



## คำปริญญ์อักษรียะ

โดย

|                 |                 |                       |
|-----------------|-----------------|-----------------------|
| นาย เฉลิมศักดิ์ | อรุณโรจน์ปัญญา  | รหัสประจำตัว B5412570 |
| นาย กิตติวัฒน์  | สุขหงษ์         | รหัสประจำตัว B5415168 |
| นาย กันต์ธร     | กิริติเพียรเลิศ | รหัสประจำตัว B5418657 |

รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 527499 โครงงานวิศวกรรมโทรคมนาคม  
หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรปรับปรุง พ.ศ. 2554  
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี  
ประจำภาคการศึกษาที่ 1 ปีการศึกษา 2557

## คู่มือปริญญ์อักษรียะ

คณะกรรมการสอบโครงการ



(ผู้ช่วยศาสตราจารย์ เรืออากาศเอก ดร. ประ โยชน์ คำสวัสดิ์)  
อาจารย์ที่ปรึกษาโครงการ



(ผู้ช่วยศาสตราจารย์ ดร. ปิยาภรณ์ มีสวัสดิ์)  
กรรมการ



(อาจารย์ ดร.รณเสฏฐ์ ทศศิริพัฒน์)  
กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับรายงานโครงการฉบับนี้ เป็นส่วนหนึ่งของการศึกษา  
ระดับปริญญาตรี สาขาวิชาวิศวกรรมโทรคมนาคม รายวิชา 527499 โครงการวิศวกรรมโทรคมนาคม  
ประจำปีการศึกษา 2557

|                  |                                                       |                 |                       |
|------------------|-------------------------------------------------------|-----------------|-----------------------|
| โครงการ          | ตู้ไปรษณีย์อัจฉริยะ                                   |                 |                       |
| ผู้ดำเนินงาน     | 1. นาย เฉลิมศักดิ์                                    | อรุณโรจน์ปัญญา  | รหัสประจำตัว B5412570 |
|                  | 2. นาย กิตติวัฒน์                                     | สุขหงษ์         | รหัสประจำตัว B5415168 |
|                  | 3. นาย กันต์ธร                                        | กิริติเพียรเลิศ | รหัสประจำตัว B5418657 |
| อาจารย์ที่ปรึกษา | ผู้ช่วยศาสตราจารย์ เรืออากาศเอก ดร.ประโยชน์ คำสวัสดิ์ |                 |                       |
| สาขาวิชา         | วิศวกรรมโทรคมนาคม                                     |                 |                       |
| ภาคการศึกษา      | 1/2557                                                |                 |                       |

บทคัดย่อ  
(Abstract)

โครงการนี้นำเสนอ ตู้ไปรษณีย์อัจฉริยะ โดยการนำเทคโนโลยีไร้สายจากอุปกรณ์ที่เรียกว่า X-Bee รุ่น Series2 (ZB) มาใช้ในการรับ-ส่งข้อมูลสัญญาณ โดยปรับตั้งให้งานในโหมด API ซึ่งเป็นโหมดที่ X-Bee สามารถทำงานเป็นตัวควบคุม (Controller) ในตัว โดยสามารถใช้ควบคุมอุปกรณ์ที่มาเชื่อมต่อได้โดยตรง ในการทดสอบการทำงานของตู้ไปรษณีย์สามารถแจ้งเตือนเจ้าของบ้านว่ามีจดหมายมาส่งแล้ว ทั้งในรูปแบบของเสียง และ ส่งข้อความ (SMS) เข้าโทรศัพท์มือถือ โดยผู้ออกแบบได้ใช้เซนเซอร์ ตรวจสอบการสะท้อน เพื่อตรวจสอบว่ามีจดหมายเข้ามาหรือไม่ หลังจากนั้นตู้ไปรษณีย์จะส่งสัญญาณแบบไร้สายไปยังเครื่องรับที่อยู่ที่บ้านเพื่อเตือนเจ้าของบ้านให้ทราบว่า มีจดหมายมาส่ง ทั้งในกรณีที่ผู้อยู่อาศัยอยู่ภายในบ้านเป็นเสียงแจ้งเตือน และกรณีที่ผู้อาศัยไม่ได้อยู่บ้านจะเป็นการส่งข้อความเข้าโทรศัพท์มือถือ

## กิตติกรรมประกาศ

(Acknowledgement)

การจัดทำโครงการเรื่องคู่มือปริญญ์อักษรียะได้ประสบความสำเร็จด้วยดีเนื่องจากได้รับความอนุเคราะห์ในการให้คำปรึกษาในด้านต่างๆในระหว่างการดำเนินการจากบุคคลหลายท่านที่ได้ให้ความช่วยเหลือและให้คำปรึกษารวมทั้งข้อเสนอแนะที่เป็นประโยชน์ในการทำโครงการครั้งนี้ซึ่งบุคคลเหล่านี้ประกอบไปด้วย

ผศ.ร.อ. ดร.ประโยชน์ คำสวัสดิ์ (อาจารย์ที่ปรึกษาที่ปรึกษาโครงการ)

นายปัญญา หันตุลา (นักศึกษาปริญญาเอกสาขาวิชาวิศวกรรมโทรคมนาคม)

ข้าพเจ้าใคร่ขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลและเป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ตลอดจนให้การดูแลและให้ความเข้าใจเกี่ยวกับพื้นฐานการใช้งานโปรแกรม ซึ่งข้าพเจ้าขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ด้วย



นาย เฉลิมศักดิ์ อรุณโรจน์ปัญญา

นาย กิตติวัฒน์ สุขหงษ์

นาย กันต์ธร กิรติเพียรเลิศ

คณะผู้จัดทำ

## สารบัญ

| เรื่อง                                                      | หน้า |
|-------------------------------------------------------------|------|
| บทคัดย่อ                                                    | ก    |
| กิตติกรรมประกาศ                                             | ข    |
| สารบัญ                                                      | ค    |
| สารบัญรูป                                                   | ช    |
| สารบัญตาราง                                                 | ญ    |
| <b>บทที่ 1 บทนำ</b>                                         | 1    |
| 1.1 ความเป็นมา                                              | 1    |
| 1.2 วัตถุประสงค์                                            | 2    |
| 1.3 ขอบเขตงาน                                               | 2    |
| 1.4 ขั้นตอนการดำเนินงาน                                     | 3    |
| 1.5 ประโยชน์ที่คาดว่าจะได้รับ                               | 3    |
| <b>บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง</b>                 | 4    |
| 2.1 บทนำ                                                    | 4    |
| 2.2 โมดูล X-Bee Pro รุ่น Series 2                           | 5    |
| 2.2.1 Introduction X-Bee BASIC                              | 6    |
| 2.2.2 การจัดการการทำงานของ X-Bee                            | 8    |
| 2.2.3 การออกแบบรูปแบบการสื่อสารของโมดูล X-Bee               | 10   |
| 2.3 QTR-1RC Reflectance Sensor                              | 12   |
| 2.3.1 คุณสมบัติ                                             | 13   |
| 2.3.2 การนำไปใช้                                            | 13   |
| 2.4 วงจรบันทึกเสียงดิจิทัล รุ่น FK941                       | 14   |
| 2.4.1 ข้อมูลทางด้านเทคนิค                                   | 15   |
| 2.4.2 การทำงานของวงจร                                       | 15   |
| 2.4.3 การนำไปใช้งาน                                         | 15   |
| 2.5 การใช้งานบอร์ดไมโครคอนโทรลเลอร์ รุ่น ET-BASE GSM SIM900 | 18   |
| 2.5.1 คุณสมบัติของบอร์ด ET-BASE GSM SIM900                  | 18   |
| 2.5.2 คุณสมบัติเบื้องต้นของโมดูล SIM900                     | 19   |

## สารบัญ (ต่อ)

| เรื่อง                                                                   | หน้า |
|--------------------------------------------------------------------------|------|
| 2.5.3 ส่วนประกอบของบอร์ด ET-BASE GSM SIM900                              | 19   |
| 2.5.4 การเชื่อมต่อบอร์ด ET-BASE GSM SIM900<br>กับ บอร์ดไมโครคอนโทรลเลอร์ | 22   |
| 2.6 การใช้งานบอร์ดไมโครคอนโทรลเลอร์ รุ่น ET-BASE51 AC3                   | 23   |
| 2.6.1 คุณสมบัติของบอร์ด                                                  | 24   |
| 2.6.2 โครงสร้างบอร์ด ET-BASE51 AC3 (AT89C51AC3)                          | 25   |
| 2.6.3 ขั้วต่อสัญญาณต่างๆ                                                 | 26   |
| <b>บทที่ 3 หลักการทำงานและการออกแบบระบบของโครงงาน</b>                    | 27   |
| 3.1 บทนำ                                                                 | 27   |
| 3.2 การออกแบบฮาร์ดแวร์ระบบตู้ไปรษณีย์อัจฉริยะ                            | 27   |
| 3.2.1 หลักการทำงาน                                                       | 27   |
| 3.2.2 อุปกรณ์ที่ใช้ในการออกแบบ                                           | 28   |
| 3.2.3 หลักการทำงานของตู้ไปรษณีย์อัจฉริยะ                                 | 31   |
| 3.2.4 การใช้งานของตู้ไปรษณีย์อัจฉริยะ                                    | 32   |
| 3.3 การออกแบบซอฟต์แวร์ระบบตู้ไปรษณีย์อัจฉริยะ                            | 36   |
| <b>บทที่ 4 ผลการทดลอง</b>                                                | 38   |
| 4.1 บทนำ                                                                 | 38   |
| 4.2 การทดลองที่ 1 การทดสอบการส่ง-รับข้อมูลของ X-Bee ภาครับและภาคส่ง      | 38   |
| 4.2.1 วัตถุประสงค์                                                       | 38   |
| 4.2.2 ขั้นตอนการทดลอง                                                    | 38   |
| 4.2.3 ผลการทดลองที่ 1                                                    | 39   |
| 4.2.4 วิเคราะห์ผลการทดลองที่ 1                                           | 40   |
| 4.2.5 สรุปผลการทดลองที่ 1                                                | 42   |

## สารบัญ (ต่อ)

| เรื่อง                                                                                       | หน้า |
|----------------------------------------------------------------------------------------------|------|
| 4.3 การทดลองที่ 2 การทดลองการรับข้อมูลของโมดูลX-Bee เมื่อมีการเปลี่ยนแปลงของขา X-Bee         | 43   |
| 4.3.1 วัตถุประสงค์                                                                           | 43   |
| 4.3.2 ขั้นตอนการทดลอง                                                                        | 43   |
| 4.3.3 ผลการทดลองที่ 2                                                                        | 45   |
| 4.3.4 วิเคราะห์ผลการทดลองที่ 2                                                               | 46   |
| 4.3.5 สรุปผลการทดลองที่ 2                                                                    | 48   |
| 4.4 การทดลองที่ 3 การทดสอบการส่งข้อความของ ET-BASE GSM SIM900 ผ่านทาง บอร์ดไมโครคอนโทรลเลอร์ | 48   |
| 4.4.1 วัตถุประสงค์                                                                           | 48   |
| 4.4.2 ขั้นตอนการทดลอง                                                                        | 48   |
| 4.4.3 ผลการทดลองที่ 3                                                                        | 50   |
| 4.4.4 วิเคราะห์ผลการทดลองที่ 3                                                               | 50   |
| 4.4.5 สรุปผลการทดลองที่ 3                                                                    | 51   |
| 4.5 การทดลองที่ 4 การทดสอบการทำงานของระบบตู้ไปรษณีย์อัจฉริยะ                                 | 51   |
| 4.5.1 วัตถุประสงค์                                                                           | 51   |
| 4.5.2 ขั้นตอนการทดลอง                                                                        | 51   |
| 4.5.3 ผลการทดลองที่ 4                                                                        | 53   |
| 4.5.4 วิเคราะห์ผลการทดลองที่ 4                                                               | 55   |
| 4.5.5 สรุปผลการทดลองที่ 4                                                                    | 55   |
| 4.6 สรุปผลการทดลอง                                                                           | 56   |
| บทที่ 5 สรุปผลการทดสอบและข้อเสนอแนะ                                                          | 57   |
| 5.1 สรุปผลการทดลอง                                                                           | 57   |
| 5.2 ปัญหาและอุปสรรค                                                                          | 57   |
| 5.3 ข้อเสนอแนะ                                                                               | 57   |
| ประวัติผู้เขียน                                                                              | 58   |
| บรรณานุกรม                                                                                   | 59   |

## สารบัญ (ต่อ)

เรื่อง

หน้า

ภาคผนวก ก

60

ภาคผนวก ข

68



## สารบัญรูป

| เรื่อง                                                                         | หน้า |
|--------------------------------------------------------------------------------|------|
| รูปที่ 2.1 X-Bee PRO 50 mW UFL Connection -Series 2 (ZB)และสายอากาศ(Antenna)   | 5    |
| รูปที่ 2.2 ย่านความถี่ใช้งานตามมาตรฐาน                                         | 6    |
| รูปที่ 2.3 X-Bee Stack                                                         | 7    |
| รูปที่ 2.4 ขาการทำงานของ X-Bee                                                 | 8    |
| รูปที่ 2.5 การสื่อสารระหว่าง 2 โหนด เป็นแบบ Point-to-Point                     | 10   |
| รูปที่ 2.6 การส่งข้อมูลแบบสองทิศทาง                                            | 10   |
| รูปที่ 2.7 ทดสอบการส่งกลับของโมดูล X-Bee                                       | 11   |
| รูปที่ 2.8 การเชื่อมต่อไมโครคอนโทรลเลอร์ทั่วไปเข้ากับ โมดูล X-Bee              | 11   |
| รูปที่ 2.9 การเชื่อมต่อ โมดูล X-Bee เข้ากับเซนเซอร์                            | 12   |
| รูปที่ 2.10 QTR-1RC Reflectance Sensor                                         | 12   |
| รูปที่ 2.11 ขา QTR-1RC Reflectance Sensor                                      | 14   |
| รูปที่ 2.12 อุปกรณ์ บันทึกเสียงดิจิตอล รุ่น FK941                              | 14   |
| รูปที่ 2.13 วงจรบันทึกเสียงดิจิตอล 20-60 วินาที พร้อมขยายเสียง 2 วัตต์         | 17   |
| รูปที่ 2.14 การต่อใช้งาน                                                       | 17   |
| รูปที่ 2.15 บอร์ด ET-BASE GSM SIM900                                           | 18   |
| รูปที่ 2.16 ส่วนประกอบET-BASE GSM SIM900                                       | 19   |
| รูปที่ 2.17 การเชื่อมต่อ ET-BASE GSM SIM900 ร่วมกับบอร์ด ET-BASE AVREASY32U4   | 22   |
| รูปที่ 2.18 แผนผังวงจร บอร์ด ET-BASE GSM SIM900                                | 23   |
| รูปที่ 2.19 บอร์ด ET-BASE51 AC3 (AT89C51AC3)                                   | 23   |
| รูปที่ 2.20 โครงสร้างบอร์ด ET-BASE51 AC3 (AT89C51AC3)                          | 25   |
| รูปที่ 3.1 แผนภาพหลักการทำงาน                                                  | 27   |
| รูปที่ 3.2 บอร์ดไมโครคอนโทรลเลอร์รุ่น ET-BASE51 AC3                            | 28   |
| รูปที่ 3.3 โมดูล X-Bee PRO 50mW U.F.L Connection - Series 2 (ZB) (ด้านซ้ายมือ) | 29   |
| รูปที่ 3.3 บอร์ดเชื่อมโมดูล X-Bee (ด้านขวามือ)                                 | 29   |
| รูปที่ 3.4 สายอากาศแบบ U.F.L (ด้านซ้ายมือ) และ Interface Cable (ด้านขวามือ)    | 29   |
| รูปที่ 3.5 บอร์ด ET-BASE GSM SIM900                                            | 29   |
| รูปที่ 3.6 วงจรบันทึกเสียงดิจิตอล รุ่น FK941                                   | 30   |

## สารบัญรูป (ต่อ)

| เรื่อง                                                                                                     | หน้า |
|------------------------------------------------------------------------------------------------------------|------|
| รูปที่ 3.7 เซนเซอร์วัดการสะท้อน QTR-1RC                                                                    | 30   |
| รูปที่ 3.8 ตู้ไปรษณีย์ที่ออกแบบขึ้น                                                                        | 30   |
| รูปที่ 3.9 แผนภาพการทำงานตู้ไปรษณีย์อัจฉริยะ                                                               | 31   |
| รูปที่ 3.10 กดปุ่มรีเซ็ตที่บอร์ดไมโครคอนโทรลเลอร์                                                          | 32   |
| รูปที่ 3.11 เซนเซอร์ที่ติดอยู่ที่กล่องไปรษณีย์                                                             | 33   |
| รูปที่ 3.12 เมื่อมีวัตถุมาเป็นตัวสะท้อนกับตัวรับของเซนเซอร์ เซนเซอร์จะเปลี่ยนสถานะ                         | 33   |
| รูปที่ 3.13 เสียงเตือนว่ามีจดหมายมาส่งดังขึ้น 3 ครั้ง                                                      | 34   |
| รูปที่ 3.14 ข้อความที่ได้รับทางโทรศัพท์มือถือ                                                              | 34   |
| รูปที่ 3.15 การกดปุ่มรีเซ็ตที่บอร์ดไมโครคอนโทรลเลอร์ภายในบ้าน                                              | 35   |
| รูปที่ 3.16 หลอดไฟ LED ที่ P1.0 จะกะพริบ 3 ครั้ง                                                           | 35   |
| รูปที่ 3.17 การเขียนโปรแกรมการรับเฟรมข้อมูล (Frame Data) จาก โมดูล X-Bee                                   | 36   |
| รูปที่ 4.1 หน้าต่าง Terminal ของภาคส่ง (Coordinator) และภาครับ (Router)                                    | 39   |
| รูปที่ 4.2 เฟรมข้อมูล (Frame Data) ที่ทำการส่งจากภาคส่ง (Coordinator) ไปยังภาครับ (Router)                 | 40   |
| รูปที่ 4.3 รูปแบบของโปรโตคอล API                                                                           | 42   |
| รูปที่ 4.4 การตั้งค่า Destination และ การกำหนดขาอินพุตเป็น DI                                              | 44   |
| รูปที่ 4.5 โครงสร้าง IC ของขาอินพุต X-Bee                                                                  | 44   |
| รูปที่ 4.6 การตั้งค่า Digital IO Change Detection ให้กับขาอินพุตของ X-Bee                                  | 45   |
| รูปที่ 4.7 ผลตอบสนองของเฟรมข้อมูล (Frame Data)<br>เมื่อยังไม่มีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์             | 45   |
| รูปที่ 4.8 ผลตอบสนองของเฟรมข้อมูล (Frame Data)<br>เมื่อมีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์แล้ว               | 46   |
| รูปที่ 4.9 เชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ ET-BASE51 AC3<br>และบอร์ด ET-BASE GSM SIM900 ด้วยพอร์ต ET-RS232 | 49   |
| รูปที่ 4.10 โค้ดคำสั่งที่ใช้ในการส่งข้อความ                                                                | 49   |
| รูปที่ 4.11 การส่งข้อความว่า “ You’ve got a letter in your mail box! ”                                     | 50   |
| รูปที่ 4.12 ภาพแสดงการต่อเซนเซอร์ในตู้ไปรษณีย์                                                             | 51   |
| รูปที่ 4.13 ภาพการแสดงการต่ออุปกรณ์ภาครับภายในบ้าน                                                         | 52   |
| รูปที่ 4.14 การทดสอบการส่งจดหมายในตู้ไปรษณีย์                                                              | 54   |
| รูปที่ 4.15 กล่องภาครับในบ้านจะส่งเสียงเตือน                                                               | 54   |
| รูปที่ 4.16 การส่งข้อความเพื่อแจ้งเตือนว่ามีจดหมายมาส่ง                                                    | 55   |

## สารบัญตาราง

| เรื่อง                                                                                                                                              | หน้า |
|-----------------------------------------------------------------------------------------------------------------------------------------------------|------|
| ตารางที่ 2.1 การจัดการการทำงานของ X-Bee                                                                                                             | 9    |
| ตารางที่ 2.2 แสดงการทำงานของแต่ละขาของบอร์ด ET-BASE GSM SIM900                                                                                      | 20   |
| ตารางที่ 4.1 การส่งเฟรมข้อมูล (Frame Data) จากภาคส่ง (Coordinator)<br>ไปยังภาครับ (Router) และเฟรมข้อมูลที่ได้รับได้จากภาคส่ง                       | 39   |
| ตารางที่ 4.3 การทดสอบการส่งข้อความของ ET-BASE GSM SIM900<br>ผ่านทางบอร์ดไมโครคอนโทรลเลอร์                                                           | 50   |
| ตารางที่ 4.4 การตรวจสอบการรับ-ส่งสัญญาณระหว่างโมดูล X-Bee<br>ภาครับกับโมดูล X-Bee ภาคส่งที่ต่อกับเซนเซอร์และการส่งข้อความผ่าน<br>ET-BASE GSM SIM900 | 53   |



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมา

ปัจจุบันเทคโนโลยีการสื่อสารแบบไร้สายได้รับความนิยมและมีบทบาทที่สำคัญอย่างมากในชีวิตประจำวัน รูปแบบของการสื่อสารแบบไร้สายก็มีหลายรูปแบบ เช่น Wifi, Bluetooth, Wireless เป็นต้น โมดูล X-Bee ก็ถือได้ว่าเป็นอุปกรณ์อีกรูปแบบหนึ่งที่สามารถสื่อสารแบบไร้สายได้ โดยโมดูล X-Bee มีการรับส่งข้อมูลที่มีลักษณะคล้ายกับ Wifi และ Bluetooth แต่สื่อสารกันไม่ได้เพราะว่าโปรโตคอล (Protocol) ที่ใช้ในการสื่อสารนั้นไม่เหมือนกัน โดย X-Bee จะนำเอา Physical Layer และ MAC Layer ของ IEEE 802.15.4 ซึ่งเป็นมาตรฐานการกำหนดการสื่อสารไร้สายแบบ WPAN (Wireless Personal Area Network) และยังมีข้อดีคือ มีอัตราการรับส่งข้อมูลต่ำ ใช้พลังงานต่ำ ราคาถูก และสามารถสร้างระบบที่เรียกว่า Wireless Sensor Network ได้ ซึ่งระบบนี้จะสามารถทำงาน ในร่ม กลางแจ้ง ทนแดด ทนฝน และอยู่ได้ด้วยแบตเตอรี่ก้อนเล็ก เช่น ถ่าน AA 4 ก้อน

ดังนั้น คณะผู้จัดทำจึงมีความสนใจที่จะนำโมดูล X-Bee ไปประยุกต์ใช้กับตู้รับไปรษณีย์ โดยโครงการนี้จะใช้โมดูล X-Bee Pro รุ่น Series 2 (ZB) จำนวน 2 ตัว เป็นด้านส่ง และด้านรับ ซึ่งใช้รูปแบบการส่งแบบ Point to Point และมีการทำงานในโหมด API โดยที่โมดูล X-Bee ที่อยู่ภายนอกบ้านทำหน้าที่เป็นภาคส่งซึ่งต่อกับเซนเซอร์ตรวจจับการสะทอน เมื่อมีจดหมายหย่อนเข้าไปในตู้ X-Bee ภาคส่งจะส่งเฟรมไปยัง X-Bee ฟังก์ชันรับซึ่งอยู่ภายในบ้านซึ่งต่ออยู่กับไมโครคอนโทรลเลอร์ ซึ่งจะนำข้อมูลไปประมวลผล และไมโครคอนโทรลเลอร์ จะส่งคำสั่งไปเพื่อให้มีเสียงเตือน “มีจดหมายมาส่ง” พร้อมทั้งส่ง ข้อความ (SMS) ผ่านทาง บอร์ด GSM-SIM900 ด้วย เพื่อให้เจ้าของบ้านได้รับรู้ทั้งในกรณีที่อยู่ที่บ้านและอยู่นอกบ้าน

## 1.2 วัตถุประสงค์

1. เพื่อศึกษาการทำงานของ การสื่อสารไร้สายใช้โมดูล X-Bee Pro รุ่น Series 2 (ZB)
2. เพื่อศึกษาโปรแกรม X-CTU, โปรแกรม X-Bee API Demo, โปรแกรม Flip 2.4.4 ,  
โปรแกรม Keil uVision3 โปรแกรม Tera Term
3. เพื่อศึกษาการทำงานของโมดูล บันทึกเสียงดิจิทัล รุ่น FK941, บอร์ด  
ไมโครคอนโทรลเลอร์รุ่น ET-BASE51 AC3, บอร์ด ET-BASE GSM SIM900
4. เพื่อศึกษาการทำงานของ เซนเซอร์ตรวจจับการสะท้อน รุ่น QTR-1RC
5. เพื่อฝึกการแก้ไขปัญหาต่างๆที่เกิดขึ้นในระหว่างปฏิบัติงานจริง

## 1.3 ขอบเขตการทำงาน

1. ศึกษาการทำงานของ การสื่อสารไร้สายใช้โมดูล X-Bee Pro รุ่น Series 2 (ZB)
2. ศึกษาโปรแกรม X-CTU, โปรแกรม X-Bee API Demo, โปรแกรม Flip 2.4.4 ,  
โปรแกรม Keil uVision3 โปรแกรม Tera Term
3. ศึกษาการทำงานของโมดูล บันทึกเสียงดิจิทัล รุ่น FK941, บอร์ดไมโครคอนโทรลเลอร์  
รุ่น ET-BASE51 AC3, บอร์ด ET-BASE GSM SIM900
4. ศึกษาการทำงานของ เซนเซอร์ตรวจจับการสะท้อน รุ่น QTR-1RC
5. ทดสอบระบบการบอกสถานะของตู้ไปรษณีย์โดยผ่าน X-Bee Pro รุ่น Series 2 (ZB)

