



การพัฒนาหุ่นยนต์เคลื่อนที่ได้เอง ระยะที่ 3

THE DEVELOPMENT OF MOBILE ROBOT IN STAGE III

นาย สมบูรณ์	มาตรคำจันทร์	34162176
นาง สุนันท์พงษ์	แก้วคอนไทย	34162178
นาย เอกชัย	สันกำแพง	34162181

อาจารย์ที่ปรึกษา
อ.เกษตร์ ศิริสันติสัมฤทธิ์
Advisor
Master.Kaset Sirisantiamrid

ปริญญาโทสำหรับปริญญาอุตสาหกรรมศาสตรบัณฑิต
สาขาเทคโนโลยีคอมพิวเตอร์อุตสาหกรรม
ภาควิชาเทคโนโลยีการควบคุมทางอุตสาหกรรม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032725

ภาควิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม

สาขา เทคโนโลยีคอมพิวเตอร์อุตสาหกรรม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาหุ่นยนต์เคลื่อนที่ได้เอง ระยะที่ 3

THE DEVELOPMENT OF MOBILE ROBOT IN STAGE III

ผู้จัดทำ

1. นาย สมบูรณ์ มาตรการจันทร์ 34162176
2. นาย สุทัศน์พงษ์ แก้วคอนไทย 34162178
3. นาย เอกชัย สันกำแพง 34162181


.....อาจารย์ที่ปรึกษา
(อาจารย์ เกษตร์ ศิริสันติสัมฤทธิ์)

ทิวชัยวิธานิพนธ์

นักศึกษ

ระดับการศึกษา

อาจารย์ที่ปรึกษา

ปีการศึกษา

การพัฒนาหุ่นยนต์เคลื่อนที่ได้เอง ระยะที่ 3

สมบูรณ์ มาตรคำจันทร์

สุทัศน์พงษ์ แก้วคอนไทย

เอกชัย สันกำแพง

อุตสาหกรรมศาสตรบัณฑิตทางเทคโนโลยีคอมพิวเตอร์-

อุตสาหกรรม

เกษตร ศิริสันติสัมฤทธิ์

2535

บทคัดย่อ

ในปัจจุบันบทบาทของยานพาหนะที่เคลื่อนที่ได้เอง มีปรากฏอยู่ในหลายวงการ เช่น งาน อุตสาหกรรม, งานทางทหาร, งานสำรวจ ฯลฯ จึงทำให้ต้องมีการพัฒนายานพาหนะเหล่านี้ให้ทันสมัยและตอบสนองต่อความต้องการของผู้ใช้ได้

เป้าหมายหลักของการพัฒนาหุ่นยนต์เคลื่อนที่ได้เอง คือ สามารถเคลื่อนที่จากจุดเริ่มต้นไปยังตำแหน่งใด ๆ ในสภาพแวดล้อมที่กำหนดแล้วเคลื่อนที่กลับมายังจุดเริ่มต้นได้ โดยไม่ชนกับวัตถุ กีดขวางใด ๆ จากที่ได้มีการพัฒนาแล้ว 2 ระยะ หุ่นยนต์สามารถที่จะหลบหลีกวัตถุได้ วัดระยะทาง การเคลื่อนที่ได้ วัดระยะทางระหว่างวัตถุกับหุ่นยนต์ได้ และปรับมุมเลี้ยวได้ 60 องศา แต่ยังมี ความผิดพลาดอยู่มาก ในปริิธานิพนธ์การพัฒนาหุ่นยนต์เคลื่อนที่ได้เองในระยะที่ 3 จึงได้ปรับปรุงแก้ไขความผิดพลาดที่มีอยู่ในระบบให้มีความถูกต้องและเที่ยงตรงสูงขึ้น โดยการเปลี่ยน และจัดระบบควบคุมฮาร์ดแวร์ทางอิเล็กทรอนิกส์และดิจิทัลทั้งหมด ซึ่งความสามารถที่ได้ปรับปรุงใหม่นี้ สามารถสั่งให้หุ่นยนต์เคลื่อนที่ได้ด้วยระยะทางที่แม่นยำตามคำสั่ง สามารถเลี้ยวได้ละเอียดถึง 1 องศา และหลบหลีกวัตถุกีดขวางได้โดยไม่ต้องหยุดการเคลื่อนที่

THESIS TITLE	The Development of Mobile Robot in Stage III
STUDENT NAME	Somboon Martkamjan 34.162176 Supatpong Keawkonthai 34.162178 Ekachai Sankampang 34.162181
LEVEL OF STUDY	Bachelor's Degree of Industrial Computer- Technology. Department of Instrumentation Technology.
ADVISOR	Kaset Sirisantiamrid
ACADEMIC YEAR	1992

Abstract

At present, the role of Autonomous Vehicles are in many fields, such as industry, military, civil etc. These must having development of the vehicle which is updated and can be responded to the advantage of users.

The goal of the development of the Mobile Robot is to permits it can moves from the original point to any locations in the determined environment, then turns back to the original point without colliding with the obstacles. In two primitive stages of the development, the robot provided with avoiding collision to obstacles, measuring distance of movement, measuring distance between the robot and obstacles, and turning with angles of 60 degrees, but these remained with many errors. For the project of the Development of Mobile Robot in stage III have been enhanced the errors in the system to adding with higher corrections and precisions by changing and organizing in all of analog and digital of hardware control systems. These modification can be taken the robot can moved with the accurat distance, turn with an accuracy in 1 degree, and avoid collision with obstacles without stopping the car.

คำนำ

ในปัจจุบันบทบาทของยานพาหนะที่เคลื่อนที่ได้เอง (Autonomous Vehicles) มีปรากฏอยู่ในหลายวงการ เช่นงานอุตสาหกรรม, งานทางทหาร,งานสำรวจ ฯลฯ จึงทำให้ต้องมีการพัฒนายานพาหนะเหล่านี้ให้ทันสมัยและตอบสนองต่อความต้องการของผู้ใช้ได้.

Mobile Robot ก็จัดอยู่ในประเภทหนึ่งของยานพาหนะที่เคลื่อนที่ได้เองซึ่งเป็นที่ได้รับความสนใจเป็นอย่างมากในต่างประเทศ แต่สำหรับในประเทศไทยแล้วอาจจะกล่าวได้ว่ายังอยู่ในขั้นเริ่มต้นอันเนื่องมาจากสาเหตุต่าง ๆ ซึ่งพอจะสรุปได้ดังต่อไปนี้

1. รัฐบาลยังคงให้ความสนใจกับงานวิจัยด้านนี้น้อย
2. ความจำเป็นในการใช้แรงงานหุ่นยนต์ยังคงเป็นงานที่ใช้งบประมาณสูง
3. อุปกรณ์และเทคโนโลยีที่นำมาใช้ในการสร้างยังคงมีราคาแพง

จากเหตุผลข้างต้นจึงทำให้ Mobile Robot เป็นเพียงงานวิจัยในห้องทดลองเท่านั้น

ในปี พ.ศ. 2533 ณ ภาควิชาเทคโนโลยีการควบคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ได้มีการสร้าง Mobile Robot ขึ้น โดยความคิดริเริ่มของ อ. เกษตร์ ศิริสันติสัมฤทธิ์ และได้มีการพัฒนาขีดความสามารถมาเรื่อย ๆ จนถึงปัจจุบัน(พ.ศ. 2535)และประสบผลสำเร็จอยู่ในขั้นหนึ่ง แต่ก็ยังไม่สามารถทำงานได้ตามวัตถุประสงค์ที่ต้องการ คือ สามารถเคลื่อนที่จากจุดเริ่มต้นไป ณ ตำแหน่งใด ๆ ในสภาพแวดล้อมที่กำหนด แล้วเคลื่อนที่กลับมายังตำแหน่งเดิมโดยไม่ชนกับวัตถุกีดขวางใดๆ จึงทำให้ต้องมีการพัฒนาในระยะที่ 3 ขึ้นและในระยะนี้ได้มีการจัดระบบใหม่ทั้งหมด จากเดิมใช้ CPU หลัก 1 ตัวควบคุมการเคลื่อนที่ของล้อหน้าและล้อหลัง และใช้ CPU รอง 1 ตัว ควบคุมระบบนำร่อง ก็เปลี่ยนมาใช้ CPU หลัก 1 ตัวควบคุม CPU รอง 3 ตัว คือ CPU ควบคุมระบบนำร่อง, CPU ควบคุมระบบขับเคลื่อนล้อหลัง,และ CPU ควบคุมการเลี้ยวของล้อหน้า ซึ่งการจัดระบบแบบนี้จะมีความเป็นอิสระมากกว่าเดิมและคาดว่าจะสามารถทำให้หุ่นยนต์สามารถเคลื่อนที่ได้เองตามต้องการ.

