

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การหาตำแหน่งและติดตามเป้าหมายในโครงข่ายเซ็นเซอร์ไร้สาย

LOCALIZATION AND TRACKING IN
WIRELESS SENSOR NETWORK



T119530

นางสาวกนกวรรณ โจมพรหม

KANOKWAN JOMPROM

นายณัฐพล นาคราช

NUTTAPON NAKARACH

นายธนเศรษฐ์ บ่อคำ

THANASATE BOKAM

เลขหมู่.....
เลขทะเบียน 119530
วัน,เดือน,ปี - 8 S.ค. 2554

b. 10360579
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**LOCALIZATION AND TRACKING IN
WIRELESS SENSOR NETWORK**

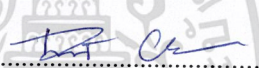


**THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2010**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาบัตร	การตำแหน่งและติดตามเป้าหมายใน โครงข่ายเซ็นเซอร์ไร้สาย	
รายนามนักศึกษา	นางสาวกนกวรรณ โจมพรม	รหัสนักศึกษา 50010004
	นายณัฐพล นาคราช	รหัสนักศึกษา 50010486
	นายธนเศรษฐ์ บ่อคำ	รหัสนักศึกษา 50010629
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมสารสนเทศ	
พ.ศ.	2553	
อาจารย์ที่ปรึกษาปริญญาบัตร	ดร.พนารัตน์ เชิญถนอมวงศ์	
	ผศ.บุญยชนะ ภูระหงษ์	

ปริญญาบัตรฉบับนี้ ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรวิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง


.....
(ดร.พนารัตน์ เชิญถนอมวงศ์)

.....
(ผศ.บุญยชนะ ภูระหงษ์)

อาจารย์ผู้ควบคุมปริญญาบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การหาตำแหน่งและติดตามเป้าหมายในโครงข่ายเซ็นเซอร์ไร้สาย	
รายชื่อนักศึกษา	นางสาวกนกวรรณ โจมพรม	รหัสนักศึกษา 50010004
	นายณัฐพล นาคราช	รหัสนักศึกษา 50010486
	นายธนเศรษฐ์ บ่อคำ	รหัสนักศึกษา 50010629
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมสารสนเทศ	
พ.ศ.	2553	
อาจารย์ที่ปรึกษาปริญญานิพนธ์	ดร.พนารัตน์ เชิญถนอมวงศ์ ผศ.บุญยัชนะ ภูระหงษ์	

บทคัดย่อ

ในปัจจุบันนี้เทคโนโลยีโครงข่ายเซ็นเซอร์ไร้สายถือได้ว่าเป็นปัจจัยสำคัญของระบบอัจฉริยะและยังเป็นตัวแก้ปัญหาในระบบอุตสาหกรรม เช่น ในการใช้งานในพื้นที่ที่อันตรายและยากแก่การเข้าถึง

การระบุตำแหน่ง เป็นสิ่งจำเป็นสำหรับการประยุกต์ใช้งานที่ต้องการทราบตำแหน่งของวัตถุเป้าหมายที่เราสนใจ เป็นการคิดที่จะนำเทคโนโลยีโครงข่ายเซ็นเซอร์ไร้สาย มาใช้ในพื้นที่ที่จีพีเอสไม่สามารถเข้าถึงได้ โดยเฉพาะในบริเวณภายในอาคาร

โครงการนี้ได้ศึกษาวิธีการสำหรับนำมาใช้ในการระบุตำแหน่งได้อย่างมีประสิทธิภาพ โดยใช้โครงข่ายเซ็นเซอร์ไร้สายบนพื้นฐานของมาตรฐานซิกบี โดยตัวแปรที่ใช้คือ ความแรงของระดับสัญญาณที่รับได้จากเซ็นเซอร์ไร้สายและวิธีการระบุตำแหน่งที่ใช้แบ่งเป็น 2 ประเภท คือวิธีการระบุตำแหน่งบนพื้นฐานของการวัดระยะทาง และวิธีการระบุตำแหน่งบนพื้นฐานของ fingerprint นอกจากนี้ยังได้พัฒนาโปรแกรมการเชื่อมต่อกับผู้ใช้แบบกราฟฟิก เพื่อใช้แสดงตำแหน่งของเป้าหมายแบบ real-time อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Localization and Tracking in Wireless Sensor Network	
Student	Miss. Kanokwan Jomprom	Student ID. 50010004
	Mr. Nuttapon Nakarach	Student ID. 50010486
	Mr. Thanasate Bokam	Student ID. 50010629
Degree	Bachelor of Engineering	
Program	Information Engineering	
Year	2010	
Thesis Advisor	Dr. Panarat Cherntanomwong	
	Assoc.Prof.Boonchana Phurahong	

ABSTRACT

Localization is necessary for applications in which the location of the target is needed to be known. Wireless sensor network (WSN) can be used for localization in the area that GPS can not cover, especially for indoor environment.

This thesis studies the effective localization methods using WSN technology based on ZigBee standard. The parameter used for localization is the signal strength received by the wireless sensor. There are two main types of localization methods used in this thesis. One is the range-based technique and another one is the fingerprint technique. Moreover, the user-friendly graphic user interface (GUI) is developed to demonstrate the estimated location of the target in real time.

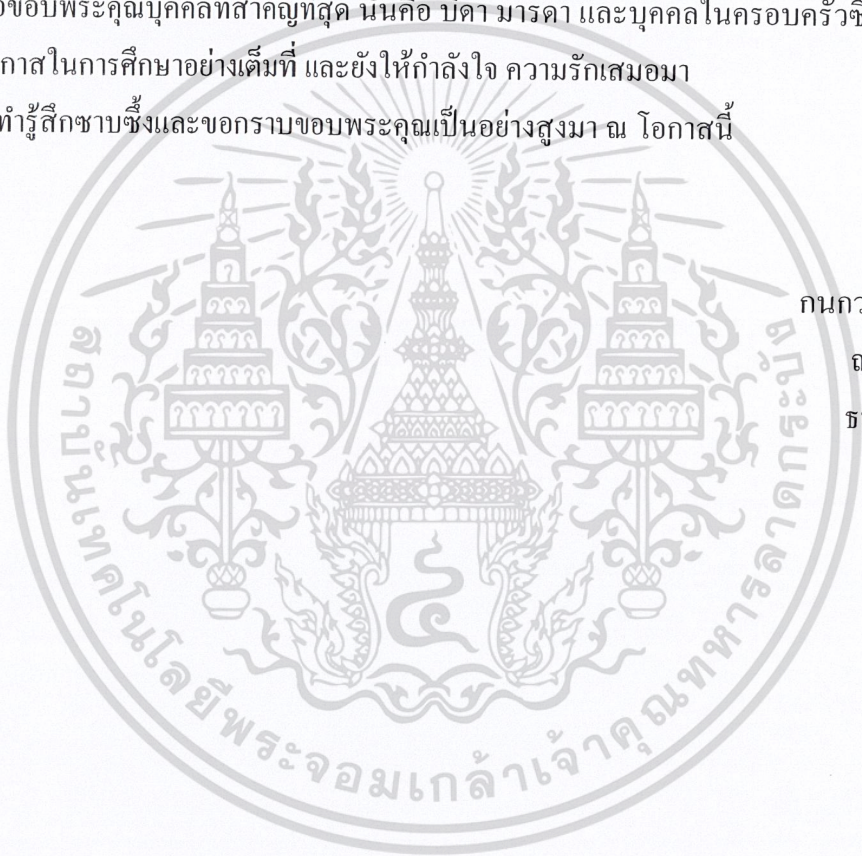
กิตติกรรมประกาศ

ปริญญาโทฉบับนี้ ได้ดำเนินการจนสำเร็จลุล่วงได้ด้วยดี เพราะความร่วมมือของคณะผู้จัดทำ และความกรุณาจาก ดร.พนารัตน์ เชิญถนอมวงศ์ และผศ.บุญยชนะ ภูระหงษ์ อาจารย์ที่ปรึกษา ผู้ซึ่งให้การสนับสนุน

ซึ่งแนวทางให้คำปรึกษาอย่างดีมาโดยตลอด และขอขอบคุณอาจารย์ทุกท่านที่คอยอบรมสั่งสอน และให้ความรู้
ขอขอบคุณสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง คณะวิศวกรรมศาสตร์ ภาควิชา
วิศวกรรมสารสนเทศ ที่ได้เอื้อเฟื้ออุปกรณ์และสถานที่ ในการค้นหาข้อมูลในการจัดทำปริญญาโท และขอ

ขอบคุณเพื่อนๆ พี่ๆ น้องๆ ทุกคนที่คอยเป็นกำลังกาย กำลังใจในการทำงาน
สุดท้ายขอขอบพระคุณบุคคลที่สำคัญที่สุด นั่นคือ บิดา มารดา และบุคคลในครอบครัวซึ่งได้ดูแลเอาใจ
ใส่ พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ ความรักเสมอมา

คณะผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณเป็นอย่างสูงมา ณ โอกาสนี้



กนกวรรณ โจมพรม

ณัฐพล นาคราช

ธนเศรษฐ์ ป่อคำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา III และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ	IV
สารบัญตาราง	IV
สารบัญรูป.....	IV
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบเขตของโครงการ	2
1.4 อุปกรณ์ที่ต้องใช้.....	3
1.4.1 ฮาร์ดแวร์	3
1.4.2 ซอฟต์แวร์	3
1.5 ผลที่คาดว่าจะได้รับ.....	3
1.6 ขั้นตอนการดำเนินงานโครงการ	3
บทที่ 2 ทฤษฎีพื้นฐานที่เกี่ยวข้อง.....	4
2.1 เครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Networks).....	4
2.1.1 ความหมายของเครือข่ายเซ็นเซอร์ไร้สาย	4
2.1.2 วิวัฒนาการของเครือข่ายเซ็นเซอร์ไร้สาย.....	4
2.1.3 สถาปัตยกรรมเครือข่ายเซ็นเซอร์ไร้สาย.....	5
2.1.4 ลักษณะของเครือข่ายเซ็นเซอร์ไร้สาย	6
2.1.5 หน่วยร่วมเซ็นเซอร์.....	6
2.1.5.1 กลุ่มส่วนประกอบหลัก	7
2.1.5.2 กลุ่มส่วนประกอบเพิ่มเติม	8
2.1.6 ระบบเครือข่ายเซ็นเซอร์ไร้สาย.....	8
2.1.6.1 โพรโทคอลสแตค (protocol stack) หรือระดับชั้น	9
2.1.6.2 การพิจารณาระบบเครือข่าย.....	10
2.1.7 การประยุกต์ใช้งานเครือข่ายเซ็นเซอร์ไร้สาย.....	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อ IV และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.1.7.1 แบบซักถาม (querying).....	10
2.1.7.2 แบบแบ่งงาน (tasking).....	11
2.2 เทคโนโลยี ZigBee.....	11
2.2.1 คุณสมบัติ ZigBee	11
2.2.2 ลักษณะของ ZigBee	12
2.2.3 การรับส่งข้อมูลเบื้องต้น.....	12
2.2.4 NETWORK TOPOLOGIES.....	13
2.2.4.1 Star Topology.....	13
2.2.4.2 Peer-to-Peer Topology.....	14
2.2.4.3 Cluster tree Topology.....	14
2.2.5 ZigBee Architecture	15
2.3 Xbee.....	17
2.3.1 XBee Topology.....	17
2.3.1.2 Cluster Tree (Tree) Network	18
2.3.1.3 Mesh	18
2.3.2 Feature Summary ของ XBee	19
2.3.3 กำลังส่งสายอากาศและสัญญาณรบกวนของ XBee	19
2.3.4 XBee Association	19
2.3.5 XBee Addressing	20
2.3.6 XBee Operation Mode	20
2.3.7 Data Throughput ของ XBee	21
2.4 Localization	22
2.4.1 Received Signal Strength Indication (RSSI)	22
2.4.1.1 การวัดค่าหาความแรงของสัญญาณ	23
2.4.2 Link Quality Indicator (LQI).....	24
2.4.3 เทคนิคที่ใช้พื้นฐานของ Fingerprint (Location Fingerprint Technique)	24
2.4.4 เทคนิคที่ใช้พื้นฐานของการหาระยะทาง (Range-based)	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.4.4.1 Time of Arrival (TOA)	25
2.4.4.2 Time Difference of Arrival (TDOA)	27
2.2.4.3 Angle of Arrival (AOA).....	29
2.4.5 Maximum Likelihood Algorithm	30
2.4.6 Min-Max Algorithm	32
2.5 บทสรุป	34
บทที่ 3 การวิเคราะห์และออกแบบระบบ	35
3.1 แนวคิด	35
3.2 การสร้างและออกแบบระบบ	36
3.2.1 รูปแบบเฟรมที่ใช้ในการสื่อสารในระดับชั้นเน็ตเวิร์กเลเยอร์.....	36
3.2.2 ประเภทเฟรม	36
3.2.3 ส่วนซอฟต์แวร์	40
3.2.3.1 ซอฟต์แวร์ส่วนติดต่อผู้ใช้.....	40
3.3 แนวทางการศึกษาและการทดลอง	46
3.3.1 การทดลองเกี่ยวกับ Receive Signal Strength Indicator	46
3.3.2 คุณสมบัติของหน่วยร่วมเซ็นเซอร์และเครือข่ายเซ็นเซอร์ไร้สาย	47
3.4 บทสรุป	47
บทที่ 4 การทดลองและผลการทดลอง	48
4.1 การทดลองเรื่องการวัดค่า RSSI	48
4.2 การทดลองการหาค่าตัวแปรของสมการเพื่อใช้ในการหาระยะทางและหาตำแหน่งของวัตถุ.....	52
4.3 การทดลองหาตำแหน่งของวัตถุด้วยค่าRSSI (Receive strength Signal Indicator)	58
4.4 การทดลองเก็บค่า RSSI (Receive strength Signal Indicator) เพื่อเป็นฐานข้อมูลของ Fingerprint Algorithm ภายในอาคาร	60
4.5 การทดลองเก็บค่า RSSI (Receive strength Signal Indicator) ณ ตำแหน่งต่างๆ ภายในอาคาร	65
4.6 การทดลองหาตำแหน่งของวัตถุ และเปรียบเทียบค่าความผิดพลาดของแต่ละอัลกอริทึม	68
4.7 การทดลองหาตำแหน่งของวัตถุของแต่ละอัลกอริทึม แบบเคลื่อนที่ในอาคาร	100
4.8 บทสรุป	132

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา VI และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 5 บทวิจารณ์และสรุปผล.....	133
5.1 บทสรุป.....	136
5.1.1 กรณีการเคลื่อนที่จากซ้ายล่งไปยังขวาบนของพื้นที่.....	132
5.1.2 กรณีการเคลื่อนที่จากขวาล่งไปยังซ้ายบนของพื้นที่.....	133
5.1.3 กรณีการเคลื่อนที่ตามแนวนอนของพื้นที่.....	134
5.1.4 กรณีการเคลื่อนที่ตามแนวตั้งของพื้นที่.....	135
5.1.5 กรณีการเคลื่อนที่ตามแนวพิกัดที่กำหนดไว้ของพื้นที่.....	135
5.1.6 สรุป.....	136
5.2 ปัญหาและอุปสรรคในการทำงาน.....	136
5.3 สิ่งที่สามารถนำไปพัฒนาต่อไปในอนาคต.....	137
บรรณานุกรม.....	138

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา^{VII} และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางแสดงค่า คุณลักษณะของมาตรฐาน XBee	20
4.1 ตารางแสดงค่าเฉลี่ยของค่าตัวแปร A	55
4.2 ตารางแสดงค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 4 ตัว	62
4.3 ตารางแสดงค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 6 ตัว	64
5.1 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่ง กรณีการเคลื่อนที่จากซ้ายล่างไปยังขวาบนของพื้นที่	133
5.2 ตารางแสดงประสิทธิภาพของการระบุตำแหน่ง กรณีการเคลื่อนที่จากซ้ายล่างไปยังขวาบนของพื้นที่	133
5.3 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่จากขวาล่างไป ยังซ้ายบนของพื้นที่	134
5.4 ตารางแสดงประสิทธิภาพของการระบุตำแหน่ง กรณีการเคลื่อนที่จากขวาล่างไป ยังซ้ายบนของพื้นที่	134
5.5 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวอนพื้นที่	134
5.6 ตารางแสดงประสิทธิภาพของการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวอนพื้นที่	134
5.7 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวตั้งของพื้นที่	135
5.8 ตารางแสดงประสิทธิภาพของการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวตั้งพื้นที่	135
5.9 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่ตามพิกัดที่กำหนดไว้ของพื้นที่	135
5.10 ตารางแสดงประสิทธิภาพของการระบุตำแหน่งกรณีการเคลื่อนที่ตามพิกัดที่กำหนดไว้ของพื้นที่	136

สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างแบบจำลองเครือข่ายเซ็นเซอร์ไร้สาย	5
2.2 ส่วนประกอบของหน่วยร่วมเซ็นเซอร์.....	7
2.3 ระดับชั้นโปรโตคอลของเครือข่ายสื่อสารไร้สาย	8
2.4 ลักษณะการใช้งานเครือข่ายเซ็นเซอร์ไร้สายแบบซักถามและแบบแบ่งงาน	11
2.5 Operating Frequency Bands	13
2.6 การเชื่อมต่อแบบดาว(Star)ในเครือข่ายเซ็นเซอร์ไร้สาย	14
2.7 การเชื่อมต่อแบบ Peer-to-Peer ในเครือข่ายเซ็นเซอร์ไร้สาย	14
2.8 การเชื่อมต่อแบบ Cluster tree ในเครือข่ายเซ็นเซอร์ไร้สาย	15
2.9 ZigBee และ IEEE 802.15.4 Protocol	15
2.10 Star (Broadcast) Network	17
2.11 Cluster Tree (Tree) Network	18
2.12 Mesh Network.....	18
2.13 XBee.....	19
2.14 รูปสายอากาศแบบ Wire (Whip) , Chip ตามลำดับ.....	20
2.15 Internal Data Flow Diagram	21
2.16 แสดงการหาตำแหน่งด้วยการวัดแบบ TOA	26
2.17 แสดงการหาตำแหน่งด้วยการวัดแบบ TDOA	29
2.18 แสดงการหาตำแหน่งด้วยการวัดแบบ AOA	30
2.19 แสดงการหาระยะทาง(d) ระหว่างโหนดอ้างอิง.....	31
2.20 แสดงการหาค่า Min-Max ระหว่างโหนดอ้างอิง.....	33
2.21 แสดงการหาตำแหน่งของวัตถุโดยใช้ Min-Max	34
3.1 แผนภาพการทำงานของระบบ	35
3.2 แสดงภาพรวมการทำงานของระบบ	36
3.3 แสดงรายละเอียดของของแพ็คเก็จขนาด 8 Byte	36
3.4 แพ็คเก็จ Remote AT Command Request (0x17).....	37
3.5 แพ็คเก็จ Explicit Addressing ZigBee Command Frame (0x11)	38
3.6 แพ็คเก็จ ZigBee Transmit Request (0x10).....	39
3.7 ภาพรวมการออกแบบซอฟต์แวร์.....	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา^{IX} และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.8 หน้าของโปรแกรม ก่อนการเชื่อมต่อ	41
3.9 ส่วนของการตั้งค่าการเชื่อมต่อแบบ Serial Port	42
3.10 ส่วนของการใส่เฟรมข้อมูลที่ต้องการส่ง	42
3.11 มุมมองการแสดงตำแหน่งของวัตถุ ระหว่างมุมมองแบบกราฟิก หรือ มุมมองแบบ Text	43
3.12 ตัวอย่างข้อมูลที่โปรแกรมได้ทำการเก็บไว้.....	44
3.13 ส่วนของการแสดงวิธีการใช้งานของโปรแกรม.....	45
3.14 ส่วนการตั้งค่าเพิ่มเติมเมื่อมีการเชื่อมต่อ Serial Port แล้ว	45
3.15 การทดลองวัดค่า RSSI ในระยะต่างๆ.....	46
3.16 การทดลองหาตำแหน่ง โดยใช้ Maximum Likelihood Algorithm	46
4.1 บอร์ด XBee Module	48
4.2 ภาพจำลองแบบการทดลองการจัดวางตำแหน่งของ XBee Module เพื่อหาค่า RSSI	49
4.3 สถานที่ในการทดลองภายในอาคาร	49
4.4 สถานที่ในการทดลองภายนอกอาคาร	50
4.5 กราฟแสดงค่า RSSI ทุกๆช่วง 1 เมตร ในระยะทาง 10 เมตร ภายในอาคาร.....	50
4.6 กราฟแสดงค่า RSSI ทุกๆช่วง 1 เมตร ในระยะทาง 10 เมตร ภายนอกอาคาร.....	51
4.7 กราฟเปรียบเทียบค่า RSSI ทั้งภายในและภายนอกอาคาร.....	51
4.8 แสดงการจัดวางตำแหน่งของ XBee Module เพื่อทำการหาค่า A	54
4.9 การทดลองหาค่า A และ n ทุกๆช่วง 1 เมตร	54
4.10 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 1	55
4.11 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 2	56
4.12 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 3	56
4.13 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 4	57
4.14 กราฟแสดงค่า RMSE ของ Node 1-4 โดยที่ค่ามีค่า n แตกต่างกัน.....	57
4.15 ภาพจำลองการจัดพื้นที่เพื่อที่จะคำนวณหาตำแหน่งของวัตถุ	59
4.16 แสดงการทดลองหาตำแหน่งของวัตถุ	59
4.17 แสดงผลตำแหน่งของวัตถุที่คำนวณได้	60
4.18 ภาพจำลองการจัดพื้นที่เพื่อทำการเก็บค่า RSSI	61
4.19 สถานที่ในการทดลองภายในอาคาร	62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา X และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.20 ภาพจำลองการเก็บค่า RSSI ตามแนวทแยงของพื้นที่ทดลอง จากมุมด้านล่างซ้ายมือถึง มุมด้านบนขวามือ	66
4.21 ภาพจำลองการเก็บค่า RSSI ตามแนวทแยงของพื้นที่ทดลอง จากมุมด้านล่างขวามือถึง มุมด้านบนซ้ายมือ	66
4.22 ภาพจำลองการเก็บค่า RSSI ตามแนวอนกึ่งกลางของพื้นที่ทดลอง	67
4.23 ภาพจำลองการเก็บค่า RSSI ตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง	67
4.24 ภาพจำลองการเก็บค่า RSSI ทั้งหมด 36 ตำแหน่ง ของพื้นที่ทดลอง	68
4.25 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	70
4.26 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	71
4.27 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	71
4.28 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	72
4.29 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	72
4.30 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	73
4.31 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	73
4.32 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	74
4.33 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	74
4.34 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา^{XI} และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.35 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	75
4.36 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	76
4.37 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตาม แนวอนกึ่งกลางของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	76
4.38 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตาม แนวตั้งกึ่งกลางของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	77
4.39 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	77
4.40 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	78
4.41 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	78
4.42 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	79
4.43 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	79
4.44 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	80
4.45 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	80
4.46 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	81
4.47 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	81

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.48 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว.....	82
4.49 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว.....	82
4.50 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่าง ซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	83
4.51 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่าง ขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	83
4.52 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	84
4.53 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	84
4.54 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	85
4.55 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	85
4.56 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	86
4.57 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	86
4.58 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	87
4.59 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	87
4.60 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	88
4.61 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อ XIII และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.62 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่าง ขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	89
4.63 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอน กึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	89
4.64 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอน กึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	90
4.65 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่ง กลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว.....	90
4.66 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอน กึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	91
4.67 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้ง กึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	91
4.68 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้ง กึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	92
4.69 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้ง กึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	92
4.70 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้ง กึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	93
4.71 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว.....	93
4.72 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว	94
4.73 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	94
4.74 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว	95

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.75 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	95
4.76 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	96
4.77 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	96
4.78 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว ..	97
4.79 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	97
4.80 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood อัลกอริทึม, Min Max อัลกอริทึม และ Finger Print แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่าง ซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	98
4.81 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood อัลกอริทึม, Min Max อัลกอริทึม และ Finger Print แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่าง ขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	98
4.82 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood อัลกอริทึม, Min Max อัลกอริทึม และ Finger Print แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอน กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	99
4.83 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood อัลกอริทึม, Min Max อัลกอริทึม และ Finger Print แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้ง กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	99
4.84 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood อัลกอริทึม, Min Max อัลกอริทึม และ Finger Print แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว	100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อที่ ^{XV} และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.85 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวมุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	101
4.86 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวมุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	102
4.87 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวมุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	103
4.88 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวมุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	104
4.89 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวอน กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	105
4.90 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวอน กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	106
4.91 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวตั้ง กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	107
4.92 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood อัลกอริทึมตามแนวตั้ง กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	108
4.93 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	109
4.94 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	110
4.95 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	111
4.96 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	112
4.97 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนวอนกึ่งกลางของพื้นที่ การทดลอง โดยใช้ตัวรับ 4 ตัว	113
4.98 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนวอนกึ่งกลางของพื้นที่ การทดลอง โดยใช้ตัวรับ 6 ตัว	114

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

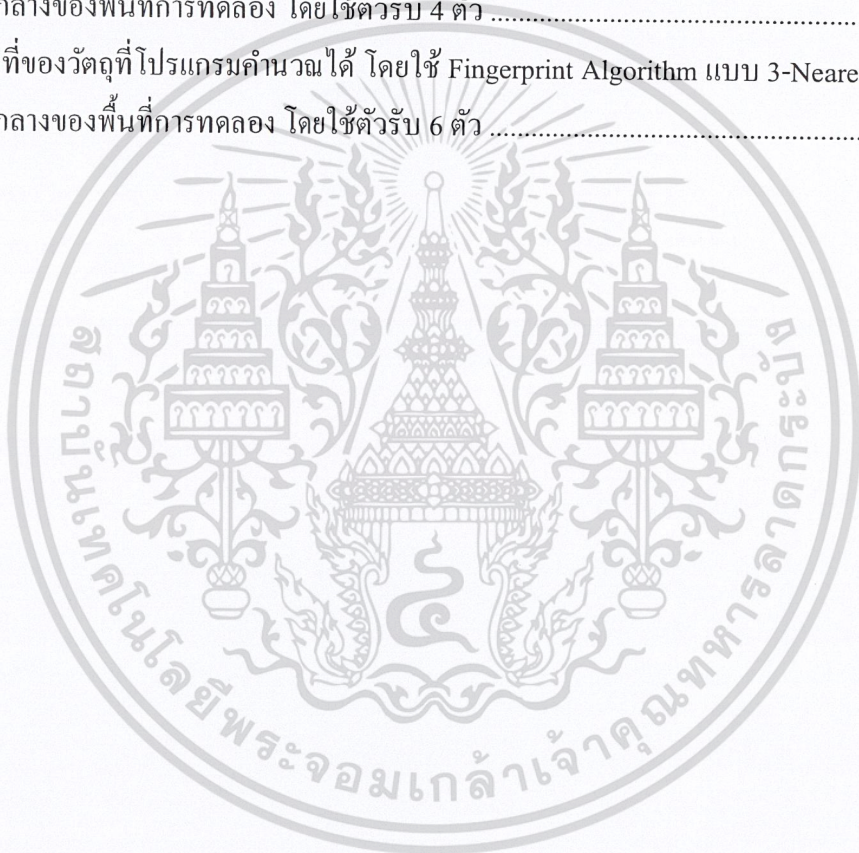
สารบัญรูป(ต่อ)

รูปที่	หน้า
4.99 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนวตั้งกึ่งกลางของพื้นที่ การทดลอง โดยใช้ตัวรับ 4 ตัว.....	115
4.100 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนวตั้งกึ่งกลางของพื้นที่ การทดลอง โดยใช้ตัวรับ 6 ตัว.....	116
4.101 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวมุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	117
4.102 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวมุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	118
4.103 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวมุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	119
4.104 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวมุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	120
4.105 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวอน กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว.....	121
4.106 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวอน กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว.....	122
4.107 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวตั้ง กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว.....	123
4.108 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวตั้ง กึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว.....	124
4.109 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวมุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	125
4.110 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวมุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	126
4.111 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวมุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว	127
4.112 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวมุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว	128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา XVII และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.113 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนว นอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว.....	129
4.114 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตาม แนวนอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว.....	130
4.115 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตาม แนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว.....	131
4.116 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตาม แนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว.....	132



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

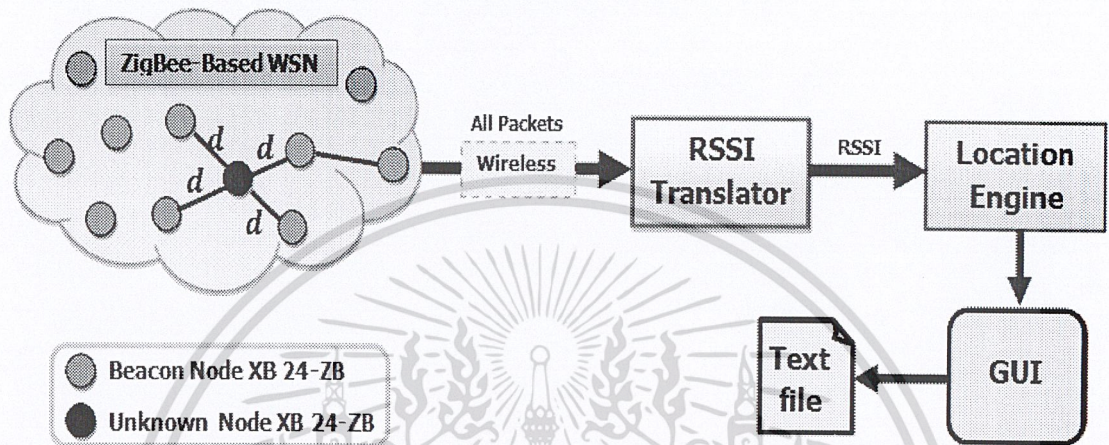
ในปัจจุบันเทคโนโลยีการสื่อสารไร้สายมีการพัฒนาไปอย่างมาก มีการใช้งานหลายรูปแบบ โดยโครงการนี้ได้ทำการศึกษาฟังก์ชันการใช้งานสำหรับการหาตำแหน่ง และการติดตามเป้าหมาย โดยใช้เทคโนโลยีโครงข่ายเซ็นเซอร์แบบไร้สาย (Wireless Sensor Network : WSN) ซึ่งมีบทบาทอย่างมากในการระบุตำแหน่ง เนื่องจากอุปกรณ์มีราคาถูก มีฟังก์ชันในการหาตำแหน่งทั้งภายในและภายนอกอาคาร แต่ก็มีข้อเสียคือ ต้องทำการติดตั้งอุปกรณ์จำนวนมาก เพื่อให้ได้ข้อมูลที่ถูกต้อง

ในการระบุตำแหน่งนั้นสามารถแบ่งได้เป็น 2 เทคนิคหลัก คือ เทคนิคที่ใช้พื้นฐานของการหาระยะทาง (Range-based) จะเป็นการหาตำแหน่งของเป้าหมาย โดยใช้การคำนวณระยะทางระหว่างโหนดอ้างอิงกับเป้าหมาย สามารถหาได้หลายวิธี ขึ้นอยู่กับพารามิเตอร์ที่ใช้ในการค้นหา และเทคนิคที่ใช้พื้นฐานของการคาดคะเน (Proximity-based) เป็นการหาตำแหน่งจากการเปรียบเทียบพารามิเตอร์ของสัญญาณของเป้าหมายเทียบกับพารามิเตอร์ของโหนดใกล้เคียง ตำแหน่งเป้าหมายสามารถประมาณค่าจากโหนดอ้างอิงที่มีค่าพารามิเตอร์ใกล้เคียงกับโหนดเป้าหมายมากที่สุด เรียกได้อีกแบบว่า Location Fingerprint Technique จะมีความแม่นยำกว่าการใช้เทคนิคแรก แต่มีกระบวนการในการวิเคราะห์และประมวลผลที่ซับซ้อนกว่าและใช้เวลามากกว่า

การประยุกต์ใช้งานโครงข่ายเซ็นเซอร์แบบไร้สายในการระบุตำแหน่งนั้น มักนิยมใช้การวัดระดับความเข้มของสัญญาณที่รับได้ (Receive Signal Strength Indicator : RSSI) ในการค้นหาเป้าหมาย โดยกำหนดตำแหน่งที่แน่นอนของกลุ่มโหนดอ้างอิง ซึ่งมักจะถูกเรียกว่า โหนดบีกอน (beacon node) และใช้งานได้โดยการวัดความแรงของระดับสัญญาณระหว่างโหนดบีกอน กับเป้าหมาย โดยใช้พื้นฐานของ 2 เทคนิคที่ได้กล่าวข้างต้น

ทางผู้เสนอโครงการได้สังเกตเห็นว่าอุปกรณ์ WSN ที่จะสามารถรองรับและสามารถนำไปทำการทดลองในส่วนที่สนใจนั้นก็คือ ZigBee ซึ่งเทคโนโลยีนี้เป็นไปตามมาตรฐาน IEEE 802.15.4 และยังมีทั้งพารามิเตอร์ RSSI ซึ่งเราสามารถนำพารามิเตอร์นี้ไปคำนวณหาตำแหน่งและสามารถระบุพิกัดได้ เป็นเหตุให้มีการเผยแพร่และนำเสนอเทคนิค และอัลกอริทึมใหม่ๆ ออกมาอยู่เสมอ อีกทั้ง ZigBee นั้นมีราคาที่ไม่แพง ครอบคลุมพื้นที่กว้าง และรองรับการจัดการโครงข่ายได้หลายประเภทเช่น Mesh, Ad-hoc และ Star topology เป็นต้น

ด้วยเหตุนี้ทางผู้เสนอโครงการ จึงต้องการนำเสนอเทคนิคและวิธีการใหม่ในการระบุตำแหน่งเป้าหมายโดยใช้ WSN บนพื้นฐานของเทคโนโลยี ZigBee โดยมีเป้าหมายหลัก คือพัฒนาระบบวิธีการในการระบุตำแหน่งได้อย่างถูกต้องและมีประสิทธิภาพสูง และสร้างระบบที่ง่ายต่อการพัฒนา ในต้นทุนที่ต่ำและสามารถนำไปใช้งานในเชิงพาณิชย์ อันจะนำไปสู่การพัฒนาประยุกต์ใช้ในงานที่หลากหลาย มีการผลิตซึ่งมีต้นทุนต่ำลงและมีประสิทธิภาพมากขึ้น



รูปที่ 1.1 สถาปัตยกรรมของระบบการหาตำแหน่งของเป้าหมายในโครงข่ายเซ็นเซอร์ไร้สาย

1.2 วัตถุประสงค์ของโครงการ

- เพื่อศึกษาและทำการทดลองเทคนิคในการค้นหาตำแหน่งใน WSN ด้วย RSSI สำหรับการประยุกต์ใช้งาน ทั้งในและนอกอาคาร
- เพื่อพัฒนาอัลกอริทึมที่สอดคล้องกับเทคนิคในกรณีศึกษาของข้อที่แล้ว
- ออกแบบชิ้นงานที่ต้นทุนต่ำ โดยเลือกใช้วัสดุที่มีความเหมาะสม
- สามารถติดตั้งและใช้งานได้จริง

1.3 ขอบเขตของโครงการ

- สามารถสื่อสารกันระหว่างโหนดแต่ละโหนดได้
- สามารถดึงพารามิเตอร์จาก Protocol ของ ZigBee มาใช้ในการระบุตำแหน่งได้
- พัฒนาอัลกอริทึมใหม่ที่ใช้สำหรับการระบุตำแหน่ง
- สามารถนำค่าตัวแปรที่ได้จาก WSN มาแสดงผลบน Graphic User Interface

1.4 อุปกรณ์ที่ต้องใช้

1.4.1 ฮาร์ดแวร์

- XB-24ZB ZigBee Module
- Microcontroller AVR

1.4.2 ซอฟต์แวร์

- โปรแกรมในตระกูล AVR GCC complier
- X-CTU
- โปรแกรม Visual c#

1.5 ผลที่คาดว่าจะได้รับ

- มีความรู้ความเข้าใจเกี่ยวกับการหาตำแหน่ง และการติดตามเป้าหมายโดยใช้เทคโนโลยีโครงข่ายเซ็นเซอร์แบบไร้สาย
- เพิ่มประสิทธิภาพและความแม่นยำในการค้นหาและระบุตำแหน่ง รวมไปถึงการจัดการในด้านพลังงาน
- สามารถออกแบบและพัฒนาเพื่อนำไปใช้ประโยชน์ได้จริง
- สามารถต่อยอดทางความคิดได้ในอนาคต

1.6 ขั้นตอนการดำเนินงาน

D	Task Name	2553									
		กุมภาพันธ์	มีนาคม	สิงหาคม	กันยายน	ตุลาคม	พฤศจิกายน	ธันวาคม	มกราคม	กุมภาพันธ์	มีนาคม
1	ศึกษาทฤษฎี Wireless Sensor Network โครงสร้างของ Zigbee และเครื่องมือที่เกี่ยวข้อง	[Progress bar]									
2	วิเคราะห์ระบบการทำงาน โดยรวม	[Progress bar]									
3	ออกแบบฮาร์ดแวร์และส่วนติดต่อผู้ใช้	[Progress bar]									
4	ออกแบบฐานข้อมูล	[Progress bar]									
5	การนำไปใช้	[Progress bar]									
6	ฮาร์ดแวร์	[Progress bar]									
7	สร้างอรรถทดลอง	[Progress bar]									
8	ซอฟต์แวร์	[Progress bar]									
9	พัฒนาซอฟต์แวร์ส่วนติดต่อผู้ใช้	[Progress bar]									
10	การทดลอง	[Progress bar]									
11	การทดลองระบุตำแหน่งโดยใช้หาวามิเตอร์ RSSI	[Progress bar]									
12	การทดลองระบุตำแหน่งโดยใช้หาวามิเตอร์ RSSI คับอัลกอริทึมต่างๆ	[Progress bar]									
13	เปรียบเทียบ วิเคราะห์และสรุปผล	[Progress bar]									
14	จัดทำเอกสาร	[Progress bar]									

เอกสารนี้เป็นเอกสารทรัพย์สินทางปัญญาสำหรับการแข่งขันเพื่อการศึกษาเท่านั้น มิได้อนุญาตให้เผยแพร่ไปใช้ประโยชน์อื่นใดในการค้า

บทที่ 2

ทฤษฎีพื้นฐานที่เกี่ยวข้อง

2.1 เครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Networks : WSN)

2.1.1 ความหมายของเครือข่ายเซ็นเซอร์ไร้สาย

เครือข่ายเซ็นเซอร์ไร้สาย Wireless Sensor Networks (WSN) คือการใช้อุปกรณ์ sensor เล็กๆ จำนวนมากเพื่อตรวจวัดคุณสมบัติต่างๆของสิ่งแวดล้อมที่เราสนใจและประมวลผลข้อมูลเหล่านั้นเพื่อสร้างองค์ความรู้ใหม่เกี่ยวกับสิ่งแวดล้อมรอบตัวเรา หรือตอบสนองกับการเปลี่ยนแปลงของสภาพแวดล้อมได้โดยอัตโนมัติ WSN เกิดขึ้นจากการผสมผสานระหว่างเทคโนโลยีระบบสมองกลฝังตัวและการสื่อสารไร้สาย ที่ส่งผ่านข้อมูลระหว่างเซ็นเซอร์โหนดด้วยรูปแบบเครือข่ายแบบ ad-hoc จุดเด่นของเครือข่ายเซ็นเซอร์ที่อาศัยโปรโตคอลแบบ ad-hoc คือ ไม่จำเป็นต้องมีการติดตั้งอุปกรณ์พื้นฐานสำหรับเครือข่ายเช่นเดียวกับเครือข่ายแบบ WLAN หรือ GSM นอกจากนี้การออกแบบเซ็นเซอร์โหนดให้มีขนาดเล็กและใช้พลังงานน้อยทำให้ สามารถติดตั้งได้ในสภาพแวดล้อมที่หลากหลาย เทคโนโลยีเครือข่ายเซ็นเซอร์จึงได้ถูกคาดการณ์ว่าจะเป็นเทคโนโลยีหลักในการขับเคลื่อนสู่ยุคของคอมพิวเตอร์ทุกแห่งหน (ubiquitous computing, pervasive computing) ด้วยการสร้างสภาพแวดล้อมประดิษฐ์ในรอบๆตัวของเราทุกคน

2.1.2 วิวัฒนาการของเครือข่ายเซ็นเซอร์ไร้สาย

เครือข่ายเซ็นเซอร์ในยุคแรกพัฒนาขึ้นเพื่อใช้ในการสงคราม ในช่วงสงครามเย็น Sound Surveillance System (SOSUS) ใช้ในมหาสมุทรเพื่อตรวจจับเรือดำน้ำของสหภาพโซเวียต โดยใช้ระบบแถวเซ็นเซอร์ไฮโดรโฟน (hydrophone array) ทำหน้าที่เปลี่ยนพลังงานเสียงเป็นพลังงานไฟฟ้าใต้น้ำและส่งข้อมูลด้วยสายเคเบิลใต้น้ำ ในเวลาเดียวกัน ระบบเครือข่ายเรดาร์ทางอากาศ (networks of air defense radars) ได้รับการพัฒนาเพื่อป้องกันภาคพื้นสหรัฐอเมริกาและแคนาดา ในยุคแรกระบบเครือข่ายมีรูปแบบเป็นลำดับชั้น (hierarchical) การประมวลผลจะทำตามลำดับชั้น และใช้มนุษย์เป็นหลักในการประมวลผลและทำงาน

ยุคที่สองเกิดขึ้นหลังจากการพัฒนาอินเทอร์เน็ตประมาณ พ.ศ.2523 เป็นยุคของการพัฒนาเครือข่ายเซ็นเซอร์แบบกระจายตัว (Distributed Sensor Network) เป็นโครงการวิจัยของ Defense Advanced Research Projects Agency (DARPA) โดยเน้นที่การพัฒนาการประมวลผลข้อมูลแบบกระจายตัว การประมวลผลสัญญาณ (signal processing) การติดตามวัตถุ เครือข่ายใช้โปรโตคอลสื่อสารระดับสูง แต่ด้วยเทคโนโลยีในยุคนั้นทำให้หน่วยร่วมเซ็นเซอร์มีขนาดใหญ่ และถูกออกแบบให้เป็นรถเซ็นเซอร์เพื่อให้เป็นหน่วยร่วมเซ็นเซอร์แบบเคลื่อนที่ได้

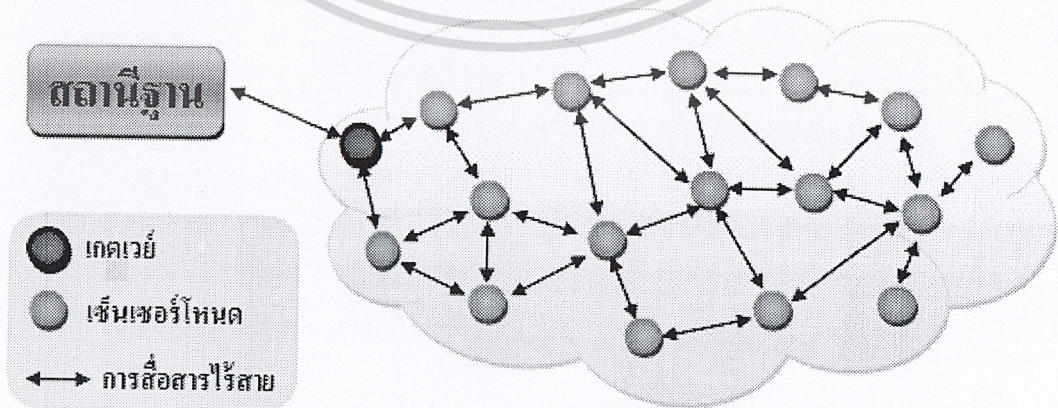
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

เครือข่ายเซ็นเซอร์ยุค ปี พ.ศ.2550 เทคโนโลยีการผลิตไมโครอิเล็กทรอนิกส์ได้รับการพัฒนาถึงระดับที่สามารถผลิตอุปกรณ์อิเล็กทรอนิกส์ที่มีขนาดเล็ก ราคาถูก และประหยัดพลังงาน ซึ่งเหมาะสำหรับสร้างหน่วยประมวลผล และหน่วยส่งข้อมูลของเซ็นเซอร์ไร้สาย และด้วยเทคโนโลยีระบบเครื่องกลจุลภาค การสร้างเซ็นเซอร์ขนาดเล็กและวงจรรีเลย์ทรอนิกส์ขนาดเล็กเกิดขึ้นได้จริง ทำให้เครือข่ายเซ็นเซอร์ไร้สายยุคนี้เข้าใกล้เครือข่ายเซ็นเซอร์ไร้สาย ในจินตนาการของนักวิจัยยุคก่อนๆ อย่างไรก็ตามเครือข่ายเซ็นเซอร์ไร้สายนี้ยังมีการวิจัยอย่างต่อเนื่อง

2.1.3 สถาปัตยกรรมเครือข่ายเซ็นเซอร์ไร้สาย

สถาปัตยกรรมเครือข่ายเซ็นเซอร์ไร้สายประกอบด้วย 3 ส่วนได้แก่ หน่วยร่วมเซ็นเซอร์เกตเวย์ และสถานีฐาน (base station) ดังรูปที่ 2.1 หน่วยร่วมเซ็นเซอร์จำนวนมากฝังตัวในสภาพแวดล้อมเพื่อเก็บข้อมูล โดยแต่ละหน่วยร่วมเซ็นเซอร์ติดต่อสื่อสารแบบไร้สายกับหน่วยร่วมเซ็นเซอร์ข้างเคียง ซึ่งขึ้นอยู่กับความสามารถในการรับส่งแบบไร้สาย แต่ละหน่วยร่วมเซ็นเซอร์ควบคุมและจัดการงานของตัวเอง (self-organize) ทุกๆ หน่วยร่วมเซ็นเซอร์ที่ติดต่อถึงกันทำงานร่วมกัน (collaboration) เป็นเครือข่ายเซ็นเซอร์ไร้สายทำให้แต่ละหน่วยร่วมเซ็นเซอร์สามารถส่งข้อมูลไปหากันได้ แม้ว่าหน่วยร่วมเซ็นเซอร์ปลายทางไม่สามารถติดต่อ กับหน่วยร่วมเซ็นเซอร์ต้นทางได้โดยตรง โดยให้หน่วยร่วมเซ็นเซอร์ระหว่างทางช่วยส่งข้อมูลต่อๆ กันตั้งแต่ต้นทางถึงปลายทาง วิธีการส่งแบบนี้เรียกว่าการส่งแบบมัลติฮอป (multi-hop) เกตเวย์ทำหน้าที่รับ-ส่งข้อมูลระหว่างสถานีฐานและเครือข่ายเซ็นเซอร์ไร้สายโดยเกตเวย์อาจเป็นหน่วยร่วม เซ็นเซอร์ธรรมดาหรือเป็นหน่วยร่วมเซ็นเซอร์ ที่มีความสามารถพิเศษในเครือข่ายเซ็นเซอร์ไร้สาย สถานีฐานทำหน้าที่เก็บข้อมูลที่วัดได้จากหน่วยร่วมเซ็นเซอร์ในเครือข่าย เซ็นเซอร์ไร้สาย ควบคุมการทำงาน และติดต่อกับผู้ใช้งาน หรืออาจติดต่อกับเครือข่ายอื่นๆ เช่น อินเทอร์เน็ต

เครือข่ายเซ็นเซอร์ไร้สาย



รูปที่ 2.1 โครงสร้างแบบจำลองเครือข่ายเซ็นเซอร์ไร้สาย [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

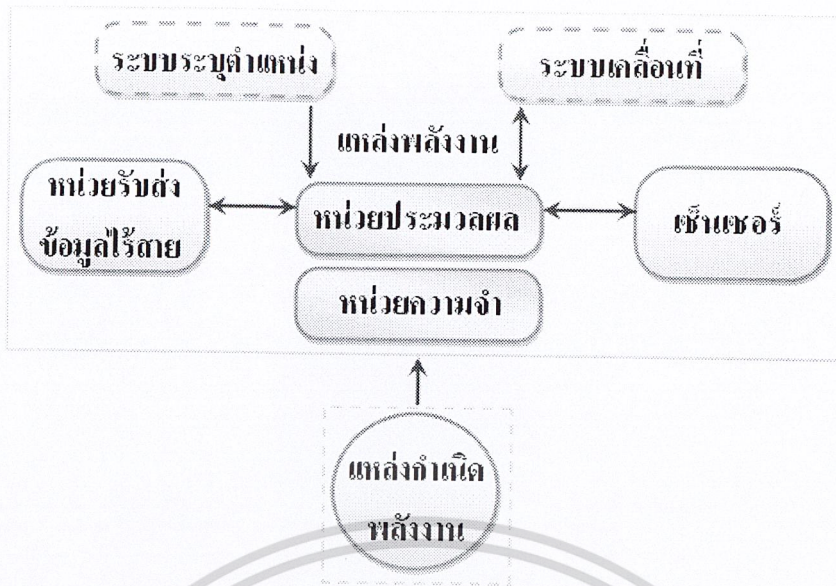
เนื่องจากการทำงานแบบไร้สาย ทำให้แต่ละหน่วยร่วมเซ็นเซอร์ใช้แหล่งพลังงานภายในหน่วยร่วมเซ็นเซอร์เอง หรือในบางกรณีอาจใช้แหล่งกำเนิดพลังงาน เพื่อให้หน่วยร่วมเซ็นเซอร์ทำงานได้อย่างต่อเนื่อง ด้วยเหตุนี้ทำให้เครือข่ายมีการเปลี่ยนแปลงอยู่ตลอดเวลาเนื่องจากหน่วยร่วมเซ็นเซอร์อาจหยุดทำงานเพราะพลังงานหมดหรือกลับขึ้นมาทำงานได้อีก ครั้งเมื่อมีพลังงานเพียงพอ รวมไปถึงในบางเครือข่ายที่มีหน่วยร่วมเซ็นเซอร์ที่เคลื่อนที่ได้ การเปลี่ยนแปลงของหน่วยร่วมเซ็นเซอร์นั้นมีผลต่อโครงสร้าง (topology) ของเครือข่าย และส่งผลถึงเส้นทางในการส่งข้อมูลของหน่วยร่วมเซ็นเซอร์ โดยเส้นทางในการส่งข้อมูลในแต่ละโครงสร้างนั้นขึ้นอยู่กับวิธีการหาเส้นทาง (routing algorithm) ซึ่งวิธีการหาเส้นทางในแต่ละเครือข่าย จะขึ้นอยู่กับวัตถุประสงค์ของเครือข่ายนั้น ๆ

2.1.4 ลักษณะของเครือข่ายเซ็นเซอร์ไร้สาย

- หน่วยร่วมเซ็นเซอร์มีราคาต่ำเพื่อการสร้างเครือข่ายที่ต้องใช้หน่วยร่วมเซ็นเซอร์จำนวนมาก และเหมาะสำหรับการนำไปใช้ครั้งเดียว
- หน่วยร่วมเซ็นเซอร์มีขนาดเล็กเพื่อฝังตัวในสภาพแวดล้อม
- หน่วยร่วมเซ็นเซอร์มีแหล่งพลังงานและความสามารถในการประมวลผลจำกัด
- หน่วยร่วมเซ็นเซอร์และเครือข่ายเซ็นเซอร์ไร้สายสามารถจัดการตัวเองได้ โดยไม่ต้องมีมนุษย์เข้าไปควบคุมหรือช่วยเหลือ
- หน่วยร่วมเซ็นเซอร์จำนวนมาก กระจายตัวครอบคลุมบริเวณทำการของเครือข่ายเซ็นเซอร์ไร้สายเพื่อเก็บข้อมูล
- เครือข่ายเซ็นเซอร์ไร้สายทนทานต่อความเสียหายเมื่อหน่วยร่วมเซ็นเซอร์บางส่วนทำงานไม่ได้
- โครงสร้างเครือข่ายเซ็นเซอร์ไร้สายเป็น โครงสร้างที่ไม่แน่นอนและเปลี่ยนแปลงได้ตลอดเวลา

2.1.5 หน่วยร่วมเซ็นเซอร์

การทำงานของหน่วยร่วมเซ็นเซอร์คือการวัดและเก็บข้อมูลที่ได้จากสภาพแวดล้อม นำข้อมูลไปประมวลผล สร้างเครือข่ายเซ็นเซอร์ไร้สายและส่งข้อมูล ทำให้หน่วยร่วมเซ็นเซอร์มีส่วนประกอบดังรูปที่ 2.2



รูปที่ 2.2 ส่วนประกอบของหน่วยร่วมเซ็นเซอร์ [1]

ส่วนประกอบของหน่วยร่วมเซ็นเซอร์แบ่งออกเป็นสองกลุ่มคือ กลุ่มส่วนประกอบหลักที่จำเป็นเพื่อให้เครือข่ายเซ็นเซอร์ไร้สายทำงานได้ โดยในรูปที่ 2.2 จะเป็นส่วนประกอบที่มีเส้นรอบรูปเป็นเส้นทึบและกลุ่มส่วนประกอบเพิ่มเติมเพื่อเพิ่มคุณสมบัติพิเศษให้กับหน่วยร่วมเซ็นเซอร์ โดยในรูปที่ 2.2 จะเป็นส่วนประกอบที่มีเส้นรอบรูปเป็นเส้นประ

2.1.5.1 กลุ่มส่วนประกอบหลัก

- **เซ็นเซอร์ (sensor)** ทำหน้าที่วัดค่าต่างๆ จากสภาพแวดล้อมตามแต่ชนิดของ เซ็นเซอร์ เช่น ความชื้น อุณหภูมิ ความเข้มแสง ความเร็ว แรงสั่นสะเทือน ความเคลื่อนไหว ความลึก ความเป็นกรดหรือด่าง เป็นต้น
- **หน่วยรับ-ส่งข้อมูลไร้สาย (transceiver unit)** ทำหน้าที่รับ-ส่งข้อมูลแบบไร้สายในย่านความถี่สาธารณะ (ISM band) เพื่อรับ-ส่งข้อมูลระหว่างหน่วยร่วมเซ็นเซอร์ข้างเคียง
- **หน่วยประมวลผล (processing unit)** ติดต่อกับเซ็นเซอร์เพื่อสั่งงานหรือรับข้อมูลที่วัดได้จากเซ็นเซอร์ เพื่อนำไปประมวลผลเป็นข้อมูล จัดเก็บลงในหน่วยความจำ รอการร้องขอข้อมูลหรืออาจส่งข้อมูลทันทีผ่านทางหน่วยรับ-ส่งข้อมูลไร้สาย หน่วยประมวลผลกลางอาจรับข้อมูลจากระบบระบุตำแหน่งเพื่อช่วยในการประมวลผลต่างๆ หรือหน่วยประมวลผลกลางอาจทำหน้าที่ควบคุม การเคลื่อนที่ของหน่วยร่วมเซ็นเซอร์ผ่านทางระบบเคลื่อนที่ นอกจากนี้หน่วยประมวลผลกลางยังทำหน้าที่ประมวลผลเครือข่ายและหาเส้นทางในการส่งข้อมูลของหน่วยร่วมเซ็นเซอร์

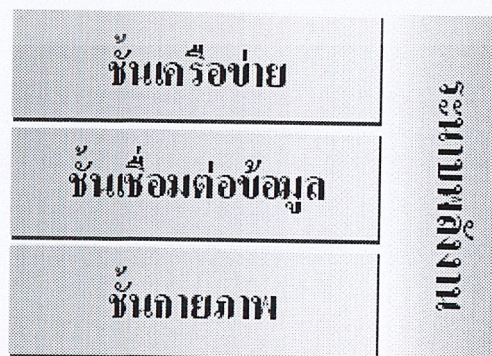
- **แหล่งพลังงาน (power unit)** เก็บสะสมพลังงานและให้พลังงานกับทุกส่วนประกอบบนหน่วยร่วมเซ็นเซอร์ แหล่งพลังงานจะรับพลังงานจากแหล่งกำเนิดพลังงานหากหน่วยร่วมเซ็นเซอร์มีแหล่งกำเนิดพลังงาน

2.1.5.2 กลุ่มส่วนประกอบเพิ่มเติม

- **ระบบระบุตำแหน่ง (positioning unit)** เป็นหน่วยระบุตำแหน่งของหน่วยร่วมเซ็นเซอร์โดยใช้ GPS เพื่อนำข้อมูลตำแหน่งไปใช้ประมวลผล เช่น หาเส้นทางเพื่อส่งข้อมูล หาตำแหน่งสำหรับการเคลื่อนที่ของหน่วยร่วมเซ็นเซอร์ เป็นต้น
- **ระบบเคลื่อนที่ (mobilizing unit)** ทำหน้าที่เคลื่อนย้ายตำแหน่งของเซ็นเซอร์ เพื่อวัตถุประสงค์ต่างๆ เช่น จัดรูปแบบโครงสร้างเครือข่าย ติดตามวัตถุ เคลื่อนที่หาสัญญาณสื่อสาร เป็นต้น
- **แหล่งกำเนิดพลังงาน (power generator unit)** ทำหน้าที่กำเนิดพลังงานจากสิ่งแวดล้อม เช่น พลังงานลม ความร้อน ปฏิกิริยาเคมี การสั่นสะเทือน เป็นต้น ให้เป็นพลังงานไฟฟ้าเพื่อเก็บสะสมและใช้ต่อไป เพื่อชดเชยพลังงานที่ถูกใช้ไป ทำให้ตัวเซ็นเซอร์ไร้สายทำงานได้เป็นเวลานาน

2.1.6 ระบบเครือข่ายเซ็นเซอร์ไร้สาย

เนื่องจากข้อจำกัดในหลายๆ ด้าน เครือข่ายเซ็นเซอร์ไร้สายจึงมีระบบเครือข่ายที่แตกต่างกันขึ้นอยู่กับการใช้งาน เช่น ในระบบเตือนภัย ระบบเครือข่ายต้องการความเร็วสูงในการส่งข้อมูล และจะส่งข้อมูลเมื่อเกิดอุบัติเหตุขึ้นเท่านั้นเพื่อประหยัดพลังงาน ในระบบวัดและควบคุมทางเกษตร ระบบเครือข่ายไม่ต้องการความเร็วสูงในการส่งข้อมูล แต่จะส่งข้อมูลตลอดเวลาโดยอาจมีการรวมข้อมูล (data fusion) ในระหว่างเส้นทางการเดินทางของข้อมูลเพื่อลดจำนวนครั้งของการสื่อสารและประหยัดพลังงาน ดังนั้นการเลือกชนิดและออกแบบเครือข่ายจะต้องพิจารณาให้เหมาะสมกับการใช้งาน



รูปที่ 2.3 ระดับชั้นโปรโตคอลของเครือข่ายสื่อสารไร้สาย [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6.1 โพรโทคอลสแตค (protocol stack) หรือระดับชั้น

เครือข่ายเซ็นเซอร์ไร้สายให้ความสำคัญกับพลังงานมาก เพราะเซ็นเซอร์ไร้สายมักมีแหล่งพลังงานที่จำกัด ทำให้พลังงานมีผลมากกับเซ็นเซอร์ไร้สายและเครือข่ายเซ็นเซอร์ไร้สาย ในด้านการออกแบบจึงปรับปรุงโปรโตคอลสแตคของ Open Systems Interconnection (OSI) model โดยแบ่งเป็นสามชั้นและหนึ่งระนาบดังรูปที่ 2.3 แต่ละชั้นจะทำหน้าที่เฉพาะของตัวเอง คอยให้ความช่วยเหลือชั้นบนและขอความช่วยเหลือจากชั้นล่างที่ติดกับชั้นตัวเอง ส่วนระนาบซึ่งเชื่อมโยงกับทุกชั้นจะควบคุมบริหารจัดการในทุกๆ ชั้นให้ทำงานตามวัตถุประสงค์ของระนาบนั้นๆ อย่างไรก็ตามการใช้งานที่แตกต่างกันก็ส่งผลถึงรูปแบบโปรโตคอลสแตคที่แตกต่างกัน

- **ชั้นกายภาพ (physical layer)** รับผิดชอบการรับ-ส่งสัญญาณไร้สายในด้านกายภาพ เช่น ช่วงความถี่สัญญาณ การมอดูเลต(modulation) การเข้ารหัสระดับช่องสัญญาณ (channel coding) ชั้นกายภาพในประเทศไทยจะใช้ช่วงความถี่สาธารณะและกำลังส่งตามกฎหมายกำหนด
- **ชั้นเชื่อมต่อข้อมูล (data link layer)** รับผิดชอบการรับ-ส่งข้อมูลระหว่างเซ็นเซอร์ข้างเคียง การเข้าใช้ช่องสัญญาณ (medium access control (MAC)) การควบคุมข้อผิดพลาด (error control) ของข้อมูล เพื่อให้การสื่อสารระหว่างหน่วยร่วมเซ็นเซอร์ถูกต้องและเชื่อถือได้ ปัจจุบันการเข้าใช้ช่องสัญญาณของหน่วยร่วมเซ็นเซอร์จะเป็นแบบสุ่มเข้าใช้งาน(random access) ที่เป็นเช่นนี้เพราะการใช้งานเครือข่ายเซ็นเซอร์ไร้สายประกอบไปด้วยหน่วยร่วมเซ็นเซอร์อยู่เป็นจำนวนมากและไม่มีโครงสร้างที่แน่นอนทำให้การควบคุมแบบรวมศูนย์ทำได้ยาก และการใช้ช่องสัญญาณแบบสุ่มทำให้เกิดความเท่าเทียมกันในการเข้าใช้ช่องสัญญาณ
- **ชั้นเครือข่าย (network layer)** รับผิดชอบการรับ-ส่งข้อมูลระดับเครือข่าย เนื่องจากเครือข่ายเซ็นเซอร์ไร้สายใช้การส่งข้อมูลแบบมัลติฮอปเพื่อส่งข้อมูลจากหน่วยร่วมเซ็นเซอร์ไปยังสถานีฐาน การคำนวณหาเส้นทางที่เหมาะสมในการส่งข้อมูลเป็นหน้าที่หลักของชั้นนี้
- **ระนาบพลังงาน(power plane)** รับผิดชอบควบคุมการใช้พลังงานในชั้นต่างๆ ของหน่วยร่วมเซ็นเซอร์ และเครือข่ายเซ็นเซอร์ไร้สายให้มีประสิทธิภาพ โดยอาจประสานงานข้ามชั้น (cross layer) เช่น หน่วยร่วมเซ็นเซอร์ที่เหลือพลังงานน้อย อาจลดพลังงานในการส่งข้อมูลในชั้นกายภาพ โดยประสานงานกับชั้นเครือข่าย เพื่อเลือกเส้นทางที่ควรส่งข้อมูลในกรณีที่ระยะส่งข้อมูลลดลงเนื่องการลดพลังงานในการส่งข้อมูล

2.1.6.2 การพิจารณาระบบเครือข่าย

- การใช้พลังงานอย่างมีประสิทธิภาพ เนื่องจากหน่วยร่วมเซ็นเซอร์มีพลังงานอยู่จำกัด ซึ่งพลังงานที่มีเก็บสะสมอยู่เป็นตัวแปรหนึ่งของอายุการใช้งานหน่วยร่วมเซ็นเซอร์และอายุของเครือข่ายเซ็นเซอร์ไร้สาย
- ความล่าช้าของข้อมูล การใช้งานเครือข่ายเซ็นเซอร์ไร้สายในบางประเภทต้องการให้ได้ข้อมูลเร็วที่สุดหรือมีระดับความล่าช้าของข้อมูลไม่เกินค่าที่กำหนดหลังจากหน่วยร่วม เซ็นเซอร์เก็บข้อมูลได้ เช่น ในงานสัญญาณเตือนภัยสึนามิ ไฟป่า หรือในงานควบคุมที่ความล่าช้าของข้อมูล(delay-sensitive) มีผลต่อประสิทธิภาพการทำงานของระบบควบคุม เป็นต้น
- ความถูกต้องของข้อมูล เครือข่ายเซ็นเซอร์ไร้สายควรจะให้ข้อมูลที่ถูกต้องสูง ซึ่งขึ้นอยู่กับความแม่นยำของเซ็นเซอร์และการประมวลผลข้อมูลในหน่วยร่วม เซ็นเซอร์หรือในเครือข่ายเพื่อให้ได้ข้อมูลที่ถูกต้องที่สุด
- ความทนทานต่อความเสียหาย เหตุการณ์ที่หน่วยร่วมเซ็นเซอร์ไม่สามารถทำงานได้มีโอกาสเกิดขึ้นได้ ดังนั้นเครือข่ายเซ็นเซอร์ไร้สายต้องปรับเปลี่ยน โครงสร้างเพื่อให้เครือข่ายเซ็นเซอร์ไร้สายยังคงทำงานต่อไปได้ เช่น การมีเส้นทางสำรองในการเชื่อมต่อเครือข่ายและในการส่งข้อมูล
- ความสามารถในการส่งข้อมูล แบ่งออกเป็น ความสามารถในการส่งข้อมูลของแต่ละหน่วยร่วมเซ็นเซอร์ และความสามารถในการส่งข้อมูลของเครือข่ายเซ็นเซอร์ไร้สายในกรณีที่ส่งข้อมูลไม่พร้อมกัน ความสามารถในการส่งข้อมูลของแต่ละหน่วยร่วมเซ็นเซอร์เป็นสิ่งสำคัญ แต่ในกรณีที่ส่งข้อมูลพร้อมๆ กันจำนวนมาก ความสามารถในการส่งข้อมูลของเครือข่ายเป็นสิ่งสำคัญ ซึ่งขึ้นอยู่กับการใช้งานเครือข่ายเซ็นเซอร์ไร้สาย
- จำนวนหน่วยร่วมเซ็นเซอร์ในเครือข่ายเซ็นเซอร์ไร้สายเป็นปัจจัยหลักกับค่าใช้จ่ายและความคุ้มค่าในการเลือกใช้ระบบเครือข่ายเซ็นเซอร์ไร้สาย

2.1.7 การประยุกต์ใช้งานเครือข่ายเซ็นเซอร์ไร้สาย

การใช้งานเครือข่ายเซ็นเซอร์ไร้สาย แบ่งออกตามลักษณะการใช้งานได้เป็นสองประเภทคือ ประเภทซักถาม (querying) กับประเภทแบ่งงาน (tasking)

