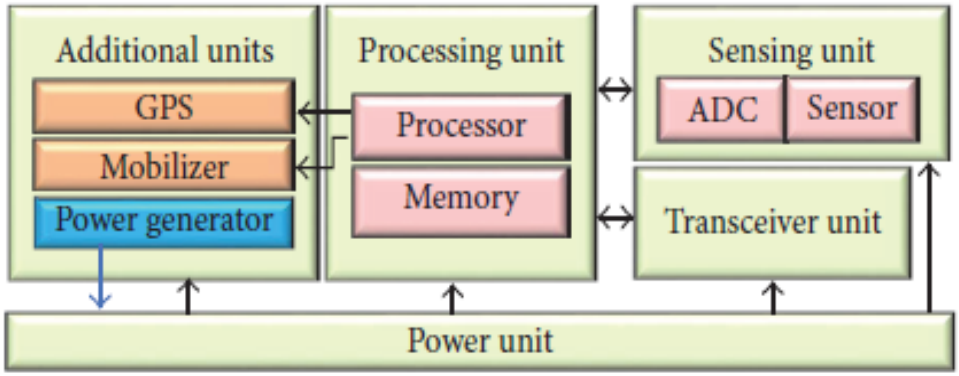

Figure 2.2 Sensor node architecture.

A sensing unit normally contains different types of sensors for quantifying physical changes as electrical signals, which can be examined with a processing unit. At that point, information processing becomes possible (e.g. information encoding). The processing unit also controls the node. The communication unit builds wireless associations among various nodes, while the power supply unit provides power to each node. The Mica Group (The mica group of sheet silicate (phyllosilicate) minerals includes several closely related materials having nearly perfect basal cleavage) mirrors the underlying generic elements in WSN applications, including processing, communication, and storage [29]. MICA2 mote was created by UC Berkeley and launched by Crossbow Tech Inc. in 2002. The MICA2 node can be used for a wide range of sensors (e.g., light, temperature, attraction, and acoustics), and the Atmel ATmega128L MCU [30] is used to associate transceivers and sensors [31].

2.4 WSN Layers

WSN communications are tracking the essential open system interconnection (OSI) layers. WSN represents five out of the seven layers in the OSI framework. These are presented in the following sections.

2.4.1 Physical Layer

The basic layer of the network is the physical layer, which includes networking hardware. This layer functions as an electrical and mechanical interface to the transmission medium and oversees signals and communication media. In OSI

engineering, the physical layer interprets the logical address that connects data link layers to various tasks, as shown in Figure 2.3.

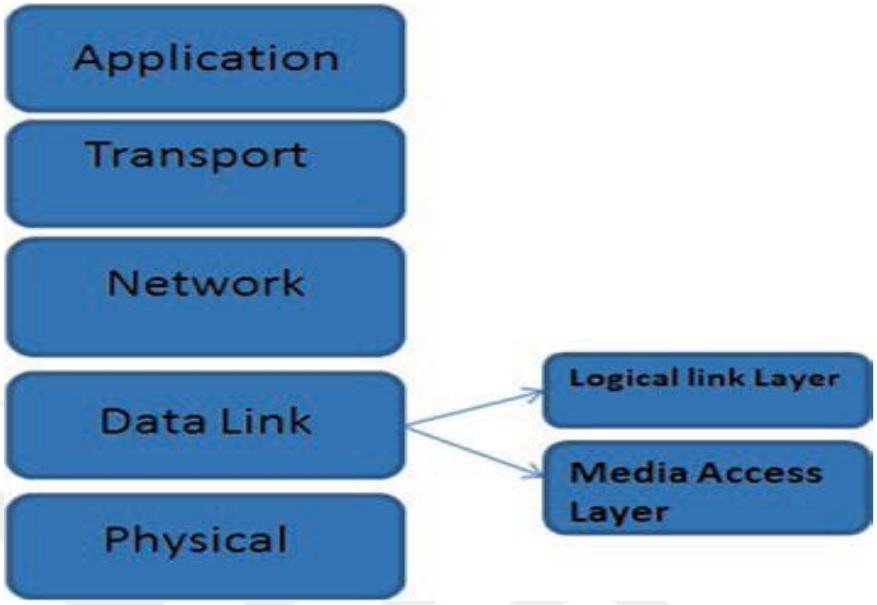

Figure 2.3 Architecture of a WSN protocol.

2.2.2 Data Link Layer

The data link layer is the second layer in the OSI. The responsibility of this class is to specify physical addressing and provide resources for linking between systems. In addition, it detects faults in the physical layer and attempts to resolve them. Encoding and decoding data into bits are the essential functions of this layer. The data link layer is split into the following:

- **Logical link control (LLC):** The responsibility of this sublayer is frame (F) organization and error inspection. It provides a multiplexing system that transports data over a single medium.
- **Media access control (MAC):** This layer oversees addressing and channel access to control components.

2.4.3 Network Layer

The network layer is used for logical addressing within the infrastructure in a virtual circuit. It is used to transmit information between nodes and determines which path this information should travel. The network layer uses tracking, switching, and routing

technologies. A major functionality of this layer is error detection in packet management, congestion control, sequencing, and addressing. This layer also ensures quality when serving queries in the transport layer.

2.4.4 Transport Layer

The transport layer is delivering and confirmation the transmission of data and authoritative data service to the next layer. This layer provides verification of successful data transmission as well.

2.4.5 Application Layer

This layer is defined as a user interface layer. The application layer oversees information display and traffic administration and provides programming to various applications that transpose information to an understandable format [21].

2.5 WSN Applications

In recent decades, new WSN applications have emerged from the rapid developments resulting from intelligent networks. WSNs are relevant to many tactical applications, which include:

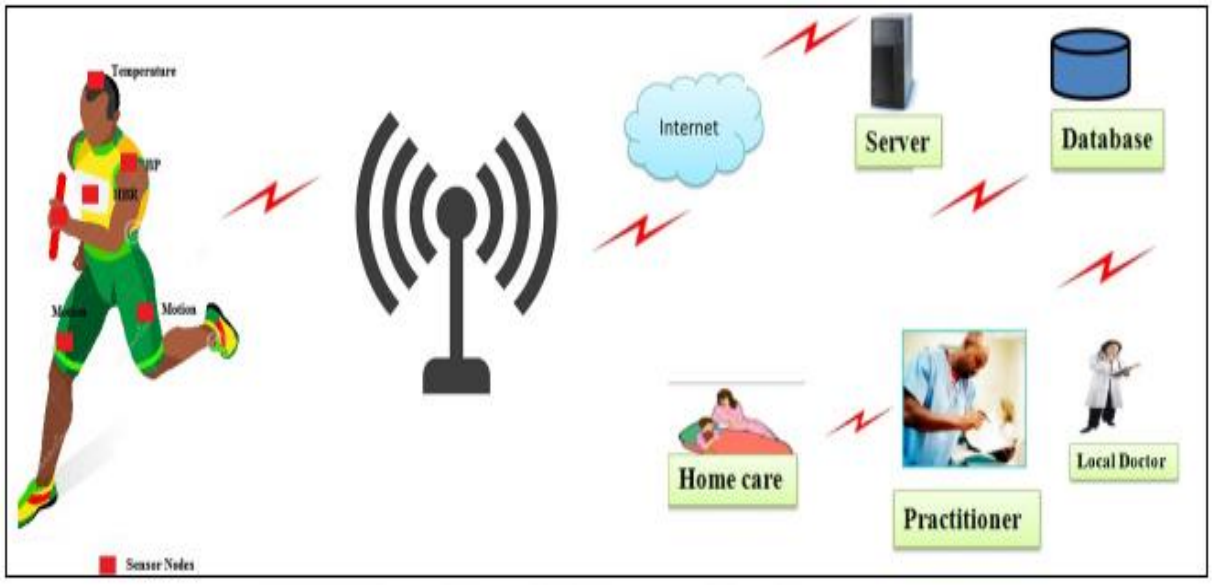

Figure 2.4 Example applications of WSNs.

Military applications

- Monitoring friendly forces and their equipment;
- Monitoring enemy forces;
- Detection of biological, chemical, and nuclear attacks;
- Battle damage assessment;
- Military-theatre or battlefield surveillance;
- Targeting

Environmental monitoring

- Disaster monitoring;
- Precision agriculture;
- Air or water quality monitoring;
- Hazard monitoring;
- Habitat monitoring
- Microclimates
- Forest fire detection
- Flood detection

Home intelligence

- Remote metering
- Instrumented environment
- Smart homes
- Automated meter reading
- Cross-layer design
- Home automation

Healthcare applications

- Behaviour monitoring
- Medical monitoring
- Drug administration
- Monitoring and tracking the doctors and patients inside a hospital
- Remote monitoring of physiological data

- Elderly assistance

Commercial applications

- Vehicle tracking and detection
- Environmental control at industrial and office buildings
- Inventory control
- Traffic flow surveillance [32, 33]

2.6 Security in WSNs

One of the most important challenges regarding security for WSNs is the protection of data and equipment against attacks and misbehaviour. WSN security requirements include:

2.6.1 Data Confidentiality

Confidentiality universally refers to concealing data and preventing unauthorized persons from accessing that data, often by means of numerous applications via nodes that convey extremely critical data. A WSN node should not reveal data to nearby networks. The preferred method for maintaining sensitive data reliability in a WSN is cryptography with a secret key that only exists within the network. Public key cryptography is very expensive in resource-restricted sensor networks. The proposed protocols mostly use symmetric key encryption procedures. Symmetric key encryption with a key distribution mechanism is extremely robust compared to asymmetric encryption.

2.6.2 Authentication

Authentication guarantees dependability when identifying the original message. In a WSN, validation must address prerequisites, such as confirming that any nodes moved from bundles are verified to have originated from a real node. Authentication requires that two sides have a secret key determining the message validation code (MVC). A recipient confirms authentication of the received message using a MAC key.

2.6.3 Integrity

Integrity refers to protecting data from any unauthorized changes.

2.6.4 Availability

Availability guarantees that data can be retrieved on demand. There are numerous dangers that could interrupt data accessibility, including sensor node capture and denial of service attacks [34].

2.6.5 Time synchronization

One significant requirement for sensor networks is time synchronization. Any security component or sensor in a WSN ought to be synchronized in time.

2.6.6 Secure localization

The location of a node within a WSN determines its security requirements.

2.6.7 Self-Organization

An individual node in a WSN can be characterized as self-healing and self-organizing. This characteristic of a WSN poses a major security risk. Dynamic WSN nodes occasionally have difficultly sharing a pre-introduced key between nodes and the BS [35- 37].

2.7 Routing Protocols in a WSN

Routing is the procedure of determining a path from the source to the destination when an information transmission request is made. In WSN routing, the network layer is the layer on which all data routing is dependent. Moreover, a routing algorithm employs the logic used to decide where the connection is to be made for each exchanged packet. Routing algorithms are characterized as follows:

- ***Static:*** Routing choices, network topology, or traffic loads are constant.
- ***Dynamic:*** Routing choices are contingent upon network topology and traffic loads [22].

Assorted steering conventions have been proposed to cover the inadequacies of the response-imperative nature of WSNs. Figure 2.5 shows the organization of WSN routing protocols. These routing protocols are partitioned into two general classifications: network architecture-based and operation-based [38].

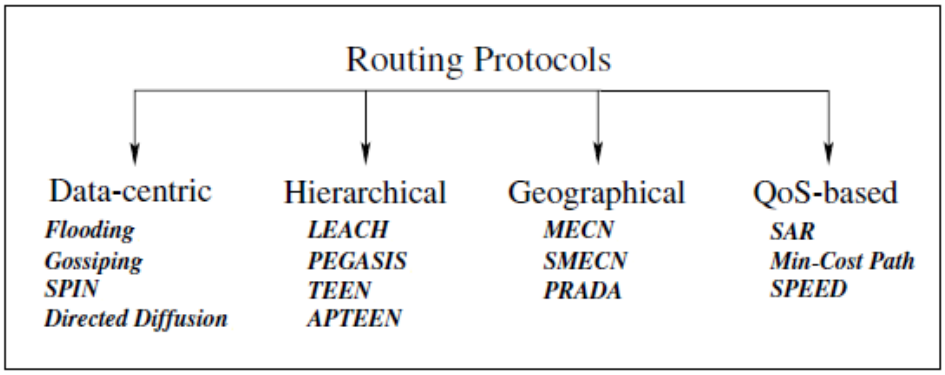

Figure 2.5 Overview of routing protocols.

Architecture-based routing protocols are partitioned into three subcategories based on their functionalities:

- Hierarchical-based routing
- Flat-based routing
- Location-based routing

Operation-based routing protocols are also categorized based on their functionalities:

- Negotiation-based routing
- Query-based routing
- Coherent routing
- Multipath routing protocols
- Quality of services (QoS)-based routing

Many challenges are encountered during routing in a WSN where the major goal of routing protocols is distributing data between the sink and the sensors. Energy consumption is the chief barrier preventing the expansion of any WSN routing protocol. As a result, information should be distributed in the most energy-efficient manner without sacrificing accuracy. Numerous traditional routing approaches which transmit data along a direct route may not be appropriate. The primary effects can be classified according to the energy consumption during guidance: the first step is neighbourhood discovery where the data-centric protocol might require locating every sensor in the network. In each situation, the nodes consume energy by trading information throughout the network. To decrease energy consumption in various routing protocols, local data exchange should be reduced without sacrificing accuracy.

Second, Communication and Computation It is worth mentioning that computation is frequently less energy-intensive than connection. In a WSN, the aim is to distribute data rather than single packets. Thus, harmonization with traditional packet exchange procedures requires that computation occur alongside routing to reduce energy consumption. Correspondingly, computation can be performed at every node without requiring redundancy.

2.7.1 Scalability

Scalability in a network is another imperative field of research. Time-division multiple access (TDMA) tables must have the capacity to accommodate high node densities. Monitoring physical phenomena in detail requires high-quality association among nodes. Completely comprehensive protocols that work with constrained topology learning should be used to provide scalability. In addition, a high-level data forwarding convention should be used to support the arrangement by aggregating information from nodes in the network without increasing energy consumption.

2.7.2 Addressing

The presence of a great number of sensor nodes makes it difficult to use a unique addressing system. Nevertheless, local address techniques can be used to create connections between neighbours. Address-based routing protocols are infeasible due to the considerable overhead required. Users are concerned with gathering data from various sensors instead of information from independent sensors.

2.7.3 Robustness

Routing protocols should be robust against any event happening in the node, as lost packets will be caused routing protocol failure. The routing protocol must provide successful delivery between the sink and the nodes under any condition.

2.7.4 WSN Topology

Many difficult-to-reach sensor nodes in wireless networks make topology monitoring and maintenance challenging. One significant task is arranging sensor nodes in the field such that they can be easily observed. In WSNs, topologies indicate the location

of nodes and their operational status. Some techniques for topology management are shown in Figure 2.6 [36].

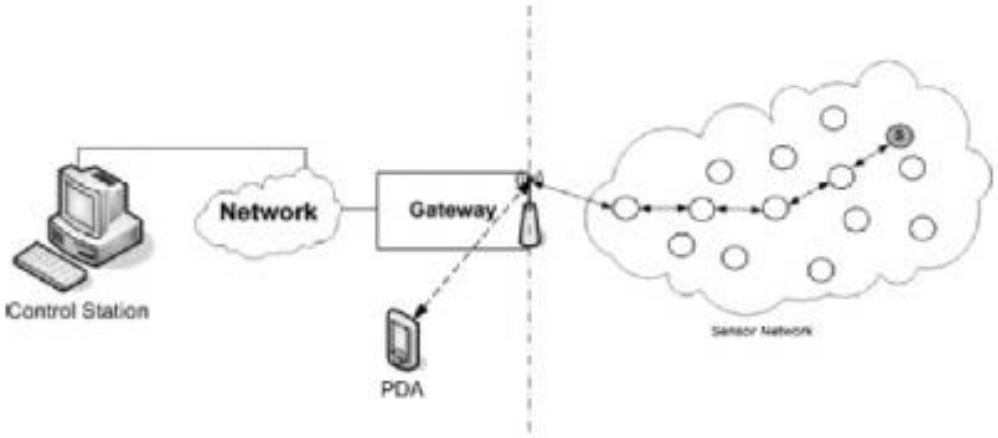

Figure 2.6 Typical Topology of a WSN [1].

