



ระบบระบุตำแหน่งภายในอาคารโดยใช้ Wireless

Wireless Indoor Positioning Systems

นายจาภูวนันท์ วิวัฒนากกตี รหัส 51371208
นายชินรพัฒน์ เกษียรนาถเรศ รหัส 51371253

ปริญญาในพนธน์เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต
สาขาวิชาชีวกรรมคอมพิวเตอร์ ภาควิชาชีวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธรรมศาสตร์
ปีการศึกษา 2555

| | |
|------------------------------|------------------|
| ห้อง: | กรรมศาสตร์ |
| วันที่รับ: | 25 ต.ค. 2556 |
| เลขทะเบียน: | 16240384 |
| แบบเรียบร้อย: | ผศ. |
| มหาวิทยาลัยธรรมศาสตร์ จ. 338 | |

2/๖๔



ใบรับรองโครงการวิศวกรรม

| | | |
|------------------|---|---------------|
| หัวข้อโครงการ | ระบบระบุตำแหน่งภายในอาคาร โดยใช้ Wireless | |
| ผู้ดำเนินโครงการ | นายชาญวัฒน์ วิวัฒนภักดี | รหัส 51371208 |
| | นายชินรพัฒน์ เกษียรนาถุเรศ | รหัส 51371253 |
| อาจารย์ที่ปรึกษา | อาจารย์ศรษฐา ตั้งค้าวานิช | |
| สาขาวิชา | วิศวกรรมคอมพิวเตอร์ | |
| ภาควิชา | วิศวกรรมไฟฟ้าและคอมพิวเตอร์ | |
| ปีการศึกษา | 2555 | |

คณะกรรมการศึกษาดูงาน
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ อนุมัติให้โครงการนี้เป็นส่วนหนึ่งของ
การศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิชาชีวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงการวิศวกรรม

ไดร์ฟ ลงตัว ประธานกรรมการ
(อาจารย์ศรษฐา ตั้งค้าวานิช)

P. Ut กรรมการ
(ดร. พงศ์พันธ์ กิจสนาโภธิน)

กุล อุ๊ กรรมการ
(ดร. วรลักษณ์ คงเด่นฟ้า)

นัน พ. กรรมการ
(อาจารย์ภาณุพงศ์ สอนกม.)

| | |
|------------------|---|
| หัวข้อโครงการ | ระบบระบุตำแหน่งภายในอาคารแบบโดยใช้ Wireless |
| ผู้ดำเนินโครงการ | นายจากรุ๊ดเน่ วิรัฒนภักดี รหัส 51371208 |
| | นายชิษรพัฒน์ เกษียรนาถurek รหัส 51371253 |
| อาจารย์ที่ปรึกษา | อาจารย์ศรอนรุํา ตั้งก้าวานิช |
| สาขาวิชา | วิศวกรรมคอมพิวเตอร์ |
| ภาควิชา | วิศวกรรมไฟฟ้าและคอมพิวเตอร์ |
| ปีการศึกษา | 2555 |

บทคัดย่อ

ในปัจจุบันการแสดงตำแหน่งของวัตถุนั้นเป็นสิ่งที่น่าสนใจและมีการพัฒนาอย่างแพร่หลายมากขึ้น โดยที่ระบบนำทางส่วนใหญ่จะใช้ความเที่ยงในการคำนวณหาตำแหน่งที่อยู่ของวัตถุหรือที่เรียกว่า GPS (Global Positioning System) ที่เราคุ้นเคยดี แต่มีข้อจำกัดในบางประการคือ จะต้องไม่มีสิ่งกีดขวางมาบังเวลารับส่งสัญญาณ โดยจะต้องอยู่ในที่โล่งแจ้งเท่านั้น ถ้าอยู่ภายในอาคารก็จะไม่สามารถใช้งานได้ ด้วยเหตุนี้จึงคิดระบบระบุตำแหน่งภายในอาคารขึ้นมา เพื่อที่จะมาทดแทนข้อจำกัดของระบบแบบ GPS

ในการพัฒนาโครงการนี้ได้นำเทคโนโลยีระบบเครือข่ายไร้สายมาใช้ในการติดตามตำแหน่งของวัตถุ โดยใช้การแปลงความแรงสัญญาณ RSSI (Received Signal Strength Indication) ออกมารูปเป็นระยะทาง ซึ่งต้องมีอุปกรณ์ ZigBee ติดตั้งไว้ตามจุดอ้างอิงอีก 3 จุดเพื่อกำหนดจุดอ้างอิง แล้วนำระยะทางดังกล่าวมาคำนวณหาตำแหน่งของวัตถุและแสดงผลที่คอมพิวเตอร์

| | | | |
|------------------------|--------------------------------------|----------------------------------|------------------------------|
| Project Title | Wireless Indoor Positioning Systems. | | |
| Name | Mr.Jaruwat Mr.Chinrapat | Viwattanapugdee Kasianmatures | ID. 51371208 ID. 51371253 |
| Project Advisor | Mr.Settha Thangkawanit | | |
| Major | Computer Engineering. | | |
| Department | Electrical and Computer Engineering. | | |
| Academic Year | 2012 | | |

ABSTRACT

The current position of the object of interest is the development and more widespread. The satellite navigation system is mainly used to calculate the location of the object, known as GPS (Global Positioning System) that we talk to as well. However, there are some limitations. Will be no obstacles to block the signal. It must be in the open air only. If inside a building, it will not work. To this end, we are building up a positioning system. In order to compensate for the limitations of GPS.

This project, object location tracking system uses RSSI (Received Signal Strength Indication) measurement. The received signal strength can be transformed into distance value using statistic process. Then calculate into location values using triangulate method and finally graphically display the result location on the computer.

กิตติกรรมประกาศ

โครงการวิศวกรรมคอมพิวเตอร์สำเร็จได้ด้วยดี เนื่องจากความอนุเคราะห์จากท่านอาจารย์ที่ปรึกษาโครงการ อาจารย์เพรนฐา ตั้งค้านนิช และอาจารย์สิริกพ ครรัตน์ที่กรุณาสละเวลาให้คำแนะนำในการทำงาน ตลอดจนถึงการตรวจสอบการทำงานพร้อมทั้งเสนอแนวทางการแก้ไขปัญหาตลอดระยะเวลาการทำงาน ตลอดจนถึงการดำเนินการที่มีประสิทธิภาพ ทุกคนที่เข้าร่วมโครงการได้รับการสนับสนุนอย่างดี ทำให้สามารถสำเร็จลุล่วงไปได้ด้วยดี

ขออภัยในความไม่ดีที่อาจมีส่วนใดส่วนหนึ่งที่ทำให้โครงการไม่สำเร็จลุล่วงไป แต่ขออภัยในความไม่ดีที่อาจมีส่วนใดส่วนหนึ่งที่ทำให้โครงการไม่สำเร็จลุล่วงไป ขออภัยในความไม่ดีที่อาจมีส่วนใดส่วนหนึ่งที่ทำให้โครงการไม่สำเร็จลุล่วงไป

นายราวัตโน วิวัฒนภักดี
นายชิรพัฒโน เกษียรนาถเรศ

สารบัญ

| | หน้า |
|--|-------|
| บทคัดย่อ | ก |
| ABSTRACT | ข |
| กิตติกรรมประกาศ | ก |
| สารบัญ | ง |
| สารบัญตาราง | ช |
| สารบัญรูป | ฉ |
| บทที่ 1 บทนำ | 1 |
| 1.1 ที่มาและความสำคัญของโครงการ | 1 |
| 1.2 วัตถุประสงค์ของโครงการ | 2 |
| 1.3 ขอบข่ายของโครงการ | 2 |
| 1.4 ขั้นตอนการดำเนินการ | 2 |
| 1.5 แผนการดำเนินงาน | 3 |
| 1.6 ผลที่คาดว่าจะได้รับ | 3 |
| 1.7 งบประมาณที่ใช้ | 4 |
| บทที่ 2 หลักการและทฤษฎี | 5 |
| 2.1 หลักการทำงานของการค้นหาตำแหน่งวัตถุในระบบเครือข่ายไร้สาย | 5 |
| 2.2 อัลกอริทึมที่ใช้ในการหาตำแหน่งวัตถุ | 5 |
| 2.2.1 Received-Signal-Strength (RSS) | 5 |
| 2.2.2 กฎของครามเมอร์ | 6 |
| 2.3 สิ่งที่ทำให้เกิดข้อผิดพลาดของสัญญาณ | 9 |
| 2.3.1 Multipath Fading and shadowing | 9 |
| 2.3.2 Nonline-of-Sight (NLOS) | 10 |
| 2.3.3 Multiple-Access Interference | 10 |
| 2.3.4 Fluctuations in Signal Propagation Speeds | 10 |

สารบัญ(ต่อ)

| | |
|--|-----------|
| | หน้า |
| 2.4 Moving Average Filter | 10 |
| 2.4.1 การทำงานของตัวกรอง | 11 |
| บทที่ 3 เทคโนโลยี | 12 |
| 3.1 ZigBee | 12 |
| 3.1.1 คุณสมบัติของ ZigBee | 13 |
| 3.1.2 โครงสร้างของ ZigBee | 13 |
| 3.2 Xbee คืออะไร | 15 |
| 3.3 การศึกษาการใช้งาน Xbee ในเบื้องต้น | 18 |
| 3.4 กำลังส่ง สายอากาศ และสัญญาณรับกวน ของ Xbee | 18 |
| 3.4.1 Xbee Association | 19 |
| 3.4.2 Xbee Addressing | 19 |
| 3.4.3 Xbee Operation Mode | 20 |
| บทที่ 4 วิธีดำเนินงาน | 21 |
| 4.1 โครงสร้างของระบบ | 21 |
| 4.2 การทำงานของระบบ | 21 |
| 4.2.1 จงชาร Reference Node | 21 |
| 4.2.2 จงชาร Object Node & Moniter Node | 24 |
| 4.3 การหาความแรงของสัญญาณ | 26 |
| 4.3.1 การส่งข้อมูลในรูปแบบของ API | 26 |
| 4.3.2 รูปแบบของ API Frame | 26 |
| 4.3.3 ประเภทของ API | 27 |
| 4.3.4 ตัวอย่างการใช้ API frame packet | 31 |
| 4.4 การทำแผนผังวัดดู | 32 |
| 4.4.1 การแปลงระยะทางจากค่าความแรงของสัญญาณ | 32 |
| 4.4.2 การใช้ Moving Average Filter ช่วยในการกรองค่าความเข้มสัญญาณ | 36 |

สารบัญ(ต่อ)

| | หน้า |
|---|--------|
| 4.4.3 การคำนวณตำแหน่งจากระยะทางที่ทำการแปลงแล้ว | 41 |
| 4.5 โปรแกรมระบุตำแหน่ง | 43 |
| 4.5.1 ส่วนค่าๆของโปรแกรม | 43 |
| 4.6 แผนภาพการทำงานของโปรแกรม | 50 |
| 4.7 หน้าที่การทำงานของฟังก์ชัน ค่าๆในโปรแกรม | 51 |
| บทที่ 5 ผลการทดลอง | 54 |
| 5.1 วัดระยะทางภายในอาคารจาก Node3 ไป Node1 เพื่อหาจำนวน Blok ที่เหมาะสม | 54 |
| 5.2 วัดระยะทางภายในอาคาร โดยใช้ Moving Average Filter | 55 |
| 5.3 วัดระยะทางภายในอาคาร โดยใช้จุดสังเกต 3 จุด | 59 |
| บทที่ 6 สรุปผล | 65 |
| 6.1 ผลการทดลอง | 65 |
| 6.1.1 วัดระยะทางภายในอาคาร โดยใช้ที่ระยะ 10 เมตร | 65 |
| 6.1.2 วัดระยะทางภายในอาคาร โดยใช้ Moving Average Filter 10 block | 65 |
| 6.1.3 วัดระยะทางภายในอาคาร โดยใช้จุดสังเกต 3 จุด | 65 |
| 6.2 ปัญหาและอุปสรรคในการพัฒนา | 65 |
| 6.3 แนวทางการพัฒนาต่อในอนาคต | 66 |
| 6.4 สรุป | 66 |
| เอกสารอ้างอิง | 67 |
| ภาคผนวก ก. เครื่องมือในการพัฒนา | 68 |
| ประวัติผู้เขียนโครงงาน | 69 |

สารบัญตาราง

| ตารางที่ | หน้า |
|---|------|
| ตารางที่ 1.1 แผนการดำเนินงาน | 3 |
| ตารางที่ 3.1 การเปรียบเทียบทุกโนโลยีไร้สายในแบบต่างๆ | 12 |
| ตารางที่ 4.1 การส่งคำสั่ง ATND | 32 |
| ตารางที่ 4.2 การตอบกลับมาหลังจากส่งคำสั่ง ATND | 32 |
| ตารางที่ 4.3 ACK ของคำสั่ง ATND | 32 |
| ตารางที่ 4.4 การเก็บค่า RSSI มาใช้สร้างสมการระเบททาง | 33 |
| ตารางที่ 4.5 การเลือกค่า RSSI มาใช้สร้างสมการระเบททาง | 34 |
| ตารางที่ 4.6 แปลงสมการ Trend Line กลับออกมาเป็นระเบททาง | 35 |
| ตารางที่ 4.7 เปรียบเทียบค่า RSSI เดิมกับค่า RSSI ใหม่ที่ใช้ Moving Average Filter | 40 |
| ตารางที่ 5.1 แสดงเวลาที่ใช้กันเปอร์เซ็นต์ความคลาดเคลื่อน | 54 |

สารบัญรูป

| รูปที่ | หน้า |
|--|------|
| รูปที่ 2.1 Function block diagram ของระบบคันหาด้าแห่งในเครือข่ายไร้สาย | 5 |
| รูปที่ 2.2 วิธีการหาตำแหน่งของวัตถุ | 6 |
| รูปที่ 2.3 แสดงวงกลม 3 วงตัดกันทำให้ได้ตำแหน่งวัตถุ | 8 |
| รูปที่ 2.4 แสดงผลจากการใช้อัลกอริทึม Moving Average Filter | 11 |
| รูปที่ 3.1 ปริมาณการใช้ Throughput ของมาตรฐานการสื่อสาร ไร้สายแบบต่างๆ | 13 |
| รูปที่ 3.2 สายอากาศแบบ Wire (Whip) , Chip ตามลำดับ | 16 |
| รูปที่ 3.3 สายอากาศแบบ RPSMA Connector , UFL Connector ตามลำดับ | 16 |
| รูปที่ 3.4 Xbee Series1 Pro Whip ant | 16 |
| รูปที่ 3.5 Datasheet ของ Xbee | 17 |
| รูปที่ 3.6 เครือข่าย Zigbeeแบบ Star , Cluster , Mesh | 17 |
| รูปที่ 3.7 ระบบการรับส่ง โดยใช้สายต่อชานิดกัน | 19 |
| รูปที่ 4.1 วงจร Reference Node แบบใช้แบตเตอรี่ | 22 |
| รูปที่ 4.2 วงจร Reference Node แบบใช้แบตเตอรี่ (ที่ใช้หดคลอง) | 22 |
| รูปที่ 4.3 วงจร Reference Node แบบใช้ไฟจาก Adepter 5V | 23 |
| รูปที่ 4.4 วงจร Reference Node แบบใช้ไฟจาก Adepter 5V (ที่ใช้หดคลอง) | 23 |
| รูปที่ 4.5 วงจร Object Node | 24 |
| รูปที่ 4.6 วงจร Object Node (ที่ใช้หดคลอง) | 25 |
| รูปที่ 4.7 ภาพรวมการทำงานของระบบ | 25 |
| รูปที่ 4.8 รูปแบบของ API Packet frame Mode 1 | 26 |
| รูปที่ 4.9 รูปแบบของ API Packet frame | 27 |
| รูปที่ 4.10 รูปแบบของ API Packet frame ประเภท AT Command | 27 |
| รูปที่ 4.11 รูปแบบของ API Packet frame ประเภท AT Command Queue Parameter Value | 28 |
| รูปที่ 4.12 รูปแบบของ API Packet frame ประเภท AT Command Response | 28 |
| รูปที่ 4.13 รูปแบบของ API Packet frame ประเภท Tx (Transmit) Request : 64-bit address | 29 |
| รูปที่ 4.14 รูปแบบของ API Packet frame ประเภท Tx (Transmit) Request : 16-bit address | 29 |
| รูปที่ 4.15 รูปแบบของ API Packet frame ประเภท Tx (Transmit) | 30 |
| รูปที่ 4.16 รูปแบบของ API Packet frame ประเภท RX (Receive) Packet 64-bit address | 30 |

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|--|------|
| รูปที่ 4.17 รูปแบบของ API Packet frame ประเภท RX (Receive) Packet 16-bit address | 31 |
| รูปที่ 4.18 กราฟแสดงค่าเฉลี่ยของระยะทางกับค่า RSSI | 34 |
| รูปที่ 4.19 กราฟแสดงค่าเฉลี่ยของระยะทางกับค่า RSSI โดยตัดข้อมูลที่กระโดดออก | 35 |
| รูปที่ 4.20 กราฟแสดงค่า RSSI เทียบกับระยะทาง | 36 |
| รูปที่ 4.21 รูปกราฟแสดงค่า RSSI ที่ไม่ได้ใช้อัลกอริทึม | 40 |
| รูปที่ 4.22 รูปกราฟที่ใช้ Moving Average Filter กรองค่า RSSI | 41 |
| รูปที่ 4.23 โปรแกรมระบุตำแหน่งโดยรวม | 43 |
| รูปที่ 4.24 แสดงส่วนประกอบต่างๆของโปรแกรม | 44 |
| รูปที่ 4.25 แสดงส่วนประกอบของ connection | 45 |
| รูปที่ 4.26 แสดงส่วนประกอบของ Set position Xbee node | 45 |
| รูปที่ 4.27 แสดงส่วนประกอบของ AT and API function | 46 |
| รูปที่ 4.28 แสดงส่วนประกอบของ Status datalog | 46 |
| รูปที่ 4.29 แสดงส่วนประกอบของ Map tracking | 47 |
| รูปที่ 4.30 แสดงส่วนประกอบของ Moniter of sensor node | 47 |
| รูปที่ 4.31 แสดงส่วนประกอบของ Control position | 48 |
| รูปที่ 4.32 แสดงส่วนประกอบของ Datalog | 48 |
| รูปที่ 4.33 แสดงส่วนประกอบของ Function datalog | 49 |
| รูปที่ 4.34 แสดงส่วนประกอบของ Position control | 49 |
| รูปที่ 4.35 แสดงส่วนประกอบของ Position datalog | 49 |
| รูปที่ 4.36 แผนภาพการทำงานของโปรแกรม | 50 |
| รูปที่ 5.1 ภาพแสดงการวัดค่าจาก Node 3 ไปยัง Node 1 เพื่อทำการหาค่า M ที่เหมาะสม | 54 |
| รูปที่ 5.2 ภาพแสดงการวัดค่าในพื้นที่จริงเพื่อทำการหาค่า M ที่เหมาะสม | 55 |
| รูปที่ 5.3 ที่ระยะ 0 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 1.000417 เมตร | 56 |
| รูปที่ 5.4 ที่ระยะ 5 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 4.894357 เมตร | 56 |
| รูปที่ 5.5 ที่ระยะ 10 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 10.89215 เมตร | 56 |
| รูปที่ 5.6 ที่ระยะ 15 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 14.08812 เมตร | 57 |

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|--|------|
| รูปที่ 5.7 ที่ระยะ 20 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 19.62941 เมตร | 57 |
| รูปที่ 5.8 ที่ระยะ 25 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 25.52645 เมตร | 57 |
| รูปที่ 5.9 ที่ระยะ 30 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 16.77712 เมตร | 58 |
| รูปที่ 5.10 ที่ระยะ 40 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 11.23613 เมตร | 58 |
| รูปที่ 5.11 กราฟแสดงจำนวนครั้งที่ทำการวัดกับระยะทางจริง | 58 |
| รูปที่ 5.12 ภาพการทดลองการวัดระยะทาง | 59 |
| รูปที่ 5.13 วัดระยะที่พิกัด (27,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (26.93,12.07) | 59 |
| รูปที่ 5.14 วัดระยะที่พิกัด (25,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (25.43,10.98) | 60 |
| รูปที่ 5.15 วัดระยะที่พิกัด (23,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (23.13,11.18) | 60 |
| รูปที่ 5.16 วัดระยะที่พิกัด (20,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (18.72,12.09) | 60 |
| รูปที่ 5.17 วัดระยะที่พิกัด (29,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (29.22,10.77) | 61 |
| รูปที่ 5.18 วัดระยะที่พิกัด (31,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (31.87,12.41) | 61 |
| รูปที่ 5.19 วัดระยะที่พิกัด (34,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (34.50,11.18) | 61 |
| รูปที่ 5.20 วัดระยะที่พิกัด (12,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (11.49,17.50) | 62 |
| รูปที่ 5.21 วัดระยะที่พิกัด (40,15) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (49.49,-49.30) | 62 |
| รูปที่ 5.22 กราฟแสดงค่าความคลาดเคลื่อนระหว่างค่าที่คำนวณได้กับค่าจริง | 63 |
| รูปที่ 5.23 การติดตั้ง XBee Reference Node ที่ 1 บนเพดาน | 63 |
| รูปที่ 5.24 การติดตั้ง XBee Reference Node ที่ 2 บนเพดาน | 64 |
| รูปที่ 5.25 การติดตั้ง XBee Reference Node ที่ 3 บนเพดาน | 64 |
| รูปที่ 5.26 การวัดค่า RSSI จาก XBee Tracking Node | 64 |

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

ในปัจจุบันระบบระบุตำแหน่งที่เราใช้กันอย่างแพร่หลายนั้นส่วนใหญ่จะใช้การวัดเวลาในการรับส่งข้อมูลระหว่างความเที่ยงกับวัดถูกที่เราสนใจ ซึ่งการใช้วิธีการนี้ในการระบุตำแหน่งนั้นจะมีข้อจำกัดที่ว่าวัดถูกที่เราสนใจจะต้องอยู่ในสภาพแวดล้อมที่ดีไม่มีสิ่งกีดขวางในการรับส่งข้อมูลกับความเที่ยมเห็น จะต้องอยู่ในที่โล่งแจ้ง ภายนอกอาคาร ถนน ทางเดินเท้า ฯลฯ ซึ่งข้อจำกัดเหล่านี้ทำให้ไม่สามารถระบุตำแหน่งวัดถูกที่อยู่ภายในอาคาร หรือในที่ที่สัญญาณดาวเทียมไม่สามารถเข้าถึงได้

ด้วยข้อจำกัดเหล่านี้ จึงเป็นเหตุผลให้เราเกิดความคิดที่จะพัฒนาระบบที่สามารถระบุตำแหน่งวัดถูกที่เราสนใจภายในอาคารได้ โดยการใช้เครือข่าย Wireless Personal Area Network (WPANs) ซึ่งจะแก้ปัญหาข้อจำกัดในการรับส่งสัญญาณผ่านสิ่งกีดขวางของสัญญาณดาวเทียมได้เหตุผลที่เราเลือกใช้เครือข่าย wireless ใน การค้นหาตำแหน่งวัดถูกที่เราสนใจภายในอาคารเพราะเทกโนโลยีนี้สามารถรับส่งข้อมูลด้วยความเร็วสูงและสามารถเดินทางผ่านสิ่งกีดขวางต่างๆ ได้ แต่จะมีข้อจำกัดในเรื่องของรัศมีของการส่งสัญญาณซึ่งจะต้องศึกษาการจัดวางเครือข่ายให้ค่อนข้างติดตั้งระบบ ในปัจจุบันมีการนำเทคโนโลยีการสื่อสารแบบไร้สายเข้ามาใช้ในการรับส่งข้อมูลกันอย่างแพร่หลายและปัจจุบันเทคโนโลยีนี้ได้มีการพัฒนาอยู่ปัจจุบันที่มีประสิทธิภาพในการรับส่งข้อมูลสูงขึ้นและมีราคาที่ถูกลงกว่าเดิมมาก ซึ่งการพัฒนาระบบระบุตำแหน่งภายในอาคารนี้จะทำให้สามารถเก็บข้อมูลของวัดถูกที่เราสนใจและนำข้อมูลเหล่านี้ไปช่วยในการวิเคราะห์ ตัดสินใจ หรือนำไปประยุกต์ใช้กับงานด้านต่างๆ เช่น การติดตามสิ่งของ บุคคล หรือการวิเคราะห์พฤติกรรมการเคลื่อนไหวของสิ่งที่เราสนใจได้ โดยโปรแกรมจะทำการโช่วยวัดตำแหน่งของวัดถูกที่เราสนใจ ซึ่งได้ทำการคำนวณระยะทางความห่างของวัดถูกที่เราสนใจกับจุดที่เราทำการสั่งทำการประมวลผลจากคอมพิวเตอร์แบบ real-time โดยจะทำการเก็บตำแหน่งพิกัดของวัดถูกที่เราสนใจไว้ ซึ่งสามารถนำข้อมูลที่ระบบบันทึกไว้มาวิเคราะห์ หรือใช้ในการตัดสินใจได้ เช่น การตรวจสอบสิ่งของที่ผู้ร้ายนำขึ้นมาจากห้องที่มีการติด Tag ไว้ที่สิ่งของ ซึ่งเราสามารถที่จะวิเคราะห์ได้ว่าส่วนใหญ่ผู้ร้ายจะใช้เส้นทางไหนเป็นเส้นทางหลบหนีจากการเก็บข้อมูลการเดินของผู้ร้ายและอาจจะสามารถเก็บหลักฐานของขึ้นได้ เช่น รอยเท้า หรือชิ้นส่วนบางอย่างที่ผู้ร้ายทิ้งไว้เป็นหลักฐานจากการเก็บข้อมูลตำแหน่งการเดินทาง เป็นต้น