สารบัญ

หน้า

บทคัดย่อ

Abstract

คำนำ

บทนำ.....	1
บทที่ 1. ความเป็นมาของโครงการ.....	7
1.1 ข้อดีและข้อเสียของระบบเดิม.....	8
1.2 หัวข้อหลักของการพัฒนาระยะที่ 3.....	8
บทที่ 2. โครงสร้างของ Mobile Robot.....	10
2.1 โครงสร้างทางแมคคาไนค.....	10
2.2 โครงสร้างทางระบบควบคุม.....	12
บทที่ 3. ระบบนำร่อง(Navigation System).....	14
3.1 หลักการทำงานของระบบนำร่อง.....	14
3.2 การออกแบบ.....	15
3.3 การทดลอง.....	17
3.4 สรุปผลการทดลอง.....	18
บทที่ 4. การควบคุมตำแหน่งล้อหน้า.....	23
4.1 Optical Encoder.....	26
4.2 ชุดควบคุม DC motor.....	29
บทที่ 5. ระบบขับเคลื่อนล้อหลัง.....	33
5.1 ความเป็นมา.....	33
5.2 การพัฒนาระบบ.....	33
5.3 อธิบายรายละเอียดการพัฒนา.....	34
5.4 การตรวจจับความเร็วและระยะทาง.....	34
5.5 ทฤษฎีการหาระยะทาง.....	35

5.6 ทฤษฎีการตรวจจับความเร็ว.....	35
บทที่ 6. การติดต่อสื่อสารข้อมูล.....	41
6.1 การใช้งาน 8255 ใน Mode 1.....	43
6.2 Control Word.....	46
6.3 Polling versus interrupts.....	50
6.4 รูปแบบการ Set Bit.....	51
6.5 โครงสร้างการจัดลำดับการอินเทอร์รัพท์.....	51
6.6 การอินเทอร์รัพท์ทางซอฟต์แวร์.....	53
บทที่ 7. การทดลองการหลบหลีกสิ่งกีดขวางของหุ่นยนต์.....	55
7.1 ลักษณะการโปรแกรม.....	55
7.2 การผิดพลาดจากตำแหน่งเป้าหมาย.....	55
7.3 การโปรแกรมใช้งาน.....	61
7.4 ฟังก์ชันการควบคุม.....	61
บทที่ 8. สรุปผลการทำงานและการทดลอง.....	63
8.1 สรุปผลการทำงานของระบบนำร่อง.....	63
8.2 ข้อดีของระบบนำร่องโดยใช้อัลตราโซนิก.....	63
8.3 ข้อบกพร่องของระบบนำร่องโดยใช้อัลตราโซนิก.....	63
8.4 แนวทางในการพัฒนาระบบนำร่อง.....	64
8.5 สรุปผลการทำงานของระบบบังคับเลี้ยว.....	64
8.6 ข้อเสนอแนะสำหรับการพัฒนาในขั้นต่อไป.....	64
8.7 สรุปผลการทำงานของระบบควบคุมความเร็ว.....	65
8.8 สรุป.....	65
8.9 ข้อเสนอแนะ.....	66
กิตติกรรมประกาศ.....	67
เอกสารอ้างอิง.....	68

บทนำ

ความหมายของคำว่าหุ่นยนต์ (Robot)

ความหมายของคำว่า "Robot" นั้นได้มีผู้ให้ความหมายไว้หลายความหมายต่างกัน องค์กรที่เกี่ยวข้องกับหุ่นยนต์ทั้งหลายต่างก็ได้ให้คำจำกัดความของหุ่นยนต์อุตสาหกรรมของตนเอง และถึงแม้ว่า จะมีส่วนที่เหมือนกันอยู่บ้าง แต่ก็ต่างกันพอที่จะทำให้การสับสน ว่าควรจะเรียก Robot ตามความหมายไหนดี องค์กรต่างๆ เหล่านี้ได้แก่

1. RIA (Robotics International Association) แห่งประเทศสหรัฐอเมริกา ได้ให้คำจำกัดความของคำว่า "Robot" ว่า "Reprogrammable multifunctional manipulator designed to move materials, parts, tools or specialized devices through variable programmed motion for performance of variety of tasks" จะเห็นว่าความหมายนี้เน้นเฉพาะ Manipulator เท่านั้น ซึ่งเป็นเพียงส่วนประกอบหนึ่งของ Robot มิใช่ความหมายทั้งหมดของ Robot

ล่าสุด RIA ได้ให้นิยามคำว่า "Robot" ใหม่ในปี 1987 ว่า "Reprogrammable Multifunctional machine designed to manipulate material, parts, tools or specialized devices through variable programmed motion for the performance of variety of tasks" ซึ่งความหมายนี้ ดีกว่าความหมายแรก แต่ยังคงต้องการชี้ให้เห็นว่า ต้องเป็น Manipulator มีแขนสำหรับหยิบจับชิ้นงานได้

2. BRA (British Robot Association) ก็ได้ให้นิยามที่คล้ายกับ RIA คือ " An industrial robot is a reprogrammable device designed to both manipulate and transport parts, tools or specialized manufacturing implements through variable programmed motions for the performance of specific manufacturing tasks"

คำนิยามของอังกฤษไม่ได้ระบุว่า Robot คือ Manipulator เพียงแต่สามารถทำงานเสมือน Manipulator ในคำนิยามทั้งสอง "แขน" เป็นเพียงส่วนประกอบของ Robot เท่านั้น นิยามของอังกฤษได้เพิ่มเติมความสามารถในการขนส่ง แต่ต้องใช้ในโรงงานอุตสาหกรรมเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นถ้า Robot ที่ไม่ได้ทำงานในทางอุตสาหกรรม หรือถูกนำมาใช้งานในทางธุรกิจ ก็ไม่ถือว่าเป็น Robot ตามนิยามข้างต้นนี้

3.SWIRA (The Swedish Industrial Robot Association) ได้ให้นิยามของ "Robot" ไว้ว่า " An automatically controlled, reprogrammable, multi-purpose manipulative machine with or without locomotion for use in industrial automation applications" ซึ่งระบุว่า "Robot" สามารถเคลื่อนที่ได้ ใช้งานอุตสาหกรรม ทั่วไป ไม่เฉพาะงานในโรงงานเท่านั้น

4.JIRA (Japanese Industrial Robot Association) ได้ให้คำจำกัดความของ หุ่นยนต์อุตสาหกรรม ในมาตรฐานอุตสาหกรรม หมายเลข JIS B0134-1986 และได้แบ่ง ระดับของหุ่นยนต์ออกเป็น class ต่างๆดังนี้

- Class 1: Manual manipulator-controlled by the operator (such as a teleoperated robot)
- Class 2: Fixed sequence robot-a stand alone robot operating in sequence and performing a predetermined and unchanging task
- Class 3: Variable sequence robot-similar to class 2, but with preset data that is easily modified
- Class 4: Playback robot-the robot is trained by a human operator and then repeatedly performs the required steps in sequence
- Class 5: Numeric control robot-the human operator controls the robot through changing a program or entering numbers, rather than through a training mode
- Class 6: Intelligence robot-the robot has the means to understand its environment and adapt to changing conditions as it completes its tasks

โดยมากแล้ว หุ่นยนต์ในประเทศญี่ปุ่นอยู่ใน class 2 และ 4 จากการศึกษาพบว่า ในปี 1982 ญี่ปุ่นมีหุ่นยนต์ 42% ที่เป็นหุ่นยนต์ใน class 2 และ 39% เป็นหุ่นยนต์ใน class 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. องค์การหุ่นยนต์ของฝรั่งเศสที่มีชื่อว่า AFRI แบ่งหุ่นยนต์ออกเป็น 4 ประเภท บริษัทผลิตหุ่นยนต์ของฝรั่งเศส ก็เป็นผู้นำทางด้านการผลิต teleoperated Robot เช่นกัน และได้แบ่งเป็นประเภทต่าง ๆ เปรียบเทียบกับหุ่นดังนี้

Type A: เทียบเท่ากับ class 1 ของหุ่น

Type B: เทียบเท่ากับ class 2 และ class 3 ของหุ่น

Type C: เทียบเท่ากับ class 4 และ class 5 ของหุ่น เป็น first generation robot

Type D: เทียบเท่ากับ class 6 ของหุ่น เป็น second generation robot

6. Webster's Dictionary ได้นิยามคำว่า Robot ว่า " A robot is an automatic device that performs functions ordinarily ascribed to human beings" จากนิยามนี้ เครื่องซักผ้าก็อาจถูกนิยามว่าเป็นหุ่นยนต์ด้วย

ประวัติของหุ่นยนต์

เรื่องราวของมนุษย์หุ่นยนต์ หรือมนุษย์กลไก มีบรรพบุรุษมาจาก หุ่นยนต์ C3PO จากภาพยนตร์เรื่อง Star Wars และเรื่องราวจากเทพนิยายกรีกและจากภาพยนตร์ที่สร้างมาจากนิยายวิทยาศาสตร์ เช่น George Melies นักสร้างภาพยนตร์ชั้นนำของฝรั่งเศส สร้างภาพยนตร์ชื่อ The Clown and Automation ในปี 1897 เมื่อหลายศตวรรษที่ผ่านมา ได้มีผู้เชี่ยวชาญทางด้านกลไก ได้พัฒนาหุ่นยนต์ขึ้นมาหลายรุ่น ถึงแม้คำว่า "Robot" จะยังไม่มีใช้ จนกระทั่งถึงศตวรรษที่ 20 ก็ตาม ในเวลาต่อมาได้มีผู้ประดิษฐ์มนุษย์กลไกที่เรียกว่า "Automata" และได้นำลักษณะการเคลื่อนไหวนั้นมาใช้งาน เช่น เคาะระฆังบอกชั่วโมง

ในปี 1700 นักประดิษฐ์สมองใสก็ได้เพิ่มความสามารถอีกหลายอย่างให้กับ "Automata" จนเกือบจะเรียกได้ว่าเป็นสุดยอดของระบบกลไกอัตโนมัติเลยทีเดียว นักประดิษฐ์คนนั้นก็คือ Pierre Jacquet-Droz เป็นนักประดิษฐ์ตุ๊กตา ผลงานชิ้นหนึ่งของเขาคือ "The Scribe" (แปลว่า นักเขียน) ซึ่งยังเก็บรักษาไว้ที่พิพิธภัณฑ์ Neuchatel (ประเทศสวิตเซอร์แลนด์) เขาสร้างเป็นรูปชายหนุ่มที่กำลังนั่งเขียนหนังสืออยู่บนโต๊ะเขียนหนังสือ ตุ๊กตาตัวนี้สามารถเอปากกา ลงจุ่มในขวดหมึก จากนั้นก็เขียนเป็นตัวอักษร ที่มีความยาวขนาด 40 ตัวอักษร ตุ๊กตาตัวนี้มีจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำเสนอใจอยู่ที่การเขียนโปรแกรมโดยการเปลี่ยนตำแหน่งของลูกเบี้ยว และ คาน (เช่นเดียวกันกับการเปลี่ยนคำสั่งในคอมพิวเตอร์) ตึกตาตัวนี้สามารถเขียนข้อความใดๆก็ได้

ถึงแม้ว่าการสร้างตึกตาอัตโนมัติจะมีมานานแล้วก็ตาม แต่ก็ไม่มีใครว่าได้มีการนำเอาเทคโนโลยีเหล่านั้นมาใช้กับงานอุตสาหกรรมแต่อย่างใด จนกระทั่งกลางปี 1950 หุ่นยนต์อัตโนมัติก็ได้ถูกนำมาใช้ในงานอุตสาหกรรมที่เป็นอันตรายแทนมนุษย์

