

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์กู้ภัย

ROBOT RESCUE



จัดทำโดย

นายกรวิทย์ เสวตนันท์

นายจักรพงษ์ แสงนิล

ฉพ.
ก182 ๓
2547

เลขหมู่.....
เลขทะเบียน 61412
วัน,เดือน,ปี 17 ก.ค. 2548

b. 1159 6090
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์กู้ภัย
ROBOT RESCUE

จัดทำโดย

นายกรวิทย์ เศวตนันท์ รหัส 45015267

นายจักรพงศ์ แสงนิล รหัส 45015271



อาจารย์ที่ปรึกษา

รศ.ดร. สุรพันธุ์ เอื้อไพฑูริย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ ปีการศึกษา 2547

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง หุ่นยนต์กู้ภัย

ROBOT RESCUE

ผู้จัดทำ

1. นายกรวิทย์ เสวตนันท์
2. นายจักรพงษ์ แสงนิล




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานเรื่อง หุ่นยนต์กู้ภัย
ROBOT RESCUE
จัดทำโดย นายกรวิทย์ เสวตนันท์ รหัส 45015267
นายจักรพงษ์ แสงนิล รหัส 45015271
อาจารย์ที่ปรึกษา รศ.ดร.สุรพันธุ์ เอื้อไพบูลย์



รายงานฉบับนี้ได้ผ่านการตรวจสอบจากอาจารย์ที่ปรึกษาแล้ว

ลงชื่อ  อาจารย์ที่ปรึกษา
(รศ.ดร.สุรพันธุ์ เอื้อไพบูลย์)
วันที่...../...../.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์กู้ภัย

นายกรวิทย์ เสวตนันท์ 45015267

นายจักรพงษ์ แสงนิล 45015271

รศ.ดร. สุรพันธุ์ เอื้อไพฑูริย์ (อาจารย์ที่ปรึกษา)

ปีการศึกษา 2547

บทคัดย่อ

ในปัจจุบันหุ่นยนต์ได้เข้ามามีบทบาทแทนมนุษย์ในการทำงานที่เสี่ยงภัย หรือในสถานที่อันตราย ดังนั้นจึงจำเป็นอย่างยิ่งที่หุ่นยนต์จะต้องมีประสิทธิภาพ โดยสามารถตอบสนองความต้องการของมนุษย์ได้ทุกอย่าง สามารถควบคุมการทำงานของหุ่นยนต์ในระยะไกลๆ ได้ และสามารถส่งข้อมูลกลับมาได้ครบถ้วน

ในโครงการนี้เราสามารถสร้างหุ่นยนต์ที่สามารถควบคุมระยะไกล โดยใช้จอยสติคควบคุมผ่านคอมพิวเตอร์และผ่านคลื่นวิทยุ โดยมีการติดตั้งตัวรับส่งอัลตราโซนิก อุปกรณ์เซนเซอร์ความร้อนและเสียง แล้วยังมีการติดตั้งกล้อง เพื่อส่งข้อมูลต่างๆ กลับมายังคอมพิวเตอร์อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ROBOT RESCUE

Mr. Korrawit Sawettanun 45015267

Mr. Jakkaphong Sangnil 45015271

Assoc.Prof. Surapan Airphaiboon (Adviser)

Academic 2004

Abstract

In the present , a robot has many roles in doing dangerous tasks instead of human. Also the robot need to have efficiency and should be able to response all human being's desirability such as being controlled wirelessly and return data correctly.

In this thesis, we can built the remote-controlled robot controlled by joystick through computer and radio wave . Ultrasonic sender and receiver, temperature and sound sensor devices were installed in order to survey area and it has a camera which is installed to send the data back to the control station.

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	I
Abstract	II
สารบัญ	III
สารบัญรูปภาพ	V
สารบัญตาราง	VII
บทที่ 1 บทนำ	
1.1 วัตถุประสงค์	1
1.2 ขอบเขตของโครงการ	2
1.3 ประโยชน์ที่ได้รับจากโครงการ	2
1.4 รายละเอียดโดยย่อของโครงการ	2
1.5 รายละเอียดของปริญญานิพนธ์	3
บทที่ 2 ทฤษฎีและหลักการ	
2.1 ไมโครคอนโทรลเลอร์ MCS-51	4
2.2 การใช้งานไมโครคอนโทรลเลอร์กับพอร์ตอนุกรม	14
2.3 การสื่อสารข้อมูล	20
2.4 ดีซีมอเตอร์ (DC Motor)	25
2.5 เพาเวอร์มอสเฟต	26
2.6 การเชื่อมต่อทางแสง	30
2.7 หลักการออกแบบส่วนประกอบและโครงสร้างต่างๆ	30
บทที่ 3 การออกแบบส่วนต่างๆ ของหุ่นยนต์	
3.1 โครงสร้างการทำงานของหุ่นยนต์	34
3.2 การออกแบบทางด้านวงจรควบคุมและโครงสร้างหุ่นยนต์ (Hardware)	35
3.2.1 โครงสร้างตัวถังของหุ่นยนต์	35
3.2.2 วงจรการเชื่อมต่อของ MCS-51	37
3.2.3 วงจรควบคุมการสื่อสารระหว่างคอมพิวเตอร์กับหุ่นยนต์	38
3.2.4 วงจรควบคุมอุปกรณ์ต่างๆ	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
3.2.5 วงจรควบคุมฐานของกล้อง	41
3.2.6 วงจรควบคุมดีซีมอเตอร์	42
3.3 การออกแบบทางด้าน โปรแกรม (Software)	44
3.3.1 ส่วนควบคุมการเคลื่อนที่ของหุ่นยนต์	44
3.3.2 ส่วนควบคุมการส่งข้อมูลผ่านพอร์ทอนุกรม	46
3.3.3 ส่วนควบคุมฐานกล้อง	46
บทที่ 4 ผลการทดสอบ	
4.1 การเคลื่อนที่ของหุ่นยนต์	47
4.2 การบังคับทิศทางของฐานกล้อง	47
4.3 การแสดงภาพบนหน้าจอคอมพิวเตอร์	48
4.4 การใช้อุปกรณ์ต่างๆ	48
4.5 การทดลองในสถานที่ที่จำลองจากเหตุการณ์จริง	49
4.6 สนามที่ใช้แข่งขันในรอบสุดท้าย	51
บทที่ 5 สรุปและวิจารณ์	52
ภาคผนวก	53
ROBOT RESCUE V1.0 BY VB.6	54
ROBOT RESCUE V1.0 BY MCS-51	60
กิตติกรรมประกาศ	ก.
เอกสารอ้างอิง	ข.

สารบัญรูปลูกภาพ

หน้า

รูปที่ 2.1 แสดงโครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51	5
รูปที่ 2.2 แสดงรูปร่างและการจัดวางขาต่างๆ ของไมโครคอนโทรลเลอร์ MCS-51	6
รูปที่ 2.3 แสดงการใช้หน่วยความจำสำหรับเก็บ โปรแกรม	7
รูปที่ 2.4 แสดงการจัดพื้นที่ของหน่วยความจำโปรแกรมภายในและภายนอก	7
รูปที่ 2.5 แสดงหน่วยความจำสำหรับเก็บข้อมูลภายในไมโครคอนโทรลเลอร์ MCS-51	8
รูปที่ 2.6 แสดงการจัดหน่วยความจำข้อมูล	9
รูปที่ 2.7 แสดงการต่อกับหน่วยความจำข้อมูลภายนอกไอซี	10
รูปที่ 2.8 แสดงการเลือกใช้รีจิสเตอร์ใช้งานทั่วไป R0-R7 ในแต่ละกลุ่ม	11
รูปที่ 2.9 ตารางแสดงถึงหน้าที่ของขาต่าง ๆ	15
รูปที่ 2.10 แสดงสัญญาณต่างๆที่ส่งในรูปแบบอนุกรม	16
รูปที่ 2.11 แสดงการเชื่อมต่อของสายสัญญาณ	18
รูปที่ 2.12 รูปคลื่นของสัญญาณที่ส่ง	19
รูปที่ 2.13 แสดงโครงสร้างภายในและตำแหน่งขาต่างๆของ Max232	20
รูปที่ 2.14 แสดงการรับส่งข้อมูลแบบทางเดียว (Simplex)	21
รูปที่ 2.15 แสดงการรับส่งข้อมูลสวนทางกันได้แบบผลัดการส่ง (Half Duplex)	22
รูปที่ 2.16 แสดงการรับส่งข้อมูลแบบสวนทางกันได้พร้อมกัน (Full Duplex)	22
รูปที่ 2.17 แสดงการส่งข้อมูลแบบขนาน	23
รูปที่ 2.18 แสดงการส่งข้อมูลแบบอนุกรม	23
รูปที่ 2.19 แสดงการส่งข้อมูลแบบสัมพันธ์	24
รูปที่ 2.20 แสดงการส่งข้อมูลแบบไม่สัมพันธ์	24
รูปที่ 2.21 โครงสร้างของเพาเวอร์มอสเฟต	27
รูปที่ 2.22 แรงดันพังทลายที่ตรงกับซอสเมื่ออุณหภูมิของรอยต่อเปลี่ยนไป	27
รูปที่ 2.23 แสดงตัวอย่างของ Gate Charge Chart	28
รูปที่ 2.24 (ก) แสดงการชาร์ตประจุที่ขาเกตตามเวลาที่มีผลต่อการนำกระแส	29
รูปที่ 2.24 (ข) แสดงผลเมื่อเริ่มหยุดนำกระแสของเพาเวอร์มอสเฟต	29
รูปที่ 3.1 แสดงโครงสร้างการทำงานหลักๆ ของหุ่นยนต์	34
รูปที่ 3.2 แสดงโครงสร้างการทำงานโดยละเอียดของหุ่นยนต์	34
รูปที่ 3.3 แสดงโครงสร้างของหุ่นยนต์มองจากด้านข้าง	35
รูปที่ 3.4 แสดงโครงสร้างของหุ่นยนต์มองจากด้านบน	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.5 แสดงโครงสร้างของหุ่นยนต์บีมองจากด้านบน	36
รูปที่ 3.6 แสดงการต่ออุปกรณ์ภายนอกกับไมโครคอนโทรลเลอร์	37
รูปที่ 3.7 แสดงวงจรภาคส่งสัญญาณจากคอมพิวเตอร์ไปยังไมโคร	38
รูปที่ 3.8 แสดงวงจรภาครับสัญญาณจากไมโครไปยังไมโครคอนโทรลเลอร์	38
รูปที่ 3.9 แสดงวงจรเปิด-ปิดไฟส่องทาง	39
รูปที่ 3.10 แสดงวงจรควบคุมฐานของกล้อง	40
รูปที่ 3.11 แสดงวงจรควบคุมดีซีมอเตอร์	41
รูปที่ 3.12 แสดงตำแหน่งของปุ่มต่างๆ บนจอยสติค	43
รูปที่ 3.13 โฟลว์ชาร์ตแสดงการควบคุมทิศทางการเคลื่อนที่ของหุ่นยนต์	44
รูปที่ 3.14 โฟลว์ชาร์ตแสดงการควบคุมฐานกล้อง	46
รูปที่ 4.1 แสดงการบังคับทิศทางของฐานกล้อง	47
รูปที่ 4.2 แสดงภาพหน้าต่างของโปรแกรม	48
รูปที่ 4.3 แสดงการใช้ไฟส่องทาง	48
รูปที่ 4.4 แสดงสภาพของสนามที่จำลองจากเหตุการณ์จริง	49
รูปที่ 4.5 แสดงการค้นหาเหยื่อผู้เคราะห์ร้าย	49
รูปที่ 4.6 แสดงภาพหน้าจอของมอเตอร์ต่างๆ ขณะที่หุ่นยนต์กำลังทำงาน	49
รูปที่ 4.7 แสดงอุปกรณ์ทั้งหมดที่ใช้ในการปฏิบัติการกู้ภัย	50
รูปที่ 4.8 แสดงการวัดอุณหภูมิจากเหยื่อผู้เคราะห์ร้าย	50
รูปที่ 4.9 แสดงสภาพสนามระดับความยากน้อยที่สุด	51
รูปที่ 4.10 แสดงสภาพสนามระดับความยากปานกลาง	51
รูปที่ 4.11 แสดงสภาพสนามระดับความยากมากที่สุด	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 3.1 แสดงการทำงานของวงจรควบคุมฐานของกล้อง	41
ตารางที่ 3.2 แสดงการทำงานของวงจรควบคุมการเคลื่อนที่ของหุ่นยนต์	42



บทที่ 1 บทนำ

ในปัจจุบันพบว่าเทคโนโลยีใหม่ๆ มีบทบาทต่อการพัฒนาสังคม หรือแม้แต่การพัฒนา ประเทศ เนื่องจากเทคโนโลยีเป็นพื้นฐานสำคัญอย่างหนึ่งในการพัฒนาทางด้านต่าง ๆ ได้แก่ ด้านการ แพทย์ การศึกษา การสื่อสารคมนาคม และการรักษาความปลอดภัย เป็นต้น ซึ่งเทคโนโลยีที่ได้มีการพัฒนาขึ้นมาใหม่ ๆ นี้ก็มีจำนวนมากมายและมีความหลากหลาย แต่ในที่นี่ เราจะกล่าวถึงเทคโนโลยีที่กำลังได้รับความสนใจมากในตอนนี้คือ หุ่นยนต์ ซึ่งเราจะสังเกตเห็นได้จากสื่อหลาย ๆ แขนง และบุคคลทุกเพศทุกวัยให้ความสนใจเป็นจำนวนมาก ซึ่งหุ่นยนต์ที่มีในปัจจุบันก็มีการใช้งานอยู่หลายประเภท เช่น หุ่นยนต์ที่ใช้ในโรงงานอุตสาหกรรม หุ่นยนต์ที่เป็นของเล่น และหุ่นยนต์กู้ภัย เป็นต้น

ประโยชน์ของหุ่นยนต์ก็จะขึ้นอยู่กับประเภทของหุ่นยนต์ โดยในที่นี้เราจะกล่าวถึง หุ่นยนต์กู้ภัย ซึ่งจะมีประโยชน์ทางการช่วยเหลือหรือการสำรวจพื้นที่ที่เสี่ยงอันตรายซึ่งมนุษย์ไม่สามารถเข้าไปสำรวจได้ ในส่วนของการช่วยเหลือหุ่นยนต์จะช่วยเหลือในด้านการค้นหา และบอกสภาพของผู้บาดเจ็บหรือผู้รอดชีวิตในพื้นที่ที่เกิดภัยพิบัติหรือภัยพิบาศซึ่งเกิดจากธรรมชาติหรือเกิดจากการก่อการร้าย ดังนั้นหุ่นยนต์กู้ภัยจึงจำเป็นอย่างมากในปัจจุบัน ทำให้ต้องมีการวิจัยและพัฒนาหุ่นยนต์ประเภทนี้เป็นอย่างยิ่ง นอกจากประโยชน์ที่ได้รับดังที่กล่าวไว้ข้างต้นแล้ว หุ่นยนต์ประเภทนี้ ยังทำทหายความสามารถและต้องใช้ความรู้หลายๆ แขนงในการสร้าง อีกทั้งยังได้สัมผัสกับอุปกรณ์ที่เป็นเทคโนโลยีใหม่ๆ และไม่เคยพบเห็นได้จากห้องเรียน จึงทำให้คณะผู้จัดทำมีความสนใจที่จะนำหุ่นยนต์กู้ภัยตัวนี้มาเป็นโครงการ ซึ่งโครงการนี้อาจจะยังไม่สามารถนำไปใช้งานจริง ๆ ได้ เนื่องจากอุปกรณ์ที่นำมาสร้างยังไม่เหมาะสมกับสถานการณ์จริงๆ แต่ก็สามารถนำไปพัฒนาให้ใช้งานจริงๆ ได้เช่นกัน

1.1 วัตถุประสงค์

1. เพื่อศึกษาหลักการและประยุกต์ใช้งานควบคุมมอเตอร์
2. เพื่อศึกษาหลักการและประยุกต์ใช้งานติดต่อระหว่างคอมพิวเตอร์กับอุปกรณ์อิเล็กทรอนิกส์
3. เพื่อศึกษาหลักการและประยุกต์ใช้งานอุปกรณ์เซนเซอร์ต่างๆ
4. เพื่อศึกษาหลักการและประยุกต์ใช้งานการสร้างโครงสร้างตัวถังของหุ่นยนต์
5. เพื่อศึกษาหลักการและประยุกต์ใช้งานอุปกรณ์ที่เป็นเทคโนโลยีใหม่ๆ
6. เพื่อศึกษาวิธีการเขียนโปรแกรมควบคุมหุ่นยนต์ด้วยไมโครคอนโทรลเลอร์ MCS-51
7. เพื่อศึกษาวิธีการเขียนโปรแกรมด้วย Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 ขอบเขตของโครงการ

หุ่นยนต์กู้ภัย จะสามารถเคลื่อนที่ได้ทุกสภาพผิวและยังสามารถเคลื่อนที่เข้าไปในพื้นที่ที่ไม่มีแสงสว่างได้ เนื่องจากการติดตั้งไฟส่องทางไว้ด้วย โดยจะมีการส่งภาพจากกล้องที่ติดตั้งอยู่บนหุ่นยนต์ไปแสดงบนจอมอนิเตอร์ของคอมพิวเตอร์ ส่วนการควบคุมการเคลื่อนที่จะควบคุมด้วยจอยสติค โดยจะสื่อสารข้อมูลกันแบบไร้สายระหว่างหุ่นยนต์กับคอมพิวเตอร์ ซึ่งจะสามารถสื่อสารได้ในระยะทาง 30-100 เมตร ในส่วนของกล้องจะสามารถปรับมุมมองได้ 2 ระดับคือ แนวนอนและแนวตั้งและสามารถตรวจวัดอุณหภูมิโดยไม่ต้องสัมผัสกับวัตถุหรือเหยื่อผู้เคราะห์ร้ายได้ ซึ่งค่าของอุณหภูมิจะแสดงค่าเป็นตัวเลข มีหน่วยเป็นองศาเซลเซียส ผ่านจอมอนิเตอร์ ส่วนของการตรวจสอบเสียงของเหยื่อผู้เคราะห์ร้ายและเสียงอื่นๆจะใช้ไมล์ไร้สาย และส่วนของการบอกตำแหน่งของวัตถุหรือเหยื่อผู้เคราะห์ร้าย เราจะใช้แสงเลเซอร์ยังเป็นเส้นขนานกับตัวหุ่นยนต์โดยจำกัดระยะห่างระหว่างสองเส้นให้มีค่าเท่ากับ 50 เซนติเมตร และมีระยะห่างออกจากด้านหน้าของตัวหุ่นยนต์ไป 50 เซนติเมตร การบอกตำแหน่งเราจะบอกโดยจะอ้างอิงกับแผนที่ของสถานที่นั้นๆ

1.3 ประโยชน์ที่จะได้รับจากโครงการ

ประโยชน์จากโครงการนี้ คือได้นำทฤษฎีความรู้ที่ได้จากการศึกษามาประยุกต์ใช้งานจริงๆ เช่น การออกแบบวงจรต่างๆ การวิเคราะห์วงจร เป็นต้น หรือจะเป็นการเขียนโปรแกรมควบคุม การทำงานของอุปกรณ์ต่างๆ ด้วยไมโครคอนโทรลเลอร์ MCS-51 การเขียนโปรแกรมด้วย Visual Basic ซึ่งเป็นโปรแกรมที่เราต้องศึกษาด้วยตัวเอง นอกจากนี้ยังได้ความรู้และประสบการณ์ใหม่ๆ ซึ่งจะทำให้เรามีความสามารถและทักษะเพิ่มขึ้น และยังสามารถนำไปใช้ในการทำงานในอนาคตได้อีกด้วย

1.4 รายละเอียดโดยย่อของโครงการ

หุ่นยนต์กู้ภัย จะมีโครงสร้างที่สร้างมาจากอะลูมิเนียมตันหนา 6 mm. หุ่นยนต์จะมีล้อทั้งหมด 6 ล้อ ด้านละ 3 ล้อ เพื่อลดช่วงความห่างระหว่างล้อทำให้สามารถข้ามผ่านพื้นผิวที่เป็นขั้นได้ โดยมีขนาด กว้างXยาวXสูง = 44X47X45 cm. โดยมีน้ำหนักประมาณ 25 กิโลกรัม หุ่นยนต์จะขับเคลื่อนโดยใช้ดีซีมอเตอร์ 2 ตัว ส่งกำลังโดยใช้โซ่และเฟือง ส่วนของฐานกล้องก็ทำมาจากอะลูมิเนียมเช่นเดียวกัน โดยใช้ดีซีมอเตอร์ 2 ตัว เป็นตัวปรับมุมมองของกล้อง ส่วนของกล้องเป็นกล้อง CCD ที่เป็นภาพสีโดย จะต่อเข้ากับการ์ดแคปเจอร์ (Capture Card) ภายในคอมพิวเตอร์และการส่งสัญญาณจะส่งโดยใช้ Video Sender มาแสดงภาพที่คอมพิวเตอร์ ในการควบคุมการเคลื่อนที่ของหุ่นยนต์จะควบคุมโดยส่ง ข้อมูลผ่านพอร์ทอนุกรม และส่งไปยังหุ่นยนต์แบบไร้สายโดยจะผ่าน โมดูลเรดิโอเมทริกซ์ (Radio Matrix) ซึ่งใช้ความถี่ในการส่งคือ 433 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมีการส่งแบบฮาร์ฟดูเพล็กซ์ (Half Duplex) ส่วนของการตรวจ สอบคุณภาพจะใช้เซนเซอร์คุณภาพที่เป็นแบบไม่สัมผัส (Non contact) ส่วนของเสียงจะใช้ ไมค์คอนเดนเซอร์ เป็นตัวตรวจ สอบเสียง และส่วนของการหาตำแหน่งของวัตถุจะใช้ อัลตราโซนิก (Ultrasonic) เป็นตัวค้นหาตำแหน่ง ซึ่งจะใช้หลักการการสะท้อนของคลื่นอัลตราโซนิก โดยคลื่นจะสะท้อนกับผนังหรือวัตถุต่างๆ แล้วก็จะสะท้อนกลับไปที่หุ่นยนต์ ส่วนของอุปกรณ์ต่างๆ ก็จะมีการติดตั้งไฟส่องทางที่ด้านหน้าของหุ่นยนต์ เพื่อใช้ในพื้นที่ยังไม่มีแสงหรือแสงสว่างไม่เพียงพอ

ส่วนของโปรแกรมการทำงานจะใช้ Visual Basic ความคุมส่วนที่เป็นซอฟต์แวร์ (Software) โดยจะใช้จอยสติคควบคุมทิศทางเคลื่อนที่ของตัวหุ่นยนต์

1.5 รายละเอียดของปริญญานิพนธ์

โครงการ “หุ่นยนต์กู้ภัย” เป็นโครงการที่รวมเอาความรู้หลายๆ แขนงมาใช้สร้างโครงการนี้ จึงทำให้รายละเอียดของเนื้อหา และทฤษฎีที่เกี่ยวข้องของโครงการมีมากพอสมควร โดยจะมีการแบ่งเนื้อหาออกเป็นบทต่างๆ ดังต่อไปนี้

บทที่ 1 บทนำ จะกล่าวถึงรายละเอียดความเป็นมา, ขอบเขต และรายละเอียดโดยย่อของปริญญานิพนธ์

บทที่ 2 ทฤษฎีและหลักการ โดยจะกล่าวถึงทฤษฎีและหลักการพื้นฐานต่างๆ ที่นำมาใช้ในการออกแบบ และแสดงให้เห็นถึงความสัมพันธ์ในการทำงานของอุปกรณ์ในแต่ละส่วน

บทที่ 3 การออกแบบส่วนต่างๆ ของหุ่นยนต์ โดยจะกล่าวถึงขั้นตอนในการออกแบบส่วนต่างๆ ของวงจรควบคุมและโครงสร้างตัวถังของหุ่นยนต์ และการออกแบบโปรแกรมควบคุมการทำงานส่วนต่างๆ ของหุ่นยนต์

บทที่ 4 ผลการทดสอบ โดยจะกล่าวถึงผลจากการทดสอบวงจรและโปรแกรมต่างๆ ว่ามีการทำงานเป็นอย่างไรบ้าง

บทที่ 5 สรุปและวิจารณ์ กล่าวถึงปัญหาที่พบในการทำงานเพื่อนำไปพัฒนาและปรับปรุงต่อไปในอนาคต

บทที่ 2 ทฤษฎีและหลักการ

2.1 ไมโครคอนโทรลเลอร์ MCS-51

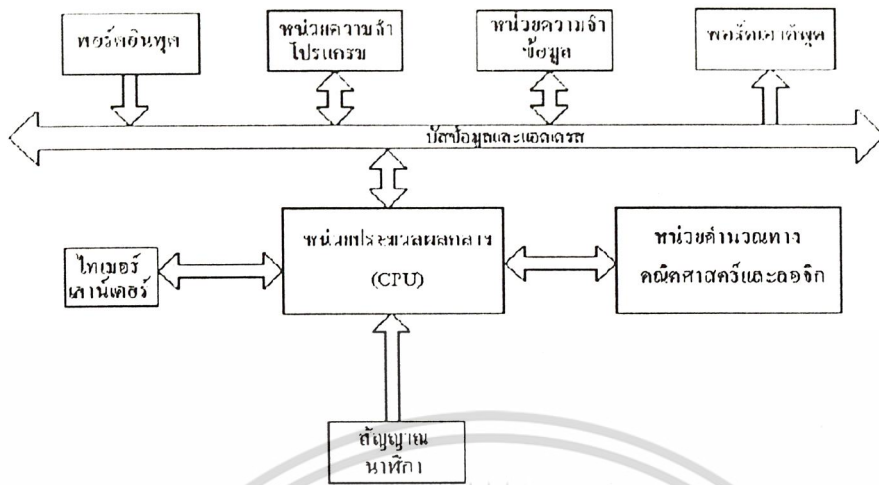
ไมโครคอนโทรลเลอร์ MCS-51 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิตที่มี อุปกรณ์สนับสนุน ประกอบอยู่ในหลายอย่าง ได้แก่ หน่วยความจำสำหรับเก็บข้อมูล หน่วยความจำสำหรับเก็บ โปรแกรม ตัวตั้งเวลา/ตัวนับ อุปกรณ์รับส่งข้อมูลแบบอนุกรม เนื่องจากโครงสร้างของไมโครคอนโทรลเลอร์มีอุปกรณ์สนับสนุนประกอบอยู่ในตัวเอง ทำให้การใช้งานง่ายขึ้นและมีประสิทธิภาพมากขึ้น โดยไม่ต้องมีการเชื่อมต่ออุปกรณ์ภายนอกเพิ่มเติมมากเหมือนกับ ตัวไมโครโปรเซสเซอร์ทั่วไป นอกจากนี้หากเราต้องการใช้งานไมโครคอนโทรลเลอร์ร่วมกับ อุปกรณ์อื่นเพิ่มเติม เช่น ไอซี 8255 หรือหน่วยความจำภายนอก เรายังสามารถนำมาเชื่อมต่อเพิ่มเติมเข้ากับไมโครคอนโทรลเลอร์ได้อีกด้วย