Similarly, topology conservation is essential after building the underlying arrangement. After sending data, then conventional criteria for tasks can be received by the network topology. Generally, establishing many nodes requires topology upkeep. Topology maintenance and changes are described with the three following phases:

- **Pre-Deployment and Deployment Phase**

Sensor nodes can be organized in the sensor network either in a cluster or one by one. Despite the sheer number of sensors and their arrangement, the plans for sending a response from a sensor for an event must decrease the cost of establishing the sending plan, also

- Remove requests during the processing of the pre-deployment and the deployment phase
- Increase the adaptability of the plan; and
- Enhance self-association and adaptation to internal failure

- **Post-Deployment Phase**

During the organization stage, the topology may shift due to variations in the sensor state. In portable WSNs, the development of sensors influences the topology of the system. If necessary, the topology may change over a long distance. Furthermore, the availability of nodes may change due to noise, obstruction, commotion, or moving impediments. These variables likewise influence the topology of the system for a brief period. Subsequently, networking protocol conventions ought to have the capacity to adjust to these transient, occasional, and long-term topology changes.

- **Deployment of Additional Nodes**

Various types of topologies may require that extra nodes be deployed if availability and adaptation to internal network failure are influenced by the topology changes. Adding more nodes requires revamping the system. Adapting to recurrent topology fluctuations in an ad hoc organization with many nodes and exceptionally severe power limitations necessitates extraordinary protocols. [37].

2.7.5 Application

Suitably, static routing can be used to maintain transmission of the information for the lifetime of the network. In event-based applications, the sensor sleeps during operation. However, when an event occurs, a priority route is generated to transmit data regarding the event. The routing procedure is specifically identified with an application, and exceptional systems may be required for various applications [37].

2.8 Advantages and Disadvantages of WSNs

The most prominent advantages and disadvantages of WSNs are briefly presented as follows:

Advantages of WSNs

- Scalability is very straightforward due to the minimal cost and small size of sensor nodes.

- Generally, the sensor network protocols, algorithms, and self-organizing abilities are planned and communicated to create a WSN node. Thus, the entire network consists of low-cost nodes that are resistant to failure and topology variations.
- Sensor nodes can transmit data in harsh conditions to complete certain assignments without human intervention.

Disadvantages of WSNs

- Limited Energy and low Battery cause the applications of WSN are constrained due to restricted handling energy. As well the battery lifetime is low, and nodes are need charging in regular interval.
- Short correspondence ranges can lead to energy losses, thus multi-hop connections are essential for information transmission between the source node and the sink. The restricted energy capacity lead to unexpected changes in information handling capacity that require intensive processing because energy may be consumed rapidly and render the network non-functional for a given assignment.
- Constrained energy also implies that multi-hop connections cannot be maintained for a long period of time. Meanwhile, sensors nodes are utilized in numerous applications over broad areas, thus regularly changing or energizing batteries is unacceptable. Energy efficient sensors are important under such conditions [21, 38].

2.9 Communication Types Used with WSNs

2.9.1 Wired Communication

Wired information transmission is used for phone networks, satellite TV, web access, and fibre-optic communication. Neighbourhood phone organization regularly shapes wired communication utilized by both private and business clients in a region. Most networks today depend on fibre-optic communication. Fibre optics can be used to transmit more signals than traditional copper wiring utilized in the past while maintaining signal strength over larger distances. Figure 2.5 shows an Ethernet module

that is designed for connection to an Arduino board, and an Ethernet-powered Arduino board.

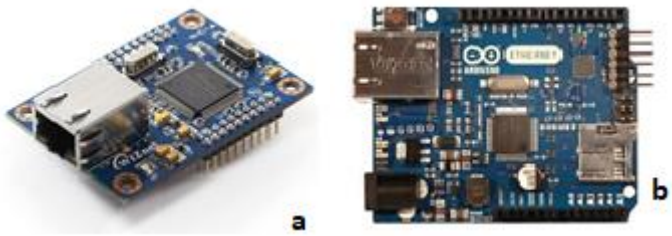

Figure 2.7 (a) Ethernet module that can be associated with any sensor; (b) Arduino board with installed Ethernet interface.

2.9.2 Wireless Communication Technologies

Wireless communication technology has become indispensable, as it enables clients to access even the most remote regions. The modules selected are described by their interfaces, which are uniquely designed to connect with microcontrollers, in this work we used an Arduino's open-source prototyping capabilities.

- **RFID**

Radio frequency identification (RFID) uses the electromagnetic fields to automatically identify and track tags attached to things. Track and trace applications are long range or vicinity applications. It is the most widely recognized innovation behind resource tracking and distinguishing objects (e.g. in programmed toll accumulation). RFID tags are categorized as passive, active, and battery-assisted passive (BAP).

Figure 2.8 RFID module and its receiving wire, which can connect directly to the Arduino via a serial protocol.

Most RFID tags contain a coordinated circuit for storing and handling data, as well as fundamental radio frequency (RF) segments for transmitting data remotely and

receiving wirelessly. RFID was first described as empowering correspondence control for the Internet of Things (IoT) because of its minimal effort, high portability, and usefulness for identifying objects and devices.

- **WiMAX**

WiMAX is a wireless broadband framework that offers quick web surfing without association through a link or DSL. WiMAX will blanket a radius of 30 miles (50 km) with wireless access. The increased range is due to the frequencies used and the power of the transmitter. Although WiMAX offers information rates greater than 30 Mbps, suppliers offer normal information rates of 6 Mbps and frequently less, thus it operates more slowly than hard-wired broadband. The actual cost of information access via WiMAX generally depends on the distance from the transmitter. WiMAX is one form of 4G remote access in telephones, such as Sprint's 4G.

- **Wi-Fi**

Wi-Fi is a type of medium wireless communication protocol used in numerous electronic devices, such as PCs, frameworks, and smart phones. A typical wireless router in an indoor point-to-multipoint arrangement using 802.11n and a stock antenna might have a range of 50 meters (160 ft) or less. In Wi-Fi, a wireless router forms the communication centre as shown in Figure 2.9. Mobile applications, Business applications, Home applications, Computerized application, Automotive segment, Browsing internet and Video conference are applications of Wi-Fi. These networks are to a great degree constrained due to the low transmission intensity, allowing clients to interface only within the vicinity of a router or signal repeater. Wi-Fi is a basic in-home network application that provides mobility without any links.

Figure 2.9 Pluggable XBee Wi-Fi module.

- **ZigBee**

ZigBee is a wireless transmission standard intended for low cost, low power, and easy-to-control networks and designed based on IEEE 802.15.4 specification with range of 10 to 100 meters (line of sight). ZigBee can be used anywhere, and it has been used to transfer data over simple networks. Zigbee is typically used in low data rate applications that require long battery life

ZigBee is one of the most recent and most progressive wireless advances being widely incorporated in home automation and smart technologies worldwide. The ZigBee standard what is grouped at 2.4 GHz, 900 MHz, and 868 MHz. ZigBee applications are Smart home applications, Healthcare devices and applications, Smart energy management, hybrid vehicle charging, programmable communicating thermostats etc…, electronics shopping tags, Food safety monitoring and Enhanced wireless devices to improve shopping experience

Nevertheless, it still requires a gateway with Internet availability (e.g. a PC workstation) that can forward data between the Internet and the ZigBee module.

Figure 2.10 Pluggable ZigBee module (XBee).

- **RF Links**

Another alternative for associating devices and broadcasting is the basic RF interface. These are inexpensive and small (ideal size) and can communicate at ranges between 100 m and 1 km (contingent upon the transmission power and antenna).

RF broadcasting modules (as shown in Figure 2.11) connect to microcontrollers and other devices using serial ports. Their disadvantages include low information rates (up to 1 Mpbs) and a required Internet-empowered gateway that provides devices with access to IoT devices.

Figure 2.11 RF transceiver.

- **BT**

BT is a wireless protocol for short run, low cost devices that is expected to supplant cables in wireless personal area networks (WPANs). Bluetooth Applications are Cordless Desktop, Ultimate headset, Automatic synchronization and Multimedia Transfer. BT operates in the 2.45 GHz ISM band and uses frequency jumping to overcome obstructions and noise. BT has a crude information rate of 1 Mbps. Consequently, the information rate on a BT network is roughly 723 kbps [4].

Figure 2.12 BT

In addition, BT can be used to broadcast from 10 to 100 m and allows data transfer rates reaching 1 Mbps. BT is standardized in IEEE 802.15.1. BT is administered by the BT Special Interest Group and is documented in BT Core Specification Version 4.0. This group presented the BT low energy (BLE) protocol, which specifies the design of extraordinary low-cost BT smart devices that are powered with coin-cell batteries. Presently, BT devices operating from 2400 MHz to 2480 MHz may be used to broadcast up to 100 m. One master device is constrained to interface with 7 other devices in a ‘piconet’ [39].

CHAPTER 3

STATE-OF-THE-ART AND FRAMEWORK LATFORMS

Most WSNs require actuators to respond rapidly to external stimuli [39]. As mentioned previously, power is the most crucial issue in this field, and many studies have discussed and treated the problem of power consumption by employing different approaches via communication between nodes and a variety of optimization algorithms [40]. As in other networks, there are three widespread challenges that impact the availability and efficiency of a network:

- Utilizing network protocols to reduce the number of control and information packets.
- Choosing the best topology by placing the nodes in the right areas.
- Establishing a routing algorithm for transmitting information through a network from an initial node to the goal node [20].

Consequently, the main portions of a classic WSN include the following:

Wireless Sensor Node: Sensor nodes are the centre of the network as they oversee information gathering, and the transfer of data to a sink.

Sensor Field: A sensor field could be the region in which the nodes are situated.

BS: A BS provides control of the network, collects data from the network, and spreads data throughout the network. It also acts as an entryway to different networks, provides information handling and storage capacity, and provides a human-computer interface. The BS may be either a laptop or a workstation.

Sink: A sensor node that must accept, handle, and store information from other sensor nodes is known as a sink. Sink nodes serve to decrease the aggregate number of messages that should be sent, subsequently decreasing the amount of energy consumed by the network. Sinks are otherwise called data accumulation foci [21].

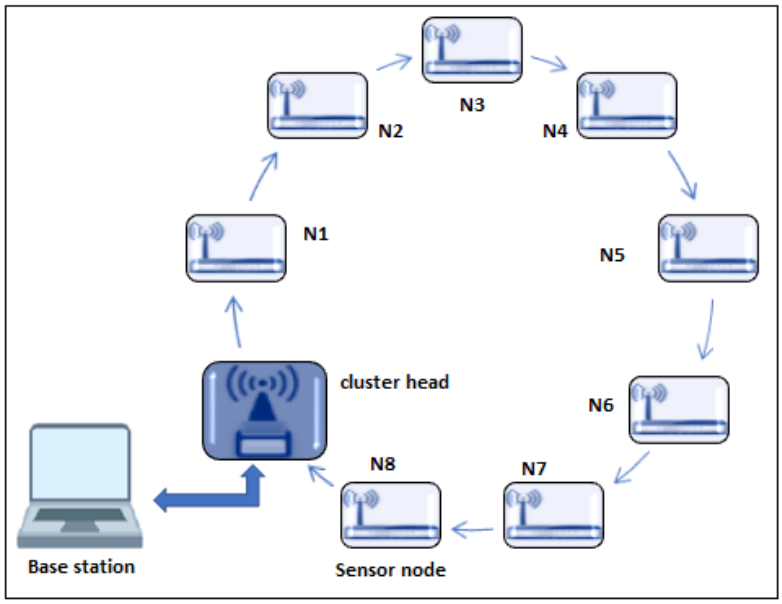

Figure 3.2 Hardware master/slave communication.

Hardware and software platform implementation are discussed in this chapter. All devices are shown in a clear and detailed way. We present three encryption algorithms that were implemented in the software platform, where the hardware platform contains a WSN node. Cluster nodes use handshake communication. Every node is named and uses the Caesar encryption algorithm for routing. The software platform was implemented by reinforcing the handshake, then, three encryption algorithms (Caesar cipher, CR4, and AES) were used to secure the cluster. The nodes were homogeneously distributed throughout the network in our simulations.

3.1 Hardware Platform

In this section, we discuss the hardware platform in two parts. The first part focuses on the design of a sensor node, while the second part focuses on handshaking for communication among WSN nodes.

3.1.1 Node Structure

A sensor is a device, module, or subsystem used to identify and measure changes in specific variables, and to send data to different hardware devices, usually a PC. A wireless sensor node be formed of:

- **Sensor**

The sensor is an electronic device, subsystem, or module with the purpose of detecting and responding to the physical events in its area. It then sends data to the aggregated electronic device. The specific data may be pressure, motion, light, heat, or any one of a number of other environmental phenomena.

Figure 3.2 Sensor hardware.

- **Processor**

Generally, the processor is an electronic integrated circuit that processes instructions from a computer. Four processor tasks are: fetching instruction, decoding, writeback, and executing. All processing activities were conducted with a PC.

- **Transceiver**

In this work, we use a BT transceiver operating in master, slave, or loopback modes during communication. When the BT module is in the automatic connection work mode, it will pursue the default mode set, waiting to transmit any data automatically. When the module is in the request reaction work mode, the user sends the AT command to the module to set the unit control parameters [41]. The HC-05 is a master/slave module and is by far the most popular and inexpensive module.

The role of this module (master/slave) can be configured with AT commands. The BT module at the BS starts in the master mode, while the BT module in the eight other nodes behave as slaves, as shown in Figure 3.3. A piconet network establishes communication between two nodes dynamically and automatically. BT radios are one example where a device may operate as master and slave simultaneously. The HC 05

entrenched BT serial connection module was used in this study. This module operates in two modes: an order-response work mode, and an automatic communication mode.

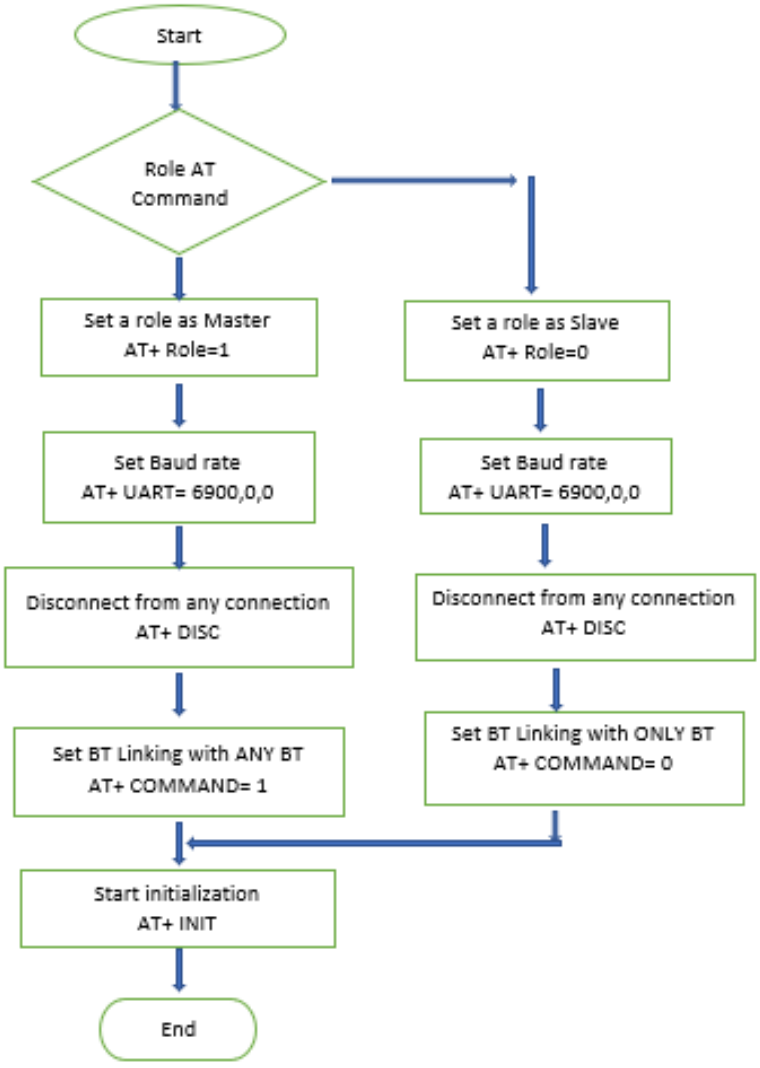

Figure 3.3 Master and slave modes in a BT module

Arduino Board

An Arduino board has a special program that is used to communicate with various devices and complete a wireless node. Arduino is an electronic open-source board with simple equipment and programming. Arduino hardware includes an Atmel AVR processor and on-board I/O connections [42]. Likewise, Arduino programming uses a standard programming language and boot loader that runs on the board. Arduino is considered a physical or inserted registering stage. It may be used to create intelligent

devices, taking inputs from various switches or sensors and controlling an assortment of lights, engines, and other devices. The Arduino programming language is used to run the communication, a comparative physical registering stage, which manages optical and acoustic data. Arduino has 14 advanced I/O pins (6 can be used for PWM (Pulse Width Modulation) output), 6 analogue inputs, 16 MHz ceramic resonators, a USB connector, a power jack, an ICSP (in-circuit serial programming) header, and a reset button, as shown in Figure 3.4. The user can connect the Arduino to a PC with a USB link and provide power with a DC power supply or a battery [43].

