สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การออกแบบและสร้างชุดกระจายเสียงแบบดิจิตอล
DESIGN AND DISTRIBUTES DIGITAL BROADCAST

T104325

โดย

นางสาว	ขวัญใจ	ฤทธิ์จอหอ	รหัส 49015046
นาย	สมภพ	เทียนทอง	รหัส 49015075
นาย	เอกนรินทร์	ทาไสย	รหัส 49015083

ส/ว.
ว 259 ก
2551

เลขหมู่.....
เลขทะเบียน 104325
วัน,เดือน,ปี - 2 พ.ย. 2552

b 12110383
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2551

ภาควิชา
วิศวกรรมโทรคมนาคม

ผ่านการตรวจรูปเล่มแล้ว
(ลงชื่อ).....ผู้ตรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบและสร้างชุดกระจายเสียงแบบดิจิตอล
DESIGN AND DISTRIBUTES DIGITAL BROADCAST

โดย

นางสาวขวัญใจ	ฤทธิ์จอหอ	49015046
นายสมภพ	เทียนทอง	49015075
นายเอกนรินทร์	ทาไสย	49015083

อาจารย์ที่ปรึกษา

รศ. เกรียงไกร วงศ์โรจนภรณ์

รศ.ดร. สุวิพล สิทธิชีวภาค

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2551

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การออกแบบและสร้างชุดกระจายเสียงแบบดิจิตอล
DESIGN AND DISTRIBUTES DIGITAL BROADCAST

ผู้จัดทำ

1. นางสาวขวัญใจ ฤทธิ์จอหอ รหัส 49015046
2. นายสมภพ เทียนทอง รหัส 49015075
3. นายเอกนรินทร์ ทาไสย รหัส 49015083

_____ อาจารย์ที่ปรึกษา

(รศ. เกรียงไกร วงศ์โรจนภรณ์)

_____ อาจารย์ที่ปรึกษา

(รศ.ดร. สุวิพล สิทธิชีวภาค)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงงานเลขที่ 512507

การออกแบบและสร้างชุดกระจายเสียงแบบดิจิตอล
DESIGN AND DISTRIBUTES DIGITAL BROADCAST

โดย

นางสาววัญใจ	ฤทธิ์จอหอ	รหัส 49015046
นายสมภพ	เทียนทอง	รหัส 49015075
นายเอกนรินทร์	ทาไสย	รหัส 49015083

อาจารย์ที่ปรึกษา รศ. เกรียงไกร วงศ์โรจนภรณ์

รศ.ดร. สุวิพล สิทธิชีวภาค

บทคัดย่อ

โครงงานนี้เป็นการออกแบบและสร้างชุดกระจายเสียงแบบดิจิตอลโดยอาศัยอาร์เอฟโมดูลในการรับและส่ง มีระยะในการรับส่งประมาณ 0.5 – 1 กิโลเมตร ในระยะไลน์ออฟไซด์ โดยใช้ย่านความถี่ 2.4 GHz ในหนึ่งชุดจะสามารถทำงานได้ทั้งรับและส่ง ในส่วนของวงจรภาคส่งได้รับการออกแบบให้สามารถส่งสัญญาณเสียงในรูปแบบดิจิตอลโดยใช้ไมโครคอลโทรเลอร์เป็นตัวควบคุมและสามารถกำหนดช่องสัญญาณในการส่งได้ ในส่วนของวงจรภาครับได้รับการออกแบบให้สามารถระบุแอดเดรสประจำเครื่องได้และเมื่อรับสัญญาณเข้ามาก็จะถูกส่งออกสู่ลำโพงต่อไป

ABSTRACT

This project aim's at designing Digital Broadcast System which uses RF module for sending and receiving signals at the approximate visible distance of 0.5 to 1 kilometer and at the range of 2.4 GHz frequency. Each set functions as all-in-one transmitter. The signal-sending circuit is designed to send digital audio signal by using microcontroller to control and set the channel.The signal-receiving circuit, is received, it will then be send to the speakers.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์ 1

1.2 วัตถุประสงค์ของปริญญานิพนธ์ 1

1.3 ขอบเขตของปริญญานิพนธ์ 1

1.3.1 ขอบเขตของอุปกรณ์ทางด้านฮาร์ดแวร์ 1

1.3.2 ขอบเขตทางด้านซอฟต์แวร์ 2

1.4 ประโยชน์ที่คาดว่าจะได้รับจากปริญญานิพนธ์ 2

1.5 แผนการดำเนินงาน 2

บทที่ 2 ทฤษฎีและหลักการ

2.1 การสื่อสารไร้สาย 3

2.2 ความรู้ทั่วไปเกี่ยวกับเครือข่าย Zigbee 4

2.2.1 มาตรฐาน IEEE 802.15.4 และ Layer 5

2.2.1.1 Network layer 7

2.2.1.2 Application layer 7

2.2.1.3 Media access control (MAC) layer 7

2.2.1.4 Physical layer 9

2.2.2 ความรู้ทั่วไปเกี่ยวกับ AODV Protocol 9

2.2.2.1 Sequence Numbers 12

2.2.2.2 Error Messages 12

2.2.2.3 สรุปคุณสมบัติของ AODV 13

2.2.3 Received Signal Strength Indication(RSSI) 14

2.3 ความรู้ทั่วไปเกี่ยวกับโมดูล Xbee Pro 14

2.3.1 คุณสมบัติโดยทั่วไปของ Zigbee 15

2.3.2 การส่งข้อมูลในรูปแบบของ API 17

2.3.2.1 รูปแบบของ API Frame 17

2.3.2.2 ประเภทของ API 18

2.3.2.3 AT Command 18

2.3.2.4 AT Command Queue Parameter Value 19

2.3.2.5 AT Command Response 19

2.3.2.6 Tx (Transmit) Request : 64-bit address 20

2.3.2.7 Tx (Transmit) Request : 16-bit address 20

2.3.2.8 Tx (Transmit) Status 21

2.3.2.9 RX (Receive) Packet 64-bit address 21

2.3.2.10 RX (Receive) Packet 16-bit address 22

2.3.2.11 ความปลอดภัยของ Zigbee 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.4 การใช้โปรแกรม X-CTU	24
2.4.1 รีจิสเตอร์ที่ควรทราบของโมดูล XBee-PRO	27
2.5 ไมโครคอนโทรลเลอร์ AVR	28
2.5.1 ขาพอร์ตอินพุตเอาต์พุต	30
2.5.2 การควบคุมพอร์ตไมโครคอนโทรลเลอร์ AVR	31
2.5.2.1 รายละเอียดขาพอร์ตไมโครคอนโทรลเลอร์	32
2.5.2.2 การใช้งานพอร์ตอินพุตเอาท์พุตของ AVR	34
2.5.2.3 ความสัมพันธ์ระหว่างรีจิสเตอร์ DDRx กับบิต PUD	34
2.5.3 การสื่อสารข้อมูลอนุกรมผ่านโมดูล USART	35
2.5.3.1 คุณสมบัติที่สำคัญของโมดูล USART	37
2.5.3.2 รีจีสเตอร์ที่เกี่ยวข้องกับการทำงานในโหมด USART	37
2.5.4 การเชื่อมต่ออุปกรณ์อนุกรม (SPI)	38
2.5.4.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานในโหมด SPI	39
2.5.4.2 หลักการสำคัญของการใช้ SPI	40
2.5.5 การเชื่อมต่ออุปกรณ์อนุกรมด้วยสาย 2 เส้น (TWI)	40
2.5.5.1 การสื่อสารข้อมูลบนระบบบัส I^2C	40
2.5.5.2 การสื่อสารข้อมูลบนระบบบัส I^2C กับโมดูล TWI	41
2.5.6 การใช้งานโมดูลแปลงสัญญาณอะนาลอกเป็นดิจิตอล	42
2.5.7 การสื่อสารอนุกรมแบบ RS-232	43
2.6 การแปลงสัญญาณอนาล็อก-ดิจิตอล (Analog to Digital Converter)	44
2.7 การแปลงสัญญาณทางดิจิตอลเป็นอะนาล็อก (Digital to Analog Converter)	46
2.8 การรับส่งข้อมูลแบบอนุกรม	50
บทที่ 3 การออกแบบและการสร้าง	
3.1 การออกแบบและสร้างชุดกระจายเสียงแบบดิจิตอลในส่วนของฮาร์ดแวร์	53
3.1.1 อธิบายชุดรับ – ชุดส่ง	53
3.1.1.1 ขณะชุดกระจายเสียงเป็นชุดส่ง	53
3.1.1.2 ขณะชุดกระจายเสียงเป็นชุดรับ	54
3.1.2 การออกแบบและสร้างบอร์ดควบคุม	54
3.1.3 วงจรปรีไมค์ (Sounder detector)	57
3.2 การออกแบบและการเขียนโปรแกรมในส่วนของซอฟแวร์	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 4 การทดลองและผลการทดลอง	
4.1 การวัดระดับความแรงของสัญญาณ	60
4.2 การทดลองวัดความผิดพลาดในการรับ – ส่งข้อมูลโดยใช้โมดูล Zigbee เป็นตัวรับส่งข้อมูล	65
4.3 การทดลองการส่งข้อมูลที่เป็นเสียงพูดแสดงออกทาง X–CTU ทางฝั่งรับ	67
4.4 การทดลองวัดสัญญาณต่างๆ	71
4.4.1 การวัดรูปสัญญาณในส่วนของการรับ–ส่งข้อมูลระหว่างโมดูลโดยการส่งข้อมูลเป็นรหัสแอสกี้	71
4.4.2 การทดลองวัดรูปสัญญาณการส่งข้อมูลเป็นอักษร A โดยที่อัตราเร็วในการส่งเปลี่ยนแปลงไป	75
4.4.3 การทดลองวัดรูปสัญญาณในส่วนของสัญญาณอินพุตที่เป็นเสียงพูด	80
4.4.4 การวัดรูปสัญญาณในส่วนของฝั่งส่งที่รับข้อมูลมาจากปรีไมค์	81
4.4.5 การทดลองวัดรูปสัญญาณทางฝั่งรับที่ได้รับข้อมูลจากฝั่งส่ง	83
4.4.6 การทดลองวัดรูปสัญญาณทางด้านเอาต์พุต	85
บทที่ 5 บทสรุปและวิจารณ์	
5.1 สรุปผลการทดลอง	86
5.1.1 ผลการทดลองในการวัดสัญญาณเปรียบเทียบกับระยะทาง	86
5.1.1.1 ผลการทดลองในการวัดระดับความแรงของสัญญาณระยะทางต่าง	86
5.1.1.2 ผลการทดลองในการวัดค่าความผิดพลาดเปรียบเทียบกับระยะทาง	86
5.1.2 ผลการทดลองในการส่งข้อมูลที่เป็นเสียงพูดแสดงออกทาง X–CTU ทางฝั่งรับ	86
5.1.3 ผลการวัดรูปสัญญาณโดยใช้ออสซิโลสโคป	86
5.1.3.1 ผลการวัดรูปสัญญาณจากบอร์ดควบคุม Zigbee	86
5.1.3.2 ผลการวัดรูปสัญญาณจากบอร์ดควบคุมไมโครคอนโทรลเลอร์	87
5.2 ปัญหาและแนวทางแก้ไข	87
5.2.1 ปัญหาเกี่ยวกับการออกแบบลายวงจรและการลงอุปกรณ์	87
5.2.2 ปัญหาเกี่ยวกับทางด้านเอาท์พุต	87
ภาคผนวก	
กิตติกรรมประกาศ	
เอกสารอ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 2.1 การตั้งขอบเขตให้กับ node ต่างๆ ใน wireless network	9
รูปที่ 2.2 การทำงานของ AODV Protocol	10
รูปที่ 2.3 การทำงานของ AODV Protocol (ต่อ)	11
รูปที่ 2.4 วิธีการใช้ Sequence Number	12
รูปที่ 2.5 สาเหตุของการเกิด ERROR Message	13
รูปที่ 2.6 ตัวโมดูล Zigbee	14
รูปที่ 2.7 โครงข่ายของเครือข่าย Zigbee	16
รูปที่ 2.8 การเชื่อมต่อแบบต่างๆของ Zigbee	16
รูปที่ 2.9 รูปแบบของ API Packet frame Mode 1	17
รูปที่ 2.10 รูปแบบของ API Packet frame	18
รูปที่ 2.11 รูปแบบของ API Packet frame ประเภท AT Command	18
รูปที่ 2.12 รูปแบบของ API Packet frame ประเภท AT Command Queue Parameter Value	19
รูปที่ 2.13 รูปแบบของ API Packet frame ประเภท AT Command Response	19
รูปที่ 2.14 รูปแบบของ API Packet frame ประเภท Tx Request: 64-bit address	20
รูปที่ 2.15 รูปแบบของ API Packet frame ประเภท Tx Request: 16-bit address	20
รูปที่ 2.16 รูปแบบของ API Packet frame ประเภท Tx	21
รูปที่ 2.17 รูปแบบของ API Packet frame ประเภท RX Packet 64-bit address	21
รูปที่ 2.18 รูปแบบของ API Packet frame ประเภท RX Packet 16-bit address	22
รูปที่ 2.19 เฟรม MAC กับการรักษาความปลอดภัย	23
รูปที่ 2.20 การรักษาความปลอดภัยใน Network layer	23
รูปที่ 2.21 การรักษาความปลอดภัยใน Application layer	23
รูปที่ 2.22 โปรแกรม X-CTU	24
รูปที่ 2.23 การพบโมดูล Zigbee	24
รูปที่ 2.24 โปรแกรม X-CTU	25
รูปที่ 2.25 โปรแกรม X-CTU (ต่อ)	25
รูปที่ 2.26 โปรแกรม X-CTU (ต่อ)	26
รูปที่ 2.27 โปรแกรม X-CTU (ต่อ)	26
รูปที่ 2.28 บล็อกไดอะแกรม AVR (ATmega16)	29
รูปที่ 2.29 ตัวถังแบบ PDIP และ TQFP/MLF และขาพอร์ตAVR (ATmega16)	30
รูปที่ 2.30 ขาพอร์ตไมโครคอนโทรลเลอร์ AVR	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 2.31 บล็อกดะแกรมของโมดูล USART	36
รูปที่ 2.32 ลักษณะของเฟรมข้อมูลอนุกรม	37
รูปที่ 2.33 โครงสร้างบล็อกไดอะแกรมการรับส่งข้อมูลแบบ SPI	38
รูปที่ 2.34 การใช้งานโมดูล SPI ที่ใช้สายสัญญาณทั้ง 4 เส้น	39
รูปที่ 2.35 รูปแบบของบิตข้อมูลบนโมดูล TWI	41
รูปที่ 2.36 รูปแบบของสัญญาณ	41
รูปที่ 2.37 วงจรโมดูล ADC	43
รูปที่ 2.38 การทำงานของ Counting Converter	45
รูปที่ 2.39 Output voltage ของ A/D converter	45
รูปที่ 2.40 ระบบการแปลงสัญญาณดิจิตอลเป็นอนาลอก	46
รูปที่ 2.41 Transfer Curve ในอุดมคติของ DAC 3 บิต	46
รูปที่ 2.42 คลื่นไซน์ที่สร้างจาก DAC	47
รูปที่ 2.43 Summed Source DAC	48
รูปที่ 2.44 Switched Voltage R-2R DAC	49
รูปที่ 2.45 Switched Current R-2R DAC	50
รูปที่ 2.46 การใช้มาตรฐาน RS-232เชื่อมต่ออุปกรณ์	51
รูปที่ 3.1 บล็อกไดอะแกรมของชุดกระจายเสียงแบบดิจิตอล	53
รูปที่ 3.2 ลายวงจร	54
รูปที่ 3.3 การลงอุปกรณ์	55
รูปที่ 3.4 ภายในของชุดกระจายเสียง	55
รูปที่ 3.5 ภายนอกของชุดกระจายเสียง	56
รูปที่ 3.6 ชุดวงจรที่สมบูรณ์ขณะทำงาน	56
รูปที่ 3.7 วงจรปรีไมค์	57
รูปที่ 3.8 ชุดปรีไมค์	57
รูปที่ 3.9 ผังงานแสดงการเขียนโปรแกรมควบคุมทางภาคส่ง	58
รูปที่ 3.10 ผังงานแสดงการเขียนโปรแกรมควบคุมทางภาครับ	59
รูปที่ 4.1 หน้าต่างที่ใช้วัดความแรงของสัญญาณ	60
รูปที่ 4.2 กราฟเปรียบเทียบความแรงกับระยะทางของโมดูลตัวที่ 4	63
รูปที่ 4.3 กราฟเปรียบเทียบความแรงกับระยะทางของโมดูลตัวที่ 3	64
รูปที่ 4.4 กราฟเปรียบเทียบความแรงกับระยะทางของโมดูลตัวที่ 3และ 4	64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.5 หน้าต่างของ Range test ที่ใช้วัดความผิดพลาดการรับ – ส่ง	65
รูปที่ 4.6 กราฟเปรียบเทียบค่าความผิดพลาดกับระยะทาง	66
รูปที่ 4.7 ฝั่งส่งข้อมูลเสียง	67
รูปที่ 4.8 ฝั่งรับข้อมูล	67
รูปที่ 4.9 ระดับข้อมูลในขณะที่ยังไม่ได้รับสัญญาณเสียง	68
รูปที่ 4.10 ระดับข้อมูลในขณะที่ยังไม่ได้รับสัญญาณเสียง	69
รูปที่ 4.11 การดึงข้อมูลจากทางฝั่งส่งแสดงออกจอ	70
รูปที่ 4.12 ตำแหน่งการจับสัญญาณ	71
รูปที่ 4.13 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น Aแสดงเป็น binary จะได้เท่ากับ 01000001	71
รูปที่ 4.14 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น E แสดงเป็น binary จะได้เท่ากับ 01000101	72
รูปที่ 4.15 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น I แสดงเป็น binary จะได้เท่ากับ 01001001	72
รูปที่ 4.16 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น Q แสดงเป็น binary จะได้เท่ากับ 01010001	73
รูปที่ 4.17 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น U แสดงเป็น binary จะได้เท่ากับ 01010101	73
รูปที่ 4.18 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น Y แสดงเป็น binary จะได้เท่ากับ 01011001	74
รูปที่ 4.19 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น 4 แสดงเป็น binary จะได้เท่ากับ 00110100	74
รูปที่ 4.20 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น 9 แสดงเป็น binary จะได้เท่ากับ 00111001	75
รูปที่ 4.21 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 1200 bps โดยการส่งอักษรA	75
รูปที่ 4.22 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 2400 bps โดยการส่งอักษรA	76
รูปที่ 4.23 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 4800 bps โดยการส่งอักษรA	76
รูปที่ 4.24 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 9600 bps โดยการส่งอักษรA	77
รูปที่ 4.25 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 19200 bps โดยการส่งอักษรA	77
รูปที่ 4.26 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 38400 bps โดยการส่งอักษรA	78
รูปที่ 4.27 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 57600 bps โดยการส่งอักษรA	78
รูปที่ 4.28 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 115200 bps โดยการส่งอักษรA	79

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 4.29 จุดวัดสัญญาณอินพุท 80
รูปที่ 4.30 รูปสัญญาณเสียงพูดที่จะป้อนเป็นอินพุท 80
รูปที่ 4.31 จุดวัดสัญญาณในส่วนของฝั่งส่งที่รับข้อมูลมาจากปรีไมค์ 81
รูปที่ 4.32 สัญญาณที่วัดได้จากไมโครคอนโทรลเลอร์ 81
รูปที่ 4.33 จุดวัดสัญญาณที่ขา TX ของ Zigbee 82
รูปที่ 4.34 สัญญาณที่วัดได้จากขา TX ของ Zigbee 82
รูปที่ 4.35 จุดวัดสัญญาณที่ขา RX ของ Zigbee 83
รูปที่ 4.36 สัญญาณดิจิตอลที่รับได้จากขา RX 83
รูปที่ 4.37 จุดวัดสัญญาณก่อนที่จะเข้าไมโครคอนโทรลเลอร์ 84
รูปที่ 4.38 สัญญาณที่ก่อนส่งเข้าไมโครคอนโทรลเลอร์ 84
รูปที่ 4.39 จุดวัดสัญญาณทางด้านเอาต์พุต 85
รูปที่ 4.40 สัญญาณที่วัดได้จากด้านเอาต์พุต 85

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 1.1 ตารางแผนการดำเนินงาน	2
ตารางที่ 2.1 IEEE 802.15.4 frequency bands and data transfer rates	6
ตารางที่ 2.2 ขาพอร์ต PORTA (PA0 – PA7)	32
ตารางที่ 2.3 ขาพอร์ต PORTB (PB0 – PB7)	33
ตารางที่ 2.4 ขาพอร์ต PORTC (PC0 – PC7)	33
ตารางที่ 2.5 ขาพอร์ต PORTD (PD0 – PD7)	34
ตารางที่ 2.6 การกำหนดรูปแบบของขาพอร์ต	35
ตารางที่ 2.7 ชื่อขาสัญญาณของ DB-9	52
ตารางที่ 4.1 ระดับความแรงของสัญญาณ	62
ตารางที่ 4.2 ระดับความแรงของสัญญาณที่หาค่าเฉลี่ยแล้วในรูปฐาน 16	62
ตารางที่ 4.3 ระดับความแรงของสัญญาณที่หาค่าเฉลี่ยแล้วในรูปฐาน 10	63
ตารางที่ 4.4 ข้อมูลการวัดความผิดพลาดในการรับส่งข้อมูล	66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์

ในปัจจุบันเทคโนโลยีได้มีการพัฒนาออกไปอย่างกว้างขวางโดยเฉพาะเทคโนโลยีที่ใช้ในด้านการสื่อสารซึ่งมีความสำคัญอย่างยิ่งกับมนุษย์ ทำให้ในปัจจุบันการสื่อสารได้มีการพัฒนาไปอย่างมากจึงทำให้มนุษย์มีความสะดวกสบายมากขึ้นทั้งนี้เพราะว่าการสื่อสารเป็นสิ่งสำคัญที่ใช้ในการเชื่อมโยงระหว่างกิจกรรมของมนุษย์ เทคโนโลยีจึงมีส่วนช่วยให้ประหยัดเวลาและประหยัดค่าใช้จ่ายได้

ในปัจจุบันระบบการกระจายเสียงแบบอนาลอกจะมีสัญญาณรบกวนจากภายนอกเกิดขึ้นสูงทั้งตัวอุปกรณ์เอง ความถี่อื่นๆ สภาพภูมิอากาศ เป็นต้นจึงทำให้เกิดการผิดเพี้ยนของสัญญาณนั้นได้ง่ายอีกทั้งความแม่นยำในการส่งข้อมูลก็ไม่แม่นยำเท่าที่ควรดังนั้นจึงได้มีการคิดระบบกระจายเสียงแบบดิจิตอลขึ้นเพื่อลดปัญหาเรื่องสัญญาณรบกวนที่เกิดขึ้นและช่วยเพิ่มความแม่นยำในการรับส่งข้อมูลโดยในชุดกระจายเสียงแบบดิจิตอลนั้นได้มีการประยุกต์นำเอาเทคโนโลยีโมดูลไร้สายมาใช้เพื่อรับและส่งสัญญาณไร้สายด้วย

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1.2.1 เพื่อศึกษาการออกแบบและสร้างชุดกระจายเสียงไร้สายแบบดิจิตอล

1.2.2 เพื่อศึกษาการทำงานของโมดูลไร้สายสำหรับการกระจายเสียงแบบดิจิตอล

1.2.3 เพื่อศึกษาการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

1.2.4 เพื่อศึกษาการแปลงสัญญาณดิจิตอลเป็นสัญญานอนาลอก

1.2.5 เพื่อศึกษาและทำความเข้าใจเกี่ยวกับการเขียนโปรแกรมภาษาซีเพื่อกำหนดการทำงาน

1.2.6 เพื่อศึกษาการใช้งานไมโครคอนโทรลเลอร์ตระกูล AVR

1.3 ขอบเขตของปริญญานิพนธ์

1.3.1 ขอบเขตของอุปกรณ์ทางด้านฮาร์ดแวร์

1.3.1.1 ชุดกระจายเสียงแบบดิจิตอลทั้งภาครับและภาคส่งสามารถติดต่อหากันได้

1.3.1.2 ใช้เทคโนโลยีโมดูลอาร์เอฟ (RF Module) Zigbee 2.4 GHz ในการรับ – ส่งสัญญาณแบบ ไร้สายในย่านความถี่ 2.4 กิกะเฮิรตซ์ ระยะการใช้งานของโมดูลอาร์เอฟมีข้อจำกัดสองลักษณะคือการส่งสัญญาณวิทยุในที่โล่งและการส่งสัญญาณวิทยุภายในอาคารซึ่งความเร็วในการส่งข้อมูลสูงสุด 250 กิโลบิตต่อวินาทีซึ่งถ้าเป็นที่โล่งจะสามารถส่งได้ระยะประมาณ 0.5 – 1 กิโลเมตรส่วนภายในอาคารจะส่งได้ระยะประมาณ100 – 300 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.2 ขอบเขตทางด้านซอฟต์แวร์

1.3.2.1 เขียนโปรแกรมภาษาซีในไมโครคอนโทรลเลอร์เพื่อให้ควบคุมการทำงานของชุดรับสัญญาณและชุดส่งสัญญาณของโมดูลอาร์เอฟ

1.3.2.2 สามารถกำหนดช่องสัญญาณในการส่งและระบุรหัสของเครื่องรับได้

1.4 ประโยชน์ที่คาดว่าจะได้รับจากปริญญานิพนธ์

1.4.1 สามารถนำสิ่งประดิษฐ์นี้ไปใช้ในการรับ-ส่งการกระจายเสียงแบบดิจิตอลได้

1.4.2 สามารถทำให้การสื่อสารสะดวกยิ่งขึ้น

1.4.3 สามารถนำไปประยุกต์ใช้กับข่ายงานที่เกี่ยวข้องได้

1.5 แผนการดำเนินงาน

หลังจากที่ผู้จัดทำโครงการได้ทำการกำหนดจุดประสงค์ของโครงการและขอบเขตของโครงการเป็นที่เรียบร้อยแล้ว จึงได้วางแผนการดำเนินงานซึ่งสรุปเป็นตารางแผนภูมิการทำงานได้ดังนี้

ขั้นตอนการดำเนินงาน	ระยะเวลาการดำเนินงาน								
	2551							2552	
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
1. ศึกษาพื้นฐานการสื่อสารไร้สาย	↔								
2. ศึกษาข้อมูลเกี่ยวกับโมดูล Zigbee		↔							
3. ออกแบบวงจรควบคุม Zigbeeและทำการทดลองและบันทึกผลการทดลอง		↔	↔	↔					
4. ศึกษาข้อมูล IC Microcontroller ตระกูล AVR เบอร์ ATmega16 L				↔	↔	↔			
5. ศึกษาชุดคำสั่งต่างๆที่ใช้ในการเขียนโปรแกรม						↔	↔		
6. ออกแบบวงจร						↔	↔	↔	
7. ทำการทดลองและบันทึกผลการทดลอง							↔	↔	
8. รายงาน และ สรุปผลโครงการ								↔	↔

ตารางที่ 1.1 ตารางแผนการดำเนินงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีหรือหลักการ

การออกแบบและสร้างเครื่องรับส่งสัญญาณเสียงแบบดิจิตอลมีเอกสารและข้อมูลที่เกี่ยวข้องตามรายละเอียดดังต่อไปนี้

2.1 การสื่อสารไร้สาย

การสื่อสารไร้สายเป็นเทคโนโลยีที่ช่วยให้การติดต่อสื่อสารระหว่างอุปกรณ์สื่อสารหรือเครื่องคอมพิวเตอร์ 2 เครื่อง หรือกลุ่มของเครื่องคอมพิวเตอร์ที่สามารถสื่อสารกันได้รวมถึงการติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์เครือข่ายคอมพิวเตอร์ด้วยเช่นกัน โดยปราศจากการใช้สายสัญญาณในการเชื่อมต่อ แต่จะใช้คลื่นวิทยุเป็นช่องทางการสื่อสารแทนการรับ – ส่งข้อมูลระหว่างกันจะผ่านอากาศทำให้ไม่ต้องเดินสายสัญญาณ และติดตั้งใช้งานได้สะดวกขึ้น ระบบเครือข่ายไร้สายใช้แม่เหล็กไฟฟ้าผ่านอากาศเพื่อรับ – ส่งข้อมูลข่าวสารระหว่างเครื่องคอมพิวเตอร์ และระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์เครือข่าย โดยคลื่นแม่เหล็กไฟฟ้านี้อาจเป็นคลื่นวิทยุ (radio) หรืออินฟาเรด (Infrared) ก็ได้

ระบบเครือข่ายไร้สาย (Wireless LANs) เกิดขึ้นครั้งแรกในปี ค.ศ. 1971 บนเกาะฮาวาย โดย Project ของ นักศึกษาของมหาวิทยาลัยฮาวาย ที่ชื่อว่า “ALOHNET” ขณะนั้นลักษณะการส่งข้อมูลเป็นแบบ Bidirectional ส่งไป-กลับง่ายๆ ผ่านคลื่นวิทยุสื่อสารกันระหว่างคอมพิวเตอร์ 7 เครื่อง ซึ่งตั้งอยู่บนเกาะ 4 เกาะโดยรอบและมีศูนย์กลางการเชื่อมต่ออยู่ที่เกาะๆหนึ่ง ที่ชื่อว่า Oahu

เทคโนโลยีการเข้าถึงข้อมูลความเร็วสูงหรือที่เรียกกันโดยทั่วไปว่า การเข้าถึงบรอดแบนด์ (Broadband Access) มีได้หลายประเภท ทั้งบรอดแบนด์ผ่านสาย (สายทองแดง สายเคเบิล และสายใยแก้วนำแสง) และบรอดแบนด์ไร้สายผ่านทางคลื่นวิทยุ ซึ่งปัจจุบันสามารถส่งผ่านข้อมูลด้วยความเร็วสูงยิ่งขึ้น เนื่องจากมีรูปแบบของการมอดูเลชั่นที่ใช้คลื่นวิทยุได้คุ้มค่ามากขึ้น เทคโนโลยีบรอดแบนด์ในปัจจุบันหรือที่จะมีใช้งานในอนาคตอันใกล้ เทคโนโลยีการเข้าถึงข้อมูลเฉพาะในส่วนที่เป็นการเข้าถึงไร้สาย (Broadband Wireless Access: BWA) นั้น สามารถแยกกลุ่มออกได้ตามลักษณะของการเข้าถึงได้ดังต่อไปนี้

- Personal Area Network (PAN)
- Local Area Network (LAN)
- Metropolitan Area Network (MAN)
- Wide Area Network (WAN)

โดยที่มาตรฐาน IEEE 802.15.4 ซึ่งเป็นมาตรของ Zigbee อยู่ในกลุ่มของ Personal Area Network (PAN) ซึ่ง PAN คือ เครือข่ายที่ช่วยให้การติดต่อสื่อสารระหว่างมนุษย์กับสิ่งที่อยู่รอบตัวโดยทำการเชื่อมต่อระหว่างมนุษย์กับอุปกรณ์ไร้สายเข้าด้วยกันในมนุษย์ที่อยู่ไกลอาจเชื่อมต่อได้โดยการใช้เสียง อีกทั้งยังสามารถทำการติดต่อสื่อสารระหว่างมนุษย์กับภายนอกโลกได้ สำหรับเครือข่ายที่มีอุปกรณ์อยู่ใกล้ๆกัน PAN จะใช้ wireless ในการเชื่อมต่อ, ใช้สนามไฟฟ้าของร่างกายมนุษย์เป็นสื่อทางไฟฟ้า, ใช้สนามไฟฟ้า และอื่น โดยเมื่อ PAN ใช้ wireless ในการเชื่อมต่อ PAN จะอ้างอิงถึง wireless PAN (WPAN) WPAN ทำให้เกิด wireless ที่มีโครงสร้างแบบ “bubble” ขึ้นรอบๆตัวมนุษย์ ซึ่งอ้างอิงถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

personal operating space (POS) ความคิดพื้นฐานของระบบ WPAN คือ การรักษาการเชื่อมต่อระหว่างสอง WPAN ที่มีอุปกรณ์ครบถ้วนสมบูรณ์ โดยมีระยะห่างกันประมาณ 10 เมตร WPAN สามารถสร้างการเชื่อมต่อกันได้เอง อุปกรณ์ WPAN สามารถเชื่อมต่อกับอินเตอร์เน็ต หรือเชื่อมต่อกับเซ็นเซอร์ได้

WPAN ใช้ wireless ระยะทางสั้นๆ เหมือนกับ WLAN อย่างไรก็ตาม WPAN และ WLAN ต้องมีตำแหน่งที่ตั้งแน่นอน WPAN มีราคาต่ำ และใช้พลังงานน้อย โดยค่าใช้จ่ายจะสูงขึ้นเมื่อความเร็วในการสื่อสารและขนาดข้อมูลสูงขึ้น ในขณะที่ WLAN มีราคาสูง ใช้พลังงานมากกว่า และมีระยะทางในการสื่อสารมากกว่า

2.2 ความรู้ทั่วไปเกี่ยวกับเครือข่าย Zigbee

เทคโนโลยี Zigbee เป็นการสื่อสารที่ออกแบบขึ้นสำหรับการสื่อสารในเครือข่ายเซ็นเซอร์แบบไร้สาย (Wireless Sensor Network) โดยเริ่มจากการกำหนดมาตรฐานการรับ - ส่งข้อมูลแบบ IEEE 802.15.4 ที่เน้นการสื่อสารแบบประหยัดพลังงาน ความเร็วการรับส่งข้อมูลต่ำและมีราคาถูก การสื่อสารลักษณะนี้ได้ถูกนำมาใช้สำหรับการสื่อสารระหว่างเครื่องตรวจวัดหรือเซ็นเซอร์ที่ต้องการสื่อสารแบบไร้สายเพื่อลดความยุ่งยากซับซ้อนสำหรับการติดตั้ง เช่น บริเวณโรงงานหนึ่งๆ อาจจะต้องใช้จำนวนเซ็นเซอร์ปริมาณมากๆ และเครื่องรับส่งที่มีราคาถูกและประหยัดพลังงาน โดยการสื่อสารระยะใกล้แบบซิกบีแตกต่างจากการสื่อสารแบบบลูทูธ ดังนี้

- มีการเชื่อมต่ออย่างซับซ้อนเพื่อรองรับการเชื่อมต่อสำหรับเครือข่ายขนาดใหญ่
- การใช้งานแบบประหยัดพลังงานเพื่อการใช้งานได้ยาวนานจากพลังงานแบตเตอรี่
- การสื่อสารระยะใกล้ในระยะ 100-1500 เมตร
- เหมาะสำหรับการเฝ้าระวัง (Monitor) และการควบคุม (Control) ใช้งานอุตสาหกรรม งานสิ่งแวดล้อม งานก่อสร้าง และงานทางการแพทย์
- เน้นการสื่อสารข้อมูลที่ความเร็วประมาณ 125 – 250 กิโลบิตต่อวินาที (kbps)

มาตรฐาน IEEE 802.15.4 กำหนดขึ้นสำหรับการรับส่งข้อมูลเบื้องต้นในวงจรเครื่องรับส่งวิทยุ (Physical Layer) และการควบคุมการรับส่ง (Link Layer) ดังต่อไปนี้ การสื่อสารใช้คลื่นวิทยุความถี่ 2.4 กิกะเฮิรตซ์ (GHz) แบ่งออกเป็น 16 ช่องสัญญาณๆ ละ 5 เมกะเฮิรตซ์ (MHz) สำหรับความถี่ 900 เมกะเฮิรตซ์ (MHz) แบ่งออกเป็น 10 ช่องสัญญาณๆ ละ 2 MHz ใช้การผสมสัญญาณ (modulation) แบบ Offset Quadrature Phase Shift Keying (Offset-QPSK) และใช้การแก้ปัญหาสัญญาณรบกวนแบบ Direct Sequence Spread Spectrum (DSSS) ที่มีอัตราการสเปรดดิง (spreading) 2 ล้าน chip/sec ควบคุมการรับส่งข้อมูลโดยใช้โปรโตคอลแบบ Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) และเพื่อให้การสื่อสารเครือข่ายเซ็นเซอร์ไร้สายเป็นมาตรฐานเดียวกัน จึงกำหนดมาตรฐานเพิ่มสำหรับการเชื่อมต่อเป็นเครือข่าย (Network Layer) และการนำไปใช้งาน (Application Layer) ร่วมกับมาตรฐาน IEEE 802.15.4 เป็นมาตรฐานใหม่ที่กำหนดโดยองค์กร Zigbee Alliance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อเป็นโครงข่ายของเครือข่ายเซ็นเซอร์ไร้สาย กำหนดโดยมาตรฐาน Zigbee มีอยู่สองรูปแบบ ได้แก่ แบบดาว (Star) และแบบระดับเดียว (Peer-to-Peer) ซึ่งการเชื่อมต่อแบบ Star เหมือนการเชื่อมต่อแบบโครงข่ายจิ๋ว (Piconet) สำหรับเครือข่ายบลูทูธโดยการสื่อสารระหว่างอุปกรณ์เซ็นเซอร์ไร้สายสามารถทำได้โดยผ่านอุปกรณ์ Personal Area Network (PAN) Coordinator หรือ Gateway Node สำหรับการเชื่อมต่อแบบระดับเดียวเป็นการเชื่อมต่อเพื่อขยายโครงข่ายให้กว้างออกไปโดยในการเชื่อมต่อนั้นจะต้องมี PAN Coordinator ซึ่งเชื่อมต่ออยู่กับอุปกรณ์เซ็นเซอร์ที่มีความสามารถเต็ม (Full-Function Device) ในการหาเส้นทาง (Routing) และอุปกรณ์เซ็นเซอร์แบบที่มีความสามารถลดลง (Reduced-Function Device) จะเชื่อมต่อกับอุปกรณ์ที่มีความสามารถเต็มอีกทีหนึ่ง ซึ่งอุปกรณ์ที่มีความสามารถเต็มเทียบเท่าได้กับอุปกรณ์บลูทูธแบบ Master ที่มีหน้าที่เชื่อมต่อกับ Master ตัวอื่นๆ เพื่อให้เกิดเครือข่ายที่ใหญ่ขึ้น

ส่วนสำคัญสำหรับเครือข่ายเซ็นเซอร์ไร้สายคือกลไกการประหยัดพลังงาน ซึ่งตัวอุปกรณ์เซ็นเซอร์สามารถกำหนดระยะเวลาในการเข้าสู่การหลับหรือพักการทำงาน (Sleep Mode) โดยตัวอุปกรณ์จะทำหน้าที่ลดการใช้พลังงานให้เหลือน้อยที่สุดเพื่อเป็นการประหยัดพลังงาน การกำหนดการหลับนี้ทำได้โดยการร้องขอไปที่อุปกรณ์ Full Function Device เพื่อบอกระยะเวลาที่จะทำการหลับ และเมื่อมีการส่งข้อมูลจากอุปกรณ์ตัวอื่นมายังอุปกรณ์ที่หลับอยู่ อุปกรณ์ Full Function Device จะเก็บข้อมูลไว้ให้ชั่วคราวและถามหาอุปกรณ์ตัวนั้นเป็นระยะเมื่ออุปกรณ์ตัวนั้นตื่นหรือพร้อมทำงานต่อจะได้รับการถามหาอุปกรณ์ตัวนั้นจึงส่งการร้องขอข้อมูลที่ได้เก็บไว้ให้ และจึงทำการรับส่งข้อมูลจนได้รับข้อมูลครบสมบูรณ์ต่อไป

2.2.1 มาตรฐาน IEEE 802.15.4 และ layer

มาตรฐาน 802.15.4 ถ้ามีการใช้งานเกิดขึ้นจริงนั้นจะมีผลต่อการใช้ชีวิตประจำวันของเราอย่างมาก เช่น ในทางชีวะอนามัย นอกจากนี้ระบบ IEEE 802.15.4 จะช่วยเตือนภัยจากสิ่งแวดล้อม รวมถึงอุบัติภัยต่างๆ เช่น ไฟไหม้ น้ำท่วม แผ่นดินไหว เป็นต้น ทั้งนี้ระบบเตือนภัยในปัจจุบันไม่ได้เชื่อมต่อกันเป็นระบบเครือข่าย และตัวอุปกรณ์เองมีช่วงการใช้งานจากแบตเตอรี่สั้น นอกจากนี้ยังมีราคาแพงอีกด้วย แต่ในระบบ IEEE 802.15.4 สามารถใช้ได้กับอุปกรณ์พื้นฐานเช่น เซ็นเซอร์และ Actuators ที่มีราคาถูกทำให้สามารถติดตามเหตุการณ์ต่างๆ และอุปกรณ์จะทำงานอย่างอัตโนมัติตามที่เราต้องการ

สำหรับการประยุกต์ระบบ IEEE802.15.4 มาใช้ภายในบ้านจะสามารถทำให้อุปกรณ์ และเครื่องใช้ไฟฟ้าที่มาจากหลากหลายผู้ผลิตสื่อสารระหว่างกันเป็นระบบเครือข่ายได้ เช่นเมื่อมีสายโทรศัพท์เข้ามา โทรทัศน์จะลดเสียงลงอัตโนมัติทั้งนี้เราเองไม่ต้องตั้งค่าต่างๆให้กับอุปกรณ์ แต่อุปกรณ์จะเรียนรู้เองจากพฤติกรรมของเราภายในบ้านโดยแต่ละอุปกรณ์จะตรวจจับค่าต่างๆ เช่น ความเข้มแสงหลอดไฟ อุณหภูมิ เพลง ช่องโทรทัศน์ และ เว็บไซต์โดยแต่ละอุปกรณ์จะเรียนรู้รสนิยมของแต่ละคนและบันทึกไว้ นอกจากนี้เราสามารถนำระบบ 802.15.4 มาประยุกต์ใช้ในการโทรคมนาคม กับอุปกรณ์พื้นฐานต่างๆที่อยู่ตามท้องถนน ทางด่วน และที่อื่น ๆ ทั้งนี้อุปกรณ์ต่าง ๆจะสื่อสารกันเองเป็นระบบเครือข่าย ในระหว่างการเดินทางของรถบนท้องถนน อุปกรณ์ที่อยู่ข้างทางจะส่งข้อมูลที่จำเป็นในการเดินทางสำหรับถนนที่รถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิ่งอยู่ เช่น ความเร็วสูงสุดที่วิ่งได้ เส้นทางเป็นรถเดินทางเดียวหรือสองทาง สภาพการจราจร ข้อมูลอุบัติเหตุ เป็นต้น นอกจากนี้อุปกรณ์ต่างๆที่อยู่ข้างถนนจะมีการทำงานแบบอัตโนมัติด้วยเช่น ไฟส่องทางจะลดความเข้มลงเมื่อไม่มีรถวิ่งผ่านมา และระบบควบคุมการจราจร เป็นต้นประโยชน์อีกอย่างหนึ่งคือสามารถตรวจหาตำแหน่งของรถได้ ซึ่งจะคล้ายกับระบบ GPS แต่ระบบ GPS นี้ไม่สามารถตรวจหาตำแหน่งในบางสถานที่ได้ เช่นในอุโมงค์ ภายในอาคาร เป็นต้น และระบบ GPS ยังมีความคลาดเคลื่อนของตำแหน่งมากกว่า การใช้งานในระบบ 802.15.4 อีกด้วย

จากคุณสมบัติของมาตรฐาน IEEE 802.15.4 ในบางข้อทำให้เกิดการประยุกต์ใช้งานเพิ่มขึ้นโดยโครงงานชิ้นนี้ได้นำเอาโมดูล Zigbee ตามมาตรฐาน IEEE 802.15.4 มาใช้ในการสื่อสารข้อมูลที่เป็นสัญญาณเสียงแทนเพื่อที่จะนำไปใช้ประโยชน์ได้ในอนาคต เช่น การนำไปใช้เป็นหอกระจายเสียงตามหมู่บ้าน เป็นต้น

สำหรับมาตรฐาน IEEE 802.15.4 ถูกกำหนดไว้ในชั้นของ Physical Layer และ Media Access Control ที่มีการสื่อสารด้วยอัตราข้อมูลต่ำ ส่วน topology ที่รองรับนั้นจะมี 2 แบบคือ one-hop star เมื่อมีรัศมีการสื่อสารน้อยกว่า 10 เมตร และ multihop สำหรับ peer-to-peer topology ทั้งนี้แต่ละอุปกรณ์จะมีแอดเดรส ที่มีความยาว 64 หรือ 16 บิต (รองรับได้ 64,000 อุปกรณ์) สำหรับ ชั้น Physical Layer ของ มาตรฐาน IEEE 802.15.4 มี 3 แถบความถี่ คือ 2.4 GHz, 896 MHz และ 915 MHz

สำหรับแถบความถี่ 2.4 GHz มี 16 ช่องสัญญาณสื่อสาร ด้วยอัตราข้อมูล 250 Kbps สำหรับแถบความถี่ 896 MHz มี 10 ช่องสัญญาณสื่อสารด้วยอัตราข้อมูล 40 Kbps และสำหรับแถบความถี่ 915 MHz มี 1 ช่องสัญญาณสื่อสาร ด้วยอัตราข้อมูล 20 Kbps นอกจากนี้ในมาตรฐาน 802.15.4 มีระบบความปลอดภัยของข้อมูลที่ส่งผ่าน และสามารถตั้งค่าต่าง ๆด้วยตัวเองได้

Band (MHz)	Frequency Band	Bit Rate (kbps)	Symbol Rate (kbps)	DSS Spreading Parameter	
				Modulation Technique	Chip Rate
868	868-868.6 MHz	20	20	BPSK	300 kcps
915	902-928 MHz	40	40	BPSK	600 kcps
2400	2400-2483.5 MHz	250	62.5	O-QPSK	2 mcps

ตารางที่ 2.1 IEEE 802.15.4 frequency bands and data transfer rates

มาตรฐานของ Zigbee มีการแบ่งเป็น layer ซึ่ง layer เหล่านี้จะทำให้การใช้งานได้มีประสิทธิภาพมากขึ้นราคาถูก ติดตั้งง่าย การส่งข้อมูลที่น่าเชื่อถือ ใช้พลังงานน้อย แบ่งเป็น Layer ต่างๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.1 Network layer

Network layer ถูกออกแบบมาเพื่อทำให้การส่งข้อมูลใน network ใช้พลังงานไม่มาก สามารถจัดการกับ Network ที่มีจำนวนโนดมากๆโดยใช้ latency ทำหน้าที่ดังนี้

- Starting a network: สามารถสร้าง network ขึ้นใหม่ได้
- Joining and leaving a network: สามารถเข้าร่วมและออกจาก network ได้
- Configuring a new device: สามารถกำหนดค่าของ stack ได้
- Addressing ZigBee coordinator: กำหนด address ให้กับอุปกรณ์แต่ละตัวได้
- Synchronization within a network: สามารถติดต่อกับอุปกรณ์อื่นๆแบบ Synchronization ได้
- Security: ทำให้เฟรมรับส่งมีความปลอดภัย
- Routing: จัดหาเส้นทางของเฟรมปลายทาง

2.2.1.2 Application layer

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

- Discovery สามารถค้นหาและระบุได้ว่าอุปกรณ์ตัวใดติดต่ออยู่กับอุปกรณ์ตัวใดอยู่
- Binding สามารถจับคู่อุปกรณ์ไว้ด้วยกันได้โดยใช้ table of binding และ forward messages ระหว่างอุปกรณ์

ZigBee Device Object (ZDO) ทำหน้าที่ดังนี้

- กำหนดหน้าที่ของอุปกรณ์แต่ละตัวภายใน network เช่น ZigBee coordinator or end device
- สร้างหรือตอบสนอง binding requests
- สร้างระบบรักษาความปลอดภัยระหว่าง network device

2.2.1.3 Media access control (MAC) layer

ชั้น MAC Layer มาตรฐาน IEEE 802.15.4 จัดเตรียมการเพื่อ support ระหว่างสอง wireless network topology เช่น การเชื่อมต่อแบบ star และ peer-to-peer โดยสำหรับการใช้งานในบ้านนั้นจะเป็นแบบ star และในทางอุตสาหกรรมและธุรกิจจะใช้การเชื่อมต่อแบบ peer-to-peer สำหรับโครงสร้างแบบ star การเชื่อมต่อภายในเครือข่าย (network) จะถูกควบคุมด้วย PAN coordinator เพียงอันเดียว โดย PAN coordinator นี้จะทำหน้าที่เป็นเหมือนผู้ดูแลเครือข่าย (network master) ทำหน้าที่ initialize เครือข่าย จัดการ node ในเครือข่ายและเก็บข่าวสารของ node ในเครือข่าย เพียงอุปกรณ์แบบ FFD (full-function device) สามารถส่ง beacon frame ก็อาจกลายเป็น PAN coordinator ได้ อย่างไรก็ตามอุปกรณ์แบบ RFDs (Reduced-function device) อาจมีส่วนร่วมในเครือข่ายแบบ star ได้ ยิ่งกว่านั้น เครือข่ายแบบ star จะทำงาน independent กับเครือข่ายอื่นๆใน IEEE 802.15.4 FFD อาจจะสร้าง เครือข่าย star หลังจากทำ scan channel ถ้า FFD ไม่สามารถตรวจพบการส่ง beacon frame และเริ่มต้นทำหน้าที่ PAN coordinator โดยการส่ง beacon frame ที่มี unique network identifier หรือ ID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทุกอุปกรณ์ที่มีส่วนร่วมอยู่ใน LR-WPANs ต่างใช้ IEEE 64-bit Address ที่ unique เมื่อมีอุปกรณ์ตัวหนึ่งเริ่มทำการส่ง beacon frame อุปกรณ์อื่นก็อาจจะมีการติดต่อเข้าเพื่อขอมีส่วนร่วมโดยส่ง association message เข้ามาที่ ad hoc network

เครือข่ายแบบ Peer-to-peer ซึ่งเป็นการส่งผ่านข้อมูลในเครือข่ายอุปกรณ์ จะอนุญาตให้อุปกรณ์ FFD ติดต่อกับ FFD อื่นที่อยู่ภายในอาณาเขตของตัวมันเองและส่ง message ไปยัง FFD ที่อยู่นอกเขตอีกด้วย ซึ่งเครือข่ายแบบ peer-to-peer นี้สามารถทำให้ ad hoc wireless network ซับซ้อนขึ้นด้วยการเพิ่มการครอบคลุมพื้นที่เนื่องจาก multihop และ mesh network จะกำหนดคุณสมบัติที่อนุญาตในการส่ง message ส่วน RFDs อาจจะสามารถเข้ามาร่วมในเครือข่าย peer-to-peer แต่จะไม่สามารถทำ relay ได้

อีกหนึ่งชนิดของเครือข่ายแบบ peer-to-peer เครือข่าย cluster-tree ซึ่งจะเป็นการรวมกันของแบบ Star และ Mesh เพื่อที่จะได้ประโยชน์จากทั้งสองแบบสามารถรองรับจำนวนของ node และสามารถใช้งานจากแบตเตอรี่ได้นาน โดยเครือข่ายชนิดนี้จำนวนของอุปกรณ์เครือข่ายจะอยู่บนหลักการ “cluster heads” โครงสร้างนี้จะจัดหา path ให้กับอุปกรณ์เครือข่ายชนิดต่างๆและสามารถมีการขยายเครือข่ายออกไปให้กว้างขึ้น ในหลายๆกรณีเครือข่ายแบบ peer-to-peer จะเลือกอุปกรณ์ตัวหนึ่งให้ทำหน้าที่ PAN coordinator ระหว่างการสร้าง เครือข่ายแบบ peer-to-peer จะต้องมีตรวจสอบแล้วว่าการทำงานต่างๆนั้นรองรับการทำงานและ service ของอุปกรณ์เครือข่ายทำให้สามารถทำงานได้ ในการเข้าถึง channel ของโมดูล Zigbee นั้นใช้หลักการของ CSMA/CA ดังนั้นเราจึงต้องศึกษาการทำงานต่อไปนี้

เทคนิค CSMA/CA with Acknowledgement สำหรับการจัดสรรการเข้าใช้ช่องสัญญาณของแต่ละสถานีเพื่อแก้ไขปัญหาเหล่านี้ ซึ่งการทำงานของกลไก CSMA/CA โดยหลักแล้วเป็นเช่นเดียวกับในส่วนของ CSMA with Random Back-Off แต่จะมีรายละเอียดเพิ่มเติมเกี่ยวกับการหลีกเลี่ยงไม่ให้เกิดการชนกันของสัญญาณและเทคนิคสำหรับการตรวจสอบว่าเกิดการชนของสัญญาณหรือไม่แบบเป็นนัย โดยสถานีผู้ส่งสัญญาณข้อมูลจะต้องรอรับ Acknowledgement จากสถานีที่ส่งข้อมูลไปให้ หากไม่ได้รับ Acknowledgement กลับมาภายในเวลาที่กำหนดจะถือว่าเกิดการชนของสัญญาณขึ้นและต้องทำการส่งข้อมูลเดิมซ้ำอีกต่อไป ซึ่งกลไกของCSMA with Random Back-Off มีกลไกการทำงานดังนี้

กลไก CSMA (Carrier Sense Multiple Access) with Random Back-Off เป็นเทคนิคอย่างง่ายสำหรับจัดสรรการเข้าใช้ช่องสัญญาณของผู้ใช้แต่ละคน (ซึ่งต้องแบ่งกันใช้ช่องสัญญาณร่วมนี้) อย่างยุติธรรม กลไกนี้เป็นที่ยอมรับและนิยมใช้กันอย่างแพร่หลาย เช่น ในมาตรฐาน IEEE 802.3 Ethernet LAN หลักการทำงานของกลไก CSMA คือ เมื่อสถานีหนึ่งต้องการเข้าใช้ช่องสัญญาณ สถานีดังกล่าวจะต้องตรวจสอบช่องสัญญาณก่อนว่ามีสถานีอื่นทำการรับส่งสัญญาณข้อมูลอยู่หรือไม่และรอจนกว่าช่องสัญญาณจะว่าง เมื่อช่องสัญญาณว่างแล้วสถานีที่ต้องการเข้าใช้ช่องสัญญาณจะต้องรอต่อไปอีกระยะหนึ่ง (Random Back-Off) ซึ่งแต่ละสถานีได้กำหนดระยะเวลาในการรอดังกล่าวไว้แล้วด้วยการสุ่มค่าหลังจากเสร็จการใช้ช่องสัญญาณครั้งก่อน สถานีที่สุ่มได้ค่าระยะเวลาในการรอน้อยกว่าก็จะมีสิทธิ์ ในการเข้าใช้ช่องสัญญาณก่อน แต่อย่างไรก็ตามในบางกรณีกลไกดังกล่าวอาจจะกำหนดให้สถานีมากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนึ่งสถานีส่งข้อมูลในเวลาพร้อมๆ กันซึ่งจะทำให้เกิดการชนกันของสัญญาณได้ ซึ่งหากเกิดการชนกันของสัญญาณขึ้นจะต้องมีการส่งสัญญาณข้อมูลเดิมซ้ำอีกครั้งด้วยกลไกที่กล่าวมาแล้วข้างต้น

2.2.1.4 Physical Layer

ถูกออกแบบมาเพื่อลดต้นทุนของความต้องการโดยการใช้วิธี Direct sequence ซึ่งทำให้วงจรไฟฟ้ามีความง่ายมากขึ้น ทำให้ราคาของการติดตั้งน้อยลง Physical type device ที่จะช่วยลดต้นทุนของระบบมี 2 อย่างคือ full function devices (FFD) และ reduced function devices (RFD)

Full function device (FFD)

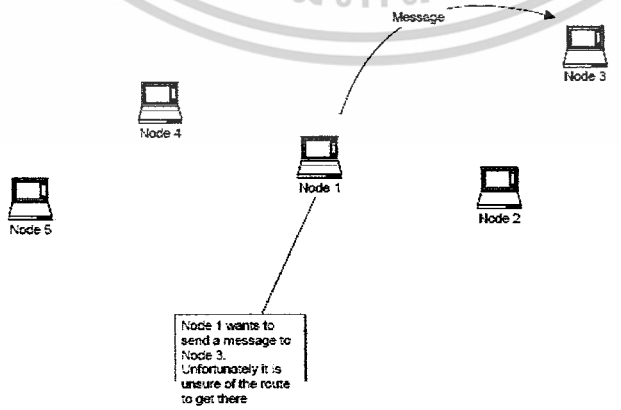
- สามารถฟังก์ชันได้ในทุกๆ topology
- สามารถทำเป็น network coordinator ได้
- สามารถทำเป็น coordinator ได้
- สามารถติดต่อได้กับทุกๆ device

Reduced function device (RFD)

