รถสำรวจ

(OBSERVATION MOBILE ROBOT)

โดย

นาย ธนาชิต ตั้งสุขสันต์ 36014181

วัน เดือน ปี 29 ก.ย. 2541
เลขทะเบียน 038048
เลขเรียกหนังสือ T 3906 ธ 2539 ร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถสำรวจ

(OBSERVATION MOBILE ROBOT)

โดย

นาย ธนาชิต ตั้งสุขสันต์ 36014181

อาจารย์ที่ปรึกษา

อาจารย์ สุมิตร พนาอุดมทรัพย์

อาจารย์ อรพินท์ อัสรางชัย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์หรือรายงานการศึกษา

ภาควิชา วิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง รถสำรวจ

ผู้จัดทำ

นาย ธนาชิต ตั้งสุขสันต์ 36.014181

.....อาจารย์ที่ปรึกษา

(อาจารย์ สุมิตร พนาอุดมทรัพย์)

.....อาจารย์ที่ปรึกษา

(อาจารย์ อรพินท์ อัสรางชัย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถสำรวจ

นาย ธนาชิต ตั้งสุขสันต์
อ. สุมิตร พนาอุดมทรัพย์ อาจารย์ที่ปรึกษา
อ. อรพินท์ อัสรางชัย อาจารย์ที่ปรึกษา
ปีการศึกษา 2539

บทคัดย่อ

โครงงานนี้เป็นการออกแบบรถสำรวจ โดยสามารถใช้คอมพิวเตอร์ในการติดต่อและควบคุมรถสำรวจได้โดยอาศัยการสื่อสารข้อมูลแบบอนุกรม สำหรับการควบคุมที่ตัวรถสำรวจจะใช้การควบคุมจากไมโครคอนโทรลเลอร์ ในการสำรวจเบื้องต้นจะเป็นการสำรวจสิ่งกีดขวาง โดยใช้การวัดระยะทางด้วยคลื่นอุลตร้าโซนิคและส่งข้อมูลผ่านสายสัญญาณไปยังคอมพิวเตอร์ เพื่อทำการทั้งแสดงผลการวัดระยะทางระหว่างสิ่งกีดขวางและรถสำรวจที่มุมต่างๆรอบตัวรถสำรวจ บนจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Observation Mobile Robot

Tanachit Tangsukson
Sumitr Panaudomsuk Advisor
Orapin Asrangchai Advisor
1996

ABSTRACT

This project presents Observation Mobile Robot designed to be controlled by serial communication . The Observation Mobile Robot itself is controlled by Microcontroller , while the basic observation work is to observe obstructing objects using Ultrasonic beam and then to transfer data to the computer through the serial port . After that , the distance at all angle between obstructing object and the Observation Mobile Robot will appear on the screen of the computer .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
บทที่ 1	บทนำ	1
	วัตถุประสงค์	1
	ขอบข่ายของโครงงาน	2
บทที่ 2	ทฤษฎีและการทำงานของ Observation Mobile Robot	4
	2.1 ขั้นตอนการทำงาน	4
	2.2 บล็อกไดอะแกรมของระบบ	4
	2.3 ไมโครคอนโทรลเลอร์ตระกูล 8051	5
	2.3.1 คุณลักษณะพื้นฐานของ 8051	6
	2.3.2 ฐานเวลาในการทำงานของซีพียูภายใน 8051	8
	2.3.3 หน่วยความจำข้อมูล	11
	2.3.3.1หน่วยความจำภายใน	11
	2.3.3.1.ก หน่วยความจำขนาด 128 ไบต์แรก	11
	2.3.3.1.ข หน่วยความจำขนาด 128 ไบต์ถัดไป	14
	2.3.3.1.ค รีจีสเตอร์หน้าที่พิเศษ	14
	2.3.4 พอร์ตอินพุตและเอาต์พุต	16
	2.3.4.ก พอร์ตแบบขนานของ 8051	17
	2.3.4.ข โครงสร้างการทำงานของพอร์ต 8051	17
	2.3.4.ค การใช้งานพอร์ตเป็นการอินพุต	17
	2.3.4.ง การใช้งานพอร์ตเป็นการเอาต์พุต	18
	2.3.4.จ ลักษณะสมบัติของพอร์ตอินพุต/เอาต์พุต	18
	2.3.5 การอินเตอร์รัปต์	20
	2.3.5.ก ประเภทของการอินเตอร์รัปต์	20
	2.3.5.ข โครงสร้างการอินเตอร์รัปต์	20
	2.3.5.ค การควบคุมอินเตอร์รัปต์	22
	2.3.5.ง ระดับความสำคัญของการอินเตอร์รัปต์	23
	2.3.5.จ การจัดการอินเตอร์รัปต์	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6 การสื่อสารข้อมูลอนุกรม 26
2.3.6.ก จังหวะเวลาของการสื่อสารข้อมูลอนุกรม 27
2.3.6.ข รูปแบบของข้อมูลอนุกรม 28
2.3.6.ค การจัดการข้อมูลอนุกรมของ 8051 29
2.3.7 การอินเตอร์รัปของการสื่อสารอนุกรม 32
2.3.8 กระบวนการรับและส่งข้อมูลอนุกรมของ 8051 32
2.4 ทฤษฎีเกี่ยวกับอุลตร้าโซนิค 33
2.4.1 การเกิดอุลตร้าโซนิค 33
2.4.2 ความถี่และความยาว 33
2.4.3 ความเร็วของคลื่นอุลตร้าโซนิค 34
2.4.4 ปริมาณพลังงานของคลื่นอุลตร้าโซนิค 34
2.4.5 การลดทอนของคลื่นอุลตร้าโซนิค 35
2.4.6 การทำงานของทรานสดิวเซอร์ตัวส่งและตัวรับ 36
2.4.7 แนวคิดและหลักการวัดระยะทาง 37
2.5 ทฤษฎีและหลักการของการควบคุม Stepping Motor 38
2.5.1 ส่วนของ Hardware 39
2.5.1.1 ไมโครคอนโทรลเลอร์ 39
2.5.1.2 สเต็ปปิ้งมอเตอร์ 39
2.5.1.2.ก โครงสร้างและหลักการทำงานของ สเต็ปปิ้งมอเตอร์ 39
2.5.1.2.ข การกระตุ้นเฟสสเต็ปปิ้งมอเตอร์ 41
2.5.1.2.ข.1 การกระตุ้นของสเตเตอร์ 43
2.5.1.2.ค การเกิดการออสซิลเลตของสเต็ปปิ้งมอเตอร์ 47
บทที่ 3 การออกแบบและส่วนประกอบของโครงงาน 48
3.1 ส่วนคอมพิวเตอร์ 49
3.2 ส่วนนำร่อง 49
3.2.1 การใช้อุลตร้าโซนิคในการวัดระยะทาง 50
3.2.2 การทำงานของวงจรวัดระยะทางอุลตร้าโซนิค 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนการควบคุมและจัดการข้อมูล 54
3.4 การออกแบบส่วนขับเคลื่อน 55
3.4.1 ส่วนการออกแบบการตรวจวัดระยะทางการเคลื่อนที 55
3.4.2 ส่วนขับเคลื่อน 56
3.5 การออกแบบโปรโตคอล 57
3.6 โปรแกรมการควบคุมที่คอมพิวเตอร์ 58
3.7 โปรแกรมการควบคุมและจัดการข้อมูลที่ไมโครคอนโทรลเลอร์ 59
บทที่ 4 ผลการทดลอง 63
4.1 การทดลองส่วนการสำรวจ 63
4.2 การทดลองการทำงานของ Observation Mobile Robot 66
บทที่ 5 บทสรุปและวิจารณ์ 70
ภาคผนวก
หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่		หน้า
1	แผนภาพบล็อกไดอะแกรมแสดงการทำงานของ Observation Mobile Robot	3
2	แผนภาพบล็อกแสดงหน่วยการทำงานพื้นฐานของไมโครคอนโทรลเลอร์	7
3	การกำหนดหน้าที่ขาสัญญาณของไอซี 8051	8
4	แสดงแผนภาพเวลาพื้นฐานของ 8051 และลำดับของช่วงเวลา State ในการทำคำสั่งหนึ่งไบต์	9
5	แสดงการใช้คริสตอลภายนอกต่อเข้ากับวงจรออสซิลเลเตอร์ภายใน 8051	10
6	การจัดพื้นที่หน่วยความจำข้อมูลสำหรับไมโครคอนโทรเลอร์ 8051	10
7	การจัดพื้นที่หน่วยความจำข้อมูล (a) ช่วงตั้งแต่แอดเดรส และ (b) ช่วงแอดเดรสซึ่ง กำหนดให้เป็นพื้นที่ของรีจิสเตอร์หน้าที่พิเศษ	13
8 (a)	แผนภาพแสดงการส่งข้อมูลผ่านทางพอร์ตอนุกรม ซึ่งเป็นเส้นจำนวน 8 เส้น สำหรับส่งข้อมูลจากระบบไปให้กับอุปกรณ์ภายนอก	16
8 (b)	แผนภาพแสดงการส่งข้อมูลจากอุปกรณ์ภายนอกเข้ามายังพอร์ตอินพุตของระบบ ซึ่งมีลักษณะเป็นเส้นสัญญาณจำนวน 8 เส้น	16
9	โครงสร้างของแต่สะบิตภายในพอร์ตอินพุต/เอาท์พุตของ8051	19
10	แผนภาพแสดงโครงสร้างระบบอินเตอร์รัปต์ของ 8051	21
11 (a)	ข้อมูลสื่อสารแบบอนุกรม ข้อมูลหนึ่งไบต์จะถูกส่งออกทีละบิตเป็นลำดับไปจนครบจำนวนทั้ง 8	26
11 (b)	ข้อมูลสื่อสารแบบขนาน ข้อมูลแต่ละบิตภายในหนึ่งไบต์จะถูกส่งออกมาพร้อมกันในลักษณะแบบขนาน	26
12 (a)	บิตข้อมูลสื่อสารแบบขนานภายในไบต์ข้อมูลจะถูกส่งออกมาในลักษณะแบบอนุกรม	27
12 (b)	รูปแบบสัญญาณไฟฟ้าของข้อมูลในรูป 12 (a) ซึ่งเป็นการส่งข้อมูลแบบอนุกรมด้วยอัตรา 2400 บอด	27
13	แผนภาพสัญญาณเวลาของข้อมูลแบบอนุกรมจำนวน 8 บิต พร้อมด้วยบิตเริ่มต้นพาริตี้และบิตสุดท้าย	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14 แผนภาพแสดงการทำงานของวงจรส่วนการรับและส่งข้อมูลอนุกรม ของ 8051 31
15 แสดง Timing Diagramของการรับ - ส่งคลื่น Ultrasonic 37
16 บล็อกไดอะแกรมของส่วนควบคุมและชุดขับ 38
17 โครงสร้างของสเตปปิ้งมอเตอร์ 39
18 เส้นแรงแม่เหล็กที่ทำให้เกิดแรงบิด 40
19 แสดงการเคลื่อนที่ทีละสเต็ป เมื่อกระตุ้น เฟส 1 เฟส 2 41
20 แสดงการกระตุ้นแบบเดินหน้าและถอยหลัง 42
21 แสดงการทำงานของสเต็ปปิ้งมอเตอร์ 42
22 การกระตุ้นเฟสแบบเฟสเดียวของสเต็ปปิ้งมอเตอร์แบบ 3และ4เฟส 43
23 การกราะตุ้นแบบสองเฟสของเต็ปมอเตอร์ 3 เฟส และ 4 เฟส 44
24 การกระตุ้นแบบครึ่งสเต็ปของมอเตอร์แบบ 3เฟสและ 4 เฟส 45
26 ลักษณะเส้นแรงแม่เหล็กเมื่อขับแบบ 2 เฟส 46
27 วงจรขับ Stepping Motor 49
28 แสดงบล็อกไดอะแกรมของเครื่องวัดระยะทางอุลตร้าโซนิค 50
29 วงจรสมบูรณ์เครื่องวัดระยะทางอุลตร้าโซนิค 53
30 แสดงวงจรใช้งานของไมโครคอนโทรลเลอร์ 54
31 แสดงวงจร Sensor 55
32 แสดงวงจรควบคุมส่วนขับเคลื่อน 56
33 แสดงวงจรวัดระยะทางด้วยอุลตร้าโซนิค 65
34 แสดงการจัดตั้งสิ่งกีดขวาง 67
35 แสดงตัวอย่างผลการสำรวจ ที่ปรากฏบนจอภาพคอมพิวเตอร์ 68
36 แสดงภาพรถสำรวจด้านหน้า 69
37 แสดงภาพรถสำรวจด้านหลัง 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่		หน้า
1	ตารางแสดงไมโครคอนโทรเลอร์ตระกูล MCS-51 ของบริษัทอินเทล	6
2	บริเวณแอดเดรส 00H - 1FH	12
3	บริเวณแอดเดรสของกลุ่มรีจิสเตอร์	12
4	รีจิสเตอร์ใช้งานพิเศษ (Special Function หรือ SFR)	15
5	แสดงสัญญาณอินเตอร์รัปต์ ทั้ง 5 แหล่ง	20
6	บิตต่าง ๆ ของการอนุญาตการอินเตอร์รัปต์	22
7	บิตต่าง ๆ ของการจัดระดับความสำคัญของการอินเตอร์รัปต์	24
8	แสดงตำแหน่งของการอินเตอร์รัปต์	25
9	โหมดต่าง ๆ ของสื่อสารอนุกรม	30
10	รีจิสเตอร์ควบคุมการทำงานและบอกสถานะการสื่อสารข้อมูลอนุกรม	31
11	แสดงความเร็วของคลื่นในตัวกลางต่าง ๆ	35
12	แสดงทิศการเคลื่อนที่ของรถสำรวจ	56
13	แสดงรหัสคำสั่งที่ใช้ในการควบคุม	57
14	แสดงผลการ ผลการวัดระยะทางด้วยอุลตร้าโซนิค	64
15	แสดงผลการสำรวจ	67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1 บทนำ

ปัจจุบันวิวัฒนาการทางด้านเทคโนโลยีได้มีการนำหุ่นยนต์มาใช้ในลักษณะงานต่าง ๆ กันอย่างแพร่หลายซึ่งสามารถแบ่ง หุ่นยนต์ได้เป็น 2 ประเภทใหญ่ ๆ คือ

1. Fixed Robot
2. Mobile Robot

Fixed Robot คือ หุ่นยนต์ที่ฐานถูกยึดอยู่กับที่ ไม่สามารถเคลื่อนที่ได้อิสระ เช่น แขนกล ซึ่งทำหน้าที่ในการประกอบชิ้นส่วนอุปกรณ์ต่างๆในโรงงานอุตสาหกรรมเป็นต้น

Mobile Robot คือ หุ่นยนต์ที่ถูกตรึงกับฐาน ที่สามารถเคลื่อนที่ได้โดยอิสระ โดยฐานอาจมีล้อหรือระบบการเคลื่อนที่อื่นก็ได้ โดยงานที่สำคัญอย่างหนึ่งของหุ่นยนต์ชนิดนี้ก็คืองานสำรวจ (Observation)

โดยการที่นำ Moblie Robot มาใช้งานสำรวจหรือปฏิบัติงาน เช่น งานที่เสี่ยงอันตรายต่อชีวิตมนุษย์ ตัวอย่าง เช่น งานกู้วัตถุระเบิด หรือ งานสำรวจที่มนุษย์ไม่สามารถเข้าไปสำรวจได้ด้วยตัวเองได้ โดยอาจมีปัญหาทางด้านมลภาวะทางด้านสิ่งแวดล้อมการทำงานต่าง ๆ อันได้แก่ อุณหภูมิ,รังสี,สารเคมี, ฝุ่นละออง เป็นต้น ยกตัวอย่าง เช่น ขณะที่เกิดไฟไหม้ในแหล่งที่มีการเก็บสารเคมีอันตรายแทนที่จะส่งมนุษย์เข้าไปตรวจสอบ สำรวจพื้นที่ซึ่งเป็นการเสี่ยงต่อการสูญเสียชีวิตมากเราสามารถที่จะส่ง Mobile Robot เข้าไปสำรวจแทนได้โดยการที่เราจะได้ข้อมูลชนิดใดขึ้นกับการติดตั้ง Sensor หรืออุปกรณ์สำรวจต่าง ๆ

วัตถุประสงค์ของโครงงาน

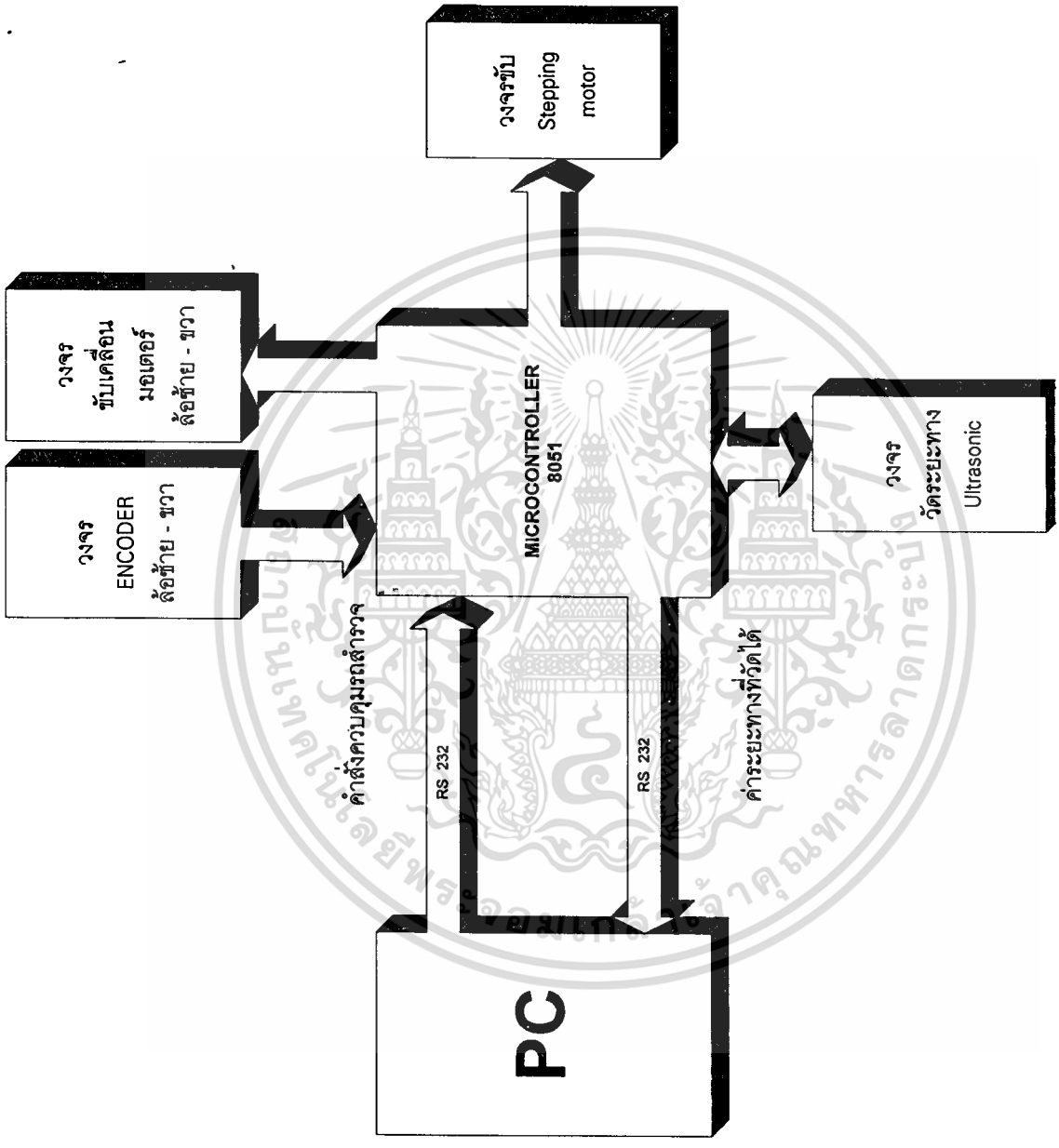
1. เพื่อทำหน้าที่ในการสำรวจแทนมนุษย์ในสถานที่ ที่มีอันตรายหรือสถานที่มนุษย์ไม่สามารถเข้าถึงได้
2. เพื่อที่จะสามารถทำการตรวจวัดปัจจัยทางกายภาพต่าง ๆ ได้อย่างแม่นยำและถูกต้องผ่าน Sensor หรือ อุปกรณ์ การสำรวจต่าง ๆ เช่น กล้อง CCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบข่ายของโครงงาน

1. สามารถควบคุมและป้อนโปรแกรมได้จาก คอมพิวเตอร์
2. สามารถตรวจจับสิ่งกีดขวางและวัดระยะได้รอบตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1 แผนภาพบล็อกไดอะแกรมแสดงการทำงานของ Observation Mobile Robot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการทำงานของ Observation Mobile Robot

2.1 ขั้นตอนการทำงาน

โดยเริ่มต้นเราจะทำการป้อนโปรแกรมซึ่งเป็น Protocal ในการสื่อสารอนุกรมให้กับคอมพิวเตอร์ และ ป้อนโปรแกรมซึ่งใช้ในการควบคุมว่าต้องการจะให้ Mobile Robot ทำงานอย่างไรให้กับ ไมโครคอนโทรลเลอร์ ใน Mobile Robot จากนั้นเราก็จะสามารถควบคุม Mobile Robot จากคอมพิวเตอร์ ได้โดยเมื่อเราป้อนคำสั่งซึ่งบอกถึงรายละเอียดต่าง ๆ ที่ใช้ในการสำรวจ เช่น เข้าไปสำรวจที่ไหน อย่างไร ให้กับคอมพิวเตอร์แล้วจึงสั่งให้ส่งข้อมูลผ่าน พอร์ตอนุกรม และเมื่อ ไมโครคอนโทรลเลอร์ ได้รับรู้คำสั่งก็จะเริ่มทำการสั่งให้ส่วนต่าง ๆ ทำงานตามคำสั่งโดยอาจเป็นส่วนขับเคลื่อน หรือ ส่วนนำร่อง ให้ทำงานตามคำสั่งเป็นต้นโดยจะมีลำดับการทำงานดังบล็อกไดอะแกรมรูปที่ 1

2.2 บล็อคไดอะแกรมของระบบ โดยแต่ละส่วนมีหน้าที่ดังนี้

1. ส่วนคอมพิวเตอร์ จะทำหน้าที่รับการป้อนโปรแกรมที่จะใช้ในการควบคุม Mobile Robot โดนเมื่อผู้ใช้ทำการป้อนข้อมูลในการควบคุมเข้าไปแล้ว คอมพิวเตอร์จะทำการแปลงข้อมูลที่ใช้ในการควบคุมให้เป็นรูปแบบที่จะใช้ส่งผ่าน พอร์ตอนุกรม โดยจะใช้ โปรโตคอลที่โปรแกรมไว้เป็นตัวแปลงข้อมูลแล้วจึงส่งต่อไปที่ พอร์ตอนุกรมของ ไมโครคอนโทรลเลอร์

2. ไมโครคอนโทลเลอร์ จะทำหน้าที่ในการควบคุมการทำงานของตัว Mobile Robot ทั้งหมดโดยจะรับโปรแกรมการควบคุมจาก พอร์ตอนุกรม และทำการควบคุมส่วนต่างๆดังนี้

- ควบคุมส่วนวงจรขับเคลื่อนล้อซ้าย - ล้อขวา โดยจะทำการป้อนสัญญาณดิจิตอลควบคุมการขับเคลื่อน ว่าจะให้ล้อใดหมุนแล้วจึงรับค่าจาก Encoder วัดระยะทางที่เพลาล้อเพื่อตรวจดูว่าได้ระยะทาง ตามที่ต้องการหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ควบคุมการหมุนของ Stepping Motor ของส่วนนำร่อง โดยไมโครคอนโทรเลอร์จะทำการป้อนไฟให้กับแต่ละเฟสของ Stepping Motor เพื่อให้ได้ทิศทางที่จะวัดระยะตามที่ต้องการ
- ควบคุมส่วนวัดระยะทางอุลตร้าโซนิกว่าจะให้วัดที่ตำแหน่งใด โดยจะเป็นการป้อนสัญญาณให้เครื่องทำงาน และเมื่อได้ค่าระยะทางกลับมาแล้ว ไมโครคอนโทรเลอร์จะนำค่าที่ได้ไปเก็บในหน่วยความจำที่กำหนดไว้โดยการเก็บค่าจะมีความสัมพันธ์กับตำแหน่งการหมุนของ Stepping Motor ด้วยเพื่อจะได้รู้ว่าในทิศทางใดมีสิ่งกีดขวางห่างออกไปเท่าไร

3. ส่วนขับ Stepping Motor จะทำหน้าที่หมุนส่วนรับและตัวส่งคลื่นอุลตร้าโซนิกส์ไปในทิศทางต่าง ๆ เพื่อที่จะสามารถตรวจสอบสิ่งกีดขวางได้รอบตัวโดยสามารถควบคุมทิศ ทางได้ตามต้องการ

4. ส่วนวัดระยะทางอุลตร้าโซนิก จะทำหน้าที่วัดระยะระหว่างตัว Mobile Robot กับสิ่งกีดขวางแล้วทำการส่งสัญญาณบอกระยะห่างออกมาเป็นจำนวนพัลส์ ให้ไมโครคอนโทลเลอร์สามารถวัดได้

2.3 ไมโครคอนโทรลเลอร์ตระกูล 8051

บรรดาไมโครคอนโทรลเลอร์ที่มีการผลิตจากบริษัทต่าง ๆ จำนวนมากนั้น ไมโครคอนโทรลเลอร์จากบริษัท อินเทล (Intel Corporation) ในตระกูล MCS-51 ได้มีการนำไปใช้งานกันแพร่หลายมากนับตั้งแต่ปีค.ศ. 1980 เป็นต้นมา ในระยะที่ผ่านมาก็ได้มีอีกหลายบริษัท เช่น บริษัท Phillips และ Siemens เป็นต้น ได้รับลิขสิทธิ์ในการไปผลิตจำหน่ายและได้มีการเพิ่มประสิทธิภาพและหน่วยการทำงานต่าง ๆ มากขึ้น ทำให้ในปัจจุบันมีไมโครคอนโทรลเลอร์จากผู้ผลิตต่าง ๆ ที่มีพื้นฐานมาจากไมโครคอนโทรลเลอร์ MCS- 51 ของบริษัทอินเทลอยู่เป็นจำนวนมาก

ไมโครคอนโทรลเลอร์ตระกูล Mcs-51 ประกอบด้วย ไมโครคอนโทรลเลอร์หลายรุ่น (version) ซึ่งมีสถาปัตยกรรมพื้นฐานที่เหมือนกัน เพียงแต่มีขนาดหรือจำนวนของหน่วยทำงานภายในที่แตกต่างกันออกไป เพื่อความเหมาะสมในงานประยุกต์ต่าง ๆ ตามความต้องการดังแสดงให้เห็นในตารางที่ 1 โดยมีทั้งลักษณะที่ใช้เทคโนโลยีการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไอซีวงจรรวมความจุสูงมาก (LSI) แบบ HMOS หรือ CHMOS ซึ่งมีคุณลักษณะที่สูงมาก ขึ้น และสิ้นเปลืองกำลังไฟฟ้าน้อยกว่ามาก

