สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์วัดระยะทางโดยใช้คอมพิวเตอร์

THE DISTANCE MEASUREMENT ROBOT BY COMPUTER

นายจรรยา ยศคำลือ
นายสุวินัย อ่องสถาน
นางสาวสุปราณี ไมมา

วท.
ว๑๒๒๘ ส
๒๕๕๑
เลขหมู่.....
เลขทะเบียน 82923
วัน,เดือน,ปี 29 ก.ค. 2551
b 11952725x
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THE DISTANCE MEASUREMENT ROBOT BY COMPUTER

MR.CHANYA YOSKHUMLU

MR.SUWINAI OANGSATHAN

MISS.SUPRANEE MAIMA

A THESIS SUBMITTED IN PARTIAL FULFLLMENT OF

THE REQUIRENT FOR THE THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ หุ่นยนต์วัดระยะทางโดยใช้คอมพิวเตอร์

ชื่อนักศึกษา นายจรรยา ยศคำลือ รหัสนักศึกษา 48015569

นายสุวินัย อ่องสถาน รหัสนักศึกษา 48015602

นางสาวสุปราณี ไมมา รหัสนักศึกษา 48015643

อาจารย์ที่ปรึกษา ผศ.สมภพ แก้วมีชัย

อ.สรพงษ์ วชิรรัตนพรกุล

ภาควิชา วิศวกรรมสารสนเทศ

ปีการศึกษา 2550

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(ผศ. สมภพ แก้วมีชัย)

อาจารย์ที่ปรึกษา

.....

(อ.สรพงษ์ วชิรรัตนพรกุล)

อาจารย์ที่ปรึกษาร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	หุ่นยนต์วัดระยะทางโดยใช้คอมพิวเตอร์		
ชื่อนักศึกษา	นายจรรยา	ยศคำลือ	รหัสนักศึกษา 48015569
	นายสุวินัย	อ่องสถาน	รหัสนักศึกษา 48015602
	นางสาวสุปราณี	ไมมา	รหัสนักศึกษา 48015643
อาจารย์ที่ปรึกษา	ผศ.สมภพ	แก้วมีชัย	
	อ.สรพงษ์	วชิรรัตนพรกุล	
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตร์ สาขาวิศวกรรมสารสนเทศ		
ภาควิชา	วิศวกรรมสารสนเทศ		
ปีการศึกษา	2550		

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการนำเสนอการใช้คลื่นอัลตร้าโซนิกในการวัดระยะซึ่งจะใช้หุ่นยนต์ในการนำพาตัวอัลตร้าโซนิกไปวัดระยะในพื้นที่ที่ไม่สามารถจะไปเข้าถึงดังเช่นในบริเวณที่มืดหรือที่ที่มีอันตราย โดยจะแบ่งการทำงานทั้งหมดออกเป็น 2 ส่วน คือ ส่วนซอร์ฟแวร์ ส่วนฮาร์ดแวร์ ซึ่งประกอบด้วยไอซีไมโครคอนโทรลเลอร์เบอร์ AT89C52,พอร์ต RS232,กล้องไวเลทไร้สาย,เซอร์โวมอเตอร์ ซึ่งการควบคุมทั้งหมดใช้โปรแกรมภาษาวิชวลเบสิกและภาษาซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	THE DISTANCE MEASUREMENT ROBOT BY COMPUTER		
Student	MR.Chanya	Yoskhumlu	ID. 48015569
	MR.Suwinai	Oangsathan	ID. 48015602
	MISS.Supranee	Maima	ID. 48015643

Advisor	Asst.Prof.Sompoph	Kaewmechai
	MR.Sorapong	Wachirarattanpornkul
Graduate Level	Bachelor Degree Of Informatinon Engineering	
Department	Information Engineering	
Academic Year	2007	

ABSTRACT

The degree composes this is the lead presents using waves is thin the NIC in distance measurement profoundly will use a robot in bringing is thin the NIC goes to the temple the distance in the area at will can not go to appreciate such as , in the area where is come dark

By the degree will compose this divide the all work goes out to are the part is , software part , hardware part , consist of IC microcontroller AT89C52, Port RS232,A camera wireless quickly , The circuit controls small stump hill , Profoundly the all supervision uses Visual Basic language program and the C programming language

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงได้เป็นอย่างดีด้วยคำแนะนำ ชี้แนะแนวทางในการแก้ปัญหาจนกระทั่งโครงงานสำเร็จไปได้ด้วยดีจากผู้ช่วยศาสตราจารย์สมภพ แก้วมีชัยซึ่งเป็นผู้ที่คอยดูแลมาโดยตลอดผู้จัดทำโครงการรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณอาจารย์สรพงษ์ วชิรรัตนพรกุลอาจารย์ประจำภาควิชาวิศวกรรมารสนเทศที่คอยให้คำแนะนำและชี้แนะแนวทางในการแก้ปัญหาและขอขอบคุณเพื่อนๆ ที่น่ารักทุกคนที่คอยช่วยเหลือให้คำแนะนำและคอยให้กำลังใจมาโดยตลอดจนวิทยานิพนธ์นี้เสร็จสมบูรณ์

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	X
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการศึกษา	1
1.3 ขอบเขตของโครงงาน	1
1.4 ประโยชน์ที่คาดว่าจะได้รับจากหุ่นยนต์วัดระยะ	2
1.5 วิธีการดำเนินงานโครงงาน	2
บทที่ 2 ทฤษฎี	
2.1 โมดูตรวจจับและวัดระยะทางด้วยอัลตร้าโซนิก	3
2.2 กล้องส่งภาพแบบไร้สาย	9
2.3 เซอร์โวมอเตอร์	10
2.4 พอร์ตอนุกรม	19
2.5 ไมโครคอนโทรลเลอร์ MCS-51	24
2.6 Wireless	30
2.7 ET-RF24G V1.0	33
2.8 ตัวตรวจจับล้อแบบสะท้อนแสง	38
บทที่ 3 โครงสร้างของระบบโครงงาน	
3.1 โครงสร้างโดยรวม	40
3.2 การออกแบบทางด้าน Soft Ware	45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 4 ผลการทดลอง	
4.1 ทดสอบการรับภาพจากกล้องโดยใช้โปรแกรมวิชวลเบสิก	49
4.2 ผลการทดลองการรับค่าอัลตร้าโซนิก	50
4.3 สรุปผลการทดลอง	56
4.4 ปัญหาและแนวทางแก้ไข	56
บรรณานุกรม	
ภาคผนวก ก โปรแกรม	
ภาคผนวก ข คู่มือการใช้งาน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
รูปที่ 2.1	แสดงหลักการตรวจจับวัตถุโดยใช้ความถี่เหนือเสียงหรืออัลตร้าโซนิก	4
รูปที่ 2.2	แสดงขาสัญญาณของ SRF05 และการกำหนดโหมดทำงาน	4
รูปที่ 2.3	แสดงวงจรของบอร์ด ADX-SRF04 และการเชื่อมต่อกับโมดู SRF05	6
รูปที่ 2.4	แสดงไดอะแกรมเวลาแสดงสัญญาณที่ส่งไปยัง SRF05 และสัญญาณที่ตอบรับกลับมาจาก SRF05 ในกรณีติดต่อแบบ 2 สัญญาณ	8
รูปที่ 2.5	แสดงไดอะแกรมเวลาแสดงสัญญาณที่ส่งไปยัง SRF05 และสัญญาณที่ตอบรับกลับมาจาก SRF05 ในกรณีติดต่อแบบ 1 สัญญาณ	8
รูปที่ 2.6	แสดงกล้องส่งภาพแบบไร้สาย	9
รูปที่ 2.7	แสดงโครงสร้างชุดเซอร์โวมอเตอร์	10
รูปที่ 2.8	แสดงส่วนประกอบของเซอร์โวมอเตอร์	11
รูปที่ 2.9	แสดงการเชื่อมต่อเซอร์โวมอเตอร์	11
รูปที่ 2.10	แสดง DC SERVO MOTOR	12
รูปที่ 2.11	แสดงลักษณะของ Pulse สำหรับควบคุมให้มอเตอร์หมุนทวนเข็มนาฬิกา	13
รูปที่ 2.12	แสดงลักษณะของ Pulse สำหรับควบคุมให้มอเตอร์หมุนตามเข็มนาฬิกา	13
รูปที่ 2.13	แสดงลักษณะของ Pulse สำหรับควบคุมให้มอเตอร์หยุดหมุน	14
รูปที่ 2.14	แสดงลักษณะการเคลื่อนที่ของรถหุ่นยนต์โดยการขับเคลื่อนจาก DC SERVO MOTOR	15
รูปที่ 2.15	แสดงการบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหน้า	16
รูปที่ 2.16	แสดงการบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหลัง (ถอยหลัง)	17
รูปที่ 2.17	แสดงการบังคับให้รถหุ่นยนต์เลี้ยวซ้ายโดยการเคลื่อนที่ของล้อขวาอย่างเดียว	17
รูปที่ 2.18	แสดงการบังคับให้รถหุ่นยนต์เลี้ยวซ้ายโดยให้ล้อซ้ายถอยหลังส่วนล้อขวาเดินหน้า	18
รูปที่ 2.19	แสดงการบังคับให้รถหุ่นยนต์เลี้ยวขวาโดยการเคลื่อนที่ของล้อซ้ายอย่างเดียว	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
รูปที่ 2.20	แสดงการบังคับให้รถหุ่นยนต์เลี้ยวขวาโดยให้ล้อขวาถอยหลังส่วนล้อซ้ายเดินหน้า	19
รูปที่ 2.21	แสดงการสื่อสารแบบอนุกรม	19
รูปที่ 2.22	แสดงพอร์ตอนุกรมของ PC และของอุปกรณ์ภายนอก	20
รูปที่ 2.23	แสดง DB9 ตัวผู้ เมื่อมองจากด้านหลัง	20
รูปที่ 2.24	แสดงการเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย DB9	21
รูปที่ 2.25	แสดงระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL	22
รูปที่ 2.26	แสดงการสื่อสารแบบซิงโครนัส	23
รูปที่ 2.27	แสดงการสื่อสารแบบอะซิงโครนัส	23
รูปที่ 2.28	แสดงบอร์ดไมโครคอนโทรลเลอร์	25
รูปที่ 2.29	แสดงหน่วยความจำรอมและหน่วยความจำแรม	26
รูปที่ 2.30	แสดงหน่วยความจำแรมภายใน MCS-51	26
รูปที่ 2.31	แสดงผังตำแหน่ง Program memory	27
รูปที่ 2.32	แสดงผัง Internal data memoryและ External data memory	28
รูปที่ 2.33	แสดงผังหน่วยความจำภายใน	28
รูปที่ 2.34	แสดง Lower 128 bytes of internal data memory	29
รูปที่ 2.35	แสดง Upper 128 bytes of internal data memory	30
รูปที่ 2.36	แสดงElectromagnetic spectrum	30
รูปที่ 2.37	แสดงข้อมูล Wireless ของช่วงคลื่นความถี่ Radio	31
รูปที่ 2.38	แสดงการโรมมิ่ง และการแม็บกับมาตรฐานในชั้นที่สูงขึ้น	32
รูปที่ 2.39	แสดงการเลือกโหมดการทำงานสำหรับใช้งานปกติ (Run Mode)	34
รูปที่ 2.40	แสดงการเลือกโหมดการทำงานสำหรับกำหนดค่า Configuration	35
รูปที่ 2.41	แสดงโปรแกรมที่ใช้สำหรับกำหนดค่า Configuration ของET-RF 24G V1.0	36
รูปที่ 2.42	แสดงตัวตรวจจับรหัสล้อแบบสะท้อนแสง	39
รูปที่ 2.43	แสดงวงจรตัวตรวจจับรหัสล้อแบบสะท้อนแสง	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
รูปที่ 3.1	แสดงบล็อกไดอะแกรมของการรับส่งสัญญาณโดยใช้โปรแกรม visual basic	40
รูปที่ 3.2	แสดงบล็อกไดอะแกรมของการรับส่งสัญญาณภาพ	41
รูปที่ 3.3	แสดงบล็อกไดอะแกรมของการควบคุมมอเตอร์	42
รูปที่ 3.4	แสดงการใช้งานไมโครคอนโทรลเลอร์ติดต่อกับพอร์ตอนุกรม	43
รูปที่ 3.5	แสดงวงจรทดลองติดต่อไมโครคอนโทรลเลอร์ AT89C52 กับ SRF05 โมดูลวัดระยะทาง	44
รูปที่ 3.6	โฟลว์ชาร์ตแสดงการควบคุมทิศทางการเคลื่อนที่ของหุ่นยนต์	45
รูปที่ 3.7	โฟลว์ชาร์ตแสดงการทำงานของหุ่นยนต์	46
รูปที่ 3.8	โฟล์วชาร์ตแสดงการส่งค่าอัลตร้าโซนิกและเอ็น โค๊ดเดอร์	47
รูปที่ 3.9	แสดงด้านหน้าของหุ่นยนต์	48
รูปที่ 3.10	แสดงด้านข้างของหุ่นยนต์	48
รูปที่ 3.11	แสดงด้านบนของหุ่นยนต์	48
รูปที่ 4.1	แสดงหน้าต่างโปรแกรมการรับส่งภาพ	49
รูปที่ 4.2	แสดงการวัดระยะในระยะ 25 เซนติเมตร	50
รูปที่ 4.3	แสดงการวัดระยะในระยะ 50 เซนติเมตร	50
รูปที่ 4.4	แสดงการวัดระยะในระยะ 75 เซนติเมตร	51
รูปที่ 4.5	แสดงการวัดระยะในระยะ 100 เซนติเมตร	51
รูปที่ 4.6	แสดงการวัดระยะในระยะ 125 เซนติเมตร	51
รูปที่ 4.7	แสดงการวัดระยะในระยะ 150 เซนติเมตร	51
รูปที่ 4.8	แสดงการวัดระยะในระยะ 175 เซนติเมตร	52
รูปที่ 4.9	แสดงการวัดระยะในระยะ 200 เซนติเมตร	52
รูปที่ 4.10	แสดงการวัดระยะในระยะ 225 เซนติเมตร	52
รูปที่ 4.11	แสดงการวัดระยะในระยะ 250 เซนติเมตร	52
รูปที่ 4.12	แสดงการวัดระยะในระยะ 275 เซนติเมตร	53
รูปที่ 4.13	แสดงการวัดระยะในระยะ 300 เซนติเมตร	53
รูปที่ 4.14	แสดงการวัดระยะในระยะ 325 เซนติเมตร	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
รูปที่ 4.15	แสดงการวัดระยะในระยะ 350 เซนติเมตร	53
รูปที่ 4.16	แสดงการวัดระยะในระยะ 375 เซนติเมตร	54
รูปที่ 4.17	แสดงการวัดระยะในระยะ 400 เซนติเมตร	54
รูปที่ 4.18	แสดงการทำงานในที่มืด	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

		หน้า
ตารางที่ 2.1	แสดง Description แต่ละชนิด	20
ตารางที่ 4.1	ผลการทดลองระยะการเคลื่อนที่ของรถสำรวจและระยะการรับภาพ	50
ตารางที่ 4.2	แสดงค่าที่วัดได้	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันนี้การวัดพื้นที่ยังคงใช้ตลับเมตรและค่าที่วัดได้จากความกว้าง ความยาวเราจะต้องนำมาคำนวณเพื่อที่จะได้พื้นที่จริงๆออกมา ซึ่งจะเห็นได้ว่าเกิดความล่าช้าและการที่จะเข้าไปวัดระยะในห้องที่มืดและแคบอาจจะทำได้ยาก ซึ่งอาจจะไม่รู้เลยว่ามีอันตรายหรือเปล่าบางทีอาจจะเสี่ยงต่อสัตว์มีพิษหรือสารพิษได้

ดังนั้นทางกลุ่มข้าพเจ้าจึงเล็งเห็นว่าเราน่าจะประดิษฐ์หุ่นยนต์วัดระยะพื้นที่ที่สามารถจะวัดพื้นที่ได้ค่าที่ใกล้เคียงกับความเป็นจริงให้มากที่สุดและเพื่อความสะดวกและรวดเร็วของการวัดพื้นที่ และปัจจุบันนี้เทคโนโลยีมีความเจริญก้าวหน้าไปอย่างรวดเร็วมีการพัฒนาสิ่งต่างๆไปอย่างมาก เช่น เทคโนโลยีคอมพิวเตอร์ หุ่นยนต์ ซึ่งคอมพิวเตอร์ได้เข้ามามีบทบาทอย่างมากในการควบคุมอุปกรณ์ต่างๆและหุ่นยนต์วัดระยะพื้นที่นี้ก็จะทำงานเชื่อมต่อกับคอมพิวเตอร์เช่นกัน ซึ่งคอมพิวเตอร์จะทำหน้าที่ประมวลผลเป็นค่าตัวเลขออกมา

1.2 วัตถุประสงค์ของการศึกษา

1. เพื่อความสะดวกในการวัดพื้นที่ในที่สว่างและที่มืด
2. เพื่อที่จะได้ค่าในการวัดพื้นที่ที่ใกล้เคียงกับความเป็นจริงให้มากที่สุด
3. เพื่อนำความรู้ที่เรียนมาประยุกต์ใช้ให้เกิดประโยชน์
4. เพื่อความปลอดภัยในการเข้าไปในพื้นที่ที่เสี่ยงภัย

1.3 ขอบเขตของโครงงาน

1. วัดระยะได้ในที่มืดและในที่ที่มีแสงสว่าง
2. สามารถควบคุมหุ่นยนต์วัดระยะทางได้ไม่ต่ำกว่า 50 เมตร
3. ขับเคลื่อนได้โดยใช้เซอร์โวมอเตอร์
4. ใช้โปรแกรม Visual Basic ในการติดต่ออุปกรณ์ระหว่างคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่คาดว่าจะได้รับจากหุ่นยนต์

1.วัดระยะทางโดยใช้คอมพิวเตอร์ที่สามารถวัดระยะทางได้จริง

2. สามารถวัดระยะทางในที่แคบได้

3. สามารถทำงานในที่มืดได้

4.ได้ศึกษาการใช้โปรแกรมเพื่อควบคุมหุ่นยนต์

1.5 วิธีการดำเนินงานโครงงาน

1. ศึกษาการเขียนโปรแกรม Visual Basic 6.0
2. ออกแบบโครงสร้างของ Hardware
3. ทดสอบวงจรโดยใช้ Photo Broad
4. สร้างวงจรโดยสมบูรณ์
5. ทดสอบการทำงานทั้งหมดของระบบ
6. วิเคราะห์ข้อบกพร่องของโครงงานและตรวจสอบความถูกต้องในการใช้งาน
7. สรุปผลและเขียนรายงานจัดทำโครงงานพร้อมทั้งนำเสนอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่2
ทฤษฎี

2.1 โมดูลตรวจจับและวัดระยะทางด้วยอัลตร้าโซนิก

คุณสมบัติ

1.ใช้ไฟเลี้ยง +5V ต้องการกระแสไฟฟ้า 30 mA

2.ใช้ตัวรับและส่งคลื่นอัลตร้าโซนิกใช้ความถี่ 40 kHz ในการทำงาน

3.วัดระยะทางในช่วง 1 เซนติเมตรถึง 4 เมตร

4. สัญญาณพัลส์สำหรับกระตุ้นการทำงานต้องมีความกว้างอย่างน้อย 10 ไมโครวินาที

5.ให้ผลลัพธ์จากการวัดระยะเป็นค่าความกว้างพัลส์ซึ่งเป็นสัดส่วนกับระยะทางที่วัดได้

6. มีขนาดเล็กคือ 43 มม. x 20 มม. x 17 มม. (กว้างxยาวxสูง)

7. สื่อสารกับไมโครคอนโทรลเลอร์ ยอดนิยมได้ทุกตระกูล อาทิ เบสิกแสตมป์ 2SX/2P, PIC, MCS-51,PSoC, 68HC11

8. สามารถติดต่อได้ 2 แบบคือ แบบ 2 สัญญาณ (Echo กับ Trigger)

อุปกรณ์เสริม

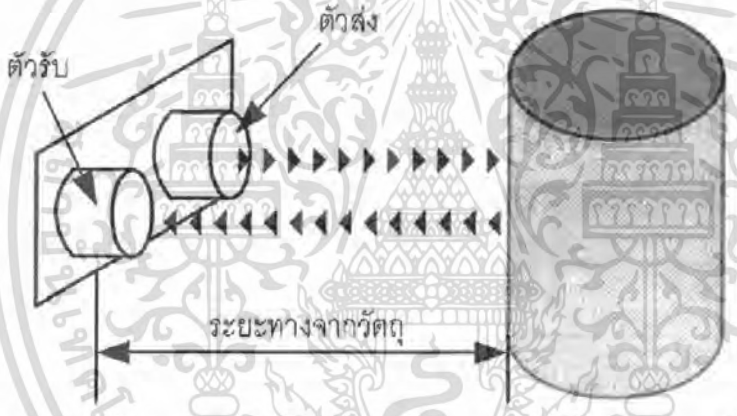
1. บอร์ด ADX-SRF04 ซึ่งเป็นบอร์ดอะแด็ปเตอร์สำหรับอำนวยสะดวกในการเชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ และแผงต่อวงจรหรือเบรดบอร์ด

2. สาย PCB3A สำหรับเชื่อมต่อกับบอร์ดควบคุมหุ่นยนต์ ของ i-nex SRF05 เป็นแผงวงจร

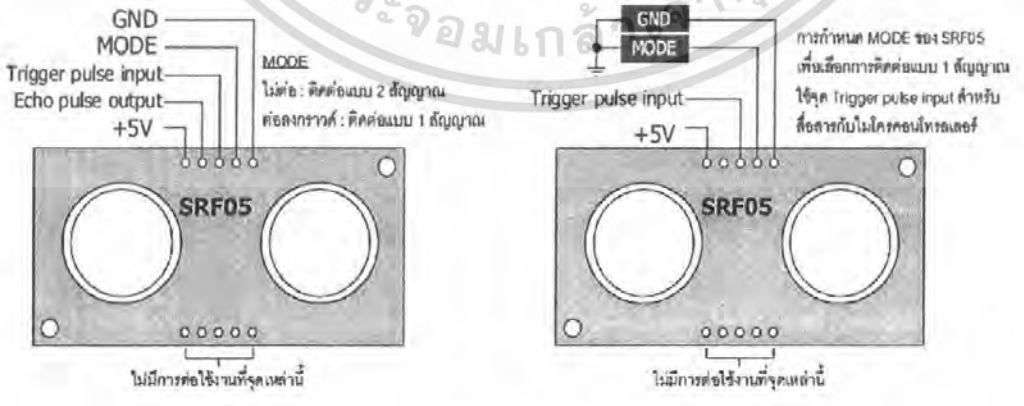
วัดตรวจจับและวัดระยะทางด้วยคลื่นอัลตร้าโซนิกที่มีความเที่ยงตรงสูงโดยสามารถวัดระยะได้ตั้งแต่ 1 เซนติเมตรไปจนถึง 4 เมตร SRF05 ถูกออกแบบมาให้ใช้งานกับไมโครคอนโทรลเลอร์ได้ง่ายโดยใช้ ขาเชื่อมต่อเพียง1หรือ 2 ขาขึ้นอยู่กับการกำหนดรูปแบบการทำงานทางฮาร์ดแวร์ เหมาะอย่างยิ่งกับการประยุกต์ใช้งานด้านหุ่นยนต์ SRF05 จะทำการส่งสัญญาณคลื่นอัลตร้าโซนิกออกไป แล้ววัดระยะเวลาที่มีสัญญาณสะท้อนตอบกลับมาเอาต์พุตที่ได้จากSRF05จะอยู่ในรูปของความกว้างพัลส์ซึ่งสัมพันธ์กับระยะทางของวัตถุที่ตรวจจับได้ ความถี่สัญญาณอัลตร้าโซนิกของ SRF05 คือ 40 kHz ถูกส่งออกอากาศด้วยความเร็ว 1.125 ฟุตต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มิลลิวินาที (ประมาณ 346 เมตรต่อวินาที) ดังนั้นเมื่อทราบความเร็วในการเคลื่อนที่ของคลื่น, เวลาเริ่มต้นส่งคลื่นและเวลาที่รับเสียงสะท้อนกลับมาจึงสามารถคำนวณหาค่าของระยะทางได้ ดังแสดงหลักการตรวจจับในรูปที่ 2.1 ระยะทางที่ได้นั้นจะต้องมีการคำนวณค่ากลับทางคณิตศาสตร์ เมื่อใช้กับไมโครคอนโทรลเลอร์ ถือว่าเป็นเรื่องยุ่งยากพอสมควร ดังนั้น SRF05 จึงประมวลผลค่าทางคณิตศาสตร์ต่างๆ เหล่านี้ไว้เรียบร้อยแล้ว จากนั้นส่งผลลัพธ์ที่วัดได้ออกมาเป็นพัลส์ที่มีความกว้างสัมพันธ์กับระยะทางที่วัดได้ การส่งผลลัพธ์ที่วัดได้ออกมาเป็นในเชิงความกว้างของสัญญาณพัลส์อาจจะดูว่ายุ่งยากกว่าการส่งเป็นข้อมูลดิจิตอลออกมาแต่การส่งออกมาเป็นข้อมูลดิจิตอลอาจต้องใช้สายสัญญาณจำนวนมากซึ่งทำให้ต้องใช้ขาพอร์ตในการเชื่อมต่อเป็นจำนวนมากตามไปด้วยดังนั้นหากส่งผลลัพธ์ออกมาในรูปของสัญญาณพัลส์จะใช้สายสัญญาณเพียงเส้นเดียวจึงทำให้สะดวกมากในการนำมาเชื่อมต่อกับไมโครคอนโทรลเลอร์



รูปที่ 2.1 แสดงหลักการตรวจจับวัตถุโดยใช้สัญญาณความถี่เหนือเสียงหรืออัลตร้าโซนิก



รูปที่ 2.2 แสดงขาสัญญาณของอัลตร้าโซนิกและการกำหนดโหมดทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1. จุดต่อใช้งานของ SRF05

จุดต่อใช้งานของ SRF05 มีจุดต่อสำหรับใช้งานอยู่ทั้งหมด 5 จุดตามรูปที่ 2.2

1. ขาไฟเลี้ยง (+5V) สำหรับต่อไฟเลี้ยงแรงดัน +5V

2. ขา Echo Pulse Output (ECHO) เป็นขาเอาต์พุตสำหรับส่งสัญญาณพัลส์ออกจาก SFR05 ซึ่งการใช้งานจะนำขานี้ไปต่อเข้ากับพอร์ตอินพุตของไมโครคอนโทรลเลอร์เพื่อตรวจจับความกว้างของสัญญาณพัลส์ที่ส่งออกมาเพื่อแปลความหมายออกมาเป็นระยะทางอีกครั้ง

3. ขา Trigger Pulse Input (TRIGGER) เป็นขาอินพุตรับสัญญาณพัลส์ที่มีความกว้างอย่างน้อย 10 ไมโครวินาทีเพื่อกระตุ้นการสร้างคลื่นอัลตร้าโซนิกความถี่ 40 kHz ออกสู่อากาศจากตัวส่งดังนั้นเมื่อคลื่นความถี่ดังกล่าวนี้เคลื่อนที่ไปกระทบสิ่งกีดขวางที่อยู่เบื้องหน้าก็จะเกิดการสะท้อนกลับเข้ามายังตัวรับและถูกแปลงออกมาเป็นความกว้างของสัญญาณพัลส์ที่จะส่งออกไปทางขา Echo Pulse Output นอกจากนี้ในโหมด 1 สัญญาณจะใช้จุดนี้เป็นจุดสื่อสารข้อมูลอนุกรมเพื่อรับส่งค่าการวัดกับไมโครคอนโทรลเลอร์

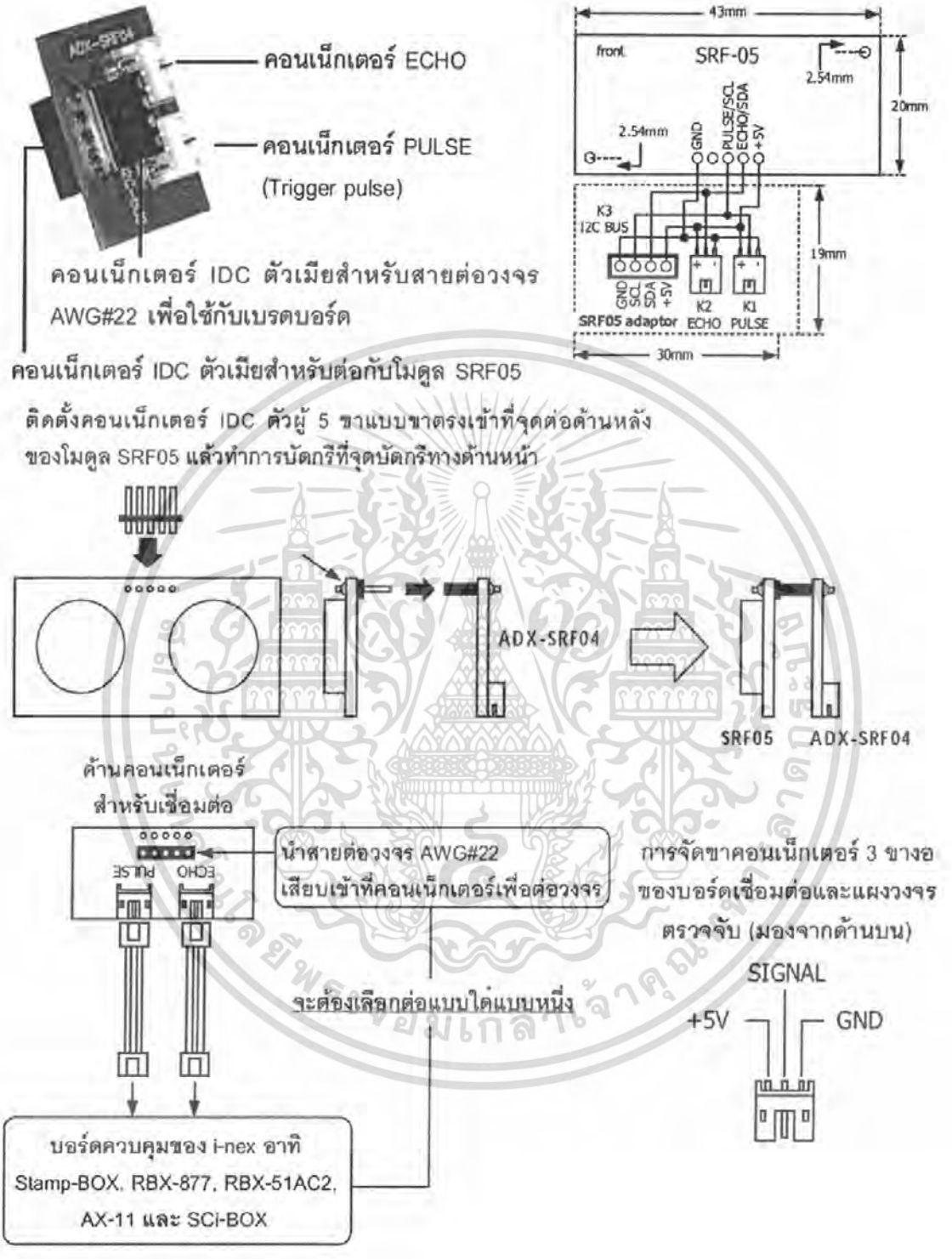
4. ขา MODE สำหรับเลือกรูปแบบการติดต่อกับ SRF05

ปล่อยลอยไว้ (NC) : เลือกให้ติดต่อแบบ 2 สัญญาณผ่านจุดต่อ ECHO และ TRIGGER

ต่อลงกราวด์ : เลือกให้ติดต่อแบบ 1 สัญญาณผ่านจุดต่อ TRIGGER

5. ขา GND สำหรับต่อกราวด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงวงจรของบอร์ด ADX-SRF05 และการเชื่อมต่อกับโมดูล SRF05

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

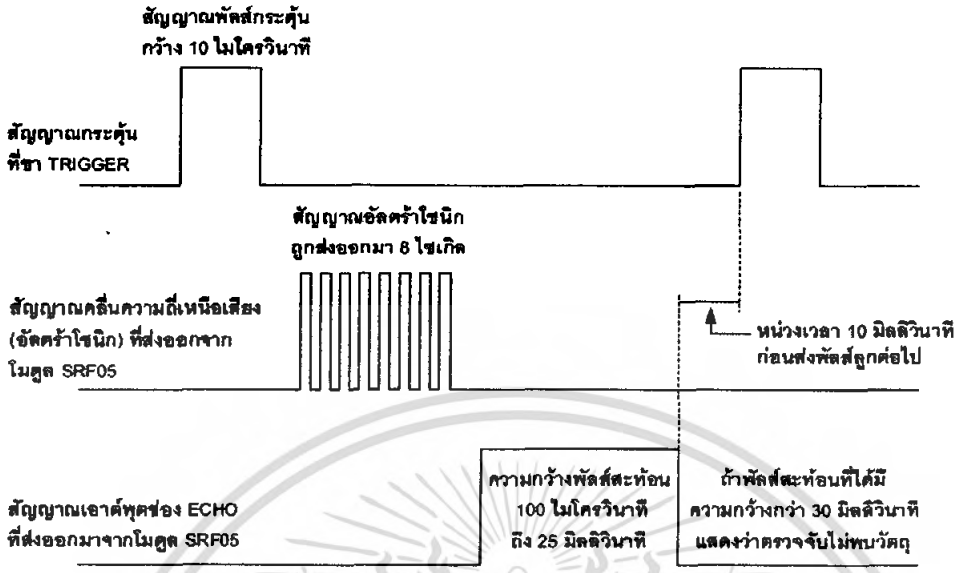
2.1.2 บอร์ด ADX-SRF05

เพื่ออำนวยความสะดวกแก่ผู้ใช้งานกับบอร์ดควบคุมหุ่นยนต์ของบริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด (i-nex:เป็นตัวแทนจำหน่ายสินค้าของ Devantechในประเทศไทยอย่างเป็นทางการ) จึงได้พัฒนาบอร์ดอะแดปเตอร์รุ่น ADX-SRF04 เพื่อให้นำโมดูล SRF05 มาติดตั้ง (จัดมาพร้อมกับสายเชื่อมต่อในชุดของ SRF05) เพื่อให้ใช้งานได้สะดวกขึ้นบนบอร์ด ADX-SRF05 ได้จัดเตรียมคอนเน็กเตอร์ PCB 3 ขาตัวผู้ 2 ตัวแยกกันระหว่างสัญญาณ ECHO และ TRIGGER สำหรับเชื่อมต่อกับบอร์ดควบคุมหุ่นยนต์และคอนเน็กเตอร์ IDC ตัวเมียแถวเดี่ยว 4 ขาสำหรับเสียบสายต่อวงจรเบอร์ AWG#22 เพื่อต่อกับแผงต่อวงจรหรือเบรดบอร์ดของ ADX-SRF05 และการติดตั้งเพื่อใช้งานแสดงในรูปที่ 2.3

2.1.3. การใช้งานกับไมโครคอนโทรลเลอร์

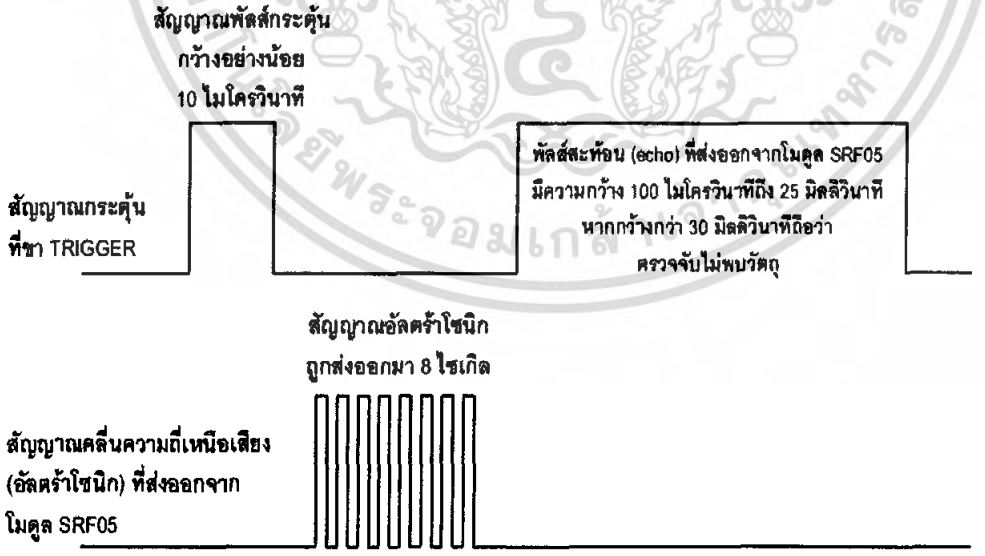
1.ฟังก์ชั่นโปรแกรมภาษา C สำหรับติดต่อระหว่าง AT89C52 กับโมดูล SRF05ในโหมดการติดต่อแบบ 2 สัญญาณในการเขียนโปรแกรมภาษา C สำหรับไมโครคอน โทรลเลอร์ AT89C52 เพื่อใช้งานโมดูล SRF05 สิ่งที่ควรทำคือสร้างฟังก์ชั่นในการติดต่อเพื่ออ่านค่าและประมวลผล ตัวอย่างที่นำมาเสนอในบทนี้คือ ฟังก์ชั่น distance ฟังก์ชั่นนี้นำมาใช้อ่านค่าระยะทางที่วัดได้จากโมดูล SRF05 เริ่มต้นด้วยการกำหนดให้ ขาพอร์ต P2.3 ของไมโครคอนโทรลเลอร์AT89C52 ต่อกับขา ECHO ของโมดูล SRF05 และกำหนดให้เป็นพอร์ตอินพุตโดยกำหนดให้มีลอจิก 1 เพื่อเริ่มต้นทำงานส่วนขา TRIGGER ของโมดูล SRF05 ต่อเขากับขาพอร์ต P2.4 ดังนั้นจึงต้องกำหนดให้เป็นเอาต์พุตเพื่อสร้างสัญญาณพัลส์บวกที่มีความกว้าง 5 ถึง 10 ไมโครวินาที เพื่อเป็นกระตุ้นให้โมดูล SRF05 เริ่มต้นกระบวนการวัดระยะทางทันทีที่โมดูล SRF05 ได้รับสัญญาณพัลส์บวกเข้าที่ขา TRIGGER โมดูล SRF05 จะดำเนินการสร้างขบวนพัลส์ความถี่ 40 kHz ออกสู่อากาศขบวนพัลส์ความถี่เหนือเสียงหรืออัลตร้าโซนิกกลุ่มนี้จะเดินทางเป็นเส้นตรงจนกระทั่งกระทบวัตถุที่ขวางอยู่เบื้องหน้า ทำให้ขบวนพัลส์อัลตร้าโซนิกสะท้อนกลับเข้ามายังตัวรับของโมดูล SRF05 หลังจากนั้นหน่วยประมวลผลภายในโมดูล SRF05 จะวิเคราะห์และให้สัญญาณเอาต์พุตออกมาเป็นความกว้างพัลส์บวกที่เป็นสัดส่วนโดยตรงกับระยะทางที่ตรวจจับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 แสดงไดอะแกรมเวลาแสดงสัญญาณที่ส่งไปยัง SRF05 และสัญญาณที่ตอบรับกลับมาจาก SRF05 ในกรณีติดต่อแบบ 2 สัญญาณ

ในกรณีติดต่อแบบ 2 สัญญาณ



รูปที่ 2.5 ไดอะแกรมเวลาแสดงสัญญาณที่ส่งไปยัง SRF05 และสัญญาณที่ตอบรับกลับมาจาก SRF05 ในกรณีแบบติดต่อแบบ 1 สัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ในกรณีติดต่อแบบ 1 สัญญาณ (ขุดต่อ MODE ต้องต่อลงกราวด์) สำหรับการนับความกว้างพัลส์ที่ขา ECHO จะใช้ไทเมอร์ 1 ในโหมด 16 บิต เป็นตัวนับเวลาความกว้างของสัญญาณพัลส์โดยจะเริ่มต้นนับเมื่อพบว่ามีการเปลี่ยนแปลงลอจิกแบบขาขึ้นจาก 0 ไปเป็น 1 และหยุดการนับเมื่อพบการเปลี่ยนแปลงลอจิกแบบขาลงจาก 1 ไปเป็น 0 หลังจากนั้นค่าที่ได้จากการนับนำไปหารด้วย 56 จะเป็นค่าประมาณที่ใกล้เคียงจากการแปลงข้อมูลระยะทางที่ได้จากโมดูล SRF05 ไทเมอร์ของไมโครคอนโทรลเลอร์ที่ใช้นั้นจะอ้างอิงสัญญาณนาฬิกาความถี่ 11.0592 MHz และไมโครคอนโทรลเลอร์ AT89C52 ทำงานด้วยความเร็ว 6 Clock