คำว่า Robot นั้นถูกใช้เป็นครั้งแรก โดยนักประพันธ์บทละคร ชาวเชคโกสโลวเกีย ชื่อว่า Corel Capek ซึ่งเขียนเรื่อง OIipee ขึ้นในปี 1917 แต่ก็ยังเป็นที่รู้จักน้อยกว่า ผลงานชิ้นโบว์แดงที่เขาประพันธ์ขึ้นภายหลัง ที่มีชื่อว่า R.U.R. หรือมีชื่อเต็มว่า Rossums' Universal Robots

R.U.R. ถูกนำไปแสดงครั้งแรกที่ Prague ในปี 1921 และได้รับความสนใจจากผู้ชมเป็นอย่างมาก เมื่อครั้งเปิดการแสดงครั้งแรกที่ Newyork ในปี 1922 ได้พัฒนาให้มีขนาดเล็ก มีลักษณะคล้ายคนและเชื่อฟังคำสั่งผู้เป็นนาย คำว่า Robot นั้นมาจากคำว่า Robota ซึ่งหมายถึงกรรมกรผู้ใช้แรงงานหรือทาส และได้ใช้ศัพท์คำนี้กับวงการวิทยาศาสตร์และวงการภาพยนตร์ในเวลาต่อมา

ในช่วงสงครามโลกครั้งที่ 2 คำว่า Robot มีความหมายรวมไปถึง เครื่องบินทิ้งระเบิด ระบบนำร่องอัตโนมัติ ดังจะเห็นได้จากคำว่า Robot bom ซึ่งแสดงให้เห็นความสัมพันธ์อยู่สองระดับ คือการพัฒนาคอมพิวเตอร์และการควบคุมการทำงานในระยะไกล หรือปัจจุบันที่รู้จักกันในชื่อ หุ่นยนต์ทำงานระยะไกล (Teleoperated robots หรือ Telechirs)

นับแต่นั้นก็ได้มีการพัฒนาหุ่นยนต์เรื่อยมา การพัฒนาในขั้นต่อมา ก็ได้มีการนำเอาระบบ Servo, ระบบควบคุมเชิงตัวเลข (Numerical control), การควบคุมโปรเซส(Process control) และหุ่นยนต์อุตสาหกรรม (Industrial robot) เข้าด้วยกัน ในปี 1954 หุ่นยนต์อุตสาหกรรมก็ถูกสร้างขึ้น และได้มีการจัดสิทธิบัตรหุ่นยนต์อุตสาหกรรมขึ้นเป็นครั้งแรก 2 ฉบับ จึงนับว่าเป็นยุครุ่งอรุณของหุ่นยนต์

ในปี 1954 บริษัทผลิตหมอน้ำรถยนต์แห่งหนึ่งได้ว่าจ้างบริษัท Planet Corporation of Lansing ในมลรัฐมิชิแกนให้ออกแบบระบบจับเข้าหล่อเพื่อตักของเหลวแล้วเทลงไปในแบบ บริษัท Planet ได้ใช้เวลาในการออกแบบ 3 ปี และได้งานชิ้นนี้ไปเผยแพร่ต่อสาธารณชนในงาน International Trade Fair on Automation ที่เมือง Stockholm ในปี 1957 และตั้งชื่อว่า "Planobot" ใช้ระบบกำลังแบบไฮดรอลิก ควบคุมด้วยระบบอิเล็กทรอนิกส์ทั้งหมด

ระหว่างปี 1954 นักประดิษฐ์ชาวอังกฤษที่ชื่อ Cyril W. Kenward ก็ได้สร้าง Manufacturing Apparatus ขึ้นมา เป็นแบบที่มี 2 มีเซนสามารถหมุนได้อิสระ 4 ทิศทางในระบบคาร์ทีเซียน

ปี 1961 ได้มีการผลิตหุ่นยนต์อุตสาหกรรมตัวแรกขึ้น มีลักษณะคล้ายกับหุ่นยนต์ของบริษัท Planet ซึ่งพัฒนามาใช้กับงานที่อันตรายและน่าเบื่อ โดย บริษัท General Motor และได้้นำออกแสดงในงานนิทรรศการที่สถาบัน Smithsonian รัฐวอชิงตัน

ในปี 1965 หลายบริษัทก็ให้ความสนใจกับระบบชาตูลลาด โดยเพิ่มเติมระบบนี้เข้าไปกับหุ่นยนต์ของบริษัทตน แต่ก็ยังเป็นเพียงก้าวเล็กๆเท่านั้น

มหาวิทยาลัยต่าง ๆ เริ่มหันมาวิจัยระบบที่เกี่ยวข้องกับหุ่นยนต์และมีความก้าวหน้าในเรื่องความสามารถในการเคลื่อนที่ของหุ่นยนต์ (Robot mobility), ระบบมองเห็น (Vision systems) และความฉลาด (Artificial Intelligence) ซึ่งเริ่มวิจัยกันที่ Stanford Research Institute (ปัจจุบันคือ SRI international) หุ่นยนต์เคลื่อนที่ (Mobile Robot) ตัวแรกนั้นก็ได้ถูกพัฒนาขึ้นมาโดย RSI ชื่อว่า Shakey เพราะว่ามันมีการเคลื่อนไปแล้วหยุดคิด จากนั้นจึงเคลื่อนที่ต่อไป แต่นั่นก็แสดงให้เห็นความชาญฉลาดขึ้นไปอีกระดับหนึ่งในขณะนั้น ผลงานที่สร้างชื่อให้กับ Shakey ก็คือ เมื่อให้มันเข้าหาเป้าหมายที่วางอยู่ในระดับสูงกว่า หรือวางบนชันลาดเนื่องจากว่าตัวหุ่นยนต์ติดอยู่บนฐานล้อ มันจึงไม่สามารถปีนขึ้นไปหาวัตถุนั้นได้ แต่มันจะยิงกระสุนใส่วัตถุเพื่อให้วัตถุหล่นลงมา จากนั้นจึงเข้าถึงวัตถุใน

หุ่นยนต์เคลื่อนที่อัตโนมัติ คือผลจากการศึกษาระบบควบคุมอัตโนมัติ (Automatic control) และระบบผู้เชี่ยวชาญ (Artificial Intelligence) ซึ่งเรียกว่า "การควบคุมแบบชาตูลลาด" (Artificial Control) เป็นศาสตร์ทางด้านวิศวกรรมผสมผสาน เข้ากับแนวความคิด ซึ่งมีเป้าหมายเพื่อการวิจัยและพัฒนาเพื่อที่จะนำเอาผลของการวิจัย ไปประยุกต์ใช้งาน และ เรียกได้ว่าเป็นหัวใจสำคัญของระบบเลยทีเดียว

ประโยชน์ของหุ่นยนต์นั้น ไม่เพียงแต่นำมาทำงานอันน่าเบื่อและอันตรายแทนคนเท่านั้น แต่เรายังได้นำเอา ความรวดเร็วของการประมวลผล ความแม่นยำในการทำงาน รวมทั้งความสามารถในการคงรูปแบบของงานนั้นเมื่อเมื่อต้องการทำซ้ำหลาย ๆ ครั้ง เรียกได้นำมาชดเชยข้อด้อย ของมนุษย์ ซึ่งมีแนวโน้มว่า จะมีการใช้แพร่หลายยิ่งขึ้นในอนาคต

ในยุคแรก ๆ นั้น คำว่า หุ่นยนต์อัตโนมัติ ซึ่งน่าจะเข้าใจตามความหมาย ว่า "อุปกรณ์ที่ทำงานได้โดยอิสระ" ยังไม่มี เนื่องจากหุ่นยนต์ในยุคนั้น ยังไม่ใช่หุ่นยนต์อัตโนมัติ ปกติการทำงานของหุ่นยนต์มนุษย์จะเป็นผู้ควบคุม หรือมีการวางโปรแกรมไว้ล่วงหน้า โดยที่หุ่นยนต์ไม่ได้ใช้ความ

คิดตัดสินใจเอง หุ่นยนต์อัตโนมัติที่แท้จริง จะต้องทำงานได้เองโดยอิสระ ไม่ต้องให้มนุษย์คอยควบคุม โดยมีคอมพิวเตอร์เป็นสมองสำหรับคิด และตัดสินใจมีแหล่งจ่ายพลังงานที่มีอายุการใช้งานยาวนานบรรจุอยู่ภายในและสามารถติดต่อสื่อสารกับโลกภายนอกได้เอง

แต่.. อะไรเป็นสิ่งที่ปัญหา เราลองมาพิจารณาว่าหลังจากที่ได้ ตัดการควบคุมของมนุษย์ออกไปทั้งหมด การรับรู้เรื่องราวทั้งหมดจะตกเป็นภาระของหุ่นยนต์ การที่เราจะเพิ่มตัวตรวจจับความรู้สึก หรือประสาทสัมผัสทั้ง 5 ให้กับหุ่นยนต์นั้นจึงเป็นเรื่องยาก ทั้งนี้และทั้งนี้ก็เนื่องมาจากข้อจำกัดทางการภาพของเจ้าตัวตรวจจับทั้งหลาย รวมทั้งวิธีการประมวลผล และการจัดจำของหุ่นยนต์ ซึ่งเป็นขีดจำกัดของการพัฒนาทางด้านนี้ ที่ต้องรอการเติบโตของเทคโนโลยี และการหาวิธีการใหม่ที่มีความรวดเร็วในการแก้ปัญหา ลดเวลาการประมวล เพื่อให้การตอบสนองเป็นแบบเวลาจริง (Real Time) มากยิ่งขึ้น หลังจากหุ่นยนต์ในยุคก่อนใช้วิธีการประมวลคำสั่งแบบกลุ่ม (Batch Processing) คือ ต้องมีการรอการประมวลผลของกระบวนการหนึ่งให้เสร็จ ก่อนที่จะประมวลผลกระบวนการอื่น ในปัจจุบันมีการเปลี่ยนมาใช้ในการประมวลผลแบบเวลาจริง ในทางอุตสาหกรรม การประมวลผลแบบเวลาจริงต้องตอบสนองแบบทันทีทันใด แต่ในทางปฏิบัติแล้ว จะมีการหน่วงเวลาเอาไว้ช่วงหนึ่ง (Delay Time)