EMBEDDED CONTROLLERS										
Feature	**8051AH**	**8031AH**	**8751H**	**80C51BH**	**80C31BH**	**87C51**	**8052AH**	**8032AH**	**8752**	**8044H**
Program Memory (Bytes)	4K	-	4K	4K	-	4K	8K	-	8K	4K
RAM Memory (Bytes)	128	128	128	128	128	128	256	256	256	192
Program Memory Expansion (Off Chip) (Bytes)	64K	64K	64K	64K	64K	64K	64K	64K	64K	64K
Data Memory Expansion (Off Chip) (Bytes)	64K	64K	64K	64K	64K	64K	64K	64K	64K	64K
Max Clock Frequency (MHz)	12	12	12	16	16	16	16	12	12	12
Typical instruction Time (uS)	1	1	1	0.75	0.75	0.75	1	1	1	1
16-Bit Timer / Counters	2	2	2	2	2	2	3	3	3	2
Serial Communications	Synchronous Mode, Asynchronous Mode, 9 or 10 - Bit Programmable									HDLC/SDLC
No. of I/O Lines	32	16	32	32	16	32	32	16	32	32
Interrupt Sources	5	5	5	5	5	5	6	6	6	5
(Two Priority Levels)										
Power Requirements 125	125	250	24	24	29	175	175	175	200	
(ICC Max. mA)										
Programmable Power Modes	-	-	-	4.0 mA	4.0 mA	4.0 mA	-	-	-	-
Idle Power Down				50 uA	51 uA	52 uA				30 mA

ตารางที่ 1 ตารางแสดงไมโครคอนโทรเลอร์ตระกูล MCS-51 ของบริษัทอินเทล

2.3.1 คุณลักษณะพื้นฐานของ 8051

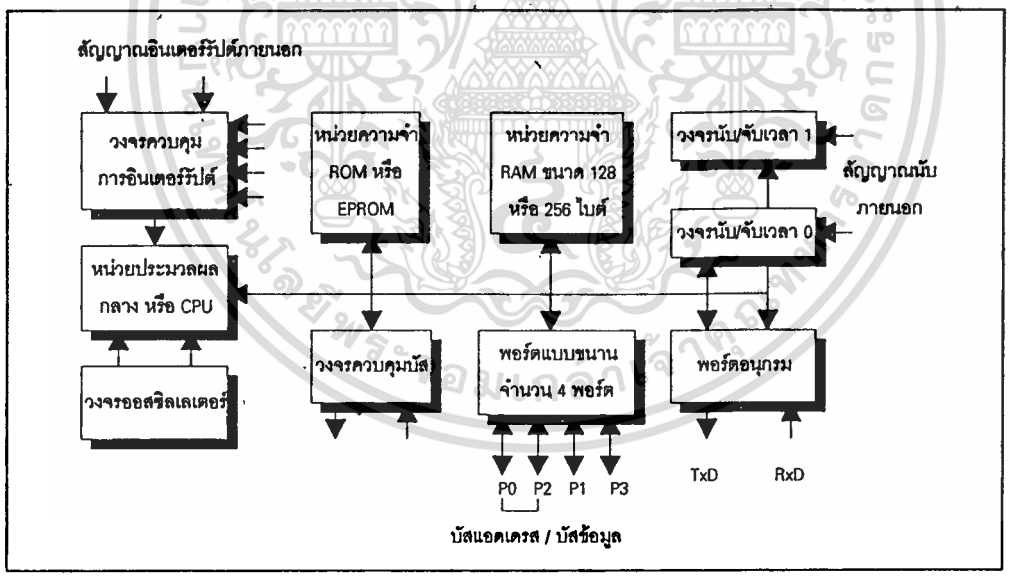
จากแผนภาพในรูปที่ 2 แสดงให้เห็นถึงหน่วยการทำงานพื้นฐานของไมโครคอนโทรลเลอร์เบอร์ต่าง ๆ ที่จัดอยู่ภายในตระกูล MCS-51 นี้ ประกอบด้วย

- หน่วยประมวลผลกลางขนาด 8 บิต
- หน่วยประมวลผลสำหรับข้อมูลแบบบิต (Boolean Processor)
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำโปรแกรม 64 กิโลไบต์
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำข้อมูล 64 กิโลไบต์
- หน่วยความจำโปรแกรมภายในขนาด 4 กิโลไบต์ แบบ EPROM (เบอร์8751) หรือแบบ ROM (เบอร์ 8051)
- หน่วยความจำแบบ RAM ภายในจำนวน 128 ไบต์
- พอร์ตอินพุต/ เอาต์พุตแบบขนานจำนวน 32 เส้น ซึ่งสามารถแยกทำงานได้อย่างอิสระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

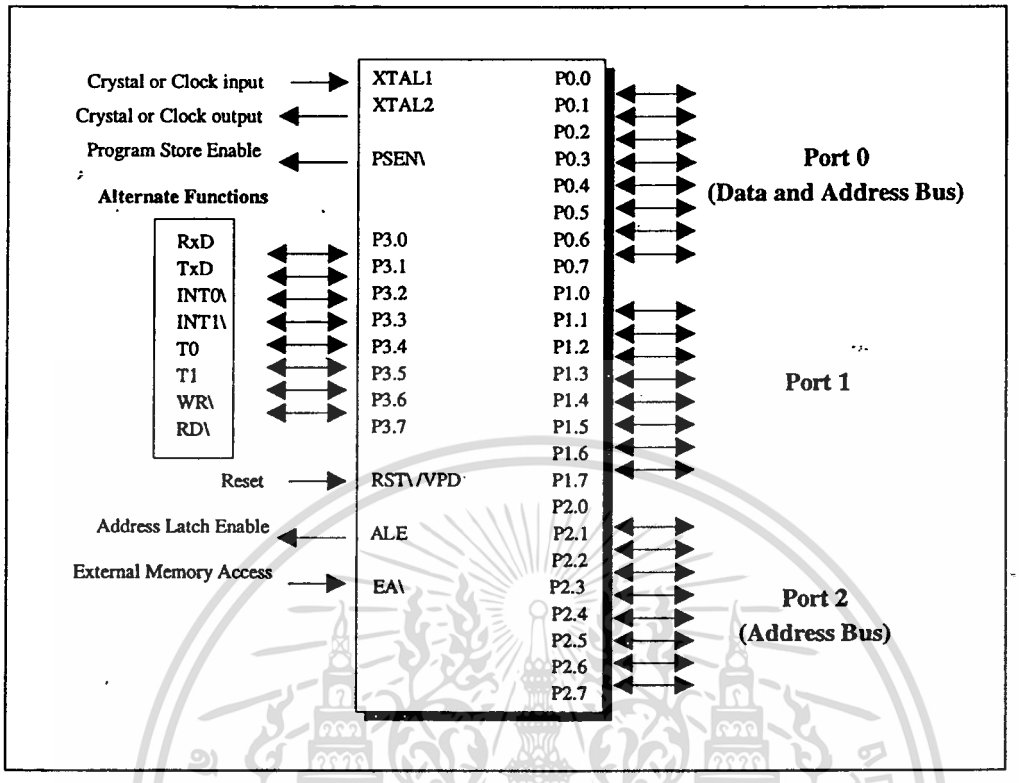
- วงจรนับ/จับเวลาขนาด 16 บิต จำนวนสองวงจร
- วงจรสื่อสารแบบอนุกรมแบบฟูลดูเพล็กซ์ (Full Dulplex)
- วงจรควบคุมการอินเตอร์รัปต์จากแหล่งกำเนิดสัญญาณ 6 ประเภท พร้อมการกำหนดลำดับความสำคัญได้สองระดับ
- วงจรออสซิลเลเตอร์ภายใน

โดยมากแล้วไมโครคอนโทรลเลอร์ตระกูลนี้ มักจะมีรูปร่างของไอซีเป็นแบบ DIP ขนาด 40 ขา ดังแผนภาพในรูปที่ 3 ซึ่งแต่ละขาสัญญาณจะมีหน้าที่ที่ระบุชัดเจนตามสัญลักษณ์ชื่อย่อที่กำกับในแต่ละขา อย่างไรก็ตามจะมีบางขาสัญญาณที่อาจจะมีหน้าที่ได้มากกว่าหนึ่งอย่าง(ซึ่งเขียนกำกับไว้ ว่า Alternate Functions ในรูปที่ 3) ซึ่งจะไม่สามารถใช้งานในเวลาเดียวกันได้



รูปที่ 2 แผนภาพบล็อกแสดงหน่วยการทำงานพื้นฐานของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3 การกำหนดหน้าที่ขาสัญญาณของไอซี 8051

2.3.2 ฐานเวลาในการทำงานของซีพียูภายใน 8051

8051 มีวงจรออสซิลเลเตอร์อยู่ภายใน สำหรับการสร้างพัลส์ของสัญญาณนาฬิกา ซึ่งจะนำไปเป็นฐานเวลาหรือการกำหนดจังหวะการทำงานของหน่วยการทำงานทั้งหมดให้ สอดคล้องกัน (Synchronization) โดยปกติแล้วก็มักจะทำโดยการใช้คริสตอลเชื่มต่อเข้ากับขาสัญญาณ XTAL1 และ XTAL2 พร้อมกับตัวเก็บประจุดังลักษณะในรูปที่ 4 หรืออาจจะเป็นสัญญาณนาฬิกาจากภายนอกก็ได้

พัลส์ความถี่ของสัญญาณนาฬิกาจะเรียกว่า Pulse (ใช้สัญลักษณ์เป็นตัวอักษร P) และคาบของสัญญาณนาฬิกานี้ เรียกว่า คาบเวลาออสซิลเลเตอร์ (Oscillator period) คาบเวลาออสซิลเลเตอร์จำนวนสองคาบเรียกว่า State (ใช้สัญลักษณ์เป็นตัวอักษร S) ซึ่งจะนำไปใช้เป็นช่วงเวลาพื้นฐานการทำงานย่อยของไมโครคอนโทรลเลอร์ เช่น การนำคำสั่ง (Fetch) การถอดความหมาย (Decode) การประมวลผล (Execute) และการเขียน ข้อมูล (Write) เป็นต้น ดังแสดงเป็นแผนภาพในรูปที่ 5 ช่วงเวลาของ State จำนวนหกครั้ง จะเรียกว่า แมชชีนไซเคิล (Machine Cycle)ดังนั้นค่าหนึ่งแมชชีนไซเคิลจะใช้เวลา 12 คาบเวลาออสซิลเลเตอร์ ค่าของแมชชีนไซเคิลนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

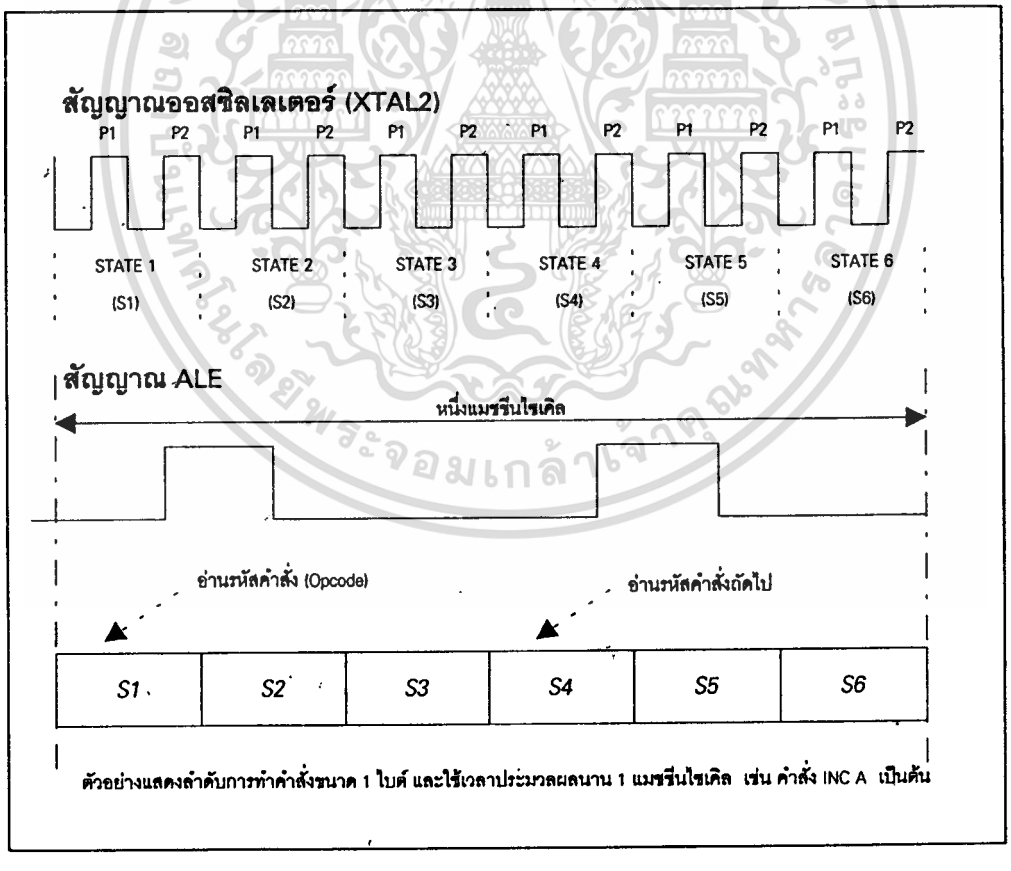
จัดว่าเป็นช่วงเวลาที่น้อยที่สุดในการทำคำสั่งใดคำสั่งหนึ่ง ซึ่งหากว่าเป็นคำสั่งที่ซับซ้อนมากก็จะต้องใช้เวลานานสองถึงสามแมชชีนไซเคิล

การคำนวณหาว่าเวลาที่ใช้ในการทำคำสั่งใดจนเสร็จสิ้น จะต้องดูว่าคำสั่งนั้นใช้จำนวนแมชชีนไซเคิลเป็นเท่าไรในการประมวลผล เวลาที่ใช้จะคำนวณตามสูตร

$$T = \frac{Cx\ 12}{\text{Crystal Frequency}}$$

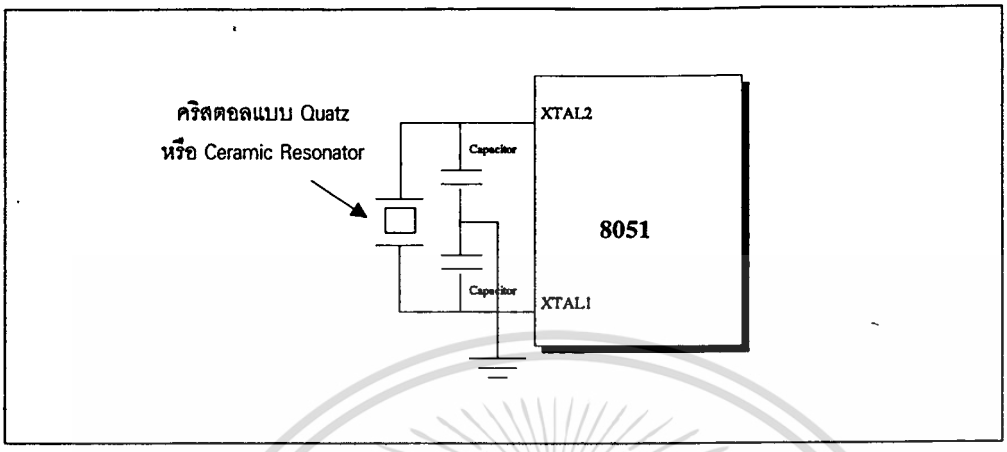
โดย C เป็นค่าจำนวนแมชชีนไซเคิลของคำสั่ง

Crystal Frequency เป็นค่าความถี่ของคริสตอลที่ใช้กับ 8051

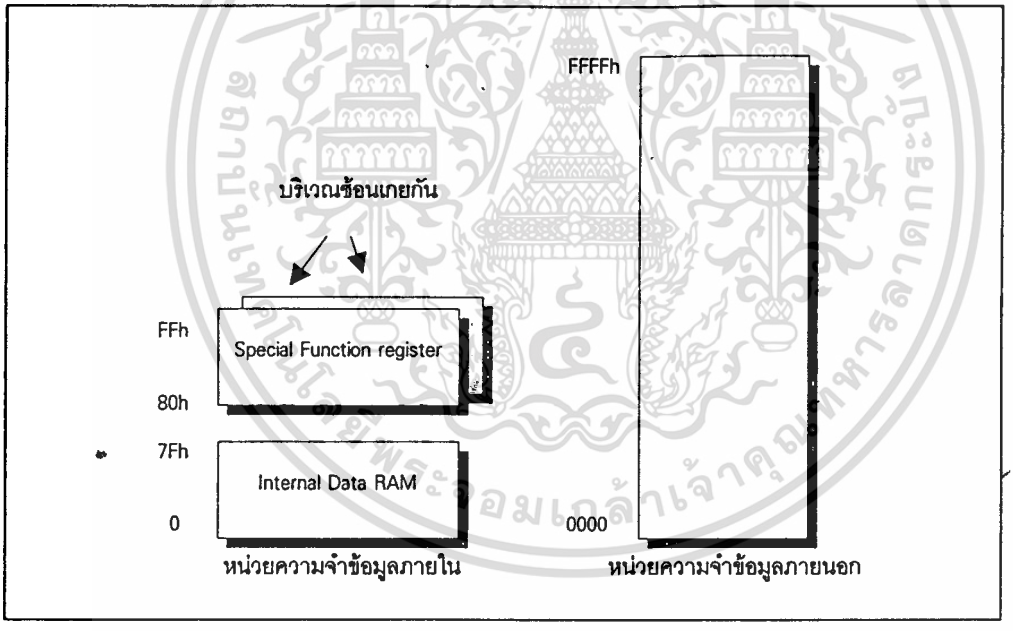


รูปที่ 4 แสดงแผนภาพเวลาพื้นฐานของ 8051 และลำดับของช่วงเวลา State ในการทำคำสั่งหนึ่งไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5 แสดงการใช้คริสตอลภายนอกต่อเข้ากับวงจรออสซิลเลเตอร์ภายใน 8051



รูปที่ 6 การจัดพื้นที่หน่วยความจำข้อมูลสำหรับไมโครคอนโทรเลอร์ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 หน่วยความจำข้อมูล

หน่วยความจำข้อมูลมีหน้าที่สำหรับเก็บข้อมูล หรือตัวแปรที่เกิดขึ้นในขณะที่กำลังประมวลผลโปรแกรมไว้เป็นการชั่วคราว โดยพื้นฐานแล้วหน่วยความจำข้อมูลจัดเป็นหน่วยความจำ RAM แบบสแตติก ดังนั้นเมื่อไม่มีการจ่ายไฟฟ้าให้กับระบบ ก็จะมีผลทำให้ข้อมูลที่จัดเก็บไว้ภายในหน่วยความจำนี้สูญหายไป พื้นที่ ของหน่วยความจำข้อมูลของ 8051 สามารถมีได้สูงสุดไม่เกิน 64 กิโลไบต์ ละแยกประเภทออกเป็นสองลักษณะตามตำแหน่งที่ตั้งของหน่วยความจำนั้น ดังลักษณะของแผนภาพในรูปที่ 6 คือ หน่วยความจำโปรแกรมภายใน (Internal Data Memory) ซึ่งเป็น RAM ที่อยู่ภายในตัวของไอซีไมโครคอนโทรลเลอร์เอง และหน่วยความจำข้อมูลภายนอก (External Data Memory) ซึ่งเป็นการใช้ไอซีหน่วยความจำ RAM มาเพิ่มเติมเข้าไปในวงจรลักษณะ เดียวกับการนำไอซี EPROM มาใช้งานเป็นหน่วยความจำโปรแกรมนั่นเอง

2.3.3.1 หน่วยความจำข้อมูลภายใน

หน่วยความจำข้อมูลภายในของ 8051 มีจำนวนทั้งหมด 256 ไบต์ โดยจำแนกออกได้เป็นสองลักษณะคือ พื้นที่เฉพาะสำหรับตัวประมวลผลกลาง หรือซีพียู ใช้งานเท่านั้น ซึ่งเรามักจะเรียกกันในอีกชื่อหนึ่งว่า รีจิสเตอร์ และพื้นที่ของหน่วยความจำข้อมูลภายในของ 8051 ซึ่งจำแนกออกเป็นสองส่วน ดังนี้

2.3.3.1.กหน่วยความจำขนาด 128 ไบต์แรก

บริเวณนี้จะมีตำแหน่งแอดเดรสอยู่ในช่วง OOH-7FH ซึ่งยังได้มีการจำแนกย่อยออกไปอีกเป็นสามส่วนตามประเภทของการใช้งาน ดังนี้

บริเวณแอดเดรส OOH-1FH จำนวน 32 ไบต์ จำแนกออกเป็นกลุ่ม (หรือ แบงก์ (Bank) ข้อมูลจำนวน 8 ไบต์ รวมทั้งหมดสี่กลุ่ม พื้นที่ข้อมูลในแต่ละกลุ่มจะถูกใช้งานในฐานะของรีจิสเตอร์ใช้งานทั่วไป ซึ่งมีชื่อเรียกว่า รีจิสเตอร์ RO-R7 ดังตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดเดรส	รีจีสเตอร์แบงก์	ชื่อรีจีสเตอร์ใช้งาน
00H-07H	0	R0-R7
08H-0FH	1	R0-R7
10H-17H	2	R0-R7
18H-1FH	3	R0-R7

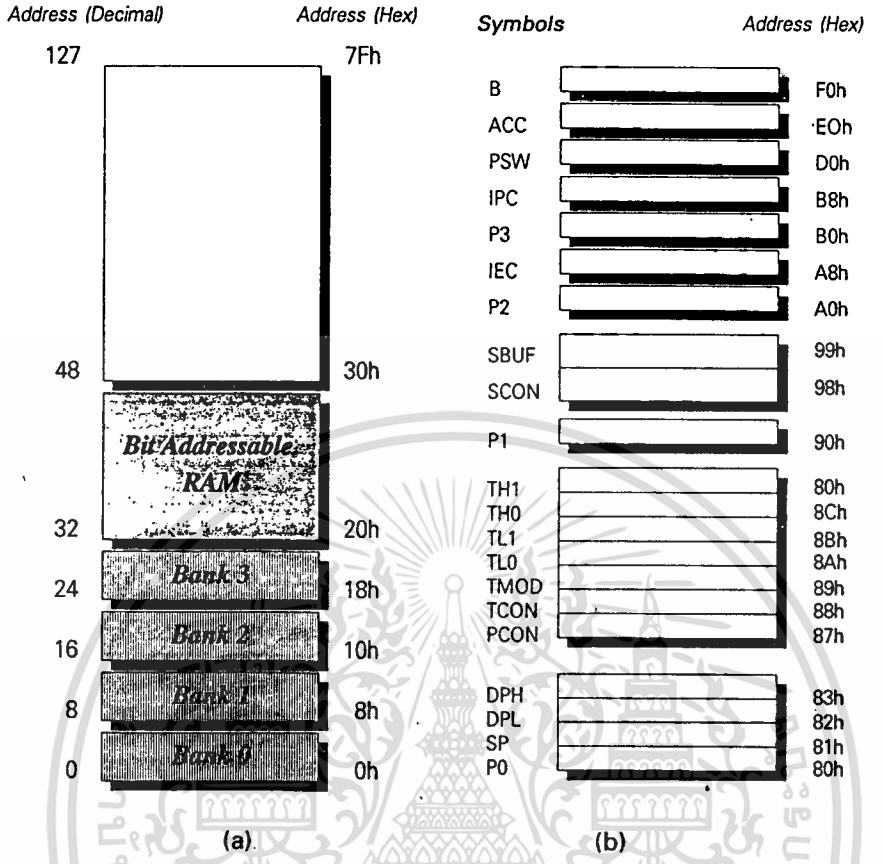
ตารางที่ 2 บริเวณแอดเดรส 00H - 1FH

จะเห็นได้ว่าชื่อของรีจิสเตอร์ไม่ว่าจะอยู่ในรีจิสเตอร์แบงก์ใด ก็จะมีชื่อ RO ถึง R7 เหมือนกันทั้งสิ้น (ดูรูปที่ 7) ดังนั้นในการใช้งานผู้ใช้จะต้องให้ความระมัดระวังว่าต้องการรีจิสเตอร์นั้น ๆ จากแบงก์ใด การสวิตช์เลือกแต่ละกลุ่มของรีจิสเตอร์นี้ก็ทำได้ง่าย เพียงการกำหนดค่าของบิตที่อยู่ภายในรีจิสเตอร์ PSW เท่านั้นตามตารางต่อไปนี้

รีจีสเตอร์	บิต RS0	บิต RS1	ตำแหน่งหน่วยความจำ
แบงก์ 0	0	0	0000H
แบงก์ 1	0	1	0008H
แบงก 2	1	0	0010H
แบงก์ 3	1	1	0018H

ตารางที่ 3 บริเวณแอดเดรสของกลุ่มรีจิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7 การจัดพื้นที่หน่วยความจำข้อมูล (a) ช่วงตั้งแต่แอดเดรส และ (b) ช่วงแอดเดรสซึ่งกำหนดให้เป็นพื้นที่ของรีจิสเตอร์หน้าที่พิเศษ (Special function register)

อย่างไรก็ตามโดยทั่วไปก็มักจะมีการใช้งานรีจิสเตอร์ R0-R7 เฉพาะในแบงก์ O เท่านั้น ดังนั้นพื้นที่ของแบงก์อื่น ๆ ที่เหลือก็สามารถนำมาใช้ในลักษณะของหน่วยความจำข้อมูลภายในปกติด้วยการอ้างถึงหมายเลขของแอดเดรสนั้น ๆ โดยตรง

บริเวณแอดเดรส 20H - 2FH จำนวน 16 ไบต์ บริเวณพื้นที่เป็นส่วนสำหรับผู้ใช้ซึ่งจะมีความพิเศษต่างไปจากหน่วยความจำส่วนอื่น ๆ เนื่องจากผู้ใช้อาจสามารถอ้างถึงหน่วยความจำบริเวณนี้ได้ทั้งในลักษณะของ ไบต์ข้อมูล เช่นปกติ หรืออาจจะเป็น บิตข้อมูลได้โดยตรงดังนั้นหากว่ามองในลักษณะบิตข้อมูลแล้ว ก็จะมีพื้นที่ตัวแปรแบบบิตให้ใช้งานได้มากถึง 128บิต โดยตำแหน่งแรกของบิตจะเป็นบิตซึ่งเริ่มต้นนับจากบิตนัยสำคัญต่ำสุด (LSB) ของแอดเดรส 20 H เรื่อยไปจนกระทั่งถึงบิตที่ 127 ซึ่งเป็นบิตนัยสำคัญสูงสุด (MSB) ของแอดเดรส 2FH (ดูรูปที่ 7)

ความสามารถในการใช้งานพื้นที่ส่วนนี้แบบบิตข้อมูลโดยตรงนี้นับว่าน่าสนใจมาก และถือเป็นการใช้งาน 8051 อย่างเต็มประสิทธิภาพที่เดียว เนื่องจากว่า 8051 ได้รับการออกแบบมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับงานควบคุมเป็นพื้นฐานอยู่แล้ว ซึ่งส่วนมากงานลักษณะเช่นนี้หากเป็นการนำเข้าข้อมูลก็มักจะเป็นเพียงการอ่านคำสถานะลอจิกของเส้นสัญญาณ หรือกรณีการส่งออกข้อมูลก็จะเป็นการกำหนดสภาวะลอจิกให้กับวงจรภายนอกผ่านทางบิตใดบิตหนึ่งอยู่แล้ว ดังนั้นหากว่ามีการกำหนดบิตหรืออ่านค่าของบิตมาโดยตรง แทนที่จะต้องทำลอจิกขั้นต้นกับข้อมูลทั้ง ไบต์เพื่อต้องการทราบผลเพียงหนึ่งบิตเช่นที่กระทำกันในโปรเซสเซอร์โดยทั่วไป ก็จะเพิ่มความสะดวกและรวดเร็วในการเขียนโปรแกรมควบคุมมาก รายละเอียดในส่วนนี้จะได้กล่าวถึงอีกครั้งหนึ่งเมื่อศึกษาถึงการใช้งานพอร์ตอินพุต/เอาต์พุตต่อไป