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนาระบบระบุตำแหน่งของวัตถุที่เราสนใจภายในอาคาร โดยอาศัยการรับส่งข้อมูลระหว่างอุปกรณ์ภายในเครือข่าย Wireless Personal Area Network (WPANs)
2. เพื่อพัฒนาโปรแกรมที่สามารถนำไปประยุกต์ใช้กับระบบรักษาความปลอดภัยได้
3. เพื่อพัฒนาระบบการจัดวางตำแหน่งของอุปกรณ์การรับส่งสัญญาณในตำแหน่งที่เหมาะสม

1.3 ขอบข่ายของโครงการ

1. โปรแกรมสามารถคำนวณหาตำแหน่งของวัตถุที่สนใจได้โดยใช้การติด Wi-Fi Tag กับอุปกรณ์ที่สนใจ
2. โปรแกรมสามารถแสดงผลการเคลื่อนที่ของวัตถุที่สนใจแบบ Real-Time ได้
3. โปรแกรมสามารถบันทึกพิกัดการเคลื่อนที่ของวัตถุที่สนใจลงในฐานข้อมูลระบบและเรียกดูข้อมูลได้

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาเก็บข้อมูลและกำหนดขอบเขตของงาน
2. เขียนโปรแกรมทำเป็นซอฟต์แวร์
3. ทำการทดสอบซอฟต์แวร์ และทดลองเชื่อมต่อเครือข่าย เพื่อคำนวณหาตำแหน่งพิกัดของวัตถุที่สนใจ
4. ทดสอบประสิทธิภาพซอฟต์แวร์โดยผู้ใช้
5. ทำการสรุปรายรวมข้อมูลทั้งหมดของโครงการ
6. จัดทำ Report

1.5 แผนการดำเนินงาน

ตารางที่ 1.1 แผนการดำเนินงาน

| กิจกรรม | เดือน - ปี | | | | | | | | | |
|---|------------|----------|---------|---------|--------|-----------|---------|--------|------------|--------|
| | 2553 | | 2554 | | | | | | | |
| | พฤษภาคม | มิถุนายน | กรกฎาคม | กันยายน | ตุลาคม | พฤศจิกายน | ธันวาคม | มกราคม | กุมภาพันธ์ | มีนาคม |
| 1. ศึกษาค้นคว้าหาข้อมูล และกำหนดขอบเขตของงาน | | | ↔ | | | | | | | |
| 2. เขียนโปรแกรมทำเป็นซอฟต์แวร์ | | | | ↔ | ↔ | | | | | |
| 3. ทำการทดสอบซอฟต์แวร์ และทดลองเชื่อมต่อเครือข่าย เพื่อคำนวณหาตำแหน่งพิกัดของวัตถุที่สนใจ | | | | | | | ↔ | | | |
| 4. ทดสอบประสิทธิภาพ ซอฟต์แวร์โดยผู้ใช้ | | | | | | | | ↔ | | |
| 5. ทำการสรุป รวบรวม ข้อมูลทั้งหมดของโครงการ | | | | | | | | | ↔ | |
| 6. จัดทำ Report | | | | | | | | | | ↔ |

1.6 ผลที่คาดว่าจะได้รับ

1. ได้โปรแกรมที่สามารถนำไปใช้ระบบดำเนินการในอาคาร
2. สามารถนำโปรแกรมไปประยุกต์ใช้กับระบบปรึกษาความปลอดภัยได้

3. ได้รับความรู้ ความเข้าใจเกี่ยวกับการรับส่งข้อมูลกับอุปกรณ์รับส่งสัญญาณ โดยใช้เครือข่าย wireless

1.7 งบประมาณที่ใช้

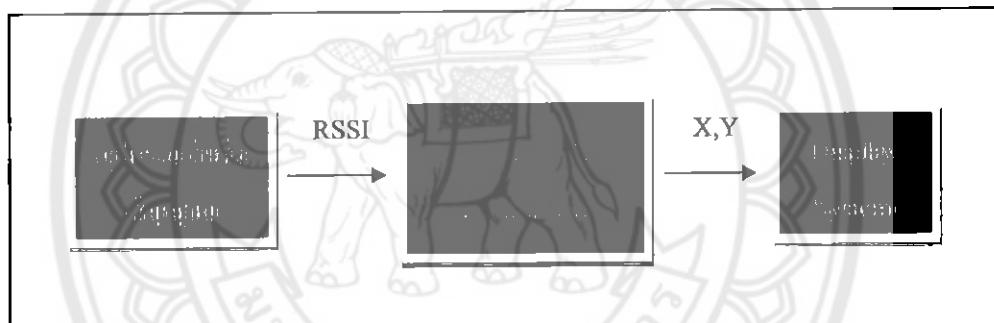
| | | |
|--|-------|------------|
| 1. ค่าวัสดุ อุปกรณ์ ในการจำลองการติดตั้งเครือข่าย wireless | 8,000 | บาท |
| 2. ค่าอุปกรณ์ที่ใช้ในการติดตามด้วยวัดดู | 2,000 | บาท |
| 3. ค่าจัดพิมพ์เอกสารและถ่ายเอกสาร | 400 | บาท |
| รวม | | 10,400 บาท |



บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 หลักการทำงานของการค้นหาตำแหน่งวัตถุในระบบเครือข่ายไร้สาย

การค้นหาตำแหน่งวัตถุในระบบเครือข่ายไร้สายมีส่วนประกอบหลักๆ 3 ส่วน คือ อุปกรณ์สำหรับระบุตำแหน่ง, อัลกอริทึมสำหรับค้นหาตำแหน่ง และอุปกรณ์แสดงผล โดยอุปกรณ์สำหรับระบุตำแหน่งวัตถุจะรับสัญญาณจากวัตถุที่ถูกติดตัวรับสัญญาณไว้ และนำสัญญาณนั้นมาคำนวณโดยใช้อัลกอริทึมค่างๆในการค้นหาตำแหน่ง เช่น การใช้สามเหลี่ยมระยะ การใช้ Moving Average Filter [9] เป็นต้น หลังจากได้ตำแหน่งของวัตถุแล้วจึงนำไปแสดงผลดังรูปที่ 2.1



รูปที่ 2.1 Function block diagram ของระบบค้นหาตำแหน่งในเครือข่ายไร้สาย

2.2 อัลกอริทึมที่ใช้ในการหาตำแหน่งวัตถุ

2.2.1 Received-Signal-Strength (RSS)

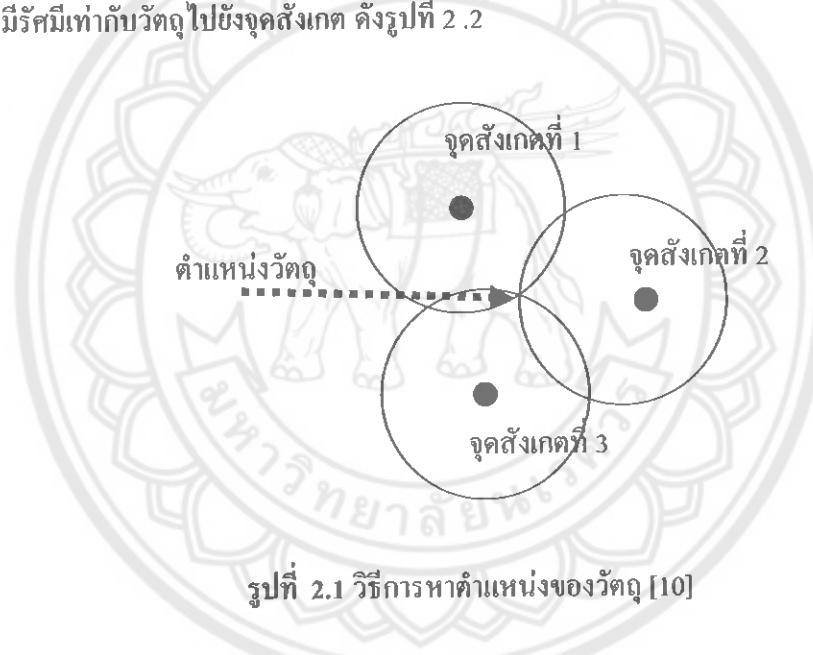
Received-Signal-Strength [1] เป็นวิธีการวัดความแรงของสัญญาณอาศัยหลักการลดthonของสัญญาณ (Signal attenuation) ต่อระยะทางเพื่อใช้ในการคำนวณหาระยะห่างของวัตถุ โดยการวัดความแรงของสัญญาณจะใช้วงจรที่แปลงค่าความแรงของสัญญาณมาเป็นความแรงของแรงดันไฟฟ้า (RSSI)

ข้อดีของการใช้การวัดแบบนี้คือ ไม่ต้องการฮาร์ดแวร์ที่มีราคาสูงและง่ายต่อการติดตั้ง เหมาะสำหรับการวัดในเมืองหรือที่ร่ม ใน Cellular และ WLAN network มีอุปกรณ์สำหรับวัดค่า RSS อยู่แล้ว โดยไม่ต้องเพิ่มอุปกรณ์ใดๆ สำหรับการวัดแบบ direct measurement ไม่มีความน่าเชื่อถือ เพราะว่ามี Standard deviation ของ Error มากซึ่ง ได้รับผลกระทบจาก Shadow fading ก่อนข้างมาก การทำให้ RSS

น่าเชื่อถือมากขึ้นจะต้องสร้างระบบในการจดจำพื้นที่นั้นๆ โดยการวัดค่าและเก็บข้อมูลไว้หลายครั้ง เมื่อวิเคราะห์ภายนอกก็จะมีความผิดพลาดน้อยลง

วิธีนี้ไม่นิยมใช้ในงานที่ต้องการความแม่นยำสูงเนื่องจากค่าความแปรปรวนของสัญญาณจากสภาพแวดล้อมต่างๆ กัน โดยเฉพาะอย่างยิ่งเมื่อสัญญาณมาจากหลายทิศทางและเกิดเงาสะท้อนของสัญญาณ (Shadowing effect) อย่างไรก็ตามวิธีการวัดความแรงของสัญญาณ (Signal-strength-based) มักนำไปใช้ในการประมาณค่าตำแหน่งวัตถุหรือนำໄไปใช้ร่วมกับวิธีอื่นในการพิจารณา

นอกจากนี้เราสามารถเพิ่มความแม่นยำของการทำแผนที่วัตถุได้มากขึ้น โดยใช้หลักการทำงานคณิตศาสตร์โดยในครั้งที่ใช้วัดระยะทางต้องมีอย่างน้อย 3 ในครั้งในระบบ 2 มิติ (หากเป็นระบบ 3 มิติ ต้องการ 4 ระยะทางจาก 4 จุดสังเกต) ตำแหน่งของวัตถุจะอยู่ที่จุดตัด (Intersection) ของวงกลม 3 วงที่แต่ละวงมีรัศมีเท่ากับวัตถุไปยังจุดสังเกต ดังรูปที่ 2.2



รูปที่ 2.1 วิธีการหาตำแหน่งของวัตถุ [10]

2.2.2 กฎของครามอร์

กฎของครามอร์ (Cramer's rule) [10] เป็นระบบที่ใช้แก้ระบบสมการขนาดเล็กได้โดยง่าย หัวใจของกฎครามอร์คือการหาค่าตัวกำหนด (determinant) โดยกฎของครามอร์ก่อให้ว่า ตัวไม่รู้ค่า x_i ของระบบสมการหาได้จาก

$$x_i = \frac{\det[A_i]}{\det[A]} \quad \text{สมการ (2.1)}$$

โดย $\det[A]$ แทนค่าตัวกำหนดของเมตริกซ์ $[A]$ และ $\det[A]$ แทนค่าตัวกำหนดของเมตริกซ์ $[A]$ หลังจากที่เมทริกซ์ $[A]$ นี้ได้เปลี่ยนค่าในแนวเดาตั้ง; ด้วยค่าในเวกเตอร์ $\{B\}$ แล้ว การหาผลลัพธ์ค่าวิธี Kraemer

ตัวอย่าง การใช้กฎของรามอร์ในการแก้ระบบสมการ

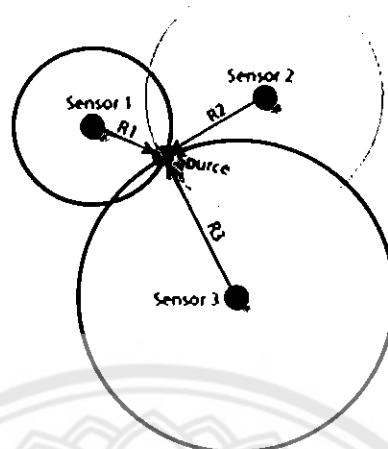
$$\begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 4 \\ -1 \end{bmatrix} \quad \text{สมการ (2.2)}$$

$$x_1 = \frac{\det[A]_1}{\det[A]} = \frac{\begin{vmatrix} 4 & 1 \\ -1 & -1 \end{vmatrix}}{\begin{vmatrix} 2 & 1 \\ 1 & -1 \end{vmatrix}} = \frac{-4+1}{-2-1} = 1 \quad \text{สมการ (2.3)}$$

$$x_2 = \frac{\det[A]_2}{\det[A]} = \frac{\begin{vmatrix} 2 & 4 \\ 1 & -1 \end{vmatrix}}{\begin{vmatrix} 2 & 1 \\ 1 & -1 \end{vmatrix}} = \frac{-2-4}{-2-1} = 2 \quad \text{สมการ (2.4)}$$

เราใช้สมการวงกลม 3 สมการ เมื่อแทนค่าตัวแปรต่างๆจะเหลือตัวแปร 2 ตัวคือ X และ Y ซึ่งคือพิกัดแกน X และแกน Y ของทำหมา่งวัตุที่ต้องการทันท่วงที

$$\text{สมการวงกลม} \quad (X-h)^2 + (Y-k)^2 = R^2 \quad \text{สมการ (2.5)}$$



รูปที่ 2.3 แสดงวงกลม 3 วงตัดกันทำให้ได้ตัวແນ່ງວັດຖຸ [10]

การหาผลเฉลยของระบบสมการเชิงเส้นจากกฎของครามอร์ (Cramer's Rule)

การหาผลเฉลยของระบบสมการเชิงเส้นจากกฎของครามอร์ เป็นวิธีทางผลเฉลยวิธีหนึ่งของระบบสมการเชิงเส้นที่มี n สมการ และ n ตัวแปร โดยใช้ตัวกำหนดเข้ามาช่วยในการหาผลเฉลย **กฎถูกต้อง** ถ้า $AX = B$ เป็นระบบสมการเชิงเส้น n สมการ n ตัวแปร และ $|A| \neq 0$ แล้ว ระบบสมการเชิงเส้นนี้มีผลเฉลยเพียงผลเฉลยเดียว คือ

$$x_i = \frac{|A_i|}{|A|} ; i = 1, 2, \dots, n \quad \text{สมการ (2.6)}$$

โดยที่ A_i เป็นเมทริกซ์ที่ได้จากเมทริกซ์ A โดยการแทนที่แนวตั้งที่ i ของ A ด้วยเมทริกซ์ B

ตัวอย่าง การแก้สมการหาตัวແນ່ງວັດຖຸ โดยใช้หลักເກີມທົ່ວໂລມ

$$\text{เมื่อ } (X_1, Y_1) = (2, 1), (X_2, Y_2) = (5, 4), (X_3, Y_3) = (8, 2) \quad R_1 = \sqrt{10} \quad R_2 = 2 \quad R_3 = 3$$

จากสมการวงกลมหลัก 3 สมการดังนี้

$$\begin{aligned}
 (X_1 - X_u)^2 + (Y_1 - Y_u)^2 &= R_1^2 \\
 (X_2 - X_u)^2 + (Y_2 - Y_u)^2 &= R_2^2 \\
 (X_3 - X_u)^2 + (Y_3 - Y_u)^2 &= R_3^2
 \end{aligned} \quad \text{สมการ (2.7)}$$

จัดให้ออยู่ในรูปสมการเมทริกซ์

$$2 \begin{bmatrix} X_3 - X_1 & Y_3 - Y_1 \\ X_3 - X_2 & Y_3 - Y_2 \end{bmatrix} \begin{bmatrix} X_u \\ Y_u \end{bmatrix} = \begin{bmatrix} (R_1^2 - R_3^2) - (X_1^2 - X_3^2) - (Y_1^2 - Y_3^2) \\ (R_2^2 - R_3^2) - (X_2^2 - X_3^2) - (Y_2^2 - Y_3^2) \end{bmatrix} \quad \text{สมการ (2.8)}$$

วิธีที่ 1

$$A = 2 \begin{bmatrix} X_3 - X_1 & Y_3 - Y_1 \\ X_3 - X_2 & Y_3 - Y_2 \end{bmatrix}, B = \begin{bmatrix} X_u \\ Y_u \end{bmatrix} \quad \text{สมการ (2.9)}$$

แทนค่าจากที่กำหนด

$$2 \begin{bmatrix} 6 & 1 \\ 3 & -2 \end{bmatrix} \begin{bmatrix} X_u \\ Y_u \end{bmatrix} = \begin{bmatrix} 64 \\ 22 \end{bmatrix} \quad \text{สมการ (2.10)}$$

จะได้ $|A| = -15, |A_1| = -150, |A_2| = -60$

$$\text{ทำให้ได้ } X_1 = \frac{|A_1|}{A} = \frac{-75}{-15} = 5, X_2 = \frac{|A_2|}{A} = \frac{-30}{-15} = 2$$

ดังนั้นวัตถุที่เราต้องการหาอยู่ในพิกัด $(X, Y) = (5, 2)$

2.3 สิ่งที่ทำให้เกิดข้อผิดพลาดของสัญญาณ

การประมาณค่าตำแหน่งจากการวัดที่มีสัญญาณรบกวนสามารถปรับปรุงให้ดีขึ้นได้ หากเราเข้าใจถึงพฤติกรรมของข้อผิดพลาดที่เกิดขึ้น ชนิดของข้อผิดพลาดที่เกิดขึ้นกับชนิดของสัญญาณและสิ่งแวดล้อมของระบบ

2.3.1 Multipath Fading and shadowing

ในการวัดความแรงของสัญญาณคลื่นวิทยุปัจจุบัน Multipath fading และ shadowing เป็นสาเหตุให้เกิดความแปรปรวนได้ถึง 30-40 dB ในช่วงครึ่งความยาวคลื่น การกระจายไก่เครื่องรับจะทำ

ให้เกิดการเบร์ผันทางมุม (Angle of arrival) ทำให้ค่าการวัดคลาดเคลื่อนไป ถ้าใช้วิธี Time-of-Arrival เมื่อเกิด Multipath fading จะทำให้เกิดค่าหน่วงเวลา (Delay)

2.3.2 Nonline-of-Sight (NLOS)

สำหรับวิธีวัดมุม (AoA) เมื่อใช้กับระบบทางไกลๆ จะทำให้เกิดความคลาดเคลื่อนสูงสำหรับวิธีวัด ความต่างของเวลา (ToF) ถ้าสั่นทางไปบังเครื่องรับที่ใกล้ที่สุดถูกกีดขวางจะทำให้ผลของการวัดระบบทางไกลกว่าที่เป็นจริง

2.3.3 Multiple-Access Interference

ปัญหานี้มักเกิดกับระบบ CDMA โดยเครื่องที่มีกำลังสูงไปบุกงานเครื่องที่มีกำลังต่ำกว่าสามารถเกิดกับระบบคลื่นเสียงและระบบคลื่นอัลตราโซนิก (Ultrasonic) ในระบบที่ใช้การวัดความแรงของสัญญาณที่ได้ (received signal strength indicator (RSSI)) จากคลื่นวิทยุราคายุก เมื่อระบบไม่ใช้อุปกรณ์ที่มีความแม่นยำสูง ค่าความแปรปรวนจะเกิดจากความคลาดเคลื่อนเมื่อใช้เครื่องเปล่งกำลัง

2.3.4 Fluctuations in Signal Propagation Speeds

เกิดกับคลื่นเสียงที่การแพร์อูกรบกวนจากปัจจัยภายนอก เช่น แรงลมหรืออุณหภูมิและค่าความชื้นในอากาศ ปัจจัยเหล่านี้จะทำให้ค่าความคลาดเคลื่อนสูงขึ้นเมื่อระยะเวลามากขึ้น

2.4 Moving Average Filter

การกรอง (filter) ข้อมูลเป็นการประยุกต์ใช้งานคอมพิวเตอร์ทางฟิสิกส์แบบเบื้องต้น ซึ่งมีจุดประสงค์ในการจัดการกับข้อมูลที่วัดได้ หรือเก็บได้จากเครื่องมือต่าง ๆ ให้มีความสวยงามและน่าเชื่อถือมากขึ้น ในสมัยก่อนจะใช้ผู้คนลงเองในการตัดสิน โดยการลากเส้นตัวแทนหรือหาค่าเฉลี่ยที่คงค่า อาจตัดทิ้งหรือปรับค่าบางจุดให้เหมาะสม (มักเรียกว่า makeข้อมูล) ซึ่งกระบวนการแบบนี้มักเชื่อถือไม่ได้ และขึ้นอยู่กับคนที่ใช้งานเป็นสำคัญ หากผู้ที่จัดการข้อมูลมีความลำเอียงอยู่อาจทำให้ข้อมูลสำคัญสูญหายไปก็ได้ การหากะบานการมาตรฐานเพื่อนำมาปรับแต่งข้อมูลดังกล่าว จะทำให้การจัดการข้อมูลมีมาตรฐานเดียวกัน และสามารถนำไปอ้างอิงได้โดยปราศจากข้อกังขา

Moving average filter เป็นตัวกรองเชิงดิจิตอลที่ง่ายที่สุด มีหลักการที่เรียบง่าย และนำไปใช้งานได้จริง โดยเริ่วโดยจะเป็นการหาผลลัพธ์ที่ผ่านการกรองแล้วโดยใช้ค่าข้อมูลเดิม ผ่านการคำนวณที่ง่าย ๆ ไม่ซับซ้อน โดยส่วนใหญ่จะเป็นการคูณและบวกเท่านั้น ดังนั้นเทคนิคนี้มักจะถูกเรียกว่า Multiply-and-Accumulate (MAC) อีกทั้งໄร์ก็ตี ตัวกรองแบบนี้สามารถปรับแต่งให้ทำหน้าที่ต่าง ๆ ได้มากมาย แล้วแต่ความต้องการผู้ใช้

2.4.1 การทำงานของตัวกรอง

ตัวกรองแบบนี้จะทำงานโดยใช้ข้อมูลที่ได้รับ (Input Data - X_i) จำนวนไม่น้อยกว่า 1 ค่าเพื่อคำนวณค่าผลลัพธ์(Output - Y_i) โดยมีความสัมพันธ์ที่พราะจะเขียนได้ดังนี้

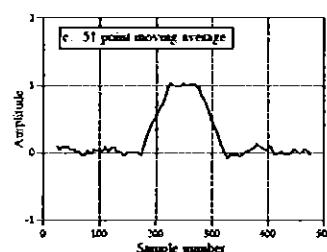
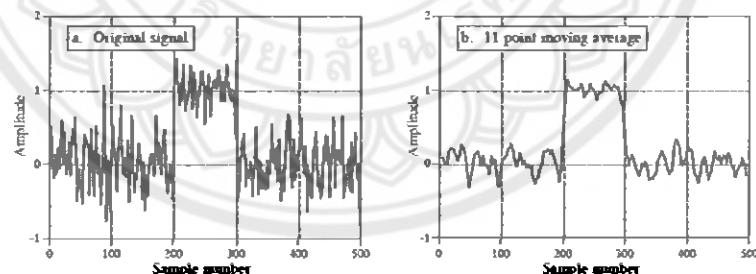
$$Y[i] = \frac{1}{M} \sum_{j=0}^{M-1} X[i+j] \quad \text{สมการ (2.11)}$$

เมื่อ M เป็นจำนวนข้อมูลเฉลี่ยทั้งหมดตัวกรองแบบนี้จะเรียกว่า boxcar filter ซึ่งเป็นตัวกรองแบบง่ายที่สุด สามารถนำไปใช้งานได้เร็ว

ตัวอย่างสมมุตินิข้อมูล 6 ค่า เราสามารถหา Moving Average Filter ได้ดังนี้

$$Y[0] = \frac{X[0]+X[1]+X[2]+X[3]+X[4]}{5}$$

$$Y[1] = \frac{X[1]+X[2]+X[3]+X[4]+X[5]}{5}$$



รูปที่ 2.4 แสดงผลจากการใช้อัลกอริทึม Moving Average Filter

บทที่ 3

เทคโนโลยี

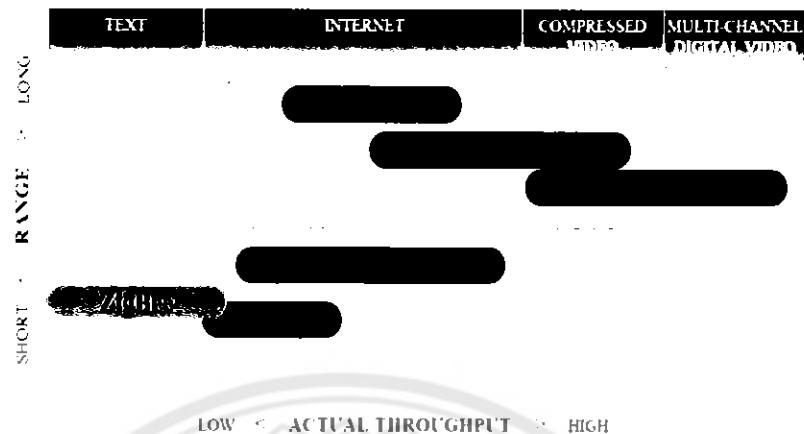
ในบทที่ 3 นี้จะกล่าวถึงเทคโนโลยีที่เกี่ยวข้องและนำมาใช้ในโครงงาน อันได้แก่ ZigBee , คุณสมบัติของ ZigBee , โครงสร้างของ ZigBee , Xbee คืออะไร , การศึกษาการใช้งาน Xbee ในเบื้องต้น , กำลังส่ง สายอากาศ และสัญญาณรบกวนของ Xbee, Xbee Association , Xbee Addressing , Xbee Operation Mode