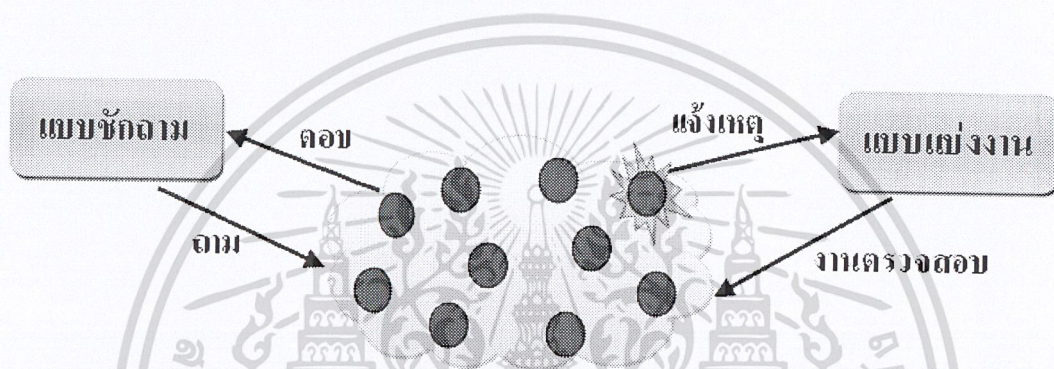
2.1.7.1 แบบซักถาม (querying)

เมื่อเครือข่ายเซ็นเซอร์ไร้สายถูกถามหาข้อมูลหน่วยร่วมเซ็นเซอร์ใดๆที่มีข้อมูลจะตอบกลับผ่านเครือข่ายเซ็นเซอร์ไร้สายดังรูปที่ 2.4 ตัวอย่างเช่น ในงานการเกษตร หน่วยร่วมเซ็นเซอร์

จะถูกถามเมื่อ โปรแกรมที่ใช้งานต้องการใช้ข้อมูลความชื้นและอุณหภูมิ หน่วยร่วมเซ็นเซอร์จะ
 ตอบด้วยข้อมูลที่ถูเก็บบันทึกไว้ในแต่ละหน่วยร่วมเซ็นเซอร์

2.1.7.2 แบบแบ่งงาน (tasking)

เครือข่ายเซ็นเซอร์ไร้สายรับหน้าการตรวจสอบเหตุการณ์ต่างๆเมื่อเกิดเหตุการณ์ที่
 ได้รับมอบหมายให้ตรวจสอบ หน่วยร่วมเซ็นเซอร์จะแจ้งข้อมูลผ่านเครือข่ายเซ็นเซอร์ไร้สายกลับ
 ทันทีดังรูปที่ 2.4 ตัวอย่างเช่น ในการเตือนภัยไฟป่า เครือข่ายเซ็นเซอร์ไร้สายได้รับมอบหมายให้
 ฝ้าวัดความร้อนและควันไฟ หน่วยร่วมเซ็นเซอร์จะแจ้งข้อมูลกลับทันทีเมื่อตรวจวัดความร้อน
 และควันไฟได้เกินระดับที่ตั้งไว้เท่านั้น



รูปที่ 2.4 ลักษณะการใช้งานเครือข่ายเซ็นเซอร์ไร้สายแบบชักถามและแบบแบ่งงาน [1]

2.2 เทคโนโลยี ZigBee

เทคโนโลยีการสื่อสารระยะใกล้เทคโนโลยี ZigBee เป็นการสื่อสารที่ออกแบบขึ้นสำ
 หรับการสื่อสารในเครือข่ายเซ็นเซอร์แบบไร้สาย (Wireless Sensor Network) โดยเริ่มจากการกำ
 หนดมาตรฐานการรับ-ส่งข้อมูลแบบ IEEE 802.15.4 ที่เน้นการสื่อสารแบบประหยัดพลังงาน
 ความเร็วการรับส่งข้อมูลต่ำและมีราคาถูก การสื่อสาร ลักษณะนี้ได้ถูกนำมาใช้สำหรับการสื่อสาร
 ระหว่างเครื่องตรวจวัดหรือเซ็นเซอร์ ที่ต้องการสื่อสารแบบไร้สาย เพื่อลดความยุ่งยากซับซ้อน
 สำหรับการติดตั้ง เช่น บริเวณโรงงานหนึ่งๆ อาจจะต้องใช้จำนวนเซ็นเซอร์ปริมาณมากๆ และ
 เครื่องรับส่งที่มีราคาถูก และประหยัดพลังงาน

2.2.1 คุณสมบัติ ZigBee

คุณสมบัติ ZigBee แบบการสื่อสารระยะใกล้ มีดังนี้

- มีการเชื่อมต่ออย่างซับซ้อนเพื่อรองรับการเชื่อมต่อสำหรับเครือข่ายขนาดใหญ่
- การใช้งานแบบประหยัดพลังงานเพื่อการใช้งานได้ยาวนานจากพลังงานแบตเตอรี่

- การสื่อสารระยะใกล้ในระยะ 10 – 100 เมตร
- เหมาะสำหรับการเฝ้าระวัง (Monitor) และการควบคุม (Control) ใช้งานอุตสาหกรรม งานสิ่งแวดล้อม งานก่อสร้างและงานทางการแพทย์
- เน้นการสื่อสารข้อมูลที่มีความเร็วประมาณ 125 – 250 กิโลบิตต่อวินาที (kbps)

2.2.2 ลักษณะของ ZigBee

มุ่งเน้นการใช้งานภายใต้เครือข่าย IEEE 802.15.4 มาตรฐาน ZigBee รวมถึงคุณสมบัติของการใช้พลังงานต่ำ ที่จำเป็นสำหรับใช้งานใน 2 โหมดหลัก (Tx / Rx หรือ Sleep), ความหนาแน่นของโหนดสูงต่อระบบ, มีต้นทุนต่ำและใช้งานง่าย

คุณลักษณะดังกล่าวมีลักษณะดังต่อไปนี้

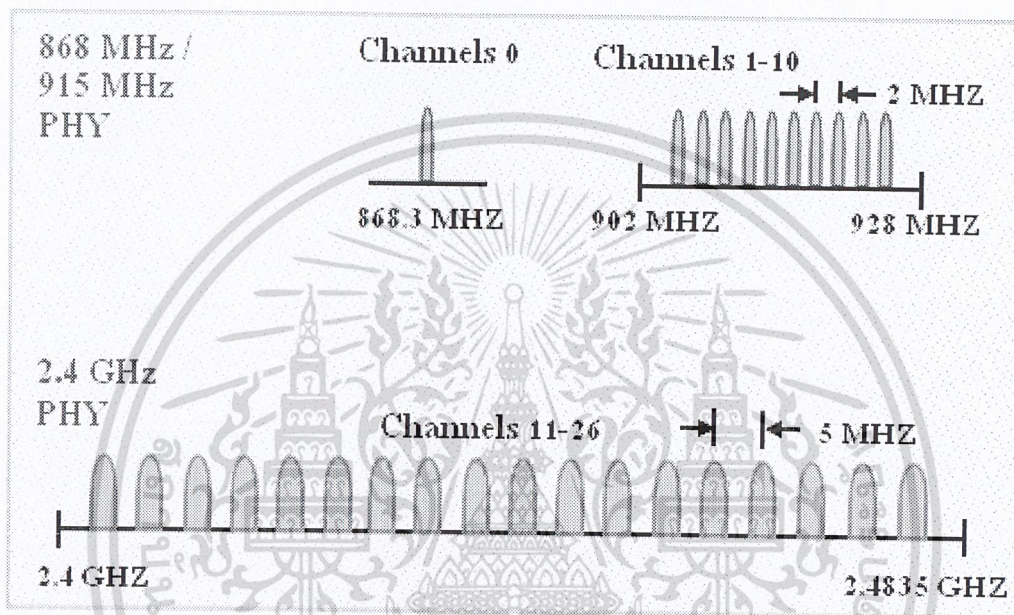
- 2.4 GHz and 868/915 MHz dual PHY modes.
- มี 3 ย่านความถี่ : 2.4-2.4835 GHz, 868-870 MHz และ 902 -928 MHz โดยแต่ละย่าน จะมีช่องสัญญาณ 16 ช่อง , 10 ช่อง และ 1 ช่อง ตามลำดับ
- อัตรารับส่งข้อมูล จะอยู่ที่ 250 Kbps, 40 Kbps, 20 Kbps ตามลำดับเช่นกัน
- มีการจัดสรรการใช้งานที่ 16 bit หรือ 64 บิตขยาย
- การจัดสรรในเรื่องการรับประกันช่วงเวลา (GTSS)
- มีทางเข้าช่องสัญญาณหลายๆ ทาง(CSMA - CA) เพื่อหลีกเลี่ยงการชนกัน
- การเชื่อมต่อโปรโตคอลมีความน่าเชื่อถือ เป็นที่ยอมรับ
- ใช้พลังงานต่ำ
- ตัวตรวจจับพลังงาน(ED)
- ตัวชี้วัดคุณภาพของการเชื่อมโยง(LQI)

2.2.3 การรับส่งข้อมูลเบื้องต้น

มาตรฐาน IEEE 802.15.4 กำหนดขึ้นสำหรับการรับส่งข้อมูลเบื้องต้นในวงจรเครื่องรับส่งวิทยุ (Physical Layer) และการควบคุมการรับส่ง (Link Layer) ดังต่อไปนี้ การสื่อสารใช้คลื่นวิทยุ ความถี่ 2.4 กิกะเฮิรตซ์ (GHz) แบ่งออกเป็น 16 ช่องสัญญาณๆ ละ 5 เมกะเฮิรตซ์ (MHz) สำหรับความถี่ 900 เมกะเฮิรตซ์ (MHz) แบ่งออกเป็น 10 ช่องสัญญาณๆ ละ 2 เมกะเฮิรตซ์(MHz) ใช้การผสมสัญญาณ (modulation) แบบ Offset Quadrature Phase Shift Keying (Offset-QPSK) และใช้การแก้ปัญหาสัญญาณรบกวนแบบ Direct Sequence Spread Spectrum (DSSS) ที่มีอัตราการสเปรคดิ่ง (spreading) 2 ล้าน chip/sec ควบคุมการรับส่งข้อมูลโดยใช้โปรโตคอลแบบ Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA) และเพื่อให้การสื่อสารเครือข่ายเซ็นเซอร์ไร้สายเป็นมาตรฐานเดียวกัน จึงกำหนดมาตรฐานเพิ่มสำหรับการเชื่อมต่อเป็นเครือข่าย (Network) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Layer) และการนำไปใช้งาน (Application Layer) ร่วมกับมาตรฐาน IEEE 802.15.4 เป็นมาตรฐานใหม่ที่กำหนดโดยองค์กร ZigBee Alliance

ดังนั้นจึงอาจกล่าวได้ว่า Zigbee กำหนด ย่านความถี่ใช้งานตามมาตรฐานไว้ 3 ย่านความถี่ คือ ย่าน 2.4 GHz , ย่าน 915 MHz และย่าน 868 MHz โดยแต่ละย่านจะมีช่องสัญญาณ 16 ช่อง, 10 ช่อง และ 1 ช่อง ตามลำดับ ส่วน อัตรารับส่งข้อมูล (ทางอากาศ) จะอยู่ที่ 250 Kbps , 40 Kbps , 20 Kbps ตามลำดับเช่นกัน



รูปที่ 2.5 Operating Frequency Bands [5]

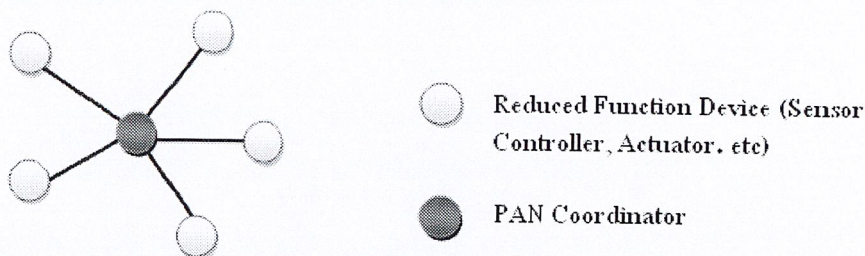
2.2.4 NETWORK TOPOLOGIES

NETWORK TOPOLOGIES หรือ การเชื่อมต่อเป็นโครงข่ายของเครือข่ายเซ็นเซอร์ไร้สาย กำหนดโดยมาตรฐาน ZigBee มีอยู่สองรูปแบบ ได้แก่แบบดาว (Star Topology) และแบบระดับเดียว (Peer-to-Peer Topology)

2.2.4.1 Star Topology

การเชื่อมต่อแบบ Star เหมือนการเชื่อมต่อแบบโครงข่ายจิ้ง (Piconet) ดังแสดงในรูปที่ 2.6 ใน Topology แบบ Star การสื่อสารเป็นที่ยอมรับกันระหว่างอุปกรณ์และการควบคุมส่วนกลางเดียว (Single Central Controller) ที่เรียกว่า PAN Coordinator ที่เป็นตัวขับเคลื่อนหลัก การใช้ประโยชน์จากเครือข่ายที่มี Topology แบบนี้ เช่น แบบ Home Automation, Personal Computer(PC) Peripherals, Toys และ Games หลังจาก FFD Active สำหรับช่วงเวลาแรกมันอาจจะสร้างเครือข่ายของมันเองและกลายมาเป็น PAN Coordinator ในแต่ละ Star Network จะเลือก

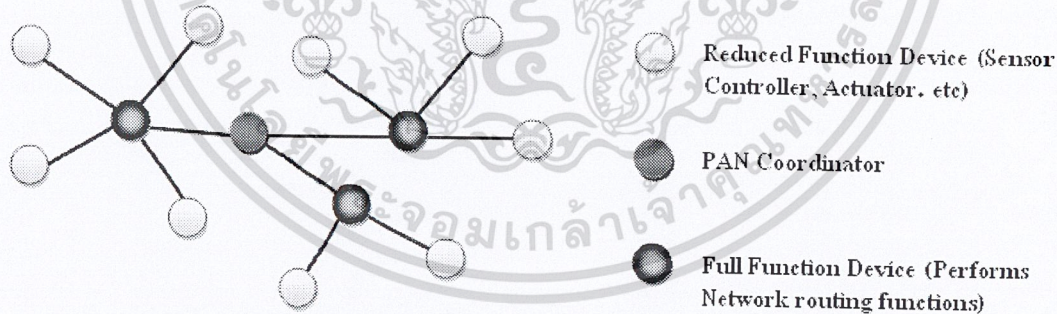
PAN Id ซึ่งปัจจุบันไม่ได้ถูกใช้โดยเครือข่ายอื่นๆภายใน Radio Sphere นี้จะยอมให้แต่ละ Star Network ทำงานอย่างอิสระ



รูปที่ 2.6 การเชื่อมต่อแบบดาว(Star)ในเครือข่ายเซ็นเซอร์ไร้สาย [2]

2.2.4.2 Peer-to-Peer Topology

สำหรับการเชื่อมต่อแบบระดับเดียวเป็นการเชื่อมต่อเพื่อขยายโครงข่ายให้กว้างออกไปดังแสดงในรูปที่ 2.7 โดยในการเชื่อมต่อนั้นจะต้องมี PAN Coordinator ซึ่งเชื่อมต่ออยู่กับอุปกรณ์เซ็นเซอร์ที่มีความสามารถเต็ม (Full-Function Device) ในการหาเส้นทาง (Routing) และอุปกรณ์เซ็นเซอร์แบบที่มีความสามารถลดลง (Reduced-Function Device) จะเชื่อมต่อกับอุปกรณ์ที่มีความสามารถเต็มอีกทีหนึ่ง ซึ่งอุปกรณ์ที่มีความสามารถเต็มเทียบเท่าได้กับอุปกรณ์บลูทูธแบบ Master ที่มีหน้าที่เชื่อมต่อกับ Master ตัวอื่นๆ เพื่อให้เกิดเครือข่ายที่ใหญ่ขึ้น



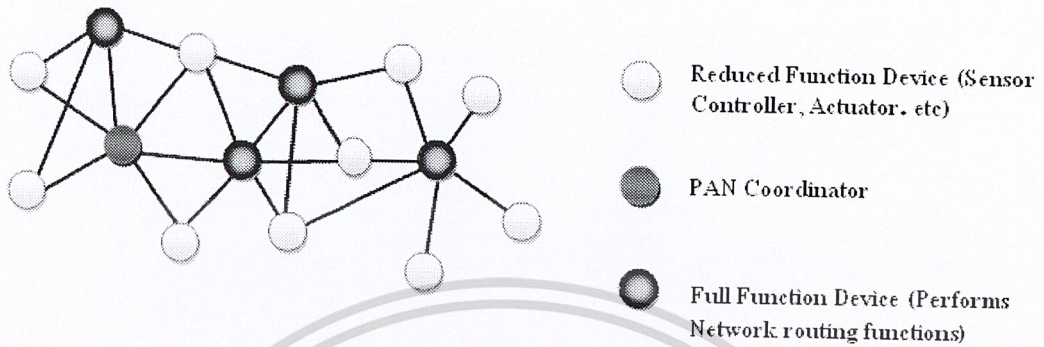
รูปที่ 2.7 การเชื่อมต่อแบบ Peer-to-Peer ในเครือข่ายเซ็นเซอร์ไร้สาย [2]

2.2.4.3 Cluster tree Topology

ในการสร้างระบบ Cluster tree PAN Coordinator จะใช้ตัวเองเป็น Cluster แรกเรียกว่า Cluster Head (CLH) และ Cluster Identifier (CID) เป็นศูนย์ หลังจากนั้นทำการเลือก PAN Identifier และทำการ Broadcasting Beacon Frame ไปยังอุปกรณ์ข้างเคียง และเมื่ออุปกรณ์ตัวอื่นได้รับก็จะส่ง Request ไปที่ CLH จากนั้น CLH ทำการยอมรับและเพิ่มชื่ออุปกรณ์ใน Neighbor

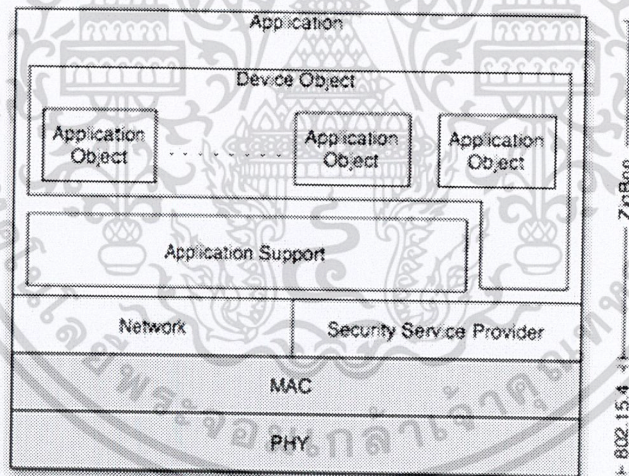
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

List แบบเป็นลูกข่าย(child) ในขณะที่อุปกรณ์ที่เข้ามาใหม่จะเพิ่มชื่อ CLH ตัวนั้นใน Neighbor List ว่าเป็นแม่ข่าย(parent) หลังจากนั้นก็ทำการส่ง Beacon ไปยังอุปกรณ์ตัวอื่นๆ ต่อไปดังแสดงในรูปที่ 2.8



รูปที่ 2.8 การเชื่อมต่อแบบ Cluster tree ในเครือข่ายเซ็นเซอร์ไร้สาย [2]

2.2.5 ZigBee Architecture



รูปที่ 2.9 ZigBee และ IEEE 802.15.4 Protocol Stack [5]

ZigBee นำ Physical Layer และ MAC Layer ของ IEEE 802.15.4 ซึ่งเป็นมาตรฐานการกำหนดการสื่อสารไร้สายแบบ WPAN (Wireless Personal Area Network) มาทำงานใน Layer ที่ต่ำกว่า เช่น เรื่องของ ระดับกำลังสัญญาณ , Link Quality , Access control , Security และ Application Layers ซึ่งสถาปัตยกรรมของ ZigBee แสดงดังรูปที่ 2.9

ส่วนลักษณะของ ZigBee คือมีทางเข้าช่องสัญญาณโดยการใช้ Carrier Sense Multiple Access with Collision Avoidance (CSMA - CA) หรือมีทางเข้าช่องสัญญาณหลายๆ ทาง เพื่อหลีกเลี่ยงการชนกันของสัญญาณที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลียงการชนกันชนิดอุปกรณ์ของ ZigBee มีอยู่ 2 ชนิดคือ แบบ Physical Device และ Logical Device และแบบ Physical Device สามารถแบ่งได้ 2 ประเภท คือ

- **Full Function Device : FFD** เป็นเราเตอร์ที่เป็นสื่อกลางในการส่งข้อมูลจากอุปกรณ์อื่นๆ ใช้พลังงานจาก Power Line ทำงานได้ในทุก Topology และสามารถทำเป็นจุดเชื่อมต่อกันได้ โดย FFD สามารถที่จะทำงานได้สาม mode คือ PAN coordinator , Coordinator หรือ Device ซึ่งใน ZigBee Network ต้องมี FFD อย่างน้อยหนึ่งตัวซึ่งจะทำหน้าที่เป็น PAN coordinator โดย FFD สามารถที่จะติดต่อได้ทั้ง FFD และ RFD แต่ RFD จะสามารถติดต่อได้เพียง FFD เท่านั้น
- **Reduced Function Device : RFD** เหมาะแก่การเชื่อมต่อภายในเครือข่ายในพลังงานจากแบตเตอรี่ ไม่สามารถถ่ายทอดข้อมูลจากอุปกรณ์อื่นๆ ได้ ทำได้ง่ายในเครือข่ายที่เป็นแบบ Star แบบ Logical Device มี 3 ประเภท คือ ZigBee Coordinators เป็นจุดที่ประสานเชื่อมต่อ ทำหน้าที่ในการจัดเก็บข้อมูลในเครือข่าย ZigBee Routers ทำหน้าที่จัดการเส้นทางของข้อความที่ส่งผ่านภายใน โครงข่ายระหว่างคู่ของโหนดใดๆ และ ZigBee และ Devices เป็นโหนดที่อยู่ในส่วนของผู้ใช้งาน โดยสามารถเป็นได้ทั้งแบบ RFD และ FFD

ดังนั้น ZigBee แบ่งตามลักษณะการทำงานได้ 3 แบบ คือ

1. **Coordinator** มีหน้าที่สร้างการสื่อสาร เชื่อมโยงเครือข่าย ระหว่าง End Device กับ Router หรือ Coordinator กับ Coordinator ด้วยกัน หรือ Coordinator กับ Router กำหนด address ให้กับ device ที่อยู่ในวงเครือข่าย ไม่ให้ซ้ำกัน ดูแลจัดการเรื่องการ Routing เส้นทาง ซึ่งเทียบได้กับ FFD

2. **End Device** เป็นอุปกรณ์ปลายทางสุด ซึ่งจะใช้รับสัญญาณจาก Sensor ที่ปลายทาง โดยที่ใช้พลังงานต่ำในการทำงาน เทียบได้กับ RFD หรือ FFD บางกรณี ขึ้นอยู่กับ sensor ที่ใช้

3. **Router** มีหน้าที่รับส่งข้อมูล ในเส้นทางต่าง ๆ ของเครือข่าย ซึ่งเทียบได้กับ FFD

ส่วนสำคัญสำหรับเครือข่ายเซ็นเซอร์ไร้สาย คือกลไกการประหยัดพลังงานซึ่งตัวอุปกรณ์เซ็นเซอร์สามารถกำหนดระยะเวลาในการเข้าสู่การหลับหรือพักการทำงาน (Sleep Mode) โดยตัวอุปกรณ์จะทำหน้าที่ลดการใช้พลังงานให้เหลือน้อยที่สุดเพื่อเป็นการประหยัดพลังงาน การกำหนดการหลับนี้ทำได้โดยการร้องขอไปที่อุปกรณ์ Full Function Device เพื่อบอกระยะเวลาที่จะทำการหลับและเมื่อมีการส่งข้อมูลจากอุปกรณ์ตัวอื่นมายังอุปกรณ์ที่หลับอยู่ อุปกรณ์ Full Function Device จะเก็บข้อมูลไว้ให้ชั่วคราว และถามหาอุปกรณ์ตัวนั้นเป็นระยะเมื่ออุปกรณ์ตัว

นั่นต้นหรือพร้อมทำงานต่อจะได้รับการถามหา อุปกรณ์ตัวนั้นจึงส่งการร้องขอข้อมูลที่ได้เก็บไว้ให้ และจึงทำการรับส่งข้อมูลจนได้รับข้อมูลครบสมบูรณ์ต่อไป

2.3 XBee

XBee เป็นอุปกรณ์ที่ใช้รับส่งข้อมูลแบบไร้สาย และสามารถสร้างระบบเครือข่ายขึ้นมาได้โดย ZigBee ได้อ้างอิง มาตรฐานตาม IEEE 802.15.4 โดยสามารถแบ่งหน้าที่การทำงานของ XBee ตามรูปแบบการสื่อสารในลักษณะต่างๆ ได้ อาทิ Coordinator, Router หรือ End Device

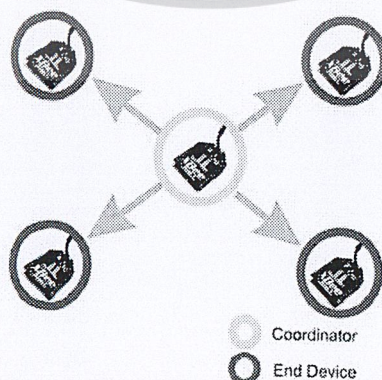
XBee เป็นอุปกรณ์ที่มี Microcontroller และ RF IC อยู่ภายใน ทำหน้าที่เป็น อุปกรณ์ transceiver (อุปกรณ์รับ-ส่งสัญญาณ) แบบ แบบ Half Duplex ย่านความถี่ 2.4 GHz มีการจัดการโดยใช้พลังงานต่ำ ใช้งานง่าย มี interface ที่ใช้รับและส่งข้อมูลกับ XBee เป็น UART (TTL) ซึ่งสำหรับทางด้านไมโครคอนโทรลเลอร์ สามารถนำขาที่ใช้ติดต่อสื่อสาร UART ของ XBee ต่อเข้ากับ UART ของไมโครคอนโทรลเลอร์ได้เลย

2.3.1 XBee Topology

ในการสร้างโครงข่ายไร้สายของ ZigBee นั้น จะต้องประกอบด้วยโหนด จำนวนอย่างน้อยที่สุด 2 ชนิด คือ Coordinator node และ node ลูกข่าย ชนิดใดชนิดหนึ่ง (Router/End device) จึงจะสามารถสื่อสารและทำงานในรูปแบบของ PAN (Personal area network) ได้โดย ZigBee สามารถแบ่งรูปแบบ เครือข่าย ได้เป็น 3 รูปแบบ ดังนี้

2.3.1.1 Star (Broadcast)

การเชื่อมต่อแบบ Star หรือ แบบ Broadcast เป็นการรับส่งข้อมูลแบบไม่เฉพาะเจาะจง จุดหมายปลายทาง หรือ XBee ทุกตัวที่อยู่ในระบบเครือข่ายเดียวกัน สามารถรับข้อมูลทุกข้อมูลได้ทุกตัว

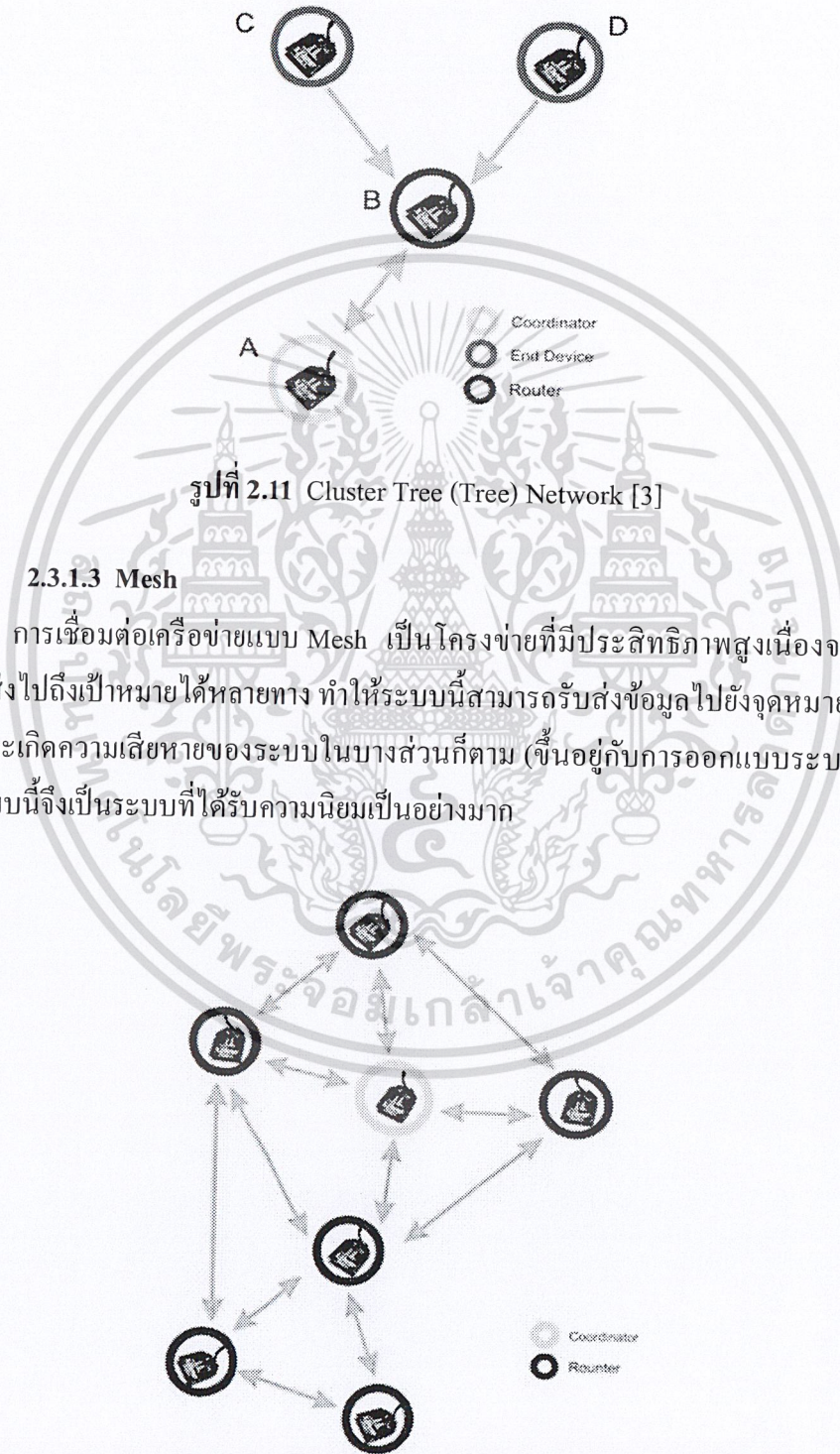


รูปที่ 2.10 Star (Broadcast) Network [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังขอสงวนเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.2 Cluster Tree (Tree) Network

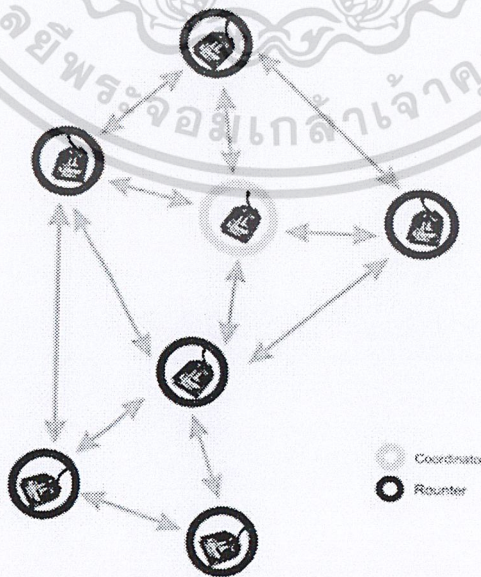
เป็นการรับส่งข้อมูลแบบส่งผ่าน เช่น A ต้องการติดต่อกับ C แต่ C อยู่ไกลจาก A จน A ไม่สามารถติดต่อกับ C ได้ แต่พอดีมี B ที่อยู่ระหว่าง A กับ C ดังนั้น Cluster Tree จะใช้ B เป็นเหมือนตัวกลางเชื่อมการติดต่อ (Repeater) ระหว่าง A กับ C



รูปที่ 2.11 Cluster Tree (Tree) Network [3]

2.3.1.3 Mesh

การเชื่อมต่อเครือข่ายแบบ Mesh เป็นโครงข่ายที่มีประสิทธิภาพสูงเนื่องจาก ข้อมูลสามารถส่งไปถึงเป้าหมายได้หลายทาง ทำให้ระบบนี้สามารถรับส่งข้อมูลไปยังจุดหมายปลายทางได้ แม้จะเกิดความเสียหายของระบบในบางส่วนก็ตาม (ขึ้นอยู่กับการออกแบบระบบของผู้ใช้ด้วย) ระบบนี้จึงเป็นระบบที่ได้รับความนิยมเป็นอย่างมาก

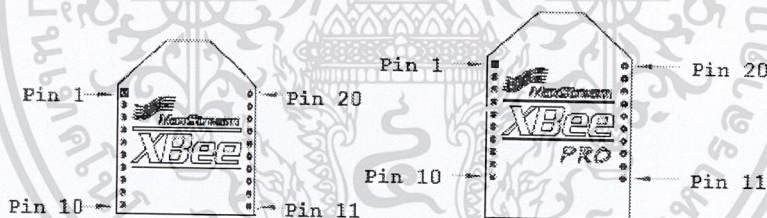


รูปที่ 2.12 Mesh Network [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 Feature Summary ของ XBee

- Operating Frequency ISM Band 2.4 GHz
(ISM Band หมายถึงย่านความถี่ใช้งานเพื่อการวิจัยซึ่งจะอนุญาตให้ใช้กับอุตสาหกรรม (Industrial) วิทยาศาสตร์ (Scientific) และ ทางการแพทย์ (Medical) รวม เป็น ISM)
- มีสายอากาศให้เลือกใช้หลายแบบ คือ แบบ Chip Ant, Whip Ant, UFL con, RPSMA con โดย 2 แบบหลังเราต้องไปหาเสาอากาศย่าน 2.4 GHz ที่เป็น connector แบบ UFL หรือ SMA
- Supply Voltage อยู่ที่ 2.8-3.4 V
- Power Down Current < 10 uA
- มี RF data rate อยู่ที่ 250 Kbps (เป็นส่วนของสัญญาณที่ส่งผ่านอากาศ)
- มี Serial interface data rate อยู่ระหว่าง 1200–115200 Bps (เป็นส่วนที่ติดต่อสื่อสารกับไมโครคอนโทรลเลอร์)
- เป็น Spread Spectrum ชนิด DSSS (Direct Sequence)
- การกำหนด addressing มีลำดับลักษณะคือ กำหนด PAN ID สำหรับเครือข่ายหนึ่ง ๆ , กำหนด Channel และ กำหนด address ของแต่ละตัว



รูปที่ 2.13 XBee [4]

2.3.3 กำลังส่งสายอากาศและสัญญาณรบกวนของ XBee

XBee นั้นใช้ย่านความถี่ 2.4 GHz ซึ่งเป็นย่านเดียวกันกับ Bluetooth หรือ Wireless Lan สัญญาณมันจะเกิดการกวนกัน เช่น มี wireless lan ใช้ Access Point 1 ตัว และเราก็ยังใช้เครื่องโทรศัพท์แบบถือไปมาได้ (2.4 GHz cordless telephones) ซึ่งก็ใช้ย่าน 2.4 GHz เช่นกัน แล้วเราก็ทดสอบใช้งาน XBee ผลปรากฏว่า การรับส่งสัญญาณก็ขาดหายไปบ้างแต่บางครั้งก็ครบถ้วนนั้นอาจเป็นเพราะกำลังส่งของ XBee และระยะทางของ node ที่เราทดสอบใกล้กันมาก

ช่องสัญญาณย่านความถี่ 2.4 GHz นี้เรียกว่าเป็นย่านไมโครเวฟ หลักสำคัญของย่านไมโครเวฟอย่างหนึ่งคือ การวางตำแหน่งตัวรับส่งสัญญาณนั้น ต้องตั้งแบบ line of sight (ไม่มีสิ่งกีดขวางใดๆ) ถึงจะได้กำลังส่งสูงสุด สำหรับกำลังส่งของ XBee ในรุ่น Pro จะใช้ 50-60 mW ในเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้เชิงพาณิชย์ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

datasheet บอกว่าได้ไกลถึง 1.5 km. แต่ก็ต้องเป็นลักษณะของ line of sight หากไม่ใช่เงื่อนไขนี้ เราจะได้ระยะการรับส่งสัญญาณที่ลดลง นอกเหนือจากเรื่อง line of sight แล้ว ยังมีเรื่องสัญญาณรบกวนต่างๆ (Interference)

สายอากาศตัว XBee มีให้เลือกที่เป็นแบบสำเร็จรูปพร้อมใช้ ไม่ต้องหาสายอากาศมาต่อเพิ่มคือ สายอากาศแบบ chip ant และ whip ant ซึ่ง Pattern การแพร่กระจายคลื่นจะบ่งบอกเราอย่างหนึ่ง(จากหลาย ๆ อย่าง) ได้ว่าสายอากาศนี้ จะมีอัตราการขยายได้อย่างไร

Chip ant นั้น ก็มีข้อดีตรงที่มันทำให้ขนาด (Dimension) รวมมันเล็กกลง แต่อัตราการขยายน้อยกว่าแบบ Whip ant , Chip ant จึงมีระยะรับส่งข้อมูลที่ลดลงจาก spec ยกตัวอย่างเช่น รุ่น Pro ที่บอกว่าสามารถส่งได้ไกลสูงสุด 1.5 km แบบ line of sight แต่ถ้าเราเลือก Chip ant แล้ว จะได้ระยะสูงสุดอยู่ที่ 500 กว่าเมตร



รูปที่ 2.14 รูปสายอากาศแบบ Wire (Whip) , Chip ตามลำดับ [4]

ตารางที่ 2.1 ตารางแสดงคุณลักษณะของมาตรฐาน XBee [4]

Module	Antenna	Outdoor Distance (Visual Line of sight)	Indoor Distance (Office Building)	Indoor Distance (Warehouse)
	Type			
XBee	Chip	470 ft. (143 m)	80 ft. (24 m)	-
	Whip	845 ft. (258 m)	80 ft. (24 m)	84 ft. (26 m)
XBee-PRO	Chip	1690 ft. (515 m)	140 ft. (43 m)	-
	Whip	4382 ft. (1335 m)	140 ft. (43 m)	355 ft. (106 m)

2.3.4 XBee Association

ในเครือข่าย ZigBee ต้องมีการทำงานในโหมดประหยัดพลังงาน ในช่วงเวลาที่ไม่มีการทำงาน รับส่งข้อมูล ดังนั้นตัว XBee จึงมี Parameter ที่จะกำหนดการทำงานสำหรับ Sleep mode (Parameter A1,A2,SP,ST)

2.3.5 XBee Addressing

ตัว XBee จะสามารถกำหนดค่าประจำตัวอ้างอิงของมัน (Address) 2 แบบ คือ แบบ 16 bit

address และ 64 bit address ปกติแล้ว XBee ทุกตัวจะถูกกำหนดค่ามาจากโรงงานเป็น Address 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

bit อยู่แล้ว ซึ่งจะสามารถอ่านค่าได้จาก parameter SH+SL การใช้งาน Address 64 bit สามารถทำได้โดยกำหนด parameter MY ให้มีค่า 0xFFFF หรือ 0xFFFE ส่วนการกำหนด 16 bit address นั้นทำได้โดยกำหนด parameter MY ให้มีค่าน้อยกว่า 0xFFFE โดยจะเรียกเป็น mode การทำงาน 2 ประเภทคือ

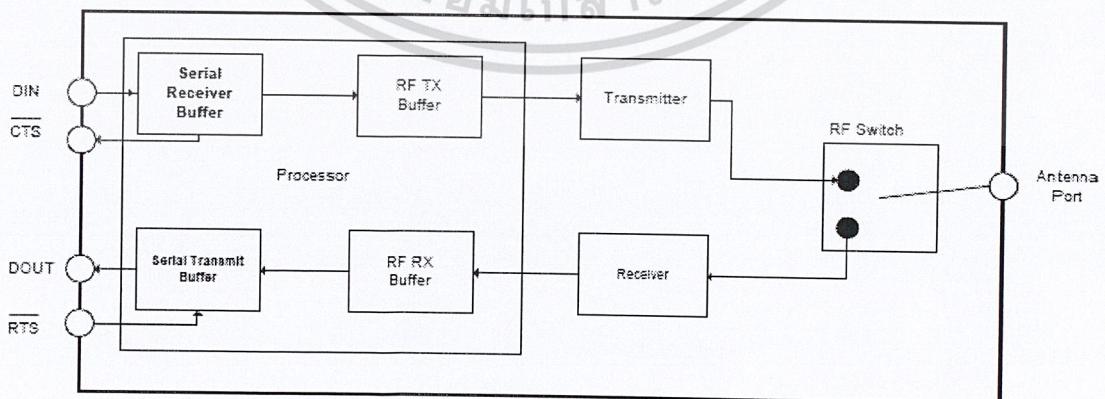
- **Unicast Mode** คือ การรับส่งข้อมูลโดยอาศัยหลักการ Acknowledgement คือหากทางด้านส่งนั้นส่งข้อมูลไป แต่ไม่ได้รับ Ack ตอบกลับจากตัวรับ ก็จะทำให้การส่งข้อมูลใหม่
- **Broadcast Mode** คือการส่งข้อมูลไปยังปลายทางให้ได้รับข้อมูลทุกตัว

2.3.6 XBee Operation Mode

XBee จะสามารถแบ่งช่วงการทำงานได้เป็น 5 แบบ คือ

- **Idle Mode** โหมดนี้จะเป็นโหมดที่ไม่ได้รับส่งข้อมูล ตัว XBee เตรียมที่จะทำงานในโหมดอื่น ๆ ต่อไปทันที หากมีเงื่อนไขบางอย่าง
- **Transmit / Receive Mode** (พุทรวม 2 Mode) คือช่วงที่ XBee มีการรับ หรือ ส่งข้อมูล โดยจะแบ่งลักษณะการทำงานย่อยออกเป็น Direct กับแบบ Indirect , การกำหนด Address ต้นทางและปลายทาง , Clear Channel Assessment และการตอบรับ Acknowledgement
- **Sleep Mode** คือ ช่วงที่ XBee อยู่ในสถานการณ์ทำงานพลังงานต่ำที่สุดเมื่อไม่มีการใช้งาน
- **Command Mode** คือ เป็นส่วนการปรับ parameter ของ XBee ซึ่งจะมีการกำหนด 2 แบบคือ แบบ AT command กับแบบ API Command

2.3.7 Data Throughput ของ XBee



รูปที่ 2.15 Internal Data Flow Diagram [4]

โดยทั่วไปการใช้งาน RF Module ควรจะกำหนดให้มี Buffer ด้วย เพื่อการปรับอัตรา รับส่งข้อมูลระหว่างตอนที่รับส่งทางอากาศ กับตอนที่รับส่งไปยังไมโครคอนโทรลเลอร์ หรือ อุปกรณ์อื่น ๆ ได้อย่างเหมาะสม

Data ที่รับส่งระหว่าง MCU กับ XBee จะมีข้อจำกัดเรื่อง Packet อาจถูก Drop ได้ เนื่องจาก Data Overflow โดยสำหรับด้านการส่งข้อมูลไปที่ XBee เพื่อออกอากาศนั้น ที่ขา DI จะมี Buffer อยู่ประมาณ 202 Bytes หากส่งเกิน Buffer จะเกิดการ Drop packet ที่ขา DI ซึ่งทางฝั่งรับ ข้อมูล ที่ขา DO ก็มี Buffer อยู่เช่นกัน โดยจะมี Parameter ที่เกี่ยวข้องกับ Data Throughput คือ RO และ BD

ค่า RO คือค่า Packetization Timeout ซึ่งเป็น delay ของข้อมูลที่อยู่ใน DI Buffer ก่อนที่จะถูก encapsulate ไปที่ส่วน RF transmission เพื่อส่งข้อมูลออกอากาศ หากตั้ง RO = 0 Data ที่รับเข้ามาจาก MCU จะถูก XBee Encapsulate Packet ส่งออกอากาศทันที ดังนั้นเราจะมี Parameter RO และ BD ที่จะช่วยในการปรับ Data รับส่งให้สามารถรับส่งกันได้ทัน ไม่ให้มีการ Drop Packet ได้ ในกรณีที่ส่งข้อมูลเกิน 200 Bytes

นอกจากนี้ยังมี PIN CTS(ขา12) และ RTS(ขา16) ช่วยเตือนเราเวลาที่ Buffer ภายในใกล้จะเต็ม โดยในฝั่งส่ง DI Buffer จะส่ง Signal มาทาง CTS เมื่อ DI Buffer เหลือพื้นที่จัดเก็บอยู่อีก 17 Bytes และส่ง Clear Signal ที่ CTS เมื่อ DI Buffer เหลือพื้นที่จัดเก็บมากกว่า 34 Bytes

ดังนั้น สำหรับการเขียน โปรแกรม รับส่งข้อมูลกับ XBee ต้องคำนึงเรื่อง Buffer ด้วย แต่ในทางปฏิบัติสำหรับงาน Sensor Network ก็ไม่ได้รับส่งข้อมูล Stream Data ยาว ๆ เท่าไหร่

2.4 Localization

เทคโนโลยีด้านเครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Network) มีการนำมาใช้ในการ วัดและการตรวจสอบการเปลี่ยนแปลงต่างๆ มีการนำเทคโนโลยีทางด้านนี้มาประยุกต์ใช้ในการ หาตำแหน่ง (Localization) ระยะทาง (Distances) โดยเทคโนโลยีนี้เหมาะสมสำหรับการใช้ฟังก์ชันในการหาตำแหน่งทั้งภายในและภายนอกอาคาร จากการประยุกต์ใช้งานกันอย่างมากมายสำหรับการหาตำแหน่งและการติดตามเป้าหมาย ระบบและอัลกอริทึมในการหาตำแหน่งของเป้าหมายได้รับการวิจัยและค้นคว้าอย่างกว้างขวาง โดยความถูกต้องในการหาตำแหน่งและความน่าเชื่อถือของระบบการหาตำแหน่งถือเป็นปัจจัยหลักที่ต้องการ

2.4.1 Received Signal Strength Indication (RSSI)

การประยุกต์ใช้งาน โครงข่ายเซ็นเซอร์แบบไร้สายในการระบุตำแหน่งนั้นมักนิยมใช้การ วัดระดับความเข้มของสัญญาณที่ได้รับ (Receive Signal Strength Indicator : RSSI) ในการค้นหาเป้าหมาย โดยกำหนดตำแหน่งที่แน่นอนของกลุ่มโหนดอ้างอิง ซึ่งมักจะถูกเรียกว่า โหนดบีมอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิใช่สัญญาเห็นชอบหรือการรับประกันค่า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(beacon node) และใช้งานได้โดยการวัดความแรงของระดับสัญญาณ ระหว่างโหนดบีคอนกับเป้าหมาย โดยใช้พื้นฐานของ 2 เทคนิคที่ได้กล่าวข้างต้น

Received Signal Strength Indication หรือ RSSI เป็นวิธีการวัดความแรงของสัญญาณในการส่งสัญญาณวิทยุ ตามความเป็นจริงของสัญญาณ โดยทั่วไป เมื่อระยะทางไกลออกไปสัญญาณก็จะเบาลง ดังนั้นค่า RSSI นั้นก็จำเป็นต่อการส่งสัญญาณเช่นกัน เพราะจะรู้ว่าเครื่องรับสัญญาณที่ใช้งานอยู่นั้นจะมีประสิทธิภาพที่ระยะทางเท่าใด หรือกำลังในการส่งเท่าใดจึงจะเหมาะสม โดยค่า RSSI มีหน่วยเป็น เดซิเบลมิลลิวัตต์ (dBm)

สำหรับการวัดความแรงของสัญญาณ ที่แปลงค่าความแรงมาเป็นแรงดันไฟฟ้า นั้น และมักไม่ใช้ในงานที่ต้องการความแม่นยำสูง เนื่องจากมีความแปรปรวนของสัญญาณตามสภาพแวดล้อมสูง เพราะแต่ละสภาพแวดล้อมมีความแตกต่างกัน การจะทำให้ RSSI มีความน่าเชื่อถือคือจะต้องเก็บข้อมูลไว้หลาย ๆ ครั้ง เพื่อนำมาวิเคราะห์ภายหลัง จะทำให้เกิดความผิดพลาดน้อยลง ความผิดพลาดที่เกิดขึ้นนั้นอาจจะมีสาเหตุมาจากต่าง ๆ ดังนี้

- **Nonline of Sight (NLOS)** คือ การสื่อสารแบบไม่เป็นเส้นตรง ทำให้การวัดมุมในระยะทางที่ไกลจะก่อให้เกิดความผิดพลาดขึ้นได้
- **Multiple Access Interference** คือ เป็นปัญหาที่มักเกิดในระบบ CDMA เกิดจากเครื่องส่งที่มีกำลังส่งสูงกว่าส่งสัญญาณไปรบกวนเครื่องส่งที่มีกำลังส่งต่ำกว่า
- **Fluctuation in Signal Propagation Speeds** เกิดกับคลื่นเสียงที่การแพร่ถูกรบกวนจากปัจจัยภายนอก เช่น แรงลมหรืออุณหภูมิ และค่าความชื้นในอากาศ ปัจจัยเหล่านี้จะทำให้เกิดความคลาดเคลื่อนสูงเมื่อระยะทางมากขึ้น

2.4.1.1 การวัดค่าหาความแรงของสัญญาณ

การวัดค่าหาความแรงของสัญญาณเป็นวิธีการวัดค่าความแรงของสัญญาณ วิทยุ ระหว่างเซ็นเซอร์ไร้สายตัวหนึ่งกับเซ็นเซอร์ไร้สายอีกตัวหนึ่ง มีความสัมพันธ์กันตามสมการ 2.1

$$RSSI = -10n \log_{10}(d) + A \quad (2.1)$$

เมื่อ RSSI คือค่าความแรงของสัญญาณวิทยุ มีหน่วยเป็น เดซิเบลมิลลิวัตต์ (dBm)

n คือค่าคงที่การกระจายสัญญาณของสถานที่นั้นๆ

d คือระยะห่างระหว่างเซ็นเซอร์ไร้สาย (เมตร)

A คือค่าความแรงของสัญญาณที่ได้รับใน 1 เมตร (dBm)

จากสมการข้างต้น ถ้าค่าระยะทาง (d) เพิ่มมากขึ้นก็จะส่งผลให้ค่า RSSI ลดลง ซึ่งบอกค่าความแรงของสัญญาณ สามารถนำผลนี้ไปใช้เพื่อหาระยะทางและปริมาณตำแหน่งวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 Link Quality Indicator (LQI)

การใช้ RSSI นั้น ยังไม่สามารถทำงานได้อย่างเต็มประสิทธิภาพ เนื่องจากปัจจัยทางสภาพแวดล้อมหลายๆอย่างซึ่งสาเหตุเหล่านี้ ทำให้ประสิทธิภาพในการใช้งานจริงถูกลดทอนไปมาก จึงได้มีการนำเสนอวิธีการอื่นที่เข้ามาช่วยในการวัดระยะทาง นั่นคือ ตัวชี้วัดคุณภาพของการเชื่อมโยง (link quality indicator : LQI) มาช่วยในการหาระยะทางตามมาตรฐาน IEEE 802.15.4 ได้ ระบุว่า LQI เป็นตัวบ่งชี้คุณสมบัติ หรือคุณภาพของชุดข้อมูลที่ได้รับและเป็นสัดส่วนกับ RSSI อีกทั้งยังเป็นพารามิเตอร์ที่มีความน่าเชื่อถือมากกว่า RSSI

ในการระบุตำแหน่งนั้นสามารถแบ่งได้เป็น 2 เทคนิคหลัก คือ เทคนิคที่ใช้พื้นฐานของการคาดคะเน (Proximity-based) หรือบางครั้งเรียกว่าเทคนิคที่ใช้พื้นฐานของ Fingerprint และเทคนิคที่ใช้พื้นฐานของการหาระยะทาง (Range-based)

2.4.3 เทคนิคที่ใช้พื้นฐานของ Fingerprint (Location Fingerprint Technique)

เทคนิคที่ใช้พื้นฐานของ Fingerprint (Location Fingerprint Technique) เป็นการหาตำแหน่ง จากการเปรียบเทียบพารามิเตอร์ของสัญญาณของเป้าหมายเทียบกับพารามิเตอร์ของโหนดใกล้เคียง ตำแหน่งเป้าหมายสามารถประมาณค่าจากโหนดอ้างอิงที่มีค่าพารามิเตอร์ใกล้เคียงกับโหนดเป้าหมายมากที่สุด เรียกได้อีกแบบว่า Proximity-based จะมีความแม่นยำมาก แต่มีกระบวนการในการวิเคราะห์และประมวลผลที่ซับซ้อนกว่าและใช้เวลามากกว่า

เป็นการระบุตำแหน่งแบบการคาดคะเนจากค่า RSSI ที่ระบบได้ทำการเรียนรู้ไว้ เรียกว่า Fingerprint โดยจะทำการเก็บค่า ทุก ๆ 1 เมตร ตลอดพื้นที่การทดลอง ดังสมการ (2.2)

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_m \end{bmatrix} = \begin{bmatrix} \text{RSSI}_{11} & \cdots & \text{RSSI}_{1n} \\ \vdots & \ddots & \vdots \\ \text{RSSI}_{m1} & \cdots & \text{RSSI}_{mn} \end{bmatrix} \quad (2.2)$$

โดยเมื่อ m คือ ตำแหน่งของ Fingerprint และ n คือ จำนวนโหนด และอยู่ในตำแหน่งดังสมการ (2.3)

$$FL(F) = \begin{bmatrix} FL(F_1) \\ \vdots \\ FL(F_m) \end{bmatrix} = \begin{bmatrix} x_1, y_1 \\ \vdots \\ x_m, y_m \end{bmatrix} \quad (2.3)$$

เมื่อทำการระบุตำแหน่งจะทำการดึงค่า RSSI ระหว่างโหนดอ้างอิงต่างๆกับโหนดเป้าหมายค่าออกมาดังสมการ (2.4)

$$OB = [\text{RSSI}_1 \quad \cdots \quad \text{RSSI}_n] \quad (2.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นจะนำค่า RSSI ระหว่างโหนดอ้างอิงต่างๆ กับ โหนดเป้าหมาย มาเปรียบเทียบกับค่า ที่ได้เก็บไว้ เพื่อหาตำแหน่ง โดยเมื่อ E ค่าความคลาดเคลื่อนมีค่าดังสมการ (2.5)

$$E = \begin{bmatrix} E_1 \\ \vdots \\ E_m \end{bmatrix} \quad (2.5)$$

$$= \begin{bmatrix} \sqrt{(RSSI_{11} - RSSI_1)^2 + (RSSI_{12} - RSSI_2)^2 + (RSSI_{13} - RSSI_3)^2 + (RSSI_{14} - RSSI_4)^2} \\ \vdots \\ \sqrt{(RSSI_{m1} - RSSI_1)^2 + (RSSI_{m2} - RSSI_2)^2 + (RSSI_{m3} - RSSI_3)^2 + (RSSI_{m4} - RSSI_4)^2} \end{bmatrix}$$

โดยในที่นี้จะใช้วิธีการหาตำแหน่งของเป้าหมาย 2 วิธีด้วยกัน วิธีการแรกคือ Nearest เป็นการหาตำแหน่งจากค่าความผิดพลาดน้อยที่สุด ระหว่างตำแหน่งที่ได้เรียนรู้กับ โหนดเป้าหมาย ซึ่งตำแหน่งที่ได้จะเป็นตำแหน่งของ FL ให้ค่าความผิดพลาดน้อยที่สุดมีค่าดังสมการ (2.6)

$$P(OB) = FL(\min(E_i)) \quad (2.6)$$

เมื่อ $i = 1, \dots, m$

อีกวิธีหนึ่งที่ใช้คือวิธี 3-Nearest วิธีนี้จะเหมือนกับ Nearest แต่จะใช้ค่า ความผิดพลาดที่น้อยที่สุด 3 ค่ามาหาตำแหน่งเฉลี่ย ดังสมการ (2.7)

$$P(OB) = (P(\min_1(FL)) + P(\min_2(FL)) + P(\min_3(FL)))/3 \quad (2.7)$$

2.4.4 เทคนิคที่ใช้พื้นฐานของการหาระยะทาง (Range-based)

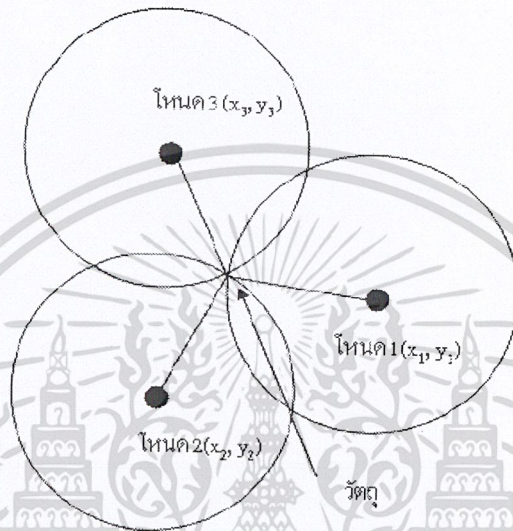
เทคนิคที่ใช้พื้นฐานของการหาระยะทาง (range-based) จะเป็นการหาตำแหน่งของเป้าหมาย โดยใช้การคำนวณระยะทางระหว่างโหนดอ้างอิงกับเป้าหมายซึ่งมีความแม่นยำน้อยกว่าเทคนิคข้างต้น สามารถหาได้หลายวิธี ขึ้นอยู่กับพารามิเตอร์ที่ใช้ในการค้นหา

2.4.4.1 Time of Arrival (TOA)

การหาตำแหน่งโดยอ้างอิงจากเวลาของการมาถึง (TOA) ในกรณีที่เป้าหมายเป็นตำแหน่ง 2 มิติ จะต้องพิจารณาสัญญาณอย่างน้อย 3 จุด ตามรูปที่ 2.16 และจับเวลาในการเคลื่อนที่ของสัญญาณ

ทางเดียว (One Way) เพื่อคำนวณระยะทางระหว่างหน่วยวัดกับเครื่องส่งสัญญาณ (Signal Transmitter) โดยทั่วไปวิธี TOA จะพบปัญหา 2 ข้อคือ

- เครื่องส่งและรับสัญญาณในระบบจะต้องทำให้พร้อมกัน เป็นจังหวะเดียวกัน (Synchronize)
- ไทม์สแตมป์ (Timestamp) จะถูกติดไว้ที่สัญญาณที่ถูกส่งออกมา เพื่อหาระยะทางที่สัญญาณเดินทาง



รูปที่ 2.16 แสดงการหาตำแหน่งด้วยการวัดแบบ TOA

สำหรับวิธีการคำนวณตำแหน่งแบบวิธีการอ้างอิงจากเวลาของการมาถึงคือ การหาระยะห่างระหว่างวัตถุและโหนด หลักการคือเราสามารถหาระยะห่าง R_i ซึ่งเป็นระยะทางระหว่างวัตถุกับโหนดที่ i ได้จากสมการ (2.8)

$$R_i = ct_i \quad (2.8)$$

เมื่อ c คือค่าความเร็วแสงมีค่าเท่ากับ 3×10^8 เมตรต่อวินาที

และ t_i คือระยะเวลาในการเดินทางสัญญาณ (TOA) ระหว่างวัตถุไปที่โหนด i

แต่วิธีการนี้ต้องมีวงจรในการสร้างความสอดคล้องของสัญญาณนาฬิกา ระหว่างวัตถุกับโหนด จึงจะทำให้สามารถวัดค่า t_i ได้ พิจารณาจากสมการ (2.9) สมการมาตรฐานของวงกลมที่มีรัศมี R

$$R_i = \sqrt{(X_i - x)^2 + (Y_i - y)^2 + (Z_i - z)^2} \quad \text{โดย } i = 1, 2, 3, \dots, n \quad (2.9)$$

เมื่อ (X_i, Y_i, Z_i) เป็นพิกัดของ โหนด i

และ (x, y, z) เป็นพิกัดของตำแหน่งของวัตถุที่ต้องการทราบตำแหน่ง

จากที่ได้กล่าวไปแล้วข้างต้นว่าการหาตำแหน่งของวัตถุด้วยวิธีนี้มีข้อเสียของวิธีการค้นหาตำแหน่งด้วยวิธีนี้ได้แก่การต้องการความสอดคล้องกันของสัญญาณนาฬิการะหว่างวัตถุและโหนดจะพบว่าเมื่อนับระยะเวลาของสัญญาณจากวัตถุไปที่โหนดผิดพลาด 1 ไมโครวินาทีจะให้วัฏระยะทาง ผิดพลาดถึง 300 เมตร

2.4.4.2 Time Difference of Arrival (TDOA)

การหาตำแหน่งโดยอ้างอิงจากความแตกต่างของเวลาของสัญญาณที่มาถึง (TDOA) ได้แนวคิดมาใช้เพื่อแก้ปัญหาด้านความถูกต้องของความสอดคล้องของสัญญาณนาฬิการะหว่างวัตถุกับโหนด (Synchronize) โดยเนื่องจากสัญญาณเคลื่อนที่ด้วยความเร็วแสงทำให้พบว่าในกรณีที่วัดเวลาผิดพลาดเพียง 1 ไมโครวินาทีทำให้เกิดข้อผิดพลาดในการวัดระยะทางถึง 300 เมตร ดังนั้นเพื่อแก้ปัญหาเรื่องของการสอดคล้องของสัญญาณนาฬิกาทำให้เกิดแนวคิดแทนที่จะทำการนับเวลาที่มาถึงโหนดเราจะทำการนับเวลาที่แตกต่างกันที่รับได้จากโหนดจำนวน 2 คู่เพื่อนำข้อมูลมาสร้างสมการไฮเพอร์โบลิก เพื่อหาตำแหน่งของวัตถุโดยพิจารณาการหาค่าตำแหน่งของโหนดได้โดยใช้หลักการของการนับเวลาเวลาที่แตกต่างกันที่นับได้จากคู่ของสัญญาณจากวัตถุไปที่โหนดตำแหน่งที่ i และตำแหน่งที่ j เพื่อสร้างของสมการไฮเพอร์โบลิกพิจารณาความสัมพันธ์ของผลต่างของระยะทางระหว่างโหนด ตำแหน่งที่ i และตำแหน่งที่ j ดังสมการ (2.10)

$$R_{i,j} = ct_{i,j} = R_i - R_j \quad (2.10)$$

โดยที่ $R_{i,j}$ เป็นผลต่างของระยะทางระหว่างโหนดตำแหน่งที่ i และตำแหน่งที่ j และ $t_{i,j}$ เป็นความแตกต่างของเวลาของสัญญาณจากวัตถุไปที่ โหนดตำแหน่งที่ i และตำแหน่งที่ j และ c คือค่าความเร็วแสงมีค่าเท่ากับ 3×10^8 เมตรต่อวินาทีและ R_i คือระยะห่างระหว่างโหนดที่ตำแหน่งที่พิกัด (x, y) กับวัตถุซึ่งมีพิกัดเป็น (X_i, Y_i) ดังสมการ (2.11)นี้

$$R_i = \sqrt{(X_i - x)^2 + (Y_i - y)^2} = \sqrt{X_i^2 + Y_i^2 - 2X_i x - 2Y_i y + x^2 + y^2} \quad (2.11)$$

และสามารถหาความสัมพันธ์ของผลต่างของระยะทางได้จากวัตถุไปที่โหนด i, j โดยมีความสัมพันธ์ตามสมการ (2.12)

$$R_{i,j} = \sqrt{(X_i - x)^2 + (Y_i - y)^2} - \sqrt{(X_j - x)^2 + (Y_j - y)^2} \quad (2.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญตให้นำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ $R_{i,j}$ เป็นผลต่างของระยะทางระหว่างวัตถุตำแหน่งที่ i และตำแหน่งที่ j และ (X_i, Y_i) และ (X_j, Y_j) เป็นพิกัดของ โหนด ที่ i และ j และ (x, y) เป็นตำแหน่งของวัตถุที่ต้องการทราบตำแหน่ง เราสามารถหาค่าตอบของสมการได้เมื่อจำนวนสมการเท่ากับจำนวนตัวแปรที่ไม่ทราบค่าเนื่องจากสมการที่ 2.12 ทำการแก้สมการค่อนข้างยากทำให้ต้องจัดรูปแบบของสมการใหม่ดังสมการ (2.13)

$$R_i^2 = (R_{i,j} + R_1)^2 \quad (2.13)$$

ทำให้ทำการจัดรูปของสมการได้ดังสมการ (2.14)

$$R_{i,1}^2 + 2R_{i,1}R_1 + R_1^2 = X_i^2 + Y_i^2 - 2X_i x - 2Y_i y + x^2 + y^2 \quad (2.14)$$

ทำการลบค่าของสมการ R_i^2 ที่ $i = 1$ ออกจากสมการข้างต้น จะได้ดังสมการ (2.15)

$$R_{i,1}^2 + 2R_{i,1}R_1 = X_i^2 + Y_i^2 - 2X_{i,1}x - 2Y_{i,1}y \quad (2.15)$$

เมื่อ $X_{i,1}$ และ $Y_{i,1}$ มีค่าเท่ากับ $(X_i - X_1)$ และ $(Y_i - Y_1)$ ตามลำดับ ซึ่งสามารถหาค่าตอบของสมการได้ดังสมการ (2.16)

$$\begin{bmatrix} x \\ y \end{bmatrix} = - \begin{bmatrix} X_{2,1} & Y_{2,1} \\ X_{3,1} & Y_{3,1} \end{bmatrix}^{-1} \times \left\{ \begin{bmatrix} R_{2,1} \\ R_{3,1} \end{bmatrix} R_1 + \frac{1}{2} \begin{bmatrix} R_{2,1}^2 - K_2 + K_1 \\ R_{3,1}^2 - K_3 + K_1 \end{bmatrix} \right\} \quad (2.16)$$

เมื่อค่า K มีค่าดังสมการ (2.17)

$$\begin{aligned} K_1 &= X_1^2 + Y_1^2 \\ K_2 &= X_2^2 + Y_2^2 \\ K_3 &= X_3^2 + Y_3^2 \end{aligned} \quad (2.17)$$