2.2 กล้องส่งภาพแบบไร้สาย

คุณสมบัติ

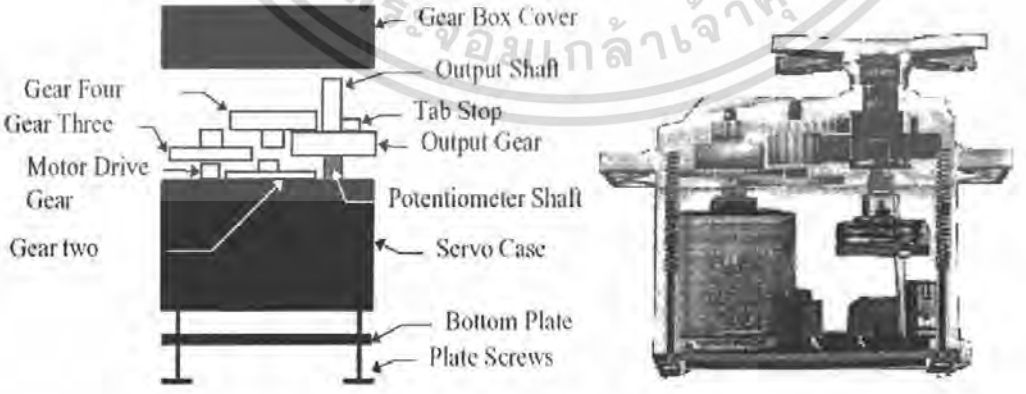
1. ตัวกล้องมีขนาด กว้าง 3.5 ซม. ยาว 3.5 ซม. หนา 2 ซม.
2. ความละเอียดของภาพอยู่ที่ 380 เส้น
3. เลนส์กล้องกว้าง 0.6 ซม. สามารถเห็นภาพมุมกล้องได้ดี
4. ส่งสัญญาณภาพสีแบบไร้สาย โดยมีเครื่องส่งฝังอยู่ในตัวกล้อง (ขนาด 1 ซม.)
5. มีขาตั้งกล้องติดกับตัวกล้อง สามารถขันน็อตยึดติดกับกำแพงหรือเพดานได้
   ตัวกล้องใช้ระบบไฟ Minimum 9 V. Maximum 12 V. ความถี่ 900-1200 KHz
6. ส่งภาพได้ไกล 50-100 เมตร

รูปที่ 2.6 แสดงกล้องส่งภาพแบบไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

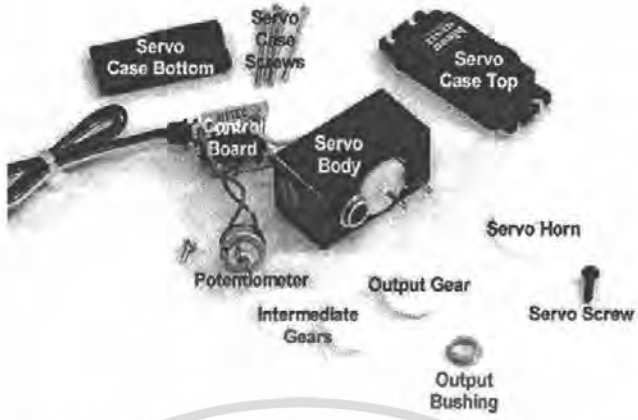
2.3 เซอร์โวมอเตอร์

เซอร์โวมอเตอร์ คือมอเตอร์ไฟฟ้ากระแสตรง (DC motor) ที่ถูกประกอบรวมกับชุดเฟืองและส่วนควบคุมต่างๆ ไว้ในโมดูลเดียวกันหรือภายในกล่องพลาสติกเดียวกัน โดยมอเตอร์ชนิดนี้จะมีสายต่อใช้งานเพียง 3 เส้นเท่านั้นคือ VCC,GND และสายสัญญาณควบคุม(Control Line) ซึ่งสามารถควบคุมให้มอเตอร์หมุนซ้ายหรือขวาได้จากสายสัญญาณเพียงเส้นเดียว โดยสัญญาณที่ใช้ควบคุมนี้จะเป็นสัญญาณพัลส์วิดมอด (PWM) แบบ TTL Level ระดับแรงดันที่จ่ายให้มอเตอร์นี้จะอยู่ในช่วงประมาณ 4 ถึง 6 โวลท์ ขึ้นอยู่กับคุณสมบัติของมอเตอร์แต่ละตัวข้อดีของมอเตอร์ชนิดนี้ก็คือจะมีขนาดเล็กน้ำหนักเบา,ให้แรงบิดสูง ,กินพลังงานน้อย และสามารถควบคุมด้วยแรงดันลอจิกที่เป็นTTLได้โดยตรงไม่จำเป็นต้องต่อวงจรขับ(Driver)อื่นๆ เพราะมอเตอร์ชนิดนี้จะมีวงจรควบคุมบรรจุไว้ภายในอยู่แล้ว ซึ่งมอเตอร์ชนิดนี้สามารถควบคุมให้หมุนไปในตำแหน่งหรือทิศทางองศาที่ต้องการได้โดยอาศัยสัญญาณความกว้างพัลส์ที่ป้อนให้มอเตอร์ แต่เซอร์โวมอเตอร์นี้จะหมุนได้แค่เพียงในช่วงประมาณ 180° หรือครึ่งรอบเท่านั้นหรือบางรุ่นอาจหมุนได้ถึง 210° แต่จะไม่สามารถหมุนเป็นวงรอบได้เนื่องจากโครงสร้างภายในจะประกอบด้วยตัวต้านทานชนิดปรับค่าได้ (VR) ที่ทำหน้าที่ตรวจสอบตำแหน่งการหมุนของมอเตอร์และตัวต้านทานนี้จะถูกยึดติดกับแกนหมุนของมอเตอร์ ซึ่งจากการที่ตัวต้านทานปรับค่านี้ไม่สามารถหมุนเป็นวงรอบได้ดังนั้นเซอร์โวมอเตอร์จึงถูกออกแบบให้หมุนได้เพียงแค่ประมาณ 180 องศาหรือครึ่งรอบเท่านั้นเพื่อป้องกันความเสียหายที่จะเกิดกับตัวต้านทานปรับค่าได้ แต่ถ้าหากเราต้องการให้มอเตอร์หมุนเป็นวงรอบ(360°)นั้นก็สามารถทำได้โดยจะต้องทำการปรับแต่ง (Modify) ดัดแปลงชิ้นส่วนบางอย่างของมอเตอร์



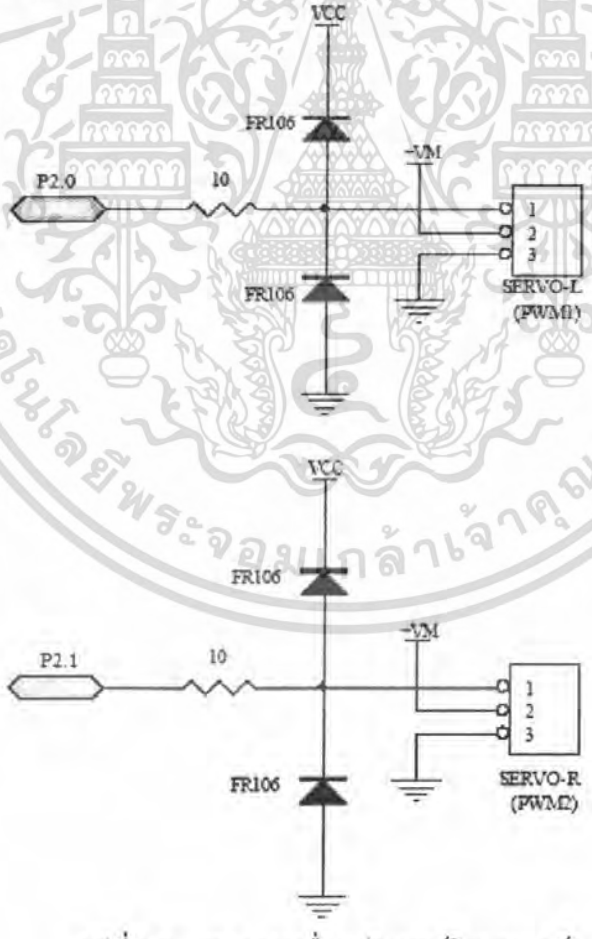
รูปที่ 2.7 แสดงโครงสร้างชุดเซอร์โวมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงส่วนประกอบของเซอร์โวมอเตอร์

2.3.1 การเชื่อมต่อเซอร์โวมอเตอร์



รูปที่ 2.9 แสดงการเชื่อมต่อเซอร์โวมอเตอร์

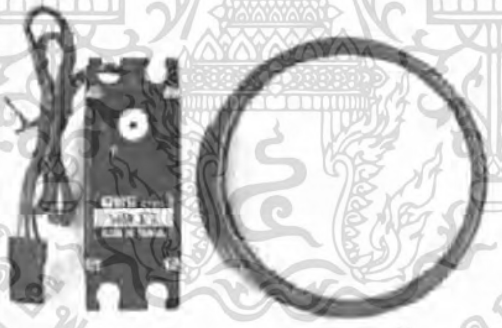
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 การควบคุมการเคลื่อนที่ของหุ่นยนต์

อุปกรณ์สำหรับขับเคลื่อนตัวรถหุ่นยนต์นั้นนับว่าเป็นอุปกรณ์ที่มีความสำคัญมาก ทั้งนี้ก็เนื่องมาจากว่าการที่จะทำให้ตัวรถสามารถขับเคลื่อนไปในทิศทางต่างๆ ได้นั้นจะต้องอาศัยตัวขับเคลื่อนพาไป สำหรับอุปกรณ์ที่ใช้ในการขับเคลื่อนตัวรถหุ่นยนต์นั้นจะอาศัยมอเตอร์แบบ DC SERVO MOTOR เป็นตัวขับเคลื่อนซึ่งการที่รถจะสามารถเคลื่อนที่ไปในทิศทางต่างๆ ได้นั้นก็จะขึ้นอยู่กับทิศทางการหมุนของ DC SERVO MOTOR เป็นหลักด้วยเหตุนี้เองในอันดับแรกก่อนที่จะเริ่มต้นทำการเขียนโปรแกรมเพื่อสั่งงานให้ DC SERVO MOTOR หมุนได้นั้นเราจำเป็นจะต้องทราบและเข้าใจถึงหลักการควบคุมมอเตอร์ชนิดนี้เสียก่อนสำหรับวิธีการควบคุมการทำงานของ DC SERVO MOTOR นั้นมีรายละเอียดดังต่อไปนี้

หลักการทำงานของ DC SERVO MOTOR

ตามปกติทั่วๆ ไปแล้วนั้น DC SERVO MOTOR นั้นจะสามารถหมุนไปในทิศทางตามเข็มนาฬิกาและทวนเข็มนาฬิกาได้เพียงแค่180 องศาหรือครึ่งวงกลมเท่านั้นซึ่งวิธีการสั่งงานให้ SERVO หมุนไปใน ตำแหน่งใดๆนั้นจะอาศัยสัญญาณ Pulse เป็นตัวบ่งบอก

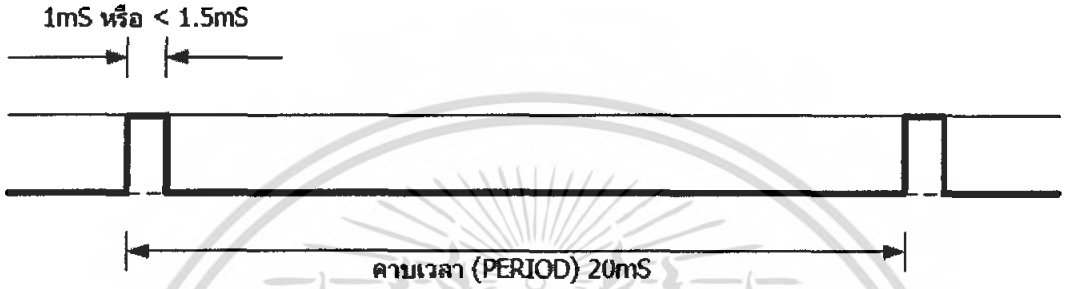


รูปที่ 2.10 แสดง DC SERVO MOTOR

แต่สำหรับ DC SERVO MOTOR ที่เรานำมาใช้ในการขับเคลื่อนล้อเพื่อนำพาตัวรถให้เคลื่อนที่ไปในทิศทางต่างๆได้นั้นจะต้องทำการดัดแปลงหรือ Modify ให้สามารถหมุนได้รอบตัวหรือ 360 องศา เสียก่อนโดย DC SERVO MOTOR ที่ได้นำมาใช้นั้น ได้ผ่านการปรับแต่งการทำงานของมอเตอร์ให้สามารถหมุนเป็นวงรอบ (360 องศา) ได้เป็นที่เรียบร้อยแล้วโดยวิธีในการควบคุมให้มอเตอร์ ซึ่งทำการดัดแปลงแล้วให้หมุนไปในทิศทางต่างๆนั้น จะมีลักษณะดังนี้

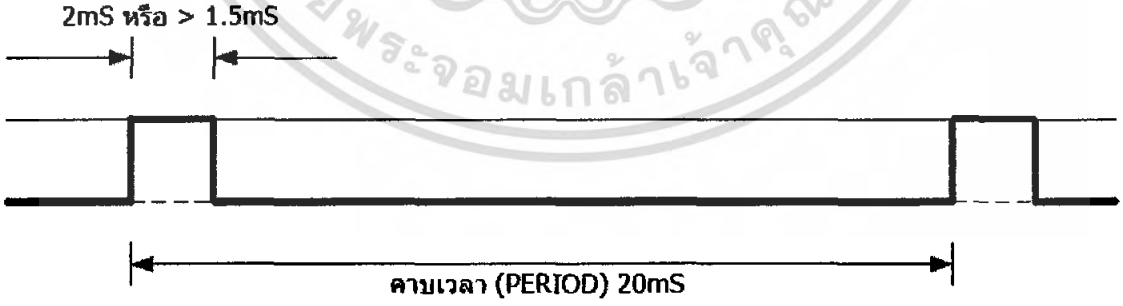
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การควบคุมให้มอเตอร์หมุนทางด้านซ้ายหรือหมุนตามทิศทางทวนเข็มนาฬิกาจะต้องป้อนสัญญาณ Pulse ที่มีขนาดความกว้างด้านบวก 1 mS หรือ ให้น้อยกว่า 1.5 mSโดยจะต้องป้อนสัญญาณ Pulse นี้ทุกๆ 20 mS (หรือในช่วงประมาณ 20mS – 30mS) เพื่อให้มอเตอร์หมุนต่อเนื่องไปในทิศทางเดิม



รูปที่ 2.11 แสดงลักษณะของ Pulse สำหรับควบคุมให้มอเตอร์หมุนทวนเข็มนาฬิกา

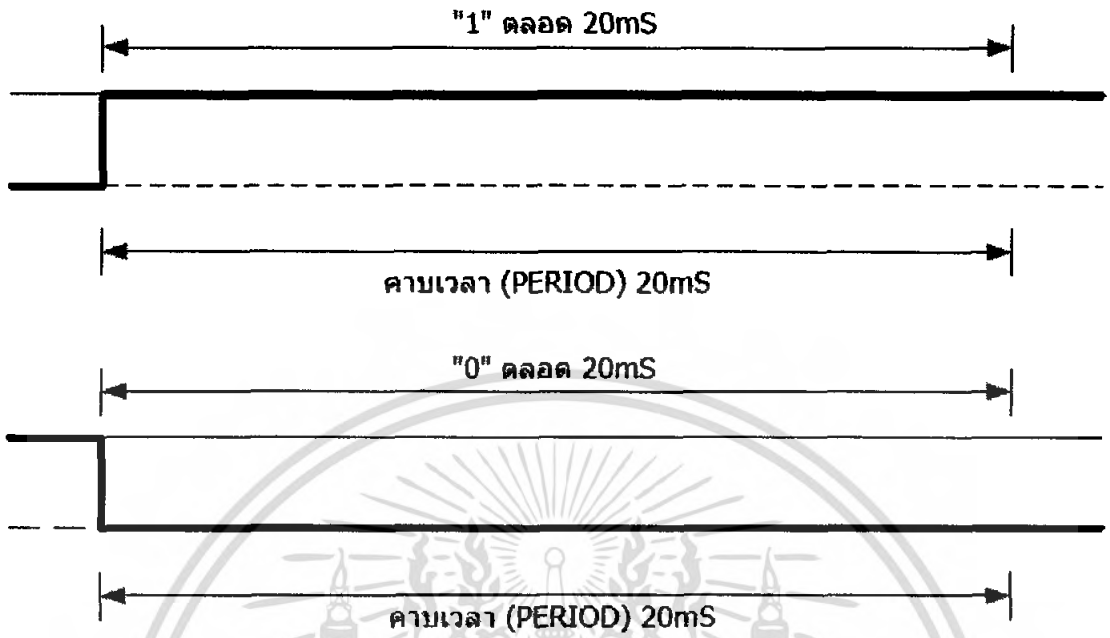
2. การควบคุมให้มอเตอร์หมุนทางด้านขวาหรือทิศทางตามเข็มนาฬิกาจะต้องป้อนสัญญาณ Pulse ที่มีขนาดความกว้างด้านบวก 2 mS หรือ ไม่ต่ำกว่า 1.5 mS และจะต้องป้อนสัญญาณPulseนี้ ทุกๆ 20 mS (หรือในช่วงประมาณ 20mS – 30mS) เพื่อควบคุมให้มอเตอร์หมุนต่อเนื่องไปในทิศทางเดิม



รูปที่ 2.12 แสดงลักษณะของ Pulse สำหรับควบคุมให้มอเตอร์หมุนตามเข็มนาฬิกา

การควบคุมให้มอเตอร์หยุดหมุนทำได้โดยการส่งโลจิก “0” หรือ “1” ให้กับมอเตอร์ตลอดคาบเวลาที่ต้องการให้มอเตอร์หยุดหมุน ซึ่งก็คือการไม่จ่ายสัญญาณ Pulse ให้กับมอเตอร์นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แสดงลักษณะของ Pulse สำหรับควบคุมให้มอเตอร์หยุดหมุน (STOP)

ซึ่งจะเห็นได้ว่าในการควบคุมการทำงานของมอเตอร์นั้นจะใช้วิธีการสร้างสัญญาณ Pulse เพื่อส่งไปบังคับตัวมอเตอร์ให้หมุนไปในทิศทางต่างๆตามต้องการ ซึ่งในกรณีของรถหุ่นยนต์นั้นจะต้องทำการสร้างสัญญาณ Pulse ซึ่งมีขนาดความกว้างของสัญญาณด้านบวก 1 mS หรือ 2 mS เพื่อควบคุมให้มอเตอร์หมุนไปในทิศทางทวนเข็มนาฬิกาหรือตามเข็มนาฬิกาตามต้องการและเมื่อต้องการให้มอเตอร์ทำการขับเคลื่อนล้อเพื่อนำพาตัวรถให้เคลื่อนที่ไปในทิศทางเดิมต่อเนื่องกันไปนั้นก็จะต้องทำการส่งสัญญาณ Pulseแบบเดียวกันซ้ำๆออกไปภายในเวลาประมาณทุกๆ 20mS ด้วยเสมอ

ซึ่งจากคุณสมบัติของสัญญาณ Pulse ดังกล่าวข้างต้นนั้นก็คือลักษณะของสัญญาณ Pulse Width Modulation หรือ PWM นั่นเอง ดังนั้นจะได้ว่าเมื่อต้องการให้มอเตอร์หมุนก็จะต้องทำการส่งสัญญาณ Pulse Width Modulation หรือ PWM ซึ่งมีคาบเวลา 20mS ออกไปให้กับมอเตอร์ ซึ่งก็คือค่า Period ของสัญญาณ PWM นั่นเอง ส่วนทิศทางการหมุนของมอเตอร์นั้นจะกำหนดจากความกว้างของสัญญาณ Pulse ในขณะที่เป็นบวกอยู่ ซึ่งก็คือค่า Duty Cycle ของสัญญาณ PWM นั่นเอง ดังนั้นเมื่อต้องการควบคุมการหมุนของ DC SERVO MOTOR ด้วยสัญญาณ PWM ก็สามารถสรุปได้ว่า

1. เมื่อต้องการให้มอเตอร์หมุนไปในทิศทางตามเข็มนาฬิกา ก็จะต้องสร้างสัญญาณ PWM ที่มีค่า Period ขนาด 20mS โดยมีค่า Duty Cycle ของสัญญาณ 1mS
2. เมื่อต้องการให้มอเตอร์หมุนไปในทิศทางทวนเข็มนาฬิกาก็จะต้องสร้างสัญญาณ PWM

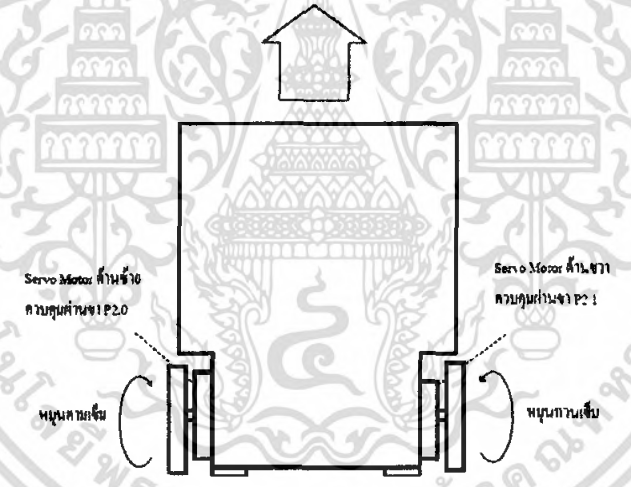
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่มีค่า Period ขนาด 20 mS โดยมีค่า Duty Cycle ของสัญญาณ 2 mS

3. เมื่อต้องการให้มอเตอร์หยุดหมุนก็จะต้องทำการหยุดการสร้างสัญญาณ PWM ซึ่งทำได้โดยการส่งสัญญาณที่มีสภาวะเป็นโลจิก "0" หรือ "1" ให้กับตัวมอเตอร์ก็ได้ซึ่งในทางปฏิบัติควรป้อนเป็นโลจิก"0"ให้กับมอเตอร์จะดีที่สุดเนื่องจากจะไม่เกิดปัญหาหรือความผิดพลาดเมื่อต้องการสั่งให้มอเตอร์เริ่มหมุนใหม่ในครั้งต่อไป

2.3.3 การควบคุมการเคลื่อนที่ของตัวรถหุ่นยนต์

เราได้ศึกษาถึงวิธีการควบคุมการทำงานของ DC SERVO MOTOR ในการบังคับล้อให้หมุนไปในทิศทางตามเข็มนาฬิกาหรือทวนเข็มนาฬิกาหรือหยุดหมุนกันมาแล้ว ซึ่งลำดับต่อไปนี้เราจะมาทำความเข้าใจเกี่ยวกับวิธีการควบคุมทิศทางการเคลื่อนที่ของตัวรถหุ่นยนต์ให้เคลื่อนที่ไปในทิศทางต่างๆไม่ว่าจะเป็นเดินหน้าถอยหลัง เลี้ยวซ้าย หรือเลี้ยวขวา เป็นต้น



รูปที่ 2.14 แสดงลักษณะการเคลื่อนที่ของรถหุ่นยนต์โดยการขับเคลื่อนจาก DC SERVO MOTOR

จากรูปจะเห็นได้ว่าลักษณะการติดตั้งตัว DC SERVO MOTOR เข้ากับล้อของรถหุ่นยนต์นั้นจะสังเกตเห็นได้ว่าตัวมอเตอร์จะถูกจัดวางอยู่ในทิศทางที่ตรงกันข้าม ดังนั้นในการควบคุมการเคลื่อนที่ของตัวรถหุ่นยนต์ให้เคลื่อนที่ไปในทิศทางเดียวกันนั้นตัวมอเตอร์ที่อยู่ทางด้านซ้ายและขวาจะต้องทำงานในทิศทางที่ตรงกันข้ามกัน โดยมอเตอร์ที่ติดตั้งอยู่ทางด้านล้อขวาจะใช้วิธีการควบคุมแบบปกติ ส่วนมอเตอร์ที่ติดตั้งอยู่กับล้อด้านซ้ายจะใช้วิธีการควบคุมแบบกลับทางหรือตรงกันข้ามกับทิศทางการเคลื่อนที่ที่ต้องการ ตัวอย่างเช่นเมื่อต้องการให้ตัวรถขับเคลื่อนไปข้างหน้าก็จะต้องบังคับให้มอเตอร์ตัวที่บังคับการหมุนของล้อทางด้านขวาหมุนไปในทิศทางตามเข็มนาฬิกาเพื่อ

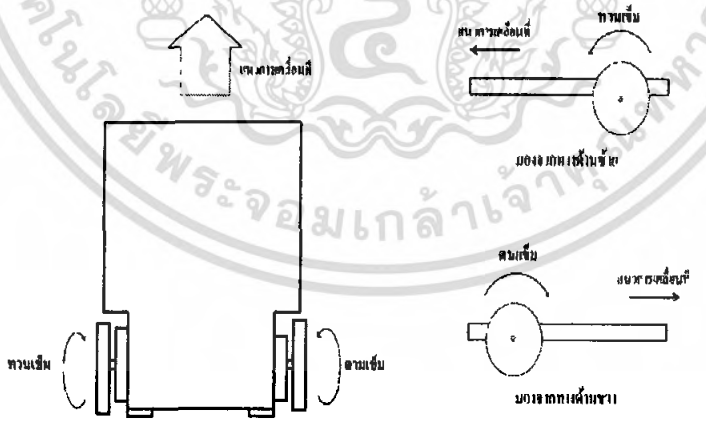
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บังคับให้ล้อขวาหมุนไปข้างหน้าส่วนมอเตอร์ตัวที่ทำหน้าที่บังคับการหมุนของล้อด้านซ้ายก็จะต้องถูกบังคับให้หมุนไปในทิศทางตรงกันข้าม คือทวนเข็มนาฬิกาเพื่อบังคับให้ล้อด้านซ้ายหมุนไปข้างหน้าเป็นต้น สำหรับวิธีการบังคับเลี้ยวของรถหุ่นยนต์นั้นจะแตกต่างจากการบังคับเลี้ยวของรถยนต์ที่เราพบเห็นอยู่ทั่วไปในชีวิตประจำวัน เนื่องจากรถหุ่นยนต์ไม่มีระบบพวงมาลัยในการบังคับล้อให้เลี้ยวซ้ายหรือขวาตามต้องการเหมือนรถยนต์ทั่วๆไป แต่อย่างไรก็ตามเราสามารถทำการบังคับให้รถหุ่นยนต์ทำการเลี้ยวไปในทิศทางที่เราต้องการได้เช่นเดียวกัน โดยใช้วิธีการควบคุมการหมุนของล้อหลังซึ่งเป็นตัวขับเคลื่อนตัวรถหุ่นยนต์ เช่นเมื่อต้องการให้รถหุ่นยนต์เลี้ยวไปทางด้านซ้ายก็สามารถทำได้โดยการบังคับให้ล้อซ้ายหยุดหมุนส่วนล้อขวาก็ให้เคลื่อนที่ไปข้างหน้าตามปกติหรือถ้าต้องการให้รถเลี้ยวซ้ายอย่างรวดเร็วก็สามารถทำได้โดยการบังคับให้ล้อซ้ายหมุนถอยหลังส่วนล้อขวาก็ให้หมุนไปข้างหน้าอย่างนี้ เป็นต้น

2.3.4 ลักษณะการทำงาน

1.การบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหน้า

สำหรับวิธีการบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหน้านั้นสามารถทำได้โดยการบังคับให้มอเตอร์ที่ใช้ควบคุมการหมุนของล้อซ้ายหมุนแบบทวนเข็มนาฬิกาหรือถอยหลัง (ส่ง Pulse ความกว้าง 2 mS) ส่วนมอเตอร์ที่ใช้ควบคุมการหมุนของล้อขวาก็ต้องบังคับให้หมุนแบบตามเข็มนาฬิกาหรือเดินหน้า (ส่ง Pulse ความกว้าง 1 mS) ดังรูปที่ 2.15



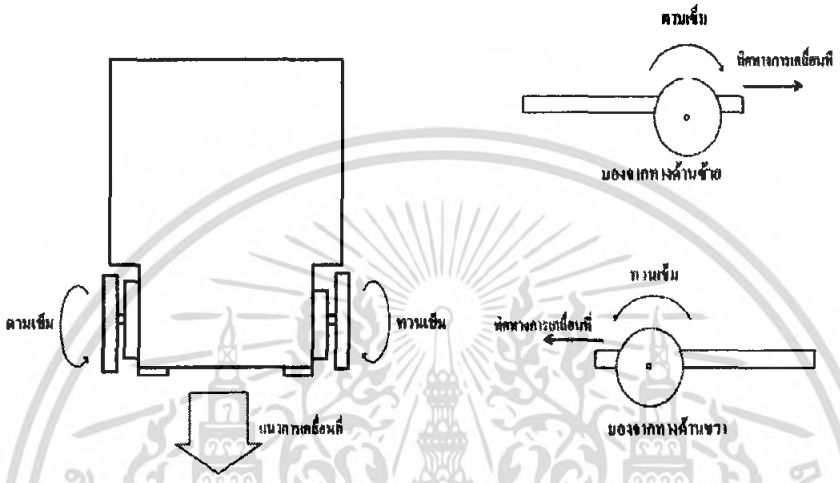
รูปที่ 2.15 แสดงการบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหน้า

2. การบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหลัง

สำหรับวิธีการบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหลังนั้นจะมีวิธีการตรงกันข้ามกับการเคลื่อนที่ไปข้างหน้าซึ่งก็คือการกลับทิศทางการหมุนของมอเตอร์ทั้งสองตัวให้เป็นตรงข้ามกับการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

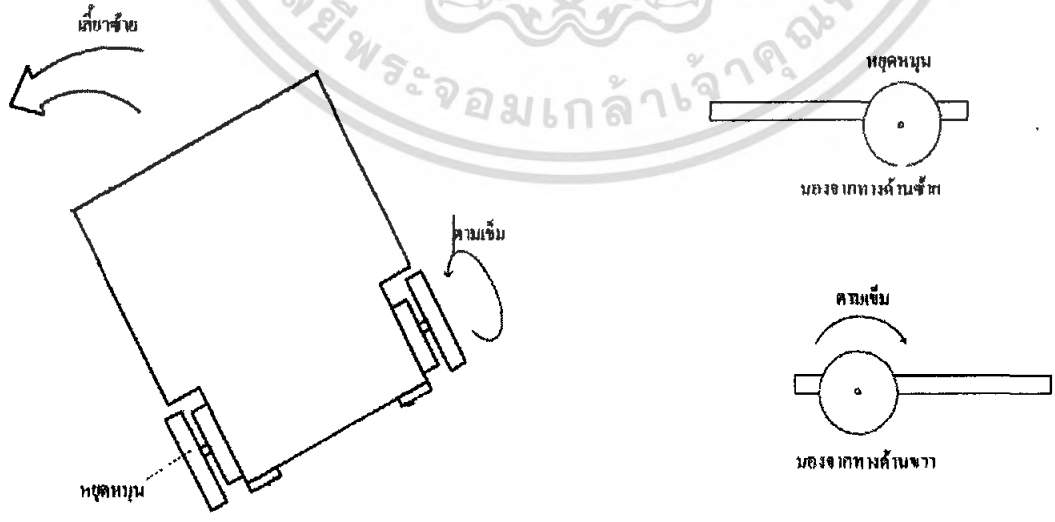
เคลื่อนที่ไปข้างหน้านั่นเองซึ่งสามารถทำได้โดยการบังคับให้มอเตอร์ที่ใช้ควบคุมการหมุนของล้อซ้ายหมุนแบบตามเข็มนาฬิกาหรือเดินหน้า (ส่ง Pulse ความกว้าง 1 mS)ส่วนมอเตอร์ที่ใช้ควบคุมการหมุนของล้อขวาก็ต้องบังคับให้หมุนแบบทวนเข็มนาฬิกาหรือถอยหลัง (ส่ง Pulse ความกว้าง 2 mS)



รูปที่ 2.16 แสดงการบังคับให้รถหุ่นยนต์เคลื่อนที่ไปข้างหลัง (ถอยหลัง)

3. การบังคับให้รถหุ่นยนต์เลี้ยวซ้ายแบบปกติ

สำหรับวิธีการบังคับให้ตัวรถหุ่นยนต์เลี้ยวซ้ายแบบนี้จะอาศัยล้อซ้ายเป็นจุดหมุนของการเลี้ยวซ้าย โดยวิธีการในการบังคับให้รถหุ่นยนต์เลี้ยวซ้ายแบบนี้จะต้องทำการหยุดการหมุนของล้อทางด้านซ้ายส่วนล้อทางด้านขวาก็ปล่อยให้หมุนไปข้างหน้าตามปกติ ดังรูปที่ 2.17

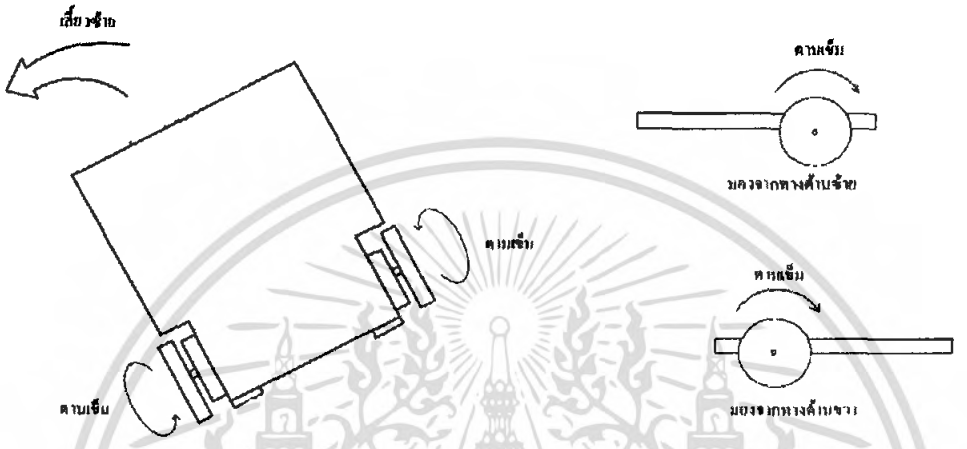


รูปที่ 2.17 แสดงการบังคับให้รถหุ่นยนต์เลี้ยวซ้ายโดยการเคลื่อนที่ของล้อขวาอย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
82923

4. การบังคับให้รถหุ่นยนต์เลี้ยวซ้ายแบบรวดเร็ว

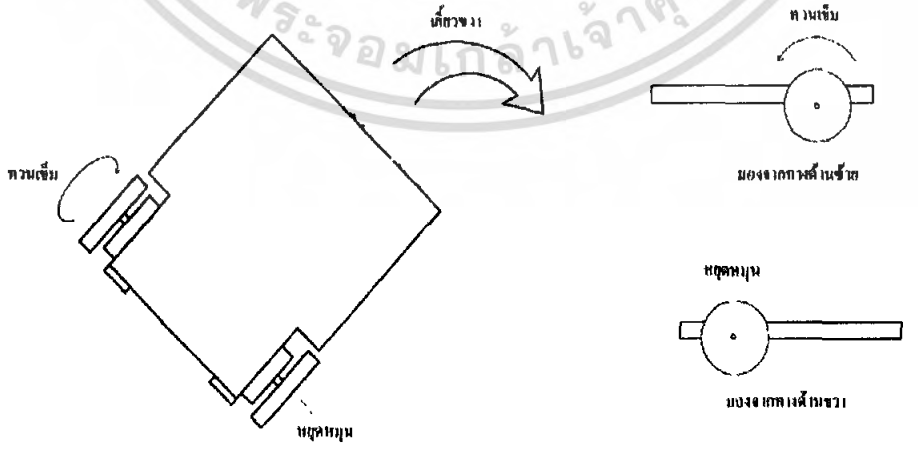
สำหรับวิธีการบังคับเลี้ยวแบบนี้สามารถกระทำได้โดยการบังคับให้ล้อทางด้านซ้ายและขวาหมุนไปในทิศทางที่ตรงกันข้ามกล่าวคือ ล้อทางด้านซ้ายจะถูกบังคับให้หมุนถอยหลังส่วนล้อทางด้านขวาจะถูกบังคับให้เดินไปทางหน้า ดังรูปที่ 2.18



รูปที่ 2.18 แสดงการบังคับให้รถหุ่นยนต์เลี้ยวซ้ายโดยให้ล้อซ้ายถอยหลังส่วนล้อขวาเดินหน้า

5. การบังคับให้รถหุ่นยนต์เลี้ยวขวาแบบปกติ

สำหรับวิธีการบังคับให้ตัวรถหุ่นยนต์เลี้ยวขวาแบบนี้จะอาศัยล้อขวาเป็นจุดหมุนของการเลี้ยวโดยวิธีการในการบังคับให้รถหุ่นยนต์เลี้ยวขวาแบบนี้จะต้องทำการหยุดการหมุนของล้อทางด้านขวาส่วนล้อทางด้านซ้ายก็ปล่อยให้หมุนไปข้างหน้าตามปรกติ ดังรูปที่ 2.19

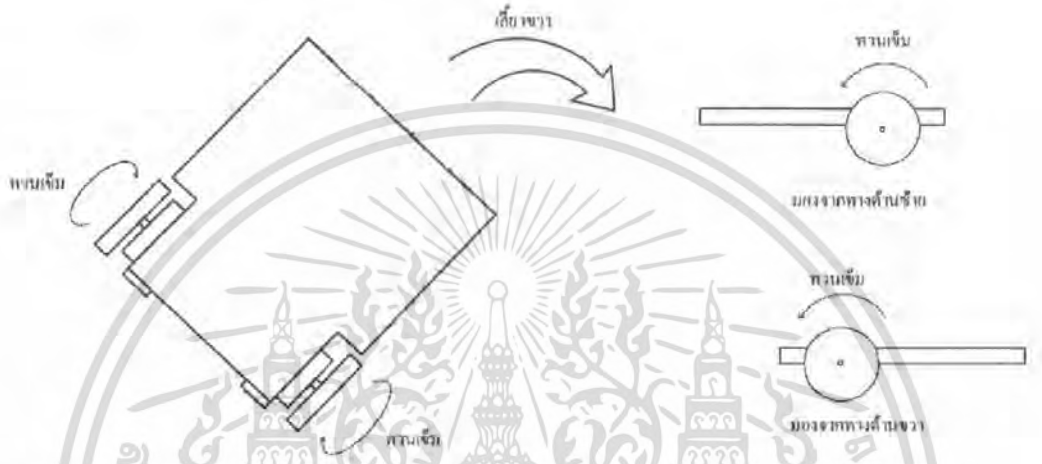


รูปที่ 2.19 แสดงการบังคับให้รถหุ่นยนต์เลี้ยวขวาโดยการเคลื่อนที่ของล้อซ้ายอย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.การบังคับให้รถหุ่นยนต์เลี้ยวขวาแบบรวดเร็ว

สำหรับวิธีการบังคับเลี้ยวแบบนี้สามารถกระทำได้โดยการบังคับให้ล้อทางด้านซ้ายและขวาหมุนไปในทิศทางที่ตรงกันข้ามกล่าวคือ ล้อทางด้านขวาจะถูกบังคับให้หมุนถอยหลังส่วนล้อทางด้านซ้ายจะถูกบังคับให้เดินไปทางหน้า ดังรูปที่ 2.20



รูปที่ 2.20 แสดงการบังคับให้รถหุ่นยนต์เลี้ยวขวาโดยให้ล้อขวาถอยหลังส่วนล้อซ้ายเดินหน้า

2.4 พอร์ตอนุกรม

รูปที่ 2.21 แสดงการสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมนับว่ามีความสำคัญต่อการใช้งานไมโครคอนโทรลเลอร์มากเพราะสามารถใช้แป้นพิมพ์และจอภาพของ PC เป็นอินพุตและเอาต์พุตในการติดต่อหรือควบคุมไมโครคอนโทรลเลอร์ด้วยสัญญาณอย่างน้อย เพียง 3 เส้นเท่านั้น คือ

1. สายส่งสัญญาณ TX
2. สายรับสัญญาณ RX
3. สาย GND

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

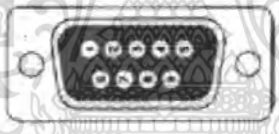
โดยปกติพอร์ตอนุกรม RS-232C จะสามารถต่อสายได้ยาว 50 ฟุตโดยประมาณขึ้นอยู่กับชนิดของสายสัญญาณ, ระยะทาง, และปริมาณสัญญาณรบกวน

พอร์ตอนุกรมของ PC
DB9 ตัวผู้ (Male)
(ก)

พอร์ตอนุกรมของอุปกรณ์ภายนอก
DB9 ตัวเมีย (Female)
(ข)

รูปที่ 2.22 แสดงพอร์ตอนุกรมของ PC และของอุปกรณ์ภายนอก

พอร์ตอนุกรมของ PC จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวผู้ (Male) รูปที่ 2.22 (ก)
พอร์ตอนุกรม ของอุปกรณ์ภายนอก จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวเมีย (FeMale) รูปที่ 2.22 (ข)