บริเวณแอดเดรส 30 H - 7FH เป็นบริเวามารนาถไปใช้งานได้อย่างอิสระ โดยสามารถอ้างถึงได้เฉพาะในลักษณะของไบต์ข้อมูลตามปกติเท่านั้น

2.3.3.1.ข หน่วยความจำขนาด 128 ไบต์ถัดไป

พื้นที่ตั้งแต่บริเวณตั้งแต่แอดเดรส 80H-FFH เป็นบริเวณของหน่วยความจำที่มีการใช้งานเฉพาะจาก 8051 เท่านั้น โดยจะนำมาใช้เป็นตำแหน่งของ รีจิสเตอร์หน้าที่พิเศษ Special Function Register หรือ SFR) จำนวน 20 ตำแหน่ง ดังแสดงเป็นแผนภาพในรูปที่ 4

สำหรับไมโครคอนโทรลเลอร์เบอร์ 8051 จะมีหน่วยความจำข้อมูลภายในสำหรับการใช้งานเพิ่มมากขึ้นกว่าเบอร์อื่น ๆ เช่น 8031 หรือ 8751 อีก 128 ไบต์ โดยจะอยู่บริเวณช่วงแอดเดรส 80 H ถึง FFH เช่นกัน ซึ่งแม้ว่าจะเป็นพื้นที่ที่มีหมายเลขแอดเดรสเดียวกับส่วนของรีจิสเตอร์หน้าที่พิเศษแต่ในความเป็นจริงแล้วจะเป็นพื้นที่หน่วยความจำอีกบริเวณหนึ่ง ซึ่งมีการซ้อนเกย (Overlap) กันให้อยู่ในบริเวณแอดเดรสส่วนนี้ ซึ่งหากว่าผู้ใช้งานต้องการจะเก็บข้อมูลในพื้นที่บริเวณนี้ก็จะต้องใช้การอ้างถึงหน่วยความจำแบบโดยอ้อม (Indirect Addressing) เท่านั้น

2.3.3.1.ค รีจิสเตอร์หน้าที่พิเศษ

รีจิสเตอร์หน้าที่พิเศษ (SFR) เป็นรีจีสเตอร์สำหรับการควบคุมหน้าที่และ การทำงานของอุปกรณ์ หรือพอร์ตของ 8051 ทั้งหมด โดยมีตำแหน่งอยู่ในบริเวณแอดเดรส 80H-FFH (อ้างถึงรูปที่ 3.4) การใช้งานรีจิสเตอร์หน้าที่พิเศษเหล่านี้ สามารถทำได้ทางการระบุถึงชื่อของรีจิสเตอร์หรือตำแหน่งแอดเดรสที่เป็นของรีจิสเตอร์นั้นก็ได้

ตารางต่อไปนี้แสดงเห็นลักษณะการจัดพื้นที่หน่วยความจำ สำหรับรีจิสเตอร์หน้าที่พิเศษเหล่านี้โดยมีข้อสังเกตว่า รีจิสเตอร์ที่อยู่ใน ตำแหน่งแอดเดรสที่เป็นจำนวนทวีคูณของค่า 8 จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถอ้างถึงในระดับบิตได้ด้วย คือแอดเดรส 80H ,88H ,90H ,98H ,A0H ,B0H ,B8H ,D0H ,E0H และ F0H

ชื่อรีจีสเตอร์	คำจำกัดความ	การอ้างถึงแบบบิต
ACC	Accumulator	ได้
B	B register	ได้
PSW	Program Status Word	ได้
SP	Stack Pointer	ได้
DPTR	Data Pointer (DPH & DPL)	ได้
P0	Port 0	ได้
P1	Port 1	ได้
P2	Port 2	ได้
P3	Port 3	ได้
IP	Interrupt Priority	ได้
IE	Interrupt Enable	ได้
TMOD	Timer / Counter mode	ได้
TCON	Timer / Counter control	ไม่ได้
TH0	Timer / Counter 0 (high)	ไม่ได้
TL0	Timer / Counter 0 (low)	ไม่ได้
TH1	Timer / Counter 1 (high)	ไม่ได้
TL1	Timer / Counter 1 (low)	ไม่ได้
SCON	Serial control	ไม่ได้
SBUF	Serial data buffer	ไม่ได้
PCON	Power control	ไม่ได้

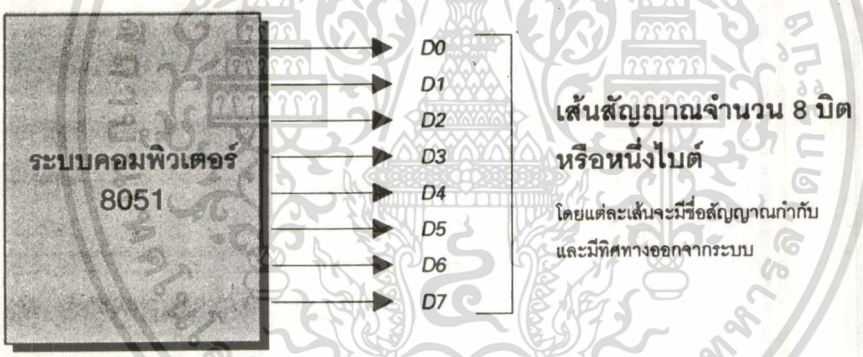
ตารางที่ 4 รีจิสเตอร์ใช้งานพิเศษ (Special Function หรือ SFR)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

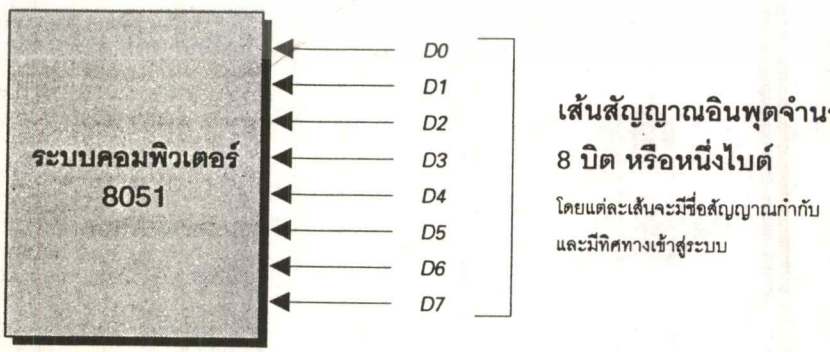
2.3.4 พอร์ตอินพุตและเอาต์พุต

พอร์ต มีความหมายถึงแอดเดรสหนึ่งที่ได้รับการกำหนดไว้เพื่อการโอนย้ายข้อมูลระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก การกำหนดประเภทของการติดต่อขึ้นอยู่กับทิศทางการไหลของข้อมูลเมื่อพิจารณาจากไมโครคอนโทรลเลอร์เป็นหลัก (ดูรูปที่ 8) ดังนั้นการนำเข้าข้อมูลจากวงจรภายนอกจึงเรียกว่า การอินพุต (input) และในกรณีตรงกันข้ามเพื่อส่งออกข้อมูลก็จะเรียกว่า การเอาต์พุต (output)

เมื่อพิจารณาถึงวิธีการส่งข้อมูลภายในพอร์ตจะสามารถแยกประเภทของพอร์ตออกได้เป็นสองลักษณะ คือ พอร์ตแบบขนาน (Parallel port) ซึ่งทำการส่งจำนวนบิตข้อมูลทั้งหมดออกมาหรือนำเข้าไปพร้อมกันในคราวเดียว



รูปที่ 8 (a) แผนภาพแสดงการส่งข้อมูลผ่านทางพอร์ตอนุกรม ซึ่งเป็นเส้นจำนวน 8 เส้น สำหรับส่งข้อมูลจากระบบไปให้กับอุปกรณ์ภายนอก



รูปที่ 8 (b) แผนภาพแสดงการส่งข้อมูลจากอุปกรณ์ภายนอกเข้ามายังพอร์ตอินพุตของระบบ ซึ่งมีลักษณะเป็นเส้นสัญญาณจำนวน 8 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4.ก พอร์ตแบบขนานของ 8051

8051 มีโครงสร้างของพอร์ตที่สามารถใช้งานแบบขนานได้จำนวนทั้งหมดสี่พอร์ต เรียกชื่อเรียงตามลำดับว่า พอร์ต 0,1,2, และ 3 และเป็นพอร์ตขนาด 8 บิตทั้งหมด การใช้งานพอร์ตสามารถทำได้ทั้งในลักษณะของเส้นสัญญาณเดี่ยว ๆ หรือกลุ่มของสัญญาณได้ นอกจากนี้พอร์ต 0,2 และ 3 ยังสามารถนำไปใช้งานอื่น ๆ ที่ไม่ใช่เป็นพอร์ตอินพุต/เอาต์พุตได้ โดยพอร์ต 0 จะทำหน้าที่มัลติเพล็กซ์ ระหว่างบัสแอดเดรสไบต์ต่ำและบัสข้อมูลสำหรับการติดต่อกับวงจรประกอบร่วมกับข้อมูลบัสแอดเดรสไบต์สูงซึ่งจะส่งออกมาทางพอรต์ 2 ดังได้อธิบายในบทที่ผ่านมา สำหรับพอร์ต 3 นั้น นอกเหนือไปจากความสามารถเช่นพอร์ตปกติแล้ว สามารถนำไปเป็นขาสัญญาณของการอินเตอร์รัปต์ต่างๆ ซึ่งรวมทั้งการสร้างสัญญานควบคุม RD/ และ WR/ เพื่อทำหน้าที่อ่านหรือเขียนหน่วยความจำข้อมูลภายนอกด้วย การใช้งานพอร์ตลักษณะงานแบบอื่น ๆ ที่ไม่ใช่เป็นพอร์ตแบบอินพุต/เอาต์พุต นี้จะดำเนินการโดย 8051 เองโดยอัตโนมัติ

2.3.4.ข โครงสร้างการทำงานของพอร์ต 8051

จากลักษณะโครงสร้างของแต่ละบิตภายในพอร์ตทั้งหมดของ 8051 ซึ่งได้แสดงไว้ในรูปที่ 9 นั้น จะเห็นว่ามีความคล้ายคลึงกันตามลักษณะโครงสร้างที่เรียกว่า Quasi-bidirectional port ยกเว้น พอร์ต 0 ซึ่งเพียงแต่ไม่มีตัวต้านทานทำหน้าที่ Pull-up สัญญาณไว้ภายในเท่านั้น วงจรประกอบอื่นภายในยังมีฟลิปฟลอป แบบ D ซึ่งมีผลทำให้พอร์ตสามารถแลตช์หรือค้างสภาวะของสัญญาณได้ นอกจากนี้ในส่วนเอาต์พุตของฟลิปฟลอปเฉพาะของพอร์ต 0 และพอร์ต 2 จะมีโครงสร้างที่ทำหน้าที่คล้ายกับสวิตช์เพิ่มเติมขึ้น เพื่อควบคุมเอาต์พุตนี้ต่อเข้ากับส่วนของทรานซิสเตอร์ในระหว่างที่ไม่ได้มีการทำงาน ในลักษณะของบัสแอดเดรสหรือบัสข้อมูลด้วย สำหรับบัฟเฟอร์จำนวนสองตัวของทุกบิตในพอร์ตนั้นมีการทำงานแยกกันโดยอิสระ โดยตัวที่อยู่ทางด้านบนจะยอมให้สัญญาณผ่านได้ก็ต่อเมื่อการอ่านค่าข้อมูลที่ค้างไว้ ส่วนอีกตัวหนึ่งซึ่งอยู่ทางด้านล่างจะถูกใช้งานเฉพาะเมื่อได้มีการอ่านสถานะของขาสัญญาณเท่านั้น

2.3.4.ค การใช้งานพอร์ตเป็นการอินพุต

การใช้งานพอร์ตเป็นการอินพุตข้อมูลจะต้องเริ่มด้วยการส่งข้อมูลที่มีค่าเป็น 1 ออกมาทางบิตของพอร์ตนั้นก่อนเป็นลำดับแรก เพื่อหยุดการทำงานของทรานซิสเตอร์ที่ทำหน้าที่ขับสัญญาณเอาต์พุตของบิตนั้น ทำให้ขาสัญญาณของบิตถูกต่อเข้ากับตัวต้านทางซึ่งทำหน้าที่ Pull-up ภายในซึ่งมีผลให้บิตนั้น ๆ ของพอร์ต 1,2 และ 3 เป็นสภาวะของลอจิกสูงตัวต้านทานนี้มีค่าประมาณ 50 K Ω ซึ่งเป็นค่าที่สูงมาก และทำให้อุปกรณ์ภายนอกสามารถขับสัญลักษณ์ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตเหล่านี้เป็นลอจิกต่ำได้ง่าย สำรหับบิตของพอร์ต 0 นั้น แม้นว่าจะมีหลักการทำงานที่คล้ายคลึงกันกับบิตของพอร์ตอื่น ๆ แต่เนื่องจากการที่ไม่มีตัวต้านทานทำหน้าที่ ๆ Pull - up ภายในไว้ ทำให้เมื่อทรานซิสเตอร์ที่ทำหน้าที่ขับสัญญาณเอาต์พุตนั้นหยุดการทำงาน ก็จะเป็นผลให้ขาสัญญาณนี้อยู่ในสภาวะอิมพีแดนซ์สูงแทน

2.3.4.ง การใช้งานพอร์ตเป็นการเอาต์พุต

เมื่อมีการส่งข้อมูลที่มีค่าเป็น 0 ให้กับแต่ละบิตของพอร์ตทุกพอร์ต ข้อมูลนี้จะถูกส่งให้กับฟลิปฟลอป ซึ่งจะค้างค่านี้ไว้ และมีผลทำให้ทรานซิสเตอร์ที่ทำหน้าที่ขับสัญญาณเอาต์พุตนั้นทำงานดังนั้นขาสัญญาณก็จะมีสภาวะลอจิกเป็นลอจิกต่ำด้วย

ส่วนการส่งข้อมูลที่มีค่าเป็น 1 ออกมานั้น ในกรณีที่เป็นการทำงานแต่ละบิตของพอร์ต 1 ,2 หรือ 3 จะทำให้ทรานซิสเตอร์ที่ทำหน้าที่ขับสัญญาณเอาต์พุตนั้นหยุดการทำงาน มีผลทำให้ขาของสัญญาณเป็นลอจิกสูงด้วยตัวต้านทานที่ Pull-up อยู่ภายในนั้น แต่สำหรับการทำงานในแต่ละบิตทางพอร์ต 0 นั้นจะมีผลที่แตกต่างออกไป โดยขาสัญญาณจะเป็นสภาวะอิมพีแดนซ์สูงแทน เนื่องจากไม่มีตัวต้านทานภายในเชื่อมต่อยู่นั่นเอง ดังนั้นในการใช้งานพอร์ต 0 เป็นการเอาต์พุตข้อมูล จึงจำเป็นต้องใช้ตัวต้านทางภายนอก Pull- Up สัญญาณไว้กับลอจิกสูงแทน

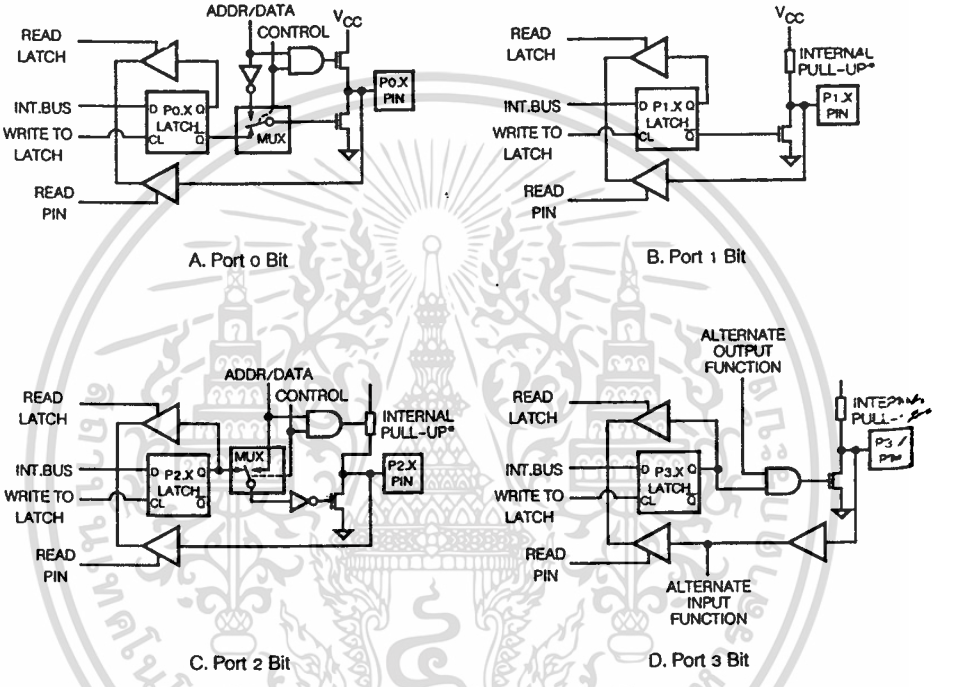
ความสามารถอีกประการหนึ่งเกี่ยวกับพอร์ตอินพุต/เอาต์พุตของ 8051 เป็นวิธีการอ่านค่าลอจิกจากพอร์ตซึ่งมีได้สองวิธี คือการอ่านค่าลอจิกที่ขาสัญญาณ (Port pin) และการอ่านค่าลอจิกของการแลตซ์ที่พอร์ต (Port latch) ดังจะสังเกตได้จากรูปที่ 9 วิธีการอ่านค่าจากพอร์ตทั้งสองแบบนี้จะช่วยให้ระบบทำงานได้ด้วยความถูกต้องมากยิ่งขึ้น ยกตัวอย่างเช่น หากว่า พอร์ตถูกนำไปต่อกับขาเบสของทรานซิสเตอร์แบบ NPN และขาอิมิตเตอร์ต่อกับกราวด์
ของระบบเมื่อมีการส่งค่า 1 ออกไปจะมีผลทำให้ทรานซิสเตอร์ทำงาน ในขณะนั้นถ้าซีพียูมีการอ่านค่าลอจิกจากขาสัญญาณของพอร์ตนี้ก็จะได้ค่าลอจิกต่ำ เนื่องจากมองเห็นค่าศักย์ไฟฟ้าระหว่างขาเบสและขาอิมิตเตอร์ซึ่งมีค่าประมาณ 0.6 โวลต์แทน ดังนั้นในกรณีเช่นนี้หากว่าเป็นการอ่านค่าจากลอจิกของการแลตซ์ ก็จะได้รับค่าระดับลอจิกสูงซึ่งเป็นค่าที่ถูกต้องสภาพที่เป็นจริง

2.3.4.จ ลักษณะสมบัติของพอร์ตอินพุต/เอาต์พุต

ดังได้กล่าวแล้วว่าพอร์ต 1,2 และ 3 ของ 8051 มีตัวต้านทาน (ซึ่งสร้างขึ้นจาก FET) ทำหน้าที่ Pull-up ขาสัญญาณไว้และมีค่าประมาณ 50 K Ω ซึ่งถือว่ามีค่าที่สูงมาก เป็นผลให้การเปลี่ยนแปลงระดับสัญญาณลอจิกจากสูงไปต่ำทำได้อย่างรวดเร็ว แต่ในกรณีตรงข้ามจะใช้เวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้อยมาก ดังนั้นในการแก้ปัญหาจึงได้มีการออกแบบตัวต้านทางเพิ่มขึ้นอีกหนึ่งตัวขนานไว้โดยมีค่าประมาณ 1 K Ω เรียกว่า Speed-up resistor ซึ่ง ยอมให้กระแสไหลผ่านได้มากขึ้นประมาณ 50 - 100 เท่า และจะมีการเชื่อมต่อตัวต้านทานที่เพิ่มขึ้นนี้เฉพาะเมื่อมีการเปลี่ยนระดับสัญญาณจากลอจิกต่ำไปเป็นลอจิกสูงเท่านั้น โดยใช้เวลาประมาณ 2 คล็อกไซเคิล



รูปที่ 9 โครงสร้างของแต่ละบิตภายในพอร์ตอินพุต/เอาท์พุตของ8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038048

2.3.5 การอินเตอร์รัปต์

การอินเตอร์รัปต์เป็นการขัดจังหวะการทำงานอย่างใดอย่างหนึ่งซึ่งกำลังดำเนินอยู่

2.3.5.ก ประเภทของการอินเตอร์รัปต์

ไมโครคอนโทรลเลอร์ 8051 สามารถเกิดการอินเตอร์รัปต์ โดยจำแนกตามแหล่งที่มาของสัญญาณ (Signal Source) ของสัญญานอินเตอร์รัปต์นั้น ๆ ได้แก่

- สัญญาณอินเตอร์รัปต์จากภายนอก (External Interrupt) การตรวจสอบสัญญาณที่เข้ามาอินเตอร์รัปต์นี้ จะสามารถกำหนดให้มีการตรวจสอบในลักษณะเมื่อได้มีการเปลี่ยนแปลงระดับสัญญาณ (Level - sensitive) ไปแล้ว หรือในช่วงเวลาขณะเริ่มมีการเปลี่ยนแปลงสัญญาน จากลอจิกสูงไปเป็นลอจิกต่ำ (Edge-sensitive)
- สัญญาณอินเตอร์รัปต์ภายใน (Internal Interrupt) แหล่งกำเนิดของสัญญาณนี้จะเป็นวงจรภายในของไมโครคอนโทรลเลอร์เอง

2.3.5.ข โครงสร้างการอินเตอร์รัปต์

สัญญาณที่เข้ามาทำการอินเตอร์รัปต์ 8051 เกิดขึ้นได้ 5 ลักษณะคือ

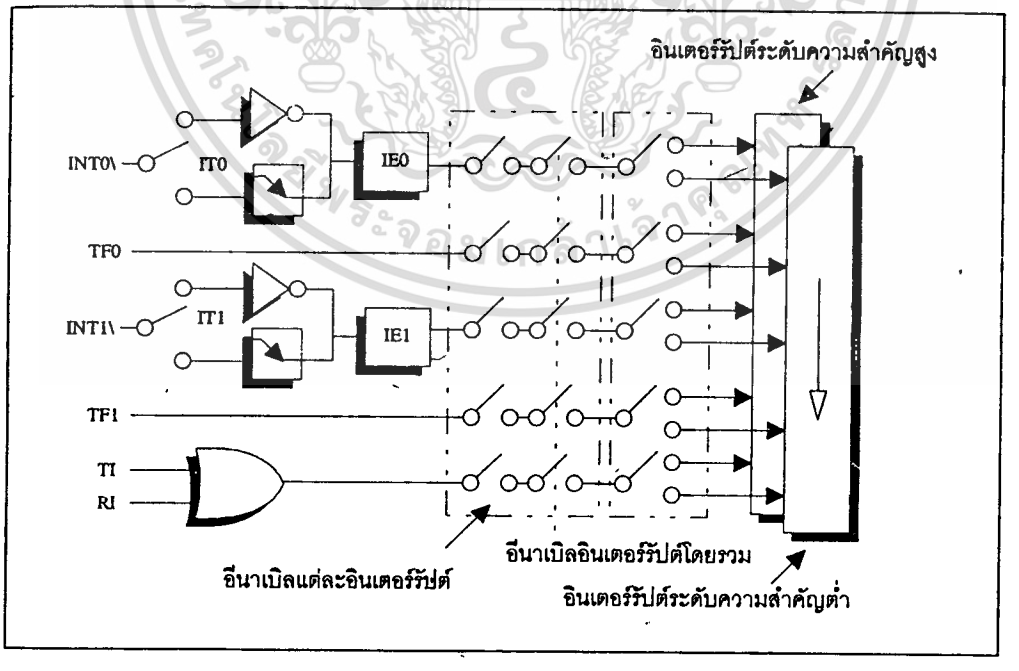
สัญญาณ	ความหมาย
INTO	สัญญาณอินเตอร์รัปต์จากภายนอกทางขาสัญญาณ P3.2 โดย 8051 จะทำการ สุ่มตัวอย่างเมื่อสิ้นสุดทุกแมชชีนไซเคิล
INT1	สัญญาณอินเตอร์รัปต์จากภายนอกทางขาสัญญาณ P3.3 โดย 8051 จะทำการ สุ่มตัวอย่างเมื่อสิ้นสุดทุกแมชชีนไซเคิล
Timer0	สัญญาณการเกิดโอเวอร์โฟลว์ของ Timer 0
Timer1	สัญญาณการเกิดโอเวอร์โฟลว์ของ Timer 1
พอร์ตอนุกรม	การเกิดอินเตอร์รัปต์ที่เกิดขึ้นจากการรับ/ส่งข้อมูลอนุกรม ทำให้มีผลต่อแฟล็กอินเตอร์รัปต์ RI และ TI ตามลำดับ

ตารางที่ 5 แสดงสัญญาณอินเตอร์รัปต์ ทั้ง 5 แหล่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากแผนภาพโครงสร้างระบบอินเตอร์รัปต์ของ 8051 ในรูปที่ 10 จะเห็นว่าเมื่อเกิดการอินเตอร์รัปต์สัญญาณต่างๆ ขึ้น จะส่งผลให้มีการควบคุมเพื่อสั่งให้โปรเซสเซอร์กระโดดไปทำงานที่ตำแหน่งแอดเดรสต่าง ๆ ตามประเภทของแหล่งกำเนิดสัญญาณอินเตอร์รัปต์ที่เกิดขึ้น ซึ่งปกติแล้วควรจะต้องมีการสร้างโปรแกรมที่ตำแหน่งเหล่านี้ไว้ เพื่อทำหน้าที่เป็นโปรแกรมย่อยบริการอินเตอร์รัปต์

การกำหนดให้ 8051สามารถตอบรับการอินเตอร์รัปต์แต่ละประเภท ทำได้โดยการกำหนดบิตของข้อมูลที่เกี่ยวข้อง ซึ่งมักจะอยู่ภายในรีจิสเตอร์ TCON และ SCON หากว่าได้มีการกำหนดบิตของข้อมูลที่เกี่ยวข้อง ซึ่งมักจะอยู่ภายในรีจิสเตอร์ IE (Interrupt Enable Register) ด้วยแล้ว ก็จะสามารถตอบรับการอินเตอร์รัปต์ของสัญญาณนั้น ๆ ได้ นอกจากนี้แล้วตามแผนภาพในรูปที่ 10 ยังแสดงให้เห็นว่าสัญญาณอินเตอร์รัปต์แต่ละประเภท ยังสามารถกำหนดระดับความสำคัญ (Priority) ของการอินเตอรัปต์ได้สองลักษณะ คือระดับความสำคัญสูงหรือต่ำ (Higher Low Priority) กล่าวคือ ขณะที่กำลังประมวลผลอยู่ภายในส่วนของโปรแกรมย่อยบริการอินเตอร์รัปต์ของสัญญาณที่มีระดับความสำคัญต่ำอยู่ ก็อาจจะถูกขัดจังหวะให้ไปประมวลผลของสัญญาณที่มีระดับความสำคัญสูงกว่าได้ แต่หากว่าเป็นสัญญาณอินเตอร์รัปต์ที่มีระดับความสำคัญต่ำเช่นเดียวกันแล้วก็จะต้องรอให้เสร็จสิ้นการประมวลผลที่ดำเนินการอยู่ก่อน