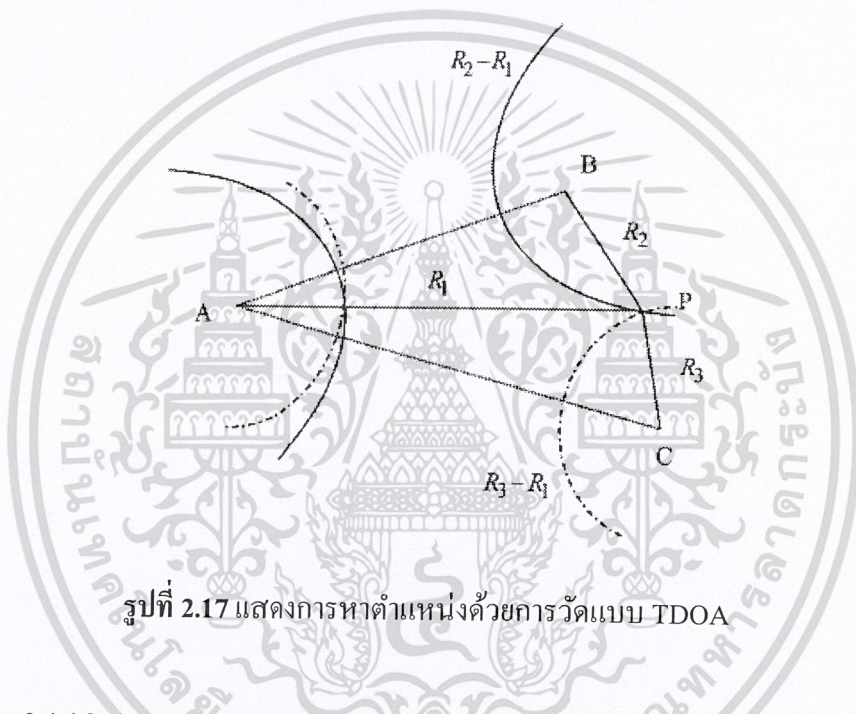
เราสามารถทำการหาค่า x และ y ได้ในเทอมของ R_1 ได้โดยการแทนค่าของ x และ y ในสมการ (2.17) ไปแทนค่าในสมการ (2.16) และจัดรูปแบบสมการในเทอมของ R_1 ในรูปแบบของสมการ Quadratic ดังสมการ (2.18)

$$aR_1^2 + bR_1 + c = 0 \quad (2.18)$$

สามารถหาคำตอบของสมการ (2.18) ตามสูตรได้เป็นสมการ (2.19)

$$R_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (2.19)$$

เลือกใช้รากของสมการเพียงค่าเดียวเนื่องจาก R_1 เป็นระยะห่างระหว่างวัตถุและโหนดที่ 1 ทำให้รากของสมการมีค่าเป็นบวกเพียงค่าเดียวแล้วนำค่าที่ได้ไปแทนค่าในสมการ (2.16)



รูปที่ 2.17 แสดงการหาตำแหน่งด้วยการวัดแบบ TDOA

2.4.4.3 Angle of Arrival (AOA)

วิธีการหาตำแหน่งด้วยมุม (Angle of Arrival :AOA) เป็นเทคนิคการหาตำแหน่ง โดยใช้ค่ามุมของสัญญาณที่ส่งมาจากวัตถุ โดยเราสามารถค้นหาตำแหน่งของวัตถุจากค่ามุมที่ได้รับที่โหนดจำนวน 2 มุมในแต่ละโหนดและนำข้อมูลที่ได้รับมาสร้างสมการเส้นตรงหาตำแหน่งของวัตถุได้ตั้งสมการ (2.20) และ สมการ (2.21)

$$(y - y_i) = \tan(\theta_i)(x - x_i) \quad (2.20)$$

$$(y - y_j) = \tan(\theta_j)(x - x_j) \quad (2.21)$$

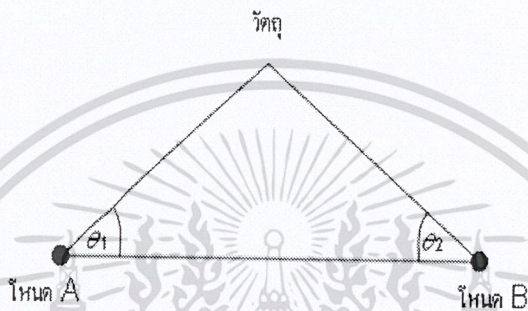
โดยที่

$$\theta_i = \tan^{-1} \left(\frac{|y_i - y|}{|x_i - x|} \right) \quad (2.22)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ
$$\theta_i = \tan^{-1} \left(\frac{|y_j - y|}{|x_j - x|} \right) \quad (2.23)$$

และ (x, y) เป็น พิกัดที่ต้องการทราบของวัตถุที่ต้องการทราบตำแหน่ง
 (x_i, y_i) เป็น พิกัดที่ต้องการทราบของโหนด ตามแนวแกน x และ y จุดที่ i
 (x_j, y_j) เป็น พิกัดที่ต้องการทราบของโหนด ตามแนวแกน x และ y จุดที่ j



รูปที่ 2.18 แสดงการหาตำแหน่งด้วยการวัดแบบ AOA

จากรูปที่ 2.19 แสดงวิธีการวัดมุมจากทิศทางของสัญญาณคือต้องการจำนวนโหนดจำนวนเพียง 2 โหนดเพื่อการคำนวณตำแหน่งซึ่งน้อยกว่าวิธีการคำนวณตำแหน่งด้วยวิธีอ้างอิงจากเวลาของการมาถึง โดยวิธีอ้างอิงจากความแตกต่างของเวลาของสัญญาณที่มาถึง จะต้องมีการวัดมุมอย่างน้อยจำนวน 3 โหนดขึ้นไปแต่ข้อเสียของการค้นหาตำแหน่งด้วยวิธีนี้คือการเกิดเส้นทางของสัญญาณหรือการขาดเส้นทางในแนวเส้นสายตา (Line of Sight: LOS) ที่ส่งผลกระทบต่อวิธีการวัดมุมที่ผิดพลาดมากยิ่งขึ้นและการเพิ่มค่าใช้จ่ายในการปรับแต่งเสาอากาศเพื่อให้วัดค่าได้อย่างถูกต้องอย่างสม่ำเสมอ

2.4.5 Maximum Likelihood Algorithm

เป็นหนึ่งในวิธีการคำนวณหาตำแหน่ง บนพื้นฐานของการคำนวณระยะทาง ซึ่งสามารถหาได้โดยสมการ (2.1)

$$RSSI = -10n \log_{10}(d) + A$$

เมื่อ RSSI คือค่าความแรงของสัญญาณวิทยุ มีหน่วยเป็น เดซิเบลมิลลิวัตต์ (dBm)

n คือค่าคงที่การกระจายสัญญาณของสถานที่นั้นๆ

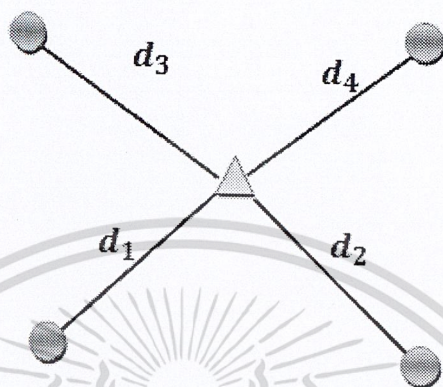
d คือระยะห่างระหว่างเซ็นเซอร์ไร้สาย (เมตร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A คือค่าความแรงของสัญญาณที่ได้รับใน 1 เมตร (dBm)

จากสมการ (2.1) ถ้าค่าระยะทาง (d) เพิ่มมากขึ้นก็จะส่งผลให้ค่า RSSI ลดลง ซึ่งบอกค่าความแรงของสัญญาณ สามารถนำผลนี้ไปใช้เพื่อหาระยะทางและปริมาณตำแหน่งวัตถุ แสดงการหาระยะทาง(d) ระหว่างโหนดอ้างอิงดังรูปที่ 2.19



รูปที่ 2.19 แสดงการหาระยะทาง(d) ระหว่างโหนดอ้างอิง

เมื่อกำหนดหาค่า d ได้แล้ว จะนำเข้าไปใส่สมการของ Maximum Likelihood จะทำการแปลงให้อยู่ในรูปสมการของพีทาคาร์ที่เขียนดังสมการ (2.24)

$$\begin{aligned} (x_1 - x)^2 + (y_1 - y)^2 &= d_1^2 \\ (x_2 - x)^2 + (y_2 - y)^2 &= d_2^2 \\ &\vdots \\ (x_n - x)^2 + (y_n - y)^2 &= d_n^2 \end{aligned} \quad (2.24)$$

สามารถจัดรูปแบบได้ดังสมการ (2.25)

$$\begin{cases} x^2 - x_n^2 - 2(x_1 - x_n)x + y_1^2 \\ -y_n^2 - 2(y_1 - y_n)y = d_1^2 - d_n^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 - 2(x_{n-1} - x_n)x + y_{n-1}^2 \\ -y_n^2 - 2(y_{n-1} - y_n)y = d_{n-1}^2 - d_n^2 \end{cases} \quad (2.25)$$

โดยที่ x_n, y_n เป็น ตำแหน่งในแกน x และแกน y ของโหนดอ้างอิงใดๆ และ y เป็นตำแหน่งในแกน x และแกน y ของโหนดเป้าหมาย และทำการจัดรูปให้อยู่ในรูปสมการ (2.26)

$$Xb = a \quad (2.26)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ

$$\mathbf{X} = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix} \quad (2.27)$$

$$\mathbf{a} = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 - d_1^2 + d_n^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 - d_{n-1}^2 + d_n^2 \end{bmatrix} \quad (2.28)$$

และ

$$\mathbf{b} = \begin{bmatrix} x \\ y \end{bmatrix} \quad (2.29)$$

โดยสามารถหาค่า \mathbf{b} ได้จากสมการ (2.30)

$$\hat{\mathbf{b}} = \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{a} \quad (2.30)$$

2.4.6 Min-Max Algorithm

เป็นอีกหนึ่งในวิธีการคำนวณหาตำแหน่งบนพื้นฐานของการคำนวณระยะทางซึ่งสามารถหาได้โดยสมการ (2.1)

$$RSSI = -10n \log_{10}(d) + A$$

เมื่อ RSSI คือค่าความแรงของสัญญาณวิทยุ มีหน่วยเป็น เดซิเบลมิลลิวัตต์ (dBm)

n คือค่าคงที่การกระจายสัญญาณของสถานที่นั้นๆ

d คือระยะห่างระหว่างเซ็นเซอร์ไร้สาย (เมตร)

A คือค่าความแรงของสัญญาณที่ได้รับใน 1 เมตร (dBm)

จากสมการข้างต้น ถ้าค่าระยะทาง (d) เพิ่มมากขึ้นก็จะส่งผลให้ค่า RSSI ลดลง ซึ่งบอกค่าความแรงของสัญญาณ สามารถนำผลนี้ไปใช้เพื่อหาระยะทางและปริมาณตำแหน่งวัตถุ

เมื่อคำนวณหาค่า d ได้แล้ว จะนำค่าที่ได้ ไปหาค่ามากที่สุดและน้อยที่สุดของทุกๆ โหนด อ้างอิงดังสมการ (2.31),(2.32),(2.33),(2.34) ซึ่งแสดงการหาค่า Min-Max ระหว่างโหนดอ้างอิงดังรูปที่ 2.20

$$x_{max} = \begin{bmatrix} x_{1max} = x_1 + d_1 \\ x_{2max} = x_2 + d_2 \\ \vdots \\ x_{nmax} = x_n + d_n \end{bmatrix} \quad (2.31)$$

$$y_{max} = \begin{bmatrix} y_{1max} = y_1 + d_1 \\ y_{2max} = y_2 + d_2 \\ \vdots \\ y_{nmax} = y_n + d_n \end{bmatrix} \quad (2.32)$$

$$x_{min} = \begin{bmatrix} x_{1min} = x_1 - d_1 \\ x_{2min} = x_2 - d_2 \\ \vdots \\ x_{nmin} = x_n - d_n \end{bmatrix} \quad (2.33)$$

$$y_{min} = \begin{bmatrix} y_{1min} = y_1 - d_1 \\ y_{2min} = y_2 - d_2 \\ \vdots \\ y_{nmin} = y_n - d_n \end{bmatrix} \quad (2.34)$$



รูปที่ 2.20 แสดงการหาค่า Min-Max ระหว่างโหนดอ้างอิง

ต่อมาจะนำค่ามากที่สุดและน้อยที่สุดไปหาค่าที่น้อยที่สุดของค่ามากที่สุดของฟังก์ชันและหาค่าที่มากที่สุดของค่าที่น้อยที่สุดของฟังก์ชัน ดังสมการ (2.35),(2.36),(2.37),(2.38) ซึ่งแสดงดังรูปที่ 2.21

$$x_{min-max} = \min(x_{max}) \quad (2.35)$$

$$x_{max-min} = \max(x_{min}) \quad (2.36)$$

$$y_{min-max} = \min(y_{max}) \quad (2.37)$$

$$y_{max-min} = \max(y_{min}) \quad (2.38)$$

สามารถประมาณค่าฟังก์ชันได้ดังสมการ (2.39)

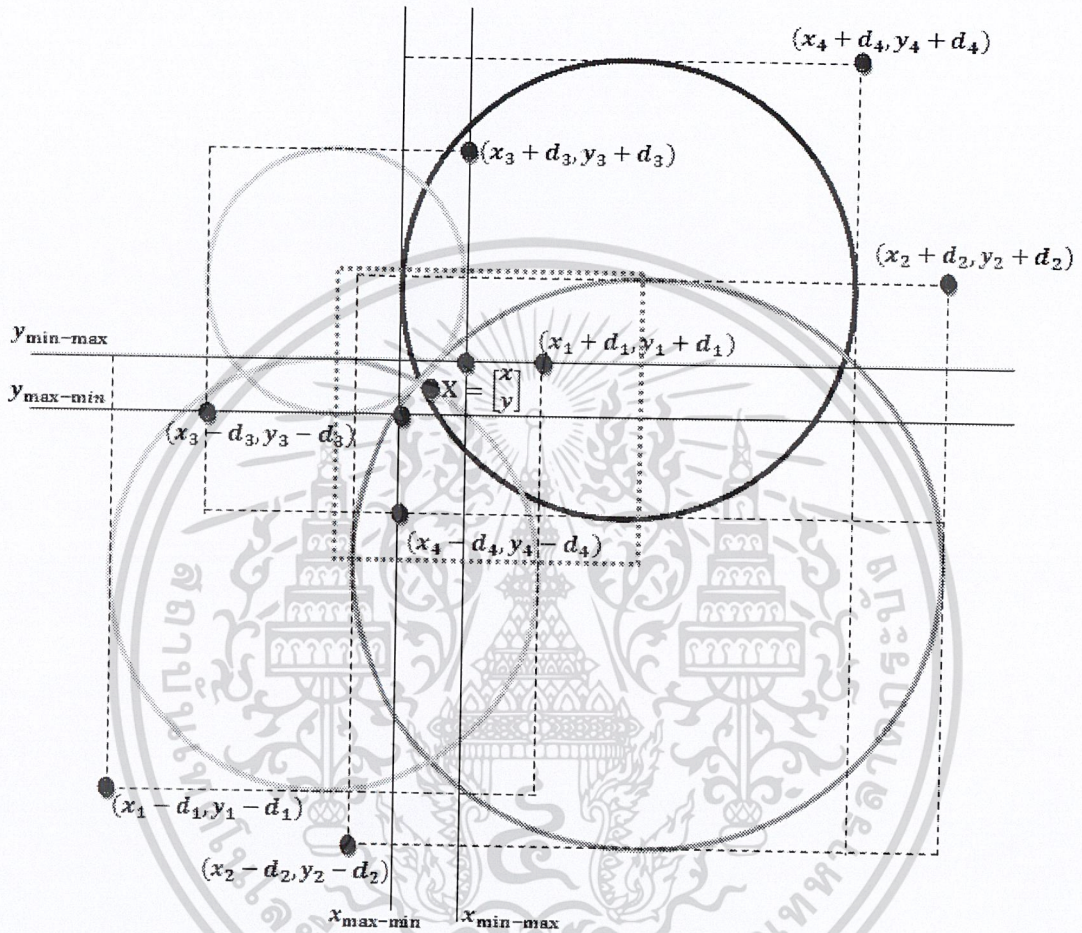
$$X = \begin{bmatrix} x \\ y \end{bmatrix} \quad (2.39)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่มีค่า x และ y ดังสมการ (2.40) และ (2.41)

$$x = (x_{\min-\max} + x_{\max-\min})/2 \quad (2.40)$$

$$y = (y_{\min-\max} + y_{\max-\min})/2 \quad (2.41)$$



รูปที่ 2.21 แสดงการหาตำแหน่งของวัตถุโดยใช้ Min-Max

2.5 บทสรุป

ในบทนี้ได้กล่าวถึงโครงสร้างการทำงานของโครงข่ายเซนเซอร์ไร้สาย รวมไปถึงการทำงานของ XBee และเทคนิคที่ใช้ในการระบุตำแหน่ง ซึ่งประกอบไปด้วยการระบุตำแหน่งแบบ Range-Based และแบบ Fingerprint โดยอยู่บนพื้นฐานของโครงข่ายเซนเซอร์ไร้สาย ซึ่งจะนำไปใช้ในการทดลองสำหรับการระบุตำแหน่งต่อไป

บทที่ 3

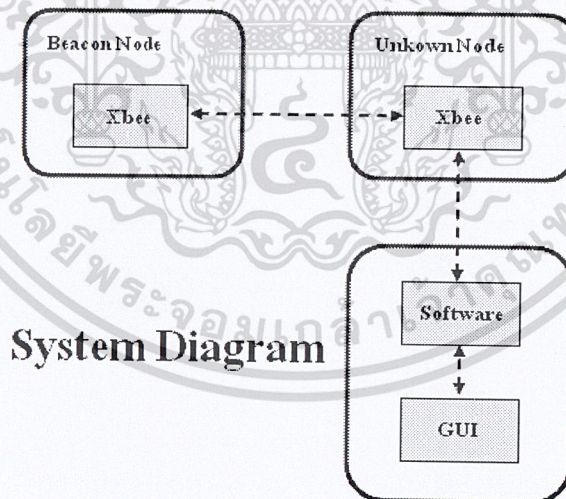
การวิเคราะห์และออกแบบระบบ

3.1 แนวคิด

จากการศึกษาเรื่อง การระบุพิกัดโดยการประมาณค่า (Proximity localization) ทางคณะผู้จัดทำมีความสนใจที่จะใช้ ระบบเครือข่ายไร้สาย ไร้สาย ในมาตรฐาน ZigBee มาใช้ในการทดลอง เพื่อทำการระบุตำแหน่ง โดยจะนำเอาพารามิเตอร์ที่อยู่ภายในเช่น RSSI มาทำการทดลอง จากนั้นนำไปคำนวณและประมวลผล เพื่อให้ทราบถึงประสิทธิภาพในการค้นหาและระบุตำแหน่ง

3.2 การสร้างและออกแบบระบบ

การออกแบบระบบนั้นต้องคำนึงถึงองค์ประกอบที่สำคัญของเครือข่ายเซ็นเซอร์ไร้สาย ได้แก่ การทำงานแบบเวลาจริง และส่วนติดต่อผู้ใช้ที่ง่ายต่อการใช้งาน โดยในการออกแบบระบบนี้ได้แยกการออกแบบเป็น 4 ส่วน คือ ในส่วนของ Beacon, Unknown, Gateway และ Software ซึ่งได้แสดงดังรูปที่ 3.1



รูปที่ 3.1 แผนภาพการทำงานของระบบ

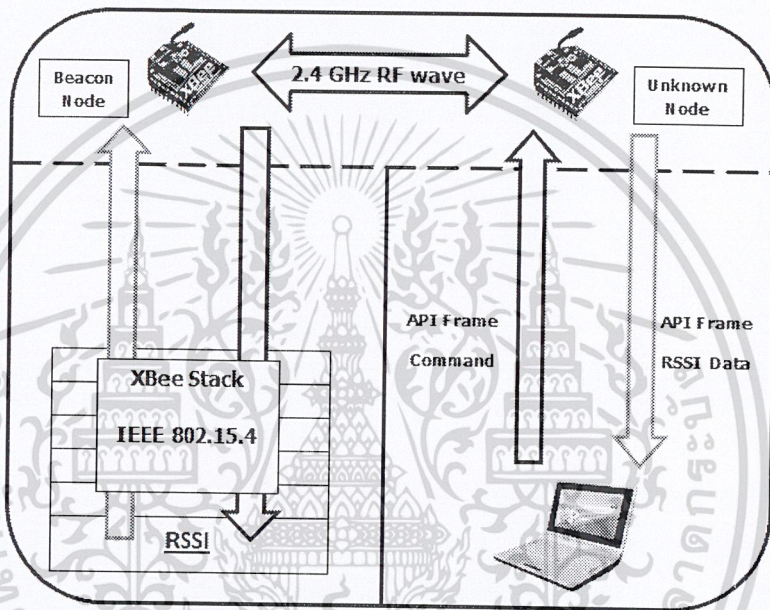
จากแผนภาพการทำงานของระบบ จะเริ่มต้นจากการเลือกฮาร์ดแวร์ที่เป็นองค์ประกอบสำคัญของระบบ ได้แก่

- **ตัวรับส่งสัญญาณวิทยุ (Transceiver)** ได้เลือกใช้โมดูล XBEE-24ZB ของบริษัท DIGI

ซึ่งมีจุดเด่นอยู่ที่ มีข้อมูลรีจิสเตอร์ในมาตรฐานของ ZigBee และสามารถสร้างRouting เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Topology ได้หลายแบบ และง่ายต่อการใช้งานและการกำหนดค่า โดยทำการเชื่อมต่อ กับ ไมโครคอนโทรลเลอร์ผ่านโมดูล USART และสามารถทำการกำหนดค่าได้โดย โปรแกรม X-CTU ของทางบริษัทผู้ผลิต

ต่อมาทำการออกแบบระบบ ในการทดลอง GUI จะสั่งให้ Unknown Node จะทำการร้องขอข้อมูลพารามิเตอร์ต่างๆ จาก Beacon Node ทุกตัว ผ่านทาง Serial Port เมื่อได้รับข้อมูลกลับมา แล้ว จะทำการประมวลผลและระบุตำแหน่งต่อไป ซึ่งสามารถแสดงภาพรวมการทำงานของระบบ ดังรูปที่ 3.2



รูปที่ 3.2 แสดงภาพรวมการทำงานของระบบ

3.2.1 รูปแบบเฟรมที่ใช้ในการสื่อสารในระดับชั้นเน็ตเวิร์กเลเยอร์

Header	Frame Length	Frame ID	Data	Checksum
1 Bytes	2 Bytes	1 Bytes	Depend on Frame type	1 Bytes

รูปที่ 3.3 แสดงรายละเอียดของของแพ็คเกจขนาด 8 Byte

รูปแบบเฟรมที่ใช้ในการสื่อสารในระดับชั้นเน็ตเวิร์กเลเยอร์มีรายละเอียดดังนี้

- Header ขนาด 1 Byte ใช้สำหรับการทำ Byte Stuffing
- Frame length ขนาด 2 Byte ใช้สำหรับระบุขนาดของเฟรม
- Frame type ขนาด 1 Byte ใช้สำหรับระบุประเภทของเฟรม
- Frame ID ขนาด 1 Byte ระบุหมายเลขของเฟรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

- DATA ขึ้นอยู่กับประเภทของเฟรม
- Checksum ขนาด 1 Byte เป็นข้อมูลของผล Checksum ตั้งแต่ Byte ที่ 4

3.2.2 ประเภทของเฟรม

ประเภทของเฟรมที่ใช้นั้น ประกอบไปด้วย

- Remote AT Command Request (0x17)
(แสดงดังรูปที่ 3.4 แพ้คเกจ Remote AT Command Request (0x17))
- Explicit Addressing ZigBee Command Frame (0x11)
(แสดงดังรูปที่ 3.5 แพ้คเกจ Explicit Addressing ZigBee Command Frame (0x11))
- ZigBee Transmit Request (0x10)
(แสดงดังรูปที่ 3.6 แพ้คเกจ ZigBee Transmit Request (0x10))

Frame Fields		Offset	Example	Description	
API P a c k e t	Start Delimiter	0	0x7E		
	Length	MSB 1	0x00	Number of bytes between the length and the checksum	
		LSB 2	0x10		
	Frame-specific Data	Frame Type	3	0x17	
	Frame ID		4	0x01	Identifies the UART data frame for the host to correlate with a subsequent ACK (acknowledgement). If set to 0, no response is sent.
			MSB 5	0x00	
	64-bit Destination Address		6	0x13	Set to the 64-bit address of the destination device. The following addresses are also supported: 0x0000000000000000 - Reserved 64-bit address for the coordinator 0x000000000000FFFF - Broadcast address
			7	0xA2	
			8	0x00	
			9	0x40	
			10	0x40	
			11	0x11	
	16-bit Destination Network Address		MSB 13	0xFF	Set to the 16-bit address of the destination device, if known. Set to 0xFFFF if the address is unknown, or if sending a broadcast.
			LSB 14	0xFE	
Remote Command Options		15	0x02 (apply changes)	Bitfield to enable various remote command options. Supported values include: 0x01 - Disable ACK 0x02 - Apply changes on remote. (If not set, AC command must be sent before changes will take effect.) 0x40 - Use the extended transmission timeout for this destination. Setting the extended timeout bit causes the stack to set the extended transmission timeout for the destination address (see chapter 4). All unused and unsupported bits must be set to 0.	
AT Command		16	0x42 (B)	Name of the command	
		17	0x48 (H)		
Command Parameter		18	0x01	If present, indicates the requested parameter value to set the given register. If no characters present, the register is queried.	
Checksum		19	0xF5	0xFF - the 8 bit sum of bytes from offset 3 to this byte.	

รูปที่ 3.4 แพ้คเกจ Remote AT Command Request (0x17) [19].

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Frame Fields		Offset	Example	Description
API P a c k e t	Start Delimiter	0	0x7E	
	Length	MSB 1	0x00	Number of bytes between the length and the checksum
		LSB 2	0x1A	
Frame-specific Data	Frame Type	3	0x11	
	Frame ID	4	0x01	Identifies the UART data frame for the host to correlate with a subsequent ACK (acknowledgement). If set to 0, no response is sent.
	64-bit Destination Address	MSB 5	0x00	Set to the 64-bit address of the destination device. The following addresses are also supported: 0x0000000000000000 - Reserved 64-bit address for the coordinator 0x000000000000FFFF - Broadcast address
		6	0x00	
		7	0x00	
		8	0x00	
		9	0x00	
		10	0x00	
	16-bit Destination Network Address	MSB 13	0xFF	Set to the 16-bit address of the destination device, if known. Set to 0xFFFE if the address is unknown, or if sending a broadcast.
		LSB 14	0xFE	
	Source Endpoint	15	0xA0	Source endpoint for the transmission.
	Destination Endpoint	16	0xA1	Destination endpoint for the transmission.
	Cluster ID	17	0x15	Cluster ID used in the transmission
	18	0x54		
	Profile ID	19	0xC1	Profile ID used in the transmission
	20	0x05		
	Broadcast Radius	21	0x00	Sets the maximum number of hops a broadcast transmission can traverse. If set to 0, the transmission radius will be set to the network maximum hops value.
	Transmit Options	22	0x00	Bitfield of supported transmission options. Supported values include the following: 0x01 - Disable ACK 0x20 - Enable APS encryption (if EE=1) 0x40 - Use the extended transmission timeout for this destination
				Enabling APS encryption decreases the maximum number of RF payload bytes by 4 (below the value reported by NP).
				Setting the extended timeout bit causes the stack to set the extended transmission timeout for the destination address. (See chapter 4.)
				All unused and unsupported bits must be set to 0.
	Data Payload	23	0x54	Data that is sent to the destination device
		24	0x78	
		25	0x44	
		26	0x61	
		27	0x74	
	28	0x61		
	Checksum	29	0x3A	0xFF - the 8 bit sum of bytes from offset 3 to this byte.

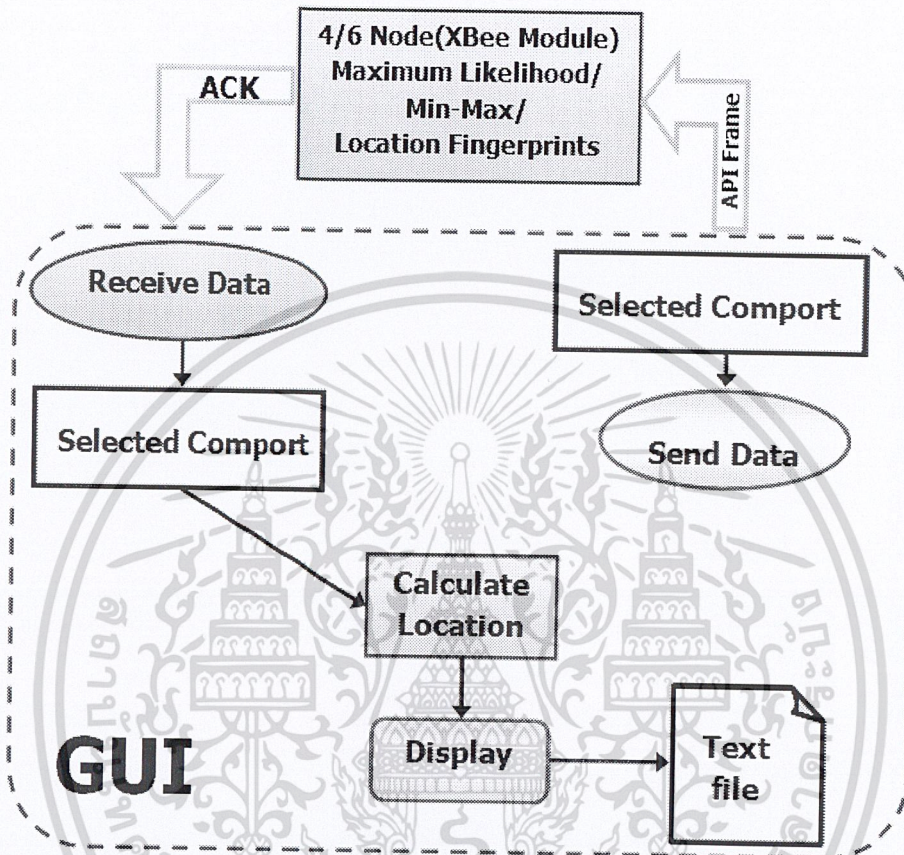
รูปที่ 3.5 แพ็กเกจ Explicit Addressing ZigBee Command Frame (0x11) [19]

Frame Fields		Offset	Example	Description	
A P I P a c k e t	Start Delimiter	0	0x7E		
	Length	MSB 1	0x00	Number of bytes between the length and the checksum	
		LSB 2	0x16		
	Frame-specific Data	Frame Type	3	0x10	
		Frame ID	4	0x01	Identifies the UART data frame for the host to correlate with a subsequent ACK (acknowledgement). If set to 0, no response is sent
		64-bit Destination Address	MSB 5	0x00	Set to the 64-bit address of the destination device. The following addresses are also supported: 0x0000000000000000 - Reserved 64-bit address for the coordinator 0x000000000000FFFF - Broadcast address
			6	0x13	
			7	0xA2	
			8	0x00	
			9	0x40	
			10	0x0A	
			LSB 12	0x27	
		16-bit Destination Network Address	MSB 13	0xFF	Set to the 16-bit address of the destination device, if known. Set to 0xFFFE if the address is unknown, or if sending a broadcast.
			LSB 14	0xFE	
		Broadcast Radius	15	0x00	Sets maximum number of hops a broadcast transmission can occur. If set to 0, the broadcast radius will be set to the maximum hops value.
		Options	16	0x00	Bitfield of supported transmission options. Supported values include the following: 0x01 - Disable ACK 0x20 - Enable APS encryption (if EE=1) 0x40 - Use the extended transmission timeout for this destination
					Enabling APS encryption decreases the maximum number of RF payload bytes by 4 (below the value reported by NP).
					Setting the extended timeout bit causes the stack to set the extended transmission timeout for the destination address. (See chapter 4.)
					All unused and unsupported bits must be set to 0.
		RF Data	17	0x54	Data that is sent to the destination device
			18	0x78	
			19	0x44	
			20	0x61	
			21	0x74	
			22	0x61	
	23		0x30		
	24	0x41			
	Checksum	25	0x13	0xFF - the 8 bit sum of bytes from offset 3 to this byte.	

รูปที่ 3.6 แพ็คเกจ ZigBee Transmit Request (0x10) [19]

3.2.3 ส่วนซอฟต์แวร์

ในส่วนซอฟต์แวร์นั้นได้มีการออกแบบดังแสดงในรูปที่ 3.7 โดยเน้นการใช้งานและการจัดการข้อมูลอย่างง่าย และสามารถนำมาประยุกต์ใช้กับการประมวลผลในขั้นสูง



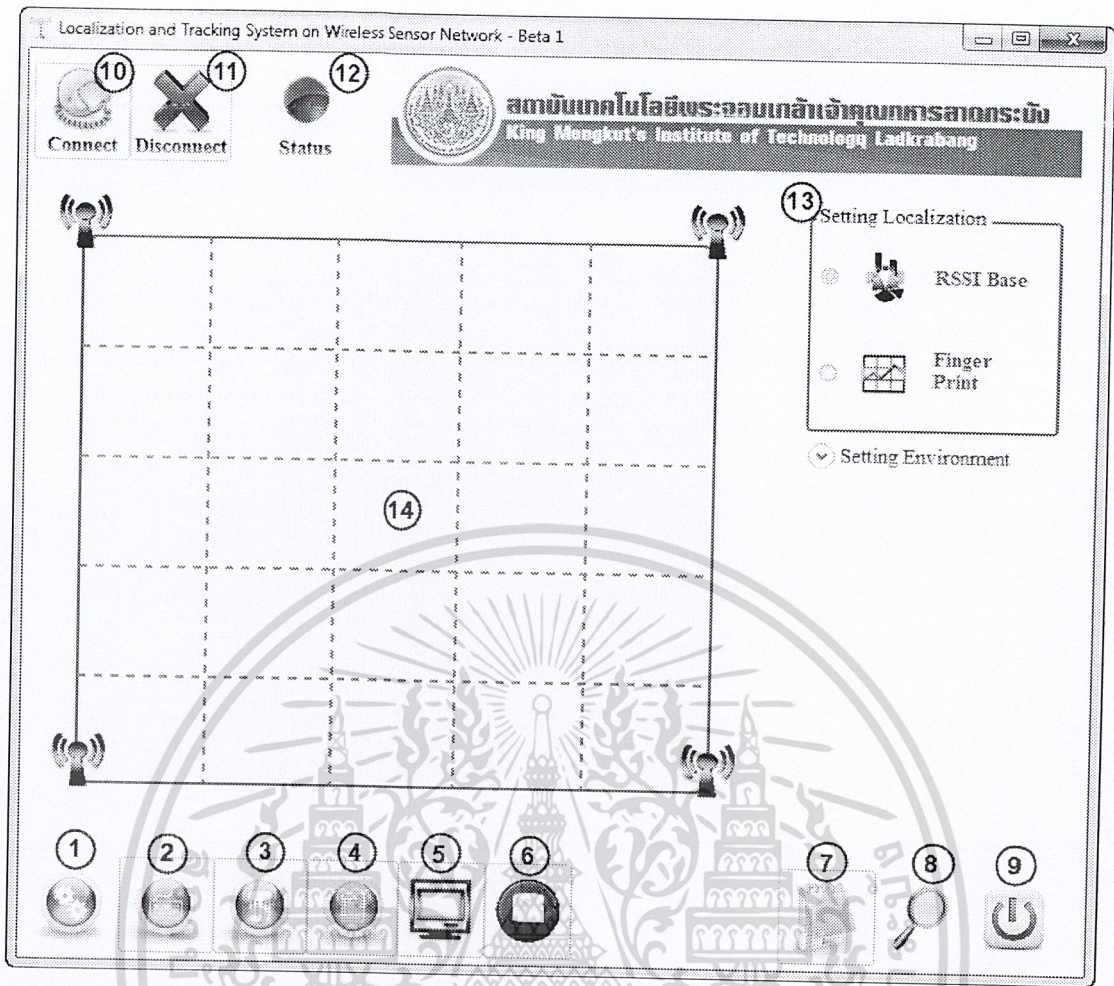
รูปที่ 3.7 ภาพรวมการออกแบบซอฟต์แวร์

จากภาพรวมการออกแบบซอฟต์แวร์ จะเริ่มต้นที่การส่งข้อมูลไปยังฮาร์ดแวร์ ผ่านคอมพิวเตอร์ที่เลือกไว้ (Selected Comport) จากนั้นฮาร์ดแวร์จะทำการส่งข้อมูลกลับมาเข้ามาในส่วนติดต่อผู้ใช้ ผ่านคอมพิวเตอร์ที่เลือกไว้ (Selected Comport) จากนั้นนำข้อมูลที่ได้รับมาทำการคำนวณหาตำแหน่ง และทำการแสดงผล ในส่วนของการแสดงผลนั้นสามารถที่จะทำการเก็บข้อมูลที่แสดงผลเป็น Text file ได้

การออกแบบซอฟต์แวร์นั้นมีการออกแบบในส่วนหลักๆ คือ

3.2.2.1 ซอฟต์แวร์ส่วนติดต่อผู้ใช้

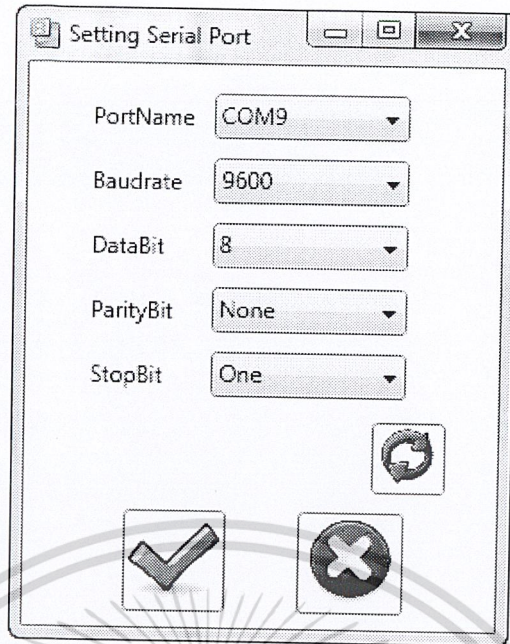
ในส่วนซอฟต์แวร์ส่วนติดต่อผู้ใช้ได้มีการสร้างหน้าโปรแกรมรับส่งข้อมูลกับ Node การทำงาน 4 Nodes และ 6 Nodes เพื่อทำการรับส่งข้อมูลระหว่างเกตเวย์และคอมพิวเตอร์



รูปที่ 3.8 หน้าของโปรแกรม ก่อนการเชื่อมต่อ

จากรูปที่ 3.8 โปรแกรมมีการทำงานในส่วนต่างๆดังนี้

- หมายเลข 1 เป็นส่วนการตั้งค่าการเชื่อมต่อ เช่น PortName , BuadRate เป็นต้น แสดงดังรูปที่ 3.9



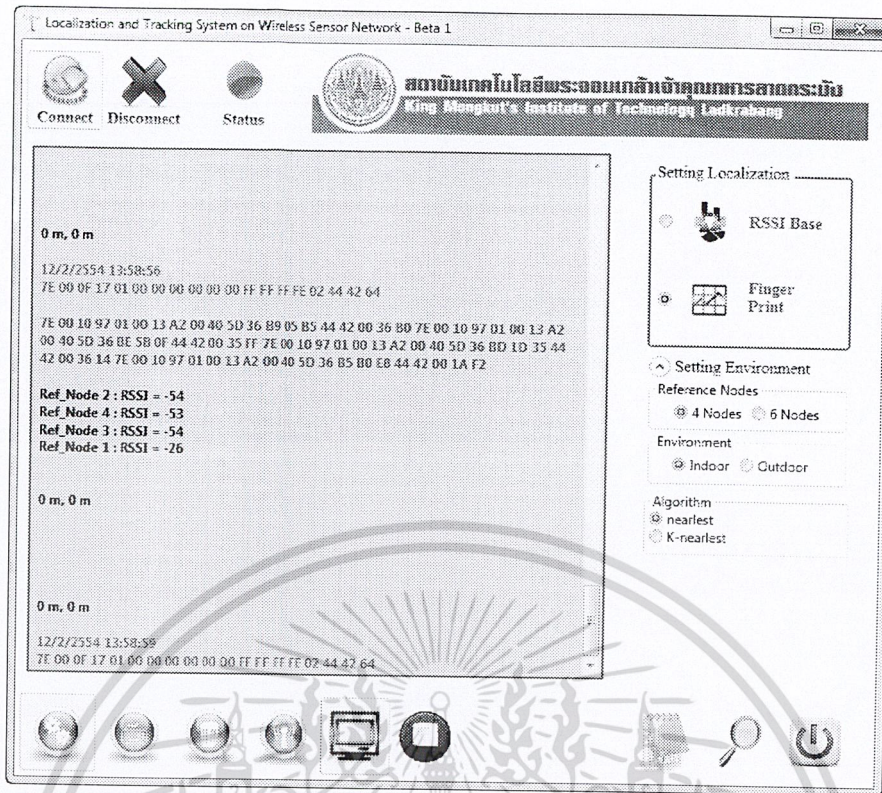
รูปที่ 3.9 ส่วนของการตั้งค่าการเชื่อมต่อแบบ Serial Port

- หมายเลข 2 เป็นส่วนของการใส่เฟรมข้อมูลที่ต้องการส่ง แสดงดังรูปที่ 3.10

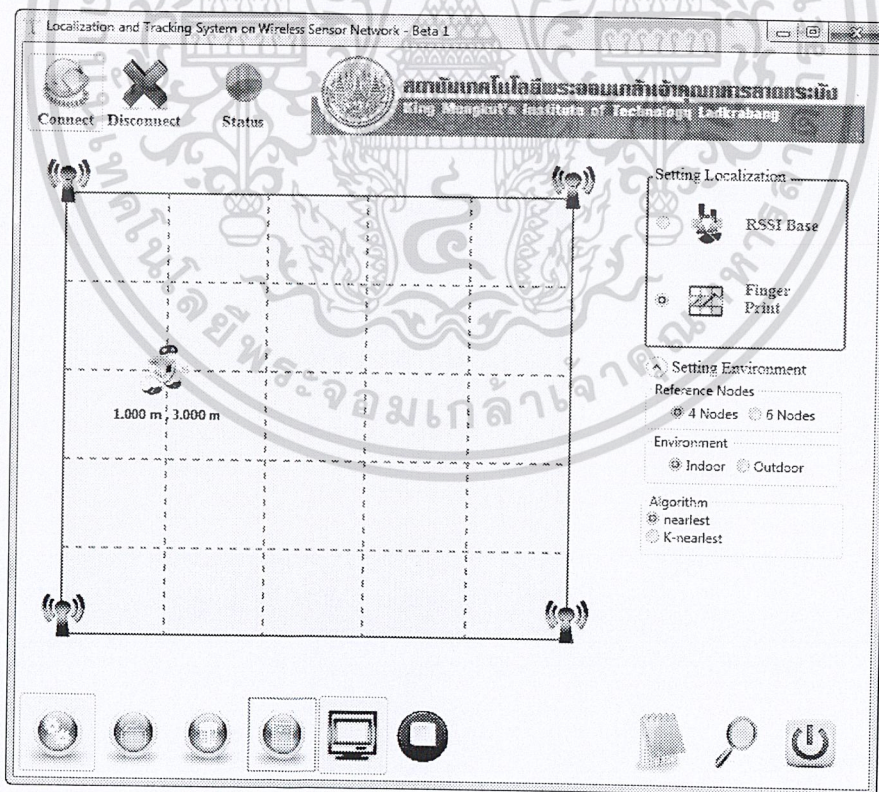


รูปที่ 3.10 ส่วนของการใส่เฟรมข้อมูลที่ต้องการส่ง

- หมายเลข 3 จะทำการลบค่าที่ทางโปรแกรมได้แสดงออกมาในรูปของ Text
- หมายเลข 4 เป็นส่วนของมุมมองของการแสดงตำแหน่งของวัตถุ ระหว่างมุมมองแบบกราฟิก หรือ มุมมองแบบ Text แสดงดังรูปที่ 3.11



(a) มุมมองแบบ Text



(b) มุมมองแบบกราฟฟิก

รูปที่ 3.11 มุมมองการแสดงตำแหน่งของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หมายเลข 5 เป็นส่วนของการเริ่มการหาตำแหน่งของวัตถุ เมื่อกดปุ่มนี้โปรแกรมจะทำการคำนวณหาตำแหน่งของวัตถุเองอัตโนมัติ ตามการตั้งค่าต่างๆที่ได้เลือกไว้
- หมายเลข 6 เป็นส่วนของการหยุดการหาตำแหน่งของวัตถุ
- หมายเลข 7 เป็นส่วนของการแสดงข้อมูลต่างๆที่โปรแกรมได้เก็บได้อยู่ในรูปของ Text file แสดงดังรูปที่ 3.12

```

12/2/2554 13:58:35E 00 0F 17 01 00 00 00 00 00 00 FF FF FF FE 02 44 42 64
7E 00 10 97 01 00 13 A2 00 40 5D 36 BE 58 0F 44 42 00 37 FD 7E 00 10 97 01 0
Ref_Node 4 : RSSI = -55Ref_Node 2 : RSSI = -58Ref_Node 3 : RSSI = -54Ref_Nod
0 m, 1 m

0 m, 1 m
12/2/2554 13:58:38E 00 0F 17 01 00 00 00 00 00 00 FF FF FF FE 02 44 42 64
7E 00 10 97 01 00 13 A2 00 40 5D 36 BE 58 0F 44 42 00 39 FB 7E 00 10 97 01 0
Ref_Node 4 : RSSI = -57Ref_Node 3 : RSSI = -55Ref_Node 2 : RSSI = -53Ref_Nod
0 m, 0 m

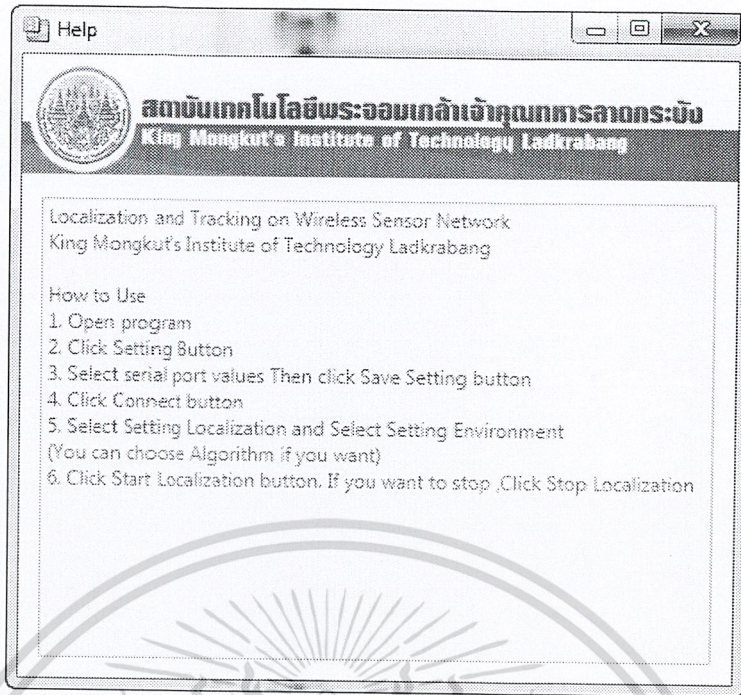
0 m, 0 m
12/2/2554 13:58:40E 00 0F 17 01 00 00 00 00 00 00 FF FF FF FE 02 44 42 64
7E 00 10 97 01 00 13 A2 00 40 5D 36 B9 05 B5 44 42 00 3D A9 7E 00 10 97 01 0
Ref_Node 2 : RSSI = -61Ref_Node 4 : RSSI = -58Ref_Node 3 : RSSI = -57Ref_Nod
0 m, 1 m

0 m, 1 m
12/2/2554 13:58:43E 00 0F 17 01 00 00 00 00 00 00 FF FF FF FE 02 44 42 64
7E 00 10 97 01 00 13 A2 00 40 5D 36 BD 1D 35 44 42 00 37 13 7E 00 10 97 01 0
Ref_Node 3 : RSSI = -55Ref_Node 2 : RSSI = -52Ref_Node 4 : RSSI = -56Ref_Nod
0 m, 0 m

```

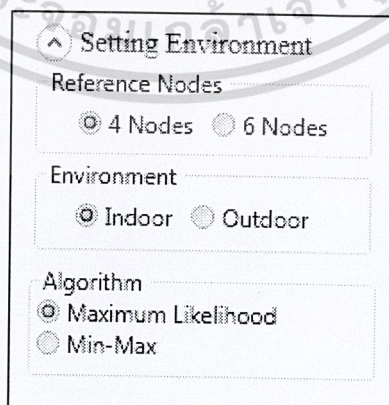
รูปที่ 3.12 ตัวอย่างข้อมูลที่โปรแกรมได้ทำการเก็บไว้

- หมายเลข 8 เป็นส่วนของการแสดงวิธีการใช้งานของ โปรแกรม แสดงดังรูปที่ 3.13



รูปที่ 3.13 ส่วนของการแสดงวิธีการใช้งานของ โปรแกรม

- หมายเลข 9 เมื่อกดปุ่มนี้จะเป็นการออกจาก โปรแกรม
- หมายเลข 10 ทำการเชื่อมต่อ Serial Port ตามค่าที่ได้เลือกจากหมายเลข 1
- หมายเลข 11 ทำการยกเลิกการเชื่อมต่อ Serial Port
- หมายเลข 12 แสดงสถานะการเชื่อมต่อของ Serial Port
- หมายเลข 13 เป็นส่วนการตั้งค่าของวิธีการหาตำแหน่งของวัตถุ เมื่อทำการเชื่อมต่อตามหมายเลข 10 แล้ว ส่วนนี้ถึงจะใช้งานได้ และสามารถตั้งค่าอื่นๆเพิ่มเติมได้ เช่น สภาพแวดล้อม อัลกอริทึมที่จะใช้ เป็นต้น แสดงดังรูปที่ 3.14



รูปที่ 3.14 ส่วนการตั้งค่าเพิ่มเติมเมื่อมีการเชื่อมต่อ Serial Port แล้ว

- หมายเลข 14 เป็นส่วนพื้นที่ของการแสดงตำแหน่งของวัตถุที่โปรแกรมคำนวณได้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ขึ้นด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

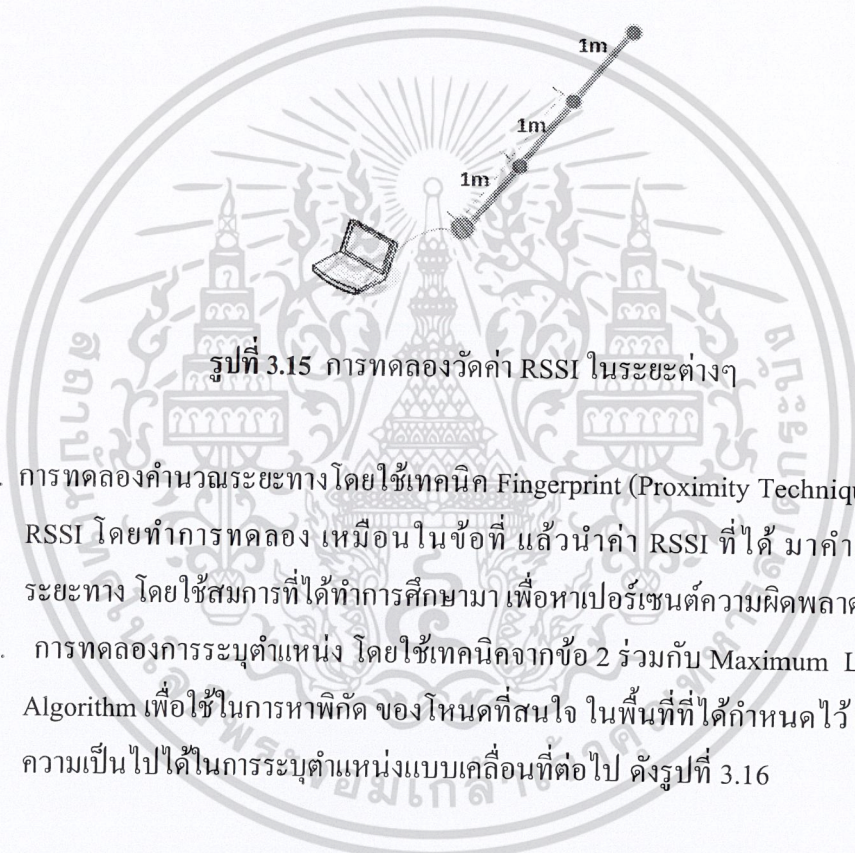
3.3 แนวทางการศึกษาและการทดลอง

การศึกษาและการทดลองนั้น แบ่งออกเป็น สองส่วน คือ การศึกษาเกี่ยวกับค่า Receive Signal Strength Indicator (RSSI) และ การทดลองการระบุตำแหน่ง

3.3.1 การทดลองเกี่ยวกับ Receive Signal Strength Indicator

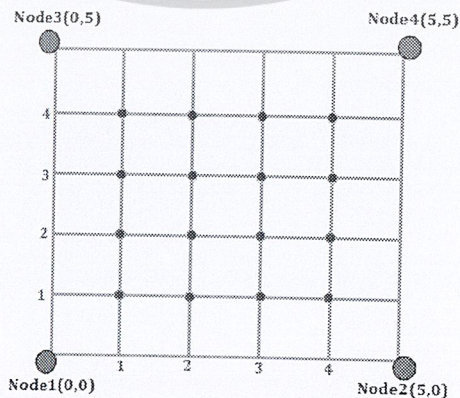
แบ่งการทดลองออกเป็นสามส่วน คือ

1. การศึกษาคุณแนวโน้มการเปลี่ยนแปลงของค่า RSSI โดยจะทำการวัดค่า RSSI จาก XBee ในระยะต่างๆ โดยทำการเพิ่มระยะขึ้นไปเรื่อยๆ ดังรูปที่ 3.15



รูปที่ 3.15 การทดลองวัดค่า RSSI ในระยะต่างๆ

2. การทดลองคำนวณระยะทาง โดยใช้เทคนิค Fingerprint (Proximity Technique) จากค่า RSSI โดยทำการทดลอง เหมือนในข้อที่ แล้วนำค่า RSSI ที่ได้ มาคำนวณเป็นระยะทาง โดยใช้สมการที่ได้ทำการศึกษา มา เพื่อหาเปอร์เซ็นต์ความผิดพลาด
3. การทดลองการระบุตำแหน่ง โดยใช้เทคนิคจากข้อ 2 ร่วมกับ Maximum Likelihood Algorithm เพื่อใช้ในการหาพิกัด ของ โหนดที่สนใจ ในพื้นที่ที่ได้กำหนดไว้เพื่อศึกษาความเป็นไปได้ในการระบุตำแหน่งแบบเคลื่อนที่ต่อไป ดังรูปที่ 3.16



รูปที่ 3.16 การทดลองหาตำแหน่ง โดยใช้ Maximum Likelihood Algorithm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ขึ้นด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การทดลองการระบุตำแหน่ง

ในส่วนนี้ได้ทำการทดลองการระบุตำแหน่ง ทั้งหมด 3 แบบ คือ Maximum Likelihood Algorithm , Min-Max Algorithm และ Location Fingerprint (Nearest / 3-Nearest) ตามลำดับซึ่งแบ่งเป็นการทดลองดังต่อไปนี้

1. การทดลองเก็บค่า RSSI เพื่อเป็นฐานข้อมูลของวิธีการหาตำแหน่งแบบ Location Fingerprint โดยจะทำการเก็บค่า RSSI เฉลี่ย ทั้งหมด 10 ครั้ง ระหว่างโหนดอ้างอิง กับ โหนดเป้าหมาย ในทุกๆตำแหน่งบนพื้นที่ที่ได้กำหนดไว้
2. ทำการเก็บค่า RSSI เพื่อใช้ในการทดลองระบุตำแหน่งทั้ง 3 วิธี โดยจะทำการทดลองทั้งหมด 5 ครั้ง ครั้งละ 5 เส้นทาง
 - 2.1 ตามแนวมุมด้านล่างซ้ายมือถึงมุมด้านบนขวามือของพื้นที่ทดลองตั้งแต่ (0,0) โดยเลื่อนตำแหน่งทีละ (0.33,0.33) เมตร
 - 2.2 ตามแนวมุมด้านล่างขวามือถึงมุมด้านบนซ้ายมือของพื้นที่ทดลองตั้งแต่ (5,0) โดยเลื่อนตำแหน่งทีละ (-0.33,0.33) เมตร
 - 2.3 ตามแนวนอนของพื้นที่ ตั้งแต่ (0,2.50) จนถึง (5,2.50) โดยเลื่อนตำแหน่งทีละ (0.25,0) เมตร
 - 2.4 ตามแนวตั้งของพื้นที่ ตั้งแต่ (2.50,0) จนถึง (2.50,5) โดยเลื่อนตำแหน่งทีละ (0,0.25) เมตร
 - 2.5 ตามตำแหน่ง 36 ตำแหน่ง ตั้งแต่ (0,0),(1,0) จนถึง (5,5)
3. การทดลองระบุตำแหน่งวัตถุ ในแต่ละวิธี และทำการเปรียบเทียบค่าความผิดพลาด โดยจะนำค่าที่ได้จากการทดลองในข้อที่ 2 มาทำการคำนวณหาตำแหน่ง ในแต่ละวิธี แล้วนำมาเปรียบเทียบกันเพื่อหาประสิทธิภาพของแต่ละวิธี

3.4 บทสรุป

ในบทนี้ได้กล่าวถึงโครงสร้างของระบบ ทั้งในส่วนฮาร์ดแวร์ การเชื่อมต่อและการใช้งาน Graphic User Interface รวมไปถึงแนวทางและวิธีการทดลองการระบุตำแหน่งทั้งในแบบหยุดนิ่งและแบบเคลื่อนที่ โดยใช้วิธีการระบุตำแหน่งแบบ Range-Based และ Fingerprint

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองเรื่องการวัดค่า RSSI (Receive strength Signal Indicator)

จุดประสงค์

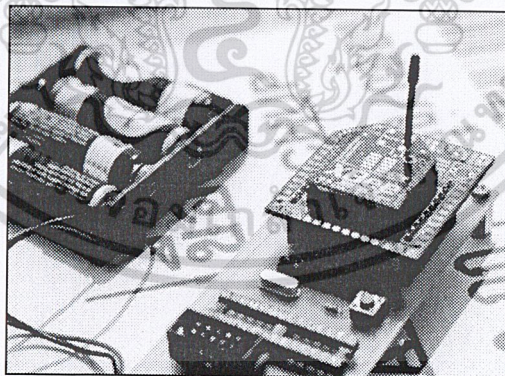
- เพื่อดูแนวโน้มการเปลี่ยนแปลงของค่า RSSI ทุกๆระยะ 1 เมตร รวมทั้งสิ้น 10 เมตร ว่าเป็นไปตามผลทางทฤษฎีหรือไม่ ทั้งภายในอาคารและภายนอกอาคาร

สมมติฐาน

- ผลการเปลี่ยนแปลงค่า RSSI คาดว่าจะต้องลดลง(ค่าเป็นลบมากขึ้น) เมื่อระยะทางเพิ่มขึ้น

อุปกรณ์

- XBee Module 2 ตัว (ตัวส่ง 1 ตัว, ตัวรับ 1 ตัว) แสดงดังรูปที่ 4.1
- คอมพิวเตอร์ 1 เครื่อง
- กล่อง ขนาด 21x29.7 ซม. 2 กล่อง
- โปรแกรมใช้ในการหาค่า RSSI



รูปที่ 4.1 บอร์ด XBee Module

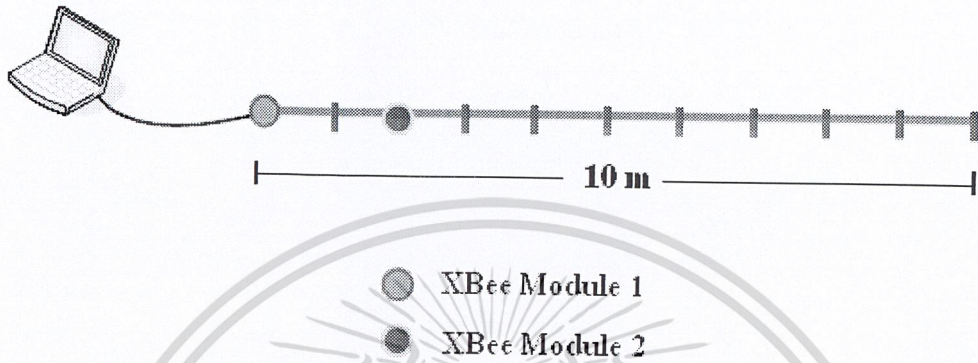
วิธีการทดลอง

- ทำการจัดวาง XBee Module 2 ตัว ให้ห่างกันเป็นระยะ 1 เมตร แสดงดังรูปที่ 4.2
- หาค่า RSSI จากโปรแกรมที่ได้เตรียมไว้ เก็บค่าทั้งหมด 50 ครั้ง และทำการหาค่าเฉลี่ย

และบันทึกผลค่าที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 48.จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

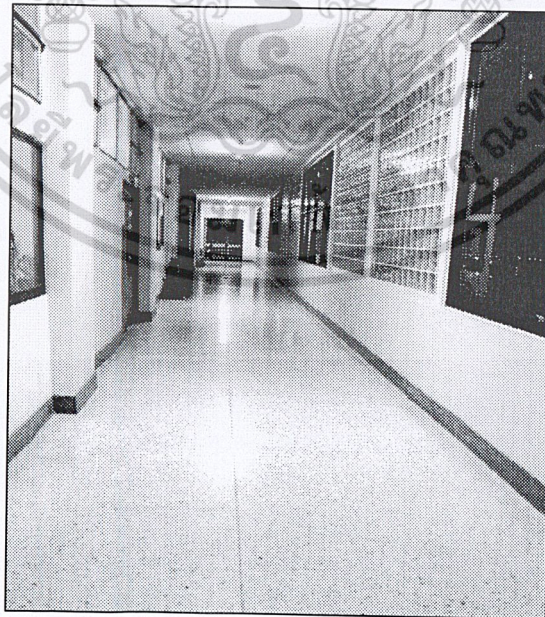
- เมื่อเสร็จสิ้นข้อ 2 ให้ทำตามข้อที่ 1 และ 2 โดยจัดวาง XBee Module ให้ห่างกันทุกๆ ระยะ 1 เมตร จนกระทั่งระยะห่างระหว่าง XBee Module เป็น 10 เมตร และบันทึกผลที่ได้ทุกๆ ระยะ 1 เมตร
- ทำตามขั้นตอนที่ 1-3 ทั้งภายในอาคารและนอกอาคาร และทำการบันทึกผล



รูปที่ 4.2 ภาพจำลองแบบการทดลองการจัดวางตำแหน่งของ XBee Module เพื่อหาค่า RSSI

สภาพแวดล้อม

- ภายในอาคาร ช่วงทางเดินหน้าห้อง ชั้น 12 ตึก E12 คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง แสดงดังรูปที่ 4.3



รูปที่ 4.3 สถานที่ในการทดลองภายในอาคาร

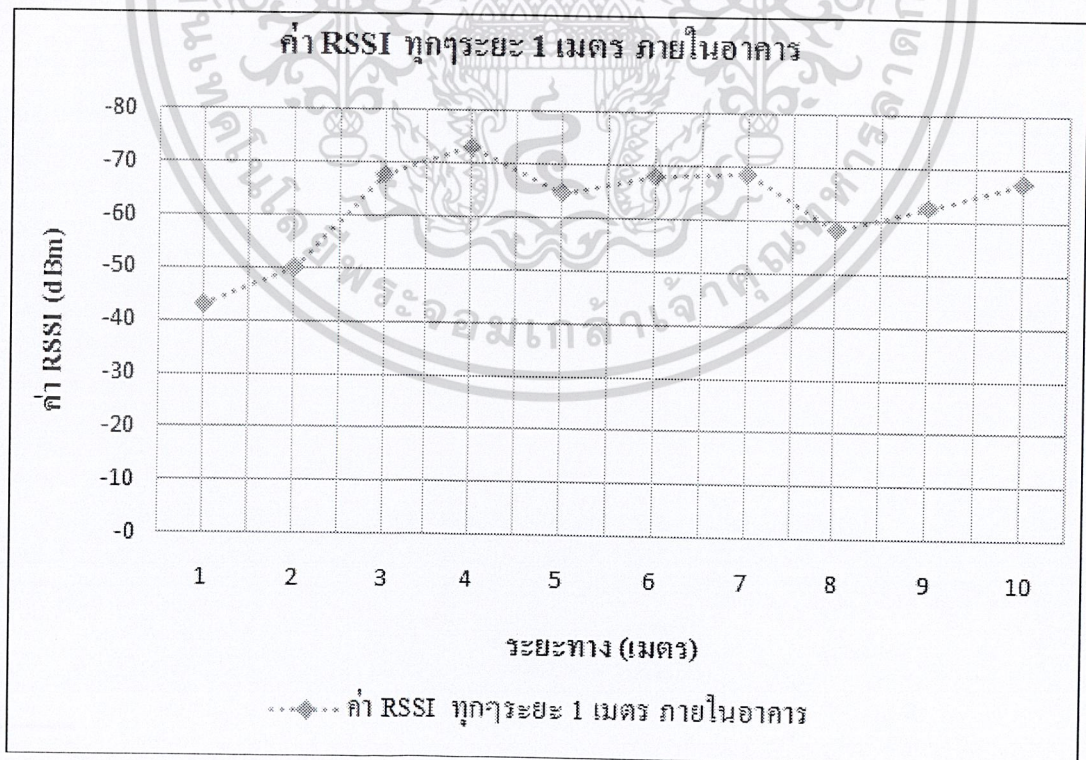
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา⁴⁹ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ภายนอกอาคาร ลานกิจกรรมบริเวณลานกิจกรรมด้านข้างสนามแบดมินตัน สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง แสดงดังรูปที่ 4.4



