สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การออกแบบและการสร้างระบบการสื่อสารระหว่างรถยนต์

DESIGN AND IMPLEMENTATION OF INTER VEHICLE COMMUNICATION SYSTEM

โดย

นางสาวชฎาพร จันทร์พาณิชระวี

นายวิศรุต ศุขหมอก

TK
5105.5
.C14 P
2551

เลขหมู่.....
เลขทะเบียน 83152
วัน,เดือน,ปี - 6 ส.ค. 2551

b 11961879
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DESIGN AND IMPLEMENTATION OF INTER VEHICLE COMMUNICATION SYSTEM

BY

Miss CHADAPON CHANPANITRAVEE

Mr. VISARUT SUKMOK

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMNT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การออกแบบและการสร้างระบบการสื่อสารระหว่างรถยนต์
ชื่อนักศึกษา	นางสาว ชฎาพร จันทร์พาณิชระวี รหัส 47010133
	นาย วิศรุต ศุขหมอก รหัส 47010718
อาจารย์ที่ปรึกษา	ผศ. มนต์ชัย แช่มช้อย
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
	สาขา วิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2550

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตร์บัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(ผศ. มนต์ชัย แช่มช้อย)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การออกแบบและการสร้างระบบการสื่อสารระหว่างรถยนต์
ชื่อนักศึกษา	นางสาว ชฎาพร จันทร์พาณิชระวี รหัส 47010133
	นาย วิศรุต ศุขหมอก รหัส 47010718
อาจารย์ที่ปรึกษา	ผศ. มนต์ชัย แช่มช้อย
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
	สาขา วิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2550

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ ได้ทำการออกแบบและสร้างการสื่อสารระหว่างรถยนต์ เพื่อใช้ลดอุบัติเหตุที่เกิดขึ้นบนท้องถนนซึ่งเป็นส่วนหนึ่งของระบบขนส่งและจราจรอัจฉริยะ (Intelligent Transport Systems : ITS) โดยทำการส่งข้อมูลที่เกี่ยวข้องกับความปลอดภัยไปให้กับผู้ใช้รถที่อยู่โดยรอบผ่านระบบสมองกลฝังตัว (Embedded System) ที่เป็นสื่อกลางในการติดต่อสื่อสารระหว่างรถยนต์ ซึ่งในส่วนควบคุม ประกอบด้วยไมโครคอลโทรเลอร์ (Microcontroller) ตระกูล AVR ในส่วนของการรับ-ส่งข้อมูลใช้อุปกรณ์สื่อสารตามมาตรฐาน IEEE 802.15.4/Zigbee อุปกรณ์ที่ได้นี้จะสามารถรับ-ส่งข้อมูลได้ที่ระยะทางสูงสุดประมาณ 1 กิโลเมตร มีกำลังงานในการส่งสูงสุด 18 dBm และมีอัตราการรับ-ส่งข้อมูลสูงสุดได้ที่ 250 kbps ซึ่งเป็นการส่งที่มีประสิทธิภาพมากเพราะเกิดความผิดพลาดในการรับ-ส่งข้อมูลที่ 1% (1% packet error rate) เมื่อรับ-ส่งข้อมูลในรูปแบบแพ็คเก็ต โดยในโครงงานนี้ได้เสดงรูปแบบของการติดต่อสื่อสารแบบไร้สายที่สามารถควบคุมและสั่งงานได้จากทางเครื่องคอมพิวเตอร์ อีกทั้งยังสามารถนำไปพัฒนาเพื่อใช้งานร่วมกับระบบ GPS บนรถยนต์ทำให้ระบุตำแหน่งที่ชัดเจน เพื่อความสะดวกและความปลอดภัยของผู้ใช้งานได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	DESIGN AND IMPLEMENTATION OF INTER VEHICLE COMMUNICATION SYSTEM
Student	Miss CHADAPON CHANPANITRAVEE ID 47010133
	Mr. VISARUT SUKMOK ID 47010718
Advisor	Asst. Prof. Monchai Chamchoy
Graduate Level	Bachelor Degree of Information Engineering
Department	Information Engineering
Academic Year	2007

Abstract

This thesis is dedicated to the designing and creating an inter-vehicle network system between cars or other mobile objects. Our objective is to decrease the number in car accidents and provide more information to the drivers. This system is similar to the ITS(Intelligent Transport Systems) system in most countries. The embedded system is the link between the cars, using the AVR microcontroller and the IEEE standard 802.15.4/ZigBee transceiver with a radius of 1 kilometer and 18 dBm transmit power with a bandwidth of 250 kbps. The packet error rate as low as 1 percent it is most suitable for controlling devices with low data transfer. This project will simulate the driver's actions via keyboard input for testing. We hope the system will prevent accident from happening throughout the world. The Project was implemented based on wireless technology controlled by software programming. The AVR microcontroller can be equipped with various devices which its data can be transmitted via ZigBee also. For example the AVR can be equipped with a GPS module thus enhancing its capability, comfort and safety.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีผู้เขียนขอกราบขอบพระคุณ อาจารย์มนต์ชัย แช่มช้อย อาจารย์ที่ปรึกษาโครงงานที่ให้ความช่วยเหลือรวมทั้งช่วยแก้ไขความผิดพลาดที่เกิดขึ้น และให้คำแนะนำที่เป็นประโยชน์ในการปรับปรุงเนื้อหาในโครงงาน รวมทั้ง อาจารย์ พิชญ สุพรรณกูล เป็นอย่างสูงที่ช่วยชี้แนะแนวทางในการศึกษา การแก้ปัญหา และให้กำลังใจผู้เขียนให้มีความสามารถในการทำงานนี้จนสำเร็จลุล่วงไปได้ ตลอดจนตรวจทานต้นฉบับของปริญญานิพนธ์นี้จนเสร็จสมบูรณ์

ขอขอบคุณ และรุ่นพี่ภายในห้องเลป ที่ให้คำปรึกษาแนะนำเกี่ยวกับหลักการและทฤษฎีสำคัญที่เกี่ยวข้องกับงาน ซึ่งล้วนแล้วแต่เป็นประโยชน์อย่างยิ่งต่อปริญญานิพนธ์ฉบับนี้

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ช
สารบัญตาราง	ฌ
บทที่ 1 บทนำ	
1.1 แนวคิดและที่มาของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงงาน	2
1.4 ขั้นตอนการศึกษา และผลที่คาดว่าจะได้รับ	2
1.5 ลักษณะโดยรวมของโครงงาน	3
บทที่ 2 ทฤษฎีและหลักการ	
2.1 บทนำ	4
2.2 ระบบ ITS	4
2.2.1 ระบบการขนส่งอัจฉริยะ และการให้บริการ	5
2.3 ระบบสมองกลฝังตัว (Embedded Systems)	6
2.3.1 ประเภทของ Embedded System	7
2.4 IEEE 802.15.4/Zigbee	
2.4.1 ประวัติ ZigBee	7
2.4.2 โครงสร้างของเครือข่าย ZigBee	11
2.4.3 ประเภทของอุปกรณ์ที่เชื่อมต่อกับเครือข่าย	11
2.4.4 ลักษณะรูปแบบเครือข่ายแบบต่างๆ	11
2.4.5 วิธีการควบคุมหรือการเปลี่ยนแปลงช่องสัญญาณ (Modulation Technique)	15
2.4.6 โครงสร้างของโพรโทคอล ZigBee	17
2.4.7 ขั้นตอนการทำงานของโพรโทคอล ZigBee	18
2.4.8 I/O (Input/Output)	20
2.4.8.1 ขั้นตอนการทำงานของ ZigBee end-device	21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.4.9 การส่งข้อมูลของ Zigbee	22
2.4.10 การทำงานของ Zigbee	23
2.4.11 เทคโนโลยีไร้สายทั้งสี่ชนิด (WiMax, Mobile-Fi, ZigBee และ Ultrawideband)	27
บทที่ 3 การออกแบบระบบสื่อสาร	
3.1 คำนำ	29
3.2 การออกแบบโครงงานทางด้านฮาร์ดแวร์	30
3.2.1 การออกแบบและวิธีการใช้งาน Zigbee	33
3.2.2 ขั้นตอนการใช้งานเครื่องเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์	41
3.2.3 การออกแบบวงจรด้วย Protel 99 SE	52
3.3 การออกแบบโครงงานทางด้านซอฟท์แวร์	54
3.3.1 ขั้นตอนการใช้งานโปรแกรม Code VisionAVR	56
3.3.2 ตัวอย่างโปรแกรมที่ใช้ในการควบคุมการติดต่อสื่อสาร ระหว่างอุปกรณ์ทั้ง 2 ชิ้น	61
บทที่ 4 ผลการออกแบบและผลการดำเนินงาน	
4.1 อุปกรณ์ที่ใช้ในการทดลอง	64
4.2 การทดทองการสูญเสียของสัญญาณภายในอาคาร	66
4.3 การทดทองการสูญเสียของสัญญาณในที่โล่ง (Free Space Loss)	70
4.4 การทดลองเชื่อมต่อกับระบบสมองกลอัจฉริยะ	71
บทที่ 5 สรุปผลการดำเนินโครงงาน	
5.1 สรุปผลการดำเนินโครงงาน	74
5.2 แนวทางการพัฒนาโครงงาน	75
บรรณานุกรม	76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 1.1 ลักษณะการทำงานโดยรวมของโครงงาน	3
รูปที่ 2.1 การสื่อสารระหว่างรถยนต์	5
รูปที่ 2.2 วงจรระบบสมองกลฝังตัว	7
รูปที่ 2.3 รูปแบบการเชื่อมต่อเป็นเครือข่าย (Mesh Topology)	8
รูปที่ 2.4 การรับ-ส่งข้อมูลภายในตัวโมดูล Zigbee	10
รูปที่ 2.5 รูปแบบการเชื่อมต่อของโครงข่ายแบบ Star, Mesh, Cluster Tree	15
รูปที่ 2.6 กราฟอัตราส่วนของสัญญาณที่ได้รับเทียบกับสัญญาณรบกวน	15
รูปที่ 2.7 แสดงช่องสัญญาณและช่วงความถี่	16
รูปที่ 2.8 รายละเอียดการทำงานในชั้นเน็ตเวิร์ตเลเยอร์	16
รูปที่ 2.9 รายละเอียดการทำงานในชั้นแอปพลิเคชั่นเลเยอร์	17
รูปที่ 2.10 การหาเส้นทางจากปลายต้นทางไปยังปลายทางของชั้นเน็ตเวิร์ตเลเยอร์	17
รูปที่ 2.11 โครงสร้างโพรโทคอล ZigBee	18
รูปที่ 2.12 การหาเส้นทางโดยใช้ Ad hoc	19
รูปที่ 2.13 โหมดการทำงานของ Zigbee	23
รูปที่ 2.14 ตัวอย่างเครือข่าย Pico-Network หรือ PAN	24
รูปที่ 2.15 มาตรฐานของการสื่อสารแบบไร้สาย	26
รูปที่ 2.16 โครงสร้างของ WPAN	27
รูปที่ 3.1 การเชื่อมต่อระหว่างอุปกรณ์กับเครื่องคอมพิวเตอร์	29
รูปที่ 3.2 วงจร Zigbee	30
รูปที่ 3.3 เครื่องเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์ ET-AVR ISP ISB V 1.0	31
รูปที่ 3.4 บล็อกไดอะแกรมแสดงการทำงานของโครงงาน	32
รูปที่ 3.5 การทำงานและการกำหนดค่าให้กับโมดูล Zigbee	33
รูปที่ 3.6 โปรแกรม X-CTU	34
รูปที่ 3.7 การกำหนดค่าต่างๆเพื่อทำการเชื่อมต่อ	38
รูปที่ 3.8 ทำการกำหนดค่าการเชื่อมต่อและส่งข้อมูลระหว่างอุปกรณ์	39
รูปที่ 3.9 การกำหนดค่าเพื่อเริ่มการเชื่อมต่อระหว่างอุปกรณ์	40
รูปที่ 3.10 การเชื่อมต่อกับ AVR เบอร์ ATMEGA32	42
รูปที่ 3.11 ตำแหน่งขาสัญญาณของ ET-AVR ISP USB V1.0 โดยมองจากด้านล่าง	43
รูปที่ 3.12 วงจรเชื่อมต่อระหว่างเครื่องมือเขียนโปรแกรมกับ Embedded System	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.13 การใช้งานเครื่องมือเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์	50
รูปที่ 3.14 การรับ-ส่งข้อมูลภายในตัวโมดูล Zigbee	52
รูปที่ 3.15 แผนผังวงจร AVR + ZigBee	53
รูปที่ 3.16 วงจร AVR + Zigbee	53
รูปที่ 4.1 อุปกรณ์ที่ใช้ในการทดลอง	64
รูปที่ 4.2 เครื่องมือเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์	65
รูปที่ 4.3 อุปกรณ์ที่ใช้ทำการทดลองเชื่อมต่อกับเครื่องคอมพิวเตอร์	65
รูปที่ 4.4 การเชื่อมต่อระหว่างระบบสมองกลอัจฉริยะกับ เครื่องมือเขียนโปรแกรม	66
รูปที่ 4.5 ลักษณะการสูญเสียของสัญญาณ	67
รูปที่ 4.6 ค่าการสูญเสียของสัญญาณต่อระยะทาง	70
รูปที่ 4.7 แสดงการกำหนดค่าการเชื่อมต่อและชุดข้อมูลที่ส่ง	71
รูปที่ 4.8 แสดงข้อความเมื่ออุปกรณ์ Zigbee ทำการเชื่อมต่อเสร็จเรียบร้อยแล้ว	72
รูปที่ 4.9 แสดงผลการส่งข้อมูลระหว่างอุปกรณ์ Zigbee กับ Embedded System	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงการเปรียบเทียบคุณสมบัติของ Xbee และ Xbee-Pro	9
ตารางที่ 2.2 แสดงการกำหนดค่าของขาแต่ละขาภายใน Xbee	10
ตารางที่ 2.3 เทคโนโลยีที่สำคัญต่างๆ	28
ตารางที่ 3.1 คำสั่งที่ใช้ในการทดลองโดยทั่วไป	35
ตารางที่ 3.2 คำสั่งที่ป้อนลงไปในโมดูลทั้ง 2	39
ตารางที่ 3.3 การต่อขากับอุปกรณ์ต่างๆ	51
ตารางที่ 4.1 ค่าการสูญเสียของ MaxStream (Friis Transmission Equation)	67
ตารางที่ 4.2 ค่าการสูญเสียของคลื่นผ่านวัตถุต่างๆ	68

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 แนวคิดและที่มาของปัญหา

เนื่องจากปัจจุบันการจราจรบนท้องถนนนั้นมีความคับคั่งและมีความเสี่ยงในการเกิดอุบัติเหตุที่สูงมาก การที่จะลดปัญหาการจราจรที่คับคั่งโดยการสร้างถนนเพิ่มขึ้นนั้นก็ดูจะไม่ใช่หนทางที่จะใช้แก้ปัญหาที่ถูกต้องและคงไม่สามารถช่วยบรรเทาความแออัดของรถบนท้องถนนได้อีกต่อไปแล้ว ไม่เพียงแต่การจัดการจราจรให้มีประสิทธิภาพจะมีความสำคัญแล้วแต่ยังรวมถึงการคิดหาและพัฒนาแนวความคิดใหม่ๆที่จะช่วยลดปัญหาการจราจร การนำเอาเทคโนโลยีใหม่ๆมาผสมผสานกันเพื่อให้เกิดอุปกรณ์ที่จะช่วยให้การใช้ชีวิตในยุคปัจจุบันมีความปลอดภัยและเกิดความสะดวกรวดเร็วในการเดินทางจึงมีความสำคัญเป็นอย่างมาก ถ้าในการเดินทางไปยังที่ใดก็ตามผู้ขับสามารถติดต่อกับผู้ใช้รถในบริเวณใกล้เคียงเพื่อบอกให้ทราบถึงความต้องการ และยังช่วยผู้ใช้รถอื่นๆให้หลีกเลี่ยงอุบัติเหตุที่เกิดขึ้นจากหักเลี้ยวกะทันหัน หรือการเบรกอย่างทันทีทันใด โดยการส่งข้อมูลด้านความปลอดภัยของรถยนต์ที่บอกถึงสถานะในการขับขี่ในขณะนั้นให้กับผู้ใช้รถอื่นโดยรอบ ซึ่งจะช่วยลดอุบัติเหตุที่อาจเกิดขึ้นได้ หรืออาจช่วยให้ผู้ใช้รถที่เกิดอุบัติเหตุในที่ซึ่งห่างไกลจากชุมชนสามารถส่งข่าวขอความช่วยเหลือได้อีกด้วย

1.2 วัตถุประสงค์

จากที่ได้กล่าวไปแล้วในตอนต้นถึงความสำคัญของการศึกษาเทคโนโลยีใหม่ๆที่จะช่วยในการลดปัญหาจราจรบนท้องถนนซึ่งเป็นสาเหตุส่วนใหญ่ที่ทำให้ผู้ใช้รถเสียทั้งชีวิตและทรัพย์สิน ทั้งนี้ปริญญานิพนธ์ฉบับนี้ก็มีวัตถุประสงค์ในการศึกษาระบบสมองกลฝังตัว หรือที่รู้จักในชื่อของ Embedded System โดยได้นำเอาความรู้เรื่องนี้รวมทั้งศึกษาเพื่อออกแบบระบบการส่งผ่านข้อมูลสำหรับระบบโครงข่ายสื่อสารระหว่างรถยนต์มาประยุกต์เพื่อสร้างสิ่งที่เป็นระโยชน์ในการลดอุบัติเหตุบนท้องถนนให้น้อยลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

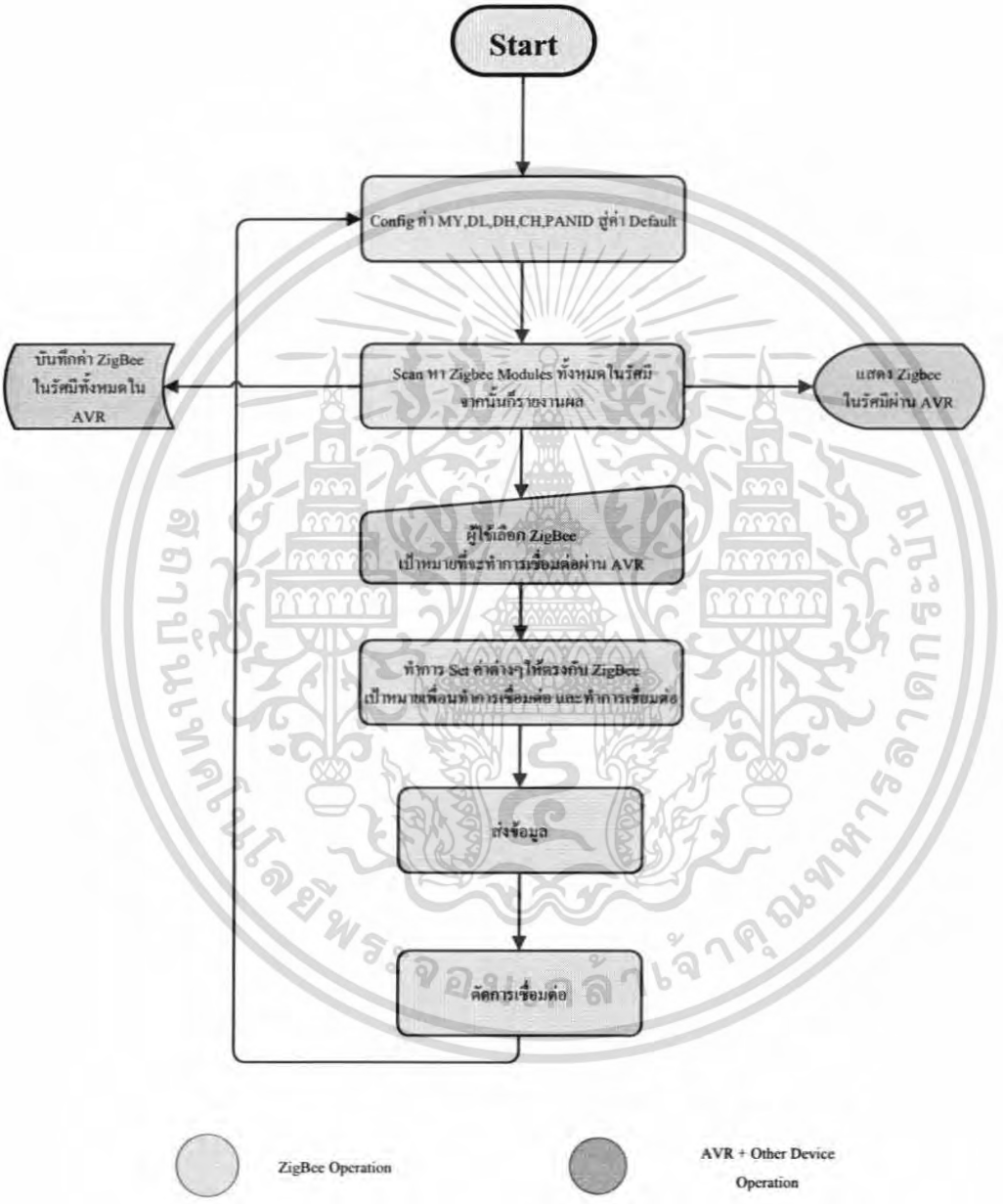
สำหรับปริญญานิพนธ์ฉบับนี้ได้มุ่งเน้นไปที่การสร้างอุปกรณ์ที่สามารถช่วยลดอุบัติเหตุบนท้องถนนได้ สามารถนำเอาความรู้เรื่องระบบสมองกลฝังตัวมาประยุกต์ใช้ในการพัฒนาระบบโครงข่ายสื่อสารระหว่างรถยนต์ สามารถนำอุปกรณ์ที่ใช้ในการสื่อสารส่งข้อมูลระหว่างกันเป็นเครือข่าย Zigbee มาประยุกต์ใช้ร่วมกับไมโครคอลโทรเลอร์เพื่อใช้ในการส่งข้อมูลระหว่างรถยนต์

1.4 ขั้นตอนการศึกษา และผลที่คาดว่าจะได้รับ

จากแนวความคิดและหลักการที่กล่าวมาแล้วข้างต้น สามารถสรุปขั้นตอนของการศึกษาได้อันดับแรกคือทำการศึกษารูปแบบและหลักการทำงานของอุปกรณ์ที่จะใช้ในการส่งข้อมูลระหว่างรถยนต์ (ZigBee) รวมทั้งคำสั่งต่างๆที่จำเป็นในการควบคุมการส่งข้อมูลให้เป็นไปตามที่ต้องการ อีกทั้งศึกษาข้อดีข้อเสียของอุปกรณ์ตัวนี้เพราะเป็นเทคโนโลยีที่กำลังได้รับความนิยม เพื่อจะได้นำมาใช้ได้หลากหลายและเกิดประโยชน์สูงสุด ขั้นต่อมาทำการศึกษารูปแบบและหลักการทำงานของอุปกรณ์ที่เลือกใช้นำมาควบคุม ZigBee นั้นคือไมโครคอลโทรเลอร์ตระกูล AVR : ATmega128 เพื่อนำมาประกอบรวมกันเป็นระบบสมองกลฝังตัวอัจฉริยะอุปกรณ์ที่ใช้ในการติดต่อสื่อสารแบบโครงข่ายระหว่างรถยนต์ รวมทั้งศึกษาถึงสภาพแวดล้อมต่างๆที่มีผลต่ออุปกรณ์เพื่อหาวิธีป้องกันความผิดพลาด เพื่อให้ได้อุปกรณ์ที่มีประสิทธิภาพสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ลักษณะโดยรวมของโครงงาน



รูปที่ 1.1 ลักษณะการทำงานโดยรวมของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 บทนำ

ตั้งแต่อดีตมนุษย์และสัตว์เดินทาง (การขนส่ง) เพื่อเป็นการเอาตัวรอดและความดำรงอยู่ของเผ่าพันธุ์ กล่าวได้ว่าการขนส่งมีความจำเป็นที่ทำให้เกิดความหลากหลายทางกิจกรรมและความอยู่รอด ปัจจุบันจุดประสงค์ของการขนส่งนั้นได้เปลี่ยนแปลงไปจากแรกเริ่มที่เป็นการขนส่งเพื่อการดำรงเผ่าพันธุ์ของสัตว์ กลายเป็นการรองรับการเจริญเติบโตทางสังคมและเศรษฐกิจ รวมทั้งยังเป็นการเชื่อมโยงผู้คนเข้าด้วยกัน ซึ่งเป็นบทบาทที่มีความสำคัญเป็นอย่างมากของการขนส่งในยุคปัจจุบัน ยิ่งกว่านั้นในศตวรรษที่ 21 โลกได้มีการวิวัฒนาเปลี่ยนแปลงไปอย่างรวดเร็วเพื่อเข้าไปสู่ยุคของเทคโนโลยีสารสนเทศซึ่งการติดต่อสื่อสารถึงกันเป็นเรื่องที่ง่ายดายและสะดวกมากขึ้น การขนส่งจะมีโฉมหน้าใหม่ที่มีความจำเป็นที่ควรจะต้องคำนึงถึงในด้านของความปลอดภัยและความสะดวกสบายของการขนส่งมากขึ้น

2.2 ระบบ ITS

โดยในการศึกษาหัวข้อนี้จะเป็นหัวข้อหนึ่งในเรื่องของระบบการขนส่งอัจฉริยะ (ITS) ซึ่งได้รับการยอมรับว่าจะเป็นส่วนหนึ่งของโครงสร้างพื้นฐานใหม่สำหรับระบบการสื่อสารในยุคใหม่ที่กำลังจะมาถึงนี้ โดยแบ่งออกเป็นหัวข้อที่เกี่ยวข้องโดยคร่าวๆได้ดังนี้ คือ

การสื่อสารระหว่างสถานีท้องถิ่นข้างถนนกับรถยนต์ (Road-to-Vehicle Communication)

การสื่อสารระหว่างรถยนต์ (Inter-Vehicle Communication)

ในส่วนของการสื่อสารระหว่างสถานีท้องถิ่นที่ถนนกับรถนั้นได้มีการนำมาปฏิบัติจริงแล้วในประเทศญี่ปุ่นที่เรียกว่า การเก็บเงินค่าผ่านทางอิเล็กทรอนิกส์ (Electronics Toll Collection) ซึ่งเป็นระบบการเก็บเงินค่าทางด่วนโดยที่ไม่ต้องมีการหยุดเพื่อจ่ายเงิน ซึ่งจะช่วยให้ผู้ใช้รถเกิดความสะดวกสบายและรวดเร็วในการเดินทาง ส่วนการสื่อสารระหว่างรถกับรถนั้นได้ถูกพัฒนาขึ้นเพื่อจะรองรับบริการต่างๆที่มีประโยชน์แก่ผู้ใช้งาน เช่น การบริการโทรศัพท์เซลลูลาร์ การบริการอินเตอร์เน็ต และรวมถึงแนวคิดที่จะนำมาพัฒนาในปริญญานิพนธ์ฉบับนี้ก็คือ การติดต่อสื่อสารส่งข้อความถึงกันระหว่างรถยนต์โดยผ่านทางการสื่อสารไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.1 การสื่อสารระหว่างรถยนต์

2.2.1 ระบบการขนส่งอัจฉริยะ และการให้บริการ

ITS และการให้บริการนั้นเป็นเทคโนโลยีที่ถูกพัฒนาขึ้นเพื่อให้บริการการแลกเปลี่ยนข้อมูลข่าวสารระหว่างผู้เดินทาง เพื่อที่จะช่วยพัฒนาและปรับปรุงประสิทธิภาพของการขนส่งและกิจกรรมในด้านต่างๆเกี่ยวกับการขนส่ง โดยผู้ขับขี่ยานพาหนะส่วนตัวจะใช้ ITS เพื่อช่วยค้นหาเส้นทางหรือหลีกเลี่ยงเส้นทางที่มีการจราจรหนาแน่นและช่วยลดการเกิดอุบัติเหตุบนท้องถนน

ในปัจจุบันได้มีการพัฒนาระบบใหม่ๆ ขึ้นเพื่อรองรับความต้องการและวิธีการในการดำเนินชีวิตในโลกยุคปัจจุบัน โดยมีการคิดค้นและออกแบบอุปกรณ์สื่อสารที่มีความสะดวกในการใช้งานและให้ความปลอดภัยกับผู้ใช้รถใช้ถนนได้มากที่สุด ซึ่งระบบที่ถูกสร้างขึ้นมาเพื่อเทคโนโลยีใหม่นี้จะประกอบด้วย

- หน่วยเคลื่อนที่สำหรับการสื่อสาร เช่น หน่วยที่ติดอยู่กับรถ โทรศัพท์เคลื่อนที่เซลลูลาร์
- การสื่อสารแบบไร้สาย เช่น การสื่อสารไร้สายในยุคที่ 3 (3-G mobile communication) และการสื่อสารวิทยุ
- การระบุตำแหน่ง เช่น ระบบดาวเทียมบอกพิกัด (Global Positioning System : GPS)
- เทคโนโลยี GIS (Global Information System)

โดยระบบที่กล่าวมานี้ได้มีส่วนช่วยให้การดำเนินชีวิตของผู้คนมีความสะดวกสบายและได้รับความปลอดภัยจากการใช้รถมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ระบบสมองกลฝังตัว (Embedded Systems)

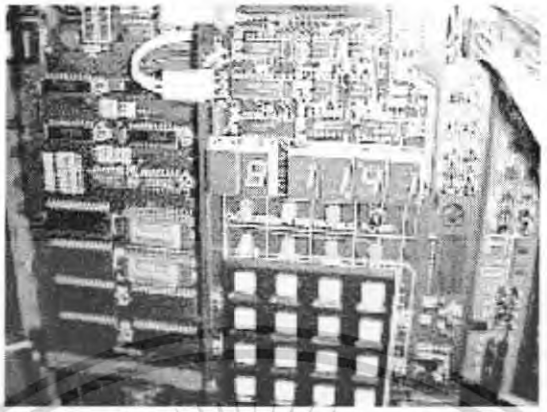
ระบบสมองกลฝังตัวเป็นระบบอิเล็กทรอนิกส์ที่ใช้สำหรับงานควบคุมรวมถึงการแสดงผลการทำงานต่าง ๆ โดยที่ระบบเหล่านี้ส่วนใหญ่จะถูกใช้เป็นส่วนหนึ่งของระบบและอุปกรณ์ควบคุมเครื่องมือ เครื่องจักรต่าง ๆ การที่ใช้คำว่า "ระบบแบบฝังตัว" เนื่องจากระบบเหล่านี้จะเป็นส่วนหนึ่งของระบบใหญ่ ในหลายกรณีที่ผู้ใช้ทั่วไปอาจไม่ทราบว่าอุปกรณ์ควบคุม เครื่องมือ เครื่องจักร รวมถึงระบบใดที่ใช้งานเป็นประจำเหล่านั้นเป็นระบบแบบฝังตัว ในบางครั้งแม้แต่ผู้ที่มีความรู้ทางด้านเทคนิคก็ไม่สามารถระบุได้แน่ชัดว่าใดมีระบบแบบฝังตัวอยู่ จนกว่าจะมีการทำงานและตรวจสอบกับระบบและอุปกรณ์ควบคุมนั้นระยะหนึ่งเลยที่เดียว

ระบบแบบฝังตัว (Embedded System) นี้แม้ไม่ใช่เครื่องคอมพิวเตอร์ แต่ก็มีระบบคอมพิวเตอร์อยู่ภายใน อาจจะเป็นเพียงไมโครโพรเซสเซอร์ (Microprocessor) หรือชิป (chip) ธรรมดาหรือโพรเซสเซอร์ ที่ประกอบด้วย ชิป ที่มีวงจรซับซ้อน โดยจะมีหลักการทำงาน คือ มีสัญญาณข้อมูลเข้า (Input) จากอุปกรณ์ เซนเซอร์ (Sensor) เข้าสู่ระบบ และมีสัญญาณผลลัพธ์ (Output) ของระบบไปควบคุมบังคับสวิตซ์เครื่องควบคุมต่าง ๆ เช่นสวิตซ์เครื่องจักร หรือ วาล์วควบคุมทิศทางการไหลของท่อทางต่าง ๆ

นอกจากนี้แบบและรุ่นของระบบแบบฝังตัว (Embedded System) ก็มีมากมายมีทั้งระบบที่เป็นแบบง่าย ๆ การทำงานไม่ซับซ้อน ตลอดจนแบบระบบที่ซับซ้อน ซึ่งขึ้นอยู่กับประเภทและจำนวนไมโครโพรเซสเซอร์ รวมถึงงานโปรแกรมควบคุมในระบบ

ระบบสมองกลฝังตัวเป็นระบบคอมพิวเตอร์ชนิดหนึ่งที่ถูกออกแบบมาเพื่อใช้เฉพาะงานนั้นๆ ไม่เหมือนกับคอมพิวเตอร์ส่วนตัว (Personal Computer หรือ PC) โดยระบบที่พัฒนาขึ้นมานั้นจะทำงานแค่ไม่กี่อย่างด้วยทรัพยากรที่เจาะจงและจำกัด และมักรวมไปถึงอุปกรณ์ทางฮาร์ดแวร์ (hardware) และ เครื่องกลต่างๆ ที่ไม่พบในการใช้งานคอมพิวเตอร์ทั่วไปและเนื่องจากมันทำงานที่เจาะจง ทำให้วิศวกรสามารถที่จะทำการปรับเปลี่ยนมันให้เหมาะสมกับงานได้ ทำให้ลดขนาด ราคา และทรัพยากรอื่นๆ ด้วย โดยระบบสมองกลฝังตัวที่ออกแบบมานั้นจะมีขนาดตั้งแต่ เครื่องเล่น MP3 ไปยังถึง ไฟจราจรเลยทีเดียว โทรศัพท์มือถือ หรือคอมพิวเตอร์พกพาก็มีส่วนที่คล้ายกับระบบสมองกลฝังตัว ดังเช่น ระบบปฏิบัติการบนเครื่องคอมพิวเตอร์ (OS) และ CPU แต่ก็ไม่ใช่ embedded system ที่แท้จริง เพราะจุดประสงค์การทำงานของอุปกรณ์เหล่านี้กว้างเกินไป เพราะสามารถที่จะเรียก application และ อุปกรณ์เสริมอื่นๆ มาใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 วงจรระบบสมองกลฝังตัว

2.3.1 ประเภทของ Embedded System

- แบบไมโครโพรเซสเซอร์เดี่ยวเป็นระบบซึ่งใช้อยู่ในอุปกรณ์ขนาดเล็ก เช่นอุปกรณ์ไฟฟ้า วงจรไฟฟ้าต่าง ๆ เครื่องตรวจจับต่าง ๆ
- แบบไมโครโพรเซสเซอร์ หลายตัวรวมกันในวงจร ซึ่งเป็นระบบซึ่งใช้อยู่ในอุปกรณ์ควบคุมที่ซับซ้อน เช่น อุปกรณ์ควบคุมการไหลของแก๊ส ของเหลว กระแสไฟฟ้า อุปกรณ์ขยายสัญญาณต่าง ๆ อุปกรณ์ปิดวาล์ว เครื่องควบคุมเครื่องจักรในโรงงาน ซึ่งจะมีทั้งที่ไม่ทำหน้าที่เกี่ยวกับเวลาและทำหน้าที่เกี่ยวกับเวลา

2.4 IEEE 802.15.4/Zigbee

2.4.1 ประวัติ ZigBee

- แนวคิดเครือข่าย ZigBee เริ่มใน ค.ศ. 1998 เมื่อผู้ผลิตพบว่าเทคโนโลยีบลูทูธ (Bluetooth) และไวไฟ (WiFi) นั้น ไม่เหมาะกับ แอปพลิเคชั่น (application) หลายตัว ทำให้วิศวกร จำนวนหนึ่งได้เห็นถึงความสำคัญของการจัดการระบบเครือข่ายได้ด้วยวิธีการ ad-hoc networks
- มาตรฐาน IEEE 802.15.4 ร่างเสร็จใน เดือนพฤษภาคม ค.ศ. 2003
- ในปี 2003 บริษัท Philips Semiconductor ได้ถอนเงินทุนสนับสนุน ZigBee แต่ทางบริษัทได้ให้เงินทุนสนับสนุน ZigBee ต่อ
- กลุ่ม ZigBee Alliance ได้ประกาศตัวในเดือนตุลาคม ค.ศ. 2004 และมีสมาชิกเพิ่มขึ้นมากกว่า 100 บริษัท 22 ประเทศ และเมื่อ เดือน เมษายน ค.ศ. 2005 ก็มีสมาชิกมากกว่า 150 บริษัท และเมื่อเดือนธันวาคม ค.ศ. 2005 ก็มีเกิน 200 บริษัทแล้ว

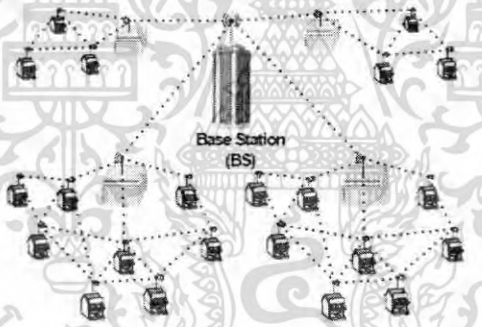
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีการร่างและพิจารณาคุณลักษณะที่ต้องมีการกำหนดไว้โดยเฉพาะ ในวันที่ 14 ธันวาคม ค.ศ. 2004

- ได้มีการประกาศคุณลักษณะเฉพาะที่ต้องมีการกำหนดขึ้น รุ่นที่ 1.0 ในวันที่ 13 มิถุนายน ค.ศ. 2005 หรือ ZigBee 2004 Specification

- กลุ่ม ZigBee Alliance ได้ประกาศถึงความเสร็จสมบูรณ์ของ ZigBee และคุณสมบัติเพิ่มเติมในรุ่นใหม่ในเดือนกันยายน ค.ศ. 2006 หรือ ZigBee 2006 Specification

ชื่อ ZigBee ได้มาจากพฤติกรรมการสื่อสารของผึ้ง โดยผึ้งจะบินแบบซิกแซ็ก และจะให้ข้อมูลข่าวสารระหว่างผึ้งด้วยกัน ที่เกี่ยวกับ ตำแหน่ง ระยะทาง และทิศทางของอาหารที่พวกมันกำลังหาอยู่ โดยที่ ZigBee ถูกสร้างขึ้นในการทำระบบเครือข่ายไร้สายส่วนบุคคล (WPAN) อยู่ภายใต้มาตรฐาน IEEE 802.15.4 โดยมาตรฐานนี้ใช้งานสำหรับการสื่อสารข้อมูลความเร็วต่ำ ใช้กำลังไฟฟ้าน้อย อุปกรณ์ราคาถูก และมีคุณสมบัติการจัดการตัวเองได้ เป็นเทคโนโลยีไร้สายที่ร่วมกันสื่อสารข้อมูลผ่านเซ็นเซอร์ขนาดเล็กจิ๋ว จำนวนเป็นพันๆ หมื่นๆ ชิ้นที่ฝังอยู่ตามส่วนต่างๆ ในอาคาร สำนักงาน โรงงาน หรือแม้แต่ในบ้าน



รูปที่ 2.3 รูปแบบการเชื่อมต่อเป็นเครือข่าย (Mesh Topology)

การทำงานของ ZigBee จะเป็นการรับ-ส่งคลื่นสัญญาณข้อมูล ผ่านชิปเล็กจิ๋วนี้จุดต่อจุดไปเรื่อยๆ จนถึงปลายทางที่ต้องการดาวน์โหลดข้อมูลลงในเครื่องคอมพิวเตอร์เพื่อใช้ในการวิเคราะห์ข้อมูล ข้อมูลที่ได้อาจจะเป็นการวัดอุณหภูมิ การเคลื่อนไหวของสิ่งมีชีวิต จับปริมาณมลพิษในอากาศ ปริมาณน้ำ ท่อแก๊สโดยใช้พลังงานแสงอาทิตย์หรือแบตเตอรี่ขนาดเล็กที่กินไฟน้อยมาก

