สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

อุปกรณ์แจ้งเตือนการออกนอกพื้นที่บริเวณบ้านของเด็กและคนชรา

Out of Home Caution Device of Children and Elder

T104265

โดย

นายธงชัย เด็กใจตรง

นายธนพงษ์ คุณากรพล

นางสาวนิศานาถ ยลมานะ

เลขหมู่.....
เลขทะเบียน 104265
วัน,เดือน,ปี 30 ต.ค. 2552

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUT OF HOME CAUTION DEVICE OF CHILDREN AND ELDER

BY

MR. THONGCHAI TEKCHAITRONG

MR. THANAPHONG KUNAKORNPOLN

MS. NISANART YOLMANA

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2008

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	อุปกรณ์แจ้งเตือนการออกนอกพื้นที่บริเวณบ้านของเด็กและคนชรา			
	Out of home Caution Device of children and elder			
นักศึกษา	นายธงชัย	เต็กใจตรง	รหัสนักศึกษา	48010333
	นายธนพงษ์	คุณากรพล	รหัสนักศึกษา	48010342
	นางสาวนิศานาถ	ยลมานะ	รหัสนักศึกษา	48010465
อาจารย์ที่ปรึกษา	รศ.อรลาภ แสงอรุณ			
	อ. สรพงษ์ วชิรรัตนพรกุล			
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต			
	สาขาวิศวกรรมสารสนเทศ			
ภาควิชา	วิศวกรรมสารสนเทศ			
ปีการศึกษา	2551			

ปริญญานิพนธ์ฉบับนี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว

.....

(อาจารย์สรพงษ์ วชิรรัตนพรกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน	อุปกรณ์แจ้งเตือนการออกนอกพื้นที่บริเวณบ้านของเด็กและคนชรา			
	Out of Home Caution Device of Children and Elder			
นักศึกษา	นายธงชัย	เต็กใจตรง	รหัสนักศึกษา	48010333
	นายธนพงษ์	คุณากรพล	รหัสนักศึกษา	48010342
	นางสาวนิศานาถ	ยลมานะ	รหัสนักศึกษา	48010465
อาจารย์ที่ปรึกษา	รศ. อรลาภ	แสงอรุณ		
	อาจารย์สรพงษ์	วชิรรัตนพรกุล		
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต			
	สาขาวิศวกรรมสารสนเทศ			
ภาควิชา	วิศวกรรมสารสนเทศ			
ปีการศึกษา	2551			

บทคัดย่อ

ปริญญานิพนธ์นี้ทำการออกแบบและสร้างอุปกรณ์แจ้งเตือนการออกนอกพื้นที่บริเวณบ้านของเด็กและคนชรา โดยนำเทคโนโลยีการสื่อสารไร้สาย และเทคโนโลยีสมองกลฝังตัวบนระบบปฏิบัติการลินุกซ์(Embeded Linux System) มาประยุกต์การทำงานเข้าไว้ด้วยกัน เพื่อสะดวกในการควบคุมดูแลเด็กและคนชราให้อยู่ภายในพื้นที่บ้านหรือพื้นที่ที่จำกัดไว้ โดยผู้ดูแลสามารถรับสัญญาณแจ้งเตือนได้จากการทำงานของหลอดไฟ LED และลำโพงบนอุปกรณ์ภาครับสัญญาณที่ติดตั้งภายในบริเวณบ้าน นอกจากนี้ผู้ใช้งานยังสามารถตรวจสอบสถานะการออกนอกพื้นที่และการแจ้งเหตุฉุกเฉินจากเด็กและคนชราในรูปแบบเวลาจริง(Real-time) หรือเรียกดูข้อมูลสถานะย้อนหลังผ่านทางหน้าเว็บแอพพลิเคชันได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	Out of Home Caution Device of Children and Elder		
Student	Mr. Thongchai	Tekchaitrong	ID.48010333
	Mr. Thanaphong	Kunakornpoln	ID.48010342
	Miss Nisanart	Yolmana	ID.48010465
Advisor	Assoc.Prof.Ornlarp	Sangaroon	
	Mr. Sorapong	Wachirarattanapornkul	
Graduate Level	Bachelor Degree of Information Engineering		
Department	Information Engineering		
Academic Year	2551		

ABSTRACT

This project designed and built the caution device for the leaving home area of children and elder people by utilizing the Wireless Technology and the Embedded Linux System. For the purpose of keeping children and elder in the housing area or the designated location, the caretaker will receive the alarm signal through the LED light and speaker on the reception installed at home. Moreover, the user will be able to monitor the status in case of children and elder leaving the area and receive emergency alarm signal in Real time, or track back the data via web application.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จเรียบร้อยได้ด้วยดี ก็ด้วยได้รับคำแนะนำและชี้แนวทางในการศึกษาค้นคว้าข้อมูล รายละเอียด และช่วยแก้ไขในส่วนที่บกพร่องต่างๆ จากอาจารย์รศ. อรลาภ แสงอรุณ และอาจารย์สรพงษ์ วชิรรัตนพรกุล คณะผู้จัดทำขอขอบพระคุณท่านอาจารย์ทั้งสองเป็นอย่างสูง

ขอขอบคุณพี่แบงค์(คุณพงศ์ภัทร), พี่จ้อน(คุณณัฐนันท์), พี่แป๊ะ(คุณสมชาย) และน้องกฤษณ์ที่สละเวลาให้คำปรึกษาเรื่องปริญญานิพนธ์ และช่วยแก้ไขปัญหาต่างๆเป็นอย่างดีเสมอมา

กราบขอบพระคุณคุณพ่อ คุณแม่ ที่คอยดูแลห่วงใย มอบกำลังใจและให้การสนับสนุนในทุกๆเรื่อง และยังช่วยอำนวยความสะดวกในทุกๆการเดินทาง เพื่อความปลอดภัยของลูก ไม่ว่าจะเป็นเวลาใด รวมทั้งขอขอบคุณพี่ๆและน้องๆ ทุกคนที่เป็นกำลังใจและให้การช่วยเหลือในด้านต่างๆเสมอมาโดยตลอด

สุดท้ายขอขอบคุณเพื่อนๆห้องโปรเจค และเพื่อนๆภาควิศวกรรมสารสนเทศรุ่น 8 ที่คอยช่วยเหลือ ไม่ทอดทิ้งกัน คณะผู้จัดทำขอขอบคุณจากใจจริง

นายธงชัย เด็กใจตรง
นายธนพงษ์ คุณากรพล
นางสาวนิศานาถ ยลมานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
Abstract	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูปภาพ	ช
สารบัญตาราง	ฌ
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	1
1.3 ขอบเขตของปริญญานิพนธ์	2
1.4 ประโยชน์ที่คาดว่าจะได้รับจากปริญญานิพนธ์	2
1.5 ขั้นตอนการดำเนินงาน	3
1.6 อุปกรณ์ที่ต้องใช้	
1.6.1 ฮาร์ดแวร์(Hardware)	4
1.6.2 ซอฟต์แวร์(Software)	4
1.7 เนื้อหาภายในปริญญานิพนธ์	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	
2.1 ทฤษฎีการสื่อสารไร้สาย	5
2.1.1 องค์ประกอบของระบบสื่อสารไร้สาย	5
2.1.2 ตัวกลางของระบบการสื่อสาร	6
2.1.3 การรับส่งข้อมูลของการสื่อสารไร้สาย	7
2.2 ชุดโมดูลสื่อสารไร้สาย DCBT-24AX	8
2.2.1 คุณสมบัติของ DCBT-24AX	8
2.2.2 รูปร่างและขนาดของ DCBT-24AX	9
2.2.3 ฟังก์ชันการทำงานแต่ละขาของ DCBT-24AX	10
2.2.4 ไมโครคอนโทรลเลอร์ตระกูล AVR รุ่น ATmega168V	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.2.5 ชิพรับส่งสัญญาณไร้สาย nRF24L01	14
2.2.6 คำสั่งการตั้งค่าการใช้งานโมดูล DCBT-24AX	17
2.3 iBoard	20
2.3.1 คุณสมบัติทางด้านฮาร์ดแวร์	20
2.3.2 คุณสมบัติทางด้านซอฟต์แวร์	20
2.4 เสียง	22
2.4.1 ลำโพง	23
2.5 ภาษา PHP	24
2.5.1 ประเภทข้อมูล	25
2.5.2 การแปลงประเภทข้อมูล	26
2.5.3 การทำงานไฟล์ และไดเรคทอรี	26
2.6 ระบบจัดการฐานข้อมูล SQLite	27
2.6.1 ข้อดีของ SQLite	27
2.6.2 ภาษาโปรแกรม ที่สามารถใช้ SQLite	28
2.7 เทคนิค AJAX ในเว็บแอพพลิเคชัน	28
2.7.1 XML	31
2.7.2 Javascript	31
บทที่ 3 การออกแบบ	
3.1 หลักการทำงานของระบบโดยรวม	33
3.2 ส่วนการทำงานของภาคส่งสัญญาณ	34
3.2.1 วงจรภายในภาคส่งสัญญาณ ชุดโมดูล DCBT-24AX	34
3.2.2 การทำงานของ DCBT-24AX ในการส่งข้อมูล	35
3.3 ส่วนการทำงานของภาครับสัญญาณ	37
3.3.1 วงจรภายในภาครับสัญญาณ ชุดโมดูล DCBT-24AX	37
3.3.2 การทำงานของ DCBT-24AX ในการรับข้อมูล	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
3.4 ส่วนการทำงานของ iBoard	40
3.4.1 วงจรการเชื่อมต่อระหว่างอุปกรณ์ภาคส่งสัญญาณและ iBoard ผ่านทางสายพอร์ตอนุกรม	40
3.4.2 หลักการตรวจสอบข้อมูลที่รับเข้ามา เพื่อเก็บในฐานข้อมูล	40
3.4.3 การแสดงผลผ่านทางเว็บแอพพลิเคชัน	41
บทที่ 4 ผลการทดลอง	
4.1 ผลการทดลองการรับ-ส่งข้อมูลของอุปกรณ์ภาครับและภาคส่งสัญญาณไร้สาย	48
4.2 ผลการทดลองการรับข้อมูลของ iBoard โดยการส่งข้อมูลจากอุปกรณ์ภาครับผ่านทางสายพอร์ตอนุกรม	52
4.3 ผลการทดลองการทำงานของทั้งระบบ	56
บทที่ 5 สรุปผลการดำเนินงาน	
5.1 สรุปผล	58
5.2 ปัญหาและข้อจำกัดของการดำเนินงาน	58
5.3 แนวทางในการพัฒนาต่อ	59
บรรณานุกรม	60
ภาคผนวก ก.	61
ภาคผนวก ข.	80
ภาคผนวก ค.	82
ภาคผนวก ง.	87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

		หน้า
รูปที่ 2.1	ส่วนประกอบของระบบสื่อสาร	6
รูปที่ 2.2	โมดูล DCBT-24AX	9
รูปที่ 2.3	ส่วนประกอบของโครงสร้าง DCBT-24AX	10
รูปที่ 2.4	ฟังก์ชั่นขาการทำงานของตัวโมดูล DCBT-24AX	10
รูปที่ 2.5	โครงสร้างของไมโครคอนโทรลเลอร์ ATmega168V แบบ 32 ขา	12
รูปที่ 2.6	บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ ATmega168V	14
รูปที่ 2.7	ลักษณะขาของชิพไอซี nRF24L01	15
รูปที่ 2.8	การเชื่อมต่อการทำงานของขาภายในโมดูล DCBT 24-AX ระหว่างไมโครคอนโทรลเลอร์ ATmega168 กับ ชิพรับส่งข้อมูล nRF24L01	16
รูปที่ 2.9	ขาของ iBoard ทั้ง 40 ขา	21
รูปที่ 2.10	บล็อกไดอะแกรมของโครงสร้าง iBoard	21
รูปที่ 2.11	ก.) การกระจายของโมเลกุลอากาศ	22
รูปที่ 2.11	ข.) กราฟระหว่างความดันอากาศ – ระยะทาง	22
รูปที่ 2.11	ค.) กราฟระหว่างการกระจัดของอากาศ – ระยะทาง	22
รูปที่ 2.12	Buzzer แบบ diaphragm -Piezo	24
รูปที่ 2.13	โครงสร้างการทำงานของ PHP	25
รูปที่ 2.14	เปรียบเทียบการทำงานระหว่างเว็บแอพพลิเคชันแบบเดิมกับแบบที่ใช้AJAX	29
รูปที่ 2.15	เปรียบเทียบการติดต่อสื่อสาร ระหว่างเว็บแอพพลิเคชันแบบดังเดิมกับแบบที่ใช้ AJAX	30
รูปที่ 3.1	หลักการทำงานของระบบโดยรวม	33
รูปที่ 3.2	วงจรภายในโมดูล DCBT-24AX ของภาคส่งสัญญาณ	34
รูปที่ 3.3	Flow Chart การทำงานของภาคส่งสัญญาณ	36
รูปที่ 3.4	วงจรภายในโมดูล DCBT-24AX ของภาครับสัญญาณ	37
รูปที่ 3.5	Flow Chart การทำงานของภาครับสัญญาณ	39
รูปที่ 3.6	วงจรการเชื่อมต่อระหว่างอุปกรณ์ภาคส่งสัญญาณและ iBoard	40
รูปที่ 3.7	Flow Chart การเก็บข้อมูลลงฐานข้อมูลใน iBoard	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

		หน้า
รูปที่ 3.8	หน้าต่างโฮมเพจ	42
รูปที่ 3.9	Diagram การ Login เข้าสู่การใช้งานเรียกดูข้อมูล	43
รูปที่ 3.10	หน้าต่างเว็บเพจ Login	43
รูปที่ 3.11	หน้าต่างเว็บเพจหน้าเมนูหลัก	44
รูปที่ 3.12	หน้าต่างเว็บเพจหน้า Status	45
รูปที่ 3.13	หลักการทำงานของเซิร์ฟเวอร์ในการเรียกใช้ข้อมูลในรูปแบบเวลาจริง	45
รูปที่ 3.14	หลักการทำงานของเซิร์ฟเวอร์ในการเรียกใช้ข้อมูลย้อนหลัง	46
รูปที่ 3.15	หน้าต่างเว็บเพจหน้า Statistics	47
รูปที่ 3.16	NIAM-MODEL ของฐานข้อมูล	47
รูปที่ 4.1	ชุดอุปกรณ์ภาคส่งด้านหน้าและภาครับสัญญาณไร้สาย	49
รูปที่ 4.2	ชุดอุปกรณ์ภาคส่งด้านหลังและภาครับสัญญาณไร้สาย	49
รูปที่ 4.3	การทำงานของอุปกรณ์ภาคส่งสัญญาณ เมื่อมีการส่งข้อมูล	50
รูปที่ 4.4	การแจ้งเตือนของอุปกรณ์ภาครับสัญญาณ	50
รูปที่ 4.5	ชุดอุปกรณ์รับสัญญาณและ iBoard	53
รูปที่ 4.6	หน้าจอแสดงผลการรับข้อมูลของ iBoard เมื่อมีการกดสวิตช์ส่งสัญญาณฉุกเฉิน	54
	ก.) อุปกรณ์ส่งสัญญาณตัวที่ 1 มีการกดสวิตช์ส่งสัญญาณฉุกเฉิน	
	ข.) อุปกรณ์ส่งสัญญาณตัวที่ 2 มีการกดสวิตช์ส่งสัญญาณฉุกเฉิน	
	ค.) อุปกรณ์ส่งสัญญาณตัวที่ 1 และ 2 มีการกดสวิตช์ส่งสัญญาณฉุกเฉินพร้อมกัน	
รูปที่ 4.7	หน้าจอแสดงผลการรับข้อมูลของ iBoard เมื่อภาคส่งสัญญาณอยู่นอกระยะส่ง	55
	ก.) อุปกรณ์ส่งสัญญาณตัวที่ 1 ออกนอกระยะส่งสัญญาณ	
	ข.) อุปกรณ์ส่งสัญญาณตัวที่ 2 ออกนอกระยะส่งสัญญาณ	
	ค.) อุปกรณ์ส่งสัญญาณตัวที่ 1 และ 2 ออกนอกระยะส่งสัญญาณพร้อมกัน	
รูปที่ 4.8	หน้าต่าง Login	56
รูปที่ 4.9	หน้าต่าง Status	56
รูปที่ 4.10	หน้าต่าง Statistics	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 1.1 ขั้นตอนการดำเนินงาน	3
ตารางที่ 2.1 ฟังก์ชั่นการทำงานของขาของตัวโมดูล DCBT-24AX	10
ตารางที่ 2.2 ฟังก์ชั่นการทำงานของขาของ ATmega168V	12
ตารางที่ 2.3 ฟังก์ชั่นการทำงานของขาต่างๆของชิพ nRF24L01 ภายในโมดูล DCBT-24AX	15
ตารางที่ 2.4 คำสั่งต่างๆ ของอุปกรณ์รับ-ส่งข้อมูล ชิพ nRF24L01	17
ตารางที่ 2.5 ตำแหน่ง Registers ที่เกี่ยวข้อง	17
ตารางที่ 2.6 การตั้งค่า Configuration (00)	18
ตารางที่ 2.7 การตั้งค่า RF Gain (06)	18
ตารางที่ 2.8 การแสดงสถานะของชิพ nRF24L01 (07)	19
ตารางที่ 2.9 การอ่านสถานะของชิพ nRF24L01 (FF)	19
ตารางที่ 3.1 การเก็บข้อมูลสถานการณ์ออกนอกพื้นที่หรือการแจ้งเหตุฉุกเฉิน	47
ตารางที่ 4.1 ผลการทดลองระยะทางที่อุปกรณ์ภาครับสัญญาณสามารถรับข้อมูลได้ ในแต่ละระดับกำลังการส่ง ของอุปกรณ์ภาคส่งตัวที่ 1	51
ตารางที่ 4.2 ผลการทดลองระยะทางที่อุปกรณ์ภาครับสัญญาณสามารถรับข้อมูลได้ ในแต่ละระดับกำลังการส่ง ของอุปกรณ์ภาคส่งตัวที่ 2	51
ตารางที่ 4.3 ผลการทดลองระยะทางที่อุปกรณ์ภาครับสัญญาณสามารถรับข้อมูลได้ ในแต่ละระดับกำลังการส่ง โดยใช้อุปกรณ์การส่งสัญญาณทั้ง 2 ตัวพร้อมกัน	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์

สภาพเศรษฐกิจปัจจุบันผู้คนจำเป็นต้องออกมาหางานทำกันมากขึ้น ส่งผลให้เวลาที่มีในบ้านมีอย่างจำกัด ด้วยเวลาภายในบ้านที่น้อยนิดนั้น ยังต้องแบ่งไปเพื่อทำความสะอาดบ้านหรือทำงานบ้านอีกด้วย นั่นคือแบ่งสรรกันอย่างไม่สามารถหลีกเลี่ยงได้ อาจทำให้ละเลยการดูแลเด็กหรือคนชราภายในบ้าน ทำให้เกิดอุบัติเหตุภายนอกบ้านต่างๆขึ้นกับเด็กหรือคนชราได้ จากสถิติการเกิดอุบัติเหตุในเด็กและผู้สูงอายุไทย พบว่าเด็กและผู้สูงอายุไทยเสียชีวิตจากการเกิดอุบัติเหตุประมาณ 5,000-6,000 คนต่อปี ส่วนใหญ่พบในเด็กเล็กและผู้สูงอายุที่ผู้ปกครองหรือผู้ดูแล ไม่สามารถดูแลได้อย่างทั่วถึง และอุบัติเหตุที่พบบ่อยเกิดจากการจมน้ำ อุบัติเหตุจากการจราจร และการพลัดตกหกล้ม โดย 50% ของผู้สูงอายุที่หกล้ม อาจมีอันตรายถึงขั้นเสียชีวิตได้ สาเหตุของการเกิดอุบัติเหตุเกิดจาก ธรรมชาติของเด็กและผู้สูงอายุ ไม่ชอบอยู่นิ่ง ไม่เชื่อฟังคำสั่งของผู้ปกครองหรือคนดูแล และชอบที่จะหลบออกมาภายนอกบ้าน ทำให้ได้รับอุบัติเหตุได้ง่าย

จึงได้เกิดแนวความคิดในการสร้างอุปกรณ์แจ้งเตือนการออกนอกพื้นที่บริเวณบ้านของเด็กและคนชรา มีอุปกรณ์รับส่งข้อมูลแบบไร้สาย และการนำเอาเทคโนโลยีของระบบสมองกลฝังตัว (Embedded System) ที่กำลังมีการนำมาใช้อย่างแพร่หลายมาประยุกต์ใช้ในการแสดงผลผ่านทางเว็บแอพพลิเคชัน(Web Application) ในรูปแบบเวลาจริง(Real-time) อีกด้วย อีกทั้งมีการเก็บข้อมูลการออกนอกพื้นที่ ซึ่งจะทำให้ง่ายต่อการสถิติการออกนอกพื้นที่ย้อนหลัง อีกทั้งยังเป็นการป้องกันและหลีกเลี่ยงอุบัติเหตุเมื่อเด็กและคนชราไม่อยู่ในสายตาของผู้ดูแลด้วย

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาการทำงานของอุปกรณ์สื่อสารไร้สาย
2. เพื่อศึกษาโครงสร้างและการทำงานของไมโครคอนโทรลเลอร์ตระกูล AVR
3. เพื่อศึกษาการเขียนโปรแกรมภาษาซี ที่ใช้ควบคุมการทำงานของไมโครคอนโทรล-เลอร์
4. เพื่อศึกษาโครงสร้างและการทำงานของระบบสมองกลฝังตัวด้วย iBoard และระบบปฏิบัติการ uClinux สำหรับควบคุมอุปกรณ์ iBoard
5. เพื่อศึกษาการทำงานระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ iBoard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เพื่อศึกษาการเขียนโปรแกรมการจัดเก็บข้อมูล โดยใช้โปรแกรม SQLite
7. เพื่อศึกษาและพัฒนาเว็บเพจ(Webpage) โดยใช้โปรแกรมภาษา PHP และนำเทคนิค AJAX เข้ามามีส่วนร่วมในการสร้างเว็บเพจด้วย
8. เพื่อออกแบบและสร้างชุดอุปกรณ์แจ้งเตือนการออกนอกพื้นที่บริเวณบ้าน

1.3 ขอบเขตของปริญญานิพนธ์

1. ออกแบบและสร้างอุปกรณ์แจ้งเตือนการออกนอกพื้นที่ทางภาครับและภาคส่งแบบไร้สายที่สามารถส่งข้อมูล เพื่อนำไปเก็บไว้ในฐานข้อมูลของ iBoard

2. ใช้ระบบสมองกลฝังตัวบนระบบปฏิบัติการลินุกซ์(Linux) ของ iBoard สำหรับประมวลผล, จัดเก็บข้อมูลการทำงาน และทำการแสดงผลออกทางเว็บในรูปแบบเวลาจริงทางหน้าเว็บเพจ เพื่อแสดงผลออกมาให้ผู้ใช้งานสามารถตรวจสอบได้ในรูปแบบเวลาจริง และผู้ใช้สามารถเรียกตรวจสอบข้อมูลย้อนหลังได้

1.4 ประโยชน์ที่คาดว่าจะได้รับจากปริญญานิพนธ์

1. เรียนรู้การทำงานของอุปกรณ์สื่อสารไร้สาย
2. สามารถใช้ไมโครคอนโทรลเลอร์สื่อสารระหว่างกันแบบไร้สายได้
3. ได้เรียนรู้การเขียนโปรแกรมภาษาซี ที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์
4. สามารถเรียนรู้การทำงานของระบบสมองกลฝังตัวด้วย iBoard
5. ศึกษาและเรียนรู้วิธีใช้งาน ระบบปฏิบัติการ uClinux
6. สามารถเชื่อมต่อไมโครคอนโทรลเลอร์กับ iBoard โดยผ่านพอร์ตอนุกรมได้
7. สามารถแสดงผลแบบและเก็บข้อมูลได้ โดยการทำงานของ iBoard
8. สามารถนำอุปกรณ์ที่สร้างขึ้น ไปใช้งานได้จริง เพื่อให้เกิดประโยชน์แก่สังคม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนการดำเนินงาน

ตารางที่ 1.1 ขั้นตอนการดำเนินงาน

ขั้นตอนที่	วิธีการดำเนินงาน	ระยะเวลาการดำเนินงาน								
		มิถุนายน 51	กรกฎาคม 51	สิงหาคม 51	กันยายน 51	ตุลาคม 51	พฤศจิกายน 51	ธันวาคม 51	มกราคม 52	กุมภาพันธ์ 52
1.	ศึกษาและรวบรวมทฤษฎีที่เกี่ยวข้อง	←	—	→						
2.	วิเคราะห์และออกแบบฮาร์ดแวร์ในส่วนของภาครับและภาคส่งแบบไร้สาย พร้อมทั้งออกแบบเว็บเพจ ในการแสดงข้อมูลเบื้องต้น		←	—	→					
3.	จัดซื้ออุปกรณ์และทำการเขียนโปรแกรมควบคุมการรับส่งสัญญาณลงไมโครคอนโทรลเลอร์ AVR Board			↔						
4.	ทดสอบและแก้ไขวงจรขั้นต้น			←	—	→				
5.	จัดทำเอกสารปริญญานิพนธ์ของภาคเรียนที่ 1			←	—	→				
6.	เขียนโปรแกรมควบคุมการทำงานของ iBoard โดยใช้ระบบปฏิบัติการ uClinux				←	—	→			
7.	เขียนโปรแกรมเพิ่มในส่วนของความสามารถในการทำงานของไมโครคอนโทรลเลอร์				←	—	—	→		
8.	เขียนเว็บเพจเพื่อแสดงข้อมูล ตามที่ได้ออกแบบไว้					←	—	→		
9.	เชื่อมต่อการส่งข้อมูลจาก AVR Board ไปเก็บยัง iBoard โดยใช้พอร์ตอนุกรมเป็นตัวเชื่อมต่อ และนำข้อมูลที่รับมาแสดงออกทางเว็บเพจ ในรูปแบบเวลาจริง						←	—	→	
10.	ทดสอบการทำงานของระบบ							←	—	→
11.	จัดทำเอกสารปริญญานิพนธ์ฉบับสมบูรณ์								↔	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 อุปกรณ์ที่ต้องใช้

1.6.1 ฮาร์ดแวร์(Hardware)

- ชุดโมดูล DCBT-24AX จำนวน 3 ชุด
- ชุดพัฒนา Embedded Linux iBoard จำนวน 1 เครื่อง
- เครื่องคอมพิวเตอร์สำหรับพัฒนาโปรแกรม จำนวน 1 เครื่อง
- หลอดไฟ LED จำนวน 4 ดวง

1.6.2 ซอฟต์แวร์(Software)

- ระบบปฏิบัติการ uClinux
- โปรแกรมภาษาซี สำหรับเขียนคำสั่งควบคุมการทำงานของ DCBT-24AX
- โปรแกรมการเขียนหน้าเว็บเพจ
- โปรแกรม SQLite สำหรับการเก็บฐานข้อมูล

1.7 เนื้อหาภายในปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ประกอบไปด้วยเนื้อหาในบทต่างๆดังนี้

บทที่ 1 บทนำเป็นบทที่กล่าวถึงความเป็นมาและความสำคัญของปริญญานิพนธ์ วัตถุประสงค์ ขอบเขต ประโยชน์ที่ได้รับจากการทำปริญญานิพนธ์ ขั้นตอนการดำเนินงาน และอุปกรณ์ที่ใช้ในการทำปริญญานิพนธ์

บทที่ 2 นำทฤษฎีข้อมูลของเนื้อหาที่เกี่ยวข้องกับปริญญานิพนธ์ฉบับนี้มารวบรวมไว้ เพื่อเป็นพื้นฐานในการศึกษาหรือการทดลองตามปริญญานิพนธ์นี้

บทที่ 3 การออกแบบการทดลองเป็นการออกแบบโครงสร้างทั้งด้านซอฟต์แวร์และด้านฮาร์ดแวร์ เพื่อเตรียมให้ชุดอุปกรณ์พร้อมใช้งานจริง

บทที่ 4 เป็นการนำเสนอข้อมูลผลการทดลองตามความเป็นจริงที่ได้จากการทดลอง เพื่อเตรียมนำไปสรุปผล

บทที่ 5 บทสุดท้ายที่นำผลการทดลองในบทที่ 4 มาสรุปผล แล้วชี้แจงปัญหาและอุปสรรคที่เกิดขึ้นระหว่างการออกแบบและสร้างชุดอุปกรณ์ หรือระหว่างการทดลอง พร้อมชี้นำแนวทางในการศึกษาและพัฒนาชุดอุปกรณ์ต่อๆไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

ปริญญานิพนธ์นี้มีวัตถุประสงค์คือ การประยุกต์ใช้เทคโนโลยีสมองกลฝังตัวบนระบบปฏิบัติการลินุกซ์(Embedded Linux System) และอุปกรณ์สื่อสารแบบไร้สาย โดยในการทำงานจะแบ่งได้สองส่วน คือในส่วนแรกเป็นอุปกรณ์ภาครับและภาคส่งที่ใช้ในการติดต่อสื่อสารแบบไร้สาย เพื่อตรวจสอบการออกนอกพื้นที่ โดยในการรับส่งข้อมูลระหว่างสองจุดนี้ใช้ชุดโมดูล DCBT-24AX ที่ภายในโมดูลประกอบด้วยไอซีไมโครคอนโทรลเลอร์เบอร์ ATmega168 และชิพ nRF24L01 เพื่อใช้ในการรับส่งข้อมูลทั้งสองจุด และในส่วนที่สองไมโครคอนโทรลเลอร์ที่จุดรับสัญญาณจะควบคุมการส่งข้อมูลต่อไป โดยผ่านพอร์ตอนุกรมส่งข้อมูลไปยังบอร์ดสมองกลฝังตัวรุ่น iBoard เป็นตัวรับข้อมูล เพื่อเก็บในฐานข้อมูล และแสดงผลผ่านทางเว็บเซิร์ฟเวอร์ สามารถเรียกดูได้ในแบบเวลาจริง ดังนั้นเพื่อให้อุปกรณ์ทำงานได้ตามขอบเขตที่กำหนดไว้ จึงต้องศึกษาข้อมูลทฤษฎีที่สำคัญ ซึ่งสามารถแบ่งออกเป็นกลุ่มต่างๆ ตามลักษณะการทำงานดังนี้

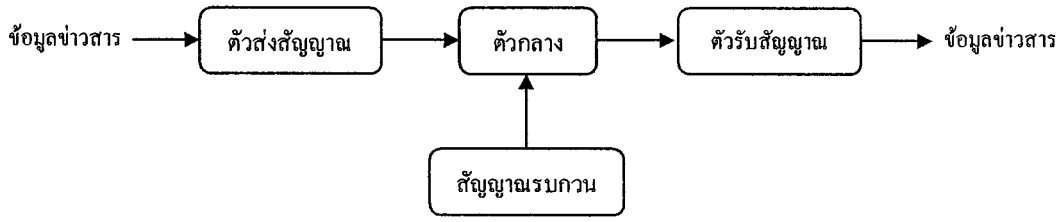
2.1 ทฤษฎีการสื่อสารไร้สาย

การสื่อสารคือ กระบวนการสำหรับแลกเปลี่ยนข้อมูล รูปแบบอย่างง่ายของข้อมูล คือจะต้องส่งจากผู้ส่งข้อมูลหรืออุปกรณ์เข้ารหัส ไปยังผู้รับข้อมูลหรืออุปกรณ์ถอดรหัสที่สามารถแปลความหมายของข้อมูลข่าวสารได้ แต่การสื่อสารก็มีอุปสรรคด้านระยะทางที่ห่างไกลกันมากระหว่างผู้ส่งสารและผู้รับสาร ดังนั้นจึงมีการพัฒนาวิธีการและเทคนิคใหม่ๆ เพื่อลดขีดจำกัดในด้านต่างๆ ของการสื่อสาร สำหรับคำว่าระบบสื่อสารไร้สายนั้น แสดงให้ทราบว่าสื่อตัวกลางที่นำพาสัญญาณนั้นไม่จำเป็นต้องใช้สาย แต่มีการใช้เทคนิคและวิธีการเข้ามาช่วยในการนำพาข้อมูลข่าวสารผ่านตัวกลางที่ไม่สามารถมองเห็นได้

2.1.1 องค์ประกอบของระบบสื่อสารไร้สาย

องค์ประกอบของระบบสื่อสารแบบไร้สายนั้น มีลักษณะเช่นเดียวกับองค์ประกอบโดยทั่วไปของระบบการสื่อสารทุกประเภท โดยมีอยู่ 4 ส่วน ดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 ส่วนประกอบของระบบสื่อสาร

จากรูปที่ 2.1 ระบบสื่อสารจะมีส่วนประกอบพื้นฐานที่เหมือนกันอยู่ 4 อย่างดังนี้

ตัวส่งสัญญาณ(Transmitter)

ตัวส่งสัญญาณคือ วงจรอิเล็กทรอนิกส์ที่ถูกออกแบบมาเพื่อที่จะแปลงข้อมูลข่าวสารให้เป็นสัญญาณที่สามารถที่จะส่งออกไปในตัวกลางหรือช่องสัญญาณ ได้แก่ อุปกรณ์ที่ใช้ในการส่งสัญญาณคลื่นวิทยุไมโครเวฟ เครื่องส่งวิทยุกระจายเสียง เป็นต้น

สื่อตัวกลางหรือหรือช่องสื่อสาร(Communication Channel)

เป็นตัวกลางที่ให้สัญญาณอิเล็กทรอนิกส์สามารถเดินทางจากสถานที่ที่หนึ่งผ่านไปยังอีกสถานที่หนึ่งได้ ตัวกลางในระบบการสื่อสารไร้สายได้แก่ คลื่นวิทยุ แต่ในตัวกลางนี้มีส่วนทำให้สัญญาณถูกลดทอนลงไปได้ส่วนหนึ่ง

สัญญาณรบกวน(Noise)

เป็นสัญญาณของพลังงานรูปแบบต่างๆ ที่มีลักษณะไม่แน่นอนเข้ามารบกวนในระบบสื่อสาร มีผลรบกวนสัญญาณข้อมูลที่ถูกส่งมาในช่องสื่อสาร ในบางครั้งสัญญาณรบกวนอาจเกิดขึ้นในวงจรของตัวรับสัญญาณก็ได้ สำหรับระบบสื่อสารไร้สายนั้น สัญญาณรบกวนที่มีอยู่ทั่วไปในอากาศ เช่น สัญญาณจากปรากฏการณ์ฟ้าแลบ เป็นต้น

ตัวรับสัญญาณ(Receiver)

เป็นวงจรอิเล็กทรอนิกส์ที่ถูกออกแบบให้รับสัญญาณที่ถูกส่งออกมาทางช่องสื่อสารและทำการแปลงสัญญาณให้กลับไปอยู่ในรูปแบบที่ผู้รับปลายทางสามารถเข้าใจได้

2.1.2 ตัวกลางของระบบการสื่อสาร

ในระบบการสื่อสารไร้สาย จะใช้สื่อหรือตัวกลางประเภทคลื่นความถี่หรือที่เรียกว่า “คลื่นพาหะ” ในการรับส่งสารผ่านบรรยากาศ มีอยู่ 2 รูปแบบ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบที่ใช้คลื่นวิทยุเป็นพาหะ(Radio Carier)

ได้แก่ ระบบโทรศัพท์เคลื่อนที่ ระบบสื่อสารผ่านดาวเทียม ระบบวิทยุกระจายเสียง ระบบคลื่นไมโครเวฟ และระบบแลนไร้สาย จุดเด่นของระบบนี้อยู่ที่การมีมาตรฐานที่ยอมรับกันทั่วหมายถึงสามารถใช้งานร่วมกันได้ และเป็นมาตรฐานกลางที่กำหนดโดย IEEE มีการวางรูปแบบให้รับส่งกันได้อย่างดี โดยเฉพาะเรื่องการรักษาความปลอดภัยของคลื่นสัญญาณที่อาจถูกดักฟังได้ กรณีนี้ก็มีวิธีการเข้ารหัส การสร้างระบบเอ็นคริปข้อมูล การให้บริการการใช้งาน และการดูแลรักษาเครือข่ายทำได้ง่ายกว่าแบบใช้สายมาก ทั้งนี้เพราะระบบได้รับการออกแบบมาให้เป็นแบบอัตโนมัติและตรวจสอบกันได้

เนื่องจากในปัจจุบันมีการกำหนดคลื่นความถี่สาธารณะ(Unlicense Band) คลื่นความถี่นี้เป็นคลื่นความถี่ที่ใช้เฉพาะในวงพื้นที่แคบ ๆ เพื่อลดการกวนระหว่างกัน โดยในยุคแรกใช้ความถี่ 900 MHz และต่อมาใช้ความถี่ 2.4 GHz ในอนาคตอันใกล้จะใช้ที่ความถี่ 5.6 GHz ดังนั้นในปริญญานิพนธ์ฉบับนี้ จึงได้นำอุปกรณ์รับ-ส่งสัญญาณไร้สายในรูปแบบคลื่นวิทยุ ความถี่ 2.4 GHz มาพัฒนา ซึ่งมีมาตรฐานรูปแบบ IEEE 802.11b รองรับ

ระบบที่ใช้คลื่นแสงเป็นพาหะ(Light Carrier)

ได้แก่ ระบบสื่อสารข้อมูลผ่านแสงอินฟราเรดที่นำไปประยุกต์ใช้ เช่น ระบบโลคอลแอเรียเน็ตเวอร์ก(Local Area Network: LAN) ไร้สาย ระบบสื่อสารผ่านเลเซอร์อินฟราเรด การสื่อสารผ่านใยแก้วนำแสง(Fiber Optic)

2.1.3 การรับส่งข้อมูลของการสื่อสารไร้สาย

แบบคลื่นความถี่แคบ(Narrowband)

จะรับส่งข้อมูลโดยแปลงเป็นบางช่วงสเปกตรัมของคลื่นแม่เหล็กไฟฟ้า เรียกว่า ISM(Industrial / Scientific / Medical) ที่มีความถี่แบ่งเป็น 3 ช่วง ได้แก่ 902-928 MHz, 2.14 - 2.484 GHz และ 5.725 - 5.850 GHz โดยการใช้งานต้องมีการขออนุญาตก่อนจากองค์กร FCC(Federal Communication Committee)

แบบคลื่นความถี่วิทยุแบบสเปรดสเปกตรัม(Spread Spectrum)

เป็นการวิธีการเปลี่ยนแปลงสัญญาณข้อมูลเพื่อให้ครอบคลุมพื้นที่ความถี่วิทยุมากกว่าความต้องการ เพื่อป้องกันคลื่นรบกวนและการดักฟัง โดยแบ่งความถี่ออกเป็น 2 ช่วง ได้แก่ 902 - 928 MHz และ 2.4 - 2.484 GHz ซึ่งไม่ต้องได้รับอนุญาตจาก FCC และการส่งแบบสเปรดสเปกตรัมที่ใช้กันโดยทั่วไปมีอยู่ 2 แบบ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. แบบ Frequency Hopping Spread Spectrum(FHSS)

การส่งสัญญาณรูปแบบนี้จะใช้ความถี่พาหะเพียงความถี่เดียว ที่จะมีการจะเปลี่ยนแปลงไปมาอย่างต่อเนื่องในลักษณะหรือรูปแบบที่เป็นที่เข้าใจตรงกันระหว่างผู้ส่งกับผู้รับ เมื่อทางด้านผู้ส่งกับผู้รับสามารถทำงานประสานกันได้แล้ว จะทำให้วิธีการส่งแบบนี้ป้องกันสัญญาณรบกวนที่เกิดจากความถี่ข้างเคียงได้เป็นอย่างดี นอกจากนั้นแล้วรูปแบบที่ตกลงกันไว้ระหว่างผู้ส่งกับผู้รับก็สามารถที่จะปรับเปลี่ยนไปได้ตลอดเวลาทำให้เกิดความปลอดภัยของข้อมูลสูง

2. แบบ Direct Sequence Spread Spectrum(DSSS)

รูปแบบนี้จะใช้การเข้ารหัส โดยการแปลงเลขฐานสองแต่ละบิทในข้อมูลเดิมที่จะส่ง ไปให้อยู่ในรูปแบบเลขฐานสองที่มีความยาวเพิ่มมากขึ้น ตัวอย่างเช่น ข้อมูลเลขฐานสอง 1 อาจจะถูกแปลงเป็น 0010010101 และข้อมูล 0 จะถูกแปลงเป็นอินเวอร์ส(Inverse) ของ 1 คือ 1101101010 แล้วข้อมูลที่แปลงแล้วนี้จะถูกส่งไปพร้อมๆ กัน ในลักษณะขนาน หากผู้รับสามารถจดจำรูปแบบการแปลงข้อมูลได้ก็จะได้รับข้อมูลที่ส่งไป โดยที่สัญญาณรบกวนไม่สามารถทำให้ข้อมูลเสียหายไปได้ หรือหากรูปแบบที่ส่งไปเกิดผิดพลาด ทางฝ่ายรับก็สามารถที่จะใช้เทคนิคในทางสถิติเพื่อกู้ข้อมูลให้กลับคืนมาได้ โดยการเลือกใช้งานแบบต้องการประสิทธิภาพจะใช้แบบ DSSS แต่ถ้าต้องการด้านการใช้ตัวแปลงขนาดเล็กและราคาไม่แพง เช่นในโน้ตบุ๊ค ก็ควรเลือกแบบ FHSS

แบบคลื่นความถี่วิทยุสัญญาณอินฟราเรด

โดยอินฟราเรดเป็นส่วนหนึ่งของสเปกตรัมแม่เหล็กไฟฟ้าที่อยู่เหนือคลื่นวิทยุและต่ำกว่าแสงที่มองเห็น แสงอินฟราเรดสามารถใช้ส่งข้อมูลได้ถึงแม้ว่าการส่งจะถูกจำกัดให้เป็นแนวเส้นตรงที่จะต่อเครื่องพีซีเข้ากับเครื่องพิมพ์หรือคอมพิวเตอร์เครื่องอื่นเพื่อแลกเปลี่ยนข้อมูลได้โดยไร้สาย

2.2 ชุดโมดูลสื่อสารไร้สาย DCBT-24AX

เป็นโมดูลสำเร็จรูปใช้สำหรับรับส่งข้อมูล ผ่านคลื่นวิทยุระยะไกล มีความสามารถในการส่งข้อมูลที่รวดเร็ว มีเสถียรภาพสูง ขนาดเล็ก ปรับแต่งสำเร็จรูป ประกอบด้วย ชิพ nRF24L01 พร้อมมีไมโครคอนโทรลเลอร์ AVR รุ่น Mega168V ภายในตัวโมดูล ทำให้สามารถนำมาประยุกต์ใช้งานเป็นอุปกรณ์ควบคุมแบบไร้สาย หรือใช้งานอื่นๆได้อีกหลากหลาย

2.2.1 คุณสมบัติของ DCBT-24AX

1. สามารถใช้เป็นตัวรับและส่งในตัว ในรูปแบบ Half-Duplex (การรับ-ส่งข้อมูลต่างช่วงเวลา) โดยมี transceiver IC รุ่น nRF24L01 พร้อมสายอากาศแบบ Multilayer chip +2 dBi gain

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. มีไมโครคอนโทรลเลอร์ในตัว รุ่น AVR 8-bit RISC ATmega168V

3. ส่งได้ระยะ 30 m ในที่โล่ง

4. ความถี่ในการใช้งานที่ 2.4GHz ISM band และสามารถตั้งช่องความถี่ ในการใช้งานได้ถึง 84 ช่องสัญญาณ (2.400 - 2.483 GHz)