2.1.1 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

โครงสร้างภายในพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แสดงในรูปที่ 2.1 ประกอบด้วย อุปกรณ์ต่างๆ ดังนี้

- หน่วยประมวลผลกลางขนาด 8 บิต
- หน่วยประมวลผลสำหรับข้อมูลแบบบิต (BOOLEAN PROCESSOR)
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำโปรแกรม 64 กิโลไบต์
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำข้อมูล 64 กิโลไบต์
- หน่วยความจำโปรแกรมภายในขนาด 4 กิโลไบต์ แบบ อีพรอม (เบอร์ 8451)
- หน่วยความจำแบบ แรม ภายในจำนวน 128 ไบต์
- พอร์ตอินพุต/เอาต์พุตแบบขนานจำนวน 32 เส้น ซึ่งสามารถแยกทำงานได้อย่างอิสระ
- วงจรรนับ/จับเวลาขนาด 16 บิต จำนวนสองวงจร
- วงจรสื่อสารแบบอนุกรมแบบคูเพิ้ลเต็ม(FULL DUPLEX)
- วงจรรควบคุมการอินเตอร์รัปต์จากแหล่งกำเนิดสัญญาณ 6 ประเภท พร้อมการกำหนดลำดับวงจรมลิตสัญญาณนาฬิกาภายในซึ่ง โครงสร้างการทำงานทั้งหมดของไมโครคอนโทรลเลอร์จะอาศัยหลักการทำงานที่เกี่ยวข้องกัน โดยอาศัยหลักการการทำงานที่เป็นไปตามโครงสร้างเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

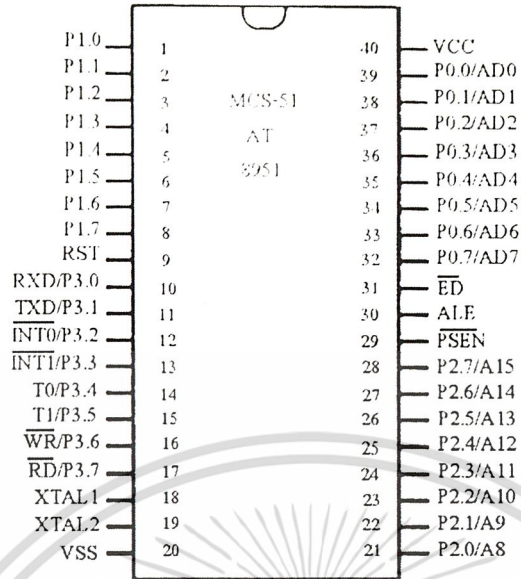


รูปที่ 2.1 แสดงโครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

โดยมากแล้วไมโครคอนโทรลเลอร์ตระกูลนี้มักจะมีรูปร่างของไอซีเป็นแบบขนาด 40 ขา ดังแสดงในรูปที่ 2.2 ซึ่งแต่ละขาสัญญาณจะมีหน้าที่ที่ระบุชัดเจนตามสัญลักษณ์ชื่อย่อที่กำกับในแต่ละขา อย่างไรก็ตามจะมีบางขาสัญญาณที่อาจจะมีหน้าที่ได้มากกว่าหนึ่งอย่าง (ซึ่งเขียนกำกับไว้ว่า ALTERNATE FUNCTION ในรูปที่ 2.2) ซึ่งจะไม่สามารถใช้งานในเวลาเดียวกันได้ ตัวอย่างเช่น ขาสัญญาณบิต 0 ของพอร์ต 3 (ใช้ตัวย่อเป็น P3.0) อาจจะใช้เป็นขาสัญญาณเอาต์พุต หรืออินพุตตามปกติ ภายในไมโครคอนโทรลเลอร์ MCS-51 ซึ่งประกอบด้วยหน่วยการทำงานต่างๆ ภายในไอซี MCS-51 จำนวนมาก โดยแต่ละบล็อกซึ่งเป็นวงจรควบคุมรีจิสเตอร์ (REGISTER) หรือหน่วยความจำภายในของไอซี MCS-51 จะถูกเชื่อมต่อเข้าด้วยกันผ่านทางเส้นสัญญาณที่เรียกว่าบัสข้อมูลภายใน รีจิสเตอร์และหน่วยความจำเหล่านี้จะถูกนำไปใช้ระหว่างการประมวลผลคำสั่ง หน้าที่ของโปรแกรมที่ผู้ใช้สร้างขึ้นมาก็เป็นการควบคุมการรับหรือส่งข้อมูลระหว่างรีจิสเตอร์เหล่านี้ ซึ่งอาจจะมีการดำเนินการร่วมกับหน่วยการดำเนินงานประมวลผลทางคณิตศาสตร์และลอจิก หรือเรียกว่า ARITHMETIC AND LOGIC UNIT :ALU

2.1.2 โครงสร้างหน่วยความจำภายในไมโครคอนโทรลเลอร์ MCS-51

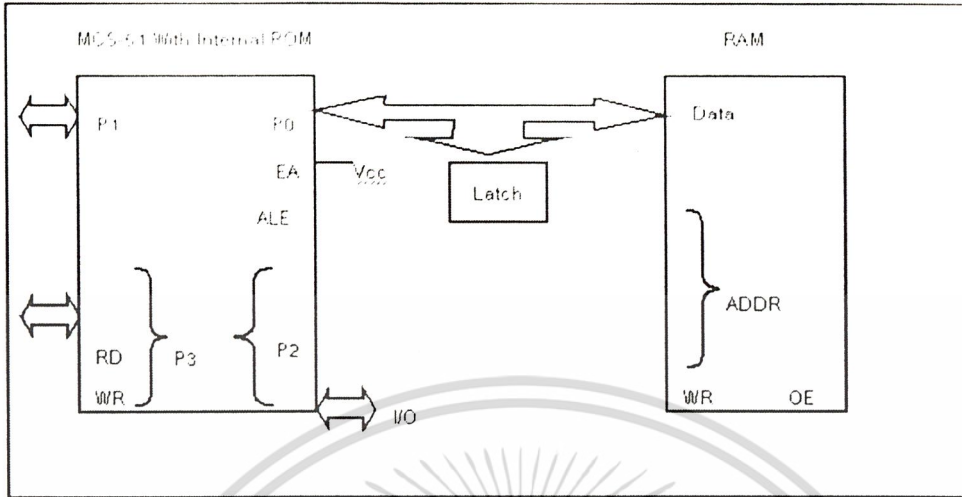
ไมโครคอนโทรลเลอร์ MCS-51 แยกการจัดการหน่วยความจำออกเป็นสองส่วนอย่างชัดเจน คือ หน่วยความจำโปรแกรม (PROGRAM MEMORY) และหน่วยความจำข้อมูล (DATA MEMORY) หน่วยความจำทั้งสองนี้ มีหน้าที่แตกต่างกัน และใช้วิธีการอ้างแอดเดรสสัญญาณการติดต่อแยกออกจากกัน



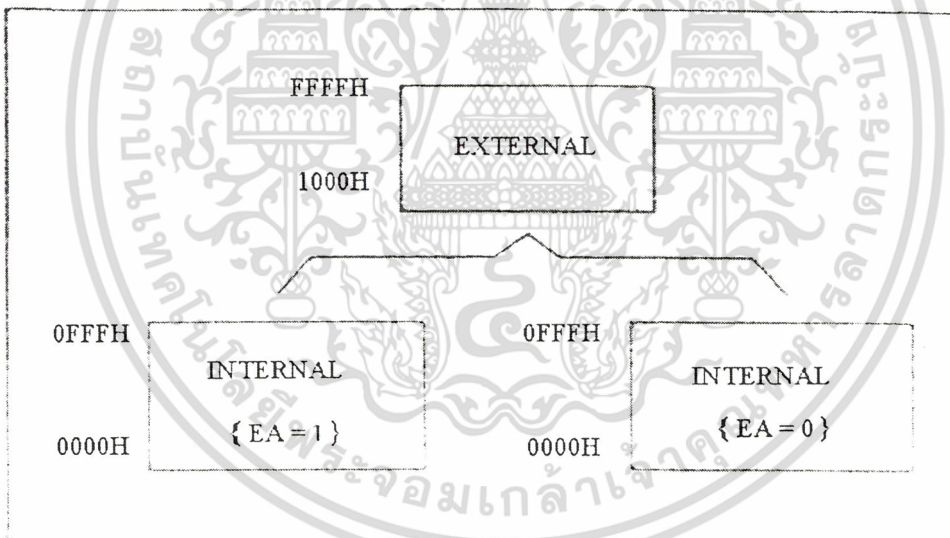
รูปที่ 2.2 แสดงรูปร่างและการจัดวางขาต่างๆ ของไมโครคอนโทรลเลอร์ MCS-51

2.1.3 หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 เป็นบริเวณหน่วยความจำสำหรับเก็บข้อมูลและคำสั่งใช้งานต่างๆ ซึ่งแม้ว่าจะไม่มีกระแสไฟฟ้าให้กับระบบข้อมูลเหล่านี้ก็ยังคงอยู่ไม่สูญหายโครงสร้างของหน่วยความจำโปรแกรม มีลักษณะเช่นเดียวกับหน่วยความจำที่บรรจุอยู่ในไมโครคอนโทรลเลอร์ MCS-51 ของหน่วยความจำ ประเภทต่างๆ เช่น หน่วยความจำแบบรอม (READ ONLY MEMORY) หรืออีพรอม (ERASABLE PROGRAMABLE READ ONLY MEMORY) ในไมโครคอนโทรลเลอร์ MCS-51 สามารถอ่านข้อมูลหน่วยความจำโปรแกรมนี้ได้สูงสุดไม่เกิน 64 กิโลไบต์ และแยกประเภทของหน่วยความจำโปรแกรมเป็น 2 ลักษณะ ตามตำแหน่งของหน่วยความจำ นั่นคือ หน่วยความจำโปรแกรมภายใน (INTERNAL PROGRAM MEMORY) ซึ่งเป็นหน่วยความจำรอมหรืออีพรอม ที่อยู่ภายในตัวไอซีของไมโครคอนโทรลเลอร์เอง และหน่วยความจำโปรแกรมภายนอก (EXTERNAL PROGRAM MEMORY) ซึ่งเป็นการใช้ไอซีหน่วยความจำมาทำหน้าที่เป็นหน่วยความจำโปรแกรมของระบบ



รูปที่ 2.3 แสดงการใช้หน่วยความจำสำหรับเก็บโปรแกรม



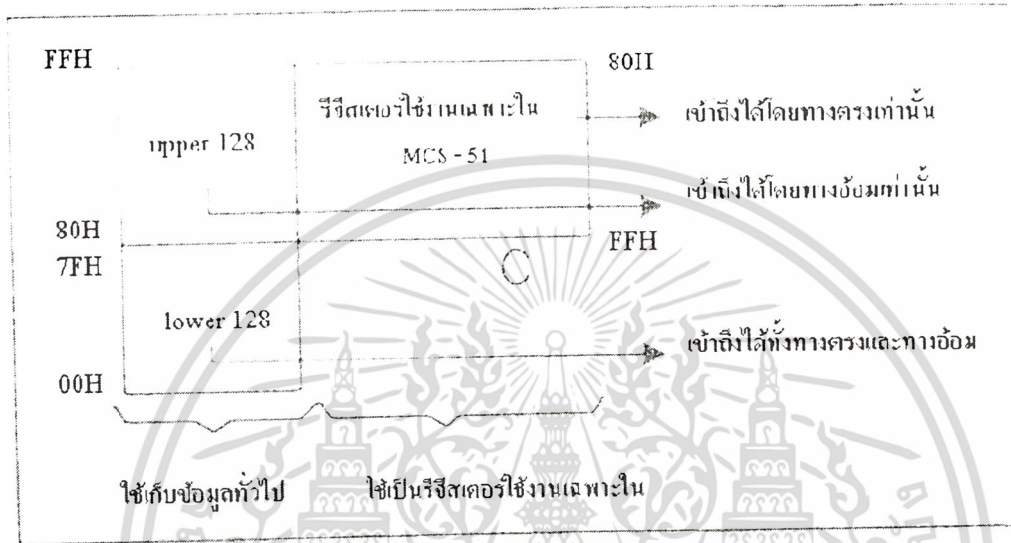
รูปที่ 2.4 แสดงการจัดพื้นที่ของหน่วยความจำโปรแกรมภายในและภายนอก

ไมโครคอนโทรลเลอร์เบอร์ต่างๆ ของตระกูล 8051 นี้สามารถขยายให้ใช้งานในหน่วยความจำภายนอกได้ทั้งสิ้น โดยกรณีที่มีหน่วยความจำโปรแกรมภายในอยู่แล้วการอ้างตำแหน่งแอดเดรสที่มีทั้งในหน่วยความจำโปรแกรมภายใน และภายนอกนั้นจะต้องทำการควบคุมระดับลอจิกของสัญญาณในขณะนั้นด้วยขนาดหน่วยความจำโปรแกรมภายในของไมโครคอนโทรลเลอร์เบอร์ต่างๆ ภายในตระกูล 8051 จะแตกต่างกันออกไป เพื่อความเหมาะสมกับการนำไปใช้งานลักษณะต่างๆ -8051 และ 8052 มีหน่วยความจำแบบรวม 4 และ 8 กิโลไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-8751 มีหน่วยความจำแบบ อีพรอม ขนาด 4 กิโลไบต์ข้อมูลที่จัดเก็บภายในนี้ ซึ่งสามารถใช้แสวงหาข้อมูลได้อัตโนมัติ และนำกลับไปประมวลโปรแกรมใหม่ได้อีกครั้งหนึ่ง

-8031 และ 8032 ไม่มีหน่วยความจำโปรแกรมอยู่ภายในตัวไอซี ดังนั้นในการนำไปใช้งานจึงจำเป็นต้องอาศัยหน่วยความจำโปรแกรมภายนอกเสมอ

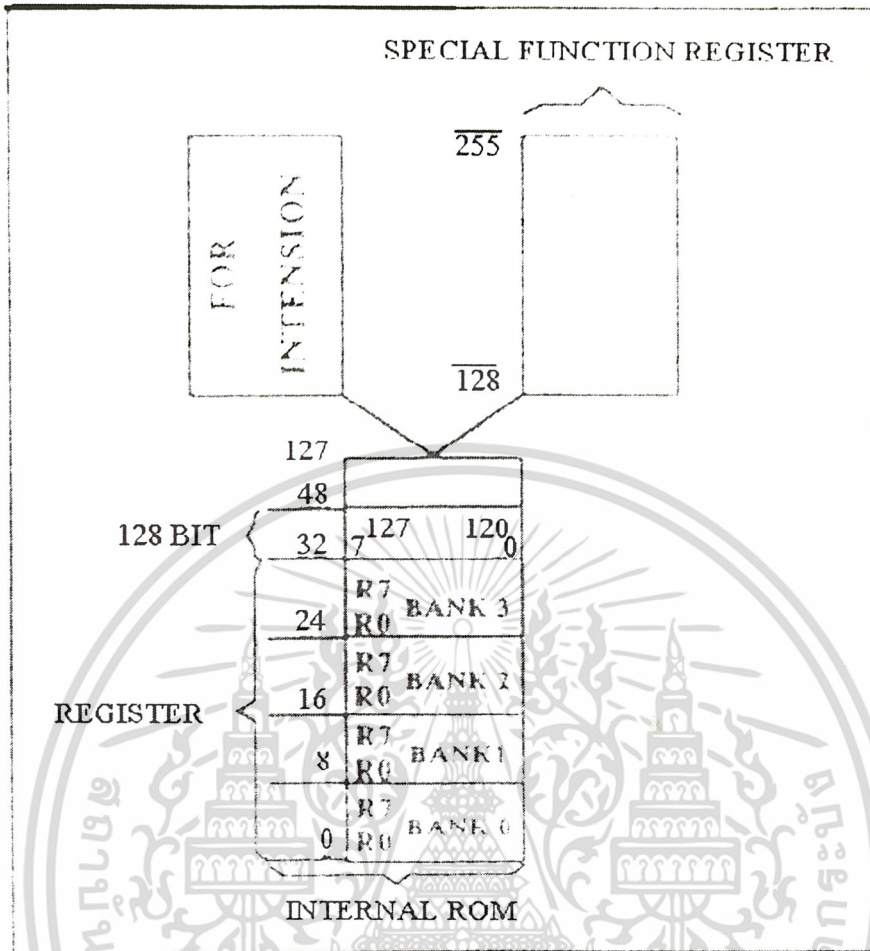


รูปที่ 2.5 แสดงหน่วยความจำสำหรับเก็บข้อมูลภายในไมโครคอนโทรลเลอร์ MCS-51

2.1.4 หน่วยความจำข้อมูล

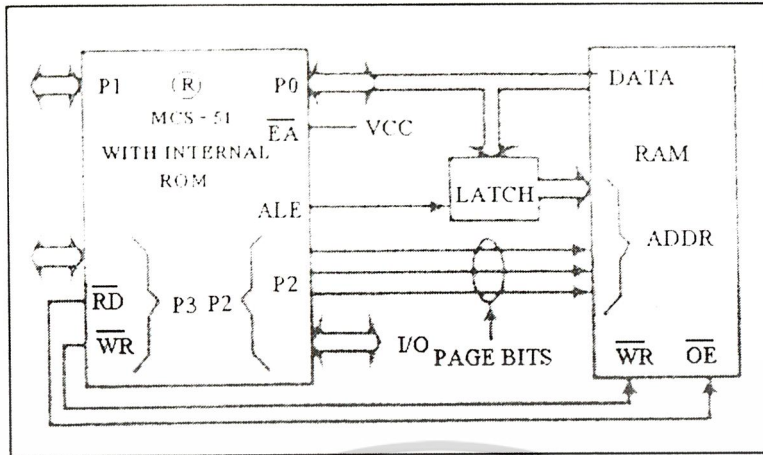
หน่วยความจำข้อมูล (DATA MEMORY) ซึ่งโดยพื้นฐานแล้วเป็นหน่วยความจำที่สามารถเขียนหรืออ่านข้อมูลได้ (READ OR WRITE MEMORY) ใช้สำหรับเก็บข้อมูลหรือตัวแปรที่เกิดขึ้นในขณะที่กำลังประมวลผลโปรแกรมไว้เป็นการชั่วคราว ซึ่งโดยพื้นฐานแล้วหน่วยความจำข้อมูลจัดเป็นหน่วยความจำแบบสแตติก ดังนั้นเมื่อไม่มีการจ่ายไฟฟ้าให้กับระบบก็จะมีผลทำให้ข้อมูลที่จัดเก็บไว้ภายในหน่วยความจำนี้สูญหายไป พื้นที่ของหน่วยความจำข้อมูลของไมโครคอนโทรลเลอร์ MCS-51 มีได้สูงสุดไม่เกิน 64 กิโลไบต์ และแยกประเภทออกเป็นสองลักษณะตามตำแหน่งที่ตั้งของหน่วยความจำนั้น ตามลักษณะของหน่วยความจำโปรแกรมภายใน ซึ่งก็เป็นแรมที่อยู่ภายในตัวไอซีในตระกูลของไมโครคอนโทรลเลอร์ และหน่วยความจำข้อมูลภายนอกซึ่งเป็นการใช้ไอซีหน่วยความจำแรมมาเพิ่มเติมเข้าไปในวงจรลักษณะเดียวกับ การนำไอซีอีพรอมมาใช้งานเป็นหน่วยความจำโปรแกรมนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงการจัดหน่วยความจำข้อมูล

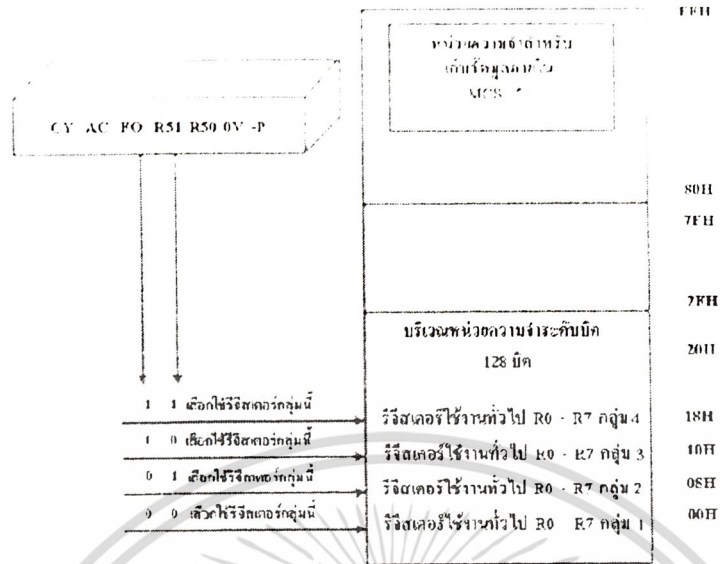
โดยที่หน่วยความจำสำหรับเก็บข้อมูลของไมโครคอนโทรลเลอร์ MCS-51 นี้สามารถแบ่งออกเป็น 2 ส่วน คือ ในส่วนที่เป็นหน่วยความจำสำหรับเก็บข้อมูลภายในไอซี และหน่วยความจำสำหรับเก็บข้อมูลภายนอกไมโครคอนโทรลเลอร์ MCS-51 ทุก ๆ เบอร์จะมีหน่วยความจำเก็บข้อมูลทั่ว ๆ ไปภายในไอซี อย่างน้อยคือ 128 ไบต์ ไปจนถึง 256 ไบต์ ทั้งนี้ขึ้นกับเบอร์ของไอซี หน่วยความจำสำหรับเก็บข้อมูลภายในไอซีในบริเวณ 128 ไบต์เรียกว่า LOWER 128 และในบริเวณ 128 ไบต์หลังที่มีเพิ่มในบางเบอร์มีชื่อเรียกว่า UPPER



รูปที่ 2.7 แสดงการต่อกับหน่วยความจำข้อมูลภายนอกไอซี

2.1.5 รีจิสเตอร์ที่เกี่ยวข้องกับไมโครคอนโทรลเลอร์ MCS-51

รีจิสเตอร์ในกลุ่มนี้ จะเป็นรีจิสเตอร์ขนาด 16 บิตที่ใช้งาน เพื่อเก็บข้อมูลของตัวแอดเดรสเป็นสำคัญ โดยค่าที่อยู่ภายในแอดเดรสนี้จะนำไปเป็นค่าของข้อมูลที่ส่งออกไปทางบัสแอดเดรส ในส่วนของไมโครคอนโทรลเลอร์ เพื่อบอกตำแหน่งที่ต้องการติดต่อ รีจิสเตอร์ที่จัดในกลุ่มนี้ประกอบด้วยรีจิสเตอร์ใช้งานทั่วไป (GENERAL-PURPOSE REGISTERS) รีจิสเตอร์ในกลุ่มนี้จัดเป็นพื้นที่หน่วยความจำที่ใช้ในการสนับสนุนในการประมวลผล การทำงานจากหน่วยประมวลผลทางคณิตศาสตร์และลอจิก (ALU) เพื่อให้สามารถจัดการข้อมูลให้เร็วที่สุด นอกจากนี้โปรแกรมที่ไม่ได้ใช้คำสั่งเหล่านี้ก็ยังใช้เป็นการเก็บข้อมูลตัวแปรภายในโปรแกรม จะเห็นได้ว่าชื่อของรีจิสเตอร์ไม่ว่าจะอยู่ในรีจิสเตอร์แบงก์ใด ก็จะมีชื่อว่า R0 ถึง R7 เหมือนกันทั้งสิ้น ดังนั้นในการใช้งานผู้ใช้จะต้องให้ความระมัดระวังว่า ต้องการรีจิสเตอร์นั้นๆ จากแบงก์ใดๆ ซึ่งการกำหนดเลือกแต่ละกลุ่มของรีจิสเตอร์นี้ก็ทำได้ง่าย เพียงการกำหนดค่าของบิตที่อยู่ภายในแฟล็ก (PSW) เท่านั้นอย่างไรก็ตาม โดยทั่วไปก็มักจะมีการใช้งานรีจิสเตอร์ R0 ถึง R7 เฉพาะในแบงก์ 0 เท่านั้น ดังนั้นพื้นที่ของแบงก์อื่นๆ ที่เหลือก็สามารถนำมาใช้ในลักษณะของหน่วยความจำแรม



รูปที่ 2.8 แสดงการเลือกใช้รีจิสเตอร์ใช้งานทั่วไป R0-R7 ในแต่ละกลุ่ม

- รีจิสเตอร์หน้าที่พิเศษ เป็นรีจิสเตอร์หน้าที่พิเศษ (SFR) เป็นรีจิสเตอร์สำหรับการควบคุมหน้าที่ และการทำงานของอุปกรณ์หรือพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 ทั้งหมด ตำแหน่งของรีจิสเตอร์เหล่านี้จะจัดอยู่ในบริเวณแอดเดรส 80H-FFH การใช้รีจิสเตอร์หน้าที่พิเศษเหล่านี้สามารถทำได้ทั้งการระบุชื่อของรีจิสเตอร์ หรือตำแหน่งแอดเดรส ที่เป็นของรีจิสเตอร์นั้นก็ได้รับการจัดพื้นที่หน่วยความจำสำหรับรีจิสเตอร์หน้าที่พิเศษเหล่านี้ โดยมีข้อสังเกตว่ารีจิสเตอร์ที่อยู่ในตำแหน่งแอดเดรสที่มีจำนวนเป็นทวีคูณของค่า 8 จะสามารถอ้างถึงในระดับบิตได้ด้วย (นั่นคือแอดเดรส 80H 88H 90H A0H A8H B0H B8H D0H E0H และ F0H)

- แอควิวมูลเตอร์ (ACCUMULATOR) หรือ ACC เป็นรีจิสเตอร์ขนาด 8 บิต ทำหน้าที่ในการเก็บข้อมูลที่จะส่งให้กับหน่วยทำงานภายในหน่วยประมวลผลกลาง และเก็บผลลัพธ์ที่ได้จากการทำงานเท่านั้น การทำงานของรีจิสเตอร์นี้มีลักษณะเช่นเดียวกับตัวแอควิวมูลเตอร์ของโปรเซสเซอร์ทั่วไป การใช้งานในโปรแกรมซึ่งใช้เรียกเป็น รีจิสเตอร์ A

2.1.6 ชุดคำสั่งของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 ประกอบด้วยคำสั่งทั้งหมดจำนวนมาก ซึ่งสามารถจะจัดกลุ่มคำสั่งเหล่านี้ตามลักษณะและหน้าที่การทำงานที่คล้ายคลึงกัน เพื่อความสะดวกต่อการศึกษาทำความเข้าใจและใช้งาน ดังนี้