ลักษณะของ ZigBee คือมีทางเข้าช่องสัญญาณโดยการใช้ Carrier Sense Multiple Access with Collision Avoidance (CSMA - CA) หรือมีทางเข้าช่องสัญญาณหลายๆ ทาง เพื่อหลีกเลี่ยงการชนกัน ระยะทางโดยทั่วไปประมาณ 50 เมตร มี โทโปรโลยี (Topology) เป็นแบบ star , peer-to-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

peer, mesh ทั้งนี้แต่ละอุปกรณ์จะมีแอดเดรส ที่มีความยาว 64 หรือ 16 บิต (รองรับได้ 64,000 อุปกรณ์)

ข้อกำหนดและคุณสมบัติสำคัญต่างๆเปรียบเทียบทั้ง 2 รุ่นของ Xbee

ตารางที่ 2.1 การเปรียบเทียบคุณสมบัติของ Xbee และ Xbee-Pro

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 (software selectable)	1mW (0 dBm)	60 mW (18 dBm) conducted, 100 mW (20 dBm) EIRP*
RF Data Rate	250,000 bps	250,000 bps
Serial 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)	45mA (@ 3.3 V)	If PL=0 (10dBm): 137mA(@3.3V), 135mA(@3.0V) PL=1 (12dBm): 155mA (@3.3V), 153mA(@3.0V) PL=2 (14dBm): 170mA (@3.3V), 171mA(@3.0V) PL=3 (16dBm): 188mA (@3.3V), 195mA(@3.0V) PL=4 (18dBm): 215mA (@3.3V), 227mA(@3.0V)
Idle / Receive Current (typical)	50mA (@ 3.3 V)	55mA (@ 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	Integrated Whip, Chip or U.FL Connector	Integrated Whip, Chip or U.FL Connector
Networking & Security		
Supported Network Topologies	Point-to-point, Point-to-multipoint & Peer-to-peer	
Number of Channels (software selectable)	16 Direct Sequence Channels	12 Direct Sequence Channels
Addressing Options	PAN ID, Channel and Addresses	PAN ID, Channel and Addresses
Agency Approvals		
United States (FCC Part 15.247)	OUR-XBEE	OUR-XBEEPRO
Industry Canada (IC)	4214A XBEE	4214A XBEEPRO
Europe (CE)	ETSI	ETSI (Max. 10 dBm transmit power output)*
Japan	n/a	005NYCA0379 (Max. 10 dBm transmit power output)**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดขา pin ของ Xbeeโดยที่ Xbee มีทั้งหมด 20 ขา และกำหนดค่าของแต่ละขาไว้ดังนี้

ตารางที่ 2.2 การกำหนดค่าของขาแต่ละขาภายใน Xbee

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	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	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
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

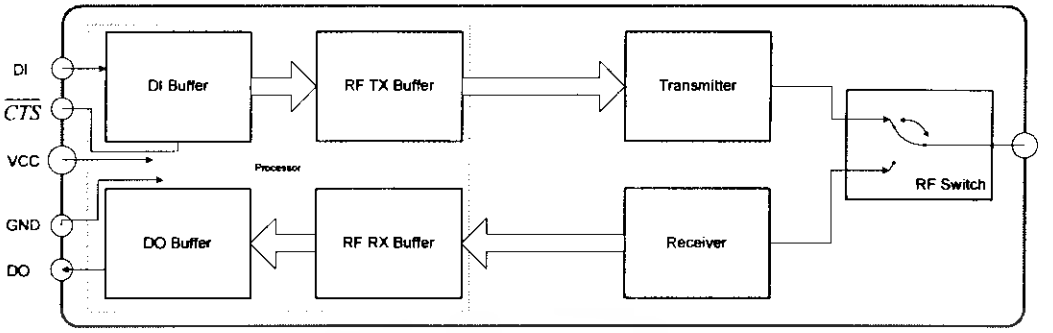
แผนภาพการส่งข้อมูลภายใน Zigbee

DI : Data IN

DO : Data Out

Antenna : สายอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การรับ-ส่งข้อมูลภายในตัวโมดูล Zigbee

2.4.2 โครงสร้างของเครือข่าย ZigBee

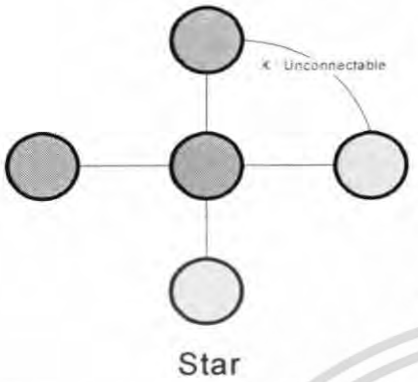
คำว่าโครงสร้างของเครือข่ายในที่นี้หมายถึงการตั้งค่าของฮาร์ดแวร์แต่ละตัวและลักษณะของการส่งข้อมูลผ่านเครือข่ายผ่านการตั้งค่านั้นๆ ระบบ ZigBee และมาตรฐาน IEEE 802.15.4 นั้นสนับสนุนเครือข่ายการเชื่อมต่อทั้งหมด 3 แบบ คือ แบบดาว (star) แบบเมช (mesh) และแบบต้นคลัสเตอร์ (cluster-tree) หรือรูปแบบผสมระหว่างแบบดาวและแบบเมชนั่นเอง ในทั้งสามรูปแบบรูปแบบต้นคลัสเตอร์นั้นมีประสิทธิภาพน้อยที่สุดและมีการนำมาใช้งานจริงน้อยมาก และจะไม่พูดถึงหรือนำมาใช้ในโครงงานนี้

2.4.3 ประเภทของอุปกรณ์ที่เชื่อมต่อกับเครือข่าย

- Coordinator ทำหน้าที่ตั้งค่าให้กับระบบ คอยวัดประสิทธิภาพการทำงานของระบบและรายงานผล
- Routers หรือเรียกอีกชื่อนึงว่า FFDs (Full functional device) ทำหน้าที่ขยายอาณาเขตของเครือข่าย จัดสรรเลือกเส้นทางที่เหมาะสม และหาเส้นทางการไหลของข้อมูลสำรองถ้าเกิดมีปัญหากับระบบ ตัวมันสามารถติดต่อกับ coordinator และ router อื่นๆได้และมีอุปรณ์ในเครือได้ด้วย
- End Devices มีอีกชื่อนึงว่า reduced function devices สามารถที่จะส่งและรับข้อความต่างๆจากระบบได้แต่ไม่สามารถที่จะส่งข้อมูลไปตมเส้นทางได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4 ลักษณะรูปแบบเครือข่ายแบบต่างๆ



แบบดาว(Star Topology)

ตัวอุปกรณ์จะส่งข้อมูลมายังตัว coordinator ซึ่งข้อมูลเหล่านั้นจะถูกส่งต่อออกไปยังจุดหมายลายทางที่กำหนดไว้ การสร้างการเชื่อมต่อระหว่างตัวอุปกรณ์เองไม่สามารถที่จะทำได้

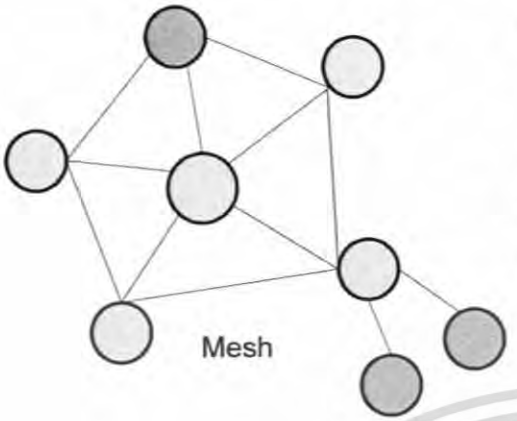
ข้อดีของระบบเครือข่ายแบบดาว

- ความเรียบง่ายในการใช้งานคือจุดเด่นของลักษณะเครือข่ายแบบนี้ ไม่ต้องใช้การบริหารที่ซับซ้อนมาก
- มีประสิทธิภาพในการใช้งานสูง โดยมากแล้วข้อมูลจะสามารถกระโดดระหว่างตัวอุปกรณ์ได้ประมาณ 2 ครั้ง

ข้อจำกัดของระบบเครือข่ายแบบดาว

- ระบบเครือข่ายแบบนี้ไม่มีเส้นทางสำรองในการไหลของข้อมูลระหว่างตัวอุปกรณ์และตัว coordinator ในเมื่อเป็นเช่นนี้เมื่อเส้นทางการไหลของข้อมูลมีอุปสรรคใดๆมาขวางกั้นการเชื่อมต่อของระบบจะหายไป
- รัศมีของเครือข่ายถูกจำกัดด้วยรัศมีการส่งของตัวอุปกรณ์ โดยส่วนใหญ่จะมีรัศมีการส่งประมาณ 100-800 เมตร
- ระบบเครือข่ายต้องถูกออกแบบมาอย่างดีเพื่อให้มีประสิทธิภาพสูงสุด ต้องออกแบบให้หลีกเลี่ยงสิ่งกีดขวาง อุปสรรค และ คลื่นรบกวนต่างๆ ระหว่างตัว coordinator กับตัวอุปกรณ์ลูก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แบบเมช (Mesh Topology)

การสื่อสารในเครือข่ายแบบเมชนั้นสนับสนุนการส่งข้อมูลแบบกระโดด (multi-hop) ซึ่งข้อมูลจะถูกส่งจากอุปกรณ์ตัวหนึ่งไปยังอุปกรณ์อีกตัวหนึ่ง โดยใช้ช่องทางการสื่อสารที่มีความเชื่อถือมากที่สุดและสั้นที่สุดจนกว่าข้อมูลจะไปถึงเส้นหมายปลายทาง

เทคนิคการทำการกระโดดข้อมูลนั้นจะช่วยลดข้อผิดพลาดที่จะเกิดขึ้นจากระบบ ถ้าเมื่อใดที่อุปกรณ์ตัวใดตัวหนึ่งมีปัญหา เสียหาย หรือไม่สามารถทำงานได้อย่างปกติ ตัวเครือข่ายจะสามารถที่จะปรับเปลี่ยนเส้นทางการไหลของข้อมูลด้วยตัวเอง โดยใช้ทรัพยากรที่เหลือของระบบเครือข่าย

ข้อดีของเครือข่ายแบบเมช

- โครงสร้างเครือข่ายในลักษณะนี้มีความเชื่อถือสูง เวลาใดที่ router ตัวใดตัวหนึ่งไม่สามารถทำงานได้ปกติ ก็จะทำการหาและใช้เส้นทางการไหลของข้อมูลสำรอง
- การใช้ตัวอุปกรณ์เป็นสื่อในการส่งผ่านข้อมูลนั้นหมายความว่ารัศมีและอาณาเขตการให้บริการของระบบนั้นสามารถเพิ่มเข้าไปได้โดยง่าย ทำให้สามารถที่จะเพิ่มประสิทธิภาพของระบบได้ง่ายด้วยเช่นกัน
- สัญญาณที่อ่อนหรือจุดอับสัญญาณสามารถที่จะแก้ไขได้โดยง่าย เพียงแค่เพิ่มตัว router เข้าไปในเครือข่ายให้มากขึ้นในบริเวณนั้น

ข้อจำกัดของระบบเครือข่ายแบบเมช

- ตัวเครือข่ายลักษณะนี้มีการส่งผ่านข้อมูลอยู่ตลอดเวลามากว่าโครงสร้างแบบดาว ซึ่งจะส่งผลให้เกิดการส่งข้อมูลล่าช้าได้(latency) และมีประสิทธิภาพการเชื่อมต่อที่ลดลง
- การจัดสรรเส้นทางการไหลของข้อมูลของระบบเครือข่ายแบบเมชนั้นมีความซับซ้อนมาก ซึ่งหมายความว่าตัว router นั้นต้องใช้ทรัพยากรของระบบ embedded มากขึ้น ซึ่งจะส่งผลให้เพิ่มปริมาณการใช้พลังงานของระบบและเพิ่มค่าใช่จ่ายด้วย

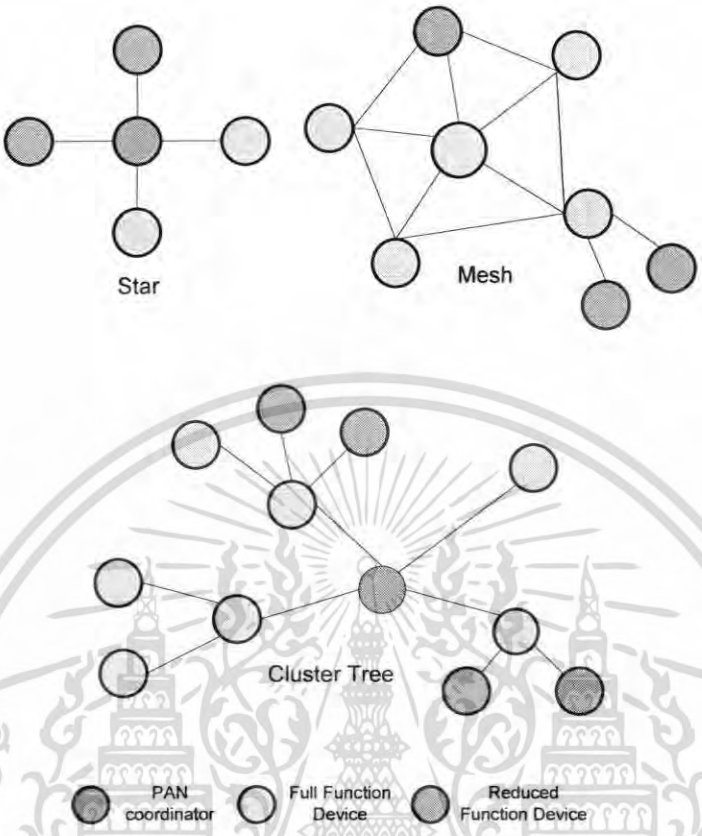
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความน่าเชื่อถือของระบบเครือข่ายแบบเมช

เมื่อใดที่เราต้องการความน่าเชื่อถือในการสื่อสารเป็นหลักหรือพื้นฐานการสื่อสาร เครือข่ายแบบเมชเป็นทางป้องกันที่ดีที่สุดเพราะเนื่องมาจากความสามารถในการรักษาตัวเองของระบบ บวกกับความยืดหยุ่นของเครือข่ายที่สามารถที่จะสร้างเส้นทางการไหลข้อมูลสำรองไว้ใช้งานเวลาเกิดปัญหาใดๆ

- ตัวเครือข่ายสามารถรักษาตนเองได้ในแง่การสร้างเส้นทางการสื่อสาร สามารถที่จะทำงานแบบไดนามิกส์ และสามารถจัดการตนเองได้เมื่อระบบเครือข่ายมีปัญหาการสร้างการเชื่อมต่อให้อุปกรณ์แต่ละตัวมีการเชื่อมต่อกับอุปกรณ์อื่นๆ สองตัวขึ้นไป (redundancy) ถ้าอุปกรณ์อันใดอันหนึ่งขาดหายไปจากเครือข่าย ตัวอุปกรณ์ข้างเคียงก็เพียงแค่หาเส้นทางใหม่เท่านั้นเอง
- ความสามารถในการตั้งค่าของเครือข่ายสามารถที่จะทำให้ระบุได้เมื่อมีอุปกรณ์ถูกเพิ่มเข้ามาในเครือข่าย เช่น ความสามารถที่จะระบุประเภทของอุปกรณ์และอุปกรณ์ที่อยู่ข้างเคียงกับอุปกรณ์ตัวนั้น และเส้นทางที่ดีที่สุดในการสื่อสารคือทางใด และหลังจากทำการตั้งค่าเรียบร้อยแล้ว
- ถ้าต้องการ เราสามารถที่จะเพิ่มคุณภาพและความน่าเชื่อถือของสัญญาณได้โดยการลดระยะทางระหว่างอุปกรณ์ไร้สายสองตัวนี้ อาจจะทำโดยการเคลื่อนย้ายอุปกณ์ตัวเดิมหรือตัวใหม่เข้าหากันเท่านั้น
- และเนื่องจากเครือข่ายแบบเมชมีการใช้พลังงานในปริมาณที่ต่ำในการสื่อสารกับอุปกรณ์ใกล้เคียงเท่านั้น จึงมีโอกาสที่จะได้รับคลื่นรบกวนจากคลื่นวิทยุอื่นๆน้อยมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

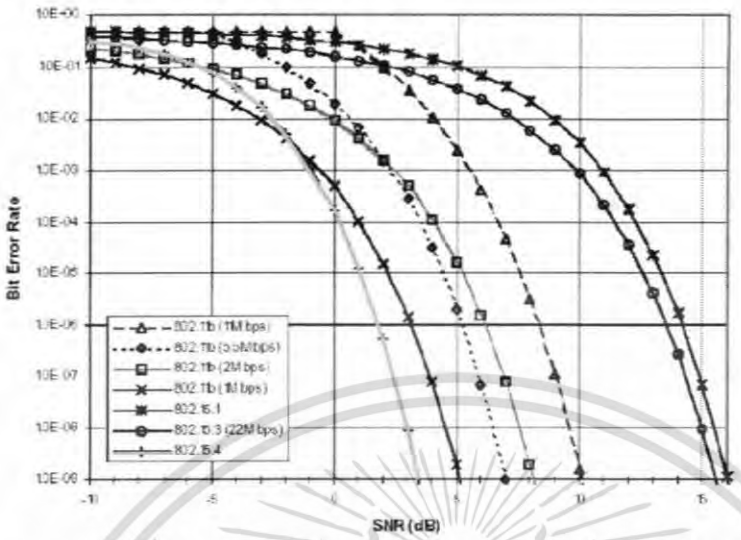


รูปที่ 2.5 รูปแบบการเชื่อมต่อของโครงข่ายแบบ Star, Mesh, Cluster Tree

2.4.5 วิธีการควบคุมหรือการเปลี่ยนแปลงช่องสัญญาณ (Modulation Technique)

เทคนิคการ มอดดูเลชั่น (modulate) อยู่บนมาตรฐาน IEEE 802.15.4 ใช้เทคนิคที่มีความแน่นอนของสัญญาณมากกว่า ซึ่งเรียกวิธีการเข้ารหัสช่องสัญญาณนี้ว่า Phase-Shift Keying(PSK) เพราะเป็นวิธีการที่พิจารณาลักษณะของเฟสของช่องสัญญาณที่ถูกส่งเข้ามาแทนการใช้เทคนิค Frequency Shift Keying(FSK) ที่เป็นการพิจารณาในส่วนของความถี่ ซึ่งมีประสิทธิภาพต่ำกว่า แต่งายและสะดวกต่อการนำไปใช้เมื่อเปรียบเทียบที่ความถี่ 2400 MHz จะถูกนำไปใช้รวมกับวิธีการอื่นซึ่งจะเรียกโดยรวมว่า Offset Quadrature PSK ในขณะที่ความถี่ต่ำกว่านี้จะใช้ Binary PSK ซึ่งเทคนิคการปรับแต่งช่องสัญญาณ ทั้งสองมีอัตราความผิดพลาดของการส่งข้อมูล (bit error rate) และอัตราส่วนของสัญญาณที่ได้รับเทียบกับสัญญาณรบกวน (signal to noise ratio) ที่ต่ำ ดังแสดงให้เห็นในรูปที่ 2.5 แสดงถึงประสิทธิภาพการใช้งานร่วมกับเทคโนโลยีอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

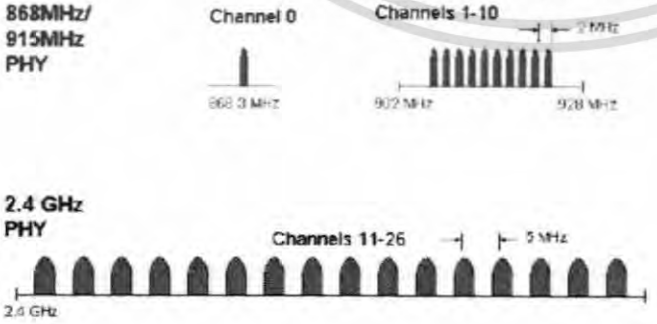


รูปที่ 2.6 กราฟอัตราส่วนของสัญญาณที่ได้รับเทียบกับสัญญาณรบกวน

การทำงานในชั้นฟิสิคัลเลเยอร์ (Physical Layer)

ฟิสิคัลเลเยอร์ (Physical Layer) ประกอบด้วยเสาสัญญาณในด้านรับ (RF Transceiver) และส่วนควบคุมสัญญาณระดับต่ำ (Low Level Control Mechanism) MAC Sublayer ช่วยในการติดต่อกับ ช่องสัญญาณในชั้นฟิสิคัลเลเยอร์ ในการสื่อสารต่างๆ

การทำงานของ Physical Layer มีสองรูปแบบคือ การรับ-ส่งข้อมูล และ การจัดการการให้บริการ โดยในเลเยอร์นี้จะสามารถที่จะ เปิด-ปิด ตัวกลางในการส่งสัญญาณคลื่นวิทยุ (radio-transceiver) , การตรวจจับสัญญาณที่ส่งผ่านมา (Energy Detection) , คุณภาพของการเชื่อมต่อของสัญญาณ (Link Quality Indication) , (channel selection) , Clear Channel Assessment , และ รับส่ง packet ผ่านทางสื่อกลางในย่านความถี่ 2.4 GHz ถึง 2.4835 GHz มีช่องทางการสื่อสารทั้งหมด 16 ช่องสัญญาณ ดังรูป

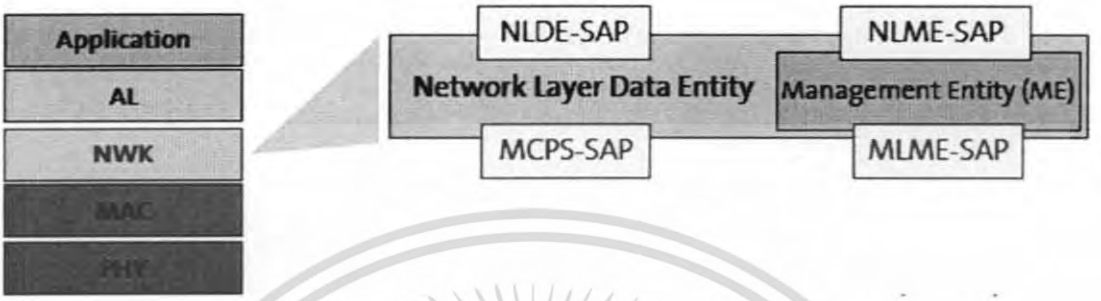


รูปที่ 2.7 ช่องสัญญาณและช่วงความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานในชั้นเน็ตเวิร์คเลเยอร์ (Network Layer)

ทำงานเกี่ยวกับ การกำหนดค่าต่างๆของระบบเนตเวิร์ก และการเลือกเส้นทาง(Routing)

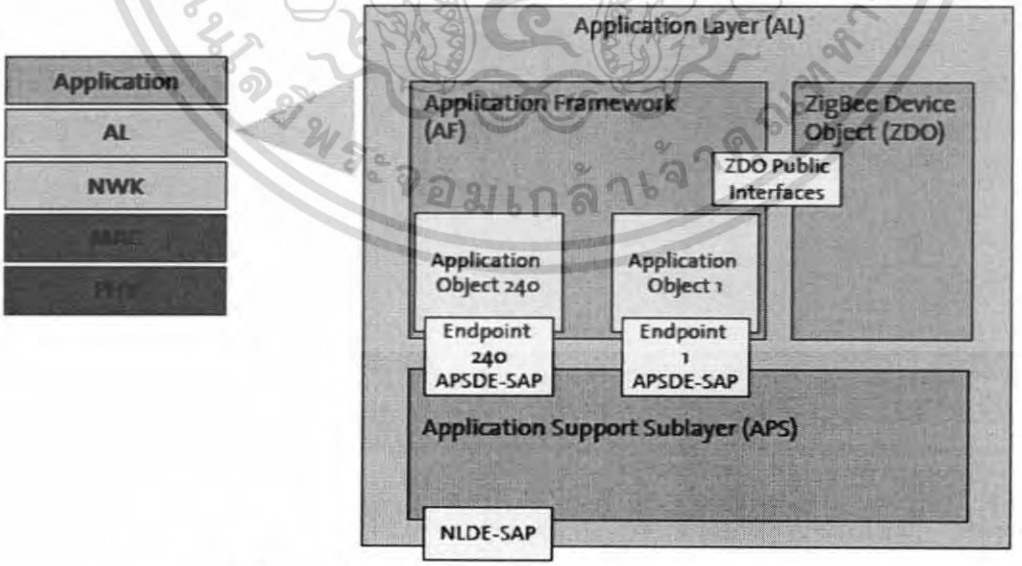


รูปที่ 2.8 รายละเอียดการทำงานในชั้นเน็ตเวิร์ตเลเยอร์

การทำงานในชั้นแอปพลิเคชั่นเลเยอร์ (Application Layer)

ทำงานเกี่ยวกับฟังก์ชัน ของอุปกรณ์นั้นๆIEEE 802.2 Logical Link Control (LLC) ใช้ในการเข้าถึง MAC Sublayer ผ่านทาง Service Specific Convergence Sublayer (SSCS)

โพรโทคอล ZigBee ถูกออกแบบมาเฉพาะในส่วนของแอปพลิเคชั่นเลเยอร์ และส่วนที่ให้การสนับสนุน และเน็ตเวิร์ต เท่านั้น แต่ใช้ MAC layer และฟิสิคัลเลเยอร์ ตามมาตรฐาน IEEE 802.15.4



รูปที่ 2.9 รายละเอียดการทำงานในชั้นแอปพลิเคชั่นเลเยอร์

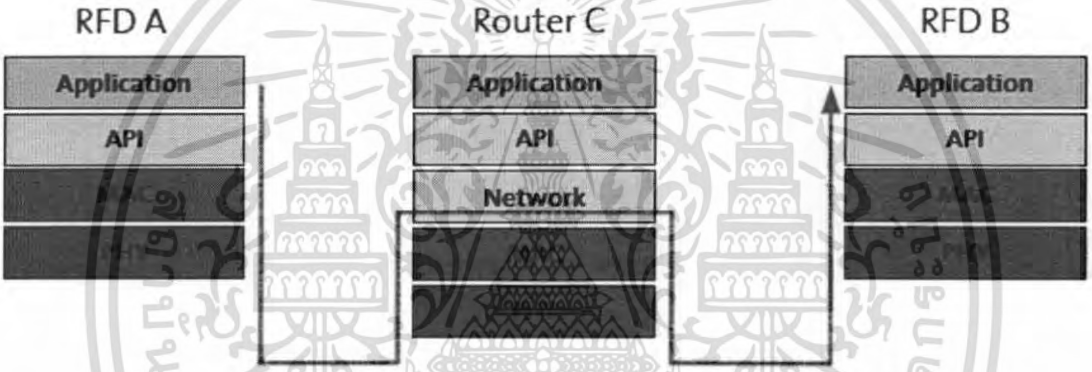
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
83152

2.4.6 โครงสร้างของโพรโทคอล ZigBee

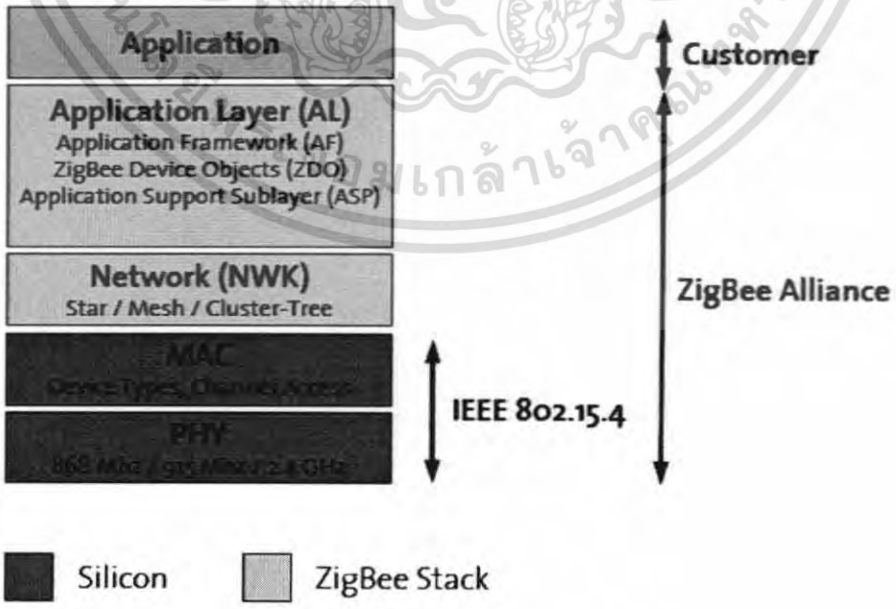
แอปพลิเคชั่นเลเยอร์เป็นชั้นที่มีส่วนของจุดปลายทางอยู่ เรียกว่า Application framework โดยมี ZigBee Device Object (ZDO) ทำหน้าที่ในการจัดการในการเข้าถึงและใช้งานแอปพลิเคชั่นเลเยอร์

Application support sub-layer ทำหน้าที่ในการสร้างเฟรมของแอปพลิเคชั่นเลเยอร์ และทำหน้าที่ในการรับส่งข้อมูลรวมถึงการจัดการด้านต่างๆ ที่เกี่ยวข้องกับแอปพลิเคชั่นเลเยอร์

เน็ตเวิร์คเลเยอร์ ทำหน้าที่ใช้ในการหาเส้นทางในการส่งข้อมูลจากต้นทางไปยังปลายทางที่อาจอยู่ภายในเครือข่ายเดียวกันหรือต่างเครือข่ายกัน



รูปที่ 2.10 การหาเส้นทางจากปลายต้นทางไปยังปลายทางของชั้นเน็ตเวิร์คเลเยอร์



รูปที่ 2.11 โครงสร้างโพรโทคอล ZigBee

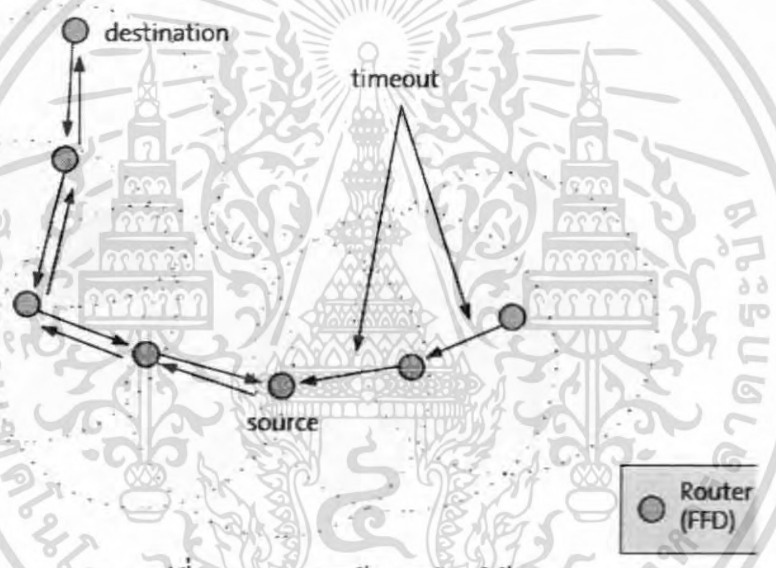
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.7 ขั้นตอนการทำงานของโพรโทคอล ZigBee

ขั้นตอนการทำงานของ ZigBee coordinator

ZigBee coordinator จะเริ่มต้นเครือข่าย โดยการตรวจสอบการใช้ช่องสัญญาณวิทยุภายในบริเวณรอบๆ ถ้ามีช่องสัญญาณที่ไม่ถูกใช้โดย coordinator ตัวอื่น ก็สามารถเริ่มต้นเครือข่ายได้ หลังจากนั้น coordinator ก็จะทำหน้าที่เป็นศูนย์กลางของเครือข่าย รองรับการเข้าร่วมเครือข่ายของ ZigBee end-device และรองรับการร้องขออื่นๆ ตามมาตรฐานด้วยเช่นกัน ในโครงงานนี้ coordinator รองรับการเข้าร่วมเครือข่าย การออกจากเครือข่าย และการร้องขอการ Binding เท่านั้น

Ad hoc On Demand Distance Vector (AODV) Algorithm:



รูปที่ 2.12 การหาเส้นทางโดยใช้ Ad hoc

Receiver Energy Detection (ED)

ค่า ED จะถูกนำไปใช้ในเน็ตเวิร์คเลเยอร์ ซึ่งจะเกี่ยวกับวิธีการในการเลือกช่องสัญญาณ (Chennnel Selection Algorithm) ค่า ED เป็นค่าของความแรงของสัญญาณ (Signal Power) ที่สามารถรับได้

Link Quality Indication (LQI)

LQI เป็นการวัดค่าความแข็ง (Strength) และ ค่าของคุณภาพของแพ็กเก็ตที่รับได้ ซึ่งจะใช้ค่า ED และค่า ในการคำนวณค่า LQI และจะถูกนำไปใช้โดยการส่งข้อมูลในชั้นเน็ตเวิร์ตเลเยอร์

Clear Channel Assessment (CCA)

- Energy above threshold เป็นการตรวจจับพลังงานของสัญญาณว่าอยู่เหนือค่าของ ED threshold หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Carrier sense only เป็นการตรวจจับว่ามีคลื่นสัญญาณที่อยู่ในมาตรฐาน IEEE 802.15.4หรือไม่

- Carrier sense above energy threshold จะตรวจจับสัญญาณในรูปแบบของ IEEE 802.15.4 และมีพลังงานอยู่เหนือค่า ED threshold

2.4.8 I/O (Input/Output)

OEM (Original equipment manufacturer)

บริษัทผู้ผลิตชิ้นส่วนที่ใช้ใส่ในเครื่องที่ผู้อื่นประกอบและขาย WSN (wireless sensor networks) เครือข่ายเซ็นเซอร์ไร้สาย

ISM (industrial, scientific and medical)

ISM band : radio band : ย่านความถี่วิทยุ

UART (Universal asynchronous receiver/transmitter)

เป็นวงจรรวมในคอมพิวเตอร์ ทำหน้าที่รับและส่งข้อมูล ทางสายอนุกรม ในการสื่อสารแบบไม่ประสานจังหวะ(asynchronous) โดยแปลงข้อมูลแบบขนานที่ผ่านไปมาภายในคอมพิวเตอร์ให้เป็นแบบอนุกรมที่อยู่ตำแหน่งที่อยู่การอ้างหรือจัดการกับตำแหน่งต่างๆบนอุปกรณ์เก็บข้อมูล

Broadcast : การส่งข่าวสารในเครือข่ายไปให้แก่ผู้ใช้ทุกคนที่ใช้เครือข่ายอยู่

Unicast or Pointcast : เป็นการกระจายข่าวสารจากเซิร์ฟเวอร์ไปยัง client ในลักษณะเจาะจงตัวเป็นแบบหนึ่งเดียว

Multicast : การสื่อสารแบบนี้แตกต่างจากแบบบรอดคาสอยู่บ้าง เพราะบรอดคาสกระจายข่าวสารทั่วทั้งเครือข่าย แต่มัลติคาสกระจายแบบเจาะจงไปยังผู้ใช้ตามที่ได้เรียกขอมา

ชนิดอุปกรณ์ของ ZigBee มีอยู่ 2 ชนิดคือ แบบอุปกรณ์ในชั้นฟิสิคัลเลเยอร์ (Physical Device) และอุปกรณ์ในชั้นโลจิคัลเลเยอร์ (Logical Device)

แบบ ฟิสิคัลเลเยอร์ มี 2 ประเภท คือ

อุปกรณ์แบบเต็มฟังชั่น (Full Function Device : FFD) เป็น เราเตอร์ที่เป็นสื่อกลางในการส่งข้อมูลจากอุปกรณ์อื่นๆ ใช้พลังงานจากกำลังงานทำงานได้ในทุกโทโปรโลยี และสามารถทำเป็นจุดเชื่อมต่อกันได้ และ

อุปกรณ์แบบลดฟังชั่น (Reduced Function Device : RFD) เหมาะแก่การเชื่อมต่อภายในเครือข่าย ใช้พลังงานจากแบตเตอรี่ ไม่สามารถถ่ายทอดข้อมูลจากอุปกรณ์อื่นๆ ได้ ทำได้ง่ายในเครือข่ายที่เป็นแบบสตาร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบโลจิคัลเลเยอร์ มี 3 ประเภท คือ

1. ZigBee Coordinators เป็นจุดที่ประสานเชื่อมต่อกัน ทำหน้าที่ในการจัดเก็บข้อมูลในเครือข่าย
2. ZigBee Routers ทำหน้าที่จัดการเส้นทางของข้อความที่ส่งผ่านภายในโครงข่ายระหว่างคู่ของโนดใด ๆ และ
3. ZigBee End Devices เป็นโนดที่อยู่ในส่วนของผู้ใช้งาน โดยสามารถเป็นได้ทั้งแบบ RFD และ FFD

การประยุกต์ใช้งาน ZigBee นั้นจะแบ่งแยกตามประเภทของข้อมูลข่าวสาร ที่มีอยู่ 3 แบบคือ ข้อมูลแบบแบ่งเป็นช่วงเวลาของข้อมูลเป็นช่วงเวลา โปรแกรมสามารถควบคุมอัตราการส่ง และตัวตรวจจับสัญญาณกระตุ้น เช็คข้อมูลและทำให้ข้อมูลไม่เคลื่อนไหว ใช้สำหรับ เซนเซอร์ และ มิเตอร์ ข้อมูลแบบ Intermittent เป็นลักษณะที่มีการส่งผ่านข้อมูลเมื่อมีการใช้งาน เช่น สวิตซ์ไฟ และข้อมูลแบบมีการทวนของสัญญาณต่ำ (Repetitive low latency) ใช้ในงานที่ต้องการทวนของสัญญาณน้อย ๆ โดยการสื่อสารจะใช้วิธีจัดสรรช่องเวลา และสามารถใช้กลไกแบบ GTS เพื่อรับประกันคุณภาพของการบริการ นำไปใช้ในงาน เช่น เมาส์ไร้สาย

2.4.8.1 ขั้นตอนการทำงานของ ZigBee end-device

ZigBee end-device จะเริ่มต้นการทำงาน โดยการร้องขอการเข้าร่วมเครือข่ายไปยัง coordinator ประจำเครือข่ายนั้นๆ โดยการตรวจสอบผ่านช่องสัญญาณต่างๆ ว่า coordinator ใช้ช่องสัญญาณใดอยู่เมื่อเข้าร่วมเครือข่ายแล้ว end-device จึงสามารถทำการร้องขอคำสั่งอื่นๆ ผ่านทาง coordinator ได้ เช่น การส่งข้อความทั่วไป (Message), การร้องขอการ Binding (Binding request), การขอออกจากเครือข่าย

นอกจากนี้ยังสามารถประยุกต์ใช้กับอุปกรณ์พื้นฐานที่หลากหลายในชีวิตประจำวัน ถ้าเรานำมาตรฐานเครือข่ายแบบไร้สาย IEEE 802.15.4 มาประยุกต์ใช้แบบยูบิควิตัส โดยเป็นการสื่อสารระหว่างอุปกรณ์ กับอุปกรณ์ หรือ อุปกรณ์ กับมนุษย์ ที่ผ่านระบบเครือข่ายไร้สาย ทั้งนี้ประโยชน์ที่ได้รับมีดังนี้

- ระบบการควบคุมอัตโนมัติ ที่บ้าน โรงงาน และ โกดังเก็บสินค้า เป็นต้น
- ระบบการติดตามสำหรับ ความปลอดภัย ชีวะอนามัย และสิ่งแวดล้อม เป็นต้น
- การตรวจหาตำแหน่งที่นำไปใช้ใน การปฏิบัติการทางทหาร การทำงานของนักผจญเพลิง และบริษัทที่ต้องการการตรวจหาตำแหน่งแบบเวลาจริง
- ให้ความบันเทิง เช่น เกมฝึกทักษะ และของเล่นแบบ interactive