3.1 ZigBee

ZigBee [4] เป็นเทคโนโลยีการสื่อสารแบบไร้สายที่ถูกพัฒนาให้มีจุดเด่นกว่าเทคโนโลยีไร้สายแบบอื่นๆ เช่น ราคาต่ำ ใช้พลังงานน้อยสามารถติดตั้งไว้ได้นานและสามารถสร้างเครือข่ายได้ซึ่งหมายความว่า กับการใช้งานด้านเซนเซอร์ไร้สาย ตรวจสอบตำแหน่งของวัตถุ ตรวจสอบสภาพแวดล้อม ดังตารางที่ 3.1 และรูปที่ 3.1

ตารางที่ 3.1 การเปรียบเทียบเทคโนโลยีไร้สายในแบบต่างๆ [4]

| Standard | ZigBee® 802.15.4 | Wi-Fi™ 802.11b | Bluetooth™ 802.15.1 |
|-----------------------------|----------------------|-------------------|------------------------|
| Transmission Range (meters) | 1 – 100 | 1 – 100 | 1 – 10 |
| Battery Life (days) | 100 – 1,000 | 0.5 – 5.0 | 1 – 7 |
| Network Size (# of nodes) | > 64,000 | 32 | 7 |
| Application | Monitoring & Control | Web, Email, Video | Cable Replacement |
| Stack Size (KB) | 4 – 32 | 1,000 | 250 |
| Throughput kb/s) | 20 – 250 | 11,000 | 720 |
| | | | |



รูปที่ 3.1 ปริมาณการใช้ Throughput ของมาตรฐานการสื่อสารไร้สายแบบต่างๆ [4]

3.1.1 คุณสมบัติของ ZigBee

- 3.1.1.1 อัตราการส่งข้อมูล 250 kbps (2.4 GHz), 40 kbps (915 MHz), and 20 kbps (868 MHz)
- 3.1.1.2 High throughput และ low latency Duty Cycle ต่ำ (< 0.1%)
- 3.1.1.3 มีการเข้าถึง Channel แบบ Channel access using Carrier Sense Multiple Access with Collision Avoidance (CSMA - CA)
- 3.1.1.4 สามารถรองรับ Address ได้ถึง 64 bit IEEE address (65535 network)
- 3.1.1.5 รับประกันการส่งแบบ Full handshaked protocol
- 3.1.1.6 เขื่อมต่อ Topology ได้หลายแบบ เช่น Star, Peer-to-peer และ Mesh
- 3.1.1.7 ใช้พลังงานต่ำ (สามารถใช้ได้หลายเดือนจนถึงปี)
- 3.1.1.8 ระยะทางการส่งพื้นฐาน 5-500 เมตร

3.1.2 โครงสร้างของ ZigBee

มาตรฐานของ ZigBee มีการแบ่งเป็น layer ชั้น layer เหล่านี้จะทำให้การใช้งานได้มีประสิทธิภาพมากขึ้น ราคาถูกติดตั้งง่าย การส่งข้อมูลที่น่าเชื่อถือ ใช้พลังงานน้อย แบ่งเป็น Layer ต่างๆ ดังนี้

3.1.2.1 Network layer

Network layer ถูกออกแบบมาเพื่อทำให้การส่งข้อมูลใน network ใช้พลังงานไม่นัก สามารถจัดการกับ Network ที่มีจำนวนโอนมากๆ โดยใช้ latency ทำหน้าที่ดังนี้

1. Starting a network สามารถสร้าง network ขึ้นใหม่ได้
2. Joining and leaving a network สามารถเข้าร่วมและออกจาก network ได้
3. Configuring a new device สามารถกำหนดค่าของ stack ได้
4. Addressing ZigBee coordinator กำหนด address ให้กับอุปกรณ์แต่ละตัวได้
5. Synchronization within a network สามารถติดต่อกับอุปกรณ์อื่นๆแบบ synchronization ได้
6. Security ทำให้เพิ่มรับส่งมีความปลอดภัย
7. Routing จัดหนาแน่นทางของเพิ่มปลายทาง

3.1.2.2 Application layer

Application layer ของ ZigBee ประกอบด้วย ZigBee Device Object (ZDO), User-defined application profile(s) and the Application Support (APS) sub-layer.

ZigBee Device Object (ZDO) ทำหน้าที่ดังนี้

1. Discovery สามารถค้นหาและระบุได้ว่าอุปกรณ์ตัวใดติดต่อกับอุปกรณ์ตัวใดอยู่
2. Binding สามารถจับคู่อุปกรณ์ไว้คู่กันได้โดยใช้ table of binding และ forward messages ระหว่างอุปกรณ์

ZigBee Device Object (ZDO) ทำหน้าที่ดังนี้

1. กำหนดหน้าที่ของอุปกรณ์แต่ละตัวภายใน network เช่น ZigBee coordinator or end device
2. สร้างหรือตอบสนอง binding requests
3. สร้างระบบรักษาความปลอดภัยระหว่าง network device โดยเลือกจาก ZigBee's security methods เช่น public key, symmetric key

User-defined application หมายถึง end device ซึ่งเป็นไปตามมาตรฐาน ZigBee

3.1.2.3 Physical layer

Physical layer IEEE802.15.4 ถูกออกแบบมาเพื่อลดต้นทุนของความต้องการ โดยการใช้วิธี direct sequence ซึ่งทำให้วงจรไฟฟ้ามีความจำากขึ้น ทำให้ราคาของการติดตั้งน้อยลง Physical type device ที่จะช่วยลดต้นทุนของระบบมี 2 อย่างคือ full function devices (FFD) และ reduced function devices (RFD)

Full function device (FFD)

- สามารถพิงก์ชันได้ในทุกๆ topology

- สามารถทำเป็น network coordinator ได้
- สามารถทำเป็น coordinator ได้
- สามารถติดต่อได้กับทุกๆ device

Reduced function device (RFD)

- ทำได้เฉพาะใน topology แบบ star
- ไม่สามารถเป็น network coordinator ได้
- สามารถคุยกับ network coordinator เท่านั้น
- สะดวกในการติดตั้ง

IEEE 802.15.4 / ZigBee network ต้องการอย่างน้อย FFD ที่เป็น network coordinator อย่างน้อยหนึ่งตัว แต่ตัว Endpoint device จะต้องเป็น RFD เพื่อที่จะลดต้นทุน

3.2 Xbee คืออะไร

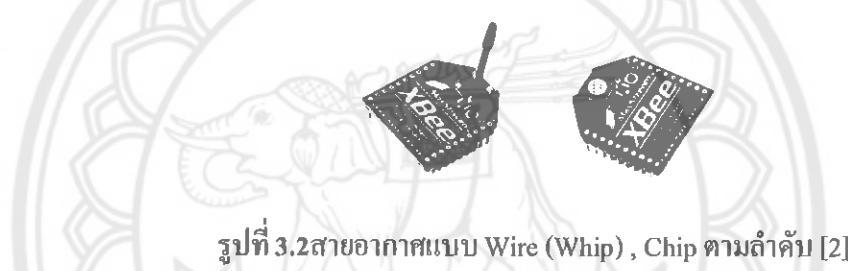
Xbee [4] เป็นอุปกรณ์ที่มี Microcontroller และ RF IC อยู่ภายใน ทำหน้าที่เป็นอุปกรณ์ Transceiver (อุปกรณ์รับ-ส่งสัญญาณ) แบบ Half Duplex ย่านความถี่ 2.4 Ghz มีการจัดการโดยใช้ พลังงานต่ำ ใช้งานง่ายมี Interface ที่ใช้รับและส่งข้อมูลกับ Xbee เป็น UART (TTL) ซึ่งสำหรับทางด้าน ไมโครคอนโทรลเลอร์ เราสามารถใช้ติดต่อสื่อสาร UART ของ Xbee ต่อเข้ากับ UART ของ ไมโครคอนโทรลเลอร์ได้เลย

Xbee สามารถใช้งานตามมาตรฐาน Zigbee ได้ โดยไม่ต้องเขียนโปรแกรมสร้าง เครือข่าย Zigbee เลย เพราะว่าทางผู้ผลิตได้จัดทำ Firmware ที่จะโหลดเข้าไปในตัว Xbee ให้เราสามารถ Set parameter ผ่าน Software interface (X-CTU หรือโปรแกรมที่เขียนขึ้นเอง) ผ่านทาง AT command (เหมือนกับการควบคุม GSM Module) โดยใช้ Hyper terminal หรือ ผ่านทางการรับส่งข้อมูลด้วย ไมโครคอนโทรลเลอร์ ได้อย่างง่ายดาย โดยเมื่อ Set Xbee ให้ทำงานเป็นอุปกรณ์ในเครือข่าย Zigbee แล้ว เราจะเรียก Xbee แต่ละตัวว่าเป็น Node

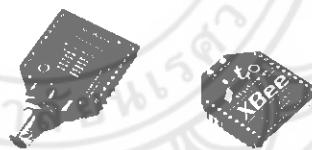
Feature Summary ของ Xbee โดยรวมที่เหมือนกัน

1. Operating Frequency ISM Band 2.4Ghz (ISM Band หมายถึง ย่านความถี่ใช้งานเพื่อการวิจัย ซึ่งจะอนุญาตให้ใช้กับ อุตสาหกรรม (Industrial) วิทยาศาสตร์ (Scientific) และ ทางการแพทย์ (Medical) รวมเป็น ISM)

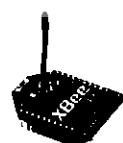
2. มีสายอากาศให้เลือกใช้หลายแบบ คือ แบบ Chip Ant , Whip Ant , UFL con , RPSMA con โดย 2 แบบหลัง เราต้องไปหาเสาอากาศย่าน 2.4Ghz ที่เป็น connector แบบ UFL หรือ SMA ครับ
3. Supply Voltage อยู่ที่ 2.8-3.4 V
 4. Power Down Current <10uA
 5. มี RF data rate อยู่ที่ 250 Kbps (เป็นส่วนของ สัญญาณที่ส่งผ่านอากาศ)
 6. มี Serial interface data rate อยู่ระหว่าง 1200 – 115200 Bps (เป็นส่วนที่ติดต่อสื่อสารกับไมโครคอนโทรลเลอร์)
 7. เป็น Spread Spectrum ชนิด DSSS (Direct Sequence)
 8. การกำหนด addressing มีลำดับลักษณะคือ กำหนด PAN ID สำหรับเครือข่ายหนึ่ง ๆ , กำหนด Channel และ กำหนด address ของแต่ละตัว



รูปที่ 3.2สายอากาศแบบ Wire (Whip) , Chip ตามลำดับ [2]



รูปที่ 3.3สายอากาศแบบ RPSMA Connector , UFL Connector ตามลำดับ [2]

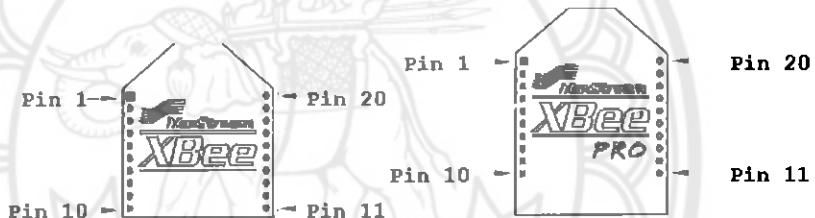


รูปที่ 3.4Xbee Series1 Pro Whip ant [2]

Xbee จะมีอยู่ 20 ขา ยกตัวอย่างจาก datasheet เป็นของรุ่น Xbee Series1 (Digital ซึ่งเป็นบริษัทผู้ผลิต Xbee) อธิบายหน้าที่แต่ละขาตามตาราง Datasheet Xbee ข้างมีขาที่เป็น Digital I/O และ Analog to Digital ขนาด 10 bits อีกด้วย

แต่เราไม่สามารถนำ Xbee ต่อลง Protoboard ได้โดยตรง เพราะความห่างช่องขาของ Xbee แคบกว่าช่อง Protoboard ท่านจะต้องทำ PCB ขึ้นมาเพื่อต่อใช้งานเลย, หรือไม่ก็หา Socket แปลงขาใช้งาน เป็นระยะห่างมาตรฐาน (Pitch) เท่ากับ 2.54 mm หรือ 1 mil

Xbee จะมีอยู่ 2 รุ่นคือ รุ่น series 1 และ รุ่น series 2 (ZB) และขั้งมีขนาด power ให้เลือกอีก 2 แบบ คือ แบบธรรมดา (1 mw – 2 mw) และ แบบ PRO (50mw- 60 mw) ซึ่งจะมีผลเรื่องระยะทางการรับส่งข้อมูล โดยแต่ละ series นั้น สามารถสร้างเครือข่ายได้หลายแบบ แต่จะมีเพียง series 2 เท่านั้นที่จะทำเครือข่ายแบบ mesh ได้ ซึ่งขั้งมีรายละเอียดปลีกย่อยในเรื่องของความแตกต่างในแต่ละ series



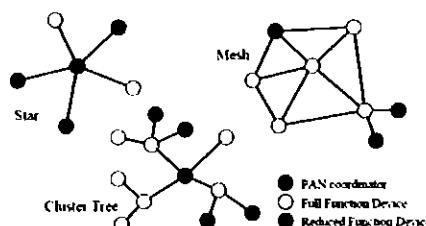
รูปที่ 3.5 Datasheet ของ Xbee [2]

Xbee ทั้ง 2 Series นี้สามารถสร้าง Topology ได้ดังนี้

รุ่น series 1: Peer-to-peer, point-to-point, point-to-multipoint (Broadcast)

รุ่น series 2 (ZB) : Mesh, Peer-to-peer, point-to-point, point-to-multipoint (Broadcast)

หมายเหตุ สำหรับ Xbee series2 ที่ทำ mesh ได้ จะมี parameter มากกว่า series 1 หากใช้ Series2 ส่วนแบบ point-to-point จะวุ่นวายกว่า Series1



รูปที่ 3.6 เครือข่าย Zigbee แบบ Star , Cluster , Mesh [2]

peer-to-peer network หมายถึง เครือข่ายที่อยู่ในระดับชั้นเดียวกัน ยกตัวอย่าง ใน OSI Layer เช่น ระดับ Transport Network Layer กับ Transport Network Layer นั่นก็คือ TCP Protocol ระหว่างคอมพิวเตอร์ 2 เครื่อง แต่สำหรับ XBee คำว่า peer-to-peer network หรือ Non Beacon Network คือ การที่เรา set node เป็น End Device หมดทุกตัว ไม่มีการกำหนดตายตัวว่า ตัวใดจะเป็น Master ตัวใดจะเป็น Slave แต่จะให้ระบบจัดการกันเอง โดยในเครือข่าย จะต้องกำหนด parameter ID (PAN ID) และ CH (Channel) XBee สามารถ Set ให้เป็น End-Device , Router , Coordinator ตามเงื่อนไขของ Zigbee

3.3 การศึกษาการใช้งาน XBee ในเบื้องต้น

เราสามารถทดสอบด้วยการปรับ parameter ที่สำคัญต่างๆ ผ่าน software user interface ได้ สามารถ download software user interface ที่ใช้ร่วมกันกับ XBee ชื่อ X-CTU มาได้ฟรีจาก Digi การใช้งาน สามารถอ่านจากคู่มือ X-CTU Configuration & Test Utility Software User Guide

นอกจากใช้ software แล้ว ต้องมีอุปกรณ์ที่จะเชื่อมต่อ XBee เข้ากับคอมพิวเตอร์เพื่อทำการติดต่อสื่อสารกับ X-CTU ด้วย อุปกรณ์ที่ว่านี้ คือตัวที่จะนำข้อมูลของ XBee มาต่อเข้ากับ max232 เพื่อเปลี่ยนระดับสัญญาณ TTL ให้สามารถติดต่อสื่อสารกับคอมพิวเตอร์ผ่าน RS232 (DB9) ได้ หรือเรา จะใช้ FT232RL สำหรับแปลง serial เป็น USB ในกรณีที่ computer ไม่มีพอร์ต DB9

ตัวอย่างอุปกรณ์ที่ใช้เชื่อมต่อกับคอมพิวเตอร์เพื่อ Update , Config Parameter โปรแกรม firmware ใหม่ และสำหรับใช้ทำการทดสอบเบื้องต้น เช่น XBee Socket , XBee USB Dongle , XBee Breadboard , XBee RS232 (DB9) Dongle อุปกรณ์เหล่านี้ถือว่าเป็นอุปกรณ์เสริมที่ทดแทนงานทางด้าน Hardware ได้ส่วนหนึ่ง ในระยะเริ่มต้นก็จะสะดวก แต่หากต้องการจะทำ Hardware ขึ้นเอง ก็สามารถดูวงจร Schematic จาก Digi

3.4 กำลังส่ง สายอากาศ และสัญญาณรบกวน ของ XBee

XBee นั้น ใช้บานความถี่ 2.4Ghz ซึ่งเป็นบานเดียวกันกับ Bluetooth หรือ Wireless Lan ดังนั้น หลายคนอาจจะสงสัยว่า สัญญาณมันจะกวนกันหรือไม่ คำตอบคือ กวนกันครับ ยกตัวอย่าง เช่น เราเมื่owireless lan ใช้ Access Point 1 ตัว และเราจะใช้เครื่องโทรศัพท์แบบ ถือไปมาได้ (2.4 GHz cordless telephones) ซึ่งก็ใช้บาน 2.4Ghz เช่นกัน แล้วเราจะทดสอบใช้งาน XBee ผลปรากฏว่า การรับส่งสัญญาณ ก็ขาดหายไปบ้าง แต่บางครั้งก็ครบถ้วน นั่นอาจเป็นเพราะกำลังส่งของ XBee และระยะทางของ node ที่เราทดสอบใกล้กันมาก

ช่องสัญญาณย่านความถี่ 2.4 Ghz นี้เรียกว่าเป็นย่านไมโครเวฟ หลักสำคัญของย่านไมโครเวฟอย่างหนึ่งคือ การวางแผนตัวรับส่งสัญญาณนั้น ต้องตั้งแบบ line of sight (ไม่มีสิ่งกีดขวางใด ๆ) ดึงจะได้ дальสูงสุด สำหรับ дальสูงของ Xbee ในรุ่น Pro จะใช้ 50-60 mw ใน datasheet บอกว่าได้ไกลถึง 1.5 km. แต่ที่ต้องเป็นลักษณะของ line of sight ครับ หากไม่ใช่เช่นนี้ เราจะได้ระบบการรับส่งสัญญาณที่ลดลง นอกเหนือจากเรื่อง line of sight แล้ว ยังมีเรื่องสัญญาณรบกวนต่าง ๆ (Interference)

เรื่องของสายอากาศ ตัว Xbee มีให้เลือกที่เป็นแบบสำเร็จรูปพร้อมใช้ ไม่ต้องหาสายอากาศมาต่อเพิ่มคือ สายอากาศแบบ chip ant และ whip ant ซึ่ง Pattern การแพร่กระจายคลื่นจะบ่งบอกเรอย่างหนึ่ง(จากหลาย ๆ อายุ) ได้ว่า สายอากาศนี้ จะมีอัตราการขยายตื้อย่างไร สามารถดูเอกสารจาก Digi

Chip ant นั้น ก็มีข้อดีตรงที่มันทำให้ขนาด Dimension รวมมันเล็กลง แต่ Gain น้อยกว่าแบบ Whip ant Chip ant ซึ่งมีระยะรับส่งข้อมูลที่ลดลงจาก spec ใน datasheet บอกว่าย่างเขิน รุ่น Pro ที่บอกว่าสามารถส่งได้ไกลสูงสุด 1.5 km แบบ line of sight แต่ถ้าเราเลือก chip ant แล้ว จะได้ระยะสูงสุดอยู่ที่ 500 กว่าเมตร

| Module | Antenna Type | Outdoor Distance (Visual Line-of-Sight) | Indoor Distance (Office Building) | Indoor Distance (Warehouse) |
|----------|--------------|---|-----------------------------------|-----------------------------|
| 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) | 366 ft (108 m) |

รูปที่ 3.7 ระบบการรับส่งโดยใช้สายต่างชนิดกัน [2]

3.4.1 XBee Association

ในเครือข่าย Zigbee ต้องมีการทำงานในโหมดประหยุดพักงาน ในช่วงเวลาที่ไม่มีการทำงานรับส่งข้อมูล ดังนั้นตัว Xbee จึงมี Parameter ที่จะกำหนดการทำงานสำหรับ Sleep mode อยู่ (Parameter A1 , A2 , SP , ST)

3.4.2 XBee Addressing

ตัว Xbee จะสามารถกำหนดค่าประจำตัวอ้างอิงของมัน (Address) 2 แบบ คือ แบบ 16 bit address และ 64 bit address ปกติแล้ว Xbee ทุกตัวจะถูกกำหนดค่ามาจากโรงงานเป็น Address 64 bit อยู่แล้ว ซึ่งจะสามารถอ่านค่าได้จาก parameter SH+SL การใช้งาน Address 64 bit สามารถทำได้โดยกำหนดค

parameter MY ให้มีค่า 0xFFFF หรือ 0xFFFE ส่วน การกำหนด 16 bit addressนั้นทำได้โดย กำหนด parameter MY ให้มีค่าสูงกว่า 0xFFFFE โดยจะเรียกเป็น mode การทำงาน 2 ประเภทคือ

1. Unicast Mode กือ การรับส่งข้อมูล โดยอาศัยหลักการ Acknowledgement กือหากทางค้าน ส่งนั้น ส่งข้อมูลไป แต่ไม่ได้รับ Ack ตอบกลับจากตัวรับ ก็จะทำการส่งข้อมูลใหม่

2. Broadcast Mode กือการส่งข้อมูลไปยังปลายทางให้ได้รับข้อมูลทุกตัว

3.4.3 Xbee Operation Mode

Xbee จะสามารถแบ่งช่วงการทำงานได้เป็น 5 แบบ กือ

1. Idle Mode โหมดนี้ จะเป็นโหมดที่ไม่ได้รับส่งข้อมูล ด้วย Xbee เตรียมที่จะทำงานในโหมด อื่น ๆ ต่อไปทันที หากมีเงื่อนไขบางอย่าง

2-3. Transmit / Receive Mode (ผูกรวม 2 Mode) กือช่วงที่ Xbee มีการรับ หรือ ส่งข้อมูล โดย จะแบ่งลักษณะการทำงานย่อยออกเป็น Direct กับแบบ Indirect ,การกำหนด Address ต้นทางและ ปลายทาง , Clear Channel Assessment และ การตอบรับ Acknowledgement

4. Sleep Mode กือ ช่วงที่ Xbee อยู่ในสถานการณ์ทำงานหลังงานค้างที่สุด เมื่อไม่มีการใช้งาน

5. Command Mode กือ เป็นส่วนการปรับ parameter ของ Xbee ซึ่งจะมีการกำหนด 2 แบบกือ แบบ AT command กับแบบ API Command

บทที่ 4

วิธีดำเนินงาน

4.1 โครงสร้างของระบบ

ระบบการติดตามตำแหน่งวัตถุที่ได้ออกแบบไว้ประกอบไปด้วย 2 ส่วน ใหญ่ๆ ดังนี้

1. Reference Node เป็นส่วนที่เป็นจุดอ้างอิง ส่วนนี้เป็นจุดที่อ้างอิงถึงความแรงของสัญญาณ ต้องติดตั้งตามจุดต่างๆ 3 จุด

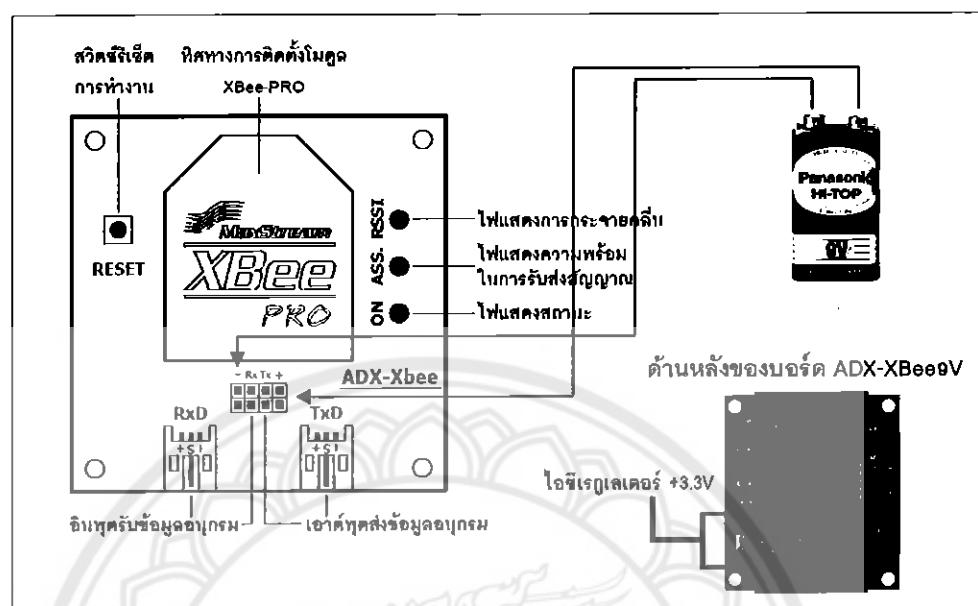
2. Object Node & Monitor Node เป็นส่วนที่อยู่กับติดวัตถุที่ต้องการทราบตำแหน่ง (ในที่นี่ได้ทำการติดกับคอมพิวเตอร์โน๊ตบุ๊ก) ส่วนนี้ใช้หาค่าความแรงของสัญญาณเทียบกับแต่ละโนดอ้างอิงและส่งค่าความแรงค้างกล่าวไปยังคอมพิวเตอร์ เพื่อใช้ในการคำนวณและแสดงผลตำแหน่งของวัตถุ โดยโปรแกรมแสดงผลลัพธ์ผ่านทางคอมพิวเตอร์