## 1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาเรื่องที่ต้องการทำโครงการแล้วนำมาปรึกษาอาจารย์ที่ปรึกษา และเลือกโครงการ  
เรื่อง ตู้ไปรษณีย์อัจฉริยะ
2. ศึกษาข้อมูลนำมาประกอบกับความรู้ทางทฤษฎีที่เกี่ยวข้องกับหัวข้อ หาข้อมูลเพิ่มเติม  
เกี่ยวกับหัวข้อที่ต้องการศึกษา
3. ทำการจัดซื้ออุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในการทำโครงการ ศึกษาคุณลักษณะอุปกรณ์  
และการใช้งานของอุปกรณ์อิเล็กทรอนิกส์เหล่านั้น
4. ศึกษาโปรแกรมในการควบคุมการทำงาน โดยใช้โปรแกรม X-CTU, โปรแกรม X-Bee  
API Demo, โปรแกรม Flip 2.4.4 ,โปรแกรม Keil uVision3 โปรแกรม Tera Term

- ทดลองใช้งานตัวโปรยณียัจฉริยะ และปรับแต่งการทำงานตรวจสอบโปรแกรมและอุปกรณ์อิเล็กทรอนิกส์ ตลอดจนการเตรียมเอกสารจัดทำเอกสารและนำเสนอโครงการ

### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

- มีความรู้เกี่ยวกับการทำงานของการสื่อสารไร้สายโดยใช้โมดูล X-Bee Pro รุ่น Series2 (ZB)
- มีความรู้เกี่ยวกับโปรแกรม X-CTU, โปรแกรม X-Bee API Demo, โปรแกรม Flip 2.4.4 ,โปรแกรม Keil uVision3 โปรแกรม Tera Term
- มีความรู้เกี่ยวกับการทำงานของโมดูลบันทึกเสียงดิจิทัลรุ่น FK941,บอร์ดไมโครคอนโทรลเลอร์รุ่น ET-BASE51 AC3, บอร์ด ET-BASE GSM SIM900
- มีความรู้เกี่ยวกับการทำงานของเซนเซอร์ตรวจจับการสะท้อน รุ่น QTR-1RC
- สามารถทดสอบระบบการบอกสถานะของตัวโปรยณียัจฉริยะโดยผ่านโมดูล X-Bee Pro รุ่น Series2 (ZB)
- สามารถแก้ไขปัญหาจากปฏิบัติงานจริง เช่น การหาพอร์ตของ X-Bee ไม่พบ เป็นต้น



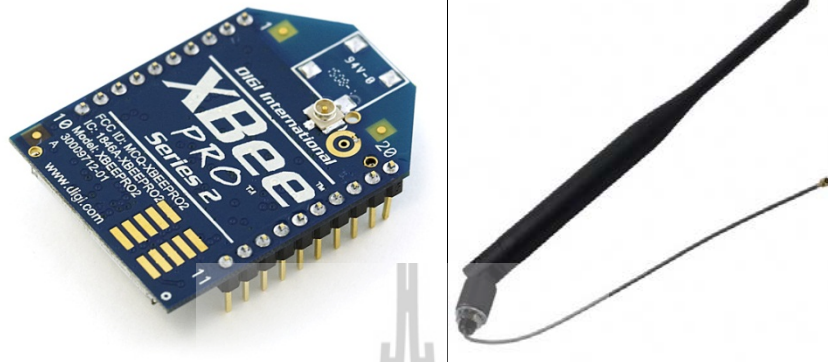
## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

#### 2.1 บทนำ

ระบบสื่อสารไร้สาย (Wireless Telecommunication) มีมากมายหลายรูปแบบ เช่น GSM , CDMA, วิทยุย่าน 27 MHz และ 433 MHz , Wireless Lan , Wifi , WiMax ฯลฯ สำหรับในสมัยก่อน การจะทำเครื่องส่งเครื่องรับต้องมีความรู้ทางด้าน RF Engineer ซึ่งจะสามารถออกแบบวงจร เครื่องส่งเครื่องรับ มีหน้าที่ทำการ Matching สายอากาศ การออกแบบสายอากาศสำหรับย่านความถี่ ต่างๆ ฯลฯ แต่ในปัจจุบันนี้มี IC RF ที่ Integrated รวมเอาภาคการออกแบบของ RF Engineer เข้าไป หลายส่วน ทำให้ง่ายในระดับที่ไม่ต้องมีความรู้ทางด้าน RF Engineer ก็สามารถสร้างวงจร ส่งและรับได้แล้วงานทางด้านไมโครคอนโทรลเลอร์จะมีความเกี่ยวข้องกับการสื่อสารทั้งสิ้น เช่น การสร้างเครือข่ายของระบบหนึ่งๆการติดต่อสื่อสารใช้งานอุปกรณ์ RF Module และที่คุ้นเคยกัน ดีก็คือการสื่อสารเพื่อใช้งานติดต่อกับอุปกรณ์อื่นๆ ผ่าน Interface ต่าง ๆ เช่น RS232 (UART), SPI, I2C, CAN, RS485, Ethernet, LAN, TCP/IP, USB ฯลฯ ผู้ที่เคยเขียนซอฟต์แวร์ที่เกี่ยวข้องกับการ สื่อสารต่างๆจะมีความเข้าใจเกี่ยวกับโปรโตคอลสื่อสารสามารถเรียนรู้ การสื่อสารแบบอื่นๆ ได้ไม่ยากการเขียนซอฟต์แวร์ลักษณะการรับ stream data เพื่อมาเก็บในบัฟเฟอร์แล้วทำการ encapsulate de-capsulate ข้อมูล C (เช่น การเขียนโปรแกรมทาง ด้าน network security , UART, I2C ฯลฯ) แล้วนำข้อมูลไปใช้งานเป็นสิ่ง จำเป็นสำหรับงานทาง ด้านการติดต่อสื่อสารเกือบทุกรูปแบบ

## 2.2 โมดูล X-Bee PRO รุ่น Series 2



รูปที่ 2.1 X-Bee PRO 50mW UFL Connection - Series 2 (ZB) และสายอากาศ (Antenna)

X-Bee Series 2 (ZB) เป็นโมดูล ความถี่ 2.4 GHz รับส่งสัญญาณไร้สาย ย่าน ตามมาตรฐานโปรโตคอล Xbee/IEEE 802.15.4 โดยใช้พลังงานต่ำ (ที่ 3.3 โวลต์) รับส่งข้อมูลอัตราความเร็ว 250 Kbps จุดต่อสายอากาศแบบ U.FL. (ต้องซื้อ Interface Cable และ เสาคู่เพิ่ม) รองรับเครือข่ายแบบ Mesh

### คุณสมบัติ

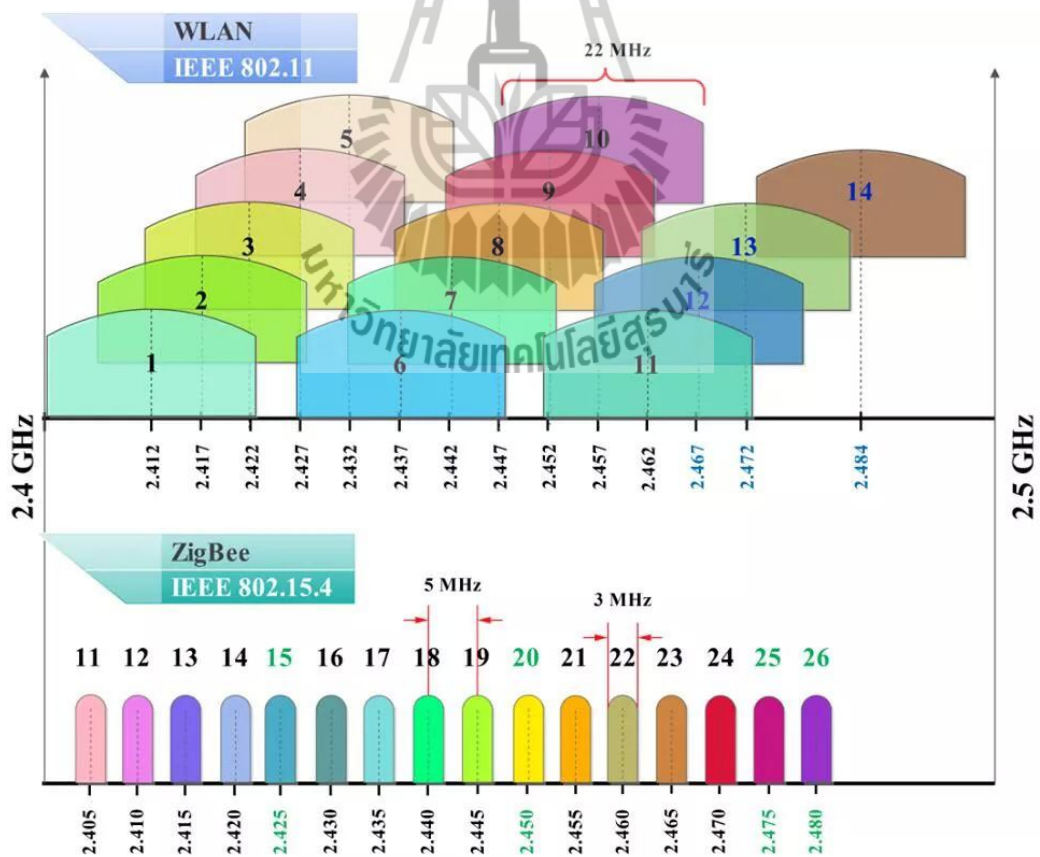
- 3.3 โวลต์ 295 มิลลิแอมป์
- 250 kbps Max data rate
- 50 mW output (+17 dBm)
- 1 mile (1600 m) range
- U.FL antenna connection
- Fully FCC certified
- 4 10-bit ADC input pins
- 10 digital IO pins
- 128-bit encryption
- Local or over-air configuration
- AT or API command set

### 2.2.1 Introduction X-Bee BASIC

- X-Bee คืออะไร

X-Bee มาตรฐานสากลกำหนดโดย X-Bee Alliance เป็นการสื่อสารแบบไร้สาย ที่มีอัตราการรับส่งข้อมูลต่ำ ใช้พลังงานต่ำ ราคาถูก จุดประสงค์ก็เพื่อให้สามารถสร้างระบบที่เรียกว่า Wireless Sensor Network ได้ ซึ่งระบบนี้ จะสามารถทำงานในร่มกลางแจ้ง ทุ่งนา หนองน้ำ และอยู่ได้ด้วยแบตเตอรี่ก้อนเล็ก (เช่น ถ่าน AA2 ก้อน) นานเป็นเดือนเป็นปี เหมาะสมใช้งานกับพวก Monitoring ต่าง ๆ

X-Bee กำหนดย่านความถี่ใช้งานตามมาตรฐานไว้ 3 ย่านความถี่คือ ย่าน 2.4 GHz , ย่าน 915 MHz และย่าน 868 MHz โดยแต่ละย่านจะมีช่องสัญญาณ 16 ช่อง , 10 ช่อง และ 1 ช่อง ตามลำดับ ส่วนอัตรารับส่งข้อมูล (ทางอากาศ) จะอยู่ที่ 250 Kbps , 40 Kbps , 20 Kbps ตามลำดับ



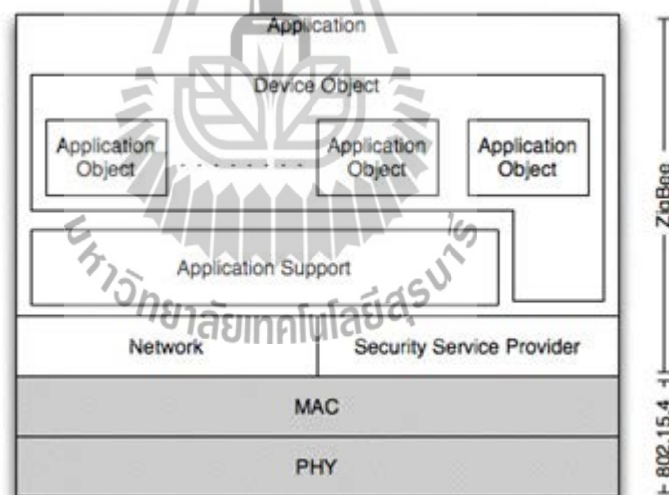
รูปที่ 2.2 ย่านความถี่ใช้งานตามมาตรฐาน

สรุปได้ว่าย่านความถี่มาตรฐาน มีดังนี้

1. ย่านความถี่ 2.4 GHz มี 16 ช่องสัญญาณ อัตรารับส่งข้อมูล 250 Kbps
2. ย่านความถี่ 915 GHz มี 10 ช่องสัญญาณ อัตรารับส่งข้อมูล 40 Kbps
3. ย่านความถี่ 868 GHz มี 1 ช่องสัญญาณ อัตรารับส่งข้อมูล 20 Kbps

โดยมากมักสับสน ระหว่าง X-Bee กับ Wifi โดยผู้ที่เริ่มศึกษาจะมีคำถามว่า X-Bee กับ Wifi (หรือ โมดูล 2.4 GHz) นั้น ย่านความถี่เหมือนกัน จะสามารถสื่อสารกันได้หรือไม่ คำตอบคือ สื่อสารกันไม่ได้ เพราะทางกายภาพ ถึงแม้จะเป็นย่านความถี่เดียวกัน แต่ Protocol ที่ใช้สื่อสารกันนั้นไม่เหมือนกัน

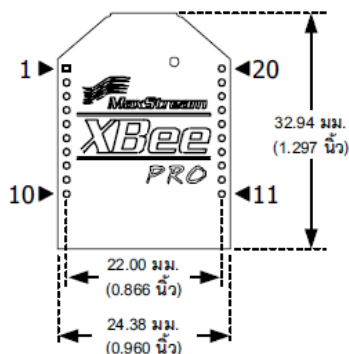
X-Bee นำ Physical Layer และ MAC Layer ของ IEEE 802.15.4 ซึ่งเป็นมาตรฐานการกำหนดการสื่อสารไร้สายแบบ WPAN (Wireless Personal Area Network) มาทำงานใน Layer ที่ต่ำกว่า (2 Layer ล่างสุด) เช่น เรื่องของระดับกำลังสัญญาณ , Link Quality , Access control , Security ฯลฯ แต่ใน Layer ถัดไปจะเป็นรูปแบบของ X-Bee



รูปที่ 2.3 X-Bee Stack

จากที่กล่าวมา X-Bee จะสามารถสร้างเป็นเครือข่ายได้เพราะอิงมาตรฐานตาม IEEE 802.15.4 และมีการจัดการในแบบของ X-Bee ใน Layer ถัดไป ทั้งนี้ IEEE 802.15.4 แบ่งชนิดอุปกรณ์ในเครือข่ายออกเป็น 2 ประเภท คือ FFD (Full Function Device) ซึ่งหมายถึง อุปกรณ์ ที่สามารถทำงานได้ทุกอย่างในเครือข่าย และ RFD (Reduce Function Device) ซึ่งหมายถึง อุปกรณ์ที่ถูกลดความสามารถการทำงานในเครือข่าย

### 2.2.2 การจัดขาการทำงานของ X-Bee



รูปที่ 2.4 ขาการทำงานของ X-Bee

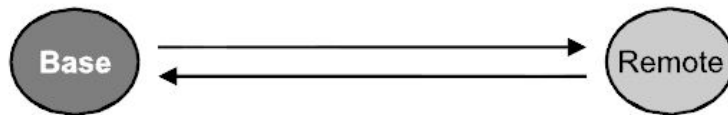
| ขาที่ | ข้อขา/การทำงาน                                                                                  |
|-------|-------------------------------------------------------------------------------------------------|
| 1     | Vcc : ขาต่อไฟเลี้ยง +3.3 โวลต์                                                                  |
| 2     | DOUT : ขาเอาต์พุตส่งข้อมูลอนุกรม                                                                |
| 3     | DIN : ขาอินพุตรับข้อมูลอนุกรม                                                                   |
| 4     | DO8 : ขาเอาต์พุตดิจิตอล ช่อง 8                                                                  |
| 5     | RESET : ขารีเซ็ตหลัก (แอกทีฟ “0”)                                                               |
| 6     | PWM0/RSSI : ขาอินพุต PWM ช่อง 0 และขาเอาต์พุตแสดงแรงของการรับสัญญาณ                             |
| 7     | PWM1 : ขาเอาต์พุต PWM ช่อง 1                                                                    |
| 8     | ไม่ใช้งาน                                                                                       |
| 9     | DTR/SLEEP_RQ/DI8 : ขาอินพุตรับสัญญาณให้หยุดทำงานเข้าสู่โหมดSleep หรือเป็นขาอินพุตดิจิตอล ช่อง 8 |

|    |                                                                                                                                                                    |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 10 | GND : ขาต่อกราวด์                                                                                                                                                  |
| 11 | AD4/DIO4 : ขาอินพุตอะนาล็อก 4 หรือ ขาอินพุตเอาต์พุตดิจิทัล 4                                                                                                       |
| 12 | CTS/DIO7 : ขาอินพุตรับสัญญาณแจ้งการส่งข้อมูลจาก โฮสต์ (Clear-to-Send)<br>ใช้ในการควบคุมจังหวะการรับส่งข้อมูล หรือ เป็นขาอินพุตเอาต์พุตดิจิทัล 7                    |
| 13 | ON/SLEEP : ขาแสดงสถานะการทำงาน<br>“1” : อยู่ในโหมดทำงานปกติ<br>“0” : อยู่ในโหมดsleep                                                                               |
| 14 | VREF : ขาต่อแรงดันอ้างอิงสำหรับ โมดูลแปลงสัญญาณอนาล็อกเป็นดิจิทัล<br>ภายใน X-Bee-PRO                                                                               |
| 15 | Associated/AD5/DIO5 : ขาแสดงสถานะการเชื่อมต่อ หรือ ขาอินพุตอะนาล็อก 5<br>หรือ ขาอินพุตเอาต์พุตดิจิทัล 5                                                            |
| 16 | RTS/AD6/DIO6 : ขาเอาต์พุตแจ้งความพร้อมในการส่งข้อมูล (Ready-To-Send)<br>ใช้ควบคุมจังหวะการรับส่งข้อมูล หรือเป็นขาอินพุตอะนาล็อก หรือ เป็นขาอินพุตเอาต์พุตดิจิทัล 6 |
| 17 | AD3/DIO3 : ขาอินพุตอะนาล็อก 3 หรือ ขาอินพุตเอาต์พุตดิจิทัล 3                                                                                                       |
| 18 | AD2/DIO2 : ขาอินพุตอะนาล็อก 2 หรือ ขาอินพุตเอาต์พุตดิจิทัล 2                                                                                                       |
| 19 | AD1/DIO1 : ขาอินพุตอะนาล็อก 1 หรือ ขาอินพุตเอาต์พุตดิจิทัล 1                                                                                                       |
| 20 | AD0/DIO0 : ขาอินพุตอะนาล็อก 0 หรือ ขาอินพุตเอาต์พุตดิจิทัล 0                                                                                                       |

ตารางที่ 2.1 การจัดขาการทำงานของ X-Bee

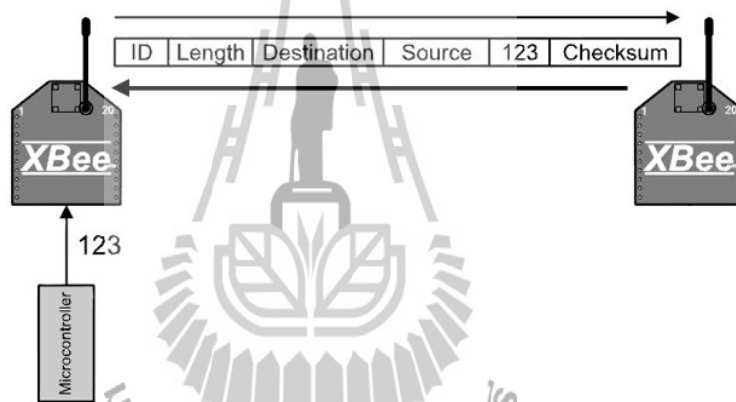
### 2.2.3 การออกแบบรูปแบบการสื่อสารของโมดูล X-Bee

ในโครงงานนี้โมดูล X-Bee ใช้รูปแบบการสื่อสารแบบ Point-to-Point Network



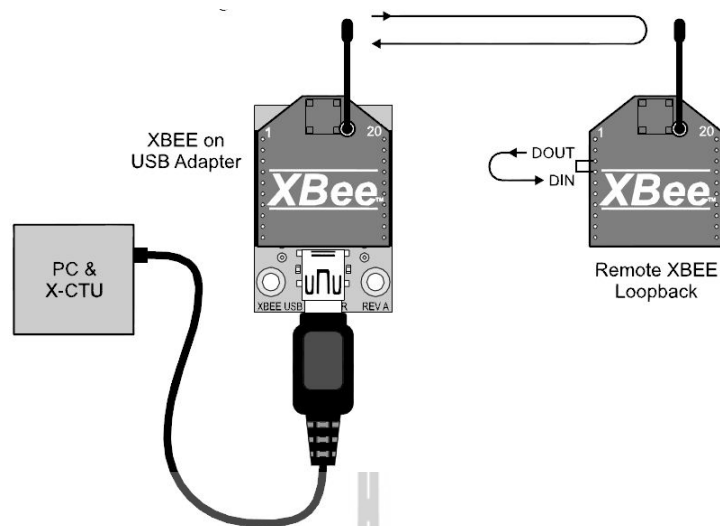
รูปที่ 2.5 การสื่อสารระหว่าง 2 โหนด เป็นแบบ Point-to-Point

ลักษณะการส่งข้อมูลของ โมดูล X-Bee ที่ใช้ เป็นรูปแบบโหมด API มีการส่งข้อมูลเป็นเฟรมข้อมูล



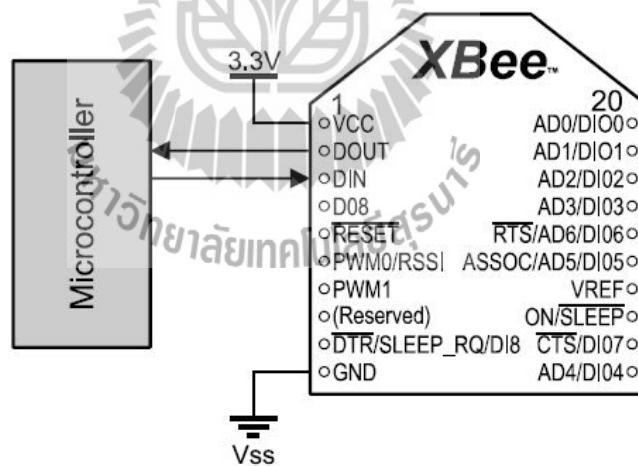
รูปที่ 2.6 การส่งข้อมูลแบบสองทิศทาง

ทดสอบการส่งข้อมูลของโมดูล X-Bee ผ่านทางเครื่องคอมพิวเตอร์โดยใช้โปรแกรม X-CTU ในการตั้งค่าพารามิเตอร์ต่างๆของอุปกรณ์โดยขั้นตอนการทดสอบได้แสดงไว้ในบทที่ 4 และภาคผนวก ข.