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐาน 802.15.4 หรือ ZigBee ถ้ามีการใช้งานเกิดขึ้นจริงนั้นจะมีผลต่อการใช้ชีวิตประจำวันของเราอย่างมาก เช่น ในทางชีวะอนามัย นอกจากนี้ระบบ 802.15.4 จะช่วยเตือนภัยจากสิ่งแวดล้อม รวมถึงอุบัติภัยต่างๆ เช่น ไฟไหม้
น้ำท่วม แผ่นดินไหว เป็นต้น ทั้งนี้ระบบเตือนภัยในปัจจุบันไม่ได้เชื่อมต่อกันเป็นระบบเครือข่าย และตัวอุปกรณ์เองมีช่วงการใช้งานจากแบตเตอรี่สั้น นอกจากนี้ยังมีราคาแพงอีกด้วย แต่ในระบบ 802.15.4 สามารถใช้ได้กับอุปกรณ์พื้นฐานเช่น เซนเซอร์ และ Actuators ที่มีราคาถูก ทำให้สามารถติดตามเหตุการณ์ต่างๆ และอุปกรณ์จะทำงานอย่างอัตโนมัติตามที่เราต้องการ
สำหรับการประยุกต์ระบบ 802.15.4 มาใช้ภายในบ้านจะสามารถทำให้อุปกรณ์ และเครื่องใช้ไฟฟ้าที่มาจากหลากหลายผู้ผลิต สื่อสารระหว่างกันเป็นระบบเครือข่ายได้ เช่น เมื่อมีสายโทรศัพท์เข้ามา โทรทัศน์จะลดเสียงลงอัตโนมัติ ทั้งนี้เราเองไม่ต้องเซตค่าต่างๆให้กับอุปกรณ์ แต่อุปกรณ์เองจะเรียนรู้เองจากพฤติกรรมของเราภายในบ้าน โดยแต่ละอุปกรณ์จะตรวจจับค่าต่างๆ เช่น ความเข้มแสงหลอดไฟ อุณหภูมิ เพลง ช่องโทรทัศน์ และ Web Site โดยแต่ละอุปกรณ์จะเรียนรู้รสนิยมของแต่ละคนและบันทึกไว้

นอกจากนี้เราสามารถนำระบบ 802.15.4 มาประยุกต์ใช้ในการคมนาคม กับอุปกรณ์พื้นฐานต่างๆที่อยู่ตามท้องถนนทางด่วน และที่อื่น ๆ ทั้งนี้อุปกรณ์ต่างๆจะสื่อสารกันเองเป็นระบบเครือข่าย ในระหว่างการเดินทางของรถบนท้องถนน
อุปกรณ์ที่อยู่ข้างทางจะส่งข้อมูลที่จำเป็นในการเดินทางสำหรับถนนที่รถวิ่งอยู่ เช่น ความเร็วสูงสุดที่วิ่งได้ เส้นทางเป็นรถเดินทางเดียวหรือสองทาง สภาพการจราจร ข้อมูลอุบัติเหตุ เป็นต้น
นอกจากนี้อุปกรณ์ต่างๆที่อยู่ข้างถนนจะมีการทำงานแบบอัตโนมัติด้วยเช่น ไฟส่องทางจะลดความเข้มลงเมื่อไม่มีรถวิ่งผ่านมา และระบบควบคุมการจราจร เป็นต้น
ประโยชน์อีกอย่างหนึ่งคือ สามารถตรวจหาตำแหน่งของรถได้ ซึ่งจะคล้ายกับระบบ GPS แต่ระบบ GPS นี้ไม่สามารถตรวจหาตำแหน่งในบางสถานที่ได้ เช่นในอุโมงค์ ภายในอาคาร เป็นต้น และระบบ GPS ยังมีความคลาดเคลื่อนของตำแหน่งมากกว่า การใช้งานในระบบ 802.15.4 อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.9 การส่งข้อมูลของ Zigbee

ระบบเครือข่ายของ Zigbee

1. NonBeacon เป็นระบบเครือข่ายแบบ Peer-to-Peerโดยไม่มีความสัมพันธ์ของ master/server แต่ละ โมดูลจะเป็นทั้งผู้ใช้งานและเซิร์ฟเวอร์

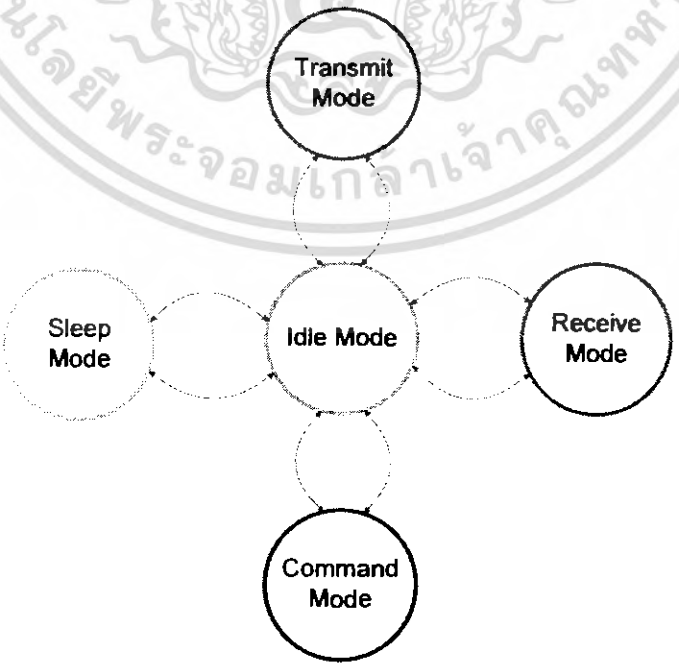
2. NonBeacon (w/ Coordinator) การทำงานในระบบนี้จะใช้โครงสร้างของการรับส่งโดยตรงหรือโดยอ้อม (direct or indirect Transmissions)

การส่งข้อมูลแบบRFของแต่ละแพ็กเกจในส่วนของเฮดเดอร์ จะประกอบไปด้วยที่อยู่ของผู้ส่งทางด้านต้นทางและที่อยู่ของผู้รับทีปลายทาง โดยที่ IEEE802.15.4 จะมีโครงสร้าง 2 แบบ นั่นคือ แบบจำนวนบิตที่อยู่น้อยคือ 16 บิต และแบบยาวคือ 64 บิต ซึ่งจะสามารถอ่านคำสั่ง SL (Serial Number Low) และ SH (Serial Number High) และการส่งข้อมูลแบบRFจะส่งได้ 2โหมด คือ Unicast Mode และ Broadcast Mode

การส่งแพ็คเกจโดยใช้โครงสร้าง 16 บิตให้ตั้งค่าตัวแปร DL (Destination Address Low) ให้เท่ากับ ตัวแปร MY และตั้งค่าตัวแปร DH (Destination Address High) เป็น '0'
การส่งแพ็คเกจโดยใช้โครงสร้าง 64 บิต ให้ตั้งค่าที่อยู่ผู้รับที่ปลายทาง (DL + DH) ให้เข้ากับค่าของที่อยู่ของผู้ส่งทางด้านต้นทาง Address (SL + SH) ของปลายทางที่เราจะส่งแพ็คเกจไป

2.4.10 การทำงานของ Zigbee

การทำงานของ Zigbee แบ่งได้ออกเป็น 5 โหมด ดังภาพต่อไปนี้



รูปที่ 2.13 โหมดการทำงานของ Zigbee

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Idle Mode เป็นโหมดที่ไม่มีการรับส่งข้อมูล และเป็นโหมดกลางที่สามารถเปลี่ยนไปยังโหมดต่างๆได้

Transmit Mode มีการส่งข้อมูลได้สองวิธี

- Direct Transmission – ข้อมูลทั้งหมดจะถูกส่งไปยังที่อยู่ของผู้รับทางด้านปลายทางทันที
- Indirect Transmission – packetจะถูกเก็บไว้จนกว่าจะถึงเวลาส่งเท่านั้น และจะส่งไปยังที่มีการตอบรับมา (Source Address = Destination Address)

Receive Mode ข้อมูล RF จะถูกรับทางสายอากาศ

Sleep Mode RF อยู่ในสถานะที่มีการใช้กำลังไฟฟ้าต่ำหรือไม่มีการใช้ การเข้ามาอยู่ในโหมดนี้นั้นจะต้องเป็นไปตามเงื่อนไขต่อไปนี้อย่างน้อยหนึ่งอย่าง (ค่าของตัวแปรSMต้องไม่เป็น0)

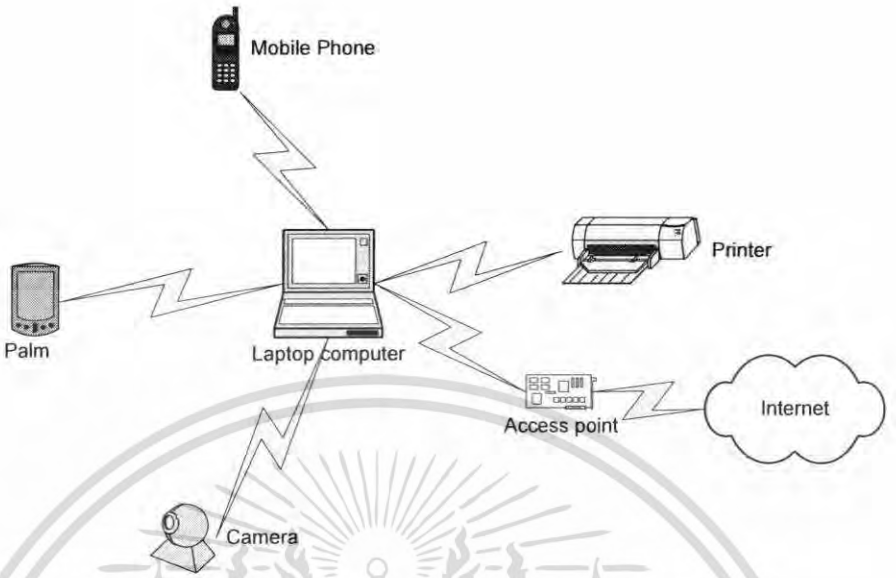
- มีการใช้งานที่ Sleep_RQ (pin 9)
- อยู่โหมด idle (ไม่มีการรับส่งข้อมูล)เป็นเวลานานมากกว่าที่กำหนดไว้ที่ตัวแปร ST (Time before Sleep)

Command Mode เป็นโหมดคำสั่งโดยจะใช้ลำดับเป็นสำคัญ

รูปแบบเครือข่ายตามมาตรฐาน IEEE 802.15.4 แบ่งออกเป็น 2 แบบ ได้แก่ แบบดาว และแบบตาข่ายร่างแห มีตัวอย่างการสร้างเครือข่ายดังรูปที่ 1 โดยในแต่ละเครือข่าย จะต้องมี FFD 1 ตัว ทำหน้าที่เป็นศูนย์กลางของเครือข่าย เรียกว่า PAN coordinator และ RFD จะเข้าร่วมเครือข่ายกับ PAN coordinator ประจำเครือข่ายนั้นๆ โดยที่ PAN นั้นเป็นส่วนหนึ่งของการส่งข้อมูลที่ถูกแบ่งออกตามระยะทางในการสื่อสารเป็น 3 แบบดังนี้

1. PAN ย่อมาจาก Personal Area Network หรือบางครั้งก็เรียกว่า Personal Area Connectivity (PAC) เมื่อเติม W แทน Wireless ข้างหน้าเป็น WPAN คือ อุปกรณ์โทรคมนาคมส่วนบุคคลที่ติดต่อสื่อสารในระยะใกล้ไม่เกิน 10 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 ตัวอย่างเครือข่าย Pico-Network หรือ PAN

2. LAN ย่อมาจาก Local Area Network เราคงได้ยินกันบ่อยๆ เมื่อเติม W แทน Wirelesข้างหน้าเป็น WLAN คือ คือ อุปกรณ์โทรคมนาคมที่ติดต่อสื่อสารในระยะกลางหรือในระดับท้องถิ่น ไม่เกิน 100 เมตร ลักษณะการเชื่อมต่อเครือข่ายคอมพิวเตอร์ถึงกันทั้งหมด จึงมีการแบ่งแยกเครือข่ายเป็นการเชื่อมโยงเครือข่ายภายในพื้นที่ใกล้ ๆ กัน เรียกว่า LAN (Local Area Network) และการเชื่อมโยงระยะไกล ที่เรียกว่า WAN (Wide Area Network) เครือข่าย LAN เป็นเครือข่ายที่เชื่อมโยงกันในพื้นที่ใกล้เคียงกัน เช่นอยู่ในอาคารเดียวกัน สามารถ ดูแลได้เอง การเชื่อมโยงเครือข่าย LAN ที่นิยมใช้กันมี 2 รูปแบบ ดังนี้

เครือข่าย LAN แบบอีเทอร์เน็ต มีการรับส่งข้อมูลด้วยความเร็ว 10-100 Mbps. มีพื้นฐานรูปแบบการเชื่อมโยงร่วมกันแบบบัส คือ ทุกอุปกรณ์จะเชื่อมต่อกันบนสายสัญญาณเส้นเดียว ดังนั้นการรับส่งต้องมีการจัดการไม่ให้รับส่งพร้อมกันเกินกว่าหนึ่งคู่ ขบวนการรับส่งข้อมูลจึงถูกกำหนดขึ้น โดยให้อุปกรณ์ที่จะส่งข้อมูลตรวจสอบว่ามีข้อมูลใดวิ่งอยู่บนสายหรือไม่ หากไม่มีจึงส่งได้ และถ้ามีการชนกันของข้อมูลบนสายก็จะส่งใหม่ การหลีกเลี่ยงการชนกันจึงกระทำได้ในเครือข่ายระยะใกล้ เครือข่าย LAN แบบโทเก็นริง มีความเร็ว 16 Mbps. เชื่อมต่อกันเป็นวงแหวนโดยแพ็กเก็ตข้อมูลจะวิ่งวนในทิศทางใดทางหนึ่ง ถ้ามีแอดเดรสปลายทางเป็นของใคร อุปกรณ์นั้นจะรับข้อมูลไป การจัดการรับส่งข้อมูลในวงแหวนจึงเป็นไปอย่างมีระเบียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครือข่าย LAN ที่อยู่ในมาตรฐานเดียวกันสามารถเชื่อมโยงเข้าหากัน แต่ทุกตัวจะมีแอดเดรสประจำ และแอดเดรสเหล่านี้จะซ้ำกันไม่ได้ โดยปกติผู้ผลิตอุปกรณ์เชื่อมโยงเครือข่ายได้กำหนดแอดเดรสเหล่านี้มาให้แล้ว เพื่อจะให้เชื่อมโยงเครือข่ายต่างมาตรฐานกันได้นั้น มีวิธีการพัฒนาให้ระบบสามารถนำแพ็กเก็ต เฉพาะของเครือข่ายมาใส่ในแพ็กเก็ตกลางที่เชื่อมโยงระหว่างกันได้ เช่น TCP/IP ตัวอย่าง เช่น ถ้าต้องการเชื่อมเครือข่าย LAN หลาย ๆ เครือข่ายเข้าด้วยกันให้เป็นเครือข่ายเดียวกัน เครือข่ายอีเทอร์เน็ตมีแพ็กเก็ตเฉพาะเมื่อจะส่งออก ก็นำแพ็กเก็ตเฉพาะมาเปลี่ยนถ่ายลงในแพ็กเก็ต TCP/IP แล้วส่งต่อ.. แพ็กเก็ต TCP/IP จึงเป็นแพ็กเก็ตกลางที่พร้อมรับแพ็กเก็ตย่อยอื่นได้ ดังนั้นการเชื่อมต่อระหว่างเครือข่าย เช่น อีเทอร์เน็ตในปัจจุบันจึงเกิดขึ้นได้

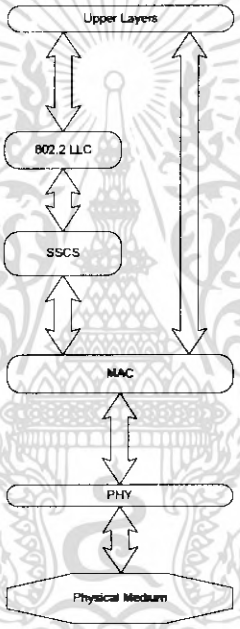
รูปที่ 2.15 มาตราฐานของการสื่อสารแบบไร้สาย

3. WAN ย่อมาจาก Wide Area Network หรือบางท่านเรียกว่า MAN หรือ Metropolitan Area Network และเมื่อเติม W แทน Wireless ข้างหน้าเป็น WWAN คือ อุปกรณ์โทรคมนาคมที่ติดต่อสื่อสารในระยะไม่เกิน 10 กิโลเมตร เครือข่าย WAN เป็นเครือข่ายเชื่อมโยงกันในระยะทางที่ห่างไกล อาจจะเป็นหลาย ๆ กิโลเมตร ดังนั้นความเร็วในการเชื่อมโยงระหว่างกันอาจไม่สูงมากนัก เพราะระยะทางไกลทำให้มีสัญญาณรบกวนได้สูง ความเร็วจึงอยู่ในระดับช่วง 9.6-64 Kbps และ 1.5-2 Mbps ขึ้นอยู่กับแอพพลิเคชั่นและขนาดของข้อมูล ทั้งเครือข่ายแบบ LAN และ WAN ล้วนแล้วแต่ใช้หลักการของแพ็กเก็ตสวิตชิ่ง กล่าวคือ มีการกำหนดวิธีการรับส่งข้อมูลเป็นแพ็กเก็ต โดยให้แต่ละอุปกรณ์มีแอดเดรสประจำ วิธีการรับส่งมีได้หลากหลาย เราเรียกวิธีการว่า "โปรโตคอล (Protocol)" ดังนั้นจึงมีมาตรฐานการเชื่อมโยงระยะไกลมีการกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดเดรส เช่นในเครือข่าย X.25 ข้อมูลจากที่หนึ่งส่งเป็นแพ็กเก็ตส่งต่อไปยังปลายทางได้ ข้อมูลเป็นแพ็กเก็ตจากจุดเริ่มต้น มีแอดเดรสกำกับตำแหน่งปลายทางและตำแหน่งต้นทาง แอดเดรส เหล่านี้เป็นรหัสที่รับรู้ได้ อุปกรณ์สวิตช์จะเลือกทางส่งไปให้ หากมีปัญหาใดทำให้ปลายทางรับได้ไม่ถูกต้อง เช่นมีสัญญาณรบกวน ระบบจะมีการเรียกร้องให้ส่งให้ใหม่ เพื่อว่าการรับส่งข้อมูลจะต้องถูกต้องเสมอ ระบบการโต้ตอบเหล่านี้จึงเป็นมาตรฐานที่กำหนดของเครือข่ายนั้น ๆ

LR - WPAN Device Architecture



รูปที่ 2.16 โครงสร้างของ WPAN

2.4.11 เทคโนโลยีไร้สายทั้งสี่ชนิด (WiMax, Mobile-Fi, ZigBee และ Ultrawideband)

WiMax จะมีลักษณะคล้ายกับการทำงานของ Wi-Fi ที่สร้าง hot spot หรือจุดรับคลื่นในบริเวณใดบริเวณหนึ่งรอบๆ เสาสัญญาณ โดยใช้แล็ปทอปหลายเครื่องพร้อมกันได้คล้ายๆ กับระบบ LAN แต่ไร้สาย WiMax แตกต่างกับ Wi-Fi ตรงที่ WiMax จะมีรัศมีการส่งคลื่นสัญญาณได้ไกลกว่าหลายสิบเท่า ว่ากันว่ามีรัศมีการส่งคลื่นได้ถึงประมาณ 40-50 กิโลเมตรเลยทีเดียว

ในขณะที่ **mobile-Fi** ก็กำลังพัฒนาให้ลูกค้า บรอดแบนด์สามารถใช้งานอินเทอร์เน็ตความเร็วสูงในขณะเคลื่อนที่อยู่บนรถยนต์หรือรถไฟได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ **ultrawideband** ได้ถูกออกแบบมาให้เป็นเทคโนโลยีไร้สายที่สามารถสนับสนุนการรับส่งไฟล์ขนาดใหญ่ได้ในระยะทางสั้นๆ เช่น การรับส่งไฟล์ภาพยนตร์ภายในบ้านจากเครื่องคอมพิวเตอร์ตั้งโต๊ะไปที่ทีวีดิจิทัล หรือบนรถไฟที่สนับสนุนการใช้เทคโนโลยี mobile-Fi แล้วมีการรับส่งไฟล์ขนาดใหญ่ภายในรถไฟได้

สำหรับ zigbee เป็นเทคโนโลยีไร้สายที่ร่วมกันสื่อสารข้อมูลผ่านเซ็นเซอร์ขนาดเล็กจิ๋วจำนวนเป็นพันๆ หมื่นๆ ชิ้นที่ฝังอยู่ตามส่วนต่างๆ ในอาคาร สำนักงาน โรงงาน หรือแม้แต่ในบ้าน การทำงานของมันจะเป็นการรับ-ส่งคลื่นสัญญาณข้อมูล ผ่านชิปเล็กจิ๋วนี้จุดต่อจุดไปเรื่อยๆ จนถึงปลายทางที่ต้องการดาวน์โหลดข้อมูลลงในเครื่องคอมพิวเตอร์เพื่อใช้ในการวิเคราะห์ข้อมูล ข้อมูลที่ได้อาจจะเป็นการวัดอุณหภูมิ การเคลื่อนไหวของสิ่งมีชีวิต จับปริมาณมลพิษในอากาศ ปริมาณน้ำท่อแก๊สโดยใช้พลังงานแสงอาทิตย์หรือแบตเตอรี่ขนาดเล็กที่กินไฟน้อยมาก จึงสามารถฝังทิ้งไว้ในที่ห่างไกลได้เป็น 10 ปี ว่ากันว่าเทคโนโลยี zigbee นี้จะช่วยทำให้บริษัทที่เกี่ยวข้องกับการส่งพลังงานเช่น น้ำมัน ประปา น้ำในเขื่อน ท่อแก๊ส สามารถประหยัดการสูญเสียได้อย่างน้อย 10-15% และในอนาคตอันใกล้นี้เมื่อเทคโนโลยีนาโนก้าวหน้ามากขึ้น เซ็นเซอร์ zigbee จะมีขนาดเล็กเท่าหัวเข็มหมุด สามารถฝังได้แม้กับในร่างกายของสิ่งมีชีวิตก็ได้

ตารางที่ 2.3 เทคโนโลยีที่สำคัญต่างๆ

Technology	Standard	Application	Coverage (m)	Frequency (GHz)
UWB	802.15.3a	Wireless PAN	10	Flexible
Bluetooth	802.15.1	Wireless PAN	10	2.4
Zigbee	802.15.4	Wireless PAN	10	Not identified
WiFi	802.11a	Wireless LAN	100	5
	802.11b	Wireless LAN	100	2.4
	802.11g,n	Wireless LAN	100	2.4
WiMAX	802.16d	Wireless MAN	6400 - 9600	11
	802.16e	Mobile Wireless MAN	1600 – 4800	2 – 6
WCDMA	IMT-2000 (3G)	Wireless WAN	1600 - 8000	1.8, 1.9, 2.1
cdma2000	IMT-2000 (3G)	Wireless WAN	1600 - 8000	0.4, 0.8, 0.9, 1.7, 1.8, 1.9, 2.1
MBWA	802.20	Mobile Wireless WAN	4000 - 12000	3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบโครงงานทาง ฮาร์ดแวร์ และ ซอฟท์แวร์

3.1 คำนำ

ในบทนี้จะกล่าวถึงการออกแบบ ขั้นตอนการคิดและการรวบรวมข้อมูลซึ่งจะแบ่งออกเป็นการออกแบบโครงงานทางด้านฮาร์ดแวร์และซอฟท์แวร์ โดยเมื่อเสร็จสิ้นการจัดทำโครงงานสำเร็จแล้วจะได้เป็นระบบสมองกลฝังตัวที่สามารถสื่อสารส่งข้อมูลระหว่างรถยนต์ได้ ในส่วนของฮาร์ดแวร์ เราจะทำการออกแบบวงจรที่สามารถรองรับการใช้งานและเหมาะสมกับจุดประสงค์การใช้งาน เป็นวงจรที่สามารถเข้าใจได้โดยง่าย จากนั้นในส่วนของซอฟท์แวร์ เราจะทำการออกแบบโปรแกรมควบคุมการทำงานส่วนต่างๆของวงจรที่เราได้ออกแบบมุ่งเน้นไปตรงจุดที่เกี่ยวข้องกับการสื่อสารส่งข้อมูลระหว่างรถยนต์

สำหรับในเทอมนี้การเชื่อมต่อระหว่างอุปกรณ์ทั้ง 2 ด้านซึ่งเป็นตัวแทนของการรับ-ส่งข้อมูลด้านส่งและด้านรับ ได้ทำการทดลองรับ-ส่งสัญญาณข้อมูลเป็นรูปแบบตัวอักษรโดยผ่านทางเครื่องคอมพิวเตอร์เพื่อตรวจสอบถึงคำสั่งและวิธีการที่จำเป็นในการใช้ในการรับ-ส่งข้อมูลระหว่างอุปกรณ์ทั้ง 2 ตัว โดยใช้โปรแกรมที่ทางด้านผู้ผลิตฮาร์ดแวร์ Zigbee พัฒนาขึ้นเพื่อตรวจสอบการรับ-ส่งของข้อมูลระหว่างอุปกรณ์ Zigbee ว่าสามารถใช้งานได้จริง โปรแกรมที่กล่าวมานี้คือ X-CTU ที่จะได้กล่าวถึงในส่วนของการออกแบบซอฟท์แวร์

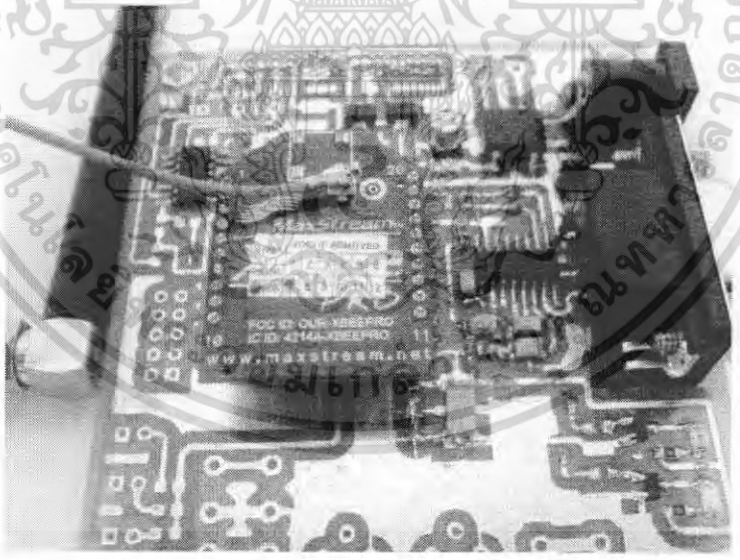
รูปที่ 3.1 การเชื่อมต่อระหว่างอุปกรณ์กับเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบโครงงานทางด้านฮาร์ดแวร์

สำหรับการออกแบบวงจรที่ใช้ควบคุมการส่งข้อมูลสื่อสารระหว่างรถยนต์นั้น ต้องพิจารณาถึงปัจจัยที่เกี่ยวข้องหลายประการดังเช่น

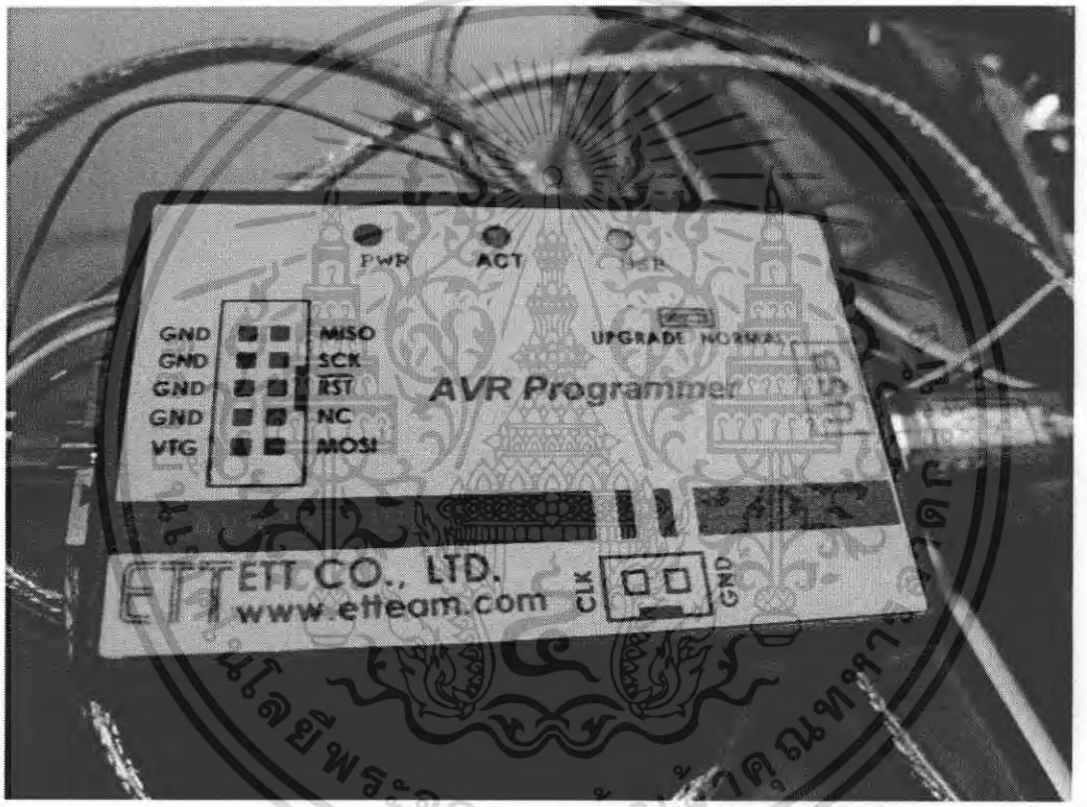
- จำนวนรถยนต์ที่ต้องการให้ติดต่อสื่อสาร
- ลักษณะหรือรูปแบบการสื่อสารที่เหมาะสม เช่น การติดต่อสื่อสารควรใช้แบบ star หรือการติดต่อสื่อสารแบบ mesh
- มีความจำเป็นหรือไม่ที่ต้องตรวจสอบผู้ที่จะเข้ามาสื่อสาร ก่อนทำการอนุญาตให้ทำการเชื่อมต่อได้
- ควรเลือกไมโครคลอโทรเลอร์ที่ใช้ควบคุมการสื่อสารระหว่างรถยนต์เป็นชนิดใด ที่จะเกิดความเหมาะสม และง่ายต่อผู้พัฒนา และผู้ที่ต้องการศึกษาเรียนรู้
- ขนาดของอุปกรณ์ที่จะทำการติดตั้งบนรถยนต์ เหมาะสมกับการใช้งานจริง
- อุปกรณ์ที่ได้มีความคงทน และใช้งานได้ง่าย ไม่ยุ่งยาก เรียนรู้ได้ง่ายและสะดวกต่อผู้ใช้งาน
- ราคาโดยรวมของอุปกรณ์มีความเหมาะสม



รูปที่ 3.2 โมดูล Zigbee

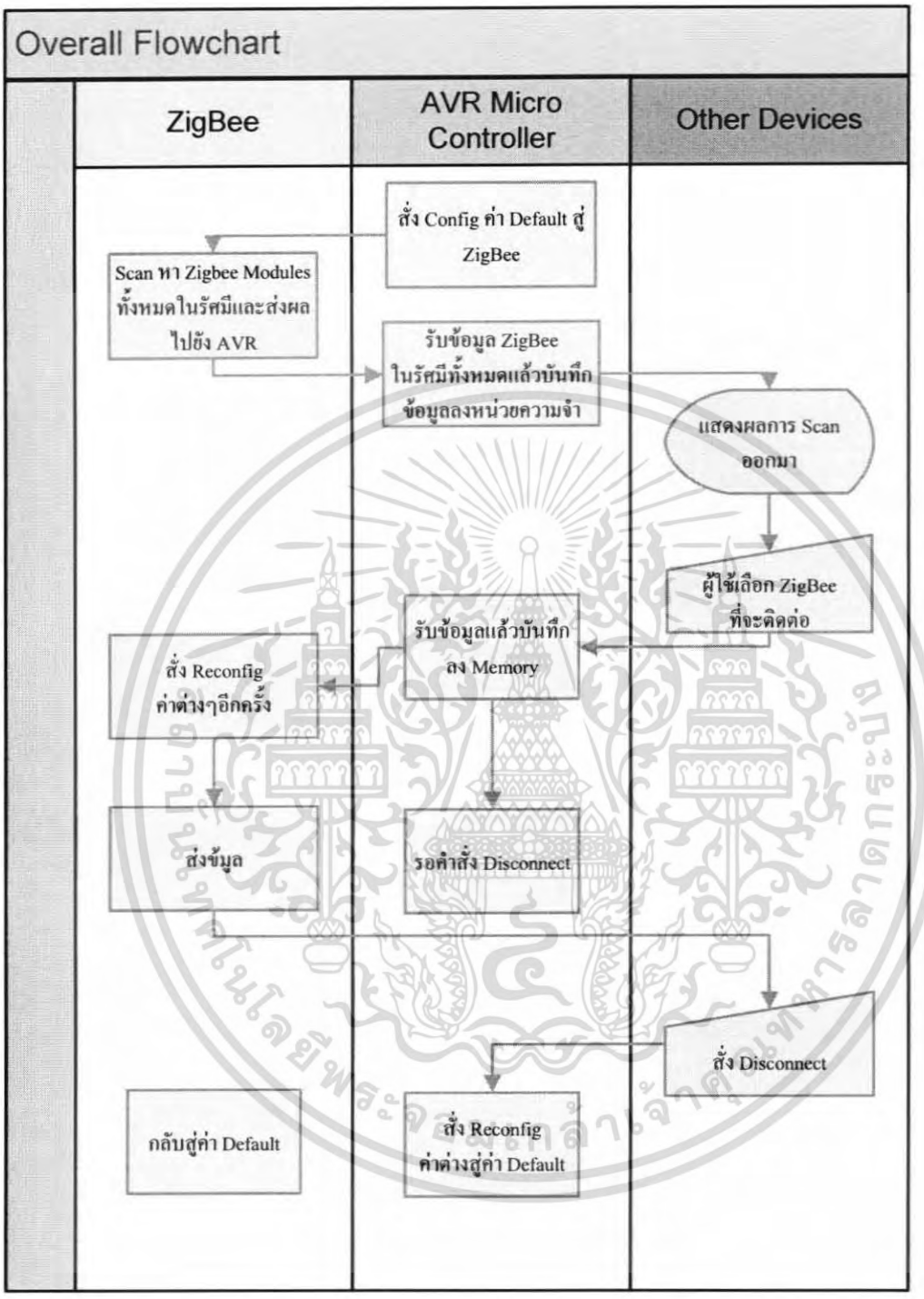
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ยังได้มีการนำอุปกรณ์ที่ช่วยในการเขียนโปรแกรมที่เราออกแบบลงบนไมโครคอลโทรเลอร์ AVR โดยผ่านทางโปรแกรมที่ได้ออกแบบไว้แล้วทำให้เราสามารถควบคุมอุปกรณ์ Zigbee ได้โดยไม่ต้องอาศัยเครื่องคอมพิวเตอร์ ซึ่งก็คือหลักการของระบบสมองกลอัจฉริยะนั่นเอง โดยอุปกรณ์ดังกล่าวนี้จะมีให้เลือกได้ตามความเหมาะสมกันความต้องการในการใช้งานของผู้ใช้ โดยผู้ผลิตจะมีคู่มือแนะนำการใช้งาน และโปรแกรมติดตั้งมาให้



รูปที่ 3.3 เครื่องเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์ ET-AVR ISP ISB V 1.0

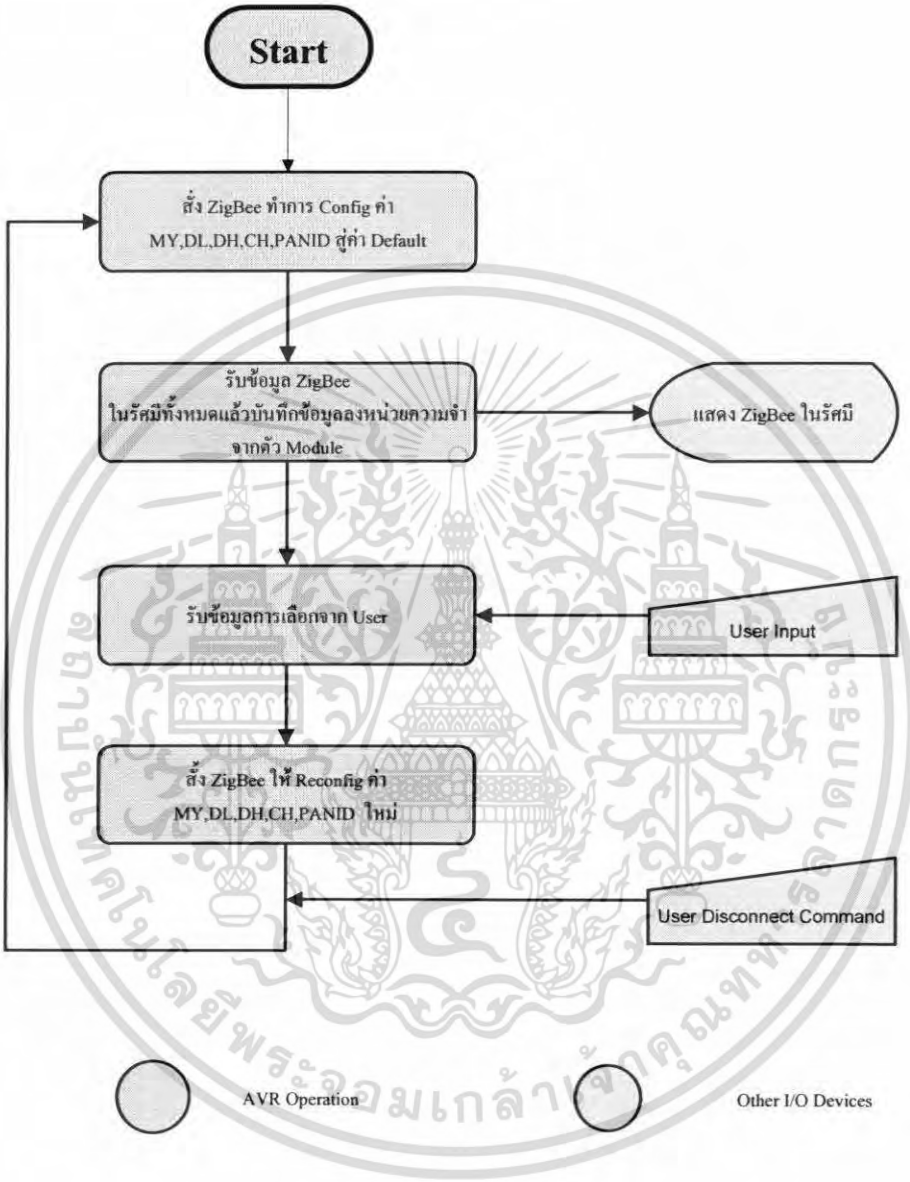
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 บล็อกไดอะแกรมแสดงการทำงานของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 การออกแบบและวิธีการใช้งาน Zigbee