4.2 การทำงานของระบบ

4.2.1 วงจร Reference Node

วงจนี้ทำหน้าที่เป็นจุดอ้างอิงถึงความแรงของสัญญาณกับตัววัตถุ ซึ่งจะต้องติดตั้งตามจุดต่างๆ เป็นจำนวน 3 จุด เมื่อ Object Node มีการใช้คำสั่ง ATND ร้องขอความแรงของสัญญาณ Reference Node จะส่งสัญญาณกลับไป วงจนี้ต้องป้อนแรงดันกระแสดร่องขนาด 9 โวลต์เข้าทางคอนเนคเตอร์ 2 ขา IC LM11173.3 จะทำหน้าที่แปลงแรงดันจาก 9 โวลต์เป็น 3.3 โวลต์แล้วจับไฟต่อให้ ZigBee ทำงาน หลอดไฟ LED สีแดงจะติดสว่างเมื่อมีแรงดัน 9 โวลต์เข้ามา หลอดไฟ LED สีเหลืองเป็นการบอกสถานะของ ZigBee จะระบุว่าทุกๆ 1 วินาที เมื่อ ZigBee ทำงาน

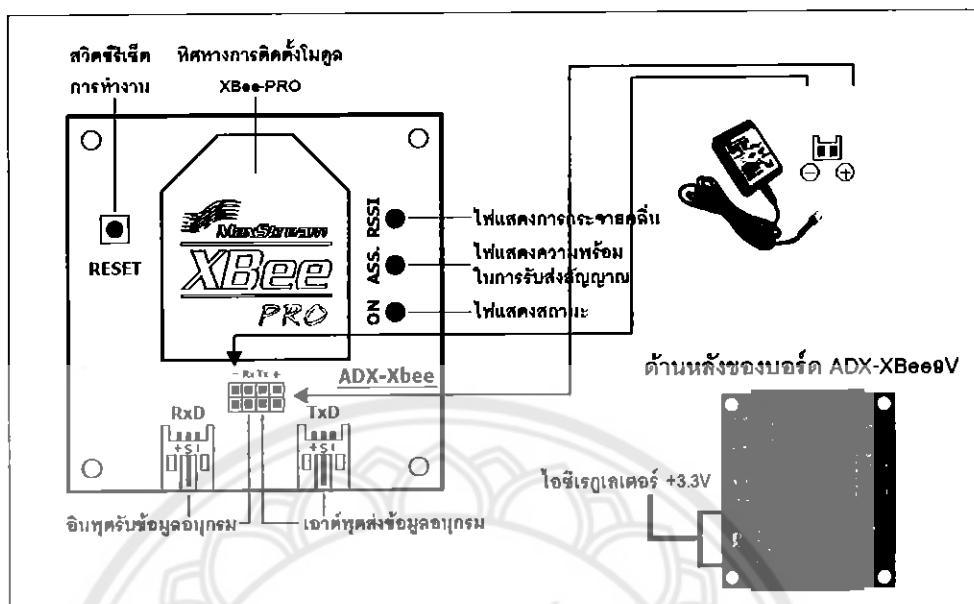
วงจนี้สามารถใช้ไฟจาก Adepter ที่ทำการแปลงไฟบ้าน 220V เป็น DC 5V มาใช้ในการป้อนกระแสไฟฟ้าได้ ซึ่งจะให้กระแสไฟที่คงที่กว่าการใช้แบตเตอรี่ อุปกรณ์เราสามารถใช้ไฟเลี้ยงวงจรได้จาก 2 แหล่งแล้วแต่การนำไปใช้งาน ซึ่งถ้าต้องการค่าความเข้มของสัญญาณที่นิ่ง ก็จะต้องใช้ไฟจากตัว Adepter แปลงไฟเพื่อทำการจ่ายกระแสไฟฟ้าอย่างสม่ำเสมอ แต่ถ้าต้องการความสะท้อนในการติดตั้งก็สามารถใช้ไฟจากแบตเตอรี่แทน ได้เช่นกัน ซึ่งคุณภาพของการรับส่งสัญญาณจะคล่องกว่าการใช้ไฟจาก Adepter



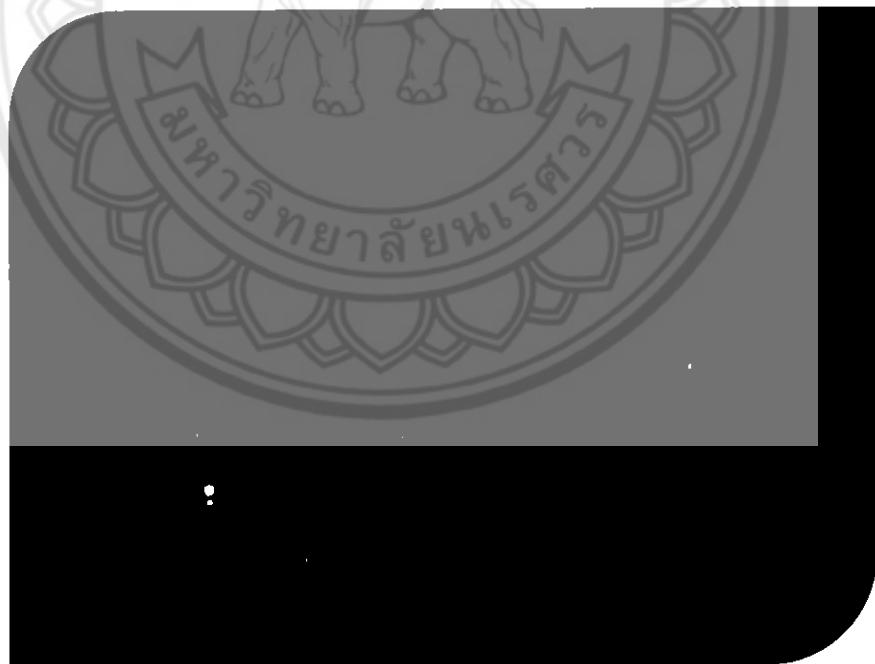
รูปที่ 4.1 วงจร Reference Node แบบใช้แบบตเตอรี่



รูปที่ 4.2 วงจร Reference Nodeแบบใช้แบบตเตอรี่ (ที่ใช้ทดลอง)



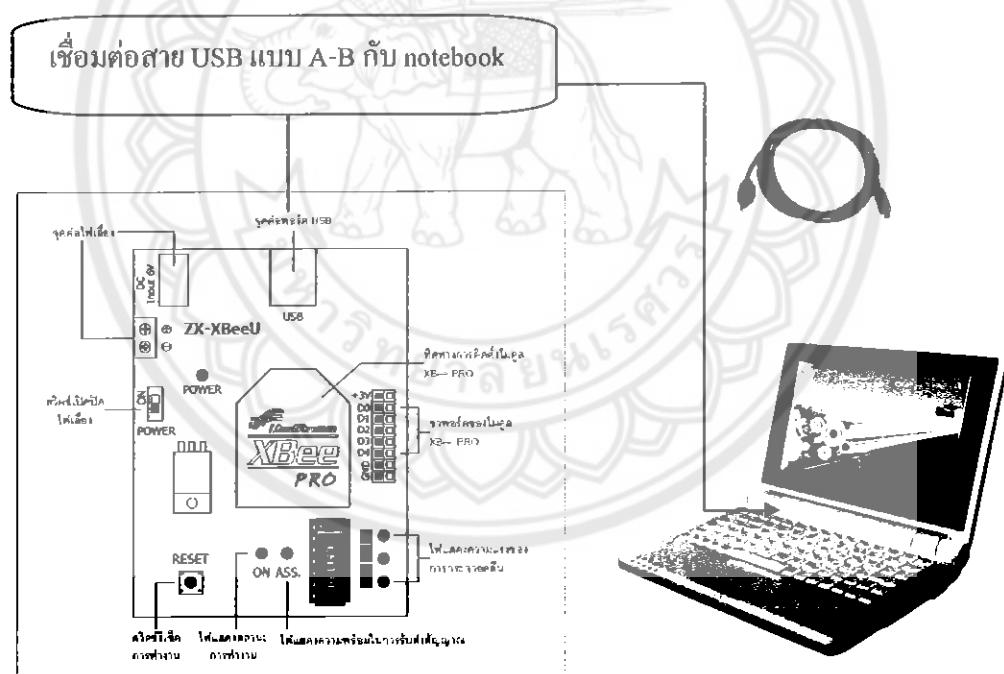
รูปที่ 4.3 วงจร Reference Nodeแบบใช้ไฟจาก Adapter 5V



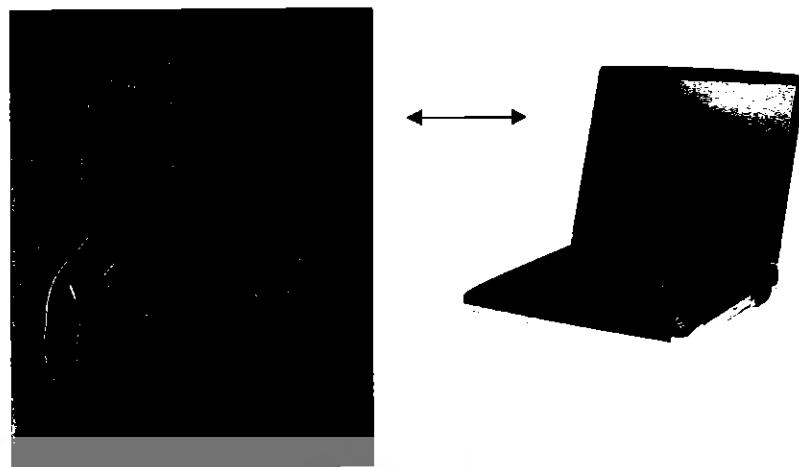
รูปที่ 4.4 วงจร Reference Nodeแบบใช้ไฟจาก Adapter 5V (ที่ใช้ท็อกล่อง)

4.2.2 วิธีการ Object Node & Monitor Node

วงจรนี้มีหน้าที่ควบคุมการทำงานของ ZigBee ที่ติดอยู่กับวัตถุที่ต้องการทราบตำแหน่ง(ในที่นี้ให้ติดกับ Notebook) โดยหาค่าความแรงของสัญญาณที่ยิงกับแต่ละโนดอ้างอิงและส่งค่าความแรงดังกล่าวไปยังคอมพิวเตอร์ค่าความแรงที่ส่งจาก Reference Node เข้าคอมพิวเตอร์ผ่านทางพอร์ต USB เพื่อมาคำนวณหาตำแหน่งของวัตถุผ่านโปรแกรมที่ได้พัฒนา การติดต่อระหว่าง ZigBee กับคอมพิวเตอร์ผ่านทางพอร์ต USB จะต้องมี IC FT232RL เพื่อปรับแรงดันให้เป็นแบบ TTL ได้ ขาส่งข้อมูลของ ZigBee จะตอกับขารับข้อมูลของคอมพิวเตอร์ และขารับข้อมูลของ ZigBee จะตอกับขาส่งข้อมูลของคอมพิวเตอร์ วงจรนี้ต้องการแรงดัน 3.3 โวลต์ป้อนเข้าที่ขา VCC และ GND ของ ZigBee ไฟ LED สีเขียวจะติดสว่างเมื่อแรงดันเข้ามา 3.3 โวลต์ ไฟ LED สีแดงเป็นการบอกสถานะของ ZigBee จะกระพริบทุกๆ 1 วินาทีเมื่อ ZigBee ทำงาน

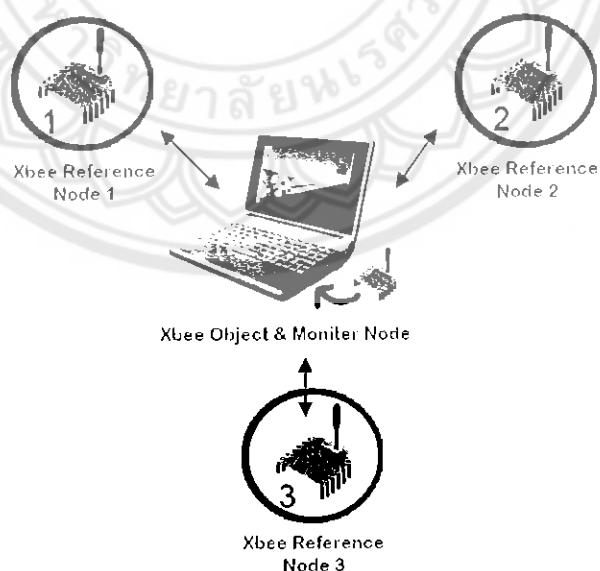


รูปที่ 4.5 วงจร Object Node



รูปที่ 4.6 วงจร Object Node(ที่ใช้ทดลอง)

สรุปการทำงานของวงจรทั้งสองวงจร เริ่มจากวงจร Object Node ที่ติดอยู่กับ Notebook ร้องขอค่าความแรงของสัญญาณจากการของ Reference Node ทั้งสามจุด วงจร Reference Node จะส่งสัญญาณกลับไปทำให้ Object Node ได้ค่าความแรงของสัญญาณมา แล้วส่งค่าความแรงของสัญญาณไปยังคอมพิวเตอร์เพื่อที่จะนำค่าความแรงไปคำนวณหาตำแหน่งของวัตถุผ่านทางโปรแกรมแสดงผลดังรูปที่ 4.5



รูปที่ 4.7 ภาพรวมการทำงานของระบบ

4.3 การหาความแรงของสัญญาณ

เราใช้คอมพิวเตอร์ในการคำนวณหาระยะทางจากการรับค่าความเข้มของสัญญาณ โดยคอมพิวเตอร์จะทำหน้าที่ควบคุมการทำงานของ ZigBee ที่ติดต่ออยู่กับคอมพิวเตอร์ โดยทำการร้องขอค่าความแรงของสัญญาณจาก Reference Node ทั้ง 3 จุด เมื่อ Reference Node แต่ละจุดได้รับการร้องขอ ก็จะส่งค่าความแรงของสัญญาณกลับมาบัง Monitor Node โดยการส่งข้อมูลจะอยู่ในรูปแบบของ API Mode (ข้อมูลในรูปแบบ API Mode จะเป็น Hex Code ต้องทำการแปลงก่อนถึงจะทำไปใช้ได้)

4.3.1 การส่งข้อมูลในรูปแบบของ API

การส่งข้อมูลในรูปแบบ API (Application Programming Interface) คือการจัดเก็บข้อมูลที่ต้องการจะติดต่อสื่อสารกับโมดูลอื่นๆ ในรูปแบบโครงสร้างที่แน่นอน โดยจัดเก็บเป็น frame packet ซึ่งสามารถบอกได้ว่า packet ที่ส่งไปนั้นมีค่าสั่งอะไร การส่งข้อมูลถูกต้องหรือไม่ และมีสถานะของการส่งและการรับของ packet นั้นๆ ด้วย ซึ่งข้อมูลที่ส่งไปจะอยู่ในรูปแบบของเลขฐานสิบหก

4.3.2 รูปแบบของ API Frame

เราสามารถส่งข้อมูลในรูปแบบของ API Mode ได้โดยการตั้งค่าพารามิเตอร์ของค่าสั่ง AP (API Enable) ดังนี้

AP = 0 เป็นการตั้งค่าการส่งข้อมูลให้อยู่ในรูปแบบของ UART แบบปกติ API Mode จะถูกยกเลิก

AP = 1 เป็นการตั้งค่าการส่งข้อมูลให้อยู่ในรูปแบบของ API Mode

AP = 2 เป็นการตั้งค่าการส่งข้อมูลให้อยู่ในรูปแบบของ API Mode (Escaped Character)

ในโครงการที่เราพัฒนาเราใช้การส่งข้อมูลในรูปแบบของ API Mode (AP = 1)

API Mode (AP = 1)

เมื่อพารามิเตอร์ AP ถูกตั้งค่าเป็น 1 โครงสร้างของข้อมูลที่ส่งจะมีลักษณะดังรูปที่ 4.6

| Start Delimiter (Byte 1) | Length (Byte 2-3) | Frame Data (Byte 4-n) | Checksum (Byte n+1) |
|-----------------------------|----------------------|--------------------------|------------------------|
| 0x7E | MSB LSB | API-specific Structure | 1 Bytes |

รูปที่ 4.8 รูปแบบของ API Packet frame Mode 1 [4]

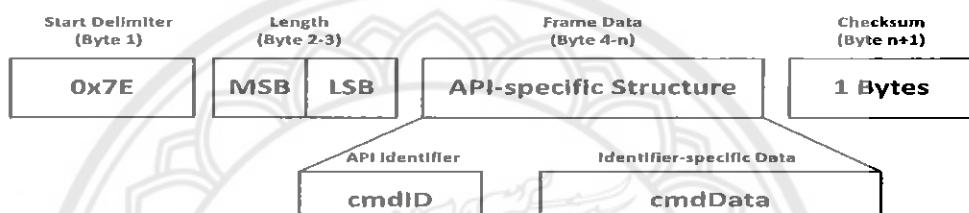
MSB = Most Significant Byte

LSB = Last Significant Byte

Checksum คือ ไบต์ที่ตรวจสอบความถูกต้องของข้อมูลที่ได้รับหรือส่ง สามารถคำนวณหาค่า checksum ได้โดยนำวง Frame Data ทุกไบต์ในลักษณะของเลขฐานสิบหก หลังจากนั้นนำไปต่อกันทีละ byte ไปจนออกจาก 0xFF ก็จะได้ค่า checksum ที่ถูกต้อง

4.3.3 ประเภทของ API

รูปแบบของการส่งข้อมูลมีโครงสร้างแพ็คเกตดังรูปที่ 4.7



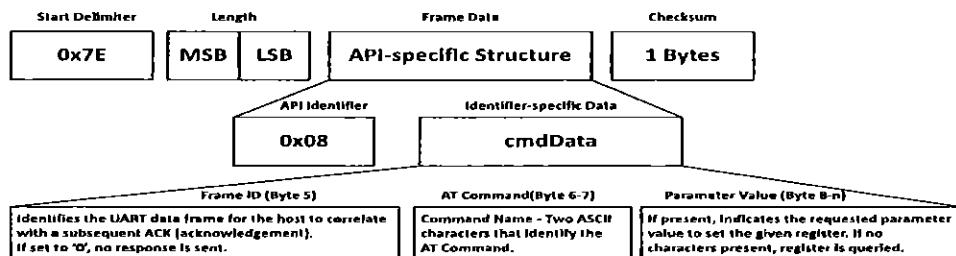
รูปที่ 4.9 รูปแบบของ API Packet frame[4]

cmdID (API Identifier) คือ ไบต์ที่เป็นตัวกำหนดรูปแบบของ API ว่าอยู่ในประเภทใด เช่น อยู่ในรูปแบบของ AT command การรับข้อมูล หรือการส่งข้อมูลเป็นต้น

cmdData (Identifier-specific Data) คือ รูปแบบของข้อมูลที่ต่างๆ ที่ถูกกำหนดโดย cmdID

4.3.3.1 AT Command

API frame ประเภท AT command เป็นแพ็คเกตที่สามารถใช้เปลี่ยนหรืออ่านค่าพารามิเตอร์ของโมดูล ZigBee ได้ ซึ่งใช้คำสั่ง AT Command ในรูปแบบนี้ไม่จำเป็นต้องอยู่ในโหมด AT command ค่าต่างๆ สามารถอ่านหรือแก้ไขได้ทันที โดยจะมีค่า API Identifier เป็น 0x08 โครงสร้างของ AT Command frame ดังรูปที่ 4.8



รูปที่ 4.10 รูปแบบของ API Packet frame ประเภท AT Command[4]

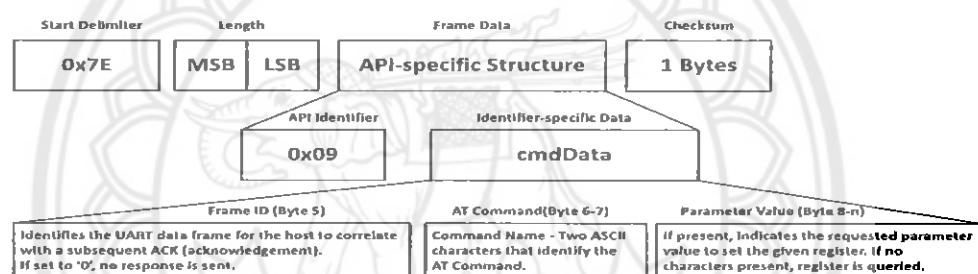
Frame ID คือ ค่า ID ของ frame ที่ส่งไป ถ้าถูกตั้งค่าเป็น 0 ACK จะไม่ถูกส่งกลับมา

AT Command คือ ชื่อของ AT Command ประกอบด้วยตัวอักษร 2 ตัว

Parameter Value คือ ค่าพารามิเตอร์ของ AT Command นั้นๆ ที่ต้องการตั้งค่า

4.3.3.2 AT Command Queue Parameter Value

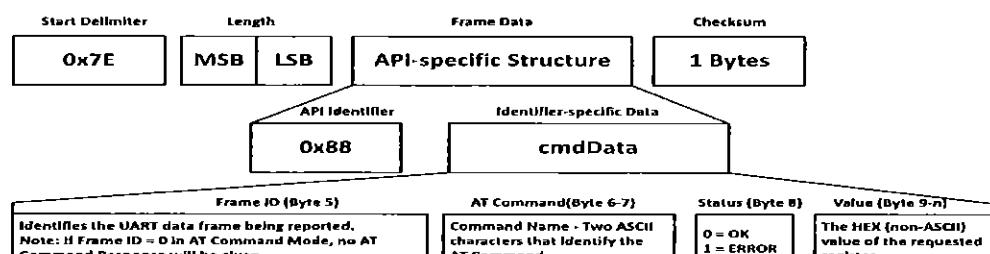
API frame ประเภทนี้จะเหมือนกับ API frame AT Command ซึ่งสามารถใช้เขียนหรืออ่านค่าพารามิเตอร์ของโมดูล ZigBee ได้โดยใช้ AT Command แต่ API frame ประเภทนี้จะทำงานก็ต่อเมื่อ AT Command (API Identifier =0x08) ทำงานเสร็จ ซึ่ง AT Command Queue Parameter Value มีค่า API Identifier เป็น 0x09 โครงสร้างของ API frame ประเภทนี้เป็นดังนี้



รูปที่ 4.11 รูปแบบของ API Packet frame ประเภท AT Command Queue Parameter Value [4]

4.3.3.3 AT Command Response

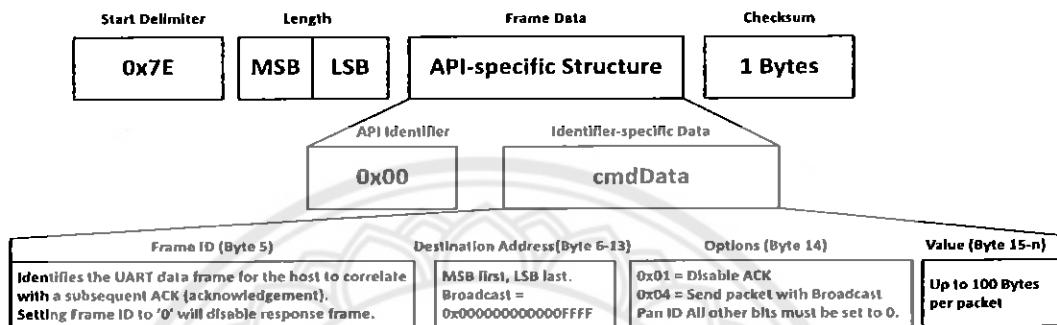
API frame ประเภทนี้เป็นผลจากการใช้ API frame AT Command ซึ่งจะแสดงผลลัพธ์ของข้อความ AT Command ที่ส่งไป บางคำสั่ง เช่น ND (Node Discovery) จะส่งผลลัพธ์กลับมาหลายๆ frame ซึ่ง frame สุดท้ายจะบอกสถานะของการส่ง API frame AT Command Response มีค่า API Identifier เป็น 0x88 โครงสร้างของ API frame ประเภทนี้เป็นดังนี้



รูปที่ 4.12 รูปแบบของ API Packet frame ประเภท AT Command Response [4]

4.3.3.4 Tx (Transmit) Request : 64-bit address

API frame ประเภทนี้จะส่งข้อมูลในรูปแบบของ packet ไปยังโนดุลปลายทาง โดยใช้ที่อยู่แบบ 64 bit และมี API Identifier เป็น 0x00 โครงสร้างของ API frame ประเภทนี้เป็นดังนี้



รูปที่ 4.13 รูปแบบของ API Packet frame ประเภท Tx (Transmit) Request : 64-bit address[4]

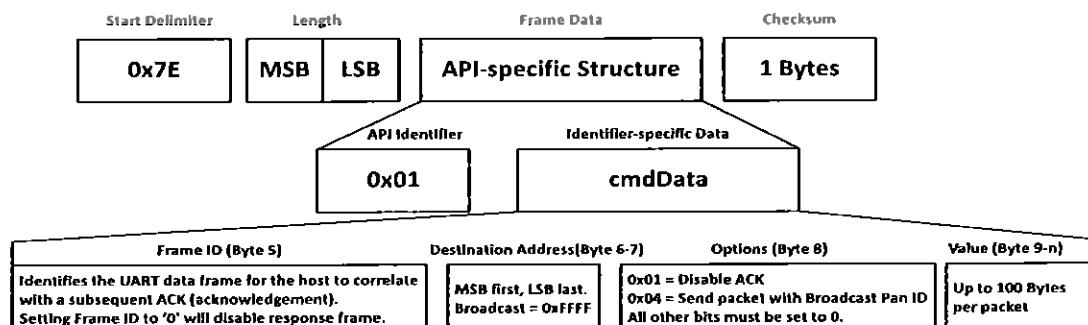
Destination Address กือ ที่อยู่ของโนดุลปลายทาง เมื่อถูกตั้งค่าเป็น 0x0000000000000000FFFF ข้อมูลจะถูกส่งแบบ broadcast