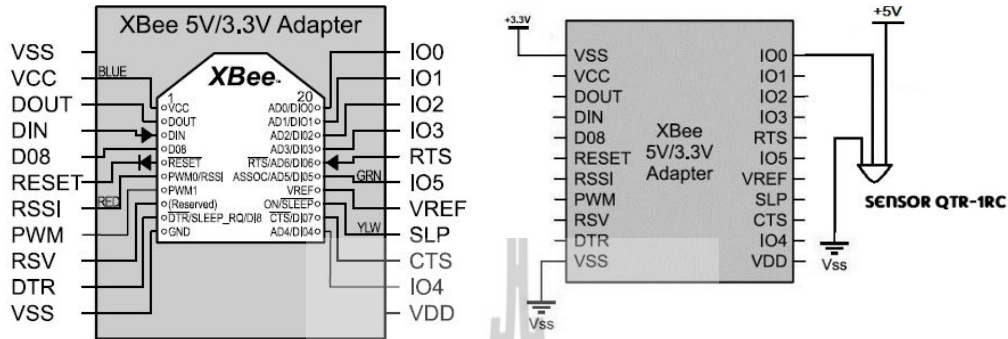
รูปที่ 2.7 ทดสอบการส่งกลับของโมดูล X-Bee

การเชื่อมต่อโมดูล X-Bee ด้านตัวที่เป็นภาคส่ง (Coordinator) เข้ากับไมโครคอนโทรลเลอร์



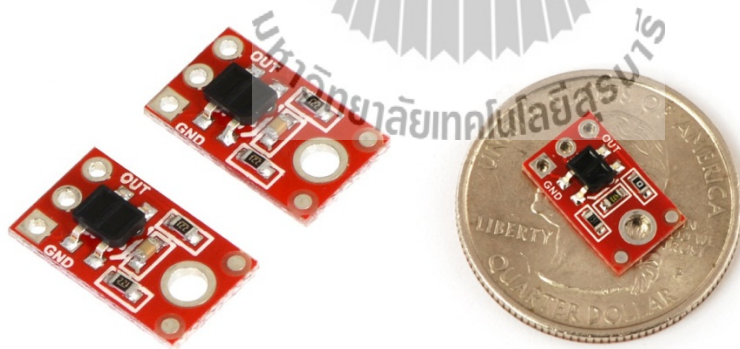
รูปที่ 2.8 การเชื่อมต่อไมโครคอนโทรลเลอร์ทั่วไปเข้ากับ โมดูล X-Bee

การเชื่อมต่อโมดูล X-Bee ด้านตัวที่เป็นภาครับ (Router) เลือกขาอินพุตเอาต์พุตดิจิทัล เพื่อต่อเข้ากับเซนเซอร์จากรูปที่ 2.9 เลือก ขาอินพุตเอาต์พุตดิจิทัล DIO0 รับค่าการทำงานของ เซนเซอร์



รูปที่ 2.9 การเชื่อมต่อ โมดูล X-Bee เข้ากับเซนเซอร์

## 2.3 เซนเซอร์ตรวจจับการสะท้อน (QTR-1RC Reflectance Sensor)



รูปที่ 2.10 QTR-1RC Reflectance Sensor

QTR-1RC เป็น เซนเซอร์ ตรวจจับการสะท้อนกลับ ประกอบด้วย IR LED 1 ตัว และ Phototransistor 1 คู่ บนโมดูลขนาดเล็ก ๆ แค่ 0.5 นิ้ว x 0.3 นิ้ว โดยเอาต์พุตนั้นถูกออกแบบมาให้ วัดได้แบบ digital I/O บนตัวไมโครคอนโทรลเลอร์สามารถอ่านค่าการสะท้อนของ IR จากเวลา ในการปล่อยของตัวเก็บประจุได้ เวลาที่สั้นที่สุดของการปล่อยนั้นเป็นตัวบ่งบอกถึงการสะท้อน

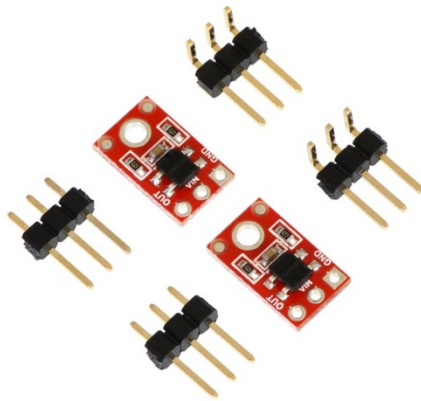
ที่ดี การจ่ายกระแสให้กับ LED นั้นตั้งค่าไว้ที่ประมาณ 20-25 มิลลิแอมป์ เมื่อจ่ายแรงดันไฟฟ้าที่ 5 โวลต์ ซึ่งกระแสไฟฟ้าที่มันต้องการนั้นบางครั้งยังสามารถรับมาได้จาก I/O lines บางตัว ของ ไมโครคอนโทรลเลอร์ ได้อีกด้วยด้วยขนาดที่เล็ก ๆ ของเซนเซอร์ตัวนี้ ทำให้มันสามารถนำไปใช้งานได้หลากหลายรูปแบบ อย่างเช่น line sensing และ Edge detection เป็นต้น นอกจากนี้ยังมีการนำโมดูลนี้ไปต่อกัน 8 โมดูลซึ่งสามารถดูได้จากตัว QTR-8RC (EPLL022)

### 2.3.1 คุณสมบัติ

- ขนาด : 0.3" x 0.5" x 0.1"
- แรงดันไฟฟ้า : 5.0 โวลต์
- Supply current: 17 มิลลิแอมป์
- Output format: digital I/O-compatible signal that can be read as a timed high pulse
- ระยะการตรวจจับที่ดีที่สุด : 0.125" (3 mm)
- ระยะการตรวจจับที่มากที่สุด: 0.375" (9.5 mm)

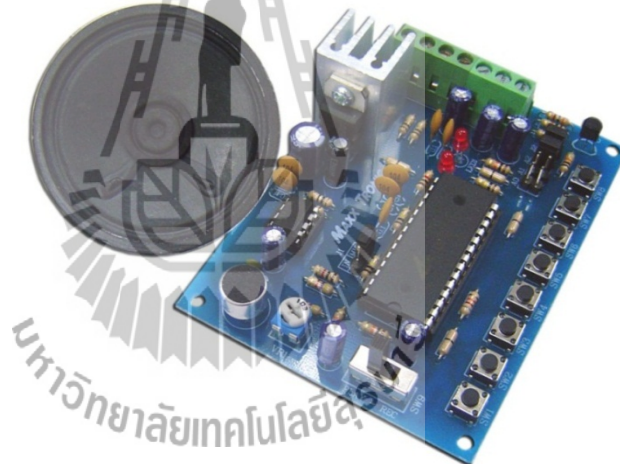
### 2.3.2 การนำไปใช้งาน

- เราจะแบ่งสถานะทางดิจิตอล ออกเป็นสองสถานะ คือ High , Low หรือ ON – OFF บางที่จะเรียกสัญญาณทาง ดิจิตอลว่า “0” กับ “1”
- ในทางกายภาพแล้ว เราจะแบ่งสถานะทางดิจิตอล ด้วยแรงดันไฟ
- สำหรับสถานะแบบใช้แรงดัน คือ “HIGH” คือ ไฟระดับ 5 โวลต์ และ “LOW” คือ ไฟระดับ 0 โวลต์
- เมื่อมีสิ่งของมาผ่านเซนเซอร์ เซนเซอร์จะเปลี่ยนสถานะจาก 1 เป็น 0



รูปที่ 2.11 ขา QTR-1RC Reflectance Sensor

## 2.4 วงจรบันทึกเสียงดิจิทัล รุ่น FK941 (DIGITAL VOICE)



รูปที่ 2.12 อุปกรณ์ บันทึกเสียงดิจิทัล รุ่น FK941

วงจรบันทึกเสียงชุดนี้ เป็นวงจรบันทึกเสียงแบบดิจิทัลซึ่งบันทึกเสียงเข้าไปเก็บไว้ในตัวไอซี โดยวงจรนี้แม้ไม่มีไฟเลี้ยงก็จะยังเก็บเสียงไว้ได้ ประมาณ 100 ปี ที่เดียวสามารถบันทึกซ้ำได้ถึงหนึ่งล้านครั้งวงจรนี้เหมาะสำหรับนำไปเป็นเครื่องประชาสัมพันธ์สินค้าตามสถานที่ขายสินค้าต่างๆ เช่น สถานที่ขายน้ำ, ขายอาหาร เป็นต้น เป็นการเพิ่มความน่าสนใจให้กับร้านค้าได้เป็นอย่างดี

#### 2.4.1 ข้อมูลทางด้านเทคนิค

- ใช้แหล่งจ่ายไฟขนาด 12 โวลต์ดีซี
- ใช้กระแสสูงสุด 27 มิลลิแอมป์ (ขณะสแตนด์บาย), 600 มิลลิแอมป์ (ขณะทำงาน)
- สามารถบันทึกได้ตั้งแต่ 20-60 วินาที
- ขนาดแผ่นวงจรพิมพ์ 3.12 x 2.54 นิ้ว

#### 2.4.2 การทำงานของวงจร

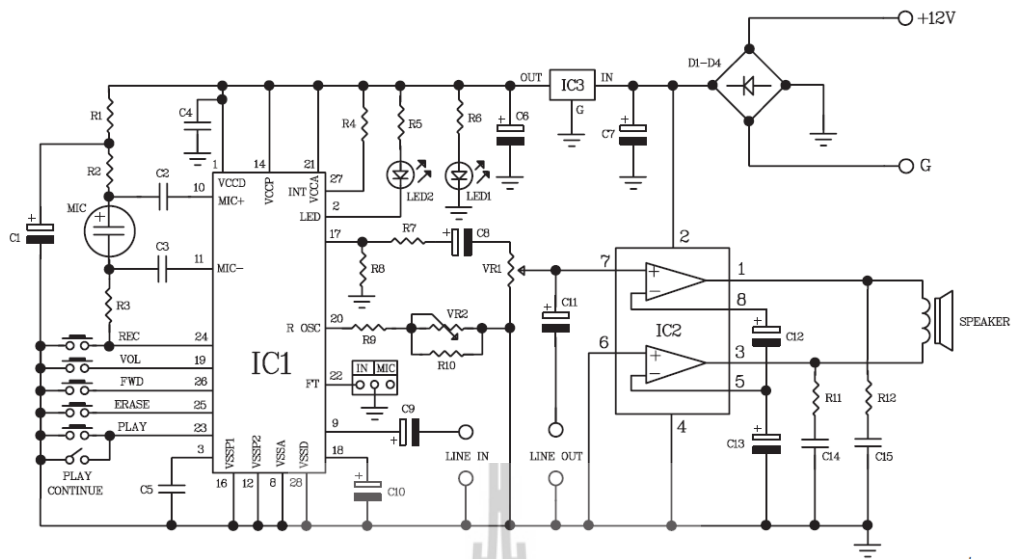
วงจรบันทึกเสียงชุดนี้จะแสดงอยู่ในรูปที่ 1 สามารถแบ่งการทำงานได้ 2 ส่วนใหญ่ๆ ด้วยกัน คือ ส่วนแรกบันทึกเสียงและเล่นกลับ ส่วนที่สอง ส่วนขยายเสียง 2 วัตต์

การทำงานของวงจรจะเริ่มจากส่วนบันทึกเสียงและเล่นกลับ ซึ่งสามารถอธิบายการทำงาน ได้เป็น 2 ส่วน คือ ส่วนบันทึกเสียงและส่วนเล่นกลับ การทำงานของส่วนบันทึกเสียงจะเริ่มจาก การจิ้ม J1 เพื่อเลือกแหล่งกำเนิดเสียงที่ต้องการจะบันทึก ตามรูปที่ 2 จากนั้นทำการกดสวิทช์ REC ค้างเอาไว้ IC1 จะเริ่มบันทึกเสียง ในขณะที่ทำการบันทึกเสียง ถ้าเราทำการปล่อยสวิทช์ REC หรือ IC1 ทำการบันทึกเสียงจนเต็ม IC1 จะทำการหยุดบันทึกเสียงทันที ในกรณีที่บันทึกเสียงไม่เต็ม เราก็สามารถบันทึกเสียงต่อไปได้ การทำงานของส่วนเล่นกลับ จะเริ่มจากเมื่อเราทำการกดสวิทช์ PLAY ตัว IC1 ก็จะทำการเล่น กลับทันทีสัญญาณเสียงที่เราบันทึกเอาไว้ก็จะไปเข้าส่วนขยายเสียง 2 วัตต์ ในกรณีที่ต้องการให้ เล่นกลับแบบต่อเนื่อง เราสามารถใช้สวิทช์เลื่อน OFF-PLAY ก็ได้ โดยถ้าเราทำการเลื่อนสวิทช์ไปที่ PLAY ตัว IC1 ก็จะทำการเล่นไปเรื่อยๆ โดยไม่มีการหยุดจนกว่าเราจะเลื่อนสวิทช์ OFF-PLAY มาที่ OFF ตัว IC1 จึงจะหยุดเล่นกลับ ในส่วนขยายเสียง 2 วัตต์ เมื่อ IC1 ทำการเล่นกลับ มันจะทำการส่งสัญญาณเสียงออกทางขา 17 ไปเข้า IC2 เพื่อทำการขยายเสียงให้ดังมากยิ่งขึ้นสำหรับ VR2 จะเป็นตัวปรับระยะเวลาในการบันทึก ในส่วนของสวิทช์ ERASE จะใช้สำหรับลบข้อความที่เราบันทึกเอาไว้ โดยถ้ากดสวิทช์ครั้งเดียวก็จะลบข้อความในตำแหน่งที่ข้อความเล่นอยู่ แต่ถ้ากดสวิทช์ ค้างเอาไว้ก็จะเป็นการลบข้อความทั้งหมด

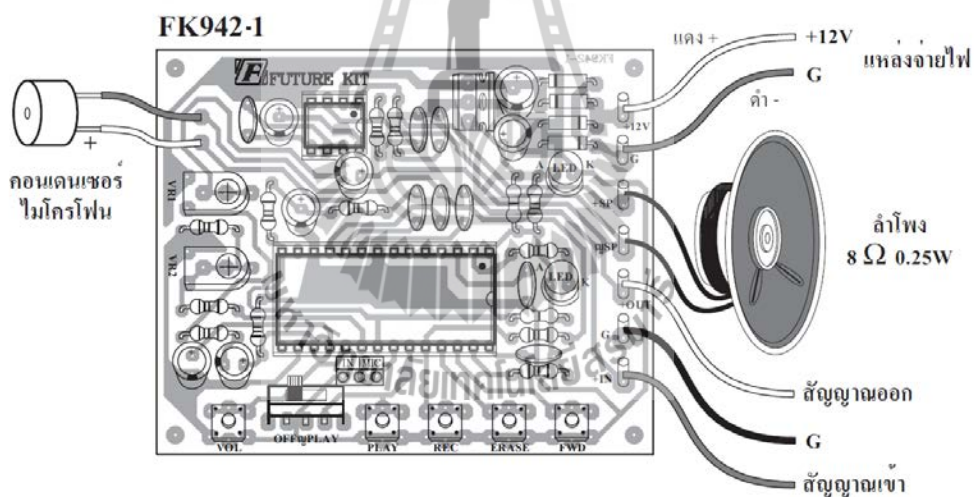
#### 2.4.3 การนำไปใช้งาน

วงจรนี้สามารถดัดแปลงเพื่อนำไปใช้ในฉายต่างๆ ได้ เช่น เครื่องฝากข้อความ หรือเครื่อง โทรแจ้งเหตุต่างๆ เป็นต้น สำหรับสวิทช์ต่างๆ จะมีการทำงาน ดังนี้

- สวิตช์ PLAY กดครั้งเดียว เมื่อต้องการเล่นกลับ LED2 จะกระพริบ ในขณะที่เล่นกลับ เมื่อหมดข้อความที่บันทึกเอาไว้วงจรจะหยุดเล่นกลับเองโดยอัตโนมัติหรือถ้าต้องการหยุดเอง ก็ให้ทำการกดสวิตช์ PLAY อีกครั้ง (ในกรณีที่ข้อความว่าง LED1 จะกระพริบ 2 ครั้งแล้วดับไป) แต่ถ้าเรากดสวิตช์ PLAY ค้างเอาไว้ วงจรจะเล่นข้อความจนหมดแล้วเริ่มใหม่ ไปเรื่อยๆจน กระทั่งปล่อยสวิตช์
- สวิตช์ REC กดค้างไว้เมื่อต้องการบันทึกเสียง โดย LED2 จะติดค้าง ในขณะที่บันทึกเสียง ในกรณีที่เราไม่ต้องการบันทึกจนหมด ก็ให้ทำการปล่อยสวิตช์ REC ได้เลยทันที แต่เมื่อเราต้องการบันทึกข้อความต่อ ก็ให้กดสวิตช์ REC ค้างไว้อีกครั้ง ข้อความต่อไปก็จะกลายเป็นข้อความที่ 2 และจะเป็นอย่างนี้เรื่อยๆ จนกว่าข้อความจะเต็ม (ถ้าบันทึกจนข้อความเต็มแล้ว เมื่อกดสวิตช์ REC ตัว LED2 จะกระพริบ 2 ครั้ง ตัว IC1 ก็จะไม่ทำการบันทึก)
- สวิตช์ ERASE ใช้ในกรณีที่ต้องการลบข้อความที่บันทึกเอาไว้ โดยถ้ากดเพียงครั้งเดียว IC1 จะทำการลบข้อความในตำแหน่งปัจจุบันที่ไอซีเล่นกลับ LED2 จะกระพริบ 2 ครั้ง แต่ถ้าเรากด สวิตช์ค้างเอาไว้ LED2 จะกระพริบ 2 ครั้งก่อน แล้วจึงกระพริบ 7 ครั้ง นั่นแสดงว่าไอซีจะลบข้อความทั้งหมด
- สวิตช์ FWD กดครั้งเดียว เมื่อต้องการเลื่อนข้อความที่ต้องการเล่นกลับ ไปยังข้อความต่อไป (ใช้ในกรณีที่บันทึกหลายข้อความ) ถ้าข้อความที่เลื่อนเป็นข้อความถัดไป LED2 จะกระพริบเพียง 1 ครั้ง แต่ถ้าเป็นข้อความสุดท้ายแล้วย้อนกลับไป ข้อความแรก LED2 จะกระพริบ 2 ครั้ง
- สวิตช์เลื่อน OFF-PLAY ใช้ในกรณีที่ต้องการให้วงจรเล่นข้อความแบบต่อเนื่อง โดยอัตโนมัติ โดยถ้าเลื่อนสวิตช์ไปที่ PLAY ตัว IC1 จะทำการเล่นข้อความต่อไปเรื่อยๆ แต่ถ้าเราเลื่อนสวิตช์ไปที่ OFF ตัว IC1 จะรอจนหมดข้อความแล้วจึงหยุดเล่น
- สวิตช์ VOL ใช้สำหรับลดและเร่งระดับเสียงในขณะที่เล่นกลับ
- VR1 10K ใช้สำหรับลดและเร่งระดับเสียงในขณะที่เล่นกลับ โดยการหมุน
- VR2 500K ใช้สำหรับปรับระยะเวลาในการบันทึกเสียง โดยถ้าปรับไปทางซ้ายมือสุดจะบันทึกได้ประมาณ 20 วินาที แต่ถ้าปรับไปทางขวามือสุดจะบันทึกได้ ประมาณ 60 วินาที ในขณะที่เล่นกลับถ้าเราทำการปรับ VR2 เสียงที่กำลังเล่นกลับอยู่จะมีความเร็วเพิ่มหรือลดลงตาม การปรับของเรา ซึ่งระยะเวลาในการบันทึกเสียงยิ่งนานคุณภาพเสียงก็จะค่อยๆลดลงไปด้วย

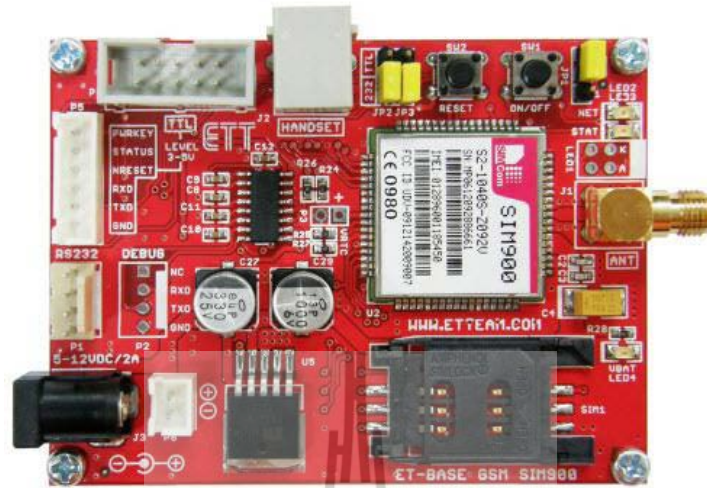


รูปที่ 2.13 วงจรบันทึกเสียงดิจิทัล 20-60 วินาที พร้อมขยายเสียง 2 วัตต์



รูปที่ 2.14 การต่อใช้งาน

## 2.5 บอร์ดไมโครคอนโทรลเลอร์ รุ่น ET-BASE GSM SIM900



รูปที่ 2.15 บอร์ด ET-BASE GSM SIM900

### 2.5.1 คุณสมบัติของบอร์ด ET-BASE GSM SIM900

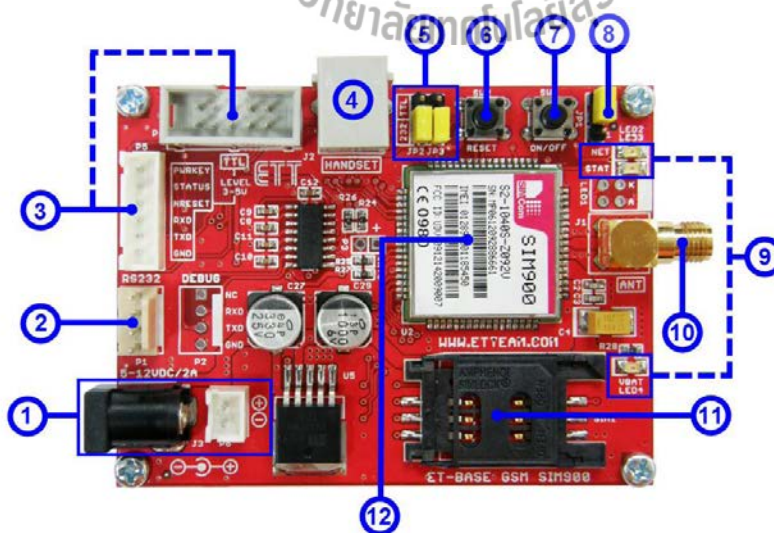
- มีสวิตช์แบบ Push-Button สำหรับใช้สั่ง เปิด-ปิด การทำงานของโมดูลภายในบอร์ด
- มีสวิตช์แบบ Push-Button สำหรับใช้สั่ง RESET การทำงานของโมดูลภายในบอร์ด
- มี Socket SIM รองรับ SIM Card พร้อมวงจร ESD ป้องกัน SIM เสียหาย
- มีวงจร Regulate แยกอิสระ จำนวน 2 ชุด สามารถใช้กับแหล่งจ่ายตั้งแต่ 5-12 โวลต์ได้ซึ่งสามารถจ่ายกระแสให้กับโมดูล SIM900 และอุปกรณ์เชื่อมต่อต่างๆ ได้อย่างเพียงพอ
- มีวงจร Regulate ขนาด 4.2 โวลต์ / 3 แอมแปร์ สำหรับจ่ายให้กับโมดูล SIM900 ได้อย่างเพียงพอสามารถใช้กับ SIM ของระบบ GSM900MHz แบบ 2-Watt ได้อย่างไม่เกิดปัญหา
- มีวงจร Regulate ขนาด 2.8 โวลต์ / 150 มิลลิแอมป์ สำหรับจ่ายให้กับวงจรแปลงระดับสัญญาณalog
- มีวงจร Line Driver สำหรับแปลงระดับสัญญาณalogจากโมดูล SIM900 ให้เป็น RS232 (1200 bps - 115200 bps) สำหรับพอร์ตที่ใช้ในการสื่อสารสำหรับส่งงานโมดูล
- มีวงจรแปลงระดับสัญญาณalog TTL ระดับแรงดัน 3 โวลต์ ถึง 5 โวลต์ ทำให้สามารถเชื่อมต่อกับไมโครคอนโทรลเลอร์ได้โดยตรงโดยไม่ต้องผ่านวงจร Line Driver RS232

- มี LED แสดงสถานะพร้อมในบอร์ด สำหรับแสดงสถานะของแหล่งจ่ายไฟ สถานะพร้อมทำงานของโมดูล สถานะในการเชื่อมต่อกับ Network และ สถานะ Power-ON/Power-OFF ของโมดูล
- มีขั้วสำหรับเชื่อมต่อกับ Handset (ชุดปากพูด และหูฟัง ของโทรศัพท์บ้าน) โดยใช้ขั้วต่อแบบ RJ11 มาตรฐาน พร้อมวงจร Voice Filter สามารถนำชุด Handset ของโทรศัพท์บ้าน ต่อเข้ากับบอร์ดทาง ขั้วต่อแบบ RJ11 สำหรับใช้พูดคุย โทรออก และ รับสายได้โดยสะดวก

### 2.5.2 คุณสมบัติเบื้องต้นของโมดูล SIM900

- รองรับความถี่ GSM/GPRS 850/900/1800/1900MHz
- รองรับ GPRS Multi-Slot Class10 และ GPRS Mobile Station Class B
- รองรับมาตรฐานคำสั่ง AT Command
- รองรับ SIM Applications Toolkit
- ทำงานที่ย่านแรงดัน 3.2 โวลต์ ถึง 4.8 โวลต์
- ใช้ได้กับ SIM card 1.8โวลต์ และ 3 โวลต์
- มีวงจร Analog Audio (MIC & Speaker)

### 2.5.3 ส่วนประกอบของบอร์ด ET-BASE GSM SIM900



รูปที่ 2.16 ส่วนประกอบ ET-BASE GSM SIM900

- หมายเลข 1 เป็นขั้วต่อไฟเลี้ยงเข้าบอร์ดโดยมีให้เลือกต่อ 2 แบบ คือ แบบ DC JACK ซึ่ง ขั้วด้านนอกเป็นไฟบวก ด้านในเป็นลบ และ ขั้วต่อแบบ JST โดยแรงดันไฟเลี้ยงที่จ่ายให้บอร์ดสามารถใช้ได้ตั้งแต่ 5-12 โวลต์ดีซี กระแสอย่างน้อย 2 แอมป์
- หมายเลข 2 เป็นขั้วต่อสัญญาณ RS232 แบบ 4 PINS (มาตรฐานอีทีที) สำหรับเชื่อมต่อกับอุปกรณ์ที่ใช้การรับส่งข้อมูลด้วย RS232 เช่น คอมพิวเตอร์ หรือ บอร์ดไมโครคอนโทรลเลอร์ต่างๆ ที่ต่อผ่านวงจร Line Driver RS232
- หมายเลข 3 เป็นขั้วต่อสัญญาณระดับ TTL 3-5 โวลต์ สำหรับเชื่อมต่อกับไมโครคอนโทรลเลอร์ได้โดยตรงโดยไม่ต้องผ่านวงจร Line Driver RS232

| ชื่อขาสัญญาณ | ทิศทาง | รายละเอียด                                                                                                                                        |
|--------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| PWRKEY       | INPUT  | ใช้ควบคุมการเปิดปิดโมดูล SIM900 โดยจะทำงานที่ลอจิก 1                                                                                              |
| STATUS       | OUTPUT | ใช้บอกสถานะว่าโมดูล SIM900 เปิดการทำงานอยู่ ถ้าเป็นลอจิก 1 แสดงว่าโมดูลเปิดอยู่ ถ้าเป็นลอจิก 0 แสดงว่าแสดงว่าโมดูลปิดอยู่หรืออยู่สภาวะ power down |
| NRESET       | INPUT  | ใช้สำหรับรีเซ็ตการทำงานของโมดูล SIM900 โดยจะทำงานที่ลอจิก 1                                                                                       |
| RXD          | INPUT  | Receive data                                                                                                                                      |
| TXD          | OUTPUT | Transmit data                                                                                                                                     |
| RTS          | INPUT  | Request to send                                                                                                                                   |
| CTS          | OUTPUT | Clear to send                                                                                                                                     |
| DCD          | OUTPUT | Data carrier detect                                                                                                                               |
| DTR          | INPUT  | Data terminal ready                                                                                                                               |
| NC           | -      | ขาวางไม่ได้ใช้งาน                                                                                                                                 |
| GND          |        | กราวด์                                                                                                                                            |