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

2.2.2 ความรู้ทั่วไปเกี่ยวกับ AODV Protocol

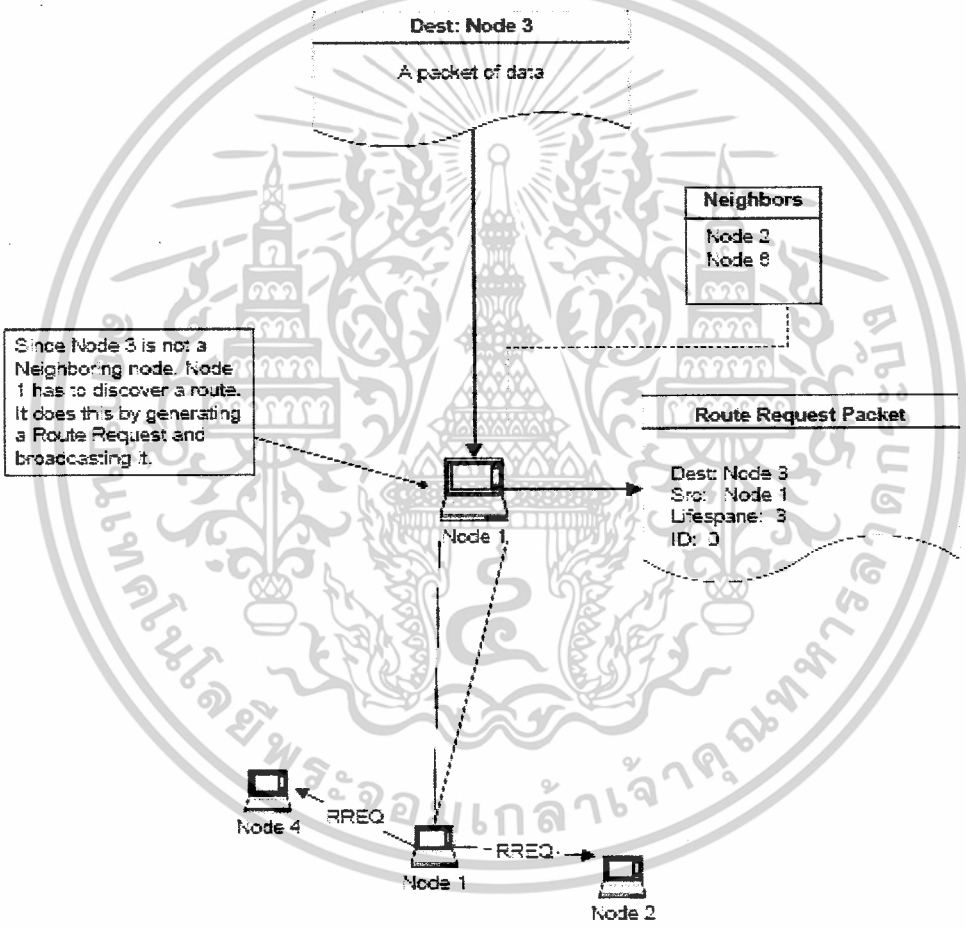
AODV เป็นโปรโตคอลที่ใช้อยู่ในตัวของโมดูล Zigbee โดยที่เป็นแบบแผนของสื่อสารระหว่างเครื่องคอมพิวเตอร์ที่เคลื่อนย้ายได้ (Mobile computer) หรือ node ในการส่ง message ผ่านไปยัง node ข้างเคียง (neighbor) เพื่อไปยัง node ที่ต้นทางไม่สามารถติดต่อได้โดยตรงในระหว่างทางที่ message ถูกส่งผ่านไป AODV ก็จะทำการค้นหาเส้นทางไป โดยจะมั่นใจได้ว่าจะไม่เกิดการวนลูป (loop) และพยายามหาเส้นทางที่สั้นที่สุดที่จะเป็นไปได้อีก ทั้ง AODV ยังสามารถที่จะควบคุมการเปลี่ยนแปลงของเส้นทาง (route) และสามารถสร้างใหม่หากเกิดข้อผิดพลาดได้อีกด้วย



รูป 2.1 การตั้งขอบเขตให้กับnodeต่างๆใน wireless network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

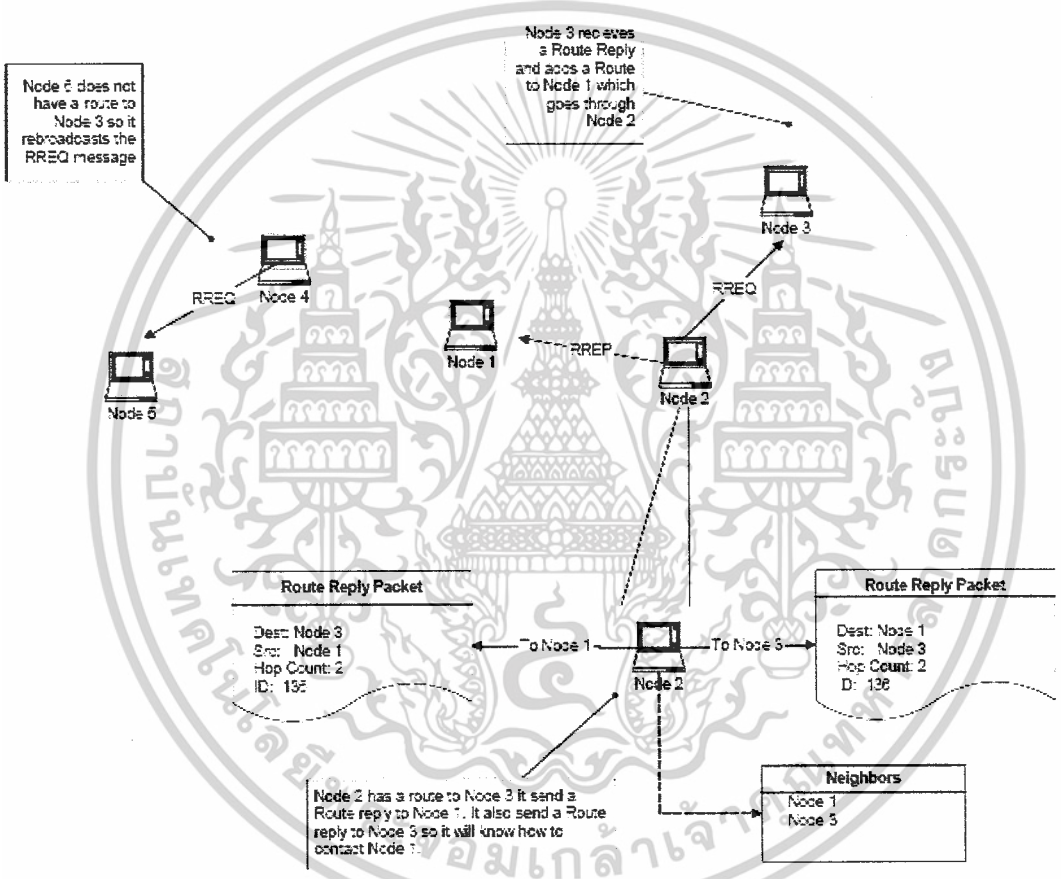
จากรูปเป็นการแสดงการตั้งค่าให้กับ Node ใน wireless network วงกลมที่เราเห็นอยู่นั้นจะแสดงขอบเขตการติดต่อสื่อสารของแต่ละ node และเพราะว่ามีการกำหนดขอบเขตการเชื่อมต่อทำให้แต่ละ nodeสามารถติดต่อกับ node ที่อยู่ถัดจากตัวเองเท่านั้น สำหรับ node ที่เราสามารถติดต่อได้โดยตรงนั้นเราจะเรียกว่า neighbor โดย node จะเก็บข้อมูลของ neighbor เมื่อได้รับ HELLO message ที่แต่ละ node จะทำการ broadcast ออกมาตามช่วงเวลาที่ได้ตั้งไว้ เมื่อมี node ใดๆต้องการส่ง message ไปยัง node อื่นที่ๆไม่ใช่ neighbor มันจะทำการ broadcast a Route Request message (RREQ) ซึ่งใน RREQ นี้จะประกอบไปด้วยคีย์บิตของข้อมูลหลายตัว เช่น ต้นทาง (source) ปลายทาง (destination) อายุ (lifespan) ของ message และหมายเลขลำดับ (sequence number) ที่เป็น unique ID



รูป 2.2 การทำงานของ AODV Protocol

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในตัวอย่าง node1 ต้องการจะส่ง message ไปยัง node3 โดยที่ node1 มี neighbors คือ node2 และ 4 ดังนั้น node1 จะไม่สามารถติดต่อกับ node3 ได้ตรงๆจึงต้องส่ง RREQ ไปที่ node2 และ nod4 เมื่อ neighbor ของ node1 ได้รับ RREQ message จะมีสองเงื่อนไขคือถ้า node เหล่านั้นรู้จักเส้นทางที่จะไปหาปลายทางหรือเป็นปลายทางเอง มันก็จะสามารถส่ง Route Reply message กลับมาที่ node1 แต่ถ้าไม่ใช่กรณีดังกล่าวก็จะต้อง rebroadcast RREQ ไปยัง neighbor ของตน และถ้ามีการ เวลา lifespan หมดไปแต่ node1 ยังไม่ได้รับ reply ก็จะมีการส่ง RREQ ออกไปใหม่โดยจะเพิ่มเวลา lifespan และสร้าง ID ใหม่ขึ้นมาแนบออกไปด้วย โดย nodeทั้งหมดจะใช้ Sequence number ใน RREQ เพื่อตรวจสอบว่าเป็นการ rebroadcast หรือไม่



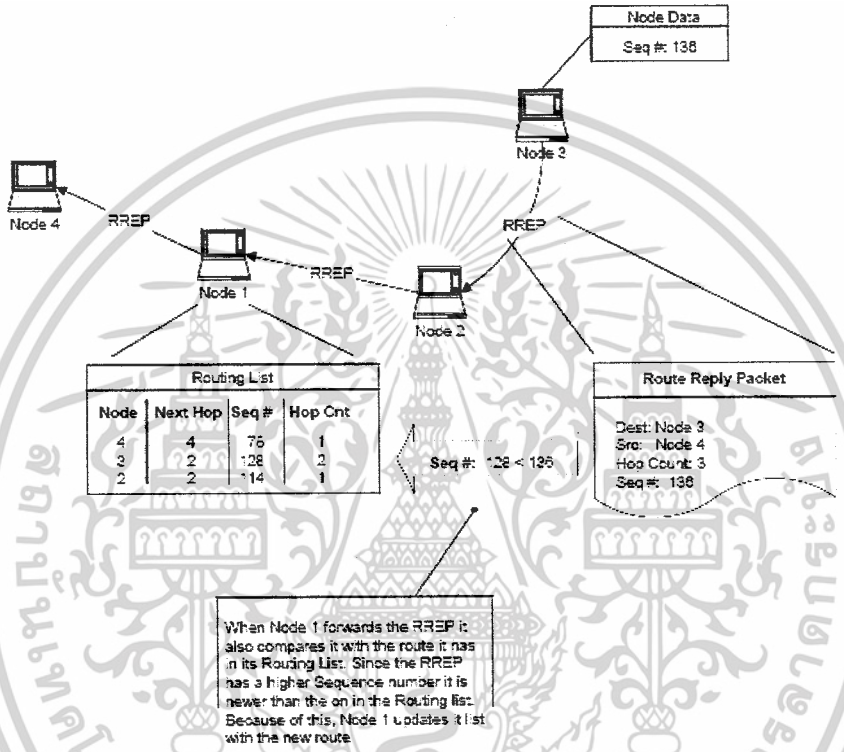
รูป 2.3 การทำงานของ AODV Protocol (ต่อ)

ในตัวอย่าง node2 มีเส้นทางที่จะไปที่ node3 มันจะทำการ reply RREQ โดยการส่ง RREP ส่วน node4 ที่ไม่มีข้อมูลเส้นทางไปยัง node3 ก็จะทำการ rebroadcast RREQ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.1 Sequence Numbers

Sequence Numbers จะให้บริการคล้ายการระบุเวลา (time stamp) โดยยอมให้ node นำไปเปรียบเทียบว่าข้อมูลอันไหนของมันใหม่ (fresh) ที่สุด ซึ่งโดยทุกครั้งที่ node มีการส่ง message ชนิดต่างๆออกไป มันก็จะทำการเพิ่มค่า Sequence Numbers ของตัวเองแต่ละ node จะทำการบันทึก Sequence Numbers ของ node อื่นที่มันติดต่อด้วย Sequence Numbers ที่สูงกว่าจะแสดงเส้นทางที่ใหม่กว่า



รูป 2.4 วิธีการใช้ Sequence Number

ในตัวอย่าง node1 ส่ง RREQ ไปยัง node4 สังเกตจาก route ใน RREP จะมี Sequence Numbers มากกว่าใน Routing list ดังนั้น node1 ก็จะทำการแทนที่ด้วย route ใหม่ที่อยู่ใน RREP

2.2.2.2 Error Messages

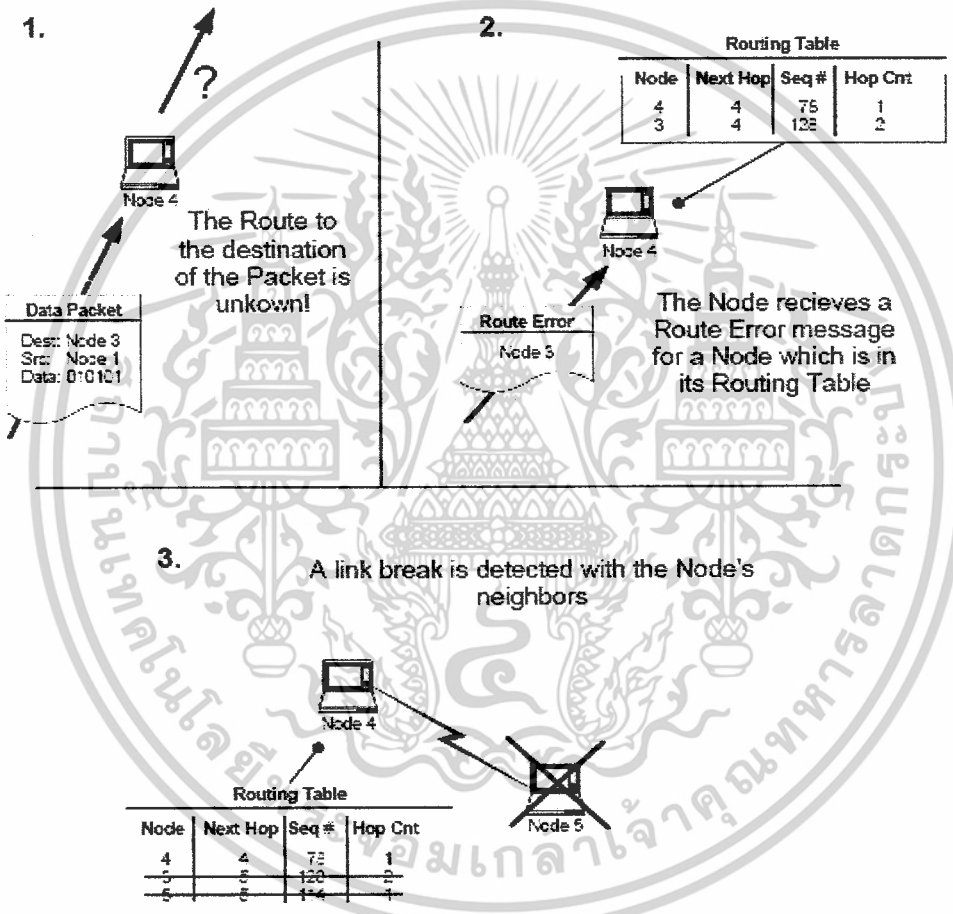
Route Error Message (RERR) ยอมให้ AODV ปรับปรุงเส้นทางเมื่อnodeมีการเคลื่อนย้ายเมื่อ node ได้รับ RERR ก็จะไปพิจารณาที่ Routing Table แล้วก็จะทำการ Remove node ที่มีปัญหา (bad Nodes) ออกไปจากไดอะแกรมที่แสดงเหตุการณ์ที่ node จะส่ง RERR ให้กับ neighbors

- รูป 1 node ได้รับ Data packet ที่เชื่อว่าสามารถส่งให้ปลายทางได้เลยแต่ว่าที่จริงแล้วไม่สามารถทำได้ เพราะว่าไม่รู้เส้นทางที่จะไปยังปลายทาง ปัญหาที่แท้จริงในข้อนี้ไม่ใช่ไม่รู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้นทางที่จะส่งไปแต่ปัญหาที่แท้จริงคือ node อื่นเข้าใจว่า node นี้เป็นเส้นทางที่ถูกต้องในการส่งให้ปลายทางต้องส่งผ่าน node นี้

- รูป 2 node ได้รับ RERR Message เนื่องจาก node อย่างน้อยหนึ่งตัวในเส้นทางใช้ไม่ได้ โดย node ที่ส่งมานั้นจะเป็น node ที่มีอยู่ใน Routing table ของตัวมันเอง
- รูป 3 node ตรวจพบว่า neighbor ไม่สามารถติดต่อได้ เมื่อเกิดเหตุการณ์นี้ขึ้น node จะทำการตรวจเช็คที่ Routing table ว่ามีการใช้ Neighbor นั้นๆเป็น next hop หรือไม่ถ้ามีก็จะทำการ mask ไว้ว่าเป็น invalid จากนั้นจะส่ง RERR ที่มีทั้ง Neighbor และ invalid routes



รูป 2.5 สาเหตุของการเกิด ERROR Message

2.2.2.3 สรุปคุณสมบัติของ AODV

- จะหาเฉพาะเส้นทางที่ต้องการเท่านั้น
- ใช้ Sequence Number ในการเก็บข้อมูลที่ถูกต้องที่สุด
- จะเก็บข้อมูลเฉพาะ Next hop ของ route แทนการเก็บข้อมูลทั้งหมด
- ใช้ Periodic HELLO Message ในการ track Neighbor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 Received Signal Strength Indication (RSSI)

Received Signal Strength Indication หรือ RSSI เป็นวิธีการวัดความแรงของสัญญาณในการส่งสัญญาณวิทยุ ตามความเป็นจริงของสัญญาณโดยทั่วไป เมื่อระยะทางไกลออกไปสัญญาณก็จะเบาลง ดังนั้นค่า RSSI นั้นก็จำเป็นต่อการส่งสัญญาณเช่นกัน เพราะจะได้รู้ว่า เครื่องรับสัญญาณที่ใช้งานอยู่นั้นจะมีประสิทธิภาพที่ระยะทางเท่าใด หรือกำลังในการส่งเท่าใดจึงจะเหมาะสม

สำหรับการวัดความแรงของสัญญาณ ที่แปลงค่าความแรงมาเป็นแรงดันไฟฟ้านั้นและมักไม่ใช้ในงานที่ต้องการความแม่นยำสูง เนื่องจากมีความแปรปรวนของสัญญาณตามสภาพแวดล้อมสูงเพราะแต่ละสภาพแวดล้อมมีความแตกต่างกัน การจะทำให้ RSSI มีความน่าเชื่อถือคือจะต้องเก็บข้อมูลไว้หลายๆ ครั้ง เพื่อนำมาวิเคราะห์ภายหลัง จะทำให้เกิดความผิดพลาดน้อยลง ความผิดพลาดที่เกิดขึ้นนั้นอาจจะมีสาเหตุมาจากต่างๆ ดังนี้

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

2.3 ความรู้ทั่วไปเกี่ยวกับโมดูล Xbee Pro

รูปที่ 2.6 ตัวโมดูล Zigbee

เป็นโมดูล อุปกรณ์เครือข่ายแบบไร้สายระยะใกล้ความเร็วต่ำโดยภายใต้มาตรฐาน IEEE 802.15.4 มีอัตราการรับส่งข้อมูลต่ำกว่าหรือเท่ากับ 250 Kbps ใช้กำลังไฟน้อย ซึ่งเทคโนโลยีที่ใช้มาตรฐานนี้ในปัจจุบันคือ Zigbee และ เป็นเทคโนโลยีการสื่อสารแบบไร้สายที่ถูกพัฒนาให้มีจุดเด่นกว่าเทคโนโลยีไร้สายแบบอื่นๆเหมาะสำหรับการใช้งานด้านเซ็นเซอร์ไร้สาย สำหรับตรวจสอบตำแหน่งของวัตถุ ตรวจสอบสภาพแวดล้อม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 คุณสมบัติโดยทั่วไปของ Zigbee

- อัตราการส่งข้อมูล 250 Kbps (2.4GHz), 40 kbps (915 MHz), 20 kbps (80 MHz)
- High throughput และ low latency Duty Cycle ต่ำ (< 0.1%)
- มีการเข้าถึง Channel แบบ Channel access using Carrier Sense Multiple Access with Collision Avoidance (CSMA – CA)
- สามารถรองรับ Address ได้ถึง 64 bit IEEE address (65535 networks)
- รับประกันการส่งแบบ Full hand shake protocol
- เชื่อมต่อ Topology ได้หลายแบบเช่น Star, Peer-to-Peer, Mesh
- ใช้พลังงานต่ำ
- ระยะทางการส่งพื้นฐาน 5 - 500 เมตร

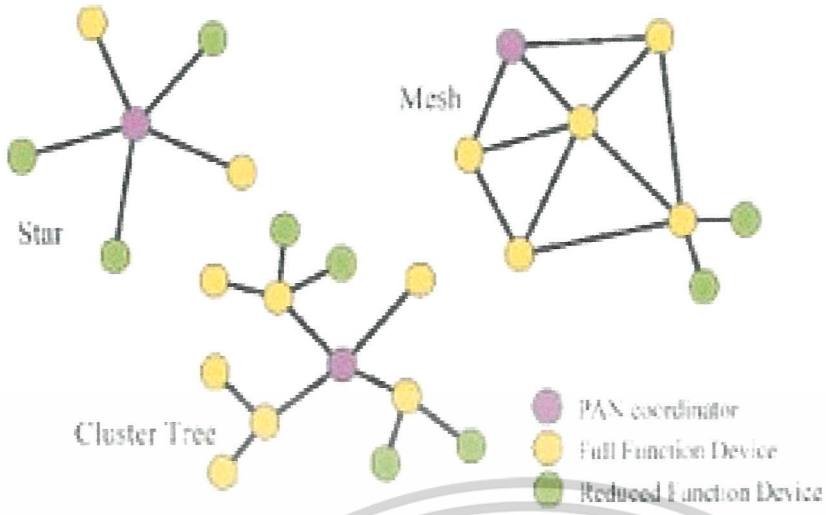
โปรโตคอล ZigBee สามารถสร้างระบบเครือข่าย ได้สามรูปแบบคือ Star, Peer-to-Peer และ Cluster Tree

- Star Topology เมื่อ FFD ทำงานเป็นครั้งแรก มันจะทำการสร้าง Network ของตนเอง และจะกลายเป็น PAN coordinator หลังจากนั้นจะทำการเลือก PAN Identifier ซึ่งจะต้องไม่ถูกใช้ในระบบเน็ตเวิร์คอื่น Application : Home Automation , Personal Computer Peripheral , Toys and Games

- Peer-to-Peer Topology จะมีอุปกรณ์ตัวหนึ่งที่ทำหน้าที่เป็น PAN Coordinator ซึ่งอุปกรณ์ทุกตัวสามารถสื่อสารกับอุปกรณ์อื่นๆแบบ Peer-to-Peer ได้ และ Peer-to-Peer Network สามารถที่จะเป็นได้ทั้ง ad hoc , Self-Organizing และ Self-healing Application : industrial control and monitoring , wireless sensor networks asset and inventory tracking

- Cluster tree Topology ในการสร้างระบบ Cluster tree PAN Coordinator จะใช้ตัวเองเป็น Cluster แรกเรียกว่า Cluster Head (CLH) และ Cluster Identifier (CID) เป็นศูนย์ หลังจากนั้นทำการเลือก PAN Identifier และทำการ Broadcasting Beacon Frame ไปยังอุปกรณ์ข้างเคียง และเมื่ออุปกรณ์ตัวอื่นได้รับก็จะส่ง Request ไปที่ CLH จากนั้น CLH ทำการยอมรับและเพิ่มชื่ออุปกรณ์ใน Neighbor List แบบเป็นลูกข่าย (child) ในขณะที่อุปกรณ์ที่เข้ามาใหม่จะเพิ่มชื่อ CLH ตัวนั้นใน Neighbor List ว่าเป็นแม่ข่าย (parent) หลังจากนั้นก็ทำการส่ง Beacon ไปยังอุปกรณ์ตัวอื่นๆ ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 โครงข่ายของเครือข่าย Zigbee

อุปกรณ์ Zigbee แบ่งออกเป็น 3 ส่วน คือ

- Zigbee Coordinator ทำหน้าที่สร้างโครงข่าย จัดการโนดในโครงข่ายและเก็บข่าวสารของโนดในโครงข่าย
- Zigbee Router ทำหน้าที่จัดการเส้นทาง ของ ข้อมูลที่ส่งผ่านภายในโครงข่ายระหว่างโนด
- Zigbee end Device เป็นจุดปลายของโครงสร้างเครือข่าย อยู่ในส่วนของผู้ใช้งาน โดยสามารถเป็นได้ทั้งแบบ RFD และ FFD ซึ่งแสดงไว้ดังรูปที่

รูปที่ 2.8 การเชื่อมต่อแบบต่างๆของ Zigbee

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.3.2 การส่งข้อมูลในรูปแบบของ API

การส่งข้อมูลของโมดูลจะใช้การส่งข้อมูลในรูปแบบ API (Application Programming Interface) คือ การจัดเก็บข้อมูลที่ต้องการจะติดต่อสื่อสารกับโมดูลอื่นๆในรูปแบบโครงสร้างที่แน่นอน โดยจัดเก็บเป็น Frame packet ซึ่งสามารถบอกได้ว่า packet ที่ส่งไปนั้นมีคำสั่งอะไร การส่งข้อมูลถูกต้องหรือไม่ และมีสถานะของการส่งและการรับของpacket นั้นๆด้วย ซึ่งข้อมูลทีส่งไปจะอยู่ในรูปแบบของเลขฐานสิบหก

2.3.2.1 รูปแบบของ API Frame

เราสามารถส่งข้อมูลในรูปแบบของ API Mode ได้โดยการตั้งค่าพารามิเตอร์ของคำสั่ง AP (API Enable) ดังนี้

AP = 0 เป็นการตั้งค่าการส่งข้อมูลให้อยู่ในรูปแบบของ UART แบบปกติ API Mode จะถูกยกเลิก

AP = 1 เป็นการตั้งค่าการส่งข้อมูลให้อยู่ในรูปแบบของ API Mode

AP = 2 เป็นการตั้งค่าการส่งข้อมูลให้อยู่ในรูปแบบของ API Mode (Escaped Character)

รูปที่ 2.9 รูปแบบของ API Packet frame Mode 1

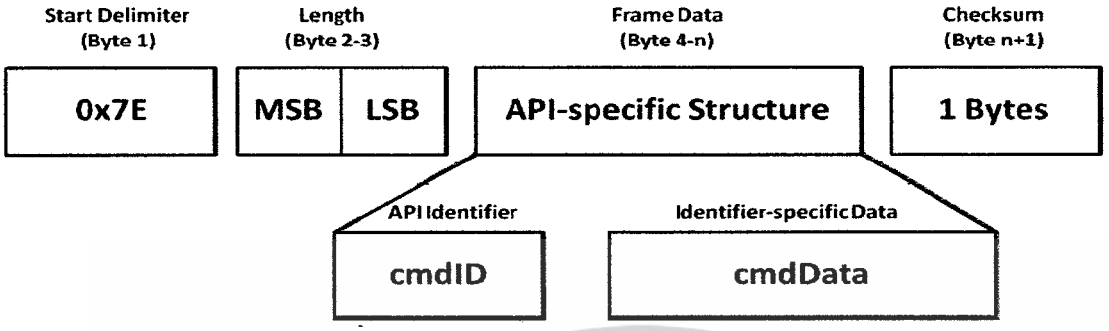
MSB = Most Significant Byte
LSB = Last Significant Byte

Checksum คือ ไบต์ที่ตรวจสอบความถูกต้องของข้อมูลที่ได้รับหรือส่ง สามารถคำนวณหาค่า checksum ได้โดยบวก Frame Data ทุกไบต์ในลักษณะของเลขฐานสิบหก หลังจากนั้นนำไบต์สุดท้ายไปลบออกจาก 0xFF ก็จะได้ค่า checksum ที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

104325

2.3.2.2 ประเภทของ API



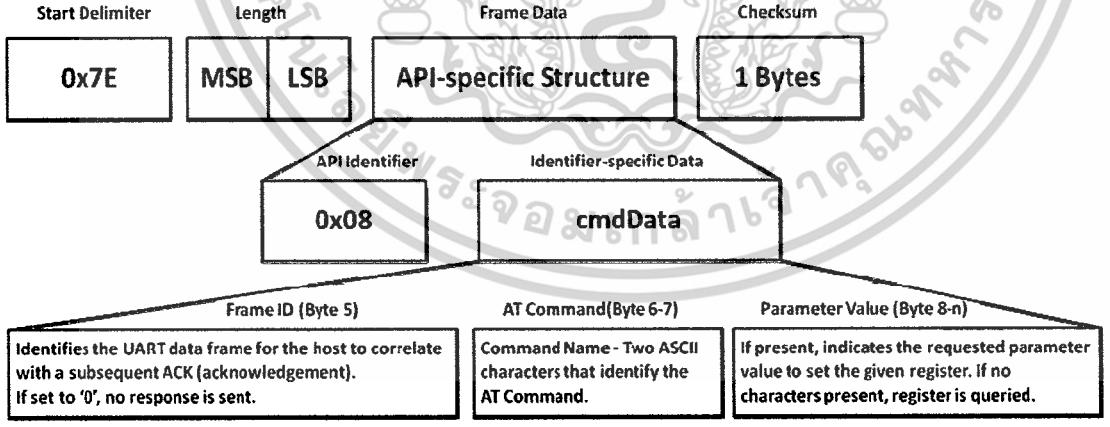
รูปที่ 2.10 รูปแบบของ API Packet frame

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

cmdData (Identifier-specific Data) คือ รูปแบบของข้อมูลที่ต่างๆ ที่ถูกกำหนดโดย cmdID

2.3.2.3 AT Command

API frame ประเภท AT command เป็นแพคเก็ตที่สามารถใช้เขียนหรืออ่านค่าพารามิเตอร์ของโมดูล ZigBee ได้ ซึ่งใช้คำสั่ง AT Command ในรูปแบบนี้ไม่จำเป็นต้องอยู่ในโหมด AT command ค่าต่างๆสามารถอ่านหรือแก้ไขได้ทันที โดยจะมีค่า API Identifier เป็น 0x08 โครงสร้างของ AT Command frame ดังรูป



รูปที่ 2.11 รูปแบบของ API Packet frame ประเภท AT Command

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

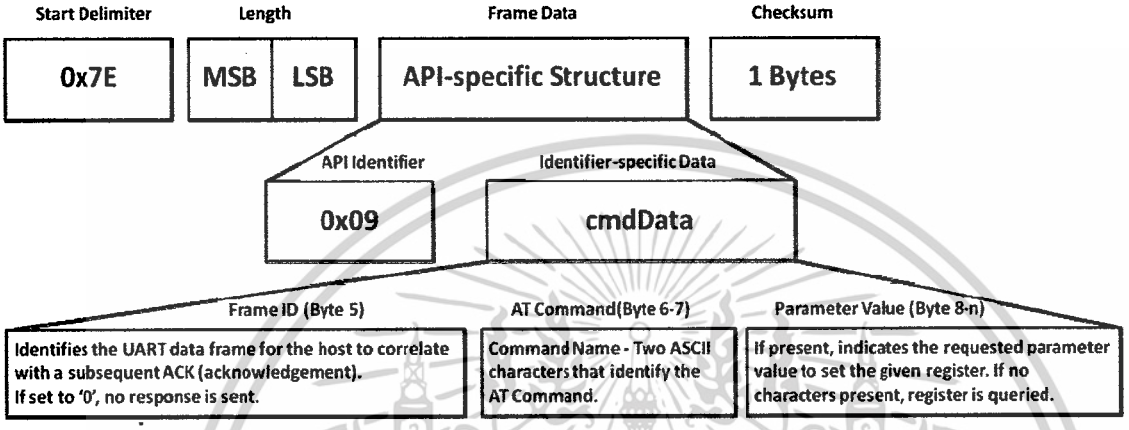
AT Command คือ ชื่อของ AT Command ประกอบด้วยตัวอักษร 2 ตัว

Parameter Value คือ ค่าพารามิเตอร์ของ AT Command นั้นๆที่ต้องการตั้งค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.4 AT Command Queue Parameter Value

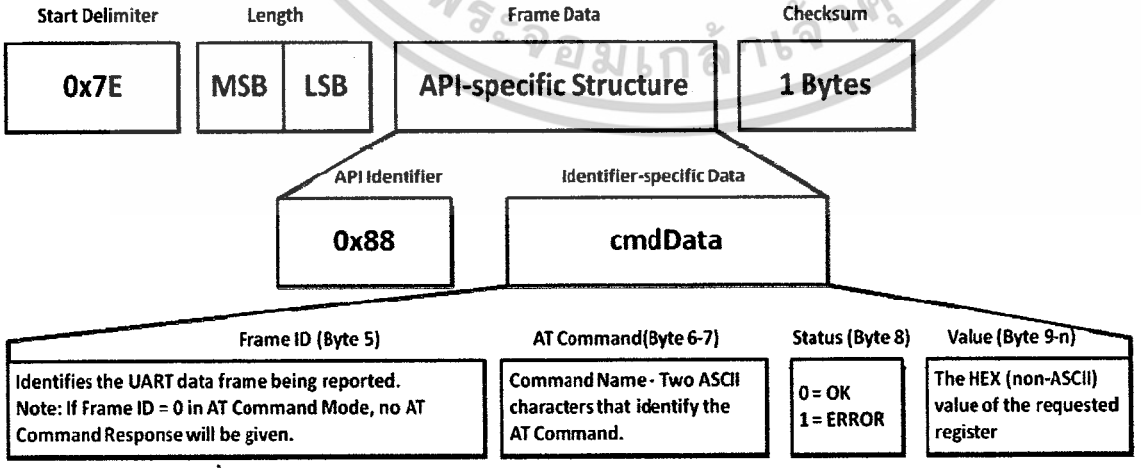
API frame ประเภทนี้จะเหมือนกับ API frame AT Command ซึ่งสามารถใช้เขียนหรืออ่านค่าพารามิเตอร์ของโมดูล ZigBee ได้โดยใช้ AT Command แต่ API frame ประเภทนี้จะทำงานก็ต่อเมื่อ AT Command (API Identifier = 0x08) ทำงานเสร็จ ซึ่ง AT Command Queue Parameter Value มีค่า API Identifier เป็น 0x09 โครงสร้างของ API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



รูปที่ 2.12 รูปแบบของ API Packet frame ประเภท AT Command Queue Parameter Value

2.3.2.5 AT Command Response

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

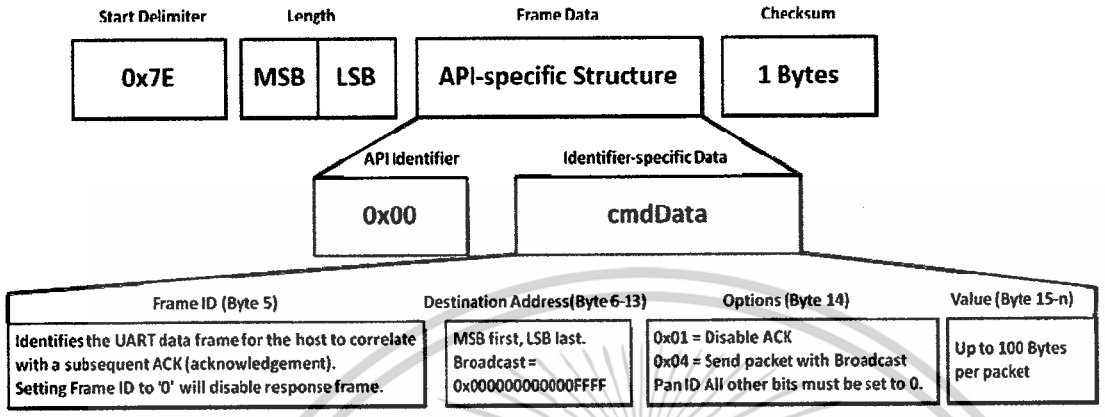


รูปที่ 2.13 รูปแบบของ API Packet frame ประเภท AT Command Response

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.6 Tx (Transmit) Request: 64-bit address

API frame ประเภทนี้จะส่งข้อมูลในรูปแบบของ packet ไปยังโมดูลปลายทาง โดยที่ใช้อยู่แบบ 64bit และมี API Identifier เป็น 0x00 โครงสร้างของ API frame ประเภทนี้เป็นดังนี้



รูปที่ 2.14 รูปแบบของ API Packet frame ประเภท Tx (Transmit) Request: 64-bit address

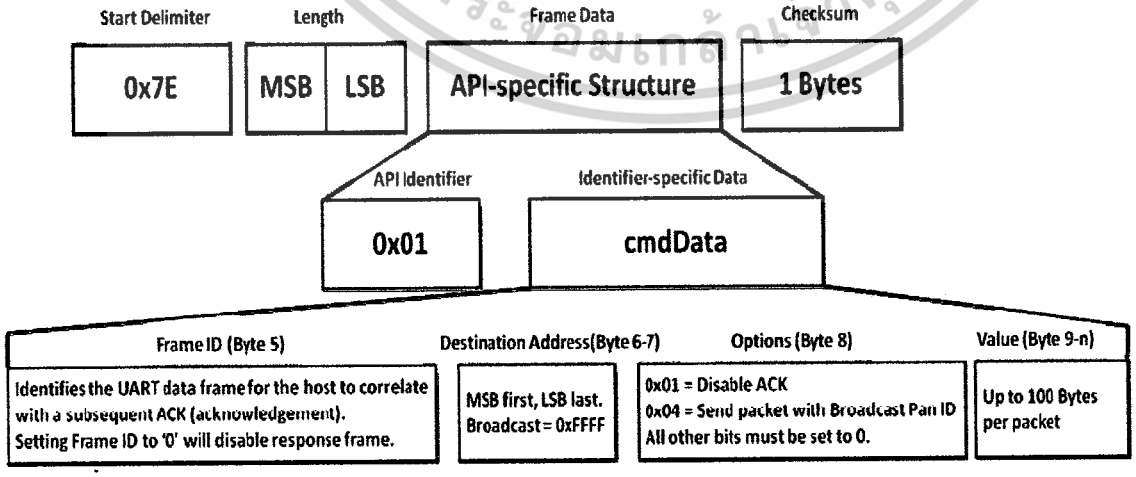
Destination Address คือ ที่อยู่ของโมดูลปลายทาง เมื่อถูกตั้งค่าเป็น 0x000000000000FFFF ข้อมูลจะถูกส่งแบบ broadcast

Option Byte คือ ตัวเลือกของการส่ง เมื่อถูกตั้งค่าเป็น 0x01 จะยกเลิกการส่ง ACK เมื่อตั้งค่าเป็น0x04 จะส่ง packet ไปยังทุกโมดูลที่อยู่ Pan ID เดียวกัน

RF Data คือ ข้อมูลที่ต้องการส่ง

2.3.2.7 Tx (Transmit) Request: 16-bit address

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

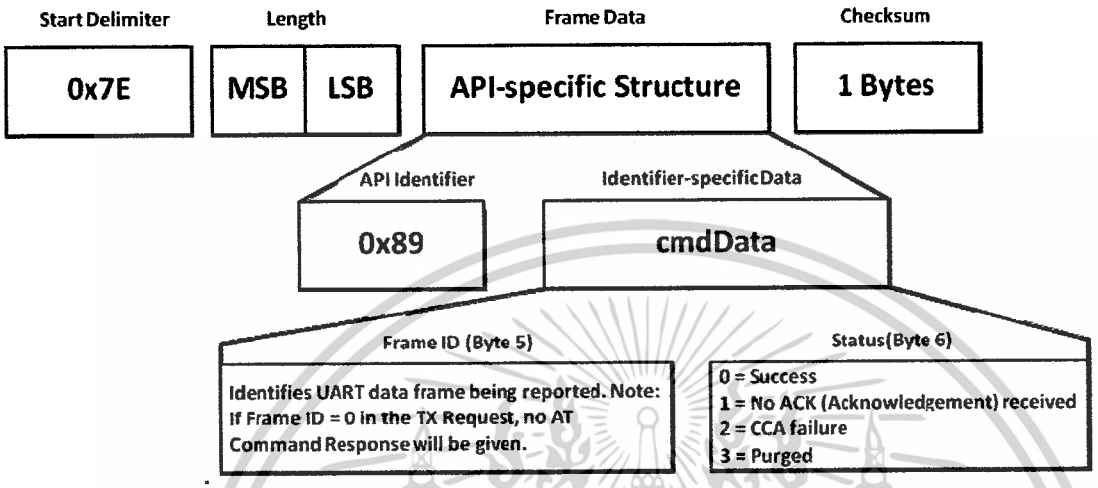


รูปที่ 2.15 รูปแบบของ API Packet frame ประเภท Tx (Transmit) Request : 16-bit address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.8 Tx (Transmit) Status

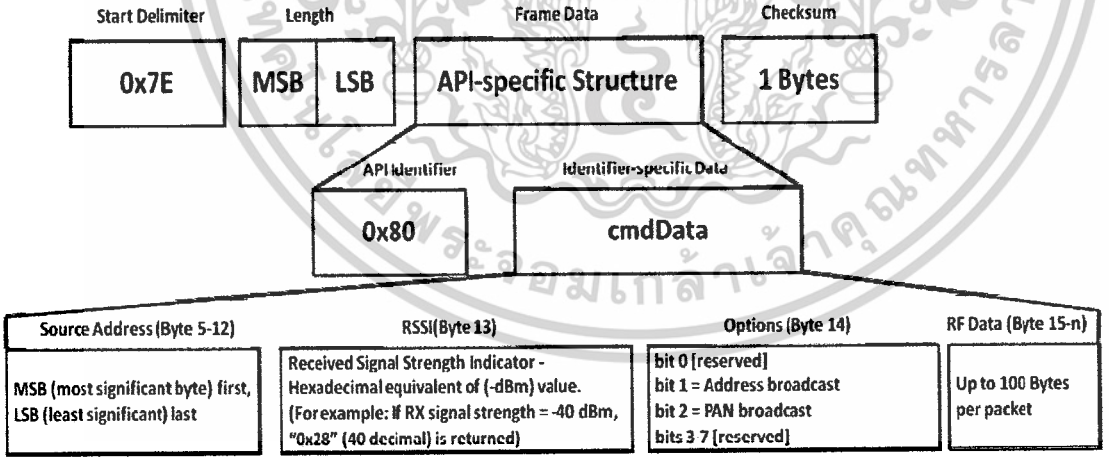
เมื่อการส่งข้อมูลเสร็จสมบูรณ์ โมดูลที่ได้รับข้อมูลจะส่งสถานะของการส่งกลับมายังโมดูลที่ส่งข้อมูล API frame ประเภทนี้เป็นสถานะของการส่งข้อมูลว่าสำเร็จหรือไม่ มี API Identifier เป็น 0x89โครงสร้างของ API frame ประเภทนี้เป็นดังนี้



รูปที่ 2.16 รูปแบบของ API Packet frame ประเภท Tx (Transmit)

2.3.2.9 RX (Receive) Packet 64-bit address

API frame ประเภทนี้มี API Identifier เป็น 0x80 เมื่อโมดูลได้รับข้อมูลจากการส่งแบบที่อยู่ 64 bitแล้ว frame packet ที่ได้รับจะมีโครงสร้างดังนี้



รูปที่ 2.17 รูปแบบของ API Packet frame ประเภท RX (Receive) Packet 64-bit address

Source Address คือ ที่อยู่ของโมดูลที่ส่งข้อมูลมายังตัวรับ

RSSI คือ ความแรงของสัญญาณ

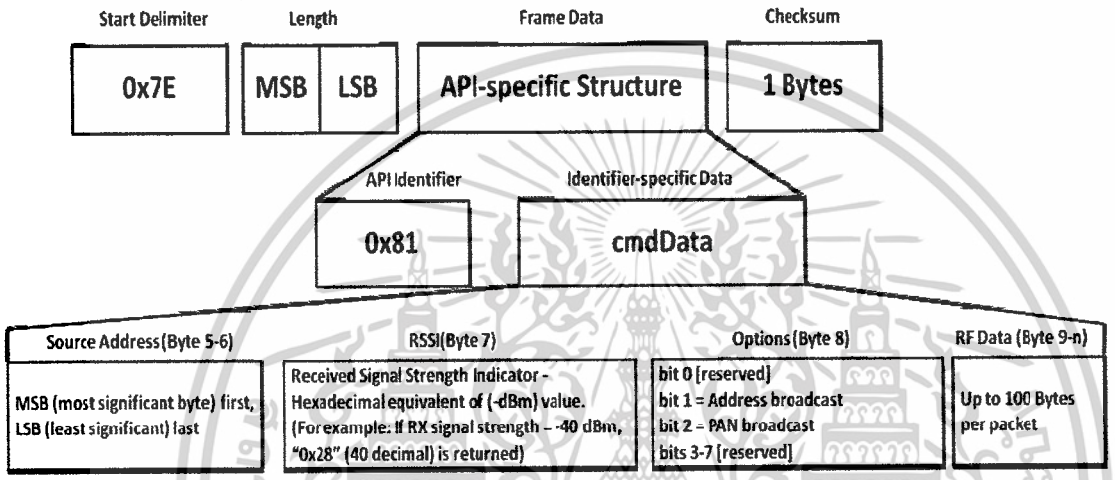
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

RF Data คือ ข้อมูลที่ได้รับ

2.3.2.10 RX (Receive) Packet 16-bit address

API frame ประเภทนี้มี API Identifier เป็น 0x81 เมื่อโมดูลได้รับข้อมูลจากการส่งแบบที่อยู่ 16 bitแล้ว frame packet ที่ได้รับจะมีโครงสร้างดังนี้



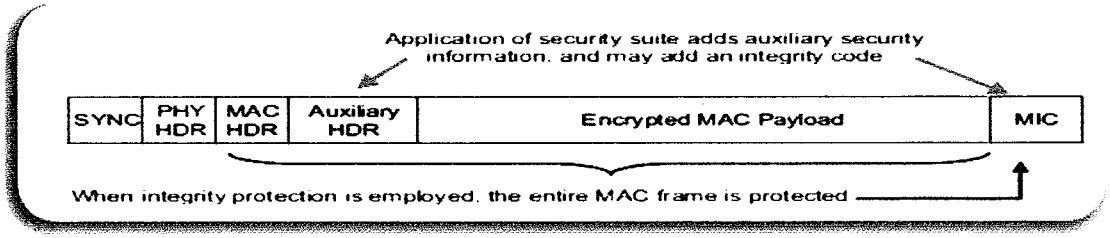
รูปที่ 2.18 รูปแบบของ API Packet frame ประเภท RX (Receive) Packet 16-bit address

2.3.2.11 ความปลอดภัยของ Zigbee

ความสำเร็จของ Zigbee network layer คือ การระบุตำแหน่งเครือข่ายเครือข่ายแบบประสานเวลา (Synchronization) ความปลอดภัยของเฟรม (Frames security) และการหาเส้น ทางของข้อมูล โดยที่ Zigbee มีการรักษาความปลอดภัยโดยใช้กุญแจสมมาตร (Symmetric key) มีการรักษาความปลอดภัยระหว่าง Layer ครอบคลุมทั้งเครือข่ายและ Application layer สนับสนุนความปลอดภัยระหว่างโนดคือต้นทางและปลายทางจะต้องมีการใช้กุญแจตัวเดียวกันในการเข้าถึงข้อ มูลนอกจากนั้นหากเฟรม Mac ต้องการความปลอดภัย MAC layer ก็จะช่วยรักษาความปลอดภัยใน MAC layer 802.15.4 AES มีกลไกการรักษาความปลอดภัยอย่างเหมาะสมมีการป้องกันความเป็นส่วนตัวความมั่นคงการระบุตัวตนของเฟรม MAC

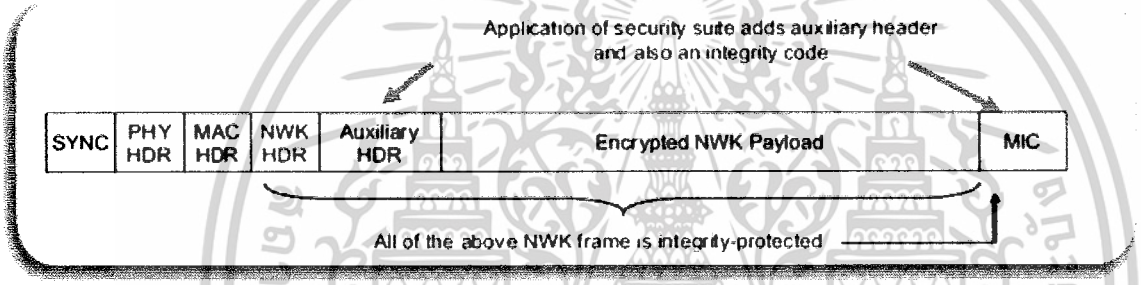
Auxiliary header ที่อยู่หน้า Mac payload เป็นตัวบอกว่าเฟรมนั้นมีการเข้ารหัสข้อมูลหรือไม่ความสมบูรณ์ของเฟรมตรวจสอบโดยคำนวณ MIC ที่อยู่หลัง MAC payload มีการใช้ AES ในการตรวจสอบตัวบุคคลในบางโอกาส สำหรับรูปแบบการใช้งาน AES มีการใช้หลายโหมด เช่น สำหรับการเข้ารหัสใช้โหมด Counter (CRT) การตรวจสอบความสมบูรณ์ของเฟรมใช้โหมด CBC-MAC และโหมด CCM ที่สามารถใช้ในการเข้ารหัสและการตรวจสอบความสมบูรณ์ของเฟรมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



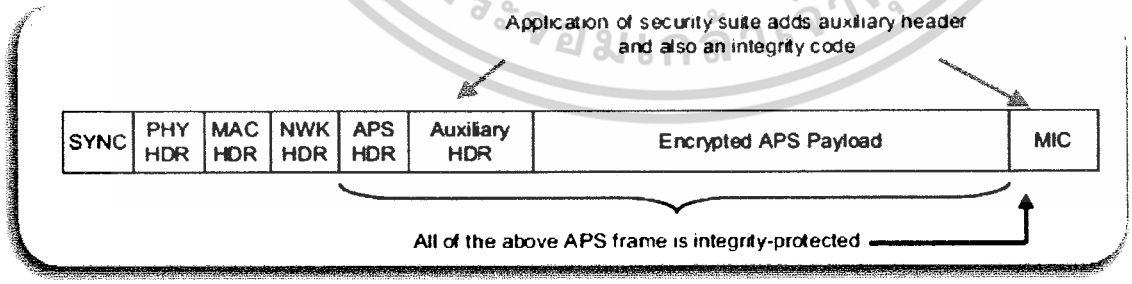
รูปที่ 2.19 เฟรม MAC กับการรักษาความปลอดภัย

ใน Network layer ใช้ CCM ในการเข้ารหัสเพราะ Network layer ใช้เฉพาะโหมด CCM สำหรับทุกๆ รูปแบบของการรักษาความปลอดภัย ข้อความที่มีการรักษาความปลอดภัยมีความ คล้ายคลึงกับ MAC frame



รูปที่ 2.20 การรักษาความปลอดภัยใน Network layer

ความปลอดภัยใน Application layer ทำงานคล้าย network และ MAC layer โดยใช้ "link key" หรือ "Network key" ในการป้องกันข้อมูลและห่อหุ้มข้อมูลเอาไว้ภายใน

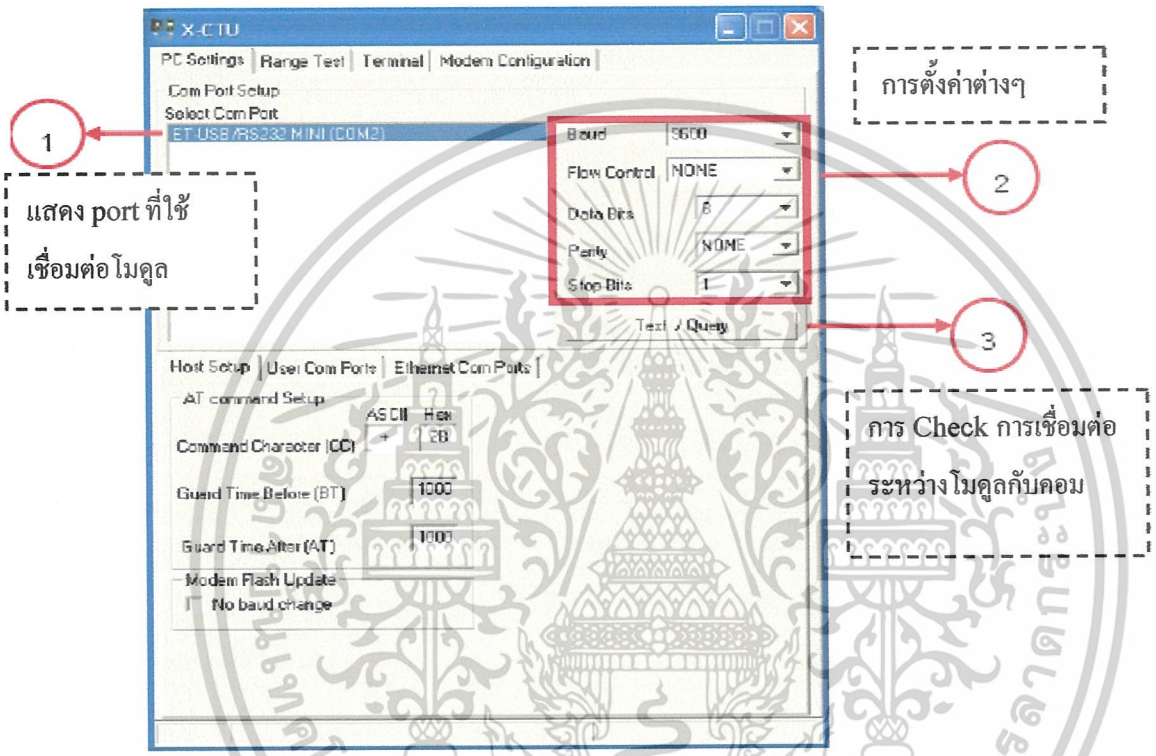


รูปที่ 2.21 การรักษาความปลอดภัยใน Application layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

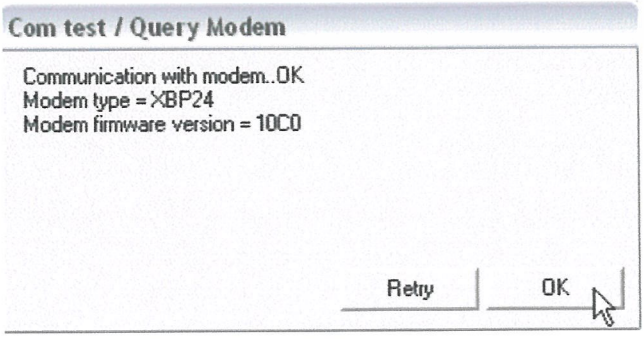
2.4. การใช้โปรแกรม X-CTU

เปิดโปรแกรม X-CTU โดยดับเบิลคลิกที่ไอคอน บน Desktop ของคอมพิวเตอร์ หรือคลิกที่ Start ⟶ All Programs ⟶ Digit-Max stream X-CTU หน้าต่างกำหนดการเชื่อมต่อจะปรากฏขึ้นมาซึ่งจากหน้าต่างของโปรแกรม X-CTUจะเห็นว่าเราสามารถที่จะกำหนด Baud rate ได้เองซึ่งค่าสูงสุดที่สามารถกำหนดได้คือ 230400 bps และสามารถกำหนด Flow Control, Data Bits, Parity และ Stop Bits และเวลาเราจะเชื่อมต่อเราก็ต้องเลือก Com Port ตามที่ตั้งไว้



รูปที่ 2.22 โปรแกรม X-CTU

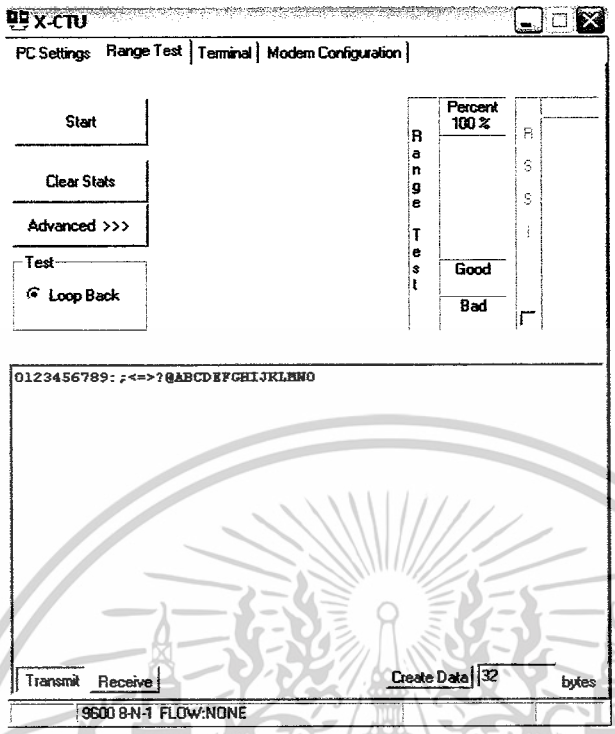
เมื่อการตรวจสอบระหว่างคอมพิวเตอร์กับบอร์ด Zigbee ค้นพบกันสำเร็จก็จะแสดงรูปด้านล่างนี้ขึ้นมาซึ่งภายในหน้าต่างนี้ก็จะบอกรายละเอียดเกี่ยวกับ รุ่นของโมดูลและ Version ของโปรแกรมที่โมดูลตัวนั้นใช้อยู่ในปัจจุบัน



รูปที่ 2.23 การพบโมดูล Zigbee

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

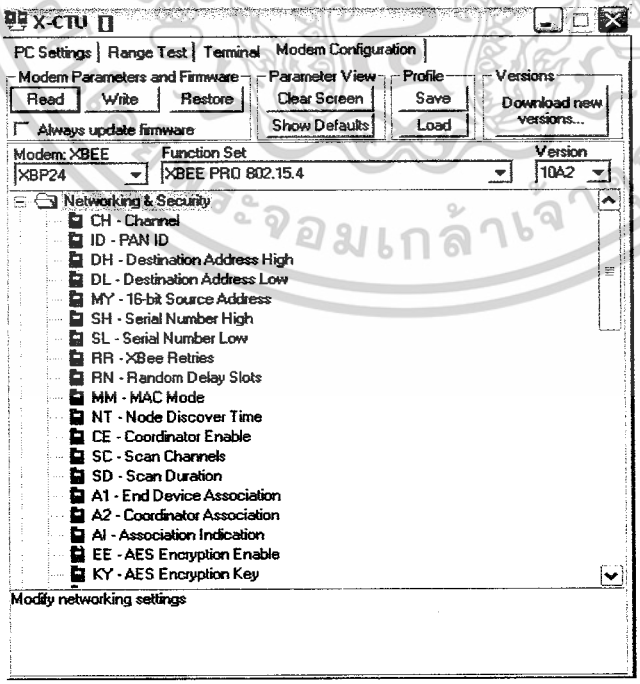
- Range Test



รูปที่ 2.24 โปรแกรม X-CTU

จากรูปที่ เป็นการแสดงในหน้าต่างที่ 2 ซึ่งในหน้าต่างนี้จะเป็นการทดสอบการส่งข้อมูลระหว่างกันซึ่งเราสามารถกำหนดขนาดของข้อมูล รูปแบบ และดูผลของการรับส่งข้อมูลได้จากหน้านี้