2.1.6.1 กลุ่มการถ่ายเทข้อมูล คือ กลุ่มคำสั่งในการโอนย้ายข้อมูล ทำหน้าที่ในโอนย้ายข้อมูลระหว่างรีจิสเตอร์ หรือหน่วยความจำภายในแรม โดยมีรายละเอียดดังนี้ ชุดคำสั่งในการถ่ายเทแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในนั้น ซึ่งเวลาที่ใช้ในหนึ่งคำสั่งนั้นจะเป็นเวลา เมื่อขณะที่ความถี่ในการทำงานของหน่วยประมวลผลกลางที่ความถี่ 12 เมกะเฮิร์ตซ์ และรายละเอียดของแต่ละคำสั่งมีดังนี้ MOV จะทำงานในลักษณะเป็นการถ่ายเทข้อมูลที่มีขนาดเป็นไบต์ หรือ บิตก็ได้ จากแหล่งกำเนิดเข้าสู่ตัวรับข้อมูลในฟิลด์โอเปอร์เรนด์ PUSH จะทำงานโดยเพิ่มค่ารีจิสเตอร์ SP ก่อนแล้วจึงทำการถ่ายเทข้อมูล 1 ไบต์จากแหล่งกำเนิดไปบริเวณสแต็คตามตำแหน่งที่รีจิสเตอร์ SP กำหนด POP: การถ่ายเทข้อมูลขนาด 1 ไบต์จากบริเวณตำแหน่งที่รีจิสเตอร์ SP กำหนดไปยังรีจิสเตอร์ที่โอเปอร์เรนด์ กำหนดและหลังจากนั้นรีจิสเตอร์ SP จะลดค่าลง XCH: คำสั่งแลกเปลี่ยนไบต์ระหว่างแหล่งกำเนิดโอเปอร์เรนด์กับรีจิสเตอร์ AXCHD คำสั่งในการแลกเปลี่ยนขนาดนิบเบิลทางอันดับต่ำของแหล่งกำเนิดโอเปอร์เรนด์กับนิบเบิลอันดับต่ำของแอกคิวมูลเตอร์ ตัวอย่างเช่นทำการเลื่อนข้อมูลไป 2 ไบต์ทางขวามือ ซึ่งจะมี 2 วิธีคือ ใช้คำสั่ง MOV หรือใช้คำสั่ง XCH รายละเอียดการใช้คำสั่งทั้ง 2 แบบ

2.1.6.2 กลุ่มคำสั่งทางคณิตศาสตร์ เช่น การบวก ลบ คูณ และหารข้อมูลภายในตัวรีจิสเตอร์ต่างๆ ช่วงเวลาการทำงานของแต่ละคำสั่งนั้นจะกำหนดที่ความถี่ของสัญญาณนาฬิกาที่ 12 เมกะเฮิร์ตซ์ คำสั่งทางคณิตศาสตร์ส่วนใหญ่ใช้เวลา 1 ms ยกเว้นคำสั่ง INC DPTR ซึ่งใช้เวลา 2 ms โดยที่คำสั่งการคูณและหารใช้เวลา 4 ms โดยมีรายละเอียดการใช้คำสั่งมีดังนี้ INC: เป็นการบวกหนึ่งกับโอเปอร์เรนด์และใส่ค่าใหม่กลับเข้าไปที่ตัวโอเปอร์เรนด์นั้นๆ DEC: เป็นการลบออกจากตัวเลขที่อยู่ในแหล่งกำเนิดโอเปอร์เรนด์ และนำผลลัพธ์ที่ได้มาเก็บไว้ที่ตัวโอเปอร์เรนด์นั้น ADD: เป็นการบวกในแอกคิวมูลเตอร์เข้ากับค่าในแหล่งกำเนิดโอเปอร์เรนด์ ADDC: เป็นการบวกค่าต่างๆ ในแอกคิวมูลเตอร์เข้ากับค่าในแหล่งกำเนิดโอเปอร์เรนด์และบวกกับบิตทดด้วย SUBB: เป็นการนำเลขที่แหล่งกำเนิดโอเปอร์เรนด์ลบออกจากตัวเลขใน A และนำค่าบิตตัวทดมาลบออกอีกและผลลัพธ์ที่ได้นำมาใส่ลงในแอกคิวมูลเตอร์ A MUL: เป็นการคูณแบบไม่คิดตัวเครื่องหมายของตัวเลขที่อยู่ใน แอกคิวมูลเตอร์กับ เลขในรีจิสเตอร์ B แล้วได้ผลลัพธ์ 2 ไบต์ นำมาเก็บไว้ที่ AB โดย A จะรับอันดับต่ำส่วน B จะรับอันดับสูง DIV: เป็นคำสั่งในการหารแบบไม่คิดเครื่องหมายที่อยู่ในแอกคิวมูลเตอร์ แล้วหารตัวเลขในรีจิสเตอร์ B แล้วนำผลลัพธ์ไปเก็บในแอกคิวมูลเตอร์ และเศษของการหารตัวเลขจะเก็บไว้ในรีจิสเตอร์ B DA: สำหรับการบวกกันทางตัวเลข BCD เป็นการปรับค่ารวม ซึ่งเป็นผลมาจากการบวกกันทางไบนารีของระบบตัวเลข BCD ขนาด 2 หลักสองจำนวน การปรับค่าตัวเลขผลรวมด้วยการใช้คำสั่ง DA จะได้ผลลัพธ์กลับมาที่แอกคิวมูลเตอร์ รายละเอียดการใช้คำสั่ง

2.1.6.3 กลุ่มคำสั่งทางตรรกศาสตร์หรือแบบลอจิก ทำหน้าที่เกี่ยวกับการประมวลผลแบบลอจิกต่างๆ เช่น การ AND OR หรือ EX-OR ระหว่างข้อมูลในรีจิสเตอร์ A นั้นเอง โดยมีการใช้คำสั่งดังนี้ CPL: เป็นการใช้คำสั่งกลับค่าหรือคอมพลิเมนต์ ข้อมูลในแอกคิวมูลเตอร์จะไม่มีผลใดๆต่อค่าของแฟล็กหรือการอ้างถึงตำแหน่งแอดเดรสนั้นตามบิตนั้นๆ RL, RLC, RR, RRC, SWAP: ทั้ง 5 คำสั่งนี้เป็นคำสั่งในการทำงานการวนบิตบนตัวของแอกคิวมูลเตอร์ ซึ่ง RL เป็นการวนบิตทางขวา, RLC เป็นการทำการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วนทางซ้ายผ่านบิตทด , RRC เป็นการวนขวาผ่านบิตทด และ SWAP เป็นการวนซ้ายสี่ครั้ง ANL:เป็นการ ADD กันทางตรรกศาสตร์ ระหว่างแหล่งกำเนิดสองโอเปอร์แรนด์ ซึ่งจะสั่งให้ทำงานในรูปแบบของตรรกศาสตร์ทางข้อมูลขนาดเป็นไบต์หรือบิตกลุ่มคำสั่งแบบบูลีนหรือแบบบิตซึ่งเป็นความสามารถของไมโครคอนโทรลเลอร์ MCS-51 ที่จะดำเนินการประมวลผลแบบบิต แทนที่จะเป็นข้อมูล ทั้งไบต์ เช่นปกติ โดยมีชุดคำสั่งที่จัดการโดยตรง ทุกคำสั่งจะเข้าถึงข้อมูลโดยตรงในระดับบิต โดยมีการบิตแอดเดรสได้ตั้งแต่ 00H - 7FH ในพื้นที่ 128 บิต หน่วยความจำข้อมูลภายในและบิตแอดเดรส 80H-FFH ในบริเวณกลุ่มรีจิสเตอร์ฟังก์ชันพิเศษ (SFR)

2.1.6.4 กลุ่มคำสั่งในการกระโดดไปยังตำแหน่งต่างๆภายในโปรแกรม ซึ่งจะเปลี่ยนลำดับของการประมวลผลภายในโปรแกรมไปยังส่วนต่างๆแทนที่จะดำเนินการไปเป็นลำดับ ต่อเนื่องโดยที่คำสั่ง JMP จะแบ่งเป็น 3 ลักษณะ คือ SJMP, LJMP, AJMP ซึ่งในแต่ละคำสั่งจะมีข้อแตกต่างของการกระโดดไปยังแอดเดรสไกลสุดที่ต่างกันคำสั่ง JMP ซึ่งเป็นแบบโมนิซิกที่สามารถจะใช้ได้ โดยมีรายละเอียดการใช้งานของคำสั่งดังต่อไปนี้ SMP:จะเป็นการกระโดดแบบการย้ายอันดับตำแหน่งของแอดเดรสตำแหน่งเดิมซึ่งจะสามารถกระโดดได้ -128 ถึง +127 ไบต์ AJMP:ลักษณะแบบนี้จะสามารถกระโดดได้ไกลสุดประมาณ 2 กิโลไบต์ ซึ่งจะใช้หน่วยความจำเพียง 2 ไบต์เท่านั้น ในการกำหนด LJMP:ลักษณะแบบนี้จะสามารถกระโดดได้ไกลสุดประมาณ 64 กิโลไบต์ ซึ่งจะใช้หน่วยความจำเพียง 3 ไบต์เท่านั้นในการกำหนด JMP @A+DPTR:เป็นการควบคุมการกระโดดไปยังโปรแกรมที่ต้องการเฉพาะภายในส่วนต่างๆ

2.1.7 โครงสร้างการอินเทอร์รัปต์ของไมโครคอนโทรลเลอร์ MCS-51

จากแผนภาพโครงสร้างระบบอินเทอร์รัปต์ของไมโครคอนโทรลเลอร์ MCS-51 สัญญาณที่เข้ามาทำการอินเทอร์รัปต์ MCS-51 นั้นเกิดขึ้นได้ 5 ลักษณะตามตารางข้อมูล ในรูปที่ 2.6 โดยจะเห็นได้ว่าสามารถที่จะกำหนดเลือกเพื่อยินยอม (หรืออีนาเบิล : ENABLE) และห้าม (หรือดิสเอเบิล:DISABLE) ไม่ให้มีการอินเทอร์รัปต์แต่ละประเภทได้ โดยการกำหนดบิตของข้อมูลที่เกี่ยวข้อง ซึ่งมักจะอยู่ภายในรีจิสเตอร์ TCONและSCON นอกจากนี้ยังมีตำแหน่งบิตภายในรีจิสเตอร์ IE(INTERRUPT ENABLE REGISTER) ซึ่งทำหน้าที่เสมือนกับเป็นสวิตช์หลักที่เกี่ยวข้องกับสัญญาณอินเทอร์รัปต์ทั้งหมด หากว่ากำหนดไม่ให้เกิดการอินเทอร์รัปต์ แล้วการกำหนดบิตเพื่อห้ามหรือยินยอมของแต่ละอินเทอร์รัปต์ ก็จะไม่มีความสำคัญเกิดขึ้นยังแสดงให้เห็นว่าสัญญาณอินเทอร์รัปต์ แต่ละประเภทยังสามารถกำหนดระดับความสำคัญ (PRIORITY) ของการอินเทอร์รัปต์ได้สองลักษณะ คือระดับความสำคัญสูงหรือต่ำ (HIGH OR LOW PRIORITY) กล่าวคือ ขณะที่กำลังประมวลผลอยู่ภายในส่วนของโปรแกรมย่อยบริการอินเทอร์รัปต์ของสัญญาณที่มีระดับความสำคัญต่ำอยู่ ก็อาจจะถูกขัดจังหวะให้ไปประมวลผลของสัญญาณอินเทอร์รัปต์ที่มีระดับความสำคัญสูงกว่า แต่หากว่าเป็นสัญญาณอินเทอร์รัปต์ที่มีระดับความสำคัญต่ำเช่นเดียวกันแล้ว ก็ต้องรอให้เสร็จสิ้นการประมวลผลที่ ดำเนินการอยู่ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.8 การรีเซต โดยความหมายของการรีเซตเป็นการบังคับให้มีการเริ่มต้นใหม่อีกครั้งหนึ่ง ซึ่งมักจะกระทำโดยการกำหนดสถานะของสัญญาณที่รีเซตของไอซี MCS-51 ให้เป็นระดับลอจิกที่เหมาะสมเท่านั้น การรีเซตด้วยวิธีนี้ถือว่าการอินเทอร์รัปต์อย่างหนึ่งได้ แต่จะมีลักษณะต่างออกไปจากการอินเทอร์รัปต์ของสัญญาณนี้ได้ ซึ่งมีศัพท์เฉพาะเรียกว่า NON-MASKABLE INTERRUPT นอกจากนี้การดำเนินการของโปรแกรมก็แตกต่างออกไปด้วย โดยจะไม่มีกรเก็บค่าของคำสั่งที่กำลังจะไปทำในลำดับต่อไปภายในรีจิสเตอร์ PC เมื่อมีการรีเซตเกิดขึ้น โปรแกรมจะถูกส่งให้กระโดดไปยังแอดเดรส 0000 ทันทึ ซึ่งตำแหน่งนี้จะเป็นตำแหน่งเริ่มต้นของการทำงานของไมโครคอนโทรลเลอร์ MCS-51 เมื่อเริ่มจ่ายไฟให้กับระบบเมื่อใดก็ตามที่มีการรีเซตเกิดขึ้นค่าสถานะต่างๆ ภายในไมโครคอนโทรลเลอร์จะถูกกำหนดกลับไปเป็นค่าเริ่มต้นใหม่อีกครั้ง

2.2 การใช้งานไมโครคอนโทรลเลอร์กับพอร์ตอนุกรม

การติดต่อกับพอร์ตอนุกรมนั้นจะยากกว่าการติดต่อกับกับพอร์ตขนาน การสื่อสารของอุปกรณ์ที่ต่อกับพอร์ตอนุกรมจะถูกเปลี่ยน (Convert) เป็นสัญญาณแบบขนาน แล้วจึงนำไปประมวลผลต่อ ซึ่งจะใช้ Universal Asynchronous Receive / Transmitter (UART) เป็นตัวทำหน้าที่ส่วนทางด้านโปรแกรมที่มีรีจิสเตอร์ที่ต้องจัดการมากกว่า Standard Parallel Port (SPP) อีกหลายตัว

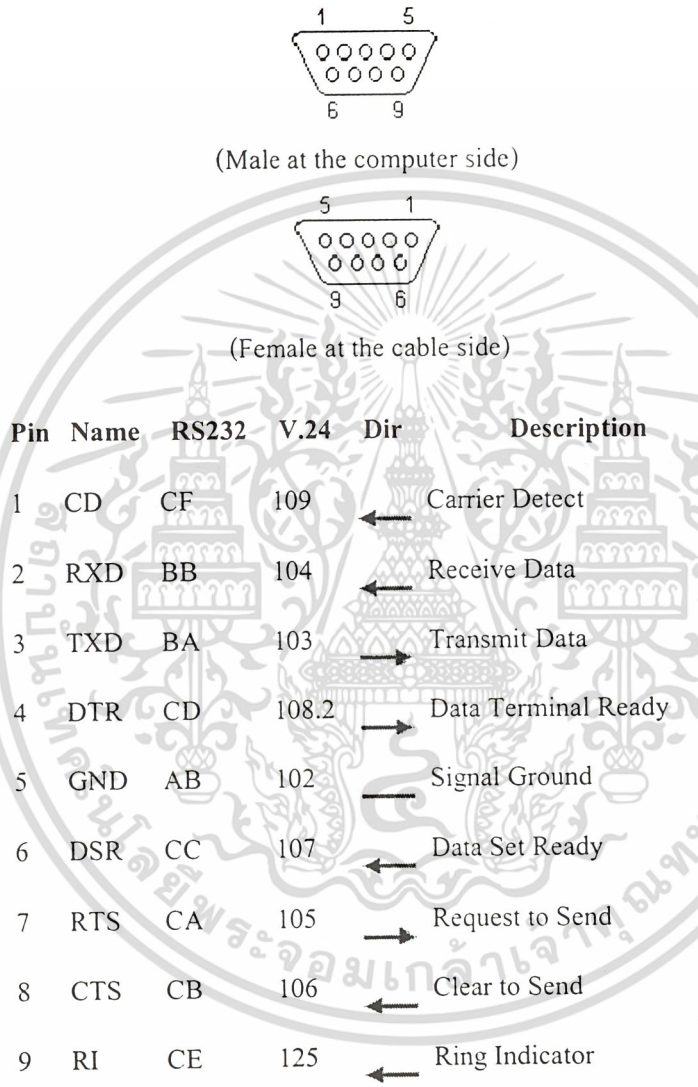
ข้อดีของพอร์ตอนุกรม

1. ระยะของสายอนุกรมสามารถมีความยาวได้มากกว่าสายของขนานมาก ทั้งนี้เพราะสัญญาณของพอร์ตอนุกรม ซึ่งส่วนใหญ่ใช้มาตรฐาน RS-232C จะมีค่า -3 Volt ถึง -15 Volt สำหรับ Logic "1" หรือ "Mark" และมีค่า +3 Volt ถึง +15 Volt สำหรับ Logic "0" หรือ "Space" (สำหรับช่วง +3 Volt ถึงถึง -3 Volt เป็นช่วง Undefined) ส่วนสัญญาณของสายขนานนั้น Logic "1" จะมีค่า +5 Volt และ logic "0" จะมีค่า 0 Volt ทำให้สัญญาณของสายอนุกรมสามารถรับการสูญเสียของสาย (Cable loss) ได้มากกว่าสัญญาณของสายขนานปรกติสายขนานจะไปได้เพียง 5 ฟุต ส่วนสาย RS-232 จะไปได้ถึง 50 ฟุตที่ความเร็วสูงสุดของมัน
2. สายอนุกรมจะใช้จำนวนสายไฟน้อยกว่าสายขนานถ้าต่อในลักษณะ NullModemจะใช้สายเพียง 3 เส้น ขณะที่แบบขนาน จะต้องใช้สาย 19 ถึง 25 เส้น
3. การสื่อสารแบบไร้สายเช่นการใช้ Infra Red การส่งพร้อมกันทีละ 8 บิต แบบขนาน จะทำให้ไม่สามารถแยกแยะได้ว่า Bit ไหนเป็น Bit0 หรือ Bit1 ... เป็นต้น ปัจจุบันอุปกรณ์ IrDA มีความเร็วไม่ต่ำกว่า 115.2K Baud แต่มี Pulse length เพียง 3/16 ของ RS-232 เพื่อประหยัดพลังงาน เพราะส่วนมากใช้ในอุปกรณ์แบบพกพาเช่น Laptop หรือ Palmtop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ปัจจุบันไมโครคอนโทรลเลอร์ มักมีการผนวกพอร์ตการสื่อสารแบบอนุกรมไว้ด้วย เพราะใช้จำนวนขาน้อยกว่าแบบขนาน

RS-232C กำหนด Baud rate ไว้ไม่เกิน 20K Baud ปัจจุบันได้แก้ไขให้รองรับกับเทคโนโลยีใหม่ได้ จึงมีการปรับปรุงถึง RS-232E ซึ่งมีรายละเอียดอีกหลายอย่าง



รูปที่ 2.9 ตารางแสดงถึงหน้าที่ของขาต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Abbreviation	Full Name	Originator	Function
TD	Transmit Data	DTE	Serial data output (TXD) from DTE.
RD	Receive Data	DCE	Serial data input (RXD) to DTE.
CTS	Clear To Send	DCE	Tell DTE that DCE is ready to exchange data.
(D)CD	(Data) Carrier Detect	DCE	Carrier from remote DCE is detected.
DSR	Data Set Ready	DCE	Tell DTE that DCE is ready to establish a link.
DTR	Data Terminal Ready	DTE	Tell DCE that DTE is ready to establish a link.
RTS	Ready To Send	DTE	Tell DCE that DTE is ready to exchange data.
RI	Ring Indicator	DCE	Ringling signal from the phone line is detected.

รูปที่ 2.10 แสดงสัญญาณต่างๆที่ส่งในรูปแบบอนุกรม

2.2.1 หน้าที่ของสัญญาณต่าง ๆ มีดังนี้

Protective ground เป็นจุดที่ต่อกับตัวเปลือกของอุปกรณ์ และไม่ต่อกับสัญญาณใด ๆ ในระบบ เพื่อใช้ต่อลงดิน เป็นการป้องกันอันตรายจากไฟลัดวงจร และใช้เป็น Shield ป้องกันการรบกวน

Signal ground หรือ Common return เป็นจุดสำคัญที่สุดที่ต้องมีในระบบ RS-232 เพราะเป็นจุดอ้างอิงของทุกสัญญาณ ยกเว้น Protective ground

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Transmit data เป็นสัญญาณข้อมูลที่ส่งจาก DTE ไปยัง DCE ในขณะที่ไม่ได้ส่งข้อมูลมันจะมีสถานะเป็น Logic "1" หรือ "Mark" หรือ "Off" หรือ -5 V ถึง -15 V ที่ด้านส่ง (DTE) หรือ -3 V ถึง -15 V ที่ด้านรับ (DCE) การส่งข้อมูลจะเกิดขึ้นได้ต้องมีสัญญาณควบคุมที่เกี่ยวข้อง "On" ก่อนคือสัญญาณ RTS, CTS, (D)CD, DTR และ DSR

Recieve data เป็นสัญญาณข้อมูลจาก DCE มายัง DTE ในขณะที่ไม่ได้ส่งข้อมูลมันจะมีสถานะเป็น Logic "1" หรือ "Mark" หรือ "Off" หรือ -5 V ถึง -15 V ที่ด้านส่งหรือ -3 V ถึง -15 V ที่ด้านรับ ในขณะที่ไม่ได้ส่งข้อมูลมันจะมีสถานะเป็น Logic "1" หรือ "Mark" หรือ "Off" หรือ -5 V ถึง -15 V ที่ด้านส่ง (DCE) หรือ -3 V ถึง -15 V ที่ด้านรับ (DTE) กรณีที่เป็นการสื่อสารแบบ Half-duplex สัญญาณ RD จะอยู่สถานะ "Off" ขณะที่สัญญาณ RTS อยู่ในสถานะ "On" และสัญญาณ RD จะยังคงอยู่ในสถานะ "Off" อีกชั่วระยะหนึ่งหลังจากที่สัญญาณ RTS เปลี่ยนจากสถานะ "On" มาเป็น "Off" แล้วเพื่อให้การรับ-ส่งข้อมูลเสร็จสมบูรณ์

Request to send เป็นสัญญาณจาก DTE ส่งไปให้ DCE เพื่อขอส่งข้อมูลไปปรกติจะอยู่ในสถานะ "Off" เมื่อต้องการส่งข้อมูลจะเปลี่ยนเป็นสถานะ "On" จนกว่าการส่งเสร็จสิ้นจึงเปลี่ยนกลับมาที่สถานะ "Off" ตามเดิม ทั้งนี้ทาง DTE ต้องได้รับสัญญาณ CTS จึงจะสามารถส่งข้อมูล TD ไปยัง DCE ได้ และสัญญาณ RTS ที่กลับสู่สถานะ "Off" จะไม่สามารถเปลี่ยนเป็น "On" ใหม่ ขณะที่สัญญาณ CTS อยู่ในสถานะ "On" ต้องรอจนกว่าสัญญาณ CTS เปลี่ยนมาอยู่ในสถานะ "Off" ก่อน เพื่อป้องกันการเกิด Overrun

Clear to send เป็นสัญญาณที่ DCE ส่งให้ DTE เพื่อแจ้งว่าพร้อมรับการส่งข้อมูลจาก DTE

Data set ready เป็นสัญญาณที่ DCE ส่งให้ DTE เพื่อแจ้งว่า DCE สามารถเชื่อมโยงไปยังปลายทาง และพร้อมที่จะติดต่อกแล้ว

Data Terminal ready เป็นสัญญาณที่ DTE ส่งให้ DCE เพื่อแจ้งว่า DTE พร้อมหรือต้องการจะติดต่อสื่อสาร ซึ่งสัญญาณ DTR นี้ต้องเกิดก่อนทาง DCE จึงจะทำการติดต่อไปยังปลายทางและเมื่อติดต่อกได้แล้วจึงส่งสัญญาณ DSR มายัง DTE เพื่อแจ้งให้รู้ว่าพร้อมรับการสื่อสารแล้วและถ้า DTR เปลี่ยนเป็น Off แปลว่า DTE ไม่ต้องการติดต่อสื่อสารแล้วทาง DCE ก็จะปิดช่องสื่อสารและเปลี่ยนสัญญาณ DSR เป็น Off ทันที คู่สัญญาณระหว่าง RTS กับ CTS เป็นเรื่องของความพร้อมเกี่ยวกับช่องสื่อสารระหว่าง DTE กับ DCE ส่วนคู่สัญญาณ DTR กับ DSR เป็นเรื่องของความพร้อมเกี่ยวกับตัวอุปกรณ์ DTE กับ DCE

Data carrier detect เป็นสัญญาณที่ DCE ส่งให้ DTE เพื่อแจ้งว่าได้รับสัญญาณพาหะจาก DCE ที่อยู่อีกด้านหนึ่งของการสื่อสาร ซึ่งหมายความว่า ช่องการสื่อสารระหว่าง DCE ทั้ง 2 ไม่ขาดตอน พร้อมทั้งจะทำการสื่อสารได้ ซึ่งอุปกรณ์ DTE หรือโปรแกรมที่ควบคุมการสื่อสารมักจะตรวจ สัญญาณนี้ ถ้าไม่อยู่ที่ On แสดงว่าช่องการสื่อสารขาด ก็จะไม่ทำการรับหรือส่งข้อมูล

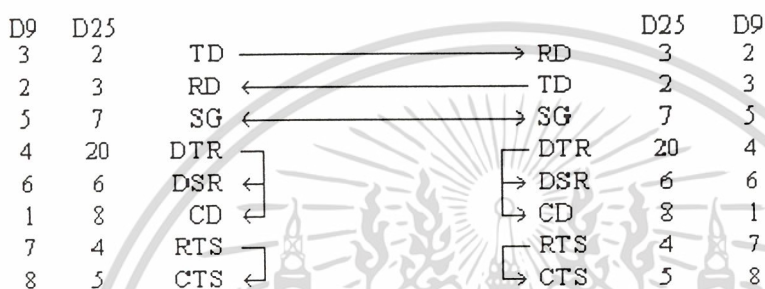
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ring Indicator เป็นสัญญาณจาก DCE แจ้งให้ DTE รู้ว่ามีการเรียกจาก DCE ที่อยู่ติดด้านหนึ่งของการสื่อสาร ซึ่งมักจะใช้ในระบบ Automatic answering

2.2.2 Null Modems

Null Modem ใช้สำหรับเชื่อมโยงระหว่าง DTE 2 ตัวเข้าด้วยกันโดยตรง ซึ่งโดยมากใช้ในการถ่ายข้อมูลระหว่างคอมพิวเตอร์ หรือใช้ในการพัฒนาระบบไมโครโปรเซสเซอร์



รูปที่ 2.11 แสดงการเชื่อมต่อของสายสัญญาณ

2.2.3 Flow Control

การติดต่อระหว่าง DTE และ DCE จะมีการควบคุมการไหลของข้อมูล เพื่อไม่ให้เกิดกา Over flow ขึ้นได้ ซึ่งมีอยู่ 2 แบบคือ Hardware flow control และ Software flow control