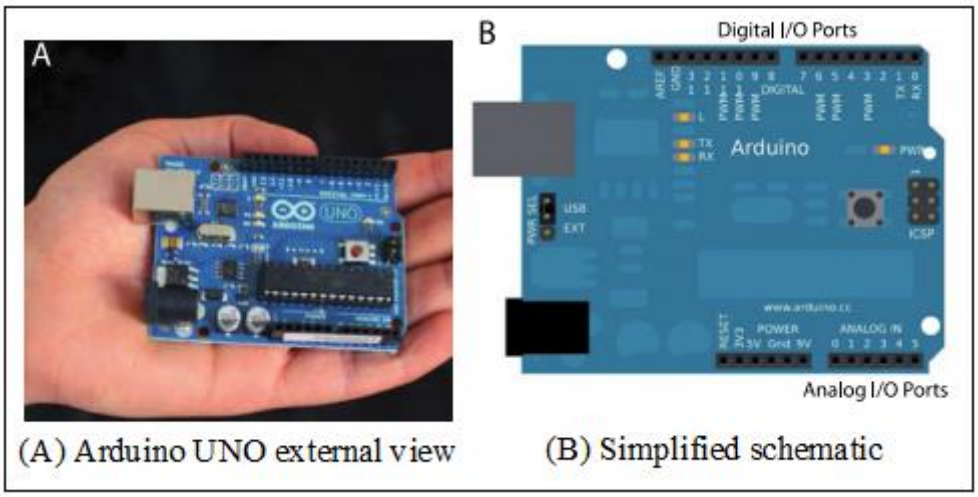

Figure 3.4 Arduino board.

The basic preferred use of Arduino is in large-scale open-source wireless networks [42].

3.1.1.1 WSN Node Creation

Several techniques should be used to design the WSN node. A Bluetooth should be used with the Arduino board, which is connected to a sensor, as shown in Figure 3.5. During this process, each node is given a private name, as shown in Table 3.1.

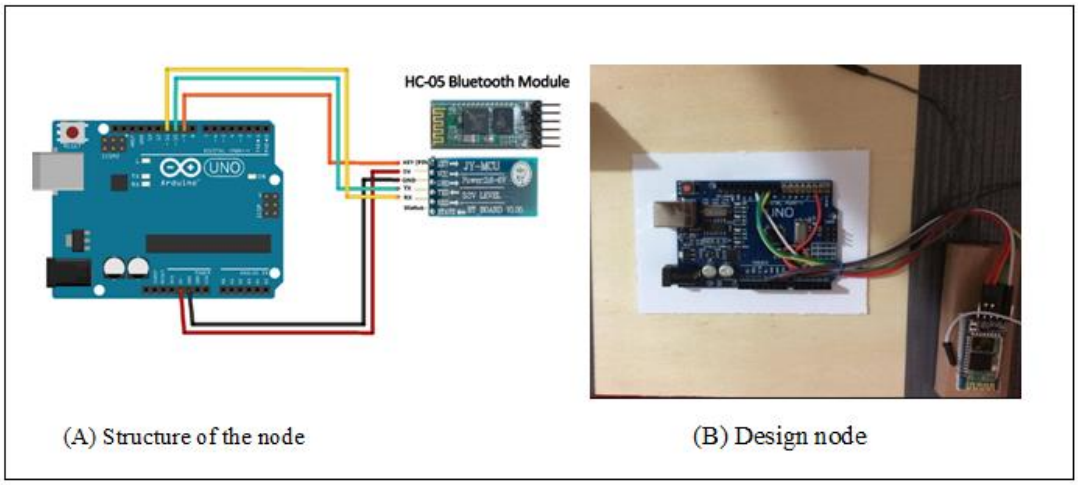

Figure 3.5 Designing the WSN node.

Table 3.1 Node Names

Node no.	Node name
1	ALPHA
2	BETA
3	GAMMA
4	DELTA
5	EPSILON
6	ZETA
7	ETA
8	THETA

3.1.2 WSN Node Clustering

Generally, a cluster is a gathering of nodes. Each node in a cluster is responsible for aggregating data from other nodes; such a node in the cluster is called a CH [44]. The sensor is the heart of a WSN node; the sensor node itself is designed with four basic components: a sensor, a processor, a power supply or battery, and a transceiver. Clustering is used to reduce energy consumption in sensor networks with hierarchical structures to enhance network performance [45]. In addition, a WSN node cluster is very significant because it allows control of the topology to minimize transmission

overhead, prevent redundancy, aggregate data from nodes, as well clustering is used for minimizing traffic and congestion in the clustering channel [45].

Hierarchical routing includes two stages. A cluster is formed in the first stage, followed by routing in the second stage [46]. Numerous hierarchical routing protocols have been used to reduce energy consumption and improve the scalability of WSNs. CHs can also form an additional cluster layer before reaching the sink. Some of the hierarchical protocols proposed for sensor networks include power-efficient gathering in sensor information systems (PEGASIS), low-energy adaptive clustering hierarchy (LEACH), adaptive threshold-sensitive energy-efficient sensor network (APTEEN), and threshold-sensitive energy-efficient sensor network (TEEN) [47].

The advantages of clustering protocols in a WSN include:

- **Scalability:** Protocols that are cluster-based limit the transmission of data between nodes to allow many nodes to be organised in the network.
- **Collision reduction:** The CH is responsible for executing most of the tasks for the nodes. A few nodes insist on accessing the channel, thereby refining the channel access protocol efficiency.
- **Energy efficiency:** Most of the time, the CH is active, and the remaining nodes awaken only during a specified interval when transmitting data to the CH.
- **Information locality:** The exchange of intra-cluster information between nodes and the CH. In other words, the CH is responsible for obtaining a briefing of the situation in the local network, and for sensing the status regarding information.
- **Routing backbone:** Cluster-based approaches also facilitate the efficient building of a routing backbone in the network, providing reliable paths from sensor nodes to the sink. Route-through traffic in the network is reduced because information is only sent to the sink from CHs [48].

Although several wireless communication devices are available, BT transceiver was used in this study. BT is a standardized protocol for sending and receiving data via a 2.4-GHz wireless link. This is a secure protocol and is suitable for short-range, low-power, low-cost wireless transmission between electronic devices.

3.1.2.1 Hierarchical Protocols

In a hierarchical architecture, the nodes are grouped in *clusters* and the local interactions between cluster members are controlled through a CH. Numerous hierarchical routing protocols have been used to reduce energy consumption and improve the scalability of WSNs. Some hierarchical protocols that were proposed for sensor networks include PEGASIS, LEACH, APTEEN, and the TEEN [49].

- **PEGASIS Algorithm**

PEGASIS is a rerouting technique where a chain-based approach is used in a greedy algorithm [20], such that every node sends to and receives from only one of its neighbours. An arbitrary CH is selected during each cycle. At that point, the sensor nodes will sense information and send it to their close neighbours, thus information is sent to the closest neighbours until it reaches its destination. The CH node summarizes the information and sends it to the sink [21].

The CH controls the communication request by passing a token among the nodes. A chain is built with the goal that every node receives data and sends it to an adjacent neighbour [50].

The PEGASIS structure is contingent upon a greedy chain. While a greedy chain cannot always guarantee minimal energy consumption, randomized leader selection does not consider a node's energy consumption or transmission distance [51]. It uses only one transmission to the BS per round.

3.1.2.2 Handshaking Communication

Handshaking is an automatic communication process between two devices that occurs before normal communication. Handshaking is often used to verify the quality or speed of a connection.

This work focuses fundamentally on using the handshake protocol to link eight nodes with a BS via a CH, as shown in Figure 3.6. The base station (BS) is an intermediate where transmission begins and ends from it, to compete forming a cluster of nodes via BT.

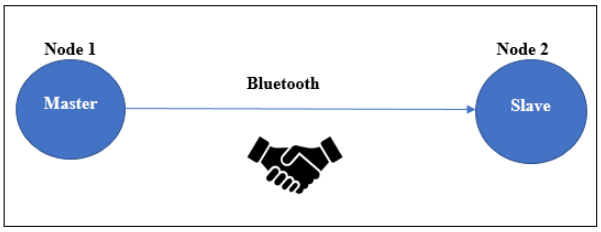

Figure 3.6 Handshake process.

Time delay during handshaking is defined as the time required to deliver a packet from an origin to a destination. Here, this time is the period during which data are collected from closer sources that might have to be withheld at intermediate nodes (i.e., at the BS) [40].

3.2.1 Security Requirements in WSNs

The most significant challenges for WSNs regarding security issues is the protection of data and resources from attacks and misbehaviour. Message authentication is desired in many WSN applications [41]. Consequently, WSNs are usually deployed in open and unattended environments where the nodes do not have any physical protection and are vulnerable to malicious attacks in a hostile environment, such as in a battlefield [36, 20]. Conventional security mechanisms are not directly applicable to WSNs because they do not consider unique constraints. To protect a sensor network, designing effective security mechanisms or protocols is crucial to maintaining control and preventing illegitimate nodes from participating in a network, which preserves much of a network's operation in most scenarios. Cryptographic mechanisms that use key distribution are the most common solutions for controlling network access. This is a very important focus in this study [52]. Attacks are generally considered to be either active or passive. Other private attacks include the following:

Data confidentiality: A sensor node should not permit its readings to be accessed by its neighbours unless they are authorized [53].

Data integrity: Messages must not be modified as they navigate from the sender to the receiver.

Availability: WSN services ought to be accessible at all times, even during an attack, e.g. during a denial-of-service attack. Various approaches have been proposed by researchers to accomplish this objective.

Data freshness: Information should be neoteric, guaranteeing that no foe can replay old messages. This is particularly important when WSN nodes use shared keys for messaging, because a potential enemy can dispatch a replay attack using the old key, then the new key is being revived and spread to all nodes in the WSN.A nonce or time-explicit counter might be added to check the age of each packet.

Self-organization: Every node in a WSN ought to be self-organizing and self-healing. This characteristic of a WSN represents another incredible security test. The dynamic nature of a WSN occasionally makes it impossible for the BS to send any pre-installed key to the nodes [54].

Secure localization: In many situations, it becomes essential to accurately and automatically locate every sensor node in a WSN. The architect of a WSN requires sensor nodes be precisely placed in a specific area.

Time synchronization: Applications involving sensor systems require time synchronization. Similarly, any security instrument for WSN ought to be time-synchronized [55].

Authentication: A recipient must have a system to check that the received packets originated from the real sender node. In the event of correspondence between two nodes, information can be authenticated with the MVC registered from the mutual private key.

3.2.1.1 Cryptography

A security system provides a solution against malicious attacks by making messages difficult to read and understand. Cryptography consists of encryption and decryption [37]. Encryption techniques fall into two categories [56]:

- Symmetric cipher: the same key is used for encryption and decryption.
- Asymmetric cipher: encryption and decryption use different keys.

Figure 3.7 shows symmetric and asymmetric ciphers.

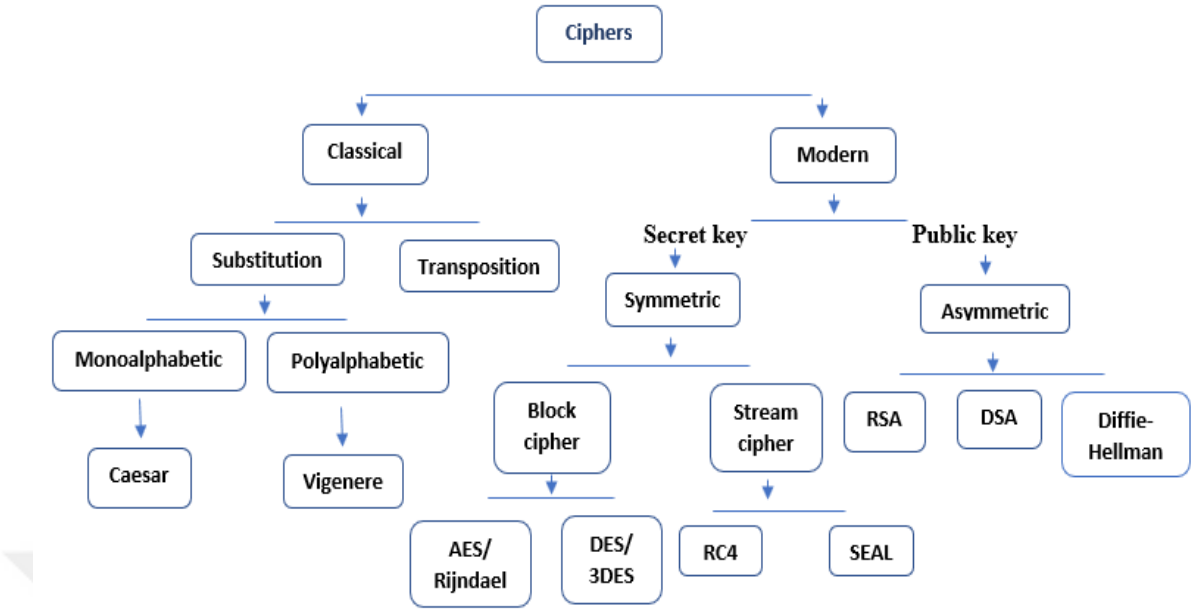

Figure 3.7 Classification Methods of encryption classification

From figure 3.7 of cryptography classification;

Classic Cryptography: It manipulates traditional characters, i.e., letters and digits directly. It is mainly based on 'security through obscurity'. The techniques employed for coding were kept secret and only the parties involved in communication knew about them and It requires the entire cryptosystem for communicating confidentially.

Modern Cryptography: It operates on binary bit sequences. It relies on publicly known mathematical algorithms for coding the information. Secrecy is obtained through a secrete key which is used as the seed for the algorithms. The computational difficulty of algorithms, absence of secret key, etc., make it impossible for an attacker to obtain the original information even if he knows the algorithm used for coding. Modern cryptography requires parties interested in secure communication to possess the secret key only.

- **Encryption Using the Caesar Cipher**

Encryption is an important technique for securing precise data but keeping encryption keys secure is an important issue. Encryption algorithms convert plain text (authentic

message before encryption) into cipher text (mixed message after encryption). Several encryption algorithms are used in information technology.

The Roman ruler Julius Caesar utilized an extremely straightforward cipher for secret information, known as a Caesar cipher in military circles [57, 58, 62]. The encryption in a Caesar cipher is achieved using the following function [58, 65]:

$E(x) = (x + n) \bmod 26$ (**3.1)**

The decryption is achieved using the following function:

$D(x) = (x – n) \bmod 26$ (**3.2)**

The Caesar cipher algorithm can be used to encrypt and decrypt a message in less time than used by other methods. Thus, it was used in this work to create a proposal for the protocol [62]. It is monoalphabetic and classified as a substitution cipher [66].

3.3.4 Data Aggregation (Data fusion)

In general, data aggregation refers to the collection of data from various sources, and the data can be processed in many ways [39]. Data aggregation algorithms are used to enhance network lifetime and improve energy efficiency [50]. Data fusion techniques have been used in sensor networks with the purpose of combining data from various sensors. However, these mechanisms can also be applied in other domains, such as text processing. The aim of data fusion in sensor networks is to reduce detection error and increase reliability by using data from multiple distributed sources [59]. Moreover, this process requires fewer transmissions and eliminates redundancy without hampering or affecting the data [14, 22, 33]. Sensor restrictions determine which routing protocol is convenient. Where these constraints correspond in the features of sensor network essentially the separate sensors, the nature of the sensor field, network behaviour, and sensing application requirements in terms of several desirable metrics [33]. Routing protocols for WSNs include directed aggregation, COUGAR, GEAR, ACQUIRE, RR, GAF LEACH, TEEN, PEGASIS, APTEEN, and SPIN [22].

Two main routing protocols differ in the way that data are sent from the sources to the sink:

Address-Centric Protocol (AC): With this protocol, every source independently transmits information to the BS along a shortened path (‘end-to-end routing’).

Data-Centric Protocol (DC): The sources send information to the sink; however, the nodes reroute the information and aggregate/consolidate information from multiple sources at the sink [60].