Option Byte กือ ตัวเลือกของการส่ง เมื่อถูกตั้งค่าเป็น 0x01 จะยกเลิกการส่ง ACK เมื่อตั้งค่าเป็น 0x04 จะส่ง packet ไปยังทุกโนดุลที่อยู่ Pan ID เดียวกัน

RF Data กือ ข้อมูลที่ต้องการส่ง

4.3.3.5 Tx (Transmit) Request : 16-bit address

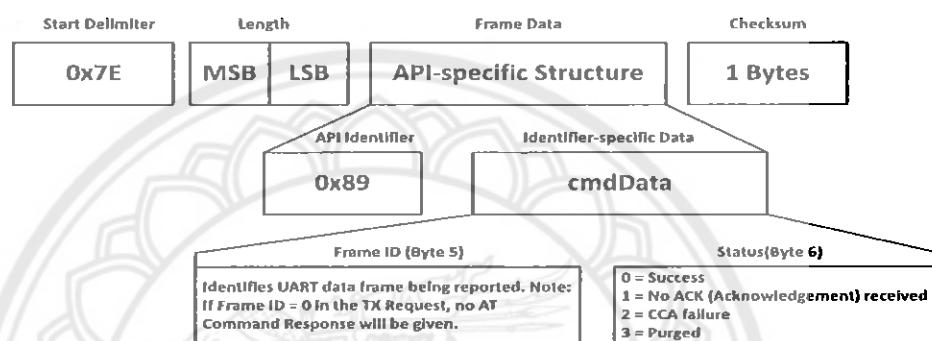
API frame ประเภทนี้จะส่งข้อมูลในรูปแบบของ packet ไปยังโนดุลปลายทาง โดยใช้ที่อยู่แบบ 16 bit และมี API Identifier เป็น 0x01 โครงสร้างของ API frame ประเภทนี้เป็นดังนี้



รูปที่ 4.14 รูปแบบของ API Packet frame ประเภท Tx (Transmit) Request : 16-bit address[4]

4.3.3.6 Tx (Transmit) Status

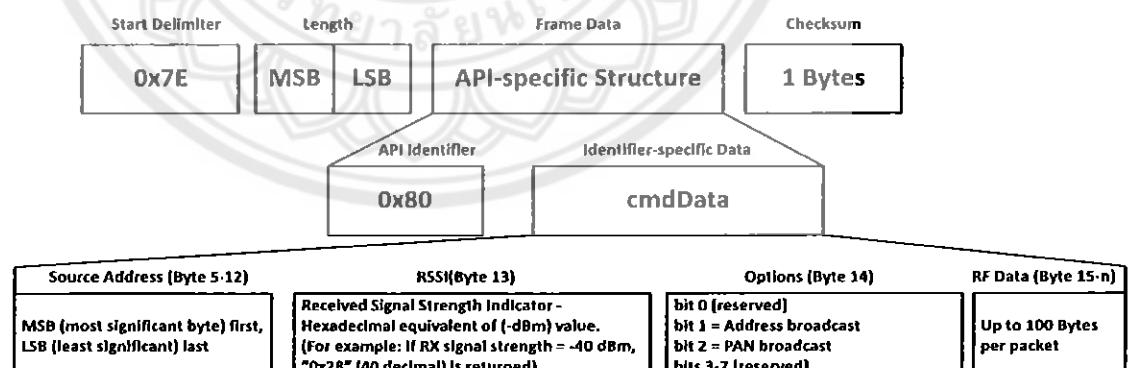
เมื่อการส่งข้อมูลเสร็จสมบูรณ์ โนดูลที่ได้รับข้อมูลจะส่งสถานะของการส่งกลับมาบังโนดูลที่ส่งข้อมูล API frame ประเภทนี้เป็นสถานะของการส่งข้อมูลว่าสำเร็จหรือไม่มี API Identifier เป็น 0x89 โครงสร้างของ API frame ประเภทนี้เป็นดังนี้



รูปที่ 4.15 รูปแบบของ API Packet frame ประเภท Tx (Transmit) [4]

4.3.3.7 RX (Receive) Packet 64-bit address

API frame ประเภทนี้ API Identifier เป็น 0x80 เมื่อ โนดูล ได้รับข้อมูลจากการส่งแบบที่อยู่ 64 bit แล้ว frame packet ที่ได้รับจะมีโครงสร้างดังนี้



รูปที่ 4.16 รูปแบบของ API Packet frame ประเภท RX (Receive) Packet 64-bit address[4]

Source Address กือ ที่อยู่ของโนดูลที่ส่งข้อมูลมาบังตัวรับ

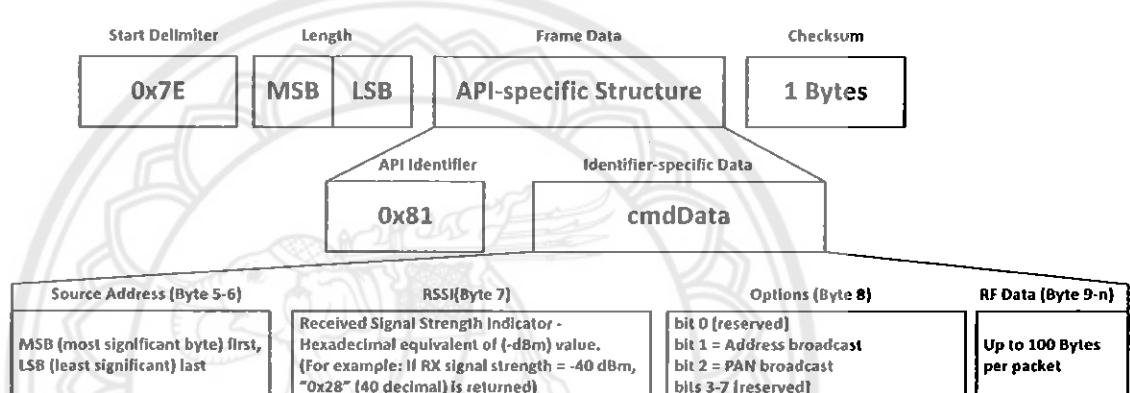
RSSI กือ ความแรงของสัญญาณ

Option Byte คือ รูปแบบของการส่ง ถ้าเป็น 0x01 จะเป็นการส่งแบบหนึ่งต่อหนึ่ง ถ้าเป็น 0x02 จะเป็นการส่งแบบ PAN broadcast

RF Data คือ ข้อมูลที่ได้รับ

4.3.3.8 RX (Receive) Packet 16-bit address

API frame ประเภทนี้มี API Identifier เป็น 0x81 เมื่อโ้มูลได้รับข้อมูลจากการส่งแบบที่อยู่ 16 bit แล้ว frame packet ที่ได้รับจะมีโครงสร้างดังนี้



รูปที่ 4.17 รูปแบบของ API Packet frame ประเภท RX (Receive) Packet 16-bit address[4]

4.3.4 ตัวอย่างการใช้ API frame packet

การใช้งานในโหมด API frame packet จะต้องตั้งค่าพารามิเตอร์ AP ให้เป็น 1 ก่อนจึงจะสามารถใช้งานโหมดนี้ได้ และข้อมูลที่อยู่ใน API frame packet จะอยู่ในรูปแบบของเลขฐานสิบหก

4.3.4.1 การส่งและรับคำสั่ง ATND จาก ZigBee

คำสั่ง ATND เป็นคำสั่งที่ใช้สำหรับแสดงรายการเบื้องต้นทุกๆ โนดที่อยู่ในระบบการส่งข้อมูล โดยสิ่งที่เราต้องการจากคำสั่ง ATND คือค่าความแรงสัญญาณของแต่ละโนดที่อยู่ในรัศมี ในการส่งคำสั่ง ATND ไปยัง ZigBee เราใช้ API Modeในการส่งคำสั่งซึ่งมีลักษณะดังนี้

ตารางที่ 4.1 การส่งคำสั่ง ATND

| Byte | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-------|----|----|----|----|----|----|----|----|
| Value | 7E | 00 | 04 | 08 | 52 | 4E | 44 | 13 |

เมื่อ API frame packet ดังกล่าวถูกส่งไปยัง ZigBee แล้ว ZigBee จะตอบกลับมาในรูปแบบของ API frame packet ดังนี้

- Byte ที่ 9-10 คือ MY (Source Address)
- Byte ที่ 11-14 คือ SH (Serial Number High)
- Byte ที่ 15-18 คือ SL (Serial Number Low)
- Byte ที่ 19 คือ DB (Receive Signal Strength)
- Byte ที่ 20 ขึ้นไปคือ NI (Node Identifier)

ตารางที่ 4.2 การตอบกลับมาหลังจากส่งคำสั่ง ATND

| Byte | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| value | 7E | 00 | 12 | 88 | 52 | 4E | 44 | 00 | FF | FF | 00 | 13 | A2 | 00 | 40 | 01 | 3B | 80 | 29 | 20 | 00 | 9B |

และ ACK ของคำสั่ง ATND อีก 9 Byte ดังนี้

ตารางที่ 4.3 ACK ของคำสั่ง ATND

| Byte | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-------|----|----|----|----|----|----|----|----|----|
| value | 7E | 00 | 05 | 88 | 52 | 4E | 44 | 00 | 93 |

4.4 การหาตำแหน่งวัตถุ

การหาตำแหน่งวัตถุจะใช้ค่าความแรงของสัญญาณ (RSSI) และทำการแปลงค่าที่ได้เป็นระยะทาง แล้วทำการใช้ทฤษฎีทางคณิตศาสตร์ที่เกี่ยวข้องมาคำนวณหาตำแหน่งดังนี้

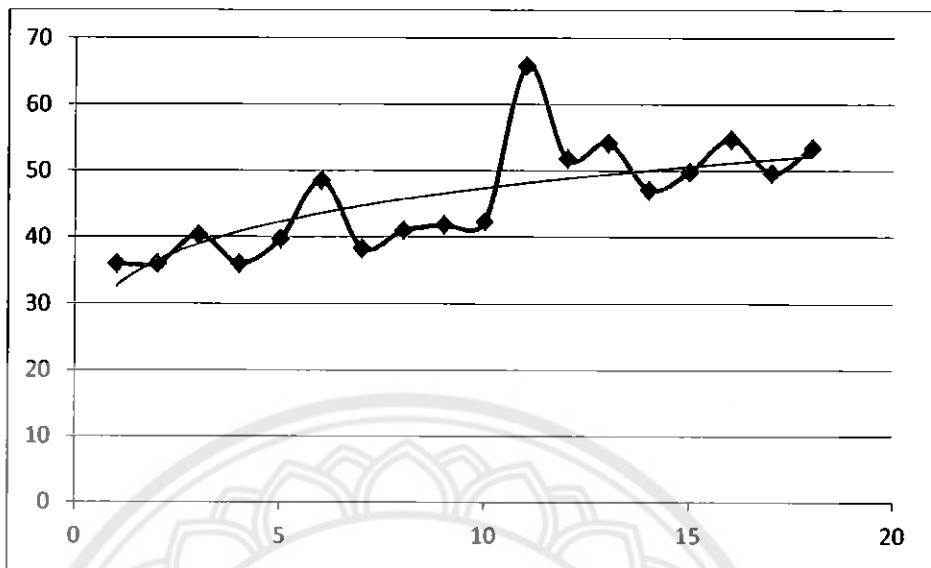
4.4.1 การแปลงระยะทางจากค่าความแรงของสัญญาณ

เราจะทำการเก็บค่าความแรงของสัญญาณ เพื่อนำมาสมการที่ใช้ในการคำนวณระยะทาง โดยการเก็บค่าความแรงโดยเลื่อนตำแหน่งเพิ่มทีละ 1 เมตรเป็นระยะทาง 20 เมตรแล้วนำมาหาค่าเฉลี่ย

แล้วใช้ Microsoft Excel สร้างกราฟ แล้วทำการดึงสมการจากกราฟที่ Plot ได้ เราจะได้สมการ
ระยะทางอุกมาดังนี้

ตารางที่ 4.4 การเก็บค่า RSSI มาใช้สร้างสมการระยะทาง

| ระยะทาง (เมตร) | RSSI (Min) | RSSI (Max) | RSSI (Average) |
|----------------|------------|------------|----------------|
| 1 | 36 | 36 | 36 |
| 2 | 36 | 39 | 36.06 |
| 3 | 38 | 42 | 40.4 |
| 4 | 36 | 39 | 36.06 |
| 5 | 37 | 41 | 39.76 |
| 6 | 45 | 51 | 48.56 |
| 7 | 37 | 40 | 38.34 |
| 8 | 40 | 42 | 41 |
| 9 | 41 | 42 | 41.86 |
| 10 | 42 | 47 | 42.38 |
| 11 | 61 | 72 | 65.8 |
| 12 | 49 | 61 | 51.84 |
| 13 | 50 | 57 | 54.24 |
| 14 | 47 | 49 | 47.2 |
| 15 | 48 | 55 | 49.90 |
| 16 | 52 | 57 | 54.78 |
| 17 | 49 | 52 | 49.68 |
| 18 | 52 | 55 | 53.52 |
| 19 | 52 | 57 | 55.86 |
| 20 | 49 | 50 | 49.98 |

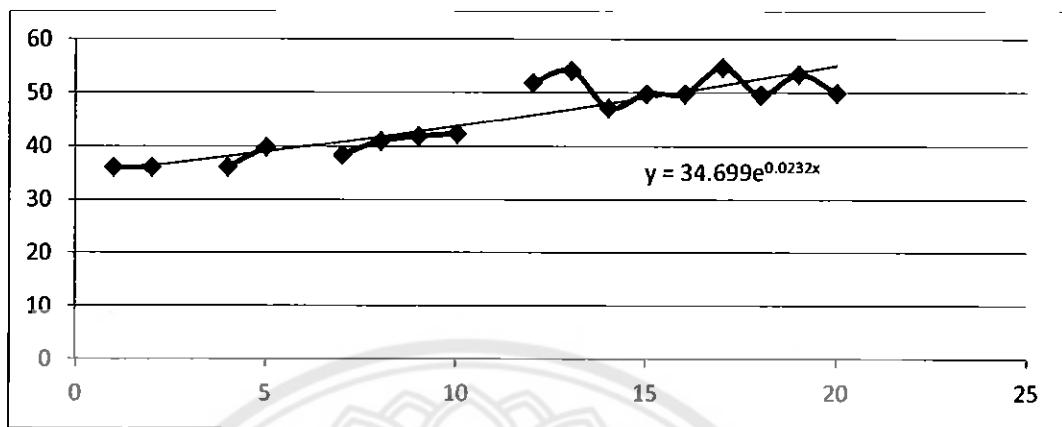


รูปที่ 4.18 กราฟแสดงค่าเฉลี่ยของระยะทางกับค่า RSSI

จะเห็นว่ามีบางจุดที่มีค่ากระโดด เราจึงต้องเลือกเฉพาะค่าที่มีความต่อเนื่องเท่านั้นมาใช้ในการสร้างสมการเส้นตรง(แต่ที่มีແບບสี่ม่วงคือค่าเฉลี่ยที่ถูกตัดทิ้ง)

ตารางที่ 4.5 การเลือกค่า RSSI มาใช้สร้างสมการระยะทาง

| ระยะทาง (เมตร) | ค่า RSSI (เฉลี่ย) | ระยะทาง (เมตร) | ค่า RSSI (เฉลี่ย) |
|----------------|-------------------|----------------|-------------------|
| 1 | 36 | 9 | 49.98 |
| 2 | 36.06 | 12 | 51.84 |
| 3 | 39.76 | 13 | 54.24 |
| 4 | 36.06 | 14 | 47.2 |
| 5 | 39.76 | 15 | 49.90 |
| 6 | 41.86 | 16 | 54.78 |
| 7 | 38.34 | 17 | 49.68 |
| 8 | 41 | 18 | 53.52 |
| 9 | 41.86 | 19 | 55.86 |
| 10 | 42.38 | 20 | 49.98 |



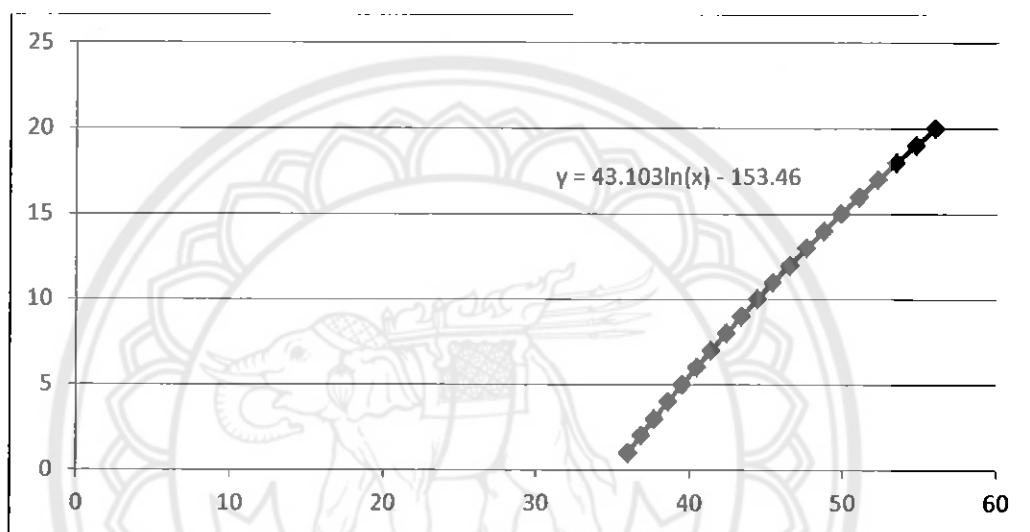
รูปที่ 4.19 กราฟแสดงค่าเฉลี่ยของระยะทางกับค่าRSSIโดยตัดข้อมูลที่กระโดดออก

จะเห็นว่าตอนนี้กราฟจะมีแนวโน้มไปทางเดียวกัน เมื่อเราลากเส้น Trendline แล้วให้ Excel สร้างสมการจากเส้น Trendline แล้วจะได้สมการคือ $y = 34.699e^{0.0232x}$ จากนั้นทำการแปลงสมการ $y = 34.699e^{0.0232x}$ ออกมาเป็นระยะทางอีกริ้ง

ตารางที่ 4.6 แปลงสมการ Trend Line กลับอุณหภูมิเป็นระยะทาง

| ระยะทาง (เมตร) | ค่า RSSI | ระยะทาง (เมตร) | ค่า RSSI |
|----------------|----------|----------------|----------|
| 1 | 36.0006 | 11 | 45.40107 |
| 2 | 36.84558 | 12 | 46.46668 |
| 3 | 37.71039 | 13 | 47.55731 |
| 4 | 38.5955 | 14 | 48.67354 |
| 5 | 39.50138 | 15 | 49.81597 |
| 6 | 40.42853 | 16 | 50.98521 |
| 7 | 41.37743 | 17 | 52.1819 |
| 8 | 42.34861 | 18 | 53.40667 |
| 9 | 43.34258 | 19 | 54.66019 |
| 10 | 44.35989 | 20 | 55.94313 |

หลังจากนั้นนำค่าที่ได้มา Plot กราฟ จะได้สมการระยะทางจากการแปลงค่า RSSI ซึ่งจะได้กราฟที่มีความต่อเนื่องของข้อมูล เมื่อเราลากเส้น Trendline จะได้สมการเส้นตรงที่สามารถนำไปใช้งานจริงได้



รูปที่ 4.20 กราฟแสดงค่า RSSI เทียบกับระยะทาง

เราจะได้สมการระยะทาง คือ $Y = 43.103 * \ln(X) - 153.46$ โดยที่ค่า Y คือ ระยะทางที่คำนวณได้ ค่า X คือ ค่า RSSI ที่ได้จาก XBee Reference Node

4.4.2 การใช้ Moving Average Filter ช่วยในการกรองค่าความเข้มสัญญาณ

เมื่อเราได้ค่าความเข้มสัญญาณจาก XBee Reference Node แล้ว เราจะต้องนำค่าเหล่านี้มากรอง อีกชั้นหนึ่ง เพราะค่า RSSI ที่ได้จาก XBee Reference Node บางครั้งอาจจะคลาดเคลื่อนจากความเป็นจริงสูง อันเกิดจากปัจจัยหลายอย่าง เช่น การหักเหของค่า RSSI เมื่อผ่านสิ่งกีดขวาง หรือการลดthon ของสัญญาณกว่าจะมาถึงตัวรับ เป็นต้น ปัจจัยเหล่านี้ทำให้ได้ค่าความเข้มสัญญาณที่ผิดเพี้ยนไป เราจึงต้องนำค่าเหล่านี้มากรองก่อนที่จะนำไปใช้ โดยการใช้ Moving Average Filter มาช่วยในการกรองข้อมูล ทำให้ข้อมูลมีความน่าเชื่อถือมากขึ้น

การคำนวณค่าตัวแปรต่างๆ ในสมการ Moving Average Filter

$$\text{จากสมการ} \quad Y[i] = \frac{1}{M} \sum_{j=0}^{M-1} X[i+j] \quad \text{สมการ (4.1)}$$

| | |
|----------|---------------------------------|
| $Y[i]$ | คือ ค่าRSSI ที่ได้ทำการกรองแล้ว |
| M | คือ จำนวนข้อมูลเฉลี่ยทั้งหมด |
| $X[i+j]$ | คือ ค่าRSSI ที่ต้องการนำมากรอง |

เราจะใช้ $M = 10$ เพราะเมื่อทำการทดลองแล้วค่า $M = 10$ ใช้เวลาค่อนข้างน้อยและข้อมูลเชื่อถือได้

ตัวอย่าง สมมุติมีข้อมูลความเข้มสัญญาณดังต่อไปนี้

- | | |
|---------------------------|---------------------------|
| 1. RSSI ครั้งที่ 1 = 35 | 13. RSSI ครั้งที่ 13 = 38 |
| 2. RSSI ครั้งที่ 2 = 35 | 14. RSSI ครั้งที่ 14 = 42 |
| 3. RSSI ครั้งที่ 3 = 36 | 15. RSSI ครั้งที่ 15 = 35 |
| 4. RSSI ครั้งที่ 4 = 35 | 16. RSSI ครั้งที่ 16 = 35 |
| 5. RSSI ครั้งที่ 5 = 34 | 17. RSSI ครั้งที่ 17 = 34 |
| 6. RSSI ครั้งที่ 6 = 36 | 18. RSSI ครั้งที่ 18 = 35 |
| 7. RSSI ครั้งที่ 7 = 35 | 19. RSSI ครั้งที่ 19 = 35 |
| 8. RSSI ครั้งที่ 8 = 34 | 20. RSSI ครั้งที่ 20 = 36 |
| 9. RSSI ครั้งที่ 9 = 34 | 21. RSSI ครั้งที่ 21 = 35 |
| 10. RSSI ครั้งที่ 10 = 35 | 22. RSSI ครั้งที่ 22 = 32 |
| 11. RSSI ครั้งที่ 11 = 34 | 23. RSSI ครั้งที่ 23 = 34 |
| 12. RSSI ครั้งที่ 12 = 34 | 24. RSSI ครั้งที่ 24 = 35 |

เมื่อ $I = 0$, $M = 10$

$$\begin{aligned} Y[0] &= (X[0]+X[1]+X[2]+X[3]+X[4]+X[5]+X[6]+X[7]+X[8]+X[9])/10 \\ &= (35+35+36+35+34+36+35+34+34+35)/10 \end{aligned}$$

$$= 34.9$$

เมื่อ $I = 1, M = 10$

$$\begin{aligned} Y[1] &= (X[1]+X[2]+X[3]+X[4]+X[5]+X[6]+X[7]+X[8]+X[9]+X[10])/10 \\ &= (35+36+35+34+36+35+34+34+35+34)/10 \\ &= 34.8 \end{aligned}$$

เมื่อ $I = 2, M = 10$

$$\begin{aligned} Y[2] &= (X[2]+X[3]+X[4]+X[5]+X[6]+X[7]+X[8]+X[9]+X[10]+X[11])/10 \\ &= (36+35+34+36+35+34+34+35+34+34)/10 \\ &= 34.7 \end{aligned}$$

เมื่อ $I = 3, M = 10$

$$\begin{aligned} Y[3] &= (X[3]+X[4]+X[5]+X[6]+X[7]+X[8]+X[9]+X[10]+X[11]+X[12])/10 \\ &= (35+34+36+35+34+34+35+34+34+38)/10 \\ &= 34.9 \end{aligned}$$

เมื่อ $I = 4, M = 10$

$$\begin{aligned} Y[4] &= (X[4]+X[5]+X[6]+X[7]+X[8]+X[9]+X[10]+X[11]+X[12]+X[13])/10 \\ &= (34+36+35+34+34+35+34+34+38+42)/10 \\ &= 35.6 \end{aligned}$$

เมื่อ $I = 5, M = 10$

$$\begin{aligned} Y[5] &= (X[5]+X[6]+X[7]+X[8]+X[9]+X[10]+X[11]+X[12]+X[13]+X[14])/10 \\ &= (36+35+34+34+35+34+34+38+42+35)/10 \\ &= 35.7 \end{aligned}$$

เมื่อ $I = 6, M = 10$

$$\begin{aligned} Y[6] &= (X[6]+X[7]+X[8]+X[9]+X[10]+X[11]+X[12]+X[13]+X[14]+X[15])/10 \\ &= (35+34+34+35+34+34+38+42+35+35)/10 \\ &= 35.6 \end{aligned}$$