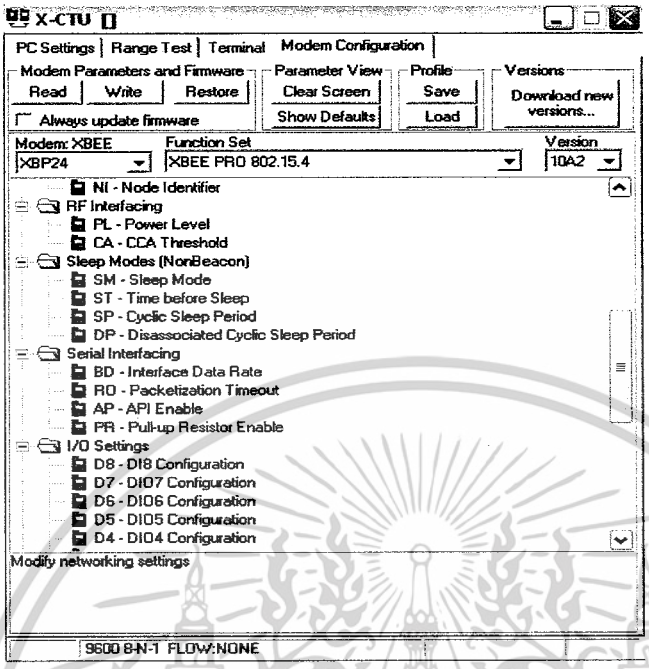
- Modem Configuration



รูปที่2.25 โปรแกรม X-CTU (ต่อ)

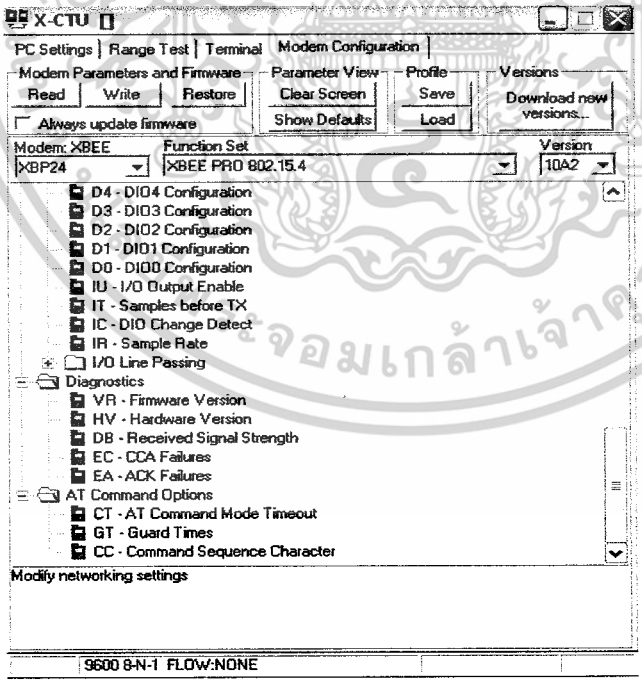
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Modem Configuration (ต่อ)



รูปที่ 2.26 โปรแกรม X-CTU (ต่อ)

- Modem Configuration (ต่อ)



รูปที่ 2.27 โปรแกรม X-CTU (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่เป็น Modem Configuration จะเห็นว่าเมื่อเราเชื่อมต่อโมดูลเข้ากับคอมพิวเตอร์ หลังจากนั้นก็กดปุ่ม Read ในกรอบ Modem Parameters and Firmware จะปรากฏข้อมูลชื่อรุ่นของโมดูล XBee-PRO ชื่อฟังก์ชั่น หมายเลขฟังก์ชั่นของเฟิร์มแวร์ และค่าพารามิเตอร์ต่างๆ เมื่อมาถึงขั้นตอนนี้ผู้ใช้งานสามารถเปลี่ยนค่าคอนฟิกูเรชั่นได้ตามต้องการ ไม่ว่าจะเป็นการเลือกอัตราบอดเรดใหม่ กำหนดรูปแบบการทำงานของขาพอร์ตของ XBee-PRO เนื่องจากขาพอร์ตของ XBee-PRO สามารถใช้งานเป็นอินพุตอนาลอกเพื่อเชื่อมต่อกับตัวตรวจจับที่ให้ผลการทำงานเป็นแรงดันส่งไปยังวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอลความละเอียด 10 บิตภายในโมดูล XBee-PRO ซึ่งผู้ใช้งานสามารถอ่านค่าออกไปได้ ดังนั้นจึงสามารถใช้งาน XBee-PRO ร่วมกับตัวตรวจจับทำงานเป็นโครงข่ายตัวตรวจจับไร้สาย (Wireless Sensor Network) ได้

- หากต้องการเปลี่ยนอัตราบอดเรดให้ไปที่หัวข้อ Serial Interfacing คลิกเลือกที่บรรทัด BD-Interface Data Rate จะปรากฏเมนูให้เลือกค่าอัตราบอดเรด 8 ค่า ตั้งแต่ 1,200 ถึง 115,200 บิตต่อวินาที
- หากต้องการเปลี่ยนการทำงานของขาพอร์ตให้เลือกไปที่หัวข้อ I/O Setting แล้วเลือกไปที่ขาพอร์ตที่ต้องการเปลี่ยนค่า จะปรากฏช่องให้เลือกฟังก์ชั่นการทำงาน ซึ่งมีด้วยกัน 5 แบบคือ
  - DISABLE หมายถึง ปิดการทำงาน (ปกติเป็นค่าตั้งต้น)
  - NA หมายถึง ยังไม่มีการกำหนดฟังก์ชั่น หรือสำรองไว้
  - ADC หมายถึง เลือกเป็นอินพุตอนาลอก
  - DI หมายถึง เลือกเป็นอินพุตดิจิตอล
  - DO LOW หมายถึง เลือกเป็นเอาต์พุตดิจิตอลลอจิกต่ำ
  - DO HIGH หมายถึง เลือกเป็นเอาต์พุตดิจิตอลลอจิกสูง
- จากนั้นกดปุ่ม Write รอสักครู่ สังเกตที่ด้านล่างของหน้าต่าง Modem Configuration จะแสดงข้อความเพื่อแจ้งสถานะการทำงาน หากเป็นดังรูปแสดงว่า การกำหนดค่าเสร็จสมบูรณ์

2.4.1 รีจิสเตอร์ที่ควรทราบของโมดูล XBee-PRO

การส่งข้อมูลและควบคุมระหว่างไมโครคอนโทรลเลอร์หรือคอมพิวเตอร์กับโมดูล XBee-PRO นั้นจะใช้การสื่อสารแบบอนุกรม USART ซึ่งโมดูล XBee-PRO สามารถใช้ความเร็วในการส่งข้อมูล (Baud rate) ได้ตั้งแต่ 1,200 จนถึง 115,200 บิตต่อวินาที (bps: bit per second) โดยค่าที่กำหนดเป็นค่าเริ่มต้นคือ 9,600 บิตต่อวินาที และสามารถเปลี่ยนความเร็วในการส่งข้อมูลได้ที่รีจิสเตอร์ BD ในการติดต่อสื่อสารระหว่างโมดูล XBee-PRO สามารถจัดเครือข่ายได้หลายรูปแบบ โดยการแยกช่องสัญญาณและเครือข่าย รีจิสเตอร์ที่ใช้สำหรับจัดการเกี่ยวกับเครือข่าย มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

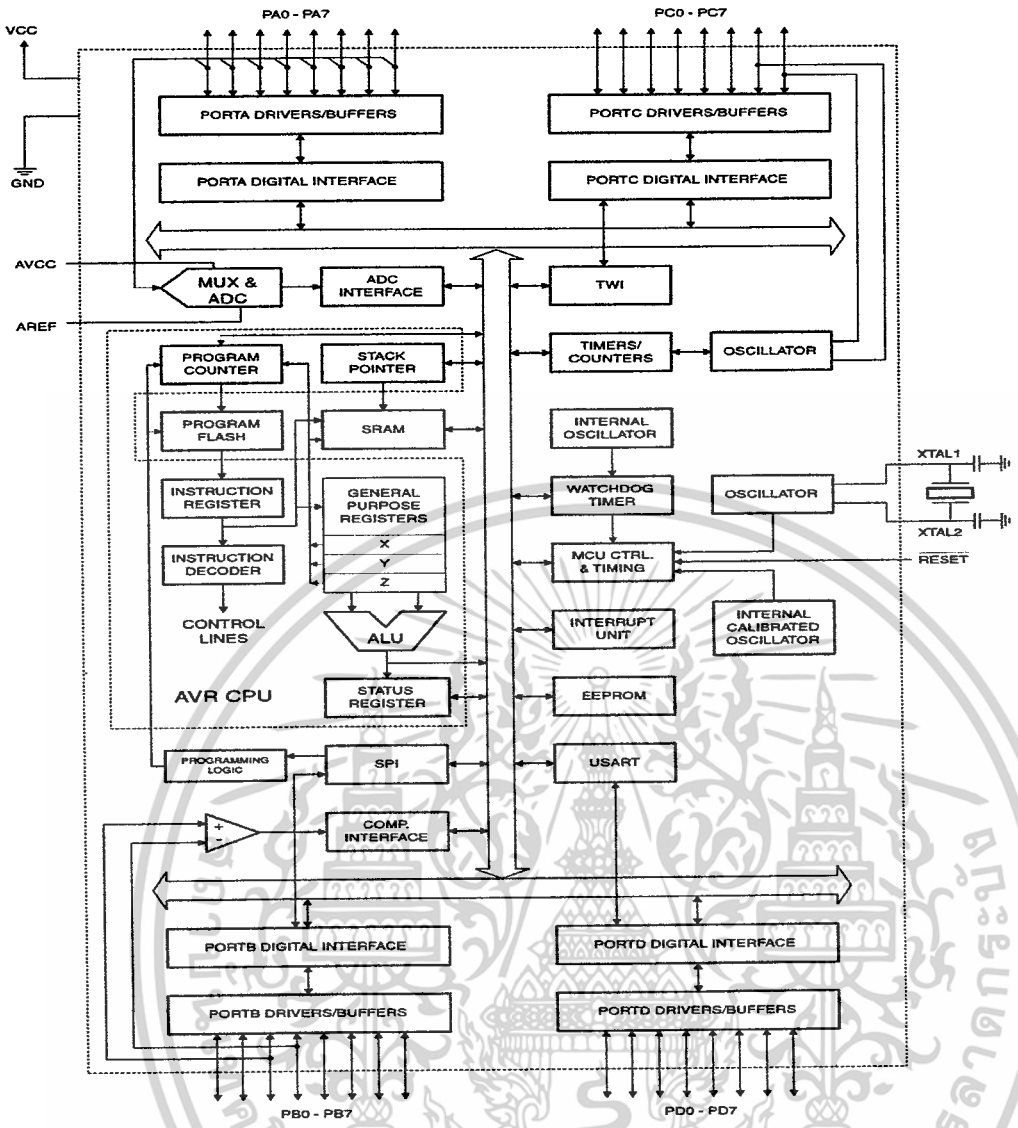
- รีจิสเตอร์ CH (Channel) ใช้กำหนดช่องสัญญาณ เลือกได้ตั้งแต่ช่องที่ 0x0C ถึง 0x17 แต่ละช่องไม่ สามารถส่งข้อมูลข้ามช่องสัญญาณกันได้
- รีจิสเตอร์ ID (PAN ID / Personal Area Network ID) ใช้กำหนดหมายเลขเครือข่าย เลือกค่าได้ตั้งแต่ 0x0000 จนถึง 0xFFFE โดยแต่ละเครือข่ายจะไม่สามารถส่งข้อมูลข้ามเครือข่ายได้ยกเว้นกำหนดด้วยค่า 0xFFFF จะสามารถส่งข้อมูลไปทุกเครือข่ายได้ แต่จะไม่สามารถรับข้อมูลจากเครือข่ายอื่นได้
- รีจิสเตอร์ MY (16-bit Source Address) ใช้กำหนดแอดเดรส 16 บิตของแต่ละโมดูล เลือกค่าได้ตั้งแต่ 0x0000 ถึง 0xFFFD และสามารถยกเลิกแอดเดรส 16 บิตนี้เพื่อไปใช้แอดเดรสขนาด 64 บิตที่รีจิสเตอร์ SH และ SL แทนได้เพื่อขยายให้มีจำนวนโมดูลลูกข่ายได้มากขึ้น โดยกำหนด MY เป็น 0xFFFE และ 0xFFFF
- รีจิสเตอร์ SH และ SL (Serial Number High / Low) เป็นรีจิสเตอร์เก็บค่าหมายเลขเฉพาะหรือ Serial number ของแต่ละโมดูล สามารถใช้เป็นแอดเดรส 64 บิต (SH รวมกับ SL) โดยต้องยกเลิกแอดเดรส 16 บิตที่รีจิสเตอร์ MY ก่อน ค่าในรีจิสเตอร์ SH และ SL ไม่สามารถเปลี่ยนแปลงได้
- รีจิสเตอร์ DH และ DL (Destination Address High / Low) ใช้กำหนดแอดเดรสของโมดูลตัวรับ
- ถ้าโมดูลตัวรับใช้รีจิสเตอร์ MY (แอดเดรส 16 บิต) ให้กำหนดค่าของรีจิสเตอร์ DH เป็น 0x0000 และDL เป็นค่า MY ของโมดูลตัวรับ
- ถ้าโมดูลตัวรับใช้รีจิสเตอร์ SH ร่วมกับ SL (แอดเดรส 64 บิต) ให้กำหนดค่าของรีจิสเตอร์ DH เป็นค่าของ SH และค่าของรีจิสเตอร์ DL เป็นค่าของ SL ของโมดูลตัวรับ

การตั้งค่าของโมดูล XBee-PRO ทำได้ 2 ทางคือ ใช้โปรแกรม X-CTU กับบอร์ด ZX-XBee ต่อกับคอมพิวเตอร์ทางพอร์ตอนุกรม อีกทางหนึ่งคือใช้ AT Command ซึ่งสามารถดูคำสั่งต่างๆ เพิ่มเติมได้จากไฟล์ในแผ่นซีดีรอม ก่อนการใช้โมดูล XBee-PRO ควรจะทำการตรวจสอบ ตั้งแต่ค่าแอดเดรสต่างๆ และรูปแบบของการส่งข้อมูลก่อนนำไปติดตั้งกับบอร์ดไมโครคอนโทรลเลอร์

2.5 ไมโครคอนโทรลเลอร์ AVR

ไมโครคอนโทรลเลอร์ AVR เป็นหนึ่งในไมโครคอนโทรลเลอร์ตระกูลใหม่จาก ATMEL มีสถาปัตยกรรมแบบ RISC (Advanced RISC architecture) คือ หนึ่งคำสั่งทำงานใช้สัญญาณนาฬิกาเพียง 1 ลูก (instructions in a signal clock cycle) เป็นไมโครคอนโทรลเลอร์ที่มีประสิทธิภาพ และความสามารถสูง แบ่งออกเป็นหลายอนุกรมในแต่ละอนุกรมยังแบ่งออกเป็นหลายเบอร์ เพื่อรองรับความต้องการที่แตกต่างของผู้ใช้งาน ในขณะที่ยังคงประสิทธิภาพที่เท่ากัน สำหรับไมโครคอนโทรลเลอร์ AVR ที่ใช้ในโครงงานนี้เบอร์ ATmega16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.28 บล็อกไดอะแกรม AVR (ATmega16)

คุณสมบัติที่สำคัญ

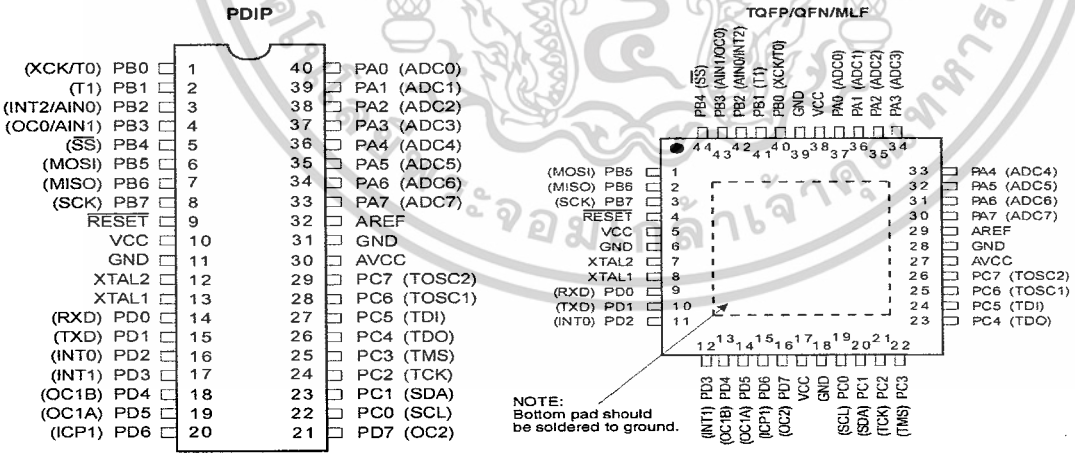
- สถาปัตยกรรมภายในเป็นแบบ Advanced RISC (Reduce Instruction Set Computer)
- มีคำสั่งควบคุมการทำงานมากกว่า 100 คำสั่ง โดยมีความเร็วในการประมวลผล 1 คำสั่งต่อ 1 สัญญาณนาฬิกา (1 MIP/1MHz)
- มีรีจิสเตอร์ใช้งานทั่วไปขนาด 8 บิต จำนวน 32 ตัว (ทำให้สะดวกต่อการพัฒนาโปรแกรมด้วยภาษา C เป็นอย่างมาก)
- ความเร็วในการทำงาน 1 MIPS ต่อ 1 MHz และมากถึง 16 MIPS เมื่อใช้ความถี่ที่ 16 MHz (ความสามารถในการใช้งานความถี่สัญญาณนาฬิกาขึ้นอยู่กับเบอร์ที่เลือกใช้งาน)
- หน่วยความจำ ROM แบบ Flash (มีโหมดป้องกันหน่วยความจำ) ขนาด 16 กิโลไบต์ (เขียน/ลบได้ 10,000 ครั้ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หน่วยความจำข้อมูลแบบ EEPROM (มีโหมดป้องกันหน่วยความจำ) ขนาด 512 ไบต์ (เขียน/ลบได้ 10,000 ครั้ง)
- หน่วยความจำข้อมูลแบบ SRAM 1 กิโลไบต์
- ไทเมอร์/เคาร์เตอร์ ทั้งแบบ 8 บิต และ 16 บิต พร้อมปรีสเกลเลอร์
- มีระบบตรวจสอบความผิดพลาดในการทำงานของซอฟต์แวร์ (Watchdog Timer with On-Chip Oscillator)
- โมดูลสร้างสัญญาณ PWM (Pulse Width Modulator) มีจำนวน 4 ช่อง
- มีโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล (ADC) ขนาด 10 มากถึง 8 ช่อง
- โมดูลเปรียบเทียบแรงดันอนาลอก (Analog Comparator)
- การสื่อสารข้อมูลอนุกรมมีทั้งแบบ UART (Universal Asynchronous Receiver Transmitters) และแบบ I^2C เป็นต้น
- พอร์ตอินพุตเอาต์พุตขึ้นอยู่กับเบอร์ AVR ที่เลือกใช้งาน มีตั้งแต่ 8 ขา จนมากกว่า 100 ขาพอร์ต (ATmega16 มีขาพอร์ตอินพุตเอาต์พุต 32 ขา)
- แรงดัน ไฟเลี้ยงและความเร็วในการทำงานขึ้นอยู่กับเบอร์ AVR ที่เลือกใช้งาน

2.5.1 ขาพอร์ตอินพุตเอาต์พุต

ขาพอร์ตอินพุตเอาต์พุตของไมโครคอนโทรเลอร์ ATmega16 มีจำนวน 40 ขาโดยแบ่งเป็นขาอินพุตเอาต์พุตอิสระจำนวน 32 ขาประกอบไปด้วย PA, PB, PC, PD ขนาด 8 บิตและขาพอร์ตที่เกี่ยวข้องกับจำนวนอนาลอกจำนวน 2 ขาพอร์ต คือ AREF และ AVCC



รูปที่ 2.29 ตัวถังแบบ PDIP และ TQFP/MLF และขาพอร์ตAVR (ATmega16)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

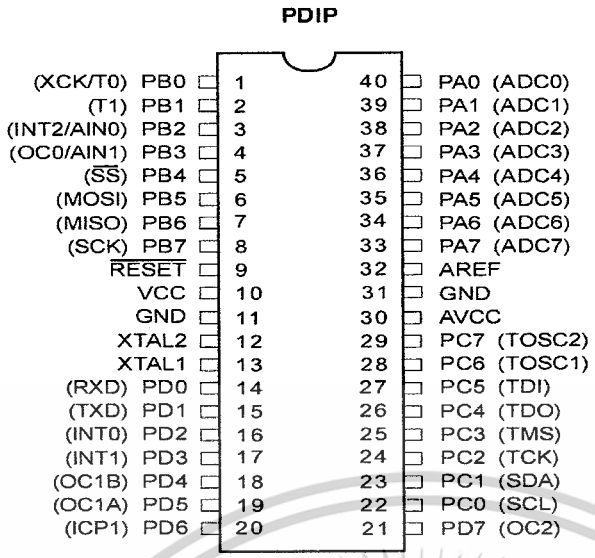
รายละเอียดในแต่ละขาพอร์ต

- VCC: ขาแรงดันไฟตรง
- GND: ขากราวด์
- Port A (PA0...PA7): ขาพอร์ตเป็นอินพุตเอาต์พุตดิจิตอล กำหนดการพูลอัพภายในขาพอร์ตได้ (internal pull-up register) และสามารถกำหนดใช้งานเป็นพอร์ตอินพุตสัญญาณอนาลอก (A/D Converter) ได้
- Port B (PB0...PB7): เป็นขาพอร์ตเป็นอินพุตเอาต์พุตดิจิตอล กำหนดการพูลอัพภายในขาพอร์ตได้ (internal pull-up register) และเป็นขาพอร์ตหน้าที่พิเศษอีกด้วย เช่น ขาสำหรับโปรแกรมชิพ ขาป้อนสัญญาณนาฬิกาภายนอก เป็นต้น
- Port C (PC0...PC7): นอกจากจะเป็นขาพอร์ตเป็นอินพุตเอาต์พุตดิจิตอล ที่กำหนดการพูลอัพภายในขาพอร์ตได้ (internal pull-up register) แล้วยังเป็นขาพอร์ตหน้าที่พิเศษ เช่น ขาเชื่อมต่อกับดีบัก และโปรแกรมด้วยการเชื่อมต่อแบบ JTAG เป็นต้น
- Port D (PD0…PD7): เป็นขาพอร์ตเป็นอินพุตเอาต์พุตดิจิตอล กำหนดการพูลอัพภายในขาพอร์ตได้ (internal pull-up register) และขาพอร์ตทำหน้าที่พิเศษ เช่น ขาเชื่อมต่อพอร์ตอนุกรมขาอินเตอร์รัปต์เนื่องจากสัญญาณภายนอก เป็นต้น
- RESET: ขารีเซตวงจร
- AXTL1: ต่อขาคริสตอลออสซิเลเตอร์ ช่องที่ 1 ด้านอินพุต
- AXTL2: ต่อขาคริสตอลออสซิเลเตอร์ ช่องที่ 2 ด้านเอาต์พุต
- AVCC: ขาแรงดันสำหรับพอร์ต A และโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล
- AREF: ขาแรงดันอนาลอกอ้างอิงสำหรับโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล

2.5.2 การควบคุมพอร์ตไมโครคอนโทรลเลอร์ AVR

ไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega16 ขนาด 40 ขา จะมีพอร์ตอินพุตเอาต์พุตสำหรับใช้งาน 4 พอร์ตคือ PORTA, PORTB, PORTC และ PORTD แต่ละพอร์ตมีขนาด 8 บิต (8 ขา) รวมแล้วมีขาพอร์ตสำหรับใช้งานทั้งหมดเท่ากับ 32 ขา แสดงดังรูป จากรูปจะเห็นได้ว่า ในแต่ละขาพอร์ตยังประกอบไปด้วยวงเล็บควบคู่กับขาพอร์ต เช่น (XCK/T0) PB0, (T1) PB1 อักษรที่แสดงอยู่ภายในวงเล็บเป็นการบอกว่าขาพอร์ตดังกล่าวสามารถทำหน้าที่ได้มากกว่าหนึ่งหน้าที่ โดยหน้าที่ที่เพิ่มเติมจะอ้างอิงกับคุณสมบัติของโมดูลต่างๆ ที่มีการใช้งานในไมโครคอนโทรลเลอร์ AVR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.30 ขาพอร์ตไมโครคอนโทรลเลอร์ AVR

2.5.2.1 รายละเอียดขาพอร์ตไมโครคอนโทรลเลอร์

ขาพอร์ต PORTA (PA0-PA7)

ขาพอร์ต	ขาพอร์ตฟังก์ชันพิเศษ
PA7	ADC7 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 7)
PA6	ADC6 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 6)
PA5	ADC5 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 5)
PA4	ADC4 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 4)
PA3	ADC3 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 3)
PA2	ADC2 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 2)
PA1	ADC1 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 1)
PA0	ADC0 (ขอพอร์ตอินพุตสัญญาณอนาลอกช่องที่ 0)

ตารางที่ 2.2 ขาพอร์ต PORTA (PA0-PA7)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาพอร์ต PORTB (PB0-PB7)

ขาพอร์ต	ขาพอร์ตฟังก์ชันพิเศษ
PB7	SCK (ขาสัญญาณนาฬิกาของระบบบัส SPI)
PB6	MISO (ขาสัญญาณอินพุตมาสเตอร์/ขาสัญญาณเอาท์พุตสเลฟ สำหรับบัส SPI)
PB5	MISI (ขาสัญญาณเอาท์พุตมาสเตอร์/ขาสัญญาณอินพุตสเลฟ สำหรับบัส SPI)
PB4	SS (ขาสัญญาณอินพุตเลือกสเลฟ สำหรับบัส SPI)
PB3	AIN1 (อินพุตสัญญาณด้านลบสำหรับโมดูลเปรียบเทียบสัญญาณอนาลอกอินพุต) OC0 (โมดูลเปรียบเทียบค่าเอาท์พุตกับไทเมอร์/เคาร์เตอร์ 0)
PB2	AIN0 (อินพุตสัญญาณด้านบวกสำหรับโมดูลเปรียบเทียบสัญญาณอนาลอกอินพุต) INT2 (อินพุตสัญญาณอินเตอร์รัปต์เนื่องจากสัญญาณภายนอกช่องที่ 2)
PB1	T1 (ไทเมอร์/เคาเตอร์1 และอินพุตรับสัญญาณจากภายนอกในโหมดเคาร์เตอร์1)
PB0	T0 (ไทเมอร์/เคาเตอร์0 และอินพุตรับสัญญาณจากภายนอกในโหมดเคาร์เตอร์0) XCK (อินพุตเอาท์พุตสัญญาณนาฬิกาจากภายนอกสำหรับโมดูล USART)

ตารางที่ 2.3 ขาพอร์ต PORTB (PB0-PB7)

ขาพอร์ต PORTC (PC0-PC7)

ขาพอร์ต	ขาพอร์ตฟังก์ชันพิเศษ
PC7	TOSC2 (ไทเมอร์ออสซิเลเตอร์ขา 2)
PC6	TOSC1 (ไทเมอร์ออสซิเลเตอร์ขา 1)
PC5	TDI (ขาเชื่อมต่อการดีบักวงจรอินพุต กับการเชื่อมต่อแบบ JTAG)
PC4	TDO (ขาเชื่อมต่อการดีบักวงจรเอาท์พุต กับการเชื่อมต่อแบบ JTAG)
PC3	TMS (ขาเชื่อมต่อกับดีบักวงจรกำหนดโหมด กับการเชื่อมต่อแบบ JTAG)
PC2	TCK (ขาเชื่อมต่อกับดีบักวงจรสัญญาณนาฬิกา กับการเชื่อมต่อแบบ JTAG)
PC1	SDA (ขาสัญญาณข้อมูลอินพุตเอาท์พุตสำหรับบัสข้อมูลอนุกรม 2 สาย หรือ I^2C บัส)
PC0	SCL (ขาสัญญาณนาฬิกาสำหรับบัสข้อมูลอนุกรม 2 สาย หรือ I^2C บัส)

ตารางที่ 2.4 ขาพอร์ต PORTC (PC0-PC7)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาพอร์ต PORTD (PD0-PD7)

ขาพอร์ต	ขาพอร์ตฟังก์ชันพิเศษ
PD7	OC2 (โมดูลเปรียบเทียบค่าเอาท์พุต กับไทเมอร์/เคาน์เตอร์ 2)
PD6	ICP1 (โมดูลอินพุตตรวจจับสัญญาณ กับไทเมอร์/เคาน์เตอร์ 1)
PD5	OC1A (โมดูลเปรียบเทียบสัญญาณเอาท์พุต A กับไทเมอร์/เคาน์เตอร์ 1)
PD4	OC1B (โมดูลเปรียบเทียบสัญญาณเอาท์พุต B กับไทเมอร์/เคาน์เตอร์ 1)
PD3	INT1 (อินเตอร์รัปต์เนื่องจากสัญญาณภายนอกช่องที่ 1)
PD2	INT0 (อินเตอร์รัปต์เนื่องจากสัญญาณภายนอกช่องที่ 0)
PD1	TXD (ขาเอาท์พุตสัญญาณสำหรับโมดูล USART)
PD0	RXD (ขาอินพุตสัญญาณสำหรับโมดูล USART)

ตารางที่ 2.5 ขาพอร์ต PORTD (PD0-PD7)

2.5.2.2 การใช้งานพอร์ตอินพุตเอาท์พุตของ AVR

ก่อนการใช้งานพอร์ตอินพุตเอาท์พุตดิจิตอลจะต้องกำหนดทิศทางของพอร์ตก่อนโดยกำหนดในรีจิสเตอร์ DDRx ว่าต้องการให้เป็นอินพุตหรือเอาท์พุตพอร์ตและต้องการให้อินพุตหรือเอาท์พุตด้วย รีจิสเตอร์ PINx หรือ PORTx รายละเอียดของรีจีสเตอร์ทั้ง 3 มีดังนี้

- รีจิสเตอร์ DDRx (Port x Data Direction Register)

รีจิสเตอร์ใช้ในการกำหนดทิศทางการทำงานของขาพอร์ต โดยหากกำหนดให้บิตใดเป็น 1 หรือลอจิกสูง (High) จะเป็นเอาท์พุตและหากกำหนดให้เป็น 0 หรือลอจิกต่ำ (Low) จะเป็นอินพุตรีจิสเตอร์นี้ประกอบไปด้วย DDRA, DDRB, DDRC และ DDRD

- รีจิสเตอร์ PINx (Port x Input Pin Address)

รีจิสเตอร์ที่ใช้สำหรับอินพุตค่าหรือรีจิสเตอร์ที่ใช้ในการอ่านสัญญาณได้อย่างเดียวเท่านั้นประกอบด้วยรีจิสเตอร์ PINA, PINB, PINC และ PIND

- รีจิสเตอร์ PORTx (Port x Data Register)

รีจิสเตอร์ที่ใช้สำหรับในการอินพุตหรือเอาท์พุตค่า ขึ้นอยู่กับการกำหนดค่าในรีจิสเตอร์ DDR ประกอบด้วยรีจิสเตอร์ PORTA, PORTB, PORTC และ PORTD

2.5.2.3 ความสัมพันธ์ระหว่างรีจิสเตอร์ DDRx กับบิต PUD

รีจีสเตอร์ DDRx นอกจากการกำหนดทิศทางของขาพอร์ตเพื่อทำหน้าที่เป็นอินพุตหรือเอาท์พุตแล้วยังมีความสัมพันธ์กับบิต PUD หรือบิตที่เกี่ยวข้องกับการพูลอัพ (Pull-up) สัญญาณที่ขาพอร์ตในรีจิสเตอร์ SFIOR (Special Function I/O Register) ซึ่งมีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DDxn	PORTxn	PUD (in SFIOR)	I/O	Pull-Up	สถานะ
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

ตารางที่ 2.6 การกำหนดรูปแบบของขาพอร์ต

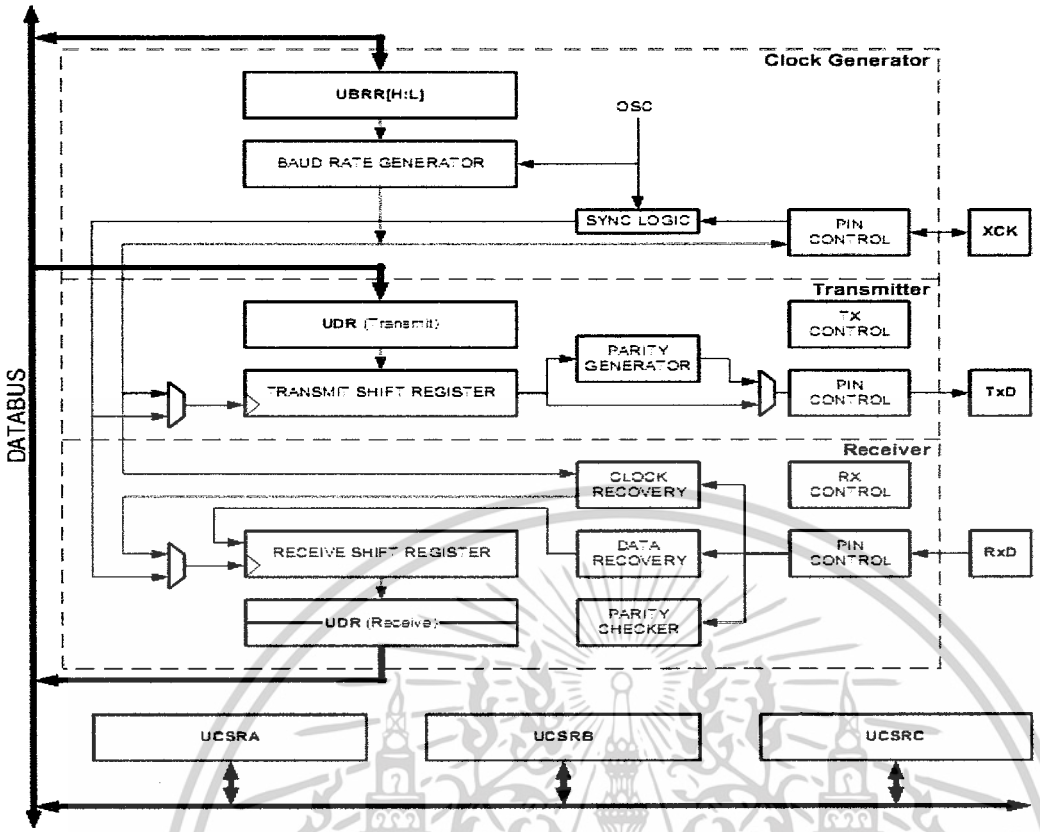
จากตาราง เมื่อกำหนดให้รีจิสเตอร์ควบคุมขาพอร์ตเป็นอินพุต โดยเขียนค่าลอจิก 0 ไปที่รีจิสเตอร์ DDxn แล้วมีการเขียน 0 ไปที่ขาพอร์ต PORTxn บิต PUD จะไม่มีผลต่อการพูลอัพสัญญาณที่ขาพอร์ต สถานะขาพอร์ตจะเป็นอิมพีแดนซ์สูง (Hi-Z) หรือสถานะของพอร์ตไม่เป็นทั้ง High หรือ Low หากกำหนดให้ขาพอร์ตเป็นอินพุตแล้วมีการเขียนค่า 1 ไปที่ขาพอร์ต การพูลอัพขาพอร์ตจะขึ้นอยู่กับ PUD ดังนี้

- บิต PUD เป็น 0 มีการพูลอัพที่ขาพอร์ต (สถานะขาพอร์ตจ่ายกระแสหากขาพอร์ตมีการต่อแบบพูลโล (Pull-low))
- บิต PUD เท่ากับ 1 ไม่มีการพูลอัพที่ขาพอร์ต (สถานะขาพอร์ตเป็นอิมพีแดนซ์สูง)

2.5.3 การสื่อสารข้อมูลอนุกรมผ่านโมดูล USART

ไมโครคอนโทรลเลอร์ AVR สามารถที่จะสื่อสารข้อมูลแบบอนุกรมได้โดยการใช้โมดูล USART (Universal Synchronous and Asynchronous serial Receiver and Transmitter) เพื่อสื่อสารข้อมูลผ่านทางพอร์ตอนุกรมได้ทั้งแบบซิงโครนัส (ข้อมูลมีความต่อเนื่อง มีการกำหนดสัญญาณมาตรฐานที่เหมือนกันทั้งทางด้านรับและด้านส่ง เพื่อให้การรับส่งมีความสัมพันธ์กัน) และอะซิงโครนัส (ข้อมูลไม่จำเป็นต้องต่อเนื่อง มีบิตเริ่มต้น (Start bit) บิตข้อมูล (Data bit) และบิตหยุด (Stop bit) มีบิตพาริตี้ (Parity bit) หรือไม่มีก็ได้) สำหรับไมโครคอนโทรลเลอร์ ATmega16 ขาพอร์ตอนุกรมกำหนดไว้ที่ขาพอร์ต PD0 (RXD) ใช้ในการรับข้อมูลอนุกรมและขาพอร์ต PD1 (TXD) ใช้ในการส่งข้อมูลอนุกรม บล็อกไดอะแกรมของโมดูล USART แสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

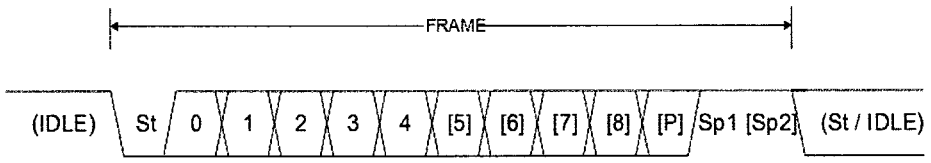


รูปที่ 2.31 บล็อกไดอะแกรมของโมดูล USART

จากรูปจะเห็นว่าโมดูล USART แบ่งออกเป็น 3 ส่วนด้วยกัน คือ

- ส่วนสร้างสัญญาณนาฬิกา (Clock Generator) เพื่อใช้ในการกำหนดอัตราบอดในการรับส่งข้อมูลโดยสามารถกำหนดได้ทั้งภายในและภายนอกผ่านทางขา XCK (Transfer Clock)
- ส่วนส่งข้อมูลอนุกรม (Transmitter) โดยส่งข้อมูลออกทางขาพอร์ต TxD
- ส่วนรับข้อมูลอนุกรม (Receiver) โดยการับข้อมูลจากขาพอร์ต RxD และมีรีจิสเตอร์ควบคุมการทำงาน 3 ตัวประกอบไปด้วย UCSRA, UCSRB และ UCSRC การส่งข้อมูลอนุกรมในรูปแบบอะซิงโครนัสจะเป็นการส่งข้อมูลเป็นเฟรม ลักษณะของเฟรมข้อมูลอนุกรมนี้ประกอบไปด้วย
    - บิตเริ่มต้นข้อมูล (Start bit)
    - บิตข้อมูล (Data bit)
    - พาริตี้บิต (Parity bit)
    - บิตหยุดข้อมูล (Stop bit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



St Start bit, always low.

(n) Data bits (0 to 8).

P Parity bit. Can be odd or even.

Sp Stop bit, always high.

รูปที่ 2.32 ลักษณะของเฟรมข้อมูลอนุกรม

2.5.3.1 คุณสมบัติที่สำคัญของโมดูล USART มีดังนี้

- การสื่อสารข้อมูลแบบฟูลดูเพลกซ์ (Full Duplex) ตัวรับและตัวส่งแยกอิสระต่อกัน สามารถรับและส่งข้อมูลได้ในเวลาเดียวกัน (พร้อมกัน)
- ทำงานได้ทั้งในโหมดซิงโครนัสและอะซิงโครนัส
- มีคุณสมบัติของพอร์ตอนุกรมครบถ้วน เช่น การกำหนดบิตข้อมูล การกำหนดบิตหยุด และการกำหนดบิตพาริตี้ เป็นต้น
- มีส่วนตรวจสอบความผิดพลาดของเฟรมข้อมูลและข้อมูลโอเวอร์รัน (Framing Error and Data OverRun Detection)
- โหมดการสื่อสารแบบมัลติโปรเซสเซอร์
- โหมดทวีคูณความเร็วในการสื่อสารข้อมูลแบบอะซิงโครนัส

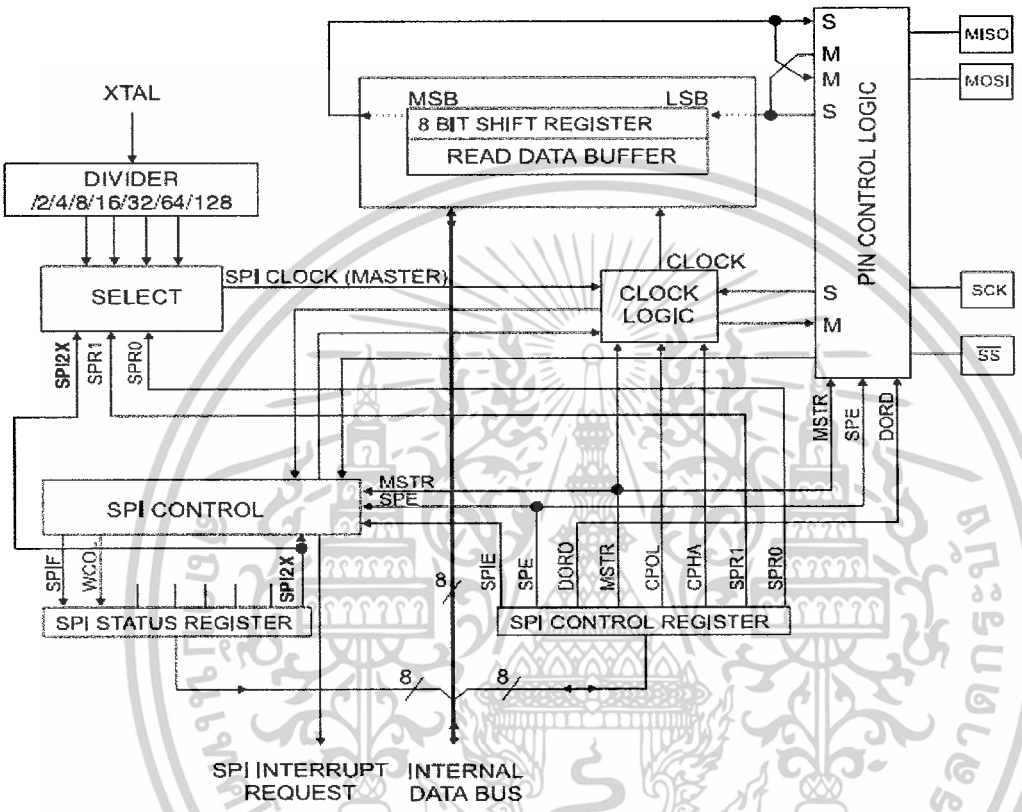
2.5.3.2 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานในโหมด USART

รีจิสเตอร์ UDR (USART I/O Data Register) รีจิสเตอร์อ่านเขียนข้อมูลขนาด 8 บิต โดยแบ่งออกเป็น 2 ตัวคือ RXB ใช้รับข้อมูลจากภายนอกเข้ามาในไมโครคอนโทรลเลอร์และ TXB ใช้สำหรับส่งข้อมูลออกจากไมโครคอนโทรลเลอร์ การอ่านเขียนข้อมูลจะทำกับรีจิสเตอร์ UDR โดยตรงรีจิสเตอร์ UCSRA (Control and Status Register A) รีจิสเตอร์ควบการทำงานและแสดงสถานะการทำงานของโมดูล USART ชุด A เกี่ยวข้องกับสถานะของการสื่อสารข้อมูลรีจิสเตอร์ UCSRB (Control and Status Register B) รีจิสเตอร์ควบการทำงานและแสดงสถานะการทำงานของโมดูล USART ชุด B เกี่ยวข้องกับบิตกำหนดอินเตอร์รัปต์และการกำหนดขนาดของข้อมูลแบบ 9 บิตข้อมูล (Data Bit) รีจิสเตอร์ UCSRC (Control and Status Register C) รีจิสเตอร์ควบการทำงานและแสดงสถานะการทำงานของโมดูล USART ชุด C เกี่ยวข้องกับการกำหนดอัตราบอตในการรับส่งข้อมูลรีจิสเตอร์ UBRRL และ UBRRH (USART Baud Rate Registers) รีจิสเตอร์กำหนดอัตราบอต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 การเชื่อมต่ออุปกรณ์อนุกรม (SPI)

การเชื่อมต่ออุปกรณ์อนุกรม (Serial Peripheral Interface) หรือ SPI เป็นการเชื่อมต่ออุปกรณ์เพื่อรับส่งข้อมูลแบบซิงโครนัส (มีสัญญาณนาฬิกาเข้ามาเกี่ยวข้อง) ระหว่างไมโครคอนโทรลเลอร์ AVR หรือจะเป็นอุปกรณ์ภายนอกที่มีการรับส่งข้อมูลแบบ SPI โครงสร้างบล็อกไดอะแกรมการรับส่งข้อมูลแบบ SPI ซึ่งแสดงดังรูป



รูปที่ 2.33 โครงสร้างบล็อกไดอะแกรมการรับส่งข้อมูลแบบ SPI

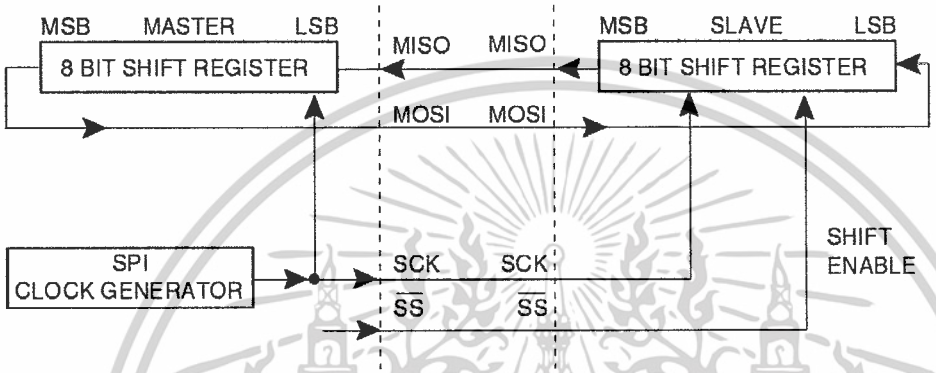
จากรูป การสื่อสารข้อมูล SPI ประกอบไปด้วยสายสัญญาณ 4 เส้น ดังนี้

- MOSI (Master Out-Slave In): ใช้ในการส่งข้อมูลออกจากอุปกรณ์มาสเตอร์ไปยังอุปกรณ์สเลฟ
- MISO (Master In-Slave Out): ใช้ในการรับข้อมูลจากอุปกรณ์สเลฟหรือมาสเตอร์อ่านข้อมูล
- SCK (Serial Clock): สายสัญญาณนาฬิกา ที่ใช้งานร่วมกันระหว่างมาสเตอร์กับสเลฟ
- SS (Slave Select): สายสัญญาณสำหรับเลือกอุปกรณ์สเลฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ที่ทำหน้าที่เป็นมาสเตอร์ โดยปกติแล้วจะเป็นไมโครคอนโทรลเลอร์ หรืออาจกล่าวได้ว่าอุปกรณ์มาสเตอร์จะต้องควบคุมอุปกรณ์สเลฟได้ อุปกรณ์สเลฟมักจะเป็นไอซีหน้าที่พิเศษต่างๆ เช่น ไอซีวัดอุณหภูมิ ไอซีฐานเวลานาฬิกาจริง (Real-Time Clock) หรืออาจเป็นไมโครคอนโทรลเลอร์ที่ทำงานในโหมดสเลฟ ก็ได้เช่นเดียวกัน

การใช้งานโมดูล SPI โดยปกติแล้วจะใช้สายสัญญาณทั้ง 4 เส้นที่กล่าวมา หากเป็นการต่อระหว่างไมโครคอนโทรลเลอร์ตัวหนึ่งทำหน้าที่เป็นมาสเตอร์ และอีกตัวหนึ่งทำหน้าที่เป็นสเลฟ จะใช้สายสัญญาณทั้ง 4 เส้น ดังรูป



รูปที่ 2.34 การใช้งานโมดูล SPI ที่ใช้สายสัญญาณทั้ง 4 เส้น

สำหรับการเชื่อมต่อกับอุปกรณ์สเลฟที่ไม่ได้เป็นไมโครคอนโทรลเลอร์ จะมีขาสำหรับใช้ในการกำหนดการทำงานของอุปกรณ์สเลฟเรียกว่า สายสัญญาณ EN (ENABLE) เพิ่มเติมขึ้นมาและสายสัญญาณ SS จะไม่มีการใช้งาน

2.5.4.1 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานในโหมด SPI

- รีจิสเตอร์ SPCR (SPI Control Register): รีจิสเตอร์ควบคุมการทำงานของโมดูล SPI เพื่อกำหนดโหมดการทำงาน การใช้งานอินเตอร์รัปต์ รวมถึงการกำหนดความถี่สัญญาณนาฬิกาในการรับส่งข้อมูล

- รีจิสเตอร์ SPSR (SPI Status Register): รีจิสเตอร์แสดงสถานการณ์ทำงานของโมดูล SPI เช่น การเซตแฟลก SPIF การตรวจสอบการชนกันของข้อมูล (Collision Flag) เป็นต้น

- รีจิสเตอร์ SPDR (SPI Data Register): รีจิสเตอร์ใช้ในการรับส่งข้อมูลของโมดูล SPI ข้อมูลการรับและส่งจะถูกกำหนดในรีจิสเตอร์นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4.2 หลักการสำคัญของการใช้ SPI

สถานะเริ่มต้นการทำงานของอุปกรณ์สเลฟ (Slave) จะเริ่มต้นด้วย Low ที่ขาสัญญาณ ENABLE อุปกรณ์ SPI จะมีเงื่อนไขให้ส่งข้อมูลที่ขอบขาขึ้นหรือขอบขาลงของขาสัญญาณ SCK แตกต่างกันไปขึ้นอยู่กับข้อกำหนดของอุปกรณ์ SPI นั้น นอกจากนี้ต้องคำนึงถึงความถี่สูงสุดของสัญญาณนาฬิกาของอุปกรณ์สเลฟที่ใช้งาน โดยค่าความถี่สูงสุดของสัญญาณนาฬิกาที่ใช้ต้องมีค่าไม่เกินความถี่สัญญาณนาฬิกาของอุปกรณ์สเลฟ

2.5.5 การเชื่อมต่ออุปกรณ์อนุกรมด้วยสาย 2 เส้น (TWI)

Inter-IC (I^2C-bus) เป็นมาตรฐานการสื่อสารข้อมูลอนุกรมระหว่างไอซีด้วยสายสัญญาณเพียง 2 เส้น เป็นการพัฒนาของ Philips รูปแบบการสื่อสารข้อมูลแบบ I^2C-bus มีลักษณะสำคัญดังนี้

- ใช้สายสัญญาณในการสื่อสารข้อมูลเพียง 2 เส้นเท่านั้น ซึ่งประกอบไปด้วยสายข้อมูลอนุกรม (Serial Data Line) หรือ SDA และสายสัญญาณนาฬิกาอนุกรม (Serial Clock Line) หรือ SCL
- มีอุปกรณ์ควบคุมการส่ง-รับข้อมูลซึ่งเรียกว่า Master และอุปกรณ์ปลายทางจะเรียกว่า Slave โดยสามารถเชื่อมต่อ Master/Slave ได้มากว่าหนึ่งในบัส I^2C เพียงบัสเดียว แต่ในความเป็นจริงแล้วการมี Master ตั้งแต่ 2 Master นั้นหากมีการโอนถ่ายข้อมูลพร้อมกันจะตรวจพบการชนกันของข้อมูลทำให้ข้อมูลที่โอนถ่ายนั้นไม่สมบูรณ์ ดังนั้นการโอนถ่ายข้อมูลจะต้องทำเมื่อบัสว่างเท่านั้น สำหรับการแยกแยะแต่ละ Slave นั้นจะใช้แอดเดรสเป็นตัวกำหนด ซึ่งส่วนหนึ่งของแอดเดรสถูกกำหนดมาแล้วจากทางผู้ผลิตชิพไอซีระบบบัส I^2C
- สื่อสารข้อมูลอนุกรมขนาด 8 บิตสองทิศทางด้วยความเร็ว 100 kbps

2.5.5.1 การสื่อสารข้อมูลบนระบบบัส I^2C

เนื่องจากบัส I^2C สามารถเชื่อมต่อกับอุปกรณ์บนบัสได้มากกว่าหนึ่งและเป็นการสื่อสารข้อมูลเพียงสาย 2 เส้น จึงต้องมีการกำหนดรูปแบบการสื่อสารข้อมูลและเชื่อมต่ออุปกรณ์

เงื่อนไขการรับส่งข้อมูลบนบัส I^2C

- การรับส่งข้อมูลจะมีได้เมื่อบัสว่างเท่านั้น
- ระหว่างรับส่งข้อมูล สายสัญญาณ SDA จะต้องคงสถานะไว้เมื่อสายสัญญาณ SCL เป็น High หากมีการเปลี่ยนสัญญาณ SDA ขณะที่สายสัญญาณ SCL เป็น High สายสัญญาณจะถูกแปลความว่าเป็นสภาวะเริ่มต้นหรือสภาวะหยุดได้

การรับส่งข้อมูลระหว่างอุปกรณ์ Master กับอุปกรณ์ Slave มีขั้นตอนดังต่อไปนี้

- เริ่มต้นรับส่งข้อมูลด้วย สภาวะเริ่มต้น
- ส่งข้อมูลแอดเดรสติดต่อกับอุปกรณ์โดยอ้างอิงแบบ 7 บิต หรือ 10 บิต
- รับ/ส่ง ข้อมูล
- หยุดรับส่งข้อมูลด้วย สภาวะหยุด

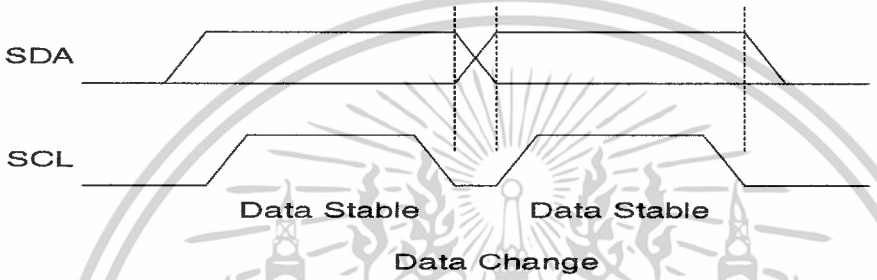
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.5.2 การสื่อสารข้อมูลบนระบบบัส I^2C กับโมดูล TWI

ไมโครคอนโทรลเลอร์ AVR สามารถที่จะเชื่อมต่อกับอุปกรณ์ที่มีการติดต่อในรูปแบบ I^2C ด้วยโมดูลเรียกว่า TWI (Two-wire Serial Interface) โดยใช้สายสัญญาณ 2 เส้นตามที่กล่าวมาข้างต้น และมีรูปแบบการรับส่งข้อมูลดังนี้

การส่งบิตข้อมูล

รูปแบบของบิตข้อมูลบนโมดูล TWI แสดงดังรูปสายสัญญาณ SDA จะต้องคงสถานะลอจิก "1" (High) หรือ ลอจิก "0" (Low) เมื่อสายสัญญาณ SCL มีสถานะลอจิก "1" การเปลี่ยนแปลงบนสายสัญญาณ SDA จะทำได้ก็ต่อเมื่อสายสัญญาณ SCL เป็นลอจิก "0" เท่านั้น



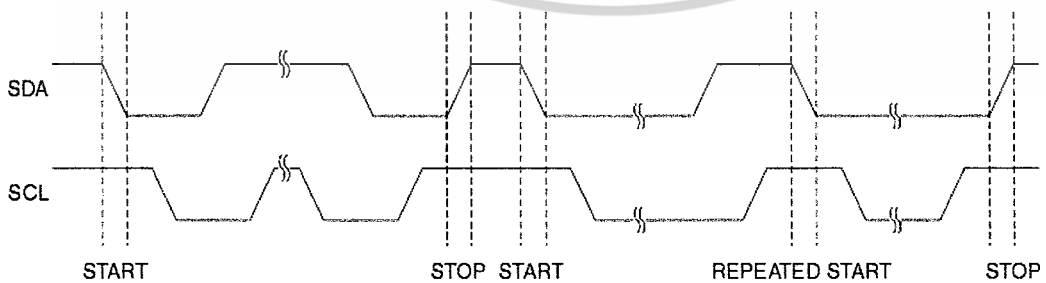
รูปที่ 2.35 รูปแบบของบิตข้อมูลบนโมดูล TWI

โดยที่

- ถ่ายโอนข้อมูล 1 สายสัญญาณ SDA คงที่ลอจิก "1" ในขณะที่สายสัญญาณ SCL เป็น "1"
- ถ่ายโอนข้อมูล 0 สายสัญญาณ SDA คงที่ลอจิก "0" ในขณะที่สายสัญญาณ SCL เป็น "1"

