BUILDING A LOW COST LOW POWER WIRELESS SENSOR NETWORK FOR AGRICULTURE PURPOSE

A THESIS
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

**MASTER OF ELECTRONICS
& COMMUNICATION
ENGINEERING**

BY

ANINDITA MONDAL

RegistrationNo:128925OF2010-2011
ExaminationRollNo:M4ETC1610

Under the guidance of

PROF. ITI SAHA MISRA

DEPARTMENT OF
ELECTRONICS &TELECOMMUNICATION
ENGINEERING

**JADAVPUR UNIVERSITY
KOLKATA-700032**

MAY 2016

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

This is to certify that the dissertation entitled "**Building a Low Cost, Low Power Wireless Sensor Network for Agricultural Application**" has been carried out by **Anindita Mondal (University Registration No.: 128925 of 2014-2015)** under my guidance and supervision and be accepted in partial fulfillment of the requirements for the degree of Master of Technology in Electronics and Telecommunication. The research results presented in the thesis have not been included in any other paper submitted for any degree to any other University or Institute.

(Prof. Iti Saha Misra)
Supervisor,
Department of Electronics and
TelecommunicationEngineering,
Jadavpur University,
Kolkata -700032.

COUNTERSIGNED

(Prof. P.Venkateswaran)
Head of the Department,
Department of Electronics
and Telecommunication,
Jadavpur University,
Kolkata -700032.

(Prof. Sivaji Bandyopadhyay)
Dean,
Faculty of Engineering
and Technology,
Jadavpur University,
Kolkata 700032.

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATION

Certificate of Approval *

The foregoing thesis is hereby approved as a creditable study of an engineering subject, carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

Committee on final examination
for the evaluation of the Thesis

---------------------------- ----------------------------

Signature of the Examiner Signature of the Supervisor

*Only in case the thesis is approved.

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC THESIS

I hereby declare that this thesis titled "**Building aLow Cost Low Power Wireless Sensor Network for Agricultural Application**" contains literature survey and original research work by the undersigned candidate, as part of his Degree of Master of Technology in Electronics and Telecommunication Engineering.

All information have been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: ANINDITA MONDAL

Examination Roll No: M4ETC1610

Thesis Title: Building a Low Cost Low Power Wireless Sensor Network for Agricultural Application.

Acknowledgement

I would like to express my deep gratitude to the Almighty God for giving me the strength and keeping me agile to do this work.

I would like to express my sincere thanks to Prof. Iti Saha Misra for her valuable guidance, advice and supervision that have served as a pillar of strength and confidence throughout my M.E project. I am privileged to witness her enthusiastic and dedicated interest towards advanced research that motivates me and enriches my growth as a student and a researcher. I am indebted to her for her unflinching support and the innovative ideas and extraordinary experiences that she have shared with me during the course of this research.

I would like to extend my appreciation to my friends and seniors without whose support and cooperation, this thesis would never have been possible. I convey my earnest thanks to Tamal Chakraborty, Tanumoy Manna, Sreya Ghosh, Sudipta Dey, Subhajit Mukherjee and Sagar Bose, for their generous assistance in writing the thesis.

I must acknowledge the academic resources that I have got from Jadavpur University. Iwould like to thank administrative and technical staff members of the Departmentwho have been kind enough to advise and help in their respective roles.

Last but not the least, I remain ever grateful to my parents and my grandparents who have selflessly devoted their lives towards fulfillment of my dreams. I also owe my deepest gratitude to my elder sister who has always stood by me through thick and thin. I dedicate this thesis to them.

Place: Kolkata ---------------------------------------

Date:

(Anindita Mondal)

Department of Electronics and Telecommunication Engineering,

Jadavpur University.

Abstract

The integration of sensing, processing and wireless communication of wireless sensor network has many applications around the globe, starting from wildlife environments to urban areas. Designing and implementation of Wireless Sensor Network in hardware domain is a challenging task because each specific application may require different hardware and software with some constraints.

In India improper management of water resources for agriculture is one of the main issues for failure to crop yield. Hence there is a demand of water management system for agriculture which will maximize the crop yield. Since the end users are farmers, the cost of the product holds great importance and it has to be minimized.

The goal of this thesis is to design and fabricate components of WSN i.e. gateway, router and sensor node according to the required specification of agriculture application. Selection of devices and integration of these devices for hardware implementation has been successfully performed. After fabrication, field implementation has been made for testing and analysis of data. In field implementation the sensor data is collected at the sensor node which is send to the gateway via router. When the sensor data is above certain threshold value the actuator is released to bring the sensed value below threshold value and hence develop a closed-loop control system. Visualization software has been developed to log the data at the gateway with the help of Raspberry Pi module. A closed loop control system has been developed by deploying proposed WSN model and the end results shows that it is working according to the required specification.

Table of Contents

List of Tables

List of Figures

Chapter 1

INTRODUCTION

1.1 Introduction to Wireless Sensor Network

In the present scenario, wireless sensor networks is an emerging technology to be deployed at an accelerated pace. It is not unreasonable to expect that in 10-15 years that the world will be covered with wireless sensor networks with access to them via the Internet. This new technology is exciting with incredible potential for numerous application areas including environmental monitoring, disaster management, precision agriculture, home controlling, predictive maintenance, energy saving smart grid and many others.

The theory of wireless sensor networks is based on a simple equation:

Sensing + Routing + Processing +Actuation =Thousands of potential applications.

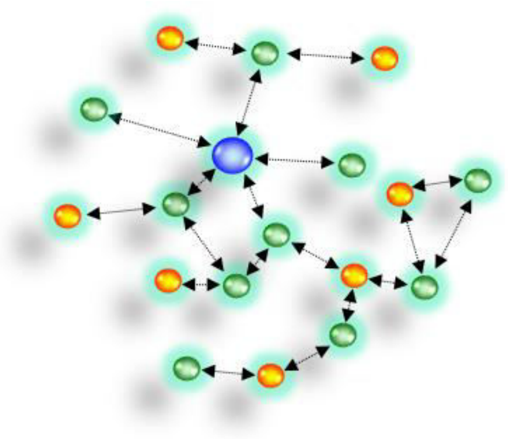


Fig. 1.1 Conceptual view of WSN

The modern field of wireless sensor networks combines sensing, computation, and communication into a single tiny device. The power of wireless sensor networks lies in the ability to deploy large numbers of tiny nodes that assemble and configure themselves.

A typical WSN mainly consists of spatially distributed random sensor nodes which independently work and collect required data which is then sent to the central analyzing center i.e. gateway where the data is collected and analyzed for sending back data to the sensor node for actuation i.e. taking for action.

1.2 Motivation

India's water crisis is predominantly a manmade problem. The climate of India is not particularly dry nor is it lacking in rivers and groundwater. Extremely poor management, industrial and human waste have caused this water supply crunch and rendered what water is available practically useless due to huge quantity of pollution. India is among the top growers of agriculture produce in the world and therefore consumption of water for irrigation is amongst the highest.

The impact of water deficit on crop growth is more than all other environmental effects. It is highly required to minimize water consumption for agriculture purpose either manually or through automation.

In order to accurately get extent of water deficit and therefore realize effective and water-saving irrigation, the application of wireless sensor network for precision irrigation system is explored.

Since WSN has huge applicative potential it offer a practical solution to the problem stated above. A closed-loop controlling system can be designed for automatic monitoring and controlling different environmental parameters which will lead to increase in overall yield and minimize wastage of water. Considering the economic condition of Indian farmers, cost of the product is one of the important issues for designing the hardware. Henceforth our focus is to design a low cost product for efficient use of water in agriculture. As we talk about designing an application specific product the first thing comes into mind is microcontroller as processing unit due to its cheaper cost and user friendly functionality. Microcontroller is considered as the heart of the project since it controls the transceiver module for transmission and reception of data and other peripheral devices required for this project. The entire designing, selection of devices and implementation of the product in real-time is a rigorous process.

The hard work and effort made for this project will be best rewarded if it can bring smile to the faces of Indian farmers and increase the Indian economy.

1.3 Objective of the Thesis

The main objective of the thesis is to design a low cost, low power WSN for the field of agriculture application. Though this product will be equally applicable to many other fields like weather monitoring, smart home controlling, asset tracking, enhanced safety and security etc. New sensor nodes have been designed to minimize the cost of entire WSN rather than using existing wireless sensor mote available in the market at much higher cost which will hamper the main motive of the thesis to minimize cost with required functionality. If the cost is minimized, eventually the Indian farmers will be benefited.

The cost of each component of WSN i.e. gateway, router and sensor node has to be minimized as much as possible.

The gateway where the sensor data is collected and processed used to be a Personal Computer which cost around Rs. 30000/- approximately. Here in this thesis a new technique has been applied to minimize the cost by replacing the traditionally employed Personal Computer by Raspberry Pi which has 900MHz quad-core ARM Cortex-A7 CPU. The astonishingly cheaper cost of Raspberry Pi around Rs 3000/- and its ability to work as microcomputer lead the foundation for incorporating it in this project to minimize the cost through a considerable margin.

The selection of electronic component and integration of these components to develop a cost effective end product for successful field implementation is the main research issues that has to be dealt with.

After completion of hardware designing and field implementation, several experiments has to be performed to check the performance metrics of WSN such as packet drop, average end-to-end delay, communication range, network lifetime and total energy consumption.

1.4 Outline of the Thesis

In this chapter an overview of WSN has been presented. Motivation and objective for this thesis work has been briefly discussed in this chapter.

The objective of chapter 2 is to present the background information necessary for foundation of this research work.

In chapter 3, the research methodology adapted during the research and objective framed for research has been discussed. Initially the components required for designing of gateway, router and sensor node has been shown followed by selection of component among other components available in the market has been discussed. Calculation of cost for each device in WSN is tabulated. Design methodologies mention the step by step proceeding for hardware interfacing, coding and its testing. Integration of module wise operation has been done in the last phase of this chapter.

In chapter 4, result after execution of code in the Arduino platform has been discussed. Required changes that have been made to get the desired result has been discussed in this chapter.

In chapter 5, conclusion from the result has been shown. Future scope of the project has discussed to make the project better

Chapter 2

Background Information

2.1 Wireless Communication

Wireless communication is the fastest growing segment of the communications industry for its flexibility and high demand in the market. Cellular systems have experienced exponential growth over the last decade. Many new applications including wireless sensor networks, automated highways and factories, smart homes and appliances and remote telemedicine, are emerging from research idea to concrete systems [1].Here in this project, we are mainly going to focus on Wireless Sensor Networks for development of low cost, low power WSN for agriculture purpose. The background information essentially for development of the product has been discussed in this chapter.

2.2 Wireless Sensor Networks

Wireless Sensor Networks consists of individual nodes that are able to interact with their environment by sensing or controlling physical parameter; these nodes have to collaborate in order to fulfill their tasks as usually, a single node is incapable of doing so; and they use wireless communication to enable this collaboration [2].

"A sensor network is a deployment of massive numbers of small, inexpensive, self-powered devices that can sense, compute, and communicate with other devices for the purpose of gathering local information to make global decisions about a physical environment".

2.2.1 Evolution of Sensor Network

Sensor network development was initiated by the United States during the Cold War. A network of acoustic sensors was placed at strategic locations on the bottom of the ocean to detect and track Soviet submarines. This system of acoustic sensors was called the Sound Surveillance System (SOSUS). Human operators played an important role in these systems. The sensor network was wired network that did not have the energy bandwidth constraints of wireless system.

Modern research on sensor networks started around 1980 with the Distributed Sensor Networks (DSN) program at the Defense Advanced Research Projects Agency (DARPA) [2]. These included acoustic sensors communication (a high-level protocol that link processes working on a common application in a resource-sharing network), processing techniques, algorithms (including self-location algorithms for sensors), and distributed software (dynamically modifiable distributed systems and language design).

2.3 Wireless Ad-hoc Network

In ad-hoc network each and every nodes are allowed to communicate with each other without any fixed infrastructure. This is actually one of the features that differentiate between ad-hoc and other wireless technology like cellular networks and wireless LAN which actually required infrastructure based communication i.e. through some base station.

Wireless sensor network are one of the category belongs to ad-hoc networks. Sensor network are also composed of nodes. Here actually the node has a specific name that is “Sensor” because these nodes are equipped with smart sensors. A sensor node is a device that converts a sensed characteristic like temperature, vibrations, pressure into a form recognize by the users. Wireless sensor networks nodes are less mobile than ad-hoc networks. So mobility in case of ad-hoc is more [3]. In wireless sensor network data are requested depending upon certain physical quantity. A sensor consists of a transducer, an embedded processor, small memory unit and a wireless transceiver and all these devices run on the power supplied by an attached battery.

2.4 Characteristic of Ad-hoc & Sensor Network

Table 2.1 Comparison between Ad-Hoc Network and Sensor Network

Parameters	Ad-Hoc Network	Sensor Network
Infrastructure	Ad-hoc is infrastructure less network	Sensor Network is infrastructure less network
Mobility	Mobility of nodes in ad-hoc networks is more.	Mobility of nodes in wireless sensor networks is less.
Multi-hoping	Has multi-hop ability due to which energy associated with each node can be conserved.	Provision of multi-hop is there in wireless sensor network.
Openness	Ad-hoc network access information and services without geographic position.	Wireless sensor network access information and services regardless of Geographic position.
Topology	Traditional ad hoc networks are based on point-to-point communications.	The topology of WSN changes very frequently.

2.5 Existing Popular Wireless Technology / Protocols

There are many different wireless protocols, but the ones that most popular are Wi-Fi and Bluetooth because these are used in common devices i.e. mobile phones and computers. There is a third alternative called ZigBee that is designed for control and instrumentation.

This section introduces the Bluetooth, UWB, ZigBee, and Wi-Fi protocols, which corresponds to the IEEE 802.15.1 802.15.3, 802.15.4, and 802.11a standards, respectively. The IEEE defines only the PHY and MAC layers in its standards. For each protocol, separate alliances of companies worked to develop specifications covering the network, security and application profile layers so that the commercial potential of the standards could be realized [4].

2.5.1 Wi-Fi or IEEE 802.11 Standard [5]

IEEE 802.11 is a set of Standards for Wireless local area network (WLAN) computer communication, developed by the IEEE LAN/MAN Standards Committee (IEEE 802) in the 5 GHz and 2.4 GHz public spectrum bands. Although the terms 802.11 and Wi-Fi are used often interchangeably, the Wi-Fi Alliance uses the term Wi-Fi to define a slightly different set of overlapping standards. Wi-Fi is a direct replacement for a wired Ethernet cable and it is used to minimize cost required for connection through wires and its flexibility to the end user. The benefit of Wi-Fi is that it can connect to an existing network hub or router, which means that a PC doesn't have to be left on to access a device using Wi-Fi. Remote access products like IP cameras use Wi-Fi so they can be connected to a router and accessed across the Internet. Wi-Fi is useful but not simple to implement unless one just wants to connect a new device to the existing network.

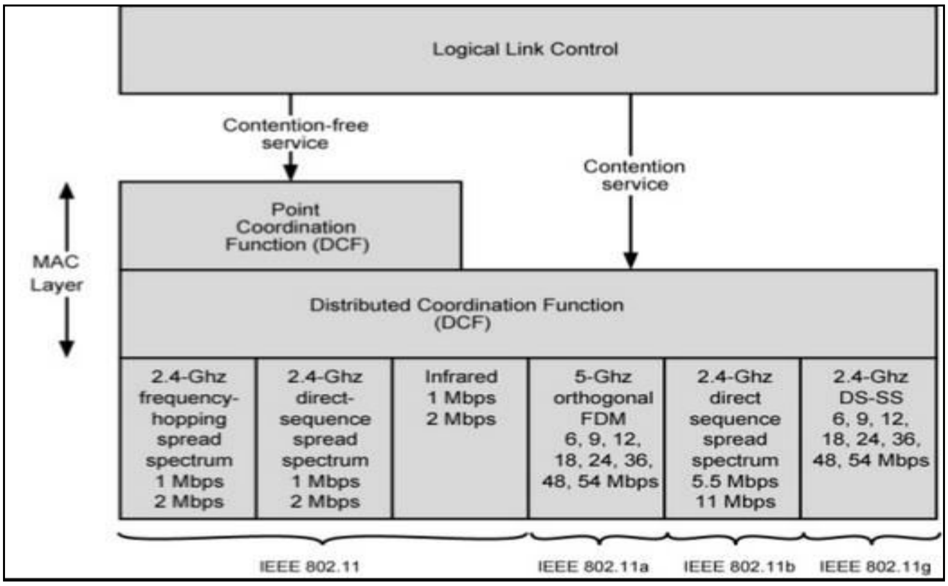


Fig. 2.1 IEEE 802.11 protocol architecture [6]

The 802.11 family includes over-the-air modulation techniques that use the same basic protocol. The most popular are those defined by the 802.11b and 802.11g protocols, and are amendments to the original standard. 802.11a was the first wireless networking standard, but 802.11b was the first widely accepted one, followed by 802.11g and 802.11n [6].

Table 2.2 Specification of 802.11 Standards

Standard	Speed (Mbps)	Range (Meter)	Frequency(GHz)
802.11b	11	45.72	2.4
802.11g	54	15.24	2.4
802.11a	54	15.24	5
802.11n	300/450	51.85	2.4/5

Security was originally purposefully weak due to export requirements of some governments, and was later enhanced via the 802.11i amendment after governmental and legislative changes. 802.11n [7] is a new multi-streaming modulation technique that is still under draft development, but products based on its proprietary pre-draft versions are being sold.

Other standards in the family (c-f, h, and j) are service amendments and extensions or corrections to previous specifications.

The Wi-Fi Alliance is a global, non-profit industry association of more than 600 member companies devoted to promoting the growth of wireless Local Area Networks (WLANs). With the aim of enhancing the user experience for wireless portable, mobile and home entertainment devices, the Wi-Fi Alliance's testing and certification programs help to ensure the interoperability of WLAN products based on the IEEE 802.11 specification.

Limitations:

- **Security concerns:** Though typically very easy to set up, securing your Wi-Fi network requires more effort. Since 1990s, Wi-Fi security algorithms have undergone multiple upgrades with outright depreciation of older algorithms and significant revision to newer algorithms. Increasing computer power and exposed vulnerabilities have rendered older encryption standards at risk. The following section briefly explains the history of Wi-Fi security and serves to highlight what should be used in the present scenario.

I. **Wired Equivalent Privacy (WEP)** - is the most widely used Wi-Fi security algorithm in the world. Despite revisions to the algorithm and an increased key size, over time numerous security flaws were discovered in the WEP standard and, as computing power increased, it became easier and easier to exploit them. The Wi-Fi Alliance officially retired WEP in 2004.

II. **Wi-Fi Protected Access (WPA) -** was formally adopted in 2003, a year before WEP was officially retired. The most common WPA configuration is WPA-PSK (Pre-Shared Key). The keys used by WPA are 256-bit; a significant increase over the 64-bit and 128-bit keys used in the WEP system .Some of the significant changes implemented with WPA included message integrity checks and the Temporal Key Integrity Protocol (TKIP). TKIP employs a per-packet key system that was radically more secure than fixed key used in the WEP system. TKIP was later superseded by Advanced Encryption Standard (AES).

III. **Wi-Fi Protected Access II (WPA2) -** WPA was replaced by WPA2 in the year 2016. One of the most significant changes between WPA and WPA2 was the mandatory use of AES algorithms and the introduction of CCMP (Counter Cipher Mode with Block Chaining Message Authentication Code Protocol) as a replacement for TKIP [8].

- **Interference from other devices:** In present scenario, many other types of devices emitting in the unlicensed band dwarf the number of 802.11 devices. These devices include microwave ovens, cordless phones, Bluetooth devices, wireless video cameras, outdoor microwave links, wireless game controllers, Zigbee devices, fluorescent lights, WiMAX, and so on. Even bad electrical connections can cause broad RF spectrum emissions. These non-802.11 types of interference typically don't work cooperatively with 802.11 devices, and can cause significant loss of throughput [9].

- **Lack of support for high-quality media streaming:** Even the fastest current Wi-Fi standards are pushed beyond their limit when trying to handle some of today's high-end media. High-definition audio and video files are bandwidth and timely delivery-intensive, and typical wireless networks have neither the transfer speeds nor the consistency to transfer them flawlessly. This problem is further compounded if there are multiple devices connected to the same access point because the bandwidth must be divided between all of the equipment [10].

2.5.2 Bluetooth or IEEE 802.15.1 Standard

The Bluetooth protocol enables short range; low power and low-cost wireless communications between Bluetooth devices. Bluetooth is generally used for point to point communication, although Bluetooth networks can be established quite easily. Typical applications we are all familiar with allow data transfer from mobile phones to PCs. Bluetooth wireless is the best solution for these point to point links, as it has transfer rate of around 3Mbps and line of sight is not required for transmission. Designed primarily as a cable replace technology, it enables ad-hoc wireless networking, which allows formation of a network without base stations [11]. The Bluetooth radio uses a low-powered transceiver that supports digital wireless communications at the 2.4GHz ISM band. The main features of Bluetooth are:

- Uses the spread-spectrum, frequency hopping, and full-duplex signal at a nominal rate of 1600 hops/sec.
- Adaptive frequency hopping (AFH) capability
- Data rate supported is up to 3 Mbps
- The operating range depends on the device class:

✓ **Class 3 radios** - have a range of up to 1 meter or 3 feet

✓ **Class 2 radios** - most commonly found in mobile devices - have a range of 10 meters or 30 feet

- ✓**Class 1 radios** - used primarily in industrial use cases - have a range of 100 meters or 300 feet

2.5.3 ZigBee or IEEE 802.15.4

ZigBee technology is a simpler, less expensive, low data rate, low power consumption; wireless mesh networking protocol targeted towards wireless personal area networks (WPANs), such as automation and remote control applications that require a low data rate, long battery life, and secure networking. The low cost allows the technology to be widely deployed in wireless control and monitoring applications, the low power-usage allows longer life with smaller batteries, and the mesh networking provides high reliability and larger range. IEEE 802.15.4 committee started working on a low data rate standard a short while later. Then the ZigBee Alliance and the IEEE decided to join forces and ZigBee is the commercial name for this technology. The standard defines the channel access mechanism, acknowledged frame delivery, network association and disassociation [12].