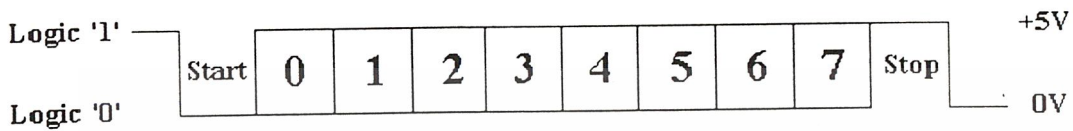
สำหรับ Software flow control มักจะเรียกว่า Xon/Xoff flow control ซึ่งใช้รหัส ASCII 17 เป็นสัญญาณ Xon และใช้รหัส ASCII 19 เป็นสัญญาณ Xoff หลักการทำงานก็ง่าย ๆ คือ Modem จะมี Buffer อยู่เมื่อ Modem รับข้อมูลจาก Computer จน Buffer ใกล้เคียงเต็มมันก็จะส่งสัญญาณ Xoff ไปให้ Computer เพื่อให้ Computer หยุดส่งข้อมูลให้มันชั่วคราว และเมื่อ Buffer มีที่ว่างถึงระดับหนึ่ง Modem ก็จะส่งสัญญาณ Xon ไปให้ Computer เพื่อให้ Computer ส่งข้อมูลให้มันต่อการควบคุมโดยวิธีนี้ประหยัดสายสัญญาณ เพราะรับ-ส่งผ่าน TD และ RD แต่อาจทำให้การสื่อสารช้าลงอย่างเห็นได้ชัดในกรณีที่ใช้กับการสื่อสารที่มี speed ต่ำ เพราะแต่ละตัวอักษร ASCII ที่รับ-ส่งจะมีขนาด 10 Bit

ส่วน Hardware flow control มักจะเรียกว่า RTS/CTS flow control จะใช้สายสัญญาณของ Serial Port ในการควบคุม ทำให้ไม่บั่นทอนความเร็วของข้อมูล หลักการทำงานคือ เมื่อ Modem มีที่ว่างเพื่อรับข้อมูล มันก็จะส่งสัญญาณ CTS ไปให้ Computer และเมื่อมันใกล้จะเต็ม มันก็จะหยุดส่งสัญญาณ CTS ไปให้ Computer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 รูปคลื่น สัญญาณ RS-232

การสื่อสารโดย RS-232 เป็นการสื่อสารแบบ asynchronous หมายความว่าสัญญาณ clock ที่ใช้ควบคุมจังหวะไม่ได้ส่งไปพร้อมกับ Data แต่จะใช้ start bit เป็นตัว sync. ในแต่ละ word ของการสื่อสาร และใช้สัญญาณ clock ภายในของแต่ละด้านเป็นตัวให้จังหวะเอง



รูปที่ 2.12 รูปคลื่นของสัญญาณที่ส่ง

แสดงลักษณะของสัญญาณจาก UART เมื่อใช้ format แบบ 8N1 คือ 8 data bits ไม่มี parity bit และมี 1 stop bit ขณะที่ idle จะอยู่ในสถานะ “Mark” หรือ logic “1” การส่งจะเริ่มจากการส่ง start bit คือ logic “0” และตามด้วย LSB bit จนหมด data bits และถ้ามี parity bit ก็ส่งที่จุดนี้แล้วลงท้ายด้วย stop bit ซึ่งมีค่าเป็น logic “1” ในรูปได้แสดง bit ที่ต่อถัดจาก stop bit ซึ่งมีค่าเป็น logic “0” หมายความว่า เป็น start bit ของ การส่ง word ถัดไป แต่ถ้ายังไม่มีการส่ง word ถัดไป ก็ต้องอยู่ในสถานะของ logic “1” ซึ่งเป็นสถานะของ idle และถ้าสายอยู่ในสถานะของ logic “0” นานกว่าเวลาของการส่ง 1 full word ระบบจะถือว่าเป็นสัญญาณ “Break” เพื่อหยุดการสื่อสาร ดังนั้นต้องไม่ลืมที่จะส่งในสายกลับสู่สถานะ idle เมื่อสิ้นสุดการส่ง การรับ-ส่งข้อมูลในลักษณะนี้เรียกว่าแบบ frame ก็คือมีกรอบปิดล้อมข้อมูลไว้ด้วย start bit และ stop bit

2.2.5 ตัวแปลงสัญญาณ RS-232

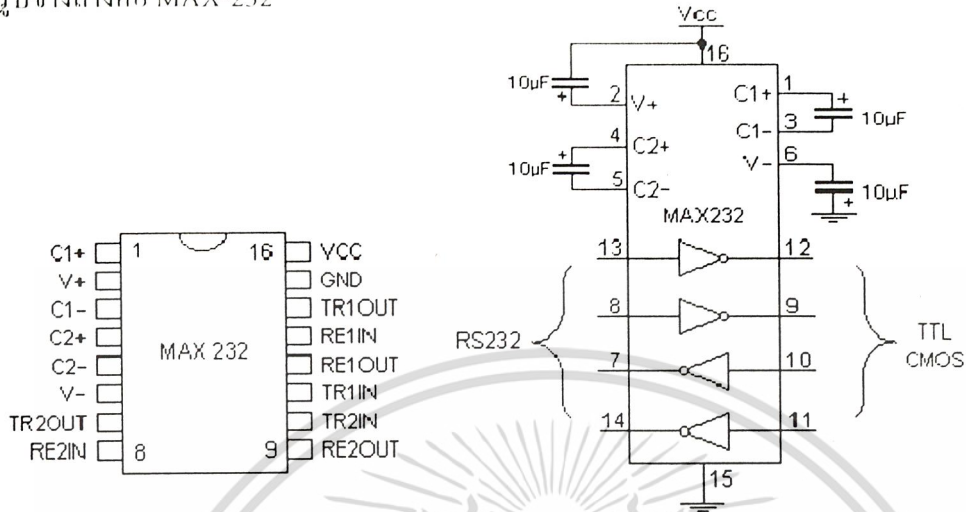
สัญญาณ RS-232 มีค่าแรงไฟต่างจากที่ใช้ใน UART ดังนั้นจึงต้องมีตัวแปลงสัญญาณเพื่อแปลงระดับสัญญาณให้เหมาะสมก่อนที่ จะเชื่อมต่อกับพอร์ตอนุกรม หรือ RS-232 port ของคอมพิวเตอร์

สัญญาณ RS-232 นั้น logic “0” จะมีค่า +3 V ถึง +25 V และ logic “1” จะมีค่า -3 V ถึง -25 V ส่วนค่าระหว่าง -3 V ถึง +3 V เป็นค่า undefined ระดับสัญญาณนี้ใช้กับทุกสัญญาณไม่ใช่เฉพาะสัญญาณรับ-ส่งข้อมูลเท่านั้นแต่ยังรวมถึงสัญญาณควบคุมต่าง ๆ เช่น DTR, RTS, CTS เป็นต้น

IC ที่ใช้มักจะเป็นเบอร์ 1488 (RS-232 Driver) และ 1489 (RS-232 Receiver) โดยภายในแต่ละตัวจะประกอบด้วย inverter 4 ตัวและต้องการไฟเลี้ยง 2 ชุดคือ +7.5 V ถึง +15 V และ -7.5 V ถึง -15 V ซึ่งอาจจะมีปัญหาในเครื่องที่มีไฟเลี้ยง +5 V เพียงชุดเดียว แต่ก็ยังมี IC อีกตัวหนึ่งคือเบอร์ MAX-232 ซึ่งมีวงจร charge pump สามารถสร้างไฟ +10 V และ -10 V จากไฟ +5 V ได้ พร้อมทั้งมี 2 Tx และ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Rx อยู่ใน package เดียวกัน และรองรับ baud rate ได้ถึง 120 Kbps จึงสะดวกมากเพราะใช้ IC เพียงตัวเดียว รูปข้างล่างคือ MAX-232



รูปที่ 2.13 แสดง โครงสร้างภายในและตำแหน่งขาต่างๆของ Max232

ส่วนการที่เราจะนำข้อมูลมาใช้งานก็ต้องแปลงเป็น parallel ก่อนซึ่งเป็นหน้าที่ของ UART ซึ่งปัจจุบันไมโครคอนโทรลเลอร์มักจะมี serial communication interface (SCI) อยู่ในตัว แต่อาจจะมีส่วนบางอย่างที่ไม่ได้ใช้ไมโครคอนโทรลเลอร์และต้องการประมวลผลข้อมูลกับพอร์ตอนุกรม เช่น ต่อ ADC เข้ากับ UART หรือต่อ LCD display เข้ากับ Serial comm. ก็ต้องใช้ UART ช่วย เช่นเบอร์ 8250 หรือ 16550A หรือเบอร์อื่น ๆ ที่ได้กล่าวมาแล้ว แต่มี UART อีกพวกหนึ่งที่แยก Tx bus กับ Rx bus ออกจากกันทำให้มีความคล่องตัวมากขึ้น

2.3 การสื่อสารข้อมูล

การสื่อสารข้อมูลมีจุดมุ่งหมายในการส่ง หรือการถ่ายทอดข่าวสารจากจุดหนึ่งไปยังอีกจุดหนึ่งอย่างถูกต้องเหมือนกับการที่ด้านส่งออกมา ในการสื่อสารรูปแบบใดๆ จะต้องประกอบไปด้วยส่วนประกอบหลักเบื้องต้น 3 ส่วนดังนี้

1. แหล่งต้นกำเนิดข่าวสาร(Source)
2. ตัวกลางในการสื่อสาร(Media)
3. แหล่งรับข่าวสาร(Receiver)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 ประเภทของการสื่อสารข้อมูล

ข้อมูลในระบบของการสื่อสารเขียนแสดงได้ด้วยค่าในระบบเลขฐานสอง โดยใช้ค่าตัวเลข 0 หรือ 1 มาประกอบกันเป็นรหัส แต่ในการส่งเราอาศัยการส่งทางไฟฟ้า ข้อมูลจะถูกเปลี่ยนให้อยู่ในรูปแบบทางไฟฟ้าโดยใช้ค่าสัญญาณไฟฟ้า 2 ระดับคือสูงและต่ำ ในการวัดอัตราเร็วในการส่งได้จากจำนวนบิตที่ส่งไปในหน่วยเวลาโดยทั่วไปใช้หน่วย bit per second (bps) ซึ่งในระบบการสื่อสารข้อมูลนั้นอาจจะมีขนาดใดก็ได้แต่ในการส่งนั้นจะต้องมีการกำหนดจุดเริ่มต้น และจุดสิ้นสุดของการแต่ละส่ง ตามปกติจะจัดแบ่งข่าวสารที่จะส่งเป็นบล็อกๆ ซึ่งหนึ่งบล็อกคือกลุ่มของบิตจำนวนหนึ่งที่ถูกส่งออกไปเป็นหน่วยเดียวกัน โดยมีการนำกลุ่มบิตนั้นไปผ่านกระบวนการบางอย่าง เพื่อใช้ในการควบคุมข้อผิดพลาดที่อาจจะเกิดขึ้น

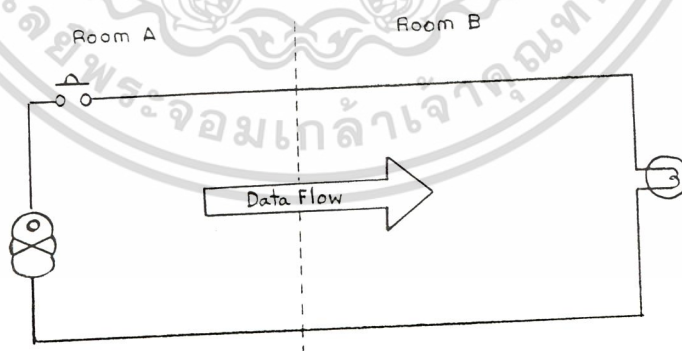
เราสามารถจำแนกวิธีการส่งข้อมูลได้หลายแบบตามคุณสมบัติต่างๆดังนี้

2.3.1.1 การจำแนกตามทิศทางการส่งข้อมูล

สามารถแบ่งการส่งข้อมูลออกได้เป็น 3 ชนิด ดังนี้

2.3.1.1.1 การรับส่งข้อมูลทางเดียว (Simplex)

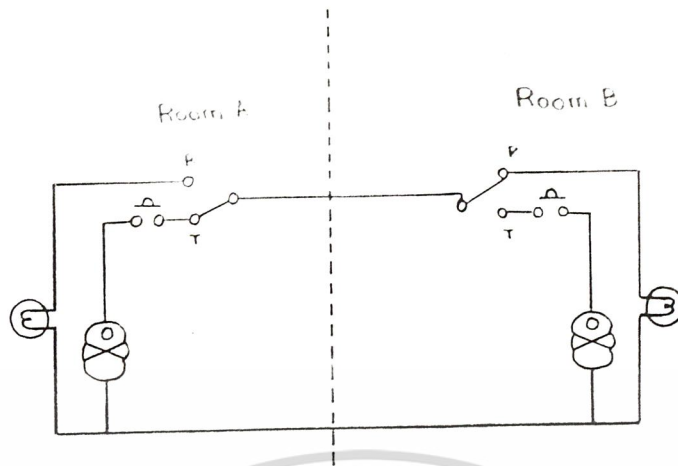
เป็นการสื่อสารทางเดียว ที่เห็นได้ชัดก็คือ การรับส่งโทรทัศน์ และวิทยุกระจายเสียงนั่นเอง สถานีโทรทัศน์จะเป็นตัวส่งและเครื่องรับทำหน้าที่รับเพียงอย่างเดียว จะส่งข่าวสารมายังสถานีส่งไม่ได้



รูปที่ 2.14 แสดงการรับส่งข้อมูลแบบทางเดียว (Simplex)

2.3.1.1.2 การรับส่งแบบผลัดกันส่ง (Half Duplex)

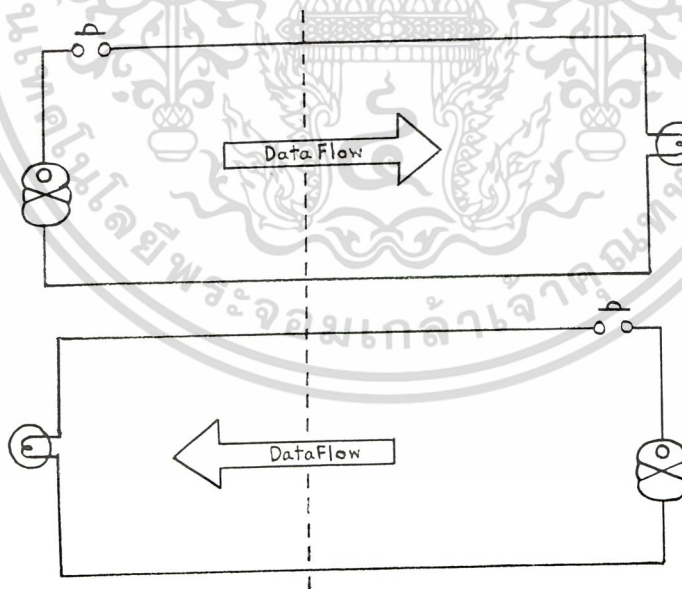
มีคุณสมบัติสามารถรับส่งข้อมูลได้ แต่จะต้องสลับการส่งโดยจะส่งพร้อมกันทั้งสองด้านไม่ได้ อุปกรณ์ที่ใช้ในการติดต่อในแบบนี้ ได้แก่ วิทยุสื่อสาร และอินเตอร์คอม เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 แสดงการรับส่งข้อมูลสวนทางกันได้แบบผลัดการส่ง(Half Duplex)

2.3.1.1.3 การรับส่งสวนทางได้พร้อมกัน (Full Duplex)

การรับส่งแบบนี้ผู้รับและผู้ส่งสามารถรับและส่งพร้อมๆกันได้ ในเวลาเดียวกันโดยไม่ต้องรอให้อีกฝ่ายหนึ่งส่งจบเสียก่อน เช่น การพูดโทรศัพท์ของเรา



รูปที่ 2.16 แสดงการรับส่งข้อมูลแบบสวนทางกันได้พร้อมกัน (Full Duplex)

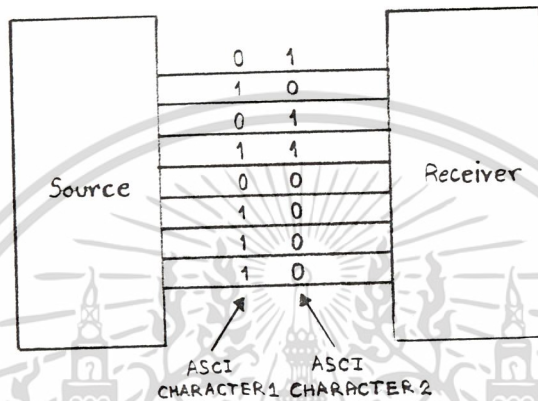
2.3.1.2 การจำแนกตามลักษณะการจัดข้อมูล

สามารถแบ่งการส่งข้อมูลออกได้เป็น 2 ชนิดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.2.1 การส่งข้อมูลแบบขนาน (Parallel Transmission)

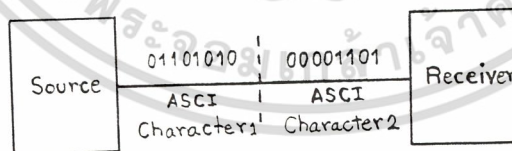
เป็นการส่งข้อมูลที่ละไบต์ คือส่งทุกบิตของรหัสที่ประกอบขึ้นเป็นอักขระไปพร้อมๆ กันในเวลาเดียวกัน มีข้อดีคือ ความสามารถในการส่งข้อมูลที่สูงขึ้น แต่จำนวนช่องทางการสื่อสารที่จะต้องมีย่านวนเท่ากับจำนวนที่บิตที่ประกอบขึ้นเป็นอักขระ ซึ่งต้อง ใช้การมัดตีเพื่อกข้อมูลต่างๆ



รูปที่ 2.17 แสดงการส่งข้อมูลแบบขนาน

2.3.1.2.2 การส่งข้อมูลแบบอนุกรม (Series Transmission)

การส่งข้อมูลแบบนี้จะกระทำทีละบิตด้วยแกนเนลเพียงแกนเนลเดียว ทางด้านรับจะมีอุปกรณ์สำหรับจัดการข้อมูลดังกล่าวเป็นชุดอักขระ ตามข้อตกลงระหว่างอุปกรณ์ปลายทางทั้งสองที่สื่อสารกัน



รูปที่ 2.18 แสดงการส่งข้อมูลแบบอนุกรม

2.3.1.3 การจำแนกตามความสัมพันธ์ของข้อมูล

ในการสื่อสารข้อมูลนั้นจะต้องพิจารณาถึงความสัมพันธ์ระหว่างข้อมูล 2 ชนิด คือ

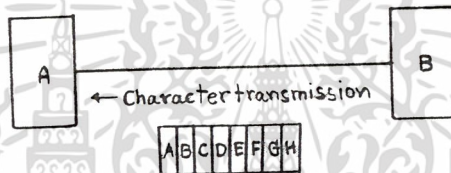
- ความสัมพันธ์ระหว่างบิต (Bit Synchronization)
- ความสัมพันธ์ของอักขระ (Character Synchronization)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการสื่อสารข้อมูลเราจำแนกวิธีการส่งข้อมูลตามการกำหนดความสัมพันธ์ของข้อมูล
ได้ 2 แบบ คือ

2.3.1.3.1 การส่งแบบสัมพันธ์ (Synchronous Transmission)

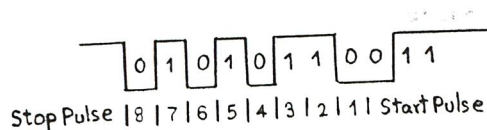
การส่งแบบนี้ใช้สำหรับการส่งข้อมูลทั้งหมดไปในครั้งเดียวดังรูป 2.19 เทคนิคการส่งแบบนี้ ช่วงเวลาระหว่างบิตต่อบิตจะมีค่าเท่ากัน การส่งมีลักษณะคล้ายกับการส่งข่าวสารในรูปของเลขฐานสองที่มีจำนวนติดต่อกันไป โดยไม่ได้แยกว่าความยาวใดเป็นช่วงอักขระใดในระบบเช่นนี้ บิตแต่ละบิตจะมีความยาวเท่ากัน ตัวอักษรแต่ละตัวมีช่วงเวลาห่างกันเท่ากับศูนย์ ทางด้านรับนั้นเพียงหาว่าบิตแรกของตัวอักษรตัวแรกคือบิตใด และทราบขนาดหรือจำนวนบิตในหนึ่งตัวอักษรพร้อมทั้งความเร็วในการส่งก็สามารถแยกข่าวสารของแต่ละอักษรออกมาได้



รูปที่ 2.19 แสดงการส่งข้อมูลแบบสัมพันธ์

2.3.1.3.2 การส่งแบบไม่สัมพันธ์ (Asynchronous Transmission)

การส่งแบบนี้ตัวอักษรจะถูกส่งออกไปที่เวลาใดๆ ก็ได้โดยไม่จำเป็นต้องมีความสัมพันธ์ระหว่างตัวอักษรว่าต้องมีเวลาที่แน่นอนอย่างไร ทางด้านรับจะต้องทราบถึงบิตเริ่มต้นของรหัสแต่ละตัวว่าเริ่มต้นเมื่อใดซึ่งเมื่อใดซึ่งสามารถกระทำได้โดยการเพิ่มบิตที่เรียกว่า พัลส์เริ่มต้น (Start Pulse) โดยเติมเข้าไปที่ข้างหน้าสุดของทุกอักขระ และเมื่อการส่งส่งครบทุกบิตของตัวอักษรแล้วจะต้องมีบิตสำหรับบอกถึงการสิ้นสุดที่เรียกว่า พัลส์การสิ้นสุด (Stop Pulse) ส่งมาให้ทางด้านรับมีเวลาสำหรับการเตรียมข้อมูลของตัวอักษรตัวต่อไปดังรูป 2.20 บางครั้งจึงเรียกระบบการส่งแบบนี้ว่า ระบบการส่งแบบเริ่มหยุด (Start-stop Transmission)



รูปที่ 2.20 แสดงการส่งข้อมูลแบบไม่สัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ดีซีมอเตอร์ (DC Motor)

2.4.1 หลักการทำงานของดีซีมอเตอร์

ดีซีมอเตอร์เป็นทรานสดิวเซอร์แรงบิดซึ่งมีการออกแบบให้มีคุณลักษณะพิเศษคือ แรงบิดของเพลลาของมอเตอร์จะเป็นสัดส่วนโดยตรงกับกระแสอาร์เมเจอร์ ความสัมพันธ์ระหว่างแรงบิดของเพลลาและกระแสคือ

$$T = K \Phi I$$

เมื่อ	T	คือแรงบิดของเพลลามีหน่วยเป็น นิวตัน-เมตร
	Φ	คือเส้นแรงแม่เหล็กมีหน่วยเป็น เวเบอร์
	I	คือกระแสมีหน่วยเป็น แอมป์
	K	คือค่าคงตัว

จะเห็นได้ว่าแรงบิดของเพลลาจะเป็นสัดส่วน โดยตรงกับผลคูณของเส้นแรงแม่เหล็กและกระแส การที่ขดลวดตัวนำเคลื่อนที่ในสนามแม่เหล็กจะทำให้เกิดแรงดันตกคร่อมตัวมันเองโดยแรงดันนี้จะเป็นสัดส่วนกับความเร็วของเพลลาของมอเตอร์และเส้นแรงแม่เหล็ก โดยมีความสัมพันธ์ดังนี้

$$E = K \Phi \omega$$

เมื่อ	E	คือแรงดันย้อนกลับมีหน่วยเป็น โวลต์
	Φ	คือเส้นแรงแม่เหล็กมีหน่วยเป็น เวเบอร์
	ω	คือความเร็วของมอเตอร์มีหน่วยเป็น เรเดียน/วินาที
	K	คือค่าคงตัว

2.4.2 ดีซีมอเตอร์แบบเส้นแรงแม่เหล็กคงที่

ฟิลด์ (Field) ของมอเตอร์ชนิดนี้นิยมใช้แม่เหล็กถาวร มอเตอร์ชนิดนี้จึงอาจจะเรียกว่า มอเตอร์แบบแม่เหล็กถาวร (Permanent-Magnet Motors) มอเตอร์แบบแม่เหล็กถาวรนี้ไม่ต้องการ กระตุ้นเพื่อให้เกิดสนามแม่เหล็ก เนื่องจากมีแกนขั้ว (Core) ทำด้วยอัลลอยซึ่งสามารถรักษาคุณสมบัติ ความเป็นแม่เหล็กหรือขดลวดฟิลด์ มอเตอร์ชนิดนี้จึงมีประสิทธิภาพสูงกว่ามอเตอร์ชนิดซันท์หรือมอเตอร์คอมปาวด์ที่มีพิคตเท่ากันนอกจากนั้นยังควบคุมความเร็วและทิศทางการหมุนได้ง่ายโดยการกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดและชั่วแรงแรงดันที่ป้อนให้แก่มอเตอร์ ซึ่งหมายความว่าถ้ากลับชั่วแรงแรงดันจะทำให้ทิศทางของกระแสที่อาร์เมเจอร์เปลี่ยนแปลงไปเพียงอย่างเดียวเท่านั้น แต่ไม่ทำให้ฟลักซ์เปลี่ยนแปลงจึงได้อัตราส่วนระหว่างกระแสอาร์เมเจอร์และแรงบิดมีค่าคงที่ ดังมีความสำคัญดังนี้

$$T=KtI$$

$$E=Ke\Omega$$

ระบบนี้จะให้ความสัมพันธ์ระหว่างกระแสอาร์เมเจอร์, แรงบิด และความเร็วยูในลักษณะเชิงเส้น สมการทางไฟฟ้าของดีซีมอเตอร์แบบนี้เขียนได้เป็น

$$V=Ke\Omega+L(di/dt)+Ri$$

เมื่อ	V	คือแรงดันที่ป้อนให้มอเตอร์
	Ke	คือค่าคงที่ของแรงดันย้อนกลับ
	L	คือค่าความเหนี่ยวนำของขดอาร์เมเจอร์
	R	คือความต้านทานที่ขั้วของมอเตอร์

2.4.3 ดีซีมอเตอร์แบบอาร์เมเจอร์เป็นแกนหลัก

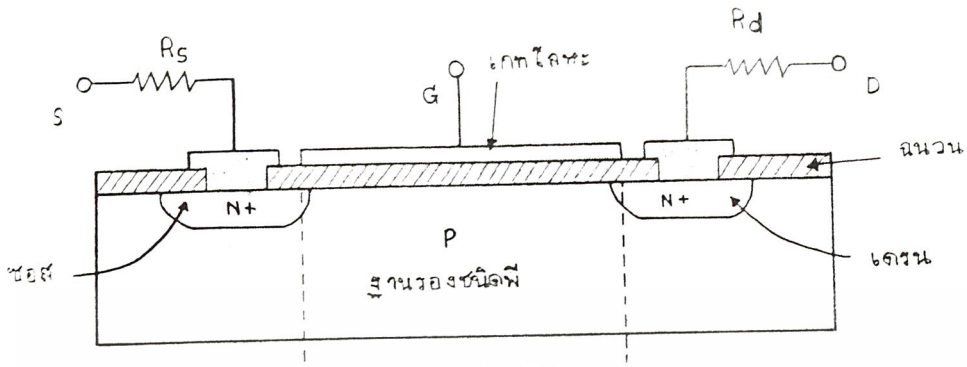
โครงสร้างของโรเตอร์และสเตเตอร์ของมอเตอร์แบบแกนหลัก โครงสร้างของมอเตอร์แบบนี้โมเมนต์ของแรงเฉื่อยสูงที่สุด และมีค่าอินดักแตนซ์ของโรเตอร์สูงที่สุดด้วย ดังนั้นมอเตอร์นี้จึงมีปริมาณการจุความร้อนได้สูง และสามารถจะทนโอเวอร์โหลดได้ในระยะเวลาที่ยาวนานโดยไม่ทำให้มอเตอร์เสียหาย