การเริ่มต้นและการหยุดการรับส่งข้อมูล

ก่อนที่จะเริ่มต้นรับและส่งข้อมูลได้นั้น จะต้องสร้างสัญญาณการเริ่มต้นก่อนเพื่อให้อุปกรณ์ฝ่ายรับพร้อมหรือรับรู้การเริ่มต้นการรับและส่งข้อมูล เมื่อรับส่งข้อมูลเรียบร้อยแล้วจะต้องสร้างสัญญาณหยุดการรับส่งข้อมูลเพื่อเป็นการบอกถึงการสิ้นสุดการรับส่งข้อมูลแล้ว โดยรูปแบบของสัญญาณจะแสดงดังรูป



รูปที่ 2.36 รูปแบบของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

- สภาวะเริ่มต้น (START condition) สายสัญญาณ SDA จะเปลี่ยนจาก “1” เป็น “0” ในขณะที่ SCL เป็น “1”
- สภาวะหยุด (STOP condition) สายสัญญาณ SDA จะเปลี่ยนจาก “0” เป็น “1” ในขณะที่ SCL เป็น “1”

2.5.6 การใช้งานโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอล

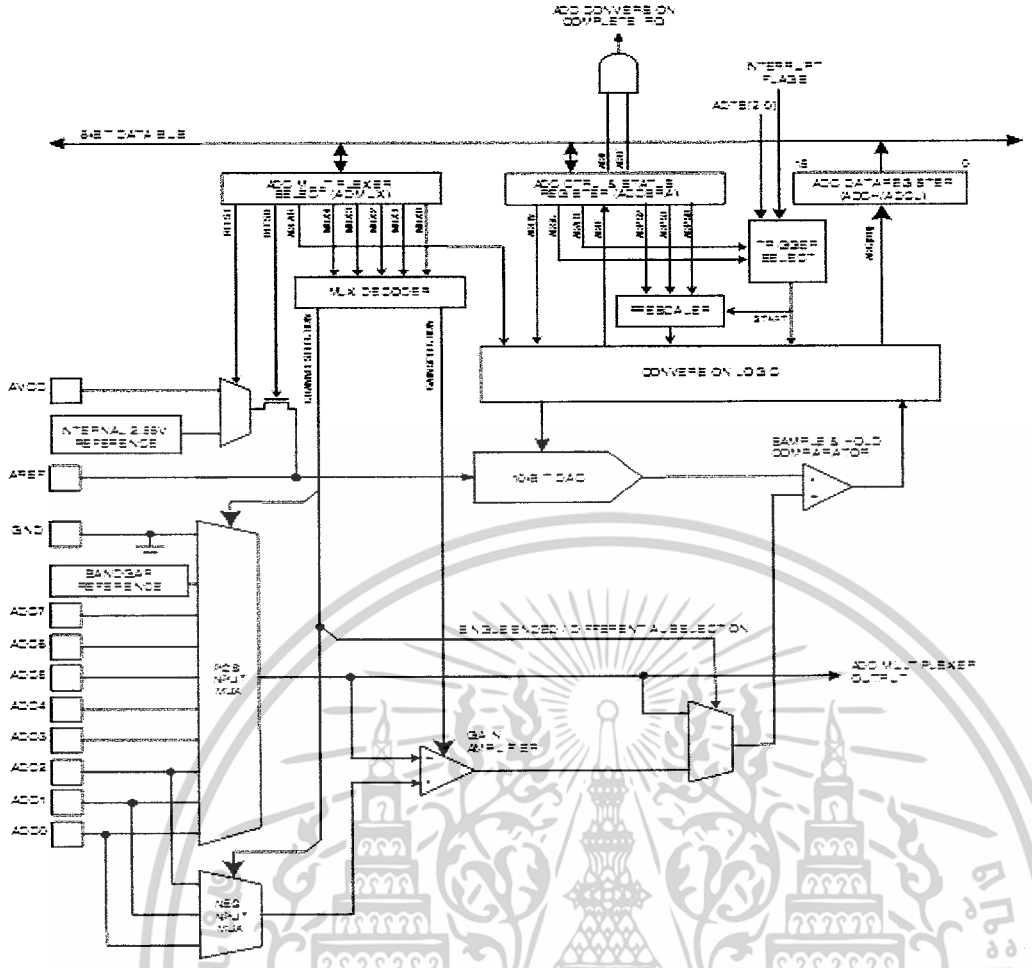
ไมโครคอลโทรลเลอร์ AVR เบอร์ ATmega16 มีโมดูลแปลงสัญญาณอนาลอกเป็นดิจิตอลหรือ ADC (Analog to Digital Converter) ความละเอียดขนาด 10 บิต (10-bit Resolution) ที่แรงดัน +5V หมายถึงเมื่อแปลงเป็นสัญญาณดิจิตอลแล้วจะได้ค่าตัวเลขอยู่ระหว่าง 0-1024 โดยมีรูปแบบการแปลงลักษณะอนาลอกเป็นดิจิตอลแบบซัคเซสซีฟ แอพพร็อกซิเมชั่น (Successive Approximation ADC) คือการแปลงแบบประมาณค่า โดยการสุ่มค่าดิจิตอลแล้วแปลงเป็นแรงดันอนาลอกภายในโมดูล เพื่อใช้เปรียบเทียบกับแรงดันอนาลอกด้านอินพุต เมื่อเปรียบเทียบได้ค่าแรงดันเท่ากันโมดูล ADC จะให้ผลลัพธ์ออกมาเป็นค่าดิจิตอล ซึ่งการใช้วิธีการนี้เป็นที่นิยมเพราะมีความเที่ยงตรงสูงและทำงานได้อย่างรวดเร็ว

โมดูล ADC จะมีจำนวน 8 ช่องอินพุตสัญญาณใช้หลักการมัลติเพล็กซ์ (Multiplexer) เพื่อเลือกการทำงานในแต่ละช่องกำหนดไว้ที่ขาพอร์ต A โดยมีแรงดันอินพุตระหว่าง 0 V (GND) ถึง VCC (แรงดันอินพุตที่ขา VCC ของไมโครคอนโทรลเลอร์ AVR) ผ่านวงจรสุ่มและเก็บค่า (Sample and Hold)

รีจิสเตอร์ที่เกี่ยวข้องกับโมดูลเปรียบเทียบแรงดันอนาลอก

- รีจิสเตอร์ ADMUX (ADC Multiplexer Selection Register) รีจิสเตอร์กำหนดแรงดันอ้างอิง (Voltage Reference) รูปแบบการเก็บข้อมูลและการกำหนดอินพุตอนาลอกอ้างอิงบวกและลบ
- รีจิสเตอร์ ADCSRA (ADC Control and Status Register) รีจิสเตอร์กำหนดและแสดงสถานะการทำงานของโมดูล ADC
- รีจิสเตอร์ ADCL และ ADCH (The ADC Data Register) รีจิสเตอร์เก็บข้อมูลที่ได้จากการแปลงสัญญาณอนาลอกเป็นดิจิตอล
- รีจิสเตอร์ SFIOR (Special Function IO Register) รีจิสเตอร์กำหนดการกระตุ้นจากแหล่งสัญญาณภายนอกให้กับโมดูล ADC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.37 วงจรโมดูล ADC

2.5.7 การสื่อสารอนุกรมแบบ RS-232

ในกรณีนี้จะต้องทำการติดตั้งไอซี Line Driver เพื่อเปลี่ยนระดับสัญญาณทางไฟฟ้าของขาสัญญาณสำหรับ รับ-ส่ง ข้อมูลแบบ TTL ของ MCU (RX และ TX) ให้เป็นระดับสัญญาณทางไฟฟ้าแบบ RS-232 (±12V) โดยการติดตั้งไอซีเบอร์ MAX232 เพื่อทำหน้าที่เปลี่ยนระดับสัญญาณ TTL จากขาสัญญาณส่งข้อมูล (TX) ของ MCU ให้เป็นระดับสัญญาณ ±12V สำหรับส่งไปยังขารับสัญญาณ (RX) ของอุปกรณ์ภายนอก และในทางกลับกัน ก็จะทำหน้าที่เปลี่ยนระดับสัญญาณส่ง (TX) แบบ RS-232 (±12V) จากอุปกรณ์ภายนอก ให้กลับมาเป็นระดับ TTL เพื่อส่งให้กับขารับข้อมูล (RX) ของ MCU ด้วย โดยเมื่อเปลี่ยนระดับสัญญาณในการรับส่งข้อมูลจาก TTL มาเป็นแบบ RS-232 นี้แล้วจะทำให้สามารถทำการ รับ-ส่ง ข้อมูลกับอุปกรณ์ภายนอกที่ใช้ระดับสัญญาณทางไฟฟ้าในการ รับ-ส่ง แบบเดียวกัน (RS-232) ได้ไกลขึ้นประมาณ 50ฟุต หรือ ประมาณ 15 เมตร โดยสามารถทำการ รับ-ส่ง ข้อมูลกับอุปกรณ์ต่างๆได้ในลักษณะของตัวต่อตัว (Point-to-Point) เท่านั้นสำหรับสายสัญญาณที่จะนำมาใช้สำหรับทำการสื่อสารแบบ RS-232 นั้น จะใช้สัญญาณเพียง 2-3เส้น เท่านั้น ทั้งนี้ขึ้นอยู่กับความต้องการในการสื่อสารว่าต้องการสื่อสารแบบทิศทางเดียวหรือสองทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสื่อสาร RS-232 แบบสองทิศทาง ซึ่งจะมีทั้งการรับข้อมูลและส่งข้อมูลไปมา ระหว่างด้านรับและด้านส่ง โดยในกรณีนี้จะต้องใช้สายสัญญาณจำนวน 3 เส้น สัญญาณรับข้อมูล (RXD) สัญญาณส่งข้อมูล (TXD) และสัญญาณอ้างอิง (GND) โดยในการเชื่อมต่อสายนั้นจะต้องทำการสลับสัญญาณกับอุปกรณ์ปลายทางด้วย คือ สัญญาณส่ง (TXD) จากบอร์ด CPJR51-ADU842 จะต้องต่อเข้ากับสัญญาณรับ (RXD) ของอุปกรณ์ และสัญญาณส่ง (TXD) จากอุปกรณ์ก็ต้องต่อกับสัญญาณรับ (RXD) ของบอร์ด ส่วนสัญญาณอ้างอิง (GND) จะต้องต่อตรงถึงกัน จึงจะสามารถทำการ รับ-ส่ง ข้อมูลกันได้

- การสื่อสาร RS-232 แบบทิศทางเดียว ซึ่งอาจเป็นการรอรับข้อมูลจากด้านส่งเพียงอย่างเดียวหรืออาจเป็นการส่งข้อมูลออกไปยังปลายทางเพียงอย่างเดียว โดยไม่มีการโต้ตอบข้อมูลซึ่งกันและกัน ซึ่งวิธีนี้จะใช้สาย สัญญาณเพียง 2 เส้น เท่านั้น โดยถ้าเป็นทางด้านส่ง ก็จะต่อเพียงสัญญาณส่ง (TXD) และสัญญาณอ้างอิง (GND) แต่ถ้าเป็นทางด้านรับ ก็จะต่อเพียงสัญญาณรับ (RXD) และ สัญญาณอ้างอิง (GND) เท่านั้นโดยขั้วต่อของสัญญาณ RS-232 ของบอร์ด CP-JR51-ADU842 นั้น จะเป็นจุดเชื่อมต่อของสัญญาณ รับ-ส่ง ข้อมูล u3617 ที่เปลี่ยนระดับสัญญาณเป็นแบบ RS-232 แล้ว ซึ่งจะมีลักษณะเป็นแบบขั้ว CPAขนาด 4 PIN สำหรับใช้เป็นจุดเชื่อมต่อสัญญาณ รับ-ส่ง ข้อมูลกับอุปกรณ์ภายนอก ซึ่งจะเห็นได้ว่าขั้วต่อสัญญาณ RS-232 ของบอร์ดนั้น จะมีทั้งหมด 4 เส้น แต่ในการ รับ-ส่ง ข้อมูลแบบปกตินั้นจะใช้สัญญาณเพียงแค่ 3 เส้น คือ RXD,TXD และ GND เท่านั้น ส่วน +VCC ซึ่งเป็นไฟเลี้ยงวงจร +5V นั้น จะไม่จำเป็นต้องนำมาใช้ในการสื่อสารกันแต่อย่างใด โดย +VCC หรือ +5V นี้ จะออกแบบเผื่อไว้ในกรณีที่อุปกรณ์ปลายทางเป็นวงจรขนาดเล็กและไม่สะดวกที่จะหาแหล่งจ่ายไฟให้กับอุปกรณ์ปลายทางด้วย ก็อาจต่อไฟเลี้ยงวงจร +VCC นี้ออกไปให้กับอุปกรณ์ปลายทางด้วยก็ได้เช่นกัน

2.6 การแปลงสัญญาณอนาลอก-ดิจิตอล (Analog to Digital Converter)

สัญญาณที่ใช้ในอุปกรณ์อิเลคทรอนิกส์ มี 2 ชนิดคือ สัญญาณอนาลอกและสัญญาณดิจิตอล สัญญาณอนาลอกจะใช้ในอุปกรณ์ทั่วๆไปและใช้ในการควบคุมแบบเก่า

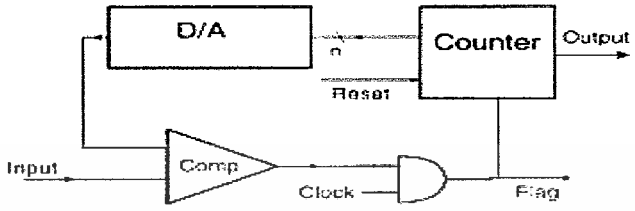
ในปัจจุบันมีไมโครโปรเซสเซอร์และไมโครคอนโทรลเลอร์ เข้ามาช่วยในการควบคุมอุปกรณ์ต่างๆ มากมายซึ่งทำให้การควบคุมนั้นทำได้ง่ายและรวดเร็วยิ่งขึ้นแต่ในการควบคุมนั้นเราจำเป็นต้องใช้สัญญาณดิจิตอลในการติดต่อกับไมโครโปรเซสเซอร์ หรือไมโครคอนโทรลเลอร์แต่ในความเป็นจริงนั้นเราใช้สัญญาณอนาลอกในการควบคุมดังนั้นเราจึงจำเป็นต้องมีการเปลี่ยนสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแล้วจึงนำสัญญาณนั้นเข้ามาสู่ไมโครโปรเซสเซอร์หรือไมโครคอนโทรลเลอร์เพื่อใช้ควบคุมระบบต่อไป

แม้ว่าสัญญาณอนาลอกนั้นมีความแน่นอนและแม่นยำสูง แต่สัญญาณอนาลอกนั้นก็ควบคุมได้ยาก เนื่อง จากในสภาพแวดล้อม มีสัญญาณรบกวนอยู่มาก และการที่จะทำให้การควบคุมแบบอนาลอกมีความสามารถควบคุม เท่ากับการควบคุมแบบดิจิตอลนั้น ทำได้ยาก เนื่องจากวงจรควบคุมแบบ อนาลอกจะต้องมีความซับซ้อนสูง อย่างไรก็ตาม สัญญาณดิจิตอลก็ไม่สามารถทดแทนความละเอียดของสัญญาณอนาลอกได้อย่างสมบูรณ์ แต่ทำให้การควบคุมนั้นทำให้ง่าย และสะดวกยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Counting Converter

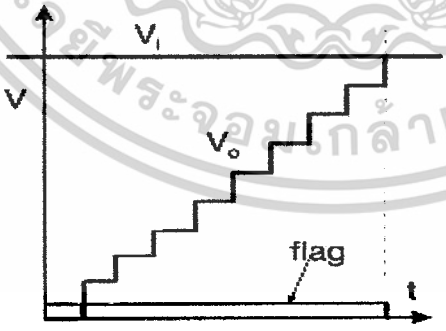
Counting Converter เป็นวิธีที่ง่ายที่สุดของการแปลงสัญญาณอนาลอก เป็นสัญญาณดิจิตอล โดยใช้อัลกอริทึม การนับค่าเพิ่มขึ้นเรื่อยๆ แล้วนำผลที่ได้จากการนับไปเปรียบเทียบกับค่าที่ต้องการที่ตั้งไว้ ลักษณะการทำงานเป็นดังรูป



รูปที่ 2.38 การทำงานของ Counting Converter

จากวงจร Counter เป็นอุปกรณ์นับค่าที่เพิ่มขึ้นทีละหนึ่ง แล้วส่งค่าที่ได้ให้ D/A มีขา Reset รับสัญญาณ Reset เมื่อต้องการให้เริ่มนับใหม่ D/A เมื่อรับค่าที่นับเพิ่มขึ้นทีละหนึ่งจากตัวนับก็แปลงค่าให้เป็นสัญญาณอนาลอกที่มีค่าความต่างศักย์ค่าๆ หนึ่งแล้วจึงส่งต่อเข้าไปที่อุปกรณ์ตัวเปรียบเทียบ (Comparator)

Comparator จะเป็นอุปกรณ์ตัวเปรียบเทียบค่าความต่างศักย์ ของอินพุต และค่าจากที่ตัวนับ ถ้าหากทั้งสองสัญญาณมีค่าเท่ากันส่งค่าความต่างศักย์ 0 โวลต์ออกมา (ลอจิก 0) ถ้าไม่เท่ากันก็จะส่งความต่างศักย์ที่ไม่ใช่ 0 โวลต์ออกมา (ลอจิก 1) ซึ่งค่าความต่างศักย์ที่ออกมา จะนำมาเข้าลอจิกเกต "และ" กับสัญญาณนาฬิกา จะได้ค่าลอจิกออกมา ถ้าผลลัพธ์ออกมาเป็นสัญญาณนาฬิกาแสดงว่ายังไม่ได้ผลลัพธ์เท่าที่ต้องการ สัญญาณนาฬิกาก็จะไปทำให้ตัวนับนับเพิ่มขึ้นต่อไป และเมื่อได้ค่าผลลัพธ์ดิจิตอลที่ต้องการแล้ว ค่าที่ได้จาก ตัวเปรียบเทียบจะให้ค่าความต่างศักย์เป็น 0 (ลอจิก 0) ซึ่งเมื่อนำมาเข้าลอจิกเกต "และ" กับสัญญาณนาฬิกาแล้ว ก็จะให้ลอจิก 0 ซึ่งทำให้ตัวนับไม่นับเพิ่มอีก ก็จะได้ค่าดิจิตอลจากตัวนับที่ต้องการ จากคำอธิบายข้างต้นจะได้กราฟของ V0 ดังนี้



รูปที่ 2.39 Output voltage ของ A/D converter

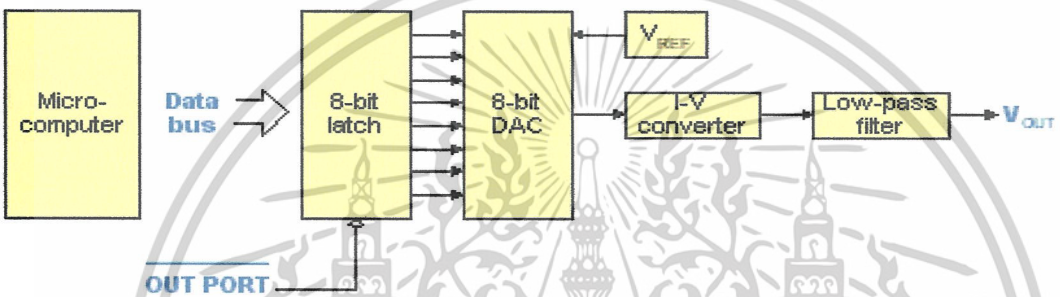
ข้อเสียของวิธีนี้ คือ การนับต้องเริ่มนับที่ 0 เสมอ และนับเพิ่มขึ้นเรื่อยๆ ทำให้ช้า เอาท์พุตที่ได้จะมี delay จึงไม่ค่อยนิยมใช้เท่าที่ควร จึงได้เปลี่ยนตัวนับเป็นแบบนับลงได้ด้วย ซึ่งจะอ้างอิงระดับจากระดับเก่า ทำให้ไม่จำเป็นต้องนับ 0 ใหม่ เมื่อมีการเปลี่ยนอินพุตใหม่ แต่ให้อ้างอิงกับผลลัพธ์เดิม ทำให้ได้ผลลัพธ์เร็วขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การแปลงสัญญาณทางดิจิตอลเป็นอนาลอก (Digital to Analog Converter)

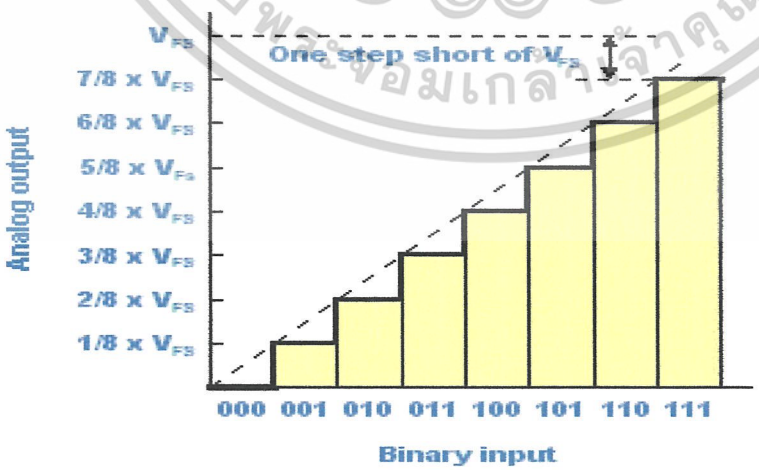
อุปกรณ์ทางไฟฟ้า/อิเล็กทรอนิกส์ โดยทั่วไปที่เป็นอนาลอก สามารถควบคุมการทำงานโดยการให้อินพุตเป็นระดับ แรงดันที่แตกต่างกัน ตัวอย่างเช่น มอเตอร์กระแส ตรง ซึ่งควบคุมความเร็วโดยเปลี่ยนระดับแรง ดัน (หรือกระแส) ของขดลวดสนาม เมื่อนำระบบดิจิตอล หรือไมโครคอนโทรลเลอร์ มาใช้ควบคุม อุปกรณ์ทางอนาลอกเหล่านี้ จึงต้องมีวงจรซึ่ง สามารถแปลงสัญญาณทางดิจิตอลเป็นระดับแรงดันต่อเนื่อง แบบอนาลอกตั้งแต่ศูนย์โวลต์จนถึงระดับสูงสุดที่ กำหนดไว้ เรียกว่าวงจร Digital to Analog Converter (DAC)

คุณสมบัติของ Digital to Analog Converter



รูปที่ 2.40 ระบบการแปลงสัญญาณดิจิตอลเป็นอนาลอก

รูปที่ด้านบนแสดงถึงส่วนประกอบหลักของระบบ DAC โดยทั่วไป ไมโครคอมพิวเตอร์จะมีเอาต์พุตเป็นค่าไบนารี วงจรแลทช์รับค่าไบนารีเข้ามาเพื่อส่งไปยัง DAC ในวงจรจะใช้แหล่งกำเนิดแรงดันหรือ กระแสคงที่เพื่ออ้างอิงในการแปลงข้อมูลไบนารีเป็นระดับกระแส ต่อมาจะมีวงจรแปลงจากกระแสเป็นระดับแรงดัน (current-to-voltage converter) ซึ่งปกติจะใช้ออปแอมป์ท้ายสุด สัญญาณอนาลอกที่ได้จะผ่านวงจร low-pass filter เพื่อกำจัดสัญญาณความถี่สูงที่แฝงอยู่ในสัญญาณที่ถูกสร้างขึ้นมา



รูปที่ 2.41 Transfer Curve ในอุดมคติของ DAC 3 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

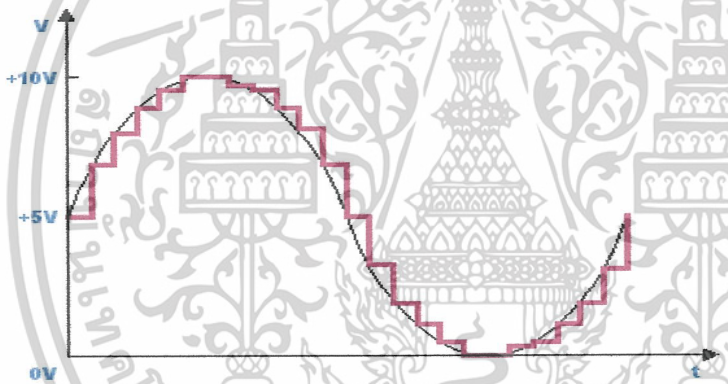
รูปที่ด้านบนเป็นกราฟแสดงถึงความสัมพันธ์ระหว่าง เอาต์พุตที่เป็นอนาลอกกับอินพุตที่เป็นดิจิตอลขนาด 3 บิตเรียกว่า transfer curve สังเกตว่าเมื่ออินพุตไบนารีเพิ่มขึ้น เอาต์พุตอนาลอกจะเพิ่ม ในลักษณะขั้นบันได ขนาดของแต่ละขั้นจะ หาได้จาก

$$\text{stepsize} = V_{FS}/2^n$$

เมื่อให้ V_{FS} คือระดับแรงดันเอาต์พุตสูงสุด

n คือจำนวนบิตของอินพุต

เนื่องจากเอาต์พุตของ DAC จะเพิ่มเป็นขั้นๆ รูปคลื่นสัญญาณ ที่ได้จาก DAC จึงมีลักษณะไม่เรียบดังตัวอย่าง ในรูปที่ด้านล่างซึ่งแสดงถึงสัญญาณไซน์ ที่สร้างจาก DAC



รูปที่ 2.42 คลื่นไซน์ที่สร้างจาก DAC

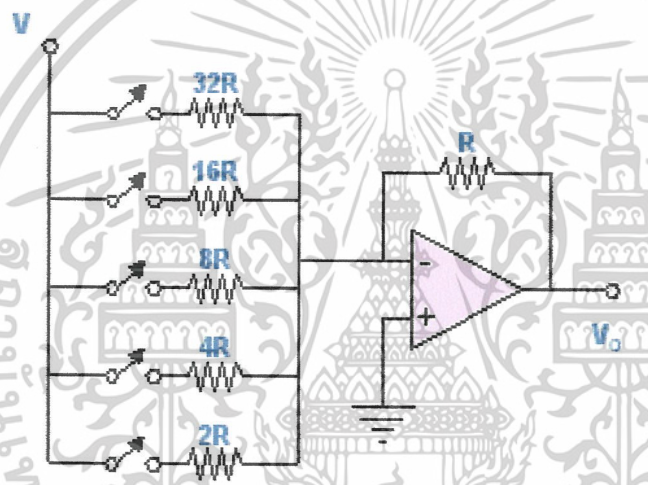
ถ้าเพิ่มจำนวนบิต ความละเอียดของ DAC จะเพิ่มขึ้น เช่น เมื่อ ใช้ DAC 12 บิต และ V_{FS} = 5.0 V ความละเอียดคือ 5.0 V / 4096 = 1.22 mV ซึ่งจะ ละเอียดกว่า DAC 8 บิตถึง 16 เท่า ความถูกต้องของ DAC ขึ้นอยู่กับหลายส่วน

- **Quantization error** DAC บิต VFS = 5.0 V เอาต์พุตจะมีความละเอียด 19.53 mV ถ้าต้องการเอาต์พุต 4.00 V DAC จะให้เอาต์พุตได้ใกล้เคียง ที่สุดคือ 4.04 V (19.53 mV x 205) ผิดพลาด 4 mV โดยทั่วไปค่าผิด พลาดจะเท่ากับ +/- 0.5 LSB (least significant bit) ตัวอย่างเช่น DAC 8 บิต ความผิดพลาดจะเป็น 1 ใน 512 หรือ +/- 0.195 %
- **Offset and gain errors** เมื่ออินพุตไบนารีเท่ากับ 0 แต่เอาต์พุตของ DAC ไม่เป็น 0 เรียก ว่า offset error และอาจเกิดร่วมกับ gain error ความผิดพลาดเหล่า นี้จะทำให้ transfer curve โค้งขึ้น หรือลง ขึ้นอยู่กับความไม่สมดุลภายใน DAC อย่างไรก็ ตาม offset error และ gain error จะแก้ได้โดยใช้ความต้าน ทานปรับค่าได้ต่อไว้ภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Nonlinearity** คือค่าความคลาดเคลื่อนสูงสุดของ transfer curve เทียบกับเส้นตรงจากจุดศูนย์และจุดสูงสุด ซึ่งจะขึ้นอยู่กับความผิดพลาดของส่วนประกอบภาย ใน DAC ใน data sheet ของ DAC จะระบุเป็นเปอร์เซ็นต์เทียบกับค่าสูงสุด หรือ ระบุเป็นเศษส่วนของ LSB (โดยทั่วไปคือ +/- 0.5 LSB)
- **Settling time** คือช่วงเวลานับแต่ให้อินพุตจนกระทั่ง DAC ให้ เอาต์พุต วัดเมื่อเอาต์พุตที่ได้ผิดพลาดจากค่าจริง น้อยกว่า 0.5 LSB ค่าเวลานี้อาจน้อยกว่า 100 ns สำหรับ DAC ความ เร็วสูง และอาจมากกว่า 100 us สำหรับ DAC ราคาถูก

Summed Source DAC

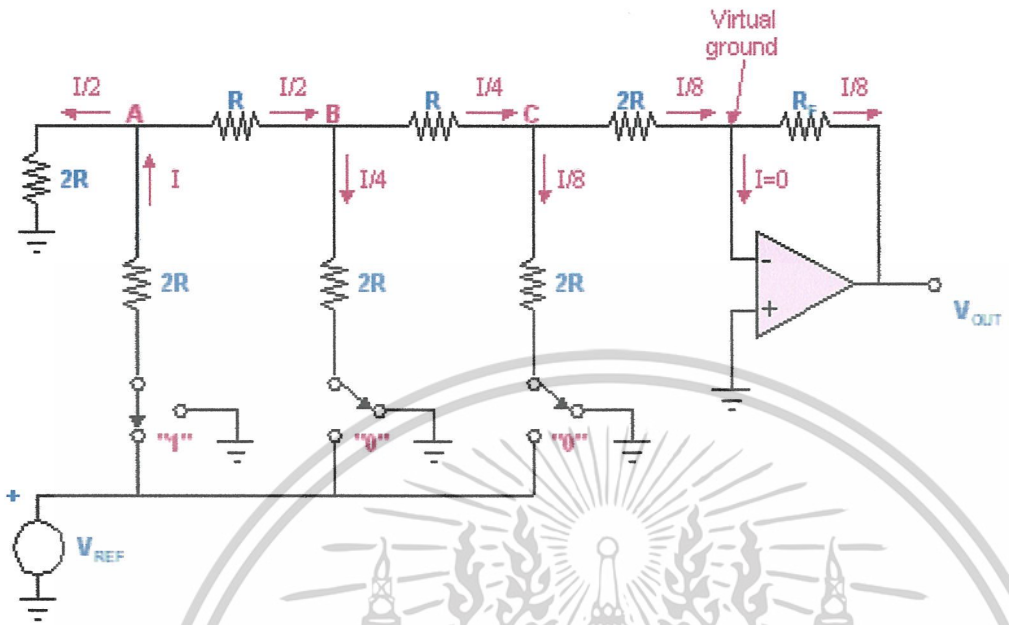


รูปที่ 2.43 Summed Source DAC

เป็นวงจรอย่างง่ายในการแปลงสัญญาณดิจิตอลเป็นอนาลอก ซึ่ง เป็นวง จร Summing Amp มีความต้านทานค่า 2R, 4R และ 8R เพื่อทำให้ กระแสที่ผ่านความต้านทานแต่ละตัวมีค่า ลดลงเป็น 2 เท่า ความต้านทานตัวล่างสุด (2R) จะ เป็น MSB ส่วนตัวบนสุดจะเป็น LSB ข้อเสียของการใช้วงจรลักษณะนี้ ในทางปฏิบัติค่าความต้านทานที่ต่างกันเป็น 2 เท่า คือ 2R, 4R, 8R, จะ ไม่สามารถหาได้ง่าย จึงมีการปรับปรุงเป็น วงจร R-2R

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Switched Voltage R-2R DAC

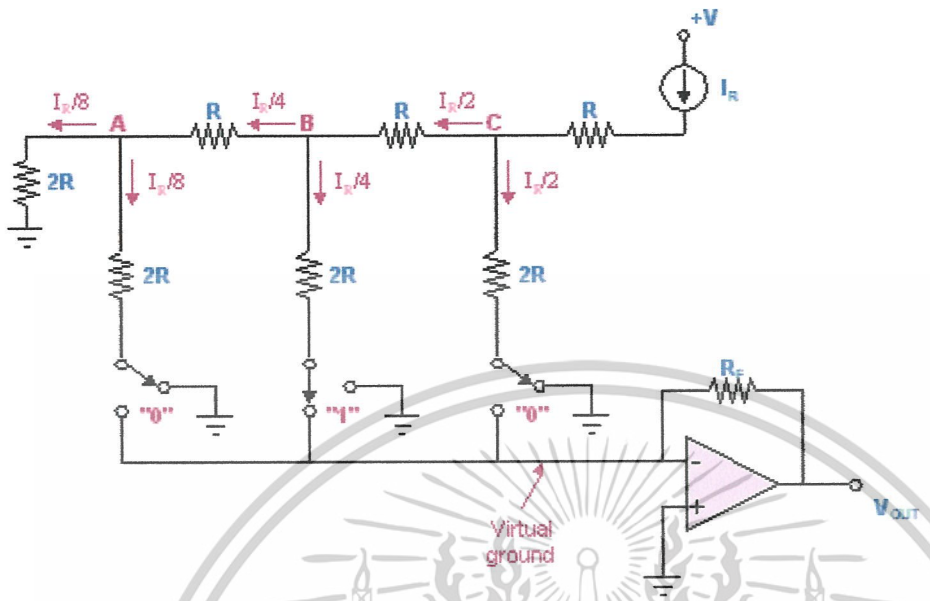


รูปที่ 2.44 Switched Voltage R-2R DAC

รูปที่ด้านบน เป็น DAC 3 บิต ใช้ออปแอมป์และความต้านทาน เพียง 2 ค่าคือ R และ 2R สังเกตว่าอินพุตดิจิตอลจะมาจากสวิตช์ทั้ง 3 ซึ่ง อาจต่อกับกราวด์ (ลอจิก 0) หรือต่อกับ V_{REF} (ลอจิก 1) ตัวอย่าง นี้อินพุตเป็น 001 พิจารณากระแส I เมื่อผ่านจุด A จะถูกแบ่งเป็นสองส่วน เท่าๆ กัน เหลือ I/2 เมื่อผ่านจุด B และ C จะถูกแบ่งอีกครั้ง เหลือ I/4 และ I/8 ตามลำดับ ดังนั้นกระแสที่ป้อนให้กับออปแอมป์จะ เหลือ I/8 เมื่อพิจารณาที่สวิตช์ตัวอื่นๆ ก็จะมีลักษณะคล้ายกัน ดังนั้นกระแสที่ผ่านออปแอมป์เมื่อปิดสวิตช์อื่นนับจากซ้าย มาขวา จะมีขนาด I/8, I/4 และ I/2 ตามลำดับ สวิตช์ซ้ายสุด จะเป็น LSB ส่วนขวาสุดจะเป็น MSB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Switched Current R-2R DAC



รูปที่ 2.45 Switched Current R-2R DAC

วงจรนี้เปลี่ยนจากการใช้แรงดันอ้างอิง (V_{REF}) มาเป็นกระแสอ้างอิง (I_R) กระแสที่ ผ่านสวิตช์แต่ละตัวจากขวามาซ้ายจะเป็น $I_R/2$, $I_R/4$ และ $I_R/8$ ตามลำดับวงจรลักษณะนี้จะมีความเร็วสูงกว่าวงจร Switched Voltage เนื่องจาก คาปาซิแตนซ์ที่รอยต่อ (junction capacitance) ของความต้าน ทานแต่ละตัวจะไม่ถูกชาร์จและดิสชาร์จเหมือนวง จร Switched Voltage ในการแปลงสัญญาณดิจิตอลเป็นอนาลอกยังมีวิธีอื่นอีกซึ่งเรายกตัวอย่างพียงเท่านี้

2.8 การรับส่งข้อมูลแบบอนุกรม

รับส่งข้อมูลแบบอนุกรม ซึ่งมีขอดีคือ ใช้สายสัญญาณน้อยและส่งได้เป็นระยะทางไกล สำหรับโครงงานนี้ใช้การติดต่อกับไมโครคอนโทรลเลอร์ รับส่งข้อมูลแบบอนุกรรมกับคอมพิวเตอร์ เพื่อทำการเปลี่ยนระดับสัญญาณไฟฟ้าได้มาตรฐานในการรับส่งข้อมูลผ่านทางพอร์ตอนุกรรม (Serial Port) ของเครื่องไมโครคอมพิวเตอร์ ซึ่งสัญญาณข่าวสารที่ได้อย่างถูกประมวลโดยโปรแกรม Visual Basic เพื่อแสดงผลออกทางหน้าจอคอมพิวเตอร์

ความรู้เบื้องต้นเกี่ยวกับมาตรฐาน RS-232

โดยปกติเรื่องไมโครคอมพิวเตอร์จะมีพอร์ตที่เป็นอนุกรรมชื่อว่า RS-232 อยู่ในตัวมันเองอยู่แล้ว ซึ่งพอร์ต RS-232 นี้ทำหน้าที่รับและส่งข้อมูลในแบบอนุกรมเรียกว่า Universal Asynchronous Adapter เหตุที่มีชื่อเรียกว่า RS-232 ก็เนื่องมาจาก สมาคมผู้ผลิตอุปกรณ์อีเล็กทรอนิกส์ ของอเมริกา หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EIA (RS-232: Recommended Standard Number 232,EIA: Electronic Industry Association) ได้กำหนดมาตรฐานของอุปกรณ์สื่อสารแบบอนุกรมเอาไว้ภายใต้ชื่อว่า RS-232 (ความจริงแล้วมาตรฐานของรับส่งข้อมูลแบบอนุกรมมีหลายมาตรฐาน แต่ที่นิยมกันมากที่สุดสำหรับเครื่องไมโครคอมพิวเตอร์)

หน้าที่สำคัญของการสื่อสารแบบอะซิงโครนัส (Asyncrous)

รับสัญญาณ

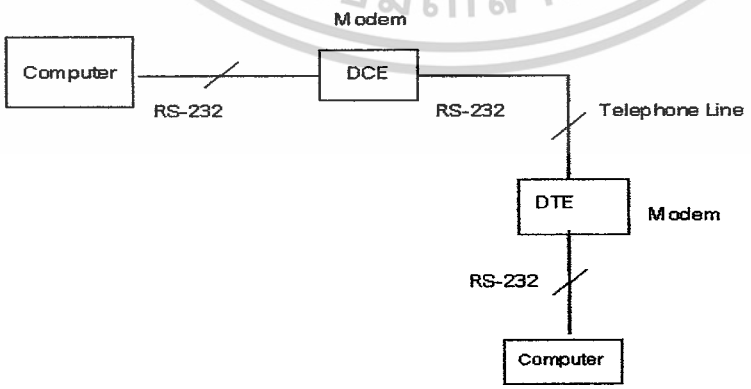
- เปลี่ยนสัญญาณ Input ที่เข้ามาแบบอนุกรมให้เป็นแบบขนาน
- ตรวจสอบความผิดพลาดของสัญญาณที่รับ
- ตัดสต็อปบิต (Stop Bit) และพาริตี้บิต (Parity Bit)
- สัญญาณให้ CPU รับรู้ว่า ได้รับสัญญาณไว้แล้ว

ส่งสัญญาณ

- เปลี่ยนสัญญาณแบบขนาดจาก CPU ค่อยๆ ทยอยส่งออกเป็นแบบอนุกรม
- เพื่อสต็อปบิตและพาริตี้บิต
- เพิ่มสัญญาณควบคุมโมเด็มที่เชื่อมต่อ (ถ้ามี)

มาตรฐาน RS-232C

มาตรฐาน RS-232C ได้จัดพิมพ์ขึ้นเมื่อ ค.ศ. 1969 RS ย่อมาจาก Recommended Standard ส่วน 232 คือหมายเลขบ่งบอกมาตรฐานตัวนี้ และ C เป็นหมายเลขฉบับสุดท้ายของมาตรฐานตัวนี้ จุดประสงค์ของมาตรฐาน RS-232 ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งข้อมูล (DCE : Data Communication Equipment) กับอุปกรณ์สื่อสารข้อมูล (DCE : Data Communication Equipment) สำหรับผู้ใช้คอมพิวเตอร์ทั่วไป DTE ก็หมายถึง ตัวไมโครคอมพิวเตอร์ ส่วน DCE หมายถึง โมเด็ม (modem) และอุปกรณ์อื่นๆ เช่นเครื่องพิมพ์ที่รับสัญญาณแบบอนุกรม อาจจะเป็นไปได้ทั้ง DTE และ DCE ซึ่งจะขึ้นอยู่กับผู้ผลิตสำหรับข้อแตกต่างของ DTE และ DCE จะเห็นได้จากรูปที่ 3.9 ซึ่งจากรูปจะเห็นได้ว่า RS-232C มีส่วนสำคัญอย่างมากสำคัญอย่างมากสำหรับการสื่อสารข้อมูลระหว่างไมโครคอมพิวเตอร์



รูปที่ 2.46 การใช้มาตรฐาน RS-232 เชื่อมต่ออุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา	ฟังก์ชั่น
1	Received Line Signal Detect
2	Received Data
3	Transmit Data
4	Data Terminal Ready
5	Signal Ground
6	Data Set Ready
7	Request to Send
8	Clear to Send
9	Ring Indicator

ตารางที่ 2.7 ชื่อขาสัญญาณของ DB-9

คำอธิบายการทำงานของแต่ละขาที่ใช้งานของ DB-9

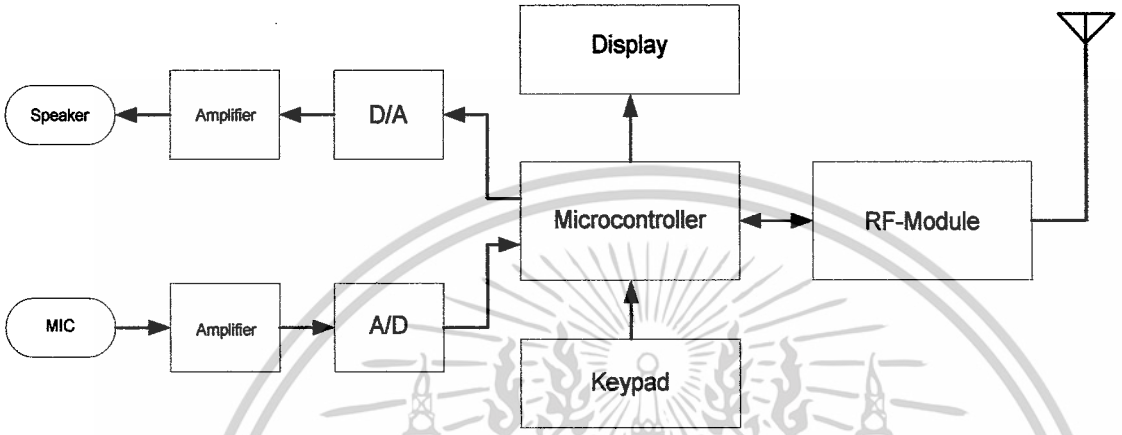
- Transmit Data (TD) เป็นสัญญาณที่ส่งออกจาก DTE (หรือตัวไมโครคอมพิวเตอร์) ไปยังโมเด็มหรือต่อกับไมโครคอมพิวเตอร์ตัวอื่น เมื่อไม่มีสัญญาณส่งออกสถานะภาพของลอกจิกที่ขานี้มีค่าเท่ากับ"1" หรือเทียบเท่า Stop Bit
- Received Data (RD) เป็นทางสัญญาณเข้าไปยัง DTE หรือไมโครคอมพิวเตอร์ เมื่อไม่มีสัญญาณรับเข้ามา ขานี้จะมีสถานะภาพลอจิกเป็น "1"
- Request to Send (RTS) ใช้สำหรับส่งสัญญาณไปยังโมเด็มหรือเครื่องพิมพ์เป็นการเรียกร้องที่จะส่งสัญญาณมาทาง TD สัญญาณนี้จะใช้คู่กับ CTS ที่อุปกรณ์รับ หากได้รับ สัญญาณ RTS จะตรวจตัวเองว่าพร้อมจะรับสัญญาณได้หรือยัง หากพร้อมก็จะส่งสัญญาณออกไปที่สาย CTS
- Clear to Send (CTS) เมื่อสายสัญญาณนี้อยู่ในสภาวะปิด (Negative Voltage หรือลอจิก"1") หมายความว่า อุปกรณ์รับกำลังบอกว่า พร้อมจะรับข้อมูลแล้ว
- Data Set Ready (DSR) เมื่อสายสัญญาณนี้อยู่ในสภาวะเปิด (ลอกจิก"0") จะเป็นการบอกไมโครคอมพิวเตอร์ว่า พร้อมที่จะส่งได้แล้ว
- Signal Ground (SG) ทำหน้าที่เป็นระดับแรงดันอ้างอิงสำหรับทุกๆสายสัญญาณจะมีแรงดันเป็น "0" เมื่อเทียบกับสายสัญญาณอื่นๆ
- Data Terminal Ready (DTR) คอมพิวเตอร์เปิดสัญญาณนี้ให้เปิด (ลอจิก"0") เมื่อพร้อมที่จะติดต่อรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

3.1 การออกแบบและสร้างชุดกระจายเสียงแบบดิจิตอลในส่วนของฮาร์ดแวร์



รูปที่ 3.1 บล็อกไดอะแกรมของชุดกระจายเสียงแบบดิจิตอล

การอธิบายบล็อกไดอะแกรมจแบ่งการอธิบายออกเป็นสองส่วนนั้นคือ ขณะเป็นตัวส่ง และ ขณะเป็นตัวรับ ซึ่งสามารถอธบายได้ดังนี้

3.1.1 อธิบายชุดรับ – ชุดส่ง

3.1.1.1 ขณะชุดกระจายเสียงเป็นชุดส่ง

จากบล็อกจะเริ่มจากเมื่อเราป้อนอินพุทที่เป็นสัญญาณเสียงเข้าไปยังช่องรับอินพุทสัญญาณเสียงที่รับเข้ามาก็จะถูกขยายสัญญาณให้มีความแรงขึ้นโดยภาค Amplifier เพื่อส่งไปยังภาคADC ซึ่งในส่วนของภาคนี้จะทำการ Sampling สัญญาณเสียงอินพุทออกมาแล้วแปลงสัญญาณเสียงนั้นเป็นสัญญาดิจิตอลเพื่อส่งไปให้กับ AVR ซึ่งเป็น Microcontroller ที่เลือกใช้ในส่วนของ AVR จะเป็นตัวกำหนดรูปแบบของเฟรมข้อมูลในการส่งและควบคุมพารามิเตอร์ต่างๆทั้งหมด เช่น อัตราเรวในการส่ง การตรวจสอบพาริตี้ ช่องสัญญาณ กำหนด Coordinate หรือ Endevice ของเครือข่าย เป็นต้น ซึ่งเมื่อกำหนดค่าต่างๆแล้วในส่วนของ AVR ก็จะส่งข้อมูลต่อไปที่อาร์เอฟโมดูลโดยการใช้ Serial Port ในส่วนของอาเอฟโมดูลก็จะกระจายเสียงออกไปตามคุณสมบัติที่ได้กล่าวไว้แล้วในบทที่ 2

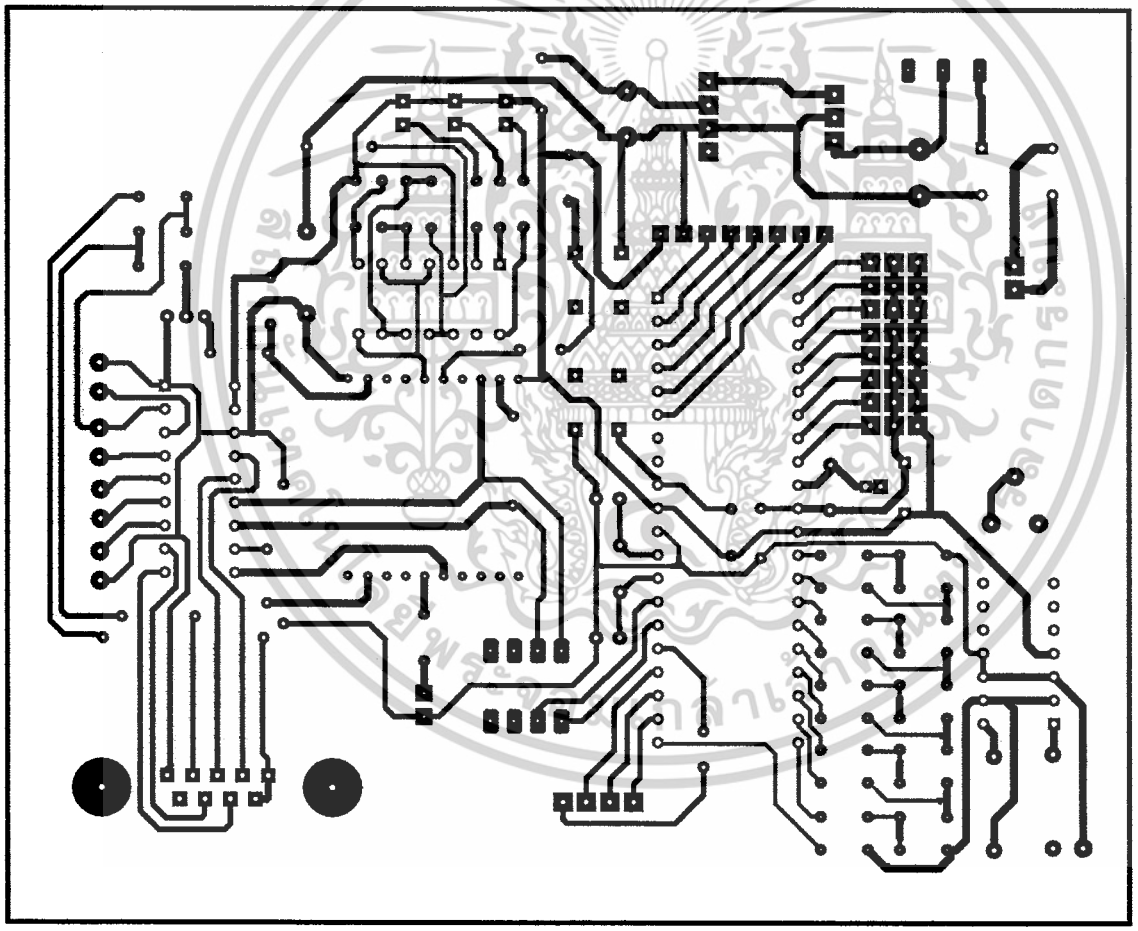
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.2 ขณะชุดกระจายเสียงเป็นชุดรับ

เริ่มจากเมื่อรับสัญญาณข้อมูลเสียงที่เป็นดิจิตอลเข้ามาทางอาร์เอฟโมดูลสัญญาณจะถูกส่งไปยัง AVR จากนั้นก็จะส่งสัญญาณที่ได้ไปสู่ภาค DAC ซึ่งภาคนี้จะเป็นการแปลงสัญญาณเสียงที่เป็นข้อมูลดิจิตอลอยู่ให้เปลี่ยนเป็นสัญญาณในรูปแบบแอนาลอกจากนั้นก็นำสัญญาณเสียงที่ผ่านการแปลงแล้วไปขยายด้วย Amplifier เพื่อส่งต่อออกลำโพง

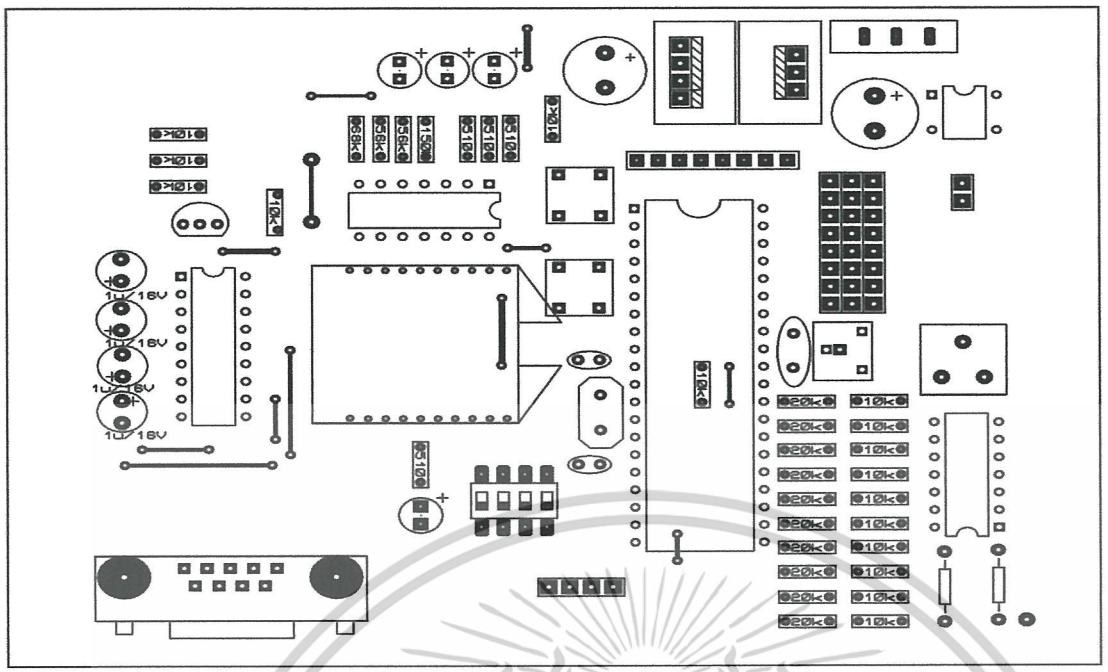
3.1.2 การออกแบบและสร้างบอร์ดควบคุม

ในการออกแบบบอร์ดควบคุมจะมีส่วนของการทำงานอยู่สองส่วนคือส่วนของการควบคุมโมดูล Zigbee และส่วนของการควบคุม AVR โดยที่ภายในบนบอร์ดจะประกอบไปด้วยวงจรภาคจ่ายไฟ วงจรบริดจ์เพื่อกันการป้อนแหล่งจ่ายผิด ส่วนของจอแสดงผล และพอร์ตเชื่อมต่อคอมพิวเตอร์ ซึ่งสามารแสดงส่วนต่างๆได้ดังรูป

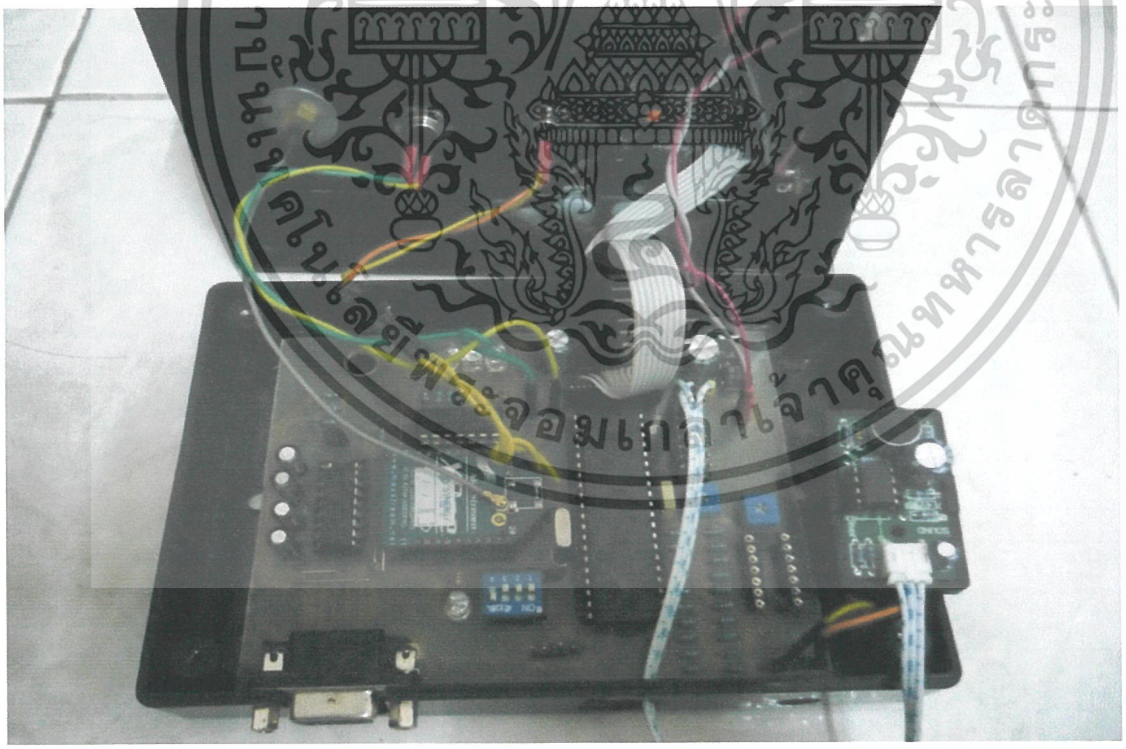


รูปที่ 3.2 ลายวงจรของบอร์ดควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 การลงอุปกรณ์



รูปที่ 3.4 ภายในของชุดกระจายเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.5 ภายนอกของชุดกระจายเสียง