ตารางที่ 2.2 แสดงการทำงานของแต่ละขา

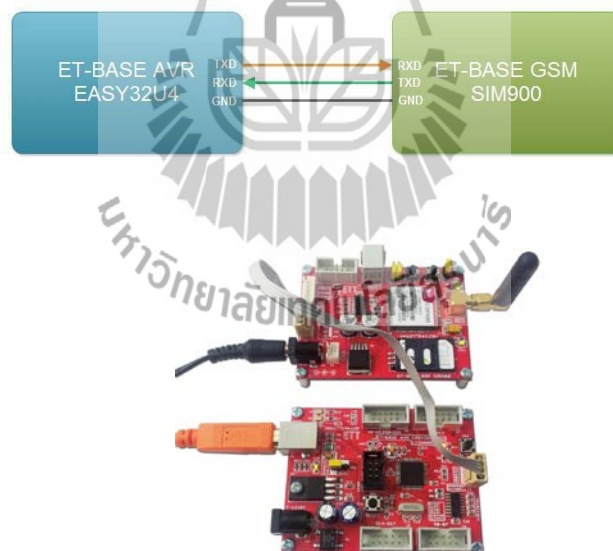
- หมายเลข 4 เป็น ขั้วต่อ RJ11 สำหรับใช้เชื่อมต่อกับชุด Handset ในกรณีที่ต้องการใช้งานโมดูล SIM900 เพื่อโทรออกและรับสาย โดยสามารถเชื่อมต่อกับ Handset มาตรฐานได้ทั่วไป
- หมายเลข 5 เป็นจัมเปอร์เลือกว่าจะต่อขาสัญญาณ RXD, TXD ของโมดูลผ่านวงจร Line Driver RS232 หรือไม่ ถ้าผู้ใช้ต้องการเชื่อมต่อผ่านขั้ว RS232 ก็ให้เลือกจัมเปอร์ JP2 และ JP3 ไปที่ตำแหน่ง 232 แต่ถ้าต้องการเชื่อมต่อทางขั้ว TTL P4, P5 ก็ให้เลือกจัมเปอร์ JP2 และ JP3 ไปที่ตำแหน่ง TTL
- หมายเลข 6 เป็น Switch Push-Button สำหรับใช้รีเซ็ตการทำงานของตัวโมดูล
- หมายเลข 7 เป็น Switch Push-Button สำหรับใช้ Power-ON และ Power-OFF ตัวโมดูล
- หมายเลข 8 เป็นจัมเปอร์สำหรับเปิดการทำงานของโมดูล SIM900 แบบอัตโนมัติทันทีเมื่อจ่ายไฟเลี้ยงเข้าบอร์ดโดยให้เลือกไปที่ตำแหน่ง AT แต่ถ้าต้องการควบคุมการเปิดปิดโดย สวิตช์ ON/OFF หรือทางขา PWRKEY ก็ให้เลือกจัมเปอร์ไปที่ตำแหน่งขา 1-2
- หมายเลข 9 เป็น LED แสดงสถานะการทำงานของบอร์ดโดยมีรายละเอียดดังนี้
  - **LED VBAT** ใช้ทำหน้าที่แสดงสถานะของแหล่งจ่ายไฟจากภายนอกที่ต่อมาให้กับบอร์ด โดย LED นี้จะติดสว่างก็ต่อเมื่อมีการจ่ายไฟให้กับบอร์ดเป็นที่เรียบร้อยแล้ว
  - **LED NET (NETLIGHT)** ใช้แสดงสถานะของโมดูล ในขณะที่ทำการเชื่อมต่อกับเครือข่ายอยู่ โดย LED ตัวนี้จะถูกควบคุมด้วยสัญญาณ NETLIGHT(PIN 52) ของโมดูล SIM900 เมื่อทำงานจะมีสถานะทางลอจิกเป็นลอจิก “1” โดยเมื่อโมดูลอยู่ในสถานะพร้อมทำงาน LED นี้จะติดกระพริบด้วยค่าความเร็วต่างๆ ซึ่งมีความหมายดังนี้
    - OFF แสดงว่าโมดูลอยู่ในสถานะของ Power OFF (ไม่ทำงาน)
    - 64mS ON / 800mS OFF แสดงว่า โมดูล SIM900 ไม่สามารถการค้นหาเครือข่ายได้
    - 64mS ON / 3000mS OFF แสดงว่าโมดูล SIM900 สามารถการค้นหาเครือข่ายได้
    - 64mS ON / 300mS OFF แสดงว่าโมดูล SIM900 อยู่ระหว่างการเชื่อมต่อกับเครือข่ายหรืออุปกรณ์อื่นๆ ด้วย GPRS อยู่

➤ **LED STAT (STATUS)** ใช้แสดงสถานะของโมดูล SIM900 ว่าทำงานอยู่หรือเปล่า ถ้า LED ติดแสดงว่าโมดูลทำงานอยู่ ถ้า LED ไม่ติดแสดงว่าโมดูลไม่ทำงาน หรืออยู่ในสภาวะ Power down mode

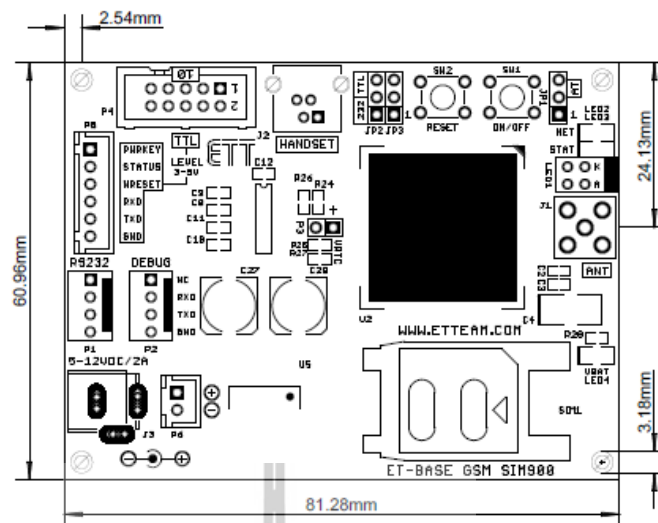
- หมายเลข 10 เป็นคอนเน็กเตอร์เสาอากาศ GSM ย่านความถี่ 850/900/1800/1900MHz
- หมายเลข 11 เป็น Socket สำหรับติดตั้ง SIM Card ให้กับโมดูล
- หมายเลข 12 โมดูล SIM900

#### 2.5.4 การเชื่อมต่อบอร์ด ET-BASE GSM SIM900 กับ บอร์ดไมโครคอนโทรลเลอร์

ตัวอย่างนี้จะขอยกตัวอย่างการเชื่อมต่อ ET-BASE GSM SIM900 ร่วมกับบอร์ด ET-BASE AVR EASY32U4 ซึ่งจะตัวอย่างนี้จะเชื่อมต่อผ่านพอร์ต RS232 โดยไดอะแกรมการเชื่อมต่อ และรูปการเชื่อมต่อดังรูป



รูปที่ 2.17 การเชื่อมต่อ ET-BASE GSM SIM900 ร่วมกับบอร์ด ET-BASE AVREASY32U4



รูปที่ 2.18 แผนผังวงจร บอร์ด ET-BASE GSM SIM900

## 2.6 บอร์ดไมโครคอนโทรลเลอร์ รุ่น ET-BASE51 AC3



รูปที่ 2.19 บอร์ด ET-BASE51 AC3 (AT89C51AC3)

ET-BASE51 AC3 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล MCS51 ขนาด 52 Pin ซึ่งเลือกใช้ไมโครคอนโทรลเลอร์ เบอร์ AT89C51AC3 ของ ATMEL เป็น MCU ประจำบอร์ด โดย MCU รุ่นนี้จะบรรจุอยู่ในตัวถังแบบ 52 Pin PLCC โดย MCU ตัวนี้จะมีจุดเด่น คือ เรื่องของความเร็วในการประมวลผล ซึ่งสามารถทำงานได้ด้วยความถี่สูงสุด 60MHz ที่ 12 Clock / 1 Machine Cycle นอกจากนี้แล้วยังมีความเพียบพร้อมด้วยอุปกรณ์พื้นฐานต่างๆที่จำเป็นต่อการใช้งาน ไม่ว่าจะเป็นหน่วยความจำสำหรับเก็บข้อมูลแบบ EEPROM ขนาด 2 KByte หรือ

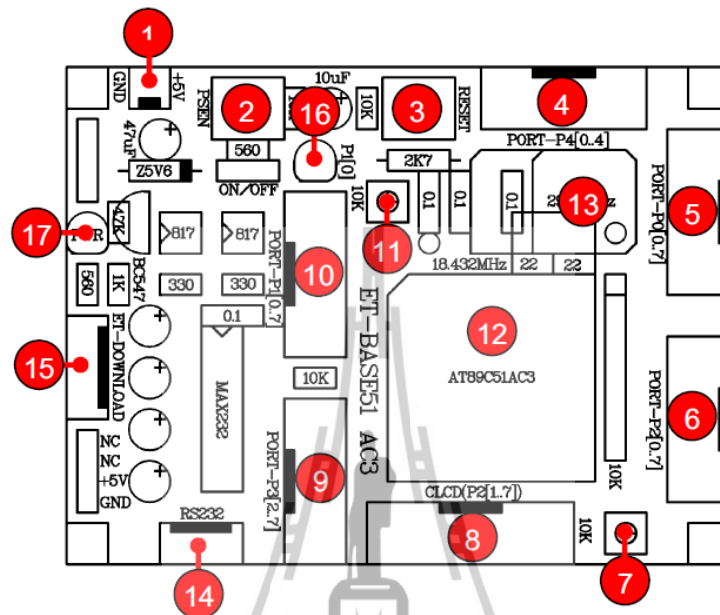
หน่วยความจำใช้งานแบบ RAM ซึ่งมีมากถึง 2304 Byte (2048+256) ส่วนในด้านของอุปกรณ์ Peripheral นั้นก็นับว่าครบถ้วนเหมาะแก่การนำไปประยุกต์ใช้งานเกี่ยวกับการควบคุมและประมวลผลต่างๆได้เป็นอย่างดี โดยจะมีทั้ง SPI, UART, Watchdog, Timer/Counter, PWM และ ADC โดยการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องขนาดของบอร์ดให้มีขนาดเล็ก เพื่อให้ง่ายต่อการนำไปประยุกต์ใช้งาน และสะดวกต่อการพัฒนาโปรแกรม

### 2.6.1 คุณสมบัติของบอร์ด

- เลือกใช้ MCU ตระกูล MCS51 เบอร์ AT89C51AC3 ของ ATMEL เป็น MCU ประจำบอร์ด โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ Oscillator Module ค่า 29.4912 MHz ซึ่งสามารถกำหนดการทำงานของ MCU ให้ทำงานในโหมดความเร็ว 2 เท่า (X2 Mode) ได้ ทำให้ MCU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 58.9824 MHz โดยคุณสมบัติเด่นๆของ MCU ได้แก่
  - มีหน่วยความจำ Flash สำหรับเขียนโปรแกรมขนาด 64Kbyte
  - มี EEPROM ขนาด 2KByte สำหรับเก็บข้อมูล และสามารถเขียนซ้ำได้กว่า 1 ล้านครั้ง
  - มีพอร์ต I/O ขนาด 8 บิต จำนวน 5 พอร์ต (P0,P1,P2,P3 และ P4(5Bit))
  - มี RAM ใช้งาน 2304 Byte (ERAM 2048 Byte + IRAM 256 Byte)
  - มีวงจรสื่อสารอนุกรม UART จำนวน 1 พอร์ต และมีวงจรสื่อสาร SPI จำนวน 1 พอร์ต
  - มีวงจร Timer/Counter ขนาด 16 บิต จำนวน 3 ชุด
  - มีวงจร ADC ขนาด 10บิต จำนวน 8 ช่อง (ใช้ Port-P1 โดยกำหนดจากโปรแกรม)
  - มีวงจร Watchdog, Power-ON Reset, Capture/Compare ,PWM
- มีขั้วต่อสัญญาณ I/O แบบ TTL แบบ Header 2x5 จำนวน 5 ชุด (P0,P1,P2,P3 และ P4)
- มีขั้วต่อ LCD แบบ Header 2x7 รองรับการเชื่อมต่อกับ LCD Character (เชื่อมต่อแบบ 4 บิต)
- มีขั้วต่อใช้งาน RS232 สำหรับใช้งาน และ ET-DOWNLOAD สำหรับ Download ผ่าน RS232
- มี LED แสดงสถานะแหล่งจ่าย Power และ Self-Test สำหรับใช้ทดสอบการทำงานของบอร์ด

- ใช้แหล่งจ่ายไฟขนาด +5 โวลต์ดีซี
- ขนาด PCB Size เล็กเพียง 8 x 6 cm

## 2.6.2 โครงสร้างบอร์ด ET-BASE51 AC3 (AT89C51AC3)



รูปที่ 2.20 โครงสร้างบอร์ด ET-BASE51 AC3 (AT89C51AC3)

- หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่ายไฟตรง +5 โวลต์ดีซี
- หมายเลข 2 เป็น Switch PSEN ใช้ร่วมกับ RESET สำหรับ Download แบบ Manual
- หมายเลข 3 เป็น Switch RESET ใช้สำหรับ Reset การทำงานของ CPU
- หมายเลข 4 เป็น Port-P4 มี ขนาด 5 Bit คือ P4.0-P4.4
- หมายเลข 5 เป็น Port-P0 มี ขนาด 8 บิต
- หมายเลข 6 เป็น Port-P2 มี ขนาด 8 บิต
- หมายเลข 7 เป็น ตัวต้านทานสำหรับปรับค่าความสว่างให้ LCD
- หมายเลข 8 เป็น Port-LCD ชนิด Character Type ใช้การเชื่อมต่อแบบ 4 บิต ผ่าน Port-P2
- หมายเลข 9 เป็น Port-P3 มีขนาด 6 บิต (P3.2-P3.7)
- หมายเลข 10 เป็น Port-P1 มีขนาด 8 บิต
- หมายเลข 11 เป็นตัวต้านทานสำหรับปรับค่าแรงดันอ้างอิงของ ADC (3 โวลต์)
- หมายเลข 12 คือ MCU เบอร์ AT89C51AC3 ซึ่งเป็น MCU ตระกูล MCS51 จาก ATMEL

- หมายเลข 13 คือ Oscillator Module ค่า 29.4912 MHz
- หมายเลข 14 คือ ขั้วต่อ RS232 สำหรับใช้งานทั่วไป และ Download แบบ Manual
- หมายเลข 15 คือ ขั้วต่อ ET-DOWNLOAD ใช้สำหรับ Download แบบ Auto
- หมายเลข 16 เป็น LED Self Test (P1.0) ใช้สำหรับทดสอบการทำงานของบอร์ด
- หมายเลข 17 เป็น LED Power ใช้แสดงสถานะของแหล่งจ่ายไฟ +5 โวลต์ดีซี

### 2.6.3 ขั้วต่อสัญญาณต่างๆ

สำหรับขั้วต่อสัญญาณของพอร์ต I/O จาก MCU นั้น จะถูกออกแบบและจัดเตรียมไว้ผ่านทางขั้วต่อแบบ IDE Header ขนาด 10Pin (2x5) จำนวน 5 ชุด คือ PORT-P0, PORT-P1, PORT P2 , PORT-P3 และ PORT-P4 ตามลำดับ โดยที่ขั้วต่อสัญญาณแต่ละชุด จะประกอบไปด้วยสัญญาณของ I/O ที่เชื่อมต่อมาจากขาสัญญาณของ MCU โดยตรงทั้งหมด โดยแต่ละพอร์ตจะมีสัญญาณพอร์ตละ 8 บิต ยกเว้น PORT-P3 และ PORT-P4 โดย PORT-P4 มีขนาด 5 บิต (P4.0-P4.4) และ PORT-P3 ซึ่งจะมีเพียง 6 บิตเท่านั้น คือ P3.2-P3.7 ส่วน P3.0 และ P3.1 จะถูกสงวนไว้ใช้งานเป็นขาสัญญาณ RXD และ TXD สำหรับรับส่งข้อมูลของ RS232 ซึ่งสัญญาณทั้ง 2 เส้น (P3.0 และ P3.1) จะถูกเชื่อมต่อผ่านวงจร Line Driver (MAX232) สำหรับแปลงระดับสัญญาณจากระดับลอจิก TTL ของ MCU ให้เป็นสัญญาณแรงดันตามมาตรฐานของ RS232 โดยสัญญาณที่ได้รับการแปลงเป็นแบบ RS232 จะถูกเชื่อมต่อไปรอไว้ที่ขั้วต่อแบบ CPA ขนาด 4 PIN (RS232) โดยการจัดเรียงสัญญาณของแต่ละชุด

## บทที่ 3

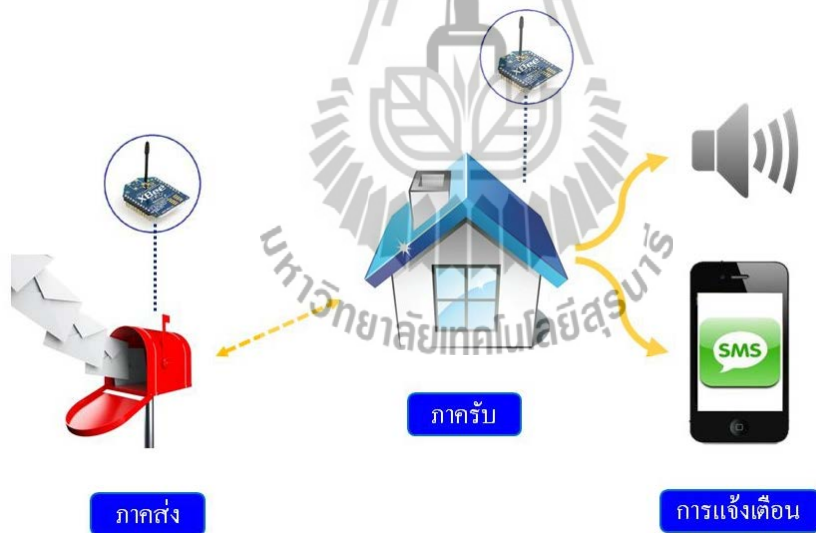
### หลักการทำงานและการออกแบบระบบของโครงการ

#### 3.1 บทนำ

ในบทนี้จะกล่าวถึงการทำงาน การใช้โปรแกรม X-CTU โปรแกรม X-Bee API Demo, โปรแกรม Keil uVision3, โปรแกรม FLIP2.4.6 , การใช้คำสั่งของบอร์ด GSM SIM900 เบื้องต้น , โมดูลFK941วงจรบันทึกเสียง, การสื่อสารแบบอนุกรมผ่านวงจร UART(Universal asynchronous receiver transmitter), การสื่อสารแบบไร้สายโดยผ่านโมดูล X-Bee

#### 3.2 การออกแบบฮาร์ดแวร์ระบบตู้ไปรษณีย์อัจฉริยะ

##### 3.2.1 หลักการทำงาน



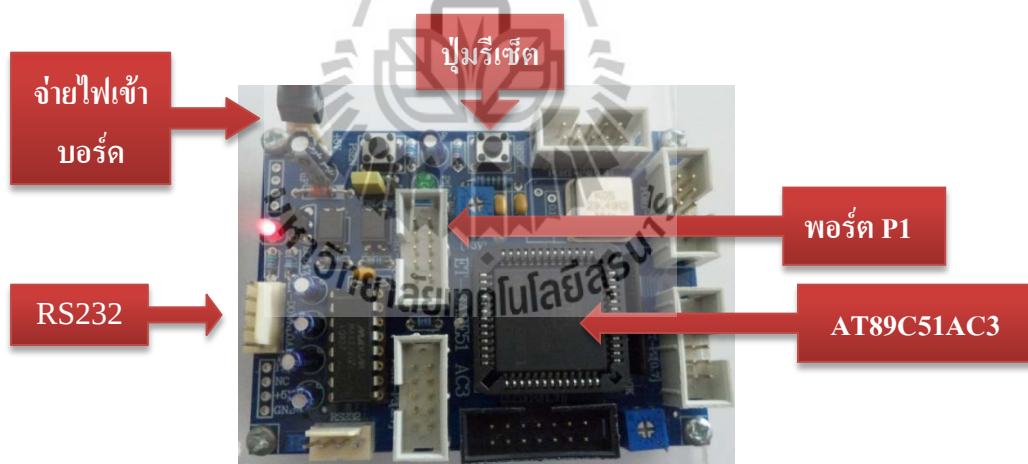
รูปที่ 3.1 แผนภาพหลักการทำงาน

### 3.2.2 อุปกรณ์ที่ใช้ในการออกแบบ

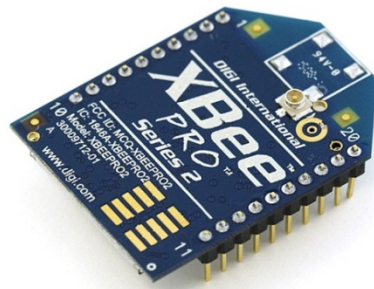
อุปกรณ์ที่ใช้ในการออกแบบการทำงานของระบบตู้ไปรษณีย์อัจฉริยะมีดังนี้

1. บอร์ดไมโครคอนโทรลเลอร์รุ่น ET-BASE51 AC3
2. โมดูล X-Bee PRO 50mW U.F.L Connection - Series 2 (ZB) 2 ตัว
3. สายอากาศแบบ U.F.L. และ Interface Cable
4. บอร์ด ET-BASE GSM SIM900
5. วงจรบันทึกเสียงดิจิทัล รุ่น FK941
6. เซนเซอร์ตรวจจับการสะท้อน QTR-1RC
7. ตู้ไปรษณีย์ที่ออกแบบขึ้น

โดยรูปของอุปกรณ์ที่ใช้ได้แสดงในรูปที่ 3.2 ถึงรูปที่ 3.8



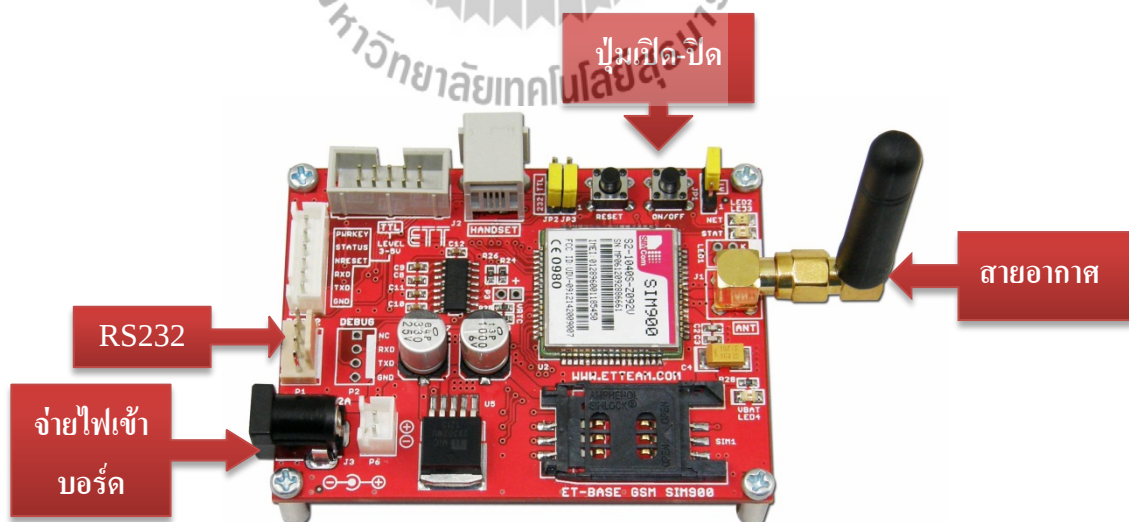
รูปที่ 3.2 บอร์ดไมโครคอนโทรลเลอร์รุ่น ET-BASE51 AC3