2.5 เพาเวอร์มอสเฟต

เพาเวอร์มอสเฟตเป็นอุปกรณ์ประเภทอุปกรณ์อิเล็กทรอนิกส์กำลังชนิดหนึ่ง ซึ่งมีลักษณะโครงสร้างของเพาเวอร์มอสเฟตจะแบ่งเป็น N-channel และ P-channel จากโครงสร้างจะพบว่าเกทกับซอร์สจะถูกแยกออกจากกันด้วยชั้นของซิลิกอน ซึ่งเปรียบเสมือนมีความจุไฟฟ้าอยู่ เมื่อป้อนแรงดันที่เหมาะสมเข้าที่ขาเกทจะทำให้ชั้นของซิลิกอนเกิดการเหนี่ยวนำจนเกิดมีกระแสไหล เพาเวอร์มอสเฟตนิยมใช้มากในวงจรสวิตช์ซึ่งเพาเวอร์ซัพพลาย และการควบคุมมอเตอร์เนื่องจากมีข้อดีที่ว่าเพาเวอร์ทรานซิสเตอร์ คือ

- กำลังสูญเสียขณะสวิตช์ต่ำ
- มีสัญญาณรบกวนต่ำกว่าแบบไบโพลาร์
- มีคุณสมบัติการทำงานที่มั่นคงและเสถียรภาพทางอุณหภูมิ
- ใช้ในวงจรความถี่สูงได้ดี

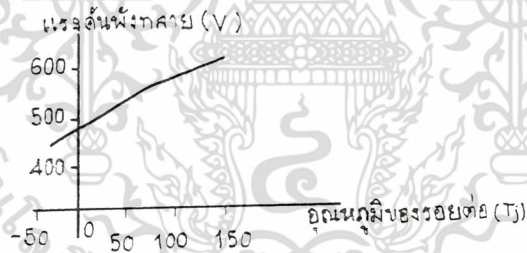
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 โครงสร้างของเพาเวอร์มอสเฟต

2.5.1 คุณลักษณะของเพาเวอร์มอสเฟต

แรงดันพังทลายเป็นระหว่างขาเดรนกับขาซอร์สดังรูป 2.23 โดยที่อุณหภูมิต่ำๆ ค่าแรงดันพังทลายจะลดลงอย่างเห็นได้ชัด ขณะที่เพาเวอร์มอสเฟตนำกระแสอย่างสมบูรณ์แรงดันที่ตกคร่อมระหว่างขาเดรนกับขาซอร์สจะแปรผันตรงกับกระแสที่ไหลผ่านตัวมัน



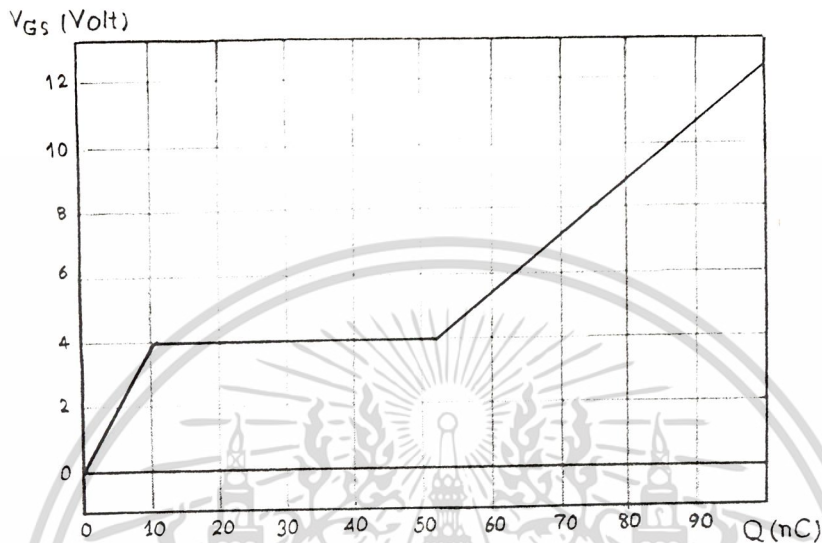
รูปที่ 2.22 แรงดันพังทลายที่เดรนกับซอร์สเมื่ออุณหภูมิขอรอยต่อเปลี่ยนแปลงไป

2.5.2 เงื่อนไขของวงจรขับเพาเวอร์มอสเฟต

การขับเพาเวอร์มอสเฟตให้นำกระแสที่แตกต่างจากการขับกระแสไบอัส ในไบโพลาร์ทรานซิสเตอร์ เนื่องจากมีเงื่อนไขการไบอัสที่แตกต่างกัน สำหรับไบโพลาร์ทรานซิสเตอร์กระแสจะไหลผ่านคอลเลกเตอร์และอิมิตเตอร์ได้ก็ต่อเมื่อมีกระแสไหลไบอัสไหลผ่านที่เบสและอิมิตเตอร์ แต่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพาเวอร์ MOSFET จะมีกระแสไหลผ่านเดรนและซอร์ส ก็ต่อเมื่อแรงดันตกร่วมขาเกตและซอร์สมีค่าอย่างต่ำเท่ากับค่าแรงดันขีดเริ่ม (Threshold Voltage) ของมัน



รูปที่ 2.23 แสดงตัวอย่างของ Gate Charge Chart

ลักษณะของกราฟแบ่งออกได้เป็น 3 ช่วง ตามผลของประจุที่เพิ่มขึ้น คือ

1. ช่วงเวลาหน่วงก่อนเริ่มนำกระแส (Time-on Delay Time) $t=0$
2. ช่วงเวลาเริ่มนำกระแส (Rise Time) $t=0-1$
3. ช่วงเวลาสะสมประจุส่วนเกิน (Excess Charge Time) $t=2-3$

เมื่อเพาเวอร์ MOSFET เริ่มชาร์จประจุที่ขาเกต จนกระทั่งพ้นช่วงเวลาหน่วงก่อนเริ่มนำกระแส เมื่อแรงดันที่ขาเกตมากกว่าแรงดันขีดเริ่มจึงจะเริ่มมีกระแสไหลผ่านเดรนและซอร์ส ที่เวลา $t=0$ ค่าของแรงดันตกร่วมเดรนกับซอร์ส V_{ds} จะไม่ลดลงจนกว่าจะถึงเวลา $t=1$ แรงดันตกร่วมเดรนกับซอร์ส V_{ds} จึงมีค่าลดลงอย่างรวดเร็ว เพาเวอร์ MOSFET จะนำกระแสได้เต็มที่ในช่วงเวลาเท่ากับ $t=1$ ถึง $t=2$ นี้เอง และช่วงนี้แรงดันที่ขาเกต V_{gs} จะคงที่

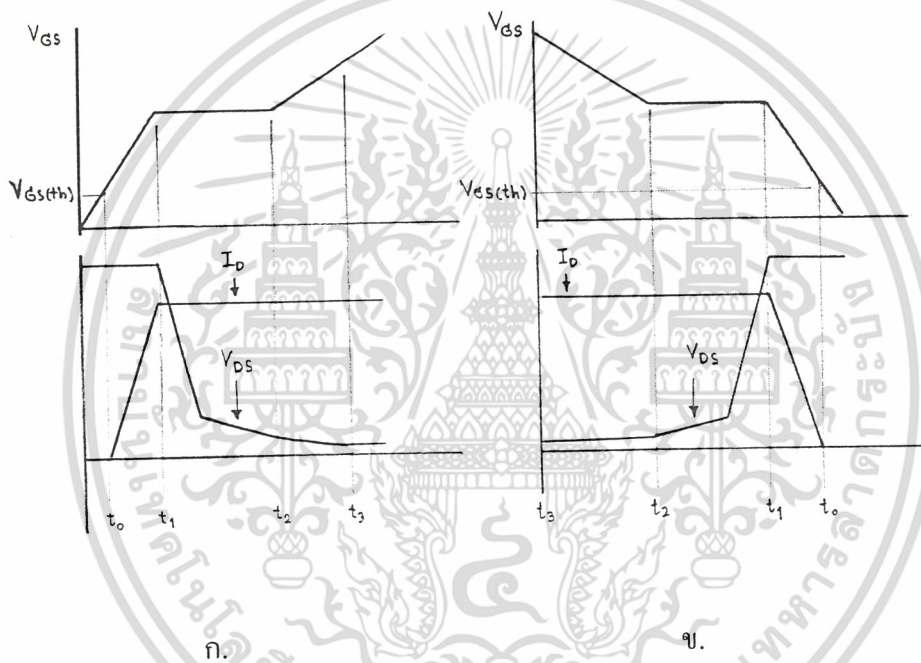
ในช่วงเวลาสะสมประจุส่วนเกินค่าความต้านทานระหว่างเดรนและซอร์ส จะลดลงเรื่อยๆ หากปล่อยให้มีการสะสมประจุต่อไปในช่วงเวลา $t=2$ ถึง $t=3$ ประจุที่สะสมเพิ่มขึ้นจะทำให้เกิดการหน่วงขณะเริ่มหยุดนำกระแส เนื่องจากเพาเวอร์ MOSFET จะต้องใช้เวลามากในการคายประจุส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกินนี้ทิ้งไป ดังนั้นการขับจากเกตด้วยแรงดันที่สูงเกินความจำเป็น จะทำให้ช่วงเวลาเริ่มหยุดนำกระแสเพิ่มขึ้นด้วย

2.5.3 การกำหนดเวลาในการเปลี่ยนสถานะ

ปกติแล้วผู้ผลิตจะให้กราฟค่า V_{gs} ที่เพิ่มขึ้นในขณะที่ค่าประจุสะสมที่ขาเกตเพิ่มขึ้น หรือที่เรียกว่า Gate Charge Chart มาในค่าดัชนีด้วย กราฟนี้มีประโยชน์มากในการคำนวณค่ากระแสไบอัสเกตและเวลาในการเปลี่ยนสถานะของเพาเวอร์มอสเฟต



รูปที่ 2.24 (ก) แสดงการชาร์จประจุที่ขาเกตตามเวลาที่มีผลต่อการนำกระแส

(ข) แสดงผลเมื่อเริ่มหยุดนำกระแสของเพาเวอร์มอสเฟต

การหยุดนำกระแสของเพาเวอร์มอสเฟตจึงทำได้ง่ายๆ ด้วยการคายประจุที่ขาเกตทิ้งไป เพื่ออัดประจุและคายประจุมีค่าเท่ากันช่วงเวลาเริ่มนำกระแสและช่วงเวลาหยุดนำกระแส จะมีค่าเท่ากันด้วย ยกเว้นหากมีประจุสะสมมากในช่วงเวลา $t=2$ ถึง $t=3$ การเริ่มหยุดนำกระแสจะมีช่วงเวลานางเพิ่มขึ้น เนื่องจากต้องใช้เวลาในการคายประจุส่วนเกินทิ้งไป

จะเห็นได้ว่าการขับเพาเวอร์มอสเฟตให้นำกระแสได้นั้น จะต้องมีการอัดและคายประจุที่ขาเกต วงจรขับเกตของเพาเวอร์มอสเฟตจะต้องมีคุณสมบัติของการจ่ายกระแสและนำกระแสได้ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประมาณ 200 ถึง 400 มิลลิแอมแปร์ด้วย และต้องให้แรงดันตกคร่อมที่ขาเกต Vgs มีค่ามากพอเพื่อที่เพาเวอร์มอสเฟตจะทำงานได้เต็มที่ (ประมาณ 10 โวลต์)

สิ่งสำคัญอีกประการคือ การเลือกใช้งานเพาเวอร์มอสเฟตเราควรจะต้องเลือกเพาเวอร์มอสเฟต ที่มีพิคคกระแสและแรงดันมากกว่ามอเตอร์ประมาณ 1.5-2 เท่า

2.6 การเชื่อมต่อทางแสง

ตัวเชื่อมโยงทางแสง (Optocoupler) หรือตัวแยกโดยใช้แสง (Opto Isolater) เป็นอุปกรณ์ที่นำมาเชื่อมโยงสัญญาณต่างๆ ของวงจรที่มีกราวด์ต่างกัน สามารถป้องกันการรบกวนซึ่งกันและกันระหว่างภาคอินพุตกับภาคเอาต์พุตได้อย่างเด็ดขาด ออปโตคัปเปิ้ลเป็นอุปกรณ์ที่ประกอบด้วยแหล่งกำเนิดแสงและตัวตรวจจับแสงอยู่ในตัวเดียวกัน แหล่งกำเนิดแสงส่วนใหญ่จะใช้ไดโอดเปล่งแสงอินฟราเรดที่ทำมาจากแกเลียมอาร์เซไนด์ (GaAs) ส่วนตัวตรวจจับแสง (Photo Detector) สามารถเป็นไปได้ทั้งโฟโตทรานซิสเตอร์, โฟโตไดโอด, SCR และ TRIAC ในออปโตคัปเปิ้ลสัญญาณอินพุตจะถูกเปลี่ยนเป็นสัญญาณแสงแล้ว จึงส่งสัญญาณแสงนี้ไปยังโฟโตดีเทคเตอร์ ออปโตคัปเปิ้ลถูกออกแบบมาเพื่อป้องกันอุปกรณ์อิเล็กทรอนิกส์ไม่ให้รับแรงกระชากสูงๆ ซึ่งเป็นต้นเหตุที่ทำให้เกิดสัญญาณเอาต์พุตที่ไม่ถูกต้อง

2.7 หลักการออกแบบส่วนประกอบและโครงสร้างต่างๆ

2.7.1 เพลา (Shaft)

เพลาอาจมีชื่อเรียกแตกต่างกันออกไป ตามลักษณะการใช้งานดังนี้

- เพลา (Shaft) เป็นชิ้นส่วนที่หมุนและใช้ในการส่งกำลัง

- แกน (Axle) เป็นชิ้นส่วนลักษณะเดียวกับเพลาแต่ไม่หมุน ส่วนมากเป็นตัวรองรับชิ้นส่วนที่หมุน เช่น ล้อ ล้อสายพาน เป็นต้น อย่างไรก็ตามทั้งเพลาและแกนก็นิยมเรียกรวมกันว่าเพลา ไม่ว่าจะชิ้นส่วนจะหมุนหรือไม่หมุนก็ตาม

- สปินเดิล (Spindl) เป็นเพลาขนาดสั้นที่ไม่หมุน เช่น เพลาที่แทนกิ้ง (Head Shaft)

เป็นเพลาที่ติดเป็นชิ้นส่วนต่อเนื่องกับเครื่องยนต์ มอเตอร์ หรือเครื่องต้นกำลังอื่นๆ มีขนาดรูปร่างและส่วนยื่นออกมาต่อกับเพลาอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **เพลานว (Line Shaft)** หรือเพลาส่งกำลัง (Power Transmission Shaft) หรือเพลาเมน (Main Shaft) เป็นเพลาคู่ที่ต่อตรงจากเครื่องต้นกำลังและใช้ในการส่งกำลังไปยังเครื่องจักรอื่นๆ โดยเฉพาะ

- **แจ็กชาฟ (Jack Shaft)** หรือเค้านเตอร์ชาฟ (Counter Shaft) เป็นเพลานาขนาดสั้นที่ต่อระหว่างเครื่องต้นกำลังกับเพลาเมนหรือเครื่องจักรกล

- **เพลอ่อน (Flexible Shaft)** เป็นเพลที่สามารถอ่อนตัวหรือโค้งงอได้ เพลประเภทนี้ทำด้วยสายลวดใหญ่ (Cable) ลวดสปริงหรือลวดเกลียว (Wire Rope) ใช้ในการส่งกำลังในลักษณะที่แกนหมุนทำมุมกันได้แต่กำลังส่งได้น้อย

เพลอาจจะรับแรงดึง แรงกด แรงบิด หรือแรงดัน หรือหลายแรงรวมกันก็ได้ ดังนั้นการคำนวณก็ต้องใช้ความเค้นผสมเข้าช่วย แรงเหล่านี้ยังอาจจะมีเปลี่ยนแปลงขนาดตลอดเวลาทำให้เพลเสียหายเพราะความล้าได้ ดังนั้นจะต้องออกแบบเพลามีความแข็งแรงเพียงพอสำหรับการใช้งาน นอกจากนี้เพลายังมีความแข็งแกร่ง (Rigidity) เพียงพอ เพื่อลดมุมบิดภายในเพลให้อยู่ในขีดจำกัดที่เหมาะสม ระยะโก่ง (Deflection) ของเพลาก็เป็นสิ่งสำคัญในการกำหนดขนาดเพลาดังเดียวกัน เพราะถ้าเพลามีระยะโก่งมากก็จะเกิดการแกว่งขณะหมุนทำให้เกิดความเร็ววิกฤต (Critical Speed) ของเพลาลดลง ซึ่งอาจทำให้เพลามีการสั่นอย่างรุนแรง ในขณะที่ความเร็วของเพลเข้าใกล้ความเร็ววิกฤตนี้ได้ ระยะโก่งนี้ยังมีผลต่อการเลือกชนิดของเพลและที่รองรับเพล เช่น บอลเบริง (Ball Bearing) ก็ต้องมีการเยื้องแนว (Misalignment) ในการใช้งานที่เหมาะสมกับเพลด้วย

การพิจารณาในการออกแบบเพล

การคำนวณหาขนาดของเพลที่เหมาะสม ขึ้นอยู่กับลักษณะการใช้งานในบางครั้ง การหาขนาดของเพล เพื่อให้เพลทนต่อแรงที่มากระทำอย่างเดียวกันนั้นไม่เพียงพอ เช่น ในกรณีของเพลาลูกเบี้ยว (Cam Shaft) ในเครื่องยนต์สันดาปภายในต้องการให้มรดำแหน่งที่เที่ยงตรง ดังนั้นมุมบิดของเพลที่เกิดขึ้นในขณะที่ใช้งานจะต้องมีค่าไม่มากกว่าที่กำหนดไว้ เป็นต้น นั่นคือเพลจะต้องมีความแข็งแรงอยู่ในพิสัยที่ต้องการ ถ้ามุมบิดมากเกินไปนอกจากจะเสียความเที่ยงตรงทางด้านตำแหน่งแล้ว ยังอาจจะก่อให้เกิดความสั่นสะเทือนซึ่งมีผลทำให้เฟืองและเบริงที่รองรับเพลอยู่เกิดความเสียหายได้ง่าย

ความแข็งแรงที่สำคัญอีกอย่างหนึ่งก็คือ ความแข็งแรงทางด้านระยะโก่ง เพราะจะต้องใช้ระยะโก่งของเพลที่อยู่ใต้แรงภายนอกเป็นตัวสำคัญในการกำหนดระยะเบียดด้วย (Clearance) ระหว่างล้อสายพาน เฟือง โครงสร้างของเครื่องจักรตลอดจนการเลือกชนิดของเบริง สำหรับรองรับเพลที่มีมุมให้เหมาะสม ถ้าเพลามีระยะโก่งมากเกินไปจะทำให้ความยาวของฟันเฟืองส่วนที่สัมผัสกันหรือเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขบกันลดลง เป็นผลให้อัตราส่วนการขบ (Contact Ratio) ของเฟืองลดลงด้วย ทำให้ส่งกำลังของเฟืองไม่ราบเรียบเท่าที่ควร การเลือกแบร์ริงชนิดที่อนุญาตให้มีการเอียงแนวได้พอเหมาะสมกับระยะโง่งของเพลลาที่จะเกิดขึ้น

จากเหตุผลดังกล่าวมาแล้วจะเห็นว่าขนาดของเพลลา อาจจะหามาได้โดยใช้ความแข็งแรงที่ต้องการ แทนที่จะเป็นความแข็งแรงในด้านการรับแรงภายนอกก็ได้ การหาระยะโง่งของเพลลาอาจทำได้โดยใช้วิธีการศาสตร์วัสดุ

2.7.2 แบร์ริง (Bearing)

เป็นตัวช่วยลดความเสียดทาน ความสึกหรอและความร้อน ที่เกิดขึ้นในชิ้นส่วนของเครื่องจักรที่มีการเสียดสีกัน การหล่อลื่นทำได้โดยการใส่สารหล่อลื่น (Lubrication Oil) หรือจารบี (Grease) หรืออากาศและอื่นๆ ซึ่งขึ้นอยู่กับการใช้งานในการเลือกใช้แบร์ริงนั้นจะต้องเลือกใช้แบบให้เหมาะสมกับเพลลาที่เราเลือกใช้ด้วย

2.7.3 เฟือง (Gear)

เฟือง เป็นชิ้นส่วนของเครื่องจักรกลอีกอย่างที่มีความสำคัญมากมีการใช้ในงานด้านวิศวกรรมอย่างกว้างขวาง เพราะสามารถส่งกำลังได้แม่นยำเที่ยงตรง ไม่มีการสั่นไหว ให้อัตราทดที่แน่นอนเช่นเดียวกับการส่งกำลังด้วยสายพาน ในการเลือกใช้เฟืองจะต้องเลือกประเภทของเฟืองให้เหมาะสมกับโซ่หรือสายพาน และต้องคำนึงถึงรูปแบบของการส่งกำลังด้วย

2.7.4 โซ่ (Chain)

การขับเคลื่อนโซ่มีการใช้งานกันมากในงานเครื่องจักรกล เนื่องจากมีลักษณะการขับเคลื่อนด้วยสายพาน โดยโซ่จะคล้องอยู่กับล้อโซ่หรือเฟืองโซ่ ซึ่งติดอยู่บนเพลลาขับและเพลลาตาม อัตราทดของการขับเคลื่อนขึ้นอยู่กับขนาดของเฟืองโซ่ทั้งสอง และการขับเคลื่อนจะไม่มีการสลลลื่นระหว่างเฟืองกับโซ่

การขับเคลื่อนโซ่มีข้อดีอยู่ระหว่างการขับเคลื่อนด้วยสายพาน และการขับเคลื่อนเฟืองด้านราคา สมรรถนะในการส่งกำลังและการบำรุงรักษา โซ่สามารถขับได้ในระยะทางไกลกว่าการขับเคลื่อนด้วยสายพาน และการขับได้พร้อมกันหลายๆเพลลา ซึ่งมีทิศทางการหมุนตามกันหรือสวนกันก็ได้ แต่การใช้โซ่จะมีข้อจำกัดในการส่งกำลังกล่าวคือเมื่อใช้งานที่มีความเร็วปานกลางจนถึงความเร็วสูงสุด จะทำให้เกิดการสึกหรอที่ข้อต่อ และความต้านทานแรงดึงของวัสดุชิ้นส่วนโซ่ นอกจากนี้แล้วถ้าใช้งานด้วยความเร็วสูงสุดจะต้องคำนึงถึงแรงที่เกิดขึ้นในโซ่ เนื่องจากแรงหนีศูนย์กลางในการขับเคลื่อนโดยโซ่หลายเส้นขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กัน อาจทำให้แรงที่กระทำกับโซ่ไม่กระจายไปเท่ากันตลอดความกว้างของโซ่ และถ้าหล่อลื่นไม่ดีพอ ก็จะเป็นสาเหตุประการสำคัญที่ทำให้โซ่สึกหรอได้มาก

โซ่ที่คล้องอยู่บนเฟืองโซ่จะมีลักษณะคล้ายรูปหลายเหลี่ยมบนเฟืองโซ่ ทำให้แกนของแรงในแนวสัมผัสกับเฟืองโซ่เปลี่ยนแปลงตลอดเวลาที่ส่งกำลัง นอกจากนี้ในระหว่างเคลื่อนที่เข้าหา และเคลื่อนที่ออกจากเฟืองโซ่ ข้อต่อโซ่แต่ละข้อจะเกิดการหมุนจนทำให้เกิดแรงเสียดทานขึ้น เป็นผลทำให้สูญเสียกำลังและสึกหรอ ผลต่อมาก็คือระยะฟิตของโซ่จะเพิ่มขึ้นจนอาจจะหลุดออกจากเฟืองโซ่ได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การออกแบบส่วนต่างๆ ของหุ่นยนต์

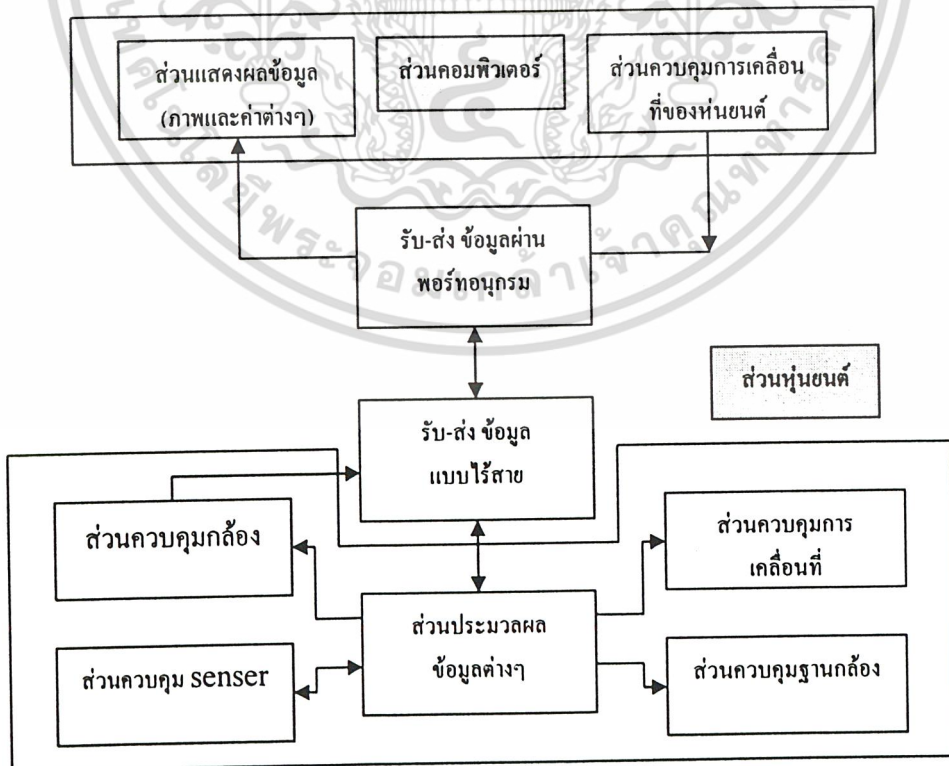
3.1 โครงสร้างการทำงานของหุ่นยนต์

โครงสร้างการทำงานหลักๆ ของหุ่นยนต์ สามารถแบ่งได้ 3 ส่วนใหญ่ๆ ดังรูปที่ 3.1



รูปที่ 3.1 แสดงโครงสร้างการทำงานหลักๆ ของหุ่นยนต์

จากรูปที่ 3.1 ในส่วนของคอมพิวเตอร์จะทำหน้าที่เป็นตัวควบคุมการเคลื่อนที่ของหุ่นยนต์ และ ยังรับ-ส่งข้อมูลระหว่างหุ่นยนต์กับคอมพิวเตอร์ โดยจะใช้พอร์ทอนุกรมเป็นส่วนรับ-ส่งข้อมูล ส่วนของการสื่อสารแบบไร้สายก็จะทำหน้าที่เป็นตัวส่งผ่านข้อมูลระหว่างหุ่นยนต์กับคอมพิวเตอร์ โดยจะผลัดกันส่งข้อมูลกันหรือเป็นการส่งแบบฮาร์ฟ ดูเพล็กซ์ (Half Duplex) ส่วนสุดท้ายคือ หุ่นยนต์ ซึ่งเป็นส่วนที่มีหน้าที่การทำงานหลากหลายมาก ดังนั้นเราจึงอธิบายการทำงานโดยละเอียด ดังรูปที่ 3.2

