A DESIGN AND APPLICATION OF LOW POWER WIRELESS SENSOR NETWORK

SORAPHOB SUPARCHAIPARNITCHAPONG

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF ENGINEERING
(MECHANICAL ENGINEERING)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2013**

Thesis
entitled

A DESIGN AND APPLICATION OF LOW POWER WIRELESS SENSOR NETWORK

…………………………………………………….
Mr. Soraphob Suparchaiparnitchapong
Candidate

…………………………………………………….
Asst. Prof. Ittichote Chuckpaiwong,
Ph.D.
Major advisor

…………………………………………………….
Assoc. Prof. Chatchai Neatpisarnvanit,
Ph.D.
Co-advisor

…………………………………………….
Asst. Prof. Auemphorn Mutchimwong,
Ph.D.
Acting Dean
Faculty of Graduate Studies
Mahidol University

…………………………………………………….
Asst. Prof. Ittichote Chuckpaiwong,
Ph.D.,
Program Director
Master of Engineering Program in
Mechanical Engineering
Faculty of Engineering,
Mahidol University

Thesis
entitled

A DESIGN AND APPLICATION OF LOW POWER WIRELESS SENSOR NETWORK

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Engineering (Mechanical Engineering)
on
July 31, 2013

……………………………………………
Mr. Soraphob Suparchaiparnitchapong
Candidate

……………………………………………
Mr. Sarawut Ninsawat,
Ph.D.
Chair

……………………………………………
Asst. Prof. Ittichote Chuckpaiwong,
Ph.D.
Member

…………………………………………..
Assoc. Prof. Chatchai Neatpisarnvanit,
Ph.D.
Member

……………………………………
Asst. Prof. Auemphorn Mutchimwong,
Ph.D.
Acting Dean
Faculty of Graduate Studies
Mahidol University

…………………………………………..
Lect. Worawit Israngkul,
M.S., Technical Management
Dean
Faculty of Engineering
Mahidol University

ACKNOWLEDGEMENTS

The success of this thesis will not be able to occur, if do not receive the assistance by the alternative support from my family, my advisor, and committee of Mechanical Engineering Mahidol University.

Above all, I would like to grateful to my parents, Mr. Chaichana Suparchaiparnitchapong and Mrs. Chaowala Suparchaiparnitchapong, who supported me for everything and encouragement. I am grateful to my advisor Asst. Prof. Ittichote Chuckpaiwong for continuous support everything especially solution, technique and English language. Aftermost, I thank to every impetus that give strength to me for passing the badness.

Soraphob Suparchaiparnitchapong

A DESIGN AND APPLICATION OF LOW POWER WIRELESS SENSOR NETWORK

SORAPHOB SUPARCHAIPARNITCHAPONG 5337638 EGMI/M

M.Eng. (MECHANICAL ENGINEERING)

THESIS ADVISORY COMMITTEE: ITTICHOTE CHUCKPAIWONG, Ph.D., CHATCHAI NEATPISARNVANIT, Ph.D.

ABSTRACT

In many scenarios where a set of data loggings must be performed but the wiring cannot be conveniently installed, a wireless sensor network is a good candidate. Each node in this network must consume limited amount of resources, such as microcontroller's processing power, memory, and power supply. The software and hardware design must focus on low-power short-range radio frequency communication circuits, which allow each sensor node to be operated by a small-capacity battery with years of lifetime.

This research describes the design of the wireless sensor network for applications with low sampling rate and non real-time data (sampling rate exceeds one second). The design, was emphasized to increase the chance of successful data transmission, even in the case when some nodes do not function. Furthermore, we also emphasized the system lifetime, by minimizing the power consumption. The user interface is designed to be simple and intuitive. The prototype system can be used to send data to a base station via a wireless network for data display.

The prototype WSN system will be tested for its operation, which can be divided into three parts. (1) Testing for power consumption of a node device, which can be calculated from the node's current and uptime. We found that a processor and a wireless transceiver consume the 10% and 90% of overall power consumption, respectively. The power consumption at various processing frequencies are compared at sleep time of 1 second. The node device at 32 MHz provides the highest lifetime. (2) Testing for the operational range between node devices. We found the operating range for indoors and outdoors were 50 and 100 meters, respectively.

(3) Testing for the performance of the WSN system. The performance consists of a node device's lifetime, successful data transfer rate (packets/s), and ability to reroute the path. In all tests, we set the sleep time of all node devices at 1 second. We found that the device lifetime depends on its location with respect to the whole network. The device lifetime is highest at 60 days and lowest at 11 days when using a battery capacity of 2,450 mAh. The successful data transfer rate is averaged at 2 packets/s. All the node devices in the system can search and change the path automatically when an irregular event occurs. The lifetime of the node device can be increased by increasing the sleep time. If the node device is set to a sleep time of 60 seconds, the node device will have a lifetime of least 676 days

KEY WORDS: WIRELESS SENSOR NETWORK/ MICROCONTROLLER

154 pages

การออกแบบและประยุกต์ใช้เครือข่ายตรวจจับไร้สายแบบพลังงานต่ำ

A DESIGN AND APPLICATION OF LOW POWER WIRELESS SENSOR NETWORK

สรภพ สุภชัยพานิชพงศ์ 5337638 EGMI/M

วศ.ม. (วิศวกรรมเครื่องกล)

คณะกรรมการที่ปรึกษาวิทยานิพนธ์: อิทธิโชติ จักรไพวงศ์, Ph.D., ฉัตรชัย เนตรพิศาลวนิช, Ph.D.

บทคัดย่อ

ในงานที่ต้องการวางเครือข่ายเพื่อเก็บข้อมูลซึ่งเครือข่ายตรวจจับแบบมีสายไม่สะดวกหรือไม่สามารถติดตั้งได้ เครือข่ายตรวจจับแบบไร้สายมีความเหมาะสม และสามารถประยุกต์ใช้ในงานได้หลายประเภท โหนดหรืออุปกรณ์ที่เชื่อมต่อในเครือข่ายแบบนี้มีทรัพยากรอยู่จำกัดมาก ทั้งความสามารถในการประมวล หน่วยความจำ กำลังไฟที่ใช้เลี้ยงวงจร และกำลังส่งสัญญาณ จึงต้องอาศัยการออกแบบเป็นอย่างดีทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ ด้วยวงจรเชื่อมต่อไร้สายที่ใช้พลังงานต่ำ ทำให้อุปกรณ์ตรวจจับแต่ละตัวสามารถทำงานได้เป็นเวลาหลายปีโดยใช้ไฟเลี้ยงจากแบตเตอรี่ขนาดเล็ก

งานวิจัยนี้นำเสนอการออกแบบเครือข่ายตรวจจับไร้สายให้นำไปประยุกต์ใช้กับงานที่มีอัตราการสุ่มข้อมูลต่ำ และไม่จำเป็นต้องใช้ข้อมูลในเวลาจริง (อัตราสุ่มข้อมูล และระยะเวลาในการส่งข้อมูลอยู่ในระดับวินาที) เครือข่ายตรวจจับไร้สายนี้ถูกออกแบบมาเพื่อเพิ่มโอกาสให้ข้อมูลสามารถไปถึงจุดปลายทางสูง แม้ว่าในกรณีที่อุปกรณ์ที่เชื่อมต่อบางตัวจะมีการชำรุดในระหว่างการใช้งาน นอกจากนี้เรายังคำนึงถึงอายุการใช้งานของระบบ โดยการพัฒนาระบบให้กินพลังงานน้อยที่สุดโดยกระบวนการทางฮาร์ดแวร์และซอฟต์แวร์ โครงสร้างจากการออกแบบเครือข่ายไม่ซับซ้อนทำให้ง่ายแก่การใช้งาน ระบบต้นแบบที่สร้างขึ้นสามารถส่งค่าข้อมูลผ่านเครือข่ายไปยังสถานีหลักเพื่อนำไปใช้แสดงผล

ระบบต้นแบบจะถูกทดสอบความสามารถในการทำงานด้านต่างๆสามส่วน ได้แก่ (1) การทดสอบอัตราการใช้พลังงานของอุปกรณ์ ซึ่งคำนวณได้จากการวัดค่ากระแส และเวลาที่อุปกรณ์ทำงาน เราพบว่าหน่วยประมวลผลและอุปกรณ์ส่งข้อมูลไร้สายใช้พลังงานประมาณ 10% และ 90% ตามลำดับ เราทำการเปรียบเทียบอัตราการใช้พลังงานที่ความถี่นาฬิกาต่างๆของหน่วยประมวลผลพบว่าที่ความถี่สูงสุดคือ 32 MHz ให้อายุการใช้งานของอุปกรณ์มากที่สุด เมื่อกำหนดค่าเวลาหลับคงที่ 1 วินาที (2) การทดสอบระยะหวังผลระหว่างอุปกรณ์ เราพบว่าในพื้นที่ในอาคารมีระยะหวังผลต่ำกว่า 50 เมตรลงมา และในพื้นที่โล่งแจ้งมีระยะหวังผลต่ำกว่า 100 เมตรลงมา

(3) การทดสอบประสิทธิภาพของระบบ ด้วยการวัดอายุการใช้งานของอุปกรณ์ อัตราการส่งข้อมูลในหน่วยแพ็กเกจต่อวินาที และความสามารถในการค้นหาเส้นทางใหม่ เราตั้งเวลาในการหลับของแต่ละอุปกรณ์ไว้ที่ 1 วินาที เราพบว่าอายุการใช้งานของอุปกรณ์ขึ้นอยู่กับตำแหน่งการวาง โดยที่มากที่สุดอยู่ที่ 60 วัน และน้อยที่สุดอยู่ที่ 11 วัน สำหรับประจุไฟฟ้า 2,450 mAh อัตราความสำเร็จของข้อมูลเฉลี่ยอยู่ที่ 2 แพ็กเกจต่อวินาที และอุปกรณ์ในระบบสามารถค้นหาและเปลี่ยนเส้นทางเองได้เมื่อมีเหตุการณ์ผิดปกติเกิดขึ้น อายุการใช้งานของอุปกรณ์สามารถเพิ่มโดยเพิ่มเวลาการหลับให้อุปกรณ์ หากอุปกรณ์ถูกตั้งค่าเวลาการหลับที่ 60 วินาที อุปกรณ์จะมีอายุการใช้งานได้อย่างน้อย 676 วัน

154 หน้า

CONTENTS

CONTENTS (cont.)

CONTENTS (cont.)

LIST OF TABLES

LIST OF TABLES (cont.)

LIST OF FIGURES

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

CHAPTER I
INTRODUCTION

1.1 Introduction and Background

Today, impact of wireless communication to human living is rising. This technology responds the increasing demand of people by evolution of high technology. It is used in industry, military, medication, business and others. There are more advantageous points than those of wire communication, such as handiness in sitting, easiness in adding another point on existing system, small size for carrying, and communication on the air. However, there are some disadvantages, such as limit of power source, distance limit in connecting, and low data rate.

The Wireless Sensor Network (WSN) began from the development of technology in sensor, microprocessor and wireless communication. The field leads to new method of data gathering which is a small size with low power consumption and low cost. By distribution of many small sizes, sensor nodes are placed into the intended environment for gathering information. For most WSN, there is only one sensor of a specific type (base on application) on the intended environment. The system is independent network without any communication with the external world. Wireless sensor network is an active area of research and currently being used in a variety of applications. The ultimate and goal of such research is the ability that permeates the physical world with a multitude of invisible, and decreases the sensor maintenance. The most work of the WSN is used for the monitoring or manipulation of the environment.

Applications of the WSN are growing because the method of the WSN reduces cost and energy. The WSN can also increase performance of work efficiency. The possible military applications of wireless sensor network are also tremendous, primarily in the area battlefield monitoring, but the hardware is not yet to the point where many of them can be realized. But so far, wireless sensor network have been used by the military for such thing as tracking vehicles [16] and locating snipers [18].

The medical industry is already using wireless sensor to monitor vital signs in hospitals [20], and one can envision sensors being used for constant monitoring of vitals and other health parameters in the elderly. WSN can also be used for building and home automation, such as to increase energy efficiency by automatically turning off lights when a room is empty, or for real-time monitoring of power consumption. In 2007, B. Zhou et al. [9] developed the wireless sensor network to collected and reported bio-information and environment parameters when emergency situation occurred for a pervasive medical supervision. In 2008, Y. Kim et al. [23] designed and developed the wireless sensor network to control an irrigation system. In 2010, M. Unhawiwat et al. [17] developed the wireless sensor network to detected information of environment for increase efficiency in agriculture.

The above data suggest that many people tried to develop the wireless sensor network to use in many high performance applications. But most of those WSN are designed by more complex software that increases power consumption. Devices in the WSN have unnecessarily options and specifications for some work, which causes high power consumption and high cost. Furthermore, setting up WSN systems is a difficult. From those suggestions, this research set a goal to develop the WSN to meet the following requirement:

- Power consumption in operation of sensor node should be reduced for long life.
- Unnecessarily options of sensor node should be removed and made for the most general work.
- Complexity of algorithm of sensor node should be reduced. The system can also continue operation while some algorithm was reduced. Furthermore, we can increase performance of the operation by short algorithm.
- Setting of WSN system can be made easily by anyone.

From those of design criteria, we designed and developed sensor node called Node Device which requires low power consumption electronics and less complex algorithm. The node device also has high efficiency, supports low power consumption, and automatically reroutes data when problem arises. The node device will still be able to work when some nearby node devices stop working, the node will send data to another node in its area. This is a part in Dynamic Routing. All nodes can sleep because they use the same algorithm, and they are also easy in the node setting. This thesis provides details about the design of node device and the algorithm of node device. Our node devices can work together to form a network in a field work.

1.2 Research Objective

Many existing hardwares or Node Devices in Wireless Sensor Network cannot work at low power consumption and have many unnecessarily options. The algorithm is also too complex for some work and cannot be used in some situations. From such problem, this research objective is to develop a new WSN system, which includes the following details.

1.2.1 Creating and developing the node device in a wireless sensor network, which can collects and gathers data in the network. We can also bring those data to use in other applications.

1.2.2 Developing the algorithm of the node device that can operate at low power consumption and complexity.

1.2.3 Developing the algorithm of the node device that can reroute data when it cannot connect to other nodes, which maybe not function.

1.2.4 Developing the algorithm of the node device that can be easily set WSN parameter via wire and wireless method.

1.3 Methodology

This research creates and develops the hardware and the software of the node device to work in the WSN system and achieves research objectives. In order to make a complete WSN, we also developed hardware of host device, and software of host device. The following tasks were performed.

1.3.1 Creating simple node devices for connections and collecting analog information. We will use an analog signal in data collection of sensing unit, which allows wide range of use. Each node device uses one analog sensor, which can be changed to any analog sensors for general use.

1.3.2 Programming to node device for running.

1.3.3 Developing hardware and software of the node device for low power consumption. It can also make in connection among node devices and collect sensor data.

1.3.4 Developing algorithm for network connection.

1.3.5 Adding the routing protocol in the algorithm for solving many problems (such as the route in communication must be able to change when old route cannot be used) in this system.

1.3.6 Adding the method of WSN parameter setting in the WSN system. The method of WSN parameter setting includes wire and wireless setting method.

1.3.7 Testing prototypes for analog data collection and network connection in surrounding (real area on outdoor and indoor) by checking on the RS-232 monitor for analysis.

1.3.8 Using ET-NXP-ARM-KIT-LPC1768 for the host device and adding some hardware in the work.

1.3.9 Programming to the host device. Working of the host device includes controlling of display touch screen for system setting, connecting to the SD card for data logger, connecting with the node device for setting basic values operation in the WSN and information received, and connecting with the TCP/IP for server.

1.3.10 Testing all devices together for analysis and making necessary modification.

CHAPTER II
LITERATURE REVIEW

2.1 Literature Review

In this work, we review the existing Wireless Sensor Network (WSN), especially those designed to save power and withstand damages. From literature review, most of WSN can save power by sleep mode and endure to damages by routing protocol. We take useful point in both techniques of saving energy and routing protocol to our system design for the best result. The literature review is divided into three parts. The first part is applications of a wireless sensor network. The second part is designs of sensor nodes. The third part is routing protocols of wireless sensor networks.

2.1.1 Applications of Wireless Sensor Network

Wireless sensor networks are used with many works for transferring and controlling data. In 2005, T. Gao et al. [20] used the WSN for vital signs monitoring and patient tracking. This system has great potential in improving problems in today's emergency response system, especially in plans to deal with mass casualty disasters. But it has limitations, and cannot provide their benefits under all circumstances. And in 2005, J. Brassard and W. Siu [12] used the WSN for sniper detection. This work use wireless nodes to identify that a rifle shot has taken place and to track the sound and shockwave created by the shot in order to work backward until the location of the sniper is found. In 2007, B. Zhou et al. [9] used the WSN for pervasive medical supervision. This paper presented a pervasive medical supervision system based on the wireless sensor network on the acceptable cost and application feasibility. Except for bio information such as the oximetry, the heart rate/pulse and the blood pressure, the system provides some associated context data, e.g., the environment temperature, and the patient's video/picture in emergency situation. In 2007, P. Kuckertz et al. [18] used the WSN for sniper fire localization. This work is to resemble as work of sniper detection in 2005. Both technology and performance of the WSN, which are high accuracy, small, and endurance, are used in military. In 2008, Y. Kim et al. [23] used the WSN for remote sensing and controlling of an irrigation system. This work discussed the design and instrumentation of variable rate irrigation, a wireless sensor network, and software for real-time in-field sensing and control of a site-specific precision linear-move irrigation system. In 2009, K. G. Aravind et al. [16] used the WSN for vehicle tracking system. Most state-of the-art technology uses GPS (Global Positioning System) for tracking vehicles which is very expensive. The work use embedded wireless sensor network to track vehicle with low cost, effective implementation as in contrast to the existing high cost tracking with the GPS. In 2010, M. Unhawiwat et al. [17] used the WSN for rubber orchard monitoring and alerting systems. This work use TmoteSky (Sensor mote module) device and nesC language in TinyOS for software. But the node cannot automatically choose the way to send data. And in 2010, V. Boonsawat et al. [21] used the WSN for temperature monitoring. This work is used for management of air conditioning systems. This WSN is to help saving the energy cost and reduce energy consumption. In 2011, D. H. Park, and J. W. Park [10] used the WSN for greenhouse environment monitoring and automatic control system for dew condensation prevention. This work is to prevent dew condensation in a greenhouse environment. The dew condensation on the leaf surface of greenhouse crops may promote diseases to the crops.

The communication is included in the technology for improving performance. The wireless sensor network is a growing technology used widely. It replaces old centralized communication technology and often used in some specific work (such as military work). In above existing works, they used the WSN in many applications by many methods. In most applications, the devices were made specifically for their applications, but they are usually high cost and may be not applicable for other applications. In our work, we will design device for reducing cost, reducing energy consumption, and use widely application.

2.1.2 Designs of Sensor Node

Sensor node design is an important part. The life time is one issue of design that must be considered. Most importantly, the nodes must be self-contained in terms of energy via a one-time battery charge or a replenishable supply of energy scavenged from the environment. Due to worthiness in use with any work, the energy source must be designed suitably. In addition to energy source design, the node design can be focused in many parts for improving performance. That design includes the miniaturized size for carrying, the controlled power for increasing life time, many sensing units in one body for keeping many data (may be just use one sensor, but the more sensing unit can support on widely applications, which will have problem in power consumption), and the ease to use. In 2002, J. M. Rabaey et al. [13] designed the sensor node by using Pico Radio. The RF is data rate lower than 1 kb/s and consumes power about 100 uW. Since this RF uses low power, the distance in communication is available about 10 m. In this system network, the multi-hop will be used for expansion of distance. The energy source uses the 1 cm^3 non-rechargeable Li-Fe battery that can feed the energy to continuous up to 6 month as shown in figure 2.1. This graph is useful for our design.

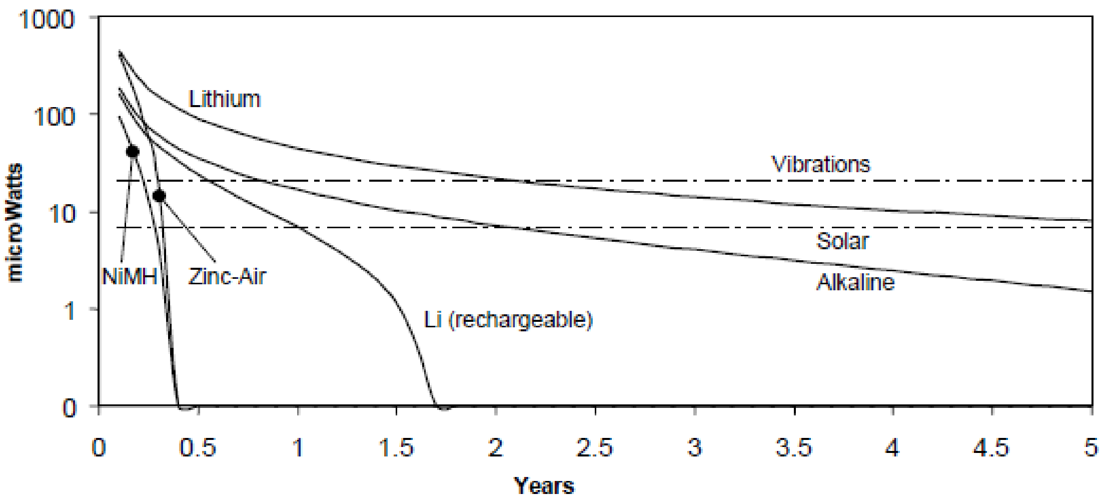


Figure 2.1 Battery lifetime, compare to continuous power source [13]

In 2005, B. Lo et al. [8] designed the Body Sensor Network (BSN) for using in monitoring and analyzing patients who had heart attacks. The BSN uses the MSP430 that is an ultra low power microcontroller in MSP family and sends data via wireless in data rate at 250 kb/s. The RF module use chip CC2420 that use the standard wireless communication IEEE 802.15.4. This design has a small prototype at about 2.6 mm, uses low power, uses standard wireless communication by TinyOS, which is easy to develop and use because this standard is open source. But this BSN uses external flash memory that may consume too much power and it also has no method in controlling power from energy source to various parts in the BSN. In 2009, S. S. Sonavane et al. [19] designed the sensor node in concept of low cost and low power. This node uses the MSP430 for processing and the nRF24L01 for transferring data via wireless. The both of the MSP430 and the nRF24L01 have low power consumption. This work uses the controlling of power RF for reducing power consumption. This approach is interesting for our development. In 2010, W. Waiyawoot [22] designed a low rate of the wireless sensor network coordinator with multi-network use. This work uses one processing that is a microcontroller in Propeller family (by parrallax) to control overall system. The ZigBee was used to connect with sensor and to transfer data. It does not need microcontroller in controlling. This work is one interesting technique in the design WSN for simple system and more sensors use. But the power consumption of the ZigBee is rather high.

The existing designs above have many techniques that we can use and further develop to improve performance of WSNs. The useful points will be adopted in our work develop. When the hardware design is high performance, the hardware must also be controlled by efficient software in order to work well. The software is the most important in the design because everything will be managed by it.

2.1.3 Routing Protocols of Wireless Sensor Networks

Wireless sensor networks consist of small nodes with sensing and wireless communications capabilities. Typically, WSNs contain hundreds or thousands of these sensor nodes, and these sensors have the ability to communicate either among each other or directly to an external base station. Since those communications have many paths, the network might have high traffic which degrades the system performance. Routing protocol is the software for controlling and managing path of wireless communication including the power allocation in nodes. Many routing, power management, and data dissemination protocols have been specifically designed for WSNs where energy awareness is an essential design issue. Routing protocols in WSNs might differ depending on the application and network architecture. In 2004, J. N. Al-Karaki, and A. E. Kamal [14] collected the routing protocols of WSN and compare them. In 2013, E. M. Kaur, and B. S. Sohi [11] reviewed the routing protocol of WSN in useful point and concept.

The routing protocol has many techniques in use depending on application and network architecture according to above. These are the good guideline for designing our routing protocol and other software parts that are yet ineffective.

2.2 Related Theory

In this section, we will describe related theory to design sensor node and topology of a wireless sensor network. It can be explained as follows.

2.2.1 Wireless Sensor Network

Architecture of Wireless Sensor Network consists of sensor node, gateway and base station. Many sensor nodes are posed into surroundings for gathering information. The information is physical data, such as the temperature, the humidity and etc. Each sensor node will transfer the data via wireless signal with the nearest neighbor node. Performance of the connection depends on performance of the wireless signal of each sensor node. Each sensor node will control and operate independently (stand alone). All sensor nodes transfer data among them as a network. They do not only communicate the data once, but they can communicate the data more than one time via the nearest neighbor and forward the data until these data arrive to base station or host. This communication by forwarding is called multi-hop. Gateway is responsible to connect between sensor nodes and base station. The gateway may be a sensor node or special node that depends on situations. Base station gathers the measured data form sensor nodes and controls the system by interfacing with user.

The network is a large communication system. It is used to define pattern of data communication. If we can make high performance communication, the system will have high performance too. The network may change the structure sometimes for balancing situation because of power limit of each sensor node and other. Changes of sensor nodes affect the system. One of the changes is the data transmission route of each sensor node. This route is important in the network and affects the system. We can control the route as we can see in next chapter.

2.2.2 Sensor Node

A sensor node is a combination of communication device and sensor device. It consists of a processing unit, a transfer unit, a power unit and a sensing unit. It has a duty to measure and to collect the data from surroundings. It will take this data to analyze (analyzing and calculation for deciding) and send to other node until the data arrives the destination node or base station for future use.

1) Sensing unit measures the value form surrounding by the sensor, such as the temperature, the humidity, intensive of the light, the acceleration and etc.

2) Transfer unit transfers the data via wireless in Industrial Scientific and Medical (ISM) band for transferring the data between sensor nodes.

3) Processing unit controls all units for operating all tasks, such as the measured data from surrounding. The data will be processed and stored in the memory of this unit or external memory. Then this unit will control the transfer unit to connect with other nodes. Furthermore, this unit determines routes in the network for controlling the data traffic between nodes.

4) Power unit collects the energy and feeds to all units.

2.2.3 Protocol Standard in Wireless Sensor Network

Those protocols are used in the specific network standard group. They are used to develop for researching and working. The sample protocol standard in WSN can be explained as below.

1) The ZigBee [7] is a specification for a suite of high level communication protocols using small, low-power digital-radio base on an IEEE 802.15.4 [2] standard for personal area networks and developed by ZigBee Alliance. ZigBee device are often used in mesh network form to transmit data over longer distances, passing data through intermediate devices to reach more distant ones. It features frequency at 2.4GHz, high data rate at 250kbit/s and operation range in 10-75 meter.

2) IEEE 1451 [1] is a set of smart transducer interface standards developed by the Institute of Electrical and Electronics Engineers (IEEE) Instrumentation and Measurement Society’s Sensor Technology Technical Committee that describe of

open, common, network-independent communication interfaces for connecting transducers to microprocessor, instrumentation systems, and control filed networks.

From the above, those protocols are too complex for this research and cannot perform some functions, such as in case of having more length of data packet than a necessity in useful.

2.2.4 Topology of Wireless Sensor Network

A topology in a wireless sensor network is the pattern that sensor nodes are working together. Design a topology, we must analyze to suite with purpose of work. Since, restrictions in WSN, the network topology must depend on purpose of work. Such as in emergency system, the network topology needs high speed data transmission and will send data when the only emergency occurred for energy saving. In control system of agriculture, the topology network does not need high speed data transmission, but it will send data all the time. Since, it must be used all time, the system must be able to sleep (no action) when no action is required. Topics in analysis of topology network can be explained as follows.

1) Efficient energy use. Life time of the WSN system depends on the energy in sensor node.

2) Accuracy of the data in transmission. The WSN system should provide the most accuracy of data, which depends on accuracy of sensor and processing data in both of sensor node and network.

3) Resistance to damages. In case of that a sensor node cannot work, the WSN must change the structure for continuing work, such as using secondary path in communication with another node for transferring data.

4) Performance of data transmission. It is divided into the performance of data transmission in sensor node and the performance of data transmission in gateway. In the case of asynchronous data transmission, performance of data transmission in sensor node is necessity. But in the case of many synchronous data transmission, performance of data transmission in gateway is necessity.

5) Number of sensor node. It is a main determinant in calculation cost and worthiness of WSN.

WSN are typically organized in one of three types of network topologies. In a star topology, each node connects directly to a gateway. In a cluster tree network, each node connects to a node higher in the tree and than to the gateway, and data is routed form the lowest node on the tree to the gateway. Finally, to offer increased reliability, mesh networks feature nodes that can connect to multiple nodes in the system and pass data through the most reliable path available. This mesh link is often referred to as a router. We can show topologies in figure 2.2.

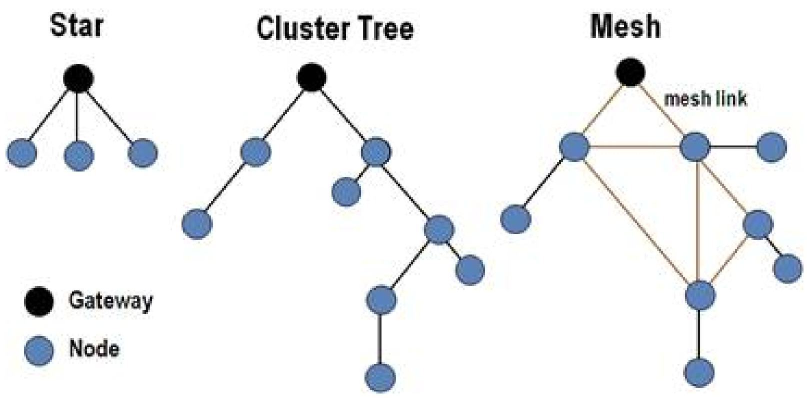


Figure 2.2 Common WSN network topologies

Using work of wireless sensor network is divided into two parts from purposes of work, which can be explained as follows.

1) Querying. The WSN system will send the data request to other sensor node in the network. If a sensor node is a ready, that node will send data to the system via the WSN. Such as in control system of agriculture, if the system sends the request of humidity and temperature value via WSN, the available sensor node will send those data from its memory to the system via WSN.

2) Tasking. In the system that is responsible to check irregular situation in surrounding area, the sensor node will send the data to the system via the WSN when an irregular situation occurred. Such as emergency system in conflagration protecting, the sensor node will send the emergency data to system via WSN when a sensor node detects overheat in surrounding area.

2.2.5 Microcontroller [3]

A microcontroller (MCU) is a small computer on a single integrated circuit containing a processor core, memory and programmable input/output peripherals. Microcontrollers are designed for embedded applications, in contrast to microprocessors used in personal computers or other general purpose applications. Microcontroller created by many vendors in many structures, such as ARM core processor (many vendors), AVR at Atmel, MCS-51 at Intel. Architecture of microcontroller is operation system that includes 8-bit, 16-bit and 32-bit.

2.2.6 RF Transceiver [4]

A transceiver is a device to contain a transmitter and a receiver. The both thing are combined and share common circuitry. The RF or radio frequency is a radio technology or wireless communication. The RF Transceiver uses RF modules or a micro electronic chip for high speed data transmission. RF works on ISM band for short range wireless transceiver.

2.2.7 Serial Peripheral Interface Bus (SPI) [5]

The Serial Peripheral Interface Bus is a synchronous serial data link standard. That bus operates in full duplex mode. Many devices communicate in master/slave mode when the master device initiates the data frame. Multiple slave devices are allowed with individual slave select lines. SPI bus can also operate between a single master device and one or more slave devices. In application, SPI bus is used in communication data of other devices, such as MCU, sensor, Ethernet, USB, USART, CAN, IEEE 802.15.4, IEEE 802.11, flash or EEPROM memory, MMC or SD card, and RF transceiver.

2.2.8 Internet Protocol Suite (TCP/IP) [6]

The internet protocol suite is the set of communications protocols used for the internet and similar networks, and generally the most popular protocol stack for wide area networks. It is commonly know as TCP/IP.

CHAPTER III
SYSTEM DESIGN

A design of the Wireless Sensor Network (WSN) developed in this work consists of the design of Node Device (ND), the network design with the algorithms for the node device, and the design of a Host Device (HD). This chapter will cover the manufacture of the hardware prototype and the network design by the software. Testing about the hardware and the software will be explained in the next chapter.

3.1 Wireless Sensor Network System Overview

The goal of WSN system is to collect information from the intended environment. The example of information, such as information of garden maintenance system that includes temperature, humidity, and light, will be acknowledged for analysis and controlling its system. Most of existing of WSN work, which has the same goal as above, has many techniques and many methods for the best system. The most of those techniques and methods are emphasized in energy reduction and lifetime, and so as this work.

The composition of WSN system is divided into device, algorithm, and data communication. The device is used for collecting, gathering, and information acknowledgement. The algorithm is used for controlling that device. The WSN, which collects many physical data, such as temperature, humidity, force, and other for the benefit in many works, is one method of communication. The physical data will be collected from any areas to transfer to other node devices. The data communication is used in that transfer.

Device of our WSN system is divided into node device and host device according to task assigned to each device. The node device will be used in data collecting and data gathering to host device. The host device is used for data gathering from the node device and information acknowledgement by showing that information via a monitor. Node device is also divided into two parts as follows.

- Sensor Node (SN). It is used to collect the physical data from sensor and send this data to the nearest neighbor via a wireless. Sensor will be installed into sensor node.
- Gateway Node (GN). It is used to collect all data from each sensor node via a wireless and send to a host device via a wire. Sensor will not be installed into gateway node.

Data communication is a process in programming. The algorithm will control data flow between devices of WSN system. If the data communication has more than one connection in system, all of data communication will be called the network. If there are many networks, it becomes a complex system. The complexity will make large amount of data flow, low available rest time access of system, and then those thing will make the system slow and use much energy. For this reason, we must allocate those data flow to minimize energy use and increase communication efficiency.

Basic use of WSN system is easy as shown in figure 3.1. The first step, a user must setup initial value to all devices including node device and host device, and its task assigned will be determined by this setting. The second step is installing those ready devices into intended environment. Final step is to run all devices for work. The intended information will be shown in monitor. The basic use will be described in extended details in topic 3.6.2.

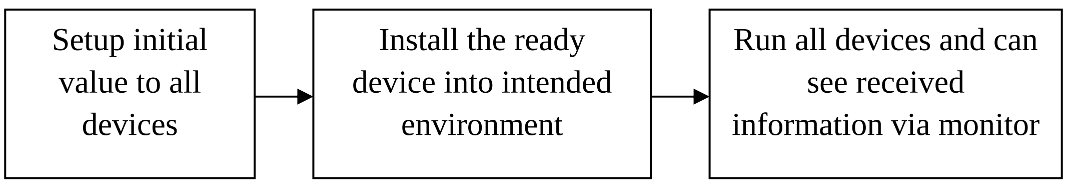


Figure 3.1 Basic use of wireless sensor network system

We can show example project of WSN system on how its system works, why we select this system, and what we receive. Example system is shown in figure 3.2. Detail of this system includes project problem, and project method.

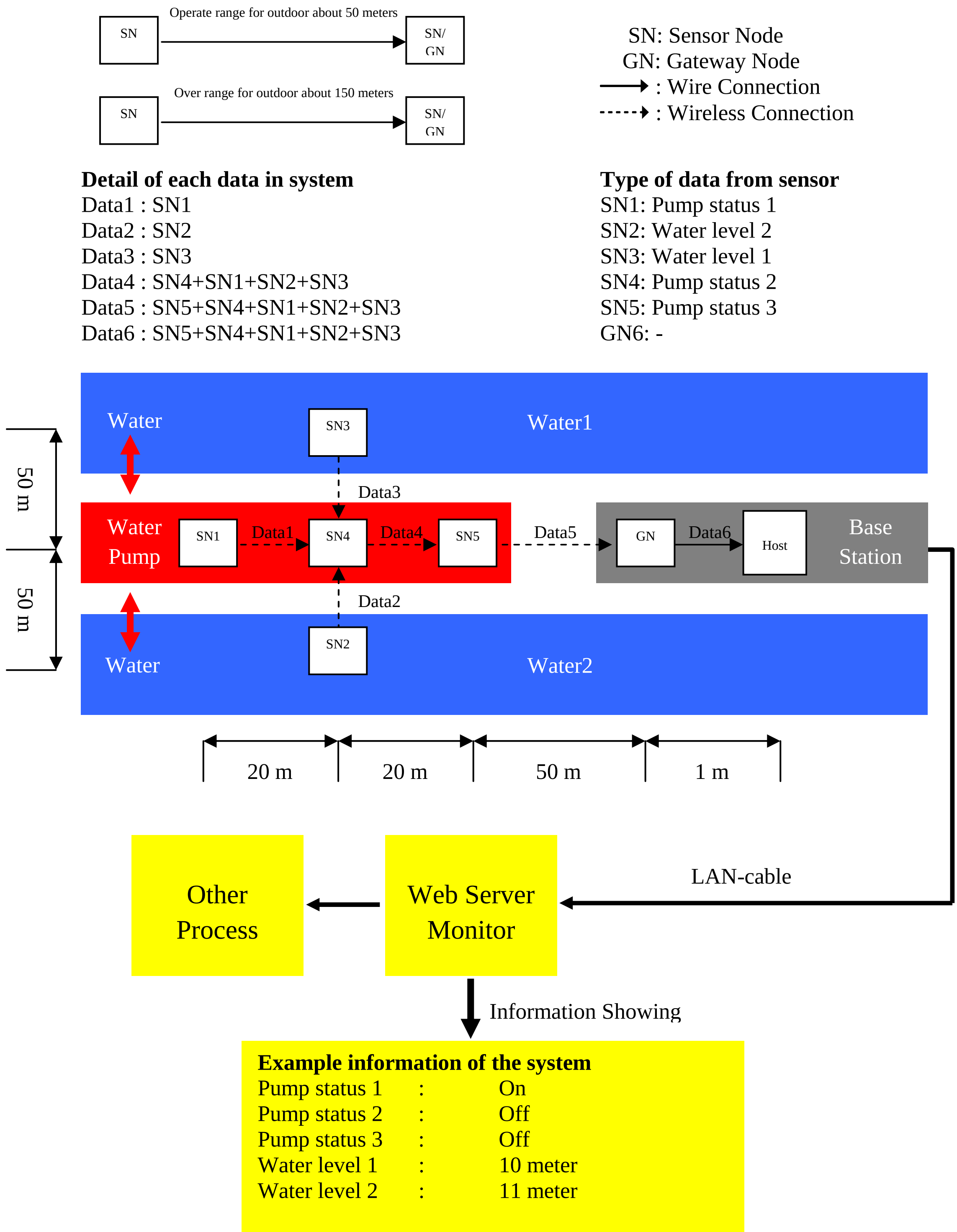


Figure 3.2 An example project of wireless sensor network system

Problem of this project example is requirement in acknowledgement of water level 1, 2 and pump status 1, 2, 3. This problem will be used for checking water transporting rate and pump status. The checking, which must be done continuously for balancing to water transporting rate, is a labor-intensive routine and we use automatic system in checking for human and time resource reduction. Existing automatic systems are divided into two types that are wire and wireless communication. Since wireless connection is easy to installing to any areas, flexible to maintenance, less cost, and easy to add new devices to old system, we will use WSN system.

Method of this project example will take WSN system to solve. Project example has three pumps that distance between them is 20 meters, and two areas of water level that their distance from pump is 50 meters. System consists of following basic setup of WSN system. We use six node devices and one host device. Those node devices, which already have task assigned, include five SNs and one GN. Host device will be installed in base station, and can be online to internet by LAN-cable. Five SNs are used for data collecting in intended area that shown in figure 3.2. Three SNs are installed with three pumps and two SNs for water level checking in the water. One GN is installed with the host device for data gathering from those SNs and data sending to host device by wire. The wireless communication inside system is shown in details of each data. The objective of the system is to show the information on the monitor. That information can be used in other process for system development.

The most important design of WSN system is data communication that will make a lot of loads to the system if we are not careful in controlling. That load will destroy the suitability of the system until the system cannot do task, and make over energy usage that has effect to lifetime of the system.

Our WSN system suits with applications in the medium speed of data transfer, the medium data response time when system queries information from any area, and the large amounts of data. Network topology, which is used in this research, includes Star topology, Cluster Tree topology, and Mesh topology. Applications in this WSN system is the Querying type which transfers data all time, such as environment monitoring, control system of agriculture, home automation monitoring, parameter analysis, and etc. Since the WSN must work all the time and support the large amounts of data, it is important to allocate and limit power consumption of the various components in the WSN for the appropriateness and the most efficiency in the work.