โครงการพัฒนา "หุ่นยนต์เคลื่อนที่อัตโนมัติในระยะที่ 3" ก็ได้มีการนำแนวความคิด ของการประมวลผลแบบ Real Time มาพัฒนาหุ่นยนต์ ซึ่งได้มีการพัฒนามาแล้ว 2 ระยะ แต่ยังไม่ได้นำเอาระบบ Real Time มาใช้ การพัฒนาในระยะที่ 3 ได้นำเอาระบบ มัลติโปรเซสเซอร์มาใช้ด้วย เพื่อให้การประมวลผลเป็นไปอย่างรวดเร็ว ตอบสนองการเปลี่ยนแปลงของแมคคาณิกได้อย่างทันทั่วทั้ง การเคลื่อนที่ของ Mobile Robot จะไม่มีการสะดุด หรือหยุดเพื่อรอการประมวลผลหรือการตัดสินใจ

ในส่วนรายละเอียดต่าง ๆ ที่เกี่ยวข้องกันระบบนี้ จะได้กล่าวถึงในบทต่อไป

บทที่ 1

ความเป็นมาของโครงการ

เนื่องจากโครงการนี้เป็นการพัฒนาในระยะที่ 3 ในบทนี้จะได้นำเอาโครงร่างของโครงการในระยะที่ 2 มากล่าว โดยแสดงถึงข้อเปรียบเทียบระหว่างการพัฒนาในระยะที่ 2 กับการพัฒนาในระยะที่ 3

สำหรับเป้าหมายในการพัฒนาโครงการขั้นที่ 1, 2 และ 3 นั้นก็คล้ายกัน เพียงแต่การพัฒนาในแต่ละขั้น ประสบความสำเร็จต่างกัน เป้าหมายหลักของการพัฒนาโครงการนี้คือ

เป้าหมายที่ 1. ให้หุ่นยนต์เคลื่อนที่ไปในทิศทางต่าง ๆ โดยการรับคำสั่งจากคีย์บอร์ด (Keyboard) และมีการแสดงระยะทาง ทิศทาง ความเร็วได้ในขณะที่กำลังเคลื่อนที่ ซึ่งตามเป้าหมายการพัฒนาในระยะที่ 3 นับได้ว่าประสบความสำเร็จถึง 100 % หลังจากที่ได้ทดสอบแล้ว

เป้าหมายที่ 2. โปรแกรมคำสั่งต่าง ๆ เช่น ทิศทางและระยะทาง แล้วให้หุ่นยนต์เคลื่อนที่ตามโปรแกรมที่วางไว้ โดยยังไม่ให้หุ่นยนต์คิดและตัดสินใจแต่อย่างใด ในการทดสอบ ได้กำหนดสภาวะแวดล้อมเอาไว้โดยไม่ให้มีสิ่งกีดขวาง ซึ่งตามเป้าหมายการพัฒนาในระยะที่ 3 นับได้ว่าประสบความสำเร็จถึง 100 % หลังจากที่ได้ทดสอบแล้ว เช่นกัน

เป้าหมายที่ 3. โปรแกรมคำสั่งต่าง ๆ เช่น ทิศทางและระยะทาง แล้วให้หุ่นยนต์เคลื่อนที่ตามโปรแกรมที่วางไว้ แต่ไม่มีกำหนดสภาวะแวดล้อมล่วงหน้า โดยหุ่นยนต์ต้องตัดสินใจหลบหลีกสิ่งกีดขวางเอง เมื่อมีสิ่งกีดขวางมากขึ้นเส้นทาง โดยการปล่อยให้หุ่นยนต์เคลื่อนที่อย่างอิสระ

ซึ่งตามเป้าหมายนี้ได้ผลประมาณ 80% เนื่องจากการเขียนโปรแกรมให้หุ่นยนต์ตัดสินใจเองนั้น เป็นเรื่องที่ต้องใช้เวลาในการพัฒนา ซึ่งก็มีอัลกอริทึม ที่เกี่ยวกับเรื่องนี้อยู่หลายวิธี จึงต้องหาวิธีที่เหมาะสมมาใช้ พบว่าวิธีของ Path planning นั้นมีความเหมาะสมมากที่สุด เนื่องจากง่ายต่อความเข้าใจ และใช้เวลาน้อย ส่วนวิธีของ Histogram นั้นค่อนข้างซับซ้อน และต้องมีฐานข้อมูลรองรับ จึงไม่พิจารณา

1.1 ที่มาและข้อดี-ข้อเสียของระบบเดิม

มีการพัฒนามาแล้ว 2 ระยะ ติดต่อกันคือปี 2533-2534 มีข้อดีและข้อเสียดังนี้

ข้อดี

1. ระบบนำร่องสามารถตรวจจับการมีอยู่ของวัตถุได้
2. ระบบควบคุมล้อหลังใช้ระบบเฟสล็อกกลูบ ทำให้ทอร์ค (Torque) ที่ความเร็วใด ๆ คงที่ตลอด

ข้อเสีย

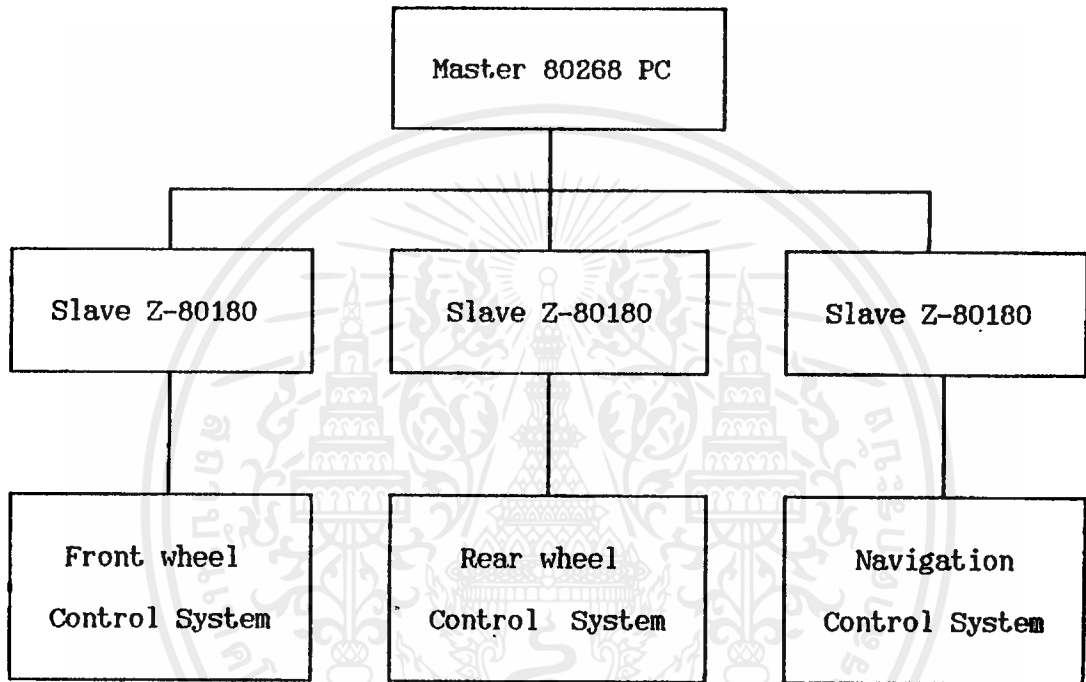
1. ระยะทางที่วัดได้ระหว่างตัวรถกับวัตถุที่ขวางมีข้อผิดพลาดที่ไม่เป็นเชิงเส้น
2. การออกตัวของรถมีอาการกระตุก
3. ความเร็วของการเคลื่อนที่ไปข้างหน้าและการถอยหลังไม่เท่ากันเมื่อเราใช้รหัสควบคุมหมายเลขเดียวกัน
4. การควบคุมมุมการเลี้ยวของล้อหน้ามีความผิดพลาดสูง เนื่องจากการใช้ตัวเซ็นเซอร์แบบรีซิสเตอร์ทำให้ค่าที่ตรวจจับได้ไม่เป็นเชิงเส้น
5. มีการใช้ Master CPU ควบคุมการเลี้ยวของล้อหน้าและระบบควบคุมความเร็วของล้อหลัง และ Slave CPU ควบคุมระบบนำร่อง ทำให้ Master CPU ต้องแบ่งเวลาในการประมวลผลระบบใหญ่ มาทำงานในระบบทั้งสอง ซึ่งส่งผลให้การทำงานไม่เป็น Real Time

1.2 หัวข้อหลักของการพัฒนาในระยะที่ 3

1. ปรับปรุงแก้ไขวงจรและโปรแกรมของระบบวัดระยะทาง ทำให้สามารถวัดระยะทางได้ถูกต้องแม่นยำมากกว่าเดิม
2. ปรับปรุงแก้ไขวงจรควบคุมการทำงานของล้อหลังทั้งหมดเพื่อให้สามารถควบคุมความเร็วและระยะทาง แรงบิด ได้เที่ยงตรงแน่นอน
3. ปรับปรุงแก้ไขวงจรควบคุมการทำงานของล้อหน้าทั้งหมด เพื่อให้การควบคุมตำแหน่งการเลี้ยวที่แม่นยำ มีความละเอียดมากยิ่งขึ้น
4. สร้างระบบตรวจจับตำแหน่งการเลี้ยวของล้อหน้าและตรวจจับความเร็วล้อหลังขึ้นมาใหม่
5. เปลี่ยนระบบ CPU ในการประมวลผลและควบคุมระบบโดยใช้ Master 1 ตัว ควบคุมการทำงานของ Slave อีก 3 ตัว Slave ทั้ง 3 นั้น แต่ละตัวควบคุมระบบนำร่อง, ระบบควบคุม



การเขียนและระบบควบคุมความเร็วตามลำดับ โดยสามารถเขียนเป็นบล็อกไดอะแกรมได้ ดังรูป
รูป 1.1