5. ความเร็วในการรับส่งข้อมูล 1 Mbpsและ 2 Mbps

6. ทำงานที่แรงดัน (power supply) 2.7 V ถึง 5.25 V

7. ใช้กระแส 17 mA ขณะส่งข้อมูล และ 18 mA ขณะรับข้อมูล

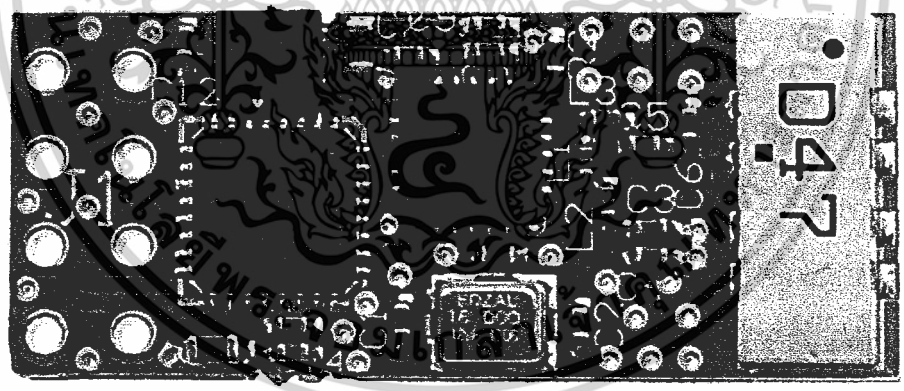
8. มีรูปแบบและความเร็วในการรับ-ส่งข้อมูลแบบ GFSK(Gaussian Frequency Shift Keying)

9. รูปแบบการ Interface แบบ SPI

10. ขนาดเล็ก (25mm x 10mm) และน้ำหนักเบา (1.28 gr)

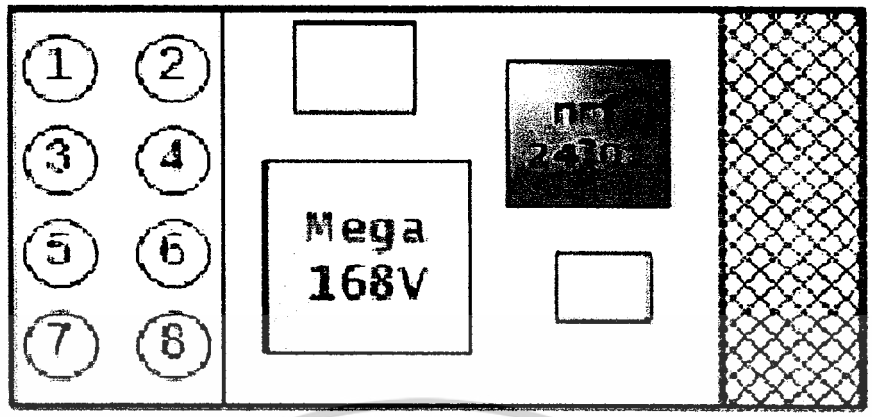
2.2.2 รูปร่างและขนาดของ DCBT-24AX

ชุดโมดูล DCBT-24AX มีขนาดเล็กมาก และมีน้ำหนักเบา ภายในประกอบด้วยไมโครคอนโทรลเลอร์ รุ่น AVR 8-bit RISC ATmega168V และชิพ nRF24L01



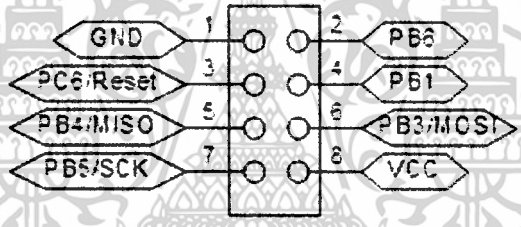
รูปที่ 2.2 โมดูล DCBT-24AX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ส่วนประกอบของโครงสร้าง DCBT-24AX

2.2.3 ฟังก์ชันการทำงานแต่ละขาของ DCBT-24AX



รูปที่ 2.4 ฟังก์ชั่นขาการทำงานของตัวโมดูล DCBT-24AX

ตารางที่ 2.1 ฟังก์ชั่นการทำงานของขาของตัวโมดูล DCBT-24AX

ขาที่	ชื่อ	รายละเอียด
1	GND	ขา Ground ต่อสายดิน(0 V)
2	PB6	ขาสำหรับต่อพอร์ตใช้งานทั่วไป
3	PC6/Reset	ขาสำหรับโปรแกรมตัวชิพ ATmega168V
4	PB1	ขาสำหรับต่อพอร์ตใช้งานทั่วไป
5	PB4/MISO	ขาสำหรับโปรแกรมตัวชิพ ATmega168V
6	PB3/MOSI	ขาสำหรับโปรแกรมตัวชิพ ATmega168V
7	PB5/SCK	ขาสำหรับโปรแกรมตัวชิพ ATmega168V
8	VCC	ไฟเลี้ยง 2.7 V - 5.25 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 ไมโครคอนโทรลเลอร์ตระกูล AVR รุ่น ATmega168V

ATmega 168 เป็นไมโครคอนโทรลเลอร์ตระกูล AVR ของบริษัท Atmel มีคุณสมบัติดังนี้

สถาปัตยกรรมขั้นสูงแบบ RISC

- สถาปัตยกรรมภายในถูกออกแบบให้ใช้แบบ RISE(Reduce Instruction Set Computer) RISE คือ ทำให้การประมวลผลมีความเร็ว 1 คำสั่ง / 1 Clock หรือ CPU สามารถประมวลคำสั่งได้ 1 MIPS / MHz
- ชุดคำสั่ง 131 คำสั่ง โดยส่วนใหญ่ทำงานที่ 1 รอบสัญญานาฬิกา (clock cycle)
- รีจิสเตอร์ขนาด 8 บิต จำนวน 32 ตัว
- ความเร็วในการประมวลผลมากกว่า 20 ล้านคำสั่งต่อวินาที(Million Instructions Per Second : MIPS)

หน่วยความจำ

- แบบ Flash 16 Kbytes สามารถเขียน/ลบโปรแกรมได้ 10,000 ครั้ง
- แบบ EEPROM 512 Bytes สามารถเขียน/ลบโปรแกรมได้ 100,000 ครั้ง
- แบบ SRAM 1 Kbytes

ไฟเลี้ยง

- ระหว่าง 1.8 ถึง 5.5 VDC

ความถี่สัญญาณนาฬิกา

- 0 ถึง 4 MHz

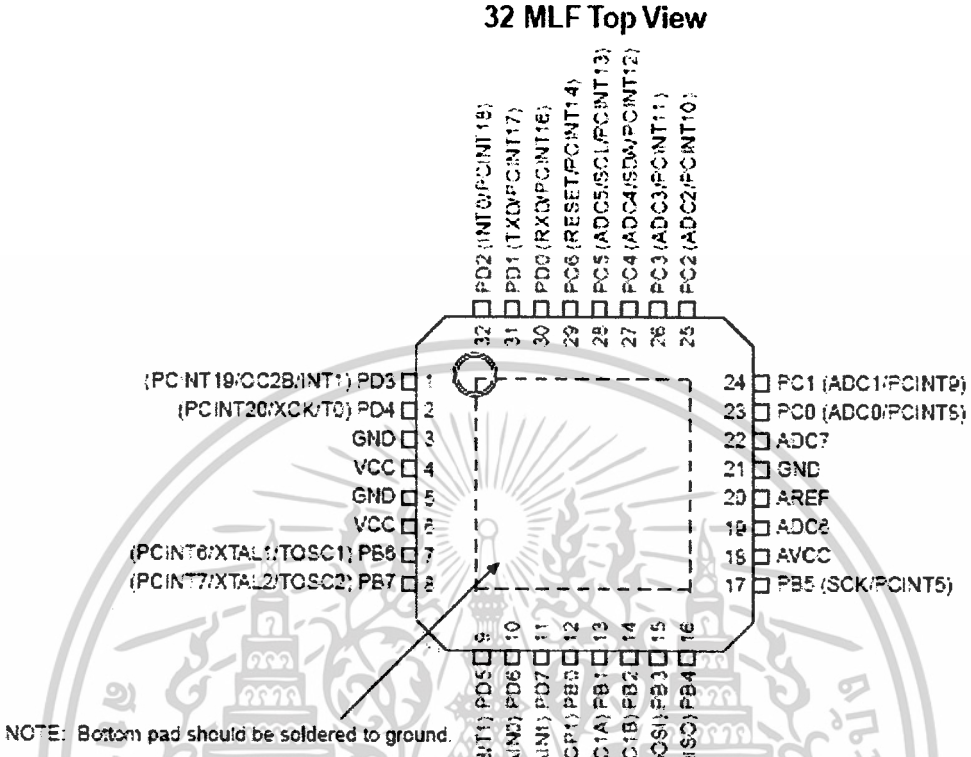
การรองรับอุปกรณ์ต่อพวง

- ตัวจับเวลา(Timer-counter) ขนาด 8 บิต 2 ตัว และขนาด 16 บิต 1 ตัว
- อุปกรณ์สื่อสารข้อมูลแบบอนุกรมแบบ SPI และ USART

อื่นๆ

- ระบบ Reset แบบอัตโนมัติเมื่อเริ่มจ่ายกระแสไฟฟ้าเข้าไมโครคอนโทรลเลอร์ (Power-on Reset)
- ฟังก์ชั่นตรวจสอบแรงดัน (Brown-out Detection)
- ระบบการขัดจังหวะจากภายในและภายนอก (Internal and External Interrupt)
- ระบบตรวจจับการทำงานผิดพลาดของซีพียู (Watchdog Timer)
- โหมดอนุรักษ์พลังงาน 5 โหมด ได้แก่ Idle, ADC Noise Reduction, Power-save, Power-down และ Standby
- ช่วงอุณหภูมิการทำงาน -40°C ถึง 85°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 โครงสร้างของไมโครคอนโทรลเลอร์ ATmega168V แบบ 32 ขา

ตารางที่ 2.2 ฟังก์ชั่นการทำงานของขาของ ATmega168V

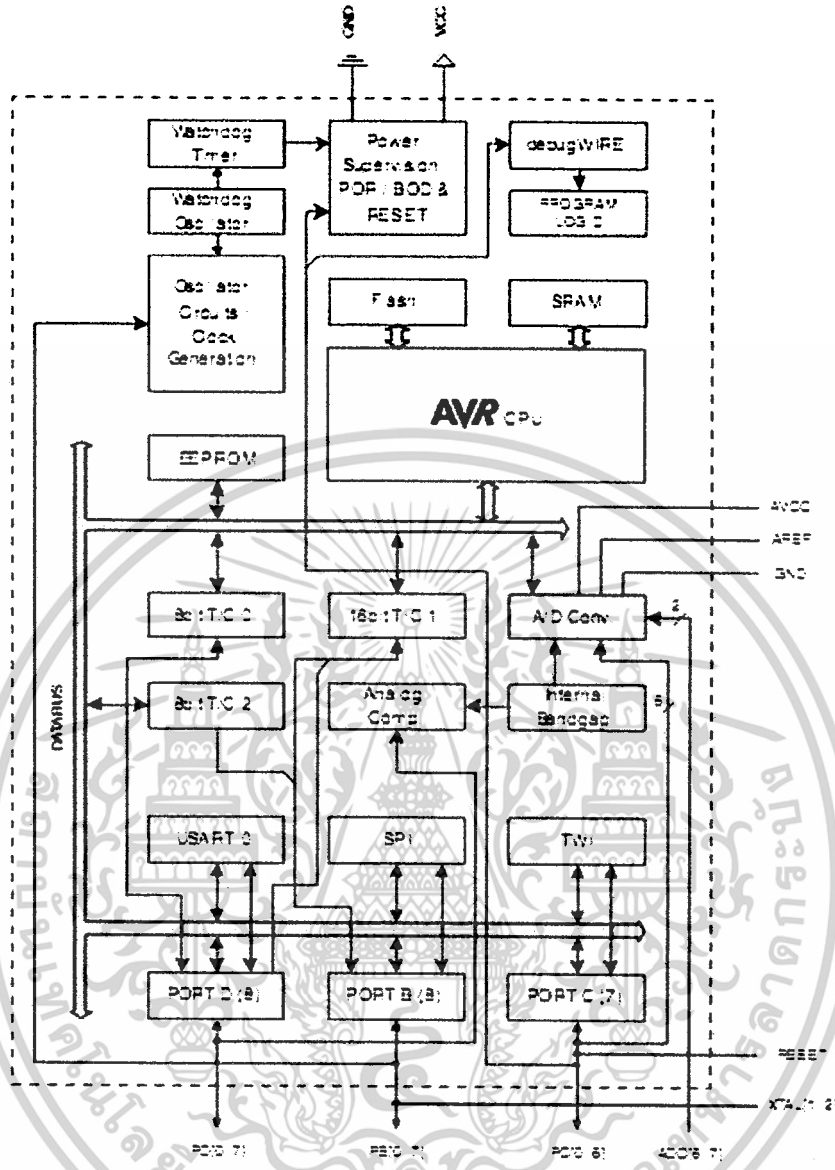
ขาที่	ชื่อ	รายละเอียด
1	GND	ขา Ground ต่อสายดิน(0 V)
2	VCC	ไฟเลี้ยง 1.8 V ถึง 5.5 V
3	Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2	- เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้แต่ละขาของพอร์ตสามารถตั้งค่าให้ pull up resistors (ภายในเป็นอิสระแยกจากกัน เพื่อดึงแรงดันของลอจิก 1 ให้เท่ากับ 5 V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 ฟังก์ชั่นการทำงานของขาของ ATmega168V (ต่อ)

ขาที่	ชื่อ	รายละเอียด
3		- สามารถใช้งานพิเศษตามความต้องการของ ATmega168 ขึ้นกับการตั้งค่าสัญญาณ clock ซึ่งขา PB6 ใช้เป็นขาแรงดัน Oscillator และ ขา input ของวงจรสัญญาณ clock ภายใน และขา PB7 ใช้เป็นขาแรงดัน Oscillator
4	Port C (PC5:0)	- เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้แต่ละขาของพอร์ตสามารถตั้งค่าให้ pull up resistors - สามารถใช้งานพิเศษตามความต้องการของ ATmega168
5	PC6/RESET	ขา Reset
6	Port D (PD7:0)	- เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้แต่ละขาของพอร์ตสามารถตั้งค่าให้ pull up resistors - สามารถใช้งานพิเศษตามความต้องการของ ATmega168
7	AVCC	ใช้จ่ายไฟให้กับวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอล มักต่อเข้ากับขา VCC
8	AREF	ขาแรงดันอ้างอิงที่ใช้งานในส่วนของวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอล มักต่อเข้ากับขา VCC
9	ADC7:6 (TQFP and QFN/MLF Package Only)	ขากำลังงานในการแปลงสัญญาณอนาล็อกเป็นดิจิตอล โดยให้ ช่องสัญญาณ 10 บิตจะมีเฉพาะในแพคเกจ TQFP และ QFN/MLF เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ ATmega168V

2.2.5 ชิพรับส่งสัญญาณไร้สาย nRF24L01

เป็นชิพไอซีที่ใช้สำหรับรับส่งข้อมูลในแบบอนุกรม ผ่านคลื่นวิทยุระยะไกล ใช้กับย่านความถี่ 2.4 GHz พร้อมมีเสาอากาศในตัว สามารถส่งข้อมูลรวดเร็ว มีเสถียรภาพสูง มีคุณสมบัติดังนี้

1. มีสถาปัตยกรรมแบบ QFN Package 20 Pin ขนาด 4x4 mm
2. ความถี่ในการใช้งานที่ 2.4 GHz
3. ความเร็วในการรับส่งข้อมูล 1 Mbps และ 2 Mbps
4. สามารถใช้เป็นตัวรับและส่งในตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

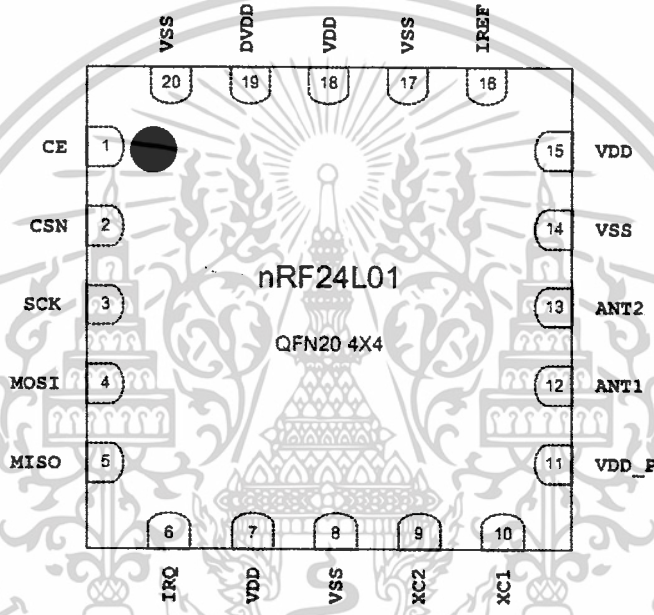
5. มีรูปแบบความเร็วสูงในการรับส่งข้อมูล โดยส่งแบบ GFSK(Gaussian Frequency Shift Keying)

6. ทำงานที่ความต่างศักย์ทางไฟฟ้า 1.9 – 3.6 V

7. กำลังงานเอาต์พุตปรับได้ 4 ระดับ คือ 0 dBm, -6 dBm, -12 dBm และ -18 dBm

8. สามารถตั้งช่องความถี่ ในการใช้งานด้วยโปรแกรมได้ถึง 126 ช่องสัญญาณ

9. โหมดของการรับส่งข้อมูล เป็นโหมด ShockBurst เพราะเกิดการผิดพลาดในการส่งข้อมูลต่ำ ส่วนมากนิยมใช้ในการส่งข้อมูลชนิดที่ต้องการความถูกต้องสูง



รูปที่ 2.7 ลักษณะขาของชิพไอซี nRF24L01

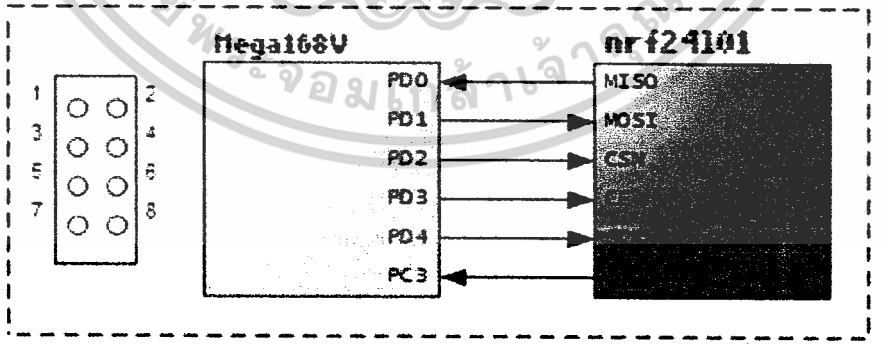
ตารางที่ 2.3 ฟังก์ชั่นการทำงานของขาต่างๆของชิพ nRF24L01 ภายในโมดูล DCBT-24AX

ขาที่	ชื่อ	รายละเอียด
1	CE	ขาควบคุมการทำงานของการรับ(Rx) และการส่ง(Tx) ข้อมูล
2	CSN	ขาควบคุมการสื่อสารของโมดูล ระหว่างไมโครคอนโทรลเลอร์ ATmega168V และ ชิพ nRF24L01
3	SCK	ขาสัญญาณนาฬิกา
4	MOSI	ขารับข้อมูลจากไมโครคอนโทรลเลอร์ ATmega168V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 ฟังก์ชั่นการทำงานของขาต่างๆของชิพ nRF24L01 ภายในโมดูล DCBT-24AX (ต่อ)

ขาที่	ชื่อ	รายละเอียด
5	MISO	ขาส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ ATmega168V
6	IRQ	เมื่อมีข้อมูลเข้ามา ขานี้จะตั้งค่าสถานะเป็น “0”
7	VDD	ไฟเลี้ยง 1.9 V ถึง 3.6 V
8	VSS	ขา Ground ต่อสายดิน(0 V)
9	XC2	ขาต่อ Crystal2
10	XC1	ขาต่อ Crystal1
11	VDD_PA	ไฟเลี้ยง 1.8 V ให้ภายในชิพ nRF24L01 และต้องเชื่อมต่อกับเสาอากาศทั้ง 2 ตัว (ANT1 และ ANT2)
12	ANT1	ขาต่อเสาอากาศ1
13	ANT2	ขาต่อเสาอากาศ2
14	VSS	ขา Ground ต่อสายดิน(0 V)
15	VDD	ไฟเลี้ยง 1.9 V ถึง 3.6 V
16	IREF	กระแสอ้างอิง โดยต่อกับตัวต้านทาน 22 kΩ ต่อลง Ground
17	VSS	ขา Ground ต่อสายดิน(0 V)
18	VDD	ไฟเลี้ยง 1.9 V ถึง 3.6 V
19	DVDD	ไฟเลี้ยงสำหรับ de-coupling
20	VSS	ขา Ground ต่อสายดิน(0 V)



รูปที่ 2.8 การเชื่อมต่อการทำงานของขาภายในโมดูล DCBT 24-AX ระหว่างไมโครคอนโทรลเลอร์ ATmega168 กับ ชิพรับส่งข้อมูล nRF24L01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.2.6 คำสั่งการตั้งค่าการใช้งานโมดูล DCBT-24AX

ในการตั้งค่าใช้งานเริ่มต้นของโมดูล DCBT-24AXนั้น มีรูปแบบดังนี้

ตารางที่ 2.4 คำสั่งต่างๆ ของอุปกรณ์รับ-ส่งข้อมูล ชิพ nRF24L01

คำสั่ง	Code	Comment
อ่านค่าจาก Register	0x00 + Address	Address = 0-0x1F
เขียนค่าลง Register	0x20 + Address	Address = 0-0x1F
อ่านค่าจาก Buffer การรับ	0x61	อ่านได้สูงสุด 32 ไบต์
เขียนค่าลง Buffer การส่ง	0xA0	เขียนได้สูงสุด 32 ไบต์
ล้าง Buffer ของการรับเข้า	0xE2	-
ล้าง Buffer ของการส่งออก	0xE1	-
อ่านสถานะ	0xFF	-

ตารางที่ 2.5 ตำแหน่ง Registers ที่เกี่ยวข้อง

ตำแหน่ง	ชื่อ Register	Comment
00	Configuration	ตั้งค่าการทำงานของชิพ nRF24L01
06	RF Gain	ตั้งค่ากำลังการส่ง
0A	RX Address P0	ตั้งรหัสของตัวรับหมายเลข 0
0B	RX Address P1	ตั้งรหัสของตัวรับหมายเลข 1
10	TX Address	ตั้งรหัสของตัวส่ง
11	RX Pipeline P0	ตั้งค่า Buffer ของการรับหมายเลข 0
12	RX Pipeline P1	ตั้งค่า Buffer ของการรับหมายเลข 1

รายละเอียดที่ต้องทำการตั้งค่านั้นมีอยู่ 4 ส่วน และหน้าที่การทำงานของแต่ละส่วน มีดังตารางที่ 2.6 – 2.9

1. Configuration (00) : **การตั้งค่าการทำงานของชิพ nRF24L01 จำนวน 8 บิต โดยแต่ละบิตมีหน้าที่ ดังนี้**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

104265

ตารางที่ 2.6 การตั้งค่า Configuration (00)

บิตที่	การทำงาน
7	ไม่ได้ใช้
6	"0" คือ เมื่อได้รับข้อมูลแล้ว จะให้ขา IRQ เป็น "0"
5	"0" คือ เมื่อส่งข้อมูลเสร็จแล้ว จะให้ขา IRQ เป็น "0"
4	"0" คือ เมื่อส่งข้อมูลค่าเดิมเสร็จแล้ว จะให้ขา IRQ เป็น "0"
3	"1" คือ ให้มีการตรวจสอบ CRC
2	จำนวนไบต์ตัวถอดรหัส CRC, "0" คือ ข้อมูล 1 ไบต์ และ "1" คือ ข้อมูล 2 ไบต์
1	"1" คือ Power Up และ "0" คือ Power Down
0	"1" คือ ตั้งให้เป็นตัวรับข้อมูล และ "0" คือ ตั้งให้เป็นตัวส่งข้อมูล

2. RF Gain (06) : การตั้งค่ากำลังการส่งของชิพ nRF24L01 จำนวน 8 บิต โดยแต่ละบิตมีหน้าที่ ดังนี้

ตารางที่ 2.7 การตั้งค่า RF Gain (06)

บิตที่	การทำงาน
7	ไม่ได้ใช้ มีค่าเป็น "0"
6	ไม่ได้ใช้ มีค่าเป็น "0"
5	ไม่ได้ใช้ มีค่าเป็น "0"
4	ใช้สำหรับการทดสอบอุปกรณ์
3	ตั้งอัตราการส่งข้อมูล "0" คือ 1 Mb/s , "1" คือ 2 Mb/s
2	ตั้งค่ากำลังวัตต์ของการส่ง "00" คือ -18 dBm, "01" คือ -12 dBm, "10" คือ -6 dBm , "11" คือ 0 dBm
1	
0	เป็น "1" เมื่อต้องการตั้งค่ากำลังการส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Status (07) : สถานะของชิพ nRF24L01จำนวน 8 บิต โดยแต่ละบิตมีหน้าที่ ดังนี้

ตารางที่ 2.8 การแสดงสถานะของชิพ nRF24L01 (07)

บิตที่	การทำงาน
7	สงวนไว้
6	เป็น “1” เมื่อได้รับข้อมูล และจะถูกยกเลิกเมื่อเขียนค่า “1” ลงไป
5	เป็น “1” เมื่อได้ส่งข้อมูล และจะถูกยกเลิกเมื่อเขียนค่า “1” ลงไป
4	เป็น “1” เมื่อได้ส่งข้อมูลเกินกว่า Buffer ของตัวส่ง และจะยกเลิกเมื่อเขียนค่า “1” ลงไป
3	จำนวนข้อมูลของ Pipeline สำหรับการเตรียมข้อมูล เพื่อการอ่านค่าจาก Buffer การรับ 000 – 101 : จำนวนข้อมูลของ Pipeline 110 : ไม่ได้ใช้ 111 : Buffer การรับว่างเปล่า
2	
1	
0	เป็น “1” เมื่อ Buffer การส่งเต็ม

4. Read Status (FF) : อ่านสถานะของชิพ nRF24L01จำนวน 8 บิต โดยแต่ละบิตมีหน้าที่ ดังนี้

ตารางที่ 2.9 การอ่านสถานะของชิพ nRF24L01 (FF)

บิตที่	การทำงาน
7	ไม่ได้ใช้ มีค่าเป็น “0”
6	เป็น “1” เมื่อต้องการส่งข้อมูลล่าสุดออกไปอีกครั้ง และจะถูก Reset เป็น “0” เมื่อมีการล้างข้อมูลใน Buffer ของการส่ง
5	เป็น “1” เมื่อ Buffer ของการส่งเต็ม
4	เป็น “1” เมื่อ Buffer ของการส่งว่างเปล่า
3	ไม่ได้ใช้ มีค่าเป็น “0”
2	ไม่ได้ใช้ มีค่าเป็น “0”
1	เป็น “1” เมื่อ Buffer การรับเต็ม
0	เป็น “1” เมื่อ Buffer การรับว่างเปล่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 iBoard

iBoard คือ สถาปัตยกรรมระบบฝังตัวที่รวบรวมความสามารถของ Linux อย่างเต็มรูปแบบมาไว้บนบอร์ดที่มีขนาดเล็กเพียง 55x65 มม. แต่เต็มเปี่ยมด้วยประสิทธิภาพและใช้พลังงานต่ำเพียง 1.5 วัตต์ ด้วยการออกแบบโมดูลให้มีความยืดหยุ่นสูงต่อการนำไปใช้งานและการนำไปพัฒนาต่อ จึงสามารถนำ iBoard ไปประยุกต์ใช้กับ Application ได้หลากหลาย คุณสมบัติที่เด่นชัดคือมี Ethernet Port พร้อมใช้งาน ที่มาพร้อมกับชุดซอฟแวร์ TCP/IP ที่มีเสถียรภาพสูงของ Linux จึงทำให้ iBoard เหมาะแก่การสร้างสรรค์โซลูชั่นที่ทำงานบนเครือข่ายอินเทอร์เน็ตเป็นอย่างมาก นอกจากนี้ยังมี USB Port ที่สามารถต่อเชื่อมกับอุปกรณ์ต่อพ่วงได้มากมาย เช่น USB Camera, USB Hard disk, USB Flash Drive, USB to Serial adaptor, Bluetooth, Wireless Lan, รวมถึง Legacy Asynchronous RS232, RS422 และ RS485 ที่มีให้พร้อมใช้งานสำหรับต่อพ่วงกับอุปกรณ์ เช่น Bacode scanner, Modem LED Message Board, LED Sign Board, RFID Reader, Industrial Sensor with MODBUS Protocol เป็นต้น นอกเหนือจากนี้ยังมีขา อินพุต/เอาต์พุต ให้อีก 48 GPIO ที่ให้สามารถปรับเปลี่ยนให้ใช้กับอุปกรณ์ได้หลากหลาย เช่น ระบบล็อคประตูด้วยกลไก, Relay, Sensor. Siren และ I^2C เป็นต้น

2.3.1 คุณสมบัติทางด้านฮาร์ดแวร์

1. มีระบบประมวลผล ETRAX 100LX มีสถาปัตยกรรม RISC เป็นระบบปฏิบัติการแบบ 32 บิต สามารถรันคำสั่งได้ 100 ล้านคำสั่งต่อวินาที
2. มีหน่วยความจำแฟลช 32 บิต ถึง 32 บิต
3. สามารถใส่หน่วยความจำภายนอก SDRAM 32 MB ถึง 64 MB
4. มีพอร์ตแลน 10/100 1 พอร์ต หัว RJ45 Ethernet
5. มีพอร์ต USB 2 พอร์ต (โดยมีบนบอร์ด 1 พอร์ตและสามารถรับเพิ่มได้)
6. มีขา I/O ทั้งหมด 20 ขา อยู่ จากทั้งหมด 48 ขา และมี 4 Serial UARTs หรือสามารถเลือกเป็น Parallel port
7. ประหยัดพลังงาน โดยใช้กระแสไฟเพียงเพียง 300mA ที่แรงดันไฟ 5V

2.3.2 คุณสมบัติทางด้านซอฟต์แวร์

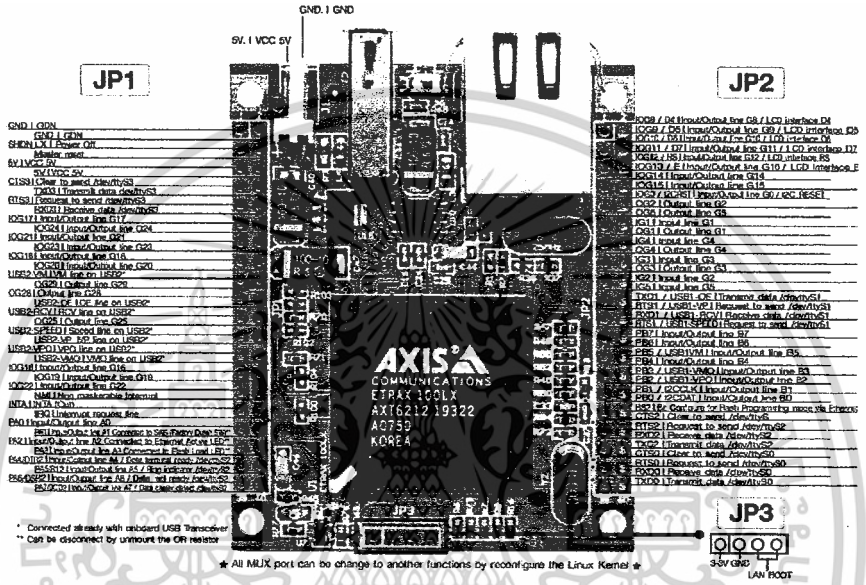
1. เริ่มการทำงานอย่างรวดเร็ว ภายใน 5 วินาที
2. มีเว็บเซิร์ฟเวอร์ อยู่ภายในตัว
3. มีระบบปฏิบัตการ Linux Kernel 2.4.x หรือ 2.6x

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

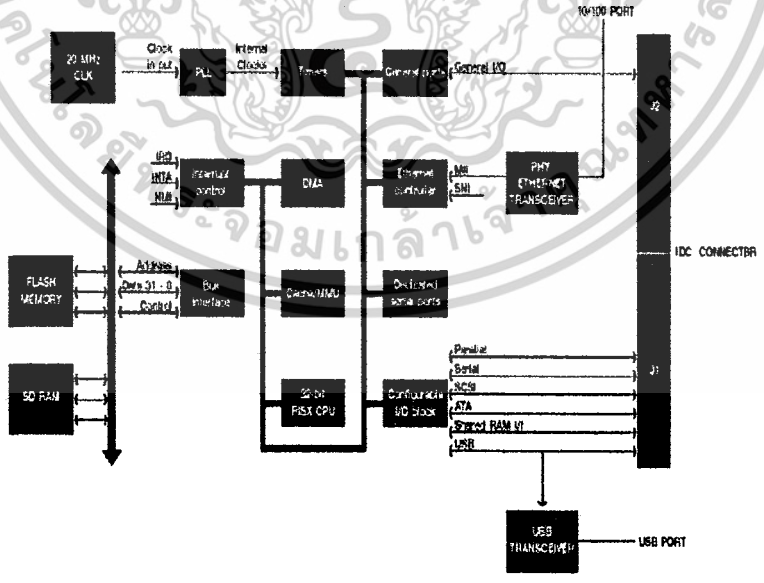
4. มีการรองรับ ipv4, ipv6

- PPP(Point-to-Point Protocol) สามารถทำการรองรับ asynchronous modem (เช่น GPRS/EDGE, Analog, ISDN)
- PPPoE ใช้สำหรับ ADSL , IPSec , VPN
- FTP, FTTP, SNMP

5. สามารถพัฒนาระบบปฏิบัติการภายในผ่านทางพอร์ตแลนได้



รูปที่ 2.9 ขาของ iBoard ทั้ง 40 ขา

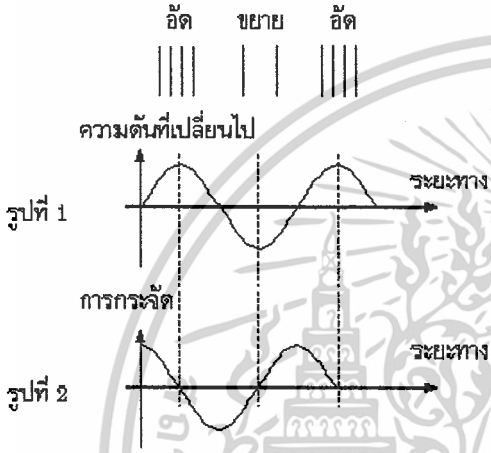


รูปที่ 2.10 บล็อกไดอะแกรมของโครงสร้าง iBoard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 เสียง

เสียงเกิดจากการสั่นของวัตถุ เมื่อแหล่งกำเนิดเสียงเกิดการสั่น จะทำให้โมเลกุลอากาศสั่นตามไปด้วย โดยมีความถี่เท่ากับการสั่นของแหล่งกำเนิดเสียง การสั่นของลำอากาศทำให้เกิดการเรียงตัวของโมเลกุลแตกต่างไปจากเดิม บางตำแหน่งโมเลกุลของอากาศจะเคลื่อนที่ไปอยู่ชิดติดกันมากขึ้นเรียกว่าช่วงอัด บางตำแหน่งโมเลกุลของอากาศจะอยู่ห่างกันมากขึ้นเรียกว่าช่วงขยาย ซึ่งพลังงานของการสั่นจะแผ่ออกไปรอบๆแหล่งกำเนิดเสียง



รูปที่ 2.11

ก.) การกระจายของโมเลกุลอากาศ

ข.) กราฟระหว่างความดันอากาศ – ระยะทาง

ค.) กราฟระหว่างการกระจัดของอากาศ – ระยะทาง

ดังนั้นคลื่นเสียงจึงเป็นคลื่นตามยาวเพราะโมเลกุลของอากาศจะสั่นในทิศเดียวกับทิศที่เสียงเคลื่อนที่ไป เสียงอาศัยตัวกลางในการเคลื่อนที่ คุณสมบัติของเสียงจึงเหมือนคลื่นทุกประการ การเปลี่ยนความดันอากาศนี้สามารถเคลื่อนที่ไปข้างหน้าจนถึงหูของผู้ฟังทำให้ได้ยินเสียง ความดังของเสียงจะขึ้นอยู่กับช่วงกว้างของการสั่น(แอมปลิจูด) ถ้าแอมปลิจูดมากเสียงจะดังมาก เมื่อคลื่นเสียงเดินทางเข้ากระทบเยื่อแก้วหู การเปลี่ยนความดันอากาศนี้สามารถเคลื่อนที่ไปข้างหน้าถึงหูของผู้ฟังทำให้เยื่อแก้วหูของผู้ฟังสั่น การสั่นจะแปลงเป็นกระแสประสาทส่งไปยังสมอง ตามปกติหูคนสามารถได้ยินเสียงในช่วงความถี่ 20 เฮิร์ตซ์ ถึง 20,000 เฮิร์ตซ์

เสียงที่คนเราสามารถได้ยินแต่ละเสียงอาจเหมือนกันหรือแตกต่างกัน ขึ้นอยู่กับคุณลักษณะของเสียงซึ่งมีอยู่ 3 ข้อ คือ

1. ความดัง(Loundness) หมายถึง ความรู้สึกได้ยินของมนุษย์ว่าดังมากดังน้อย ซึ่งเป็นปริมาณที่ไม่อาจวัดด้วยเครื่องมือใด ๆ ได้โดยตรง ความดังเพิ่มขึ้นตามความเข้มเสียง ความรู้สึกเกี่ยวกับความดังจะเป็นสัดส่วนโดยตรงกับระดับความเข้มเสียง อาจกล่าวว่า ความดังก็คือระดับความเข้มเสียง หูของคนสามารถรับเสียงที่มีความดังน้อยที่สุดคือ 0 dB และมากที่สุดคือ 120 dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. คุณภาพของเสียง(quality) หมายถึง คุณลักษณะของเสียงที่ได้ยิน เมื่อฟังเพลงจากวงดนตรีวงหนึ่งนั้น เครื่องดนตรี ทุกชนิดจะเล่นเพลงเดียวกัน แต่ผู้ฟังจะสามารถแยกได้ว่า เสียงที่ได้ยินนั้นมาจากดนตรีประเภทใด เช่น มาจากไวโอลิน หรือเปียโน เป็นต้นการที่สามารถแยกลักษณะของเสียงได้นั้นเพราะว่าคลื่นเสียงทั้งสองมีคุณภาพของเสียงต่างกัน คุณภาพของเสียงขึ้นอยู่กับจำนวนโอเวอร์โทนที่เกิดจากแหล่งกำเนิดเสียงนั้น ๆ นอกจากนี้คุณภาพของเสียงยังขึ้นกับ ความเข้มของเสียงอีกด้วย

3. ระดับเสียง(pitch) หมายถึง เสียงที่มีความยาวคลื่นและความถี่ต่างกัน โดยเสียงที่มีความถี่สูงจะมีระดับเสียงสูงส่วนเสียงที่มี ความถี่ต่ำจะมีระดับเสียงต่ำ

2.4.1 ลำโพง

ลำโพง(loudspeaker/speaker) เป็นอุปกรณ์ไฟฟ้าเชิงกลอย่างหนึ่ง ทำหน้าที่แปลงสัญญาณไฟฟ้าให้เป็นเสียง มีด้วยกันหลายแบบ

ลำโพงขนาดเล็กที่ใช้ต่อกับไมโครคอนโทรลเลอร์

1. ลำโพงแบบ Magnetic-diaphragm ขนาดเล็ก
   - diaphragm แบบโลหะ ให้เสียงดังแต่มีความต้านทานต่ำ ต้องมีทรานซิสเตอร์ช่วยขับอีกทีหนึ่ง
   - diaphragm แบบขดลวด และแม่เหล็กถาวร จะมีขนาดเล็กให้ระดับเสียงสูงได้ดี แต่ระดับเสียงต่ำไม่ค่อยดีนัก
   - diaphragm แบบพลาสติก ลำโพงแบบนี้จะให้ระดับเสียงต่ำได้ดีขึ้น
2. ลำโพงแบบเปียโซ(Piezo)
   - มีความต้านทานสูงสามารถต่อกับไมโครคอนโทรลเลอร์ได้โดยตรง
   - แผ่นลำโพงแบบเปียโซมีคุณสมบัติเมื่อป้อนแรงดันไฟฟ้าจะทำให้เกิดการบิดงอไปมาทำให้เกิดเสียงขึ้น
   - คุณสมบัติของเสียงขึ้นอยู่กับขนาดของแผ่น Piezo และกล่องกำทอน (resonance box)
3. บัซเซอร์(Buzzer)
   - คือลำโพงแบบแม่เหล็กหรือ แบบเปียโซที่มีวงจรกำเนิดความถี่ (oscillator) อยู่ภายในตัวเมื่อป้อนแรงดันสามารถกำเนิดเสียงได้ด้วยตัวเอง แต่ไม่สามารถเปลี่ยนความถี่ของเสียงได้ (ในปริญญานิพนธ์นี้ นำลำโพงแบบบัซเซอร์มาใช้ในชุดอุปกรณ์ภาครับสัญญาณ)

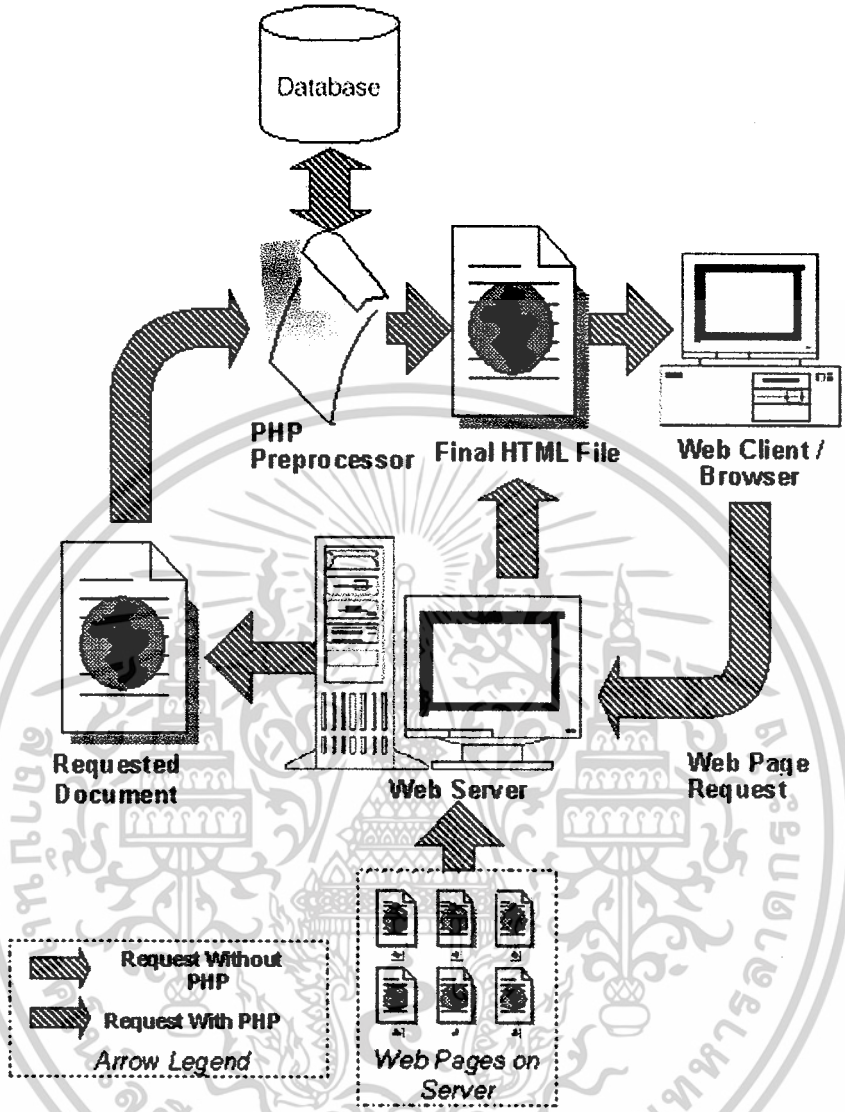
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.12 Buzzer แบบ diaphragm -Piezo