รูปที่ 10 แผนภาพแสดงโครงสร้างระบบอินเตอร์รัปต์ของ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.5.ค การควบคุมอินเตอร์รัปต์

ตามโครงสร้างด้านการจัดการอินเตอร์รัปต์ของ 8051 สามารถกำหนดเลือกเพื่อยินยอมหรือไม่ยินยอม (Enable/Disable) ให้มีการอินเตอร์รัปต์ของแต่ละสัญญาณได้ โดยใช้วิธีการกำหนดค่าของบิตภายในรีจิสเตอร์ IE ซึ่งจะมีทั้งแบบที่ระบุถึงอินเตอร์รัปต์โดยรวมทั้งหมด (บิตที่ 7) และอินเตอร์รัปต์แต่ละประเภทได้ ในกรณีที่กำหนดค่าข้อมูลเป็น 1 ให้กับบิตจะมีความหมายถึงการยินยอมให้มีการอินเตอร์รัปต์เกิดขึ้นได้ และจะเป็นกรณีตรงข้ามกันสำหรับการกำหนดค่าข้อมูลที่เป็น 0 หากลองย้อนกลับไปพิจารณาแผนภาพในรูปที่ 5.1 อีก ครั้ง จะเห็นว่าจะต้องทำการกำหนดให้ยินยอมการอินเตอร์รัปต์ทั้งหมดให้เกิดขึ้นก่อน จึงจะมีผลทำให้การกำหนดบิตเพื่อยินยอมของแต่ละอินเตอร์รัปต์มีผลขึ้นได้

ชื่อบิต:IE ตำแหน่ง: A8h ค่ากำหนดเริ่มต้น: 0x00 0000

EA	-	ET2	ES	ET1	EX1	ET0	EX0

ชื่อบิต	ตำแหน่ง	ความหมาย
EA	IE.7	อีนาเบิล/ดิสเอเบิลการเกิดอินเตอร์รัปต์โดยรวม
-	IE.6	-
ET2	IE.5	อีนาเบิล/ดิสเอเบิบการเกิดอินเตอร์รัปต์ Timer2
ES	IE.4	อีนาเบิล/ดิสเอเบิลการเกิดอินเตอร์รัปต์พอร์อนุกรม
ET1	IE.3	อีนาเบิล/ดิสเอเบิลการเกิดอินเตอร์รัปต์ Timer1
EX1	IE.2	อีนาเบิล/ดิสเอเบิลการเกิดอินเตอร์รัปต์ INT
ET0	IE.1	อีนาเบิล/ดิสเอเบิลการเกิดอินเตอร์รัปต์ Timer0
EX0	IE.0	อีนาเบิล/ดิสเอเบิลการเกิดอินเตอร์รัปต์ INT0

ตารางที่ 6 บิตต่าง ๆ ของการอนุญาตการอินเตอร์รัปต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.5.ง ระดับความสำคัญของการอินเตอร์รัปต์

การกำหนดระดับความสำคัญให้กับสัญญาณอินเตอร์รัปต์แต่ละประเภทนั้น สามารถทำได้โดยการกำหนดข้อมูลที่มีค่าเป็น 1 หรือ 0 ให้กับบิตภายในรีจิสเตอร์ IP (Interrupt Priority) ดังแสดงในตารางของรูปที่ 7 โดยหากว่ามีค่าเป็น 1 ก็จะทำให้สัญญาณอินเตอร์รัปต์นั้น ๆ มีระดับความสำคัญสูง และในกรณีตรงข้ามกันสำหรับการกำหนดค่าเป็น 0

กรณีที่สัญญาณที่เข้ามาอินเตอร์รัปต์มีระดับความสำคัญเดียวกันเกิดขึ้นพร้อมกัน ก็อาจจะทำให้เกิดปัญหาขึ้นได้แต่อย่างไรก็ตาม 8051 ก็มีโครงสร้างทางด้านฮาร์ดแวร์ในการพิจารณาตามลำดับของตารางดังต่อไปนี้

ระดับความสำคัญ	สัญญาณ	ความหมาย
1	IEO	อินเตอร์รัปต์ภายนอก 0
2	TFO	วงจรนับ/จับเวลา0
3	IE1	อินเตอร์รัปต์ภายนอก 1
4	TF1	วงจรนับ/จับเวลา1
5	RI หรือ TI	วงจรรับ/ส่งข้อมูลอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อบิต:IP ตำแหน่ง:B8h ค่าบิตเริ่มต้น: 00000000

ชื่อบิต	ตำแหน่ง	ความหมาย
-	IP.7	-
-	IP.6	-
PT2	IP.5	ระดับความสำคัญของTimer2
PS	IP.4	ระดับความสำคัญของพอร์ตอนุกรม
PT1	IP.3	ระดับความสำคัญของ Timer1
PX1	IP.2	ระดับความสำคัญของ Timer0
PT0	IP.1	สำคัญของ ระดับความINT1
PX0	IP.0	สำคัญของ ระดับความINT0

ตารางที่ 7 บิตต่าง ๆ ของการจัดระดับความสำคัญของการอินเตอร์รัปต์

2.3.5.จ การจัดการอินเตอร์รัปต์

เมื่อมีการอินเตอร์รัปต์เกิดขึ้น ไมโครคอนโทรลเลอร์จะทำคำสั่งที่กำลังดำเนินการอยู่ให้แล้วเสร็จ จากนั้นจึงจะทำการเก็บค่าตำแหน่งแอดเดรสของคำสั่งที่จะงานต่อไปไว้ยังบริเวณของหน่วยความจำที่ถูกกำหนดไว้ให้เป็นสแต็ก (Stack) และกระโดดไปยังตำแหน่งแอดเดรสที่ได้มีการกำหนดไว้แน่นอนตำแหน่งหนึ่งโดยอัตโนมัติ (ดูตารางที่ 8) ตำแหน่งนี้เรียกว่า แอดเดรสของอินเตอร์รัปต์เวเตอร์ (Interrupt Vector Address) ซึ่งผู้ใช้จะต้องทำการเขียนโปรแกรมย่อย (Subroutine)ยังตำแหน่งแอดเดรสเหล่านี้ ซึ่งเรียกว่า โปรแกรมย่อยบริการอินเตอร์รัปต์ (Interrupt Service) ตำแหน่งของแอดเดรสเหล่านี้ ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แหล่งกำเนิดสัญญาณ	สัญญาณ	ตำแหน่งแอดเดรส (Hex)
IE0	อินเตอร์รัปต์ภายนอก 0	0003
TF0	วงจรนับ/จับเวลา 0	000B
IE1	อินเตอร์รัปต์ภายนอก 1	0013
TF1	วงจรนับ/จับเวลา 1	001B
RI หรือ TI	วงจรรับ/ส่งข้อมูลอนุกรม	0023

ตารางที่ 8 แสดงตำแหน่งของการอินเตอร์รัปต์

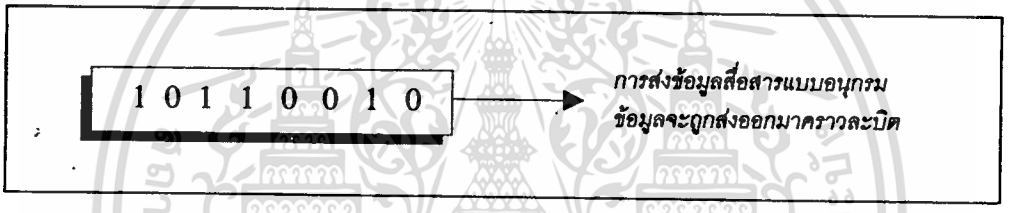
สิ่งที่ควรให้ความสนใจในการเขียนโปรแกรมย่อยบริการอินเตอร์รัปต์ คือ

- ส่วนเริ่มต้นของโปรแกรมย่อย ควรจะมีการเก็บค่าของรีจิสเตอร์หรือแฟล็กสถานะต่างๆ ที่จะต้องนำไปใช้ภายในโปรแกรมย่อย มิฉะนั้นอาจจะมีผลทำให้โปรแกรมปกติที่ทำอยู่ก่อนหน้าการมาทำงานโปรแกรมยอ่ยตอบสนองอินเตอร์รัปต์ทำงานผิดพลาดไปได้
- บรรทัดสุดท้ายของโปรแกรมย่อยจะต้องสิ้นสุดด้วยคำสั่ง RETI (return from interrupt) เพื่อสั่งให้มีการนำค่าที่ได้เก็บไว้ก่อนหน้าการกระโดดมายังโปรแกรมย่อยบริ กรอิ นเตอร์รัปต์นี้ออกจากสแต็กและกลับไปทำงานเดิมต่อไป นอกจากนี้แล้วยังมีผลทำให้แฟล็กสถานนะที่เกี่ยวข้องกับการอินเตอร์รัปต์นั้น ๆ ถูกรีเซตกลับไปเป็นค่าปกติเพื่อรอรับการอินเตอร์รัปต์ครั้งใหม่ ต่อไปด้วย

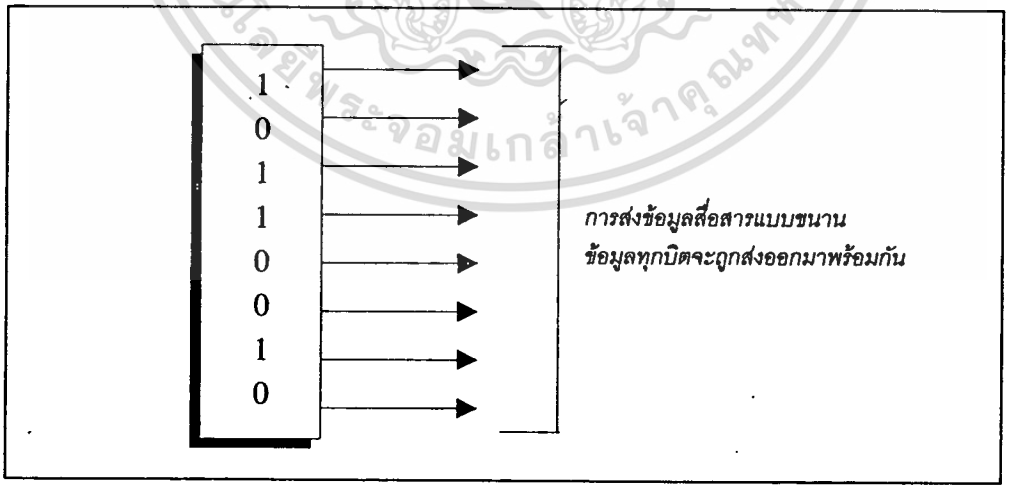
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6 การสื่อสารข้อมูลอนุกรม

การสื่อสารข้อมูลอนุกรมเป็นการรับหรือส่งข้อมูลในลักษณะของบิตหรือกลุ่มของบิต คราวละหนึ่งบิตเป็นลำดับเรื่อยไปจนสิ้นสุด การสื่อสารแบบนี้จะมีข้อแตกต่างจากการสื่อสารแบบขนานเป็นอย่างมาก เนื่องจากข้อมูลมีการโอนย้ายมาพร้อมกันจึงมีความจำเป็นต้องใช้จำนวนเส้นสัญญาณมากขึ้นตามจำนวนบิตของข้อมูลด้วย ในขณะที่การสื่อสารแบบอนุกรมนั้นต้องการเส้นสัญญาณเพียงสองหรือสามเส้นเท่านั้น ดังนั้นการสื่อสารแบบขนานจึงไม่เหมาะสมในการสื่อสารกับอุปกรณ์ภายนอกเป็นระยะทางไกล ๆ เพราะจะทำให้สิ้นเปลืองค่าใช้จ่ายมาก ลองพิจารณาเปรียบเทียบการสื่อสารทั้งสองประเภทได้จากรูปที่ 11 (A)และ11(B)



รูปที่ 11 (a) ข้อมูลสื่อสารแบบอนุกรม ข้อมูลหนึ่งไบต์จะถูกส่งออกคราวละบิตเป็นลำดับไปจนครบจำนวนทั้ง 8

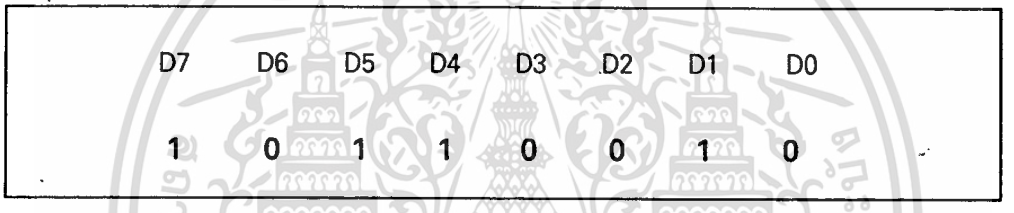


รูปที่ 11 (b) ข้อมูลสื่อสารแบบขนาน ข้อมูลแต่ละบิตภายในหนึ่งไบต์จะถูกส่งออกมาพร้อมกันในลักษณะแบบขนาน

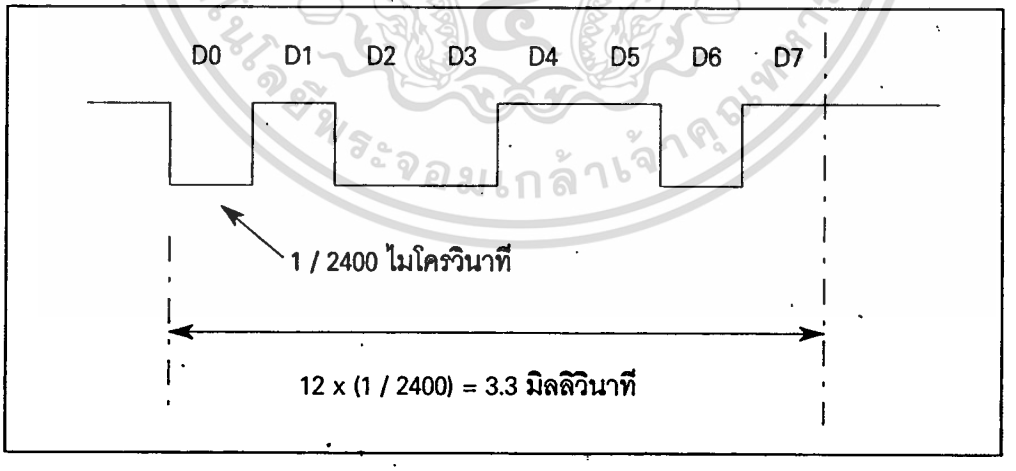
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6.ก จังหวะเวลาของการสื่อสารข้อมูลอนุกรม

เนื่องจากการสื่อสารแบบอนุกรมเป็นการรับ/ส่งข้อมูล ในลักษณะกลุ่มของบิตข้อมูล(Bit stream) ดังนั้นจึงต้องให้ความสนใจในการพิจารณาถึงเรื่องของอัตราความเร็วในการับ/ส่งบิตเหล่านี้เป็นลำดับแรก โดยทั่วไปมักจะระบุกันในหน่วยของจำนวนบิตข้อมูลภายในเวลาหนึ่งวินาทีเรียกว่า อัตราบอด (Baud Rate) ตามค่ามาตรฐานเหล่านี้ ได้แก่ 150,300,1200,2400,4800 และ9600 จากรูปที่ 8.2 แสดงให้เห็นลักษณะของรูปแบบสัญญาณข้อมูลอนุกรมที่ปรากฏในสายส่งสัญญาณข้อมูลทั้ง 8 บิตนี้หากว่าถูกส่งออกมาด้วยอัตรา 2400 บอดจะใช้เวลาในการส่งข้อมูลหนึ่งบิตมีค่าเท่ากับ 1/2400 หรือ 416 ไมโครวินาที และในการส่งข้องมูลทั้ง 8บิตมีค่าเท่ากับ (8X416)หรือ 3,328 ไมโครวินาที



รูปที่ 12 (a) บิตข้อมูลสื่อสารแบบขนานภายในไบต์ข้อมูลจะถูกส่งออกมาในลักษณะแบบอนุกรม



รูปที่ 12 (b) รูปแบบสัญญารไฟฟ้าของข้อมูลในรูป 12 (a) ซึ่งเป็นการส่งข้อมูลแบบอนุกรมด้วยอัตรา 2400 บอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6.ข รูปแบบของข้อมูลอนุกรม

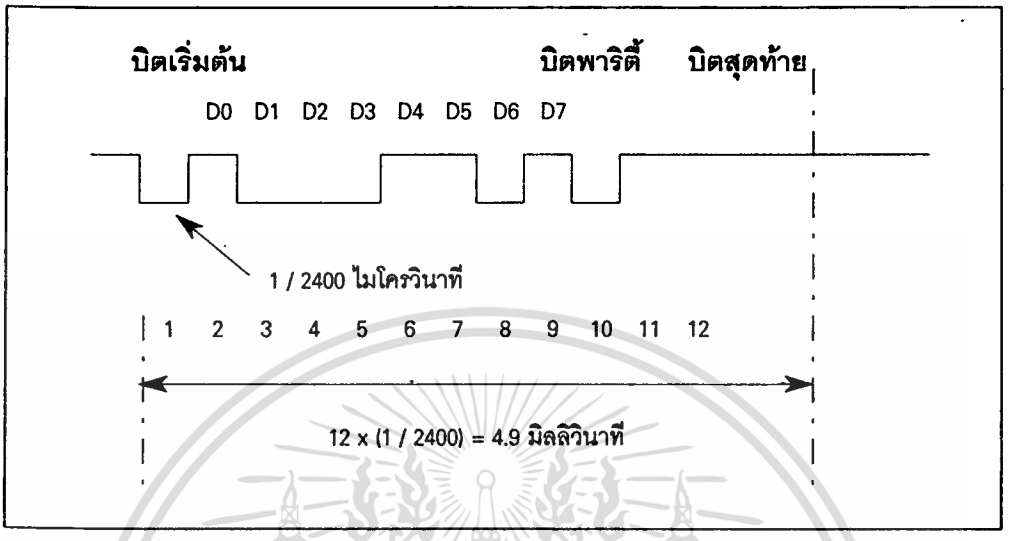
วิธีการที่จะทำให้ข้อมูลสื่อสารอนุกรมมีความถูกต้องมากยิ่งขึ้น จะใช้การเพิ่มเติมบิตข้อมูลบางอย่างร่วมไปกับการส่งข้อมูลจริง ได้แก่

- บิตเริ่มต้น (Start Bit) บิตเริ่มต้นมีหน้าที่สำหรับการบ่งบอกให้วงจรฮาร์ดแวร์ทางด้านรับทราบถึง ตำแหน่งจุดเริ่มต้นของบิตข้อมูลกลุ่มใหม่ เพื่อที่จะทำการปรับจังหวะของสัญญาณการรับข้อมูลให้ตรงกันดังนั้นบิตเริ่มต้นนี้จึงจะถูกเพิ่งเข้าไปก่อนมีการส่งข้อมูลจริง ตามปกติแล้วค่าของบิตเริ่มต้นมักจะเป็นระดับลอจิกที่ตรงข้ามกับระดับลอจิกของสภาวะของสายสื่อสาร ขณะเมื่อไม่มีการส่งข้อมูล (Idle State) ตัวอย่างเช่น หากสภาวะของสายเมื่อไม่มีข้อมูลเป็นลอจิกสูง บิตเริ่มต้นก็จะเป็นระดับลอจิกต่ำ เป็นต้น

- บิตแสดงภาวะความเป็นเลขคู่หรือเลขคี่ (Parity Bit) บิตนี้มีหน้าที่เพื่อการตรวจสอบความถูกต้องของข้อมูล โดยทั่วไปมักเรียกว่า บิตพาริตี้และจะนำไปแทรกต่อท้ายบิตข้อมูล ค่าของบิตนี้ขึ้นอยู่กับจำนวนค่าของบิตข้อมูลที่เป็น 1 ซึ่งจะเป็นได้สองลักษณะ คือ พาริตี้คู่ (Even Parity)หรือพาริตี้คี่(Odd Parity) ระบบที่ติดต่อกัน โดยระบุว่าจะใช้พาริตี้คู่ (Even Parity) ทางด้านส่งจะนำค่าข้อมูลที่จะส่งมาพิจารณาหากจำนวนของบิตที่มีค่า 1 เป็นเลขจำนวนคู่อยู่แล้ว ค่าของบิตพาร์ตี้จะมีค่าเป็น 0 แต่หากว่าจำนวนของบิตที่มีค่า 1 เป็นเลขจำนวนคี่ ค่าของบิตพาริตี้ก็จะมีค่าเป็น 1 การพิจารณาทางด้านรับเป็นการตรวจสอบจำนวนที่มีค่า 1 ของข้อมูลที่ได้รับมาทั้งหมดรวมทั้งบิตพาริตี้ ถ้ามีค่าเป็นเลขจำนวนคู่ แสดงว่าข้อมูลที่รับเข้ามานี้ถูกต้อง แต่หากไม่เป็นเลขจำนวนคู่แสดงว่าเกิดความผิดพลาดของข้อมูลขึ้น เป็นต้น

- บิตสุดท้าย (Stop Bit) บิตสุดท้ายเป็นบิตที่เพิ่มเติมขึ้นเพื่อระบุถึงขอบเขตการสิ้นสุดของกลุ่มบิตข้อมูล บิตสุดท้ายนี้อาจจะมีจำนวนมากกว่าหนึ่งบิตได้ คือ 1 บิต1.5 บิตและ 2 บิต ดังนั้นกรณีของการส่งข้อมูล 8 บิตพร้อมบิตที่เพิ่มเติมเข้าไปโดยสมบูรณ์ คือ บิตเริ่มต้น บิตพาริตี้ และบิตสุดท้าย รวมทั้งสิ้น 12 บิต ตามแผนภาพสัญญาณเวลาในรูปที่ 8.3 หากข้อมูลถูกส่งออกไปด้วยอัตราเร็ว 2400 บอด เวลาโดยรวมในการส่งข้อมูลหนึ่งไบต์จะมีค่าเป็น 12 X 416 ไมโครวินาที หรือ 4.99มิลลิวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 13 แผนภาพสัญญาณเวลาของข้อมูลแบบอนุกรมจำนวน 8 บิต พร้อมด้วยบิตเริ่มต้นพาริตี้และบิตสุดท้าย

2.3.6.ค การจัดการข้อมูลอนุกรมของ 8051

พอร์ตอนุกรมของ 8051 มีโครงสร้างการทำงานในแบบที่เรียกว่า ฟูลดูเพล็กซ์ (Full Duplex) ซึ่งหมายถึงความสามารถในการรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกัน จากรูปที่ 14 แสดงให้เห็นถึงแผนภาพการทำงานอย่างง่ายของวงจรส่วนจัดการข้อมูลอนุกรมของ 8051 โดยทางด้านวงจรของตัวส่ง (Transmitter) ประกอบด้วยรีจิสเตอร์ SBUF ทำหน้าที่เก็บข้อมูลที่จะส่งออก การใช้คำสั่งเขียนหรือโอนย้ายข้อมูลมายังรีจิสเตอร์นี้จะเป็นการส่งข้อมูลนั้นออกไปยังพอร์ตอนุกรมทางขาสัญญาณ TxD (พอร์ต 3.1) โดยอัตโนมัติ ส่วนวงจรด้านตัวรับ (Reciever) ประกอบด้วยรีจิสเตอร์ SBUF เช่นเดียวกัน แต่ทำหน้าที่เก็บข้อมูลที่นำมาจากส่วนของวงจรเลื่อนบิตหรือชิพรีจิสเตอร์ (shift register) ของวงจรจัดการข้อมูลอนุกรมภายใน สัญญาณข้อมูลอนุกรมภายในสัญญาณข้อมูลอนุกรมที่รับเข้าจะผ่านมาทางขาสัญญาณ RxD (พอร์ต 3.0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตอนุกรมของ 8051 สามารถโปรแกรมให้ทำหน้าที่ในรูปแบบต่าง ๆ กันสี่แบบ (หรือเรียกว่าโหมดการทำงาน) โดยการกำหนดค่าบิต SMO และ SMI ซึ่งอยู่ภายในรีจิสเตอร์ควบคุมและบอกสถานะ SCON ดังแสดงในตารางที่ 10 โหมดการทำงานทั้ง 4 แบบของพอร์ตอนุกรม มีดังนี้

โหมดทำงาน	คำอธิบาย
โหมด 0	เป็นการขยายพอร์ตอินพุตเอาต์พุต โดยทำงานร่วมกับไอซีชิพรีจิสเตอร์ภายนอกประเภททีทีแอลหรือซีมอส
โหมด 1	ใช้สำหรับการเชื่อมต่ออนุกรมแบบ UART (Universal asynchronous receiver/ transmitter) โดยการใช้กลุ่มข้อมูลแบบ 10 บิต และสามารถเปลี่ยนแปลงอัตราความเร็วในการส่งข้อมูลได้
โหมด 2	ใช้สำหรับการเชื่อมต่ออนุกรมแบบ UART โดยการใช้กลุ่มแบบ การเชื่อมต่ออนุกรมแบบ UART โดยการใช้กลุ่มข้อมูลแบบ 11 บิต และกำหนดอัตราความเร็วในการส่งข้อมูลคงที่
โหมด 3	ใช้สำหรับแบบ 11 บิต และสามารถเปลี่ยนแปลงอัตราความเร็วในการส่งข้อมูลได้

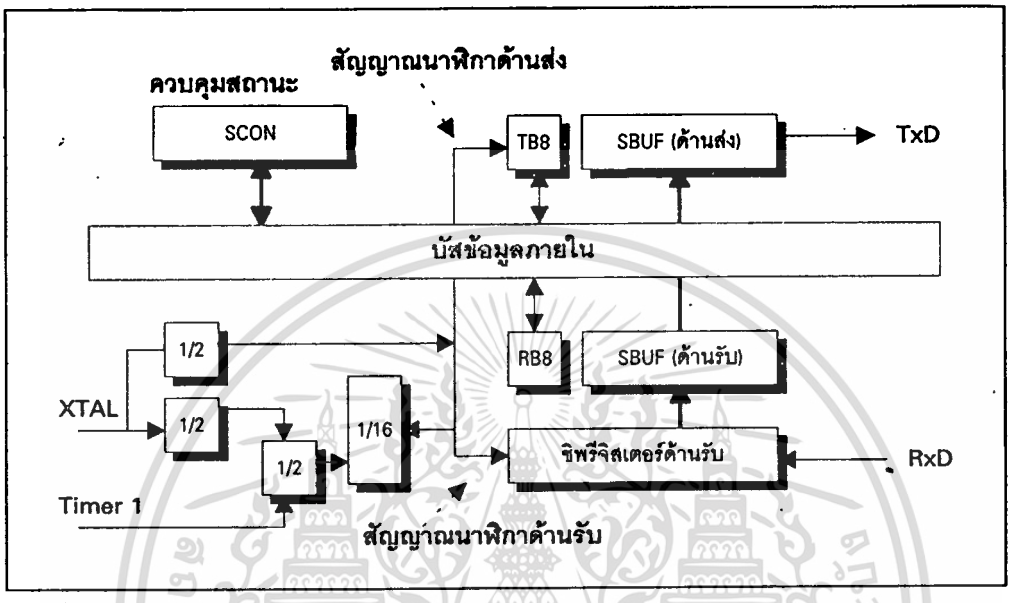
ตารางที่ 9 โหมดต่าง ๆ ของสื่อสารอนุกรม

นอกจากนี้โหมด 2 และ 3 ยังมีการดำเนินการแบบพิเศษออกไป โดยสามารถนำมาใช้ประโยชน์ในการสื่อสารข้อมูลแบบที่มีไมโครโปรเซสเซอร์หลายตัวทำงานร่วมกันได้ ซึ่งจะได้อธิบายรายละเอียดเป็นลำดับไป