3.1.4.1 Directed Data Aggregation

Directed data aggregation during routing between the sink (BS) and sensors involves four tasks, depending on the sink enquiries. These four actions are:

1. Reinforcement
2. Data delivery
3. Interest propagation
4. Gradient setup (as shown in Figure 3.8) [36]

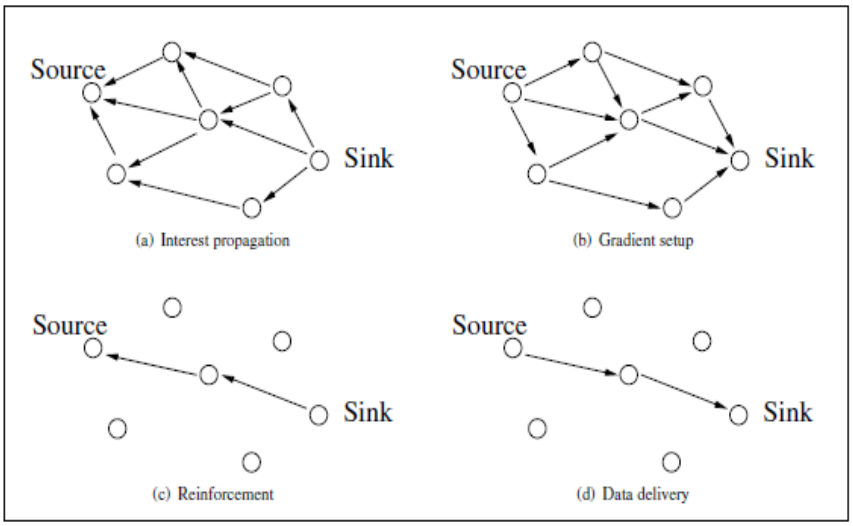

Figure 3.8 Directed diffusion protocol approaches [35]

The techniques used in the current project are data delivery and reinforcement. This is implemented with a packet conveyor represented as a frame (F). The frame automatically passes a packet between every node in the cluster. Then it sequentially fills the frame with aggregate data from these nodes. The properties of the frame include its length (64 bits with 8 cells, where the number of cells is analogous to the

number of WSN nodes). Each cell has 8 bits, and each cell is filled with the corresponding node data. This frame is shown in Figure 3.9.

Normally, the sink is responsible for collecting data from other nodes. Instead, a BS was used as an intermediate node for sequentially transmitting a single aggregate packet from every other node. Cluster head subsequently receives multiple input packets from every other node. The objective of a frame is to aggregate data from every node. A message is sent back to the BS once the frame is filled with data.

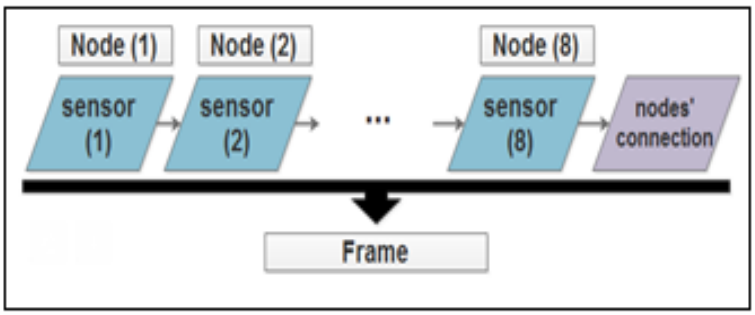

Figure 3.9 Frame user interface

3.1.5 Implementing Clustering via Handshaking Among Nodes

After the node is designed, the next step is to cluster these nodes with a suitable algorithm. A PC was used as the BS for implementing handshaking in this study. After we created eight nodes, we identified each one by name, as shown in Table 3.1. The second stage was to cluster these nodes via the handshake communication that was implemented between them using the master/slave BT AT command mode. The process starts by pressing the 'push button' key on the interface board in the BS (PC). The transmission moves from the BS toward the CH. Then, each node prepares itself to contact its nearest neighbour, where the sender node is positioned as the master mode. F is released from the CH and moves toward the first node (alpha); after the signal is received, the name of first node changed automatically and immediately be encrypted to a new name, and the process will then continue with the second node, and so on, where the Caesar cipher works automatically while the other nodes are in slave

mode. The process continues until encryption is complete and eventually returns to the CH, as shown in Figures 3.10, 3.11, and 3.12.

The CH behaves as a master when releasing the frame. Each node contacts the next node to prepare a connection. The AT command is used to configure a node as a slave:

- Set the role as slave by writing "AT+ROLE=0"
- Set the baud rate to be compatible with other baud rates by writing "AT+UART=9600,0,0"
- Free the BT from any connection by writing "AT+DISC"
- Set the connection mode to only one connection by writing "AT=CMODE=0"

Figures 3.10 to 3.13 show the hardware platform, where this platform includes:

- A simulation interface to start the transmission process
- WSN cluster node activation
- Encryption with the Caesar cipher

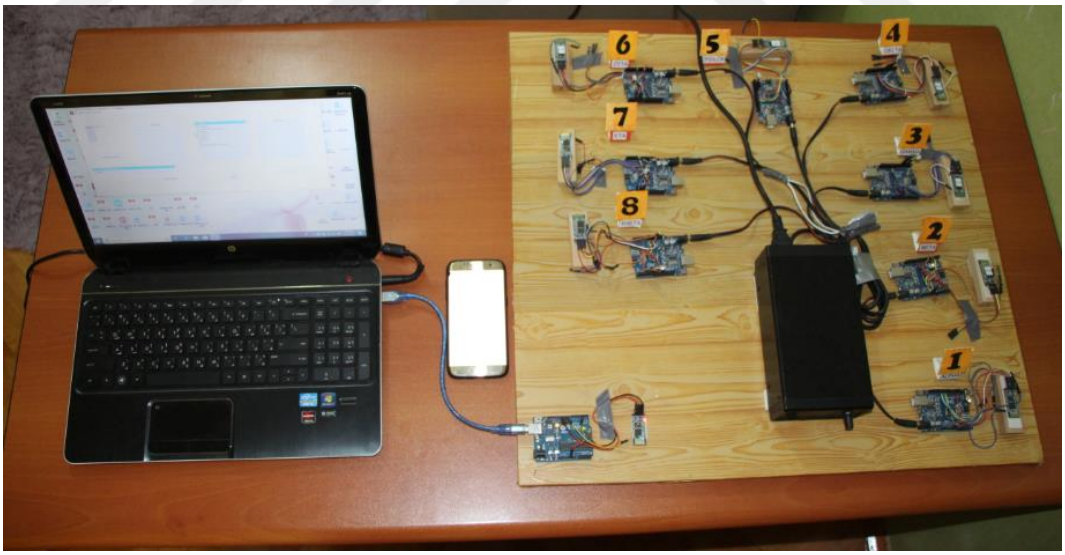

Figure 3.10 Hardware platform with eight nodes.

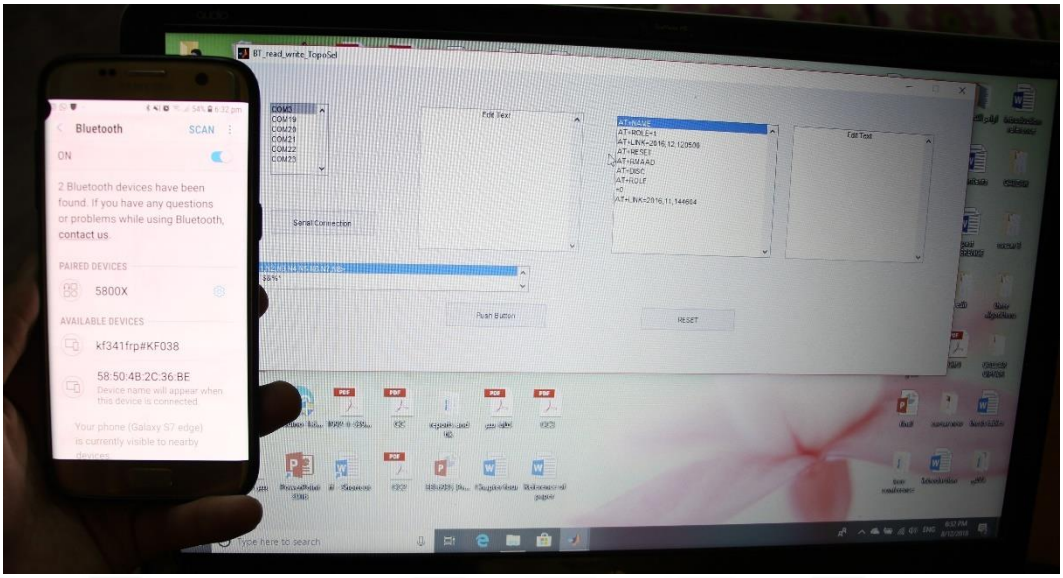

Figure 3.11 Interface started from first node.

Figure 3.12 Cluster activation.

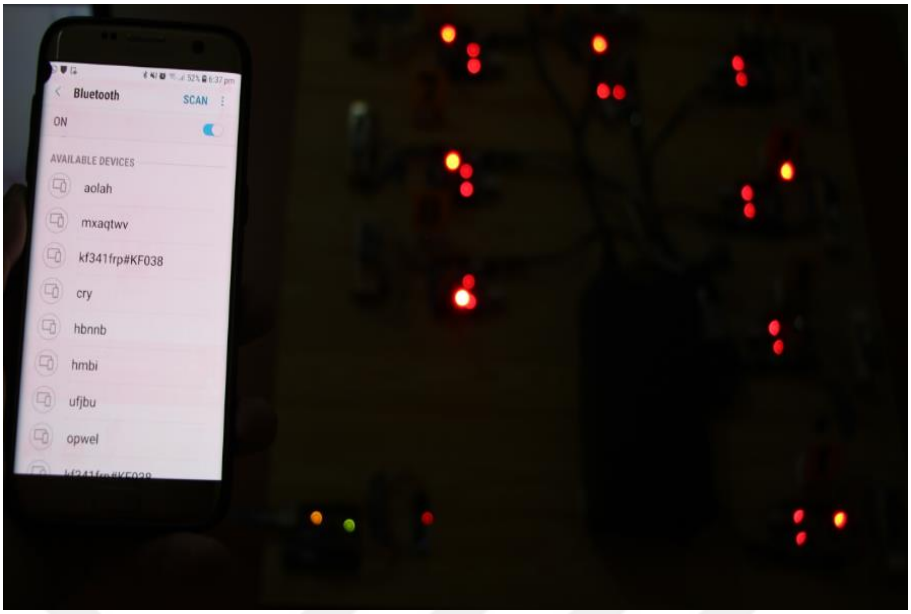

Figure 3.13 Encrypted cluster nodes.

3.2 Software Platform

This platform implements several algorithms. It is used to enhance the hardware platform and implement the necessary algorithms. Moreover, the software platform is used to determine the required parameter measurements, especially energy consumption and task execution time. Diverse architectures were developed using MATLAB.

3.2.1 Clustering nodes:

The simulation was executed in three phases. First, the cluster where the proposed simulation begins names each node, then each node joins its nearest neighbour via handshaking in a master/slave pattern. This process continues, from the BS to the CH, and then the first node, followed by all the remaining nodes. Consequently, each node is specified by its private name.

3.2.2 Aggregation data through the frame:

A frame was designed to aggregate data from cluster nodes. This frame string was sent to each node with the following command:

Fprint f (serConn, '%s', sprint ('N1, N2, N3, N4, N5, N6, N7, N8>|! -_`$&%*'));

The frame string fills with data from all eight nodes, from the first node until the eighth node, after which the frame is saved in the BS. Moreover, this is performed once communication is established. No redundant data should be transferred. Furthermore, the traffic flow should be distributed evenly among the sensor nodes to avoid congestion and network partitioning.

3.2.3 BT master/slave communication

At this stage, the BS is configured as a master via handshaking. The BS allows the aggregation of the sensor network data onto a PC or other computer platform. In the current work, the PC represents a BS, which is responsible for preparing connections between nodes and sending frames to each node. The linking process starts between a master BS and the first node by sending a frame from the BS to the node. The first node receives the frame, which collects data and subsequently prepares the connection with the second node. The frame is updated with sensor data and the BS is disconnected and reset.

The mode of the first node is subsequently changed from slave to master to prepare for communication with the second node, which is in slave mode. A frame is then sent from the first node to the second node. The first node is disconnected and reset. The operation continues with the second node, where the mode of the second node changes from slave to master. Links are then prepared with the third node, which is in slave mode, followed by a frame sent to the third node after the data is collected from the second node. The process continues until the frame is filled with all the data aggregated from all eight nodes, which is then sent back to be saved in the BS. This simulation can be summed up in several steps:

- Put the AT command mode
- Set role as slave
- Set baud rate to 9600
- Disconnect from any connection
- Enable pairing mode
- Wait to receive a frame
- Fill the frame with data

- Disconnect the current pairing
- Connect to the neighbouring node

After executing all the above points, the frame is sent, and the nodes are named and clustered.

3.2.4 Implementation Security Algorithms

The encrypted structure implements three encryption algorithms as a reasonable technique to secure the cluster nodes: Caesar cipher, RC4, and advanced encryption standard (AES).

CHAPTER 4

SIMULATION RESULTS AND MEASUREMENTS

Sensor nodes in networks act as a group to communicate with very small transceiver devices to configure WSN nodes. These nodes cooperate autonomously to form a logical cluster of WSN nodes. Data packets in this logical network are routed to a BS and then cluster towards the first node; this continues with the remaining nodes and eventually back to the CH. The data is subsequently saved in the BS. These packets are routed by the system hop by hop [61].

In the previous chapters, we presented hardware and software platforms. Consequently, in this chapter, we will present simulation results as well as measurements of the major parameters: energy consumption, execution time of tasks during the run time of the algorithms.

4.1 Implementing a Cluster (One Chain)

The software platform was used to enhance the hardware platform and access the necessary facilities to implement the algorithms. It was also used to determine the required parameter measurements, especially energy consumption and task execution time, within a variety of architectures created with WSN nodes with MATLAB. The simulation results allow us to evaluate the important parameters in the network nodes with the given algorithms as a basis for evaluating performance enhancements.

The following pseudocode describes handshake communication (one chain) in a cluster with master and slave modes:

Input: msg[1,..,8] --> Empty frame for a sensor value and connection path for each node

Output: ETX and ERX

start:

exe_time=0.043;

config node(i) as slave mode

role=0;

```
uart=9600;
cmode=0; // connect to one BT
disconnect;
slaveTime=exe_time * 4; // 4 is the number of commands
config node(i) as master mode
role=1;
uart=9600;
cmode=0; // connect to one BT
disconnect;
prepare link to next node;
masterTime=exe_time * 6; // 6 is the number of commands
while (i<9)
if i=8, node(i) connect to node (1);
node (i) connect to node(i+1);
update msg[i] with sensor value;
calc ETX(msg), ERX(msg);
send msg[1,..,8] to node (i+1);
Return ETX, ERX;
End
```

The simulation pseudocode shown above shows how each node prepares to connect to its neighbour. Communication begins with the BS in master mode while all other communication nodes are in slave mode. The first node is in slave mode when AT = 0, and the node automatically changes its mode from slave to master mode when AT = 1 after receiving the frame. Figure 4.1 shows the simulated BT during handshake communication.