2.5 ภาษา PHP

PHP เป็นภาษาตัวแปลสคริปต์ ที่เรียกว่า language engine เพื่อเรียกใช้สคริปต์ที่เขียนขึ้น โดยไม่มีขั้นตอนกลางในการคอมไพล์ หรือไปเป็นรูปแบบไบนารี สคริปต์ส่วนใหญ่ที่ใช้สร้างโปรแกรมประยุกต์เว็บอยู่ในที่เดียวกับไฟล์ HTML ตามปกติไฟล์เก็บสคริปต์จะเก็บเป็นนามสกุล .php ถึงแม้ว่าบุคคลทั่วไปจะใช้นามสกุลเก่าคือ .php3 และ .phtml พื้นที่เก็บไฟล์เหล่านี้จะขึ้นกับการตั้งค่า Configuration ให้เว็บเซิร์ฟเวอร์ส่งผ่านไฟล์เหล่านี้ไปยังตัวแปร PHP พื้นที่จัดเก็บไฟล์หรือเอกสารนี้ได้รับอ้างถึงในฐานะ document root

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 โครงสร้างการทำงานของ PHP

2.5.1 ประเภทข้อมูล

การทำงานกับประเภทข้อมูลของ PHP แตกต่างจากภาษาอื่นเล็กน้อย โดย PHP เป็นภาษา richly typed ที่ตัวแปรไม่ต้องมีการประกาศเป็นประเภทข้อมูลเจาะจง เพราะ engine กำหนดประเภทที่ใช้ตามกฎ บางครั้งเรียกสิ่งนี้ว่าประเภทข้อมูลไดนามิคส์

PHP สนับสนุนประเภทข้อมูล

- integer
- float หรือ double

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- string
- boolean
- array
- object

2.5.2 การแปลงประเภทข้อมูล

จากการที่ PHP เป็นภาษา richly type และ engine สามารถแปลงระหว่างประเภทข้อมูลต่างกันในเวลาเรียกใช้ PHP สามารถแปลงหรือเจาะจงประเภทข้อมูลได้

วิธีพื้นฐานในการแปลงประเภทข้อมูลตัวแปรมี 2 วิธีคือ

1. การแปลงประเภทเชิงนัยยะ(implicit conversion)

การแปลงเชิงนัยยะเป็นการประเภทข้อมูลที่เกิดขึ้นโดย PHP engine อย่างอัตโนมัติ เมื่อมีการประมวลผล PHP จะกำหนดประเภทข้อมูลให้ตัวแปรให้ตามความเหมาะสม

2. การแปลงประเภทเชิงประจักษ์(empirical conversion)

ถึงแม้ว่า PHP จะแปลงตัวแปรให้อย่างอัตโนมัติ แต่ยังมีทางเลือกในการระบุประเภทข้อมูลได้เองด้วยการบังคับให้ภาษาแปลงประเภทข้อมูลโดยทำในสิ่งที่เรียกว่า type cast การกำหนดให้เติมหน้าตัวแปรด้วยประเภทข้อมูลในวงเล็บ จากนั้น PHP จะพยายามแปลงให้

2.5.3 การทำงานไฟล์ และไดเรคทอรี

เก็บสารสนเทศเป็นไฟล์ข้อความ(text file) สำหรับการใช้งานต่อไป จะเป็นการเก็บและโหลดด้วยการเขียนลงไฟล์และอ่าน เมื่อมีปริมาณข้อมูลมากต้องใช้ระบบจัดการฐานข้อมูล

ภาพรวมการประมวลผลไฟล์

1. การเขียนข้อมูลลงสู่ไฟล์มี 3 ขั้นตอน คือ
   - เปิดไฟล์ ถ้าไฟล์ไม่มีอยู่จริง ต้องสร้างขึ้นมา
   - เขียนข้อมูลลงสู่ไฟล์
   - ปิดไฟล์
2. การอ่านข้อมูลจากไฟล์มี 3 ขั้นตอนเช่นกัน คือ
   - เปิดไฟล์ ถ้าไม่สามารถเปิดได้ เช่น ไม่มีไฟล์อยู่จริง จะต้องรับทราบและออก
   - อ่านข้อมูลจากไฟล์
   - ปิดไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 ระบบจัดการฐานข้อมูล SQLite

SQLite เป็นระบบจัดการฐานข้อมูลขนาดเล็ก สามารถใช้งานได้ง่าย มีการจัดการและบำรุงรักษาง่าย ไม่ยุ่งยาก ที่สำคัญมีขนาดเล็ก และทำงานได้รวดเร็ว เหมาะสำหรับทดสอบใช้งานแบบ stand alone (คือ การค้นหาข้อมูลเมต้าเสิร์ช(Meta Search) ด้วย Copernic Agent โปรแกรมที่มาช่วยค้นหาข้อมูลที่ต้องการจากเสิร์ชเอนจิ้นและแหล่งข้อมูลบนเว็บกว่าอีก 90 เว็บไซต์ ด้วยคุณสมบัติที่มีหมวดหมู่ให้เลือกค้นหาจากแหล่งเสิร์ชเอนจิ้นแนวหน้า ถ้าข้อมูลที่ได้ซ้ำกัน ก็จะแสดงผลเพียงรายการเดียว มีการตรวจสอบและลบลิงก์ที่ไม่สมบูรณ์ทิ้ง สามารถเน้นสีลงบนข้อความที่ต้องการค้นหาและเว็บเพจ บันทึกเพจในแบบออฟไลน์บราวเซอร์ และค้นหาคีย์เวิร์ดจากผลลัพธ์ที่ได้ หรือจากเว็บเพจด้วยรูปแบบของบูลีนอีกด้วย) พัฒนาด้วย C จากการทดสอบประสิทธิภาพ กับ PostgreSQL, MySQL, FirebirdSQL ค่อนข้างน่าพอใจ ถึงแม้ว่า update, delete, insert, drop จะสู้ไม่ได้ แต่มีความเร็วในการ select สูงกว่ามาก (เร็วกว่า MySQL 2-3 เท่า)

SQLite มีให้ใช้บน Linux และ Windows แต่ก็ถูก port มาให้ใช้บน platform อื่นๆ ด้วย เช่น Win CE, Nokia

ตัวฐานข้อมูลของ SQLite เอง ให้เปรียบเป็น file เล็กๆ ไฟล์หนึ่งที่เราสามารถแนบติดกับโปรแกรมของเราใช้งานได้สะดวก โดยที่เราไม่จำเป็นที่จะต้องติดตั้ง MySQL บนคอมพิวเตอร์ของผู้ใช้ จึงสะดวกแก่การนำมาใช้งาน

2.6.1 ข้อดีของ SQLite

1. SQLite มีการใช้หน่วยความจำน้อย และเพียงแค่ใช้ library เล็กๆ ก็สามารถเข้าถึงฐานข้อมูล นำติดตั้งไปพร้อมกับ application (ส่วนติดต่อกับผู้ใช้) ที่พัฒนาได้ทันที

2. SQLite สามารถนำไปใช้หลาย platform และยังสามารถทำงานได้ที่ Windows CE และ Palm OS อีกด้วย

3. SQLite มีขนาดเล็ก สามารถนำไปใช้แทนที่ฐานข้อมูลตัวเก่าได้เลย เพราะมีคุณสมบัติที่เรียกว่า consistency (ลงรอยกับรูปแบบการใช้งานแบบเก่าๆ), แยกเป็นไฟล์เดียวเลยก็ได้, ยืดหยุ่นสำหรับการใช้งาน

4. SQLite สามารถใช้ภาษา SQL สร้างฐานข้อมูลได้ตามปกติ มีคำสั่งเฉพาะบ้างเล็กน้อยในการเรียกดูข้อมูลต่างๆ ของฐานข้อมูล

5. SQLite ไม่ต้องการ Driver พิเศษ สำหรับการเข้าถึง Database เช่นพวก ODBC(Open Database Connectivity) เพียงแค่ include library และ Database file เข้าไปใน application ก็ใช้งานได้ทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 ภาษาโปรแกรม ที่สามารถใช้ SQLite

SQLite สนับสนุนหลายภาษา ส่วนใหญ่จะมี library มาให้แล้ว ภาษา Native language API สำหรับภาษา C/C++, PHP, Perl, Python, Tcl เป็นต้น แต่ก็ยังไม่มี API สำหรับ C# ให้ใช้งาน

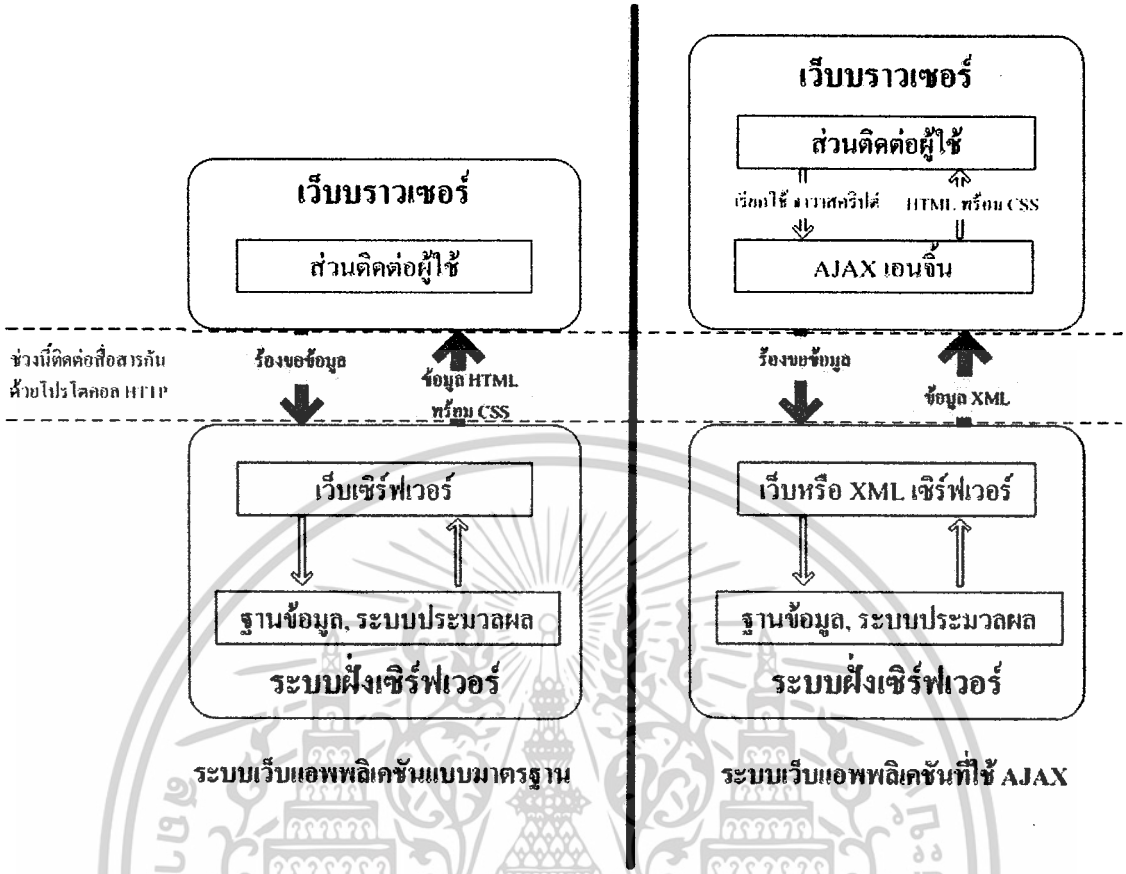
2.6.3 ประโยชน์ของ SQLite

1. ฐานข้อมูล Embeded, แนบฐานข้อมูลเข้าไปในโปรแกรมขนาดเล็ก เช่น PDA, Mobile
2. ทำเป็นฐานข้อมูลสำหรับ demo โปรแกรม โดยไม่จำเป็นต้องใช้เซิร์ฟเวอร์ของฐานข้อมูล

2.7 เทคนิค AJAX ในเว็บแอพลิเคชัน

AJAX(เอแจ็กซ์) ย่อมาจาก Asynchronous Javascript And XML โดยเป็นการทำงานร่วมกันของ Javascript และ XML โดยจะเป็นการทำงานแบบที่ไม่ต้องรอคอย เมื่อบราวเซอร์ (Browser) ร้องขอข้อมูลไปยังเซิร์ฟเวอร์ บราวเซอร์จะไปทำงานคำสั่งถัดไปทันที โดยที่ไม่ต้อง รอการตอบกลับจากเซิร์ฟเวอร์ก่อน ทำให้การตอบสนองต่อผู้ใช้งานดูรวดเร็วขึ้น นอกจากนี้ AJAX สามารถร้องขอข้อมูลจากเซิร์ฟเวอร์โดยที่ไม่จำเป็นต้อง Reload หน้า เพื่อจัดการแสดงผลใหม่ และใช้ Javascript เพื่อควบคุมการแสดงผลเพียงบางส่วนที่เปลี่ยนแปลง ทำให้การแสดงผลรวดเร็วยิ่งขึ้น โดย AJAX สามารถทำงานร่วมกับ Javascript, XML, DHML, CSS และ DOM ได้อีก เพื่อเสริมประสิทธิภาพในการใช้งานให้เป็นระเบียบ และเรียบร้อยขึ้นด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



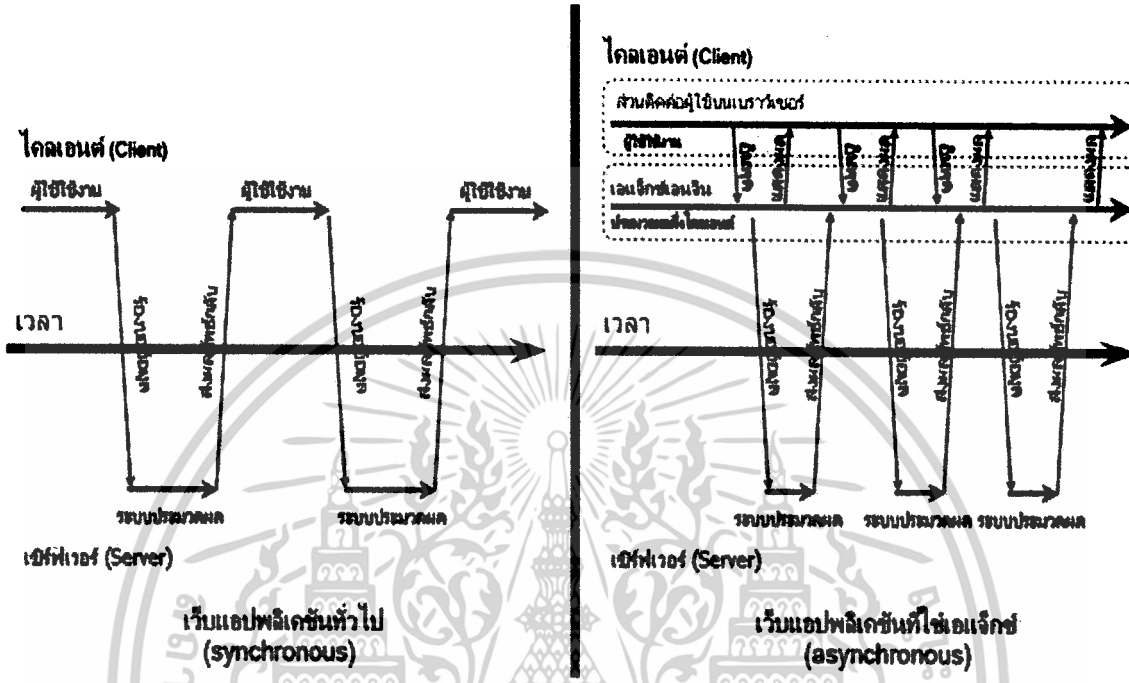
รูปที่ 2.14 เปรียบเทียบการทำงานระหว่างเว็บแอพพลิเคชันแบบเดิมกับแบบที่ใช้AJAX

เว็บเพจทั่วไปการทำงานจะเริ่มต้นเมื่อบราวเซอร์ร้องขอข้อมูลเว็บเพจไปยังเซิร์ฟเวอร์ ซึ่งเซิร์ฟเวอร์จะทำการประมวลผลคำสั่งจนเสร็จ แล้วส่งเอกสารทั้งหน้ากลับมายังบราวเซอร์ เพื่อแสดงผลอีกที จากขั้นตอนต่างๆ จะเห็นได้ว่า เราจะต้องรอคอยให้เซิร์ฟเวอร์ประมวลผลเสร็จ เซิร์ฟเวอร์จึงส่งข้อมูลกลับมาให้เซิร์ฟเวอร์ครบถ้วน จึงจะสามารถแสดงผลได้ ซึ่งถ้าเอกสารมีขนาดใหญ่ ก็จะทำให้การแสดงผลล่าช้า ทำให้ผู้ใช้งานรอนาน

AJAX สามารถแก้ปัญหานี้ โดยการแบ่งข้อมูลออกเป็นกลุ่มย่อยๆตามความจำเป็น ซึ่งเมื่อเราเรียกเพจครั้งแรก บราวเซอร์อาจจะร้องขอข้อมูลเพียงบางส่วนก่อน เช่นอาจเรียกส่วนที่เป็น พื้นหลัง Logo หรือ ตารางหลักต่างๆ ออกมาก่อน ซึ่งจะทำให้ข้อมูลที่เซิร์ฟเวอร์ตอบกลับมา มีขนาดน้อยลง ใช้เวลาโหลดและแสดงผลเร็วขึ้น ในขณะเดียวกัน AJAX จะทำหน้าที่ ร้องขอข้อมูลในส่วนที่เป็นข้อมูลจากเซิร์ฟเวอร์พร้อมๆ กับการจัดการแสดงผลในส่วนแรก(Asynchronous) ซึ่งเมื่อเซิร์ฟเวอร์ประมวลผลเสร็จเรียบร้อย เซิร์ฟเวอร์จะส่งข้อมูลกลับมายังบราวเซอร์อีกที และใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Javascript ในการจัดการแสดงผลในส่วนที่เหลือ จะพบได้ว่าการแสดงผลจะรวดเร็วขึ้นสำหรับการใช้งานของผู้ใช้



รูปที่ 2.15 เปรียบเทียบการติดต่อสื่อสาร ระหว่างเว็บแอพพลิเคชันแบบดั้งเดิมกับแบบที่ใช้ AJAX

นอกจากนี้ จากการที่เราแบ่งการแสดงผลเป็นส่วนๆ ทำให้การโหลดข้อมูลในหน้าต่อๆไป ไม่จำเป็นต้องโหลดข้อมูลทั้งหน้าขึ้นมาอีก เราจะโหลดเฉพาะในส่วนของข้อมูลที่ต้องการเท่านั้นมา ทำให้การโหลดหน้าต่อๆไป รวดเร็วขึ้นมาก เพราะไม่ต้องโหลดใหม่ทั้งหน้า

ข้อดีของการแยกการโหลดออกเป็นส่วนๆ นอกจากที่กล่าวมาแล้วยังมีอีกอย่างหนึ่งคือ การแบ่งการแสดงผลออกเป็นส่วนๆ ทำให้การเปลี่ยนแปลงข้อมูลดูนุ่มนวลขึ้น เพราะสามารถใช้ Javascript ในการควบคุมการแสดงผลเป็นส่วนๆ ได้ลดการโหลด และกระพริบของหน้าจอ และยังสามารถ แสดงผลในแบบเวลาจริงได้โดยการโหลดข้อมูล เฉพาะที่มีการเปลี่ยนแปลงเท่านั้นออกมาแสดง (Auto Refresh)

เหตุผลในการเลือก AJAX ไปใช้งานคือมีขั้นตอนของการ Auto Refresh โดยวิธีการของ Auto refresh คือ การอัปเดทข้อมูลบนเพจโดยอัตโนมัติ ตามระยะเวลาที่กำหนด ซึ่งกำหนดให้เว็บเพจ กลับไปร้องขอข้อมูลจากเซิร์ฟเวอร์ใหม่เป็นระยะๆ เพื่อให้ข้อมูลบนเพจ มีความเป็นปัจจุบันอยู่ตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากผู้ใช้สามารถร้องขอเฉพาะข้อมูลดิบ จากเซิร์ฟเวอร์ได้โดยที่ไม่จำเป็นต้องส่งข้อมูลมาทั้งหน้า แล้วจึงใช้ Javascript ในการแยกข้อมูล รวมถึงการจัดรูปแบบไปจนถึงการแสดงผล (อาจใช้ข้อมูลชนิด XML หรือ DOM หรือข้อมูลอื่นใดในส่วนนี้ก็ได้) จากการที่ข้อมูลมีขนาดที่เล็กลง ทำให้เวลาในการส่งข้อมูลน้อยลงตามไปด้วย ส่งผลดีต่อเซิร์ฟเวอร์ โดยเป็นการลดภาระของเซิร์ฟเวอร์ลงและทำให้ระบบเร็วขึ้น

2.7.1 XML

XML ย่อมาจาก Extensible Markup Language เป็นภาษาที่ให้ความชัดเจนในการให้รายละเอียดเกี่ยวกับข้อมูล และการเปลี่ยนแปลงข้อมูลโดยแอพพลิเคชันบนเว็บและใช้ฟอร์มที่ยืดหยุ่นได้ตามมาตรฐาน HTML หรือ Hyper Text Markup Language ซึ่งเป็นการแสดงข้อมูลต่างๆ มานำเสนอ ส่วน XML จะทำให้การทำงานกับข้อมูลโดยตรงที่เสริมกับการทำงานของ HTML โดย XML จะมีความพร้อมในแง่ของรายละเอียด และการนำข้อมูลตลอดจนโครงสร้างข้อมูลมาแสดงในรูปแบบ Text ผ่านทาง HTTP(HypertText Markup Language และ HyperText Transport Protocol) ที่เปิดให้ข้อมูลขึ้นใหม่และมีความสามารถในการจัดข้อมูลได้อีกด้วย ในการเขียนเว็บเพจเมื่อใช้ HTML ผู้พัฒนาสามารถกำหนดได้ว่าส่วนไหนจะเป็นตัวหนา ตัวเอียง หรือตัวอักษรเป็นแบบไหน ส่วน XML นั้นจะเป็นการเตรียมส่วนของข้อมูลที่จะนำไปใส่ในช่องที่กำหนดตามการเขียนของ HTML โดยอาจจะกล่าวได้ว่า XML ถูกออกแบบมาเพื่อใช้ อธิบายข้อมูล และสนใจแต่เพียงว่าข้อมูลคืออะไร แต่ HTML ถูกออกแบบมาเพื่อใช้ แสดงผลข้อมูล และสนใจแต่เพียงว่า ข้อมูลจะแสดงอย่างไร

2.7.2 Javascript

JavaScript เป็นภาษาสคริปต์เชิงวัตถุ หรือเรียกว่า อ็อบเจ็กโอเรียลเต็ด (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนเอกสารด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษาจาวาได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server) โดยมีลักษณะการทำงานดังนี้

1. Navigator JavaScript เป็น Client-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งไคลเอนต์ (หมายถึงฝั่งเครื่องคอมพิวเตอร์ของผู้ใช้ ไม่ว่าจะเป็นเครื่องพีซี เครื่องแมคอินทอช หรือ อื่น ๆ) จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไปเป็นส่วนใหญ่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. LiveWire JavaScript เป็น Server-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งเซิร์ฟเวอร์ (หมายถึงฝั่งเครื่อง คอมพิวเตอร์ของผู้ให้บริการเวบ โดยอาจจะเป็นเครื่องของซัน ซิลิคอมกราฟิกส์ หรือ อื่น ๆ) สามารถใช้ได้เฉพาะกับ LiveWire ของเน็ตสเคป โดยตรง

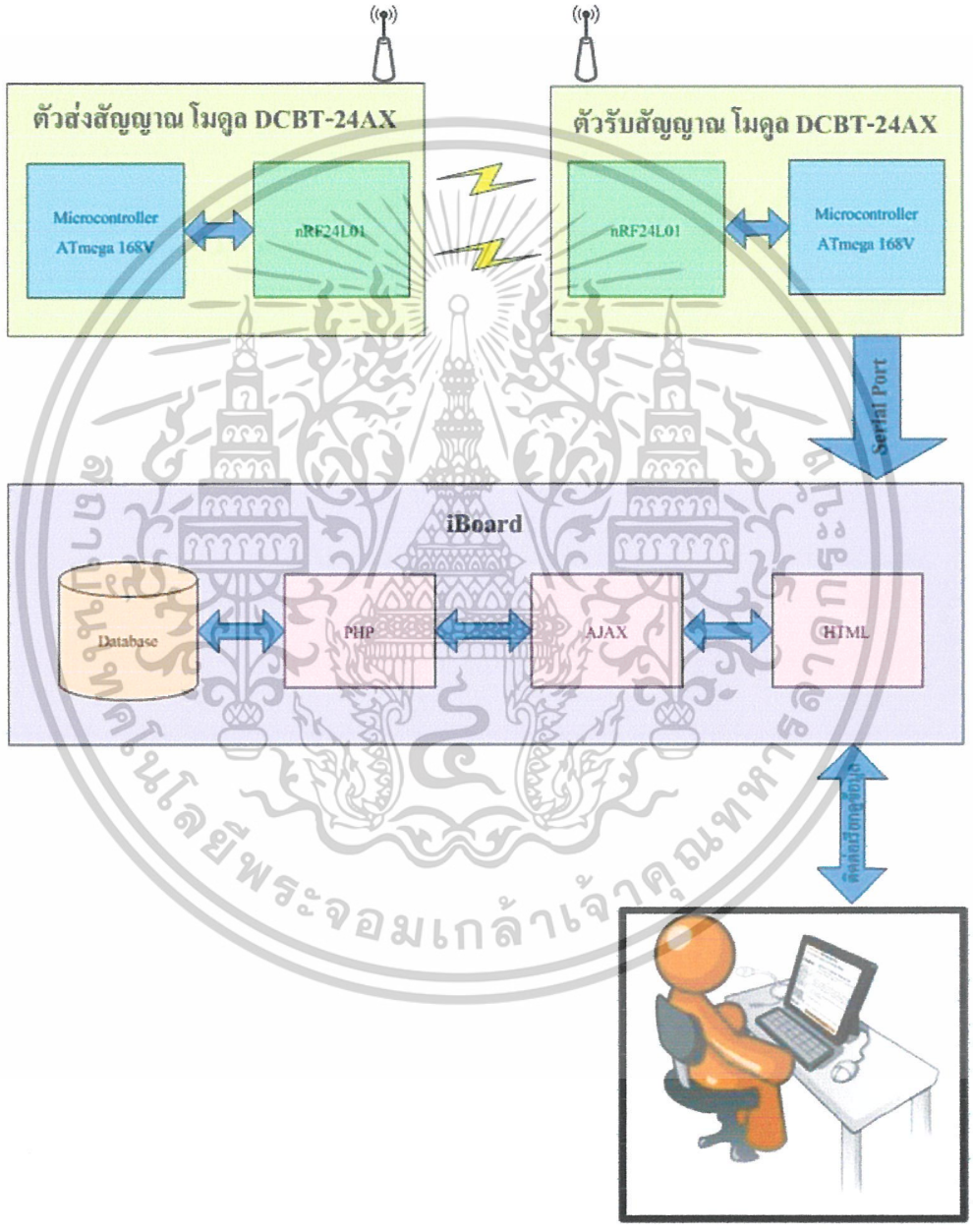
สามารถเขียนโปรแกรม JavaScript เพิ่มเข้าไปในเว็บเพจเพื่อใช้ประโยชน์สำหรับงานด้านต่าง ๆ ทั้งการคำนวณ การแสดงผล การรับ-ส่งข้อมูล และที่สำคัญคือ สามารถโต้ตอบกับผู้ใช้ได้อย่างทันทีทันใด ตัวอย่างของงาน ที่จะต้องใช้ภาษา JavaScript เช่น

- สร้างเมนูที่สามารถตอบสนองต่อการเอา mouse ไปชี้ได้
- สร้างเครื่องคิดเลข ปฏิทิน เกม ในเวบเพจ
- เปิด, ปิด, เคลื่อนย้ายตำแหน่งของ window
- ทำตัวอักษรวิ่งแบบต่างๆ
- ตรวจสอบความถูกต้องของแบบฟอร์มที่ผู้ใช้กรอก ก่อนส่งข้อมูลไปยัง Server
- และอื่นๆ อีกมากมายที่ไม่ต้องติดต่อกับ Server ซึ่ง JavaScript เขียนโปรแกรมทางฝั่ง cliend โดยมีบราวเซอร์เป็นตัวแปลภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ

3.1 หลักการทำงานของระบบโดยรวม



รูปที่ 3.1 หลักการทำงานของระบบโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

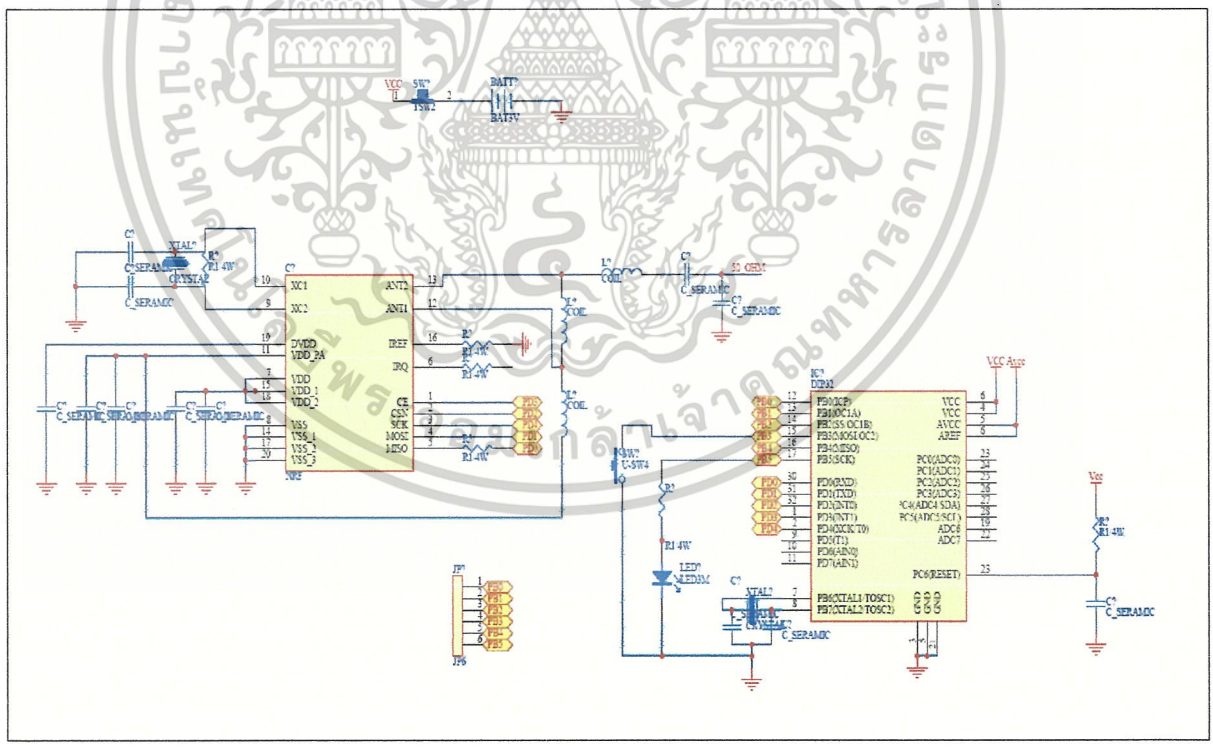
การทำงานของระบบโดยรวมจะแบ่งเป็นส่วนใหญ่ๆ ได้ 3 ส่วน คือ ส่วนการทำงานของภาคส่งสัญญาณ, ส่วนการทำงานของภาครับสัญญาณ และส่วนการทำงานของ iBoard ในส่วนสุดท้ายนี้จะทำหน้าที่เก็บข้อมูลและแสดงข้อมูลผ่านทางเว็บแอพพลิเคชัน

3.2 ส่วนการทำงานของภาคส่งสัญญาณ

การส่งสัญญาณของภาคส่งจะอยู่ในรูปแบบสัญญาณไร้สาย โดยใช้ชุดโมดูล DCBT-24AX ที่ภายในชุดโมดูลประกอบด้วย ไมโครคอนโทรลเลอร์ AVR รุ่น ATmega168V และชิพรับ-ส่งสัญญาณ nRF24L01 มีเสาอากาศในตัว สำหรับรับ-ส่งสัญญาณไร้สายในย่านความถี่ 2.4 GHz

3.2.1 วงจรภายในภาคส่งสัญญาณ ชุดโมดูล DCBT-24AX

วงจรของภาคส่งสัญญาณจะประกอบด้วย วงจรการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์ ATmega168V กับชิพ nRF24L01 และการต่อเพิ่มอุปกรณ์ ที่ประกอบด้วย หลอดไฟ LED 1 ดวง, สวิตช์ส่งสัญญาณฉุกเฉิน(SOS) 1 อัน, จัมพ์เปอร์(Jumper) สำหรับปรับระดับแรงดันในการส่งและแหล่งจ่ายไฟ 3 V จะได้รูปวงจรของอุปกรณ์ภาคส่งสัญญาณดังรูปที่ 3.2 นี้



รูปที่ 3.2 วงจรภายในโมดูล DCBT-24AX ของภาคส่งสัญญาณ

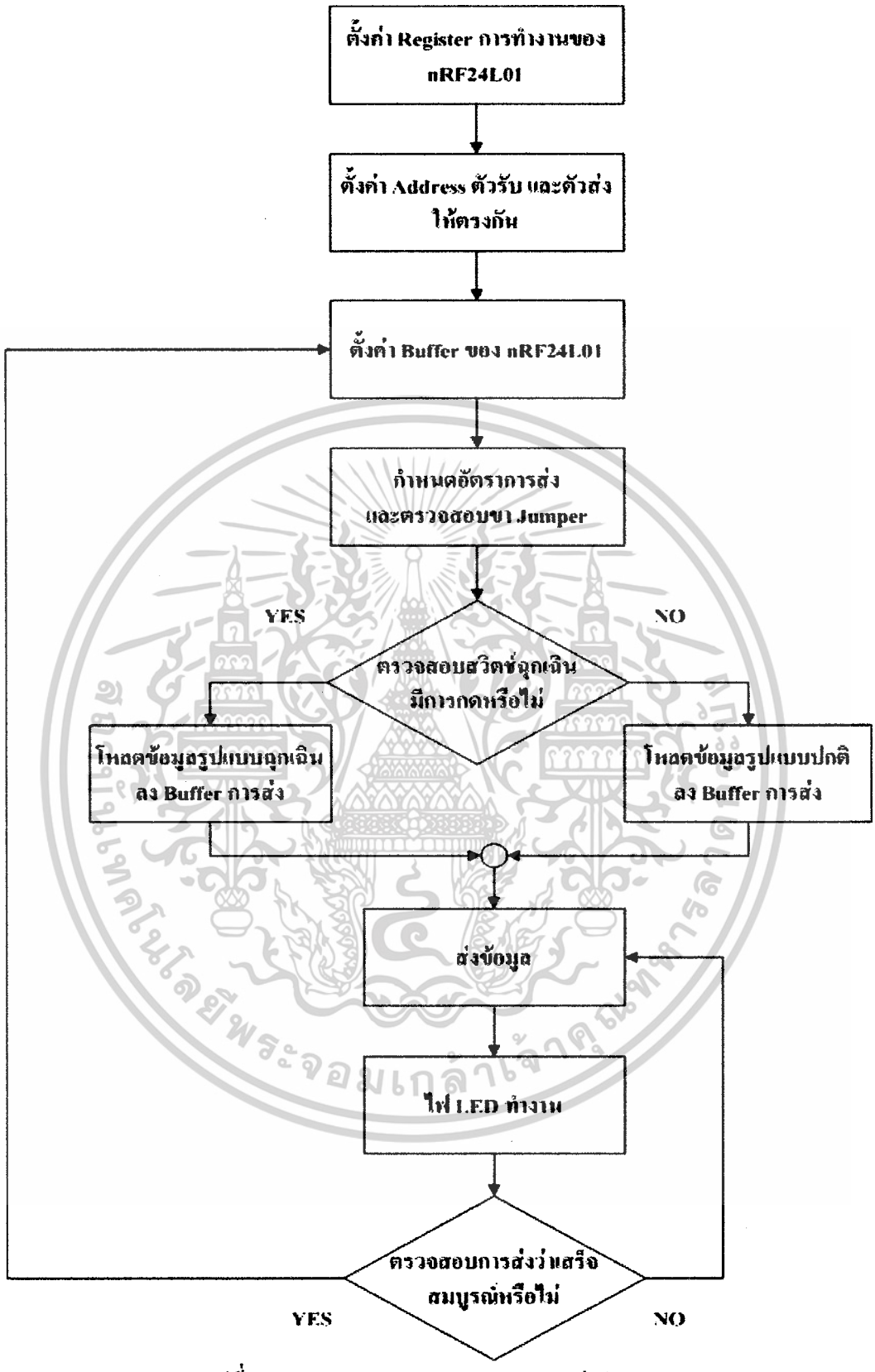
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การทำงานของ DCBT-24AX ในการส่งข้อมูล

เมื่อทำการเปิดสวิตช์อุปกรณ์ภาคส่งสัญญาณ โมดูล DCBT-24AX ของภาคส่งจะเริ่มทำการตั้งค่าอุปกรณ์ชิพ nRF24L01 โดยมี ไมโครคอนโทรลเลอร์ ATmega168V เป็นตัวควบคุมการตั้งค่าการทำงานต่างๆ รวมถึงกำหนดข้อมูลที่จะส่งออกไปยังภาครับสัญญาณด้วย ซึ่งในภาคส่งสัญญาณนี้จะมีหลักการทำงานดังนี้

1. ไมโครคอนโทรลเลอร์ ATmega168V จะควบคุมการตั้งค่าการทำงานของชิพ nRF24L01 สั่งเปิดการทำงาน และจ่ายกำลังงานให้ โดยกำหนดให้เป็นโหมดการส่งสัญญาณ พร้อมทั้งตรวจสอบ Register
2. ตั้งค่า Address ตัวส่งให้ตรงกับค่า Address ตัวรับ
3. ตั้งค่า Buffer ของ nRF24L01
4. กำหนดอัตราการส่ง (สามารถเลือกได้ 2 ระดับ คือ 1 Mb/s หรือ 2 Mb/s ในการทดลองของปริญญานิพนธ์นี้จะกำหนดให้อัตราการส่งมีค่า 2 Mb/s) และตรวจสอบการต่อขาการทำงานของ Jumper เพื่อกำหนดกำลังการส่ง (ปรับเลือกได้ 4 ระดับ คือ -18 dBm, -12 dBm, -6 dBm และ 0 dBm ขึ้นอยู่กับการตั้งค่าของผู้ใช้งาน)
5. ตรวจสอบการกดปุ่มสวิตช์ส่งสัญญาณฉุกเฉิน
   - หากมีการกดปุ่มสวิตช์ส่งสัญญาณฉุกเฉิน จะเขียนข้อมูลรูปแบบฉุกเฉินที่ได้กำหนดไว้ลง Buffer เพื่อเตรียมสำหรับส่งต่อไป
   - หากไม่มีการกดปุ่มสวิตช์ส่งสัญญาณฉุกเฉิน จะเขียนข้อมูลรูปแบบปกติที่ได้กำหนดไว้ลง Buffer เพื่อเตรียมสำหรับส่งต่อไป
6. ส่งข้อมูลถูกออกจากชิพ nRF24L01 ในรูปแบบการสื่อสารไร้สาย ด้วยความถี่ 2.4 GHz
7. หลอดไฟ LED ที่ติดอยู่บนอุปกรณ์ภาคส่งสัญญาณจะทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 Flow Chart การทำงานของภาคส่งสัญญาณ

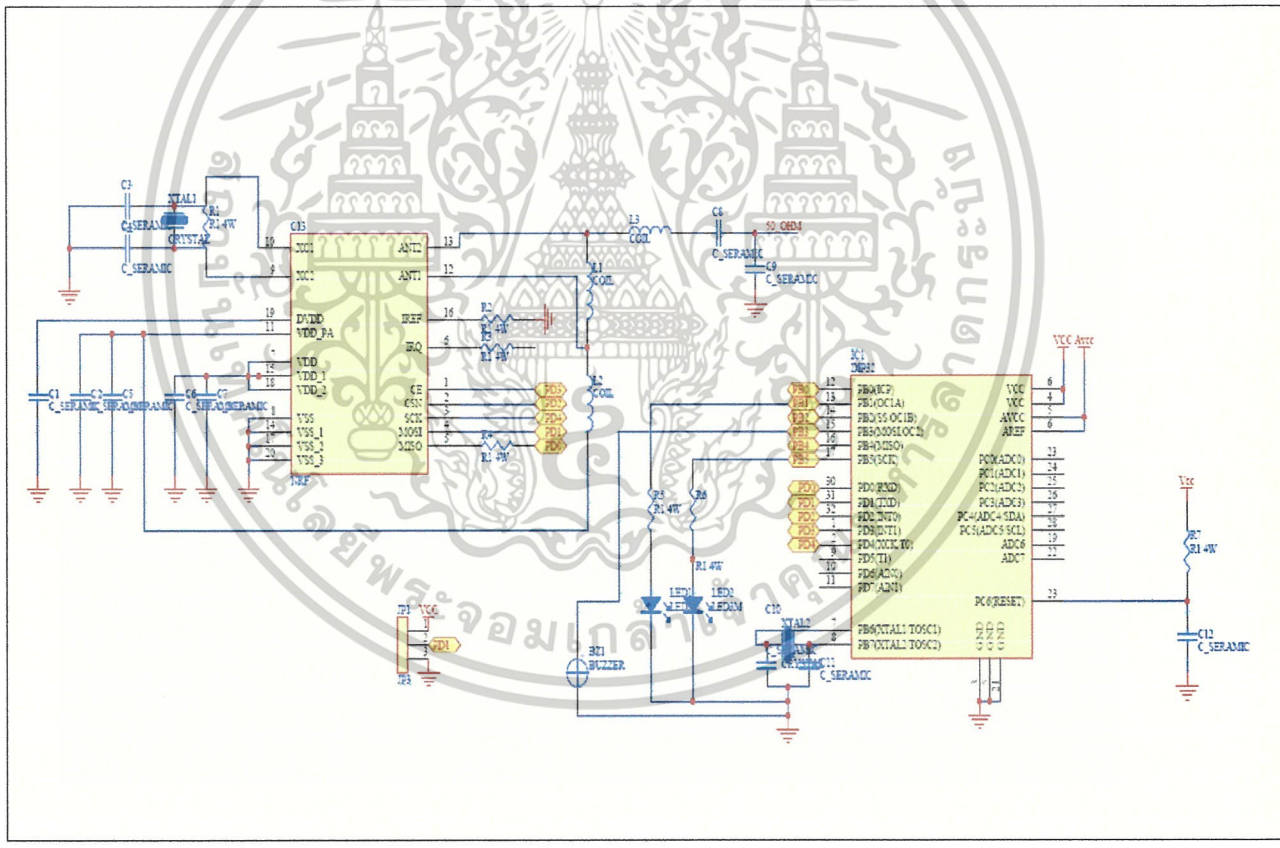
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนการทำงานของภาครับสัญญาณ

สัญญาณที่ภาครับได้รับจะอยู่ในรูปแบบสัญญาณไร้สาย โดยใช้ชุดโมดูล DCBT-24AX เหมือนกับภาคส่งสัญญาณ

3.3.1 วงจรภายในภาครับสัญญาณ ชุดโมดูล DCBT-24AX

วงจรของภาครับสัญญาณจะประกอบด้วย 2 ส่วน ส่วนที่หนึ่งวงจรการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์ ATmega168V กับชิพ nRF24L01 และส่วนที่สองการเชื่อมต่อระหว่างชุดโมดูล DCBT-24AX กับ iBoard โดยผ่านสายพอร์ตอนุกรม และบนอุปกรณ์ภาครับได้ต่อเพิ่มอุปกรณ์ ประกอบด้วย หลอดไฟ LED 2 ดวง และลำโพง Piezo สำหรับการแจ้งเตือน เพื่อแสดงสถานะการออกนอกพื้นที่หรือการแจ้งเหตุฉุกเฉินของอุปกรณ์ภาคส่งแต่ละตัว, สวิตช์สำหรับรีเซ็ตการทำงานของอุปกรณ์ภาครับ โดยจะได้รูปวงจรของอุปกรณ์ภาครับสัญญาณดังรูปที่ 3.4