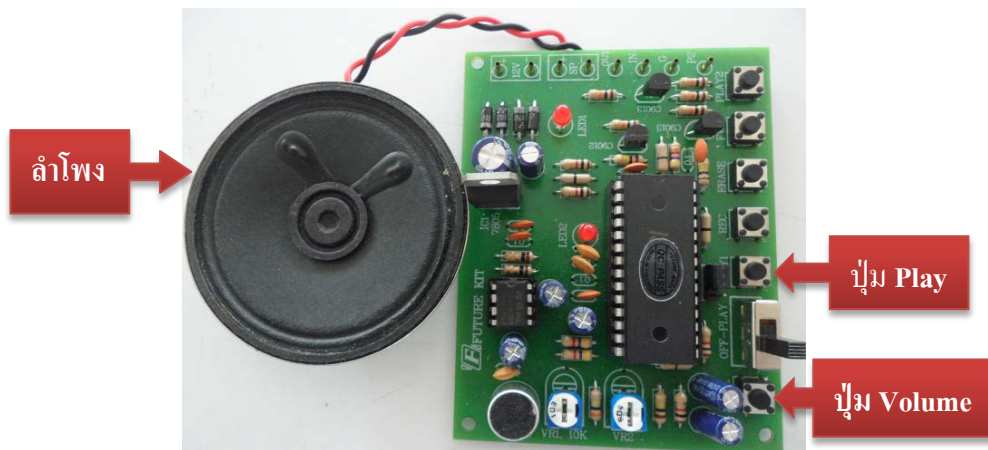
รูปที่ 3.3 โมดูล X-Bee PRO 50mW U.F.L Connection - Series 2 (ZB) (ด้านซ้ายมือ)  
และ บอร์ดเชื่อมโมดูล X-Bee (ด้านขวามือ)



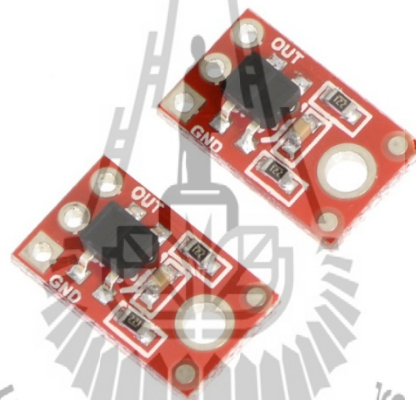
รูปที่ 3.4 สายอากาศแบบ U.F.L (ด้านซ้ายมือ) และ Interface Cable (ด้านขวามือ)



รูปที่ 3.5 บอร์ด ET-BASE GSM SIM900



รูปที่ 3.6 วงจรบันทึกเสียงดิจิทัล รุ่น FK941



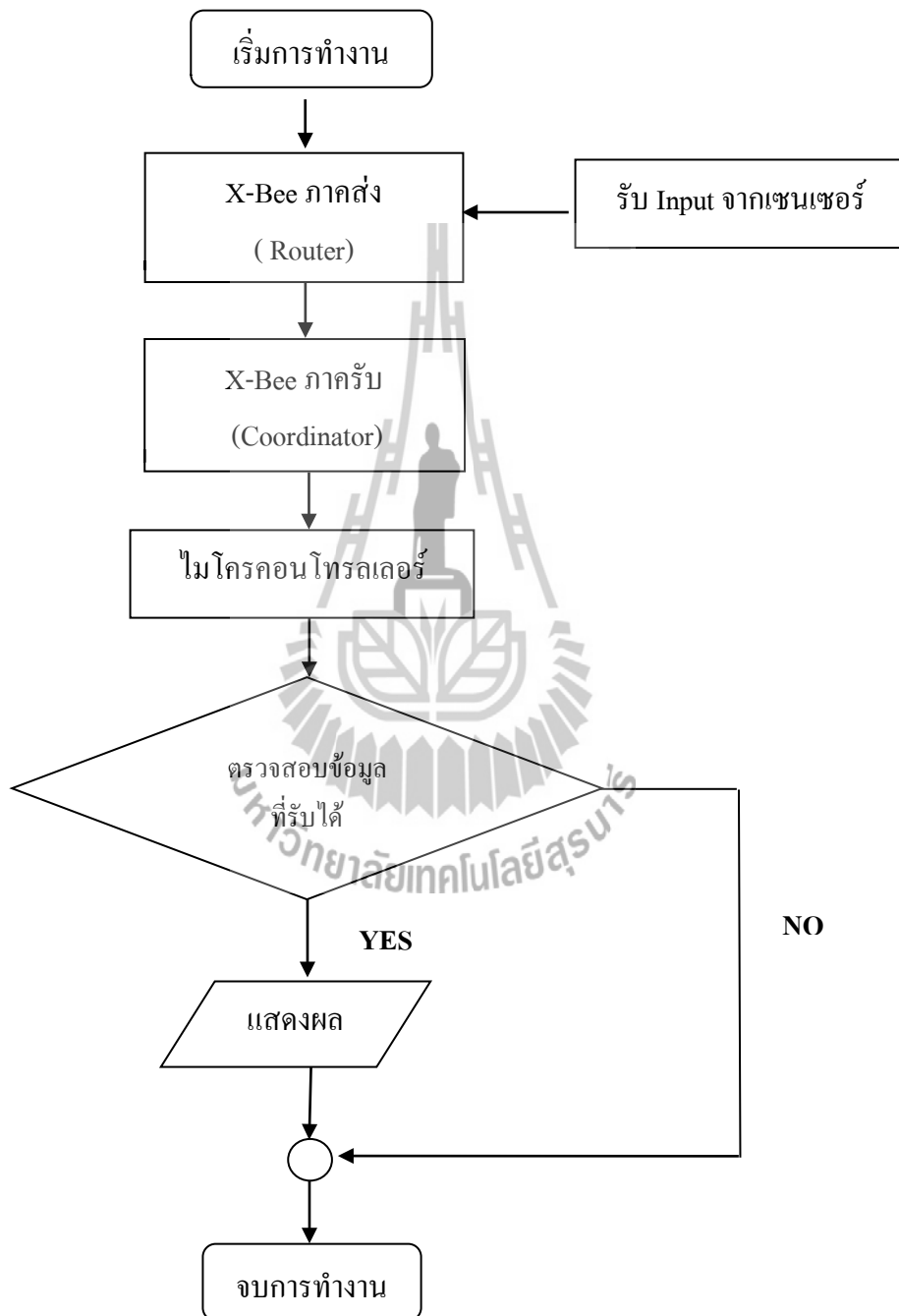
รูปที่ 3.7 เซนเซอร์วัดการสะท้อน QTR-1RC



รูปที่ 3.8 ตู้ไปรษณีย์ที่ออกแบบขึ้น

### 3.2.3 หลักการทำงานของตู้ไปรษณีย์อัจฉริยะ

- แผนภาพการทำงานของตู้ไปรษณีย์อัจฉริยะ



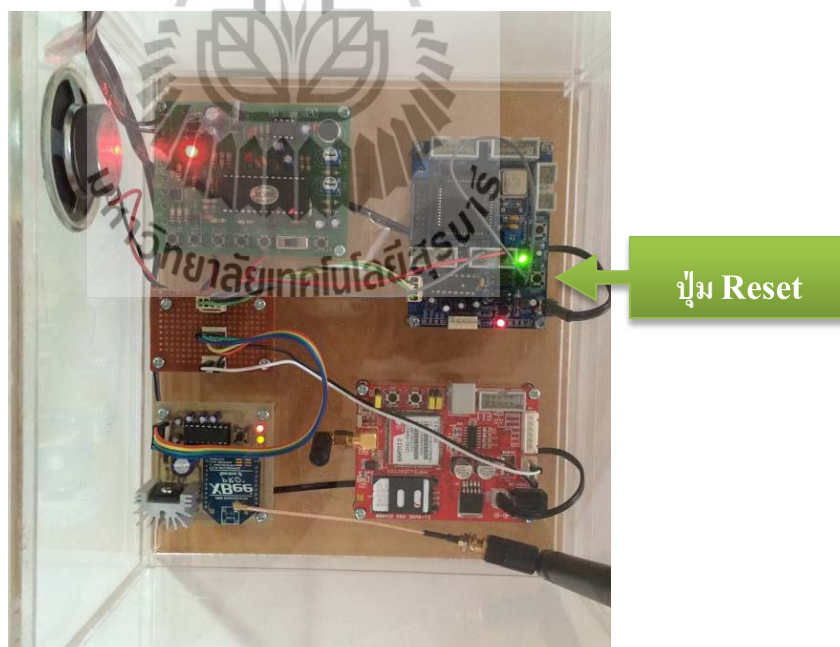
รูปที่ 3.9 แผนภาพการทำงานของตู้ไปรษณีย์อัจฉริยะ

จากรูปที่ 3.9 แผนภาพแสดงการทำงานโดยรวมของโปรแกรมการทำงานจะเริ่มต้นโดยเมื่อ X-Bee ที่ภาคส่งได้รับค่าอินพุตจากเซนเซอร์ X-Bee ที่ภาคส่งจะทำการส่งเฟรมข้อมูลไปยัง X-Bee ภาครับและภาครับก็จะทำการนำเฟรมข้อมูลที่ได้รับมาส่งต่อไปยังบอร์ดไมโครคอนโทรลเลอร์และทำการตรวจสอบข้อมูลเฟรมที่ได้รับได้ว่า X-Bee ทางด้านภาครับเซนเซอร์มีการทำงานหรือไม่ ถ้าใช่ก็ทำการแสดงผลในรูปแบบของเสียงและการส่งข้อความเข้าโทรศัพท์มือถือและจบการทำงาน แต่ถ้าตรวจสอบข้อมูลที่ได้รับมาเซนเซอร์ไม่มีการทำงานจะไม่มีการแสดงผลและจบการทำงาน

### 3.2.4 การใช้งานของผู้โปรยณีย้อจรรย์ะ

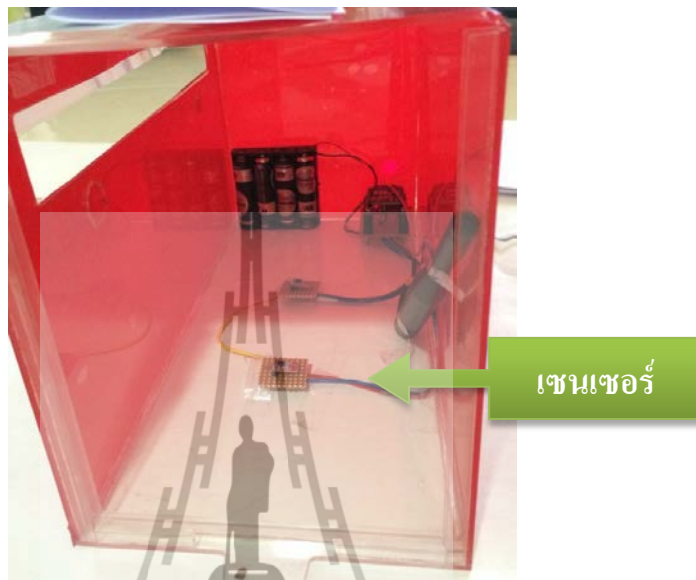
การใช้งานของของผู้โปรยณีย้อจรรย์ะ มีดังนี้

1. ทำการกดปุ่มรีเซ็ตที่บอร์ดไมโครคอนโทรลเลอร์ เพื่อเริ่มการทำงานของ ผู้โปรยณีย้อจรรย์ะ โดยไฟ LED ที่ P1.0 จะกะพริบ 3 ครั้ง

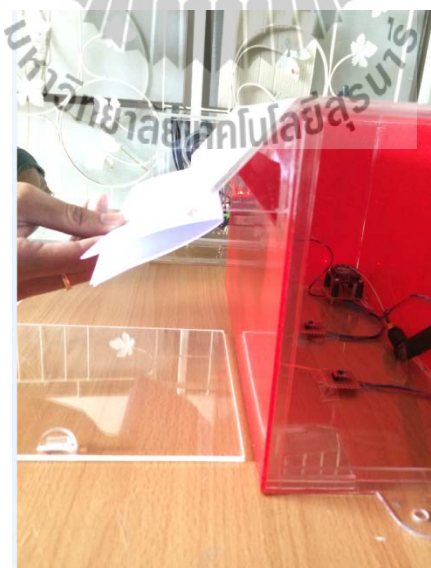


รูปที่ 3.10 กดปุ่มรีเซ็ตที่บอร์ดไมโครคอนโทรลเลอร์

2. เมื่อมีบุรุษไปรษณีย์มาส่งจดหมายลงตู้ไปรษณีย์เซนเซอร์ที่ติดอยู่กับตู้ก็จะเปลี่ยนสถานะจาก Logic 1 เป็น Logic 0 ทำให้ตัวควบคุม X-Bee ตรวจจับได้ถึงการเปลี่ยนแปลงและส่งเฟรมข้อมูลไปยัง X-bee ภาครับภายในบ้าน

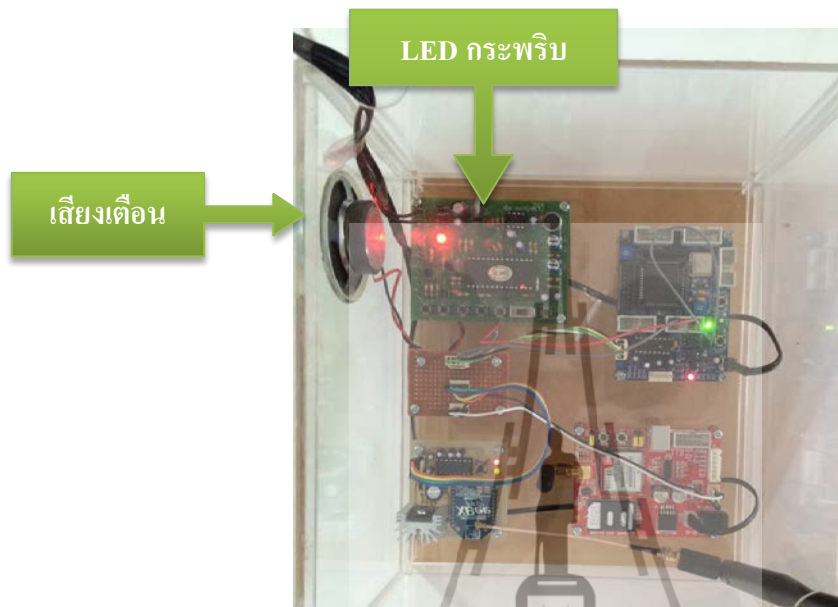


รูปที่ 3.11 เซนเซอร์ที่ติดอยู่ที่กล่องไปรษณีย์



รูปที่ 3.12 เมื่อมีวัตถุมาเป็นตัวสะท้อนกับตัวรับของเซนเซอร์ เซนเซอร์จะเปลี่ยนสถานะ

3. เมื่อมีเฟรมข้อมูลส่งมายัง X-Bee ภาครับภายในบ้านไมโครคอนโทรลเลอร์จะทำการตรวจสอบเฟรมข้อมูลที่ส่งมา ในตำแหน่งไบต์ที่ 20 เมื่อตรวจสอบพบรหัส 0x00 ก็จะทำการแสดงก็จะทำการแสดง สัญญาณไฟเตือนผ่าน LED พร้อมกับส่งเสียงเตือนว่า “มีจดหมายมาส่งค่ะ ” ผ่านลำโพง 3 ครั้ง



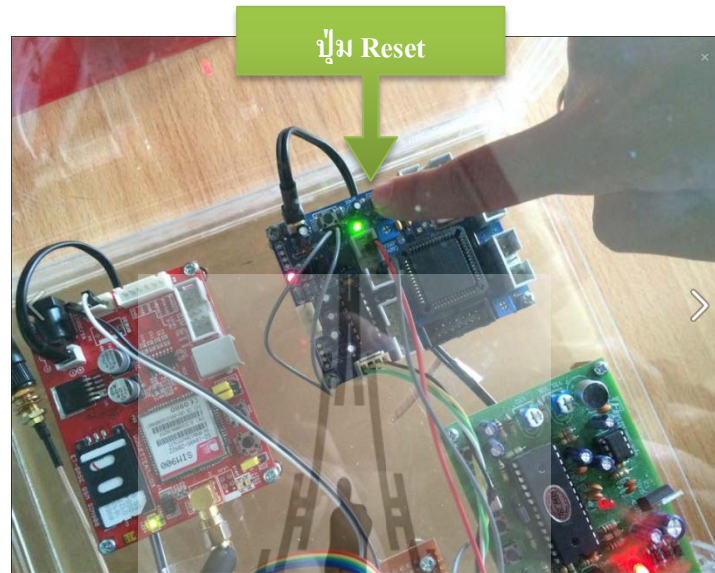
รูปที่ 3.13 เสียงเตือนว่ามีจดหมายมาส่งดังขึ้น 3 ครั้ง

และไมโครคอนโทรลเลอร์จะทำการส่งคำสั่งไปยังบอร์ด GSM SIM900 เพื่อสั่งให้ส่งข้อความ “ You’ve got a letter in your mail box ” ไปยังโทรศัพท์มือถือ

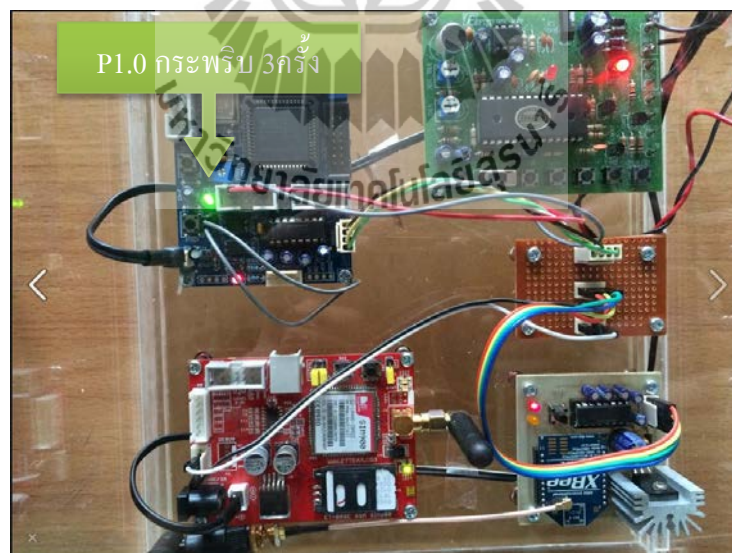


รูปที่ 3.14 ข้อความที่ได้รับทางโทรศัพท์มือถือ

4. เมื่อเจ้าของบ้านไปนำจดหมายออกจากตู้ไปรษณีย์จะต้องทำการรีเซ็ตที่บอร์ดไมโครคอนโทรลเลอร์ภายในบ้าน LED ที่ P1.0 จะกะพริบ 3 ครั้งเพื่อเป็นการเริ่มงานใหม่



รูปที่ 3.15 การกดปุ่มรีเซ็ตที่บอร์ดไมโครคอนโทรลเลอร์ภายในบ้าน

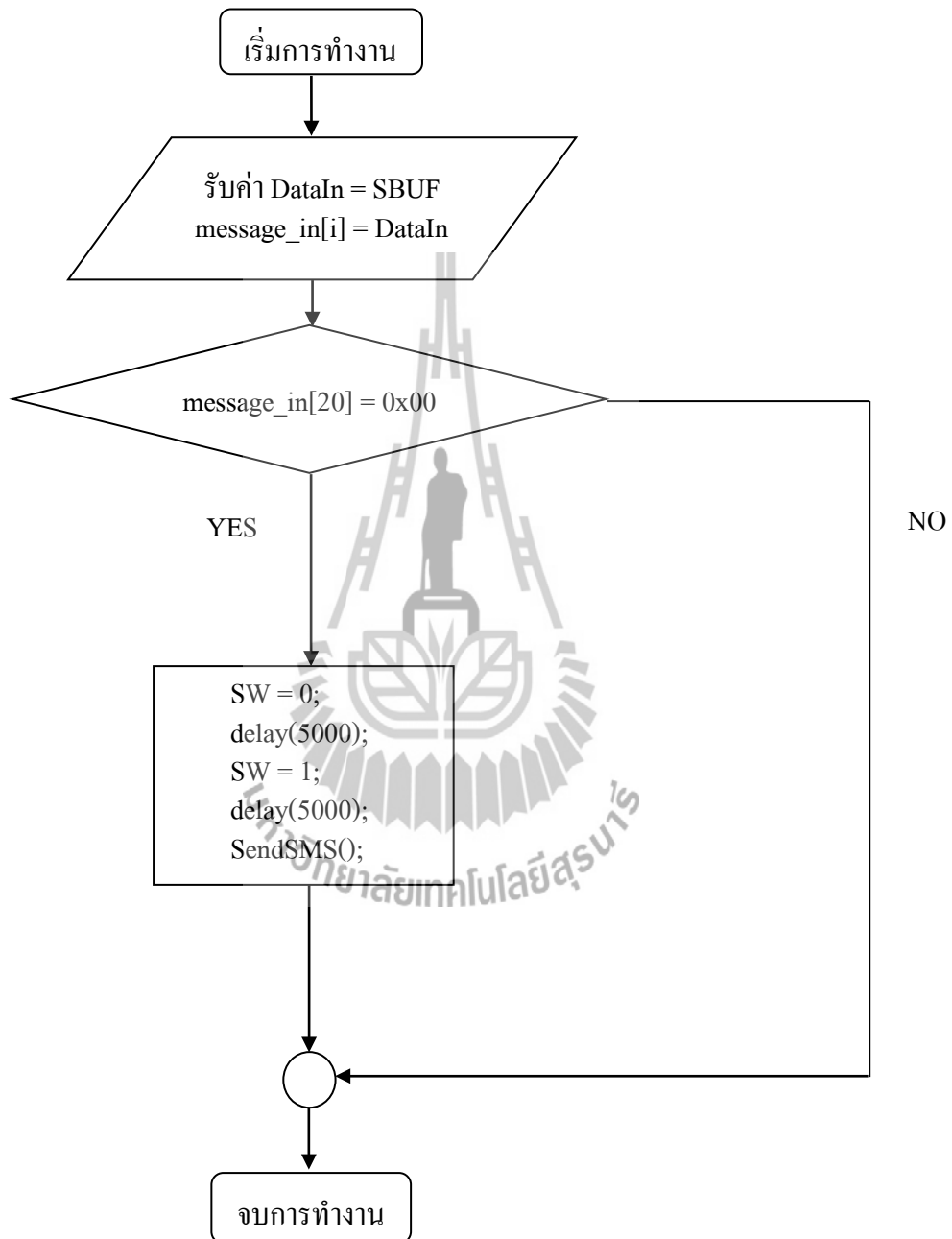


รูปที่ 3.16 หลอดไฟ LED ที่ P1.0 จะกะพริบ 3 ครั้ง

จากรูปที่ 3.16 เมื่อทำการกดปุ่มรีเซ็ต LED ที่ P1.0 จะกะพริบ 3 ครั้ง และระบบจะกลับมาอยู่ในสถานะพร้อมอีกครั้ง ดังรูปที่ 3.16

### 3.3 การออกแบบซอฟต์แวร์ระบบตู้ไปรษณีย์อัจฉริยะ

- แผนภาพการทำงานของตู้ไปรษณีย์อัจฉริยะ



รูปที่ 3.17 การเขียนโปรแกรมการรับเฟรมข้อมูล (Frame Data) จากโมดูล X-Bee

จากรูปที่ 3.17 เมื่อ X-Bee ภาครับได้รับเฟรมข้อมูลการทำงานจะเริ่มต้นโดยรับค่าเฟรมข้อมูลผ่านทาง RS232 เก็บไว้ในตัวแปร DataIn โดยทำการเก็บเฟรมข้อมูลเป็นไบต์เก็บในตัวแปร message\_in[i] จากนั้นทำการตรวจสอบข้อมูลของ message\_in[20] ว่าเท่ากับ 0x00 หรือไม่ ถ้าถูกต้อง(YES) แสดงว่ามีจดหมายมาส่งและจึงทำการให้เสียงเตือนดังขึ้น3ครั้งและส่งข้อความไปยังโทรศัพท์มือถือและจบการทำงาน แต่ถ้าไม่ถูกต้อง(NO) จะไม่มีการแสดงผลแล้วจึงจบการทำงาน



## บทที่ 4

### ผลการทดลอง

#### 4.1 บทนำ

ในบทนี้เราจะกล่าวถึง การทดสอบการส่งข้อมูล (Data) ของ X-Bee ภาครับและภาคส่ง, การ ทดลองรับค่าอินพุตจากเซนเซอร์, การทดสอบการรับ-ส่งเฟรมข้อมูลผ่าน โมดูล X-Bee, การทดลองคำสั่งของบอร์ด ET-BASE GSM SIM900 ผ่านทางบอร์ด ไมโครคอนโทรลเลอร์ และการทดสอบการทำงานของระบบตู้ไปรษณีย์อัจฉริยะ

#### 4.2 การทดลองที่ 1 การทดสอบการรับ-ส่งข้อมูลของ X-Bee ภาครับและภาคส่ง

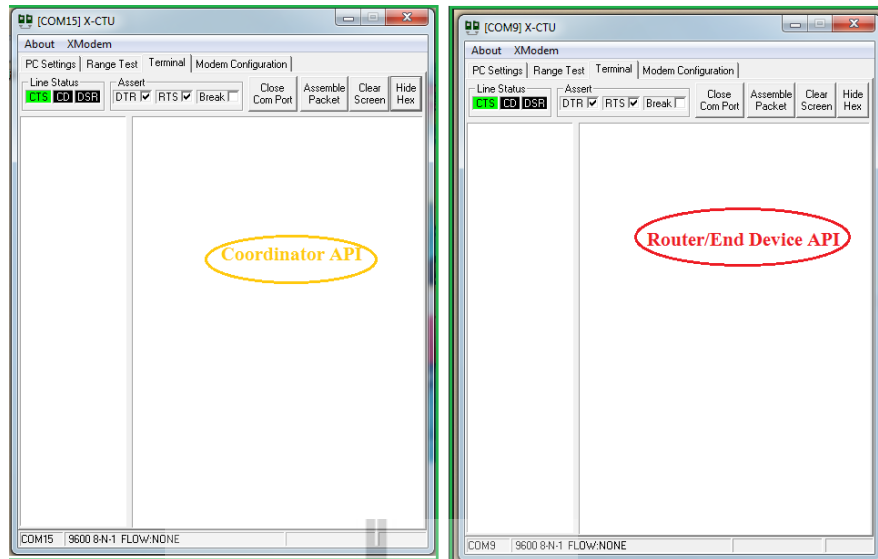
##### 4.2.1 วัตถุประสงค์

1. เพื่อการศึกษาการปรับตั้งค่าของ X-Bee ในโหมด API
2. เพื่อศึกษาการรับ – ส่ง ข้อมูลของโมดูล X-Bee โดยใช้โปรแกรม X-CTU