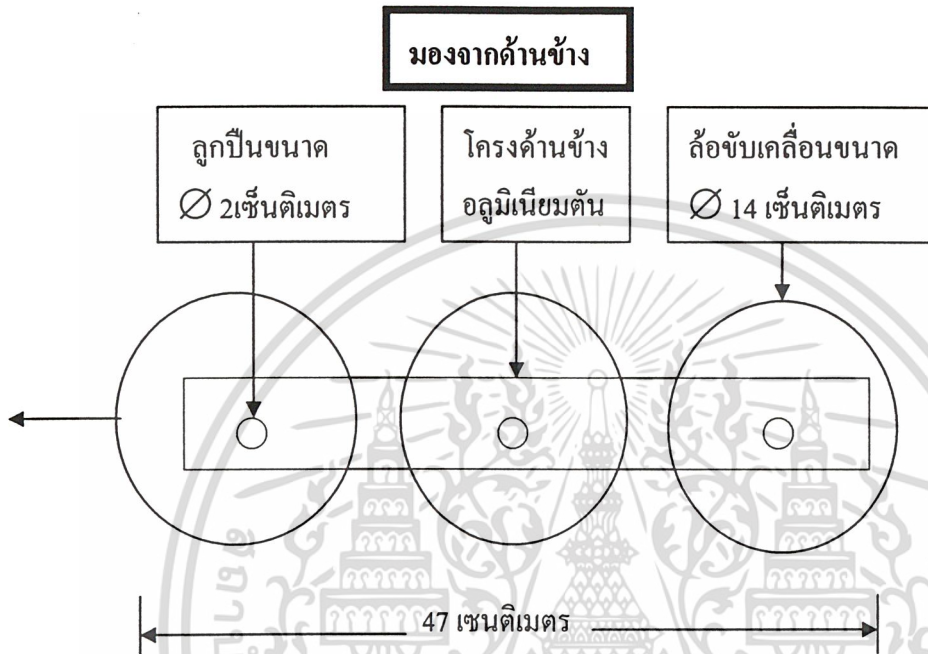


รูปที่ 3.2 แสดงโครงสร้างการทำงานโดยละเอียดของหุ่นยนต์

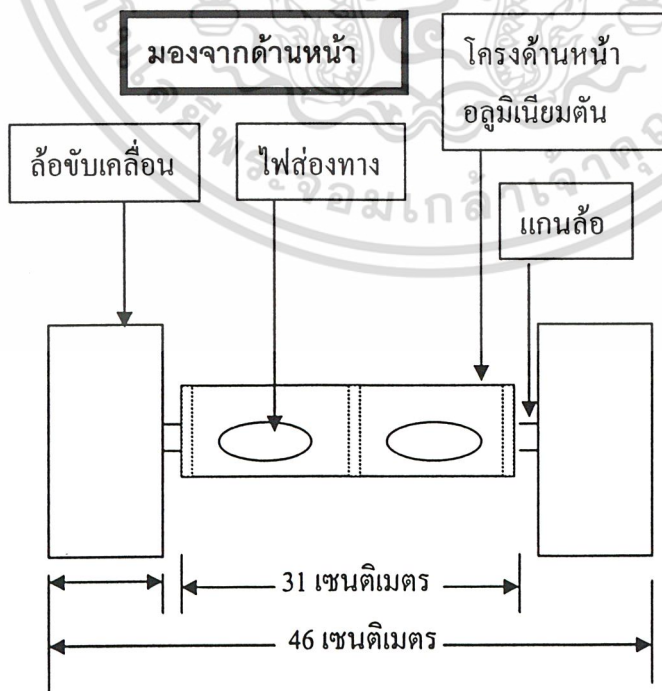
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบทางด้านวงจรควบคุมและโครงสร้างหุ่นยนต์ (Hardware)

3.2.1 โครงสร้างตัวถังของหุ่นยนต์

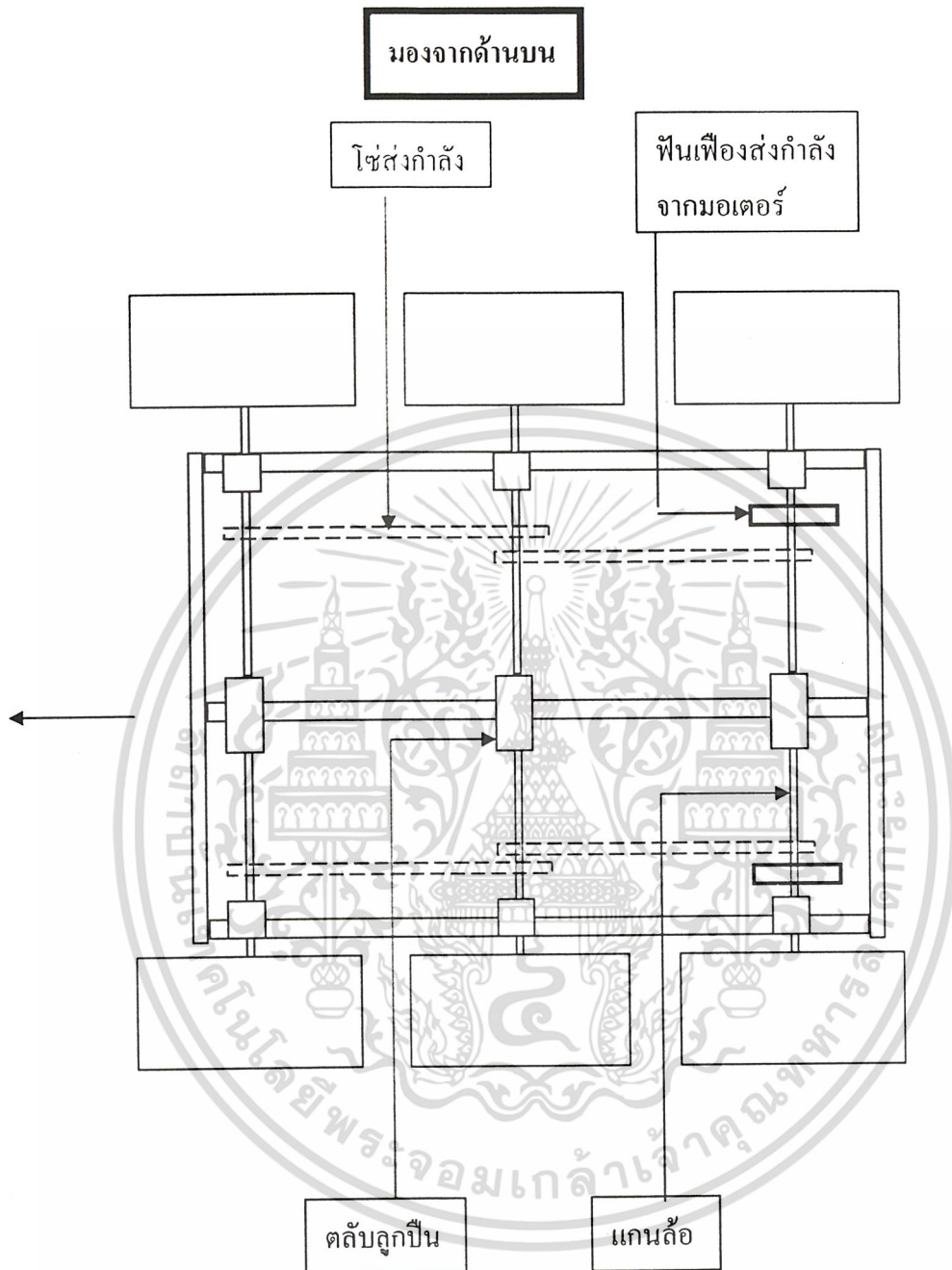


รูปที่ 3.3 แสดงโครงสร้างของหุ่นยนต์มองจากด้านข้าง



รูปที่ 3.4 แสดงโครงสร้างของหุ่นยนต์มองจากด้านหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



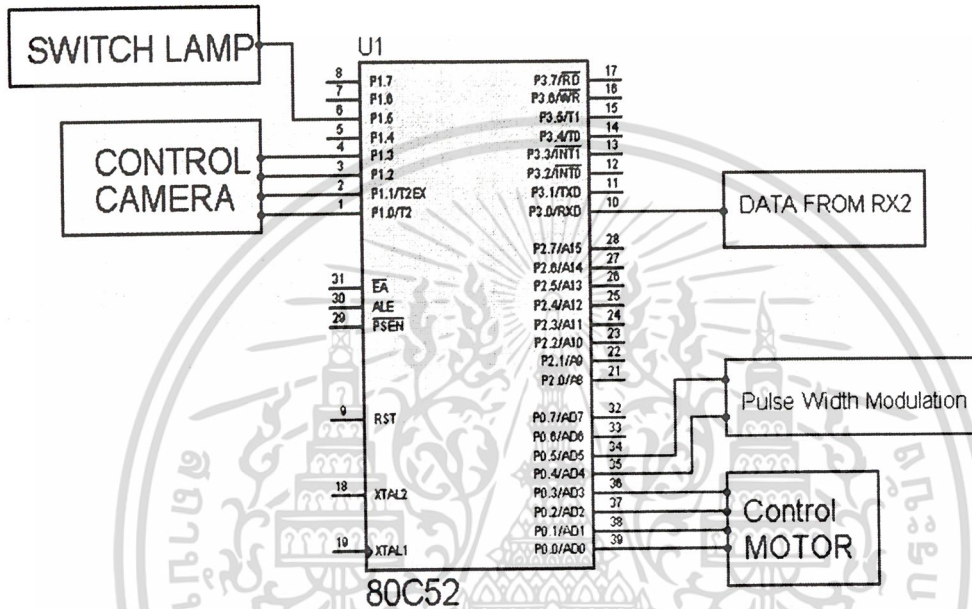
รูปที่ 3.5 แสดงโครงสร้างของหุ่นยนต์มองจากด้านบน

โครงสร้างหลักของหุ่นยนต์ทำจากอลูมิเนียมตันหนา 0.6 เซนติเมตรเพื่อความแข็งแรง โดยหุ่นยนต์จะมีขนาดประมาณ 47x44 cm. ใช้ล้อในการขับเคลื่อนจำนวน 6 ล้อ ซึ่งเรียงกันข้างละ 3 ล้อเพื่อลดช่วงความห่างทำให้สามารถผ่านพื้นผิวที่เป็นขั้นได้ โดยล้อมีขนาดเส้นผ่านศูนย์กลาง 14 เซนติเมตร กว้าง 7 เซนติเมตร ใช้ออเตอร์ข้างละ 1 ตัวส่งกำลังผ่านโซ่และเฟืองขนาดเล็กขั้วทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6 ล้อ ในการเลี้ยวจะใช้การหยุดล้อไว้ 1 ข้าง หรือหมุนล้อข้างซ้ายและขวากลับกันทำให้ช่วยลดพื้นที่ของการเลี้ยว

3.2.2 วงจรการเชื่อมต่อของ MCS-51



รูปที่ 3.6 แสดงการต่ออุปกรณ์ภายนอกกับไมโครคอนโทรลเลอร์

จากรูปที่ 3.4 จะเป็นการบอกตำแหน่งของขาแต่ละขาของพอร์ตที่จะใช้ควบคุม หรือใช้รับข้อมูลภายนอก โดยสามารถอธิบายได้เป็นส่วนๆ ดังนี้

3.3.1 ส่วนควบคุมการเคลื่อนที่ของหุ่นจะใช้พอร์ต 0.0 ถึง 0.3 โดยจะแยกควบคุมพอร์ต 0.0 และ 0.1 จะควบคุมมอเตอร์ซ้ายและพอร์ตที่ 0.2 ถึง 0.3 จะใช้ควบคุมมอเตอร์ขวา

3.3.2 ส่วนควบคุมความเร็วในการเคลื่อนที่ของหุ่นจะใช้พัลส์วิดท์มอดดูเลชัน (Pulse Width Modulation) ในการควบคุม โดยจะแบ่งได้ 2 ชุดคือ พอร์ต 0.4 จะควบคุมความเร็วมอเตอร์ซ้ายพอร์ต 0.5 จะควบคุมความเร็วมอเตอร์ขวา

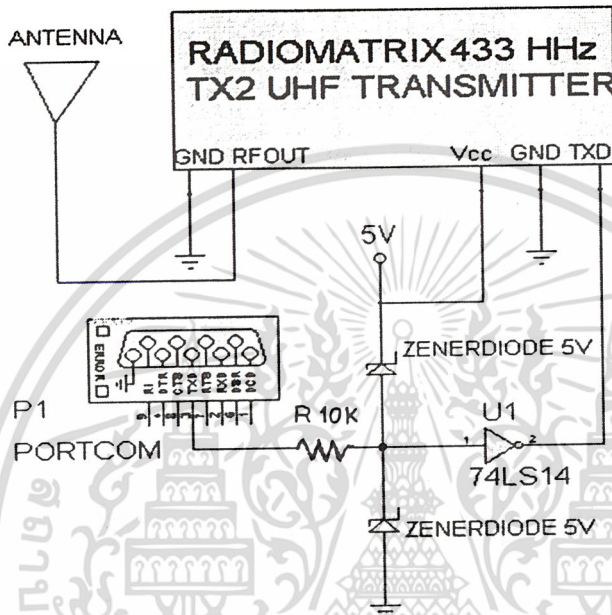
3.3.3 ส่วนควบคุมการเคลื่อนที่ของกล้องวิดีโอ จะใช้พอร์ต 1.0 และ 1.1 ในการควบคุมการหมุนขึ้น-ลงของกล้อง และจะใช้พอร์ต 1.2 และ 1.3 ในการควบคุมการหมุนซ้าย-ขวาของกล้อง

3.3.4 ส่วนควบคุมการเปิดปิดไฟส่องทาง จะใช้พอร์ต 1.5 ในการควบคุม

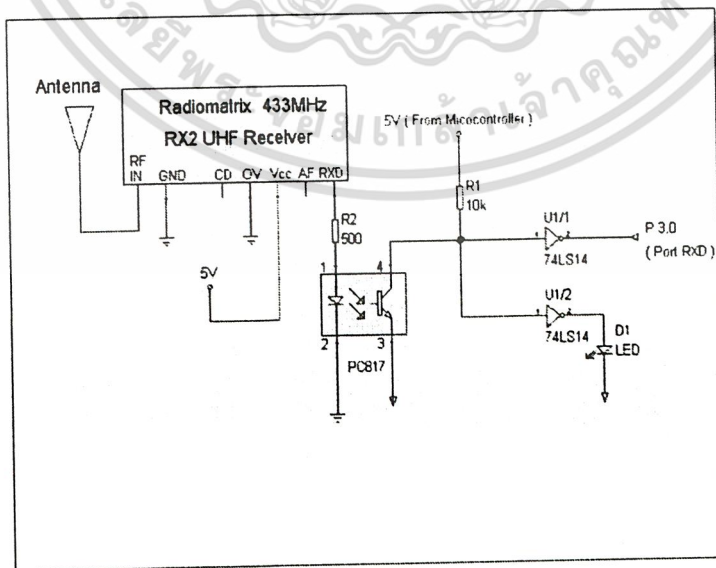
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.5 ส่วนรับสัญญาณดิจิทัลจากคอมพิวเตอร์ จะใช้ระบบส่งข้อมูลไร้สายผ่านโมดูลเรดิโอ เมทริก (Radio Matrix) ส่งเข้าไปที่ขา RXD หรือ พอร์ต3.0

3.2.3 วงจรควบคุมการสื่อสารระหว่างคอมพิวเตอร์กับหุ่นยนต์



รูปที่ 3.7 แสดงวงจรภาคส่งสัญญาณจากคอมพิวเตอร์ไปยังโมดูล



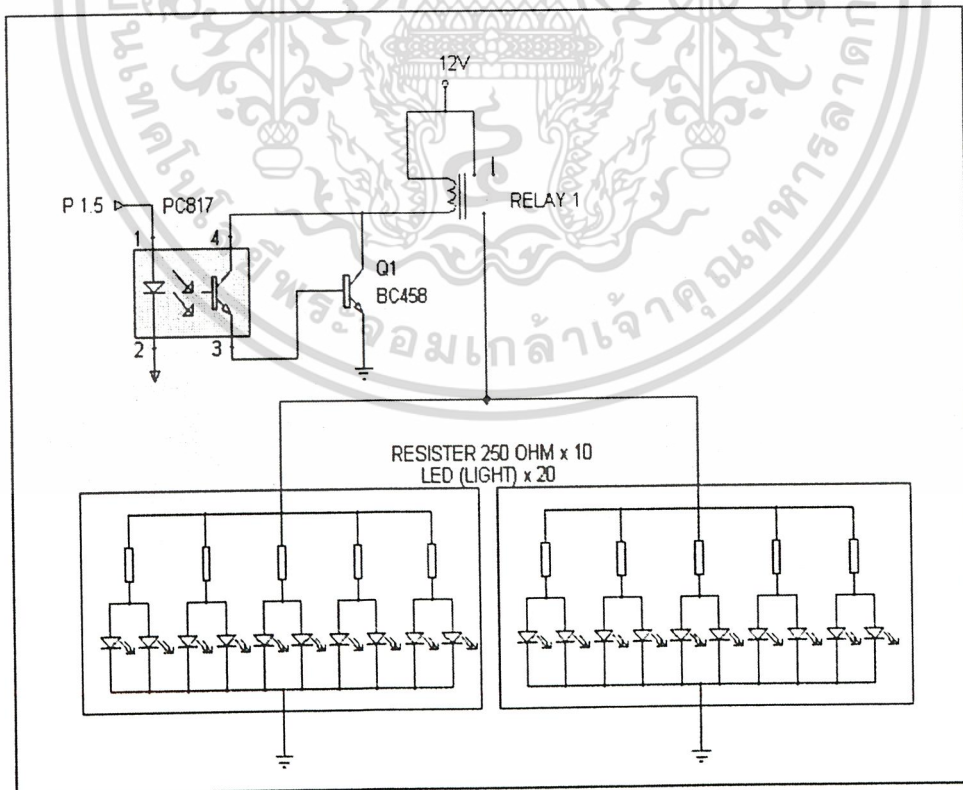
รูปที่ 3.8 แสดงวงจรภาครับสัญญาณจากโมดูลไปยังไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับข้อมูลจากคอมพิวเตอร์นั้นจะใช้เรดิโอเมทริก ในการส่งข้อมูลไร้สายไปยังหุ่นยนต์เพื่อรับคำสั่ง จากวงจรภาคส่งจะเห็นว่าเมื่อคอมพิวเตอร์ส่งข้อมูลออกมายังพอร์ตขนานขา 3 เป็นสัญญาณไฟฟ้าที่มีแรงดัน +12 และ -12 โวลต์ ดังนั้นจึงแปลงสัญญาณเป็น 5 โวลต์ โดยก่อนจะเข้านอตเกตนั้นต้องมีการลดขนาดของสัญญาณก่อนโดยใช้ซีเนอร์ไดโอด 5 โวลต์ และตัวต้านทาน จากนั้นสัญญาณที่ออกมาจะส่งเข้าไปที่ขา 5 ของเรดิโอเมทริกตัวส่ง (Tx2) จากนั้นจะมอดูเลทกับความถี่ของสัญญาณที่ความถี่ 433 เมกกะเฮิร์ตส่งผ่านสายอากาศไปยังตัวรับของภาครับสัญญาณ

ภาครับสัญญาณจะมีการรับสัญญาณที่เข้ามาโดยผ่านสายอากาศเข้าที่ขา 1 ของเรดิโอเมทริกตัวรับ (Rx2) โดยจะแยกความถี่ 433 เมกกะเฮิร์ต ออกมาเป็นสัญญาณดิจิทัลออกมาที่ขา 7 ของ เรดิโอเมทริก จากนั้นจะส่งสัญญาณไปยังออปโตไดโอด ซึ่งจะทำการส่งสัญญาณแสงผ่านไปยังทรานซิสเตอร์และรับข้อมูล จากนั้นจะผ่านนอตเกตเพื่อกลับสัญญาณส่งไปยังขารับข้อมูลอนุกรมของ ไมโครคอนโทรเลอร์ ส่วนของแอลอีดี (LED) จะเป็นตัวบอกสถานะของการส่งข้อมูล

3.2.4 วงจรควบคุมอุปกรณ์ต่างๆ

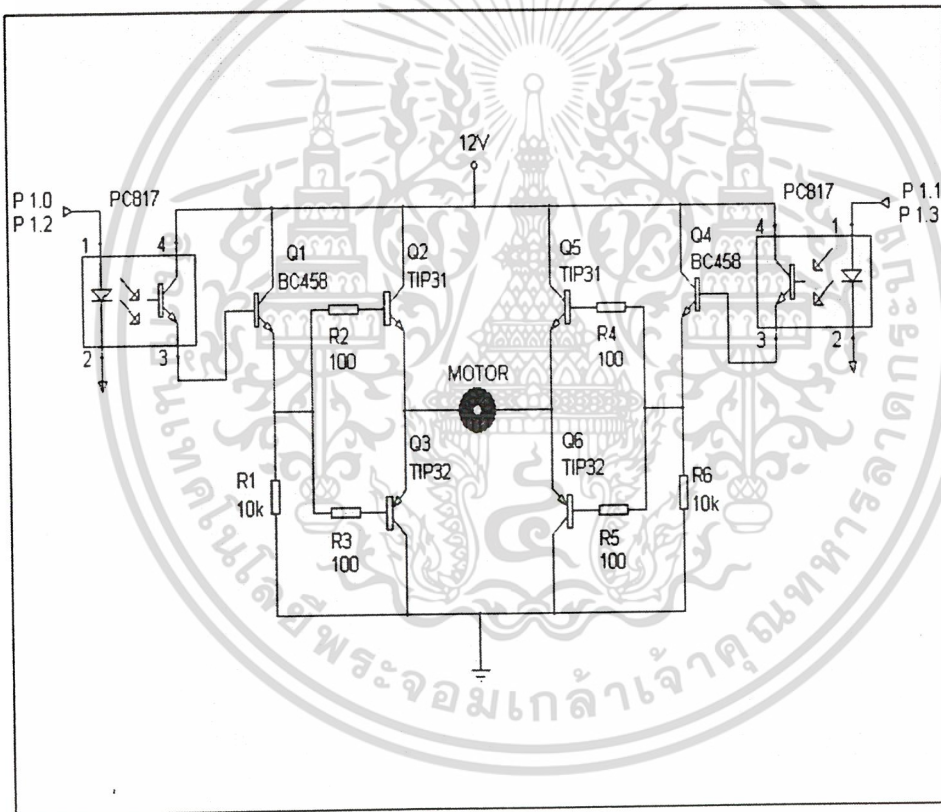


รูปที่ 3.9 แสดงวงจรเปิด-ปิดไฟส่องทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขณะปกติพอร์ต 1.5 จะมีสถานะเป็นศูนย์จะไม่มีการไหลของกระแสในวงจร เนื่องจากทรานซิสเตอร์ คัทออฟ เมื่อมีการกดสวิตช์จากปุ่มควบคุมแล้วจะมีการส่งสัญญาณมาที่ตัวหุ่น ทำให้ไมโครคอนโทรลเลอร์เซตพอร์ต 1.5 หรือทำให้พอร์ต 1.5 มีลอจิก1ค้าง มีผลทำให้แอลอีดี(LED) ภายในออปโตคัปเปิ้ล (Optocoupler) สว่างขึ้นทำให้ทรานซิสเตอร์ภายในนำกระแสจากแบตเตอรี่ 12 โวลต์ ไปไบอัสให้กับทรานซิสเตอร์ (Q1) ทำให้ทรานซิสเตอร์ (Q1) นำกระแสรีเลย์มีกระแสไหลผ่านขดลวดเหนี่ยวนำรีเลย์จึงเปลี่ยนหน้าสัมผัสมาที่ไฟ 12 โวลต์ จึงทำให้หลอดแอลอีดีติด จนกว่าจะมีการกดปุ่มควบคุมปิดแอลอีดีจะดับ

3.2.5 วงจรควบคุมฐานของกล้อง



รูปที่ 3.10 แสดงวงจรควบคุมฐานของกล้อง

จากวงจรเมื่อปกติทรานซิสเตอร์1(Q1) และทรานซิสเตอร์4(Q4)จะคัทออฟ ทำให้ทรานซิสเตอร์1 และ5 (Q1,Q5) คัทออฟ และทรานซิสเตอร์ 3 กับ 6 (Q3,Q6) จะทำงาน (on) แต่ไม่มีกระแสไหลในวงจร เมื่อมีสัญญาณไฮท์ (High) พอร์ตโคพอร์หนึ่งเช่นพอร์ต 1.0 จะทำให้ทรานซิสเตอร์ 1 ทำงาน (Q1,on) ทำให้ทรานซิสเตอร์ 2 ทำงาน (Q2,on) แต่ทรานซิสเตอร์3หยุดทำงาน (Q3,Cutoff) และทรานซิสเตอร์ 5 และ 6 (Q5,Q6) ยังคงสถานะเดิมจึงทำให้มีกระแสไหล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