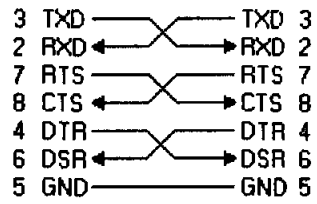
รูปที่ 2.23 แสดง DB9 ตัวผู้เมื่อมองจากด้านหลัง

ตารางที่ 2.1 แสดง Description แต่ละชนิด

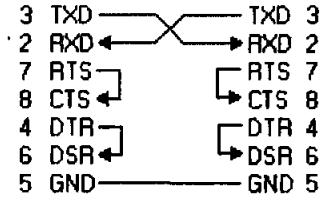
Pin	Description	Type
1	Data Carrier Detect (DCD)	Input
2	Received Data (RXD)	Input
3	Transmitted Data (TXD)	Output
4	Data Terminal Ready (DTR)	Output
5	Signal Ground (GND)	Input
6	Data Set Ready (DSR)	Input
7	Request To Send (RTS)	Output
8	Clear to Send (CTS)	Input
9	Ring Indicator (RI)	Input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย DB9



การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB9 แบบ Null modem



การต่ออุปกรณ์ภายนอกผ่าน DB9 แบบ 3 เส้น

รูปที่ 2.24 แสดงการเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย DB9

2.4.1 การทำงานของขาสัญญาณ DB9

TXD เป็นขาที่ใช้ส่งข้อมูล

RXD เป็นขาที่ใช้รับข้อมูล

DTR แสดงสภาวะพอร์ตว่าเปิดใช้งาน ,DSR ตรวจสอบว่าพอร์ตที่ติดต่อด้วยเปิดอยู่หรือไม่เมื่อเปิดพอร์ตอนุกรม ขา DTR จะ ON เพื่อให้อุปกรณ์ได้รับทราบว่าต้องการติดต่อด้วยในขณะเดียวกันก็จะตรวจสอบขา DSR ว่าอุปกรณ์พร้อมหรือไม่

RTS แสดงสภาวะพอร์ตว่าต้องการส่งข้อมูล ,CTS ตรวจสอบว่าพอร์ตที่ติดต่ออยู่ต้องการส่งข้อมูลหรือไม่เมื่อต้องการส่งข้อมูลขา RTS จะ ON และจะส่งข้อมูลออกที่ขาTXD เมื่อส่งเสร็จก็จะ OFF ในขณะเดียวกันก็จะตรวจสอบขา CTS ว่าอุปกรณ์ต้องการที่จะส่งข้อมูลหรือไม่

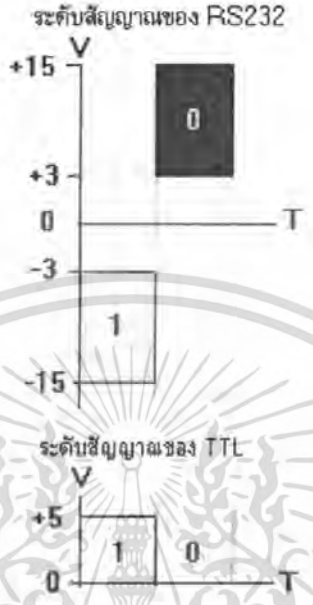
GND ขา ground

2.4.2 อัตราการส่งข้อมูล (Baud rate)

1. ความเร็วของการรับ-ส่งข้อมูล เป็นจำนวนบิทต่อวินาทีเช่น 300, 1,200, 2,400, 4,800 , 9,600 ,14,400 ,19,200, 38,400 ,56,000 เป็นต้น
2. การเลือกอัตราการส่งข้อมูลขึ้นอยู่กับชนิดของสายสัญญาณ, ระยะทาง,และปริมาณสัญญาณรบกวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 ระดับสัญญาณของ RS232



รูปที่ 2.25 แสดงระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL

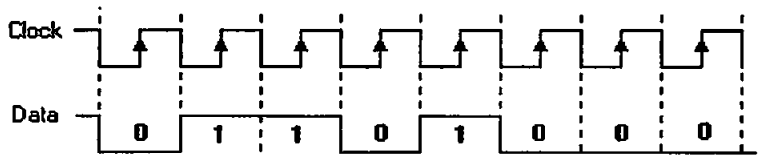
สัญญาณรบกวนที่เกิดขึ้นในสายนำสัญญาณมักจะมีแรงดันเป็นบวกเมื่อเทียบกับกราวด์ ป้องกันสัญญาณรบกวนนี้ จึงออกแบบแรงดันของโลจิก "1" เป็นลบ คืออยู่ในช่วง –3V ถึง -15V ส่วนแรงดัน ของโลจิก "0" อยู่ในช่วง +3V ถึง +15V เหตุที่ระดับสัญญาณของ RS232 อยู่ในช่วง +15V ถึง -15V ก็เพื่อให้ต่อสายสัญญาณไปได้ไกลขึ้น ดังนั้นจึงจำเป็นจะต้องมีวงจรเปลี่ยนระดับแรงดันของ RS232 มาเป็นระดับแรงดันของ TTL

2.4.4 รูปแบบการสื่อสารแบบอนุกรม

มีด้วยกันอยู่ 2 แบบ คือแบบซิงโครนัส (Synchronous) และแบบอะซิงโครนัส (Asynchronous)

1. การสื่อสารแบบซิงโครนัส (Synchronous) การรับส่งข้อมูลจะมีสัญญาณนาฬิกาซึ่งเป็นตัวกำหนดจังหวะเวลาการส่งข้อมูลร่วมอยู่ด้วยอีกเส้นหนึ่งใช้คู่กับสัญญาณข้อมูล ตัวอย่างเช่นการส่งสัญญาณจากคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

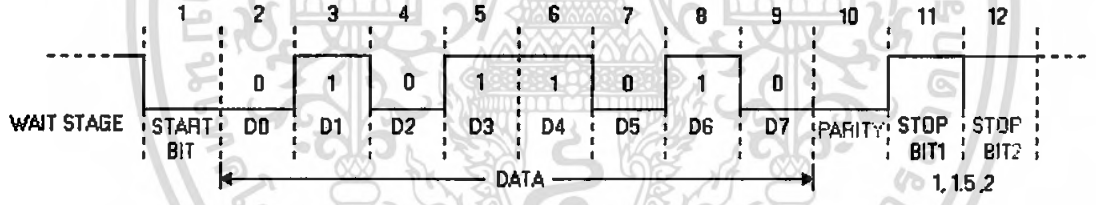


รูปที่ 2.26 แสดงการสื่อสารแบบซิงโครนัส

2. การสื่อสารแบบอะซิงโครนัส (Asynchronous) การรับส่งข้อมูลโดยที่ไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วยแต่จะใช้ให้ตัวส่งและตัวรับมีอัตราส่งข้อมูลที่เท่ากัน

1. รูปแบบข้อมูลแบบอะซิงโครนัสประกอบด้วย 4 ส่วน คือ

1. บิตเริ่มต้น (Start bit) มีขนาด1 บิต
2. บิตข้อมูล (Data) มีขนาด 5,6,7 หรือ 8 บิต
3. บิตตรวจสอบพาริตี้ (Parity bit) มีขนาด 1 บิตหรือไม่มี
4. บิตหยุด (Stop bit) มีขนาด 1, 1.5, 2 บิต



รูปที่ 2.27 แสดงการสื่อสารแบบอะซิงโครนัส

เมื่อไม่มีการส่งข้อมูลขา data จะมีสถานะเป็นโลจิก "1" หรือสถานะหยุดรอ (Waiting stage) เมื่อเริ่มต้นส่งข้อมูลจะให้ขา data เป็นโลจิก "0" เป็นจำนวน 1 บิต เรียกว่าบิตเริ่มต้น (Start bit) จากนั้นก็จะเริ่มต้นส่งข้อมูลโดยส่งบิตต่ำไปก่อน (LSB)แล้วตามด้วยพาริตี้บิต (จะมีหรือไม่ก็ได้ขึ้นอยู่กับการติดตั้งค่าของทั้งสองฝ่าย) สุดท้ายตามด้วยโลจิก "1" อย่างน้อย 1 บิต (มีขนาด 1, 1.5, หรือ 2 บิต) เพื่อแสดงว่าสิ้นสุดข้อมูล

3. การรับและส่งข้อมูลแบบอนุกรมยังแบ่งออกเป็นลักษณะการใช้งานได้ 3 แบบ คือ

1. แบบซิมเพลกซ์ (Simplex) เป็นการส่ง หรือรับข้อมูลแบบทิศทางเดียวเท่านั้น
2. แบบฮาล์ฟดูเพลกซ์ (Half Duplex) เป็นการส่งและรับข้อมูลแบบสลับกันคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อด้านหนึ่งส่งอีกด้านหนึ่งเป็นฝ่ายรับสลับกันไม่สามารถรับ-ส่งในเวลาเดียวกันได้

3. แบบฟลูดูเพลกซ์ (Full Duplex) สามารถรับ-ส่งข้อมูลในเวลาเดียวกันได้

2.5 ไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีด้วยกันหลายเบอร์ขึ้นกับโครงสร้างภายในของมัน บางเบอร์จะมีหน่วยความจำภายในเป็นแบบ ROM บางเบอร์เป็นแบบ EPROM บางเบอร์มี RAM ภายใน 128 ไบต์ บางเบอร์มี 256 ไบต์ เป็นต้น ซึ่งรายละเอียดสามารถศึกษาได้จากคู่มือได้โดยตรง และลักษณะของขาต่างๆ จะเหมือนกัน

2.5.1คุณสมบัติที่สำคัญของ MCS-51 มีดังนี้

1. มีหน่วยความจำ ROM 4 Kbyte, 8 Kbyte, 20 Kbyte
2. มีหน่วยความจำ RAM 128 byte
3. มีพอร์ท I/O ขนาด 8 บิต 4 พอร์ท
4. มี Timer 16 บิต 2 ตัว
5. สามารถอินเทอร์รัพท์ได้ 5 แหล่ง
6. มีวงจรออสซิลเลเตอร์และวงจรนาฬิกาบนชิพ
7. มีพอร์ทอนุกรมที่สามารถรับส่งข้อมูลแบบ Full Duplex ด้วยความเร็วสูง
8. อ้างหน่วยความจำโปรแกรมภายนอกได้ 64K
9. อ้างหน่วยความจำข้อมูลภายนอกได้ 64K
10. สามารถประมวลผลทีละบิตได้
11. สามารถอ้างหน่วยความจำแบบบิตได้ 210 ตำแหน่ง
12. หนึ่งวัฏจักรคำสั่งกินเวลาประมาณ 1 ไมโครวินาทีขณะทำงานด้วย Clock 11.0592 MHz

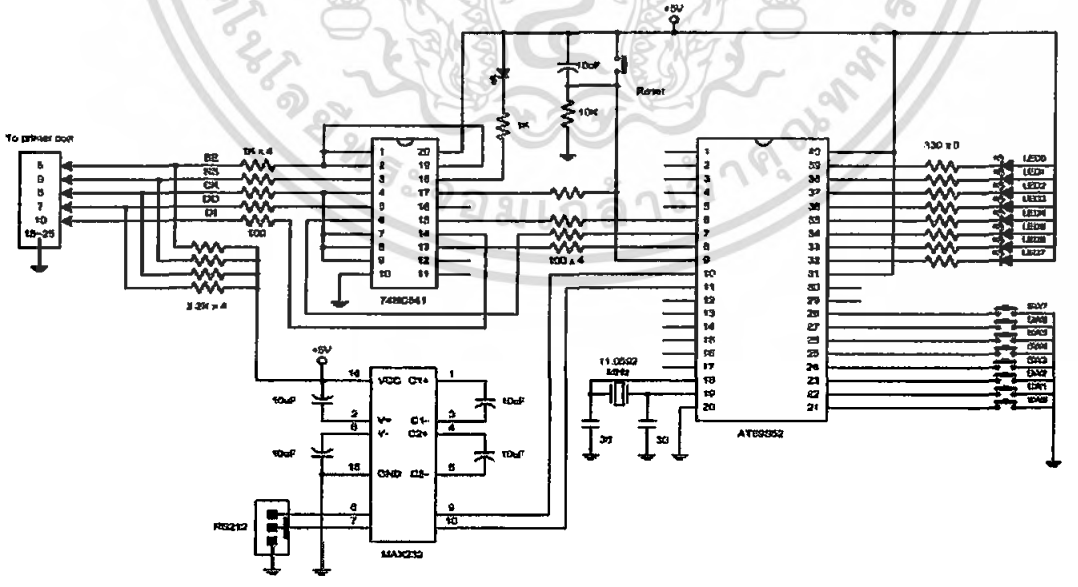
ไมโครคอนโทรลเลอร์ MCS-51 จะมีชุดคำสั่ง (Instruction Set) อยู่จำนวนหนึ่งสำหรับสั่งงานให้ทำงานต่างๆ และเนื่องจาก MCS-51 จะประมวลผลแบบ 8 บิต รหัสภาษาเครื่องจะมีขนาด 8 บิตด้วย ซึ่งชุดคำสั่งจะมีได้จำนวนสูงสุด $2^8 = 256$ ชุดคำสั่ง คำสั่งแต่ละคำสั่งอาจมีขนาด 1, 2 หรือ 3 ไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MCS-51 มีโหมดการอ้างแอดเดรส (Addressing Modes) เพื่อติดต่อกับหน่วยความจำซึ่งอาจจะเป็นการอ่านข้อมูลออกมาหรือเขียนข้อมูลเข้าไปได้ทั้งหมด 7 โหมดคือ Register, Direct, Indirect, Immediate, Relative, Absolute และ Index

2.5.2 MCS-51 จะแบ่งชุดของคำสั่งออกได้ 5 ประเภท ได้แก่

1. Arithmetic Instructions เป็นกลุ่มคำสั่งที่ทำงานด้านคณิตศาสตร์ เช่น ADD, SUBB, INC, DIV เป็นต้น
2. Logical Instructions มีลักษณะการทำงานคล้ายกับ Boolean Operation ซึ่งสามารถกระทำแบบไบต์ต่อไบต์หรือบิตต่อบิตได้ เช่น ANL, ORL
3. Data Transfer Instructions เป็นกลุ่มคำสั่งที่ใช้ในการเคลื่อนย้ายคัดลอกข้อมูล ซึ่งสามารถติดต่อกับหน่วยความจำได้หลายแบบ เช่น MOV, XCH, XCHD
4. Boolean Instructions เช่น ANL, ORL, CLR, SETB
5. Program Branching เป็นกลุ่มคำสั่งสำหรับสั่งให้โปรแกรมกระโดดไปทำงานในตำแหน่งที่ต้องการแบ่งเป็นกลุ่มย่อยได้ 2 กลุ่มคือ กระโดดแบบมีเงื่อนไข เช่น AJMP, LJMP, SJMP กับกระโดดแบบไม่มีเงื่อนไข เช่น JZ, JNZ, CJNE, DJNZ



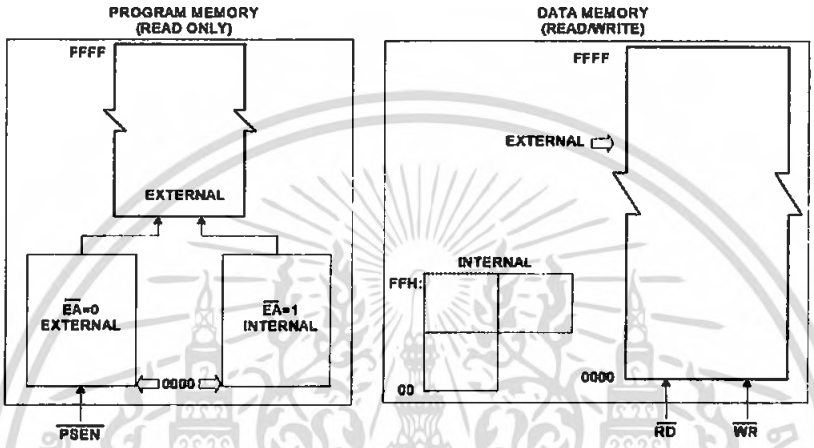
รูปที่ 2.28 แสดงบอร์ดไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 Memory Organization in MCS-51

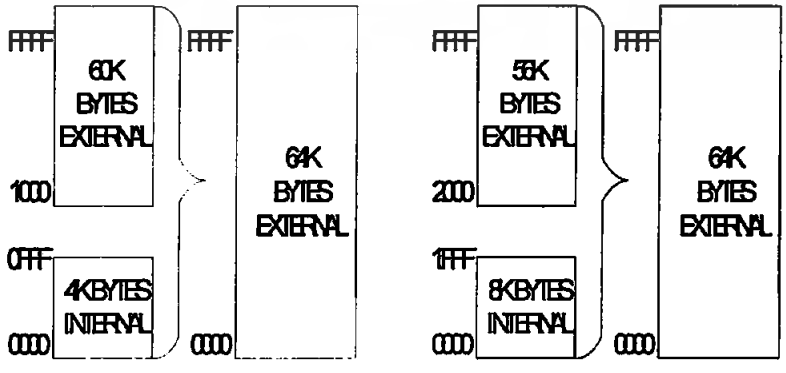
Logical separation of program and data memory

สำหรับ Microcontroller MCS-51นั้นถูกออกแบบมาให้มีหน่วยความจำสำหรับเก็บโปรแกรม (Op-code) และหน่วยความจำสำหรับเก็บข้อมูลที่แยกออกจากกัน ดังรูปที่ 2.29



รูปที่ 2.29 แสดงหน่วยความจำรอมและหน่วยความจำแรม

การออกแบบของ Data memory ที่แยกออกมานี้จะทำให้สามารถเรียกใช้งานได้โดยใช้ Address เพียง 8 Bit เท่านั้น ซึ่งจะทำได้อย่างรวดเร็วใน CPU ที่เป็น 8 Bit แต่การใช้ Address เพียง 8 Bit ทำให้อ้างถึงตำแหน่งของ Internal data memory ได้เพียง 256 ตำแหน่งเท่านั้น (00h – FFh) ซึ่งก็เพียงพอสำหรับการใช้ในการทำงานของโปรแกรมขนาดเล็กแต่ถ้าต้องการใช้งาน Data memory จำนวนมากจะต้องเป็นการใช้งานของ External data memory ซึ่งสามารถใช้การอ้างถึงตำแหน่ง Address แบบ 16 Bit โดยใช้ DPTR: Data Pointer (Data memory address register)



รูปที่ 2.30 แสดงหน่วยความจำแรมภายใน MCS-51

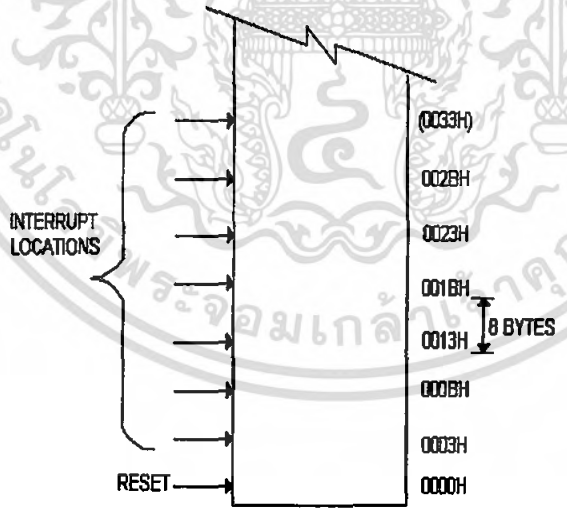
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของ Program memory จะเป็นหน่วยความจำที่อ่านได้เพียงอย่างเดียวและสามารถมีได้ทั้งหมด 64K ตำแหน่งสำหรับ MCS-51 ในแบบที่มี Program memory ภายในก็จะมีพื้นที่ในการเก็บโปรแกรมภายใน IC เอง 4K, 8K, 16K หรือ 32K Address (ตามเบอร์ของ IC ที่ใช้) สำหรับ Address ที่มากกว่านี้ก็จะเป็น Program memory ภายนอกส่วน MCS-51 ที่ไม่มี Internal program memory ส่วนของ Program memory ทั้งหมดจะอยู่ภายนอกสำหรับการอ่าน External program memory นั้น MCS-51 จะใช้ขาสัญญาณ $\overline{PSEN}$ (Program Store Enable)

1. External Data Memory

สำหรับ External data memory ซึ่งสามารถที่จะอ่านหรือเขียนข้อมูลลงไปได้ก็จะมี Address ที่แยกออกจาก Program memory และมีตำแหน่งของ External data memory ได้ทั้งหมด 64K Address ในการติดต่อกับ External data memory นั้น MCS-51 จะใช้ขาสัญญาณ $\overline{RD}$ และ $\overline{WR}$ ในกรณีที่ต้องการใช้ External program memory และ External data memory ร่วมกันนั้นจะทำได้โดยการนำสัญญาณ $\overline{PSEN}$ และ $\overline{RD}$ มารวมกันโดยใช้ AND gate จะได้สัญญาณในการอ่าน External Program/Data memory

2. Program memory



รูปที่ 2.31 แสดงผังตำแหน่ง Program memory

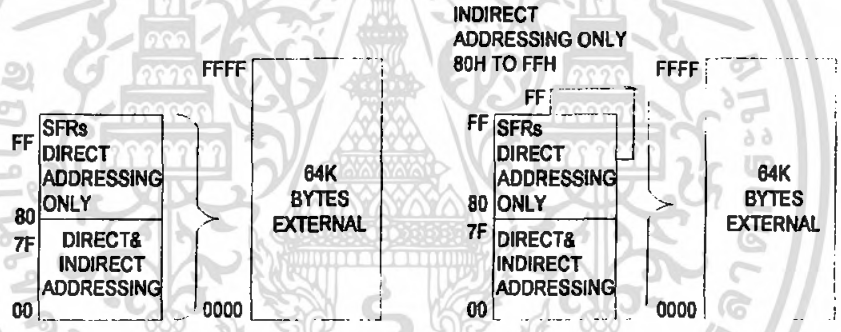
Program memory ในตำแหน่งเริ่มต้นซึ่งเมื่อ CPU เริ่มการทำงานหลังจากการReset ก็จะเริ่มการทำงานตามคำสั่งที่ Address 0000h และสำหรับ Address ที่แสดงต่อมานั้นจะมีตำแหน่งที่อยู่ของ ISR:Interrupt Service Routine โดยการทำงานของ Interruptใน MCS-51 เมื่อเกิดสัญญาณ Interrupt

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เข้ามามันก็จะกระโดดการทำงานมายังโปรแกรมในตำแหน่งที่กำหนดนี้ อย่างเช่นสำหรับInterrupt 0 จากภายนอกเมื่อ MCS-51 ได้รับสัญญาณ Interrupt นี้มันก็จะกระโดดการทำงานมายังโปรแกรมใน Address 0003h และสำหรับInterrupt ที่เกิดจาก Timer 0 ก็จะกระโดดการทำงานมายังโปรแกรมใน Address 000Bh และInterrupt 1 จากภายนอกก็จะกระโดดการทำงานมายังโปรแกรมใน Address 0013h ในแต่ละ Address กำหนดให้สำหรับ ISR นั้นจะมีพื้นในการเก็บโปรแกรมได้ 8 Address ซึ่งถ้า ISR ที่ต้องการเป็นโปรแกรมที่สั้นๆ ก็จะสามารถใส่เข้าไปได้แต่ถ้าเป็น ISR ที่ยาวมากแล้วก็จะทำได้โดยการใช้คำสั่ง Jump ไปยังโปรแกรมที่ต้องการอีกที

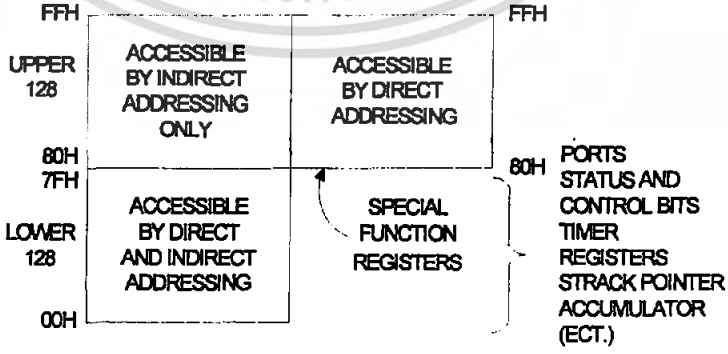
3. Data memory

Data memory ของ MSC-51 นั้นจะแบ่งออกเป็น Internal data memory และ External data memory โดยการใช้งานของ Data memory ทั้งสองส่วนนี้จะมี Address ที่แยกจากกันด้วย MCS-51 with 128 bytes Internal Data memory (ซ้าย) MCS-51 with 256 bytes Internal Data memory (ขวา)



รูปที่ 2.32 แสดงผัง Internal data memoryและ External data memory

4. หน่วยความจำภายใน (Internal data memory)

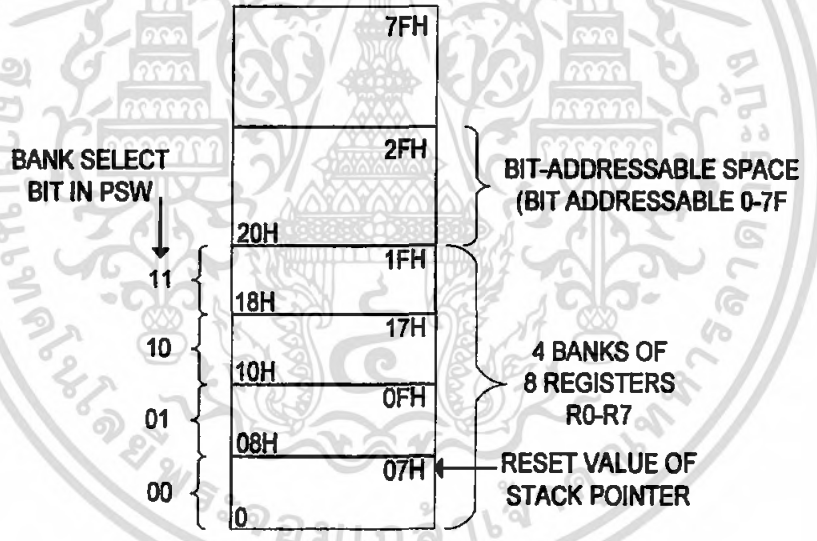


รูปที่ 2.33 แสดงผังหน่วยความจำภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผังการจัดแบ่งพื้นที่ของ Internal data memory จะเป็นดังรูปที่ 2.33 ซึ่งมันจะถูกแบ่งออกเป็น 3 ส่วนด้วยกัน คือ Lower 128, Upper 128 และ SFR และจากการที่ Internal data memory นั้นมีเพียง 256 ตำแหน่งเท่านั้นทำให้การอ้าง Address สามารถทำได้โดยใช้เพียง 8 Bit แต่ด้วยเทคนิคของการอ้างถึงตำแหน่งข้อมูล (Addressing mode) ทำให้ได้ตำแหน่งของ Internal data memory ทั้งหมด 384 bytes โดยสำหรับข้อมูลตั้งแต่ Address 80h – FFh ถ้าอ้างถึงข้อมูลที่ Address นั้นๆ ด้วยวิธีของ Direct addressing ก็จะได้ข้อมูลที่มาจากคนละส่วนกับการอ้างถึงข้อมูลที่ Address เดียวกันนั้นด้วยวิธีของ Indirect addressing จากรูปที่ 2.33 จะเห็นได้ว่า Memory ในส่วนของ Upper 128 ก็จะมี Address เดียวกับ Memory ในส่วนของ SFR แต่จะใช้วิธีการเข้าถึงข้อมูลที่แตกต่างกันนั่นเอง

5.Lower 128 bytes of internal data memory



รูปที่ 2.34 แสดง Lower 128 bytes of internal data memory

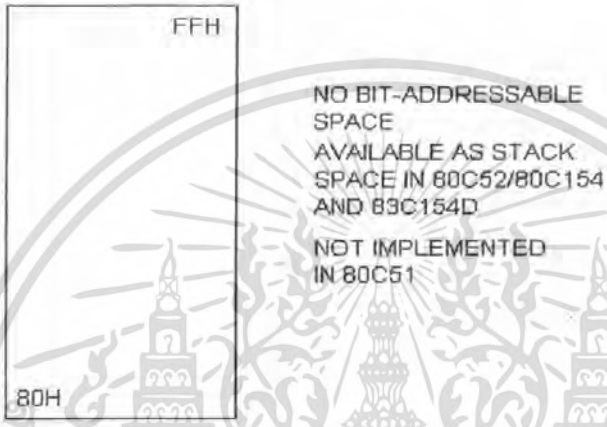
Internal data memory 128 ตำแหน่งแรกจะมีอยู่เหมือนกันทั้งหมดใน MCS-51 แบบต่างๆ โดยจะมีการจัดแบ่งดังรูปที่ 2.34 ในส่วนของ 32 bytes แรกจะใช้งานเป็น Register 4 ชุด ชุดละ 8 register คือ R0 – R7 ซึ่งสามารถเรียกใช้งานได้จากคำสั่งต่างๆ การเลือกชุดของ Register ที่จะใช้งานจะทำโดยใช้สถานะของข้อมูล 2 Bit ที่อยู่ใน PSW: Program Status Word การออกแบบในลักษณะนี้ทำให้ได้ Op-code ที่เรียกใช้ Register นั้นสั้นกว่ามาจาก Register bank อีก 16 Address จะเป็นพื้นที่เก็บข้อมูลที่สามารถเรียกใช้งานในระดับ Bit ได้ 128 bit โดยจะมี Address สำหรับการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกใช้งานระดับ Bit คือ 00h – 7Fh และข้อมูลในทุกๆ ตำแหน่งในส่วนของ Lower 128 นี้สามารถจะอ้างถึงด้วยวิธีการของ Direct /Indirect addressing จากการใช้งานของคำสั่งต่างๆ ได้

6. Upper 128 bytes of internal data memory

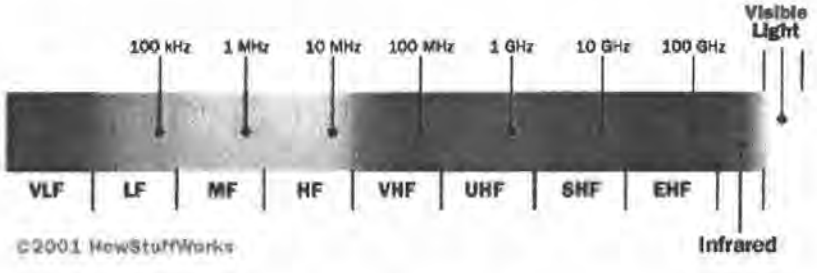
หน่วยความจำในส่วนของ Upper 128 นี้ จะมีอยู่ใน MCS-51 แบบที่มี Internal data memory 256 ตำแหน่งเท่านั้น เช่น 80C52, 83C154, 83C154D



รูปที่ 2.35 แสดง Upper 128 bytes of internal data memory

2.6 Wireless

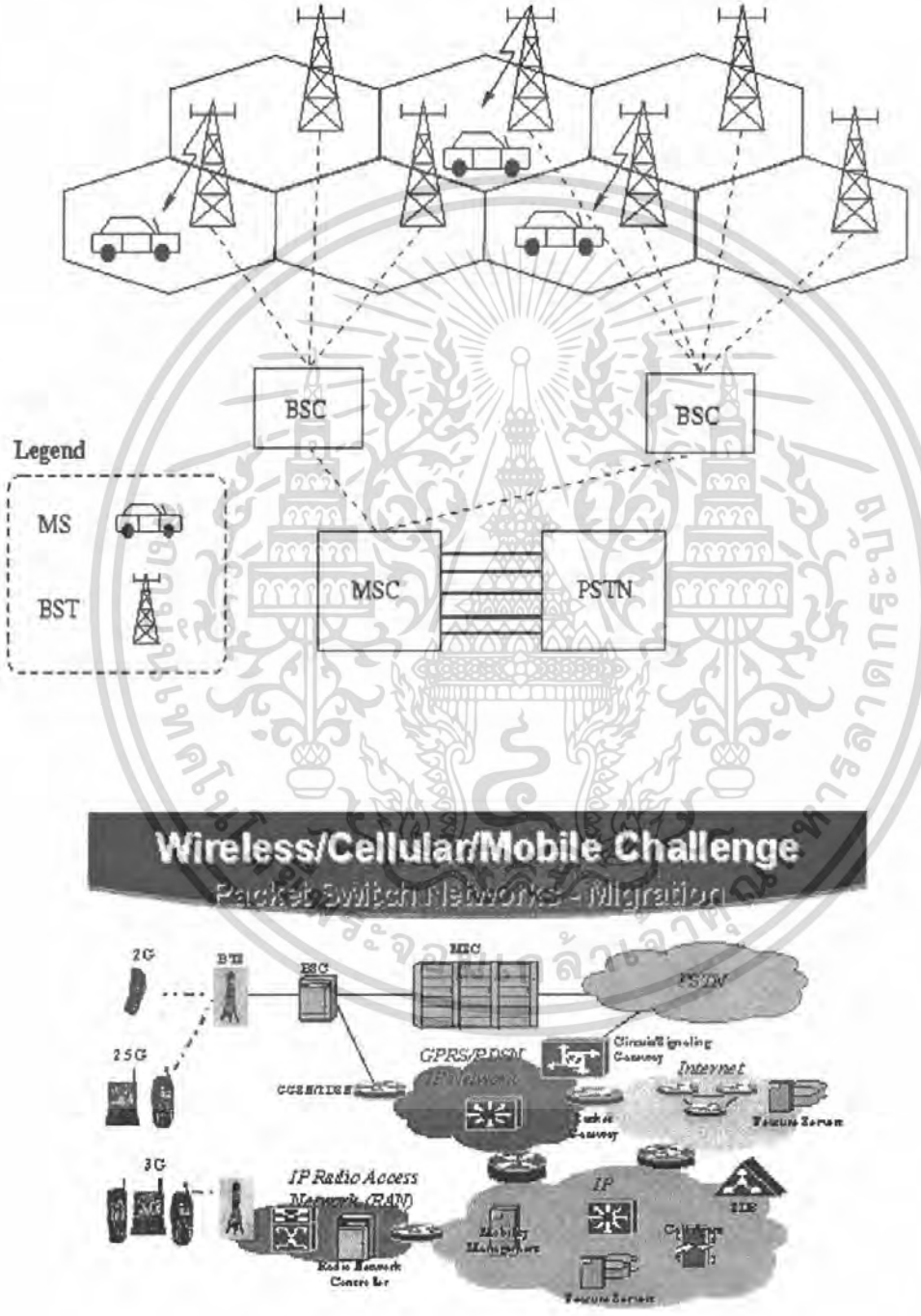
การใช้งานของคลื่นวิทยุที่มีรอบในการส่งต่อวินาทีที่แตกต่างกันทำให้เกิดคำว่า Hertz ซึ่งเป็นชื่อของ Heinrich Rudolph Hertz (1857-1894) ซึ่งนักฟิสิกซ์ชาวเยอรมันเป็นผู้ค้นพบว่าคนเราเสียงอยู่ระหว่าง 20 Hz ถึง 20KHz ซึ่งจำนวนครั้งของคลื่นเรียกว่าความถี่ (Frequency) ซึ่งถ้าเป็นช่วงความถี่ต่างๆจะเรียกว่า Electromagnetic spectrum, ซึ่งจะขึ้นอยู่กับช่วงความถี่ดังรูปที่ 2.36 แสดงถึงช่วงความถี่ที่ใช้ตั้งแต่ระดับวิทยุถึงระดับแสง



รูปที่ 2.36 แสดงElectromagnetic spectrum

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วงของความถี่วิทยุหรือสูงกว่าเรียกว่าความถี่แสงโดยเป็น Ultraviolet, X-ray, และ Gamma ray หรือสูงกว่าข้อมูล Wireless ใช้เป็นส่วนของช่วงคลื่นความถี่ Radio ซึ่งอยู่ระหว่าง 800 KHz ถึง 5 GHz แม้ว่าข้อมูล Wireless จะถูกระบุความถี่ไว้ในแต่ละประเทศจะมีการควบคุมการใช้ความถี่



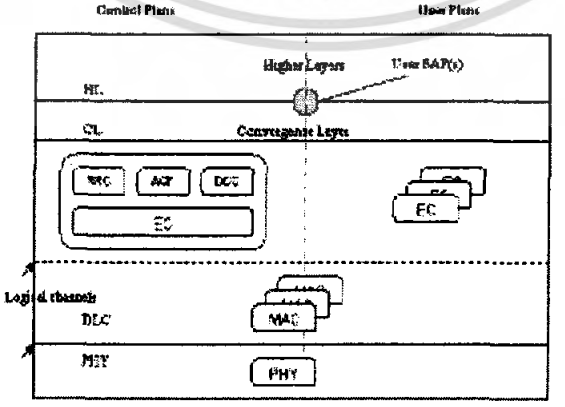
รูปที่ 2.37 แสดงข้อมูล Wireless ของช่วงคลื่นความถี่ Radio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ผ่านมาไม่กี่ปีมีการพัฒนาเทคโนโลยี่ Wireless อย่างมากซึ่งจะมีการตรวจสอบเซกชั่นและมีการพัฒนาอย่างเด่นชัด

2.6.1 คลื่นความถี่วิทยุตามบ้าน HomeRF

มาตรฐานเครือข่ายไร้สายอีกอย่างหนึ่งเรียกว่า HomeRF ซึ่งออกแบบกำหนดเป็นอินเตอร์เน็ต Broadband ที่ตามบ้านยังไม่มีการเข้าใช้ในเครือข่ายขององค์กรซึ่งรูปแบบ คือ IEEE802.11 เครือข่าย HomeRF ถูกออกแบบมารองรับการใช้งานแอพพลิเคชั่นอัตโนมัติตามบ้านและราคาดึงดูดผู้ใช้ตามบ้าน HomeRF ผลิตให้ทำงานที่ 2.4 GHz ISM band ใช้ FHSS ที่ 50-100 Hops per second ซึ่งเปลี่ยนช่อง 50-100 ครั้งต่อวินาที (หรือมากกว่า) ยุคที่ 1 ของ HomeRF (1.0) ผลิตมาให้อัตราข้อมูลสูงสุดที่ 1.6 Mbps และระยะทาง 150 ฟุต ยุคที่ 2 ของ HomeRF (2.0) ทำงานได้ถึง 10 Mbps ซึ่งใช้กำลังไฟต่ำคือน้อยกว่า 10 mW และต่อเชื่อม TCP/IP อย่างเต็มรูปแบบและมีขนาดเล็กฟีเจอร์อื่นของ HomeRF รองรับคุณภาพเสียงใกล้ Wire-line โดยมาตรฐานบริษัท Digital Enhanced Cordless Telephony (DECT) อุปกรณ์ HomeRF ในยุคที่ 3 วางแผนรองรับอัตราส่งข้อมูล 20Mbps โดยคาดว่าจะเริ่มในปี 2002 นี้ ภาพรวมเทคโนโลยี่ HomeRF เครือข่าย HomeRF เรียกว่า Broadband Internet Home ประกอบด้วยชนิดของอุปกรณ์หลากหลาย เช่น Laptop (A-node), Internet Appliance (SA-node), Audio Headset (S-node), Cordless Handset (I-node) และอุปกรณ์ควบคุม Control Point (CP Class 1) รายละเอียดของ HomeRF จะเหมือนกับมาตรฐานเครือข่ายอื่นๆที่อยู่บนชั้นที่ต่ำที่สุดคือลำดับชั้นที่ 2 ซึ่งทำงานที่ Data Link Control (DLC) และใช้ Media Access Control (MAC) Layer กำหนดชนิดของบริการข้อมูล เช่นเสียงหรือการส่งลำดับสตรีมที่ถูกกำหนดลำดับความสำคัญ ซึ่งจะมีคุณสมบัติในเรื่องความปลอดภัย การโรมมิ่งและการแม็บกับมาตรฐานในชั้นที่สูงขึ้น ดังรูปที่ 2.38



รูปที่ 2.38 แสดงการโรมมิ่งและการแม็บกับมาตรฐานในชั้นที่สูงขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ET-RF24G V1.0

ET-RF24G V1.0 เป็นชุด Signal Converter สำหรับใช้แปลงสัญญาณระหว่าง RS232 และ RF-Wirelessโดยในโหมดการทำงานของการส่งข้อมูล (Transmitter) จะทำหน้าที่รอรับข้อมูลจากพอร์ตสื่อสารอนุกรม RS232 จากขาRX แล้วแปลงเป็นสัญญาณความถี่(GFSK) ส่งออกไปในอากาศ และในทางกลับกันในโหมดการทำงานแบบรับ (Receiver) ชุด ET-RF24G V1.0 ก็จะทำหน้าที่คอยตรวจจับข้อมูลที่อยู่ในรูปของสัญญาณความถี่ (GFSK) จากด้าน RF เพื่อแปลงกลับเป็นข้อมูลแบบ RS232 ส่งออกไปทางขา TX ได้ด้วย