ZigBee is expected to provide low cost and low power connectivity for equipment that needs battery life as long as several months to several years, depending on application requirements but does not require data transfer rate highly and not designed for video streaming or large file transfer. It can be implemented in larger mesh network. ZigBee compliant wireless device are expected to transmit 5-300 meters, depending on the RF environment and the power output consumption required for a given application, and will operate in the unlicensed RF worldwide.

ZigBee uniquely offers low-latency communication between devices without the need for network synchronization delays as required by Bluetooth, for instance and hence best suited for real time applications requirements [13].

ZigBee creates robust self-healing, self-forming, wireless mesh networks. ZigBee mesh networks let all participating devices communicate with one another, and act as repeaters transferring data between devices by multi-hopping mechanism.

The ZigBee Alliance, the standards body which defines ZigBee, also publishes application profiles that allow multiple OEM vendors to create interoperable products. The current list of application profiles either published or in the works are: Home Automation, ZigBee Smart Energy, Telecommunication Applications, and Personal Home and Hospital Care etc. It is developed for applications with relaxed throughput requirements which cannot handle the power consumption of heavy protocol stacks.

IEEE and ZigBee Alliance have been working closely to specify the entire protocol stack. IEEE 802.15.4 focuses on the specification of the lower two layers of the protocol (physical and data link layer). On the other hand, ZigBee Alliance aims to provide the upper layers of the protocol stack (from network to the application layer) for interoperable data networking, security services and a range of wireless home and building control solutions, provide interoperability compliance testing, marketing of the standard, advanced engineering for the evolution of the standard. This will assure consumers to buy products from different manufacturers with confidence that the products will work together [14].

- Protocol features include:
- Service discovery
- Master / Slave topology
- Supports star or peer-to-peer operation
- Automatic network configuration
- Dynamic slave device addressing
- Up to 254 (+ master) network nodes
- TDMA slots can be allocated
- Full handshaking for packet transfers (reliable data transfer)
- CSMA/CA channel access mechanism
- Data rate of:
    - 20kbps at 868 MHz,
    - 40kbps at 915 MHz
    - 250kbps at 2.4 MHz
- Channels:

    - 16 channels in the 2450 MHz band
    - 10 channels in the 915 MHz band
    - 1 channel in the 868 MHz band
  - Power management features
  - Allocation of guaranteed time slots (GTSs)
  - Carrier sense multiple access with collision avoidance (CSMA-CA) channel access
  - Fully acknowledged protocol for transfer reliability
  - Low power consumption
  - Energy detection (ED)
  - Link quality indication (LQI)

The basic framework conceives a transfer rate of 250 kbit/s. Tradeoffs are possible to favor more radically embedded devices with even lower power requirements, through the definition of not one, but several physical layers. Lower transfer rates of 20 and 40 kbit/s were initially defined; with the 100 kbit/s rate being added in the current revision .Even lower rates can be considered with the resulting effect on power consumption. As already mentioned, the main identifying feature of IEEE 802.15.4 among WPAN's is the importance of achieving extremely low manufacturing and operation costs and technological simplicity, without sacrificing flexibility or generality.

Other important features include real-time suitability by reservation of guaranteed time slots, collision avoidance through CSMA/CA and integrated support for secure communications. Devices also include power management functions such as link quality and energy detection.

Limitation:

- Low range and low power characteristics make ZigBee to be used vastly for WPANs. It will be difficult to use ZigBee in case of high data rate transmissions.
- In practice, not all devices in a network can be battery powered, particularly those that need to be switched on all the time (and cannot sleep), such as routers and coordinators. Such devices can often be installed in a mains-powered appliance that is permanently connected to the mains supply. This stimulates the need to install a dedicated mains power connection for the network device.
- The 2.4GHz band provides the highest bit rate of 250Kbps in IEEE 802.15.4 PHY specs. The physical layer supports transfer of only small sized packets limited to 127 bytes. Due to overhead at the network, MAC and physical layers, each packet may contain no more than 89 bytes for application data. This leads to fragmentation of bit streams larger than 89 bytes. The networking layer does not perform fragmentation. Therefore, the fragmentation and reassembly should be handled at the application layer. A flow control mechanism is also needed to acknowledge and request retransmission of missing fragments above the network layer.
- Using IEEE 802.15.4 the network size is limited to seven nodes if GTSs is to be utilized. Further, because of the non-existing support for dynamic GTS allocation in real-time, the node with the highest sample rate will set the system sample rate. As nodes must have the same sample rate this leads to many nodes being over-sampled. Higher sample rate will increase the power consumption. Furthermore, the steps between the available system sample rates are rather big. Hence, all nodes must be over-sampled to match the required application sample frequency. As a result of these limitations, the 802.15.4 standard shall not be employed in a wireless control system in the heavy duty application.

2.5.3.1 ZigBee Protocol Architecture

The architecture is defined in terms of a number of blocks in order to simplify the standard. These blocks are called layers. Each layer is responsible for one part of the standard and offers services to the higher layers. The layout of the blocks is based on the open systems interconnection (OSI) seven-layer model.

The interfaces between the layers serve to define the logical links that are described in this standard. A ZigBee device comprises a PHY, which contains the radio frequency (RF) transceiver along with its low-level control mechanism, and a MAC sub-layer that provides access to the physical channel for all types of transfer. Fig. 2.2 shows these blocks in a graphical representation.

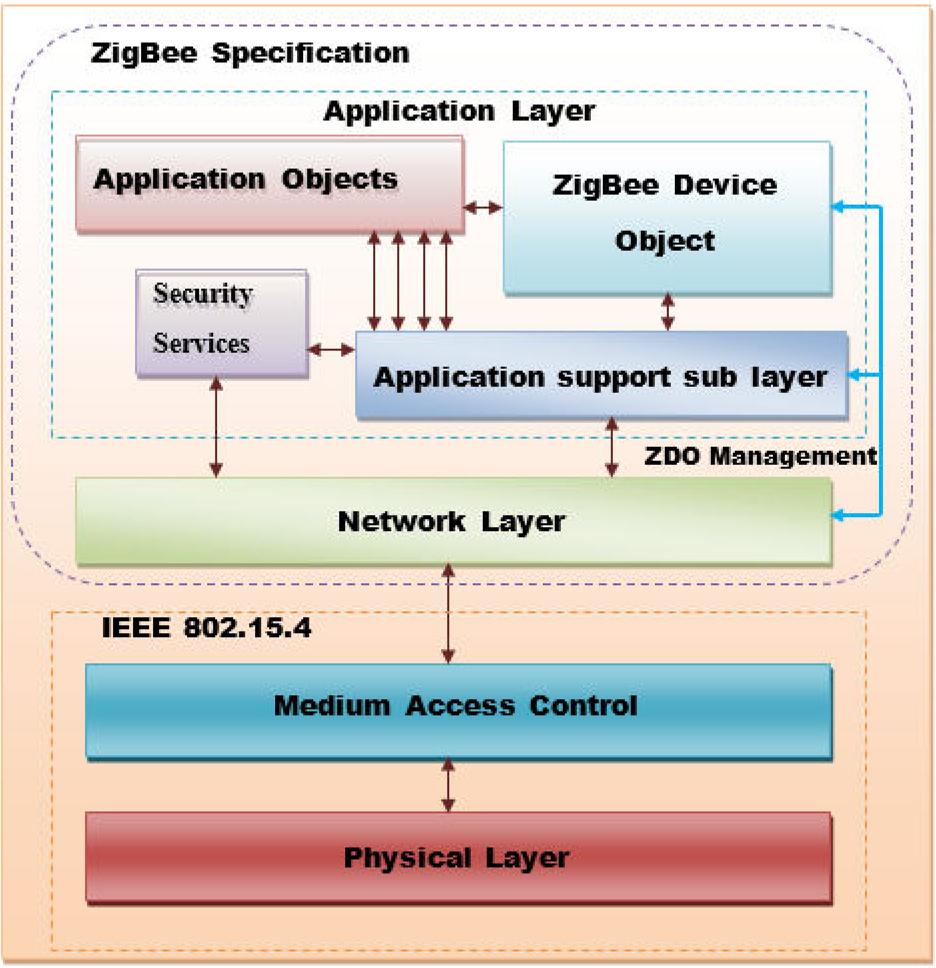


Fig. 2.2: ZigBee Protocol Stack

The ZigBee protocol architecture is divided into three sections, as follows:

- IEEE 802.15.4, which consists of the MAC and physical layers.
- ZigBee layers, which consist of the network layer, the ZigBee device objects (ZDO), the application sub-layer, and security management.
- Manufacturer application: Manufacturers of ZigBee devices can use the ZigBee application profile or develop their own application profile.

ISO-OSI network reference model for ZigBee [15]

Network devices, whether wired or wireless, are commonly described by the Open Systems Interconnection (OSI) reference model. This abstraction model was developed by the International Standards Organization (ISO), starting in the 1980 description of communication-related protocols and services. The generic seven layer model is applied to all network and media types. The adaptation ISO-OSI network reference model for ZigBee is illustrated in the Fig. 2.2. ZigBee network model does not use presentation, session or transport layer and user application is directly tied into Application layer (APL). This figure shows also IEEE, ZigBee Alliance, and ZigBee product end manufacturer particular responsibility for ZigBee certified product as well as hardware and software proportion in ZigBee.

- **IEEE 802.15.4 Standard [16]**

The IEEE standard brings the ability to identify uniquely every radio in a network as well as the method and format of communications between these radios, but does not specify beyond a peer-to-peer communications link, a network topology, routing schemes or network growth and repair mechanisms. The ZigBee Alliance selected the IEEE 802.15.4 standard, released in May 2003, as the wheels and chassis upon which ZigBee networking and applications have to be constructed. IEEE 802.15.4 defines three frequency bands to employ a standard over the world. Overview of available bands, modulation method and other properties of each is resumed in the table below.

Table 2.3 ZigBee parameters with respect to different operating frequency

Parameters	868 MHz	915 MHz	2.45 GHz
Frequency Band	ISM	ISM	ISM
Bit Rate	20 Kbps	40 Kbps	250 Kbps
Number of channels	1	10	16
Modulation	BPSK	BPSK	O-QPSK

- **General characteristics of IEEE 802.15.4 Standard**
    - Data rates of 250 kbps, 20 kbps and 40kpbs.
    - Star or Peer-to-Peer operation.
    - Support for low latency devices.
    - CSMA/CA channel access.
    - Dynamic device addressing.
    - Fully handshaked protocol for transfer reliability.
    - Low power consumption.
    - Extremely low duty-cycle (<0.1%)

2.5.3.2 IEEE 802.15.4 MAC layer

The Medium Access Control (MAC) layer is also known as the Data Link Layer. This layer is concerned with the addressing. It determines destination address for outgoing data and source address for incoming data. It is also responsible for assembling data packets or frames to be transmitted and decomposing received frames. The MAC sub layer provides two services: the MAC data service and the MAC management service interfacing to the MAC sub layer management entity (MLME) service access point (SAP) (MLMESAP). The MAC data service enables the transmission and reception of MAC protocol data units (MPDU) across the PHY data service.

The Network (NWK) layer which is right above MAC is defined by the ZigBee Alliance. It allows devices to communicate with each other. It is involved in the initialization of the device; network Self-organization and routing of data and network discovery within the network. The Application (APS) layer at the top of the stack contains application profiles, defined by ZigBee Alliance contains the applications that run on the network node. This layer determines device relationships and Supervises network initiation and association functions.

The serial communication monitoring software has front end which will simply give a Graphical User Interface (GUI) for monitoring raw value from the serial port. At the back end of the software two types of functionalities are there: one for receiving and another provision has been kept for sending for future reference.

2.5.3.3 IEEE 802.15.4 PHY layer

- Activation and deactivation of the radio transceiver.
- Energy detection within the current channel.
- Link quality indication for received packets.
- Clear channel assessment for CSMA/CA.
- Channel frequency selection.
- Data transmission and reception.

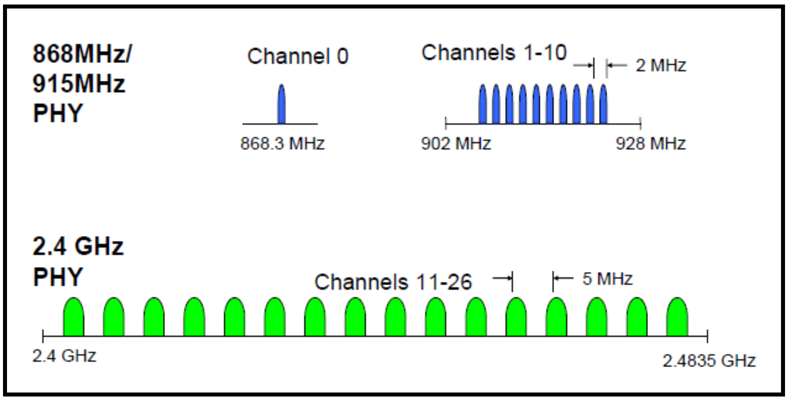


Fig. 2.3 Operating frequency bands of IEEE 802.15.4 [17]

PHY Frame Structure:

PHY packet fields

- Preamble (32 bits) – synchronization
- Start of packet delimiter (8 bits) – shall be formatted"11100101"
- PHY header (8 bits) –PSDU length
- PSDU (0 to 127 bytes) – data field

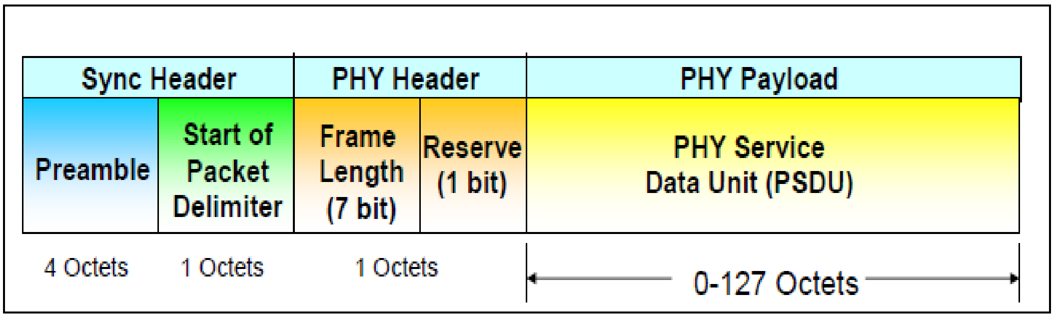


Fig. 2.4 Physical layer frame structure of IEEE 802.15.4

2.5.3.4 IEEE 802.15.4 Device Types

Two different device types are present:

- A full function device (FFD)
- A reduced function device (RFD)

- **Full-Function Device (FFD):** An IEEE 802.15.4 based wireless sensor network must include at least one Full-Function Device, operating as the PAN coordinator or Gateway of a personal area network. It plays and maintains a pivotal role for making the infrastructure of the network.

  FFDs can operate in three modes serving

  - Sensor node
  - Router
  - Gateway

- **Reduced-Function Devices (RFD):** On the other hand there is Reduced-Function Devices (RFD) which are having extremely simpler architecture than Full-Function Devices. RFDs are with very modest communication and resource requirements and due to this characteristic, they can only communicate with FFD's. A RFD can never play a role of a coordinator. It is proposed for applications that are extremely simple in nature and do not need to send large volume of data like audio or video.

RFDs can only operate in a mode serving for End Device.

2.6 Network Topology

2.6.1 Star Topology

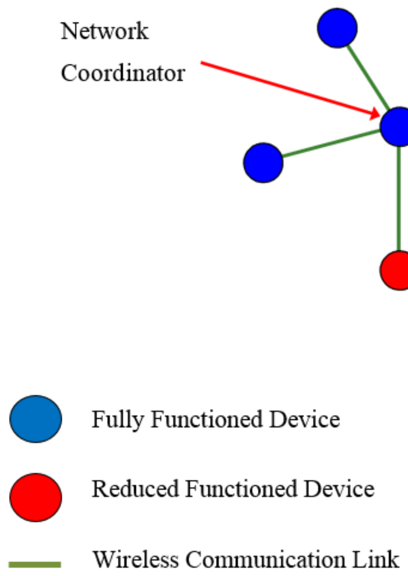


Fig. 2.5: Structure of Star Topology.

Star topology is one of the most common network setups. In this configuration, every node connects to a central network device, like a hub, switch, or computer. The central network device acts as a server and the peripheral devices act as clients.

2.6.2 Mesh Topology

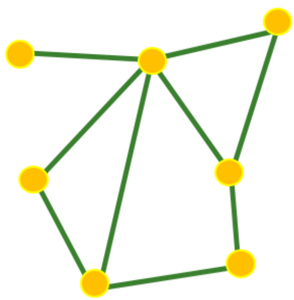


Fig. 2.6: Structure of Mesh Topology.

A mesh network employs one of two decentralized connection arrangements. Mesh Network are of two types:

- Full mesh topology - In a full mesh topology, each network node (workstation or other device) is connected directly to each of the others.

  A fully connected mesh can have

  - n.(n-1)/2 physical channel to link n devices
  - It must have n-1 I/O ports

- Partial mesh topology

In a partial mesh topology, some nodes are connected to all the others, but others are only connected to those nodes with which they exchange the most data.

2.6.3 Tree Topology

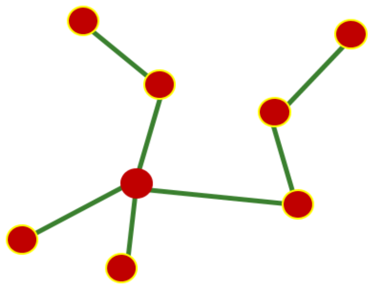


Fig. 2.7 Structure of Tree Topology

Tree Topology integrates the characteristics of Star and Bus Topology. In Tree Topology, number of Star networks is connected using Bus. The main cable seems like a main stem of a tree, and other star networks as the branches. It is also called Expanded Star Topology. Ethernet protocol is commonly used in this type of topology.

The topology and the network structure of WSN is not a strict standard and can be varied and designed as per the requirements.

2.7 Communication Protocols, Model and Routing

2.7.1 Multiple Access Protocols:

When multiple nodes desire to transmit, protocols are needed to avoid collisions and lost data.

2.7.1.1 CSMA/CA Wireless Access Technology

CSMA/CA is commonly used in computer networking. In CSMA, a station (node) wishing to transmit has to first listen to the channel for a predetermined amount of time so as to check for any activity on the channel. If the channel is sensed "idle" then the station is permitted to transmit. If the channel is sensed as "busy" the station has to defer its transmission. This is the essence of both CSMA/CA, and CSMA/CD where the abbreviation CD stands for Collision Detection. In CSMA/CA (LocalTalk), once the channel is clear, a station sends a signal telling all other stations not to transmit, and then sends its packet. In Ethernet 802.11, the station continues to wait for a time, and checks to see if the channel is still free. If it is free, the station transmits, and waits for an acknowledgement signal that the packet was received [18].

2.7.2 Routing

Since a distributed network has multiple nodes and services many messages, and each node is a shared resource, many decisions must be made. There may be multiple paths from the source to the destination. Therefore, message routing is an important topic. The main performance measures affected by the routing scheme are throughput (quantity of service) and average packet delay (quality of service). Routing methods can be fixed (i.e. pre-planned), adaptive, centralized, distributed, broadcast, etc [19].

a) **Fixed routing schemes** often use Routing Tables that dictate the next node to be routed to, given the current message location and the destination node. Routing tables can be very large for large networks, and cannot take into account real-time effects such as failed links, nodes with backed up queues, or congested links.

b) **Adaptive routing schemes** depend on the current network status and can take into account various performance measures, including cost of transmission over a given link, congestion of a given link, reliability of a path, and time of transmission. They can also account for link or node failures [20].

2.8 Choice of Wireless Technology

2.8.1 Selection of protocol for monitoring

In case of wireless communication for transmission medium, the transmission of the data can be achieved using radio frequencies, optical communication, and ultra sound; other media like magnetic inductance are only used in very special cases. Of these choices, optical communication requires Line of Sight (LOS) path between the transmitter and receiver and hence is not used frequently. Instead, RF is used as a wireless link in which the waves can penetrate a limited number of walls. Hence, Radio Frequency (RF)-based communication is by far the most relevant one as it best fits the requirements of most wireless network applications. It also provides relatively long range and high data rates, acceptable error rates at reasonable energy expenditure, and does not require line of sight between sender and receiver [21].

A protocol is a set of rules or agreed upon guidelines for communication. While communicating it is important to agree on how to do so. If one party speaks English and one Bengali the communications will most likely fail. Various communication protocols are used in respect to achieve reliability, integrity, availability, and security of the data. The recent technologies involved several wireless protocols available in Sensor-based Wireless Networks. Many wireless sensor network nodes or motes use RF transceiver operating frequency at 916 MHz, while others use a 2.4 GHz IEEE 802.15.1 (Bluetooth), or 2.4 GHz IEEE 802.11a, or 2.4 GHz IEEE 802.11b, or other bands defined by IEEE 802.15.4, IEEE 802.16 etc. According to application scenarios, we need to carefully select the communication media. In the following section we will analyze which protocol would be suitable in our proposed application.

The plethora of available wireless technologies makes even the selection of components of wireless sensor network difficult. The study of wireless sensor network is challenging in that it requires an enormous breadth of knowledge from an enormous variety of disciplines. The first decision was to taken in this project, is to select the wireless platform. Here we shall initially explore various popular wireless protocols available and then go for our choice according to our application scenario [22].