##### 4.2.2 ขั้นตอนการทดลอง

1. การเขียนเฟิร์มแวร์เป็นภาคส่งในโหมด API (Write Firmware Coordinator API)
2. การเขียนเฟิร์มแวร์เป็นภาครับในโหมด API (Write Firmware Router API)
3. การส่งเฟรมข้อมูล (Send Data)

3.1 ทำการเลือกหัวข้อ Terminal ของทั้งสองตัวดังแสดงในรูปที่ 4.1



รูปที่ 4.1 หน้าต่าง Terminal ของภาคส่ง (Coordinator) และภาครับ (Router)

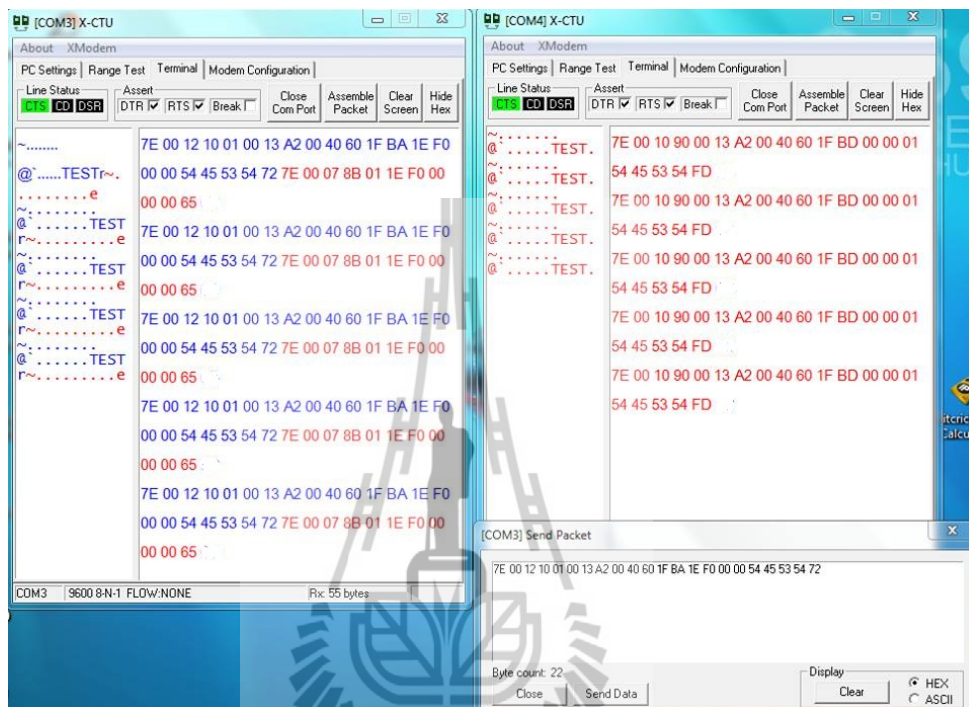
3.2 คลิกที่ Assemble Packet ของภาคส่ง (Coordinator) ซึ่งในการทดลองจะทำการส่งเฟรมข้อมูล (Frame Data) ที่ส่งคือ 7E 00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72 ทั้งหมดจำนวน 5 ครั้งด้วยกันสังเกตผลการทดลอง

#### 4.2.3 ผลการทดลองที่ 1

ตาราง 4.1 การส่งเฟรมข้อมูล (Frame Data) จากภาคส่ง (Coordinator) ไปยังภาครับ (Router) และเฟรมข้อมูลที่ได้รับได้จากภาคส่ง

| ครั้งที่ | เฟรมข้อมูลที่ส่งไปยังภาครับ                                          | เฟรมข้อมูล ที่รับได้จากภาคส่ง                                        |
|----------|----------------------------------------------------------------------|----------------------------------------------------------------------|
| 1        | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 |
| 2        | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 |
| 3        | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 |
| 4        | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 |
| 5        | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 | 7E 00 12 10 01 00 13 A2 00 40 60 1F BA<br>1E F0 00 00 54 45 53 54 72 |

จากตาราง 4.1 เป็นผลการทดลองส่งเฟรมข้อมูลจากภาคส่งไปยังภาครับ จำนวน 5 ครั้ง ซึ่งเฟรมข้อมูลที่ได้ทำการส่งไปนั้น คือ 7E 00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72 และ ภาครับสามารถรับได้ คือ 7E 00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72 ซึ่งเฟรมข้อมูลที่ส่งมา และ รับได้นั้น มีค่าตรงกัน แสดงว่าได้รับข้อมูลที่ถูกต้อง



รูปที่ 4.2 เฟรมข้อมูล (Frame Data) ที่ทำการส่งจากภาคส่ง (Coordinator) ไปยังภาครับ (Router)

จากรูปที่ 4.2 เป็นการส่งเฟรมข้อมูลผ่าน X-CTU ของโมดูล X-Bee ทั้ง 2 ตัว ซึ่งภาพด้านซ้าย คือ ภาคส่ง และ ด้านขวา คือ ภาครับ

#### 4.2.4 วิเคราะห์ผลการทดลองที่ 1

จากการทดลองได้ทำการส่งเฟรมข้อมูล (Frame Data) คือ 7E 00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72 จำนวน 5 ครั้ง ไปยังภาครับ (Router) ซึ่งเฟรมข้อมูลที่สามารถรับได้ คือ 7E 00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72 นอกจากนี้ยังมีการตอบ ACK มาจากภาครับ (Router) เพื่อยืนยันการส่งเฟรมข้อมูลคือ 7E 00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72 ซึ่งความหมายของข้อมูลเหล่านี้มีดังนี้

- เฟรมข้อมูล (Frame Data) ที่ภาคส่ง (Coordinator) ส่งออกไปคือ

7E00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72

สามารถแยกเฟรมข้อมูลออกเป็นกลุ่ม 4 กลุ่ม ได้ดังนี้

7E = เริ่มต้นการกำหนดขอบเขต หรือ เริ่มต้นของเฟรมข้อมูล

00 12 = ความยาวของเฟรมข้อมูล (Length)

10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 = เฟรมข้อมูล ของ X-Bee

Transmit Request เราจะแยกพิจารณา ซึ่งประกอบไปด้วย

10 = ชนิดของเฟรมข้อมูล เป็นไบต์ ที่บอกว่า เป็น X-Bee Transmit Request

01 = Frame ID เป็นตัวกำหนดว่า การส่งเฟรมข้อมูล ครั้งนี้ต้องการการตอบกลับจากตัวรับเพื่อยืนยันว่าการส่งสำเร็จ

00 13 A2 00 40 60 1F BA = 64 บิตข้อมูลที่อยู่ของปลายทาง ก็คือค่า SH SL ของตัวรับ

1E F0 = 16 บิต ข้อมูลที่อยู่ปลายทางของเครือข่าย ก็คือ MY ของตัวรับ

00 = Broadcast Radius ค่า จำนวน Hop สูงสุด

00 = Options ตั้งค่า ไม่ใช้ option ที่ใช้ในการส่ง

54 45 53 54 = เฟรมข้อมูลที่ส่งออกไปในที่นี้คือคำว่า test

72 = ค่า Checksum

- เฟรมข้อมูล (Frame Data) ที่ภาครับ (Router) ได้รับคือ

7E00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72

- สามารถแยกเฟรมข้อมูลออกเป็นกลุ่ม 4 กลุ่ม ได้ดังนี้

7E = เริ่มต้นการกำหนดขอบเขต หรือ เริ่มต้นของเฟรมข้อมูล

00 12 = ความยาวของเฟรมข้อมูล (Length)

10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 = เฟรมข้อมูล ของ X-Bee

Transmit Request เราจะแยกพิจารณา ซึ่งประกอบไปด้วย

10 = ชนิดของเฟรมข้อมูล เป็นไบต์ ที่บอกว่า เป็น X-Bee Transmit Request

01 = Frame ID เป็นตัวกำหนดว่า การส่งเฟรมข้อมูล ครั้งนี้ต้องการการตอบกลับจากตัวรับเพื่อยืนยันว่าการส่งสำเร็จ

00 13 A2 00 40 60 1F BA = 64 บิตข้อมูลที่อยู่ของปลายทาง ก็คือค่า SH SL ของตัวรับ  
 1E F0 = 16 บิต ข้อมูลที่อยู่ปลายทางของเครือข่าย ก็คือ MY ของตัวรับ  
 00 = Broadcast Radius ค่า จำนวน Hop สูงสุด  
 00 = Options ตั้งค่า ไม่ใช้ option ที่ใช้ในการส่ง  
 54 45 53 54 = เฟรมข้อมูลที่ส่งออกไปในที่นี้คือคำว่า test  
 72 =ค่า Checksum

- นอกจากนี้ยังมีการตอบ ACK มาจากภาครับ(Router) เพื่อยืนยันการส่งเฟรมข้อมูล (Data) 7E 00 07 8B 01 1E F0 00 00 00 65

แยกเฟรมข้อมูลออกเป็นกลุ่ม 4 กลุ่ม

7E = เริ่มต้นการกำหนดขอบเขต หรือ เริ่มต้นของเฟรมข้อมูล

00 07 = ความยาวของเฟรมข้อมูล (Length)

8B 01 1E F0 00 00 00 = เฟรมข้อมูลของ X-Bee Transmit Status ประกอบด้วย

8B = ชนิดของเฟรมข้อมูลของ X-Bee Transmit Status

01 = Frame ID

1E F0 = 16 บิต ข้อมูลที่อยู่ปลายทางของเครือข่าย คือ MY ของตัวรับ

00 = Transmit Retry Count จำนวนครั้งที่เกิดการ Retry

00 00 = สถานะของการส่งเสร็จสมบูรณ์

65 = ค่า Checksum

#### 4.2.5 สรุปผลการทดลองที่ 1

จากการทดลองส่งเฟรมข้อมูล (Frame Data) จำนวน 5 ครั้งจากภาคส่ง (Coordinator) ไปยังภาครับ (Router) ปรากฏว่า ในโหมด API การส่งเฟรมข้อมูล (Frame Data) นั้นต้องส่งเป็น โปรโตคอล API ซึ่งมีรูปแบบดังแสดงในรูปที่ 4.3



รูปที่ 4.3 รูปแบบของโปรโตคอล API

เห็นได้จากการทดลองที่ส่งคำว่า “TEST” โดยใช้เฟรมข้อมูล 7E 00 12 10 01 00 13 A2 00 40 60 1F BA 1E F0 00 00 54 45 53 54 72 เป็นโปรโตคอล API ดังนั้น เราสามารถส่งเฟรมข้อมูลที่เป็นข้อความได้โดยตรงในโหมด API ซึ่งเฟรมข้อมูลที่ส่งมา และรับได้นั้น มีค่าตรงกัน แสดงว่าได้รับข้อมูลที่ถูกต้อง

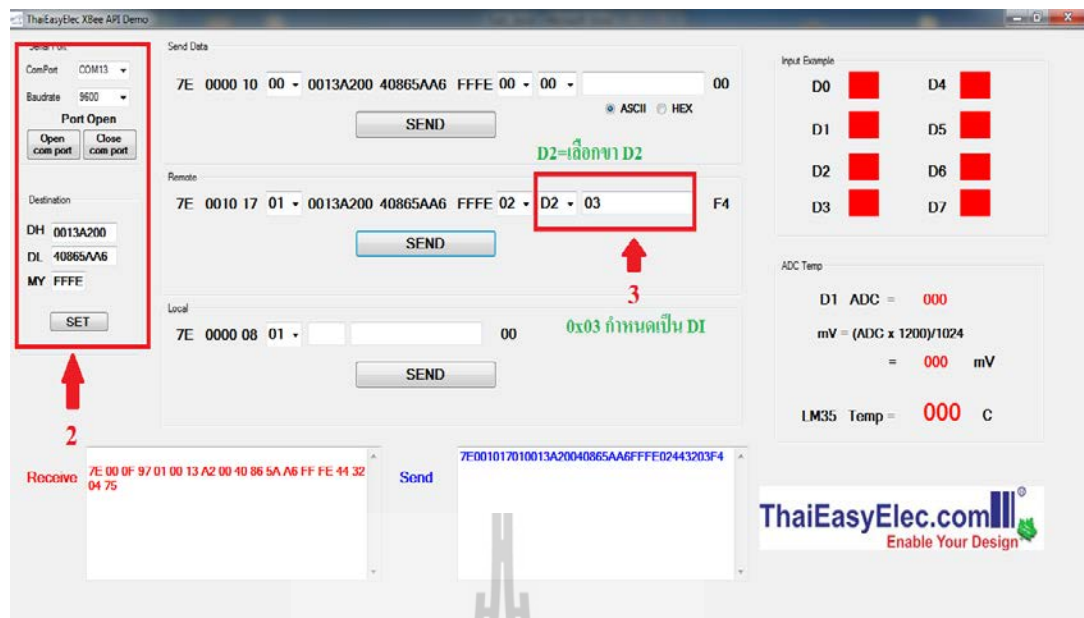
### 4.3 การทดลองที่ 2 การทดลองการรับข้อมูลของโมดูล X-Bee เมื่อมีการเปลี่ยนแปลงของขา X-Bee

#### 4.3.1 วัตถุประสงค์

1. เพื่อศึกษาการรับส่งข้อมูลของโมดูล X-Bee เมื่อมีการเปลี่ยนแปลงของขา X-Bee
2. เพื่อศึกษาการตั้งค่าและการใช้โปรแกรม X-Bee API Demo

#### 4.3.2 ขั้นตอนการทดลอง

1. ทำการต่อเซนเซอร์เข้ากับ ขา DI02 ของโมดูล X-Bee
2. ทำการเปิดโปรแกรม X-Bee API Demo เลือกพอร์ตสำหรับเชื่อมต่อแบบอนุกรม (Comport) ที่ต่ออยู่กับภาคส่ง (Coordinator) แล้วเลือก open port จากนั้นทำการกำหนดค่า DH , DL, MY ไปยังตัวที่ต้องการ Remote ในที่นี้คือ ภาครับ (Router) ซึ่งมี DH = 0013A200 , DL = 40865AA6 และ MY = FFFE จากนั้นทำการกด SET ดังแสดงในรูปที่ 4.4
3. ทำการกำหนดให้ Pin D2 ของภาคส่ง (Router) เป็น DI=0x03 จากนั้นกด SEND ดังแสดงในรูปที่ 4.4
4. Data ที่ส่งออกไป : 7E001017010013A20040865AA6FFFE02443203F4



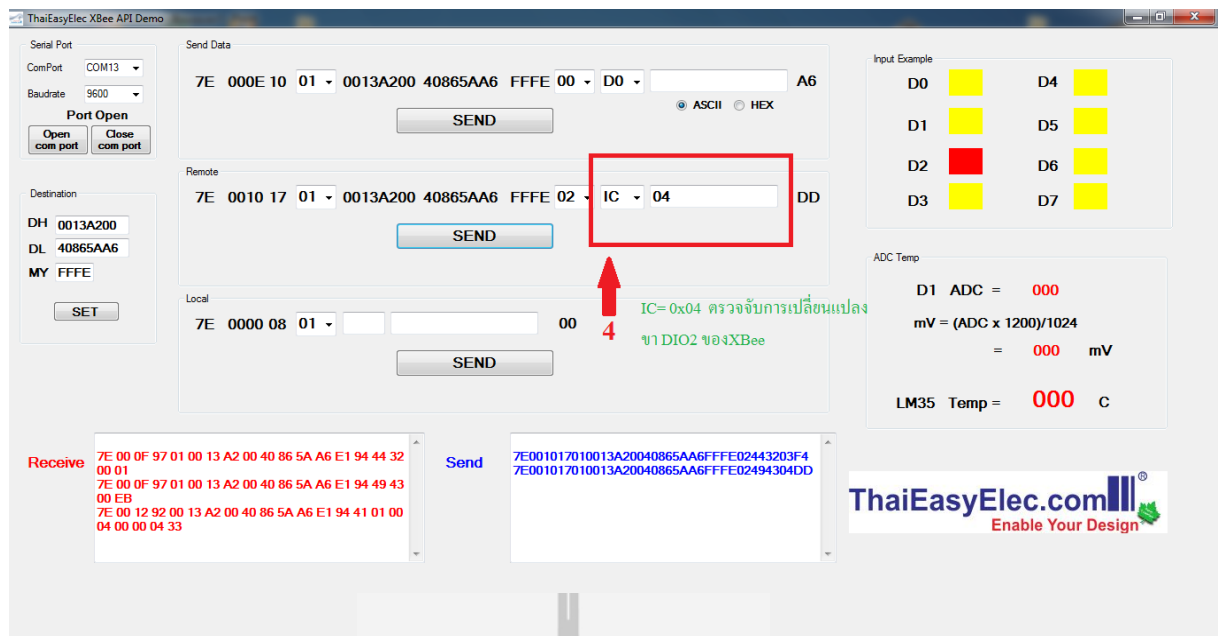
รูปที่ 4.4 การตั้งค่า Destination และการกำหนดขาอินพุตเป็น DI

- กำหนดให้ X-Bee ภาครับ (Router) ส่งเฟรมข้อมูลกลับมายังภาคส่ง (Coordinator) โดยใช้ IC (Digital IO Change Detection) เป็นตัวกำหนดให้มีการ Detect Port ที่เราต้องการเอาไว้เมื่อ Logic ที่ Port นั้นมีการเปลี่ยนแปลงให้ให้ส่งเฟรมข้อมูล (Frame Data) ไปแสดงที่ภาคส่งบนโปรแกรม X-Bee API Demo ในการทดลองนี้ เราจะตรวจจับการเปลี่ยนแปลงขา DIO2 ของ X-Bee ดังนั้น ตั้ง IC = 0x04 คัดจากโครงสร้างของขาอินพุต ดังรูปที่ 4.5 โดยเลือกใช้ที่ขา DIO2 = 000000000100 แล้วกด SEND แสดงดังรูปที่ 4.6

Data ที่ส่งออกไป : 7E001017010013A20040865AA6FFFE02494304DD



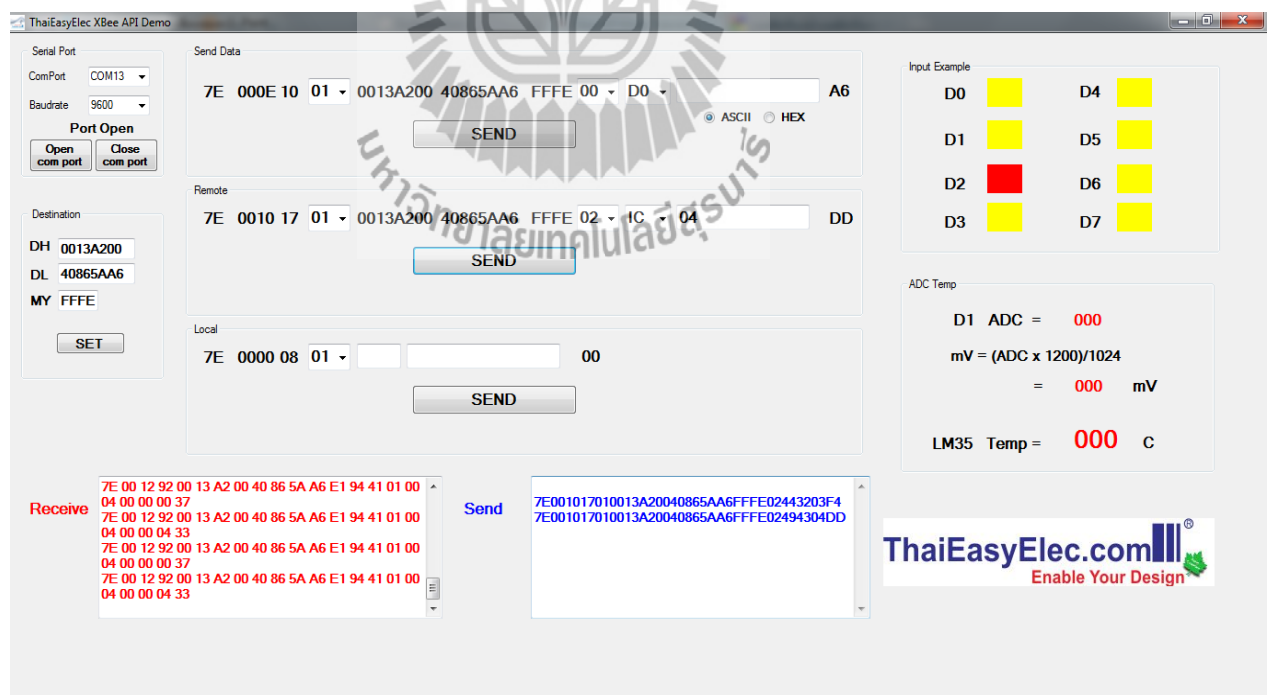
รูปที่ 4.5 โครงสร้าง IC ของขาอินพุต X-Bee



รูปที่ 4.6 การตั้งค่า Digital IO Change Detection ให้กับขาอินพุตของ X-Bee

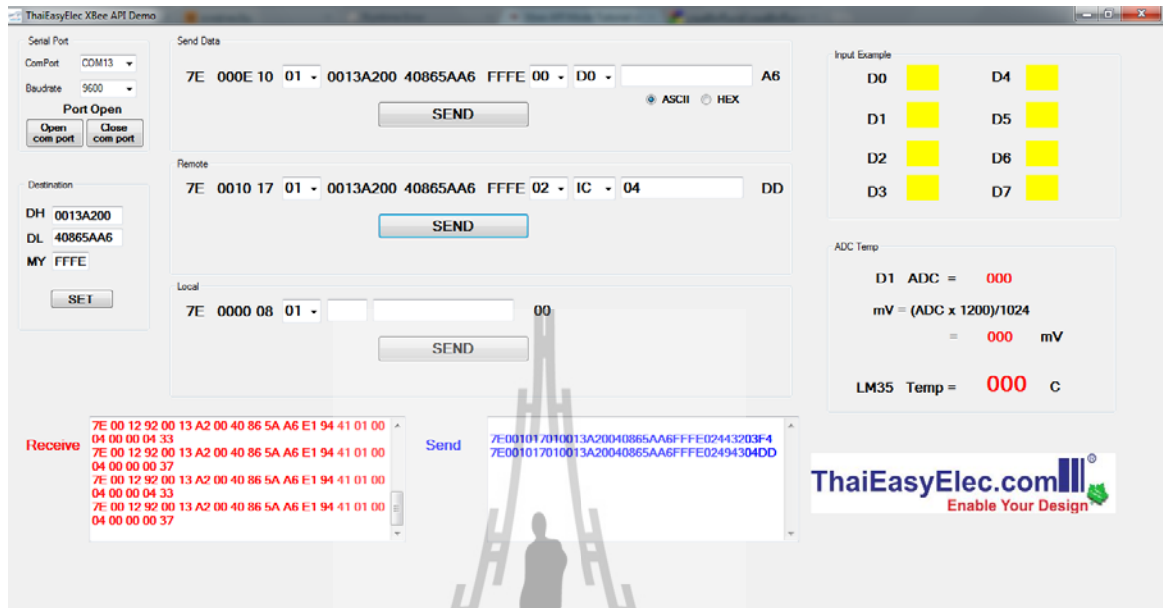
6. จากนั้นทำการนำวัตถุมาวางเป็นตัวสะท้อนของเซนเซอร์เพื่อตรวจสอบการเปลี่ยนแปลงของขา DI02 สังเกตผลการทดลอง

#### 4.3.3 ผลการทดลองที่ 2



รูปที่ 4.7 ผลตอบสนองของเฟรมข้อมูล (Frame Data) เมื่อยังไม่มีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์

จากรูปที่ 4.7 เป็นหน้าจอของโปรแกรม X-Bee API Demo ในขณะยังไม่มีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์ และทำการอ่านค่าเฟรมข้อมูล (Frame Data) ที่ได้รับ คือ **7E**  
**00 12 92 00 13 A2 00 40 86 5A A6 9E 4D 41 01 00 04 00 00 04 BD**



รูปที่ 4.8 ผลตอบสนองของเฟรมข้อมูล (Frame Data) เมื่อมีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์

จากรูปที่ 4.8 เป็นหน้าจอของโปรแกรม X-Bee API Demo ในขณะมีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์แล้ว จากนั้นทำการอ่านค่าเฟรมข้อมูล (Frame Data) ที่ได้รับ คือ **7E00 12 92 00 13 A2 00 40 86 5A A6 9E 4D 41 01 00 0400 00 00 C1**