ซึ่งจะเห็นได้ว่าชุดแปลงสัญญาณ ET-RF24G V1.0 นั้นสามารถนำไปต่อใช้งานร่วมกับพอร์ตสื่อสารอนุกรมแบบ RS232 เพื่อใช้งานในลักษณะของการสื่อสารอนุกรมแบบไร้สาย (Wireless Transceiver) ได้โดยตรงโดยจะมีข้อดีกว่าคือสามารถรับส่งข้อมูลกันได้ในระยะทางที่ไกลกว่าRS232หลายเท่าตัวและประการสำคัญคือไม่จำเป็นต้องใช้สายสัญญาณที่เป็นตัวนำสัญญาณทางไฟฟ้าในการสื่อสารข้อมูลกันทำให้สามารถเปลี่ยนแปลงหรือเคลื่อนย้ายจุดรับส่งข้อมูลได้ตลอดเวลา ซึ่งถ้าเป็นการรับส่งข้อมูลด้วยระบบ RS232 แบบที่ใช้ายสัญญาณนั้นจะเกิดความยุ่งยากในการติดตั้งสายสัญญาณเป็นอย่างมาก

อย่างไรก็ตามการรับส่งข้อมูลโดยใช้อากาศเป็นตัวกลางในการสื่อสารนั้นก็มีข้อจำกัดบางประการเหมือนกันโดยเฉพาะอย่างยิ่งเรื่องความน่าเชื่อถือของข้อมูลที่รับส่งกันซึ่งมีโอกาสผิดพลาดหรือสูญหายได้เหมือนกัน เนื่องจากในการลำเลียงข้อมูลนั้นไม่ได้ใช้สายสัญญาณเป็นตัวกลางในการรับส่งข้อมูลแต่ใช้อากาศเป็นตัวกลางในการรับส่งข้อมูลแทน ซึ่งมีโอกาสที่ข้อมูลจะเกิดการรบกวนจากสัญญาณอื่นๆ ที่มีย่านความถี่ใกล้เคียงกันแล้วทำให้ข้อมูลผิดเพี้ยนไปได้บ้างเหมือนกัน ซึ่งระบบการจัดการข้อมูลของเครื่อง ET-RF24G V1.0 นั้นมีระบบการเข้ารหัสและถอดรหัสข้อมูลที่มีความน่าเชื่อถืออยู่ในเกณฑ์ที่จัดว่าดีโดยข้อมูลแต่ละ Byte ที่มีการรับส่งกันนั้นจะมีการตรวจสอบความถูกต้องของข้อมูลให้ด้วยแล้วโดยข้อมูลที่รับได้จากด้าน RF นั้นรับประกันได้ว่าเป็นข้อมูลที่มีความถูกต้องแน่นอน แต่อย่างไรก็ตามการรับส่งข้อมูลนั้นมีโอกาสผิดพลาดในเรื่องของการสูญหายของข้อมูลบ้างเหมือนกัน เนื่องจากกลไกลในการรับส่งข้อมูลของเครื่อง ET-RF24G V1.0 นั้น จะมีการตรวจสอบข้อมูลทุก Byte ที่รับได้จาก RF เสมอ ซึ่งพบว่ามีความผิดพลาดเกิดขึ้นจะทิ้งข้อมูล Byte นั้นไป ซึ่งผู้ใช้ควรมีกลไกลในการตรวจสอบข้อมูลที่รับส่งกันว่าครบถ้วนหรือไม่ด้วย ซึ่งหากพบว่ามีการสูญหายของข้อมูลเกิดขึ้นก็ให้ร้องขอให้มีการส่งข้อมูลนั้นซ้ำนั้นๆใหม่อีกครั้งหนึ่งก็จะสามารถแก้ไขปัญหาดังกล่าวได้

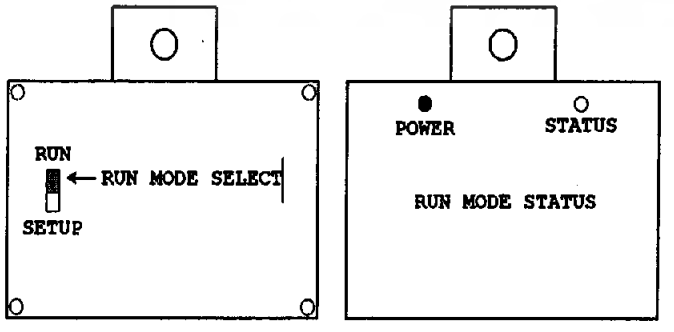
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดการทำงาน

สำหรับโหมดการทำงานของ ET-RF24G V1.0 นั้นจะแบ่งออกเป็น 2 โหมดด้วยกันโดยการกำหนดโหมดการทำงานของ ET-RF24G V1.0 นั้นจะกระทำผ่าน Switch เลือกโหมดซึ่งอยู่ด้านใต้กล่องโดยการเลือกโหมดการทำงานนั้นจะต้องกระทำให้เสร็จเรียบร้อยก่อนการจ่ายไฟให้กับ ET-RF24G V1.0 ด้วยเสมอเนื่องจากการทำงานของเครื่อง ET-RF24G V1.0 นั้นจะทำการตรวจสอบโหมดการทำงานของเครื่องจาก Switch เลือกโหมดเฉพาะในช่วงของการจ่ายไฟเลี้ยงให้เครื่องเริ่มต้นทำงานครั้งแรก (Power-ON) เท่านั้น ซึ่งการเปลี่ยนแปลงตำแหน่งการทำงานของ Switch เลือกโหมดหลังจากทำการจ่ายไฟให้กับ ET-RF24G V1.0 ไปแล้ว จะไม่มีผลต่อการทำงานของเครื่องแต่อย่างใดโดยการทำงานของเครื่อง ET-RF24G V1.0 นั้นจะมี LED แสดงสถานะการทำงานของเครื่องจำนวน 2 หลอด คือ LED POWER ซึ่งเป็น LED สีแดงโดยที่ LED POWER นี้จะติดสว่างให้เห็นตลอดเวลาที่มีการจ่ายไฟเลี้ยงให้เครื่องทำงานอยู่ ส่วน LED อีกดวงหนึ่งนั้นจะเป็น LED สีเขียวใช้แสดงสถานะการทำงานของเครื่องซึ่งเรียกว่า LED STATUS โดย LED STATUS นี้จะเกิดการกระพริบตามจังหวะของการรับส่งข้อมูลกันในแต่ละครั้งโดยในสภาวะปรกตินั้น ถ้าเครื่องทำงานอยู่ใน RUN MODE หลอด LED STATUS จะดับอยู่ตลอดเวลาถ้าไม่มีการรับส่งข้อมูล แต่ถ้าตัวเครื่องทำงานอยู่ใน SETUP MODE หลอด LED STATUS จะติดอยู่ตลอดเวลาถ้าไม่มีการรับส่งข้อมูลโดยโหมดการทำงานของ ET-RF24G V1.0 จะมีอยู่ด้วยกัน 2 โหมด คือ

2.7.1 การใช้งานเครื่อง ET-RF24G V1.0 ใน Run Mode

การใช้งานใน Run Mode ซึ่งเป็นโหมดของการใช้งานตามปกติของเครื่องโดยเมื่อเครื่อง ET-RF24G V1.0 เข้าทำงานในโหมดนี้แล้ว จะสังเกตเห็นหลอดไฟแสดงสถานะการทำงานหรือ LED STATUS ดับอยู่แต่เมื่อมีการรับหรือส่งข้อมูลเกิดขึ้นสถานะการทำงานของ LED STATUS จึงจะกระพริบตามจังหวะของการรับส่งข้อมูลนั้นๆ แต่ถ้ายังไม่มีการรับส่งข้อมูลกัน LED STATUS จะดับอยู่ตลอดเวลา

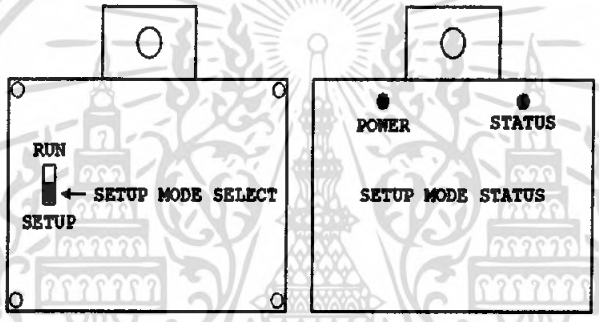


รูปที่ 2.39 แสดงการเลือกโหมดการทำงานสำหรับใช้งานปกติ (Run Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 การใช้งานเครื่อง ET-RF24G V1.0 ใน Setup Mode

การใช้งานเครื่อง ET-RF24G V1.0 ใน Setup Mode ซึ่งเป็นโหมดสำหรับใช้กำหนดค่า Configurationต่างๆ สำหรับควบคุมการทำงานของเครื่อง ET-RF24G V1.0 ที่จะใช้ในขณะที่เครื่องทำงานอยู่ใน Run Mode โดยในการ Setup ค่า Configuration ต่างๆนั้นจะกระทำร่วมกับโปรแกรม "ET_RF24G_V1.EXE" ของอีทีทีซึ่งเมื่อเครื่อง ET-RF24G V1.0 เข้าทำงานในโหมด Setup แล้ว จะสังเกตเห็นหลอดไฟแสดงสถานะการทำงานหรือ LED STATUS ติดสว่างค้างอยู่ตลอดเวลา แต่เมื่อมีการสั่งอ่านหรือเขียนข้อมูลกับบอร์ดสถานะการทำงานของ LED STATUS จึงจะกระพริบตามจังหวะของการรับส่งข้อมูล แต่ถ้ายังไม่มีการรับส่งข้อมูลกัน LED STATUS จะติดค้างอยู่ตลอดเวลา

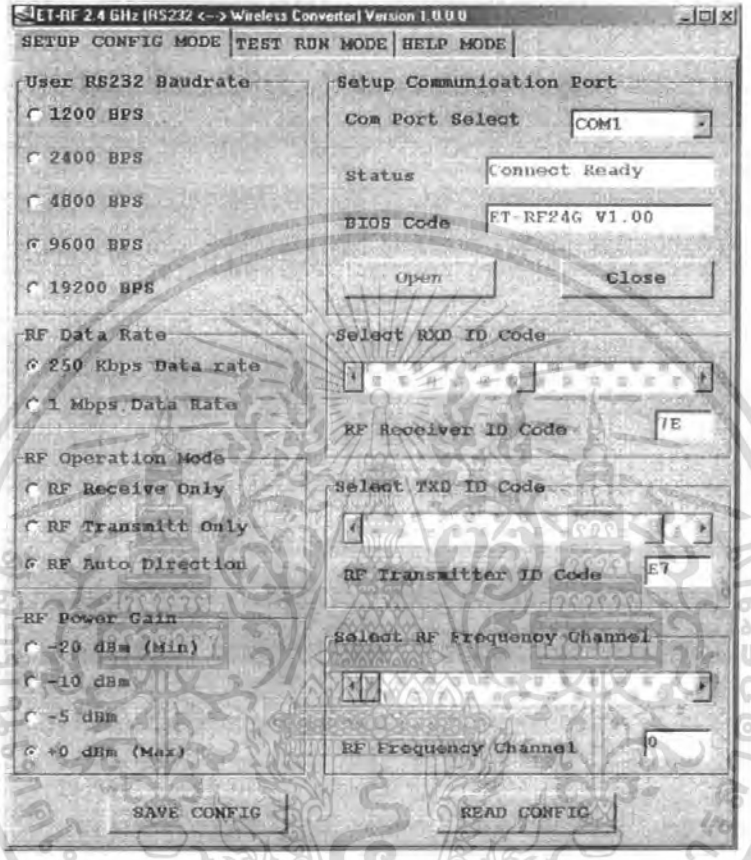


รูปที่ 2.40 แสดงการเลือกโหมดการทำงานสำหรับกำหนดค่า Configuration (Setup Mode)

ซึ่งการกำหนดค่าConfigurationให้กับET-RF24G V1.0นั้นจะต้องกระทำในขณะที่ตัวเครื่องทำงานอยู่ใน Setup Mode เท่านั้น(เลือก Switch กำหนดโหมดไว้ทางด้าน Setup แล้วจ่ายไฟให้เครื่องเริ่มต้นทำงาน) โดยค่าของConfiguration ต่างๆ นั้นจะถูกใช้สำหรับเป็นเงื่อนไขในการทำงานของ ET-RF24G V1.0 ในขณะที่อยู่ใน Run Mode ดังนั้นก่อนการเริ่มต้นใช้งานเครื่องในครั้งแรกนั้นจึงจำเป็นอย่างยิ่งที่จะต้องทำการกำหนดค่าของ Configuration ต่างๆ ให้ถูกต้องและตรงกับความต้องการที่จะใช้งานเสียก่อน โดยเมื่อทำการกำหนดค่าตัวเลือกต่างๆของ Configuration เรียบร้อยแล้วก็สามารถเปลี่ยนโหมดการทำงานของตัวเครื่องกลับเป็น Run Mode พร้อมกับการปิดไฟที่จ่ายให้กับตัวเครื่อง (Power-OFF) ชั่วขณะหนึ่งจากนั้นจึงเริ่มต้นจ่ายไฟให้กับตัวเครื่องใหม่ (Power-ON) ก็สามารถใช้งาน ET-RF24G V1.0 ตามค่าของ Configuration ที่กำหนดไว้แล้วได้ทันทีโดยค่าตัวเลือกต่างๆของConfiguration ที่ได้กำหนดไว้แล้วจะถูกเก็บไว้ภายในตัวเครื่องอย่างถาวร ถึงแม้ว่าจะไม่ได้ทำการจ่ายไฟให้กับตัวเครื่องแล้วก็ตาม ดังนั้นเมื่อทำการกำหนดค่า Configuration ต่างๆ เรียบร้อยแล้ว ถ้าไม่มีการเปลี่ยนแปลงเงื่อนไขการทำงานของตัวเครื่องต่างไปจากเงื่อนไขเดิมที่ได้กำหนดไว้แล้วก็ไม่จำเป็นต้องทำการกำหนดค่าConfigurationใหม่อีกแต่อย่างใดโดยทุกๆครั้งที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นจ่ายไฟเข้าเครื่องในครั้งแรกนั้นการทำงานของ ET-RF24G V1.0 จะเป็นไปตามเงื่อนไขที่กำหนดไว้ใน Configuration เสมอทุกๆครั้งโดยคุณสมบัติของ Configuration ต่างๆ นั้นมีดังนี้



รูปที่ 2.41 แสดงโปรแกรมที่ใช้สำหรับกำหนดค่า Configuration ของ ET-RF24G V1.0

1. User RS232 Baudrate ใช้สำหรับกำหนดค่าความเร็วในการรับส่งข้อมูลทางด้าน RS232 ของตัวเครื่องในขณะที่ทำงานอยู่ใน Run Mode ซึ่งสามารถกำหนดได้ 5 ค่าคือ
   1.1 1200 BPS
   1.2 2400 BPS
   1.3 4800 BPS
   1.4 9600 BPS
   1.5 19200 BPS
2. RF Data Rate ใช้สำหรับกำหนดความเร็วในการรับส่งข้อมูลทางด้าน RF ของ ET-RF24G V1.0 ซึ่งจะต้องกำหนดให้เครื่อง ET-RF24G V1.0 ทุกๆตัวที่จะนำมาใช้ติดต่อสื่อสารกันมีค่าอัตรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเร็วในการรับส่งข้อมูลด้าน RF หรือ RF Data Rate นี้มีค่าเท่ากันทั้งหมด ซึ่งถ้ากำหนดค่าความเร็วต่างกันจะไม่สามารถรับส่งข้อมูลกันได้ซึ่งค่าอัตราความเร็วในการส่งข้อมูลนี้จะมีผลต่อระยะทางการรับส่งข้อมูลด้วยซึ่งถ้าใช้ความเร็วในการส่งสูง (1Mbps) จะทำให้รัศมีการรับส่งข้อมูลได้ระยะทางสั้นลงแต่ถ้าใช้ความเร็วในการรับส่งข้อมูลที่ช้าลง (250Kbps) จะทำให้ได้รัศมีการรับส่งไกลขึ้นโดยค่า RFDataRate สามารถกำหนดได้ 2 ค่า คือ

2.1 250 Kbps

2.2 1 Mbps

3. RF Operation Mode ใช้สำหรับกำหนดโหมดการทำงานของ ET-RF24G V1.0 ซึ่งสามารถกำหนดหน้าที่การทำงานได้ 3 แบบ ด้วยกันคือ

3.1 RF Receive Only เป็นการกำหนดให้ ET-RF24G V1.0 ทำ หน้าที่เป็นฝ่ายรอรับข้อมูลทางด้านRF เพื่อเปลี่ยนเป็นข้อมูลแบบ RS232 และส่งออกไปทางด้านขา TX ของ RS232 ตลอดเวลา

3.2 RF Transmit Only เป็นการกำหนดให้ ET-RF24G V1.0 ทำหน้าที่เป็นฝ่ายรอรับข้อมูลทางด้าน RS232 จากขา RX เพื่อเปลี่ยนเป็นข้อมูลแบบ GFSK และส่งออกไปทางด้าน RF ตลอดเวลา

3.3 RF Auto Direction เป็นการกำหนดโหมดการทำงานแบบ Half Duplex 2 ทิศทาง ซึ่งสามารถสลับโหมดการทำงานระหว่างการรับและส่งข้อมูลได้เองโดยอัตโนมัติ โดยในโหมดการทำ งานนี้เครื่อง ET-RF24G V1.0 จะรอตรวจสอบข้อมูลทั้งจากด้าน RS232 และด้าน RF อยู่ตลอดเวลาโดยถ้าได้รับข้อมูลจากด้าน RS232 ก็จะทำการแปลงแล้วส่งออกทางด้าน RF จากนั้นก็จะกำหนดให้ด้าน RF กลับมาเป็นฝ่ายรอรับข้อมูลตามเดิม และเมื่อได้รับข้อมูลจากด้าน RF ก็จะแปลงเป็นข้อมูลแล้วส่งออกไปทางด้าน RS232 โดยอัตโนมัติ

4. RF Power Gain เป็นการกำหนดกำ ลังส่งของวงจร RF Power ที่ใช้ในการส่งข้อมูลโดยค่า+0dBm เป็นค่ากำลังส่งสูงสุดส่วน –20 dBm เป็นค่ากำลังส่งต่ำสุดโดยสามารถกำหนดได้ 4 ระดับคือ

4.1 -20dBm (กำลังส่งต่ำสุด)

4.2 -10dBm

4.3 -5dBm

4.4 +0dBm (กำลังส่งสูงสุด)

5. RXD ID Codeเป็นรหัส ID Code ของเครื่อง ET-RF24G V1.0 ในโหมดของการรับข้อมูลจาก RF โดยเมื่อเครื่อง ET-RF24G V1.0 ด้านส่งจะทำการส่งข้อมูลออกไปทาง RF นั้นจะมีการระบุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

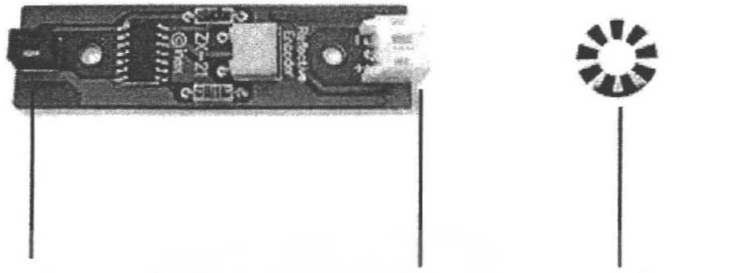
หมายเลข ID Code ของด้านรับรวมไปกับชุดข้อมูลด้วยเสมอโดยเมื่อเครื่อง ET-RF24G V1.0 ที่อยู่ทางด้านรับทำการรับข้อมูลจากด้าน RF ได้ อันดับแรกมันจะทำการเปรียบเทียบรหัส ID Code ที่รวมมากับข้อมูลที่รับมาได้ว่าตรงกับรหัสของ RXD ID Code ที่กำหนดไว้ในตัวมันหรือไม่ ซึ่งถ้าถูกต้องก็จะแยกเอาเฉพาะส่วนของข้อมูลที่รับเข้ามาได้เพื่อเปลี่ยนเป็นข้อมูลแบบ RS232 แล้วส่งออกไปทางด้าน TX ของ RS232 แต่ถ้ารหัส ID Code ที่รับมาได้ไม่ตรงกับรหัส RXD ID Code ที่กำหนดไว้เครื่อง ET-RF24G V1.0 จะทิ้งข้อมูลชุดนั้นไปทันทีโดยค่า RXD ID Code นั้นสามารถกำหนดได้ 256 ค่าในรูปแบบของเลขฐานสิบหก (00H-FFH)

6. TXD ID Code เป็นรหัส ID Code ปลายทางที่จะส่งข้อมูลไปหาโดยที่เครื่อง ET-RF24G V1.0 ที่ถูกกำหนดให้ทำหน้าที่เป็นฝ่ายส่งข้อมูลนั้นเมื่อมันสามารถรับข้อมูลจาก RS232 ได้แล้วมันจะทำการนำเอาข้อมูลนั้นไปเข้ารหัสรวมกับ TXD ID Code ที่กำหนดไว้ แล้วส่งออกไปทางด้าน RF โดยรหัสของ TXD ID Code นี้หมายถึงรหัส RXD ID Code ของฝ่ายรับที่ต้องการส่งข้อมูลไปหานั่นเองโดยค่า TXD ID Code นั้นสามารถกำหนดได้ 256 ค่าในรูปแบบของเลขฐานสิบหก (00H-FFH)
7. RF Frequency Channel เป็นการกำหนดค่าของช่องความถี่ที่จะใช้ในการรับส่งข้อมูลกันโดยสามารถเลือกกำหนดช่องความถี่ได้สูงสุดมากถึง 125 ช่อง (0-124) โดยการที่เครื่อง ET-RF24G V1.0จะทำการรับส่งข้อมูลกันได้นั้นจะต้องกำหนดช่องความถี่ที่ตรงกันและใช้อัตราความเร็ว RF Data Rate ที่เท่ากันด้วย ซึ่งที่สามารถเลือกกำหนดช่องความถี่ RF Frequency Channel ได้นั้นจะมีประโยชน์เป็นอย่างมากในกรณีที่มีการใช้งานเครื่อง ET-RF24G V1.0 จำนวนหลายๆ กลุ่มในบริเวณพื้นที่ใกล้เคียงกันโดยให้กำหนดช่องความถี่ของ ET-RF24G V1.0 กลุ่มที่จะสื่อสารข้อมูลร่วมกันไว้ที่ช่องความถี่เดียวกันส่วนกลุ่ม อื่นๆก็ให้เลือกกำหนดช่องความถี่ที่แตกต่างกันออกไปเพื่อลดปัญหาการรบกวนกัน

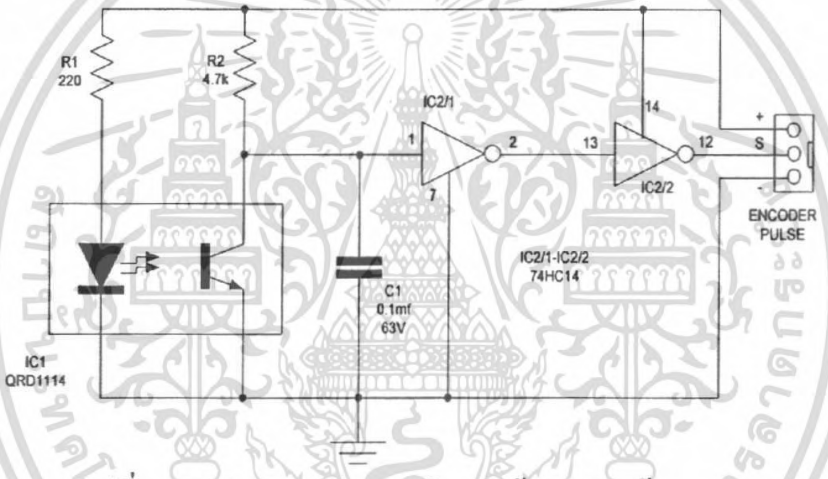
2.8 ตัวตรวจจับรหัสล้อแบบสะท้อนแสง

Zx-21 แผงวงจรตรวจจับรหัสล้อแบบสะท้อนแสงใช้ร่วมกับแผ่นรหัสล้อ 18 แถบความละเอียด 20 องศาต่อแถบเอาต์พุตเป็นสัญญาณพัลส์ต่อกับไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.42 แสดงตัวตรวจจับรหัสล้อแบบสะท้อนแสง



รูปที่ 2.43 แสดงวงจรตัวตรวจจับรหัสล้อแบบสะท้อนแสง

หัวใจของแผงวงจรนี้คือ ตัวตรวจจับแสงสะท้อนอินฟราเรดและแผ่นรหัสล้อโดยแสงอินฟราเรดที่ส่งออกจากตัวตรวจจับจะสะท้อนกับแถบรหัสหากพบแถบสีดำแสงสะท้อนที่โฟโต้ทรานซิสเตอร์รับได้จะน้อยทำให้เกิดลอจิก "1" ส่งไปยังวงจรจัดสัญญาณดิจิตอลซึ่งประกอบด้วยน๊อตเกต 2 ตัวต่อร่วมกันแต่ถ้าพบสีขาวหรือสีน้ำเงินจะเกิดการสะท้อนแสงมากโฟโต้ทรานซิสเตอร์ทำงานได้ดีเกิดลอจิก "0" เมื่อล้อหมุนไปจะเกิดการสะท้อนแสงที่สลับกันเช่นนี้ จึงได้สัญญาณพัลส์ส่งออกเอาต์พุตถ้าล้อหมุน 1 รอบจะได้พัลส์ 9 ลูก 18 ขอบสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
โครงสร้างของระบบโครงงาน

หลักการทำงานและการออกแบบของหุ่นยนต์วัดระยะทางโดยใช้คอมพิวเตอร์ แบ่งออกเป็น 2 ส่วน คือส่วนของฮาร์ดแวร์และส่วนของซอฟแวร์

ส่วนของฮาร์ดแวร์ ประกอบด้วย

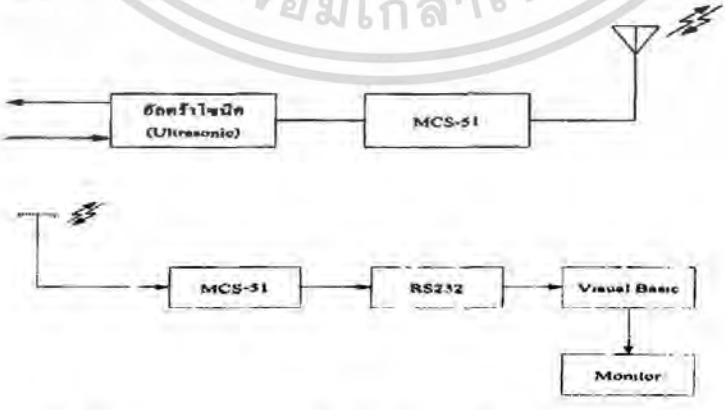
1. ภาครับ-ส่งสัญญาณของส่วนควบคุมหุ่นยนต์
2. ภาครับ-ส่งสัญญาณจากอัลตร้าโซนิก
3. ภาครับ-ส่งสัญญาณภาพของกล้อง

ส่วนของซอฟแวร์ ประกอบด้วย

1. โปรแกรมสั่งการทำงานของหุ่นยนต์ ด้วยคอมพิวเตอร์โดยใช้โปรแกรมวิชวลเบสิก
2. โปรแกรมการที่ใช้ในการติดต่อระหว่างอัลตร้าโซนิกกับไมโครคอนโทรลเลอร์
3. โปรแกรมควบคุมการทำงานของหุ่นยนต์และกล้องโดยใช้ไมโครคอนโทรลเลอร์ 8051

3.1 โครงสร้างโดยรวม

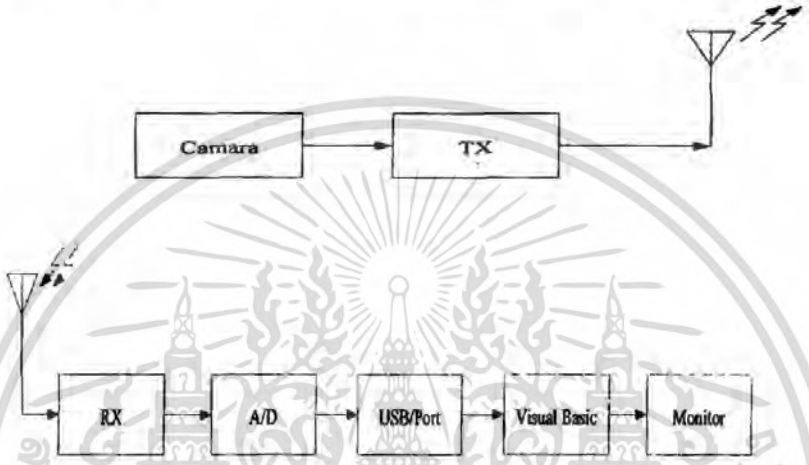
หลักการทำงานโดยรวมของส่วนฮาร์ดแวร์ คือ คอมพิวเตอร์ทำหน้าที่ควบคุมการทำงานโดยใช้โปรแกรมวิชวลเบสิกเขียนโปรแกรมสั่งงาน เพื่อที่จะควบคุมตัวหุ่นยนต์และกล้องให้ปฏิบัติตามคำสั่ง เช่น เดินหน้า หมุนกล้องซ้าย-ขวา ซึ่งส่วนนี้จะใช้ไมโครคอนโทรลเลอร์ 8051 มาทำเป็นส่วนของการส่งสัญญาณควบคุมและใช้โมดูล ET-RF24G V 1.0 ในการรับส่งสัญญาณ



รูปที่ 3.1 แสดงบล็อกไดอะแกรมของการรับส่งสัญญาณโดยใช้โปรแกรมวิชวลเบสิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

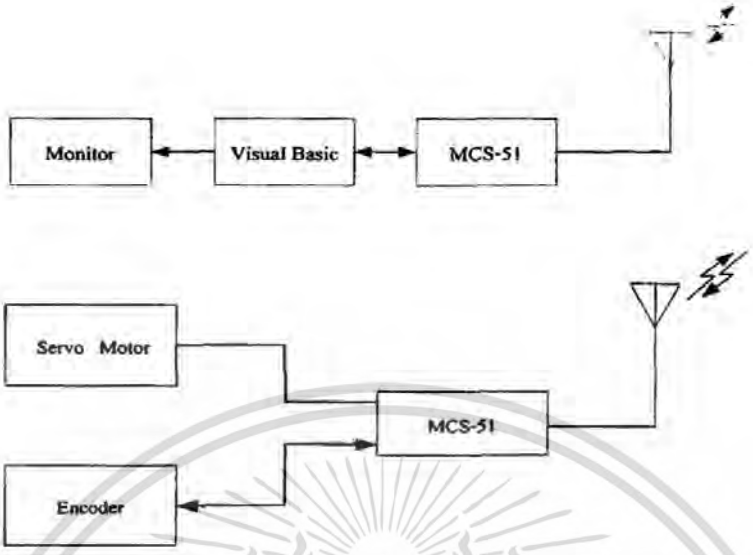
ในส่วนนี้สามารถควบคุมการการรับส่งสัญญาณโดยใช้โปรแกรมวิชวล เบสิกเมื่อทำการส่งงานให้อัลตร้าโซนิกทำการวัดระยะทาง ข้อมูลที่จะส่งผ่านพอร์ตอนุกรมไปยังบอร์ดไมโครคอนโทรลเลอร์ แล้วส่งออกไปด้วยความถี่ 40 kHz เมื่อภาครับได้สัญญาณจะสั่งให้ SRF05 ทำการส่งสัญญาณออกไปที่ขา Echo เมื่อสัญญาณกระทบวัตถุจะสะท้อนกับมาเข้าขา Trigger ของ SRF05 แล้วนำไปคำนวณหาค่าออกมาเป็นระยะทาง แล้วส่งไปยังภาครับเพื่อเก็บไว้คำนวณต่อไป



รูปที่ 3.2 แสดงบล็อกไดอะแกรมของการรับส่งสัญญาณภาพ

หลักการทำงานของบล็อกนี้คือ เมื่อกล้องทำงานจะทำการส่งสัญญาณภาพกับมาที่ภาครับด้วยความถี่ 900-1200 kHz เมื่อภาครับสัญญาณจะนำสัญญาณที่ได้ไปเข้ารหัสเพื่อแปลงสัญญาณอนาล็อกเป็นดิจิตอลแล้วส่งเข้าไปยัง USB พอร์ต โดยโปรแกรม Visual Basic 6.0 จะนำสัญญาณที่เข้ามาทางพอร์ต USB นำออกมาแสดงทางหน้าจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



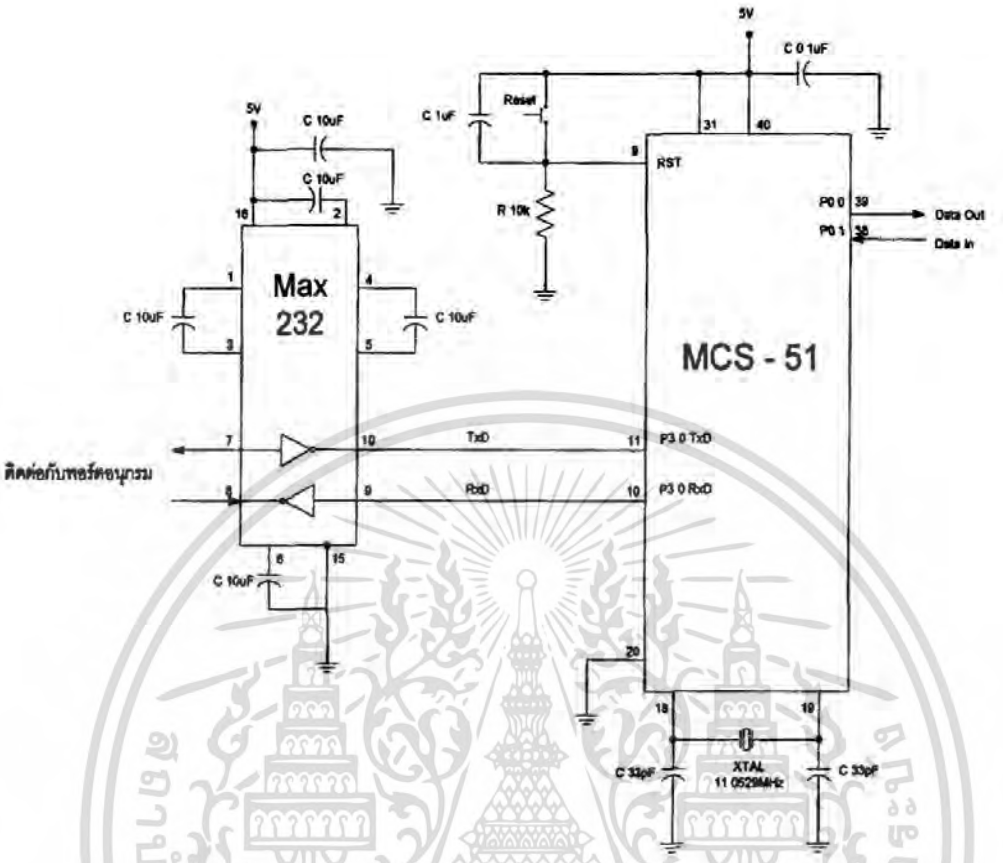
รูปที่ 3.3 แสดงบล็อกไดอะแกรมของการควบคุมมอเตอร์

การทำงานของบล็อกนี้คือ โปรแกรม Vasual basic 6.0 จะทำการควบคุมส่วนต่างๆ ของหุ่นยนต์ ซึ่งประกอบด้วยเซอร์โวมอเตอร์และเอ็นโค๊ดเดอร์ โดยที่เอ็นโค๊ดเดอร์จะทำหน้าที่ตรวจจับระยะทางแล้วนำค่าส่งไปยังภาครับเพื่อแสดงผลทางหน้าจอ

3.1.1 การควบคุมผ่านคอมพิวเตอร์

จะใช้ส่งข้อมูลผ่านพอร์ตอนุกรม โดย microcontroller ตระกูล MCS-51 ได้ทำการสร้างพอร์ตการสื่อสารแบบอนุกรมจัดเตรียมไว้ภายหลังในตัวเองแล้วดังนั้นการรับส่งข้อมูลแบบอนุกรมสามารถที่จะส่งและรับข้อมูลแบบอนุกรมโดยการเขียนโปรแกรมควบคุม MCS 51 โดยจะรับข้อมูลจากแป้นคีย์บอร์ดแล้วส่งข้อมูลที่ได้รับไปยังพอร์ตอนุกรมเป็นข้อมูล 8 บิตอัตราบอร์ดเท่ากับ 19200 บิตต่อวินาที ซึ่งเป็นรูปแบบมาตรฐานที่ใช้กับคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



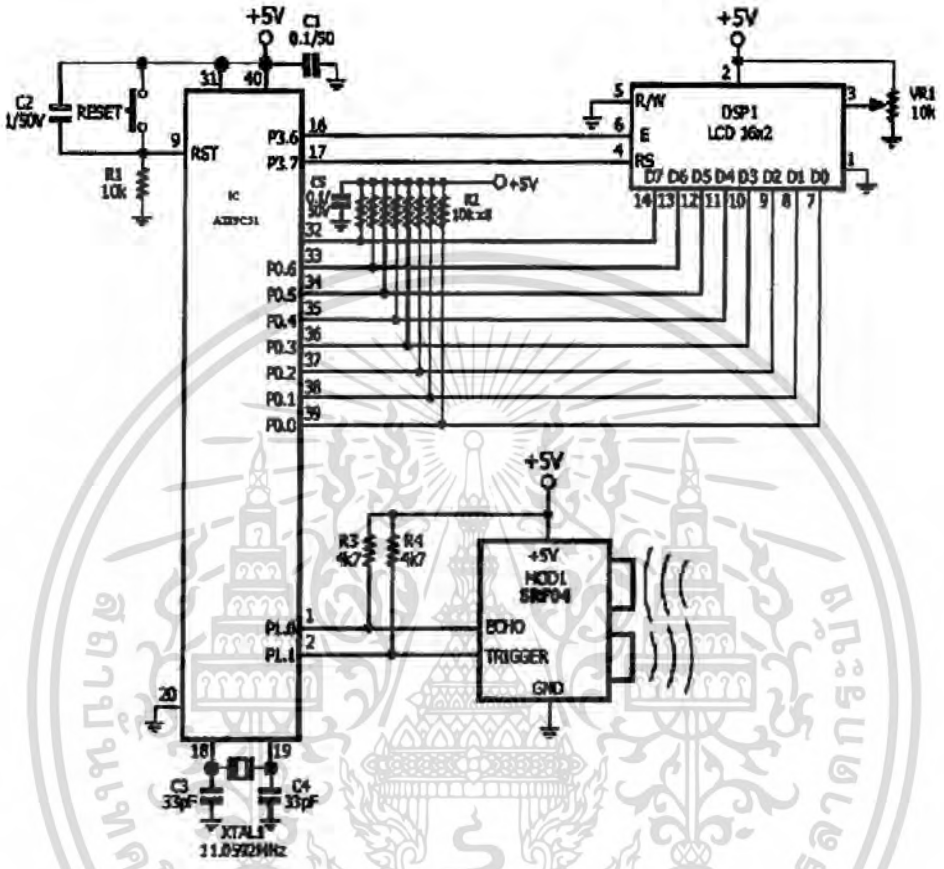
รูปที่ 3.4 การใช้งานไมโครคอนโทรลเลอร์ติดต่อกับพอร์ตอนุกรม

3.1.2 การรับ-ส่ง ข้อมูลผ่านพอร์ตอนุกรม

คอมพิวเตอร์ทำการส่งข้อมูลผ่านพอร์ตอนุกรมตามมาตรฐานRS-232โดยการต่อวงจรไมโครคอนโทรลเลอร์ดังรูปที่ 3.4 โดยต่อเข้ากับไอซี MAX 232 จากรูปวงจรที่ 3.4 เมื่อคอมพิวเตอร์ส่งข้อมูลในรูปของรหัส ASCII ไปให้ไมโครคอนโทรลเลอร์ ซึ่งจะเซตการทำงานให้ทำงานในการรับส่งข้อมูลแบบอนุกรม ตัวไมโครคอนโทรลเลอร์จะทำหน้าที่รับข้อมูลสิ่งที่สำคัญในการรับส่งข้อมูลแบบอนุกรมระหว่างคอมพิวเตอร์กับไมโครคอนโทรเลอร์ในโครงงานนี้ คือความเร็วของอัตราการรับส่งข้อมูลที่จะต้องเท่ากันระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์เนื่องจากการทำงานที่เป็นแบบอะซิงโครนัสจึงต้องกำหนดอัตรารวมเร็วที่เท่ากันโดยในโครงงานนี้เราให้ความเร็วที่ 19200 บิตต่อวินาที ซึ่งเราต้องโปรแกรมลงในไมโครคอนโทรลเลอร์ให้ทำงานที่อัตราความเร็วที่เท่ากันด้วยโดยการกำหนดในส่วนต้นของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