Our special feature of our WSN system is Dynamic Routing that can see extended details in topic 3.3.5. It is a program that is designed by us for risking reduction of data loss. One special feature of Dynamic Routing is automatic rerouting to other node when a device cannot send data or is removed from old point. In addition to Dynamic Routing, we also have data encryption for responding reduction or loading reduction of data flow that will make system to respond quickly.

3.2 Hardware Design of Node Device

The node device (ND) is a unit in a WSN. It collects the physical data by a sensor. Type of the physical data in collecting is 10-bits analog. This data will be sent to the nearest ND via wireless. In the process, each ND will collect one analog data into the its memory and finish the process when the data are sent to a destination or the HD. Algorithm of the ND is divided into the SN and the GN by task assigned in the WSN, which will be explained in extended details in topic 3.3.6. The ND is designed to be stand alone and low power consumption. The energy source comes from a battery and does not have the energy harvesting. Since the power is a necessity, the hardware and the software must be allocated to suit with the purpose of the work and the power consumption.

In a wireless communications, the ND is designed to use the radio frequency at 2.4 GHz to avoid noises in the environment and buildings, such as electricity at 50/60 Hz, working of engines at normal frequency, bad weathers and communication of the mobile phone at 900-1,800 MHz. Other than the above, the communication in 2.4 GHz frequency can be interrupted by noise. This problem can be solved by the process of the software.

One interesting design issue for the power supply is to choose between charging and non-charging features. We created both types and test them. Finally, we chose the non-charging because it is easy for maintenance, uses the low cost, and reduces chance of the damage to the system. Furthermore, we are interested in the super-capacitor and try to design in the power supply part, but we just use one source power which cannot keep the free energy into super-capacitor. The super-capacitor is also increase the load in the design. From these reasons, we do not use a super-capacitor, although we will not use it, but it has potential for the design and the development in the future. Structure of a ND unit can be shown in figure 3.3.

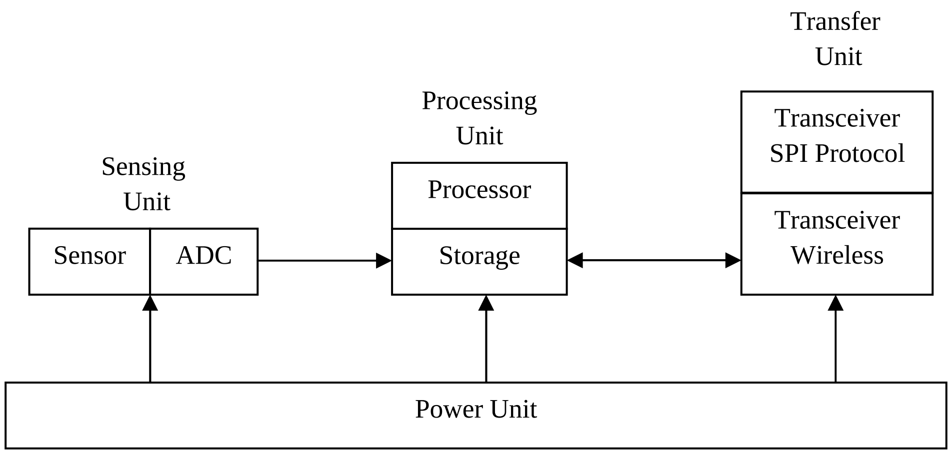


Figure 3.3 Unit of node device

1) Processing Unit. A microcontroller (MCU) PIC18F24J11 is used to be a processor and memory storage in the system. This unit will control the working of a transfer unit and a sensing unit. It also controls the power usage. This part will work all the time and cannot stop the work. The energy will be fed to the MCU all the time but the system can save the energy by setting into the deep sleep mode. Anyway, the power consumption of the MCU uses much less than other units.

2) Transfer Unit. A transceiver module TRW2.4G is used to be a convertor in converting of wireless signal to the digital. This unit will convert the signal from a processing unit to the wireless signal, and receive the wireless signal from other device for converting that signal to the digital information. This unit uses the most power consumption in the system. From that reason, we need to limit the energy using in this unit. The most energy saving from this unit will depend on the handling of uptime by the algorithm that can be seen in topic 3.4. A communication between microcontrollers uses the SPI protocol for the data connection.

3) Sensing Unit. A sensor is used to send analog values. This unit will collect the physical data and send them to a processing unit. For a sensor device, we will use the analog type or adapt it to analog for our design.

4) Power Unit. Two AA size batteries are used to be a source of energy. Voltage must not fall below 2.7 V and over 3.6 V because if voltage is over, damage will occur on system. Alternative to two AA size batteries for a source power, any other size can be used if their capacity is more than 1,000 mAh.

3.2.1 Microcontroller PIC18F24J11

A PIC18F24J11 is a MCU in the PIC family. Architecture of PIC18F24J11 is 8 bit. It features 10 channels of 10-bit Analog to Digital (A/D) converter, program memory 28 kbytes, SRAM 3776 bytes and a power saving mode. Power saving mode of PIC18F24J11 enhances the performance with nano-watt technology or Extreme Low Power (XLP) is called deep sleep mode (feature of PIC18F24J11). The deep sleep mode saves the power more than normal mode or sleep mode. In the deep sleep mode, the current is down to 13 nA. But in the sleep mode, the current is down to 105 nA. Since the high performance of extreme low power in the deep sleep mode, the power is more savable than many other devices, PIC18F24J11 makes sense in this work. Furthermore, ability in controlling other peripheral is the better than others, timer is high accuracy, and endurance for running and programming. But it also has disadvantage, one of that is small size of RAM memory which store data less than other MCUs. In above case, we accept that it is enough for our design. And last thing which is not our problem, the PIC processes slower than others such as AVR and ARM family about four times of Million Instructions per Sec (MIPS). We will compare features and power consumption of some states between many MCU with PIC18F24J11 (all data are taken from data sheet), as shown as follows.

Table 3.1 Comparison of features and the power consumption between microcontrollers

Device	Architecture	Power Consumption			
		Sleep (nA)	WDT (nA)	RTC (nA)	1MHz Run (uA)
PIC18F24J11	8 bit XLP	13	813	813	272
PIC16LF182X	8 bit XLP	20	300	600	50
PIC16LF72X	8 bit XLP	20	500	600	110
PIC16LF193X	8 bit XLP	60	500	600	150
PIC18LF1XK50	8 bit XLP	24	450	790	170
PIC18LF14K22	8 bit XLP	34	460	650	150
PIC18LF4XK22	8 bit XLP	50	600	500	250
PIC18F87K90	8 bit XLP	25	350	720	181
PIC24F04KA201	16 bit XLP	20	370	470	195
PIC24F16KA102	16 bit XLP	20	420	520	195
PIC24FJ64GB004	16 bit XLP	20	220	520	250
MSP430F2001	16 bit XLP	100	600	900	600
MSP430F2619	16 bit XLP	200	600	1,100	515

*All numbers are typical values at minimum VDD, taken from data sheet.

3.2.2 TRW2.4G Transceiver Modules

This module has a function to convert the digital signal to the wireless signal at frequency of 2.4 GHz and supports up to 125 channels. Communication with this module is easy. It can transmit and receive the data with high data rate at 1 Mbit/s, has the CRC system for encryption data, and also has high accuracy in data transfer. This module uses a NRF2401 chip to be a processor, which is low power consumption, has a Shock-Burst mode in reduction of the current consumption time, and is low price. This module will be explained to extended details as topic 3.3.1 and 3.3.2. In a wireless communications, a half-duplex mode device, which can transmit and receive data in one module, is called transceiver. Since this module is the lowest power consumption, provides high data accuracy, uses frequency at 2.4 GHz for noise reduction, and does not use standard protocol, we decided to use TRW2.4G in this work. We will compare features and power consumption between other transceiver modules with TRW2.4G that does not use Shock-Burst mode (this mode will yield even lower current). It can be shown as follows.

Table 3.2 Comparison of features and the power consumption between transceiver modules

Device	Protocol	Max Data Rate (Kbit/s)	Max Range (m)	Power Consumption		
				Standby (uA)	Transmit (mA)	Receive (mA)
TRW2.4G	-	1,000	280	1	18	19
ZigBee	IEEE802.15.4	250	100	10	45	50
MRF24J40MA	IEEE802.15.4	250	130	2	23	19

*All devices are typical values at 3V VDD and max data rate, taken from data sheet.

3.2.3 Bipolar Junction Transistor

The transistor is electronic that is created by the semiconductor. It has a function to amplify and switch the signal. Type of transistor is divided into NPN and PNP from the types of semiconductor. We choose the NPN transistor to control current in our design, and use UCT8058S that carries current up to 700 mA in collector current path that it is sufficient for our design. It is used to control power of the TRW2.4G module and the sensor. The PIC18F24J11 will control this transistor to switch the power to ON/OFF. In the power usage, the PIC18F24J11 will use the power about 1-10 percent of all, the TRW2.4G and the sensor will use the power about 90-99 percent of all as shown in figure 3.4.

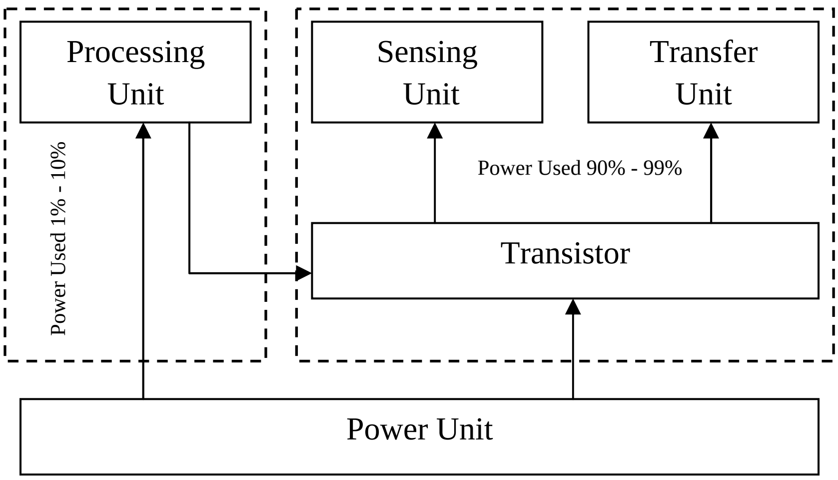


Figure 3.4 Power consumption of node device

3.2.4 Schematic of Node Device

The schematic design is emphasized on simplicity, functions as SN and GN, small size (2.5cm×4.2cm) for setting or carrying, no unnecessarily hardware, and easy maintenance. It is divided into two parts. The first part is schematic of the programmable microcontroller, and the second part is schematic of the node device, which can be explained as follows.

In the schematic of the programmable microcontroller, we will describe how to program PIC18F24J11 with PICkit2, which is a programmer for microcontroller as shown in figure 3.5.

Figure 3.5 PICkit2 programmer device of microcontroller

There are several ways of programming the MCU using BASIC, C or Assembly language, but we used CCS complier version 4.119 C language in programming. Below is a diagram showing the pin-outs of PIC18F24J11, whose programming pins can be explained as follows.

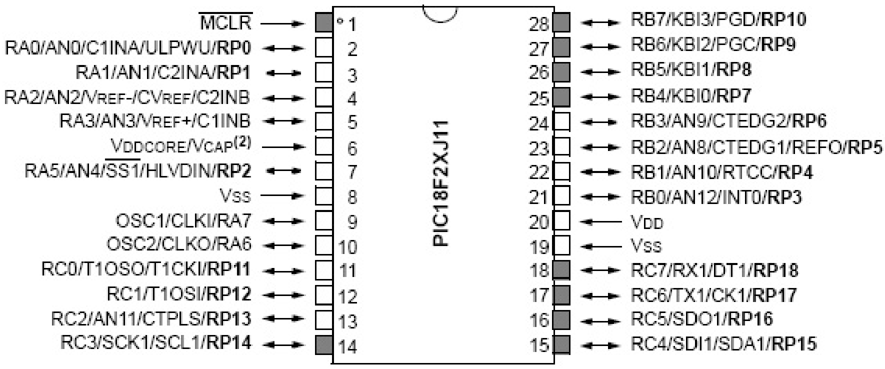


Figure 3.6 PIC1824J11 pins

VDD and VSS are the power supply pins. VDD is positive supply, and VSS is negative supply or GND. MCLR pin is used to erase the memory locations inside the MCU. PGC and PGD Pins are the program pins used for burning programs. All these pins must connect with same pins at ICSP of PICkit2 for burning program by PICkit2 software. We can show schematic as below.

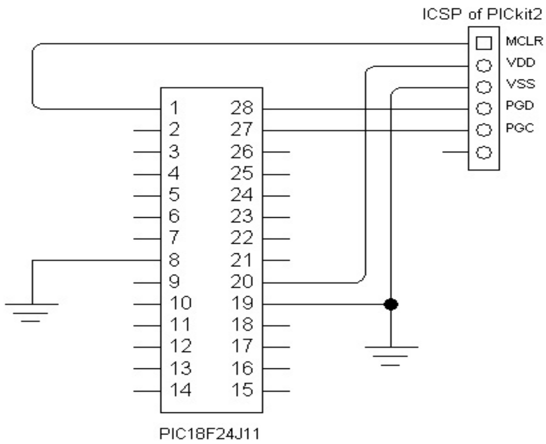


Figure 3.7 Schematic of programmable PIC18F24J11 with PICkit2

In the schematic of device operation, as shown in figure 3.12, there are component of microcontroller circuit, transceiver module, and transistor. In microcontroller circuit, we use 20 MHz crystal in external clock source, and use the switch (setting switch in figure 3.12) to clear and install WSN parameter in PIC's memory. GND of system is divided into two parts, which in first part is GND MCU that is main current reference. Second part is peripheral GND that is used in peripherals controlling to ON/OFF by NPN transistor. Connection between PIC and TRW2.4G will use seven wires and must feed power supply that dose not over 3.6 V. SPI Connector, which is micro-USB wire type, will be used to connect to HD for installing of WSN parameter before use. In addition to above installing that is made by wire, we can also install WSN parameter via wireless that will be explained in topic 3.3.4.

For clock source in the SN, microcontroller will use clock source from internal clock RTC that is self generating in operation. The microcontroller can drive RTC to 10 clock types. We tested all clock type in working, and found that the 32 MHz clock is optimal for working in normal operating (lowest power consumption and can continuously work), and will use at 31 kHz clock when sleep mode is called to use (more details in chapter 4). Clock source in the GN will be used at least 8 MHz clock, which is the maximum of clock RTC (not PLL clock). If clock is lower than that value, error may occur when connection with HD via SPI protocol. The reason that we use 20 MHz crystal in the schematic because it is designed to support in case that the above setup cannot work. In this case, we can also use the same schematic.

All available internal memory will be used from PIC microcontroller. They do not have any external memory, use RAM memory for storing and holding data transferred during several operations, and use program memory for storing WSN parameters and some important data. RAM memory cannot store those data because they will be deleted when no power feeding or resetting. Size of RAM memory will be able to indicate the quantity of ND number because we use RAM to keep the data transferred from all ND. The data storing in RAM of PIC is the best energy saving, which do not require another memory in data storing. This method is to reduce power consumption.

The Analog to Digital Converter (ADC) of the ND is divided into two parts. The first part is used to receive value from sensor in the schematic figure 3.12. The 100 Ohm resistor is installed in series with analog signal pin of sensor connector. This is the protection of over current flowing into microcontroller. If it has no resistor, damage may occur on the system. That resistor will make the range of bit to not full value at 1,024, but it is not a problem. The other part is used to check voltage of the battery. This microcontroller has a specific feature of internal reference voltage at 2.5 V. We can know capacity of battery by calculating from voltage and compare it with the reference voltage. We can see in the schematic on Pin 5 (ADC for checking battery) and 6 (reference voltage) of PIC18F24J11 that those pins linked together, and can calculate to find the capacity of battery from the formula that is $(V_{reference}*1,024)/V_{battery}$, given $V_{reference}$ = 2.5 V and $V_{battery}$ = 2.7 V (battery is low at this volt). When the ADC value on Pin 5 detects over or equal to 948, battery is low.

The energy source is two AA-size batteries. The sensor node (task assigned) will only use this energy source but the gateway node will use the energy from host device via mini-USB. Since the gateway node must be used all times, its energy source must use with unlimited energy source from the host device by AC adapter.

Sign of statement acknowledgement of the ND can be seen from LEDs. Green and Orange SMD LED on device will show when occur following statement as in table 3.3. Only important states are shown because LEDs use high current.

Table 3.3 LED statement acknowledgement

Green LED	Orange LED	Statement
Off	On	1. No WSN parameter and enter to installing wireless method.
Off	Normal Blink	2. Cannot connect with Host Device and try to connect.
Off	Quick Blink	3. Error in installing value with Host Device.
On	Off	4. Success in installing value with Host Device.
Normal Blink	Off	5. Low battery.
Quick Blink	Off	6. Cannot find another node in the near area.
Off	Off	7. Normal operating in Sensor Node.
On	On	8. Normal operating in Gateway Node.

The PCB layout designed by PCB Wizard 3.50 that is easy to use and free software. The most electronics in PCB is SMD type because it is small size, but more expensive than regular size. The connection with HD for installing information or data transfer to the ND and the connection with sensor will use micro-USB. The power, which must not exceed 3.6 V because this schematic has no regulator for converting voltage, can be fed via micro-USB or PCB pad. The hardware cost, which does not include the prototype cost, is shown in table 3.4.

Table 3.4 Hardware costs in node device

Oder	For 5 Units (THB)	Estimated for 100 Units (THB)
Parts	600	6,000
PCB	600	2,500
TRW2.4G	1,850	35,000
Total	3,050	43,500

The device and the algorithm must be developed together for the best result, such as researching of behavior between consumption of the current used in a ND and each statement (topic 3.3) in system, testing of connecting between a PIC and a TRW2.4G, and etc. We try to develop hardware by creating many prototypes, and test them. Finally, we can show prototype device, junction of connection for using, and schematic as below.

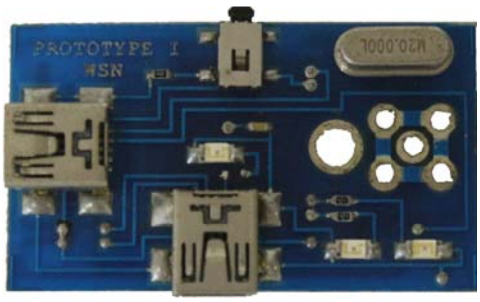


Figure 3.8 Node device (front)

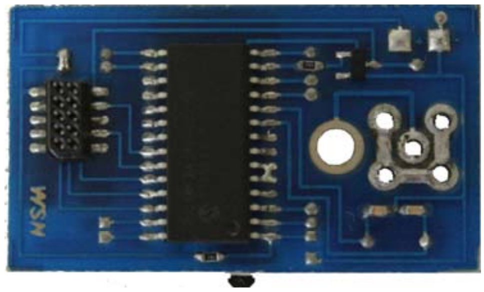


Figure 3.9 Node device (back)

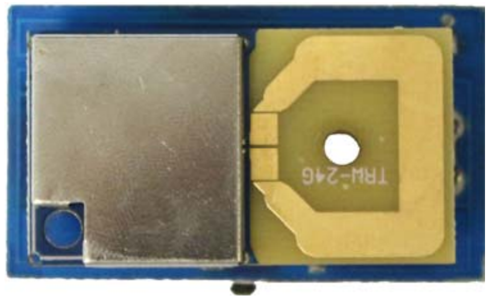


Figure 3.10 Node device with TRW2.4G (back)

(a) **(b)**

Figure 3.11 Connection with micro-USB

(a) Junction of connection between node device and host device

(b) Junction of connection between node device and analog sensor

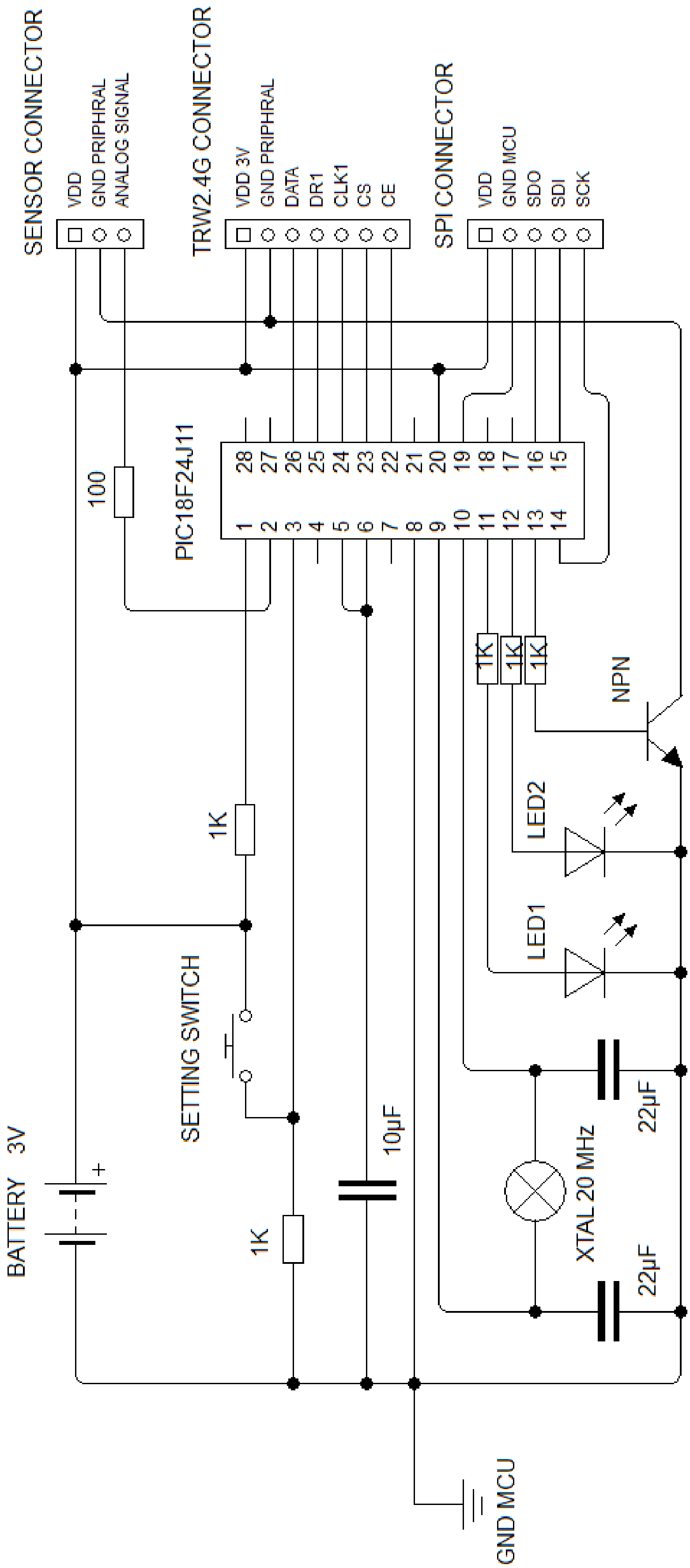


Figure 3.12 Schematic of node device

3.3 Algorithm of Node Device

In this topic, operation of node device will be controlled by algorithm. Each ND will run own task and transfer data together in communicating information. Those ND will use multi-hop wireless communication when over range. One process will succeed when the information arrive to HD. We will test and verify the functionality of ND along. The design of the algorithm is divided into two parts from the tasks assigned, which is SN and GN. Both of tasks assigned can be set into the ND by connecting with the HD via wire (SPI protocol) and wireless. In all working state of the ND, we divided into three parts, the first part is transmission state that will only send any data via wireless communication, the second part is reception state that will only receive any data via wireless communication when packet detected, the final part is sleep state that will turn off all peripheral in the ND and microcontroller will enter into deep sleep mode. Those states are basic method in the design algorithm and calculation power consumption. Integration between three states above requires many techniques for algorithm programming. In programming, the data will be transmitted how it arrive the destination node, the data must receive how it has completely data, the sleep will be used when the system requires saving energy. Those things are combined in there that can see according to any topic.

In process of data communication within the WSN system (see in figure 3.13, 3.14, and 3.15), the first type is the data communication in transmission state that begins at MCU. The MCU will control TRW2.4G by the SPI protocol for transmitting the wireless data or wireless packet. The SPI packet will be used during SPI protocol task. The TRW2.4G will transmit the wireless data via wireless signal to other node device. The second type is the data communication in reception state that is reverse process of the first type. This type will occur when the wireless signal come in. The third type is sleep state that does not perform any data communication. This type, which the PIC will use the deep sleep mode, will be used for saving energy. The last type is the data communication in between node and host device. In connection between node and host device, only SPI protocol will be used. Tasks of wireless communication are to operate wireless routing (for finding the path), wireless traffic (for allocating the path), and wireless transfer (for transferring the data). Task of the SPI protocol is to operate connection between TRW2.4G and MCUs.

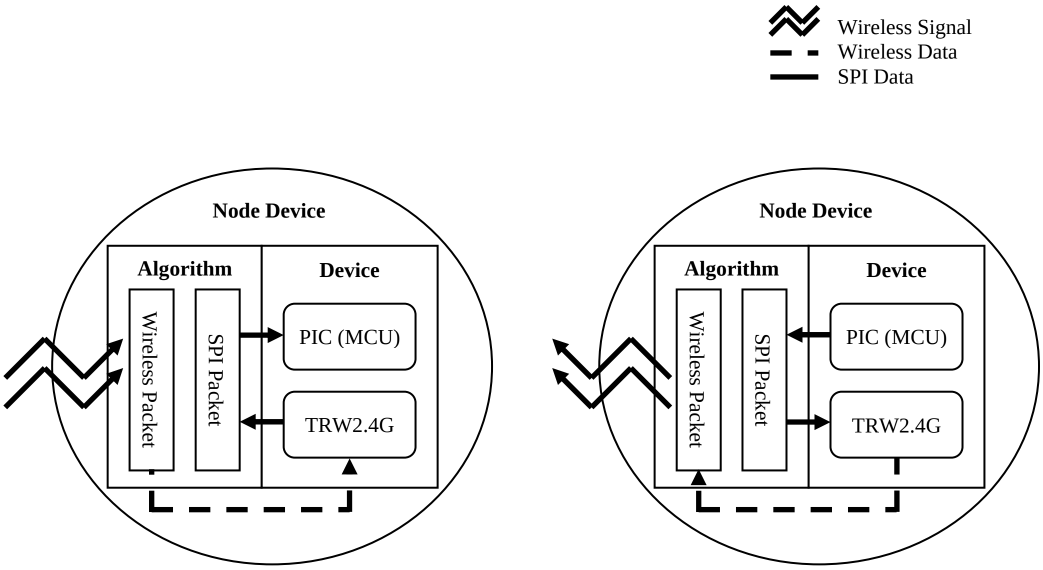


Figure 3.13 Data communication within node device in reception (left) and transmission (right) state

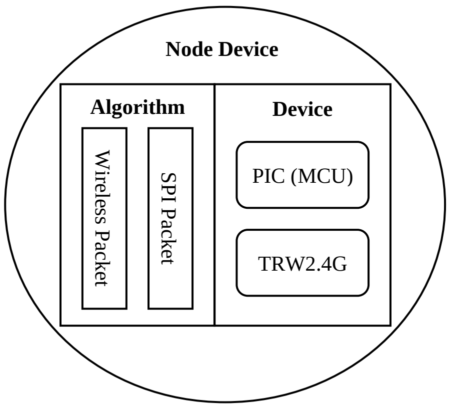


Figure 3.14 Data communication within node device in sleep state

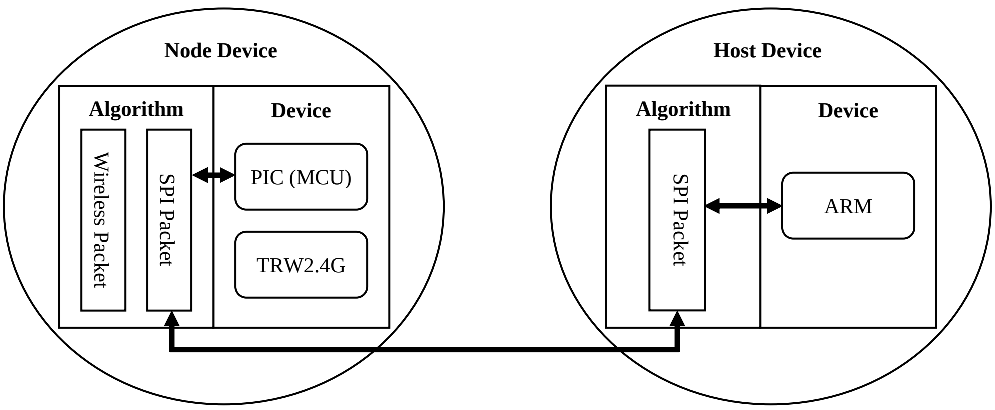


Figure 3.15 Data communication between node and host device

The data packets used in transfer within algorithm is divided into two types, which are the wireless packet for wireless communication and the SPI packet for connecting with other devices, such as TRW2.4G, and ARM. SPI packet is data communication in the SPI protocol.

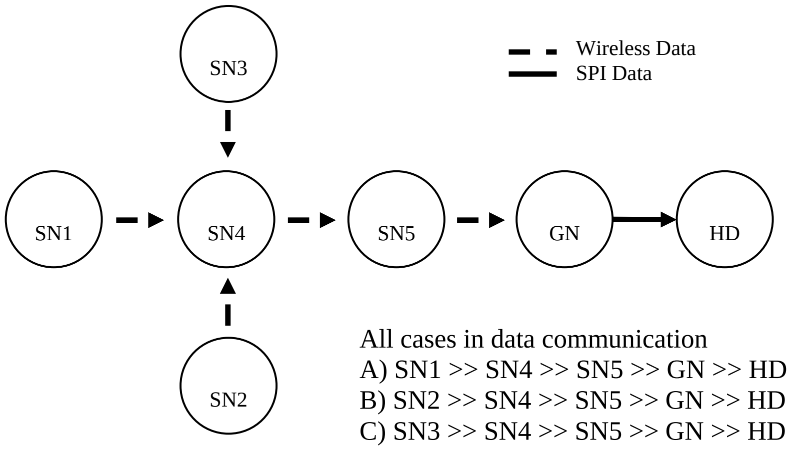


Figure 3.16 Data flow in example WSN system

An example of data flow in WSN system (overview) can be considered to many cases as shown in figure 3.16. These cases can occur one case at one time. Case A, case B, and case C may occur first. If first case occurs and does not finish yet, other cases will be able to interrupt while the first case continues.

The data may lose during communication by many events. The most events that cause from SN are damaged or no use, have to be solved by algorithm. For example of error event, the SN4 cannot be used. It will cause the losing data of SN1, SN2, SN3, and SN4. The dynamic routing mode (can be seen in special mode at topic 3.3.5) will be used to solve that problem, can see in figure 3.17. That mode will create a new path for transferring data by bypassing the SN4. The data of SN1, SN2, and SN3 will be transmitted to pass to other path. But the data of SN4 will lose completely because the SN4 is damaged and cannot be used.

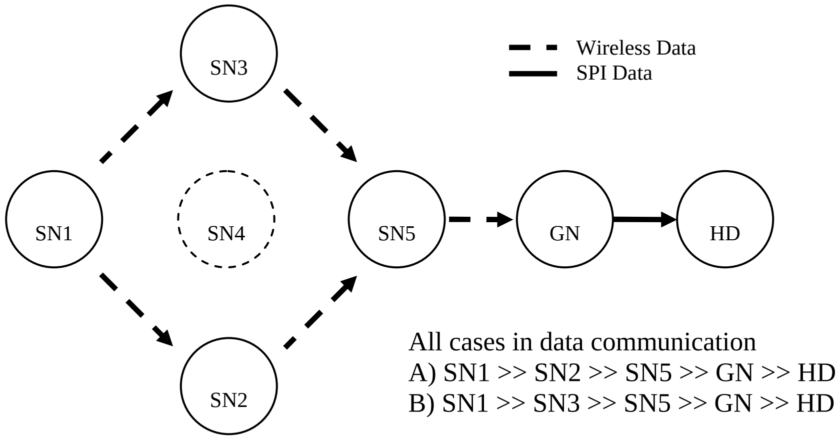


Figure 3.17 Solution of error event (SN4 missing) of data flow in example WSN system when dynamic routing mode is used

The flow contents of this topic are shown in figure 3.18. We will explain step to step. The beginning of content explains how to use the TRW2.4G (topic 3.3.1-3.3.2) and the basic of working function in the WSN system (topic 3.3.3). The TRW2.4G will be used in all following topics. The second is the assigning task to node device (topic 3.3.4). That thing must be done before use or to go to next topics. The third is special mode of algorithm (topic 3.3.5) that is a part in the algorithm of WSN system, but we will explain to extended details. The final is the algorithm of WSN system (topic 3.3.6). This is the most complex algorithm and gathers all of algorithm into this, and including the example WSN system algorithm (topic 3.3.7).

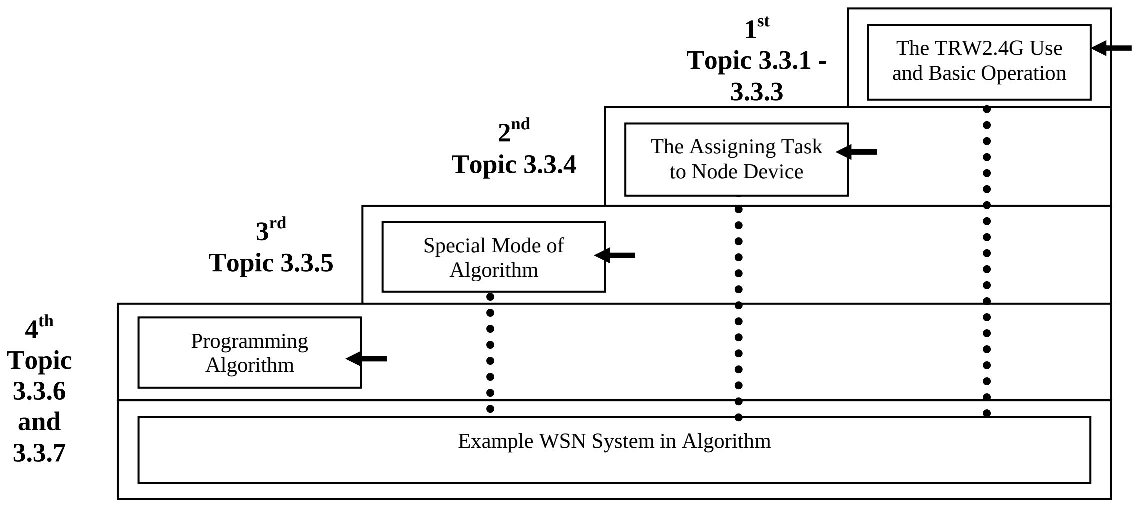


Figure 3.18 Step of content in algorithm of node device

3.3.1 TRW2.4G in Connecting Method with PIC

SPI protocol will be used in connecting between TRW2.4G and PIC as seen in figure 3.19. Pins in connecting of TRW2.4G include DR1, CE, CS, CLK1, DATA, VCC, and GND as shown in figure 3.20. The TRW2.4G can be programmed via PIC in configuration word, which always must be done before operation or first feeding power, or turning power off and on again. The configuration word will use a simple 6 wire interface except DR1, but in running on operation ‘RX’ or ‘TX’ must use 7 wire. Data rate of TRW2.4G is determined by the speed of PIC. Modes of TRW2.4G can be chosen from two modes by configuration from the SPI. We decide to choose the Shock-Burst mode, which can work and use low current in operation.

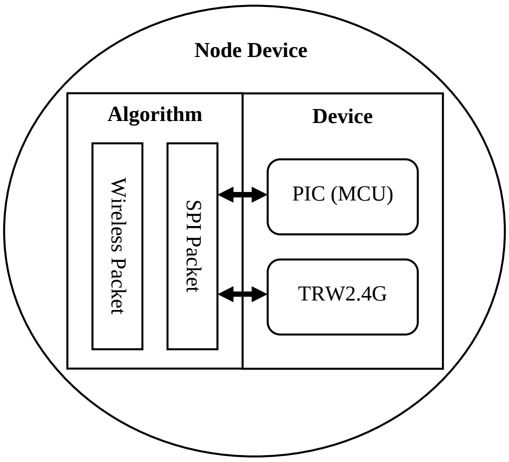


Figure 3.19 Data communication between PIC and TRW2.4G

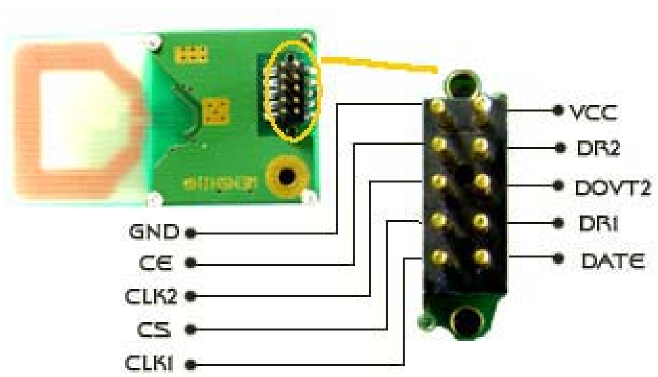


Figure 3.20 Pins of TRW2.4G

The Shock-Burst technology uses on-chip first-in-first-out (FIFO) to clock in data at a low data rate and to transmit at a very high rate thus enabling extremely power reduction. The comparison of current consumption between with and without Shock-Burst is shown in figure 3.21. By allowing the digital part of the application to run at low speed while maximizing the data rate on RF link, the Shock-Burst mode reduces the average current consumption in applications considerably. Therefore, it will be used in our system. The Shock-Burst mode offers the following benefits:

- Highly reduced current consumption
- Greatly reduced risk of 'on-air' collisions due to short transmission time

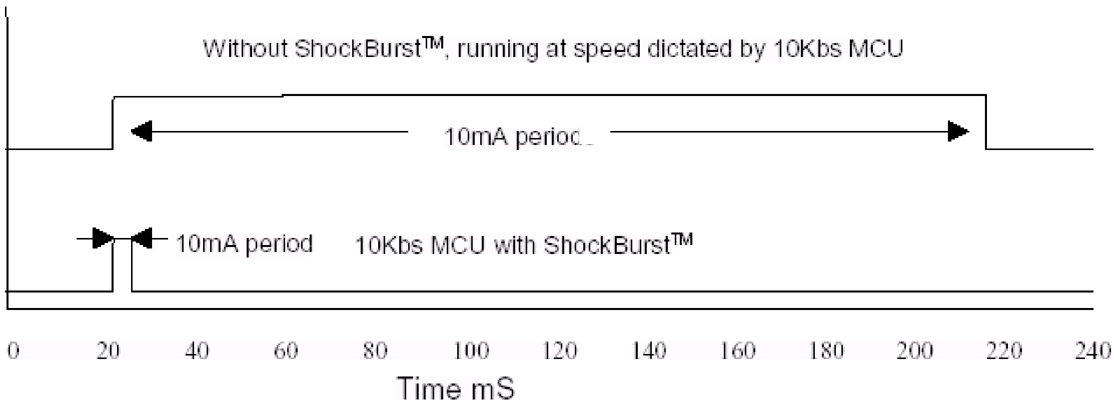


Figure 3.21 Comparing of current consumption with and without Shock-Burst (TRW2.4G datasheet)

3.3.2 TRW2.4G in Programming

The configuration word in Shock-Burst mode enables the TRW2.4G to handle the RF protocol and can be up to 15 bytes long. All configurations in position bits can be seen as in table 3.5.

Table 3.5 Configuration word in position of bit of TRW2.4G

Bit Position	Number of Bits	Name	Function
(MSB)143:120	24	Test	Reserved for testing
119:112	8	DATA2_W	Length of data payload section RX channel 2
111:104	8	DATA1_W	Length of data payload section RX channel 1
103:64	40	ADDR_2	Up to 5 bytes address for RX channel 2
63:24	40	ADDR_1	Up to 5 bytes address for RX channel 1
23:18	6	ADDR_W	Number of address bits (both RX channels)
17	1	CRC_L	8 or 16 bit CRC
16	1	CRC_EN	Enable on chip CRC generation/checking
15	1	RX2_EN	Enable two channel receive mode
14	1	CM	Communication mode (Direct or Shock-Burst)
13	1	RFDR_SB	RF data rate
12:10	3	XO_F	Crystal frequency of RF
9:8	2	RF_PWR	RF output power
7:1	7	RF_CH#	Frequency channel
(LSB)0	1	RXEN	RX or TX operation

In loading the configuration word to PIC, bit position 119 or MSB that is most significant bit, will be loaded first. The loading will end at bit position 0 or LSB that is least significant bit. All of the configuration loading have 120 bits or 15 bytes length. When the configuration word is loaded into TRW2.4G completely, only one byte, bit [7:0], need to be updated during actual operation for alternating between TX and RX mode. In addition to the alternating between TX and RX mode, the frequency channel of RF can also be changed between 1-125 channels.