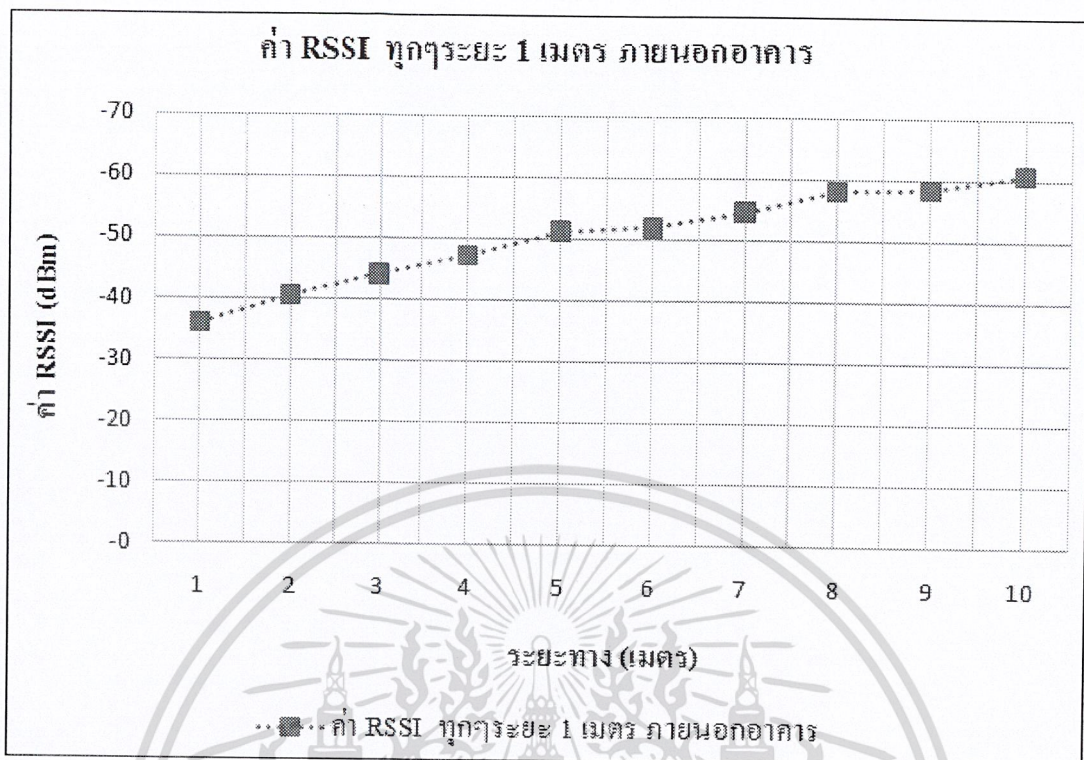
รูปที่ 4.4 สถานที่ในการทดลองภายนอกอาคาร

ผลการทดลอง

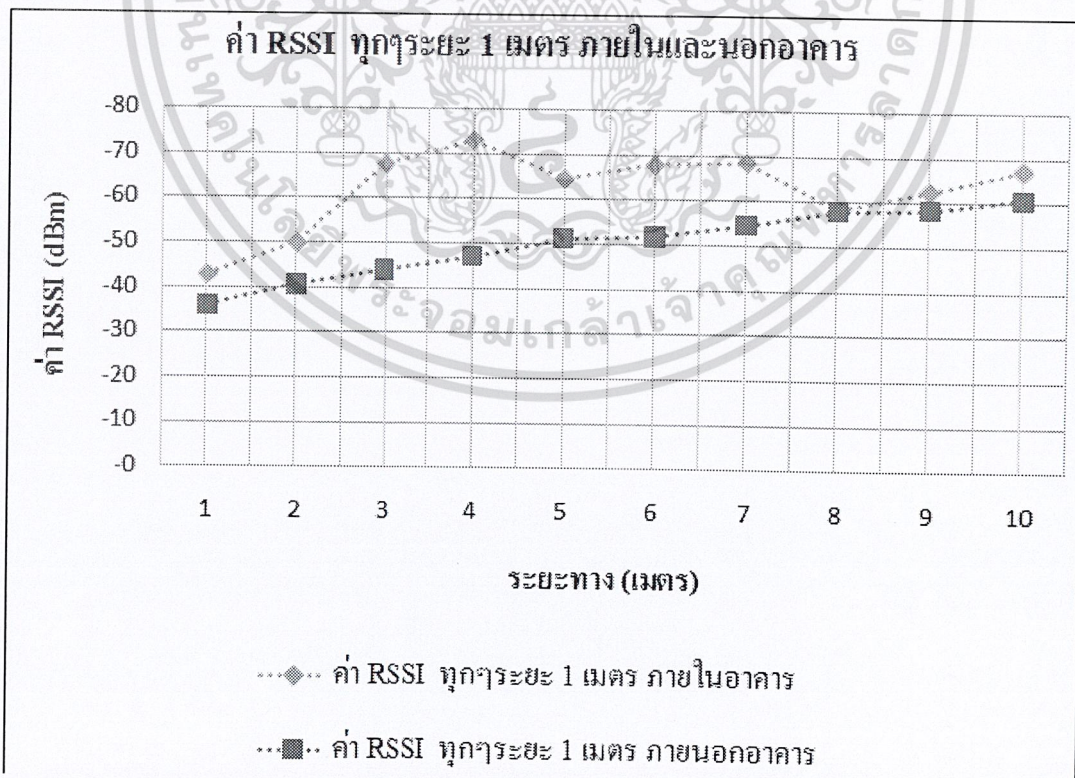


รูปที่ 4.5 กราฟแสดงค่า RSSI ทุกๆช่วง 1 เมตร ในระยะทาง 10 เมตร ภายในอาคาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 50 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 กราฟแสดงค่า RSSI ทุกๆช่วง 1 เมตร ในระยะทาง 10 เมตร ภายนอกอาคาร



รูปที่ 4.7 กราฟเปรียบเทียบค่า RSSI ทั้งภายในและภายนอกอาคาร
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 51 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

จากรูปที่ 4.5 ค่า RSSI ที่วัดได้จากภายในอาคาร มีค่าที่ไม่เพิ่มขึ้นตามระยะทาง ค่า RSSI มีค่าขึ้นลงไม่แน่นอน เนื่องจากเกิดการสะท้อน การหักเห การจางหายของสัญญาณภายในอาคาร ส่วนรูปที่ 4.6 ค่า RSSI ที่วัดได้จากภายนอกอาคาร มีค่าลดลง(ค่าลบมากขึ้น)เรื่อยๆตามระยะทางที่มากขึ้น เนื่องจากภายนอกอาคารไม่มีสิ่งกีดขวาง เพราะฉะนั้น ค่า RSSI ที่ได้จากการทดลองภายนอกอาคาร มีความน่าเชื่อถือมากกว่า สภาพแวดล้อมภายนอกอาคารจึงเหมาะสมที่จะนำมาคำนวณหาตำแหน่งของวัตถุต่อไป

4.2 การทดลองการหาค่าตัวแปรของสมการเพื่อใช้ในการหาระยะทางและหาตำแหน่งของวัตถุ

จุดประสงค์

- เพื่อหาค่าตัวแปร A และ n ภายนอกอาคาร จากสมการ (2.1)

$$RSSI = -10n \log_{10}(d) + A$$

เมื่อ RSSI คือค่าความแรงของสัญญาณวิทยุ มีหน่วยเป็น เดซิเบลมิลลิวัตต์ (dBm)

n คือค่าคงที่การกระจายสัญญาณของสถานที่นั้นๆ

d คือระยะห่างระหว่างเซ็นเซอร์ไร้สาย (เมตร)

A คือค่าความแรงของสัญญาณที่ได้รับใน 1 เมตร (dBm)

จากสมการข้างต้น ถ้าค่าระยะทาง (d) เพิ่มมากขึ้นก็จะส่งผลให้ค่า RSSI ลดลง ซึ่งบอกค่าความแรงของสัญญาณ สามารถนำผลนี้ไปใช้เพื่อหาระยะทางและปริมาณตำแหน่งวัตถุ

สมมุติฐาน

- ค่าของตัวแปร A และ n ที่ได้มา ทำให้ค่าระยะทางที่ได้จากการคำนวณเท่ากับกับระยะทางจริง

อุปกรณ์

- XBee Module 5 ตัว (ตัวส่ง 1 ตัว , ตัวรับ 4 ตัว)
- คอมพิวเตอร์ 1 เครื่อง
- กล่อง ขนาด 21x29.7 ซม. 2 กล่อง
- โปรแกรมใช้ในการหาค่า RSSI

วิธีการทดลอง

- ทำการหาค่า A โดยการวัดค่า RSSI โดยรอบตัวรับทั้งหมด 4 ตัว เป็นระยะรัศมี 1 เมตร โดยวัดทุกๆ 45° เก็บค่าครั้งละ 50 ครั้ง แล้วนำมาหาค่าเฉลี่ย และบันทึกผล ดังรูปที่ 4.8
- ทำตามข้อ 1 จนครบจำนวนของตัวรับ แล้วนำค่า RSSI ที่ทำการเฉลี่ยแล้วของแต่ละตัว มาทำการหาค่าเฉลี่ยรวม ค่าที่ได้จะเป็น ค่า A ที่ต้องการ
- ทำการหาค่า n ดังนี้
 - ตัวรับทุกตัว ทำการวัดค่า RSSI ทุกๆระยะ 1 เมตร เป็นระยะทั้งสิ้น 10 เมตร (เหมือนการทดลองที่ 1) ค่า RSSI ที่ได้นี้จะป็นค่าที่ได้จากการวัดจริง เพื่อนำไปเปรียบเทียบกับค่า RSSI ที่ได้จากการคำนวณ
 - ค่า RSSI ที่ได้จากการคำนวณ หาได้จากสมการ (2.1)

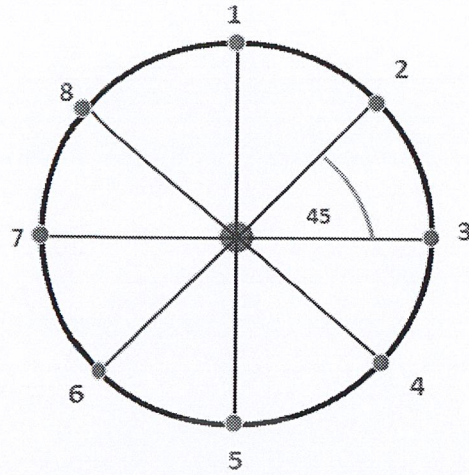
$$RSSI = -10n \log_{10}(d) + A$$

โดยการแทนค่า A ที่ได้จากข้อที่ 1 และทำการแทนค่า d (ระยะห่างระหว่างเซ็นเซอร์ไร้สาย ในหน่วยเมตร) เป็น 1,2,3,...,10 ตามลำดับ ส่วนค่า n ให้แทนค่าตั้งแต่ 1 ถึง 4 เมื่อทำการคำนวณแล้วจะได้ ค่า RSSI จากการคำนวณ ทุกๆระยะ 1 เมตร จนถึง 10 เมตร ที่มีค่า n เป็น 1 ถึง 4

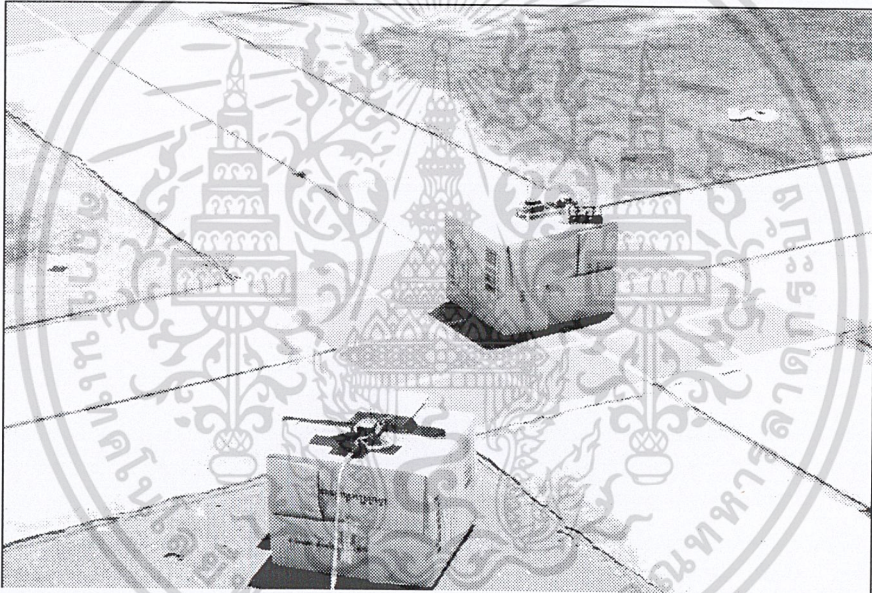
- นำค่านี้ไปเปรียบเทียบกับข้อ 3.2 ว่า ค่า RSSI ที่ได้จากการวัดจริงใกล้เคียงกับค่า RSSI ที่ได้จากการคำนวณ ที่ค่า n ต่างๆกัน มากที่สุด ค่า RSSI ที่ใกล้เคียงที่สุด จะมีค่า n ที่ต้องการ ในการหาค่า RSSI ที่ใกล้เคียงที่สุด หาได้จาก **Root Mean Square Error (RMSE)** ซึ่งแสดงดังสมการ (4.1)

$$RMSE = \sqrt{\frac{\sum(\text{ค่า RSSI ที่วัดจริง} - \text{ค่า RSSI จากการคำนวณ})^2}{\text{จำนวนค่า RSSI ที่ทำการเปรียบเทียบ}}} \quad (4.1)$$

ค่า RMSE ที่น้อยที่สุดแสดงว่ามีค่า RSSI จากการวัดจริงกับการคำนวณใกล้เคียงกันที่สุด เพราะฉะนั้นจะได้ค่า n ที่เหมาะสมที่สุด



รูปที่ 4.8 แสดงการจัดวางตำแหน่งของ XBee Module เพื่อทำการหาค่า A



รูปที่ 4.9 การทดลองหาค่า A และ n ทุกๆช่วง 1 เมตร

สภาพแวดล้อม

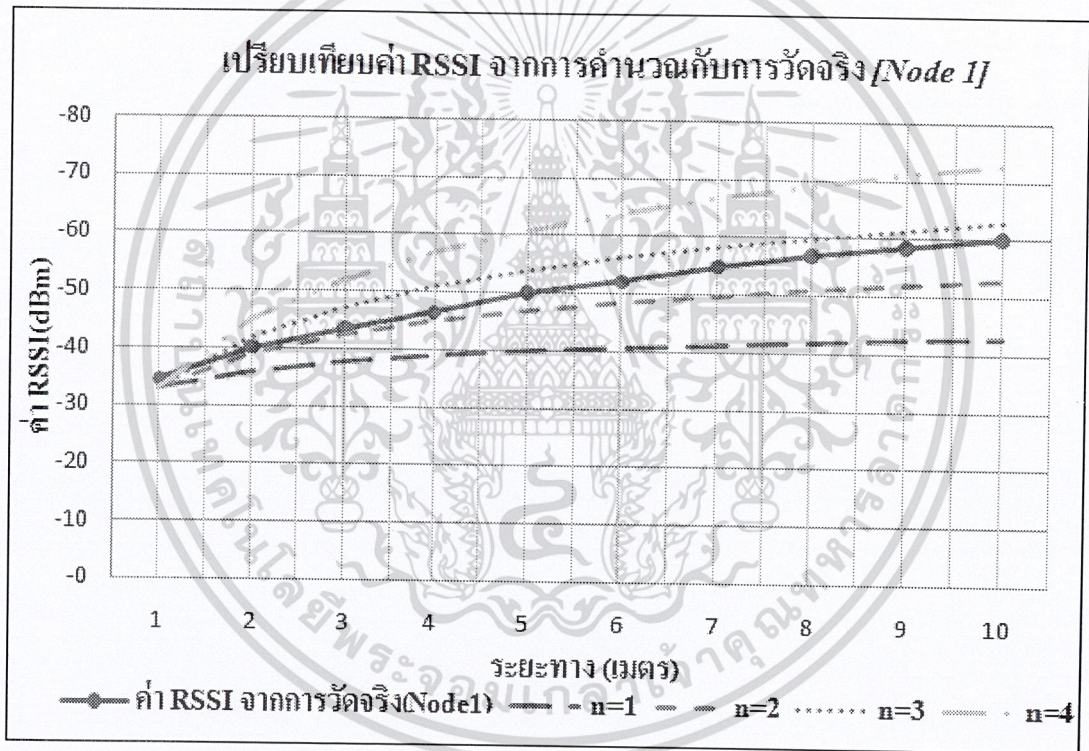
- ภายนอกอาคาร ลานกิจกรรมบริเวณลานกิจกรรมด้านข้างสนามแบดมินตัน สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (เหมือนการทดลองที่ 4.1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา⁵⁴ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

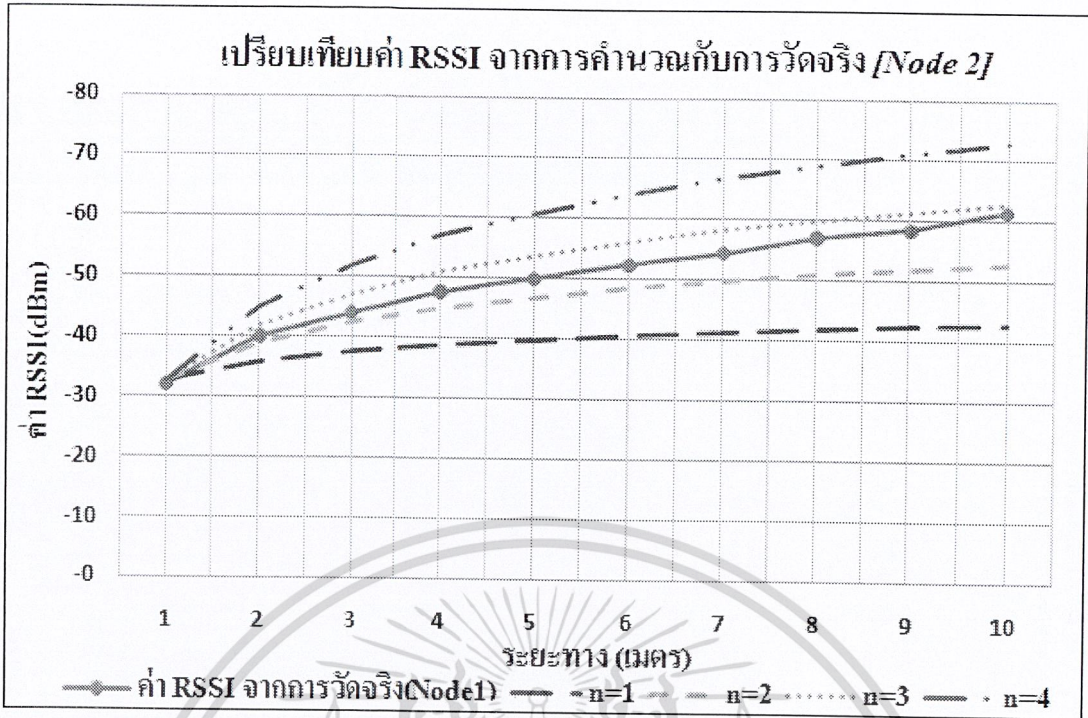
ตารางที่ 4.1 ตารางแสดงค่าเฉลี่ยของค่าตัวแปร A

Node	ค่าเฉลี่ยของค่า A แต่ละ Node
1	33.0325
2	32.1825
3	32.7450
4	33.4625
ค่าเฉลี่ยของค่า A ทั้งหมด 4 Node	32.855625

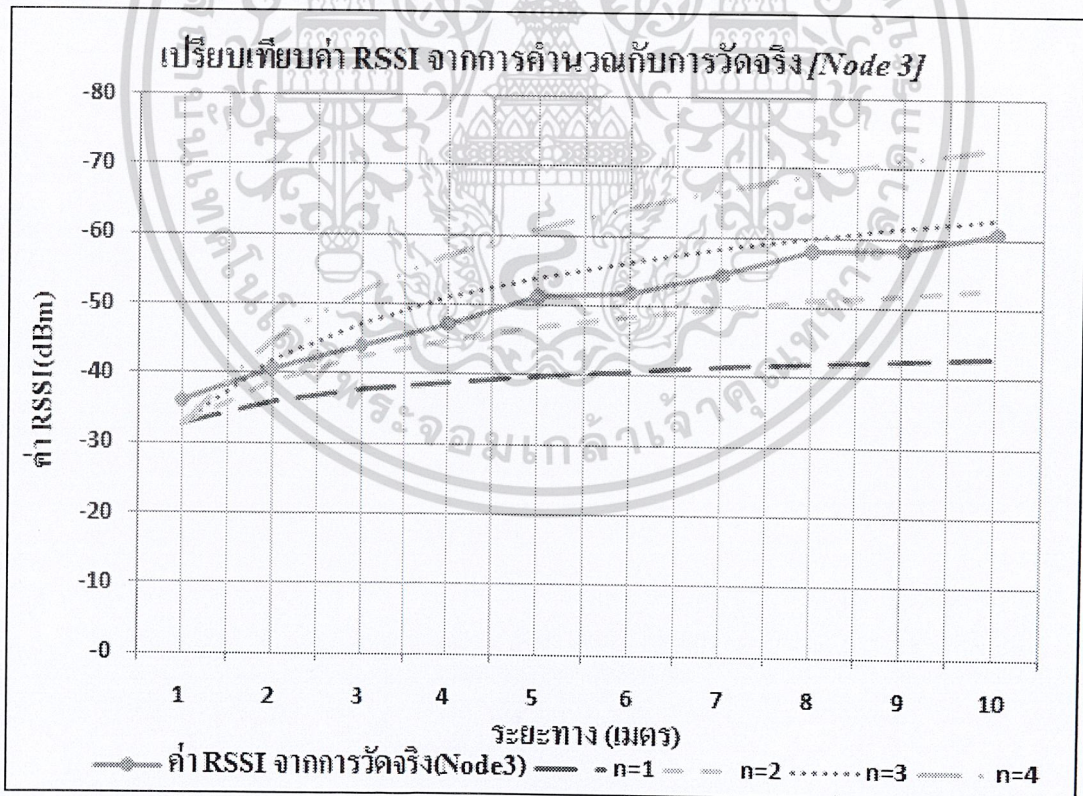


รูปที่ 4.10 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 55 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

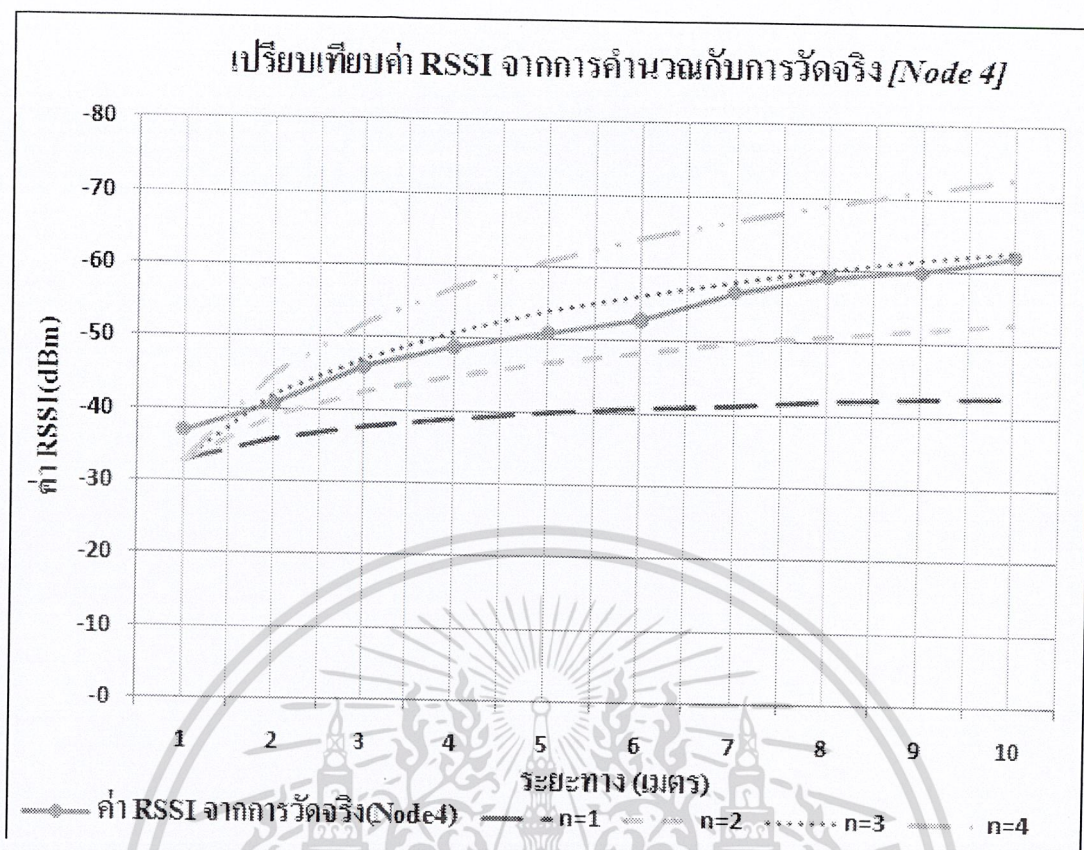


รูปที่ 4.11 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 2

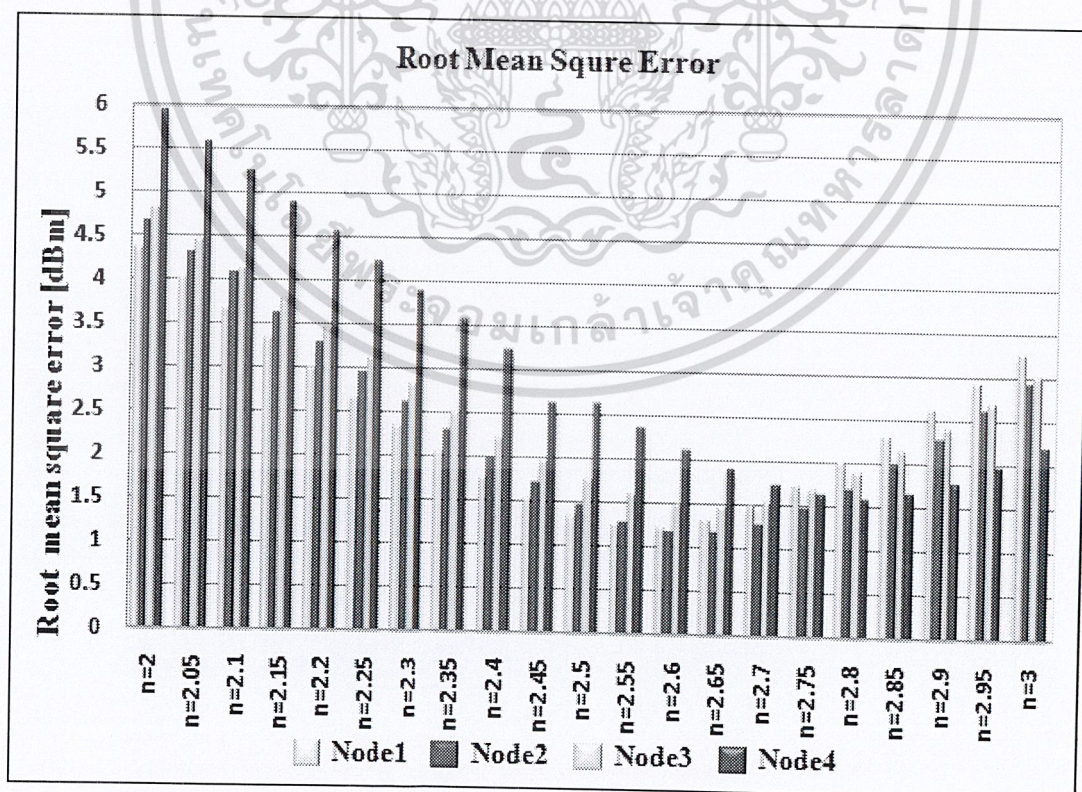


รูปที่ 4.12 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 56 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 กราฟแสดงการเปรียบเทียบค่า RSSI จากการคำนวณกับการวัดจริง ของ Node 4



รูปที่ 4.14 กราฟแสดงค่า RMSE ของ Node 1-4 โดยที่ค่ามีค่า n แตกต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 57 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

ค่าตัวแปร A และ n จากการทดลองภายนอกอาคาร ได้ว่า ค่า $A = 32.855625$ จากรูปที่ 4.10-4.13 จะเห็นได้ว่าค่า RSSI ที่ได้จากการคำนวณที่มีค่า n ระหว่าง 2-3 มีค่า RSSI ใกล้เคียงกับค่าที่วัดจริงมากที่สุด และจากรูปที่ 4.14 จะเห็นว่าค่า RMSE ที่น้อยที่สุดจะอยู่ที่ค่า n ประมาณ 2.65 เพราะฉะนั้นในการทดลองหาค่าตำแหน่งของวัตถุภายนอกอาคารจะใช้ค่า $A = 32.855625$ และค่า $n = 2.65$

4.3 การทดลองหาค่าตำแหน่งของวัตถุด้วยค่า RSSI(Receive strength Signal Indicator)

จุดประสงค์

- เพื่อหาหรือระบุตำแหน่งของวัตถุภายในพื้นที่ 5×5 เมตร ภายนอกอาคาร

สมมุติฐาน

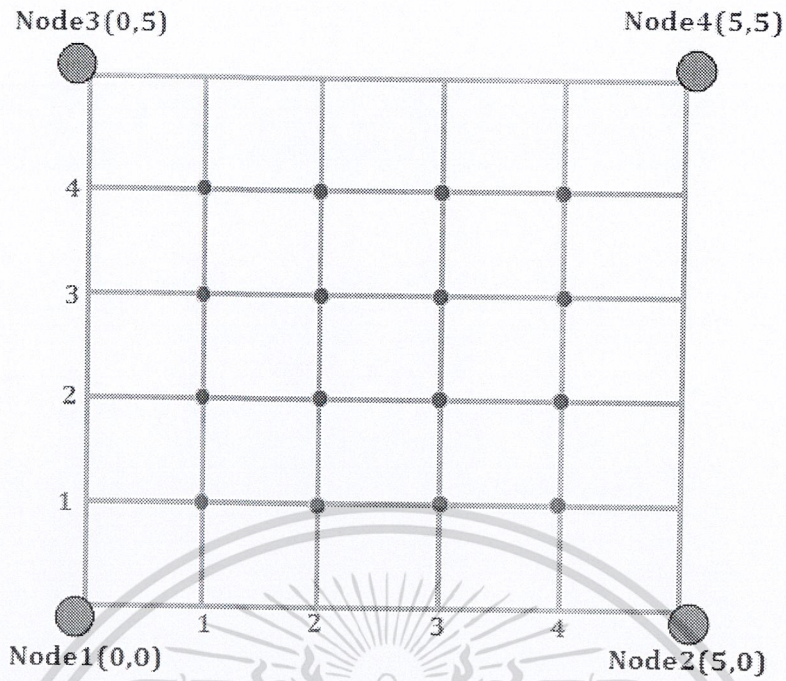
- ตำแหน่งของวัตถุที่คำนวณมาได้ตรงตามตำแหน่งจริงที่วัตถุอยู่ในขณะนั้น

อุปกรณ์

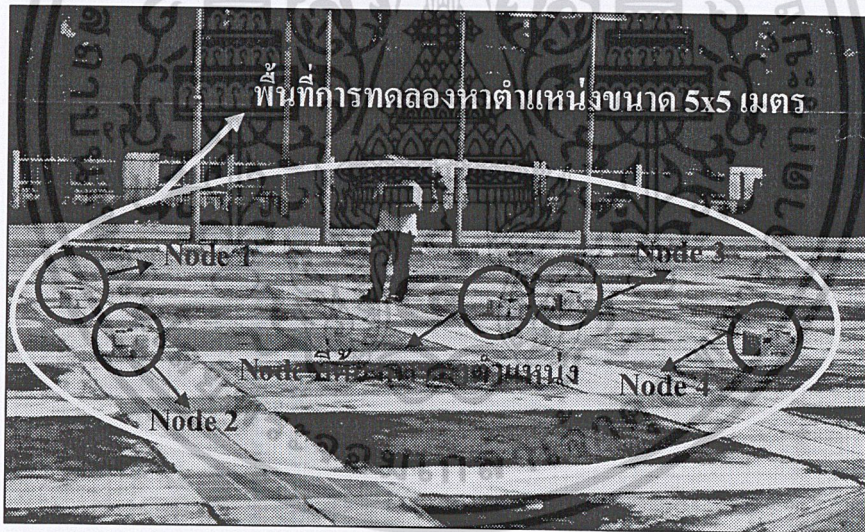
- XBee Module 5 ตัว (ตัวส่ง 1 ตัว , ตัวรับ 4 ตัว)
- คอมพิวเตอร์ 1 เครื่อง
- กล่อง ขนาด 21×29.7 ซม. 5 กล่อง
- โปรแกรมใช้ในการหาค่า RSSI และ หาค่าตำแหน่งของวัตถุ

วิธีการทดลอง

- จัดเตรียมพื้นที่ขนาด 5×5 เมตร โดยตัวรับทั้ง 4 ตัว จะอยู่ที่แต่ละมุมของพื้นที่ แสดงดังรูปที่ 4.15
- นำตัวส่ง ไปไว้ในพื้นที่ ตรงตำแหน่งที่ต้องการ และทำการส่งค่าเพื่อให้โปรแกรมคำนวณหาตำแหน่ง
- ทำตามขั้นตอนที่ 1-2 ทั้งภายในอาคารและนอกอาคาร และทำการบันทึกผล



รูปที่ 4.15 ภาพจำลองการจัดพื้นที่เพื่อที่จะคำนวณหาตำแหน่งของวัตถุ



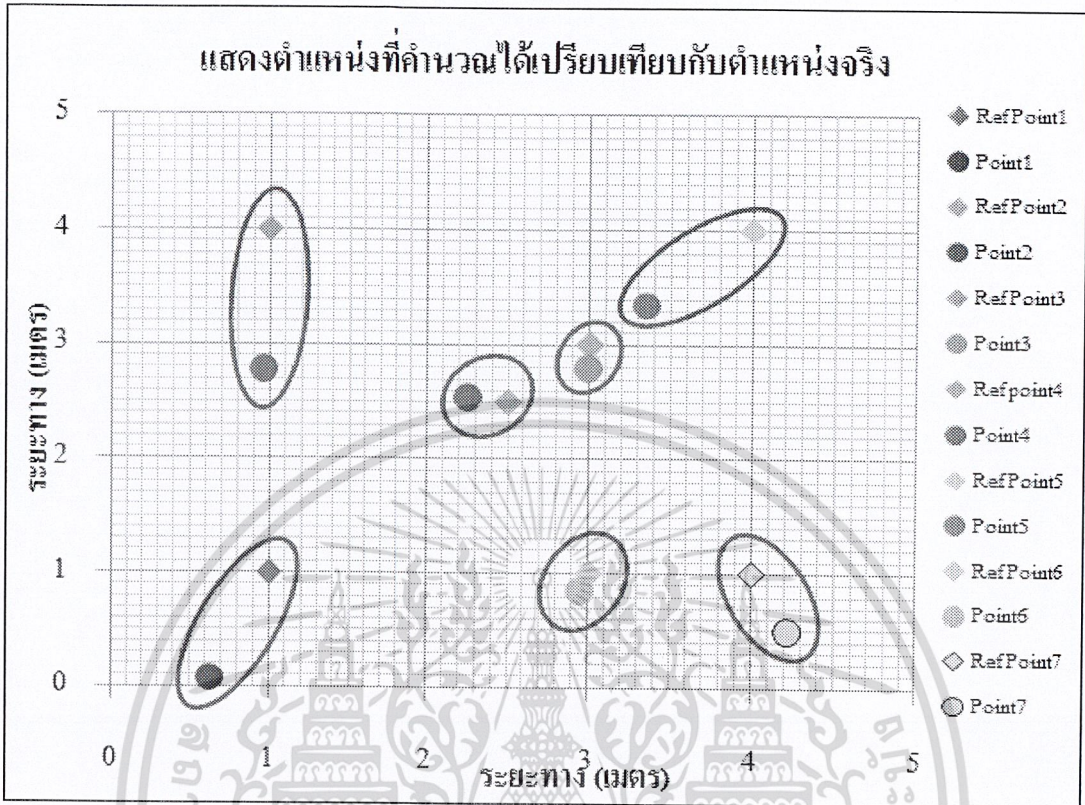
รูปที่ 4.16 แสดงการทดลองหาตำแหน่งของวัตถุ

สภาพแวดล้อม

- ภายนอกอาคาร ลานกิจกรรมบริเวณด้านข้างสนามเบดมินตัน สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (เหมือนการทดลองที่ 4.1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา⁵⁹ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง



รูปที่ 4.17 แสดงผลตำแหน่งของวัตถุที่คำนวณได้

สรุปผลการทดลอง

จากรูปที่ 4.17 จะเห็นว่าตำแหน่งที่ได้คำนวณกับตำแหน่งจริงมีความคลาดเคลื่อนพอสมควร แต่ตรงช่วงบริเวณตรงกลางของพื้นที่ตำแหน่งที่คำนวณได้กับกับตำแหน่งจริงมีความใกล้เคียงมาก

4.4 การทดลองเก็บค่า RSSI (Receive strength Signal Indicator) เพื่อเป็นฐานข้อมูลของ Fingerprint Algorithm ภายในอาคาร

จุดประสงค์

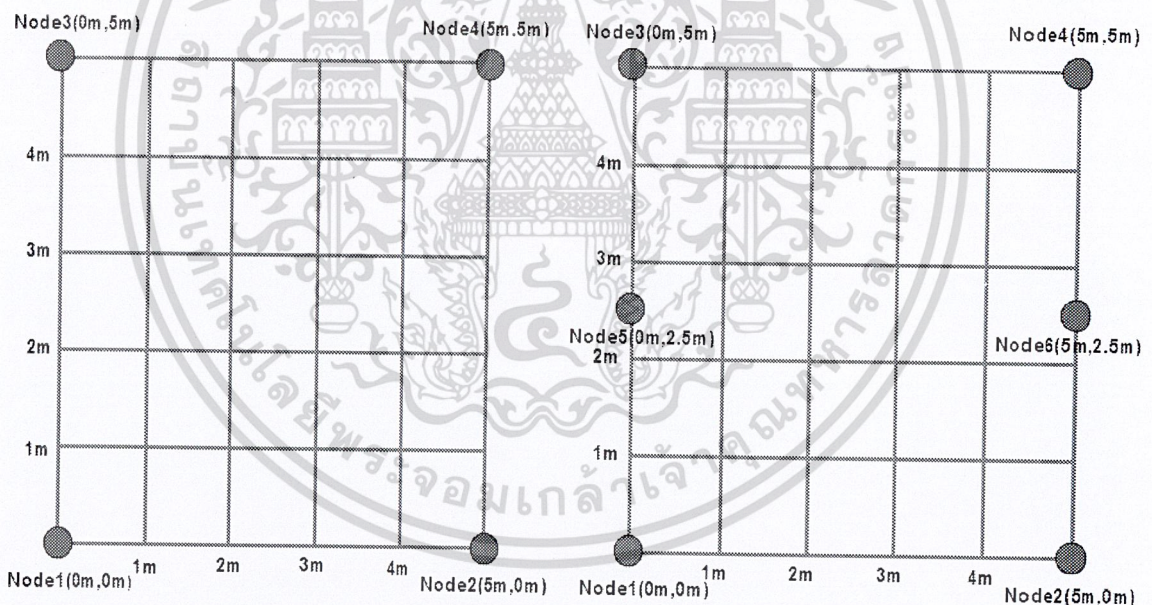
- เก็บค่า RSSI ไว้ใช้เป็นฐานข้อมูลของอัลกอริทึม Fingerprint เพื่อนำไปหาตำแหน่งของวัตถุ

อุปกรณ์

- XBee Module 7 ตัว (ตัวส่ง 1 ตัว, ตัวรับ 6 ตัว)
- คอมพิวเตอร์ 1 เครื่อง
- กล่อง ขนาด 21x29.7 ซม. 7 กล่อง
- โปรแกรมใช้ในการหาค่า RSSI

วิธีการทดลอง

- จัดเตรียมพื้นที่ขนาด 5x5 เมตร โดยใช้ตัวรับทั้ง 4 ตัว และตัวรับ 6 ตัว ตามลำดับ แสดงดังรูปที่ 4.18
- ทำการเก็บค่า RSSI ของตำแหน่งที่ (0,0),(1,0),(2,0),..., (5,5) รวมทั้งหมด 36 ตำแหน่ง โดยเก็บค่า RSSI ทั้งที่ใช้โหนดอ้างอิง 4 ตัว และ 6 ตัว ตามลำดับ
- ทำตามข้อ 2 จำนวน 10 รอบ แต่ละรอบห่างกัน 10 นาที และนำค่า RSSI ทั้ง 10 รอบของแต่ละตำแหน่งมาหาค่าเฉลี่ย แล้วทำการเก็บค่าลงในฐานข้อมูล

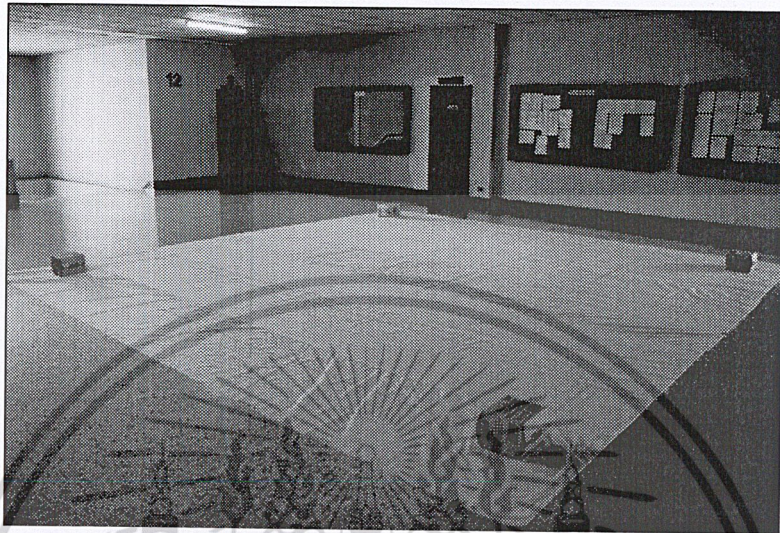


รูปที่ 4.18 ภาพจำลองการจัดพื้นที่เพื่อทำการเก็บค่า RSSI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 61 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สภาพแวดล้อม

- ภายในอาคาร บริเวณหน้าลิฟต์ ชั้น 12 ตึก E12 คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง แสดงดังรูปที่ 4.19



รูปที่ 4.19 สถานที่ในการทดลองภายในอาคาร

ผลการทดลอง

เมื่อทำการทดลองแล้วจะได้ค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 4 ตัว แสดงดังตารางที่ 4.2 และได้ค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 6 ตัว แสดงดังตารางที่ 4.3

ตารางที่ 4.2 ตารางแสดงค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 4 ตัว

ตำแหน่ง	ตัวรับที่ 1 (-dBm)	ตัวรับที่ 2 (-dBm)	ตัวรับที่ 3 (-dBm)	ตัวรับที่ 4 (-dBm)
(0m,0m)	26.0	44.9	47.0	52.0
(1m,0m)	32.5	42.8	47.7	49.9
(2m,0m)	38.3	41.0	48.5	49.7
(3m,0m)	41.3	36.9	48.3	46.8
(4m,0m)	44.3	31.4	50.9	48.9
(5m,0m)	44.2	26.0	26.0	53.2
(0m,1m)	31.2	46.6	45.5	53.2
(1m,1m)	32.3	45.8	45.4	49.3
(2m,1m)	37.0	39.0	47.9	47.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่ควรเปิดเผยให้บุคคลภายนอกได้รับทราบ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 62 จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 (ต่อ) ตารางแสดงค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 4 ตัว

ตำแหน่ง	ตัวรับที่ 1 (-dBm)	ตัวรับที่ 2 (-dBm)	ตัวรับที่ 3 (-dBm)	ตัวรับที่ 4 (-dBm)
(3m,1m)	40.4	36.1	49.4	46.8
(4m,1m)	44.9	33.6	48.5	45.3
(5m,1m)	48.9	32.1	51.4	44.8
(0m,2m)	35.1	46.4	42.9	48.8
(1m,2m)	34.4	44.3	42.7	48.1
(2m,2m)	39.6	41.1	45.2	45.9
(3m,2m)	40.8	39.5	46.8	43.0
(4m,2m)	44.8	35.2	44.8	42.0
(5m,2m)	49.2	36.3	51.3	39.2
(0m,3m)	39.3	48.2	37.5	47.4
(1m,3m)	38.6	45.3	38.4	45.2
(2m,3m)	40.2	44.3	41.2	44.1
(3m,3m)	41.5	41.2	43.2	41.1
(4m,3m)	46.4	40.3	45.6	37.5
(5m,3m)	50.3	40.3	48.8	36.5
(0m,4m)	41.2	50.4	32.6	48.8
(1m,4m)	42.0	49.3	35.8	45.1
(2m,4m)	42.0	45.0	37.4	42.7
(3m,4m)	44.8	44.5	40.1	38.3
(4m,4m)	47.4	44.5	44.9	35.2
(5m,4m)	48.9	42.7	47.7	31.7
(0m,5m)	47.4	48.7	26.0	48.3
(1m,5m)	46.0	49.2	31.1	44.2
(2m,5m)	46.7	48.6	36.5	41.7
(3m,5m)	48.5	46.9	41.3	37.5
(4m,5m)	48.6	47.6	41.7	33.5
(5m,5m)	49.4	49.0	45.3	26.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 63 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 6 ตัว

ตำแหน่ง	ตัวรับที่ 1 (-dBm)	ตัวรับที่ 2 (-dBm)	ตัวรับที่ 3 (-dBm)	ตัวรับที่ 4 (-dBm)	ตัวรับที่ 5 (-dBm)	ตัวรับที่ 6 (-dBm)
(0m,0m)	26.0	46.1	46.2	55.8	37.3	53.9
(1m,0m)	31.4	41.8	47.6	51.6	40.1	46.9
(2m,0m)	36.4	39.1	48.3	50.1	41.2	43.2
(3m,0m)	40.8	36.3	48.0	47.2	46.0	40.2
(4m,0m)	43.0	31.1	53.5	51.1	49.1	38.6
(5m,0m)	42.7	26.0	53.7	43.9	54.8	34.7
(0m,1m)	31.8	45.7	45.7	53.7	36.8	49.3
(1m,1m)	33.4	45.4	45.0	51	36.9	44.7
(2m,1m)	36.2	39.1	47.7	48.6	40.4	41.1
(3m,1m)	39.9	36.4	48.9	46.2	45.3	37.4
(4m,1m)	45.0	33.6	49.7	45.6	46.9	33.9
(5m,1m)	47.4	31.5	50.6	44.2	50.7	32.2
(0m,2m)	35.1	46.7	41.6	49.6	30.2	46.2
(1m,2m)	34.2	44.8	42.3	48.4	34.9	43.8
(2m,2m)	41.0	42.1	44.2	44.9	38.4	40.4
(3m,2m)	41.2	39.7	46.3	43.0	41.7	35.7
(4m,2m)	44.9	35.1	47.6	41.9	44.8	31.8
(5m,2m)	48.4	34.6	50.6	40.9	47.9	28.1
(0m,3m)	39.6	47.8	37.4	49.1	29.0	46.8
(1m,3m)	39.1	44.6	38.5	45.8	31.8	43.3
(2m,3m)	41.2	44.0	40.7	44.8	35.9	41.4
(3m,3m)	44.1	41.5	43.7	41.8	40.1	36.9
(4m,3m)	46.8	39.5	45.8	38.5	42.5	32.6
(5m,3m)	49.7	39.2	48.4	36.8	44.2	28.8
(0m,4m)	42.4	50.5	32.3	50.4	33.9	48.0
(1m,4m)	43.3	48.1	35.3	45.5	33.9	44.5
(2m,4m)	43.9	45.1	37.2	42.2	38.1	42.5
(3m,4m)	45.4	44.4	40.7	39.0	40.2	39.0

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ไปเผยแพร่ขึ้นต้นในการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 64 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) ค่า RSSI ของแต่ละตำแหน่งที่ใช้ตัวรับ 6 ตัว

ตำแหน่ง	ตัวรับที่ 1 (-dBm)	ตัวรับที่ 2 (-dBm)	ตัวรับที่ 3 (-dBm)	ตัวรับที่ 4 (-dBm)	ตัวรับที่ 5 (-dBm)	ตัวรับที่ 6 (-dBm)
(4m,4m)	48.1	44.0	44.5	36.2	42.4	34.7
(5m,4m)	50.9	44.2	49.0	31.9	47.7	33.6
(0m,5m)	47.2	50.8	26.0	49.0	37.8	49.1
(1m,5m)	47.3	49.9	31.3	44.2	36.6	47.4
(2m,5m)	48.3	49.7	36.9	42.1	40.1	43.1
(3m,5m)	47.5	45.0	41.2	38.8	42.2	41.7
(4m,5m)	49.3	47.1	42.9	33.8	44.7	37.1
(5m,5m)	50.7	46.7	45.8	27.0	47.1	37.5

4.5 การทดลองเก็บค่า RSSI (Receive strength Signal Indicator) ณ ตำแหน่งต่างๆ ภายในอาคาร

จุดประสงค์

- เก็บค่า RSSI ณ ตำแหน่งต่างๆภายในพื้นที่การทดลอง เพื่อนำไปกำหนดหาตำแหน่งของอัลกอริทึมการหาตำแหน่งต่างๆ

อุปกรณ์

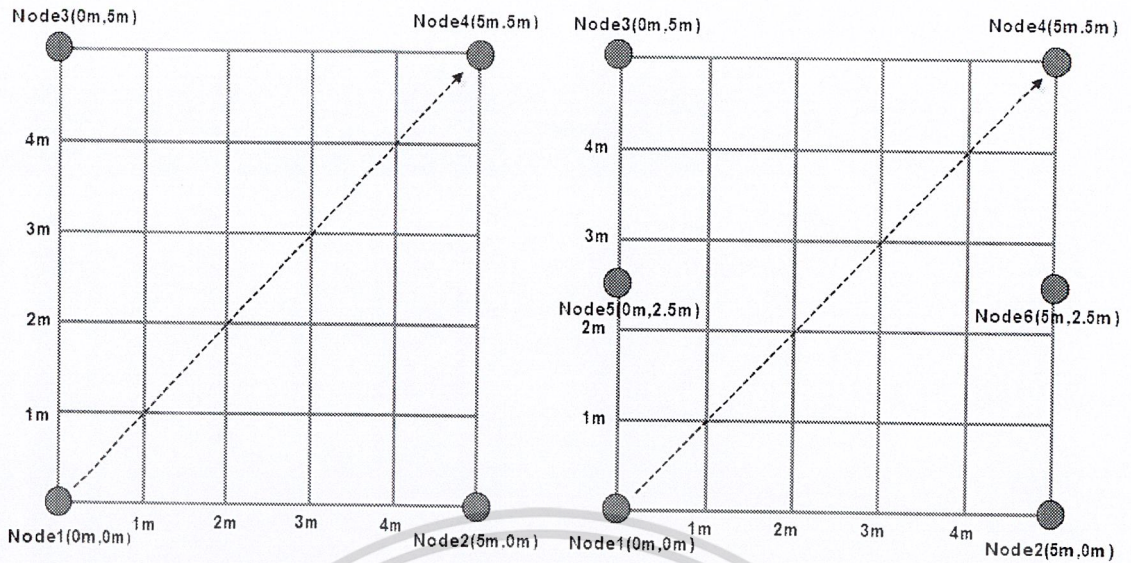
- XBee Module 7 ตัว (ตัวส่ง 1 ตัว, ตัวรับ 6 ตัว)
- คอมพิวเตอร์ 1 เครื่อง
- กล่อง ขนาด 21×29.7 ซม. 7 กล่อง
- โปรแกรมใช้ในการหาค่า RSSI

วิธีการทดลอง

- จัดเตรียมพื้นที่ขนาด 5×5 เมตร โดยใช้ตัวรับทั้ง 4 ตัว และ ตัวรับ 6 ตัว ตามลำดับ (เหมือนการทดลองที่ 4.4)
- ทำการเก็บค่า RSSI แต่ละตำแหน่งตามแนวทแยงของพื้นที่ทดลอง จากมุมด้านล่างซ้ายมือ(จุด0m,0m) ถึงมุมด้านบนขวามือ(จุด 5m,5m) แสดงดังรูปที่ 4.20 โดยมีความห่างแต่ละตำแหน่งเป็น (0m,0m), (0.33m,0.33m), (0.66m,0.66m), (1m,1m), ... , (5m,5m) ทำแบบนี้ทั้งสิ้น 5 รอบ แต่ละรอบห่างกัน 10 นาที โดยเก็บค่า RSSI ทั้งที่ใช้ตัวรับ 4 ตัว และ 6 ตัว ตามลำดับ

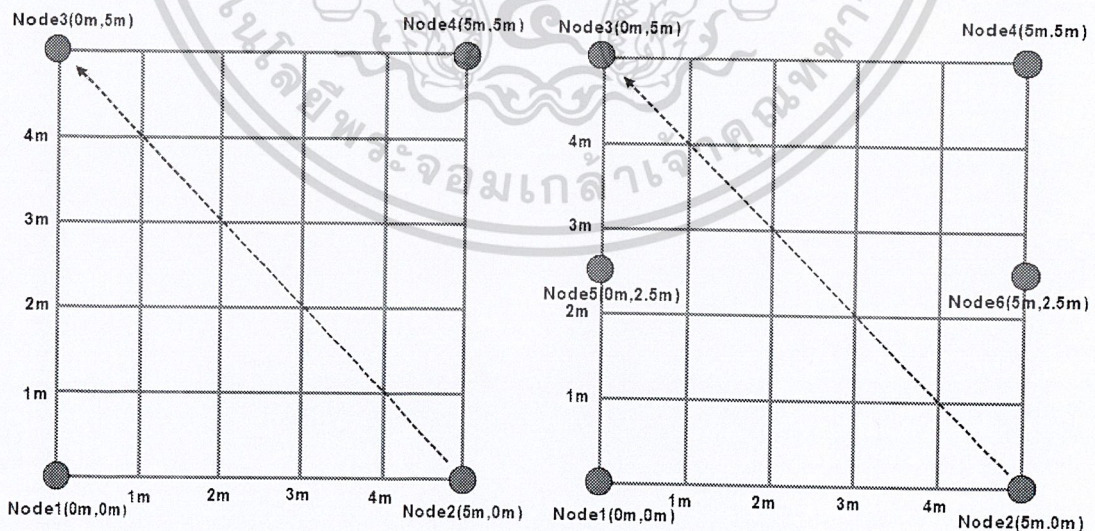
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 65 จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 ภาพจำลองการเก็บค่า RSSI ตามแนวทแยงของพื้นที่ทดลอง จากมุมด้านล่างซ้ายมือถึงมุมด้านบนขวามือ

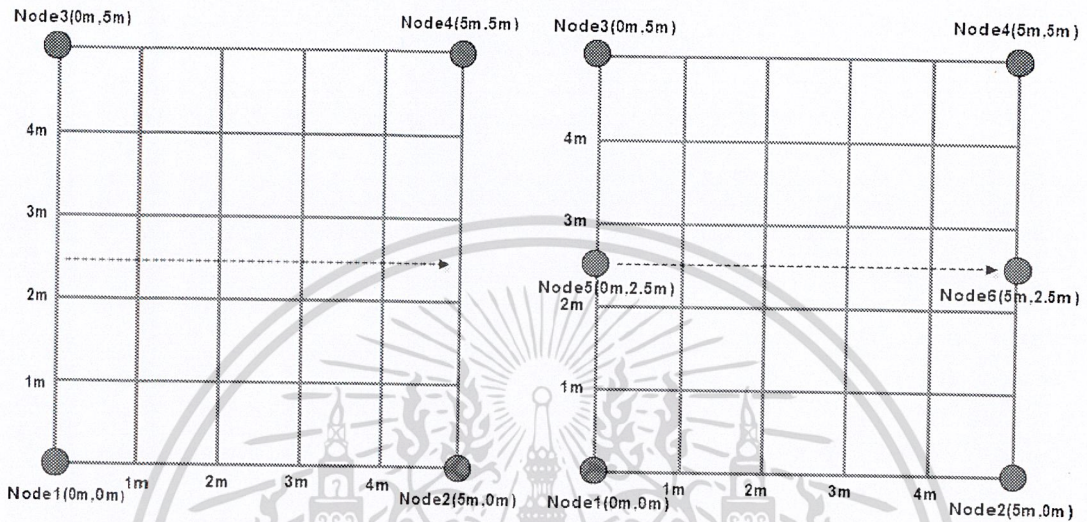
- ทำการเก็บค่า RSSI แต่ละตำแหน่งตามแนวทแยงของพื้นที่ทดลอง จากมุมด้านล่างขวามือ(จุด 5m,0m) ถึงมุมด้านบนซ้ายมือ(จุด 0m,5m) แสดงดังรูปที่ 4.21 โดยมีความห่างแต่ละตำแหน่งเป็น (5m,0m), (4.66m,0.33m), (4.33m,0.66m), (4m,1m), ... , (0m,5m) ทำแบบนี้ทั้งสิ้น 5 รอบ แต่ละรอบห่างกัน 10 นาที โดยเก็บค่า RSSI ทั้งที่ใช้ตัวรับ 4 ตัว และ 6 ตัว ตามลำดับ



รูปที่ 4.21 ภาพจำลองการเก็บค่า RSSI ตามแนวทแยงของพื้นที่ทดลอง จากมุมด้านล่างขวามือถึงมุมด้านบนซ้ายมือ

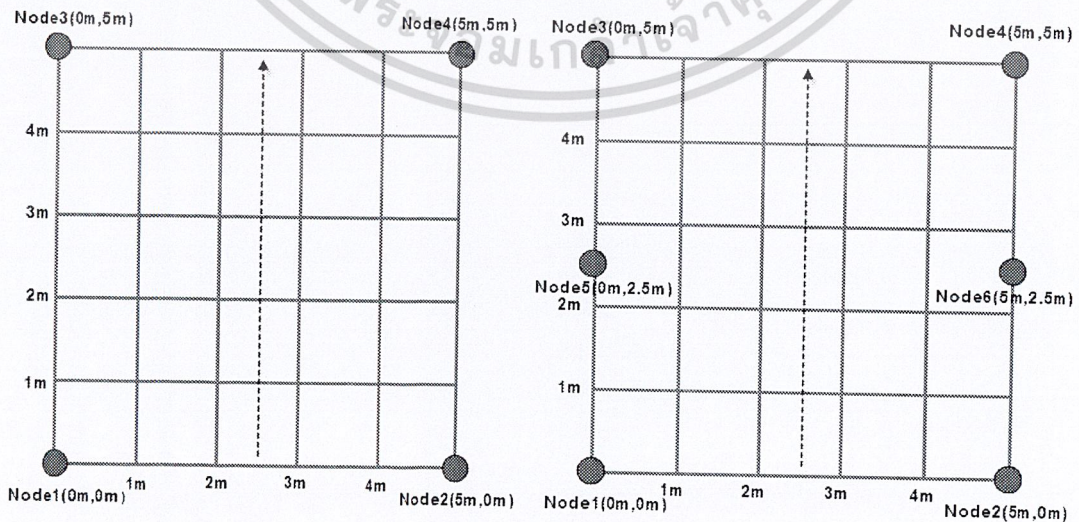
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 66 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทำการเก็บค่า RSSI แต่ละตำแหน่งตามแนวนอนกึ่งกลางของพื้นที่ทดลอง จากตำแหน่ง (0m,2.5m), (0.25m,2.5m), (0.5m,2.5m), (0.75m,2.5m), (1m,2.5m), ..., (5m,2.5m) แสดงดังรูปที่ 4.22 ทำแบบนี้ทั้งสิ้น 5 รอบ แต่ละรอบห่างกัน 10 นาที โดยเก็บค่า RSSI ทั้งที่ใช้ตัวรับ 4 ตัว และ 6 ตัว ตามลำดับ



รูปที่ 4.22 ภาพจำลองการเก็บค่า RSSI ตามแนวนอนกึ่งกลางของพื้นที่ทดลอง

- ทำการเก็บค่า RSSI แต่ละตำแหน่งตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง จากตำแหน่ง (2.5m,0m), (2.5m,0.25m), (2.5m,0.5m), (2.5m,0.75m), (2.5m,1m), ..., (2.5m,5m) แสดงดังรูปที่ 4.23 ทำแบบนี้ทั้งสิ้น 5 รอบ แต่ละรอบห่างกัน 10 นาที โดยเก็บค่า RSSI ทั้งที่ใช้ตัวรับ 4 ตัว และ 6 ตัว ตามลำดับ

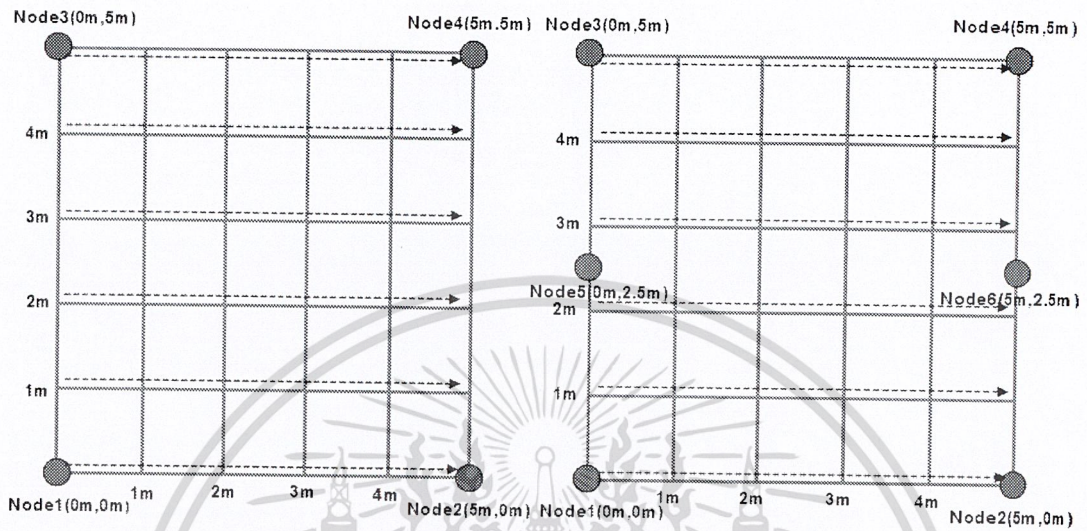


รูปที่ 4.23 ภาพจำลองการเก็บค่า RSSI ตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้เห็นไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 67 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทำการเก็บค่า RSSI ของตำแหน่งที่ $(0,0), (1,0), (2,0), \dots, (5,5)$ ของพื้นที่ทดลอง รวมทั้งหมด 36 ตำแหน่ง แสดงดังรูป 4.24 ทำแบบนี้ทั้งสิ้น 5 รอบ แต่ละรอบห่างกัน 10 นาที โดยเก็บค่า RSSI ทั้งที่ใช้ตัวรับ 4 ตัว และ 6 ตัว ตามลำดับ



รูปที่ 4.24 ภาพจำลองการเก็บค่า RSSI ทั้งหมด 36 ตำแหน่ง ของพื้นที่ทดลอง

สภาพแวดล้อม

- ภายในอาคาร บริเวณหน้าลิฟต์ ชั้น 12 ตึก E12 คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (เหมือนการทดลองที่ 4.4)

ผลการทดลอง

เมื่อทำการทดลองแล้วจะได้ค่า RSSI ของแต่ละตำแหน่งทั้งหมด 5 รูปแบบ แต่ละรูปแบบมีทั้งหมด 5 ครั้ง ที่ใช้ตัวรับ 4 ตัว และ 6 ตัว โดยเก็บค่า RSSI ในลักษณะเดียวกับตารางที่ 4.2 และ ตารางที่ 4.3

4.6 การทดลองหาตำแหน่งของวัตถุ และเปรียบเทียบค่าความผิดพลาดของแต่ละอัลกอริทึม

จุดประสงค์

- หาคำแหน่งของวัตถุ โดยใช้ Maximum Likelihood, Min Max และ Fingerprint Algorithm โดยใช้ค่า RSSI จากการทดลองที่ 4.5 และ 4.5 และเปรียบเทียบค่าความผิดพลาดของแต่ละอัลกอริทึม

สมมุติฐาน

- ค่าความผิดพลาดของตำแหน่งของวัตถุที่ใช้ Fingerprint Algorithm จะมีค่าน้อยที่สุด รองลงมาคือ Min-Max Algorithm และ Maximum Likelihood Algorithm จะมีค่ามากที่สุด

วิธีการทดลอง

● Maximum Likelihood Algorithm

- นำค่า RSSI ของแต่ละตำแหน่งที่เก็บไว้จากการทดลองที่ 4.5 ทั้ง 5 รูปแบบทั้งที่ใช้ตัวรับ 4 ตัวและ 6 ตัว นำมาคำนวณหาค่า d จากสมการ (2.1)

$$RSSI = -10n \log_{10}(d) + A$$

ให้ค่า $A = 29$ และค่า $n = 2.9$ โดยที่ค่า A และ ค่า n หาได้จากการทดลองเหมือนการทดลองที่ 4.2 แต่ทำภายในอาคาร สภาพแวดล้อมเหมือนการทดลองที่ 4.4

- นำค่า d ที่ได้มาคำนวณตามวิธีของ Maximum Likelihood Algorithm จะได้เป็นตำแหน่งของวัตถุที่ต้องการ

● Min-Max Algorithm

- ทำเหมือนวิธีของ Maximum Likelihood Algorithm แต่นำค่า d ที่ได้มาคำนวณตามวิธีของ Min-Max Algorithm จะได้เป็นตำแหน่งของวัตถุที่ต้องการ

● Fingerprint Algorithm

■ แบบ Nearest

- นำค่า RSSI ของแต่ละตำแหน่งที่เก็บไว้จากการทดลองที่ 4.5 ทั้ง 5 รูปแบบ นำมาเปรียบเทียบกับค่า RSSI ของแต่ละตำแหน่งที่เก็บไว้จากการทดลองที่ 4.4 เมื่อคำนวณค่าความผิดพลาดแล้ว แล้วค่าที่น้อยที่สุดอยู่ที่ตำแหน่งใดของฐานข้อมูล จะได้ตำแหน่งของวัตถุที่ต้องการ

■ แบบ 3-Nearest

- ทำเหมือนวิธี Nearest แต่นำค่าความผิดพลาดที่น้อยที่สุด 3 ค่า ว่าตรงกับตำแหน่งใดในฐานข้อมูล จะได้ตำแหน่งของวัตถุมา 3 ตำแหน่ง แล้วนำมาหาค่าเฉลี่ย จะได้ตำแหน่งของวัตถุที่ต้องการ

- หาค่าความผิดพลาดของตำแหน่งวัตถุที่คำนวณได้กับตำแหน่งจริงของวัตถุของแต่ละอัลกอริทึม แล้วนำมาเปรียบเทียบแต่ละอัลกอริทึม

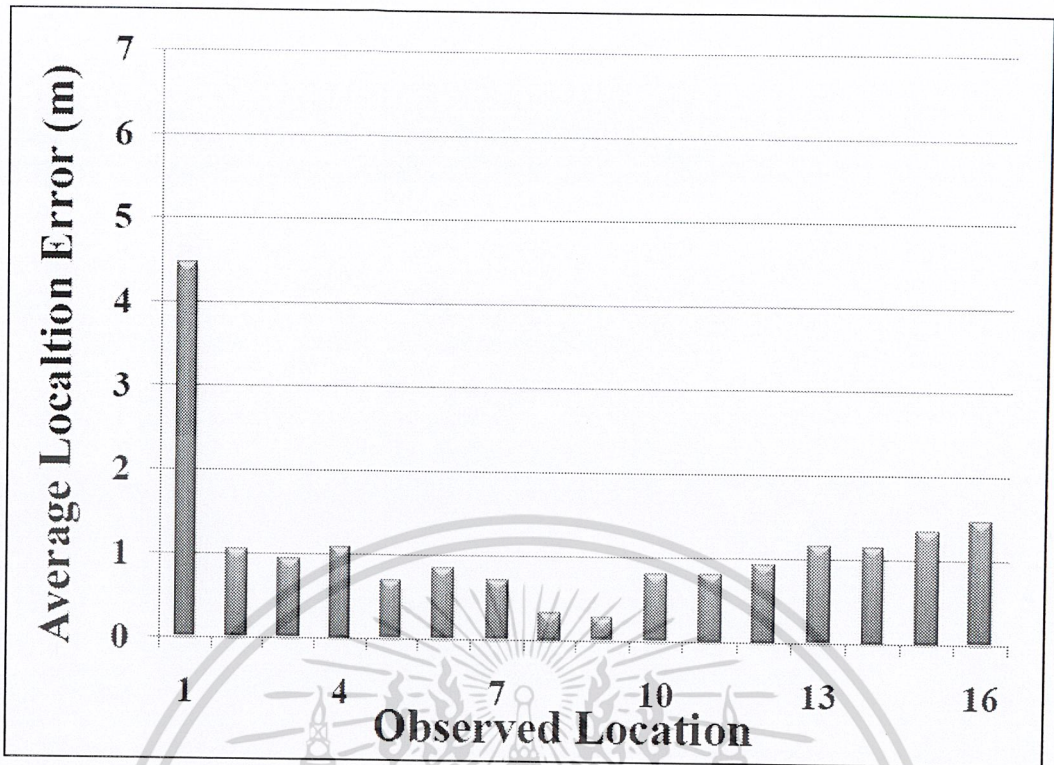
สภาพแวดล้อม

- ภายในอาคารบริเวณหน้าลิฟต์ ชั้น 12 ตึก E12 คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (เหมือนการทดลองที่ 4.4)