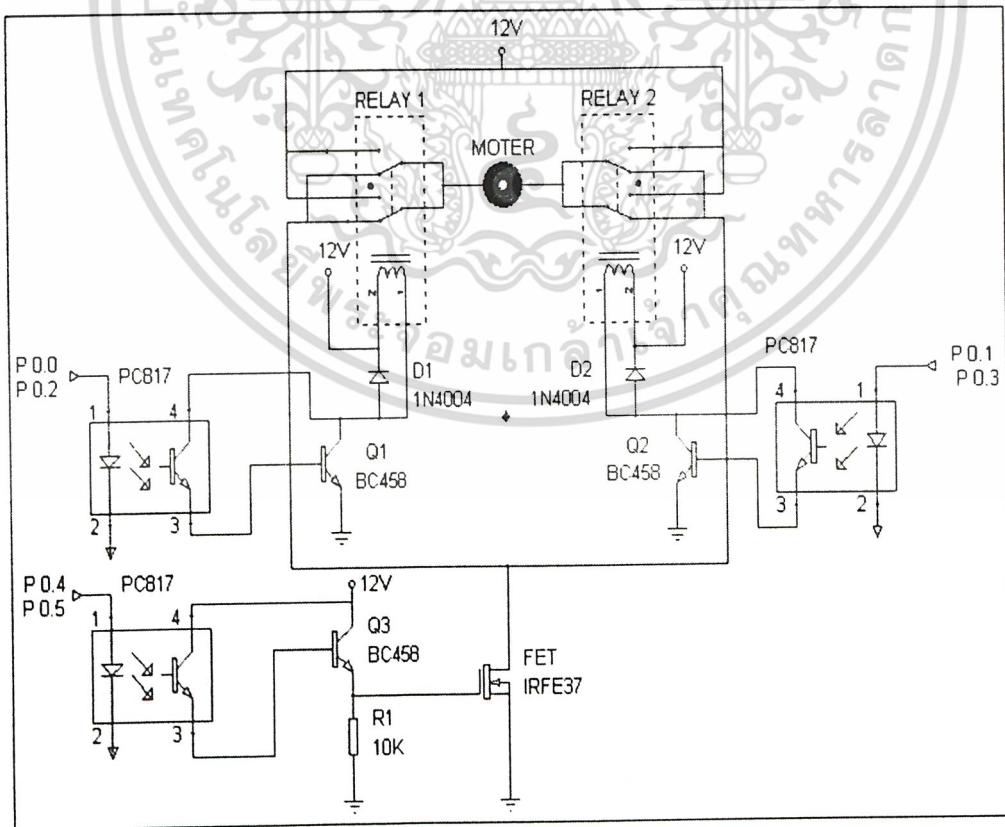
จากแบตเตอรี่ 12 โวลต์ ผ่านทรานซิสเตอร์ 2 (Q2) ไปยังมอเตอร์และไปที่ทรานซิสเตอร์ 3 (Q3) ลงกราวด์ทำให้มอเตอร์หมุน

จากวงจรจะใช้ในหุ่นยนต์ 2 ชุดคือหมุนกลิ้งขึ้นลงและหมุนกลิ้งซ้ายขวา สามารถควบคุมสรุปได้ในตาราง

สถานะพอร์ต		สถานะมอเตอร์	ทรานซิสเตอร์					
1.0,1.2	1.1,1.3		1	2	3	4	5	6
0	0	ไม่หมุน	0	0	1	0	0	1
0	1	หมุนซ้าย	0	0	1	1	1	0
1	0	หมุนขวา	1	1	0	0	0	1
1	1	ไม่หมุน	1	1	0	1	1	0

ตารางที่ 3.1 แสดงการทำงานของวงจรควบคุมฐานของกลิ้ง

3.2.6 วงจรควบคุมทิศทางมอเตอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 3.11 แสดงวงจรควบคุมทิศทางมอเตอร์ มีอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

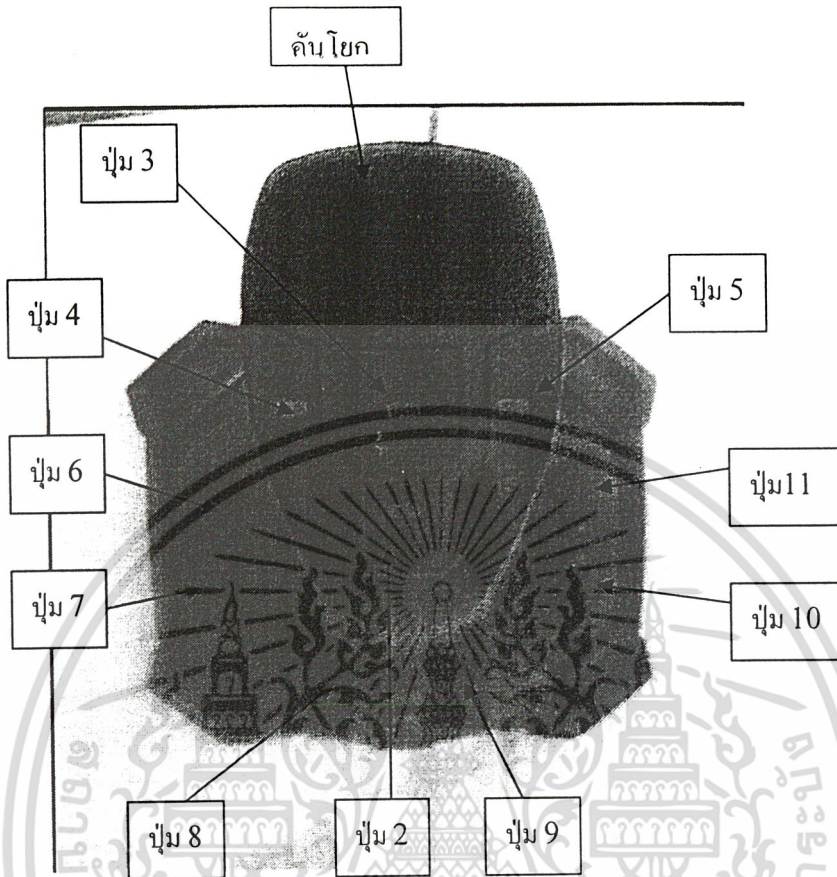
จากวงจรเมื่อสถานะปกติมอเตอร์จะไม่หมุนเนื่องจากรีเลย์ยังไม่ทำงานขาทั้ง 2 ข้างของมอเตอร์ช็อคกันอยู่ช่วยในการห้ามหมุนของล้อได้ เมื่อมีสัญญาณพอร์ตโคพอร์ตหนึ่งเช่นพอร์ต 0.0 จะทำให้โอปโตคัมเบิลทำงานทรานซิสเตอร์ภายในจะนำกระแสจะทำให้ทรานซิสเตอร์ 1(Q1) นำกระแสชดทวดภายในรีเลย์ 1 (Relay1) นำกระแสเหนี่ยวนำสวิตช์เปลี่ยนหน้าคอนแทกมาที่ไฟ 12 โวลท์ มีผลทำให้มอเตอร์ต่อกับไฟ 12 โวลท์และจะควบคุมความเร็วจากพอร์ต 0.4 และ 0.5 โดยใช้พัลส์วีดมอด-คูลเทชันในการควบคุมความเร็วโดยใช้เฟสในการขยายสัญญาณ การทำงานในสถานะอื่นๆสามารถอธิบายได้ในตาราง

สถานะพอร์ต		พัลส์วีดมอดคูลชัน	ทรานซิสเตอร์		สถานะมอเตอร์
0.0,0.2	0.1,0.3	0.4,0.5	1	2	
0	0	มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	ไม่หมุน
		ไม่มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	ไม่หมุน
0	1	มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	หมุนช้า
		ไม่มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	ไม่หมุน
1	0	มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	หมุนขวา
		ไม่มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	ไม่หมุน
1	1	มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	ไม่หมุน
		ไม่มีสัญญาณ	ไม่ทำงาน	ไม่ทำงาน	ไม่หมุน

ตารางที่ 3.2 แสดงการทำงานของวงจรควบคุมการเคลื่อนที่ของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.7 จอยสติค



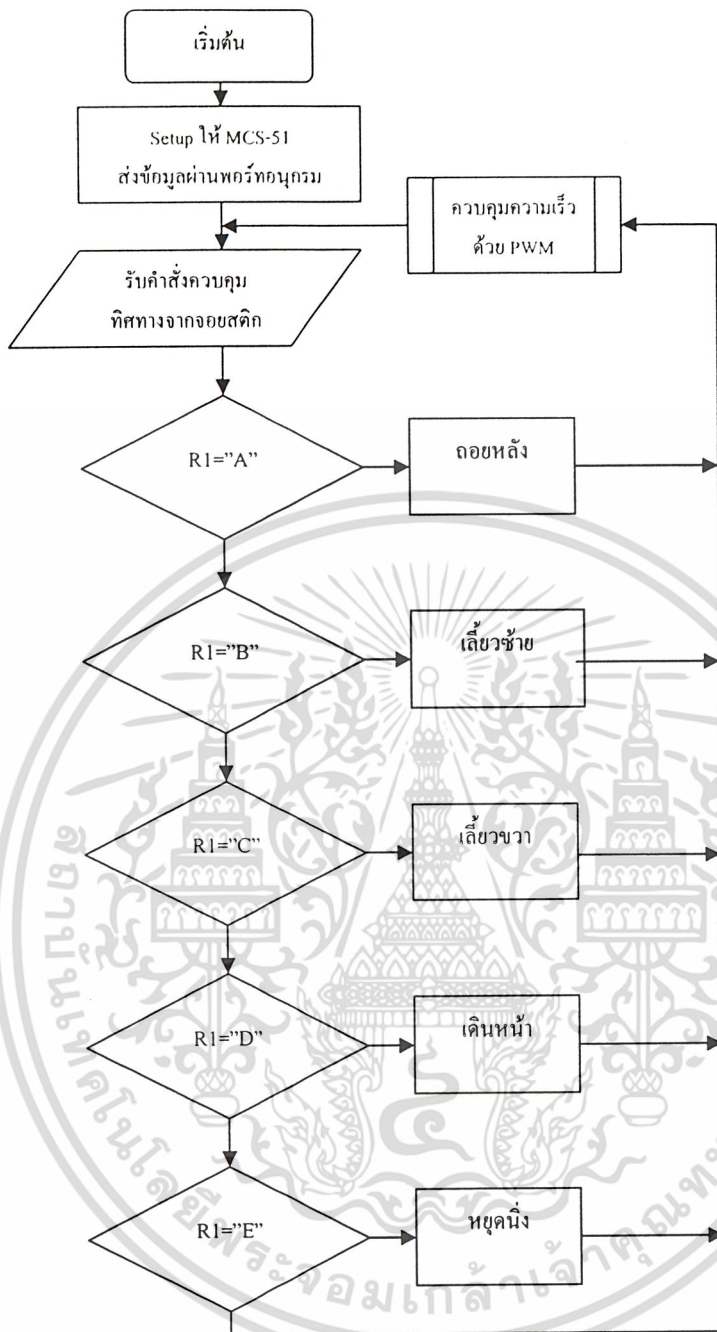
รูปที่ 3.12 แสดงตำแหน่งของปุ่มต่างๆ บนจอยสติค

3.3 การออกแบบทางด้านโปรแกรม (Software)

3.3.1 ส่วนควบคุมการเคลื่อนที่ของหุ่นยนต์

ในส่วนของการเคลื่อนที่จะประกอบไปด้วย ส่วนที่เป็นตัวบังคับทิศทางของหุ่นยนต์ (Joystick) ส่วนของการรับ-ส่งข้อมูล และมอเตอร์ซึ่งมีจำนวน 2 ตัว โดยจะควบคุมมอเตอร์ผ่านวงจร ชูคัซบเคลื่อนคังที่ได้แสดงไว้ในข้างต้น โดยจะมีหลักการการออกแบบดังรูปที่

3.7



รูปที่ 3.13 โพล์ชาร์ตแสดงการควบคุมทิศทางการเคลื่อนที่ของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ส่วนควบคุมการส่งข้อมูลผ่านพอร์ตอนุกรม

ในการออกแบบการส่งข้อมูลผ่านพอร์ตอนุกรมโดยใช้โปรแกรม Visual Basic สามารถทำได้โดยการกำหนดเงื่อนไขต่างๆ ดังนี้

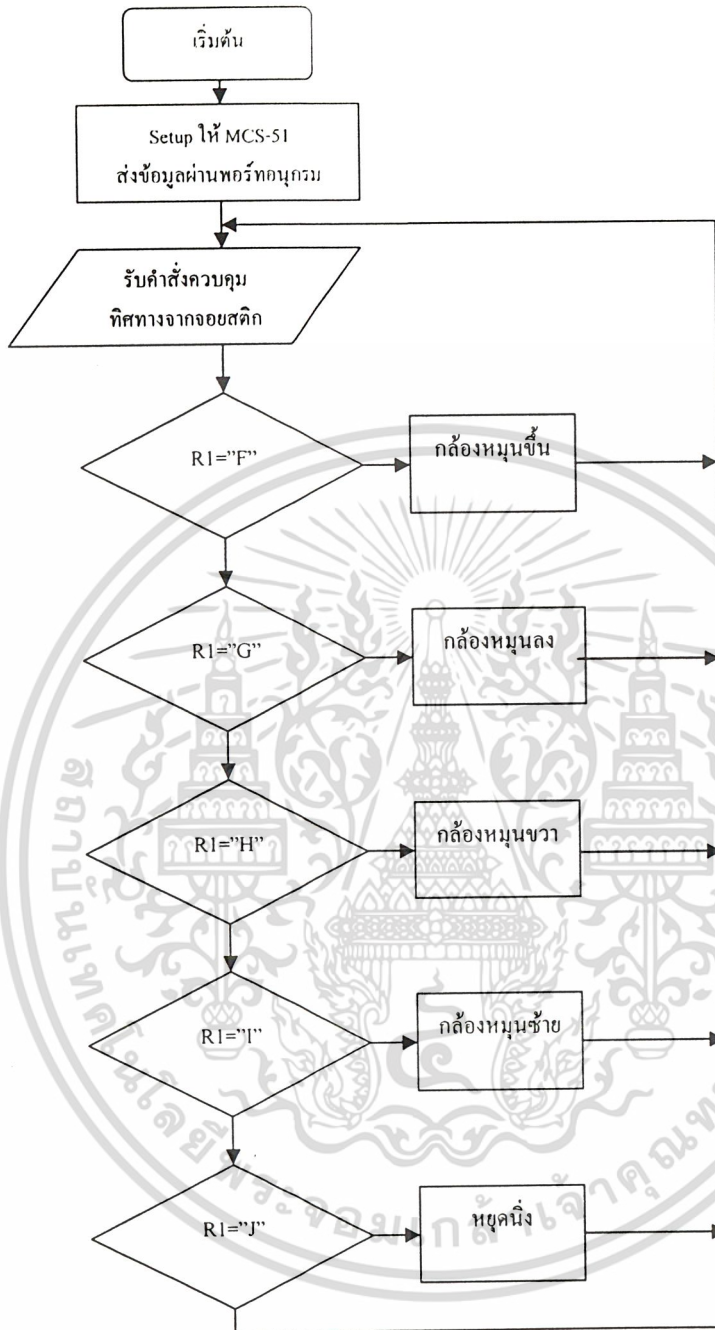
```
MSComm1.Settings = "9600,N,8,1"
MSComm1.CommPort = 1
MSComm1.InputLen = 1
MSComm1.PortOpen = True
MSComm1.RThreshold = 1
```

จากโปรแกรมจะเป็นการกำหนดอัตราการรับส่งข้อมูลทางพอร์ตอนุกรม หรือ เรียกว่า บอเดอเรต (Baud Rates) โดยมีอัตราการส่งคือ 9600 บิต/วินาที (bps: bit per second) ซึ่งมีขนาดของข้อมูลขนาด 8 บิต และมีการส่งบิตพาริตี (Parity Bit) เพื่อใช้ในการตรวจสอบข้อมูล ส่วนพอร์ตอนุกรมที่ใช้จะเป็นพอร์ต COM. 1 ซึ่งในการกำหนดเงื่อนไขต่างๆ ดังข้างต้นนี้ โปรแกรมจะเรียกใช้ฟังชันที่เป็นฟังชันของวินโดวส์ ก็คือ ส่วนของ MSComm 1. นั่นเอง และเมื่อข้อมูลถูกส่งมาที่พอร์ตอนุกรมแล้ว จากนั้นก็จะส่งข้อมูลไปยังหุ่นยนต์ โดยผ่านโมดูล Radio Matrix (Tx) ซึ่งทำหน้าที่ส่งข้อมูลเพียงอย่างเดียว

3.3.3 ส่วนควบคุมฐานกล้อง

ในส่วนของการควบคุมฐานกล้องจะประกอบด้วย จะมีลักษณะควบคุมที่คล้ายกับชุดควบคุมการเคลื่อนที่ แต่จะแตกต่างกันที่คำสั่งควบคุมเท่านั้น ซึ่งมีหลักการออกแบบดังรูปที่

3.8



รูปที่ 3.14 โพลีชาร์ตแสดงการควบคุมฐานกลิ้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

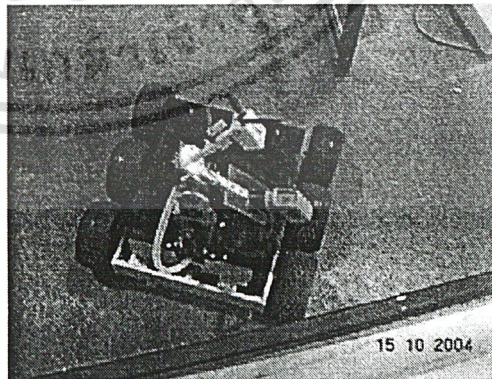
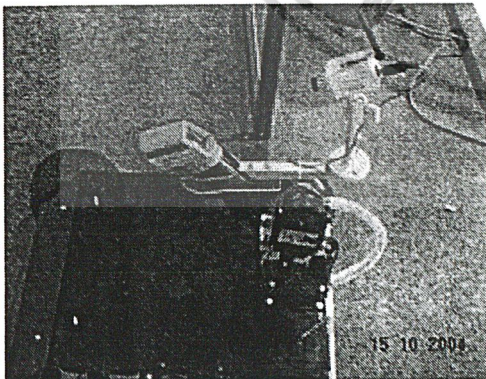
บทที่ 4 ผลการทดสอบ

4.1 การเคลื่อนที่ของหุ่นยนต์

การบังคับการเคลื่อนที่จะใช้การโยกจอยสติคในการกำหนดทิศทาง หรือจะใช้ปุ่มบนคันโยกจอยสติค กำหนดทิศทางการเคลื่อนที่ (ปุ่มที่ 2, 3, 4, 5) จากการควบคุมจะพบว่าหุ่นสามารถเคลื่อนที่ได้ 360 องศารอบทิศทางโดยการหมุนรอบตัวเองอาศัยหลักการหมุนล้อทั้งสองข้างสลับกัน เมื่อจะทำการเลี้ยวล้อข้างใดข้างหนึ่งจะหยุดส่วนล้ออีกด้านจะหมุน หรือล้อทั้งสองข้างหมุนไม่เท่ากันจึงจะเลี้ยวได้ ส่วนความเร็วนั้นถ้าใช้คันโยกควบคุมความเร็วจะขึ้นอยู่กับความเร็วการเดินคันโยกว่าเบาหรือแรงถ้าเบาๆ ก็จะทำให้รถเคลื่อนที่ช้าแต่ถ้าแรงจะทำให้รถเคลื่อนที่เร็วส่วนการควบคุม โดยใช้ปุ่มที่ 2, 3, 4, 5 จะทำให้มีความเร็วที่เท่ากันเป็นการควบคุมความเร็วคงที่ ในส่วนทิศทางการเดินทางนั้นคือปุ่มที่ 2 ถอยหลัง ปุ่มที่ 3 เดินหน้า ปุ่มที่ 4 หมุนซ้ายและปุ่มที่ 5 หมุนขวา จากการควบคุมโดยใช้ความเร็วสูงสุดจะเคลื่อนที่ในระยะ 100 เมตร ประมาณ 5 วินาที และสามารถขึ้นเนินที่มีความลาดชันประมาณ 30 องศาได้

4.2 การบังคับทิศทางของฐานกล้อง

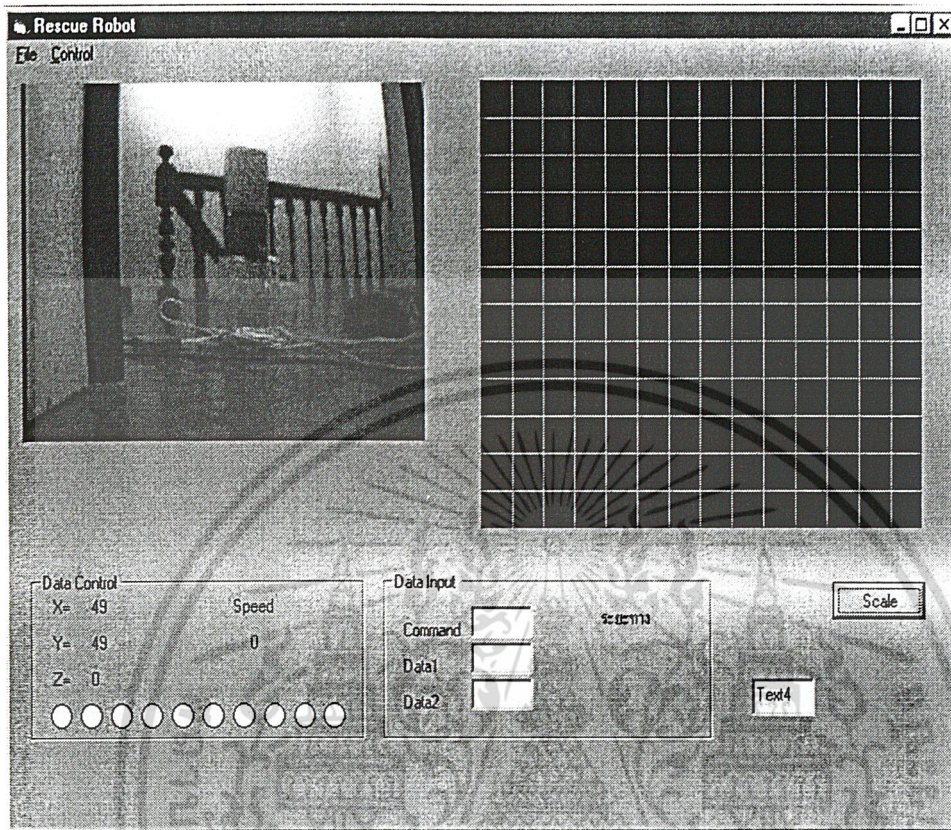
การควบคุมการหมุนของกล้องนั้นจะต้องมีการกดปุ่มที่ 6 ค้างไว้ และทำการกำหนดทิศทางหมุนขึ้นจากปุ่มที่ 3 ทิศลงจากปุ่มที่ 2 ทิศซ้ายจากปุ่มที่ 4 ทิศขวาจากปุ่มที่ 5 โดยการหมุนซ้ายขวานั้นสามารถหมุนได้รอบทิศทางหรือ 360 องศา แต่การหมุนขึ้นลงจะทำได้ประมาณ 300 องศา โดยที่ความละเอียดในการหมุนจะประมาณสละปละ 2 องศาแต่เมื่อปล่อยปุ่มที่ 6 แล้วจะไม่สามารถควบคุมกล้องได้



รูปที่ 4.1 แสดงการบังคับทิศทางของฐานกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

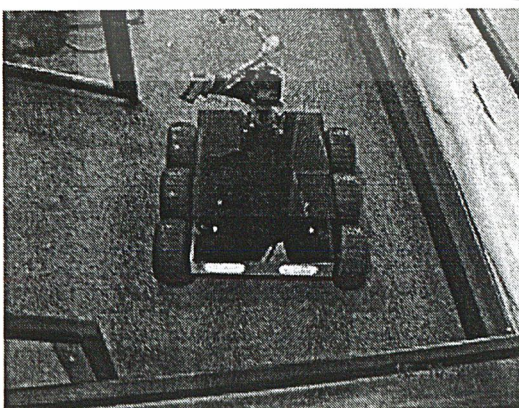
4.3 การแสดงภาพบนหน้าจอกอมพิวเตอร์



รูปที่ 4.2 แสดงภาพหน้าต่างของโปรแกรม

4.4 การใช้อุปกรณ์ต่างๆ

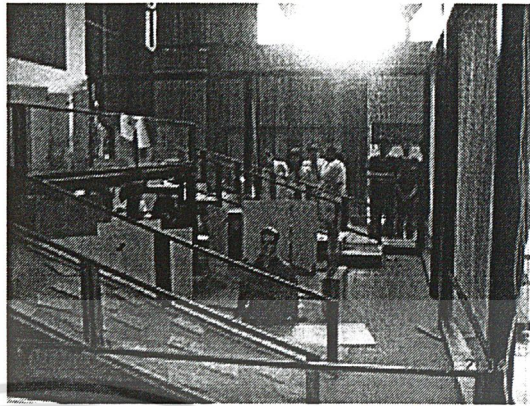
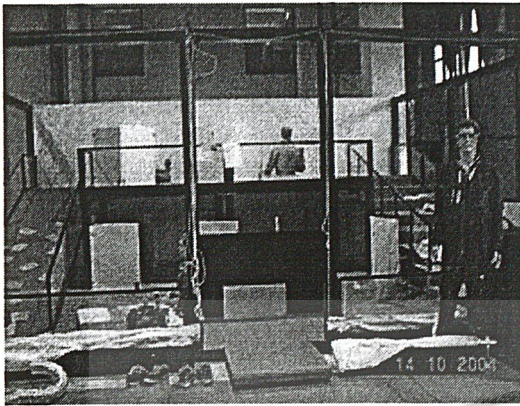
การเปิดปิดไฟส่องทางนั้นสามารถทำได้โดยการกดปุ่มที่ 8 ในการเปิดและกดปุ่มที่ 9 เพื่อทำการปิดมุมการส่องแสงข้างหน้ารถจะมีประมาณ 100 องศา บริเวณที่สว่างที่สุดเป็นบริเวณหน้ารถ



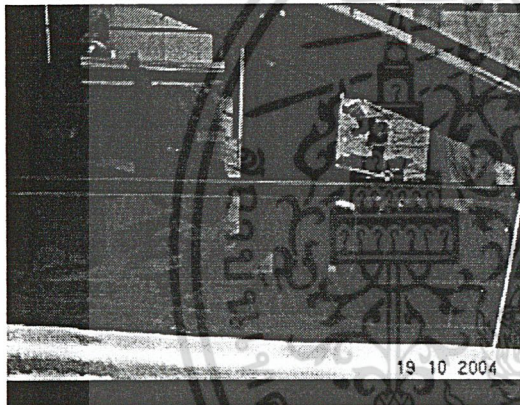
รูปที่ 4.3 แสดงการใช้ไฟส่องทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

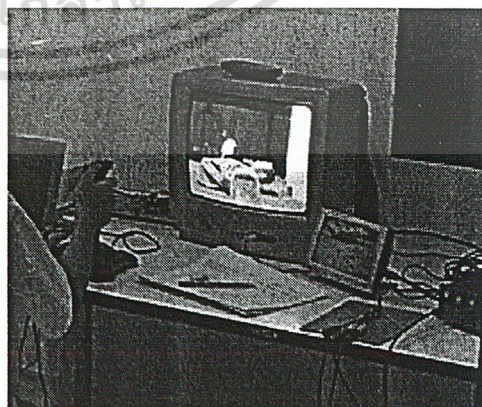
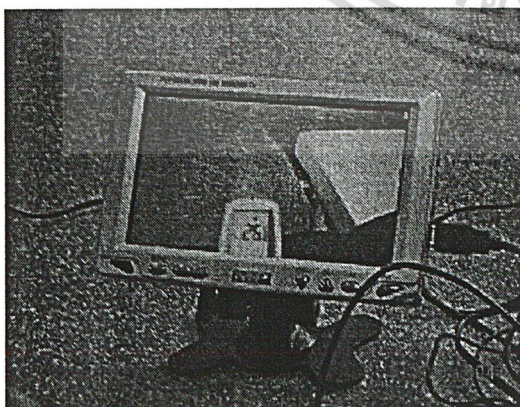
4.5 การทดลองในสถานที่ที่จำลองจากเหตุการณ์จริง