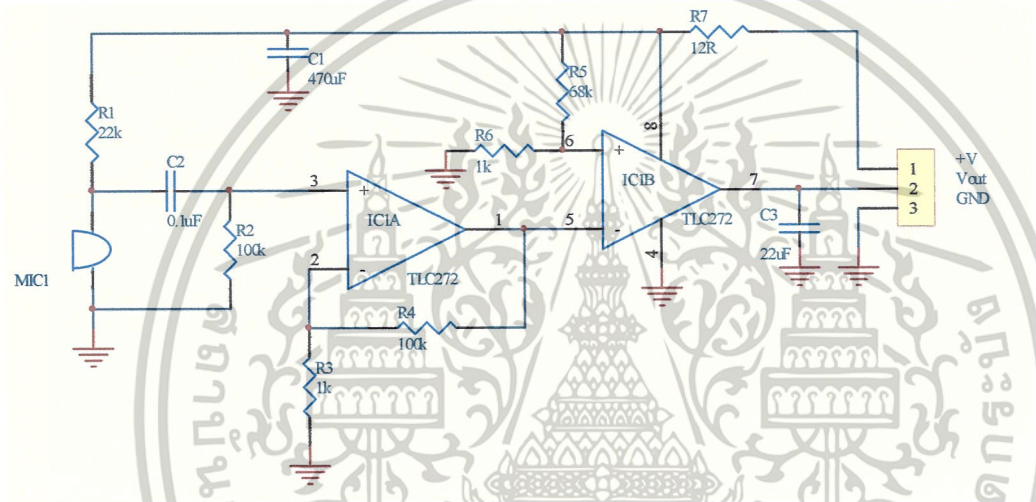
รูปที่ 3.6 ชุดวงจรที่สมบูรณ์ขณะทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

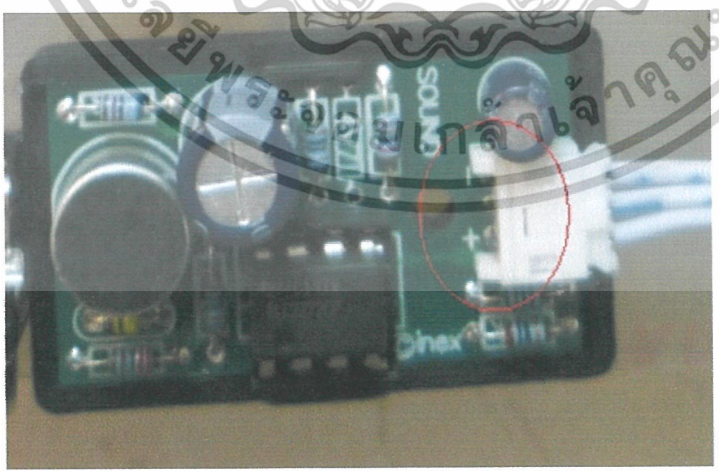
3.13 วงจรปรีไมค์ (Sounder detector)

อุปกรณ์หลักที่ทำหน้าที่เปลี่ยนพลังงานเสียงเป็นสัญญาณไฟฟ้าคือ ไมโครโฟน ในวงจรนี้จะใช้ตัวเก็บประจุ หรือเรียกว่า คอนเด็นเซอร์ไมโครโฟนเมื่อ MIC1 ได้รับสัญญาณเสียงจะมีแรงดันออกไปเข้าที่ขา 3 ของไอซี IC1/1 ซึ่งเป็นไอซีออปแอมป์ทำหน้าที่ขยายสัญญาณเสียงให้มีระดับสัญญาณที่เพิ่มขึ้นและถูกขยายอีกชั้นหนึ่งด้วย IC1/2 ค่าแรงดันเอาต์พุทโวล์เตจจะอยู่ระหว่าง 0 – 5 v ซึ่งขึ้นอยู่กับความดังของเสียงที่พูดเข้าไป

ไมโครคอนโทรลเลอร์สามารถอ่านข้อมูลเสียงนี้เข้าไปได้ก็ต่อเมื่อนำแรงดันเอาต์พุทของวงจตรวจจับสัญญาณเสียงนี้ไปแปลงเป็นสัญญาณดิจิตอล



รูปที่ 3.7 วงจรปรีไมค์



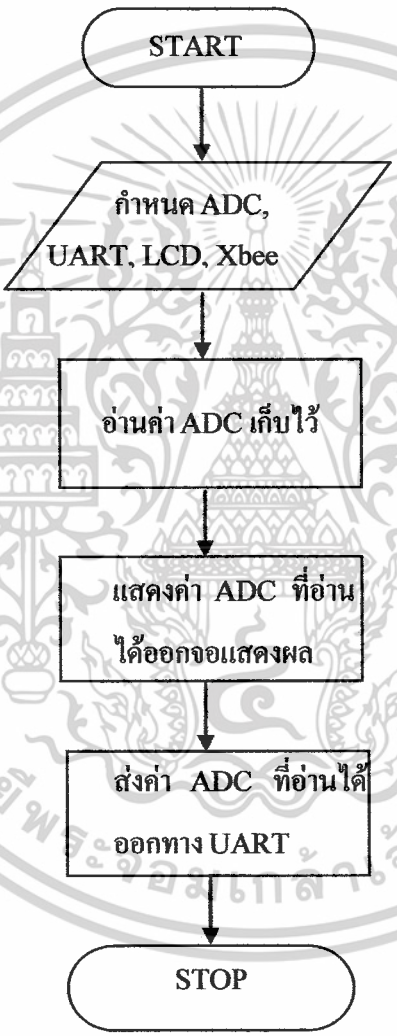
รูปที่ 3.8 ชุดปรีไมค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบและการเขียนโปรแกรมในส่วนของ ซอฟแวร์

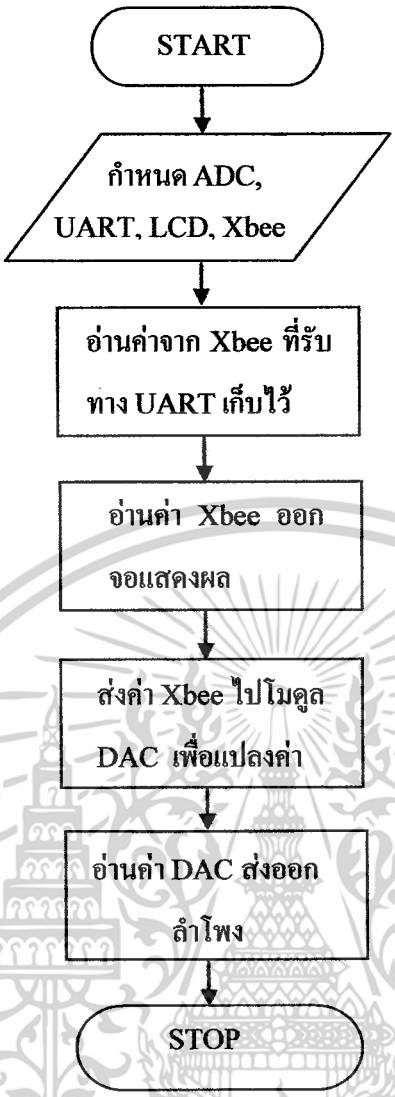
การเขียนโปรแกรมเพื่อควบคุมการทำงานของไมโครคอนโทรลเลอร์เขียนด้วยภาษาซีเพื่อ

- กำหนดสภาวะการทำงานของการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล
- กำหนดสภาวะการทำงานให้ไมโครคอนโทรลเลอร์สามารถส่งข้อมูลแบบอนุกรมไปให้โมดูลเพื่อทำการส่งข้อมูลออกไป
- กำหนดสภาวะการทำงานของการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก
- กำหนดสภาวะการทำงานเพื่อเรียกค่าพารามิเตอร์ต่างของโมดูลแสดงออกหน้าจอ



รูปที่ 3.9 ผังงานแสดงการเขียนโปรแกรมควบคุมทางภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 ผังงานแสดงการเขียนโปรแกรมควบคุมทางภาครับ

จากรูปแบบแผนผังแสดงการเขียนโปรแกรมทั้งภาครับและภาคส่งด้านบนนั้นจะเห็นว่าในส่วนของภาครับเราจะเริ่มจากการรับค่าอินพุทเข้าไปแปลงขอมูลที่โมดูล ADC และก็อ่านค่า ADC แสดงออกจอแสดงผลจากนั้นก็ส่งค่า ADC ออกทาง UART ไปให้ Zigbee ต่อไป และในส่วนของภาครับก็จะรับข้อมูลมาเก็บไว้ในตัวแปร Xbee จากนั้นก็อ่านค่า Xbee ออกจอแสดงผล จากนั้นก็ส่งค่า Xbee ไปเข้าโมดูล DAC แล้วอ่านค่า DAC ออกสู่ลำโพง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

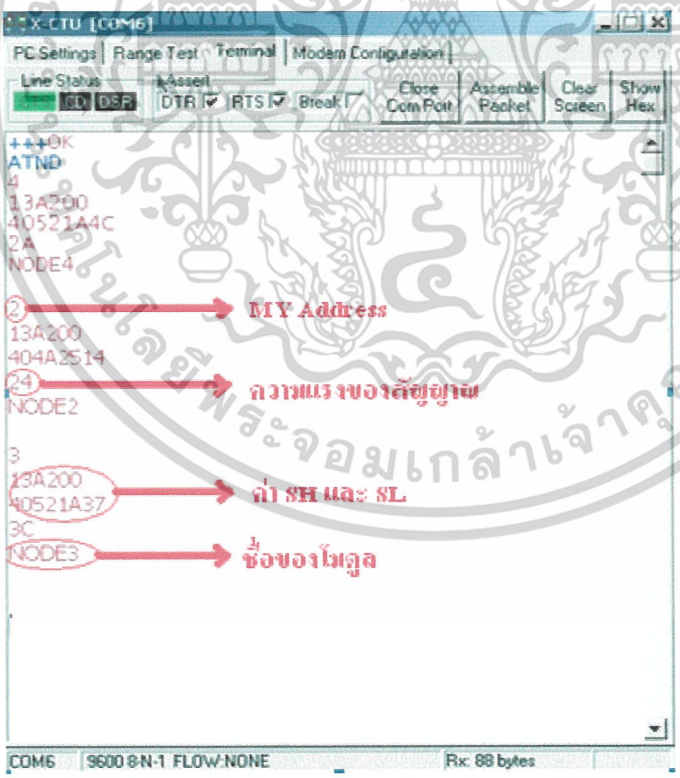
บทที่ 4

การทดลองและผลการทดลอง

เมื่อทำการประกอบวงจรในส่วนของวงจรควบคุมโมดูลและวงจรในส่วนของไมโครคอนโทรเลอร์เสร็จเรียบร้อยแล้วคณะผู้จัดทำก็ได้ทำการทดสอบวงจรดังกล่าวโดยในส่วนแรกทำการทดสอบในส่วนของการติดต่อกันระหว่างโมดูลกับโมดูลโดยมีคอมพิวเตอร์เป็นอุปกรณ์ในการแสดงผลและส่งข้อมูลโดยใช้โปรแกรม X-CTU เป็นโปรแกรมที่ใช้ในการรับ-ส่ง

4.1 การวัดระดับความแรงของสัญญาณ

การวัดความแรงของสัญญาณสามารถนั้นสามารถวัดได้โดยใช้คำสั่ง ATND ในการแสดงค่าความแรงของสัญญาณในระยะต่างๆ โดยใช้โมดูล zigbee ในการรับส่งสองตัวโดยที่มี โมดูล zigbee ที่ตั้งค่าเป็น Coordinate อยู่ตรงกลางโดยที่ระยะทางตั้งแต่ต้นทางไปยังปลายทางทั้งสิ้น 200 เมตร โดยที่จะเพิ่มขึ้นทีละ 20 เมตรซึ่งจะทำการวัดเป็นจำนวน 10 ครั้งเพื่อนำมาหาค่าเฉลี่ยความแรงของสัญญาณดังตารางด้านล่างนี้



รูปที่ 4.1 หน้าต่างที่ใช้วัดความแรงของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	62	5C	5D	5F	54	56	52	4C	3E	3C		3F	3C	4C	45	5A	50	52	62	64	6C

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	65	67	65	5C	52	56	54	47	41	3E		3C	49	4D	5A	58	54	65	62	67	6F

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	6C	61	62	5F	53	54	51	47	41	3D		3C	48	4D	59	5F	54	5F	62	64	62

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	5F	62	5F	5F	54	56	52	49	40	44		3F	4A	4C	48	5C	56	5F	5F	62	62

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	5F	62	5F	5F	55	56	52	4C	47	40		40	3C	4D	48	5D	4D	5D	65	5F	62

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	5F	62	62	5D	56	58	51	48	45	44		41	3B	46	50	5D	4C	5F	56	5F	5D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	62	5C	62	62	54	58	54	49	45	37		46	48	4C	5A	5C	56	5D	55	5C	5F

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	5F	5C	53	5D	54	56	48	47	3C	30		42	47	4B	52	5C	55	65	56	62	6C

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	61	5F	62	5D	4F	56	5C	4C	3A	3D		32	48	4C	48	5F	55	5D	54	6C	6F

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	6C	62	62	5D	54	56	52	49	3C	40		44	3B	4C	54	62	54	62	56	5F	62

ตารางที่ 4.1 ระดับความแรงของสัญญาณ

เราสามารถนำค่าที่ได้จากตารางด้านบนมาทำการหาค่าเฉลี่ยได้ดังต่อไปนี้

ตัวที่	3										1	4									
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	63	60	5F	5F	53	57	52	49	40	3D		3E	43	4B	50	5D	52	5C	5B	62	66

ตารางที่ 4.2 ระดับความแรงของสัญญาณที่หาค่าเฉลี่ยแล้วในรูปฐาน 16

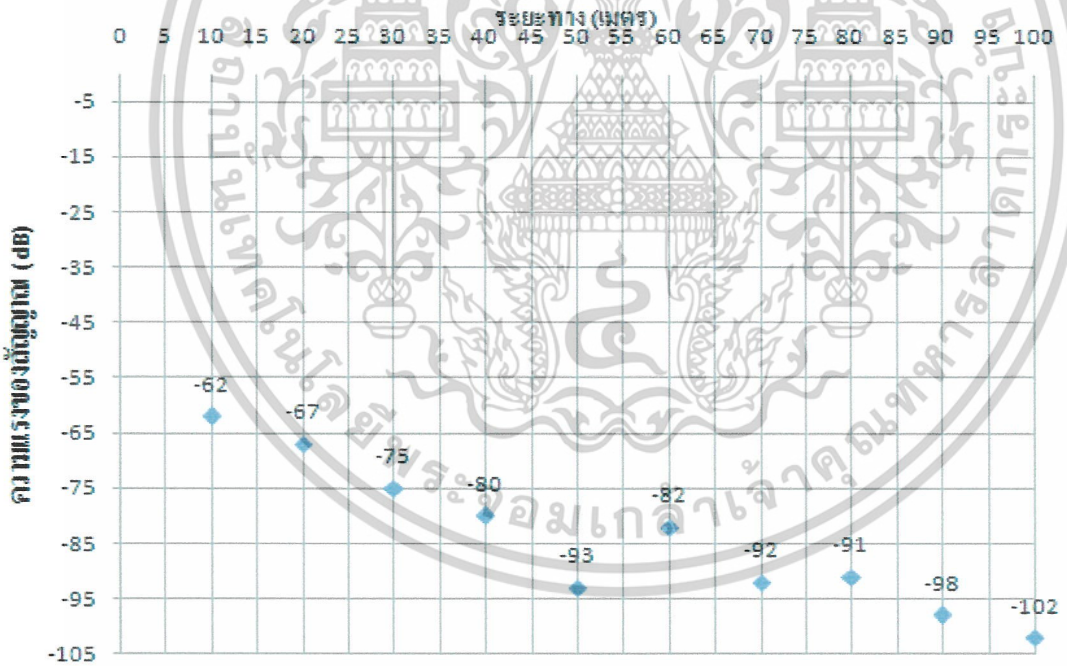
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากค่าที่เป็นฐาน 16 เมื่อทำการแปลงจากเลขฐาน 16 เป็นฐาน 10 เราจะได้ค่าความแรงของสัญญาณที่แท้จริงโดยที่ค่าเหล่านี้จะอยู่ในหน่วย dB ซึ่งค่าที่ได้จะเป็นไปตามตารางด้านล่าง

ตัวที่	3										1	4								
ระยะ	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
ความแรง	-99	-96	-95	-95	-83	-87	-82	-73	-64	-61		-62	-67	-75	-80	-93	-82	-92	-91	-98	-102

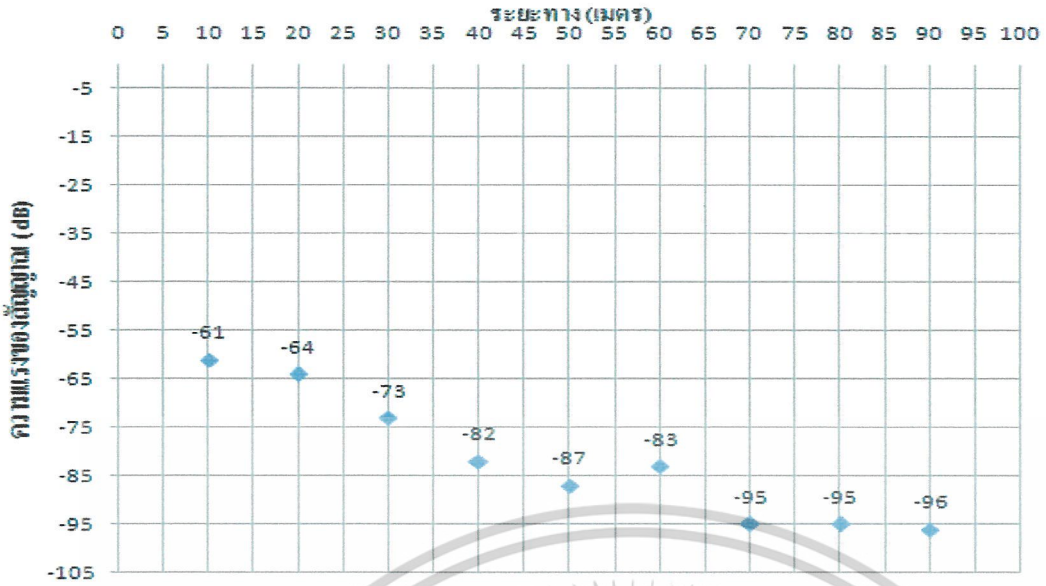
ตารางที่ 4.3 ระดับความแรงของสัญญาณที่หาค่าเฉลี่ยแล้วในรูปฐาน 10

เราสามารถเอาค่าความแรงของสัญญาณมาพอร์ตเปรียบเทียบกับระยะทางทั้งหมดซึ่งกราฟที่ได้จะเป็นไปตามกราฟด้านล่างดังนี้

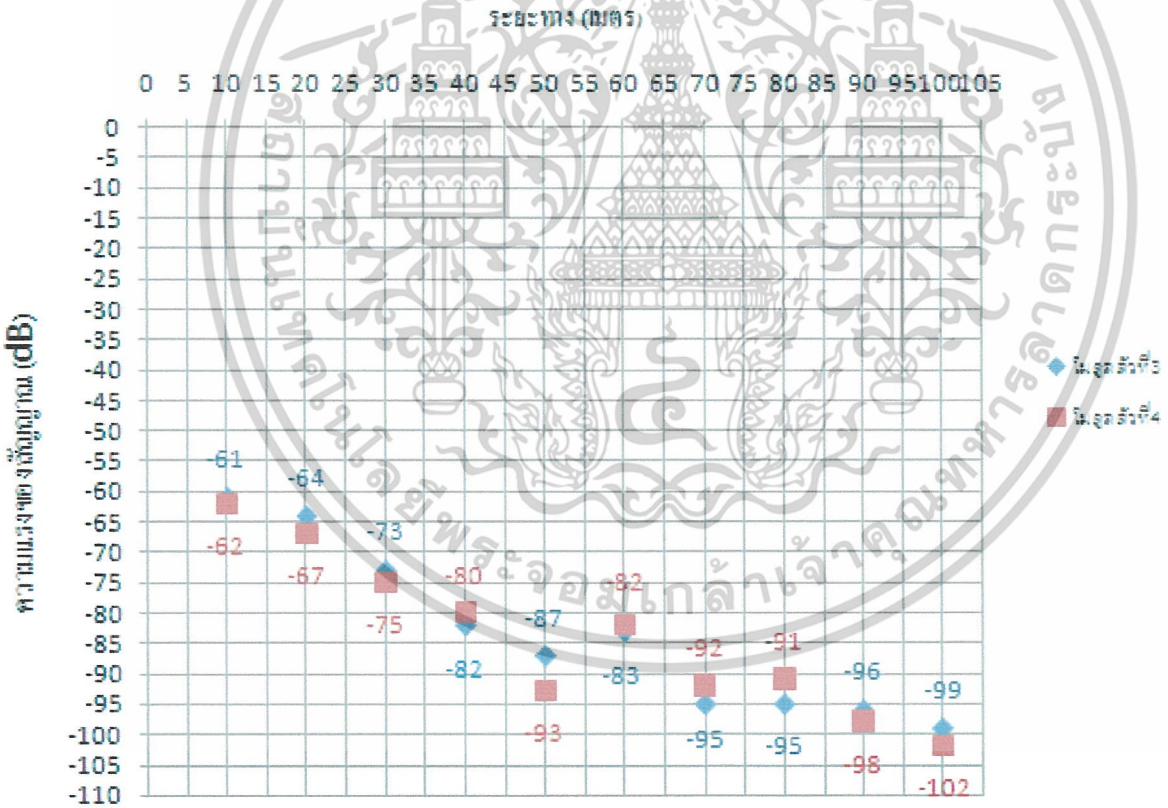


รูปที่ 4.2 กราฟเปรียบเทียบความแรงกับระยะทางของโมดูลตัวที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 กราฟเปรียบเทียบความแรงกับระยะทางของโมดูลตัวที่ 3



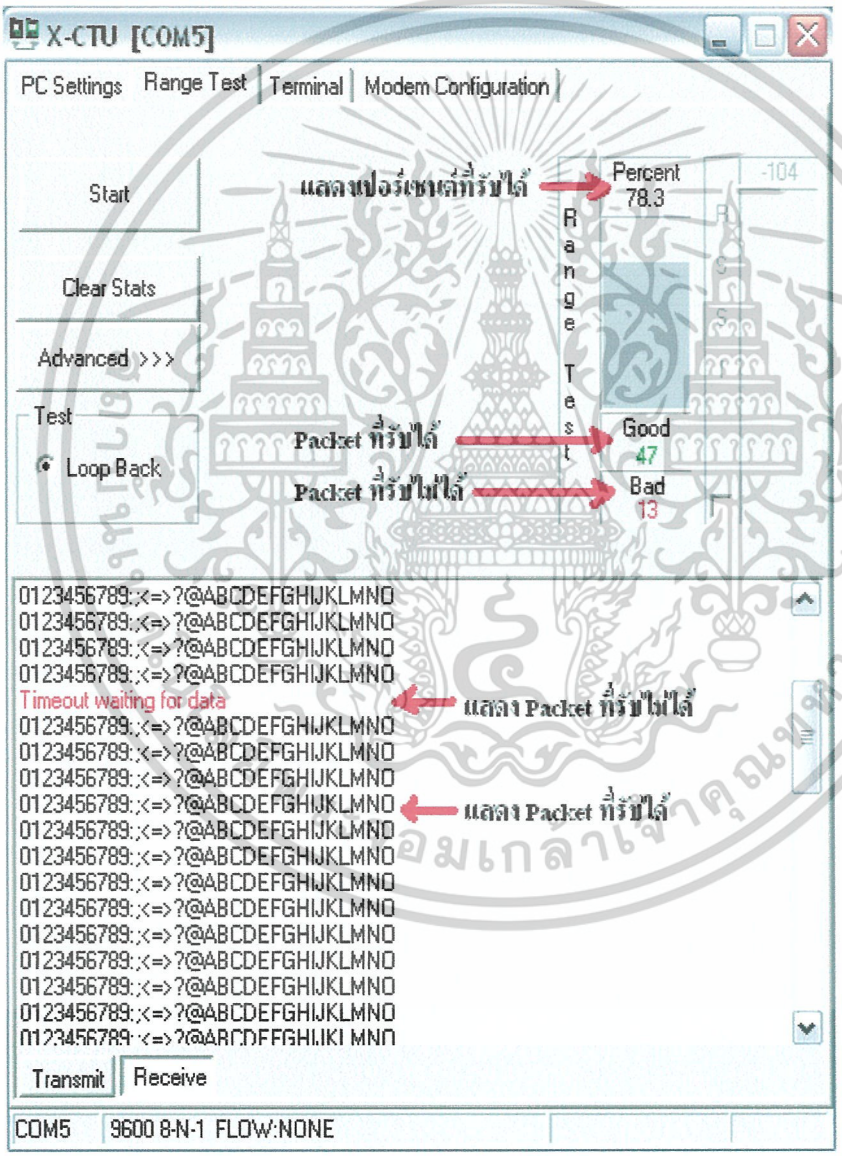
รูปที่ 4.4 กราฟเปรียบเทียบความแรงกับระยะทางของโมดูลตัวที่ 3 และ 4

จากกราฟจะเห็นว่าความแรงของสัญญาณที่รับได้นั้นจะลดลงตามความไกลของของระยะทางยิ่งระยะทางไกลขึ้นความแรงของสัญญาณก็จะยิ่งอ่อนลงตามไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองวัดความผิดพลาดในการรับ – ส่ง ข้อมูลโดยใช้โมดูล Zigbee เป็นตัวรับส่งข้อมูล

การทดลองวัดความผิดพลาดของสัญญาณเราจะอาศัยการส่งข้อมูลในรูปแบบของ Packet จำนวน 100 Packet แล้วเปลี่ยนระยะทางในการรับส่งข้อมูลออกไปทีละ 50 เมตรจำนวน 10 ระยะ แล้วสังเกตุจำนวนข้อมูลที่รับได้และข้อมูลที่สูญหายจากนั้นก็คำนวนเป็นเปอร์เซนต์ความผิดพลาดออกมาซึ่งในการวัดค่าความผิดพลาดเราจะดูจากเฟรมแวร์ในส่วนของ range test ดังต่อไปนี้



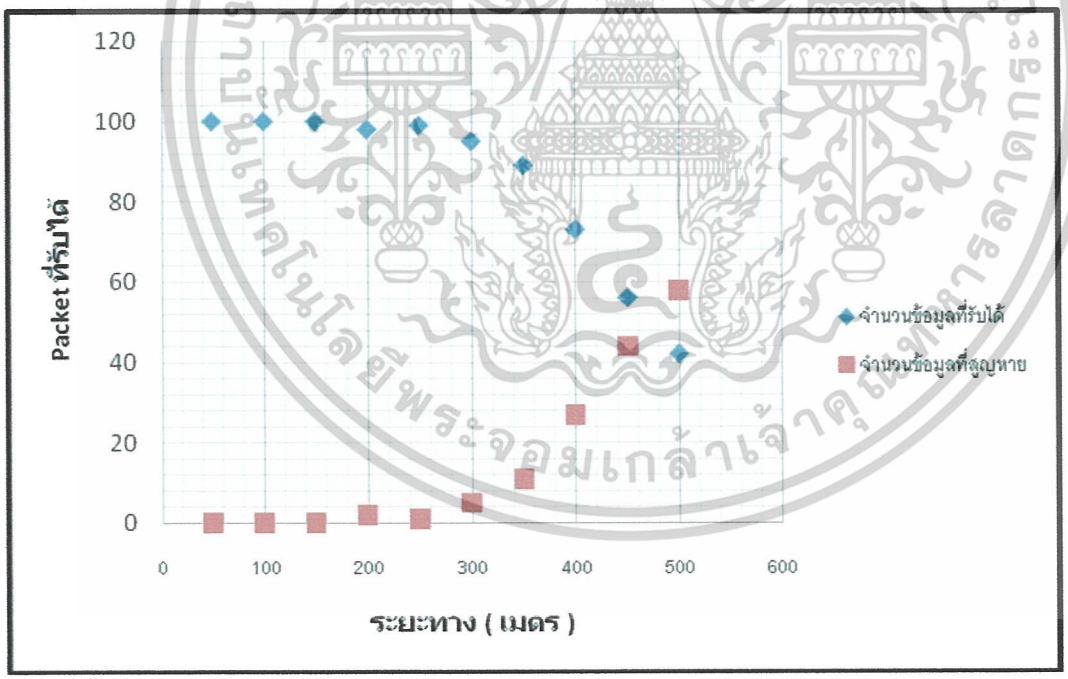
รูปที่ 4.5 หน้าต่างของ Range test ที่ใช้วัดความผิดพลาดการรับ – ส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระยะห่างในการวัด (เมตร)	50	100	150	200	250	300	350	400	450	500
จำนวนข้อมูลที่รับได้ (Packet)	100	100	100	98	99	95	89	73	56	42
จำนวนข้อมูลที่สูญหาย (Packet)	0	0	0	2	1	5	11	27	44	58
เปอร์เซนต์ความผิดพลาด	0%	0%	0%	2%	1%	5%	11%	27%	44%	58%

ตารางที่4.4 ข้อมูลการวัดความผิดพลาดในการรับส่งข้อมูล

จากตารางที่ 4.4 เราสามารถนำข้อมูลที่ได้นั้นมาทำการพรอตกราฟแสดงความสัมพันธ์ระหว่างระยะทางกับค่าความผิดพลาดที่เกิดขึ้นได้ดังนี้

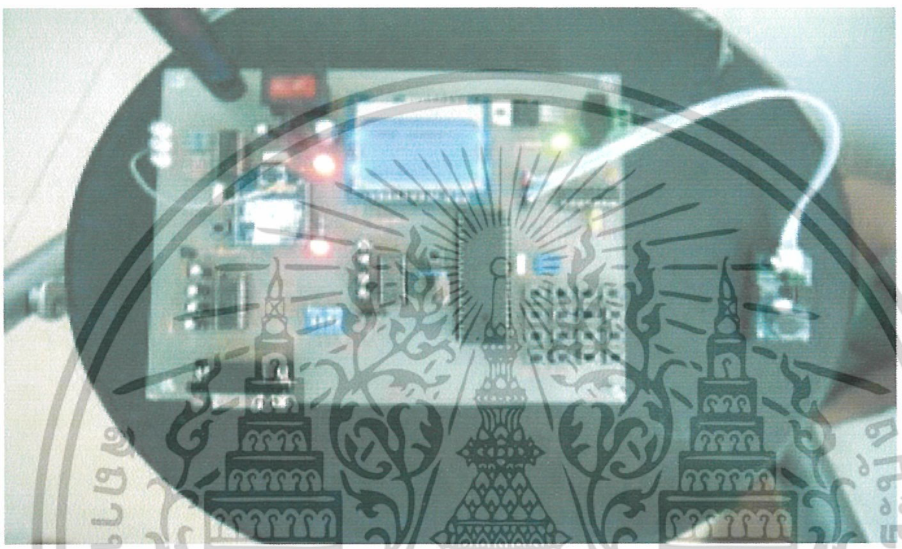


รูปที่ 4.6 กราฟเปรียบเทียบค่าความผิดพลาดกับระยะทาง

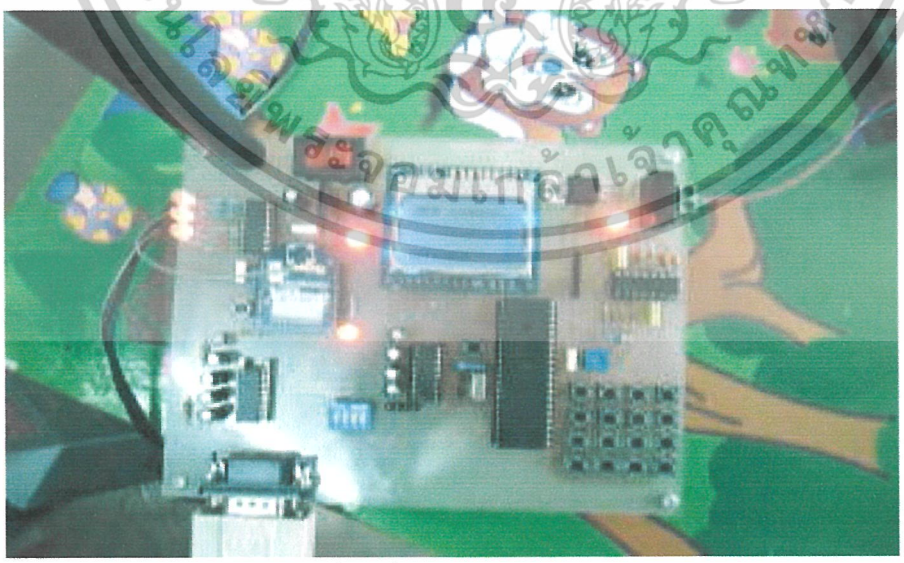
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองการส่งข้อมูลที่เป็นเสียงพูดแสดงออกทาง X-CTU ทางฝั่งรับ

จากการทดลองในส่วนของการส่งข้อมูลที่เป็นเสียงพูดผ่านโมดูลแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแล้วส่งออกทางโมดูลโดยที่ฝั่งรับรับข้อมูลทางโมดูลแล้วแสดงผลออกโปรแกรม X-CTU โดยที่ระยะทางในการทดลองใช้ระยะ 10 เมตร ซึ่งผลของการทดลองเราจะเห็นการเปลี่ยนแปลงของข้อมูลที่ได้รับเกิดการเปลี่ยนแปลงขณะมีการส่งเสียงพูดเข้าปรีไมค์ซึ่งข้อมูลเหล่านี้เราจะนำไปแปลงเป็นสัญญาณอนาลอกต่อไป



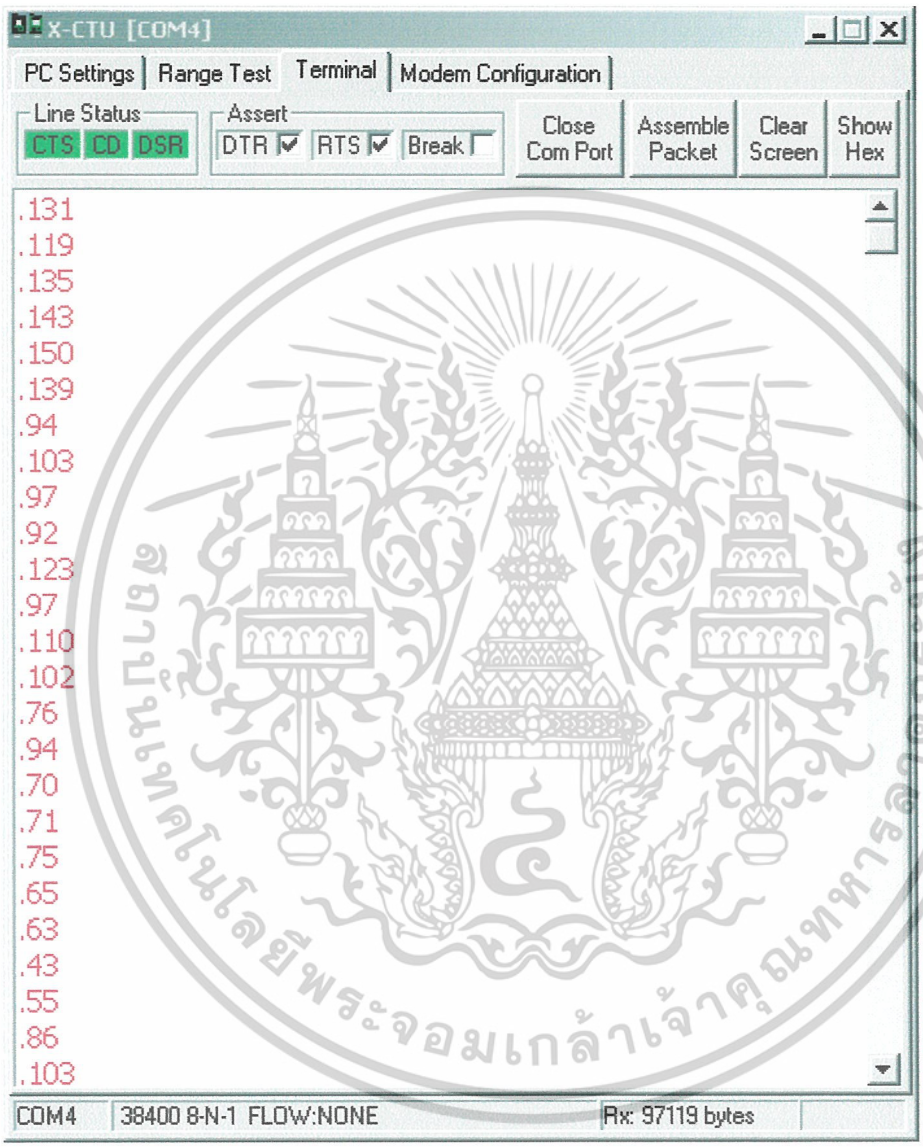
รูปที่ 4.7 ฝั่งส่งข้อมูลเสียง



รูปที่ 4.8 ฝั่งรับข้อมูล

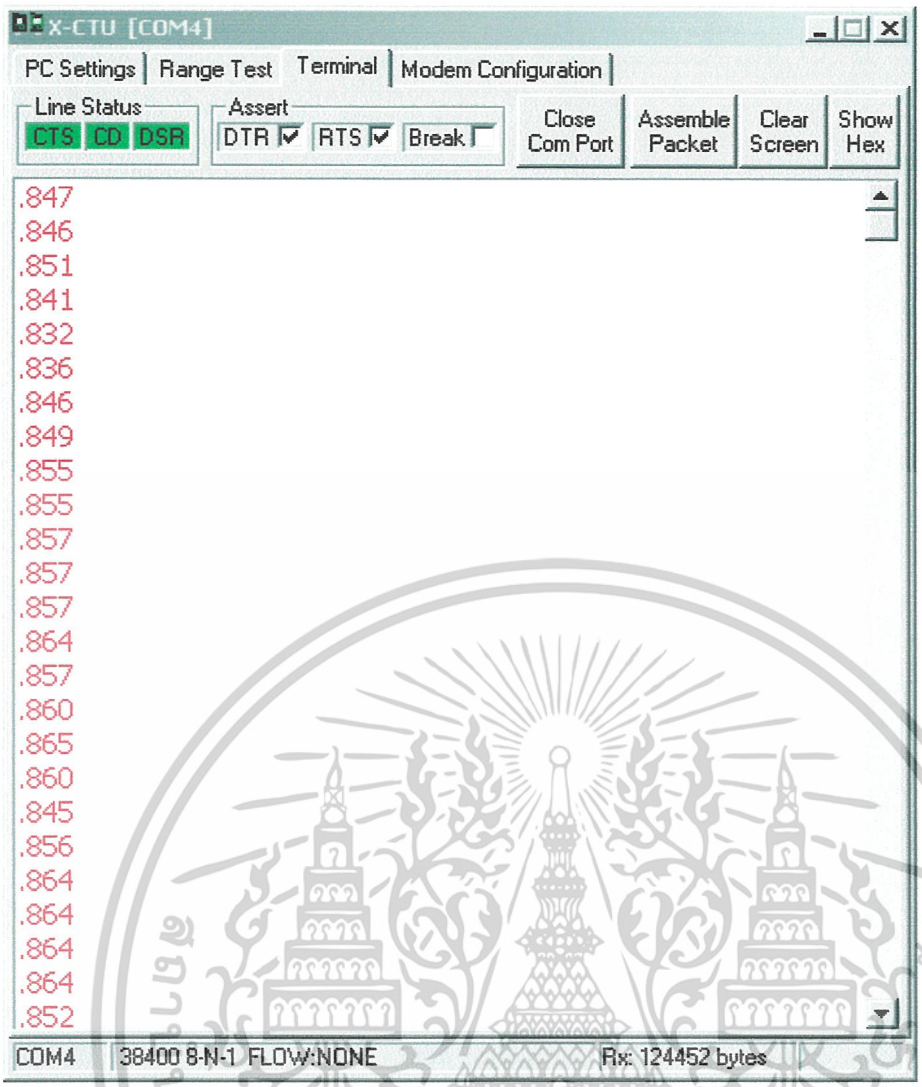
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปด้านบนจะเห็นว่าฝั่งส่งจะมีวงจรปรีไมค์ที่ใช้ในการรับเสียงพูดเพื่อส่งเข้าสู่บอร์ดควบคุม เพื่อนำไปแปลงสัญญาณอนาลอกเป็นดิจิตอลแล้วส่งออกไปสู่ฝั่งรับโดยใช้โมดูล Zigbee เป็นโมดูล รับ – ส่งข้อมูลส่วนทางด้านฝั่งรับก็รับข้อมูลเข้าทางโมดูลแล้วแสดงผลออกทาง Hyper Terminal ของโปรแกรม X-CTU ซึ่งผลการทดลองสามารถแสดงได้จากโปรแกรม X-CTU ดังนี้



รูปที่ 4.9 ระดับข้อมูลในขณะที่ยังไม่ได้รับสัญญาณเสียง

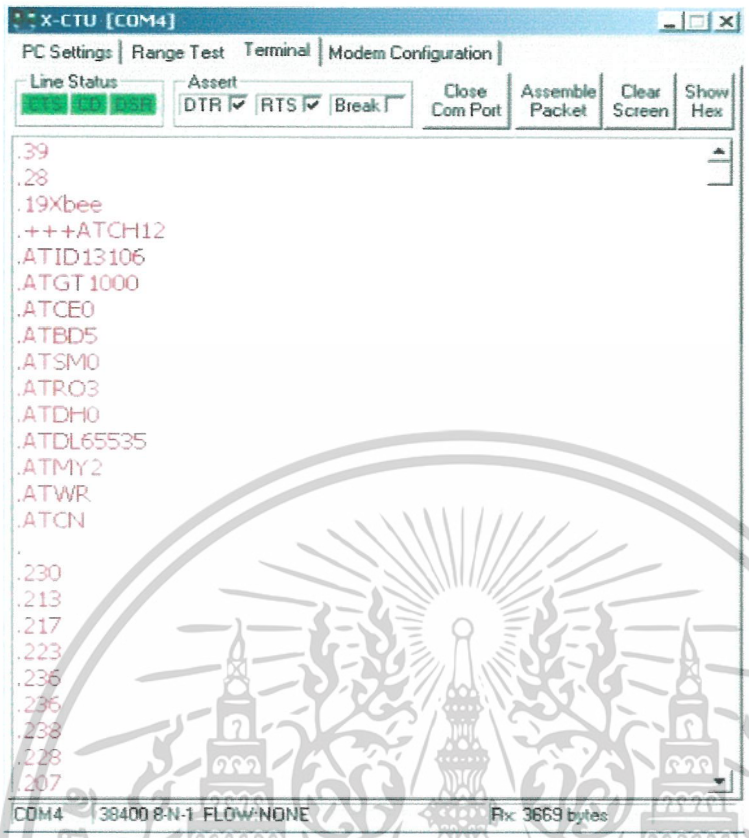
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 ระดับข้อมูลในขณะที่ยังไม่ได้รับสัญญาณเสียง

จากรูปที่ 4.10 จะเห็นว่าในขณะที่ยังไม่มีการพูดเข้าไปที่ปรีไมค์ระดับสัญญาณจะสวิงอยู่ในระดับที่ต่ำซึ่งจะอยู่ในช่วง 0 – 200 mV เหตุที่มีแรงดันเพิ่มขึ้นบ้างในบางเวลาและลดลงบ้างในบางเวลานั้นเป็นผลมาจากผลการตอบสนองของปรีไมค์ที่มีผลต่อสัญญาณเสียงจากแหล่งกำเนิดอื่นรอบๆตัวของปรีไมค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 การดึงข้อมูลจากทางฝั่งส่งแสดงออกจอ

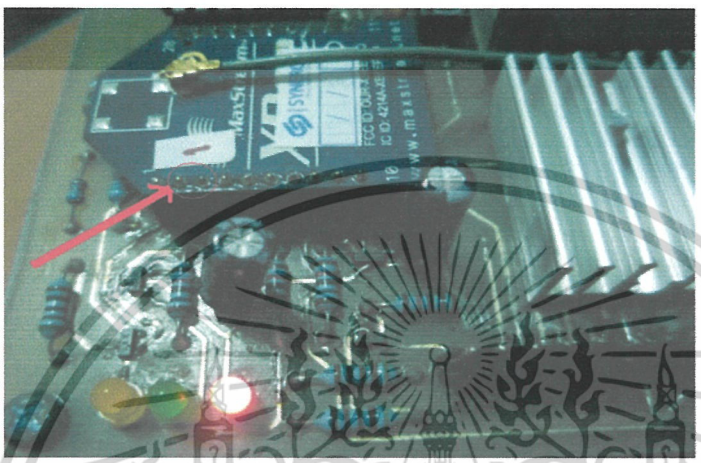
จากรูปเป็นการดึงข้อมูลของโมดูลทางฝั่งส่งแสดงออกทางโปรแกรม X – CTU ซึ่งจะเห็นว่าข้อมูลจะแสดงข้อมูลต่างๆที่ได้ทำการตั้งค่าไว้ก่อนหน้านี้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองวัดรูปสัญญาณต่างๆ

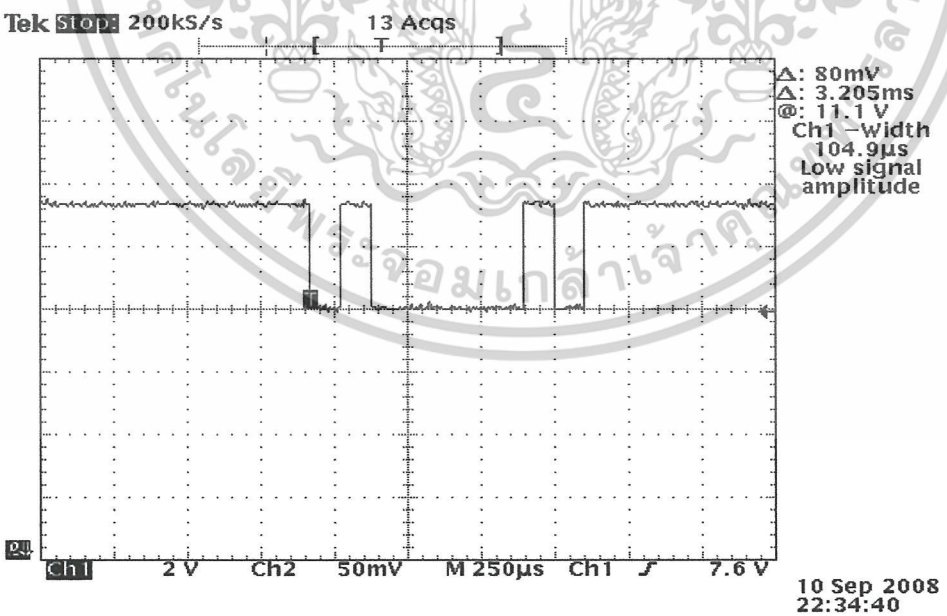
4.4.1 การวัดรูปสัญญาณในส่วนของการรับ –ส่งข้อมูลระหว่างโมดูลโดยการส่งข้อมูลเป็นรหัสแอสกี้

การวัดรูปสัญญาณการรับส่งข้อมูลระหว่างโมดูลนั้นจะอาศัยการจับรูปสัญญาณทางด้านฝั่งรับโดยที่ฝั่งส่งจะทำการส่งข้อมูลจากแป้นพิมพ์ของคอมพิวเตอร์ในรูปแบบของรหัสข้อมูล ASCII ซึ่งจุดที่วัดสัญญาณเราจะจับสัญญาณที่ขา TX ของโมดูล zigbee ฝั่งรับ ดังรูป



รูปที่ 4.12 ตำแหน่งการจับสัญญาณ

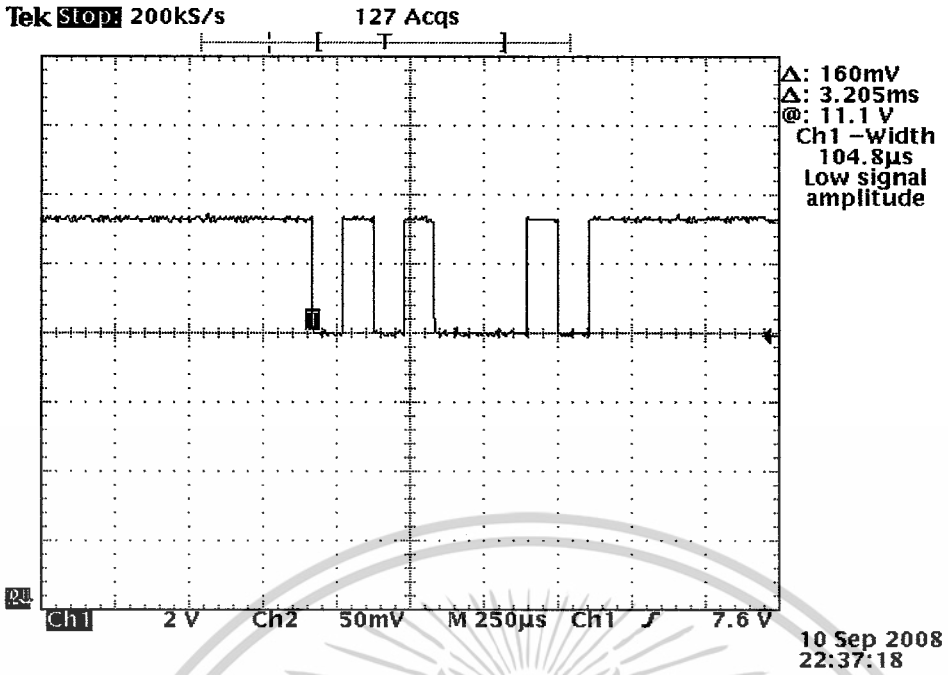
ในการวัดรูปสัญญาณได้ทำการวัดปันจำนวน 8 รูปสัญญาณจาก 8 รหัส ASCII ดังต่อไปนี้



รูปที่ 4.13 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น A

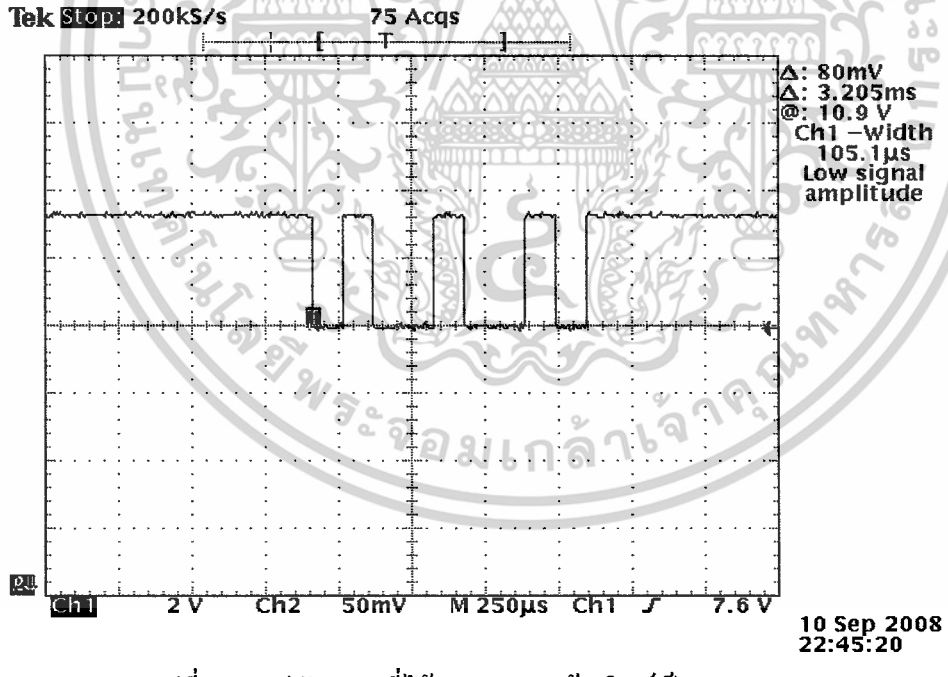
แสดงเป็น binary จะได้เท่ากับ 01000001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น E

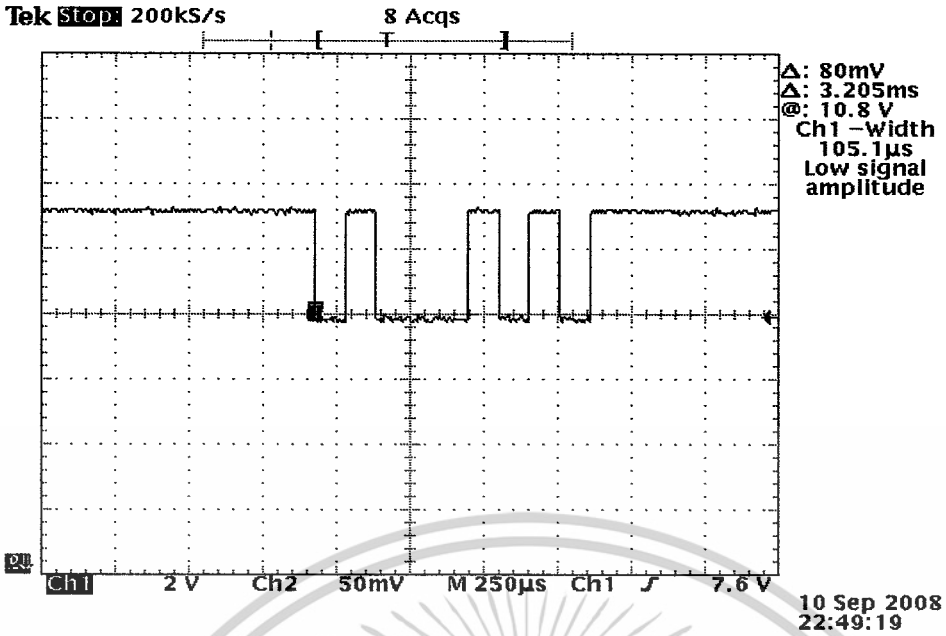
เป็น binary จะได้เท่ากับ 01000101



รูปที่ 4.15 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น I

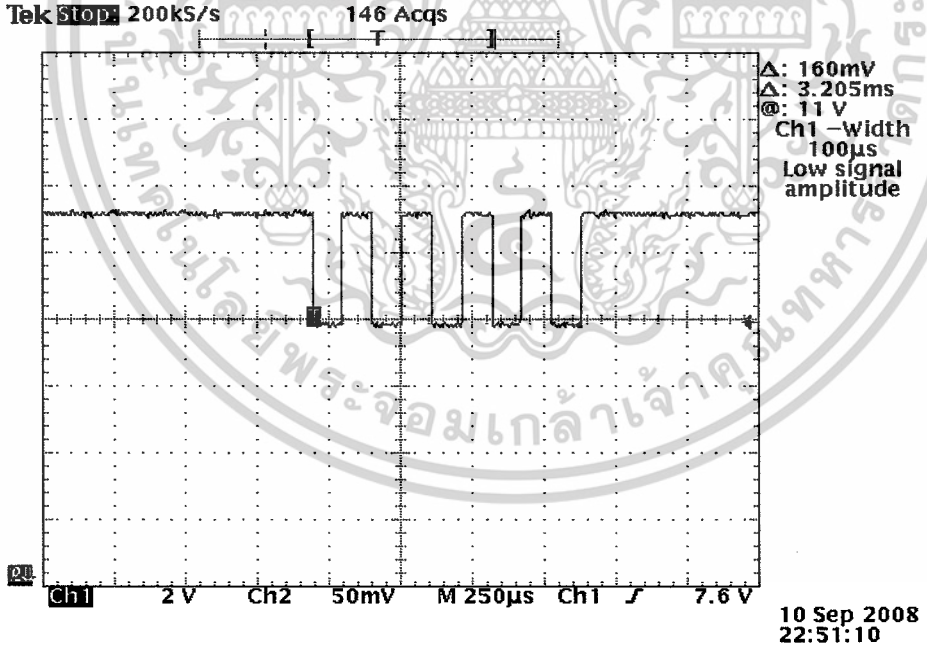
เป็น binary จะได้เท่ากับ 01001001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น Q

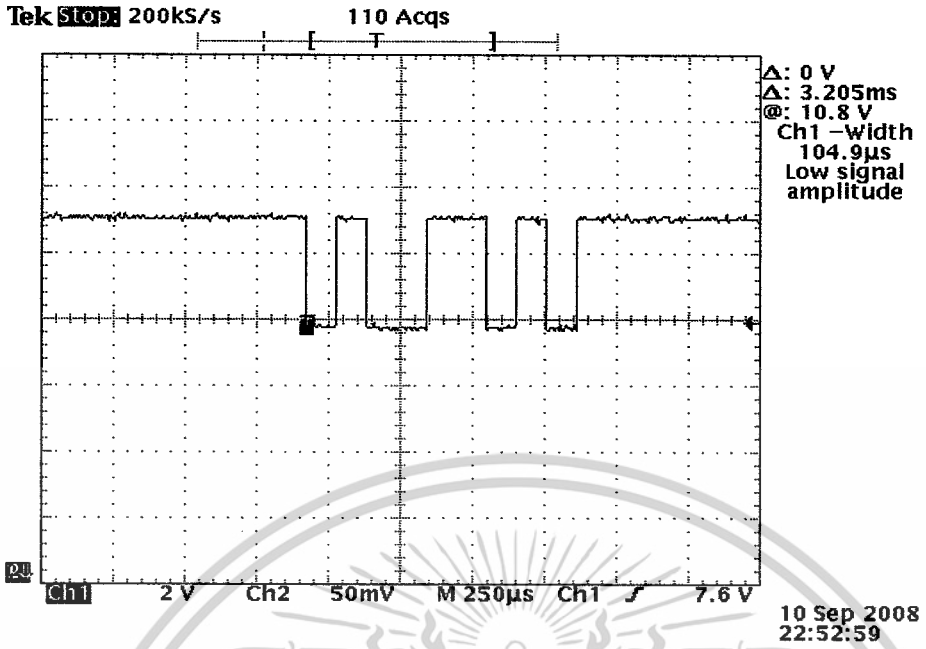
เป็น binary จะได้เท่ากับ 01010001



รูปที่ 4.17 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น U

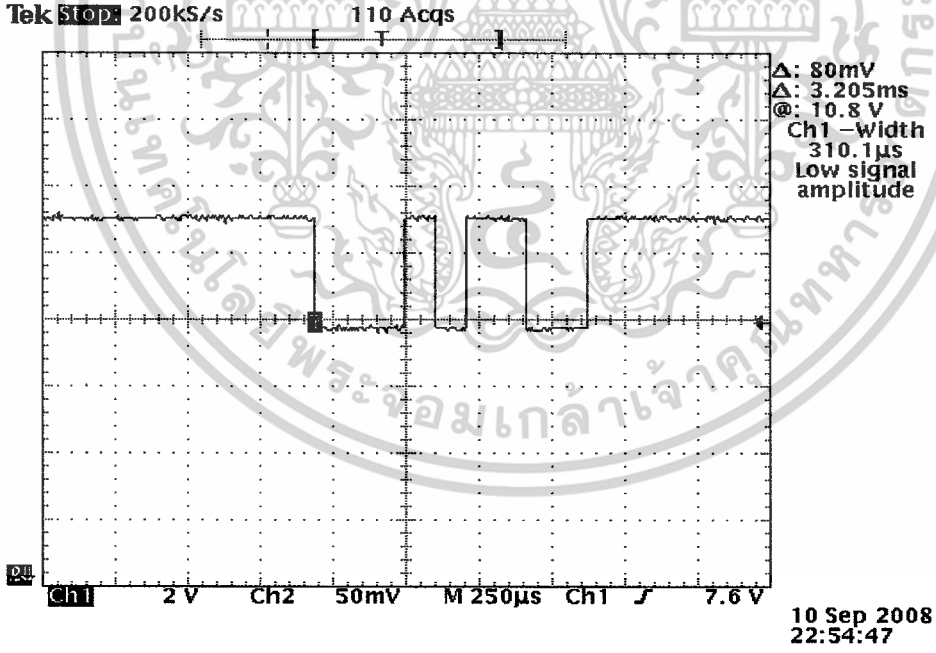
เป็น binary จะได้เท่ากับ 01010101

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 แสดงรูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็นY