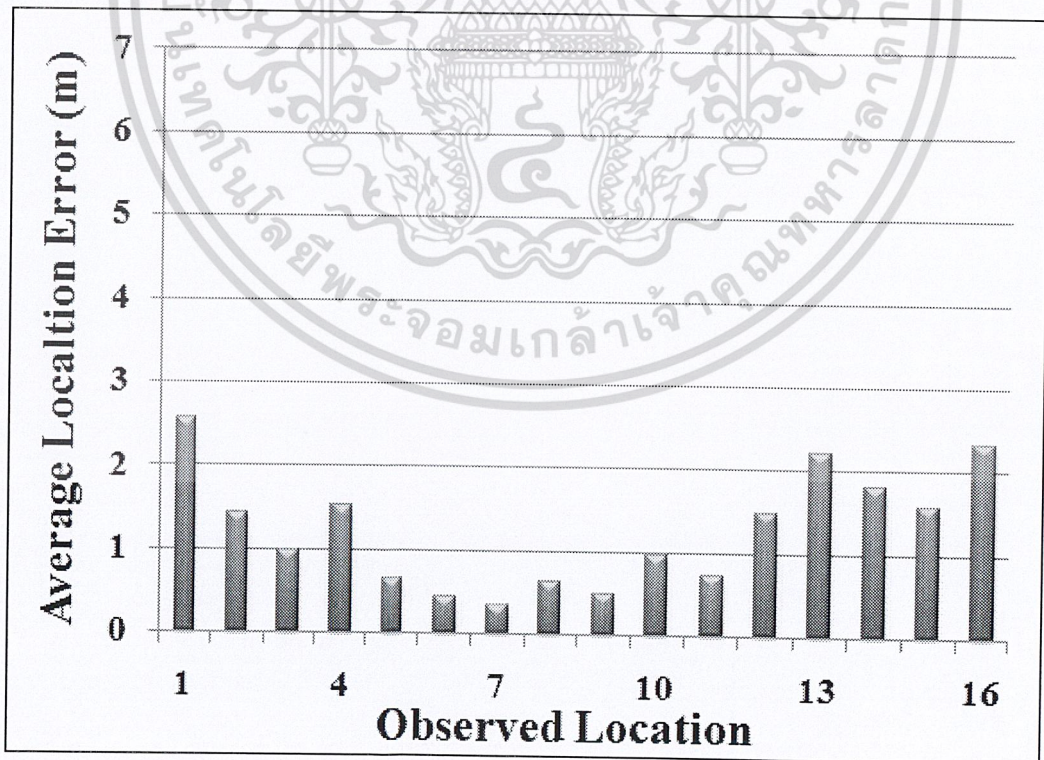
ผลการทดลอง



รูปที่ 4.25 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



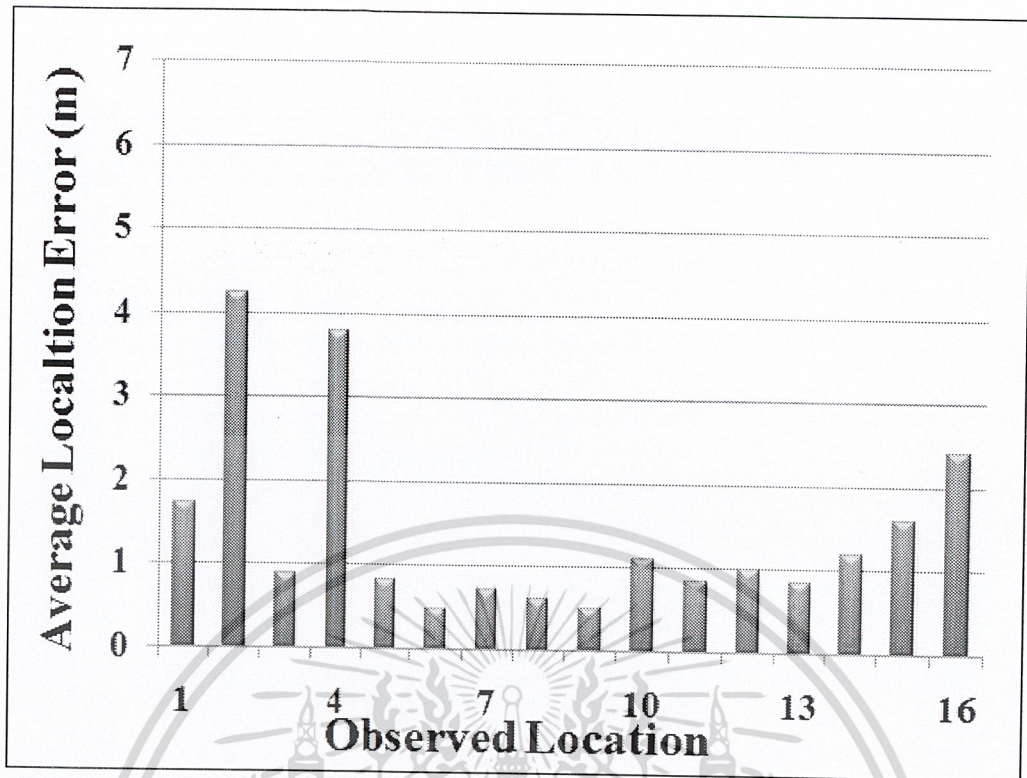
รูปที่ 4.26 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



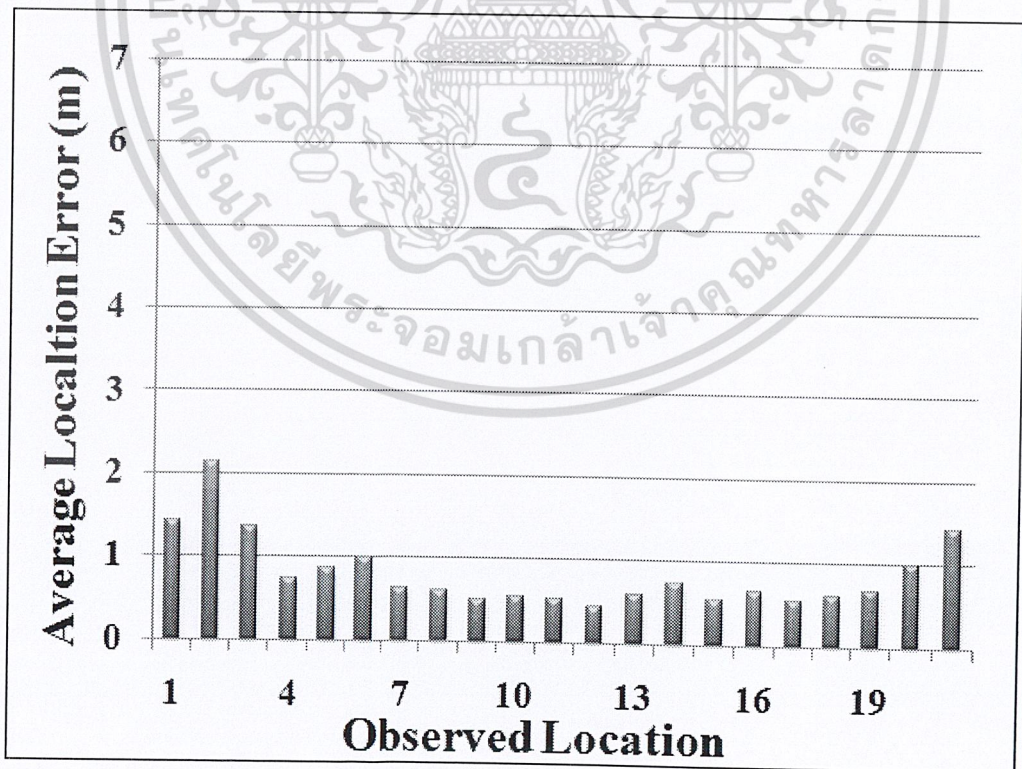
รูปที่ 4.27 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นปะไรใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา 71 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

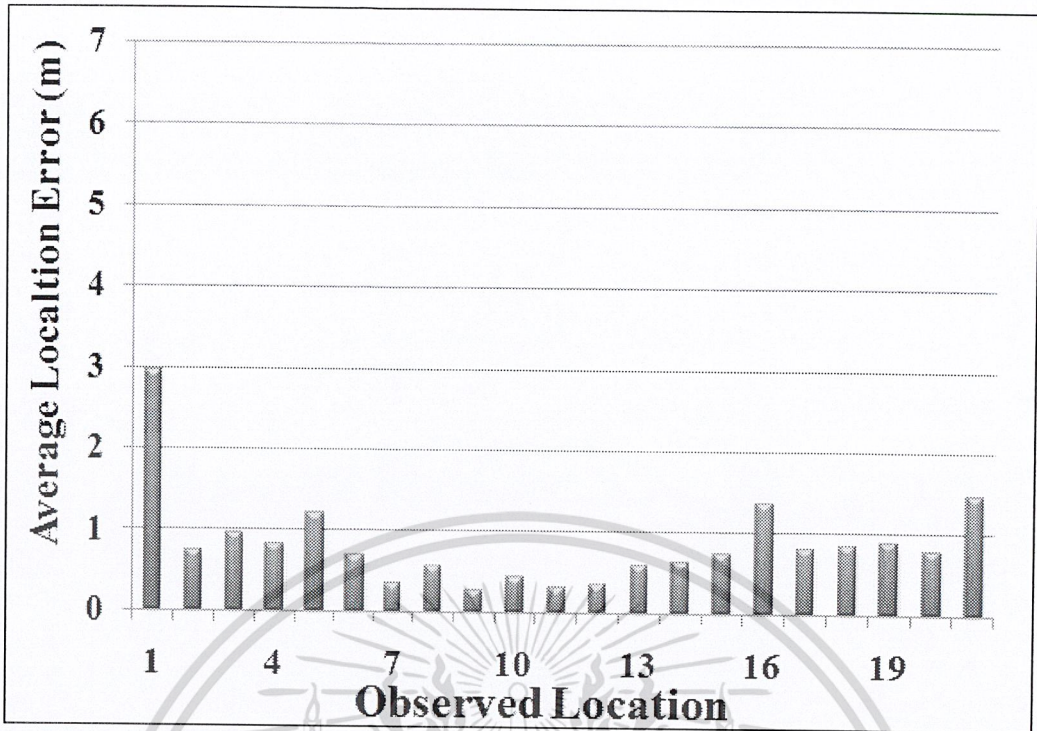


รูปที่ 4.28 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

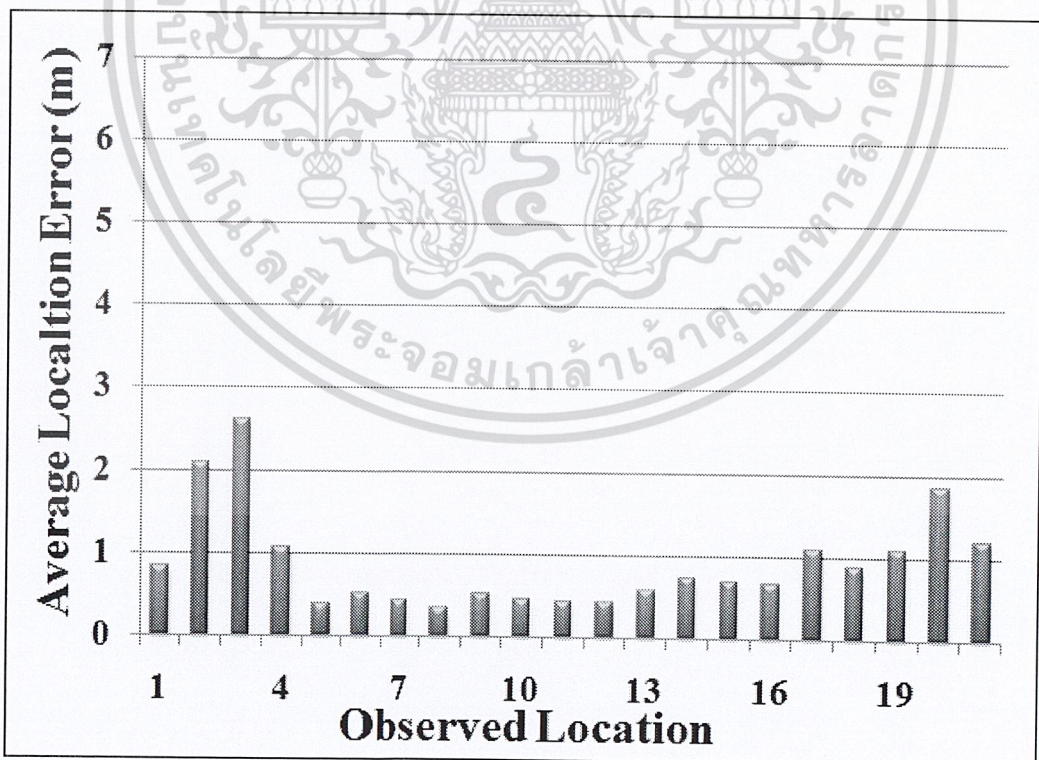


รูปที่ 4.29 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา 72 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

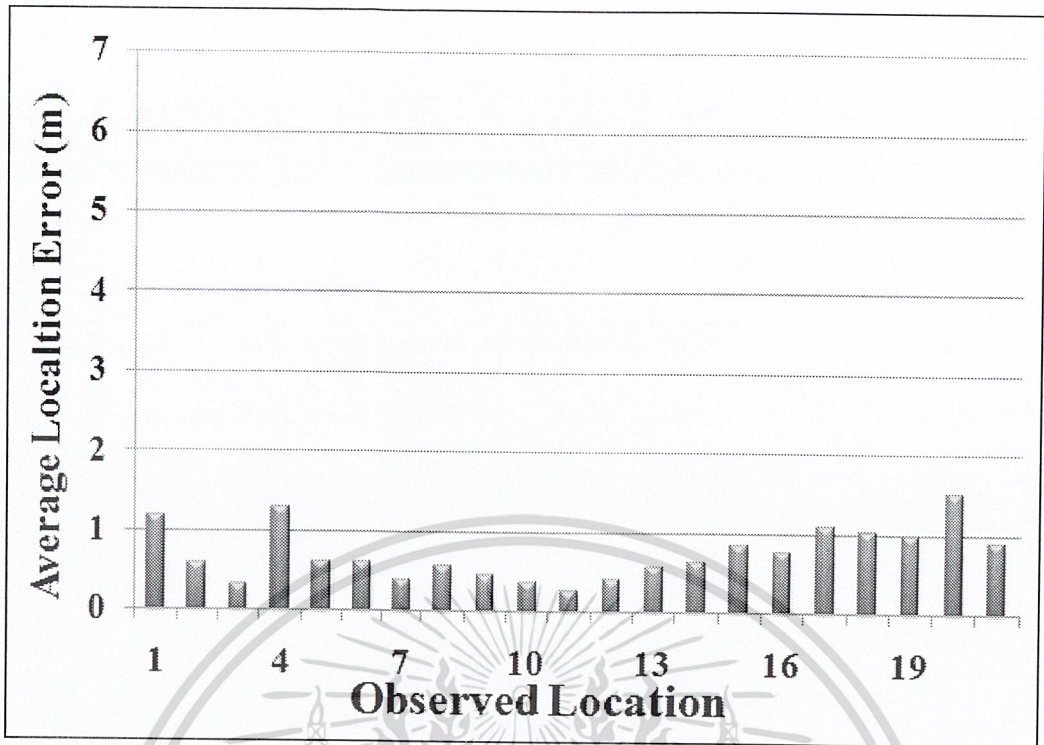


รูปที่ 4.30 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

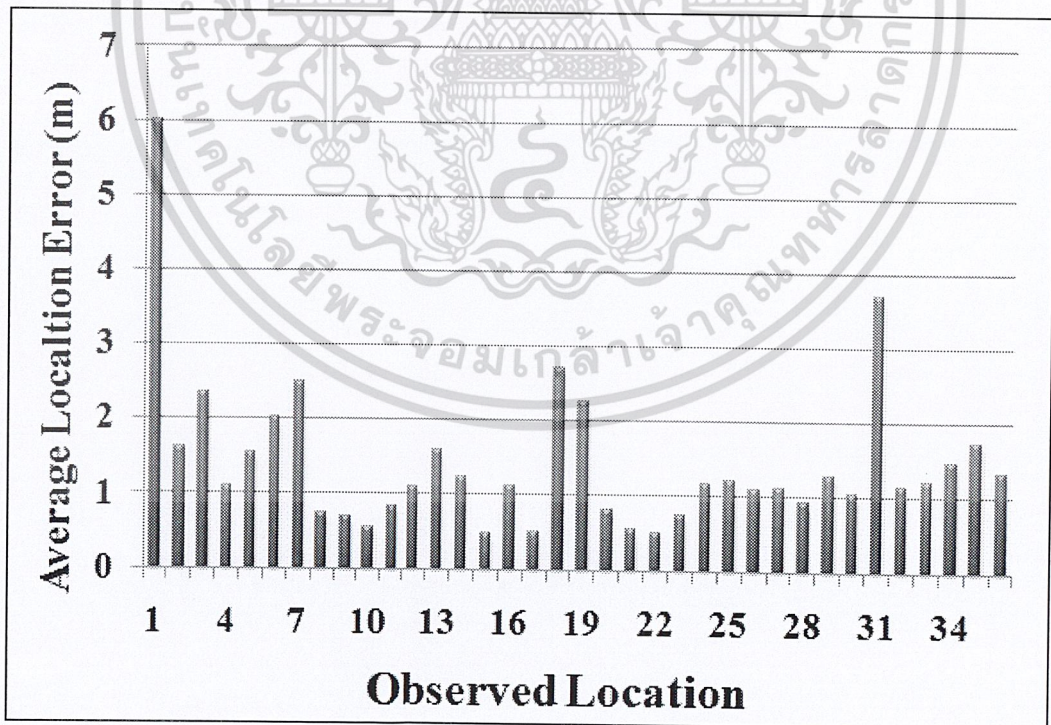


รูปที่ 4.31 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 73 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

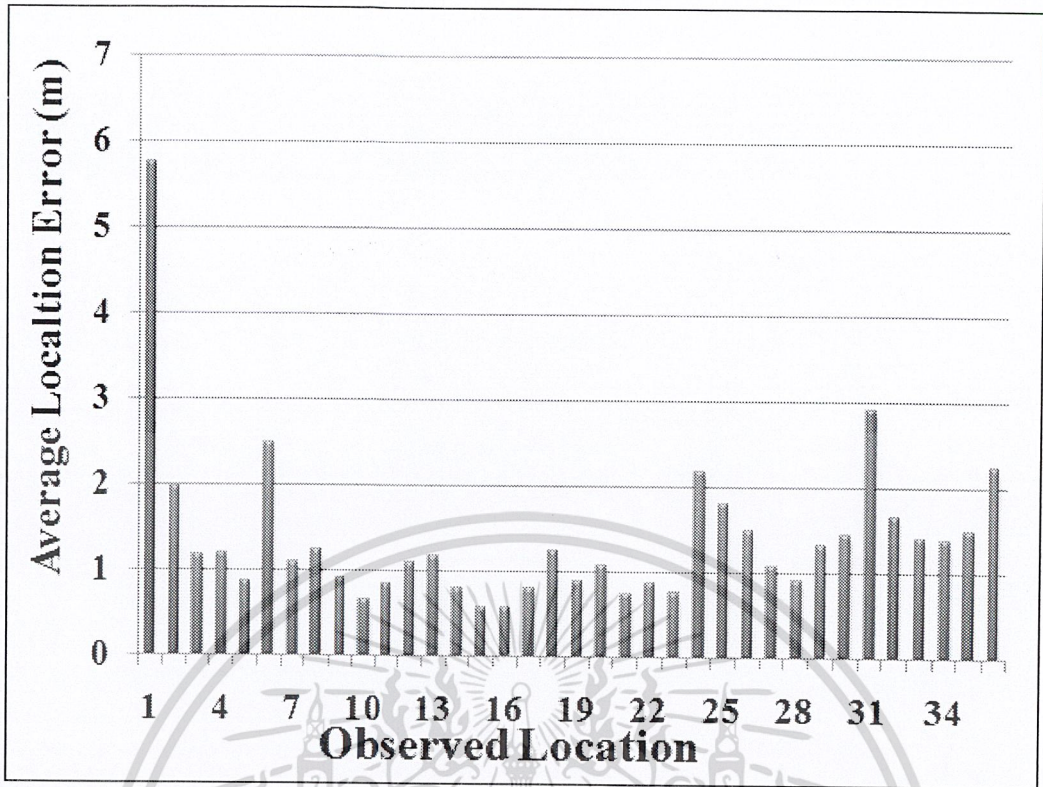


รูปที่ 4.32 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

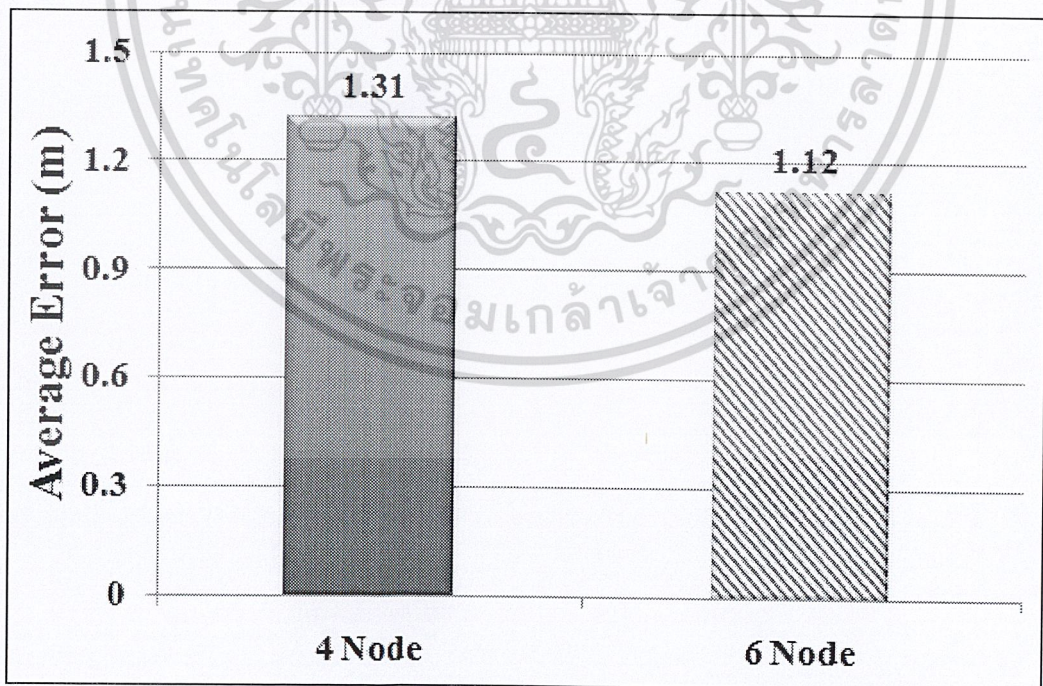


รูปที่ 4.33 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 74 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

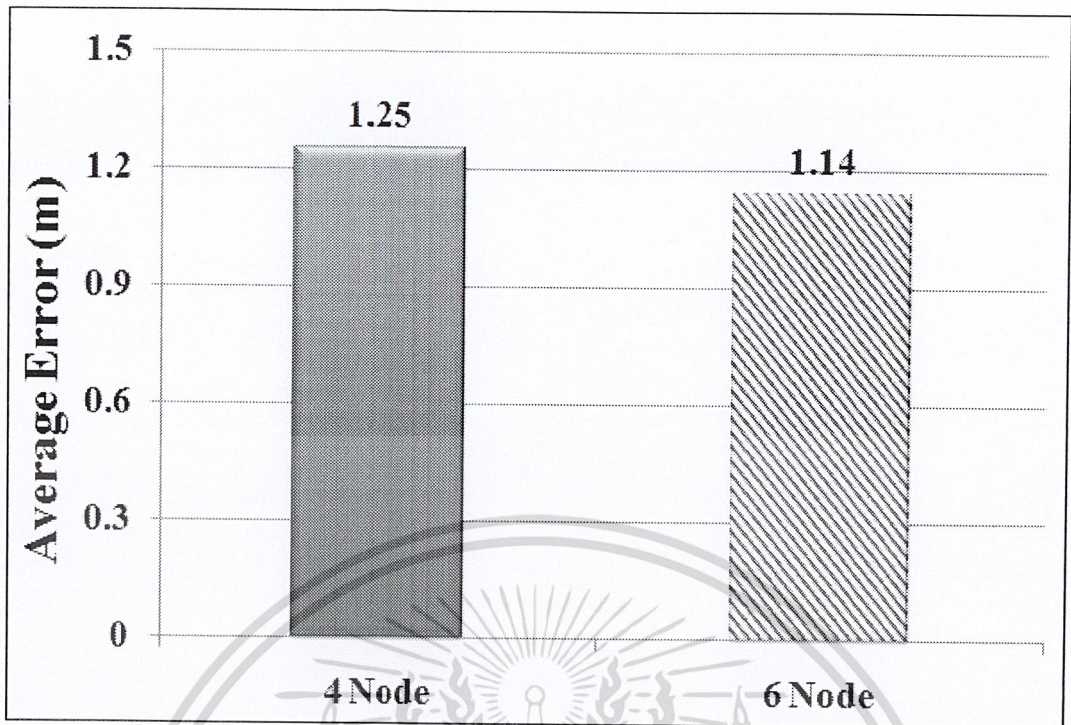


รูปที่ 4.34 ค่าความผิดพลาดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

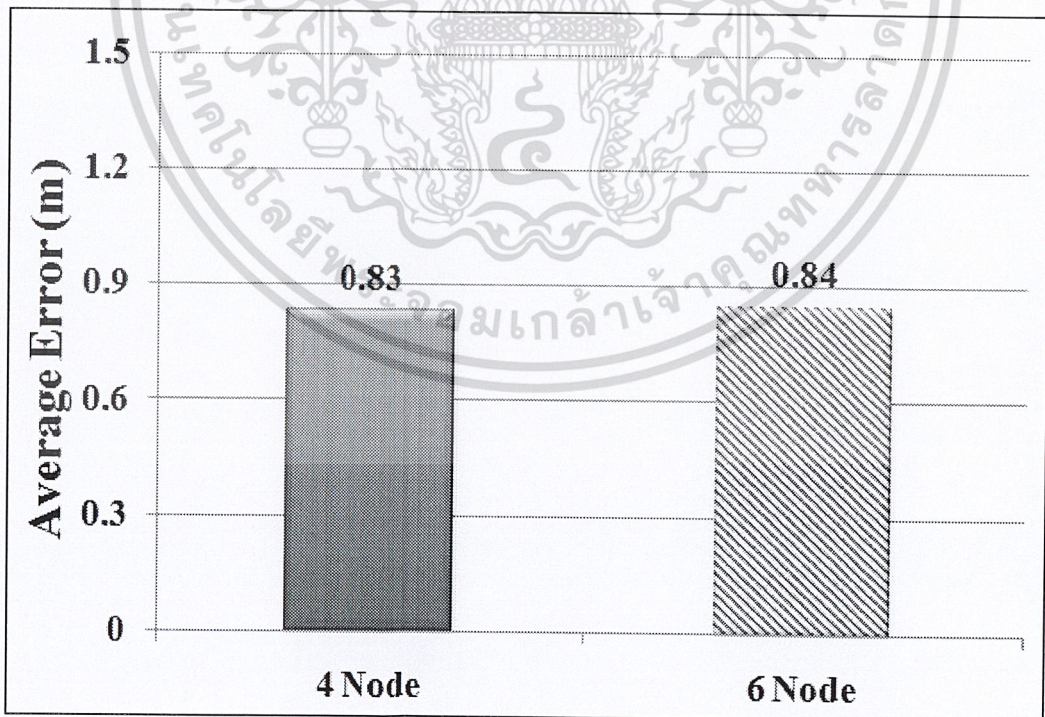


รูปที่ 4.35 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่เผยแพร่ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 75 จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

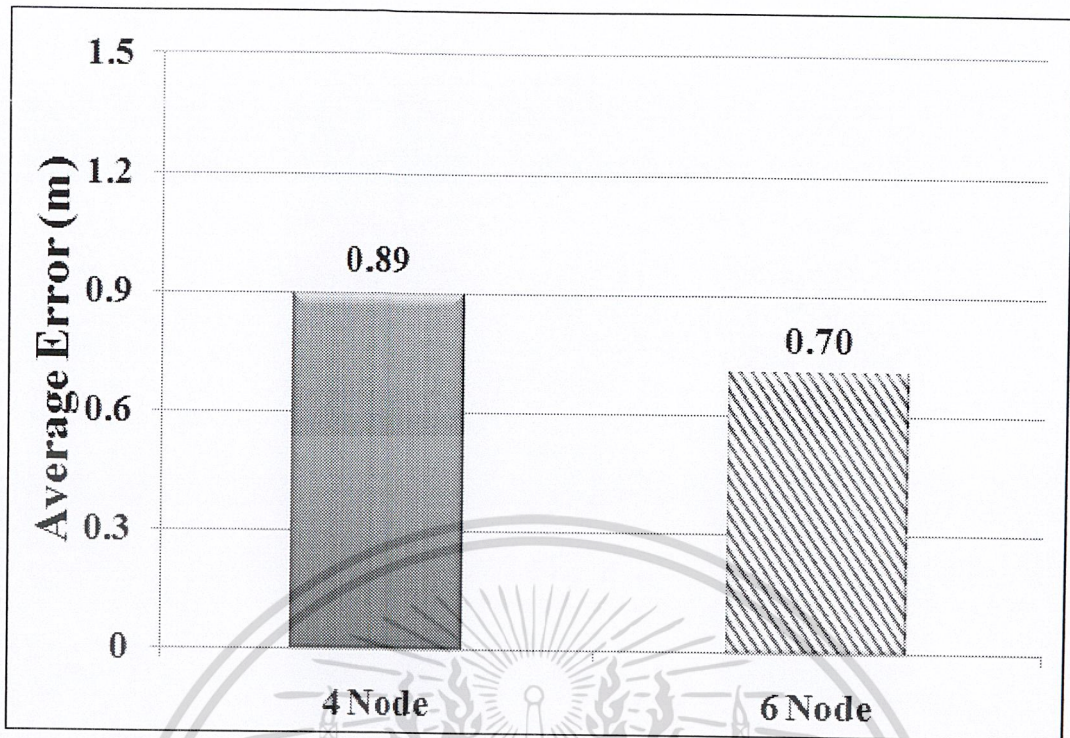


รูปที่ 4.36 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

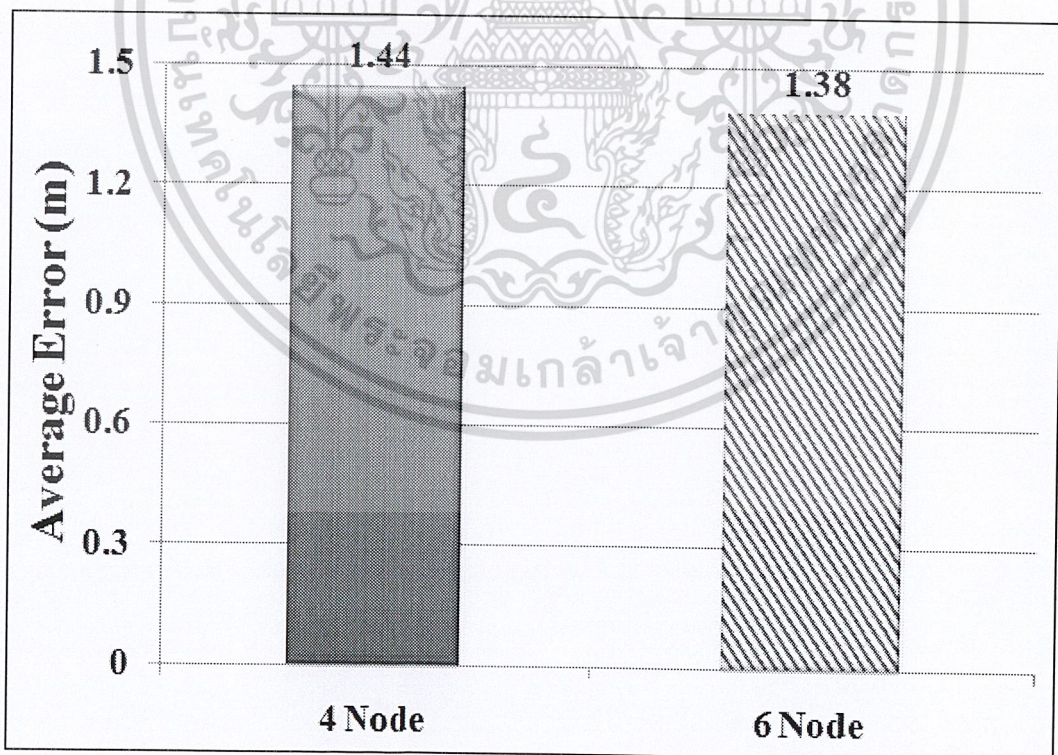


รูปที่ 4.37 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตามแนวอนกึ่งกลางของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา 76. ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

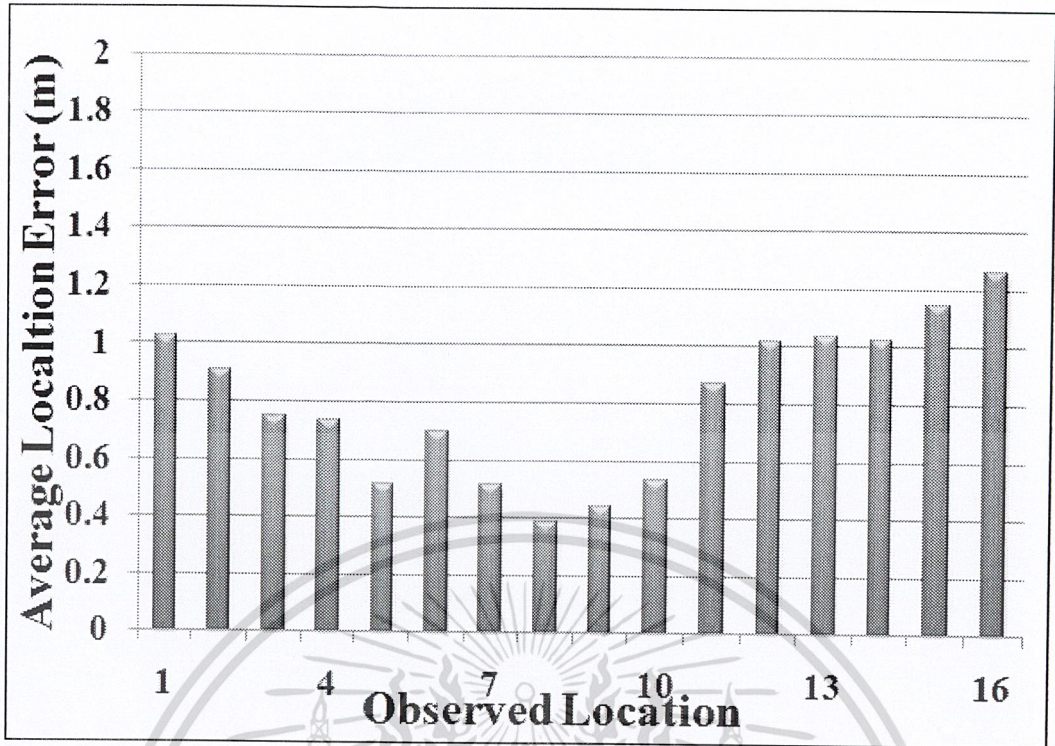


รูปที่ 4.38 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตามแนวตั้งกึ่งกลางของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

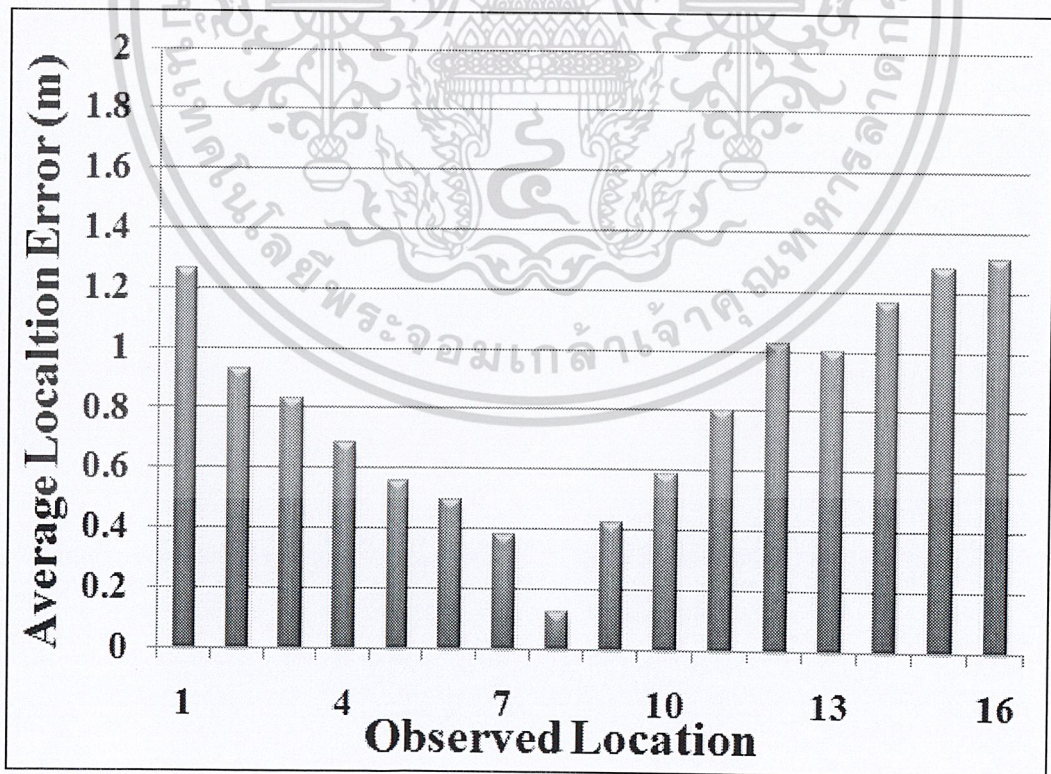


รูปที่ 4.39 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 77 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



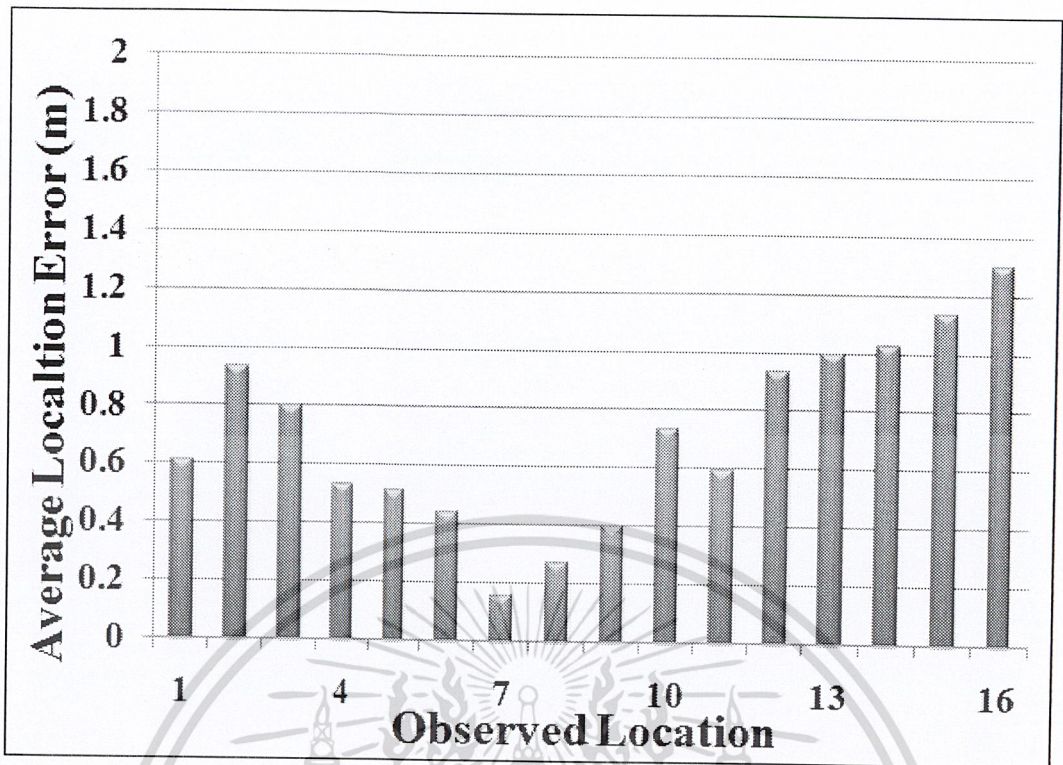
รูปที่ 4.40 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



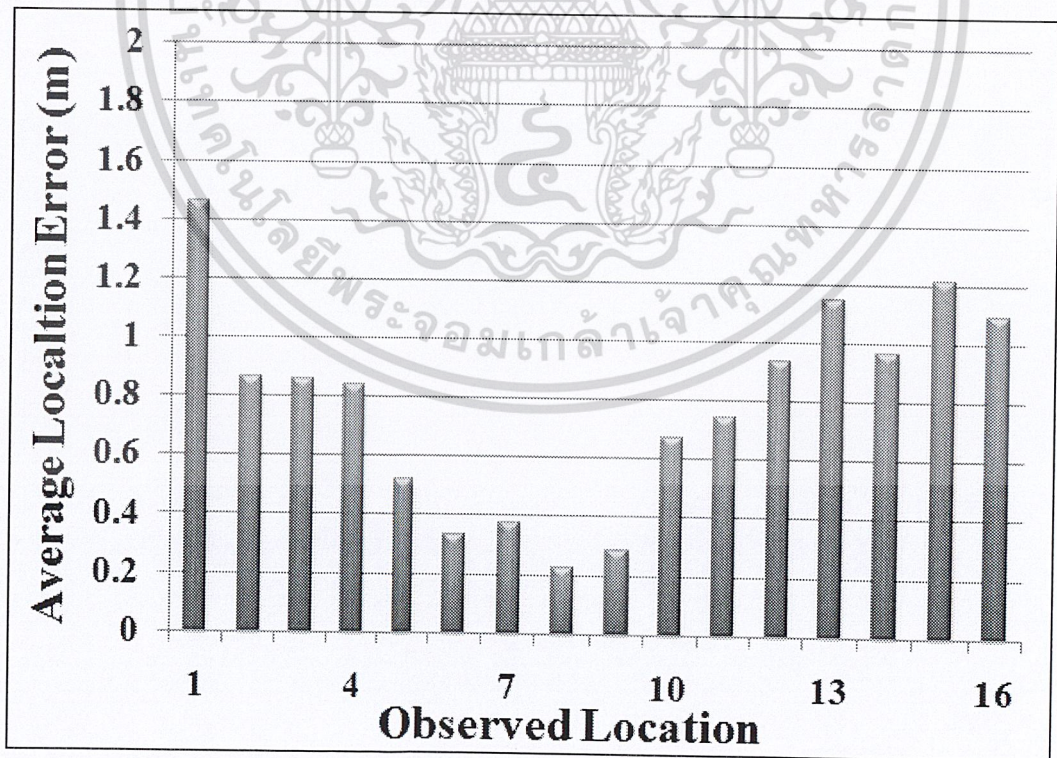
รูปที่ 4.41 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกิจกรรมเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

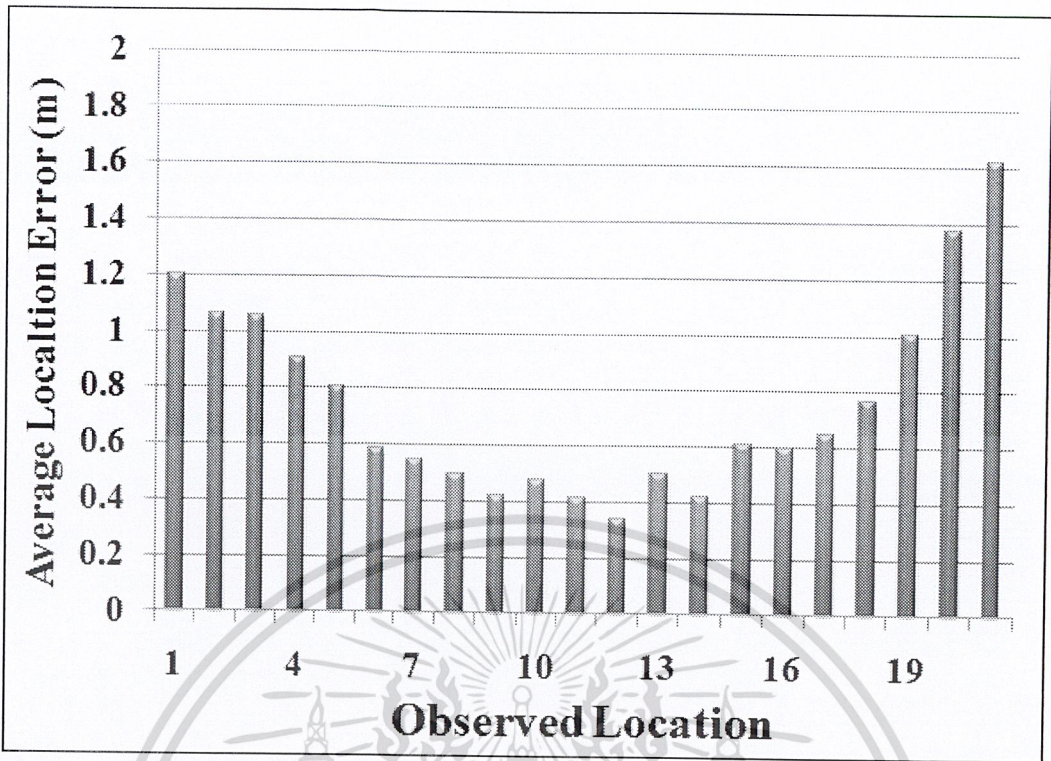
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 78 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



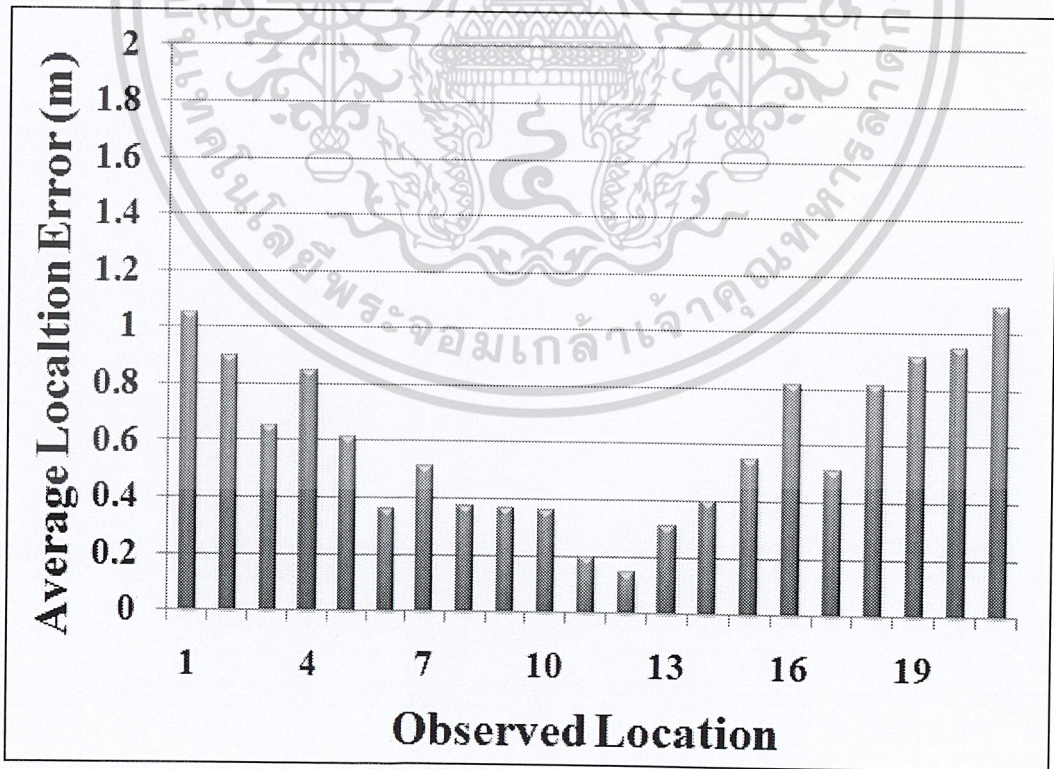
รูปที่ 4.42 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



รูปที่ 4.43 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 79. และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

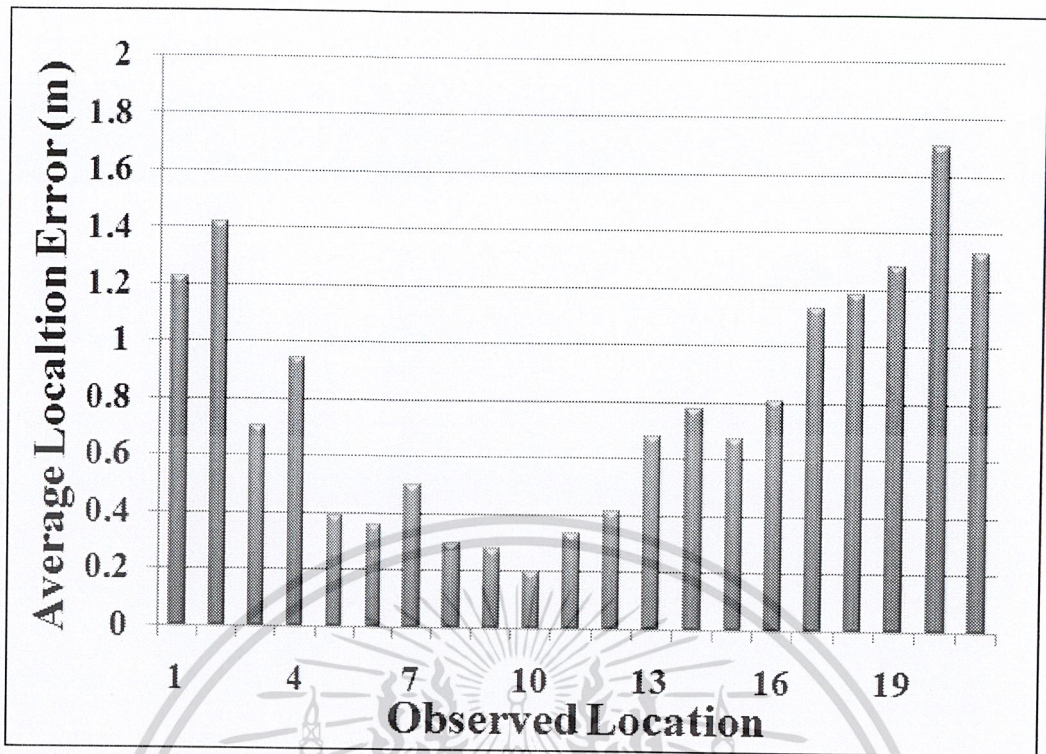


รูปที่ 4.44 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

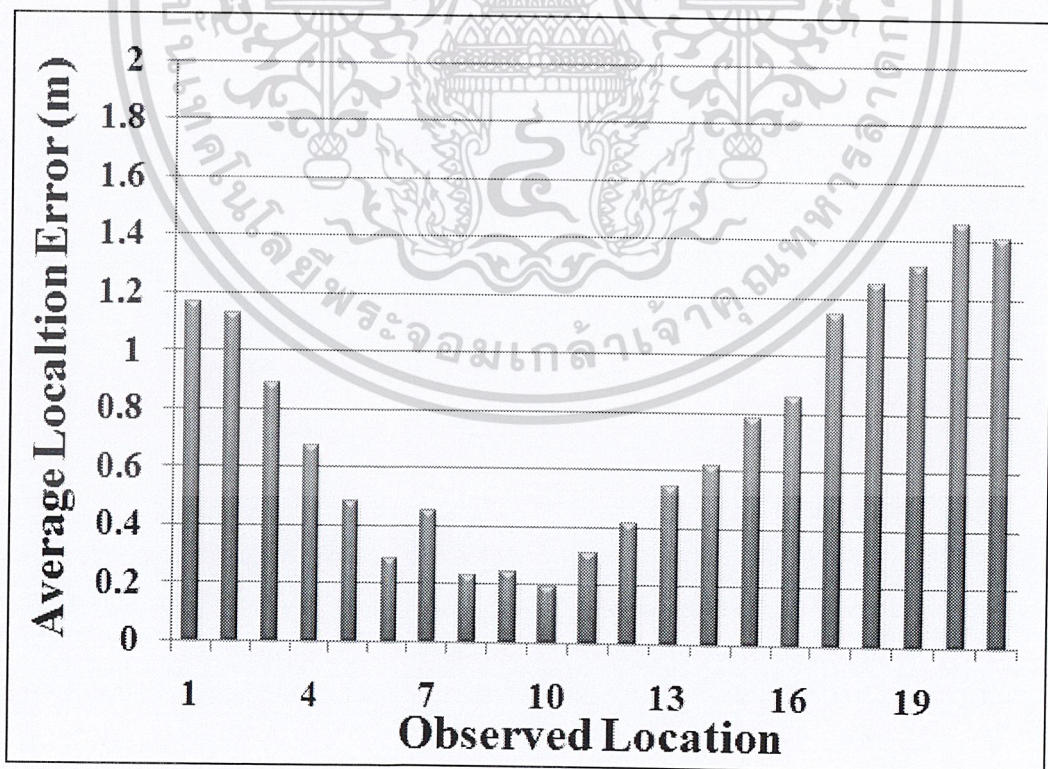


รูปที่ 4.45 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 80. ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



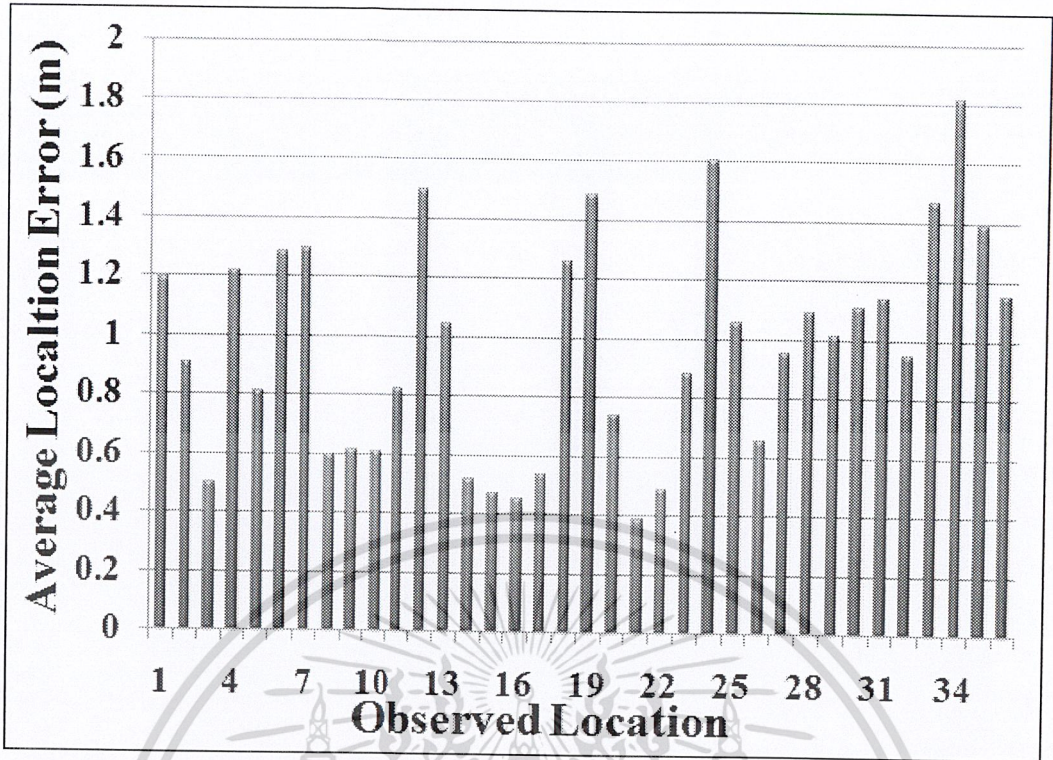
รูปที่ 4.46 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว



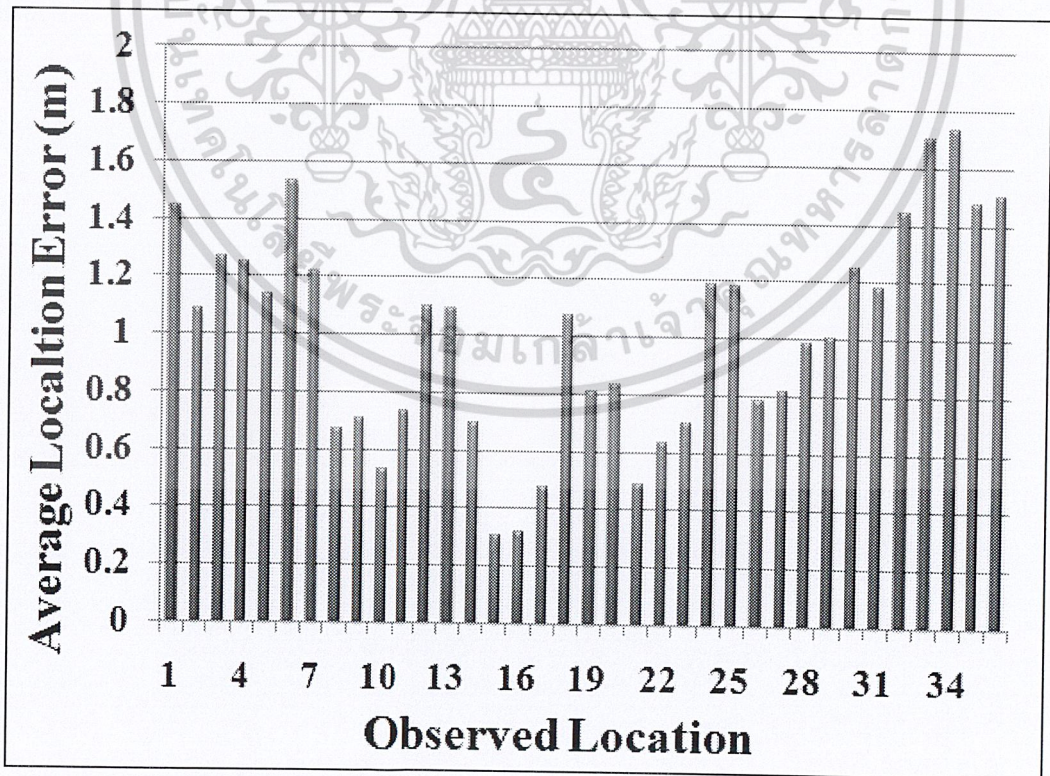
รูปที่ 4.47 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 81 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

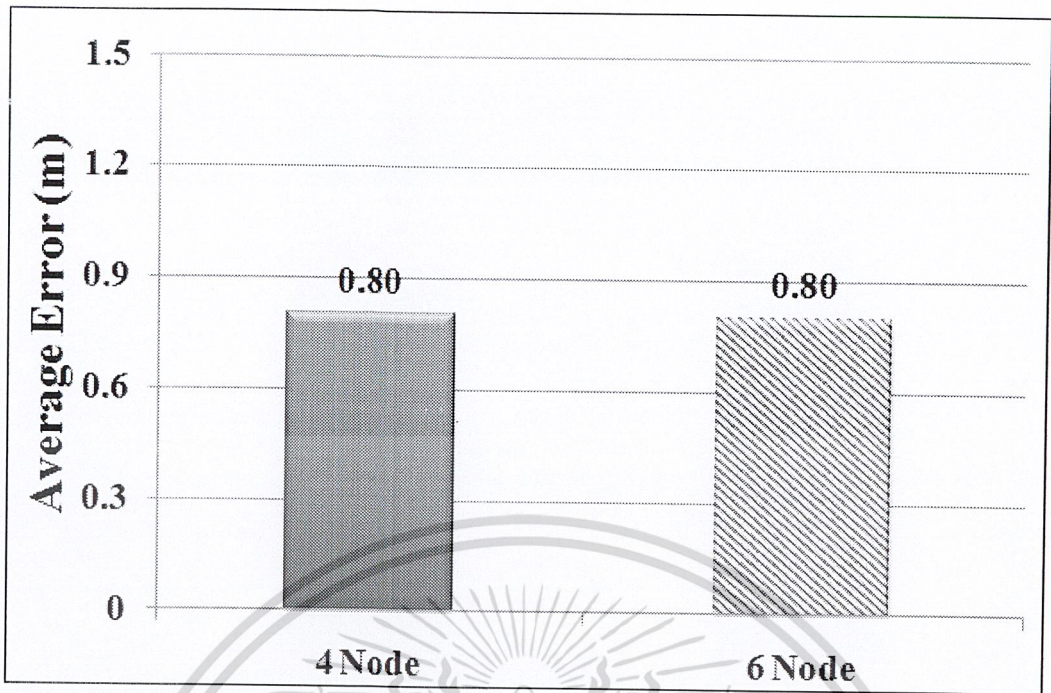


รูปที่ 4.48 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่งของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

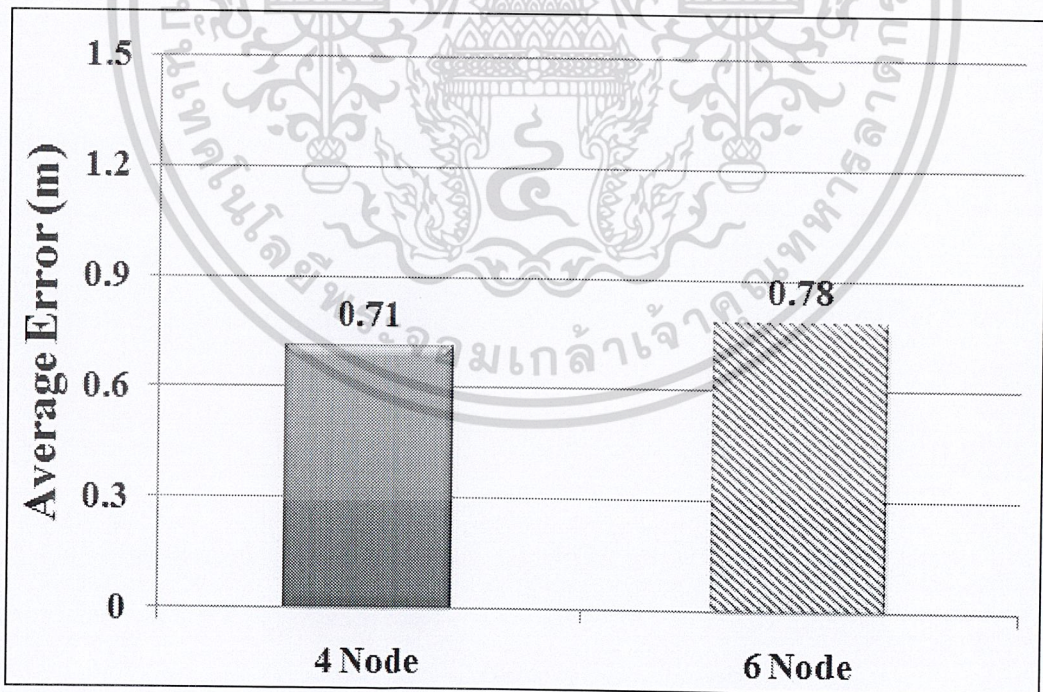


รูปที่ 4.49 ค่าความผิดพลาดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่งของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของสำนักงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา 82 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