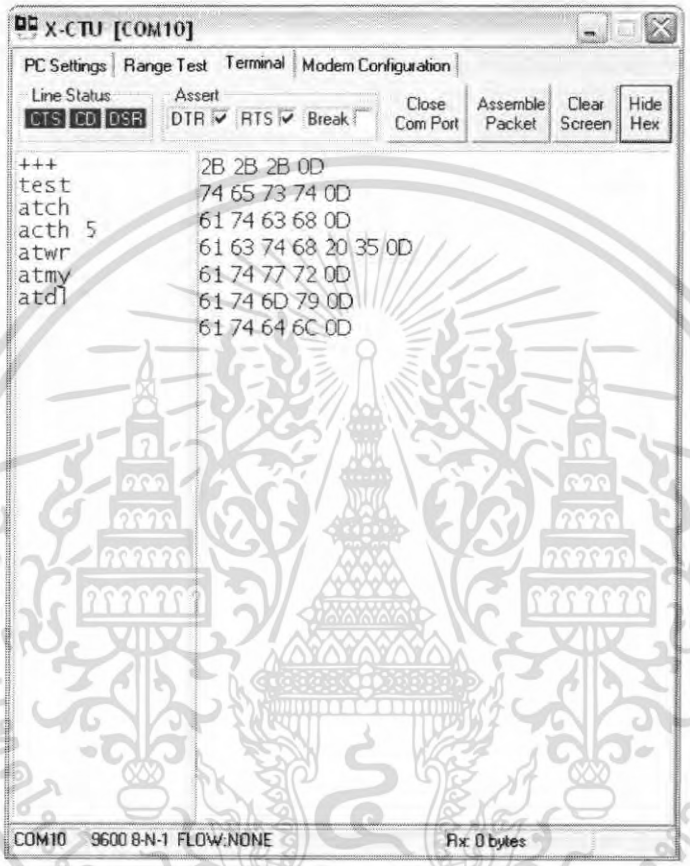


รูปที่ 3.5 การทำงานและการกำหนดค่าให้กับโมดูล Zigbee

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดค่าตัวโมดูล ZigBee

การทำงานของตัวโมดูล ZigBee นั้นควบคุมด้วยโปรแกรม X-CTU ผ่านพอร์ต RS-232 ซึ่งตัวโปรแกรมมีลักษณะดังรูป



รูปที่ 3.6 โปรแกรม X-CTU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างคำสั่ง AT Command ที่ใช้ในการทดลอง

ตารางที่ 3.1 คำสั่งที่ใช้ในการทดลองโดยทั่วไป

AT Command Function	Command Description
+++	เข้าสู่โหมด AT Command
AP	เข้าออกจากโหมด API
AS	สแกนหาโมดูลที่อยู่รอบๆทั้งหมด
CH	คำสั่งปรับค่าช่องสัญญาณที่โมดูลนั้นใช้
CN	ออกจาก Command Mode
DH	ปรับค่าโมดูลที่ต้องการติดต่อ(high)
DL	ปรับค่าโมดูลที่ต้องการติดต่อ(low)
DN	หาค่า NI ของเป้าหมาย
ID	หา ID ของ Network
MY	ปรับค่าการรับข้อมูลของโมดูล
NI	ปรับค่าอักษรประจำตัวโมดูล
RE	ปรับค่าทุกอย่างสู่ค่าเริ่มต้นจากโรงงาน
SC	สแกนพลังค่าพลังงานในช่องสัญญาณที่กำหนด
SH	ดูค่าหมายเลขประจำตัวของโมดูล(high)
SL	ดูค่าหมายเลขประจำตัวของโมดูล(low)
SM	ปรับโมดูลเข้าสู่โหมดหลับ
VR	ดูค่า firmware version
WR	เขียนคำสั่งลงสู่โมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด Transparent

โดยปกติแล้วตัวโมดูล ZigBee จะทำงานอยู่ในโหมดนี้ ในโหมดนี้ตัวโมดูลจะทำงานเปรียบเสมือนสื่อชนิดหนึ่งเท่านั้น ข้อมูลที่ได้รับจากพอร์ต UART ทั้งหมดถูกส่งออกไปในลักษณะคลื่นวิทยุ เมื่อใดที่ได้รับข้อมูลที่อยู่ในลักษณะคลื่นวิยุก็จะส่งออกไปทางขา DO

โหมด API

โหมด API เป็นอีกทางเลือกหนึ่งของการส่งข้อมูลซึ่งแตกต่างออกไปจากโหมด Transparent คือมีลักษณะการส่งข้อมูลออกไปที่เป็นเฟรม ซึ่งจะขยายความสามารถในการสื่อข้อมูลในทางเครือข่ายมากยิ่งขึ้น เมื่ออยู่ในโหมด API ข้อมูลที่เข้าและออกจากตัวโมดูลทั้งหมดจะถูกบรรจุลงไปในเฟรมที่ระบุหน้าที่หรือกระบวนการต่างๆภายในตัวโมดูลไว้ด้วย

เฟรมการส่งข้อมูล(Transmit Data Frames)

- เฟรมข้อมูลขาออก(RF Transmit Data Frame)
- เฟรมชุดคำสั่ง(Command Frame)

เฟรมการรับข้อมูล(Receive Data Frames)

- เฟรมข้อมูลขาเข้า(Receive Data Frames)
- เฟรมชุดคำสั่งตอบกลับ(Command Response)
- เฟรมตอบกลับเหตุการณ์ต่าง(Event Notification)

โหมด API ยังเป็นอีกทางเลือกหนึ่งที่จะทำการปรับเปลี่ยนค่าต่างๆภายในโมดูลและปรับเปลี่ยนการไหลของข้อมูลได้ในระดับขั้นของตัวโปรแกรม ตัวโปรแกรมนี้สามารถที่จะส่งเฟรมข้อมูลต่างๆที่บรรจุข้อมูลที่จำเป็นในการปรับค่าในตัวโมดูลโดยไม่ต้องเข้าสู่โหมดคำสั่ง (Command Mode) ตัวโมดูลจะส่งเฟรมข้อมูลไปยังตัวโปรแกรมพร้อมข้อมูลที่บ่อบอกสถานะต่างๆ จำพวก แหล่งข้อมูล ความแรงของสัญญาณ และขนาดของแพ็คเก็ตขารับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของเฟรมต่างๆในโหมด API

เราสามารถที่จะเปิดระบบ API ได้โดยการเข้าไปปรับเปลี่ยนค่าตัวแปร AP ใน command mode ได้ โดยค่า AP มีค่าได้ดังนี้

AP = 0(ค่าเริ่มต้นของโมดูล) โหมดโปร่งใส(Transparent Mode)

AP = 1 โหมด API

AP = 2 โหมด API พร้อมตัวอักษรที่รับการยกเว้น

ข้อมูลใดๆที่ได้รับมาก่อนคำสั่งจะถูกทิ้งไปโดยระบบ(ไม่มีการแจ้งเตือน) และถ้าไม่ได้รับข้อมูลตามลำดับหรือการเช็ค checksum ผิดพลาดประการใดข้อมูลเหล่านั้นจะถูกทิ้งไปโดยไม่มีการแจ้งเตือนจากระบบใดทั้งสิ้น

เมื่ออยู่ในโหมด API โครงสร้างของเฟรมจะมีลักษณะดังนี้(AP = 1)

อักษรเริ่มแพ็คเก็ต 1(ไบท์)	**ความยาว (2-3 ไบท์)**		**เฟรมข้อมูล (4-n ไบท์)**	**เช็คซัม(Checksum) (n+1 ไบท์)**
0x7E	MSB	LSB	ข้อมูล	1 ไบท์

MSB = Most Significant Byte, LSB = Least Significant Byte

เมื่ออยู่ในโหมด API พร้อมตัวอักษรพิเศษโครงสร้างของเฟรมจะมีลักษณะดังนี้(AP = 2)

อักษรเริ่มแพ็คเก็ต 1(ไบท์)	**ความยาว (2-3 ไบท์)**		**เฟรมข้อมูล (4-n ไบท์)**	**เช็คซัม(Checksum) (n+1 ไบท์)**
0x7E	MSB	LSB	ข้อมูล	1 ไบท์

มีการใส่การยกเว้นตัวอักษรพิเศษถ้ามีการใช้

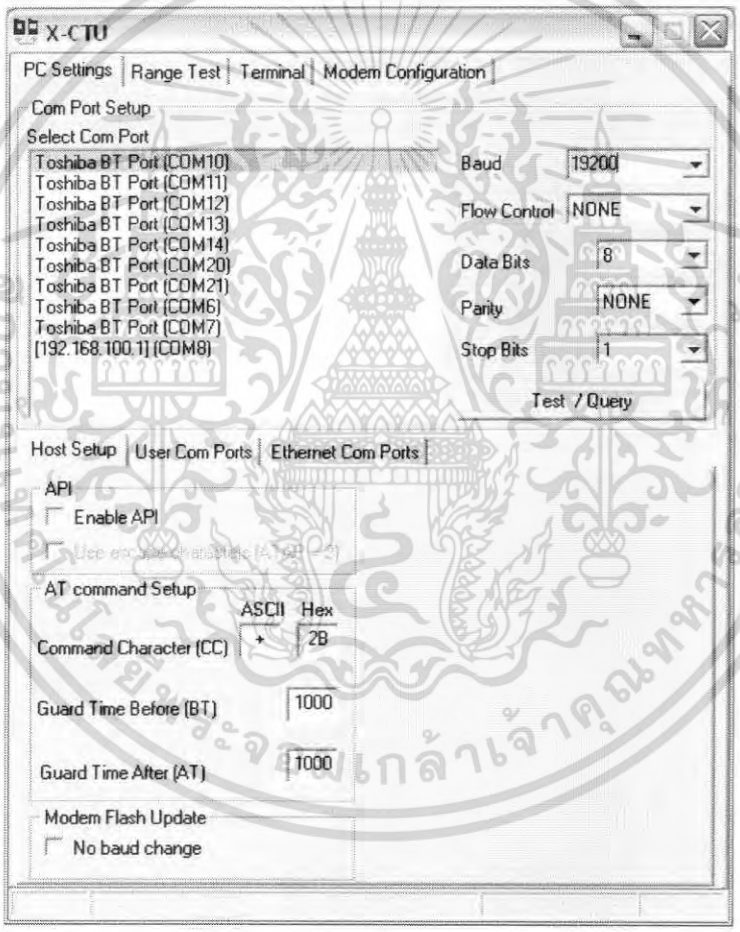
MSB = Most Significant Byte, LSB = Least Significant Byte

ตัวอักษรพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อมีการส่งหรือรับข้อมูลในลักษณะที่เป็นเฟรม ค่าพิเศษบางตัวต้องมีการทำสัญลักษณ์ไว้ (flagged) เพื่อที่จะไม่ให้มันส่งผลกระทบกับกระบวนการของพอร์ต UART เพื่อที่จะหลีกเลี่ยงปัญหานี้จึงใส่ 0x7D ไว้ด้านหน้าของแพ็คเก็ตข้อมูลตัวอักษร(พิเศษ) ตามด้วย 0x20

การส่งขอมูลในโหมดโปร่งใส (Transparent) นี้ จะส่งข้อมูลออกในรูปแบบของรหัส ASCII ซึ่งต่างออกไปจาก API ซึ่งจะส่งออกไปในลักษณะที่เป็นแพ็ตเก็ต (packet) มีการกำหนดค่าการติดต่อผ่านพอร์ต RS-232 ดังรูป



รูปที่ 3.7 การกำหนดค่าต่างๆเพื่อทำการเชื่อมต่อ

การส่งข้อมูลระหว่างตัว ZigBee ต้องทำการ configure ตัวโมดูลให้อยู่ในสถานะที่ติดต่อกันได้ เช่น ตั้งสถานะให้โมดูลตัวที่ 1 เป็น

MY = 1 DL = 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตั้งสถานะให้โมดูลตัวที่ 2 เป็น

MY = 2 DL = 1

คำสั่งที่ทำการป้อนลงไปในโมดูลตัวที่ 1 และโมดูลตัวที่ 2 เพื่อทำการตั้งสถานะดังกล่าวลงใน terminal

ตารางที่ 3.2 คำสั่งที่ป้อนลงไปในโมดูลทั้ง 2

โมดูลตัวที่ 1	**โมดูลตัวที่ 2**
+++	+++
atmy 1	atmy 2
atdl 2	atdl 1
atwr	atwr
atcn	atcn

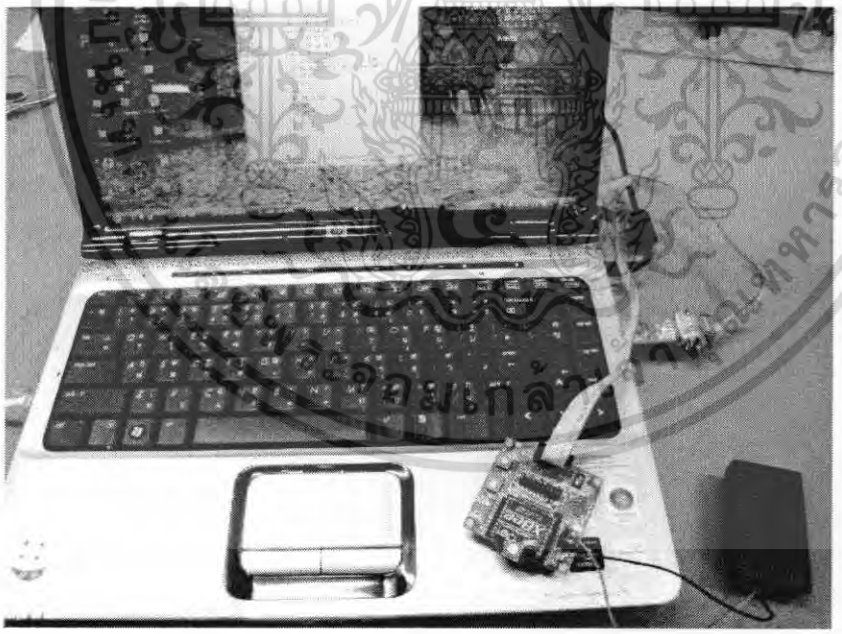
รูปที่ 3.8 ทำการกำหนดค่าการเชื่อมต่อและส่งข้อมูลระหว่างอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยถ้าทำการกำหนดค่าแบบเดิมแล้วนำไปใช้งานกับระบบสมองกลอัจฉริยะจะต้องมีการสั่งงานผ่านทางเครื่องมือที่ใช้ในการเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์จึงจะสามารถทำให้อุปกรณ์ Zigbee ทำงานสื่อสารกันได้

ตารางที่ 3.3 คำสั่งที่ป้อนลงไปในโมดูลทั้ง 2 เพื่อให้ทำการเชื่อมต่อกับระบบสมองกลอัจฉริยะ

คำสั่งที่ป้อนลง Module ทั้งสอง	
+++	(Enter AT Command)
ATRE	(Reset Module)
ATND	(Find nodes and record)
ATDL	(Set Recorded Destination Low)
ATDH	(Set Recorded Destination High)
ATCN	(Exit AT Command)



รูปที่ 3.9 การกำหนดค่าเพื่อเริ่มการเชื่อมต่อระหว่างอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ขั้นตอนการใช้งานเครื่องเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์

(ET-AVR ISP USB V1.0)

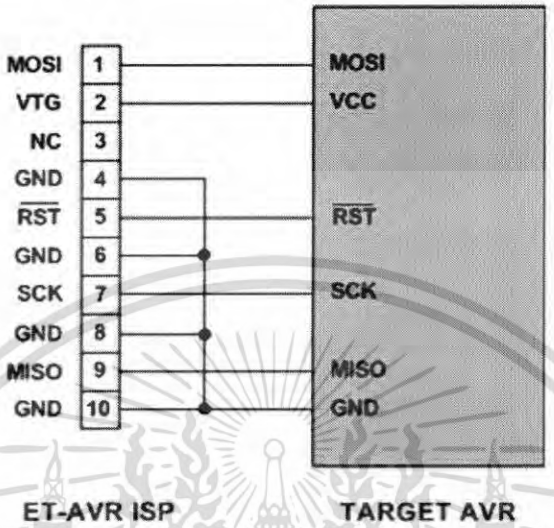
ET-AVR ISP USB V1.0 เป็นบอร์ดที่ออกแบบมาเพื่อใช้ในการดาวน์โหลด Hex File ให้กับ MCU ตระกูล AVR ของ Atmel โดยผ่านทาง ISP Interface โดยต้องใช้ร่วมกับโปรแกรม AVR Studio หรือซอฟแวร์อื่นๆ ที่รองรับ AVR ISP ซึ่งปัจจุบันมีให้เลือกใช้หลายตัว

คุณสมบัติของ ET-AVR ISP USB V1.0

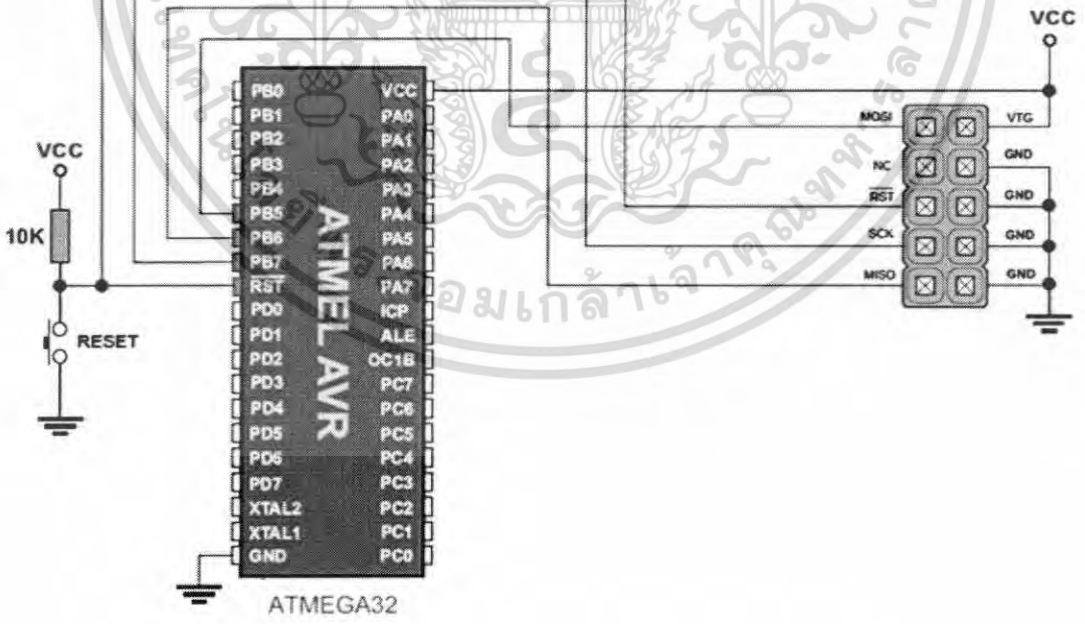
1. มีคุณสมบัติเทียบเท่า AVR ISP ของ ATMEL
2. โปรแกรมผ่านทาง ISP Interfacce
3. สามารถอัพเกรด Firmware โดยตรงผ่านโปรแกรม AVR Studio 4 โดยไม่ต้องใช้เครื่องโปรแกรมจากภายนอก เพื่อให้สามารถใช้กับ MCU เบอร์ใหม่ ๆ ได้ ซึ่ง Firmware จะติดมากับโปรแกรม AVR studio 4
4. สามารถใช้ได้กับระบบไฟเลี้ยงตั้งแต่ 2.7V – 5.5V
5. การติดต่อสื่อสารผ่านพอร์ต USB
6. มีสัญญาณ Clock ไว้จ่ายให้ MCU ในกรณีที่ Fuse Bit เลือกแหล่งของสัญญาณ Clock ผิด
7. มี LED แสดงการทำงาน Power , Activity , Usb
8. สามารถใช้กับซอฟแวร์ต่างๆ ที่รองรับ AVR ISP เช่น AVR Studio,WinAVR,ICC AVR ,CodeVision ,BASCOM-AVR เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อ ET-AVR ISP กับไมโครคอนโทรลเลอร์ AVR



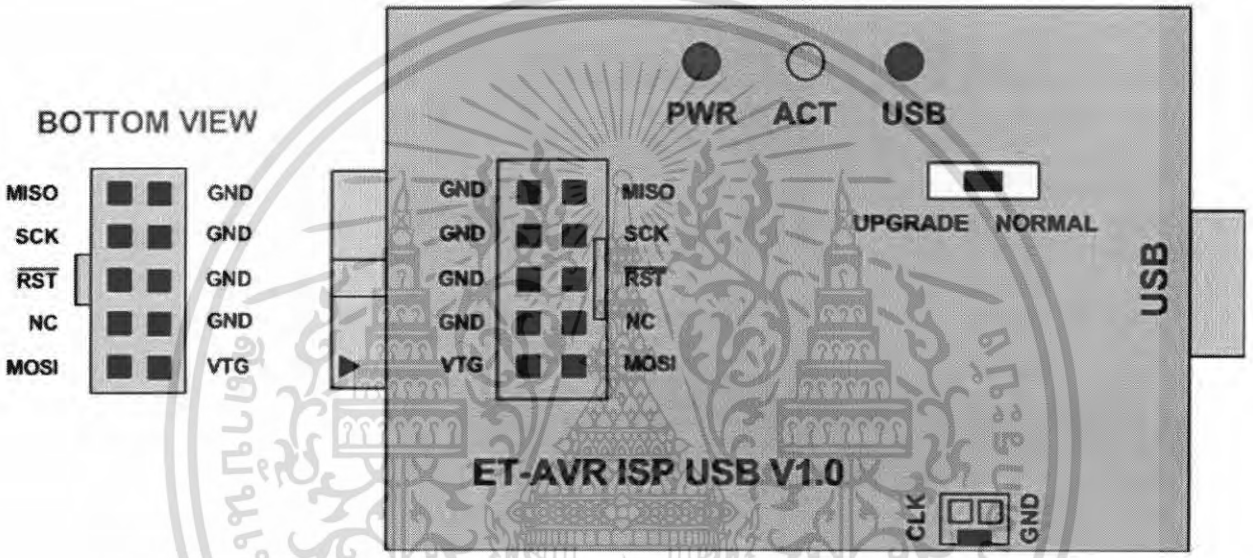
รูปที่ 4.0 การเชื่อมต่อ ET-AVR ISP กับ ไมโครคอน โทรลเลอร์ AVR



รูปที่ 3.10 การเชื่อมต่อกับ AVR เบอร์ ATMEGA32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อใช้งานจะใช้สายสัญญาณ **MISO , MOSI , SCK , RST , VTG , GND** ซึ่งจะต้องต่อกับขา ISP Interface ของ AVR ซึ่งท่านจะต้องมีไฟเลี้ยงบอร์ด Target ต่างหากไม่สามารถใช้ไฟเลี้ยงจาก ETAVR ISP USB V1.0 ได้ และที่สำคัญสวิทช์เลือกโหมดการทำงานต้องอยู่ในตำแหน่ง NORMAL เท่านั้น

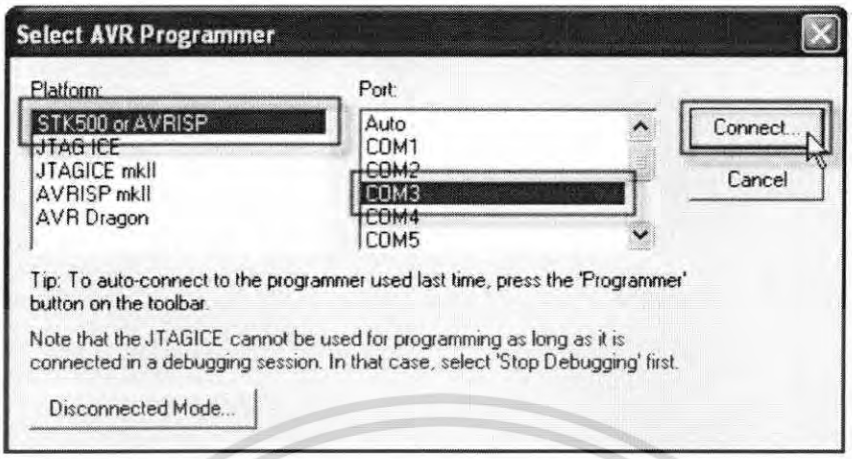


รูปที่ 3.11 ตำแหน่งขาสัญญาณของ ET-AVR ISP USB V1.0 โดยมองจากด้านล่าง

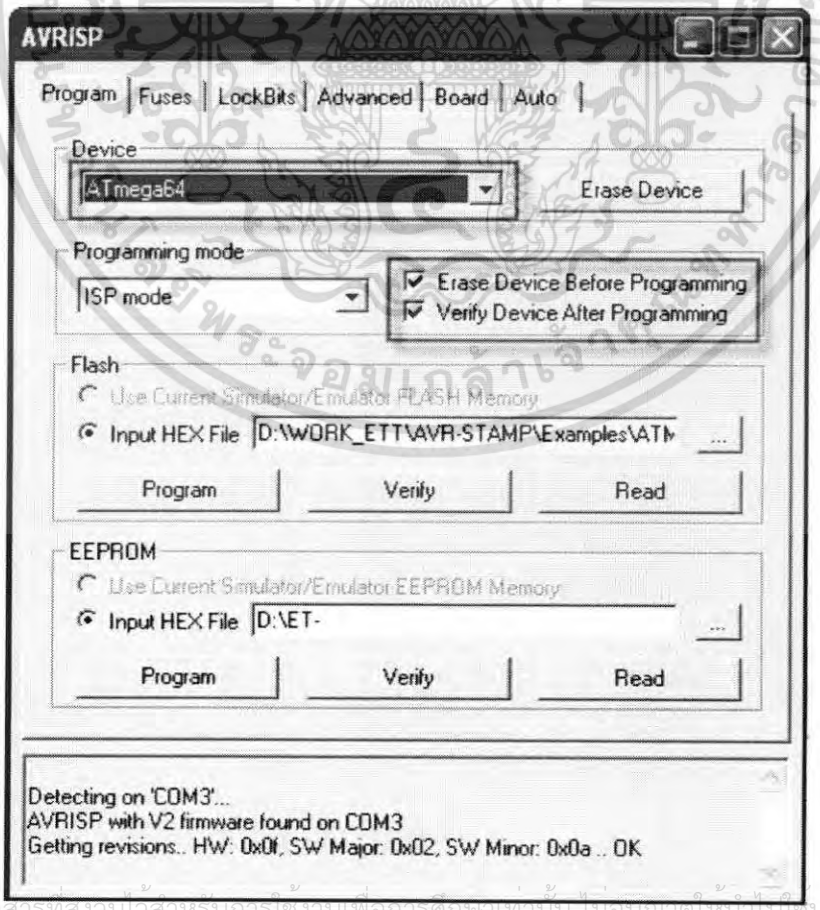
การใช้งาน ET-AVR ISP USB V1.0 ร่วมกับโปรแกรม AVR Studio 4

1. เปิดโปรแกรม AVR Studio และคลิกที่ปุ่ม
2. จากนั้นจะปรากฏหน้าต่าง Select AVR programmer ให้ทำการเลือก Platfrom เป็น STK500 or AVRISP และ Port เลือกให้เลือกพอร์ตที่ ET-AVR ISP USB V1.0 ติดตั้งอยู่ จากตัวอย่างเลือกเป็นCOM3 จากนั้นคลิก Connect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



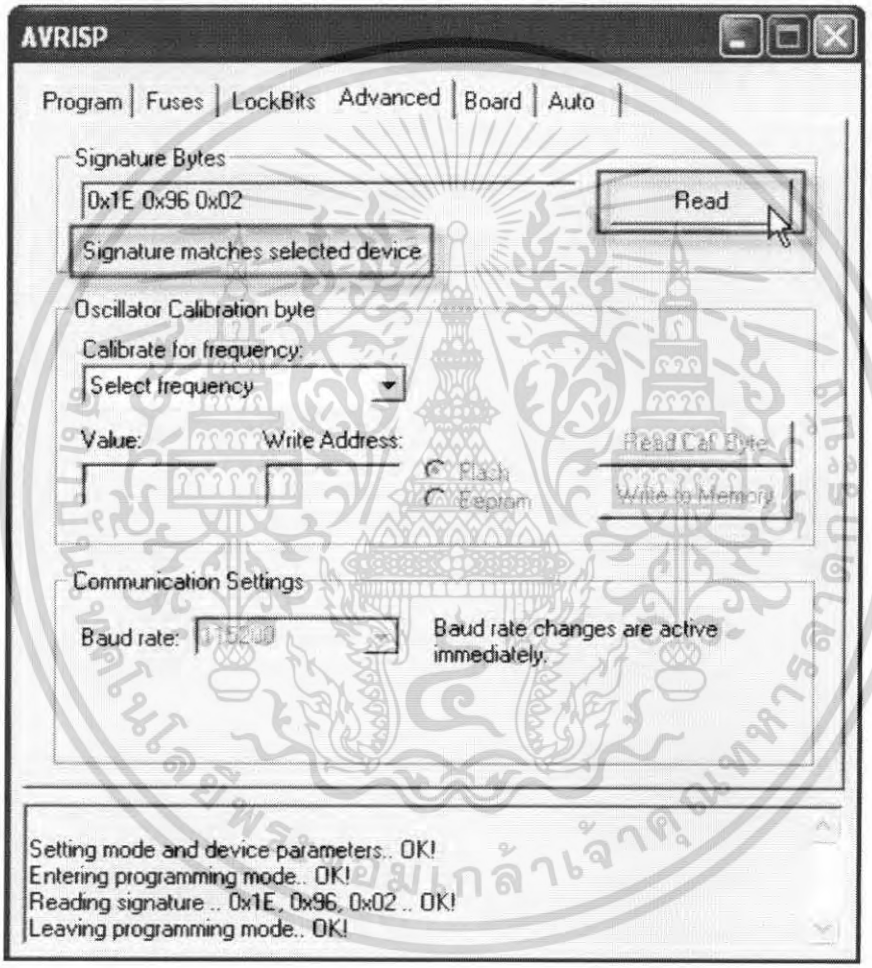
3. ถ้าโปรแกรม AVR Studio สามารถติดต่อกับ ET-AVR ISP USB V1.0 จะปรากฏหน้าต่าง AVRISP ดังรูป ให้เลือก Device ที่ต้องการโปรแกรม และเช็คที่ Erase Device Before Programming และ Verify Device After Programming เพื่อทำการลบข้อมูลเก่าออกก่อนที่จะทำการโปรแกรม และ ตรวจสอบข้อมูลหลังจากการโปรแกรมเสร็จสิ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

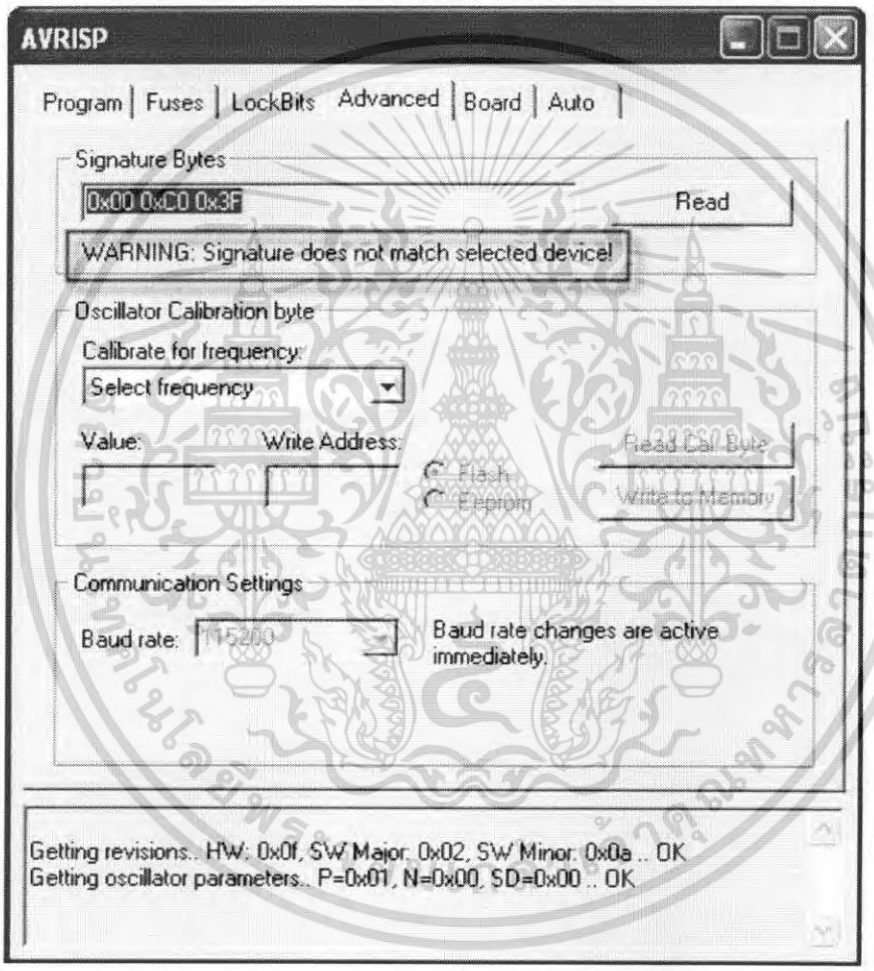
4. เลือกมาที่แท็ป Board เพื่อกำหนดความเร็วการโปรแกรม (ISP Freq) โดยค่านี้ถ้ามีความถี่สูง ความเร็วการโปรแกรม Hex File ลง MCU จะสูงตามไปด้วยในที่นี้เลือกความถี่สูงสุดคือ 921.6 kHz จากนั้นคลิกปุ่ม Write

5. เลือกแท็ปมาที่ Advance และคลิกที่ Read ถ้าความถี่ ISP Freq ที่เลือกใช้ได้จะปรากฏข้อความ Signature matches select device ดังรูป



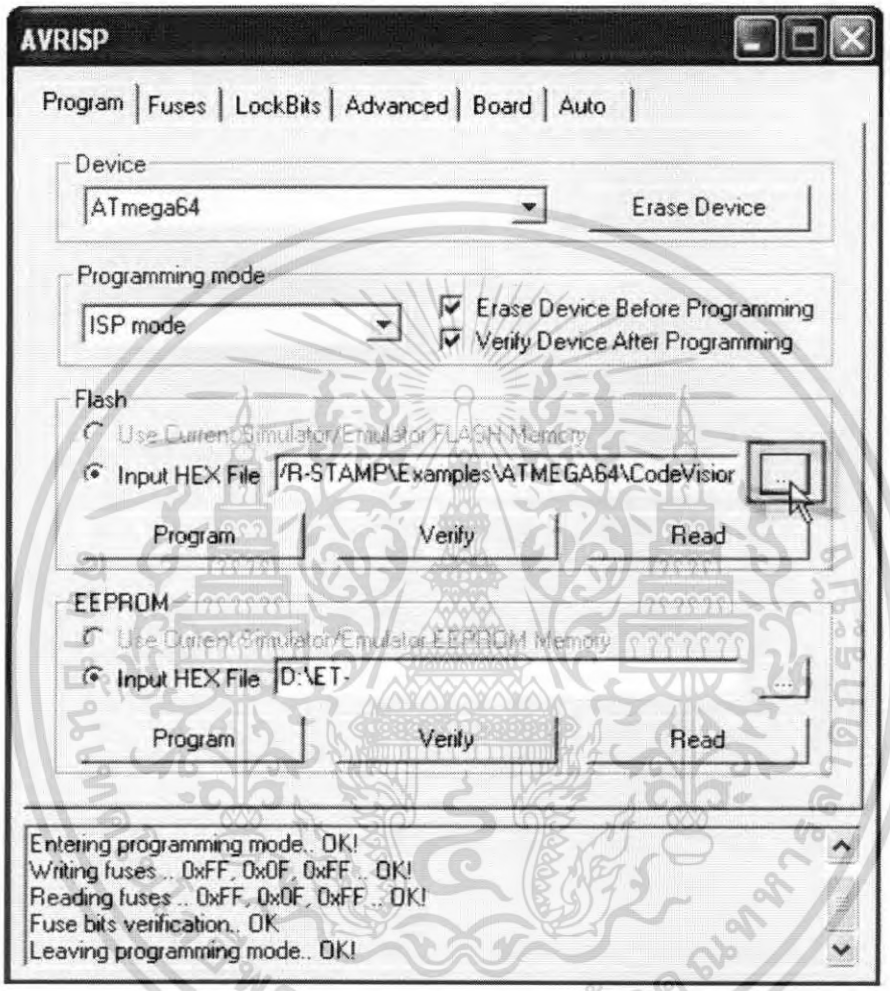
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ถ้าปรากฏข้อความ WARNING: Signature does not matches select device แสดงว่าความถี่ ISP Freq ที่เลือกสูงเกินไปหรือเลือกเบอร์ MCU ไม่ตรงกับที่ใช้งานจริง ให้ทำการลดความถี่ ISP Freq ลงหรือเลือกเบอร์ MCU ให้ตรงจากนั้นทำการอ่าน Signature Bytes อีกครั้งจนกว่าจะไม่มีข้อความเตือน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. กลับมาที่แท็ป Program และคลิกปุ่มในช่องของ Input HEX File เพื่อระบุ HEX File ที่ต้องการที่จะโปรแกรมลง MCU

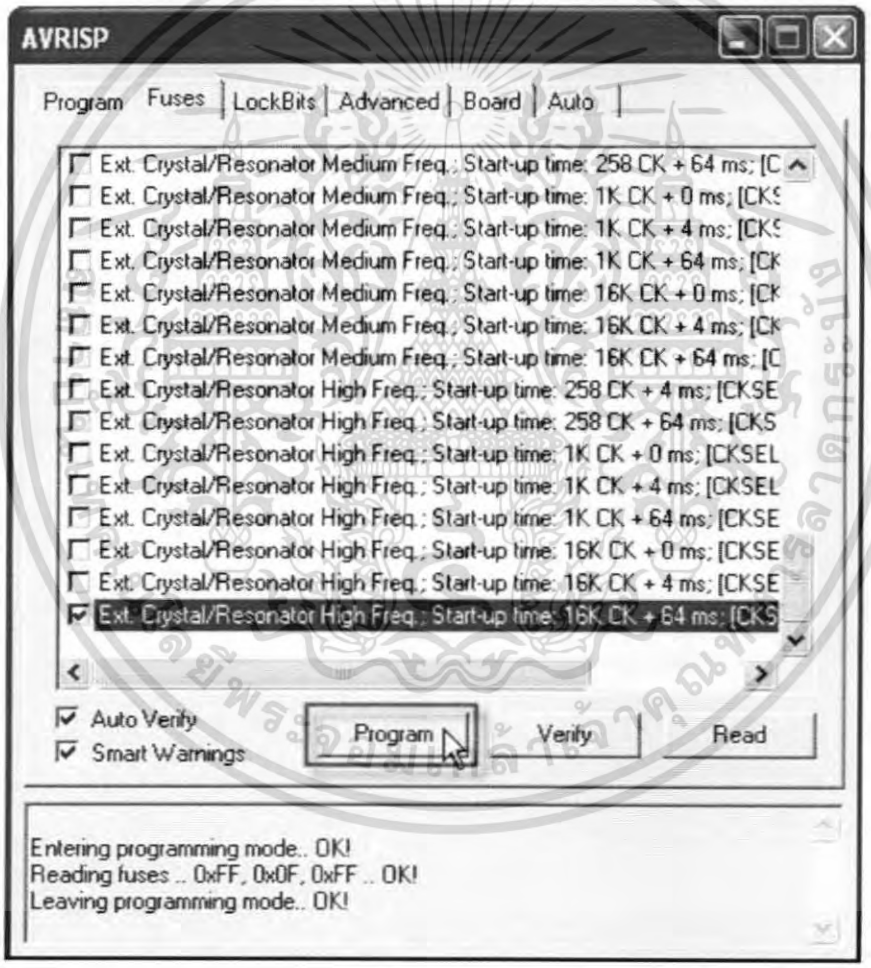


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เลือก HEX File ที่ต้องการโปรแกรม จากนั้นคลิกที่ปุ่ม Program เพื่อเริ่มการโปรแกรมถ้าการโปรแกรมเสร็จสมบูรณ์ไม่มีข้อผิดพลาดลำดับขั้นตอนการโปรแกรมจะต้อง OK! ดังรูป