รูปที่ 3.4 วงจรภายในโมดูล DCBT-24AX ของภาครับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การทำงานของ DCBT-24AX ในการรับข้อมูล

เมื่อ iBoard ถูกเปิดใช้งาน จะจ่ายไฟให้แก่อุปกรณ์ภาคส่งสัญญาณชุดโมดูล DCBT-24AX ผ่านทางสายพอร์ตอนุกรม 3.3 V ทำให้อุปกรณ์ภาคส่งสัญญาณเริ่มการทำงาน ซึ่งมีหลักการทำงานดังนี้

1. ไมโครคอนโทรลเลอร์ ATmega168V จะควบคุมการตั้งค่าการทำงานของชิพ nRF24L01 สั่งเปิดการทำงาน โดยกำหนดให้เป็นโหมดการรับสัญญาณ พร้อมทั้งตรวจสอบ Register
2. ตั้งค่า Address ตัวรับให้ตรงกับค่า Address ตัวส่ง
3. ตรวจสอบการอินเตอร์รัพท์ของชิพ nRF24L01 ว่ามีการเกิดขึ้นหรือไม่

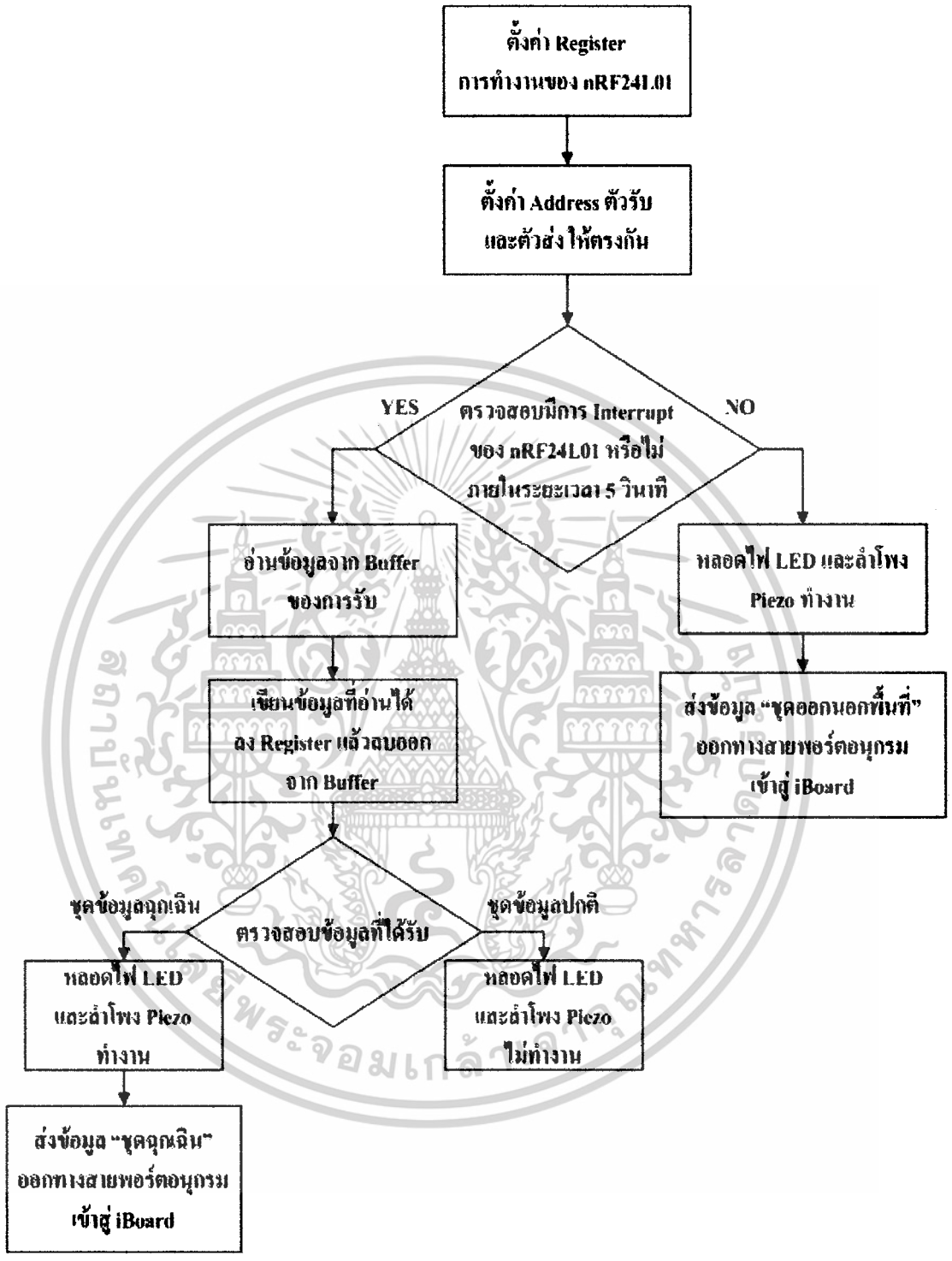
   3.1 กรณีเกิดการอินเตอร์รัพท์ แสดงว่ามีข้อมูลเข้ามา
   - จะอ่านค่าข้อมูลจาก Buffer ของการรับ
   - นำข้อมูลภายใน Buffer การรับ มาเขียนลง Register พร้อมทั้งลบออกจาก Buffer
   - ตรวจสอบข้อมูลที่ได้รับ

   3.1.1 หากเป็นข้อมูลรูปแบบฉุกเฉินที่ได้กำหนดไว้ หลอดไฟ LED และ ลำโพง piezo จะทำงาน และทำการส่งข้อมูลชุด “ฉุกเฉิน”ไปเก็บและแสดงผลยัง iBoard ผ่านทางสายพอร์ตอนุกรม

   3.1.2 หากเป็นข้อมูลรูปแบบปกติที่ได้กำหนดไว้ หลอดไฟ LED และ ลำโพง piezo จะไม่ทำงาน และไม่มีการส่งข้อมูลต่อไปยัง iBoard

   3.2 กรณีไม่เกิดการอินเตอร์รัพท์ แสดงว่าไม่มีการรับข้อมูลเข้ามา ระบบจะทำการตรวจสอบวนซ้ำในช่วงระยะเวลา 5 วินาที หากในช่วงระยะเวลานี้ยังไม่มีข้อมูลถูกส่งเข้ามา หลอดไฟ LED และ ลำโพง piezo จะทำงาน และทำการส่งข้อมูลชุด “ออกนอกสถานที่”ไปเก็บและแสดงผลยัง iBoard ผ่านทางสายพอร์ตอนุกรม
4. อุปกรณ์ภาครับสัญญาณจะทำงานวนซ้ำขั้นตอน 3 ไปเรื่อยๆ จนกว่าจะมีการกดปุ่มรีเซ็ต เพื่อเริ่มทำงานใหม่อีกครั้ง หรือไม่มีการจ่ายไฟจาก iBoard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 Flow Chart การทำงานของภาครับสัญญาณ

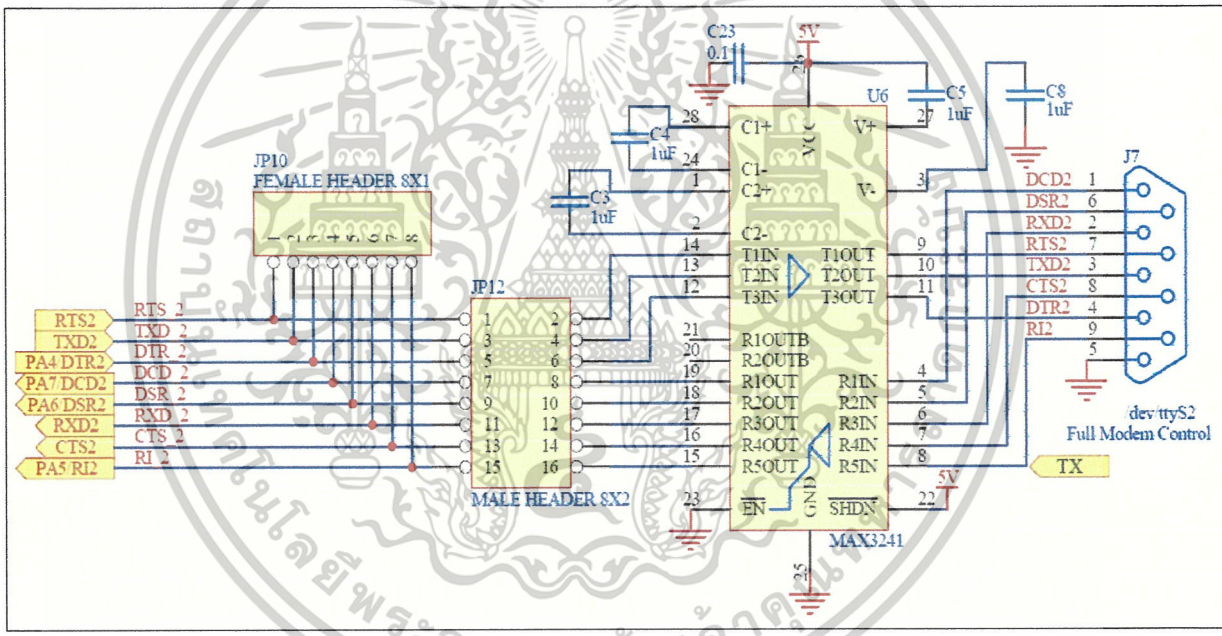
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ส่วนการทำงานของ iBoard

ภายในส่วนนี้จะเป็นการทำงานโดยรับข้อมูลจากไมโครคอนโทรลเลอร์ ATmega168V ภายในโมดูล DCBT-24AX ของอุปกรณ์ภาครับสัญญาณ ที่จะส่งข้อมูลมาให้ผ่านทางสายพอร์ตอนุกรม เมื่อได้รับข้อมูลแล้ว iBoard จะทำการเช็คแพคเกจข้อมูล แล้วนำข้อมูลที่ได้นั้นไปเก็บลงฐานข้อมูลระบบ ซึ่งหากผู้ใช้งานต้องการเรียกดูข้อมูลจะต้องผ่านเข้าสู่ระบบทางเว็บเซิร์ฟเวอร์

3.4.1 วงจรการเชื่อมต่อระหว่างอุปกรณ์ภาคส่งสัญญาณและ iBoard ผ่านทางสายพอร์ตอนุกรม

จากที่ได้กล่าวไว้ว่า iBoard ทำการรับข้อมูลจากไมโครคอนโทรลเลอร์ ATmega168V ภายในโมดูล DCBT-24AX ของอุปกรณ์ภาครับสัญญาณผ่านทางสายพอร์ตอนุกรม ซึ่งจะมีวงจรการเชื่อต่อของอุปกรณ์ทั้ง 2 ตัว ดังรูปที่ 3.6



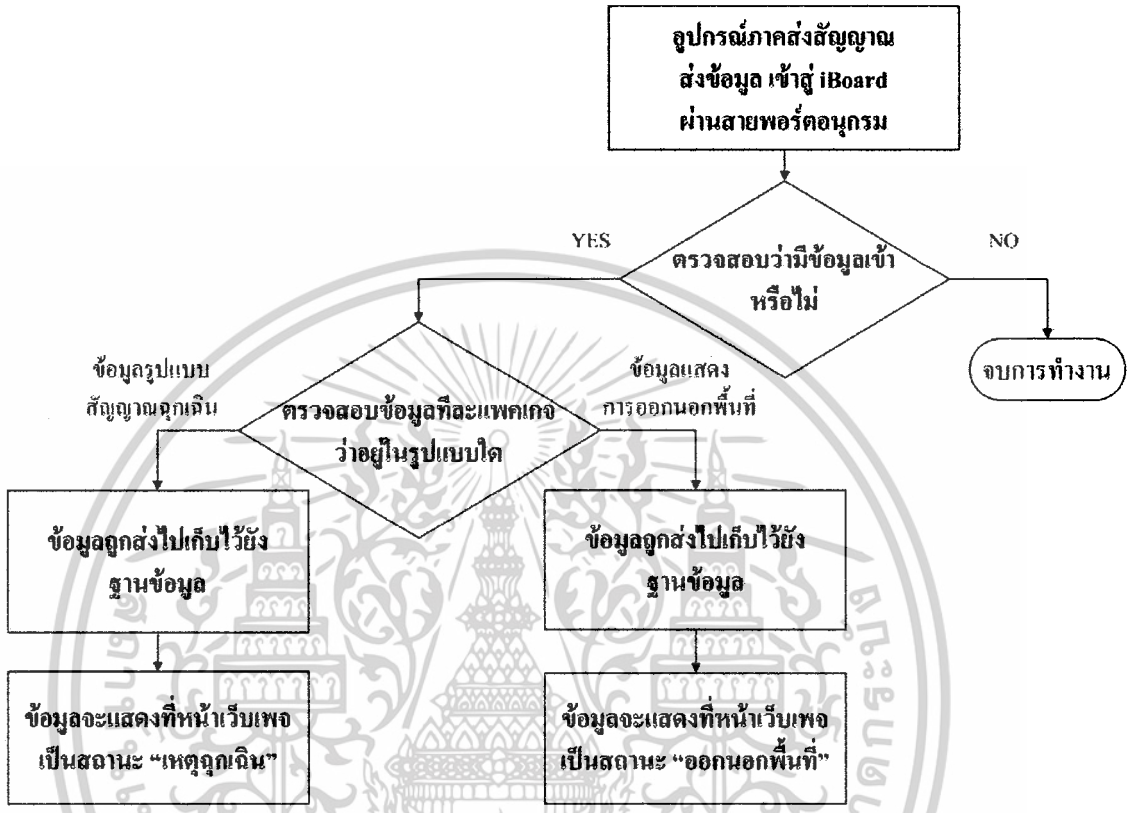
รูปที่ 3.6 วงจรการเชื่อมต่อระหว่างอุปกรณ์ภาคส่งสัญญาณและ iBoard

3.4.2 หลักการตรวจสอบข้อมูลที่รับเข้ามา เพื่อเก็บในฐานข้อมูล

เมื่อ iBoard สามารถรับข้อมูลที่ผ่านเข้ามาทางสายพอร์ตอนุกรมแล้วนั้น จะเข้าสู่ขั้นตอนการตรวจสอบข้อมูล เพื่อแยกข้อมูลที่จะนำไปเก็บลงสู่ฐานข้อมูลของระบบ โดยฐานข้อมูลจะทำการเก็บเฉพาะช่วงเวลาที่มีการส่งสัญญาณแจ้งเตือนฉุกเฉิน หรืออุปกรณ์ภาครับสัญญาณไม่ได้ข้อมูลภายในเวลา 5 วินาที เพื่อให้ระบบฐานข้อมูลไม่สิ้นเปลืองเนื้อที่ และสามารถเรียกข้อมูลมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ได้โดยง่าย ระบบฐานข้อมูลของ iBoard จะใช้โปรแกรมการจัดการ SQLite เพื่อให้ฐานข้อมูลมีขนาดที่เล็ก และใช้ง่ายได้สะดวก รวดเร็ว



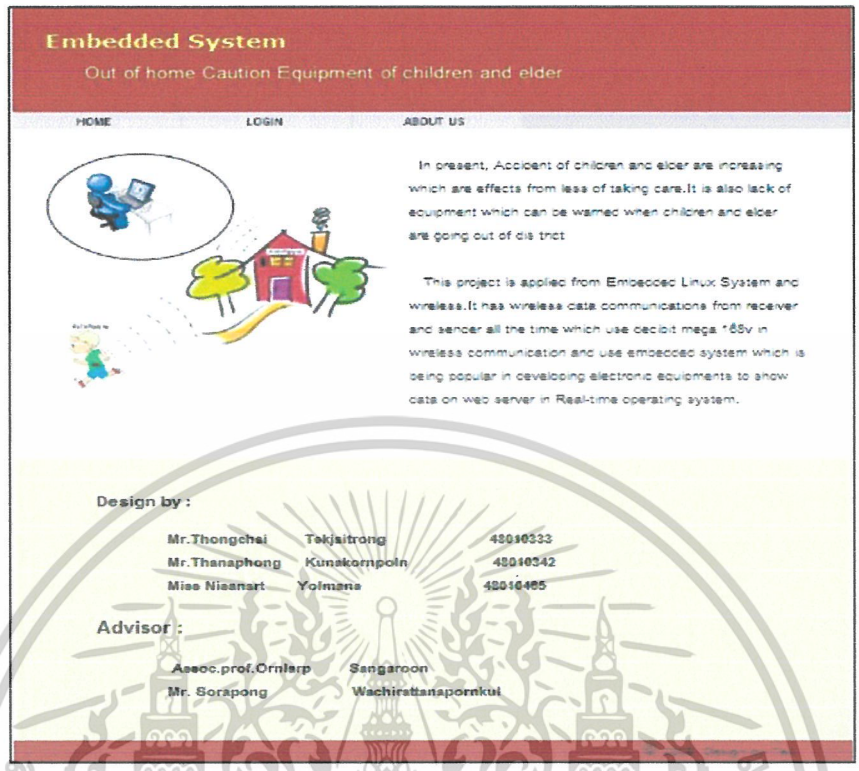
รูปที่ 3.7 Flow Chart การเก็บข้อมูลลงฐานข้อมูลใน iBoard

3.4.3 การแสดงผลผ่านทางเว็บแอพพลิเคชัน

ขั้นตอนการเรียกดูข้อมูลทางหน้าเว็บแอพพลิเคชันของผู้ใช้งาน จะมีขั้นตอนดังนี้

1. ผู้ใช้งานเข้าสู่หน้าโฮมเพจ จากหน้าเว็บเพจนี้ผู้ใช้งานจะสามารถเลือกทำรายการอื่น Login หรือ เรียกข้อมูลแสดงข้อมูลของผู้สร้างและพัฒนาระบบอุปกรณ์แจ้งเตือนการออกนอกพื้นที่ของเด็กและคนชราได้(About us)

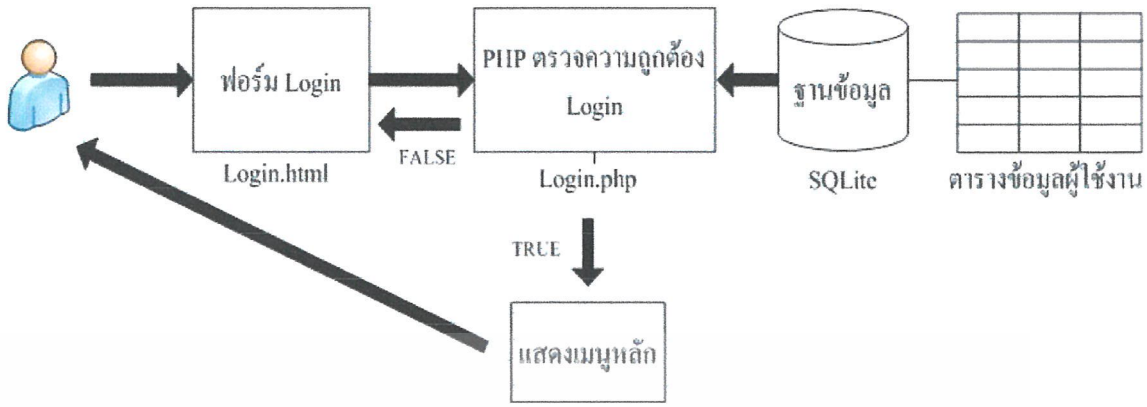
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



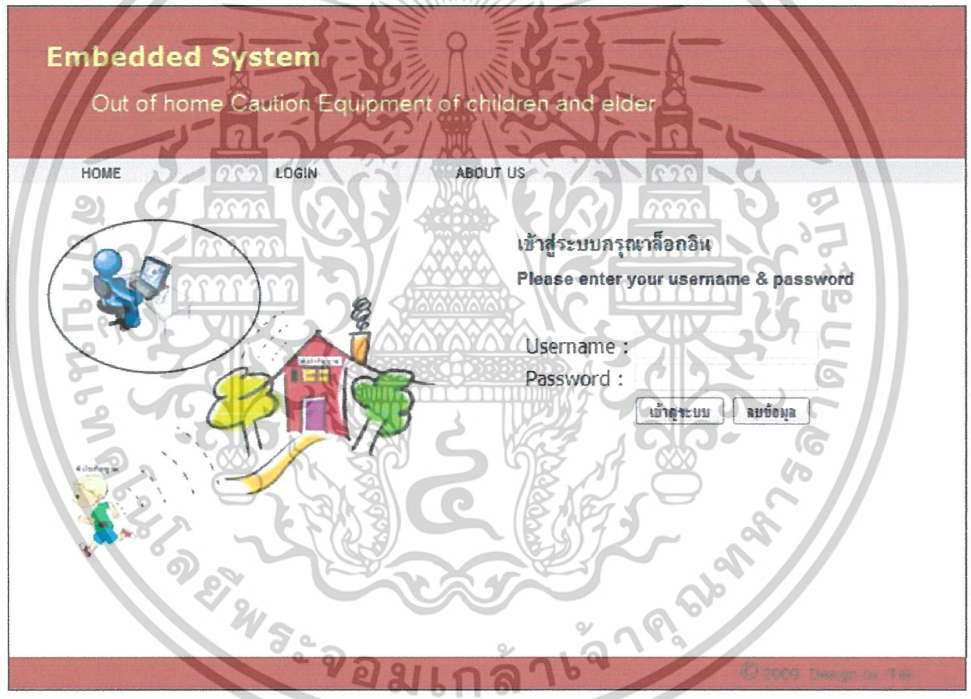
รูปที่ 3.8 หน้าต่างโฮมเพจ

2. ผู้ใช้งานสามารถเรียกดูข้อมูลสถานะโดยเลือกรายการ Login จากหน้าโฮมเพจ เพื่อกรอกชื่อและรหัสผ่านที่ตั้งไว้ จากนั้นระบบจะทำการตรวจสอบความถูกต้องจากฐานข้อมูลที่ได้กำหนดชื่อและรหัสผ่านไว้ก่อนแล้ว
3. ถ้าผู้ใช้งานกรอกชื่อและรหัสผ่านถูกต้องจะสามารถเข้าสู่หน้าเมนูหลัก เพื่อเรียกดูข้อมูลต่อไป แต่หากชื่อหรือรหัสผ่านไม่ถูกต้อง ระบบจะไม่ให้เข้าสู่ห้องเมนูหลัก ผู้ใช้งานต้องกรอกชื่อและรหัสผ่านใหม่อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 Diagram การ Login เข้าสู่การใช้งานเรียกดูข้อมูล

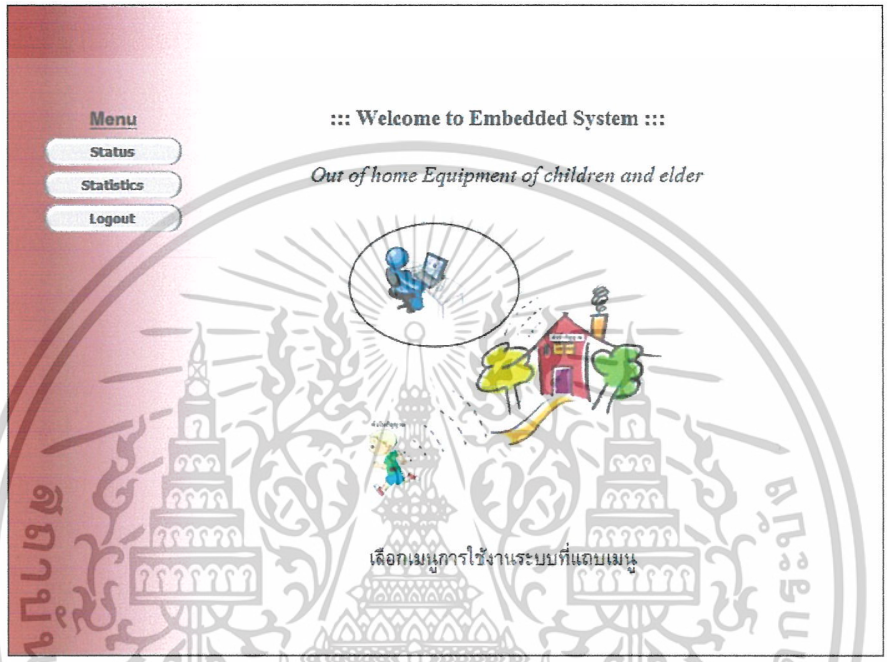


รูปที่ 3.10 หน้าต่างเว็บเพจ Login

4. เมื่อเข้าสู่หน้าเมนูหลักแล้ว ผู้ใช้งานสามารถเลือกทำรายการได้ 3 รูปแบบ คือ
   - Status การแสดงสถานะการออกนอกพื้นที่ ในรูปแบบเวลาจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Statistics การเรียกดูข้อมูลย้อนหลังการออกนอกพื้นที่ โดยระบุจากวัน/เดือน/ปี ที่ต้องการทราบ ข้อมูลเด็กและคนชราที่ออกนอกพื้นที่ หรือแจ้งเหตุฉุกเฉินจะถูกแสดงขึ้น พร้อมกับเวลาในช่วงนั้น
- Logout ออกจากการทำงานของบราวเซอร์

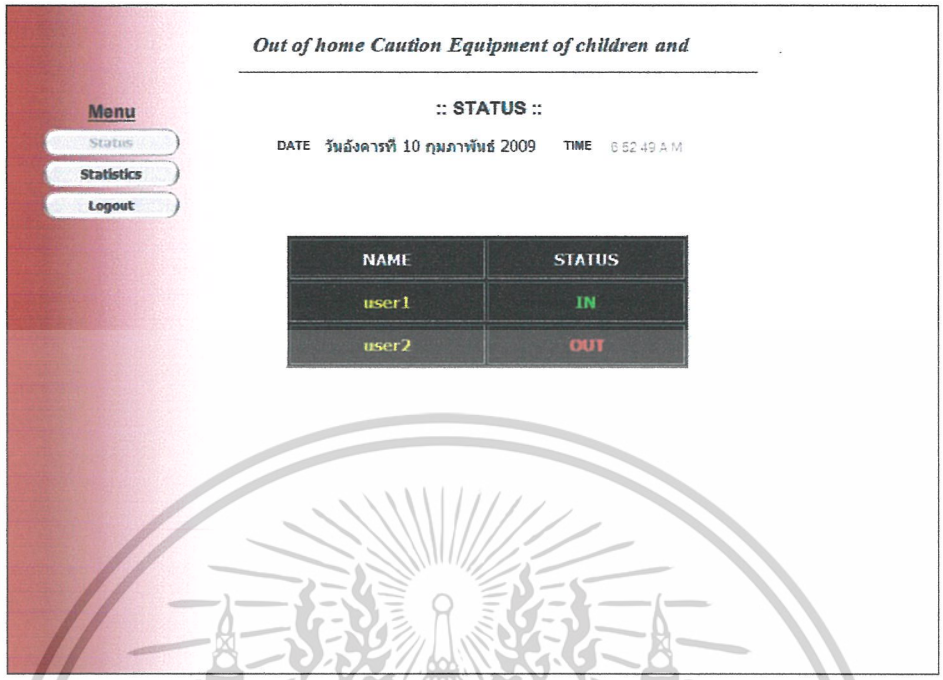


รูปที่ 3.11 หน้าต่างเว็บเพจหน้าเมนูหลัก

1. การแสดงสถานะในรูปแบบเวลาจริง

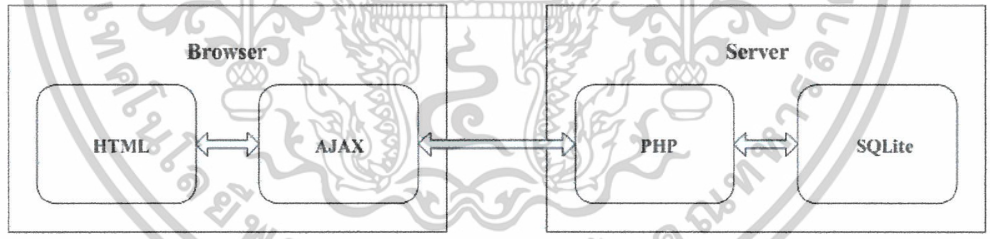
การแสดงผลในหน้าเว็บเพจนี้จะแสดงข้อมูลสถานะการออกนอกพื้นที่ หรือการส่งสัญญาณฉุกเฉินของเด็กและคนชราที่พกพาอุปกรณ์ส่งสัญญาณ ณ เวลาขณะนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 หน้าต่างเว็บเพจหน้า Status

รูปแบบการทำงาน



รูปที่ 3.13 หลักการทำงานของเซิร์ฟเวอร์ในการเรียกใช้ข้อมูลในรูปแบบเวลาจริง

- หน้าเว็บเพจที่ใช้แสดงผลจะทำงานด้วยโปรแกรมภาษา HTML เมื่อมีการร้องขอข้อมูลจากผู้ใช้งานในหน้าเว็บเพจ เทคนิค AJAX ภายในเว็บแอพพลิเคชั่นจะแบ่งข้อมูลออกเป็นกลุ่มย่อยตามความจำเป็น เพื่อส่งข้อมูลร้องขอในส่วนแรกไปประมวลผลที่ PHP ก่อน เพื่อจะไม่ต้องรอการประมวลผลคำสั่งให้เสร็จสิ้น ทำให้การแสดงผลข้อมูลที่หน้าเว็บเพจรวดเร็วขึ้น

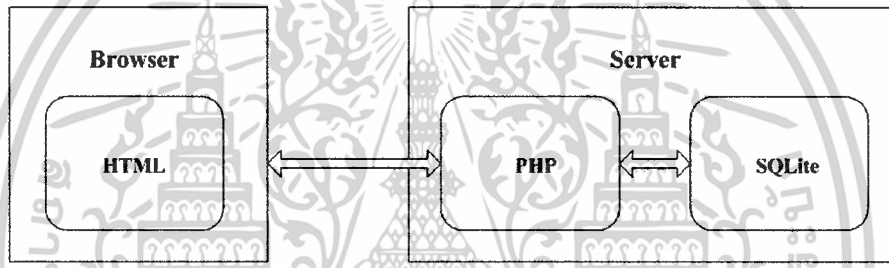
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- PHP ทำการประมวลผลและติดต่อกับฐานข้อมูล โดยการจัดการของ SQLite และทำการส่งข้อมูลที่ถูกร้องขอกลับไปให้บราวเซอร์
- ขณะ AJAX ทำการส่งข้อมูลการร้องขอนั้น จะสามารถรับข้อมูลที่ส่งกลับมาจากเซิร์ฟเวอร์ เพื่อจัดแสดงผลด้วยคำสั่งของ Javascipt ให้เป็นเว็บเพจได้ทันทีด้วย

2. การเรียกดูข้อมูลสถานะย้อนหลัง

ในส่วนนี้จะแสดงข้อมูลสถานะการออกนอกพื้นที่ หรือการแจ้งเหตุฉุกเฉินย้อนหลัง ที่ถูกเก็บไว้ในฐานข้อมูลของระบบ โดยผู้ใช้งานสามารถเลือกการแสดงผลสถานะจากการกำหนดวัน เดือน ปี และเวลาลงไปในช่องที่กำหนดไว้

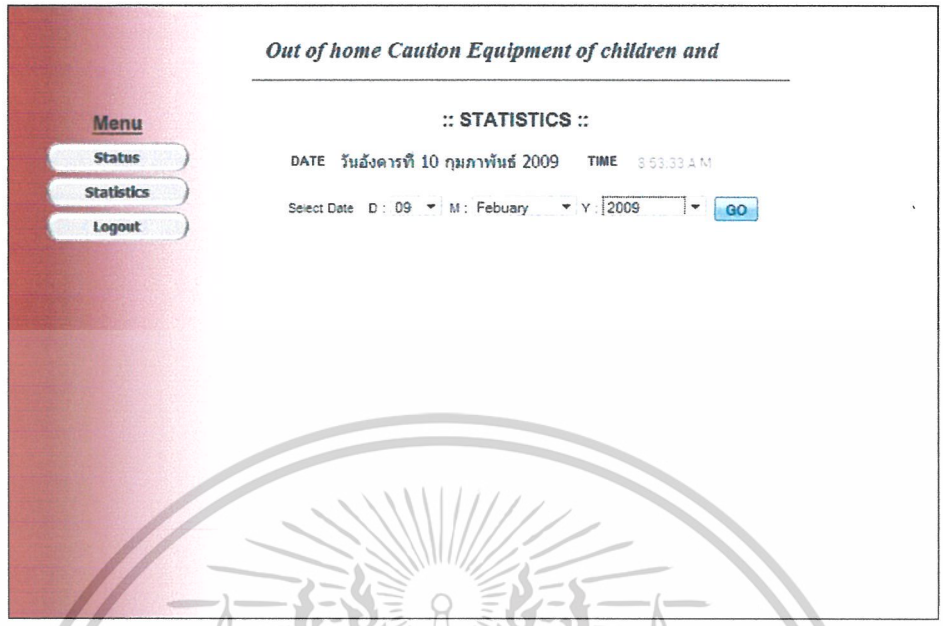
รูปแบบการทำงาน



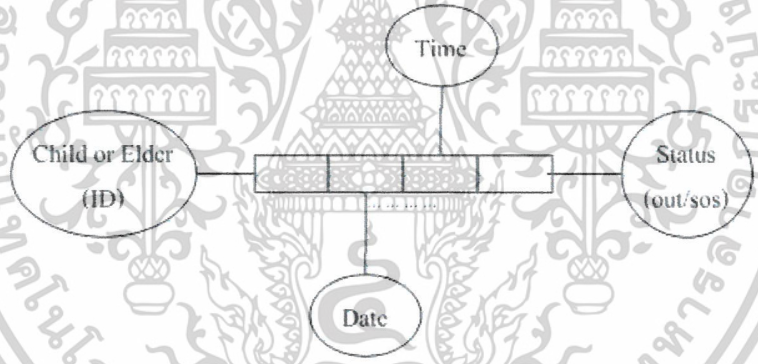
รูปที่ 3.14 หลักการทำงานของเซิร์ฟเวอร์ในการเรียกใช้ข้อมูลย้อนหลัง

จะคล้ายกับการทำงานของการแสดงสถานะในรูปแบบเวลาจริง แต่ไม่ต้องผ่านกระบวนการทำงานของ AJAX ดังรูปที่ 3.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 หน้าต่างเว็บเพจหน้า Statistics



รูปที่ 3.16 NIAM-MODEL ของฐานข้อมูล

ตารางที่ 3.1 การเก็บข้อมูลสถานะการออกนอกพื้นที่หรือการแจ้งเหตุฉุกเฉิน

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
ID Child or Elder (PK)	Text(10)	หมายเลขอุปกรณ์ภาคส่ง
Date	Text(10)	วันที่
Time	Text(10)	เวลา
Status	Text(10)	สถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

ผลการทดลองแบ่งออกเป็น 3 ส่วนคือ

1. อุปกรณ์ชุดโมดูล DCBT-24AX ภาคส่งและภาครับสัญญาณ
2. อุปกรณ์ iBoard ในการรับข้อมูลจากอุปกรณ์ชุดโมดูล DCBT-24AX ภาครับสัญญาณ
3. การทำงานของทั้งระบบ

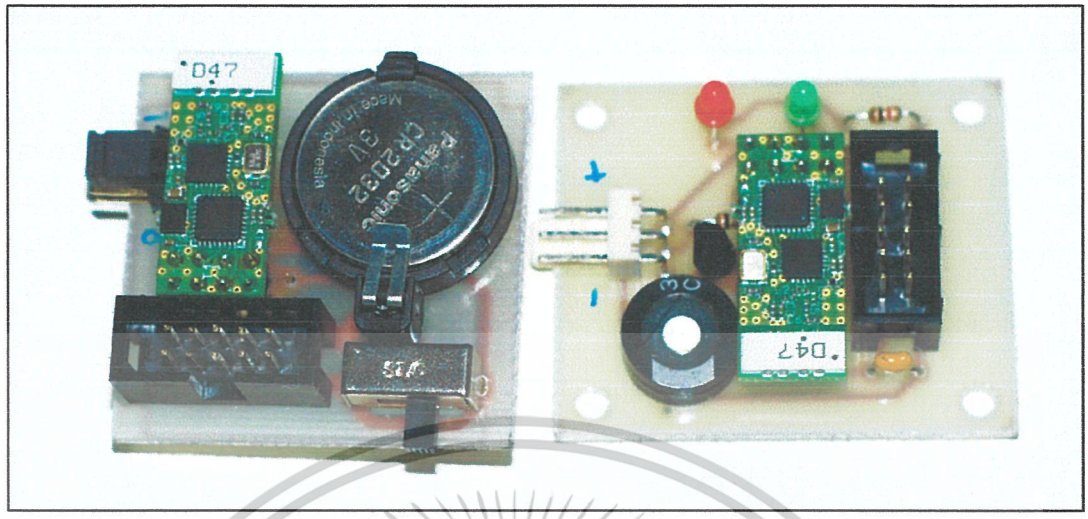
4.1 ผลการทดลองการรับ-ส่งข้อมูลของอุปกรณ์ภาครับและภาคส่งสัญญาณไร้สาย

เมื่อเปิดใช้งานอุปกรณ์ภาครับและภาคส่งสัญญาณที่ตั้งค่ากำลังการส่งเรียบร้อยแล้ว(กำลังการส่งสามารถปรับได้ 4 ระดับ ขึ้นอยู่กับผู้ใช้งานจะเลือกตั้งค่าบนอุปกรณ์ภาคส่งสัญญาณ) จะพบว่า เมื่อภาคส่งมีการส่งข้อมูลออกไป 1 แพคเกจ(3 Bytes) โดยรูปแบบข้อมูลจะมี 2 รูปแบบคือ

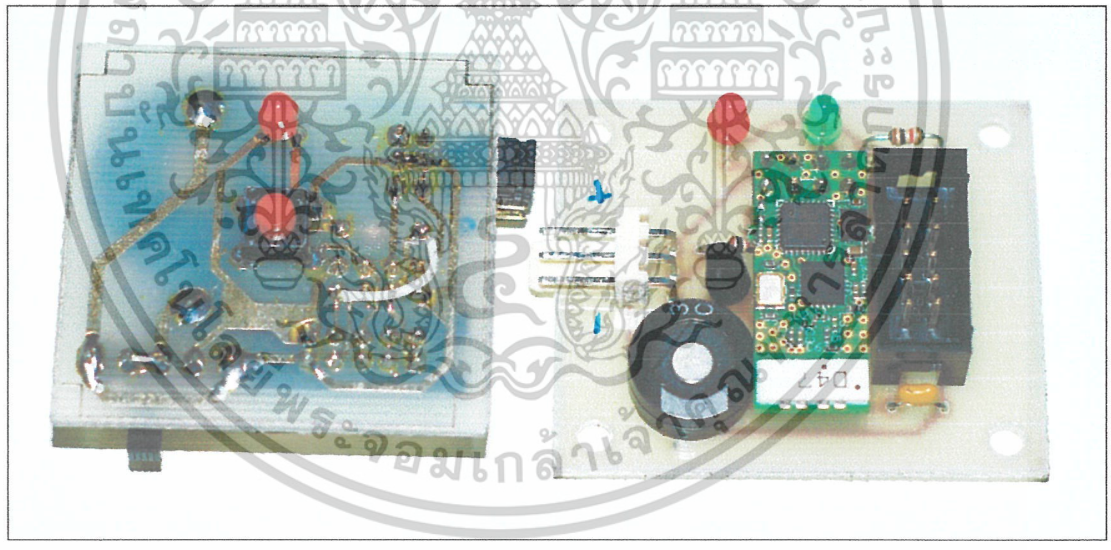
- ข้อมูลชุดปกติของอุปกรณ์ส่งสัญญาณตัวที่ 1 จะส่งข้อมูล ":", "1", "A"
- ข้อมูลชุดปกติของอุปกรณ์ส่งสัญญาณตัวที่ 2 จะส่งข้อมูล ":", "2", "A"
- ข้อมูลชุดฉุกเฉินของอุปกรณ์ส่งสัญญาณตัวที่ 1 จะส่งข้อมูล ":", "1", "S"
- ข้อมูลชุดฉุกเฉินของอุปกรณ์ส่งสัญญาณตัวที่ 2 จะส่งข้อมูล ":", "2", "S"

หลอดไฟ LED ที่อุปกรณ์ภาคส่งสัญญาณจะกระพริบขึ้น 1 ครั้งตามจังหวะการส่งข้อมูล เมื่อผู้พกพาอุปกรณ์ภาคส่งสัญญาณออกนอกพื้นที่ที่ระยะสัญญาณสามารถส่งถึงอุปกรณ์ภาครับสัญญาณที่ถูกติดตั้งไว้บริเวณหน้าอาคาร ทำให้ไม่สามารถรับข้อมูลได้ ภาครับสัญญาณจะทำการตรวจสอบข้อมูลจนกระทั่งไม่มีการรับข้อมูลเข้าเป็นเวลา 5 วินาที ทำให้หลอดไฟ LED และลำโพง Piezo ที่อยู่บนอุปกรณ์ภาครับสัญญาณทำงาน หรือในกรณีที่ผู้พกพาอุปกรณ์ภาคส่งสัญญาณกดสวิตช์แจ้งส่งสัญญาณฉุกเฉิน หลอดไฟ LED และลำโพง Piezo ก็จะทำการแจ้งเตือนเช่นเดียวกัน โดยเมื่อกดสวิตช์รีเซ็ตบนอุปกรณ์ภาครับสัญญาณการทำงานของหลอดไฟ LED และลำโพง Piezo จะหยุดทำงานลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

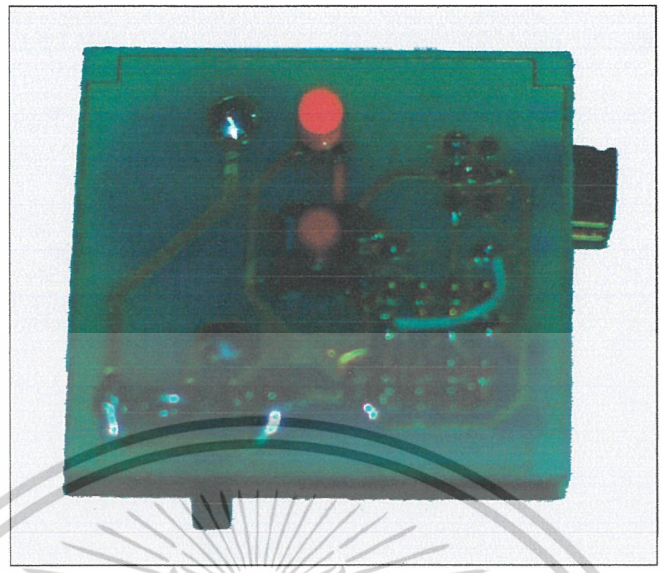


รูปที่ 4.1 ชุดอุปกรณ์ภาคส่งด้านหน้า(ทางซ้าย) และภาครับ(ทางขวา)สัญญาณไร้สาย

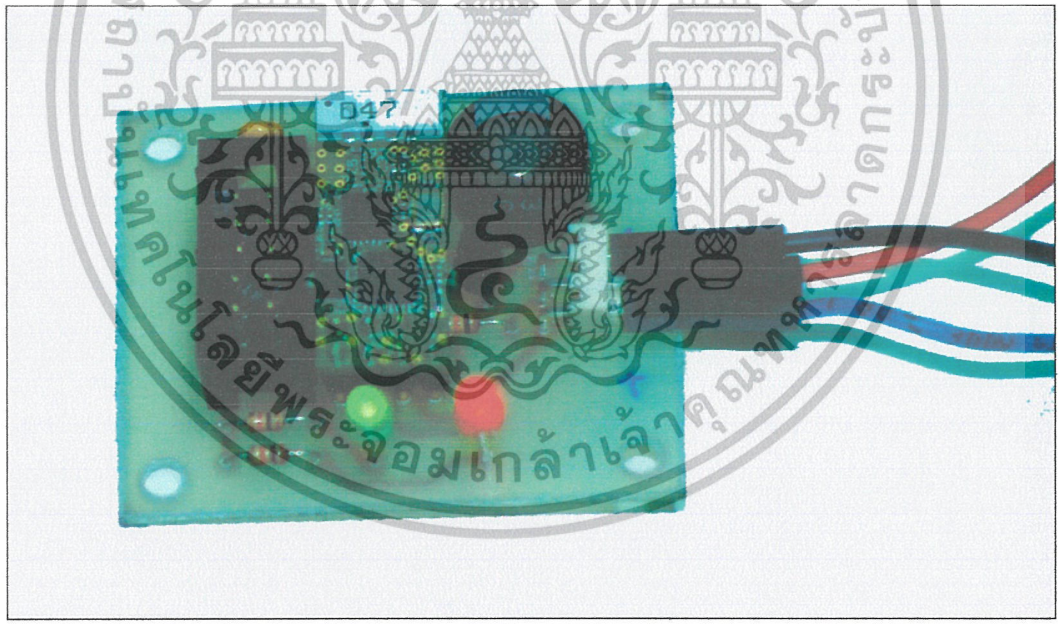


รูปที่ 4.2 ชุดอุปกรณ์ภาคส่งด้านหลัง(ทางซ้าย) และภาครับ(ทางขวา)สัญญาณไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 การทำงานของอุปกรณ์ภาคส่งสัญญาณ เมื่อมีการส่งข้อมูล(หลอดไฟ LED ทำงาน)