Table 2.4 Comparative Study of different wireless protocol

Standard	Bluetooth	ZigBee	Wi-Fi
IEEE Specification	802.15.1	802.15.4	802.11a/b/g
Frequency band	2.4 GHz	868/915MHz; 2.4GHz	2.4 GHz; 5 GHz
Max Signal Rate	1 Mbps	250Kbps	54 Mbps
Nominal Range	10 m	10-100m	100m
Nominal Tx power	0 - 10 dBm	(-25) - 0 dBm	15 - 20 dBm
Modulation type	GFSK	BPSK (+ ASK), O-QPSK	BPSK, QPSK COFDM, CCK, M-QAM
Spreading	FHSS	DSSS	DSSS, CCK, OFDM
Data protection	16-bit CRC	16-bit CRC	32-bit CRC

Since there is a trade-off between transmission data rate and power consumption it is required to choose the best suitable wireless technology for WSN.

For WSN nominal data rate i.e. in the order of 40-250 Kbps with low power consumption to increase the life-time of WSN is highly preferable. Therefore it is concluded that ZigBee is the best solution among other existing wireless technology as shown in Table 2.4

2.9 Challenges in Wireless Sensor Network

For WSNs to become truly efficient, a number of challenges and hurdles must be overcome. Challenges and limitations of wireless sensor networks are the following:

- **Lifetime of the Network:** Energy is a constraint in sensor nodes. Replacing these energy sources in the field is usually not practicable, and simultaneously a WSN must operate at least for a given mission time or as long as possible.

- **Scalability:** A WSN generally consist of a large number of nodes, the employed architectures and protocols must be able scale to these numbers.
- **Programmability:** The sensor nodes do not have to process information only , but they also have to react flexibly on changes in their tasks hence these nodes should be programmable, and their programming must be changeable during operation when new tasks become important. A constant way of information processing is insufficient.
- **Maintainability:** It is the ease with which changes can be made to a system. These changes may be necessary for the correction of faults, adaptation of the system to meet new requirements like addition of sensors or tags, addition of new functionality, removal of existing functionality or corrected when errors or deficiencies occur and can be perfected, adapted or actions taken to reduce further maintenance costs. Inspections, reviews and measurement of effort required to execute maintenance activities should be carried out to test the maintainability of the whole system before releasing the product in the market.
- **Fault tolerance:** The sensor nodes are susceptible to many problems like the energy crisis, environmental challenges, communication problems, physical damage. The failure of one of the nodes should in no way affect the working of the entire network.
- **Cost factor:** Now since the network can contain such a huge number of nodes, manufacturing cost of each of the nodes should be very less to make the network as a whole cost effective.
- **Smaller size:** A sensor node contains a sensing unit, a processing unit, a transceiver unit and a power unit, the overall size of the sensor nodes is very small, so for the node to comprise all these units, the units should be very tiny in size. Apart from the size

constraint, there are some other constraints like the node must be power saving and should be able to work unattended.

- **Power consumption:** To increase the life time of the sensor networks the power consumption should be very efficient. The nodes can be made to even harvest some energy by means like solar, thermal or vibration (electromagnetic as well as electrostatic). The required power increases if the square of the distance between the source and the destination increases, so multiple short range transmissions will require less energy.
- **Topology:** The sensor nodes can be deployed by either simply throwing them in a mass in the area or by placing each of the nodes individually , once deployed the nodes must be able to bring about changes in the topology to face the problems like communication problem .
- **Environmental factors:** The sensor nodes are deployed in any area very densely they can be used in various environments, like inside machinery, at the bottom of ocean, to monitor some patients, war area, and disaster area, to monitor weather changes.
- **Transmission channel factors:** The wireless media that can be used are radio, infrared, or optical. In order to make it globally operable, the medium must be available worldwide.
- **Limited functional capabilities:** The nodes should not only serve the purpose of routing the data, they must also have some processing capabilities.

2.10 Survey of sensor nodes available in the market

A survey of other sensor nodes was completed in order to determine if a suitable replacement existed that was targeted to meet the primary goals described above. Radio range is required to be tested as it was more of a practical characteristic. An external interface would also be required as this would allow the addition of extra components, mainly sensors, which was essential for both research and practical customization. There are many readily available sensor nodes on the market but that doesn't meet the requirement of low cost and all the functionality required for precision agriculture monitoring and controlling application. The radio transceivers on new nodes seem to be targeted towards the ZigBee specification as this allows a simpler development cycle.

The following sections discuss some commercially available sensor nodes that meet the part of the requirements but not all under a single platform. Since this research work dealt with development of low cost product it will be lucrative to design new node according to the requirements.

Each node mentioned below has an argument presenting why it was not chosen and how it did not meet the other deciding goals.

- **Redwire Econotag**

  The Redwire Econotag [23] is a development board based on a Freescale MC13224V ARM7 microcontroller with a built-in IEEE 802.15.42.4GHz wireless transceiver but it is not ZigBee compliant. It features 128kB of flash storage and 96kB of RAM. No sensors are included on the board but 36 GPIO pins are brought to headers for expansion. This board is designed as a development board and not suited for practical deployment because there is little allowance for practical work.

- **Virtenio Preon32**

  The Virtenio Preon32 [24] sensor node is designed around an unnamed ARM Cortex-M3 microcontroller with 256kB of flash and 64kB of RAM. The radio is an IEEE 802.15.42.4GHz transceiver. It is designed as module to be soldered on another PCB which provides the interfaces and power. This approach means that the cost to buy the module as well as designing and manufacturing a separate board would have been much more than the cost of producing a single board for a similar amount of work.

- **Arago Systems WiSMote**

  The Arago Systems WiSMote [25] is a development platform designed on a TI MSP430 microcontroller and has 16kB of RAM and 256kB of flash. The transceiver is a TI CC2520 2.4GHz radio that can communicate over 100m. The board is designed for research and development. As a consequence, it is less suitable for practical deployment. In particular the communication range is less than ideal.

- **Libelium Waspmote**

  The Libelium Waspmote [26] is a feature-packed sensor board designed around an Atmel ATmegal281 microcontroller with 8kB RAM and 128kB flash storage. It uses a modular approach to every peripheral and is designed to use XBee radio modules of various frequencies that support ZigBee. A GPRS socket, GPS socket and expansion socket provide the use of add-on circuit boards manufactured by Libelium. This board is unsuitable for our use due to its very high price, over $200 and the unnecessary modularity.

- **SENSEnuts**

  With a 32 bit RISC processor at its core, the Radio Module is the heart of SENSEnuts platform. The modular design facilitates the interface of a wide variety of sensors with the Radio module and harness its processing power. But the cost of each node is around Rs 10000/- which is much higher and beyond the reach of farmers to purchase it.

Chapter 3

Proposed Wireless Sensor Network Model

3.1 Components of wireless sensor network

The components of a wireless sensor network facilitate wireless connectivity within the network. A wireless sensor network greatly consists of Gateway, Router and End Device. A transparent data path is established between the application platform and the physical world.

Wireless sensor networks are used to exchange information between an application platform and one or more sensor nodes according to requirement of application. This exchange takes place in a wireless fashion.

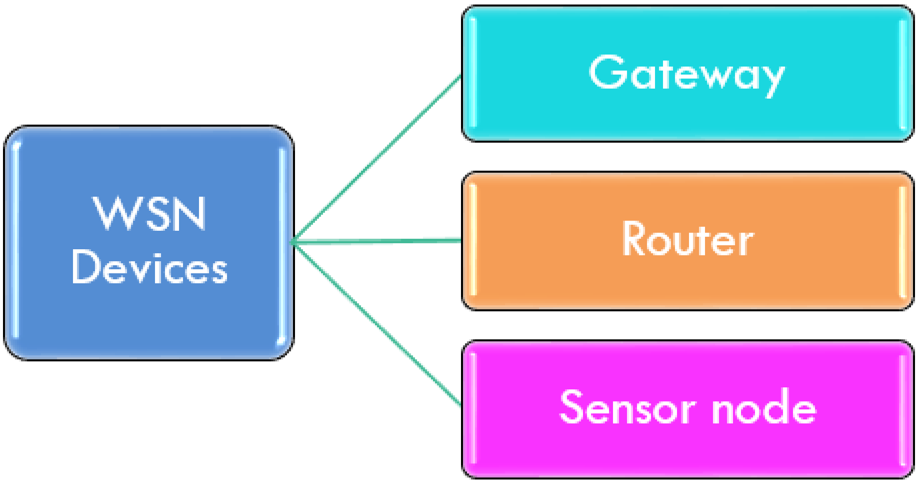


Fig. 3.1: Components of WSN

3.2 Functionality of Different Components of WSN

3.2.1 Gateway

A gateway is an interface between the application platform and the wireless nodes on the wireless sensor network. All information received from the wireless nodes is aggregated or manipulated by the gateway and forwarded to the application. That application may run on a local computer or a networked computer. In the reverse direction, when a command is issued by the application program to a wireless node, the gateway relays the information to the wireless sensor network.

All gateways can perform protocol conversion to enable the wireless network to work with other industry or non-standard network protocols.

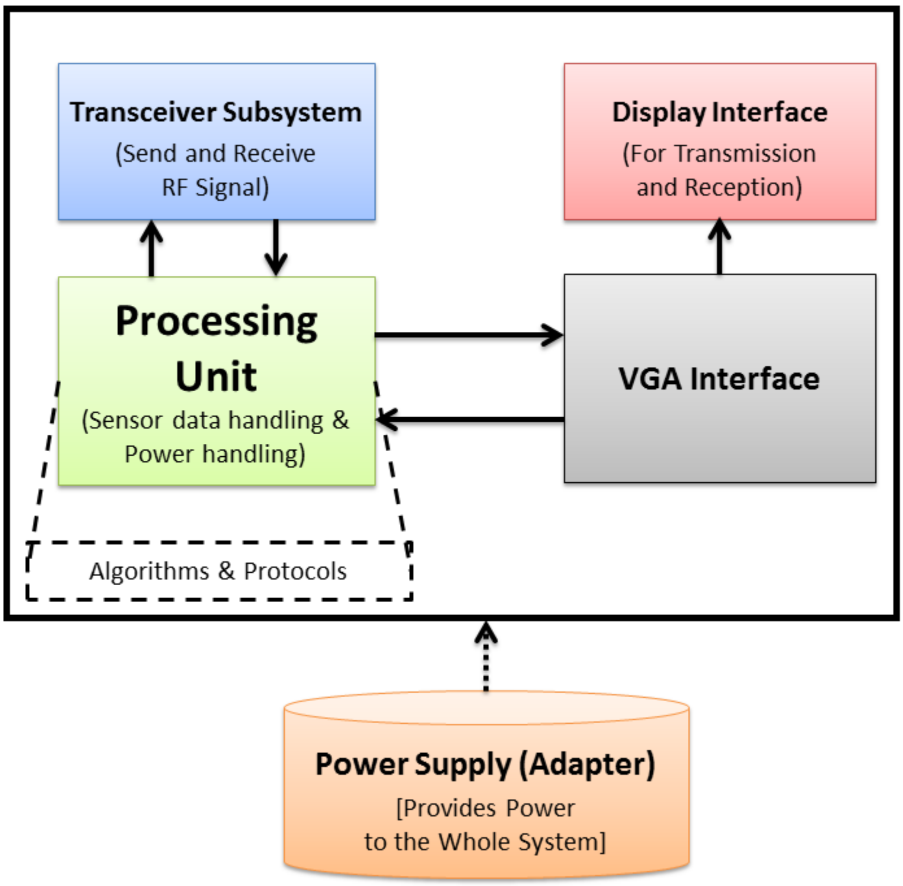


Fig. 3.2: Block Diagram of Gateway

3.2.1.1 Hardware Description of Gateway:

Transceiver Subsystem: The radio frequency (RF) communication subsystem is an essential part for Gateway, Router and End device. It is used to send and receive data packet between individual nodes whose contents can vary from sensor information.

For Real-Time communication, both a transmitter and a receiver are required in a sensor node. In the transmitter side, the major task is to receive a bit stream coming from a microcontroller and then convert it to radio waves for transmission. While in the receiver side just the opposite action takes place. For practical purposes, it is usually convenient to use a device that combines these two tasks in a single entity. Usually half-duplex operation is realized since transmitting and receiving at the same time on a wireless medium is impractical in most cases.

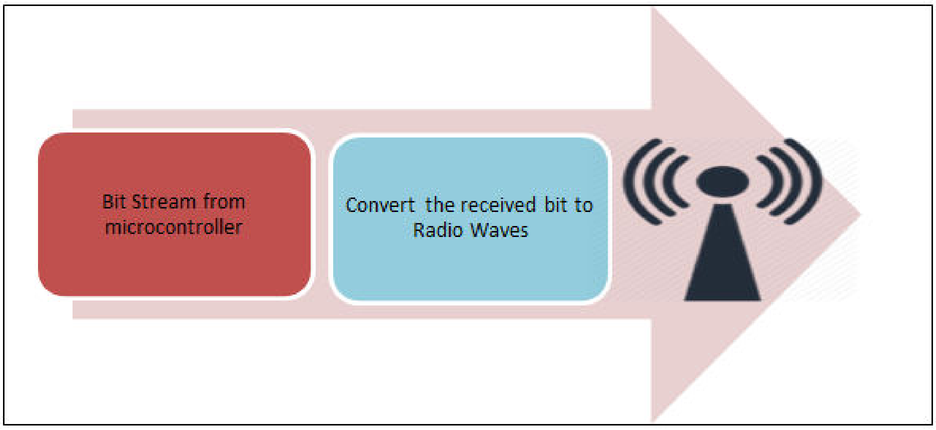


Fig. 3.3: Transmission of digital data into radio waves

3.2.1.2 Selection of Transceiver Subsystem:

The RF module selected for transceiver system among the various other modules available in the market is XBee Series 1 from Digi International. A comparative study has been given below.

Table 3.1 Comparison of RF modules with different parameters

Name of the module	ETRX357 ZigBee Modules	ATMEGA 128RF230	NXP JN5148	CC2630	RF6525	XBee Series 1
Vendor	TELEGESIS	ATMEL	NXP	TEXAS Instruments	RFMD	Digi International
Transmit Power Output	3 dBm	3dBm	2.5dBm	5dbm	22dBm	22dBm
RF Data Rate	250 kbps	250kbps	500- 667kbps	250 Kbps	250 Kbps	250 Kbps
Receiver Sensitivity	-100dBm	-101dBm	-95dBm	-100 dbm	-95dBm	-92dbm
Supply Voltage	2.1 to 3.6V	1.8 - 3.6	2.0V to 3.6V	2.0V to 3.6V	3.6 V	3.3 V
Transmit Current	31	16.5	15mA	9.1mA	200 mA	50 mA
Frequency	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz
Dimensions	$25*19mm^2$	$5*5\ mm^2$	$8*8mm^2$	$15*21*8mm^3$	$3.5*3.5*0.5mm^3$	0.0960″ * 1.087″
Operating Temperature	-40 to +85 C	-40 to +85 C	-40°C to +85°C	-40°C to +85°C	-40 to +85 C	-40 to 85 C
Cost (Rs)	1265	339	1200	412	384	1100

According to the survey of the RF module mentioned in the above Table 3.1 XBee Series 1 is compatible with RichDuino Board and cost is comparatively less considering other RF modules available in the market. We require a low data rate module for communication for this purpose we have chosen Xbee series 1. Communication range is another important issue to be considered.

The price of RF module is directly proportional to its communication range.

Communication Range ↑ - Price of RF module ↑

For experimental purpose we have used a moderate communication range module.

If there is further requirement to increase the communication range then the RF module can be replaced by XBee Pro series.

XBee Shield:

XBee Shield is mounted over RichDuino Board to interface the RF module with the RichDuino Board.

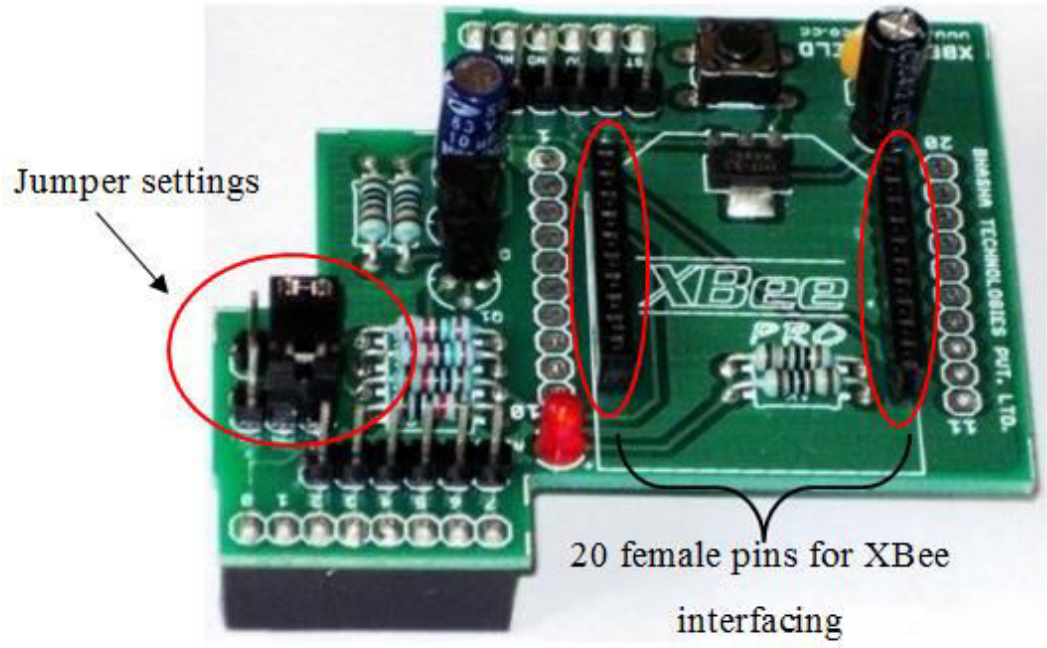


Fig. 3.4 XBee Shield [28]

- It has female connectors soldered at the bottom
- Any XBee module will work with the shield
- Works with XBEE and XBEE Pro modules.
- The module can communicate up to 100m (~300') unobstructed range [28].

The jumper settings are very important for working of XBee either in command mode or transmit mode.

Command Mode – In command mode different parameters of XBee is configured with the help XCTU software developed by Digi International. In this mode TX pin of XBee is connected with TX of RichDuino Board and similarly RX of XBee is connected to RX pin. The parameters which can be configured are as follows:

- Personal Area Network (PAN) ID
- Channel

- Source and Destination address
- Update Firmware
- AT/API mode

Transmit Mode – This mode is operated after proper configuration of XBee module in command mode. In this mode TX pin of XBee module is connected with RX pin of XBee and vice-versa. Transmit mode is further classified in AT and API mode which will be discussed later in brief.

3.2.1.3 Selection of Processing Unit: -

The processing unit selected in designing of Gateway of WSN is Raspberry Pi. Since our goal is to design a low cost WSN model here we replacing a Personal computer required for display and act as a server by Raspberry Pi. The cost of a Personal Computer is around Rs 30000/- (with display unit). Whereas the cost of Raspberry Pi is approximately Rs 3000/- + cost of display unit i.e. Rs 4000/-.

Hence the total cost is reduced by Rs (30000 – 3000 – 4000) = Rs 23000/-

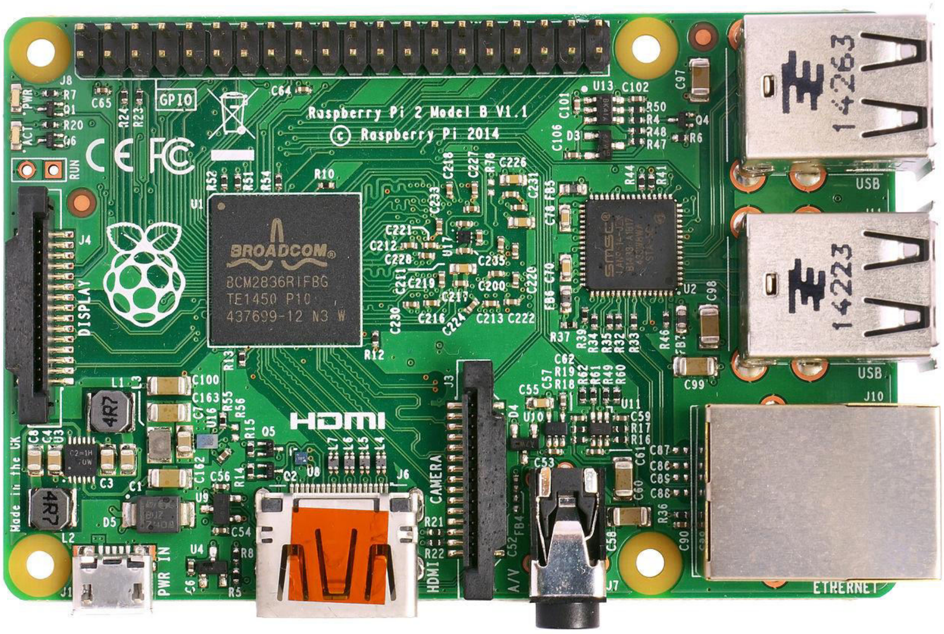


Fig. 3.5: Raspberry Pi 2 Model B [29]

A brief overview of Raspberry Pi 2 Model B

The Raspberry Pi 2 Model B is the second generation Raspberry Pi. It replaced the original Raspberry Pi 1 Model B+ in February 2015.

It has following salient features:

- A 900MHz quad-core ARM Cortex-A7 CPU
- 1GB RAM
- 4 USB ports
- 40 GPIO pins
- Full HDMI port
- Ethernet port
- Combined 3.5mm audio jack and composite video
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot
- Video Core IV 3D graphics core

Because it has an ARMv7 processor, it can run the full range of ARM GNU/Linux distributions, including Snappy Ubuntu Core, as well as Microsoft Windows 10.

The Raspberry Pi 2 has an identical form factor to the previous (Pi 1) Model B+ and has complete compatibility with Raspberry Pi 1 [29].

Cost of single Unit = Rs 3000/-

3.2.1.4 Selection of Display Interface

The display unit is chosen as VGA monitor rather than HDMI monitor to minimize the cost around Rs 5000-6000. The HDMI port is connected with HDMI to VGA converter to interface with VGA monitor.