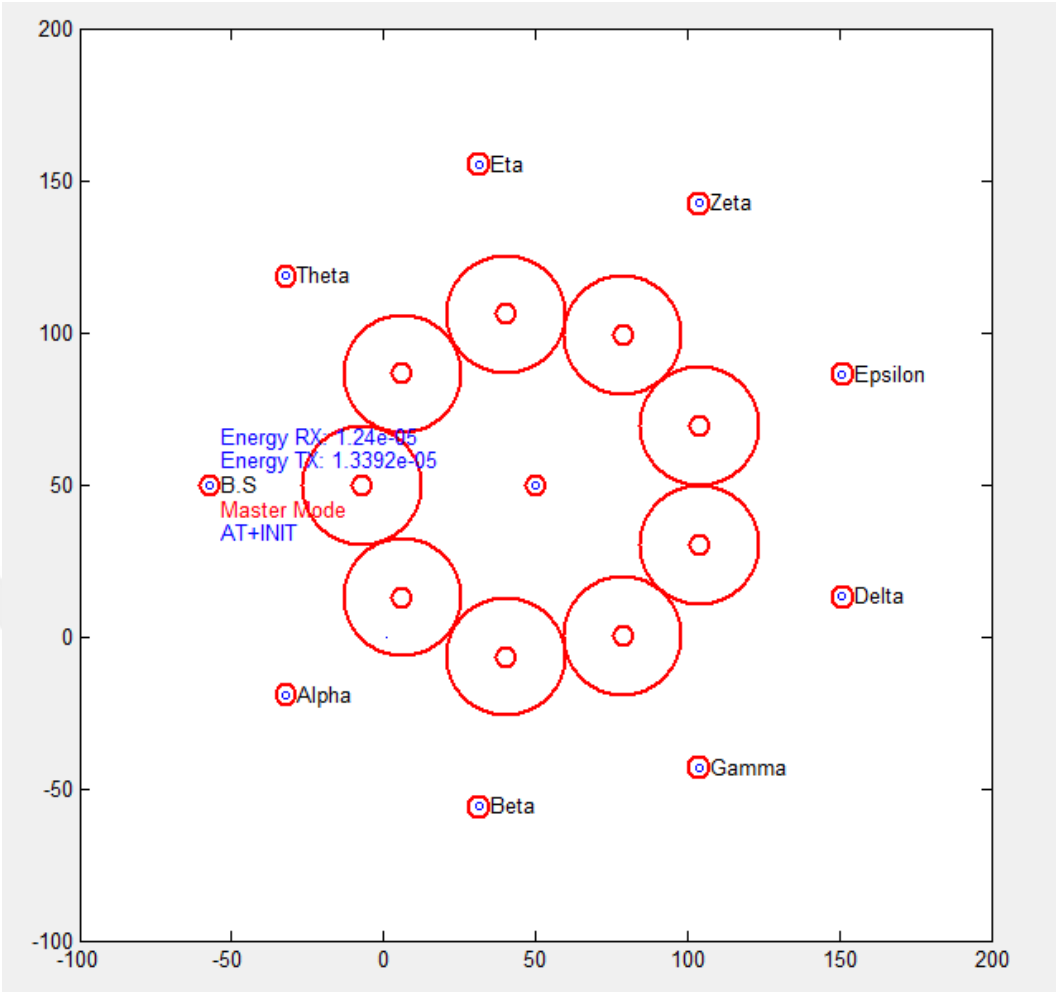

Figure 4.1 BS in master mode starting communication after naming nodes.

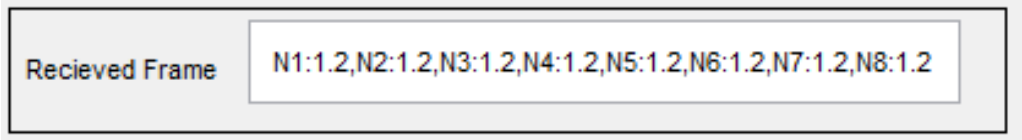

Figure 4.2 Frame filled with data.

4.2 Implementation of PEGASIS Chains

In this step, we used the same number of nodes but connected them together in two chains. A simulation of a PEGASIS algorithm was built with eight nodes. The BS was used to implement two PEGASIS chains, as shown in Figure 4.3. The BS was linked with the farthest node (giving a BT path length equal to 20 m) to connect with the nearest neighbour. This operation continued until the chains were completed.

We needed two frames (F1 and F2) to aggregate data from these chains. It is important to note the frames are simultaneously released from the BS. Frame F1 aggregated data from Node 1 to Node 4, which was then returned and saved in the BS. Similarly, F2 aggregated data from Node 5 to Node 8. F2 also returns data to the BS, where it is saved. The frames are filled with data from every node, which is then saved in the BS.

PEGASIS Pseudocode

Input: msg[1,..,8] --> Empty frame for sensor value and connection path for each node

nu=1,2,3,4 nd=5,6,7,8

Output: ETX, ERX, and Distance

start:

exe_time=0.043;

config nu(i) and nd(i) as slave mode

role=0;

uart=9600;

cmode=0; // connect to one BT

disconnect;

slaveTime=exe_time * 4; // 4 number of commands

config node(i) as master mode

role=1;

uart=9600;

cmode=0; // connect to one BT

disconnect;

prepare link to next node;

masterTime=exe_time * 6; // 6 is the number of commands

while (i<5)

nu(i) and nd(i) connect to nu(i+1) and nd(i+1);

update msg[i] with sensor value;

calc ETX(msg), ERX(msg);

send msg[1,..,8] to nu(i+1) and to nd(i+1);

Distance = 20 meter;

Return ETX, ERX, Distance;

End

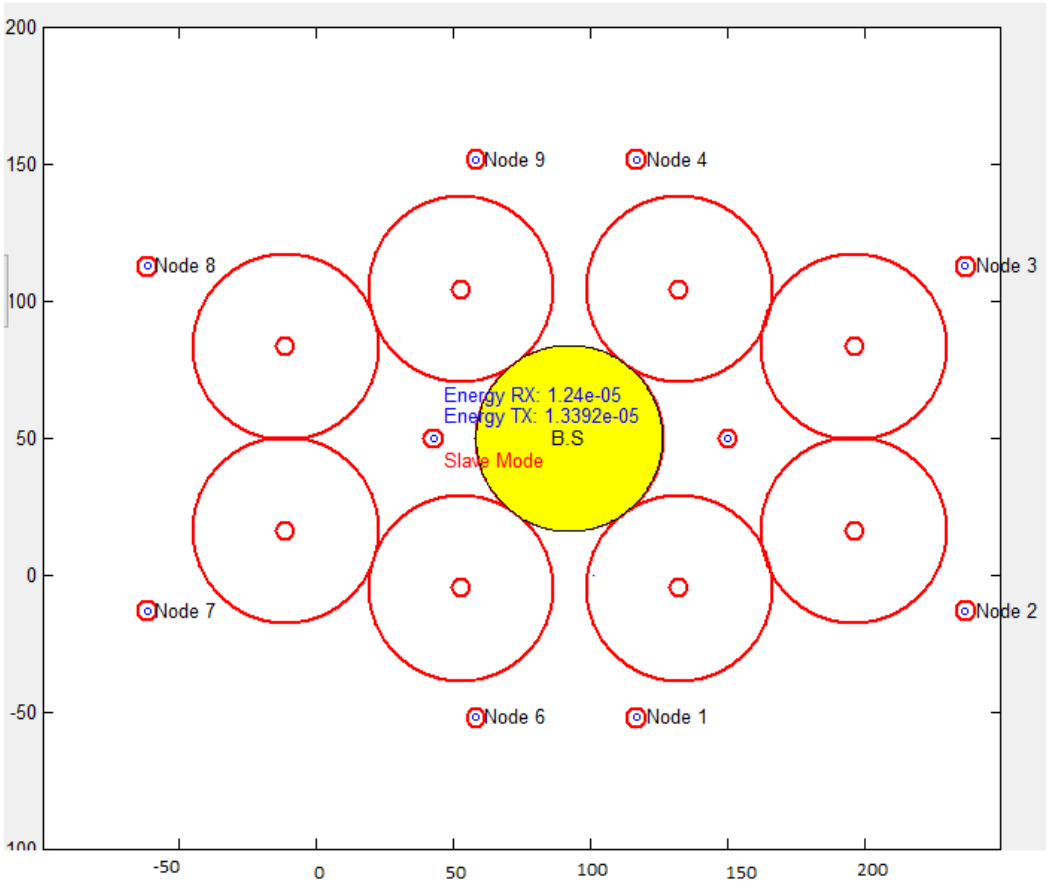

Figure 4.3 PEGASIS simulation.

4.3 Required Performance Measurements and Optimization Objectives

In order to complete the final stage of the work, the results of the study need to be discussed and compared based on the performance of the algorithms as designed and implemented. This comparison aims to determine the possibilities for each algorithm and the objectives they represent. Because the WSNs are constrained in terms of resources such as power, the proposed technique was also optimized even though large variations in the performance metrics made a comprehensive evaluation difficult.

4.3.1 Energy Consumption

One of the most important parameters is energy consumption. The total energy consumed by the two algorithms in the network must be measured to determine which uses less energy [62].

Regarding the energy consumed in wireless communication [63], many researchers have reached the same conclusion that calculate the formula of a network node transmitting bits to, or receiving bits from, another node and finally reached to the standard relations for energies. Where the transmitter energy E_{Tr} and receiver energy E_{Rx} are defined as follows [28]:

$$E_{Tr}(K; d) = E_{Tx\text{-}elec}(K) + E_{Tx\text{-}amp}(K; d) \quad \textbf{(4.1)}$$

$$E_{Tx}(K; d) = E_{elec} * K + \epsilon_{amp} * K * d_2 \quad \textbf{(4.2)}$$

$$E_{Rx}(K) = E_{Rx\text{-}elec}(K) \quad \textbf{(4.3)}$$

$$E_{Rx}(K) = E_{elec} * K \quad \textbf{(4.4)}$$

In the classical model with transmitter and receiver energy values, the ideal energy E_{idl} = 40 nJ/bit and the sleep energy are zero. E_{elec} is the energy consumed by the transmitting circuit, which is equal to 50 nJ/bit. E_{amp} is the energy consumed by the transmitting amplifier, which is equal to 100 pJ/bit/m^2. K is the size of the transmitted or received data packet; various K values for each node are shown in Table 4.3, and *d* is the distance between two nodes (*d* = 20 m) [64].

The WSN real-time environment deals with real-time environments and in many cases sensor data must be delivered in specific time constraints so that appropriate observations or actions should be taken. The calculated time in the table is the time

needed to encrypt each node and in the same moment is to complete the connection to two nodes.

Table 4.1 Relationship between nodes and bits

Node no.	K (bits)
1	8
2	16
3	24
4	32
5	40
6	48
7	56
8	64

Table 4.2 Transmitter and receiver energy consumption with handshake communication

Node no.	Time (s)	E_{Tx} (pJ)	E_{Rx} (pJ)
1	0.83	0.72	0.4
2	1.66	1.44	0.8
3	2.46	2.14	1.2
4	3.32	2.88	1.6
5	4.15	3.6	2.0
6	4.98	4.32	2.4
7	5.81	5.04	2.8
8	6.64	5.76	3.2

The above values were used to calculate E_{Rx} and E_{Tx}. Tables 4.1 and 4.2 show the energy consumed by the receiver and transmitter in each node via handshaking for one chain, and the PEGASIS algorithm with two chains.

Table 4.3 Transmitter and receiver energy consumption with PEGASIS

Node no.	Time (s)	E TX (pJ)	E RX (pJ)
1	0.83	0.72	0.4
2	1.66	1.44	0.8
3	2.46	2.14	1.2
4	3.32	2.88	1.6
5	0.83	3.6	2.0
6	1.66	4.32	2.4
7	2.46	5.04	2.8
8	3.32	5.76	3.2

A WSN cluster based on Arduino and IEEE 802.15.1/BT technology was presented in the previous section. The design was based on several ideas. We needed to measure the energy required to operate the sensor transducer. We used a power supply to provide the cluster with the power required to operate nodes at 9 V while preparing each node. Then we used an Avometer to measure the current passing through each node so that we could calculate the power in each node. The transmitter current in the BT module ranged from 20 to 30 mA, while the current was 0.2 mA during sleep, as shown in Table 4.4. In addition, we calculated the time required for the BT module to receive a signal and encrypt the name used in each node.

Table 4.4 Energy consumption of transmission via BT

Node no.	Energy Consumption (nJ)
1	4.767
2	5.008
3	6.284
4	7.365
5	8.758
6	9.373
7	10.592
8	11.837

4.4 Cluster Encryption

After clustering and naming each node, the names of the various nodes were encrypted according to three scenarios.

Table 4.5 Characters across seven bits in each name

Node no.	Node Names	Characters of Name
1	**ALPHA**	**A L P H A 0 0**
2	**BETA**	**B E T A 0 0 0**
3	**GAMMA**	**G A M M A 0 0**
4	**DELTA**	**D E L T A 0 0**
5	**EPSELON**	**E P S E L O N**
6	**ZETA**	**Z E T A 0 0 0**
7	**ETA**	**E T A 0 0 0 0**
8	**THETA**	**T H E T A 0 0**

I. Caesar Cipher

A shift algorithm (a Caesar cipher) was implemented in this scenario. Table 4.5 shows the distribution of characters of each name. We assumed that the empty fields in the names with less than seven characters were already filled with zeros so that every node name contained an equal number of bits. Five random numbers were subsequently added to each name (characters, zeros, and numbers). Thus, the encryption process used both characters and numbers in the Caesar cipher. The simulation started by assigning a unique name to each node. Figure 4.4 shows the clusters with encrypted node names during the first cycle. Table 4.6 shows the encrypted name of each node.

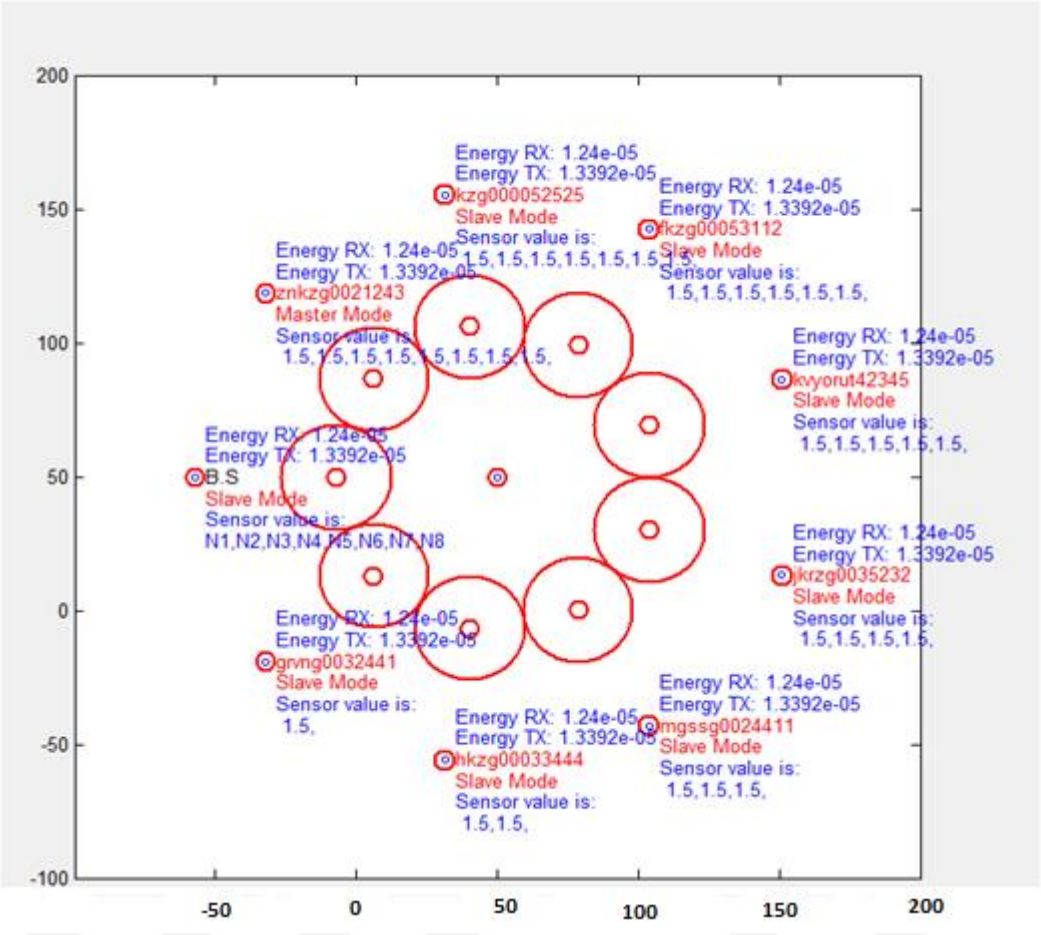

Figure 4.4 Simulation of the Caesar cipher to encrypt the cluster nodes.

Table 4.6 Names for each cluster node encrypted with the Caesar cipher

Node No.	Names of the Encrypted Nodes
1	doskd0032441
2	ehwd00033444
3	jdppd0024411
4	ghowd0035232
5	hsvlorq42345
6	chwd00053112
7	hwd000052525
8	wkhwd0021243

II. RC4 Cipher

The RC4 algorithm was used for encryption. The RC4 is a symmetric algorithm that was designed by Ron Rivest in 1987 [67] and is one of the most widely used stream ciphers in popular protocols. It is used for SSL (Secure Sockets Layer) to protect Internet traffic, secure WEP (Wired Equivalent Privacy) wireless network protocol, and TKIP (Temporal Key Integrity Protocol) also called WPA (Wi-Fi Protected Access) [68]. RC4 is based on a random permutation [69] and is categorized as being fast and simple in terms of software. RC4 was used to encrypt node names, and the encrypted node names are shown in Table 3. The simulation results after RC4 encryption are shown in Figure 4.5. We note that this algorithm uses 2^7 codes throughout Table 4.7, including letters and symbols. Five random numbers are added to each node name, and the created name is subsequently encrypted.