The configuration word values in table 3.6 are default value, which will be loaded into TRW2.4G by PIC in node device. The hexadecimal values in the table will be fixed, except the value of ADDR_1 (address of node device), RF_CH# (frequency channel of node device), and RXEN (operation mode of RF, TX/RX). The value of ADDR_1 and the RF_CH# can be set from assigning task. The RXEN will be automatically used during process of node device.

Table 3.6 Configuration word value of TRW2.4G in our design

Bit Position	Value of Bits (Hex)	Name	Detail of function use
(MSB)143:120	0x8E081C	Test	Those bits will be loaded automatically
119:112	0x00	DATA2_W	No use RX channel 2
111:104	0x60	DATA1_W	12 bytes length of data payload for RX channel 1
103:64	0x0000000000	ADDR_2	No use RX channel 2
63:24	0x000000FFXX	ADDR_1	Address for RX channel 1 (XX=number of node)
23:18	0x43	ADDR_W	Use 2 bytes length for address
17		CRC_L	Use 16 bit CRC
16		CRC_EN	Enable on chip CRC generation/checking
15	0x6F	RX2_EN	Disable RX channel 2
14		CM	Use Shock-Burst mode
13		RFDR_SB	Use 1 Mb/s RF data rate
12:10		XO_F	Use 16 MHz crystal frequency of RF
9:8		RF_PWR	RF output power
7:1	0xXX	RF_CH#	frequency channel of RF (XX=frequency of node)
(LSB)0		RXEN	Default bit at 0 (TX mode)

Those configuration words, with the exception of the CRC (optional), must be given to the TRW2.4G before use in TX and RX mode. In TX mode, the PIC must generate address and payload section (figure 3.22) that is already fitted size of payload into the configuration word of the TRW2.4G. The TRW2.4G must specify size of payload before use, which must be done in configuration word in bit position of DATA1_W (use DATA1_W only in our system). When using the TRW2.4G on-chip CRC feature, we must ensure that CRC is enabled, and the TRW2.4G must use the same length for both the TX and RX devices. In our design, when we set the RF channel in the configuration word, we will not change the RF channel, but will use same channel for all processes of node device.

The data packet for communication of TRW2.4G, which has maximum size at 256 bits, is divided into 4 sections as seen in figure 3.22. These are:

- <u>Preamble</u>: is required in the Shock-Burst mode. In TX, the preamble is automatically added to the data packet of TRW2.4G in the Shock-Burst mode and thereby gives extra space for payload. In RX, the preamble is stripped automatically. This has 8 bits length.
- <u>Address</u>: is destination address for receiving data. We must set the number of bits for address in the data packet of TRW2.4G. This enables the TRW2.4G to distinguish between address and payload data. The

address field is required in Shock-Burst mode, 8 to 40 bits length, address automatically removed from received packet in the Shock-Burst mode.

- Payload: must be specified the size of payload in configuration word before use. The size of payload enables the TRW2.4G to distinguish between payload data and the CRC bytes in a received package. This section will be generated data by MCU. The data to be transmitted in Shock-Burst mode payload size is 256 bits minus the following: (Address: 8 to 40 bits + CRC 8 or 16 bits).
- CRC: enables TRW2.4G on-chip CRC generation and de-coding. The CRC is stripped from the received output data and have 8 or 16 bits length.

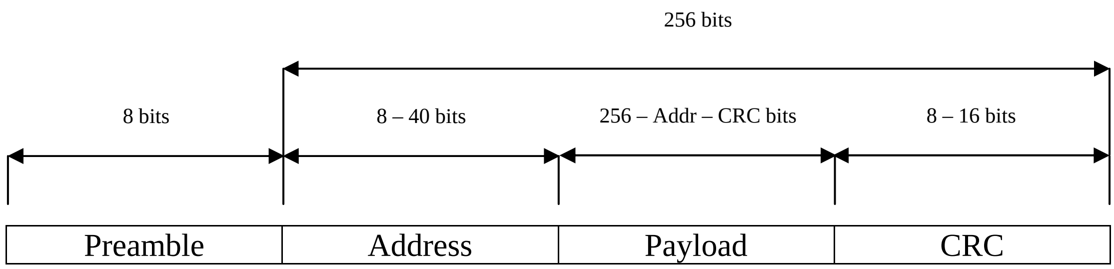


Figure 3.22 Data packets for wireless communication of TRW2.4G

The other wireless packets in the design protocol (can see in topic 3.3.6) will be contained into the payload of data packet of TRW2.4G (figure 3.22). In operation of any data packet in our system, The first thing will begin to operate the data packet of TRW2.4G when operation of wireless communication occurs, and then other packets will be operated within payload of TRW2.4G that is passed from CRC. All wireless packets of the design include, Assigning Task Packet, Dynamic Routing Packet, and Transfer Packet.

3.3.3 The Basic Operating Functions in WSN System

This section will gather the basic operating functions that are the components of each topic as below.

The WSN Parameter

This parameter will be used for assigning task of node device, which has the specific and the limit value of each, includes as follows:

- Installation method of WSN parameter: It can choose two methods to Wire or Wireless method in the installation of WSN parameter.
- Number of Node (NON): In our design, it can set up to 1,500 numbers. That capacity depends on the RAM memory of PIC18F24J11. But we set maximum of the NON at 100 because this value is the best optimal for our research.
- Frequency of Node (FON): This is a frequency channel of TRW2.4G. The same network must use the same channel that can set up to 125 channels from configuring of TRW2.4G.
- Assigning task of Node (AON): This is an indicator of working in WSN system that includes Sensor Node (SN) and Gateway Node (GN).

The Wireless Broadcasting and the Configuration Address

The wireless broadcasting is to broadcast the wireless signal via the same address, in here, it is the value of 0xFFFF. The SN and GN will be able to transfer the data together via the same address. Since all nodes use the same address in communication, which increases bandwidth of data, that method has the limit of connection in several times, thus we must test the connection of those nodes and learn the behavior of the connection for complete system. For testing of wireless broadcasting, we will explain in the next chapter.

The address configuration will be used during operation of node device. We will set 0xFFFF to be address of node device when we want to use in wireless broadcasting. But we will load the NON of each node device (this value is assigned by method in topic 3.3.4) to be address of each node device when we want to use in normal wireless transmission.

The Basic Function in Wireless Communication

The basic function will be used in algorithm design. For understanding the basics on how to use node device in WSN system, we need to describe those basics as follows:

1. Configuration Word: Before using the node device, we must configure the value to it. These values are shown in table 3.5. In the configuration of these values by programming, we will use the method from datasheet of the TRW2.4G. In setting and loading these values, we must do via the host device to assign task to node device. In configuration word, the node is just configured once, that is enough to work. But if the node needs to change that configuration, we can change it later. We can show the method of the configuration in figure 3.23.

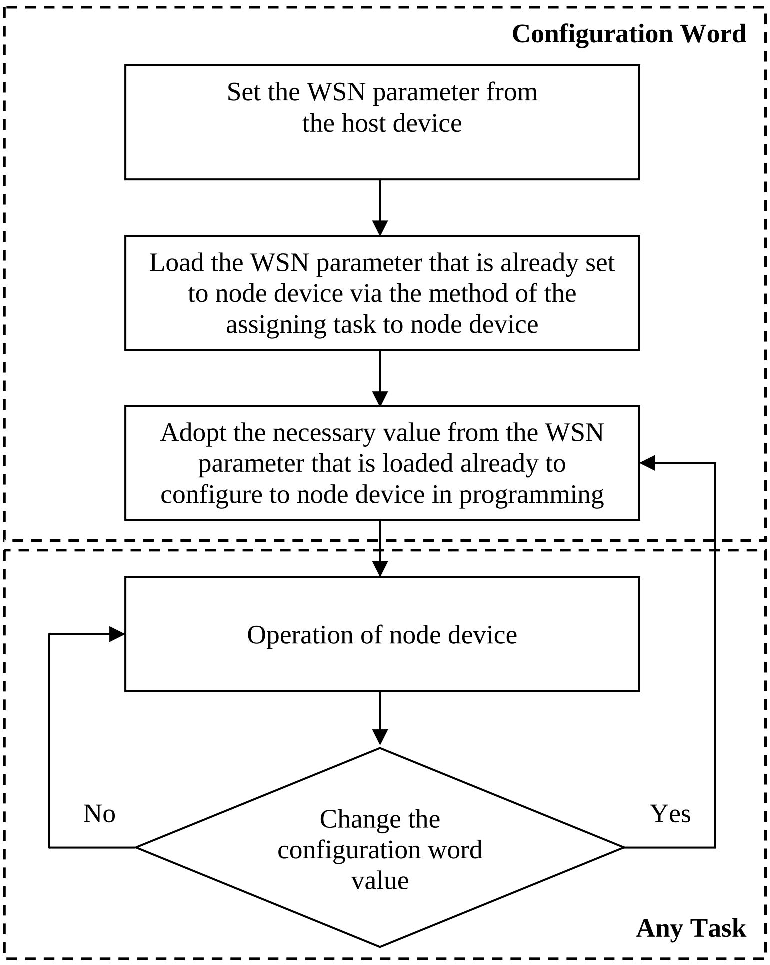


Figure 3.23 Method of configuration word of node device

2. Configuration Mode: This function will be used before transmission or reception of data. We must configure mode of the node device (TX or RX) after the configuration word is configured completely. In case of changing the task, we need to configure mode again as in figure 3.24.

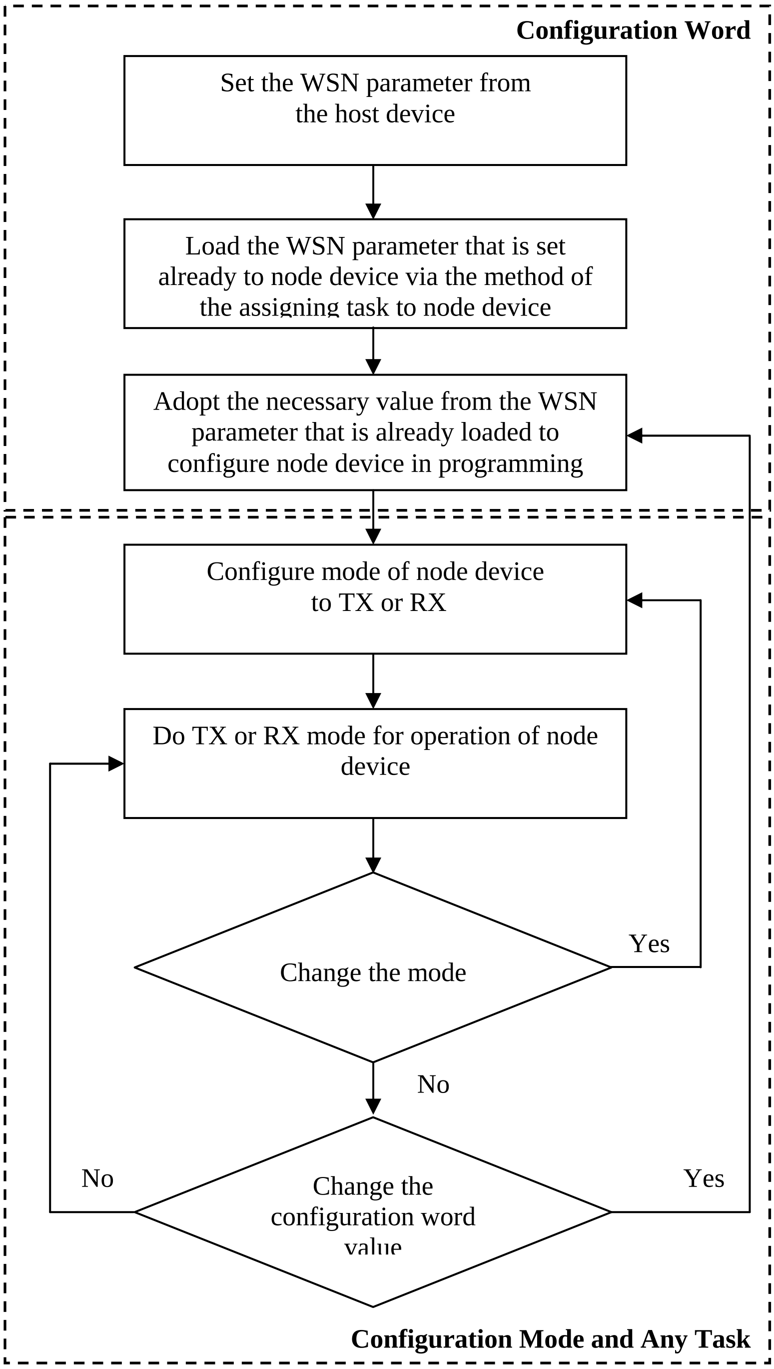


Figure 3.24 Method of configuration word and configuration mode of node device

3. Transmitter mode (TX): This function is used in sending the data via RF module. We must configure the configuration word before by the WSN parameter. Then, we will configure the configuration mode to the TX mode. The node address, and destination address in the TX mode, when the node works in broadcast mode (below topic), the both address will be set at 0xFFFF. If the node works in normal mode (not broadcast mode), the node address will be set same as the NON and the destination address will get form dynamic routing mode for system operation. We can show this function in figure 3.25.

1 time	1 time	1 time	n times
Configuration word	Set TX	TX	TX >>

Figure 3.25 Function in transmitter mode

4. Receiver mode (RX): This function is used in receiving the data via RF module. We must configure the configuration word before by the WSN parameter. And then, we will configure the configuration mode to the RX mode. The node address in the RX mode, when the node works in broadcast mode (below topic), will be set at 0xFFFF. If the node works in normal mode (not broadcast mode), the address will be set same as the NON. The destination address does not set. We can show this function in figure 3.26.

1 time	1 time	1 time	n times
Configuration word	Set RX	RX	RX >>

Figure 3.26 Function in receiver mode

5. Broadcast mode (Wireless broadcasting): This function is used in sending and receiving the data via RF module. We must configure the configuration word by the WSN parameter and 0xFFFF for wireless address of this RF module (within node device) and configure the configuration mode to the TX and RX mode. In configuration for TX and RX mode, the both tasks will be alternated to configure in some period of time, because the TRW2.4G is the half-duplex wireless communication. The half-duplex will not able to transmit and receive the data at the same time. The end of this function is to send the confirm data 10 times (the least times that target can acknowledge) to the target node. The broadcast mode is used in the assigning task to node device and the dynamic routing (see more detail in the assigning task to node device and the dynamic routing algorithm topic). We can show this function in figure 3.27.

1 time	1 time	1 time	1 time	1 time	*n* times	1 time	10 times
Configuration word	Set RX	RX	Set TX	RX	TX/RX >>	Set TX	TX
	Standard function				Standard *n* functions	If complete	

Figure 3.27 Function in broadcast mode

The Basic Working State in Node Device

A node device includes of the transmission state, the reception state, and sleep state as follows:

1. Transmission state: The CPU of PIC18F24J11 will be active and the TRW2.4G will be a transmitter.

2. Reception state: The CPU of PIC18F24J11 will be active and the TRW2.4G will be a receiver.

3. Sleep state: The CPU of PIC18F24J11 will be inactive and the TRW2.4G will be inactive as well.

The Basic Function of Sensor Node

A node device, which is assigned to be a sensor node, has the functions of sender and repeater.

1. Sender: This is the sensor node that only has the working in transmission. The sender will use the shortest time in the WSN system. That means the sender will consume the lowest power.

2. Repeater: This is the sensor node that has the working in transmission and reception. The repeater will use the most time in the WSN system. That means the repeater will consume the most power.

The Basic Function of Gateway Node

A node device, which is assigned to be a gateway node, has the functions to receive and gather the data from any sensor nodes. And then, the gateway will send the gathering data to host device via SPI protocol.

3.3.4 Assigning Task to Node Device

The task assignment, which is installation of WSN parameter, is necessary in all node devices. If the task assignment does not have in the ND, the ND will not be able to use in WSN system. The WSN parameter is stored into PIC’s memory of the ND. The task assignment to node device can be done by two methods that are wire method and wireless method as figure 3.28, as follows:

- **Assigning Task in Wire Method.** This method can install the WSN parameter into only one ND at a time. We must use USB cable in connecting between node and host device all time when we want to install to another ND. We can see all process in topic 3.6.1.
- **Assigning Task in Wireless Method.** This method can install the WSN parameter into many NDs at a time. We just use USB cable in connecting between a node and the host device for only first time that will configure the ND as ‘Task Transmitter’. And then, we will be able to install WSN parameter to other NDs via the Task Transmitter. We call those NDs to be ‘Task Receptor’.

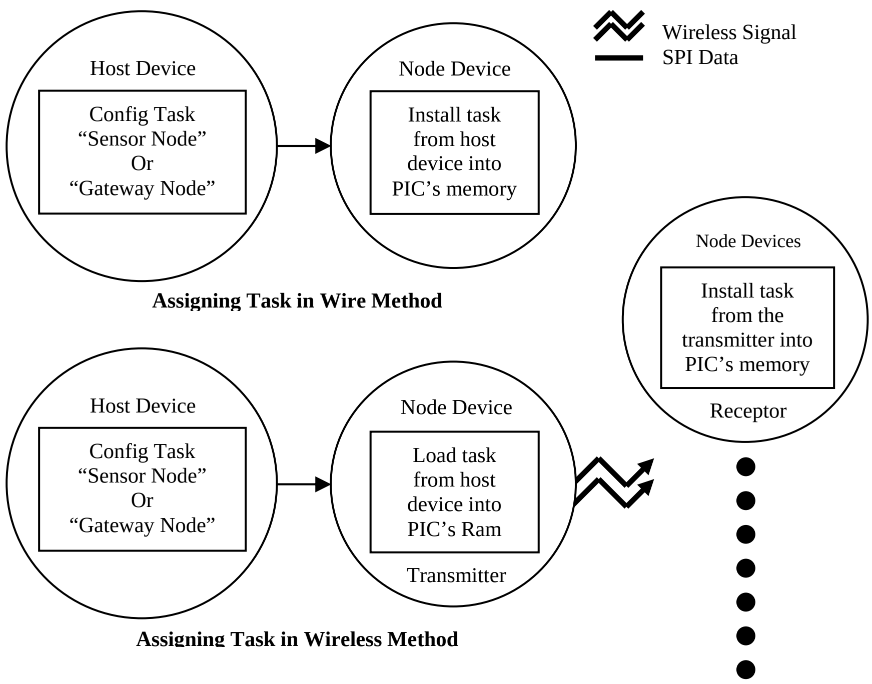


Figure 3.28 Operating flow of assigning task to node device in wire and wireless method

3.3.5 Special Mode of Algorithm in WSN System

This topic will explain extended method in the dynamic routing and the transfer mode. The both modes are used in algorithm of WSN system for many benefits. We can explain as follows.

Dynamic Routing Mode

The dynamic routing mode, which is used to search the route for transmission data via wireless to destination or another ND in the nearby area, and handle the problem be illustrated in following scenarios. When the ND cannot connect with any ND in nearest area, the ND in the nearest area will probably connect with other NDs. Furthermore, this mode can support in adding new ND into the existing network. But this mode still has a disadvantage that is long operating time. Although this mode, which increases the load to the system, will increase power consumption and decrease the lifetime of system, we use this mode anyway because the benefits out weigh the disadvantage. The above is the meaning of “routing” protocol, but what does “dynamic” means. Dynamic means continuous, which means using of routing protocol over the lifetime of system, not just once.

We will explain two cases of events when dynamic routing mode is used. The first case, a node device (SN4) is missing (figure 3.29). The second case, a node device is missing (just like the first case) and node device (SN2) is moved to a new location (figure 3.30).

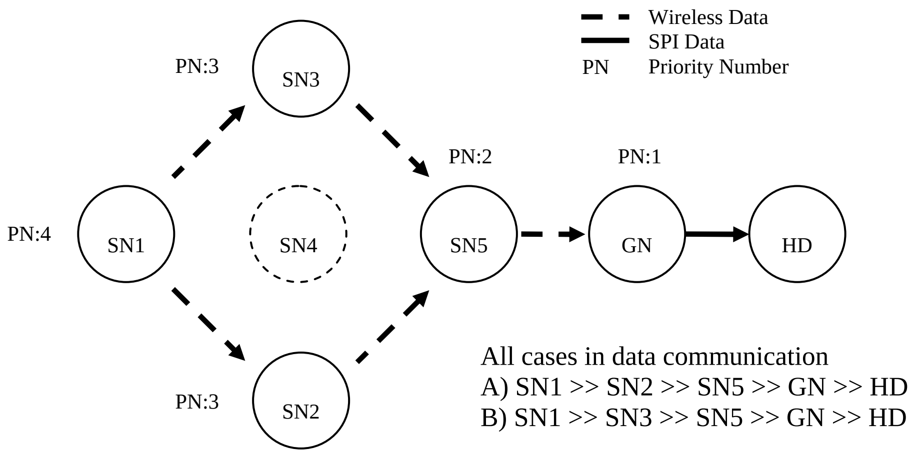


Figure 3.29 Solution of error event (SN4 missing) of data flow in WSN example system when dynamic routing mode is used

For the first case in figure 3.29, according to figure 3.17, priority numbers will be added in explanation. A priority number is used to define the direction of wireless data. It is essential in dynamic routing mode. The priority will be created automatically (see further detail in algorithm programming of dynamic routing mode). The SN4 will not be able to receive a priority number because it is missing. All SNs in example will be able to send the data except the SN4.

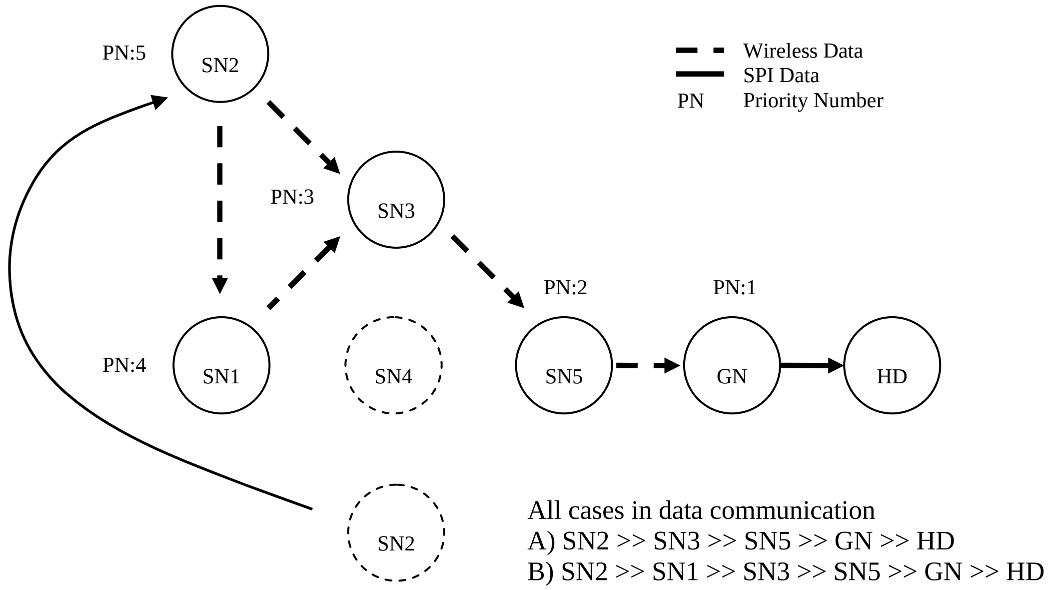


Figure 3.30 Solution of error event (SN4 missing) and SN2 is moved in WSN example system when dynamic routing mode is used

The second case is shown in figure 3.30 when the SN2 is moved. The priority number will be changed automatically from 3 to 5. It is changed because the SN2's old priority of 3 cannot be used at the new point (see more detail in algorithm programming of dynamic routing mode). We will see the cases in data communication that is changed from figure 3.29. In addition to two cases above, the dynamic routing mode is also used. In normal event as in figure 3.31 (direction of path based on priority number) where all SNs can be used, the dynamic routing mode always works for checking destination and traffic between those nodes. The checking traffic is also used for checking wireless signal in installing into any areas of each node. This mode is contained in sensor node and gateway node. For the SN, this mode will has some period of operation, and will enter to sleep state when the operating time out. This mode will change into transfer mode when it finishes finding destination.

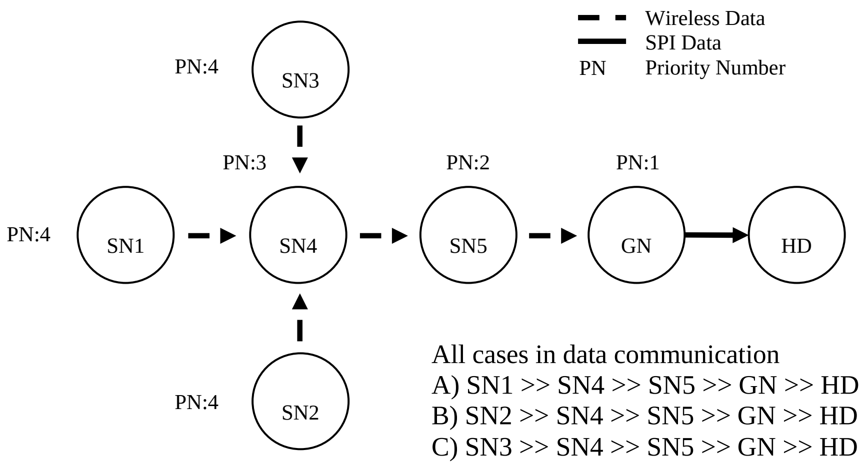


Figure 3.31 Normal event of data flow in WSN example system when dynamic routing mode is used

We will show operating flow of dynamic routing mode in figure 3.32, in fact, the algorithm of this mode will be divided into two parts for sensor node and gateway node. At the beginning step, node device must be assigned for task (SN or GN). The second step is to check wireless signal of each sensor node. The gateway node does not need to check wireless signal. The process of checking wireless signal is contained in the process of checking priority number. We can know when it is ready from LED statement in table 3.3. The next step, the timeout clock will start during the operation of following process. For the SN, the process of creating priority number will run when the SN has no priority number. If the timeout occurs during creating the priority number, the task will enter to sleep state and reprocess again. For the GN, the process of creating priority number will be used in dispensing the priority to any SN. The last step, the timeout also works similar to the previous step. The process of searching for destination node will run in the both (SN and GN) when the process of creating priority number is complete, and enter to transfer mode when the process of searching for destination node is complete. If the timeout occurs during searching the destination node, the task will enter to sleep state and reprocess again. The process of creating priority number in algorithm will be called "Making-Priority" and the process of searching destination node in algorithm will be called "Routing".

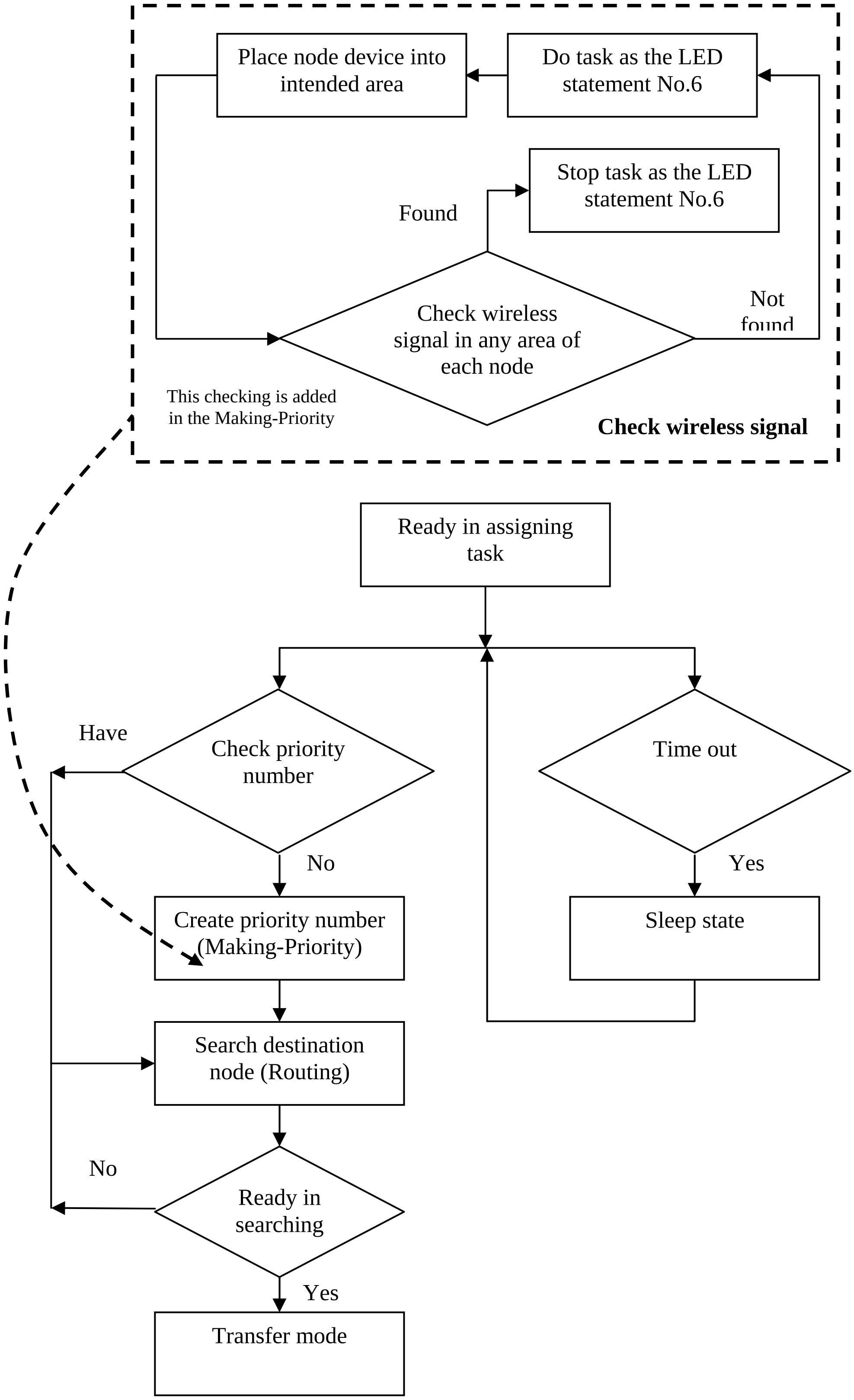


Figure 3.32 Operating flow of dynamic routing mode

Transfer Mode

This mode is used to send and receive the data from sensor and battery status of sensor node (gateway node will not use battery because it will use the energy from host device) via wireless communication when the destination is found from dynamic routing mode. The special feature in this mode is data encryption and decryption. They are created for protecting data jam between node devices (SN and GN). It is to increase accuracy in those data. The method of data encryption and decryption will be explained in programming algorithm of transfer mode. We will show operating flow of transfer mode in figure 3.33.

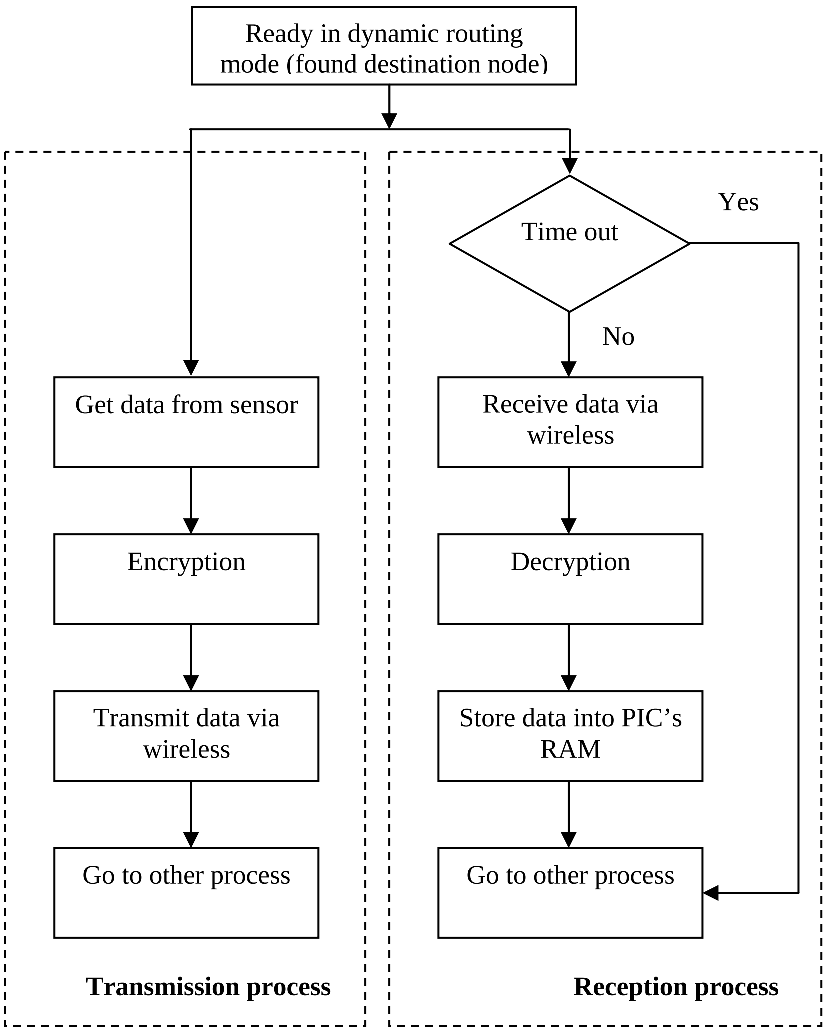


Figure 3.33 Operating flow of transfer mode

This mode (see operating flow in figure 3.33) is divided into two parts that are transmission process and reception process. The transmission process will be used in data encryption and sending process. The reception process will be used in data decryption and receiving process. The first of all, the task in this mode on both SN and GN will begin when dynamic routing mode finishes completely.

In case of transmission process, the first step is to get analog data from sensor. These data will be saved within PIC's RAM. The next step will encrypt data by algorithm into all of analog data, and transmit them to destination node (determined by dynamic routing) via wireless. We will see no timeout in this process because this process does not need to respond message to destination node. It is one way in data sending. If we use the responding message, it requires a lot of load to the system until the system is unusable. In other process of this case, it do task as the sleep state.

In case of reception process, the first step is to start together with the timeout clock. As long as the timeout is not over, the task will continue. When the timeout is over, the task will be terminated. The second step is to receive the data from destination node (determined by dynamic routing). The received data is encrypted, which must be decrypted in next step. If the data, which is decrypted, is correct, it will be stored into PIC's RAM. The task will stop when the data is received completely or the timeout occurs. In other process of this case, it may do task as the sleep state (time out or mistaken) or the dynamic routing mode (reception complete).

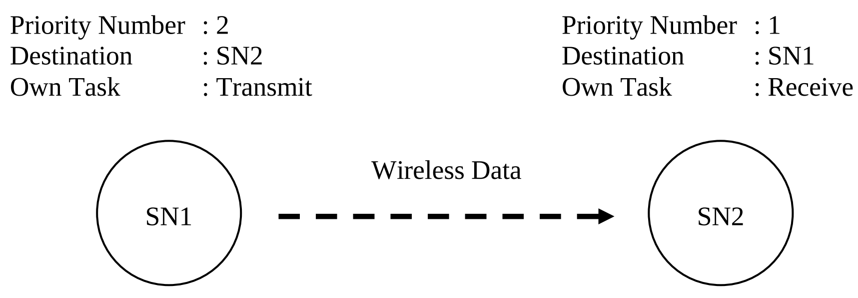


Figure 3.34 The first example in transfer mode

The first example in transfer mode is shown in figure 3.34, we will see the both SN that already have the priority number and the destination node. The SN1 is transmitter and the SN2 is receiver. This example completely has conditions and can run the task in transfer mode. In each node, it compare its and destination's priority numbers. If the SN has the priority number that is higher than other SN, that SN will do task in transfer mode as a transmitter. But if it is the other way around, the SN will do task as a receiver.

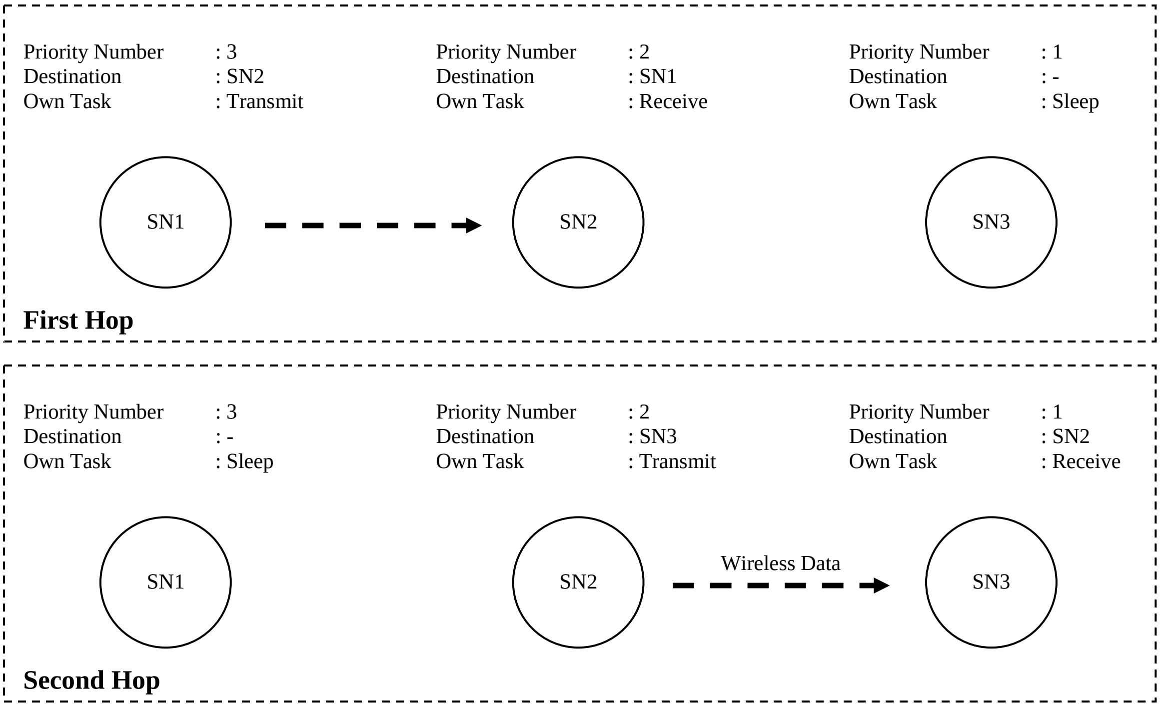


Figure 3.35 The second example in transfer mode

The second example in transfer mode is shown in figure 3.35, we will show the method that is called "Multi-Hop". The multi-hop is a communication that transfers data through many nodes. From the first example, we will see the SN that sends data between two nodes, it is the most basic in transfer mode. But in the second example, we will see the SN that sends data pass three nodes by adding the SN3. This example is more complex than the first example. The first hop and the second hop will be operated the same way as in the first example by starting at the first hop and ending at the second hop. During transfer mode, one SN will not have communication pair and enter to sleep state. As a result, SN will not find the destination node in dynamic routing mode until the first hop is complete because dynamic routing mode cannot be interrupted during transfer mode.

3.3.6 Algorithms of WSN System

We will explain in data packet of each algorithm, programming process, and relationship of all algorithms. All of algorithm will be gathered in this topic, as follows (A1 – A7 are the code name of each algorithm):

- Algorithm of assigning task in wire method (A1)
- Algorithm of assigning task in wireless method (A2)
- Algorithm of dynamic routing mode in the sensor node (A3)
- Algorithm of dynamic routing mode in the gateway node (A4)
- Algorithm of transfer mode (A5)
- Algorithm of sensor node (A6)
- Algorithm of gateway node (A7)

Overview algorithms of WSN system are separated into function blocks as in figure 3.36. To connect between algorithm of gateway and sensor node, the communication will be peer-to-peer, in which each mode in function box will be able to connect only with the same mode, such as, A3 will be able to connect with another A3 and A4, but the A3 cannot connect with A5.