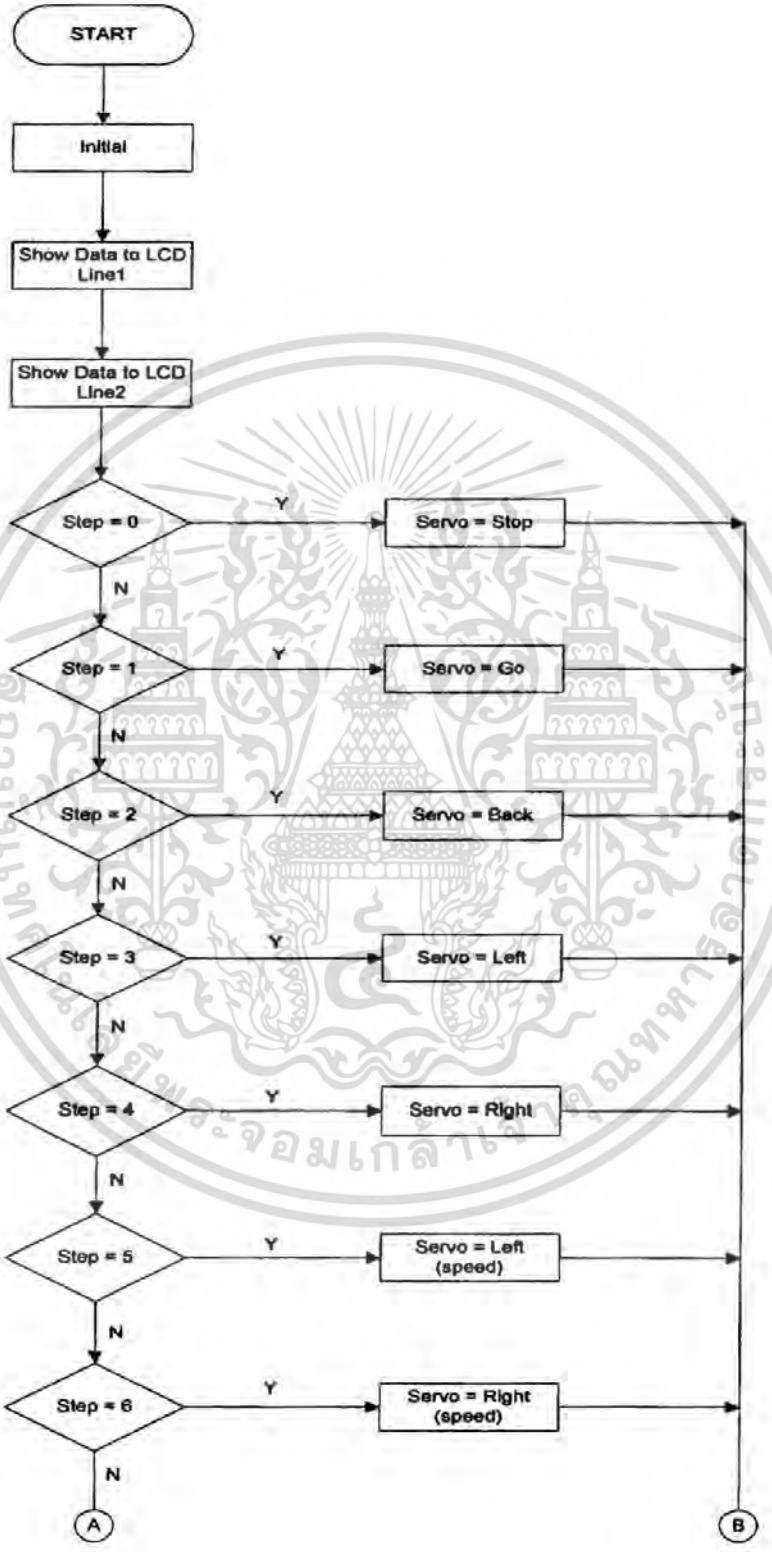
3.1.3 การใช้งานระหว่าง ไมโครคอนโทรลเลอร์ AT89C52 กับ SRF05 โมดูลวัดระยะทาง



รูปที่ 3.5 แสดงวงจรทดลองติดต่อไมโครคอนโทรลเลอร์ AT89C52 กับ SRF05 โมดูลวัดระยะทาง

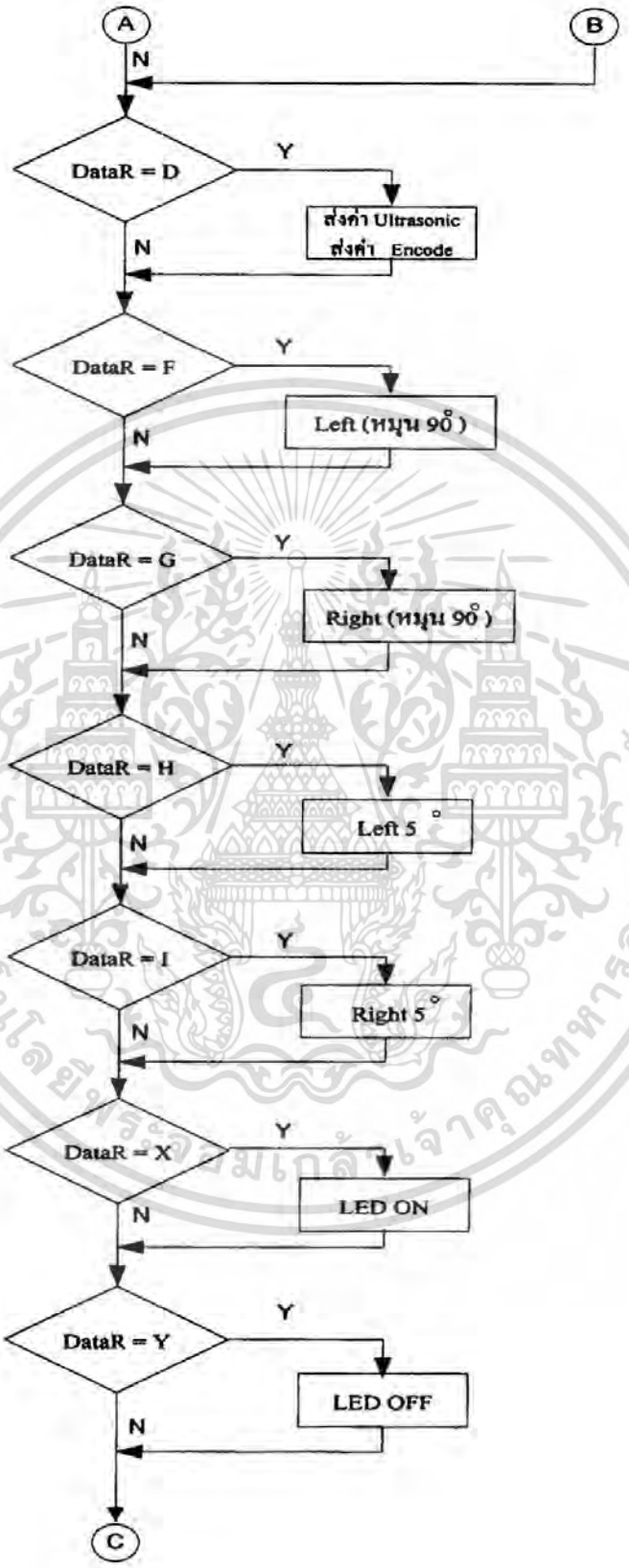
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบทางด้าน Soft Ware



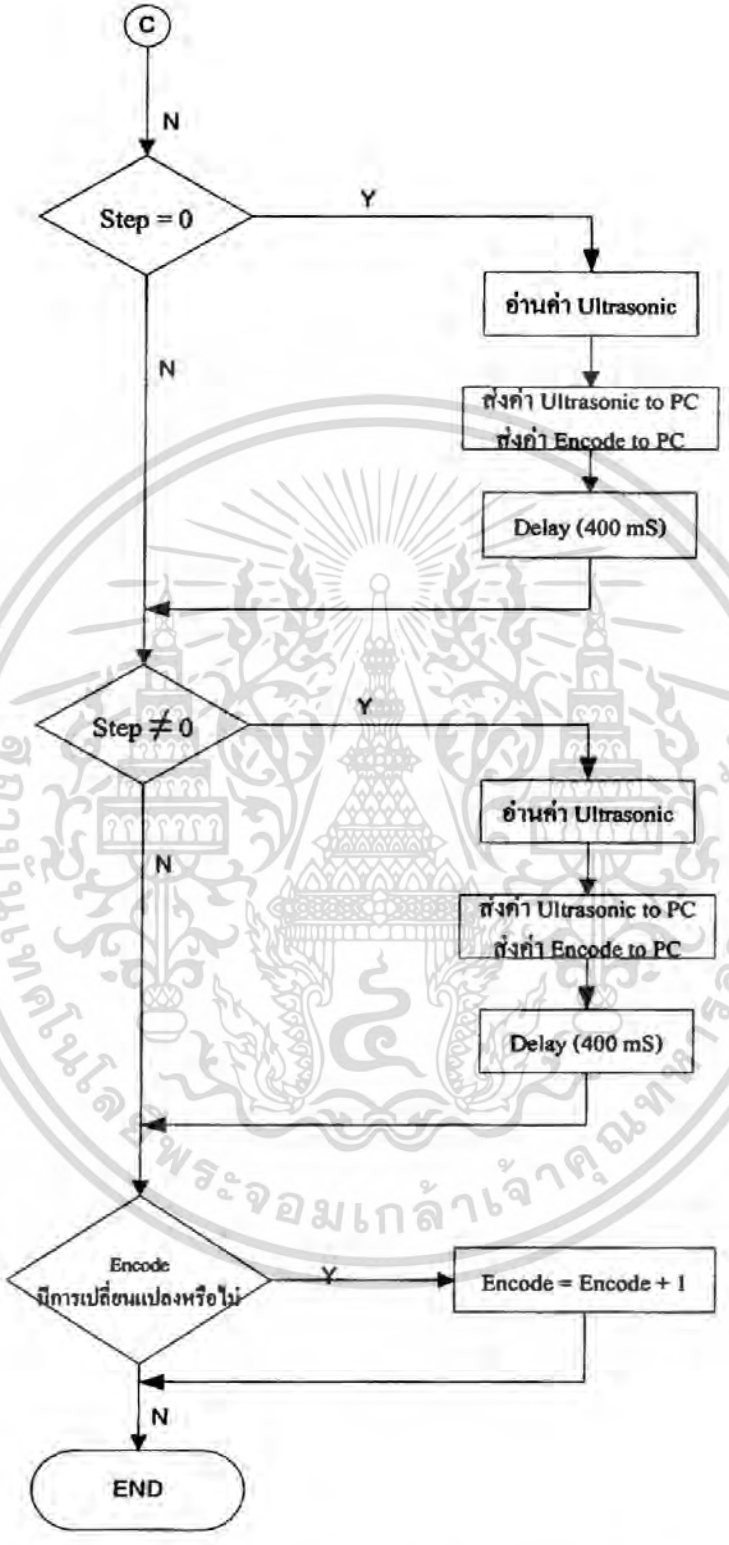
รูปที่ 3.6 โฟลว์ชาร์ตแสดงการควบคุมทิศทางการเคลื่อนที่ของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



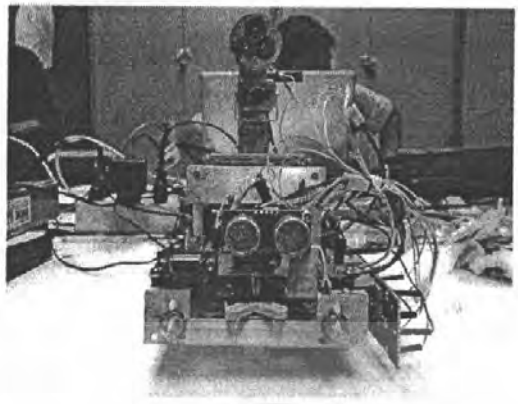
รูปที่ 3.7 โฟลว์ชาร์ตแสดงการทำงานของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



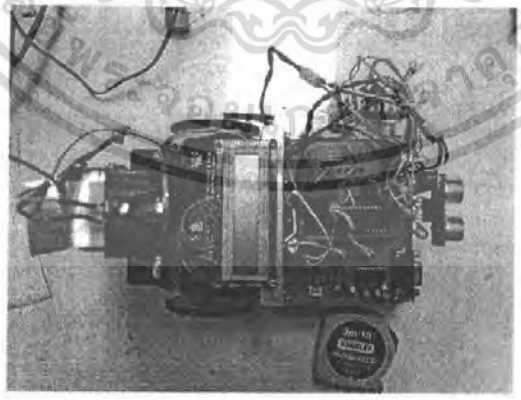
รูปที่ 3.8 โฟล์วชาร์ตแสดงการส่งค่าอัลตร้าโซนิกและเอ็นโค๊ดเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แสดงด้านหน้าของหุ่นยนต์

รูปที่ 3.10 แสดงด้านข้างของหุ่นยนต์



รูปที่ 3.11 แสดงด้านบนของหุ่นยนต์

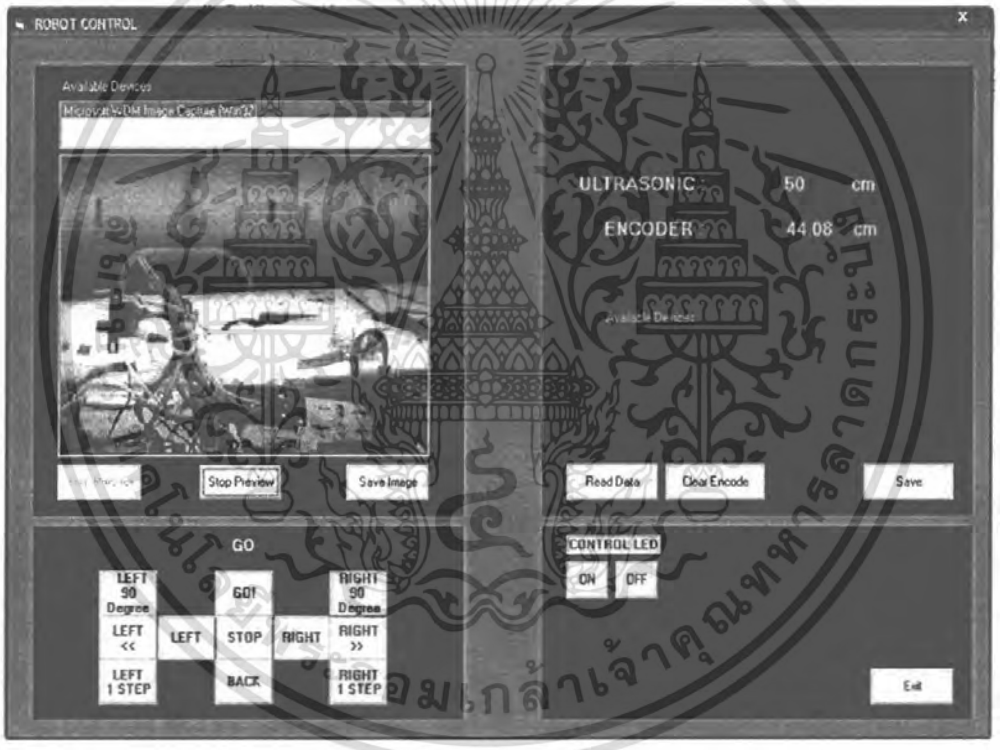
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 ทดสอบการรับภาพจากกล้องโดยใช้โปรแกรมวิชวลเบสิก

ทำการทดสอบรับ-ส่งภาพ ซึ่งสามารถรับภาพจากกล้องมาแสดงที่คอมพิวเตอร์



รูปที่ 4.1 แสดงหน้าต่างโปรแกรมการรับ-ส่งภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 ผลการทดลองระยะการเคลื่อนที่ของรถสำรวจและระยะการรับภาพ

ระยะทาง (เมตร)	สถานะการรับภาพ	
	วัดในที่โล่ง	วัดในห้อง
10	ชัดเจน	ชัดเจน
20	ชัดเจน	ชัดเจน
30	ชัดเจน	ไม่ชัด (มีสัญญาณรบกวนเล็กน้อย)
40	ชัดเจน	ไม่ชัด (มีสัญญาณรบกวนเล็กน้อย)
50	ชัด (มีสัญญาณรบกวนเล็กน้อย)	ไม่ชัด (มีสัญญาณรบกวนมาก)
60	ชัด (มีสัญญาณรบกวนเล็กน้อย)	ไม่ชัด (มีสัญญาณรบกวนมาก)
70	ชัด (มีสัญญาณรบกวนเล็กน้อย)	ไม่ชัด (มีสัญญาณรบกวนมาก)
80	ไม่ชัด (มีสัญญาณรบกวนมาก)	มองไม่เห็นภาพ
90	ไม่ชัด (มีสัญญาณรบกวนมาก)	มองไม่เห็นภาพ

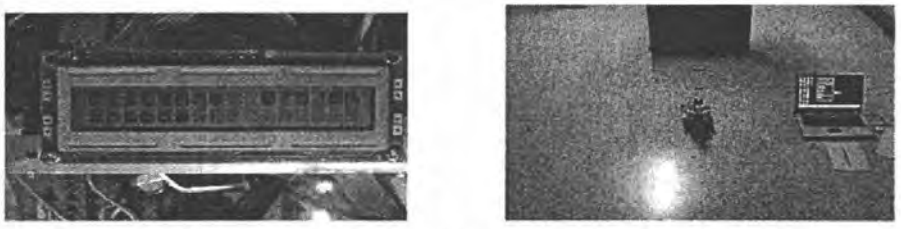
4.2 ผลการทดลองการรับค่าอัลตร้าโซนิก

เราได้ทำการทดลองโดยให้หุ่นยนต์วัดระยะทางนั้นทำการวัดระยะในระยะที่เราได้กำหนดไว้ คือ 25 cm, 50 cm, 75 cm, 100 cm, …, 400cm ผลการทดลองที่ได้เป็น ดังนี้

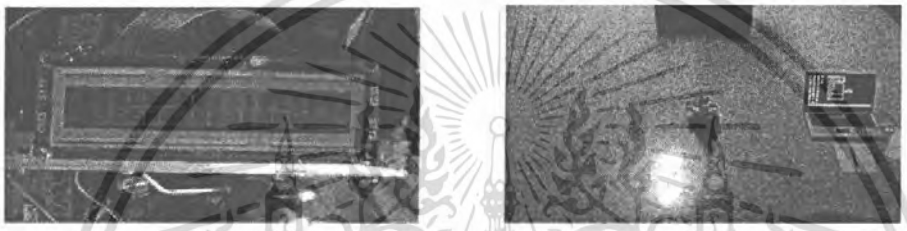
รูปที่ 4.2 แสดงการวัดระยะในระยะ 25 เซนติเมตร

รูปที่ 4.3 แสดงการวัดระยะในระยะ 50 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงการวัดระยะในระยะ 75 เซนติเมตร



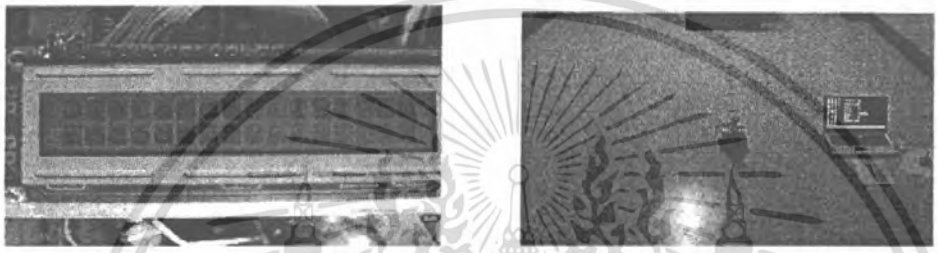
รูปที่ 4.5 แสดงการวัดระยะในระยะ 100 เซนติเมตร

รูปที่ 4.6 แสดงการวัดระยะในระยะ 125 เซนติเมตร

รูปที่ 4.7 แสดงการวัดระยะในระยะ 150 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.8 แสดงการวัดระยะในระยะ 175 เซนติเมตร

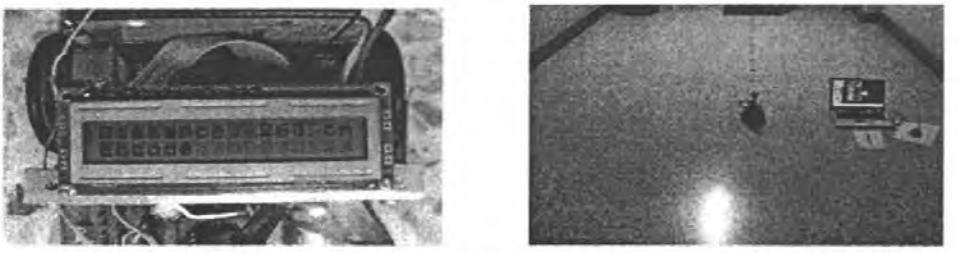


รูปที่ 4.9 แสดงการวัดระยะในระยะ 200 เซนติเมตร

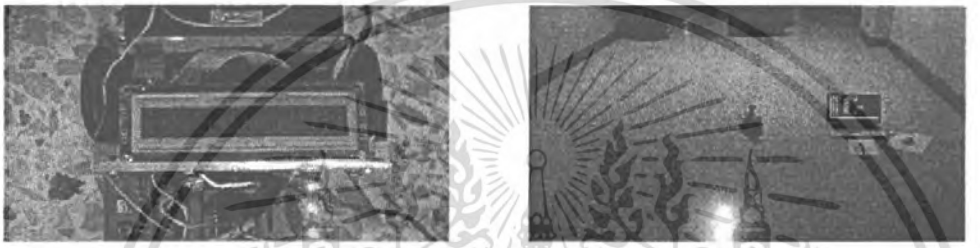
รูปที่ 4.10 แสดงการวัดระยะในระยะ 225 เซนติเมตร

รูปที่ 4.11 แสดงการวัดระยะในระยะ 250 เซนติเมตร

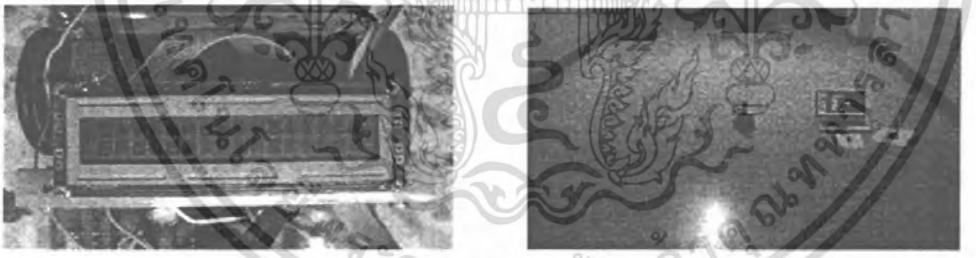
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



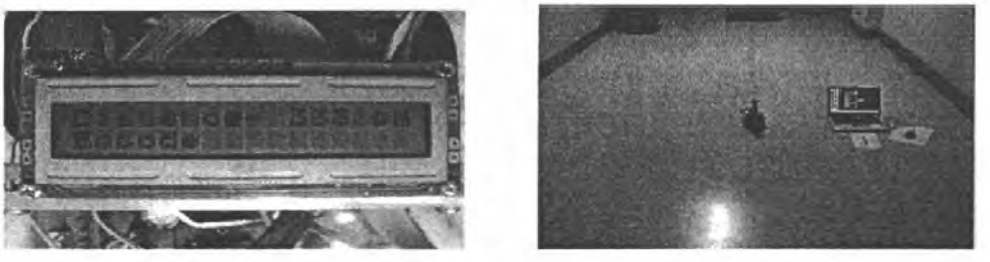
รูปที่ 4.12 แสดงการวัดระยะในระยะ 275 เซนติเมตร



รูปที่ 4.13 แสดงการวัดระยะในระยะ 300 เซนติเมตร

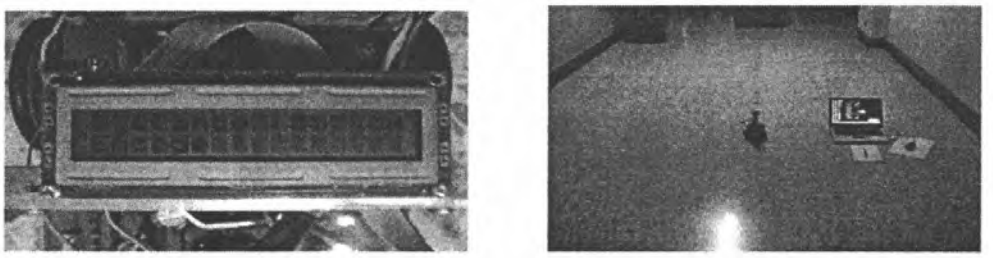


รูปที่ 4.14 แสดงการวัดระยะในระยะ 325 เซนติเมตร

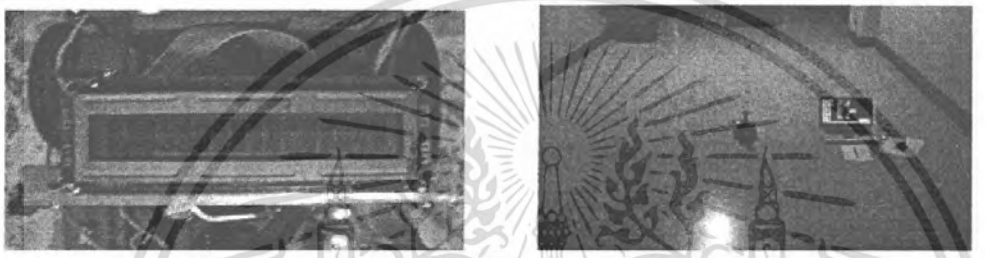


รูปที่ 4.15 แสดงการวัดระยะในระยะ 350 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 แสดงการวัดระยะในระยะ 375 เซนติเมตร



รูปที่ 4.17 แสดงการวัดระยะในระยะ 400 เซนติเมตร

รูปที่ 4.18 แสดงการทำงานในที่มืด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 แสดงค่าที่วัดได้

ระยะทาง (เซนติเมตร)	ค่าที่วัดได้	ความผิดพลาด (%)
25	25	0
50	50	0
75	75	0
100	100	0
125	122	2.40
150	144	4.00
175	167	4.50
200	191	4.50
225	215	4.40
250	230	8.00
275	263	4.30
300	297	1.00
325	311	4.30
350	333	4.80
375	357	4.80
400	405	1.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 สรุปผลการทดลอง

จากการทดลองของหุ่นยนต์วัดระยะทางนั้นสามารถสรุปผล ได้ดังนี้

1. หุ่นยนต์สามารถเคลื่อนที่ไปข้างหน้า ข้างหลัง เลี้ยวซ้าย และ เลี้ยวขวาได้ตามที่โปรแกรมสั่งงาน
2. อัลตร้าโซนิกสามารถรับ-ส่งคลื่นและสามารถวัดระยะทางได้ไกลถึง 4 เมตร
3. ความมืดไม่มีผลกระทบต่ออัลตร้าโซนิก
4. สามารถแสดงค่าที่วัดได้จากเอนโค๊ดเดอร์ทางหน้าจอ
5. สามารถนำค่าที่วัดได้จากอัลตร้าโซนิกและเอนโค๊ดเดอร์มารวมกันแล้วแสดงผลออกทางหน้าจอได้
6. กล้องสามารถมองเห็นวัตถุต่างๆได้อย่างชัดเจนในระยะ 70 เมตร
7. สามารถบันทึกค่าจากอัลตร้าโซนิกและเอนโค๊ดเดอร์ได้

4.4 ปัญหาและแนวทางแก้ไข

1. ถ้าหากเกินในระยะ 1 เมตร ขึ้นไปค่าที่แสดงผลบน LCD นั้นจะเกิดความคลาดเคลื่อนอยู่เล็กน้อยความผิดพลาดที่เกิดขึ้นแก้ไขโดยการแก้ไขคำสั่งภายในตัวโปรแกรมเพื่อเป็นการเปรียบเทียบว่าโปรแกรมนั้นผิดพลาดหรือเปล่า
2. ถ้าในที่มืดจะเห็นว่าแสงนั้นสว่างไม่เพียงพอ ไม่สามารถมองเห็นวัตถุในระยะไกลๆได้ ดังนั้นจึงควรติดตั้งอุปกรณ์ที่สามารถให้ความสว่างได้ไกล
3. การเคลื่อนที่ของหุ่นยนต์ไม่เป็นแนวตรง คือจะเคลื่อนที่ในลักษณะของการโค้งในระยะไกลอันเนื่องมาจากการหมุนของเซอร์โวมอเตอร์ไม่เท่ากันระหว่างล้อ 2 ล้อ แก้ไขได้โดย
    1. เปลี่ยนค่าในโปรแกรม
    2. เปลี่ยนตัวเซอร์โว มอเตอร์
    3. ใช้ ดีซี มอเตอร์แทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

วัชรินทร์ เคารพ. **คู่มือการใช้งาน SERVO MOTOR.** กรุงเทพมหานคร , 2546

ฉัททวุฒิ พืชผล, พิชิต สันติกุลานนท์ และพร้อมเลิศ หล่อวิจิตร. **Visual Basic 6.** ครั้งที่ 11. กรุงเทพมหานคร , 2547

ธีรวัฒน์ ประกอบผล. **การประยุกต์ใช้งานไมโครคอนโทรลเลอร์.** กรุงเทพมหานคร : สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น) , 2548

http://www.wenshing.com.tw/mozhu/TRW-2.4G-Y.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมของไมโครคอนโทรลเลอร์ MCS-51

Robot.c

```
#include<reg51.h>
#include<delay.c>
#include<lcd2l.h>
#include<ultrasonic.c>
sbit servoL = P2^0;
sbit servoR = P2^1;
sbit encode = P2^2;
sbit LED1=P1^7;
unsigned int value=0;
unsigned char timerL,timerR,step;
unsigned char encode_count=0,encode_last,encode_last1;

//=============================================================================//
void Senddata(unsigned char datas)    //Send data
        {
        SBUF = datas;
        while(~TI);
        TI = 0;
        }
//=============================================================================//
void serial_port(void)interrupt 4 //interrup serial port
        {
                unsigned char dataR;
                int x;
                EA = 0;
                if(RI == 1)
                {
                        while(~RI);
                        dataR = SBUF;
                        RI = 0;
                        if(dataR == 'R')     //start bit control
                        {
                        while(~RI);
                                dataR = SBUF;
                                RI = 0;
                                step = dataR;        //step control
                        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(dataR == 'C')
{
encode_count=0;
}
if(dataR == 'D')
{
Senddata('D');              //send distance to PC
Senddata(value);            //send distance to PC
Senddata('E');
Senddata(encode_count);     //send encode to PC
}
//=============================================================================//
if(dataR== 'F')                         //LEFT 90
{
for (x=0;x<90;x++)
        {
        timerL = 1;
        servoL = 1;
        DelayMs(timerL);            //servoL = 1ms,servoR = 1.5ms
        servoL = 0;
        DelayMs(20-timerL);
        }
}
if (dataR=='G')                         //LEFT 90
{
for (x=0;x<91;x++)
        {
        timerR = 2;
        servoR = 1;
        DelayMs(timerR);            //servoL = 1ms,servoR = 1.5ms
        servoR = 0;
        DelayMs(20-timerR);
        }
}

if (dataR=='H')             //LEFT 1 STEP
{
        timerL = 1;
        servoL = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                DelayMs(timerL);            //servoL = 1ms
                servoL = 0;
                DelayMs(20-timerL);
        }
        if (dataR=='I')                 //RIGHT 1 STEP
        {
                timerR = 2;
                servoR = 1;
                DelayMs(timerR);            //servoL = 2ms
                servoR = 0;
                DelayMs(20-timerR);
        }
        if (dataR=='X')
        {
        LED1=0;
        }
        if (dataR=='Y')
        {
        LED1=1;
        }
//=============================================================================//
        }
        EA = 1;
    }
//=============================================================================//
void main()                             //Main Program
    {
    unsigned int datadiv;
    unsigned char i;
    unsigned char datashow1[16] = "Distance      ";
    unsigned char datashow2[16] = "Encode        ";

    EA      =       1;
    ES      =       1;
    SCON = 0x50;                            // Serial Port Mode 1 (N,8,1)
    PCON = 0x80;
    TMOD = 0x21;
    TH1 = 0xFD;
    TR1 = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น มีข้อห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TI = 0;
RI = 0;
step = '0';
inti_lcd() ;                                        //set lcd

lcd_set(0x80);
for(i=0;i<16;i++)
        {
        lcd_display(datashow1[i]);          //show "Distance      "
        }
lcd_set(0xc0);

for(i=0;i<16;i++)
        {
        lcd_display(datashow2[i]);          //show "Encode        "
        }
encode_last1 = encode_last;
while(1)
{
switch(step)
        {
        case '0':                                   //STOP
                servoL = 0;
                servoR = 0;

        break;
        case '1':                                   //GO
                timerL = 1;
                timerR = 1;
                servoL = 1;
                servoR = 1;
                DelayMs(timerR);            //servoR = 1ms
                servoR = 0;
                DelayMs(timerL);            //servoL = 2ms
                servoL = 0;
                DelayMs(18);
        break;
        case '2':                                   //BACK
                timerL = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            timerR = 1;
            servoL = 1;
            servoR = 1;
            DelayMs(timerL);            //servoL = 1ms
            servoL = 0;
            DelayMs(timerR);            //servoR = 2ms
            servoR = 0;
            DelayMs(20-timerR);
    break;
    case '3':                           //LEFT
            timerR = 1;
            servoL = 0;
            servoR = 1;
            DelayMs(timerR);            //servoL = 1ms,servoR = 1ms
            servoL = 0;
            servoR = 0;
            DelayMs(20-timerR);
    break;
    case '4':                           //RIGHT
            timerR = 2;
            servoL = 1;
            servoR = 0;
            DelayMs(timerR);            //servoL = 2ms,servoR = 2ms
            servoL = 0;
            servoR = 0;
            DelayMs(20-timerR);
    break;
    case '5':                           //LEFT >>
            timerR = 1;
            servoL = 1;
            servoR = 1;
            DelayMs(timerR);            //servoL = 1ms,servoR = 1ms
            servoL = 0;
            servoR = 0;
            DelayMs(20-timerR);
    break;
    case '6':                           //RIGHT >>
            timerR = 2;
            servoL = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                servoR = 1;
                DelayMs(timerR);            //servoL = 2ms,servoR = 2ms
                servoL = 0;
                servoR = 0;
                DelayMs(20-timerR);
            break;

            }
if(step == '0')
{
        value = distance();                 //read data from ultrasonic
        lcd_set(0x8a) ;                     //set start show lcd
        datadiv = value%1000;
        lcd_display((datadiv/100)+0x30);
        datadiv = datadiv%100;
        lcd_display((datadiv/10)+0x30);
        datadiv = datadiv%10;
        lcd_display(datadiv+0x30);
        lcd_set(0x8e) ;
        lcd_display('c');
        lcd_display('m');
        Senddata('D');                      //send byte start 'D' to PC
        Senddata(value);                    //send distance to PC
        Senddata('E');                      // send byte start 'E' to PC
        Senddata(encode_count);             //send encode to PC
        DelayMs(400);
}

if(step != '0')
{
        value = distance();                 //read data from ultrasonic
        lcd_set(0x8a) ;                     //set start show lcd
        datadiv = value%1000;
        lcd_display((datadiv/100)+0x30);
        datadiv = datadiv%100;
        lcd_display((datadiv/10)+0x30);
        datadiv = datadiv%10;
        lcd_display(datadiv+0x30);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            lcd_set(0x8e) ;
            lcd_display('c');
            lcd_display('m');

            Senddata('D');                  //send byte start 'D' to PC
            Senddata(value);                //send distance to PC
            Senddata('E');                  // send byte start 'E' to PC
            Senddata(encode_count);         //send encode to PC
            //DelayMs(400);
}

if(encode_last1 != encode_last)         //check encode
            {
            encode_count = encode_count+1;      //count encode
            lcd_set(0xca) ;
            datadiv = encode_count%1000;
            lcd_display((datadiv/100)+0x30);
            datadiv = datadiv%100;
            lcd_display((datadiv/10)+0x30);
            datadiv = datadiv%10;
            lcd_display(datadiv+0x30);
            lcd_set(0xce) ;
            lcd_display('c');
            lcd_display('m');
            encode_last1 = encode_last;
            }
if (encode ==1)
{encode_last = 1;}
else
{encode_last = 0;}
            }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ultrasonic.c

```
#include<reg51.h>
sbit echo = P2^3;
sbit trigger = P2^4;
void Delay_ultra_Us(int us)
{
        int i;
        for (i=0; i<us; i++);
}
void trigger_pulse(void)
        {
        //unsigned char i;
        trigger = 1;
        //for(i=0;i<10;i++);
        Delay_ultra_Us(10);
        trigger = 0;
        }
unsigned int distance()
        {
        unsigned int mc;
        trigger = 0;
        echo = 1;
        //TMOD = 0x01; //set at main Program
        TH0 = 0x00;
        TL0 = 0x00;
        TF0 = 0;
        TR0 = 0;
        trigger_pulse();
        while(!echo);
        TR0=1;
        while(echo);
        TR0 = 0;
        TF0 = 0;
        mc = TH0;
        mc <<= 8;
        mc += TL0;
        return(mc/56);
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมหลักที่ใช้ควบคุมการทำงานของหุ่นยนต์

```
Option Explicit
Const WM_CAP As Integer = &H400
Const WM_CAP_DRIVER_CONNECT As Long = WM_CAP + 10
Const WM_CAP_DRIVER_DISCONNECT As Long = WM_CAP + 11
Const WM_CAP_EDIT_COPY As Long = WM_CAP + 30
Const WM_CAP_SET_PREVIEW As Long = WM_CAP + 50
Const WM_CAP_SET_PREVIEWRATE As Long = WM_CAP + 52
Const WM_CAP_SET_SCALE As Long = WM_CAP + 53
Const WS_CHILD As Long = &H40000000
Const WS_VISIBLE As Long = &H10000000
Const SWP_NOMOVE As Long = &H2
Const SWP_NOSIZE As Integer = 1
Const SWP_NOZORDER As Integer = &H4
Const HWND_BOTTOM As Integer = 1
Dim iDevice As Long
Dim hHwnd As Long
Private Declare Function SendMessage Lib "user32" Alias "SendMessageA" _
  (ByVal hwnd As Long, ByVal wMsg As Long, ByVal wParam As Long, _
  lParam As Any) As Long
Private Declare Function SetWindowPos Lib "user32" (ByVal hwnd As Long, _
  ByVal hWndInsertAfter As Long, ByVal x As Long, ByVal y As Long, _
  ByVal cx As Long, ByVal cy As Long, ByVal wFlags As Long) As Long
Private Declare Function DestroyWindow Lib "user32" (ByVal hndw As Long) As Boolean
Private Declare Function capCreateCaptureWindowA Lib "avicap32.dll" _
  (ByVal lpszWindowName As String, ByVal dwStyle As Long, _
  ByVal x As Long, ByVal y As Long, ByVal nWidth As Long, _
  ByVal nHeight As Integer, ByVal hWndParent As Long, _
  ByVal nID As Long) As Long
Private Declare Function capGetDriverDescriptionA Lib "avicap32.dll" (ByVal wDriver As Long, _
  ByVal lpszName As String, ByVal cbName As Long, ByVal lpszVer As String, _
  ByVal cbVer As Long) As Boolean
Private Sub CmdControl_Click(Index As Integer)
Dim bufOut As String
Select Case Index
Case 0:
    labelstatus.Caption = "GO!"
    bufOut = "R1"      'GO!
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Case 1:
    labelstatus.Caption = "BACK!"
    bufOut = "R2"      'BACK
Case 2:
    labelstatus.Caption = "LEFT"
    bufOut = "R3"      'LEFT
Case 3:
    labelstatus.Caption = "RIGHT"
    bufOut = "R4"      'RIGHT
Case 4:
    labelstatus.Caption = "STAND BY"
    bufOut = "R0"      'STOP
Case 5:
    labelstatus.Caption = "<<LEFT"
    bufOut = "R5"      '<< LEFT
Case 6:
    labelstatus.Caption = "RIGHT >>"
    bufOut = "R6"      'RIGHT >>
End Select
Call senddata(bufOut)
End Sub
Private Sub cmdExit_Click()
'------ Close Preview Camera '--------
ClosePreviewWindow
cmdStop.Enabled = False
cmdSave.Enabled = False
cmdStart.Enabled = True
End
End Sub
Private Sub CmdL1step_Click() ' LEFT 1 STEP
 labelstatus.Caption = "' LEFT 1 STEP"
Call senddata("H")
End Sub
Private Sub CmdR1step_Click() 'RIGHT 1 STEP
labelstatus.Caption = "RIGHT 1 STEP"
Call senddata("I")
End Sub
Private Sub cmdRead_Click() '--------- ULTRASONIC ---------
lblultrasonic.Caption = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
lblencode.Caption = 0
Call senddata("D")
End Sub
Private Sub cmdreset_Click() '---------- ULTRASONIC ----------
Call senddata("C")
End Sub
Private Sub cmdSave_Click()
   Dim bm As Image
   SendMessage hHwnd, WM_CAP_EDIT_COPY, 0, 0
   ClosePreviewWindow
   picCapture.Picture = Clipboard.GetData
   CommonDialog1.CancelError = True
   CommonDialog1.FileName = "Webcam1"
   CommonDialog1.Filter = "Bitmap |*.bmp"
   On Error GoTo NoSave
   CommonDialog1.ShowSave
   SavePicture picCapture.Image, CommonDialog1.FileName
NoSave:
   cmdStop.Enabled = False
   cmdSave.Enabled = False
   cmdStart.Enabled = True
End Sub
Private Sub cmdStart_Click()
   iDevice = lstDevices.ListIndex
   OpenPreviewWindow
End Sub
Private Sub cmdStop_Click()
   ClosePreviewWindow
   cmdStop.Enabled = False
   cmdSave.Enabled = False
   cmdStart.Enabled = True
End Sub
Private Sub Command1_Click()
Call senddata("X")
End Sub
Private Sub Command2_Click()
Call senddata("Y")
End Sub
Private Sub CmdSavetxt_Click()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim Total As Variant
Total = Val(lblultrasonic.Caption) + Val(lblencode.Caption) + 16
Open "C:\Encode.txt" For Append As #2
Print #2, Total
Close #2
End Sub
Private Sub Form_Load()
Form1.BorderStyle = 3
  '---------- ULTRASONIC ------------------------
    'MSComm1.PortOpen = True
  MSComm1.RThreshold = 1
  lblultrasonic.Caption = 0
  lblencode.Caption = 0
  '----------------------------------------------
  LoadDeviceList
  If lstDevices.ListCount > 0 Then
    lstDevices.Selected(0) = True
  Else
  cmdStart.Enabled = False
    lstDevices.AddItem ("äÁè¼°¡ÅéÍ§ WebCam")
  End If
  cmdStop.Enabled = False
  cmdSave.Enabled = False
End Sub
Private Sub LoadDeviceList()
  Dim strName As String
  Dim strVer As String
  Dim iReturn As Boolean
  Dim x As Long
  x = 0
  strName = Space(100)
  strVer = Space(100)
  Do
    iReturn = capGetDriverDescriptionA(x, strName, 100, strVer, 100)
    If iReturn Then lstDevices.AddItem Trim$(strName)
    x = x + 1
  Loop Until iReturn = False
End Sub
Private Sub OpenPreviewWindow()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ' Open Preview window in picturebox
    hHwnd = capCreateCaptureWindowA(iDevice, WS_VISIBLE Or WS_CHILD, 0, 0, 640, _
      480, picCapture.hwnd, 0)
    ' Connect to device
    If SendMessage(hHwnd, WM_CAP_DRIVER_CONNECT, iDevice, 0) Then
      'Set the preview scale
      SendMessage hHwnd, WM_CAP_SET_SCALE, True, 0
      'Set the preview rate in milliseconds
      SendMessage hHwnd, WM_CAP_SET_PREVIEWRATE, 66, 0
      'Start previewing the image from the camera
      SendMessage hHwnd, WM_CAP_SET_PREVIEW, True, 0
      ' Resize window to fit in picturebox
      SetWindowPos hHwnd, HWND_BOTTOM, 0, 0, picCapture.ScaleWidth, picCapture.ScaleHeight, _
          SWP_NOMOVE Or SWP_NOZORDER
      cmdSave.Enabled = True
      cmdStop.Enabled = True
      cmdStart.Enabled = False
    Else
      ' Error connecting to device close window
      DestroyWindow hHwnd
      cmdSave.Enabled = False
    End If
  End Sub
Private Sub ClosePreviewWindow()
    ' Disconnect from device
    SendMessage hHwnd, WM_CAP_DRIVER_DISCONNECT, iDevice, 0
    ' close window
    DestroyWindow hHwnd
End Sub
Private Sub Form_Unload(Cancel As Integer)
'  If cmdStop.Enabled Then
'  ClosePreviewWindow
'  End If
     ClosePreviewWindow
    cmdStop.Enabled = False
    cmdSave.Enabled = False
    cmdStart.Enabled = True
End Sub
Private Sub senddata(ByVal data As String)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MSComm1.Output = data
End Sub
Private Sub cndRead_Click() '---------- ULTRASONIC ----------
lblultrasonic.Caption = 0
lblencode.Caption = 0
Call senddata("D")
End Sub
Private Sub L90_Click() ' LEFT 90 DEGREE
labelstatus.Caption = "LEFT 90 DEGREE"
Call senddata("F")
End Sub
Private Sub MSComm1_OnComm() '---------- ULTRASONIC ----------
Timer1.Enabled = True
End Sub
Private Sub R90_Click() 'RIGHT 90 DEGREE
labelstatus.Caption = "RIGHT 90 DEGREE"
Call senddata("G")
End Sub
Private Sub Timer1_Timer() '---------- ULTRASONIC ----------
Dim start As Integer
Dim buf_in As String
On Error Resume Next
Timer1.Enabled = False
buf_in = MSComm1.Input
If buf_in <> "" Then
start = InStr(1, buf_in, "D")
lblultrasonic.Caption = Asc(Mid(buf_in, start + 1, 1))
start = InStr(1, buf_in, "E")
lblencode.Caption = Format(Asc(Mid(buf_in, start + 1, 1)) * 1.1666667, "#0.00")
Else
End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Delay.c