เมื่อ $I = 7, M = 10$

$$\begin{aligned} Y[7] &= (X[7]+X[8]+X[9]+X[10]+X[11]+X[12]+X[13]+X[14]+X[15]+X[16])/10 \\ &= (34+34+35+34+34+38+42+35+35+34)/10 \\ &= 35.5 \end{aligned}$$

ເນື້ອ I = 8 , M = 10

$$\begin{aligned} Y[8] &= (X[8]+X[9]+X[10]+X[11]+X[12]+X[13]+X[14]+X[15]+X[16]+X[17])/10 \\ &= (34+35+34+34+38+42+35+35+34+35)/10 \\ &= 35.6 \end{aligned}$$

ເນື້ອ I = 9 , M = 10

$$\begin{aligned} Y[9] &= (X[9]+X[10]+X[11]+X[12]+X[13]+X[14]+X[15]+X[16]+X[17]+X[18])/10 \\ &= (35+34+34+38+42+35+35+34+35+35)/10 \\ &= 35.7 \end{aligned}$$

ເນື້ອ I = 10 , M = 10

$$\begin{aligned} Y[10] &= (X[10]+X[11]+X[12]+X[13]+X[14]+X[15]+X[16]+X[17]+X[18]+X[19])/10 \\ &= (34+34+38+42+35+35+34+35+35+36)/10 \\ &= 35.8 \end{aligned}$$

ເນື້ອ I = 11 , M = 10

$$\begin{aligned} Y[11] &= (X[11]+X[12]+X[13]+X[14]+X[15]+X[16]+X[17]+X[18]+X[19]+X[20])/10 \\ &= (34+38+42+35+35+34+35+35+36+35)/10 \\ &= 35.9 \end{aligned}$$

ເນື້ອ I = 12 , M = 10

$$\begin{aligned} Y[12] &= (X[12]+X[13]+X[14]+X[15]+X[16]+X[17]+X[18]+X[19]+X[20]+X[21])/10 \\ &= (38+42+35+35+34+35+35+36+35+32)/10 \\ &= 35.7 \end{aligned}$$

ເນື້ອ I = 13 , M = 10

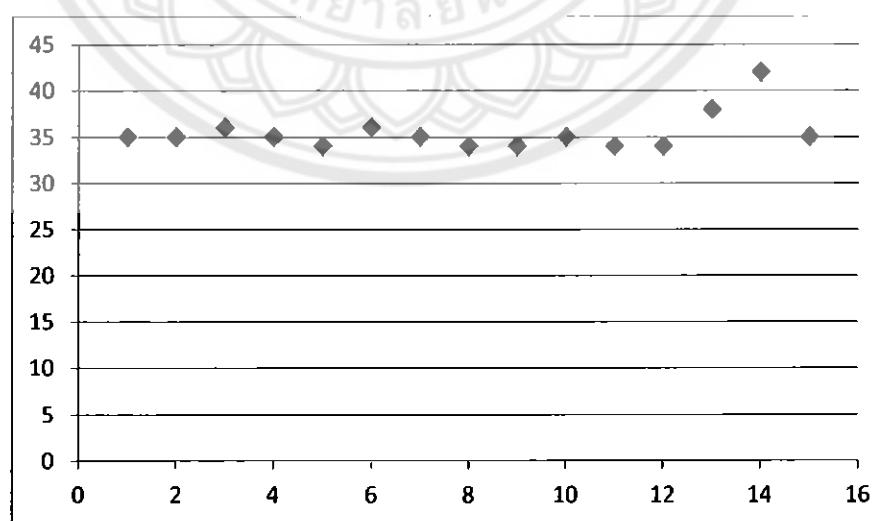
$$\begin{aligned} Y[13] &= (X[13]+X[14]+X[15]+X[16]+X[17]+X[18]+X[19]+X[20]+X[21]+X[22])/10 \\ &= (42+35+35+34+35+35+36+35+32+34)/10 \\ &= 35.3 \end{aligned}$$

ເນື້ອ I = 14 , M = 10

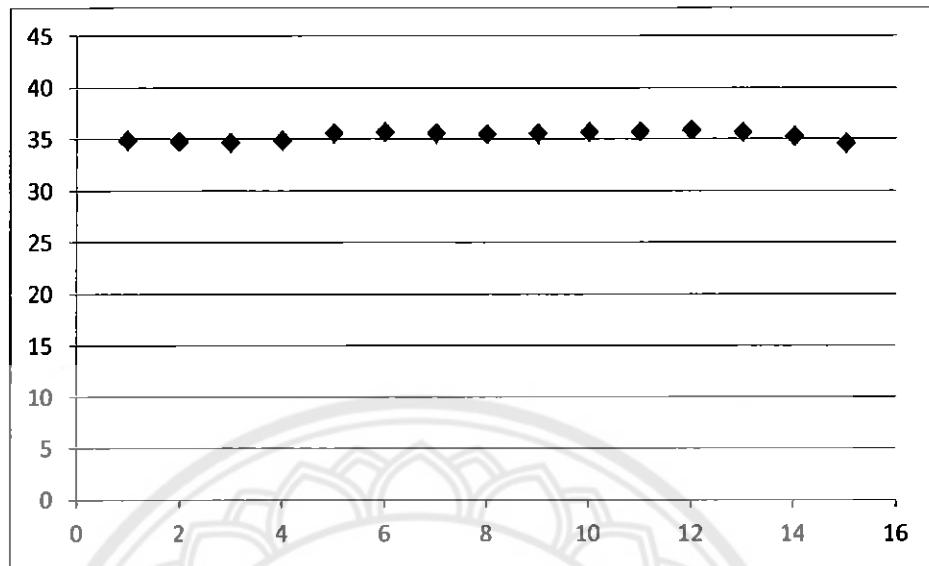
$$\begin{aligned} Y[14] &= (X[14]+X[15]+X[16]+X[17]+X[18]+X[19]+X[20]+X[21]+X[22]+X[23])/10 \\ &= (35+35+34+35+35+36+35+32+34+35)/10 \\ &= 34.6 \end{aligned}$$

ตารางที่ 4.7 เปรียบเทียบค่า RSSI เดิมกับค่า RSSI ใหม่ที่ใช้ Moving Average Filter

| ค่า RSSI เดิม | ค่า RSSI ใหม่ |
|---------------|---------------|
| 35 | 34.9 |
| 35 | 34.8 |
| 36 | 34.7 |
| 35 | 34.9 |
| 34 | 35.6 |
| 36 | 35.7 |
| 35 | 35.6 |
| 34 | 35.5 |
| 34 | 35.6 |
| 35 | 35.7 |
| 34 | 35.8 |
| 34 | 35.9 |
| 38 | 35.7 |
| 42 | 35.3 |
| 35 | 34.6 |



รูปที่ 4.21 รูปกราฟแสดงค่า RSSI ที่ไม่ได้ใช้อัลกอริทึม



รูปที่ 4.22 รูปกราฟที่ใช้ Moving Average Filter กรองค่า RSSI

เราจะสังเกตุได้ว่า เมื่อใช้ Moving Average Filter มากรองค่าความเข้มสัญญาณจะได้กราฟที่มีความต่อเนื่องไม่กระชับกระหาย ซึ่งต่างจากกราฟที่ไม่ใช้ Moving Average Filter มาช่วยในการกรอง และถ้าเราเพิ่มค่า M ให้มากขึ้นจะได้กราฟที่มีความต่อเนื่องมากขึ้น เนื่องจากมีการเฉลี่ยค่าความเข้มหลายค่ามากขึ้นนั่นเอง

4.4.3 การคำนวณตำแหน่งจากระยะทางที่ทำการแปลงแล้ว

เมื่อได้ระยะทางจากสมการที่เราสร้างขึ้นแล้ว เราจะใช้ทฤษฎีทางคณิตศาสตร์คือ กฎของคราเมอร์ มาทำการหาจุดตัดของสมการวงกลม 3 วง ก็จะได้พิกัดของวัตถุที่เราต้องการทราบ

จากสมการวงกลมหลัก 3 สมการดังนี้

$$\begin{aligned} (X_1 - X_u)^2 + (Y_1 - Y_u)^2 &= R_1^2 \\ (X_2 - X_u)^2 + (Y_2 - Y_u)^2 &= R_2^2 \\ (X_3 - X_u)^2 + (Y_3 - Y_u)^2 &= R_3^2 \end{aligned} \quad \text{สมการ (4.2)}$$

จัดให้อยู่ในรูปสมการเมทริกซ์

$$2 \begin{bmatrix} X_3 - X_1 & Y_3 - Y_1 \\ X_3 - X_2 & Y_3 - Y_2 \end{bmatrix} \begin{bmatrix} X_u \\ Y_u \end{bmatrix} = \begin{bmatrix} (R_1^2 - R_3^2) - (X_1^2 - X_3^2) - (Y_1^2 - Y_3^2) \\ (R_2^2 - R_3^2) - (X_2^2 - X_3^2) - (Y_2^2 - Y_3^2) \end{bmatrix} \quad \text{สมการ (4.3)}$$

สมมุติ เรากำหนดให้ Node 1 มีพิกัด (25.3,9.5) , Node 2 มีพิกัด (21,17) , Node 3 มีพิกัด (16.7,9.5) และ
 $R1 = 5\text{m}$, $R2 = 5\text{m}$, $R3 = 5\text{m}$

แทนค่าในสมการจะได้

$$2 \begin{bmatrix} 16.7 - 25.3 & 9.5 - 9.5 \\ 16.7 - 21 & 9.5 - 17 \end{bmatrix} \begin{bmatrix} X_u \\ Y_u \end{bmatrix} = \begin{bmatrix} (5^2 - 5^2) - (25.3^2 - 16.7^2) - (9.5^2 - 9.5^2) \\ (5^2 - 5^2) - (21^2 - 16.7^2) - (17^2 - 9.5^2) \end{bmatrix} \quad \text{สมการ (4.4)}$$

$$2 \begin{bmatrix} -8.6 & 0 \\ -4.3 & -7.5 \end{bmatrix} \begin{bmatrix} X_u \\ Y_u \end{bmatrix} = \begin{bmatrix} -361.2 \\ -360.86 \end{bmatrix} \quad \text{สมการ (4.5)}$$

$$\begin{bmatrix} -8.6 & 0 \\ -4.3 & -7.5 \end{bmatrix} \begin{bmatrix} X_u \\ Y_u \end{bmatrix} = \begin{bmatrix} -180.6 \\ -180.43 \end{bmatrix} \quad \text{สมการ (4.6)}$$

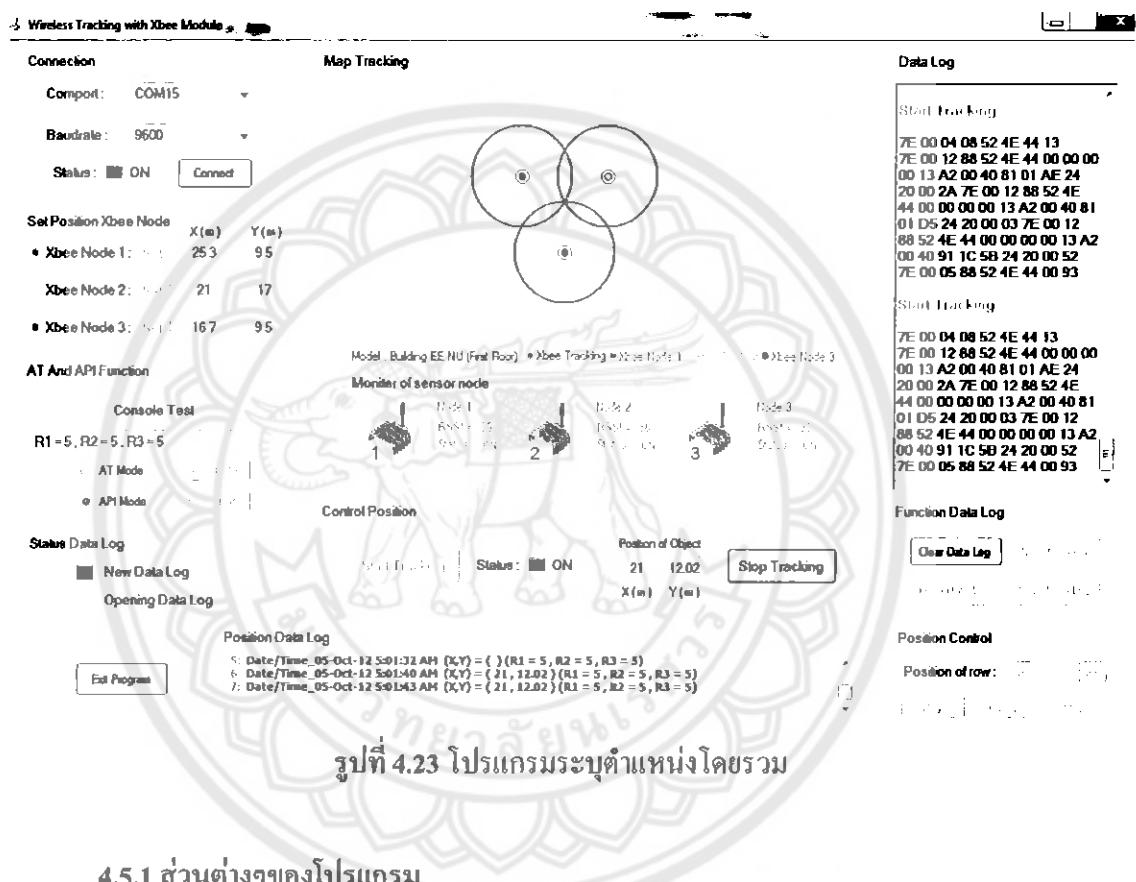
จะได้

$$X_u = \frac{\begin{bmatrix} -180.6 & 0 \\ 180.43 & -7.5 \end{bmatrix}}{\begin{bmatrix} -8.6 & 0 \\ -4.3 & -7.5 \end{bmatrix}} = \frac{1354.5}{64.5} = 21 \quad \text{สมการ (4.7)}$$

$$Y_u = \frac{\begin{bmatrix} -8.6 & -180.6 \\ -4.3 & -180.43 \end{bmatrix}}{\begin{bmatrix} -8.6 & 0 \\ -4.3 & -7.5 \end{bmatrix}} = \frac{776.58}{64.5} = 12.04 \quad \text{สมการ (4.8)}$$

4.5 โปรแกรมระบุตำแหน่ง

หลังจากที่เราได้ทำการศึกษาค้นคว้าหาทฤษฎีและหลักการต่างๆมาใช้ในการคำนวณหาตำแหน่งแล้ว ขั้นต่อไปคือการเขียนโปรแกรมรองรับการทำงานตามแผนที่ได้กำหนดไว้ โดยหน้าตาโปรแกรมเป็นดังนี้

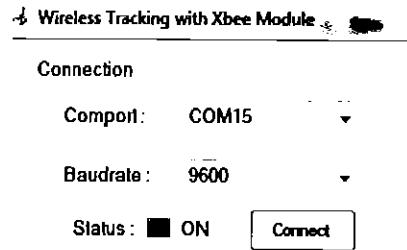


รูปที่ 4.23 โปรแกรมระบุตำแหน่งโดยรวม

4.5.1 ส่วนต่างๆของโปรแกรม

โปรแกรมนี้จะแบ่งองค์ประกอบต่างๆออกเป็น 11 ส่วนคือ

1. Connection
2. Set position Xbee node
3. AT and API function
4. Status datalog
5. Map tracking
6. Moniter of sensor node
7. Control position



รูปที่ 4.25 แสดงส่วนประกอบของ connection

4.5.1.2 Set position Xbee node

ในส่วนของ Set position Xbee node จะเป็นการ set ค่าตำแหน่งที่อยู่ของ node จ้างอิง เพื่อใช้ในการคำนวณหาระยะทางกับ node ที่เคลื่อนที่ โดย node จ้างอิงจะมี 3node คือ node 1 , node 2 , node 3 ตามลำดับ และมีสีนักกว่า node ไหนสีอะไรเพื่อง่ายต่อการคูณแผนที่ โดย node 1 สีชมพู , node 2 สีเขียวเข้ม , node 3 สีฟ้า ส่วนในการกำหนดพิกัด X,Y จะใช้หน่วยเมตร โดยสามารถกำหนดทศนิยมได้ 2 ตำแหน่ง

| Set Position Xbee Node | X (m) | Y (m) |
|------------------------|-------|-------|
| • Xbee Node 1 : [Set] | 25.3 | 9.5 |
| Xbee Node 2 : [Set] | 21 | 17 |
| • Xbee Node 3 : [Set] | 16.7 | 9.5 |

รูปที่ 4.26 แสดงส่วนประกอบของ Set position Xbee node

4.5.1.3 AT and API function

ในส่วนของ AT and API function ทำหน้าที่ส่งคำสั่งไปอ่านหรือเขียนค่าของ Xbee ได้โดยสามารถเลือกใหม่ค่าสั่งได้ 2 แบบ คือ AT กับ API โดยต่างกันที่แบบ AT จะเป็นการส่งคำสั่งเฉพาะที่ Xbee สามารถเข้าใจได้เป็นตัวอักษรแต่ก่อนที่จะส่งข้อความต้องส่ง +++ ไปก่อนเพื่อให้ Xbee ตัวที่ติดต่อรู้ว่าจะส่งข้อความแล้ว เมื่อ Xbee รู้ว่าจะส่งจะมีข้อความ OK ตอบกลับมา ก็สามารถส่งข้อความได้ เช่น +++ (รอ Xbee ตอบ) OK ->ATND จะเป็นการส่งคำสั่งแบบบอร์ดแครสไปยัง Xbee ที่อยู่ในรัศมีให้ส่งค่า Address กับค่าความเร็วสัญญาณกลับมาขึ้นตัวที่ร้องขอ ส่วนโหมด API นั้นไม่ต้องส่ง +++ ก็สามารถรับส่งข้อมูลได้เลย โดยจะต้องส่งแบบเป็น packet ตามโครงสร้างของคำสั่ง Xbee โดย packet ที่ส่งไปนั้นจะต้องเป็นเลขฐาน 16 ก่อนถึงจะสามารถส่งได้ เช่น ถ้าต้องการส่ง ATND

จะต้องส่ง packet ว่า 7E 00 04 08 52 4E 44 13 ถึงจะสามารถส่งค่าได้ ในเรื่องของเวลาการรับส่งนั้น โหนด API จะมีความเร็วกว่าแบบ AT ประมาณ 2 เท่า

AT And API Function

Console Test

- AT Mode [View Datalog]
- API Mode [View Log]

รูปที่ 4.27 แสดงส่วนประกอบของ AT and API function

4.5.1.4 Status datalog

ในส่วนของ Status datalog จะเป็นส่วนบอกสถานะว่าอยู่ในสถานะเปิดคู Datalog เก่าหรือว่า เริ่มระบบบันทึกตำแหน่งใหม่ โดยจะมีไฟสีเขียวบอกว่ากำลังอยู่ที่สถานะใหม่

Status Data Log

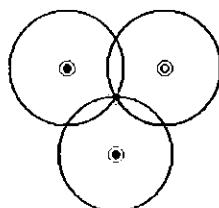
- New Data Log
- Opening Data Log

รูปที่ 4.28 แสดงส่วนประกอบของ Status datalog

4.5.1.5 Map tracking

ในส่วนของ Map tracking จะเป็นหน้าจอแสดงตำแหน่งของ Xbee Reference node กับ Xbee Tracking Node ว่าตอนนี้อยู่ตรงส่วนไหนของแผนที่ โดยจะมีลีก์กำกับอยู่ว่าสีไหนเป็น node อะไร และ จะมีรัศมีความกว้างระหว่าง node Reference กับ node tracking กำกับด้วย

Map Tracking



Model : Building EE 110 (First Floor) ▶ Xbee Tracking ■ Xbee Node 1 □ Xbee Node 2 ▢ Xbee Node 3

รูปที่ 4.29 แสดงส่วนประกอบของ Map tracking

4.5.1.6 Moniter of sensor node

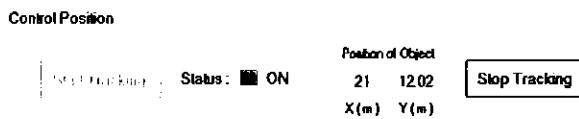
ในส่วนของ Moniter of sensor node จะเป็นการบอกสถานะของ Reference Node ว่า ON หรือ OFF และมีค่า RSSI แสดงกำกับเมื่อทำการเริ่มการ Tracking



รูปที่ 4.30 แสดงส่วนประกอบของ Moniter of sensor node

4.5.1.7 Control position

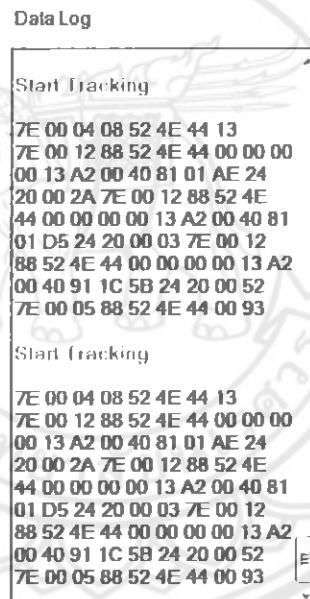
ในส่วนของ Control position จะเป็นส่วนของการควบคุมการ StartTracking หรือ Stop Tracking โดยจะมีสถานะบอกว่า Start หรือ Stop อยู่ด้วย และเมื่อทำการ Start Tracking แล้วจะมีค่า พิกัด X,Y ของวัตถุที่เคลื่อนที่แสดงขึ้นมาด้วย โดยโปรแกรมจะทำการคำนวณพิกัดทุกๆ 1 วินาทีแล้ว แสดงผลออกทางหน้าจอที่แสดงขึ้นมาด้วย โดยจะทำการส่งคำสั่ง ATND โดยทำการแปลงเป็น Hex string ก่อน(7E 00 04 08 52 4E 44 13) ส่งเป็น API packet ไปเพื่อให้ Xbee Reference Node ส่งค่า RSSI กลับมาแล้วแปลง เป็นระบบทางจากโอนดที่เคลื่อนที่กับโอนดอ้างอิง 3 โอนด แล้วใช้กู้ของรามอร์น่าแก้ระบบสมการ วงกลม 3 วง ก็จะได้ตำแหน่งวัตถุที่เราต้องการทราบตำแหน่ง แล้วทำการเก็บเป็น Data Log ไว้ สามารถ บันทึกค่าเพื่อใช้ดูย้อนหลังได้



รูปที่ 4.31 แสดงส่วนประกอบของ Control position

4.5.1.8 Datalog

ในส่วนของ Datalog จะเป็นการแสดงค่าของ การรับส่งข้อมูลระหว่าง Tracking Node กับ Reference Node ว่ารับส่งข้อมูลอะไรบ้าง โดยจะแสดงเป็นแบบยาวลงมา สามารถเลือกข้อมูลย้อนหลังได้

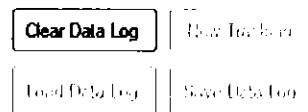


รูปที่ 4.32 แสดงส่วนประกอบของ Datalog

4.5.1.9 Function datalog

ในส่วนของ Function datalog จะเป็นส่วนจัดการเกี่ยวกับ Data Log ซึ่งสามารถ Clear Data Log ได้ถ้าข้อมูลของเกินไป สามารถ New Tracking ใหม่ได้ คือ เริ่มการระบุตำแหน่งใหม่ตั้งแต่แรก สามารถ Save/Load Data Log เก็บมาดูย้อนหลังได้

Function Data Log

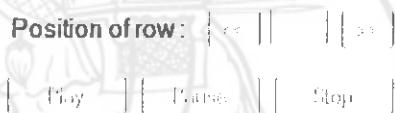


รูปที่ 4.33 แสดงส่วนประกอบของ Function datalog

4.5.1.10 Position control

ในส่วนของ Position control จะเป็นส่วนของการควบคุมการแสดงตำแหน่งของวัตถุที่ทำการ Save DataLog ไว้มาดูข้อมูลัง โดยสามารถแสดงตำแหน่งได้ต่อเนื่องหรือจะเลือกตำแหน่งที่ต้องการคุ้นจากปุ่มเลือกบรรทัด ได้