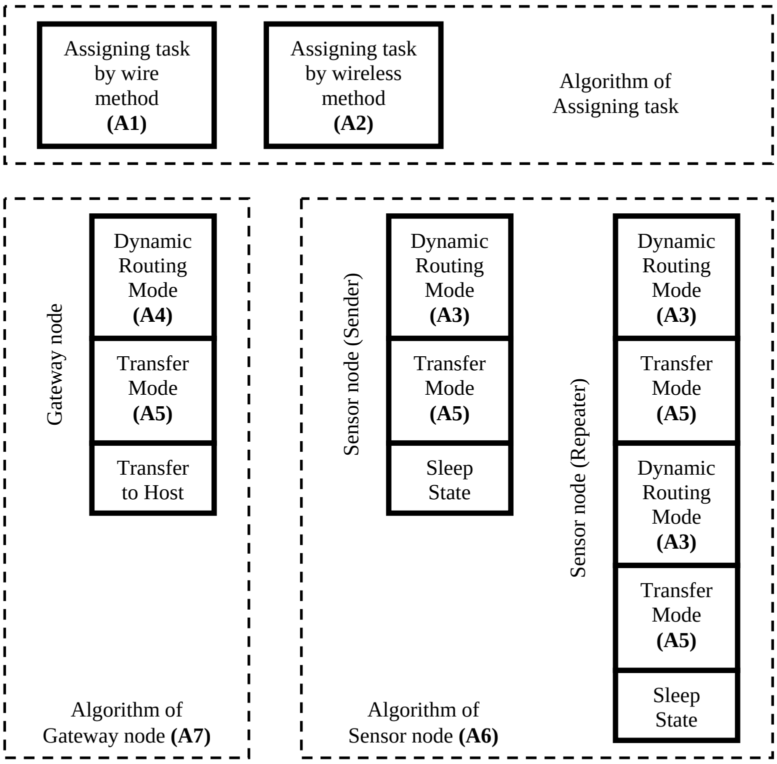


Figure 3.36 Overview algorithm of WSN system in function block

Data packets of each algorithm, those packets have the sequence of transmission and reception that are divided into byte and bit order.

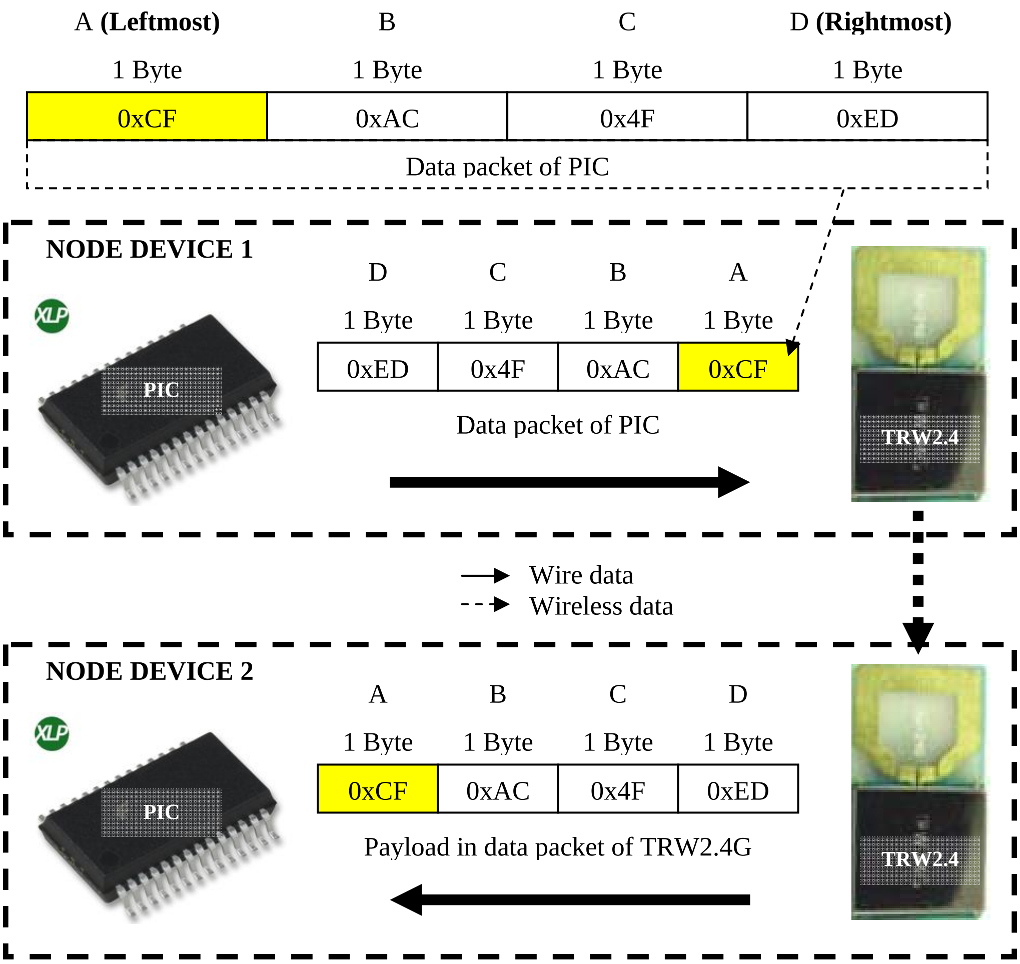


Figure 3.37 The example sequence of byte order in transmission and reception

In byte order of each data packet of each algorithm, which the byte order is big-endian, the leftmost byte will be sent first, as in figure 3.37. In the transmission (PIC sends the data packet to TRW2.4G) and the reception (PIC receives the data packet from TRW2.4G), the byte order will be done first always at the leftmost byte.

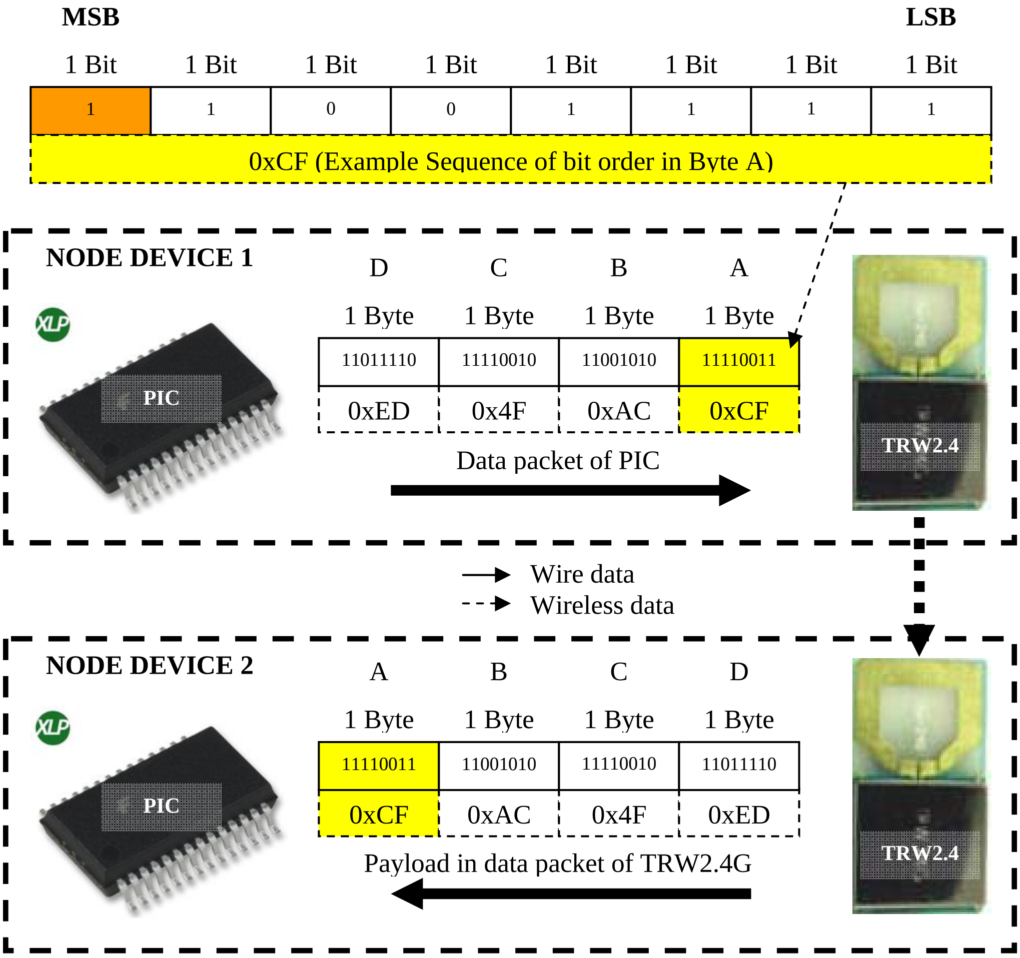


Figure 3.38 The example sequence of bit order in transmission and reception

In bit order of each byte of each data packet, which the bit order is little-endian, the leftmost bit, or MSB (Most Significant Bit) will be sent first, as in figure 3.38. In the transmission (PIC sends the data packet to TRW2.4G) and the reception (PIC receives the data packet from TRW2.4G), the bit order will be done first always at the leftmost bit of each byte.

The overview relationships of all WSN algorithms are shown in figure 3.39, which will be described as follows.

1. The first process will clear all buffers of WSN system to initialize value and return default value of peripheral of microcontroller PIC18F24J11.
2. The second process will configure peripheral of MCU, such as TIMER0, TIMER1, and ADC (Analog to Digital).
3. In the third process, WSN parameter in program memory will be read. If those parameters are not installed yet, the task will enter to **algorithm of assigning task by wireless method (A2)**. The algorithm of assigning task by wireless method will be used instantly but we can choose the assigning task by wire method instead. To select wire method, we can do by pushing the setting switch. The task will enter to the interrupt task and **algorithm of assigning task by wire method (A1)**. The task will continue to the next process when the WSN parameter is already installed.
4. The fourth process, the task will check the limiting value of those parameters (such as the frequency of node, FON is the limit value at 125). If they are over, the error display on LED of the ND will be shown and the task will be stopped in the infinite loop. If they are not over, the task will continue to check assigning task.
5. In final process, the task will enter to **algorithm of sensor node (A6)** or **algorithm of gateway node (A7)** from the assigning task. Inside of the both algorithms consists of **algorithm of dynamic routing mode (A3, A4)** and **algorithm of transfer mode (A5)**. The task will be configured the initial value of that assigning task before the task will do that algorithm. If the task in the algorithm of SN ends, task will begin again at point A and repeat infinitely until the task is interrupted or has no energy left. We can know working status from LED as in table 3.3.

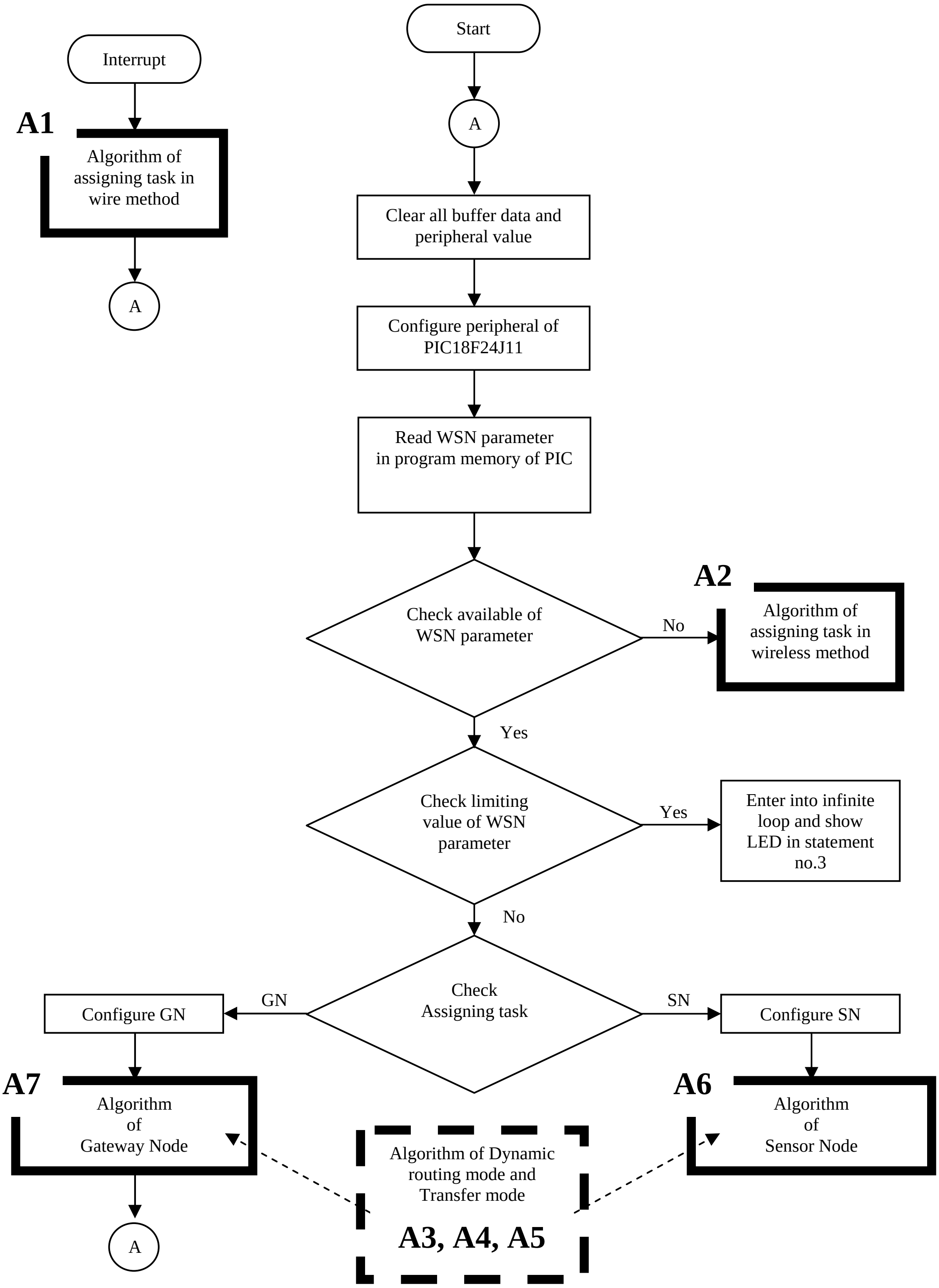


Figure 3.39 Overview algorithm of WSN system

Programming Algorithm of Assigning Task in Wire Method (A1)

For installing WSN parameter to node device in wire method, this method will use the SPI packet in wire communication as show in figure 3.40. The detail of this packet can be shown as follows:

- Master/Slave (Leftmost): has 1 byte, used in selecting master or slave mode for SPI communication.
- Addr1: has 1 byte, used in reference of AON byte position.
- AON: has 1 byte, used in setting assigning task of node device.
- Addr2: has 1 byte, used in reference of FON byte position.
- FON: has 1 byte, used in setting frequency of node device.
- Addr3: has 1 byte, used in reference of NON1 byte position.
- NON1: has 1 byte, used in setting number of node device (first byte).
- Addr4: has 1 byte, used in reference of NON2 byte position.
- NON2 (Rightmost): has 1 byte, used in setting number of node device (second byte).

Leftmost								**Rightmost**
Master/Slave	Addr1	AON	Addr2	FON	Addr3	NON1	Addr4	NON2
1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte

Figure 3.40 Assigning task packets (SPI) in wire method

Algorithm of this method is shown in figure 3.41. The algorithm will do task when the setting switch is pushed. The interrupt will occur to algorithm as in figure 3.39.

1. The first process begins at clearing and erasing PIC's memory.
2. The second process is configuration of SPI protocol.
3. The third process, this is process in choosing method of assigning task, this case will be chosen to be wire method. If this case is chosen to be wireless, the connected ND will be set to be 'Task Transmitter' (see in wireless method).
4. The fourth process, the ND will be connected with the HD for installing WSN parameter.

5. The final process, the ND will write WSN parameter into own PIC's memory.

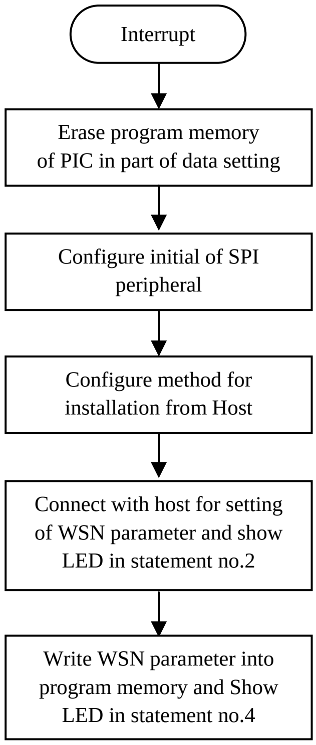


Figure 3.41 Algorithm of assigning task in wire method (A1)

Programming Algorithm of Assigning Task in Wireless Method (A2)

For installing WSN parameter to node device in wireless method, this method will use specific data packet with the length of 7 bytes in wireless communication as shown in figure 3.42. The detail of this packet can be shown as follows:

- Start (Leftmost): has 1 byte, used in starting this algorithm at value 0xA1.
- Command: has 1 byte, used in selecting type of this packet.
- AON: has 1 byte, used in assigning task of node device.
- FON: has 1 byte, used in setting frequency of node device.
- NON: has 2 bytes, used in setting number of node device.
- End (Rightmost): has 1 byte, used in ending this algorithm at value 0x1A.

Leftmost					**Rightmost**
Start	Command	AON	NON	FON	End
1 byte	1 byte	1 byte	2 bytes	1 byte	1 byte

Figure 3.42 Assigning task packets in wireless method

Algorithm for assigning task in wireless method is shown in figure 3.43. The algorithm will do task in infinite loop when this method is used. It will stop working when the installing succeeds, and can also get out from the infinite loop by resetting of PIC.

1. The first process begins the task by configuration of TRW2.4G to wireless broadcasting that will configure the same address and destination at 0xFFFF for all NDs. Those NDs will be able to transfer the data together via wireless for receiving of WSN parameter.
2. The second process is checking type Task Transmitter or Task Receptor within PIC’s memory. **Task Transmitter**, which has a function for transferring WSN parameter to other NDs, just uses in one ND that is connected with the host device. The value for Task Transmitter will be given from HD in wire method. If the ND does not

connect with the HD, this value will become a Task Receptor. **Task Receptor** has a function for receiving WSN parameter from task transmitter and setting it into PIC's memory of that ND. This method in assigning task requires at least two NDs that are one Task Transmitter and one Task Receptor. Only the Task Receptor can be installed with WSN parameter. If the Task Transmitter wants to install WSN parameter, it must do in wire method.

3. In case of Task Receptor, the task waits for coming of wireless data from host device. If that wireless data received have wireless packet, the task will write those WSN parameters from that packet received into the program memory of PIC and transmit the confirming data via wireless when writing is succeeded. The command in packet indicates working status, which may be setting data or confirming data. Finally, the task will reset the PIC to exit from this loop.
4. In case of Task Transmitter, the task will initialize SPI peripheral and wait to connect with host device. The task will wait for command from the HD when success in connecting the ND to the HD. If the coming command is exit command, the task will reset the PIC. If the coming command is not exit command, the task will still wait for the command. Next process is to load data from HD (which will use SPI packet in figure 3.33) when the command is received by the Task Transmitter ND. This ND will transmit that data to other Task Receptor ND via wireless. If the Task Transmitter ND, which is transmitting the data, receives the confirm data from the Task Receptor ND, it will stop transmitting the data and wait for other commands from the HD in the next stage.

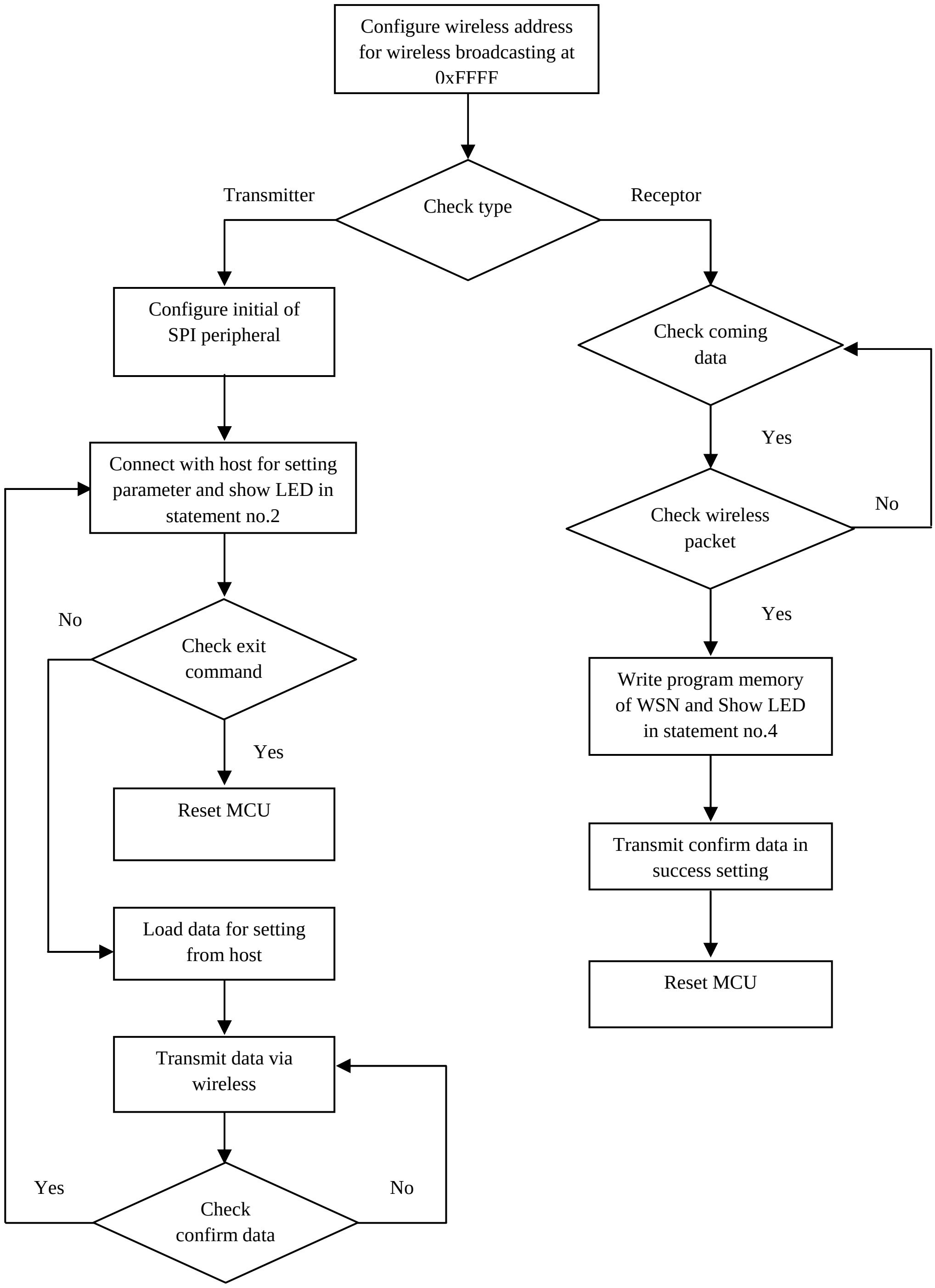


Figure 3.43 Algorithm of assigning task in wireless method (A2)

Programming Algorithm of Dynamic Routing Mode (A3, A4)

To find destination node among node devices, this mode will use data packet of 9byte size. This packet is called dynamic routing packet as shown in figure 3.44. The detail of this packet can be shown as follows:

- Start (Leftmost): has 1 byte, used in starting this algorithm at value 0xB2.
- Control: has 1 byte, used in controlling wireless traffic.
- Priority: has 2 bytes, used for indicating sequence of wireless communication.
- Address: has 2 bytes, used for indicating the current address of the node device.
- Destination: has 2 bytes, used for indicating the destination address of other node device.
- End (Rightmost): has 1 byte, used in ending this algorithm at value 0x2B.

The Priority Number (PN) is a significant sequential number for wireless communication. It is created to control the direction of data flow in the WSN system. The PN, which has 2-byte size, will be the most important when the PN is 1. Max of the PN is limited at the NON – 1. The GN will have the PN = 1 and is defined to start to dispense the PN to other SNs. A sample case includes one SN at PN = 3 and one SN at PN = 2. The SN at PN = 3 can send the data to the SN at PN = 2, but the SN at PN = 2 cannot send the data to the SN at PN = 3. The SN at PN = 2 can send the data to another node at PN = 1. An SN will only send the data to other nodes that have the PN lower than itself. Initially, all SN will have no PN and set the PN to 0. At initially distribution of PN, the GN will set the PN to 1 and dispense the PN to other SNs that have no PN yet, or 0. The SN can dispense PN to another SN when itself already has the PN. The value of the PN will increase or decrease by one according to the direction of data flow when the dispensation occurs.

Leftmost					**Rightmost**
Start	Control	Priority	Address	Destination	End
1 byte	1 byte	2 bytes	2 bytes	2 bytes	1 byte

Figure 3.44 Dynamic routing packet

Leftmost					**Rightmost**
0xB2	-	-	0xFFFF	0xFFFF	0x2B
1 byte	1 byte	2 bytes	2 bytes	2 bytes	1 byte

Figure 3.45 Initial values of dynamic routing packet

Before running task in this mode, we need initial parameter of TRW2.4G configuration to be wireless broadcasting, which means every SN or GN has the same address and destination at 0xFFFF. The initial value of dynamic routing packet is shown in figure 3.45. This algorithm, which will be used in both SNs and GN, is divided into two cases for SN and GN as explanation below.

Programming Algorithm of Dynamic Routing Mode for Sensor Nodes (A3 in SN)

This algorithm uses many processes, which is considered the most complex algorithm as shown in figure 3.46.

1. The first process, the task will configure TRW2.4G to wireless broadcasting, in which the TRW2.4G uses the address at 0xFFFF and initializes the dynamic routing packet by initial value.
2. The second process, the task is to check the PN. If the PN value is zero, the task will enter into the making-priority process. If the SN already has the PN higher than zero, the task will enter into the routing process (it means that this SN found other usable wireless signal in its own area). This process can also check wireless signal in that area by showing of LED in statement NO.6.
3. In the process of making-priority, the task will wait for dynamic routing packet and will set the priority into PIC’s memory when the PN successfully received. The task will send a responding message to the transmitter node when the PN is available. If the task meets timeout requirement, it will repeat the above again. The task will enter to next process that is the routing process when the PN is already set.
4. In the routing, this process is used in searching and handling the route of wireless communication between NDs (SNs and GN). If this process is succeeded, the task will continue in the transfer mode. If this process is still working, the task will be limited by timeout. Then, the task will clear the old priority number and enter to sleep state. The task will begin at first process again when awakened.

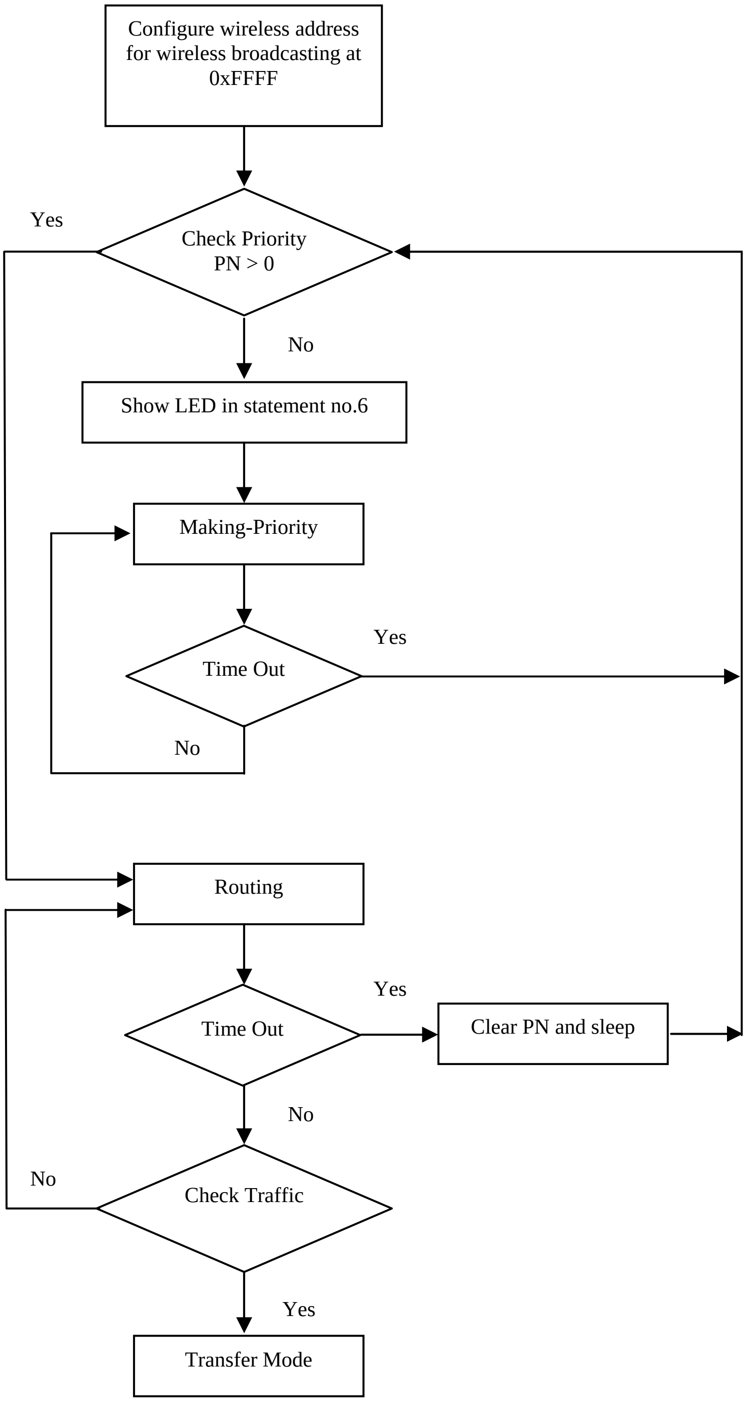


Figure 3.46 Algorithm of dynamic routing mode in sensor node (A3)

Programming Algorithm of Dynamic Routing Mode for a Gateway Node (A4 in GN)

In figure 3.47, this algorithm has the process as below.

1. The first process configures the TRW2.4G to wireless broadcasting, in which the TRW2.4G uses the address at 0xFFFF and initializes the dynamic routing packet by initial value.
2. The next process, task will check the query from SNs. If a message is received, the GN continues into the transfer mode. If the message is not received yet, the task will broadcast the priority number at zero value and repeat the loop.

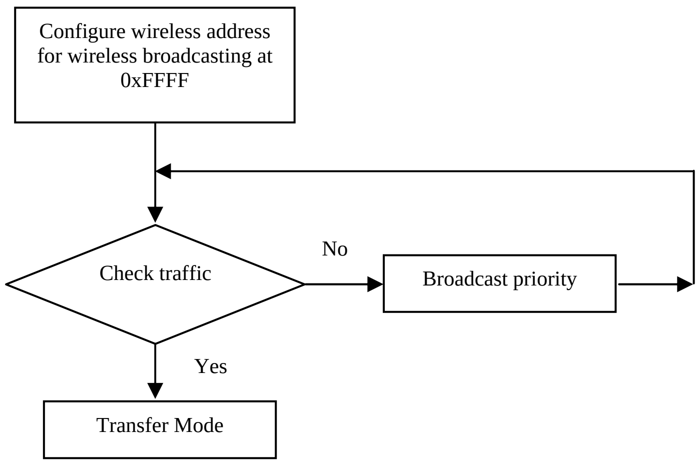


Figure 3.47 Algorithm of dynamic routing mode in gateway node (A4)

Algorithm Programming of Transfer Mode (A5)

Transfer packet, which is shown in figure 3.48, has 12-byte size. It will be used in transferring the data from sensor and battery status of sensor nodes. The detail of this packet can be shown as follows:

- Start (Leftmost): has 1 byte, used in starting this algorithm at value 0xC3.
- Command: has 1 byte, used in selecting type of this packet.
- Control: has 2 bytes, used in controlling the encryption process and the decryption process.
- Address: has 2 bytes, used in indicating the current address of the node device.
- Packet number: has 2 bytes, used for indicating the current packet number of the node device during data transfer.
- Data: has 3 bytes, used for storing the analog data and the battery status.
- End (Rightmost): has 1 byte, use in ending this algorithm at value 0x3C.

Leftmost						**Rightmost**
Start	Command	Control	Address	Packet number	Data	End
1 byte	1 byte	2 byte	2 bytes	2 bytes	3 bytes	1 byte

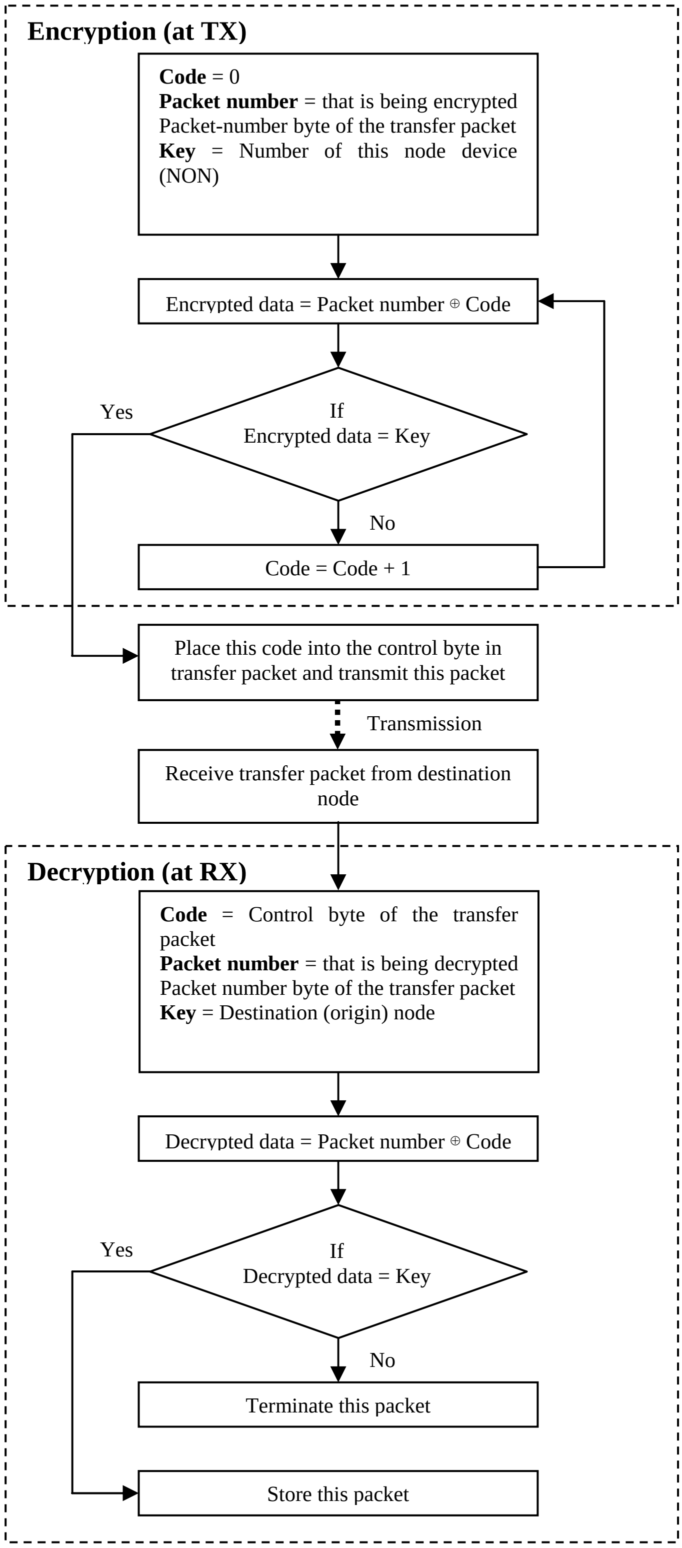
Figure 3.48 Transfer packet

For data encryption and decryption as shown in figure 3.49, these processes have three relevant variables that consist of Code, Packet number, and Key. Detail of the variables and the encryption and decryption formula can be shown as follows:

- **Code** variable has 2-byte size, and will be used in encryption and decryption formula. It is contained in the Control byte of the transfer packet.
- **Packet number** variable has 2-byte size, and will be used in encryption and decryption formula. It is contained in the Packet number byte of the transfer packet.
- **Key** variable has 2-byte size, and will be use in checking encrypted and decrypted data. For transmission process, it is the NON. For reception process, it is destination node from dynamic routing mode.
- **Encryption formula** is encrypted data = Packet number $\oplus$ Code.
- **Decryption formula** is decrypted data = Packet number $\oplus$ Code.

* $\oplus$ is logic match that implement an exclusive or (XOR)

The encrypted and decrypted data do not use the data in the transfer packet for the encryption and decryption of data. In here, the code is the encrypted data (transmission process) or the decrypted data (reception process). The data 3-byte size in the transfer packet will be used in next stage after the decryption is complete. The encrypted data (the code in transmission process) will be decrypted in the reception process. This method, which our system uses many packets that make high communication, is used for screening incorrect data.



Example in encryption and decryption

At SN1 (TX)
NON: 1
Packet stored: 2 (1,10)
At SN2 (RX)
NON: 2
Packet stored: -

The SN1 has 2 packets stored. It must transmit two processes by beginning at packet of 1 and ending at packet of 10.

First process

Encryption of SN1 (TX)
Packet number: 1
Key: 1
Code: 0
0000 0001 (Packet)
⊕
0000 0000 (Code)
0000 0001 (Key)
Encryption is complete.

Decryption of SN2 (RX)
Code: 0
Packet number: 1
Key: 1
0000 0001 (Packet)
⊕
0000 0000 (Code)
0000 0001 (Key)
Decryption is complete.

Second process

Encryption of SN1 (TX)
Packet number: 10
Key: 1
Code: 11
0000 1010 (Packet)
⊕
0000 1011 (Code)
0000 0001 (Key)
Encryption is complete.

Decryption of SN2 (RX)
Code: 11
Packet number: 10
Key: 1
0000 1010 (Packet)
⊕
0000 1011 (Code)
0000 0001 (Key)
Decryption is complete.

Figure 3.49 Encryption and decryption process

For the Encryption Process, the Code variable will be zero, the Packet number byte variable will be the current Packet number in transfer packet, and the Key variable will be the NON of this packet. Next process is to do the encryption formula. If encrypted data equals the Key, the encrypted data or the Code variable will be written into this packet in same as above position. If encrypted data does not equal the Key, the Code variable will be added one value, and process is repeated.

For the Decryption Process, the Code variable will be the current Control byte in transfer packet, the Packet number variable will be the current Packet number byte in transfer packet, and the Key variable will be the destination node from dynamic routing mode. Next process is to do the decryption formula. If decrypted data equal the Key, the data in this packet will be stored into PIC's RAM. If decrypted data does not equal the Key, this packet will be terminated.

LSB						MSB
0xC3	-	-	NON	-	-	0x3C
1 byte	1 byte	2 byte	2 bytes	2 bytes	3 bytes	1 byte

Figure 3.50 Initial values of transfer packet

Before running task in this mode, we need to initialize parameter of TRW2.4G configuration to the number of node device (NON) as the initial value of transfer packet as shown in figure 3.50. This algorithm (see in figure 3.51) will be used for both SNs and GN as the explanation below.

1. The first process, the algorithm of transfer mode begins with configuration of wireless address of TRW2.4G to be the NON. This mode can divide into two processes, which are the **Transmission process** and the **Reception process**.
2. In case of transmission process, the first process is to get the analog data from sensor. This case will only be used for the SN. The next process will check all of buffer data in PIC's RAM. Those buffers are data from other SNs. If that buffer is empty, the task will check the next buffer to the last buffer. If that buffer is not empty, the data in that

buffer will be encrypted in encryption process and transmitted to destination node (determined by the dynamic routing mode) via wireless. The final process is to transmit the confirmation data to the destination node via wireless.