```
//-------------------------------------
// Delay function
// www.GetMicro.com
//-------------------------------------
//-------------------------------------
// Delay mS function
//-------------------------------------
void DelayMs(unsigned int count)
{  // mSec Delay 11.0592 Mhz
   unsigned int i;                                  // Keil v7.5a
   while(count)
          {
     i = 115;
          while(i>0) i--;
     count--;
   }
}
//-------------------------------------
// DELAY at 11.0592MHz crystal.
// Calling the routine takes about 22us, and then
// each count takes another 17us.
// test with KEIL C51 V7.5
//-------------------------------------
void DelayUs(int us)
{
          int i;
          for (i=0; i<us; i++);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REG51.H

```
/*--------------------------------------------------------------------------
REG51.H
Header file for generic 80C51 and 80C31 microcontroller.
Copyright (c) 1988-2002 Keil Elektronik GmbH and Keil Software, Inc.
All rights reserved.
--------------------------------------------------------------------------*/

#ifndef __REG51_H__
#define __REG51_H__
/*  BYTE Register  */
sfr P0   = 0x80;
sfr P1   = 0x90;
sfr P2   = 0xA0;
sfr P3   = 0xB0;
sfr PSW  = 0xD0;
sfr ACC  = 0xE0;
sfr B    = 0xF0;
sfr SP   = 0x81;
sfr DPL  = 0x82;
sfr DPH  = 0x83;
sfr PCON = 0x87;
sfr TCON = 0x88;
sfr TMOD = 0x89;
sfr TL0  = 0x8A;
sfr TL1  = 0x8B;
sfr TH0  = 0x8C;
sfr TH1  = 0x8D;
sfr IE   = 0xA8;
sfr IP   = 0xB8;
sfr SCON = 0x98;
sfr SBUF = 0x99;
/*  BIT Register  */
/*  PSW   */
sbit CY   = 0xD7;
sbit AC   = 0xD6;
sbit F0   = 0xD5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sbit RS1  = 0xD4;
sbit RS0  = 0xD3;
sbit OV   = 0xD2;
sbit P    = 0xD0;
/* TCON */
sbit TF1  = 0x8F;
sbit TR1  = 0x8E;
sbit TF0  = 0x8D;
sbit TR0  = 0x8C;
sbit IE1  = 0x8B;
sbit IT1  = 0x8A;
sbit IE0  = 0x89;
sbit IT0  = 0x88;
/* IE  */
sbit EA   = 0xAF;
sbit ES   = 0xAC;
sbit ET1  = 0xAB;
sbit EX1  = 0xAA;
sbit ET0  = 0xA9;
sbit EX0  = 0xA8;
/* IP  */
sbit PS   = 0xBC;
sbit PT1  = 0xBB;
sbit PX1  = 0xBA;
sbit PT0  = 0xB9;
sbit PX0  = 0xB8;
/* P3 */
sbit RD   = 0xB7;
sbit WR   = 0xB6;
sbit T1   = 0xB5;
sbit T0   = 0xB4;
sbit INT1 = 0xB3;
sbit INT0 = 0xB2;
sbit TXD  = 0xB1;
sbit RXD  = 0xB0;
/* SCON */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sbit SM0  = 0x9F;
sbit SM1  = 0x9E;
sbit SM2  = 0x9D;
sbit REN  = 0x9C;
sbit TB8  = 0x9B;
sbit RB8  = 0x9A;
sbit TI   = 0x99;
sbit RI   = 0x98;
#endif
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

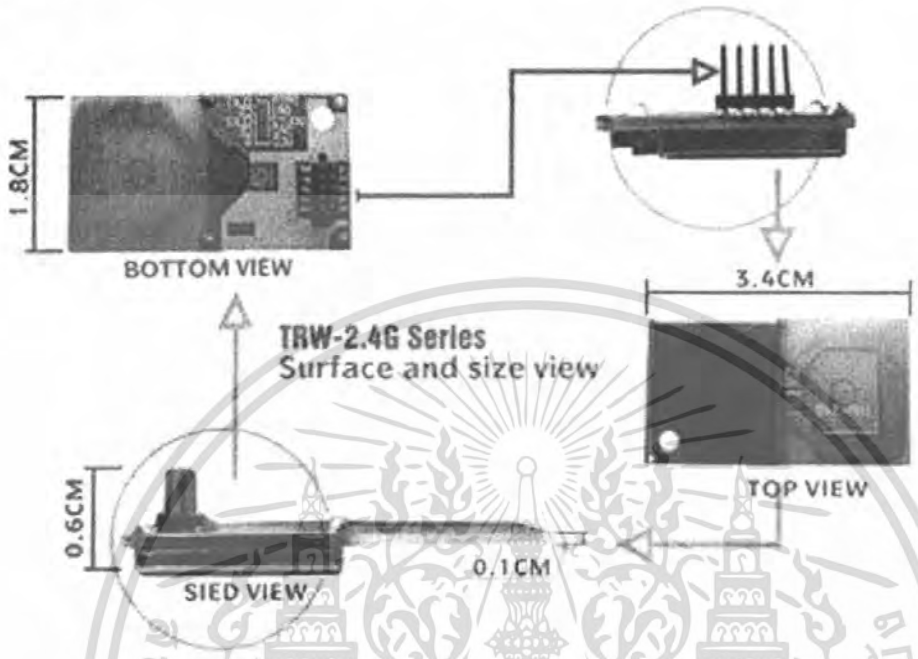
ภาคผนวก ข
คู่มือการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

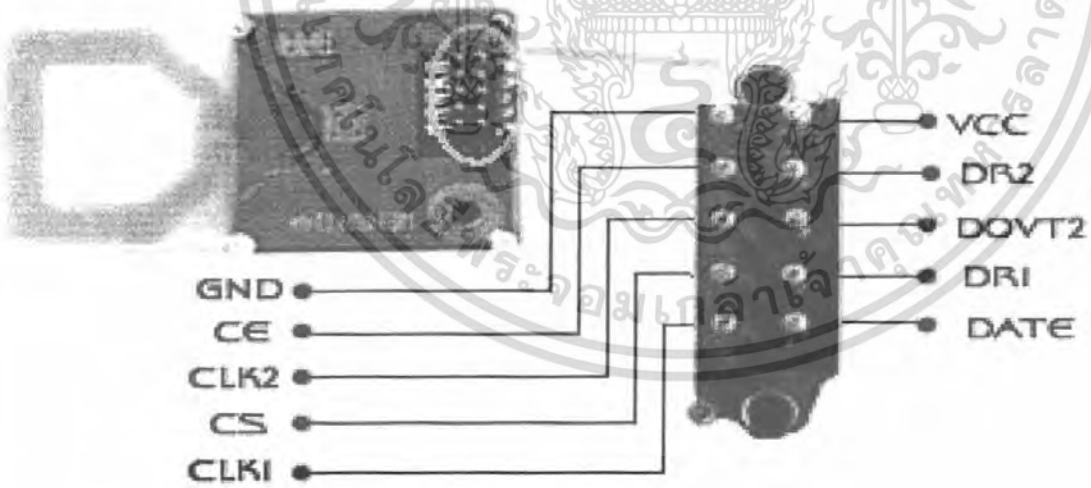
Ensemble list

Transmitter	**Receiver**	**Distance**
TRW-2.4G(250kbps)	TRW-2.4G(250kbps)	280m
TRW-2.4G(1Mbps)	TRW-2.4G(1Mbps)	150m

Suface And Size View



Wiring Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Conditions: VDD = +3V, VSS = 0V, T_A = - 40°C to + 85°C

Symbol	Parameter (condition)	Notes	Min.	Typ.	Max.	Units
	Operating conditions					
VDD	Supply voltage		1.9	3.0	3.6	V
TEMP	Operating Temperature		-40	+27	+85	°C
	Digital input pin					
V_{IH}	HIGH level input voltage		VDD- 0.3		VDD	V
V_{IL}	LOW level input voltage		Vss		0.3	V
	Digital output pin					
V_{OH}	HIGH level output voltage (I_{OH}=-0.5mA)		VDD- 0.3		VDD	V
V_{OL}	LOW level output voltage (I_{OL}=0.5mA)		Vss		0.3	V
	General RF conditions					
f_{OP}	Operating frequency	1)	2400		2524	MHz
Δf	Frequency deviation			±156		kHz
R_{GFSK}	Data rate ShockBurst™		>0		1000	kbps
$F_{CHANNEL}$	Channel spacing			1		MHz
	Transmitter operation					
P_{RF}	Maximum Output Power	4)		0	+4	dBm
P_{RFC}	RF Power Control Range		16	20		dB
P_{RFCR}	RF Power Control Range Resolution				±3	dB
P_{BW}	20dB Bandwidth for Modulated Carrier				1000	kHz
P_{RF2}	2nd Adjacent Channel Transmit Power 2MHz				-20	dBm
P_{RF3}	3rd Adjacent Channel Transmit Power 3MHz				-40	dBm
I_{VDD}	Supply current @ 0dBm output power	5)		13		mA
I_{VDD}	Supply current @ -20dBm output power	5)		8.8		mA
I_{VDD}	Average Supply current @ -5dBm output power, ShockBurst™	6)		0.8		mA
I_{VDD}	Average Supply current in stand-by mode	7)		12		µA
I_{VDD}	Average Supply current in power down			1		µA
	Receiver operation					
I_{VDD}	Supply current one channel 250kbps			18		mA
I_{VDD}	Supply current one channel 1000kbps			19		mA
I_{VDD}	Supply current two channels 250kbps			23		mA
I_{VDD}	Supply current two channels 1000kbps			25		mA
RX_{SENS}	Sensitivity at 0.1%BER (@250kbps)			-90		dBm
RX_{SENS}	Sensitivity at 0.1%BER (@1000kbps)			-80		dBm
C/I_{CO}	C/I Co-channel			6		dB
C/I_{1ST}	1st Adjacent Channel Selectivity C/I 1MHz			-1		dB
C/I_{2ND}	2nd Adjacent Channel Selectivity C/I 2MHz			-16		dB
C/I_{3RD}	3rd Adjacent Channel Selectivity C/I 3MHz			-26		dB
RX_B	Blocking Data Channel 2			-41		dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ShockBurst™

The ShockBurst™ technology uses on-chip FIFO to clock in data at a low data rate and transmit at a very high rate thus enabling extremely power reduction.
When operating the TRW-2.4G in ShockBurst™, you gain access to the high data rates (1 Mbps) offered by the 2.4 GHz band without the need of a costly, high-speed micro controller (MCU) for data processing.
By putting all high speed signal processing related to RF protocol on-chip, the TRW-2.4G offers the following benefits:

- Highly reduced current consumption
- Lower system cost (facilitates use of less expensive micro controller)
- Greatly reduced risk of 'on-air' collisions due to short transmission time

The TRW-2.4G can be programmed using a simple 3-wire interface where the data rate is decided by the speed of the micro controller.
By allowing the digital part of the application to run at low speed while maximizing the data rate on the RF link, the nRF ShockBurst™ mode reduces the average current consumption in applications considerably.

ShockBurst™ principle

When the TRW-2.4G is configured in ShockBurst™, TX or RX operation is conducted in the following way (10 kbps for the example only).

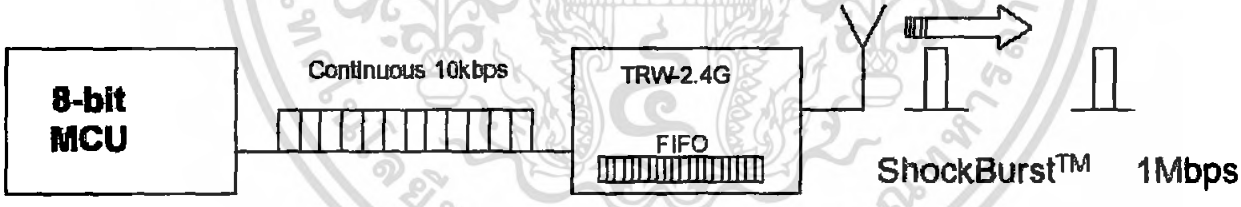
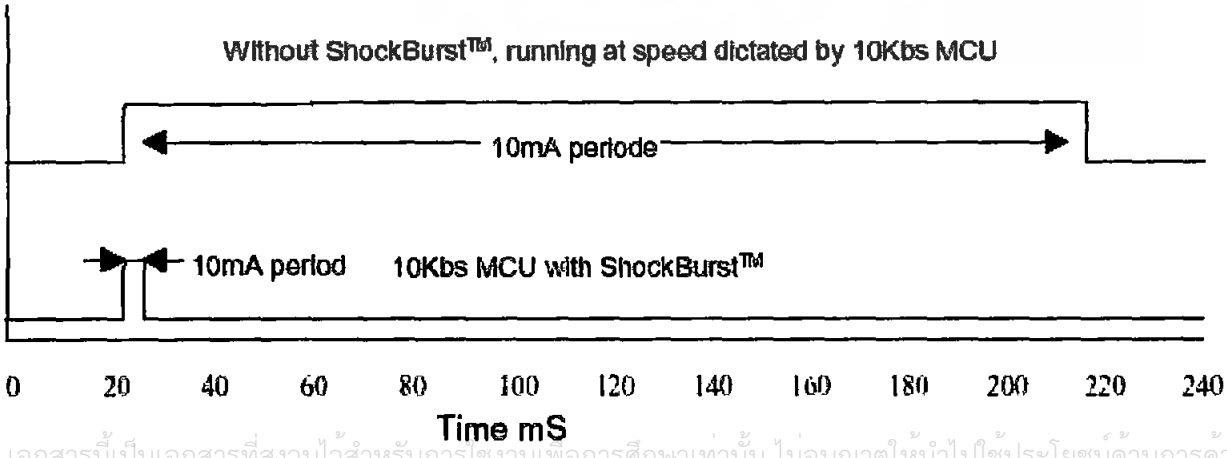


Figure 4 Clocking in data with MCU and sending with ShockBurst™ technology



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 1 Current consumption with & without ShockBurst™ technology

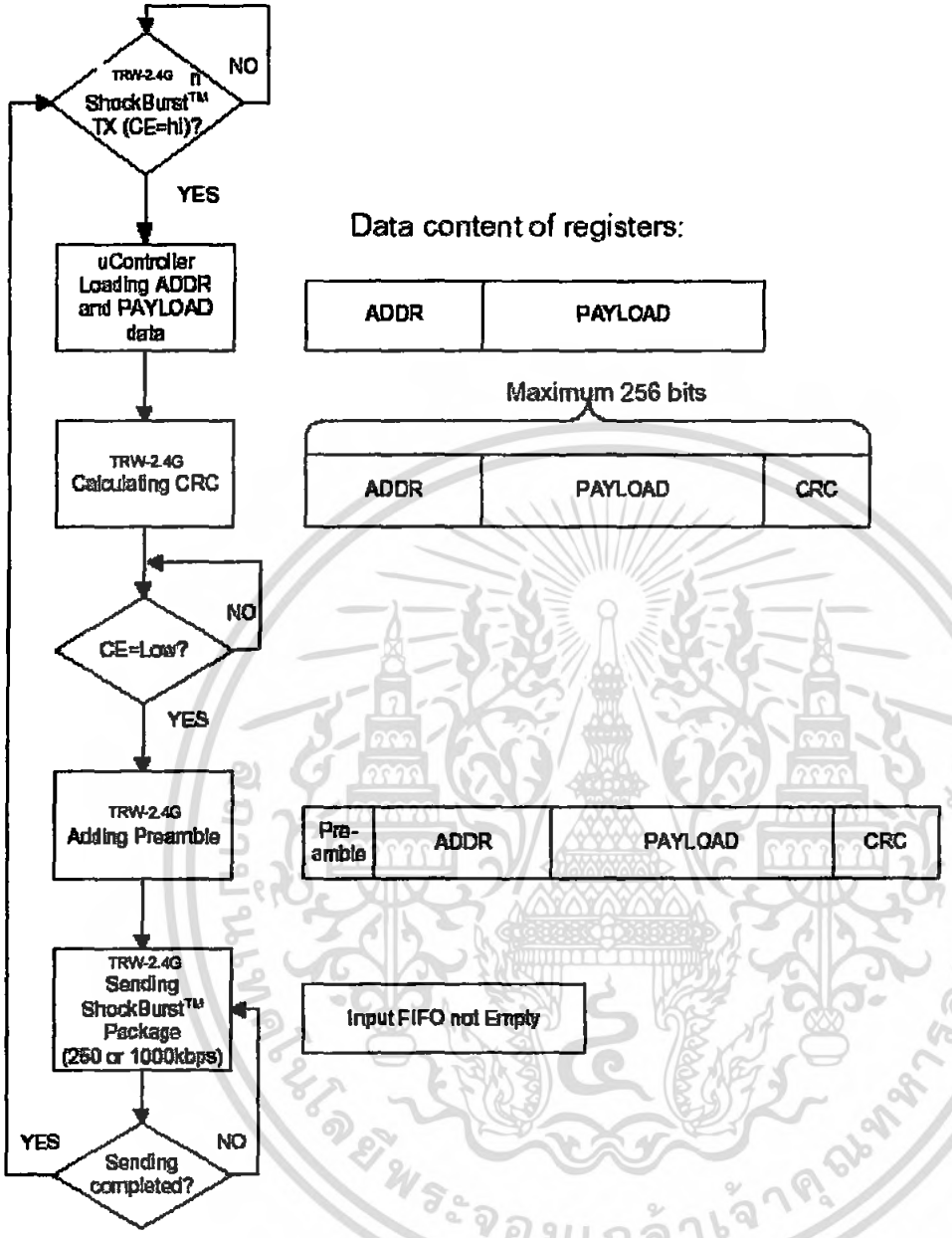


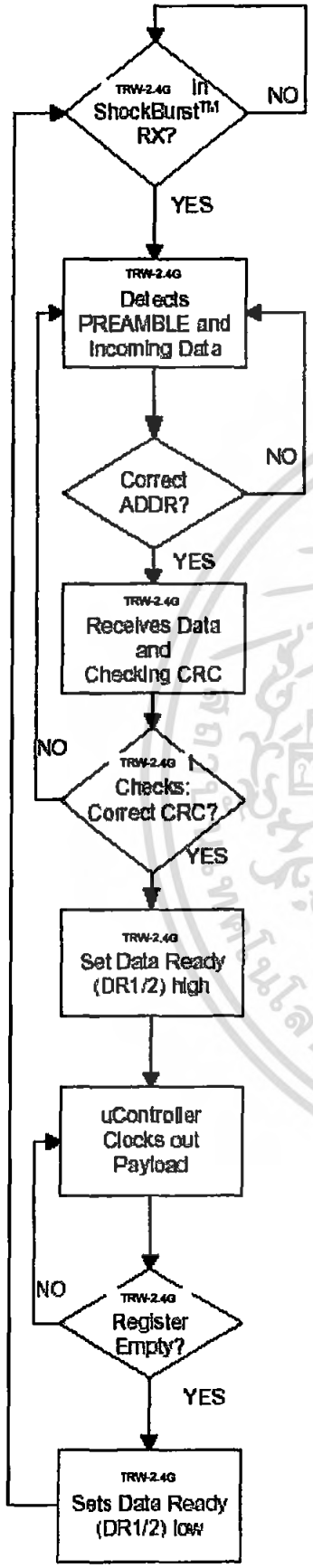
Figure 2 Flow Chart ShockBurst™ Transmit of TRW-2.4G

nRF2401 ShockBurst™ Transmit:

MCU interface pins: CE, CLK1, DATA

1. When the application MCU has data to send, set CE high. This activates TRW-2.4G on-board data processing.
2. The address of the receiving node (RX address) and payload data is clocked into the TRW-2.4G . The application protocol or MCU sets the speed <1Mbps (ex: 10kbps).
3. MCU sets CE low, this activates a TRW-2.4G ShockBurst™ transmission.
4. TRW-2.4G ShockBurst™:
   - RF front end is powered up
   - RF package is completed (preamble added, CRC calculated)
   - Data is transmitted at high speed (250 kbps or 1 Mbps configured by user).
   - TRW-2.4G return to stand-by when finished

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Data content of registers:

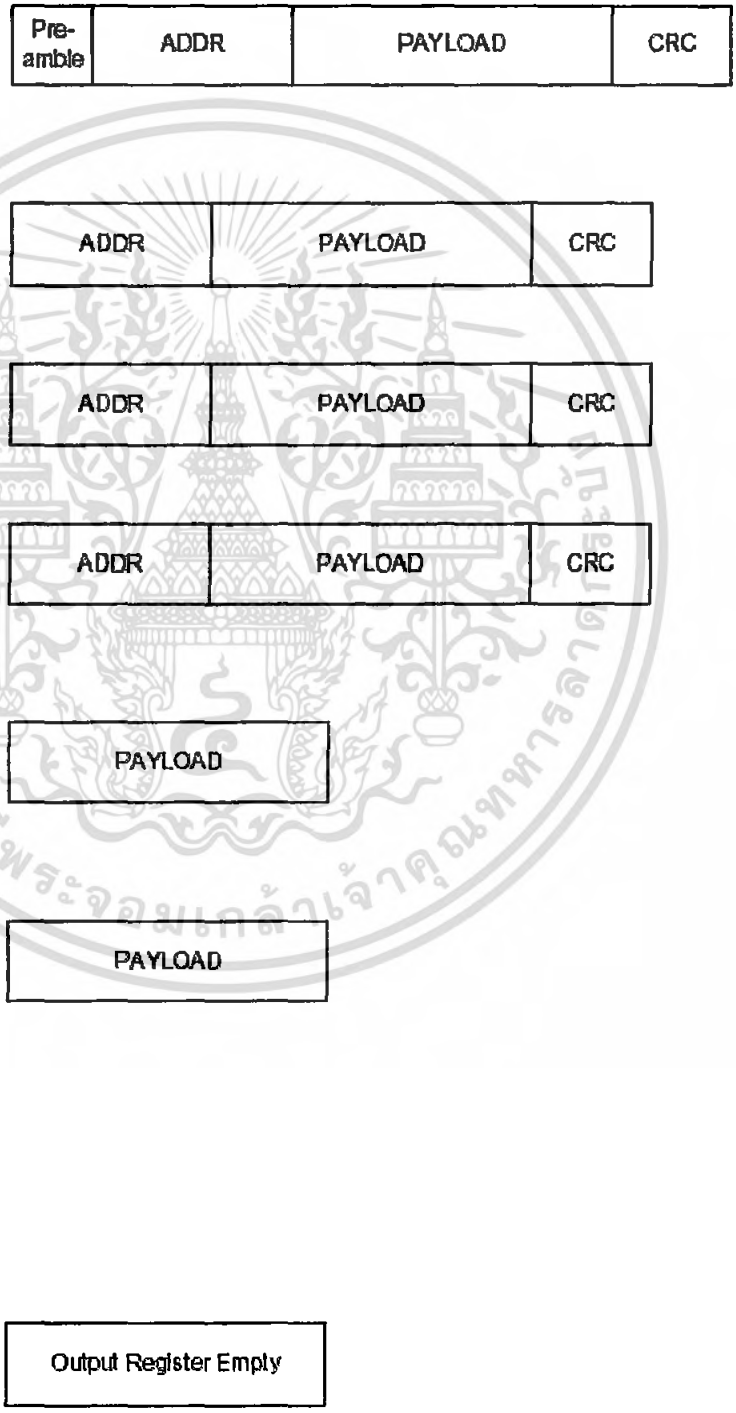


Figure 3 Flow Chart ShockBurst™ Receive of TRW-2.4G

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRW-2.4G ShockBurst™ Receive:

MCU interface pins: CE, DR1, CLK1 and DATA (one RX channel receive)

1. Correct address and size of payload of incoming RF packages are set when TRW-2.4G is configured to ShockBurst™ RX.
2. To activate RX, set CE high.
3. After 200 µs settling, TRW-2.4G is monitoring the air for incoming communication.
4. When a valid package has been received (correct address and CRC found), TRW-2.4G removes the preamble, address and CRC bits.
5. TRW-2.4G then notifies (interrupts) the MCU by setting the DR1 pin high.
6. MCU may (or may not) set the CE low to disable the RF front end (low current mode).
7. The MCU will clock out just the payload data at a suitable rate (ex. 10 kbps).
8. When all payload data is retrieved TRW-2.4G sets DR1 low again, and is ready for new incoming data package if CE is kept high during data download. If the CE was set low, a new start up sequence can begin, see Figure 12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DuoCeiver™ Simultaneous Two Channel Receive Mode

In both ShockBurst™ modes the TRW-2.4G can facilitate simultaneous reception of two parallel independent frequency channels at the maximum data rate. This means:

- TRW-2.4G can receive data from two 1 Mbps transmitters (ex: TRW-2.4G or TRW-2.4G_) 8 MHz (8 frequency channels) apart through one antenna interface.
- The output from the two data channels is fed to two separate MCU interfaces.
  - Data channel 1: CLK1, DATA, and DR1
  - Data channel 2: CLK2, DOUT2, and DR2
  - DR1 and DR2 are available only in ShockBurst™.

The TRW-2.4G DuoCeiver™ technology provides 2 separate dedicated data channels for RX and replaces the need for two, stand alone receiver systems.

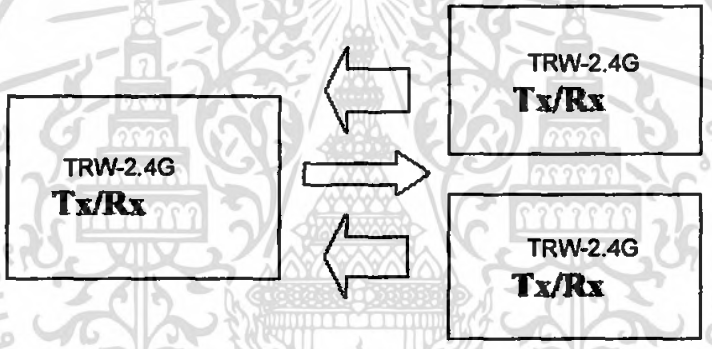


Figure 4 Simultaneous 2 channel receive on TRW-2.4G

There is one absolute requirement for using the second data channel. For the TRW-2.4G to be able to receive at the second data channel the frequency channel must be 8MHz higher than the frequency of data channel 1. The TRW-2.4G must be programmed to receive at the frequency of data channel 1. No time multiplexing is used in TRW-2.4G to fulfil this function. In direct mode the MCU must be able to handle two simultaneously incoming data packets if it is not multiplexing between the two data channels. In ShockBurst™ it is possible for the MCU to clock out one data channel at a time while data on the other data channel waits for MCU availability, without any lost data packets, and by doing so reduce the needed performance of the MCU.

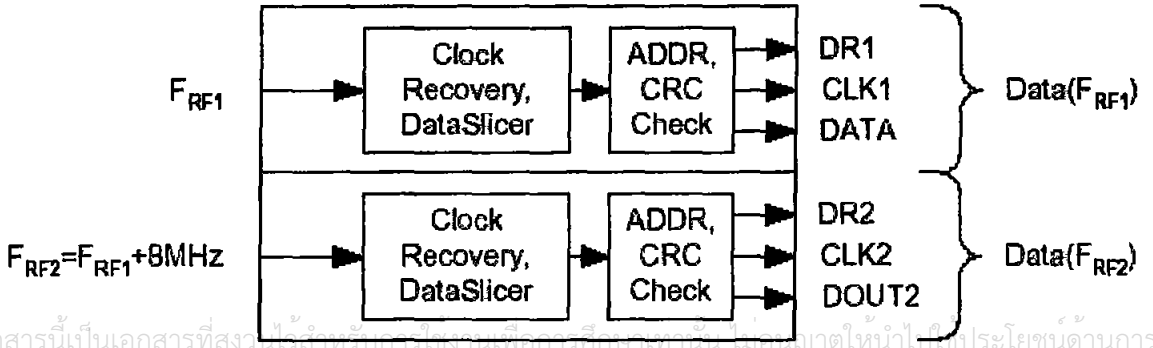


Figure 5 DuoCeiver™ with two simultaneously independent receive channels.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEVICE CONFIGURATION

All configuration of the TRW-2.4G is done via a 3-wire interface to a single configuration register. The configuration word can be up to 15 bytes long for ShockBurst™

Configuration for ShockBurst™ operation

The configuration word in ShockBurst™ enables the TRW-2.4G to handle the RF protocol. Once the protocol is completed and loaded into TRW-2.4G only one byte, bit[7:0], needs to be updated during actual operation.

The configuration blocks dedicated to ShockBurst™ is as follows:

- Payload section width: Specifies the number of payload bits in a RF package. This enables the TRW-2.4G to distinguish between payload data and the CRC bytes in a received package.
- Address width: Sets the number of bits used for address in the RF package. This enables the TRW-2.4G to distinguish between address and payload data.
- Address (RX Channel 1 and 2): Destination address for received data.
- CRC: Enables nRF2401 on-chip CRC generation and de-coding.

NOTE:
These configuration blocks, with the exception of the CRC, are dedicated for the packages that a TRW-2.4G is to receive.
In TX mode, the MCU must generate an address and a payload section that fits the configuration of the TRW-2.4G that is to receive the data.
When using the TRW-2.4G on-chip CRC feature ensure that CRC is enabled and uses the same length for both the TX and RX devices.

Figure 10 Data packet set-up

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Configuration Word overview

	Bit position	Number of bits	Name	Function
ShockBurst™ configuration	143:120	24	TEST	Reserved for testing
	119:112	8	DATA2_W	Length of data payload section RX channel 2
	111:104	8	DATA1_W	Length of data payload section RX channel 1
	103:64	40	ADDR2	Up to 5 byte address for RX channel 2
	63:24	40	ADDR1	Up to 5 byte address for RX channel 1
	23:18	6	ADDR_W	Number of address bits (both RX channels).
	17	1	CRC_L	8 or 16 bit CRC
	16	1	CRC_EN	Enable on-chip CRC generation/checking.
General device configuration	15	1	RX2_EN	Enable two channel receive mode
	14	1	CM	Communication mode (Direct or ShockBurst™)
	13	1	RFDR_SB	RF data rate (1Mbps requires 16MHz crystal)
	12:10	3	XO_F	Crystal frequency
	9:8	2	RF_PWR	RF output power
	7:1	7	RF_CH#	Frequency channel
	0	1	RXEN	RX or TX operation

Table 1 Table of configuration words.

The configuration word is shifted in MSB first on positive CLK1 edges. New configuration is enabled on the falling edge of CS.

NOTE.
On the falling edge of CS, the TRW-2.4G updates the number of bits actually shifted in during the last configuration.
Ex:
If the TRW-2.4G is to be configured for 2 channel RX in ShockBurst™, a total of 120 bits must be shifted in during the first configuration after VDD is applied.
Once the wanted protocol, modus and RF channel are set, only one bit (RXEN) is shifted in to switch between RX and TX.

Configuration Word Detailed Description

The following describes the function of the 144 bits (bit 143 = MSB) that is used to configure the TRW-2.4G

General Device Configuration: bit[15:0]
ShockBurst™ Configuration: bit[119:0]
Test Configuration: bit[143:120]

MSB	TEST						
D143	D142	D141	D140	D139	D138	D137	D136
Reserved for testing							
1	0	0	0	1	1	1	0

Default

MSB	TEST														
D135	D134	D133	D132	D131	D130	D129	D128	D127	D126	D125	D124	D123	D122	D121	D120
Reserved for testing														*Close* PLL in TX	
0	0	0	0	1	0	0	0	0	0	0	1	1	1	0	0

Default

DATA2_W							
D119	D118	D117	D116	D115	D114	D113	D112
Data width channel#2 in # of bits excluding addr/crc							
0	0	1	0	0	0	0	0

Default

DATA1_W							
D111	D110	D109	D108	D107	D106	D105	D104
Data width channel#1 in # of bits excluding addr/crc							
0	0	1	0	0	0	0	0

Default

ADDR2											
D103	D102	D101	....	D71	D70	D69	D68	D67	D66	D65	D64
Channel#2 Address RX (up to 40bit)											
0	0	0	...	1	1	1	0	0	1	1	1

Default

ADDR1											
D63	D62	D61	....	D31	D30	D29	D28	D27	D26	D25	D24
Channel#1 Address RX (up to 40bit)											
0	0	0	...	1	1	1	0	0	1	1	1

Default

ADDR_W						
D23	D22	D21	D20	D19	D18	
Address width in # of bits (both channels)						
0	0	1	0	0	0	Default

CRC		
D17	D16	
CRC Mode 1 – 16bit, 0 – 8bit	CRC 1 – enable; 0 – disable	
0	1	Default

RF-Programming															LSB	
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	
Two Ch.	BUF	OD	XO			RF Power		Channel selection							RXEN	
0	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	Default

Table 2Configuration data word

The MSB bit should be loaded first into the configuration register.
Default configuration word: h8E08.1C20.2000.0000.00E7.0000.0000.E721.0F04.

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ShockBurst™ configuration:

The section B[119:16] contains the segments of the configuration register dedicated to ShockBurst™ operational protocol. After VDD is turned on ShockBurst™ configuration is done once and remains set whilst VDD is present. During operation only the first byte for frequency channel and RX/TX switching need to be changed.

PLL_CTRL

PLL_CTRL		
D121	**D120**	**PLL**
0	0	Open TX/Closed RX
0	1	Open TX/Open RX
1	0	Closed TX/Closed RX
1	1	Closed TX/Open RX

Table 10 PLL setting.

Bit 121-120:

PLL_CTRL: Controls the setting of the PLL for test purposes. With closed PLL in TX no deviation will be present.

DATAx_W

DATA2_W							
119	118	117	116	115	114	113	112

DATA1_W							
111	110	109	108	107	106	105	104

Table 4 Number of bits in payload.

Bit 119 – 112:

DATA2_W: Length of RF package payload section for receive-channel 2.

Bit 111 – 104:

DATA1_W: Length of RF package payload section for receive-channel 1.

NOTE:

The total number of bits in a ShockBurst™ RF package may not exceed 256! Maximum length of payload section is hence given by:

$$DATAx_W(bits) = 256 - ADDR_W - CRC$$

Where:

ADDR_W: length of RX address set in configuration word B[23:18]

CRC: check sum, 8 or 16 bits set in configuration word B[17]

PRE: preamble, 4 or 8 bits are automatically included

Shorter address and CRC leaves more room for payload data in each package.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ADDRx

ADDR2											
103	102	101	....	71	70	69	68	67	66	65	64

ADDR1											
63	62	61	....	31	30	29	28	27	26	25	24

Table 5 : Address of receiver #2 and receiver #1.

Bit 103 – 64:

ADDR2: Receiver address channel 2, up to 40 bit.

Bit 63 – 24: ADDR1

ADDR1: Receiver address channel 1, up to 40 bit.

NOTE!

Bits in ADDRx exceeding the address width set in ADDR_W are redundant and can be set to logic 0.

ADDR_W & CRC

						ADDR_W		CRC_L	CRC_EN
23	22	21	20	19	18			17	16

Table 6 Number of bits reserved for RX address + CRC setting.

Bit 23 – 18:

ADDR_W: Number of bits reserved for RX address in ShockBurst™ packages.

NOTE:

Maximum number of address bits is 40 (5 bytes). Values over 40 in ADDR_W are not valid.

Bit 17:

CRC_L: CRC length to be calculated by TRW-2.4G in ShockBurst™.
Logic 0: 8 bit CRC
Logic 1: 16 bit CRC

Bit: 16:

CRC_EN: Enables on-chip CRC generation (TX) and verification (RX).
Logic 0: On-chip CRC generation/checking disabled
Logic 1: On-chip CRC generation/checking enabled

NOTE:

An 8 bit CRC will increase the number of payload bits possible in each ShockBurst™ data packet, but will also reduce the system integrity.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของลิขสิทธิ์ทุกครั้งที่มีการนำไปใช้

General device configuration:

This section of the configuration word handles RF and device related parameters.

Modes:

RX2_EN	CM	RFDR_SB	XO_F			RF_PWR	
15	14	13	12	11	10	9	8

Table 7 RF operational settings.

Bit 15:

RX2_EN:

Logic 0: One channel receive
Logic 1: Two channels receive

NOTE:

In two channels receive, the TRW-2.4G receives on two, separate frequency channels simultaneously. The frequency of receive channel 1 is set in the configuration word B[7-1], receive channel 2 is always 8 channels (8 MHz) above receive channel 1.

Bit 14:

Communication Mode:

Logic 1: nRF2401 operates in ShockBurst™ mode

Bit 13:

RF Data Rate:

Logic 0: 250 kbps
Logic 1: 1 Mbps

NOTE:

Utilizing 250 kbps instead of 1Mbps will improve the receiver sensitivity by 10 dB. 1Mbps requires 16MHz crystal.

Bit 12-10:

D12	D11	D10
0	1	1

Table 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bit 9-8:

RF_PWR: Sets TRW-2.4G RF output power in transmit mode:

RF OUTPUT POWER		
D9	D8	P [dBm]
0	0	-20
0	1	-10
1	0	-5
1	1	0

Table 9 RF output power setting.

RF channel & direction

RF_CH#							RXEN
7	6	5	4	3	2	1	0

Table 10 Frequency channel + RX / TX setting.

Bit 7 – 1:

RF_CH#: Sets the frequency channel the nRF2401 operates on.

The channel frequency in ***transmit*** is given by:

$$Channel_{RF} = 2400\ MHz + RF_CH\# \cdot 1.0\ MHz$$

RF_CH #: between 2400MHz and 2527MHz may be set.

The channel frequency in ***data channel 1*** is given by:

$$Channel_{RF} = 2400\ MHz + RF_CH\# \cdot 1.0\ MHz$$ (Receive at PIN#8)

RF_CH #: between 2400MHz and 2524MHz may be set.

NOTE:
The channels above 83 can only be utilized in certain territories (ex: Japan)

The channel frequency in ***data channel 2*** is given by:

$$Channel_{RF} = 2400\ MHz + RF_CH\# \cdot 1.0\ MHz$$ +8MHz (Receive at PIN#4)

RF_CH #: between 2408MHz and 2524MHz may be set.

Bit 0:

Set active mode:

Logic 0: transmit mode
Logic 1: receive mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA PACKAGE DESCRIPTION

PRE-AMBLE	ADDRESS	PAYLOAD	CRC

Figure 7 Data Package Diagram

The data packet for both ShockBurst™ mode and direct mode communication is divided into 4 sections. These are:

1. PREAMBLE	· The preamble field is required in ShockBurst.
2. ADDRESS	· The address field is required in ShockBurst. mode. · 8 to 40 bits length. · Address automatically removed from received packet in ShockBurst.mode
3. PAYLOAD	· The data to be transmitted · In Shock-Burst mode payload size is 256 bits minus the following:(Address: 8 to 40 bits. + CRC 8 or 16 bits).
4. CRC	· 8 or 16 bits length · The CRC is stripped from the received output data.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IMPORTANT TIMING DATA

The following timing applies for operation of nRF2401 in TRW-2.4G

TRW-2.4G Timing Information

nRF2401 timing	Max.	Min.	Name
VDD OFF → ST_BY mode	3ms		Tpd2sby
VDD OFF → Active mode (RX/TX)	3ms		Tpd2a
ST_BY → TX ShockBurst™	195µs		Tsby2txSB
ST_BY → TX Direct Mode	202µs		Tsby2txDM
ST_BY → RX mode	202µs		Tsby2rx
Minimum delay from CS to data.		5µs	Tcs2data
Minimum delay from CE to data.		5µs	Tce2data
Minimum delay from DR1/2 to clk.		50ns	Tdr2clk
Maximum delay from clk to data.	50ns		Tclk2data
Delay between edges		50ns	Td
Setup time		500ns	Ts
Hold time		500ns	Th
Delay to finish internal GFSK data		1/data rate	Tfd
Minimum input clock high		500ns	Thmin
Set-up of data in Direct Mode	50ns		Tsdm
Minimum clock high in Direct Mode		300ns	Thdm
Minimum clock low in Direct Mode		230ns	Tldm

Table 11 Switching times for nRF2401 in TRW-2.4G

When the nRF2401 is in power down it must always settle in stand-by (Tpd2sby) before it can enter configuration or one of the active modes.

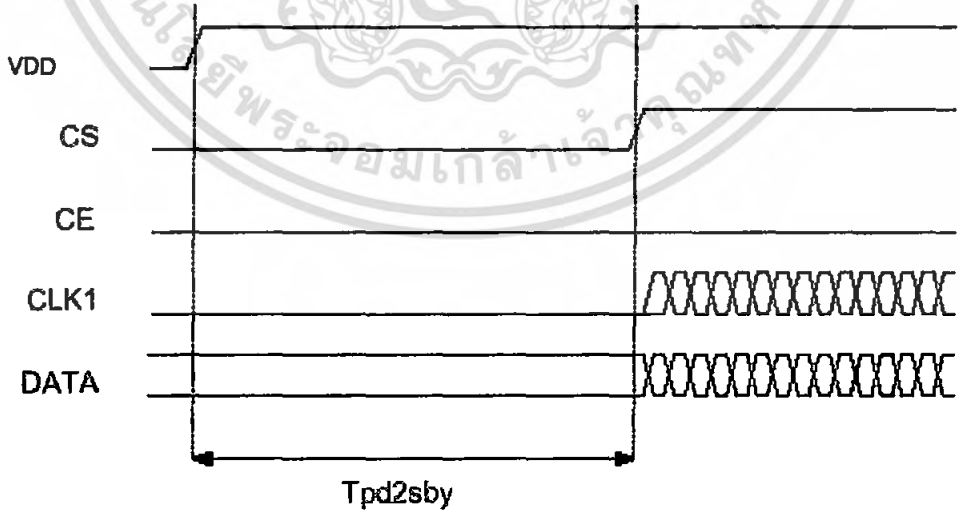


Figure 8 Timing diagram for power down (or VDD off) to stand by mode for TRW-2.4G

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

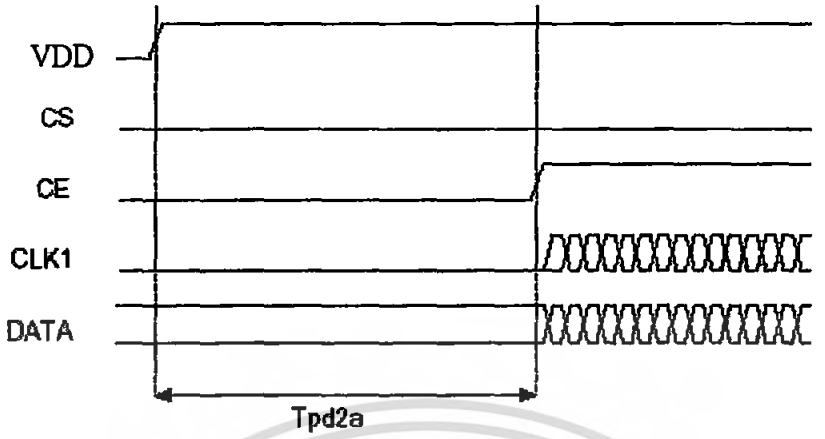


Figure 9 VDD off to active mode

Note that the configuration word will be lost when VDD is turned off and that the device then must be configured before going to one of the active modes. If the device is configured one can go directly from power down to the wanted active mode.

Note:

CE and CS may not be high at the same time. Setting one or the other decides whether configuration or active mode is entered.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Configuration mode timing

When one or more of the bits in the configuration word needs to be changed the following timing apply.

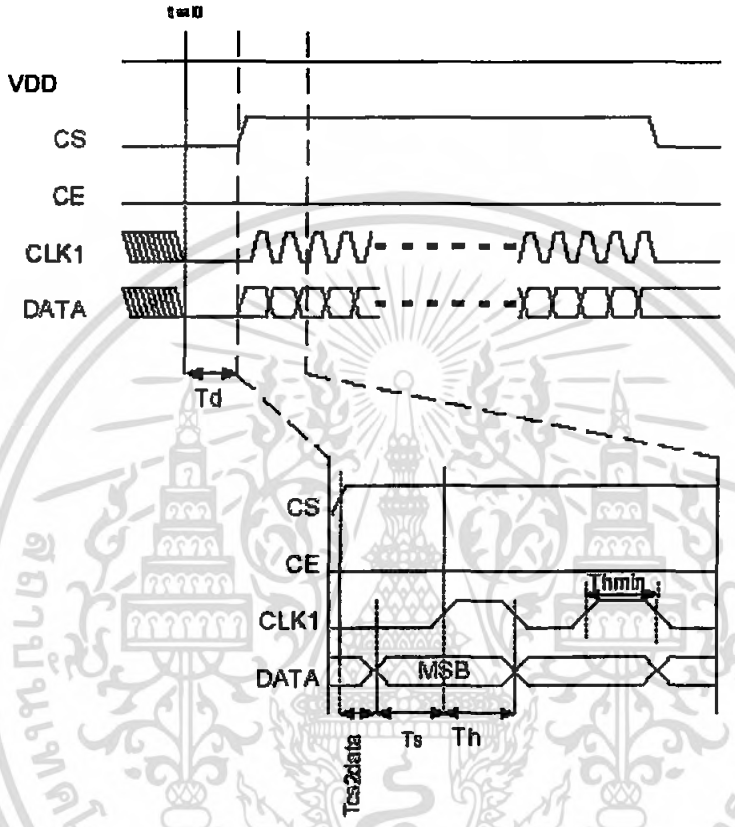


Figure 10 Timing diagram for configuration of TRW-2.4G

If configuration mode is entered from power down, CS can be set high after Tpd2sby as shown in Figure 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ShockBurst™ Mode timing

ShockBurst™ TX:

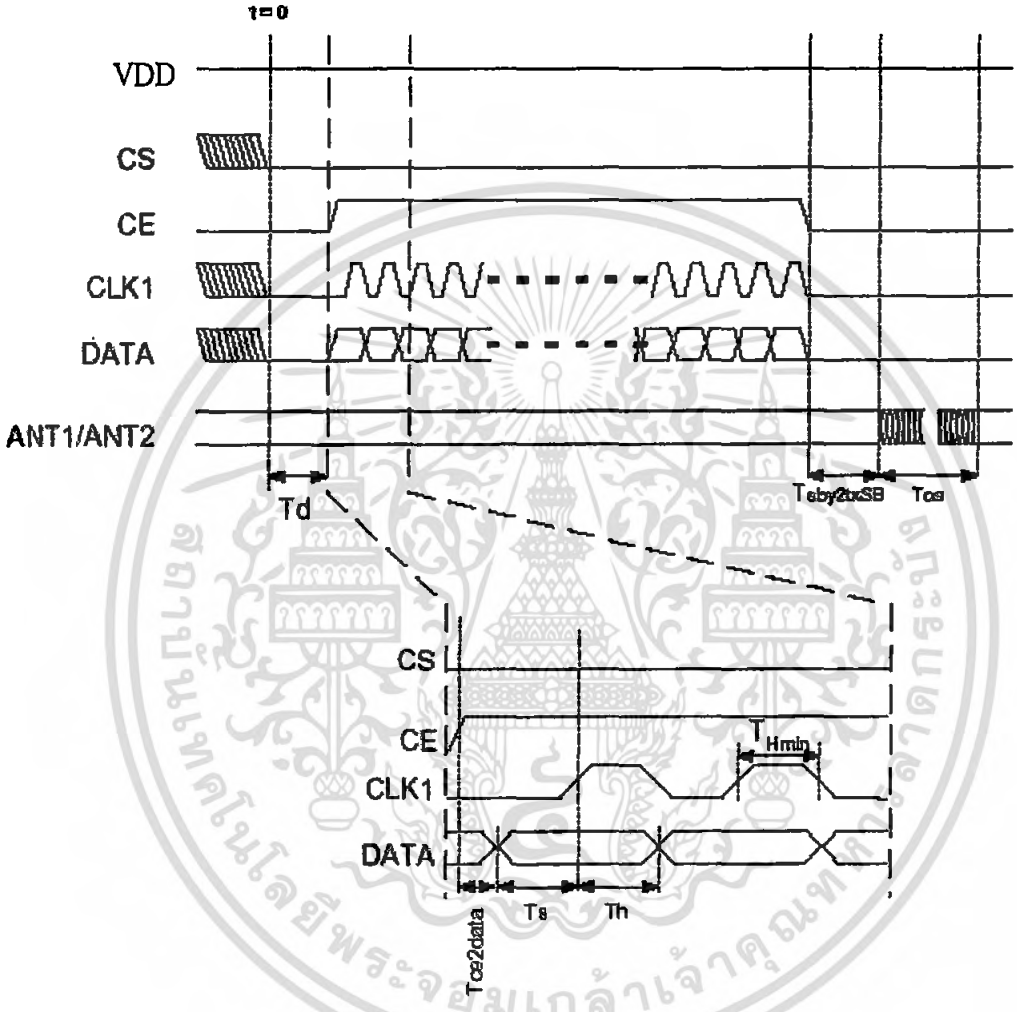


Figure 11 Timing of ShockBurst™ in TX

The package length and the data rate give the delay Toa (time on air), as shown in the equation.

$T_{oa} = 1/datarate \cdot (\#databits + 1)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ShockBurst™ RX:

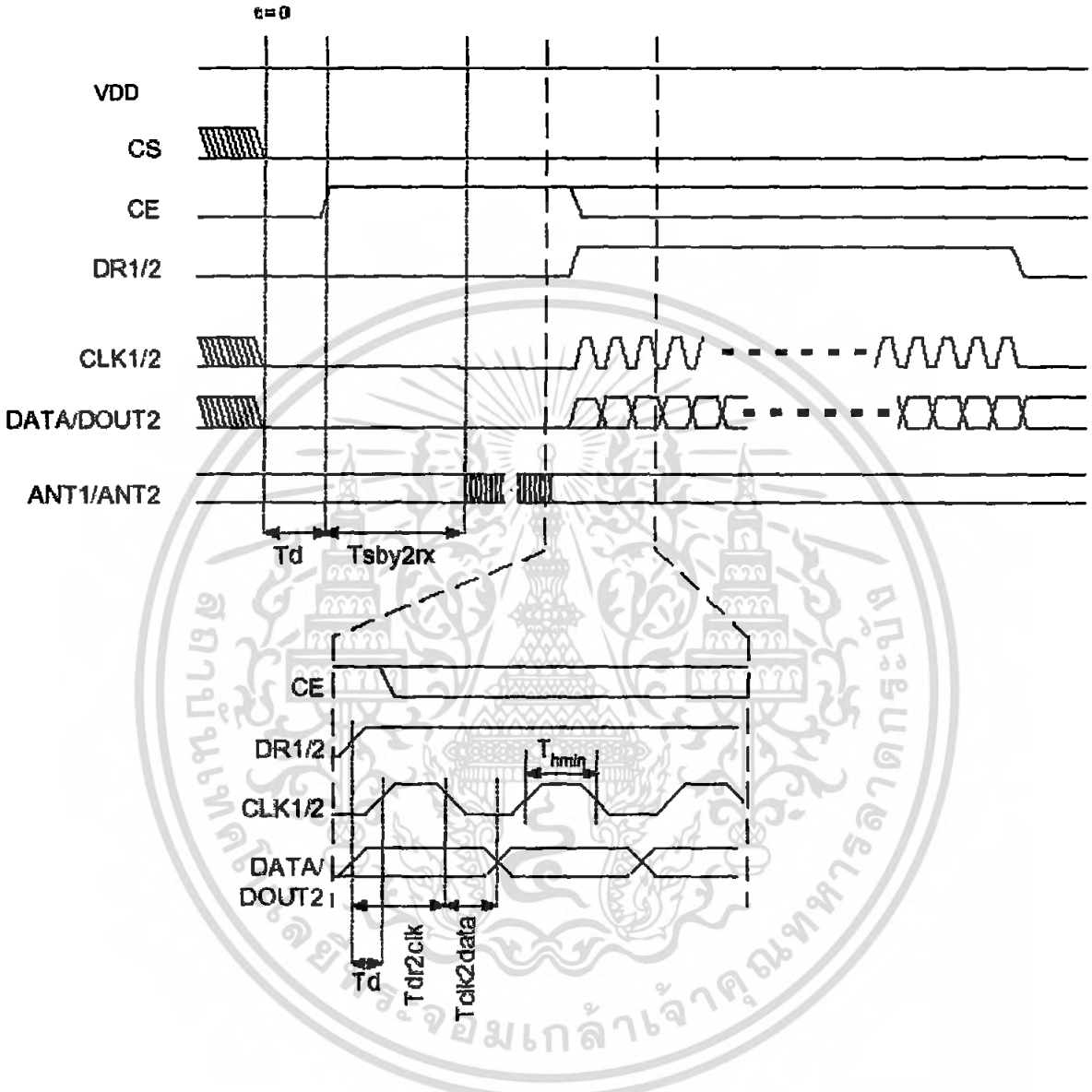


Figure 12 Timing of ShockBurst™ in RX

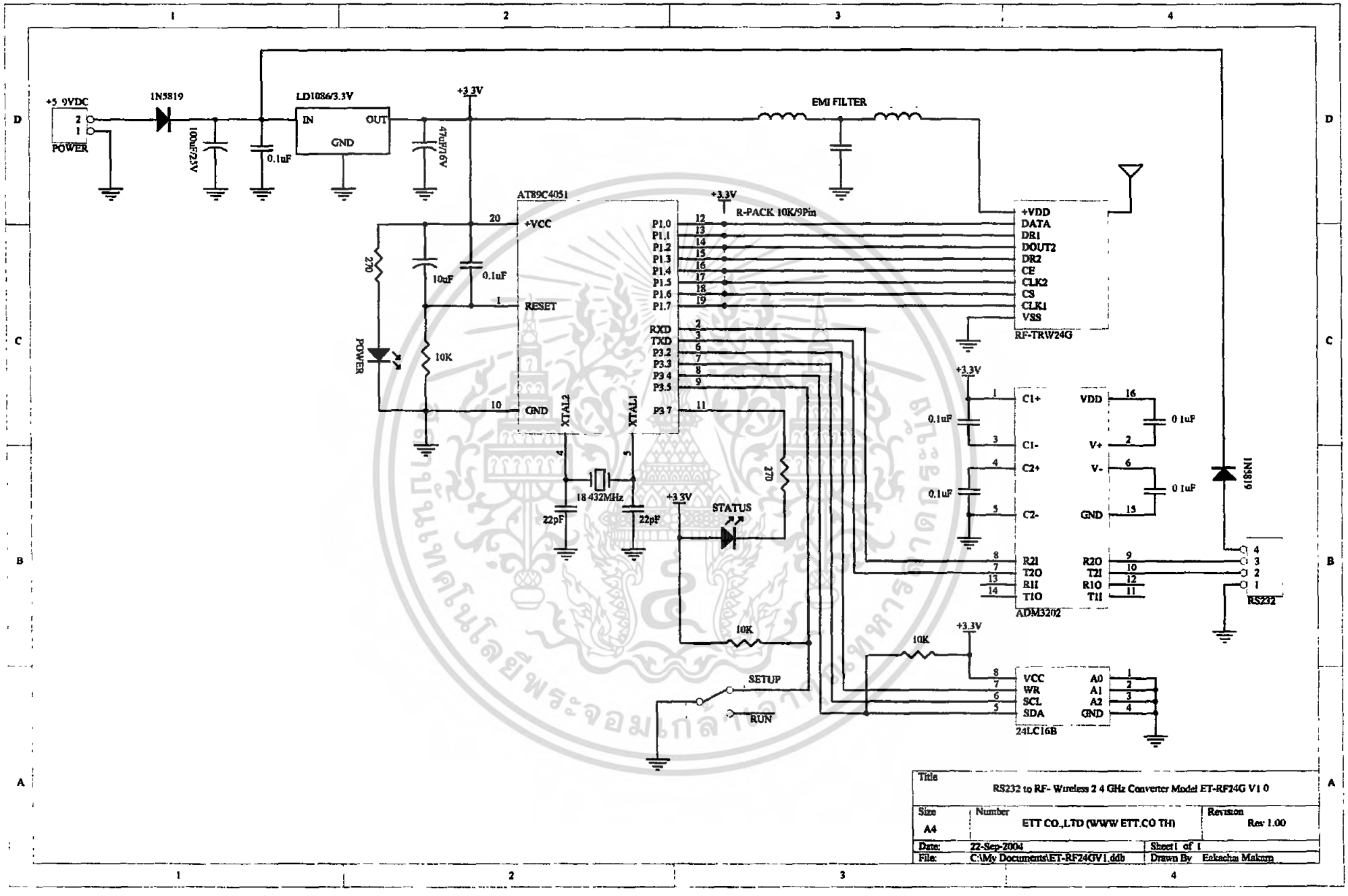
The CE may be kept high during downloading of data, but the cost is higher current consumption (18mA) and the benefit is no start-up time (200μs) after the DR1 goes low.

Output Power adjustment

Power setting bits of configuring word	RF output power	DC current consumption
11	0 dBm ±3dB	13.0 mA
10	-5 dBm ±3dB	10.5 mA
01	-10 dBm ±3dB	9.4 mA
00	-20 dBm ±3dB	8.8 mA

Conditions: VDD = 3.0V, VSS = 0V, T_A = 27°C, Load impedance = 400 Ω.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


+5 9VDC
POWER
1N5819
1000uF/25V
0.1uF
LD1086/3.3V
IN
OUT
GND
+3.3V
47uF/16V
EMI FILTER
AT89C4051
+VCC
RESET
GND
XTAL2
XTAL1
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
RXD
TXD
P3.2
P3.3
P3.4
P3.5
P3.7
R-PACK 10K/9Pin
270
10uF
0.1uF
10K
POWER
18.432MHz
22pF
22pF
STATUS
270
10K
SETUP
RUN
+VDD
DATA
DR1
DOUT2
DR2
CE
CLK2
CS
CLK1
VSS
RF-TRW24G
C1+
C1-
C2+
C2-
VDD
V+
V-
GND
0.1uF
0.1uF
0.1uF
0.1uF
R2I
T2O
R1I
T1O
R2O
T2I
R1O
T1I
ADM3202
1N5819
RS232
10K
VCC
WR
SCL
SDA
A0
A1
A2
GND
24LC16B
Title
RS232 to RF- Wireless 2 4 GHz Converter Model ET-RF24G V1 0
Size
A4
Number
ETT CO.,LTD (WWW ETT.CO TH)
Revision
Rev 1.00
Date:
22-Sep-2004
Sheet 1 of 1
File:
C:\My Documents\ET-RF24GV1.ddb
Drawn By
Enkachai Maksun

Features

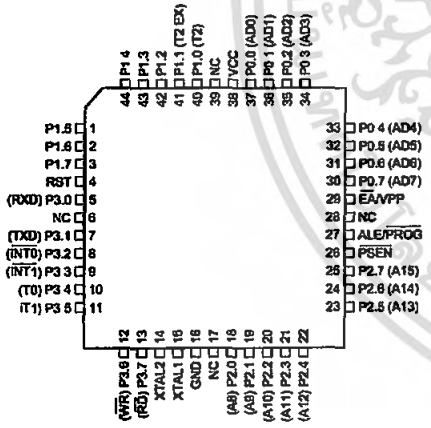
- Compatible with MCS-51™ Products
- 8K Bytes of In-System Reprogrammable Flash Memory
- Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Eight Interrupt Sources
- Programmable Serial Channel
- Low-power Idle and Power-down Modes

Description

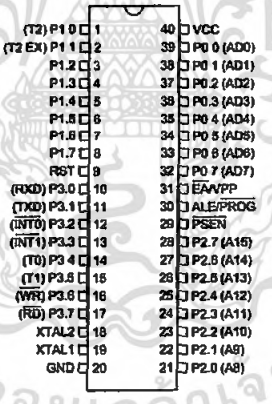
The AT89C52 is a low-power, high-performance CMOS 8-bit microcomputer with 8K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 and 80C52 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C52 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

Pin Configurations

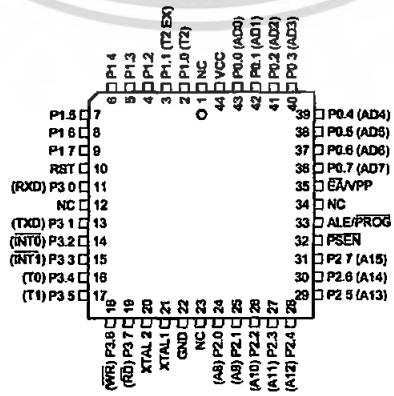
PQFP/TQFP



PDIP



PLCC



8-bit Microcontroller with 8K Bytes Flash

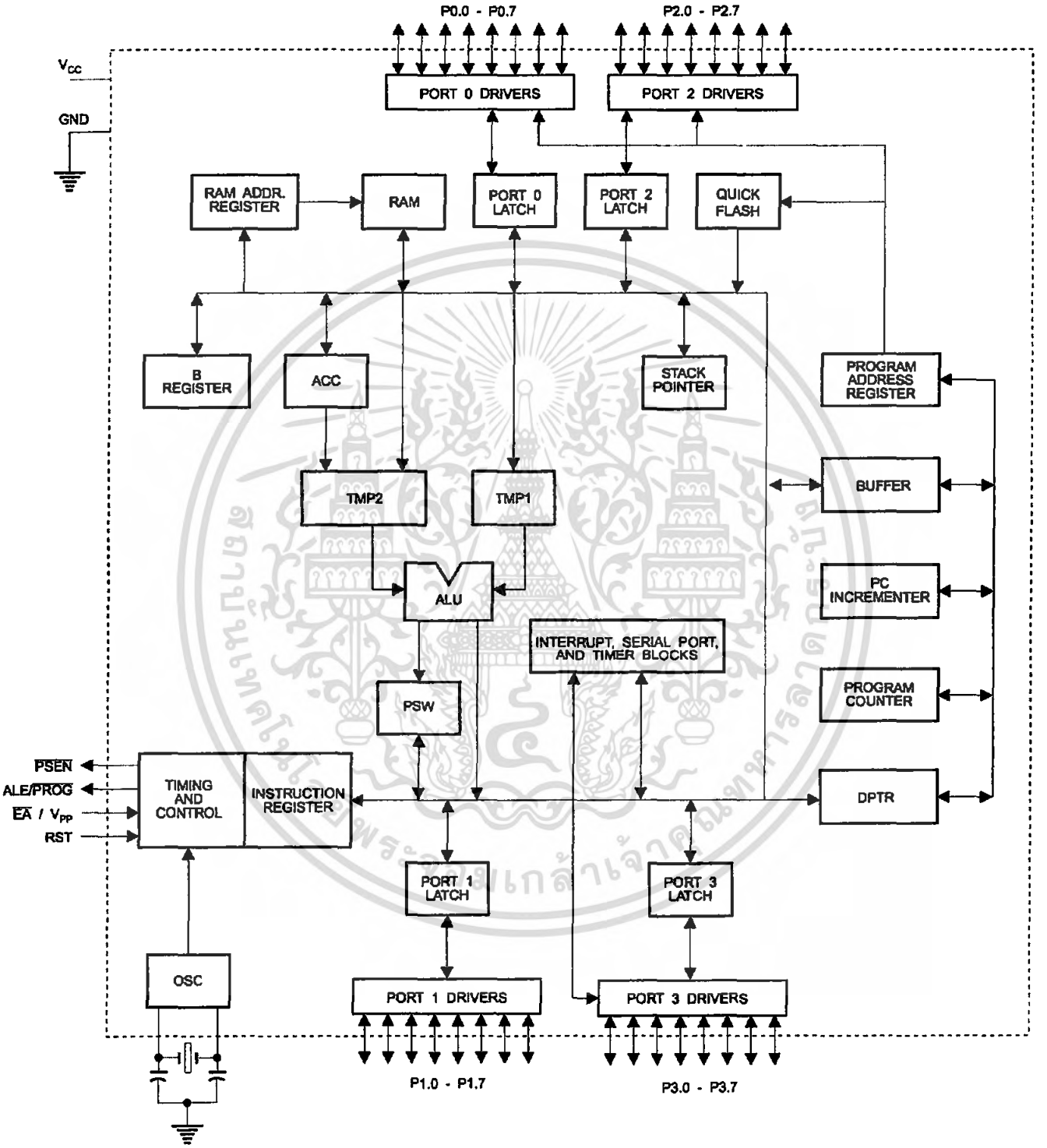
AT89C52

Not Recommended for New Designs. Use AT89S52.

Rev. 0313H–02/00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The AT89C52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full-duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89C52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next hardware reset.

Pin Description

VCC

Supply voltage.

GND

Ground.

Port 0

Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pullups.

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1

Port 1 is an 8-bit bi-directional I/O port with internal pullups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

In addition, P1.0 and P1.1 can be configured to be the timer/counter 2 external count input (P1.0/T2) and the timer/counter 2 trigger input (P1.1/T2EX), respectively, as shown in the following table.

Port 1 also receives the low-order address bytes during Flash programming and verification.

Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)

Port 2

Port 2 is an 8-bit bi-directional I/O port with internal pullups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pullups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51, as shown in the following table.

Port 3 also receives some control signals for Flash programming and verification.

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/$\overline{PROG}$

Address Latch Enable is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{PROG}$) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

$\overline{\text{PSEN}}$

Program Store Enable is the read strobe to external program memory.

When the AT89C52 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory.

$\overline{\text{EA}}$/VPP

External Access Enable. $\overline{\text{EA}}$ must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, $\overline{\text{EA}}$ will be internally latched on reset.

$\overline{\text{EA}}$ should be strapped to V_{CC} for internal program executions.

This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming when 12-volt programming is selected.

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Table 1. AT89C52 SFR Map and Reset Values

0F8H								0FFH
0F0H	B 00000000							0F7H
0E8H								0EFH
0E0H	ACC 00000000							0E7H
0D8H								0DFH
0D0H	PSW 00000000							0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000		0CFH
0C0H								0C7H
0B8H	IP XX000000							0BFH
0B0H	P3 11111111							0B7H
0A8H	IE 0X000000							0AFH
0A0H	P2 11111111							0A7H
98H	SCON 00000000	SBUF XXXXXXXX						9FH
90H	P1 11111111							97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000		8FH
80H	P0 11111111	SP 00000111	DPL 00000000	DPH 00000000				PCON 0XXX0000 87H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 1.

Note that not all of the addresses are occupied, and unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

User software should not write 1s to these unlisted locations, since they may be used in future products to invoke new features. In that case, the reset or inactive values of the new bits will always be 0.

Timer 2 Registers Control and status bits are contained in registers T2CON (shown in Table 2) and T2MOD (shown in Table 4) for Timer 2. The register pair (RCAP2H, RCAP2L) are the Capture/Reload registers for Timer 2 in 16-bit capture mode or 16-bit auto-reload mode.

Interrupt Registers The individual interrupt enable bits are in the IE register. Two priorities can be set for each of the six interrupt sources in the IP register.r

Table 2. T2CON – Timer/Counter 2 Control Registe

T2CON Address = 0C8H — Reset Value = 0000 0000B

Bit Addressable

Bit	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
	7	6	5	4	3	2	1	0

Symbol	Function
TF2	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
EXF2	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).
RCLK	Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
TCLK	Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
C/$\overline{T2}$	Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).
CP/$\overline{RL2}$	Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

Data Memory

The AT89C52 implements 256 bytes of on-chip RAM. The upper 128 bytes occupy a parallel address space to the Special Function Registers. That means the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space.

When an instruction accesses an internal location above address 7FH, the address mode used in the instruction specifies whether the CPU accesses the upper 128 bytes of RAM or the SFR space. Instructions that use direct addressing access SFR space.

For example, the following direct addressing instruction accesses the SFR at location 0A0H (which is P2).