Cost of Single Unit = Rs 4000/-

3.2.1.5 Selection of VGA Interface

The HDMI to VGA cable is connected to the HDMI port of Raspberry Pi to display the collected data in the monitor and analyze it.

Cost of single unit = Rs 150/-

Fig. 3.6: HDMI to VGA converter [30]

3.2.1.6 Selection of Power Supply

The power adapter of 9-12 V is connected with Raspberry Pi since DC voltage is required to operate it. The adapter is liable for converting alternating current provided by the wall socket to DC current in order to facilitate functioning of Raspberry Pi. Some idling power is wasted as the power supply is left running when the device power is switched off or the device is disconnected from the power supply. Specifications 9-12V, 1 AMP.

Cost of single Unit = Rs 150/-

Fig. 3.7: Power Adapter [31]

3.2.1.7 Integration of Selected Components for Designing of Gateway

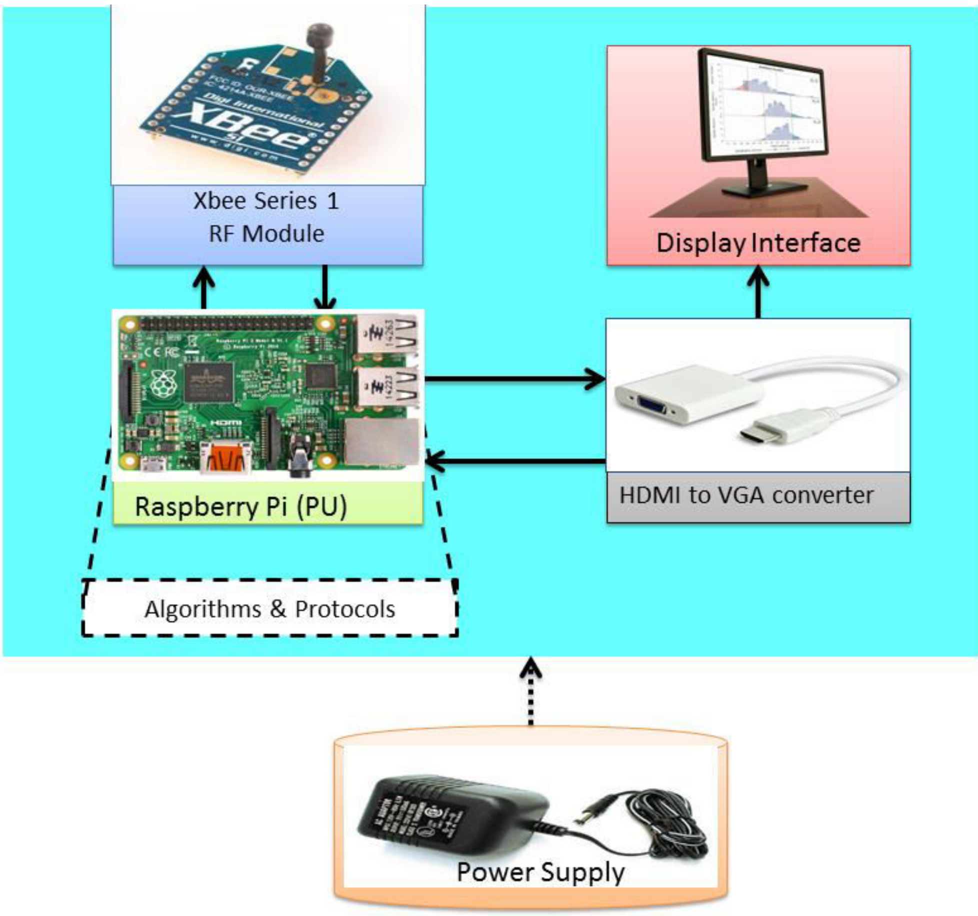


Fig. 3.8: Overview of selected components

3.2.1.8 Cost Analysis of Gateway Unit

Table 3.2 Calculation of total cost of single gateway unit

Item name	Unit Price (Rs)	No. of Unit	Total (Rs)
Raspberry Pi	3000	1	3000
Xbee Series 1	1100	1	1100
XBee Shield	400	1	400
Display Unit	4000	1	4000
Power Supply	150	1	150
HDMI to VGA converter	150	1	150
Total Price			**8800**

3.2.2 Functionality of Router

Each router is considered a full-function device (FFD). They are used to extend network coverage area, route around obstacles and provide back-up routes in case of network congestion or device failure. It also helps in minimizing the transmission power of each sensor node which leads to increase in long battery life.

3.2.2.1 Block Diagram of Router

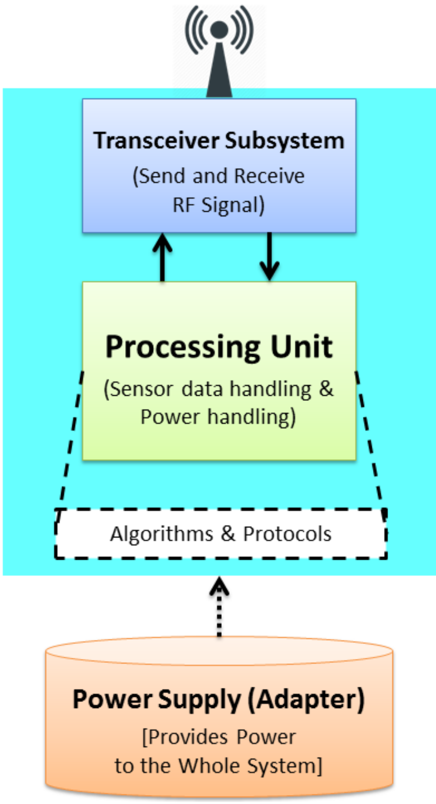


Fig. 3.9: Block Diagram of Router

3.2.2.2 Hardware Description of Router

3.2.2.3 Selection of Transceiver Subsystem

The transceiver subsystem chosen for router is same as a Gateway i.e. XBee Series 1 module. The packet format for sending and receiving data is same for all the devices in WSN. It is easier for the user to decode the transmitted data when it is sent from same RF module.

3.2.2.4 Selection of Processing Unit

The processing unit chosen for router is RichDuino board. This board is selected in place of ArduinoUno board to minimize the cost by 40%.

Richduino is a better option for researchers to offer high quality affordable boards for Arduino IDE. Low price and highest degree of production quality are two main reasons selecting choosing Richduino instead of Arduino Uno Board [32]. Richduino is inspired from open source Arduino platform specifications.

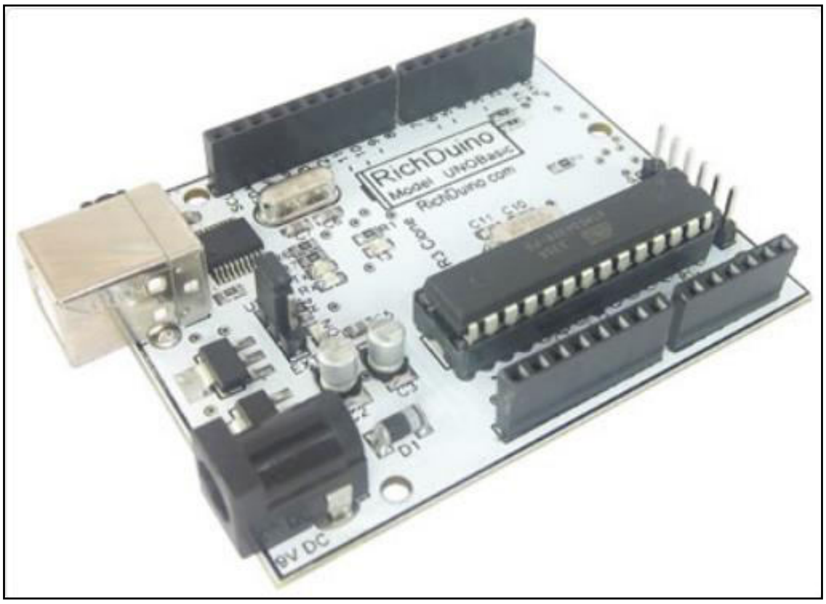


Fig 3.10: Top View of RichDuino board [32]

What is Arduino?

It is open source hardware, it shares much of the principles and approach of free and it is to understand the hardware; how it works, make changes to it, and share those changes.

Why Arduino is chosen as working platform?

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality but the reason behind choosing Arduino platform is because of its user friendly software and open source hardware with an affordable price. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

- **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less around Rs 1200/-
- **Cross-platform** - The Arduino software runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- **Simple, clear programming environment** - The Arduino programming environment is easy-to-use for beginners, yet flexible enough for advanced users to take advantage as well.
- **Open source and extensible software-** The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries.
- **Open source and extensible hardware** - The Arduino is based on Atmel's ATMEGA328P microcontroller. The plans for the modules are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money. This same concept is used to make Richduino Board which works same as Arduino but with low cost.

  Though it is easy to use, RichDuino’s underlying hardware works at the same level of sophistication that engineers employ to build embedded devices. People already working with microcontrollers are also attracted to this because of its agile development capabilities and its facility for quick implementation of ideas.

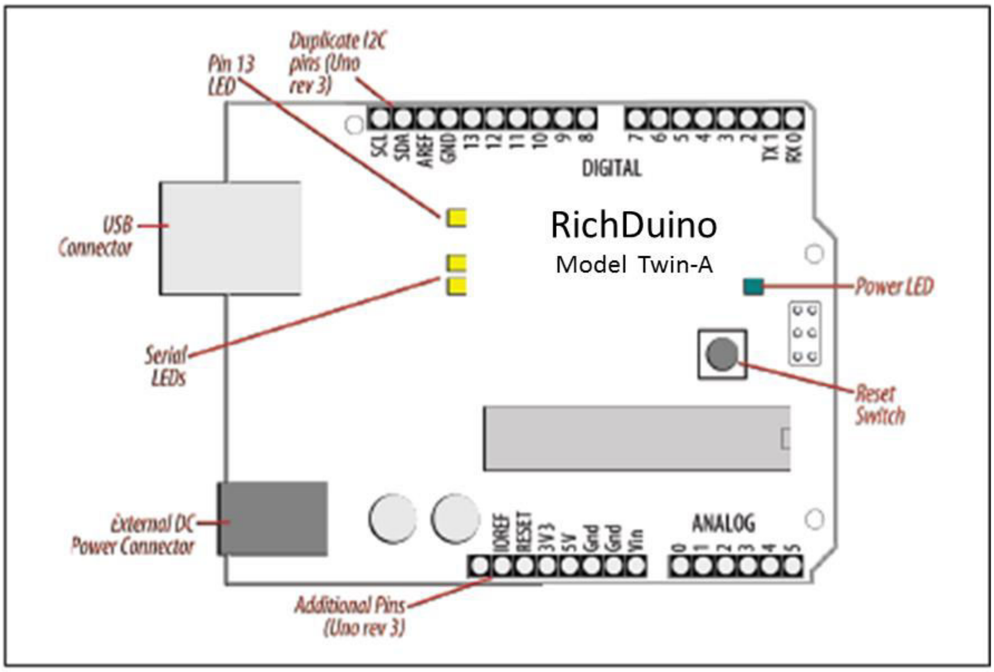


Fig. 3.11: Pin out of RichDuino Board [32]

Features of RichDuino Board

Technical specifications

- Microcontroller : Atmega328
- Operating Voltage : 5V DC
- Input Voltage (via DC Adapter) : 9V DC
- Alternate input voltage source : USB connection
- Digital I/O pins : 14 (of which 6 can be used as PWM output)
- Analog input pins : 6 (with modification in code, these pins act as normal I/O pins)
- DC Current Limit for I/O Pins : 40mA
- DC Current Limit for 3.3V Pin : 150mA (if input voltage is sourced from USB)
- Switching between power sources : Manual Jumper Pin
- Program Memory : 32KB On-chip memory of ATmega328 (of which 0.5KB is used by boot loader)
- SRAM : 2KB On-chip memory of ATmega328
- EEPROM : 1KB On-chip memory of ATmega328

- Clock Speed : 16Mhz
- Board dimensions: 69mm X 55mm (Board is rectangular)
- Cost of single unit of RichDuino board = Rs 750/-

3.2.2.5 Selection of Power Supply

The power supply used for router is same as Raspberry Pi in Gateway i.e. 9-12 V adapter. The power supply can also be provided through 6 AA batteries. The number of AA batteries is 6 because each AA supply a voltage of 1.5 V and RichDuino board requires 9V. The router has to fully functioned device since it will transfer data continuously from Sensor node to the Gateway which needs the device in power on mode for more span of time than End Device. Therefore it is concluded that it will be better to opt for continuous power supply from adapter rather than batteries.

3.2.2.6 Integration of Selected Components for Designing of Router

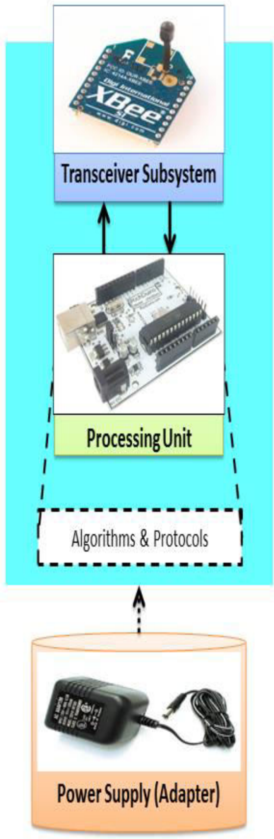


Fig. 3.12: Overview of selected component for Router

3.2.2.7 Cost Analysis of Router Unit

Table 3.3 Calculation of total cost of single router unit

Item name	Unit Price (Rs)	No. of Unit	Total (Rs)
RichDuino Board	750	1	750
XBee Series 1	1100	1	1100
XBee Shield	400	1	400
Power adapter	150	1	150
	Total Price		**2400**

3.2.3 Functionality of Sensor Node

Sensor nodes comprises of two main functions. The primary function is get data from sensor node through I/O ports and secondary is to send data to the router.

Sensor node is considered as a reduced-function device (RFD). Sensor node need not to be active all the time. Whenever it is extracting data from sensor and sending it to the router the microcontroller inside the sensor node is in active state otherwise in idle state. The power consumption of sensor node is much lesser than router and gateway hence power to the sensor node can be supplied through battery. Sensor node are interfaced with both the sensor and actuators required for a particular application.

3.2.3.1 Block Diagram of Sensor Node

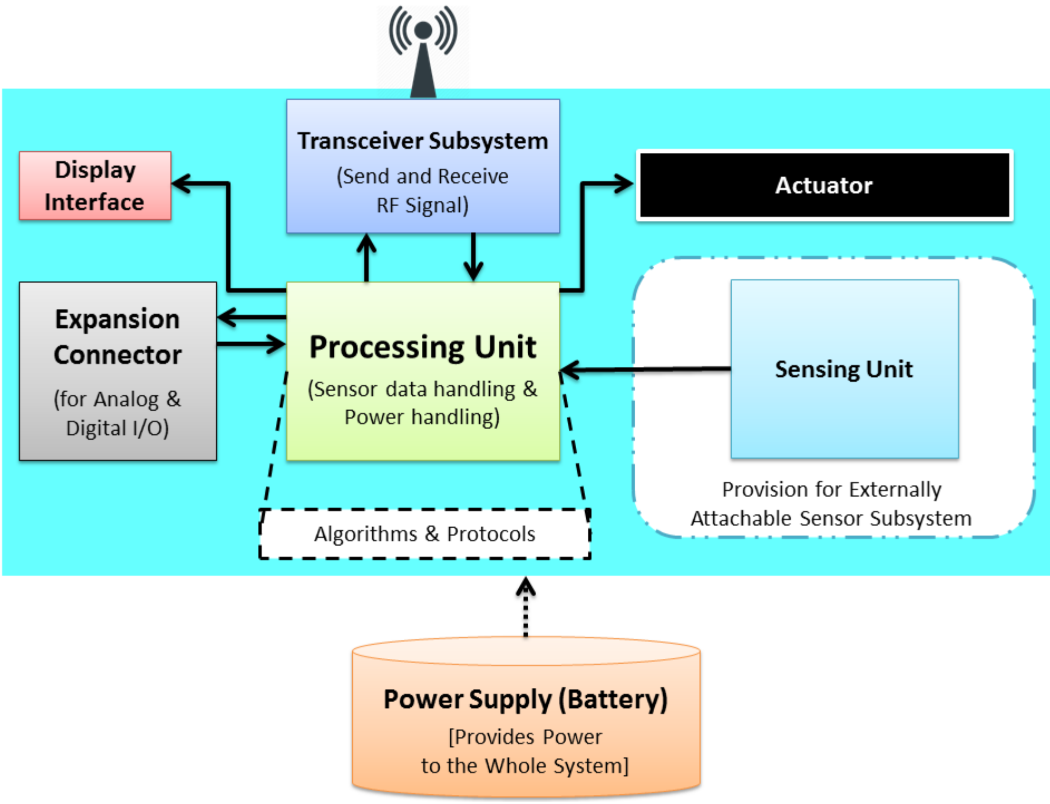


Fig. 3.13: Block Diagram of End Device

3.2.3.2 Hardware Description of Sensor Node

Transceiver Subsystem: -

The transceiver subsystem is same for all the 3 devices in WSN so that main wireless communication among the devices can occur smoothly.

Processing Unit: -

The processing unit chosen for End Device is same as Router i.e. Richduino board. The heart of this RichDuino Board is microcontroller ATMEGA328P which controls the entire functioning of the End Device through Firmware Programming.

3.2.3.3 Selection of Sensing Unit:-

a) Temperature Sensor: The temperature sensor used here is LM35 from Texas Instruments. The reason behind selecting this device is that it is cheapest among other device available in the market with all required inbuilt features. LM35 is an integrated-circuit temperature devices with an output voltage linearly-proportional to the Centigrade temperature. The LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient [6].

Features

- Calibrated Directly in Celsius (Centigrade)
- Linear + 10-mV/°C Scale Factor
- 0.5°C Ensured Accuracy (at 25°C)
- Rated for Full −55°C to 150°C Range
- Suitable for Remote Applications

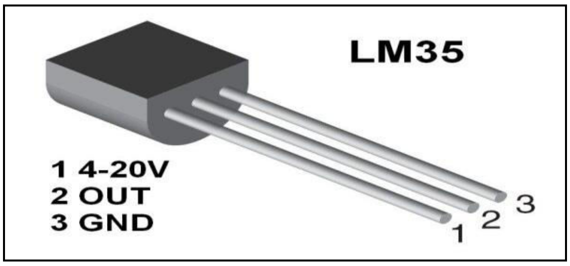


Fig. 3.14: Pin out of LM35 [33]

b) Soil Moisture Sensor: As our project is based on water management in agriculture field to minimize wastage of water and earn lofty yield at the end of the season, the sensing unit deployed here is soil moisture sensor, temperature and humidity. The sensor module has been interfaced with analog pin of RichDuino Board. Sensor module works as transducer which converts a physical quantity into electrical signal i.e. in the form of volts.

The signal received in the analog pin is later converted in digital. The Richduino Board has inbuilt analog to digital converter (ADC). The resolution of ADC is 10 bit. After proper calibration of sensor value it is transmitted and shown in the display unit.

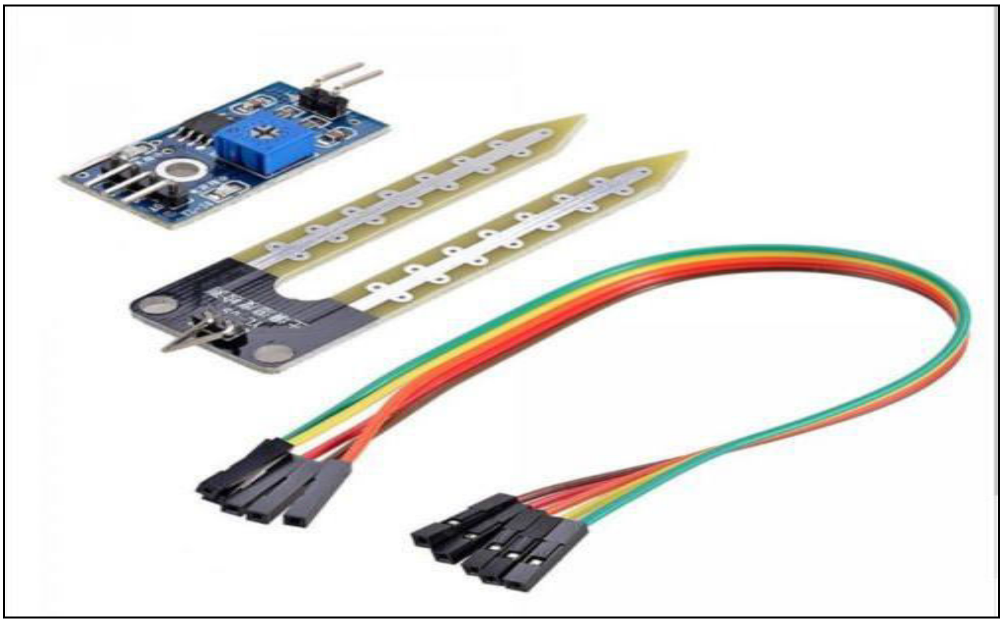


Fig. 3.15 Soil moisture sensor [34]

The expansion connector is used to make link between the analog pin of RichDuino Board and the sensor module. The sensor value is collected after certain interval of time. The interval of time depends upon the application and battery life.

Number of samples per hour ↑ - Battery life ↓

Hence there is a trade-off between number of samples and battery life.

Cost of single unit = Rs 110/-

3.2.3.4 Selection of Display Interface

The display unit incorporated in End Device is Liquid Crystal Display (LCD) 16x2. In End Device only the sensor value which is going to be transmitted to the Gateway through Router will be display.