3. In case of reception process, the task will begin when the transfer packet is received from the destination node (determined by the dynamic routing mode). The next process is to check for incoming packet. If that packet is the confirmation data, the task will be terminated. If that packet is buffer data, the task will enter to the decryption process for decrypting that packet. The decrypted data will be stored in PIC's RAM. This case will be used for both SNs and GN. The timeout must be used in this case because the task has the chance to enter infinite loop when the reception process has mistake.

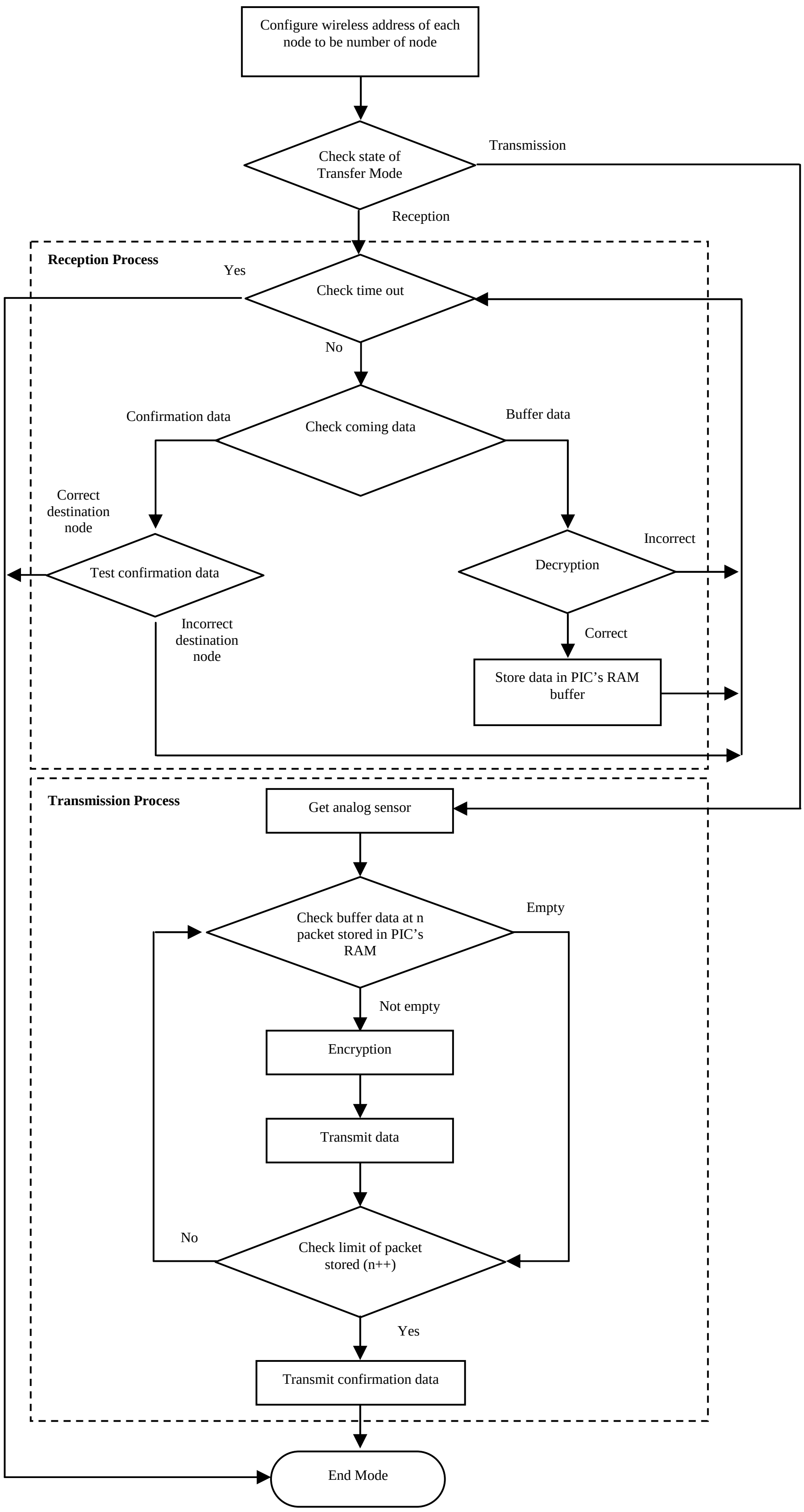


Figure 3.51 Algorithm of transfer mode (A5)

Programming Algorithm of Sensor Node (A6)

This algorithm, which is to combine with the above algorithm, can be shown in figure 3.52. It will be used when the node device is assigned to be a sensor node. This algorithm only has wireless communication. The task will restart at point A or the first process of WSN system (figure 3.39) when the task finishes or awakens from sleep state.

1. The first process of this algorithm is to run algorithm of dynamic routing mode in sensor node for receiving the destination node and type of transfer process in transfer mode. If the timeout occurs during this process, the old priority number will be cleared and the task will be entered to sleep state.
2. The second process is to run transfer mode when the task receives the destination node and type of transfer process. In case of transmission process, the task will be run into transfer mode. The task will enter to sleep state when this mode finishes. In case of reception process, the timeout will be used. If the timeout occurs, the task will enter to sleep state. The task will go to first process again when the task is complete in the transfer mode.
3. The last process is to enter sleep state. During this algorithm, the interrupt process can be called by pushing the setting switch.

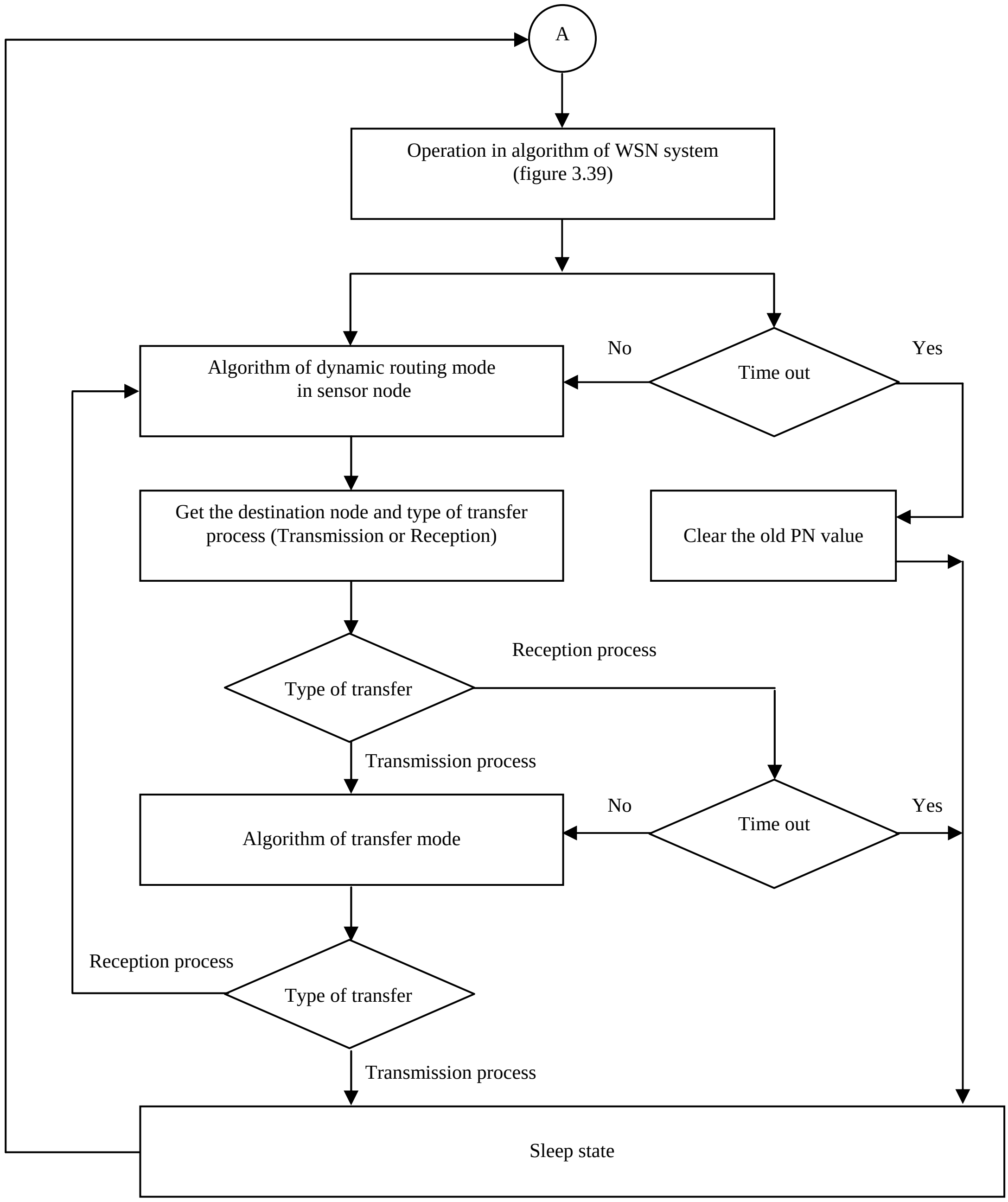


Figure 3.52 Algorithm of sensor node (A6)

Programming Algorithm of Gateway Node (A7)

This algorithm, which is to combine with the above algorithm, can be shown in figure 3.53. It will be used when the node device is assigned to be a gateway node. This algorithm has wireless and SPI communication. The task will run in this infinite loop. It can be stopped by interrupt.

1. The first process is to run algorithm of dynamic routing mode in gateway node for receiving the destination node. The transfer process in the transfer mode is only reception process. In the first process, the timeout will not be used.
2. The second process is to run algorithm of transfer mode in reception process. This process will run when the task receives the destination node. The timeout will be used.
3. The last process is to transfer all buffer data in PIC's RAM to the host device via SPI communication and go to the first process again.

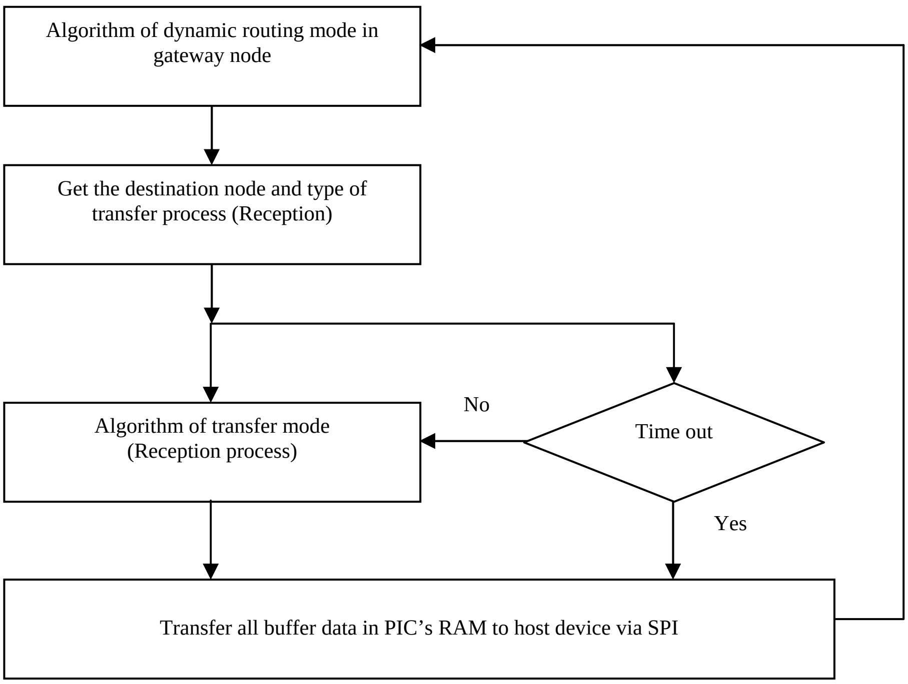


Figure 3.53 Algorithm of gateway node (A7)

3.3.7 Example of WSN System in the Algorithm

In this example, we will use the same example as in figure 3.2 for explanation. For the brief presentation, we will only show the path of data in case C of figure 3.16, which arbitrarily selected for explanation as shown in figure 3.54. All SNs can sleep and send data to host (all nodes are functional). Each algorithm takes about microseconds (us) to milliseconds (ms). But the time in sleep state takes at least 1 second. We can explain the uptime in WSN system in topic 3.4. The example of WSN system explains the algorithms as follow:

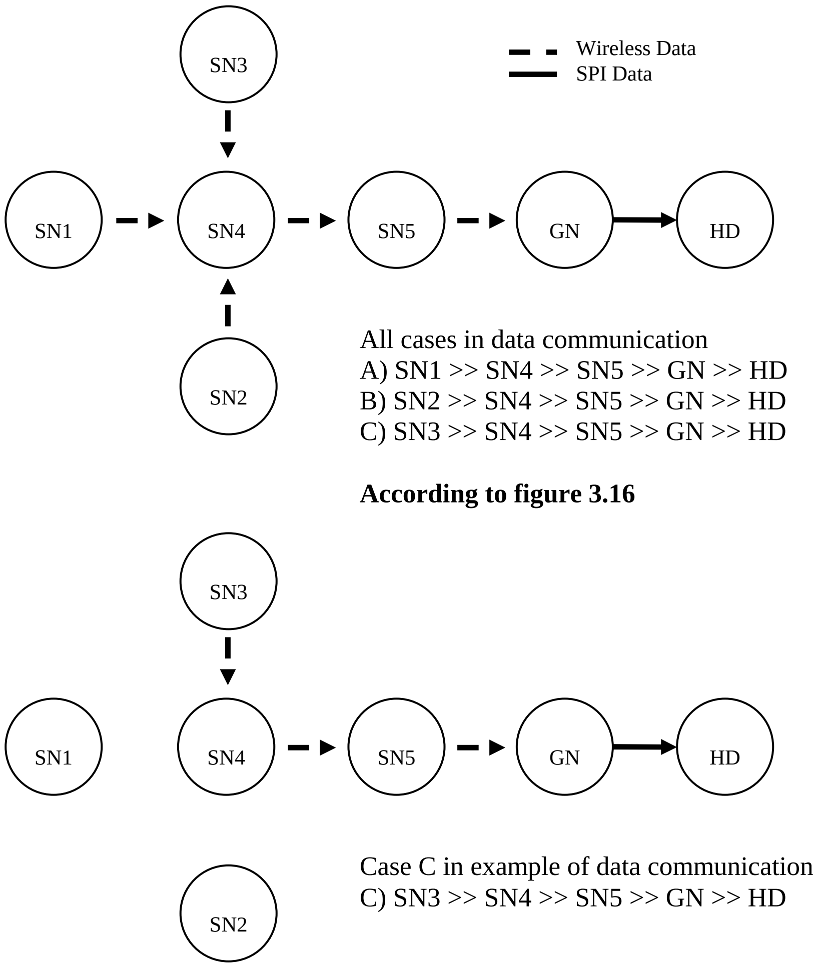
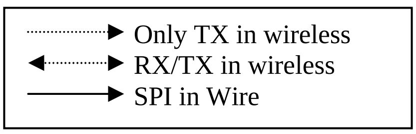
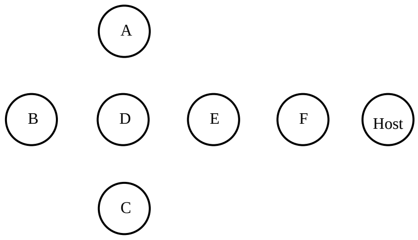


Figure 3.54 An example of data communication in WSN system (only case C)

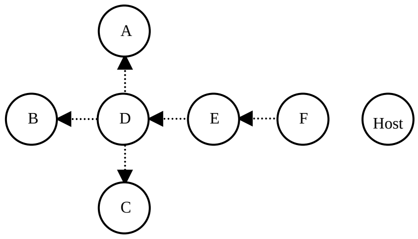




Node	Type	NON	Priority	Address	Destination	State
A	SN	1	0	-	-	Installed
B	SN	2	0	-	-	Installed
C	SN	3	0	-	-	Installed
D	SN	4	0	-	-	Installed
E	SN	5	0	-	-	Installed
F	GN	6	1	-	-	Installed

1. Assign task and install node device

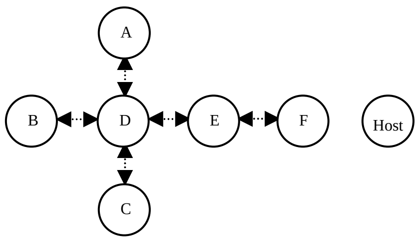
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Assigning task	Assigning task	Assigning task	Assigning task	Assigning task	Assigning task
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	-	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	Receive PN
B	SN	2	4	0xFFFF	0xFFFF	Receive PN
C	SN	3	4	0xFFFF	0xFFFF	Receive PN
D	SN	4	3	0xFFFF	0xFFFF	Receive PN
E	SN	5	2	0xFFFF	0xFFFF	Receive PN
F	GN	6	1	0xFFFF	0xFFFF	Dispense PN

2. Run making-priority for dispensing and receiving the PN together.

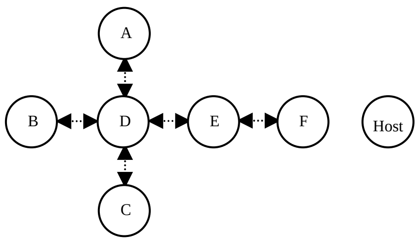
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	-	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	First searching
B	SN	2	4	0xFFFF	0xFFFF	First searching
C	SN	3	4	0xFFFF	0xFFFF	First searching
D	SN	4	3	0xFFFF	0xFFFF	First searching
E	SN	5	2	0xFFFF	0xFFFF	First searching
F	GN	6	1	0xFFFF	0xFFFF	Searching

3. Run routing process for searching node

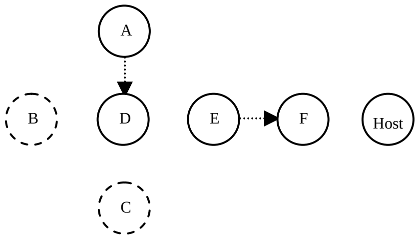
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	-	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	Found D
B	SN	2	4	0xFFFF	0xFFFF	First researching
C	SN	3	4	0xFFFF	0xFFFF	First researching
D	SN	4	3	0xFFFF	0xFFFF	Found A
E	SN	5	2	0xFFFF	0xFFFF	Found F
F	GN	6	1	0xFFFF	0xFFFF	Found E

4. If the destination node is found, the task will go to next process. If the task does not find the destination node, it will re-search for nodes.

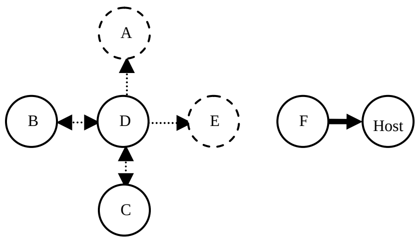
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	-	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	1	4	Transmission
B	SN	2	-	-	-	Clear PN and sleep
C	SN	3	-	-	-	Clear PN and sleep
D	SN	4	3	4	1	Reception
E	SN	5	2	5	6	Transmission
F	GN	6	1	6	5	Reception

5. Node A, D, E and F will run to transfer mode. Node B and C will be cleared the old PN and go to sleep state because timeout occurs.

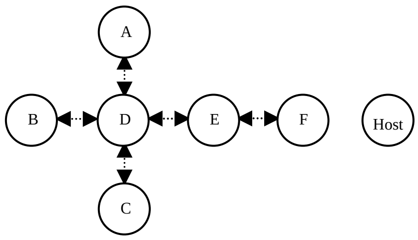
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Transfer mode	Sleep	Sleep	Transfer mode	Transfer mode	Transfer mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
A	-	-	D,A	E	E



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	-	-	Sleep
B	SN	2	4	0xFFFF	0xFFFF	Receive PN
C	SN	3	4	0xFFFF	0xFFFF	Receive PN
D	SN	4	3	0xFFFF	0xFFFF	Second searching
E	SN	5	2	-	-	Sleep
F	GN	6	1	6	Host	Transmit to host

6. Node A and E will go to sleep state. Node F is transmitting all buffer data to host. Node B and C is receiving the new PN. Node D will go to dynamic routing again for searching node.

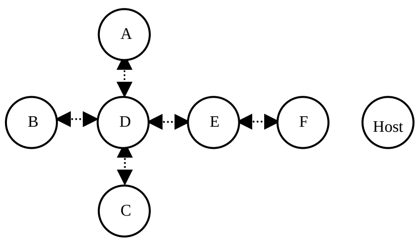
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Sleep	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Sleep	Transfer mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	D,A	-	E



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	First searching
B	SN	2	4	0xFFFF	0xFFFF	First searching
C	SN	3	4	0xFFFF	0xFFFF	First searching
D	SN	4	3	0xFFFF	0xFFFF	Second researching
E	SN	5	2	0xFFFF	0xFFFF	First searching
F	GN	6	1	0xFFFF	0xFFFF	Searching

7. All nodes are running in the routing process.

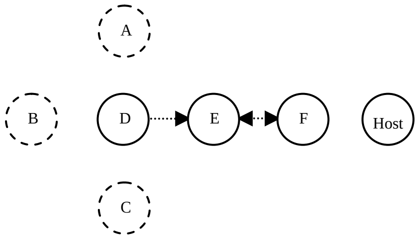
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	D,A	-	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	First researching
B	SN	2	4	0xFFFF	0xFFFF	First researching
C	SN	3	4	0xFFFF	0xFFFF	First researching
D	SN	4	3	0xFFFF	0xFFFF	Found E
E	SN	5	2	0xFFFF	0xFFFF	Found D
F	GN	6	1	0xFFFF	0xFFFF	Searching

8. All nodes are running in the routing process. Node D and E found together.

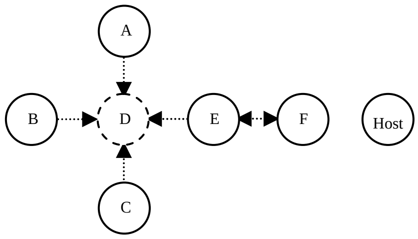
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	D,A	-	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	-	-	-	Clear PN and sleep
B	SN	2	-	-	-	Clear PN and sleep
C	SN	3	-	-	-	Clear PN and sleep
D	SN	4	3	4	5	Transmission
E	SN	5	2	5	4	Reception
F	GN	6	1	0xFFFF	0xFFFF	Searching

9. Node A, B and C will be cleared the old PN and go to sleep state because they has the timeout. Node D and E is transferring buffer data together. Node F is searching node.

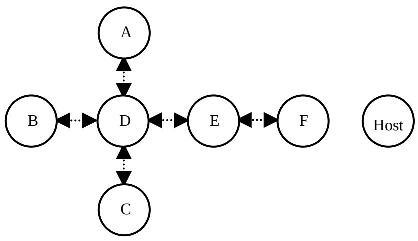
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Sleep	Sleep	Sleep	Transfer mode	Transfer mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	D,A	E,D,A	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	-	0xFFFF	0xFFFF	Waiting PN
B	SN	2	-	0xFFFF	0xFFFF	Waiting PN
C	SN	3	-	0xFFFF	0xFFFF	Waiting PN
D	SN	4	3	-	-	Sleep
E	SN	5	2	0xFFFF	0xFFFF	Second searching
F	GN	6	1	0xFFFF	0xFFFF	Searching

10. Node A, B and C is waiting for the new PN from node D because node D is sleeping now. Node E and F is continuing in searching node.

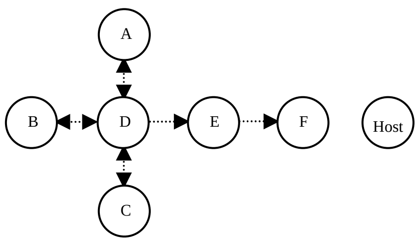
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Sleep	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	E,D,A	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	Receive PN
B	SN	2	4	0xFFFF	0xFFFF	Receive PN
C	SN	3	4	0xFFFF	0xFFFF	Receive PN
D	SN	4	3	0xFFFF	0xFFFF	First searching
E	SN	5	2	0xFFFF	0xFFFF	Found F
F	GN	6	1	0xFFFF	0xFFFF	Found E

11. Node A, B and C receive the new PN from node D. Node D is searching for nodes. Node E and F found each other.

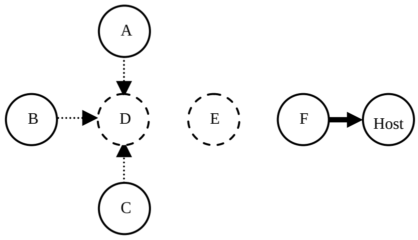
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	E,D,A	-



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	First searching
B	SN	2	4	0xFFFF	0xFFFF	First searching
C	SN	3	4	0xFFFF	0xFFFF	First searching
D	SN	4	3	0xFFFF	0xFFFF	First researching
E	SN	5	2	5	6	Transmission
F	GN	6	1	6	5	Reception

12. Node A, B, C and D is continuing in searching node. Node E and F is transferring buffer data together.

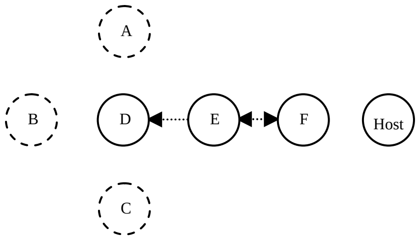
Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Transfer mode	Transfer mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	E,D,A	E,D,A



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	4	0xFFFF	0xFFFF	First researching
B	SN	2	4	0xFFFF	0xFFFF	First researching
C	SN	3	4	0xFFFF	0xFFFF	First researching
D	SN	4	-	-	-	Clear PN and sleep
E	SN	5	2	-	-	Sleep
F	GN	6	1	6	Host	Transmit to host

13. Node A, B and C is searching for nodes again. Node D has timeout that this node must clear the old PN and enter sleep state. Node E is sleeping when transmission process is complete. Node F transmits all data to host.

Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Dynamic routing mode	Dynamic routing mode	Dynamic routing mode	Sleep	Sleep	Transfer mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	-	E,D,A



Node	Type	NON	Priority	Address	Destination	State
A	SN	1	-	-	-	Clear PN and sleep
B	SN	2	-	-	-	Clear PN and sleep
C	SN	3	-	-	-	Clear PN and sleep
D	SN	4	3	0xFFFF	0xFFFF	Receive PR
E	SN	5	2	0xFFFF	0xFFFF	First searching
F	GN	6	1	0xFFFF	0xFFFF	Searching

14. Continue in own task of each node.

Algorithm of each node					
A	**B**	**C**	**D**	**E**	**F**
Sleep	Sleep	Sleep	Dynamic routing mode	Dynamic routing mode	Dynamic routing mode
Buffer data of each node					
A	**B**	**C**	**D**	**E**	**F**
-	-	-	-	-	-

3.4 Power Consumption of Sensor Node

Only sensor nodes will be explained about the power consumption because SNs use energy source from battery that has limited lifetime. The gateway node will not be explained because it uses energy source from host device or infinite source. The composition of calculation in power consumption consists of the uptime for each process and the current used of each process.

3.4.1 Overall and Calculation of the Uptime

The uptime of sensor node is shown in figure 3.55. The uptime in sensor node can be classified into the uptime of sender and repeater. For the previous example, which has 3 hops (SN1 > SN2 > SN3 > GN), a communication sequence is shown in figure 3.56. We can see that the sensor node in repeater case uses the uptime almost double the amount used in sender case. This uptime amount will be confirmed in the test.

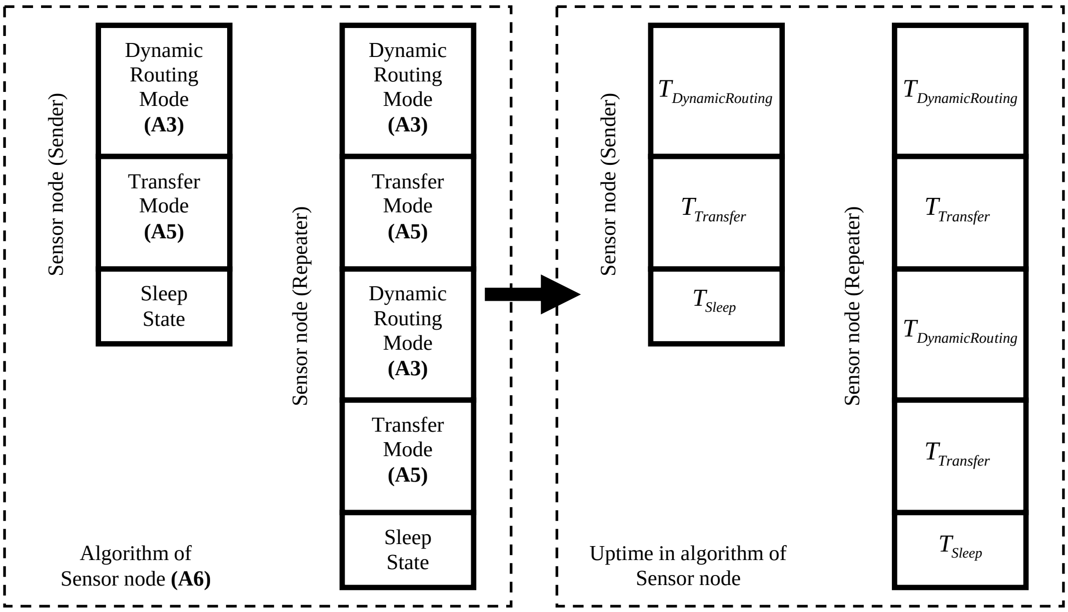


Figure 3.55 The uptime in algorithm of sensor node on function box

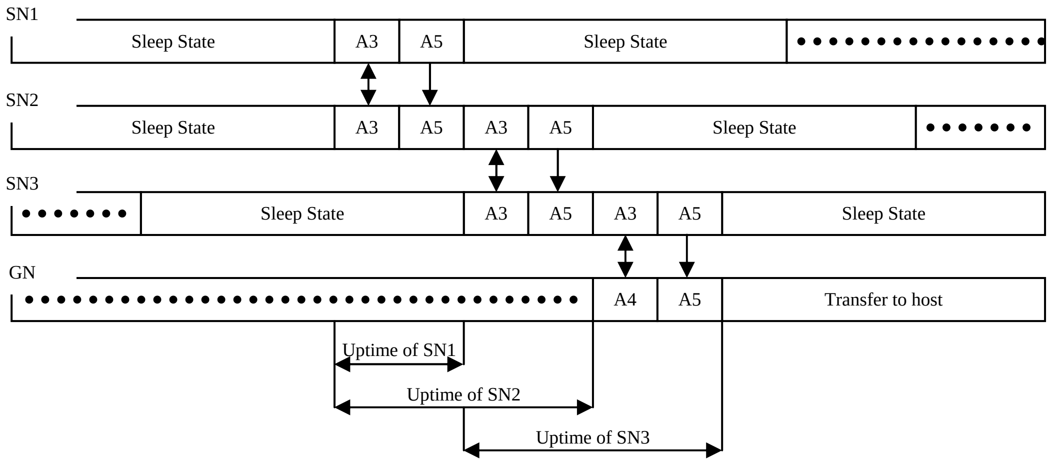


Figure 3.56 A communication sequence in the example of 3 SNs and a GN

Now, we have discussed the parameter, the algorithm, and various event of WSN system relating to uptime. Next, the uptime can be completed by combining the following functions:

- Set TX is the function for setting sensor node as a transmitter.
- Set RX is the function for setting sensor node as a receiver.
- TX is the function to transmit data via wireless.
- RX is the function to receive data via wireless.

Table 3.7 Time parameters of functions in WSN system

Function (use 1 time)	Time (ms)	
	Maximum	Minimum
Configuration Word	3.126	-
Set TX	0.008	-
Set RX	0.416	-
TX	0.513	-
RX	1.097	0.098

The table 3.7 shows basic functions of wireless communication in the WSN system. These functions are a part of many functions, such as the making-priority function, the routing function. The time parameters of these functions are determined by datasheet of the TRW2.4G and our algorithm.

Uptime in Dynamic Routing Mode

The wireless broadcasting function (see in topic 3.3.3) will be used for dynamic routing mode. The wireless broadcasting communication is to broadcast the wireless data to surrounding area by installation of the same address to all nodes. All nodes can communicate among them when they are installed already. The configuration word will always be used once in the wireless broadcasting function.

1 time	1 time	Data 1 time	1 time	Data 1 time		1 time	Confirm 10 times
Configuration Word	Set TX	TX	Set RX	RX	TX/RX >>	Set TX	TX
Configuration to Broadcast Mode	TX/RX routine (1 time)				Additional TX/RX routines (*m* - 1 times)	If complete	

Figure 3.57 Wireless broadcasting function
(which is the same as figure 3.27)

In figure 3.57, the wireless broadcasting function includes of m times for transmitting and receiving data (TX/RX routine m times), and 10 times for transmitting the confirmation data if this function is complete. The value of m is determined by the timeout in range of 1 – 1000 (This timeout is determined by the optimization of broadcasting time in chapter 4).

Table 3.8 Uptime in dynamic routing mode

Function		Time (ms)	
		Maximum	**Minimum**
Config	Configuration Word (1 time)	3.126	-
m Standard	Set TX (1 time)	0.008	-
	Set RX (1 time)	0.416	-
	TX (1 time)	0.513	-
	RX (1 time)	1.097	0.098
Confirm	Set TX (1 time)	0.008	-
	TX (10 times)	5.13	-
Total uptime		8.264+($m \times 2.034$)	8.264+($m \times 1.035$)

**m* = number of TX/RX routine, such as if *m* is one routine, the uptime of dynamic routing mode will be 10.298 (maximum) or 9.299 (minimum) ms

From table 3.8, the uptime values in this table are determined by table 3.7. We will see the total uptime of dynamic routing mode, which is divided into two ranges (maximum and minimum). Both ranges are determined by the uptime of RX, whose value ranges 0.0098–1.097 millisecond. Thus, we can define as follows.

$$T_{DynamicRouting} = 8.264 + (m \times 2.034) \text{ millisecond (maximum value).}$$

$$T_{DynamicRouting} = 8.264 + (m \times 1.035) \text{ millisecond (minimum value).}$$

Uptime in Transfer Mode

The transfer mode, which is an algorithm in WSN system, consists of the transmission process and the reception process (see in algorithm topic 3.3.6). The transmitter mode function will be used for the transmission process. The receiver mode function will be used for the reception process. These functions can be seen in topic 3.3.3. The configuration word will always be used once (which is similar to the wireless broadcasting function).

Configuration to Transfer Mode		n Packet	Confirm
1 time	1 time	n time	10 times
Configuration Word	Set TX	TX	TX

Figure 3.58 Transmitter mode function in transmission process (which is similar with figure 3.25)

Configuration to Transfer Mode		n Packet	Confirm
1 time	1 time	n time	10 times
Configuration Word	Set RX	RX	RX

Figure 3.59 Receiver mode function in reception process (which is similar with figure 3.26)

In figure 3.58, the transmission function includes n time for transmitting the n data packet, and 10 times for transmitting the confirmation data (if the data transmission is complete). In figure 3.59, the reception function includes of n time for receiving the n data packet, and at most 10 times for receiving the confirmation data (if this function receives the confirmation data from transmission process). The value of n is determined by maximum number of sensor node in range of 1 – 100 (100 for our system).

Table 3.9 Uptime of transmission process in transfer mode

Function		Time (ms)	
		Maximum	**Minimum**
Config	Configuration Word (1 time)	3.126	-
	Set TX (1 time)	0.008	-
n packets	TX (1 time)	0.513	-
Confirm	TX (10 times)	5.13	-
Total uptime		8.264+($n \times$ 0.513)	-

*n = number of data packet, such as if the transmitter node has one data packet, the uptime of transmission process will be 8.777 (maximum only) ms

Table 3.10 Uptime of reception process in transfer mode

Function		Time (ms)	
		Maximum	**Minimum**
Config	Configuration Word (1 time)	3.126	-
	Set RX (1 time)	0.416	-
n packets	RX (1 time)	1.097	0.098
Confirm	RX (10 times)	1.097	0.098
Total uptime		14.512+($n \times$ 1.097)	4.522+($n \times$ 0.098)

*n = number of data packet, such as if the receiver node has one data packet coming, the uptime of reception process will be 15.609 (maximum) or 4.62 (minimum) ms

From table 3.9 and 3.10, the uptime values in this table are determined by table 3.7. We will see the total uptime of reception process, which is divided into two ranges (maximum and minimum). Both ranges are determined by the uptime of RX, whose value ranges 0.0098–1.097 millisecond. But the transmission process only has one value. Thus, we can define as follows.

$$T_{Transmission} = 8.264 + (n \times 0.513) \text{ millisecond (maximum value)}$$

$$T_{\mathrm{Re}ception} = 14.512 + (n \times 1.097) \text{ millisecond (maximum value)}$$

$$T_{\mathrm{Re}ception} = 14.512 + (n \times 0.098) \text{ millisecond (minimum value).}$$

Calculation of the Uptime of Sensor Node

The uptime in sensor node is necessary for calculation of power consumption. We can show the formula of uptime as follows:

$$T_{Sender} = T_{DynamicRouting} + T_{Transmission} + T_{Sleep} --- (1)$$

$$T_{Repeater} = (2 \times T_{DynamicRouting}) + (T_{Transmission} + T_{Reception}) + T_{Sleep} --- (2)$$

$$8.264 + (m \times 2.034) \geq T_{DynamicRouting} \geq 8.264 + (m \times 1.035) --- (3)$$

$$T_{Transmission} = 8.264 + (n \times 0.513) --- (4)$$

$$14.512 + (n \times 1.097) \geq T_{Reception} \geq 14.512 + (n \times 0.098) --- (5)$$

- T_{Sender} is overall uptime of sensor node in sender case (Second).
- $T_{Repeater}$ is overall uptime of sensor node in repeater case (Second).
- $T_{DynamicRouting}$ is the uptime of dynamic routing mode (Second).
- $T_{Transmission}$ is the uptime of transmission process in transfer mode (Second).
- $T_{Reception}$ is the uptime of reception process in transfer mode (Second).
- T_{Sleep} is the uptime in the sleep state (Second).
- m is the number of standard function in dynamic routing mode. (range of this value is 1 – limit of timeout or 1 - 1000)
- n is the number of data packet in transfer mode. (range of this value is 1 - max sensor node or 1 – 100)

Table 3.11 Required values for calculation of each uptime of sensor node

Uptime in sensor node	Required value
T_{Sender}	$T_{DynamicRouting}$, $T_{Transmission}$, T_{Sleep} (Calculation)
$T_{Repeater}$	$T_{DynamicRouting}$, $T_{Transmission}$, $T_{Reception}$, T_{Sleep} (Calculation)
$T_{DynamicRouting}$	m (Testing)
$T_{Transmission}$	n (Testing)
$T_{Reception}$	n (Testing)
T_{Sleep}	Define by user

3.4.2 Calculation of the Current Used and Power Consumption

The current used in each state will be determined by the testing in chapter 4 as well as the uptime value (see in table 3.12).

Calculation of the Current Used

The current used in sensor node is necessary in calculation of power consumption. We can show the formula of the current used as follows:

$$\overline{I_{Sender}} = \frac{(I_{Broadcast} \times T_{DynamicRouting}) + (I_{Transmission} \times T_{Transmission}) + (I_{Sleep} \times T_{Sleep})}{T_{Sender}} --- (6)$$

$$\overline{I_{Repeater}} = \frac{(2 \times I_{Broadcast} \times T_{DynamicRouting}) + (I_{Reception} \times T_{Reception}) + (I_{Transmission} \times T_{Transmission}) + (I_{Sleep} \times T_{Sleep})}{T_{Repeater}} --- (7)$$

- $I_{Transmission}$ is the current used in the sensor node in the transmission process of the transfer mode (A).
- $I_{Reception}$ is the current used in the sensor node in the reception process of the transfer mode (A).
- I_{Sleep} is the current used in the sensor node in the sleep state (A).
- $I_{Broadcast}$ is the average current used in the sensor node in the broadcast mode (A).
- $\overline{I_{Sender}}$ is average current used in sensor node in sender case (A).
- $\overline{I_{Repeater}}$ is average current used in sensor node in repeater case (A).

$I_{Transmission}$, $I_{Reception}$, $I_{Broadcast}$ and I_{Sleep} will be averaged by gathering these values from testing for finding the correct values.

The Calculation of Power Consumption

The final calculation of power consumption is to calculate the lifetime of sensor node in use. We can show the formula of power consumption used as follows:

$$LifeTime_{Sender} = \frac{C_{Battery}}{\overline{I_{Sender}} \times 24} --- (8)$$

$$LifeTime_{Repeater} = \frac{C_{Battery}}{\overline{I_{Repeater}} \times 24} --- (9)$$

- $LifeTime_{Sender}$ is the lifetime of sensor node in sender case (Day).
- $LifeTime_{Repeater}$ is the lifetime of sensor node in repeater case (Day).
- $C_{Battery}$ is the battery capacity (AHr).