```
MOV 0A0H, #data
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instructions that use indirect addressing access the upper 128 bytes of RAM. For example, the following indirect addressing instruction, where R0 contains 0A0H, accesses the data byte at address 0A0H, rather than P2 (whose address is 0A0H).

```
MOV @R0, #data
```

Note that stack operations are examples of indirect addressing, so the upper 128 bytes of data RAM are available as stack space.

Timer 0 and 1

Timer 0 and Timer 1 in the AT89C52 operate the same way as Timer 0 and Timer 1 in the AT89C51.

Timer 2

Timer 2 is a 16-bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ in the SFR T2CON (shown in Table 2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 3.

Timer 2 consists of two 8-bit registers, TH2 and TL2. In the Timer function, the TL2 register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

Table 3. Timer 2 Operating Modes

RCLK +TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

In the Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. To ensure that a given level is sampled at least once before it changes, the level should be held for at least one full machine cycle.

Capture Mode

In the capture mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16-bit timer or counter which upon overflow sets bit TF2 in T2CON. This bit can then be used to generate an interrupt. If EXEN2 = 1, Timer 2 performs the same operation, but a 1-to-0 transition at external input T2EX also causes the current value in TH2 and TL2 to be captured into RCAP2H and RCAP2L, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set. The EXF2 bit, like TF2, can generate an interrupt. The capture mode is illustrated in Figure 1.

Auto-reload (Up or Down Counter)

Timer 2 can be programmed to count up or down when configured in its 16-bit auto-reload mode. This feature is invoked by the DCEN (Down Counter Enable) bit located in the SFR T2MOD (see Table 4). Upon reset, the DCEN bit is set to 0 so that timer 2 will default to count up. When DCEN is set, Timer 2 can count up or down, depending on the value of the T2EX pin.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 1. Timer in Capture Mode

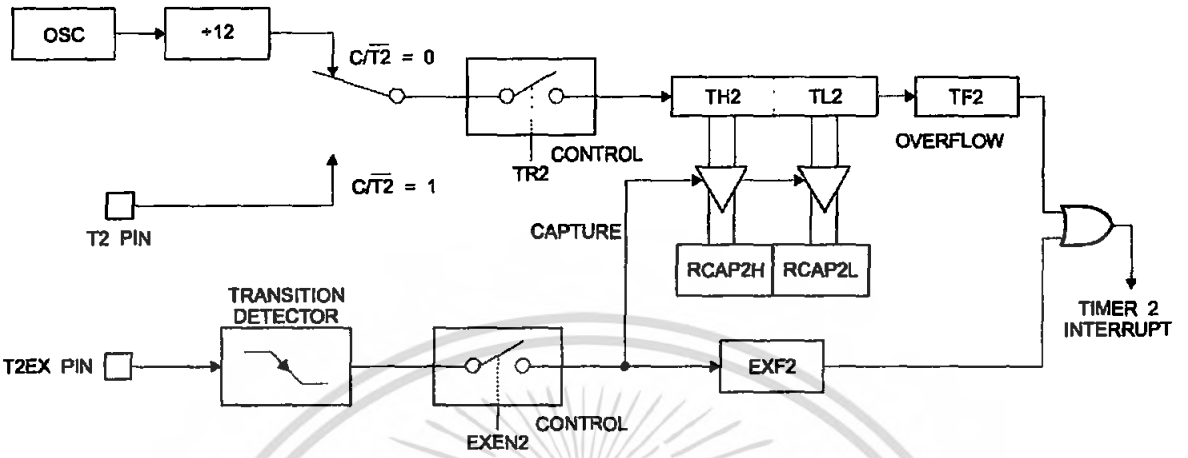


Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in Timer in Capture ModeRCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at external input T2EX. This transition also sets the EXF2 bit. Both the TF2 and EXF2 bits can generate an interrupt if enabled.

Setting the DCEN bit enables Timer 2 to count up or down, as shown in Figure 3. In this mode, the T2EX pin controls the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0FFFFH and set the TF2 bit. This overflow also causes the 16-bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

A logic 0 at T2EX makes Timer 2 count down. The timer underflows when TH2 and TL2 equal the values stored in RCAP2H and RCAP2L. The underflow sets the TF2 bit and causes 0FFFFH to be reloaded into the timer registers.

The EXF2 bit toggles whenever Timer 2 overflows or underflows and can be used as a 17th bit of resolution. In this operating mode, EXF2 does not flag an interrupt.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 2. Timer 2 Auto Reload Mode (DCEN = 0)

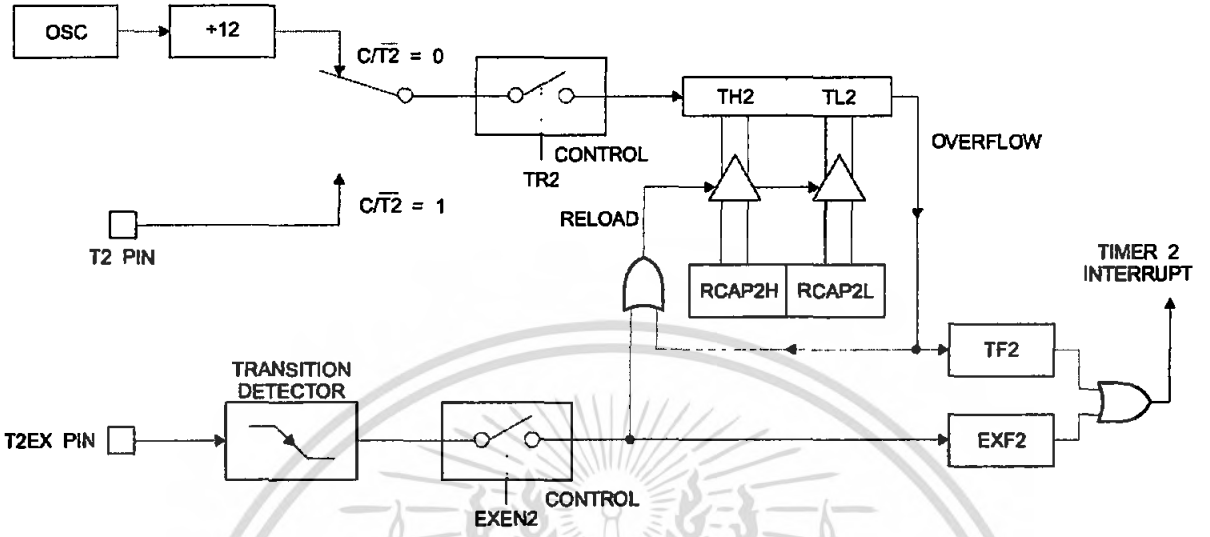


Table 4. T2MOD – Timer 2 Mode Control Register

T2MOD Address = 0C9H — Reset Value = XXXX XX00B

Not Bit Addressable

Bit	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	T2OE	DCEN

Symbol	Function
–	Not implemented, reserved for future
T2OE	Timer 2 Output Enable bit.
DCEN	When set, this bit allows Timer 2 to be configured as an up/down counter.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 3. Timer 2 Auto Reload Mode (DCEN = 1)

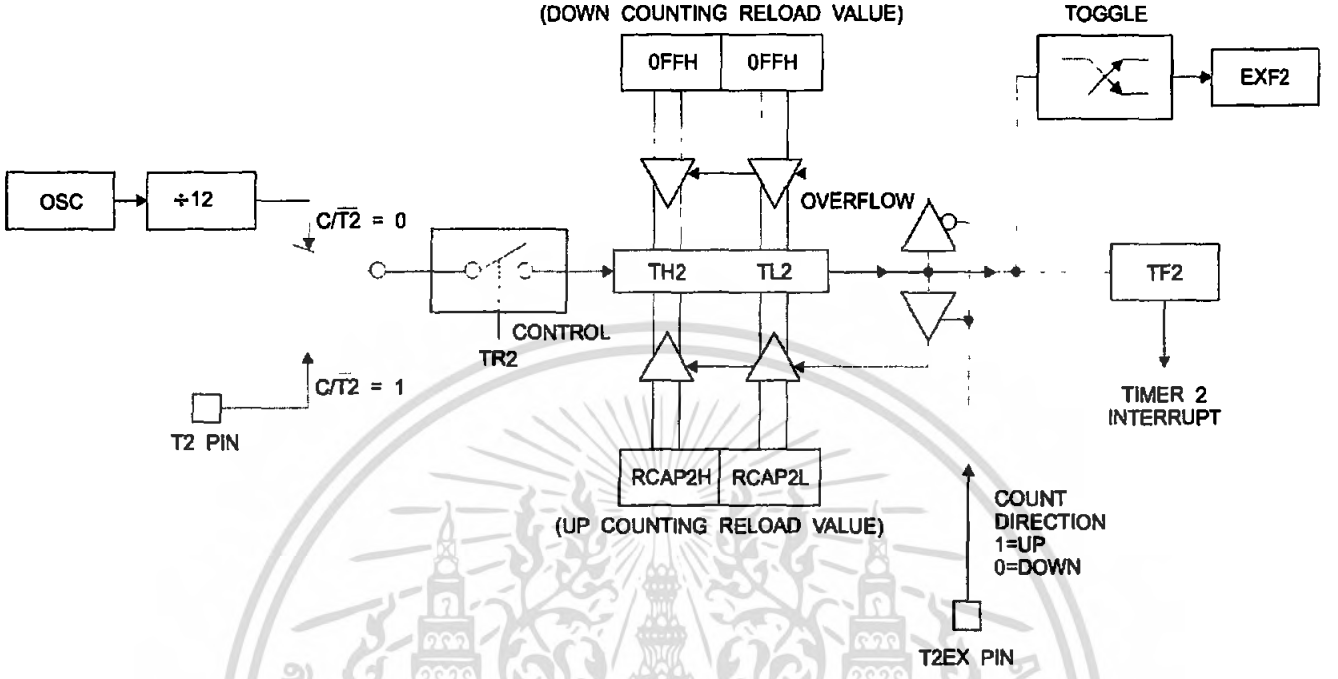
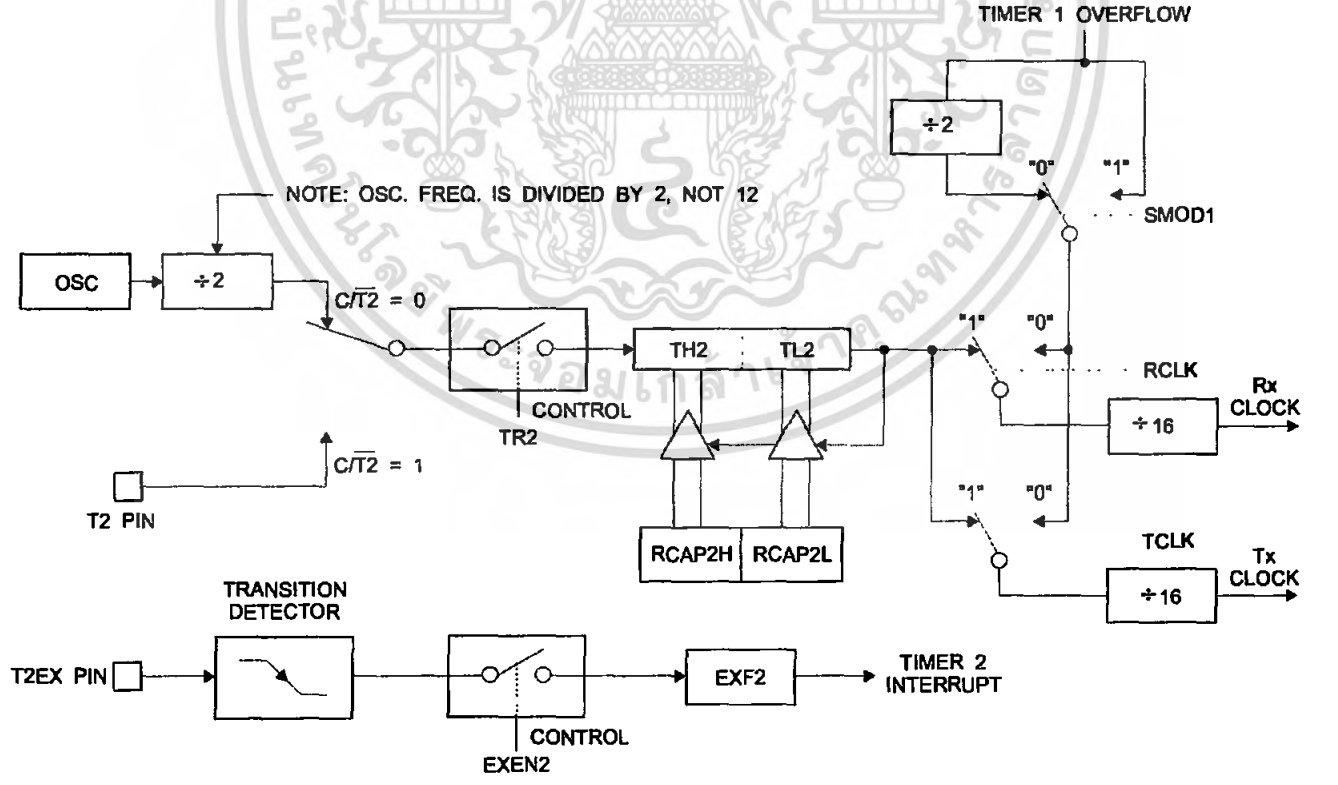


Figure 4. Timer 2 in Baud Rate Generator Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Baud Rate Generator

Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (Table 2). Note that the baud rates for transmit and receive can be different if Timer 2 is used for the receiver or transmitter and Timer 1 is used for the other function. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 4.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate according to the following equation.