รูปที่ 1.1 บล็อกไดอะแกรมของระบบ ของ Mobile Robot

บทที่ 2

โครงสร้างของ Mobile Robot

โครงสร้างของ Mobile Robot ที่เรากำลังพัฒนาอยู่ สามารถแบ่งออกเป็นสองส่วน
ใหญ่ ๆ คือ

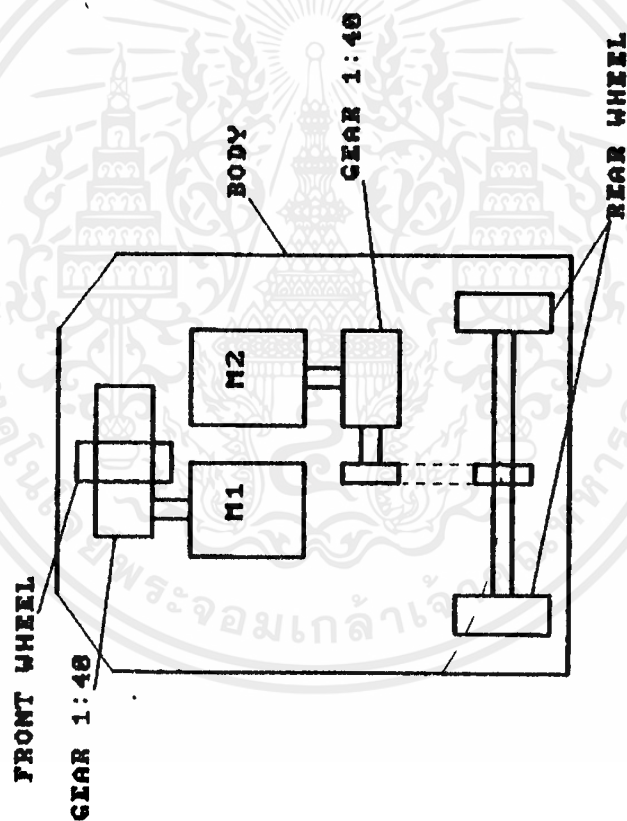
1. โครงสร้างทางแมคคาไนค (Mechanical)
2. โครงสร้างทางระบบควบคุม

2.1 โครงสร้างทางแมคคาไนค

โครงสร้างทางแมคคาไนคอาจแบ่งเป็น 3 ส่วนคือ

1. ตัวรถ
2. มอเตอร์บังคับการเลี้ยวของล้อหน้า M1
3. มอเตอร์ขับเคลื่อนล้อหลัง M2

เราสามารถแสดงโครงสร้างทางแมคคาไนคได้ดังรูปที่ 2.1 หน้า 12 M1, M2 เป็น ตีชี
มอเตอร์ซึ่งควบคุมด้วยระบบ phase locked loop การเชื่อมต่อระบบส่งกำลังจากมอเตอร์ไป
ยังล้อ ใช้เฟืองทด ที่มีอัตราทด 1:40 ทั้งล้อหน้าและล้อหลัง น้ำหนักทั้งหมดเมื่อไม่รวมแบตเตอรี่
ประมาณ 60 กิโลกรัม เนื่องจากโครงสร้างทำด้วยเหล็ก จึงทำให้มีน้ำหนักมาก



รูปที่ 2.1 แสดง โครงสร้างทางแมคคาไนคของหุ่นยนต์

Title MECHANICAL STRUCTURE

Size	Number	Revision
A4		
Date: 4-MAY-1993	Sheet: 1	Drawn by: SONGBOON
File: 80008871		

2.2 โครงสร้างทางระบบควบคุม

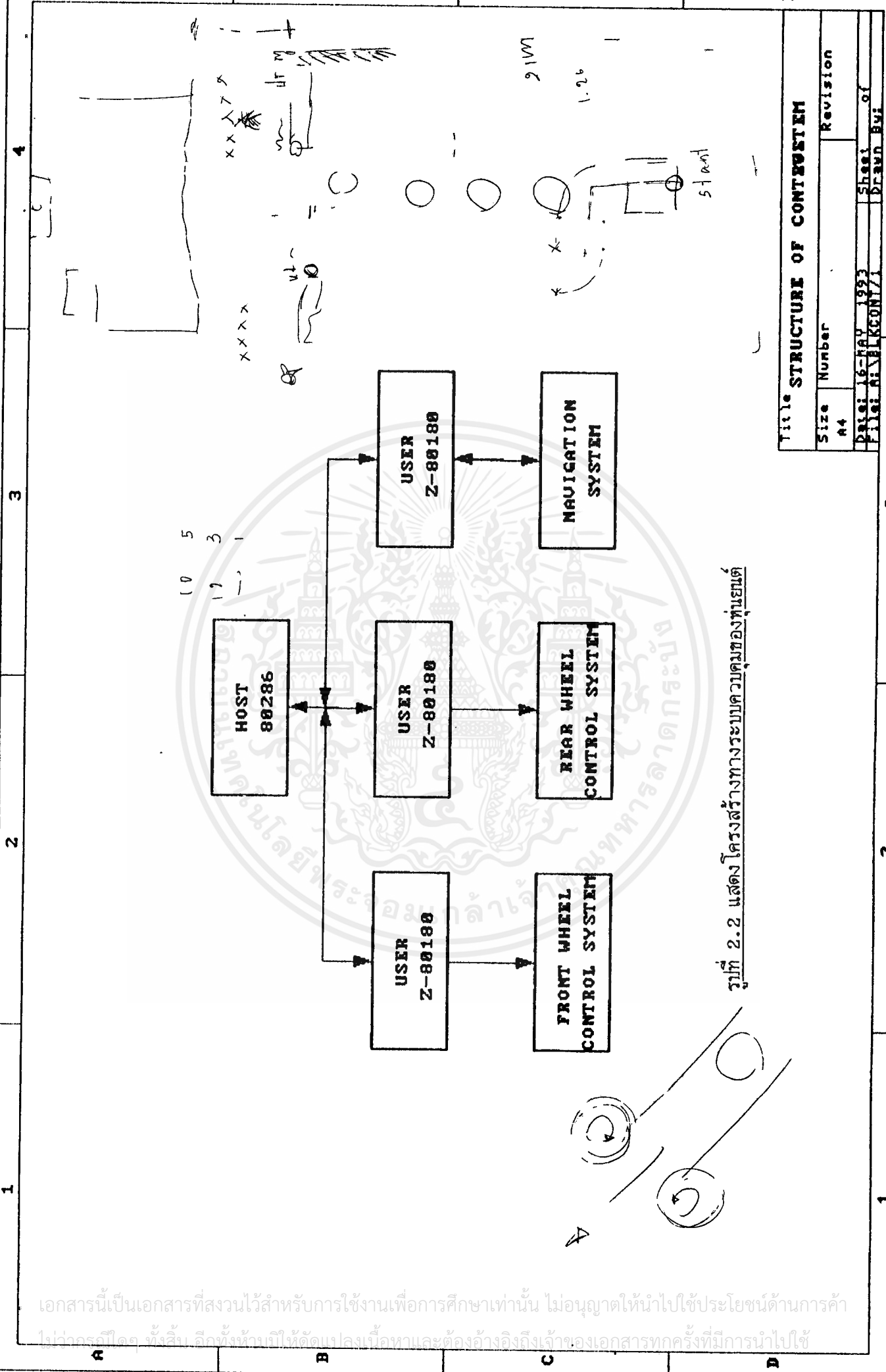
ระบบควบคุมประกอบด้วยส่วนที่เป็นคอมพิวเตอรืควบคุม, อนุาล็อกควบคุม และระบบนำร่อง คอมพิวเตอรืควบคุม จากผลการทดลองของโครงการนี้ในระยะที่ 1 และ 2 เราได้พบข้อดี และข้อเสียที่เกิดขึ้นในระบบคอมพิวเตอรืควบคุมหลายประการ เราจึงได้ออกแบบระบบควบคุมขึ้นมาใหม่ โดยที่เราคาดหวังไว้ว่าระบบใหม่นี้จะสามารถทำงานได้ดีและเป็นไปในลักษณะ Real Time มากกว่าในระยะก่อน ๆ ซึ่งในปริณญาณเณธ์นี้เราใช้ CPU เป็นตัวควบคุม 2 เบอร์คือ 80286 และ Z-80180 โดยเรากำหนดให้ 80286 เป็น MASTER (ตัวแม่) และ Z-80180 เป็น SLAVE (ตัวลูก).

อนุาล็อกควบคุม เราให้อาเล็อกควบคุมทำหน้าที่ควบคุมตามโครงสร้างทางแมคคานิค 2 ส่วน คือ

1. ระบบควบคุมการเลี้ยวของล้อหน้า
 2. ระบบควบคุมการขับเคลื่อนของล้อหลัง
- ซึ่งทั้งสองระบบนี้ทำงานด้วยระบบ phase locked loop.

ระบบนำร่อง เราใช้ ultra sonic เป็นส่วนช่วยในการนำร่อง และวัดระยะทางระหว่างวัตถุกับตัวหุ่นยนต์.