The various variables in calculating power consumption will be defined in the next chapter of testing.

Table 3.12 Required values for calculation of current used of sensor node

Uptime in sensor node	Required value
$\overline{I_{Sender}}$	$T_{DynamicRouting}$, $T_{Transmission}$, T_{Sleep} $I_{Broadcast}$, $I_{Transmission}$, I_{Sleep} (Calculation)
$\overline{I_{Repeater}}$	$T_{DynamicRouting}$, $T_{Transmission}$, $T_{Reception}$, T_{Sleep} $I_{Broadcast}$, $I_{Transmission}$, $I_{Reception}$, I_{Sleep} (Calculation)
$I_{Broadcast}$	Testing
$I_{Transmission}$	Testing
$I_{Reception}$	Testing
I_{Sleep}	Testing

3.5 Host Device

This base station is the final destination of a wireless sensor network. HD will be connected with one ND, which works as a GN, by micro-USB via SPI protocol. It also controls installing WSN parameter, and receives information by connecting with the GN. In addition to communication with NDs, it is also a data logger and a web-server. Furthermore, the setting of WSN parameter on this device can be done by TFT touch screen. Since our system need to run many tasks several times, the HD must be fast in processing and need a lot of RAM and Flash memory in programming, we chose LPC1678 of ARM family on on-the-shelf ET-NXP-ARM-KIT-LPC1768 board for the processing. That board comes ready to work and also has a few problems in the design that must create additional part to meet requirements of the work. This board is developed by keil uVision3 software.

(a)

(b)

Figure 3.60 The design in host device

(a) ET-NXP-ARM-KIT-LPC1768 board ARM microcontroller

(b) Host device complete

Figure 3.61 Interfaces of host device

3.5.1 Tasks of Host Device

Multi-tasking in HD comprises of three main tasks as seen in figure 3.62. Web server, which is the first task, must be configured its TCP/IP before use by initializing IP Address, Subnet Mask and Default Gateway values. After that, the HD will be able to connect with LAN network for Host Server. It will wait for requests of clients within limited time. But before running as a server, it must be configured for date time, check SD card, and check the GN installation. The second task is data logger, which use Embedded File Systems Library (EFSL) in creating files on SD card. The HD will do this task every 10 seconds when it receives data from the GN via SPI protocol for saving information of data in bit value into SD card. If the HD does not receive that data, it will save zero value instead. The final task is to display TFT GLCD touch screen that is used for setting parameters, such as configuration of date time and TCP/IP in the HD, setting of WSN parameter for installation to ND, controlling the server of data logger, and the display of task acknowledgment. In the display of task acknowledgment, we can preview, such as confirming in any configuration, SD card memory checking, and showing of current file of data logger in SD card. In addition to those tasks, real time clock (RTC) will be used for clock source and data logger's date and time. Those tasks do not work at the same time, but they work on time sharing in the design system because the HD has one processing core. It can do one task at a time. Since a lot of tasks and complex statement, we designed and tested those tasks together. To connect the network by TCP/IP, we will use LAN-cable, whose signal status can be checked from green LED at the side of LAN-socket.

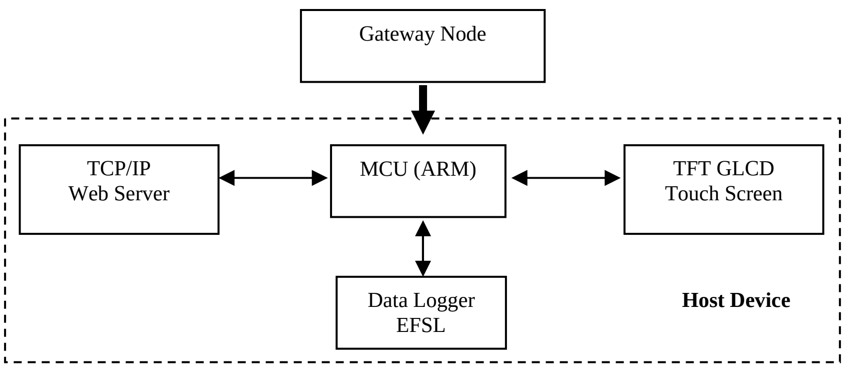


Figure 3.62 Working task of host device in web server

3.5.2 Detail of Web Server Display

In figure 3.63, this is the webpage on internet browser. The webpage is designed by HTML code in Macromedia Dreamweaver MX 2004 program. That code is designed to be simple and small for fast operating task. The code is kept into RAM memory of ARM in character type. First section will show date and time that is installed from HD. Second section will show status of wireless signal. If the HD has coming signal from the GN, the webpage will show green color instead of black color in this section. Third section is connection counter of wireless signal, which is the number of success signal. Fourth section shows capacity used of SD card in percent. In fifth section, we can see information of one ND. The webpage will show NON, status of battery and value of analog sensor when the HD has coming data. If the battery is low, the webpage will show 'LOW'. If the battery is normal, the webpage will show 'READY'. Value of analog sensor will be shown in percent. If the HD does not have the data, all part of this section will show '-'. And finally, this section is used to set page refreshing. If button is clicked to 'AUTO', this webpage will refresh page automatically and can see text 'AUTO REFRESH PAGE'. If the button is clicked to 'CLEAR', this webpage will not refresh page automatically and can see text 'NOT REFRESH PAGE'. The CLEAR button also clears the NON and makes the all part in fifth section to '-'. And last button 'INSERT NUMBER OF NODE', it will be used in setting NON of ND that we want to see.

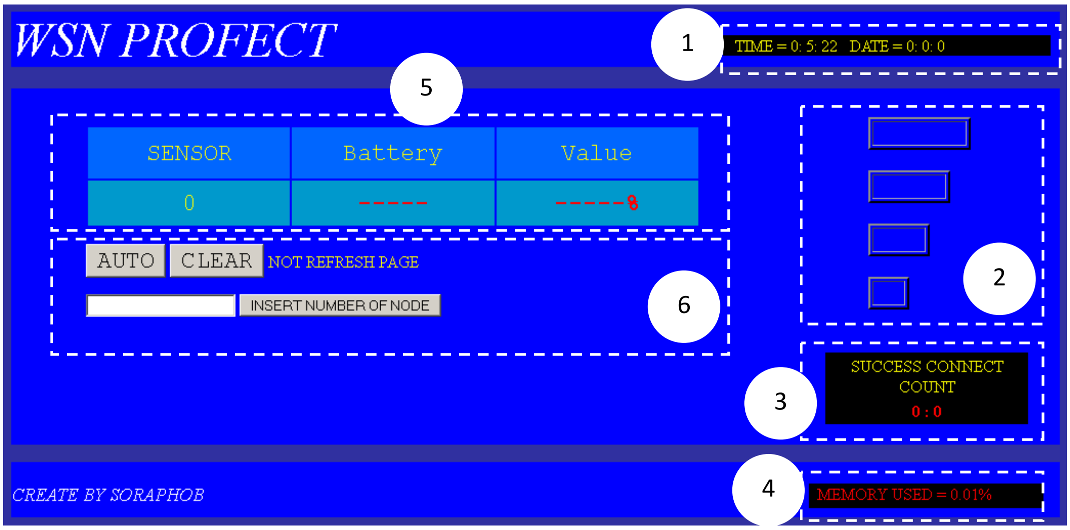


Figure 3.63 Web page display of host device

3.5.3 Hardware Design of Host Device

Various parts of PCB design is shown in figure 3.64. The designs include five parts on ET-NXP-ARM-KIT-LPC 1768 board for supporting the requirement of work. Interface component part is allocated in suitable position. Those various parts consist of connector pin for TFT GLCD, USB connector type A female for connection with GN, SD card through hole instead of micro-SD card in board for handiness, box header connector for SPI-signal connection to PCB from ET-NXP-ARM-KIT-LPC 1768 board, and inlet main power plug. The power will use adapter 5V 1A and this part has a box of fuse for protection this device from over current.

(a) (b)

Figure 3.64 Parts of PCB design in host device

(a) Front of PCB connects with ET-NXP-ARM-KIT-LPC 1768 board

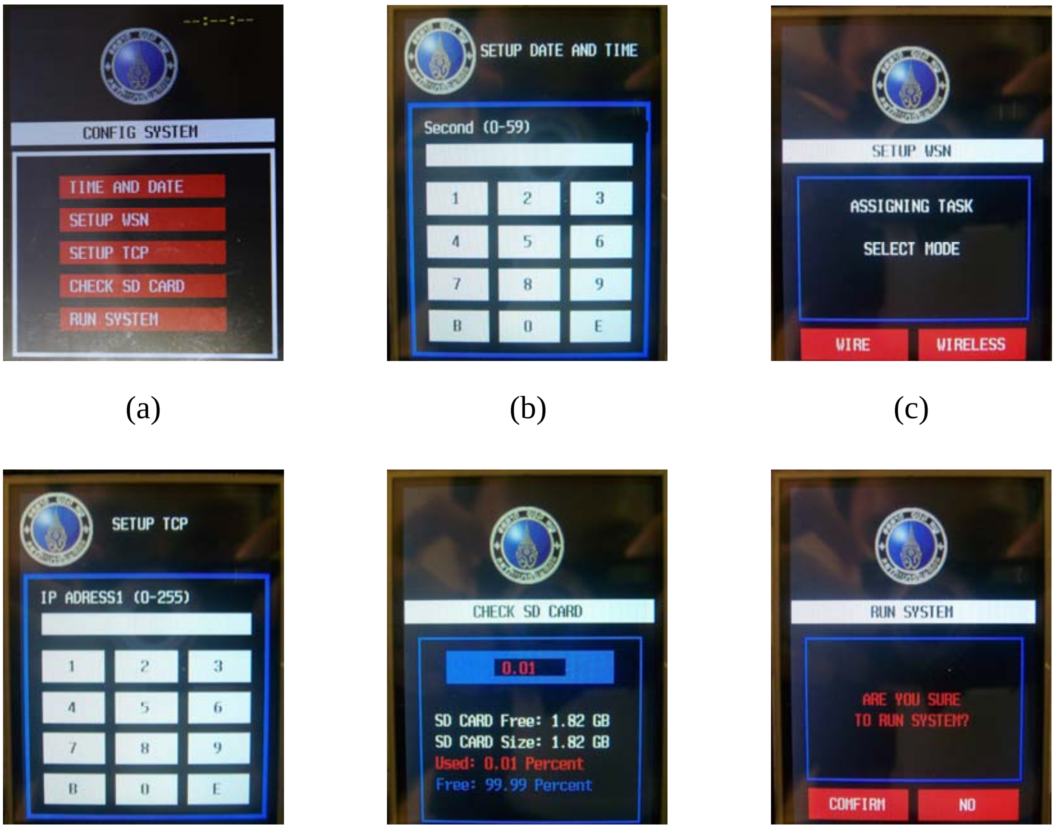
(b) Back of PCB connects with ET-NXP-ARM-KIT-LPC 1768 board

Table 3.13 Hardware costs in host device

Oder	**(THB)**
Parts	150
PCB	100
NXP-LPC1768 Board	1,850
TFT GLCD Touch Screen	1,350
Total	3,450

3.5.4 User interface Functions on Host Device

User interface function in HD includes five functions as shown in the figure 3.65. First function is Time and Date. As we can see in the figure, it appears the sign ‘--:--:--’. That means time and date has not been set. In this case, we must enter to this function for configuration. It will show sample ’00:00:01’ in hour: minute: second units which if we do not configure that, the HD will not be able to run system in the fourth function.



(a) (b) (c)

(d) (e) (f)

Figure 3.65 Touch screen display of host device in user interface function

(a) Main menu (b) Time and data function (c) Setup node device function

(d) Setup TCP function (e) Check SD card function (f) Run system function

Second function is Setup Node Device. We can choose two methods in setup which include wire and wireless method according to topic 3.3.4. We will explain all details of assigning task to node device in topic or 3.6.1. Third function is

Setup TCP. We must configure the TCP address before running system. Fourth function is Check SD Card. We can check memory capacity of SD card and completeness in SD card installation. And last function, this is Run System that can be entered when date, time and SD card is installed. In case of loss or resetting in source power, date time will return to default zero value and show in display as '--:--:--'. Since, the HD uses one clock source that is Real Time Clock (RTC) for date and time reference, the RTC will be reset when the ARM is reset, the date and time will be reset to the same as the RTC. The ARM will reset all peripherals to defaults value before use. Touch screen on the design is easy to understand and use by anyone. It requires no calibration of touch screen value before use. The touch screen can be used in two ways. The first way is one touch. The second way is to touch, then slides to the left or the right.

3.5.5 Issues in Host Device

One interesting problem in design HD is on the method of SPI protocol. At the beginning of connection between ND and HD, or PIC and ARM, which are different microcontroller families, both cannot connect together without data errors, and cause damage to peripheral from synchronization error between MCUs. This causes by the clock speed in ARM that is faster than PIC. By reducing speed in SPI operation of ARM, it works well without error. In addition to the problem in the software, it also has problem in the hardware of ARM. In early design, we tested all peripheral of ARM before use and beware anything that could cause damage to our device. But damage still occurred. We found the cause of damage is over voltage flow. The most of peripheral will be damaged from over voltage. We found that the problem is energy feeding from external peripheral to the HD. The PIC differs from ARM in endurance to over voltage in several times. In SPI connection, we take 1k Ohm resistor to connect in series between SPI peripheral of ARM and PIC. That method is to limit current and drop voltage in operation. Although ARM is better than PIC in processing, ARM is worse than PIC in endurance. So, most of our prototypes will use PIC if possible.

Another interesting problem in using HD is about SD card. The SD card is designed to use capacity that does not over than 2G bytes. We have many reasons in choosing that design, one of that is resource use in connection. It needs high speed clock in operation. If we use high speed clock in connection, it will affect overall system. We can observe abnormal events, such as over heat during connection of SD card and computer that uses high speed in connection. It is not a good thing and also shortens lifetime. Power consumption is another reason too. The connection at high speed must consume high power consumption certainly. For capacity of SD card, if it is more than 2G bytes, we will call High Capacity or show acronym as ‘SDHC’ in SD card body. The SDHC uses high speed clock in connection. For problem about SD card, we will see on display event ‘Not Install SD Card’, even when the SD card is installed already. That issue may occur from no synchronize clock in connection. This issue can be fixed by two ways. The first way is to reset power from power switch of HD. The second way is to reinstall the SD card into its slot, then reset power.

3.6 Installation and Use of WSN System

This topic will explain details in the installation method for assigning task to node devices, and the use of WSN system.

3.6.1 Installation Method for Assigning Task to Node Device

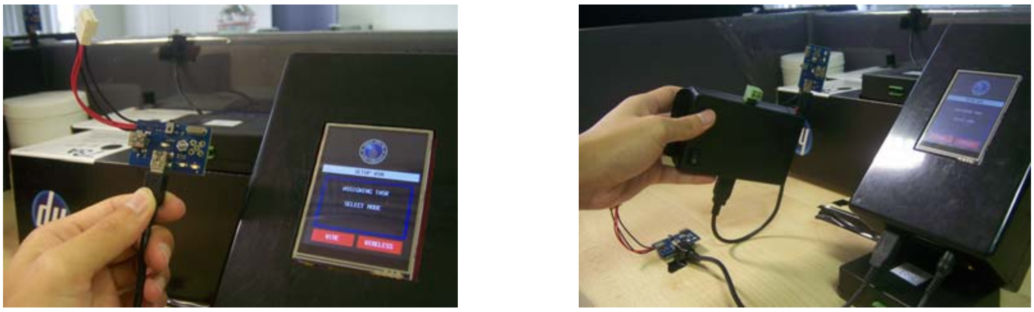
This topic consists of the connection between a node and a host device by mini-USB, and the installation of WSN parameter to node device. The detail of WSN parameter can be seen in topic 3.3.2.

To connect between node and host device, we will use mini-USB connector from the side of the ND (see in figure 3.11 a) to connect with the HD via USB type A. The Tools for assigning task includes mini-USB female of the ND, USB female type A of the HD, cable USB male type A to mini-USB, setting switch of the ND, and TFT touch screen display of the HD. We must connect USB cable before installation of WSN parameter. This connection by USB cable is used for SPI protocol.

To install WSN parameter (can see algorithm in topic 3.3.6) to node device, the values of AON, NON and FON (same for the same network) can be set by connecting them with HD. We can install those parameters on two-way methods as shown in figure 3.66 (which is similar in figure 3.28). We will know installation state from LED on the ND, in which meaning can be seen in table 3.3.

- First method is installation by wire. It uses one ND to connect with the HD. Pushing of the setting switch will enter SPI method, and will also remove all old parameters and old buffers. We can see if the connection is successful in display of the HD. And then, we will select to install by wire in the display and load WSN parameter by selecting value in the display. If the method is successful, the display will show complete text.
- Second method is installation by wireless. It has two steps and needs at least two NDs. First step is to connect one ND with the HD, which is similar to the first method, but different at selecting to install by wireless called 'Task Transmitter' (see in the programming algorithm

for assigning task in wireless mode in topic 3.3.6). And the last step, we can load WSN parameter to another ND called 'Task Receptor' via wireless of the Task Transmitter by selecting those value via the display of the HD. In this method, it reduces time and possible damage in process and is easy to use.



(a) (b)

Figure 3.66 Method of setting WSN parameter

(a) First method, which includes one node device, is installed by wire

(b) Second method, which includes two node devices, is installed by wireless

Node device cannot connect with host device via a wire when it is used as a sensor node. But it can connect with host device when its initial value or WSN parameter is cleared. Whatever task and no task is assigned, the ND can clear all parameter within memory by pushing the setting switch, if we want to change task assignment and any WSN parameter of the node device.

In our WSN system, we must select the value of WSN parameter in range that is shown in table 3.14. The NON, which can use one value per one node device, must not repeat on many node devices in installation.

Table 3.14 WSN parameter in range and limiting value

WSN parameter	value
Number of node (NON)	1 - 100 numbers (can up to 1500)
Frequency of node (FON)	1- 125 channels
Assigning task of node (AON)	Sensor node, gateway node

3.6.2 Use of WSN System

Use of WSN system is divided into two parts as shown in figure 3.67. Setup process will be done by user. We already discussed the method for assigning task to node device in topic 3.6.1. Running process will be done by software.

In the first part (setup process), this part is to setup all devices that consist of node device and host device into the intended environment, first step we must set initial value to host device before use and connection with a node device. After that, the ND must be assigned task that can be done by two methods before use. The both methods are the installation via wire and wireless. If all NDs are assigned task completely, those ND will be taken in the intended environment for data collecting. Next step, those ND must be checked for wireless signal of each area. We can see the checking of wireless signal method in dynamic routing of sensor node topic. If wireless signal is available, sensor will be installed to the ND by mini-USB (see in figure 3.11 b) and the ND will start to run. The all above is the setup process of our WSN system which must be finished before use.

In the second part (running process), this part is automatic running software according to algorithm in algorithm topic, the HD and the ND will operate according to the task assigned. The ND will collect information from its sensor in its area and send that data to other node in the nearby area. The ND has two functions that include data collecting and data sending. The last destination in data sending is the HD. The ND may be able to send the data to another ND, or the HD. The Various methods for data sending can be seen in algorithm topic. The data will eventually be gathered to the HD. The HD will take the data from the ND and put into SD card, and show the data on web browser by web server. Our system has saving energy mode for the ND. This mode is run when the ND finishes work or has an irregular event that may damage the system.

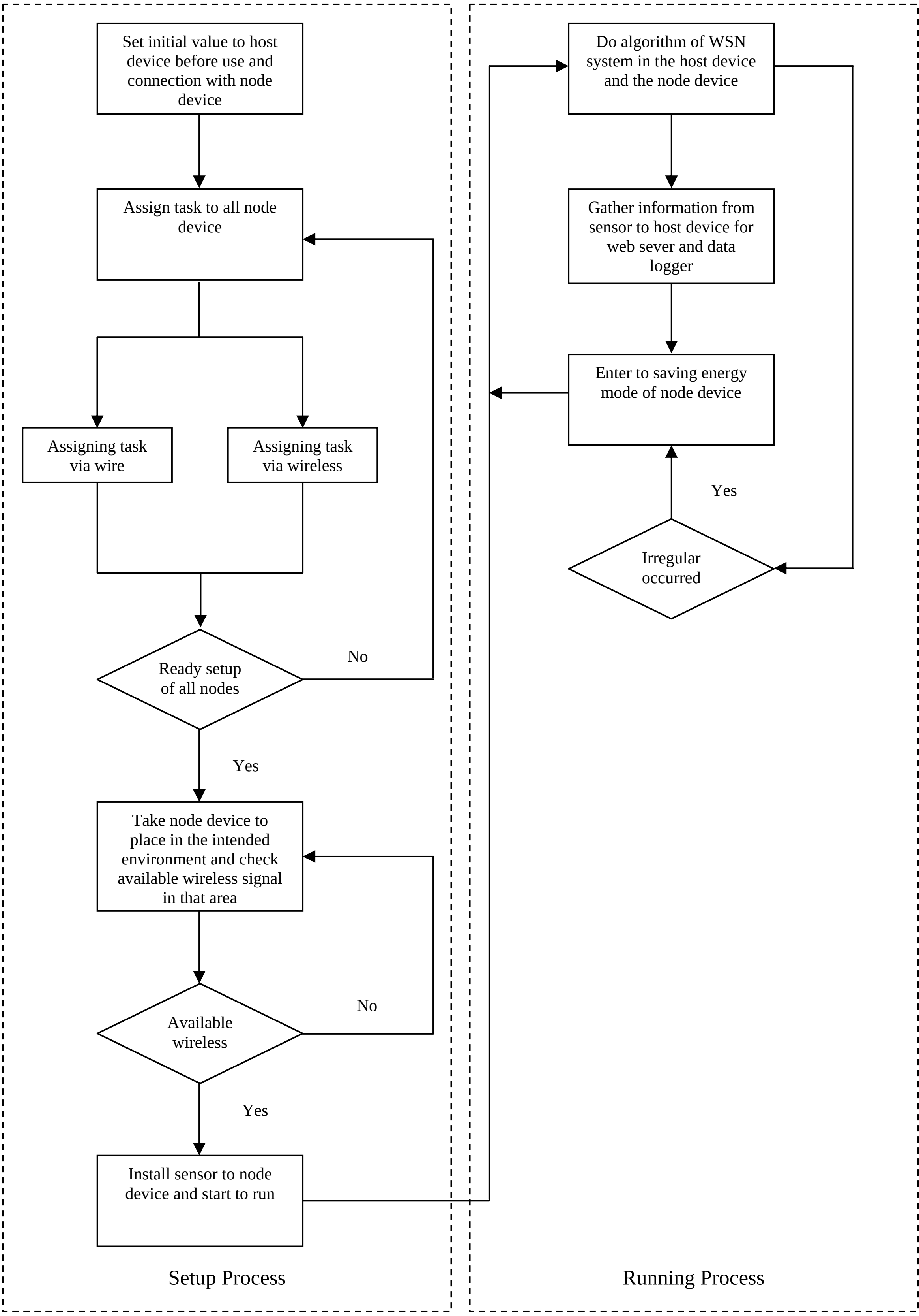


Figure 3.67 Method of WSN system in use

CHAPTER IV
TESTING AND RESULTS

4.1 Testing Concept

The testing in this chapter can be divided into three parts. The first part is to test power consumption of sensor node with respect to various PIC's frequencies for finding the frequency that consumes lowest power consumption. The second part is to test RF-range of sensor node both indoor and outdoor for finding the operation range, maximum range, and behavior of incoming data with respect to ranges. The last part is to test performance of WSN system to determine in term of system lifetime, rate of successful data packet transferred, and dynamic routing function.

4.2 Testing Method

Figure 4.1 presents the testing method. The first part of testing (testing of power consumption of sensor node at topic 4.3) begins to test broadcasting time of wireless broadcasting (topic 4.3.1). From this testing, we will get the timeout parameter (broadcasting time) for use in dynamic routing algorithm. Next step of this test is to optimize broadcasting time to our system (topic 4.3.2). And then, we test for uptime (defined in topic 3.4) of sensor node with various PIC's frequencies (topic 4.3.3). Next step, we will measure the current consumption at each state of sensor node that uses various PIC's frequencies (topic 4.3.4). Last step is to optimize the value from measuring both uptime and current consumption with various PIC's frequencies for finding the best frequency to our system by calculating the power consumption (topic 4.3.5). This frequency will impact system in stability, efficiency, and power consumption.

When all the above testing is finished, we will test for RF-range (topic 4.4). Last test of this chapter, we will test performance of WSN system in various methods for analyzing our system (topic 4.5). The performance of WSN is measured by lifetime of sensor node and success rate of received data at gateway node will be shown when the test completes. All testing will use output power of RF at 0 dBm (maximum), and data rate of RF at 1 mbps (maximum).

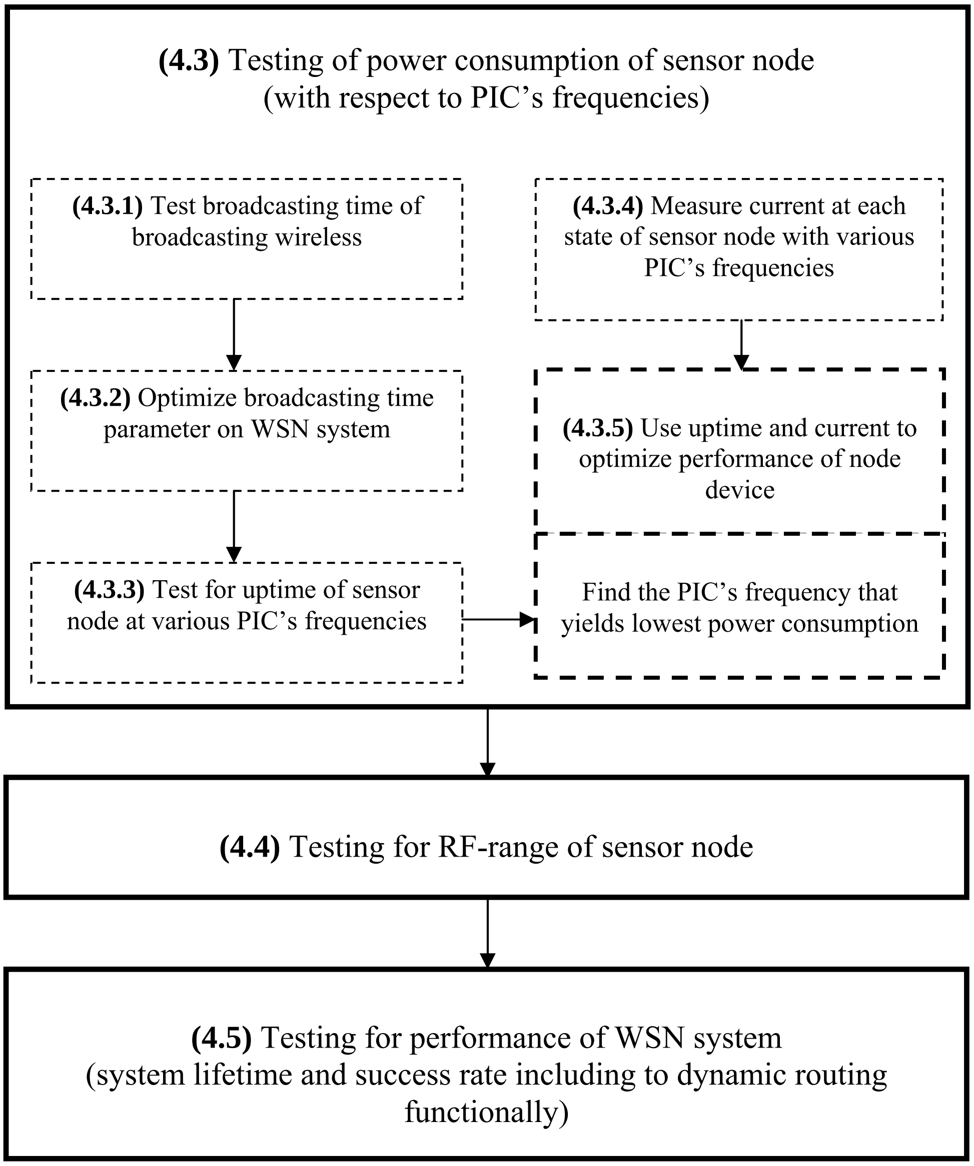


Figure 4.1 Method of test

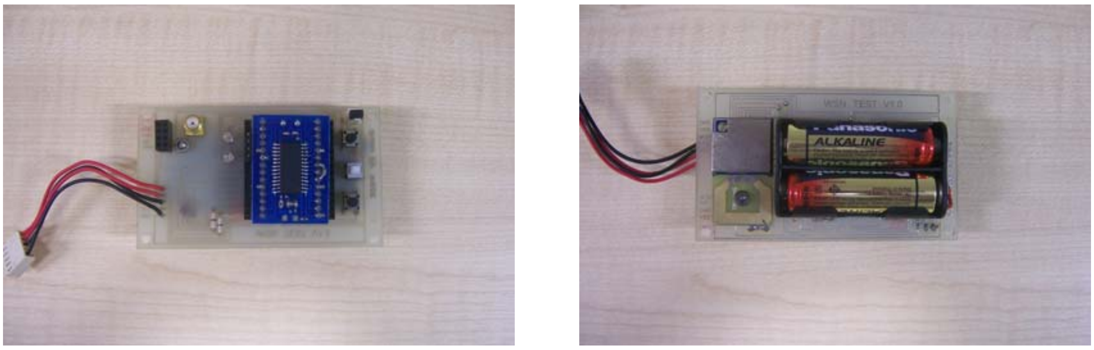
4.3 Testing of Power Consumption of Sensor Node with Various PIC's Frequencies

Before the testing, we will determine parameter of broadcasting time or timeout in wireless broadcasting to compute into algorithm of dynamic routing. This time value is important key that indicates to energy saving and efficiency in WSN system. Finally, we will calculate the power consumption of sensor node by using required value in table 3.12.

4.3.1 To Test Broadcasting Time in Wireless Broadcasting

In testing, we use three sensor nodes and one gateway node. The special gateway node, which can be seen in figure 4.2, functions as a gateway node also recording and analyzing the data packet of sensor node to obtain the broadcasting time. These data will be shown instantly via RS232, which can be directly recorded into text file via RS232. The range of testing is 10–100 centimeters from the sensor node to the gateway node. The testing has three sets to consist of 1 GN with 1 SN, 1 GN with 2 SNs, and 1 GN with 3 SNs. Each set of the test will be performed until 1,000 successful communications with the gateway node are achieved. We can show all results from testing by following measurements.

- **Succeeded number** is number of success which will succeed when the GN found to all SNs in several time.
- **Connected number** is number of communication for one succeeded number of testing.
- **Broadcasting time** is the total testing time from beginning to ending.
- **Maximum time in one success** is the maximum duration time that one successful communication occurred.
- **Minimum time in one success** is the minimum duration time that one successful communication occurred.
- **Average time in one success** is to divide of broadcasting time by 1,000 numbers.



(a) (b)

Figure 4.2 Gateway device for testing

(a) Front of gateway device

(b) Back of gateway device

In table 4.1, sensor node 1 can communicate with gateway node to succeed 1,000 times, which all succeeded numbers have the connected number only 1. The broadcasting time is 91.6 sec. The minimum time in one succeeded is very small.

In table 4.2, sensor node 1 and 2 can communicate with gateway node to succeed 1,000 times, but the test uses the longer time than the first test. The broadcasting time is 143.8 sec. The connected number at 1 of both sensor nodes has the succeeded number below from the first test. The test has maximum value of connected number at 12. Sum of multiplication between connected number and succeeded number is 1,601 in sensor node 1 and 1,545 in sensor node 2.

In table 4.3, sensor node 1, 2, and 3 can communicate with gateway node to succeed 1,000 times, but the test uses the longer time than the first and second test. The broadcasting time is 184.11 sec. The connected number at 1 of three sensor nodes has the succeeded number below from the second test. The test has maximum value of connected number at 13. Sum of multiplication between connected number and succeeded number is 2,255 in sensor node 1, 1,671 in sensor node 2, and 1,898 in sensor node 3.

Table 4.1 Result of testing of 1 GN with 1 SN

Sensor node 1			Time (Sec)			
Connected number	**Succeeded number**	**Multiplication**	**Broadcasting time**	**Max time in one success**	**Min time in one success**	**Average time in one success**
1	1,000	1,000				
2	0	0				
3	0	0				
4	0	0				
5	0	0				
6	0	0				
7	0	0				
8	0	0				
9	0	0				
10	0	0				
11	0	0				
12	0	0				
13	0	0	91.6	0.57	0	0.0916
Sum	1,000	1,000				

Table 4.2 Result of testing of 1 GN with 2 SNs

Sensor node 1			Time (Sec)			
Connected number	**Succeeded number**	**Multiplication**	**Broadcasting time**	**Max time in one success**	**Min time in one success**	**Average time in one success**
1	723	723				
2	133	266				
3	62	186				
4	33	132				
5	25	125				
6	14	84				
7	4	28				
8	1	8				
9	3	27				
10	1	10				
11	0	0				
12	1	12				
13	0	0	143.8	0.84	0.01	0.1438
Sum	1,000	1,601				
Sensor node 2						
Connected number	**Succeeded number**	**Multiplication**				
1	750	750				
2	106	212				
3	65	195				
4	44	617				
5	20	100				
6	6	36				
7	5	35				
8	0	0				
9	1	9				
10	2	20				
11	0	0				
12	1	12				
13	0	0				
Sum	1,000	1,545				

Table 4.3 Result of testing of 1 GN with 3 SNs

Sensor node 1			Time (Sec)			
Connected number	**Succeeded number**	**Multiplication**	**Broadcasting time**	**Max time in one success**	**Min time in one success**	**Average time in one success**
1	527	527				
2	164	328				
3	123	369				
4	72	288				
5	37	185				
6	35	210				
7	16	112				
8	12	96				
9	8	72				
10	2	20				
11	1	11				
12	2	24				
13	1	13	184.11	0.63	0.02	0.18411
Sum	1,000	2,255				

Sensor node 2		
Connected number	**Succeeded number**	**Multiplication**
1	675	675
2	148	296
3	90	270
4	45	180
5	21	105
6	9	54
7	8	56
8	1	8
9	3	27
10	0	0
11	0	0
12	0	0
13	0	0
Sum	1,000	1,671

Sensor node 3		
Connected number	**Succeeded number**	**Multiplication**
1	596	596
2	187	374
3	90	270
4	47	188
5	39	195
6	25	150
7	7	49
8	6	48
9	2	18
10	1	10
11	0	0
12	0	0
13	0	0
Sum	1,000	1,898

The sum of multiplication (the underlined) between connected number and succeeded number is the total numbers of communication between SN and GN. If the underlined value is large (more than 1,000 too), the broadcasting time will increase because the data connection in WSN system increases (resulting in higher bandwidth requirement).

4.3.2 To Optimize Broadcasting Time Parameter on WSN System

From testing result of all tables, the result of time values will be compared together. We see relationship of those values that are same pattern, and can write a formula as follows:

$$\overline{T_{Broadcast_a}} = \overline{T_{Broadcast_1}} + \overline{\Delta T} \times (a-1) --- (10)$$

- a is number of sensor nodes within operating range.
- $\overline{T_{Broadcast_1}}$ is average of broadcasting time of one sensor node.

  $\overline{T_{Broadcast_1}} \approx 0.1$ s (get from table 4.1 in average time in one success)
- $\Delta\overline{T}$ is average of difference broadcasting time between a sensor node and $a-1$ sensor node.

  $\Delta\overline{T} \approx 0.04$ s (get from table 4.1, 4.2, and 4.3 in average time in one success, which is $\overline{T_{Broadcast_2}} - \overline{T_{Broadcast_1}}$ and $\overline{T_{Broadcast_3}} - \overline{T_{Broadcast_2}}$)
- $\overline{T_{Broadcast_a}}$ is average broadcasting time of a sensor node (Second).

From formula above, we can guess the relationship between broadcasting time and number of sensor node during communication of wireless broadcasting. Example, if we choose number of sensor node as 50 nodes, the average broadcasting time of them will be 2.06 second or estimate 2 second. This example implies that they will communicate through 50 nodes in 2 second. If sensor nodes are more than 50, the system will also work, but the system will slow down.

In case of wireless broadcasting, communication will occur when intersection of operating range occurs (see in figure 4.3). We will define the broadcasting time for suitably with number of sensor nodes.

In our WSN system, **the acceptable maximum broadcasting time will be set at 2 second**. This time value, which gives most stability to our system, means to communicate between of 50 sensor nodes in 2 second, or maximum intersection of operating range at most 50 nodes. If the sensor node cannot communicate with other nodes within 2 second, the sensor node will terminate the broadcasting task. The broadcasting time in 2 seconds is a source from limiting value (m) in uptime of dynamic routing mode (chapter 3, topic 3.4). The limiting value of m, which is limited as maximum value at 1,000 (TX/RX routine 1,000 times of broadcast mode), is estimated number of TX/RX routine by calculating the timeout (broadcasting time, $T_{DynamicRouting}$) at 2 second with equation 3.

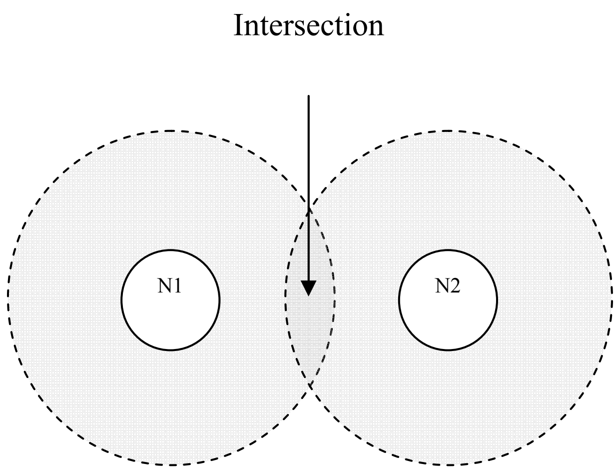


Figure 4.3 Intersection of operating range between nodes

4.3.3 To Test for Uptime of Sensor Node with Various PIC's Frequencies

This testing uses one sensor node and one gateway node by fixing the sleep time of sensor node at 1,000 ms. The special gateway node for testing, which can be seen in figure 4.2, is responsible as a gateway node, and to analyze the data packet of sensor node for storing the uptime. These data will be shown instantly via RS232, which we can directly record these data into text file via RS232. The range of testing is 10-100 centimeters. We test to find uptime of sensor node with various PIC's frequencies. The sensor node has task as a sender only. The uptime is average of uptime value of sensor node when the sensor node finish testing, which it is $T_{DynamicRouting} + T_{Transmission}$. Each of the tests, which use each of PIC's frequency, will be tested for 100 times. We can show the result in table 4.4.

Table 4.4 Comparison between the uptime of sensor node and various PIC's frequencies

PIC's frequencies (kHz)	Average Uptime (ms)	Maximum Uptime (ms)	Minimum Uptime (ms)
1,000	1,826	2,669	478
2,000	686	2,243	237
4,000	234	797	144
8,000	122	187	104
16,000	95	160	85
20,000	89	133	82
32,000	75	97	53

From table 4.4, the testing can be done with at least 1,000 kHz of PIC's frequency. We will see at 32 MHz, which provides minimum uptime, and at 1 MHz, which provides maximum uptime. These values will be used in topic 4.3.5.

4.3.4 The Measurement between Current Consumption of Sensor Node in Each State with Various PIC's Frequencies

The node device has the assignment task that includes the sensor node (SN) and the gateway node (GN). The gateway node in our design will consume the power via the host device. That means the gateway node will not have the problem of lifetime. The sensor node, which differs from the gateway node in term of mobility, must be installed at the intended environment, etc. Thus, the build-in energy source is required for the sensor node. For build-in energy source, we consider many ways such as the non-rechargeable battery, the rechargeable battery, the energy scavenged from the environment such as, solar cell, etc. From figure 2.1, this is the comparison of power used and lifetime between other battery types, solar cell, and vibrations. We can see that the lithium battery has the highest value in both of power used and lifetime. But the lithium battery exhibits high cost and high risk of damage. We should choose the alkaline battery in our application because it gives high value for power used and lifetime, which is second from the lithium battery, but with no risk of damage.