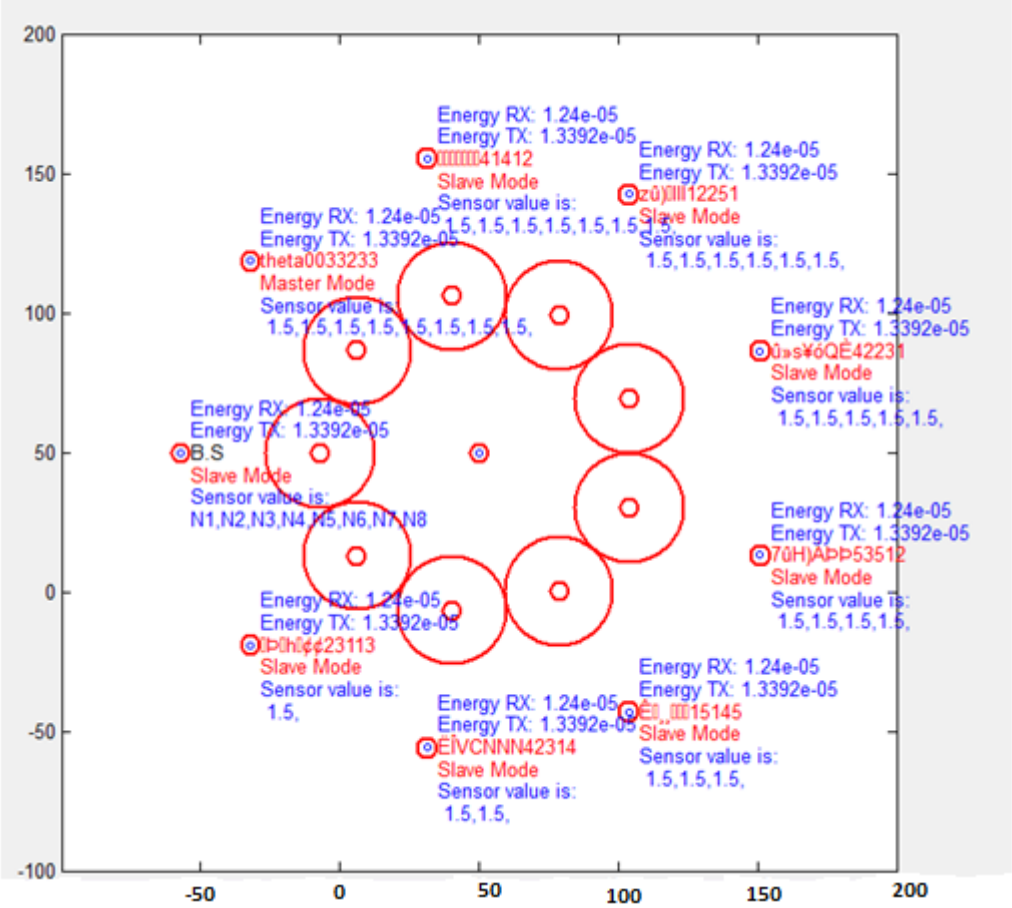

Figure 4.5 Simulation of RC4 to encrypt the cluster nodes with AT command.

Table 4.7 Names for each cluster node encrypted with the RC4 algorithm

Node No.	Names of the Encrypted Nodes
1	ÁHI,ÁÞÞ15412
2	|5°′51553
3	Ê”,,”“‘44424
4	deÞtČ¢¢25132
5	Îs¹52324
6	zeta00045531
7	etČ¢¢¢¢23113
8	VJÎVCNN42314

III. AES Cipher

The third security algorithm tested here is the advanced encryption standard (AES) algorithm. AES is symmetric and was created by Joan Daemin and Vincent Rijmain [70]. The functions of the AES are organized in a 4 × 4 array of bytes (128 bits). The cluster nodes encrypted with the AES algorithm were encrypted. These nodes are shown in Figure 4.6. Table 4.8 shows the name of each encrypted node via AES.

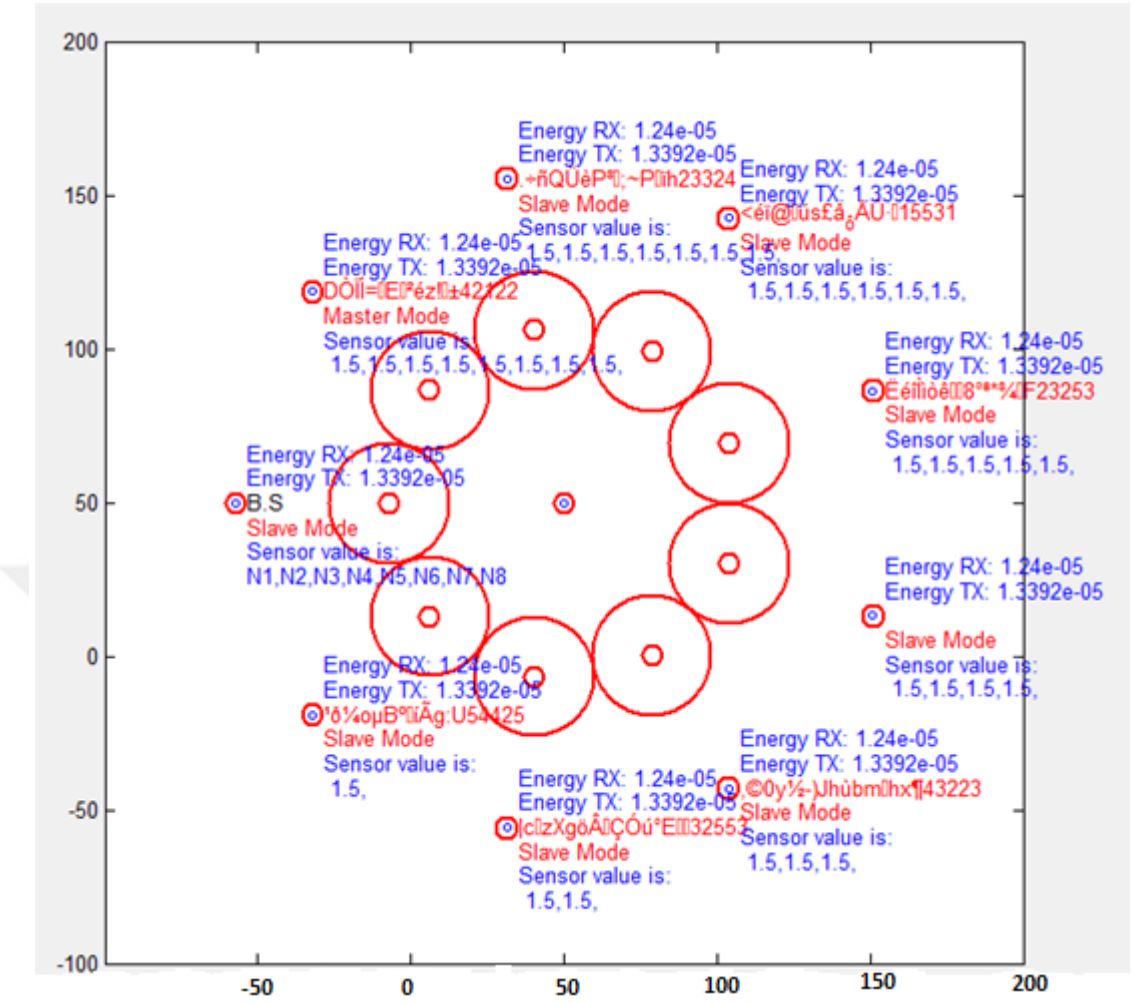

Figure 4.6 Simulation of the AES to encrypt the cluster nodes.

Table 4.8 Names for each cluster node encrypted with the AES algorithm

Node No.	Names of the Encrypted Nodes
1	¹ð¼oµ-Bºí Ãg:U54425
2	|czXgöÂÇÓú°E—32553
3	,©0y½-)Jhùbm,hx¶43223
4	ªôÈÝC_Â¦r:¡õJ*25121
5	ËéíÌìòê—8°ª*¾^F23253
6	<éï@ús£â_ôÅÛ·˜15531
7	.÷ñQÜèP³';~Pïh23324
8	D-ÒÏÍE²éz!Œ±42122

Each state during AES defines an encryption or decryption key for converting plain text data to encrypted data. The algorithm consists of four iterations, depending on the length of the plain text. The state of each node during the implementation of the AES algorithm is shown in Table 4.9.

Table 4.9 The state of nodes during AES.

Cluster Head Node	Node 1	Node 2
97 97 0 0 **108 48 0 0** **112 48 0 0** **104 0 0 0**	98 48 0 0 101 48 0 0 116 48 0 0 97 0 0 0	98 48 0 0 101 48 0 0 116 48 0 0 97 0 0 0
Node 3	**Node 4**	**Node 5**
103 97 0 0 **97 48 0 0** **109 48 0 0** **109 0 0 0**	100 97 0 0 101 48 0 0 108 48 0 0 116 0 0 0	101 108 0 0 112 111 0 0 115 110 0 0 105 0 0 0
Node 6	**Node 7**	**Node 8**
122 48 0 0 **101 48 0 0** **116 48 0 0** **97 0 0 0**	101 48 0 0 116 48 0 0 97 48 0 0 48 0 0 0	116 97 0 0 104 48 0 0 101 48 0 0 116 0 0 0

4.4.1 Security Analysis

In this section we will make comparison between Caesar cipher, CR4 and AES based on avalanche effect with same key and plaintext.

For the first node by using Caesar

KEY ASCII (litters)

PLAINTEXT

ALPHA

CIPHER

D o s k d 0 0 3 2 4 4 1

For the same node by using CR4

KEY ASCII

PLAINTEXT

ALPHA

CIPHER

Á H I , Á Þ Þ 1 5 4 1 2

For the same node by using AES

KEY ASCII

PLAINTEXT

ALPHA

CIPHER

¹ ð ¼ o µ - B º í Ã g : U 5 4 4 2 5

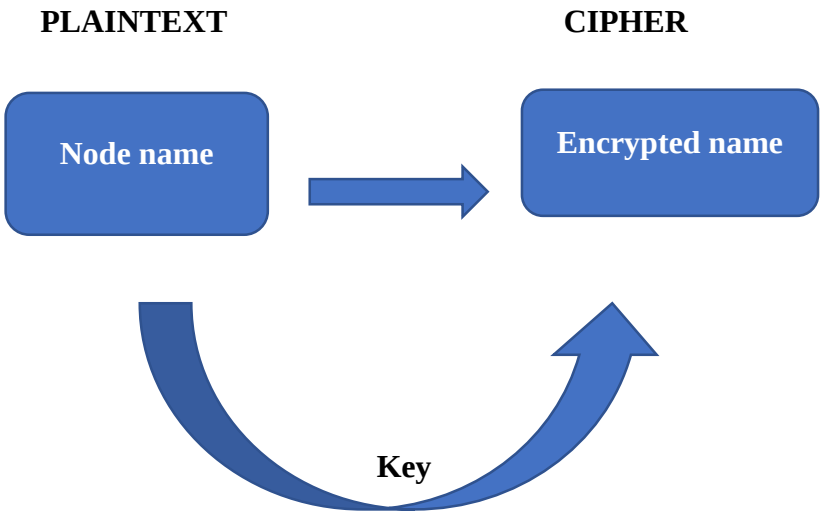

Figure 4.7: Security of Node

The comparison among the three ciphers in bits to calculate the difference and found out that there was a change; in Caesar, 5-bit where used the alphabet letters as well zeros' and 5 random numbers. While in CR4 8- bit which used all letters as well zeros' and 5 random numbers. Finally, in AES 128-bit used all letters and symbols in ASCII as well zeros' and 5 random numbers.

4.5 Discussion and Comparison of Results

Based on the previous results of the design and implementation of two significant platforms in a WSN, we claim several major achievements:

- Caesar cipher (on the hardware platform) was fast when encrypting the cluster nodes. This is because Caesar cipher has the least number of steps in execution, so it is considered as a quick algorithm
- WSN applications always use many nodes, whereas this project used only eight nodes because of the prototype hardware design, noting that the work could be represented by numerous nodes in the multi-cluster.

- In addition, from classic chart with the methods for classified classic encryption, two modern algorithms were selected, RC4 and AES. A classical algorithm (the Caesar cipher) was employed with the computational function to generate numbers, and adding zeros made this cipher strong and difficult to penetrate. The cipher can be made more secure by perform-in multiple rounds of such permutations.
- After completing the implementation of the three algorithms (Caesar cipher, RC4, and AES), generally, to break the cipher, we must estimate the length of the encryption key. This length can be estimated by using the logic that plaintext words separated by multiples of the length of the key will get encoded in the same way. By using the brute force to examine the time required for attackers to break the code online for examining our security algorithms. The results showed that extra time would be needed to break the code in all our algorithms. By examining fracture results of the codes, we concluded that all of the algorithms were very strong and would be suitable for creating a safe and convenient environment to hold WSN nodes. Even the old and traditional algorithms would be suitable if they were fortified with arithmetical and computational methods. Moreover, the speed of implementation was very high.
- PEGASIS algorithm where a multiple chains can be used if time of task is limited, then more frames are required for data aggregation.
- The total energy consumed during a handshake (one chain) is the sum of all energies consumed by all nodes. Data from Tables 4.1, and 4.2 are shown in Figure 4.8, and the results show that PEGASIS can be used to decrease the total energy consumption and computational time.

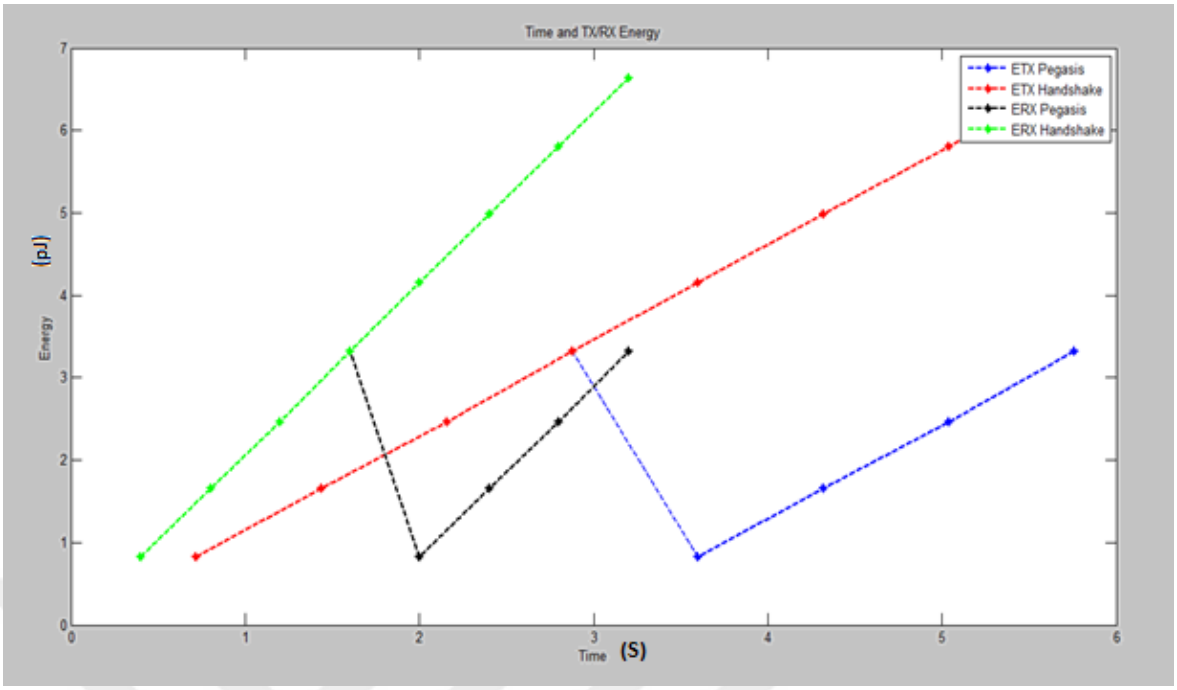

Figure 4.8 Relationship between transmitter and receiver energies over one and two chains.

4.6 Case Study

In essence, wireless sensor networks require technologies from three different research areas: sensing, communication and computing.

A. Clustering process in Wireless Sensor Networks

Clusters are an important application of sensor networks for both monitoring in smart building and health systems. There are two main steps in clustering, which are CH selection and cluster formation. The clustering of sensor nodes poses different challenges due to communication, processing and energy constraints. Sensors should collaborate and share data to exploit the benefits of sensor data aggregation; however, this should be done without sending data requests to, or collecting data from, every sensor, thereby avoiding an overload of the network and using up the energy supply. The base station is fixed and located far from the sensors. All nodes are homogeneous, and energy constrained, and all nodes can reach the base station without any collisions while broadcasting.