รูปที่ 4.4 แสดงสภาพของสนามที่จำลองจากเหตุการณ์จริง

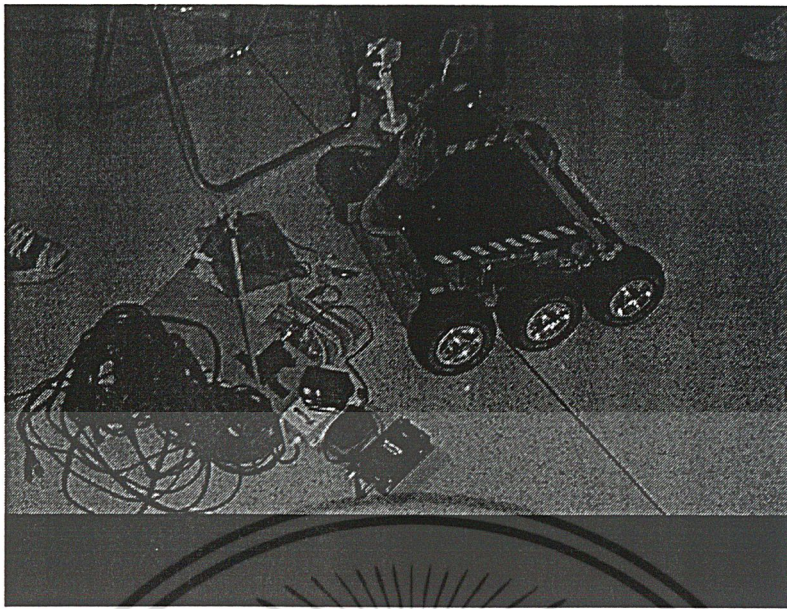


รูปที่ 4.5 แสดงการค้นหาเหยื่อผู้เคราะห์ร้าย



รูปที่ 4.6 แสดงภาพหน้าจอของมอนิเตอร์ต่างๆ ขณะที่หุ่นยนต์กำลังทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

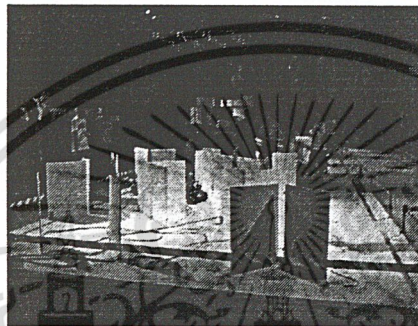
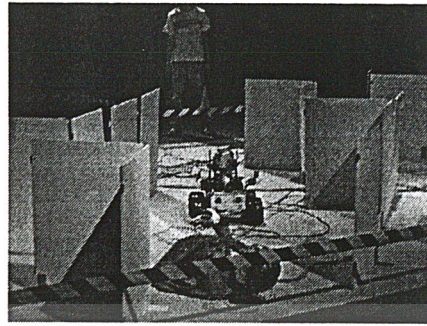
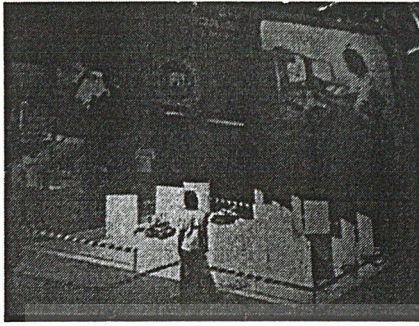


รูปที่ 4.7 แสดงอุปกรณ์ทั้งหมดที่ใช้ในการปฏิบัติการกิจกู้ภัย

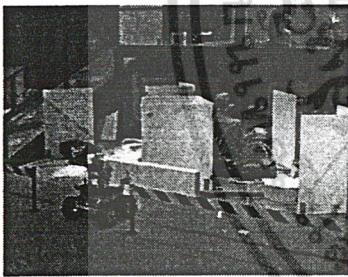


รูปที่ 4.8 แสดงการวัดอุณหภูมิจากผู้เคราะห์ร้าย
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

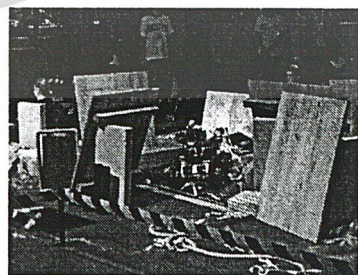
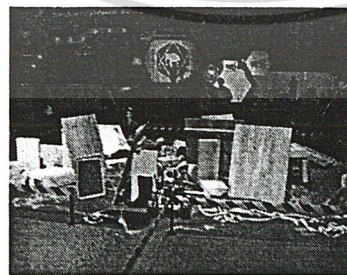
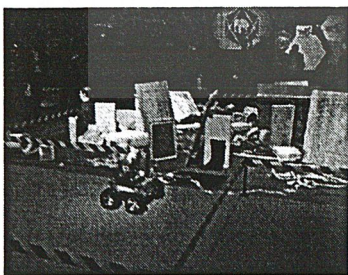
4.6 สนามที่ใช้แข่งขันในรอบสุดท้าย



รูปที่ 4.9 แสดงสภาพสนามระดับความยากน้อยที่สุด



รูปที่ 4.10 แสดงสภาพสนามระดับความยากปานกลาง



รูปที่ 4.11 แสดงสภาพสนามระดับความยากมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 สรุปและวิจารณ์

โครงการนี้จัดทำขึ้นมาเพื่อเป็นแนวทางในการสร้างหุ่นยนต์ที่ใช้ในการช่วยเหลือหรือการกู้ภัยในบริเวณที่เป็นอันตรายต่อมนุษย์ เช่น บริเวณที่มีแก๊สพิษรั่วไหล หรือภายในอาคารที่เสี่ยงต่อการเกิดอาคารถล่ม เป็นต้น หุ่นยนต์ตัวนี้ใช้กล้อง CCD แบบไร้สายในการมองเห็นระยะวิสัยของบริเวณที่ต้องการตรวจสอบ อีกทั้งยังสามารถตรวจสอบเสียงและความร้อนจากวัตถุหรือเหยื่อผู้เคราะห์ร้าย เพื่อจะบอกได้ว่าสภาพของเหยื่อผู้เคราะห์ร้ายอยู่ในสภาพใด

จากการนำหุ่นยนต์ไปทดสอบในสถานที่ที่จำลองจากเหตุการณ์อาคารถล่ม พบว่าหุ่นยนต์ตัวนี้สามารถใช้งานได้อย่างมีประสิทธิภาพ โดยระบุได้ว่าสภาพของเหยื่อผู้เคราะห์ร้ายมีสภาพเป็นเช่นไร อยู่ตำแหน่งใดของสถานที่จำลอง และบอกสภาพแวดล้อมของสถานที่นั้นว่าเป็นเช่นไร เพื่อให้หน่วยกู้ภัยที่เข้าไปช่วยเหลือผู้เคราะห์ร้ายในภายหลัง สามารถเข้าไปช่วยเหลือได้อย่างรวดเร็วและมีประสิทธิภาพ

ข้อจำกัดของหุ่นยนต์ตัวนี้ คือ ไม่สามารถเข้าไปตรวจสอบเหยื่อผู้เคราะห์ร้ายในสถานที่ที่มีความร้อนสูง เช่น อาคารที่กำลังเกิดอัคคีภัย เพราะใช้อุปกรณ์และโครงสร้างของหุ่นยนต์ไม่สามารถทนความร้อนได้ และในการระบุตำแหน่งของเหยื่อผู้เคราะห์ยังไม่มีความมีประสิทธิภาพเท่าที่ควร เพราะหุ่นยนต์ตัวนี้ใช้แสงเลเซอร์ในการกำหนดตำแหน่ง เมื่อหุ่นยนต์อยู่ในจุดที่ลาดเอียงหรือไม่สมดุล จะทำให้การบอกตำแหน่งมีการคลาดเคลื่อน ดังนั้นหุ่นยนต์ตัวนี้เมื่อนำไปพัฒนาแก้ไขข้อจำกัดต่างๆ จะทำให้หุ่นยนต์ตัวนี้มีประสิทธิภาพมากขึ้นและเป็นประโยชน์ต่อสังคมต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROGRAM MAIN CONTROL BY VB. 6

```
.....
****
Option Explicit
Dim lwndC As Long
Dim INFrames As Long
Private Declare Function joyGetPos Lib "winmm.dll" (ByVal uJoyID As Long, pji As JOYINFO) As Long
```

```
Private Type JOYINFO
    wXpos As Long
    wYpos As Long
    wZpos As Long
    wButtons As Long
End Type
```

```
Dim my_joy As JOYINFO
Dim joy_ox As Integer
Dim joy_oy As Integer
Dim joy_oz As Integer
Dim joy_x As Integer
Dim joy_y As Integer
Dim joy_z As Integer
Dim output1 As Byte
Dim output2 As Byte
```

```
Dim rs_index As Integer
Dim command_in As Variant
Dim data1 As Variant
Dim data2 As Variant
Dim dist1 As Integer
Dim dist2 As Integer
Dim pre_data As Variant
```

```
Sub ResizeCaptureWindow(ByVal lwnd As Long)
    Dim CAPSTATUS As CAPSTATUS
    capGetStatus lwnd, VarPtr(CAPSTATUS), Len(CAPSTATUS)
    SetWindowPos lwnd, HWND_BOTTOM, 0, 0, _
        CAPSTATUS.uiImageWidth, _
        CAPSTATUS.uiImageHeight, _
        SWP_NOMOVE Or SWP_NOZORDER
    Call Map_draw_scale
End Sub
```

```
Private Sub Command1_Click()
    Call Map_draw_scale
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Form_Activate()
    Call Map_draw_scale
End Sub
```

```
Private Sub Form_Load()
    Dim lpszName As String * 100
    Dim lpszVer As String * 100
    Dim Caps As CAPDRIVERCAPS
    capGetDriverDescriptionA 0, lpszName, 100, lpszVer, 100
    lwndC = capCreateCaptureWindowA(lpszName, WS_CHILD Or
    WS_VISIBLE, 10, 10, 160, 120, Me.hwnd, 0)
    capDriverConnect lwndC, 0
    capDriverGetCaps lwndC, VarPtr(Caps), Len(Caps)
    If Caps.fHasDlgVideoSource = 0 Then mnuSource.Enabled = False
    If Caps.fHasDlgVideoFormat = 0 Then mnuFormat.Enabled = False
    capSetCallbackOnVideoStream lwndC, AddressOf MyVideoStreamCallback
    capSetCallbackOnFrame lwndC, AddressOf MyFrameCallback
    capPreviewRate lwndC, 66
    capPreview lwndC, True
    ResizeCaptureWindow lwndC
    MSComm1.Settings = "9600,N,8,1"
    MSComm1.CommPort = 1
    MSComm1.InputLen = 1
    MSComm1.PortOpen = True
    MSComm1.RThreshold = 1
    rs_index = 1
    dist2 = 0
    dist1 = 0
    pre_data = 0
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
    capSetCallbackOnError lwndC, vbNull
    capSetCallbackOnStatus lwndC, vbNull
    capSetCallbackOnYield lwndC, vbNull
    capSetCallbackOnFrame lwndC, vbNull
    capSetCallbackOnVideoStream lwndC, vbNull
    capSetCallbackOnWaveStream lwndC, vbNull
    capSetCallbackOnCapControl lwndC, vbNull
End Sub
```

```
Private Sub mnuExit_Click()
    Unload Me
End Sub
```

```
Private Sub mnuFormat_Click()
    capDlgVideoFormat lwndC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    ResizeCaptureWindow lwndC
End Sub

```

```

Private Sub mnuPreview_Click()
    mnuPreview.Checked = Not (mnuPreview.Checked)
    capPreview lwndC, mnuPreview.Checked
End Sub

```

```

Private Sub mnuSource_Click()
    capDlgVideoSource lwndC
End Sub

```

```

Private Sub MSComm1_OnComm()
    If pre_data = 0 Then
        pre_data = MSComm1.Input
    ElseIf pre_data = "h" Then
        If rs_index = 1 Then
            command_in = MSComm1.Input
            Text1.Text = command_in
            rs_index = 2
        ElseIf rs_index = 2 Then
            data1 = MSComm1.Input
            Text2.Text = Hex(Asc(data1))
            If Hex(Asc(data1)) = 1 Then
                dist1 = dist1 + 1
                Label11.Caption = dist1
            End If
            rs_index = 3
        ElseIf rs_index = 3 Then
            data2 = MSComm1.Input
            Text3.Text = Hex(Asc(data2))
            If Hex(Asc(data2)) = 1 Then
                dist2 = dist2 + 1
                Label12.Caption = dist2
            End If
            rs_index = 1
            pre_data = 0
        End If
        End If
        Text4.Text = "IN"
    End Sub

```

```

Private Sub Timer1_Timer()
    Dim i As Integer
    joyGetPos 0, my_joy

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

joy_y = Int(my_joy.wYpos / 65280 * 100)
joy_z = Int(my_joy.wZpos / 65280 * 100)
Label2.Caption = joy_x
Label4.Caption = joy_y
Label7.Caption = joy_z
For i = 0 To 9
    If (my_joy.wButtons And 2 ^ i) = 2 ^ i Then
        Shape(i).BackColor = vbRed
    Else
        Shape(i).BackColor = QBColor(15)
    End If
Next i
End Sub

Private Sub Timer2_Timer()
    If joy_oy <> joy_y Or joy_ox <> joy_x Then
        If joy_y > 54 And joy_x < 60 And joy_x > 40 Then
            output1 = ((joy_y - 50) * 4)
            Labels.Caption = output1
            joy_oy = joy_y
            joy_ox = joy_x
            MSComm1.Output = Chr(170) & "R" & "D" & Chr(output1) &
Chr(output1)
            MSComm1.Output = Chr(170) & "R" & "D" & Chr(output1) &
Chr(output1)
            ElseIf joy_y < 45 And joy_x < 60 And joy_x > 40 Then
                output1 = ((50 - joy_y) * 4)
                Labels.Caption = output1
                joy_oy = joy_y
                joy_ox = joy_x
                MSComm1.Output = Chr(170) & "R" & "A" & Chr(output1) &
Chr(output1)
                MSComm1.Output = Chr(170) & "R" & "A" & Chr(output1) &
Chr(output1)
                ElseIf joy_x > 54 And joy_y < 60 And joy_y > 40 Then
                    output1 = ((joy_x - 50) * 4)
                    Labels.Caption = output1
                    joy_oy = joy_y
                    joy_ox = joy_x
                    MSComm1.Output = Chr(170) & "R" & "C" & Chr(output1) &
Chr(output1)
                    MSComm1.Output = Chr(170) & "R" & "C" & Chr(output1) &
Chr(output1)
                    ElseIf joy_x < 45 And joy_y < 60 And joy_y > 40 Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

output1 = ((50 - joy_x) * 4)
Labels.Caption = output1
joy_oy = joy_y
joy_ox = joy_x
MSComm1.Output = Chr(170) & "R" & "B" & Chr(output1) &
Chr(output1)
MSComm1.Output = Chr(170) & "R" & "B" & Chr(output1) &
Chr(output1)
ElseIf joy_x < 40 And joy_y < 44 Then
output1 = ((50 - joy_y) * 4)
Labels.Caption = output1
output2 = 1
joy_oy = joy_y
joy_ox = joy_x
MSComm1.Output = Chr(170) & "R" & "B" & Chr(output1) &
Chr(output2)
MSComm1.Output = Chr(170) & "R" & "B" & Chr(output1) &
Chr(output2)
ElseIf joy_x > 60 And joy_y < 40 Then
output2 = ((50 - joy_y) * 4)
Labels.Caption = output1
output1 = 1
joy_oy = joy_y
joy_ox = joy_x
MSComm1.Output = Chr(170) & "R" & "C" & Chr(output1) &
Chr(output2)
MSComm1.Output = Chr(170) & "R" & "C" & Chr(output1) &
Chr(output2)
ElseIf joy_y > 60 And joy_x < 40 Then
output1 = ((joy_y - 50) * 4)
Labels.Caption = output1
output2 = 1
joy_oy = joy_y
joy_ox = joy_x
MSComm1.Output = Chr(170) & "R" & "C" & Chr(output1) &
Chr(output2)
MSComm1.Output = Chr(170) & "R" & "C" & Chr(output1) &
Chr(output2)
ElseIf joy_x > 60 And joy_y > 60 Then
output2 = ((joy_y - 50) * 4)
Labels.Caption = output1
output1 = 1
joy_oy = joy_y
joy_ox = joy_x

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MSComm1.Output = Chr(170) & "R" & "B" & Chr(output1) &
Chr(output2)
MSComm1.Output = Chr(170) & "R" & "B" & Chr(output1) &
Chr(output2)
Elseif joy_y < 55 And joy_y > 44 And joy_x < 55 And joy_x > 44 Then
joy_oy = joy_y
joy_ox = joy_x
Labels.Caption = 0
MSComm1.Output = Chr(170) & "R" & "E" & Chr(0) & Chr(0)
MSComm1.Output = Chr(170) & "R" & "E" & Chr(0) & Chr(0)
MSComm1.Output = Chr(170) & "R" & "E" & Chr(0) & Chr(0)
MSComm1.Output = Chr(170) & "R" & "E" & Chr(0) & Chr(0)
End If
End If
End Sub

Sub Map_draw_scale()
Dim x As Integer
Dim y As Integer
Map1.Scale (0, 0)-(14, 12)
Map1.ForeColor = vbGreen
For x = 1 To 14
Map1.Line (x, 0)-(x, 12)
Next x
For y = 1 To 12
Map1.Line (0, y)-(14, y)
Next y
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ROBOT RESCUE V1.0 BY MCS-51

```

*****
,
***
MOTOR_R_F      BIT      P0.0
MOTOR_R_R      BIT      P0.1
MOTOR_L_F      BIT      P0.2
MOTOR_L_R      BIT      P0.3

PWM_R          BIT      P0.4
PWM_L          BIT      P0.5

LASER          BIT      P1.4
LIGHT          BIT      P1.5

CAMERA_UP      BIT      P1.0
CAMERA_DOWN    BIT      P1.1
CAMERA_LEFT    BIT      P1.2
CAMERA_RIGHT   BIT      P1.3

COMMAND        EQU      29H
DATA_1         EQU      2AH
DATA_2         EQU      2BH

RX_COMMAND     EQU      20H
RX_SER_DATA1   EQU      21H
RX_SER_DATA2   EQU      22H
RX_SER_VECTOR  EQU      23H

MOR_ON_R       EQU      24H
MOR_ON_L       EQU      25H
MOR_OFF_R      EQU      26H
MOR_OFF_L      EQU      27H

MOVEMENT       EQU      28H
PRE_DATA       EQU      34H
DISTAN_L       EQU      35H
DISTAN_R       EQU      36H

MOR_ON_DATA1   EQU      2CH
MOR_ON_DATA2   EQU      2DH
MOR_OFF_DATA1  EQU      2EH
MOR_OFF_DATA2  EQU      2FH
STATUS         EQU      28H

TX_COMMAND     EQU      30H
TX_DATA_1      EQU      31H
TX_DATA_2      EQU      32H
*****
,
***
:
:
:
:
:
*****
***
                                INTERUB VECTOR
*****
ORG      0000H
AJMP     MAIN
ORG      0003H
LJMP     DIST_L
ORG      0013H

```

เอกสารนี้เป็นเอกสารที่สงวนเวลาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

RUN_NEXT4:
  CJNE      R1,#'F',RUN_NEXT5
  SETB     CAMERA_UP
  CLR      CAMERA_DOWN
  CLR      CAMERA_RIGH
  CLR      CAMERA_LEFT
  LJMPL    RUN_RETURN
RUN_NEXT5:
  CJNE      R1,#'G',RUN_NEXT6
  CLR      CAMERA_UP
  SETB     CAMERA_DOWN
  CLR      CAMERA_RIGH
  CLR      CAMERA_LEFT
  LJMPL    RUN_RETURN
RUN_NEXT6:
  CJNE      R1,#'H',RUN_NEXT7
  CLR      CAMERA_UP
  CLR      CAMERA_DOWN
  SETB     CAMERA_RIGH
  CLR      CAMERA_LEFT
  LJMPL    RUN_RETURN
RUN_NEXT7:
  CJNE      R1,#'I',RUN_NEXT8
  CLR      CAMERA_UP
  CLR      CAMERA_DOWN
  CLR      CAMERA_RIGH
  SETB     CAMERA_LEFT
  LJMPL    RUN_RETURN
RUN_NEXT8:
  CJNE      R1,#'J',RUN_NEXT9
  CLR      CAMERA_UP
  CLR      CAMERA_DOWN
  CLR      CAMERA_RIGH
  CLR      CAMERA_LEFT
  LJMPL    RUN_RETURN
RUN_NEXT9:
  CJNE      R1,#'K',RUN_NEXT10
  SETB     LASER
  LJMPL    RUN_RETURN
RUN_NEXT10:
  CJNE      R1,#'L',RUN_NEXT11
  CLR      LASER
  LJMPL    RUN_RETURN
RUN_NEXT11:
  CJNE      R1,#'M',RUN_NEXT12
  SETB     LIGHT
  LJMPL    RUN_RETURN
RUN_NEXT12:
  CJNE      R1,#'N',RUN_RETURN
  CLR      LIGHT
  LJMPL    RUN_RETURN
RUN_RETURN:
  MOV      STATUS,#00H
  JMP      RUN_ALWAYS

```


*
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.*
;
GENERAL SET TO FIRST

.*
;

.*****
;
***
FIST_SETUP:
    MOV     STATUS,#00H
    MOV     COMMAND,#00H
    MOV     TH0,#07FH
    RET

.*****
;
***
.*
;
SET SERIES TO INTERUB MODE 9600

.*****
;
***
SERIER_SETUP:
    MOV     TMOD,#022H
    MOV     PCON,#080H
    MOV     SCON,#050H
    MOV     TH1,#0FDH
    MOV     RX_SER_VECTOR,#01H
    SETB    TR1
    MOV     IE,#10010010B
    RET

.*****
;
***
.*
;
SER_INT SERVICE RECIVE DATA 3 BYTE

.*****
;
***
SER_INT:
    PUSH    ACC
    CLR     EA
    JNB     RI,SER_INT_EXIT
    MOV     A,SBUF
    MOV     R0,RX_SER_VECTOR
    CJNE   R0,#01H,SER_INT1
    MOV     RX_COMMAND,A
    MOV     RX_SER_VECTOR,#02H
    AJMP   SER_INT_EXIT

SER_INT1:
    CJNE   R0,#02H,SER_INT2
    MOV     RX_SER_DATA1,A
    MOV     RX_SER_VECTOR,#03H
    AJMP   SER_INT_EXIT

SER_INT2:
    CJNE   R0,#03H,SER_INT_EXIT
    MOV     RX_SER_DATA2,A
    MOV     RX_SER_VECTOR,#01H
    MOV     STATUS,#0000001B
    MOV     COMMAND,RX_COMMAND
    MOV     DATA_1,RX_SER_DATA1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV          DATA_2,RX_SER_DATA2

SER_INT_EXIT:
CLR          RI
POP          ACC
SETB        EA
RETI

```

```

*****
***
.*
.*
.*
.*
.*
.*

```

TRANSMISSION DATA 3 BYTE

```

*****
***

```

```

SEND_DATA:
CLR          TI
MOV          SBUF,COMMAND
JNB          TI,$
ACALL        DELAY
MOV          SBUF,DATA_1
JNB          TI,$
ACALL        DELAY
MOV          SBUF,DATA_2
JNB          TI,$
CLR          TI
MOV          STATUS,#00H
RET

```

```

*****
***
:
:
:
:
:
:

```

MORTOR SET TO MOVE

```

*****
***

```

```

SET_DATA_MOVE:
MOV          A,DATA_1
MOV          MOR_ON_R,DATA_1
SUBB        A,#100
MOV          MOR_OFF_R,A
MOV          A,DATA_2
MOV          MOR_ON_L,DATA_2
SUBB        A,#100
MOV          MOR_OFF_L,A
MOV          MOR_ON_DATA1,MOR_ON_R
MOV          MOR_ON_DATA2,MOR_ON_L
MOV          MOR_OFF_DATA1,MOR_OFF_R
MOV          MOR_OFF_DATA2,MOR_OFF_L
RET

```

```

*****
***
:
:
:
:
:
:

```

PWM

```

*****
***

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RETI

```
*****
;
***
```

CALCURATE DISTANT R INT_EX0

```
*****
;
***
```

DIST_R:

```
CLR      EX0
CLR      EX1
CLR      PT0
MOV      TX_COMMAND,MOVEMENT
MOV      TX_DATA_2,#01H
MOV      TX_DATA_1,#00H
LCALL    SEND_DATA
SETB     EX0
SETB     EX1
SETB     PT0
RETI
```

```
*****
;
***
```

DELAY TIME

```
*****
;
***
```

DELAY:

```
MOV      R1,#02H
```

LOOP_DELAY:

```
MOV      R2,#07FH
DJNZ     R2,$
DJNZ     R1,LOOP_DELAY
RET
```

```
*****
;
***
```

END

```
*****
;
***
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการนี้จะไม่สามารถคล่องไปได้ ถ้าไม่ได้รับความร่วมมือและความช่วยเหลือจากหลาย ๆ ฝ่าย โดยเฉพาะอาจารย์ รศ.ดร. สุรพันธุ์ เอื้อไพฑูลย์ (อาจารย์ที่ปรึกษา) , อาจารย์ ผศ.ดร.บุษนา กิจใจเดียว , อาจารย์พลศาสตร์ เลิศประเสริฐ และอาจารย์ในภาควิชาอิเล็กทรอนิกส์ทุกท่าน ที่ให้การอุปการะในการให้คำปรึกษาและแนะนำเกี่ยวกับโครงการในครั้งนี้ และให้ยืมใช้เครื่องมือและอุปกรณ์อิเล็กทรอนิกส์ในการทดลอง และสั่งสอนให้ความรู้จนสามารถนำมาประยุกต์ใช้งานในการทำโครงการครั้งนี้ อีกทั้งเพื่อนที่ได้ร่วมทำงานและแลกเปลี่ยนความรู้ ความคิดเห็นซึ่งกันและกัน จนทำให้ผลงานที่ได้มีประสิทธิภาพและเป็นประโยชน์ต่อส่วนรวมอีกด้วย



นายกรวิทย์ เสวตนันท์
 นายจักรพงษ์ แสงนิล
 ผู้จัดทำโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] วรพจน์ กรแก้ววัฒนกุล, ชัยวัฒน์ ลิ้มพรจิตรวิไล, เรียนรู้และปฏิบัติการ ไมโครคอนโทรลเลอร์ MCS-51, กรุงเทพฯ, อินโนเวตีฟ เอ็กเพอริเมนส์, 2538
- [2] ฉันทวุฒิ พีชผล, คู่มือการเรียน Visual Basic, กรุงเทพฯ, โปรวิชั่น, 2547
- [3] อภิชาติ ภู่อุปถัมภ์, สนุกกับการประยุกต์ใช้ Visual Basic, นนทบุรี, อินโฟเพรส, 2546
- [4] วรลักษณ์ อินทร์มี, วิทยา กสิบเมฆ, “รถสำรวจโดยคอมพิวเตอร์” ปริญญาานิพนธ์ วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมอิเล็กทรอนิกส์, คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, กรุงเทพฯ, 2546



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้