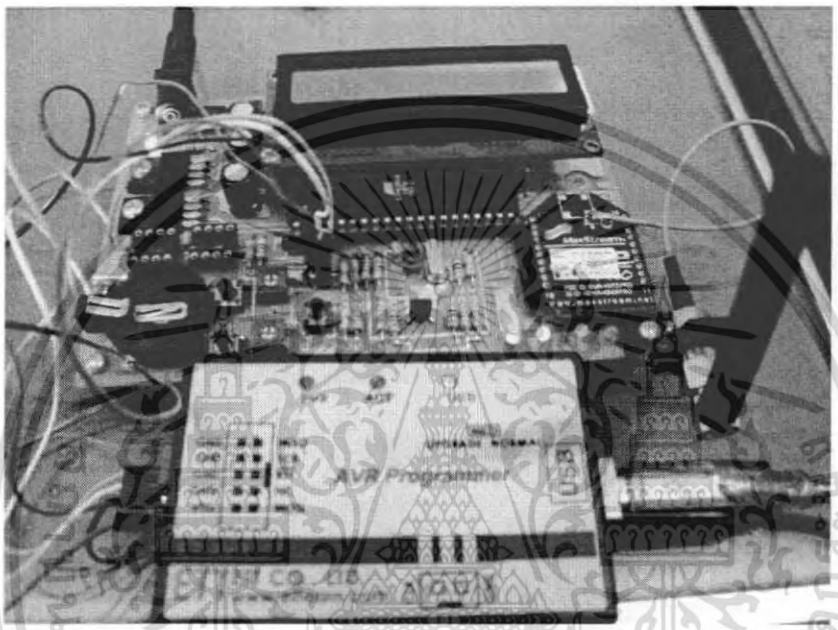
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. AVRISP สามารถที่จะโปรแกรม Fuses และ LockBits ได้โดยเลือกไปที่แท็ป Fuses หรือ LockBitsจากนั้นทำการการเลือกบิตที่ต้องการจะโปรแกรมและคลิกที่ปุ่ม Program ดังรูป แต่การโปรแกรมFuses ต้องมีความระมัดระวังเป็นพิเศษเรื่องบิตที่ใช้เลือกสัญญาณนาฬิกาของ CPU เพราะว่าถ้าโปรแกรมผิดจะให้ไม่สามารถโปรแกรมผ่านทาง ISP ได้ แต่ก็มีวิธีแก้ ซึ่งจะกล่าวถึงต่อไป เรื่องรายละเอียดของ Fuses และ LockBits นี้สามารถดูรายละเอียดได้จาก Data Sheet ของ แต่ละเบอร์ (ถ้าใช้บอร์ดของ ETT ค่า Fuses ต่างๆ จะถูกโปรแกรมไปแล้วผู้ใช้ไม่ต้องโปรแกรมอีก)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการกำหนดค่าเพื่อเริ่มการเชื่อมต่อเสร็จแล้ว ก็จะทำการทดสอบการเชื่อมต่อโดยการส่งข้อมูลง่ายๆ ผ่านตัวอุปกรณ์ทั้ง 2 ต่อจากนั้นก็จะทำการเตรียมอุปกรณ์สำหรับเขียนโปรแกรมลงบนไมโครคอลโครเลอร์เพื่อใช้ในการควบคุมอุปกรณ์ Zigbee ผ่านทางไมโครคอลโทรเลอร์ AVR



รูปที่ 3.12 วงจรเชื่อมต่อระหว่างเครื่องมือเขียนโปรแกรมกับ Embedded System

รูปที่ 3.13 การใช้งานเครื่องมือเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของขาสำคัญต่างๆของตัวโมดูลและการเชื่อมต่อภายในวงจร

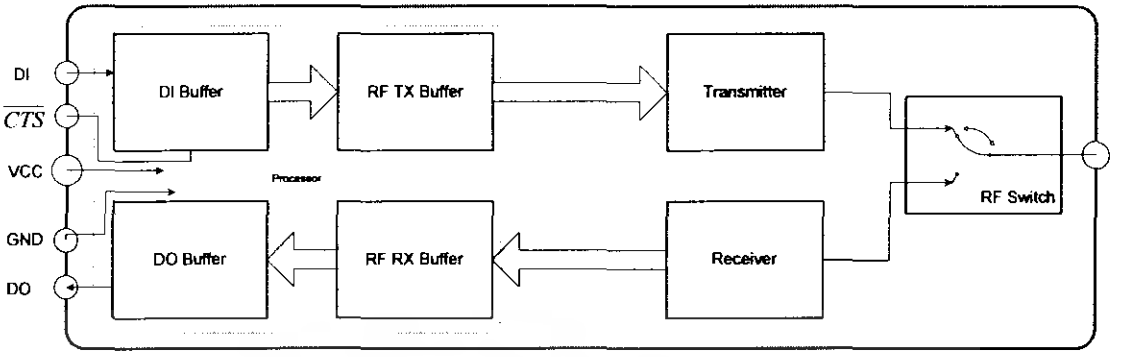
ตารางที่ 3.3 การต่อขากับอุปกรณ์ต่างๆ

ZigBee Module Pin	Description	Connected To:	AVR pin number	Description
2	Dout		14	PD0/RXd
3	Din		15	PD1/TXd
5	Reset		20	PD6/ICP1
6	RSSI(Receive Signal Strength Indicator)		36	PA4/AD4
9	Pin Sleep Control Line		19	PD5/OC1A

คำอธิบาย

- ขาที่ 2 ของตัวโมดูลเพื่อใช้ในการส่งข้อมูลออกไปยังตัวไมโครคอนโทรลเลอร์
- ขาที่ 3 ของตัวโมดูลเพื่อใช้ในการรับข้อมูลจากตัวไมโครคอนโทรลเลอร์
- ขาที่ 5 เพื่อให้ตัวไมโครคอนโทรลเลอร์สามารถสั่ง reset ตัวโมดูลได้
- ขาที่ 6 เพื่อที่ตัวโมดูลจะสามารถส่งค่าความแรงของสัญญาณมายังตัวไมโครคอนโทรลเลอร์เพื่อการคำนวณได้
- ขาที่ 9 เพื่อที่ตัวไมโครคอนโทรลเลอร์สามารถที่จะเปลี่ยนสถานะของตัวโมดูลให้เข้าหรือออกจากโหมดหลับ (sleep mode) ได้เพื่อประหยัดพลังงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 การรับ-ส่งข้อมูลภายในตัวโมดูล Zigbee

3.2.3 การออกแบบวงจรด้วย Protel 99 SE

กระบวนการสร้าง PCB

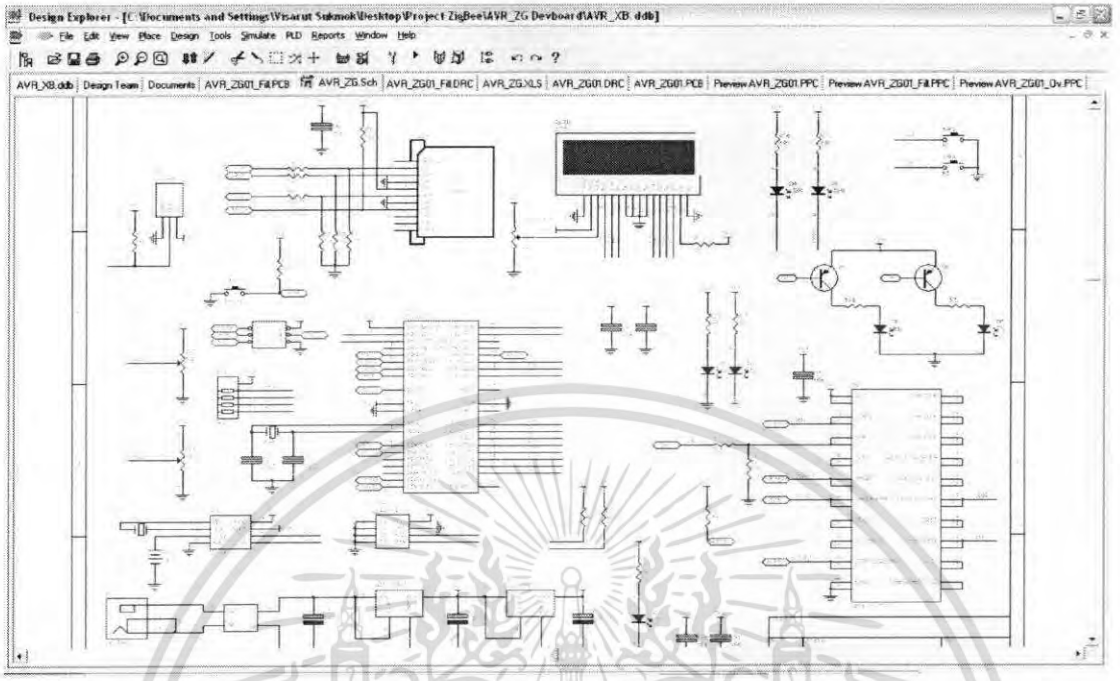
ขั้นตอนการออกแบบ

เป็นกระบวนการออกแบบโดยใช้ซอฟท์แวร์ช่วยในการออกแบบ โดยในที่นี้จะใช้โปรแกรม Protel 99 ในการออกแบบ มีขั้นตอนดังนี้

1. Schematic Capture – การใส่วงจรด้วยซอฟท์แวร์ จะใช้สัญลักษณ์แทนอุปกรณ์จริงซึ่งจะออกแบบตามรูปร่างมาตรฐานและทำการกำหนดการเชื่อมต่อสัญญาณทางไฟฟ้าจากอุปกรณ์หนึ่งไปยังอีกอุปกรณ์หนึ่ง โดยที่ซอฟท์แวร์จะสามารถตรวจสอบความผิดพลาดในการออกแบบเบื้องต้นได้
2. PCB Layout – นำข้อมูลซึ่งอยู่ในรูปเน็ทลิสต์ (Netlist) จากวงจรที่ทำการออกแบบมาเชื่อมต่อกับฟุทปริ้นท์ (Footprint) ซึ่งจะเป็นรูปร่างที่แท้จริงของอุปกรณ์บน PCB รวมกันแล้วทำการสร้างเป็นลายวงจร โดยการอาศัยซอฟท์แวร์ทำให้การออกแบบสามารถทำได้ทั้งแบบง่าย และแบบที่เป็นการออกแบบลายวงจรที่ซับซ้อน
3. Artwork Generation – หลังการสร้าง PCB เสร็จเรียบร้อยแล้ว ผลลัพธ์ที่ได้คือ อาร์ทเวิค (Artwork) หรือแผ่นฟิล์มซึ่งจะมีลายทึบแสงในตำแหน่งลายเส้นทองแดง โดยที่อาร์ทเวิคที่ได้นี้จะนำไปทำแผ่น PCB โดยที่ซอฟท์แวร์ที่ใช้สามารถระบุความละเอียดของงานที่จะทำได้ตามความต้องการ

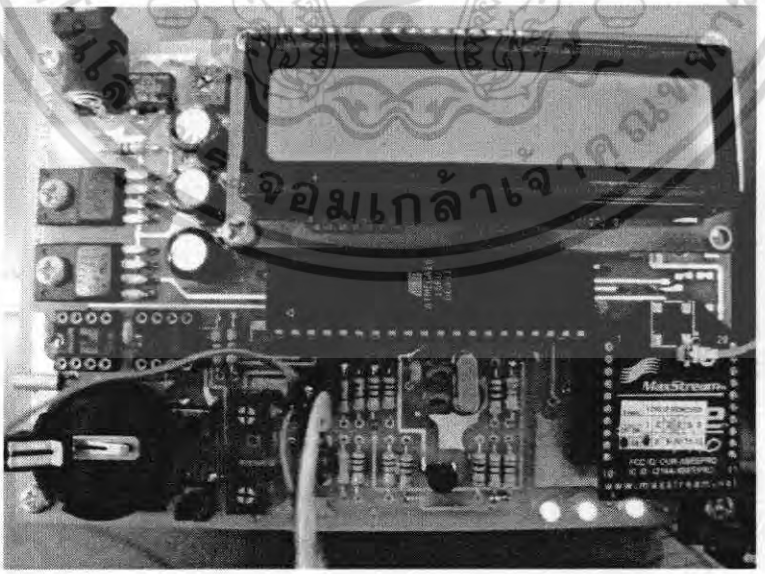
การออกแบบวงจรไมโครคอนโทรลเลอร์ AVR ที่นำมาควบคุมโมดูล ZigBee ผ่านพอร์ต UART เพื่อสร้างระบบสมองกลฝังตัวขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 แผนผังวงจร AVR + ZigBee

เราจะใช้โปรแกรมนี้ในการออกแบบแผนผังวงจรและลายวงจรและนำไปกัดปรินต์เพื่อสร้างวงจรสมองกลฝังตัว ระหว่างตัวโมดูลกับไมโครคอนโทรลเลอร์ AVR



รูปที่ 3.16 วงจร AVR + Zigbee

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบโครงงานทางด้านซอฟท์แวร์

ในส่วนนี้จะทำการออกแบบโปรแกรมที่ใช้ควบคุมการทำงานในโหมดต่างๆของโมดูล Zigbe ซึ่งจะทำการควบคุมผ่านทางไมโครคอลโทรเลอร์ตระกูล AVR โดยการออกแบบโปรแกรมที่ใช้งานนั้นจะออกแบบและทดลองเขียนข้อมูลลงใน AVR จริงผ่านทางโปรแกรมที่จะกล่าวถึงต่อไป ในส่วนของโปรแกรมอื่นที่ใช้ก็ได้แก่ โปรแกรมที่ใช้ในการทดลองหาคำสั่งที่จำเป็นในการใช้งานโมดูล Zigbee คือ X-CTU ซึ่งเป็นโปรแกรมที่ทาง Maxstream ออกแบบมาให้ใช้ตรวจสอบการใช้งานเบื้องต้น

การเขียนโปรแกรมเพื่อมาควบคุมการทำงานของตัวไมโครคอนโทรลเลอร์ AVR นั้นเราเลือกใช้ภาษา C ในการเขียน โดยมีจุดประสงค์สองอย่างหลักๆ คือ เพื่อการคิดคำนวณและเก็บข้อมูลที่ได้มากจากตัวโมดูล ZigBee และเพื่อปรับเปลี่ยนสถานะของตัวโมดูลให้เหมาะกับเหตุการณ์และสถานการณ์ต่างๆ

การเขียนโปรแกรมการควบคุมส่วนใหญ่อยู่ที่การรับข้อมูลและส่งข้อมูลออกไปโดยอาจจะมีการคำนวณอยู่บ้าง การรับส่งข้อมูลระหว่างตัวไมโครคอนโทรลเลอร์ AVR กับตัวโมดูล ZigBee นั้นเป็นไปได้สองรูปแบบ คือ แบบโปร่งใสหรือแบบแพ็คเก็ต แต่การติดต่อเพื่อทำการควบคุมตัวโมดูลนั้นเราจะใช้คำสั่งในลักษณะ AT Command หรือชุดคำสั่ง AT

สิ่งที่ต้องระวังคือการแบ่งแยกคำสั่งสองรูปแบบนี้เพราะตัวโมดูลจะใช้สลับกันได้ โดยสิ่งที่แบ่งแยกสองคำสั่งนี้คือช่วงเวลาที่ตัวโมดูลทำการเข้าและออกจากโหมด AT Command

การทำงานของระบบสมองกลฝังตัว นั้นต้องทำให้ระบบทำงานในลักษณะที่ซ้ำๆกันไปเรื่อยๆโดยยังคงรักษาการทำงานที่เป็นขั้นตอน ควรมีการใช้จำนวนคำสั่งให้น้อยและสั้นที่สุดเท่าที่จะเป็นไปได้ ซึ่งสนับสนุนในโหมด API

โครงงานชุดนี้เราจะใช้การทำงานในโหมด API ในการใช้งานจริง เพราะมีความสามารถในการบริหารเครือข่ายและสัญญาณได้ดีกว่าอีกทั้งสามารถที่จะปรับเปลี่ยนคำสั่งให้สั้นกว่าในระบบโปร่งใส (transparent) แต่เราจะใช้โหมดโปร่งใสในการทดลองและการทำงานในขั้นต้นและขั้นทดลองก่อนเพื่อที่จะเรียนรู้ชุดคำสั่งและขั้นตอนการทำงาน จากนั้นถึงทำการศึกษาการทำงานของระบบ API เพิ่มเติม เช่น ศึกษาลักษณะของแพ็คเก็ตที่รับส่งไปมาระหว่างกัน ศึกษาข้อมูลในแพ็คเก็ตที่เกี่ยวข้องกับการบริหารเครือข่ายและการนำข้อมูลเหล่านั้นมาคำนวณออกมาเป็นสื่อที่เข้าใจง่ายต่อผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมต้องมีการรับและส่งข้อมูล การจะเขียนโปรแกรมของทั้งสองแบบต้องมีการอ่านแพ็คเก็ดก่อน การศึกษาแพ็ดเก็ตจึงมีความสำคัญมาก ในอนาคตถ้าเราต้องการเพิ่มเติมอุปกรณ์ใดๆ ลงไปในวงจรก็สามารถทำได้โดยง่าย และยังสามารถส่งข้อมูลจากอุปกรณ์เหล่านั้นออกไปได้ในลักษณะเดียวกับข้อความต่างๆ ดังเช่น สามารถส่งพิกัดจากเครื่องรับสัญญาณดาวเทียม(GPS) เพื่อคำนวณเส้นทางและระยะทางได้ สามารถส่งสถานะของมิเตอร์ต่างๆในรถยนต์ไปยังจุดรับสัญญาณต่างๆได้ สามารถส่งข้อมูลฉุกเฉินเพื่อร้องขอความช่วยเหลือจากผู้รับสัญญาณข้างเคียงพร้อมกับการระบุสถานะของรถยนต์ได้

ขั้นตอนการดำเนินโครงงาน

1.ออกแบบและสร้างแผงวงจร AVR Microcontroller + ZigBee Module มาตามโครงสร้างหรือ schematics ที่กำหนดไว้ก่อนหน้านี้

2.เขียนโปรแกรมควบคุมการสั่งการระหว่าง

ZigBee Module ↔ AVR Microcontroller ↔ LCD Monitor

ตามที่กำหนดไว้ใน flowchart ก่อนหน้านี้ ด้วยโปรแกรมสำหรับสร้างงานด้วยภาษา C คือ โปรแกรม Code VisionAVR

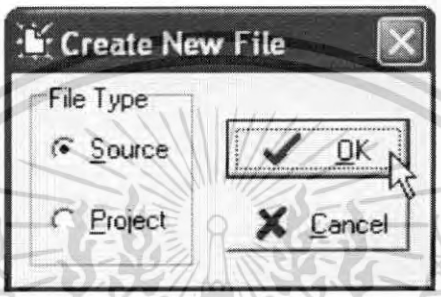
3.ทำการ Burn ตัวซอฟต์แวร์หรือตัวโปรแกรมลงไปในไมโครคอนโทรลเลอร์(AVR) โดยใช้โปรแกรม AVR Studio 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1 ขั้นตอนการใช้งานโปรแกรม Code VisionAVR

สำหรับโปรแกรมที่จะใช้ในการเขียนโปรแกรมภาษาซี ก็คือโปรแกรม CodeVisionAVR ขั้นตอนการใช้งานโปรแกรมมีดังนี้

1. เปิดโปรแกรม CodeVisionAVR C Compiler และคลิกเลือกที่เมนูคำสั่ง File □ New
2. เลือก File Type เป็น Source เพื่อสร้างไฟล์ภาษาซีใหม่และคลิกปุ่ม OK ดังรูป



3. จากนั้นจะปรากฏหน้าต่าง Editor ให้ทำการเขียนโปรแกรมตามที่ได้กำหนดไว้
4. ทำการบันทึกโปรแกรมภาษาซีที่เขียนโดยเลือกเมนู File □ Save ทำการตั้งชื่อไฟล์และกดปุ่ม Save ดังรูป

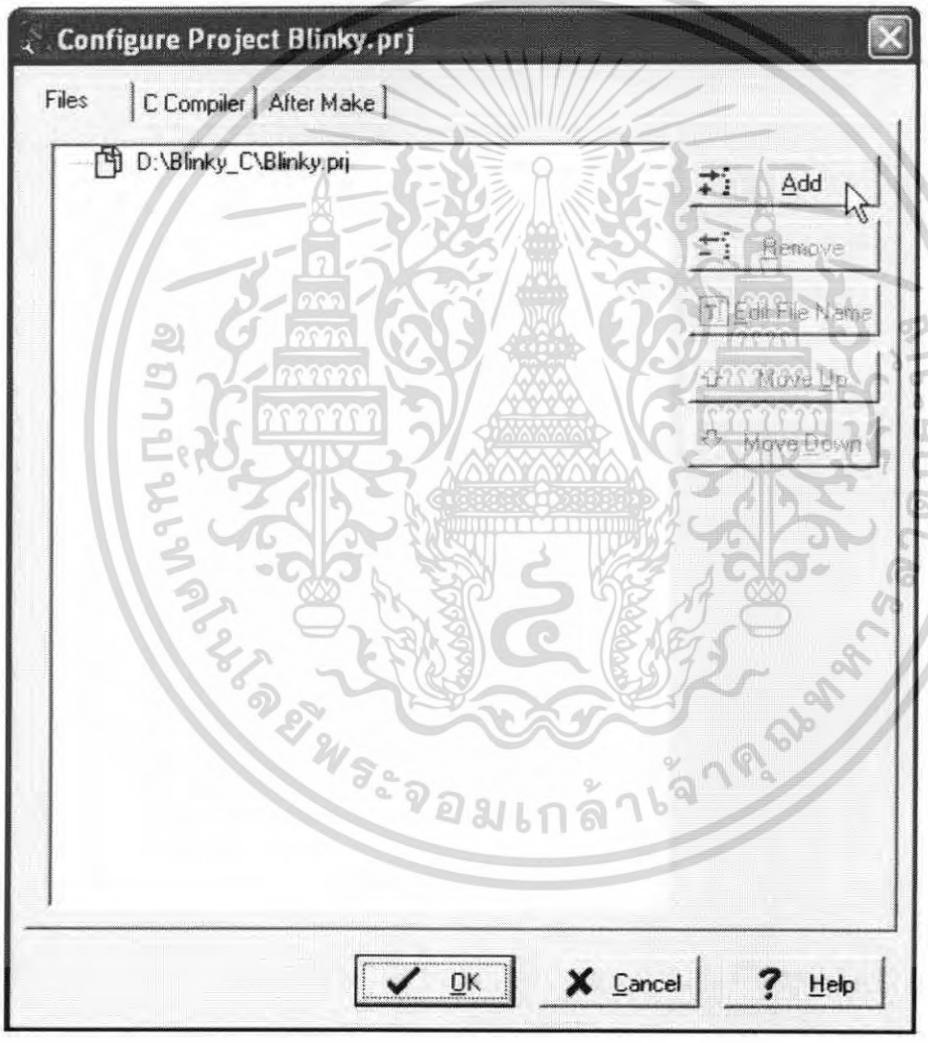
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เลือกที่เมนู File □ New และเลือก File Type เป็น Project เพื่อสร้างโปรเจกต์ใหม่และคลิกปุ่ม OK

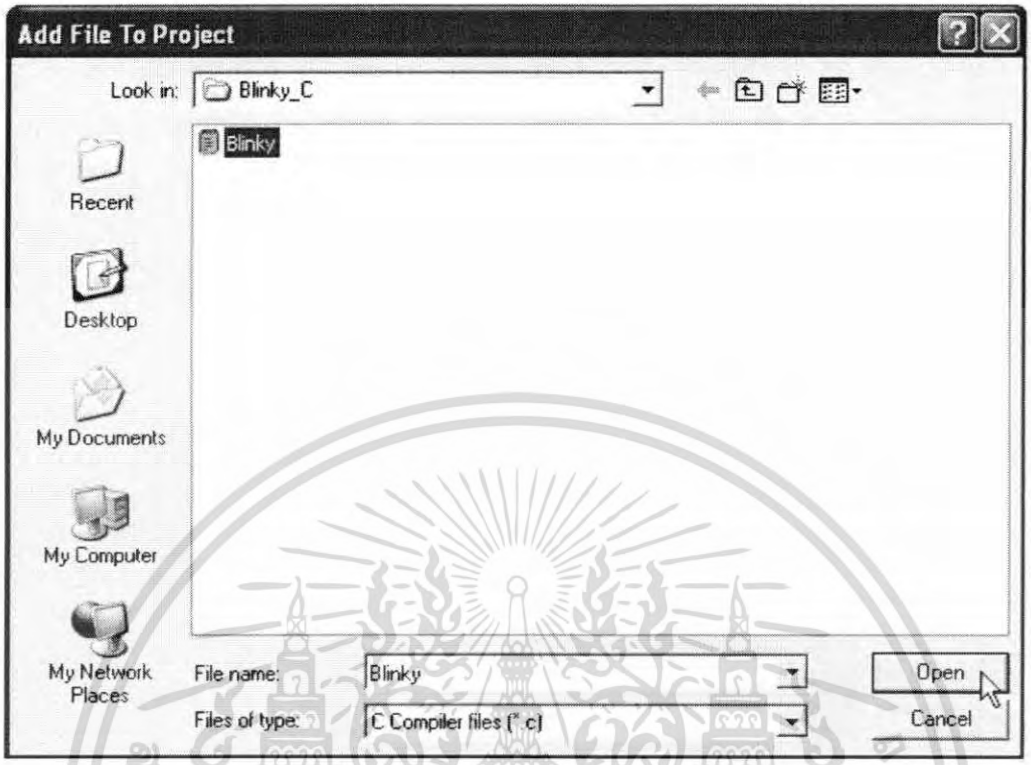
6. คลิกปุ่ม No เพื่อไม่ใช้ตัวช่วยในการสร้างโปรเจกต์ (CodeWizard)

7. ทำการตั้งชื่อโปรเจกต์ตามต้องการและคลิกปุ่ม Save

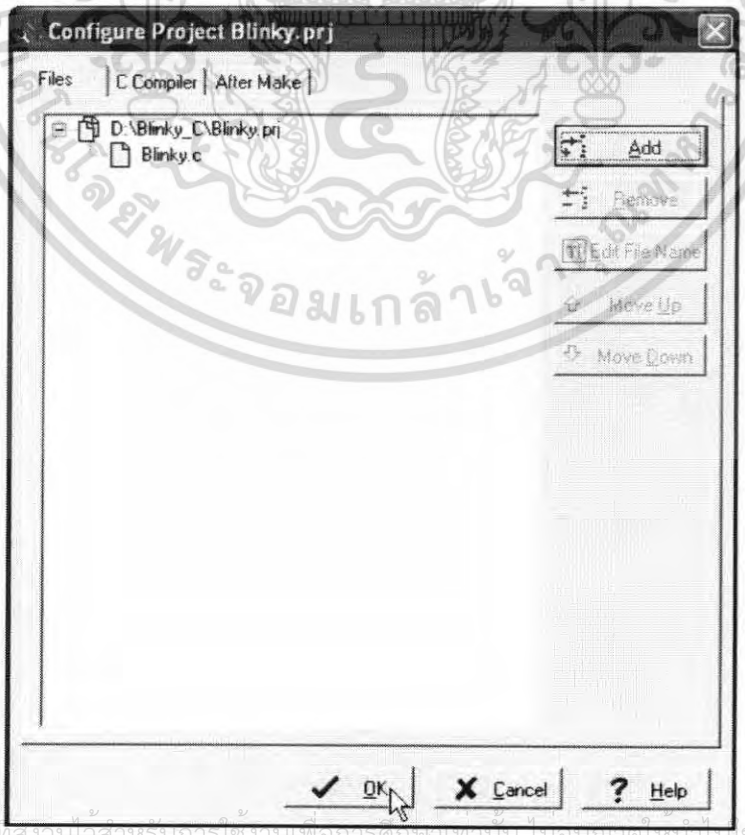
8. ทำการเพิ่มไฟล์ภาษาซีที่เขียนไว้ก่อนหน้านี้เข้ามาในโปรเจกต์โดยการคลิกปุ่ม Add ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

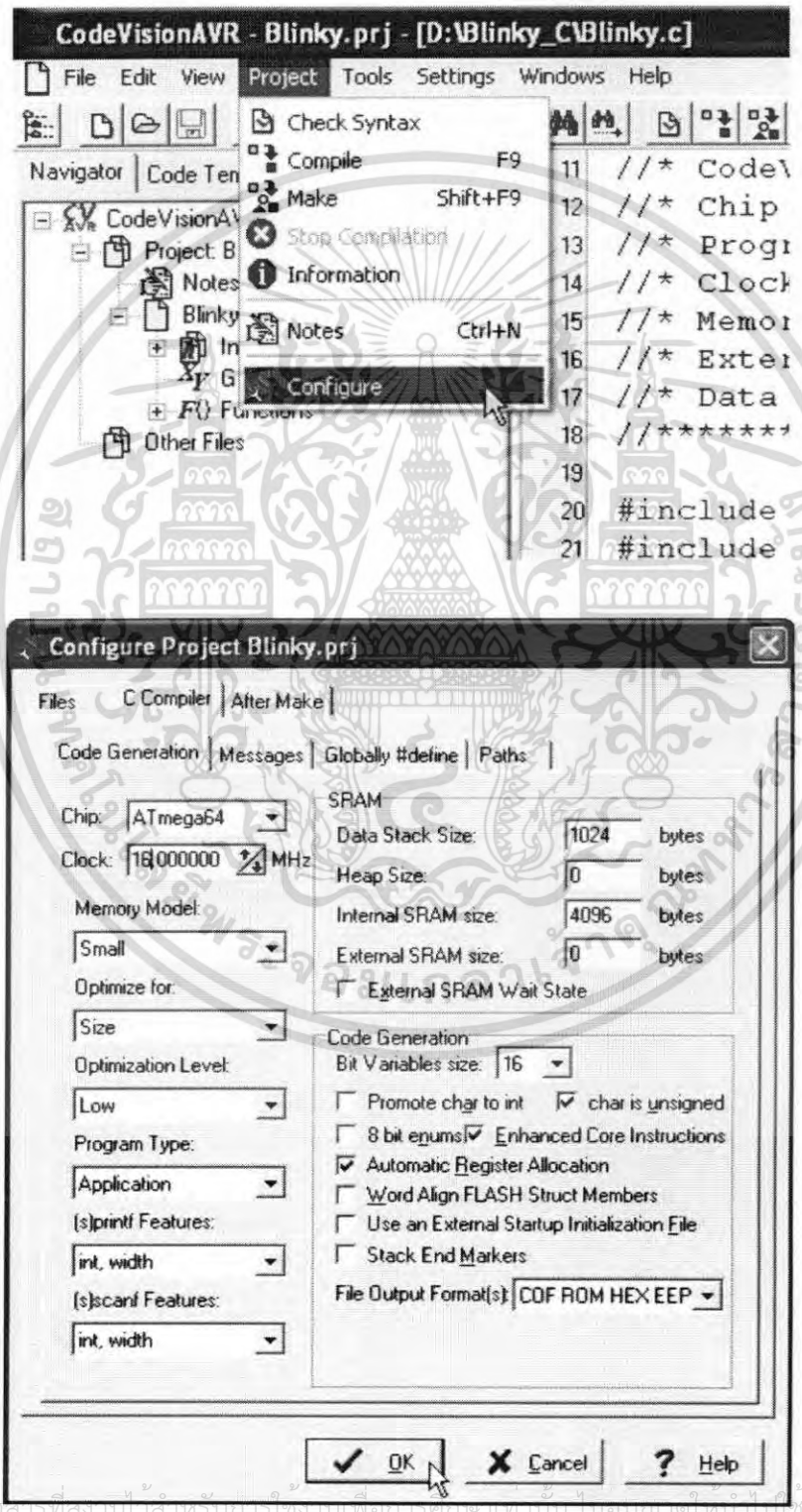


9. เมื่อทุกอย่างเรียบร้อยคลิกปุ่ม OK ดังรูป



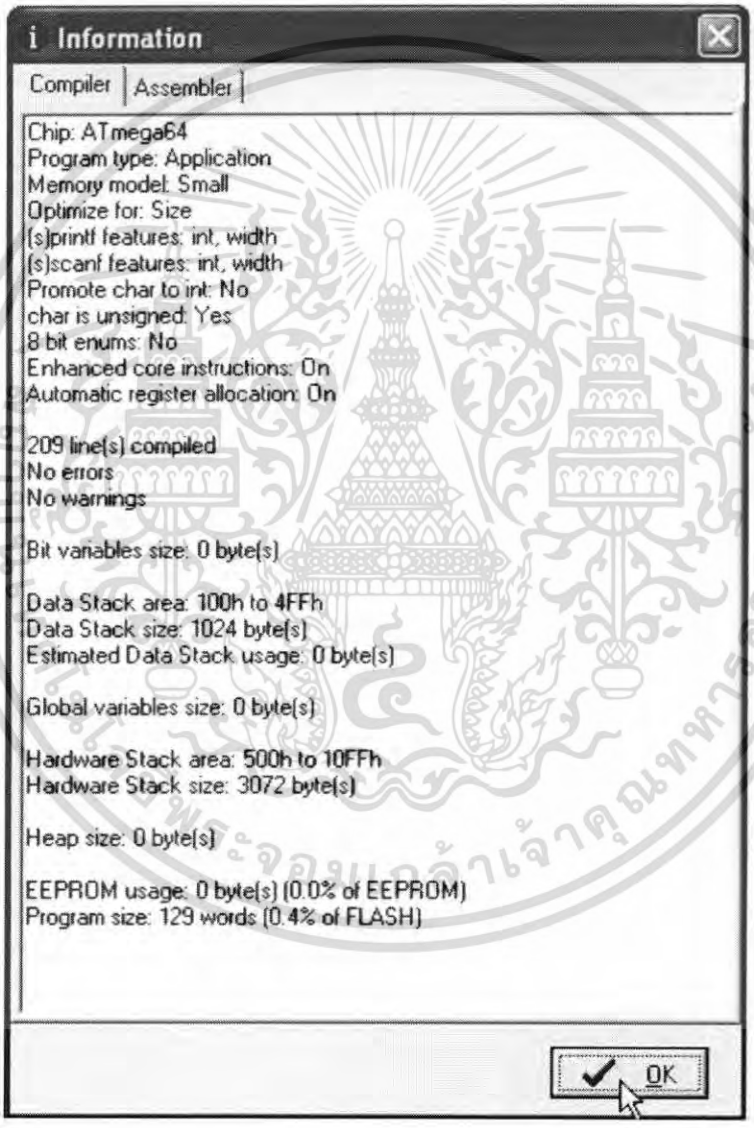
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ทำการตั้งค่าต่างของโปรเจกต์โดยการคลิกเมาส์ที่เมนูคำสั่ง Project □ Configure จากนั้น ทำการกำหนดเบอร์ MCU เป็น ATmega64 ค่าคริสตอลเท่ากับ 16.000000 MHz และ File OutputFormat(s) เป็น COF ROM HEX EEP



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ให้ทำการสั่งแปลโปรแกรมที่เราเขียนขึ้น โดยการคลิกเมาส์ที่เมนูคำสั่ง Project □ Make ซึ่งหลังจากแปลโปรแกรมแล้วได้ผลถูกต้องและไม่เกิดข้อผิดพลาดใด ๆ จะปรากฏข้อความ No errors, No warnings ต่อจากนี้ผู้ใช้ก็สามารถนำ Hex File ที่ได้จากสั่งแปลโปรแกรมนี้ไปทำการ Download ลง MCU ได้ทันที



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ตัวอย่างโปรแกรมที่ใช้ในการควบคุมการติดต่อสื่อสารระหว่างอุปกรณ์ทั้ง 2 ชิ้น

กำหนดค่าต่างๆและฟังก์ชันที่ต้องใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดฟังก์ชันและลักษณะการติดต่อสื่อสารระหว่างอุปกรณ์ทั้ง 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function ที่ใช้

- delay() ใช้หน่วงเวลาการทำงานของ microcontrollor ให้กับ zigbee module
- check_ok() ยืนยันสถานภาพการทำงานของตัวโมดูล
- to_default() คืนค่าสถานะต่างๆ ภายในตัวโมดูล สู่ค่าตั้งต้นจากโรงงาน (ไปหาเพิ่มจาก manual)
- scan() ทำการสำรวจ zigbee module ที่อยู่รอบๆ ทั้งหมด พร้อมทั้งบันทึกค่าต่างๆ ลงไว้ใน memory ของ microcontroller แล้วส่งผลสู่จอ led
- connect() ทำการปรับค่าต่างๆ ให้เหมาะสมกับการเชื่อมต่อกับ module อื่น

<u>ขั้นตอนการทำการเชื่อมต่อและส่งข้อมูล</u>

phase 1 to_defualt()

check_ok → led output

---------- delay

phase 2 scan() → led output

---------- delay

phase 3 connect() → led output

---------- delay

phase 4 transmit data

---------- delay

phase 5/1 disconnect and return to default

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

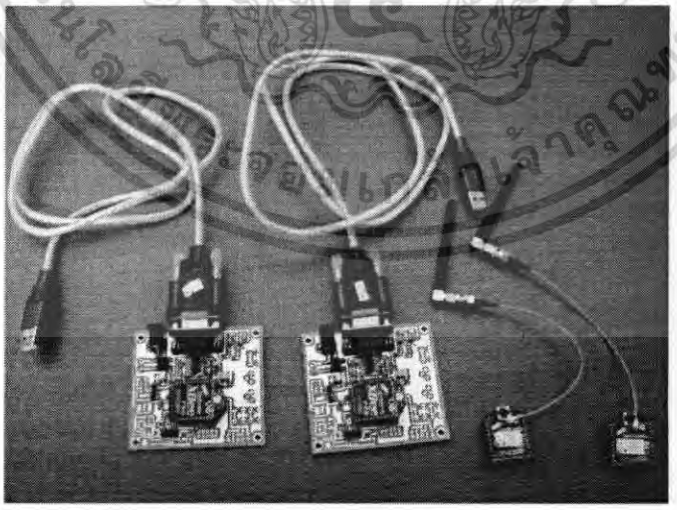
บทที่ 4
การทดลองและการวิเคราะห์

4.1 อุปกรณ์ที่ใช้ในการทดลอง

ในการที่จะสร้างระบบสมองกลฝังตัวระหว่างรถยนต์นั้นนั้น จะประกอบด้วย 2 ส่วนใหญ่ๆ ซึ่งก็คือ ตัวไมโครคอนโทรลเลอร์และตัวโมดูล เราต้องทราบการทำงานของทั้งสองส่วนนี้พร้อมทั้งทำการเชื่อมต่อกันแล้วสร้างระบบที่สัมพันธ์กันระหว่างทั้งสองส่วนนี้ ส่วนของโมดูลต้องศึกษา AT Command ของตัวโมดูล ส่วนตัวไมโครคอนโทรลเลอร์นั้นต้องศึกษาภาษา C หรือ Dynamic C แล้วก็เชื่อมต่อสองระบบนี้ด้วยพอร์ต UART

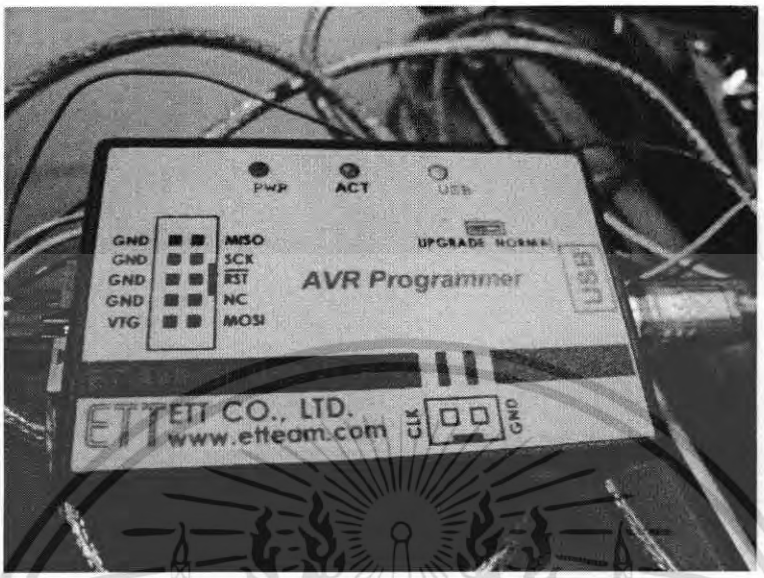
อุปกรณ์ที่ใช้ในการทดลอง

1. ตัวโมดูล ZigBee	2	ตัว
2. แผงวงจรควบคุม ZigBee	2	แผง
3. สายแปลงหัว USB to RS-232	2	เส้น
4. คอมพิวเตอร์	2	เครื่อง
5. เครื่องมือเขียนโปรแกรมลงบน ไมโครคอลโทรเลอร์	1	เครื่อง
6. แผงวงจร AVR + Zigbee	1	แผง