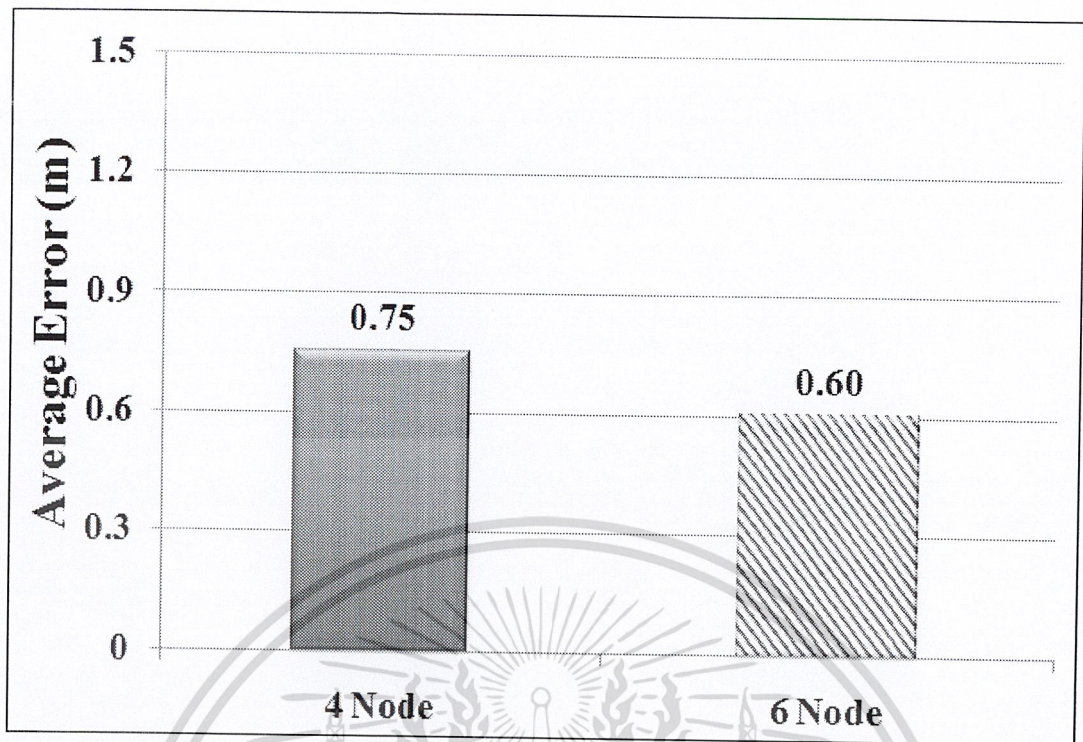


รูปที่ 4.50 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

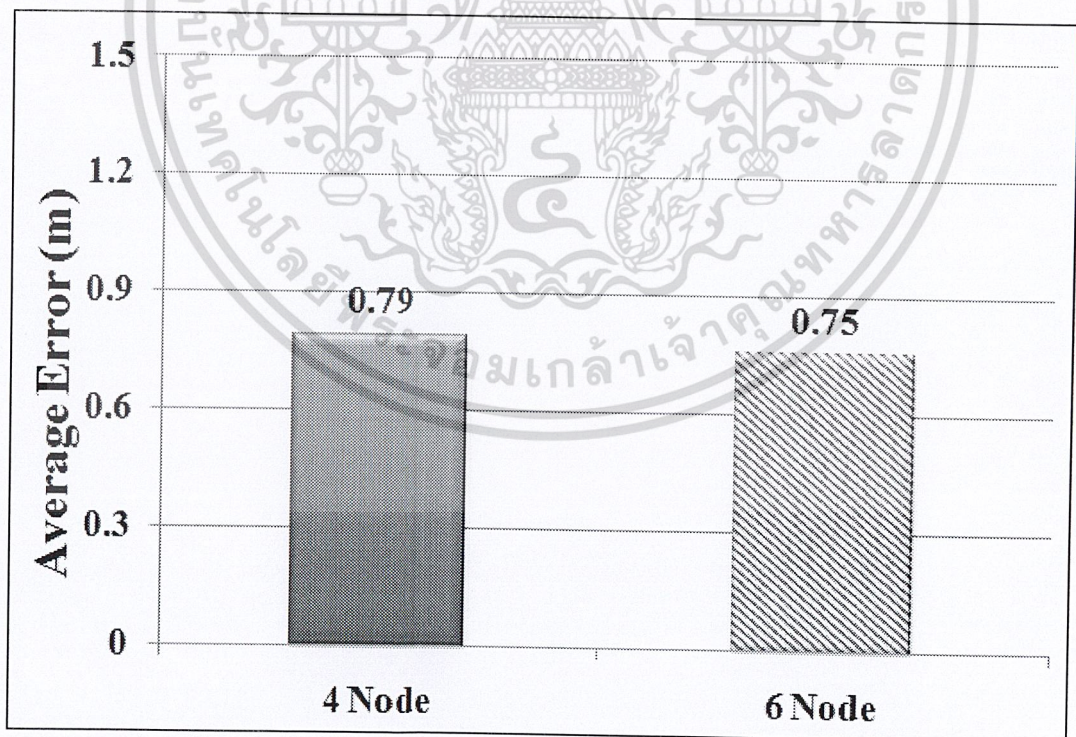


รูปที่ 4.51 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 83. ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

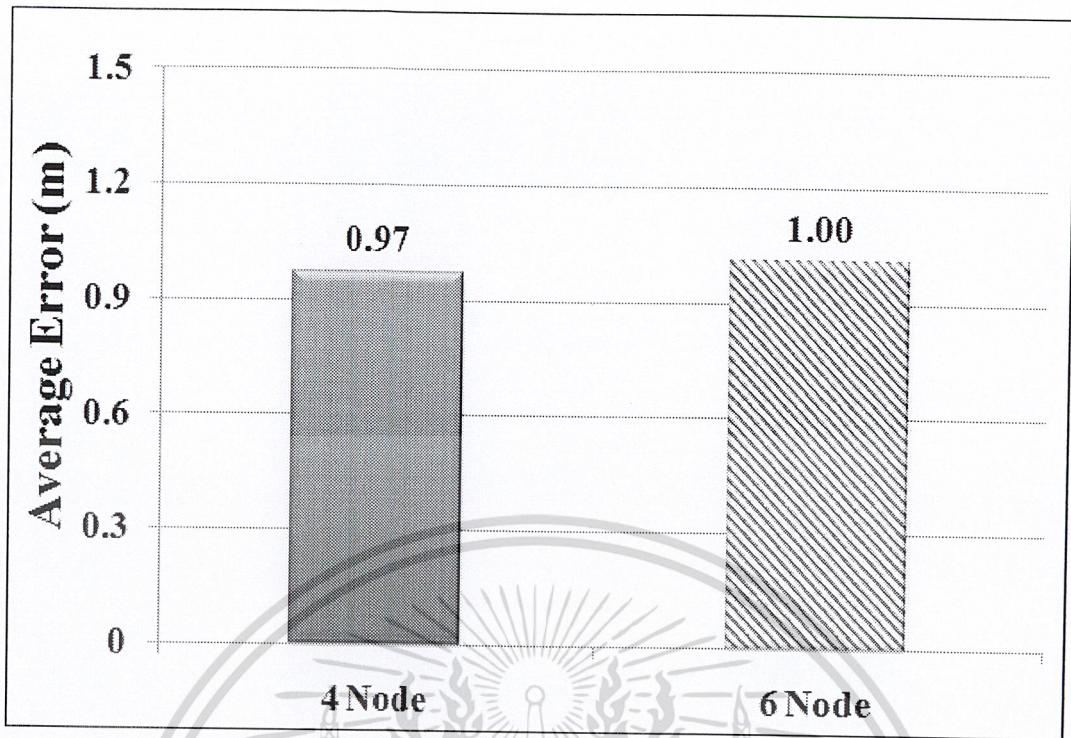


รูปที่ 4.52 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตามแนวอนกึ่งกลางของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

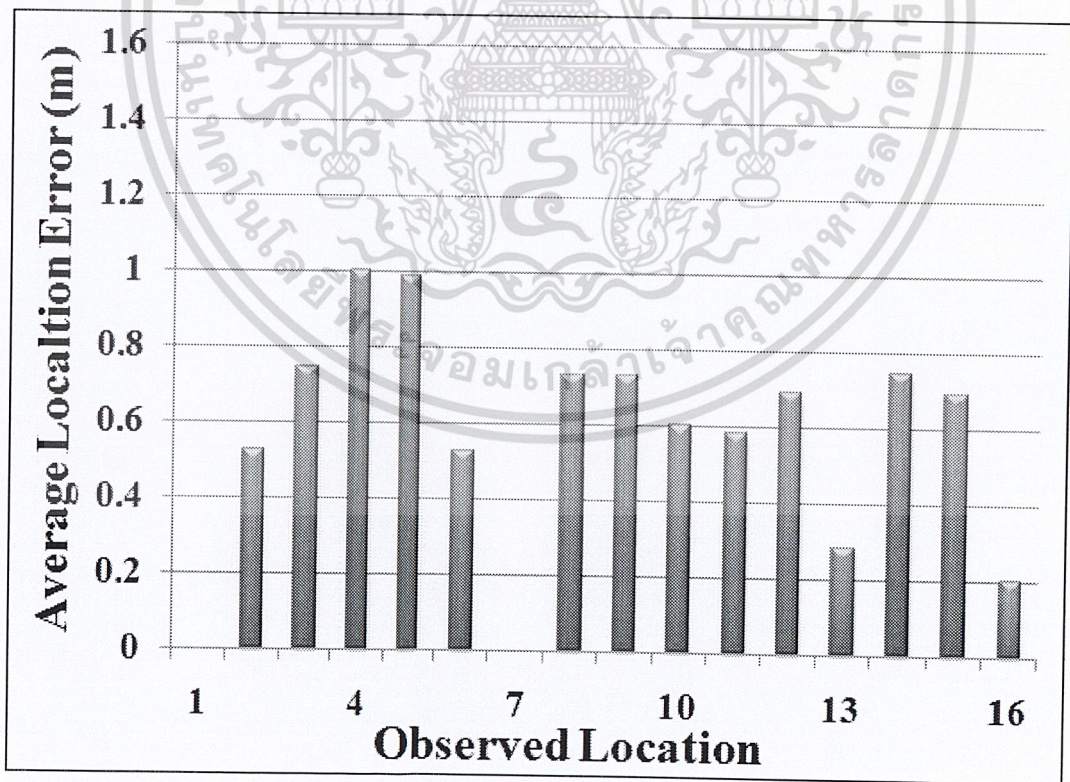


รูปที่ 4.53 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตามแนวตั้งกึ่งกลางของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 84 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.54 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Min-Max Algorithm ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

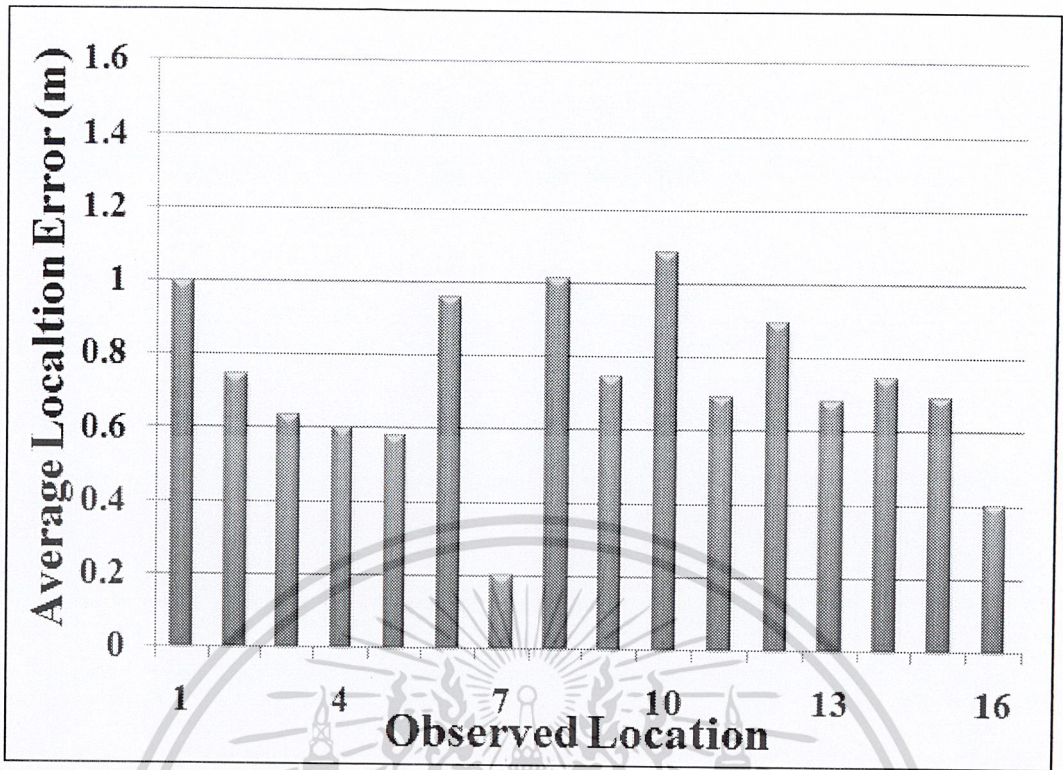


รูปที่ 4.55 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้

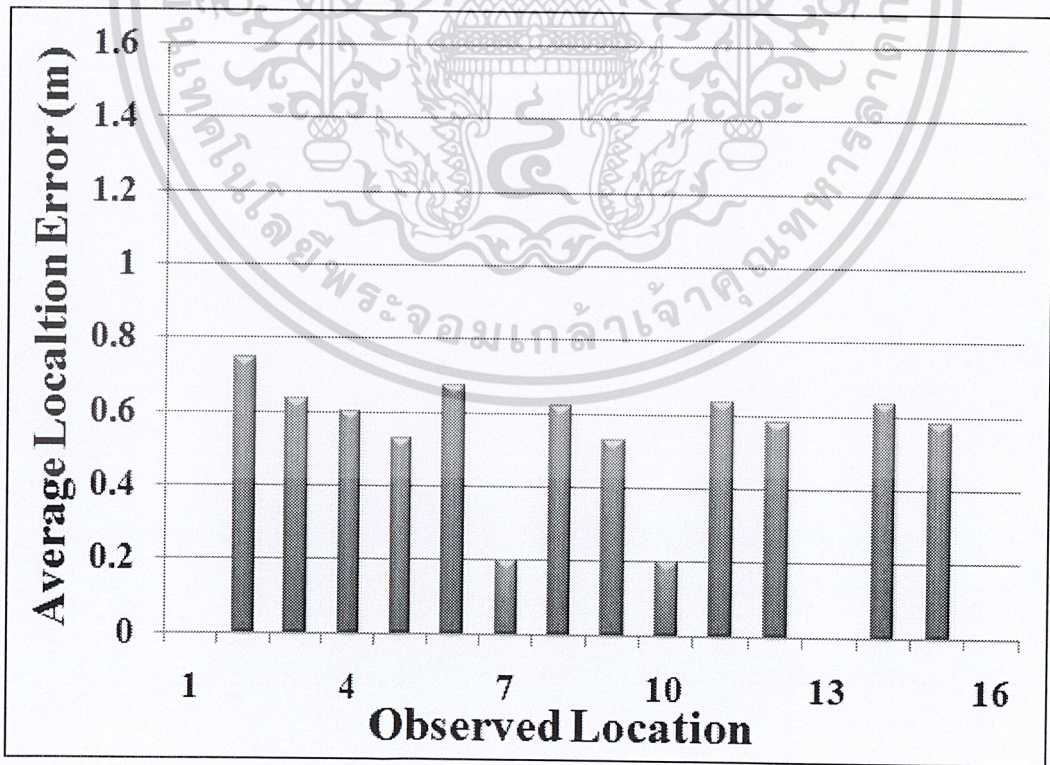
จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา 85 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

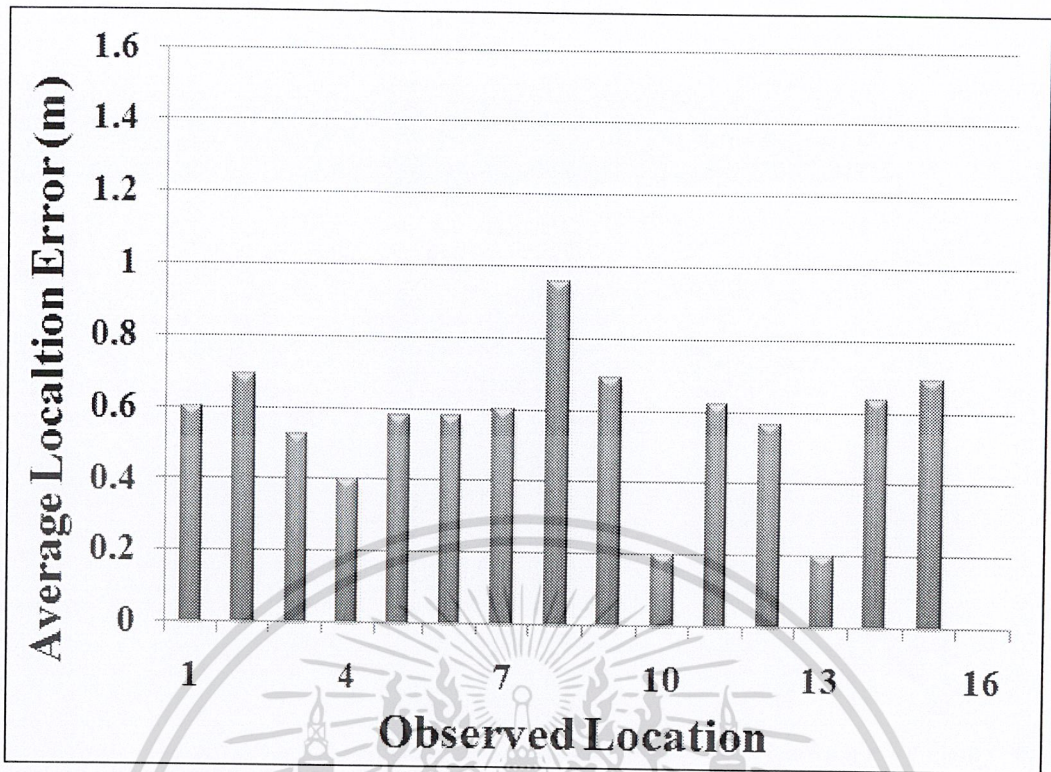


รูปที่ 4.56 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

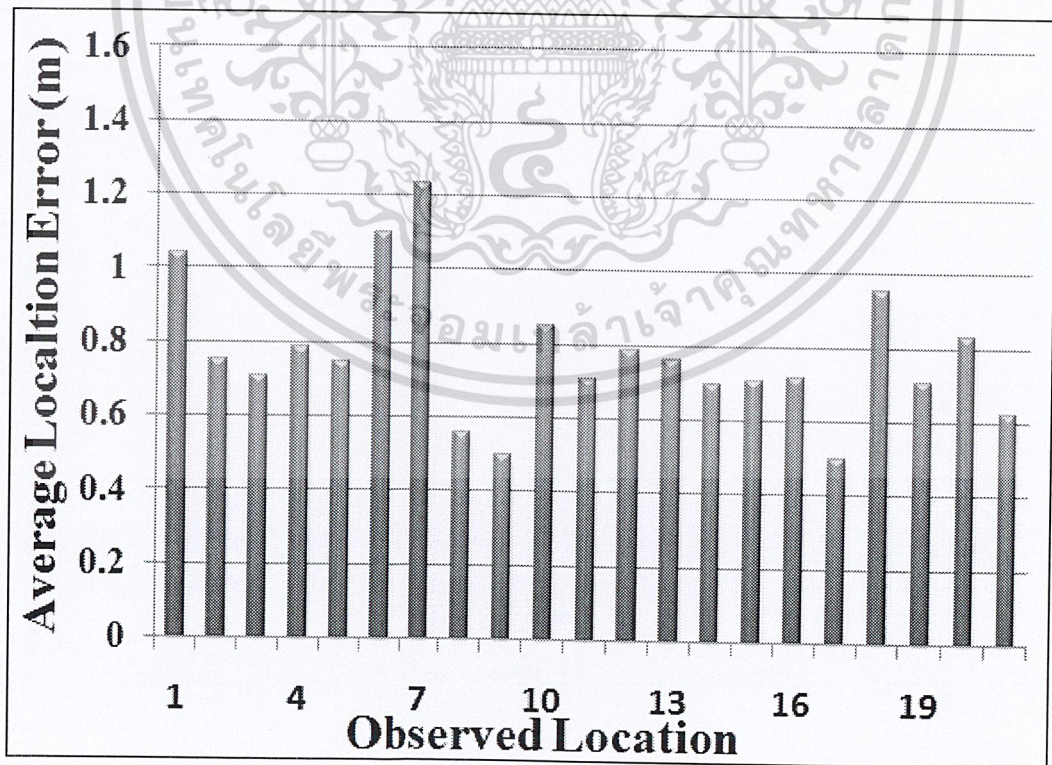


รูปที่ 4.57 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้

จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



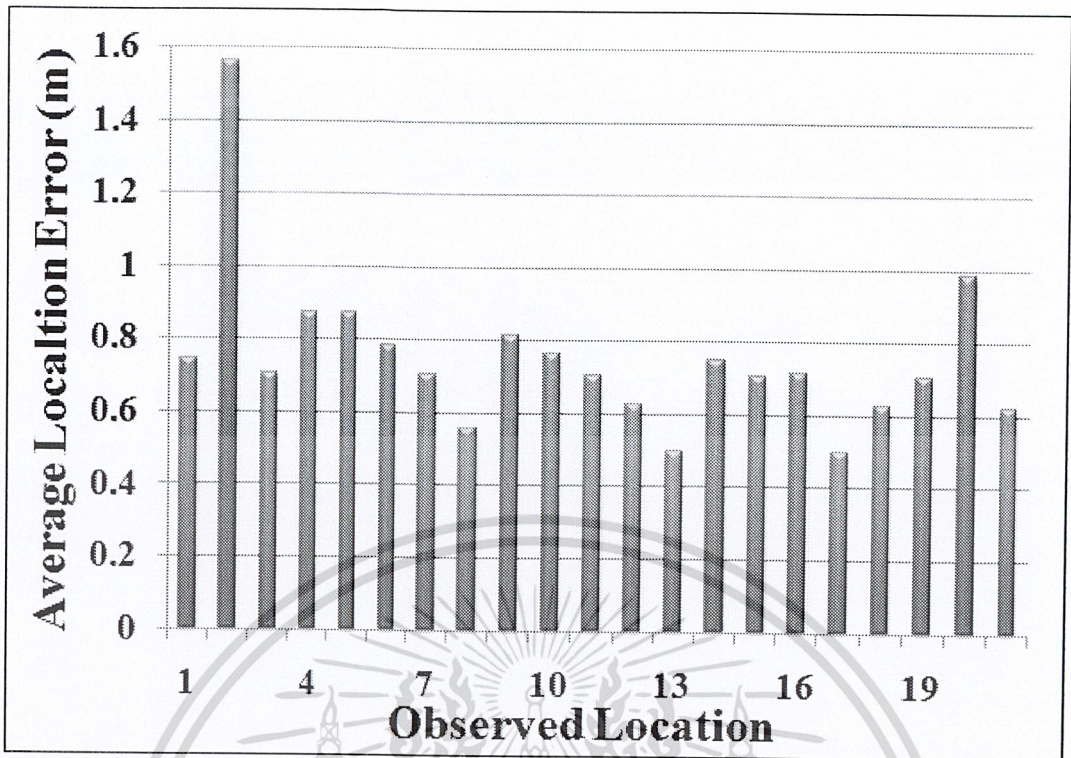
รูปที่ 4.58 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



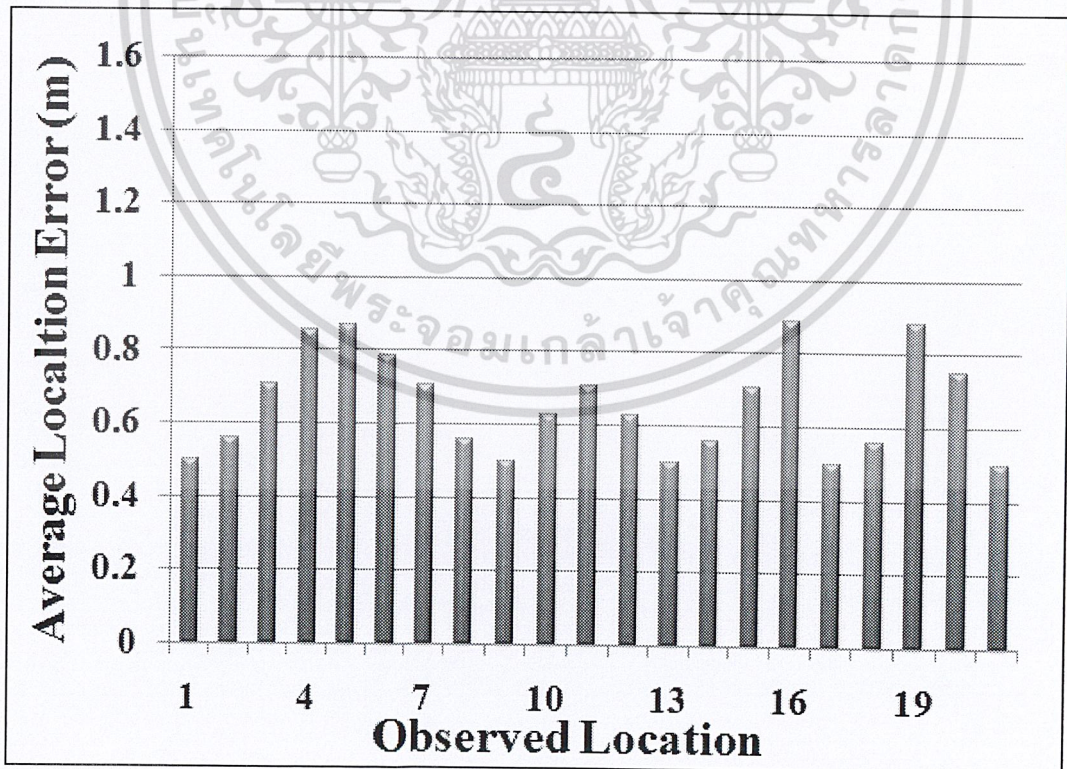
รูปที่ 4.59 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้

จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว ประโยชน์ด้านการค้า

ไม่ว่าการณ์ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 87 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.60 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

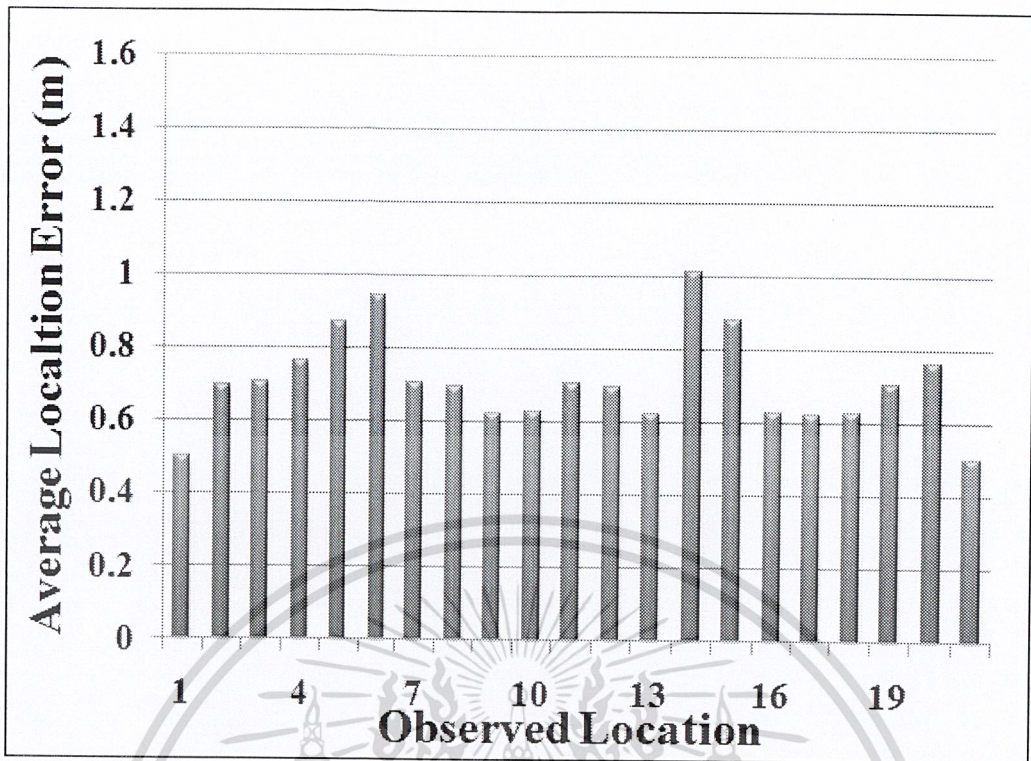


รูปที่ 4.61 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้

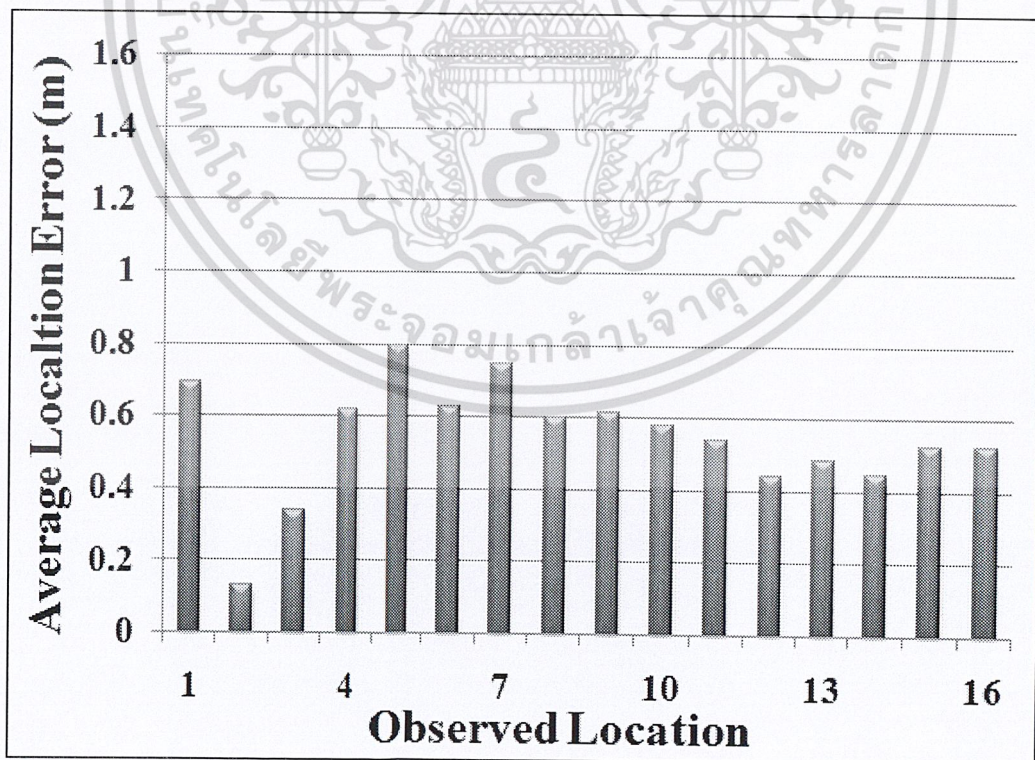
จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับการใช้ในเพื่อการศึกษาเท่านั้น เมื่อผู้จัดทำเอกสารได้ดำเนินการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 88 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.62 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากมุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

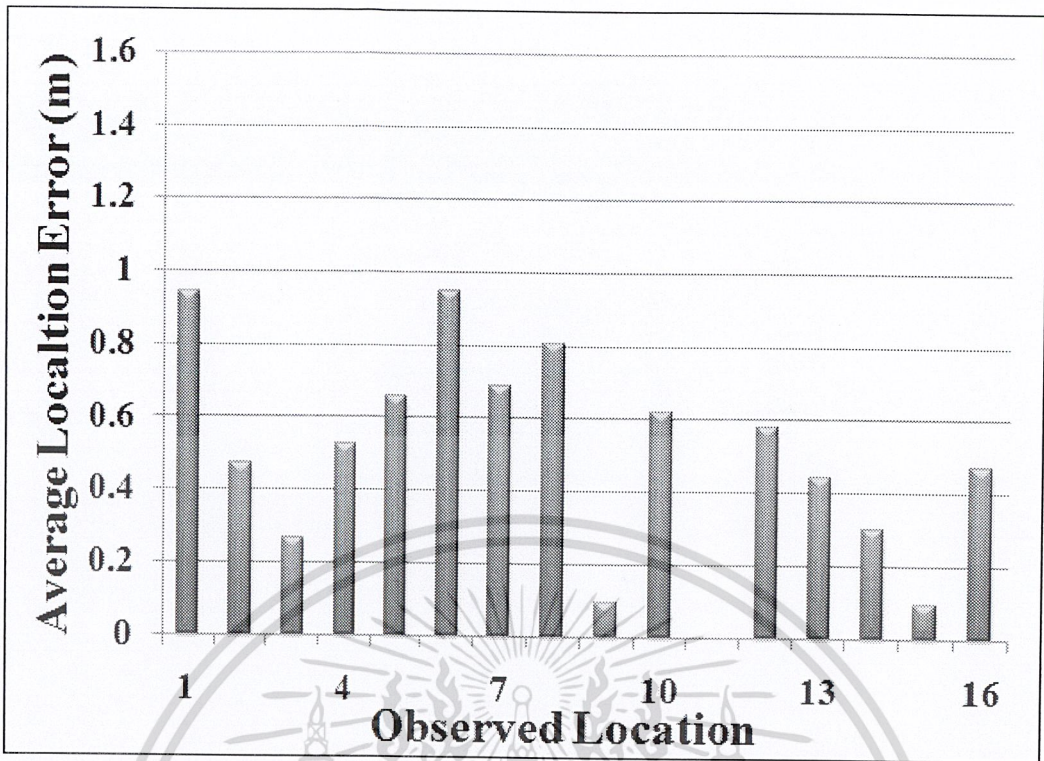


รูปที่ 4.63 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้

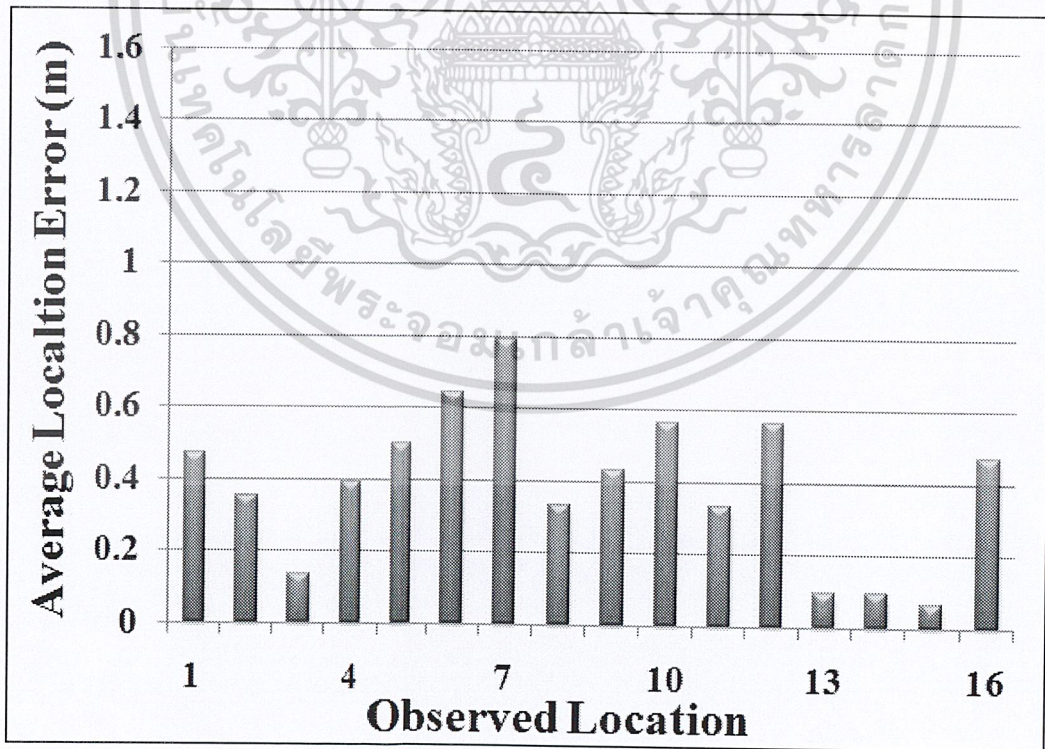
จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 89. และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

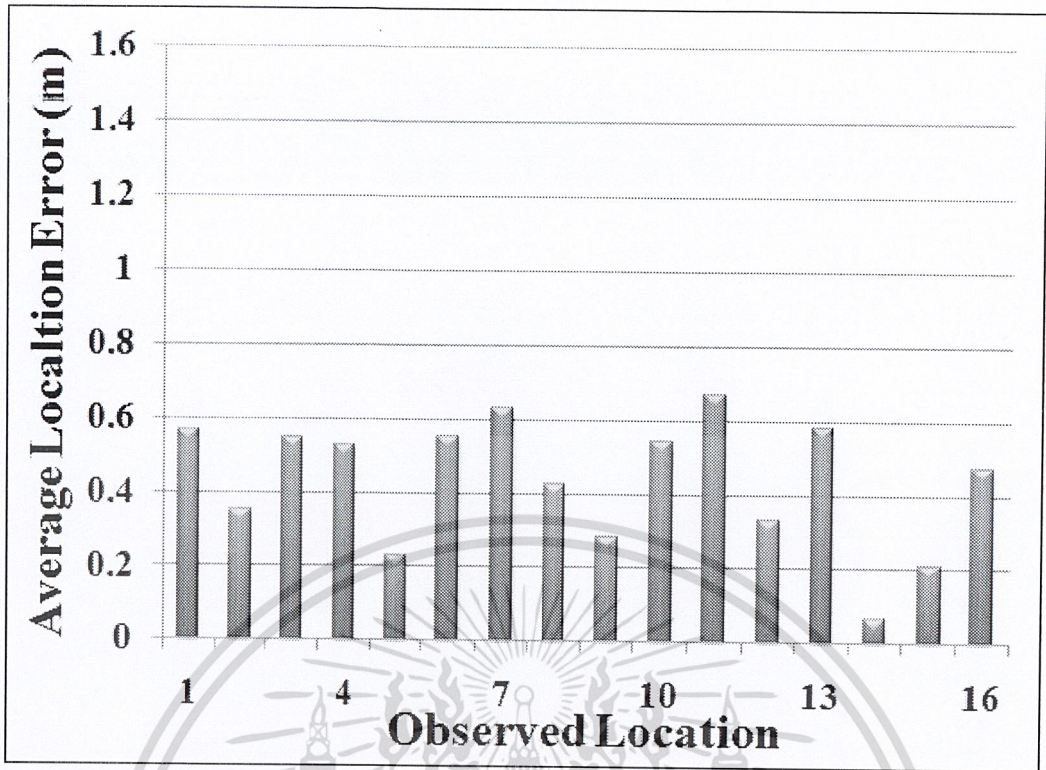


รูปที่ 4.64 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ จากตามแนวนอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

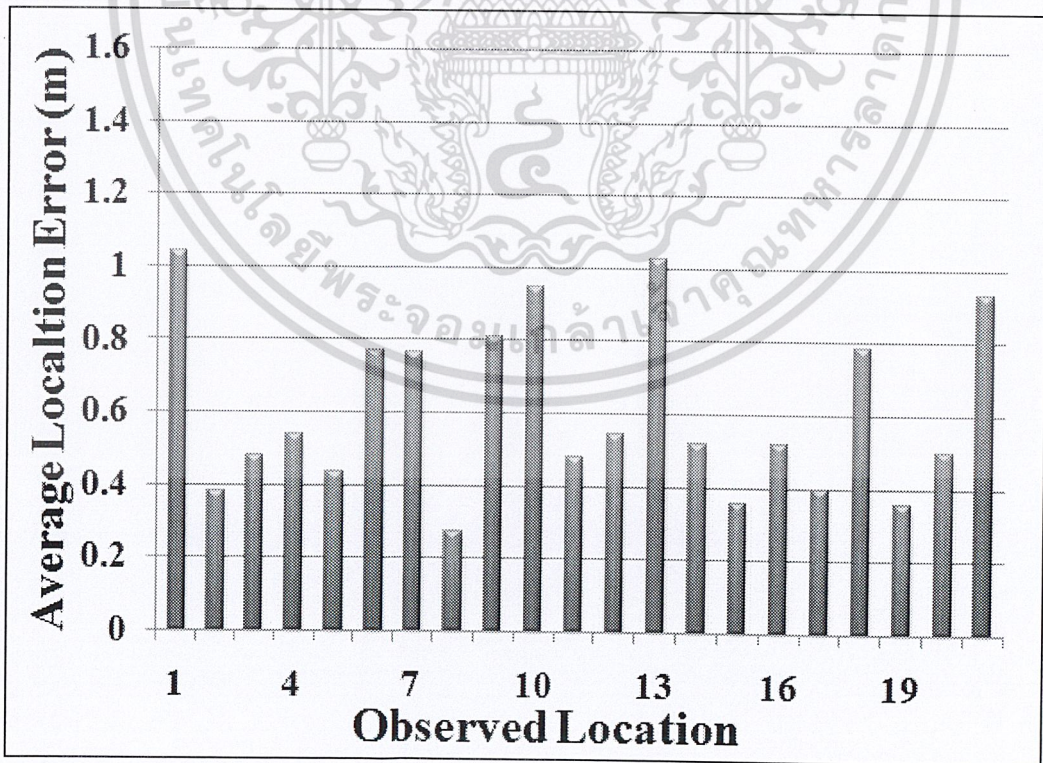


รูปที่ 4.65 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ จากตามแนวนอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

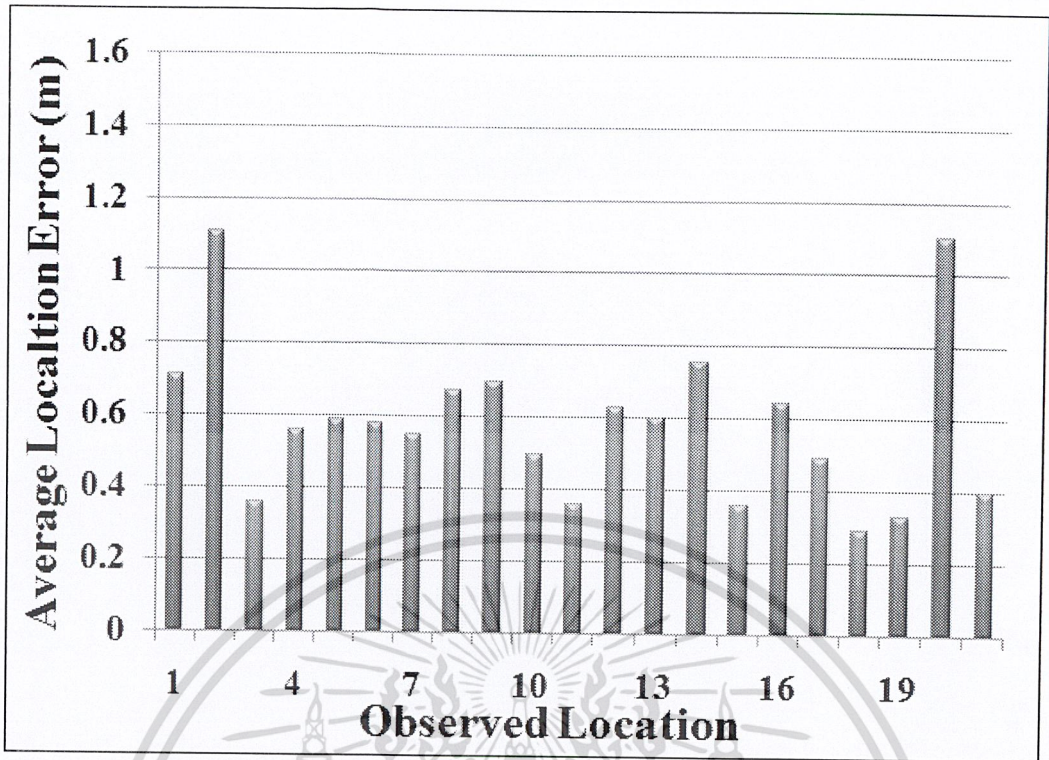
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 90. ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



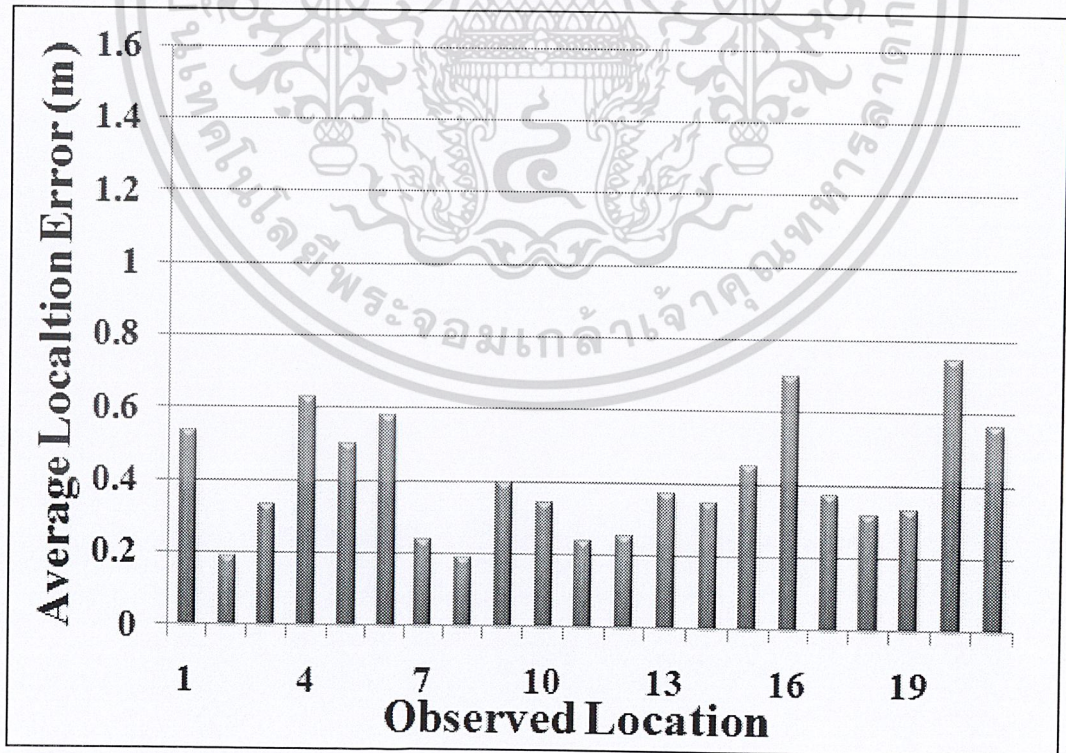
รูปที่ 4.66 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ จากตามแนวอนกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว



รูปที่ 4.67 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

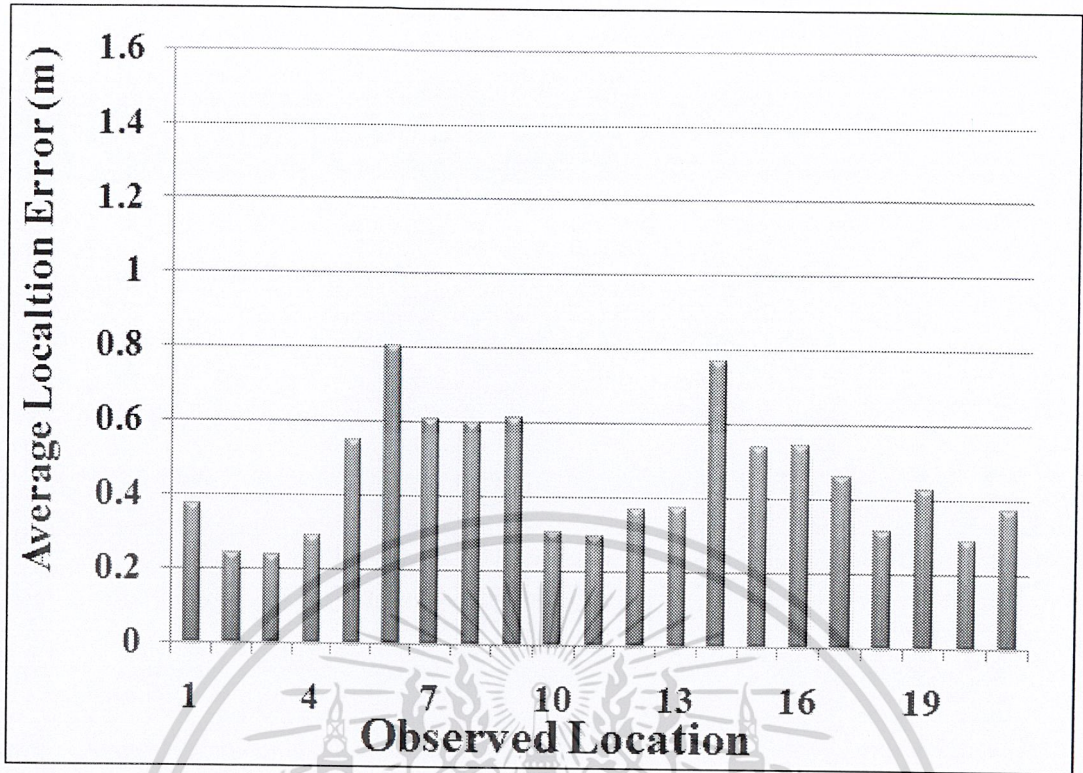


รูปที่ 4.68 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

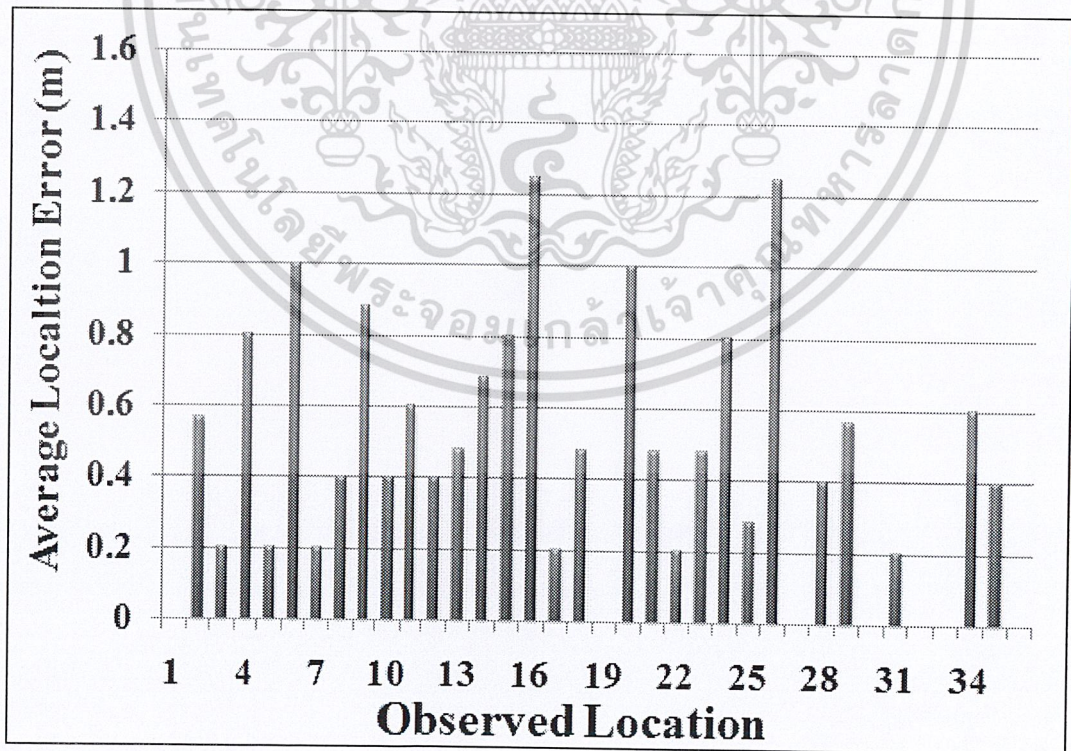


รูปที่ 4.69 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 92 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.70 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ จากตามแนวตั้งกึ่งกลางของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

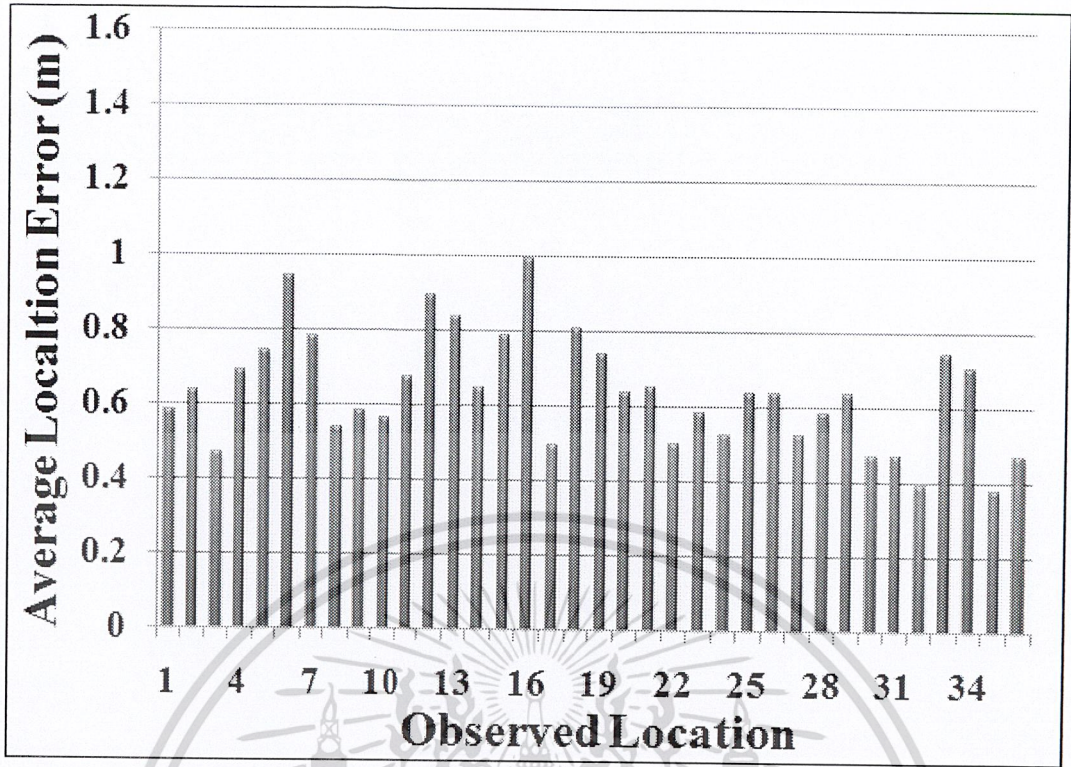


รูปที่ 4.71 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้

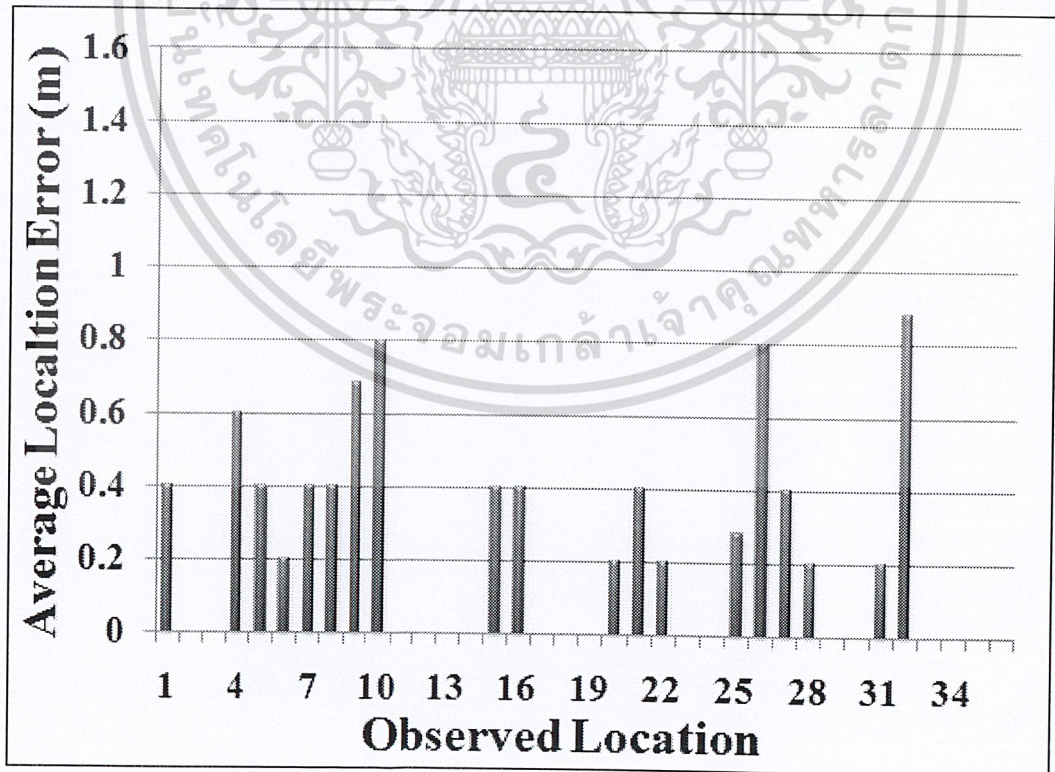
จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 93 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

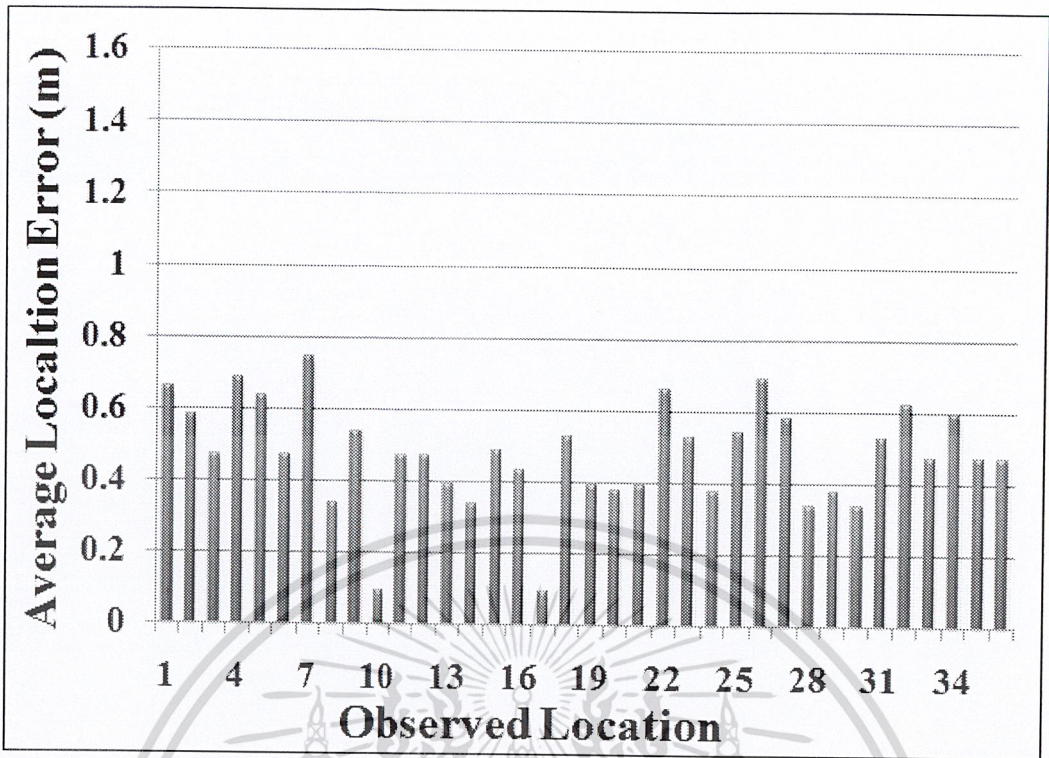


รูปที่ 4.72 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 4 ตัว

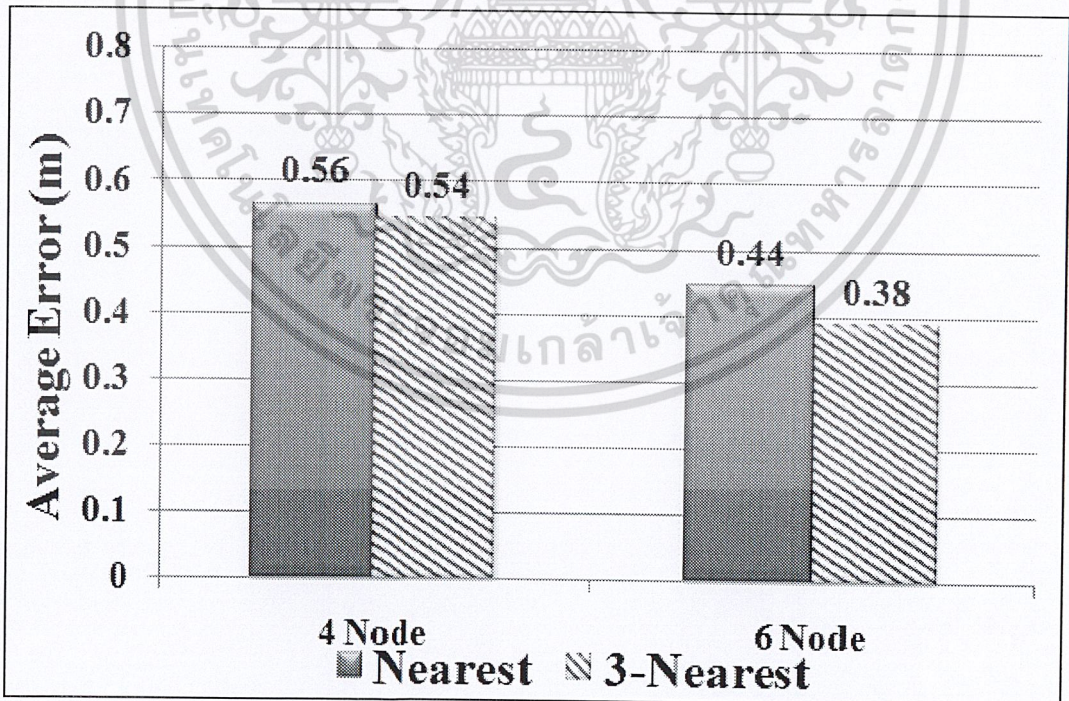


รูปที่ 4.73 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ Nearest ที่ใช้ค่า RSSI ที่เก็บไว้

จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว
 เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 94 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

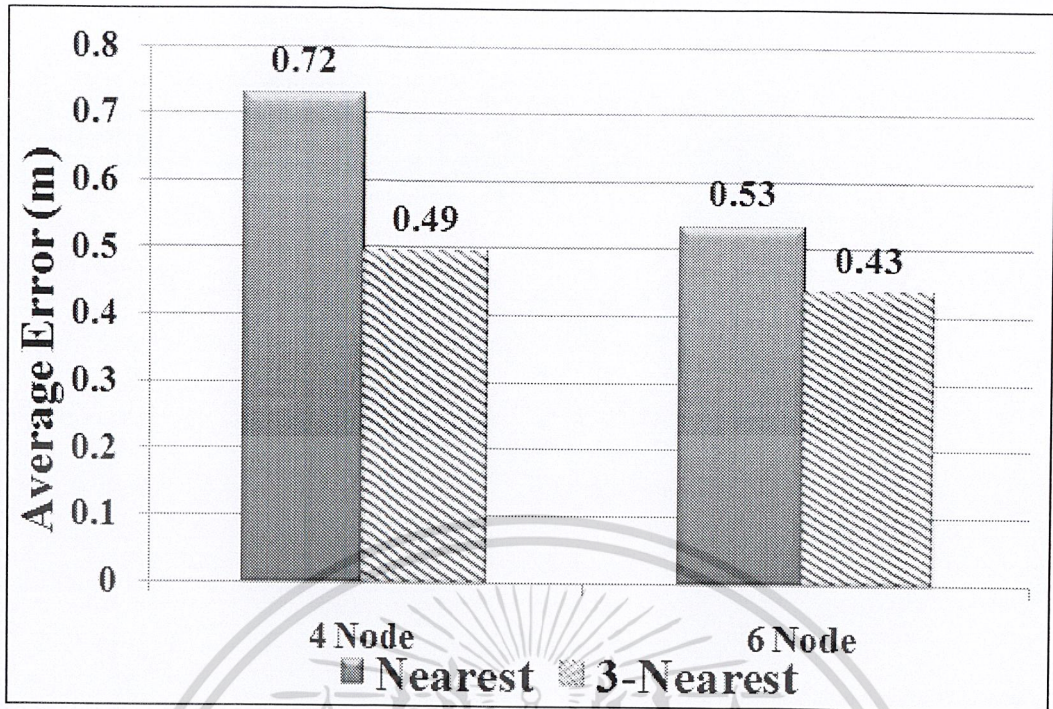


รูปที่ 4.74 ค่าความผิดพลาดของวิธี Fingerprint Algorithm แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก 36 ตำแหน่ง ของพื้นที่ทดลอง โดยใช้ตัวรับ 6 ตัว

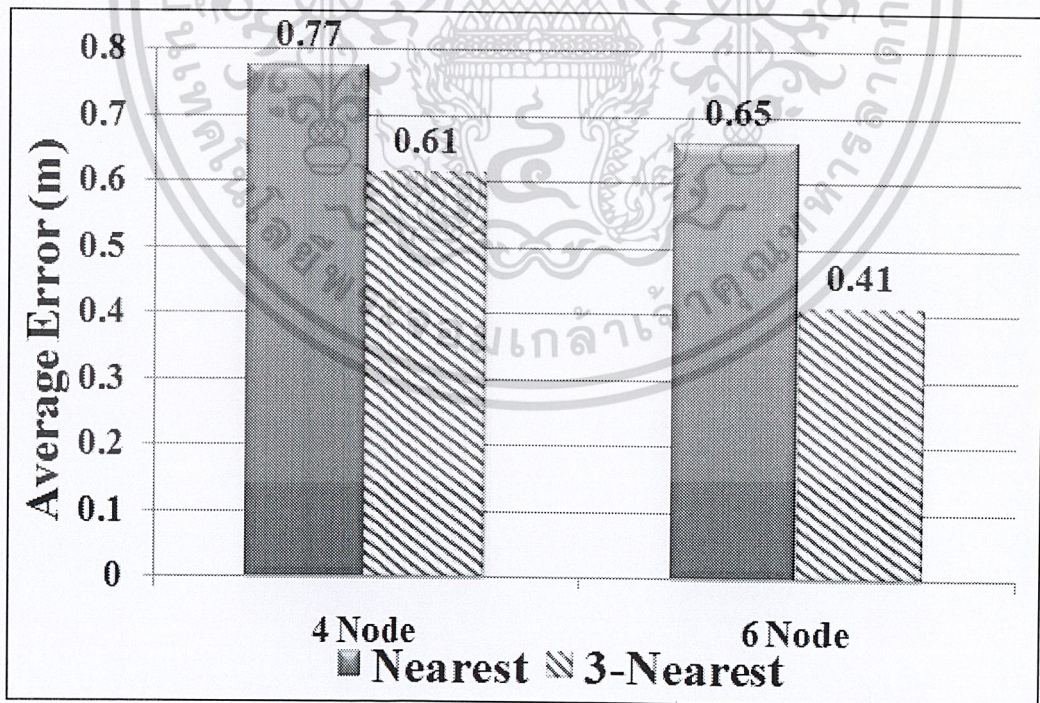


รูปที่ 4.75 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และแบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