The states of sensor node consist of transmission state, reception state, and sleep state. In wireless broadcasting, the sensor node will includes of the transmission and reception state. The RF module (TRW2.4G) requires the most power consumption, which is more than 90 percent of overall power consumption of our design. In sleep state, the power consumption will be reduced mostly, this state is the method why we must use the time in the sleep state as long as we can. In addition to the reduction of power consumption, we must also see in the efficiency to the system. If a system has low operating power consumption, but with poor operation, the system will have poor overall power consumption.

In measuring the current of each state of sensor node, we will measure the current 100 times for each PIC's frequency. We will use two alkaline batteries (1.5V) that have capacity at 2450 mAh. We set the testing to divide into three states as follows:

- **Transmission state:** MCU ON, TRW2.4G ON (TX), excluding sensor
- **Reception state:** MCU ON, TRW2.4G ON (RX), excluding sensor
- **Sleep state:** MCU OFF, TRW2.4G OFF, excluding sensor

Table 4.5 Comparison between the current consumption of each state of sensor node and various PIC's frequencies

PIC's frequencies (kHz)	Current consumption (mA)			
	Transmission State (TX) $I_{Transmission}$	Reception State (RX) $I_{Reception}$	Sleep State I_{Sleep}	Broadcast (TX+RX)/2 $I_{Broadcast}$
31	Not available	Not available	0.0008	Not available
125	Not available	Not available	0.0008	Not available
250	Not available	Not available	0.0008	Not available
500	Not available	Not available	0.0008	Not available
1,000	3.69	21.85	0.0008	12.77
2,000	4.12	22.20	0.0008	13.16
4,000	4.96	22.74	0.0008	13.85
8,000	6.55	23.76	0.0008	15.16
16,000	9.59	25.92	0.0008	17.76
20,000	11.60	27.43	0.0008	19.52
32,000	14.65	29.31	0.0008	21.98

From table 4.5, TRW2.4G can work at least of PIC's frequencies at 1 MHz, we will see 'Not available' in the table, which it cannot work. In sleep state, the current consumption with various PIC's frequencies has the same value as 0.0008 mA. This value is performance limit of multi-meter, which this value is the least value that the multi-meter can show. The broadcast current is obtained from calculating. These current values will be used in calculation of overall test for finding power consumption.

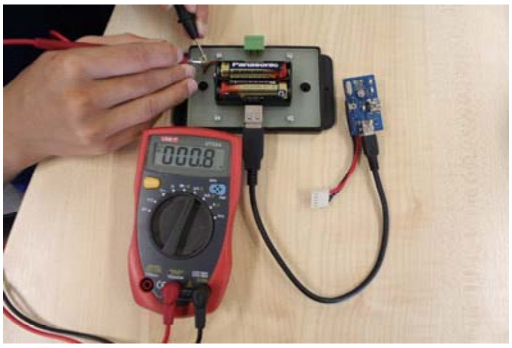


Figure 4.4 Measuring current of each state of sensor node

4.3.5 To Optimize Uptime and Current Consumption of Sensor Node for Finding PIC's Frequencies

This section is to optimize these parameters by fixing sleep time of sensor node at 1,000 ms for the lowest power consumption. The PIC's frequency will affect to the current consumption as shown in table 4.5. Furthermore, the PIC's frequency also affect to the stability of system. If the clock speed is too low, some task on PIC will not work because the PIC does not have enough clock resource for feeding to the task.

In optimization, we will calculate power consumption of each of PIC's frequencies. Eventually, we will get the average current of overall process of sensor node, and we can calculate the lifetime of sensor node by using the battery at 2,450 mAh (alkaline battery). The required values for calculation, which is value of the sender, consist of T_{Sender}, $\overline{I_{Sender}}$, and $LifeTime_{Sender}$, which can see in table 3.11 and 3.12.

Table 4.6 Comparison between the uptime, current consumption, average current, and lifetime with various PIC's frequencies

PIC's frequencies (kHz)	Uptime of Broadcast (ms) $T_{DynamicRouting}$	Current of Broadcast (mA) $I_{Broadcast}$	Uptime of TX (ms) $T_{Transmission}$	Current of TX (mA) $I_{Transmission}$	Sleep Time (ms) T_{Sleep}	Current of Sleep (mA) I_{Sleep}	Average Current (mA) $\overline{I_{Sender}}$	Lifetime (Day) $LifeTime_{Sender}$
1,000	1,817.233	12.77	8.777	3.69	1,000	0.0008	8.22334	12.414
2,000	677.803	13.16	8.777	4.12	1,000	0.0008	5.31066	19.222
4,000	225.473	13.85	8.777	4.96	1,000	0.0008	2.56604	39.782
8,000	113.483	15.16	8.777	6.55	1,000	0.0008	1.58492	64.409
16,000	87.193	17.76	8.777	9.59	1,000	0.0008	1.49048	68.490
20,000	81.163	19.52	8.777	11.60	1,000	0.0008	1.54771	65.958
32,000	66.973	21.98	8.777	14.65	1,000	0.0008	1.48868	68.573

From table 4.6, we use the uptime in table 4.4 (from testing) for calculating, but it is uptime summation between $T_{DynamicRouting}$ and $T_{Transmission}$. Thus, $T_{DynamicRouting}$ can get from the uptime of the test minus $T_{Transmission}$, which $T_{Transmission}$ gets from calculation in equation (4) of topic 3.4 (n = 1, amount of data packet of sender).

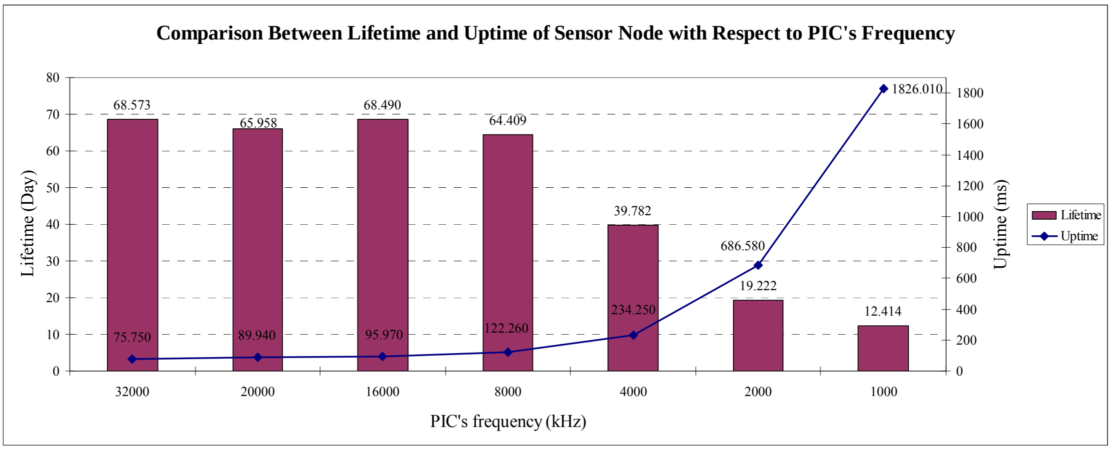


Figure 4.5 Graph of comparison between lifetime and uptime of sensor node with various PIC's frequencies

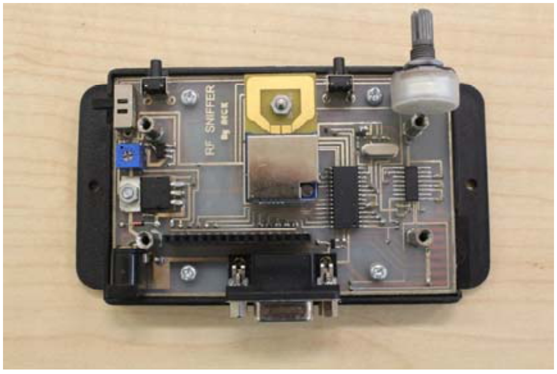
From optimization results as shown in figure 4.5, we will see the PIC's frequency that is 32 MHz, which has longest lifetime.

Although, the 32 MHz PIC's frequency consumes maximum power (from measuring current), but it provides minimum uptime. The all uptime of sensor node consist of uptime and sleep time, which the sleep time is set at 1,000 ms. If the uptime is very small while the sleep time is constant, lifetime of sensor node will be high because power consumption is averaged by small uptime and constant sleep time. The heavy current has small duration (small uptime) while the light current has high duration constantly (sleep time 1,000 ms). Thus, the uptime and the current used will affect on lifetime, we must optimize these parameters together for longest lifetime.

Furthermore, the lifetime that is the most important, which we use it for the first consideration. The uptime is considered to affects to system, which it is dependent on PIC's frequency. Thus, we choose the PIC's frequency that is 32 MHz for sensor node during operation, because this frequency provides highest performance at maximum lifetime and minimum uptime.

4.4 Testing for RF-Range of Sensor Node

This section is to test of RF-range of sensor node both indoor and outdoor for finding the best range and behavior of incoming data. We made the test at Faculty of Engineering, Mahidol University. Both indoor and outdoor do not have barricade in way of test. The measurement devices consist of one sensor node and one host node. In the sensor node, it is a sender only and uses 32 MHz PIC's frequency. In the host node, it is responsible as a gateway node to combine with base-station, which is called Host-Meter Device as shown in figure 4.6. All RF devices (TRW2.4G) are configured to use output power 0 dBm, data transfer rate at 1 Mbit/s, 2 alkaline batteries (2,450 mAh 1.5 V).



(a)

(b)

Figure 4.6 Host-Meter device

(a) Host-Meter device inside

(b) Host-Meter device complete

The testing of RF-range are performed both indoor and outdoor, we record data of successful data packet transferred in one minute at every ten meters. The number of successful data packet transferred is data packet from the sender that arrive the Host-Meter. These data will be recorded into the Host-Meter device. We can see these data directly via this device. Furthermore, we also kept uptime and overall time of sensor node. Height of sensor node and Host-meter in the test is more than 1 meters from the ground.

From the test of indoor and outdoor, we found success rate maximum at 49 packets/minute, which we can be estimated time on air from overall time of sensor node (uptime + sleep time + time on air) because we know uptime and sleep time. The time on air will has value about 150–200 ms, which we assume the overall time of sensor node least at 1,226 ms (uptime: 76 ms + sleep time: 1,000 ms + time on air: 150 ms). The uptime is used at 76 ms. This value is lowest from this test. Thus, in one minute, the Host-meter will be able to receive packet maximum at 49 packets (1,226 ms × 49 packets ≈ 60,000 ms). This value will be used to calculate percent of success rate. The percent of success rate must be more than 80% to be considered within the operation range of sensor node. For other percentages of success rate that is lower than 80%, sensor node will also work in those ranges, but performance of sensor node will be dropped. For maximum range, if the Host-Meter can detect the packet at least one packet within one minute, this range will be used as maximum range.

4.4.1 To Test of RF-Range Indoor

In the test of RF-range indoor as shown the location in figure 4.7, the result of the test can be seen in table 4.7. The height for the ceiling in indoor is 2.5 meters from the floor.

Figure 4.7 Indoor location of testing

Table 4.7 Result of RF-range indoor

Range (m)	Number of successful data packet transferred (packets/minute)	Percent of success rate (%)
10	48	97.96
20	46	93.88
30	48	97.96
40	44	89.80
50	39	79.59
60	35	71.43
70	31	63.27
80	28	57.14
90	25	51.02
100	27	55.10
110	23	46.94
120	18	36.73
130	15	30.61
140	9	18.37

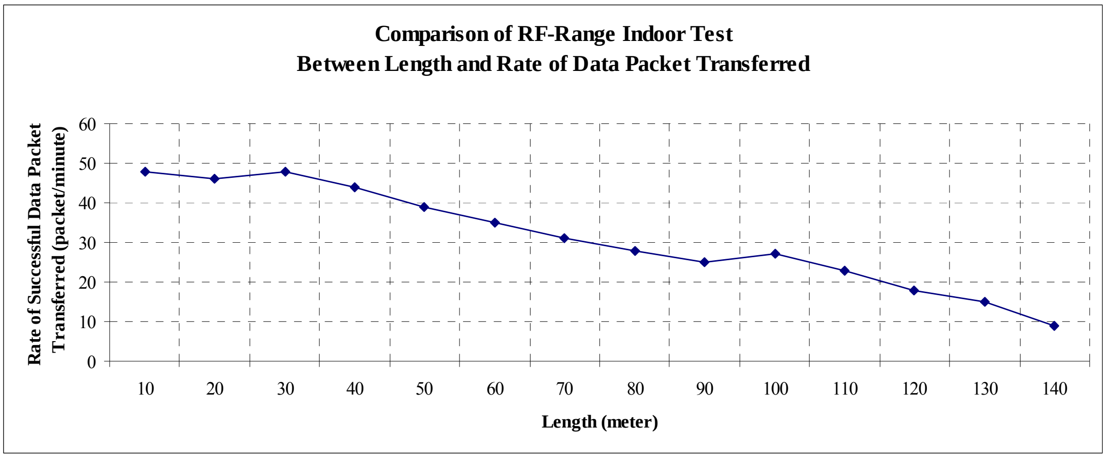


Figure 4.8 Graph of comparison of RF-range indoor test between length and rate of data packet transferred

From graph in figure 4.8, we can define operation range of sensor node from result of the test, which is lower 50 meters or 0-47 meters (percent of success rate is lower than 80%). The maximum range in indoor is 140 meters, which get from the test. The range that is more than 140 meters cannot detect the data from sensor node.

4.4.2 To Test of RF-Range Outdoor

In the test of RF-range outdoor as shown the location in figure 4.9, the result of the test can be seen in table 4.8.

Figure 4.9 Outdoor location of testing

Table 4.8 Result of RF-range outdoor

Range (m)	Number of successful data packet transferred (packets/minute)	Percent of success rate (%)	Range (m)	Number of successful data packet transferred (packets/minute)	Percent of success rate (%)
10	49	100.00	160	29	59.18
20	48	97.96	170	30	61.22
30	48	97.96	180	26	53.06
40	47	95.92	190	25	51.02
50	47	95.92	200	23	46.94
60	45	91.84	210	24	48.98
70	46	93.88	220	22	44.90
80	43	87.76	230	18	36.73
90	40	81.63	240	19	38.78
100	37	75.51	250	15	30.61
110	38	77.55	260	14	28.57
120	35	71.43	270	11	22.45
130	37	75.51	280	8	16.33
140	35	71.43	290	5	10.20
150	33	67.35	300	7	14.29

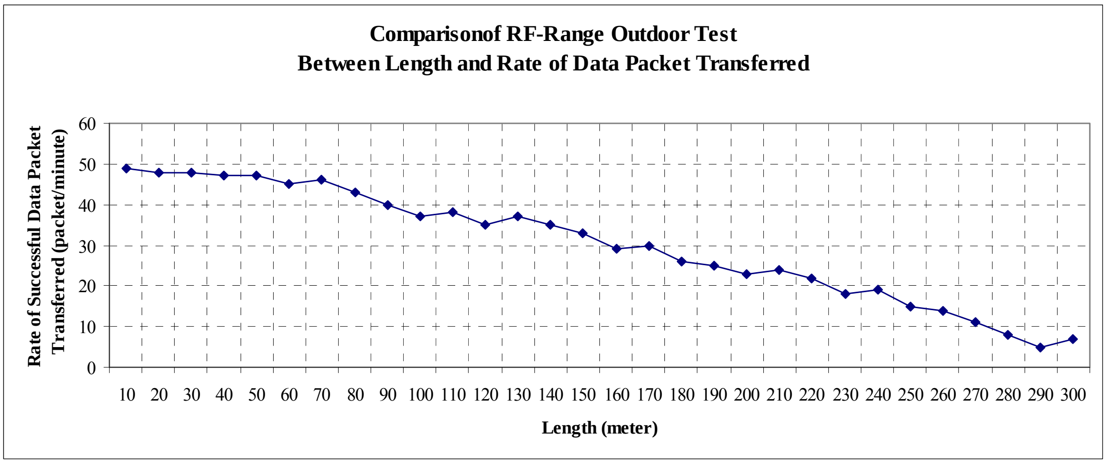


Figure 4.10 Graph of comparison of RF-range outdoor test between length and rate of data packet transferred

From graph in figure 4.10, we can define operation range of sensor node from result of the test, which is lower 100 meters or 0-90 meters (percent of success rate is lower than 80%). The maximum range in outdoor is 300 meters, which get from the test. The range that is more than 300 meters cannot detect the data from sensor node.

4.5 Testing for Performance of WSN System

This section is to test for performance of WSN system. In this test, the system will consist of four sensor nodes and one Host-Meter device as shown in figure 4.11.

Figure 4.11 Devices for performance testing

4.5.1 Method of Testing for Performance of WSN System

In the test, we will use the same indoor location as in topic 4.4.1. This location can be easily used for our testing conditions, which consists of many cases as shown various locations in figure 4.12. In the test, we will record the data of each sensor node that consists of uptime ($T_{DynamicRouting} + T_{Transfer}$) and data packet holding (amount of data packet). Furthermore, overall testing time will be recorded for calculating the rate of successful data packet transferred. We compare these data that are previously told with various patterns of network, which can be seen in the result of each testing.

The test will be achieved when the Host-Meter can store the data packet of each sensor nodes to 100 data packets and repeats 10 sets. Each set will consist least of 400 data packets of all SNs. Each case of network pattern in testing can total to least 4,000 data packets (4,000 samples). All testing (7 cases of network pattern) can total to least 28,000 data packets (28,000 samples).

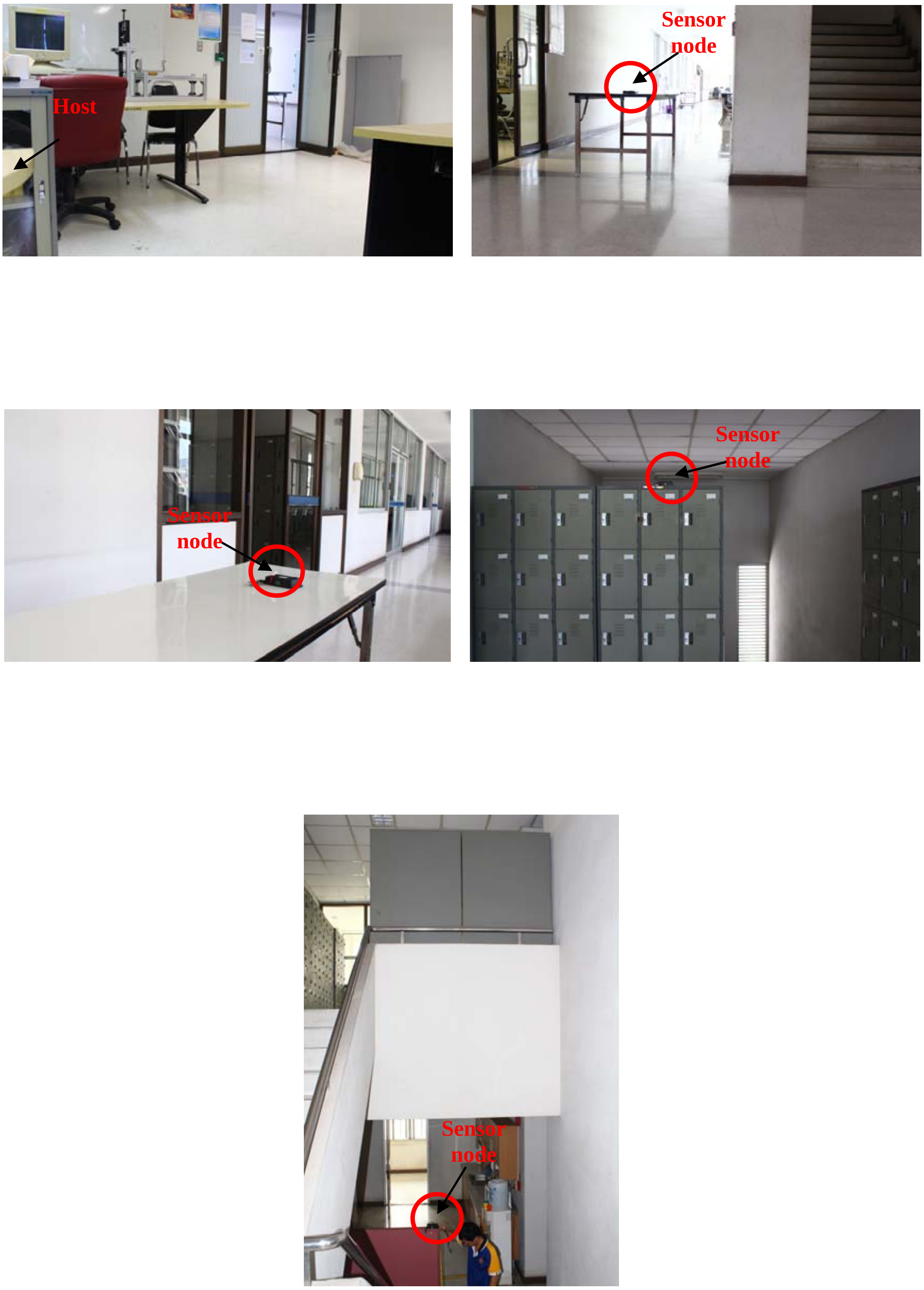


Figure 4.12 Locations in overall testing

This testing consists of 7 cases by network pattern. The RF (TRW2.4G) is configured by using data transfer rate at 1 Mbit/s and output power 0 dBm. The sensor node uses two alkaline batteries (2,450 mAh 1.5V), set sleep time at 1 second, and uses 32 MHz PIC's frequency. The distance between nodes in the test is higher than 10 meters (about 10–20 meters) and is in indoor location. The data from testing can be seen as below.

The uptime is time summation of dynamic routing mode and transfer mode. We record this value when the Host-Meter receive packet from any SN (SN1-SN4). If the uptime is small, the system will be fast operation, high success rate, and high lifetime. The uptime affected from many things. The network pattern is one factor, which we will compare those uptimes from the test in all cases of network pattern. The last, those uptime will be used to calculate lifetime of sensor node.

Number of hold data packet is the number that a SN holds the data packet. We record this value when the Host-Meter receive packet from any SN (SN1-SN4). This value can be maximum 4 data packets in this test because we have 4 SNs. This value can indicate task of sensor node and number of hop in communication. If this value is 1, task will be sender and has maximum 1 hop. If this value is more than 1, task will be repeater and has maximum more than 1 hop. Furthermore, the holding data packet also can indicate the number of used path of communication and calculates percent of hop (percent of data packet transferred with n hops) in communication.

Overall testing time can get from ending of the test, which it will be used to calculate rate of successful data packet transferred. The success rate of data packet, which has range between 100,000–400,000 ms, has highest value at 4 packets/s. We use formula for calculating success rate as follow.

$$Success\ of\ Dara\ Packet\ Transferred\,(packets/s) = \frac{400}{OverallTestingTime(s)}$$

Give, 400 is number of data packet of all SNs that be succeeded in one set.

Sensor node will have sample rate about 1 sample/s, which this value gets from data packet dividing by all uptime (Dynamic Routing + Transfer + Sleep). The sleep time is set at 1 second while both time of dynamic routing and transfer can be changed by algorithm, which we can calculate sample rate by using these information. We will get one data packet that is transferred successful in the time, which are uptime (Dynamic Routing + Transfer) + 1 second. The uptime is very small (minimum at 76 ms from the test). Thus, we can calculate the sample rate of each sensor nodes, which the sample rate is maximum 0.93 samples/s or about 1 sample/s. Hence, the rate of successful data packet transferred at 4 packets/s is the best value, which calculates from summation of maximum sample rate of four sensor nodes.

Network patterns are divided by communications between Host-SNs, which can see as follows (In case 2, 4, and 6, we will show number of paths for communication that is higher than that for case 1, 3, and 5 respectively to support with dynamic routing method, which has more paths for communication).

- Case 1 and 2 have number of communications between Host-SNs at 4
- Case 3 and 4 have number of communications between Host-SNs at 3
- Case 5 and 6 have number of communications between Host-SNs at 2
- Case 7 has number of communications between Host-SNs at 1

Indicator of this test is **Uptime** and **Overall testing time**. Those indicators will indicate the performance of the systems, which is lifetime of sensor node (the lowest lifetime will be seen as lifetime of overall system), and rate of successful data packet transferred. Furthermore, we monitor path of SN4 when SN2 loses (dynamic routing function) by using **Number of hold data packet**.

Case 1 of Network Pattern

This case, which is shown in figure 4.13, appears maximum of 1 hop. This network is star topology. Communications between Host and SN found maximum of four (SN1→Host, SN2→Host, SN3→Host, SN4→Host). Communications between SNs are not found. All four SNs are responsible as senders only. We can see the result of case 1 as table 4.9 (Communication path), table 4.10 (sensor node uptime), and table 4.11 (various parameters).

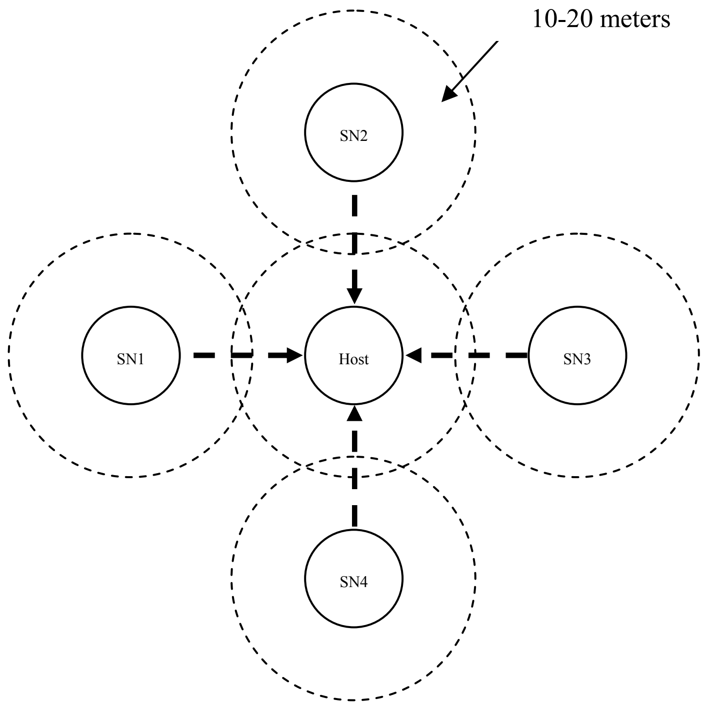


Figure 4.13 Network pattern in case 1

Table 4.9 Communication path in case 1

Path Number	Hop	All Path	Number of Used Path
1	1	SN1→Host	✓
2	1	SN2→Host	✓
3	1	SN3→Host	✓
4	1	SN4→Host	✓
SUM		4	4

Table 4.10 Result of sensor node uptime in case 1

Node	Uptime of SN (ms)			Percent of Uptime within the Range (%)					
	Average	Maximum	Minimum	>2000 ms	1000-2000 ms	500-1000 ms	250-500 ms	100-250 ms	0-100 ms
SN1	86	202	76	0	0	0	0	11.3	88.7
SN2	103	837	76	0	0	0.7	0.1	32.3	66.9
SN3	93	1,312	76	0	0.3	0.3	0.1	13.8	85.5
SN4	117	1,333	76	0	0.4	0.4	0.1	52.8	46.3
$\overline{\text{SUM}}$				0	0.175	0.35	0.075	27.55	71.85

Table 4.11 Result of various parameters in case 1

Average of Overall Testing Time	Rate of Successful Data Packet Transferred	Node	Percent of Number of Hold Data (%)			
			4 Packet	3 Packet	2 Packet	1 Packet
144,924 ms	2.68 packets/s	SN1	0	0	0	100
		SN2	0	0	0	100
		SN3	0	0	0	100
		SN4	0	0	0	100
$\overline{\text{SUM}}$			0	0	0	100

Case 2 of Network Pattern

This case, which appears all 1-4 hops as the case 1, is shown in figure 4.14. But from the test, we found maximum at 3 hops, such as SN1→SN4→SN3→Host. This network is mesh topology. Communications between Host and SN found maximum at four (SN1→Host, SN2→Host, SN3→Host, SN4→Host). Communications between SNs found maximum at two, which appears for all SNs (such as at SN4, SN4↔SN1, SN4↔SN3). All four SNs can be both a sender and repeater. We can see the result of case 2 as table 4.12 (Communication path), table 4.13 (sensor node uptime), and table 4.14 (various parameters).

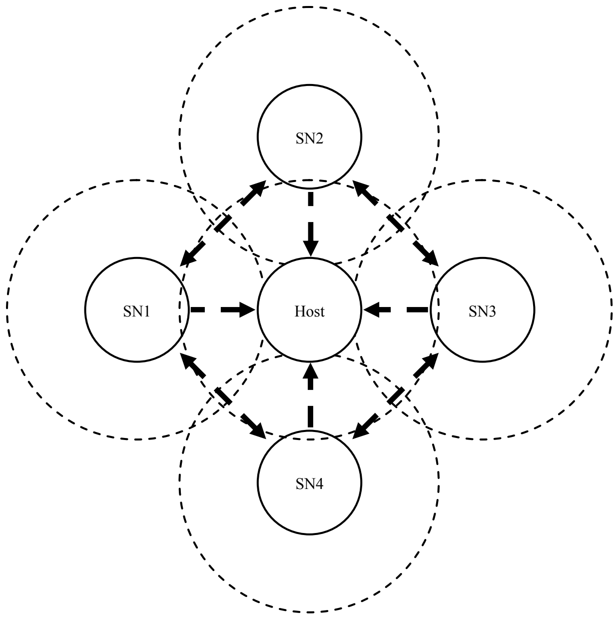


Figure 4.14 Network pattern in case 2

Table 4.12 Communication path in case 2

Path Number	Hop	All Path	Number of Used Path
1	1	SN1→Host	✓
2	1	SN2→Host	✓
3	1	SN3→Host	✓
4	1	SN4→Host	✓
5	2	SN1→SN2→Host	✓
6	2	SN1→SN4→Host	✓
7	2	SN2→SN1→Host	✓
8	2	SN2→SN3→Host	✓
9	2	SN3→SN2→Host	✓
10	2	SN3→SN4→Host	✓
11	2	SN4→SN1→Host	✓
12	2	SN4→SN3→Host	✓
13	3	SN1→SN2→SN3→Host	✓
14	3	SN1→SN4→SN3→Host	✓
15	3	SN2→SN1→SN4→Host	✓
16	3	SN2→SN3→SN4→Host	✓
17	3	SN3→SN2→SN1→Host	✓
18	3	SN3→SN4→SN1→Host	✓
19	3	SN4→SN1→SN2→Host	×
20	3	SN4→SN3→SN2→Host	×
21	4	SN1→SN2→SN3→SN4→Host	×
22	4	SN1→SN4→SN3→SN2→Host	×
23	4	SN2→SN1→SN4→SN3→Host	×
24	4	SN2→SN3→SN4→SN1→Host	×
25	4	SN3→SN2→SN1→SN4→Host	×
26	4	SN3→SN4→SN1→SN2→Host	×
27	4	SN4→SN1→SN2→SN3→Host	×
28	4	SN4→SN3→SN2→SN1→Host	×
SUM		28	18

Table 4.13 Result of sensor node uptime in case 2

Node	Uptime of SN (ms)			Percent of Uptime within the Range (%)					
	Average	Maximum	Minimum	>2000 ms	1000-2000 ms	500-1000 ms	250-500 ms	100-250 ms	0-100 ms
SN1	155	2,114	76	0.1	3.5	2	1.2	12.1	81.1
SN2	163	2,667	76	0.4	2.8	2.4	3.1	17.3	74
SN3	168	2,138	76	0.2	3.5	3.2	3	13.9	76.2
SN4	355	2,114	76	0.9	10.5	9.5	22	24	33.1
$\overline{\text{SUM}}$				0.4	5.075	4.275	7.325	16.825	66.1

Table 4.14 Result of various parameters in case 2

Average of Overall Testing Time	Rate of Successful Data Packet Transferred	Node	Percent of Number of Hold Data (%)			
			4 Packet	3 Packet	2 Packet	1 Packet
166,373 ms	2.41 packets/s	SN1	0	0.1	9.7	90.2
		SN2	0	0	10.5	89.5
		SN3	0	0.1	12.8	87.1
		SN4	0	0.5	42.9	56.6
$\overline{\text{SUM}}$			0	0.175	18.975	80.85

Case 3 of Network Pattern

This case, which is shown in figure 4.15, appears maximum of 2 hops (SN4→SN2→Host). This network is cluster tree topology. Communications between Host and SN found maximum at three (SN1→Host, SN2→Host, SN3→Host). Communications between SNs found maximum at one, which appears between SN4 and SN2 (SN4→SN2). SN1, SN3, and SN4 are responsible as sender only, but SN2 can be both a sender and repeater. We can see the result of case 3 as table 4.15 (Communication path), table 4.16 (sensor node uptime), and table 4.17 (various parameters).

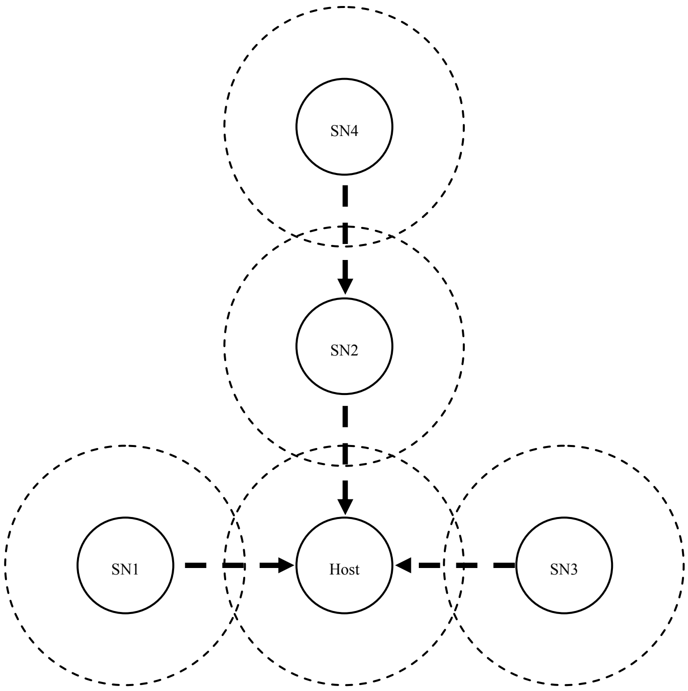


Figure 4.15 Network pattern in case 3

Table 4.15 Communication path in case 3

Path Number	Hop	All Path	Number of Used Path
1	1	SN1→Host	✓
2	1	SN2→Host	✓
3	1	SN3→Host	✓
4	2	SN4→SN2→Host	✓
SUM		4	4

Table 4.16 Result of sensor node uptime in case 3

Node	Uptime of SN (ms)			Percent of Uptime within the Range (%)					
	Average	**Maximum**	**Minimum**	**>2000 ms**	**1000-2000 ms**	**500-1000 ms**	**250-500 ms**	**100-250 ms**	**0-100 ms**
SN1	91	5,141	76	0.1	0	0	0.4	8.5	91
SN2	285	1,577	77	0	3	0	54.5	42	0.5
SN3	95	1,729	76	0	0.1	0.1	0.3	20.4	79.1
SN4	134	1,455	76	0	0.7	0.2	0	73.3	25.8
$\overline{\text{SUM}}$				0.025	0.95	0.075	13.8	36.05	49.1

Table 4.17 Result of various parameters in case 3

Average of Overall Testing Time	Rate of Successful Data Packet Transferred	Node	Percent of Number of Hold Data (%)			
			4 Packet	**3 Packet**	**2 Packet**	**1 Packet**
141,887 ms	2.82 packets/s	SN1	0	0	0	100
		SN2	0	0	99.3	0.7
		SN3	0	0	0	100
		SN4	0	0	0	100
$\overline{\text{SUM}}$			0	0	24.825	75.175

Case 4 of Network Pattern

This case, which appears all 1-3 hops as the case 3, is shown in figure 4.16. But from the test, we found maximum at 2 hops, such as SN4→SN2→Host. This network is mesh topology. Communications between Host and SN found maximum at three (SN1→Host, SN2→Host, SN3→Host). Communications between SNs found maximum at two, which appears at SN4 (SN4→SN2, SN4→SN3). SN2, SN3, and SN4 can be both a sender and repeater, but SN1 are responsible as sender only. We can see the result of case 4 as table 4.18 (Communication path), table 4.19 (sensor node uptime), and table 4.20 (various parameters).

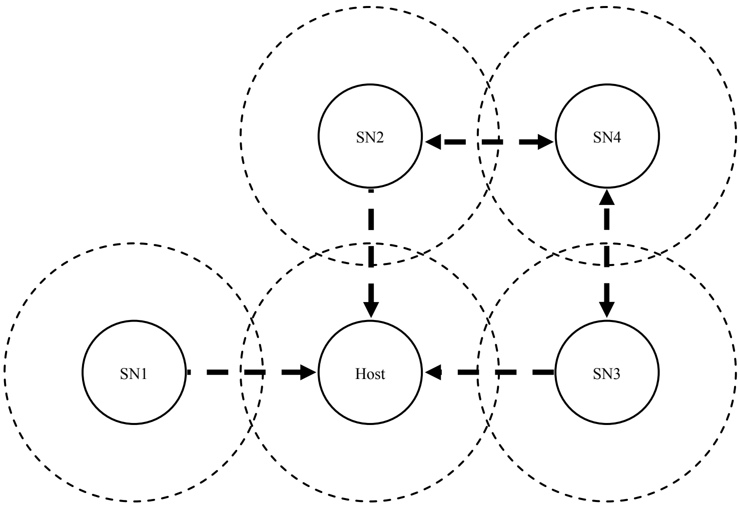


Figure 4.16 Network pattern in case 4

Table 4.18 Communication path in case 4

Path Number	Hop	All Path	Number of Used Path
1	1	SN1→Host	✓
2	1	SN2→Host	✓
3	1	SN3→Host	✓
4	2	SN4→SN2→Host	✓
5	2	SN4→SN3→Host	✓
6	3	SN2→SN4→SN3→Host	×
7	3	SN3→SN4→SN2→Host	✓
SUM		7	6

Table 4.19 Result of sensor node uptime in case 4

Node	Uptime of SN (ms)			Percent of Uptime within the Range (%)					
	Average	**Maximum**	**Minimum**	**>2000 ms**	**1000-2000 ms**	**500-1000 ms**	**250-500 ms**	**100-250 ms**	**0-100 ms**
SN1	97	9,094	76	0.1	0	0.1	0.4	10.6	88.8
SN2	190	1,970	76	0	2.4	0	21.5	36.5	39.6
SN3	230	2,092	76	0.2	3.6	0.2	24.9	33.6	37.5
SN4	129	744	76	0	0	0.4	0.5	77.7	21.4
$\overline{\text{SUM}}$				0.075	1.5	0.175	11.825	39.6	46.825

Table 4.20 Result of various parameters in case 4

Average of Overall Testing Time	Rate of Successful Data Packet Transferred	Node	Percent of Number of Hold Data (%)			
			4 Packet	**3 Packet**	**2 Packet**	**1 Packet**
147,915 ms	2.71 packets/s	SN1	0	0	0	100
		SN2	0	0.1	42	57.9
		SN3	0	0	57	43
		SN4	0	0	0.1	99.9
$\overline{\text{SUM}}$			0	0.025	24.775	75.2

Case 5 of Network Pattern

This case, which is shown in figure 4.17, appears maximum of 3 hops (SN4→SN3→SN2→Host). This network is cluster tree topology. Communications between Host and SN found maximum at two (SN1→Host, SN2→Host). Communications between SNs found maximum at two, which appears at SN3 (SN4→SN3, SN3→SN2). SN2 and SN3 can be both a sender and repeater, but SN1 and SN4 are responsible as sender only. We can see the result of case 5 as table 4.21 (Communication path), table 4.22 (sensor node uptime), and table 4.23 (various parameters).