รูปที่ 4.1 อุปกรณ์ที่ใช้ในการทดลอง

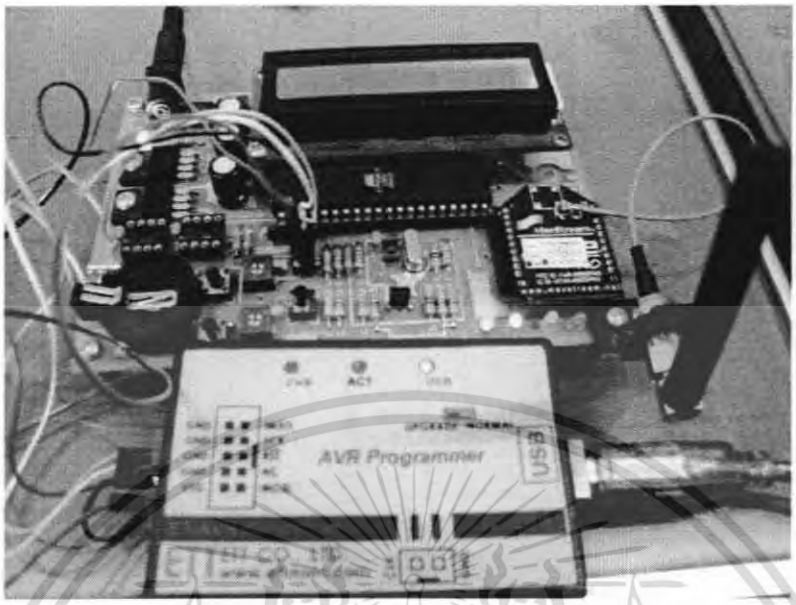
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 เครื่องมือเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์

รูปที่ 4.3 อุปกรณ์ที่ใช้ทำการทดลองเชื่อมต่อกับเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 การเชื่อมต่อระหว่างระบบสมองกลฝังตัวกับ เครื่องมือเขียนโปรแกรม

ในการทดลองขั้นแรกต้องทำการทดลองตัวโมดูลกับเครื่องคอมพิวเตอร์เสียก่อน เพื่อที่จะได้รู้ลำดับคำสั่งที่จะใช้ในอนาคต และเพื่อทดสอบและยืนยันการทำงานของตัวโมดูลด้วย

4.2 การทดทองการสูญเสียของสัญญาณภายในอาคาร

เมื่อเราทำการออกแบบระบบ นักออกแบบและวิศวกรมักต้องการที่จะทราบถึงความสามารถของระบบ ในสภาพแวดล้อมต่างๆ ถ้าเป็นในระบบจำพวก microprocessor คำถามเหล่านี้จะถูกตอบในรูปแบบของ clock speed คำสั่งต่อวินาทีหรือ throughput แต่เมื่อกล่าวถึงการสื่อสารไร้สายแน่นอนว่าต้องเรื่องรัศมีการสื่อสารเข้ามาเกี่ยวข้อง

การวัดค่าเชิงคุณภาพของระบบสื่อสารไร้สายมีตัวแปรมากมายขึ้นอยู่กับสภาพแวดล้อม ณ เวลานั้น วัตถุจำพวกต้นไม้ และอาคารต่างๆ หรือแม้แต่ความสูงของเสาอากาศล้วนอาจส่งผมต่อคุณภาพของสัญญาณ เพื่อที่จะทำการประมาณระยะทางการสื่อสารของระบบ เราต้องพิจารณาสี่ตัวแปรเหล่านี้

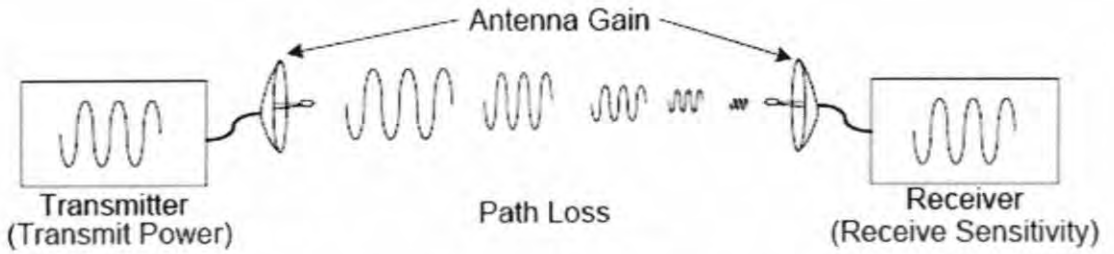
1.Transmit Power ซึ่งก็คือพลังงานที่ถูกส่งออกไปโดยตัวส่งสัญญาณ โดยปกติจะอยู่ในรูปแบบของ Watts หรือ Milliwatts

2. Receive Sensitivity ซึ่งก็คือค่าความแข็งแรงต่ำสุดที่ตัวรับ(receiver)จะสามารถรับได้

3. Antenna gain ซึ่งก็คือปริมาณ Signal Gain ของตัวเสาอากาศ

4. Path Loss ซึ่งก็คือค่าการสูญเสียเมื่อคลื่นวิทยุเดินทางผ่านอากาศและสิ่งกีดขวางต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 ลักษณะการสูญเสียของสัญญาณ

โดยปกติแล้วค่า Path Loss หรือ Attenuation นั้นจะแปรตามระยะทาง สิ่งกีดขวางระหว่างตัวรับ (receiver) และตัวส่ง (transmitter) ก็เช่นกัน ปริมาณการสูญเสียนั้นจะแปรผันตามความถี่ของคลื่นวิทยุ ประเภทวัตถุที่คลื่นกระทบ และความหนาแน่นของวัตถุ โดยทั่วไปแล้วคลื่นวิทยุ ณ ความถี่ต่ำๆจะสามารถเดินทางผ่านอากาศและวัตถุได้ดีกว่า ถ้ามีระบบรับส่ง 2 ระบบที่มีความถี่การส่งที่ 900 MHz และอีกระบบที่ 2.4 GHz แล้ว ระบบ 900 MHz จะมี path loss น้อยกว่าระบบ 2.4 GHz ค่าเหล่านี้จะถูกนำไปใช้ในการประมาณค่าระยะทางการส่งข้อมูล

Link Margin = Transmit Power – Receiver Sensitivity + Antenna Gain – Path Loss

ตารางที่ 4.1 ค่าการสูญเสียของ MaxStream (Friis Transmission Equation)

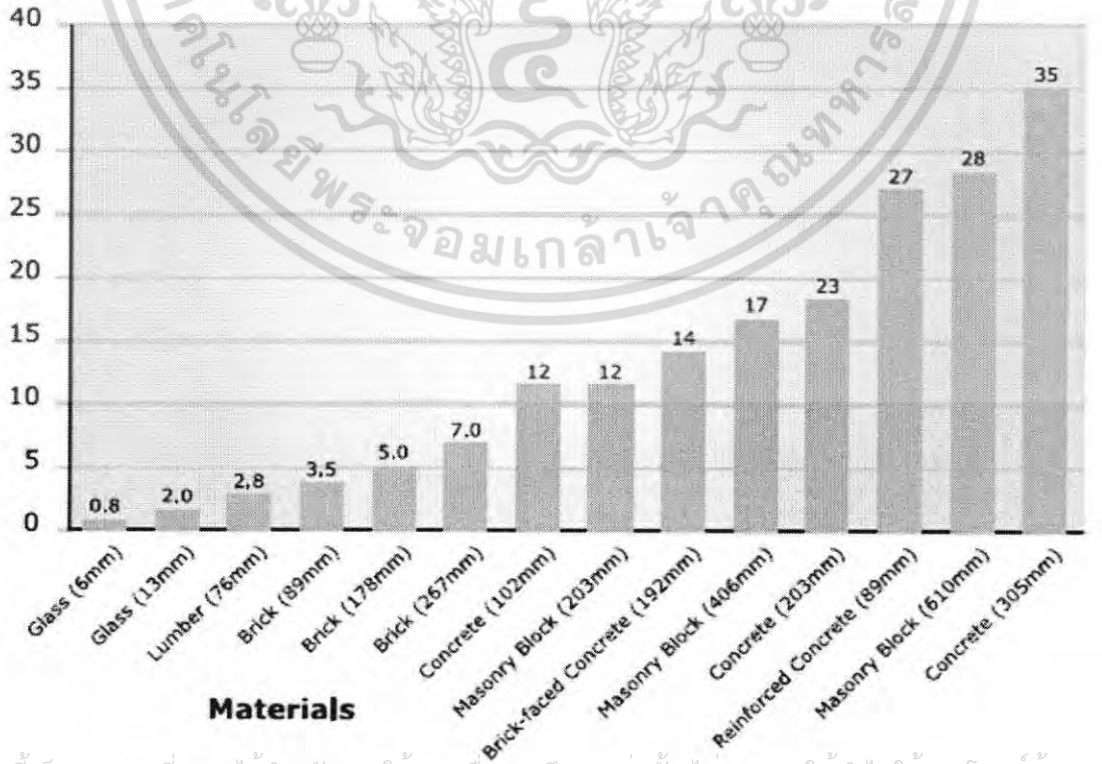
Distance	900 MHz free-space loss	2.4 GHz free-space loss
10 meters	72.5 dB	81 dB
100 meters	92.5 dB	101 dB
1000 meters	112.5 dB	121 dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสื่อสารภายในอาคาร

ตารางที่ 4.2 ค่าการสูญเสียของคลื่นผ่านวัตถุต่างๆ

Material	Attenuation @ 900 MHz
Glass 0.25" (6mm)	0.8 dB
Glass 0.5" (13mm)	2 dB
Lumber 3" (76mm)	2.8 dB
Brick 3.5" (89mm)	3.5 dB
Brick 7" (178mm)	5 dB
Brick 10.5" (267mm)	7 dB
Concrete 4" (102mm)	12 dB
Masonry Block 8" (203mm)	12 dB
Brick faced concrete 7.5 " (192mm)	14dB
Masonry Block 16" (406mm)	17dB
Concrete 8" (203mm)	23dB
Reinforced Concrete 3.5" (203mm)	27dB
Masonry Block 24" (610mm)	28dB
Concrete 12" (305mm)	35dB



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากในตารางเราจะพบได้ว่าไม่มีวัตถุจำพวกโลหะเพราะคลื่นวิทยุไม่สามารถผ่านโลหะได้ ในทางปฏิบัติ มวลของวัตถุจำพวกโลหะที่อยู่ในวิถีการเดินทางของคลื่น เช่น หอคอย รั้ว หรือแม้แต่รถยนต์ จะทำหน้าที่ในลักษณะสิ่งสะท้อนและกระจายคลื่นวิทยุ

ตัวอย่าง

สมมุติสถานการณ์ให้คลื่นวิทยุติดต่อสื่อสารกับเป้าหมายที่อยู่ห่างออกไป 100 เมตร ผ่านกำแพงหินมาตรฐาน 4 ชั้นและกำแพงคอนกรีต 1 ชั้น จากตารางแสดงให้เห็นว่าการสูญเสียในที่ว่างสำหรับระยะทาง 100 เมตรที่ความถี่ 900MHz นั้นอยู่ที่ 93dB ในสถานการณ์แบบนี้เราต้องคิดถึงผลกระทบที่กำแพงจะมีต่อการสื่อสารด้วย ห้องนี้มีกำแพงหินประกอบด้วยหิน 2 ชั้น(ชั้นละ -0.8dB) และไม้ (ชั้นละ -2.8dB) ซึ่งรวมแล้วมีค่าการสูญเสียทั้งหมด 4.4dB ต่อกำแพงหนึ่งชั้น กำแพงคอนกรีตหนา 102mm และมีค่าการสูญเสีย 12dB เราใช้สมการ

Power (TX) - Sensitivity (RX) >= Signal Attenuation

132 dB >= 93 dB + (4 walls * 4.4 dB) + (12 dB)

13(transmit power 2dB >= 122.6 dB

เราจะพบว่า Power (TX) มีค่ามากกว่าค่าการสูญเสียทำให้สามารถมีการติดต่อสื่อสารเกิดขึ้นได้ แต่ค่าเหล่านี้เป็นเพียงแค่การประมาณค่าทางทฤษฎีเท่านั้น ดังนั้นควรมีการทดสอบเพื่อยืนยันสถานการณ์และค่าการประมาณ

สรุปผลการทดลองเรื่องการสูญเสีย

ในการที่จะหาความสามารถในการสื่อสารภายในอาคารนั้น ปัญหาหลักอยู่ที่การหาเส้นทางการเดินทางของคลื่นและสิ่งกีดขวางที่อยู่ระหว่างเครื่องรับและเครื่องส่ง โดยต้องคำนึงถึงชนิดของวัตถุและความหนาของวัตถุด้วย การทดทองตามสถานการณ์จริงนั้นเป็นเพียงวิธีการเดียวที่จะสามารถระบุได้ว่าสามารถที่จะสื่อสารระหว่างกันสำเร็จหรือไม่

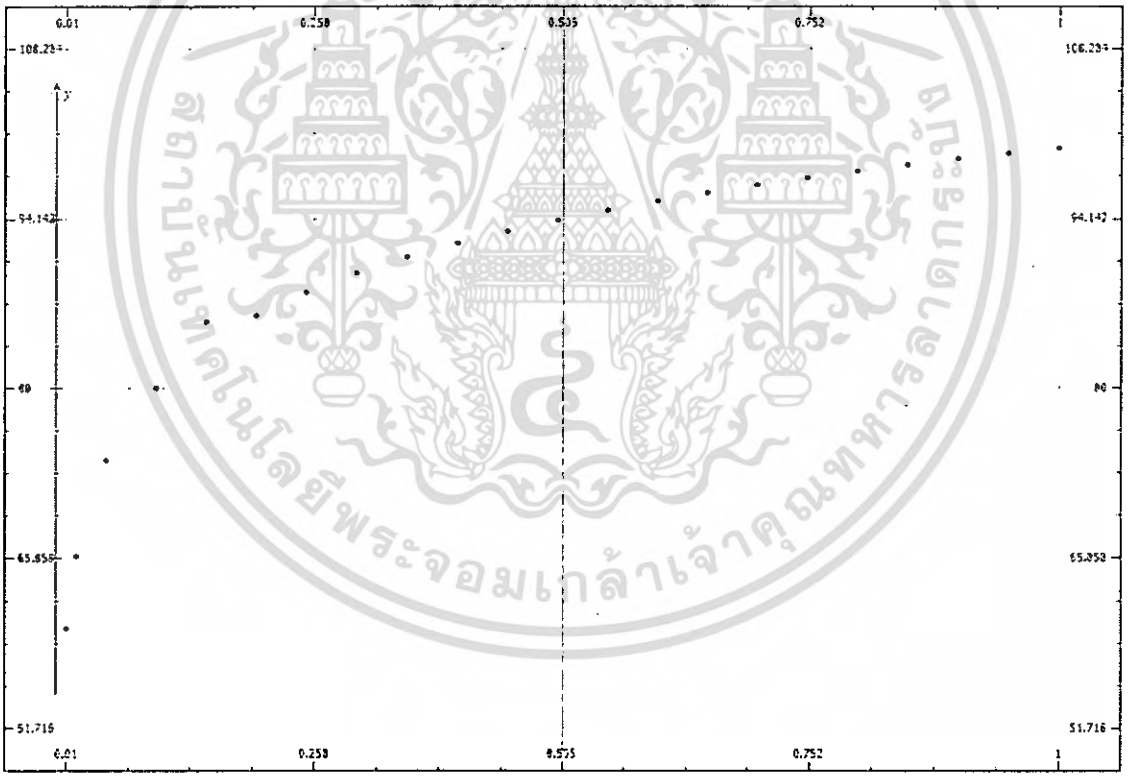
ค่าตัวกำลังการส่ง ค่าความอ่อนไหวของเครื่องรับ (sensitivity) และค่าความถี่ (frequency) ควรที่จะนำมาพิจารณาในระบบการสื่อสารไร้สายทุกระบบ การวางเสาอากาศและตัวส่งวิทยุอาจจะช่วยลดจำนวนสิ่งกีดขวางลงได้ แต่ในระบบส่วนใหญ่ผู้ออกแบบไม่สามารถที่จะระบุได้ถึงเส้นทางการเดินทางระบบและวัตถุที่มันต้องผ่าน เหลือแต่เพียงค่ากำลังส่ง ค่าความอ่อนไหวของเครื่องรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และค่า เกน ของเสาอากาศ (antenna gain) ที่เป็นค่าพารามิเตอร์ที่สามารถปรับค่าได้ในสถานการณ์จริง ถ้าการสื่อสารไม่มีความมั่นคงหรือมีค่าการสูญเสียมาก การเลือกลักษณะการสื่อสารอาจจะสำคัญต่อความน่าเชื่อถือในการสื่อสาร ในปัจจุบันมีการออกแบบโปรโตคอลที่สามารถที่จะช่วยกู้ข้อมูลเมื่อมีความเสียหายในระว่างการส่งข้อมูล แต่ไม่ว่าสถานการณ์จะเป็นเช่นไร การประมาณสถานการณ์ ประมาณค่าสูญเสียและการทดสอบระบบที่สถานที่จริงสามารถที่จะช่วยในการตั้งค่าระดับคุณภาพการสื่อสารในช่วงขั้นตอนการออกแบบระบบ

4.3 การทดทองการสูญเสียของสัญญาณในที่โล่ง (Free Space Loss)

การสูญเสียของสัญญาณไม่ได้มีเฉพาะในตัวอาคารเท่านั้นแต่ก็ยังเกิดขึ้นในที่โล่งได้เช่นกัน ถึงแม้จะไม่มากเท่า และจากการทดลองพบว่ามีการสูญเสียดังกราฟ



รูปที่ 4.6 ค่าการสูญเสียของสัญญาณต่อระยะทาง

โดยมี แกน x เป็น ค่าระยะห่างระหว่างอุปกรณ์ทั้ง 2 ตัว มีหน่วยเป็น km

แกน y เป็น ค่าการสูญเสียของสัญญาณระหว่างการส่ง มีหน่วยเป็น db

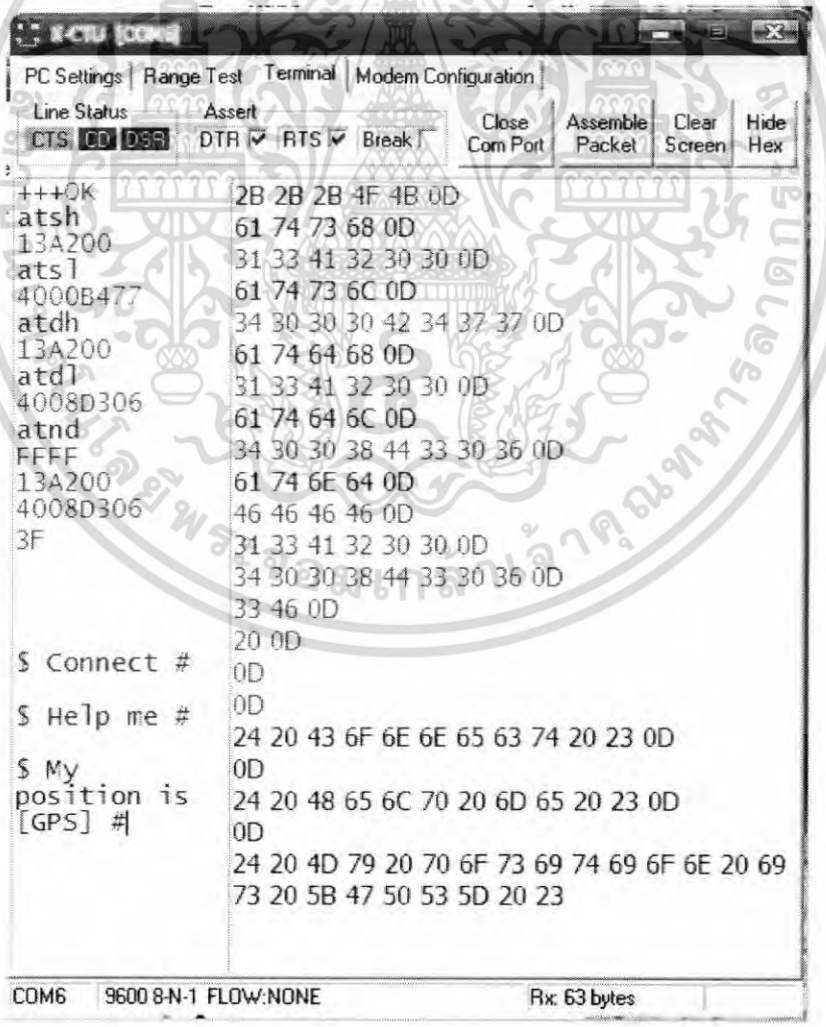
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองเชื่อมต่อกับระบบสมองกลฝังตัว

จากการทดลองที่ผ่านมาทำให้เราทราบค่าความสูญเสียที่เกิดขึ้นกับการติดต่อระหว่างอุปกรณ์สื่อสาร Zigbee แต่ในการทดลองต่อไปนี้จะแสดงการติดต่อสื่อสารระหว่างอุปกรณ์สมองกลอัจฉริยะที่ติดต่อกับอุปกรณ์สื่อสาร Zigbee

โดยมีขั้นตอนในการทำการทดลองดังนี้

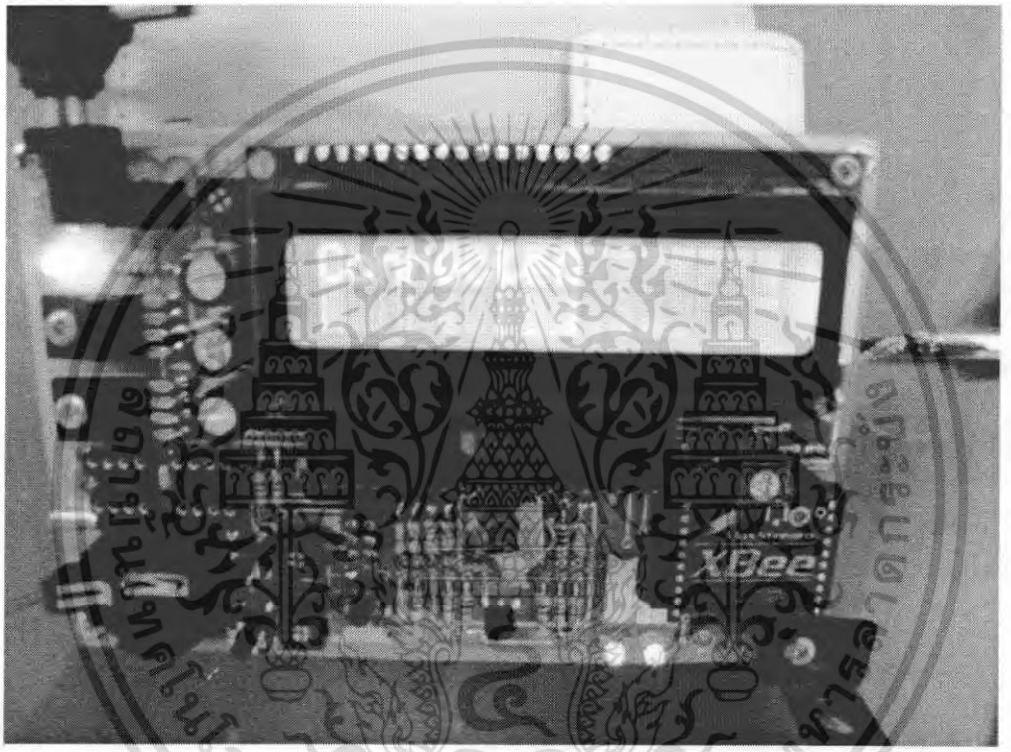
1. ในด้านของอุปกรณ์สื่อสาร Zigbee จะเป็นฝั่งอุปกรณ์ที่เชื่อมต่อและสั่งงานผ่านคอมพิวเตอร์
2. ในขั้นทดลองนี้จะกำหนดให้ทำการส่งค่าการเชื่อมต่อกับระบบสมองกลอัจฉริยะ แล้วทำการส่งข้อความขอความช่วยเหลือในรูปแบบของชุดข้อมูลไปยังระบบสมองกลอัจฉริยะ



รูปที่ 4.7 การกำหนดค่าการเชื่อมต่อและชุดข้อมูลที่ส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

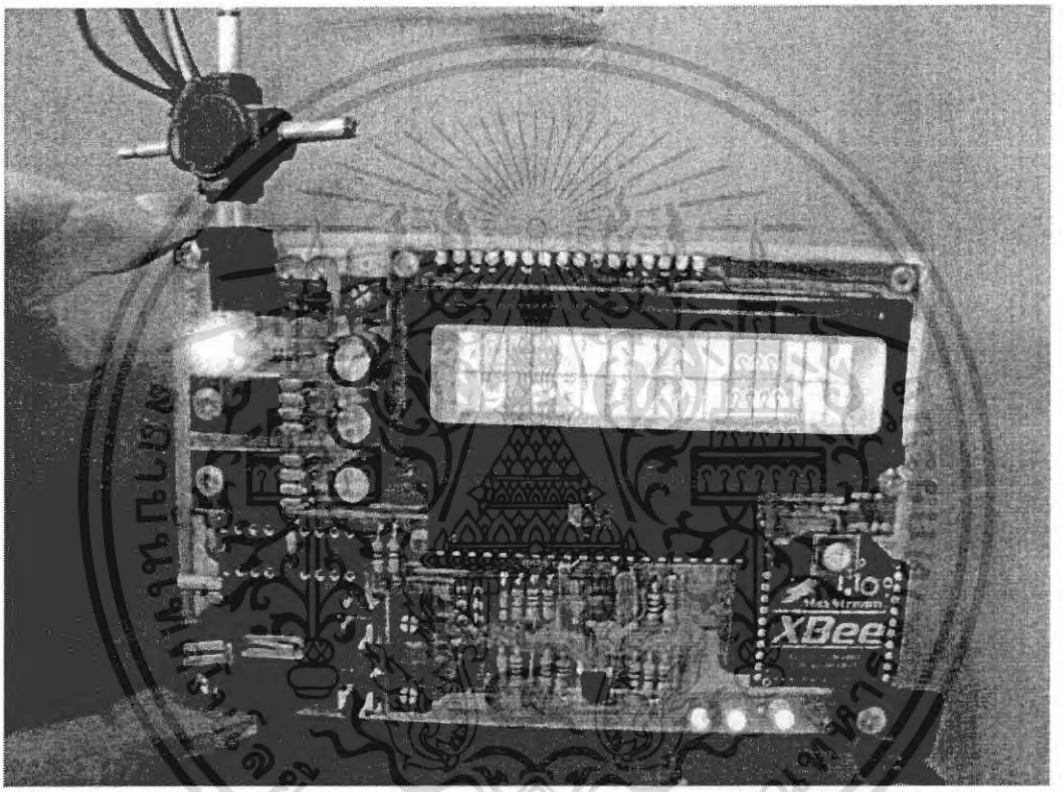
3. ในฝั่งของระบบสมองกลอัจฉริยะที่ได้ทำการกำหนดโปรแกรมให้ทำการเชื่อมต่อกับอุปกรณ์ Zigbee ที่อยู่ในรัศมีขอบค่ายของสัญญาณ เมื่อพบแล้วให้ทำการเชื่อมต่อ จากนั้นจึงจะทำการรับส่งข้อมูลที่ได้จากอุปกรณ์ Zigbee โดยเขียนโปรแกรมตรวจสอบหากทำการเชื่อมต่อสำเร็จให้แสดงผลที่หน้าจอ LCD ปรากฏคำว่า connect



รูปที่ 4.8 ข้อความเมื่ออุปกรณ์ Zigbee ทำการเชื่อมต่อเสร็จเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. นำชุดข้อมูลที่ได้นี้ไปแสดงผลให้ปรากฏขึ้นบนหน้าจอ LCD ให้ผู้ใช้ได้รับทราบลักษณะของข้อมูล รวมทั้งลักษณะของการเชื่อมต่อที่กระทำด้วย



รูปที่ 4.9 แสดงผลการส่งข้อมูลระหว่างอุปกรณ์ Zigbee กับ Embedded System

5. เมื่อทำการสื่อสารแลกเปลี่ยนข้อมูลถึงกันเรียบร้อยแล้วโปรแกรมจะทำการตัดการติดต่อสื่อสารระหว่างอุปกรณ์ทั้ง 2 ทันที เพื่อที่จะรอทำการสื่อสารกับอุปกรณ์ Zigbee ตัวอื่นๆต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินโครงงาน

5.1 สรุปผลการดำเนินโครงงาน

ปริญญานิพนธ์ฉบับนี้ ได้นำวิธีการสื่อสารระหว่างตัวโมดูล (ZigBee) เพื่อให้เกิดการสื่อสารระหว่างตัวรถยนต์ในท้องถนน โดยจะทำการเชื่อมต่อเพื่อส่งข้อมูลที่จำเป็นแก่รถยนต์บริเวณรอบๆก่อน แล้วเลือกสร้างเครือข่ายส่วนตัวได้ถ้าต้องการ อีกทั้งสามารถส่งข้อความฉุกเฉินในเวลาจำเป็น พร้อมทั้งสามารถที่จะระบุหาโมดูลเฉพาะในเครือข่ายได้ ทำให้สามารถหาโมดูลที่ต้องการได้ง่าย เพราะโมดูลแต่ละตัวมีเอกลักษณ์เฉพาะตัว

เนื่องจากตัวโมดูลเองมีจุดเด่นที่กินพลังงานในระดับที่ต่ำมาก สามารถทำงานเป็นปีๆ ด้วยการบรรจุแบตเตอร์รี่เพียงครั้งเดียวทำให้สามารถใช้งานแม้กระทั่งตอนที่รถจอดอยู่หรือไม่ได้ใช้งานอยู่ แต่ก็สามารถที่จะมีส่วนช่วยเหลือในการเพิ่มอาณาเขตให้กับตัวเครือข่ายด้วย ทั้งตัวโมดูลทั้งตัวไมโครคอนโทรลเลอร์สามารถที่จะเพิ่มพอร์ต I/O ได้ทำให้มีความยืดหยุ่นในการใช้งาน สามารถส่งข้อมูลที่ตัวเองมีออกไปได้หลากหลายขึ้น ตั้งแต่ สถานะของรถ ความเร็ว และอื่นๆ

ในส่วนการทำงานในเทอมนี้ได้ทำการศึกษาคุณสมบัติและความสามารถในการใช้งานของโมดูล Zigbee และเลือกคำสั่งในการใช้งานให้เหมาะสม อีกทั้งยังออกแบบและทดลองวงจรที่จะนำมาทดลองปฏิบัติจริงในการสื่อสารข้อมูลระหว่างอุปกรณ์ทั้ง 2 ตัว หรือ ระบบสมองกลฝังตัวนั่นเอง

การทำงานในเทอมที่ 2 จะเป็นการทำงานทั้งสองส่วนของโครงงาน โดยในส่วนแรกจะเป็นการพัฒนา ปรับแก้ และสร้างงานในส่วนของฮาร์ดแวร์หรือระบบสมองกลอัจฉริยะ(Embedded System) และเลือกอุปกรณ์ที่จะนำมาใช้ในการเขียนโปรแกรมลงบนไมโครคอลโทรเลอร์ AVR ศึกษาเพิ่มเติมเกี่ยวกับการทำงานและใช้งานส่วนที่เกี่ยวข้องกับโครงการทั้งหมด และในส่วนที่สองจะเป็นการศึกษา และสร้างโปรแกรมที่นำมาใช้ควบคุมการทำงานของไมโครคอลโทรเลอร์เพื่อจัดการกับข้อมูลที่ได้รับมาจากโมดูล Zigbee และนำไปแสดงผลที่หน้าจอ LCD

โดยภาพรวมของโครงการนี้ยังสามารถนำไปพัฒนาต่อโดยใช้ร่วมกับ GPS เพื่อใช้ระบุตำแหน่งของรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 แนวทางการพัฒนาโครงงาน

1. พัฒนาระบบ I/O ของทั้งตัวโมดูลและของไมโครคอนโทรลเลอร์ให้สามารถรับอินพุทจากอุปกรณ์หลายตัวได้ พร้อมทั้งส่งเอาท์พุทออกไป โดยในเทอมหน้าจะทำการเพิ่มคีบอร์ดและหน่วยความจำเพื่อเป็นการจำลองระบบสมองกลฝังตัว

2. พัฒนาระบบเครือข่ายของตัวโมดูลให้สามารถรองรับและสร้างระบบเครือข่ายที่มีความซับซ้อนขึ้นกว่านี้ได้ โดยจะทำการเพิ่มการเชื่อมต่อจากโมดูล 2 ตัว ให้สามารถทำการเชื่อมต่อกับโมดูลตั้งแต่ 3 ตัวขึ้นไปได้

3. ออกแบบระบบอินเตอร์เฟซให้สามารถใช้งานได้ง่าย โดยจะอยู่ในช่วงการพัฒนาโปรแกรมที่เชื่อมต่อกับส่วนควบคุม

4. ออกแบบการเชื่อมต่อระหว่างระบบสมองกลฝังตัว กับ infrastructure

5. ออกแบบและทดลองในส่วนของซอฟท์แวร์ให้มีความสมบรูณ์มากขึ้น มีฟังก์ชั่นการทำงานที่หลากหลาย รองรับการใช้งานที่เป็นประโยชน์ และสามารถใช้งานได้จริง

6. ในอนาคตควรสนับสนุนให้มีการพัฒนาให้ใช้ร่วมกับระบบ GPS ซึ่งสามารถระบุตำแหน่งให้แก่รถยนต์ที่ใช้งานได้ จะเกิดประโยชน์และได้ลักษณะการทำงานที่หลากหลายและครอบคลุมมากขึ้น

7. ออกแบบลักษณะภายนอกให้เหมาะสมและเพิ่มความสวยงามให้แก่อุปกรณ์เพื่อดึงดูดผู้ใช้

8. ออกแบบฟังก์ชันการใช้งานให้ง่าย สะดวก และเกิดประโยชน์แก่ผู้ใช้สูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. **http://www.maxstream.net/support/knowledgebase/full-list.php**
2. **http://www.wara.com/modules.php**
3. **http://www.zigbee.org/en/index.asp**
4. **http://www.palowireless.com/zigbee/articles.asp**
5. **http://www.embedded-th.com/x/index.php**
6. **http://te.eng.dpu.ac.th/tj53-Triratana.pdf**
7. **http://www.adslcool.com/network/viewrecord.php?id=80**
8. **http://www.vcharkarn.com/include/article/showarticle.php?Aid=17947&page=1**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XBee™/XBee-PRO™ OEM RF Modules

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

Product Manual v1.xAx - 802.15.4 Protocol
For OEM RF Module Part Numbers: XB24-...-001, XBP24-...-001

IEEE® 802.15.4 OEM RF Modules by MaxStream, Inc.

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 support)

M100232
2006.10.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. XBee/XBee-PRO OEM RF Modules

The XBee and XBee-PRO OEM RF Modules were engineered to meet 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 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

Long Range Data Integrity

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

Source/Destination Addressing

Unicast & Broadcast Communications

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

Coordinator/End Device operations

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: 215 mA (@3.3 V)
- RX Current: 55 mA (@3.3 V)
- Power-down Current: < 10 µA

ADC and I/O line support

Analog-to-digital conversion, Digital I/O

I/O Line Passing

Easy-to-Use

No configuration necessary for out-of box RF communications

Free X-CTU Software (Testing and configuration software)

AT and API Command Modes for configuring module parameters

Extensive command set

Small form factor

Free & Unlimited RF-XPert Support

1.1.1. Worldwide Acceptance

FCC Approval (USA) Refer to Appendix A [p57] for FCC Requirements. Systems that contain XBee/XBee-PRO RF Modules inherit MaxStream 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 the **United States**, **Canada**, **Australia**, **Israel and Europe**. Contact MaxStream for complete list of government agency approvals.

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 (software selectable)	1mW (0 dBm)	60 mW (18 dBm) conducted, 100 mW (20 dBm) EIRP*
RF Data Rate	250,000 bps	250,000 bps
Serial 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)	45mA (@ 3.3 V)	If PL=0 (10dBm): 137mA(@3.3V), 139mA(@3.0V) PL=1 (12dBm): 155mA (@3.3V), 153mA(@3.0V) PL=2 (14dBm): 170mA (@3.3V), 171mA(@3.0V) PL=3 (16dBm): 188mA (@3.3V), 195mA(@3.0V) PL=4 (18dBm): 215mA (@3.3V), 227mA(@3.0V)
Idle / Receive Current (typical)	50mA (@ 3.3 V)	55mA (@ 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	Integrated Whip, Chip or U.FL Connector	Integrated Whip, Chip or U.FL Connector
Networking & Security		
Supported Network Topologies	Point-to-point, Point-to-multipoint & Peer-to-peer	
Number of Channels (software selectable)	16 Direct Sequence Channels	12 Direct Sequence Channels
Addressing Options	PAN ID, Channel and Addresses	PAN ID, Channel and Addresses
Agency Approvals		
United States (FCC Part 15.247)	OUR-XBEE	OUR-XBEEPRO
Industry Canada (IC)	4214A XBEE	4214A XBEEPRO
Europe (CE)	ETSI	ETSI (Max. 10 dBm transmit power output)*
Japan	n/a	005NYCA0378 (Max. 10 dBm transmit power output)**

* When operating in Europe: XBee-PRO RF Modules must be configured to operate at a maximum transmit power output level of 10 dBm. The power output level is set using the PL command. The PL parameter must equal "0" (10 dBm).

Additionally, European regulations stipulate an EIRP power maximum of 12.86 dBm (19 mW) for the XBee-PRO and 12.11 dBm for the XBee when integrating high-gain antennas.

** When operating in Japan: Transmit power output is limited to 10 dBm. A special part number is required when ordering modules approved for use in Japan. Contact MaxStream for more information [call 1-801-765-9885 or send e-mails to sales@maxstream.net].

Antenna Options: The ranges specified are typical when using the integrated Whip (1.5 dBi) and Dipole (2.1 dBi) antennas. The Chip antenna option provides advantages in its form factor; however, it typically yields shorter range than the Whip and Dipole antenna options when transmitting outdoors. For more information, refer to the "XBee Antenna" application note located on MaxStream's web site (http://www.maxstream.net/support/knowledgebase/article.php?kb=153).

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3. Mechanical Drawings

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

1.4. Mounting Considerations

The XBee/XBee-PRO RF Module was designed to mount into a receptacle (socket) and therefore does not require any soldering when mounting it to a board. The XBee Development Kits contain RS-232 and USB interface boards which use two 20-pin receptacles to receive modules.

Figure 1-02. XBee Module Mounting to an RS-232 Interface Board

The receptacles used on MaxStream development boards are manufactured by Century Interconnect. Several other manufacturers provide comparable mounting solutions; however, MaxStream currently uses the following receptacles:

- Through-hole single-row receptacles -
  Samtec P/N: MMS-110-01-L-SV (or equivalent)
- Surface-mount double-row receptacles -
  Century Interconnect P/N: CPRMSL20-D-0-1 (or equivalent)
- Surface-mount single-row receptacles -
  Samtec P/N: SMM-110-02-SM-S

MaxStream also recommends printing an outline of the module on the board to indicate the orientation the module should be mounted.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5. Pin Signals

Figure 1-03. XBee/XBee-PRO RF Module Pin Numbers
(top sides shown - shields on bottom)

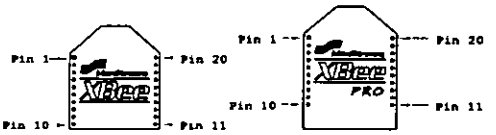


Table 1-02. Pin Assignments for the XBee and XBee-PRO Modules
(Low-asserted signals are distinguished with a horizontal line above signal name.)