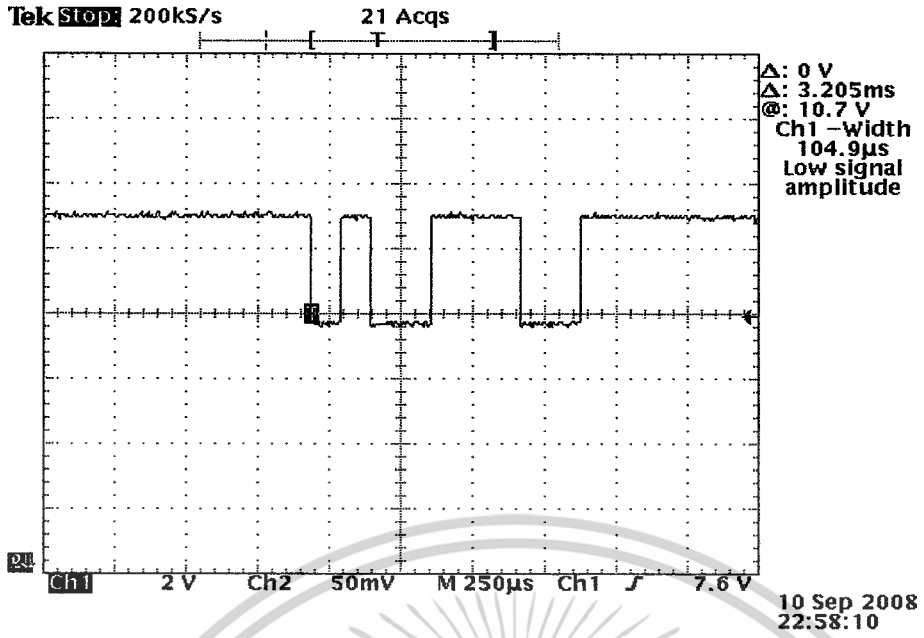
แสดงเป็น binary จะได้เท่ากับ 01011001



รูปที่ 4.19 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น 4

เป็น binary จะได้เท่ากับ 00110100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

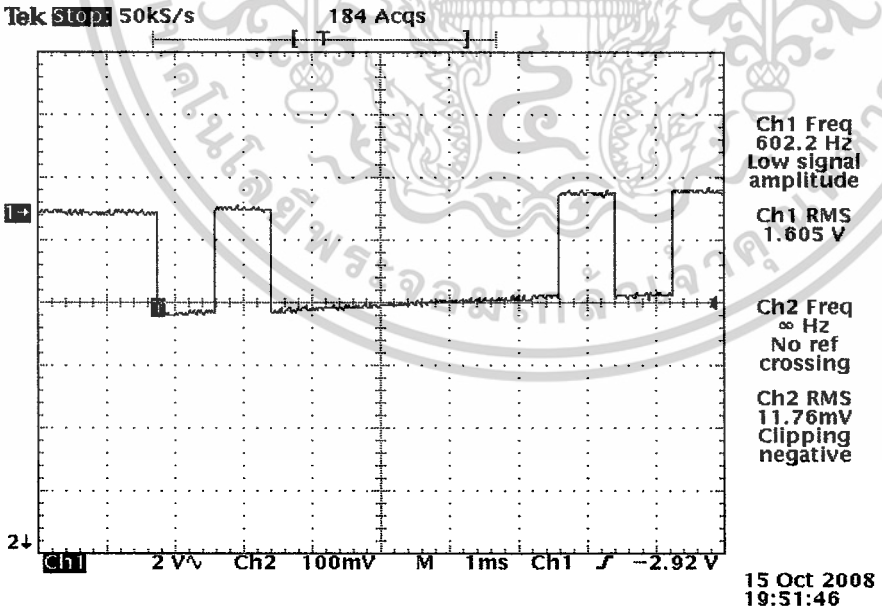


รูปที่ 4.20 รูปสัญญาณที่ได้จากการกดแป้นพิมพ์เป็น 9

เป็น binary จะได้เท่ากับ 00111001

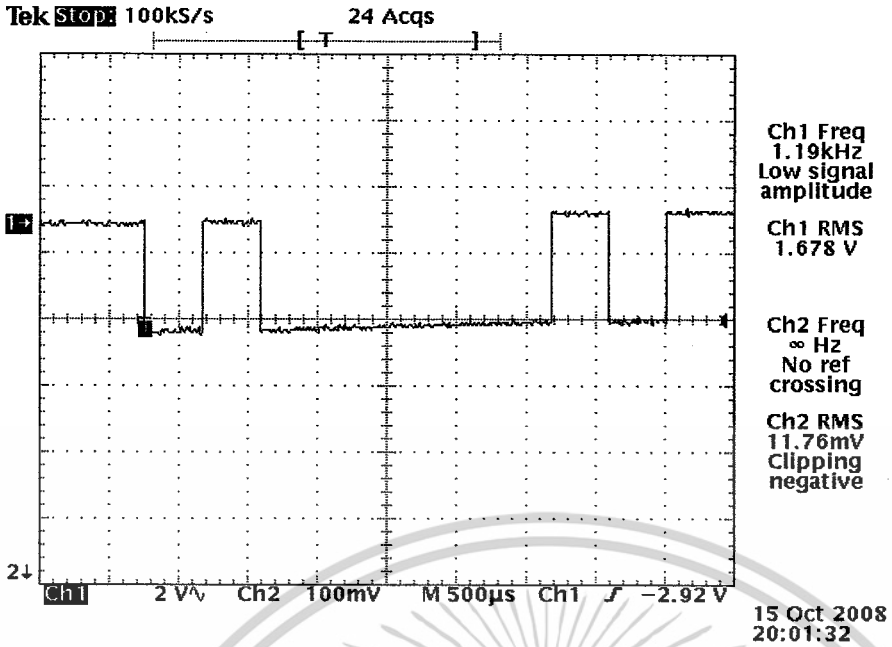
4.4.2 การทดลองวัดรูปสัญญาณการส่งข้อมูลเป็นอักษร A โดยที่อัตตราร็วในการส่งเปลี่ยนแปลงไป

การทดสอบเปลี่ยนค่า อัตราเร็วข้อมูล (Baud Rate) ทั้งหมด 8 ค่า โดยใช้การส่งข้อมูลเป็นตัวอักษร A เพียงอย่างเดียวแล้วเปรียบเทียบผลที่รับได้ทางฝั่งรับ

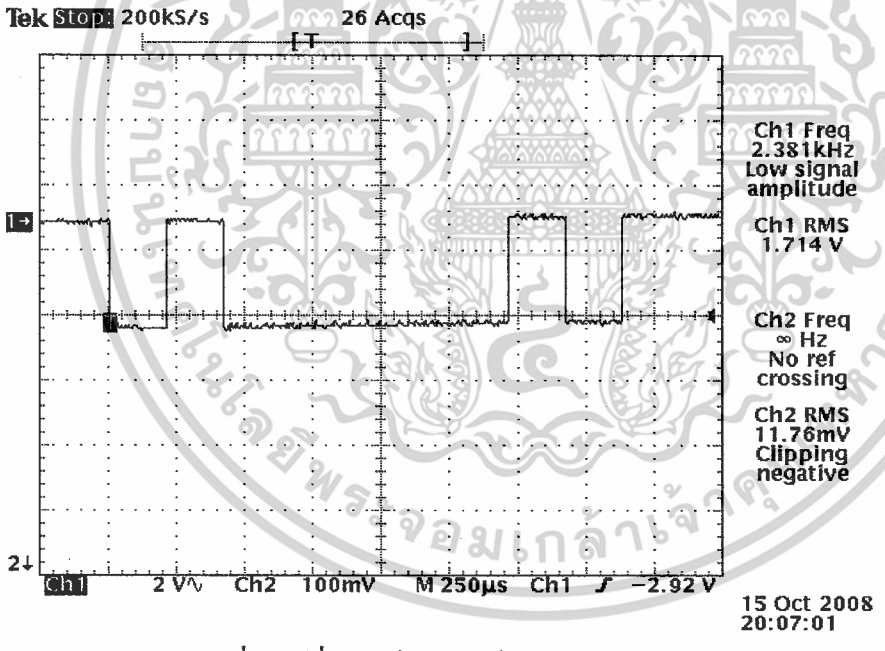


รูป 4.21 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 1200 bps โดยการส่งอักษร A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

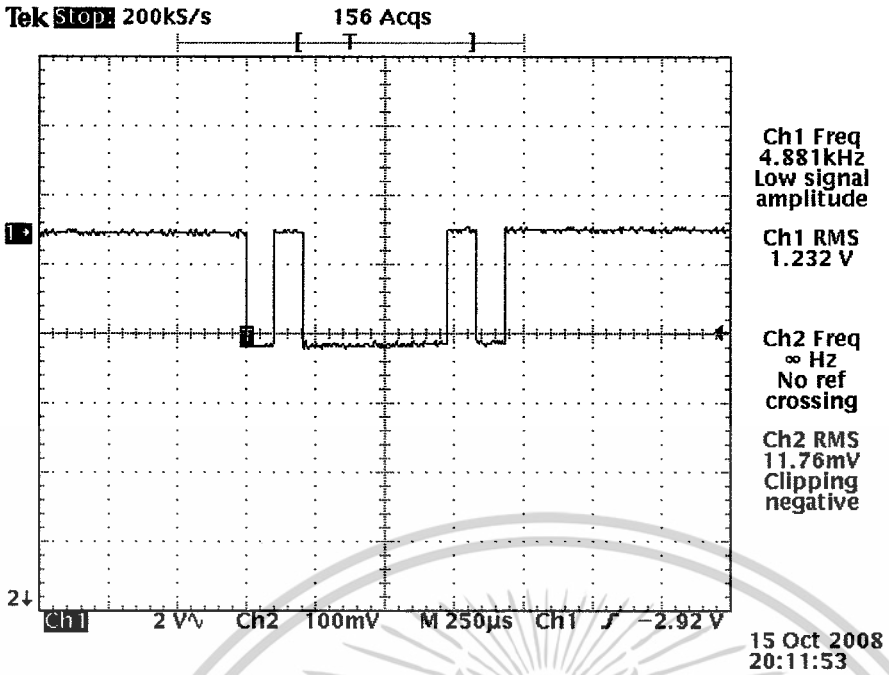


รูป 4.22 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 2400 bps โดยการส่งอักษร A

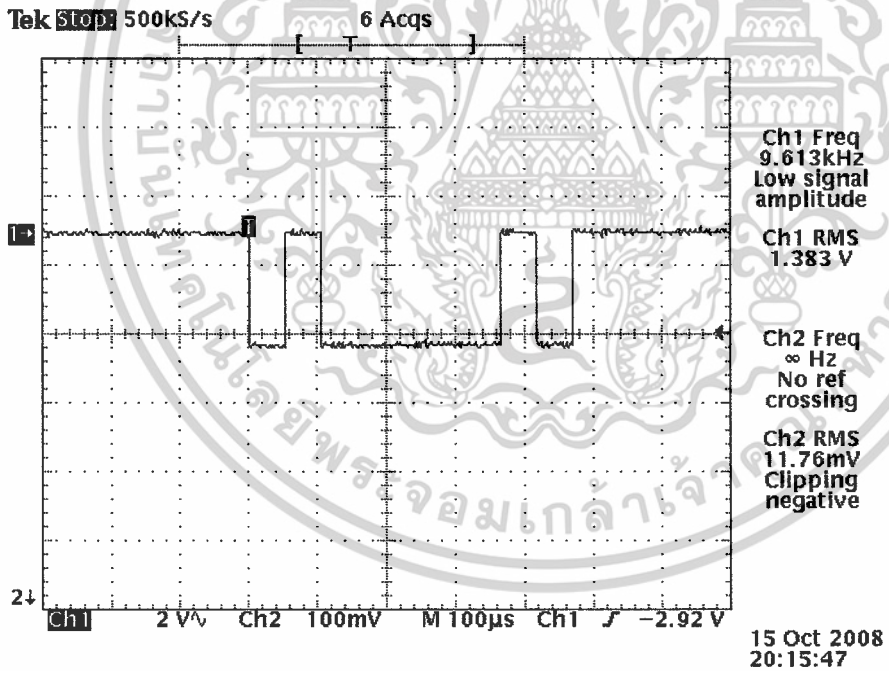


รูป 4.23 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 4800 bps โดยการส่งอักษร A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

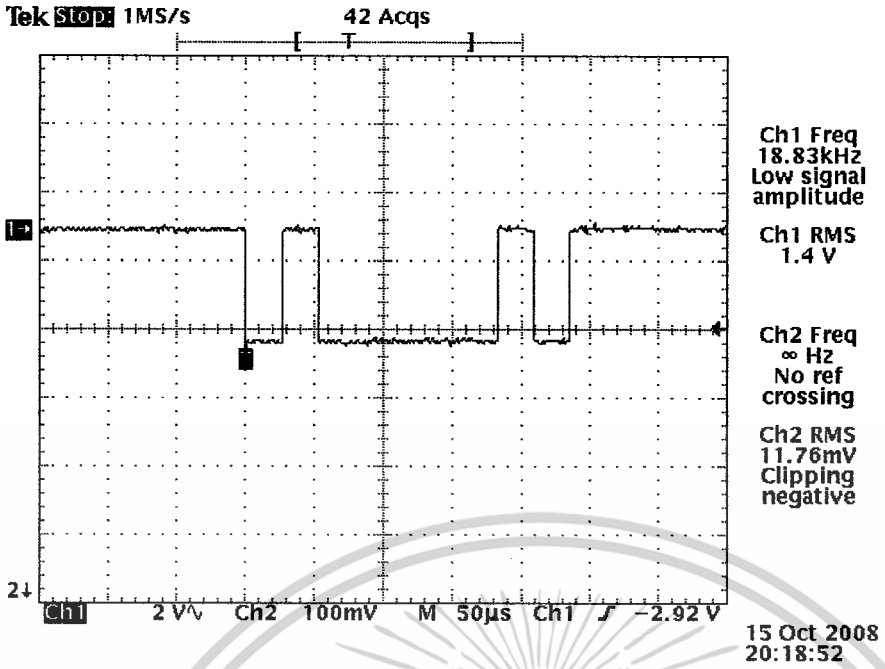


รูป 4.24 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 9600 bps โดยการส่งอักษร A

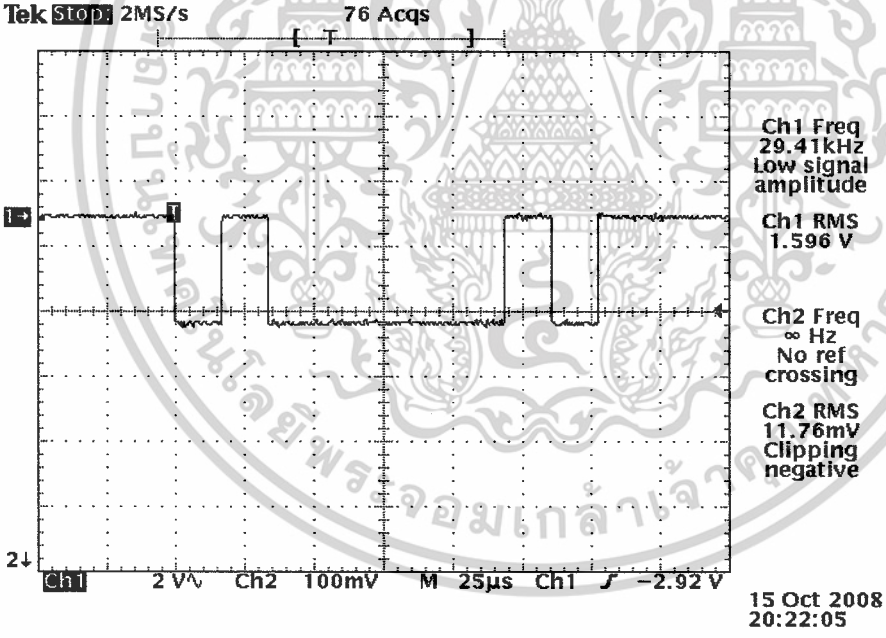


รูป 4.25 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 19200 bps โดยการส่งอักษร A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

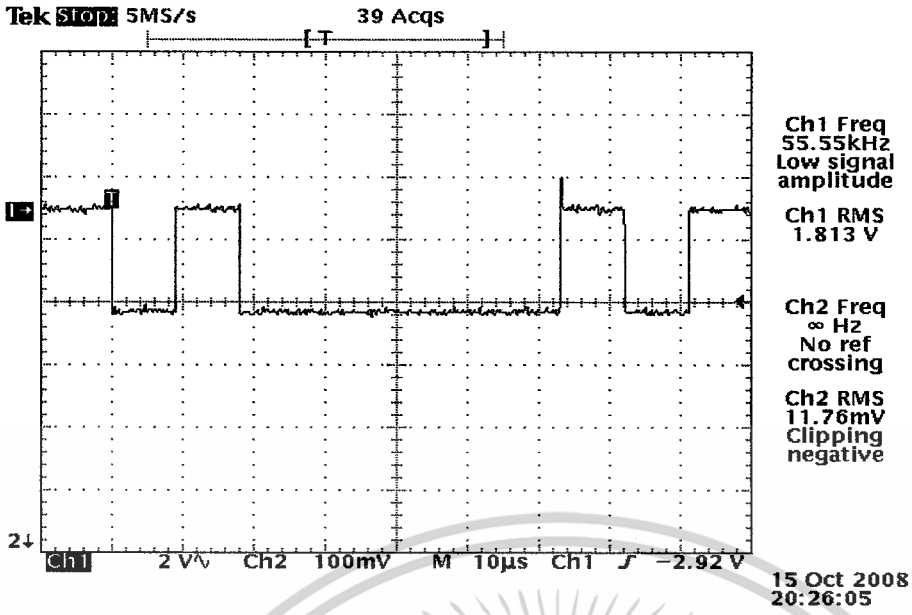


รูป 4.26 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 38400 bps โดยการส่งอักษร A



รูป 4.27 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 57600 bps โดยการส่งอักษร A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

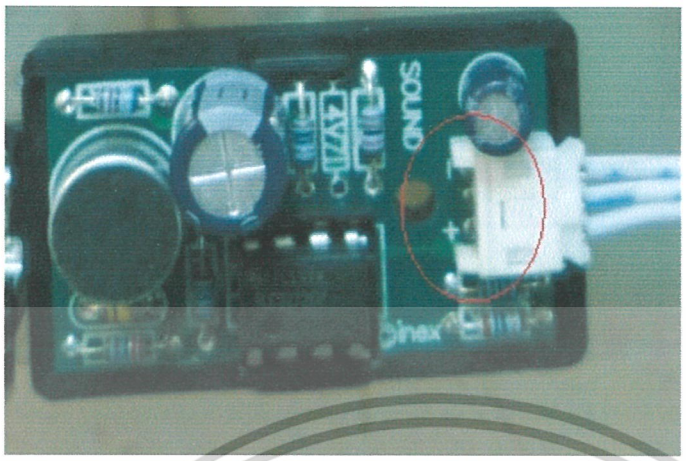


รูป 4.28 สัญญาณที่วัดได้ที่อัตราเร็วข้อมูลเป็น 115200 bps โดยการส่งอักษร A

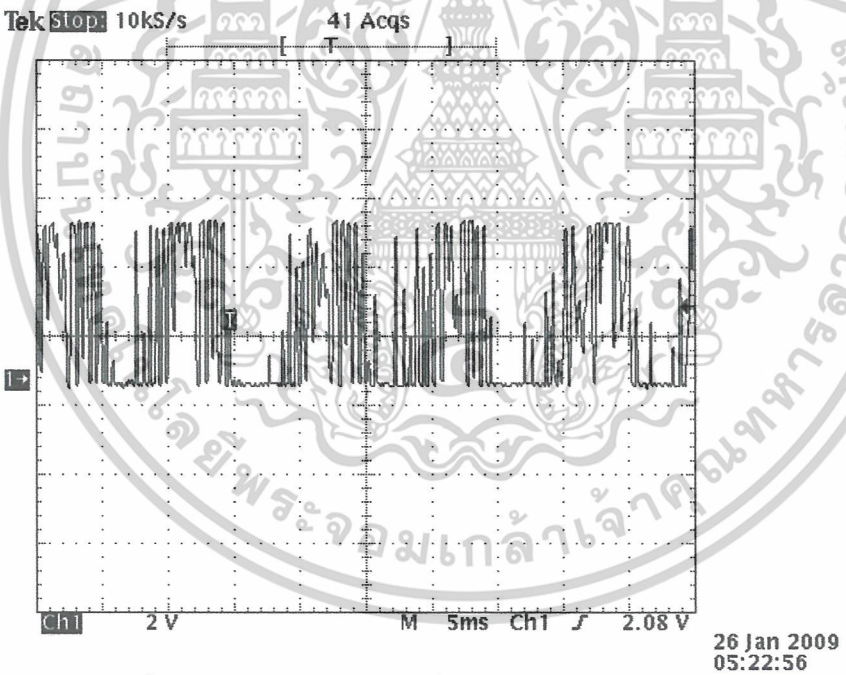
จากรูปสัญญาณของการเปลี่ยนอัตราเร็วข้อมูลที่ต่างกันไปจะพบว่าเมื่อค่าอัตราเร็วข้อมูลสูงขึ้นจะทำให้เวลาในการส่งข้อมูลในแต่ละบิตน้อยลงแต่อาจจะเกิดการผิดพลาดในการรับ – ส่งสัญญาณสูงขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.3 การทดลองวัดรูปสัญญาณในส่วนของสัญญาณอินพุทที่เป็นเสียงพูด



รูปที่ 4.29 จุดวัดสัญญาณอินพุต

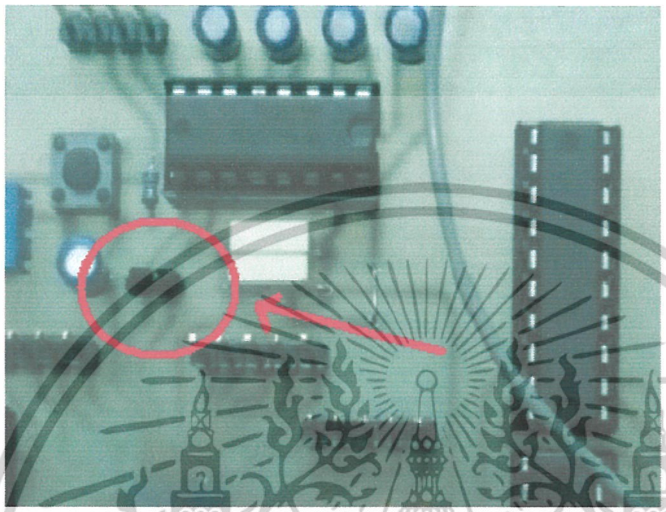


รูปที่ 4.30 รูปสัญญาณเสียงพูดที่จะป้อนเป็นอินพุต

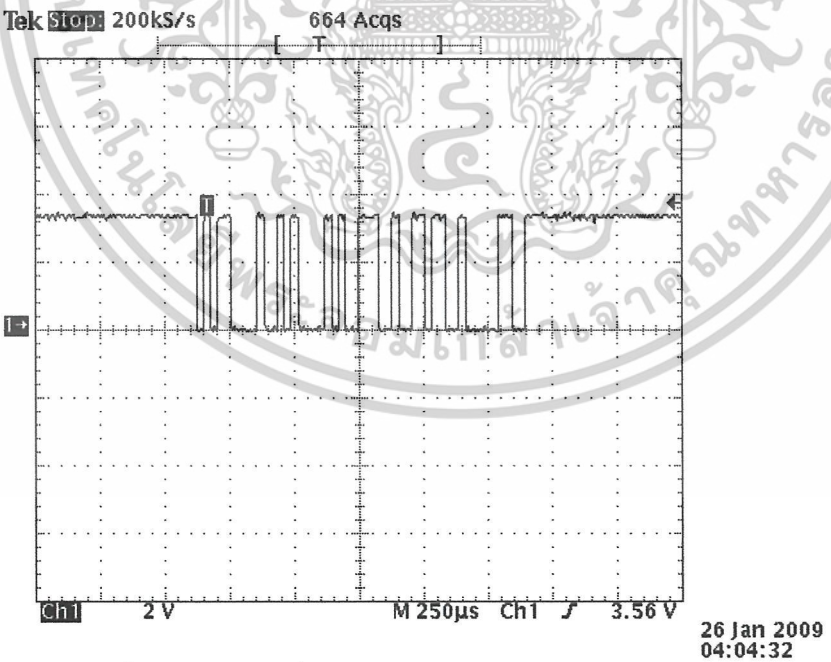
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.4 การวัดรูปสัญญาณในส่วนของฝั่งส่งที่รับข้อมูลมาจากปรีไมค์

การวัดสัญญาณทางฝั่งส่งเราจะอาศัยการรับอินพุตจากปรีไมค์แล้วส่งมาที่ไมโครคอนโทรลเลอร์ แล้วทำการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลเพื่อส่งไปให้ Zigbee โดยผ่านบัฟเฟอร์เพื่อลดระดับแรงดันจากนั้นโมดูลก็จะทำการส่งข้อมูลออกไป

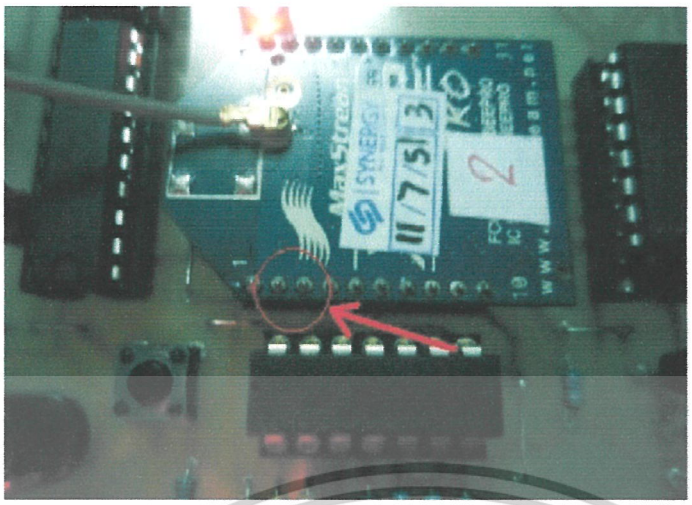


รูปที่ 4.31 จุดวัดสัญญาณในส่วนของฝั่งส่งที่รับข้อมูลมาจากปรีไมค์

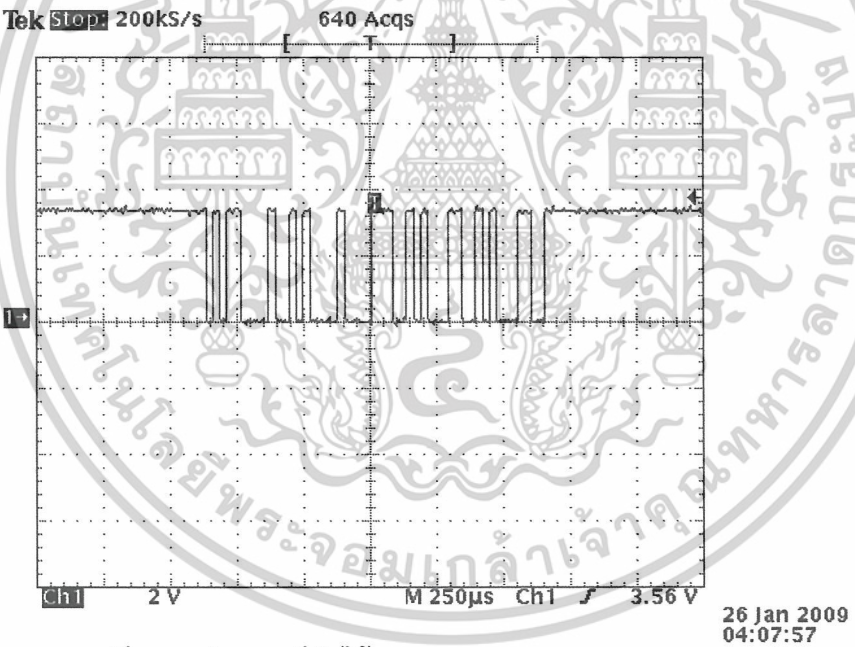


รูปที่ 4.32 สัญญาณที่วัดได้จากไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.33 จุดวัดสัญญาณที่ขา TX ของ Zigbee

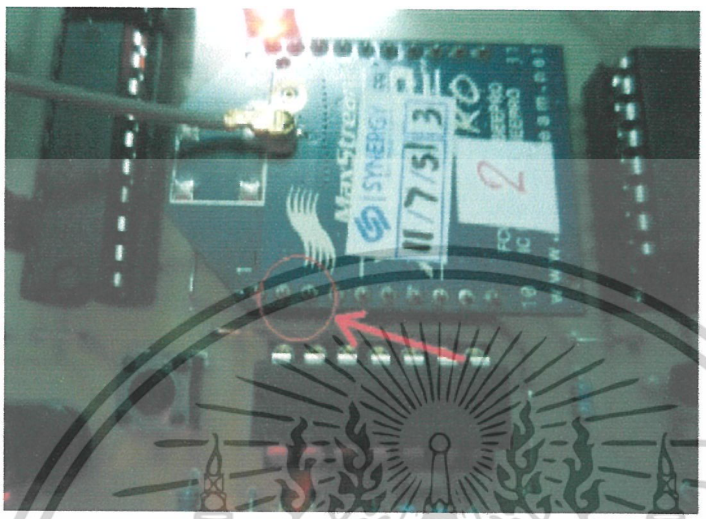


รูปที่ 4.34 สัญญาณที่วัดได้จากขา TX ของ Zigbee

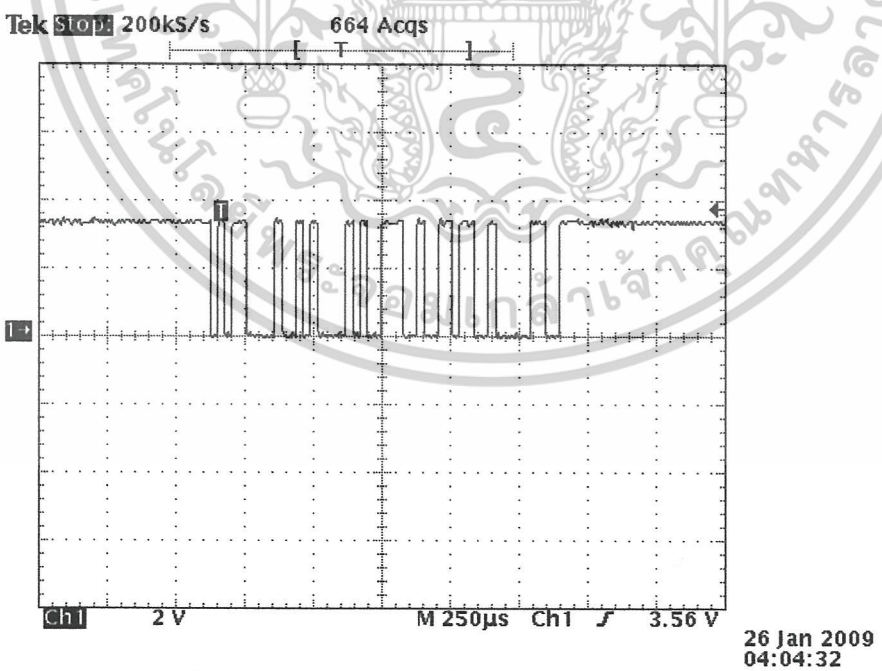
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.5 การทดลองวัดรูปสัญญาณทางฝั่งรับที่ได้รับข้อมูลจากฝั่งส่ง

ส่วนของการวัดสัญญาณทางฝั่งรับเราจะอาศัยการรับข้อมูลเข้ามาทางโมดูลแล้วนำข้อมูลที่ได้มาทำการปรับแรงดันเพื่อส่งไปให้ไมโครคอนโทรลเลอร์เพื่อแปลงเป็นสัญญาณอนาลอกส่งออกลำโพง

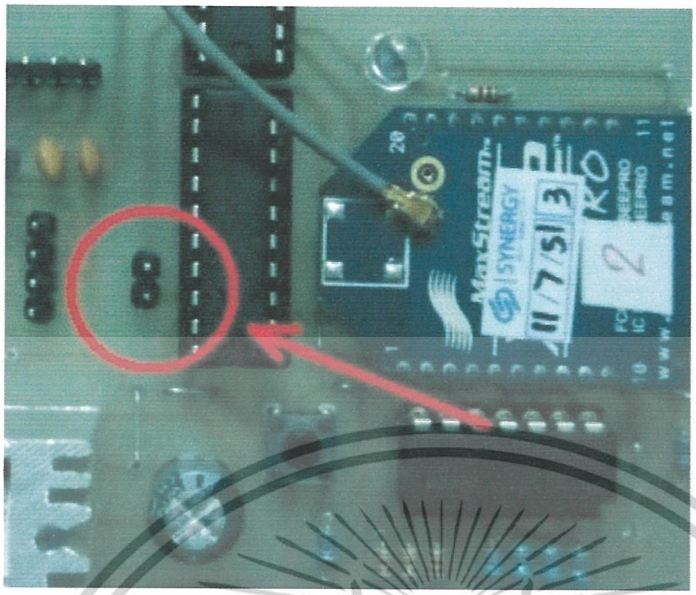


รูปที่ 4.35 จุดวัดสัญญาณที่ขา RX ของ Zigbee

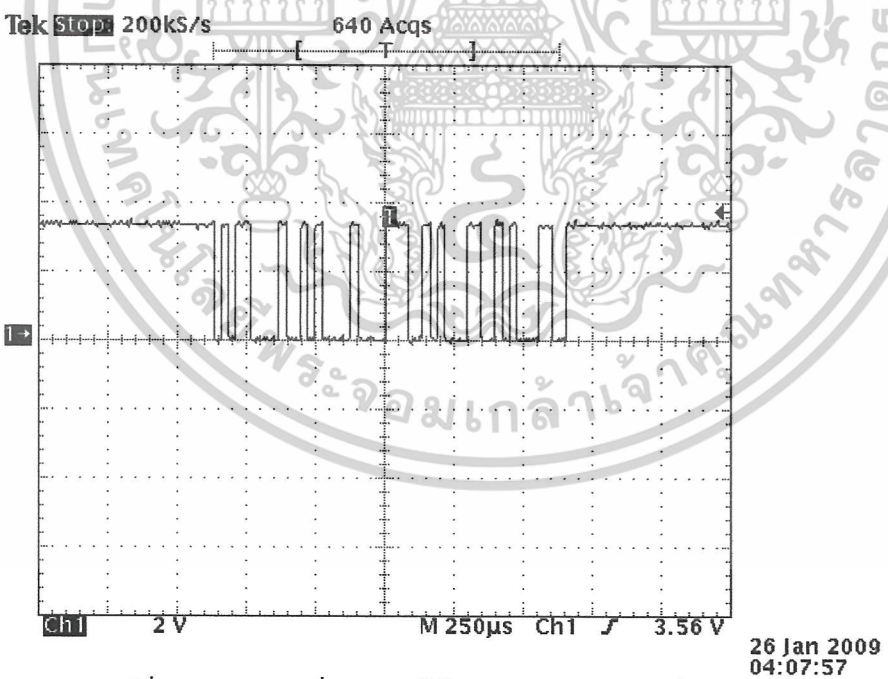


รูปที่ 4.36 สัญญาณดิจิตอลที่รับได้จากขา RX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.37 จุดวัดสัญญาณก่อนที่จะเข้าไมโครคอนโทรลเลอร์

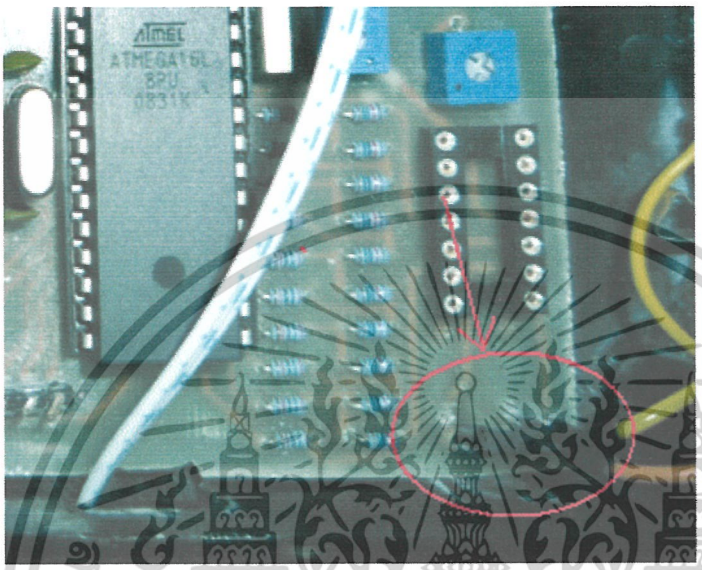


รูปที่ 4.38 สัญญาณที่ก่อนส่งเข้าไมโครคอนโทรลเลอร์

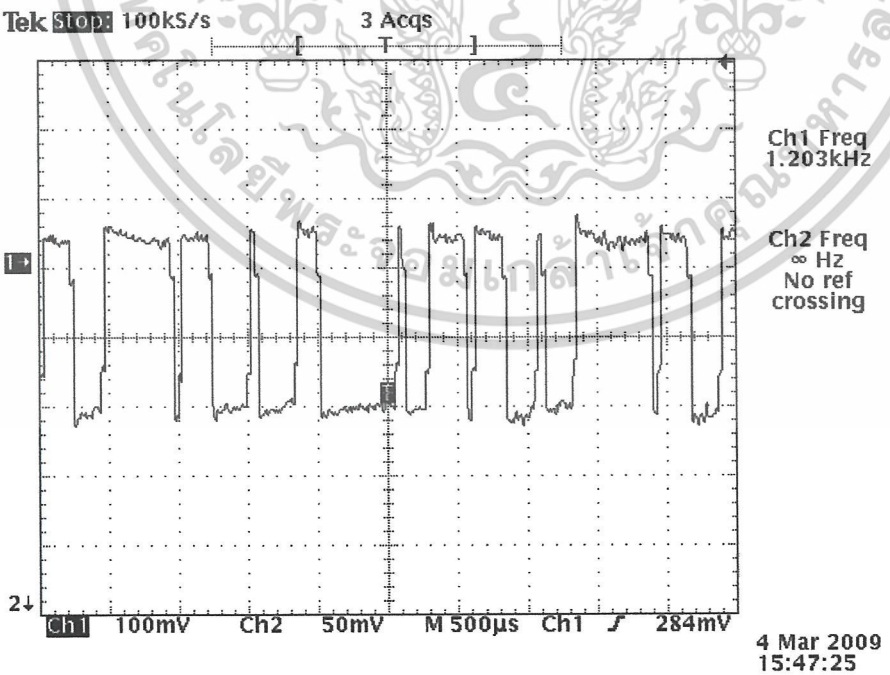
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.6 การทดลองวัดรูปสัญญาณทางด้านเอาต์พุต

ในส่วนของการวัดสัญญาณทางด้านเอาต์พุตเราจะอาศัยการวัดสัญญาณที่ได้หลังจากการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอกโดยการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอกนั้นเราจะใช้การต่อ Resister ขนานกันโดยที่เราจัดทำให้เป็นการแปลงแบบ 10 บิตซึ่งสามารถแสดงจุดวัดได้ดังนี้



รูปที่ 4.39 จุดวัดสัญญาณทางด้านเอาต์พุต



รูปที่ 4.40 สัญญาณที่วัดได้จากด้านเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและวิจารณ์

5.1 สรุปผลการทดลอง

5.1.1 ผลการทดลองในการวัดสัญญาณเปรียบเทียบกับระยะทาง

5.1.1.1 ผลการทดลองในการวัดระดับความแรงของสัญญาณระยะทางต่างๆ

จากการทดลองวัดความแรงของสัญญาณแล้วบันทึกค่าจากระยะทางทั้งหมด 100 เมตร พบว่ายิ่งระยะทางไกลขึ้นค่าความแรงของสัญญาณจะลดลงตามระยะทางจากการทดลองเราสามารถเพิ่มระยะทางได้อีกแต่ระดับความแรงของสัญญาณก็จะลดลงไปด้วย

5.1.1.2 ผลการทดลองในการวัดค่าความผิดพลาดเปรียบเทียบกับระยะทาง

จากการทดลองวัดค่าความผิดพลาดของการรับ – ส่งสัญญาณจากระยะทางทั้งหมด 500 เมตร พบว่าเมื่อระยะทางไกลขึ้นค่าความผิดพลาดหรือข้อมูลที่สูญหายก็จะเพิ่มมากขึ้นโดยที่ระยะทางที่สามารถรับ – ส่งข้อมูลได้ผลดีที่สุดอยู่ในระยะ 0 – 300 เมตร ซึ่งตรงตามมาตรฐานของ IEEE 802.15.4

5.1.2 ผลการทดลองในการส่งข้อมูลที่เป็นเสียงพูดแสดงออกทาง X-CTU ทางฝั่งรับ

จากการทดลองรับ – ส่งข้อมูลที่เป็นเสียงพูดโดยที่ข้อมูลถูกแสดงออกทางโปรแกรม X- CTU ซึ่งผลการทดลองพบว่าในขณะที่ยังไม่มีข้อมูลส่งออกไปนั้นฝั่งรับจะมีค่าประมาณ 100 – 200 mV แต่เมื่อมีการพูดส่งออกไปค่าตัวเลขที่แสดงจะเพิ่มขึ้นตามความดังของเสียงที่พูดออกไป

5.1.3 ผลการวัดรูปสัญญาณโดยใช้ ออสซิโลสโคป

5.1.3.1 ผลการวัดรูปสัญญาณจากบอร์ดควบคุม Zigbee

จากการทดลองวัดสัญญาณโดยใช้การรับ – ส่งระหว่างโมดูลสองชุดที่ระยะ10 เมตรพบว่ารูปสัญญาณที่วัดได้ทางฝั่งรับจะเป็นสัญญาณในรูปแบบดิจิตอลที่เรียงบิตข้อมูลจากทางขวาไปซ้ายโดยที่บิตข้อมูลที่รับได้นั้นเป็นไปตามรหัส ASCII ของตัวอักษรที่กดส่งออกไปจากฝั่งส่งซึ่งจากการทดลองส่งข้อมูลเป็นจำนวน 8 ค่าพบว่าฝั่งรับสามารถรับได้ถูกต้องทุกค่าและจากการทดลองเปลี่ยนอัตราเร็วข้อมูลทั้งหมด 8 ระดับพบว่ายิ่งค่าอัตราเร็วบิตข้อมูลสูงขึ้นจะทำให้เวลาในการส่งข้อมูลแต่ละบิตเร็วขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.3.2 ผลการวัดรูปสัญญาณจากบอร์ดควบคุมไมโครคอนโทรลเลอร์

การทดลองวัดสัญญาณต่างๆ เช่น สัญญาณเสียงพูดที่ใช้เป็นอินพุตที่ได้จากวงจรปรีไมค์, สัญญาณที่ได้จากฝั่งรับและฝั่งส่ง เป็นต้น ซึ่งจากการวัดรูปสัญญาณนั้นพบว่าในส่วนของการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลนั้นได้รูปสัญญาณที่เป็นการแปลงอย่างถูกต้อง แต่ในส่วนของการแปลงสัญญาณดิจิตอลกลับมาเป็นสัญญาณอนาลอกนั้นได้สัญญาณที่ถูกต้องแต่มีระดับแรงดันที่ต่ำ และในส่วนของการวัดสัญญาณส่วนอื่นๆก็ได้ผลการวัดที่เป็นไปตามทฤษฎี

5.2 ปัญหาและแนวทางแก้ไข

5.2.1 ปัญหาเกี่ยวกับการออกแบบลายวงจรและการลงอุปกรณ์

ปัญหาในเรื่องของการออกแบบวงจรและการทำลายวงจรนั้นจะเกี่ยวกับการที่ยังมีจุดที่ผิดพลาดอยู่หลายจุดแต่ก็สามารถแก้ไขได้ด้วยการออกแบบและทำลายวงจรใหม่และอีกปัญหาคือการที่เราออกแบบบอร์ดแยกกัน โดยที่ไม่มีวงจรป้องกันการสลับขั้วของแหล่งจ่ายและไม่มีวงจรบริดจ์ช่วยในการแปลงแรงดันเวลาเข้า – ออก โมดูล Zigbee ทำให้โมดูลได้รับไฟเกินและเกิดการกระชากของไฟเราจึงได้ทำการออกแบบบอร์ดใหม่โดยเพิ่มวงจรป้องกันดังกล่าวเข้าไปทำให้ใช้งานได้ตามปกติ

5.2.2 ปัญหาเกี่ยวกับทางด้านเอาต์พุต

เนื่องจากเอาต์พุตที่ออกมานั้นมีเสียงที่ไม่สมบูรณ์และบางจังหวะฟังไม่ออกว่าเป็นเสียงอะไรเราจึงได้ทำการเพิ่มอัตราเร็วการส่งข้อมูลให้สูงที่สุดและเปลี่ยนตัวไมโครคอนโทรลเลอร์จาก AVR เป็น ADuC842 ที่มี ADC และ DAC ขนาด 12 บิตแต่เสียงที่ออกมาก็ไม่ดีไปกว่า AVR มากและราคาของ ADuC842 ก็มีราคาสูงเราจึงได้กลับไปใช้ AVR เช่นเดิมแต่ปรับเปลี่ยนในส่วนของ DAC ใหม่จึงได้คุณภาพเสียงที่ดีขึ้นมาเล็กน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก โปรแกรมควบคุมชุดกระจายเสียงแบบดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รวม Source Code ต่างๆที่ใช้กับชุดระบบกระจายเสียงแบบดิจิตอล

- **Source Code ฝั่งส่ง**

```
/************************************************************** Includes */
#include <stdio.h>                      // Standard Input/Output
#include <avr/io.h>                     // AVR device-specific IO definitions
#include <avr/interrupt.h>              // Interrupt Service routine
#include <util/delay.h>                 // util_delay
#include <compat/deprecated.h>          // Use sbi(), cbi() function
#include "font.h"
#include "LCD_5110.h"
#define  Vadc  1024
#define  F_CPU 8000000UL                // CPU clock frequency (in Hertz)
#define  SW1   (PINB&(1<<PINB0))
unsigned char RX_AT[5];

/********************************************************** Prototype functions */
static int uart_putchar(char c, FILE *stream);
static FILE uart_str = FDEV_SETUP_STREAM(uart_putchar, NULL, _FDEV_SETUP_WRITE);
void Init_Serial(unsigned int baudrate);
static void delay_1s(void);
void delay_ms(unsigned int i);
void set_MY_add(unsigned int add);
void set_dest_add(unsigned int add);
void xbee_init();
unsigned int c;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/********************************************************************** delay_ms */
void delay_ms(unsigned int i)
{
        for (;i>0;i--)
        _delay_ms(1);
}
/********************************************************************** delay_1s */
static void delay_1s(void)
{
  uint8_t i;
  for (i = 0; i < 100; i++)
    _delay_ms(10);
}
/********************************************************************** USART putchar */
static int uart_putchar(char c, FILE *stream)
{
  if (c == '\a')
            {
    fputs("*ring*\n", stderr);
    return 0;
     }
  if (c == '\n')
    uart_putchar('\r', stream);

  loop_until_bit_is_set(UCSRA, UDRE);
  UDR = c;
  return 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/************************************************************ Initialize UART */
static void USART_Init(unsigned int baud)
{
        // Set baud rate
        UBRRH = (unsigned char) (baud>>8);
        UBRRL = (unsigned char) baud;
        // Enable receiver and tramsmitter
        UCSRB = (1<<RXEN) | (1<<TXEN);
         // Enable Interrupt
        // Set frame format: 8data, NoneParity, 1stop bit
        UCSRC = (1<<URSEL)|(3<<UCSZ0);
         // Set address uart_str to stdout
           stdout = &uart_str;
}
/************************************************************ Main Functions */
int main(void)
{
   unsigned char i = 0;
        unsigned int adc,ss;
        DDRA =0b10000000;
        DDRB =0b00111111;                                   // PB[7,6] = Input,PB[5..0] = Output
        DDRC =0xFF;
          DDRD =0xFF;
        PORTC=0x00;
          PORTD=0x00;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*************************************************/
/* Initial GPIO Signal Interface LCD Nokia-5110 */
/*************************************************/
LCD5110_RES_LOW();                          // Active Reset
LCD5110_RES_HIGH();                         // Normal Operation
LCD5110_DC_HIGH();                          // D/C = High("1"=Data)
LCD5110_LED_HIGH();                         // LED = High(ON LED)
LCD5110_SDIN_LOW();          // Standby SPI Data
LCD5110_SCLK_LOW();           // Standby SPI Clock
LCD5110_SCE_LOW();                          // SCE = Low(Enable)

/*************************************************/
/* Start Initial & Display Character to LCD    */
/*************************************************/
lcd_initial();                              // Initial LCD
lcd_clear_screen();                         // Clear Screen Display
USART_Init(12);                             // (51)for9,600 (25)for19200 (12)for38500
ADMUX = (0<<REFS1)|(1<<REFS0);              // AVCC with external capacitor at AREF pin
ADCSRA = (1<<ADEN)|(0<<ADATE);              // ADC Enable & Auto Trigger Disable
ADCSRA |= (0<<ADPS2)|(1<<ADPS1)|(1<<ADPS0);  // XTAL/8
    lcd_gotoxy(0,0);                        // Set Cursor = Line-1
    lcd_print_string("ZBEE communi..",14);  // Display LCD Line-1
    lcd_print_string("Baud 38400 bps",14);  // Display LCD Line-2
    lcd_print_string("Coordinat    ",14);   // Display LCD Line-3
    lcd_print_string("PAN 123      ",14);   // Display LCD Line-4
    lcd_print_string("Channal 2    ",14);   // Display LCD Line-5
  //lcd_print_string("",14);                // Display LCD Line-6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
while(1)
{
   ADCSRA |= (1<<ADSC);                    // ADC Start Conversion
    while (!(ADCSRA &(1<<ADIF)));          // Wait Coversion completes
    {
     adc=ADCW;
     adc =1024-adc;
     i=adc/4;
     UDR=i;
     }
   }
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Source Code ฝั่งรับ**

```
/************************************************************** Includes */
#include <stdio.h>                          // Standard Input/Output
#include <avr/io.h>                         // AVR device-specific IO definitions
#include <avr/interrupt.h>                  // Interrupt Service routine
#include <util/delay.h>                     // util_delay
#include <compat/deprecated.h>              // Use sbi(), cbi() function
#include "font.h"
#include "LCD_5110.h"
#define  Vadc  1024
#define  F_CPU 8000000UL                    // CPU clock frequency (in Hertz)
#define  SW1   (PINB&(1<<PINB0))
unsigned char RX_AT[5];