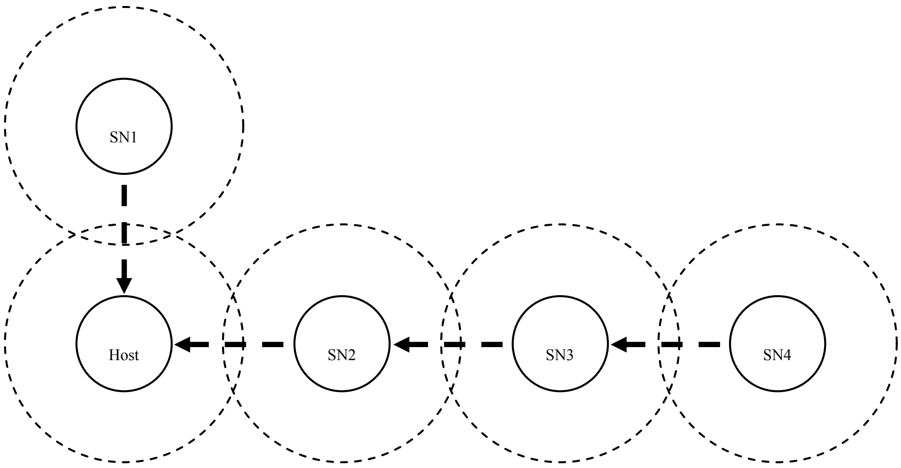


Figure 4.17 Network pattern in case 5

Table 4.21 Communication path in case 5

Path Number	Hop	All Path	Number of Used Path
1	1	SN1→Host	✓
2	1	SN2→Host	×
3	2	SN3→SN2→Host	✓
4	3	SN4→SN3→SN2→Host	✓
SUM		4	3

Table 4.22 Result of sensor node uptime in case 5

Node	Uptime of SN (ms)			Percent of Uptime within the Range (%)					
	Average	**Maximum**	**Minimum**	**>2000 ms**	**1000-2000 ms**	**500-1000 ms**	**250-500 ms**	**100-250 ms**	**0-100 ms**
SN1	104	1,251	76	0	1.2	0	0	13	85.8
SN2	334	2,067	116	0.1	6.5	2.8	41	49.6	0
SN3	313	2,278	77	0.5	3.9	1.8	40	53.6	0.2
SN4	154	1,253	76	0	1.1	1.8	2	80.6	14.5
$\overline{\text{SUM}}$				0.15	3.175	1.6	20.75	49.2	25.125

Table 4.23 Result of various parameters in case 5

Average of Overall Testing Time	Rate of Successful Data Packet Transferred	Node	Percent of Number of Hold Data (%)			
			4 Packet	**3 Packet**	**2 Packet**	**1 Packet**
152,162 ms	2.63 packets/s	SN1	0	0	0	100
		SN2	0	93.8	6.2	0
		SN3	0	0	99.6	0.4
		SN4	0	0	0	100
$\overline{\text{SUM}}$			0	23.45	26.45	50.1

Case 6 of Network Pattern

This case, which appears all 1-4 hops as the case 5, is shown in figure 4.18. But from the test, we found maximum at 3 hops, such as SN4→SN3→SN2→Host. This network is mesh topology. Communications between host and SN found maximum at two (SN1→Host, SN2→Host). Communications between SNs found maximum at three, which appears at SN3 (SN4→SN3, SN3→SN2, SN3→SN1). SN1, SN2, and SN3 can be both a sender and repeater, but SN4 are responsible as sender only. We can see the result of case 6 as table 4.24 (Communication path), table 4.25 (sensor node uptime), and table 4.26 (various parameters).

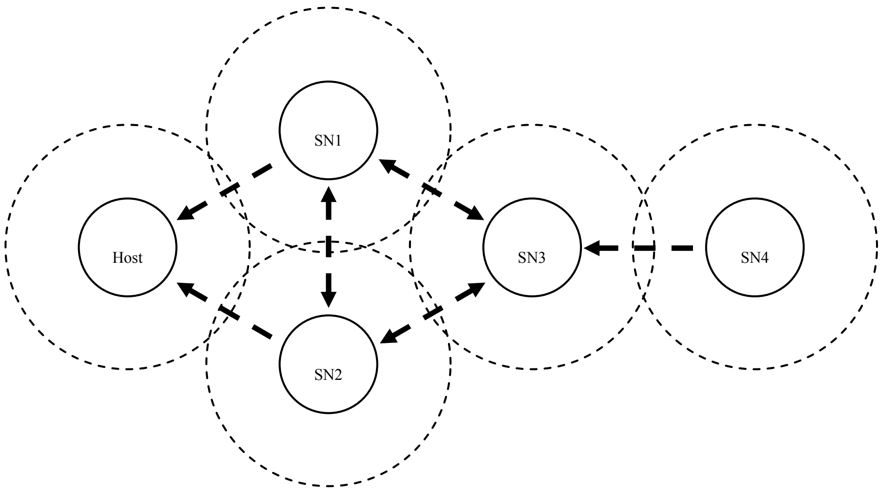


Figure 4.18 Network pattern in case 6

Table 4.24 Communication path in case 6

Path Number	Hop	All Path	Number of Used Path
1	1	SN1→Host	✓
2	1	SN2→Host	✓
3	2	SN1→SN2→Host	✓
4	2	SN2→SN1→Host	✓
5	2	SN3→SN1→Host	×
6	2	SN3→SN2→Host	✓
7	3	SN1→SN3→SN2→Host	×
8	3	SN2→SN3→SN1→Host	×
9	3	SN3→SN1→SN2→Host	×
10	3	SN3→SN2→SN1→Host	×
11	3	SN4→SN3→SN1→Host	✓
12	3	SN4→SN3→SN2→Host	✓
13	4	SN4→SN3→SN1→SN2→Host	×
14	4	SN4→SN3→SN2→SN1→Host	×
SUM		14	7

Table 4.25 Result of sensor node uptime in case 6

Node	Uptime of SN (ms)			Percent of Uptime within the Range (%)					
	Average	Maximum	Minimum	>2000 ms	1000-2000 ms	500-1000 ms	250-500 ms	100-250 ms	0-100 ms
SN1	219	1,598	76	0	3.7	2.2	8.9	45.1	40.1
SN2	216	1,603	76	0	3.7	1.4	16.2	37.5	41.2
SN3	313	2,757	114	0.7	3.4	2.1	43.7	50.1	0
SN4	157	1,595	76	0	1.3	2.2	0.7	86.6	9.2
$\overline{\text{SUM}}$				0.175	3.025	1.975	17.375	54.825	22.625

Table 4.26 Result of various parameters in case 6

Average of Overall Testing Time	Rate of Successful Data Packet Transferred	Node	Percent of Number of Hold Data (%)			
			4 Packet	3 Packet	2 Packet	1 Packet
152,791 ms	2.62 packets/s	SN1	0	51.9	2.0	46.1
		SN2	0	49.6	1.9	48.5
		SN3	0	0	99.9	0.1
		SN4	0	0	0	100
$\overline{\text{SUM}}$			0	25.375	25.95	48.675

Case 7 of Network Pattern

This case, which is shown in figure 4.19, appears maximum of 4 hops (SN4→SN3→SN2→SN1→Host). This network is mesh topology. Communications between host and SN found one only (SN1→Host). Communications between SNs found maximum at two, which appears at SN2 (SN3→SN2, SN2→SN1) and SN3 (SN4→SN3, SN3→SN2). SN1, SN2, and SN3 can be both a sender and repeater, but SN4 are responsible as sender only. We can see the result of case 7 as table 4.27 (Communication path), table 4.28 (sensor node uptime), and table 4.29 (various parameters).

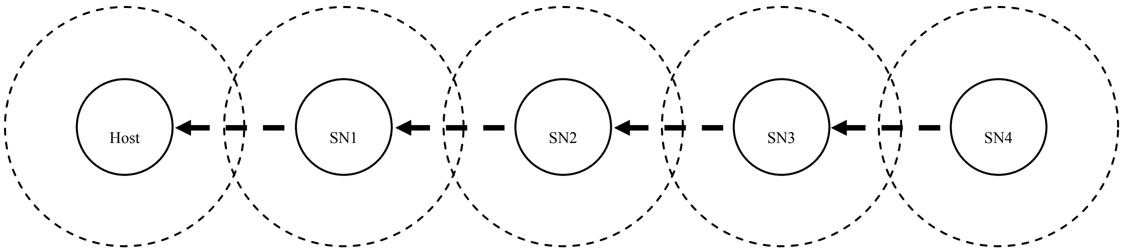


Figure 4.19 Network pattern in case 7

Table 4.27 Communication path in case 7

Path Number	Hop	All Path	Number of Used Path
1	1	SN1→Host	✓
2	2	SN2→SN1→Host	×
3	3	SN3→SN2→SN1→Host	✓
4	4	SN4→SN3→SN2→SN1→Host	✓
SUM		4	3

Table 4.28 Result of sensor node uptime in case 7

Node	Uptime of SN (ms)			Percent of Uptime within the Range (%)					
	Average	Maximum	Minimum	>2000 ms	1000-2000 ms	500-1000 ms	250-500 ms	100-250 ms	0-100 ms
SN1	341	2,000	79	0.1	9.8	1.1	5.8	82.9	0.3
SN2	486	3,819	145	0.5	6.3	36.8	26.4	30	0
SN3	632	2,591	83	1	4	63	27.2	4.7	0.1
SN4	141	1,836	76	0	0.6	1.1	0.3	84.8	13.2
$\overline{\text{SUM}}$				0.4	5.175	25.5	14.925	50.6	3.4

Table 4.29 Result of various parameters in case 7

Average of Overall Testing Time	Rate of Successful Data Packet Transferred	Node	Percent of Number of Hold Data (%)			
			4 Packet	3 Packet	2 Packet	1 Packet
157,025 ms	2.55 packets/s	SN1	97.5	1.4	0.5	0.6
		SN2	0	99.3	0.7	0
		SN3	0	0	99.8	0.2
		SN4	0	0	0	100
$\overline{\text{SUM}}$			24.375	25.175	25.25	25.2

4.5.2 Effect of Uptime of Sensor Node on System Lifetime

Uptime of sensor node is an indicator for sensor node lifetime, which if uptime is low, lifetime will be high. For system lifetime, the lowest lifetime of sensor nodes will be defined as the overall lifetime. Typically, lifetime of a sender is doubled that of a repeater because the sender uses about half uptime of the repeater. If a task has both sender and repeater, the uptime will be averaged between them. Uptime of sensor node is divided into six ranges, which these ranges will tell behavior of uptime. We can show distribution of uptime (out of 4,000 samples in term of percentage) in all cases as in figure 4.20.

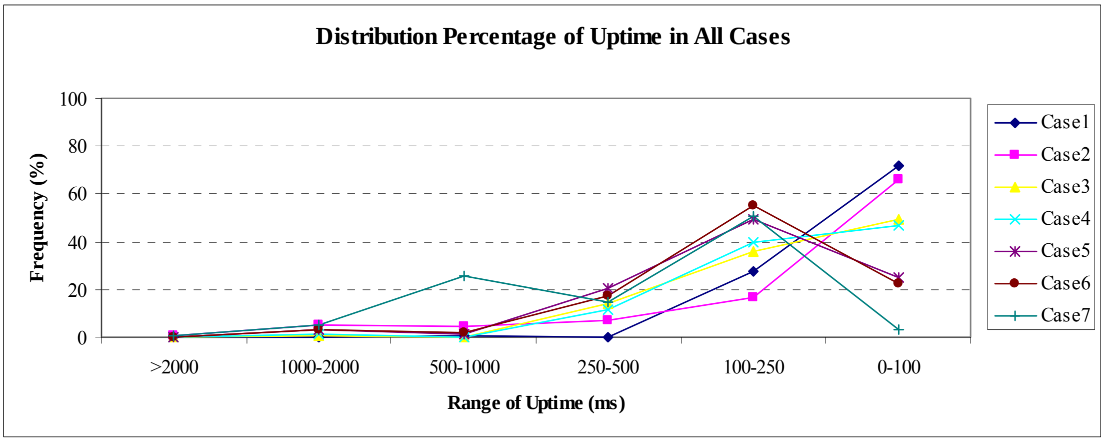


Figure 4.20 Graph of distribution percentage of uptime in all cases

From graph in figure 4.20, ranges of uptime that are lower than 500 ms are acceptable, but ranges of uptime that are higher than 500 ms are poor. The overall percentage of each range of uptime, which is calculated by 28,000 samples of all cases, can be shown as follows.

0-100 ms: 40.72%, this range is the best time.

100-250 ms: 39.24%, this range is good time.

250-500 ms: 12.30%, this range is normal time.

500-1,000 ms: 4.85%, this range is bad time.

1,000-2,000 ms: 2.72%, this range is very bad time.

> **2,000 ms:** 0.17%, this range is the worst time. The caused may be from timeout of dynamic routing.

The most uptime occurs from the dynamic routing. The 92.26% of the overall test has acceptable uptime. In case 1-4, we found maximum range of uptime at 0-100 ms because these cases have many senders. In case 5-7, we found maximum range of uptime at 100-250 ms because these cases have many repeaters.

Table 4.30 Summarize of uptime and task of sensor node in all cases

Case of Network pattern	SN1		SN2		SN3		SN4	
	Average Uptime (ms)	Task	Average Uptime (ms)	Task	Average Uptime (ms)	Task	Average Uptime (ms)	Task
1	86	Sender	103	Sender	93	Sender	117	Sender
2	155	Sender, Repeater	163	Sender, Repeater	168	Sender, Repeater	355	Sender, Repeater
3	91	Sender	285	Sender, Repeater	95	Sender	134	Sender
4	97	Sender	190	Sender, Repeater	230	Sender, Repeater	129	Sender, Repeater
5	104	Sender	334	Repeater	313	Sender, Repeater	154	Sender
6	219	Sender, Repeater	216	Sender, Repeater	313	Sender, Repeater	157	Sender
7	341	Sender, Repeater	486	Repeater	632	Sender, Repeater	141	Sender
Summation of uptime and type of task of sensor node								
Sender	**Weight**	**Average Uptime**	**Repeater**	**Weight**	**Average Uptime**	**Sender-Repeater**	**Weight**	**Average Uptime**
	42.86%	114 ms		7.14%	410 ms		50%	265 ms

From table 4.30, those uptimes are averaged from 10 repetitions of tests (4,000 samples). Tasks of sensor nodes that we found from the test consists of sender only, repeater only, and both sender-repeater. The sender-repeater refers to mixing between sender and repeater case. The percentages time that the sensor nodes act as a sender, repeater, and sender-repeater are 42.86%, 7.14%, and 50%, respectively (full scale of weight is 28 that refer from four SNs in all cases). The average uptime of sender, repeater, and sender-repeater are 114 ms, 410 ms, and 265 ms, respectively. These data refer to repeater, which has low chance to occur, but node device must be placed in area that will be able to use dynamic routing method.

Lifetime of each sensor node can be seen in table 4.31. The lifetime calculation, which the calculation can be divided into lifetime of sender and repeater, will use method and formula in topic 3.4 of chapter 3. We will use uptime and type of task from table 4.30 to calculate. Current of sensor node for 32 MHz PIC's frequency will be used. For case of sender-repeater task in table 4.30, we will use percent of number of hold data for choosing type of calculation. We can show the consideration example of SN1 in case 7 (table 4.29) as follows. The percent of number of hold data has percent of sender as 0.6% and percent of repeater as 99.4%. The repeater type will be used for lifetime calculation of SN1 in case 7.

Table 4.31 Summarize of lifetime and calculation type of sensor node in all cases (sleep time 1,000 ms, battery 2,450 mAh)

Case of Network pattern	SN1		SN2		SN3		SN4	
	Lifetime (Day)	Type of Calculation	Lifetime (Day)	Type of Calculation	Lifetime (Day)	Type of Calculation	Lifetime (Day)	Type of Calculation
1	60	Sender	51	Sender	56	Sender	45	Sender
2	35	Sender	33	Sender	32	Sender	17	Sender
3	57	Sender	20	Repeater	55	Sender	39	Sender
4	54	Sender	29	Sender	24	Repeater	41	Sender
5	50	Sender	18	Repeater	19	Repeater	35	Sender
6	25	Repeater	25	Repeater	19	Repeater	34	Sender
7	18	Repeater	14	Repeater	11	Repeater	38	Sender

*mark number is lowest lifetime, which will be used for system lifetime.

The system lifetime can be calculated by consideration from various tasks (sender, repeater, and sender-repeater) as seen in figure 4.21, 4.22, and 4.23 and compare the node lifetime with various sleep times. In calculation the lifetime, the average uptime from table 4.32 and the current of sensor node for 32 MHz PIC's frequency will be used to calculate by method in topic 3.4. The sender will be calculated by sender lifetime method and uses one packet (n=1). The repeater and the sender-repeater will be calculated by repeater lifetime method and use two packets (n=2). Sender will has the most lifetime, sender-repeater and repeater respectively. Sender has a lifetime that is 72% higher than repeater and 57% higher than sender-repeater. Thus, the Repeater will make the lowest performance (lowest lifetime) system.

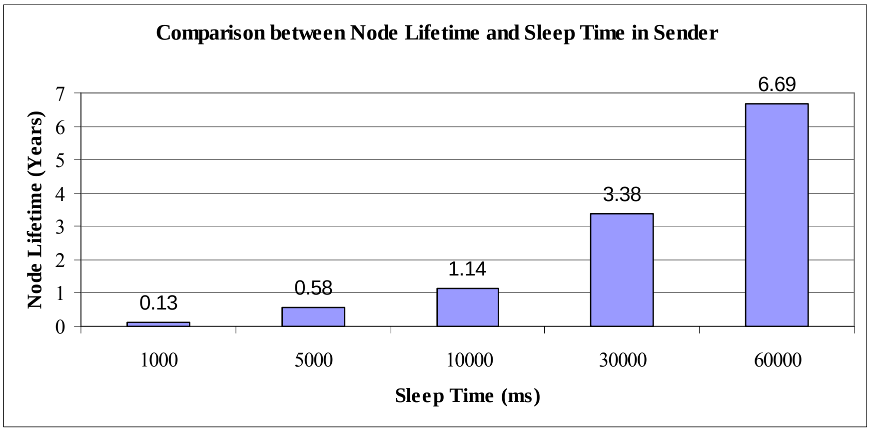


Figure 4.21 Graph of comparison between node lifetime and sleep time in sender

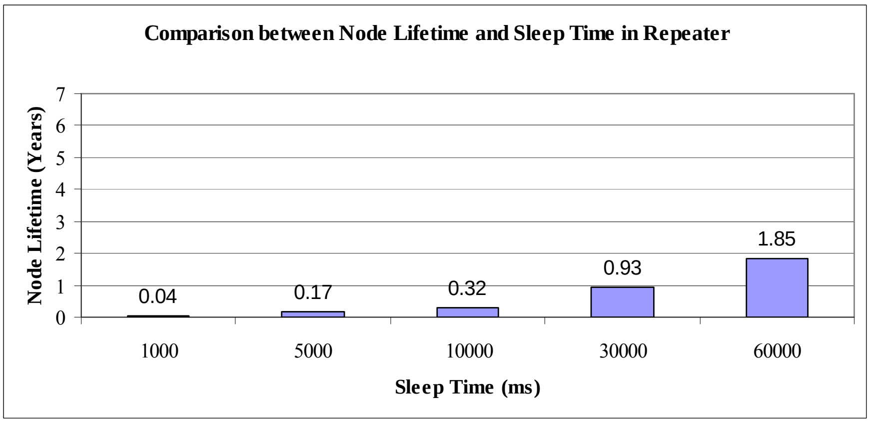


Figure 4.22 Graph of comparison between node lifetime and sleep time in repeater

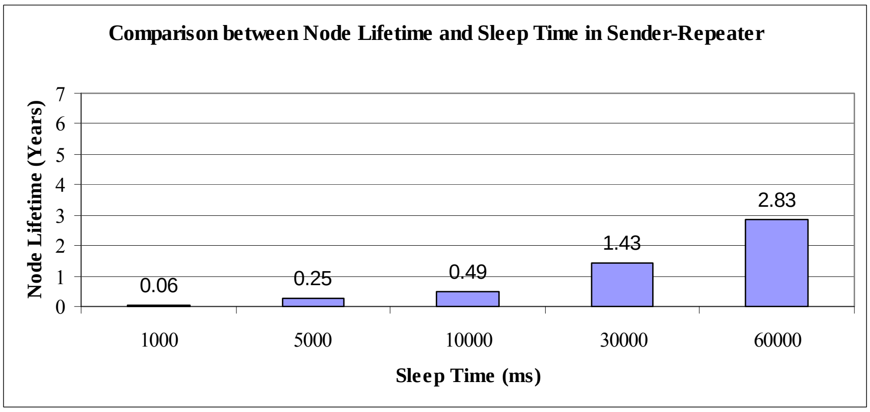


Figure 4.23 Graph of comparison between node lifetime and sleep time in sender-repeater

4.5.3 Effect of Numbers of Connections on Rate of Successful Data Packet Transferred

The rate of successful data packet transferred is calculated from overall testing time. The factor that affects on the success rate is connections between Host-SNs, number of used paths, and number of hops.

In analysis from the communication feature of the test as seen in table 4.32, if the number of connection between Host-SNs is high, the system will has the high success rate. The system will have highest success rate at 3 Host-SNs connections, which the success rate is 2.81 packets/s in case 3. Host does not have sleep time and is designed for fastest task. Thus, the connection between Host-SNs will have high speed task. The host suits with 3 Host-SNs connections. If the connection between Host-SNs is more than 3, system will still work but will slow down. In addition to Host-SNs connections, if the number of used path and number of hop is high, system will have low success rate. Furthermore, percent of hop from the table 4.32 affect on the success rate. If percent of hop is high (such as 4), system will have the low success rate.

Table 4.32 Summarize of communications feature in all cases

Case of Network pattern	Rate of Successful Data Packet transfer (packets/s)	Connections between Host-SNs	Number of Used Path	Data Packer Transferred with n Hop (Percent of Hop)				Maximum Number of Hop
				4 Hop (4 Packet)	3 Hop (3 Packet)	2 Hop (2 Packet)	1 Hop (1 Packet)	
1	2.68	4	4	0	0	0	100	1
2	2.41	4	18	0	0.175	18.975	80.85	3
3	2.82	3	4	0	0	24.825	75.175	2
4	2.71	3	6	0	0.025	24.775	75.2	3
5	2.63	2	3	0	23.45	26.45	50.1	3
6	2.62	2	7	0	25.375	25.95	48.675	3
7	2.55	1	3	24.375	25.175	25.25	25.2	4

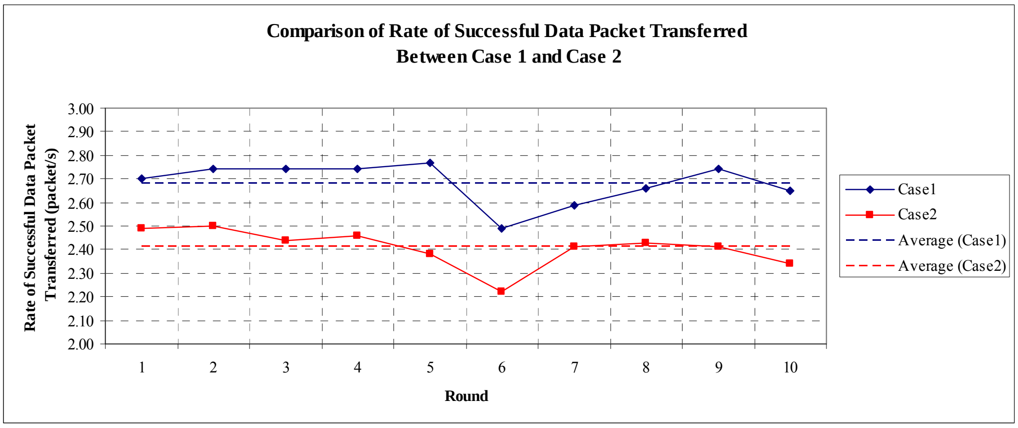


Figure 4.24 Graph of comparison of rate of successful data packet transferred between case 1 and case 2

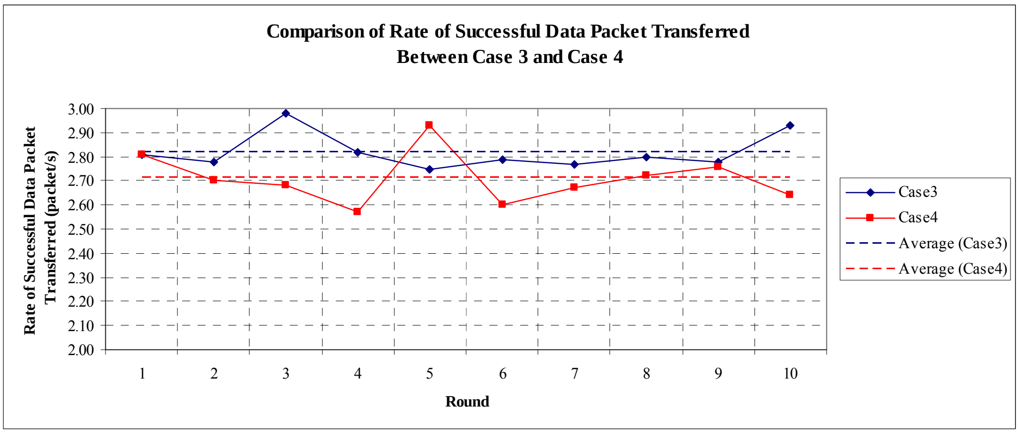


Figure 4.25 Graph of comparison of rate of successful data packet transferred between case 3 and case 4

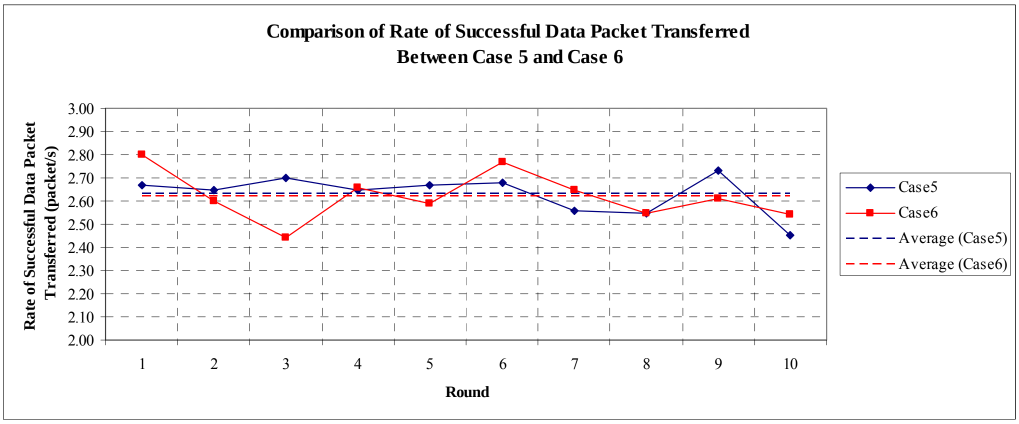


Figure 4.26 Graph of comparison of rate of successful data packet transferred between case 5 and case 6

From figure 4.24, 4.25, and 4.26, we show graph of the rate of successful data packet transferred both 10 sets, which graph will compare the success rate at same number of Host-SNs connections. The graph in figure 4.24, the different line between the success rates can be seen clearly, but not in the graph in figure 4.25 and 4.26 especially the graph in figure 4.26. Thus, the connection between Host-SNs is not only factor for the rate of successful data packet transferred.

The factor that affects on number of used path (can refer to number of connection) consist of number of Host-SNs connection, number of SN-SNs connections, and number of hop. If those numbers are more, the number of used path will be high. The number of used path that is more, which sensor node must consider choosing the path more, will reduce the success rate. In the test, maximum and minimum success rate are 2.82 packets/s (case 3) and 2.41 packets/s (case 2) respectively, which both value just differ 14.54% while both of number of used path (4 paths in case 3 and 18 paths in case 2) differ as 77.78%. Thus, if the number of connection is high, the rate of successful data packet transferred will be decreased a little.

4.5.4 Robustness of Dynamic Routing

The key indicators of dynamic routing performance, as seen in table 4.33, which will be discussed as following.

Available Path: When SN2 fails, the available path for SN4 is required to support dynamic routing method. In such case, the data packet of SN4 will be able to send to host via other nodes that are not as SN2. We will see available path in cases 2, 4, and 6, which are tested in the event that SN2 fails.

Number of Used Path: Since, the number of used path increase, system lifetime and success rate will be decreased. The many paths that occur will increase consideration and uptime of sensor node.

Rate of Successful Data Packet Transferred: The factor of success rate is discussed in topic 4.5.2, which we found the percentage of increasing path at 77.78% which reduce the success rate just 14.54%. This reduction of success rate is very low when we compare with the percentage of increasing path.

System Lifetime: Although, uptime of sensor node will increase when the number of path increases. Those paths increase the chance task occurred in three types (sender, repeater, and sender-repeater). The WSN system that has large proportion of sender and the sender-repeater task will reduce average uptime of sensor node, and increase system lifetime.

Table 4.33 Robustness of dynamic routing function

Case of Network pattern	Available Path for SN4 When SN2 fails	Number of Used Path	Performance	
			Rate of Successful Data Packet transfer (packets/s)	Lifetime (Day)
1	×	4	2.68	45 (SN4)
2	✓	18	2.41	17 (SN4)
3	×	4	2.82	20 (SN2)
4	✓	6	2.71	24 (SN3)
5	×	3	2.63	18 (SN2)
6	✓	7	2.62	19 (SN3)
7	×	3	2.55	11 (SN3)

The previous factors make us know robustness of dynamic routing when it is used. The dynamic routing will increase activities when the system has a large number of used paths, which will increase alternative path. Data packet of all sensor nodes will increase the chance of success (arrive host). The success rate may be decreased a little. The system lifetime may be decreased or increased depending on the chance of task occurred in various types.

CHAPTER V
CONCLUSION

In this research, we created and developed the Wireless Sensor Network (WSN) system for low sampling rate and low power applications. The hardware or node device is designed for low power consumption. The key factor for system efficiency is hardware and software of node device. Especially, software or algorithm of node device will indicate various performances of node device, which is power consumption and communication method.

In the first part, we developed the hardware of node device, which is emphasized on simplicity. The node device is designed to consume low energy, small size (2.5cm×4.2cm) for setting or carrying, and no unnecessarily hardware. The node device consists of two main parts, which are processor unit (PIC18F24J11) and wireless communication (TRW2.4G). The node devices will use analog data from various sensors for operation (one analog data per one node device). The power consumption will depend on feature of electronic devices. In our case, TRW2.4G consumes about 90% and PIC18F24J11 consumes about 10%. Since TRW2.4G consume most of the power, we must control TRW2.4G to work at high efficiency and low uptime (low power consumption) by software.

In the second part, we developed the software of node device. The software of node device consists of many algorithms. The algorithms are programmed into PIC18F24J11, which is responsible in controlling node device. The algorithm will allocate all resources, which consist of memory, power, and operation time. We developed the algorithm to have small operation time for low power consumption. Dynamic Routing is the most important algorithm, which is used in rerouting data when the node device, which maybe out of order, cannot connect to other nodes. Lastly, we developed the algorithm for assigning the task to node device via wire or wireless.

Furthermore, we also developed Host device in both hardware and software for complete WSN system. The WSN system provides the analog data from sensor in percentage, battery status of sensor nodes, and data logging function on a storage device, and display via internet browser (TCP/IP). Our WSN system can be used with many applications such as temperature monitor, pump status monitor, etc. The network pattern of our WSN system consist Star, Cluster-Tree, and Mesh. Those patterns depend on user configuration and Dynamic Routing algorithm.

The node device is designed to be easy to use by anyone. All node devices are installed with the same software. This means that all nodes can use same function (such as dynamic routing, sleep mode). When node device gets the task assigned (SN or GN) from user, the node device will operate automatically, even if the node device has irregular events (such as node loss or removal). Furthermore, our node device is designed to support small to medium network (2 - 100 nodes).

For experiment, we created many additional measurement devices. Gateway device and Host-Meter is created for the test. Host-meter, which often is used for our test, is designed and created for detecting sensor node's uptime, number of data packet, number of priority, and type of task. Those data will be shown and used via RS232. The test can be divided into three parts as follows.

The first test, we tested power consumption of sensor node with respect to PIC's frequencies for finding PIC's frequency of sensor node that provides lowest power consumption. We found that the PIC's frequency at 32 MHz consumes the lowest power consumption. For current measurement, we use multi-meter to measure directly.

The second test, we tested RF-range of sensor node for finding operation range. We define the operation range, which must have the percent of success rate at least 80% of maximum success rate (49 packets/s). We found the operation range of sensor node for indoor at 50 meters and outdoor at 100 meters.

The last test, we tested performance of WSN system, which are system lifetime, rate of successful data packet transferred, and dynamic routing function.

Efficiency of WSN system can be shown as follows.

The system lifetime depends on the minimum lifetime of a sensor node in the system. The lifetime of sensor node will be calculated from uptime and type of task. We found sender has highest lifetime, which is 72% higher than repeater and 57% higher than sender-repeater.

The success rate or rate of successful data packet transferred (packet/s) is based on number of connections, which is emphasized in the number of connections between Host-SNs and number of communication hops. The connection between Host-SNs is the main factor to the success rate because the host has highest capacity in transferring data because it never sleeps. The number of connections between Host-SNs at 3, which provides the highest success rate, will make system be high performance and stability. The number of hops in the test, which is defined in network pattern by location, can create maximum at 4 hops (4 SNs). Furthermore, the connections between SN-SNs also affect on the success rate. All numbers of connections between Host-SNs, the SN-SNs, and number of hops in communication are factors for creating the communication path. The sensor node that has higher number of used paths requires longer time to determine the active path (uptime increases in dynamic routing), will reduce the success rate. In the test, maximum and minimum success rate are 2.82 packet/s (case 3) and 2.41 packet/s (case 2) respectively, which both values just differ 14.54% while both of number of used path (4 paths in case 3 and 18 paths in case 2) differ as 77.78%. Thus, the path increases 77.78% while the success rate decreases 14.54%.

Robustness of dynamic routing can be summarized by using performance of WSN system. The dynamic routing will increase number of used path, which it will increase alternative for choosing the path. Data packet of all sensor nodes will increase the chance of success (data arrival to host). But this method will reduce the success rate, which is very low when we compare with the percentage of increasing path. Furthermore, uptime of sensor node will increase (low lifetime) when number of used path increases, but the percentage of sender and sender-repeater task will increase. The percentage occurred in the test of sender, repeater, and sender-repeater is 42.86%, 7.14%, and 50%, respectively. If sender task occurs at highest percentage, the value of node lifetime and system lifetime will increase. If repeater task occurs at highest percentage, the value of node lifetime and system lifetime will decrease. Thus, when the dynamic routing is used, the success rate may decrease a little and the system lifetime may decrease or increase depending on the chance of task occurred in various types.

Most issues of this research occur from the switching time between TX and RX mode of sensor node. RF of sensor node, which is half-duplex, must switch between TX and RX modes sometimes. If the time between them is not synchronizing, the uptime will be large because TX and RX mode cannot connect together. But we can solve this by retrying a few times, which will be randomized until TX and RX can connect together.

For the extension of our WSN, we can use all cases of network (case 1-7) that already tested and are small networks for creating large network. Our small network consists of four sensor nodes and one gateway node. We can combine those cases by adding many gateway nodes (one gateway node per four sensor nodes) into the large network for supporting the data transferring from many sensor nodes.

For the future work, our WSN system can increase performance by developing hardware and software of node device such as, multi sensors per one node device, full-duplex in wireless communication, and central unit for identifying type of sensor and using many signals in sensor connection. Furthermore, the electronic device in the future may have higher performance in low power consumption, high efficiency, small, and low price. Those points make development easily and can create the WSN system by using many ideas. The need for data communication will increase in the future. This research will be used to response the consumption by application of WSN system.

REFERENCES

1 Wikipedia. IEEE 1451 [Internet]. 2013 [cited 2013 April 10]. Available from http://en.wikipedia.org/wiki/IEEE_1451

2 Wikipedia. IEEE 802.15.4 [Internet]. 2013 [cited 2013 April 10]. Available from http://en.wikipedia.org/wiki/IEEE_802.15.4

3 Wikipedia. Microcontroller [Internet]. 2013 [cited 2013 April 10]. Available from http://en.wikipedia.org/wiki/Microcontroller

4 Wikipedia. RF Transceiver [Internet]. 2013 [cited 2013 April 10]. Available from http://en.wikipedia.org/wiki/Transceiver#RF_Transceiver

5 Wikipedia. Serial Peripheral Interface Bus [Internet]. 2013 [cited 2013 April 10]. Available from http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus

6 Wikipedia. TCP/IP [Internet]. 2013 [cited 2013 April 10]. Available from http://en.wikipedia.org/wiki/TCP/IP

7 Wikipedia. ZigBee [Internet]. 2013 [cited 2013 April 10]. Available from http://en.wikipedia.org/wiki/ZigBee

8 B. Lo, S. Thiemjarus, R. King, and G. Z. Yang. Body sensor network-a wireless sensor platform for pervasive healthcare monitoring. 3th International conference on pervasive computing (Vol. 13). 2005.

9 B. Zhou, C. Hu, H. Wang, R. Guo, and M. H. Meng. A wireless sensor network for pervasive medical supervision. International conference on integration technology IEEE ICIT'07. 2007; 740-744.

10 D. H. Park, and J. W. Park. Wireless sensor network-based greenhouse environment monitoring and automatic control system for dew condensation prevention. Sensors 11(4). 2011; 3640-3651.

11 E. M. Kaur, and B. S. Sohi. Routing Protocols of Wireless Sensor Networks: A Review. International journal of computer networks and wireless communications LJCBWC (Vol. 3). 2013; ISSN: 2250-3501.

12 J. Brassard, and W. Siu. Sniper detection using wireless sensor networks. 2005.

13 J. M. Rabaey, J. Ammer, T. Karalar, S. Li, B. Otis, M. Sheets, and T. Tuan. PicoRadios for wireless sensor networks: the next challenge in ultra-low power design. IEEE International conference on solid-state circuits ISSCC (Vol. 1). 2002; 200-201.

14 J. N. Al-Karaki, and A. E. Kamal. Routing techniques in wireless sensor networks: a survey. Wireless communications IEEE 11(6). 2004; 6-18.

15 J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. Computer networks 52(12). 2008; 2292-2330.

16 K. G. Aravind, T. Chakravarty, M. G. Chandra, and P. Balamuralidhar. On the architecture of vehicle tracking system using wireless sensor devices. International conference on ultra modern telecommunications and workshops IEEE ICUMT'09. 2009; 1-5.

17 M. Unhawiwat, P. Manoyut, and W. Santiamorntat. Wireless sensor network for rubber orchard monitoring and alerting systems. 2th conference on computer engineering and application IEEE ICCEA. 2010; 156-160.

18 P. Kuckertz, J. Ansari, J. Riihijarvi, and P. Mahonen. Sniper fire localization using wireless sensor networks and genetic algorithm based data fusion. Military Communications Conference IEEE MILCOM. 2007; 1-8.

19 S. S. Sonavane, V. Kumar, and B. P. Patil. MSP430 and nRF24L01 based Wireless Sensor Network Design with Adaptive Power Control. ICGST-CNIR Journal (Vol. 8), Issue 2. 2009.

20 T. Gao, D. Greenspan, M. Welsh, R. Juang, nad A. Alm. Vital signs monitoring and patient tracking over a wireless network. 27th Annual International Conference of the Engineering in Medicine and Biology Society IEEE-EMBS. 2005; 102-105.

21 V. Boonsawat, J. Ekchamanonta, K. Bumrungkhet, and S. Kittipiyakul. XBee wireless sensor networks for temperature monitoring. 2th conference on application research and development ECTI-CARD 2010. Chon Buri, Thailand.

22 W. Waiyawoot. A Design of Low Rate Wireless Sensor Network Coordinator with Multi-Network Using Single Processor. 2010.

23 Y. Kim, R. G. Evans, and W. M. Iversen. Remote sensing and control of an irrigation system using a distributed wireless sensor network. IEEE transactions on instrumentation and measurement 57(7). 2008; 1379-1387.

BIOGRAPHY

NAME	Mr. Soraphob Suparchaiparnitchapong
DATE OF BIRTH	14 December 1989
PLACE OF BIRTH	Bangkok, Thailand
INSTITUTIONS ATTENDED	Mahidol University, 2006-2010
	Bachelor of Engineering
	(Mechanical Engineering)
	Mahidol University, 2010-2013
	Master of Engineering
	(Mechanical Engineering)
HOME ADDRESS	146 Muu.4 Banlueng, Photharam,
	Ratchaburi, Thailand 70120
	Tel.081-0167302
	E-mail:sorapop@msn.com
PUBLICATION/PRESENTATION	The 25^{th} Conference of Mechanical Engineering Network of Thailand (A Design and Application of Low Power Wireless Sensor Network)