#### 4.3.4 วิเคราะห์ผลการทดลองที่ 2

จากการทดลอง รับค่าอินพุตจากเซนเซอร์ซึ่งในการทดลองได้ทำการตรวจสอบขาอินพุต DI02 ของโมดูล X-Bee ผลปรากฏว่า เมื่อมีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์แล้วที่ขา DI02 มีการเปลี่ยนแปลง โดยการส่งเฟรมข้อมูล (Frame Data) ตอบกลับมายังโปรแกรม X-Bee API Demo สังเกตจากโปรแกรม ขณะที่ยังไม่มีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์นั้น ช่อง Input Example ของ DI02 นั้นมีสีแดงแต่เมื่อมีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์ ช่อง Input Example ของ DI02 มีสีเหลือง แสดงว่า ที่ขา DI02 มีการเปลี่ยนแปลงเกิดขึ้นนั่นเอง และData ที่ได้รับมีดังนี้

- เฟรมข้อมูลที่ได้รับจากภาครับ (Router) ขณะไม่มีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์

7E 00 12 92 00 13 A2 00 40 86 5A A6 9E 4D 41 01 00 04 00 00 04 BD

สามารถแยกเฟรมข้อมูลออกเป็นกลุ่ม 4 กลุ่ม ได้ดังนี้

7E = เริ่มต้นการกำหนดขอบเขต หรือ เริ่มต้นของเฟรมข้อมูล

00 12 = ความยาวของเฟรมข้อมูล (Length)

92 00 13 A2 00 40 86 5A A6 9E 4D 41 01 00 04 00 00 04 = เฟรมข้อมูลของ X-Bee

IO Data Sample Rx Indicator

92 = ชนิดของเฟรมข้อมูล คือ X-Bee IO Data Sample Rx Indicator

00 13 A2 00 40 86 5A A6 = 64 บิต ของที่อยู่ของภาครับ

9E 4D = 16 บิต ข้อมูลที่อยู่ของปลายทาง

01 = ตั้งค่าให้มีการส่งค่ากลับ

00 04 = ตรวจสอบการเปลี่ยนแปลงของ DIO2

00 04 = ภา DIO2 ไม่มีการเปลี่ยนแปลง

BD = Checksum

- เฟรมข้อมูลที่ได้รับจากภาครับ (Router) ขณะมีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์

7E 00 12 92 00 13 A2 00 40 86 5A A6 9E 4D 41 01 00 04 00 00 00 C1

สามารถแยกเฟรมข้อมูลออกเป็นกลุ่ม 4 กลุ่ม ได้ดังนี้

7E = เริ่มต้นการกำหนดขอบเขต หรือ เริ่มต้นของเฟรมข้อมูล

00 12 = ความยาวของเฟรมข้อมูล (Length)

92 00 13 A2 00 40 86 5A A6 9E 4D 41 01 00 04 00 00 00 = เฟรมข้อมูลของ X-Bee

IO Data Sample Rx Indicator

92 = ชนิดของเฟรมข้อมูล คือ X-Bee IO Data Sample Rx Indicator

00 13 A2 00 40 86 5A A6 = 64 บิต ของที่อยู่ของภาครับ

9E 4D = 16 บิต ข้อมูลที่อยู่ของปลายทาง

01 = ตั้งค่าให้มีการส่งค่ากลับ

00 04 = ตรวจสอบการเปลี่ยนแปลงของ DIO2

0000 = ขา DIO2 มีการเปลี่ยนแปลง

C1 = Checksum

จากเฟรมข้อมูลที่ได้รับได้จากภาครับ (Router) นั้น จะเห็นได้ว่า มีเฟรมข้อมูลที่เหมือนกัน ยกเว้นตำแหน่งของไบต์ที่ 21 และ 22 ซึ่งตำแหน่งของไบต์ที่ 21 ต่างกันเมื่อมีวัตถุมารเป็นตัวสะท้อนของเซนเซอร์และไม่มีวัตถุมารเป็นตัวสะท้อนของเซนเซอร์คือ 00 และ 04 ตามลำดับ ส่วนตำแหน่งของไบต์ที่ 22 เป็น Checksum ซึ่งจะแตกต่างกันตามลักษณะของเฟรมข้อมูลนั่นเอง

#### 4.3.5 สรุปผลการทดลองที่ 2

จากการทดลองสรุปได้ว่า โมดูล X-Bee สามารถตรวจจับค่าอินพุตที่ขาของโมดูล X-Bee โดยเมื่อมีการเปลี่ยนแปลงสถานะที่ขาของโมดูล X-Bee ก็มีการส่งเฟรมข้อมูลไปยังภาคส่ง (Coordinator) เพื่อแสดงผลที่โปรแกรม X-Bee API Demo ซึ่งเฟรมข้อมูลที่ได้รับมานั้น เมื่อมีการเปลี่ยนแปลงของขาโมดูล X-Bee จะมีความแตกต่างของเฟรมข้อมูลในตำแหน่งของไบต์ที่ 21 และ 22

### 4.4 การทดลองที่ 3 การทดสอบการส่งข้อความของ ET-BASE GSM SIM900 ผ่านทาง บอร์ดไมโครคอนโทรลเลอร์

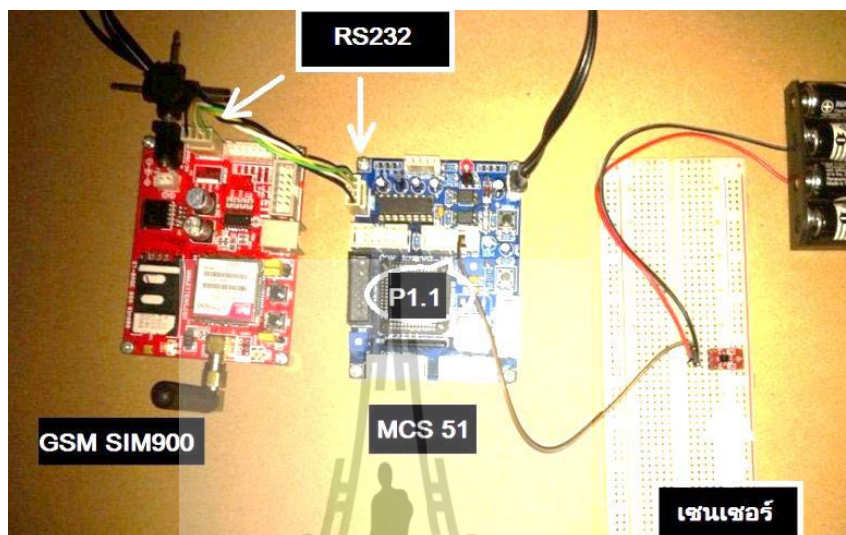
#### 4.4.1 วัตถุประสงค์

1. เพื่อการศึกษาคำสั่งที่ใช้งานในการส่งข้อความของ บอร์ด ET-BASE GSM SIM900
2. เพื่อศึกษาโปรแกรม Tera term และ โปรแกรมภาษา C เบื้องต้น

#### 4.4.2 วิธีการทดลอง

1. เขียนโค้ดโปรแกรมภาษา C ใน Kiel version 3 สำหรับโปรแกรมของบอร์ดไมโครคอนโทรลเลอร์
2. แล้วทำการโหลดโปรแกรมที่ได้เขียนไว้แล้วใส่ลงในบอร์ดไมโครคอนโทรลเลอร์ ในที่นี้ใช้ ET-BASE51 AC3 (การ Download Hex file ให้กับ MCU ของบอร์ดด้วยโปรแกรม FLIP 2.4.6 แบบ Manual)

3. เชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ ET-BASE51 AC3 และบอร์ด ET-BASE GSM SIM 900 ด้วยพอร์ต ET-RS232
4. ต่อขาเอาต์พุตของเซนเซอร์เข้ากับ พอร์ต P1.1 ของบอร์ดไมโครคอนโทรลเลอร์
5. ทดสอบเซนเซอร์โดยการนำวัตถุมาเป็นตัวสะท้อนของเซนเซอร์ สังเกตผลการทดลอง



รูปที่ 4.9 เชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ ET-BASE51 AC3 และบอร์ด ET-BASE GSM SIM900 ด้วยพอร์ต ET-RS232

```
void SendSMS()
{
    printf("AT\n");
    delay(2000);
    printf("AT+CMGF=1\n");
    delay(2000);
    printf("AT+CSMP=17,167,0,0\n");
    delay(2000);
    printf("AT+CMGS=\"+66804653615\"\n");
    delay(2000);
    printf("You've got a letter in your mailbox!\n");
    putchar(0x1A);
    printf("\n");
    putchar(0x0A);
    putchar(0x0D);

    return;
}
```

รูปที่ 4.10 โค้ดคำสั่งที่ใช้ในการส่งข้อความ



รูปที่ 4.11 การส่งข้อความว่า “ You’ve got a letter in your mail box! ”

#### 4.4.3 ผลการทดลองที่ 3

ตารางที่ 4.3 การทดสอบการส่งข้อความของ ET-BASE GSM SIM900 ผ่านทางบอร์ด ไมโครคอนโทรลเลอร์

| ครั้งที่ | ได้/ไม่ได้รับข้อความ |
|----------|----------------------|
| 1        | ได้รับ               |
| 2        | ได้รับ               |
| 3        | ได้รับ               |
| 4        | ได้รับ               |
| 5        | ได้รับ               |

#### 4.4.4 วิเคราะห์ผลการทดลอง

จากผลการทดลองที่ 3 จะเห็นได้ว่า เมื่อเซนเซอร์มีการตรวจจับการสะท้อน เซนเซอร์จะทำงานโดยการส่ง logic ไปยัง พอร์ต 1.1 แล้วบอร์ด ไมโครคอนโทรลเลอร์จะทำการประมวลผล แล้วส่งคำสั่งผ่านทางพอร์ต RS232 ไปยังบอร์ด ET-BASE GSM SIM900 แล้วจึงส่งข้อความเข้าโทรศัพท์มือถือ จากผลการทดลองสามารถส่งข้อความได้ทั้งหมดโดยที่ไม่เกิดความผิดพลาดเลย

#### 4.4.5 สรุปผลการทดลองที่ 3

จากผลการทดลอง การใช้บอร์ดไมโครคอนโทรลเลอร์สั่งงานให้บอร์ด ET-BASE GSM SIM900 ส่งข้อความออกไปนั้นสามารถส่งข้อความได้โดยที่ไม่เกิดความผิดพลาดเลย ทำให้ทราบว่าโปรแกรมที่เขียนให้กับบอร์ดไมโครคอนโทรลเลอร์นั้นสามารถใช้งานได้

### 4.5 การทดลองที่ 4 การทดสอบการทำงานของระบบตู้ไปรษณีย์อัจฉริยะ

#### 4.5.1 วัตถุประสงค์

1. ศึกษาระยะทางที่โมดูล X-Bee สามารถส่งถึงกันและกันได้ เสียงเตือนและส่งข้อความเข้าโทรศัพท์มือถือเพื่อเป็นการแจ้งเตือน
2. เพื่อทดสอบการแจ้งเตือนผู้ใช้งานทั้งการแจ้งเตือนโดยใช้สัญญาณไฟ (LED) สัญญาณเสียงจากอุปกรณ์บันทึกเสียงแบบดิจิทัล และส่ง SMS เข้ามือถือ
3. เพื่อทดสอบการทำงานของระบบ

#### 4.5.2 ขั้นตอนการทดลอง

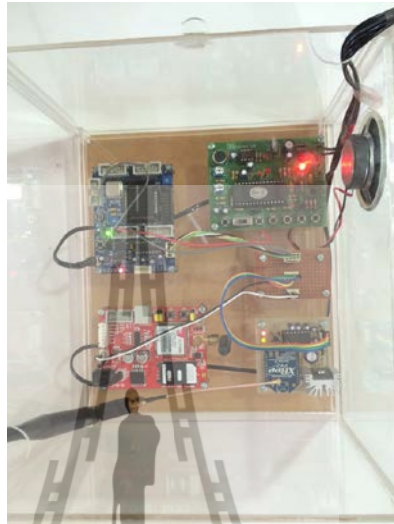
1. ทำการต่อโมดูล X-Beeของภาคส่ง (Router) เข้ากับเซนเซอร์ ดังแสดงในรูปที่ 4.12



รูปที่ 4.12 ภาพแสดงการต่อเซนเซอร์ในตู้ไปรษณีย์

2. เขียนโค้ดโปรแกรมภาษา C ใน Kiel version 3 ดังที่แสดงไว้ในภาคผนวก ก
3. ทำการจ่ายไฟ 5V AC ให้กับบอร์ดไมโครคอนโทรลเลอร์ ET-BASE51 AC3

4. ทำการต่อ ET-RS232 Download เข้ากับพอร์ตที่แผงไมโครคอนโทรลเลอร์ แล้วทำการโหลดโปรแกรมที่ได้เขียนไว้แล้วในเครื่องคอมพิวเตอร์ลงในบอร์ดไมโครคอนโทรลเลอร์ ในที่นี้ใช้ ET-BASE51 AC3
5. ต่อบอร์ดไมโครคอนโทรลเลอร์ ET-BASE51 AC3, ET-BASE GSM SIM900 และ X-Bee ดังแสดงในรูปที่ 4.13



รูปที่ 4.13 ภาพแสดงการต่ออุปกรณ์ภายในบ้าน

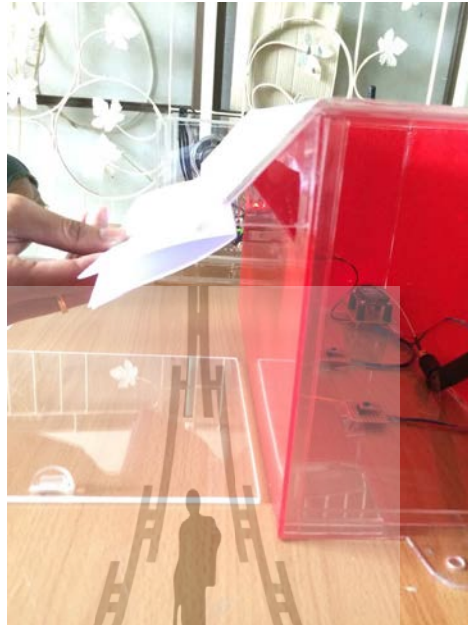
6. ต่อพอร์ต P1.1 เข้ากับสวิทช์ของบอร์ดบันทึกเสียงดิจิตอลและพอร์ต GND เข้ากับ Ground ของบอร์ดบันทึกเสียงดิจิตอล
7. ทำการวาง X-Bee ฝั่งภาครับที่ต่อกับเซนเซอร์อยู่ภายนอกบ้านให้ห่างจาก X-Bee ฝั่งภาคส่งที่ต่อกับบอร์ดไมโครคอนโทรลเลอร์ภายในบ้านห่างประมาณ 10 เมตร
8. กด Reset เพื่อทำการส่งโปรแกรมให้กับ X-Bee ภาครับ สังเกตผลการทดลอง
9. จากนั้นทำการวาง X-Bee ฝั่งภาครับที่ต่อกับเซนเซอร์อยู่ภายนอกบ้านให้ห่างจาก X-Bee ฝั่งภาคส่งที่ต่อกับบอร์ดไมโครคอนโทรลเลอร์ภายในบ้านห่างเพิ่มทีละ 10 เมตร บันทึกผลลงในตาราง 4.4

#### 4.5.3 ผลการทดลองที่ 4

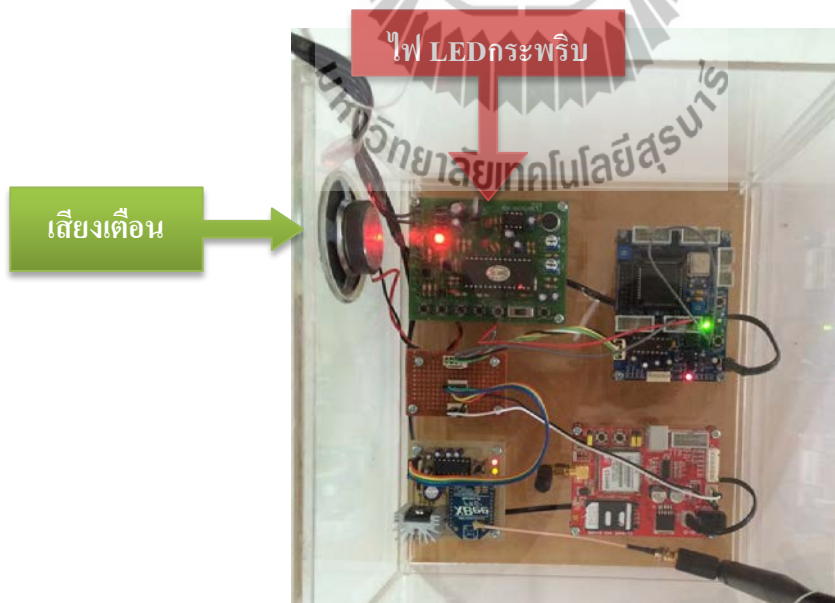
| ระยะห่างของโมดูล X-Bee ภาคส่งกับ<br>โมดูล X-Bee ภาครับที่ต่อกับเซนเซอร์ | สถานะของวงจร<br>บันทึกเสียงดิจิทัล | การส่ง SMS เข้า<br>โทรศัพท์มือถือ |
|-------------------------------------------------------------------------|------------------------------------|-----------------------------------|
| 10                                                                      | มีเสียง                            | ได้รับ                            |
| 20                                                                      | มีเสียง                            | ได้รับ                            |
| 30                                                                      | มีเสียง                            | ได้รับ                            |
| 40                                                                      | มีเสียง                            | ได้รับ                            |
| 50                                                                      | มีเสียง                            | ได้รับ                            |
| 60                                                                      | มีเสียง                            | ได้รับ                            |
| 70                                                                      | มีเสียง                            | ได้รับ                            |
| 80                                                                      | มีเสียง                            | ได้รับ                            |
| 90                                                                      | มีเสียง                            | ได้รับ                            |
| 100                                                                     | มีเสียง                            | ได้รับ                            |
| 110                                                                     | มีเสียง                            | ได้รับ                            |
| 120                                                                     | มีเสียง                            | ได้รับ                            |
| 130                                                                     | ไม่มีเสียง                         | ไม่ได้รับ                         |
| 140                                                                     | ไม่มีเสียง                         | ไม่ได้รับ                         |
| 150                                                                     | ไม่มีเสียง                         | ไม่ได้รับ                         |

ตารางที่ 4.4 การตรวจสอบการรับ-ส่งสัญญาณระหว่างโมดูล X-Bee ภาครับกับโมดูล X-Bee ภาคส่งที่ต่อกับเซนเซอร์และการส่งข้อความผ่าน ET-BASE GSM SIM900

จากตาราง 4.4 ในระยะ 10 – 120 นิ้วตัวรับและตัวส่งสามารถส่งข้อมูลหากันได้ เห็นได้จากมีเสียงจากอุปกรณ์บันทึกเสียงดังขึ้น และสามารถส่งข้อความเข้าโทรศัพท์มือถือได้ แต่เมื่อระยะทางเกิน 120 เมตร จะไม่มีการแสดงผลทั้งเสียงและข้อความ



รูปที่ 4.14 การทดสอบการส่งจดหมายในตู้ไปรษณีย์



รูปที่ 4.15 กล่องภาครับในบ้านจะส่งเสียงเตือน



รูปที่ 4.16 การส่งข้อความเพื่อแจ้งเตือนว่ามีจดหมายมาส่ง

#### 4.5.4 วิเคราะห์ผลการทดลองที่ 4

จากผลการทดลองในตาราง 4.4 ได้ทำการทดสอบระยะทางที่โมดูล X-Bee ทั้งสองตัวสามารถส่งสัญญาณหากันได้ ซึ่งจะเห็นได้จากระยะทาง 0 – 120 เมตร เมื่อมีวัตถุมาเป็นตัวสะท้อนที่เซนเซอร์ แล้วก็จะทำเสียงเตือนก็ดังพร้อมทั้งส่งข้อความแจ้งเตือนเข้าโทรศัพท์มือถือ แต่เมื่อเพิ่มระยะทางให้มากกว่า 120 เมตร โมดูล X-Bee ทั้งสองตัวไม่สามารถรับ – ส่งข้อมูลติดต่อกันได้ ทำให้ไมโครคอนโทรลเลอร์ไม่สามารถตรวจจ็บรหัส 0x00 ได้ จึงส่งผลให้เสียงเตือนไม่ดัง และไม่สามารถส่งข้อความเพื่อแจ้งเตือนว่ามีจดหมาย มาส่งแล้ว ทั้งนี้เนื่องจากมีสิ่งกีดขวางระหว่างตัวรับและตัวส่ง ทำให้ระยะทางในการรับ – ส่งสั้นลง ซึ่งที่ระยะทาง 120 เมตร ก็ถือว่าเพียงพอต่อการใช้งานจริง

#### 4.5.5 สรุปผลการทดลองที่ 4

ผลจากการทดลองได้แสดงให้เห็นว่าโมดูล X-Bee สามารถรับ – ส่งสัญญาณได้ในระยะทางสูงสุด 120 เมตร ซึ่งเมื่อมีวัตถุมาเป็นตัวสะท้อนของเซนเซอร์ในระยะนี้ผลตอบสนองของภาครับที่อยู่ภายในบ้านก็จะมีเสียงพูดว่า “มีจดหมายมาส่งแล้วค่ะ” พร้อมทั้งส่งข้อความทางโทรศัพท์มือถือ ทำให้ทราบว่าภายนอกบ้านมีไปรษณีย์มาส่งจดหมาย และสามารถรับรู้ข่าวสาร การแจ้งเตือนจดหมายได้จากการส่งข้อความ(SMS)เพื่อความสะดวก ในกรณีที่เรายอยู่นอกบ้านหรือไปต่างจังหวัด เราก็สามารถรับรู้การแจ้งเตือนได้เหมือนกัน ดังนั้นจึงสรุปได้ว่า ระบบที่ออกแบบขึ้นสามารถทำงานได้ตามวัตถุประสงค์ที่ต้องการ

#### 4.6 สรุปผลการทดลอง

การทดสอบการรับ-ส่งข้อมูลของ X-Bee ภาครับและภาคส่ง, การทดลองการรับข้อมูลของโมดูล X-Bee เมื่อมีการเปลี่ยนแปลงของขา X-Bee, การทดสอบการส่งข้อความของ ET-BASE GSM SIM900 ผ่านทางบอร์ดไมโครคอนโทรลเลอร์และการทดสอบการทำงานของระบบตู้ไปรษณีย์อัจฉริยะ เมื่อทำการทดลองทั้งหมดแล้วสรุปได้ว่า เมื่อนำมาประกอบกัน ตู้ไปรษณีย์อัจฉริยะจะทำงานได้ที่ระยะทางสูงสุด 120 เมตร ซึ่งระยะนี้ก็เพียงพอต่อการใช้งานจริงตามวัตถุประสงค์



## บทที่ 5

### สรุปผลการทดลองและข้อเสนอแนะ

#### 5.1 สรุปผลการทดลอง

โครงการนี้ได้ทำการศึกษาถึงการเขียนโปรแกรมควบคุมการทำงานของการทำงานของการรับ-ส่งข้อมูลทางการสื่อสารแบบไร้สายระหว่างโมดูล X-Bee Pro รุ่น Series 2 (ZB) จำนวน 2 ตัวในโหมด API เพื่อให้ผู้อาศัยภายในบ้านทราบว่า มีจดหมายมาส่งหรือมีจดหมายอยู่ในตู้ไปรษณีย์ที่หน้าบ้าน ทำให้เกิดความสะดวกสบายแก่ผู้อาศัยและประหยัดเวลาในการเดินมาดูจดหมายที่ตู้ไปรษณีย์หน้าบ้าน

จากการทดลอง พบว่าเมื่อเขียนโปรแกรมควบคุมการทำงานของ X-Bee Pro รุ่น Series 2 (ZB) ในโหมด API สามารถรับ-ส่งข้อมูลหากันได้ และสามารถตรวจจับการเปลี่ยนแปลงของเฟรมข้อมูลได้

จากการทดลองตู้ไปรษณีย์อัจฉริยะพบว่าสามารถแจ้งเตือนเจ้าของบ้านเมื่อมีจดหมายมาส่งทั้งในรูปแบบของเสียงเตือนและสัญญาณไฟและส่งข้อความผ่านมือถือโดยผู้ออกแบบได้ใช้เซนเซอร์ในการตรวจจับจดหมายที่ตู้ไปรษณีย์และส่งสัญญาณแบบไร้สายไปยังเครื่องรับที่อยู่ในบ้านเพื่อเตือนเจ้าของบ้าน ผลจากการทดลองได้แสดงให้เห็นว่า ตู้ไปรษณีย์อัจฉริยะที่ออกแบบขึ้นสามารถทำงานได้จริงตรงตามวัตถุประสงค์ของการออกแบบทุกประการ

#### 5.2 ปัญหาและอุปสรรค

ปัญหาและอุปสรรคที่เกิดขึ้นในระหว่างทำการทดสอบสามารถสรุปได้ดังนี้

1. ปัญหาเกี่ยวกับการตั้งค่า X-Bee
2. ปัญหาเกี่ยวกับการจ่ายกระแสไฟให้กับเซนเซอร์
3. วงจรบันทึกเสียงดิจิทัลใช้แหล่งจ่าย 12 โวลต์ บอร์ดไมโครคอนโทรลเลอร์สามารถจ่ายกำลังไฟฟ้าได้เพียง 5 โวลต์และบอร์ด ET-BASE GSM SIM900 ใช้แหล่งจ่าย 5 โวลต์ ทำให้ต้องใช้แหล่งจ่ายไฟ (Adapter) ถึง 3 ตัว