จากแผนภาพในรูปที่ 14 ชิพรีจิสเตอร์ภายในตัวส่งจะทำหน้าที่ในการเลื่อนบิตข้อมูลออกไปภายนอกโดยไม่มีการบัฟเฟอร์ และเมื่อใดที่มีการเขียนข้อมูลให้กับรีจิสเตอร์ SBUF แสดงว่ามีความต้องการที่จะส่งข้อมูลนี้ออกไปแบบอนุกรม สำหรับชิพรีจิสเตอร์ทางด้านรับจะทำการเลื่อนบิตข้อมูลที่รับเข้ามาเก็บไว้ เมื่อบิตของข้อมูลที่รับมาครบถ้วนตามจำนวนที่กำหนดไว้ตามลักษณะโหมดการทำงานต่าง ๆ แล้ว จะถูกย้ายไปเก็บยังรีจิสเตอร์ SBUF ต่อไป อย่างไรก็ตามการย้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลนี้จะเกิดขึ้น ก็ต่อเมื่อรีจิสเตอร์ SBUF นั้นไม่มีข้อมูลที่จะทำการส่งหรือได้ส่งข้อมูลออกไปเสร็จสิ้นแล้ว



รูปที่ 14 แผนภาพแสดงการทำงานของวงจรส่วนการรับและส่งข้อมูลอนุกรมของ 8051

ชื่อบิต	ตำแหน่ง	ความหมาย
SM0	SCON.7	แฟล็กกำหนดการทำงานแบบมัลติโปรเซสเซอร์
SM1	SCON.6	บิตเลือกโหมดการทำงาน
SM2	SCON.5	บิตเลือกโหมดการทำงาน
REN	SCON.4	แฟล็กยอมให้มีการรับข้อมูล
TB8	SCON.3	ค่าของบิตที่9 สำหรับการส่งข้อมูล
RB8	SCON.2	ค่าของบิตที่ 9 ของข้อมูลที่รับเข้า
TI	SCON.1	แฟล็กแสดงการอินเตอร์รัปต์ภายหลังการส่งข้อมูล
RI	SCON.0	แฟล็กแสดงอินเตอร์ปต์เมื่อข้อมูลรับเข้ามา

ตารางที่ 10 รีจิสเตอร์ควบคุมการทำงานและบอกสถานะการสื่อสารข้อมูลอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.7 การอินเตอร์รัปต์ของการสื่อสารอนุกรม

เนื่องจากการส่งหรือรับข้อมูลอนุกรมในการส่งข้อมูลไบต์หนึ่ง ๆ ค่อนข้างจะใช้เวลานานหลายมิลลิวินาที ดังนั้นเพื่อให้การจัดการเกี่ยวกับการสื่อสารแบบนี้เป็นไปอย่างมีประสิทธิภาพ 8051 จึงได้กำหนดให้บิตหรือแฟล็กสถานะที่เกี่ยวข้องทั้งหมดจัดรวมอยู่ภายในรีจิสเตอร์ SCON เท่านั้น เช่น แฟล็ก TI ซึ่งจะมีค่าเป็น 1 เมื่อข้อมูลได้ทำการส่งออกไปภายนอกเสร็จสิ้นแล้ว และแฟล็ก RI ซึ่งจะมีค่าเป็น 1 จะมีผลทำให้เกิดการอินเตอร์รัปต์ขึ้น ดังนั้นภายในโปรแกรมจะต้องทำการตรวจสอบจากสภาวะของแฟล็กเหล่านี้เองว่ามีการอินเตอร์รัปต์ขึ้น ดังนั้นภายในโปรแกรมจะต้องทำการตรวจสอบจากสภาวะของแฟล็กเหล่านี้เองว่า มีการอินเตอร์รัปต์ขึ้นด้วยสาเหตุใด จากนั้นจึงค่อยทำการกำหนดค่า 0 ให้กับแฟล็กนั้น ลักษณะดังกล่าวนี้ จะมีความแตกต่างไปจากการอินเตอร์รัปต์จากสัญญาณอื่น ๆ เช่น วงจรนับ / จับเวลา เป็นต้น ซึ่งจะมีการกำหนดค่า 0 ให้กับแฟล็กสถานะที่เกี่ยวข้องโดยอัตโนมัติ ภายหลังจากที่ได้เข้าไปทำงานยังส่วนของโปรแกรมย่อยบริการอินเตอร์รัปต์ ดังนั้นจึงขอให้สังเกตความแตกต่าง ในส่วนนี้ไว้ด้วย

2.3.8 กระบวนการรับและส่งข้อมูลอนุกรมของ 8051

การส่งข้อมูลออกทางพอร์อนุกรมของ 8051 จะเริ่มต้นขึ้น ภายหลังเมื่อมีการเขียนข้อมูลลงใน รีจิสเตอร์ SBUF ข้ อมูลนี้จะถูกจัดการด้วยวิธีการทางด้าน ฮาร์ดแวร์ในการเลื่อนบิตและส่งสัญญาณออกไปภายนอกโดยอัตโนมัติ เมื่อข้อมูลเหล่านี้ได้ส่งออกครบถ้วนแล้ว จึงจะทำการกำหนดค่าของแฟล็ก TI ให้เป็น 1 เพื่อแจ้งให้ทราบว่า ขณะนี้รีจิสเตอร์ SBUF ว่าง และพร้อมที่จะส่งข้อมูลไบต์ต่อไปแล้ว ในกรณีที่ผู้ใช้เขียนข้อมูลใหม่ลงในรีจิสเตอร์ SBUF โดยไม่รอให้แฟล็ก TI มีค่าเป็น 1 ก่อนจะมีผลทำให้ข้อมูลที่ส่งออกไปผิดพลาดได้

สำหรับการรับข้อมูลจากพอร์ตอนุกรมจะต้องเริ่มต้นโดยการกำหนดค่าบิต REN (Reciever Enable) ให้มีค่าเป็น 1 ก่อน หลังจากนั้นเมื่อมีบิตของข้อมูลถูกส่งเข้ามาจากภายนอก ระบบฮาร์ดแวร์ของ 8051 จึงจะทำการเลื่อนบิตเหล่านี้เข้ามาโดยอัตโนมัติ และเมื่อบิตสุดท้ายถูกเลื่อนเข้ามาเรียบร้อยแล้ว ข้อมูลนั้นจะถูกย้ายมาเก็บไว้ยังรีจิสเตอร์ SBUF และทำการกำหนดให้แฟล็ก RI ให้มีค่าเป็น 1 ซึ่งมีผลทำให้เกิดการอินเตอร์รัปต์โปรแกรมขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GAS	VELOCITY (m/s)
ARGON	319
AIR (DRY 0 องศาเซลเซียส)	331.45
CARBON MONOXIDE	338
CARBON DIOXIDE	259
HELIUM	965
HYDROGEN	1284
METHANE	430
NEON	435
NITOGEN	334
OXYGEN	316
STREAM (134 องศาเซลเซียส)	494

ตารางที่ 11 แสดงความเร็วของคลื่นในตัวกลางต่าง ๆ

2.4.5 **การลดทอนของคลื่นอุลตร้าโซนิค** (Attenuation)

เมื่อคลื่นเดินทางผ่านตัวกลาง ลำคลื่น (Beam) ของคลื่นอุลตร้าโซนิคจะสูญเสียความเข้ม ซึ่งเกิดจากการสูญออกของลำคลื่นอุลตร้าโซนิค หรือเกิดจากการกระจายพลังงานของคลื่นออกจากลำคลื่น เนื่องจากความไม่ต่อเนื่องในตัวกลาง และอาจเกิดจากการดูดซับพลังงานส่วน หนึ่งของคลื่นโดยตัวกลางที่คลื่นเสียงเคลื่อนที่ผ่าน พลังงานที่ดูดซับนี้จะเปลี่ยนไปเป็นพลังงานความร้อน การดูดซับพลังงานนี้ขึ้นอยู่กับคุณลักษณะของวัตถุ ความยืดหยุ่นและความหนาแน่นรวมทั้งความถี่ของคลื่นที่ใช้ ยิ่งความถี่สูงพลังงานจะยิ่งถูกดูดซับมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.6 การทำงานของทรานสดิวเซอร์ตัวส่งและตัวรับ

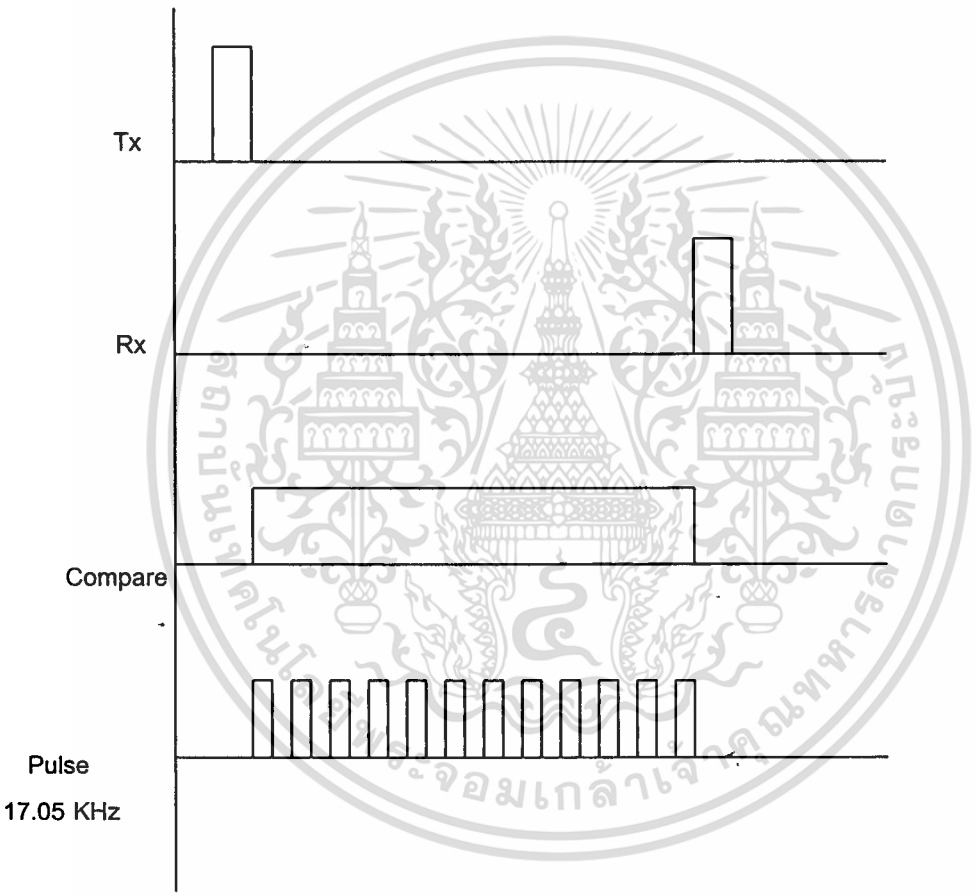
เมื่อเซรามิคได้รับสัญญาณแรงดันมาตกคร่อม จะทำให้ขึ้นสารเซรามิคโก่งงอทำให้เกิดการอัดอากาศโดยรอบเกิดเป็นคลื่นออกมาดังนั้นถ้าป้อนสัญญาณเป็นห่วง (Electrically) จากออสซิลเลเตอร์ก็จะทำให้ชิ้นสารโก่งงอ มากน้อยหรือทิศทางใดตามขนาดและ ทิศทางการเปลี่ยนแปลงขนาดของสัญญาณไฟฟ้าจากออสซิเลเตอร์นั้นออกไป โดยทั่วไปกำลังเอาท์พุทที่ออกมาจะตกลงประมาณ 10 % ของกำลังไฟฟ้าที่ป้อนให้แต่เอาท์พุทจะสูงที่ค่านี้โดยประมาณก็ต่อเมื่อความถี่ของสัญญาณออสซิสเลเตอร์ที่ป้อนเข้าชิ้นสารเซรามิคตรงกันกับความถี่เรโซแนนท์ที่เป็นความถี่ทางกลธรรมชาติของชิ้นสารเซรามิคนั้น ๆ ส่วนที่ความถี่อื่น ๆ กำลังเอาท์พุทจะลดลงกว่านี้ส่วนการทำงานของทรานสดิวเซอร์ตัวรับ มีการทำงานตรงกันข้ามกับตัวส่ง คือ เมื่อมีคลื่นเสียงที่มีความถี่ต่างกับความถี่เรโซแนนท์ของชิ้นสารเซรามิคเข้ามา จะทำให้ชิ้นสารโก่งตัวไปมาและเกิดสัญญาณแรงดันไฟฟ้าขึ้นซึ่งมีขนาดเล็กขึ้น คร่อมขั้วทั้งสองของตัวมัน

ทั้งตัวส่งและตัวรับจะมีทิศทางคล้ายคลึงกันมาก กล่าวคือ ที่ตำแหน่งแบนจากแนวแกนของตัวส่งไปประมาณ 30 องศา ความแรงของคลื่นเสียงที่ถูกส่งออกไปจะลดลงจากแนวแกนประมาณ 10 dB ในทำนองเดียวกันถ้าคลื่นเสียงพุ่งเข้ามาในแนวที่เบี่ยงเบนไปจากแนวแกนของตัวรับประมาณ 30 องศา ความไวหรือขนาดแรงดันที่ออกมาก็ลดลง จึงควรพยายามให้ทั้งตัวรับและตัวส่งอยู่ในแนวที่พุ่งตรงเข้าหากันให้มากที่สุด อย่างไรก็ตาม ในกรณีที่อยู่ในห้องอาจจะเบี่ยงเบน จากกันได้มาก เพราะคลื่นเสียงอุลตร้าโซนิคสามารถสะท้อนกับกำแพง และวัตถุที่อยู่ภายในห้องทำให้คลื่นเสียงเข้าไปหาตัวรับได้หลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.7 แนวคิดและหลักการวัดระยะทาง

โดยอาศัยหลักการส่งเคลื่อนสัญญาณไปในอากาศในทิศทางที่เราต้องการทราบค่าระยะทางจากวัตถุกับตัว ROBOT และคอยรับสัญญาณที่สะท้อนกลับมาจากวัตถุจากนั้นก็นำสัญญาณที่ส่งไปและรับกลับมานำมาเปรียบเทียบเวลาที่สูญหายไปดังรูปที่ 15



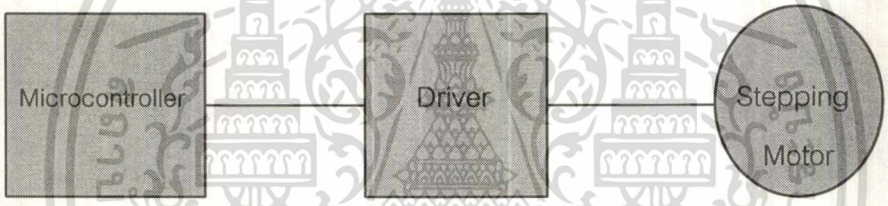
รูปที่ 15 แสดง Timing Diagramของการรับ - ส่งคลื่น Ultrasonic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ทฤษฎีและหลักการของการควบคุม Stepping Motor

การควบคุม Stepping motor สามารถที่จะควบคุมได้โดยคำสั่งที่ผ่านมาจาก Microcontroller ทั้งนี้เพราะว่า Stepping motor ที่สามารถ ควบคุมตำแหน่งของงาน (Position Control) ได้ค่อนข้างแน่นอน และง่ายควบคุมการทำงานแบบ Open Loop Control คือ เป็นมอเตอร์ที่นิยมใช้มากในงานควบคุมอุตสาหกรรมขนาดย่อมโดยเฉพาะในเครื่องจักรที่ต้องควบคุมตำแหน่งงาน

ลักษณะของโครงสร้างสำหรับชุดควบคุมและชุดขับ Stepping motor สำหรับโครงงานนี้เขียนเป็น Block Diagram ได้ดังรูป



รูปที่ 16 บล็อกไดอะแกรมของส่วนควบคุมและชุดขับ

จากการควบคุมทั้งสองกรณี สามารถแบ่งเป็นส่วนใหญ่ ๆ ได้ 2 ส่วน

- ส่วนของ Hardware
- ส่วนของ Solfware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 **ส่วนของ** Hardware

2.5.1.1 **ไมโครคอนโทรลเลอร์**

อุปกรณ์นี้ใช้ในการคำนวณ และควบคุมอุปกรณ์ต่างๆ ได้อย่างมีประสิทธิภาพ โดยไมโครคอนโทรลเลอร์จะถูกออกแบบมาให้เหมาะกับการควบคุมแบบบิต ดังนั้นจึงเหมาะสมอย่างยิ่งที่จะใช้ไมโครคอนโทรลเลอร์ควบคุมสเต็ปปิ้งมอเตอร์

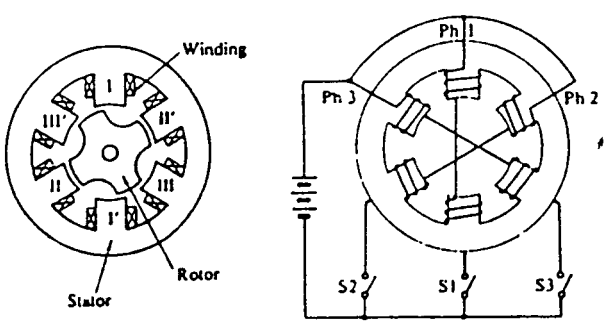
2.5.1.2 **สเต็ปปิ้งมอเตอร์** (Stepping Motor)

ในปัจจุบันนี้จะเห็นว่าสเต็ปปิ้งมอเตอร์มีใช้อยู่หลาย ๆ งาน เช่นในเครื่องพิมพ์พล็อดเตอร์ X-Y ,ในแขนกล หรือในเครื่องถ่ายเอกสารเพราะในสเต็ปปิ้งมอเตอร์ มีข้อได้เปรียบมอเตอร์ไฟตรง (DC. Motor) แบบธรรมดา คือสเต็ปปิ้งมอเตอร์มีการควบคุมแบบลูปเปิด (Open Loop) ทำให้งานต่อ การควบคุม และเรายังสามารถที่จะรู้ตำแหน่งที่แน่นอนจึงมักเลือกใช้สเต็ปปิ้งมอเตอร์

การควบคุมสเต็ปปิ้งมองเตอร์ ส่วนใหญ่ใช้ คอมพิวเตอร์ หรือชิงเกิ้ลบอร์ดควบคุม โดยจะทำชุดต่อเชื่อม(interface) ระหว่างคอมพิวเตอร์ กับสเต็ปปิ้งมอเตอร์ และจะต้องมีซอฟแวร์เพื่อควบคุมสเต็ปปิ้งมอเตอร์ และจะต้องมีซอฟแวร์เพื่อควบคุมสเต็ปปิ้งมอเตอร์ได้แน่นอนถูกต้องและมีความละเอียดสูงมาก

2.5.1.2.ก **โครงสร้างและหลักการทำงานของ** Stepping Motor

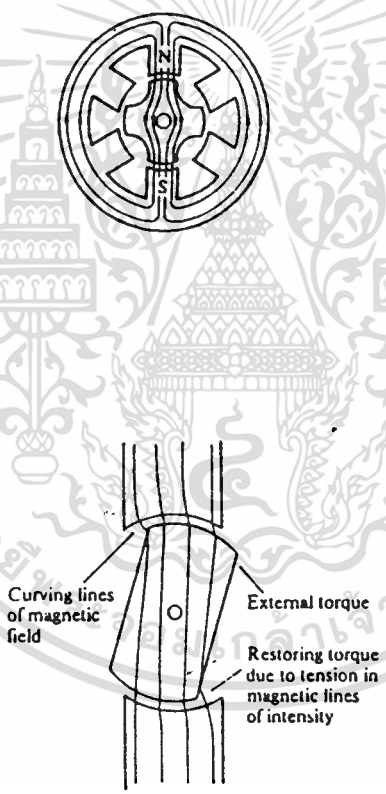
ภายในสเต็ปปิ้งมอเตอร์ประด้วยด้วย สเตเตอร์ (stator),โรเตอร์ (rotor) และขดลวด (coil) ประกอบเข้าด้วยกันดังรูปที่ 17



รูปที่ 17 โครงสร้างของสเตปปิ้งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากสเต็ปปิ้งมอเตอร์โรเตอร์เป็น เหล็กอ่อน ซึ่งมีคุณสมบัติที่พยายามปรับตัวเองในแนวที่เส้นแรงแม่เหล็ผ่านมากที่สุด ดังใน รูปที 18 เมื่อเกิดเส้นแรงแม่เหล็กขึ้นที่ตัวสเตเตอร์ ผ่านโรเตอร์ผ่านโรเตอร์ ตัวโรเตอร์ก็จะพยายามปรับตัวเองให้เส้นแรงแม่เหล็กผ่านมากที่สุด โดยการหมุนตัวเอง ทำให้เคลื่อนที่ไปเกิดมุมของการหมุน เมื่อเส้นแรงแม่เหล็กที่ตัดผ่านตัวมันถึงจุดมากที่สุด

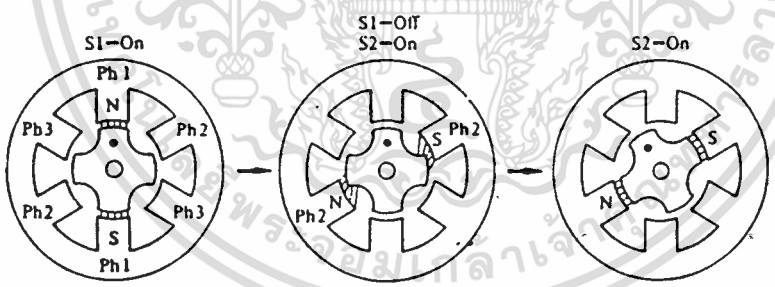


รูปที่ 18 เส้นแรงแม่เหล็กที่ทำให้เกิดแรงบิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำให้สเต็ปปิ้งมอเตอร์หมุนนั้นทำได้ โดยอาศัยหลักการนี้แต่ต้องให้เส้นแรงแม่เหล็กเกิดขึ้นโดยรับช่องต่อกันไปเรื่อย ๆ ดังรูปที่ 19 ก ข และ ค ซึ่งแสดงถึงการหมุนของมอเตอร์โดยมีทิศทาง ขึ้นอยู่กับการขับกระแสเข้าขดลวด ว่าจะให้ไปทางไหนและเมื่อต้องการให้มอเตอร์หยุดก็โดยที่ขับโรเตอร์ มอเตอร์ก็จะหยุด ณ ตำแหน่งสุดท้ายที่มีทิศทาง ขึ้นอยู่กับการขับกระแสเข้าขดลวดว่าจะให้ไปทางไหน และ เมื่อต้องการให้มอเตอร์หยุดก็โดยการขับโรเตอร์ มอเตอร์ก็จะหยุด ณ ตำแหน่งสุดท้ายที่มีการขับที่สเตเตอร์ ดังนั้นเราจึงสามารถ รู้ตำแหน่งของมอเตอร์ได้ โดยการนับจำนวนพัลซ์ ที่ป้อนให้มอเตอร์โดยใช้สูตร

มุมที่เปลี่ยนไป = ค่ามุมต่อสเต็ป + จำนวนพัลซ์ที่ป้อนให้

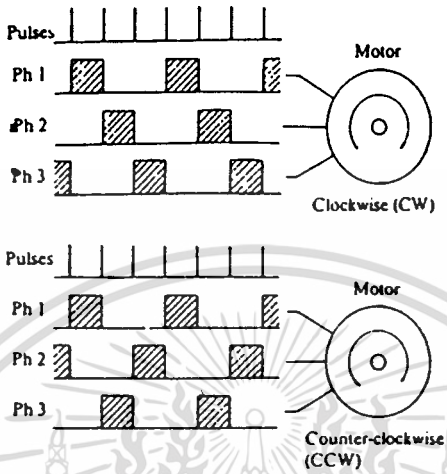


รูปที่ 19 แสดงการเคลื่อนที่ทีละสเต็ป เมื่อกระตุ้น เฟส 1 เฟส 2

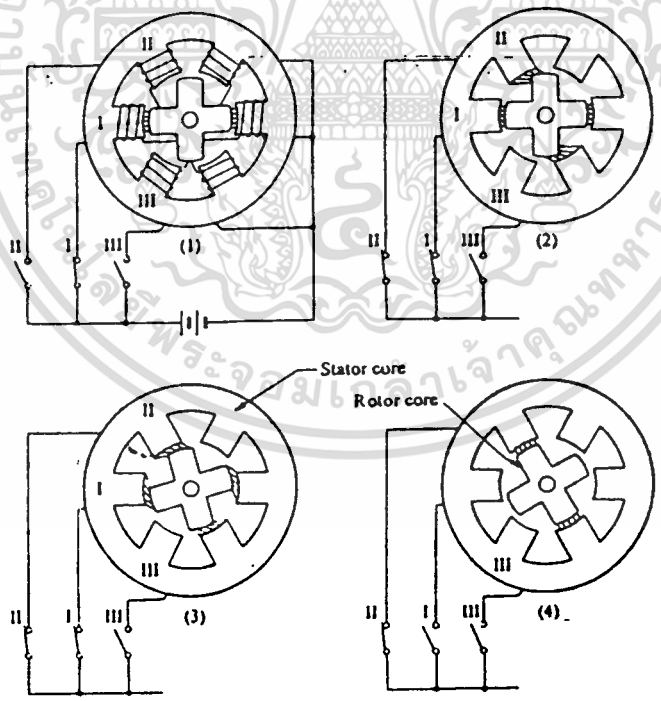
2.6.1.2.ข การกระตุ้นเฟสสเต็ปปิ้งมอเตอร์

ดังที่รู้ว่าในการให้สเต็ปปิ้งมอเตอร์หมุนได้นั้น จะต้องกระตุ้นเฟสของขดลวดสเตเตอร์ให้เรียงกันไปเรื่อย ๆ ทางใดทางหนึ่ง ถ้าต้องการให้กลับ ก็กระตุ้นเฟสในทิศทางกลับกัน ดังรูปที่ 20 การทำงานของสเต็ปปิ้งมอเตอร์ ดังรูปที่ 21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 20 แสดงการกระตุ้นแบบเดินหน้าและถอยหลัง