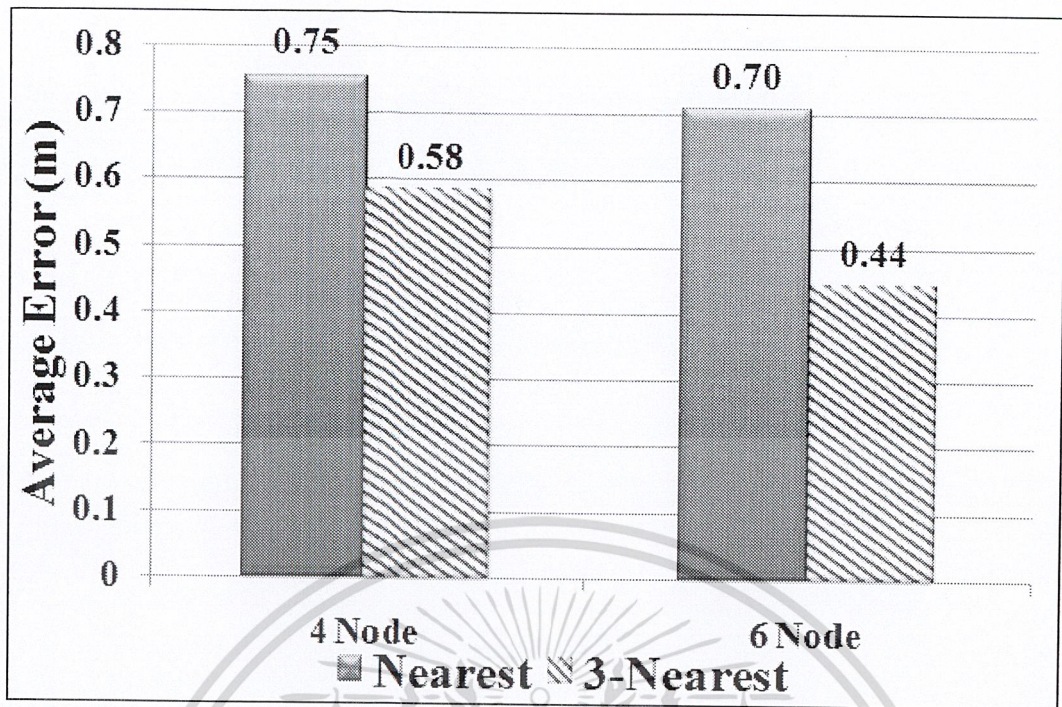
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 95.จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



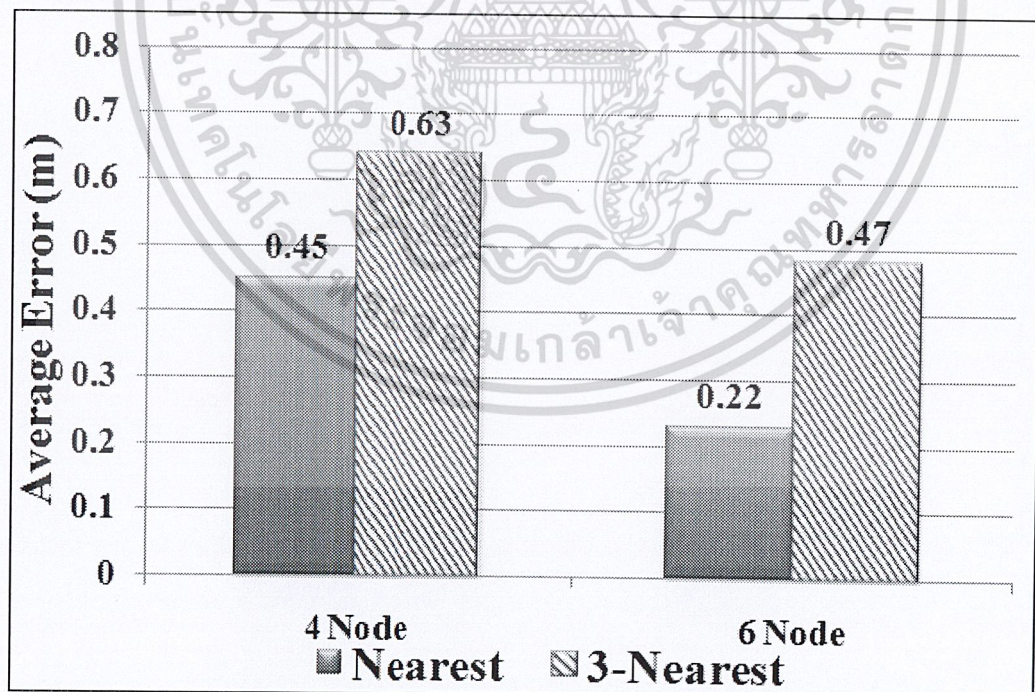
รูปที่ 4.76 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และแบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว



รูปที่ 4.77 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และแบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

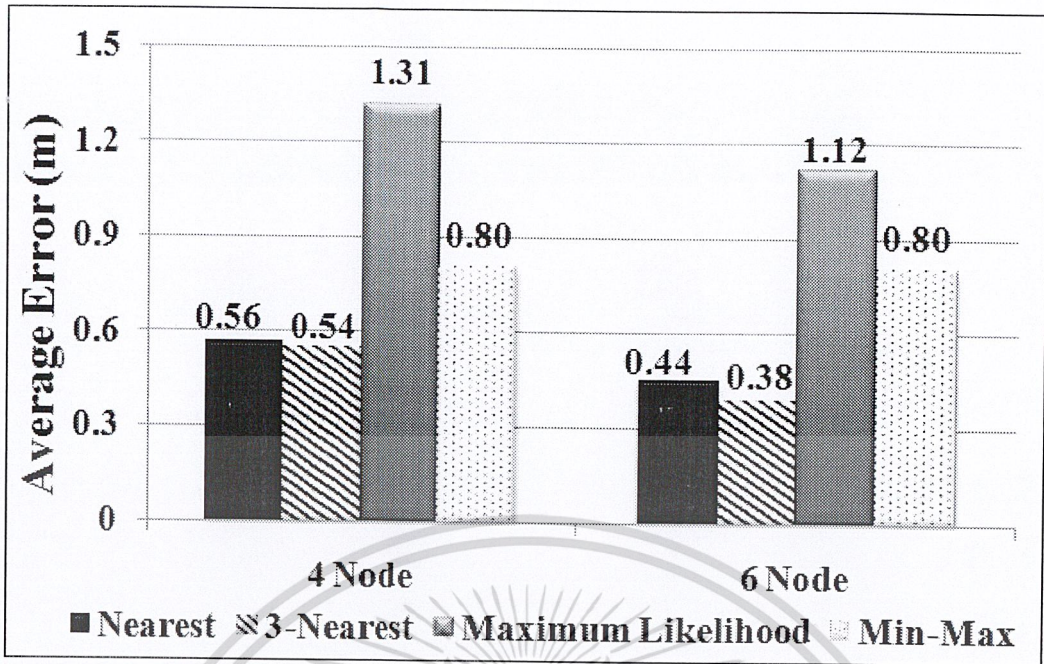


รูปที่ 4.78 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และแบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

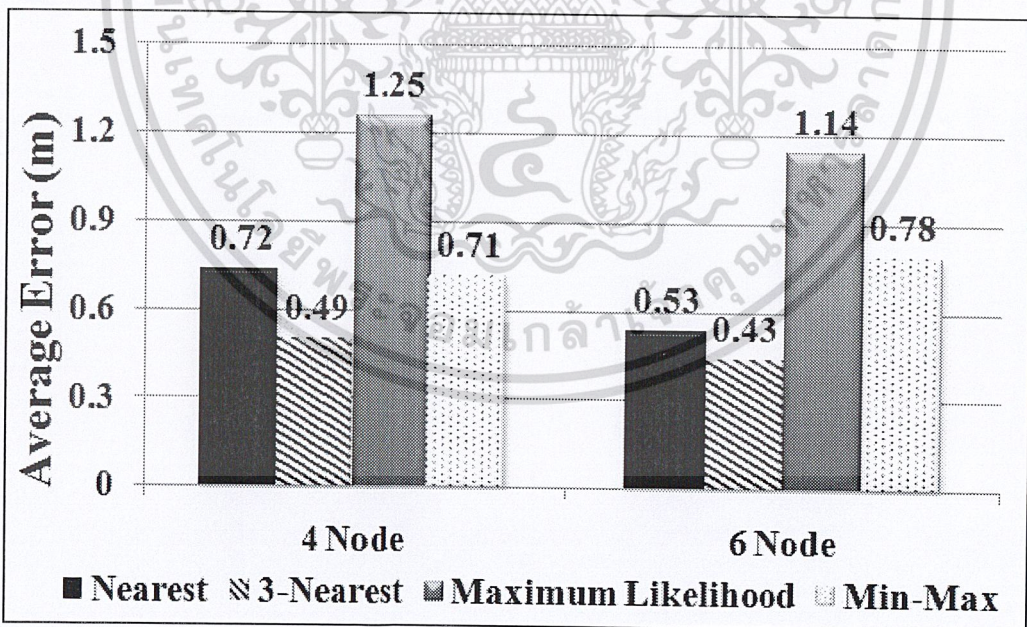


รูปที่ 4.79 ค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Fingerprint Algorithm แบบ Nearest และแบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อ 97 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

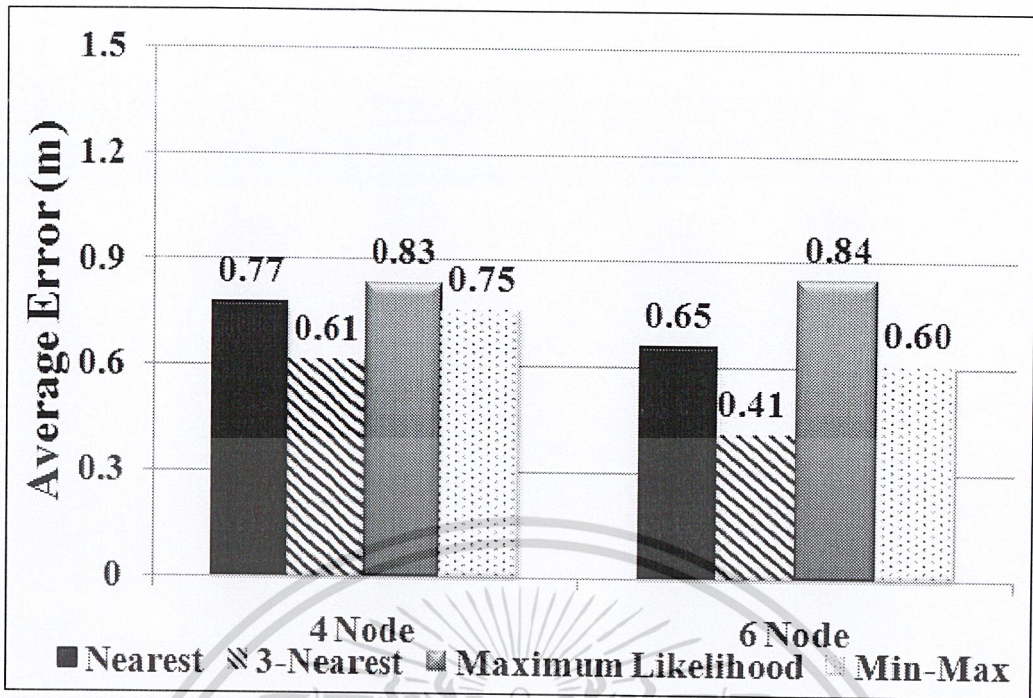


รูปที่ 4.80 เปรียบเทียบค่าความผิดพลาดผลเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm, Min-Max Algorithm และ Fingerprint แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

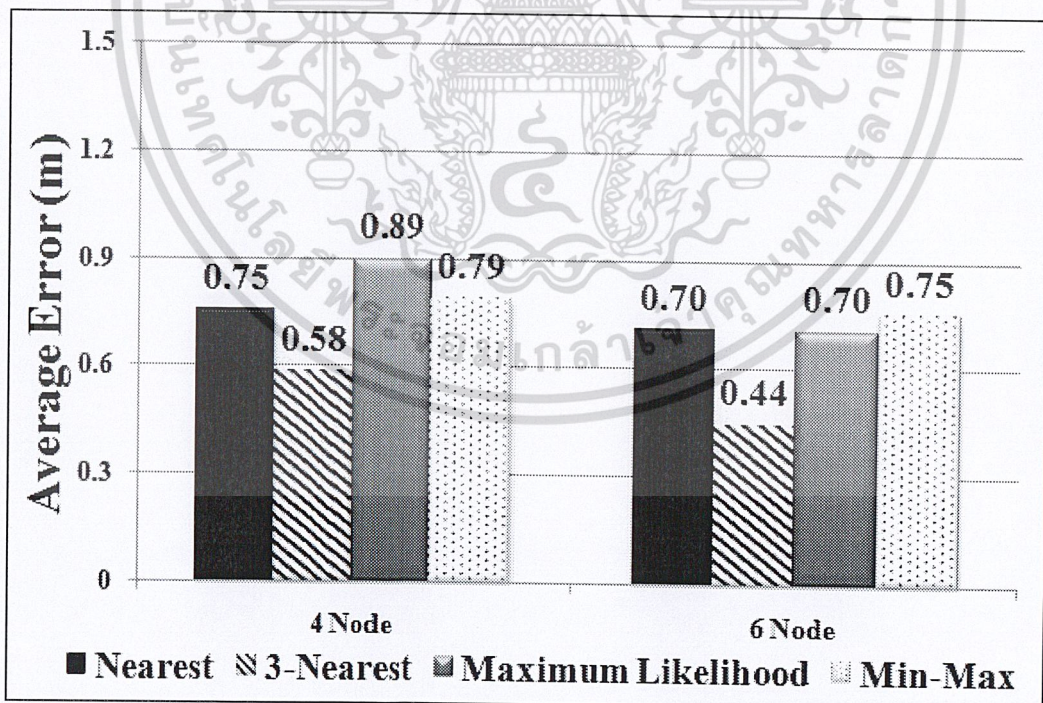


รูปที่ 4.81 เปรียบเทียบค่าความผิดพลาดผลเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm, Min-Max Algorithm และ Fingerprint แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จาก มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

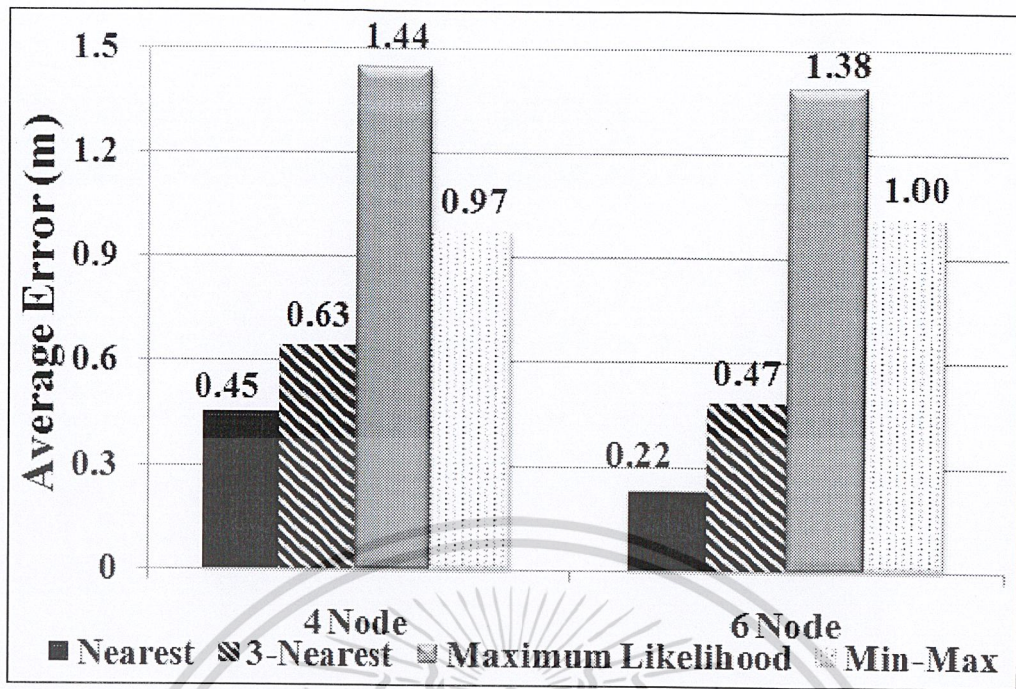
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา.98 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.82 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm, Min-Max Algorithm และ Fingerprint แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว



รูปที่ 4.83 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm, Min-Max Algorithm และ Fingerprint แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้จากตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 99 และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.84 เปรียบเทียบค่าความผิดพลาดเฉลี่ยรวมทั้งหมดของวิธี Maximum Likelihood Algorithm, Min-Max Algorithm และ Fingerprint แบบ Nearest และ แบบ 3-Nearest ที่ใช้ค่า RSSI ที่เก็บไว้ตาม 36 ตำแหน่ง ของพื้นที่ทดลองโดยใช้ตัวรับ 4 ตัว และตัวรับ 6 ตัว

4.7 การทดลองหาค่าตำแหน่งของวัตถุของแต่ละอัลกอริทึม แบบเคลื่อนที่ในอาคาร

จุดประสงค์

- คู่มือโปรแกรมเคลื่อนที่ของวัตถุที่คำนวณได้กับการเคลื่อนที่จริง

อุปกรณ์

- XBee Module 7 ตัว (ตัวส่ง 1 ตัว, ตัวรับ 6 ตัว)
- คอมพิวเตอร์ 1 เครื่อง
- กล่อง ขนาด 21×29.7 ซม. 7 กล่อง
- โปรแกรมใช้ในการหาค่าตำแหน่งของวัตถุ

วิธีการทดลอง

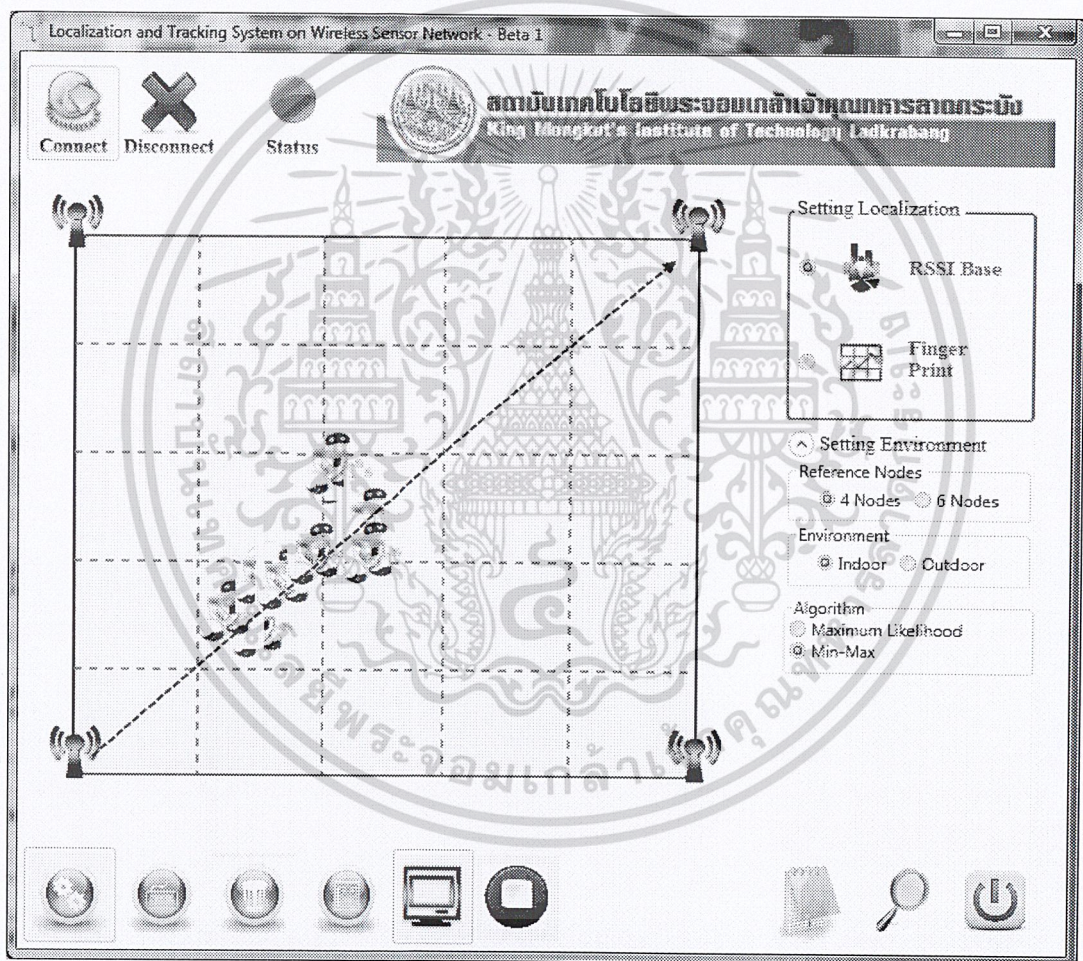
- จัดเตรียมพื้นที่ขนาด 5×5 เมตร โดยใช้ตัวรับทั้ง 4 ตัว และ ตัวรับ 6 ตัว ตามลำดับ (เหมือนการทดลองที่ 4.4)
- ใช้โปรแกรมหาค่าตำแหน่งของวัตถุ แสดงดังรูปที่ 3.7 หาค่าตำแหน่งของวัตถุที่เคลื่อนที่ตามแนวการเคลื่อนที่ 4 แบบ เหมือนการทดลองที่ 4.5 แสดงดังรูปที่ 4.20, 4.21, 4.22 และ 4.23 ตามลำดับ

- ทำตามข้อ 2 โดยใช้ Maximum Likelihood Algorithm, Min-Max Algorithm, Finger print Algorithm ทั้งแบบ Nearest และ 3-Nearest โดยทั้งหมดใช้ตัวรับทั้ง 4 ตัว และ 6 ตัว

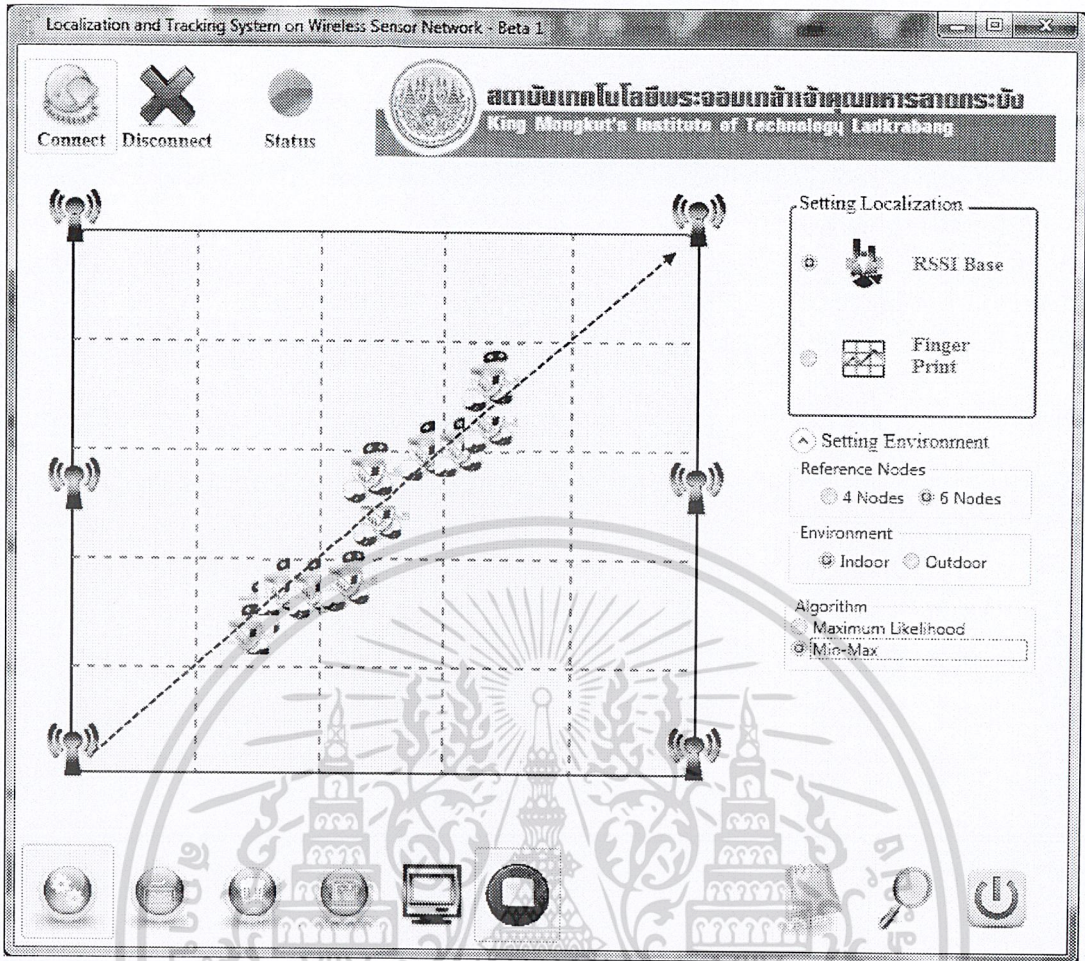
สภาพแวดล้อม

- ภายในอาคาร บริเวณหน้าลิฟต์ ชั้น 12 ตึก E12 คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (เหมือนการทดลองที่ 4.4)

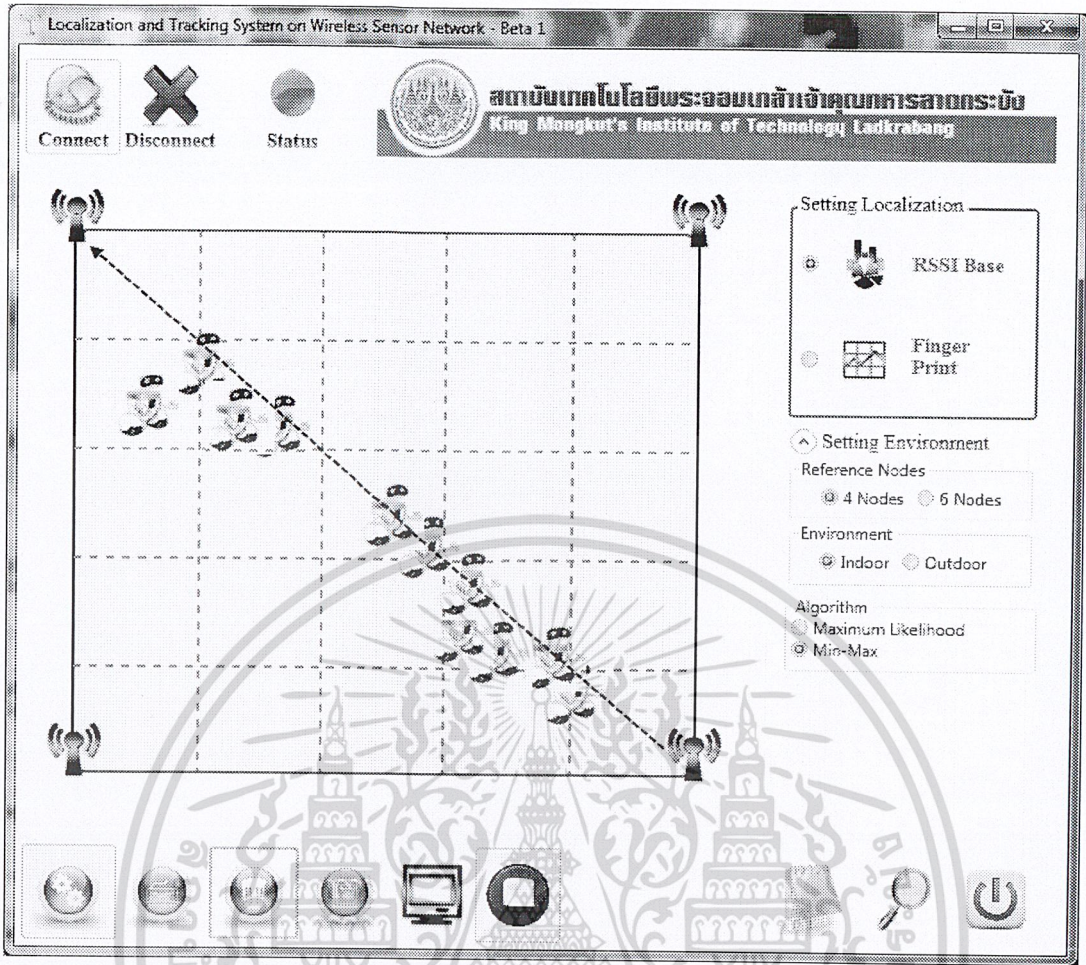
ผลการทดลอง



รูปที่ 4.85 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

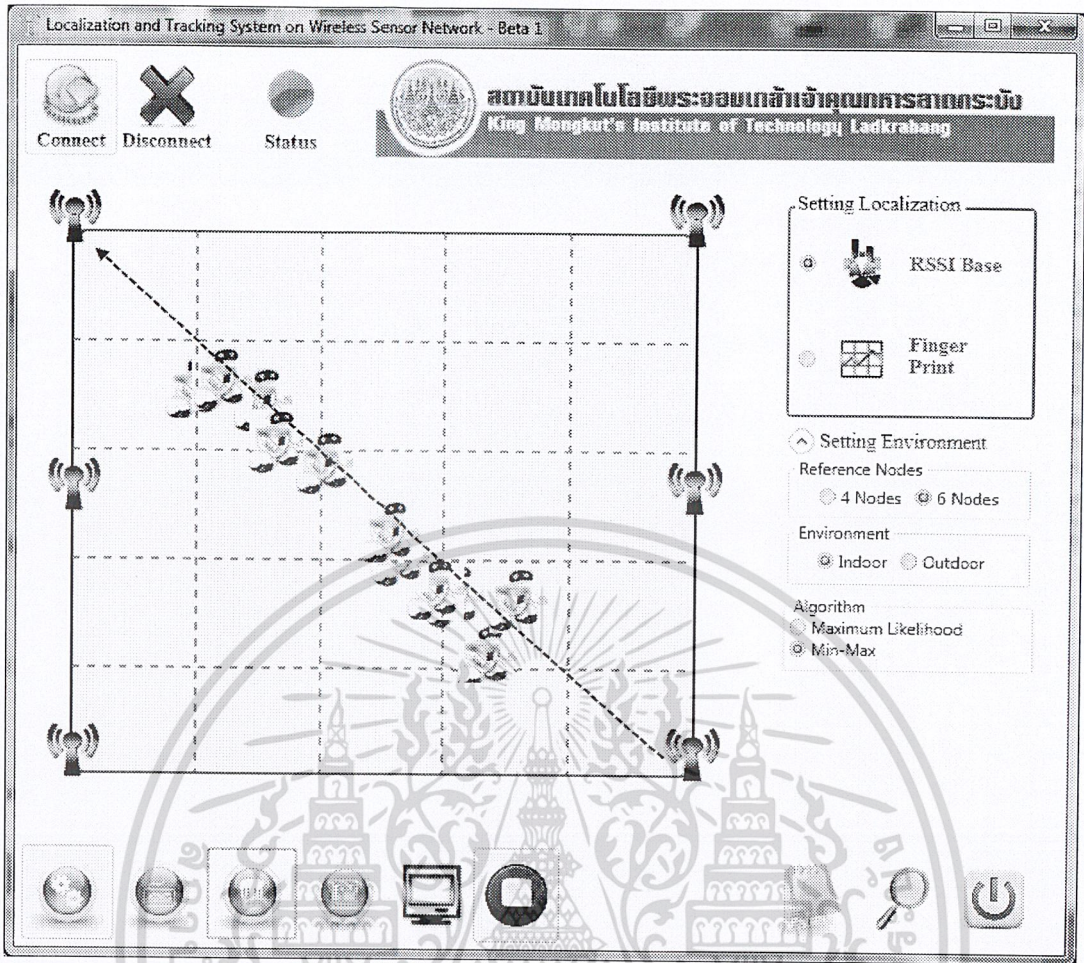


รูปที่ 4.86 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

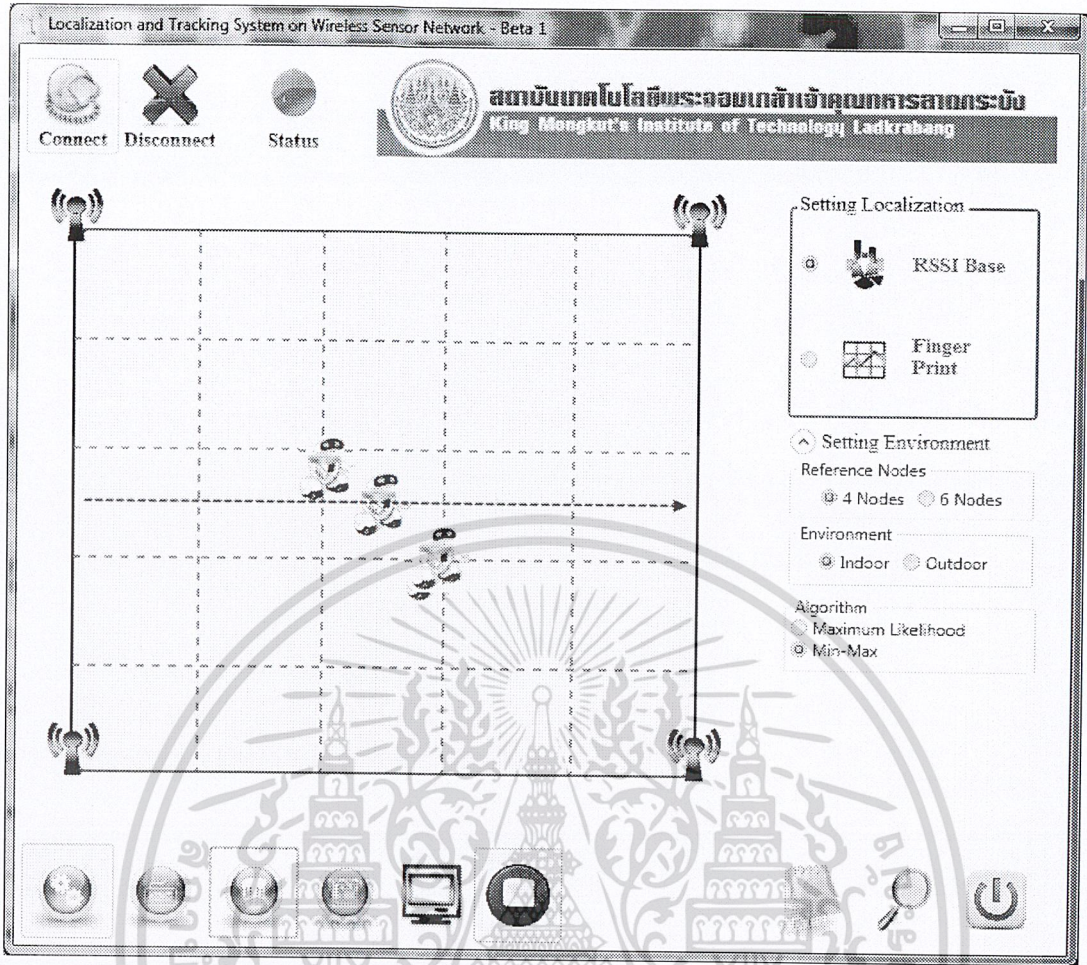


รูปที่ 4.87 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนว มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 103 จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

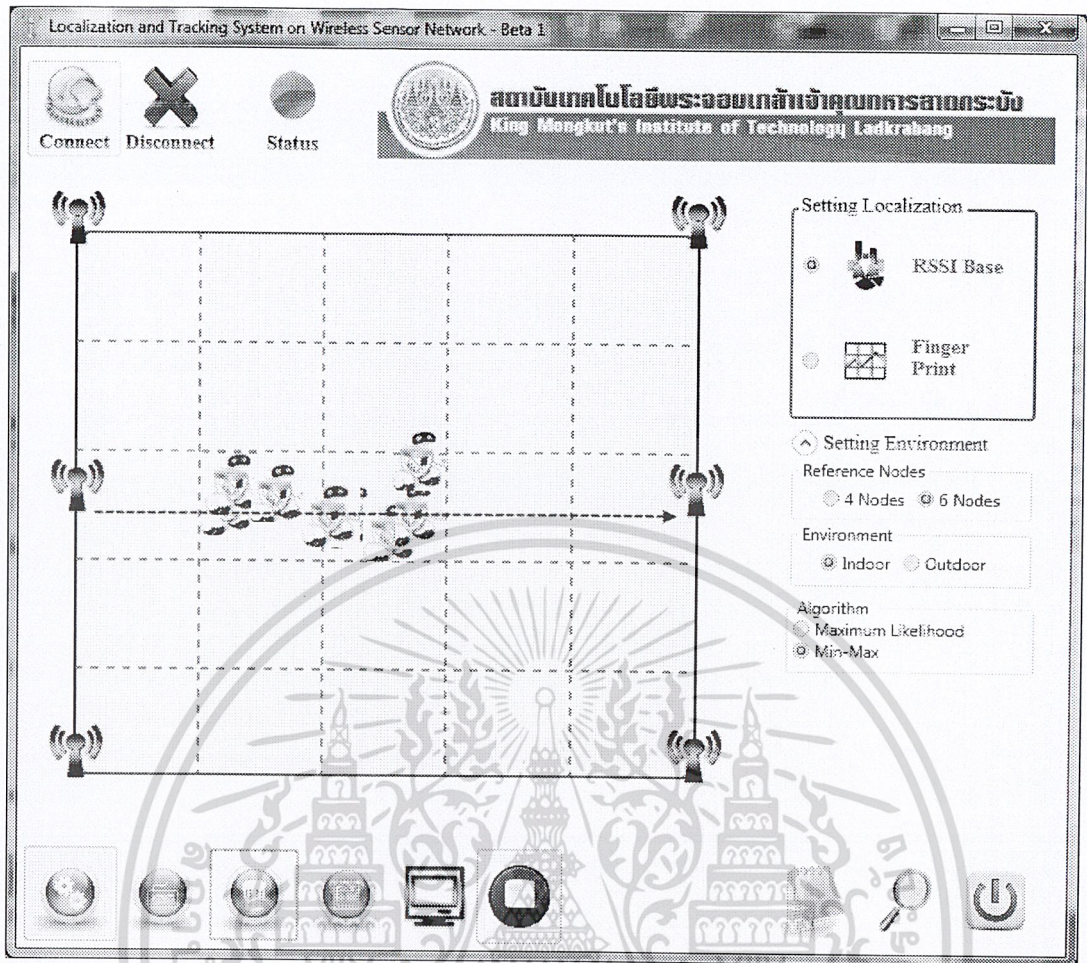


รูปที่ 4.88 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนว มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



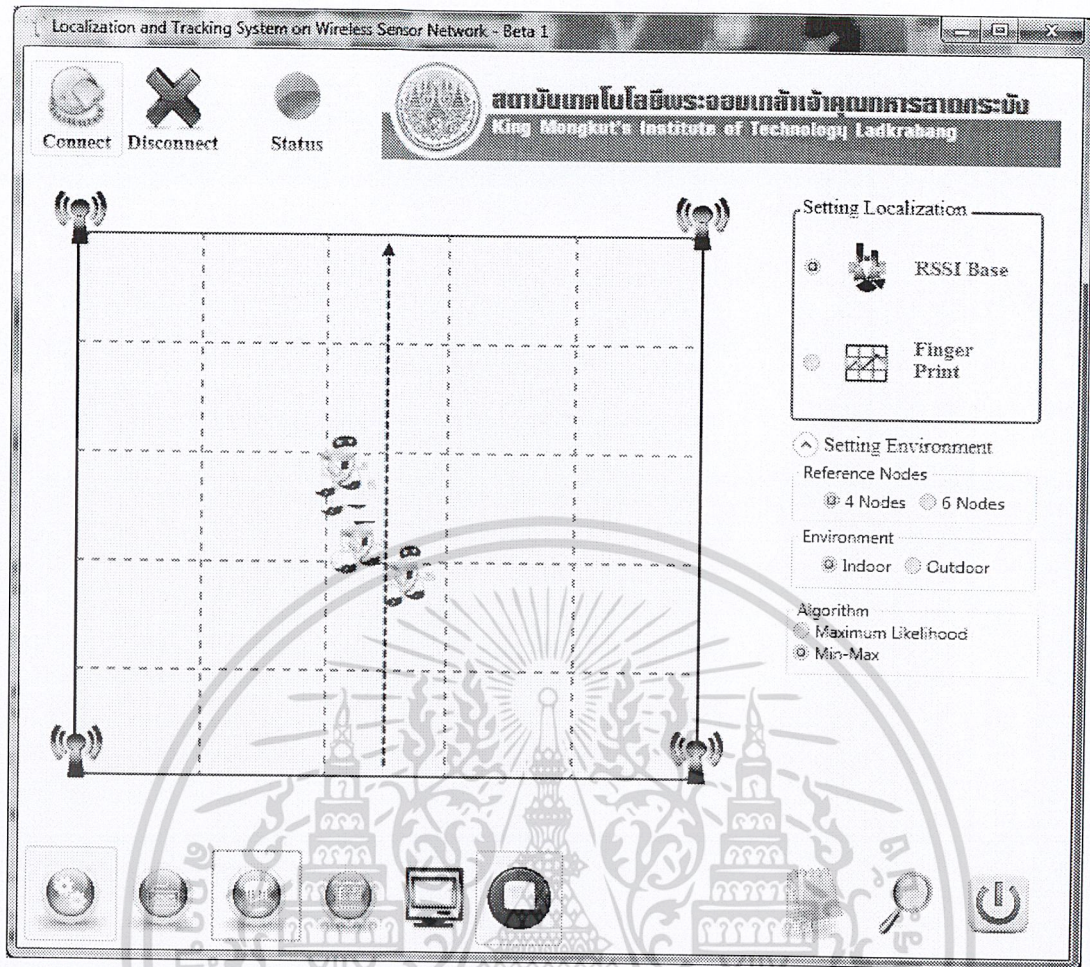
รูปที่ 4.89 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนวอนึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 105 จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



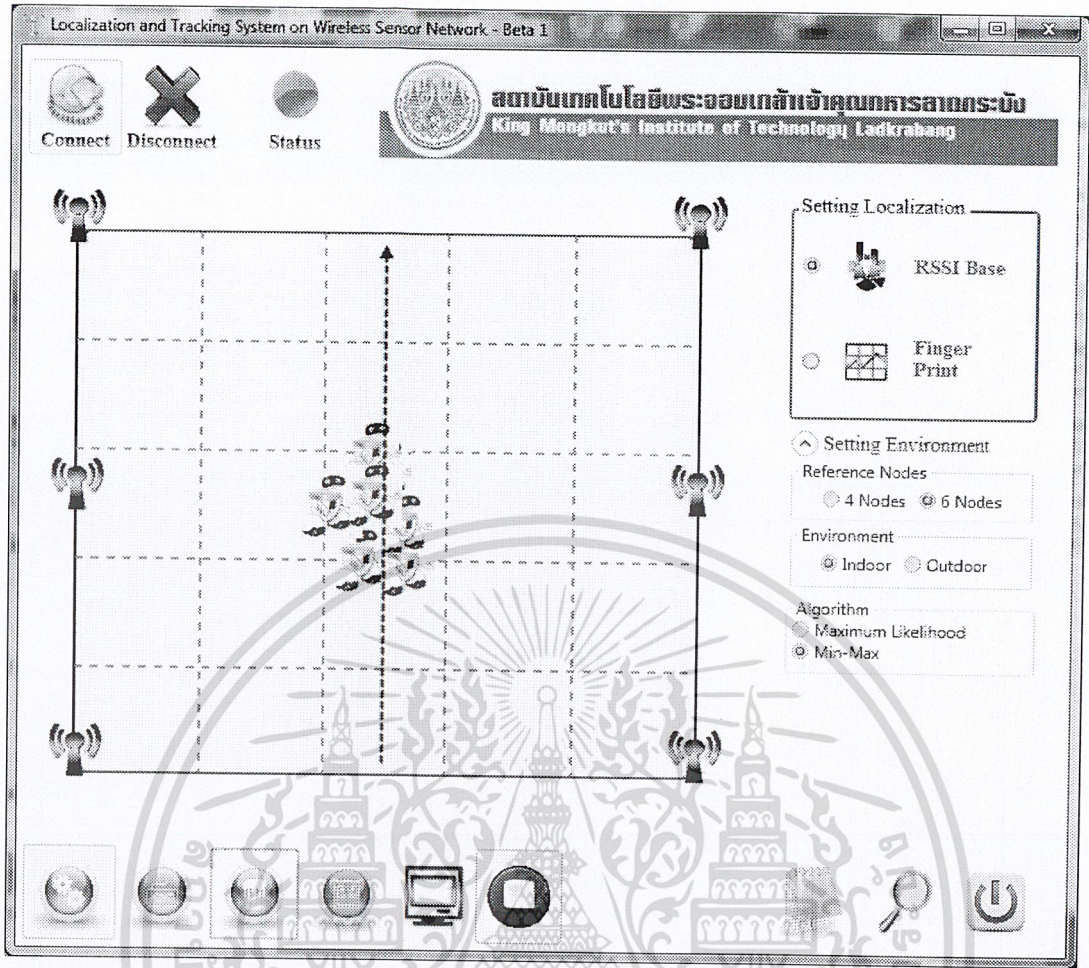
รูปที่ 4.90 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนวอนึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 106 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



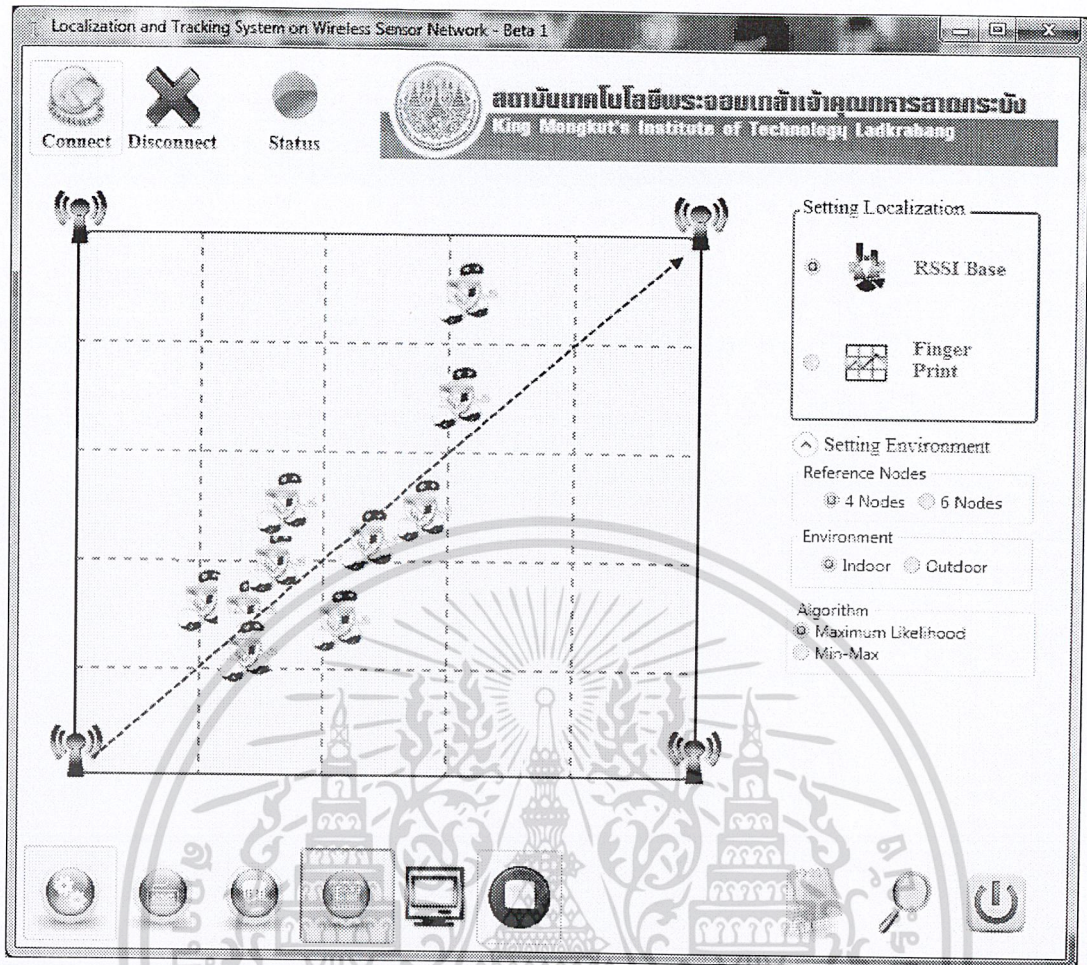
รูปที่ 4.91 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 107 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



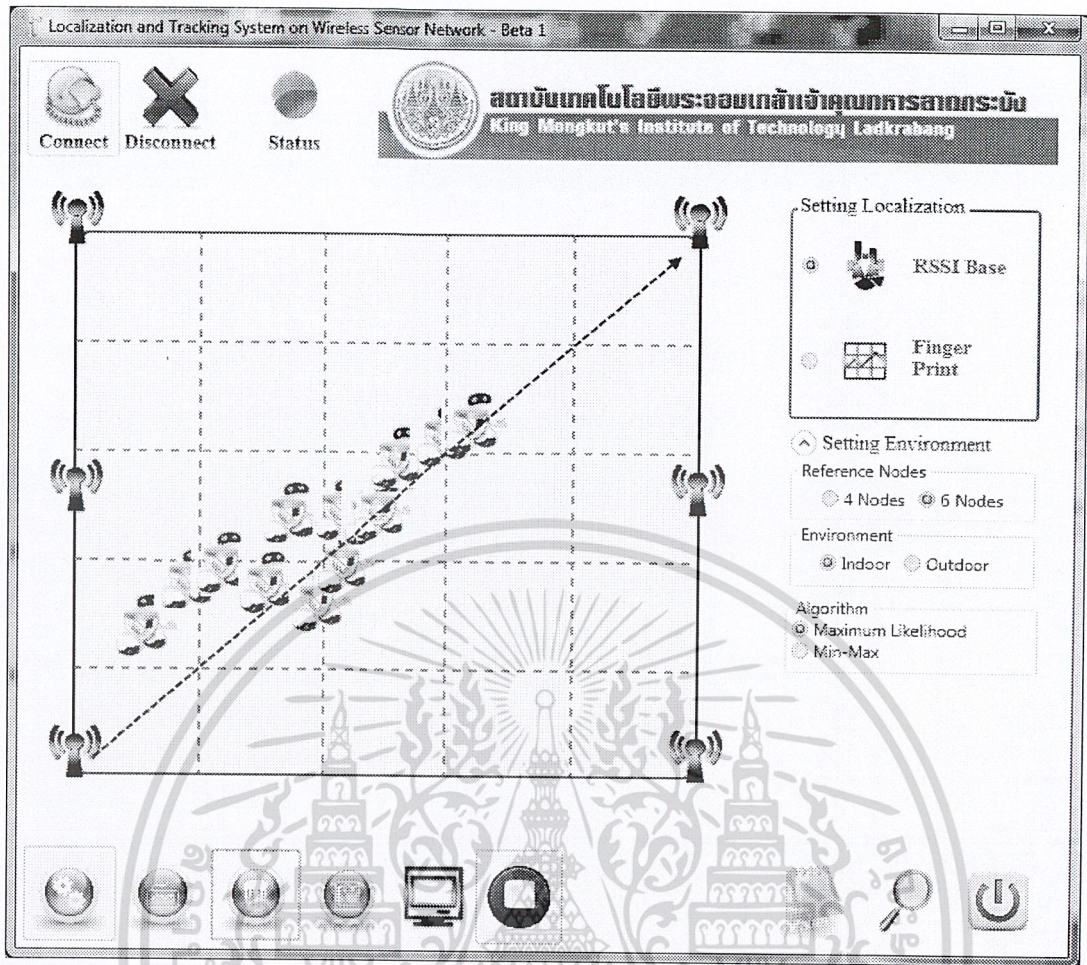
รูปที่ 4.92 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Maximum Likelihood Algorithm ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 108 จะต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

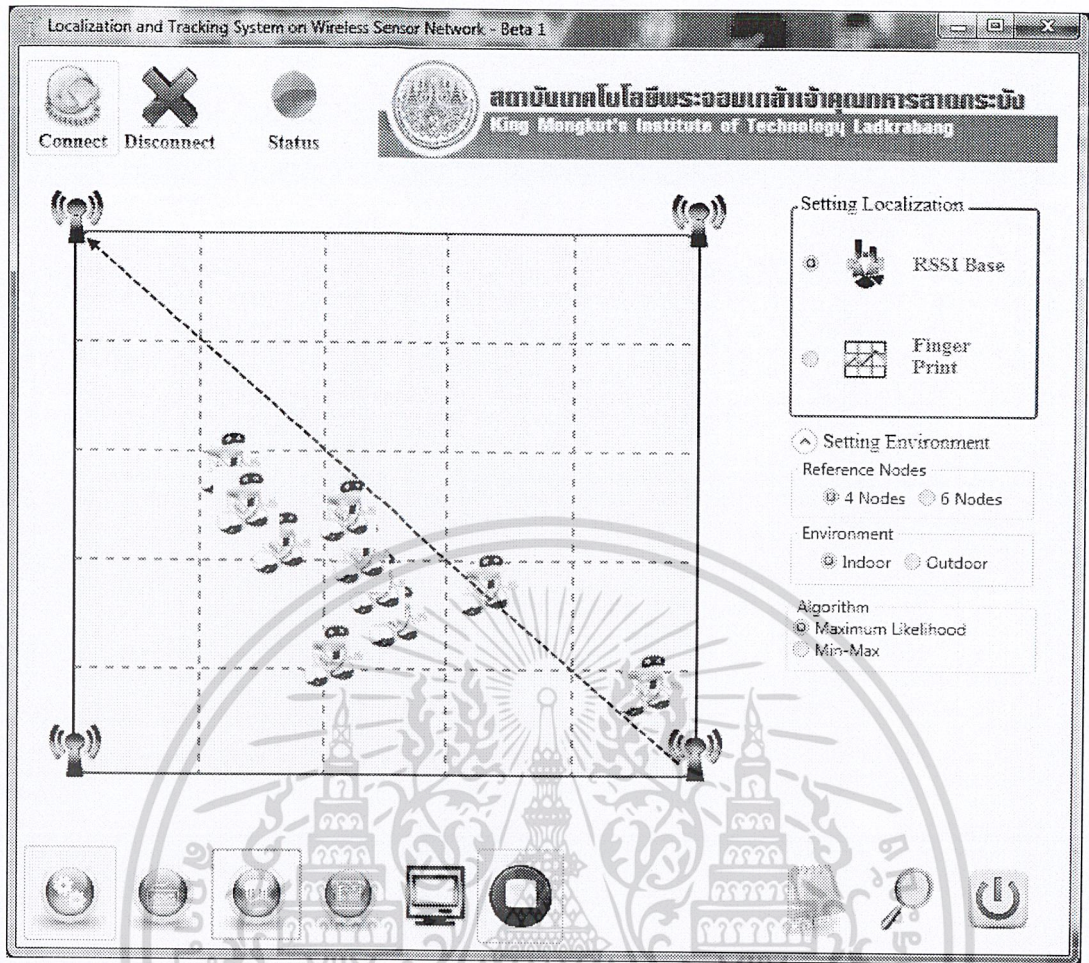


รูปที่ 4.93 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 109 ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

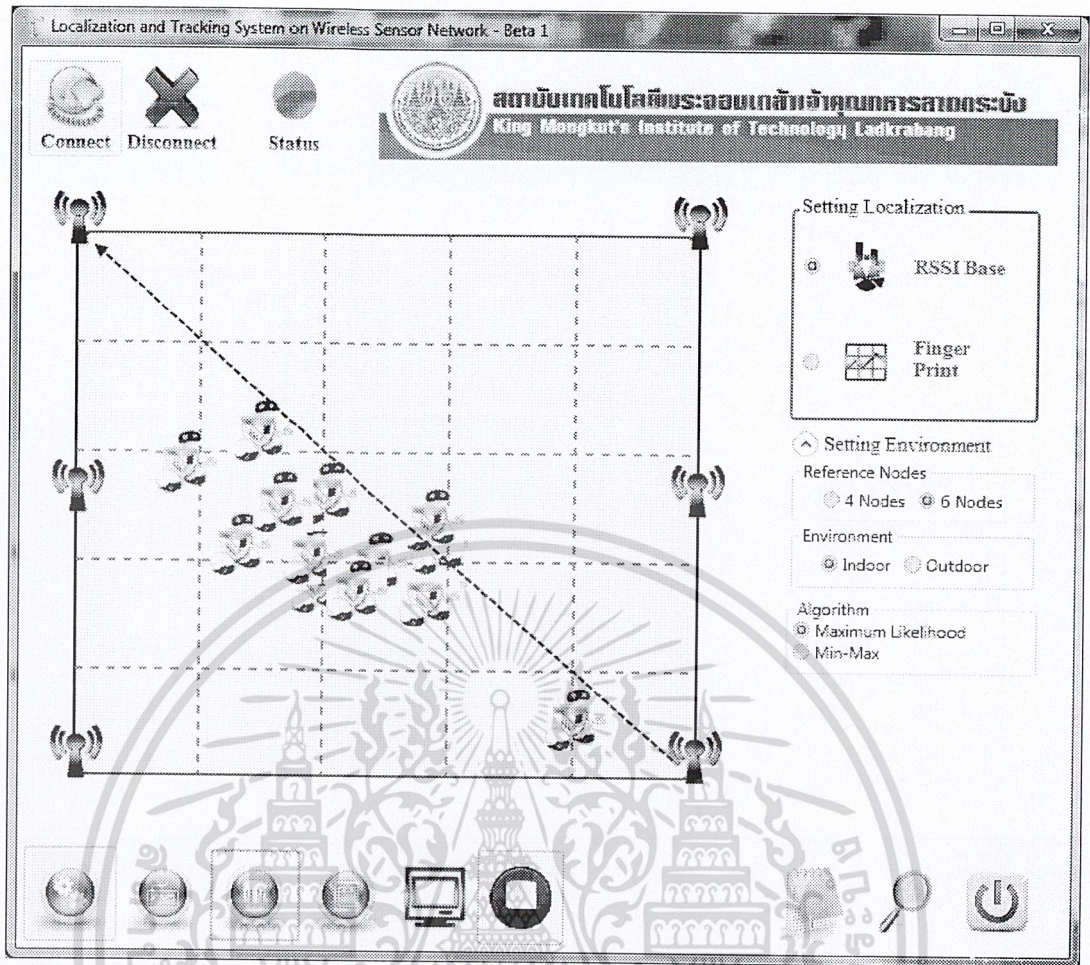


รูปที่ 4.94 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุม ด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



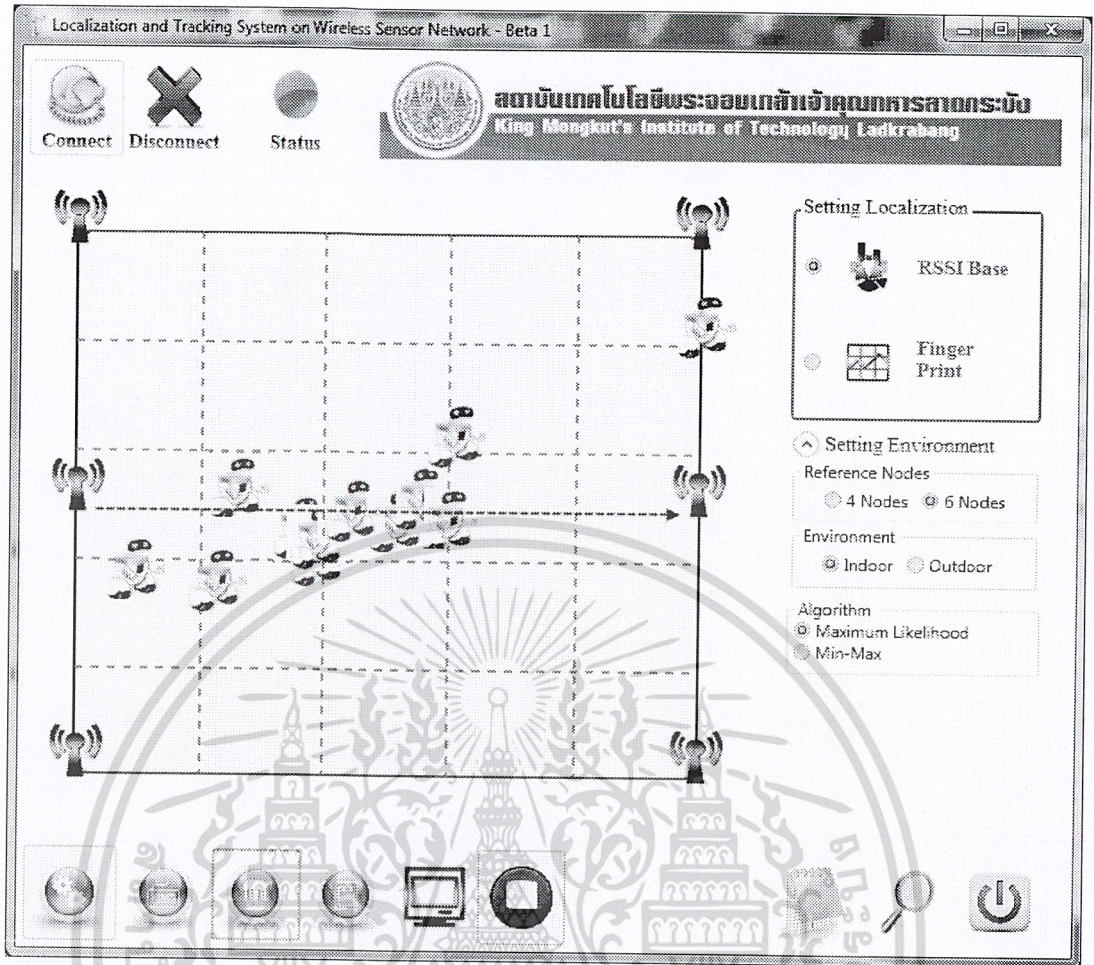
รูปที่ 4.95 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา111ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



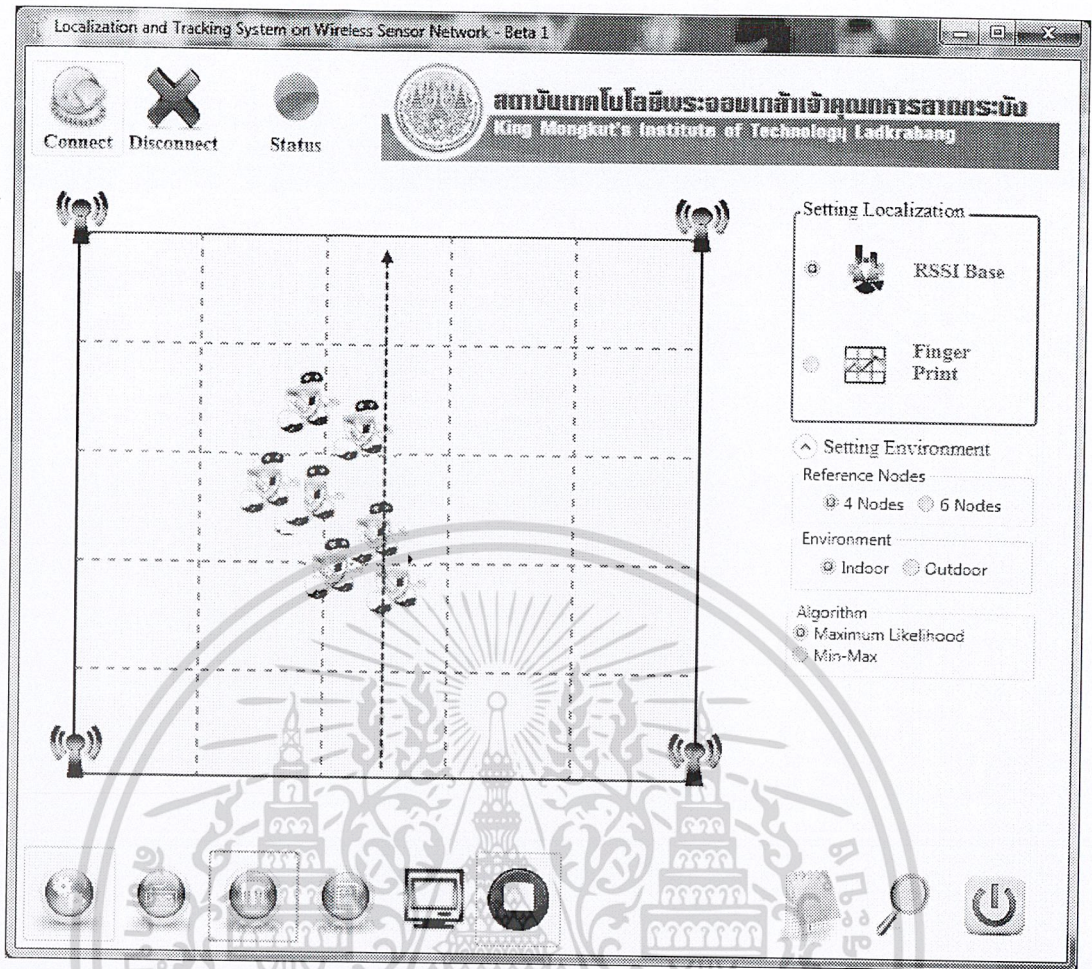
รูปที่ 4.96 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนว มุม ด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ 112 องค์กรอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



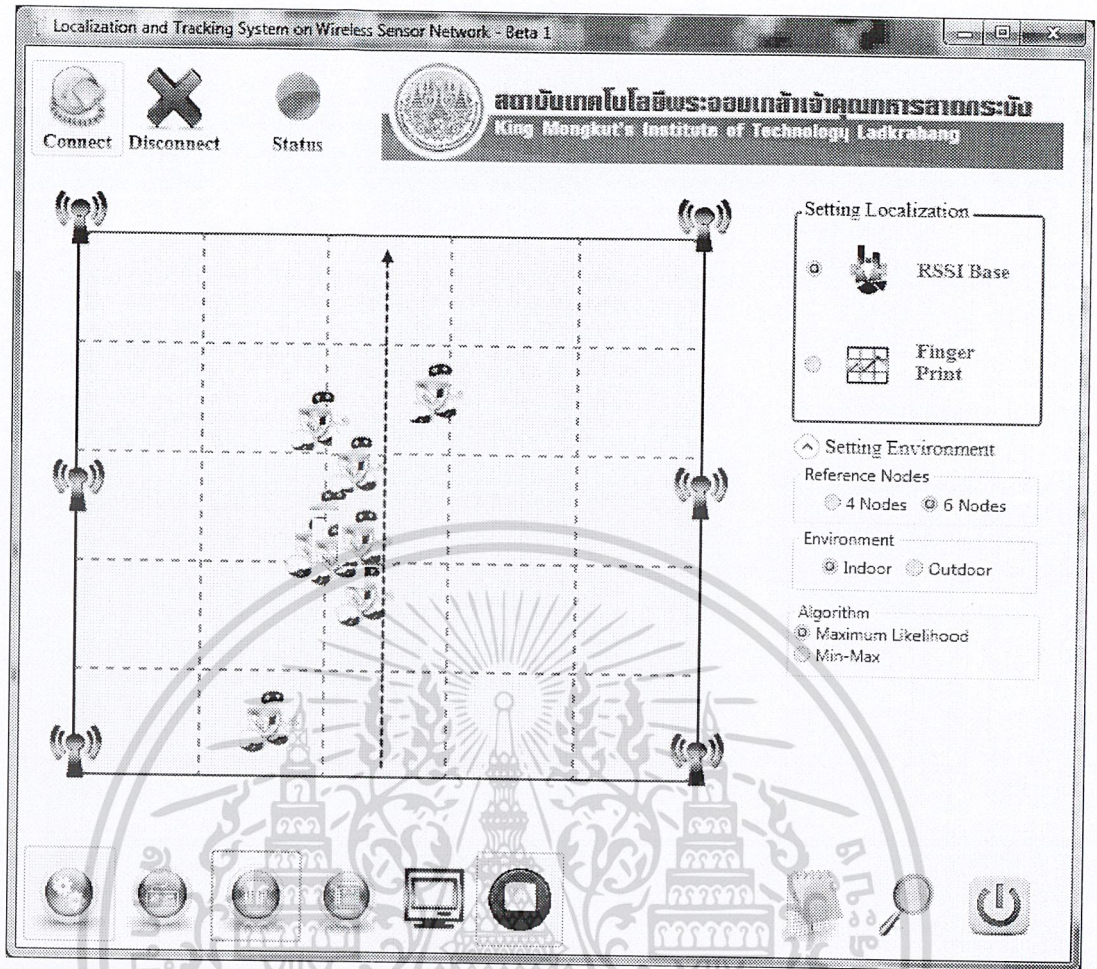
รูปที่ 4.98 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