รูปที่ 4.4 การแจ้งเตือนของอุปกรณ์ภาครับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 ผลการทดลองระยะทางที่อุปกรณ์ภาครับสัญญาณสามารถรับข้อมูลได้ ในแต่ละระดับกำลังการส่ง ของอุปกรณ์ภาคส่งตัวที่ 1

กำลังการส่ง	ภายในอาคาร (m)	พื้นที่โล่งแจ้ง (m)
-18 dBm	1.5	1.8
-12 dBm	3.2	4.0
-6 dBm	7.5	9.5
0 dBm	10.0	14.6

ตารางที่ 4.2 ผลการทดลองระยะทางที่อุปกรณ์ภาครับสัญญาณสามารถรับข้อมูลได้ ในแต่ละระดับกำลังการส่ง ของอุปกรณ์ภาคส่งตัวที่ 2

กำลังการส่ง	ภายในอาคาร (m)	พื้นที่โล่งแจ้ง (m)
-18 dBm	1.6	1.8
-12 dBm	3.2	4.0
-6 dBm	7.7	9.7
0 dBm	10.0	15.0

จากการทดลองทั้ง 2 ครั้ง โดยใช้อุปกรณ์ภาคส่งสัญญาณครั้งละ 1 ตัว สรุปได้ว่า

1. ที่กำลังการส่ง -18 dBm จะสามารถส่งสัญญาณออกไปได้ระยะทางใกล้ที่สุด โดยมีกำลังการส่ง -12 dBm และ -6 dBm ได้ระยะทางที่มากขึ้นตามลำดับ และที่กำลังการส่ง 0 dBm สามารถส่งสัญญาณได้ระยะทางที่ไกลสุด

2. การส่งสัญญาณจากพื้นที่โล่งแจ้งจะสามารถส่งได้ระยะทางที่มากกว่าการส่งสัญญาณจากภายในตัวอาคารเล็กน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ผลการทดลองระยะทางที่อุปกรณ์ภาครับสัญญาณสามารถรับข้อมูลได้ ในแต่ละระดับกำลังการส่ง โดยใช้อุปกรณ์การส่งสัญญาณทั้ง 2 ตัวพร้อมกัน

กำลังการส่ง	ภายในอาคาร		พื้นที่โล่งแจ้ง	
	อุปกรณ์ส่งสัญญาณตัวที่ 1(m)	อุปกรณ์ส่งสัญญาณตัวที่ 2(m)	อุปกรณ์ส่งสัญญาณตัวที่ 1(m)	อุปกรณ์ส่งสัญญาณตัวที่ 2(m)
-18 dBm	1.3	1.5	1.5	1.7
-12 dBm	2.8	3.0	3.2	3.2
-6 dBm	6.5	6.6	7.0	7.0
0 dBm	11.4	11.5	12.9	13.0

จากการทดลอง โดยใช้อุปกรณ์ภาคส่งสัญญาณพร้อมกันทั้ง 2 ตัว พบว่า การส่งสัญญาณพร้อมกันของอุปกรณ์ภาคส่ง มีผลต่อระยะทางที่อุปกรณ์ภาคส่งจะสามารถส่งสัญญาณได้ ทำให้ระยะทางในแต่ละระดับกำลังการส่งน้อยกว่าระยะทางที่ส่งสัญญาณด้วยอุปกรณ์ส่งสัญญาณเพียงตัวเดียว

4.2 ผลการทดลองการรับข้อมูลของ iBoard โดยการส่งข้อมูลจากอุปกรณ์ภาครับ ผ่านทางสายพอร์ตอนุกรม

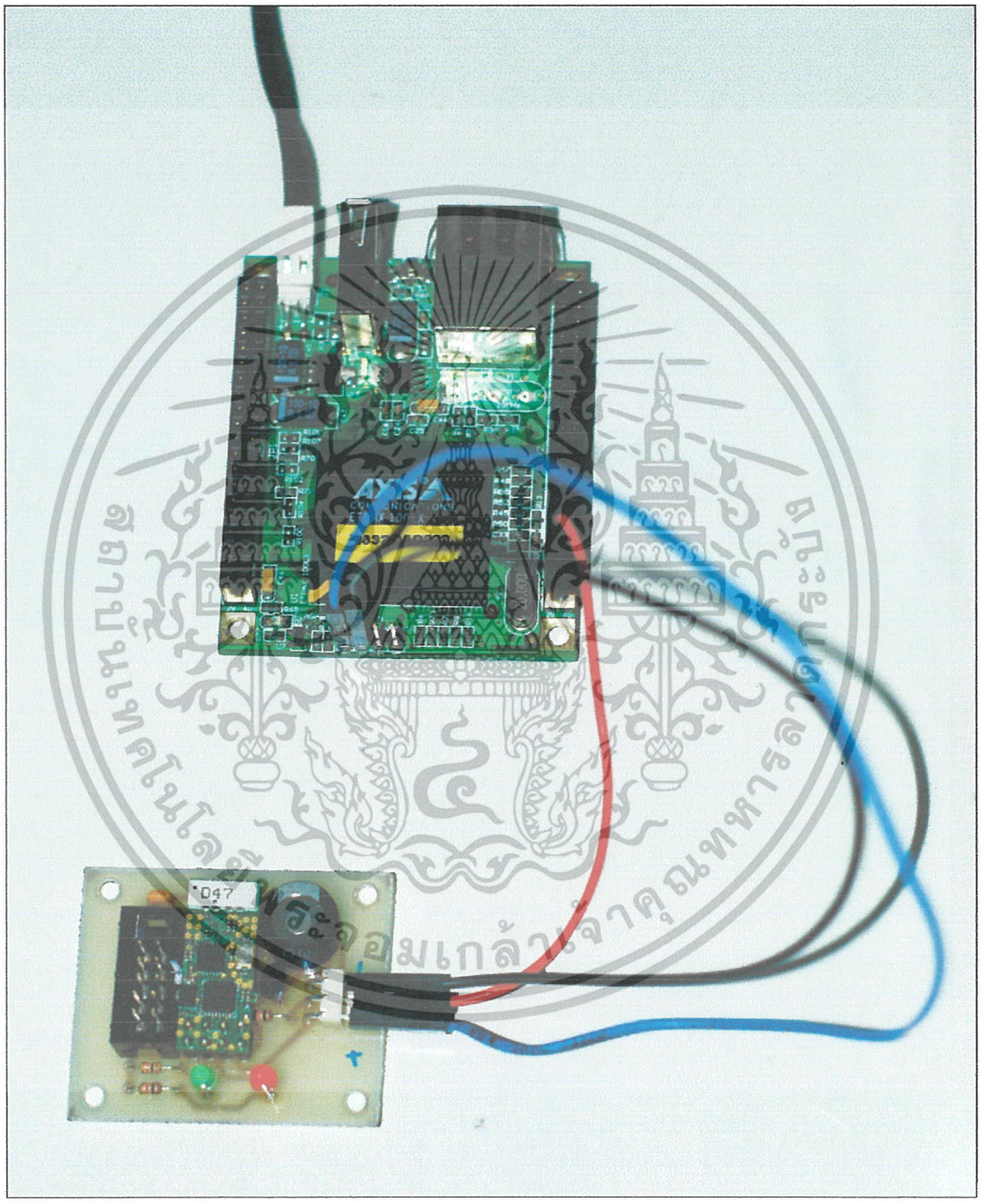
เมื่ออุปกรณ์ภาครับสัญญาณได้รับข้อมูลจากอุปกรณ์ส่งสัญญาณแล้วนั้น อุปกรณ์ภาครับสัญญาณจะทำการตรวจสอบข้อมูลที่ได้รับว่าเป็นข้อมูลชุดปกติหรือข้อมูลชุดฉุกเฉิน และตรวจสอบในกรณีที่ไม่ได้รับสัญญาณเกินกว่า 5 วินาที เพื่อเตรียมส่งต่อข้อมูลไปเก็บไว้ยังระบบฐานข้อมูลของ iBoard โดยผ่านทางสายพอร์ตอนุกรมที่เชื่อมการติดต่อระหว่างอุปกรณ์ภาครับสัญญาณและ iBoard ในการทดลองนี้จะแสดงผลการรับข้อมูลของ iBoard ทางระบบปฏิบัติการลินุกซ์ภายใน iBoard ซึ่งข้อมูลที่ได้กำหนดไว้ภายในอุปกรณ์ภาครับสัญญาณจะมี 2 รูปแบบ คือ ข้อมูลชุดฉุกเฉิน และข้อมูลที่เกิดจากการไม่ได้รับสัญญาณจากภาคส่งเป็นเวลาเกินกว่า 5 วินาทีจะถูกส่งเมื่อผู้พกพาอุปกรณ์ภาคส่งสัญญาณกดสวิตซ์บนตัวอุปกรณ์ รูปแบบข้อมูลที่กำหนดไว้ในการทดลอง มีรูปแบบดังนี้

- ที่เกิดจากการไม่ได้รับสัญญาณจากอุปกรณ์ภาคส่งตัวที่ 1 เป็นเวลาเกินกว่า 5 วินาที จะส่งข้อมูล ":", "1", "A", "n"
- ที่เกิดจากการไม่ได้รับสัญญาณจากอุปกรณ์ภาคส่งตัวที่ 2 เป็นเวลาเกินกว่า 5 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

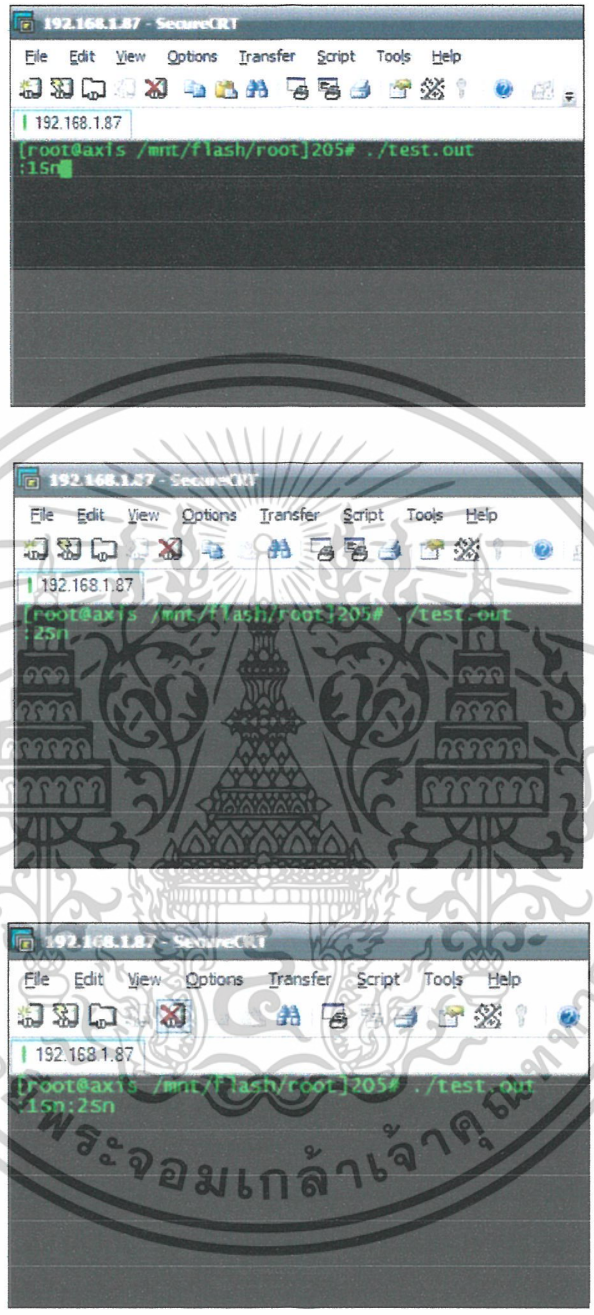
จะส่งข้อมูล ":", "2", "A", "n"

- ข้อมูลชุดฉุกเฉินของอุปกรณ์ส่งสัญญาณตัวที่ 1 จะส่งข้อมูล ":", "1", "S", "n"
- ข้อมูลชุดฉุกเฉินของอุปกรณ์ส่งสัญญาณตัวที่ 2 จะส่งข้อมูล ":", "2", "S", "n"



รูปที่ 4.5 ชุดอุปกรณ์รับสัญญาณและ iBoard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



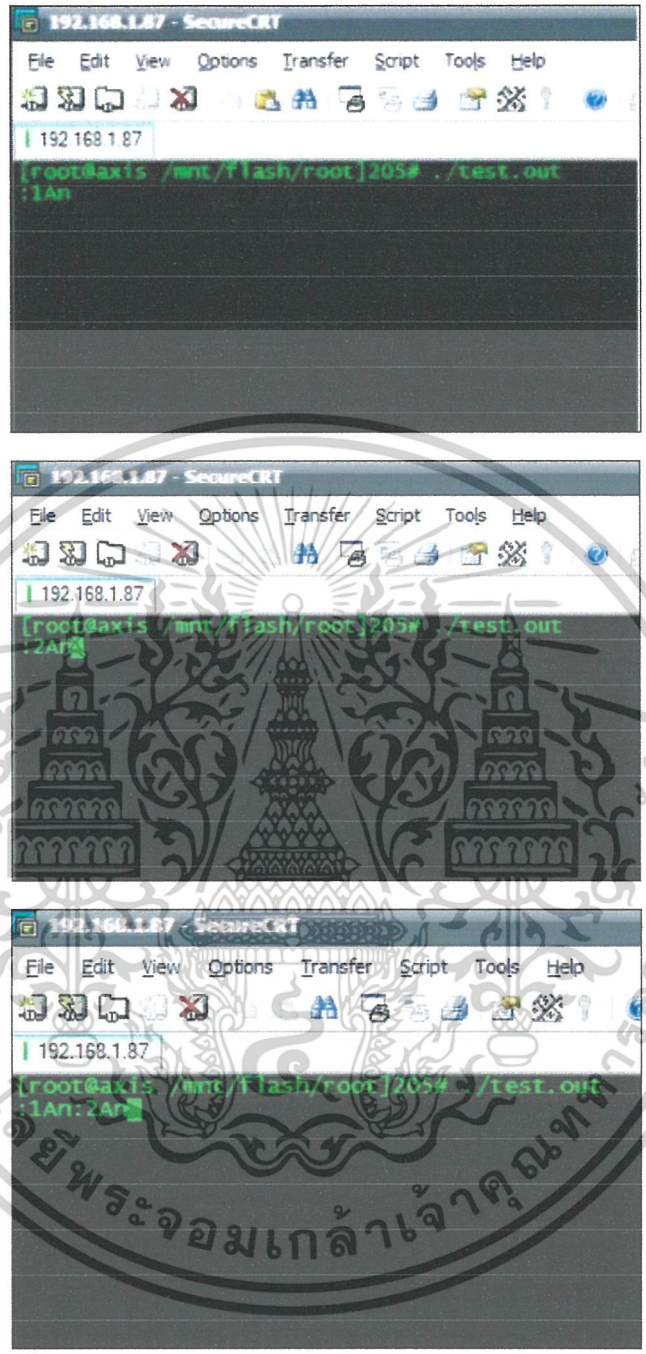
รูปที่ 4.6 หน้าจอแสดงผลการรับข้อมูลของ iBoard เมื่อมีการกดสวิตช์ส่งสัญญาณฉุกเฉิน

ก.) อุปกรณ์ส่งสัญญาณตัวที่ 1 มีการกดสวิตช์ส่งสัญญาณฉุกเฉิน

ข.) อุปกรณ์ส่งสัญญาณตัวที่ 2 มีการกดสวิตช์ส่งสัญญาณฉุกเฉิน

ค.) อุปกรณ์ส่งสัญญาณตัวที่ 1 และ 2 มีการกดสวิตช์ส่งสัญญาณฉุกเฉินพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 หน้าจอแสดงผลการรับข้อมูลของ iBoard เมื่อภาคส่งสัญญาณอยู่นอกระยะส่ง

ก.) อุปกรณ์ส่งสัญญาณตัวที่ 1 ออกนอกระยะส่งสัญญาณ

ข.) อุปกรณ์ส่งสัญญาณตัวที่ 2 ออกนอกระยะส่งสัญญาณ

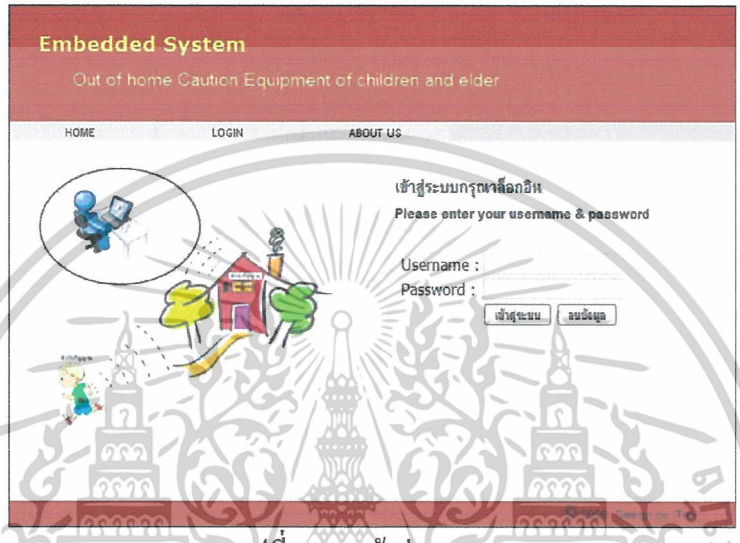
ค.) อุปกรณ์ส่งสัญญาณตัวที่ 1 และ 2 ออกนอกระยะส่งสัญญาณพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลองการทำงานของทั้งระบบ

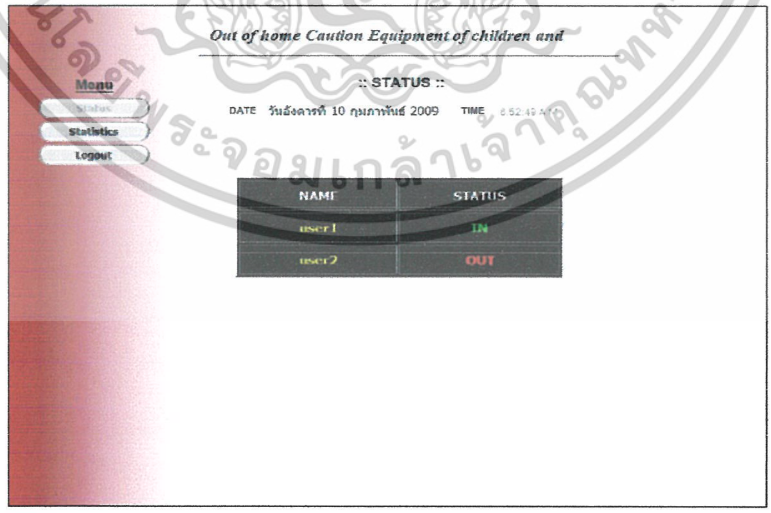
จากการทดลองที่ 4.1 และ 4.2 เมื่อ iBoard ได้รับข้อมูลเรียบร้อยแล้ว จะทำการเก็บข้อมูลลงระบบฐานข้อมูล ทำให้ผู้ใช้งานสามารถเรียกดูข้อมูลสถานะของผู้พกพาอุปกรณ์ภาคส่งสัญญาณได้ทั้งในรูปแบบเวลาจริงหรือเรียกดูข้อมูลย้อนหลังได้โดยผ่านทางหน้าเว็บแอพพลิเคชัน

1. ผู้ใช้งาน Login เข้าสู่หน้าเมนูหลัก โดยใส่ Username และ Password ที่ได้ตั้งไว้



รูปที่ 4.8 หน้าต่าง Login

2. เข้าสู่หน้าต่าง Status ดูข้อมูลสถานะของผู้พกพาอุปกรณ์ภาคส่งสัญญาณในรูปแบบเวลาจริง ในตารางแสดงผลสถานะจะประกอบด้วย หมายเลขของอุปกรณ์ภาคส่งสัญญาณ(1, 2) และสถานะการออกนอกพื้นที่หรือการแจ้งเหตุฉุกเฉิน(IN, OUT, SOS)



รูปที่ 4.9 หน้าต่าง Status

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เข้าสู่หน้าต่าง Statistics ดูข้อมูลสถานะของผู้พกพาอุปกรณ์ภาคส่งสัญญาณย้อนหลัง ในตารางแสดงผลสถานะจะประกอบด้วย หมายเลขของอุปกรณ์ภาคส่งสัญญาณ(1, 2) และสถานะการออกนอกพื้นที่หรือการแจ้งเหตุฉุกเฉิน(OUT, SOS)

09-02-2009

Date/Time	Name	Status
09/02/2009 08:13	user1	OUT
09/02/2009 08:16	user2	OUT
09/02/2009 08:24	user2	SOS
09/02/2009 08:30	user1	OUT
09/02/2009 08:33	user2	OUT
09/02/2009 08:56	user1	OUT
09/02/2009 09:12	user2	OUT
09/02/2009 09:30	user1	SOS
09/02/2009 09:58	user1	OUT
09/02/2009 10:01	user2	OUT
09/02/2009 10:23	user1	OUT
09/02/2009 10:30	user2	SOS
09/02/2009 10:35	user2	OUT

รูปที่ 4.10 หน้าต่าง Statistics แสดงข้อมูลสถานะย้อนหลังตามวันที่ที่เลือกไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินงาน

5.1 สรุปผล

ปริญญานิพนธ์นี้ได้ทำการออกแบบและสร้างชุดอุปกรณ์แจ้งเตือนการออกนอกพื้นที่บริเวณบ้าน เพื่อความสะดวกต่อการดูแลเด็กและคนชราให้อยู่ภายในพื้นที่ที่จำกัดไว้ โดยการดำเนินงานของระบบอุปกรณ์ได้เป็น 3 ส่วน คือ

1. ส่วนการทำงานของอุปกรณ์ภาครับและภาคส่งสัญญาณไร้สาย โดยใช้อุปกรณ์ชุดโมดูล DCBT-24AX มีอุปกรณ์ภาครับสัญญาณ 1 ชุด และภาคส่งสัญญาณ 2 ชุด การทำงานของอุปกรณ์สามารถทำการแจ้งเตือนสถานะของเด็กและคนชราผ่านทางหลอดไฟ LED และลำโพงที่อุปกรณ์ภาครับได้ดีทั้งในพื้นที่โล่งแจ้งและในอาคาร การรบกวนกันของสัญญาณจากภาคส่งจะเกิดขึ้นเพียงเล็กน้อยเท่านั้น

2. ส่วนการส่งข้อมูลจากอุปกรณ์ภาครับสัญญาณสามารถส่งข้อมูลที่ได้รับออกสู่ iBoard ผ่านทางสายพอร์ตอนุกรม เพื่อเก็บข้อมูลสถานะของเด็กและคนชราเข้าสู่ระบบฐานข้อมูล พร้อมที่จะนำไปแสดงผลทางเว็บแอพพลิเคชันต่อไป

3. ส่วนการแสดงผล ผู้ใช้งานสามารถตรวจสอบสถานะของเด็กและคนชราได้ ทั้งในเวลาจริงและข้อมูลย้อนหลัง โดยเรียกดูผ่านทางหน้าเว็บแอพพลิเคชัน

5.2 ปัญหาและข้อจำกัดของการดำเนินงาน

1. เนื่องจากอุปกรณ์ฮาร์ดแวร์ที่ได้นำมาสร้างและออกแบบ เป็นอุปกรณ์ที่ออกวางขายในระยะเวลาที่ไม่นานนัก ทั้งในส่วนของ DCBT-24AX และ iBoard ทำให้การศึกษาหาข้อมูลอุปกรณ์เพื่อนำมาเขียนโปรแกรมพัฒนาและการเชื่อมต่อกันนั้นใช้เวลาที่นานกว่าที่วางแผนไว้

2. จากการส่งสัญญาณไร้สายของอุปกรณ์ภาคส่งสัญญาณ ในบางครั้งเกิดการรบกวนจากสัญญาณภายนอกขึ้น ทำให้ส่งสัญญาณได้ระยะทางที่ใกล้กว่าที่ทดลองไว้

3. อุปกรณ์ iBoard ในปัจจุบันมีราคาที่ค่อนข้างแพง จึงทำให้ผู้ที่นิยมพัฒนาบอร์ดไมโครคอนโทรลเลอร์ไม่นิยมนำอุปกรณ์นี้มาพัฒนา นอกจากโรงงานหรือบริษัทต่างๆ ที่เริ่มหันมาสนใจพัฒนา iBoard กันมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางในการพัฒนาต่อ

ปริญญานิพนธ์นี้สามารถนำไปพัฒนาต่อได้อย่างหลากหลาย เนื่องจากได้ทำการพัฒนาทั้งฮาร์ดแวร์และซอฟต์แวร์ควบคุมการทำงานของตัวรับส่งสัญญาณไร้สายชุดโมดูล DCBT-24AX โดยสามารถพัฒนาเชื่อมต่อไมโครโฟน และลำโพงเพิ่มเข้ามา เพื่อให้เด็กและคนชราสามารถโต้ตอบกับผู้ใช้งานระบบได้ หรือการนำไปเชื่อมต่อการทำงานกับ GPS, GPRS เพื่อแสดงผลระบุตำแหน่งของเด็กและคนชรา ในปริญญานิพนธ์ยังได้ทำการเก็บข้อมูลเข้าสู่ฐานข้อมูลในเทคโนโลยีสมองกลฝังตัว iBoard และนำมาแสดงผลผ่านทางหน้าเว็บแอพพลิเคชัน ซึ่งปัจจุบัน iBoard กำลังเป็นที่นิยมในการนำมาพัฒนาเทคโนโลยีอุปกรณ์ต่างๆ โดยผู้พัฒนาสามารถทดลองต่อวงจรได้บนโฟโต้บอร์ด(Photo Board) ได้เลย เช่น Relay, LCD Character, LCD Graphic เป็นต้น พร้อมกับมีพอร์ตอนุกรม(RS232, RS485) และพอร์ต USB รองรับการเชื่อมกับอุปกรณ์ภายนอก ทำให้สามารถพัฒนาแอพพลิเคชันได้อย่างรวดเร็ว และสะดวกขึ้น ในปริญญานิพนธ์นี้นำ iBoard มาพัฒนาในเรื่องการเก็บฐานข้อมูลและแสดงผลโดยเว็บเซิร์ฟเวอร์ ดังนั้นจึงสามารถนำ iBoard ไปพัฒนาต่อทั้งในด้านการแสดงผลทางหน้าจอ LCD หรือการแสดงผลผ่านทางโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] กาญจนา ตันวิสุทธิ์. **เก่ง Ajax+PHP ให้ครบสูตร.** กรุงเทพฯ : วิตตี้ กรุ๊ป, 2551.

[2] ประภาพร ช่างไม้. **สร้างเว็บสวยด้วย Dreamweaver 8**. นนทบุรี : ไอดีซี, 2550

[3] วรรณวิภา ติตถะสิริ. **คู่มือเรียน SQL ด้วยตัวเอง**. กรุงเทพฯ : โปรวิชั่น, 2545

[4] อนรรฆมนงค์ คุณมณี. **Basis of PHP**. นนทบุรี : ไอดีซีฯ, 2550.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

คู่มือการใช้งานและการติดตั้งโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. คู่มือการใช้งาน

1.1 การเตรียมอุปกรณ์ก่อนใช้งานกล่องภาคส่งสัญญาณและกล่องภาครับสัญญาณ

1. ติดตั้งกล่องภาครับสัญญาณ(ภายในจะประกอบด้วยอุปกรณ์รับสัญญาณไร้สาย และ iBoard) บริเวณหน้าบ้านหรือหน้าอาคาร เพื่อความสะดวกแก่การรับสัญญาณจากภาคส่งสัญญาณไร้สายควรติดตั้งให้สูง และไม่มีสิ่งกีดขวางในการรับ-ส่งสัญญาณ

2. เมื่อเริ่มใช้งาน เสียบสาย Adapter จากกล่องภาครับสัญญาณ เพื่อจ่ายไฟให้อุปกรณ์ทำงาน เมื่อกระแสไฟเข้าสู่อุปกรณ์เรียบร้อยแล้ว หลอดไฟ LED ทั้ง 2 ดวงจะกระพริบขึ้นพร้อมกัน

3. สับสวิตช์เปิดการทำงานของภาคส่งสัญญาณที่ข้างกล่องภาคส่งสัญญาณ สังเกตเมื่อกล่องอุปกรณ์ทำงาน หลอดไฟ LED ที่หน้ากล่องจะกระพริบขึ้นทุกครั้งเมื่อมีการส่งสัญญาณออกไป

4. นำกล่องภาคส่งสัญญาณให้เด็กและคนชราที่ต้องการดูแล ป้องกันการออกนอกพื้นที่พกติดตัวไว้ โดยหากเกิดเหตุฉุกเฉิน สามารถกดปุ่มสีดำที่บนตัวกล่องแจ้งให้ผู้ปกครองทราบ หากมีการใช้งานกล่องภาคส่งสัญญาณพร้อมกันทั้ง 2 เครื่อง ให้ตรวจสอบก่อนว่าเด็กหรือคนชราคนใดพกกล่องภาคส่งสัญญาณตัวที่ 1 หรือ 2

1.2 การแจ้งเตือนเมื่อเด็กและคนชราออกนอกพื้นที่หรือแจ้งเหตุฉุกเฉินที่กล่องภาครับสัญญาณ

1. กล่องภาครับสัญญาณจะมีการแจ้งเตือน เมื่อมีการออกนอกพื้นที่หรือมีการกดส่งสัญญาณแจ้งเหตุฉุกเฉินจากเด็กและคนชรา โดยมีเสียงดังขึ้น และหลอดไฟ LED จะสว่าง ซึ่งหลอดสีเขียวหมายถึง กล่องภาคส่งสัญญาณตัวที่ 1 และหลอดสีแดงหมายถึง กล่องภาคส่งสัญญาณตัวที่ 2

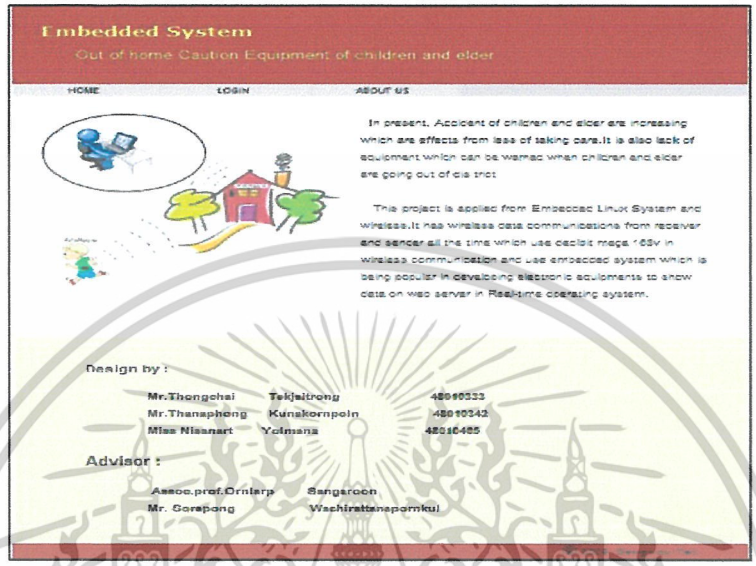
2. การแจ้งเตือนของกล่องภาครับสัญญาณจะสิ้นสุดลง เมื่อเด็กและคนชราที่ออกนอกพื้นที่เดินกลับเข้าในพื้นที่

3. ในกรณีมีการแจ้งเหตุฉุกเฉิน การแจ้งเตือนจะสิ้นสุดลง เมื่อมีการปิดการทำงานของกล่องภาครับสัญญาณ โดยการดึงสาย Adapter ออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

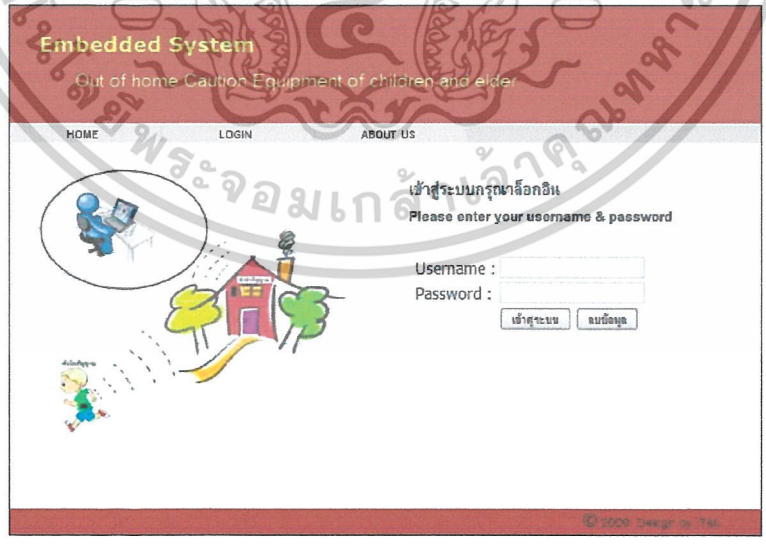
1.3 การตรวจสอบสถานะของเด็กและคนชราผ่านเว็บแอพพลิเคชัน

1. เปิดโปรแกรมระบบการแจ้งเตือนการออกนอกพื้นที่บริเวณบ้านของเด็กและคนชรา จะเข้าสู่หน้าโฮมเพจ



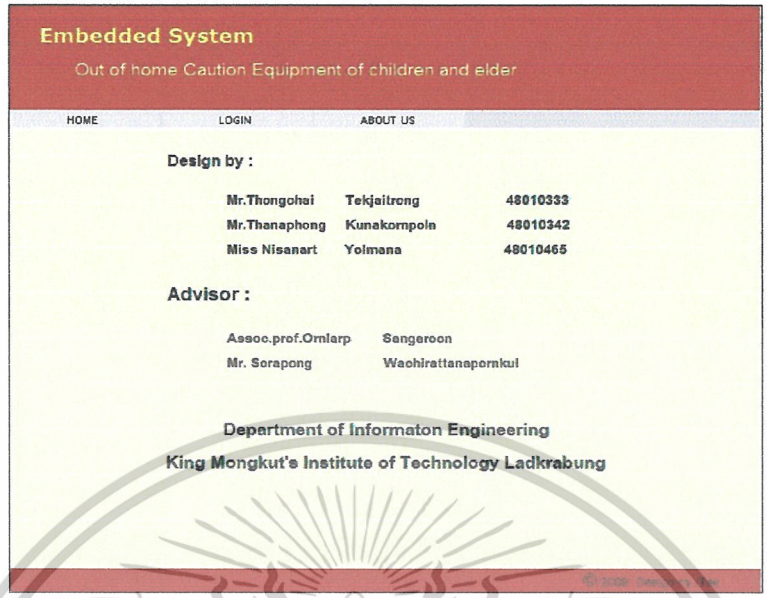
รูปที่ ก.1 หน้าต่างโฮมเพจ

ผู้ใช้งานจะสามารถเลือกทำรายการ Login เข้าตรวจสอบสถานะของเด็กและคนชราหรือเรียกแสดงข้อมูลผู้สร้างและพัฒนาระบบอุปกรณ์แจ้งเตือนการออกนอกพื้นที่ของเด็กและคนชราได้โดย Click ที่ Login หรือ About us ด้านบน



รูปที่ ก.2 หน้าต่าง Login

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.3 หน้าต่าง About us

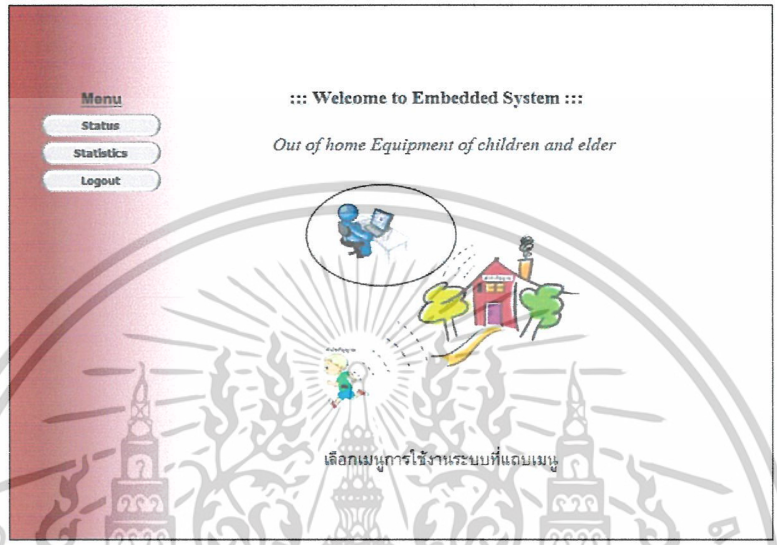
2. เมื่อผู้ใช้งานเข้าสู่หน้าต่าง Login ให้ใส่ Username และ Password ในช่องที่เว้นว่างไว้ แล้ว Click เข้าสู่ระบบ จะเข้าสู่หน้างต่างเมนูหลัก หากผู้ใช้งานใส่ Username และ Password ผิด ผู้ใช้งานจะไม่สามารถเข้าสู่หน้าเมนูหลักได้ โดยระบบจะกลับสู่หน้าเว็บเพจ Login

รูปที่ ก.4 หน้าต่าง เมื่อผู้ใช้งานใส่ Username และ Password ผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

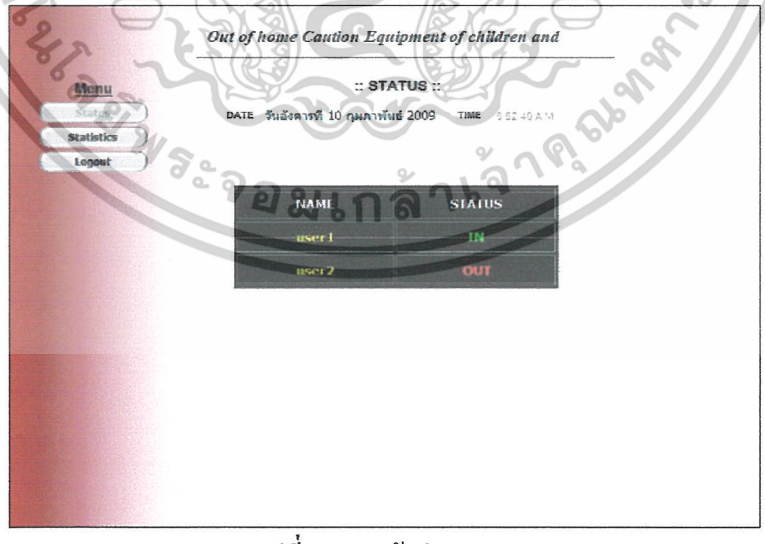
3. ที่หน้าต่างเมนูหลักผู้ใช้งานจะเลือก Click ทำรายการได้ 3 อย่าง คือ

- **Status** สำหรับดูสถานะของเด็กและคนชราในรูปแบบเวลาจริง
- **Statistics** สำหรับดูข้อมูลสถานะของเด็กและคนชราย้อนหลัง
- **Logout** เมื่อต้องการออกจากระบบ



รูปที่ ก.5 หน้าต่างเมนูหลัก

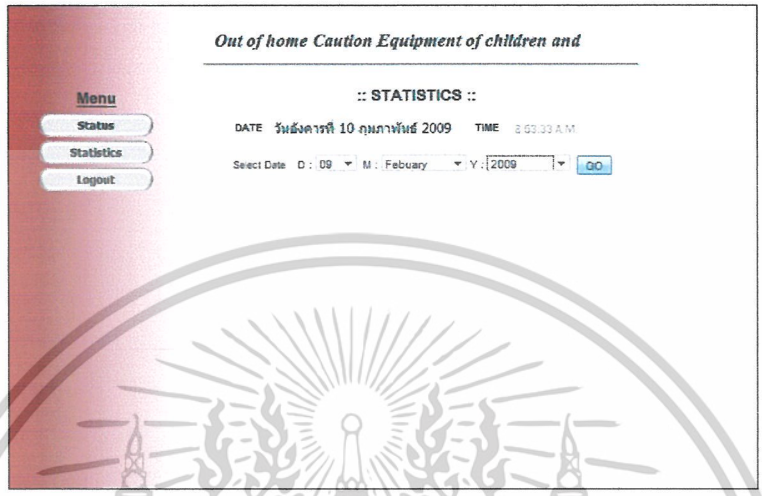
ด้านบนของหน้าต่าง Status จะแสดงวันที่และเวลาปัจจุบัน และในตารางแสดงสถานะของเด็กและคนชราจะประกอบด้วย หมายเลขของกล่องภาคส่งสัญญาณ(1, 2) และสถานะการออกนอกพื้นที่หรือการแจ้งเหตุฉุกเฉิน(IN, OUT, SOS)



รูปที่ ก.6 หน้าต่าง Status

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าต่าง Statistics ผู้ใช้งานจะต้องเลือกวันที่ เดือน และปีที่ต้องการทราบข้อมูล โดยจะแสดงผลหมายเลขของกล่องภาคส่งสัญญาณ(1, 2) และสถานะการออกนอกพื้นที่หรือการแจ้งเหตุฉุกเฉิน(OUT, SOS) ตามวันที่ที่ได้เลือกไว้ลงในตาราง



รูปที่ ก.7 หน้าต่าง Statistics

09-02-2009

Date/Time	Name	Status
09/02/2009 08:13	user1	OUT
09/02/2009 08:16	user2	OUT
09/02/2009 08:24	user2	SOS
09/02/2009 08:30	user1	OUT
09/02/2009 08:33	user2	OUT
09/02/2009 08:56	user1	OUT
09/02/2009 09:12	user2	OUT
09/02/2009 09:30	user1	SOS
09/02/2009 09:58	user1	OUT
09/02/2009 10:01	user2	OUT
09/02/2009 10:23	user1	OUT
09/02/2009 10:30	user2	SOS
09/02/2009 10:35	user2	OUT

รูปที่ ก.8 หน้าต่าง Statistics แสดงข้อมูลสถานะย้อนหลังตามวันที่ที่เลือกไว้

4. Click Logout เมื่อผู้ใช้งานต้องการออกจากระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การติดตั้งโปรแกรม

สามารถแบ่งโปรแกรมที่เกี่ยวข้องในการสร้างและพัฒนาระบบได้ 3 แบบ

2.1 โปรแกรมสำหรับโปรแกรมคำสั่งลงชุดโมดูล DCBT-24AX

ในการโปรแกรมคำสั่งลงชุดโมดูล DCBT-24AX จะใช้โปรแกรม AVRPROG เป็นผลงานของ Atmel การติดตั้งจะติดตั้งพร้อมกับโปรแกรม AVR Studio