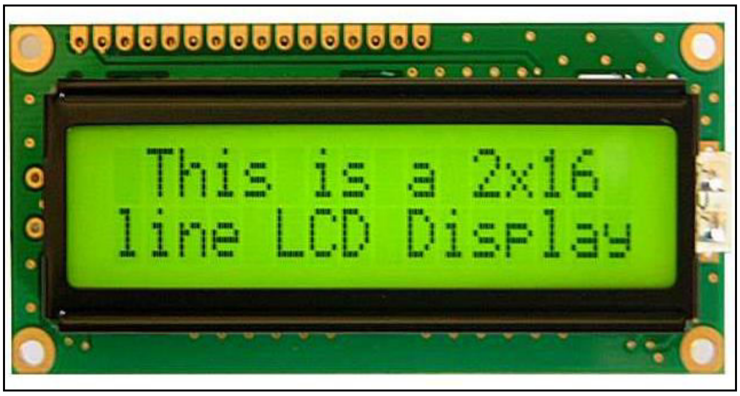


Fig. 3.16: Liquid Crystal Display [35]

LCD (Liquid Crystal Display) is used in all the electronics projects to display the status of the process. A 16x2 alphanumeric LCD is most widely used module of LCD nowadays. There are several others type of LCD available in market also.

The reason for choosing LCD over other display component or devices is that it is

- Low cost
- Easily programmable
- Large number of display character etc.

16x2 LCD has 2 horizontal line which comprising a space of 16 displaying character. It has two type of register inbuilt that is

- Command Register (CR) - The CR stores instruction codes, such as display clear and cursor shift, and address information for display data RAM (DDRAM) and character generator RAM (CGRAM).
- Data Register (DR) - the DR temporarily stores data to be written into DDRAM or CGRAM and temporarily stores data to be read from DDRAM or CGRAM. The DR is also used for data storage when reading data from DDRAM or CGRAM.

The choice between the two registers is made by the register selector (RS) signal as detailed the following table:

Table 3.4 Configuration of RS and R/W pin for Register Selection in LCD

Register Selector		
RS	R/W	
0	0	Sends a command to LCD
0	1	Read busy flag (DB7) and address counter (DB0 to DB6)
1	0	Sends information to LCD
1	1	Reads information from LCD

Pin Diagram of 16x2 LCD:

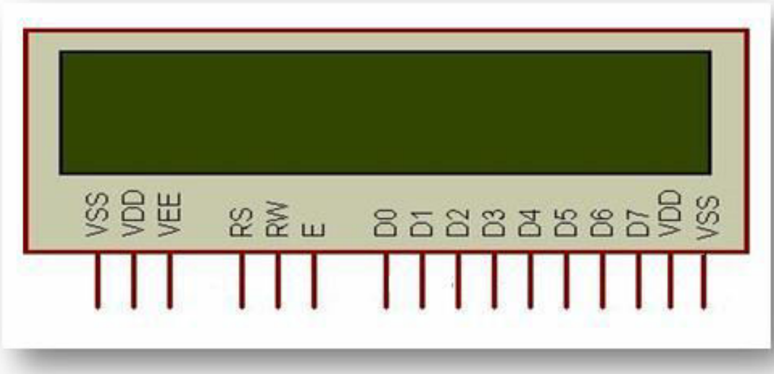


Fig. 3.17: Pin Diagram of LCD [35]

Unit cost of 16x2 LCD = Rs 114/-

Table 3.5 Pin Description of LCD

Sr. No	Pin No.	Pin Description
1	Pin 1 (GND)	This is a ground pin to apply a ground to LCD.
2	Pin 2 (VCC)	This is the supply voltage pin to apply voltage to LCD.
3	Pin 3 (VEE)	This is the pin for adjusting a contrast of the LCD display by attaching a variable resistor in between VCC and GND.
4	Pin 4 (RS)	RS stands for Register Select. This pin is used to select command/data register. If RS=0 then command register is selected. If RS=1 then data register is selected.
5	Pin 5 (R/W)	R/W stands for Read/Write. This pin is used to select the operation Read/Write. If R/W=0 then Write operation is performed. If R/W=1 then Read operation is performed.
6	Pin 6 (EN)	En stands for Enable signal. A positive going pulse on this pin will perform a read/write function to the LCD.
7	Pin 7-14 (DB0-DB7)	This 8 pin is used as a Data pin of LCD.
8	Pin 15 (LED+)	This pin is used with pin 16(LED-) to setting up the illumination of back light of LCD. This pin is connected with VCC.

Inter-Integrated Circuit (I2C) for LCD

To interface 16x2 LCD directly with RichDuino Board, minimum 6 pins are needed: RS, EN, D7, D6, D5, and D4. With this I2C interface LCD module, it requires only 2 lines (I2C) to display information. Since there are only 13 digital input/output pin available in RichDuino Board. It is highly required to minimize the interfacing pin with RichDuino Board and save remaining I/O pins for sensor and actuator interfacing.

Cost of single unit of I2C = Rs 116/-

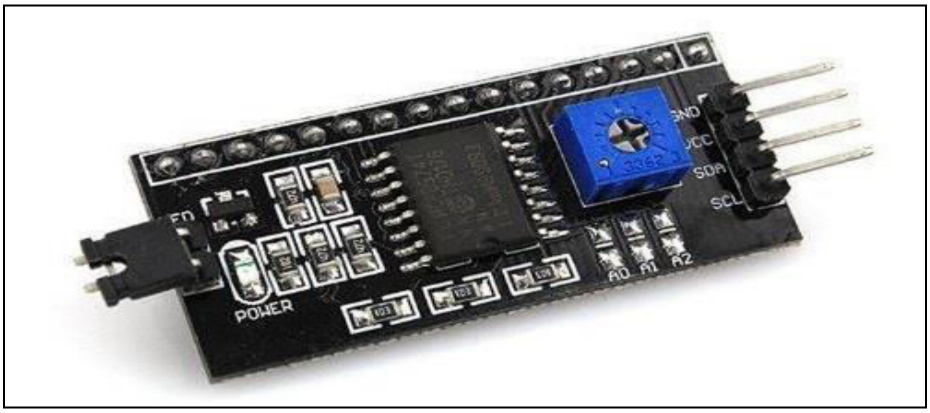


Fig. 3.18: I2C for LCD [36]

3.2.3.5 Actuator

The actuator deployed here is Plastic Water Solenoid Valve which is perfect for controlling flow of water to a drip irrigation system. In this paper we are controlling a solenoid valve with a RichDuino and a transistor.

Before choosing a solenoid valve there are a few things that should be considered:

- Water can only flow in one direction through this valve.
- This solenoid valve is not rated for food safety or use with anything but only with water.

Fig. 3.19: Solenoid valve [37]

Working Principle of Solenoid valve

The inlet water pressure holds the valve closed so it is required to have a minimum inlet water pressure of 3psi to keep the valve closed.

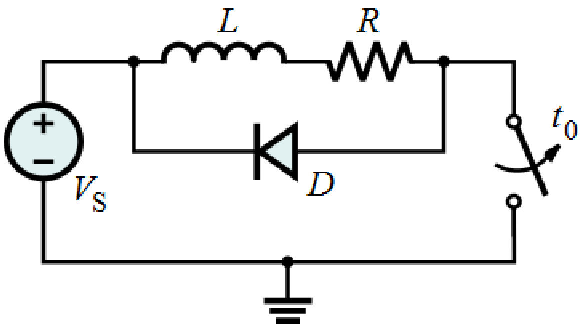


Fig. 3.20: Snubber diode [38]

Since a solenoid is an inductive load we need to include a snubber diode across the contacts. Snubber diodes helps to eliminate transient voltages caused when a magnetic coil suddenly loses power. Without this diode in place the transient voltage spikes can damage other elements of the circuit.

The snubber is placed from the negative side of the coil to the positive side. Since diodes only allow current to flow in one direction we need to make sure we get this right, otherwise it will be a dead short between power and ground.

The current draw of this solenoid is higher than a standard transistor can handle so we will be using a TIP120 Darlington Transistor. A Darlington transistor is actually a pair of transistors that act as a single transistor with a high current gain.

Expansion Connector

Fig. 3.21: Expansion Connector

The expansion connector are used to interconnect between the pins of different devices during development of the project. Soldering can be avoided during the designing process with the help of these connectors.

3.2.3.6 Integration of Selected Component for Designing of Sensor Node

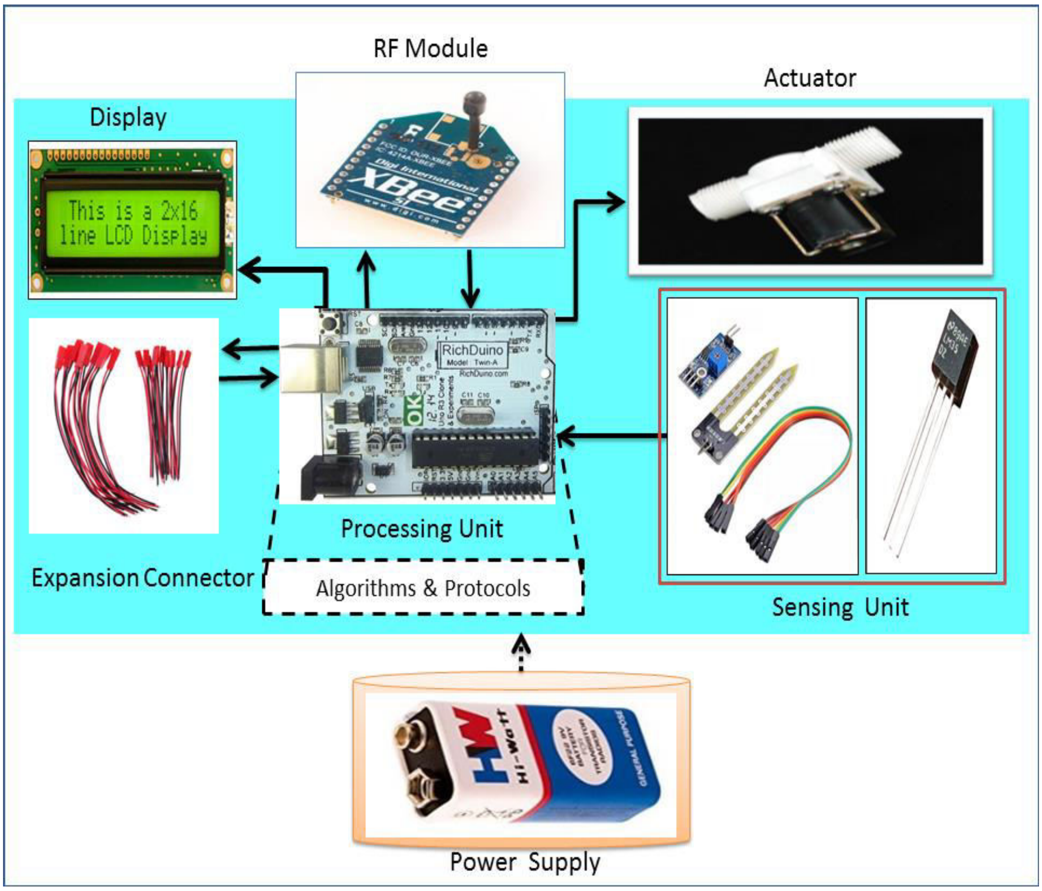


Fig. 3.22: Integration of Selected Components for Designing of Sensor Node

3.2.3.7 Cost Analysis of Sensor node

Table 3.5 Calculation of total cost of single unit of Sensor Node

Item name	Unit Price (Rs)	No. of Unit	Total (Rs)
RichDuino Board	750	1	750
XBee Series 1	1100	1	1100
XBee Shield	400	1	400
Battery	150	1	15
LCD	144	1	144
I2C	110	1	110
Soil moisture sensor	110	1	110
LM35	50	1	50
Solenoid Valve	200	1	200
Total Price			2879

3.3 Firmware Description

In the field of electronics and computing, firmware is defined as a type of software_program or set of instructions programmed on a hardware device. It provides the necessary instructions for how the device communicates with other hardware interfaced with the processing unit. Generally firmware is stored in non-volatile memory devices such as ROM, EPROM, or flash memory.

Firmware can be regarded as "semi-permanent" since it remains the same unless it is updated. Typical examples of devices containing firmware are embedded systems (such as traffic lights, consumer appliances, remote controls and digital watches), computers, computer peripherals, mobile phones, and digital cameras. The firmware contained in these devices provides the low-level control program for the device.

Changing the firmware of a device may rarely or never be done during its economic lifetime; some firmware memory devices are permanently installed and cannot be changed after manufacture. Common reasons for updating firmware include fixing bugs or adding features to the device. This may require flash memory to be reprogrammed through burning of the chip.

3.3.1 Firmware Performance Requirements

i) The network formed by gateway, router and sensor node should be self-forming and self-healing. If sufficient redundant routers are placed at strategic locations in the network, then a node will always be able to get an alternate path to send data to the destination in case of failure of one route.

ii) The network should have a long lifetime.

Coding in embedded C

The high level language used for coding purpose is C programming since it is the most widely used programming language for embedded processors and controllers. Assembly language is also used but mainly to implement those portions of the code where very high timing accuracy, code size efficiency, etc. are prime requirements. A standard coding style should be maintained while writing the program.

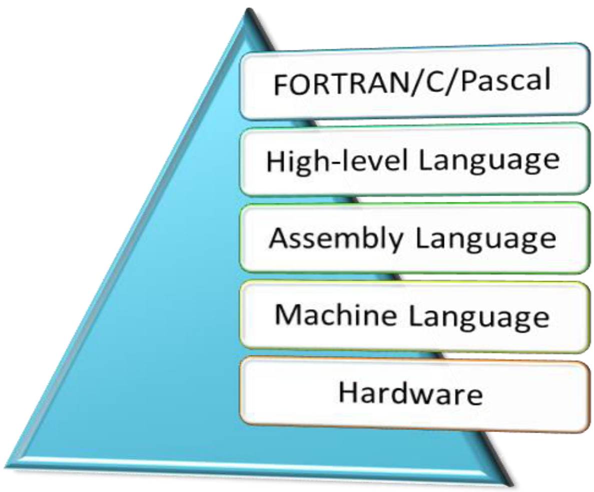


Fig. 3.23: Hierarchy of Programming Languages

A Standard coding style has following features:

- It has uniform appearance of code
- It provides sound understanding of code and encourages good programming practices.

As here we are developing an embedded system programming synchronization among the peripherals is highly required.

3.3.1 Platform used for Firmware Programming

Arduino IDE which is an open source software to write code and burn the program on the microcontroller. It is compatible with Windows, Mac OS X, and Linux operating system. The IDE is developed in Java and based on Processing and other open-source software.

3.3.2.1 Features of Arduino IDE

- Writing Sketches - Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension.ino. The message area gives feedback while saving and exporting and also displays errors. The bottom right hand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.
  - Verify
    Checks your code for errors compiling it.
  - Upload
    Compiles your code and uploads it to the configured board.
  - New
    Creates a new sketch.
  - Open
    presents a menu of all the sketches in your sketchbook. Clicking one will open it within the current window overwriting its content.
  - Save

Stores your sketch.

- Serial Monitor
  Opens the serial monitor.

- Verify/Compile
  Checks your sketch for errors compiling it; it will report memory usage for code and variables in the console area.

- Upload
  Compiles and loads the binary file onto the configured board through the configured Port.

- Include Library
  Adds a library to your sketch by inserting #include statements at the start of your code. Additionally, from this menu item you can access the Library Manager and import new libraries from .zip files.

- Add File
  Adds a source file to the sketch (it will be copied from its current location). The new file appears in a new tab in the sketch window. Files can be removed from the sketch using the tab menu accessible clicking on the small triangle icon below the serial monitor one on the right side of the toolbar.

3.3.3 Firmware development

Microcontroller has been programmed to test the hardware as well to achieve the goal of WSN application, which involved the following steps:

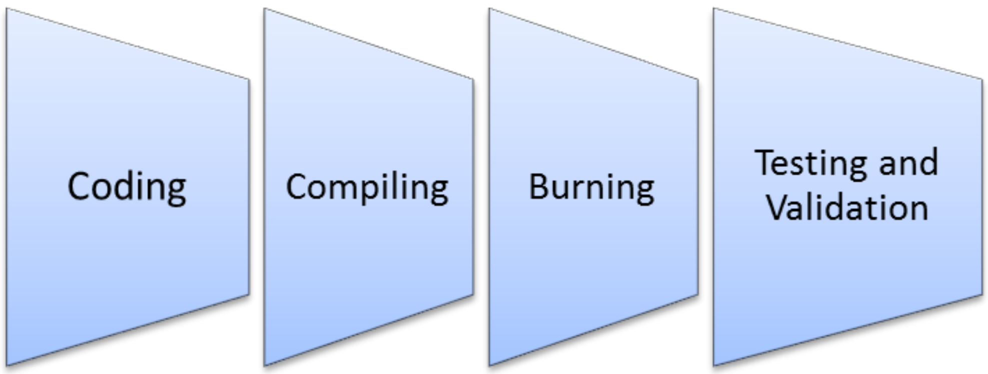


Fig. 3.24: Process for Firmware Development

- Coding in a high-level language (such as C, or Java) or assembler.

- A compiler for a high level language helps to reduce production time.
- Burning the developed program in the microcontroller chip through the USB cable.
- Test the device for proper functioning in critical environment. If it is working well validate it, otherwise update the program and transfer it to the microcontroller. It is a continuous rigorous process till the desired end result is achieved.

3.4 Software Requirements

3.4.1 X-CTU Software

X-CTU is a free software tool available from Digi International to interface with Xbee modules. The tool provides a GUI and terminal interface to configure the modules as well as a built in tool to test the Xbee range and reliability of packet transmissions.

Features:

- Support for all Digi products.
- Integrated terminal window.
- Easy to use loopback range test.
- Display of Receive Signal Strength Indictator (RSSI).
- Upgrade RF module firmware in the field on all Digi RF Products.
- Display both ASCII and hexadecimal characters in terminal window.
- Compose test packets in either ASCII or hexadecimal for transmitting in terminal interface.
- Save and retrieve commonly used module configurations (profiles).
- Automatically detect module type.
- Restore factory default parameters.
- Display help about each of the radio parameters.
- Program radio profiles in a production environment using command line interface.

The software is easy to use and allows testing the radio modems in the actual environment with just a computer and the items included with the radio modems.

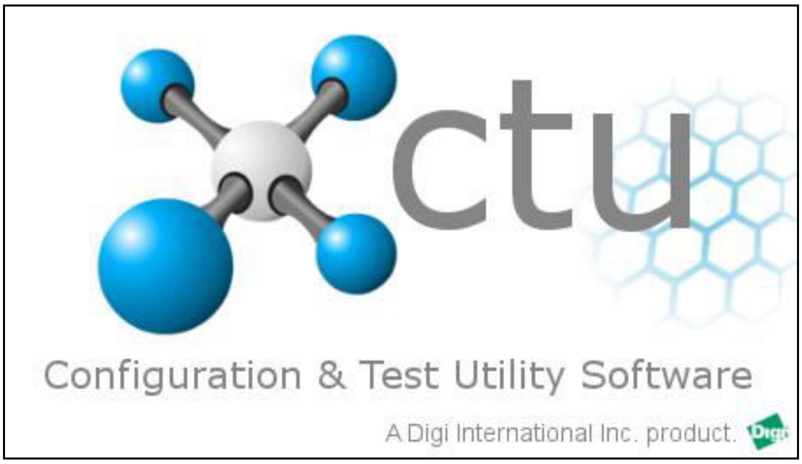


Fig. 3.24: Screenshot of X-CTU Software [38]

3.5 Step by Step Design Methodology of Developing WSN

3.5.1 LCD interfacing:

The display unit for sensor node is LCD. The processing unit of sensor node is RichDuino, hence it is required to interface the LCD with the RichDuino board. Initially 16 pins of LCD are soldered with I2C. Then the 4 pin out of I2C i.e. SCL, SDA, VCC, GND are connected to corresponding pin of RichDuino board with the help of female connectors as shown below.