Position Control



รูปที่ 4.34 แสดงส่วนประกอบของ Position control

4.5.1.11 Position datalog

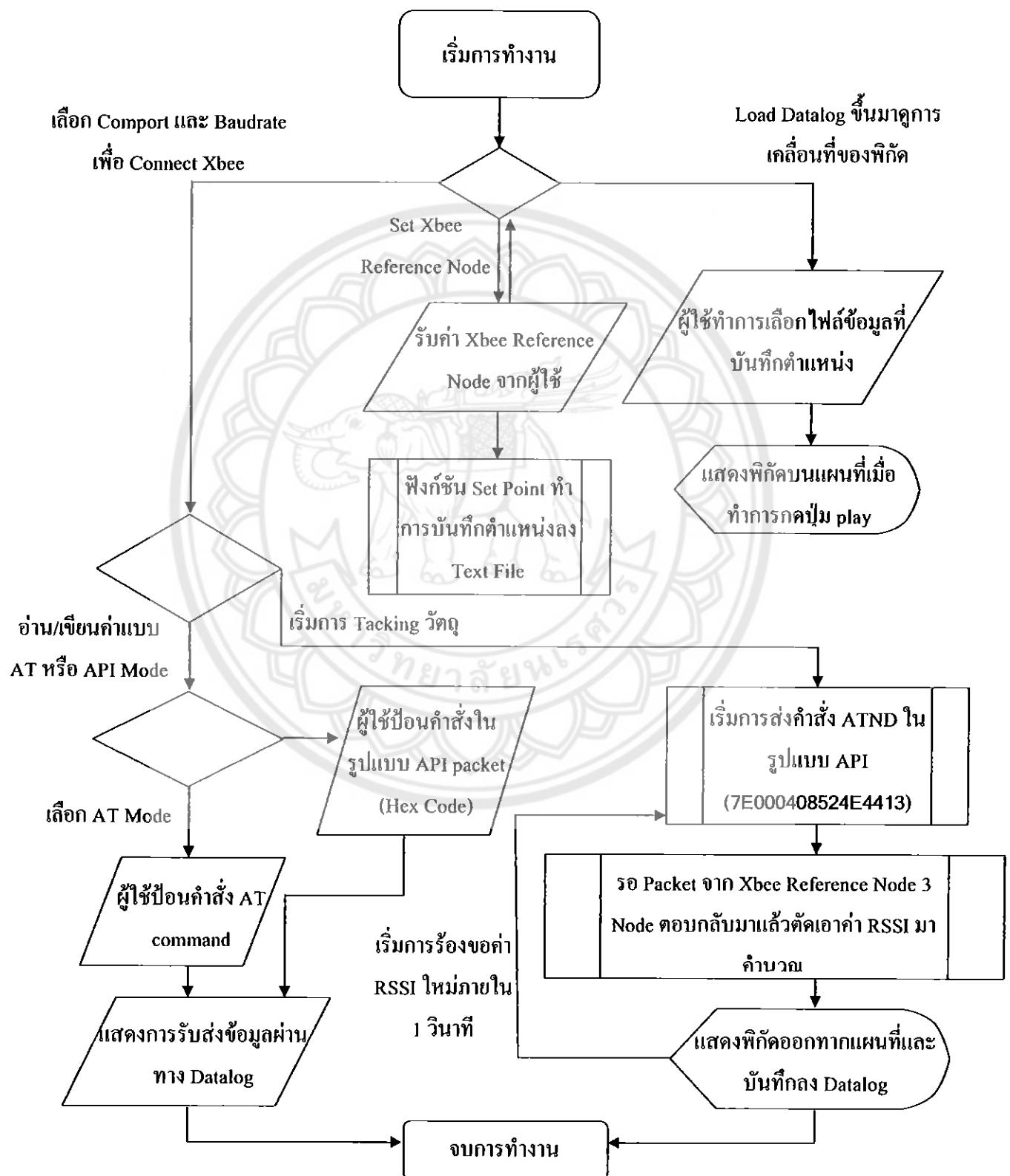
ในส่วนของ Position datalog จะแสดงจำนวนพิกัดทั้งหมด โดยจะมีหมายเลขกำกับ แสดงวันเวลาที่บันทึกพิกัด แสดงพิกัดของวัตถุที่เคลื่อนที่ และค่ารัศมีระหว่าง Reference Node 3 จุดกับ Tracking Node ตามลำดับ

Position Data Log

```
5: Date/Time_05-Oct-12 5:01:32 AM (X,Y) = ( ) (R1 = 5, R2 = 5, R3 = 5)
6: Date/Time_05-Oct-12 5:01:40 AM (X,Y) = ( 21, 12.02 ) (R1 = 5, R2 = 5, R3 = 5)
7: Date/Time_05-Oct-12 5:01:43 AM (X,Y) = ( 21, 12.02 ) (R1 = 5, R2 = 5, R3 = 5)
```

รูปที่ 4.35 แสดงส่วนประกอบของ Position datalog

4.6 แผนภาพการทำงานของโปรแกรม



รูปที่ 4.36 แผนภาพการทำงานของโปรแกรม

4.7 หน้าที่การทำงานของฟังก์ชันต่างๆในโปรแกรม

โปรแกรมระบุตำแหน่งนี้มี Class 1 Class ชื่อ Form1 โดยมี Method ต่างๆ ดังนี้

1. Log_Messages_Type ชนิด Enum มีค่า 5 ค่า คือ Incoming , Outgoing , Normal , Warning , Error ทำหน้าที่บอกว่าข้อความที่รับส่งเป็นชนิดใด โดยแสดงที่ Datalog

2. Log_Messages_TypeColor ชนิด Color มี 5 สี คือ Color.Blue , Color.Green , Color.Black , Color.Orange , Color.Red ทำหน้าที่แยกประเภทของข้อมูลที่รับส่งให้ชัดเจนง่ายต่อการวิเคราะห์

3. Position_data_log ชนิด Enum มีค่า 3 ค่า คือ row , date , position ทำหน้าที่บอกว่าข้อความที่รับส่งเป็นชนิดใด โดยแสดงที่ Position Datalog

4. Position_data_log_color ชนิด Color มี 3 สี คือ Color.Orange , Color.Blue , Color.Red ทำหน้าที่แยกประเภทของข้อมูลที่รับส่งให้ชัดเจนง่ายต่อการวิเคราะห์

5. Form1 จะทำงานเมื่อรันโปรแกรมขึ้นมาครั้งแรก ทำหน้าที่เรียกใช้งาน Method พื้นฐาน 3 Method คือ InitializeComponent() , InitializeSerialComponent() , InitializeGeneralComponent()

6. InitializeComponent ทำหน้าที่เรียก component ของระบบที่ต้องใช้เมื่อรันโปรแกรม

7. InitializeSerialComponent ทำหน้าที่ดึงรายชื่อ Port ต่างๆที่ต่ออยู่กับคอมพิวเตอร์

8. InitializeGeneralComponent ทำหน้าที่ Disable/Enable ปุ่มต่างๆ เมื่อเริ่มโปรแกรมและเรียกฟังก์ชัน Set การทำงานของ Reference Node

9. exit_program_Click ทำหน้าที่ปิดโปรแกรมเมื่อปุ่ม Exit ถูกกด

10. clear_data_log_Click ทำหน้าที่เคลียร์ค่าใน Datalog RichTextbox เมื่อปุ่ม Clear ถูกกด

11. clear_text_Click ทำหน้าที่เคลียร์ค่าใน Console data เมื่อปุ่ม Clear Text ถูกกด

12. send_data_Click ทำหน้าที่ส่ง Data ที่ผู้ใช้กรอกในช่อง Console Data ไปที่ Xbee

13. CommandTxt_KeyDown กับ CommandTxt_KeyPress ทำหน้าที่ร่วมกันเมื่อผู้ใช้กดปุ่ม Enter จะส่งข้อมูลที่อยู่ใน Console Data ไปที่ Xbee แทนการกดปุ่ม Send Data และเมื่อผู้ใช้กดปุ่ม Delete จะทำการ Delete ข้อมูลใน Console Data แทนการกดปุ่ม Clear Text เพื่อสะดวกต่อผู้ใช้

14. setNode_3_Node ทำหน้าที่อ่านพิกัด Xbee Reference Node 3 Node จาก Text File ที่ Save อยู่ใน Bin ของโปรแกรมมาใส่ไว้ในตัวแปรค่าพิกัด

15. connectXbee จะทำหน้าที่ Connect Xbee เมื่อเลือก Comport และ Baudrate ถูกต้อง

16. connect_Click จะทำหน้าที่เรียก Method Connect Xbee เมื่อปุ่ม Connect ถูกกด

17. New_track ทำหน้าที่เริ่มการ Tracking ใหม่โดยจะทำการเคลียร์ค่าและ Set ปุ่มต่างๆ เมื่อปุ่ม New Track ถูกกด

18. begin ทำหน้าที่บอกผู้ใช้ว่าควรทำอย่างไรเมื่อเริ่มโปรแกรม โดยจะมีข้อความบอกใน Datalog ว่าต้องทำอย่างไรบ้าง
19. Log ทำหน้าที่แปลงข้อความที่รับส่ง Map กับ Enum ว่าเป็นชนิดใดສีดีและทำให้เป็นตัวพิมใหญ่และเลือกแบบอักษรที่กำหนดแล้วแสดงออกที่ Datalog RichTextbox
20. Position_Log ทำหน้าที่แปลงข้อความที่ส่งกับ enum ของ position ว่าเป็นชนิดใดและสีอะไรแล้วแสดงออก Position Datalog RichTextbox
21. ByteArrayToString ทำหน้าที่แปลงข้อมูลจาก ByteArray เป็น Hex String
- 22.HexStringToByteArray ทำหน้าที่แปลงข้อมูลจาก Hex String เป็น ByteArray
23. SendData ทำหน้าที่ส่งข้อมูลจาก Console Data ไปยัง Xbee โดยคุณผู้ใช้เลือกโหมดได้
24. receiveDataXbee ทำหน้าที่เช็คข้อมูลที่ส่งมาจาก Xbee Reference Node ว่าส่งมาครบหรือไม่ถ้าไม่ครบก็จะทำการส่งใหม่อีก 6 ครั้ง ถ้าข้อมูลไม่ครบก็จะแสดงข้อความเตือนผู้ใช้
25. DataReceived ทำหน้าที่รับข้อมูลจาก Comport โดยถ้าเลือก AT Mode ก็จะนำค่าที่รับมาแสดงใน Datalog RichTextbox แต่ถ้าเป็น API Mode จะนำไปประมวลผลตามที่กำหนดต่อไป
26. checkData_ofXbee ทำหน้าที่รับข้อมูลจาก receiveDataXbee มาตัดเอาเฉพาะค่า RSSI แล้วทำการคำนวณแปลงเป็นรัศมีและแสดงค่า RSSI และรัศมีผ่านทางโปรแกรม
27. calculate_point ทำหน้าที่รับค่ารัศมีและพิกัดของ Reference Node มาคำนวณหาพิกัดโดยใช้程式码หานการจุดตัดของวงกลม 3 วงตัดกันแล้วแสดงผลออกมาโปรแกรมและบันทึกพิกัดลง Position Datalog RichTextbox
28. set_point1_Click ทำหน้าที่ Set ค่าพิกัด Reference Node 1 ลง Text File โดยรับค่าจาก point_x1 และ point_y1 ทาง Text Box เมื่อผู้ใช้กดปุ่ม Set
29. set_point2_Click ทำหน้าที่ Set ค่าพิกัด Reference Node 2 ลง Text File โดยรับค่าจาก point_x2 และ point_y2 ทาง Text Box เมื่อผู้ใช้กดปุ่ม Set
30. set_point3_Click ทำหน้าที่ Set ค่าพิกัด Reference Node 3 ลง Text File โดยรับค่าจาก point_x3 และ point_y3 ทาง Text Box เมื่อผู้ใช้กดปุ่ม Set
31. OnPaint ทำหน้าที่ Paint ตำแหน่งของพิกัด Xbee Reference Node และ Xbee Tracking ออกทางแผนที่
32. draw_wave_point1 ทำหน้าที่แสดงค่ารัศมี pointx1 และ pointy1 แสดงบนแผนที่
33. draw_wave_point2 ทำหน้าที่แสดงค่ารัศมี pointx2 และ pointy2 แสดงบนแผนที่
34. draw_wave_point3 ทำหน้าที่แสดงค่ารัศมี pointx3 และ pointy3 แสดงบนแผนที่

35. draw_wave_point1 ทำหน้าที่นำค่าพิกัด pointx1 และ pointy1 แสดงบนแผนที่
36. draw_wave_point2 ทำหน้าที่นำค่าพิกัด pointx2 และ pointy2 แสดงบนแผนที่
37. draw_wave_point3 ทำหน้าที่นำค่าพิกัด pointx3 และ pointy3 แสดงบนแผนที่
38. draw_point_XY ทำหน้าที่แสดงค่าพิกัด Xbee Tracking จากการคำนวณบนแผนที่
39. begin_tracking ทำหน้าที่เริ่มทำการ Tracking โดยการส่งคำสั่ง ATND แบบ API Packet ไปยัง Xbee Reference Node โดยทำการส่งแบบต่อเนื่อง โดยรอบ 1 วินาที ก่อนส่งใหม่มีผู้ใช้กดปุ่ม Start Tracking
40. stop_track ทำหน้าที่หยุดการส่งคำสั่ง ATND และหยุดการทำงานของโปรแกรมเมื่อผู้ใช้กดปุ่ม Stop Tracking
41. save_data_log_Click ทำหน้าที่ Save Position Datalog RichTextbox ลง Text File เมื่อผู้ใช้กดปุ่ม Save Datalog
42. load_data_log_Click ทำหน้าที่ Load Datalog ข้อมูลการเคลื่อนที่ของตำแหน่งที่ทำการบันทึกเก็บไว้ใน Text File เมื่อผู้ใช้กดปุ่ม Load Datalog
43. play_tack ทำหน้าที่แสดงตำแหน่งของย่างต่อเนื่อง เมื่อผู้ใช้กดปุ่ม Play โดยจะแสดงตำแหน่งครั้งละ 1 วินาที
44. stop_Click ทำหน้าที่หยุดการแสดงตำแหน่ง เมื่อผู้ใช้กดปุ่ม Stop
45. pause_Click ทำหน้าที่หยุดพักการแสดงตำแหน่งชั่วคราว เมื่อผู้ใช้กดปุ่ม Pause
46. back_Click ทำหน้าที่เลื่อนตำแหน่งข้อนกลับไปครั้งละ 1 ตำแหน่ง เพื่อย้อนคืนข้อมูลตามต้องการได้ เมื่อผู้ใช้กดปุ่ม Back
47. next_Click ทำหน้าที่เลื่อนตำแหน่งไปข้างหน้าครั้งละ 1 ตำแหน่ง เพื่อเลื่อนคืนข้อมูลถัดไปตามต้องการได้ เมื่อผู้ใช้กดปุ่ม Next

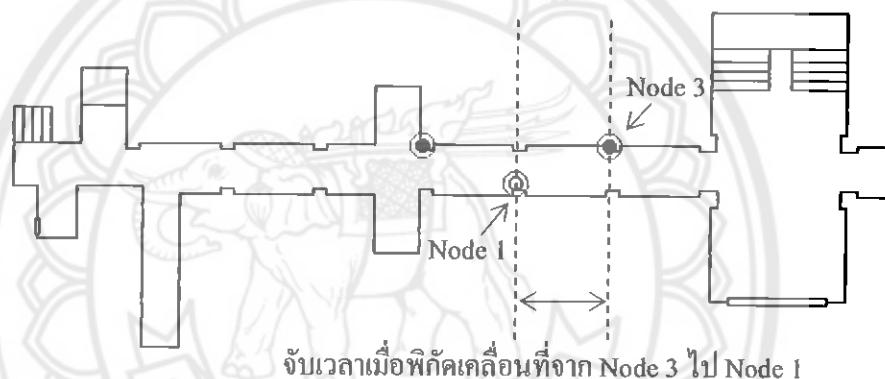
บทที่ 5

ผลการทดสอบ

5.1 วัดระยะเวลาที่พิกัดเคลื่อนที่จาก Node3 ไป Node1 เพื่อหาจำนวน Block ที่เหมาะสมของ Moving Average Filter

ทำการจับเวลาที่พิกัดเคลื่อนที่จาก Node 3 ไป Node 1 โดยใช้ขนาดของ Block ของ Moving Average Filter ท่ากับ 5, 10, 15, 20, 25, 30, 35, 40 เพื่อที่จะกำหนด block อย่างเหมาะสมจากสมการ

$$Y[i] = \frac{1}{M} \sum_{j=0}^{M-1} X[i+j]$$

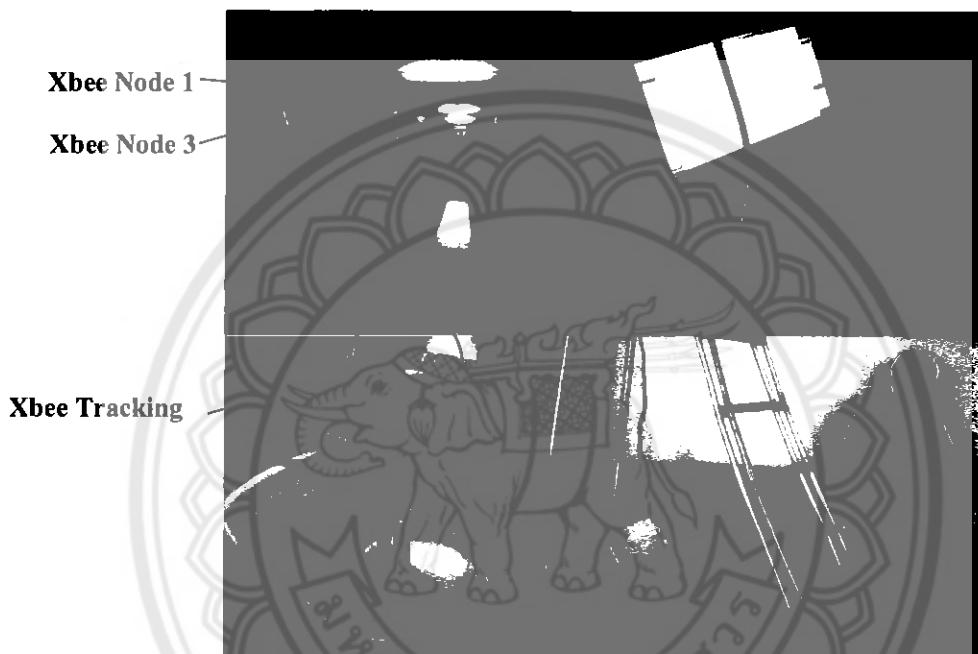


รูปที่ 5.1 ภาพแสดงการวัดค่าจาก Node 3 ไปยัง Node 1 เพื่อทำการหาค่า M ที่เหมาะสม

ตารางที่ 5.1 แสดงเวลาที่ใช้กับเบอร์เข็นความคลาดเคลื่อน

| จำนวน Block ของข้อมูล | เวลาที่ใช้(วินาที) | เบอร์เข็นความคลาดเคลื่อนของพิกัด |
|-----------------------|--------------------|----------------------------------|
| M = 5 | 3 | 45.3 % |
| M = 10 | 4 | 41.7 % |
| M = 15 | 5 | 36.2 % |
| M = 20 | 7 | 21.8 % |
| M = 25 | 9 | 10.3 % |
| M = 30 | 12 | 8.1 % |
| M = 35 | 15 | 5.4 % |
| M = 40 | 18 | 3.2 % |

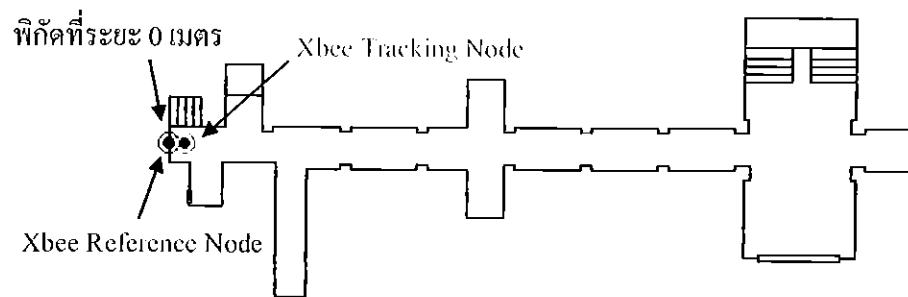
จากการจะเห็นได้ว่า ถ้าใช้จำนวนข้อมูลเฉลี่ยเพิ่มขึ้นจะทำให้เปอร์เซ็นต์ความคลาดเคลื่อนน้อยลง แต่ก็ต้องแลกกับเวลาที่เสียไปกว่าค่าหนึ่งจะเคลื่อนที่มาถึงจุดที่อยู่จริง ซึ่งถ้าต้องการความไวและความแม่นยำที่สามารถรับได้ จะต้องใช้ $M = 25$ ถึงจะเหมาะสม



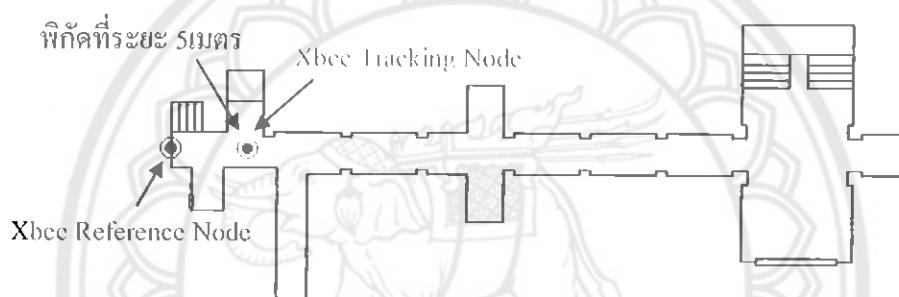
รูปที่ 5.2 ภาพแสดงการวัดค่าในพื้นที่จริงเพื่อทำการหาค่า M ที่เหมาะสม

5.2 วัดระยะทางภายในอาคารโดยใช้ Moving Average Filter

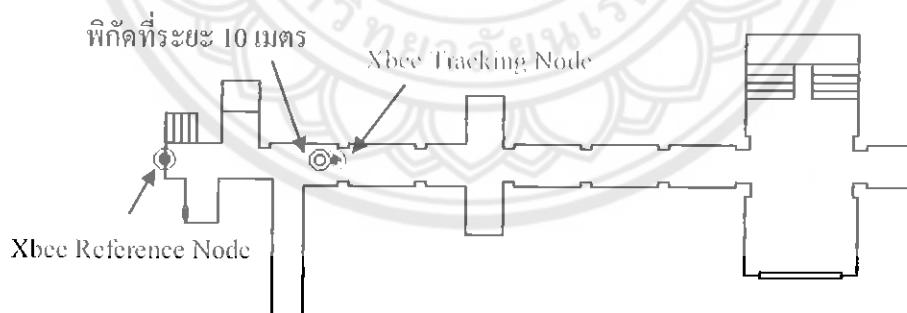
เพื่อนำมาเปรียบเทียบระหว่างค่าที่วัดได้จากโปรแกรมและค่าจริง โดยแสดงเป็นกราฟระหว่างจำนวนครั้งที่ทำการวัด (ทำการวัดค่า 100 ครั้ง) กับระยะทางจริงจะได้ผลการทดลองดังรูปที่ 5.3



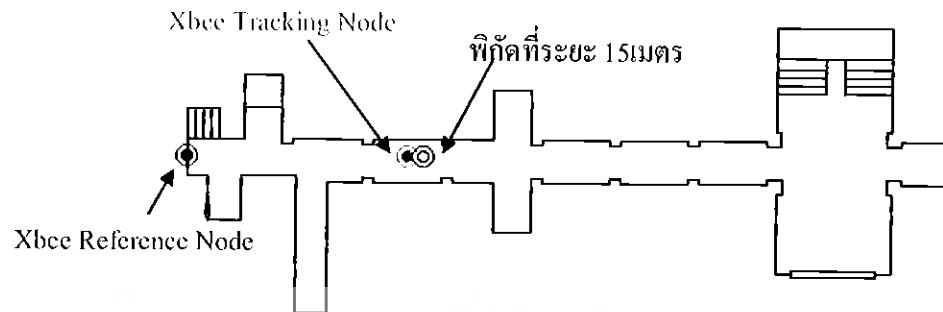
รูปที่ 5.3 ที่ระยะ 0 เมตร ระยะเกลี่ยที่คำนวณได้จากโปรแกรมคือ 1.000417 เมตร



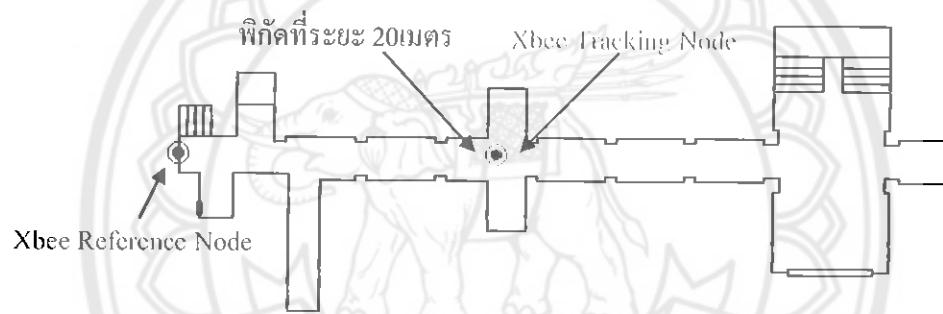
รูปที่ 5.4 ที่ระยะ 5 เมตร ระยะเกลี่ยที่คำนวณได้จากโปรแกรมคือ 4.894357 เมตร



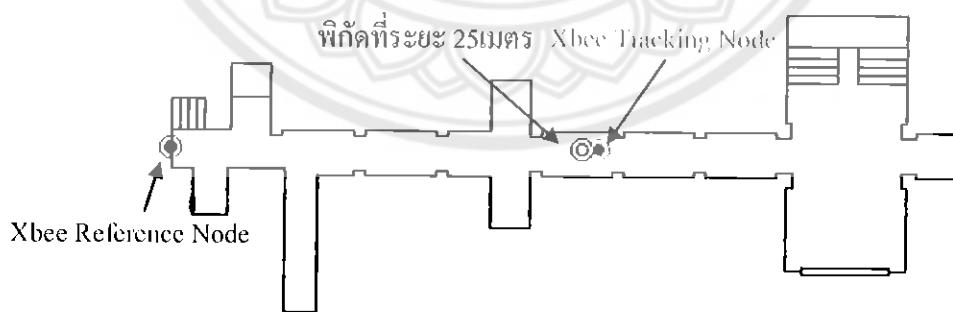
รูปที่ 5.5 ที่ระยะ 10 เมตร ระยะเกลี่ยที่คำนวณได้จากโปรแกรมคือ 10.89215 เมตร



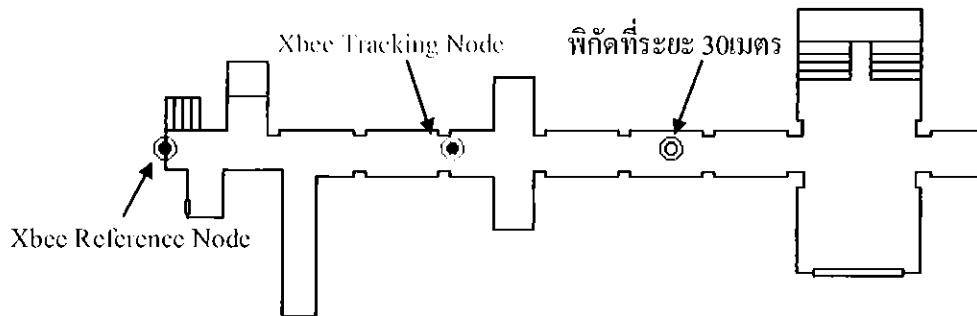
รูปที่ 5.6 ที่ระยะ 15 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 14.08812 เมตร