โครงสร้างของระบบควบคุมสามารถเขียนเป็นบล็อก ไดอะแกรมได้ดังรูปที่ 2.2 หน้า 13



รูปที่ 2.2 แสดงโครงสร้างทางระบบควบคุมของหุ่นยนต์

STRUCTURE OF CONTENTS		
Size	Number	Revision
A4		
Date: 16-MAY-1993		
File: A: ELKONTI		
Sheet: 1 of 1		
Drawn By:		

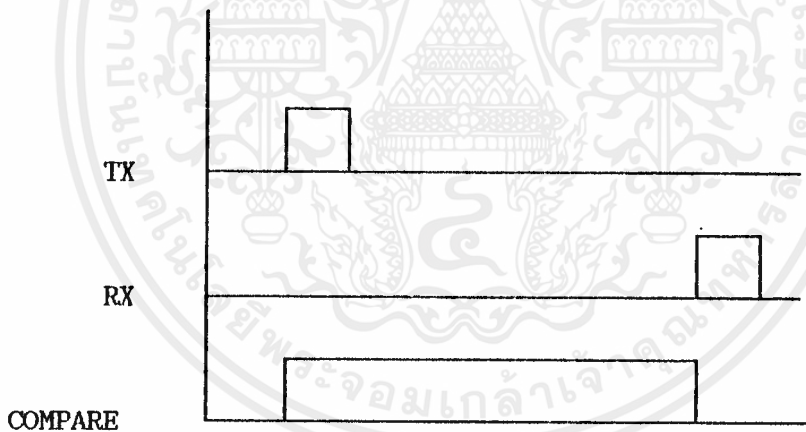
บทที่ 3

ระบบนำร่อง (Navigation System)

ระบบนำร่องเป็นส่วนประกอบที่สำคัญส่วนหนึ่งของ Mobile Robot อาจเปรียบได้ว่าเป็นเสมือนตาของหุ่นยนต์เลยทีเดียว เพราะนอกจากจะสามารถตรวจจับได้ว่ามีวัตถุกีดขวางอยู่หรือไม่แล้วยังสามารถวัดระยะห่างระหว่างวัตถุกีดขวางกับหุ่นยนต์ได้อีกด้วย

3.1 หลักการทำงานของระบบนำร่อง

โดยอาศัยหลักการส่งคลื่นสัญญาณไปในอากาศในทิศทางที่เราต้องการทราบระยะทาง จากวัตถุกับตัวหุ่นยนต์ แล้วคอยรับสัญญาณที่สะท้อนกลับมาจากวัตถุ จากนั้นก็นำสัญญาณที่ส่งไปและที่รับเข้ามานำมาเปรียบเทียบหาเวลาที่สูญหายไปดังแสดงในรูปที่ 3.1



รูปที่ 3.1 Timing Diagram ของระบบนำร่อง

ซึ่งระยะทางที่วัดได้นั้น จะแปรตามค่าเวลาที่หายไปโดยเราสามารถคำนวณหาระยะทางได้จากสูตร

$$S = VT/2$$

S : ระยะทางระหว่างวัตถุและตัว sensor เมตร(m)

T : เวลาที่ใช้ในการเดินทางไปกลับ.....วินาที(s)

V : ความเร็วของคลื่นในอากาศ.....เมตรต่อวินาที(m/s)

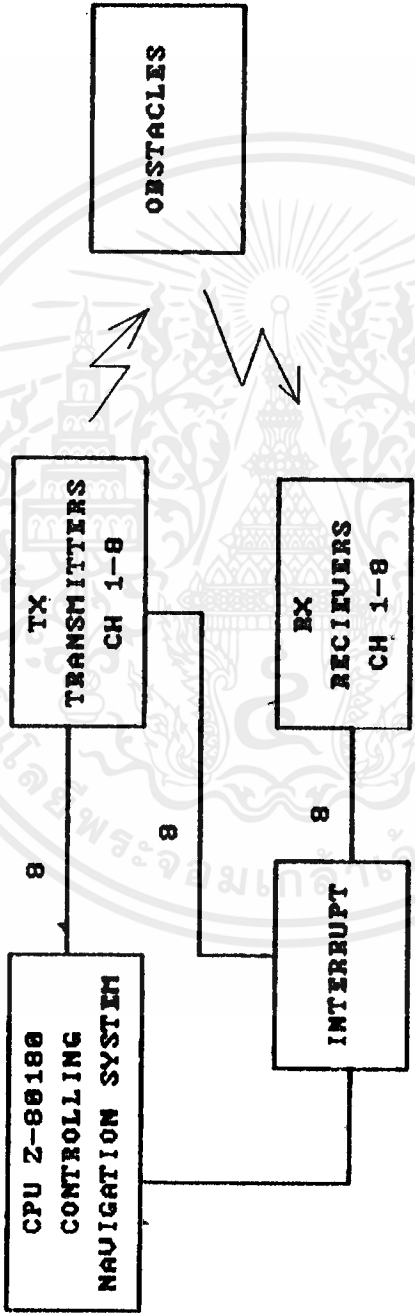
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบวงจร

บล็อกไดอะแกรม แสดงในรูปที่ 3.2 บล็อกไดอะแกรมของระบบนำร่อง จาก บล็อกไดอะแกรมเราใช้ CPU Z-80180 เป็นตัวควบคุมระบบนำร่อง ซึ่งประกอบด้วยชุดรับส่ง อุลตราโซนิก 8 ชุด, วงจรรวมการรับและส่ง 8 Channel, และวงจรส่งสัญญาณ Interrupt ให้กับ CPU ควบคุม.

เมื่อเราต้องการทราบระยะทางของวัตถุเราจะส่งข้อมูลจาก I/O PORT ของ CPU ไปยัง channel ที่เราต้องการทราบระยะทาง ตัวส่งสัญญาณ Tx ก็จะส่งสัญญาณที่มีลักษณะเป็นพัลส์ขนาด ความกว้าง 1 ms ออกไปในอากาศ (พร้อมกับสั่งให้ Timer ภายในของ CPU เริ่มนับและ ตัวส่งสัญญาณอินเตอร์รัพท์ในช่องที่ต้องการวัดระยะทางเริ่มทำงานเพื่อรอรับสัญญาณจาก Rx ไป กำเนิดสัญญาณอินเตอร์รัพท์) โดยการมอดูเลตไปกับความถี่ที่เป็นพัลส์ขนาด 40 KHz พัลส์ที่ส่งออกไปจะไปกระทบกับวัตถุที่กีดขวางแล้วสะท้อนกลับมายังภาครับ Rx เมื่อได้รับพัลส์ที่สะท้อนกลับมา แล้ว Rx ก็จะทำการดีมอดูเลตให้ได้พัลส์ขนาด 1 ms เหมือนกับที่ส่งออกไป แล้วส่งพัลส์ 1 ms นี้ออกไปยังภาค Interrupt เพื่อกำเนิดสัญญาณอินเตอร์รัพท์ให้กับ CPU และเมื่อ CPU ได้รับสัญญาณอินเตอร์รัพท์แล้วก็จะสั่งให้ Timer ภายในหยุดทำงาน เราก็ได้ค่าเวลา T ซึ่งอยู่ใน รูปเลขฐานสิบหก

รูปที่ 3.2 บล็อกไดอะแกรมของระบบนำร่อง



รูปที่ 3.2 แสดงบล็อกไดอะแกรมของระบบนำร่อง

Title BLOCK DIAGRAM NAVIGATION SYSTEM			
Size	Number	Revision	
A4			
Date: 4-MAY-1993	Sheet	of	
File: BLKNAV1	Drawn By:		

เนื่องจากการนับลงแต่ละครั้งของตัวตั้งเวลา (Timer) จะใช้เวลา $20 \Delta t$ หมายถึง จะใช้สัญญาณนาฬิกา (Clock) ของระบบ 20 ลูกในการนับลง 1 ครั้ง แต่เนื่องจาก $\Delta t = 9.216 \text{ MHz}$ ดังนั้นเราจะสามารถหาค่าเวลา T ได้จาก

$$T = [(/Timer) . 20] / \phi$$

$$= (/Timer) \times 2.17 \times 10^{-6} \quad \text{s}$$

Timer : เลขฐานสิบหกใน timer register

เราจะเห็นว่าค่า Timer ต้องผ่านการ complement ก่อนเนื่องจาก Timer มีการนับในลักษณะนับลง (จาก FFFFH - 0000H) จากนั้นเราสามารถคำนวณหาค่าระยะทางได้จาก

$$S = VT/2$$

$V = 346.625 \text{ m/s}$; ความเร็วของคลื่นอัลตราโซนิคอากาศที่อุณหภูมิ ห้อง 25 องศาเซลเซียส

ดังนั้น

$$S = 3.76 \times 10^{-4} \times (/Timer)$$

เมื่อเราได้ระยะทางมาแล้ว CPU Z-80180 ก็จะส่งระยะทางนี้ไปยัง CPU หลัก คือ 80286 เพื่อประมวลผลในการหลบหลีกวัตถุที่กีดขวางอยู่ แล้วนำข้อมูลที่ได้ไปปรับการขับเคลื่อนของล้อหลัง และบังคับการเลี้ยวของล้อหน้าต่อไป.

จากบล็อกไดอะแกรมเราสามารถออกแบบระบบและวงจรดังนี้

3.3 การทดลอง

จากการทดลองส่งสัญญาณเพื่อหาวัตถุกีดขวาง เราสามารถตรวจวัดได้ประมาณ 4 เมตร ในกรณีที่ เป็นวัตถุใหญ่ๆเท่านั้น แต่เราจะโปรแกรมให้ระบบนี้ทำงานตรวจวัดวัตถุในระยะ 3 เมตร ก็เป็นการเพียงพอในการประมวลผล เพื่อหลบหลีกวัตถุกีดขวาง และเราทดลองกับวัตถุกีดขวาง ขนาดต่างๆ ตามตารางผลการทดลอง ในหน้า 18