Code for LCD interfacing:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x3F, 16, 2);
void setup () {
Serial.begin(9600);
lcd.begin(16,2);
lcd.backlight();
}
void loop()
```

```
{
lcd.setCursor(0,0);
lcd.print("Welcome to");
lcd.setCursor(0,1);
lcd.print("OPNET LAB");
delay(1000);
}
```

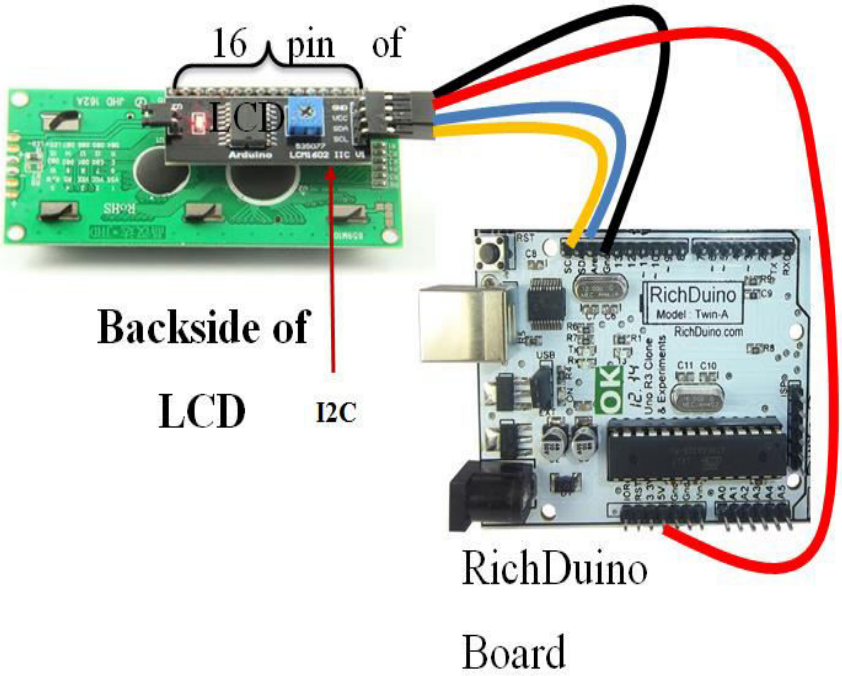


Fig. 3.25: LCD interfacing with RichDuino Board

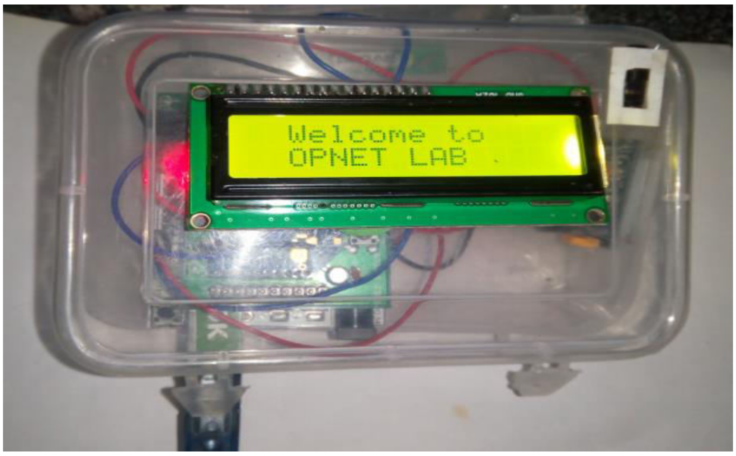


Fig. 3.25: Output obtained after execution of LCD program

3.5.2 Interfacing of Temperature Sensor (LM35):

The temperature sensor i.e. LM35 is interfaced with analog pin (A0) of the RichDuino Board. The analog voltage received is then converted to 10 bit digital value by the inbuilt ADC in RichDuino Board. The received input is then calibrated to observe the output in Celcius. 5000 mV is divided by 2^{10} i.e. 1024 then multiplied with received input to get the output in mV, then again it is divided by 10 to get result in ^{o}C. According to LM35 datasheet there is 10 mV increase per ^{o}C.

Code for Temperature Sensor Interfacing:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x3F, 16, 2);
const int sensorPin = 0; // connect sensor to analog input 0
void setup()
{
Serial.begin(9600);
pinMode(sensorPin, INPUT);// enable output on the led pin
}
void loop()
{
intvalue1 = analogRead(sensorPin);
float mv = ( value1/1024.0)*5000;
float cel = mv/10;
lcd.clear();
lcd.print("Sensor Value ");
lcd.setCursor(0,1);
lcd.print(cel);
lcd.print(" Celcius");
delay(1000);
}
```

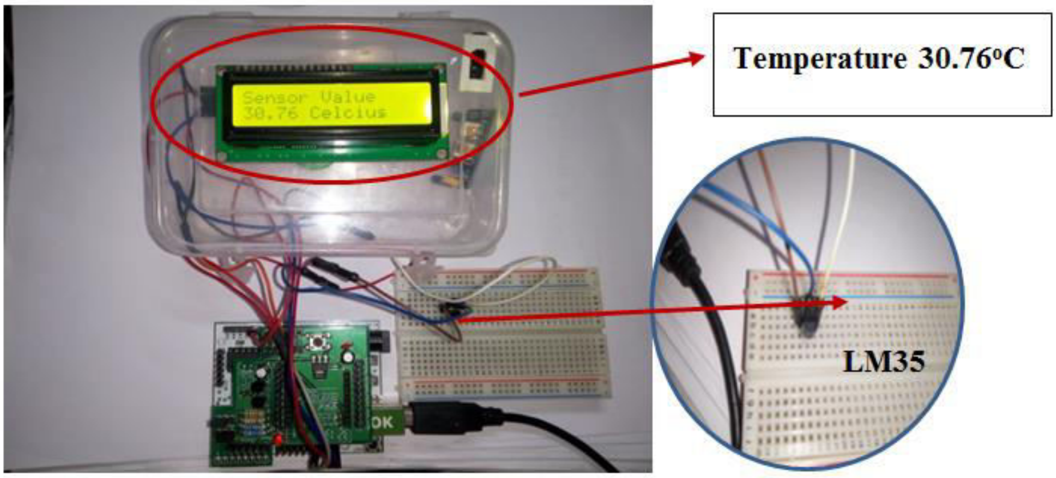


Fig. 3.26 Set-up for Temperature Sensor (LM35) Interfacing and its output on LCD

3.5.3 Interfacing of Soil Moisture Sensor:

The soil moisture sensor is interfaced with RichDuino Board to estimate the water content in the soil. While experimentation it is observed that soil content the soil content is inversely proportional to the received analog voltage. Therefore in the calibration part we have to divide a standard value with the received voltage. To measure the volumetric content of water i.e. ratio of volume of water to volume of soil need to be calculated.

Code for Soil Moisture Sensor

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
const int sensorPin = 0;
XBee xbee = XBee();
LiquidCrystal_I2C lcd(0x3F, 16, 2);
void setup()
{
   Serial.begin(9600);
   lcd.begin();
   lcd.backlight();
}
void loop()
{
  int value1 = analogRead(sensorPin);
```

```
  float mv = ( value1/1024.0)*5000;
  float soil = mv/10;
  lcd.clear();
  lcd.print("Soil Moisture ");
  lcd.setCursor(0,1);
  lcd.print(soil);
  lcd.print("Value");
  delay(1000);
}
```

- **Experiment Set-up to measure water content in dry soil**

Fig. 3.27 Experiment Set-up with Soil Moisture Sensor in dry soil and its output on LCD

- **Experiment Set-up to measure water content in wet soil**

Fig. 3.28: Experiment Set-up with Soil Moisture Sensor in wet soil and its output on LCD

3.5.4 Configuring XBee through X-CTU Software and USB explorer module.

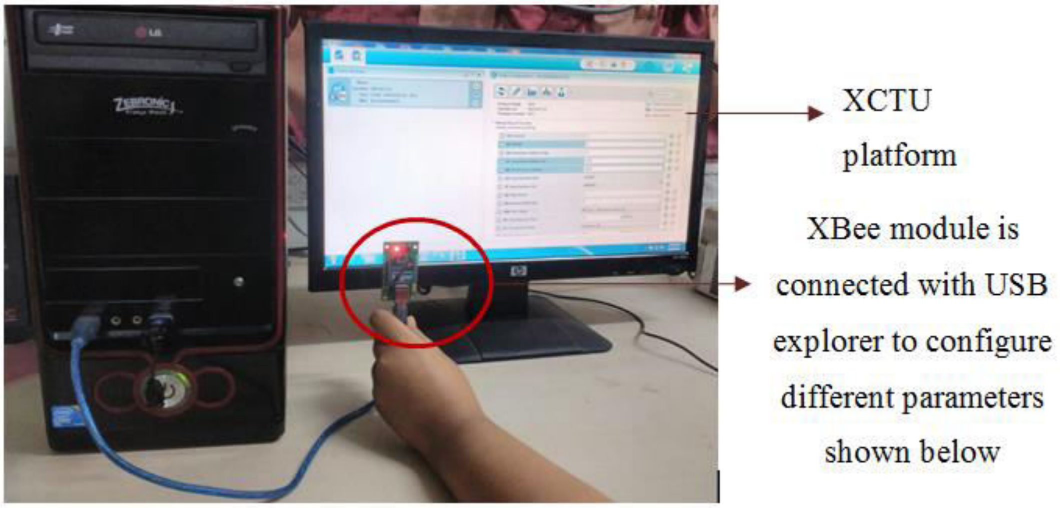


Fig. 3.29a: Configuring XBee through X-CTU Software and USB explorer

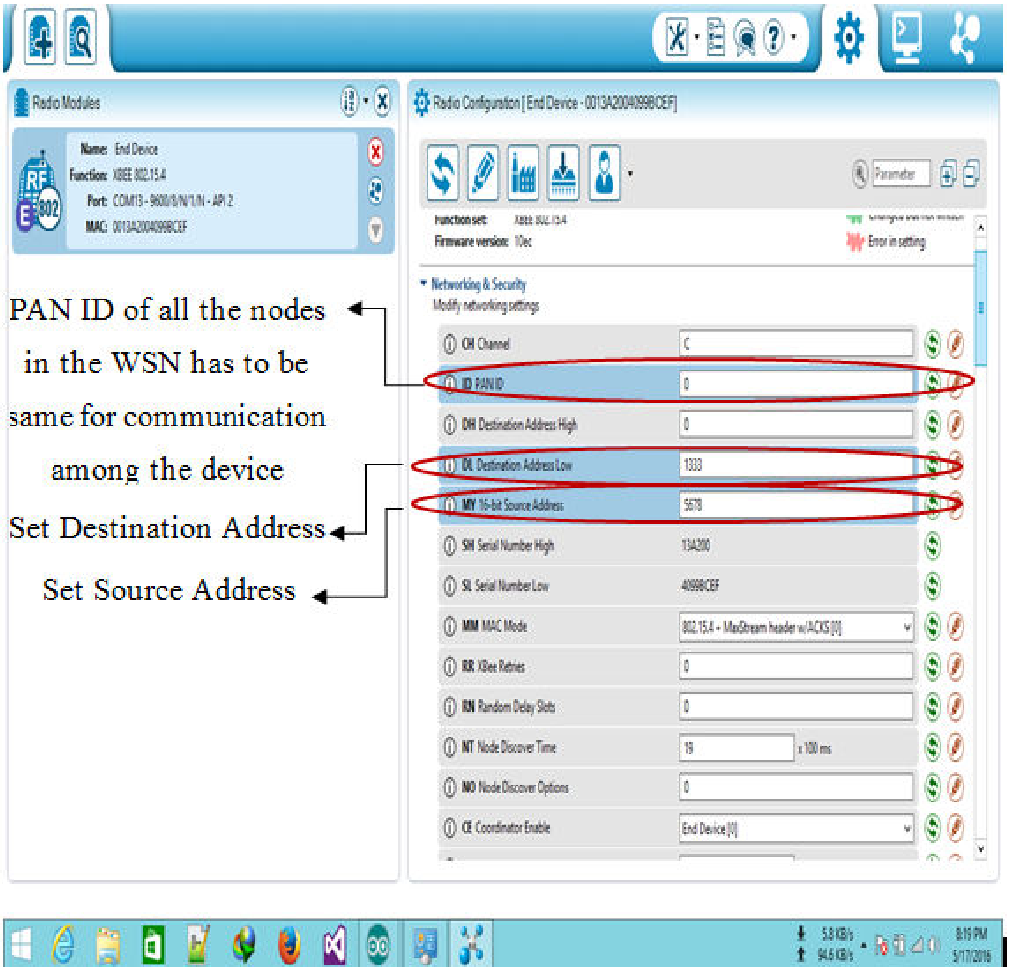


Fig. 3.29b: Configuring XBee through X-CTU Software and USB explorer module. Parameters that are required to configure has been encircled.

3.5.5 Communication set-up between XCTU USB explorer and RichDuino board

Transmitter = USB explorer + RF module

Receiver = RichDuino + RF module

Code for Receiver:

```
int readvalue=0;
void setup(){
Serial.begin(9600);
}
void loop(){
if(Serial.available()>14){
for(inti=0;i<14;i++){
Serial.print(Serial.read(),HEX);
Serial.print(",");
}
Serial.println();
}
}
```

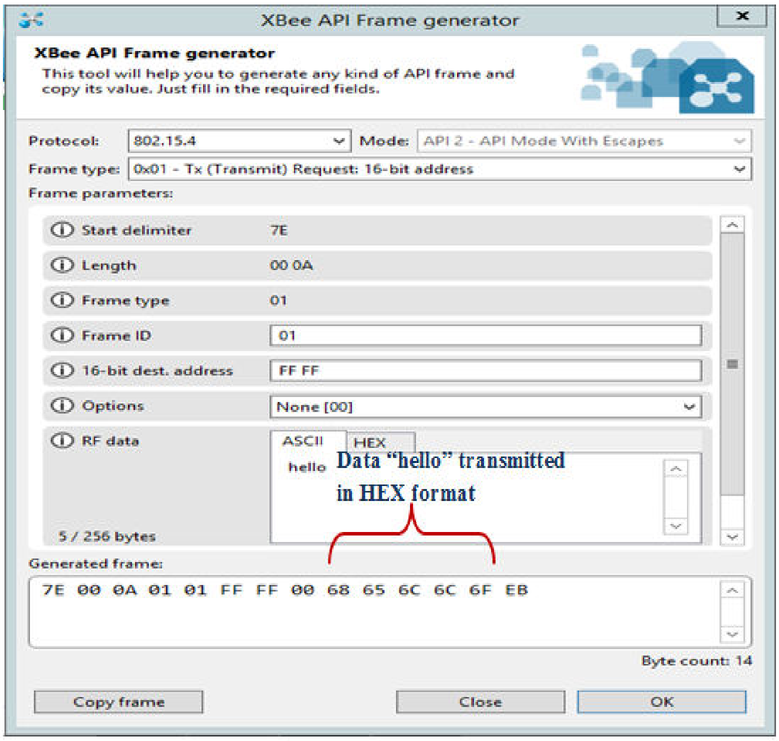


Fig. 3.30: API packet formation to send data to the Receiver

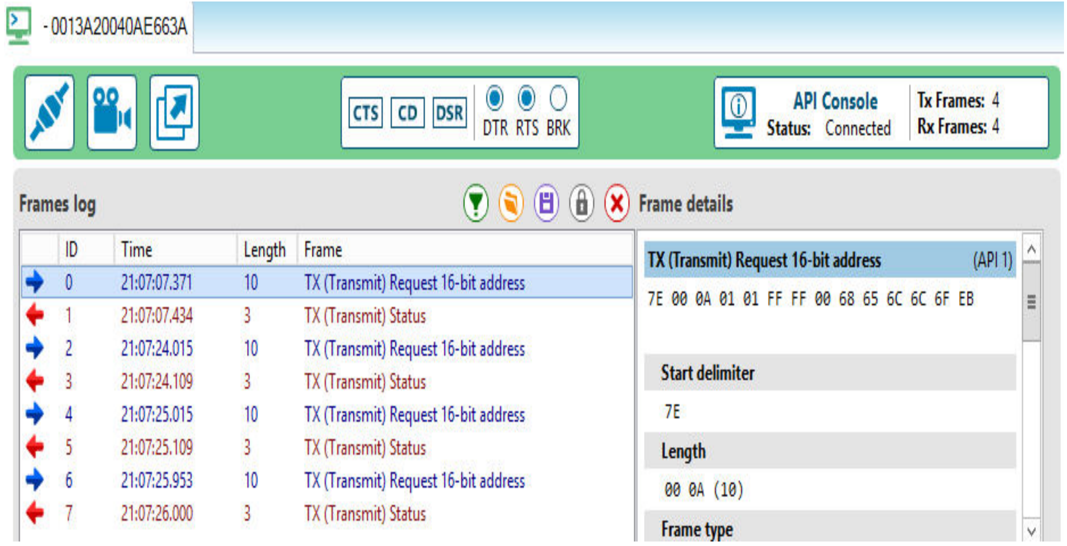


Fig. 3.31 Screenshot of Transmission of data and received status in X-CTU platform

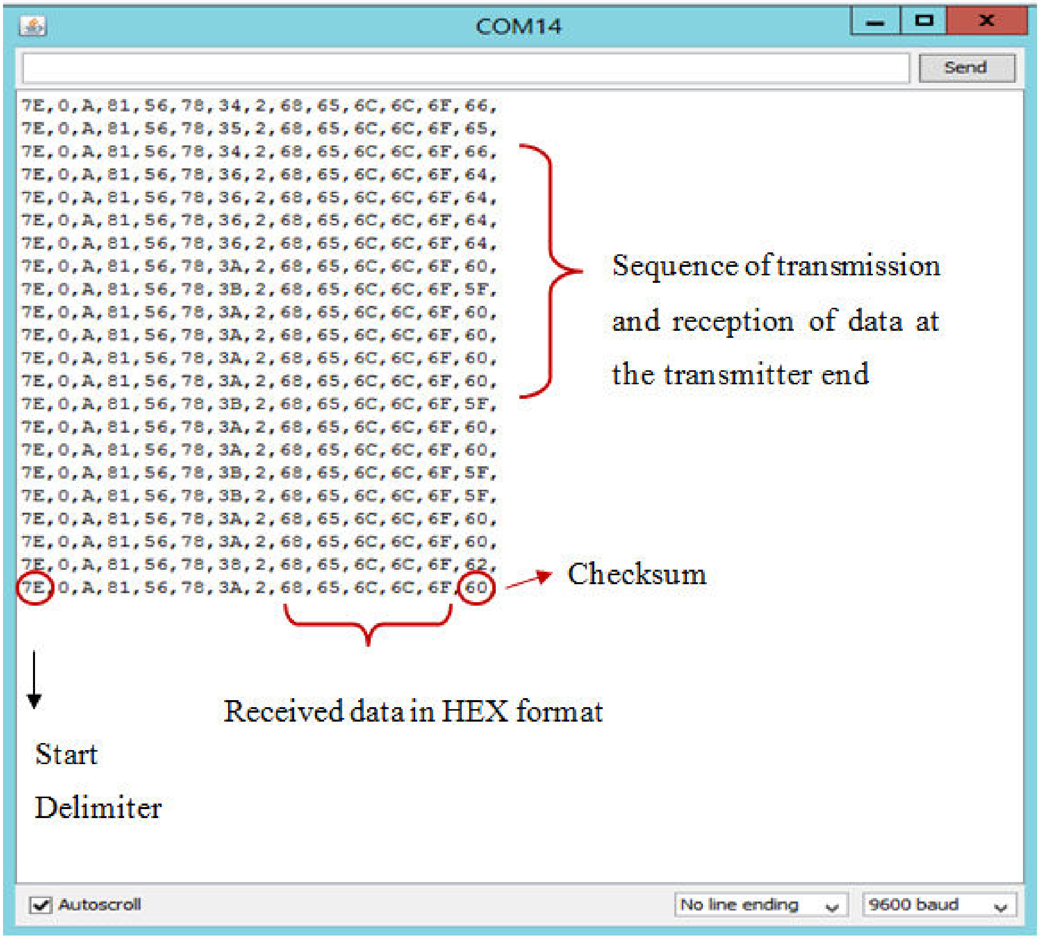


Fig 3.32 Screenshot of Received Data in the Arduino IDE

3.5.6 Transmitting mode of XBee: API and AT

XBee's support two modes of operation: API and AT. In API mode, we can communicate with the radio by sending and receiving packets. In AT (transparent) mode, the XBee radio simply relays serial data to the receiving XBee, as it is to the destination address [39]. A brief overview of the main advantages of API (Application Program Interface) Mode vs. AT (Transparent Mode) has been discussed in following section:

AT (Transparent) Mode

- Simple
- Compatible with any device that send data serially.
- Primarily for point to point communication between two XBees. It's possible to communicate with multiple XBees but this requires entering command mode each time to change the destination address.

API (Packet) Mode

- I/O Samples. This feature allows an XBee to receive I/O data from 1 or more remote XBees
- Acknowledgement (ACK) and Retries. When sending a packet, the transmitting radio receives an ACK, indicating the packet was successfully delivered. The transmitting radio will resend the packet if it does not receive an ACK.
- Receive packets (RX), contain the source address of transmitting radio
- Configure a remote radio with the Remote AT feature
- Easily address multiple radios and send broadcast TX packets
- Obtain RSSI (signal strength) of an RX packet
- Packets include a checksum for data integrity

XBee Series 1 radios support both AT and API modes with a single firmware version, allows to switch between the modes with X-CTU. However, Series 2 requires a specific firmware for API mode which is tedious process. Source address, RSSI value, checksum can be easily extracted from the received API packet. Even in the transmitter side acknowledgement is received to confirm whether the packet is received or dropped.

3.5.7 Transmission and Reception of "Hello" packet in AT mode

Transmitter = RichDuino + RF module

Receiver = RichDuino + RF Module

While transmitting the data in AT mode the destination address must be set in command mode before transmission of data in transmit mode. Here configureRadio() function is used to configure the radio in command mode. To enter into command mode "+++" is used and to come out of command mode "ATCN" AT command is given.

Code for Transmitter:

```
boolean configured;
boolean configureRadio() {
// put the radio in command mode:
Serial.print("+++");
String ok_response = "OK\r"; // the response we expect.
// Read the text of the response into the response variable
String response = String("");
while (response.length() < ok_response.length()) {
if (Serial.available() > 0) {
response += (char) Serial.read();
}
}
// If we got the right response, configure the radio and return true.
if (response.equals(ok_response)) {
Serial.print("ATDH0013A200\r"); // destination high
Serial.print("ATDL4099BCEF\r"); // destination low
Serial.print("ATCN\r"); // back to data mode
return true;
} else {
return false; // This indicates the response was incorrect.
}
}
void setup () {
Serial.begin(9600); // Begin serial
configured = configureRadio();
}
void loop () {
```

```
        if (configured) {
        Serial.println("Hello!");
        delay(3000);
        }
        else {
        delay(30000); // Wait 30 seconds
        configured = configureRadio(); // try again
        }
}
```

Code for Receiver:

```
        int readvalue=0;
        char response;
        void setup(){
        Serial.begin(9600);
        }

        void loop(){
        while(Serial.available()>0){

        response=(char) Serial.read();
        Serial.print(response);
           }
 }
```

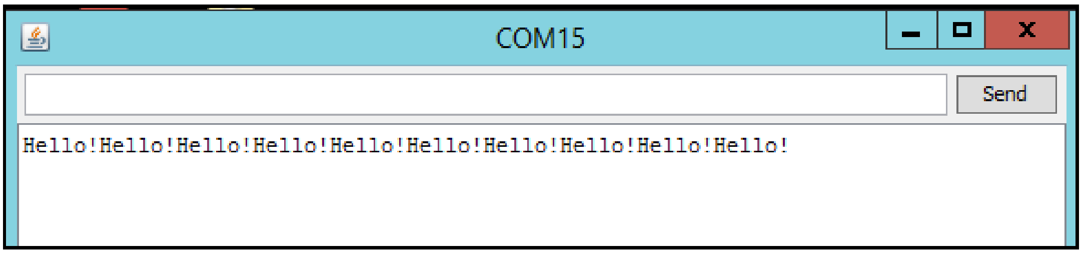


Fig. 3.33: Screenshot of Received "Hello" packet as it (not in HEX) in Serial monitor of Arduino IDE.

In the initial phase of development we started with AT mode since it is easy to implement than API mode and it was successfully performed then we proceeded with transmission of data in API mode due to its advantages over AT mode.

3.5.8 Transmission and Reception of "Hello" packet through XBee in API mode

Code for Transmitter:

```
#include <XBee.h>
XBee xbee = XBee();
uint8_t payload[] = { 'H', 'e', 'l', 'l', 'o'};
Tx16Request tx16 = Tx16Request(0x5678, payload, sizeof(payload));
void setup()
{
Serial.begin(9600);
xbee.setSerial(Serial);
}
void loop()
{
xbee.send( tx16 );
delay(50);
}
```

Code for Receiver:

```
#include <XBee.h>
XBee xbee = XBee();
XBeeResponse response = XBeeResponse();
Rx16Response rx16 = Rx16Response();
void setup() {
Serial.begin(9600);
xbee.setSerial(Serial);
}
void loop() {
String sample;
xbee.readPacket(100);
if (xbee.getResponse().isAvailable()) {
Serial.print("API ID = ");
Serial.println(xbee.getResponse().getApiId());
if (xbee.getResponse().getApiId() == RX_16_RESPONSE) {
xbee.getResponse().getRx16Response(rx16);
for (int i = 0; i < rx16.getDataLength(); i++) {
```

```
sample += (char)rx16.getData(i);
}
Serial.println(sample);
}
}else if (xbee.getResponse().isError()) {
Serial.println("Error reading packet.  Error code: ");
Serial.println(xbee.getResponse().getErrorCode());
}
}
```

API ID specifies the type of transmitted data packet. It is required to know the transmitted packet format for extraction of reqiured data from the packet.

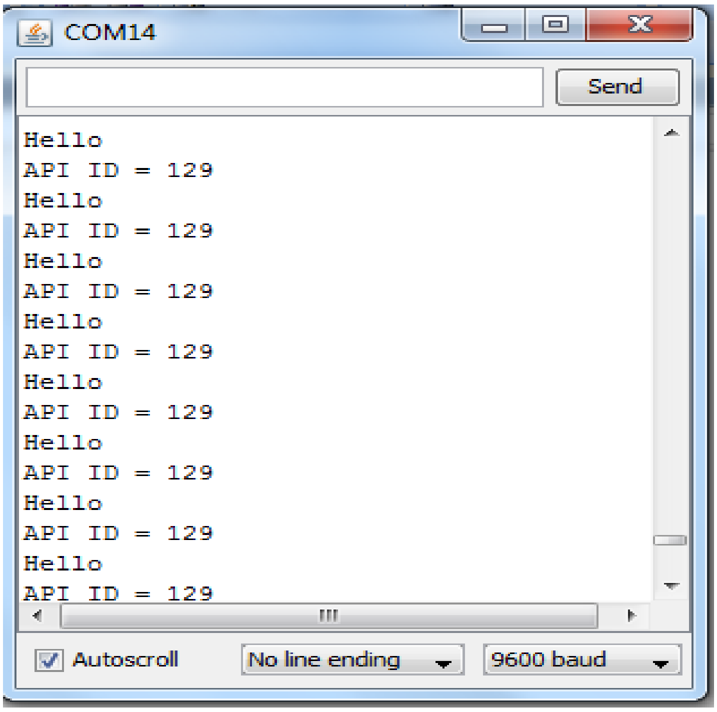


Fig. 3.34: Screenshot of Received "Hello" packet and API ID in Serial monitor of Arduino IDE.

3.5.9 Reception of RSSI value

Code for RSSI

```
#include <XBee.h>
XBee xbee = XBee();
Rx16Response rx16 = Rx16Response();
void setup()
{
```

```
  Serial.begin(9600);
  xbee.setSerial(Serial);
}
void loop()
{
   xbee.readPacket(100);
   if (xbee.getResponse().isAvailable())
   {
      if (xbee.getResponse().getApiId() == RX_16_RESPONSE)
      {
         xbee.getResponse().getRx16Response(rx16);
         Serial.print("RSSI Value ");
         Serial.print( rx16.getRssi());
         Serial.println();
      }
   }
  }
```

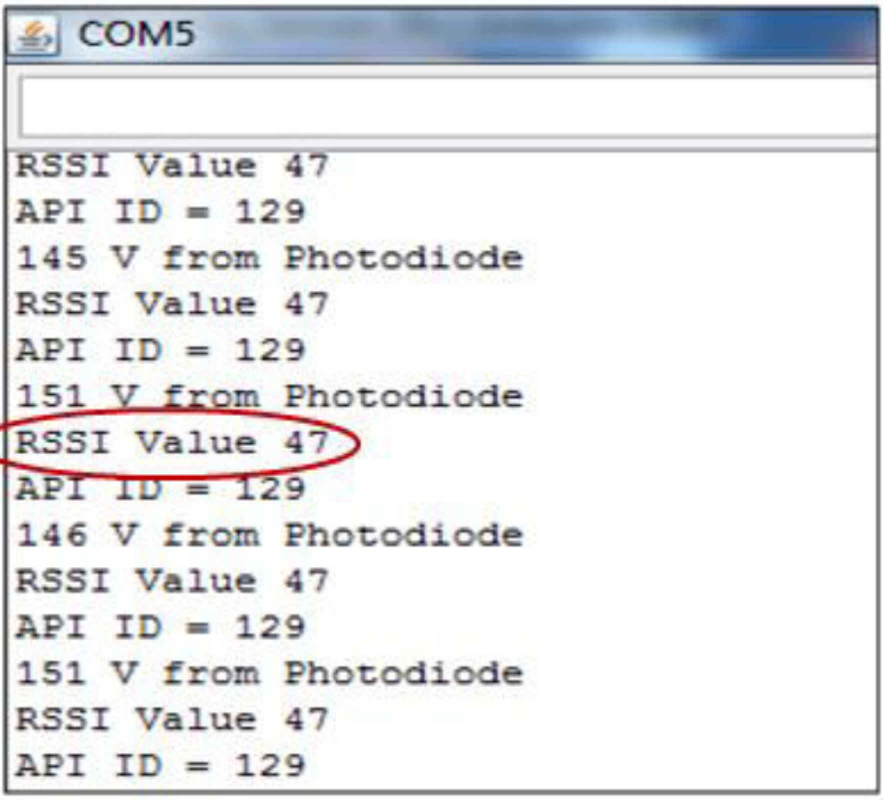
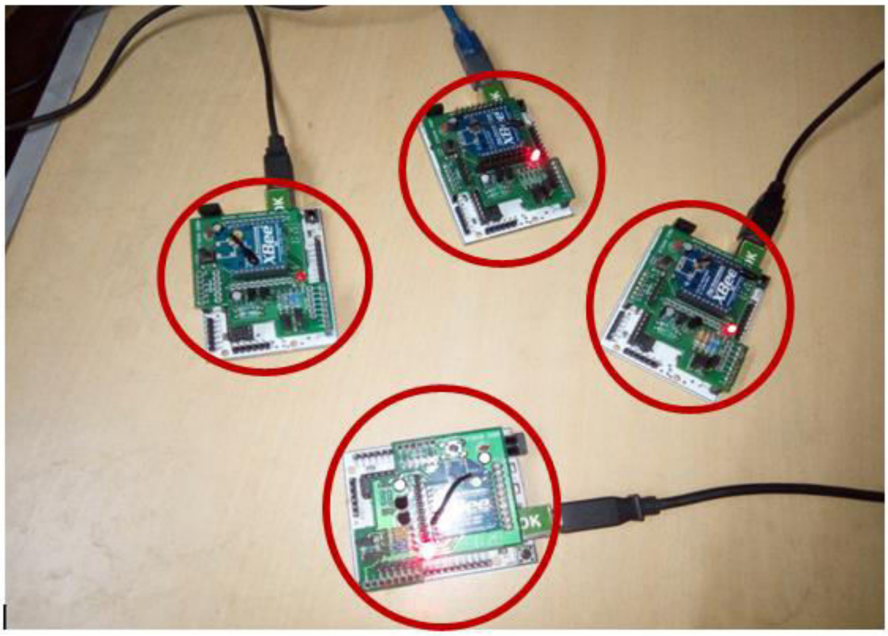
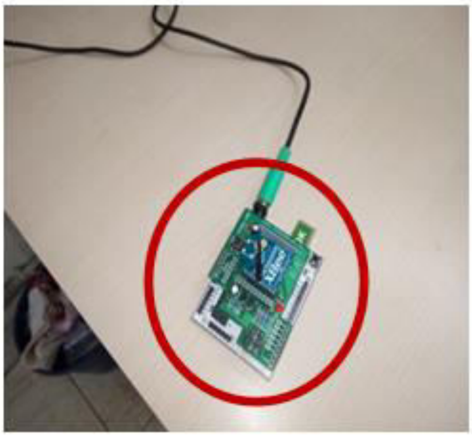


Fig. 3.35: Screenshot of received RSSI value in Serial monitor of Arduino IDE.

3.5.10 Complete WSN Experiment using 6 designed node i.e. 4 Sensor Node as S1, S2, S3 and S4



1 Router



1 Gateway

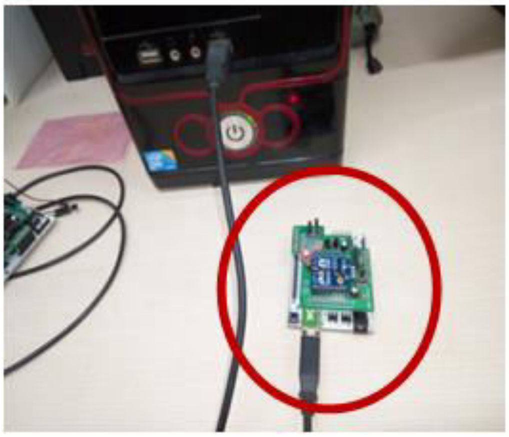


Fig. 3.36: Overview of 6 designed node as Sensor Node, Router and Gateway

- All node consist of RichDuino + RF Module

Code for Sensor Node

```
#include <XBee.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
const int sensorPin = 0;
XBee xbee = XBee();
```

```
LiquidCrystal_I2C lcd(0x3F, 16, 2);
uint8_t payload[] = {0,0,0,0};
void setup()
{
    Serial.begin(9600);
    lcd.begin();
    lcd.backlight();
    xbee.setSerial(Serial);
    pinMode(sensorPin, INPUT);
}
void loop()
{
  int value1 = analogRead(sensorPin);
float mv = ( value1/1024.0)*5000;
float cel = mv/10;
  lcd.clear();
  lcd.print("Sensor Value ");
  lcd.setCursor(0,1);
  lcd.print(cel);
  lcd.print(" Celcius");
  delay(1000);
  //break down 10-bit reading into two bytes and place in payload
  payload[0] = value1 >> 8 & 0xff;
  payload[1] = value1 & 0xff;
  payload[2] = 'S';
  payload[3] = '1';
  delay(500);

    Tx16Request tx16 = Tx16Request(0x5678, payload, sizeof(payload));
    xbee.send(tx16);
    delay(50);
}
```

Code for Router

```
#include <XBee.h>
XBee xbee = XBee();
XBeeResponse response = XBeeResponse();
```

```
Rx16Response rx16 = Rx16Response();
uint8_t payload1[] = {0,0,0,0};
void setup()
{
  Serial.begin(9600);
  xbee.setSerial(Serial);
}
void loop() {
 xbee.readPacket(100);
  if (xbee.getResponse().isAvailable()) {
   Serial.print("API ID = ");
   Serial.println(xbee.getResponse().getApiId());
   if (xbee.getResponse().getApiId() == RX_16_RESPONSE) {
    xbee.getResponse().getRx16Response(rx16);
    payload1[0] = rx16.getData(0);
    payload1[1] = rx16.getData(1);
    payload1[2] = rx16.getData(2);
    payload1[3] = rx16.getData(3);
    uint8_t analogMSB = rx16.getData(0);
   uint8_t analogLSB = rx16.getData(1);
   int value1 = analogLSB + (analogMSB * 256);
   delay(50);
      Tx16Request tx16 = Tx16Request(0x1333, payload1, sizeof(payload1));
  xbee.send(tx16);
  delay(50);
        Serial.print("RSSI Value ");
        Serial.print(rx16.getRssi());
        Serial.println();
   }
  }else if (xbee.getResponse().isError()) {
   Serial.println("Error reading packet.  Error code: ");
   Serial.println(xbee.getResponse().getErrorCode());
 }
}
```

Code for Gateway