/******************************************************** Prototype functions */
static int uart_putchar(char c, FILE *stream);
static FILE uart_str = FDEV_SETUP_STREAM(uart_putchar, NULL, _FDEV_SETUP_WRITE);
void Init_Serial(unsigned int baudrate);
static void delay_1s(void);
void delay_ms(unsigned int i);
void set_MY_add(unsigned int add);
void set_dest_add(unsigned int add);
void xbee_init();
unsigned int c;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/********************************************************************* delay_ms */
void delay_ms(unsigned int i)
{
        for (;i>0;i--)
        _delay_ms(1);
}
/********************************************************************* delay_1s */
static void delay_1s(void)
{
   uint8_t i;
   for (i = 0; i < 100; i++)
      _delay_ms(10);
}
/********************************************************************* USART putchar */
static int uart_putchar(char c, FILE *stream)
{
   if (c == '\a')
              {
      fputs("*ring*\n", stderr);
      return 0;
       }
   if (c == '\n')
      uart_putchar('\r', stream);
           loop_until_bit_is_set(UCSRA, UDRE);
           UDR = c;

           return 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/************************************************************ Initialize UART */
static void USART_Init(unsigned int baud)
{
        // Set baud rate
        UBRRH = (unsigned char) (baud>>8);
        UBRRL = (unsigned char) baud;
        // Enable receiver and tramsmitter
        UCSRB = (1<<RXEN) | (1<<TXEN);
        // Set frame format: 8data, NoneParity, 1stop bit
        UCSRC = (1<<URSEL)|(3<<UCSZ0);
         // Set address uart_str to stdout
           stdout = &uart_str;
}
/************************************************************ Main Functions */
int main(void)
{
   unsigned char i = 0;
        unsigned int adc,ss;
        DDRA =0b10000000;
         DDRB =0b00111111;                    // PB[7,6] = Input,PB[5..0] = Output
        DDRC =0xFF;
          DDRD =0xFF;
        PORTC=0x00;
           PORTD=0x00;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*************************************************/
/* Initial GPIO Signal Interface LCD Nokia-5110 */
/*************************************************/
LCD5110_RES_LOW();                          // Active Reset
LCD5110_RES_HIGH();                         // Normal Operation
LCD5110_DC_HIGH();                          // D/C = High("1"=Data)
LCD5110_LED_HIGH();                         // LED = High(ON LED)
LCD5110_SDIN_LOW();          // Standby SPI Data
LCD5110_SCLK_LOW();          // Standby SPI Clock
LCD5110_SCE_LOW();                          // SCE = Low(Enable)
/*************************************************/
/* Start Initial & Display Character to LCD    */
/*************************************************/
lcd_initial();                              // Initial LCD
lcd_clear_screen();                         // Clear Screen Display
USART_Init(12);                             // (51)for9,600 (25)for19200 (12)for38500
ADMUX = (0<<REFS1)|(1<<REFS0);              // AVCC with external capacitor at AREF pin
ADCSRA = (1<<ADEN)|(0<<ADATE);              // ADC Enable & Auto Trigger Disable
ADCSRA |= (0<<ADPS2)|(1<<ADPS1)|(1<<ADPS0);// XTAL/8

   lcd_gotoxy(0,0);                         // Set Cursor = Line-1
   lcd_print_string("ZBEE communi..",14);   // Display LCD Line-1
   lcd_print_string("Baud 38400 bps",14);   // Display LCD Line-2
   lcd_print_string("END Device    ",14);   // Display LCD Line-3
   lcd_print_string("PAN 123       ",14);   // Display LCD Line-4
   lcd_print_string("Channal 2     ",14);   // Display LCD Line-5
  //lcd_print_string("",14);                // Display LCD Line-6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
while(1)
{
 /*   ADCSRA |= (1<<ADSC);                    // ADC Start Conversion
    while (!(ADCSRA &(1<<ADIF)));             // Wait Coversion completes
    {
    adc=ADCW;
    adc =1024-adc;
    i=adc/4;
    UDR=i;*/
          i=UDR;
          ss=i*4;
           PORTD=(ss/255)<<6;
          PORTC=(ss%255);
   }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code ต่างๆที่เรียกใช้

```
// Define LCD Nokia-5110 PinI/O Interface Mask Bit
#define  LCD5110_SCE_HIGH()   PORTB |= 0b00000001          // SCE(PB0) = '1'(Disable)
#define  LCD5110_SCE_LOW()    PORTB &= 0b11111110          // SCE(PB0) = '0'(Enable)
#define  LCD5110_RES_HIGH()   PORTB |= 0b00000010          // RES(PB1) = '1'(Normal)
#define  LCD5110_RES_LOW()    PORTB &= 0b11111101          // RES(PB1) = '0'(Reset)
#define  LCD5110_DC_HIGH()      PORTB |= 0b00000100        // D/C(PB2) = '1'(Data)
#define  LCD5110_DC_LOW()       PORTB &= 0b11111011        // D/C(PB2) = '0'(Command)
#define  LCD5110_SDIN_HIGH()  PORTB |= 0b00001000          // LED(PB3) = '1'(Logic "1")
#define  LCD5110_SDIN_LOW()   PORTB &= 0b11110111          // LED(PB3) = '0'(Logic "0")
#define  LCD5110_SCLK_HIGH() PORTB |= 0b00010000           // LED(PB4) = '1'(Shift Data)
#define  LCD5110_SCLK_LOW()  PORTB &= 0b11101111           // LED(PB4) = '0'(Stand By)
#define  LCD5110_LED_HIGH()   PORTB |= 0b00100000          // LED(PB5) = '1'(LED ON)
#define  LCD5110_LED_LOW()      PORTB &= 0b11011111        // LED(PB5) = '0'(LED OFF)
// End of Define For LCD Nokia-5110

/*************************/
/* Initial LCD Nokia-5110 */
/*************************/
void lcd_initial(void)
{
 LCD5110_RES_LOW();                          // Active Reset
 LCD5110_RES_HIGH();                         // Normal Operation
 lcd_write_command(32+1);                    // Function Set = Extend
Instruction(00100+PD,V,H=00100+0,0,1)
 lcd_write_command(128+38);              // Set VOP(1+VOP[6..0] = 1+0100110)
 lcd_write_command(4+3);                 // Temp Control(000001+TC1,TC0=000001+1,1)
 lcd_write_command(16+3);                // Bias System(00010,BS2,BS1,BS0=00010,0,1,1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 lcd_write_command(32+0);          // Function Set = Basic Instruction(00100+PD,V,H = 00100+0,0,0)
 lcd_write_command(12);            // Display Control = Normal Mode(00001D0E=00001100)
}
/*****************************/
/* Clear Screen Display LCD */
/*****************************/
void lcd_clear_screen(void)
{
 unsigned int  i=0;                                  // Memory Display(Byte) Counter
 lcd_write_command(128+0);                           // Set X Position = 0(0..83)
 lcd_write_command(64+0);                            // Set Y Position = 0(0..5)
 for(i=0;i<504;i++)                                  // All Display RAM = 504 Byte
 lcd_write_data(0);                                  // Clear Screen Display
}
/*********************************/
/* Write Data or Command to LCD */
/* D/C = "0" = Write Command    */
/* D/C = "1" = Write Display    */
/*********************************/
void lcd_write_data(unsigned char DataByte)
{
 unsigned char Bit = 0;                              // Bit Counter
 LCD5110_DC_HIGH();                                  // Active DC = High("1"=Data)
 for (Bit = 0; Bit < 8; Bit++)                       // 8 Bit Write
 {
  if ((DataByte & 0x80) == 0x80)                     // MSB First of Data Bit(7..0)
  {
   LCD5110_SDIN_HIGH();                              // SPI Data = "1"
  }
  else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
   LCD5110_SDIN_LOW();                    // SPI Data = "0"
  }
  LCD5110_SCLK_HIGH();                    // Strobe Bit Data
  SPI_Delay();                            // Delay Clock
  LCD5110_SCLK_LOW();                     // Next Clock
  DataByte <<= 1;                         // Next Bit Data
 }
}
/*********************************/
/* Write Data or Command to LCD */
/* D/C = "0" = Write Command    */
/* D/C = "1" = Write Display    */
/*********************************/
void lcd_write_command(unsigned char CommandByte)
{
 unsigned char Bit = 0;                   // Bit Counter
 LCD5110_DC_LOW();                        // Active DC = Low("0"=Command)
 for (Bit = 0; Bit < 8; Bit++)            // 8 Bit Write
 {
  if ((CommandByte & 0x80) == 0x80)       // MSB First of Data Bit(7..0)
  {
   LCD5110_SDIN_HIGH();                   // SPI Data = "1"
  }
  else
  {
   LCD5110_SDIN_LOW();                    // SPI Data = "0"
  }
  LCD5110_SCLK_HIGH();                    // Strobe Bit Data
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   SPI_Delay();                          // Delay Clock
   LCD5110_SCLK_LOW();                   // Next Clock
   CommandByte <<= 1;                    // Next Bit Data
  }
 }
/*****************************/
/* Set Cursor X,Y Position */
/* X[0-83]: 84 Column Data */
/* Y[0-5] : 6 Row(48 Dot)  */
/*****************************/
void lcd_gotoxy(unsigned char x,unsigned char y)
{
 lcd_write_command(128+x);             // Set X Position(1+x6,x5,x4,x3,x2,x1,x0)
 lcd_write_command(64+y);              // Set Y Position(01000+y2,y1,y0)
}
/*****************************/
/* Delay SPI Clock Signal */
/*****************************/
void SPI_Delay(void)
{
 int x;                                // Short Delay Counter
 x++;
 x++;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/***************************/
/* Put Char to LCD Display */
/***************************/
void lcd_put_char(unsigned char character)
{
 unsigned char font_size_count = 0;                // Font Size Counter
 unsigned int  font_data_index;                    // Font Data Pointer
 font_data_index = character-32;                   // Skip 0x00..0x1F Font Code
 font_data_index = font_data_index*5;              // 5 Byte / Font
 while(font_size_count<5)                          // Get 5 Byte Font & Display on LCD
 {
  lcd_write_data(tab_font[font_data_index]);       // Get Data of Font From Table & Write LCD
  font_size_count++;                               // Next Byte Counter
  font_data_index++;                               // Next   Byte Pointer
 }
 lcd_write_data(0);                                // 1 Pixel Dot Space
}
/*******************************/
/* Print String to LCD Display */
/*******************************/
void lcd_print_string(unsigned char *string , unsigned char CharCount)
{
 unsigned char i=0;                                // Dummy Character Count
 while(i<CharCount)
 {
  lcd_put_char(string[i]);                         // Print 1-Char to LCD
  i++;                                             // Next Character Print
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*******************************/
/* Print char to LCD Display */
/*******************************/
void lcd_print_char(unsigned char data_char)
{
    unsigned char buffer_h=0;
            unsigned char buffer_l=0;
            buffer_h = data_char/10;
            buffer_l = data_char%10;
            lcd_put_char(buffer_h+0x30);
            lcd_put_char(buffer_l+0x30);
}
/*******************************/
/* Print char to LCD Display */
/*******************************/
void lcd_print_int(unsigned int data_int)
{
    unsigned char buffer_0=0;
            unsigned char buffer_1=0;
            unsigned char buffer_2=0;
            unsigned char buffer_3=0;
            unsigned char buffer_4=0;
            buffer_0=data_int%10000;
    buffer_1=(data_int%10000)/1000;
            buffer_2=((data_int%10000)%1000)/100;
    buffer_3=(((data_int%10000)%1000)%100)/10;
    buffer_4=((((data_int%10000)%1000)%100)%10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//          lcd_put_char(buffer_0+0x30);
            lcd_put_char(buffer_1+0x30);
            lcd_put_char(buffer_2+0x30);
            lcd_put_char(buffer_3+0x30);
            lcd_put_char(buffer_4+0x30);
}
/*****************************************/
/* Long Delay Time Function(1..4294967295) */
/*****************************************/
void delay(unsigned long i)
{
  while(i > 0) {i--;}                                    // Loop Decrease
Counter
  return;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code จอโนเกีย

```
/************************************************************************ Includes */
#include <avr/io.h>                          // AVR device-specific IO definitions
#include <compat/deprecated.h>               // Deprecated items
#define F_CPU 8000000UL                      // 8 MHz
#include <util/delay.h>                      // header file implement simple delay loops
#include <stdio.h>                           // Standard Input/Output
#include "font.h"
#include "LCD_5110.h"
#include "keyboard.h"
#include "lib_UART.c"                        // Use Module USART
/************************************************************** Main Functions */
/************************************/
/* ATMEGA64 Interface LCD Nokia-5110 */
/* -> ATMEGA64   --> LCD Nokia-5110  */
/* -> PB0(I/O)   --> SCE(Active "0") */
/* -> PB1(I/O)   --> RES(Active "0") */
/* -> PB2(I/O)   --> D/C("1"=Data    */
/*                   "0"=Command)*/
/* -> PB3(I/O)   --> SDIN            */
/* -> PB4(I/O)   --> SCLK            */
/* -> PB5(I/O)   --> LED(Active "1") */
/************************************/
void main(void)
{
  unsigned char k,j=0;
  DDRB = 0b00111111;                 // PB[7,6] = Input,PB[5..0] = Output
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/***********************************************/
/* Initial GPIO Signal Interface LCD Nokia-5110 */
/***********************************************/
LCD5110_RES_LOW();                          // Active Reset
LCD5110_RES_HIGH();                         // Normal Operation
LCD5110_DC_HIGH();                          // D/C = High("1"=Data)
LCD5110_LED_HIGH();                         // LED = High(ON LED)
LCD5110_SDIN_LOW();                         // Standby SPI Data
LCD5110_SCLK_LOW();                         // Standby SPI Clock
LCD5110_SCE_LOW();                          // SCE = Low(Enable)
/***********************************************/
/* Start Initial & Display Character to LCD    */
/***********************************************/
lcd_initial();                              // Initial LCD
lcd_clear_screen();                         // Clear Screen Display
while(1)
{
  lcd_gotoxy(0,0);                           // Set Cursor = Line-1
  lcd_print_string("BASE AVRMEGA64",14);     // Display LCD Line-1
//  lcd_print_string("BY ETT CO.,LTD",14);    // Display LCD Line-2
//  lcd_print_string("Demo SPI & LCD",14);    // Display LCD Line-3
//  lcd_print_string("LCD Nokia-5110",14);    // Display LCD Line-4
//  lcd_print_string("Demo Character",14);    // Display LCD Line-5
//  lcd_print_string("14Char x 6Line",14);    // Display LCD Line-6
                                            // Display LCD Line-6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                lcd_gotoxy(5,3);
            lcd_print_char(k);
        k++;
                lcd_gotoxy(5,4);
        scankey();
        delay_ms(100);
    /*      lcd_clear_screen();
        lcd_fill_picture();                                     // Display Graphic
        delay_ms(1000);
            lcd_clear_screen(); */
     }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code Uart

```
/*****************************************
* Chip type           : ATmega16
* Clock frequency     : 4,000000 MHz
*****************************************/
#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/signal.h>
#include <inttypes.h>
#include <avr/iom16.h>
#define F_OSC 4000000                                   /* oscillator-frequency in Hz */
#define UART_BAUD_RATE 9600
#define UART_BAUD_CALC(UART_BAUD_RATE,F_OSC)
((F_OSC)/((UART_BAUD_RATE)*16l)-1)

void delay_ms(unsigned short ms) {
        unsigned short outer1, outer2;
        outer1 = 200;
        while (outer1) {
                outer2 = 1000;
                while (outer2) {
                        while ( ms ) ms--;
                        outer2--;
                }
                outer1--;
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void usart_putc(unsigned char c) {
  // wait until UDR ready
        while(!(UCSRA & (1 << UDRE)));
        UDR = c;    // send character
}
void uart_puts (char *s) {
        //  loop until *s != NULL
        while (*s) {
                usart_putc(*s);
                s++;
        }
}
void init(void) {
        // set baud rate
        UBRRH = (uint8_t)(UART_BAUD_CALC(UART_BAUD_RATE,F_OSC)>>8);
        UBRRL = (uint8_t)UART_BAUD_CALC(UART_BAUD_RATE,F_OSC);
        // Enable receiver and transmitter; enable RX interrupt
        UCSRB = (1 << RXEN) | (1 << TXEN) | (1 << RXCIE);
        //asynchronous 8N1
        UCSRC = (1 << URSEL) | (3 << UCSZ0);
}
// INTERRUPT can be interrupted
// SIGNAL can't be interrupted
SIGNAL (SIG_UART_RECV) { // USART RX interrupt
        unsigned char c;
        c = UDR;
        usart_putc(c);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int main(void) {
        init(); // init USART
        sei();  // enable interrupts
        // send initial character
        while(!(UCSRA & (1 << UDRE)));
        UDR = 0x43; // "C"
        while(!(UCSRA & (1 << UDRE)));
        UDR = 0x0d;
        // enable  PD5 as output
        DDRD |= (1<<PD5);
        while (1) {
                // PIN5 PORTD clear -> LED off
                PORTD &= ~(1 << PD5);
                delay_ms(500);
                // PIN5 PORTD set -> LED on
                PORTD |= (1 << PD5);
                delay_ms(500);
        }
        return 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code Font ที่ใช้

```
/***********************************/
/* Font Code Size 5:Byte/Font Table */
/***********************************/
 #ifndef font_h
 #define font_h
 unsigned char tab_font[ ] =
{
        0x00, 0x00, 0x00, 0x00, 0x00,   // sp
         0x00, 0x00, 0x2f, 0x00, 0x00,   // !
         0x00, 0x07, 0x00, 0x07, 0x00,   // "
         0x14, 0x7f, 0x14, 0x7f, 0x14,   // #
         0x24, 0x2a, 0x7f, 0x2a, 0x12,   // $
         0x62, 0x64, 0x08, 0x13, 0x23,   // %
         0x36, 0x49, 0x55, 0x22, 0x50,   // &
         0x00, 0x05, 0x03, 0x00, 0x00,   // '
         0x00, 0x1c, 0x22, 0x41, 0x00,   // (
         0x00, 0x41, 0x22, 0x1c, 0x00,   // )
         0x14, 0x08, 0x3E, 0x08, 0x14,   // *
         0x08, 0x08, 0x3E, 0x08, 0x08,   // +
         0x00, 0x00, 0xA0, 0x60, 0x00,   // ,
         0x08, 0x08, 0x08, 0x08, 0x08,   // -
         0x00, 0x60, 0x60, 0x00, 0x00,   // .
         0x20, 0x10, 0x08, 0x04, 0x02,   // /
         0x3E, 0x51, 0x49, 0x45, 0x3E,   // 0
         0x00, 0x42, 0x7F, 0x40, 0x00,   // 1
         0x42, 0x61, 0x51, 0x49, 0x46,   // 2
         0x21, 0x41, 0x45, 0x4B, 0x31,   // 3
         0x18, 0x14, 0x12, 0x7F, 0x10,   // 4
         0x27, 0x45, 0x45, 0x45, 0x39,   // 5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0x3C, 0x4A, 0x49, 0x49, 0x30,   // 6
0x01, 0x71, 0x09, 0x05, 0x03,   // 7
0x36, 0x49, 0x49, 0x49, 0x36,   // 8
0x06, 0x49, 0x49, 0x29, 0x1E,   // 9
0x00, 0x36, 0x36, 0x00, 0x00,   // :
0x00, 0x56, 0x36, 0x00, 0x00,   // ;
0x08, 0x14, 0x22, 0x41, 0x00,   // <
0x14, 0x14, 0x14, 0x14, 0x14,   // =
0x00, 0x41, 0x22, 0x14, 0x08,   // >
0x02, 0x01, 0x51, 0x09, 0x06,   // ?
0x32, 0x49, 0x59, 0x51, 0x3E,   // @
0x7C, 0x12, 0x11, 0x12, 0x7C,   // A
0x7F, 0x49, 0x49, 0x49, 0x36,   // B
0x3E, 0x41, 0x41, 0x41, 0x22,   // C
0x7F, 0x41, 0x41, 0x22, 0x1C,   // D
0x7F, 0x49, 0x49, 0x49, 0x41,   // E
0x7F, 0x09, 0x09, 0x09, 0x01,   // F
0x3E, 0x41, 0x49, 0x49, 0x7A,   // G
0x7F, 0x08, 0x08, 0x08, 0x7F,   // H
0x00, 0x41, 0x7F, 0x41, 0x00,   // I
0x20, 0x40, 0x41, 0x3F, 0x01,   // J
0x7F, 0x08, 0x14, 0x22, 0x41,   // K
0x7F, 0x40, 0x40, 0x40, 0x40,   // L
0x7F, 0x02, 0x0C, 0x02, 0x7F,   // M
0x7F, 0x04, 0x08, 0x10, 0x7F,   // N
0x3E, 0x41, 0x41, 0x41, 0x3E,   // O
0x7F, 0x09, 0x09, 0x09, 0x06,   // P
0x3E, 0x41, 0x51, 0x21, 0x5E,   // Q
0x7F, 0x09, 0x19, 0x29, 0x46,   // R
0x46, 0x49, 0x49, 0x49, 0x31,   // S
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        0x01, 0x01, 0x7F, 0x01, 0x01,   // T
        0x3F, 0x40, 0x40, 0x40, 0x3F,   // U
        0x1F, 0x20, 0x40, 0x20, 0x1F,   // V
        0x3F, 0x40, 0x38, 0x40, 0x3F,   // W
        0x63, 0x14, 0x08, 0x14, 0x63,   // X
        0x07, 0x08, 0x70, 0x08, 0x07,   // Y
        0x61, 0x51, 0x49, 0x45, 0x43,   // Z
    };
#endif
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข เอกสารเพิ่มเติม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XBee™/XBee-PRO™ OEM RF Modules

XBee/XBee-PRO OEM RF Modules
RF Module Operation
RF Module Configuration
Appendices

Product Manual v1.06
For OEM RF Module Part Numbers: XB24-...-001, XB24-...-002
XBP24-...-001, XBP24-...-002

ZigBee™/IEEE® 802.15.4 OEM RF Modules by MaxStream, Inc.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

MaxStream®
355 South 520 West, Suite 180
Lindon, UT 84042
Phone: (801) 765-9885
Fax: (801) 765-9895

rf-xperts@maxstream.net
www.MaxStream.net (live chat suport)

M100232
2005.10.28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

© 2005 MaxStream, Inc. All rights reserved

No part of the contents of this manual may be transmitted or reproduced in any form or by any means without the written permission of MaxStream, Inc.

XBee™ and XBee-PRO™ are trademarks of MaxStream, Inc.

ZigBee™ is a registered trademark of the ZigBee Alliance.

Technical Support: Phone: (801) 765-9885

Live Chat: www.maxstream.net

E-mail: rf-xperts@maxstream.net

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Contents

1. XBee/XBee-PRO OEM RF Modules 4

1.1. Key Features 4

1.1.1. Worldwide Acceptance 4

1.2. Specifications 5

1.3. Mechanical Drawings 5

1.4. Pin Signals 6

1.5. Electrical Characteristics 6

2. RF Module Operation 7

2.1. Serial Communications 7

2.1.1. UART Data Flow 7

2.1.2. Flow Control 8

2.2. Modes of Operation 9

2.2.1. Idle Mode 9

2.2.2. Transmit & Receive Modes 9

2.2.3. Sleep Mode 11

2.2.4. Command Mode 13

3. RF Module Configuration 14

3.1. Programming the RF Module 14

3.1.1. Programming Examples 14

3.1.2. Command Reference Tables 15

3.2. Command Descriptions 16

Appendix A: Agency Certifications 23

FCC Certification 23

OEM Labeling Requirements 23

FCC Notices 23

FCC-Approved Antennas (2.4 GHz) 24

European Certification (pending) 25

OEM Labeling Requirements 25

Restrictions 25

Declarations of Conformity 25

Appendix B: Development Guide 26

Development Kit Contents 26

Interfacing Options 26

RS-232 Interface Board 27

Physical Interface 27

RS-232 Pin Signals 28

Wiring Diagrams 29

Adapters 30

USB Interface Board 31

Physical Interface 31

USB Pin Signals 31

Appendix C: Additional Information 32

1-Year Warranty 32

Ordering Information 32

Contact MaxStream 33

 © 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. XBee/XBee-PRO OEM RF Modules

XBee and XBee-PRO Modules were engineered to meet ZigBee/IEEE 802.15.4 standards and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of critical data between devices.

The modules operate within the ISM 2.4 GHz frequency band and are pin-for-pin compatible with each other.

1.1. Key Features

High Performance, Low Cost

XBee
- Indoor/Urban: up to 100' (30 m)
- Outdoor line-of-sight: up to 300' (100 m)
- Transmit Power: 1 mW (0 dBm)
- Receiver Sensitivity: -92 dBm

XBee-PRO
- Indoor/Urban: up to 300' (100 m)
- Outdoor line-of-sight: up to 1 mile (1500 m)
- Transmit Power: 100 mW (20 dBm) EIRP
- Receiver Sensitivity: -100 dBm

RF Data Rate: 250,000 bps

Advanced Networking & Security

Retries and Acknowledgements

DSSS (Direct Sequence Spread Spectrum)

Each direct sequence channels has over 65,000 unique network addresses available

Point-to-point, point-to-multipoint and peer-to-peer topologies supported

128-bit Encryption (downloadable firmware version coming soon)

Self-routing/Self-healing mesh networking (downloadable firmware version coming soon)

Low Power

XBee
- TX Current: 45 mA (@3.3 V)
- RX Current: 50 mA (@3.3 V)
- Power-down Current: < 10 µA

XBee-PRO
- TX Current: 270 mA (@3.3 V)
- RX Current: 55 mA (@3.3 V)
- Power-down Current: < 10 µA

Easy-to-Use

No configuration necessary for out-of box RF communications

Free X-CTU Software (Testing and configuration software)

AT Command Mode for simple configuration of module parameters

Small form factor

Network compatible with other ZigBee/802.15.4 devices

Free & Unlimited Technical Support

1.1.1. Worldwide Acceptance

FCC Approval (USA) Refer to Appendix A [p23] for FCC Requirements. Systems that include XBee/XBee-PRO Modules inherit MaxStream's Certifications.

ISM (Industrial, Scientific & Medical) 2.4 GHz frequency band

Manufactured under **ISO 9001:2000** registered standards

XBee/XBee-PRO RF Modules are optimized for use in **US, Canada, Australia, Israel and Europe** (contact MaxStream for complete list of approvals).

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

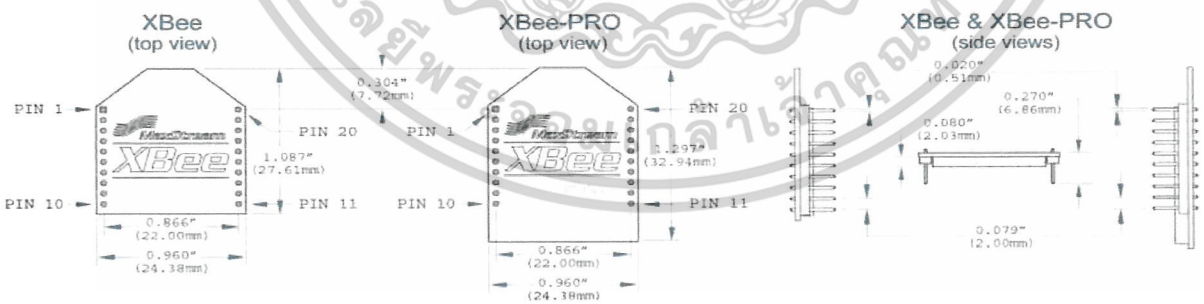
1.2. Specifications

Table 1-01. Specifications of the XBee/XBee-PRO OEM RF Modules

Specification	XBee	XBee-PRO
Performance		
Indoor/Urban Range	up to 100 ft. (30 m)	Up to 300' (100 m)
Outdoor RF line-of-sight Range	up to 300 ft. (100 m)	Up to 1 mile (1500 m)
Transmit Power Output	1mW (0 dBm)	60 mW (18 dBm) conducted, 100 mW (20 dBm) EIRP
RF Data Rate	250,000 bps	250,000 bps
Interface Data Rate (software selectable)	1200 - 115200 bps (non-standard baud rates also supported)	1200 - 115200 bps (non-standard baud rates also supported)
Receiver Sensitivity	-92 dBm (1% packet error rate)	-100 dBm (1% packet error rate)
Power Requirements		
Supply Voltage	2.8 – 3.4 V	2.8 – 3.4 V
Transmit Current (typical)	45 mA (@ 3.3 V)	270 mA (@ 3.3 V)
Receive Current (typical)	50 mA (@ 3.3 V)	55 mA (@ 3.3 V)
Power-down Current	< 10 µA	< 10 µA
General		
Operating Frequency	ISM 2.4 GHz	ISM 2.4 GHz
Dimensions	0.960" x 1.087" (2.438cm x 2.761cm)	0.960" x 1.297" (2.438cm x 3.294cm)
Operating Temperature	-40 to 85° C (industrial)	-40 to 85° C (industrial)
Antenna Options	U.FL Connector, Chip Antenna or Whip Antenna	U.FL Connector, Chip Antenna or Whip Antenna
Networking & Security		
Supported Network Topologies	Point-to-Point, Point-to-Multipoint, Peer-to-Peer and Mesh (coming soon)	Point-to-Point, Point-to-Multipoint, Peer-to-Peer and Mesh (coming soon)
Number of Channels (software selectable)	16 Direct Sequence Channels	13 Direct Sequence Channels
Filtration Options	PAN ID, Channel and Source/Destination Addresses	PAN ID, Channel and Source/Destination Addresses
Agency Approvals		
FCC Part 15.247	OUR-XBEE	pending
Industry Canada (IC)	pending	pending
Europe	pending	pending

1.3. Mechanical Drawings

Figure 1-01. Mechanical drawings of the XBee/XBee-PRO OEM RF Modules (antenna options not shown)
XBee and XBee-PRO RF Modules are pin-for-pin compatible.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4. Pin Signals

Figure 1-02. XBee/XBee-PRO RF Module Pin Number
(top sides shown - shields on bottom)

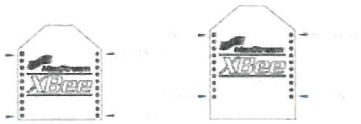


Table 1-02. Pin Assignments for the XBee and XBee-PRO Modules
(Low-asserted signals are distinguished with a horizontal line above signal name.)

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	CD* / DOUT_EN* / DO8*	Output	Carrier Detect, TX_enable or Digital Output 8
5	RESET	Input	Module Reset
6	PWM0 / RSSI	Output	PWM Output 0 or RX Signal Strength Indicator
7	[reserved]	-	Do not connect
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	RF_TX* / AD4* / DIO4*	Either	Transmission Indicator, Analog Input 4 or Digital I/O 4
12	CTS* / DIO7*	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON / SLEEP	Output	Module Status Indicator
14	VREF*	Input	Voltage Reference for A/D Inputs
15	Associate / AD5* / DIO5*	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS* / AD6* / DIO6*	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	COORD_SEL* / AD3* / DIO3*	Either	Analog Input 3, Digital I/O 3 or Coordinator Select
18	AD2* / DIO2*	Either	Analog Input 2 or Digital I/O 2
19	AD1* / DIO1*	Either	Analog Input 1 or Digital I/O 1
20	AD0* / DIO0*	Either	Analog Input 0 or Digital I/O 0

* Functions not supported at the time of this release.

Design Notes:

- Minimum connections are: VCC, GND, DOUT and DIN
- Signal Direction is specified with respect to the module
- Module includes a 50k pull-up resistor attached to RESET
- Unused pins should be left disconnected.

1.5. Electrical Characteristics

Table 1-03. DC Characteristics of the XBee & XBee-PRO (VCC = 2.8 - 3.4 VDC)

Symbol	Parameter	Condition	Min	Typical	Max	Units
V_{IL}	Input Low Voltage	All Digital Inputs	-	-	0.35 * VCC	V
V_{IH}	Input High Voltage	All Digital Inputs	0.7 * VCC	-	-	V
V_{OL}	Output Low Voltage	I_{OL} = 2 mA, VCC >= 2.7 V	-	-	0.5	V
V_{OH}	Output High Voltage	I_{OH} = -2 mA, VCC >= 2.7 V	VCC - 0.5	-	-	V
II_{IN}	Input Leakage Current	V_{IN} = VCC or GND, all inputs, per pin	-	0.025	1	uA
II_{OZ}	High Impedance Leakage Current	V_{IN} = VCC or GND, all I/O High-Z, per pin	-	0.025	1	uA
TX	Transmit Current	VCC = 3.3 V	-	45 (XBee) / 270 (PRO)	-	mA
RX	Receive Current	VCC = 3.3 V	-	50 (XBee) / 55 (PRO)	-	mA
PWR-DWN	Power-down Current	SM parameter = 1	-	< 10	-	uA

© 2005 MaxStream, Inc. Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. RF Module Operation

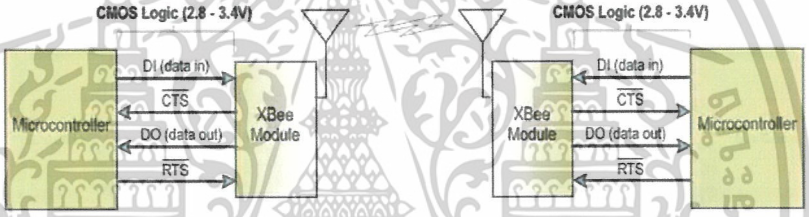
2.1. Serial Communications

The XBee/XBee-PRO OEM RF Modules interface to a host device through a logic-level asynchronous serial port. Through its serial port, the module can communicate with any logic and voltage compatible UART; or through a level translator to any serial device (For example: RS-232/485/422 or USB interface board).

2.1.1. UART Data Flow

Devices that have a UART interface can connect directly to the pins of the RF module as shown in the figure below.

Figure 2-01. Figure 2-01.System Data Flow Diagram in a UART-interfaced environment
(Low-asserted signals distinguished with horizontal line over signal name.)

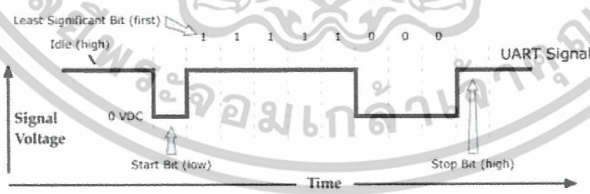


Serial Data

Data enters the module UART through the DI pin (pin 3) as an asynchronous serial signal. The signal should idle high when no data is being transmitted.

Each data byte consists of a start bit (low), 8 data bits (least significant bit first) and a stop bit (high). The following figure illustrates the serial bit pattern of data passing through the module.

Figure 2-02. UART data packet 0x1F (decimal number "31") as transmitted through the RF module
Example Data Format is 8-N-1 (bits - parity - # of stop bits)



The module UART performs tasks, such as timing and parity checking, that are needed for data communications. Serial communications depend on the two UARTs to be configured with compatible settings (baud rate, parity, start bits, stop bits, data bits)

Both the module and host (PC) settings can be viewed and adjusted using MaxStream's proprietary X-CTU Software. Use the "PC Settings" tab to configure host settings. Use the "Terminal" or "RF Module Configuration" tab to configure the module settings.

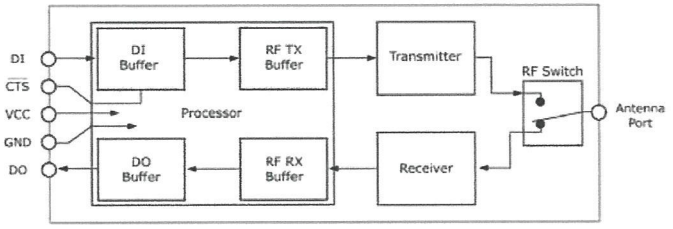
NOTE: Failure to enter AT Command Mode is most commonly due to baud rate mismatch. Ensure the 'Baud' setting on the "PC Settings" tab matches the interface data rate of the RF module (by default, BD parameter = 3 (which is associated to 9600 bps)).

© 2005 MaxStream, Inc. Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2. Flow Control

Figure 2-03. Internal Data Flow Diagram



DI (Data In) Buffer

When serial data enters the RF module through the DI pin (pin 3), the data is stored in the DI Buffer until it can be processed.

Hardware Flow Control (CTS). When the DI buffer is 17 bytes away from being full; by default, the module de-asserts CTS (high) to signal to the host device to stop sending data [refer to D7 (DIO7 Configuration) parameter]. CTS is re-asserted after the DI Buffer has 34 bytes of memory available.

How to eliminate the need for flow control:

1. Send messages that are smaller than the DI buffer size.
2. Interface at a lower baud rate [BD (Interface Data Rate) parameter] than the throughput data rate.

Case in which the DI Buffer may become full and possibly overflow:

If the module is receiving a continuous stream of RF data, any serial data that arrives on the DI pin is placed in the DI Buffer. The data in the DI buffer will be transmitted over-the-air when the module is no longer receiving RF data in the network.

NOTE: CTS hardware flow control is not supported in this release (v1.06). Contact MaxStream support to download firmware that supports this function.

DO (Data Out) Buffer

When RF data is received, the data enters the DO buffer and is sent out the serial port to a host device. Once the DO Buffer reaches capacity, any additional incoming RF data is lost.

Hardware Flow Control (RTS). If RTS is enabled for flow control (D6 (DIO6 Configuration) Parameter = 1), data will not be sent out the DO Buffer as long as RTS (pin 16) is de-asserted.

Two cases in which the DO Buffer may become full and possibly overflow:

1. If the RF data rate is set higher than the interface data rate of the module, the module will receive data from the transmitting module faster than it can send the data to the host.
2. If the host does not allow the module to transmit data out from the DO buffer because of being held off by hardware or software flow control.

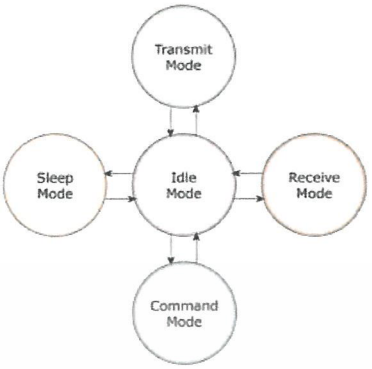
NOTE: RTS hardware flow control is not supported in this release (v1.06). Contact MaxStream support to download firmware that supports this function.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2. Modes of Operation

XBee/XBee-PRO RF Modules operate in five modes.

Figure 2-04. XBee/XBee-PRO RF Module Modes of Operation



2.2.1. Idle Mode

When not receiving or transmitting data, the RF module is in Idle Mode. The RF module shifts into the other modes of operation under the following conditions:

- Transmit Mode: Serial data is received in the DI Buffer
- Receive Mode: Valid RF data is received through the antenna
- Sleep Mode: Sleep Mode condition is met
- Command Mode: Command Mode Sequence is issued

2.2.2. Transmit & Receive Modes

Addressing

When communication occurs between two networked devices, each data packet contains a <Source Address> and a <Destination Address> field. The XBee/XBee-PRO RF Module conforms to the 802.15.4 specification and supports both short 16-bit addresses and long 64-bit addresses. A unique 64-bit IEEE source address is assigned at the factory and can be read with the SL (Serial Number Low) and SH (Serial Number High) parameters. Short addressing must be configured manually. An RF module will use its unique 64-bit address as its Source Address if its MY value is "0xFFFF" or "0xFFFE".

To send a packet to a specific RF module using 64-bit addressing, set the Destination Address (DL + DH) to match the Source Address (SL + SH) of the intended destination RF module. To send a packet to a specific RF module using 16-bit addressing, set the DL (Destination Address Low) parameter to the MY (Source Address) parameter and set the DH (Destination Address High) parameter to "0".

Unicast Mode

Unicast Mode enables acknowledged communications. While in this mode, receiving modules send an ACK (acknowledgement) of RF packet reception to the transmitter. If the transmitting module does not receive the ACK, the transmitter will re-send the packet up to three times until the ACK is received.

Unicast Mode is the only mode that supports retries.

Short 16-bit addresses. The module can be configured to use short 16-bit addresses as the Source Address by setting (MY < 0xFFFE). Setting the DH parameter (DH = 0) will configure the Destination Address to be a short 16-bit address (if DL < 0xFFFE). For two modules to communicate using short addressing, the Destination Address of the transmitter module must match the MY parameter of the receiver.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The following table shows a sample network configuration that would enable Unicast Mode communications using 16-bit short addresses.

Table 2-01. Sample Unicast Configuration (using 16-bit addressing)

Parameter	RF Module 1	RF Module 2
MY (Source Address)	0x01	0x02
DH (Destination Address High)	0	0
DL (Destination Address Low)	0x02	0x01

Long 64-bit addresses. The RF module's serial number (SL parameter concatenated to the SH parameter) can be used as a 64-bit source address when the MY (16-bit Source Address) parameter is disabled. When the MY parameter is disabled (set MY = 0xFFFF or 0xFFFE), the module's source address is set to the 64-bit IEEE address stored in the SH and SL parameters.

When an End Device associates to a Coordinator, its MY parameter is set to 0xFFFE to enable 64-bit addressing. The 64-bit address of the module is stored as SH and SL parameters. To send a packet to a specific module, the Destination Address (DL + DH) on one module must match the Source Address (SL + SH) of the other.

Broadcast Mode

Any RF module will accept a packet that contains a broadcast address. When configured to operate in Broadcast Mode, receiving modules do not send ACKs (Acknowledgements) and transmitting RF modules do not automatically re-send packets as is the case in Unicast Mode.

To send a broadcast packet to all modules regardless of 16-bit or 64-bit addressing, set destination addresses of all the modules as shown below.

Sample Configuration (All modules in the network):

- DL (Destination Low Address) = 0x0000FFFF
- DH (Destination High Address) = 0x00000000

NOTE: When programming the module, parameters are entered in hexadecimal notation (without the "0x" prefix). Leading zeros may be omitted.

 © 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3. Sleep Mode

Sleep Modes enable the RF module to enter states of low-power consumption when not in use. In order to enter Sleep Mode, one of the following conditions must be met (in addition to the module having a non-zero SM parameter value):

- Sleep_RQ (pin 9) is asserted.
- The module is idle (no data transmission or reception) for the amount of time defined by the ST (Time before Sleep) parameter. [NOTE: ST is only active when SM = 4-5.]

Table 2-02. Sleep Mode Configurations

Sleep Mode Setting	Transition into Sleep Mode	Transition out of Sleep Mode (wake)	Characteristics	Related Commands	Power Consumption
Pin Hibernate (SM = 1)	Assert (high) Sleep_RQ (pin 9)	De-assert (low) Sleep_RQ	Pin/Host-controlled / NonBeacon systems only / Lowest Power	(SM)	< 10 µA (@3.0 VCC)
Pin Doze (SM = 2)	Assert (high) Sleep_RQ (pin 9	De-assert (low) Sleep_RQ	Pin/Host-controlled / NonBeacon systems only / Fastest Wake-up	(SM)	< 50 µA
Cyclic Sleep (SM = 4 - 5)	Automatic transition to Sleep Mode as defined by the SM (Sleep Mode) and ST (Time before Sleep) parameters.	Transition occurs after the cyclic sleep time interval elapses. The time interval is defined by the SP (Cyclic Sleep Period) parameter.	RF Module wakes in pre-determined time intervals to detect if RF data is present / When SM = 5, NonBeacon systems only	(SM), SP, ST	< 50 µA when sleeping

The SM command is central to setting Sleep Mode configurations. By default, Sleep Modes are disabled (SM = 0) and the module remains in Idle/Receive Mode. When in this state, the module is constantly ready to respond to serial or RF activity.

Pin/Host-controlled Sleep Modes

Pin Hibernate (SM = 1)

- Pin/Host-controlled
- Typical power-down current: < 10 µA (@3.0 VCC)
- Wake-up time: 13.2 msec

Pin Hibernate Mode minimizes quiescent power (power consumed when in a state of rest or inactivity). This mode is voltage level-activated; when Sleep_RQ is asserted, the module will finish any transmit, receive or association activities, enter Idle Mode and then enter a state of sleep. The module will not respond to either serial or RF activity while in pin sleep.

To wake a sleeping module operating in Pin Hibernate Mode, de-assert Sleep_RQ (pin 9). The module will wake when Sleep_RQ is de-asserted and is ready to transmit or receive when the $\overline{\text{CTS}}$ line is low.

Pin Doze (SM = 2)

- Pin/Host-controlled
- Typical power-down current: < 50 µA
- Wake-up time: 2 msec

Pin Doze Mode functions as does Pin Hibernate Mode; however, Pin Doze features faster wake-up time and higher power consumption.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Cyclic Sleep Modes

Cyclic Sleep Remote (SM = 4)

- Typical Power-down Current: < 50 µA (when asleep)
- Wake-up time: 2 msec

The Cyclic Sleep Modes allow modules to periodically check for RF data. When the SM parameter is set to '4', the module is configured to sleep, then wakes once a cycle to check for data from a module configured as a Cyclic Sleep Coordinator (SM = 6). The Cyclic Sleep Remote sends a poll request to the coordinator at a specific interval set by the SP (Cyclic Sleep Period) parameter. The coordinator will transmit any queued data addressed to that specific remote upon receiving the poll request. If no data is queued for the remote, the coordinator will not transmit and the remote will return to sleep for another cycle. If queued data is transmitted back to the remote, it will stay awake to allow for back and forth communication until the ST (Time before Sleep) timer expires.

Also note that $\overline{\text{CTS}}$ will go low each time the remote wakes, allowing for communication initiated by the remote host if desired.

Cyclic Sleep Remote with Pin Wake-up (SM = 5)

Use this mode to wake a sleeping remote module through either the RF interface or by the de-assertion of Sleep_RQ for event-driven communications. The cyclic sleep mode works as described above (Cyclic Sleep Remote) with the addition of a pin-controlled wake-up at the remote module. The module will wake quickly when a low is detected and set $\overline{\text{CTS}}$ low as soon as it is ready to transmit or receive. Any activity will reset the ST (Time before Sleep) timer so the module will go back to sleep only after Sleep_RQ is asserted and there is no activity for the duration of the timer.

 © 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4. Command Mode

To modify or read RF Module parameters, the module must first enter into Command Mode - a state in which incoming characters are interpreted as commands. Two command modes are supported: AT Command Mode and ATI Command Mode.

A robust set of AT Commands is available for programming and customizing the module.

AT Command Mode

To Enter AT Command Mode:

Send the 3-character command sequence "+++" and observe guard times before and after the command characters. [Refer to the "Default AT Command Mode Sequence" below.]

Default AT Command Mode Sequence (for transition to Command Mode):

- No characters sent for one second [GT (Guard Times) parameter = 0x3E8]
- Input three plus characters ("+++") within one second [CC (Command Sequence Character) parameter = 0x2B.]
- No characters sent for one second [GT (Guard Times) parameter = 0x3E8]

All of the parameter values in the sequence can be modified to reflect user preferences.

To Send AT Commands:

Send AT commands and parameters using the syntax shown below.

Figure 2-05. Syntax for sending AT Commands

To read a parameter value stored in the RF module's register, leave the parameter field blank.

The preceding example would change the RF module Destination Address (Low) to "0x1F". To store the new value to non-volatile (long term) memory, subsequently send the WR (Write) command.

For modified parameter values to persist in the module's registry, changes must be saved to non-volatile memory using the WR (Write) Command. Otherwise, parameters are restored to previously saved values after the module is powered off and then on again (or re-booted).

System Response. When a command is sent to the RF module, the module will parse and execute the command. Upon successful execution of a command, the module returns an "OK" message. If execution of a command results in an error, the module returns an "ERROR" message.

To Exit AT Command Mode:

1. Send ATCN (Exit Command Mode) Command.

   [OR]

2. If no valid AT Commands are received within the time specified by CT (Command Mode Timeout) Command, the RF module automatically returns to Idle Mode.

For an example of programming the RF module using AT Commands and descriptions of each configurable parameter, refer to the "RF Module Configuration" chapter [p14].

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. RF Module Configuration

3.1. Programming the RF Module

Refer to the "Command Mode" section [p13] for more information about entering Command Mode, sending AT commands and exiting Command Mode.

3.1.1. Programming Examples

Setup

The programming examples in this section require the installation of MaxStream's X-CTU Software and a serial connection to a PC. (MaxStream stocks RS-232 and USB boards to facilitate interfacing to a PC.)

1. Install MaxStream's X-CTU Software to a PC by double-clicking the "setup_X-CTU.exe" file. (The file is located on the MaxStream CD and under the 'Software' section of the following web page: www.maxstream.net/helpdesk/download.php)
2. Mount the RF module to an interface board, then connect the module assembly to a PC.
3. Launch the X-CTU Software and select the 'PC Settings' tab. Verify the baud and parity settings of the Com Port match those of the RF module.

NOTE: Failure to enter AT Command Mode is most commonly due to baud rate mismatch. Ensure the 'Baud' setting on the 'PC Settings' tab matches the interface data rate of the RF module (by default, BD parameter = 3 (which corresponds to 9600 bps)).

Sample Configuration: Modify RF Module Destination Address

Example: Utilize the 'Terminal' tab of the X-CTU Software to change the RF module's DL (Destination Address Low) parameter and save the new address to non-volatile memory.

After establishing a serial connection between the RF module and a PC [refer to the 'Setup' section above], select the 'Terminal' tab of the X-CTU Software and enter the following command lines ('CR' stands for carriage return):

Method 1 (One line per command)

Send AT Command	System Response
+++	OK <CR> (Enter into Command Mode)
ATDL <Enter>	{current value} <CR> (Read Destination Address Low)
ATDL1A0D <Enter>	OK <CR> (Modify Destination Address Low)
ATWR <Enter>	OK <CR> (Write to non-volatile memory)
ATCN <Enter>	OK <CR> (Exit Command Mode)

Method 2 (Multiple commands on one line)

Send AT Command	System Response
+++	OK <CR> (Enter into Command Mode)
ATDL <Enter>	{current value} <CR> (Read Destination Address Low)
ATDL1A0D,WR,CN <Enter>	OK <CR> (Execute commands)

Sample Configuration: Restore RF Module Defaults

Example: Utilize the 'Modem Configuration' tab of the X-CTU Software to restore default parameter values of the RF module.

After establishing a connection between the RF module and a PC [refer to the 'Setup' section above], select the 'Modem Configuration' tab of the X-CTU Software.

1. Select the 'Read' button.
2. Select the 'Restore' button.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2. Command Reference Tables

Table 3-01. XBee/XBee-PRO Commands (RF modules expect numerical values in hexadecimal. Hexadecimal values are designated by the "0x" prefix. Decimal equivalents are designated by the "d" suffix.)

AT Command	Command Category	Name and Description	Parameter Range	Default
BD	Serial Interfacing	**Interface Data Rate.** Set/Read the serial interface data rate for communications between the RF module serial port and host.	0 - 7 (custom rates also supported)	3
CC	AT Command Mode Options	**Command Sequence Character.** Set/Read the ASCII character value to be used between Guard Times of the AT Command Mode Sequence (GT+CC+GT). The AT Command Mode Sequence enters the RF module to AT Command Mode.	0 - 0xFF	0x2B ('+' ASCII)
CH	Networking & Security	**Channel.** Set/Read the channel number used for transmitting and receiving between RF modules. Uses 802.15.4 protocol channel numbers.	0x0B - 0x1A (XBee) 0x0C - 0x18 (XBee-PRO)	0x0C (12d)
CN	AT Command Mode Options	**Exit Command Mode.** Explicitly exit AT Command Mode.	-	-
CT	AT Command Mode Options	**Command Mode Timeout.** Set/Read the period of inactivity (no valid commands received) after which the RF module automatically exits AT Command Mode and returns to Idle Mode.	2 - 0xFFFF [x 100 ms]	0x64 (100d)
DB	Diagnostics	**Received Signal Strength.** Read signal level [in dB] of last good packet received (RSSI). Absolute value is reported. (For example: 0x58 = -88 dBm) Reported value is accurate between -40 dBm and RX sensitivity.	0 - 0x64 [read-only]	-
DH	Networking & Security	**Destination Address High.** Set/Read the upper 32 bits of the 64-bit destination address. When combined with DL, it defines the destination address used for transmission. To transmit using a 16-bit address, set DH parameter to zero and DL less than 0xFFFF. 0x000000000000FFFF is the broadcast address for the PAN.	0 - 0xFFFFFFFF	0
DL	Networking & Security	**Destination Address Low.** Set/Read the lower 32 bits of the 64-bit destination address. When combined with DH, DL defines the destination address used for transmission. To transmit using a 16-bit address, set DH parameter to zero and DL less than 0xFFFF. 0x000000000000FFFF is the broadcast address for the PAN.	0 - 0xFFFFFFFF	0
GT	AT Command Mode Options	**Guard Times.** Set required period of silence before and after the Command Sequence Characters of the AT Command Mode Sequence (GT+ CC + GT). The period of silence is used to prevent inadvertent entrance into AT Command Mode.	0x02 - 0xFFFF [x 1 ms]	0x3E8 (1000d)
ID	Networking & Security	**PAN ID.** Set/Read the PAN (Personal Area Network) ID. 0xFFFF indicates a message for all PANs.	0xFFFF	0x3332 (13106d)
MY	Networking & Security	**16-bit Source Address.** Set/Read the RF module 16-bit source address. Set MY = 0xFFFF to disable reception of packets with 16-bit addresses. 64-bit source address (serial number) and broadcast address (0x000000000000FFFF) is always enabled.	0 - 0xFFFF	0
P0	Diagnostics	**PWM0 Configurations.** Select/Read function for PWM0.	0 - 1	1
PL	RF Interfacing	**Power Level.** Select/Read power level at which the RF module transmits.	0 - 4	4
RE	(Special)	**Restore Defaults.** Restore RF module parameters to factory defaults. Follow with WR command to save values to non-volatile memory.	-	-
RN	Networking & Security	**Random Delay Slots.** Set/Read the minimum value of the back-off exponent in the CSMA-CA algorithm that is used for collision avoidance. If RN = 0, collision avoidance is disabled during the first iteration of the algorithm (802.15.4 - macMinBE).	0 - 3	0
RO	Serial Interfacing	**Packetization Timeout.** Set/Read number of character times of inter-character delay required before transmission. Set to zero to transmit characters as they arrive instead of buffering them into one RF packet.	0 - 0xFF [x character times]	3
RP	Diagnostics	**RSSI PWM Timer.** Enable a PWM (pulse width modulation) output (on pin 3 of the RF modules) which shows RX signal strength.	0 - 0xFF [x 100 ms]	0x28 (40d)
SH	Diagnostics	**Serial Number High.** Read high 32 bits of the RF module's unique IEEE 64-bit address. 64-bit source address is always enabled.	0 - 0xFFFFFFFF [read-only]	Factory-set
SL	Diagnostics	**Serial Number Low.** Read low 32 bits of the RF module's unique IEEE 64-bit address. 64-bit source address is always enabled.	0 - 0xFFFFFFFF [read-only]	Factory-set
SM	Sleep (Low Power)	**Sleep Mode.** Set/Read Sleep Mode configurations.	0 - 5	0
SP	Sleep (Low Power)	**Cyclic Sleep Period.** Set/Read sleep period for cyclic sleeping remotes. Maximum sleep period is 268 seconds (0x68B0).	0x01 - 0x68B0 [x 10 ms]	0x64 (100d)
ST	Sleep (Low Power)	**Time before Sleep.** Set/Read time period of inactivity (no serial or RF data is sent or received) before activating Sleep Mode. The ST parameter is only valid with Cyclic Sleep settings (SM = 4 - 6). Set ST on Cyclic Sleep Coordinator to match Cyclic Sleep Remotes.	0x01 - 0xFFFF [x 1 ms]	0x1388 (5000d)
VR	Diagnostics	**Firmware Version.** Read firmware version of the RF module.	0 - 0xFFFF [read-only]	Factory-set
WR	(Special)	**Write.** Write parameter values to RF module's non-volatile memory so that modifications persist through subsequent power-up or reset.	-	-

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2. Command Descriptions

Command descriptions in this section are listed alphabetically. Command categories are designated within "< >" symbols that follow each command title. XBee-PRO RF modules expect parameter values in hexadecimal (designated by the "0x" prefix).

BD (Interface Data Rate) Command

<Serial Interfacing> The BD command is used to set and read the serial interface data rate (baud rate) used between the RF module and host. This parameter determines the rate at which serial data is sent to the RF module from the host. Modified interface data rates do not take effect until the CN (Exit AT Command Mode) command is issued and the system returns the 'OK' response.

When parameters 0-7 are sent to the RF module, the respective interface data rates are used (as shown in the table on the right).

The RF data rate is not affected by the BD parameter. If the interface data rate is set higher than the RF data rate, a flow control configuration may need to be implemented.

AT Command: ATBD

Parameter Range: 0 – 7 (standard rates)

Parameter	Configuration (bps)
0	1200
1	2400
2	4800
3	9600
4	19200
5	38400
6	57600
7	115200

Default Parameter Value:3

Non-standard Interface Data Rates:
When parameter values outside the range of standard baud rates are sent, the closest interface data rate represented by the number is stored in the BD register. For example, a rate of 19200 bps can be set by sending the following command line "ATBD4B00". NOTE: When using MaxStream's X-CTU Software, non-standard interface data rates can only be set and read using the X-CTU 'Terminal' tab. Non-standard rates are not accessible through the 'Modem Configuration' tab.

When the BD command is sent with a non-standard interface data rate, the UART will adjust to accommodate the requested interface rate. In most cases, the clock resolution will cause the stored BD parameter to vary from the parameter that was sent (refer to the table below). Reading the BD command (send "ATBD" command without an associated parameter value) will return the value that was actually stored to the BD register.

Table 3-02. Parameters Sent Versus Parameters Stored

BD Parameter Sent (HEX)	Interface Data Rate (bps)	BD Parameter Stored (HEX)
0	1200	0
4	19,200	4
7	115,200	7
12C	300	12B
1C200	115,200	1B207

CC (Command Sequence Character) Command

<AT Command Mode Options> The CC command is used to set and read the ASCII character used between guard times of the AT Command Mode Sequence (GT + CC + GT). This sequence enters the RF module into AT Command Mode so that data entering the modem from the host is recognized as commands instead of payload.

AT Command: ATCC

Parameter Range: 0 – 0xFF

Default Parameter Value: 0x2B (ASCII "+")

Related Commands: GT (Guard Times)

Refer to the Command Mode section [p13] for more information regarding the AT Command Mode Sequence.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CH (Channel) Command

<Networking {Addressing}> The CH command is used to set and read the channel on which RF connections are made between RF modules. The channel is one of three filtration layers available to the RF module. The other layers are the PAN ID (ID command) and destination addresses (DL & DH commands).

AT Command: ATCH

Parameter Range: 0x0B – 0x1A (XBee)
0x0C – 0x18 (XBee-PRO)

Default Parameter Value: 0x0C (12 decimal)

Related Commands: ID (PAN ID), DL (Destination Address Low, DH (Destination Address High)

In order for RF modules to communicate with each other, the RF modules must share the same channel number. Different channels can be used to prevent RF modules in one network from listening to transmissions of another.

The RF module uses channel numbers of the 802.15.4 standard.

*Center Frequency = 2.405 + (CH - 11d) * 5 MHz (d = decimal)*

Refer to the "Addressing" section [p9] for more information.

CN (Exit AT Command Mode) Command

<AT Command Mode Options> The CN command is used to explicitly exit the RF module from AT Command Mode.

AT Command: ATCN

CT (Command Mode Timeout) Command

<AT Command Mode Options> The CT command is used to set and read the amount of inactive time that elapses before the RF module automatically exits from AT Command Mode and returns to Idle Mode.

Use the CN (Exit AT Command Mode) command to exit AT Command Mode manually.

AT Command: ATCT

Parameter Range: 2 – 0xFFFF
[x 100 milliseconds]

Default Parameter Value: 0x64 (100 decimal, which equals 10 decimal seconds)

Number of bytes returned: 2

Related Command: CN (Exit AT Command Mode)

DB (Received Signal Strength) Command

<Diagnostics> DB parameter is used to read the received signal strength (in dBm) of the last RF packet received. Reported values are accurate between -40 dBm and the RF module's receiver sensitivity.

AT Command: ATDB

Parameter Range: 0 – 0x64 [read-only]

Absolute values are reported. For example: 0x58 = -88 dBm (decimal). If no packets have been received (since last reset, power cycle or sleep event), "0" will be reported.

DH (Destination Address High) Command

<Networking {Addressing}> The DH command is used to set and read the upper 32 bits of the RF module's 64-bit destination address. When combined with the DL (Destination Address Low) parameter, it defines the destination address used for transmission.

AT Command: ATDH

Parameter Range: 0 – 0xFFFFFFFF

Default Parameter Value: 0

Related Commands: DL (Destination Address Low), CH (Channel), ID (PAN VID), MY (Source Address)

An RF module will only communicate with other RF modules having the same channel (CH parameter), PAN ID (ID parameter) and destination address (DH + DL parameters).

To transmit using a 16-bit address, set the DH parameter to zero and the DL parameter less than 0xFFFF. 0x000000000000FFFF (DL concatenated to DH) is the broadcast address for the PAN.

Refer to the "Addressing" section [p9] for more information.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DL (Destination Address Low) Command

<Networking {Addressing}> The DL command is used to set and read the lower 32 bits of the RF module's 64-bit destination address. When combined with the DH (Destination Address High) parameter, it defines the destination address used for transmission.

AT Command: ATDL

Parameter Range: 0 - 0xFFFFFFFF

Default Parameter Value: 0

Related Commands: DH (Destination Address High), CH (Channel), ID (PAN VID), MY (Source Address)

An RF module will only communicate with other RF modules having the same channel (CH parameter), PAN ID (ID parameter) and destination address (DH + DL parameters).

To transmit using a 16-bit address, set the DH parameter to zero and the DL parameter less than 0xFFFF. 0x000000000000FFFF (DL concatenated to DH) is the broadcast address for the PAN.

Refer to the "Addressing" section [p9] for more information.

GT (Guard Times) Command

<AT Command Mode Options> GT Command is used to set the DI (data in from host) time-of-silence that surrounds the AT command sequence character (CC Command) of the AT Command Mode sequence (GT + CC + GT).

AT Command: ATGT

Parameter Range: 2 – 0xFFFF [x 1 millisecond]

Default Parameter Value: 0x3E8 (1000 decimal)

Related Command: CC (Command Sequence Character)

The DI time-of-silence is used to prevent inadvertent entrance into AT Command Mode.

Refer to the Command Mode section [p13] for more information regarding the AT Command Mode Sequence.

ID (Pan ID) Command

<Networking {Addressing}> The ID command is used to set and read the PAN (Personal Area Network) ID of the RF module. Only RF modules with matching PAN IDs can communicate with each other. RF modems with non-matching PAN IDs will not receive unintended data transmission.

AT Command: ATID

Parameter Range: 0 – 0xFFFF

Default Parameter Value: 0x3332 (13106 decimal)

Setting the ID parameter to 0xFFFF indicates a global message for all PANs.

Refer to the "Addressing" section [p9] for more information.

MY (16-bit Source Address) Command

<Networking {Addressing}> The MY command is used to set and read the 16-bit source address of the RF module.

AT Command: ATMY

Parameter Range: 0 – 0xFFFF

Default Parameter Value: 0

Related Commands: DH (Destination Address High), DL (Destination Address Low), CH (Channel), ID (PAN ID)

By setting MY to 0xFFFF, the reception of RF packets having a 16-bit address is disabled. The 64-bit address is the module serial number and is always enabled.

Refer to the "Addressing" section [p9] for more information.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P0 (PWM0 Configuration) Command

<Diagnostics> The P0 command is used to select and read the function for PWM0 (Pulse Width Modulation output 0 - pin 6).

Note: The second character in the command is a zero ("0"), not the letter "O".