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	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	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
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

* Function is not supported at the time of this release

Design Notes:

- Minimum connections: VCC, GND, DOUT & DIN
- Minimum connections for updating firmware: VCC, GND, DIN, DOUT, RTS & DTR
- Signal Direction is specified with respect to the module
- Module includes a 50k Ω pull-up resistor attached to RESET
- Several of the input pull-ups can be configured using the PR command
- Unused pins should be left disconnected

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6. Electrical Characteristics

Table 1-03. DC Characteristics (VCC = 2.8 - 3.4 VDC)

Symbol	Characteristic	Condition	Min	Typical		Max	Unit
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	µA
II_{OZ}	High Impedance Leakage Current	V_{IN} = VCC or GND, all I/O High-Z, per pin	-	0.025		1	µA
TX	Transmit Current	VCC = 3.3 V	-	45 (XBee)	215 (PRO)	-	mA
RX	Receive Current	VCC = 3.3 V	-	50 (XBee)	55 (PRO)	-	mA
PWR-DWN	Power-down Current	SM parameter = 1	-	< 10		-	µA

Table 1-04. ADC Characteristics (Operating)

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
V_{REFH}	VREF - Analog-to-Digital converter reference range		2.08	-	V_{DDAD}	V
I_{REF}	VREF - Reference Supply Current	Enabled	-	200	-	µA
		Disabled or Sleep Mode	-	< 0.01	0.02	µA
V_{INDC}	Analog Input Voltage[1]		V_{SSAD} - 0.3	-	V_{DDAD} + 0.3	V

1. Maximum electrical operating range, not valid conversion range.

Table 1-05. ADC Timing/Performance Characteristics[1]

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
R_{AS}	Source Impedance at Input[2]		-	-	10	kΩ
V_{AIN}	Analog Input Voltage[3]		V_{REFL}		V_{REFH}	V
RES	Ideal Resolution (1 LSB)[4]	2.08V ≤ V_{DDAD} ≤ 3.6V	2.031	-	3.516	mV
DNL	Differential Non-linearity[5]		-	±0.5	±1.0	LSB
INL	Integral Non-linearity[6]		-	±0.5	±1.0	LSB
E_{ZS}	Zero-scale Error[7]		-	±0.4	±1.0	LSB
E_{FS}	Full-scale Error[8]		-	±0.4	±1.0	LSB
E_{IL}	Input Leakage Error[9]		-	±0.05	±5.0	LSB
E_{TU}	Total Unadjusted Error[10]		-	±1.1	±2.5	LSB

1. All ACCURACY numbers are based on processor and system being in WAIT state (very little activity and no IO switching) and that adequate low-pass filtering is present on analog input pins (filter with 0.01 µF to 0.1 µF capacitor between analog input and VREFL). Failure to observe these guidelines may result in system or microcontroller noise causing accuracy errors which will vary based on board layout and the type and magnitude of the activity.

Data transmission and reception during data conversion may cause some degradation of these specifications, depending on the number and timing of packets. It is advisable to test the ADCs in your installation if best accuracy is required.

2. R_{AS} is the real portion of the impedance of the network driving the analog input pin. Values greater than this amount may not fully charge the input circuitry of the ATD resulting in accuracy error.

3. Analog input must be between V_{REFL} and V_{REFH} for valid conversion. Values greater than V_{REFH} will convert to $3FF.

4. The resolution is the ideal step size or 1LSB = (V_{REFH}–V_{REFL})/1024

5. Differential non-linearity is the difference between the current code width and the ideal code width (1LSB). The current code width is the difference in the transition voltages to and from the current code.

6. Integral non-linearity is the difference between the transition voltage to the current code and the adjusted ideal transition voltage for the current code. The adjusted ideal transition voltage is (Current Code–1/2)*(1/(((V_{REFH}+E_{FS})–(V_{REFL}+E_{ZS}))).

7. Zero-scale error is the difference between the transition to the first valid code and the ideal transition to that code. The Ideal transition voltage to a given code is (Code–1/2)*(1/(V_{REFH}–V_{REFL})).

8. Full-scale error is the difference between the transition to the last valid code and the ideal transition to that code. The ideal transition voltage to a given code is (Code–1/2)*(1/(V_{REFH}–V_{REFL})).

9. Input leakage error is error due to input leakage across the real portion of the impedance of the network driving the analog pin. Reducing the impedance of the network reduces this error.

10. Total unadjusted error is the difference between the transition voltage to the current code and the ideal straight-line transfer function. This measure of error includes inherent quantization error (1/2LSB) and circuit error (differential, integral, zero-scale, and full-scale) error. The specified value of E_{TU} assumes zero E_{IL} (no leakage or zero real source impedance).

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. RF Module Operation

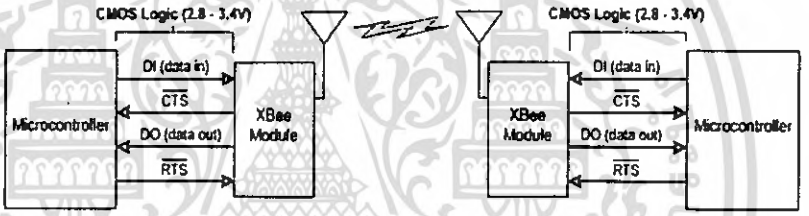
2.1. Serial Communications

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: Through a MaxStream proprietary RS-232 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. System Data Flow Diagram in a UART-interfaced environment
(Low-asserted signals distinguished with horizontal line over signal name.)

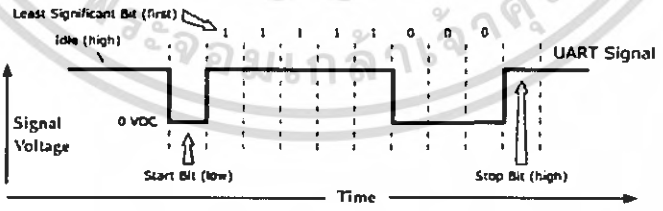


Serial Data

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).

© 2006 MaxStream, Inc.

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2. Transparent Operation

By default, XBee/XBee-PRO RF Modules operate in Transparent Mode. When operating in this mode, the modules act as a serial line replacement - all UART data received through the DI pin is queued up for RF transmission. When RF data is received, the data is sent out the DO pin.

Serial-to-RF Packetization

Data is buffered in the DI buffer until one of the following causes the data to be packetized and transmitted:

1. No serial characters are received for the amount of time determined by the RO (Packetization Timeout) parameter. If RO = 0, packetization begins when a character is received.
2. The maximum number of characters that will fit in an RF packet (100) is received.
3. The Command Mode Sequence (GT + CC + GT) is received. Any character buffered in the DI buffer before the sequence is transmitted.

If the module cannot immediately transmit (for instance, if it is already receiving RF data), the serial data is stored in the DI Buffer. The data is packetized and sent at any RO timeout or when 100 bytes (maximum packet size) are received.

If the DI buffer becomes full, hardware or software flow control must be implemented in order to prevent overflow (loss of data between the host and module).

2.1.3. API Operation

API (Application Programming Interface) Operation is an alternative to the default Transparent Operation. The frame-based API extends the level to which a host application can interact with the networking capabilities of the module.

When in API mode, all data entering and leaving the module is contained in frames that define operations or events within the module.

Transmit Data Frames (received through the DI pin (pin 3)) include:

- RF Transmit Data Frame
- Command Frame (equivalent to AT commands)

Receive Data Frames (sent out the DO pin (pin 2)) include:

- RF-received data frame
- Command response
- Event notifications such as reset, associate, disassociate, etc.

The API provides alternative means of configuring modules and routing data at the host application layer. A host application can send data frames to the module that contain address and payload information instead of using command mode to modify addresses. The module will send data frames to the application containing status packets; as well as source, RSSI and payload information from received data packets.

The API operation option facilitates many operations such as the examples cited below:

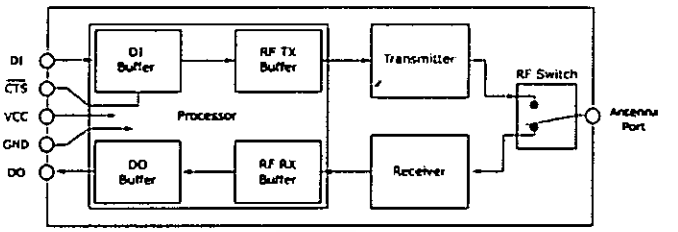
-> Transmitting data to multiple destinations without entering Command Mode
-> Receive success/failure status of each transmitted RF packet
-> Identify the source address of each received packet

To implement API operations, refer to API sections [p52].

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4. 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 ($\overline{CTS}$). When the DI buffer is 17 bytes away from being full; by default, the module de-asserts $\overline{CTS}$ (high) to signal to the host device to stop sending data [refer to D7 (DIO7 Configuration) parameter]. $\overline{CTS}$ 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.

Refer to the RO (Packetization Timeout), BD (Interface Data Rate) and D7 (DIO7 Configuration) command descriptions for more information.

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 ($\overline{RTS}$). If $\overline{RTS}$ is enabled for flow control (D6 (DIO6 Configuration) Parameter = 1), data will not be sent out the DO Buffer as long as $\overline{RTS}$ (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.

Refer to the D6 (DIO6 Configuration) command description for more information.

เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. RF Module Configuration

3.1. Programming the RF Module

Refer to the Command Mode section [p24] for more information about entering Command Mode, sending AT commands and exiting Command Mode. For information regarding module programming using API Mode, refer to the API Operation sections [p52].

3.1.1. Programming Examples

Refer to the 'X-CTU' section of the Development Guide (Appendix B) for more information regarding the X-CTU configuration software.

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 with 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/support/downloads.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, the BD parameter = 3 (which corresponds to 9600 bps).

Sample Configuration: Modify RF Module Destination Address

Example: Utilize the X-CTU "Terminal" tab 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, OK, OK <CR> (Command execution is triggered upon each instance of the comma)

Sample Configuration: Restore RF Module Defaults

Example: Utilize the X-CTU "Modem Configuration" tab to restore default parameter values.

After establishing a connection between the 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.

ไม่วากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2. Command Reference Tables

XBee/XBee-PRO RF Modules expect numerical values in hexadecimal. Hexadecimal values are designated by a "0x" prefix. Decimal equivalents are designated by a "d" suffix. Commands are contained within the following command categories (listed in the order that their tables appear):

- Special
- Networking & Security
- RF Interfacing
- Sleep (Low Power)
- Serial Interfacing
- I/O Settings
- Diagnostics
- AT Command Options

All modules within a PAN should operate using the same firmware version.

Special

Table 3-01. XBee-PRO Commands - Special

AT Command	Command Category	Name and Description	Parameter Range	Default
WR	Special	**Write.** Write parameter values to non-volatile memory so that parameter modifications persist through subsequent power-up or reset. Note: Once WR is issued, no additional characters should be sent to the module until after the response "OK\r" is received.	-	-
RE	Special	**Restore Defaults.** Restore module parameters to factory defaults.	-	-
FR (v1.x80*)	Special	**Software Reset.** Responds immediately with an OK then performs a hard reset ~100ms later.	-	-

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

Networking & Security

Table 3-02. XBee/XBee-PRO Commands - Networking & Security (Sub-categories designated within {brackets})

AT Command	Command Category	Name and Description	Parameter Range	Default
CH	Networking {Addressing}	**Channel.** Set/Read the channel number used for transmitting and receiving data between RF modules (uses 802.15.4 protocol channel numbers).	0x0B - 0x1A (XBee) 0x0C - 0x17 (XBee-PRO)	0x0C (12d)
ID	Networking {Addressing}	**PAN ID.** Set/Read the PAN (Personal Area Network) ID. Use 0xFFFF to broadcast messages to all PANs.	0 - 0xFFFF	0x3332 (13106d)
DH	Networking {Addressing}	**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 {Addressing}	**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
MY	Networking {Addressing}	**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
SH	Networking {Addressing}	**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	Networking {Addressing}	**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
RR (v1.xA0*)	Networking {Addressing}	**XBee Retries.** Set/Read the maximum number of retries the module will execute in addition to the 3 retries provided by the 802.15.4 MAC. For each XBee retry, the 802.15.4 MAC can execute up to 3 retries.	0 - 6	0
RN	Networking {Addressing}	**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 [exponent]	0
MM (v1.x80*)	Networking {Addressing}	**MAC Mode.** Set/Read MAC Mode value. MAC Mode enables/disables the use of a MaxStream header in the 802.15.4 RF packet. When Mode 0 is enabled (MM=0), duplicate packet detection is enabled as well as certain AT commands. Modes 1 and 2 are strict 802.15.4 modes.	0 - 2 0 = MaxStream Mode 1 = 802.15.4 (no ACKs) 2 = 802.15.4 (with ACKs)	0

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 3-02. XBee/XBee-PRO Commands - Networking & Security (Sub-categories designated within {brackets})

AT Command	Command Category	Name and Description	Parameter Range	Default
NI (v1.x80")	Networking {Identification}	**Node Identifier.** Stores a string identifier. The register only accepts printable ASCII data. A string can not start with a space. Carriage return ends command. Command will automatically end when maximum bytes for the string have been entered. This string is returned as part of the ND (Node Discover) command. This identifier is also used with the DN (Destination Node) command.	20-character ASCII string	-
ND (v1.x80")	Networking {Identification}	**Node Discover.** Discovers and reports all RF modules found. The following information is reported for each module discovered (the example cites use of Transparent operation (AT command format) - refer to the long ND command description regarding differences between Transparent and API operation). MY<CR> SH<CR> SL<CR> DB<CR> NI<CR><CR> The amount of time the module allows for responses is determined by the NT parameter. In Transparent operation, command completion is designated by a <CR> (carriage return). ND also accepts a Node Identifier as a parameter. In this case, only a module matching the supplied identifier will respond.	optional 20-character NI value	
NT (v1.xA0")	Networking {Identification}	**Node Discover Time.** Set/Read the amount of time a node will wait for responses from other nodes when using the ND (Node Discover) command.	0x01 - 0xFC	0x19
DN (v1.x80")	Networking {Identification}	**Destination Node.** Resolves an NI (Node Identifier) string to a physical address. The following events occur upon successful command execution: 1. DL and DH are set to the address of the module with the matching Node Identifier. 2. "OK" is returned. 3. RF module automatically exits AT Command Mode If there is no response from a module within 200 msec or a parameter is not specified (left blank), the command is terminated and an "ERROR" message is returned.	20-character ASCII string	-
CE (v1.x80")	Networking {Association}	**Coordinator Enable.** Set/Read the coordinator setting.	0 - 1 0 = End Device 1 = Coordinator	0
SC (v1.x80")	Networking {Association}	**Scan Channels.** Set/Read list of channels to scan for all Active and Energy Scans as a bitfield. This affects scans initiated in command mode (AS, ED) and during End Device Association and Coordinator startup: bit 0 - 0x0B bit 4 - 0x0F bit 8 - 0x13 bit12 - 0x17 bit 1 - 0x0C bit 5 - 0x10 bit 9 - 0x14 bit13 - 0x18 bit 2 - 0x0D bit 6 - 0x11 bit 10 - 0x15 bit14 - 0x19 bit 3 - 0x0E bit 7 - 0x12 bit 11 - 0x16 bit 15 - 0x1A	0 - 0xFFFF [bitfield] (bits 0, 14, 15 not allowed on the XBee-PRO)	0x1FFE (all XBee-PRO Channels)
SD (v1.x80")	Networking {Association}	**Scan Duration.** Set/Read the scan duration exponent. *End Device* - Duration of Active Scan during Association. On beacon system, set SD = BE of coordinator. SD must be set at least to the highest BE parameter of any Beaconing Coordinator with which an End Device or Coordinator wish to discover. *Coordinator* - If 'ReassignPANID' option is set on Coordinator [refer to A2 parameter], SD determines the length of time the Coordinator will scan channels to locate existing PANs. If 'ReassignChannel' option is set, SD determines how long the Coordinator will perform an Energy Scan to determine which channel it will operate on. 'Scan Time' is measured as (# of channels to scan] * (2 ^ SD) * 15.36ms). The number of channels to scan is set by the SC command. The XBee can scan up to 16 channels (SC = 0xFFFF). The XBee PRO can scan up to 13 channels (SC = 0x3FFE). Example: The values below show results for a 13 channel scan: If SD = 0, time = 0.18 sec SD = 8, time = 47.19 sec SD = 2, time = 0.74 sec SD = 10, time = 3.15 min SD = 4, time = 2.95 sec SD = 12, time = 12.58 min SD = 6, time = 11.80 sec SD = 14, time = 50.33 min	0-0x0F [exponent]	4
A1 (v1.x80")	Networking {Association}	**End Device Association.** Set/Read End Device association options. bit 0 - ReassignPanID 0 - Will only associate with Coordinator operating on PAN ID that matches module ID 1 - May associate with Coordinator operating on any PAN ID bit 1 - ReassignChannel 0 - Will only associate with Coordinator operating on matching CH Channel setting 1 - May associate with Coordinator operating on any Channel bit 2 - AutoAssociate 0 - Device will not attempt Association 1 - Device attempts Association until success Note: This bit is used only for Non-Beacon systems. End Devices in Beacon-enabled system must always associate to a Coordinator bit 3 - PollCoordOnPinWake 0 - Pin Wake will not poll the Coordinator for indirect (pending) data 1 - Pin Wake will send Poll Request to Coordinator to extract any pending data bits 4 - 7 are reserved	0 - 0x0F [bitfield]	0

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 3-02. XBee/XBee-PRO Commands - Networking & Security (Sub-categories designated within {brackets})

AT Command	Command Category	Name and Description	Parameter Range	Default
A2 (v1.x80*)	Networking {Association}	**Coordinator Association.** Set/Read Coordinator association options. bit 0 - ReassignPanID 0 - Coordinator will not perform Active Scan to locate available PAN ID. It will operate on ID (PAN ID). 1 - Coordinator will perform Active Scan to determine an available ID (PAN ID). If a PAN ID conflict is found, the ID parameter will change. bit 1 - ReassignChannel - 0 - Coordinator will not perform Energy Scan to determine free channel. It will operate on the channel determined by the CH parameter. 1 - Coordinator will perform Energy Scan to find a free channel, then operate on that channel. bit 2 - AllowAssociation - 0 - Coordinator will not allow any devices to associate to it. 1 - Coordinator will allow devices to associate to it. bits 3 - 7 are reserved	0 - 7 [bitfield]	0
AI (v1.x80*)	Networking {Association}	**Association Indication.** Read errors with the last association request: 0x00 - Successful Completion - Coordinator successfully started or End Device association complete 0x01 - Active Scan Timeout 0x02 - Active Scan found no PANs 0x03 - Active Scan found PAN, but the CoordinatorAllowAssociation bit is not set 0x04 - Active Scan found PAN, but Coordinator and End Device are not configured to support beacons 0x05 - Active Scan found PAN, but the Coordinator ID parameter does not match the ID parameter of the End Device 0x06 - Active Scan found PAN, but the Coordinator CH parameter does not match the CH parameter of the End Device 0x07 - Energy Scan Timeout 0x08 - Coordinator start request failed 0x09 - Coordinator could not start due to invalid parameter 0x0A - Coordinator Realignment is in progress 0x0B - Association Request not sent 0x0C - Association Request timed out - no reply was received 0x0D - Association Request had an Invalid Parameter 0x0E - Association Request Channel Access Failure. Request was not transmitted - CCA failure 0x0F - Remote Coordinator did not send an ACK after Association Request was sent 0x10 - Remote Coordinator did not reply to the Association Request, but an ACK was received after sending the request 0x11 - [reserved] 0x12 - Sync-Loss - Lost synchronization with a Beaconing Coordinator 0x13 - Disassociated - No longer associated to Coordinator	0 - 0x13 {read-only}	-
DA (v1.x80*)	Networking {Association}	**Force Disassociation.** End Device will immediately disassociate from a Coordinator (if associated) and reattempt to associate.	-	-
FP (v1.x80*)	Networking {Association}	**Force Poll.** Request indirect messages being held by a coordinator.	-	-

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 3-02. XBee/XBee-PRO Commands - Networking & Security (Sub-categories designated within {brackets})

AT Command	Command Category	Name and Description	Parameter Range	Default
AS (v1.x80*)	Networking {Association}	Active Scan. Send Beacon Request to Broadcast Address (0xFFFF) and Broadcast PAN (0xFFFF) on every channel. The parameter determines the time the radio will listen for Beacons on each channel. A PanDescriptor is created and returned for every Beacon received from the scan. Each PanDescriptor contains the following information: CoordAddress (SH, SL)<CR> CoordPanID (ID)<CR> CoordAddrMode <CR> 0x02 = 16-bit Short Address 0x03 = 64-bit Long Address Channel (CH parameter) <CR> SecurityUse<CR> ACLEntry<CR> SecurityFailure<CR> SuperFrameSpec<CR> (2 bytes): bit 15 - Association Permitted (MSB) bit 14 - PAN Coordinator bit 13 - Reserved bit 12 - Battery Life Extension bits 8-11 - Final CAP Slot bits 4-7 - Superframe Order bits 0-3 - Beacon Order GtsPermit<CR> RSSI<CR> (RSSI is returned as -dBm) TimeStamp<CR> (3 bytes) <CR> A carriage return <CR> is sent at the end of the AS command. The Active Scan is capable of returning up to 5 PanDescriptors in a scan. The actual scan time on each channel is measured as Time = [(2 ^SD PARAM) * 15.36] ms. Note the total scan time is this time multiplied by the number of channels to be scanned (16 for the XBee and 13 for the XBee-PRO). Also refer to SD command description.	0 - 6	-
ED (v1.x80*)	Networking {Association}	Energy Scan. Send an Energy Detect Scan. This parameter determines the length of scan on each channel. The maximal energy on each channel is returned & each value is followed by a carriage return. An additional carriage return is sent at the end of the command. The values returned represent the detected energy level in units of -dBm. The actual scan time on each channel is measured as Time = [(2 ^ED) * 15.36] ms. Note the total scan time is this time multiplied by the number of channels to be scanned (refer to SD parameter).	0 - 6	-
EE (v1.xA0*)	Networking {Security}	AES Encryption Enable. Disable/Enable 128-bit AES encryption support. Use in conjunction with the KY command.	0 - 1	0 (disabled)
KY (v1.xA0*)	Networking {Security}	AES Encryption Key. Set the 128-bit AES (Advanced Encryption Standard) key for encrypting/decrypting data. The KY register cannot be read.	0 - (any 16-Byte value)	-

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

RF Interfacing

Table 3-03. XBee/XBee-PRO Commands - RF Interfacing

AT Command	Command Category	Name and Description	Parameter Range	Default
PL	RF Interfacing	Power Level. Select/Read the power level at which the RF module transmits conducted power. *NOTE: XBee-PRO RF Modules optimized for use in Japan contain firmware that limits transmit power output to 10 dBm. If PL=4 (default), the maximum power output level is fixed at 10 dBm.*	0 - 4 (XBee / XBee-PRO) 0 = -10 / 10 dBm 1 = -6 / 12 dBm 2 = -4 / 14 dBm 3 = -2 / 16 dBm 4 = 0 / 18 dBm	4
CA (v1.x80*)	RF Interfacing	CCA Threshold. Set/read the CCA (Clear Channel Assessment) threshold. Prior to transmitting a packet, a CCA is performed to detect energy on the channel. If the detected energy is above the CCA Threshold, the module will not transmit the packet.	0 - 0x50 [-dBm]	0x2C (-44d dBm)

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sleep (Low Power)

Table 3-04. XBee/XBee-PRO Commands - Sleep (Low Power)

AT Command	Command Category	Name and Description	Parameter Range	Default
SM	Sleep (Low Power)	**Sleep Mode.** <NonBeacon firmware> Set/Read Sleep Mode configurations.	0 - 5 0 = No Sleep 1 = Pin Hibernate 2 = Pin Doze 3 = Reserved 4 = Cyclic sleep remote 5 = Cyclic sleep remote w/ pin wake-up 6 = [Sleep Coordinator] for backwards compatibility w/ v1.x6 only; otherwise, use CE command.	0
ST	Sleep (Low Power)	**Time before Sleep.** <NonBeacon firmware> Set/Read time period of inactivity (no serial or RF data is sent or received) before activating Sleep Mode. ST parameter is only valid with Cyclic Sleep settings (SM = 4 - 5). Coordinator and End Device ST values must be equal. Also 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.	1 - 0xFFFF [x 1 ms]	0x1388 (5000d)
SP	Sleep (Low Power)	**Cyclic Sleep Period.** <NonBeacon firmware> Set/Read sleep period for cyclic sleeping remotes. Coordinator and End Device SP values should always be equal. To send Direct Messages, set SP = 0. *End Device* - SP determines the sleep period for cyclic sleeping remotes. Maximum sleep period is 268 seconds (0x68B0). *Coordinator* - If non-zero, SP determines the time to hold an indirect message before discarding it. A Coordinator will discard indirect messages after a period of (2.5 * SP).	0 - 0x68B0 [x 10 ms]	0
DP (1.x80*)	Sleep (Low Power)	**Disassociated Cyclic Sleep Period.** <NonBeacon firmware> *End Device* - Set/Read time period of sleep for cyclic sleeping remotes that are configured for Association but are not associated to a Coordinator. (i.e. If a device is configured to associate, configured as a Cyclic Sleep remote, but does not find a Coordinator, it will sleep for DP time before reattempting association.) Maximum sleep period is 268 seconds (0x68B0). DP should be > 0 for NonBeacon systems.	1 - 0x68B0 [x 10 ms]	0x3E8 (1000d)

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

Serial Interfacing

Table 3-05. XBee-PRO Commands - Serial Interfacing

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. Request non-standard baud rates with values above 0x80 using a terminal window. Read the BD register to find actual baud rate achieved.	0 - 7 (standard baud rates) 0 = 1200 bps 1 = 2400 2 = 4800 3 = 9600 4 = 19200 5 = 38400 6 = 57600 7 = 115200 0x80 - 0x1C200 (non-standard baud rates)	3
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
AP (v1.x80*)	Serial Interfacing	**API Enable.** Disable/Enable API Mode.	0 - 2 0 = Disabled 1 = API enabled 2 = API enabled (w/escaped control characters)	0
PR (v1.x80*)	Serial Interfacing	**Pull-up Resistor Enable.** Set/Read bitfield to configure internal pull-up resistor status for I/O lines Bitfield Map: bit 0 - AD4/DIO4 (pin11) bit 1 - AD3 / DIO3 (pin17) bit 2 - AD2/DIO2 (pin18) bit 3 - AD1/DIO1 (pin19) bit 4 - AD0 / DIO0 (pin20) bit 5 - RTS / AD6 / DIO6 (pin16) bit 6 - DTR / SLEEP_RQ / DI8 (pin9) bit 7 - DIN/CONFIG (pin3) Bit set to "1" specifies pull-up enabled; "0" specifies no pull-up	0 - 0xFF	0xFF

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I/O Settings

Table 3-06. XBee-PRO Commands - I/O Settings (sub-category designated within [brackets])

AT Command	Command Category	Name and Description	Parameter Range	Default
D8	I/O Settings	**DI8 Configuration.** Select/Read options for the DI8 line (pin 9) of the RF module.	0 - 1 0 = Disabled 3 = DI (1,2,4 & 5 n/a)	0
D7 (v1.x80*)	I/O Settings	**DIO7 Configuration.** Select/Read settings for the DIO7 line (pin 12) of the RF module. Options include CTS flow control and I/O line settings.	0 - 1 0 = Disabled 1 = CTS Flow Control 2 = (n/a) 3 = DI 4 = DO low 5 = DO high	1
D6 (v1.x80*)	I/O Settings	**DIO6 Configuration.** Select/Read settings for the DIO6 line (pin 16) of the RF module. Options include RTS flow control and I/O line settings.	0 - 1 0 = Disabled 1 = RTS flow control 2 = (n/a) 3 = DI 4 = DO low 5 = DO high	0
D5 (v1.x80*)	I/O Settings	**DIO5 Configuration.** Configure settings for the DIO5 line (pin 15) of the RF module. Options include Associated LED indicator (blinks when associated) and I/O line settings.	0 - 1 0 = Disabled 1 = Associated indicator 2 = ADC 3 = DI 4 = DO low 5 = DO high	1
D0 - D4 (v1.xA0*)	I/O Settings	**(DIO4 -DIO4) Configuration.** Select/Read settings for the following lines: AD0/DIO0 (pin 20), AD1/DIO1 (pin 19), AD2/DIO2 (pin 18), AD3/DIO3 (pin 17), AD4/DIO4 (pin 11). Options include: Analog-to-digital converter, Digital Input and Digital Output.	0 - 1 0 = Disabled 1 = (n/a) 2 = ADC 3 = DI 4 = DO low 5 = DO high	0
IU (v1.xA0*)	I/O Settings	**I/O Output Enable.** Disables/Enables I/O data received to be sent out UART. The data is sent using an API frame regardless of the current AP parameter value.	0 - 1 0 = Disabled 1 = Enabled	1
IT (v1.xA0*)	I/O Settings	**Samples before TX.** Set/Read the number of samples to collect before transmitting data. Maximum number of samples is dependent upon the number of enabled inputs.	1 - 0xFF	1
IS (v1.xA0*)	I/O Settings	**Force Sample.** Force a read of all enabled inputs (DI or ADC). Data is returned through the UART. If no inputs are defined (DI or ADC), this command will return error.	8-bit bitmap (each bit represents the level of an I/O line setup as an output)	-
IO (v1.xA0*)	I/O Settings	**Digital Output Level.** Set digital output level to allow DIO lines that are setup as outputs to be changed through Command Mode.	-	-
IC (v1.xA0*)	I/O Settings	**DIO Change Detect.** Enables/Disables the monitoring of the change detect feature on DIO lines 0-7. If a change is detected, data is transmitted with DIO data only. Any samples queued and waiting for transmission will be sent first.	0 - 0xFF [bitfield]	0 (disabled)
IR (v1.xA0*)	I/O Settings	**Sample Rate.** Set/Read sample rate. When set, this parameter causes the module to sample all enabled inputs at a specified interval.	0 - 0xFFFF [x 1 msec]	0
AV (v1.xA0*)	I/O Settings	**ADC Voltage Reference.** <XBee-PRO only> Set/Read ADC reference voltage switch.	0 - 1 0 = VREF pin 1 = Internal	0
IA (v1.xA0*)	I/O Settings (I/O Line Passing)	**I/O Input Address.** Set/Read addresses of module to which outputs are bound. Setting all bytes to 0xFF will not allow any received I/O packet to change outputs. Setting address to 0xFFFF will allow any received I/O packet to change outputs.	0 - 0xFFFFFFFFFFFFFFFF	0xFFFFFFFF FFFFFFFF
T0 - T7 (v1.xA0*)	I/O Settings (I/O Line Passing)	**(D0 - D7) Output Timeout.** Set/Read Output timeout values for lines that correspond with the D0 - D7 parameters. When output is set (due to I/O line passing) to a non-default level, a timer is started which when expired will set the output to it default level. The timer is reset when a valid I/O packet is received.	0 - 0xFF [x 100 ms]	0xFF
P0	I/O Settings (I/O Line Passing)	**PWM0 Configuration.** Select/Read function for PWM0 pin.	0 - 2 0 = Disabled 1 = RSSI 2 = PWM Output	1
P1 (v1.xA0*)	I/O Settings (I/O Line Passing)	**PWM1 Configuration.** Select/Read function for PWM1 pin.	0 - 2 0 = Disabled 1 = RSSI 2 = PWM Output	0
M0 (v1.xA0*)	I/O Settings (I/O Line Passing)	**PWM0 Output Level.** Set/Read the PWM0 output level.	0 - 0x03FF	-

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 3-06. XBee-PRO Commands - I/O Settings (sub-category designated within [brackets])

AT Command	Command Category	Name and Description	Parameter Range	Default
M1 (v1.xA0*)	I/O Settings (I/O Line Passing)	**PWM1 Output Level.** Set/Read the PWM0 output level.	0 - 0x03FF	-
PT (v1.xA0*)	I/O Settings (I/O Line Passing)	**PWM Output Timeout.** Set/Read output timeout value for both PWM outputs. When PWM is set to a non-zero value: Due to I/O line passing, a time is started which when expired will set the PWM output to zero. The timer is reset when a valid I/O packet is received.]	0 - 0xFF [x 100 ms]	0xFF
RP	I/O Settings (I/O Line Passing)	**RSSI PWM Timer.** Set/Read PWM timer register. Set the duration of PWM (pulse width modulation) signal output on the RSSI pin. The signal duty cycle is updated with each received packet and is shut off when the timer expires.]	0 - 0xFF [x 100 ms]	0x28 (40d)

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

Diagnostics

Table 3-07. XBee/XBee-PRO Commands - Diagnostics

AT Command	Command Category	Name and Description	Parameter Range	Default
VR	Diagnostics	**Firmware Version.** Read firmware version of the RF module.	0 - 0xFFFF [read-only]	Factory-set
VL (v1.x80*)	Diagnostics	**Firmware Version - Verbose.** Read detailed version information (including application build date, MAC, PHY and bootloader versions).	-	-
HV (v1.x80*)	Diagnostics	**Hardware Version.** Read hardware version of the RF module.	0 - 0xFFFF [read-only]	Factory-set
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]	-
EC (v1.x80*)	Diagnostics	**CCA Failures.** Reset/Read count of CCA (Clear Channel Assessment) failures. This parameter value increments when the module does not transmit a packet because it detected energy above the CCA threshold level set with CA command. This count saturates at its maximum value. Set count to "0" to reset count.	0 - 0xFFFF	-
EA (v1.x80*)	Diagnostics	**ACK Failures.** Reset/Read count of acknowledgment failures. This parameter value increments when the module expires its transmission retries without receiving an ACK on a packet transmission. This count saturates at its maximum value. Set the parameter to "0" to reset count.	0 - 0xFFFF	-
ED (v1.x80*)	Diagnostics	**Energy Scan.** Send 'Energy Detect Scan'. ED parameter determines the length of scan on each channel. The maximal energy on each channel is returned and each value is followed by a carriage return. Values returned represent detected energy levels in units of -dBm. Actual scan time on each channel is measured as Time = [(2 ^ SD) * 15.36] ms. Total scan time is this time multiplied by the number of channels to be scanned.	0 - 6	-

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

AT Command Options

Table 3-08. XBee/XBee-PRO Commands - AT Command Options

AT Command	Command Category	Name and Description	Parameter Range	Default
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)
CN	AT Command Mode Options	**Exit Command Mode.** Explicitly exit the module from AT Command Mode.	-	-
AC (v1.xA0*)	AT Command Mode Options	**Apply Changes.** Explicitly apply changes to queued parameter value(s) and re-initialize module.	-	-
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.	2 - 0x0CE4 [x 1 ms]	0x3E8 (1000d)
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 into AT Command Mode.	0 - 0xFF	0x2B ('+' ASCII)

* Firmware version in which the command was first introduced (firmware versions are numbered in hexadecimal notation.)

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4. API Operation

By default, XBee/XBee-PRO RF Modules act as a serial line replacement (Transparent Operation) - all UART data received through the DI pin is queued up for RF transmission. When the module receives an RF packet, the data is sent out the DO pin with no additional information.

Inherent to Transparent Operation are the following behaviors:

- If module parameter registers are to be set or queried, a special operation is required for transitioning the module into Command Mode.
- In point-to-multipoint systems, the application must send extra information so that the receiving module(s) can distinguish between data coming from different remotes.

As an alternative to the default Transparent Operation, API (Application Programming Interface) Operations are available. API operation requires that communication with the module be done through a structured interface (data is communicated in frames in a defined order). The API specifies how commands, command responses and module status messages are sent and received from the module using a UART Data Frame.

3.4.1. API Frame Specifications

Two API modes are supported and both can be enabled using the AP (API Enable) command. Use the following AP parameter values to configure the module to operate in a particular mode:

- AP = 0 (default): Transparent Operation (UART Serial line replacement) API modes are disabled.
- AP = 1: API Operation
- AP = 2: API Operation (with escaped characters)

Any data received prior to the start delimiter is silently discarded. If the frame is not received correctly or if the checksum fails, the data is silently discarded.

API Operation (AP parameter = 1)

When this API mode is enabled (AP = 1), the UART data frame structure is defined as follows:

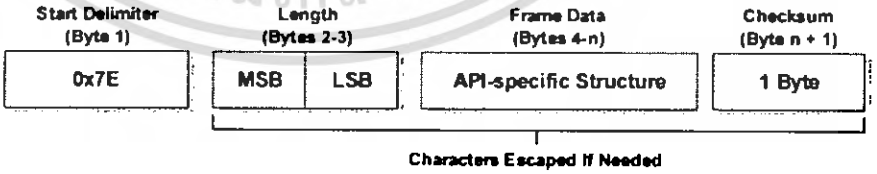
Figure 3-01. UART Data Frame Structure:

MSB = Most Significant Byte, LSB = Least Significant Byte

API Operation - with Escape Characters (AP parameter = 2)

When this API mode is enabled (AP = 2), the UART data frame structure is defined as follows:

Figure 3-02. UART Data Frame Structure - with escape control characters:



MSB = Most Significant Byte, LSB = Least Significant Byte

Escape characters. When sending or receiving a UART data frame, specific data values must be escaped (flagged) so they do not interfere with the UART or UART data frame operation. To escape an interfering data byte, insert 0x7D and follow it with the byte to be escaped XOR'd with 0x20.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data bytes that need to be escaped:

- 0x7E – Frame Delimiter
- 0x7D – Escape
- 0x11 – XON
- 0x13 – XOFF

Example - Raw UART Data Frame (before escaping interfering bytes):
0x7E 0x00 0x02 0x23 0x11 0xCB

0x11 needs to be escaped which results in the following frame:
0x7E 0x00 0x02 0x23 0x7D 0x31 0xCB

Note: In the above example, the length of the raw data (excluding the checksum) is 0x0002 and the checksum of the non-escaped data (excluding frame delimiter and length) is calculated as: 0xFF - (0x23 + 0x11) = (0xFF - 0x34) = 0xCB.

Checksum

To test data integrity, a checksum is calculated and verified on non-escaped data.

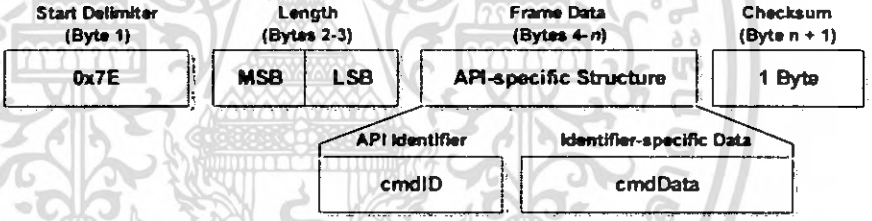
To calculate: Not including frame delimiters and length, add all bytes keeping only the lowest 8 bits of the result and subtract from 0xFF.

To verify: Add all bytes (include checksum, but not the delimiter and length). If the checksum is correct, the sum will equal 0xFF.

3.4.2. API Types

Frame data of the UART data frame forms an API-specific structure as follows:

Figure 3-03. UART Data Frame & API-specific Structure:

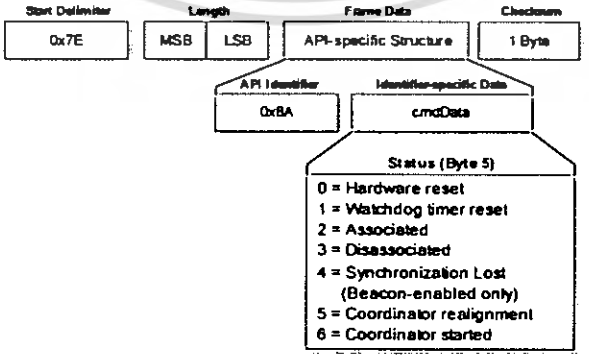


The cmdID frame (API-identifier) indicates which API messages will be contained in the cmdData frame (Identifier-specific data). Refer to the sections that follow for more information regarding the supported API types. Note that multi-byte values are sent big endian.