B. Distributed Classification in Wireless Sensor Networks Using Base Station Agents
In a traditional sensor network, data are collected by individual sensors and sent to the cluster heads nodes (CHs) which are responsible for all processing, followed by saving the data aggregation at the base station.

C. Data Aggregation
As discussed previously, data aggregation eradicates duplication of data. That is collecting data from multiple sensor nodes can potentially provide the best performance. The sensor nodes only communicate with sensors nodes within a neighbourhood.

D. Cluster Security
The cluster security services are installed on a node. A private key for that node is created and a public key is then derived from the private key. The private key remains on the node in a protected file that only the root user can access.

E. My interventions targeted three areas
The cluster in hardware platform is feasible from this technical, operational, and financial standpoint depending on our designed Hardware platform. Use the appropriate algorithms for applications that require less time in execution such as PIGASIS. Achieve reliability by implementing different types of encryption algorithms.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

In recent times, preliminary European projects for WSNs have been financed, especially since 2001. In the sixth and seventh generations of programmers, several projects dedicated to communication protocols, and architectural and technical resolutions for integrated systems were financed by the European Commission (EC). The first projects to be launched were WISENTS, e-SENSE, CRUISE, and CONET [71].

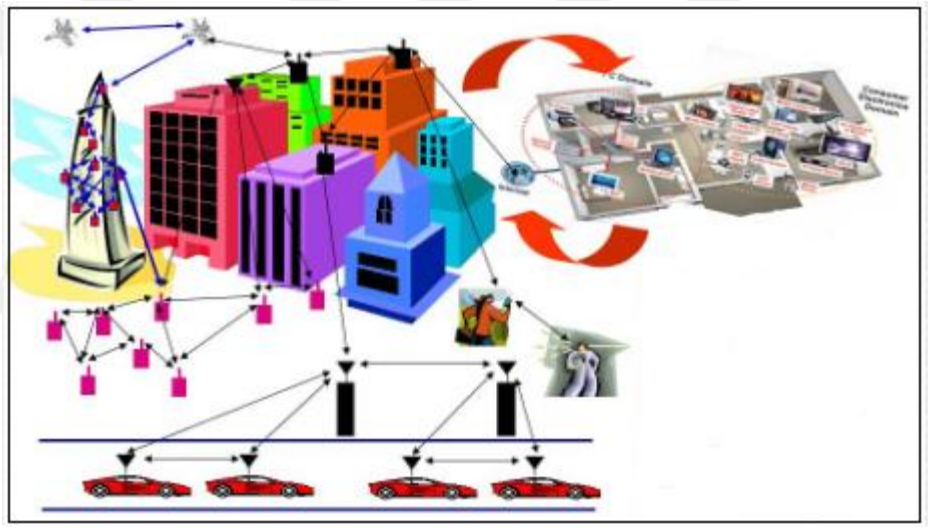

Figure 5.1 WSNs in the future.

Through demanding users of WSN appliances and systems with better capabilities and higher levels in functionality as shown in figure 5.1. Sensors in these devices and systems are used to provide measurements or to identify control states. These sensors are candidates for smart technologies, where smart sensors can communicate measurements directly to an instrument or a system. Transducer (i.e., sensor or actuator) networks in a system can provide flexibility, improve system performance, and facilitate system installation, upgrades, and maintenance [43].

Future applications of WSNs may include industrial process automation, aircraft control systems, and traffic management systems. WSN applications will develop

further and become better known with the progression of sensor innovation, and a greater amount of sensor data will be generated. Protocols and procedures will then be extremely critical, leading to standardized examination and analysis. The aim would be to introduce this thesis to support interests by using research to develop applications.

5.1 Aim of the Thesis

WSN environments have multiple factors like other networks which depend on, approaches of communication, computation, and algorithms, where the researchers are tracking these fields. In the current work, various schemes were proposed for cluster-based WSNs. The first is a proposed clustering method for communication nodes. Research is focused on improving time the required for executing various sensor tasks during data aggregation, as well the way nodes communicate with each other. An algorithm is selected to aggregate data in less time and reduce energy consumption. Secure authentication is also an important technique for WSNs because they exist in open environments that are easy to penetrate. The first achievement was the creation of a practical protocol using a cipher algorithm for nodes that worked in conjunction with packet relay between nodes, illustrating the possibility for designing a public key encryption technique for a cluster of WSN nodes in a hardware platform. The second goal of this study was to decrease the time required to complete distinct tasks in a WSN. We found that the time for these tasks could be reduced by increasing the number of PEGASIS chains.

One of the main challenges was designing a cluster of WSNs with handshake communication involving a BS and providing high quality encryption with three algorithms via hardware and software platforms. This was followed by testing the security of the WSN cluster. Different performance measures were analysed during the design and implementation phase, including energy consumption, the implementation of cryptographic algorithms, and the time required to break the code. It is possible to enhance the integrity of a network in the presence of reception and transmission errors. Furthermore, the different algorithms could be used to secure clusters of WSN nodes. Validation and authentication were two goals, wherein each node had a private key that it only shared with the nearest neighbour to ensure

authenticity. This project could be improved, expanded, and implemented using any type of sensor.

5.2 Future Research Directions

There are large number of avenues for future Research could be done. by adding assurances in terms of increasing energy consumption, consequently increasing the lifetime of the network. Also, the trade-off between these additional functions and their cost should be suited to the requirements of various applications. In addition, there is a growing desire to broaden the scope of WSNs, which requires numerous advanced algorithms. Every algorithm is specialized for specific tasks associated with the appropriate type of WSN and application. Before techniques that could be used in the future are presented, they should fulfil the following requirements:

- They ought to consider the limited resources available to a WSN and its capabilities, such as node network topologies and mobility.
- All spatial and temporal conditions surrounding a neighbouring node must be considered due to the use of heterogeneous data.
- Techniques must be free of computational complexity and must be easy to implement.

The following techniques may be used in the future:

- *Laser*: This is a fast data aggregation technique used in WSN nodes.
- *Optical fibre*: This can be used to release the frame from the BS and send it to the nodes.
- *BS*: Two or more BSs can be used to protect data without data loss, especially in the case of confidential data.
- *Web service*: This can be used to extend the network service and its applications.
- *Algorithm:* Special studies should be conducted so that appropriate algorithms for various tasks can be identified.
- *Transceiver*: Instead of using BT on the hardware platform, other IEEE wireless transceivers can be used.

The requirements of sensor networks should be considered when designing future studies.

REFERENCES

[1] Mahalik, N. P. (2007). *Sensor Networks and Configuration*. Springer-Verlag Berlin Heidelberg.

[2] NANDINI MUKHERJEE, SARMISTHA NEOGY and SARBANI ROY, 2016 "BUILDING WIRELESS SENSOR NETWORKS THEORETICAL & PRACTICAL PERSPECTIVES". by Taylor & Francis Group, LLC, pp. 15

[3] Mukherjee, N., Neogy, S. and Roy, S., 2015. *Building Wireless Sensor Networks: Theoretical and Practical Perspectives*. CRC Press.

[4] Parvathi, R.M.S., 2012. Secure authentication technique for data aggregation in wireless sensor networks. In *Journal of Computer Science*.

[5] Puccinelli, D., & Haenggi, M. (2005). "Wireless sensor networks: applications and challenges of ubiquitous sensing". *IEEE Circuits and systems magazine*, *5*(3), 19-31.

[6] Naranjo PG, Shojafar M, Mostafaei H, Pooranian Z, Baccarelli E. P-SEP: A prolonged stable election routing algorithm for energy-limited heterogeneous fog-supported wireless sensor networks. The Journal of Supercomputing. 2017 Feb 1;73(2):733-55.

[7] Raymond JW, Olwal TO, Kurien AM. Cooperative Communications in Machine to Machine (M2M): Solutions, Challenges and Future Work. IEEE Access. 2018 Feb 21.

[8] Nguyen TG, So-In C, Nguyen NG, Phoemphon S. A novel energy-efficient clustering protocol with area coverage awareness for wireless sensor networks. Peer-to-Peer Networking and Applications. 2017 May 1;10(3):519-36.

[9]] Liu X, Dong M, Liu Y, Liu A, Xiong NN. Construction Low Complexity and Low Delay CDS for Big Data Code Dissemination. Complexity. 2018;2018.

[10] Shorey, R., Ananda, A., Chan, M. C., & Ooi, W. T. (Eds.). (2006). *Mobile, wireless, and sensor networks: technology, applications, and future directions*. John Wiley & Sons.

[11] Jens-Peter Kaps.2006, *Cryptography for Ultra-Low Power Devices*. PhD thesis, ECE Department, Worcester Polytechnic Institute, Worcester, Massachusetts, USA.

[12] D.W. Carman, P.S. Kruus, and B.J. Matt.2000, Constraints and approaches for distributed sensor network security. NAI Labs Technical Report 00-010.

[13] Cerpa, A., Elson, J., Estrin, D., Girod, L., Hamilton, M., & Zhao, J. (2001). Habitat monitoring: Application driver for wireless communications technology. *ACM SIGCOMM Computer Communication Review*, *31*(2 supplement), 20-41.

[14] Chong, C. Y., & Kumar, S. P. (2003). Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, *91*(8), 1247-1256.

[15] Estrin, D., Culler, D., Pister, K., & Sukhatme, G. (2002). Connecting the physical world with pervasive networks. *IEEE pervasive computing*, *1*(1), 59-69.

[16] Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., & Pister, K. (2000). System architecture directions for networked sensors. *ACM SIGOPS operating systems review*, *34*(5), 93-104.

[17] Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., & Anderson, J. (2002, September). Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications* (pp. 88-97). Acm.

[18] Akyildiz, I. F., & Kasimoglu, I. H. (2004). Wireless sensor and actor networks: research challenges. *Ad hoc networks*, *2*(4), 351-367.

[19] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A survey on sensor networks. *IEEE Communications magazine*, *40*(8), 102-114.

[20] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci, E., 2002. Wireless sensor networks: a survey. *Computer networks*, *38*(4), pp.393-422.

[21] Ahmed, J., Siyal, M. Y., Tayyab, M., & Nawaz, M. (2015). *RFID-WSN integrated architecture for energy and delay-aware routing: A simulation approach*. Springer.

[22] Pour, N. K. (2016). Energy Efficiency in Wireless Sensor Networks. *arXiv preprint arXiv:1605.02393*.

[23] Rabaey, J. M., Ammer, M. J., Da Silva, J. L., Patel, D., & Roundy, S. (2000). PicoRadio supports ad hoc ultra-low power wireless networking. *Computer*, *33*(7), 42-48.

[24] Callaway Jr, E. H. (2003). *Wireless sensor networks: architectures and protocols*. CRC press.

[25] G. J. Pottie and W. J. Kaiser, Wireless Integrated Network Sensors, Communications of the ACM, vol. 43, no. 5, pp. 51-58, 2000.

[26] N. Correal and N. Patwari, Wireless Sensor Networks: Challenges and Opportunities, Virginia Tech MPRG Symposium on Wireless Personal Communication, Blacksburg, VA, USA, June 2001.

[27] Buratti, C., Conti, A., Dardari, D., & Verdone, R. (2009). An overview on wireless sensor networks technology and evolution. *Sensors*, *9*(9), 6869-6896.

[28] Zhu, N. (2013). *Simulation and optimization of energy consumption in wireless sensor networks* (Doctoral dissertation, Ecole Centrale de Lyon).

[29] Crossbow Technology Inc., "Mica2 datasheet," [Online], Available at: https://www.eol.ucar.edu/rtf/facilities/isa/internal/CrossBow/DataSheets/mica2.pdf.

[30] Atmel Corporation, "Atmega128 datasheet," [Online], Available at: http://www.atmel.com/Images/doc2467. pdf, 2011.

[31] "Micaz datasheet," [Online], Available at: http://www.openautomation.net/uploadsproductos/micaz_datasheet. Pdf.

[32] Zheng, J., & Jamalipour, A. (2009). *Wireless sensor networks: a networking perspective*. John Wiley & Sons.

[33] Eschenauer, L., & Gligor, V. D. (2002, November). A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security* (pp. 41-47). ACM.

[34] Newsome, J., Shi, E., Song, D., & Perrig, A. (2004, April). The sybil attack in sensor networks: analysis & defenses. In *Proceedings of the 3rd international symposium on Information processing in sensor networks* (pp. 259-268). ACM.

[35] Panda, M. (2014). Security in wireless sensor networks using cryptographic techniques. *AJER*, *3*, 50-6.

[36] Akyildiz, I. F., & Vuran, M. C. (2010). *Wireless sensor networks* (Vol. 4). John Wiley & Sons.

[37] Singhal, V., & Suri, S. (2014). Comparative study of hierarchical routing protocols in wireless sensor networks. *International Journal of Computer Science and Engineering*, *2*(5), 142-147.

[38] Ramlee, R. A., Tang, D. H. Z., & Ismail, M. M. (2012, September). Smart home system for disabled people via wireless Bluetooth. In *System Engineering and Technology (ICSET), 2012 International Conference on* (pp. 1-4). IEEE.

[39] Shen, H., & Li, Z. (2016). A Kautz-based wireless sensor and actuator network for real-time, fault-tolerant and energy-efficient transmission. *IEEE Transactions on Mobile Computing*, *15*(1), 1-16.

[40] Abbas, N. H., Ismaeel, T. Z., & Ibrahim, R. N. (2013). Optimization of Energy Consumption in Wireless Sensor Networks based on Nature-Inspired Algorithms. *International Journal of Computer Applications*, *77*(14).

[41] Hac, A. (2003). *Wireless sensor network designs*. John Wiley & Sons Ltd.

[42] D'Ausilio, A. (2012). Arduino: A low-cost multipurpose lab equipment. *Behavior research methods*, *44*(2), 305-313.

[43] Mowad, M. A. E. L., Fathy, A., & Hafez, A. (2014). Smart home automated control system using android application and microcontroller. *International Journal of Scientific & Engineering Research*, *5*(5), 935-939.

[44] Rani A, Kumar S. A survey of security in wireless sensor networks. In Computational Intelligence & Communication Technology (CICT), 2017 3rd International Conference on 2017 Feb 9 (pp. 1-5). IEEE.

[45] Al-Karaki JN, Kamal AE. Routing techniques in wireless sensor networks: a survey. IEEE wireless communications. 2004 Dec;11(6):6-28.

[46] 46-Mhatre V, Rosenberg C. Design guidelines for wireless sensor networks: communication, clustering and aggregation. Ad hoc networks. 2004 Jan 1;2(1):45-63.

[47] Ahmed J, Siyal MY, Tayyab M, Nawaz M. RFID-WSN integrated architecture for energy and delay-aware routing: A simulation approach. Springer; 2015 Mar 30. Book. pp. 9, 21, 23

[48] Mhatre V, Rosenberg C. Design guidelines for wireless sensor networks: communication, clustering and aggregation. Ad hoc networks. 2004 Jan 1;2(1):45-63.

[49] Singh G. A study of encryption algorithms (RSA, DES, 3DES and AES) for information security. International Journal of Computer Applications. 2013 Jan 1;67(19).

[50] Rajagopalan, Ramesh, and Pramod K. Varshney. (2006), "Data aggregation techniques in sensor networks: A survey."

[51] Almajadub, F., & Elleithy, K. (2014). Performance advancement of wireless sensor networks using low power techniques and efficient placement of nodes. *arXiv preprint arXiv:1407.0091*.

[52] Panda, M. (2014). Security in wireless sensor networks using cryptographic techniques. *AJER*, *3*, 50-6.

[53] 53 - Stallings, W., & Tahiliani, M. P. "*Cryptography and network security: principles and practice"* (2014). (Vol. 6). London: Pearson., Book, Fourth Edition". pp. 59, 63,

[54] Wang, L., Peng, D., & Zhang, T. (2015). Design of smart home system based on Wi-Fi smart plug. *Int. J. Smart Home*, *9*(6), 173-182.

[55] Weiser, M. (1991). The Computer for the 21 st Century. *Scientific American*, *265*(3), 94-105.

[56] Ma, J., Yang, L., Apduhant, B., Huang, R., Barolli, L., and Takizawa, M. (2005) "Towards a Smart World and Ubiquitous Intelligence: A Walkthrough from Smart Things to Smart Hyperspaces and UbicKids," Journal of Pervasive Computing and Communications, Vol.1, No.1, pp. 53-68

[57] Singh, G. (2013). A study of encryption algorithms (RSA, DES, 3DES and AES) for information security. *International Journal of Computer Applications*, *67*(19).