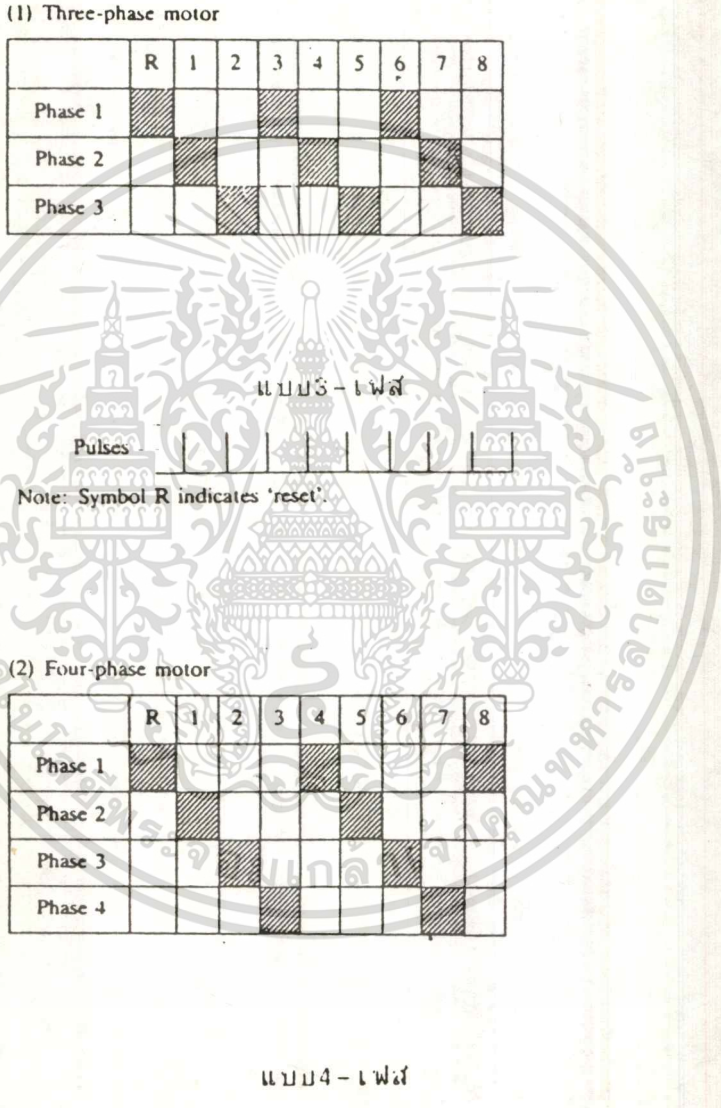


รูปที่ 21 แสดงการทำงานของสเต็ปปิ้งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1.2.ข.1 การกระตุ้นของสเตเตอร์มีอยู่ 3 แบบ

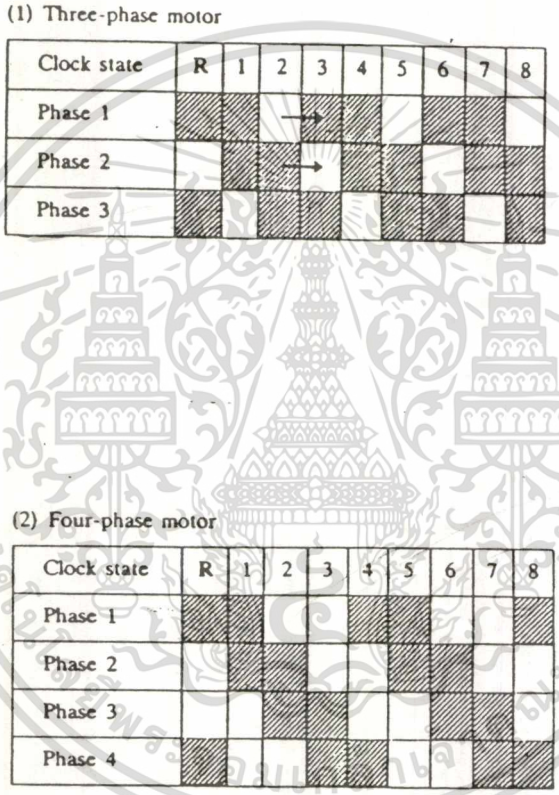
- การกระตุ้นแบบนี้ซึ่งจะมีแรงบิดย่อยกว่า การกระตุ้นแบบ 2 เฟส และการเข้าตำแหน่งจะช้ากว่าแบบ 2 เฟส



รูปที่ 22 การกระตุ้นเฟสแบบเฟสเดียวของสเต็ปปิ้งมอเตอร์แบบ 3และ4เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

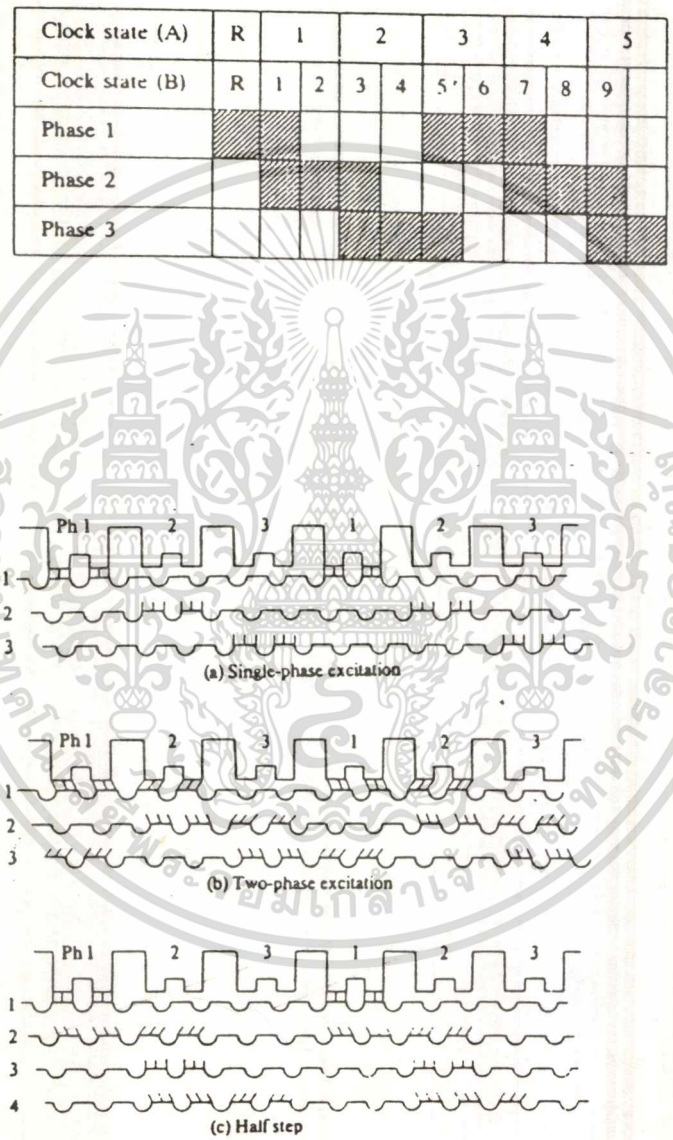
- การกระตุ้นสองเฟสเรียกว่า แบบ Two phase excitation ดังรูปที่ 23 การกระตุ้นแบบ2 เฟสนี้ มอเตอร์จะมีแรงบิดดีกว่าเฟสเดียว และสามารถเข้าตำแหน่งได้เร็วกว่า แบบ 1 เฟส



รูปที่23 การกราะตุ้นแบบสองเฟสของเต็ปมอเตอร์ 3 เฟส และ 4 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การกระตุ้นโดยใช้แบบ 1 และ 2 เฟส สลับกลับกัน เรียกว่า แบบ One- Two phase excitation หรือแบบ hafe step operation ดังรูปที่ 24ซึ่งการต่อแบบนี้จะให้แรงบิดดีกว่า 1เฟส และ 2 เฟส แต่การเข้าตำแหน่ง จะช้ากว่าแบบ 1 เฟส และ 2 เฟส

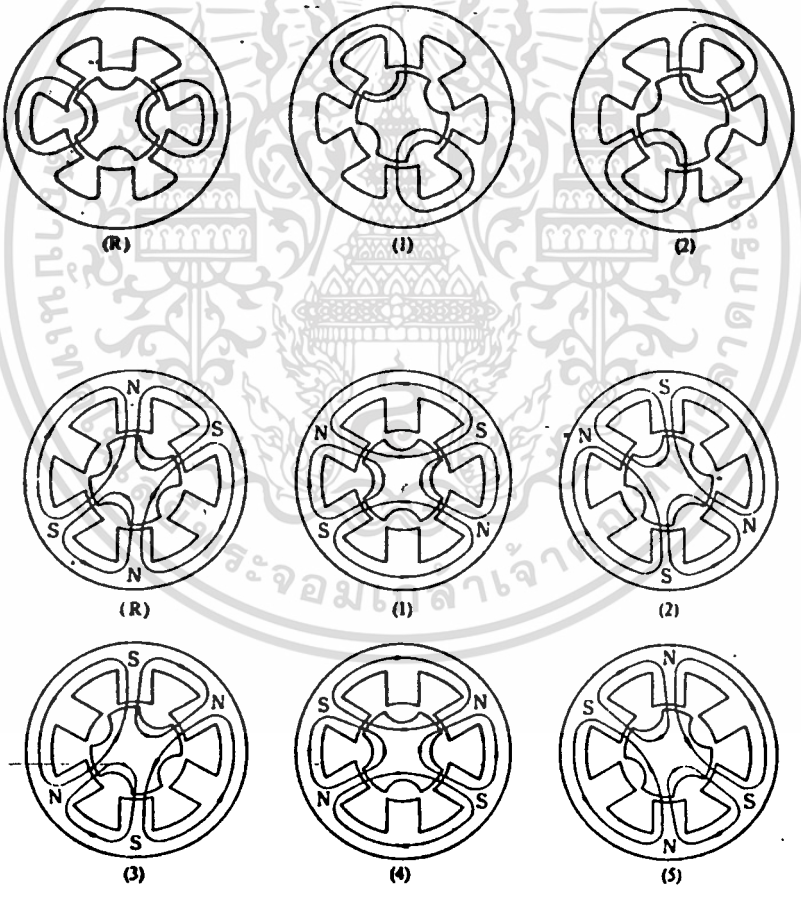


รูปที่ 24 การกระตุ้นแบบครึ่งสเต็ปของมอเตอร์แบบ 3เฟสและ 4 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการขับแบบกระตุ้น 2 เฟส เส้นแรงแม่เหล็ก จะไม่ผ่านแกนเหล็กเป็นเส้นตรงเลยทีเดียว เหมือนกระตุ้น แบบเฟสเดียวเหมือนกระตุ้นแบบเฟสเดียว แต่จะวกกลับมาทางด้านข้าง ดังรูปที่ 25 และเส้นแรงแม่เหล็กส่วนหนึ่งมาจากแกนตรงกันข้ามดังรูปที่ 26

การขับสเต็ปปิ้งมอเตอร์แบบกระตุ้น 2 เฟสนี้ จะมีลักษณะเดียวกันกับ การขับแบบกระตุ้นเฟสเดียวแต่ละครั้งนั้นจะต้องกระตุ้นทีเดียวพร้อมกันทั้ง 2 เฟส จะต่างกันก็ตรงที่การขับแบบ 2 เฟส จะต่างกันก็ตรงที่การขับแบบ 2 เฟส เข้าตำแหน่งแต่ละสเต็ปได้เร็วกว่าแบบ เฟสเดียว และแรงบิด มากกว่าแบบเฟสเดียวด้วย



รูปที่ 26 ลักษณะเส้นแรงแม่เหล็กเมื่อขับแบบ 2 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1.2.ค การเกิดการออสซิลสเลตของสเต็ปปิ้งมอเตอร์

ความเร็วของการหยุดสเต็ปปิ้งมอเตอร์เป็นแบบลิเนียร์กับความถี่ที่ป้อนให้ สเต็ปปิ้งมอเตอร์แต่เราป้อนความถี่เพิ่มขึ้นเรื่อย ๆ จนถึงความถี่ค่าหนึ่ง สเต็ปปิ้งมอเตอร์ก็จะหยุดหมุน เนื่องจากการที่โรเตอร์หมุนตามฟลักซ์แม่เหล็กไม่ทัน เราเรียกว่ามอเตอร์เกิด การออสซิลเลต ที่ความถี่ประมาณ 500 Hz. ในการขับกระตุ้น 1 หรือ 2 เฟส และจะมีความถี่ประมาณ 1 kHz. เมื่อขับแบบครึ่งสเต็ป

แต่เมื่อลดความถี่ให้ต่ำ ลงมอเตอร์จะไม่หมุนทันที และเมื่อความถี่ประมาณความถี่หนึ่ง มอเตอร์จึงเริ่มหมุนอีกครั้ง นั้นคือ มอเตอร์มี อีสเทอรี ซิสซิ่ง มอเตอร์จะเริ่มหมุนที่ความถี่ประมาณ 200 Hz. ในการขับแบบ 1 หรือ 2 เฟส และมีความถี่ประมาณ 1KHz. เมื่อขับแบบครึ่งสเต็ป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและส่วนประกอบของโครงงาน

ส่วนประกอบทาง Hardware ประกอบด้วยกัน 4 ส่วนคือ

1. **ส่วน** คอมพิวเตอร์ (Personnel Computer)
2. **ส่วน** นำร่อง (Navigation)
3. **ส่วน** การควบคุม และการจัดการข้อมูล (ไมโครคอนโทรลเลอร์)
4. **ส่วน** ขับเคลื่อน

ส่วนประกอบทาง Software ประกอบด้วยกัน 3 ส่วนคือ

1. โปรโตคอลในการสื่อสารอนุกรม
2. โปรแกรมการควบคุมที่คอมพิวเตอร์
3. โปรแกรมการคบคุมและจัดการข้อมูลที่ ไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 ส่วนคอมพิวเตอร์ (Personnel Computer)

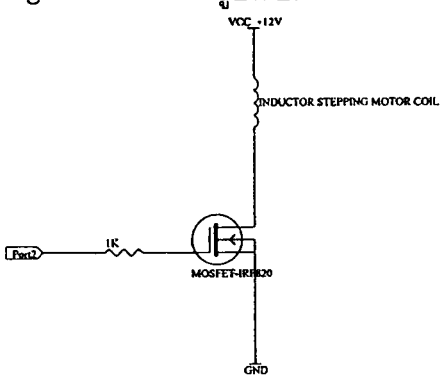
ในส่วนนี้จะใช้คอมพิวเตอร์ รุ่น Pentium - 133 MHz โดยมีส่วนประกอบดังนี้

- CPU PENTIUM 133 MHz
- Monitor Super VGA
- RAM 24 MB
- Avaliable COM 1 FOR SERIAL PORT

3.2 ส่วนนำร่อง (Navigation)

ระบบนำร่อง เป็นส่วนที่คิดคำนวณและประมวลผลเพื่อที่จะนำพาสิ่งใดสิ่งหนึ่ง เช่น ฐานหุ่นยนต์ หรือ รถสำรวจไปยังเป้าหมายและหลบหลีกวัตถุกีดขวางได้

การนำร่องจะใช้เครื่องวัดระยะทาง Ultrasonic ในการตรวจจับสิ่งกีดขวางที่มีอยู่รอบตัวของ รถสำรวจ โดยหลักการของการนำร่อง (Navigation) คือจะทำการหมุนตัวรับและตัวส่งคลื่น Ultrasonic ในมุมต่าง ๆ โดยเราจะนำ Stepping Motor มาใช้ในการควบคุมทิศทางของการวัดระยะโดย เมื่อเราต้องการจะวัดระยะห่างจากสิ่งรอบ ๆ ตัว ต่าง ๆ เช่น ถ้าหากเราต้องการจะวัดที่ตำแหน่งหน้ารถเราก็จะป้อนสัญญาณพัลส์ ให้ Stepping Motor เพื่อหมุนทิศทางของ Ultrasonic Transmitter And Reciever ไปยังทิศทางหน้ารถเพื่อวัดระยะโดยสรุปก็คือ เราสามารถที่จะวัดระยะห่างจากตัว รถสำรวจ ไปยังสิ่งกีดขวางได้ รอบตัวคือ 360 องศา โดยการที่เราจะวัดระยะในทิศใดก็ขึ้นอยู่กับว่า เราต้องการจะวัดในทิศใด ก็จะบังคับให้ Stepping Motor หมุน Ultrasonic Transmitter And Reciever ไปในทิศทางนั้นเพื่อวัดระยะ โดยสัญญาณควบคุมการหมุนของ Stepping Motor ก็จะรับมาจากส่วนควบคุม ซึ่งจะเป็นการควบคุมจากตัว ไมโครคอนโทรลเลอร์ โดยวงจรที่ใช้ในการขับ Stepping Motor จะมีดังรูปที่ 27



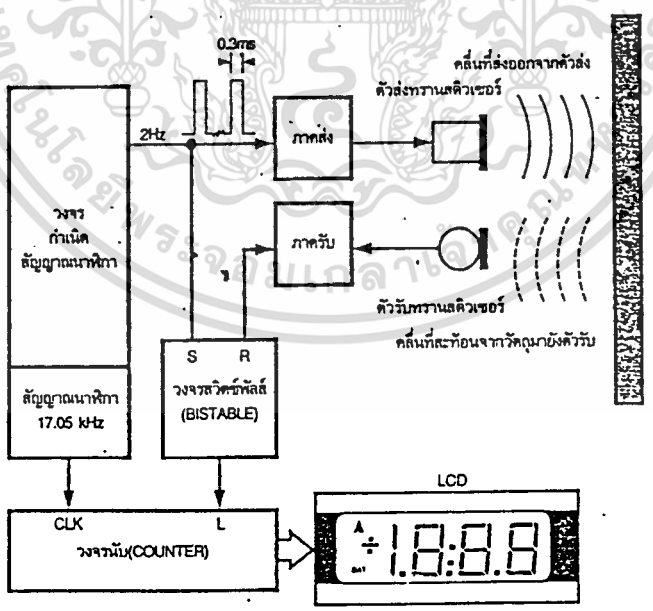
รูปที่ 27 วงจรขับ Stepping Motor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 การใช้อุลตร้าโซนิคในการวัดระยะทาง

จากรูป จะแสดงบล็อกไดอะแกรมการทำงานของเครื่อง ซึ่งประกอบไปด้วย ภาครับ,ภาคส่ง ซึ่งใช้ตัวส่ง และรับอุลตร้าโซนิคทรานสดิวเซอร์ความถี่ 40 KHz, ภาคแสดงผลใช้ LCD ขนาด 3 ½ หลัก, วงจรนับ, วงจรฐานเวลา, และ วงจรไบสเตเบิล การทำงานจะเริ่มที่ตัวส่งซึ่งทำหน้าที่ส่งความถี่ 40 KHz ซึ่งกำเนินโดยวงจรออสซิลเลเตอร์ผ่านตัวมันออกไป แล้วคลื่นจะสะท้อนจากวัตถุที่วัดเข้ามายังตัวรับ ขณะเดียวกันเมื่อเริ่มส่งวงจรนับจะเริ่มทำการนับจำนวนสัญญาณนาฬิกาไว้ จนกว่าตัวรับจะรับความถี่ที่สะท้อนเข้ามาได้ จากนั้นวงจรไบ สเตเบิล ถี่ที่คำนวณได้จากความเร็วของเสียงที่เดินทางในบรรยากาศ ซึ่งจะมีความเร็วประมาณ 341 เมตร/วินาที ในกรณีที่จะกำหนดให้วงจรนับหยุดนับและนำสัญญาณนาฬิกาที่นับได้มาเปลี่ยนเป็นตัวเลข และคงสภาวะตัวเลนั้นไว้ อย่าวนั้นจนกวาระยะทางที่วัดจะเปลี่ยนไป จึงจะทำตามขั้นตอนเดิมอีก

สัญญาณนาฬิการความถี่ 17.05 KHz นั้นเป็นความระยะการวัดใกล้เกินไป จะทำให้คลื่นไม่สามารถสะท้อนจากวัตถุที่วัดเข้ามายังตัวรับได้ ซึ่งระยะนี้จะอยู่ในช่วงการวัดตั้งแต่ 0 ถึง 25 เซนติเมตร สำหรับวงจรนี้



รูปที่ 28 แสดงบล็อกไดอะแกรมของเครื่องวัดระยะทางอุลตร้าโซนิค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การทำงานของวงจรวัดระยะทางอุลตร้าโซนิค

สำหรับวงจรเครื่องวัดระยะทางอุลตร้าโซนิคนี้ แสดงในรูปที่ภาคส่งจะใช้ IC1 ซึ่งเป็นประเภทซีมอสเบอร์ 4093 ภายในจะประกอบด้วย อินเวอร์เตอร์ 6 ตัว ซึ่ง 4 ตัว แรกคือ IC1/1 IC1/2 และ IC1/5 IC1/6 ซึ่งจะจับคู่กันโดยต่อในลักษณะขนานกันเพื่อขับตัวส่ง การทำเช่นนี้เพื่อที่จะเพิ่มขนาดของสัญญาณที่ตกคร่อมตัวส่งให้สูงขึ้นเป็น 2 เท่า เมื่อเทียบกับการต่อตัวส่งเข้ากับเอาต์พุตของอินเวอร์เตอร์เพียงตัวเดียว ซึ่งตัวส่งจะถูกออกแบบมาให้แปลงสัญญาณไฟฟ้าที่จ่ายให้แก่ตัวมัน อันมีความถี่ประมาณ 40 KHz มาเป็นคลื่นเสียงย่านอุลตร้าโซนิคเพื่อส่งให้กับตัวขับต่อไป ส่วน C1 ต่อไว้เพื่อป้องกันไฟตรง (ไม่ใช่ความถี่ 40 KHz) ผ่านเข้าไปในตัวส่งมีค่าสูงสุด

สำหรับแหล่งจ่ายแรงดันที่จ่ายให้กับวงจรนี้ จะใช้แบตเตอร์รี่ 9 โวลต์ โดยจะจ่ายให้กับ IC1 กับ IC5 และ IC7 ส่วนตัวอื่น ๆ จะใช้แรงดันแค่ 5 โวลต์ เท่านั้นโดยนำแรงกดดันจากแหล่งจ่ายมาผ่าน IC3 เบอร์ 78L05 โดยมี C3 ทำหน้าที่กรองแรงดันให้เรียบก่อนนำมาลดระดับแรงดัน วงจรออสซิลเลเตอร์จะกำเนิดความถี่ 40 KHz เพื่อให้เรโซแนนซ์กับตัวส่ง ซึ่งสามารถปรับความถี่ให้ได้ประมาณ 40 KHz โดยการปรับ 0 วงจรคอมพาราเตอร์จะใช้ IC7 เบอร์ LM393 ทำหน้าที่เปรียบเทียบสัญญาณที่รับเข้ามาได้ ซึ่ง IC7/22 จะทำหน้าที่เปรียบเทียบ ระดับลอจิกของวงจรออสซิลเลเตอร์ระดับสูง 5 โวลต์ กับเอาต์พุตของวงจร 9 โวลต์ โดยสัญญาณทางด้านเอาต์พุตที่ขา 7 จะมีความถี่ 40 KHz เพื่อส่งให้กับ IC1 ต่อไป

วงจรกำเนินสัญญาณนาฬิกาจะถูกสร้างโดย IC4 คือ เมื่อให้ระดับลอจิกที่ขา 12 เป็นระดับลอจิกสูง R2 และ C11 จะเป็นตัวกำหนดความถี่ให้ได้ประมาณ 40 KHz ขณะเดียวกันเอาต์พุตของ IC5/1 จะมีระดับลอจิกสูงซึ่งกระแสจะไหลผ่าน D1 เข้าสู่คอมพาราเตอร์ตัวที่ 2 คือ IC7/1 เพื่อยกระดับของสัญญาณให้สูงขึ้นและนำไปทริก IC2/4 ได้โดยที่แรงดันทางด้านอินพุตต์ของ IC5/1 ที่ขานอนอินเวอร์ติ้งจะมีค่าประมาณ 2.5 โวลต์ ขณะเดียวกันวงจรไบสเตเบิ้ล ซึ่งประกอบด้วย IC2/3 และ IC2/4 นี้จะถูกเซต ดังนั้นขา 31 ของ IC8 ซึ่งเป็นขา CNT INH (Count Inihibit) ถูกกำหนดให้มีสัญญาณเป็น ดิสเอเบิ้ล(Disable) เพราะฉะนั้นจะทำให้ขา 32 (CNT) ของ IC8 รับสัญญาณจากภายนอกเข้ามาได้ ซึ่ง IC4 จะเป็นตัวกำเนิดความถี่ 17.05 KHz รออยู่แล้วจึงทำให้ความถี่นี้ผ่านเข้าไปยัง IC8 ได้ เพื่อเป็นสัญญาณนาฬิกาใช้สำหรับอ้างอิงในการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

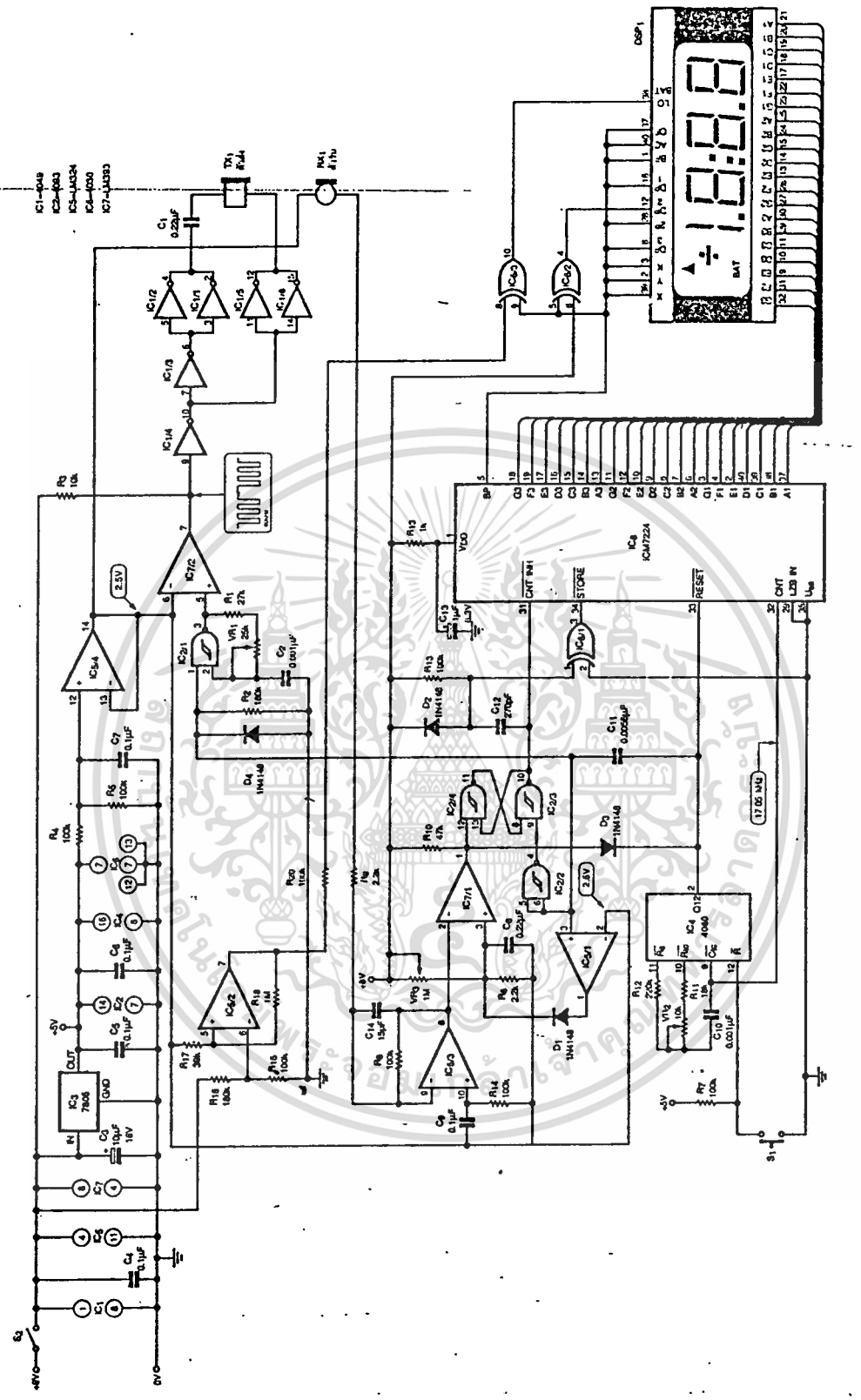
ส่วนวงจรในด้านภาครับ ใช้ตัวรับคลื่นอุลตร้าโซนิคความถี่ 40 KHz เพื่อให้รับคลื่นจจากตัวส่งได้ โดยสัญญาณที่ได้จะถูกขยายโดย IC5/3 ให้มีอัตราการขยายประมาณ 33 เดซิเบลซึ่งคำนวนได้จากสูตร

$$20 \log (R8/R9)$$

โดยวงจรขยายชุดนี้จะใช้การคัปปลิ้งแบบไฟสลับ คือ ใช้ตัวเก็บประจุเป็นตัวคัปปลิ้ง (C9) ภาคขยายชุดนี้ไม่จำเป็นต้องปรับแรงกันอินพุตออฟเซต เพราะมี R14 ช่วยในการปรับอย่างอัตโนมัติอยู่แล้ว โดยใช้กระแสไฟฟ้าไบแอสทางด้านอินพุตเป็นตัวช่วย และแรงดันออฟเซ็ตที่ถูกปรับจะถูกปรับให้มีค่าน้อยๆ เพราะมีความสำคัญมาก เนื่องจากแรงดันนี้ไม่ได้ถูกปรับเฉพาะ IC5/3 เพียงตัวเดียว แต่มีนจะมีผลต่อแรงดันออฟเซตของ IC7/1 ด้วย และยังมีผลต่อความไวในการรับอีก