รูปที่ ก.9 ไอคอนโปรแกรม AVRPROG

การติดตั้งโปรแกรม AVR Studio

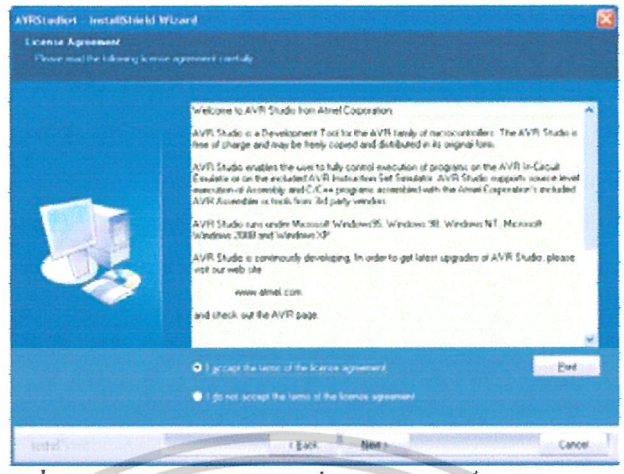
คอมพิวเตอร์ที่ใช้ควรติดตั้งระบบปฏิบัติการ Window ME ขึ้นไป

1. Double Click ที่ไฟล์ติดตั้งของโปรแกรม AVR Studio
2. หลังจากนั้นเข้าสู่หน้าต่างแรกของการติดตั้ง AVR Studio ดังรูป จากนั้น Click ที่ปุ่ม Next

รูปที่ ก.10 หน้าต่างเริ่มการติดตั้ง AVR Studio

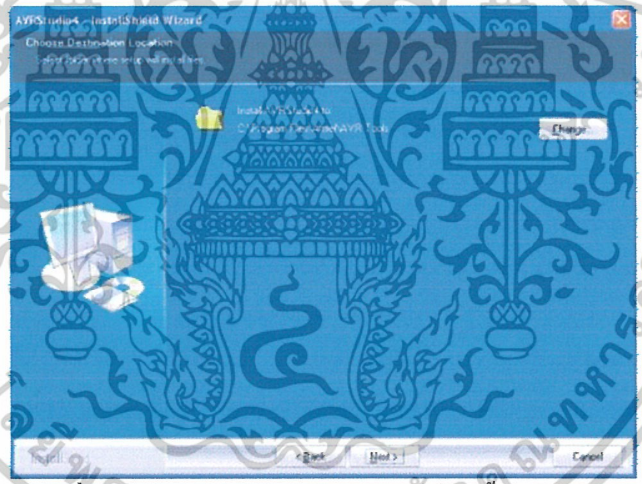
3. เข้าสู่หน้าต่างข้อตกลงเงื่อนไขลิขสิทธิ์และการใช้งาน ซึ่งจะมีรายละเอียดคุณสมบัติของเครื่องคอมพิวเตอร์ และระบบปฏิบัติการที่สามารถรองรับการทำงานของ AVR Studio รวมไปถึงชื่อบริษัทผู้ผลิตและอื่นๆ โดยให้เลือกช่อง I accept the terms of the license agreement จากนั้น Click ที่ปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.11 หน้าต่างข้อตกลงเงื่อนไขลิขสิทธิ์และการใช้งาน

4. เข้าสู่หน้าต่างการติดตั้งในลำดับถัดไป ที่แสดงชื่อไดเรกทอรี่ในการติดตั้งโปรแกรม AVR Studio ผู้ติดตั้งสามารถเปลี่ยนไดเรกทอรี่ได้ โดย Click ที่ปุ่ม Change แล้วกำหนดไดเรกทอรี่ใหม่ที่ต้องการ จากนั้น Click ที่ปุ่ม Next



รูปที่ ก.12 หน้าต่างแสดงไดเรกทอรี่ที่ติดตั้งโปรแกรม

5. หลังจากนั้นเข้าสู่หน้าต่างการอัพเกรดไดรเวอร์ ให้ Click ที่ปุ่ม Next ผ่านไป เนื่องจากไดรเวอร์ดังกล่าวไม่มีการใช้งานใดๆกับชุดโมดูล DCBT-24AX

6. เข้าสู่หน้าต่างเริ่มการติดตั้งโปรแกรม ให้ Click ปุ่ม Install

7. หลังจากนั้นจะเริ่มกระบวนการติดตั้งโปรแกรม AVR Studio รอจนกระทั่งการติดตั้งเสร็จสมบูรณ์

8. เมื่อการติดตั้งเสร็จสมบูรณ์ ให้ Click ปุ่ม Finish เป็นอันเสร็จสิ้นการติดตั้งโปรแกรม AVR Studio

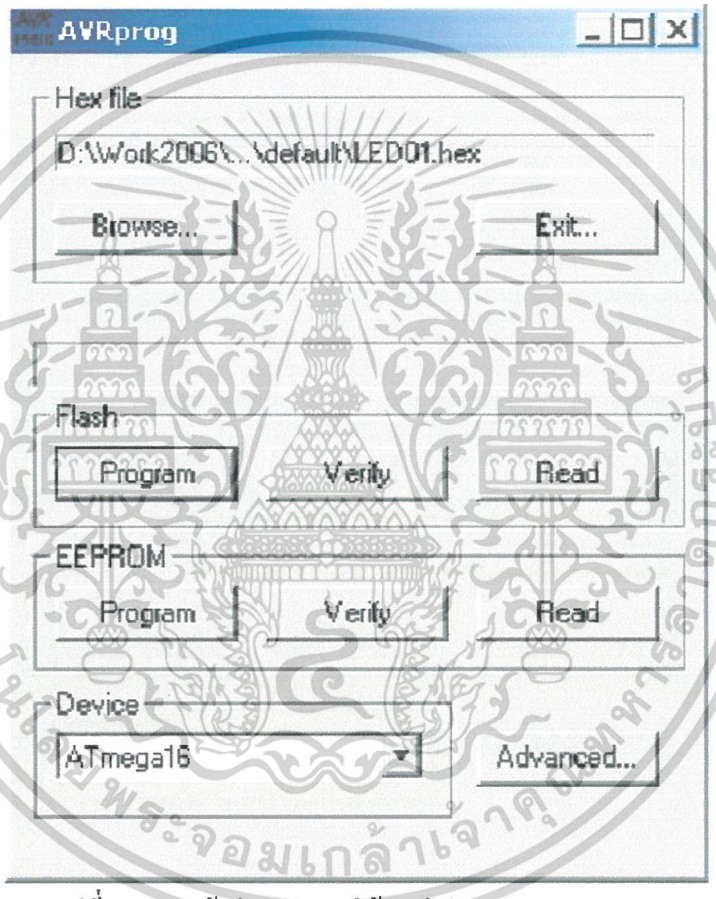
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การโปรแกรมคำสั่งลงชุดโมดูล DCBT-24AX

1. เปิดซอร์ฟแวร์ AVGPROG

2. สำหรับกรณีที่ไม่ได้จ่ายไฟ โปรแกรมจะเตือนว่า "ไม่สามารถตรวจพบเครื่องโปรแกรมได้" จะต้องจ่ายไฟให้แผงวงจรไมโครคอนโทรลเลอร์ AVR และต่อสายเข้ากับคอมพิวเตอร์ให้เรียบร้อยก่อนใช้งาน

3. สำหรับโปรแกรม AVGPROG ออกแบบมาเฉพาะเพื่อใช้งานกับไมโครคอนโทรลเลอร์ AVR ซึ่งมีเมนูต่างๆที่ใช้งานง่าย



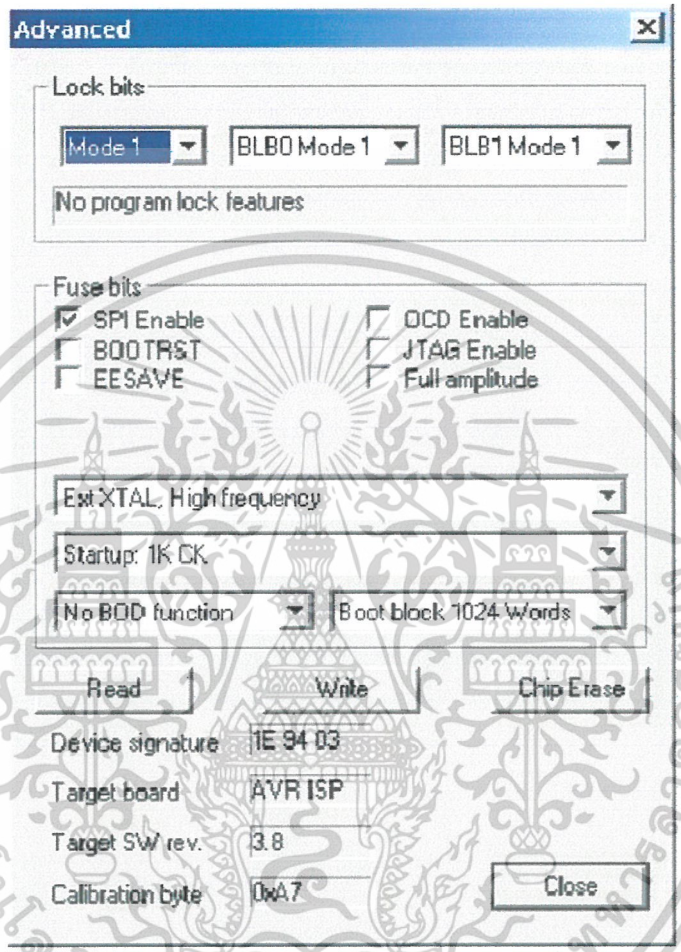
รูปที่ ก.13 หน้าต่างเมนูการใช้งานโปรแกรม AVRPROG

4. เริ่มต้นจะต้อง Click ที่ปุ่ม Advanced เพื่อเข้าสู่การตั้งค่าพิเศษก่อน ในครั้งต่อไปไม่จำเป็นต้องกำหนดค่าต่างๆ นี้อีก ประกอบด้วย

- กำหนดโหมดในการ Lock Bit เป็น Mode 1 กรณีไม่ต้องการป้องกันการอ่านข้อมูลย้อนหลังจากที่ทำการโปรแกรมเรียบร้อยแล้ว(สามารถกำหนดค่าใหม่เพื่อป้องกันการอ่านได้)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลือกช่อง SPI Enable (สำคัญมาก ห้ามเลือกออก เพราะจะทำให้ไมโครคอนโทรล-เลอร์ตัวนั้นไม่สามารถทำการโปรแกรมในครั้งต่อๆไปได้)
- กำหนดโหมดความถี่ของออสซิลเลเตอร์ เป็นแบบ Ext XTAL.,High frequency



รูปที่ ก.14 หน้าต่างการตั้งค่าพิเศษ

5. กดปุ่ม Write เพื่อบันทึกรูปแบบที่กำหนดไว้ ผู้ใช้งานสามารถกดปุ่ม Chip Erase เพื่อลบข้อมูลในหน่วยความจำโปรแกรมภายในไมโครคอนโทรลเลอร์ แล้วกดปุ่ม Close เพื่อจบขั้นตอนการตั้งค่าพิเศษ

6. เรียกไฟล์นามสกุล .hex โดย Click ที่ Browse…

7. กดปุ่ม Program เพื่อโปรแกรมข้อมูลไปยังไมโครคอนโทรลเลอร์ ไฟสีแดงในตำแหน่ง Busy ของเครื่องโปรแกรมจะติดสว่างจนกว่าการโปรแกรมจะเสร็จสิ้น ในขณะที่แถบแสดงสถานะจะแสดงผลการทำงานในขั้นตอนต่างๆ ในการโปรแกรมจนเสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 โปรแกรมสำหรับใช้งาน iBoard

วิธีการเตรียมเครื่องคอมให้คอมไพล์ให้ iBoard ใช้งานได้

โดยในการใช้งานครั้งนี้ได้ใช้ Ubuntu 8.10 มาเป็นตัวใช้งาน โดยต้อง install แพคเกจเหล่านี้ก่อน

```
$ sudo apt-get install libc6-dev
$ sudo apt-get install libncurses5-dev
$ sudo apt-get install pmake
$ sudo apt-get install zlib1g-dev
$ sudo apt-get install flex
$ sudo apt-get install bison
$ sudo apt-get install subversion
```

เปลี่ยนพาธของเครื่องมาเป็นพาธที่ใช้ไลบารี่ของ iBoard

```
$ sudo ln -sf /bin/bash /bin/sh
```

เปลี่ยนกับเมื่อต้องการใช้ไลบารี่ของเครื่อง

```
$ sudo ln -sf /bin/dash /bin/sh
```

ดาวน์โหลดไฟล์ที่ใช้ในการคอมไฟล์ cris-dist_1.63-1_i386.deb แล้วทำการลงโปรแกรมให้เรียบร้อย

```
$ su
Password: your_root_password
# dpkg -i cris-dist_1.63-1_i386.deb
```

ดาวน์โหลดตัวสคริปต์ไฟล์ที่ใช้ในการลง os ของตัว iBoard ให้สามารถสร้างบนเครื่องคอมพิวเตอร์ได้โดยดาวน์โหลดตัวนี้ install_svn_sdk.sh แล้วเขียนคำสั่งดังนี้ลงในเทอร์มินอล

```
# wget
http://www.acmesystems.it/download/install_svn_sdk.sh
```

ทำการรันไฟล์ดังนี้

```
# chmod +x install_svn_sdk.sh
# ./install_svn_sdk.sh
...
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากรันแล้วต้องรอเป็นเวลาในการลงเมื่อเสร็จแล้วจะได้ดังนี้

```
...
### Selected product: "fox" ###
etrax100boot must be run by root.
To make this easier (but less secure) you can make
etrax100boot setuid root.
Do you want to make etrax100boot setuid root now [yn]?
(default n):
```

เมื่อเสร็จแล้วให้เข้าไปในไดเรคทอรี่โดยใช้คำสั่งดังนี้

```
# cd devboard-R2_01
```

การลงเว็บเซิร์ฟเวอร์ foxserve ให้ตัวiboard

1. ทำการสร้าง fimage ของไฟล์ขึ้นมาก่อนโดย

```
# ./configure
### Selected product: "fox" ###
 * Using previously fetched packages... done
 * Using compiler "/usr/local/cris/bin/cris-axis-linux-
gnu-gcc" (revision "R63").
 * Using previously fetched packages... done
 * Generating "Makefile"... done
Creating init_env... ETC...

# ./make
...
```

2. ดาวน์โหลดไฟล์ xspatch.tar.gz มาแล้วทำการแตกไฟล์นั้น

```
cd /home/kanta (/home/kanta is my path to the patch,
replace it with your own)
tar xzvf xspatch.tar.gz
```

3. แล้วทำการเก็บค่าหลักของไฟล์ไว้และลงไฟล์ใหม่ดังนี้

```
cp .config .config_orig
cp kernelconfig-2.6 kernelconfig-2.6_orig
cp /PATH_TO_THE_PATCH/.config .
cp /PATH_TO_THE_PATCH/kernelconfig-2.6 .
```

4. คอนฟิกไฟล์ตัวใหม่ที่ได้ลงมา

```
# ./configure
### Selected product: "fox" ###
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 * Using previously fetched packages... done
 * Using compiler "/usr/local/cris/bin/cris-axis-linux-
gnu-gcc" (revision "R63").
 * Using previously fetched packages... done
 * Generating "Makefile"... done
Creating init_env... ETC...
```

5. ก็อปปี้ appfoxserve.tar.gz ลงใน ไดเรคทอรี่ devboard แล้วทำการแก้ซิปออกดังนี้

```
cp /PATH_TO_THE_PATCH/appfoxserve.tar.gz .
tar xzvf appfoxserve.tar.gz
```

6. คัดลอกตัว makefile (เป็นตัวที่ไว้ไปสร้างไฟล์ fimage ขึ้นมา)

```
cp Makefile Makefile.orig
cp /PATH_TO_THE_PATCH/Makefile .
```

7. สร้างไฟล์โดยใช้คำสั่ง

```
make
```

8. ขั้นตอนสุดท้ายนำไฟล์ที่ได้ไปลงใน iboard โดยจะได้ค่า IP เครื่องที่เป็นค่าแรกคือ 192.168.0.96

วิธีการลงระบบปฏิบัติการใหม่ให้แก่ iboard

เข้าไปในไดเรคทอรี่ ...../devboard-R2_01/ ก่อนแล้วใช้คำสั่ง

```
# . init_env
# boot_linux -F -i image_filename
...
```

เมื่อรันแล้วจะขึ้นคำสั่งตามด้านล่างให้ทำการเปลี่ยนสวิตช์ เบอร์ 3 ของตัว iBoard ลงแล้วเสียบสายไฟเข้าตัว iBoard

```
Using internal boot loader: INTERNAL_NW - Network boot
(default).
Starting boot...
We're doing a flash write, this may take up to a few
minutes...
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการจ่ายไฟเข้า iboard รอซักครู่จะมีการรันดังนี้

```
...
0x80360000: Writing 0x00010000 bytes
0x80370000: Writing 0x00010000 bytes
0x80380000: No need to write
0x80390000: No need to write
0x803a0000: Writing 0x00010000 bytes
0x80000000: Verifying...OK
JUMP
0x00000000
END
Exiting with code 0
```

ทำการเปลี่ยนสวิตช์กลับที่เดิมและทำการรีเซทเครื่องใหม่เป็นขั้นตอนสุดท้าย

2.3 โปรแกรมสำหรับสร้างเว็บแอพพลิเคชัน

โปรแกรมสำหรับสร้างเว็บแอพพลิเคชันในระบบนี้ ประกอบด้วย 2 โปรแกรม คือ โปรแกรม Dreamweaver 8 และโปรแกรม Appserv

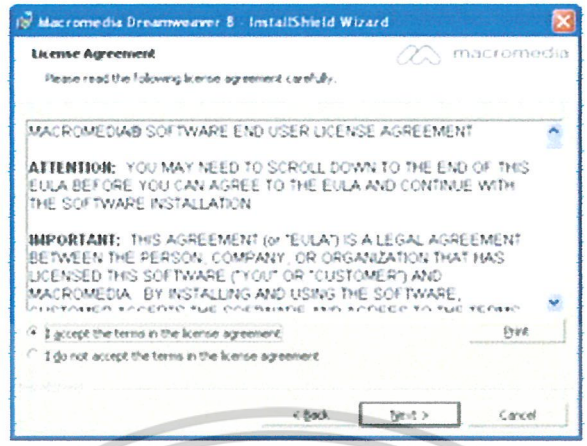
1. โปรแกรม Dreamweaver 8

Macromedia Dreamweaver 8 เป็นโปรแกรมที่ใช้สำหรับสร้างเว็บเพจ และบริหารเว็บไซต์ที่ได้รับความนิยมสูงสุด ในกลุ่มโปรแกรมประเภทเดียวกัน ในปัจจุบัน เนื่องจากคุณสมบัติของโปรแกรมที่มีความสะดวกต่อการใช้งาน มีฟังก์ชันที่ทำให้ผู้ใช้สามารถจัดวางข้อความ รูปภาพ ตาราง ฟอร์ม วิดีโอ รวมถึงองค์ประกอบอื่น ๆ ภายในเว็บเพจได้อย่างสวยงามตามที่ผู้ใช้ต้องการ โดยไม่ต้องใช้ภาษาสคริปต์ที่ยุ่งยากซับซ้อนเหมือนก่อน

การติดตั้งโปรแกรม Dreamweaver 8

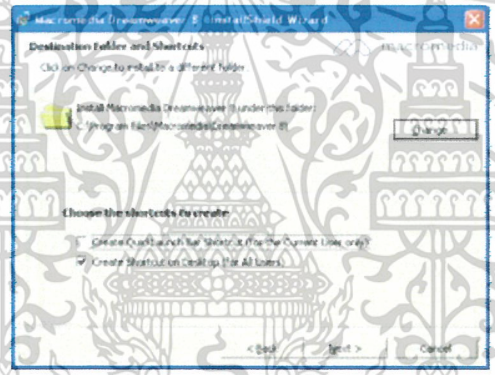
1. ใส่แผ่นซีดีรอม โปรแกรม Dreamweaver 8 เข้าไปในไดร์ฟซีดีรอม
2. โปรแกรมจะเริ่มทำงานโดยอัตโนมัติ หรือถ้าเปิดเข้าไปที่ไดรฟ์ซีดีรอม Double Click ที่ไอคอน Setup โปรแกรม
3. จะแสดงหน้าต่างข้อตกลงสิทธิการใช้งาน(License Agreement) เลือก I accept ... แล้ว Click ปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



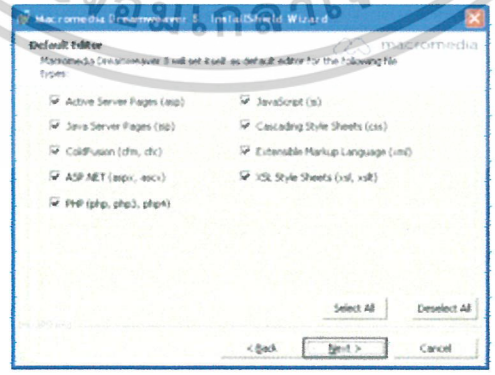
รูปที่ ก.15 หน้าต่างข้อตกลงสิทธิการใช้งาน

4. เลือกไดเรกทอรี่ที่ต้องการ เช่น C:\Program File ... ถ้าต้องการเปลี่ยนไดเรกทอรี่ Click ที่ปุ่ม Change แล้วกำหนดไดเรกทอรี่ใหม่ที่ต้องการ จากนั้น Click ที่ปุ่ม Next



รูปที่ ก.16 หน้าต่างเลือกไดเรกทอรี่ที่ต้องการเก็บโปรแกรม

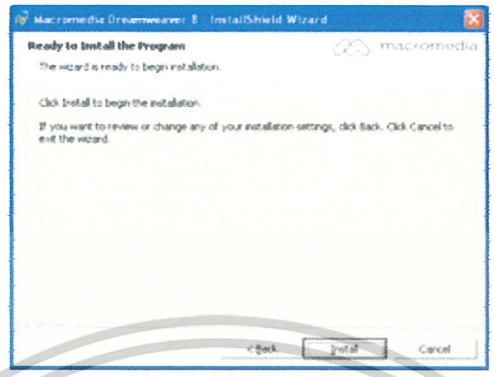
5. เลือกชนิดของโปรแกรมที่จะใช้ร่วมกับ Dreamweaver (Default Editor) ถ้าเลือกทั้งหมดแล้วคลิกปุ่ม Select All แล้ว Click ปุ่ม Next



รูปที่ ก.17 หน้าต่างเลือกโปรแกรมที่ต้องการใช้ร่วม

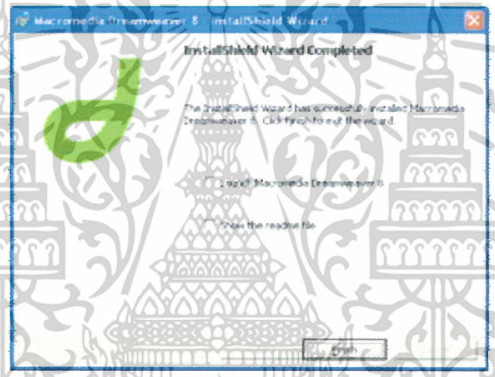
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. Click ปุ่ม Install เพื่อติดตั้งโปรแกรม



รูปที่ ก.18 หน้าต่างเลือกโปรแกรมที่ต้องการใช้ร่วม

7. เริ่มติดตั้งโปรแกรมลงในเครื่องจนเสร็จสมบูรณ์ Click ปุ่ม Finish



รูปที่ ก.19 หน้าต่างการติดตั้งโปรแกรมเสร็จสิ้น

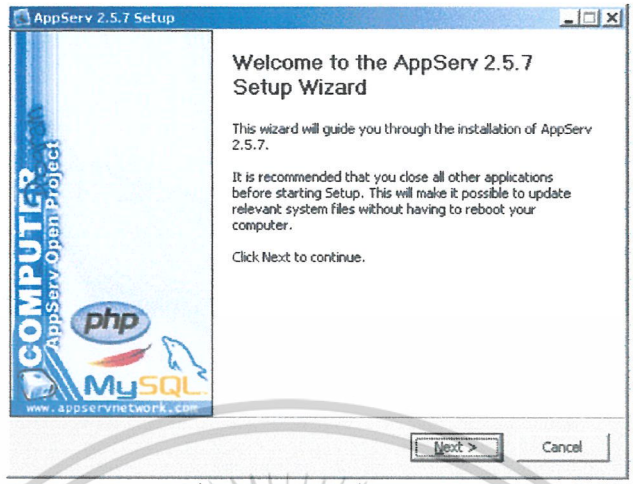
2. โปรแกรม AppServ

AppServ คือ โปรแกรมที่รวบรวมเอา Open Source Software หลายๆ อย่างมารวมกัน โดยมี Package หลักดังนี้ Apache, PHP ,MySQL, phpMyAdmin เพื่อลดขั้นตอนการติดตั้ง ทำให้การติดตั้งโปรแกรมต่างๆ ที่ได้กล่าวมาให้ง่ายขึ้น และรวดเร็วขึ้น ทั้งยังสามารถนำไปเป็น Web Server หรือ Database Server ได้ทันที

การติดตั้งโปรแกรม AppServ

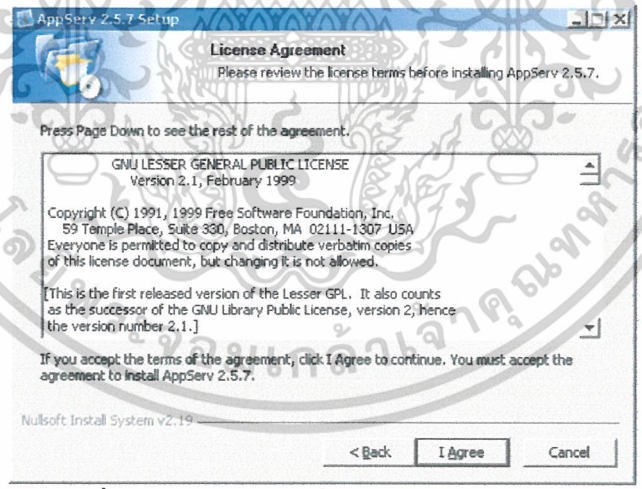
1. ดาวน์โหลดโปรแกรม AppServ จากเว็บไซต์ http://www.appservnetwork.com
2. Double Click ที่ appserv-win32-x.x.x.exe เพื่อทำการติดตั้ง จะปรากฏหน้าจอตามรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.20 ขั้นตอนการติดตั้งโปรแกรม AppServ

3. เข้าสู่ขั้นตอนเงื่อนไขการใช้งานโปรแกรม โดยโปรแกรม AppServ ได้แจกจ่ายในรูปแบบ GNU License หากผู้ติดตั้งอ่านเงื่อนไขต่างๆ เสร็จสิ้นแล้ว หากยอมรับเงื่อนไขให้ Click Next เพื่อเข้าสู่การติดตั้งในขั้นต่อไป แต่หากว่าไม่ยอมรับเงื่อนไข ให้กด Cancel เพื่อออกจากการติดตั้งโปรแกรม AppServ ดังรูปตัวอย่างที่ 2



รูปที่ ก.21 รายละเอียดเงื่อนไขการ GNU License

4. เข้าสู่ขั้นตอนการเลือกปลายทางที่ต้องการติดตั้ง หากต้องการเปลี่ยนปลายทางที่ติดตั้งให้กด Browse แล้วเลือกปลายทางที่ต้องการ เมื่อเลือกปลายทางเสร็จสิ้นให้ Click ปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เลือก Package Components ที่ต้องการติดตั้ง โดยค่าเริ่มต้นนั้นจะให้เลือกลงทุก Package แต่หากว่าผู้ใช้งานต้องการเลือกลงเฉพาะบาง Package ก็สามารถเลือกตามข้อที่ต้องการออก เมื่อทำการเลือก Package เรียบร้อยแล้ว ให้ Click Next โดยรายละเอียดแต่ละ Package มีดังนี้

Apache HTTP Server คือ โปรแกรมที่ทำหน้าเป็น Web Server

MySQL Database คือ โปรแกรมที่ทำหน้าเป็น Database Server

PHP Hypertext Preprocessor คือโปรแกรมประมวลผลการทำงานของภาษา PHP

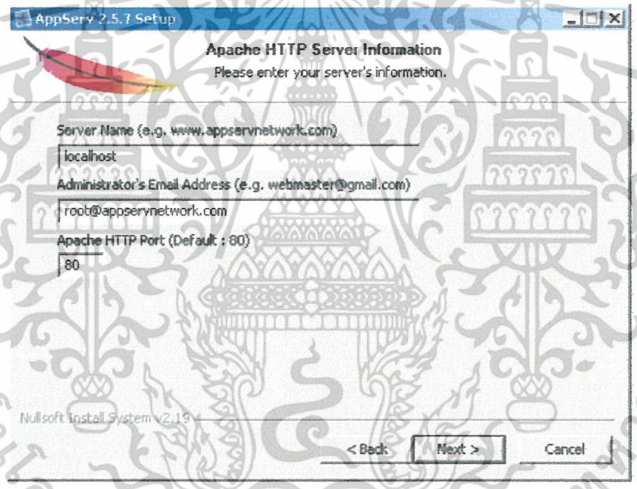
phpMyAdmin คือ โปรแกรมที่ใช้ในการบริหารจัดการฐานข้อมูล MySQL ผ่าเว็บไซต์

6. กำหนดค่าคอนฟิกของ Apache Web Server มีอยู่ด้วยกันทั้งหมด 3 ส่วน คือ

Server Name คือช่องป้อนข้อมูลชื่อ Web Server เช่น www.appservnetwork.com

Admin Email คือช่องสำหรับป้อนอีเมล์ผู้ดูแลระบบ เช่นroot@appservnetwork.com

HTTP Port คือช่องสำหรับระบุ Port ที่จะเรียกใช้งาน Apache Web Server



รูปที่ ก.22 กำหนดค่าคอนฟิกค่า Apache Web Server

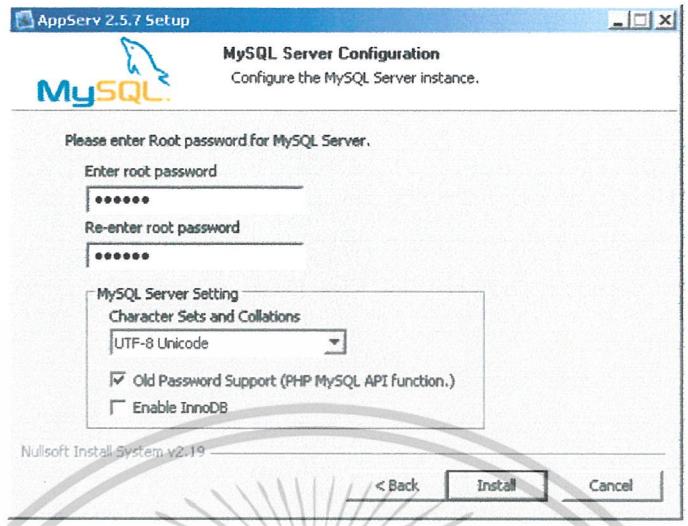
7. กำหนดค่าคอนฟิกของ MySQL Database มีอยู่ด้วยกันทั้งหมด 3 ส่วน คือ

Root Password คือช่องสำหรับป้อน รหัสผ่านการเข้าใช้งานฐานข้อมูลของ Root หรือผู้ดูแลระบบทุกครั้งที่เข้าใช้งานฐานข้อมูลในลักษณะที่เป็นผู้ดูแลระบบ ให้ระบุ user คือ root Character Sets ใช้ในการกำหนดค่าระบบภาษาที่ใช้ในการจัดเก็บฐานข้อมูล, เรียงลำดับฐานข้อมูล, Import ฐานข้อมูล, Export ฐานข้อมูล, ติดต่อฐานข้อมูล

Old Password ในกรณีมีปัญหาเกี่ยวกับการใช้งาน PHP กับ MySQL API เวอร์ชั่นเก่า ให้เลือกในส่วนของ Old Password

Enable InnoDB หากต้องการใช้งานฐานข้อมูลในรูปแบบ InnoDB ให้เลือกในส่วนนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.23 กำหนดค่าคอนฟิกของ MySQL Database

8. สิ้นสุดขั้นตอนการติดตั้งโปรแกรม AppServ สำหรับขั้นตอนสุดท้ายนี้จะมีให้เลือกว่าต้องการสั่งให้มีการรัน Apache และ MySQL ทันทีหรือไม่ จากนั้น Click ปุ่ม Finish เพื่อเสร็จสิ้นการติดตั้งโปรแกรม AppServ

รูปที่ ก.24 หน้าจอขั้นตอนสิ้นสุดการติดตั้งโปรแกรม AppServ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

คุณสมบัติของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติของระบบแจ้งเตือนการออกนอกพื้นที่บริเวณบ้านของเด็กและคนชรา

1. สามารถแจ้งเตือนโดยเสียงและไฟจาก LED ที่กล่องภาครับสัญญาณ เมื่อเด็กและคนชราออกนอกพื้นที่บริเวณบ้านส่งสัญญาณแจ้งเหตุฉุกเฉิน

2. สามารถแจ้งเตือนผ่านทางเว็บแอพพลิเคชัน เมื่อเด็กและคนชราออกนอกพื้นที่บริเวณบ้านส่งสัญญาณแจ้งเหตุฉุกเฉิน

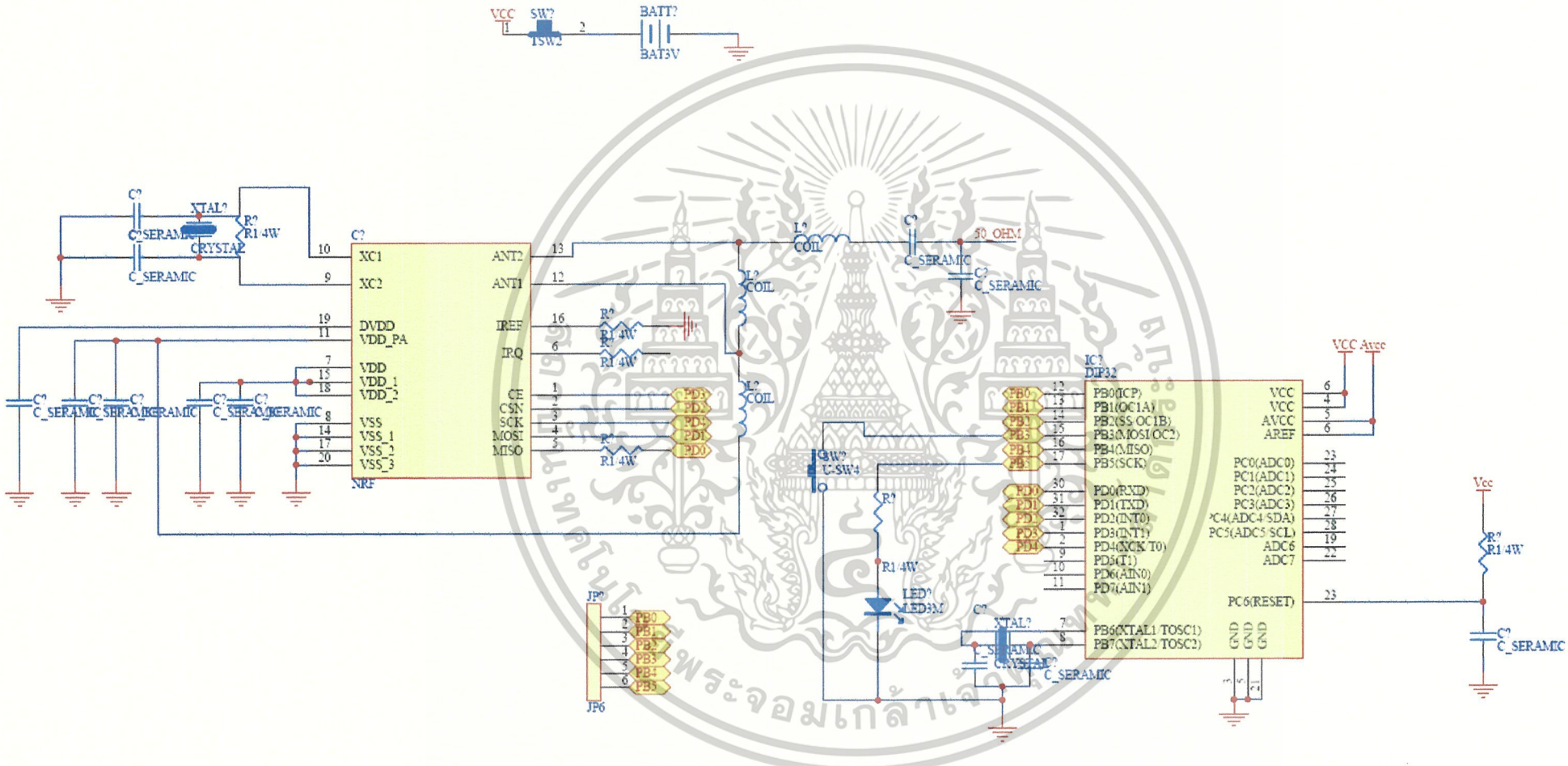
3. สามารถตรวจสอบข้อมูลย้อนหลังได้ผ่านทางเว็บแอพพลิเคชัน เพื่อตรวจสอบวันและเวลาที่เด็กและคนชราออกนอกพื้นที่การแจ้งและข้อมูลเหตุฉุกเฉิน และยังเอื้อประโยชน์เป็นสถิติสังเกตวันและเวลาที่เด็กและคนชราให้ง่ายต่อการควบคุมดูแลเด็กและคนชรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

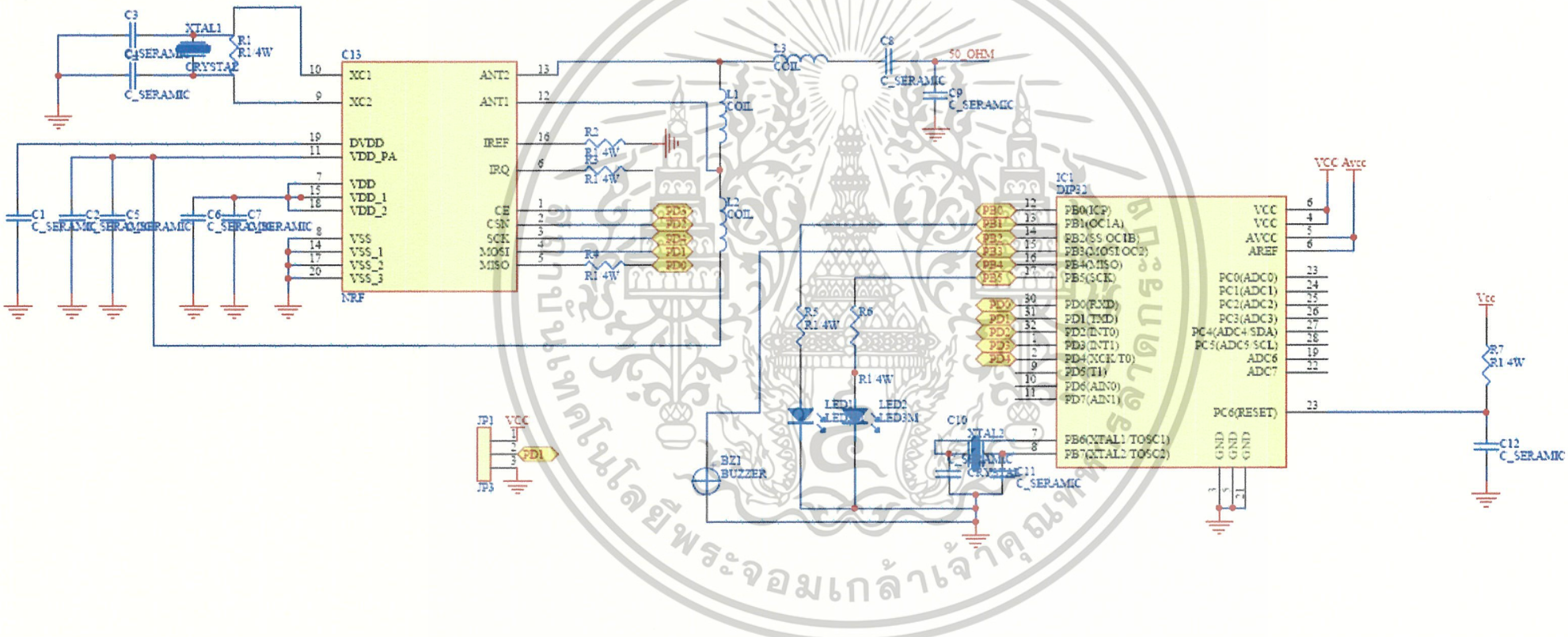
ภาคผนวก ค.