#### 5.3 ข้อเสนอแนะ

1. ที่ตู้ไปรษณีย์ใช้แหล่งจ่ายจากโซล่าเซลล์เพื่อประหยัดพลังงาน
2. ลดจำนวนของแหล่งจ่ายไฟของกล่องรับสัญญาณที่อยู่ในบ้าน

## บรรณานุกรม

- [1] <http://www.thaieasyelec.com/article-wiki/embedded-electronics-application/xbee-api-mode-tutorial-and-lab.html>
- [2] <http://www.pololu.com/product/959>
- [3] <http://ett.co.th/prod2013/et-base%20gsm%20sim900/et-base%20gsm%20sim900.html>
- [4] ผศ.ร.อ.ดร.ประ โยชน์ คำสวัสดิ์ คู่มือปฏิบัติการ 438301 ปฏิบัติการไมโครคอนโทรเลอร์



# ภาคผนวก ก



## โค้ดที่ใช้ในการทดลอง

```
/* Include Section */

#include <reg52.h>                // Standard 8052 SFR : File
#include <stdio.h>                // For printf I/O functions
#include <stdlib.h>
#include <string.h>

/* User Define Function */

void delay(unsigned long);       // Delay Time Function(1..4294967295)

void put_Char (char);

void SEND (void);

void getkey(void);

void SendSMS (void);

/* AT89C51AC3 SFR */

sfr CKCON = 0x8F;                // Clock Control

sbit SW      = P1^1;             // P1.0 = LED ON/OFF

sbit led      = P1^0;

unsigned char SEND_FRAM1[] = {    0x7E,0x00,0x10,0x17,0x01,0x00,0x13,0xA2
,0x00,0x40,0x86,0x5A,0xA6,0xFF,0xFE,0x02,0x44,0x32,0x03,0xF4,0x0D  };

unsigned char SEND_FRAM2[] = {    0x7E,0x00,0x10,0x17,0x01,0x00,0x13,0xA2
,0x00,0x40,0x86,0x5A,0xA6,0xFF,0xFE,0x02,0x49,0x43,0x04,0xDD,0x0D  };

char message_in[24],DataIn=0;

int Buff_Get=1,i=0,start_Message_In=0,Buff_Logic = 1;

/*****/

void main()

{

    CKCON = 0x01;                // Initial X2 Mode (BUS      Clock = 58.9824 MHz)

    TMOD &= 0x0F;                // Reset old Timer1 Mode Configure
```

```

TMOD |= 0x20;                // Update Timer1 = 8 Bit Auto Reload
SCON = 0x50;                // Serial Port Mode 1 (N,8,1)
ES = 1;                     // Disable Serial Interrupt
ET1 = 0;                   // Disable Timer1 Interrupt
PCON &= 0x7F;              // SMOD1 = 0 (Disable Double Baudrate)
TH1 = 0xF0;               // Setup Timer1 Baudrate 9600BPS / 58.9824 MHz
TL1 = 0xF0;
TR1 = 1;                   // Start Timer1 Generate Baudrate
TI = 0;                   // Set TI to send First char of UART

    RI = 0;
    EA = 1;
    SEND();
    led = 0;
    delay(5000);
    led = 1;
    delay(5000);
    led = 0;
    delay(5000);
    led = 1;
    delay(5000);
    led = 0;
    delay(5000);
    SW = 1;
    while(1)
    {
        getkey();
        if (message_in[20] == 0x00)
        {
            //led = 0;

```



```
if(Buff_Logic == 1)
{
    led = 1;

    SW = 0;

    delay(500);

    SW = 1;

    delay(50000);

    SW = 0;

    delay(500);

    SW = 1;

    delay(50000);

    SW = 0;

    delay(500);

    SW = 1;

    delay(50000);

    SendSMS();

    Buff_Logic = 0;
}

}

else
{

    if(Buff_Logic == 0)

    {

        SW = 1;

        Buff_Logic = 1;

    }

}
```

```

    }

}

/*****

/*  getkey 16 Chr  */

*****/

void getkey(void)
{
    int i;

    //Buff_Get = 1;

    //led = 0;

    //----- message 1 -----
    strcpy(message_in,"");
    for(i=0;i<22;i++)
    {
        Buff_Get = 1;
        while(Buff_Get){led = 0;} // Clear and wait next ASCII
        //led = 1;
        message_in[i] = DataIn;
    }

    led = 1;

    return;

}

*****/

/* Print Data(ASCII) to LCD */

*****/

void SEND (void)
{

```

```

char *p;

p = SEND_FRAM1;

do
    // Get ASCII & Write to LCD Until null
{
    put_Char(*p);
    // Write ASCII to LCD

    p++;
    // Next ASCII
}

while(*p != 0x0D);
// End of ASCII (null)

p = SEND_FRAM2;

do
    // Get ASCII & Write to LCD Until null
{
    put_Char(*p);
    // Write ASCII to LCD

    p++;
    // Next ASCII
}

while(*p != 0x0D);
// End of ASCII (null)

TI = 1;

RI = 0;

return;
}

```

```

/*****/

```

```

void put_Char (unsigned char CH)

```

```

{

    SBUF = CH;

    while(!TI);

    TI = 0;

    return;

}

```

```

/*****/

```

```

/*      INTERRUPT EXTERNAL (IN0)      */

/*****

void Interrupt_RS232 (void) interrupt 4

{

    if (RI == 1)

    {
        RI = 0;

        DataIn = SBUF;

        Buff_Get = 0;

        //SBUF = DataIn;

    }

    return;

}

/*****

/* Long Delay Time Function(1..4294967295) */

/*****

void delay(unsigned long i)

{

    while(i > 0) {i--;}

    return;

}

/*****

/*      SendSMS      */

/*****

void SendSMS()

{

    printf("AT\n");

    delay(2000);

    printf("AT+CMGF=1\n");

```

```
    delay(2000);

    printf("AT+CSMP=17,167,0,0\n");

    delay(2000);

    printf("AT+CMGS=\"+66804653615\"\n");

    delay(2000);

    printf("You've got a letter in your mailbox!\n");           // Output Message

    putchar(0x1A);

    printf("\n");

    putchar(0x0A);

    putchar(0x0D);

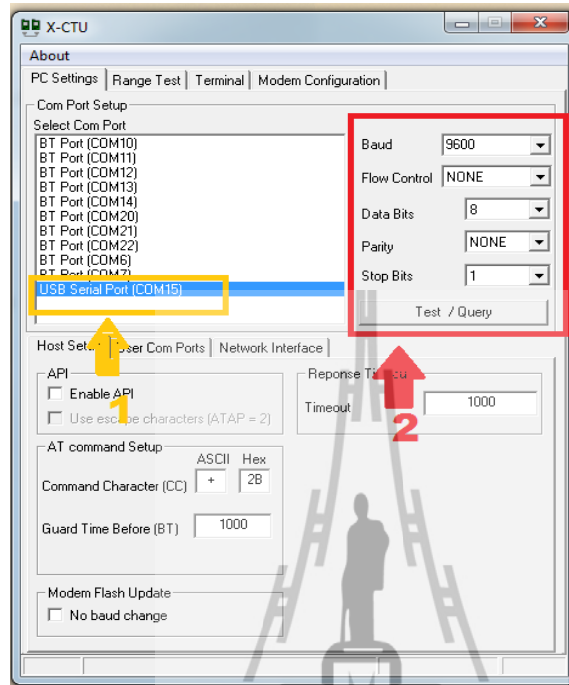
    return;
}
```



# ภาคผนวก ข

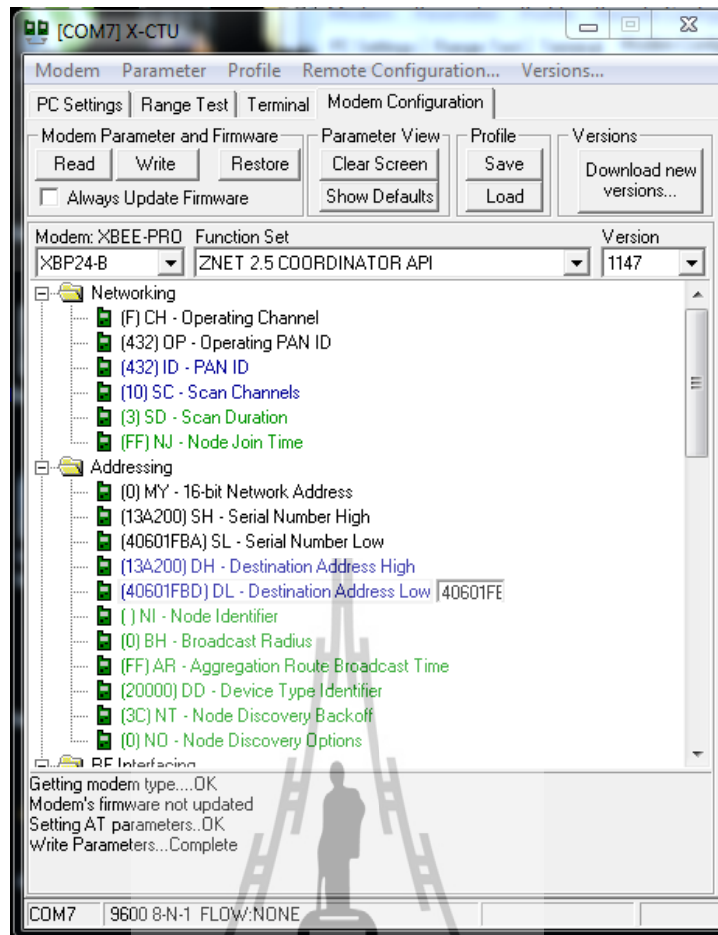


- การเขียนเฟิร์มแวร์เป็นภาคส่งในโหมด API (Write Firmware Coordinator API)
  1. ทำการเปิดโปรแกรม X-CTU เลือกComport ที่เชื่อมอยู่กับ X-Bee ตัวที่เป็นภาครับ (Router)
  2. ทำการตั้งค่า Baud rate และค่าต่าง ๆ ดังรูปที่ ข.1



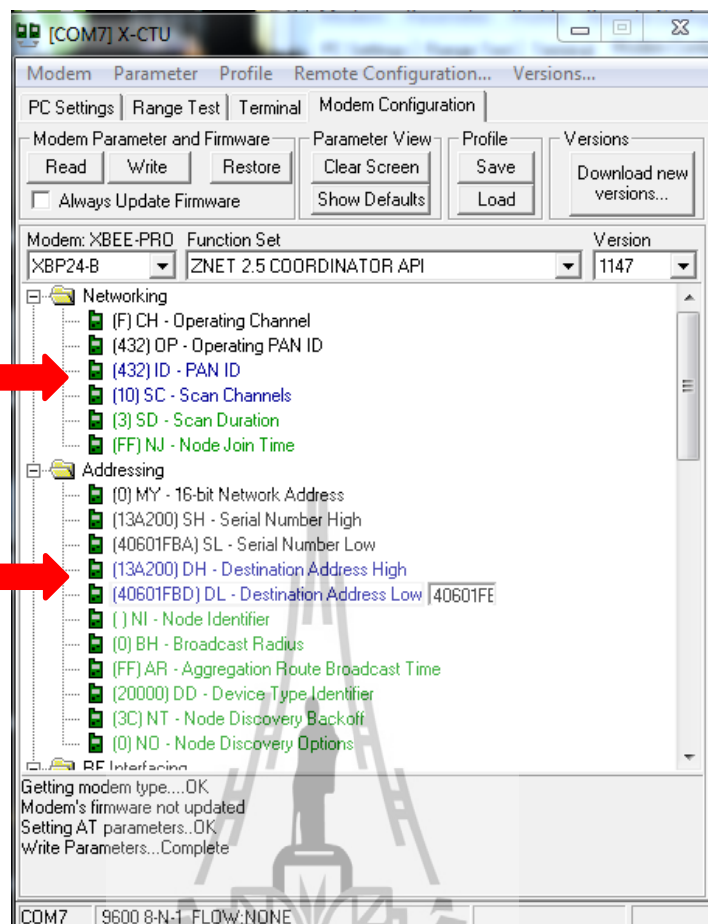
รูปที่ ข.1 การตั้งค่า Baud rate

3. ทำการเลือก Modem Configuration ดังแสดงในรูปที่ ข.2
4. ทำการเลือก Modem X-Bee ให้ตรงกับรุ่นที่ใช้
5. ทำการเลือก Function Set เป็น Coordinator API
6. ทำการคลิกที่ Button Clear Screen
7. ทำการเลือกเครื่องหมายถูกที่ Always update firmware
8. จากนั้นทำการคลิก Write
9. รอสักครู่จนกว่าเขียนเฟิร์มแวร์เสร็จ



รูปที่ ข.2 การเขียน โมดูล X-Bee เป็นภาคส่ง (Coordinator) ในโหมด API

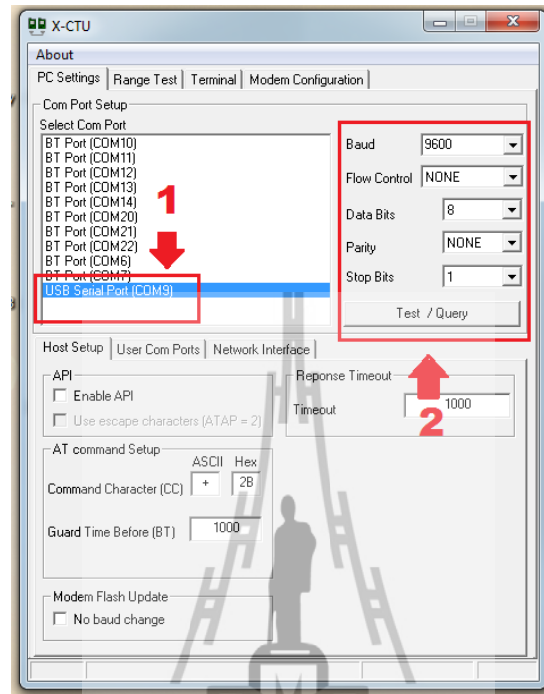
10. กลับมาที่หน้าต่าง PC setting
11. ทำการเลือกเครื่องหมายถูกที่ Enable API
12. แล้วกลับไปยังหน้า Modem Configure
13. ทำการคลิกที่ Show Defaults
14. ทำการเลือกเครื่องหมายถูกที่ Always Update ออก
15. คลิก Write
16. เมื่อทำการเขียนเฟิร์มแวร์เสร็จก็ทำการกด Read เพื่ออ่านข้อมูล แล้วทำการกำหนดค่า Destination ที่เราต้องการจะส่งเฟรมข้อมูล ในที่นี้เป้าหมายที่เราจะส่งค่าไปคือ ภาครับ (Router) ซึ่งมี Serial Number High (SH) = 0013A200 Serial Number Low (SL) = 40801FBD ใส่ค่าใน DH และ DL ตามลำดับดังแสดงในรูปที่ ข.3
17. ทำการตั้งค่า Scan channels(SC) = 10, ค่า PAN ID = 432 และ Power level = 4 (17 dBm) ดังแสดงในรูปที่ ข.3
18. ทำการคลิก Write อีกครั้ง เสร็จการตั้งค่าเป็นภาคส่งในโหมด API (Coordinator API)



รูปที่ ข.3 การตั้งค่าปลายทางที่ภาคส่งจะทำการส่งเฟรมข้อมูลออกไป

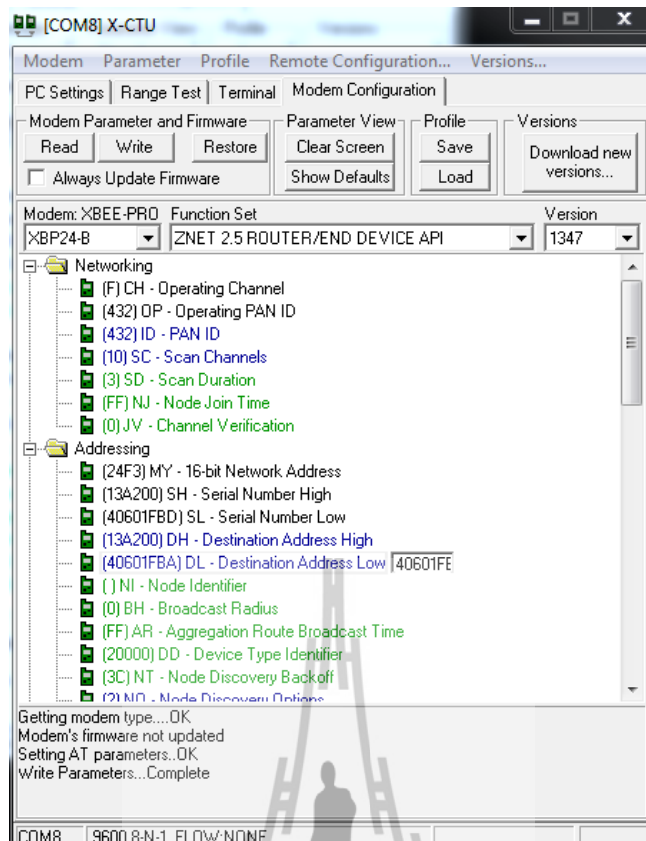
- การเขียนเฟิร์มแวร์เป็นภาครับในโหมด API (Write Firmware Router/End Device API)

1. ทำการเปิดโปรแกรม X-CTU เลือก Comport ที่เชื่อมต่ออยู่กับ X-Bee ของภาครับ (Router)
2. ทำการตั้งค่า Baud rate และค่าต่างๆดังแสดงในรูปที่ ข.4



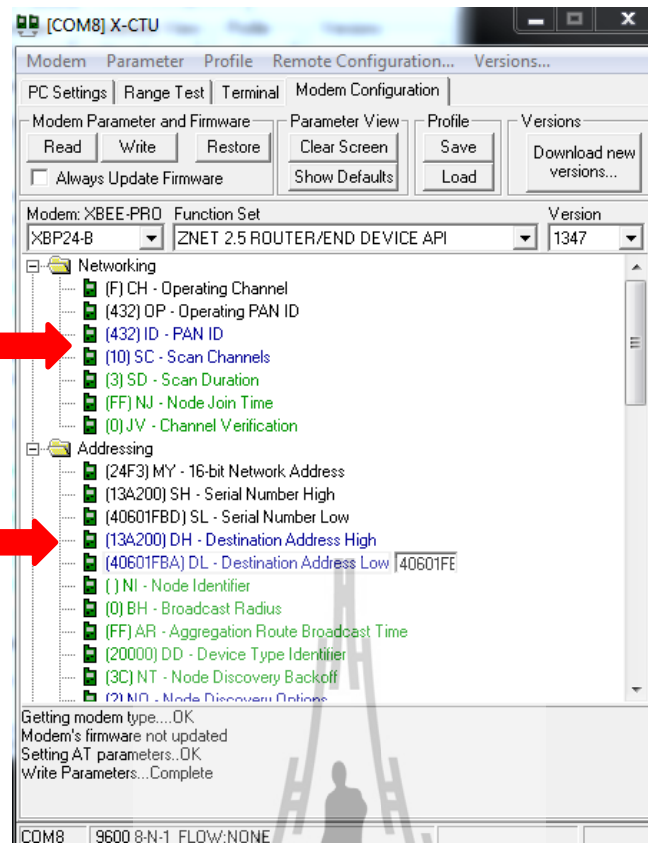
รูปที่ ข.4 การตั้งค่า Baud rate

3. ทำการเลือก Modem Configurationดังแสดงในรูปที่ ข.5
4. ทำการเลือก Modem X-Bee ให้ตรงกับรุ่นที่ใช้อยู่
5. จากนั้นทำการเลือก Function Set เป็น Router/End Device API
6. คลิกที่ Button Clear Screen
7. ทำการเลือกเครื่องหมายถูกที่ Always update firmware
8. จากนั้นทำการคลิกที่ Write รอสักครู่จน
9. เมื่อทำการ write เสร็จจะปรากฏ ดังแสดงในรูปที่ ข.5



รูปที่ ข.5 การตั้งค่าโมดูล X-Bee เป็นภาครับในโหมด API (Router/End Device API)

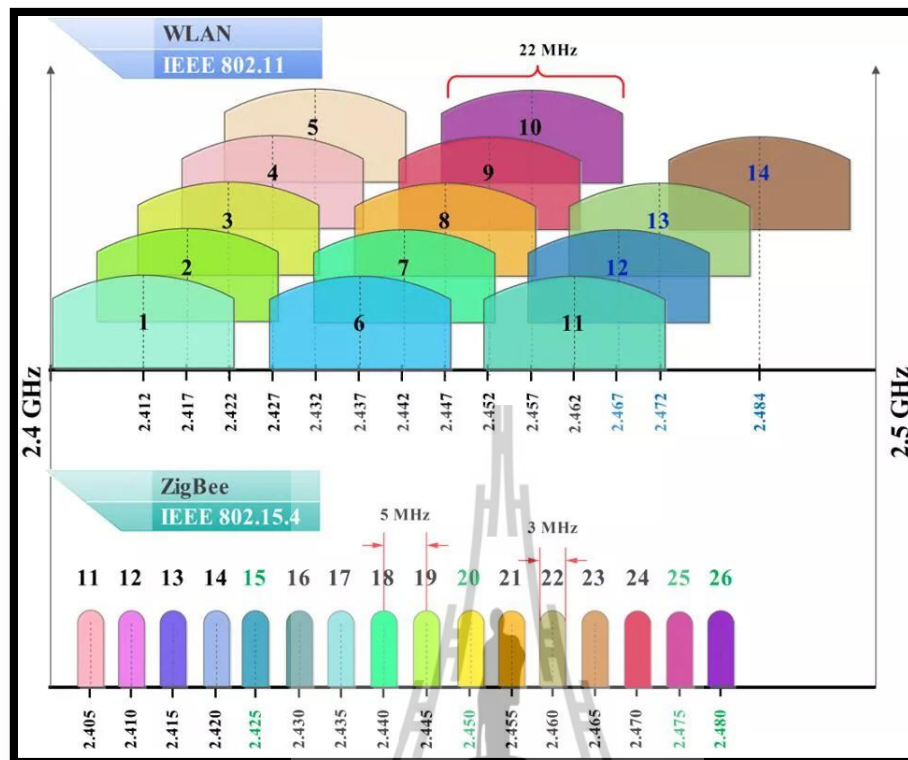
10. แล้วกลับมาที่หน้าต่าง PC setting
11. ทำการเลือกเครื่องหมายถูกที่ Enable API
12. แล้วกลับไปยังหน้า Modem Configure
13. คลิกที่ Show Defaults
14. ทำการเลือกเครื่องหมายถูกที่ Always Update ออก
15. คลิกที่ Write
16. เมื่อทำการ write เสร็จก็กดที่ Read เพื่ออ่านข้อมูล แล้วทำการกำหนดค่า Destination ที่เราต้องการรับเฟรมข้อมูล ในที่นี้รับข้อมูลจากภาคส่ง (coordinator) ซึ่งมี Serial Number High (SH) = 0013A200 Serial Number Low (SL) = 40601FBA ใส่ค่าใน DH และ DL ตามลำดับ ดังแสดงในรูปที่ ข.6
17. ทำการตั้งค่า Scan channels(SC) = 10, ค่า PAN ID = 432 และ Power level = 4 (17 dBm)
18. คลิกที่ Write อีกครั้ง เสร็จการตั้งค่าเป็นภาครับในโหมด API (Router/End Device API)



รูปที่ ข.6 การตั้งค่าปลายทางที่ภาครับจะทำการรับเฟรมข้อมูลที่ส่งมาจากภาคส่ง



- การเลือกใช้ Channels ของ X-Bee ในย่านความถี่ 2.4 GHz



รูปที่ ข.7 ช่องความถี่ที่ใช้สำหรับย่านความถี่ 2.4 GHz

สามารถเลือกใช้ได้ 4 channels คือ channels 15,20,25 และ 26 เนื่องจากช่องสัญญาณทั้ง 4 ช่องนั้นจะเป็นการลดปัญหาของการรบกวนกันของอุปกรณ์ X-Bee และ อุปกรณ์ที่ใช้งานด้าน WLAN เช่น Wifi เป็นต้น

## ประวัติผู้เขียน



นายเฉลิมศักดิ์ อรุณโรจน์ปัญญา เกิดเมื่อวันที่ 15 สิงหาคม พ.ศ. 2535 ภูมิลำเนาอยู่ที่ 403 หมู่ 1 ตำบลลำปลายมาศ อำเภอลำปลายมาศ จังหวัดบุรีรัมย์ สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลายจากโรงเรียนมารีย์วิทยานครราชสีมา อำเภอมือง จังหวัดนครราชสีมา ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา



นายกิตติวัฒน์ สุขหงษ์ เกิดเมื่อวันที่ 5 กรกฎาคม พ.ศ. 2535 ภูมิลำเนาอยู่ที่ 65 หมู่ 1 ตำบลบางแพ อำเภอบางแพ จังหวัดราชบุรี สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลายจากโรงเรียนเบญจมราชูทิศ ราชบุรี อำเภอมือง จังหวัดราชบุรี ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา



นายกัณฑ์ธร กิรติเพียรเลิศ เกิดเมื่อวันที่ 6 มิถุนายน พ.ศ. 2536 ภูมิลำเนาอยู่ที่ 340/6 หมู่บ้านราชธานี ตำบลหมื่นไวย อำเภอมือง จังหวัดนครราชสีมา สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลายจากโรงเรียนมารีย์วิทยานครราชสีมา อำเภอมือง จังหวัดนครราชสีมา ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา