การออกแบบและการสร้างหุ่นยนต์หลบหลีก

Design and Construction of Mobile Robot with an Obstacle Avoidance

นาย ธนากร เขียวแก้ว รหัสนิสิต 44362242
นาย เทิดพิทักษ์ ประจง รหัสนิสิต 44362481

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2547

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 17 ส.ค. 2549
เลขทะเบียน 49.00021
เลขเรียกหนังสือ
มหาวิทยาลัยนเรศวร

TK5625666 e.2
ส5
ธ2310
2547

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน การออกแบบและการสร้างหุ่นยนต์หลบหลีก

ผู้ดำเนินโครงงาน นายธนากร เขียวแก้ว รหัส 44362242

นายเทิดพิทักษ์ ประจง รหัส 44362481

อาจารย์ที่ปรึกษา อาจารย์พนัส นัถฤทธิ์

สาขาวิชา วิศวกรรมไฟฟ้า

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2547

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ

(อาจารย์พนัส นัถฤทธิ์)

..... กรรมการ

(อาจารย์สมยศ เกียรติวนิชวิไล)

..... กรรมการ

(อาจารย์ธนิต มาลากร)

หัวข้อโครงงาน	การออกแบบและการสร้างหุ่นยนต์หลบหลีก	
ผู้ดำเนินโครงงาน	นายธนากร เขียวแก้ว	รหัส 44362242
	นายเทิดพิทักษ์ ประจง	รหัส 44362481
อาจารย์ที่ปรึกษา	อาจารย์พนัส นัถฤทธิ์	
สาขาวิชา	วิศวกรรมไฟฟ้า	
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์	
ปีการศึกษา	2547	

.....

บทคัดย่อ

โครงงานนี้เป็นการนำเสนอเกี่ยวกับการออกแบบ และสร้างหุ่นยนต์หลบหลีกสิ่งกีดขวางที่สามารถทำงานได้โดยอัตโนมัติ ซึ่งจะเป็นการศึกษาหลักการทำงานของระบบไมโครคอนโทรลเลอร์ ที่จะใช้ในการเขียนโปรแกรม เพื่อออกคำสั่งให้หุ่นยนต์สามารถควบคุมและตัดสินใจในการหลบหลีกสิ่งกีดขวางต่างๆที่เป็นอุปสรรคในการเคลื่อนที่ของหุ่นยนต์ และสามารถเคลื่อนที่สู่จุดหมายที่กำหนดไว้ได้

ผลที่คาดว่าจะได้รับจากโครงงานนี้คือ สามารถสร้างหุ่นยนต์หลบหลีกสิ่งกีดขวางเพื่อเป็นต้นแบบในการประยุกต์ใช้งานจริงในด้านต่างๆ เช่น เข้าถึงพื้นที่เสี่ยงอันตรายที่มนุษย์ไม่สามารถเข้าถึงได้ หรือเพื่อนำไปพัฒนาต่อเป็นหุ่นยนต์อเนกประสงค์ใช้งานในด้านอื่นๆได้

Project title	Design and Construction of Mobile Robot with an Obstacle Avoidance	
Name	Mr. Tanakorn Keawkaw	ID. 44362242
	Mr. Thirdpitak Prajoung	ID. 44362465
Project advisor	Mr. Panus Nattharith	
Major	Electrical Engineering	
Department	Electrical and Computer Engineering	
Academic year	2004	

.....

ABSTRACT

This project presents the design and invention of the robot which can avoid the barriers automatically. It is the study on the principle of microcontroller working system to control the robot for deciding to avoid the barriers which is the main problem of the robot movement, and to direct the robot to reach the fixed destination.

The result of this project is the capability of invention of the robot which can avoid the barriers as the applied model for various converted utilization. For instance, the robot can enter through the poisonous areas where the human can not enter, or it can be developed to apply in the various fields.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ด้วยความช่วยเหลือจากหลายๆ ท่าน ผู้จัดทำจึงถือโอกาสนี้ขอกราบขอบพระคุณ อาจารย์พนัส นัถฤทธิ์ ซึ่งเป็นอาจารย์ที่ปรึกษาและคณะกรรมการสอบโครงงานทุกท่านที่ได้ให้คำปรึกษาชี้แนะแนวทางและข้อคิดเห็นต่างๆ ในการแก้ปัญหาที่เป็นประโยชน์อย่างสูงในการทำโครงงานนี้ให้สำเร็จลุล่วงด้วยดี

ขอบคุณอาจารย์ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ทุกท่าน ที่ได้ประสิทธิ์ประสาทวิชาจนมีความรู้ความสามารถเพียงพอที่จะทำโครงงาน และทำงานจริงได้

ขอบคุณพี่ๆ เพื่อนๆ และน้องๆ นิสิตภาควิชาวิศวกรรมไฟฟ้าทุกคนที่ได้ให้ความช่วยเหลือในหลายๆ ด้าน ทั้งเรื่องส่วนตัวและเรื่องเรียนด้วยดีเสมอมา

ท้ายนี้ผู้จัดทำโครงงานขอกราบขอบพระคุณบิดา มารดา และญาติพี่น้องของข้าพเจ้าที่เลี้ยงดูและคอยสนับสนุนด้านการเงิน รวมทั้งเป็นกำลังใจให้ผู้จัดทำเสมอมาจนสำเร็จการศึกษา

คณะผู้จัดทำโครงงาน
นายธนากร เขียวแก้ว
นายเทิดพิทักษ์ ประจง

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ก
บทคัดย่อภาษาอังกฤษ.....ข
กิตติกรรมประกาศ.....ค
สารบัญ.....ง
สารบัญรูป.....ฉ

บทที่ 1 บทนำ
1.1 ที่มาและความสำคัญของโครงงาน.....1
1.2 วัตถุประสงค์.....1
1.3 ขอบข่ายงาน.....1
1.4 กิจกรรมการดำเนินงาน.....2
1.5 ผลที่คาดว่าจะได้รับ.....2
1.6 งบประมาณ.....2

บทที่ 2 หลักการและทฤษฎีเบื้องต้น
2.1 พื้นฐานของระบบไมโครโพรเซสเซอร์.....3
2.2 การควบคุมทิศทางการหมุนของ DC Motor.....17
2.3 เซ็นเซอร์ตรวจจับวัตถุ.....20

บทที่ 3 การออกแบบและการวิเคราะห์การทำงานของวงจร
3.1 วงจรภาคตรวจจับวัตถุ (วงจรภาคเซ็นเซอร์).....21
3.2 วงจรควบคุม (วงจรภาคไมโครคอนโทรลเลอร์).....22
3.3 วงจรขับมอเตอร์.....23
3.4 วงจรภาคจ่ายไฟ.....24
3.5 โครงสร้างของตัวหุ่นยนต์.....25
3.6 แผนภาพการไหล.....29

สารบัญ(ต่อ)

หน้า

บทที่ 4 การทดลองและวิธีการทดลอง

4.1 การทดลองที่ 1 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์.....30

4.2 การทดลองที่ 2 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์.....33

4.3 การทดลองที่ 3 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า , ซ้ายและขวาของหุ่นยนต์.....35

บทที่ 5 สรุปผลการทดลอง

5.1 วิเคราะห์ผลการทดลอง.....41

5.2 สรุปขั้นตอนการทำงานของหุ่นยนต์หลบหลีก.....42

5.3 ปัญหาและแนวทางแก้ไข.....43

5.4 ข้อเสนอแนะ.....43

เอกสารอ้างอิง.....44

ภาคผนวก.....45

ประวัติผู้เขียน โครงงาน.....49

สารบัญรูป

รูปที่ หน้า
รูปที่ 2.1 แสดงไดอะแกรมวงจรภายในชิพ AT 89C52....3
รูปที่ 2.2 ไดอะแกรมสถาปัตยกรรมภายภายใน AT 89C52....4
รูปที่ 2.3 ไดอะแกรมสถาปัตยกรรมภายในและขาสัญญาณของตัวถังพลาสติก 40 ขา แบบ PDIP และ PLCC....5
รูปที่ 2.4 ไดอะแกรมผังเวลาลำดับการ Fetch และ Execute....6
รูปที่ 2.5 ไดอะแกรมผังเวลาเมื่อใช้คำสั่งติดต่อหน่วยความจำภายนอก....6
รูปที่ 2.6 โครงสร้างของหน่วยความจ าโปรแกรมและหน่วยความจำข้อมูล....7
รูปที่ 2.7 ผังหน่วยความจำข้อมูลและการแบ่งพื้นที่ตำแหน่ง 00H – 7FH....8
รูปที่ 2.8 Program Status Word Register....10
รูปที่ 2.9 ผังหน่วยความจำแสดงตำแหน่งไบท์และตำแหน่งบิทที่ใช้เป็นพื้นที่ของรีจิสเตอร์หน้าที่พิเศษ....13
รูปที่ 2.10 รูปจำลองของวงจร H-Bridge....17
รูปที่ 2.11 วงจรขณะสวิตช์ S1 และ S3 ปิดวงจร....17
รูปที่ 2.12 วงจรขณะสวิตช์ S2 และ S4 ปิดวงจร....18
รูปที่ 2.13 วงจร H-Bridge ที่ใช้ทรานซิสเตอร์แทนสวิตช์....18
รูปที่ 2.14 วงจรขณะ Q1 และ Q3 ทำงาน....19
รูปที่ 2.15 วงจรขณะ Q2 และ Q4 ทำงาน....19
รูปที่3.1 วงจรภาคตรวจจับวัตถุ....21
รูปที่3.2 วงจรควบคุม....22
รูปที่3.3 วงจรขับมอเตอร์....23
รูปที่3.4 วงจรภาคจ่ายไฟ....24
รูปที่3.5 แสดงโครงสร้างหุ่นยนต์ด้านข้าง....25
รูปที่3.6 แสดงโครงสร้างหุ่นยนต์ด้านหน้า....25
รูปที่3.7 แสดงโครงสร้างหุ่นยนต์ด้านบน....26
รูปที่ 3.7 ด้านหน้าของตัวหุ่นเมื่อสร้างเสร็จแล้ว....27
รูปที่ 3.8 ด้านข้างของตัวหุ่นเมื่อสร้างเสร็จแล้ว....27
รูปที่ 3.9 ด้านหลังของตัวหุ่นเมื่อสร้างเสร็จแล้ว....28
รูปที่ 3.10 ด้านข้างของตัวหุ่นเมื่อสร้างเสร็จแล้ว....28

สารบัญรูป(ต่อ)

รูปที่ หน้า
รูปที่3.11 Flowchart แสดงทำงานของหุ่นยนต์หลบหลีก.....29
รูปที่ 4.1 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์.....30
รูปที่ 4.2 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์ (ต่อ).....31
รูปที่ 4.3 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์ (ต่อ).....31
รูปที่ 4.4 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์ (ต่อ).....32
รูปที่ 4.5 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์(ต่อ.....32
รูปที่ 4.6 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์.....33
รูปที่ 4.7 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ).....33
รูปที่ 4.8 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ).....34
รูปที่ 4.9 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ.....34
รูปที่ 4.10 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ).....35
รูปที่ 4.11 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์.....36
รูปที่ 4.12 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....36
รูปที่ 4.13 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....37
รูปที่ 4.14 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....37
รูปที่ 4.15 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....38
รูปที่ 4.16 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....38
รูปที่ 4.17 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....39
รูปที่ 4.18 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....39
รูปที่ 4.19 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....40

สารบัญรูป(ต่อ)

รูปที่ หน้า

รูปที่ 4.20 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า ,
ซ้ายและขวาของหุ่นยนต์(ต่อ).....40

บทที่1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ในปัจจุบันเทคโนโลยีต่างๆมีความก้าวล้ำนำสมัยขึ้นมากทุกๆวัน มนุษย์ได้นำเอาเทคโนโลยีสมัยใหม่เหล่านั้นมาสร้างสรรค์สิ่งประดิษฐ์ต่างๆ เพื่ออำนวยความสะดวกในด้านต่างๆแก่มนุษย์ หุ่นยนต์ก็เป็นอีกนวัตกรรมหนึ่งที่มนุษย์สร้างขึ้นมาเพื่อนำไปใช้ประโยชน์ในด้านต่างๆเช่น งานที่เสี่ยงภัยอันตราย งานที่ต้องการความละเอียดเที่ยงตรงและแม่นยำ หรือนำไปใช้ในงานสำรวจในพื้นที่ที่มนุษย์ไม่สามารถเข้าถึงได้ เป็นต้น

หุ่นยนต์หลบหลีกเป็นหุ่นยนต์ที่ผู้ดำเนินโครงงานสร้างขึ้น เพื่อศึกษาการทำงานของระบบไมโครคอนโทรลเลอร์ที่ใช้เขียนโปรแกรมบรรจุลงในบอร์ดที่ใช้เป็นตัวควบคุมและตัดสินใจ เพื่อเป็นตัวออกคำสั่งให้หุ่นยนต์สามารถหลบหลีกสิ่งกีดขวางต่างๆ ที่เป็นอุปสรรคในการเคลื่อนที่ของหุ่นยนต์ โดยมีตัวเซนเซอร์เป็นตัวตรวจจับสิ่งกีดขวางซึ่งหุ่นยนต์สามารถตัดสินใจได้เองโดยอัตโนมัติว่าควรจะเคลื่อนที่ต่อไปในทิศทางใดจึงจะสามารถหลบสิ่งกีดขวาง และสามารถเคลื่อนที่สู่จุดหมายที่กำหนดไว้ได้ รวมทั้งมีวงจรสำหรับขยายสำหรับขับมอเตอร์ ซึ่งถูกนำมาใช้ในการขับเคลื่อนตัวหุ่นยนต์ของโครงงานนี้ด้วย

1.2 วัตถุประสงค์

1. เพื่อศึกษาระบบไมโครคอนโทรลเลอร์ในการเขียนโปรแกรมออกคำสั่งหุ่นยนต์
2. เพื่อสร้างหุ่นยนต์ที่สามารถเคลื่อนที่หลบหลีกสิ่งกีดขวางได้
3. เพื่อสร้างหุ่นยนต์ต้นแบบสำหรับการนำไปประยุกต์ใช้งานจริงได้

1.3 ขอบข่ายงาน

โครงงานนี้เป็นการศึกษาเกี่ยวกับระบบไมโครคอนโทรลเลอร์ ซึ่งมีขอบเขตในการศึกษาดังนี้

1. ศึกษาการทำงานและการเขียนโปรแกรมสำหรับไมโครโปรเซสเซอร์เบอร์AT89C52โดยใช้ภาษาแอสแซมบลี
2. ศึกษาการทำงานของชุด sensor ชนิดอินฟราเรดและวงจรขยายสัญญาณสำหรับขับมอเตอร์
3. ศึกษาการออกแบบโครงสร้างของหุ่นยนต์
4. หุ่นยนต์สามารถเคลื่อนที่หลบหลีกสิ่งกีดขวางและเคลื่อนที่ต่อถึงจุดหมายที่กำหนดได้

1.4 กิจกรรมการดำเนินงาน

	ปี2547		ปี2548									
กิจกรรม	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
1.เขียนโครงงานการทำงาน												
2.รวบรวมข้อมูลและเอกสาร												
3.ศึกษาการเขียนโปรแกรมภาษาแอสแซมบลีและออกแบบโครงสร้างของหุ่น												
4.สร้างและทดสอบการทำงาน												
5.ปรับปรุงและแก้ไขชิ้นงาน												
6.จัดทำเอกสารและคู่มือการใช้งาน												

1.5 ผลที่คาดว่าจะได้รับ

1. สามารถสร้างหุ่นยนต์หลบหลีกได้
2. สามารถนำความรู้ในการเขียนโปรแกรมไปประยุกต์ใช้ในด้านอื่นๆได้
3. สามารถประดิษฐ์และนำไปใช้เป็นอุปกรณ์ต้นแบบได้
4. เป็นเอกสารเพื่อใช้ในการอ้างอิง และค้นคว้าต่อไป
5. ฝึกการทำงานเป็นกลุ่มและการกำหนดระยะเวลาการทำงานได้อย่างมีประสิทธิภาพ

1.6 งบประมาณ

1.ค่าวัสดุอุปกรณ์และเครื่องมือ	1,500	บาท
2.ค่าจัดทำรูปเล่มโครงงาน	300	บาท
3.ค่าใช้จ่ายอื่นๆ	200	บาท
ถัวเฉลี่ยทุกรายการ	2,000	บาท

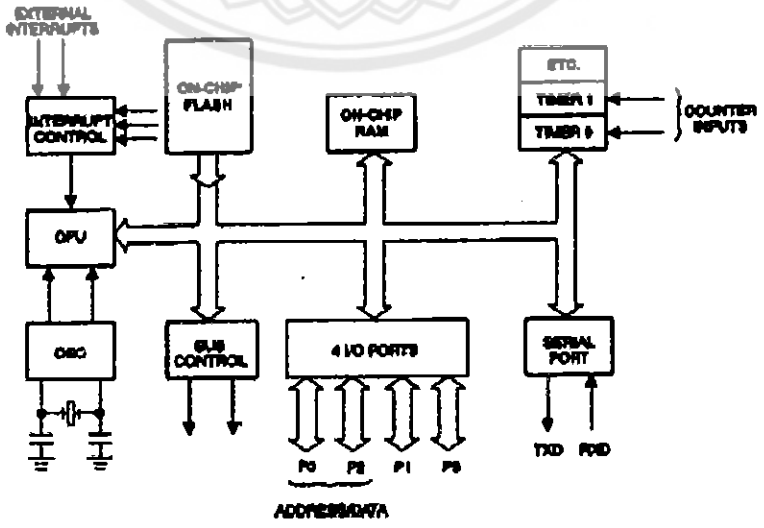
บทที่ 2
หลักการและทฤษฎีเบื้องต้น

2.1 พื้นฐานของระบบไมโครโพรเซสเซอร์

ภาษาแอสเซมบลีจัดได้ว่าเป็นภาษาระดับล่าง ไมโครโพรเซสเซอร์แต่ละตัวจะมีชุดคำสั่งรีจิสเตอร์ภายในชิพ การจัดหน่วยความจำ และฮาร์ดแวร์ที่เป็นอุปกรณ์พิเศษแตกต่างกัน การเขียนโปรแกรมภาษาแอสเซมบลีได้ดีกล่าวคือสั่งงานให้ไมโครโปรเซสเซอร์ทำงานถูกต้องนั้นจำเป็นจะต้องศึกษารายละเอียดของไมโครโพรเซสเซอร์เป็นการเฉพาะ

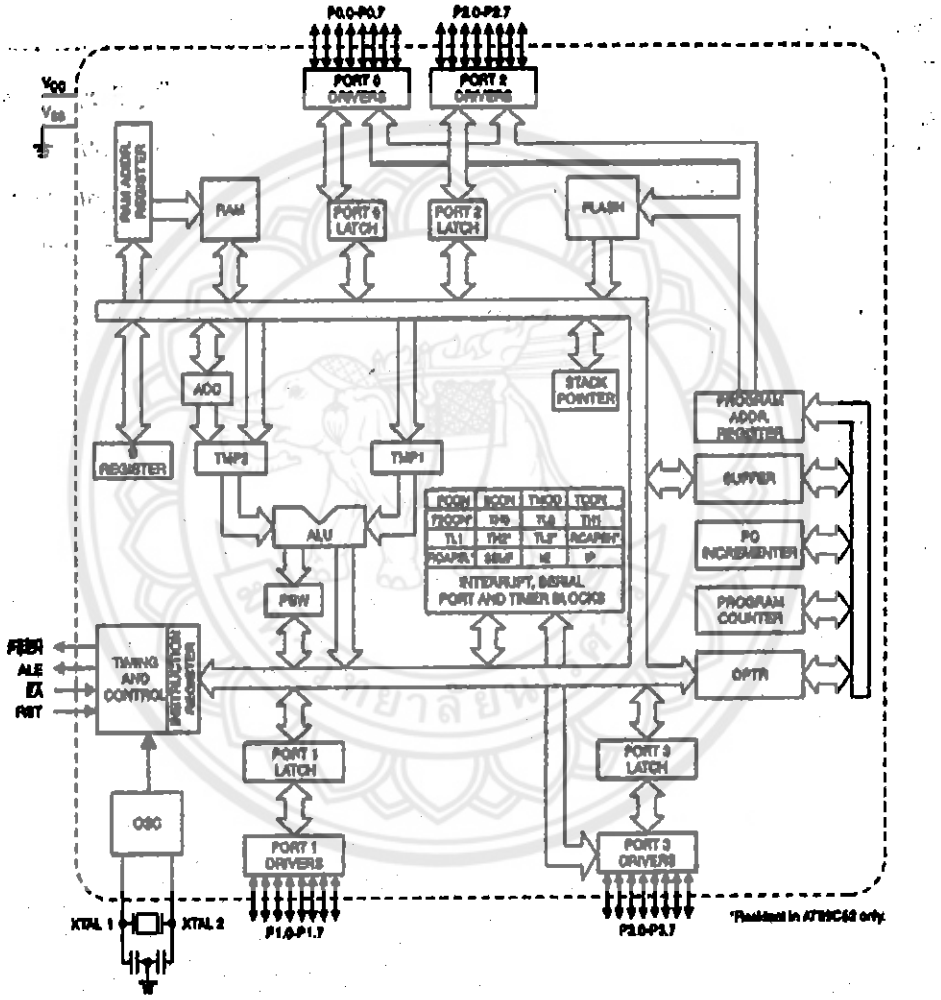
ไมโครโพรเซสเซอร์และไมโครคอนโทรลเลอร์

ไมโครคอมพิวเตอร์ มีส่วนประกอบหลักคือ ไมโครโพรเซสเซอร์ หน่วยความจำ และอินพุทเอาท์พุทไมโครโพรเซสเซอร์เรียกย่อๆได้ว่าเป็นหน่วยประมวลผลกลางหรือCPUไมโครคอนโทรลเลอร์หรืออดีตนิยมเรียกว่า ไมโครคอมพิวเตอร์ชิพเดี่ยว (Single Chip Microcomputer) ได้รวมเอาหน่วยความจำ อินพุทเอาท์พุทพอร์ท และวงจรพิเศษ เช่นวงจรตั้งเวลาเข้าไว้บนชิพแผ่นเดียว ทำให้เราไม่ต้องต่ออุปกรณ์ภายนอก แม้กระทั่งวงจรกำเนิดสัญญาณนาฬิกา เพียงแค่ต่อไฟเลี้ยง +5V วงจรรีเซ็ท และต่อคริสตอลใช้กำเนิดความถี่ในการทำงานไมโครคอนโทรลเลอร์ก็สามารถทำงานได้แล้ว รูปที่ 1 เป็นไดอะแกรมวงจรภายในชิพ AT 89C52 ภายในจะเห็นว่ามีCPU หรือไมโครโพรเซสเซอร์ต่อร่วมกับหน่วยความจำบนชิพ อินพุทเอาท์พุทพอร์ท พอร์ทอนุกรม และ ตัวตั้งเวลาเป็นต้น

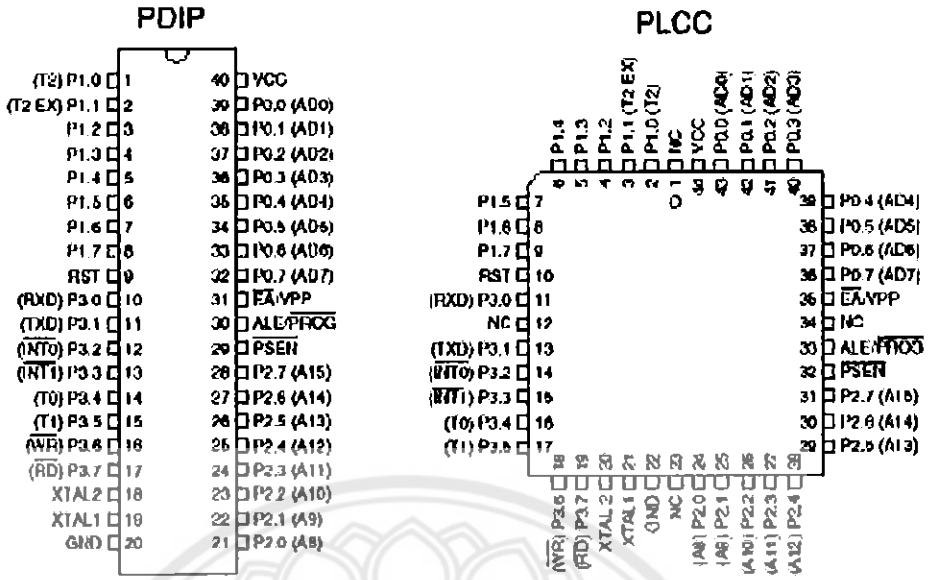


รูปที่ 2.1 แสดงไดอะแกรมวงจรภายในชิพ AT 89C52

AT 89C52 เป็นไมโครคอนโทรลเลอร์ ขนาด 8 บิท ชนิด CMOS มีหน่วยความจำโปรแกรมชนิดแฟลชขนาด 8 กิโลไบท์ สามารถลบได้ด้วยสัญญาณไฟฟ้า ชุดคำสั่งและขาสัญญาณเข้ากันได้กับไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งเป็นไมโครคอนโทรลเลอร์มาตรฐานในอุตสาหกรรมออกแบบโดย Intel จุดเด่นของไดอะแกรมวงจรภายในชิพ AT89C52 คือ มีหน่วยความจำโปรแกรมชนิดแฟลชขนาด 8 กิโลไบท์ หน่วยความจำข้อมูล 256 ไบท์ อินพุทเอาท์พุท 32 เส้น มีตัวตั้งเวลาและตัวนับขนาด 16 บิท 3ตัว อินเทอร์รัพท์ 6 ตำแหน่ง มีวงจรรับส่งข้อมูลอนุกรมชนิดฟูลดูเพล็กและมีวงจรกำเนิดสัญญาณนาฬิกา เป็นต้น



รูปที่ 2.2 ไดอะแกรมสถาปัตยกรรมภายภายใน AT 89C52



รูปที่ 2.3 ไดอะแกรมสถาปัตยกรรมภายในและขาสัญญาณของตัวถังพลาสติก 40 ขาแบบ PDIP และ PLCC

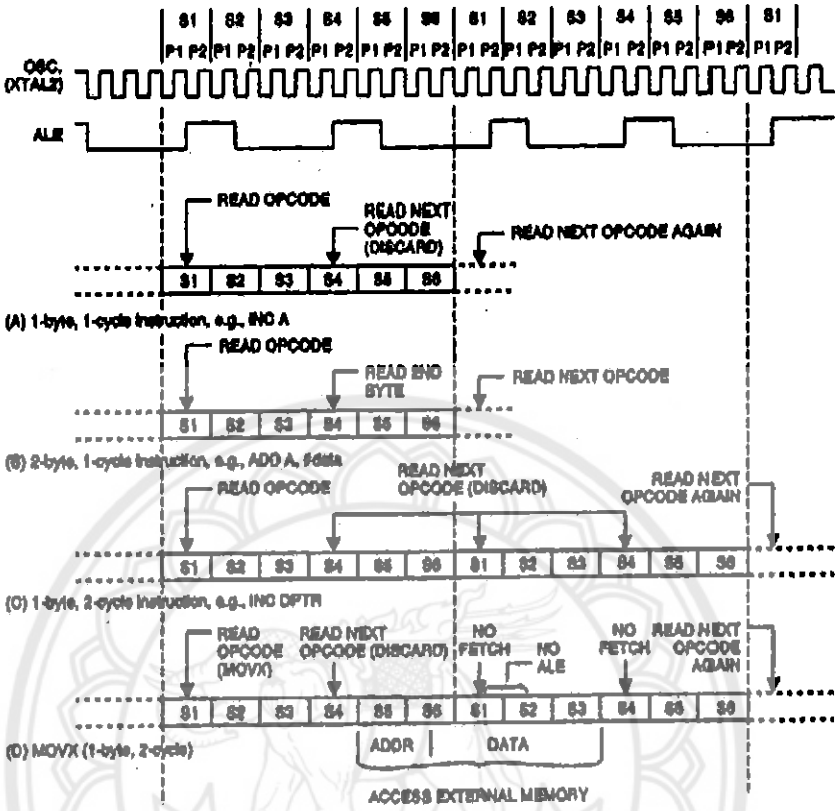
การทำงานของไมโครโพรเซสเซอร์

ไมโครโพรเซสเซอร์ทำงาน โดยการอ่านรหัสคำสั่งที่ประกอบขึ้นเป็นโ**ป**รแกรมบรร**จุ**ใน**หน่วยความ**จำโปรแกรม แล้วนำมาถอดรหัส กำเนิดสัญญาณต่างๆ ขึ้นกับว่าคำสั่งนั้นสั่งให้ CPU ทำ**อะไร** กระบวนการดังกล่าวเรียกว่ากระบวนการ Fetch และ Execute รูปที่ 2.4 แสดงไดอะแกรม**ลำดับการ**ทำงานของ CPU อ้างอิงกับสัญญาณนาฬิกา OSC หนึ่งรอบการทำงาน**ของแต่ละคำสั่ง**เรียกว่า แมชชีนไซเคิล (Machine Cycle)AT89C52 ใช้เวลาเท่ากับสัญญาณนาฬิกา 12 ลูก เช่นถ้าเรา**ใช้** X-tal ความถี่ 12MHz เวลาหนึ่งแมชชีนไซเคิลจะมีค่าเท่ากับ 1 µs คำสั่งที่มีขนาด 1 ไบท์ จะกิน**เวลา** 1 µs นั่นเอง

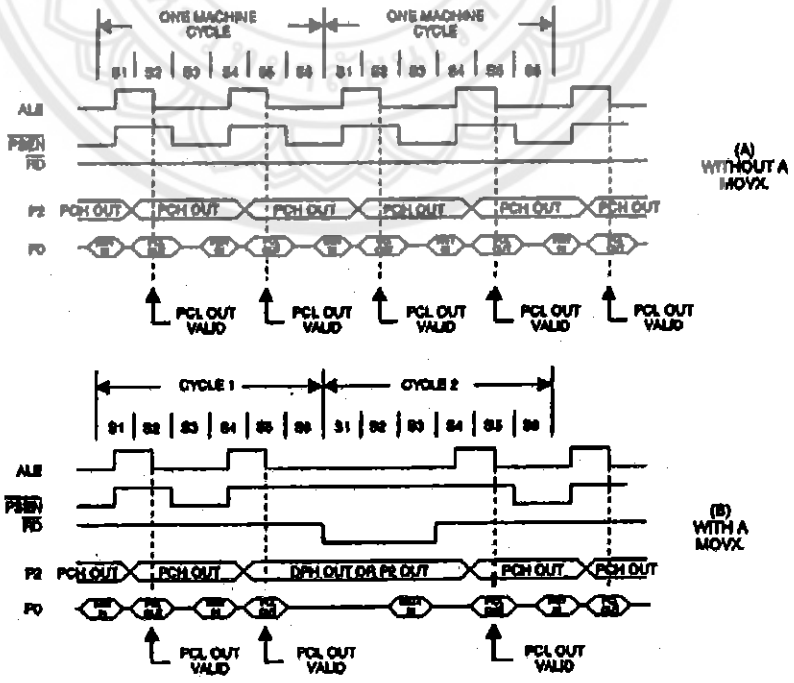
ตัวอย่างกรณี A เป็นคำสั่ง INC A ที่ใช้เวลาหนึ่งไซเคิล ใน S1 คำสั่งจะได้รับการอ่านเข้า มา**เก็บ**ในรีจิส**เตอร์**คำสั่ง (Instruction Register) และ CPU จะทำงานคำสั่งนี้โดยการ**เพิ่ม**ค่าแอคคิวมูเล**เตอร์ที่** S6 ตัวอย่างกรณีB เป็นคำสั่ง ADD A,#DATA ใช้เวลาหนึ่งไซเคิลแต่รหัสคำสั่งมี 2 ไบท์ ใน S1 คำสั่งจะได้รับการอ่านเข้ามาเก็บในรีจิสเตอร์คำสั่ง เนื่องจาก CPU รู้ว่าคำสั่งนี้มี 2 ไบท์ โปรแกรมเคาน์เตอร์จะเพิ่มค่าขึ้นหนึ่ง และที่ S4 CPU ก็จะอ่านข้อมูลไบท์ที่สอง จากนั้น CPU จะทำงานคำสั่งนี้ที่ S6โดยการนำข้อมูลไปเก็บในรีจิสเตอร์ A

ตัวอย่างกรณี C เป็นคำสั่ง INC DPTR ที่ใช้เวลาสองไซเคิลแต่รหัสคำสั่งมี 1 ไบท์ ใน S1 คำสั่งจะได้รับการอ่านเข้ามาเก็บในรีจิสเตอร์คำสั่ง และ CPU จะทำงานคำสั่งนี้ที่ S6 ของไซเคิลหนึ่งและสอง ที่ S4, S1และ S4 ดังรูป CPU จะไม่เพิ่มค่าโปรแกรมเคาน์เตอร์ตัวอย่างกรณี D เป็นคำสั่ง MOVX A,@A+DPTR ที่ใช้เวลาสองไซเคิลแต่รหัสคำสั่งมี 1 ไบท์ ใน S1 รหัสคำสั่ง MOVX จะได้รับการอ่านเข้ามาเก็บในเรจจิสเตอร์คำสั่ง และ CPU จะทำงานคำสั่งนี้ที่ S6 ของไซเคิลหนึ่งและสอง ที่ S4, S1 และ S4 ดังรูป CPU จะไม่เพิ่มค่าโปรแกรมเคาน์เตอร์ ที่ S5 CPU จะส่งค่าแอดเดรสไบท์

ต่ำออกทางพอร์ท P0 ส่วนค่าของ DPH จะนำมา OR กับ P2 ส่งออกทาง P2 ในช่วงไซเคิลที่สอง CPU จะกำเนิดสัญญาณ RD และอ่านค่าจากหน่วยความจำข้อมูลภายนอกที่ S3



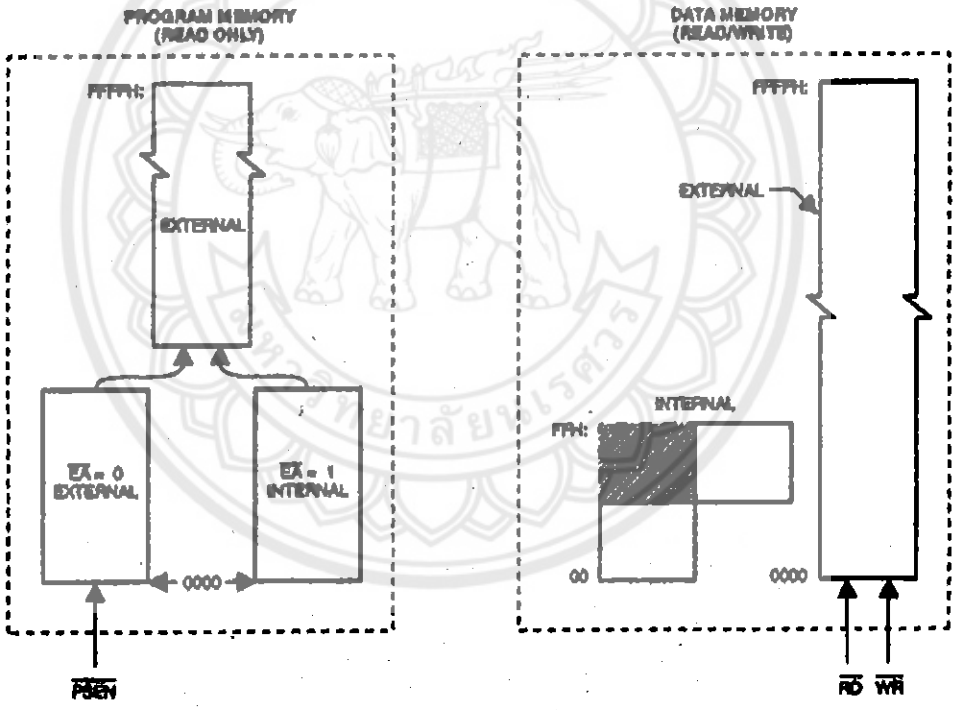
รูปที่ 2.4 ไดอะแกรมผังเวลาลำดับการ Fetch และ Execute



รูปที่ 2.5 ไดอะแกรมผังเวลาเมื่อใช้คำสั่งติดต่อหน่วยความจำภายนอก

หน่วยความจำโปรแกรม

AT89C52มีการจัดหน่วยความจำโดยแยกพื้นที่ของโปรแกรมและข้อมูลออกจากกัน กล่าวคือ หน่วยความจำโปรแกรมจะไม่สามารถเขียนข้อมูลลงได้ ใช้เป็นพื้นที่เก็บรหัสคำสั่ง ในขณะที่หน่วยความจำข้อมูลจะใช้เก็บตัวแปร และข้อมูลต่างๆ สามารถเขียนและอ่านได้ โครงสร้างของหน่วยความจำโปรแกรม แสดงดังรูปที่ 2.6 เป็นขาสัญญาณควบคุมการอ่านรหัสคำสั่ง ถ้าเราให้ EA ต่อเข้ากับ +Vcc ตำแหน่งที่ CPU เริ่มอ่านคำสั่งแรกจะเป็นตำแหน่ง 0000H โปรแกรมเคาน์เตอร์จะเพิ่มค่า เพื่ออ่านคำสั่งต่อไป จนถึงตำแหน่ง 1FFFH ซึ่งเป็นพื้นที่ของหน่วยความจำแฟลชขนาด 8 กิโลไบท์นั่นเอง กรณีที่โปรแกรมมีขนาดใหญ่กว่าหน่วยความจำโปรแกรมภายใน CPU จะอ่านรหัสคำสั่งที่เก็บอยู่ในพื้นที่ส่วนที่เหลือคือตำแหน่ง 2000H ถึง FFFFH แต่ถ้าเราให้ EA ต่อลงกราวด์หรือเป็นลอจิกศูนย์ CPU จะอ่านรหัสคำสั่งจากหน่วยความจำโปรแกรมภายนอกตำแหน่ง 0000H ถึง FFFFH โดยใช้สัญญาณ PSEN ควบคุมการอ่าน

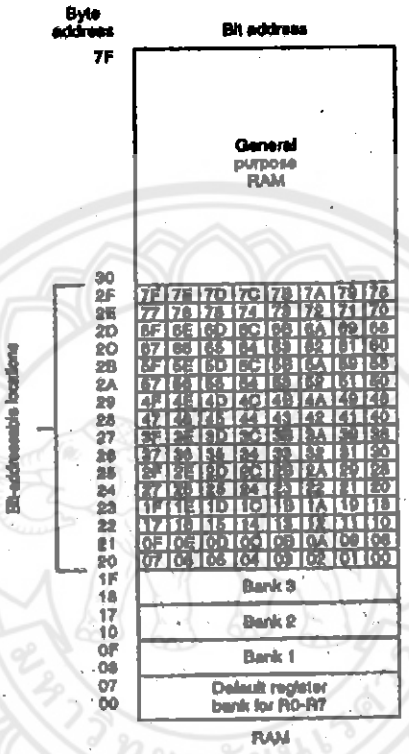


รูปที่ 2.6 โครงสร้างของหน่วยความจำโปรแกรมและหน่วยความจำข้อมูล

หน่วยความจำข้อมูล

AT89C52 แบ่งหน่วยความจำข้อมูลออกเป็นสองพื้นที่ คือ พื้นที่ของหน่วยความจำข้อมูลบนชิพ ขนาด 256ไบท์ ตำแหน่ง 00H ถึง FFH และหน่วยความจำข้อมูลภายนอก ตำแหน่ง 0000H ถึง FFFFH หรือมีขนาด 64กิโลไบท์ การอ่านหน่วยความจำข้อมูลภายนอก CPU จะกำเนิดสัญญาณ RD ส่วนการเขียนข้อมูล CPU จะกำเนิดสัญญาณ WR พื้นที่ของหน่วยความจำข้อมูลบนชิพ AT89C52

จะพิเศษกว่าหน่วยความจำข้อมูลภายนอก กล่าวคือ ตำแหน่ง 00H ถึง 1FH จะเป็นพื้นที่ของรีจิสเตอร์ใช้งานทั่วไป R0, R1, R2, R3, R4, R5, R6,R7 จำนวน 4 ชุด ตำแหน่ง 20H - 2FH เป็นพื้นที่ที่ใช้คำสั่งประมวลผลแบบบิท โดยเราสามารถระบุตำแหน่งข้อมูลเป็นบิทได้ มีค่าตำแหน่งบิทเป็น 00H - 7FH เช่นที่ตำแหน่งไบท์20H ภายในมีตำแหน่งบิทเป็น 00H -07H เป็นต้น พื้นที่ตำแหน่ง 30H-7FH เป็นพื้นที่ใช้งานทั่วไป



รูปที่ 2.7 ผังหน่วยความจำข้อมูลและการแบ่งพื้นที่ตำแหน่ง 00H – 7FH

ส่วนตำแหน่ง 80H- 0FFH นั้น AT89C52สามารถใช้งานได้โดยการอ้างตำแหน่งผ่าน R0 และ R1 เช่น MOV A,@R0 ข้อมูลใน R0 จะใช้เป็นตำแหน่งของแรมกรณีที่อ้างโดยตรง เช่น MOV A,90H จะทำการอ่านข้อมูลจากรีจิสเตอร์หน้าที่พิเศษตำแหน่ง 90H
(ตัวเลขตำแหน่งเขียนเป็น 90H เป็นเลขฐานสิบหก เราใช้เลขฐานสิบหกแทนเลขฐานสองเพื่อย่นย่อการเขียนและการอธิบาย กรณีที่เราใช้อธิบายตำแหน่ง 90H นั้นหมายถึง AT89C52 จะใช้ A0-A7 หรือสายแอดเดรสจำนวนแปดเส้นใช้เพื่อชี้ตำแหน่ง 10010000 เป็นต้น)

โปรแกรมมิ่งรีจิสเตอร์ (Programming Registers)

รีจิสเตอร์เป็นหน่วยความจำเช่นเดียวกับหน่วยความจำข้อมูล AT89C52 มีโปรแกรมเคาน์เตอร์ PC ที่ใช้พื้นที่แยกจากหน่วยความจำข้อมูล ส่วนรีจิสเตอร์ A, SP, PSW (Program Status Word) และตัวที่เหลือ AT89C52 ใช้พื้นที่ของหน่วยความจำข้อมูลทั้งสิ้น โปรแกรมมิ่งรีจิสเตอร์ใช้

เป็นตัวเก็บข้อมูลก่อนและหลังการประมวลผลที่หน่วยประมวลผล ALU สถานะทางคณิตศาสตร์และตรรกศาสตร์จะเก็บไว้ใน PSW คำสั่งที่ใช้เขียนโปรแกรมก็จะประกอบไปด้วยรีจิสเตอร์ต่างๆ ทั้งสิ้น AT89C52 เรียกพื้นที่ของหน่วยความจำข้อมูลตำแหน่ง 80H - 0FFH ว่าเป็นรีจิสเตอร์หน้าที่พิเศษ(Special Function Registers)

PC (Program Counter)

PC เป็นรีจิสเตอร์ขนาด 16 บิท ใช้เป็นตัวชี้ตำแหน่งของรหัสคำสั่ง จะมีค่าเป็น 0000H เมื่อรีเซ็ท CPU

Accumulator (0E0H)

ACC เป็นแอคคิวมูเลเตอร์รีจิสเตอร์ขนาด 8 บิท ใช้พักข้อมูลก่อนและหลังการประมวลผลทางคณิตศาสตร์และตรรกศาสตร์ ในโปรแกรมเราสามารถเขียนสั้นเป็น A ตำแหน่งของแอคคิวมูเลเตอร์รีจิสเตอร์อยู่ที่แอดเดรส 0E0H สามารถเข้าถึงข้อมูลเป็นบิทได้ ตำแหน่งบิทคือ E0H - E7H สำหรับบิท 0- 7

ตัวอย่างการใช้แอคคิวมูเลเตอร์รีจิสเตอร์

```
MOV A,#30H      ; คัดลอกค่าคงที่ 30H เข้าไปยังแอคคิวมูเลเตอร์
MOV A,30H       ; คัดลอกข้อมูลในหน่วยความจำตำแหน่ง 30H เข้าไปยังแอคคิวมูเลเตอร์
MOV C,ACC.7     ; คัดลอกข้อมูลบิท 7 ของแอคคิวมูเลเตอร์ไปยังแฟลกตัวทด
SETB ACC.0      ; เซ็ทบิท 0 ของแอคคิวมูเลเตอร์ให้เป็น 1
```

B (0F0H)

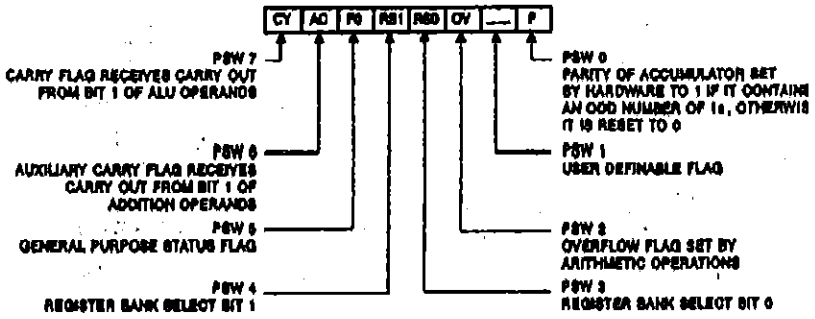
B เป็นรีจิสเตอร์ขนาด 8 บิท ใช้ในคำสั่งคูณและหาร ตำแหน่งของ B รีจิสเตอร์อยู่ที่แอดเดรส 0F0H สามารถเข้าถึงข้อมูลเป็นบิทได้ ตำแหน่งบิทคือ 0F0H - 0F7H สำหรับบิท 0- 7

```
MUL AB          ; A x B ผลคูณเก็บไว้ที่ B:A
DIV AB          ; A/B ผลหารจำนวนเต็มเก็บที่ A เศษที่เหลือเก็บที่ B
```

PSW (0D0H)

PSW เป็นรีจิสเตอร์ขนาด 8 บิท ใช้เก็บสถานะภายหลังกระทำคำสั่งทางคณิตศาสตร์และตรรกศาสตร์ สถานะที่เกิดขึ้นจะเก็บเป็นข้อมูลหนึ่งบิทหรือใช้เรียกเป็นแฟลกบ่งสถานะ เช่น CY

แฟลกตัวทด เป็นต้น นอกจากนี้บางบิทยังใช้ทำหน้าที่อื่นๆเช่น RS0 RS1 ใช้เลือกกลุ่มของ R0-R7 ส่วน F0 เป็นแฟลกใช้งานทั่วไป



รูปที่ 2.8 Program Status Word Register

SP (81H)

SP เป็นรีจิสเตอร์ขนาด 8 บิทใช้บ่งตำแหน่งของหน่วยความจำที่เป็นพื้นที่สแตกมีการใช้งานแบบชนิดเข้าทีหลังออกก่อน เรียกว่า FILO ย่อมาจาก First In Last Out การทำงานของโปรแกรมที่มีการเรียกโปรแกรมย่อยด้วยคำสั่ง CALL นั้น ตำแหน่งแอดเดรสถัดจากคำสั่ง CALL จะได้รับการเก็บลงบนหน่วยความจำข้อมูลบนชิป 89C52 ที่ตำแหน่งระบุอยู่ใน SP รีจิสเตอร์ เมื่อโปรแกรมย่อยทำงานเสร็จ คำสั่ง RET จะคืนค่าตำแหน่งถัดจากคำสั่ง CALL ให้กับโปรแกรมเคาน์เตอร์ ปกติเมื่อรีเซ็ท CPU จะทำให้ SP มีค่าเท่ากับ 07H เมื่อมีการใช้งานตำแหน่งเริ่มต้นของพื้นที่สแตกจะเป็น 08H กล่าวคือสแตกจะเพิ่มตัวเองขึ้นหนึ่งก่อนใส่ข้อมูลหนึ่งไบท์ และดึงข้อมูล กลับขึ้นมาก่อนจึงจะลดค่าตัวเองลงหนึ่ง นอกจากการใช้สแตกรีจิสเตอร์ในการเรียกโปรแกรมย่อยแล้ว SP ยังใช้ในคำสั่ง PUSH และ POP ใส่ข้อมูลและดึงข้อมูลไดเร็กไบท์อีกด้วย

DPTR (83H-82H)

DPTR เป็นรีจิสเตอร์ขนาด16 บิท ประกอบด้วย DPH และ DPL มีตำแหน่งอยู่ที่ 83H และ 82H ตามลำดับDPTR ย่อมาจาก Data Pointer Register ใช้สำหรับการอ้างอิงหน่วยความจำเพื่อการอ่านและเขียนข้อมูล

```
MOV DPTR,#8100H        ; ใส่ค่าคงที่ 16 บิท 8100H ลงใน DPTR
MOVX A,@DPTR           ; อ่านข้อมูลจากหน่วยความจ ำข้อมูลภายนอกที่ต ำแหน่ง 8100H
                         มาใส่ใน แอคคิวมูเลเตอร์
MOVC A,@A+DPTR         ; อ่านข้อมูลจากหน่วยความจ ำโปรแกรมที่ต ำแหน่ง 8100H+(A)
                         มาใส่ในแอคคิวมูเลเตอร์
```

PORT P0 (80H), P1(90H), P2(0A0H), P3(0B0H)

P0, P1, P2 และ P3 เป็นรีจิสเตอร์พิเศษต่างจากรีจิสเตอร์ที่กล่าวมาข้างต้นตรงที่ข้อมูลที่อ่านและเขียนที่รีจิสเตอร์เหล่านี้สามารถวัดได้ด้วยลอจิกโพรบที่ขาของชิป 89C52 โดยตรง กล่าวอีกนัยหนึ่ง รีจิสเตอร์เหล่านี้เป็นเส้นทางติดต่อระหว่าง CPU กับโลกภายนอกนั่นเอง เราเรียกรีจิสเตอร์เหล่านี้เป็นอินพุทเอาท์พุทพอร์ทก็ได้ พอร์ททั้งสี่นี้เป็นชนิดสองทางแต่ละพอร์ทประกอบด้วยแลทช์วงจรขับภาคเอาท์พุท และวงจรอินพุทบัพเฟอร์ เอาท์พุทแต่ละบิททำงานด้วย D Flip-Flop สามารถคงค่าที่เขียนมายังพอร์ทได้ ส่วนการใช้เป็นอินพุทพอร์ท คำสั่งบางคำสั่งจะอ่านค่าจากเอาท์พุท Q ของฟลิบฟลอบ บางคำสั่งจะอ่านจากขาอินพุทโดยตรงพอร์ททั้งสี่สามารถเข้าถึงข้อมูลเป็นบิทได้ มีตำแหน่งไบท์แอดเดรสเป็น 80H 90H 0A0H และ 0B0H สำหรับP0, P1, P2 และ P3 ตามลำดับ

```
MOV P1, #00000001B      ; เขียนข้อมูล 8 บิท 00000001 ไปยังพอร์ท P1
MOV P3.7,C              ; คัดลอกข้อมูลในแฟลกตัวทดไปยังพอร์ท P3 บิท 7
MOV C,P1.0              ; อ่านข้อมูลบิท 0 ของพอร์ท P1 เก็บไว้ในแฟลกตัวทด
CPL P1.7                ; คอมพลีเม้นส์บิท 7 ของพอร์ท P1
```

TH0 (8CH), TL0 (8AH)

Timer 0 เป็นรีจิสเตอร์ขนาด16 บิท ประกอบด้วย TH0 และ TL0 รับสัญญาณนาฬิกาจากภายนอกที่ขา T0 หรือรับสัญญาณนาฬิกาจากออสซิลเลเตอร์ภายใน (X-Tal/12) การนับเป็นแบบนับเพิ่มทีละหนึ่ง เมื่อค่าที่นับเกิดล้น (Overflow) กล่าวคือ จาก FFFF เป็น 0 บิท TF0 ไทเมอร์โอเวอร์โฟล์แฟลกจะเซ็ทเป็น 1

TH1 (8DH), TL1 (8BH)

Timer 1 เป็นรีจิสเตอร์ขนาด16 บิท ประกอบด้วย TH1 และ TL1 มีการทำงานเช่นเดียวกับไทเมอร์ 0 TMOD (89H), TCON (88H) TMOD เป็นรีจิสเตอร์ใช้เลือกโหมดการทำงานของไทเมอร์ 0 และ ไทเมอร์ 1 ส่วน TCON เป็นรีจิสเตอร์ควบคุมการทำงานของไทเมอร์ทั้งสอง

SBUF (99H), SCON (98H)

SBUF เป็นรีจิสเตอร์ขนาด 8 บิทใช้พักข้อมูลเพื่อการรับและส่งข้อมูลแบบอนุกรม ข้อมูลจะส่งออกทางขาTXD และรับทางขา RXD ส่วน SCON เป็นรีจิสเตอร์กำหนดโหมดและควบคุมการทำงานของการรับส่งข้อมูลแบบอนุกรม

TH2 (0CDH), TL2 (0CCH)

Timer 2 เป็นรีจิสเตอร์ขนาด16 บิท ประกอบด้วย TH2 และ TL2 มีการทำงานเช่นเดียวกับไทเมอร์ 0 และ1 และมีฟังก์ชั่นพิเศษทำงานร่วมกับเคพเจอร์รีจิสเตอร์ (Capture Register) ในโหมดเคพเจอร์ ถ้าสัญญาณที่ป้อนให้กับขา T2EX เปลี่ยนจาก 1-เป็น-0 และบิท EXEN2 เซ็ท ค่านับขณะนั้นจะถูกเคพเจอร์หรือจับไว้เก็บลงในเคพเจอร์รีจิสเตอร์ พร้อมกันนี้การเปลี่ยนแปลง 1 เป็น 0 นี้จะเซ็ท EXF2 เป็น 1 ถ้า TF2 เซ็ทจะกำเนิดอินเทอร์รัพท์ไทเมอร์ 2

T2MODE (0C9H), T2CON(0C8H)

เช่นเดียวกับไทเมอร์ 0 และ 1 ไทเมอร์ 2 ก็จะทำงานตามโหมดที่เลือกไว้ใน T2MODE และควบคุมการทำงานโดย T2CON

RCAP2H (0CBH), **RCAP2L** (0CAH)

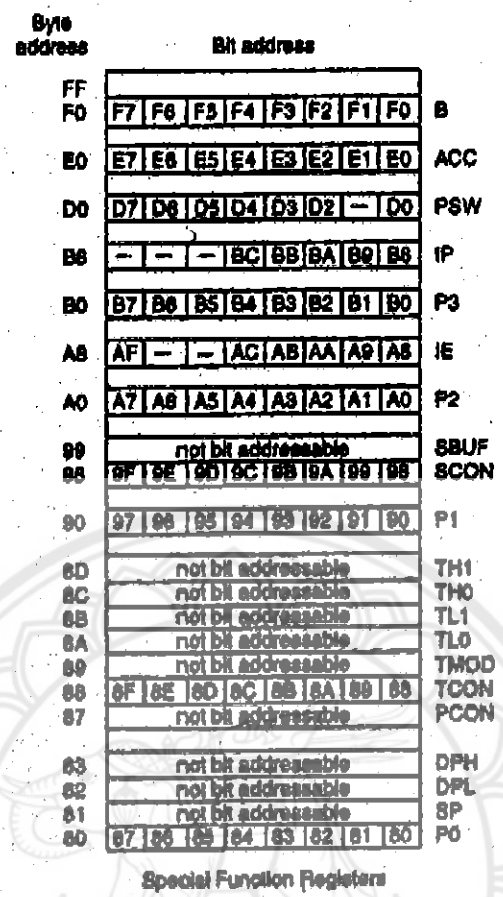
เป็นเคพเจอร์รีจิสเตอร์ขนาด 16 บิท ประกอบด้วยไบท์สูง RCAP2H และไบท์ต่ำ RCAP2L ในโหมดเคพเจอร์ เมื่อมีการเปลี่ยนแปลงจาก 1- เป็น-0 ที่ขา T2EX ข้อมูลของ TH2 และ TL2 จะถูกคัดลอกลงในRCAP2H และ RCAP2L ในโหมด 16 บิท AUTO-RELOAD ค่าของ RCAP2H และ RCAP2L จะเก็บค่าที่จะใช้ป้อนให้กับ TH2 และ TL2 เมื่อเกิดการล้นจาก FFFFH เป็น 0000H

IP, IE, PCON

IP (Interrupt Piority) เป็นรีจิสเตอร์ควบคุมลำดับความสำคัญของการอินเทอร์รัพท์

IE (Interrupt Enable) เป็นรีจิสเตอร์ควบคุมการยอมให้เกิดอินเทอร์รัพท์ได้หรือไม่ได้

PCON (Power Control) เป็นรีจิสเตอร์ควบคุมโหมดประหยัดพลังงาน



รูปที่ 2.9 ผังหน่วยความจำแสดงตำแหน่งไบท์และตำแหน่งบิทที่ใช้เป็นพื้นที่ของรีจิสเตอร์หน้าที่พิเศษเช่น ACC.7 มีค่าตำแหน่งบิทเป็น 0E7H, P1.7 มีค่าตำแหน่งบิทเป็น 97H เป็นต้น

แอดเดรสซิ่งโหมด (Addressing Modes)

การอ้างถึงข้อมูลนี้เรียกว่าแอดเดรสซิ่งโหมด AT89C52 มีแอดเดรสซิ่งโหมดทั้งหมด 5 แบบดังต่อไปนี้

IMMEDIATE

ในโหมดนี้ค่าคงที่จะอยู่ที่ไบท์ถัดมาของรหัสคำสั่ง เราสามารถใส่ข้อมูลค่าคงที่เข้าไปยังรีจิสเตอร์ หรือหน่วยความจำได้ทันที เครื่องหมาย # เป็นตัวระบุว่าเป็นค่าคงที่

```
MOV A,#00000010B    ; ใส่ข้อมูล 8 บิท 00000010 ลงในแอคคิวมูเลเตอร์
MOV DPTR,#9000H     ; ใส่ข้มูล 16 บิท 9000H ลงใน DPTR
MOV 30H,#30H        ; ใส่ข้อมูล 8 บิท 30H ลงในหน่วยความจำข้อมูลที่ตำแหน่ง 30H
MOV P1,#01111111B   ; ใส่ข้อมูล 8 บิท 011111111 ลงในพอร์ท P1
```

XRL P1,#0FFH ; เอ็กคลูซีฟออร์พอร์ท P1 ด้วยค่าคงที่ 0FFH

ADD A,#100 ; บวกค่าคงที่ 100 กับแอคคิวมูเลเตอร์ผลลัพธ์เก็บที่ A

ตัวเลขที่ระบุในคำสั่งได้ใช้รูปแบบที่ตัวแปลแอสเซมเบลอร์ทั่วๆไป เข้าใจกล่าวคือ

100, 10, 5	เป็นเลขฐานสิบ
30H, 9000H, 0FFH	เป็นเลขฐานสิบหก กรณีที่ขึ้นต้นด้วย A - F เราต้องใส่ 0 ไว้บอก แอสเซมเบลอร์ว่า เป็นค่าคงที่ตัวเลข มิใช่ลาเบล (Label)
01111111B	เป็นเลขฐานสอง

DIRECT

เราสามารถอ้างถึงข้อมูลโดยระบุเป็นตำแหน่งของหน่วยความจำข้อมูลบนชิพและรีจิสเตอร์หน้าที่พิเศษในรหัสคำสั่งไบท์ถัดมาจะเป็นตำแหน่งดังกล่าว สังเกตเครื่องหมาย # จะไม่มีในคำสั่ง

MOV A,30H ; คัดลอกข้อมูลที่อยู่ในหน่วยความจำข้อมูลแอดเดรส 30H ลงในแอคคิวมูเลเตอร์

MOV 90H,0E0H ; คัดลอกข้อมูลใน SFR แอดเดรส 0E0H ลงใน SFR แอดเดรส 90H

INC 31H ; เพิ่มค่าของข้อมูลในตำแหน่ง 31H ขึ้นหนึ่ง

SUBB A,0AH ; ลบข้อมูลในแอคคิวมูเลเตอร์ด้วยข้อมูลอยู่ในตำแหน่ง 0AH

PUSH 7 ; เพิ่มค่าตัวชี้สแตก SP ขึ้นหนึ่ง แล้วเก็บข้อมูลที่อยู่ในหน่วยความจำแอดเดรส 07 ลงในหน่วยจำพื้นที่สแตกที่ชี้โดยตำแหน่งระบุใน SP รีจิสเตอร์

นอกจากตำแหน่งไบท์ที่ระบุในคำสั่งโดยตรงแล้ว เรายังสามารถอ้างถึงข้อมูลที่เป็นบิทได้อีกด้วย

MOV P1.7,C ; คัดลอกข้อมูลหนึ่งบิทในแฟลกตัวทดลงในพอร์ท P1 บิท 7

JBC 00H,LOOP ; ตรวจสอบข้อมูลหนึ่งบิทที่ตำแหน่งบิท 00H (ตำแหน่งไบท์ 20H บิท 0) ว่าเป็นหนึ่งหรือศูนย์ ถ้าเป็นหนึ่ง จะเคลียร์เป็นศูนย์แล้วกระโดดไปตำแหน่ง LOOP แต่ถ้าเป็นศูนย์จะทำคำสั่งถัดมา

INDIRECT

ในโหมดนี้ตำแหน่งของข้อมูลใช้อ้างถึงแบบทางอ้อมผ่านรีจิสเตอร์ R0 และ R1 สำหรับตำแหน่ง 8 บิท ส่วนDPTR ใช้กับตำแหน่ง 16 บิท เราสามารถอ้างถึงข้อมูลแบบทางอ้อมนี้กับหน่วยความจำข้อมูลทั้งที่อยู่บนชิปและภายนอก สังเกตุเครื่องหมาย @ ในคำสั่ง

```
MOV R0,#30H       ; ใส่ค่าคงที่ 8 บิท 30H ลงใน R0
MOV A,@R0         ; คัดลอกข้อมูลที่อยู่ในหน่วยความจำข้อมูลแอดเดรสระบุโดย R0 ซึ่งก็คือ
                    30H ลงในแอคคิวมูเลเตอร์
MOV R1,#0FFH      ; ใส่ค่าคงที่ 8 บิท 0FFH ลงใน R1
MOVX A,@R1        ; คัดลอกข้อมูลที่อยู่ในหน่วยความจำข้อมูลภายนอกแอดเดรสระบุโดย R1
                    ซึ่งก็คือ0FFH ลงในแอคคิวมูเลเตอร์
MOV DPTR,#9000H   ; ใส่ค่าคงที่ 16 บิท 9000H ลงใน DPTR
MOV A,#10H        ; ใส่ค่าคงที่ 8 บิท 10H ลงใน A
MOVX @DPTR,A      ; เขียน 10H ลงในหน่วยความจำข้อมูลภายนอกตำแหน่ง 9000H
```

REGISTER

ในโหมดนี้ตำแหน่งของข้อมูลใช้อ้างถึงแบบโดยตรงเช่นเดียวกับDIRECT ADDRESSING แต่ใช้กับรีจิสเตอร์ใช้งานทั่วไป R0-R7 โดยเลือกชุดที่ใช้งานปัจจุบันด้วยการกำหนดบิท RS0-RS1 ใน PSW ข้อดีเมื่อเปรียบเทียบกับ DIRECT คือรหัสคำสั่งจะมีเพียงหนึ่งไบท์

```
E500  MOV A,00H   ; คัดลอกข้อมูลที่อยู่ในหน่วยความจำข้อมูลตำแหน่ง 00H ลงในแอคคิวมู
                    เลเตอร์
E8    MOV A,R0    ; คัดลอกข้อมูลในรีจิสเตอร์ R0 ลงในแอคคิวมูเลเตอร์
```

แต่การใช้REGISTER ADDRESSING รหัสคำสั่งเป็น E8 ในขณะที่โหมด DIRECT รหัสจะเป็น E500

```
ADD  A,R7         ; บวกข้อมูลใน R7 กับ A ผลลัพธ์เก็บใน A
```

INDEXED ADDRESSING

คำสั่งที่ใช้ในโหมดสำหรับการเคลื่อนย้ายข้อมูลนี้มีเพียงสองคำสั่งคือMOVC A,@A+DPTR และ MOVCA,@A+PC จะเห็นว่าเป็นคำสั่งอ่านอย่างเดียวทั้งสองคำสั่ง กล่าวคือเราสามารถใช้กับหน่วยความจำโปรแกรมเท่านั้น วัตถุประสงค์ของโหมดอินเด็กซ์นี้จะใช้สำหรับการอ่านข้อมูลจากข้อมูลที่เก็บเป็นตาราง โดยตัวชี้ DPTRหรือ PC เป็นแอดเดรสของตำแหน่งเริ่มต้นของตาราง ส่วนตัวชี้ตำแหน่งเป็นข้อมูลในแอคคิวมูเลเตอร์ตำแหน่งที่จะถูกอ่านขึ้นมาจะได้จากการบวก DPTR หรือ PC กับ Aโหมดอินเด็กซ์นี้ยังใช้กับคำสั่งกระโดดไปยังตารางคำสั่งกระโดด JMP @A+DPTR เช่นเดียวกัน DPTR เป็นตำแหน่งเริ่มต้นของตารางตำแหน่งที่จะกระโดดคำนวณจากผลบวกของข้อมูลใน A กับ DPTR

```
MOV DPTR,#9000H          ; ใส่ข้อมูลค่าคงที่ 16 บิท 9000H ลงใน DPTR
MOV A,#10                ; ใส่ข้อมูลค่าคงที่ 8 บิท 0AH ลงในแอคคิวมูเลเตอร์
MOVC A,@A+DPTR           ; อ่านข้อมูลจากหน่วยความจำโปรแกรมตำแหน่ง 900AH ใส่ใน
                           A
MOV DPTR,#JUMP_TABLE     ; ใส่ค่าตำแหน่งเริ่มต้นของตารางกระโดดใน DPTR
MOV A,INDEX_NUMBER       ; อ่านข้อมูลจากตำแหน่ง INDEX_NUMBER เข้าไว้ใน A
                           +RL A ; A = A * 2
JMP @A+DPTR              ; กระโดดไปยังตารางกระโดด
JUMP_TABLE:
AJMP CASE_0              ; กรณีที่ A = 0
AJMP CASE_1              ; กรณีที่ A = 1
AJMP CASE_2              ; กรณีที่ A = 2
AJMP CASE_3              ; กรณีที่ A = 3
```

สแตกและโปรแกรมย่อย (The Stack and Subroutines)

สแตกเป็นพื้นที่ที่จองไว้ในหน่วยความจำแรม ใช้สำหรับเก็บข้อมูลโดยการเขียนข้อมูลหรืออ่านจะกระทำผ่าน SP รีจิสเตอร์ หรือตัวชี้สแตกค่าที่อยู่ใน SP เป็นตำแหน่งที่ชี้ไปยังส่วนบนสุดของสแตก (Top of Stack) AT89C52 มีตัวชี้สแตกขนาด 8 บิตคือ SP เมื่อรีเซ็ตซิพจะทำให้ค่าในสแตกมีค่าเท่ากับ 07H เราสามารถกำหนดให้ไปชี้ที่ตำแหน่งอื่นได้โดยการใส่ค่าคงที่ที่เป็นตำแหน่งใหม่ได้ด้วยคำสั่ง

```
MOV SP,#60H
```

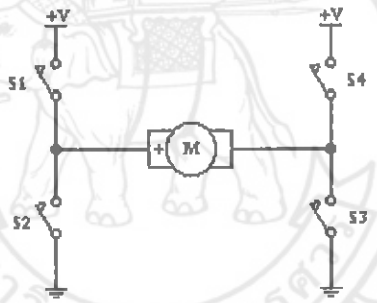
ตัวอย่างการเก็บข้อมูลลงในสแตกจะใช้คำสั่ง PUSH direct ส่วนการดึงข้อมูลกลับขึ้นมาจะใช้คำสั่ง POP direct เป็นต้น

รหัสคำสั่งและชุดคำสั่ง (Machine Code and Instruction Set)

รหัสคำสั่งได้รับการเรียกขานหลายๆแบบเป็นต้นว่า HEX CODE หรือ OPCODE รหัสคำสั่งเป็นข้อมูลเลขฐานสอง โดยมีขนาดความกว้างปรกติเท่ากับขนาดของบัสข้อมูล เช่น AT89C52 เป็นไมโครคอนโทรลเลอร์ 8 บิท ขนาดความกว้างของรหัสคำสั่งก็จะเท่ากับ 8 บิท เป็นต้น ส่วนจำนวนไบท์ของแต่ละคำสั่งก็จะแตกต่างกัน เช่น บางคำสั่งมีเพียงหนึ่งไบท์ บางคำสั่งมีสองหรือสามไบท์ เป็นต้น

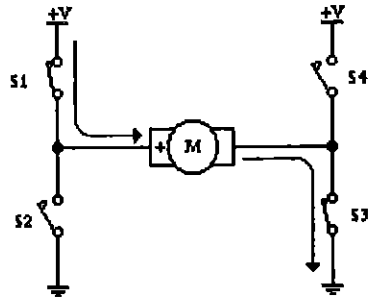
2.2 การควบคุมทิศทางการหมุนของ DC Motor

จะนำหลักการทำงานของวงจร H-Bridge Switch มาใช้ หลักการของวงจร H-Bridge นั้นจะประกอบไปด้วย สวิทช์ 4 ตัว คือ S1 ,S2 ,S3 และ S4 ในรูป ใช้ DC-Motor เป็น Load ของวงจร



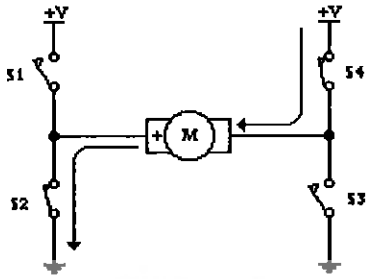
รูปที่ 2.10 รูปจำลองของวงจร H-Bridge

ในสภาวะเริ่มต้น สวิทช์ ทุกตัว Off อยู่ จะไม่มีกระแสไฟฟ้าไหลเข้าสู่มอเตอร์ (รูปบน) และเมื่อเราทำการ On สวิทช์ S1 และ S3 พร้อมกัน จะเป็นการเชื่อมวงจร ทำให้มีกระแสไฟฟ้า ไหลผ่านมอเตอร์ จากขั้วบวกของมอเตอร์ ไปยังขั้วลบของมอเตอร์ จึงทำให้มอเตอร์สามารถหมุนได้ ในทิศทาง Forward



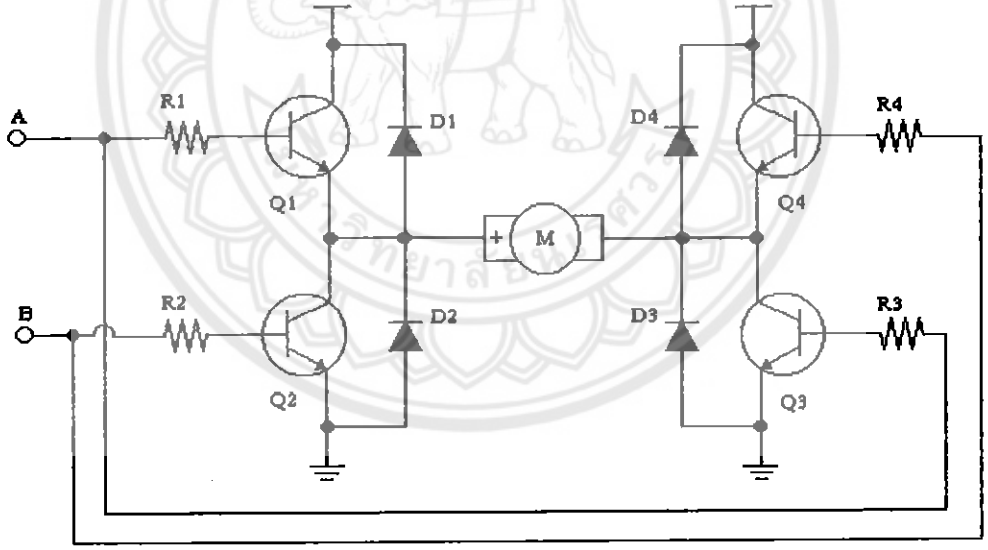
รูปที่ 2.11 วงจรขณะสวิตช์ S1 และ S3 ปิดวงจร

และในทางกลับกัน ถ้าหากเราทำการ On สวิทช์ S2 และ S4 พร้อมกัน ก็จะเป็นการเชื่อมวงจร และทำให้เกิดกระแสไฟฟ้า ไหลผ่านมอเตอร์ จากขั้วลบของมอเตอร์ ไปยังขั้วบวกของมอเตอร์ จึงทำให้มอเตอร์สามารถหมุนได้ และเป็นการหมุนในทิศทาง Reverse (กลับทิศทางกับกรณีแรก)



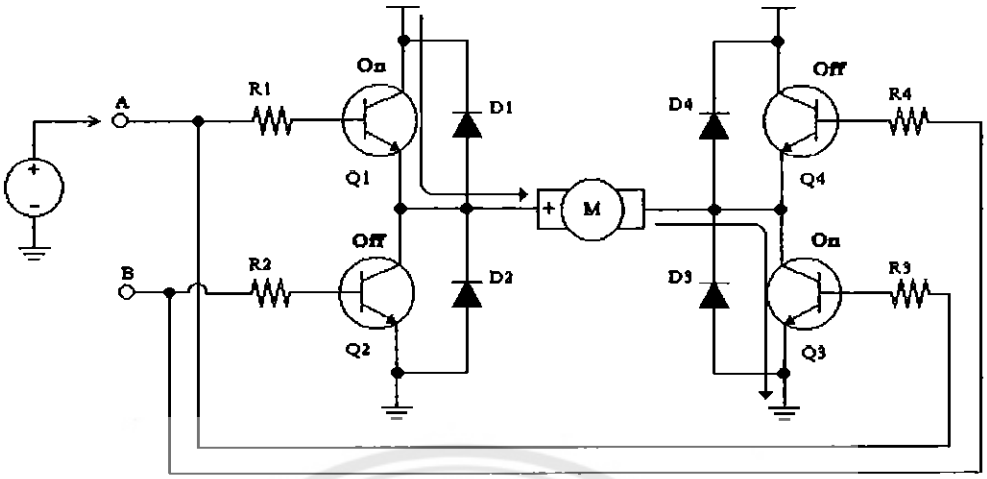
รูปที่ 2.12 วงจรขณะสวิตช์ S2 และ S4 ปิดวงจร

จากหลักการทำงานเบื้องต้นในการควบคุมทิศทางการหมุนของมอเตอร์โดยใช้สวิตช์ เราสามารถใช้อุปกรณ์สารกึ่งตัวนำประเภททรานซิสเตอร์ แทนได้ โดยต่อในลักษณะใช้เป็นสวิตช์



รูปที่ 2.13 วงจร H-Bridge ที่ใช้ทรานซิสเตอร์แทนสวิตช์

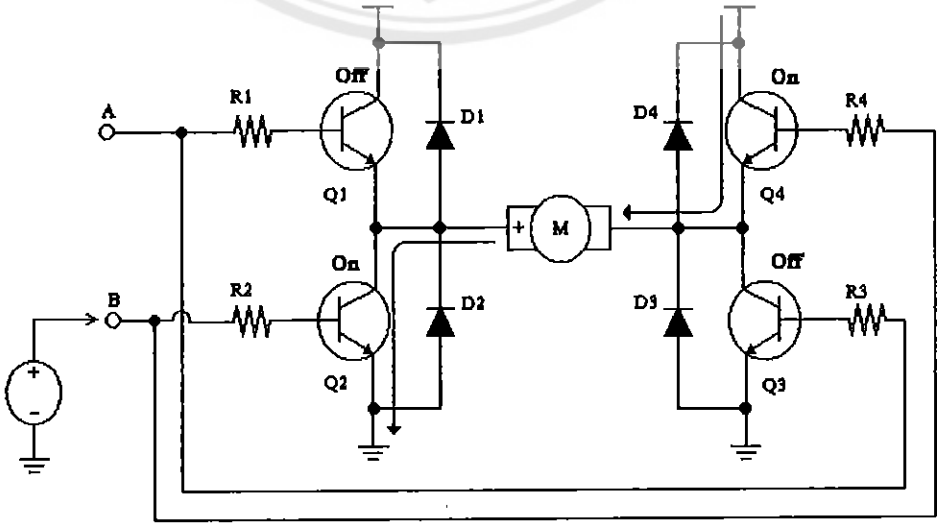
กรณีที่ Q1 และ Q3 ทำงาน



รูปที่ 2.14 วงจรขณะ Q1 และ Q3 ทำงาน

เมื่อมีการจ่ายแรงดัน เข้าที่จุด A ทำให้มีกระแสไหลผ่าน R1 เข้าสู่ขา B ของ Q1 และมีกระแสไหลผ่าน R3 เข้าสู่ขา B ของ Q3 ทำให้ Q1 และ Q3 ทำงาน (On) เปรียบเสมือนสวิทช์ปิดวงจร ส่งผลให้มีกระแสไหลจากแหล่งจ่าย ผ่านขา C และ ของ Q1 ผ่านเข้าสู่ขั้วบวก (+) ของมอเตอร์ ผ่านไปยังขา C และ E ของ Q3 ทำให้มีกระแสไหลผ่านมอเตอร์ในทิศทางบวก และครบวงจร จึงทำให้มอเตอร์สามารถหมุน ในทิศทาง Forward ได้

กรณีที่ Q2 และ Q4 ทำงาน



รูปที่ 2.15 วงจรขณะ Q2 และ Q4 ทำงาน

เมื่อมีการจ่ายแรงดัน เข้าที่จุด B ทำให้มีกระแสไหลผ่าน R2 เข้าสู่ขา B ของ Q2 และมีกระแสไหลผ่าน R4 เข้าสู่ขา B ของ Q4 ทำให้ Q2 และ Q4 ทำงาน (On) เปรียบเสมือนสวิทช์ปิดวงจร ส่งผลให้มีกระแสไหลจากแหล่งจ่าย ผ่านขา C และ E ของ Q4 ผ่านเข้าสู่ขั้วลบ (-) ของมอเตอร์ ผ่านไปยังขา C และE ของ Q2 ทำให้มีกระแสไหลผ่านมอเตอร์ในทิศทางลบ และครบวงจร จึงทำให้มอเตอร์สามารถหมุน ในทิศทาง Reverse ได้

ส่วนไดโอดที่ต่อคร่อมกับขา C และ E ของทรานซิสเตอร์แต่ละตัวนั้น เรียกว่า Free Wheeling Diode ต่อไว้เพื่อลดผลของกระแสกระชากของมอเตอร์ ซึ่งอาจจะทำให้วงจรเสียหายได้

2.3 เซ็นเซอร์ตรวจจับวัตถุ

หลักการทำงานคือ ส่งสัญญาณอินฟราเรดไปกระทบกับวัตถุแล้วสะท้อนกลับมายังตัวรับ แล้วนำค่าที่ได้ไปประมวลผลต่อไป

เซ็นเซอร์ที่ใช้เป็นเซ็นเซอร์อินฟราเรดโมดูล เบอร์ TCRT5000

<u>คุณสมบัติ</u>

มีระยะตรวจจับประมาณ 12 cm

มีทั้งตัวส่งและตัวรับอยู่ในโมดูลเดียวกัน

ต่อวงจรร่วมภายนอกไม่มากนักก็สามารถนำไปใช้งานได้

<u>จุดเด่น</u>

1. มีความไวสูง
2. สามารถปรับระยะการตรวจจับได้โดยการต่อความต้านทานภายนอกเพิ่มเติม

<u>จุดด้อย</u>

1. เซ็นเซอร์แต่ละตัวจะมีความไวไม่เท่ากัน
2. เซ็นเซอร์จะมีความไวต่อแสงธรรมชาติ ทำให้การทำงานของวงจรไม่แน่นอนหากอยู่ในที่สว่างมากๆ

บทที่ 3

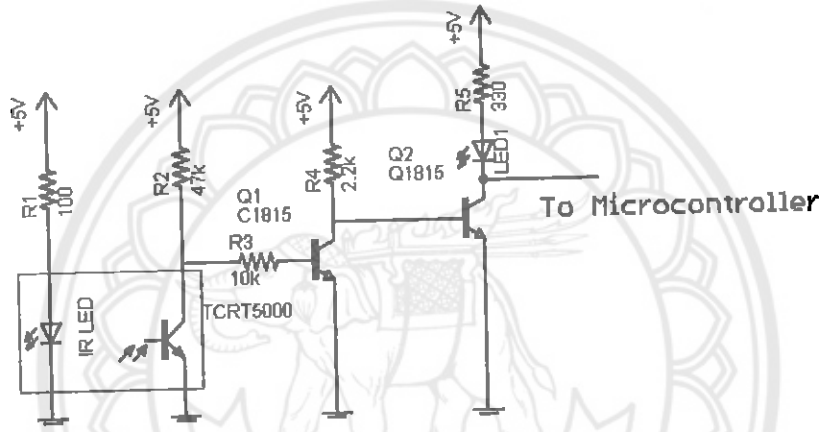
การออกแบบและการวิเคราะห์การทำงานของวงจร

หุ่นยนต์หลบหลีกสิ่งกีดขวางมีส่วนประกอบหลักในการทำงาน 4 ส่วนคือ

3.1 วงจรภาคตรวจจับวัตถุ (วงจรภาคเซ็นเซอร์)

วงจรภาคนี้มีทั้งหมด 3 ชุด เพื่อใช้สำหรับการตรวจจับวัตถุใน 3 ทิศทาง คือ ด้านหน้า ด้านซ้าย และด้านขวา หลักการทำงานของวงจรแต่ละชุดจะเหมือนกัน

วงจรภาคนี้มีส่วนประกอบสำคัญคือเซ็นเซอร์ตรวจจับวัตถุ ซึ่งเซ็นเซอร์ที่ใช้เป็นเซ็นเซอร์อินฟราเรดโมดูล เบอร์ TCRT5000

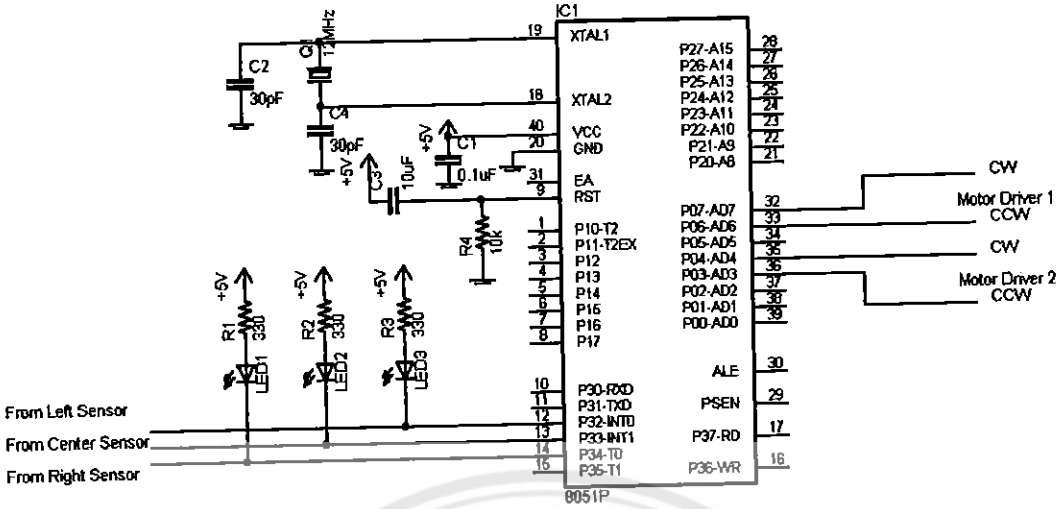


รูปที่3.1 วงจรภาคตรวจจับวัตถุ

หลักการทำงาน เมื่ออยู่ในสภาวะปกติ โฟโตทรานซิสเตอร์ในอินฟราเรดเซนเซอร์โมดูลจะไม่นำกระแส เนื่องจากไม่มีการไบแอสด้วยแสง เป็นผลให้กระแสไหลผ่าน R2 และ R3 ไปไบแอส Q1 ให้นำกระแส Q2 จะไม่นำกระแส ที่เอาท์พุท จึงมีแรงดัน 5V (ลอจิก 1) ผ่าน R1 และ LED1 ออกไปยังไมโครคอนโทรลเลอร์ LED1 จะไม่ติดสว่าง

เมื่อมีการไบแอสด้วยแสงที่โฟโตทรานซิสเตอร์ ตัวโฟโตทรานซิสเตอร์จะนำกระแส เป็นผลให้ไม่มีการไบแอสที่ Q1 ทรานซิสเตอร์ Q2 จะนำกระแส LED1จึงติดสว่าง และที่เอาท์พุท จะมีลอจิกเป็น 0 ออกไปยังไมโครคอนโทรลเลอร์

3.2 วงจรควบคุม (วงจรภาคไมโครคอนโทรลเลอร์)



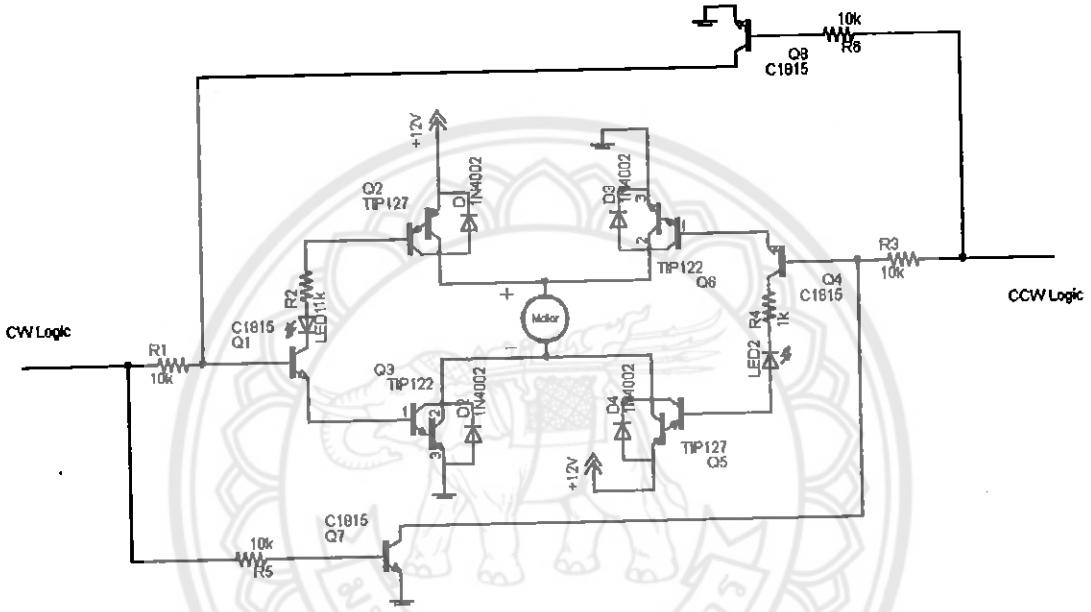
รูปที่3.2 วงจรควบคุม

วงจรภาคนี้เลือกใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89C52 เนื่องจากไมโครคอนโทรลเลอร์ตระกูลนี้มีข้อดีหลายประการ อาทิเช่น เขียนโปรแกรมเพื่อควบคุมได้ง่าย หากโปรแกรมที่เขียนมีขนาดไม่ใหญ่เกินไป สามารถใช้เพียงหน่วยความจำภายในไมโครคอนโทรลเลอร์ ไม่ต้องต่อแรมหรือรอมภายนอกเพิ่มเติม วงจรในส่วนนี้ทำงานที่สัญญาณนาฬิกา 12 MHz โดยใช้คริสตอลความถี่ 12 MHz เป็นตัวกำเนิดสัญญาณนาฬิกาให้กับไมโครคอนโทรลเลอร์

หลักการทำงานของวงจรส่วนวงจรควบคุมเริ่มต้นที่ไมโครคอนโทรลเลอร์รับค่าจากวงจรภาคตรวจจับวัตถุเข้ามาทางขา 12-14 ของไมโครคอนโทรลเลอร์ โดยปกติที่ไม่มีสัญญาณเข้า ที่ขาเหล่านี้จะมีลอจิกเป็น 1 เป็นผลให้ LED1-3 ไม่ติดสว่าง แต่ถ้าหากมีการส่งค่าจากชุดวงจรตรวจจับจะมีการส่งค่าออกมาเป็นลอจิก 0 จึงมีผลทำให้ LED1-3 ติดสว่างตามแต่ว่ามีอินพุตจากวงจรตรวจจับชุดใดเข้าที่วงจรควบคุม โดยเมื่อตัวไมโครคอนโทรลเลอร์รับค่าจากวงจรตรวจจับแล้วก็จะนำไปประมวลผลต่อไป โดยมีลอจิก 1 ที่ได้จากการประมวลผลออกเป็นเอาท์พุทที่ขา 32 และ 33 ให้กับวงจรขับมอเตอร์ชุดที่ 1 และออกที่ขา 35 และ 36 ให้กับวงจรขับมอเตอร์ชุดที่ 2 เพื่อนำไปขับมอเตอร์ โดยจะออกแยกเป็น 4 เอาท์พุท คือ มอเตอร์ชุดที่ 1 หมุนตามเข็มนาฬิกา มอเตอร์ชุดที่ 1 หมุนทวนเข็มนาฬิกา มอเตอร์ชุดที่ 2 หมุนตามเข็มนาฬิกา และมอเตอร์ชุดที่ 2 หมุนทวนเข็มนาฬิกา

3.3 วงจรขับมอเตอร์

วงจรในภาคนี้จะมีทั้งหมด 2 ชุด สำหรับการขับมอเตอร์ทางซ้ายและขวา จะใช้วงจร H-Bridge ซึ่งทรานซิสเตอร์ทุกตัวในวงจรจะทำหน้าที่เป็นสวิตช์ โดยเลือกใช้ดาร์ลิงตันทรานซิสเตอร์เบอร์ TIP122 และ TIP127 เป็นตัวขับกำลังให้มอเตอร์ เนื่องจากมีคุณสมบัติทนกระแสและแรงดันได้ค่อนข้างสูง โดยวงจรของทั้ง 2 ชุดจะมีหลักการทำงานเหมือนกัน แต่จะรับสัญญาณมาจากคนละขาของไมโครคอนโทรลเลอร์ สำหรับวงจรในภาคนี้จะมีหลักการทำงานที่เกี่ยวข้องกับสัญญาณที่มาจากวงจรควบคุมดังนี้



รูปที่ 3.3 วงจรขับมอเตอร์

เมื่อมีสัญญาณมาจากชุดวงจรควบคุมเข้ามาทางด้านการหมุนตามเข็มนาฬิกา (CW Logic) Q1 จะทำงาน ทำให้ Q2 และ Q3 ทำงาน เป็นผลให้มีกระแสไหลผ่านมอเตอร์ให้มีการหมุนตามเข็มนาฬิกาได้ โดย LED1 จะทำหน้าที่แสดงสถานะการทำงานของการหมุนตามเข็มนาฬิกา ส่วน Q7 จะมีหน้าที่กันไม่ให้เมื่อมีสัญญาณมาจากด้านตามเข็มนาฬิกา แล้วมีลอจิกมาจากด้านทวนเข็มนาฬิกาด้วยทำให้ทรานซิสเตอร์ที่ทำหน้าที่ขับมอเตอร์ทำงานพร้อมกันทั้ง 4 ตัว ซึ่งอาจจะทำให้อุปกรณ์เสียหายได้โดยหากมีลอจิกมาจากอีกด้านหนึ่งโดยที่อีกด้านหนึ่งทำงานอยู่จะผ่าน Q7 ลงกราวนด์

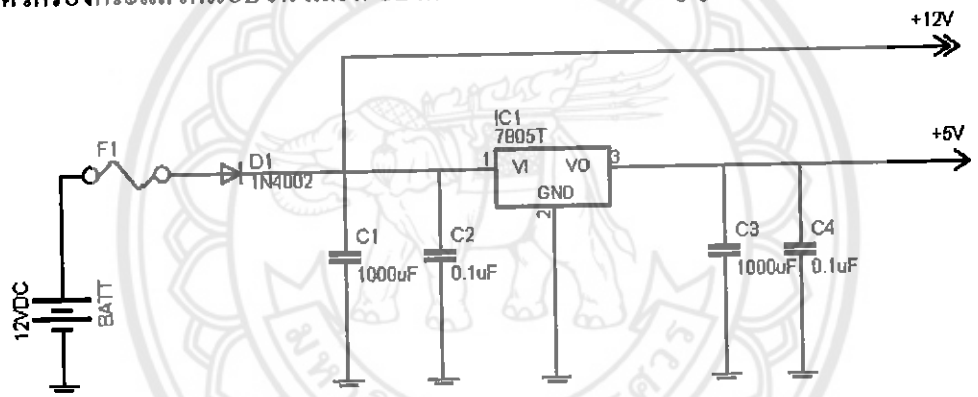
เมื่อมีสัญญาณมาจากชุดวงจรควบคุมเข้ามาทางด้านการหมุนทวนเข็มนาฬิกา (CCW Logic) Q4 จะทำงาน ทำให้ Q5 และ Q6 ทำงาน เป็นผลให้มีกระแสไหลผ่านมอเตอร์ให้มีการหมุนทวนเข็มนาฬิกาได้ โดย LED2 จะทำหน้าที่แสดงสถานะการทำงานของการหมุนตามเข็มนาฬิกา ส่วน Q8 จะทำหน้าที่เช่นเดียวกับ Q7 คือกันไม่ให้เมื่อมีสัญญาณมาจากด้านทวนเข็มนาฬิกา แล้วมีลอจิกมาจากด้านตามเข็มนาฬิกาด้วยทำให้ทรานซิสเตอร์ที่ทำหน้าที่ขับมอเตอร์ทำงานพร้อมกันทั้ง 4 ตัว ซึ่ง

อาจจะทำให้อุปกรณ์เสียหายได้โดยหากมีลอจิกมาจากอีกด้านหนึ่งโดยที่อีกด้านหนึ่งทำงานอยู่จะผ่าน Q8 ลงกราวนด์

สำหรับมอเตอร์ในวงจรนี้เลือกใช้มอเตอร์ 24 VDC แต่นำมาใช้กับแรงดัน 12 V เนื่องจากมีแรงบิดค่อนข้างสูง เพราะหุ่นค่อนข้างจะมีน้ำหนักมาก หากใช้มอเตอร์ 12 V กำลังขับอาจจะไม่เพียงพอ

3.4 วงจรภาคจ่ายไฟ

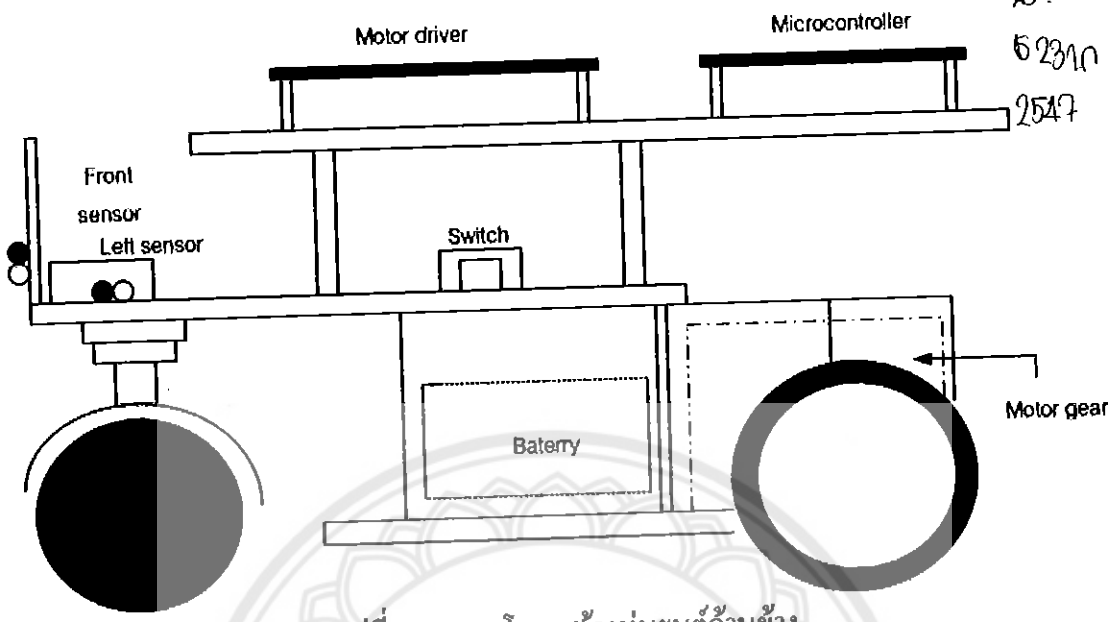
วงจรในภาคนี้ประกอบไปด้วยแบตเตอรี่ 12 V ซึ่งอาจจะเลือกอนุกรมแรงดันให้สูงขึ้นได้อีก แต่ไม่เกินพิกัดของมอเตอร์ที่ 24 V โดยผ่านฟิวส์ป้องกันกระแสและไดโอดเพื่อป้องกันการต่อกลับขั้วโดยแรงดัน 12 V จะนำไปต่อกับวงจรขับมอเตอร์ ส่วนวงจรภาคอื่นๆ ต้องใช้ไฟเลี้ยง 5 V โดยใช้ไอซีเรียงกระแสเบอร์ 7805 เพื่อลดแรงดัน 12 VDC ให้เป็นไฟตรง 5 VDC โดยมี C1 และ C3 เป็นตัวกรองกระแสให้เรียบขึ้น และมี C2 และ C4 เป็นตัวกรองสัญญาณรบกวน



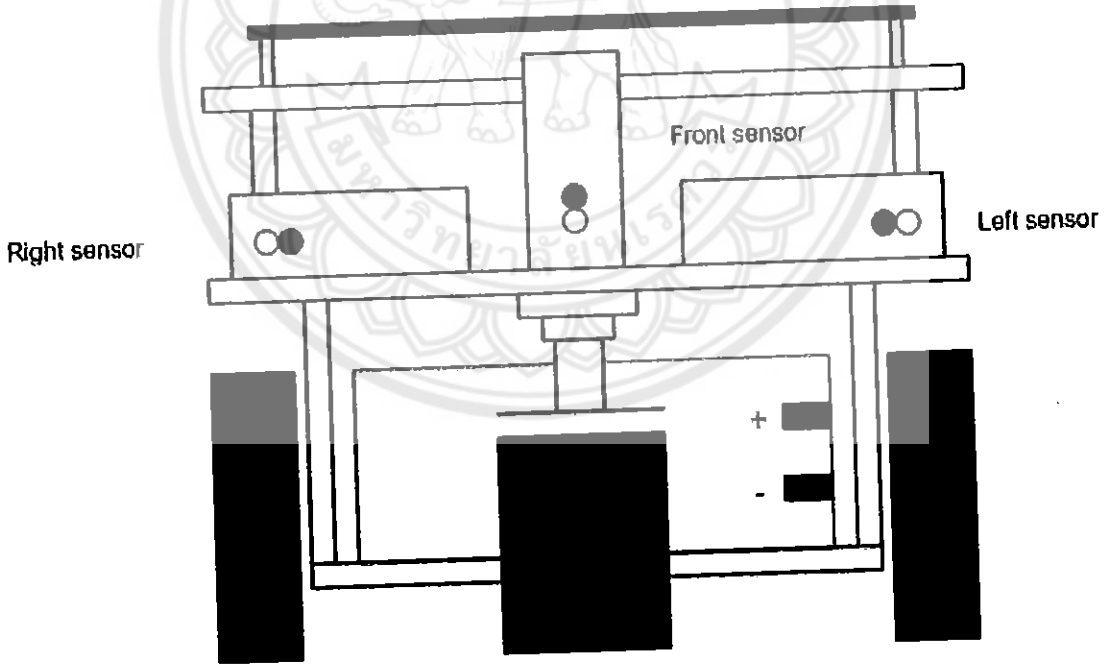
รูปที่ 3.4 วงจรภาคจ่ายไฟ

1502566. 4900021
วิ.
62310
2547

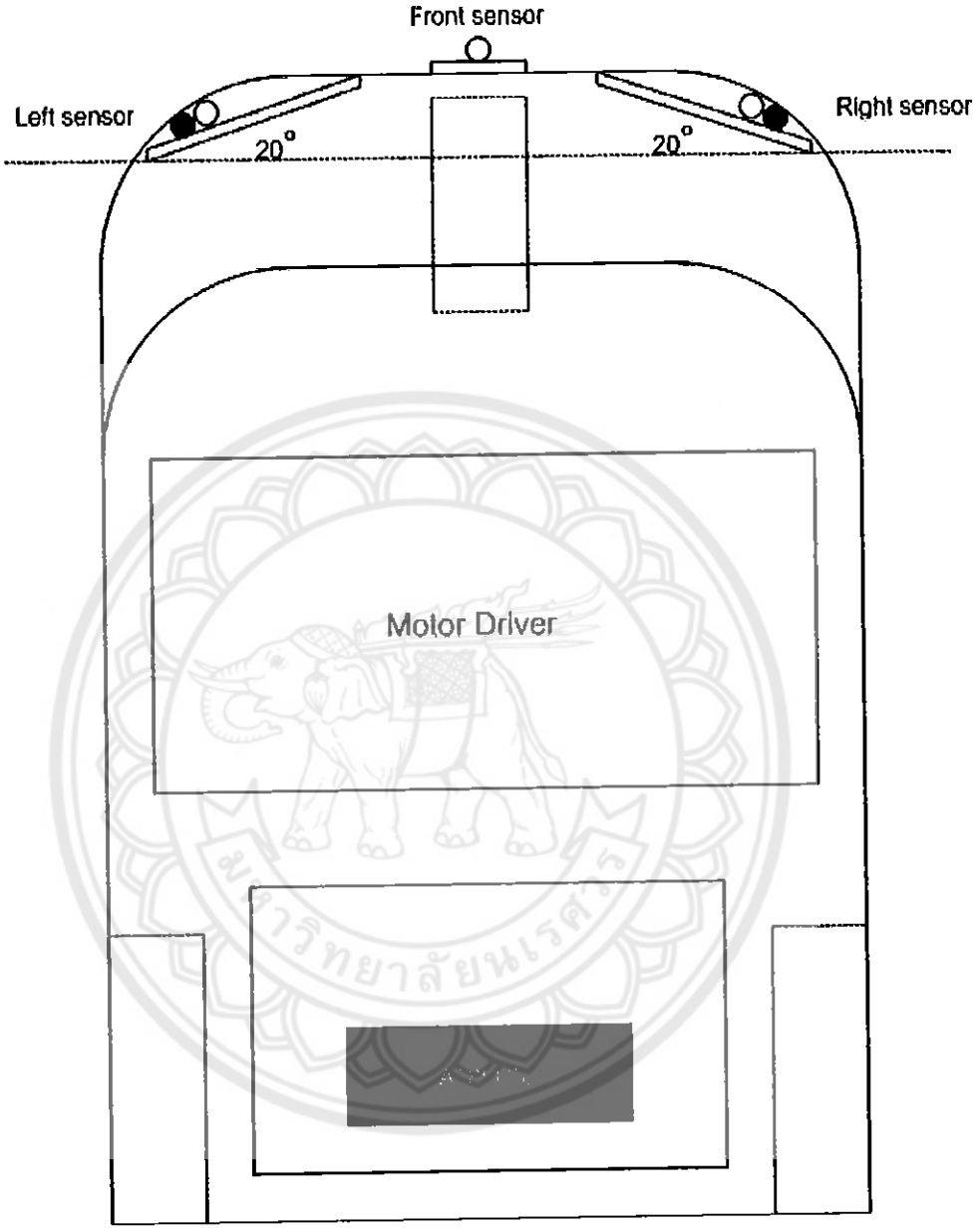
3.5 โครงสร้างของตัวหุ่นยนต์



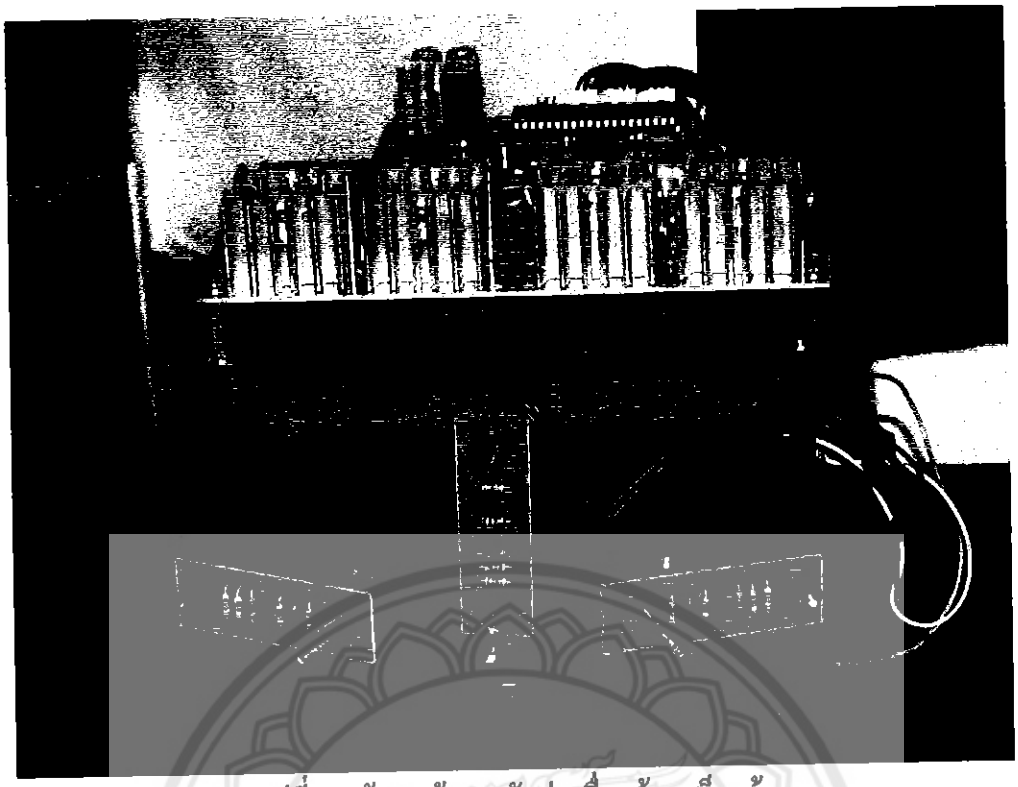
รูปที่ 3.5 แสดงโครงสร้างหุ่นยนต์ด้านข้าง



รูปที่ 3.6 แสดงโครงสร้างหุ่นยนต์ด้านหน้า

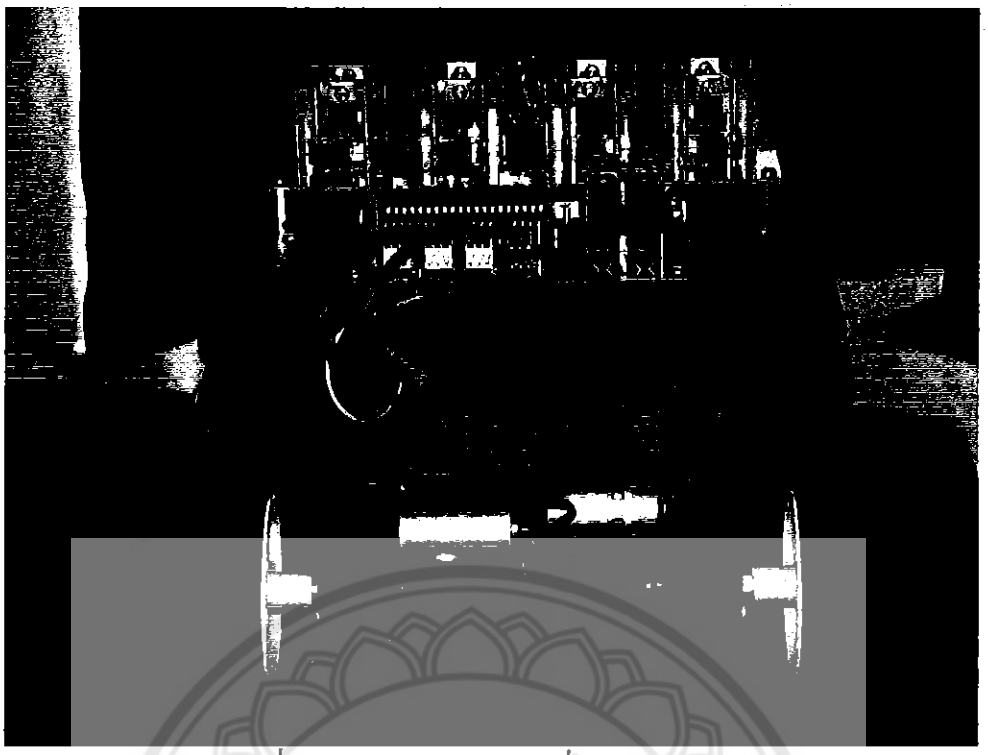


รูปที่ 3.7 แสดงโครงสร้างหุ่นยนต์ด้านบน

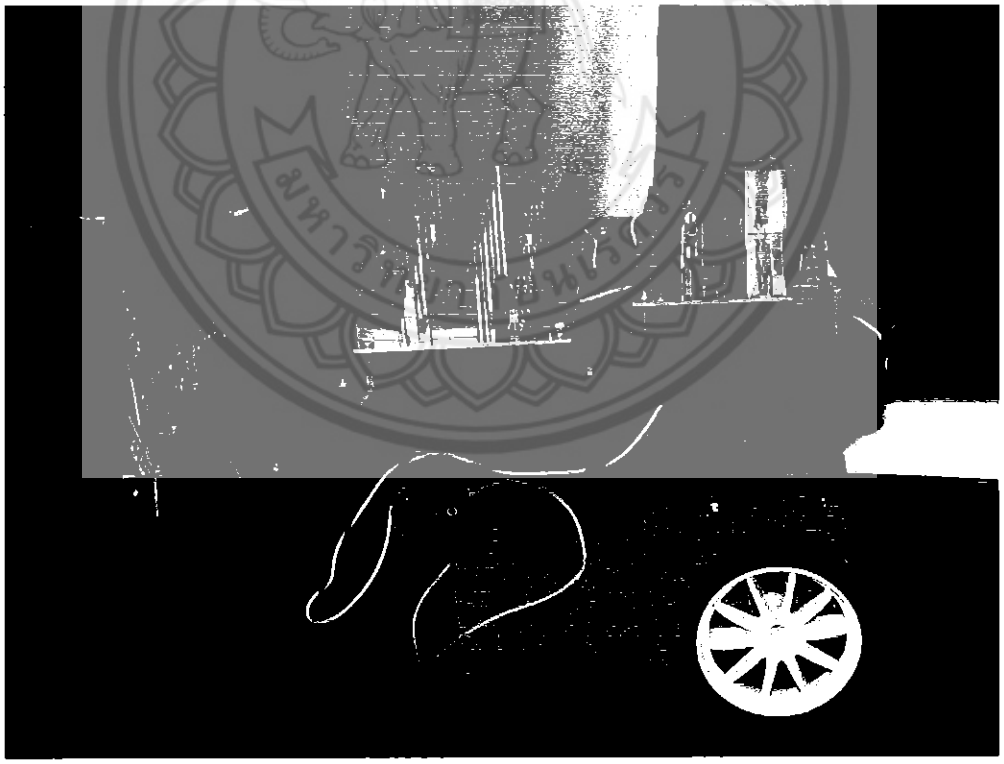


รูปที่ 3.7 ด้านหน้าของตัวหุ่นเมื่อสร้างเสร็จแล้ว

รูปที่ 3.8 ด้านข้างของตัวหุ่นเมื่อสร้างเสร็จแล้ว



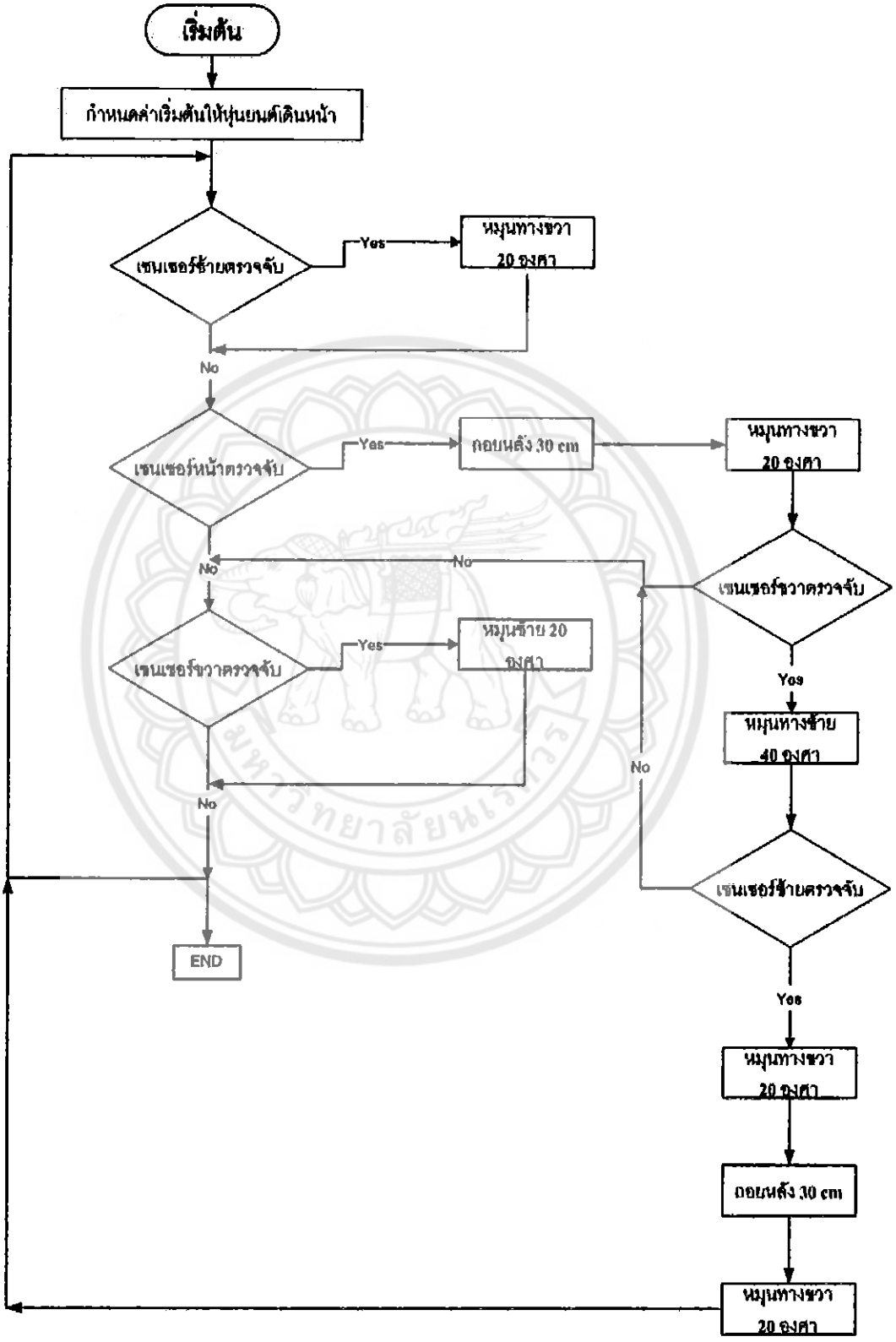
รูปที่ 3.9 ด้านหลังของตัวหุ่นเมื่อสร้างเสร็จแล้ว



รูปที่ 3.10 ด้านข้างของตัวหุ่นเมื่อสร้างเสร็จแล้ว

3.6 แผนภาพการไหล

แสดงการทำงานของหุ่นยนต์หลบหลีก



รูปที่ 3.11 แสดงทำงานของหุ่นยนต์หลบหลีก

บทที่ 4

การทดลองและวิธีการทดลอง

การทดสอบการทำงานของหุ่นยนต์หลบหลีกประกอบไปด้วยการทดลอง 3 ตอนซึ่งมีเนื้อหาและจุดประสงค์ดังนี้

1.ทดสอบการตรวจจับและระยะของเซนเซอร์

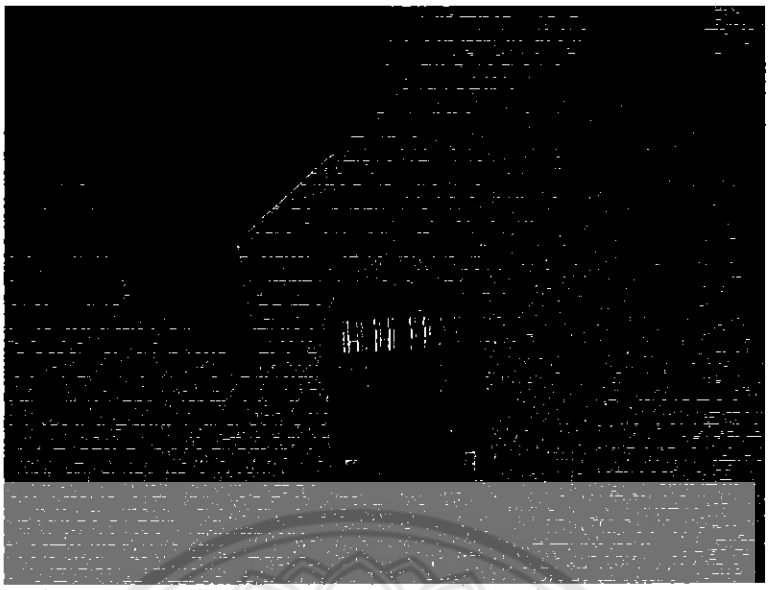
2.ทดสอบการทำงานของไมโครโพรเซสเซอร์

3.ทดสอบการทำงานของวงจรขับมอเตอร์

4.1 การทดลองที่ 1 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์

รูปที่ 4.1 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์

ทำการปล่อยให้หุ่นยนต์ออกเดินโดยให้เดินเข้าหาสิ่งกีดขวางทางด้านซ้ายมือ



รูปที่ 4.2 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์ (ต่อ)

เมื่อหุ่นยนต์เดินไปเจอกับสิ่งกีดขวางทางด้านซ้าย เซนเซอร์ซ้ายจะตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับสิ่งกีดขวาง

รูปที่ 4.3 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์ (ต่อ)

ไมโครโพรเซสเซอร์สั่งการไปที่วงจรขับมอเตอร์เพื่อให้หุ่นยนต์หลบหลีกไปทางด้านขวาประมาณ 20 องศา และเมื่อหลบสิ่งกีดขวางทางด้านซ้ายจนพ้นระยะเซนเซอร์ตรวจจับแล้ว ไมโครโพรเซสเซอร์จึงสั่งการไปที่วงจรขับมอเตอร์เพื่อสั่งให้มอเตอร์หมุนเดินขึ้นหน้าต่อไป

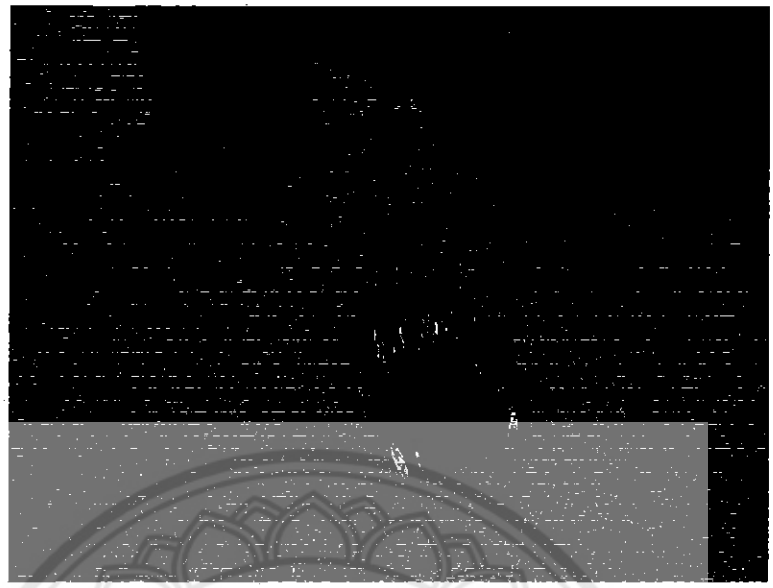
รูปที่ 4.4 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์ **(ต่อ)**

เมื่อหุ่นยนต์เดินไปเจอกับสิ่งกีดขวางทางด้านซ้ายอีกครั้ง เซนเซอร์ซ้ายจะตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับสิ่งกีดขวาง

รูปที่ 4.5 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์(ต่อ)

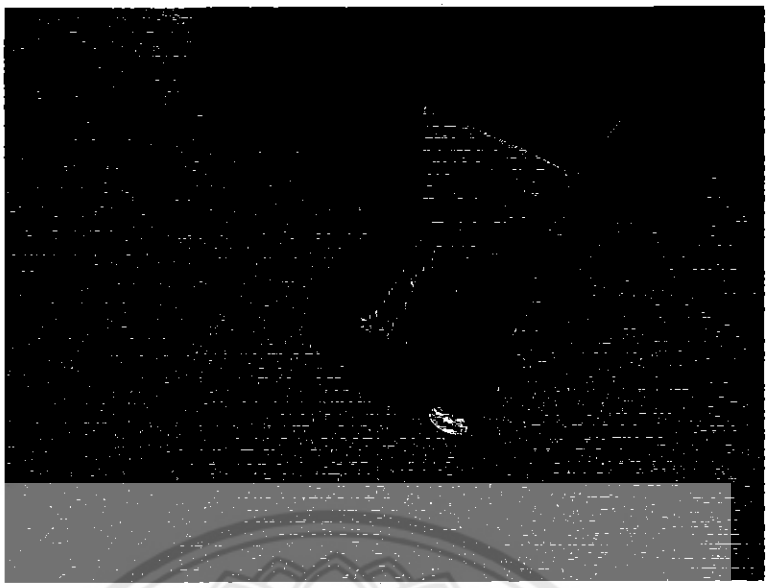
หุ่นยนต์หลบหลีกออกทางขวาประมาณ 20 องศาอีกครั้ง และทำการเดินหน้าต่อไปเรื่อยๆจนกว่าจะพบกับสิ่งกีดขวางอีกครั้ง

4.2 การทดลองที่ 2 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์



รูปที่ 4.6 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์
ทำการปล่อยให้หุ่นยนต์ออกเดินโดยให้เดินเข้าหาสิ่งกีดขวางทางด้านขวามือ

รูปที่ 4.7 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ)
เมื่อหุ่นยนต์เดินไปเจอกับสิ่งกีดขวางทางด้านขวา เซนเซอร์ขวาจะตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับสิ่งกีดขวาง

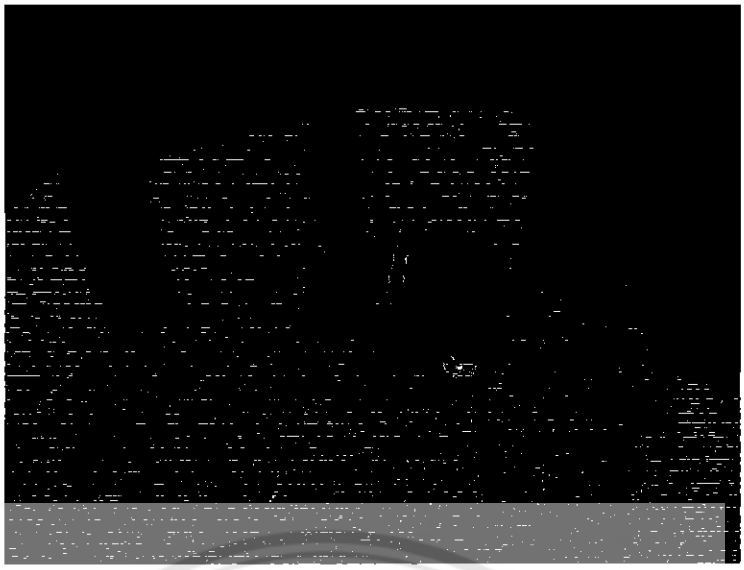


รูปที่ 4.8 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ)

ไมโครโพรเซสเซอร์สั่งการไปที่วงจรขับมอเตอร์เพื่อให้หุ่นยนต์หลบหลีกไปทางด้านซ้ายประมาณ 20 องศา และเมื่อหลบสิ่งกีดขวางทางด้านขวาจนพ้นระยะเซนเซอร์ตรวจจับแล้ว ไมโครโพรเซสเซอร์จึงสั่งการไปที่วงจรขับมอเตอร์เพื่อสั่งให้มอเตอร์หมุนเดินขึ้นหน้าต่อไป

รูปที่ 4.9 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ)

เมื่อหุ่นยนต์เดินไปเจอกับสิ่งกีดขวางทางด้านขวาอีกครั้ง เซนเซอร์ซ้ายจะตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับสิ่งกีดขวาง



รูปที่ 4.10 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์ (ต่อ)
หุ่นยนต์หลบหลีกออกทางซ้ายประมาณ 20 องศาอีกครั้ง และทำการเดินหน้าต่อไปเรื่อยๆจนกว่าจะพบกับสิ่งกีดขวางอีกครั้ง

4.3 การทดลองที่ 3 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์

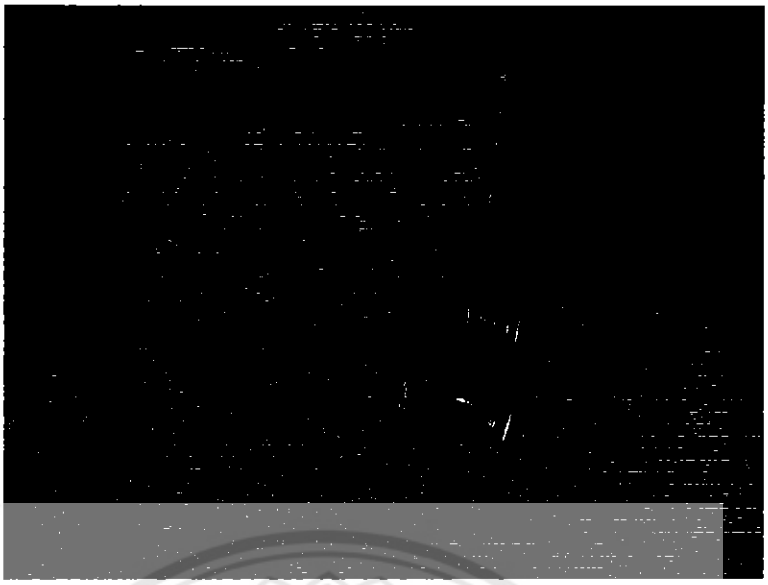
รูปที่ 4.11 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์
ทำการปล่อยให้หุ่นยนต์เดินหน้าไปเรื่อยๆ

รูปที่ 4.12 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ)

หุ่นยนต์เดินหน้าไปจนเจอกับสิ่งกีดขวางที่อยู่ด้านหน้า เซนเซอร์หน้าก็จะตรวจจับโดยส่งแสงอินฟราเรดออกไปกระทบกับสิ่งกีดขวาง

รูปที่ 4.13 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์ (ต่อ)

จากนั้นไมโครโพรเซสเซอร์จะสั่งการไปที่วงจรขับมอเตอร์เพื่อสั่งให้หุ่นยนต์เดินถอยหลังมาประมาณ 30 เซนติเมตร

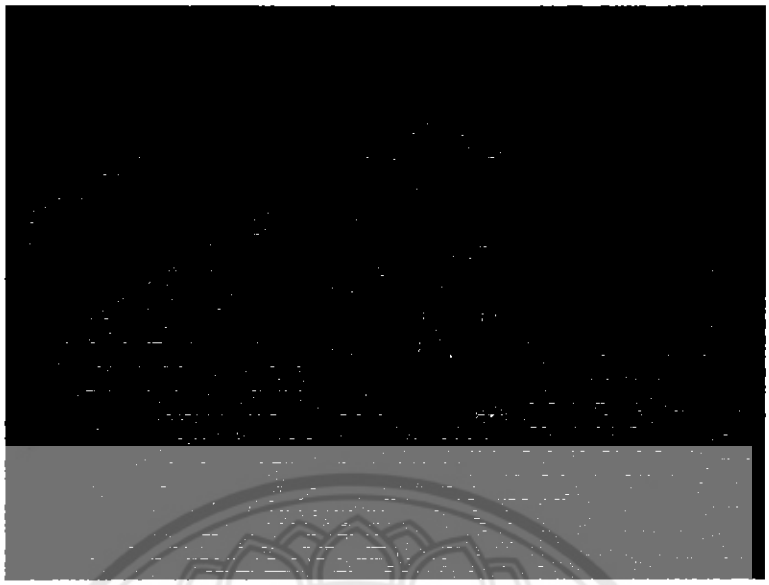


รูปที่ 4.14 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ)

จากนั้นจึงสั่งการให้หุ่นยนต์ทำการหมุนไปทางขวาต่อทันทีประมาณ 20 องศา ถ้าหากไม่มีสิ่งกีดขวางหุ่นยนต์ก็จะทำการเดินหน้าหลบออกไปได้

รูปที่ 4.14 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ)

แต่ถ้าหากหมุนหลบไปแล้วยังเจอกับสิ่งกีดขวางทางด้านขวาอีก เซนเซอร์ก็จะตรวจจับทันที

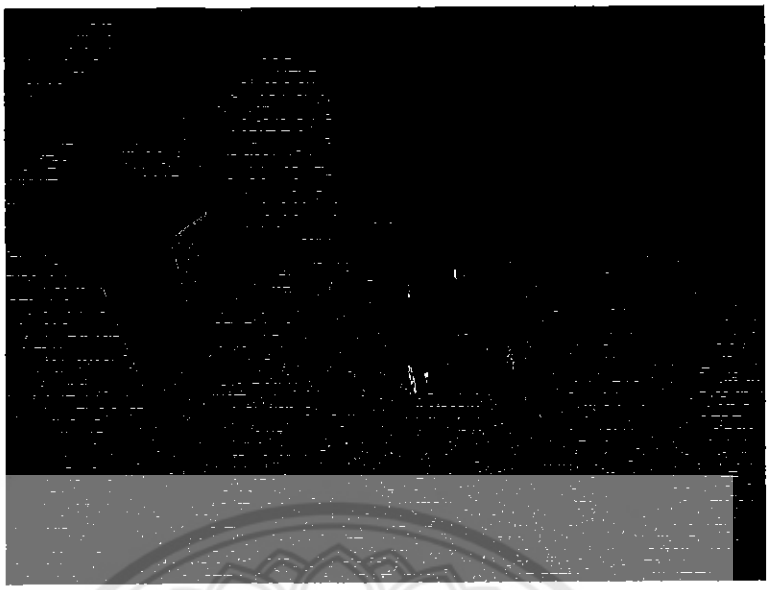


รูปที่ 4.15 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ) จากนั้นไมโครโพรเซสเซอร์ก็จะสั่งการไปที่วงจรขับมอเตอร์เพื่อสั่งให้หุ่นยนต์หมุนกลับมาทางด้านซ้ายทันทีประมาณ 40 องศา ถ้าหากไม่มีสิ่งกีดขวางหุ่นยนต์ก็จะเดินหน้าต่อไป

รูปที่ 4.16 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ) แต่ถ้าหากว่ายังมีสิ่งกีดขวางอยู่ทางด้านซ้ายอีกเซนเซอร์จะตรวจจับทันที

รูปที่ 4.17 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ) ไมโครโพรเซสเซอร์ก็จะสั่งการไปที่วงจรขับมอเตอร์เพื่อสั่งให้หุ่นยนต์หมุนกลับมาทางด้านขวาทันทีประมาณ 20 องศา

รูปที่ 4.18 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ) และไมโครโพรเซสเซอร์ก็จะสั่งการต่อทันทีไปที่วงจรขับมอเตอร์เพื่อสั่งให้หุ่นยนต์เดินถอยหลังออกมาอีกประมาณ 30 เซนติเมตร



รูปที่ 4.19 แสดงการทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์(ต่อ) และสั่งการต่อให้หุ่นยนต์เดินหลบออกไปทางด้านอื่น(ในที่นี้คือด้านขวา) เพื่อเดินออกมาให้หลุดออกจากมุมอับนั้นๆ

บทที่ 5
สรุปผลการทดลอง

5.1 วิเคราะห์ผลการทดลอง

การทดลองที่ 1 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านซ้ายของหุ่นยนต์

การทดสอบในตอนนี้เป็นการทดสอบการทำงานของเซนเซอร์ตัวซ้าย ซึ่งเมื่อหุ่นยนต์เดินหน้าไปจนเจอกับสิ่งกีดขวางที่อยู่ที่อยู่ทางด้านซ้าย เซนเซอร์ซ้ายก็จะทำการตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับสิ่งกีดขวางนั้น และจะได้ค่าออกมาค่าหนึ่งส่งไปยังไมโครโพรเซสเซอร์เพื่อทำการประมวลผลจากเงื่อนไขในตัวโปรแกรมและสั่งการต่อไปยังวงจรขับมอเตอร์เพื่อสั่งให้มอเตอร์หมุนหลบไปทางด้านขวาจนพ้นสิ่งกีดขวางและเดินหน้าต่อไป

การทดลองที่ 2 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านขวาของหุ่นยนต์

การทดสอบในตอนนี้เป็นการทดสอบการทำงานของเซนเซอร์ตัวขวา ซึ่งเมื่อหุ่นยนต์เดินหน้าไปจนเจอกับสิ่งกีดขวางที่อยู่ที่อยู่ทางด้านขวา เซนเซอร์ขวาก็จะทำการตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับสิ่งกีดขวางนั้น และจะได้ค่าออกมาค่าหนึ่งส่งไปยังไมโครโพรเซสเซอร์เพื่อทำการประมวลผลจากเงื่อนไขในตัวโปรแกรมและสั่งการต่อไปยังวงจรขับมอเตอร์เพื่อสั่งให้มอเตอร์หมุนหลบไปทางด้านซ้ายจนพ้นสิ่งกีดขวางและเดินหน้าต่อไป

การทดลองที่ 3 การทดสอบการหลบหลีกสิ่งกีดขวางทางด้านหน้า,ซ้ายและขวาของหุ่นยนต์

การทดสอบนี้เป็นการทดสอบเมื่อเวลาที่หุ่นยนต์เดินเข้าไปยังมุมๆหนึ่ง ซึ่งมีหลายๆแบบและให้หุ่นยนต์สามารถแก้สถานการณ์เดินหลบออกมาจากมุมนั้นๆได้ โดยเริ่มจากปล่อยให้หุ่นยนต์เดินหน้าจนไปเจอกับสิ่งกีดขวางด้านหน้าเซนเซอร์หน้าก็จะตรวจจับได้ ไมโครโพรเซสเซอร์จะสั่งการให้หุ่นยนต์เบี่ยงหลบไปทางด้านขวาทันทีประมาณ 20 องศา แต่ถ้าหากว่าทางด้านขวาที่หุ่นยนต์หมุนหลบไปยังมีสิ่งกีดขวางอยู่อีกเซนเซอร์ขวาก็จะตรวจจับพบทันที และไมโครโพรเซสเซอร์ก็จะสั่งการให้หุ่นยนต์หมุนกลับมาทางด้านซ้ายทันทีประมาณ 40 องศาเพื่อหลบออกจากมุมนั้นให้ได้ แต่ถ้าหากด้านซ้ายที่หมุนกลับมายังมีสิ่งกีดขวางอยู่อีกเซนเซอร์ซ้ายก็จะตรวจจับ ไมโครโพรเซสเซอร์ก็จะสั่งการให้หุ่นยนต์หมุนกลับมาทางด้านขวาอีกทีประมาณ 20 องศาแล้วสั่งให้หุ่นยนต์เดินถอยหลังออกมาในทันทีแล้วเบี่ยงหลบไปทางด้านอื่นทันที(เบี่ยงไปทางด้านขวา)เพื่อให้พ้นจากมุมนั้นให้ได้

5.2 สรุปขั้นตอนการทำงานของหุ่นยนต์หลบหลีก

หลักการทำงานของหุ่นยนต์เมื่อเราทำการเปิดสวิตซ์ บอร์ดไมโครโพรเซสเซอร์จะทำการประมวลผลจากตัวโปรแกรมที่เราบรรจุลงในชิพ AT89C52 โดยหุ่นยนต์จะหยุดนิ่งสักครู่หนึ่งหลังจากทำการเปิดสวิตซ์แล้ว หลังจากนั้นไมโครโพรเซสเซอร์จะสั่งให้หุ่นยนต์เดินหน้าไปเรื่อยๆ จนกว่าจะพบกับสิ่งกีดขวาง เมื่อหุ่นยนต์เดินหน้าไปพบกับสิ่งกีดขวางที่อยู่ทางด้านซ้ายของหุ่นยนต์ เซ็นเซอร์ซ้ายของหุ่นยนต์ก็จะทำการตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับวัตถุและสะท้อนกลับมายังตัวรับแสงอินฟราเรด จากนั้นไมโครโพรเซสเซอร์ก็จะได้รับค่าจากเซนเซอร์แล้วจึงทำการประมวลผลแล้วสั่งการไปยังวงจรขับมอเตอร์ เพื่อสั่งให้หุ่นยนต์ทำการเลี้ยวขวาไปประมาณ 20 องศา เพื่อหลบให้พ้นจากสิ่งกีดขวางที่อยู่ทางด้านซ้าย เช่นเดียวกันเมื่อหุ่นยนต์เดินมาเจอสิ่งกีดขวางที่อยู่ทางด้านขวาเซนเซอร์ขวาของหุ่นยนต์ก็จะทำการตรวจจับโดยส่งแสงอินฟราเรดไปกระทบกับวัตถุและตรวจจับการสะท้อนกลับมายังตัวรับแสงอินฟราเรด จากนั้นไมโครโพรเซสเซอร์ก็จะได้รับค่าจากเซนเซอร์แล้วจึงทำการประมวลผลแล้วสั่งการไปยังวงจรขับมอเตอร์ เพื่อสั่งให้หุ่นยนต์ทำการเลี้ยวซ้ายไปประมาณ 20 องศา เพื่อหลบให้พ้นจากสิ่งกีดขวางที่อยู่ทางด้านขวา เมื่อหุ่นยนต์เดินมาเจอกับสิ่งกีดขวางทางด้านหน้าของหุ่นยนต์เซนเซอร์หน้าของหุ่นยนต์ก็จะทำการตรวจจับ โดยส่งแสงอินฟราเรดไปกระทบกับวัตถุและสะท้อนกลับมายังตัวรับแสงอินฟราเรด จากนั้นไมโครโพรเซสเซอร์ก็จะได้รับค่าจากเซนเซอร์แล้วจึงทำการประมวลผลแล้วสั่งการไปยังวงจรขับมอเตอร์ เพื่อสั่งให้หุ่นยนต์ทำการถอยหลังมาประมาณ 30 เซนติเมตรและทำการเลี้ยวไปทางขวาไปประมาณ 20 องศา เป็นลำดับแรกก่อน จากนั้นเมื่อทางขวาไม่มีสิ่งกีดขวางหุ่นยนต์ก็จะทำการเดินหน้าต่อไป แต่ถ้าหากว่ามีสิ่งกีดขวางทางด้านขวาอยู่เซนเซอร์ขวาก็จะตรวจจับแล้วส่งค่าไปยังไมโครโพรเซสเซอร์ จากนั้นไมโครโพรเซสเซอร์ก็จะสั่งการให้หุ่นยนต์ทำการเลี้ยวกลับมาทางด้านซ้ายในทันทีเป็นมุม ประมาณ 40 องศาเพื่อทำการหลบออกจากสิ่งกีดขวางให้ได้ แต่ถ้าหากด้านซ้ายยังมีสิ่งกีดขวางอยู่อีก ไมโครโพรเซสเซอร์ก็จะสั่งการให้หุ่นยนต์ทำการเลี้ยวกลับมาทางด้านขวาประมาณ20 องศา และทำการถอยหลังออกมาทันทีอีกประมาณ 30 เซนติเมตร และสั่งให้หุ่นยนต์เลี้ยวหลบไปทางขวาอีกประมาณ 20 องศา เพื่อทำการเลี้ยวหลบสิ่งกีดขวางและทำการเดินหน้าต่อไป

5.3 ปัญหาและแนวทางแก้ไข

ปัญหาที่พบในการทำโครงงานนี้ มีดังต่อไปนี้ คือ

1) เซ็นเซอร์แต่ละตัว มีระยะตรวจจับค่อนข้างจำกัด ทำให้ประสิทธิภาพในการหลบหลีกของตัวหุ่นไม่สูงนัก และตัวเซ็นเซอร์มีความไวต่อแสงธรรมชาติ ทำให้มีความผิดพลาดในการตรวจจับเกิดขึ้น

2) มอเตอร์แต่ละตัววิ่งด้วยความเร็วไม่เท่ากัน ทำให้เมื่อสั่งให้หุ่นวิ่งตรงไปข้างหน้า อาจจะวิ่งไม่เป็นเส้นตรงได้

แนวทางการแก้ไข

1) ใช้เซ็นเซอร์อินฟราเรดที่มีระยะการตรวจจับไกลขึ้น หรือเปลี่ยนไปใช้เซ็นเซอร์ชนิดอื่น

2) ใช้การปรับตั้ง (Calibrate) ให้มอเตอร์ทั้งสองตัวมีการหมุนด้วยความเร็วที่เท่ากัน

5.4 ข้อเสนอแนะ

หากจะนำไปพัฒนาต่อให้หุ่นมีการทำงานที่ดีขึ้น เช่น ตัดสินใจในการเคลื่อนที่แบบอัตโนมัติได้ดีขึ้น สามารถเคลื่อนที่เข้าสู่เป้าหมายที่กำหนดไว้ล่วงหน้า รับน้ำหนักโหลดได้มากขึ้น อาจจะดำเนินการพัฒนาต่อในส่วนของการติดวงจรตรวจจับวัตถุเพิ่มเติม หรือเพิ่มภาครับสัญญาณวิทยุที่ส่งมาจากเป้าหมายเพื่อให้หุ่นเคลื่อนที่หลบหลีกสิ่งกีดขวางและเข้าหาเป้าหมายนั้น หรือเพียงอนุกรมแบตเตอรี่ให้มีแรงดันสูงขึ้น เพื่อให้สามารถรับน้ำหนักโหลดได้มากขึ้น เพราะมอเตอร์ที่ใช้ในตัวหุ่นเป็นมอเตอร์ที่ใช้แรงดันได้ถึง 24 V แต่ในโครงงานนี้ใช้แรงดันเพียง 12 V ก็สามารถทำได้ตามแต่แนวทางของผู้นำไปพัฒนาต่อไป

เอกสารอ้างอิง

[1] วรพจน์ กรแก้ววัฒนกุล , ชัยวัฒน์ ลิ้มพรจิตรวิลัย , เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 , พิมพ์ครั้งที่1 , กรุงเทพฯ , หจก.วีเจ พริ้นติ้ง

[2] ผศ. ธีรวัฒน์ ประกอบผล , การประยุกต์ใช้งานไมโครคอลโทรลเลอร์ , พิมพ์ครั้งที่6 ,กรุงเทพฯ , 2546

[3] ผศ. อุทัย สุขสิงห์ , ไมโครโพรเซสเซอร์และไมโครคอนโทรลเลอร์ MCS-51 , พิมพ์ครั้งที่1 , กรุงเทพฯ , 2547

[4] รศ.ดร. วีระเชษฐ์ ขันเงิน , อิเล็กทรอนิกส์กำลัง , พิมพ์ครั้งที่2 , กรุงเทพฯ , หจก.วีเจ พริ้นติ้ง , 2547

[5] สุระพล เชียรมนตรี , อิเล็กทรอนิกส์กำลัง , พิมพ์ครั้งที่1 , กรุงเทพฯ , 2545

[6] MUHAMMAD H.RASHID, POWER ELECTRONICS , NEW JERSEY : PRENTICE HALL, 1997

ภาคผนวก

Sorce Code

```
    ORG 0000H
    LJMP INIT
INIT: MOV P1,#0FFH
    MOV P2,#0FFH
    MOV P3,#0FFH
    CLR A
    MOV P0,A
    LCALL DELAY_1S


MAIN: JB P3.2,MAIN1
    LCALL RR

MAIN1: JB P3.3,MAIN2
    LCALL REW
    LCALL RR
    JB P3.4,MAIN2
    LCALL LL1
    JB P3.2,MAIN2
    LCALL RR
    LCALL REW
    LCALL RR

MAIN2: JB P3.4,MAIN3
        LCALL LL

MAIN3: JNB P3.2,MAIN
```

```
    JNB P3.3,MAIN
    JNB P3.4,MAIN
    LCALL FF
    SJMP MAIN
    RET

RR: SETB P0.3
    CLR  P0.4
    CLR  P0.6
    SETB P0.7
    LCALL DELAY_100MS
    LCALL DELAY_100MS

    CLR A
    MOV P0,A
    RET

LL: CLR P0.3
    SETB  P0.4
    SETB  P0.6
    CLR P0.7
    LCALL DELAY_100MS
    LCALL DELAY_100MS

    CLR A
    MOV P0,A
    RET

LL1: CLR P0.3
    SETB  P0.4
    SETB  P0.6
```

```
    CLR P0.7
    LCALL DELAY_100MS
    LCALL DELAY_100MS
    LCALL DELAY_100MS
    LCALL DELAY_100MS

    CLR A
    MOV P0,A
    RET


FF: CLR P0.3
    SETB  P0.4
    CLR  P0.6
    SETB P0.7
    LCALL DELAY_10MS

    CLR A
    MOV P0,A
    RET


DELAY_1MS: MOV R6,#0E6H
DELAY_1MS_1: NOP
        NOP
        DJNZ R6,DELAY_1MS_1
        RET

DELAY_10MS: MOV R7,#010
DELAY_10MS_1: MOV R6,#0E6H
DELAY_10MS_2: NOP
```

```
        NOP
        DJNZ R6,DELAY_10MS_2
        DJNZ R7,DELAY_10MS_1
        RET

DELAY_100MS: MOV R7,#100
DELAY_100MS_1: MOV R6,#0E6H
DELAY_100MS_2: NOP
        NOP
        DJNZ R6,DELAY_100MS_2
        DJNZ R7,DELAY_100MS_1
        RET

DELAY_1S: MOV R5,#100
DELAY_1S_1: ACALL DELAY_10MS
        DJNZ R5,DELAY_1S_1
        RET
        END
```

ประวัติผู้เขียนโครงงาน

ชื่อ นาย ธนากร เขียวแก้ว

ภูมิลำเนา 25/1 ม.5 ต.หาดล้า อ.ท่าปลา จ.อุตรดิตถ์

ประวัติการศึกษา

- จบการศึกษาระดับมัธยมศึกษาจากโรงเรียนอุตรดิตถ์
- ปัจจุบันกำลังศึกษาในระดับชั้นปีที่ 5

สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-mail : octagonz@gmail.com

ชื่อ นาย เทิดพิทักษ์ ประจง

ภูมิลำเนา 87/1 หมู่ 7 ต. คลองคะเชนทร์ อ.เมือง จ.พิจิตร

ประวัติการศึกษา

- จบการศึกษาระดับมัธยมศึกษาจากโรงเรียนพิจิตรพิทยาคม
- ปัจจุบันกำลังศึกษาในระดับชั้นปีที่ 5

สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-mail : pupu_ee@hotmailmail.com