[58] Abedin, S., Tasbin, T., & Hira, A. (2017, February). Optical wireless data transmission with enhanced substitution Caesar Cipher WHEEL encryption. In *Electrical, Computer and Communication Engineering (ECCE), International Conference on* (pp. 552-556). IEEE.

[59] Calle Torres, M. G. (2006). *Energy consumption in wireless sensor networks using GSP* (Doctoral dissertation, University of Pittsburgh).

[60] D'Ausilio, A. (2012). Arduino: A low-cost multipurpose lab equipment. *Behavior research methods*, *44*(2), 305-313.

[61] Krishnamachari, B., Estrin, D., & Wicker, S. (2002, June). Modelling data-centric routing in wireless sensor networks. In *IEEE infocom* (Vol. 2, pp. 39-44).

[62] Manasrah AM, Al-Din BN. Mapping private keys into one public key using binary matrices and masonic cipher: Caesar cipher as a case study. Security and Communication Networks. 2016 Jul 25;9(11):1450-61.

[63] Calle Torres, M. G. (2006). *Energy consumption in wireless sensor networks using GSP* (Doctoral dissertation, University of Pittsburgh).

[64] Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000, January). Energy-efficient communication protocol for wireless microsensor networks. In *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on* (pp. 10-pp). IEEE.

[65] Govinda, K. Multilevel cryptography technique using graceful codes. Journal of Global Research in Computer Science. 2011 Aug 3;2(7):1-5.

[66] Bruen AA, Forcinito MA. Cryptography, information theory, and error-correction: a handbook for the 21st century. John Wiley & Sons; 2011 Sep 28. Ch 2, pp. 18.

[67] Kalpana P, Kamau, L., Langat, K. and Muriithi, C., 2018. The use of RC4 Encryption to Provide Privacy for Smart Meters. *JOURNAL OF SUSTAINABLE RESEARCH IN ENGINEERING*, *4*(2), pp.69-75. A., 2018.

[68] Hamann M, Krause M, Meier W, Zhang B. Design and analysis of small-state grain-like stream ciphers. Cryptography and Communications. 2018 Sep 1;10(5):803-34.

[69] Ahmad SA. Security in 802.11 Wireless Networks using IWEP. networks. 2018 Apr.

[70] Hameed ME, Ibrahim MM, Manap NA. Review on Improvement of Advanced Encryption Standard (AES) Algorithm based on Time Execution, Differential Cryptanalysis and Level of Security. Journal of Telecommunication, Electronic and Computer Engineering (JTEC). 2018 Jan 15;10(1):139-45.

[71] Dohler, M., Barthel, D., Maraninchi, F., Mounier, L., Aubert, S., Dugas, C., Buhrig, A., Paugnat, F., Renaudin, M., Duda, A. and Heusse, M., 2007, June. The ARESA project: Facilitating research, development and commercialization of WSNs. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON'07. 4th Annual IEEE Communications Society Conference on* (pp. 590-599). IEEE.

APPENDICES

Appendix A. Hardware Platform

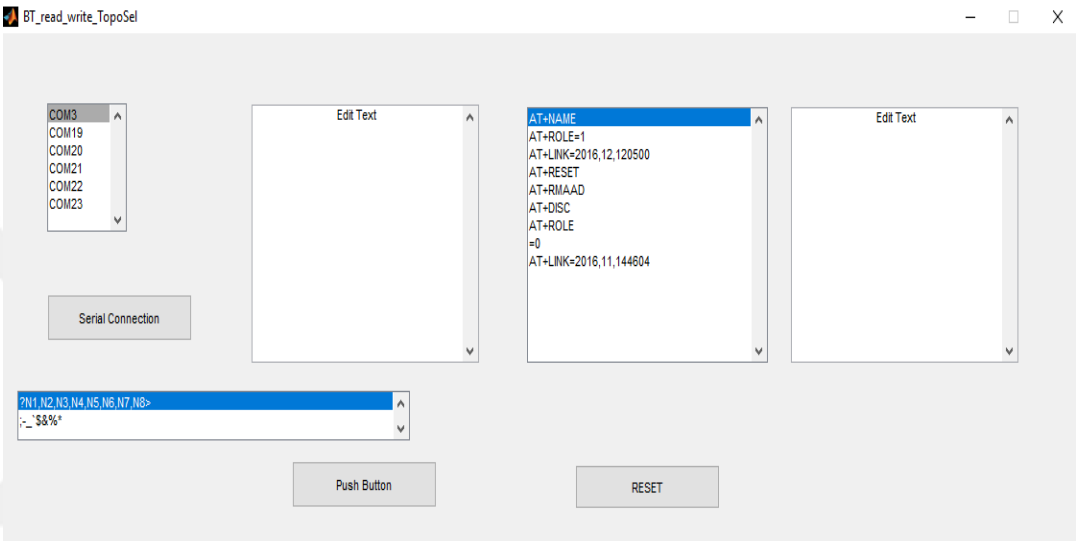

Figure A1. Interface running the process.

Figure A2. Power supply with actual nodes.

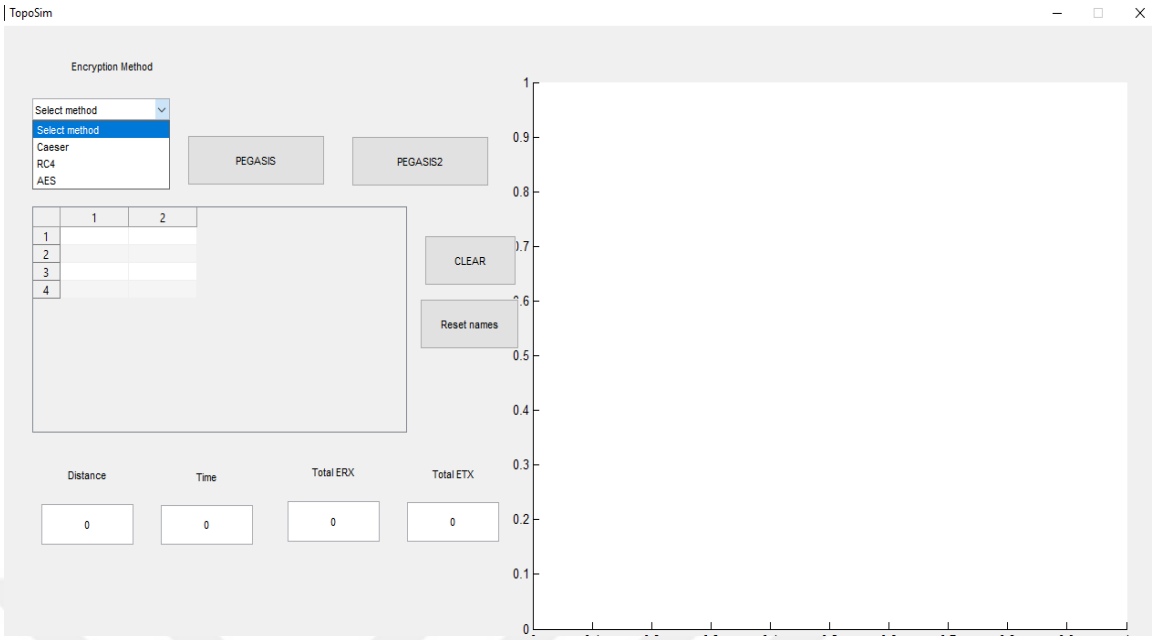

Figure B1. Interface simulation.

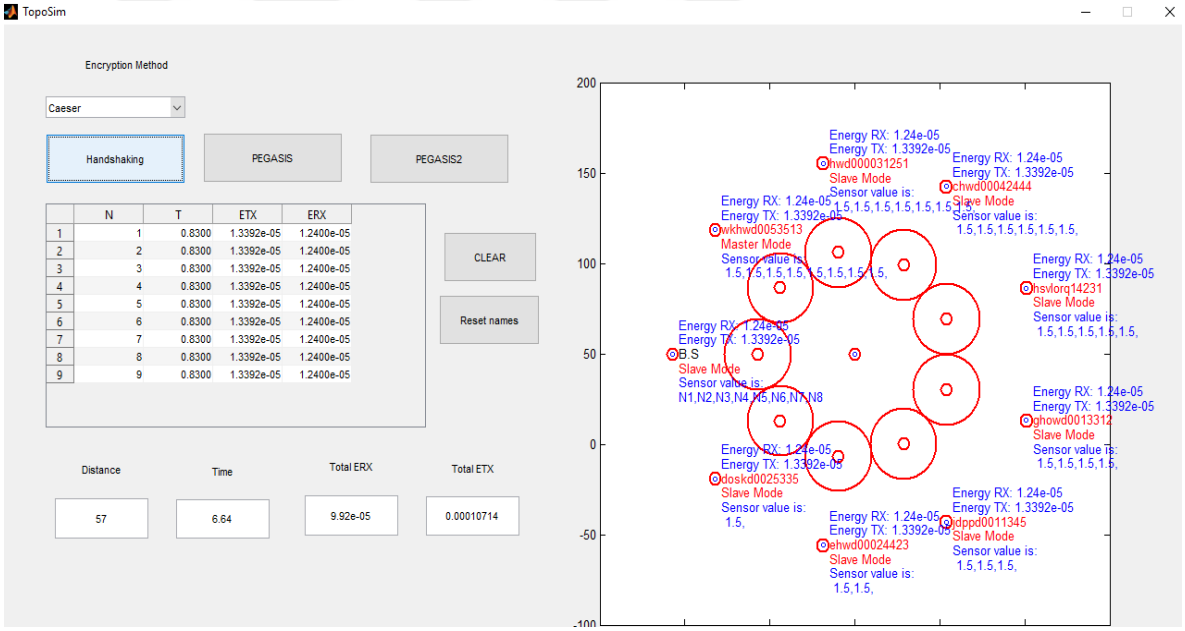

Figure B2. Interface of the encryption process for cluster nodes with the Caesar cipher completed during the first cycle.

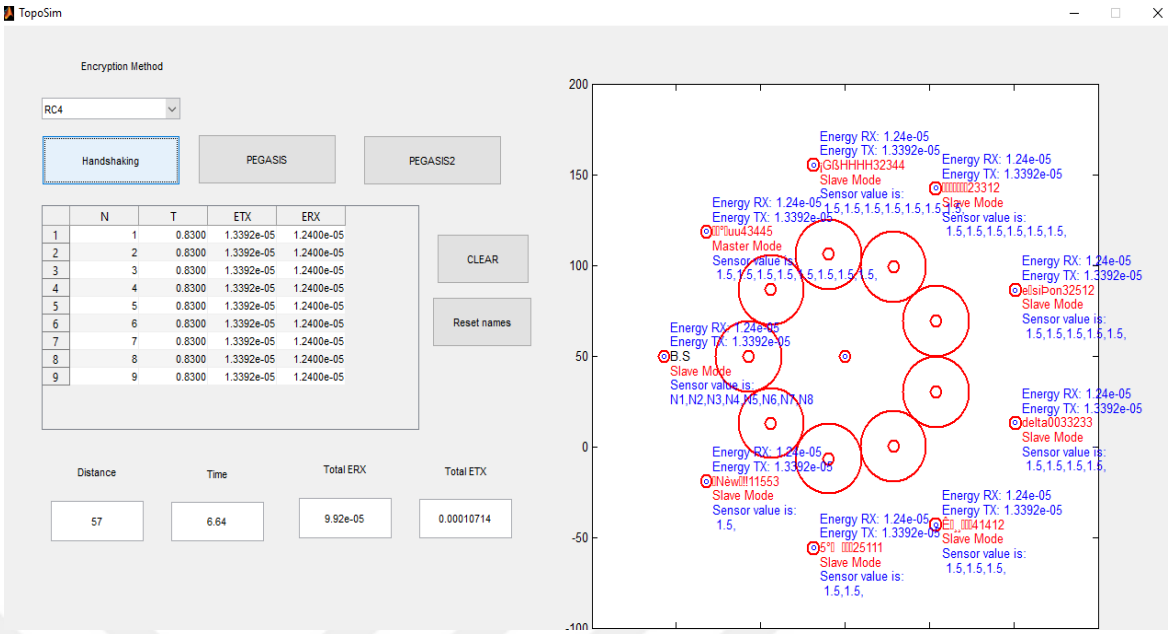

Figure B3. Interface of the Implementation of the RC4 algorithm to encrypt the named nodes.

Hex	Dec	Char		Hex	Dec	Char		Hex	Dec	Char		Hex	Dec	Char	
0x00	0	NULL	null	0x20	32	Space		0x40	64	@		0x60	96	`	
0x01	1	SOH	Start of heading	0x21	33	!		0x41	65	A		0x61	97	a	
0x02	2	STX	Start of text	0x22	34	"		0x42	66	B		0x62	98	b	
0x03	3	ETX	End of text	0x23	35	#		0x43	67	C		0x63	99	c	
0x04	4	EOT	End of transmission	0x24	36	$		0x44	68	D		0x64	100	d	
0x05	5	ENQ	Enquiry	0x25	37	%		0x45	69	E		0x65	101	e	
0x06	6	ACK	Acknowledge	0x26	38	&		0x46	70	F		0x66	102	f	
0x07	7	BELL	Bell	0x27	39	'		0x47	71	G		0x67	103	g	
0x08	8	BS	Backspace	0x28	40	(		0x48	72	H		0x68	104	h	
0x09	9	TAB	Horizontal tab	0x29	41	)		0x49	73	I		0x69	105	i	
0x0A	10	LF	New line	0x2A	42	*		0x4A	74	J		0x6A	106	j	
0x0B	11	VT	Vertical tab	0x2B	43	+		0x4B	75	K		0x6B	107	k	
0x0C	12	FF	Form Feed	0x2C	44	,		0x4C	76	L		0x6C	108	l	
0x0D	13	CR	Carriage return	0x2D	45	-		0x4D	77	M		0x6D	109	m	
0x0E	14	SO	Shift out	0x2E	46	.		0x4E	78	N		0x6E	110	n	
0x0F	15	SI	Shift in	0x2F	47	/		0x4F	79	O		0x6F	111	o	
0x10	16	DLE	Data link escape	0x30	48	0		0x50	80	P		0x70	112	p	
0x11	17	DC1	Device control 1	0x31	49	1		0x51	81	Q		0x71	113	q	
0x12	18	DC2	Device control 2	0x32	50	2		0x52	82	R		0x72	114	r	
0x13	19	DC3	Device control 3	0x33	51	3		0x53	83	S		0x73	115	s	
0x14	20	DC4	Device control 4	0x34	52	4		0x54	84	T		0x74	116	t	
0x15	21	NAK	Negative ack	0x35	53	5		0x55	85	U		0x75	117	u	
0x16	22	SYN	Synchronous idle	0x36	54	6		0x56	86	V		0x76	118	v	
0x17	23	ETB	End transmission block	0x37	55	7		0x57	87	W		0x77	119	w	
0x18	24	CAN	Cancel	0x38	56	8		0x58	88	X		0x78	120	x	
0x19	25	EM	End of medium	0x39	57	9		0x59	89	Y		0x79	121	y	
0x1A	26	SUB	Substitute	0x3A	58	:		0x5A	90	Z		0x7A	122	z	
0x1B	27	FSC	Escape	0x3B	59	;		0x5B	91	[		0x7B	123	{	
0x1C	28	FS	File separator	0x3C	60	<		0x5C	92	\		0x7C	124	|	
0x1D	29	GS	Group separator	0x3D	61	=		0x5D	93	]		0x7D	125	}	
0x1E	30	RS	Record separator	0x3E	62	>		0x5E	94	^		0x7E	126	~	
0x1F	31	US	Unit separator	0x3F	63	?		0x5F	95	_		0x7F	127	DEL	

Figure B5. ASCII Table Used in Our Implementation

CURRICULUM VITAE

PERSONAL INFORMATION

Name Surname : **ASMAA HAMMOODI**

Date of Birth : **20/5/1967**

Phone : **00905534647190**

E-mail : **asmaaphd11@gmail.com**

EDUCATION

Bachelor : B.S. degree in Electronic and Communication Engineering, 1990 University of Technology, Baghdad, Iraq.

Master's degree : M.Sc. degree in Electronic engineering, 2003,University of Technology, Baghdad, Iraq.

WORK EXPERIENCE

Research Assist. **:** Training Engineer and **Lecturer** in university of technology, Baghdad

TOPICS OF INTEREST

- **Design a PID controller of BLDC motor by using hybrid genetic-immune**
- **Implementation of Virtual Private Network by Linux Operation system**
- **Wireless Sensor Networks Nodes Distributed in Shapes of Polygons for Promote Distance, Time Delay and Optimization Energy Consumption via Bluetooth**