ลายวงจรของอุปกรณ์

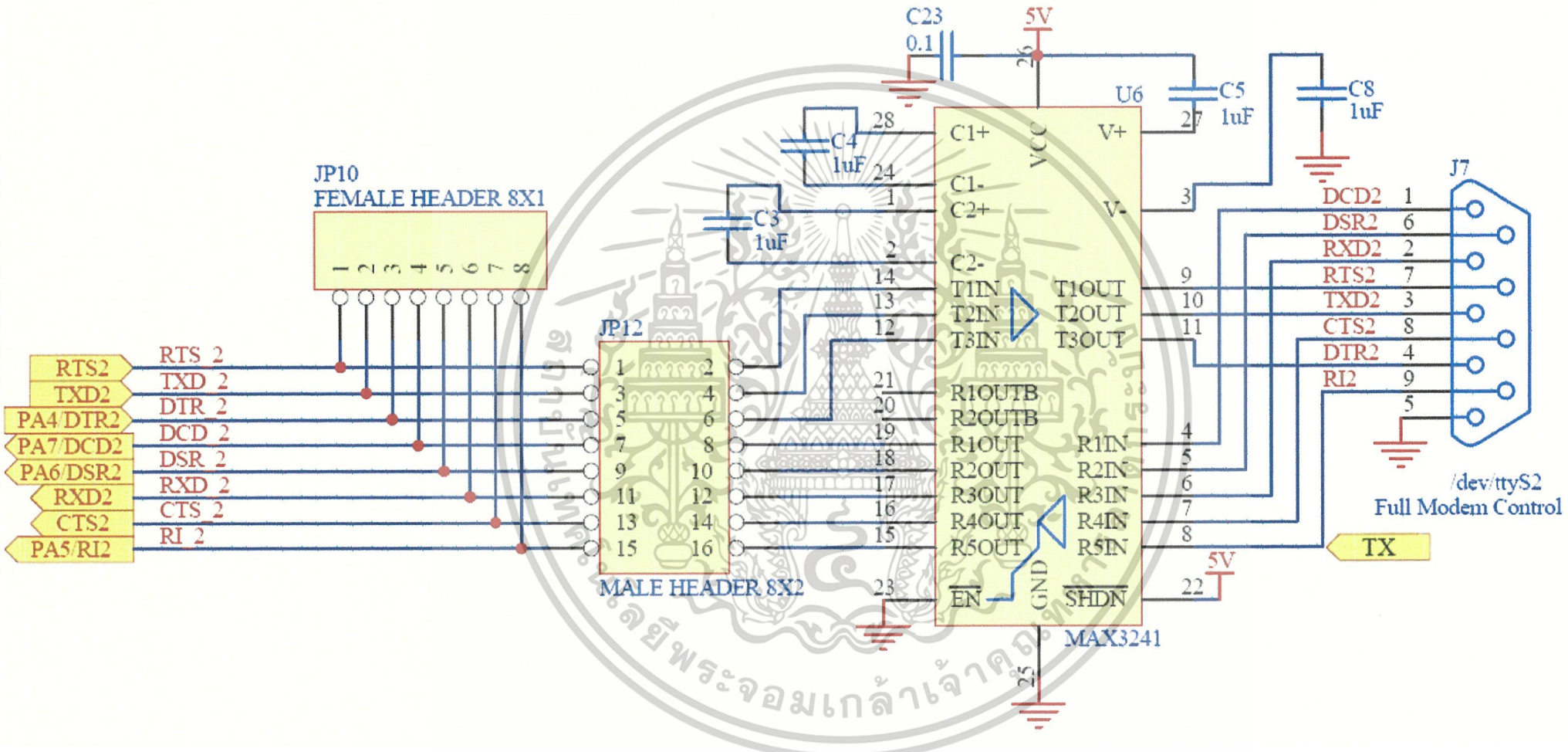
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



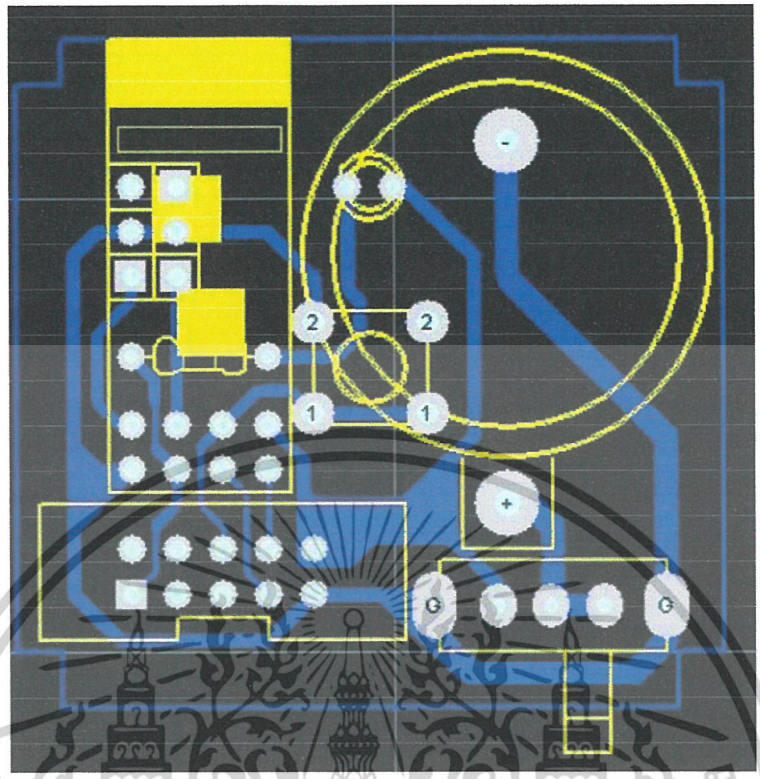
รูปที่ ค.1 ออกแบบวงจรภาคส่งสัญญาณไร้สาย



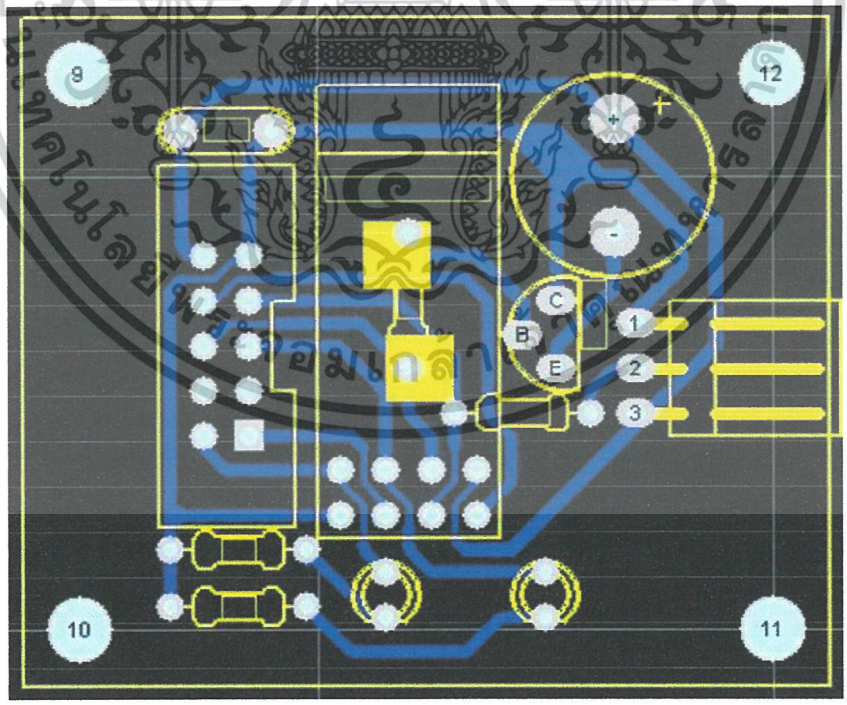
รูปที่ ค.2 ออกแบบวงจรภาครับสัญญาณไร้สาย



รูปที่ ค.3 ออกแบบวงจรการเชื่อมต่อภาครับสัญญาณกับ iBoard



รูปที่ ค.4 ลายวงจรภาคส่งสัญญาณไร้สาย



รูปที่ ค.5 ลายวงจรภาครับสัญญาณไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

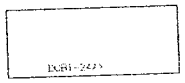
ภาคผนวก ง.

Datasheet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DCBT-24AX Low Power Small Size Programmable TRX Module

original size



component top side view

1	GND	2	I/O PB6 (Clock In)
3	I/O PC6 (Reset)	4	I/O PB1 (Clock Out)
5	I/O PB4 (MISO)	6	I/O PB3 (MOSI)
7	I/O PB5 (SCK)	8	VCC

* output power: 0, -6, -12, -18 dBm (1, 0.25, 0.063, 0.015 mW)
* antenna: multilayer chip +2 dBi gain
* weight: 1.28 gr
* size: 25mm x 10mm
* MCU: AVR 8-bit RISC ATmega168[i]
* LDO: 3.3 V
* transceiver IC: nRF24L01[ii]
* power supply: 2.7V to 5.25V DC
* active supply current:

	5 V, 8 MHz	5 V, 128 KHz
TX	17 mA	12.4 mA
RX	18 mA	13.4 mA

* idle supply current: 1.5 mA
* frequency: Worldwide 2.4GHz ISM band operation
84 channels (2.400 - 2.483 GHz)
* modulation: GFSK
* data rate: 2 Mbps or 1 Mbps
* sensitivity: -82 dBm at 2 Mbps
-85 dBm at 1 Mbps
* interface: SPI (Decibit command set)
* Development Kit: available

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Introduction

Thank you for your interest in Decibit transceiver modules.

You probably know Atmel's tiny AVR family, those MCU's with 8 pins. Think of Decibit modules being of same magnitude, 8 pins, similar pin functions, but with larger memory, more peripherals and of course RF capable. Decibit module series combines Atmel Corporation's ATmega168 AVR 8-bit RISC MCU with Nordic Semiconductor's latest nRF24L01 transceiver, together with wide range RF output power amplification options.

Module firmware provides a complete and very easy to use set of system functions and has three main features:

1) Easy Transmit / Receive functions to send data from one module to another.
2) Direct access to entire nRF24L01 register map and functions.
3) AVR and module specific system functions.

At the transmitting module, data to be sent is placed in SRAM buffer and `D_TX` system function is called. Similarly, on other end with receiving module, `D_RX` system function is called to receive a data packet to SRAM. Entire transmission is done for each direction using one single call instruction. Currently supported data length is 1..248 bytes in one go.

It is having an AVR MCU with direct RF connection to another AVR MCU, by a single system function CALL instruction. It can't get any simpler than this.

A sophisticated protocol is executed by firmware to have a fast, safe, complete and glitch free data exchange involving Nordic's Enhanced ShockBurst(TM) embedded baseband protocol engine. The total required amount of block transfers and acknowledgments will vary depending on RF environment and can not be predicted. These details are mentioned only for your information and do not matter while using these functions. Flexible timeout handling while TX (16 ms to 8 s) and RX (128 ms to 32 s) is provided for the case of a broken RF link.

Example code, sending module:

```
        ldi   r16,WD_int_2048   ;timeout 2 seconds
        call  D_WDT             ;watchdog timer control

        ldi   yh,high(IOBUF)    ;pointer to start of data in SRAM
        ldi   yl,low(IOBUF)
        ldi   r16,248           ;length of data, 1..248 bytes
        call  D_TX              ;transmit data
        cpi   r16,X_WDT         ;watchdog interrupt occurred ?
        breq  _lab0             ;branch if timed out

        ldi   r16,WD_off        ;disable WDT
        call  D_WDT             ;watchdog timer control
        ...

_lab0:
        ...
```


Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Example code, receiving module:

```
      ldi   yh,high(IOBUF)     ;pointer to start of data in SRAM
      ldi   yl,low(IOBUF)
      ;clr  r16                ;0:         no timeout
      ldi   r16,8              ;1..255:    timeout after n/8 seconds (1/8..32)
                               ;in this example timeout is 1 sec

      call  D_RX               ;system call to receive data block
      breq  _lab1              ;branch if timed out
      ...

_lab1:
      ...
```

Any watchdog setting is canceled while D_RX function call. WDT is turned off at function return and needs to re-enabled in case. As example:

```
_lab2:
      ldi   r16,WD_2048        ;watchdog reset in 2 seconds
      call  D_WDT              ;watchdog timer control
```

Simple, isn't it ?

Not yet mentioned is that some initialization has to be done before these functions are used. This is setting up desired RF channel, TX/RX addresses and few other settings like data rate and radio output power. Details can be seen from example source codes free to download from web and from development kit CD.

More information about the Development Kit can be found at link on page 29.

Although the above D_TX / D_RX functions cover most, if not all wireless data transmission needs, one might want to control the nRF24L01 directly and take the control 100% in own hands. The firmware provides all functions to do just that, with simple system function calls. As example it takes only two lines of code to set RF channel. The desired channel number goes in register R16 followed by the system call.

```
      ldi   r16,70             ;2.470 GHz
      call  W_RF_CH            ;Write RF Channel
```

It's only one line of code to read RF channel into register R16.

```
      call  R_RF_CH            ;Read RF Channel
```

System functions for previously hidden but then unveiled features of nRF24L01 are also implemented.

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Summary Of All Functions

1) Easy Transmit / Receive functions to send data from one module to another.

D_TX	Transmit data block
D_RX	Receive data block

2) Direct access to entire nRF24L01 register map and functions.

R_RX_PAYLOAD		Read RX Payload
R_RX_PL_WID		Read RX Payload Width
W_TX_PAYLOAD		Write TX Payload
W_TX_PAYLOAD_NO_ACK		Write TX Payload No ACK
W_ACK_PAYLOAD_P0		Write ACK Payload P0
W_ACK_PAYLOAD_P1		Write ACK Payload P1
W_ACK_PAYLOAD_P2		Write ACK Payload P2
W_ACK_PAYLOAD_P3		Write ACK Payload P3
W_ACK_PAYLOAD_P4		Write ACK Payload P4
W_ACK_PAYLOAD_P5		Write ACK Payload P5
N_FLUSH_TX		Flush TX FIFO
N_FLUSH_RX		Flush RX FIFO
N_REUSE_TX_PL		Reuse Last Sent Payload
N_NOOP		No Operation
N_ACTIVATE		Activate hidden features
R_CONFIG	W_CONFIG	Read / Write Configuration Register
R_EN_AA	W_EN_AA	Read / Write Enable Auto Acknowledgment Function
R_EN_RXADDR	W_EN_RXADDR	Read / Write Enabled RX Addresses
R_SETUP_AW	W_SETUP_AW	Read / Write Setup Of Address Widths
R_SETUP_RETR	W_SETUP_RETR	Read / Write Setup Of Automatic Retransmission
R_RF_CH	W_RF_CH	Read / Write RF Channel
R_RF_SETUP	W_RF_SETUP	Read / Write RF Setup Register
R_STATUS	W_STATUS	Read / Write Status Register
R_OBSERVE_TX		Read Transmit Observe Register
R_CDETECT		Read Carrier Detect
R_RX_ADDR_P0	W_RX_ADDR_P0	Read / Write Receive Address P0
R_RX_ADDR_P1	W_RX_ADDR_P1	Read / Write Receive Address P1
R_RX_ADDR_P2	W_RX_ADDR_P2	Read / Write Receive Address P2
R_RX_ADDR_P3	W_RX_ADDR_P3	Read / Write Receive Address P3
R_RX_ADDR_P4	W_RX_ADDR_P4	Read / Write Receive Address P4
R_RX_ADDR_P5	W_RX_ADDR_P5	Read / Write Receive Address P5
R_TX_ADDR	W_TX_ADDR	Read / Write Transmit Address
R_RX_PW_P0	W_RX_PW_P0	Read / Write Receive Payload Width P0
R_RX_PW_P1	W_RX_PW_P1	Read / Write Receive Payload Width P1
R_RX_PW_P2	W_RX_PW_P2	Read / Write Receive Payload Width P2
R_RX_PW_P3	W_RX_PW_P3	Read / Write Receive Payload Width P3
R_RX_PW_P4	W_RX_PW_P4	Read / Write Receive Payload Width P4
R_RX_PW_P5	W_RX_PW_P5	Read / Write Receive Payload Width P5
R_FIFO_STATUS		Read FIFO Status Register
R_EN_DYNPD	W_EN_DYNPD	Read / Write Enable Dynamic Payload Length
R_FEATURE	W_FEATURE	Read / Write Feature Register

3) AVR and module specific system functions.

D_CLK_OUT	Clock output control on CLK_OUT pin
D_WDT	Watchdog timer control
D_DELAY	Short delay (1000 cycles)
D_FW_ID	Read firmware ID
D_W_PAGE	Erase and write flash memory page
D_LDO_ON	Turn on LDO and enable 3V TRX power supply
D_LDO_OFF	Turn off LDO and shut down TRX section
D_ERASE	Erase user code (then can load new user code via SPI again)

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

How It Works

At power on or after a reset, system firmware is executed from $1C00 and general initialization is made (port pins, AVR I/O registers, stack) and user code is checked for validity. If found valid, then user code is run by a jump to $0000, else system firmware awaits new user code to be downloaded via SPI.

Entire SRAM ($100-4FF) is available for user code, except 32 bytes named RFBUF ($100-$11F) are reserved while nRF24L01 related function calls.

```
RFBUF:
  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 ;RFBUF[n]
                                                                                                ;1 byte payload
 HL                                                                                             ;2 byte payload
 Hx xL                                                                                          ;5 byte payload
" D  C  B  T  0"
 Hx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xL ;32 byte payload
```

Stack is set to RAMEND (..$4FF) by firmware and is shared with user code. Maximum of 24 bytes might be pushed to stack while system function calls.

Interrupt vectors are placed in user code section. Interrupts can be used in user code as usual but are automatically disabled by AVR hardware when system functions are executing from system firmware code section. Global interrupt enable and sleep instructions are placed in user code section, which is needed for interrupt response while accessing the nRF24L01 transceiver. Thus user code interrupts are not blocked while RF data transfer or nRF24L01 register access, since most of the time the AVR will be in idle mode waiting for the nRF24L01 to complete it's job.

```
INT_JMP:                ;placed within user code section
      sei               ;set global interrupt flag
      sleep             ;wait for interrupt
      ret
```

System function calls do not modify any registers. R0 to R31 are untouched, except that register R16 holds the resulting byte for most function calls. For those nRF24L01 commands with no data bytes, the nRF24L01 STATUS byte is returned in R16 and RFBUF is not modified. nRF24L01 commands with one or more bytes data reading or writing result the last data byte in R16 and always modify RFBUF in same amount of bytes. As example for one byte data reading, R16 and first location of RFBUF contain the same resulting byte. D_TX and D_RX functions modify RFBUF. Status code for D_TX is in R16. Interrupt routines used by system function calls (PC_INT1, WDT_INT and USART_RXC) do not modify any registers or the SREG.

Last EEPROM location ($1FF) is used to validate user code. If this location has byte $FF, then user code is not existent. In this case (after a reset or jump to $1C00) system firmware takes over and awaits for new user code to be downloaded via SPI. All other EEPROM locations are available for user code. Even location $1FF can be used if the stored value is in range $00..$FE and not $FF. User code validity is checked at each reset of module (watchdog, brown out, external, power on).

Flash program memory is divided in two sections, user code section $0000-$1BFF (112 pages) and system code section $1C00-$1FFF (16 pages). Page size is 128 bytes or 64 code words. User program code can erase and write all user program memory pages by using `D_W_PAGE` system function call. Reading program memory is done with LPM instruction. System code section is not read / write accessible. It is advised not to accidentally erase active user program code, thus locking up the module, unless if EEPROM location $1FF is $FF. At the end of system program memory is the location of jump table for system function calls.

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

There is no LPM instruction used within system code. Instead, it is placed within user code.

```
LPM_JMP:                        ;placed within user code section
      lpm   r16,z               ;LPM instruction executed from user code
      ret
```

I/O port pins (PB1,PB3,PB4,PB5,PB6) are initially and after each reset high-Z inputs. (PB7) is read only. (PC6) is the RESET pin and not an I/O port pin on the modules with configuration option "C" and "D". Do not access or modify any other port pins. It will cause excessive current flow when a pin tied to GND is set as push pull high output. (PB2) is tied to GND, which is SPI slave select pin (!SS).

These ATmega168 peripherals and interrupts are not usable due to module design:

```
jmp   I_VOID          ;IRQ0 Handler                              (n/a)
jmp   I_VOID          ;IRQ1 Handler                              (n/a)
jmp   I_VOID          ;PCINT2 Handler                            (n/a)
jmp   I_VOID          ;USART, RX Complete Handler(*)             (n/a)
jmp   I_VOID          ;USART, UDR Empty Handler                  (n/a)
jmp   I_VOID          ;USART, TX Complete Handler                (n/a)
jmp   I_VOID          ;ADC Conversion Complete Handler           (n/a)
jmp   I_VOID          ;Analog Comparator Handler                 (n/a)
jmp   I_VOID          ;2-wire Serial Interface Handler           (n/a)
jmp   I_VOID          ;Store Program Memory Ready Handler        (n/a)
```

These peripherals and interrupts can be used:

```
jmp   I_PCINT0        ;PCINT0 Handler
                      ;(use only for PB1,PB3,PB4,PB5,PB6,PB7)
jmp   J_PCINT1        ;PCINT1 Handler(*)    (use only for PC6)
jmp   J_WDT           ;Watchdog Timer Handler(**)
jmp   I_TIM2_COMPA    ;Timer2 Compare A Handler
jmp   I_TIM2_COMPB    ;Timer2 Compare B Handler
jmp   I_TIM2_OVF      ;Timer2 Overflow Handler
jmp   I_TIM1_CAPT     ;Timer1 Capture Handler
jmp   I_TIM1_COMPA    ;Timer1 Compare A Handler
jmp   I_TIM1_COMPB    ;Timer1 Compare B Handler
jmp   I_TIM1_OVF      ;Timer1 Overflow Handler
jmp   I_TIM0_COMPA    ;Timer0 Compare A Handler
jmp   I_TIM0_COMPB    ;Timer0 Compare B Handler
jmp   I_TIM0_OVF      ;Timer0 Overflow Handler
jmp   I_SPI_STC       ;SPI Transfer Complete Handler
jmp   I_EE_RDY        ;EEPROM Ready Handler
```

(*) used by firmware while nRF24L01 is accessed
(**) used while D_RX system function call.

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Transceiver Power Supply Control

LDO output can be disabled and thus power supply to TRX section on module completely shut off by calling system function `D_LDO_OFF`. To turn on LDO output and allow power supply to TRX section, call system function `D_LDO_ON`. LDO output is turned off by default at reset or start up and nRF24L01 registers need to be re-initialized.

```
call  D_LDO_OFF          ;shut down TRX section
call  D_LDO_ON           ;enable 3V TRX power supply
```

Power down of nRF24L01 alone consumes 900 nA typical. Using LDO to shut down TRX part reduces this virtually to zero. Only left is AVR power down current <1 µA typ. @ 3 V for DCBT-24R6 with BOD (brown out detector) disabled. All other modules have BOD enabled at 2.7V and consume more power while in power down mode.

Reading LDO Status

If `(PB7)` is HIGH, then LDO is regulating and TRX section voltage is the LDO output voltage. If `(PB7)` is LOW, then LDO is not regulating, but letting input voltage through to TRX section (voltage follower), which is below nominal LDO output level, down to until 2.5 V. LDO can not function below 2.5 V.

```
label:
      sbis  pinb,pinb7        ;LDO error ?
      rjmp  label             ;yes
                              ;wait until output level is 3 or 3.3 Volt
```

DCBT-24R6 has different type LDO, which lets coin cell voltage through down to 1.8 V and `(PB7)` is permanently tied to GND.

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Clock Output Function

The output signal is push pull output of 50:50 square wave. A reset will cancel this function and the pin will be high-Z input.

```
ldi   r16,DIV_4   ;system clock / 4
call  D_CLK_OUT   ;clock output control
                  ;if system clock is 8 MHz
                  ;then CLK_OUT is 2 MHz signal


ldi   r16,DIV_0   ;disabled
call  D_CLK_OUT   ;clock output control
                  ;PB1 is high-Z input
```

Ready defined divisors are as follows:

```
DIV_0             ;clock output disabled
DIV_2             ;system clock / 2
DIV_4             ;system clock / 4
DIV_6             ;system clock / 6
...
DIV_24            ;system clock / 24
```

More division can be achieved by formula:

```
ldi   r16,n       ;system clock / ((n+1)*2)
                  ;n can not be 255
                  ;only valid for for 0..254
call  D_CLK_OUT   ;clock output control
                  ;a value of 249 will result as
                  ;8 MHz / 500 = 16 KHz
```

Clocking Options

External oscillator up to 20 MHz

Internal RC oscillator @ 8 MHz
- SW tunable typically from 3 MHz to 19 MHz (not recommended by Atmel)
- Eliminating need for external oscillator

The AVR System Clock Prescaler can be invoked to reduce system clock up to 1/256 in real time.

Modules do not allow usage of quartz and crystals.

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Function Options

Modules are available with these pin function options.

Option "A"

- 6 general purpose I/O pins
- module runs with internal RC oscillator (8 MHz)

```
1  GND
2  I/O PB6
3  I/O PC6
4  I/O PB1 (CLOCK OUT)
5  I/O PB4 (MISO)
6  I/O PB3 (MOSI)
7  I/O PB5 (SCK)
8  VCC
```

Option "B"

- 5 general purpose I/O pins
- clock input pin, module runs with external clock signal

```
1  GND
2  XTAL1
3  I/O PC6
4  I/O PB1 (CLOCK OUT)
5  I/O PB4 (MISO)
6  I/O PB3 (MOSI)
7  I/O PB5 (SCK)
8  VCC
```

Option "C"

- 5 general purpose I/O pins
- active low reset pin
- module runs with internal RC oscillator (8 MHz)

```
1  GND
2  I/O PB6
3  !RESET
4  I/O PB1 (CLOCK OUT)
5  I/O PB4 (MISO)
6  I/O PB3 (MOSI)
7  I/O PB5 (SCK)
8  VCC
```

Option "D"

- 4 general purpose I/O pins
- active low reset pin
- clock input pin, module runs with external clock signal

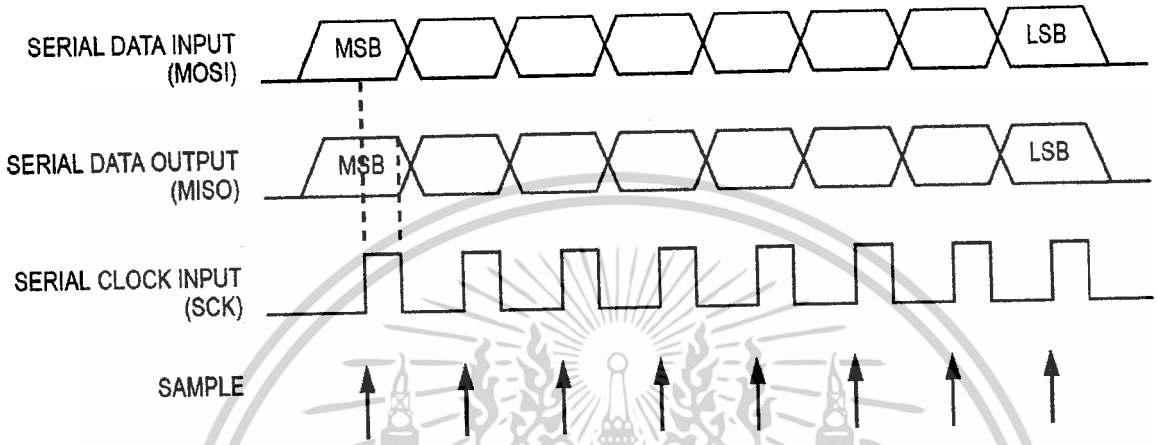
```
1  GND
2  XTAL1
3  !RESET
4  I/O PB1 (CLOCK OUT)
5  I/O PB4 (MISO)
6  I/O PB3 (MOSI)
7  I/O PB5 (SCK)
8  VCC
```

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Memory Programming Using Serial Peripheral Interface (SPI)

A regular AVR is programmed via SPI and in system with RESET pin pulled to GND using Atmel SPI command set. Decibit modules are programmed similar, through same serial interface, but under control of system firmware using different set of SPI commands. RESET pin must be left unconnected or pulled HIGH for configuration options "C" and "D". The module is a SPI slave. Pin PB2 (!SS, slave select) is tied to GND on the modules.



Both EEPROM and user program code memories are read and written under firmware control using Decibit SPI command set. There are eight SPI commands.

```
$D7 53 00 00        ;programming enable
$AA D7 53 00        ;response
```

This command must be executed initially for synchronization and to enable memory programming. A maximum of 32 tries is needed to synchronize SPI clock. Add one extra clock pulse between each try. Correct reading will be a $53 while issuing the 3rd command byte. All four command bytes have to be sent completely, even if not yet synchronized.

```
$D1 XX HH LL        ;load code word to temporary SRAM buffer
$AA D1 XX HH        ;response
```

XX Code word index ($00..$3F) within a page (64 code words, 128 bytes).
HHLL Code word itself.

Empty code word locations ($FFFF) do not have to be loaded to temporary page buffer.

```
$D2 00 00 PP        ;write page to flash memory
$AA D2 00 00        ;response
```

PP Page ($00..$6F) within user program memory (112 pages).

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
$D5 00 00 PP          ;select flash memory page for read
$AA D5 00 00          ;response
```

PP Page ($00..$6F) in flash memory.

```
$DB XX 00 00          ;read code word
$AA DB HH LL          ;response
```

XX Code word index ($00..$3F) within a page (64 code words, 128 bytes).
HHLL Code word itself.

```
$D3 0X XX 00          ;read EEPROM
$AA D3 0X ZZ          ;response
```

XXX EEPROM address ($000..$1FF).
ZZ Data byte read from EEPROM.

```
$D4 0X XX YY          ;write EEPROM
$AA D4 0X XX          ;response
```

XXX EEPROM address ($000..$1FF).
YY Data byte to be written to EEPROM.

After successful download of user program code, the last EEPROM location must be written with a data byte other than $FF to enable and execute user code at next reset or power up.

```
$D0 00 00 00          ;no operation
$AA D0 00 00          ;response
```

While memory is being written, polling with the "no operation" command will detect end of a write operation. Every SPI command will echo $AA while issuing the 1st command byte. If a different value was received, then the issued command was not executed by firmware, because a write operation is in progress. In such case, retry the same command.

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unsigned long int dcbt_SPI(unsigned long int);        //4 byte data exchange via SPI


int dcbt_wf()  //Write FLASH Pages
{
  unsigned int i, j;

  for (i = 0; i < 112; i++)
  {
    for (j = 0; j < 64; j++)
    {
      while (((dcbt_SPI(LOAD_PAGE | (j << 16) | (PMEM[i*64+j] & 0xFFFF)) >> 24) & 0xFF) != 0xAA) {}
    }
    while (((dcbt_SPI(WRITE_PAGE | i) >> 24) & 0xFF) != 0xAA) {}
    while (((dcbt_SPI(NO_OPERATION) >> 24) & 0xFF) != 0xAA) {}
  }
}


int dcbt_rf()  //Read FLASH Pages
{
  unsigned int i, j;
  unsigned long int k;

  for (i = 0; i < 112; i++)
  {
    while (((dcbt_SPI(SET_PAGE | i) >> 24) & 0xFF) != 0xAA) {}

    for (j = 0; j < 64; j++)
    {
      do
      {
        k = dcbt_SPI(READ_WORD | (j << 16));
      }
      while (((k >> 24) & 0xFF) != 0xAA);

      printf("%04X ", k & 0xFFFF);
    }
  }
}


int dcbt_we()  //Write EEPROM
{
  unsigned int i;

  for (i = 0; i < 0x1FF; i++)
  {
    while (((dcbt_SPI(WRITE_EEPROM | (i << 8) | (EMEM[i] & 0xFF)) >> 24) & 0xFF) != 0xAA) {}
    while (((dcbt_SPI(NO_OPERATION) >> 24) & 0xFF) != 0xAA) {}
  }
}


int dcbt_re()  //Read EEPROM
{
  unsigned int i;
  unsigned long int j;

  for (i = 0; i < 0x1FF; i++)
  {
    do
    {
      j = dcbt_SPI(READ_EEPROM | (i << 8));
    }
    while (((j >> 24) & 0xFF) != 0xAA);

    printf("%02X ", j & 0xFF);
  }
}
```


Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tips Regarding User Code Erasure

During code development, a front door needs to be kept open to be able to download new user code into module, even if user code fails.

Easiest method is to call D_ERASE system function right at beginning of user code. This way, user code can run until a reset or as long as power is applied to module. After a restart, new code can be downloaded right away. The catch is, that a spurious reset, as example removing module from programmer socket, will invalidate user code. Only configuration options "C" and "D" are affected from being reset prematurely, options "A" and "B" do not have a RESET pin. To prevent this, USB plug must be carefully removed from programmer before module is removed. Keep reset button pressed while inserting module in the hand held tester. Or, batteries from hand tester need to be removed before inserting module to it's socket. And finally battery insertion must be done carefully, so that module does not startup more then once.

If user code must be kept and guaranteed to be valid after a reset or power on, then a possibility could be using a trigger to execute D_ERASE system function with a level change on an I/O pin. As example, if a pulled up pin is pushed to GND. User code would check at beginning if the given I/O pin is HIGH and then would invalidate itself if detected as LOW. All example user codes found on Development Kit CD make use of this type trigger while using the hand-held-tester. See DDK documentation for details.

To achieve the same result without using I/O pins, an EEPROM counter could be used. At each startup, the counter is incremented. After reaching a certain number, user code would invalidate itself.

Once code development has reached final phase, then more convenient methods would be to send a special command or even perform a PIN code check within user application.

If user application is final and does not require an update option, then this system function call can be removed completely from user code.

D_ERASE system function erases last EEPROM location to be $FF.

Transmission Duration

Using ShockBurst(TM), transferring 1 byte payload @ 2 Mbps takes 0.17 ms time, with acknowledgment 0.33 ms. Transfer time for 32 bytes is 0.29 ms, with acknowledgment 0.45 ms.

Final Note

Spread the word around please, this is THE solution to add wireless link capability to a product. Other RF products and solutions may be just too complex, expensive, bigger size, power hungry or based on older technology. No manufacturer other than Decibit is currently providing this level of function and ease of use for a wireless link for the 2.4 GHz ISM band.

The user codes are very short and thus easy to understand. It takes no more than 10 minutes to unpack DDK, install the software, install the USB driver, load codes on modules and get a working link between two hand-held-testers. In contrary to performance and ease of use, DDK is offered for 45 EUR as cheapest on the market than any other competitor, ideal for universities, schools and the hobbyist.

Remember, your code runs in the MCU that is on the module!

Copyright © 2007-2008 Decibit Co., Ltd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

High Performance, Low Power AVR® 8-Bit Microcontroller

Advanced RISC Architecture

- 131 Powerful Instructions – Most Single Clock Cycle Execution
- 32 x 8 General Purpose Working Registers
- Fully Static Operation
- Up to 20 MIPS Throughput at 20 MHz
- On-chip 2-cycle Multiplier

Non-volatile Program and Data Memories

- 4/8/16K Bytes of In-System Self-Programmable Flash (ATmega48/88/168)
  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
- 256/512/512 Bytes EEPROM (ATmega48/88/168)
  Endurance: 100,000 Write/Erase Cycles
- 512/1K/1K Byte Internal SRAM (ATmega48/88/168)
- Programming Lock for Software Security

Peripheral Features

- Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
- One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
- Real Time Counter with Separate Oscillator
- Six PWM Channels
- 8-channel 10-bit ADC in TQFP and QFN/MLF package
- 6-channel 10-bit ADC in PDIP Package
- Programmable Serial USART
- Master/Slave SPI Serial Interface
- Byte-oriented 2-wire Serial Interface
- Programmable Watchdog Timer with Separate On-chip Oscillator
- On-chip Analog Comparator
- Interrupt and Wake-up on Pin Change

Special Microcontroller Features

- Power-on Reset and Programmable Brown-out Detection
- Internal Calibrated Oscillator
- External and Internal Interrupt Sources
- Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby

I/O and Packages

- 23 Programmable I/O Lines
- 28-pin PDIP, 32-lead TQFP and 32-pad QFN/MLF

Operating Voltage:

- 1.8 - 5.5V for ATmega48V/88V/168V
- 2.7 - 5.5V for ATmega48/88/168

Temperature Range:

- -40°C to 85°C

Speed Grade:

- ATmega48V/88V/168V: 0 - 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V
- ATmega48/88/168: 0 - 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V

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

ATmega48/V
ATmega88/V *
ATmega168/V *

* Preliminary

Rev. 2545FS–AVR–06/05

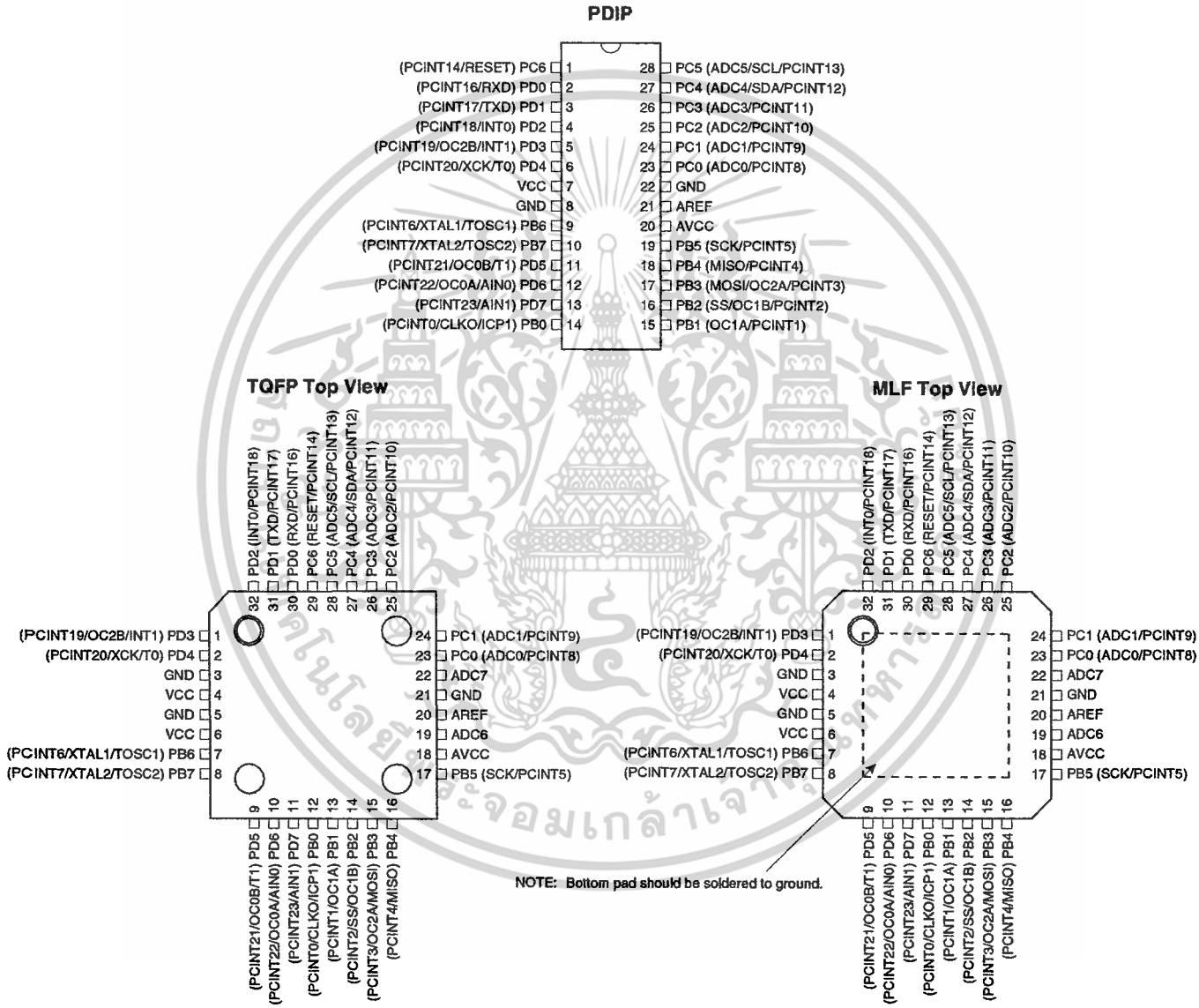
Note: This is a summary document. A complete document is available on our Web site at www.atmel.com.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Low Power Consumption**
  - **Active Mode:**
    - **1 MHz, 1.8V: 240µA**
    - **32 kHz, 1.8V: 15µA (including Oscillator)**
  - **Power-down Mode:**
    - **0.1µA at 1.8V**

1. Pin Configurations

Figure 1-1. Pinout ATmega48/88/168



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 Disclaimer

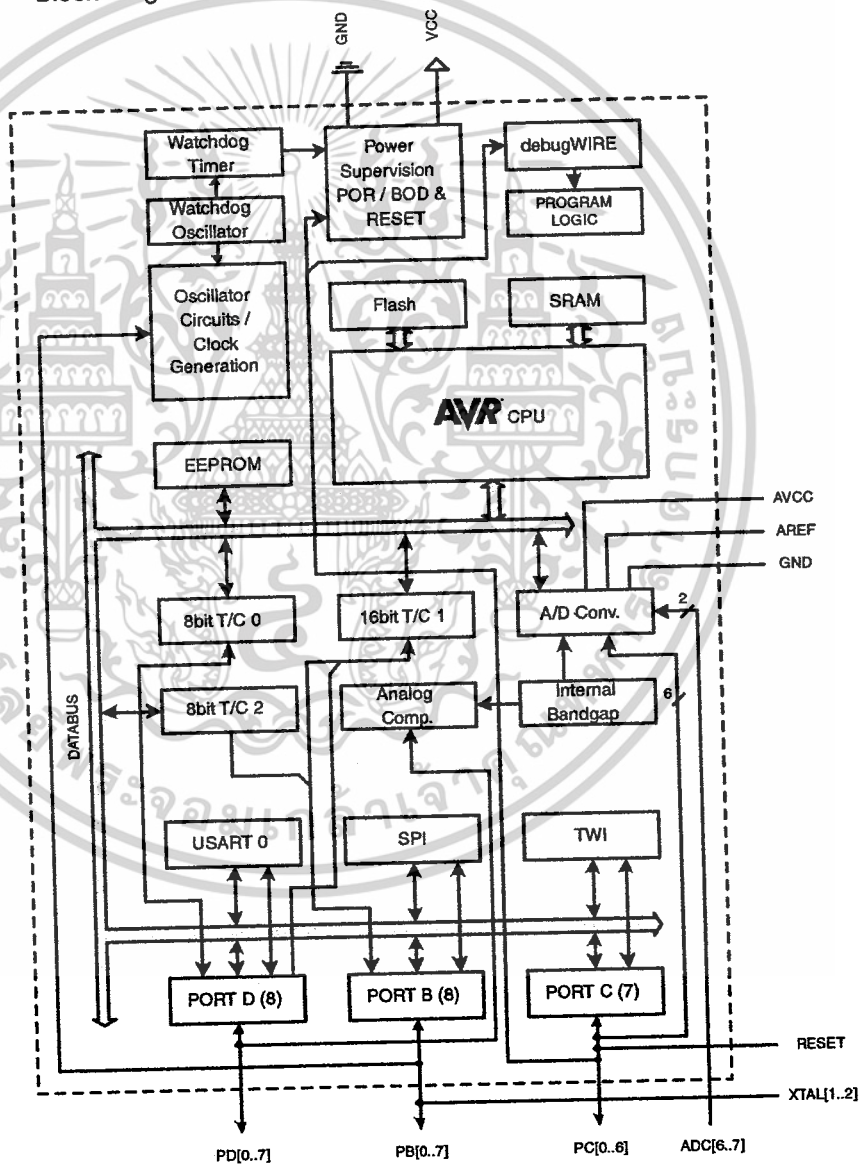
Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

2. Overview

The ATmega48/88/168 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 ATmega48/88/168 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

2.1 Block Diagram

Figure 2-1. Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

The ATmega48/88/168 provides the following features: 4K/8K/16K bytes of In-System Programmable Flash with Read-While-Write capabilities, 256/512/512 bytes EEPROM, 512/1K/1K bytes SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte-oriented 2-wire Serial Interface, an SPI serial port, a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable Watchdog Timer with internal Oscillator, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, USART, 2-wire Serial Interface, 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 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.

The device is manufactured using Atmel's high density non-volatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile 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 ATmega48/88/168 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega48/88/168 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.

2 Comparison Between ATmega48, ATmega88, and ATmega168

The ATmega48, ATmega88 and ATmega168 differ only in memory sizes, boot loader support, and interrupt vector sizes. Table 2-1 summarizes the different memory and interrupt vector sizes for the three devices.

Table 2-1. Memory Size Summary

Device	Flash	EEPROM	RAM	Interrupt Vector Size
ATmega48	4K Bytes	256 Bytes	512 Bytes	1 instruction word/vector
ATmega88	8K Bytes	512 Bytes	1K Bytes	1 instruction word/vector
ATmega168	16K Bytes	512 Bytes	1K Bytes	2 instruction words/vector

ATmega88 and ATmega168 support a real Read-While-Write Self-Programming mechanism. There is a separate Boot Loader Section, and the SPM instruction can only execute from there.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

In ATmega48, there is no Read-While-Write support and no separate Boot Loader Section. The SPM instruction can execute from the entire Flash.

2.3 Pin Descriptions

2.3.1 VCC

Digital supply voltage.

2.3.2 GND

Ground.

2.3.3 Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

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.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in "Alternate Functions of Port B" on page 76 and "System Clock and Clock Options" on page 25.

2.3.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5..0 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.

2.3.5 PC6/$\overline{\text{RESET}}$

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a 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 9-1 on page 44. Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in "Alternate Functions of Port C" on page 80.

2.3.6 Port D (PD7:0)

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

The various special features of Port D are elaborated in "Alternate Functions of Port D" on page 83.

2.3.7 AV_{CC}

AV_{CC} is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. 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. Note that PC6..4 use digital supply voltage, V_{CC}.