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The Timer can be configured for either timer or counter operation. In most applications, it is configured for timer operation (CP/$\overline{T2}$ = 0). The timer operation is different for Timer 2 when it is used as a baud rate generator. Normally, as a timer, it increments every machine cycle (at 1/12 the oscillator frequency). As a baud rate generator, however, it increments every state time (at 1/2 the oscillator frequency). The baud rate formula is given below.

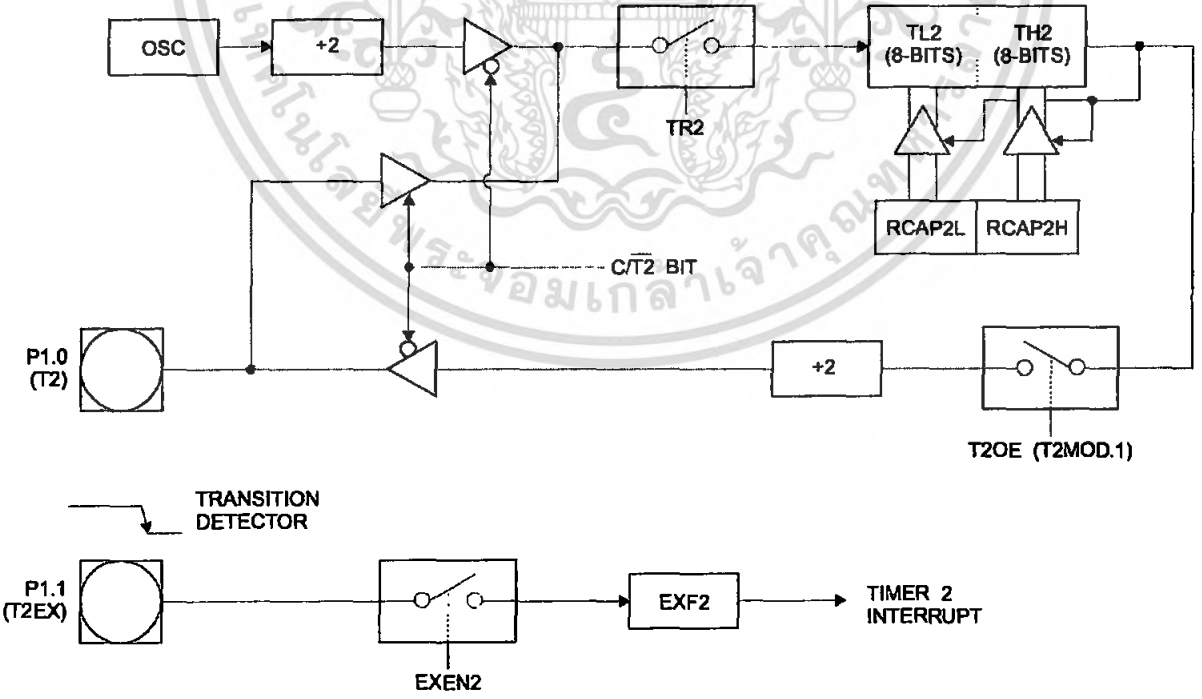
$$\frac{\text{Modes 1 and 3}}{\text{Baud Rate}} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

where (RCAP2H, RCAP2L) is the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK or TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2 and will not generate an interrupt. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt.

Note that when Timer 2 is running (TR2 = 1) as a timer in the baud rate generator mode, TH2 or TL2 should not be read from or written to. Under these conditions, the Timer is incremented every state time, and the results of a read or write may not be accurate. The RCAP2 registers may be read but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Figure 5. Timer 2 in Clock-out Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programmable Clock Out

A 50% duty cycle clock can be programmed to come out on P1.0, as shown in Figure 5. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed to input the external clock for Timer/Counter 2 or to output a 50% duty cycle clock ranging from 61 Hz to 4 MHz at a 16 MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/$\overline{T2}$ (T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

The clock-out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L), as shown in the following equation.

$$\text{Clock-Out Frequency} = \frac{\text{Oscillator Fequency}}{4 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

In the clock-out mode, Timer 2 roll-overs will not generate an interrupt. This behavior is similar to when Timer 2 is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and clock-out frequencies cannot be determined independently from one another since they both use RCAP2H and RCAP2L.

UART

The UART in the AT89C52 operates the same way as the UART in the AT89C51.

Interrupts

The AT89C52 has a total of six interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), three timer interrupts (Timers 0, 1, and 2), and the serial port interrupt. These interrupts are all shown in Figure 6.

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE. IE also contains a global disable bit, EA, which disables all interrupts at once.

Note that Table shows that bit position IE.6 is unimplemented. In the AT89C51, bit position IE.5 is also unimplemented. User software should not write 1s to these bit positions, since they may be used in future AT89 products.

Timer 2 interrupt is generated by the logical OR of bits TF2 and EXF2 in register T2CON. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, and that bit will have to be cleared in software.

The Timer 0 and Timer 1 flags, TF0 and TF1, are set at S5P2 of the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle. However, the Timer 2 flag, TF2, is set at S2P2 and is polled in the same cycle in which the timer overflows.

Table 5. Interrupt Enable (IE) Register

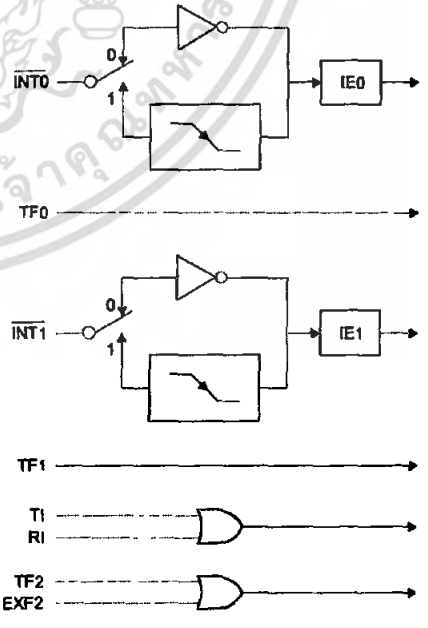
(MSB)							(LSB)
EA	–	ET2	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
–	IE.6	Reserved.
ET2	IE.5	Timer 2 interrupt enable bit.
ES	IE.4	Serial Port interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.
User software should never write 1s to unimplemented bits, because they may be used in future AT89 products.		

Figure 6. Interrupt Sources



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Oscillator Characteristics

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 7. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 8. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

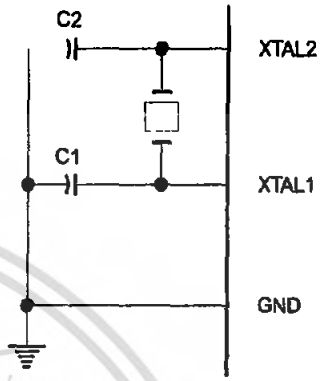
In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

Note that when idle mode is terminated by a hardware reset, the device normally resumes program execution from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when idle mode is terminated by a reset, the instruction following the one that invokes idle mode should not write to a port pin or to external memory.

Power-down Mode

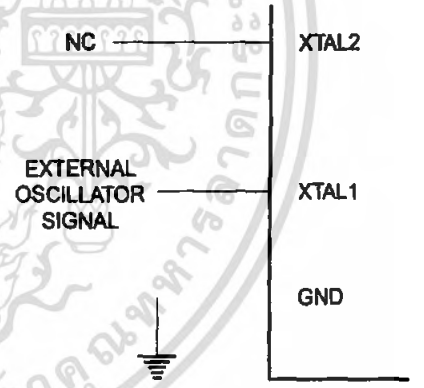
In the power-down mode, the oscillator is stopped, and the instruction that invokes power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power-down mode is terminated. The only exit from power-down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Figure 7. Oscillator Connections



Note: C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Figure 8. External Clock Drive Configuration



Status of External Pins During Idle and Powe-down Modes

Mode	Program Memory	ALE	$\overline{PSEN}$	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Program Memory Lock Bits

The AT89C52 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the following table.

Lock Bit Protection Modes

Program Lock Bits				
	LB1	LB2	LB3	Protection Type
1	U	U	U	No program lock features.
2	P	U	U	MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset, and further programming of the Flash memory is disabled.
3	P	P	U	Same as mode 2, but verify is also disabled.
4	P	P	P	Same as mode 3, but external execution is also disabled.

When lock bit 1 is programmed, the logic level at the $\overline{EA}$ pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value and holds that value until reset is activated. The latched value of $\overline{EA}$ must agree with the current logic level at that pin in order for the device to function properly.

Programming the Flash

The AT89C52 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (V_{CC}) program enable signal. The Low-voltage programming mode provides a convenient way to program the AT89C52 inside the user's system, while the high-voltage programming mode is compatible with conventional third-party Flash or EPROM programmers.

The AT89C52 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table.

	V_{PP} = 12V	V_{PP} = 5V
Top-side Mark	AT89C52 xxxx yyww	AT89C52 xxxx - 5 yyww
Signature	(030H) = 1EH (031H) = 52H (032H) = FFH	(030H) = 1EH (031H) = 52H (032H) = 05H

The AT89C52 code memory array is programmed byte-by-byte in either programming mode. *To program any non-blank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.*

Programming Algorithm Before programming the AT89C52, the address, data and control signals should be set up according to the Flash programming mode table and Figure 9 and Figure 10. To program the AT89C52, take the following steps.

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise $\overline{EA}/V_{PP}$ to 12V for the high-voltage programming mode.
5. Pulse ALE/$\overline{PROG}$ once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 1.5 ms. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

Data Polling The AT89C52 features $\overline{Data}$ Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written data on PO.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. $\overline{Data}$ Polling may begin any time after a write cycle has been initiated.

Ready/$\overline{Busy}$ The progress of byte programming can also be monitored by the RDY/$\overline{BSY}$ output signal. P3.4 is pulled low after ALE goes high during programming to indicate $\overline{BUSY}$. P3.4 is pulled high again when programming is done to indicate READY.

Program Verify If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

Chip Erase The entire Flash array is erased electrically by using the proper combination of control signals and by holding ALE/$\overline{PROG}$ low for 10 ms. The code array is written with all 1s. The chip erase operation must be executed before the code memory can be reprogrammed.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Reading the Signature Bytes The signature bytes are read by the same procedure as a normal verification of locations 030H, 031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

(030H) = 1EH indicates manufactured by Atmel
(031H) = 52H indicates 89C52
(032H) = FFH indicates 12V programming
(032H) = 05H indicates 5V programming

Programming Interface

Every code byte in the Flash array can be written, and the entire array can be erased, by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

All major programming vendors offer worldwide support for the Atmel microcontroller series. Please contact your local programming vendor for the appropriate software revision.

Flash Programming Modes

Mode		RST	$\overline{PSEN}$	ALE/$\overline{PROG}$	$\overline{EA}/V_{PP}$	P2.6	P2.7	P3.6	P3.7
Write Code Data		H	L		H/12V	L	H	H	H
Read Code Data		H	L	H	H	L	L	H	H
Write Lock	Bit - 1	H	L		H/12V	H	H	H	H
	Bit - 2	H	L		H/12V	H	H	L	L
	Bit - 3	H	L		H/12V	H	L	H	L
Chip Erase		H	L	(1)	H/12V	H	L	L	L
Read Signature Byte		H	L	H	H	L	L	L	L

Note: 1. Chip Erase requires a 10 ms $\overline{PROG}$ pulse.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 9. Programming the Flash Memory

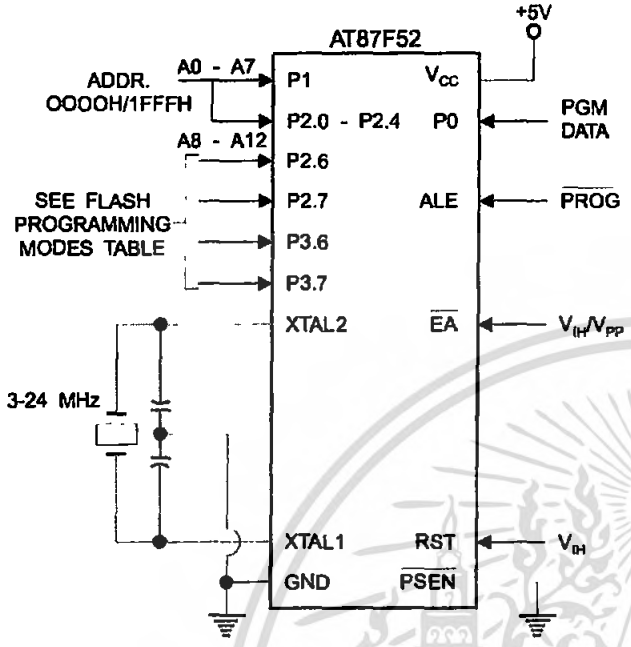
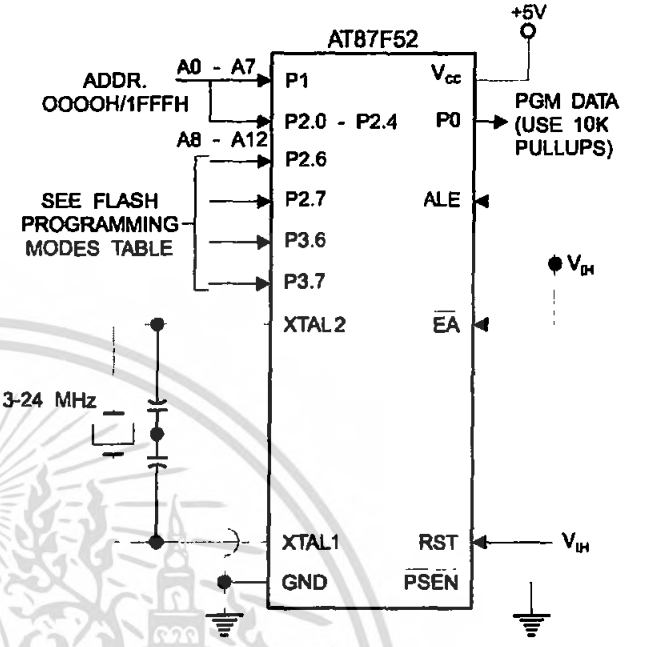


Figure 10. Verifying the Flash Memory



Flash Programming and Verification Characteristics

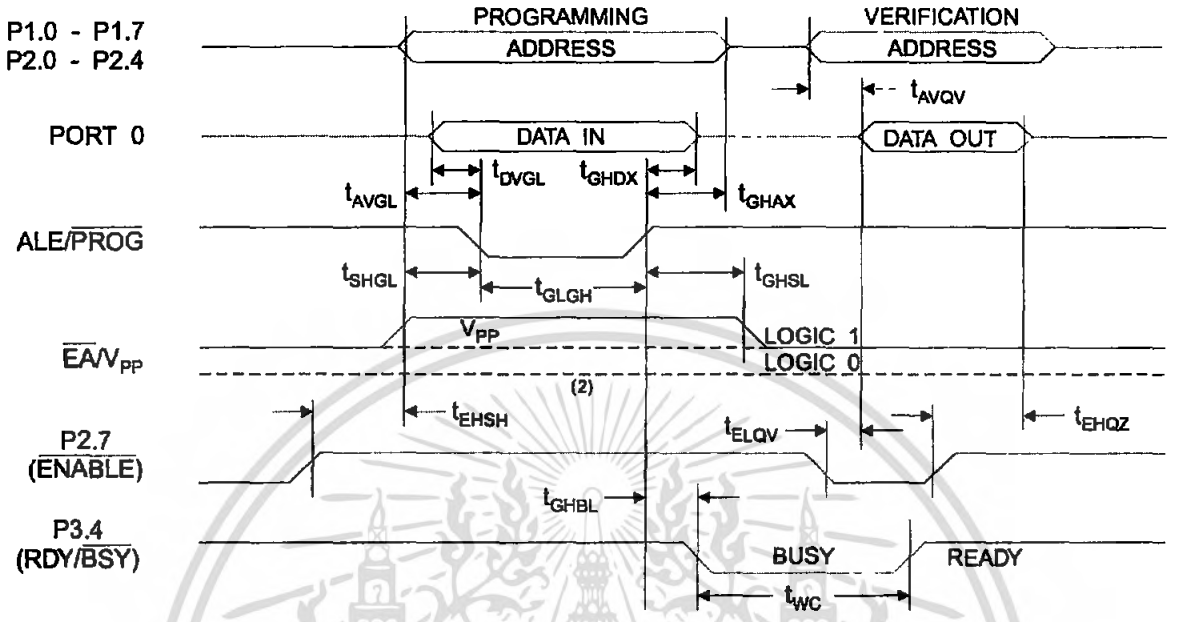
T_A = 0°C to 70°C, V_{CC} = 5.0 ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}[1]	Programming Enable Voltage	11.5	12.5	V
I_{PP}[1]	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold after $\overline{PROG}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold After $\overline{PROG}$	$48t_{CLCL}$		
t_{EHSH}	P2.7 ($\overline{ENABLE}$) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GHSL}[1]	V_{PP} Hold after $\overline{PROG}$	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQZ}	Data Float after $\overline{ENABLE}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
t_{WC}	Byte Write Cycle Time		2.0	ms

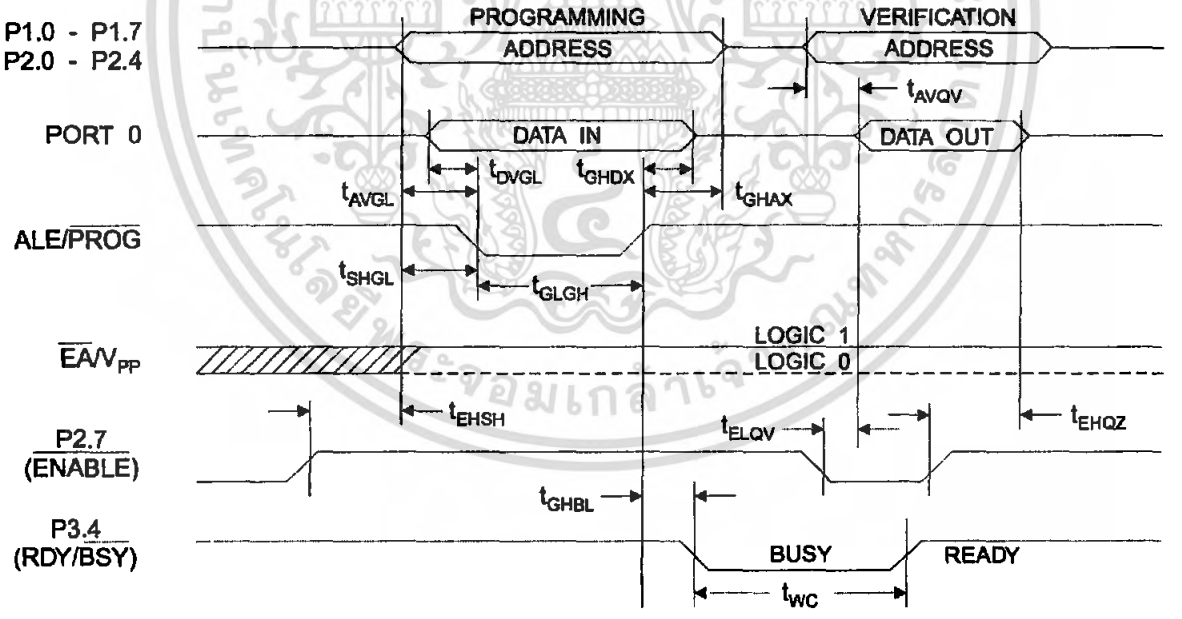
Note: 1. Only used in 12-volt programming mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Waveforms - High-voltage Mode (V_{PP}=12V)



Flash Programming and Verification Waveforms - Low-voltage Mode (V_{PP}=5V)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output Current	15.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to ebsolute maximum rating conditions for extended periods may affect device reliability.

DC Characteristics

The values shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 5.0V ±20%, unless otherwise noted.

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
V_{IL1}	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
V_{OL}	Output Low-voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low-voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ±10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ±10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pulldown Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25° C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode[1]	V_{CC} = 6V		100	µA
		V_{CC} = 3V		40	µA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
Maximum I_{OL} per port pin: 10 mA
Maximum I_{OL} per 8-bit port:
Port 0: 26 mA Ports 1, 2, 3: 15 mA
Maximum total I_{OL} for all output pins: 71 mA
If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum V_{CC} for Power-down is 2V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics

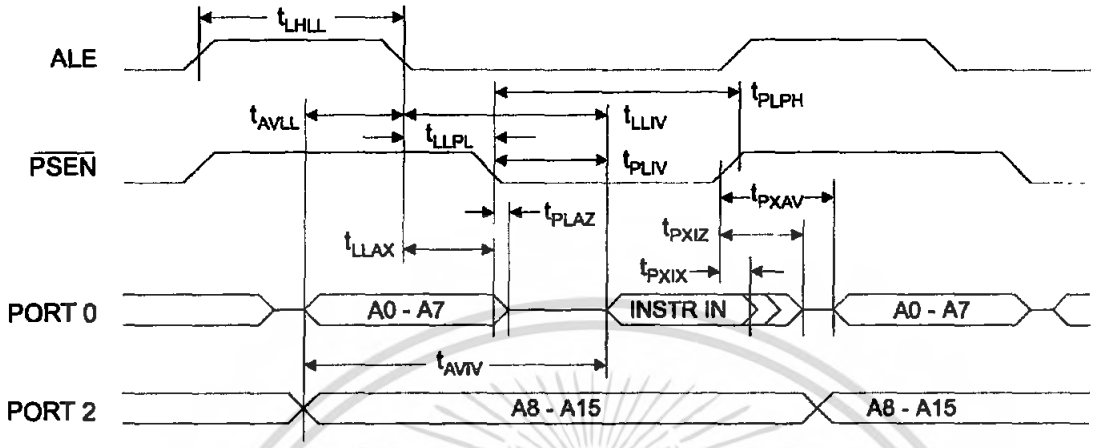
Under operating conditions, load capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; load capacitance for all other outputs = 80 pF.

External Program and Data Memory Characteristics

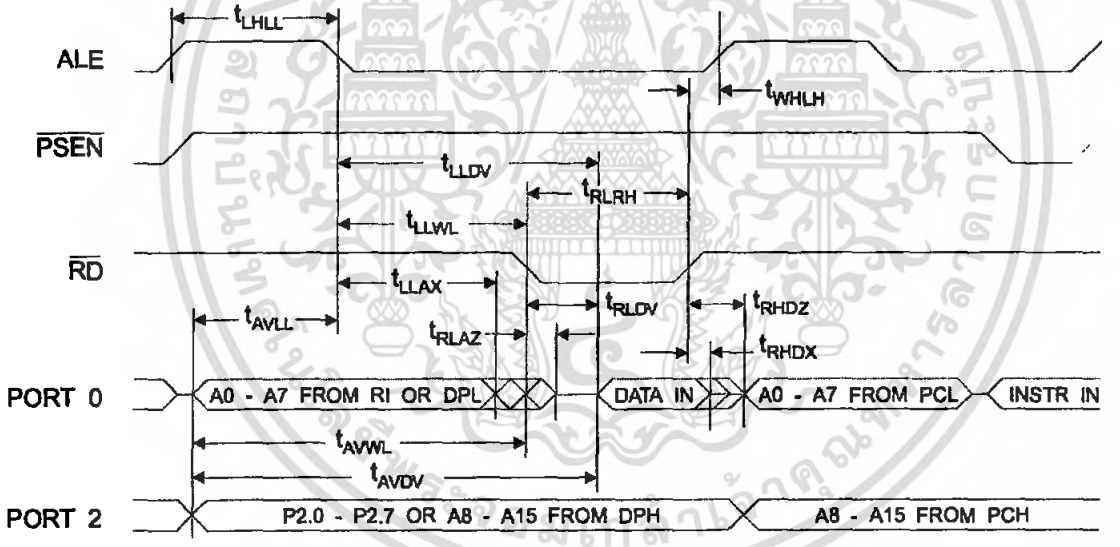
Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	43		t_{CLCL}-13		ns
t_{LLAX}	Address Hold After ALE Low	48		t_{CLCL}-20		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4t_{CLCL}$-65	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	43		t_{CLCL}-13		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	205		$3t_{CLCL}$-20		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		145		$3t_{CLCL}$-45	ns
t_{PXIX}	Input Instruction Hold after $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input Instruction Float after $\overline{PSEN}$		59		t_{CLCL}-10	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	75		t_{CLCL}-8		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}$-55	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		252		$5t_{CLCL}$-90	ns
t_{RHDX}	Data Hold After $\overline{RD}$	0		0		ns
t_{RHDZ}	Data Float After $\overline{RD}$		97		$2t_{CLCL}$-28	ns
t_{LLDV}	ALE Low to Valid Data In		517		$8t_{CLCL}$-150	ns
t_{AVDV}	Address to Valid Data In		585		$9t_{CLCL}$-165	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	203		$4t_{CLCL}$-75		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	23		t_{CLCL}-20		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	433		$7t_{CLCL}$-120		ns
t_{WHQX}	Data Hold After $\overline{WR}$	33		t_{CLCL}-20		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	43	123	t_{CLCL}-20	t_{CLCL}+25	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Program Memory Read Cycle

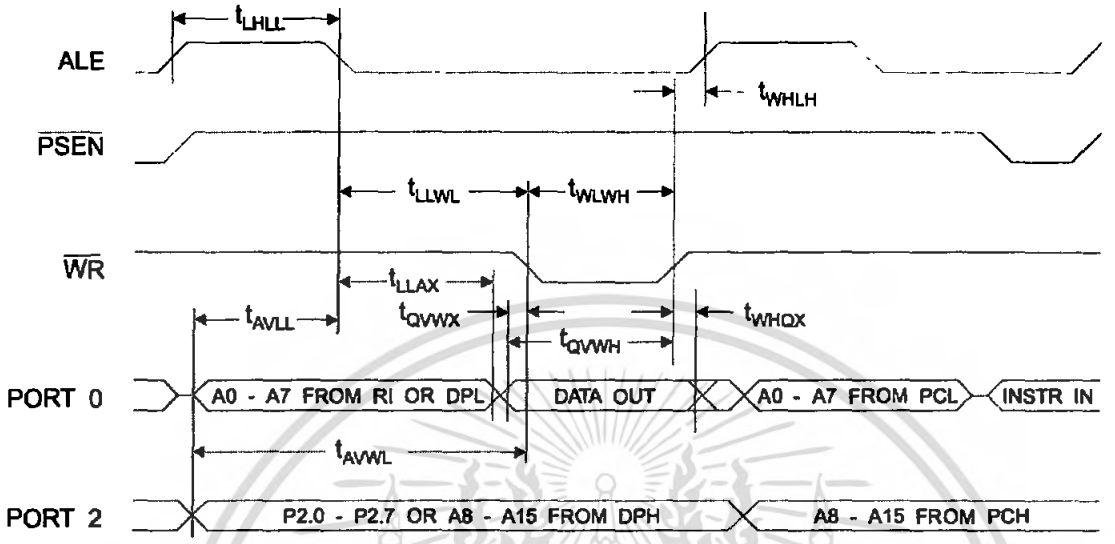


External Data Memory Read Cycle

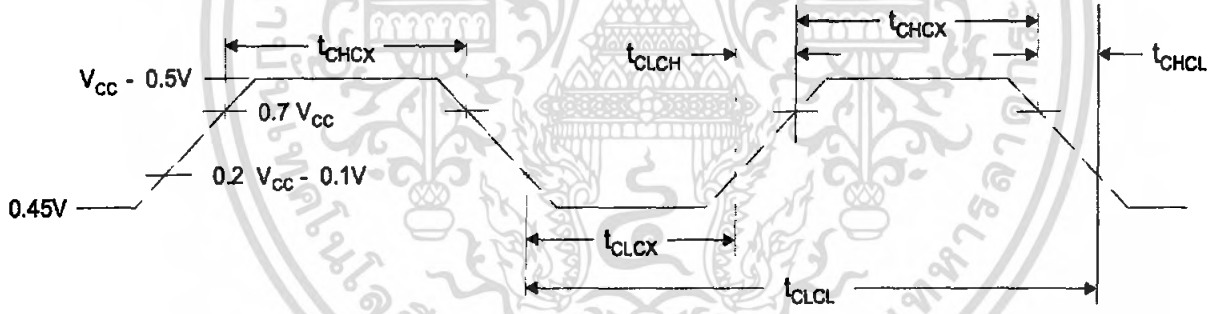


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	Min	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{CLCL}	Clock Period	41.6		ns
t_{CHCX}	High Time	15		ns
t_{CLCX}	Low Time	15		ns
t_{CLCH}	Rise Time		20	ns
t_{CHCL}	Fall Time		20	ns

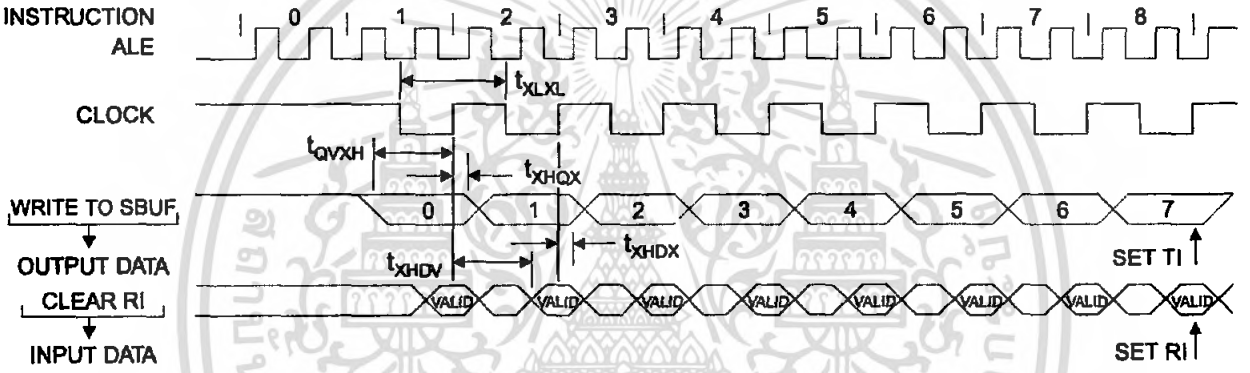
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

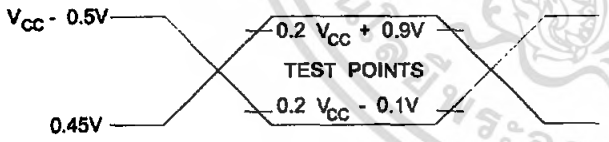
The values in this table are valid for V_{CC} = 5.0V ±20% and Load Capacitance = 80 pF.

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		µs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}$-133		ns
t_{XHQX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}$-117		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}$-133	ns

Shift Register Mode Timing Waveforms

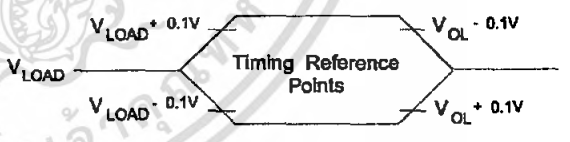


AC Testing Input/Output Waveforms[1]



Note: 1. AC Inputs during testing are driven at V_{CC} - 0.5V for a logic 1 and 0.45V for a logic 0. Timing measurements are made at V_{IH} min. for a logic 1 and V_{IL} max. for a logic 0.

Float Waveforms[1]



Note: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

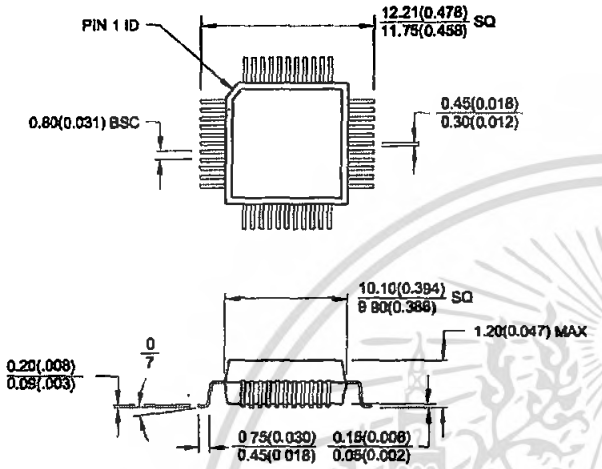
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	5V ±20%	AT89C52-12AC AT89C52-12JC AT89C52-12PC AT89C52-12QC	44A 44J 40P6 44Q	Commercial (0° C to 70° C)
		AT89C52-12AI AT89C52-12JI AT89C52-12PI AT89C52-12QI	44A 44J 40P6 44Q	Industrial (-40° C to 85° C)
16	5V ±20%	AT89C52-16AC AT89C52-16JC AT89C52-16PC AT89C52-16QC	44A 44J 40P6 44Q	Commercial (0° C to 70° C)
		AT89C52-16AI AT89C52-16JI AT89C52-16PI AT89C52-16QI	44A 44J 40P6 44Q	Industrial (-40° C to 85° C)
20	5V ±20%	AT89C52-20AC AT89C52-20JC AT89C52-20PC AT89C52-20QC	44A 44J 40P6 44Q	Commercial (0° C to 70° C)
		AT89C52-20AI AT89C52-20JI AT89C52-20PI AT89C52-20QI	44A 44J 40P6 44Q	Industrial (-40° C to 85° C)
24	5V ±20%	AT89C52-24AC AT89C52-24JC AT89C52-24PC AT89C52-24QC	44A 44J 40P6 44Q	Commercial (0° C to 70° C)
		AT89C52-24AI AT89C52-24JI AT89C52-24PI AT89C52-24QI	44A 44J 40P6 44Q	Industrial (-40° C to 85° C)

Package Type	
44A	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
44J	44-lead, Plastic J-leaded Chip Carrier (PLCC)
40P6	40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
44Q	44-lead, Plastic Gull Wing Quad Flatpack (PQFP)

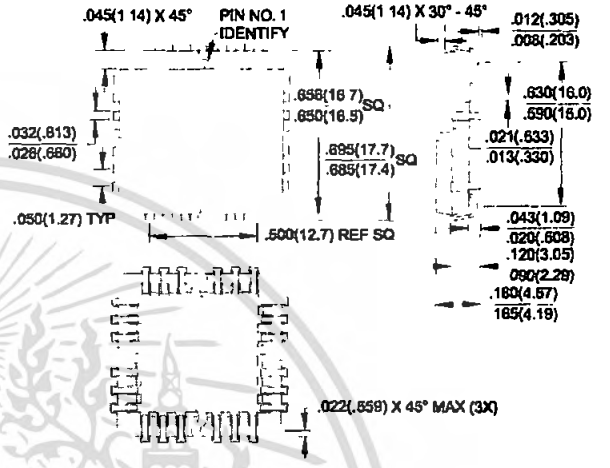
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flatpack (TQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-026 ACB

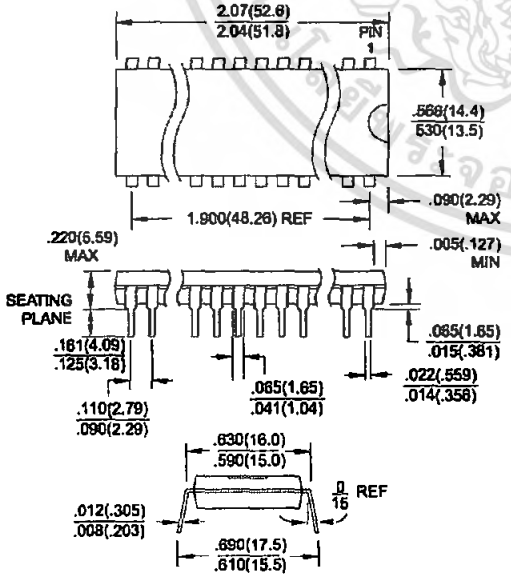


44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-018 AC

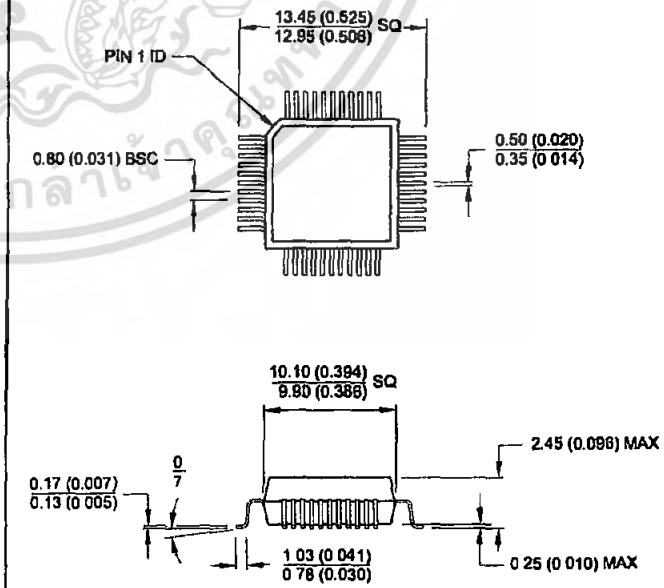


Controlling dimension: millimeters

40P6, 40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)



44Q, 44-lead, Plastic Quad Flat Package (PQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-022 AB



Controlling dimension: millimeters

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL (408) 441-0311
FAX (408) 487-2600

Europe
Atmel U.K., Ltd.
Coliseum Business Centre
Riverside Way
Camberley, Surrey GU15 3YL
England
TEL (44) 1276-686-677
FAX (44) 1276-686-697

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Fax-on-Demand
North America:
1-(800) 292-8635
International:
1-(408) 441-0732

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

© Atmel Corporation 1999.
Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

Marks bearing ® and/or ™ are registered trademarks and trademarks of Atmel Corporation.

Terms and product names in this document may be trademarks of others.

Printed on recycled paper.

0313H–02/00/xM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการคา
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้