สำหรับความไวในการรับจะถูกควบคุมโดยวงจรเวลาคงที่ (Time Constant) ซึ่งประกอบด้วย VR3, R6 และ C8 ซึ่งต่ออยู่กับ IC7/1 นั่นเอง เราสามารถจะปรับความไวในการรับได้ โดยการปรับ ให้ได้ความไวสูงสุด เมื่อตัวรับคลื่นอุลตร้าโซนิค รับคลื่นเข้ามาแล้วเอาต์พุตของ IC7/1 จะมีลอจิกต่ำ ซึ่งมีผลทำให้วงจรไบสเตเบิล (IC2/3, IC2/4) ถูกรีเซต และสัญญาณนาฬิกาความถี่ 17.05 KHz ไม่สามารถผ่านเข้าไปยัง IC8 ได้ในขณะเดียวกันพัลส์ลบซึ่งมีช่วงสั้น ๆจะผ่าน R13, C12 และ IC6/2 เข้าไปยังขา 34 (Store) ของ IC8 หยุดนับ เอาต์พุตของ IC8 ก็จะแสดงผลลัพธ์มีสภานะค้างไว้ตรงนั้นเพื่อขับภาคแสดงผลการนับใน IC8 จะถูกยกเลิกเมื่อเอาท์พุตขา 2 ของ IC4 เป็นลอจิกต่ำหรือเกิดการรีเซต วงจรก็พร้อมที่จะวัดระยะในครั้งถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 29 วงจรสมบูรณ์เครื่องวัดระยะทางอุลตร้าโซนิค

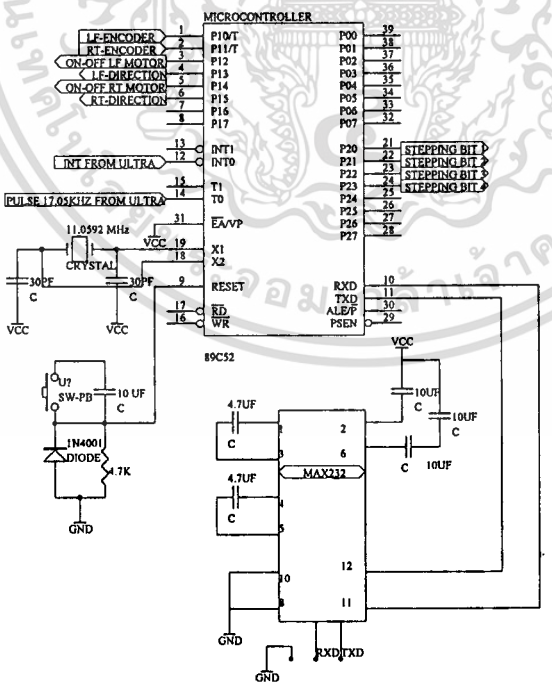
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนการควบคุมและจัดการข้อมูล

ในส่วนนี้จะใช้ไมโครคอนโทรลเลอร์ไอซีเบอร์ 89C52 ซึ่งจะทำหน้าที่ ควบคุมส่วนต่าง ๆดังนี้

1. ส่วน Stepping Motor โดยจะใช้พอร์ต2 บิตที่ P2.0 - P2.3 ในการป้อนสัญญาณควบคุมในแต่ละเฟสของStepping Motor
2. ส่วนวัดระยะทางอุลตร้าโซนิค จะใช้ขาอินเตอร์รัปต์ 0 (INT0) ต่อเข้ากับขา 31 ของไอซี8และจะใช้ขา T0 ในการต่อเข้ากับขา32ของไอซี8 เพื่อนับสัญญาณ พัลส์17.05 KHz
3. ส่วนขับเคลื่อนจะใช้บิต P1.0 และP1.1ในการรับค่าจากEncoder และใช้บิต P1.2 - P1.5 ในการควบคุมการเคลื่อนที่ของรถ
4. ส่วนสื่อสารอนุกรมจะใช้ขา RxD และ TxDต่อเข้ากับ ไอซี MAX232 ในการติดต่อแบบอนุกรมกับคอมพิวเตอร์โดยผ่านสายสัญญาณ

สำหรับวงจรใช้งานจะแสดงดังรูปที่ 30



รูปที่ 30 แสดงวงจรใช้งานของไมโครคอนโทรลเลอร์

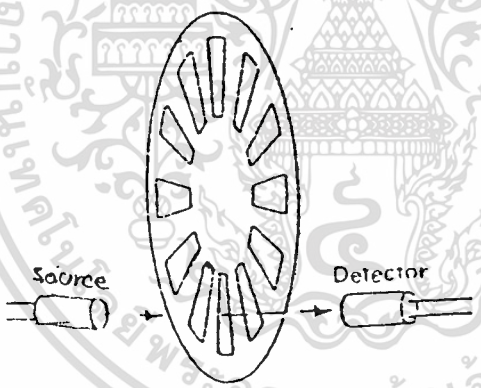
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

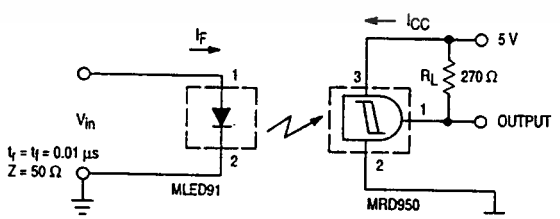
3.4 การออกแบบส่วนขับเคลื่อน

สำหรับการออกแบบ อินเตอร์เฟสในส่วนนี้จะแบ่งเป็น 2 ส่วน คือ ส่วนขับเคลื่อน และส่วนวัดระยะทาง ในส่วนขับเคลื่อนอาศัยการทำงานของ Motor โดยควบคุมการทำงานและทิศทางการเคลื่อนที่ของ Motor ทั้ง Motor ในส่วนขับเคลื่อนทางด้านซ้ายและในส่วนขับเคลื่อนทางด้านขวา

3.4.1 ส่วนการออกแบบการตรวจวัดระยะทางการเคลื่อนที

ส่วนการออกแบบการตรวจวัดระยะทางการเคลื่อนที นั้นจะอาศัยการออกแบบการตรวจจับแสงผ่านรูเล็กๆ โดยการออกแบบการตรวจจับแสงที่ผ่านรูนั้นอาศัยการทำงานของออปเตอร์คับเปิ้ลมาใช้ในการติดตั้งและทำงานดังนี้





รูปที่ 31 แสดงวงจร Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 ส่วนขับเคลื่อน

เนื่องจากลักษณะการเคลื่อนที่ของ รถสำรวจ จะเป็นแบบรถล้อตะขาบ โยจะมีทิศทางเคลื่อนของตัว รถสำรวจและทิศทางการเคลื่อนที่ของล้อดังนี้

F= Forward

B= Backward

S = Stop

ทิศการเคลื่อนที่ของรถ	ล้อซ้าย	ล้อขวา
ตรงไปข้างหน้า	F	F
ถอยหลัง	B	B
เลี้ยวซ้าย	S	F
เลี้ยวขวา	F	S

ตารางที่ 12 แสดงทิศการเคลื่อนที่ของรถสำรวจ

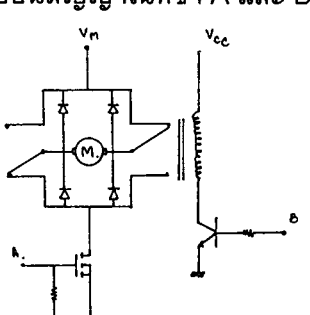
จากรูปแบบการเคลื่อนที่ดังตารางดังตารางดังนั้นเราต้องสามารถที่จะควบคุมการเปิดปิด และทิศทางการหมุนของ Motor ที่ใช้ในการเคลื่อนที่ของล้อซ้ายและขวาดังนั้นเราจึงนำวงจรรีเลย์ใช้งาน โดยล้อซ้ายและขวาจะมีวงจรขับแยกจากกัน

โดยจากวงจรเราสามารถจะควบคุมทิศทางการหมุนของ Motor

Vm ไฟที่จ่ายเพื่อเลี้ยง Motor

Vcc ไฟที่จ่ายเพื่อเลี้ยง Relay

เมื่อเราให้ Vm และ Vcc เลี้ยงวงจรไว้เราสามารถบังคับทิศทางการหมุนของ Motor ได้ โดยจะควบคุมการเคลื่อนที่โดยการป้อนสัญญาณที่ขา A และ B



รูปที่ 27 แสดงวงจรขับเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบโปรโตคอล

สำหรับการออกแบบกำหนดโปรโตคอล ที่ใช้ในการสื่อสารนั้นเราจะออกแบบไว้ให้มีลักษณะดังต่อไปนี้

1. File โปรแกรมคำสั่ง
2. รหัสจบ

File ข้อมูล	รหัสจบ

สำหรับการส่งผ่านข้อมูลจะเป็นลักษณะ Synchronous คือจะมีระยะเวลาการส่งแต่ละ Byte มีคาบเวลาที่แน่นอน ในส่วนของ File คำสั่งต่างๆ นั้น ได้ออกแบบไว้มีลักษณะโปรโตคอลดังต่อไปนี้

คำสั่งรถสำรวจ	จำนวนคำสั่ง (Byte)	รหัสคำสั่ง ASCII	
		Hex	Binary
เดินหน้า	2	31 XX	0011 0001 xxxx xxxx
เลี้ยวซ้าย	2	32 00	0011 0010 0000 0000
เลี้ยวขวา	2	33 00	0011 0001 0000 0000
ถอยหลัง	2	35 XX	0011 0001 xxxx xxxx
ตรวจสอบสิ่งกีดขวาง	2	34 00	0011 0001 0000 0000
รหัสจบ	1	FF	1111 1111 -

ตารางที่ 13 แสดงรหัสคำสั่งที่ใช้ในการควบคุม

XX หมายถึง ระยะทาง

00 หมายถึง ค่าที่กำหนดจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 โปรแกรมการควบคุมที่คอมพิวเตอร์

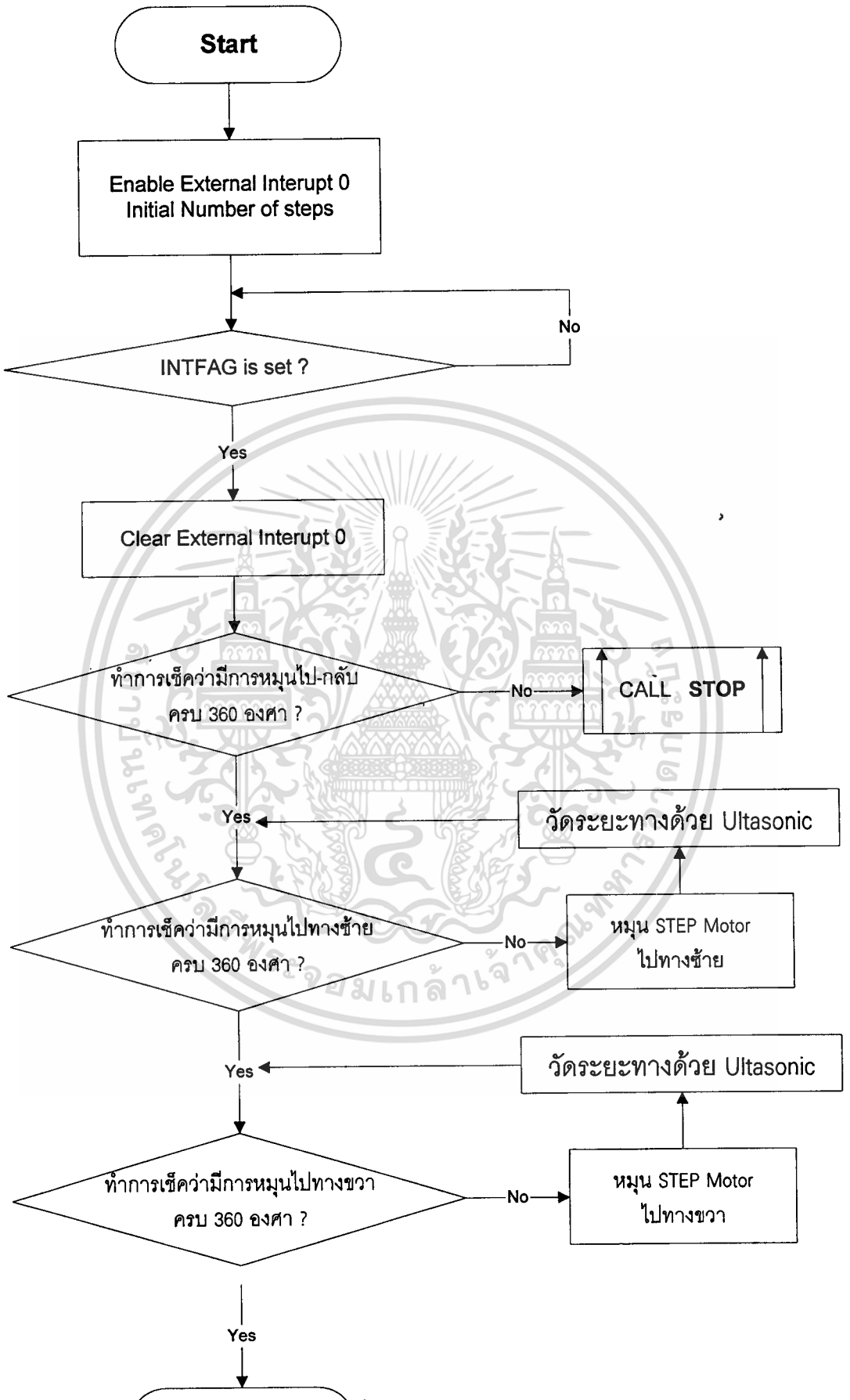
ในส่วนของโปรแกรมการควบคุมที่คอมพิวเตอร์ จะใช้ภาษา C ในการเขียนโดยจะมีขั้นตอนการทำงานตาม Flow Chart ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

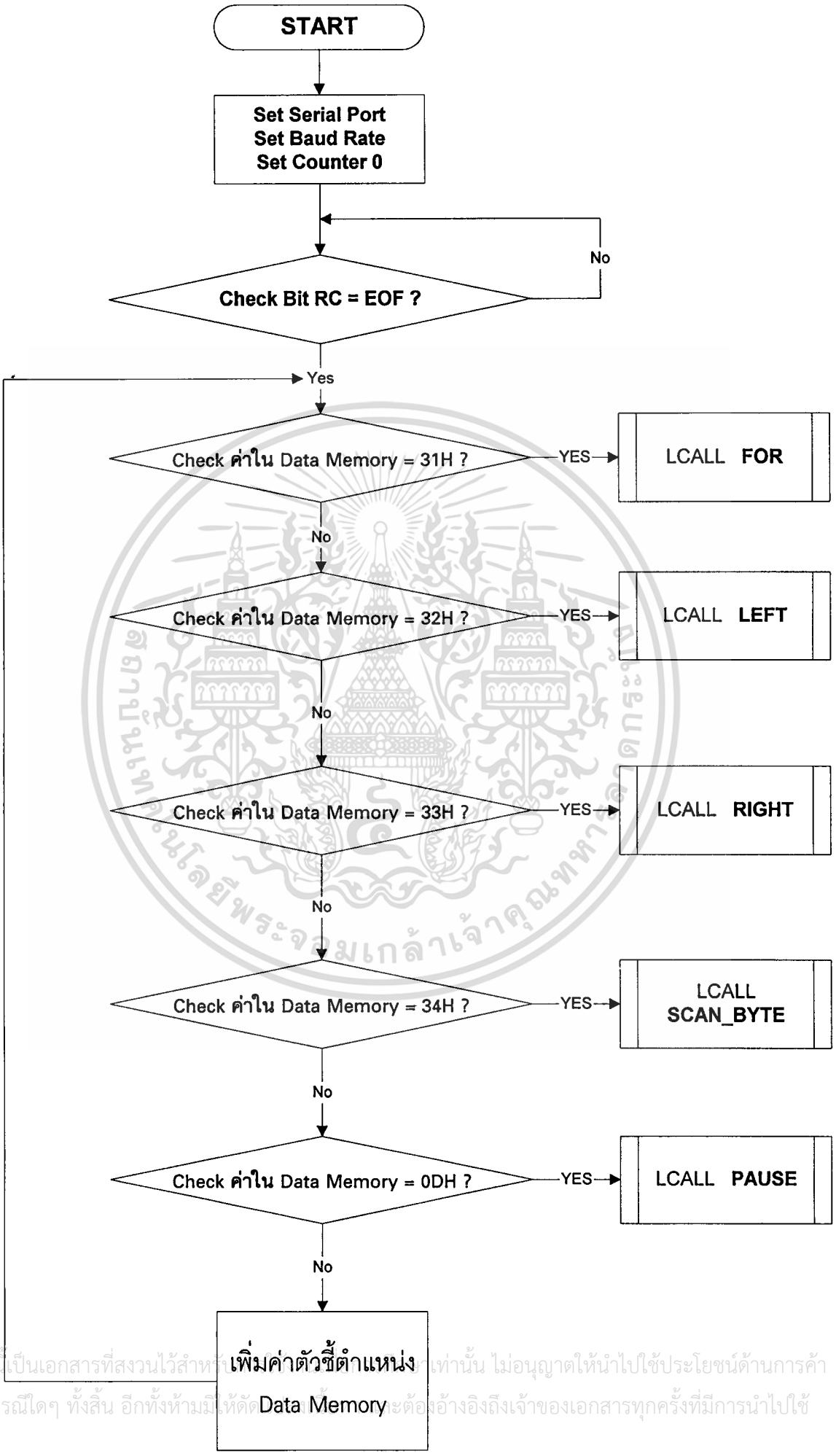
3.7 โปรแกรมการควบคุมและจัดการข้อมูลที่ไมโครคอนโทรลเลอร์

ในส่วนของโปรแกรมการควบคุมและจัดการข้อมูลที่ไมโครคอนโทรลเลอร์จะใช้ภาษาAssembly ของ 8051 ในการเขียนโดยจะมีขั้นตอนการทำงานตาม Flow Chart ดังรูป

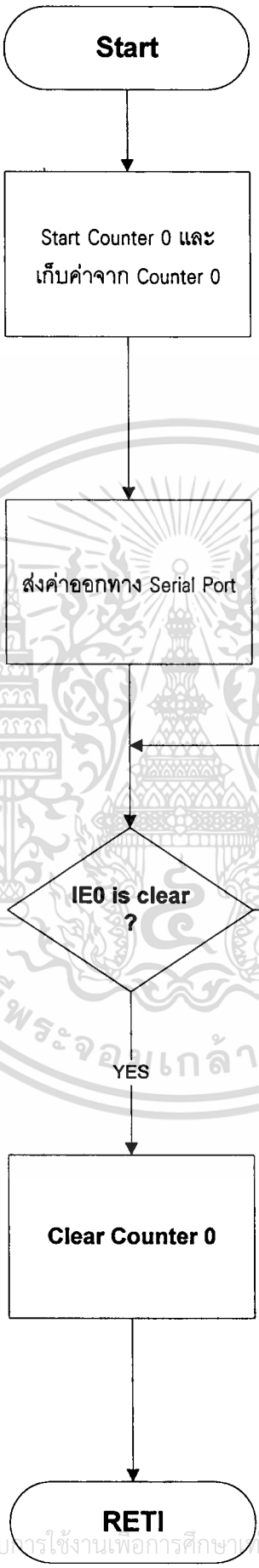
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

4.1 การทดลองส่วนการสำรวจ

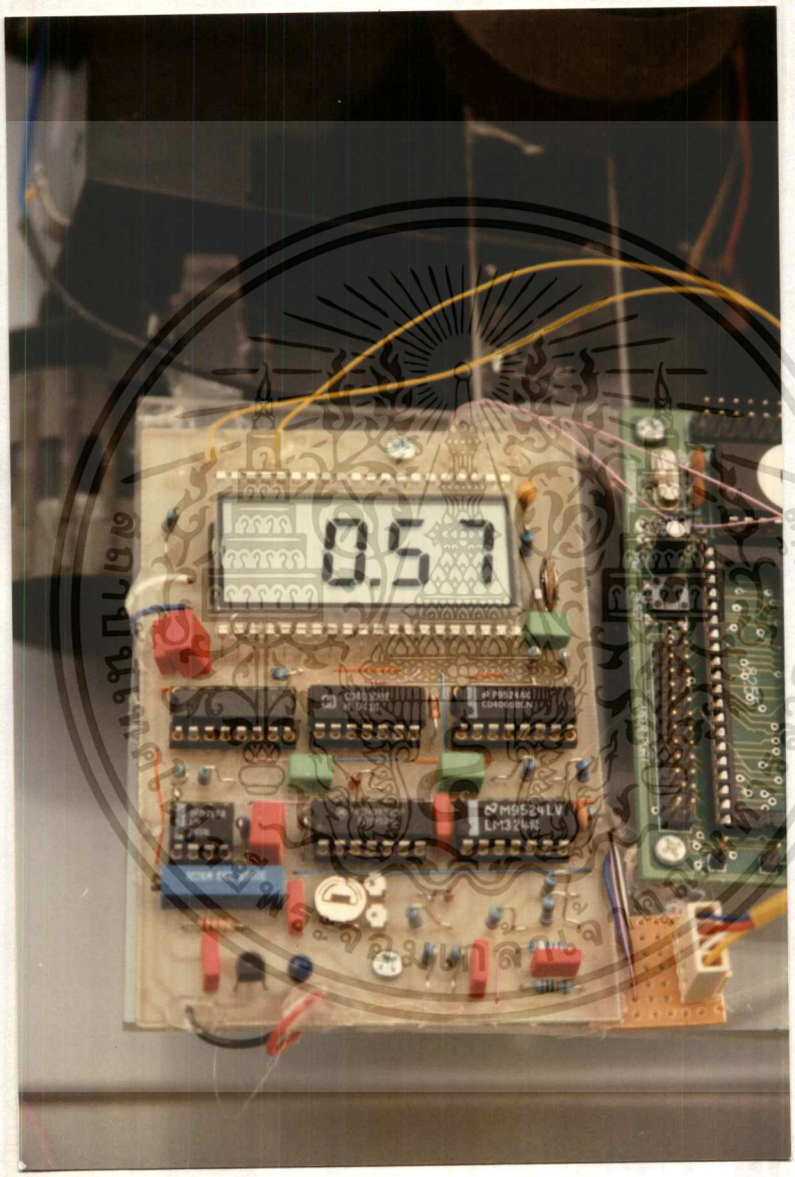
ในส่วนนี้ได้ทำการทดลอง ส่วนวัดระยะทางด้วยอุลตร้าโซนิค ซึ่งทำการทดลองวัดระยะห่างของกีดขวาง โดยกำหนดช่วงระยะทางในการวัดคือ 20 - 150 เซนติเมตร ซึ่งช่วงระยะนี้เป็นช่วงที่ใช้งานจริง โดยจากการทดลองจะได้ผลการวัดตามตารางที่ 14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 15 แสดงผลการสำรวจ

ระยะทางจริง	ระยะทางที่วัดได้
20	24
25	28
30	32
35	36
40	40
45	45
50	50
55	55
60	60
65	64
70	69
75	74
80	79
85	84
90	88
90	93
100	98
105	103
110	108
115	112
120	117
125	122
130	127
135	133
140	138
145	148
150	151

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 33 แสดงวงจรวัดระยะทางด้วยอุลตร้าโซนิค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองการทำงานของ Observation Mobile Robot

ในการทดลองการทำงานของ รถสำรวจจะมีขั้นตอนการทำงานดังนี้

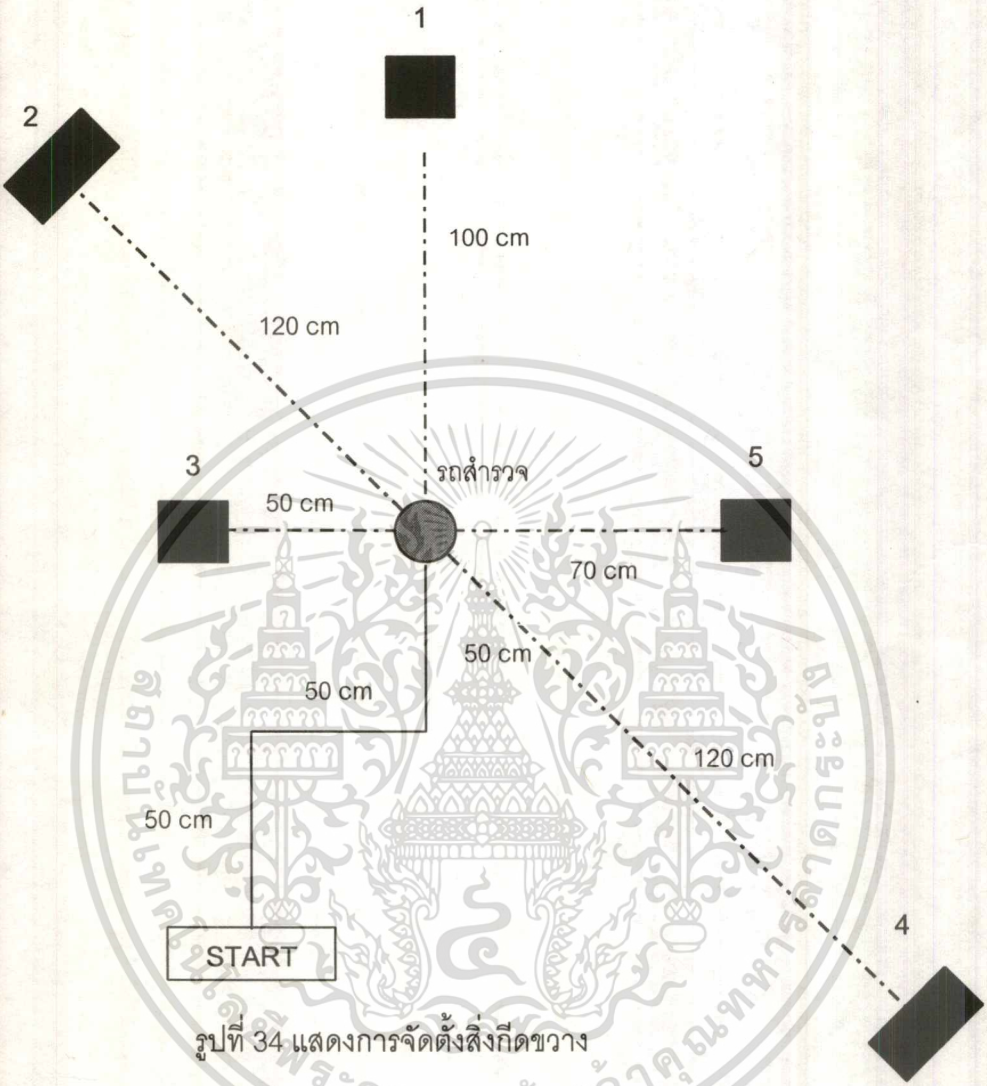
1. ทำการเรียกโปรแกรม SCAN.C และทำการคอมไพล์โปรแกรมดังกล่าว
2. การกำหนดเส้นทางการกำหนดเส้นทางสำรวจ โดยการป้อนทีละคำสั่งแล้วกด ENTER หลังสิ้นสุดแต่ละคำสั่ง และเมื่อสิ้นสุดชุดคำสั่ง จึงกด ENTER ซึ่งในการทดลองนี้เราจะทำการกำหนดเส้นทาง คือ เดินหน้า 50 เซนติเมตร เลี้ยวซ้าย 90 องศา เดินหน้า 50 เซนติเมตร เลี้ยวขวา 90 องศา เดินหน้า อีก 50 เซนติเมตร แล้วจึงทำการสำรวจสิ่งกีดขวางรอบตัว ซึ่งชุดคำสั่งที่จะป้อน คือ F2 L0 F2 R0 F2 S0 โดยแต่ละคำสั่งมีความหมายดังนี้