2.3.8 AREF

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

2.3.9 ADC7:6 (TQFP and QFN/MLF Package Only)

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

3. Resources

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

.3 ATmega168

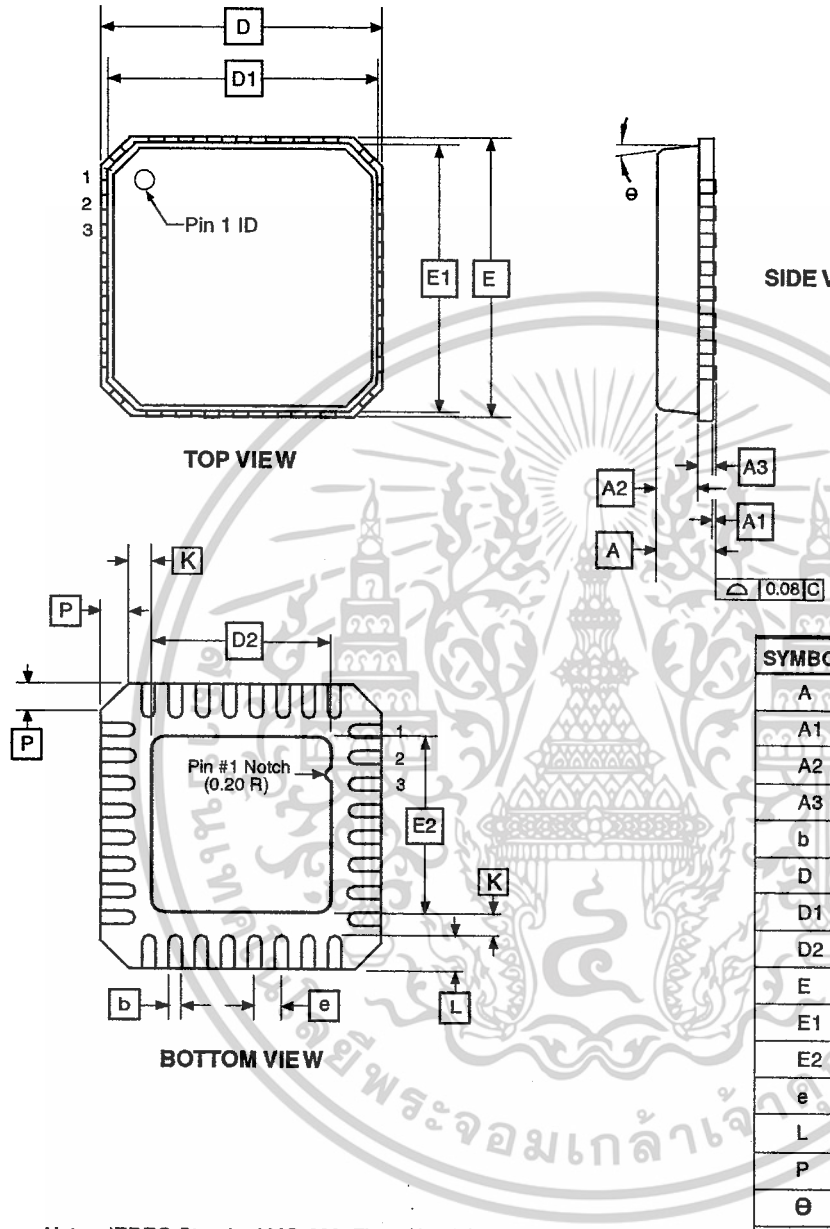
Speed (MHz)[3]	Power Supply	Ordering Code	Package[1]	Operational Range
10	1.8 - 5.5	ATmega168V-10AI ATmega168V-10PI ATmega168V-10MI ATmega168V-10AU[2] ATmega168V-10PU[2] ATmega168V-10MU[2]	32A 28P3 32M1-A 32A 28P3 32M1-A	Industrial (-40°C to 85°C)
20	2.7 - 5.5	ATmega168-20AI ATmega168-20PI ATmega168-20MI ATmega168-20AU[2] ATmega168-20PU[2] ATmega168-20MU[2]	32A 28P3 32M1-A 32A 28P3 32M1-A	Industrial (-40°C to 85°C)

lote:
1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
2. Pb-free packaging alternative, complies to the European Directive for Restriction of Hazardous Substances (RoHS directive).Also Halide free and fully Green.
3. See Figure 27-2 on page 303 and Figure 27-3 on page 303.

Package Type	
32A	32-lead, Thin (1.0 mm) Plastic Quad Flat Package (TQFP)
28P3	28-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
32M1-A	32-pad, 5 x 5 x 1.0 body, Lead Pitch 0.50 mm Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

.3 32M1-A



COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	0.80	0.90	1.00	
A1	–	0.02	0.05	
A2	–	0.65	1.00	
A3		0.20 REF		
b	0.18	0.23	0.30	
D		5.00 BSC		
D1		4.75 BSC		
D2	2.95	3.10	3.25	
E		5.00 BSC		
E1		4.75BSC		
E2	2.95	3.10	3.25	
e		0.50 BSC		
L	0.30	0.40	0.50	
P	–	–	0.60	
θ	–	–	12°	
K	0.20	–	–	

Note: JEDEC Standard MO-220, Fig. 2 (Anvil Singulation), VHHD-2.

8/19/04

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**32M1-A**, 32-pad, 5 x 5 x 1.0 mm Body, Lead Pitch 0.50 mm, 3.10 mm Exposed Pad, Micro Lead Frame Package (MLF)	32M1-A	D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NORDIC SEMICONDUCTOR

nRF24L01
Single Chip 2.4GHz Transceiver

Product Specification

Key Features

- Worldwide 2.4GHz ISM band operation
- Up to 2Mbps on air data rate
- Ultra low power operation
- 11.3mA TX at 0dBm output power
- 12.3mA RX at 2Mbps air data rate
- 900nA in power down
- 22µA in standby-I
- On chip voltage regulator
- 1.9 to 3.6V supply range
- Enhanced ShockBurst™
- Automatic packet handling
- Auto packet transaction handling
- 6 data pipe MultiCeiver™
- Air compatible with nRF2401A, 02, E1 and E2
- Low cost BOM
- ±60ppm 16MHz crystal
- 5V tolerant inputs
- Compact 20-pin 4x4mm QFN package

Applications

- Wireless PC Peripherals
- Mouse, keyboards and remotes
- 3-in-one desktop bundles
- Advanced Media center remote controls
- VoIP headsets
- Game controllers
- Sports watches and sensors
- RF remote controls for consumer electronics
- Home and commercial automation
- Ultra low power sensor networks
- Active RFID
- Asset tracing systems
- Toys

All rights reserved.
Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.
July 2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

All application information is advisory and does not form part of the specification.

Limiting values

Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the specifications are not implied. Exposure to limiting values for extended periods may affect device reliability.

Life support applications

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

Data sheet status	
Objective product specification	This product specification contains target specifications for product development.
Preliminary product specification	This product specification contains preliminary data; supplementary data may be published from Nordic Semiconductor ASA later.
Product specification	This product specification contains final product specifications. Nordic Semiconductor ASA reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.

Contact details

Visit www.nordicsemi.no for Nordic Semiconductor sales offices and distributors worldwide

Main office:

Otto Nielsens vei 12
7004 Trondheim
Phone: +47 72 89 89 00
Fax: +47 72 89 89 89
www.nordicsemi.no

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Writing Conventions

This product specification follows a set of typographic rules that makes the document consistent and easy to read. The following writing conventions are used:

- Commands, bit state conditions, and register names are written in `Courier`.
- Pin names and pin signal conditions are written in **`Courier bold`**.
- Cross references are underlined and highlighted in blue.

Revision History

Date	Version	Description
July 2007	2.0	• Restructured layout in a new template • Added details of the following features: ▸ Dynamic Payload Length (DPL) ▸ Acknowledgement Payload (`ACK_PLD`) ▸ Feature register ▸ ACTIVATE SPI command ▸ Selective Auto Acknowledgement (`NO_ACK`)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 Introduction

The nRF24L01 is a single chip 2.4GHz transceiver with an embedded baseband protocol engine (Enhanced ShockBurst™), designed for ultra low power wireless applications. The nRF24L01 is designed for operation in the world wide ISM frequency band at 2.400 - 2.4835GHz. An MCU (microcontroller) and very few external passive components are needed to design a radio system with the nRF24L01.

The nRF24L01 is configured and operated through a Serial Peripheral Interface (SPI.) Through this interface the register map is available. The register map contains all configuration registers in the nRF24L01 and is accessible in all operation modes of the chip.

The embedded baseband protocol engine (Enhanced ShockBurst™) is based on packet communication and supports various modes from manual operation to advanced autonomous protocol operation. Internal FIFOs ensure a smooth data flow between the radio front end and the system's MCU. Enhanced ShockBurst™ reduces system cost by handling all the high-speed link layer operations.

The radio front end uses GFSK modulation. It has user configurable parameters like frequency channel, output power and air data rate.

The air data rate supported by the nRF24L01 is configurable to 2Mbps. The high air data rate combined with two power saving modes makes the nRF24L01 very suitable for ultra low power designs.

Internal voltage regulators ensure a high Power Supply Rejection Ratio (PSRR) and a wide power supply range.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 Features

Features of the nRF24L01 include:

- Radio
  - Worldwide 2.4GHz ISM band operation
  - 126 RF channels
  - Common RX and TX pins
  - GFSK modulation
  - 1 and 2Mbps air data rate
  - 1MHz non-overlapping channel spacing at 1Mbps
  - 2MHz non-overlapping channel spacing at 2Mbps
- Transmitter
  - Programmable output power: 0, -6, -12 or -18dBm
  - 11.3mA at 0dBm output power
- Receiver
  - Integrated channel filters
  - 12.3mA at 2Mbps
  - -82dBm sensitivity at 2Mbps
  - -85dBm sensitivity at 1Mbps
  - Programmable LNA gain
- RF Synthesizer
  - Fully integrated synthesizer
  - No external loop filer, VCO varactor diode or resonator
  - Accepts low cost ±60ppm 16MHz crystal
- Enhanced ShockBurst™
  - 1 to 32 bytes dynamic payload length
  - Automatic packet handling
  - Auto packet transaction handling
  - 6 data pipe MultiCeiver™ for 1:6 star networks
- Power Management
  - Integrated voltage regulator
  - 1.9 to 3.6V supply range
  - Idle modes with fast start-up times for advanced power management
  - 22uA Standby-I mode, 900nA power down mode
  - Max 1.5ms start-up from power down mode
  - Max 130us start-up from standby-I mode
- Host Interface
  - 4-pin hardware SPI
  - Max 8Mbps
  - 3 separate 32 bytes TX and RX FIFOs
  - 5V tolerant inputs
- Compact 20-pin 4x4mm QFN package

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 Block diagram

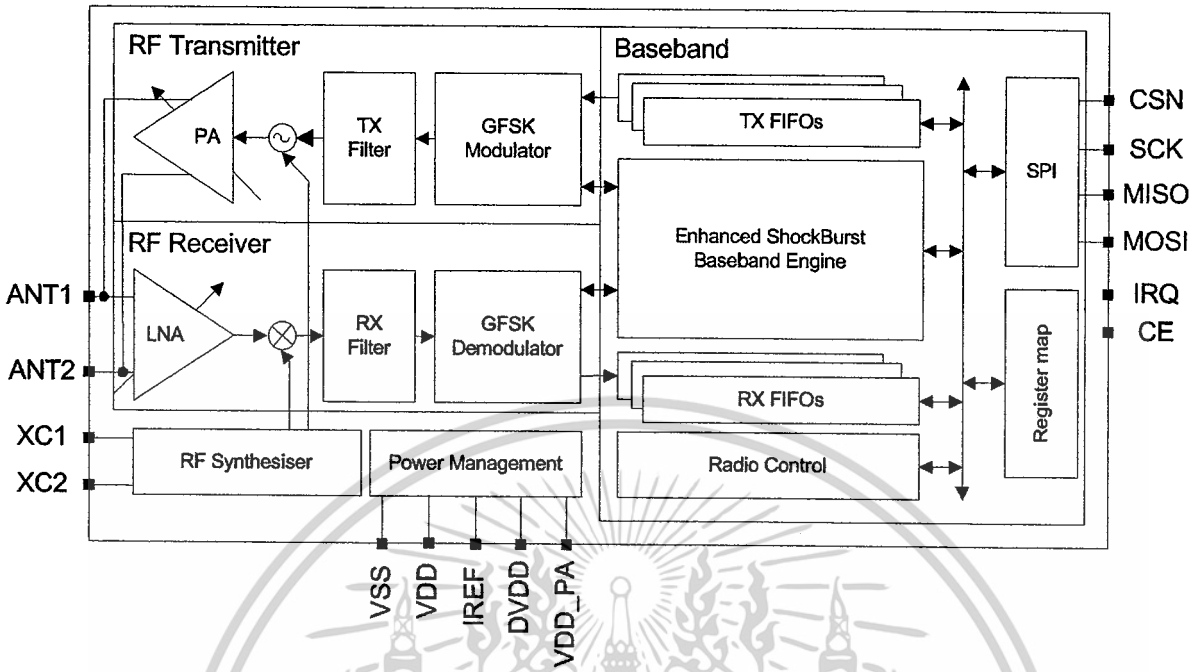


Figure 1. nRF24L01 block diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2 Pin Information

2.1 Pin assignment

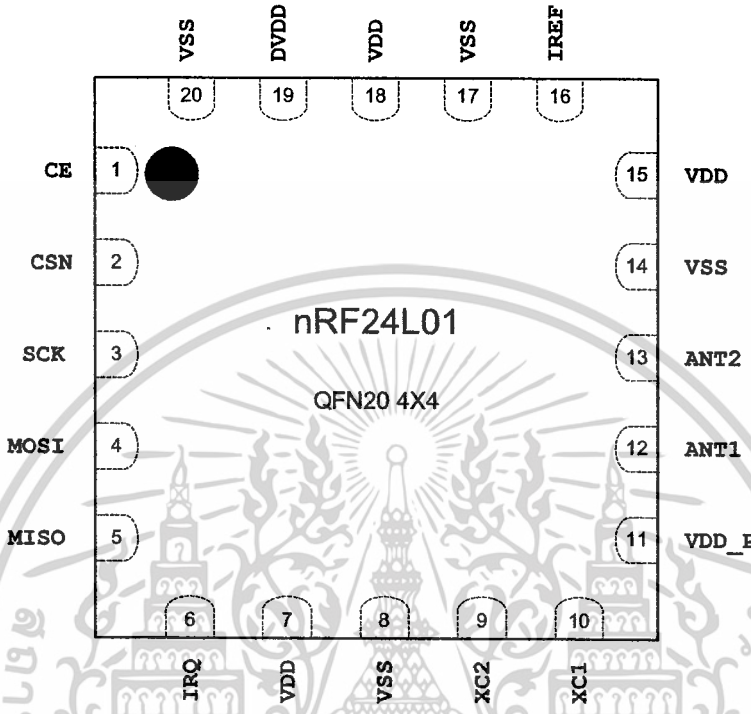


Figure 2. nRF24L01 pin assignment (top view) for the QFN20 4x4 package

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 Pin functions

Pin	Name	Pin function	Description
1	CE	Digital Input	Chip Enable Activates RX or TX mode
2	CSN	Digital Input	SPI Chip Select
3	SCK	Digital Input	SPI Clock
4	MOSI	Digital Input	SPI Slave Data Input
5	MISO	Digital Output	SPI Slave Data Output, with tri-state option
6	IRQ	Digital Output	Maskable interrupt pin. Active low
7	VDD	Power	Power Supply (+1.9V - +3.6V DC)
8	VSS	Power	Ground (0V)
9	XC2	Analog Output	Crystal Pin 2
10	XC1	Analog Input	Crystal Pin 1
11	VDD_PA	Power Output	Power Supply Output(+1.8V) for the internal nRF24L01 Power Amplifier. Must be connected to ANT1 and ANT2 as shown in Figure 30.
12	ANT1	RF	Antenna interface 1
13	ANT2	RF	Antenna interface 2
14	VSS	Power	Ground (0V)
15	VDD	Power	Power Supply (+1.9V - +3.6V DC)
16	IREF	Analog Input	Reference current. Connect a 22kΩ resistor to ground. See: Figure 30.
17	VSS	Power	Ground (0V)
18	VDD	Power	Power Supply (+1.9V - +3.6V DC)
19	DVDD	Power Output	Internal digital supply output for de-coupling purposes. See: Figure 30.
20	VSS	Power	Ground (0V)

Table 1. nRF24L01 pin function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 Absolute maximum ratings

Note: Exceeding one or more of the limiting values may cause permanent damage to nRF24L01.

Operating conditions	Minimum	Maximum	Units
Supply voltages			
VDD	-0.3	3.6	V
VSS		0	V
Input voltage			
V_I	-0.3	5.25	V
Output voltage			
V_O	VSS to VDD	VSS to VDD	
Total Power Dissipation			
P_D (T_A=85°C)		60	mW
Temperatures			
Operating Temperature	-40	+85	°C
Storage Temperature	-40	+125	°C

Table 2. Absolute maximum ratings

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The register values are maintained during standby modes and the SPI may be activated. For start up time see Table 13. on page 22.

6.1.4 RX mode

The RX mode is an active mode where the nRF24L01 radio is a receiver. To enter this mode, the nRF24L01 must have the PWR_UP bit set high, PRIM_RX bit set high and the CE pin set high.

In this mode the receiver demodulates the signals from the RF channel, constantly presenting the demodulated data to the baseband protocol engine. The baseband protocol engine constantly searches for a valid packet. If a valid packet is found (by a matching address and a valid CRC) the payload of the packet is presented in a vacant slot in the RX FIFO. If the RX FIFO is full, the received packet is discarded.

The nRF24L01 remains in RX mode until the MCU configures it to standby-I mode or power down mode. If the automatic protocol features (Enhanced ShockBurst™) in the baseband protocol engine are enabled, the nRF24L01 can enter other modes in order to execute the protocol.

In RX mode a carrier detect signal is avaliable. The carrier detect is a signal that is set high when a RF signal is detected inside the receiving frequency channel. The signal must be FSK modulated for a secure detection. Other signals can also be detected. The Carrier Detect (CD) is set high when an RF signal is detected in RX mode, otherwise CD is low. The internal CD signal is filtered before presented to CD register. The RF signal must be present for at least 128µs before the CD is set high. How to use the CD is described in Appendix E on page 74.

6.1.5 TX mode

The TX mode is an active mode where the nRF24L01 transmits a packet. To enter this mode, the nRF24L01 must have the PWR_UP bit set high, PRIM_RX bit set low, a payload in the TX FIFO and, a high pulse on the CE for more than 10µs.

The nRF24L01 stays in TX mode until it finishes transmitting a current packet. If CE = 0 nRF24L01 returns to standby-I mode. If CE = 1, the next action is determined by the status of the TX FIFO. If the TX FIFO is not empty the nRF24L01 remains in TX mode, transmitting the next packet. If the TX FIFO is empty the nRF24L01 goes into standby-II mode.The nRF24L01 transmitter PLL operates in open loop when in TX mode. It is important to never keep the nRF24L01 in TX mode for more than 4ms at a time. If the auto retransmit is enabled, the nRF24L01 is never in TX mode long enough to disobey this rule.

6.1.6 Operational modes configuration

The following table (Table 12.) describes how to configure the operational modes.

Mode	PWR_UP register	PRIM_RX register	CE	FIFO state
RX mode	1	1	1	-
TX mode	1	0	1	Data in TX FIFO. Will empty all levels in TX FIFO[a].
TX mode	1	0	minimum 10µs high pulse	Data in TX FIFO.Will empty one level in TX FIFO[b].
Standby-II	1	0	1	TX FIFO empty
Standby-I	1	-	0	No ongoing packet transmission
Power Down	0	-	-	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

a. In this operating mode if the CE is held high the TX FIFO is emptied and all necessary ACK and possible retransmits are carried out. The transmission continues as long as the TX FIFO is refilled. If the TX FIFO is empty when the CE is still high, nRF24L01 enters standby-II mode. In this mode the transmission of a packet is started as soon as the CSN is set high after a upload (UL) of a packet to TX FIFO.
b. This operating mode pulses the CE high for at least 10µs. This allows one packet to be transmitted. This is the normal operating mode. After the packet is transmittet, the nRF24L01 enters standby-I mode.

Table 12. nRF24L01 main modes

6.1.7 Timing Information

The timing information in this section is related to the transitions between modes and the timing for the CE pin. The transition from TX mode to RX mode or vice versa is the same as the transition from standby-I to TX mode or RX mode,Tstby2a.

Name	nRF24L01	Max.	Min.	Comments
Tpd2stby	Power Down → Standby mode	1.5ms		Internal crystal oscillator
Tpd2stby	Power Down → Standby mode	150µs		With external clock
Tstby2a	Standby modes → TX/RX mode	130µs		
Thce	Minimum CE high		10µs	
Tpece2csn	Delay from CE pos. edge to CSN low		4µs	

Table 13. Operational timing of nRF24L01

When nRF24L01 is in power down mode it must settle for 1.5ms before it can enter the TX or RX modes. If an *external clock* is used this delay is reduced to 150µs, see Table 13. on page 22. The settling time must be controlled by the MCU.

Note: The register value is lost if VDD is turned off. In this case, nRF24L01 must be configured before entering the TX or RX modes.

6.2 Air data rate

The air data rate is the modulated signaling rate the nRF24L01 uses when transmitting and receiving data.

The air data rate can be 1Mbps or 2Mbps. The 1Mbps data rate gives 3dB better receiver sensitivity compared to 2Mbps. High air data rate means lower average current consumption and reduced probability of on-air collisions.

The air data rate is set by the RF_DR bit in the RF_SETUP register.

A transmitter and a receiver must be programmed with the same air data rate to be able to communicate with each other.

For compatibility with nRF2401A, nRF24E1, nRF2402 and nRF24E2 the air data rate must be set to 1Mbps.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 RF channel frequency

The RF channel frequency determines the center of the channel used by the nRF24L01. The channel occupies a bandwidth of 1MHz at 1Mbps and 2MHz at 2Mbps. nRF24L01 can operate on frequencies from 2.400GHz to 2.525GHz. The resolution of the RF channel frequency setting is 1MHz.

At 2Mbps the channel occupies a bandwidth wider than the resolution of the RF channel frequency setting. To ensure non-overlapping channels in 2Mbps mode, the channel spacing must be 2MHz or more. At 1Mbps the channel bandwidth is the same as the resolution of the RF frequency setting.

The RF channel frequency is set by the RF_CH register according to the following formula:

F_0= 2400 + RF_CH [MHz]

A transmitter and a receiver must be programmed with the same RF channel frequency to be able to communicate with each other.

6.4 PA control

The PA control is used to set the output power from the nRF24L01 power amplifier (PA). In TX mode PA control has four programmable steps, see Table 14.

The PA control is set by the RF_PWR bits in the RF_SETUP register.

SPI RF-SETUP (RF_PWR)	RF output power	DC current consumption
11	0dBm	11.3mA
10	-6dBm	9.0mA
01	-12dBm	7.5mA
00	-18dBm	7.0mA

Conditions: VDD = 3.0V, VSS = 0V, T_A = 27°C, Load impedance = 15Ω+j88Ω.

Table 14. RF output power setting for the nRF24L01

6.5 LNA gain

The gain in the Low Noise Amplifier (LNA) in the nRF24L01 receiver is controlled by the LNA gain setting. The LNA gain makes it possible to reduce the current consumption in RX mode with 0.8mA at the cost of 1.5dB reduction in receiver sensitivity.

The LNA gain has two steps and is set by the LNA_HCURR bit in the RF_SETUP register.

6.6 RX/TX control

The RX/TX control is set by PRIM_RX bit in the CONFIG register and sets the nRF24L01 in transmit/ receive.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix A - Enhanced ShockBurst™ - Configuration and Communication Example

Enhanced ShockBurst™ Transmitting Payload

1. The configuration bit PRIM_RX has to be low.
2. When the application MCU has data to transmit, the address for the receiving node (TX_ADDR) and payload data (TX_PLD) has to be clocked into nRF24L01 through the SPI. The width of TX-payload is counted from number of bytes written into the TX FIFO from the MCU. TX_PLD must be written continuously while holding CSN low. TX_ADDR does not have to be rewritten if it is unchanged from last transmit. If the PTX device shall receive acknowledge, data pipe 0 has to be configured to receive the ACK packet. The RX address for data pipe 0 (RX_ADDR_P0) has to be equal to the TX address (TX_ADDR) in the PTX device. For the example in Figure 12. on page 37 the following address settings have to be performed for the TX5 device and the RX device:
   TX5 device: TX_ADDR = 0xB3B4B5B605
   TX5 device: RX_ADDR_P0 = 0xB3B4B5B605
   RX device: RX_ADDR_P5 = 0xB3B4B5B605
3. A high pulse on CE starts the transmission. The minimum pulse width on CE is 10µs.
4. nRF24L01 ShockBurst™:
   - Radio is powered up.
   - 16MHz internal clock is started.
   - RF packet is completed (see the packet description).
   - Data is transmitted at high speed (1Mbps or 2Mbps configured by MCU).
5. If auto acknowledgement is activated (ENAA_P0=1) the radio goes into RX mode immediately, unless the NO_ACK bit is set in the received packet. If a valid packet has been received in the valid acknowledgement time window, the transmission is considered a success. The TX_DS bit in the STATUS register is set high and the payload is removed from TX FIFO. If a valid ACK packet is not received in the specified time window, the payload is retransmitted (if auto retransmit is enabled). If the auto retransmit counter (ARC_CNT) exceeds the programmed maximum limit (ARC), the MAX_RT bit in the STATUS register is set high. The payload in TX FIFO is NOT removed. The IRQ pin is active when MAX_RT or TX_DS is high. To turn off the IRQ pin, the interrupt source must be reset by writing to the STATUS register (see Interrupt chapter). If no ACK packet is received for a packet after the maximum number of retransmits, no further packets can be transmitted before the MAX_RT interrupt is cleared. The packet loss counter (PLOS_CNT) is incremented at each MAX_RT interrupt. That is, ARC_CNT counts the number of retransmits that was required to get a single packet through. PLOS_CNT counts the number of packets that did not get through after maximum number of retransmits.
6. nRF24L01 goes into standby-I mode if CE is low. Otherwise next payload in TX FIFO is transmitted. If TX FIFO is empty and CE is still high, nRF24L01 enters standby-II mode.
7. If nRF24L01 is in standby-II mode, it goes to standby-I mode immediately if CE is set low.

Enhanced ShockBurst™ Receive Payload

1. RX is selected by setting the PRIM_RX bit in the CONFIG register to high. All data pipes that receive data must be enabled (EN_RXADDR register), auto acknowledgement for all pipes running Enhanced ShockBurst™ has to be enabled (EN_AA register), and the correct payload widths must be set (RX_PW_Px registers). Addresses have to be set up as described in item 2 in the Enhanced ShockBurst™ transmit payload chapter above.
2. Active RX mode is started by setting CE high.
3. After 130µs nRF24L01 is monitoring the air for incoming communication.
4. When a valid packet has been received (matching address and correct CRC), the payload is stored in the RX-FIFO, and the RX_DR bit in STATUS register is set high. The IRQ pin is active

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

when RX_DR is high. RX_P_NO in STATUS register indicates what data pipe the payload has been received in.

5. If auto acknowledgement is enabled, an ACK packet is transmitted back, unless the NO_ACK bit is set in the received packet. If there is a payload in the TX_PLD FIFO, this payload is added to the ACK packet.
6. MCU sets the CE pin low to enter standby-I mode (low current mode).
7. MCU can clock out the payload data at a suitable rate through the SPI.
8. nRF24L01 is now ready for entering TX or RX mode or power down mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix B - Configuration for compatibility with nRF24XX

How to setup nRF24L01 to receive from an nRF2401/nRF2402/nRF24E1/nRF24E2:

1. Use the same CRC configuration as the nRF2401/nRF2402/nRF24E1/nRF24E2
2. Set the PWR_UP and PRIM_RX bit to 1
3. Disable auto acknowledgement on the data pipe that is addressed
4. Use the same address width as the PTX device
5. Use the same frequency channel as the PTX device
6. Select data rate 1Mbps on both nRF24L01 and nRF2401/nRF2402/nRF24E1/nRF24E2
7. Set correct payload width on the data pipe that is addressed
8. Set CE high

How to setup nRF24L01 to transmit to an nRF2401/nRF24E1:

1. Use the same CRC configuration as the nRF2401/nRF2402/nRF24E1/nRF24E2
2. Set the PRIM_RX bit to 0
3. Set the Auto Retransmit Count to 0 to disable the auto retransmit functionality
4. Use the same address width as the nRF2401/nRF2402/nRF24E1/nRF24E2 uses
5. Use the same frequency channel as the nRF2401/nRF2402/nRF24E1/nRF24E2 uses
6. Select data rate 1Mbps on both nRF24L01 and nRF2401/nRF2402/nRF24E1/nRF24E2
7. Set PWR_UP high
8. Clock in a payload that has the same length as the nRF2401/nRF2402/nRF24E1/nRF24E2 is configured to receive
9. Pulse CE to transmit the packet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix C - Carrier wave output power

The output power of a radio is a critical factor for achieving wanted range. Output power is also the first test criteria needed to qualify for all telecommunication regulations.

Configuration

1. Set PWR_UP = 1 in the CONFIG register
2. Wait 1.5ms PWR_UP->standby
3. Clear the PRIM_RX in the CONFIG register
4. Set all auto acknowledgement functionality in the EN_AA register and the SETUP_RETR register to 0
5. Set output power
6. Set PLL_LOCK to 1
7. Configure TX address as 5 bytes with all 0xFF
8. Fill the TX payload with 32 bytes of 0xFF
9. Turn off CRC
10. Set the wanted RF channel
11. Transmit the packet by pulsing CE (minimum 10µs)
12. Wait until the transmission ends (indicated by IRQ going active, a fixed delay of 1ms can also be used)
13. Set CE high
14. Use the SPI command for re-use of last sent packet (REUSE_TX_PL)
15. Keep CE high as long as the carrier is needed

The nRF24L01 should now output a carrier.

Note: This is not a clean carrier but is slightly modulated by the preamble.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix D - Application example

nRF24L01 with single ended matching network crystal, bias resistor, and decoupling capacitors.

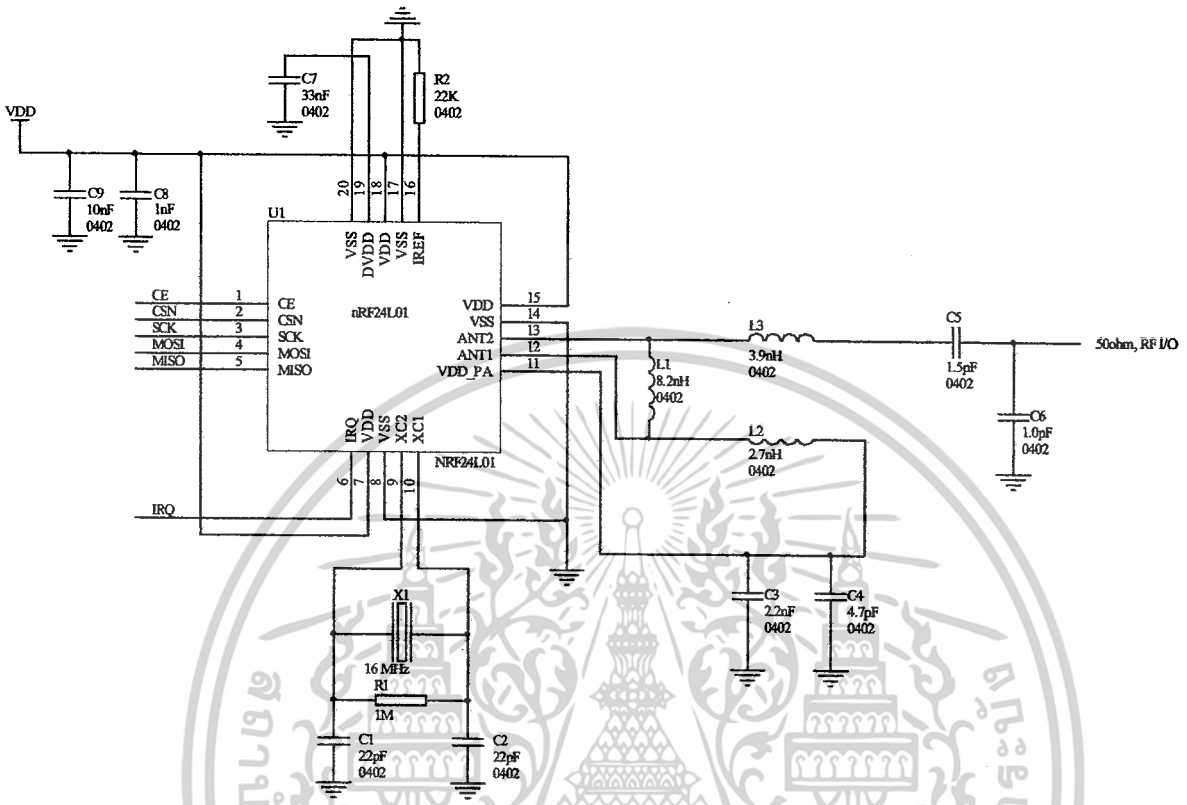


Figure 30. nRF24L01 schematic for RF layouts with single ended 50Ω RF output

Part	Designator	Footprint	Description
22pF[a]	C1	0402	NPO, +/- 2%
22pF[a]	C2	0402	NPO, +/- 2%
2.2nF	C3	0402	X7R, +/- 10%
4.7pF	C4	0402	NPO, +/- 0.25pF
1.5pF	C5	0402	NPO, +/- 0.1pF
1,0pF	C6	0402	NPO, +/- 0.1pF
33nF	C7	0402	X7R, +/- 10%
1nF	C8	0402	X7R, +/- 10%
10nF	C9	0402	X7R, +/- 10%
8,2nH	L1	0402	chip inductor +/- 5%
2.7nH	L2	0402	chip inductor +/- 5%
3,9nH	L3	0402	chip inductor +/- 5%
1MΩ	R1	0402	+/-10%
22kΩ	R2	0402	+/-1%
nRF24L01	U1	QFN20 4x4	
16MHz	X1		+/-60ppm, C_L=12pF

a. C1 and C2 must have values that match the crystals load capacitance, C_L.

Table 26. Recommended components (BOM) in nRF24L01 with antenna matching network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCB layout examples

Figure 31. on page 70, Figure 32. on page 71 and Figure 33. on page 71 show a PCB layout example for the application schematic in Figure 30. on page 69.

A double-sided FR-4 board of 1.6mm thickness is used. This PCB has a ground plane on the bottom layer. Additionally, there are ground areas on the component side of the board to ensure sufficient grounding of critical components. A large number of via holes connect the top layer ground areas to the bottom layer ground plane.

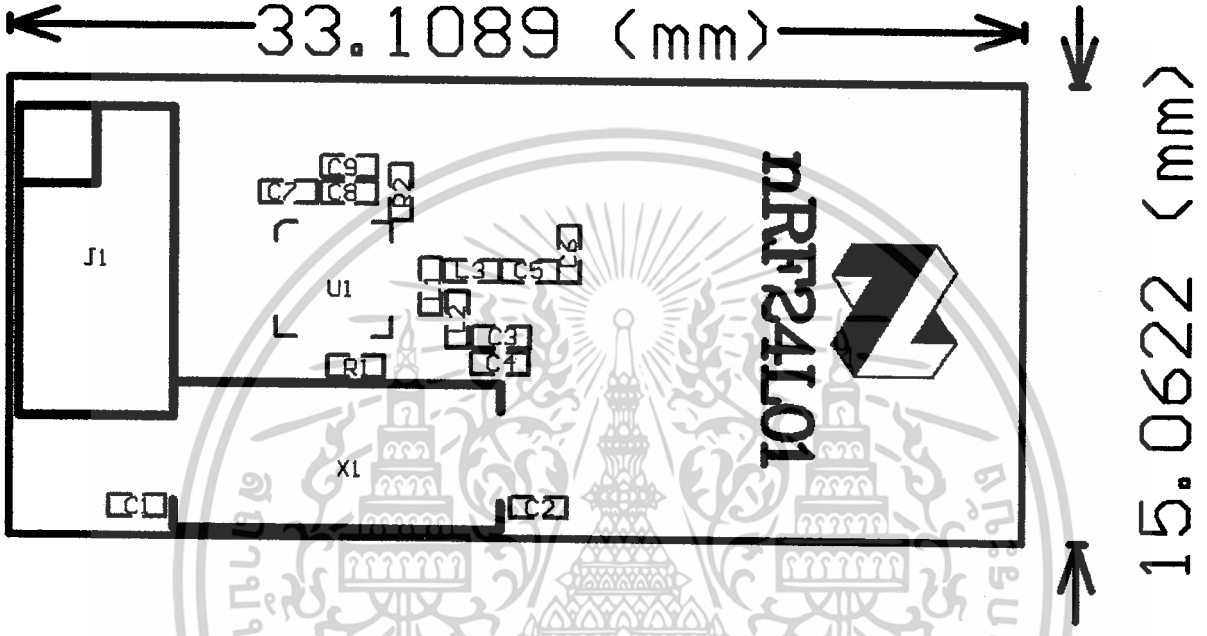


Figure 31. Top overlay (nRF24L01 RF layout with single ended connection to PCB antenna and 0402 size passive components)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

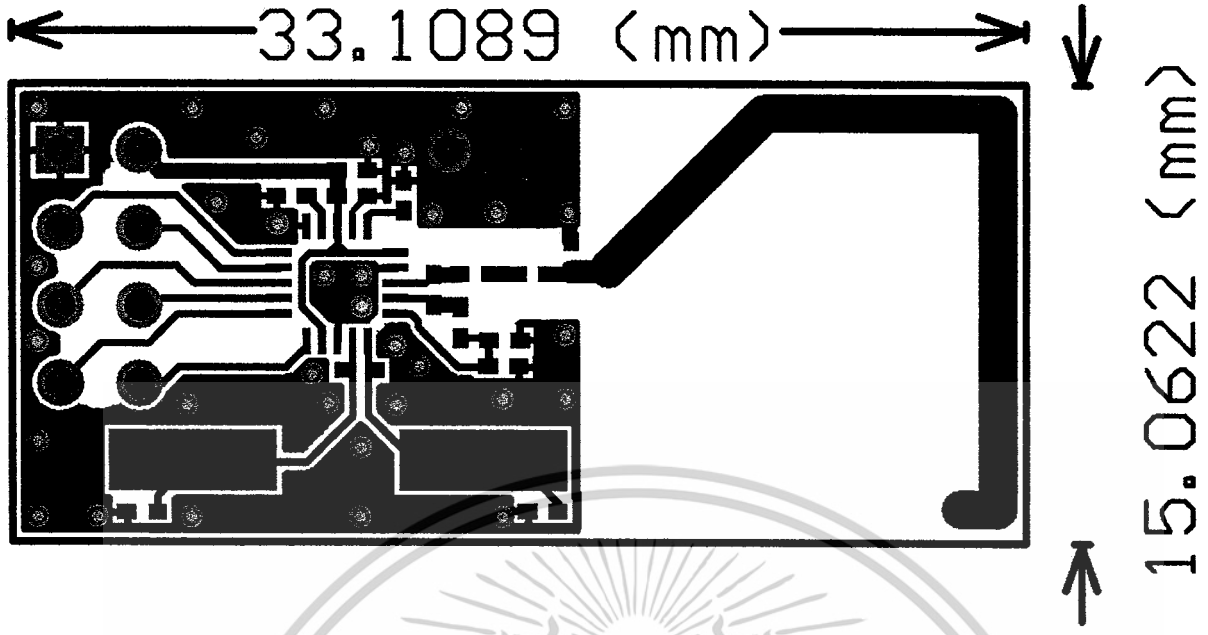


Figure 32. Top layer (nRF24L01 RF layout with single ended connection to PCB antenna and 0402 size passive components)

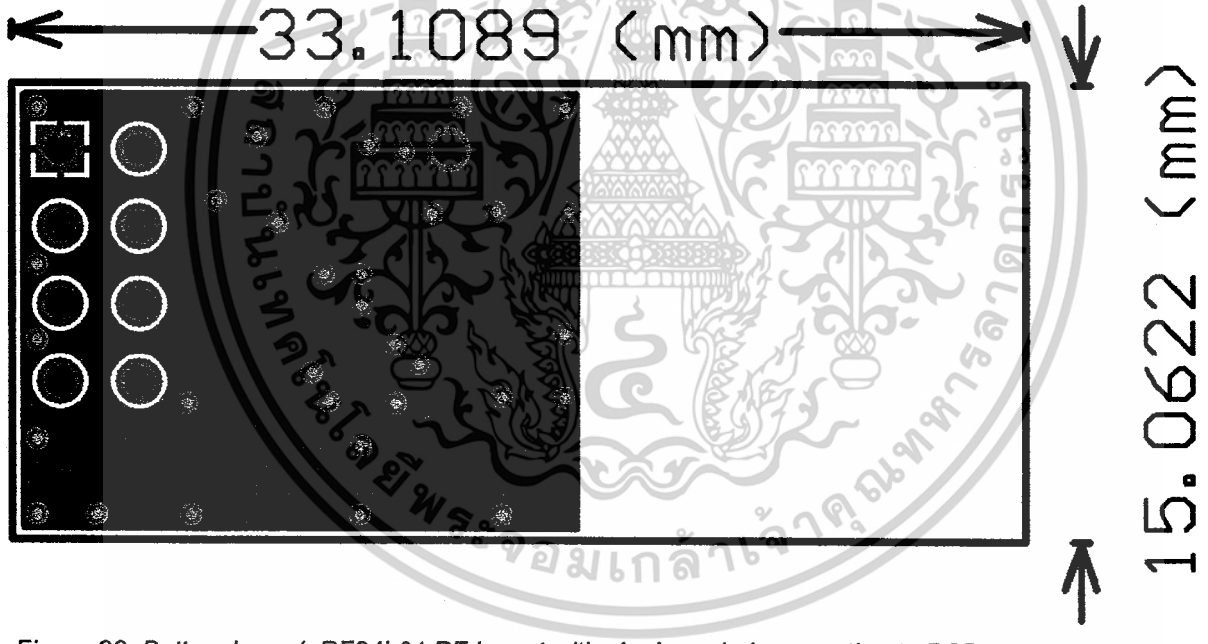


Figure 33. Bottom layer (nRF24L01 RF layout with single ended connection to PCB antenna and 0402 size passive components

The nest figure (Figure 34. on page 72, Figure 35. on page 72 and Figure 36. on page 73) is for the SMA output to have a board for direct measurements at a 50Ω SMA connector.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

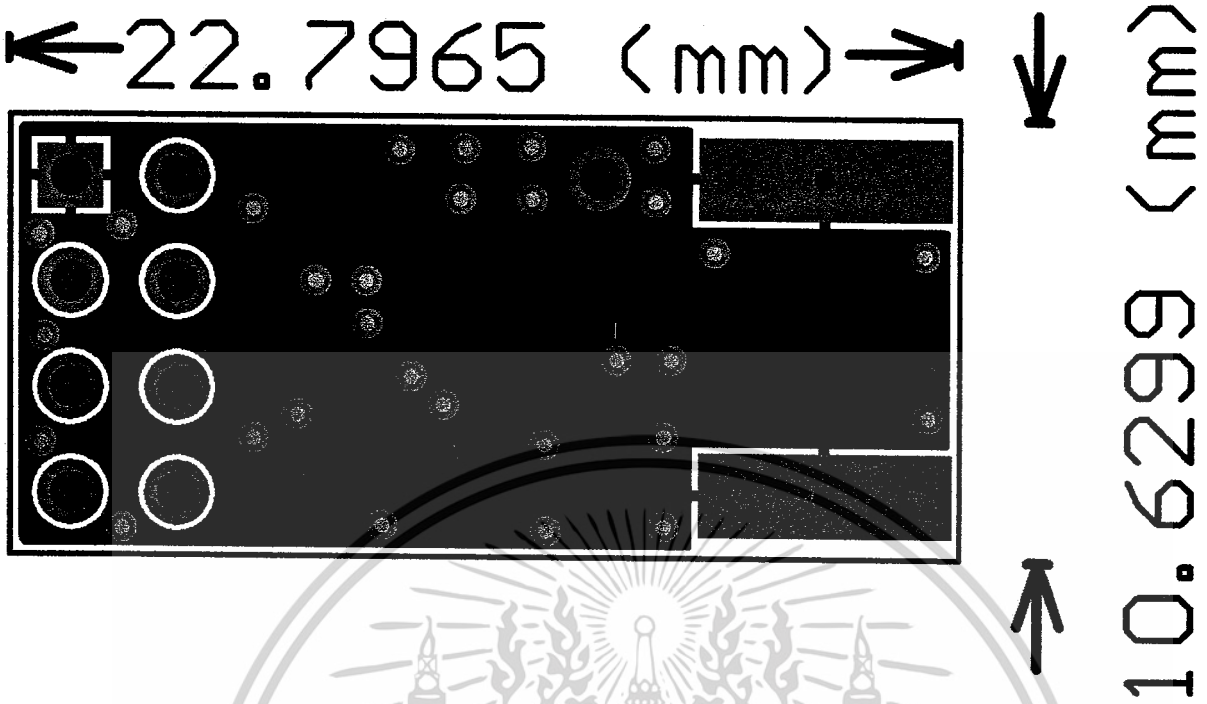


Figure 36. Bottom Layer (Module with OFM crystal and SMA connector)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้