ตารางที่ 1

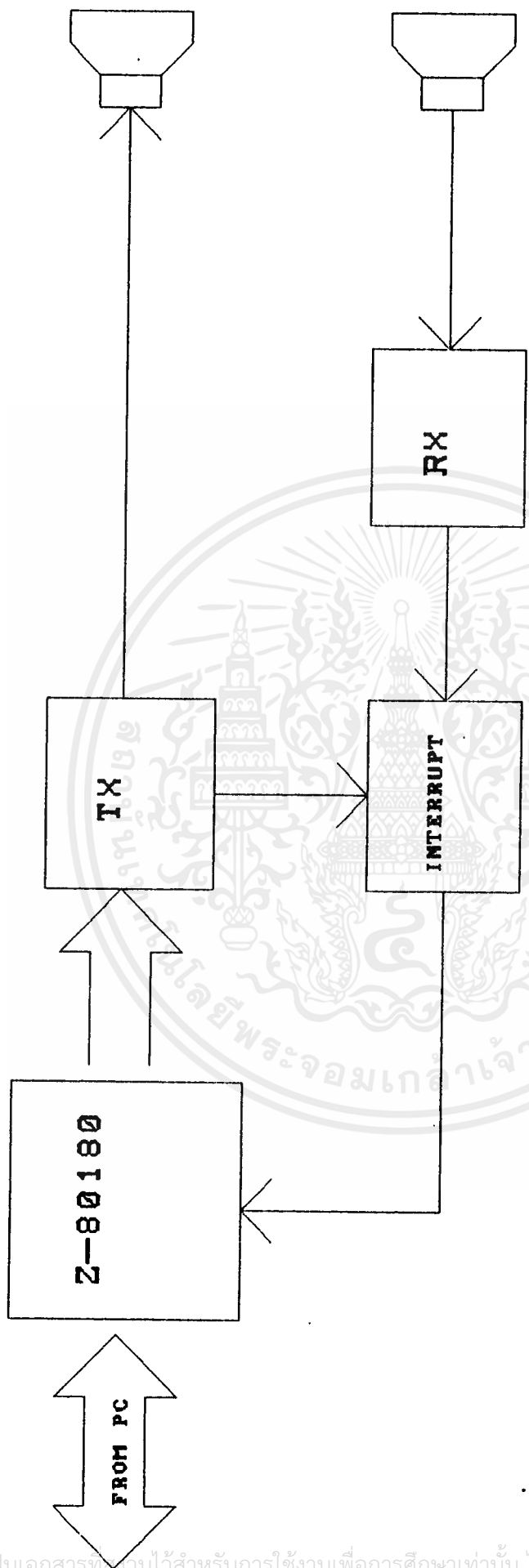
ระยะทาง จริง (cm)	ระยะทางที่วัดได้								
	ทรงเหลี่ยม (29x23cm)			ทรงกระบอก (h=25cm) (= 27.5 cm)			ชาโต๊ะ = 2.2 cm		
	timer (hex)	timer (ms)	distance (hex)	timer (hex)	timer (ms)	distance	timer (hex)	timer (ms)	distance (hex)
40	0497H	2.55	44	04E7H	2.72	47	04D6H	2.70	47
60	068BH	3.63	63	06C6H	3.76	65	06CEH	3.78	65
90	09B0H	5.38	93	09FBH	5.54	96	09E7H	5.50	95
120	0CD1H	7.11	123	0D3EH	7.36	127	0D29H	7.31	127
150	0FEBH	8.84	153	101EH	3.93	155	10A7H	9.25	160
180	1314H	1.06	184	1342H	1.06	185	-	-	-
210	1639H	1.23	214	1677H	1.25	216	-	-	-
240	195AH	1.41	275	1980H	1.42	245	-	-	-
270	1C9BH	1.59	304	1C9FH	1.60	279	-	-	-
300	1FA2H	1.76	275	1FFAH	1.78	308	-	-	-

จากตารางการทดลอง พบว่า การตรวจหาระยะทางของวัตถุทรงเหลี่ยมมีความผิดพลาดอยู่ในช่วง 3-5 cm โดยวัตถุทรงกระบอกและชาโต๊ะมีความผิดพลาดของระยะทางอยู่ในช่วง 5-9 cm มากกว่าวัตถุทรงเหลี่ยมอยู่ประมาณ 2-4 cm

3.4 สรุปผลการทดลอง

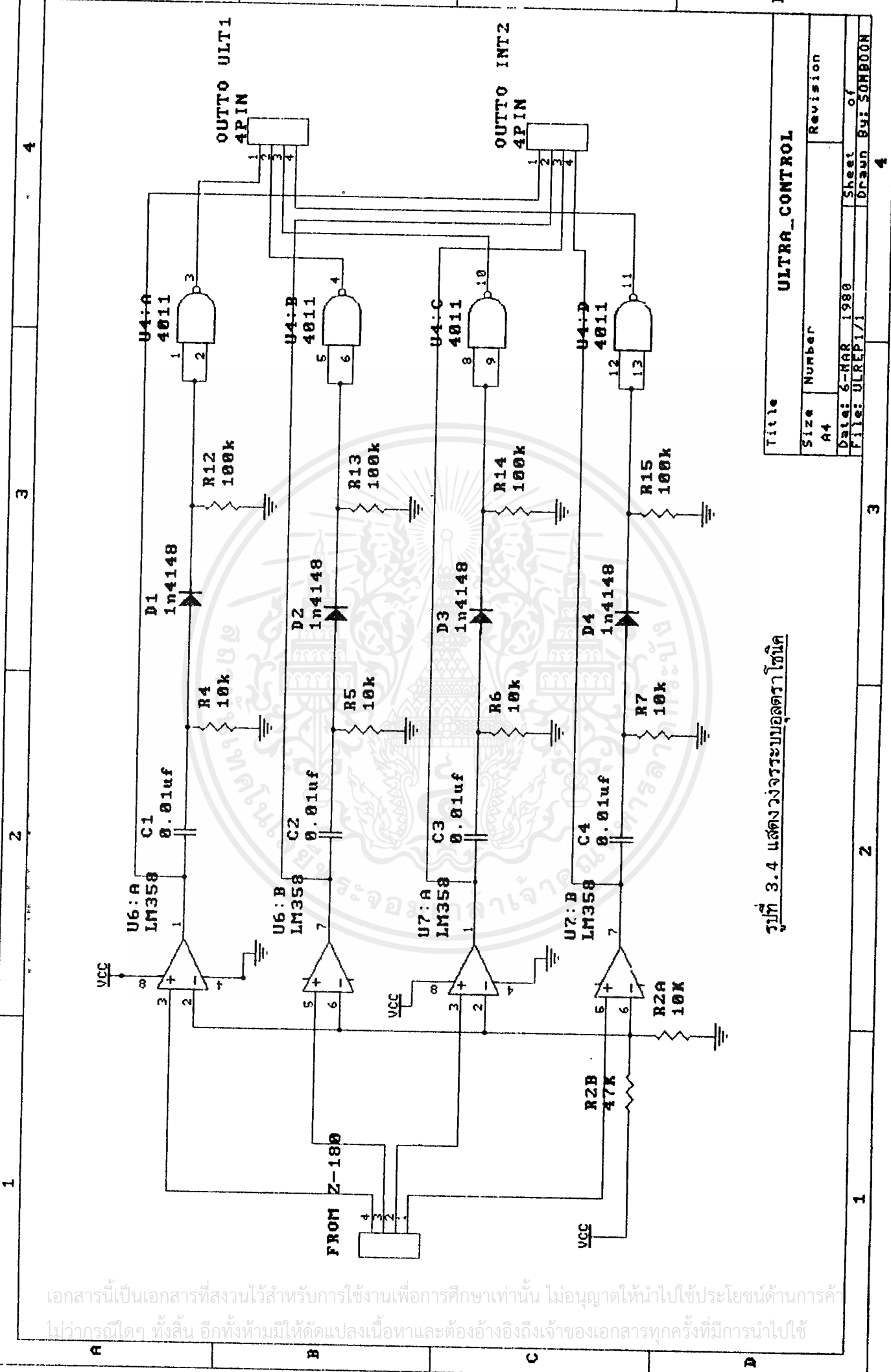
จากตารางผลการทดลองในที่ 1 เราสามารถสรุปได้ว่า อัลตราโซนิก มีความเที่ยงตรงอยู่ในระดับที่น่าพอใจในระดับหนึ่ง แต่ก็ยังมีค่าผิดพลาดอยู่ ซึ่งเราจะสามารถทำได้ข้อมูลที่ใกล้เคียงกับระยะทางที่แท้จริงได้ โดยนำค่าความผิดพลาดทั้งหมดมารวมกัน แล้วหาค่าเฉลี่ย จากนั้นนำค่าเฉลี่ยที่ได้ ไปแก้ความผิดพลาดที่เกิดขึ้น โดยกรรมวิธีทางซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงบล็อกไดอะแกรมของระบบอัลตราโซนิก

Title ULTRASONIC SYSTEM			
Size A4	Number	Revision	
Date: 6-MAR-1988	File: ULREP4/1	Sheet of	
		Drawn By:	



รูปที่ 3.4 แสดงวงจรระบบควบคุม

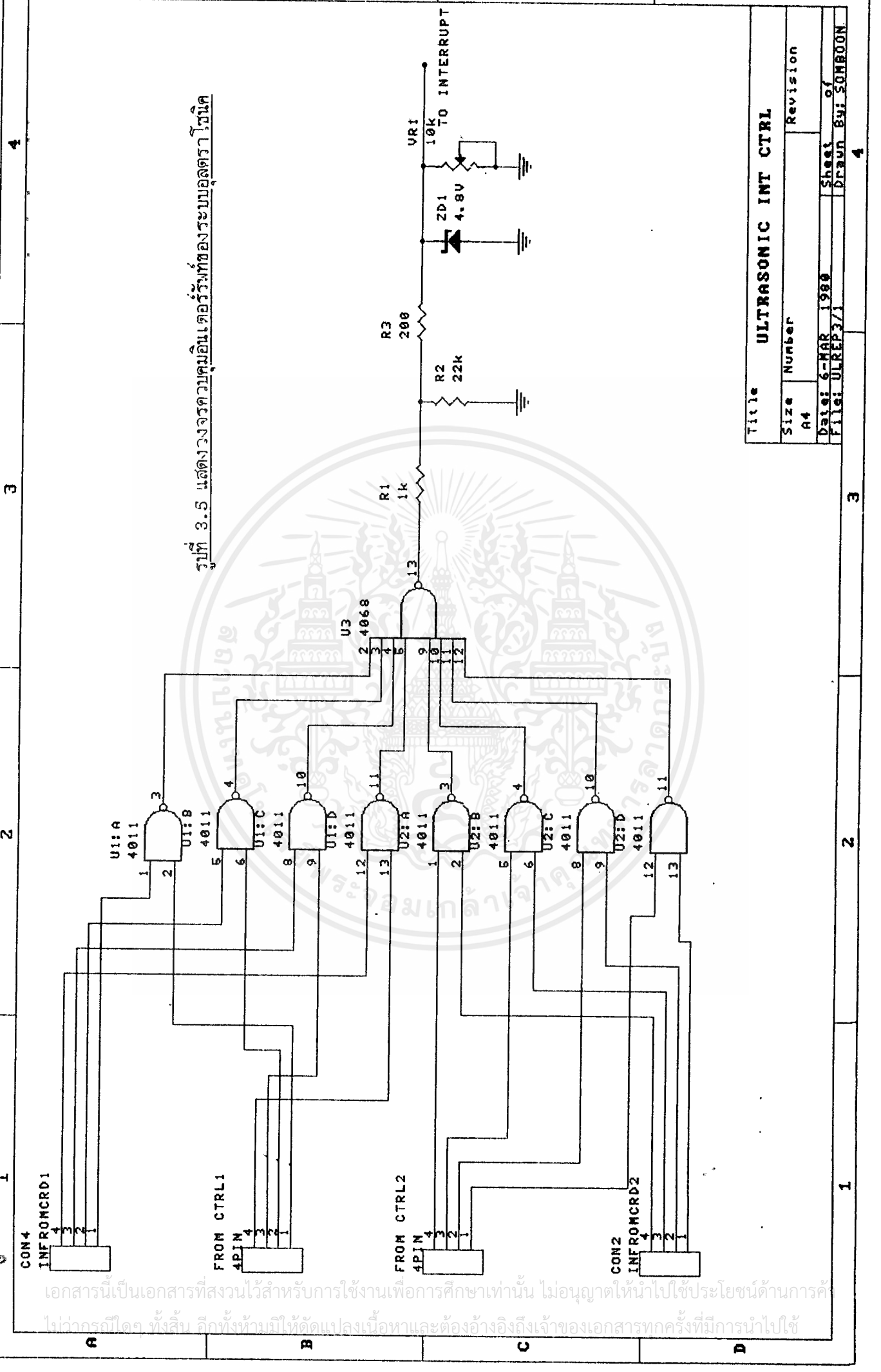
Title		
Size	Number	Revision
A4		
Date:	6-MAR 1980	Sheet of
File:	ULREP1/1	Drawn By: SOMBOON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5