```
#include <XBee.h>
XBee xbee = XBee();
XBeeResponse response = XBeeResponse();
Rx16Response rx16 = Rx16Response();
void setup() {
  Serial.begin(9600);
  xbee.setSerial(Serial);
}
void loop() {
String sample;
  xbee.readPacket(100);
    if (xbee.getResponse().isAvailable()) {
      Serial.print("API ID = ");
      Serial.println(xbee.getResponse().getApiId());
      if (xbee.getResponse().getApiId() == RX_16_RESPONSE) {
        xbee.getResponse().getRx16Response(rx16);
        uint8_t analogMSB = rx16.getData(0);
      uint8_t analogLSB = rx16.getData(1);
      int value1 = analogLSB + (analogMSB * 256);
      for (int i = 2; i < rx16.getDataLength(); i++) {
         sample += (char)rx16.getData(i);
        }
      float mv = ( value1/1024.0)*5000;
      float cel = mv/10;
      Serial.print(cel);
      Serial.print("Celcius Temperature from Sensor Node ");
      Serial.println(sample);
            Serial.print("RSSI Value ");
            Serial.print(rx16.getRssi());
            Serial.println();
      }
    }else if (xbee.getResponse().isError()) {
      Serial.println("Error reading packet.  Error code: ");
      Serial.println(xbee.getResponse().getErrorCode());
  }     }
```

➢ **Output at the Gateway**

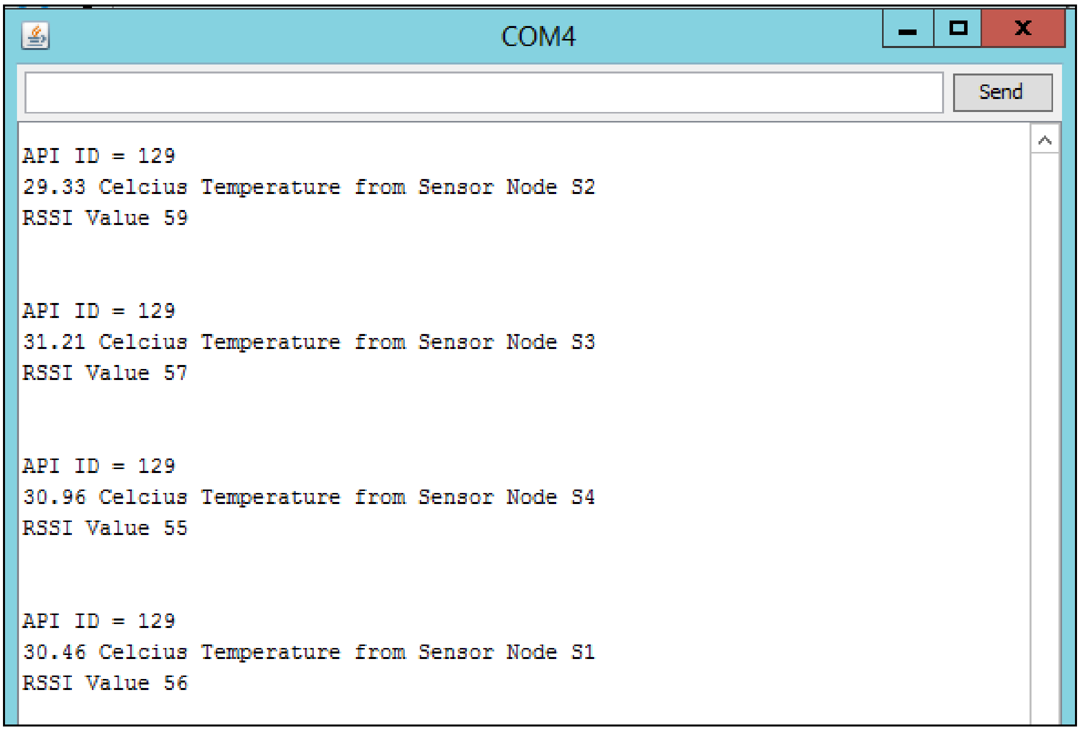


Fig. 3.37: Screenshot of received data at the gareway from 4 sensor node S1, S2, S3 and S4

3.5.11 Raspberry Pi Interfacing and Software Development in dot net

➢ **Raspberry Pi Serial Communication**

Pre-Requirements

1. A Raspberry Pi 2
2. Power Source for R'Pi (5v, 2Amp micro USB)
3. Ethernet Cable
4. Memory card for R'Pi
5. Keyboard & Mouse for R'Pi
6. Latest NOOBS zip (download from official R'Pi Website)
7. A PC with Windows 10
8. Latest Visual Studio 2015 with UWP Update [40]

Installation of Operating System

- At first extract the NOOBS zip and copy everything from the extracted folder to the newly formatted memory card.
- Insert the memory card in your Raspberry Pi and Connect Display, Mouse, Keyboard & power source with your Raspberry Pi.
- After some time, the NOOBS OS will boot up. Select **Windows 10 IoT Core** in the list and click Install.

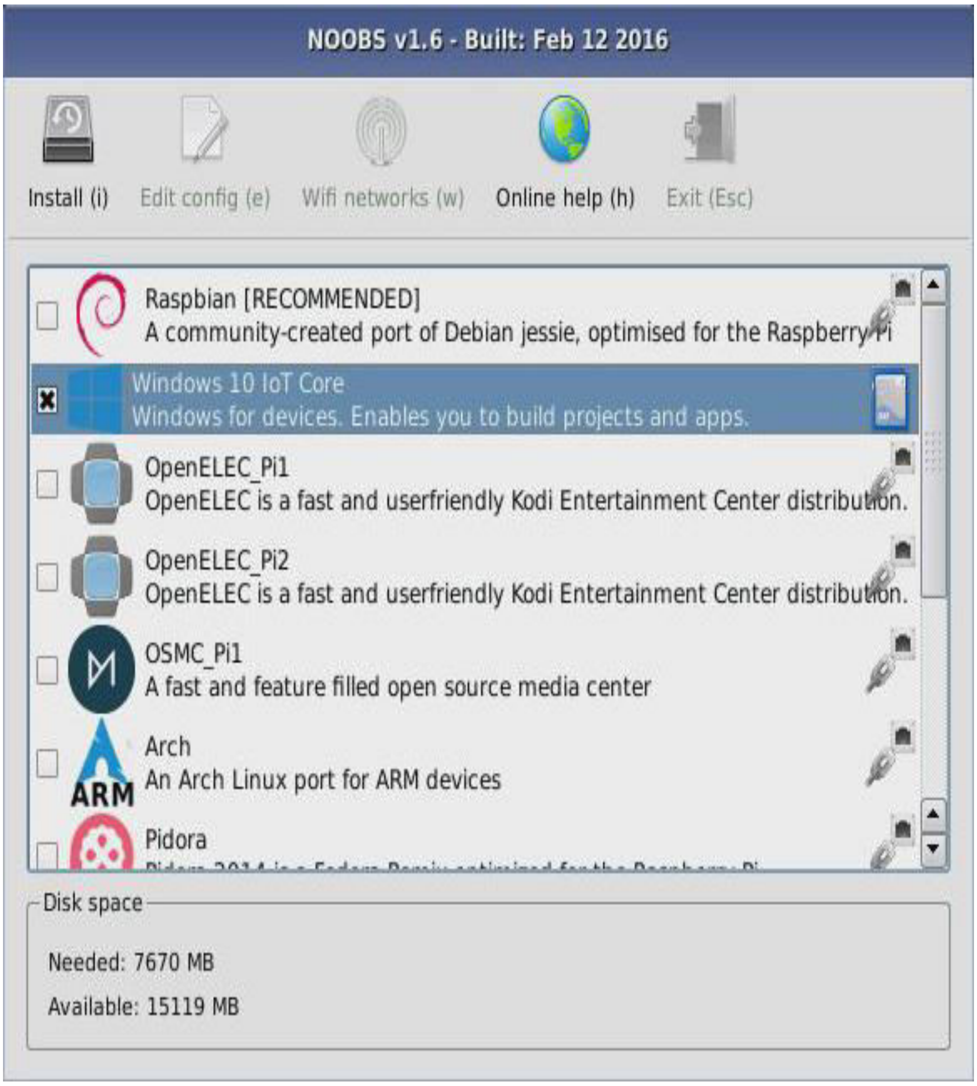


Fig. 3.38: Screenshot of NOOBS

Developed Software to Communicate with Serial Device from any Windows 10 Device:

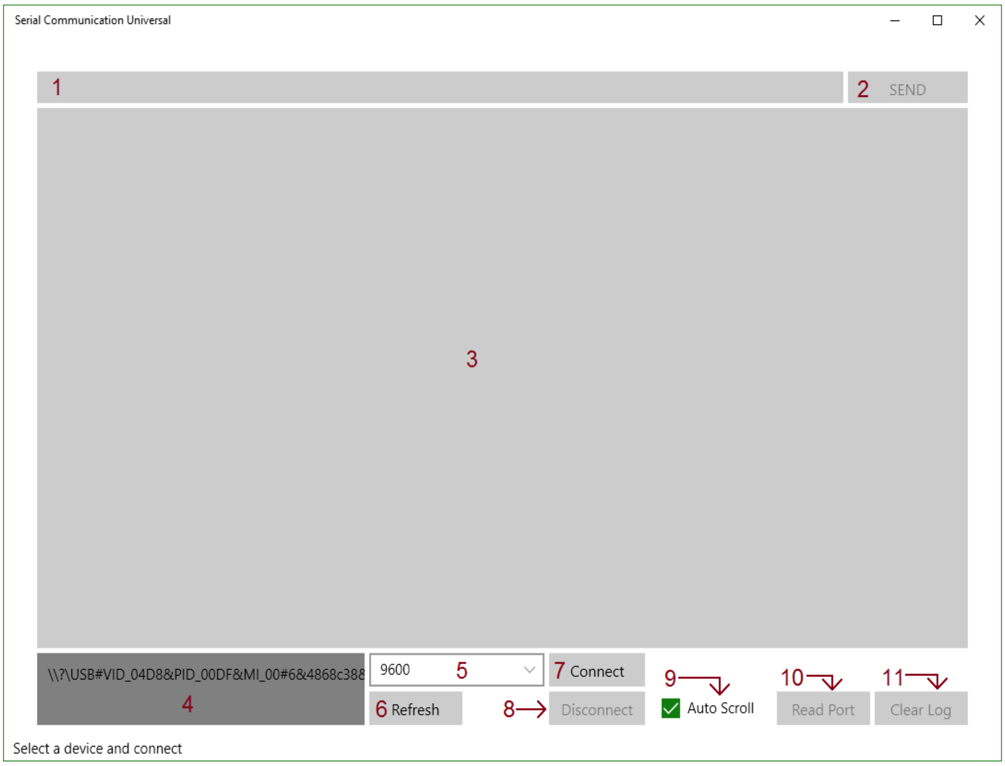


Fig. 3.39: Screenshot of Designed Software for Serial Communication

1. Write Textbox: Enter the text here what we want to send to your Serial Device.
2. Send Button: Press this button to send the text that has been entered in Write Textbox to your Serial Device.
3. Read Textbox: We can see the data from your Serial Device after connecting with the device.
4. Serial Devices List box: Here we can see all connected Serial Device in this list box and we can select one of them to connect.
5. Baud Rate Combo Box: We can choose the baud rate of your Serial Device at which it will be connected.
6. Refresh Button: Press this button to refresh the connected device list.
7. Connect Button: After selecting the device and choosing the baud rate, press this button to get connected with available Serial Device.
8. Disconnect Button: Press this button to disconnect with the Serial Device.

9. Auto Scroll: Enable it to scroll to the last text automatically in the Read Textbox.
10. Read Port Button: Press this button to start reading from the Serial Device.

Clear Log Button: Press this button to clear the Read Textbox

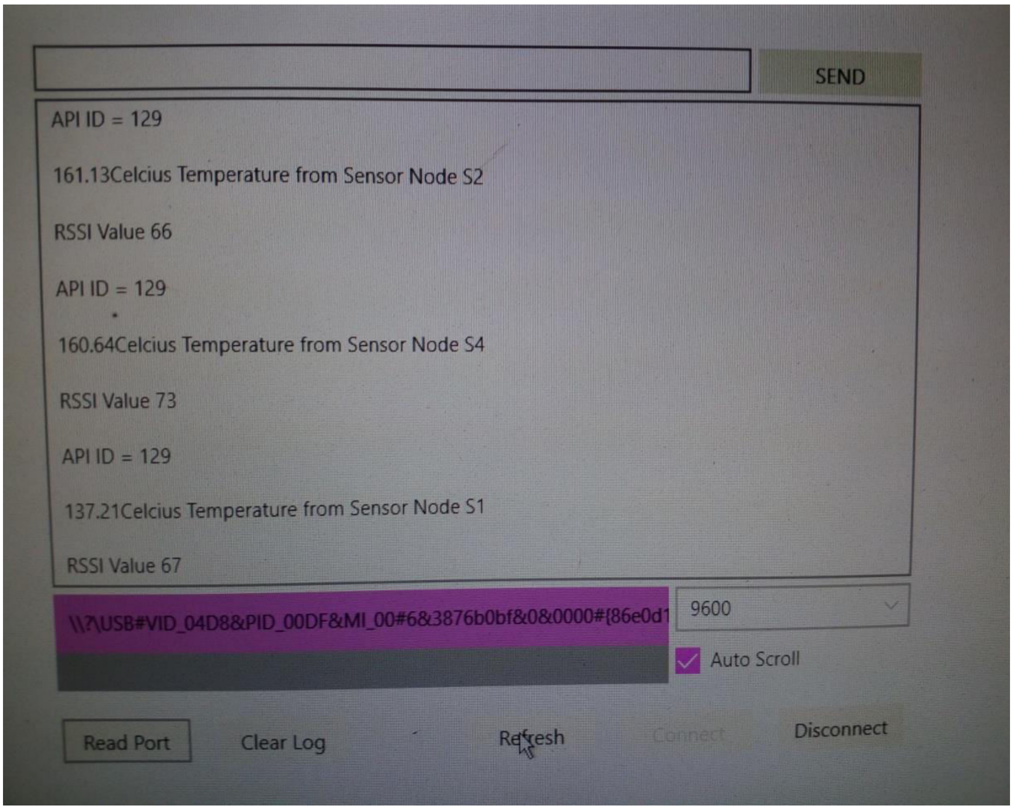


Fig. 3.40: Screenshot of Received Serial Data from Raspberry Pi.

Hence a complete project of building a low cost, low power WSN for agriculture purpose has been performed successfully.

Chapter 4

RESULT AND DISCUSSION

4.1 INTRODUCTION

We have already shown the hardware and firmware implementation of proposed WSN model. Step by step methodology of designing each component of WSN has been discussed briefly in Chapter 3. All the experiment has been performed successfully to establish communication link between the nodes. Since we have work on API mode for transmission of data reliability of network has been enhanced. The packet drop is very less around 0.1% that has been checked in X-CTU Software.

Now to get the system evaluation we need to test it in a wireless sensor networking set up. Here our goal is to implement the nodes in a wireless sensor networking set up for remote monitoring and controlling water content of soil to increase the productivity of crop.

4.2 FIELD IMPLEMENTATION

- In field implementation we have placed the 4 sensor node in the test-bed consisting of soil bed of 2 inch thickness 1m x 1m area.
- Router is placed at 2m apart from the cluster of sensor nodes i.e. S1, S2, S3 and S4, then the Gateway (Raspberry Pi) again 2m apart from router.
- We have developed static routing through firmware programming.

- We have tested the designed node in a small area since the number of nodes available is 6. Area of implementation is restricted due to communication range the RF module.
- The RF module i.e. XBee Series 1 will fail to communicate after certain distance i.e. around 30m in outdoor.
- The sensitivity of the RF module is -92dBm, but while experimentation it is observed that after -85dBm the packet is getting dropped.
- The flow of water is controlled by the solenoid valve which gets open when the soil moisture is below threshold.
- After taking various readings with different soil moisture condition, threshold for actuation has been decided

4.3 There are different types of Soil Moisture Measurement Technique as follows:

4.3.1 Gravimetric Technique: The classical method of measuring the amount of water in a soil sample is to use the Gravimetric Technique. This method involves taking a volume of soil, accurately weighing it, completely drying it out in an oven, re-weighing the dry sample and calculating soil moisture percentage from the weight loss. This is a time consuming and painstaking procedure.

4.3.2. Radioactive Technique

A method which uses radioactivity is called the Neutron Probe technique. Because of the radioactive transmissions, these instruments are very expensive and measurements need to be taken by qualified personnel. Usually the services of a Neutron Probe company are employed to take soil moisture percentage readings on a weekly basis. Shafts are permanently installed at the measurement site which the Neutron Probes lowered each time the readings are taken. This method is expensive and inflexible. Measurement sites are not easily changed, and readings are infrequent.

4.3.3 Capacitive Technique

There are several instruments which indicate the percentage of water in the soil by measuring its capacitance. These instruments give instantaneous volumetric moisture contents quickly and easily by measuring the dielectric properties of the soil. Probes are inserted into the soil to the required measurement depth and the measurement can either be displayed on a meter or can be recorded using a data logger. However, the dielectric property of the soil not only depends on the amount of water present, but also on the type of soil, it's porosity and its organic content. So for accurate volumetric soil water content readings, each measurement site should be individually calibrated [41].

4.3.4 Conductivity Technique

Generally, soil conductivity decreases with decreasing soil moisture. Resistance or gypsum block sensors measure soil conductivity and are quite inexpensive. However, conductivity of the soil water is different in different soil types (alkaline or acid soils) and can change according to the sprays or fertilisers applied. So resistance block sensors are generally used for trends in soil moisture changes only.

In this project capacitive technique is used to measure the percentage of water in the soil. Take out the probe in air, the received voltage at this condition corresponds to 0% humidity and then dip the probe inside a vessel containing water, the reading corresponds to 100% humidity. Now calibrate accordingly to determine the threshold value.

The reading at 0% humidity corresponds to 4500 mV.

The reading at 100% humidity corresponds to 500mV.

Therefore (4500 – 500) mV = 4000mV is range for calibration.

4000 mV ↓ from highest value (4500) -- 100% ↑ in humidity

40 mV ↓ from highest value -- 1% ↑ in humidity.

Table 4.1 Readings of Soil Moisture Sensor

Nature of Soil	Sensor Value (in mV)	Percentage of humidity
Extremely Dry	3400	(4500-3400)/40 = 22.5%
Dry	2500	(4500-2500)/40 = 50%
Wet	1200	(4500-1200)/40 = 82.5%

The soil surface evaporates due to heat and capillary rise of water into the branches of a crop which this leads to soil moisture deficit (SMD).Every crop has a Critical SMD level, if the soil is allowed to dry out beyond this there will be a resultant decrease in yield or quality.

By measuring soil moisture, crop water needs can be assessed and controlled to keep the SMD within the boundaries of Critical SMD to maximize crop yield and quality.

4.4 Optimization of Beaconing time

The interval of time after which the sensor data is sent to the gateway has great importance in terms of power consumption. Since all sensor nodes are powered by battery so it is highly required to minimize power consumption. If the sensor data is sent frequently, then the battery will dry out sooner. It is very much necessary to choose optimum beacon time according to the application requirement. In this project the sensor data is sent to the gateway via router at an interval of 1 hour since soil will take some time to dry

4.5 Cost Analysis of Proposed WSN model

Table 4.2 Total Cost of proposed WSN model

	Cost of Single Unit (Rs)	Number of unit	Total Cost (Rs)
Gateway	8800	1	8800
Router	2400	1	2400
Sensor Node	2879	4	11516
	Total Cost of WSN		22716

The total cost for designing the prototype for WSN has been shown above. Here we have fabricated 6 nodes including gateway at Rs 22716. Suppose we would have used the SENSEnuts nodes, then the cost of entire WSN would have been Rs 10000 x 6 = Rs 60000/- plus cost of a Personal Computer. Around Rs 90000 it would have cost. It means the cost have been decreased by 75% by fabricating own WSN model. Here I have compared only with SENSEnuts module since its cost is minimum among other available modules.

4.6 Summary

The entire wireless sensor network has been successfully implemented and several experiments are performed to check the result mentioned in Chapter 3. From the observed result the code either corrected or optimized to improve the performance of proposed WSN model.

Chapter 5

Conclusion and Scope of Future Work

5.1 Conclusion

A wireless sensor network is a collection of small wireless devices that communicate with each other and send information about its surroundings to a centralized location. These wireless nodes can be segregated into sensors, which perform the sensing duties, and forward the information from the sensors to its final remote display unit.

The market for wireless sensor networks is expected to grow rapidly in the near future. ZigBee-ready modules may have contributed to this development, since they can easily be connected to various types of sensors and to build up wireless networks. The RF module Zigbee operated at 2.4GHz ISM band really help for secure data transmission and opens possibility of numerous application scenarios.

Sometimes difficulty in procurement process and high cost, it is an issue for researcher to have deployment of wireless sensor network related applications. Lack of design documentation for the nodes would make it difficult to customize. Hence in this paper a step-by-step methodology has been performed to develop a WSN for agriculture application.

Selections of components, integration of these components to develop the prototype, integration of firmware and hardware, field implementation, optimization are the main contribution of this thesis work.

The proposed WSN model can be deployed for other application also such as environment monitoring, smart home controlling, asset tracking etc.

5.2 Scope for Future Work

The advantage of using designed WSN is the flexibility along with good performance, high reliability, low power consumption and low cost over available nodes in the market. To conclude the thesis, the following are some suggestions for the future work which can be done.

- The work conducted in this thesis is considered an initial feasibility test and demonstrate that the system has the ability to remotely monitor the soil moisture and control flow of water in the irrigation field. The system robustness and durability must be checked.
- In this project, we have used static routing strategy. The device (sensor tag/router) has a fixed parent. Each time device will send the data to its parent only. If any device loses this connection then no data will be forwarded through this device. As there is no alternative path data will be lost. If we can develop any dynamic routing strategy then it will improve the network reliability.
- Also further software development for Raspberry Pi is required to store the data in the server.
- Implementation of power minimizing algorithm is required to extend battery life to sufficient levels to prevent additional workload from battery replacement.
- In addition to this various other sensor applications has also been implemented with different sensors. We can integrate other sensors such as atmospheric humidity sensor, nitrogen content sensor, pressure sensor etc.
- Finally, interfacing with other enhanced Zigbee platforms would provide increased communication range.
- It is also envisioned that sensor networks will ultimately be connected to the Internet, through which global information sharing becomes feasible.

- Different types of shortest path and power efficient routing protocols can also be designed, developed and implemented.
- We also need to test the scalability by enhancing the load.

Abbreviations and Acronyms

AC	Alternating Current
ADC	Analog to Digital Converter
API	Application Programming Interface
AT	Attention (Instructions used to control a modem)
DC	Direct Current
DARPA	Defense Advanced Research Projects Agency
DI	Digital Input
DSP	Digital Signal Processor
DIP	Dual In-line Package
DO	Digital Output
DSSS	Direct Sequence Spread Spectrum
DTR	Data Terminal Ready
DR-QFN	Dual Row Quad Flat No-lead
e.g.	exempli gratia (for example)
EIRP	Equivalent Isotropically Radiated Power
EEPROM or E2PROM	Electrically Erasable Programmable Read-Only Memory
FFD	Full-Function Device
GPIO	General Purpose Input / Output
I/O	Input Output
IC	Integrated Circuit
IDE	Integrated Development Environment
IEEE	Institute Of Electrical And Electronics Engineers
ISM	Industrial, Scientific, And Medical
ISP	In-System Programming
LOS	Line Of Sight

MCU	Microcontroller Unit
MIPS	Million Instructions Per Second
MISO	Master IN Slave OUT
MOSI	Master OUT Slave IN
OS	Operating System
PAN	Personal Area Network
PC	Personal Computer
QFN	Quad Flat No-leads
RAM	Random Access Memory
RC	Resistance-Capacitance
RSSI	Received Signal Strength Indication
RFD	Reduced-Function Devices
RF	Radio Frequency
RISC	Reduced Instruction Set Computer
ROM	Read Only Memory
RSSI	Received Signal Strength Indication
RX	Receiver
RxD	Received Data
SCK	Serial Peripheral Interface SPI Clock
SMA	Subminiature Version A Connector
SRAM	Static Random Access Memory
SMD	Surface Mount Device
TTL	Transistor Transistor Logic
TX	Transmitter
TxD	Transmitted Data
UART	Universal Asynchronous Receiver Transmitter
USART	Universal Synchronous Asynchronous Receiver Transmitter
USB	Universal Serial Bus
Wi-Fi	Wireless Fidelity
WSM	Wireless Sensor Mote
WSN	Wireless Sensor Networks
WPAN	Wireless Personal Area Network

References

[1] Goldsmith, Andrea. *Wireless Communications*. Cambridge University Press, 2005.

[2] Abhishek Kaushik, Parvin Kumar Kaushik and Sanjay Sharma. "History Based Contention Window Control Protocol For Energy Efficiency In Wireless Sensor Network." *International Journal of Advance Research In Science And Engineering*, March 2014: Vol. No.3, Issue No.3.

[3] Saluja, Ritu Kadyan and Kamal. "Distributed Energy Efficient Clustering (Deef) In Heterogeneous Wireless Sensor Network." *International Journal of Engineering and Innovative Technology (IJEIT)*, July 2014: Volume 4, Issue 1 .

[4] Lee, Jin-Shyan, Yu-Wei Su, and Chung-Chou Shen. "A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi." In *Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE*, pp. 46-51. IEEE, 2007.

[5] Balasubramanian, Aruna, Ratul Mahajan, and Arun Venkataramani. "Augmenting mobile 3G using WiFi." In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, pp. 209-222. ACM, 2010.

[6] Raniwala, Ashish, and Tzi-cker Chiueh. "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network." In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 3, pp. 2223-2234. IEEE, 2005.

[7] Xiao, Yang. "IEEE 802.11 n: enhancements for higher throughput in wireless LANs." *Wireless Communications, IEEE* 12, no. 6 (2005): 82-91.

[8] http://www.howtogeek.com/204697/wi-fi-security-should-you-use-wpa2-aes-wpa2-tkip-or-both/

[9] http://www.cisco.com/c/en/us/products/collateral/wireless/spectrum-expert-wi-fi/prod_white_paper0900aecd807395a9.pdf

[10] http://www.crutchfield.com/Ses4IE6WgjOa/learn/learningcenter/home/wifi.html

[11] Lee, Jin-Shyan, Yu-Wei Su, and Chung-Chou Shen. "A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi." In *Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE*, pp. 46-51. IEEE, 2007.

[12] Iti Saha Misra. *Wireless Communications and Networks: 3G and Beyond*. McGraw Hill Education (India) Pvt Ltd, 2013.

[13] Ergen, Sinem Coleri. "ZigBee/IEEE 802.15. 4 Summary." *UC Berkeley, September* 10 (2004).

[14] Punitha, R., M. Banu Priya, B. Vijayalakshmi, and C. Ram Kumar."Adoptive Parent Based Framework For Zigbee Cluster Tree Networks." *IJETR. feb* (2014).

[15] http://electronicdesign.com/what-s-difference-between/what-s-difference-between-ieee-802154-and-zigbee-wireless

[16] http://www.cs.berkeley.edu/~prabal/teaching/cs294-11-f05/slides/day21.pdf

[17] Howitt, Ivan, and Jose A. Gutierrez. "IEEE 802.15. 4 low rate-wireless personal area network coexistence issues." In *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, vol. 3, pp. 1481-1486. IEEE, 2003.

[18] Petrova, Marina, Janne Riihijärvi, Petri Mähönen, and Saverio Labella. "Performance study of IEEE 802.15. 4 using measurements and simulations." In *Wireless communications and networking conference, 2006. WCNC 2006. IEEE*, vol. 1, pp. 487-492. IEEE, 2006.

[19] Ran, Peng, Mao-heng Sun, and You-min Zou. "ZigBee routing selection strategy based on data services and energy-balanced ZigBee routing." In *Services Computing, 2006. APSCC'06. IEEE Asia-Pacific Conference on*, pp. 400-404. IEEE, 2006.

[20] Cuomo, Francesca, Sara Della Luna, Ugo Monaco, and F. Melodia. "Routing in ZigBee: benefits from exploiting the IEEE 802.15. 4 association

tree." In *Communications, 2007. ICC'07. IEEE International Conference on*, pp. 3271-3276. IEEE, 2007.

[21] Heidari, Ghobad. *WiMedia UWB: technology of choice for wireless USB and Bluetooth*. John Wiley & Sons, 2008.

[22] Willig, Andreas, Kirsten Matheus, and Adam Wolisz. "Wireless technology in industrial networks." *Proceedings of the IEEE* 93, no. 6 (2005): 1130-1151.

[23] http://www.signal11.us/oss/elce2012/elce2012_wireless_networking_with_ieee802154_and_6lowpan.pdf

[24] Dannies, Alexander, Javier Palafox-Albarrán, and Reiner Jedermann1and Walter Lang. "Feasibility of shifting decision support tools for quality estimation in food logistics to the sensor node level." In *6th International Conference on Software Knowledge Information Management and Applications (SKIMA), Chengdu University*. 2012.

[25] Boano, Carlo Alberto, Hjalmar Wennerstrom, Marco Zuniga, James Brown, Chamath Keppitiyagama, Felix Oppermann, Utz Roedig, Lars-Ake Norden, Thiemo Voigt, and Kay Römer. "Hot Packets: A systematic evaluation of the effect of temperature on low power wireless transceivers." (2013).

[26] Preethichandra, D. M. G. "Design of a smart indoor air quality monitoring wireless sensor network for assisted living." In *Instrumentation and Measurement Technology Conference (I2MTC), 2013 IEEE International*, pp. 1306-1310. IEEE, 2013.

[27] http://www.sensenuts.com/

[28] http://www.bhashatech.com/shields/73-xbee-shield.html

[29] https://www.raspberrypi.org/products/raspberry-pi-2-model-b/

[30] http://www.amazon.in/PI-World-Converter-Adapter-Cable/dp/B01F6YGE7W?tag=googinhydr18418-21&tag=googinkenshoo-21&ascsubtag=c5114fcc-2f6c-4b8c-87df-2f9977bd44c2

[31] http://www.ebay.com/bhp/arduino-power-supply

[32] http://richduino.com/

[33] http://www.amazon.in/embsys-LM35-Temperature-Sensor/dp/B017MHSDMK?tag=googinhydr18418-21&tag=googinkenshoo-21&ascsubtag=c5114fcc-2f6c-4b8c-87df-2f9977bd44c2

[34] http://www.ebay.in/itm/252013166035?aff_source=Sok-Goog

[35] http://electronica4u.blogspot.in/2012/03/how-16x2-alphanumeric-lcd-works.html

[36] http://www.ebay.in/itm/322026268548?aff_source=Sok-Goog

[37] Pekhteryev, Georgiy, Zafer Sahinoglu, Philip Orlik, and Ghulam Bhatti. "Image transmission over IEEE 802.15. 4 and ZigBee networks." In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pp. 3539-3542. IEEE, 2005.

[38] https://www.**digi**.com.my

[39] http://www.ebay.in/itm/252013166035?aff_source=Sok-Goog\

[40] http://electronica4u.blogspot.in/2012/works.html

[41] http://www.ti.com/product