   F2 เดินหน้า 50 เซนติเมตร
   L0 เลี้ยวซ้าย 90 องศา
   F2 เดินหน้า 50 เซนติเมตร
   R0 เลี้ยวขวา 90 องศา
   F2 เดินหน้า 50 เซนติเมตร
   S0 ทำการสำรวจสิ่งกีดขวางรอบตัว

   โดยค่าไบต์แรก คือ คำสั่งที่กำหนดให้ทำงาน ค่าไบต์ที่ 2 ของแต่ละคำสั่งคือ ค่ารหัส ASCII ที่ใช้ในการบอกระยะทาง
3. เมื่อรถสำรวจเคลื่อนที่ไปถึงบริเวณที่จะทำการสำรวจ รถสำรวจก็จะทำการสำรวจจากนั้นเราก็สามารถที่จะดูผลของการสำรวจ ได้ที่จอภาพคอมพิวเตอร์ ในการทดลองนี้เราจะทำการทดลองโดยการสมมุติสิ่งกีดขวางขึ้น โดยจะมีการจัดตั้งสิ่งกีดขวางไว้ดังรูป 33

จากผลการสำรวจ ที่ปรากฏบนจอภาพคอมพิวเตอร์ จะได้ค่าตามตารางที่ 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 34 แสดงการจัดตั้งสิ่งกีดขวาง

สิ่งกีดขวาง(ชิ้นที่)	ระยะห่างจริง(cm)	ระยะที่วัดได้(cm)	มุมของสเต็ปปิ้งมอเตอร์
1	100	98	0
2	120	118	45
3	50	50	90
4	170	0	235
5	70	69	270

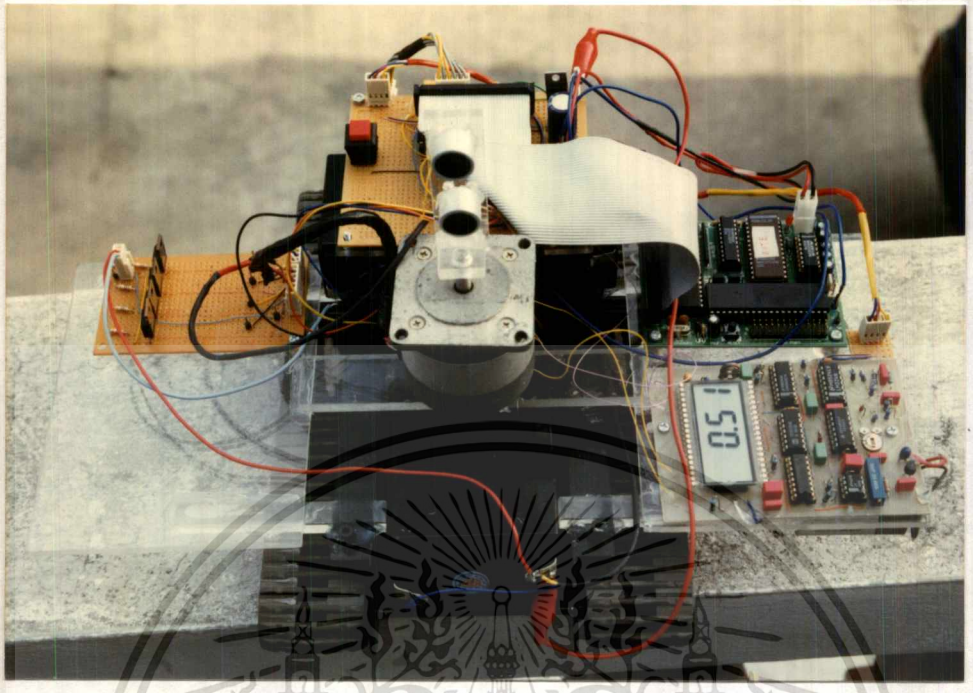
ตารางที่ 15 แสดงผลการสำรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

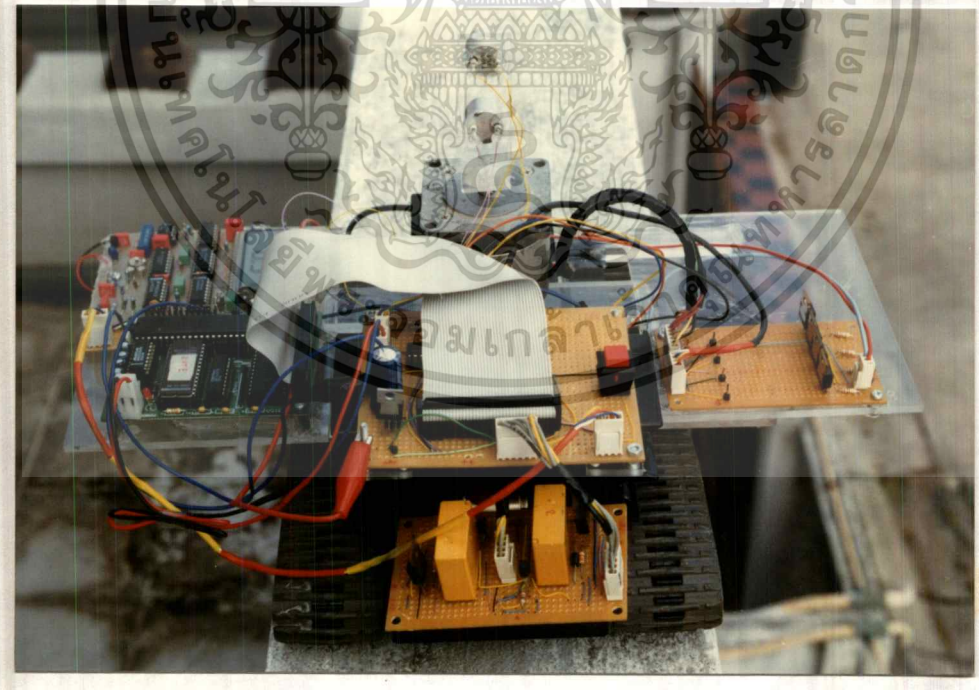
ในตารางที่ 15 แสดงถึงผลการสำรวจที่ได้จากการสำรวจและขอบเขตที่รถสำรวจสามารถสำรวจได้ โดยจากการสมมุติสิ่งกีดขวางขึ้น โดยจะมีการจัดตั้งสิ่งกีดขวางไว้ดังรูปที่ 33 เมื่อทำอ่านค่าบนจอภาพคอมพิวเตอร์ ก็จะพบว่า ที่มุม 0 องศา จะพบสิ่งกีดขวางที่ระยะ 98 cm ซึ่งระยะที่กำหนดไว้จริงคือ 100 cm ที่มุม 45 องศา จากจุดเริ่มต้น จะพบสิ่งกีดขวางที่ระยะ 117 cm ซึ่งระยะที่กำหนดไว้จริงคือ 120 cm ที่มุม 90 องศา จากจุดเริ่มต้น จะพบสิ่งกีดขวางที่ระยะ 50 cm ซึ่งระยะที่กำหนดไว้จริงคือ 50 cm ที่มุม 235 องศา จากจุดเริ่มต้น จะพบสิ่งกีดขวางที่ระยะ 0 cm ซึ่งระยะที่กำหนดไว้จริงคือ 170 cm การที่ผลเป็นเช่นนี้ เพราะในการปรับแต่งส่วนวัดระยะทางด้วยอุลตร้าโซนิค ได้ทำการปรับแต่งให้มีขอบเขตการใช้งานตั้งแต่ 20 - 150 cm ดังนั้นเมื่อมีสิ่งกีดขวางที่ระยะ 170 cm จึงไม่สามารถตรวจจับได้จึงแสดงผลเป็นค่าระยะ 0 cm ที่มุม 270 องศา จากจุดเริ่มต้น จะพบสิ่งกีดขวางที่ระยะ 69 cm ซึ่งระยะที่กำหนดไว้จริงคือ 70 cm ที่มุม 0 องศา จะพบสิ่งกีดขวางที่ระยะ 98 cm ซึ่งระยะที่กำหนดไว้จริงคือ 100 cm จากผลทดลองการทำงานในส่วนนี้จะเห็นได้ว่ารถสำรวจสามารถตรวจจับสิ่งกีดขวาง ที่อยู่ในระยะตั้งแต่ 20 - 150 cm ได้จริง

รูปที่ 35 แสดงตัวอย่างผลการสำรวจ ที่ปรากฏบนจอภาพคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 36 แสดงภาพรถสำรวจด้านหน้า



รูปที่ 37 แสดงภาพรถสำรวจด้านหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและวิจารณ์

ขอบเขตการทำงานของ Observation Mobile Robot ที่สมบูรณ์แบบ คือสามารถที่จะเคลื่อนที่ไปสำรวจในสถานที่ต่าง ๆ ได้โดยอาศัยการสื่อสารกับคอมพิวเตอร์ในแบบไร้สาย และสามารถที่จะทำการสำรวจทางกายภาพได้หลายลักษณะ ซึ่งจากโครงงานที่สร้างขึ้นนั้นปรากฏว่า Observation Mobile Robot สามารถเคลื่อนที่ได้ตามคำสั่งจริงและสามารถทำการสำรวจตำแหน่งที่ตั้งของสิ่งกีดขวาง และส่งข้อมูลผ่านสายข้อมูลกลับมาที่ คอมพิวเตอร์ ได้จริง แต่สำหรับส่วนที่ไม่ได้สร้างขึ้นในโครงงานนี้ คือ ส่วนการสื่อสารข้อมูลกับคอมพิวเตอร์ในแบบไร้สาย สำหรับ ในส่วนที่สร้างขึ้นมานั้น ซึ่งจะรวมทั้ง Software และ Hardware มีข้อบกพร่องด้วยกันหลายจุดคือ

- ส่วน ENCODER มีปัญหาคือ ยังไม่แม่นยำเท่าที่ควร เพราะแนวการหมุนของจานตัดแสง ไม่ได้ฉาก กับ Photo Detector ดังนั้นจึงควรแก้ไขโดยการสร้างและติดตั้ง ENCODER ให้ได้ฉาก
- น้ำหนักของตัว Observation Mobile Robot มีน้ำหนักมาก เนื่องมาจากต้องใช้แบตเตอรี่ 6 V ถึง 2 ก้อน จึงทำให้มีน้ำหนักมาก ควรแก้ไขโดยการเปลี่ยนมาใช้แบตเตอรี่ 12 V เพียง 1 ก้อน ก็เพียงพอ
- ในส่วนของ Software ภาษา C ที่ คอมพิวเตอร์ ควรจะเขียนให้ใช้งานได้ง่ายกว่านี้ และที่ 8051 ก็ควรจะมี Option ในการปฏิบัติงานที่เพิ่มขึ้น เช่น สามารถเดินทางกลับมาที่จุดเริ่มต้นได้ เมื่อทำการสำรวจเสร็จ เป็นต้น

เนื่องจากโครงงานนี้เป็นเครื่องต้นแบบเพื่อนำเสนอแนวคิดที่จะเพิ่มปลอดภัย ในการเข้าไปสำรวจในสถานที่ที่มีอันตรายแทนมนุษย์ จึงอาจยังไม่สมบูรณ์แบบ และยังมีสิ่งที่ต้องพัฒนาอีกหลายจุด เพื่อให้ใช้งานจริงได้อย่างมีประสิทธิภาพ คือ

1. ต้องดัดแปลงให้สามารถส่งข้อมูลแบบ 2 ทาง ไปกลับ แบบไร้สายได้ ซึ่งมีแนวคิดคือ สามารถทำได้โดยการใช้ ไอซี เบอร์ XR2206 เป็นตัวมอดดูเลต และ XR2211 เป็นตัวดีมอดดูเลต สัญญาณดิจิตอลเข้ากับ สัญญาณความถี่ FM
2. สามารถใช้งานได้ที่ความเร็วของข้อมูลสูง
3. มีวิธีการตรวจสอบข้อมูลที่ผิดพลาด ซึ่งมีแนวคิดคือ ทำการปรับปรุงโปรโตคอลที่ใช้ในการสื่อสาร
4. ทำให้ Software มีความละเอียดและมีความสามารถเพิ่มขึ้น ซึ่งมีแนวคิดคือ การทำให้สามารถ Observation Mobile Robot จำเส้นทางได้ ซึ่งอาจจะใช้การทำงานแบบสแตก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ขยายขีดจำกัดทางด้าน Ultrasonic ซึ่งมีแนวคิดคือ ควรทำปรับอย่างละเอียด โดยให้ตรงกับช่วงที่จะใช้งานที่สุด
6. ขยาย Parallel Port เพื่อที่จะใช้ติดต่อกับ Sensor ในการสำรวจเพิ่มขึ้น ซึ่งมีแนวคิดคือ การเพิ่มไอซี 8255 เข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <graphics.h>
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<io.h>
#include<dos.h>
#include<ctype.h>

char ch, ms[15],st[4],c,recieve[120] ;
int ch_number(),hostlink(),recieve_ir();
int serial();
int data[150],a,b,i,j,Rad;
char TEMP[15];
int init_p(int port,unsigned char code);
unsigned char transmis[15];
FILE *fp;
union REGS r;
void init_graph();
int Radar(int Rad,int distance);
main()
   {
    do
        { clrscr();

          init_graph();
          setfillstyle(SOLID_FILL,RED);
          bar(0,0,639,479);
          Rad=0;
          gotoxy(27,25);printf(" OBSERVATION MOBILE ROBOT ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        gotoxy(1,1);
        printf("enter your message: \n");
        recieve_ir();
        i=0;
/*******************CHANGE STANDARD INPUT TO HEXCODE**********/
        do
        {
         switch(ms[i])
          { case'F':TEMP[i]='1';
                    i++;
                    break;

            case'S':TEMP[i]='4';
                    i++;
                    break;


            case'L':TEMP[i]='2';
                    i++;
                    break;

            case'R':TEMP[i]='3';
                    i++;
                    break;

            default:printf(" check your messege");
                    getch();
                    exit(2);
          }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                exit(2);
            }
             ch_number();
        }while((ms[i])!='\0');
        TEMP[i]= 13 ;
        printf("message:%s\n",ms);

/***************KEEP DATA IN FILE AND SEND TO PORT**********/
        if(!(fp=fopen("data.c","w")))
          {
              printf(" cannot open file ");
              getch();
              exit(1);
           }
              i=0;
          do
          {
              fprintf(fp,"%c",TEMP[i]);
              i++;
          } while((TEMP[i])!='\0');
        fclose(fp);

        b=0;
        a=0;
        serial();

        }while(getch()!=27);
        closegraph();
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*********************SERIAL PORT INTERFACE*********************/
serial()
        {

         if((fp=fopen("data.c","r"))==NULL)
         {
           puts("cannot open file\n ");

           exit(1);
           }
           while(feof(fp)==0)
           fread(transmis,sizeof(transmis),1,fp);
           init_p(0,227);
          // printf("data recieve :\n");
           while((!kbhit()))
           {
           a=hostlink();
           if(Rad>=360)Rad=0;
           else Rad++;
           sscanf(&recieve[0],"%01c",&data[0]);
           Radar(Rad,data[0]);
           }
           fclose(fp);

           }

/******************* INITIAL SERIAL ************************/

            int init_p(int port,unsigned char code)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 r.x.dx=0;
 r.h.ah=0;
 r.h.al=227;
 int86(0X14,&r,&r);
}
/*************************************************************/
int hostlink()
{
 if(b!=1)
 {
 do{
   r.x.dx=0;
   r.h.ah=1;
   r.h.al=transmis[a];
   int86(0x14,&r,&r);
   a++;
  }while((r.h.al!='\0'));
 b=1;
// gotoxy(1,5);    printf("data recieve");
// delay(2000);


 }
 i=0;
 j=0;
 gotoxy(1,5);



   r.h.ah=2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            int86(0x14,&r,&r);
            if(r.h.al!=0)
            {
            recieve[j]=r.h.al;
            gotoxy(3,25); printf("data recieve:%d\n",recieve[j]);
            j++;
            }

          }
/*************************************************************/

   int ch_number()
   {int n,a ;
    for(a=0;a<1;a++)
     {
     TEMP[i] = ms[i];
     i++;
     }
   }
/*************************************************************/
void init_graph()
{ int gdrive=DETECT,gmode;
  initgraph(&gdrive,&gmode,"c:\\tc3\\bgi");
  if(graphresult()!=grOk)
  { gotoxy(10,10);printf("Graphic Error !");
    getch();exit(0);
  }
}
int Radar(int Rad,int distance)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{ int x,y;
  x=300;y=200;
  if(distance>200)distance=200;
  else if(distance<=0)distance=100;
  pieslice(x,y,Rad,Rad+1,2*distance);
  return 0;
}
/*****************************************************************/
    recieve_ir()
    {
          do
          {
           gets(st);
           if((*st)!=NULL)
           for(j=0;j<=1;j++)
           { ms[i]=st[j];
             i++;
           }
         //  printf("\n");
           }while((*st)!=NULL);
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*********************************************************************
; 8051 PROGRAM FOR OBSERVATION MOBILE ROBOT
;*********************************************************************
DC        EQU    08H
LF        EQU    09H
RT        EQU    10H
SEND      EQU    11H
CNTBUF EQU    12H
INTFAG   EQU    16H
FW        EQU    19H
RV        EQU    20H
IN_ADD  EQU    21H
RC        EQU    22H
RBIT      EQU    23H
LBIT      EQU    24H
COMP   EQU    26H

      ORG    00H

      AJMP   START

      ORG    03H
      AJMP    INT_CNT

      ORG    23H
      LJMP    INT_SER
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; ********** MAIN **********

        ORG    40H

START:      MOV    SCON,#01010000B ; Set Mode TO 8-bit  UART
            MOV    P1,#0
            SETB   EA          ; Enable All Interrupt
            SETB   ES
            SETB   REN
            MOV    PCON,#0H    ; Clear SMOD
            MOV    TMOD,#00101101B ; Set mode timer to 8-bit auto reload
            MOV    TH1,#0FDH   ; Generate 9600 baud at 11.059 MHz
            MOV    TL1,#0H
            SETB   TR1         ; Start Timer 1
            CLR    IT0         ; INT 0 ENABLE
            CLR    RC
            LCALL  CLEAN

;=============================================================
;   INTERPRETE SERIAL DATA TO OPERATE
;   #31H = FORWARD   AND   NEXT  BYTE IS DISTANCE
;   #32H = TURN LEFT         #33H = TURN RIGHT
;   #34H = ULTRASONICS SCAN    #0DH = STOP & END OF SERIAL FILE
;=============================================================

OO:     JNB     RC,OO
        MOV     R0,#080H

AA:     MOV     A,@R0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CJNE    A,#031H,BB
        LCALL   FOR

BB:     MOV     A,@R0
        CJNE    A,#032H,CC
        LCALL   LEFT

CC:     MOV     A,@R0
        CJNE    A,#033H,DD
        LCALL   RIGHT

DD:     MOV     A,@R0
        CJNE    A,#034H,EE
        LCALL   SCAN_BYTE

EE:     MOV     A,@R0
        CJNE    A,#0FFH,EX
        LCALL   PAUSE

EX:     INC     R0
        SJMP    AA

;=============================================================

INT_SER: PUSH   ACC
         PUSH   PSW

RECEIV:  MOV    A,SBUF
         MOV    @R0,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        INC^   R0

        CJNE   A,#0FFH,NO_END
        SETB   RC
       CLR    RI
       POP    PSW
       POP    ACC
         RETI

NO_END:  CLR    RI
         SJMP   ENDSER

ENDSER:  POP    PSW
         POP    ACC
         RETI

;=================================================================

FOR:   SETB  P1.2
       SETB  P1.3
       SETB  P1.4
       SETB  P1.5
       INC   R0            ; READ NEXT DATA FOR DISTANCE
       MOV   FW,@R0

POLLING: JB    P1.0,POLLING
ZERO:    JNB   P1.0,ZERO
         DJNZ  FW,NEXT
         SJMP  OUT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
NEXT:    SJMP   POLLING
OUT:     RET

;=============================================================

LEFT:  SETB   P1.2
       SETB   P1.3
              CLR    P1.4
              CLR    P1.5
              INC    R0
              MOV    LF,@R0 ; #03FH

POLLING2: JB     P1.0,POLLING2
ZERO2:    JNB    P1.0,ZERO2
          DJNZ   LF,NEXT2
          SJMP   OUT2

NEXT2:    SJMP   POLLING2
OUT2:     RET

;=============================================================

RIGHT: CLR    P1.2
       CLR    P1.3
       SETB   P1.4
       SETB   P1.5
       INC    R0
       MOV    RT,@R0 ; 03FH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
POLLING3: JB    P1.1,POLLING3
ZERO3:    JNB   P1.1,ZERO3
          DJNZ  RT,NEXT3

          SJMP  OUT3

NEXT3:    SJMP  POLLING3
OUT3:     RET

;=============================================================

PAUSE:  CLR   P1.2
        CLR   P1.3
        CLR   P1.4
        CLR   P1.5
        LJMP  START

        RET

;=============================================================

 SCAN_BYTE:
        LCALL  CLEAR
        SETB   EX0          ; INT 0 ENABLE 31

;=============================================================
; THIS SUBROUTINE IS USED TO SEND SCANNED DATA FROM ULTRASONIC IN ONE ;
BYTE PER ONE SCAN AND RUN STEPS NEXT .
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;=============================================================

        MOV   LF,#035H      ; INITIAL NUMBER OF STEPS
        MOV   RT,#035H
        MOV   DC,#070H      ; **

TRANSMIT: JNB   INTFAG,TRANSMIT

          CLR   EX0         ; CLEAR INTERRUPT FOR RUN STEPS

;=============================================================
          DJNZ  DC,WHAT
          LJMP  STOP
WHAT:
       JB    LBIT,L
       JB    RBIT,R

R:     DJNZ  RT,RR
       SETB  LBIT        ; SET 50(32H) X 4 STEPS = 200 STEPS
       CLR   RBIT
       SJMP  YY

RR:    LCALL CLEAR
       LCALL STEP_RIGHT
       LCALL CLEAR
       SJMP  YY

L:     DJNZ  LF,LL
       SETB  RBIT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CLR   LBIT
        SJMP  YY

LL:     LCALL CLEAR
        LCALL STEP_LEFT
        LCALL CLEAR
        SJMP  YY

;===============================================================

YY:     JNB   IE0,YY

        SETB  EX0
        MOV   TL0,#0          ; *
        CLR   INTFAG
        SJMP  TRANSMIT

;=====================================
;    EXTERNAL INTERRUPT SUB.
;=====================================

INT_CNT: SETB  TR0            ; INTERRUPT SUBROUTINE
         MOV   CNTBUF,TL0
         SETB  INTFAG         ; *
         JB    SEND,GO
         MOV   SBUF,CNTBUF

WAIT:    JNB   TI,WAIT
         SETB  SEND
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CLR   TI

GO:     CLR   SEND

XX:     JB    IE0,XX
        MOV   TL0,#0

        RETI

;=============================================================
; THIS PROGRAM IS USED TO CONTORL STEPPING MOTOR BY SENDING
;SEQUENCE OUT TO PORT 2 .
;=============================================================

STEP_RIGHT:

MAIN_R:  MOV  R3,#0

START_R: MOV  A,#0001000B

SND1_R:
         MOV    R1,#0FH   ; CLEAR PORT 1
SND_R:   MOV    P2,#0
         DJNZ   R1,SND_R
         MOV    P2,A
         MOV    R2,#0FFH  ; *
DX:      DJNZ   R2,DX     ; *
         INC    R0
         RR     A         ; SHIFT RIGHT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        INC     R3
        CJNE    R3,#04H,GO_R
           RET

GO_R:  CJNE    A,#10000000B,DE_R
        LCALL   DELAY
        SJMP    START_R

DE_R:  LCALL   DELAY
       SJMP    SND1_R

;=============================================================

STEP_LEFT:

MAIN_L:  MOV   R3,#0

START_L: MOV   A,#01H

SND1_L: MOV    R1,#0FH
SND_L:  MOV    P2,#0
        DJNZ   R1,SND_L
        MOV    P2,A
        MOV    R2,#0FFH   ; *
DDX:    DJNZ   R2,DDX     ; *
        INC    R0
        RL     A          ; SHIFT LEFT
        INC    R3
        CJNE   R3,#04H,GO_L
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            RET
GO_L:   CJNE    A,#010H,DE_L
        LCALL   DELAY
        SJMP    START_L

DE_L:   LCALL   DELAY
        SJMP    SND_L

;=============================================================

DELAY:
        MOV     R2,#0FFH    ; DELAY FOR EACH STEP
  D:    MOV     R1,#01FH
 D1:    DJNZ    R1,D1
 D2:    DJNZ    R2,D

        RET

;=============================================================

CLEAN:  MOV     R0,#80H

CLN:    MOV     @R0,#0
        INC     R0
        CJNE    @R0,#0,CLN

    RET

;=============================================================
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CLEAR:  MOV     A,#0
        MOV     R0,#0
        MOV     R1,#0
        MOV     R2,#0
        MOV     R3,#0
        MOV     R4,#0
        MOV     R5,#0
        MOV     R6,#0
        MOV     R7,#0
        MOV     P1,#0

            RET

;=================================================================

STOP:   MOV     P2,#0H

        LJMP    START

            END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. สุนทร วิทูสุรพจน์ ; การใช้งานไมโครคอนโทรลเลอร์ตระกูล 8051 , ซีเอ็ดยูเคชั่น , พ.ศ. 2537 หน้า 1 - 106
2. Motorola Inc ; Motorola Optoelectronics Device Data , 1993 , page
3. วารสาร เซมิคอนดัคเตอร์ อิเลคทรอนิคส์ เล่มที่ 129

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้