AT Command: ATP0

Parameter Range: 0 – 1

Parameter	Configuration
0	Disabled
1	RSSI PWM0 enabled

Default Parameter Value: 1

PL (Power Level) Command

<RF Interfacing> The PL command is used to select and read the power level at which the RF module transmits conducted power.

AT Command: ATPL

Parameter Range: 0 – 4

Parameter	XBee	XBee-Pro
0	-10 dBm	10 dBm
1	-6 dBm	12 dBm
2	-4 dBm	14 dBm
3	-2 dBm	16 dBm
4	0 dBm	18 dBm

Default Parameter Value: 4

RE (Restore Defaults) Command

<(Special)> The RE command is used to restore all configurable parameters to their factory default settings. The RE command does not write restored values to non-volatile (persistent) memory. Issue the WR (Write) command subsequent to issuing the RE command to save restored parameter values to non-volatile memory.

AT Command: ATRE

RN (Random Delay Slots) Command

<Networking & Security> The RN command is used to set and read the minimum value of the back-off exponent in the CSMA-CA algorithm. The CSMA-CA algorithm was engineered for collision avoidance (random delays are inserted to prevent data loss caused by data collisions).

AT Command: ATRN

Parameter Range: 0 – 3 [exponent]

Default Parameter Value: 0

If RN = 0, collision avoidance is disabled during the first iteration of the algorithm (802.15.4 - macMinBE).

CSMA-CA stands for "Carrier Sense Multiple Access - Collision Avoidance". Unlike CSMA-CD (reacts to network transmissions after collisions have been detected), CSMA-CA acts to prevent data collisions before they occur. As soon as a modem receives a packet that is to be transmitted, it checks if the channel is clear (no other modem is transmitting). If the channel is clear, the packet is sent over-the-air. If the channel is not clear, the RF module waits for a randomly selected period of time, then checks again to see if the channel is clear. After a time, the process ends and the data is lost.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RO (Packetization Timeout) Command

<Serial Interfacing> RO command is used to set and read the number of character times of inter-character delay required before transmission.

RF transmission commences when data is detected in the DI (data in from host) buffer and RO character times of silence are detected on the UART receive lines (after receiving at least 1 byte).

AT Command: ATRO

Parameter Range: 0 – 0xFF [x character times]

Default Parameter Value: 3

RF transmission will also commence after 100 bytes (maximum packet size) are received in the DI buffer.

Set the RO parameter to '0' to transmit characters as they arrive instead of buffering them into one RF packet.

RP (RSSI PWM Timer) Command

<Diagnostics> The RP command is used to enable PWM (Pulse Width Modulation) output on the RF module. The output is calibrated to show the level a received RF signal is above the sensitivity level of the RF module. The PWM pulses vary from zero to 95 percent. Zero to twenty-nine percent means the received RF signal is at or below the published sensitivity level of the RF module. The following table shows levels above sensitivity and PWM values.

AT Command: ATRP

Parameter Range: 0 – 0xFF [x 100 milliseconds]

Default Parameter Value: 0x28 (40 decimal)

The total period of the PWM output is 8.32 ms. Because there are 40 steps in the PWM output, the minimum step size is 0.208 ms.

Table 3-03. PWM Percentages

dB above Sensitivity	PWM percentage* (high period / total period)
10	46.0%
20	63.0%
30	80.1%

* PWM% = (295 + (17.5 * dBm above sensitivity)) / 10.24

A non-zero value defines the time that the PWM output will be active with the RSSI value of the last received RF packet. After the set time when no RF packets are received, the PWM output will be set low (0 percent PWM) until another RF packet is received. The PWM output will also be set low at power-up until the first RF packet is received. A parameter value of 0xFF permanently enables the PWM output and it will always reflect the value of the last received RF packet.

SH (Serial Number High) Command

<Diagnostics> The SH command is used to read the high 32 bits of the RF module's unique IEEE 64-bit address.

The RF module serial number is set at the factory and is read-only.

AT Command: ATSH

Parameter Range: 0 – 0xFFFFFFFF [read-only]

Related Commands: SL (Serial Number Low), MY (Source Address)

SL (Serial Number Low) Command

<Diagnostics> The SL command is used to read the low 32 bits of the RF module's unique IEEE 64-bit address.

The RF module serial number is set at the factory and is read-only.

AT Command: ATSL

Parameter Range: 0 – 0xFFFFFFFF [read-only]

Related Commands: SH (Serial Number High), MY (Source Address)

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SM (Sleep Mode) Command

<Sleep Mode (Low Power)> The SM command is used to set and read Sleep Mode settings. By default, Sleep Modes are disabled (SM = 0) and the RF module remains in Idle/Receive Mode. When in this state, the RF module is constantly ready to respond to either serial or RF activity.

SM command options vary according to the networking system type. By default, the module is configured to operate in a NonBeacon system.

AT Command: ATSM

Parameter Range: 0 – 5

Parameter	Configuration
0	Disabled
1	Pin Hibernate
2	Pin Doze
3	(reserved)
4	Cyclic Sleep Remote
5	Cyclic Sleep Remote (with Pin Wake-up)

Default Parameter Value: 0

Related Commands: SP (Cyclic Sleep Period), ST (Time before Sleep)

SP (Cyclic Sleep Period) Command

<Sleep Mode (Low Power)> The SP command is used to set and read the duration of time in which a remote RF module sleeps. After the cyclic sleep period is over, the RF module wakes and checks for data. If data is not present, the RF module goes back to sleep. The maximum sleep period is 268 seconds (SP = 0x68B0).

The SP parameter is only valid if the RF module is configured to operate in Cyclic Sleep (SM = 4-6).

AT Command: ATSP

Parameter Range: 1 – 0x68B0 [x 10 milliseconds]

Default Parameter Value: 0x64 (100d)

Related Commands: SM (Sleep Mode), ST (Time before Sleep)

ST (Time before Sleep) Command

<Sleep Mode (Low Power)> The ST command is used to set and read the period of time that the RF module remains inactive (no transmitting or receiving) before entering into Sleep Mode.

For example, if the ST parameter is set to its default value of 0x1388 (5000 decimal), the RF module will enter into Sleep mode after 5 seconds of inactivity. This command can only be used if Cyclic Sleep settings have been selected using SM (Sleep Mode) Command (SM = 4-6).

AT Command: ATST

Parameter Range: 1 – 0xFFFF [x 1 millisecond]

Default Parameter Value: 0x1388 (5000 decimal)

Related Commands: SM (Sleep Mode), SP (Cyclic Sleep Period)

NOTE: The GT parameter value must always be less than the ST value. (If GT > ST, the configuration will render the module unable to enter into command mode.) If the ST parameter is modified, also modify the GT parameter accordingly.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VR (Firmware Version) Command

<Diagnostics> The VR command is used to read which firmware version is stored in the RF module.

AT Command: ATVR

Parameter Range: 0 – 0xFFFF [read only]

WR (Write) Command

<(Special)> The WR command is used to write configurable parameters to the RF module's non-volatile memory (Parameter values remain in RF module's memory until overwritten by subsequent use of the WR Command).

AT Command: ATWR

If changes are made without writing them to non-volatile memory, the RF module reverts back to previously saved parameters the next time the RF module is powered-on.

NOTE: Once the WR command is sent to the RF module, no additional characters should be sent until after the "OK/r" response is received.

 © 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix A: Agency Certifications

FCC Certification

The XBee/XBee-PRO RF Module complies with Part 15 of the FCC rules and regulations. Compliance with the labeling requirements, FCC notices and antenna usage guidelines is required.

To fulfill FCC Certification requirements, the OEM must comply with the following regulations:

1. The system integrator must ensure that the text on the external label provided with this device is placed on the outside of the final product [Figure A-01].
2. The XBee/XBee-PRO RF Module may be used only with approved antennas that have been tested with this modem.

OEM Labeling Requirements

WARNING: The Original Equipment Manufacturer (OEM) must ensure that FCC labeling requirements are met. This includes a clearly visible label on the outside of the final product enclosure that displays the contents shown in the figure below.

Figure A-01. Required FCC Label for OEM products containing the XBee/XBee-PRO RF Module

Contains FCC ID: OUR-XBEE*

The enclosed device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference and (2) this device must accept any interference received, including interference that may cause undesired operation.

* The FCC ID for the XBee is "OUR-XBEE". The FCC certification for the XBee-PRO is pending.

FCC Notices

IMPORTANT: The XBee/XBee-PRO OEM RF Module has been certified by the FCC for use with other products without any further certification (as per FCC section 2.1091). Modifications not expressly approved by MaxStream could void the user's authority to operate the equipment.

IMPORTANT: OEMs must test final product to comply with unintentional radiators (FCC section 15.107 & 15.109) before declaring compliance of their final product to Part 15 of the FCC Rules.

IMPORTANT: The RF module has been certified for remote and base radio applications. If the module will be used for portable applications, the device must undergo SAR testing.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures: Re-orient or relocate the receiving antenna, Increase the separation between the equipment and receiver, Connect equipment and receiver to outlets on different circuits, or Consult the dealer or an experienced radio/TV technician for help.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCC-Approved Antennas (2.4 GHz)

The XBee/XBee-Pro OEM RF Module can be installed utilizing antennas and cables constructed with standard connectors (Type-N, SMA, TNC, etc.) if the installation is performed professionally and according to FCC guidelines. For installations not performed by a professional, non-standard connectors (RPSMA, RPTNC, etc.) must be used.

The modules are pre-FCC approved for use in fixed base station and mobile applications [refer to table below]. As long as the antenna is mounted at least 20 cm (8 in) from nearby persons, the application is considered a mobile application. Antennas not listed in the table must be tested to comply with FCC Section 15.203 (unique antenna connectors) and Section 15.247 (emissions).

Table A-01. Antennas approved for use with the XBee/XBee-PRO OEM RF Modules (all 2.4 GHz)

Part Number	Type (Description)	Gain	Application	Min. Separation
A24-HABMM-PSI	Dipole (Half-wave bulkhead mount articulated MMCX w/ pigtail)	2.1 dBi	Fixed/Mobile*	20 cm
A24-HBMM-PSI	Dipole (Half-wave bulkhead mount MMCX w/ pigtail)	2.1 dBi	Fixed/Mobile*	20 cm
A24-HABSM	Dipole (Articulated RPSMA)	2.1 dBi	Fixed/Mobile*	20 cm
A24-QBMM-PSI	Monopole (Quarter-wave bulkhead mount MMCX w/pigtail)	1.9 dBi	Fixed/Mobile*	20 cm
A24-QABMM-PSI	Monopole (Quarter-wave bulkhead mount articulated MMCX w/pigtail)	1.9 dBi	Fixed/Mobile*	20 cm
A24-QI	Monopole (Integrated whip)	1.9 dBi	Fixed/Mobile*	20 cm
A24-C1	Surface Mount	-1.5 dBi	Fixed/Mobile*	20 cm
A24-Y4NF	Yagi (4-element)	6.0 dBi	Fixed*	2 m
A24-Y6NF	Yagi (6-element)	8.8 dBi	Fixed*	2 m
A24-Y7NF	Yagi (7-element)	9.0 dBi	Fixed*	2 m
A24-Y9NF	Yagi (9-element)	10.0 dBi	Fixed*	2 m
A24-Y10NF	Yagi (10-element)	11.0 dBi	Fixed*	2 m
A24-Y12NF	Yagi (12-element)	12.0 dBi	Fixed*	2 m
A24-Y13NF	Yagi (13-element)	12.0 dBi	Fixed*	2 m
A24-Y15NF	Yagi (15-element)	12.5 dBi	Fixed*	2 m
A24-Y16NF	Yagi (16-element)	13.5 dBi	Fixed*	2 m
A24-Y16RM	Yagi (16-element, RPSMA connector)	13.5 dBi	Fixed*	2 m
A24-Y18NF	Yagi (18-element)	15.0 dBi	Fixed*	2 m
A24-F2NF	Omni-directional (Fiberglass base station)	2.1 dBi	Fixed/Mobile*	20 cm
A24-F3NF	Omni-directional (Fiberglass base station)	3.0 dBi	Fixed/Mobile*	20 cm
A24-F5NF	Omni-directional (Fiberglass base station)	5.0 dBi	Fixed/Mobile*	20 cm
A24-F8NF	Omni-directional (Fiberglass base station)	8.0 dBi	Fixed*	2 m
A24-F9NF	Omni-directional (Fiberglass base station)	9.5 dBi	Fixed*	2 m
A24-F10NF	Omni-directional (Fiberglass base station)	10.0 dBi	Fixed*	2 m
A24-F12NF	Omni-directional (Fiberglass base station)	12.0 dBi	Fixed*	2 m
A24-F15NF	Omni-directional (Fiberglass base station)	15.0 dBi	Fixed*	2 m
A24-W7NF	Omni-directional (Base station)	7.2 dBi	Fixed*	2 m
A24-M7NF	Omni-directional (Mag-mount base station)	7.2 dBi	Fixed*	2 m
A24-P8SF	Flat Panel	8.5 dBi	Fixed*	2 m
A24-P8NF	Flat Panel	8.5 dBi	Fixed*	2 m
A24-P13NF	Flat Panel	13.0 dBi	Fixed*	2 m
A24-P14NF	Flat Panel	14.0 dBi	Fixed*	2 m
A24-P15NF	Flat Panel	15.0 dBi	Fixed*	2 m
A24-P16NF	Flat Panel	16.0 dBi	Fixed*	2 m
A24-P19NF	Flat Panel	19.0 dBi	Fixed*	2 m

* Antennas can be approved for portable applications if integrator gains approval through SAR testing. If the antenna will be mounted closer than 20 cm to nearby persons, then the application is considered "portable" and requires an additional test performed on the final product. This test is called the Specific Absorption Rate (SAR) testing and measures the emissions from the module and how they affect the person.

RF Exposure

WARNING: To satisfy FCC RF exposure requirements for mobile transmitting devices, a separation distance of 20 cm or more should be maintained between the antenna of this device and persons during device operation. To ensure compliance, operations at closer than this distance is not recommended. The antenna used for this transmitter must not be co-located in conjunction with any other antenna or transmitter.

The preceding statement must be included as a CAUTION statement in manuals for OEM products to alert users on FCC RF Exposure compliance.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

European Certification (pending)

The XBee/XBee-PRO RF Module has been certified for use in several European countries. For a complete list, refer to www.maxstream.net.

If the XBee/XBee-PRO RF Modules are incorporated into a product, the manufacturer must ensure compliance of the final product to the European harmonized EMC and low-voltage/safety standards. A Declaration of Conformity must be issued for each of these standards and kept on file as described in Annex II of the R&TTE Directive. Furthermore, the manufacturer must maintain a copy of the XBee/XBee-PRO user manual documentation and ensure the final product does not exceed the specified power ratings, antenna specifications, and/or installation requirements as specified in the user manual. If any of these specifications are exceeded in the final product, a submission must be made to a notified body for compliance testing to all required standards.

OEM Labeling Requirements

The 'CE' marking must be affixed to a visible location on the OEM product.

Figure A-02. CE Labeling Requirements

The CE mark shall consist of the initials "CE" taking the following form:

- If the CE marking is reduced or enlarged, the proportions given in the above graduated drawing must be respected.
- The CE marking must have a height of at least 5mm except where this is not possible on account of the nature of the apparatus.
- The CE marking must be affixed visibly, legibly, and indelibly.

Restrictions

France - France imposes restrictions on the 2.4 GHz band. Go to www.art-telecom.Fr or contact MaxStream for more information.

Norway - Norway prohibits operation near Ny-Alesund in Svalbard. More information can be found at the Norway Posts and Telecommunications site (www.npt.no).

Declarations of Conformity

MaxStream has issued Declarations of Conformity for the XBee/XBee-PRO RF Modules concerning emissions, EMC and safety. Files are located in the 'documentation' folder of the MaxStream CD.

Important Note

MaxStream does not list the entire set of standards that must be met for each country. MaxStream customers assume full responsibility for learning and meeting the required guidelines for each country in their distribution market. For more information relating to European compliance of an OEM product incorporating the XBee/XBee-PRO RF Module, contact MaxStream, or refer to the following web sites:

CEPT ERC 70-03E - Technical Requirements, European restrictions and general requirements: Available at www.ero.dk/.

R&TTE Directive - Equipment requirements, placement on market: Available at www.ero.dk/.

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix B: Development Guide

Development Kit Contents

The XBee Development Kit includes the hardware and software needed to rapidly create long range wireless links between devices.

Table B-01. Items Included in the Development Kit

Item	Qty.	Description	Part #
XBee-PRO Module	2	(1) OEM RF Module w/ U.FL antenna connector (1) OEM RF Module w/ attached wire antenna	XBP24-...UI-... XBP24-...WI-...
XBee Module	3	(1) OEM RF Module w/ U.FL antenna connector (1) OEM RF Module w/ attached wire antenna (1) OEM RF Module w/ chip antenna	XB24-...UI-... XB24-...WI-... XB24-...CI-...
RS-232 Interface Board	1	Board for interfacing between modules and RS-232 devices (Converts signal levels, displays diagnostic info, & more)	XBIB-R
USB Interface Board	1	Board for interfacing between modules & USB devices (Converts signal levels, displays diagnostic info, & more)	XBIB-U
RS-232 Cable (6', straight-through)	1	Cable for connecting RS-232 interface board with DTE devices (devices that have a male serial DB-9 port - such as most PCs)	JD2D3-CDS-6F
USB Cable (6')	1	Cable for connecting USB interface board to USB devices	JU1U2-CSB-6F
Serial Loopback Adapter	1	[Red] Adapter for configuring the module assembly (module + RS-232 interface board) to function as a repeater for range testing	JD2D3-CDL-A
NULL Modem Adapter (male-to-male)	1	[Black] Adapter for connecting the module assembly (module + RS-232 interface board) to other DCE (female DB-9) devices	JD2D2-CDN-A
NULL Modem Adapter (female-to-female)	1	[Gray] Adapter for connecting serial devices. It allows users to bypass the radios to verify serial cabling is functioning properly.	JD3D3-CDN-A
9VDC Power Adapter	1	Adapter for powering the RS-232 interface board	JP5P2-9V11-6F
9V Battery Clip	1	Clip for remotely powering the RS-232 board w/ a 9V battery	JP2P3-C2C-4I
RPSMA Antenna	1	RPSMA half-wave dipole antenna (2.4 GHz, 2.1 dB)	A24-HASM-525
RF Cable Assembly	1	Adapter for connecting RPSMA antenna to U.FL connector	JF1R6-CR3-4I
CD	1	Documentation and Software	MD0010
Quick Start Guide	1	Step-by-step instruction on how to create wireless links & test range capabilities of the modules	MD0026

Interfacing Options

The development kit includes an RS-232 and a USB interface board. Both boards provide a direct connection to many serial devices and therefore provide access to the RF module registries. Parameters stored in the registry allow OEMs and integrators to customize the modules to suite the needs of their data radio systems.

The following sections illustrate how to use the interface boards for development purposes. The MaxStream Interface board provides means for connecting the module to any node that has an available RS-232 or USB connector. Since the module requires signals to enter at TTL voltages, one of the main functions of the interface board is to convert signals between TTL levels and RS-232 and USB levels.

Note: In the following sections, an OEM RF Module mounted to an interface board will be referred to as a "Module Assembly".

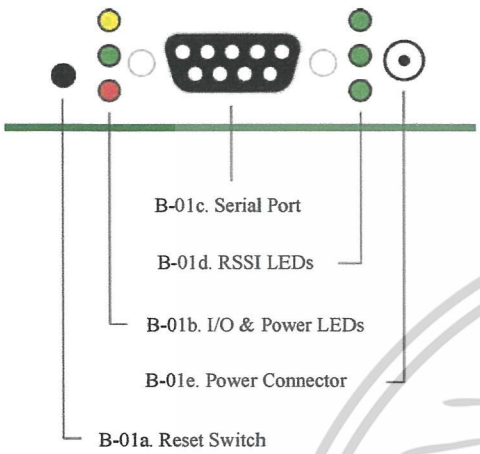
© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RS-232 Interface Board

Physical Interface

Figure B-01. Front View



B-01a. Reset Switch

The Reset Switch is used to reset (re-boot) the RF module. This switch only applies when using the configuration tabs of MaxStream's X-CTU Software.

B-01b. I/O & Power LEDs

LEDs indicate RF module activity as follows:

Yellow (top LED) = Serial Data Out (to host)
Green (middle) = Serial Data In (from host)
Red (bottom) = Power/TX Indicator (LED is on when module assembly is powered)

B-01c. Serial Port

Standard female DB-9 (RS-232) connector.

B-01d. RSSI LEDs

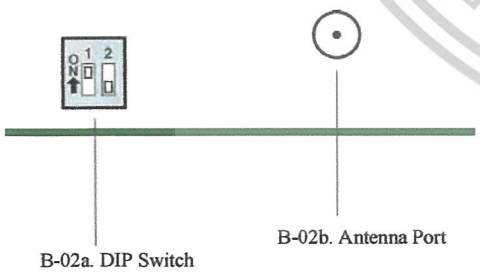
RSSI LEDs indicate the amount of fade margin present in an active wireless link. Fade margin is defined as the difference between the incoming signal strength and the modem's receiver sensitivity.

3 LEDs ON = Very Strong Signal (> 30 dB fade margin)
2 LEDs ON = Strong Signal (> 20 dB fade margin)
1 LED ON = Moderate Signal (> 10 dB fade margin)
0 LED ON = Weak Signal (< 10 dB fade margin)

B-01e. Power Connector

5-14 VDC power connector

Figure B-02. Back View



B-02a. DIP Switch

DIP Switch functions are not supported in this release. Future downloadable firmware versions will support DIP Switch configurations.

B-02b. Antenna Port

Port is a 50Ω RF signal connector for connecting to an external antenna. The connector type is RPSMA (Reverse Polarity SMA) female. The connector has threads on the outside of a barrel and a male center conductor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RS-232 Pin Signals

Figure B-03. Pins used on the female RS-232 (DB-9) Serial Connector

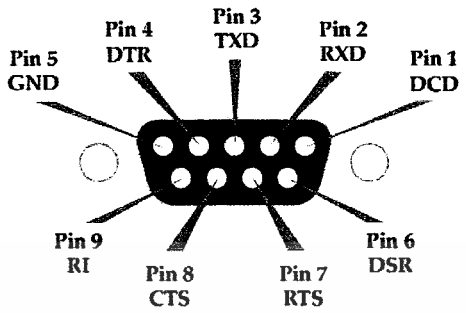


Table B-02. Pin Assignments and Implementations

DB-9 Pin	RS-232 Name	Description	Implementation*
1	DCD	Data-Carrier-Detect	Connected to DSR (pin6)
2	RXD	Received Data	Serial data exiting the module assembly (to host)
3	TXD	Transmitted Data	Serial data entering into the module assembly (from host)
4	DTR	Data-Terminal-Ready	Can enable Power-Down on the module assembly
5	GND	Ground Signal	Ground
6	DSR	Data-Set-Ready	Connected to DCD (pin1)
7	$\overline{\text{RTS}}$ / CMD	Request-to-Send / Command Mode	Provides $\overline{\text{RTS}}$ flow control or enables Command Mode
8	$\overline{\text{CTS}}$	Clear-to-Send	Provides $\overline{\text{CTS}}$ flow control
9	RI	Ring Indicator	Optional power input that is connected internally to the positive lead of the front power connector

* Functions listed in the implementation column may not be available at the time of release.

 © 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Wiring Diagrams

Figure B-04. DTE Device (RS-232, male DB-9 connector) wired to a DCE Module Assembly (female DB-9)

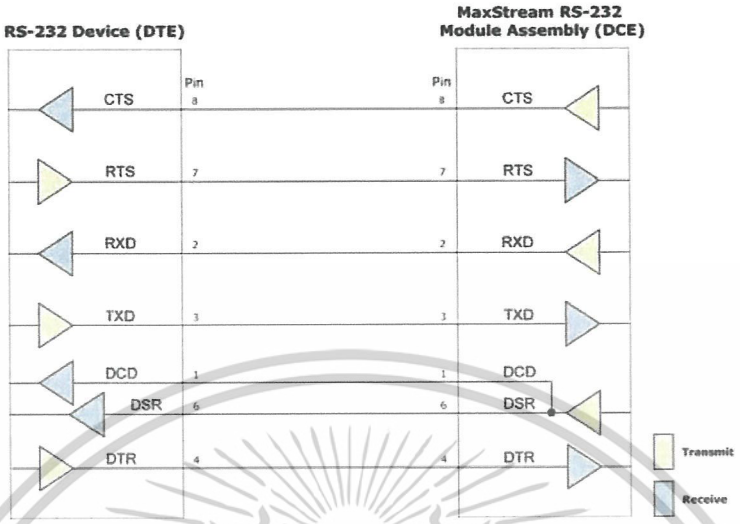
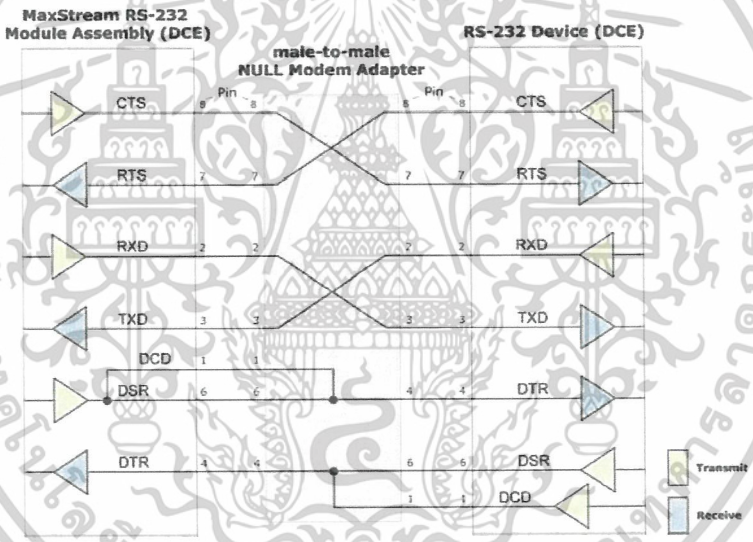
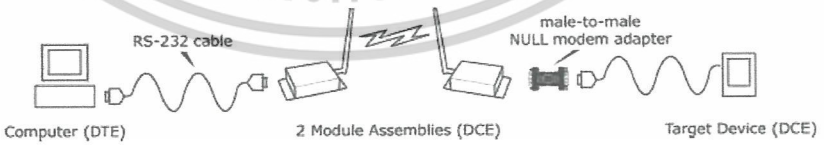


Figure B-05. DCE Module Assembly (female DB-9 connector) wired to a DCE Device (RS-232, male DB-9)



Sample Wireless Connection: DTE <--> DCE <--> DCE <--> DCE

Figure B-06. Typical wireless link between DTE and DCE devices



© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Adapters

The development kit includes several adapters that support the following functions:

- Performing Range Tests
- Testing Cables
- Connecting to other RS-232 DCE and DTE devices
- Connecting to terminal blocks or RJ-45 (for RS-485/422 devices)

NULL Modem Adapter (male-to-male)

Part Number: JD2D2-CDN-A (Black, DB-9 M-M) The male-to-male NULL modem adapter is used to connect two DCE devices. A DCE device connects with a straight-through cable to the male serial port of a computer (DTE).

Figure B-07. Male NULL modem adapter and pinouts

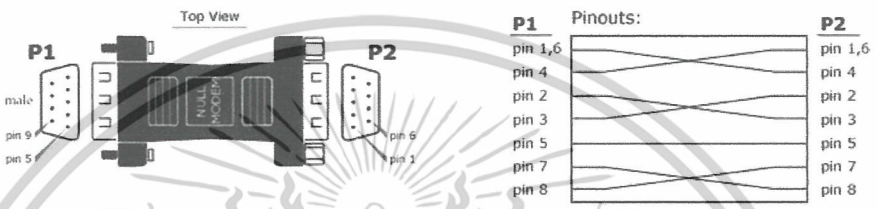
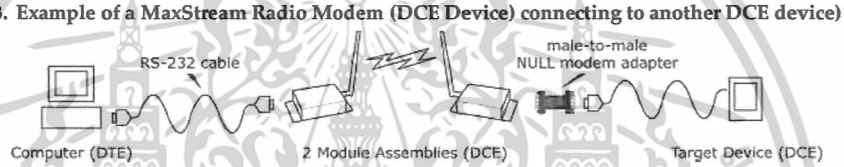


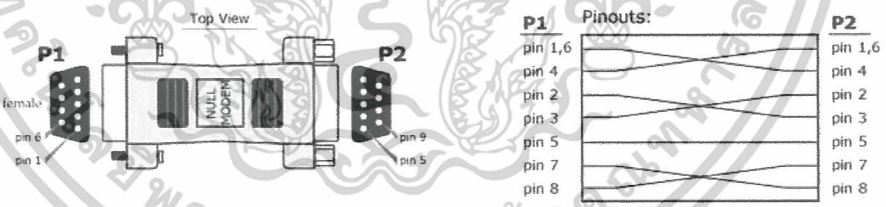
Figure B-08. Example of a MaxStream Radio Modem (DCE Device) connecting to another DCE device)



NULL Modem Adapter (female-to-female)

Part Number: JD3D3-CDN-A (Gray, DB-9 F-F) The female-to-female NULL modem adapter is used to verify serial cabling is functioning properly. To test cables, insert the female-to-female NULL modem adapter in place of a pair of module assemblies (RS-232 interface board + XTend Module) and test the connection without radio modules in the connection.

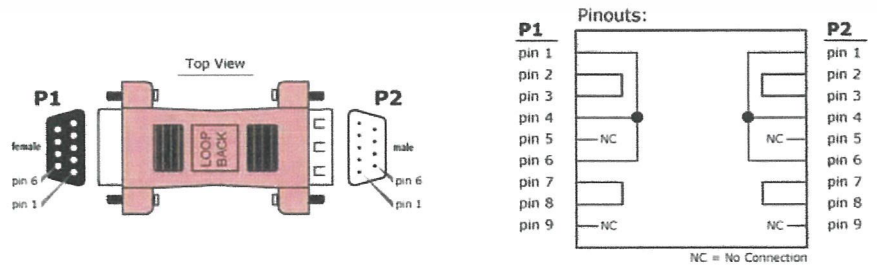
Figure B-09. Female NULL modem adapter and pinouts



Serial Loopback Adapter

Part Number: JD2D3-CDL-A (Red, DB-9 M-F) The serial loopback adapter is used for range testing. During a range test, the serial loopback adapter configures the module to function as a repeater by looping serial data back into the radio for retransmission.

Figure B-10. Serial loopback adapter and pinouts



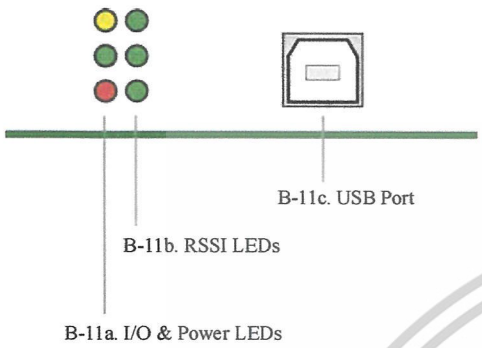
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

USB Interface Board

Physical Interface

B-11a. I/O & Power LEDs

Figure B-11. Front View



LEDs indicate RF module activity as follows:

Yellow (top LED) = Serial Data Out (to host)
Green (middle) = Serial Data In (from host)
Red (bottom) = Power/TX Indicator (Red LED is illuminated when RF module is powered)

B-11b. RSSI LEDs

RSSI LEDs indicate the amount of fade margin present in an active wireless link. Fade margin is defined as the difference between the incoming signal strength and the module's receiver sensitivity.

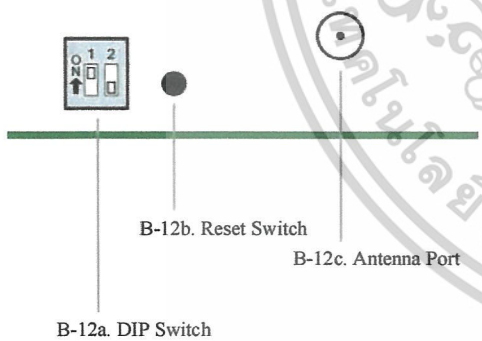
3 LEDs ON	=	Very Strong Signal (> 30 dB fade margin)
2 LEDs ON	=	Strong Signal (> 20 dB fade margin)
1 LED ON	=	Moderate Signal (> 10 dB fade margin)
0 LED ON	=	Weak Signal (< 10 dB fade margin)

B-11c. USB Port

Standard Type-B OEM connector is used to communicate with OEM host and power the RF module.

B-12a. DIP Switch

Figure B-12. Back View



DIP Switch functions are not supported in this release. Future downloadable firmware versions will support the DIP Switch configurations.

B-12b Reset Switch

The Reset Switch is used to reset (re-boot) the RF module.

B-12c. Antenna Port

Port is a 50Ω RF signal connector for connecting to an external antenna. The connector type is RPSMA (Reverse Polarity SMA) female. The connector has threads on the outside of a barrel and a male center conductor.

USB Pin Signals

Table B-03. USB signals and their implantations on the XBee/XBee-PRO RF Module

Pin	Name	Description	Implementation
1	VBUS	Power	Power the RF module
2	D-	Transmitted & Received Data	Transmit data to and from the RF module
3	D+	Transmitted & Received Data	Transmit data to and from the RF module
4	GND	Ground Signal	Ground

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix C: Additional Information

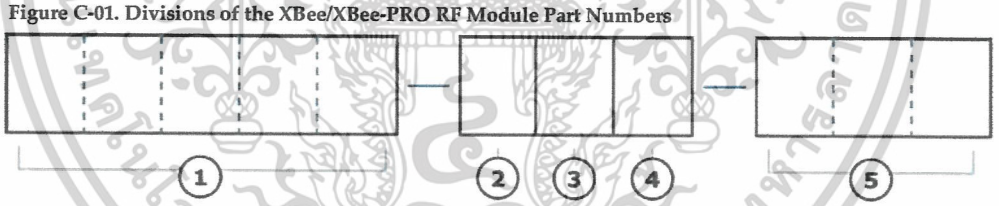
1-Year Warranty

XBee/XBee-PRO RF Modules from MaxStream, Inc. (the "Product") are warranted against defects in materials and workmanship under normal use, for a period of 1-year from the date of purchase. In the event of a product failure due to materials or workmanship, MaxStream will repair or replace the defective product. For warranty service, return the defective product to MaxStream, shipping prepaid, for prompt repair or replacement.

The foregoing sets forth the full extent of MaxStream's warranties regarding the Product. Repair or replacement at MaxStream's option is the exclusive remedy. THIS WARRANTY IS GIVEN IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, AND MAXSTREAM SPECIFICALLY DISCLAIMS ALL WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL MAXSTREAM, ITS SUPPLIERS OR LICENSORS BE LIABLE FOR DAMAGES IN EXCESS OF THE PURCHASE PRICE OF THE PRODUCT, FOR ANY LOSS OF USE, LOSS OF TIME, INCONVENIENCE, COMMERCIAL LOSS, LOST PROFITS OR SAVINGS, OR OTHER INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PRODUCT, TO THE FULL EXTENT SUCH MAY BE DISCLAIMED BY LAW. SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES. THEREFORE, THE FOREGOING EXCLUSIONS MAY NOT APPLY IN ALL CASES. This warranty provides specific legal rights. Other rights which vary from state to state may also apply.

Ordering Information

Figure C-01. Divisions of the XBee/XBee-PRO RF Module Part Numbers



(1) MaxStream Product Family
- XB24 = XBee 2.4 GHz
- XBP24 = XBee-PRO 2.4 GHz

(2) Reserved for internal use

(3) Antenna Option
- C = Chip Antenna
- U = U.FL RF Connector
- W = Attached Wire Antenna

(4) Rating
- I = Industrial (-40 to 85° C)
- D = Class I, Division 2

(5) Protocol
- 001 = 802.15.4
- 002 = ZigBee

For example:

XBP24-AWI-001 = XBee-PRO OEM RF Module, 2.4 GHz, attached wire antenna, Industrial temperature rating, IEEE 802.15.4 standard

© 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Contact MaxStream

Free and unlimited technical support is included with every MaxStream Radio Modem sold.

For the best in wireless data solutions and support, please use the following resources:

Documentation:	www.maxstream.net/helpdesk/download.php	
Technical Support:	Phone.	(866) 765-9885 toll-free U.S.A. & Canada (801) 765-9885 Worldwide
	Live Chat.	www.maxstream.net
	E-Mail.	rf-xperts@maxstream.net

MaxStream office hours are 8:00 am - 5:00 pm [U.S. Mountain Standard Time]

 © 2005 MaxStream, Inc., Confidential & Proprietary - All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single-clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
  - 16K Bytes of In-System Self-Programmable Flash
    Endurance: 10,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    In-System Programming by On-chip Boot Program
    True Read-While-Write Operation
  - 512 Bytes EEPROM
    Endurance: 100,000 Write/Erase Cycles
  - 1K Byte Internal SRAM
  - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Four PWM Channels
  - 8-channel, 10-bit ADC
    8 Single-ended Channels
    7 Differential Channels in TQFP Package Only
    2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
  - Byte-oriented Two-wire Serial Interface
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
  - 32 Programmable I/O Lines
  - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
  - 2.7 - 5.5V for ATmega16L
  - 4.5 - 5.5V for ATmega16
- Speed Grades
  - 0 - 8 MHz for ATmega16L
  - 0 - 16 MHz for ATmega16
- Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L
  - Active: 1.1 mA
  - Idle Mode: 0.35 mA
  - Power-down Mode: < 1 µA

8-bit AVR® Microcontroller with 16K Bytes In-System Programmable Flash

ATmega16
ATmega16L

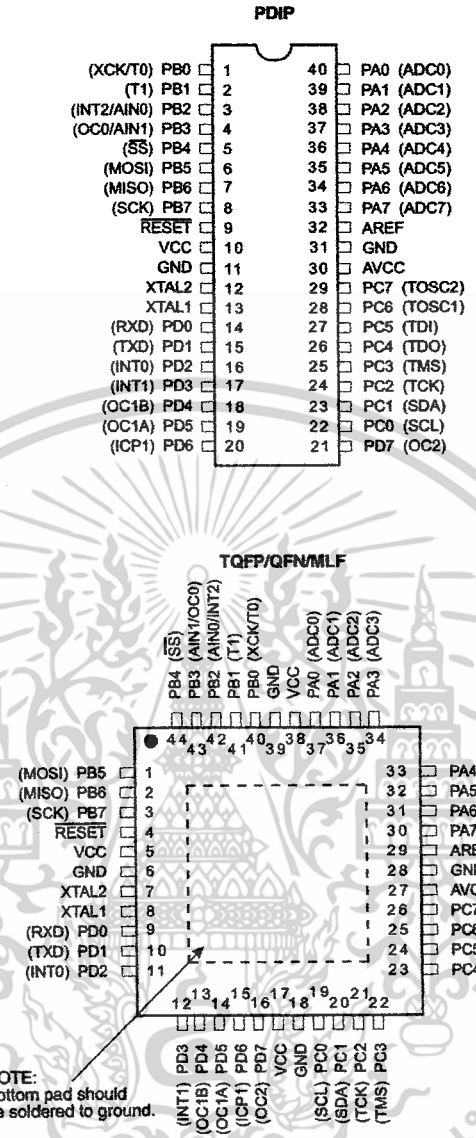
Summary

2466MS–AVR–04/06

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Configurations

Figure 1. Pinout ATmega16



Disclaimer

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

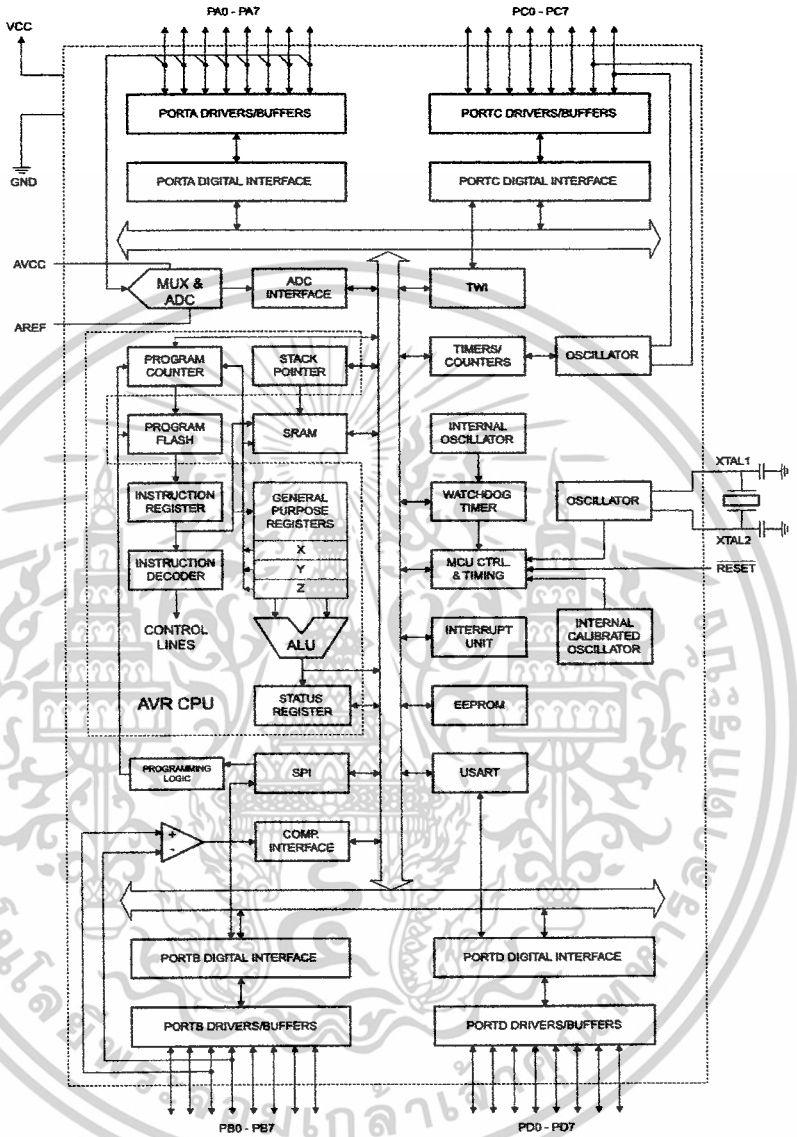
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2. Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega16 provides the following features: 16K bytes of In-System Programmable Flash Program memory with Read-While-Write capabilities, 512 bytes EEPROM, 1K byte SRAM, 32 general purpose I/O lines, 32 general purpose working registers, a JTAG interface for Boundary-scan, On-chip Debugging support and programming, three flexible Timer/Counters with compare modes, Internal and External Interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, an 8-channel, 10-bit ADC with optional differential input stage with programmable gain (TQFP package only), a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the USART, Two-wire interface, A/D Converter, SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next External Interrupt or Hardware Reset. In Power-save mode, the Asynchronous Timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega16 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega16 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Pin Descriptions

VCC

Digital supply voltage.

GND

Ground.

Port A (PA7..PA0)

Port A serves as the analog inputs to the A/D Converter.

Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Port B (PB7..PB0)

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B also serves the functions of various special features of the ATmega16 as listed on page 58.

Port C (PC7..PC0)

Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs.

Port C also serves the functions of the JTAG interface and other special features of the ATmega16 as listed on page 61.

Port D (PD7..PD0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATmega16 as listed on page 63.

$\overline{\text{RESET}}$

Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 38. Shorter pulses are not guaranteed to generate a reset.

XTAL1

Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting Oscillator amplifier.

AVCC

AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter.

AREF

AREF is the analog reference pin for the A/D Converter.

Resources

A comprehensive set of development tools, application notes and datasheets are available for download on http://www.atmel.com/avr.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$3F ($5F)	SREG	I	T	H	S	V	N	Z	C	9
$3E ($5E)	SPH	–	–	–	–	–	SP10	SP9	SP8	12
$3D ($5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	12
$3C ($5C)	OCR0	Timer/Counter0 Output Compare Register								85
$3B ($5B)	GICR	INT1	INT0	INT2	–	–	–	IVSEL	IVCE	48, 69
$3A ($5A)	GIFR	INTF1	INTF0	INTF2	–	–	–	–	–	70
$39 ($59)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	85, 116, 134
$38 ($58)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	86, 117, 134
$37 ($57)	SPMCR	SPMIE	RWWSB	–	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	253
$36 ($56)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	181
$35 ($55)	MCUCR	SM2	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00	32, 68
$34 ($54)	MCUCSR	JTD	ISC2	–	JTRF	WDRF	BORF	EXTRF	PORF	41, 69, 232
$33 ($53)	TCCR0	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	83
$32 ($52)	TCNT0	Timer/Counter0 (8 Bits)								85
$31[1] ($51)[1]	OSCCAL	Oscillator Calibration Register								30
	OCDR	On-Chip Debug Register								228
$30 ($50)	SFIOR	ADTS2	ADTS1	ADTS0	–	ACME	PUD	PSR2	PSR10	57,88,135,202,222
$2F ($4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	111
$2E ($4E)	TCCR1B	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	114
$2D ($4D)	TCNT1H	Timer/Counter1 – Counter Register High Byte								115
$2C ($4C)	TCNT1L	Timer/Counter1 – Counter Register Low Byte								115
$2B ($4B)	OCR1AH	Timer/Counter1 – Output Compare Register A High Byte								115
$2A ($4A)	OCR1AL	Timer/Counter1 – Output Compare Register A Low Byte								115
$29 ($49)	OCR1BH	Timer/Counter1 – Output Compare Register B High Byte								115
$28 ($48)	OCR1BL	Timer/Counter1 – Output Compare Register B Low Byte								115
$27 ($47)	ICR1H	Timer/Counter1 – Input Capture Register High Byte								116
$26 ($46)	ICR1L	Timer/Counter1 – Input Capture Register Low Byte								116
$25 ($45)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	129
$24 ($44)	TCNT2	Timer/Counter2 (8 Bits)								131
$23 ($43)	OCR2	Timer/Counter2 Output Compare Register								131
$22 ($42)	ASSR	–	–	–	–	AS2	TCN2UB	OCR2UB	TCR2UB	132
$21 ($41)	WDTCR	–	–	–	WDTOE	WDE	WDP2	WDP1	WDP0	43
$20[2] ($40)[2]	UBRRH	URSEL	–	–	–	UBRR[11:8]				168
	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	167
$1F ($3F)	EEARH	–	–	–	–	–	–	–	EEAR8	19
$1E ($3E)	EEARL	EEPROM Address Register Low Byte								19
$1D ($3D)	EEDR	EEPROM Data Register								19
$1C ($3C)	EECR	–	–	–	–	EERIE	EEMWE	EEWE	EERE	19
$1B ($3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	66
$1A ($3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	66
$19 ($39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	66
$18 ($38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	66
$17 ($37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	66
$16 ($36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	66
$15 ($35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	67
$14 ($34)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	67
$13 ($33)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	67
$12 ($32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	67
$11 ($31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	67
$10 ($30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	67
$0F ($2F)	SPDR	SPI Data Register								143
$0E ($2E)	SPSR	SPIF	WCOL	–	–	–	–	–	SPI2X	143
$0D ($2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	141
$0C ($2C)	UDR	USART I/O Data Register								164
$0B ($2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	165
$0A ($2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	166
$09 ($29)	UBRRL	USART Baud Rate Register Low Byte								168
$08 ($28)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	203
$07 ($27)	ADMUX	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	218
$06 ($26)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	220
$05 ($25)	ADCH	ADC Data Register High Byte								221
$04 ($24)	ADCL	ADC Data Register Low Byte								221
$03 ($23)	TWDR	Two-wire Serial Interface Data Register								183
$02 ($22)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	183

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$01 ($21)	TWSR	TWS7	TWS6	TWS5	TWS4	TWS3	–	TWPS1	TWPS0	182
$00 ($20)	TWBR	Two-wire Serial Interface Bit Rate Register								181

Notes:
1. When the OCDEN Fuse is unprogrammed, the OSCCAL Register is always accessed on this address. Refer to the debugger specific documentation for details on how to use the OCDR Register.
2. Refer to the USART description for details on how to access UBRRH and UCSRC.
3. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.
4. Some of the Status Flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O Register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers $00 to $1F only.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instruction Set Summary

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	Rd ← Rd + Rr	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	Rd ← Rd + Rr + C	Z,C,N,V,H	1
ADIW	Rdl,K	Add Immediate to Word	Rdh:Rdl ← Rdh:Rdl + K	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	Rd ← Rd - Rr	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	Rd ← Rd - K	Z,C,N,V,H	1
SBC	Rd, Rr	Subtract with Carry two Registers	Rd ← Rd - Rr - C	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	Rd ← Rd - K - C	Z,C,N,V,H	1
SBIW	Rdl,K	Subtract Immediate from Word	Rdh:Rdl ← Rdh:Rdl - K	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND Registers	Rd ← Rd • Rr	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	Rd ← Rd • K	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	Rd ← Rd v Rr	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	Rd ← Rd v K	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	Rd ← Rd ⊕ Rr	Z,N,V	1
COM	Rd	One's Complement	Rd ← $FF – Rd	Z,C,N,V	1
NEG	Rd	Two's Complement	Rd ← $00 – Rd	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	Rd ← Rd v K	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	Rd ← Rd • ($FF - K)	Z,N,V	1
INC	Rd	Increment	Rd ← Rd + 1	Z,N,V	1
DEC	Rd	Decrement	Rd ← Rd – 1	Z,N,V	1
TST	Rd	Test for Zero or Minus	Rd ← Rd • Rd	Z,N,V	1
CLR	Rd	Clear Register	Rd ← Rd ⊕ Rd	Z,N,V	1
SER	Rd	Set Register	Rd ← $FF	None	1
MUL	Rd, Rr	Multiply Unsigned	R1:R0 ← Rd x Rr	Z,C	2
MULS	Rd, Rr	Multiply Signed	R1:R0 ← Rd x Rr	Z,C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	R1:R0 ← Rd x Rr	Z,C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	R1:R0 ← (Rd x Rr) << 1	Z,C	2
FMULS	Rd, Rr	Fractional Multiply Signed	R1:R0 ← (Rd x Rr) << 1	Z,C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	R1:R0 ← (Rd x Rr) << 1	Z,C	2
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	PC ← PC + k + 1	None	2
IJMP		Indirect Jump to (Z)	PC ← Z	None	2
JMP	k	Direct Jump	PC ← k	None	3
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	None	3
ICALL		Indirect Call to (Z)	PC ← Z	None	3
CALL	k	Direct Subroutine Call	PC ← k	None	4
RET		Subroutine Return	PC ← STACK	None	4
RETI		Interrupt Return	PC ← STACK	I	4
CPSE	Rd,Rr	Compare, Skip if Equal	if (Rd = Rr) PC ← PC + 2 or 3	None	1 / 2 / 3
CP	Rd,Rr	Compare	Rd – Rr	Z, N,V,C,H	1
CPC	Rd,Rr	Compare with Carry	Rd – Rr – C	Z, N,V,C,H	1
CPI	Rd,K	Compare Register with Immediate	Rd – K	Z, N,V,C,H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if (Rr(b)=0) PC ← PC + 2 or 3	None	1 / 2 / 3
SBRS	Rr, b	Skip if Bit in Register is Set	if (Rr(b)=1) PC ← PC + 2 or 3	None	1 / 2 / 3
SBIC	P, b	Skip if Bit in I/O Register Cleared	if (P(b)=0) PC ← PC + 2 or 3	None	1 / 2 / 3
SBIS	P, b	Skip if Bit in I/O Register is Set	if (P(b)=1) PC ← PC + 2 or 3	None	1 / 2 / 3
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then PC←PC+k + 1	None	1 / 2
BRBC	s, k	Branch if Status Flag Cleared	if (SREG(s) = 0) then PC←PC+k + 1	None	1 / 2
BREQ	k	Branch if Equal	if (Z = 1) then PC ← PC + k + 1	None	1 / 2
BRNE	k	Branch if Not Equal	if (Z = 0) then PC ← PC + k + 1	None	1 / 2
BRCS	k	Branch if Carry Set	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRCC	k	Branch if Carry Cleared	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRSH	k	Branch if Same or Higher	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRLO	k	Branch if Lower	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRMI	k	Branch if Minus	if (N = 1) then PC ← PC + k + 1	None	1 / 2
BRPL	k	Branch if Plus	if (N = 0) then PC ← PC + k + 1	None	1 / 2
BRGE	k	Branch if Greater or Equal, Signed	if (N ⊕ V= 0) then PC ← PC + k + 1	None	1 / 2
BRLT	k	Branch if Less Than Zero, Signed	if (N ⊕ V= 1) then PC ← PC + k + 1	None	1 / 2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then PC ← PC + k + 1	None	1 / 2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then PC ← PC + k + 1	None	1 / 2
BRTS	k	Branch if T Flag Set	if (T = 1) then PC ← PC + k + 1	None	1 / 2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then PC ← PC + k + 1	None	1 / 2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then PC ← PC + k + 1	None	1 / 2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then PC ← PC + k + 1	None	1 / 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mnemonics	Operands	Description	Operation	Flags	#Clocks
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ← PC + k + 1	None	1 / 2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1 / 2
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	Rd ← Rr	None	1
MOVW	Rd, Rr	Copy Register Word	Rd+1:Rd ← Rr+1:Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LD	Rd, X	Load Indirect	Rd ← (X)	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	Rd ← (X), X ← X + 1	None	2
LD	Rd, - X	Load Indirect and Pre-Dec.	X ← X - 1, Rd ← (X)	None	2
LD	Rd, Y	Load Indirect	Rd ← (Y)	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	Rd ← (Y), Y ← Y + 1	None	2
LD	Rd, - Y	Load Indirect and Pre-Dec.	Y ← Y - 1, Rd ← (Y)	None	2
LDD	Rd,Y+q	Load Indirect with Displacement	Rd ← (Y + q)	None	2
LD	Rd, Z	Load Indirect	Rd ← (Z)	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	Rd ← (Z), Z ← Z+1	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	Z ← Z - 1, Rd ← (Z)	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	Rd ← (Z + q)	None	2
LDS	Rd, k	Load Direct from SRAM	Rd ← (k)	None	2
ST	X, Rr	Store Indirect	(X) ← Rr	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	(X) ← Rr, X ← X + 1	None	2
ST	- X, Rr	Store Indirect and Pre-Dec.	X ← X - 1, (X) ← Rr	None	2
ST	Y, Rr	Store Indirect	(Y) ← Rr	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	(Y) ← Rr, Y ← Y + 1	None	2
ST	- Y, Rr	Store Indirect and Pre-Dec.	Y ← Y - 1, (Y) ← Rr	None	2
STD	Y+q,Rr	Store Indirect with Displacement	(Y + q) ← Rr	None	2
ST	Z, Rr	Store Indirect	(Z) ← Rr	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	(Z) ← Rr, Z ← Z + 1	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	Z ← Z - 1, (Z) ← Rr	None	2
STD	Z+q,Rr	Store Indirect with Displacement	(Z + q) ← Rr	None	2
STS	k, Rr	Store Direct to SRAM	(k) ← Rr	None	2
LPM		Load Program Memory	R0 ← (Z)	None	3
LPM	Rd, Z	Load Program Memory	Rd ← (Z)	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc	Rd ← (Z), Z ← Z+1	None	3
SPM		Store Program Memory	(Z) ← R1:R0	None	-
IN	Rd, P	In Port	Rd ← P	None	1
OUT	P, Rr	Out Port	P ← Rr	None	1
PUSH	Rr	Push Register on Stack	STACK ← Rr	None	2
POP	Rd	Pop Register from Stack	Rd ← STACK	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P,b	Set Bit in I/O Register	I/O(P,b) ← 1	None	2
CBI	P,b	Clear Bit in I/O Register	I/O(P,b) ← 0	None	2
LSL	Rd	Logical Shift Left	Rd(n+1) ← Rd(n), Rd(0) ← 0	Z,C,N,V	1
LSR	Rd	Logical Shift Right	Rd(n) ← Rd(n+1), Rd(7) ← 0	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	Rd(0)←C,Rd(n+1)← Rd(n),C← Rd(7)	Z,C,N,V	1
ROR	Rd	Rotate Right Through Carry	Rd(7)← C,Rd(n)← Rd(n+1),C← Rd(0)	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ← Rd(n+1), n=0..6	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	Rd(3..0)←Rd(7..4),Rd(7..4)←Rd(3..0)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	T	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ← T	None	1
SEC		Set Carry	C ← 1	C	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Twos Complement Overflow.	V ← 1	V	1
CLV		Clear Twos Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	H	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mnemonics	Operands	Description	Operation	Flags	#Clocks
CLH		Clear Half Carry Flag in SREG	H ← 0	H	1
MCU CONTROL INSTRUCTIONS					
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR/timer)	None	1
BREAK		Break	For On-Chip Debug Only	None	N/A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] ประจิน พลังสันติกุล , การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ AVR 1 ด้วยภาษา C กับ WinAVR (C Compiler) , แอพซอฟต์เทค, กรุงเทพฯ, 2550.

[2] ประจิน พลังสันติกุล , การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ AVR 2 ด้วยภาษา C กับ WinAVR (C Compiler) , แอพซอฟต์เทค, กรุงเทพฯ, 2551.

[3] "ส่งข้อมูลแบบไร้สายด้วยโมดูล Zigbee" อารัมภีย์ จันทร์ใย หนังสือ เซมิคอนดักเตอร์ อิเล็กทรอนิกส์ ฉบับที่ 299 มีนาคม 2550.

[4] " MaxStream XBee/XBee Pro OEM RF Modules"[Online]:http://www.maxstream.net.

[5] Bull, D., N. Canagarajan, and A. Nix," Mobile MultimediaCommunications".[Online] Academic.

[6] Bob Blumenscheid, " Digi International".[Online]:http://www.digi.com.

[7] "ZigBeeCharacteristics".[Online].Available: http://www.tutorialreports.com/Zigbee.

[8] IEEE 802.15.4 WPAN-LR Task Group,[Online]:www.ieee802.org /TG4Expert.html , 10-2005.

[9] Zigbee Alliance,[Online]: http://www.zigbee.org/en/index.asp , 10-2005.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้