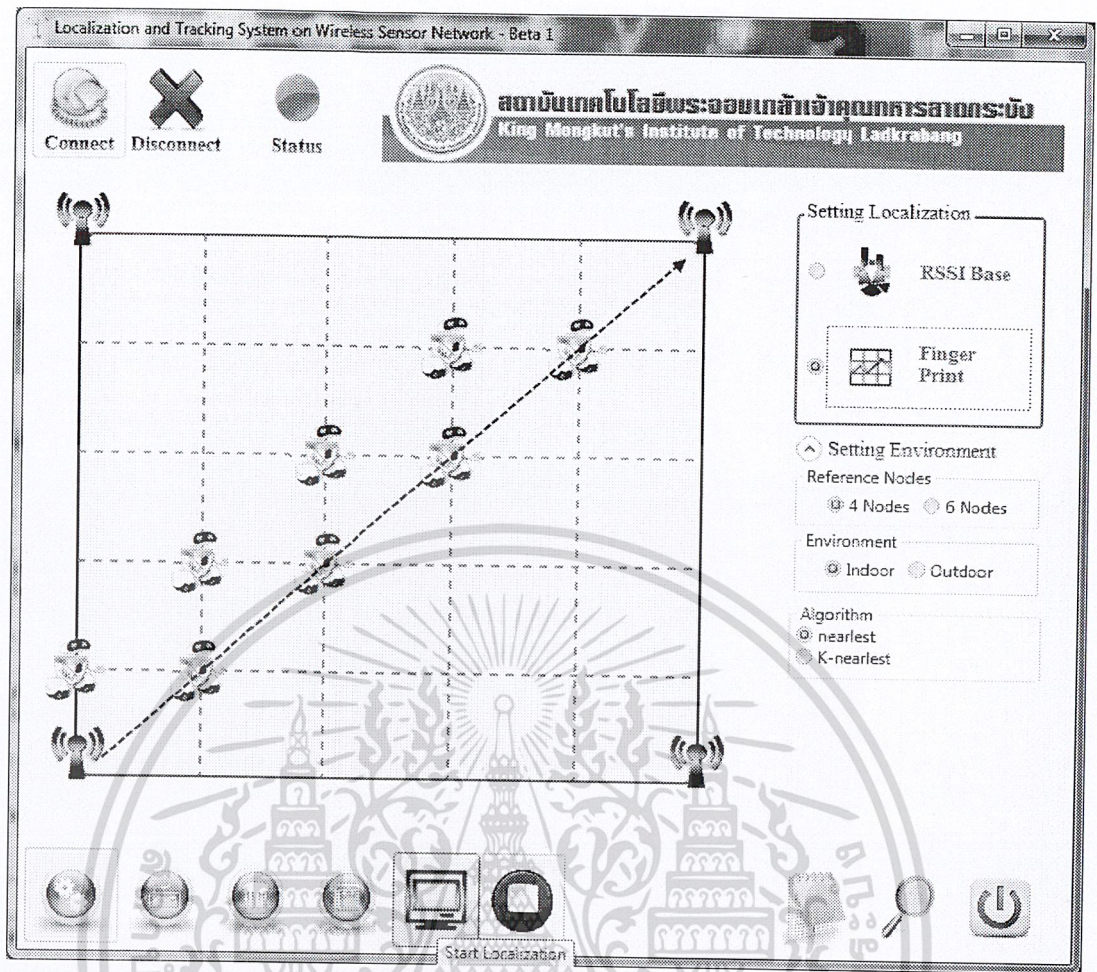


รูปที่ 4.99 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

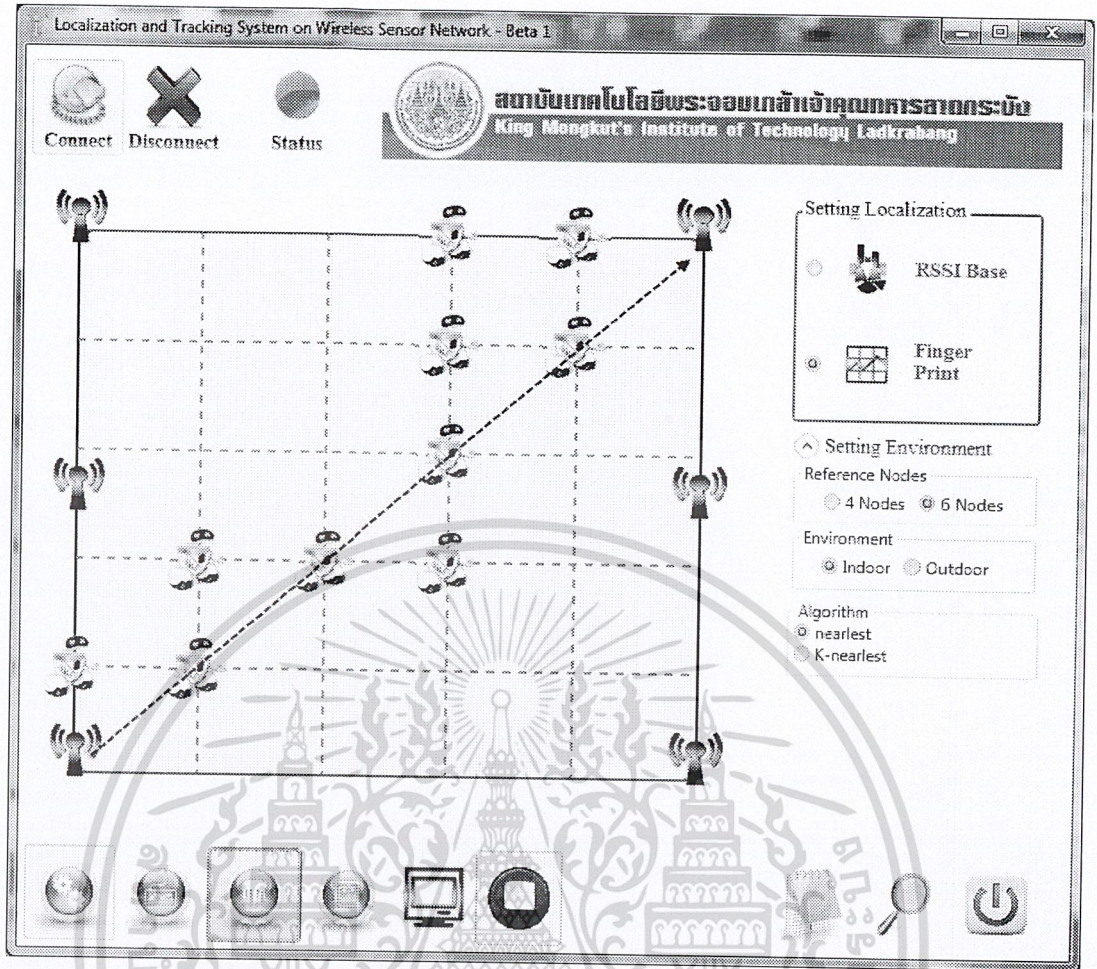
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา ๑๑๕ ้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.100 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Min-Max Algorithm ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

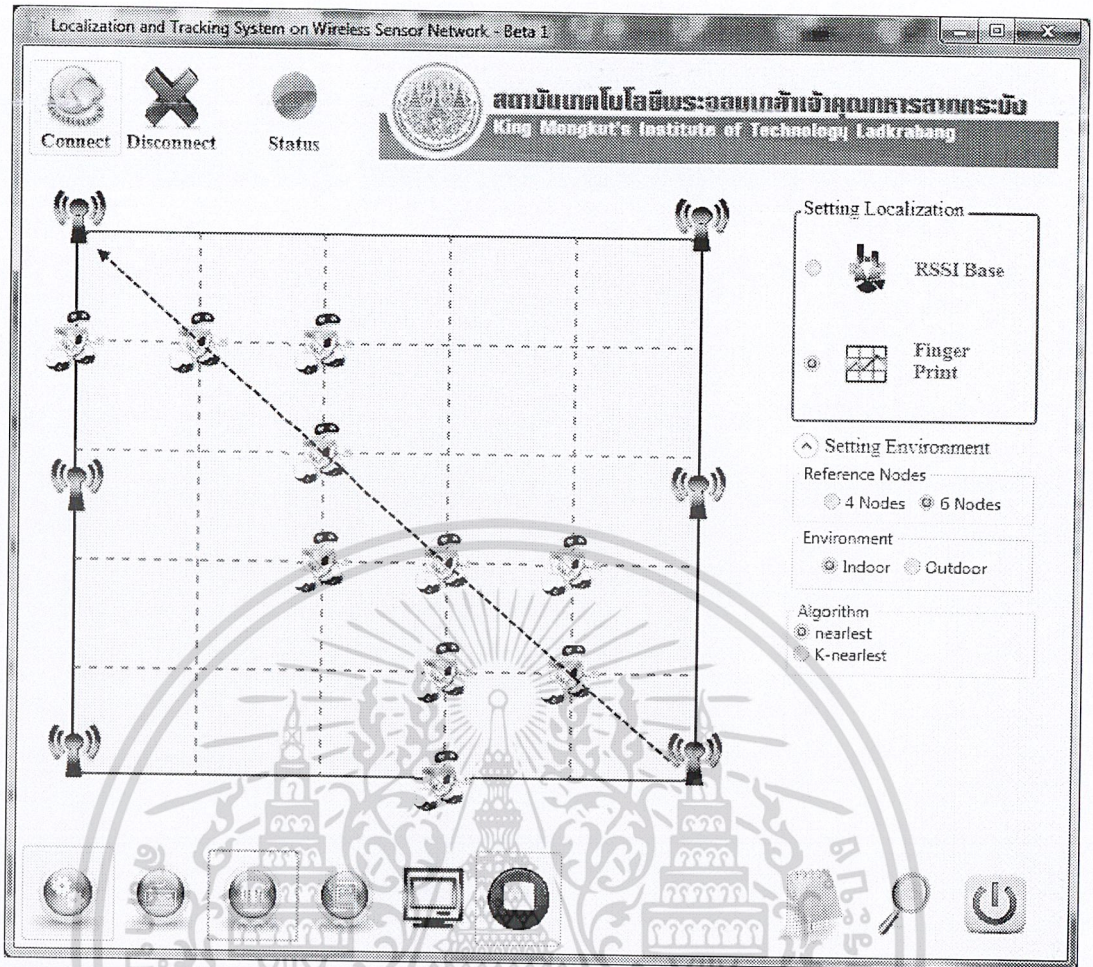


รูปที่ 4.101 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

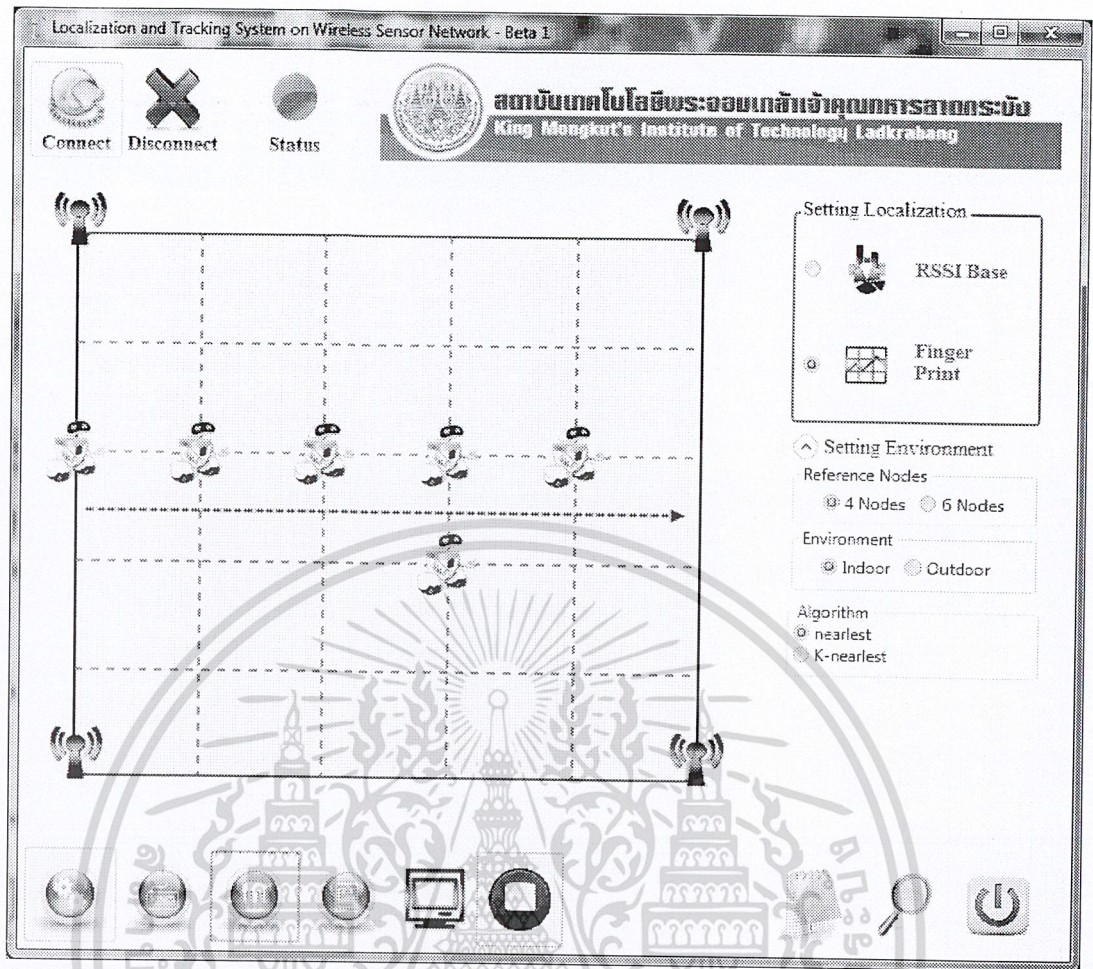


รูปที่ 4.102 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา ๑๑๘ ฮ่องกงอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

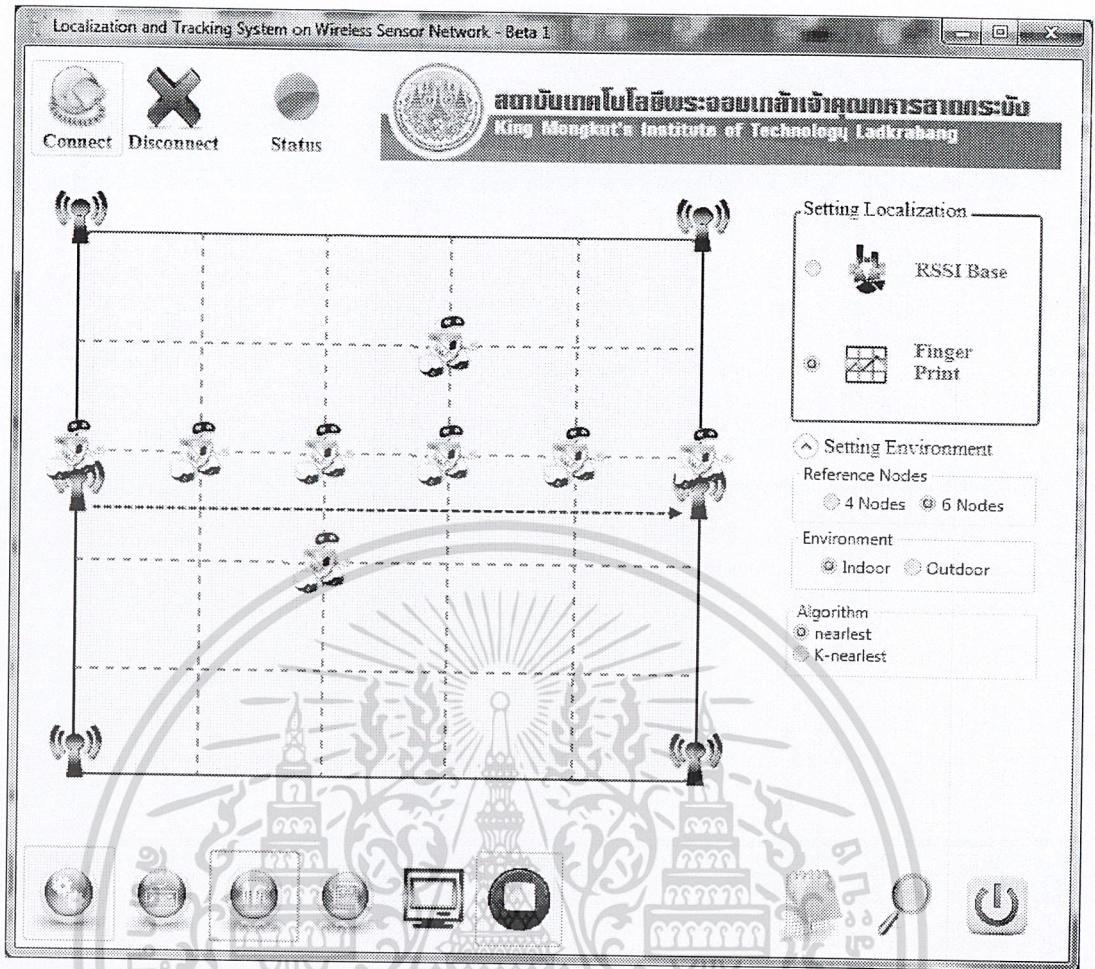


รูปที่ 4.104 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนว มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

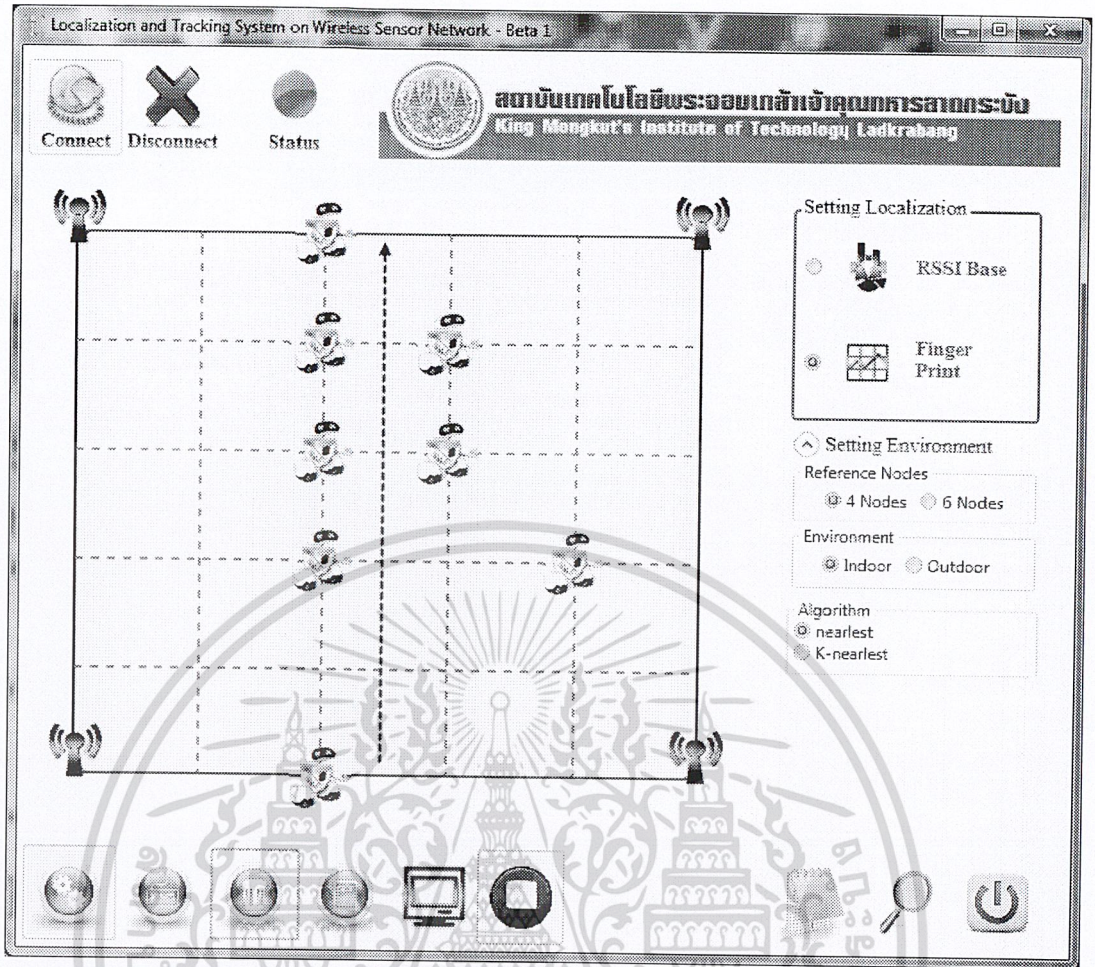


รูปที่ 4.105 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว

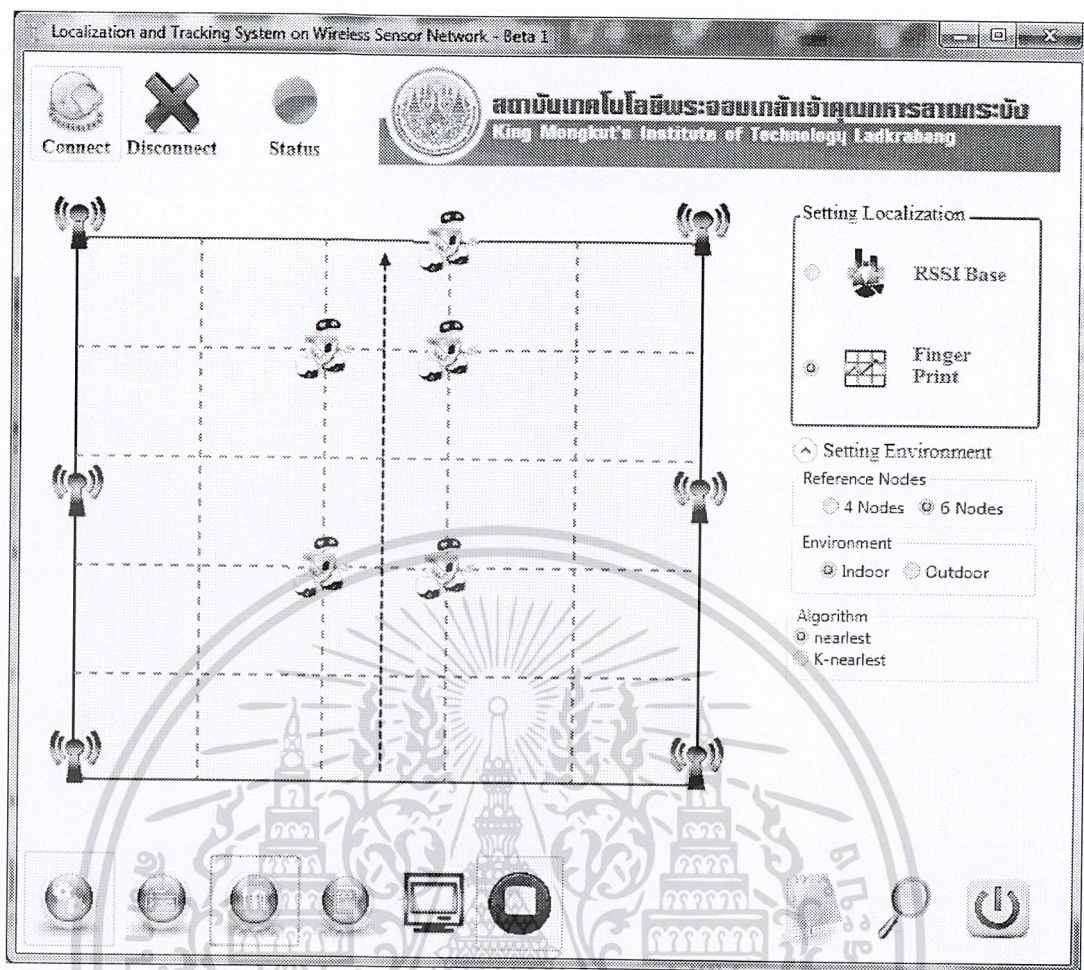
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และ121องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



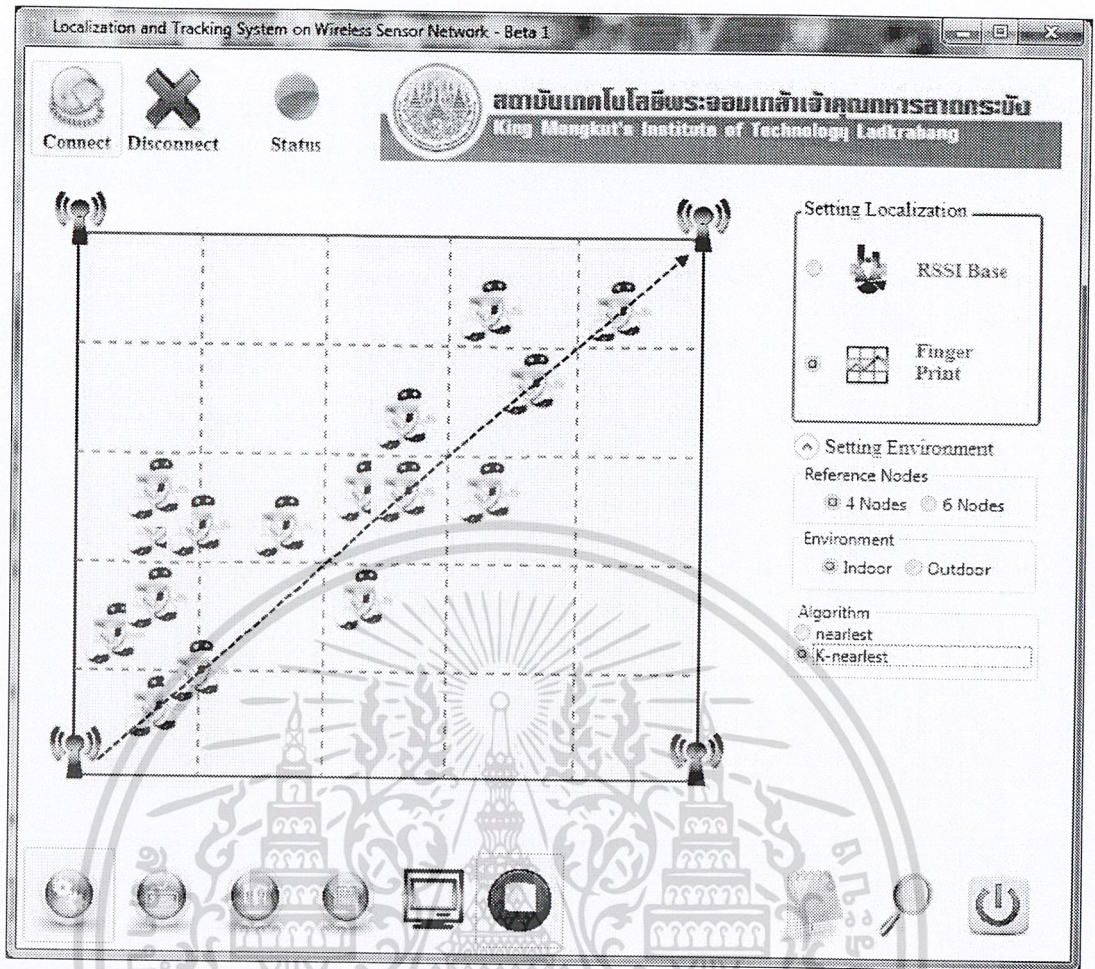
รูปที่ 4.106 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



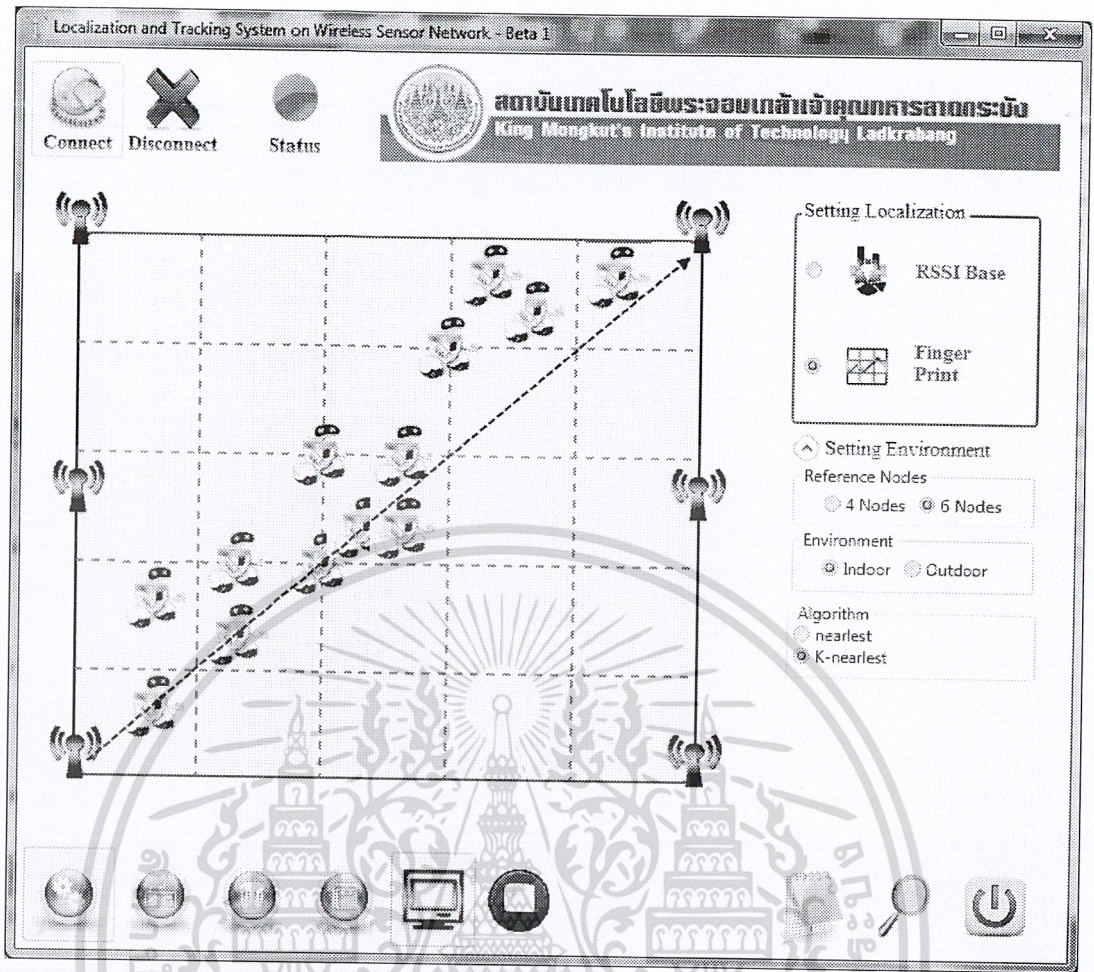
รูปที่ 4.107 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



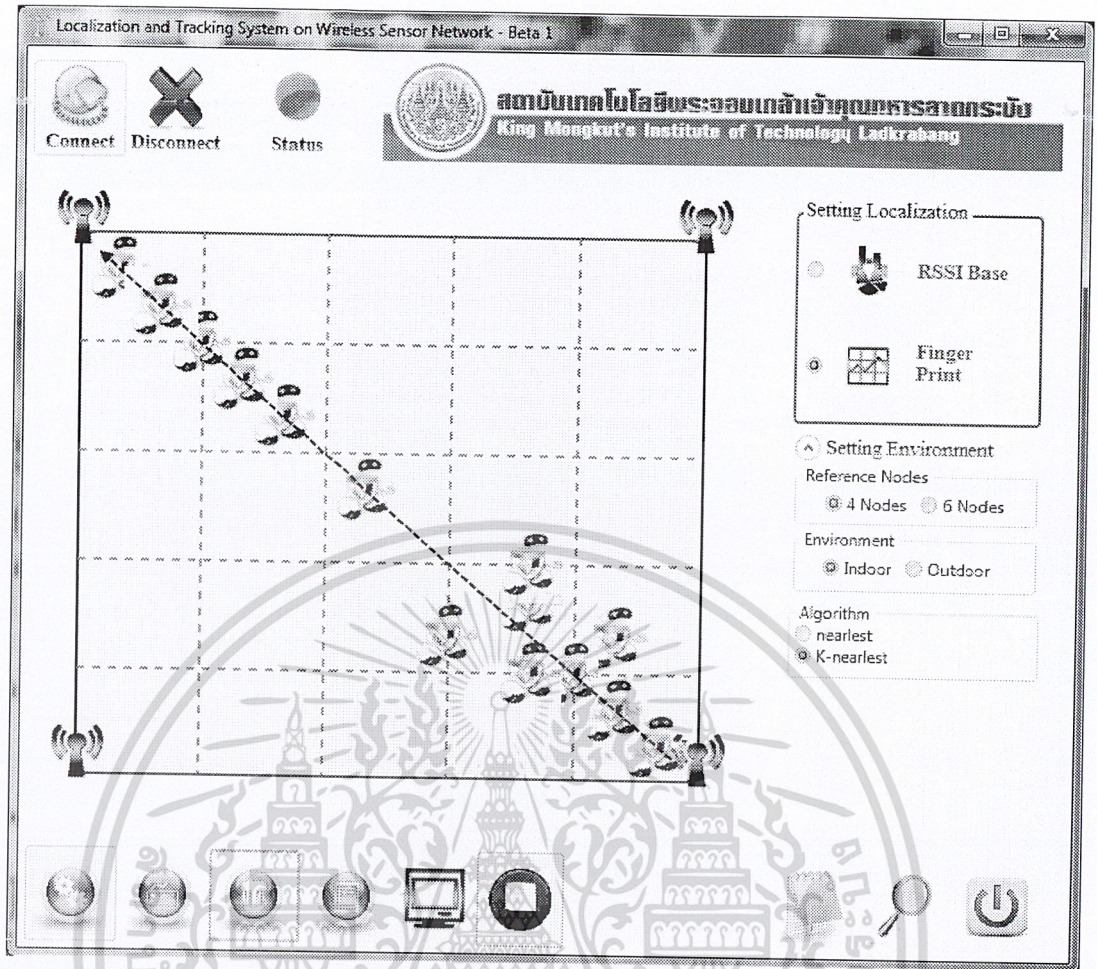
รูปที่ 4.108 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ Nearest ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



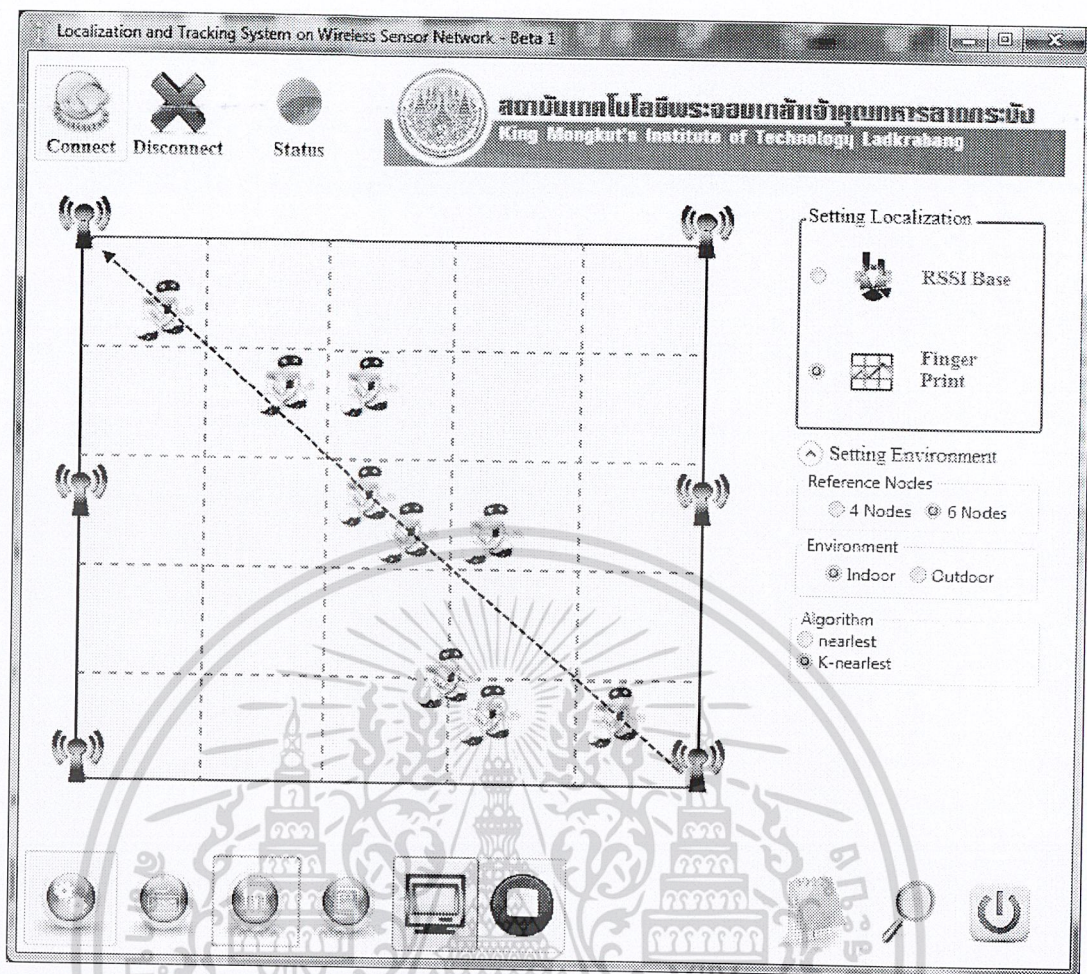
รูปที่ 4.109 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



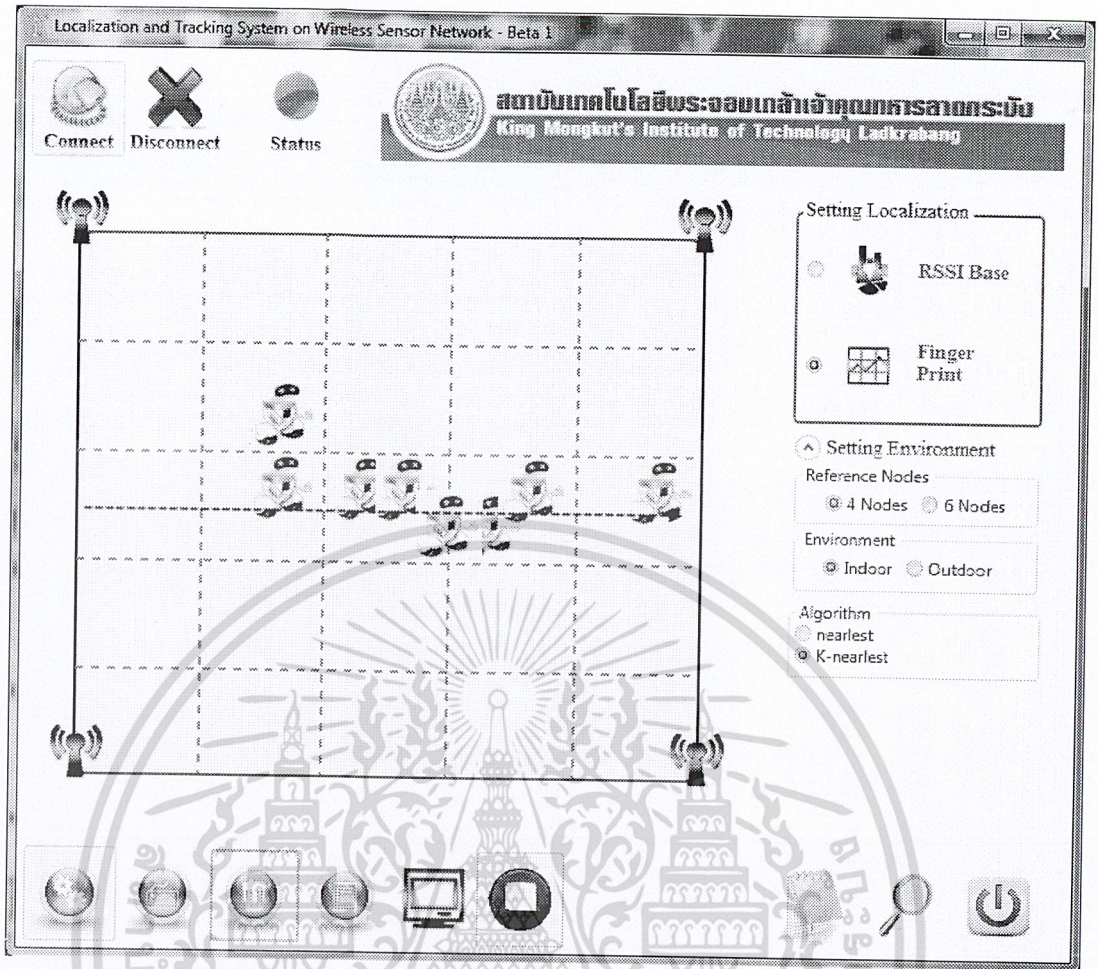
รูปที่ 4.110 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนว มุมด้านล่างซ้ายมือ ถึง มุมด้านบนขวามือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



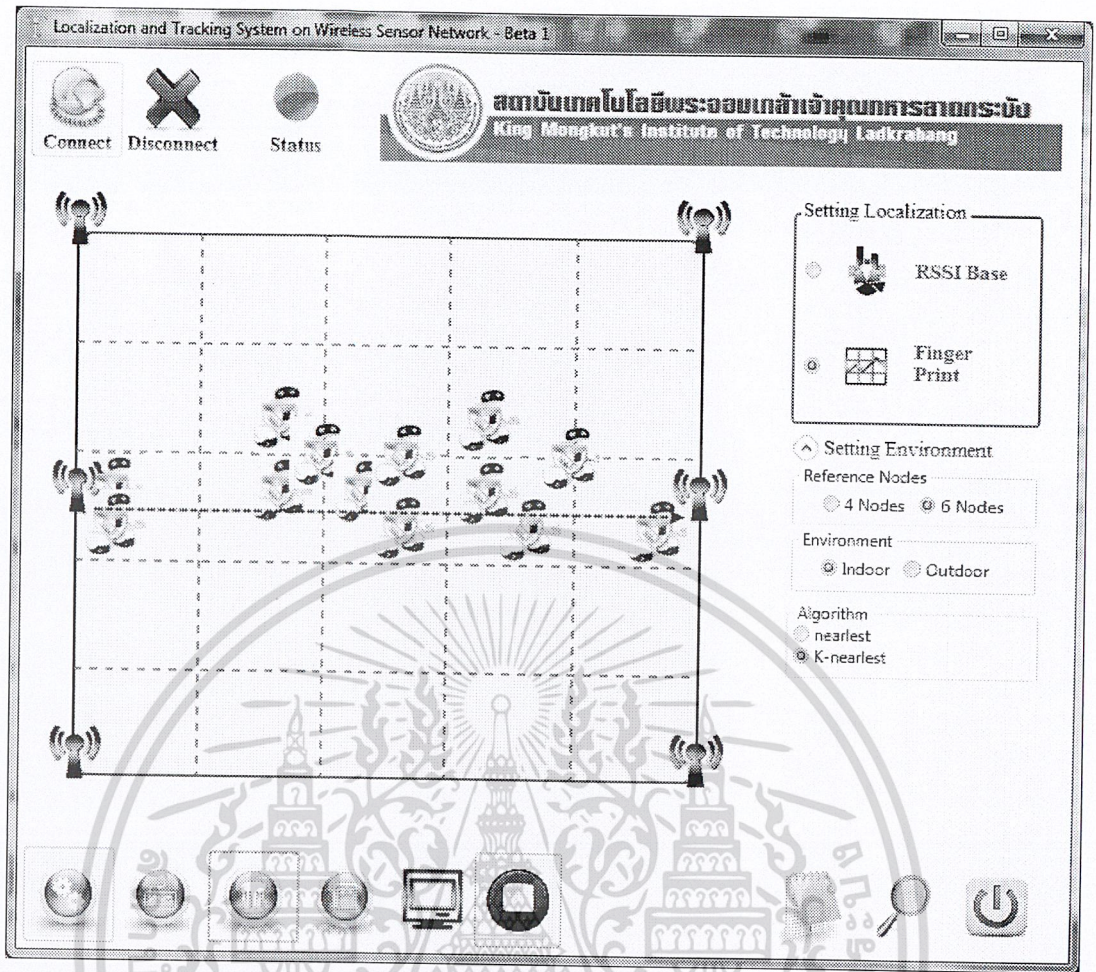
รูปที่ 4.111 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนว มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



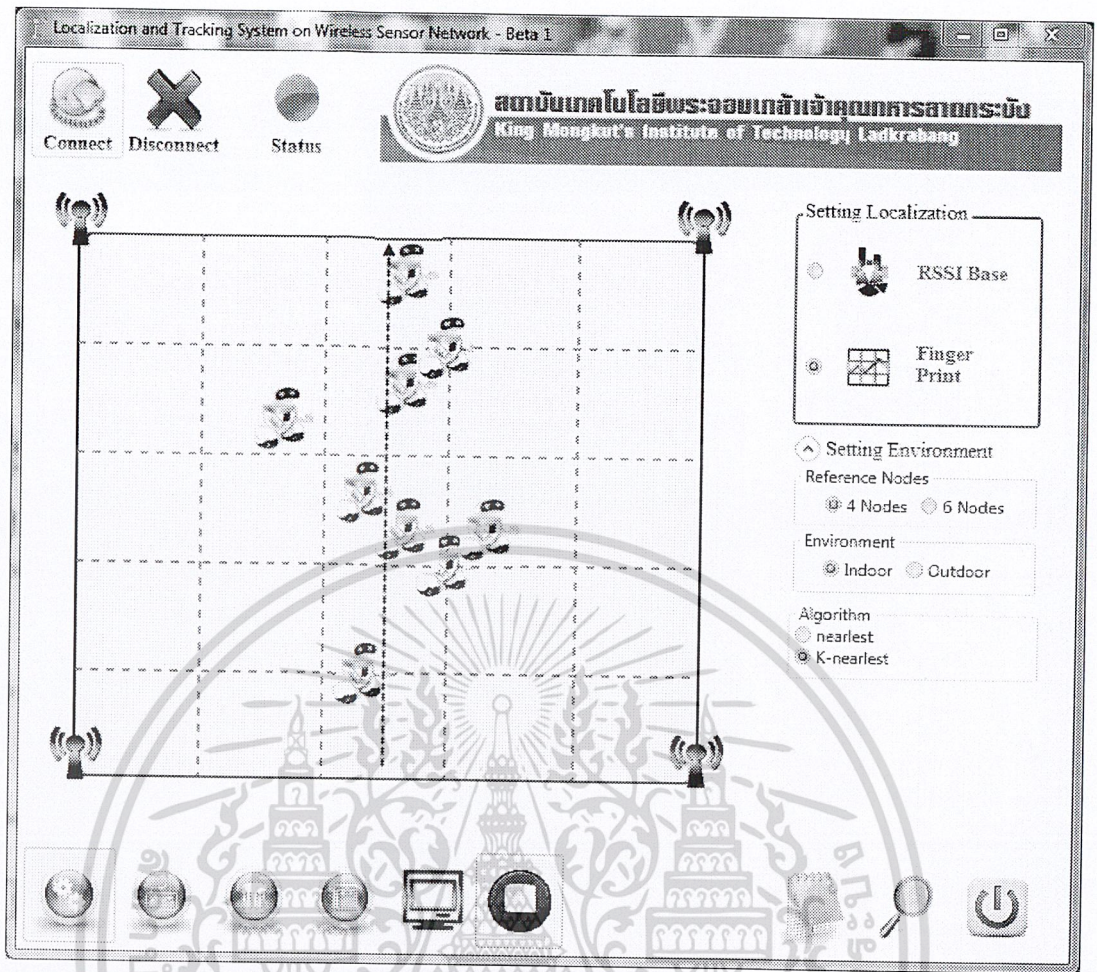
รูปที่ 4.112 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนว มุมด้านล่างขวามือ ถึง มุมด้านบนซ้ายมือ ของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



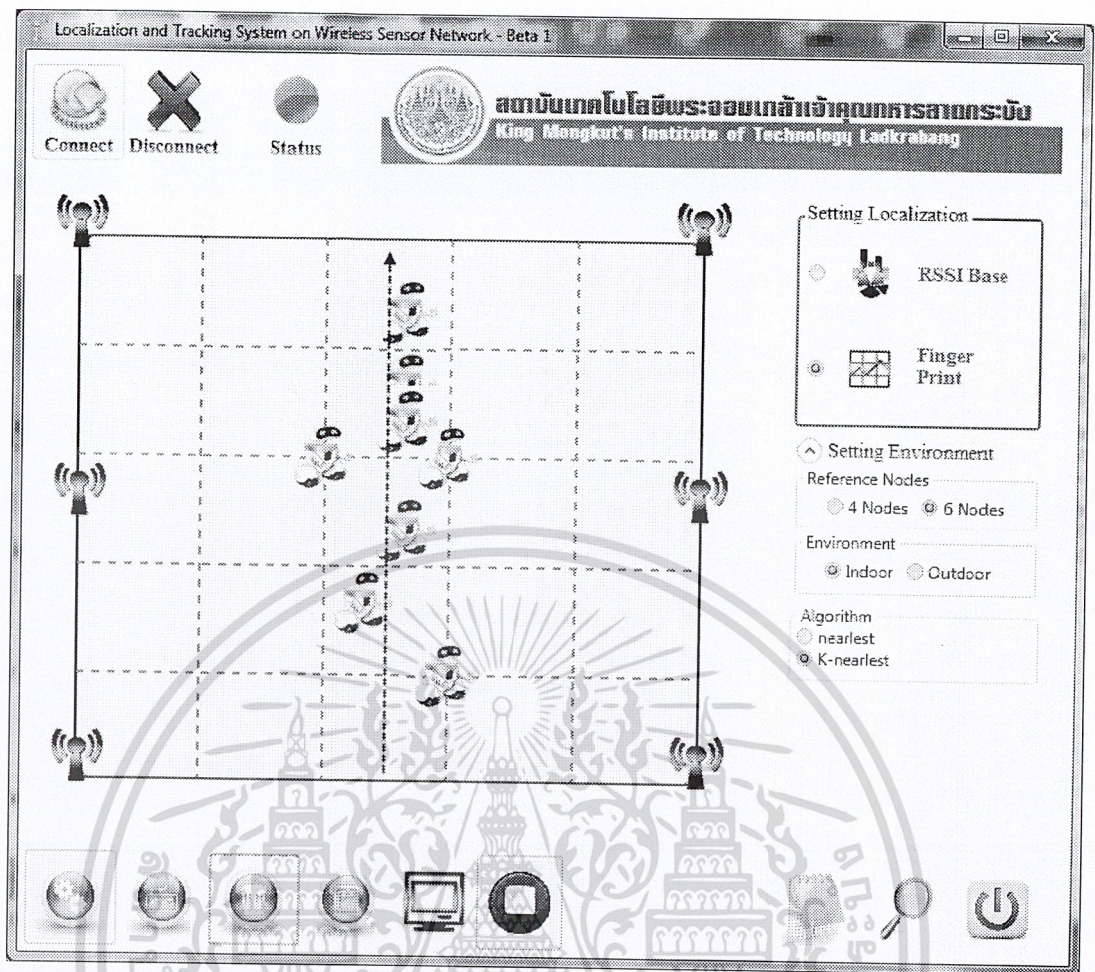
รูปที่ 4.113 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



รูปที่ 4.114 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวอนกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว



รูปที่ 4.115 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 4 ตัว



รูปที่ 4.116 การเคลื่อนที่ของวัตถุที่โปรแกรมคำนวณได้ โดยใช้ Fingerprint Algorithm แบบ 3-Nearest ตามแนวตั้งกึ่งกลางของพื้นที่การทดลอง โดยใช้ตัวรับ 6 ตัว

4.8 บทสรุป

ในบทนี้ได้กล่าวถึง โครงสร้างของระบบ ทั้งในส่วนฮาร์ดแวร์ การเชื่อมต่อและการทำงาน Graphic User Interface รวมไปถึงแนวทางและวิธีการทดลองการระบุตำแหน่งทั้งในแบบหยุดนิ่ง และแบบเคลื่อนที่ โดยใช้วิธีการระบุตำแหน่งแบบ Range-Based และ Fingerprint

บทที่ 5

บทวิจารณ์และสรุปผล

5.1 บทสรุป

จากการทดลองสามารถสรุปได้เป็น 5 กรณี ดังนี้

5.1.1 กรณีการเคลื่อนที่จากซ้ายล่างไปยังขวาบนของพื้นที่

สามารถแสดงค่าความผิดพลาดการระบุตำแหน่งได้ดังตารางที่ 5.1 และแสดงค่าประสิทธิภาพของการระบุตำแหน่งได้ดังตารางที่ 5.2 โดยค่าประสิทธิภาพของการระบุตำแหน่ง สามารถพิจารณาได้จากค่าความผิดพลาดการระบุตำแหน่ง ถ้ามีค่าความผิดพลาดการระบุตำแหน่งน้อย ก็จะมีประสิทธิภาพในการระบุตำแหน่งมาก แต่ถ้ามีค่าความผิดพลาดการระบุตำแหน่งมาก ก็จะมีประสิทธิภาพในการระบุตำแหน่งน้อย

ตารางที่ 5.1 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่ง กรณีการเคลื่อนที่จากซ้ายล่างไปยังขวาบนของพื้นที่

Localization Algorithms	4 Nodes (m)	6 Nodes (m)
Maximum Likelihood	1.31	1.12
Min-Max	0.80	0.80
Location Fingerprint(Nearest)	0.56	0.44
Location Fingerprint(3-Nearest)	0.54	0.38

ตารางที่ 5.2 ตารางแสดงประสิทธิภาพของการระบุตำแหน่ง กรณีการเคลื่อนที่จากซ้ายล่างไปยังขวาบนของพื้นที่

4 Nodes	3-Nearest > Nearest > Min-Max > Maximum Likelihood
6 Nodes	3-Nearest > Nearest > Min-Max > Maximum Likelihood

5.1.2 กรณีการเคลื่อนที่จากขวาล่างไปยังซ้ายบนของพื้นที่

สามารถแสดงค่าความผิดพลาดการระบุตำแหน่งได้ดังตารางที่ 5.3 และแสดงค่าประสิทธิภาพของการระบุตำแหน่งได้ดังตารางที่ 5.4 โดยค่าประสิทธิภาพของการระบุตำแหน่ง สามารถพิจารณาได้จากค่าความผิดพลาดการระบุตำแหน่ง ถ้ามีค่าความผิดพลาดการระบุตำแหน่งน้อย ก็จะมีประสิทธิภาพในการระบุตำแหน่งมาก แต่ถ้ามีค่าความผิดพลาดการระบุตำแหน่งมาก ก็จะมีประสิทธิภาพในการระบุตำแหน่งน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.3 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่จากขวาล่างไป ยังซ้ายบนของพื้นที่

Localization Algorithms	4 Nodes (m)	6 Nodes (m)
Maximum Likelihood	1.25	1.14
Min-Max	0.71	0.78
Location Fingerprint(Nearest)	0.72	0.53
Location Fingerprint(3-Nearest)	0.49	0.43

ตารางที่ 5.4 ตารางแสดงประสิทธิภาพของการระบุตำแหน่ง กรณีการเคลื่อนที่จากขวาล่างไปยังซ้ายบนของพื้นที่

4 Nodes	3-Nearest > Min-Max > Nearest > Maximum Likelihood
6 Nodes	3-Nearest > Nearest > Min-Max > Maximum Likelihood

5.1.3 กรณีการเคลื่อนที่ตามแนวนอนของพื้นที่

สามารถแสดงค่าความผิดพลาดการระบุตำแหน่งได้ดังตารางที่ 5.5 และแสดงค่าประสิทธิภาพของการระบุตำแหน่งได้ดังตารางที่ 5.6 โดยค่าประสิทธิภาพของการระบุตำแหน่ง สามารถพิจารณาได้จากค่าความผิดพลาดการระบุตำแหน่ง ถ้ามีค่าความผิดพลาดการระบุตำแหน่งน้อย ก็จะมีประสิทธิภาพในการระบุตำแหน่งมาก แต่ถ้ามีค่าความผิดพลาดการระบุตำแหน่งมาก ก็จะมีประสิทธิภาพในการระบุตำแหน่งน้อย

ตารางที่ 5.5 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวนอนพื้นที่

Localization Algorithms	4 Nodes (m)	6 Nodes (m)
Maximum Likelihood	0.83	0.84
Min-Max	0.75	0.60
Location Fingerprint(Nearest)	0.77	0.65
Location Fingerprint(3-Nearest)	0.61	0.41

ตารางที่ 5.6 ตารางแสดงประสิทธิภาพของการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวนอนพื้นที่

4 Nodes	3-Nearest > Min-Max > Nearest > Maximum Likelihood
6 Nodes	3-Nearest > Min-Max > Nearest > Maximum Likelihood

5.1.4 กรณีการเคลื่อนที่ตามแนวตั้งของพื้นที่

สามารถแสดงค่าความผิดพลาดการระบุตำแหน่ง ได้ดังตารางที่ 5.7 และแสดงค่าประสิทธิภาพของการระบุตำแหน่ง ได้ดังตารางที่ 5.8 โดยค่าประสิทธิภาพของการระบุตำแหน่ง สามารถพิจารณาได้จากค่าความผิดพลาดการระบุตำแหน่ง ถ้ามีค่าความผิดพลาดการระบุตำแหน่งน้อย ก็จะมีประสิทธิภาพในการระบุตำแหน่งมาก แต่ถ้ามีค่าความผิดพลาดการระบุตำแหน่งมาก ก็จะมีประสิทธิภาพในการระบุตำแหน่งน้อย

ตารางที่ 5.7 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวตั้งของพื้นที่

Localization Algorithms	4 Nodes (m)	6 Nodes (m)
Maximum Likelihood	0.89	0.70
Min-Max	0.79	0.75
Location Fingerprint(Nearest)	0.75	0.70
Location Fingerprint(3-Nearest)	0.58	0.44

ตารางที่ 5.8 ตารางแสดงประสิทธิภาพของการระบุตำแหน่งกรณีการเคลื่อนที่ตามแนวตั้งพื้นที่

4 Nodes	3-Nearest > Nearest > Min-Max > Maximum Likelihood
6 Nodes	3-Nearest > Maximum Likelihood > Nearest > Min-Max

5.1.5 กรณีการเคลื่อนที่ตามแนวพิกัดที่กำหนดไว้ของพื้นที่

สามารถแสดงค่าความผิดพลาดการระบุตำแหน่ง ได้ดังตารางที่ 5.9 และแสดงค่าประสิทธิภาพของการระบุตำแหน่ง ได้ดังตารางที่ 5.10 โดยค่าประสิทธิภาพของการระบุตำแหน่ง สามารถพิจารณาได้จากค่าความผิดพลาดการระบุตำแหน่ง ถ้ามีค่าความผิดพลาดการระบุตำแหน่งน้อย ก็จะมีประสิทธิภาพในการระบุตำแหน่งมาก แต่ถ้ามีค่าความผิดพลาดการระบุตำแหน่งมาก ก็จะมีประสิทธิภาพในการระบุตำแหน่งน้อย

ตารางที่ 5.9 ตารางแสดงค่าความผิดพลาดการระบุตำแหน่งกรณีการเคลื่อนที่ตามพิกัดที่กำหนดไว้ของพื้นที่

Localization Algorithms	4 Nodes (m)	6 Nodes (m)
Maximum Likelihood	1.44	1.38
Min-Max	0.97	1.00
Location Fingerprint(Nearest)	0.45	0.22
Location Fingerprint(3-Nearest)	0.63	0.47

ตารางที่ 5.10 ตารางแสดงประสิทธิภาพของการระบุตำแหน่งกรณีการเคลื่อนที่ตามพิกัดที่กำหนดไว้ของพื้นที่

4 Nodes	Nearest > 3-Nearest > Min-Max > Maximum Likelihood
6 Nodes	Nearest > 3-Nearest > Min-Max > Maximum Likelihood

5.1.6 สรุป

การทดลองการระบุตำแหน่งแบบเคลื่อนที่

1. โดยรวมแล้ว วิธีการระบุตำแหน่งแบบ 3-Nearest จะมีประสิทธิภาพมากที่สุด
2. ในพิกัดที่ตรงกับตำแหน่งที่ได้ทำการเก็บค่าไว้ การระบุตำแหน่งแบบ Nearest จะมีประสิทธิภาพ สูงกว่า
3. ในบริเวณอื่นๆ ที่ไม่ตรงกับตำแหน่งที่เก็บค่าไว้ Nearest กับ Min-Max จะมีประสิทธิภาพใกล้เคียงกัน
4. ในกรณีการระบุตำแหน่งที่เป็น Range-based (Maximum Likelihood , Min-Max) ตำแหน่งที่ใกล้กับตัวรับหนึ่งๆ จะมีความผิดพลาดสูง และจะมีความแม่นยำสูงขึ้นเมื่ออยู่ในบริเวณตรงกลางของพื้นที่
5. การระบุตำแหน่ง โดยใช้ตัวรับที่มากขึ้น จะทำให้ความคลาดเคลื่อนในการระบุตำแหน่งลดลง

การทดลองการระบุตำแหน่งแบบเคลื่อนที่

1. แนวโน้มของการระบุตำแหน่ง เป็นไปตามทิศทางการเคลื่อนที่ของวัตถุ
2. ค่าความผิดพลาดในการระบุตำแหน่งจะสูงกว่าการระบุตำแหน่งแบบหยุดนิ่ง
3. การระบุตำแหน่งที่มีความผิดพลาดน้อยที่สุด คือ 3-Nearest

5.2 ปัญหาและอุปสรรคในการทำงาน

การทดลอง

- อุปกรณ์ในการทดลองมีปัญหาในบางครั้งและได้ทำการแก้ไขแล้ว
- ด้วยสภาพอากาศที่เปลี่ยนแปลงทำให้ต้องทำการคำนวณหาค่าคงที่ในสมการ RSSI ใหม่
- บริเวณที่ทำการทดลอง มีสัญญาณ WIRELESS LAN อยู่ซึ่งเป็นช่วงความถี่เดียวกับ XBee อาจทำให้เกิดการรบกวนของสัญญาณได้
- บางครั้งเกิดความล่าช้าของข้อมูลเนื่องมาจากแหล่งพลังงานของ XBee เริ่มอ่อน

- ความเร็วในการเคลื่อนที่ของวัตถุเป็นข้อจำกัดในการทดลองระบุตำแหน่งแบบเคลื่อนที่

5.3 สิ่งที่สามารถนำไปพัฒนาต่อในอนาคต

การทดลอง

- ทดลองการระบุตำแหน่งในพื้นที่จริง
- พัฒนาอัลกอริทึมในการระบุตำแหน่ง
- นำไปประยุกต์ใช้กับโครงการอื่นๆ เช่น หุ่นยนต์กู้ภัย ระบบการติดตามวัตถุ เป็นต้น



บรรณานุกรม

- [1] สุชา สุพิทยภรณ์พงศ์, เครือข่ายเซ็นเซอร์ไร้สาย,
Available : <http://www.thaitelecomkm.org/TTE/topic/attach/WSN/index.php>
- [2] พงษ์ศักดิ์ กীরตวิมล, เทคโนโลยีการสื่อสารไร้สายระยะใกล้,
Available : <http://www.thaitelecomkm.org/TTE/topic/attach/Bluetooth/index.php>
- [3] ThaiEasyElec.com, ZigBee and XBee BASIC ตอน ZigBee คืออะไร,
Available : <http://www.thaitelecomkm.org/TTE/topic/attach/ZigBee/index.php>
- [4] ThaiEasyElec.com, ZigBee and XBee BASIC ตอน XBee คืออะไร,
Available : <http://www.thaieasyelec.com/Embedded-Electronics-Application/what-is-XBee.html>
- [5] The ZigBee Alliance, ZigBee Specification, 29 July 2007,
Available : <http://www.ZigBee.org>
- [6] Intelligent Automation,
Available : <http://www.i-a-i.com/view.asp?aid=354>
- [7] Location Tracking Approaches,
Available : <http://www.cisco.com/en/US/docs/solutions/Enterprise/Mobility/wifich2.pdf>
- [8] Pablo Brenner, A Technical Tutorial on the IEEE 802.11 Protocol. 2008 March 18,
Available : www.itconnection.ru.
- [9] B. Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function.
IEEE Journal on Selected Areas in Communications, March 2000
- [10] Oslo. ZigBee Architecture Overview, 2008 March 18,
Available : www.ZigBee.org.
- [11] IEEE 802.15.4 specification, Mai 2003 ,
Available : <http://standards.ieee.org/getieee802/download/802.15.4-2003.pdf>

- [12] IEEE802.15.4. ,2007 Nov 29,
Available : <http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf>
- [13] IEEE standards. 2007 Mar 21,
Available : <http://www.ieee802.org/11/>
- [14] ZigBee specification v1.0,
Available : http://www.chipcon.com/ZigBee/members_only.cfm, December 2004.
- [15] IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, 2008 March 18,
Available : <http://grouper.ieee.org/groups/802/11/index.html>
- [16] J.Desai, U.Tureli, Evaluating Performance of Various Localization Algorithms in Wireless and Sensor Networks, in: The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07), Athens, Greece, 2007, pp. 1-5,
- [17] Xiaowei Luo, William J. O'Brien, Christine L. Julien, Comparative evaluation of Received Signal-Strength Index (RSSI) based indoor localization techniques for construction jobsites, 2010 October 29,
Available : <http://mpc.ece.utexas.edu/Papers/ComparitiveEvaluationOfRSSI.pdf>
- [18] Eduardo Navarro, Benjamin Peuker, Michael Quan, Dr. Christopher Clark, Dr. Jenifer Jipson, Wi-Fi Localization Using RSSI Fingerprinting, Computer Engineering, Child Development, California Polytechnic State University
Available : <http://digitalcommons.calpoly.edu/cgi/viewcontent.cgi?article=1007&context=cpesp>
- [19] The XBee/XBee-PRO/S2B ZB Manual, Digi International Cooperation,2010