Modem Status

API Identifier: 0x8A
RF module status messages are sent from the module in response to specific conditions.

Figure 3-04. Modem Status Frames



เอกสารนี้เป็นลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT Command

API Identifier Value: 0x08

The "AT Command" API type allows for module parameters to be queried or set. When using this command ID, new parameter values are applied immediately. This includes any register set with the "AT Command - Queue Parameter Value" (0x09) API type.

Figure 3-05. AT Command Frames

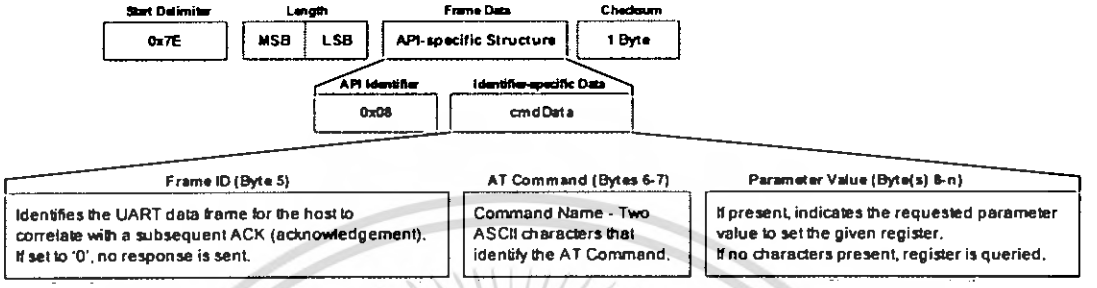
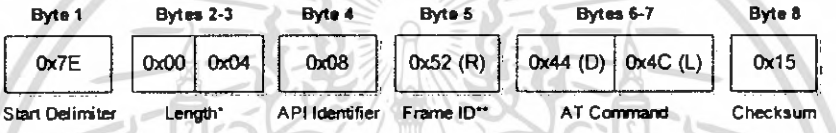


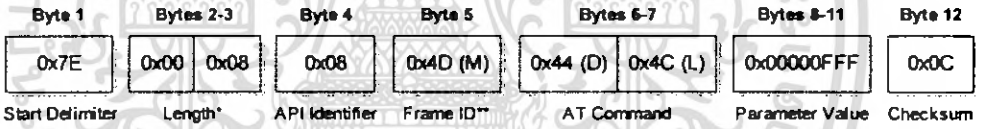
Figure 3-06. Example: API frames when reading the DL parameter value of the module.



** Length [Bytes] = API Identifier + Frame ID + AT Command*

*** "R" value was arbitrarily selected.*

Figure 3-07. Example: API frames when modifying the DL parameter value of the module.



** Length [Bytes] = API Identifier + Frame ID + AT Command + Parameter Value*

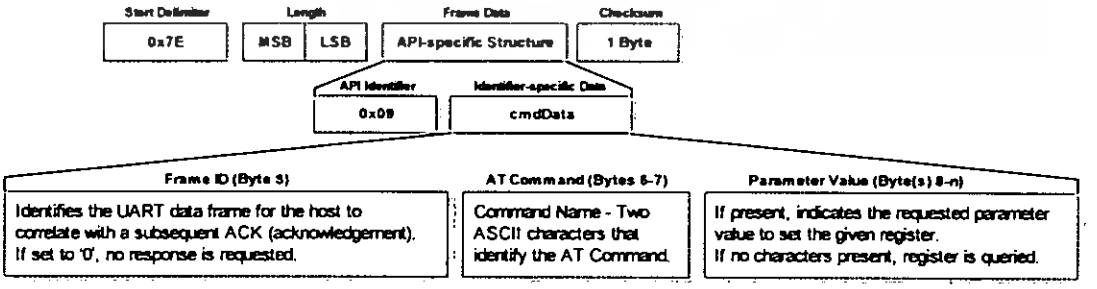
*** "M" value was arbitrarily selected.*

AT Command - Queue Parameter Value

API Identifier Value: 0x09

This API type allows module parameters to be queried or set. In contrast to the "AT Command" API type, new parameter values are queued and not applied until either the "AT Command" (0x08) API type or the AC (Apply Changes) command is issued. Register queries (reading parameter values) are returned immediately.

Figure 3-08. AT Command Frames
(Note that frames are identical to the "AT Command" API type except for the API identifier.)



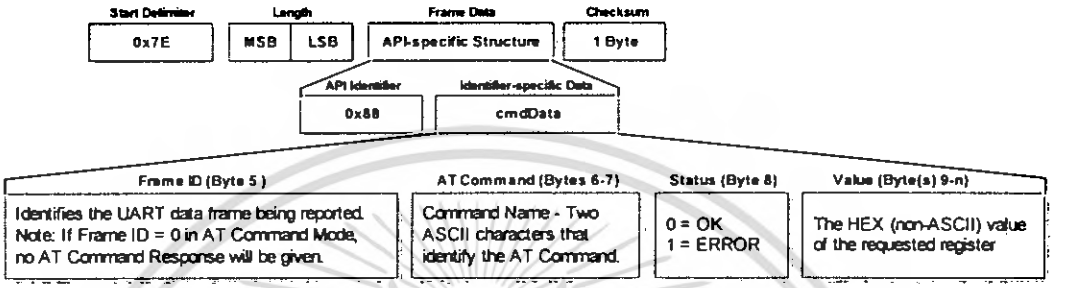
เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT Command Response

API Identifier Value: 0x88
Response to previous command.

In response to an AT Command message, the module will send an AT Command Response message. Some commands will send back multiple frames (for example, the ND (Node Discover) and AS (Active Scan) commands). These commands will end by sending a frame with a status of ATCMD_OK and no cmdData.

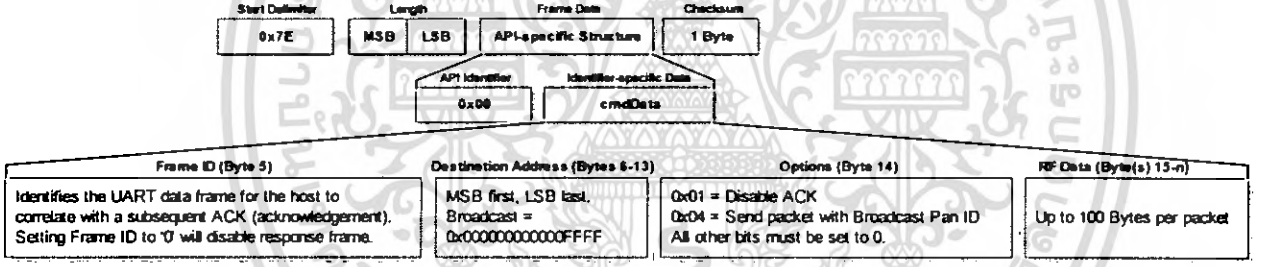
Figure 3-09. AT Command Response Frames.



TX (Transmit) Request: 64-bit address

API Identifier Value: 0x00
A TX Request message will cause the module to send RF Data as an RF Packet.

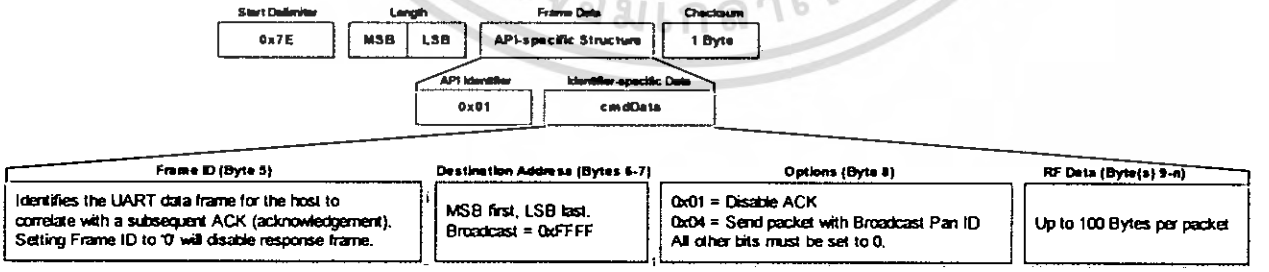
Figure 3-10. TX Packet (64-bit address) Frames



TX (Transmit) Request: 16-bit address

API Identifier Value: 0x01
A TX Request message will cause the module to send RF Data as an RF Packet.

Figure 3-11. TX Packet (16-bit address) Frames



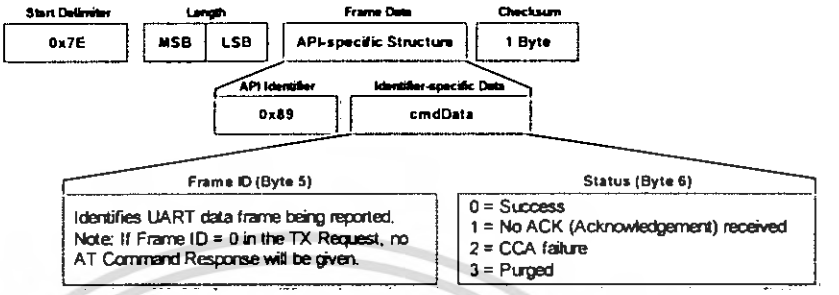
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TX (Transmit) Status

API Identifier Value: 0x89

When a TX Request is completed, the module sends a TX Status message. This message will indicate if the packet was transmitted successfully or if there was a failure.

Figure 3-12. TX Status Frames



NOTES:

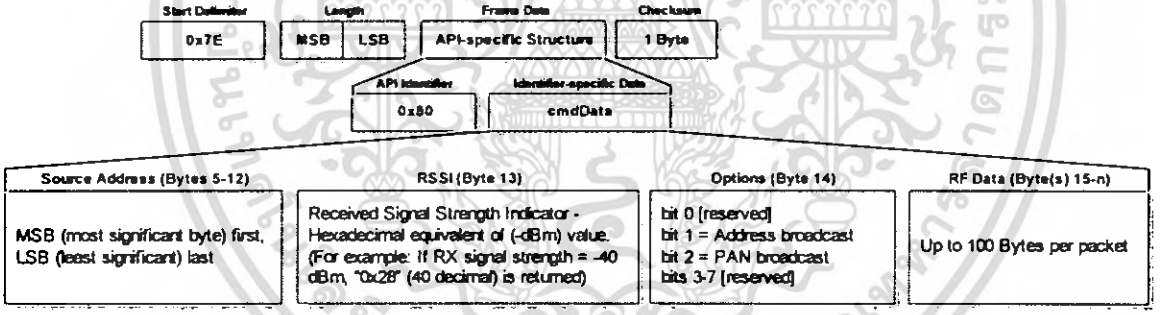
- "STATUS = 1" occurs when all retries are expired and no ACK is received.
- If transmitter broadcasts (destination address = 0x000000000000FFFF), only "STATUS = 0 or 2" will be returned.
- "STATUS = 3" occurs when Coordinator times out of an indirect transmission. Timeout is defined as (2.5 x SP (Cyclic Sleep Period) parameter value).

RX (Receive) Packet: 64-bit Address

API Identifier Value: 0x80

When the module receives an RF packet, it is sent out the UART using this message type.

Figure 3-13. RX Packet (64-bit address) Frames

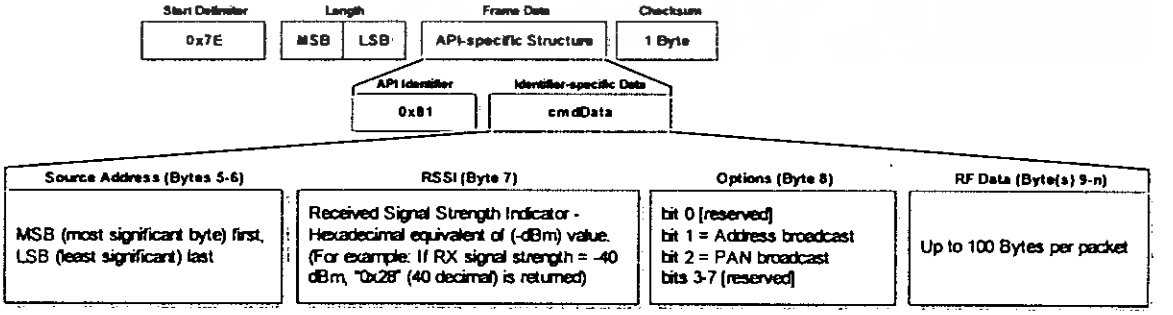


RX (Receive) Packet: 16-bit Address

API Identifier Value: 0x81

When the module receives an RF packet, it is sent out the UART using this message type.

Figure 3-14. RX Packet (16-bit address) Frames



เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 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

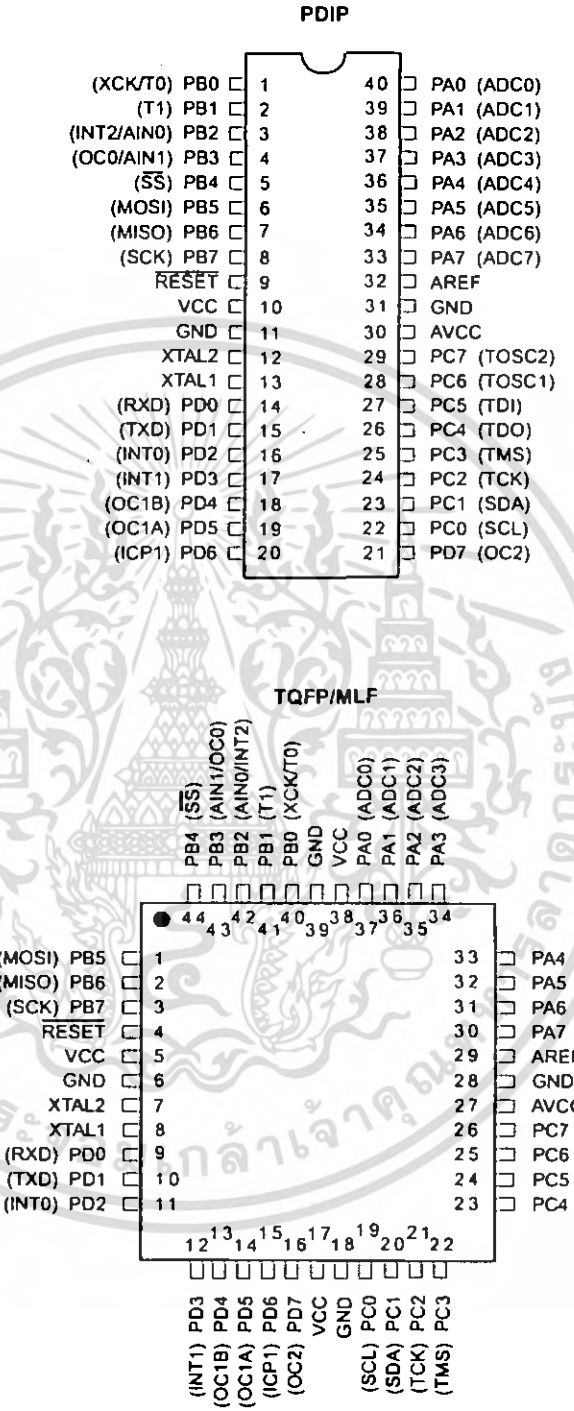
2466HS–AVR–12/03

Note: This is a summary document. A complete document is available on our Web site at www.atmel.com.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Configurations

Figure 1. Pinouts 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.

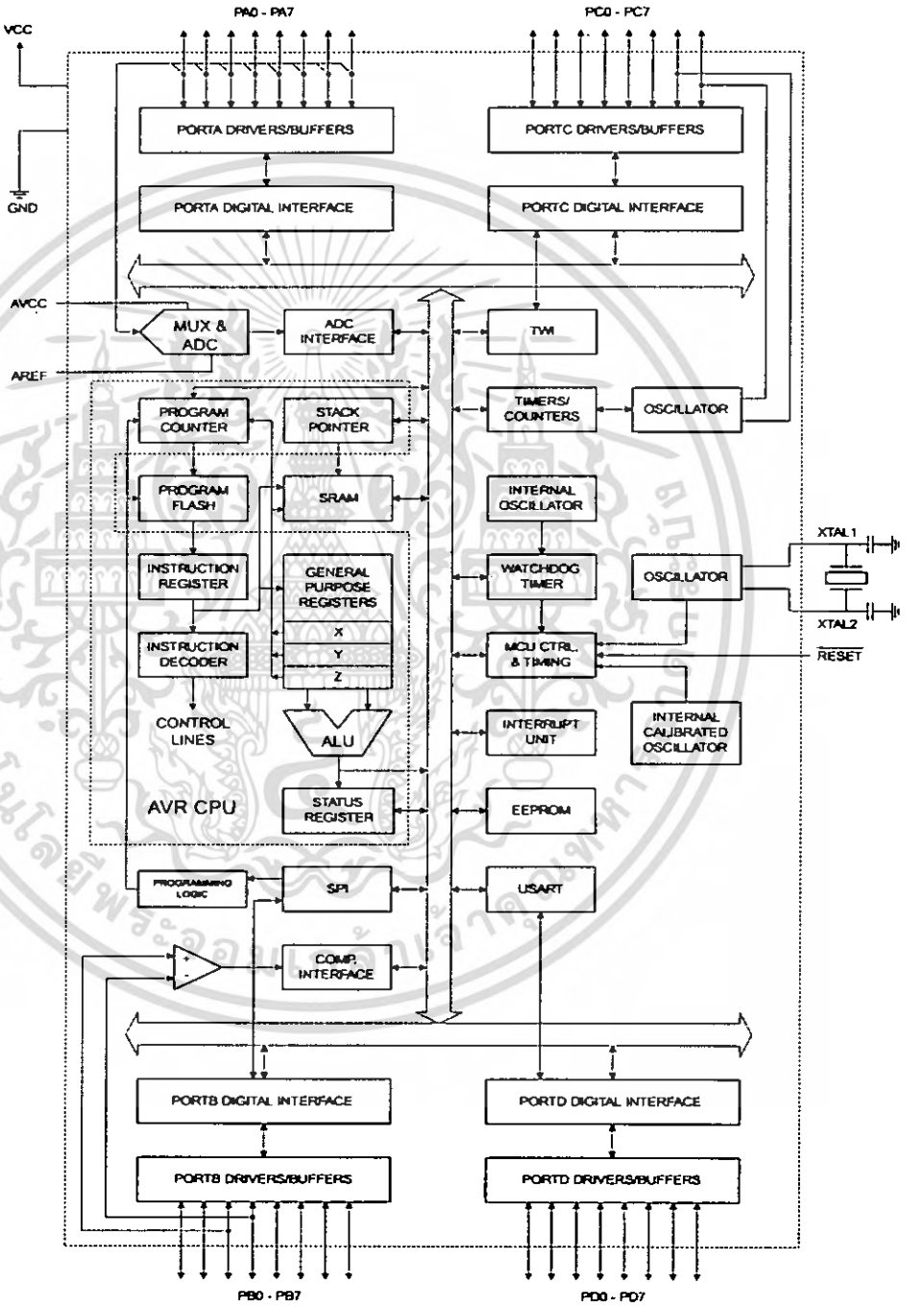
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 56.

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 59.

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 61.

$\overline{\text{RESET}}$

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 36. 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.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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	7
$3E ($5E)	SPH	–	–	–	–	–	SP10	SP9	SP8	10
$3D ($5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	10
$3C ($5C)	OCR0	Timer/Counter0 Output Compare Register								83
$3B ($5B)	GICR	INT1	INT0	INT2	–	–	–	IVSEL	IVCE	46, 67
$3A ($5A)	GIFR	INTF1	INTF0	INTF2	–	–	–	–	–	68
$39 ($59)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	83, 114, 132
$38 ($58)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	84, 115, 132
$37 ($57)	SPMCR	SPMIE	RWWSB	–	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	249
$36 ($56)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	178
$35 ($55)	MCUCR	SM2	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00	30, 66
$34 ($54)	MCUCSR	JTD	ISC2	–	JTRF	WDRF	BORF	EXTRF	PORF	39, 67, 229
$33 ($53)	TCCR0	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	81
$32 ($52)	TCNT0	Timer/Counter0 (8 Bits)								83
$31[1] ($51)[1]	OSCCAL	Oscillator Calibration Register								28
	OCDR	On-Chip Debug Register								225
$30 ($50)	SFIOR	ADTS2	ADTS1	ADTS0	–	ACME	PUD	PSR2	PSR10	55,86,133,199,219
$2F ($4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	109
$2E ($4E)	TCCR1B	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	112
$2D ($4D)	TCNT1H	Timer/Counter1 – Counter Register High Byte								113
$2C ($4C)	TCNT1L	Timer/Counter1 – Counter Register Low Byte								113
$2B ($4B)	OCR1AH	Timer/Counter1 – Output Compare Register A High Byte								113
$2A ($4A)	OCR1AL	Timer/Counter1 – Output Compare Register A Low Byte								113
$29 ($49)	OCR1BH	Timer/Counter1 – Output Compare Register B High Byte								113
$28 ($48)	OCR1BL	Timer/Counter1 – Output Compare Register B Low Byte								113
$27 ($47)	ICR1H	Timer/Counter1 – Input Capture Register High Byte								114
$26 ($46)	ICR1L	Timer/Counter1 – Input Capture Register Low Byte								114
$25 ($45)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	127
$24 ($44)	TCNT2	Timer/Counter2 (8 Bits)								129
$23 ($43)	OCR2	Timer/Counter2 Output Compare Register								129
$22 ($42)	ASSR	–	–	–	–	AS2	TCN2UB	OCR2UB	TCR2UB	130
$21 ($41)	WDTCR	–	–	–	WDTOE	WDE	WDP2	WDP1	WDP0	41
$20[2] ($40)[2]	UBRRH	URSEL	–	–	–	UBRR[11:8]				165
	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	164
$1F ($3F)	EEARH	–	–	–	–	–	–	–	EEAR8	17
$1E ($3E)	EEARL	EEPROM Address Register Low Byte								17
$1D ($3D)	EEDR	EEPROM Data Register								17
$1C ($3C)	EECR	–	–	–	–	EERIE	EEMWE	EEWE	EERE	17
$1B ($3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	64
$1A ($3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	64
$19 ($39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	64
$18 ($38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	64
$17 ($37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	64
$16 ($36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	64
$15 ($35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	65
$14 ($34)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	65
$13 ($33)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	65
$12 ($32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	65
$11 ($31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	65
$10 ($30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	65
$0F ($2F)	SPDR	SPI Data Register								140
$0E ($2E)	SPSR	SPIF	WCOL	–	–	–	–	–	SPI2X	140
$0D ($2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	138
$0C ($2C)	UDR	USART I/O Data Register								161
$0B ($2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	162
$0A ($2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	163
$09 ($29)	UBRRL	USART Baud Rate Register Low Byte								165
$08 ($28)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	200
$07 ($27)	ADMUX	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	215
$06 ($26)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	217
$05 ($25)	ADCH	ADC Data Register High Byte								218
$04 ($24)	ADCL	ADC Data Register Low Byte								218
$03 ($23)	TWDR	Two-wire Serial Interface Data Register								180
$02 ($22)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	180

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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	179
$00 ($20)	TWBR	Two-wire Serial Interface Bit Rate Register								178

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

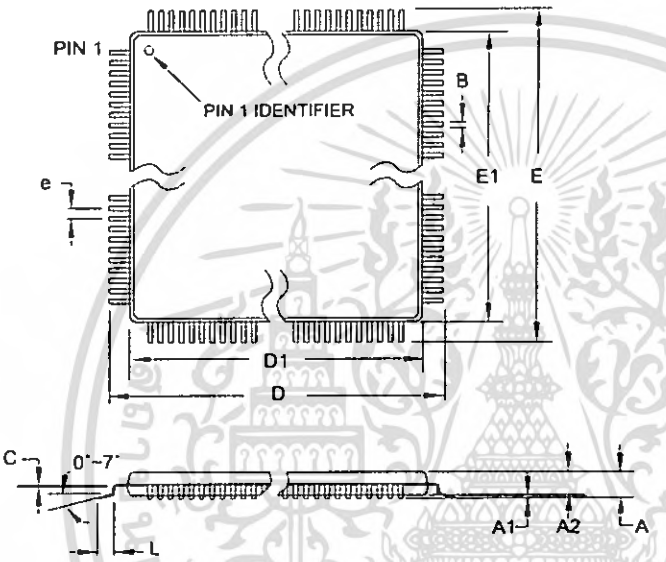
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
8	2.7 - 5.5V	ATmega16L-8AC ATmega16L-8PC ATmega16L-8MC	44A 40P6 44M1	Commercial (0°C to 70°C)
		ATmega16L-8AI ATmega16L-8PI ATmega16L-8MI	44A 40P6 44M1	Industrial (-40°C to 85°C)
16	4.5 - 5.5V	ATmega16-16AC ATmega16-16PC ATmega16-16MC	44A 40P6 44M1	Commercial (0°C to 70°C)
		ATmega16-16AI ATmega16-16PI ATmega16-16MI	44A 40P6 44M1	Industrial (-40°C to 85°C)

Package Type	
44A	44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (TQFP)
40P6	40-pin, 0.600" Wide, Plastic Dual Inline Package (PDIP)
44M1	44-pad, 7 x 7 x 1.0 mm body, lead pitch 0.50 mm, Micro Lead Frame Package (MLF)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

44A



COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	11.75	12.00	12.25	
D1	9.90	10.00	10.10	Note 2
E	11.75	12.00	12.25	
E1	9.90	10.00	10.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

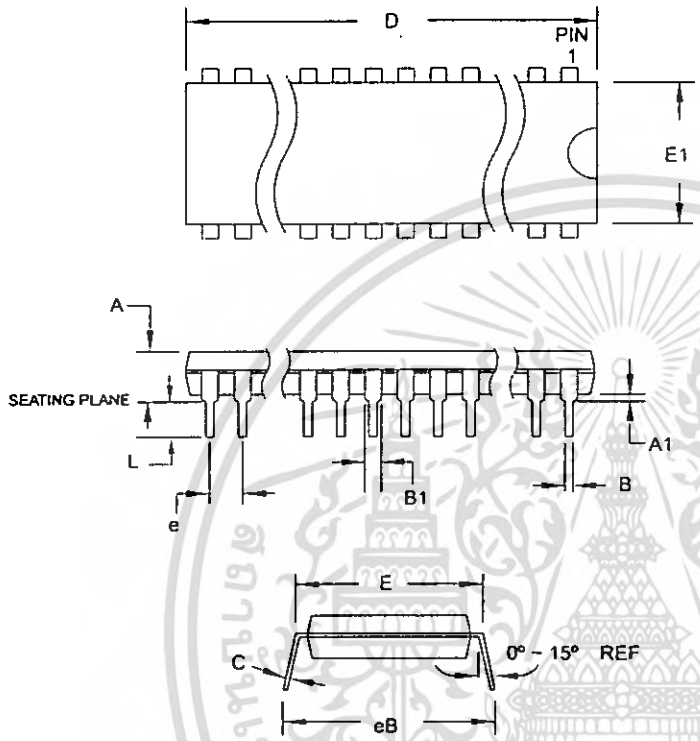
Notes:
1. This package conforms to JEDEC reference MS-026, Variation ACB.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25 mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
3. Lead coplanarity is 0.10 mm maximum.

10/5/2001

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	44A, 44-lead, 10 x 10 mm Body Size, 1.0 mm Body Thickness, 0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)	44A	B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

40P6



COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	4.826	
A1	0.381	–	–	
D	52.070	–	52.578	Note 2
E	15.240	–	15.875	
E1	13.462	–	13.970	Note 2
B	0.356	–	0.559	
B1	1.041	–	1.651	
L	3.048	–	3.556	
C	0.203	–	0.381	
eB	15.494	–	17.526	
e	2.540 TYP			

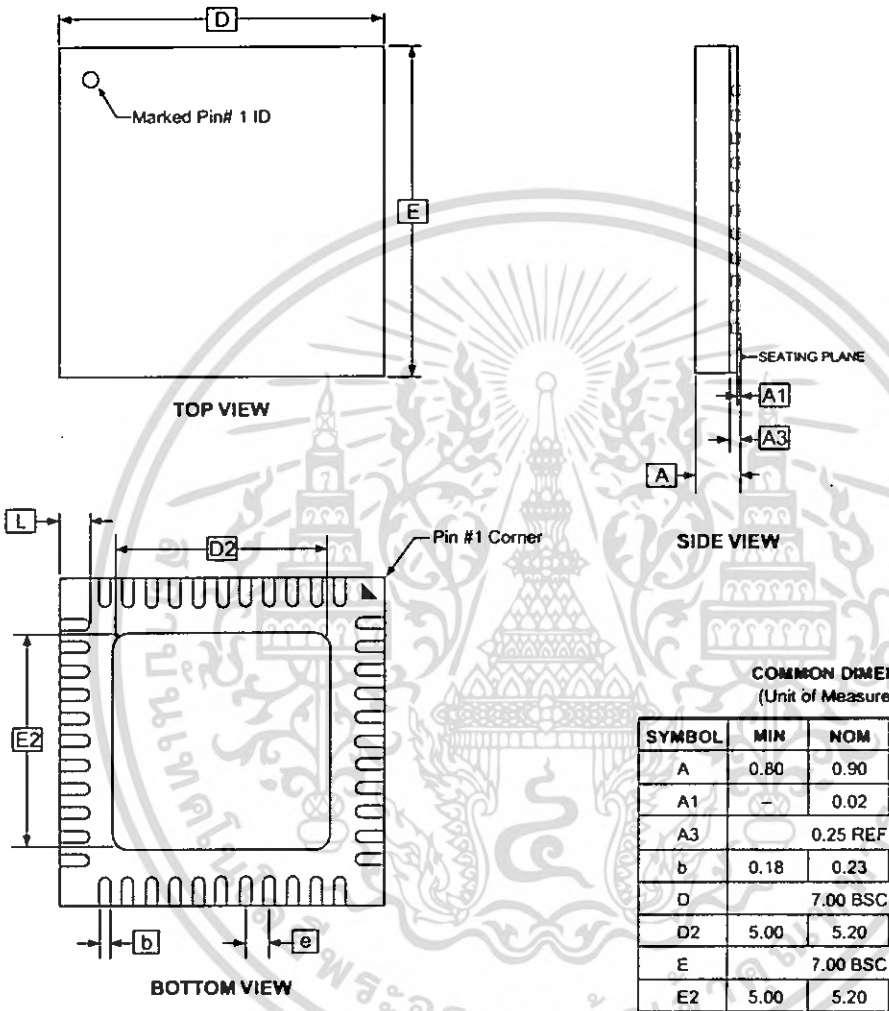
Notes:
1. This package conforms to JEDEC reference MS-011, Variation AC.
2. Dimensions D and E1 do not include mold Flash or Protrusion. Mold Flash or Protrusion shall not exceed 0.25 mm (0.010").

09/28/01

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**40P6**, 40-lead (0.600"/15.24 mm Wide) Plastic Dual Inline Package (PDIP)	40P6	B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

44M1



COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	0.80	0.90	1.00	
A1	–	0.02	0.05	
A3		0.25 REF		
b	0.18	0.23	0.30	
D		7.00 BSC		
D2	5.00	5.20	5.40	
E		7.00 BSC		
E2	5.00	5.20	5.40	
e		0.50 BSC		
L	0.35	0.55	0.75	

Notes: 1. JEDEC Standard MO-220, Fig. 1 (SAW Singulation) VKKD-1.

01/15/03

	TITLE	DRAWING NO.	REV.
ATMEL 2325 Orchard Parkway San Jose, CA 95131	**44M1**, 44-pad, 7 x 7 x 1.0 mm Body, Lead Pitch 0.50 mm Micro Lead Frame Package (MLF)	44M1	C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Errata

The revision letter in this section refers to the revision of the ATmega16 device.

ATmega16(L) Rev. I

- IDCODE masks data from TDI input

1. **IDCODE masks data from TDI input**

   The JTAG instruction IDCODE is not working correctly. Data to succeeding devices are replaced by all-ones during Update-DR.

   Problem Fix / Workaround

   – If ATmega16 is the only device in the scan chain, the problem is not visible.

   – Select the Device ID Register of the ATmega16 by issuing the IDCODE instruction or by entering the Test-Logic-Reset state of the TAP controller to read out the contents of its Device ID Register and possibly data from succeeding devices of the scan chain. Issue the BYPASS instruction to the ATmega16 while reading the Device ID Registers of preceding devices of the boundary scan chain.

   – If the Device IDs of all devices in the boundary scan chain must be captured simultaneously, the ATmega16 must be the fist device in the chain.

ATmega16(L) Rev. H

- IDCODE masks data from TDI input

1. **IDCODE masks data from TDI input**

   The JTAG instruction IDCODE is not working correctly. Data to succeeding devices are replaced by all-ones during Update-DR.

   Problem Fix / Workaround

   – If ATmega16 is the only device in the scan chain, the problem is not visible.

   – Select the Device ID Register of the ATmega16 by issuing the IDCODE instruction or by entering the Test-Logic-Reset state of the TAP controller to read out the contents of its Device ID Register and possibly data from succeeding devices of the scan chain. Issue the BYPASS instruction to the ATmega16 while reading the Device ID Registers of preceding devices of the boundary scan chain.

   – If the Device IDs of all devices in the boundary scan chain must be captured simultaneously, the ATmega16 must be the fist device in the chain.

ATmega16(L) Rev. G

- IDCODE masks data from TDI input

1. **IDCODE masks data from TDI input**

   The JTAG instruction IDCODE is not working correctly. Data to succeeding devices are replaced by all-ones during Update-DR.

   Problem Fix / Workaround

   – If ATmega16 is the only device in the scan chain, the problem is not visible.

   – Select the Device ID Register of the ATmega16 by issuing the IDCODE instruction or by entering the Test-Logic-Reset state of the TAP controller to read out the contents of its Device ID Register and possibly data from succeeding devices of the scan chain. Issue the BYPASS instruction to the ATmega16 while reading the Device ID Registers of preceding devices of the boundary scan chain.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

– If the Device IDs of all devices in the boundary scan chain must be captured simultaneously, the ATmega16 must be the fist device in the chain.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Datasheet Change Log for ATmega16

This section contains a log on the changes made to the datasheet for ATmega16.

Changes from Rev. 2466G-10/03 to Rev. 2466H-12/03

All page numbers refer to this document.

1. **Updated "Calibrated Internal RC Oscillator" on page 27.**

Changes from Rev. 2466F-02/03 to Rev. 2466G-10/03

All page numbers refer to this document.

1. **Removed "Preliminary" from the datasheet.**
2. **Changed ICP to ICP1 in the datasheet.**
3. **Updated "JTAG Interface and On-chip Debug System" on page 34.**
4. **Updated assembly and C code examples in "Watchdog Timer Control Register – WDTCR" on page 41.**
5. **Updated Figure 46 on page 101.**
6. **Updated Table 15 on page 36, Table 82 on page 215 and Table 115 on page 274.**
7. **Updated "Test Access Port – TAP" on page 220 regarding JTAGEN.**
8. **Updated description for the JTD bit on page 229.**
9. **Added note 2 to Figure 126 on page 251.**
10. **Added a note regarding JTAGEN fuse to Table 105 on page 259.**
11. **Updated Absolute Maximum Ratings* and DC Characteristics in "Electrical Characteristics" on page 289.**
12. **Updated "ATmega16 Typical Characteristics" on page 297.**
13. **Fixed typo for 16 MHz MLF package in "Ordering Information" on page 11.**
14. **Added a proposal for solving problems regarding the JTAG instruction IDCODE in "Errata" on page 15.**

Changes from Rev. 2466E-10/02 to Rev. 2466F-02/03

All page numbers refer to this document.

1. **Added note about masking out unused bits when reading the Program Counter in "Stack Pointer" on page 10.**
2. **Added Chip Erase as a first step in "Programming the Flash" on page 286 and "Programming the EEPROM" on page 287.**
3. **Added the section "Unconnected pins" on page 53.**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Added tips on how to disable the OCD system in "On-chip Debug System" on page 34.

5. Removed reference to the "Multi-purpose Oscillator" application note and "32 kHz Crystal Oscillator" application note, which do not exist.

6. Added information about PWM symmetry for Timer0 and Timer2.

7. Added note in "Filling the Temporary Buffer (Page Loading)" on page 252 about writing to the EEPROM during an SPM Page Load.

8. Removed ADHSM completely.

9. Added Table 73, "TWI Bit Rate Prescaler," on page 180 to describe the TWPS bits in the "TWI Status Register – TWSR" on page 179.

10. Added section "Default Clock Source" on page 23.

11. Added note about frequency variation when using an external clock. Note added in "External Clock" on page 29. An extra row and a note added in Table 118 on page 291.

12. Various minor TWI corrections.

13. Added "Power Consumption" data in "Features" on page 1.

14. Added section "EEPROM Write During Power-down Sleep Mode" on page 20.

15. Added note about Differential Mode with Auto Triggering in "Prescaling and Conversion Timing" on page 205.

16. Added updated "Packaging Information" on page 12.

Changes from Rev. 2466D-09/02 to Rev. 2466E-10/02

All page numbers refer to this document.

1. Updated "DC Characteristics" on page 289.

Changes from Rev. 2466C-03/02 to Rev. 2466D-09/02

All page numbers refer to this document.

1. Changed all Flash write/erase cycles from 1,000 to 10,000.

2. Updated the following tables: Table 4 on page 24, Table 15 on page 36, Table 42 on page 83, Table 45 on page 110, Table 46 on page 110, Table 59 on page 141, Table 67 on page 165, Table 90 on page 233, Table 102 on page 257, "DC Characteristics" on page 289, Table 119 on page 291, Table 121 on page 293, and Table 122 on page 295.

3. Updated "Errata" on page 15.

Changes from Rev. 2466B-09/01 to Rev. 2466C-03/02

All page numbers refer to this document.

1. Updated typical EEPROM programming time, Table 1 on page 18.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. **Updated typical start-up time in the following tables:**
   Table 3 on page 23, Table 5 on page 25, Table 6 on page 26, Table 8 on page 27, Table 9 on page 27, and Table 10 on page 28.

3. **Updated Table 17 on page 41 with typical WDT Time-out.**

4. **Added Some Preliminary Test Limits and Characterization Data.**
   Removed some of the TBD's in the following tables and pages:
   Table 15 on page 36, Table 16 on page 40, Table 116 on page 272 (table removed in document review #D), "Electrical Characteristics" on page 289, Table 119 on page 291, Table 121 on page 293, and Table 122 on page 295.

5. **Updated TWI Chapter.**
   Added the note at the end of the "Bit Rate Generator Unit" on page 176.

6. **Corrected description of ADSC bit in "ADC Control and Status Register A – ADCSRA" on page 217.**

7. **Improved description on how to do a polarity check of the ADC doff results in "ADC Conversion Result" on page 214.**

8. **Added JTAG version number for rev. H in Table 87 on page 227.**

9. **Added not regarding OCDEN Fuse below Table 105 on page 259.**

10. **Updated Programming Figures:**
    Figure 127 on page 261 and Figure 136 on page 272 are updated to also reflect that AVCC must be connected during Programming mode. Figure 131 on page 268 added to illustrate how to program the fuses.

11. **Added a note regarding usage of the "PROG_PAGELOAD ($6)" on page 278 and "PROG_PAGEREAD ($7)" on page 278.**

12. **Removed alternative algortihm for leaving JTAG Programming mode.**
    See "Leaving Programming Mode" on page 286.

13. **Added Calibrated RC Oscillator characterization curves in section "ATmega16 Typical Characteristics" on page 297.**

14. **Corrected ordering code for MLF package (16MHz) in "Ordering Information" on page 11.**

15. **Corrected Table 90, "Scan Signals for the Oscillators(1)(2)(3)," on page 233.**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

Disclaimer: Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

© Atmel Corporation 2003. All rights reserved. Atmel® and combinations thereof, AVR®, and AVR Studio® are the registered trademarks of Atmel Corporation or its subsidiaries. Microsoft®, Windows®, Windows NT®, and Windows XP® are the registered trademarks of Microsoft Corporation. Other terms and product names may be the trademarks of others

Printed on recycled paper.

2466HS–AVR–12/03

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้