รูปที่ 5.7 ที่ระยะ 20 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 19.62941 เมตร



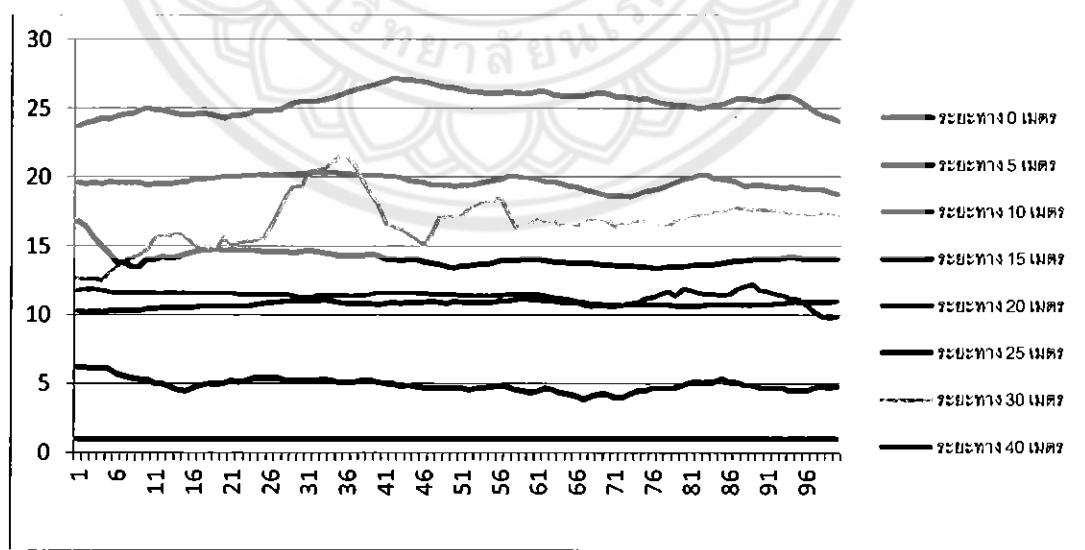
รูปที่ 5.8 ที่ระยะ 25 เมตร ระยะเฉลี่ยที่คำนวณได้จากโปรแกรมคือ 25.52645 เมตร



รูปที่ 5.9 ที่ระยะ 30 เมตร ระยะเหลี่ยมที่คำนวณได้จากโปรแกรมคือ 16.77712 เมตร



รูปที่ 5.10 ที่ระยะ 40 เมตร ระยะเหลี่ยมที่คำนวณได้จากโปรแกรมคือ 11.23613 เมตร



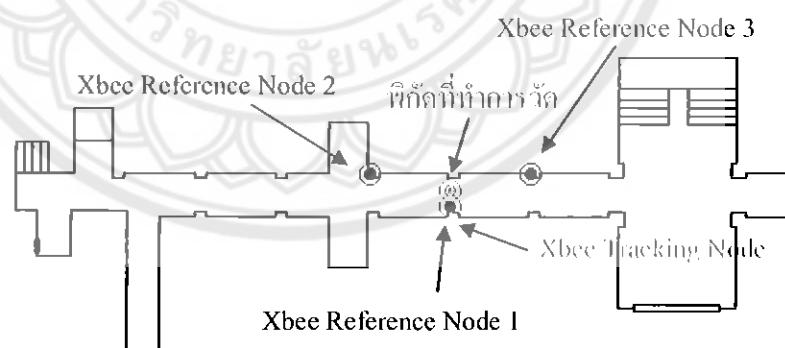
รูปที่ 5.11 กราฟแสดงจำนวนครั้งที่ทำการวัดกับระยะทางจริง



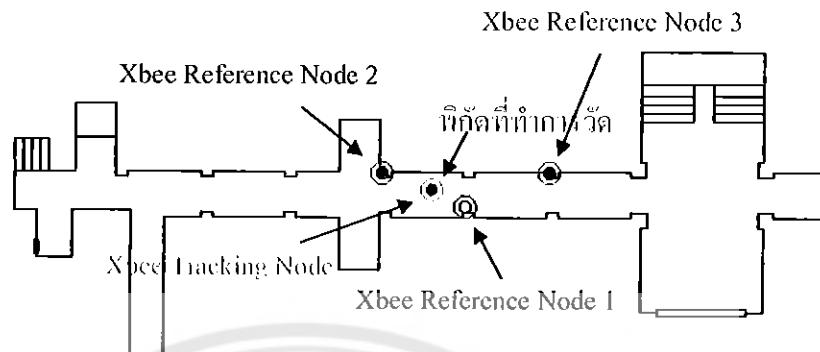
รูปที่ 5.12 ภาพการทดลองการวัดระยะทาง

5.3 วัดระยะทางภายในอาคารโดยใช้จุดสังเกต 3 จุด

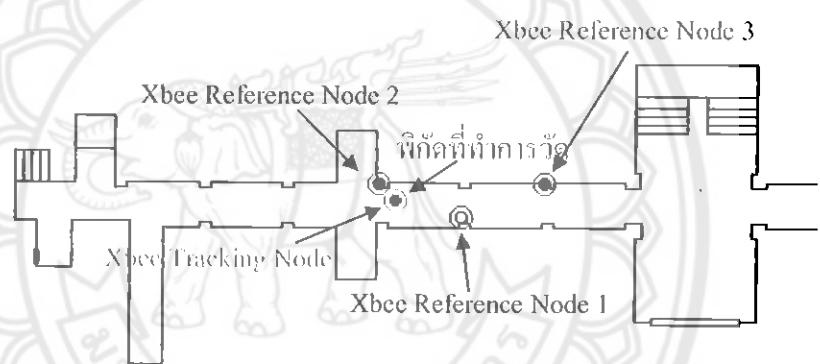
โดยทำการวัดค่าจาก Xbee Reference Node 3 Node ซึ่งถูกติดตั้งบนเพดานของชั้นที่ 1 ศึก EE เพื่อให้การส่งสัญญาณ RSSI มีสิ่งรบกวนน้อยที่สุด แล้วนำค่า RSSI มาแปลงเป็นระยะทางแล้วทำการหาพิกัดจากการคำนวณ แล้วทำการ Plot กราฟเพื่อนำมาเปรียบเทียบระหว่างค่าที่วัด ได้จากโปรแกรม และค่าจริง



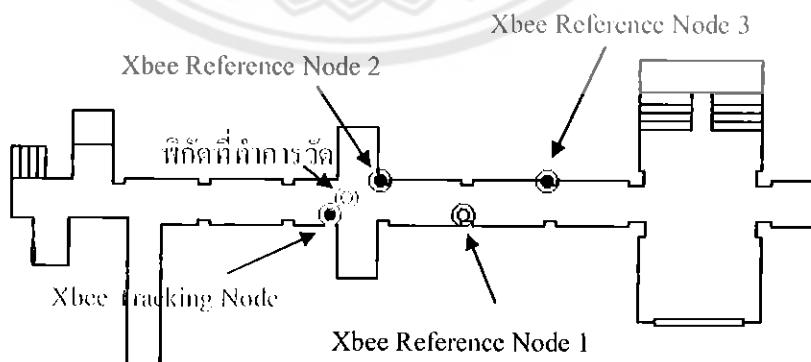
รูปที่ 5.13 วัดระยะที่พิกัด (27,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (26.93,12.07)



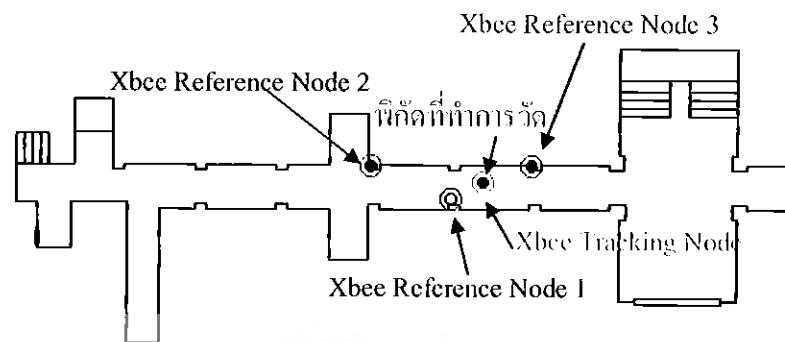
รูปที่ 5.14 วัสดุที่พิกัด (25,11) พิกัดเคลื่อนที่คำนวณได้จากโปรแกรมคือ (25.43,10.98)



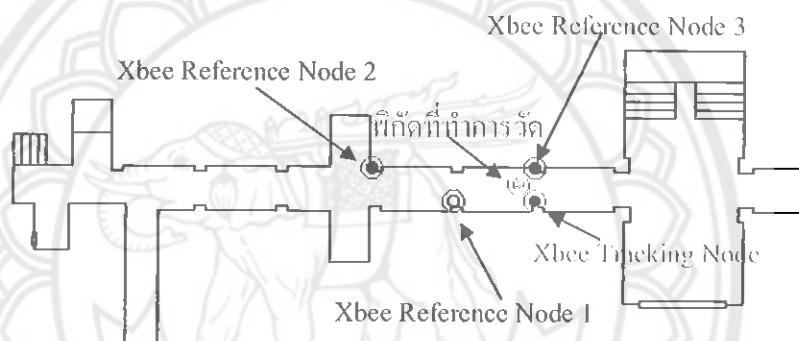
รูปที่ 5.15 วัสดุที่พิกัด (23,11) พิกัดเคลื่อนที่คำนวณได้จากโปรแกรมคือ (23.13,11.18)



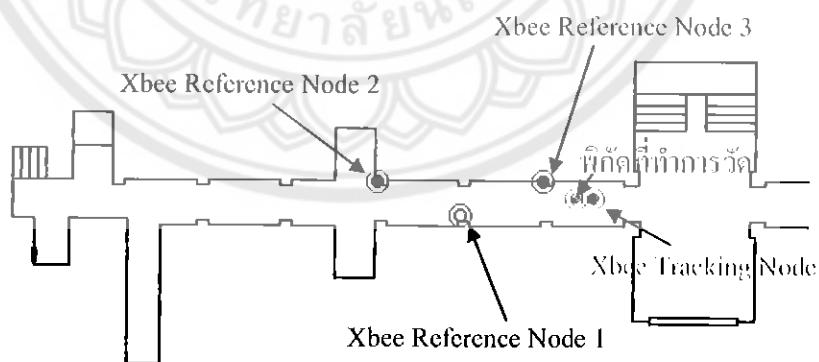
รูปที่ 5.16 วัสดุที่พิกัด (20,11) พิกัดเคลื่อนที่คำนวณได้จากโปรแกรมคือ (18.72,12.09)



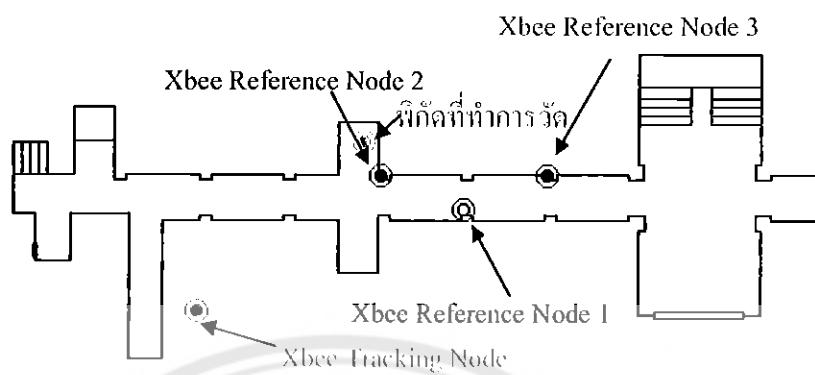
รูปที่ 5.17 วัสดุระบบที่พิกัด (29,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (29.22,10.77)



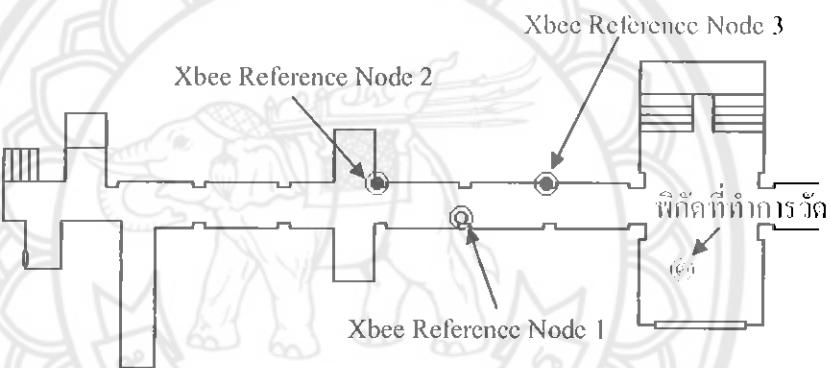
รูปที่ 5.18 วัสดุระบบที่พิกัด (31,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (31.87,12.41)



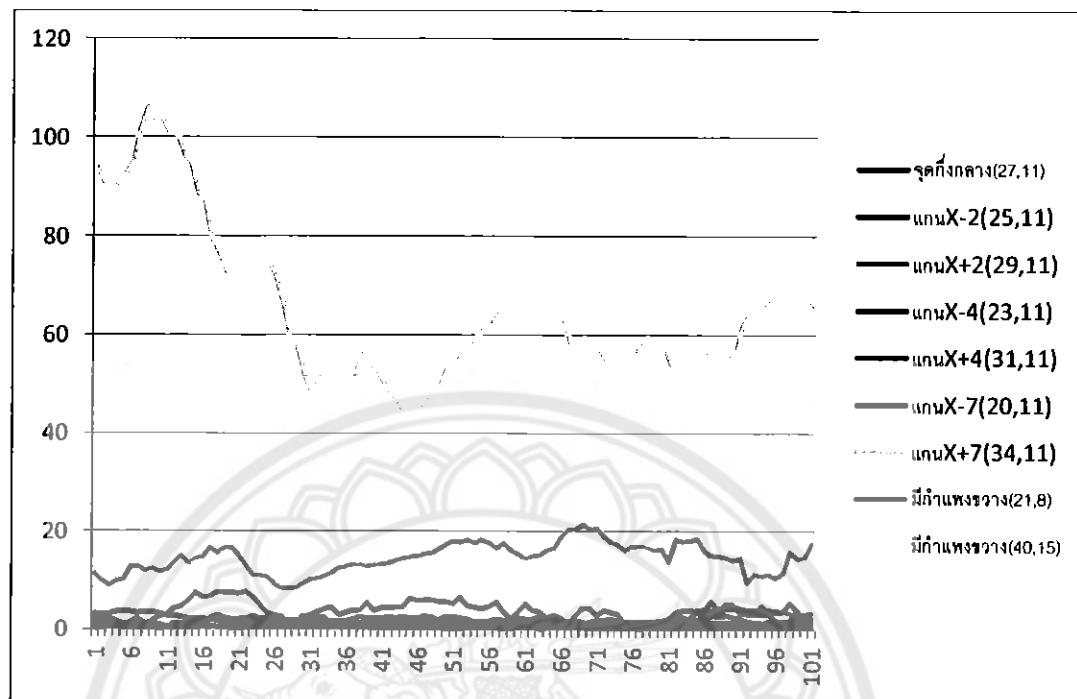
รูปที่ 5.19 วัสดุระบบที่พิกัด (34,11) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (34.50,11.18)



รูปที่ 5.20 วัสดุที่พิกัด (21,8) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (11.49,17.50)



รูปที่ 5.21 วัสดุที่พิกัด (40,15) พิกัดเฉลี่ยที่คำนวณได้จากโปรแกรมคือ (49.49,-49.30)



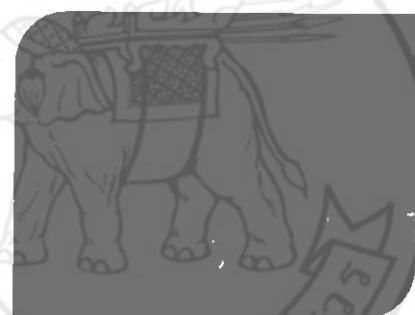
รูปที่ 5.22 กราฟแสดงค่าความคลาดเคลื่อนระหว่างค่าที่คำนวณได้กับค่าจริง



รูปที่ 5.23 การติดตั้ง Xbee Reference Node ที่ 1 บนเพดาน



รูปที่ 5.24 การติดตั้ง Xbee Reference Node ที่ 2 บนเพดาน



รูปที่ 5.25 การติดตั้ง Xbee Reference Node ที่ 3 บนเพดาน



รูปที่ 5.26 การวัดค่า RSSI จาก Xbee Tracking Node

บทที่ 6 สรุปผล

6.1 ผลการทดลอง

6.1.1 วัดระยะทางภายในอาคารจาก Node3 ไป Node1 เพื่อหาจำนวน Block ที่เหมาะสมของ Moving Average Filter

จากการทดลองจะพบว่าเมื่อใช้จำนวนข้อมูลเฉลี่ยมากจะทำให้สัญญาณหายไปในช่วงเวลาในการเกลื่อนที่ของพิกัดเนื่องจากต้องเฉลี่ยค่าพิกัดจากข้อมูลเริ่มต้นจนถึงข้อมูลสุดท้าย แต่เปอร์เซ็นต์ความคลาดเคลื่อนก็จะลดลงเมื่อใช้จำนวนข้อมูลเฉลี่ยมากขึ้น ซึ่งจุดที่เหมาะสมระหว่างเวลาที่ใช้กับเปอร์เซ็นต์ความคลาดเคลื่อนคือ $M = 25$ เพราะให้เวลาอัน้อยแต่เมื่อเปอร์เซ็นต์ความคลาดเคลื่อนต่ำ

6.1.2 วัดระยะทางภายในอาคารโดยใช้ Moving Average Filter

จากการทดลองจะเห็นได้ว่าค่าที่ระบบใกล้กับจุดสังเกตผลที่ออกมากจากการวัดระหว่างค่าที่ได้จากโปรแกรมและค่าจริงจะใกล้เคียงกัน และเมื่อระบบห่างจากจุดสังเกตออกไปจะเห็นว่าค่าที่วัดได้จากโปรแกรมและค่าที่วัดได้จริงมีความคลาดเคลื่อนมากขึ้นซึ่งระบบห่างมากก็จะคลาดเคลื่อนมากขึ้นด้วยจนกระทั่งถึงค่าหนึ่งระยะทางก็จะ Drop ลงมา

6.1.3 วัดระยะทางภายในอาคารโดยใช้จุดสังเกต 3 จุด

จากการทดลองสังเกตได้ว่าที่จุดศูนย์กลางของสามเหลี่ยม ค่าที่วัดได้ความคลาดเคลื่อนจะน้อยกว่าความคลาดเคลื่อนมากขึ้นเมื่อเกลื่อนที่ห่างออกมากจากจุดศูนย์กลางของสามเหลี่ยม และเมื่อวัดค่าในที่ที่มีกำแพงขวางกั้นค่าที่วัดได้จากโปรแกรมและค่าจริงจะแตกต่างกันมาก

6.2 ปัญหาและอุปสรรคในการพัฒนา

- พื้นที่ที่ใช้ทดสอบนั้นจะต้องกำหนดตัวแปรค่าๆ ให้ดี เพราะค่าความเข้มสัญญาณมีความแปรปรวนได้ง่ายจากหลายปัจจัย เช่น อุณหภูมิ สิ่งกีดขวาง คลื่นความถี่อื่นๆ เป็นต้น
- แรงดันไฟฟ้าที่ใช้จะต้องจ่ายกระแสไฟฟ้าที่จึงจะได้ความเข้มของสัญญาณที่แน่นอน
- ระยะทางที่ใกล้ขึ้นสัญญาณจะมีความแปรปรวนมากขึ้น ทำให้ยากต่อการวิเคราะห์หาตำแหน่งวัด
- ตัวอุปกรณ์ xbee เองมีการส่งค่า RSSI ที่ไม่แน่นอนทำให้การซึ่งเราไม่สามารถแก้ไขได้

6.3 แนวทางการพัฒนาต่อในอนาคต

1. การศึกษา Filler ที่ใช้ในการกรองข้อมูล เช่น ใช้ค่าเมนฟิวเตอร์ทำงานแทนเหตุการณ์ได้จากข้อมูลที่มีอยู่ เพื่อได้ตำแหน่งที่แน่นอน
2. เพิ่ม Node อ้างอิงที่ใช้เก็บระยะทางในการหาตำแหน่ง เมื่อมี Node อ้างอิงมากก็จะสามารถเลือกค่า RSSI ได้ยอดเยี่ยมตามไปด้วย โดยจะต้องมีอัลกอริทึมในการตัดสินใจเดือกว่า Node ใดมีค่าความเข้มที่น่าเชื่อถือ
3. มีอัลกอริทึมในการจัดวางตำแหน่งของ Reference Node ที่สามารถบอกได้ว่าตรงจุดใดที่ควรจะเป็นจุดอ้างอิงได้ดีจากการใส่ตัวแปรต่างๆ ลงไป เช่น มีตู้ขนาด 12×8 ฟุต วางอยู่ 3 ตู้ ตรงจุด (12,4) มีตู้ขนาด 10×3 ฟุต ตรงจุด (5,3) มีตู้ป้ายขนาด 8×7 ฟุต ตรงจุด (23,11) การจะวาง Reference Node ตรงไหนดึงจะดี ถึงจะสามารถส่งสัญญาณไปไม่มีสิ่งกีดขวางได้

6.4 สรุป

จากการทดลองและเก็บค่า RSSI นั้น ค่าที่ได้จะมีความแปรปรวนได้ง่าย จึงต้องกำหนดตัวแปรต่างๆ ให้คงที่เวลาที่เราวัดผล แต่ก็ยังไม่พอจึงต้องมีการใช้ Moving Average Filter มาช่วยในการกรองความเข้มสัญญาณจึงจะทำให้ได้ค่าที่น่าเชื่อถือ ซึ่งการใช้ Moving Average Filter มาช่วยในการกรองค่าความเข้มจะต้องเลือกจำนวนข้อมูลเฉลี่ยที่เหมาะสมและเวลาที่ใช้จะต้องยอมรับได้ดึงจะสามารถได้พิกัดที่แม่นยำและถูกต้องภายในเวลาที่ต้องการ และใช้กุญแจทางคณิตศาสตร์มาช่วยในการหาพิกัดของวัตถุซึ่งผลการทดลองทำให้ได้ตำแหน่งที่น่าพอใจ มีความคลาดเคลื่อนน้อย

อย่างไรก็ตามแม้จะได้ผลที่น่าพอใจ แต่ถ้าระบบทางที่ไกลออกไปการคำนวณพิกัดของวัตถุก็จะคลาดเคลื่อนตามไปด้วย จึงต้องมีกลไกการป้องกันเหตุการณ์นี้ เช่น การเพิ่ม Node อ้างอิงให้มากขึ้น และมีอัลกอริทึมที่หาเส้นทางที่ใกล้กับ Node Tracking เพื่อหา Node ที่มีระยะที่เหมาะสมในการคำนวณ เพื่อให้ได้ค่าพิกัดที่ถูกต้องได้

เอกสารอ้างอิง

- [1] Srividya Iyer, Receive Signal Strength Indicator [online] [cited 2007 Jun 22]. Available from: http://www.birds-eye.net/definition/r/rssi-receive_signal_strength_indicator.shtml
- [2] Dr.techn. Wolfgang Kastner Wireless Communication in Home and Building Automation [online] [cited 2007 Jun 22]. Available from: www.auto.tuwien.ac.at/~creinisch/WirelessCommunicationinHomeandBuildingAutomation.pdf
- [3] F. L. LEWIS Wireless Sensor Networks [online] [cited 2007 Jun 24]. Available from: <http://lecs.cs.ucla.edu/~girod/official/talks/localization-poster.ppt>
- [4] ZigBee RF Modules by Digi International 2010 Digi International [online] [cited 2010 Jun 24]. Available from: <http://www.scribd.com/doc/102884177/148/AT-Command-Queue-Parameter-Value>
- [5] IEEE SIGNAL PROCESSING MAGAZINE Locating the Nodes [Cooperative localization in wireless sensor networks]
- [6] L.S.Jayashree On the Accuracy of Centroid based Multilateration Procedure for Location Discovery in Wireless Sensor Networks[online] [cited 2007 Jun 27]. Available from: <http://ieeexplore.ieee.org/iel5/11053/34866/01666565.pdf>
- [7] Seaphahn Meguerdichian1 Sasa Slijepcevic1 Vahag Karayan2 Miodrag Potkonjak1 Localized Algorithms In Wireless Ad-Hoc Networks:Location Discovery And Sensor Exposure [online] [cited 2007 Jun 22]. Available from: <http://citeseer.ist.psu.edu/meguerdichian01localized.html>
- [8] The Hands-On XBee Lab Manual : [online] [cited 2010Nov 18]. Available from: http://books.google.co.th/books?id=kzsJ7yMcAIC&dq=xbee+ATND+hex&hl=th&source=gbs_navlinks_s
- [9] The moving average filter [online] [cited 2003Fab21]. Available from: http://www.analog.com/static/imported-files/tech_docs/dsp_book_Ch15.pdf
- [10] Cramer's rule [online] [cited 2008 Jan 05]. Available from:http://en.wikipedia.org/wiki/Cramer's_rule

ภาคผนวก ก.
เครื่องมือในการพัฒนา

- Microsoft Visual C# Studio 2010 Express
- Microsoft Office Excel 2010
- Microsoft Powerpoint 2010
- X-CTU Digi International Version 5.1.4.1
- Adobe Photoshop CS5
- Calculat
- Paint