22



รูปที่ 3.5 แสดงวงจรควบคุมอินเตอร์รัพต์ของระบบอัลตราโซนิก

Title		
ULTRASONIC INT CTRL		
Size	Number	Revision
A4		
Date: 6-MAR-1980		Sheet of
File: ULREP3/1		Drawn by: SOMBOON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การควบคุมตำแหน่งล้อหน้า (Steering system)

การควบคุมตำแหน่งล้อหน้า อาศัยการควบคุมตำแหน่ง ของดีซีมอเตอร์ (DC motor) ในส่วนบังคับการเลี้ยว โดยสามารถเลือกความเร็วในการเลี้ยวได้ด้วย ระบบที่ใช้เป็นระบบปิด (close loop system) อาศัยการตรวจจับตำแหน่งของแผ่นจานเอนโคเดอร์แบบแสง (Optical Encoder) และ ส่วนวงจรควบคุมการขับเคลื่อนแบบสวิตช์ (Switch Mode Power Driver) ในส่วนวงจรตรวจจับตำแหน่งเอนโคเดอร์แบบแสงนั้น ได้ออกแบบให้เป็นแบบที่ใช้รหัสเกรย์ (gray code) ขนาด 8 บิต ซึ่งสามารถตรวจจับตำแหน่งที่แตกต่างกันได้ถึง 256 ตำแหน่ง แต่ในการออกแบบ ได้ออกแบบให้ใช้เพียง 180 ตำแหน่งเท่านั้น แต่ละตำแหน่งมีความละเอียด 1 องศา

การออกแบบแผ่นจานเอนโคเดอร์ได้ใช้โปรแกรมสำเร็จรูปออโตแคด (AutoCad) ซึ่งเป็นซอฟต์แวร์ช่วยออกแบบที่ได้รับความนิยมในระดับมืออาชีพอย่างกว้างขวาง มาช่วยในการออกแบบ เนื่องจากซอฟต์แวร์ AutoCad นั้นให้ความเที่ยงตรงสูง โดยเฉพาะเมื่อพิมพ์โดยใช้เครื่องพิมพ์ชนิดเลเซอร์ (Laser Printer)

การออกแบบ ได้ออกแบบขนาดของจานตามขนาดของอุปกรณ์ตรวจจับ (Sensor) ซึ่งแต่ละตัวมีความกว้างด้านข้าง 4.60 มิลลิเมตร ทั้งหมดมี 8 ตัว โดยออกแบบให้แผ่นจานเอนโคเดอร์มี

เส้นผ่าศูนย์กลาง 70 มิลลิเมตร

ระยะห่างระหว่างวง = ความกว้างของตัวตรวจจับ 5.00 มิลลิเมตร

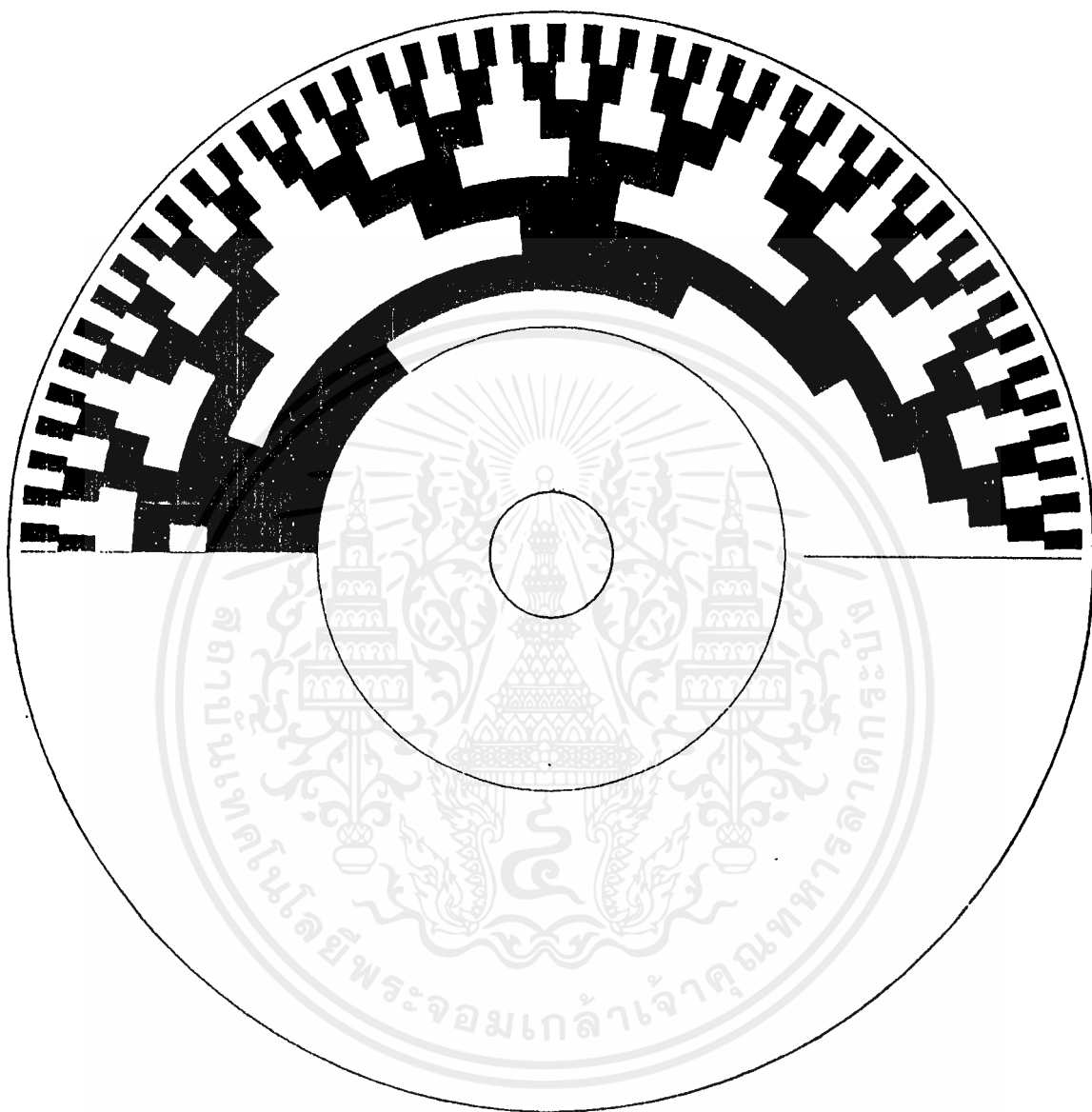
ช่องว่างแต่ละบิตนั้นกำหนดให้เท่ากับเส้นผ่าศูนย์กลาง ของรูรับ-ส่งแสงของตัวตรวจจับ คือ 1.2 มิลลิเมตร เมื่อแบ่งช่องต่างๆเสร็จแล้วก็ระบายทึบตรงช่องที่เป็น "1" ของรหัสเกรย์ ขนาด 8 บิต

Decimal	Hexadecimal	Binary	Gray
0	00h	00000000	00000000
1	01h	00000001	00000001
2	02h	00000010	00000011

Decimal	Hexadecimal	Binary	Gray
3	03h	00000011	00000010
4	04h	00000100	00000110
5	05h	00000101	00000111
6	06h	00000110	00000101
7	07h	00000111	00000100
8	08h	00001000	00001100
9	09h	00001001	00001101
10	0Ah	00001010	00001111
11	0Bh	00001011	00001110
12	0Ch	00001100	00001010
13	0Dh	00001101	00001011
14	0Eh	00001110	00001001
15	0Fh	00001111	00001000
.	.	.	.
178	0B2h	10110010	11101011
179	0B3h	10110011	11101010

จะเห็นว่าทั้งหมดมี 180 ตำแหน่ง แต่เวลาใช้งานจริงจะใช้เพียง 120 ตำแหน่ง คือการ
เลี้ยวของล้อจากจุดกลาง (เดินตรง) ไปด้านซ้ายไม่เกิน 60 องศา และเลี้ยวไปด้านขวาไม่เกิน
60 องศา เช่นกัน ถ้ามุมเลี้ยวมากกว่า 60 องศาจะทำให้เกิดการเลื่อน (Slipage) ขึ้นการ
เลี้ยวจะผิดพลาดทันที

รูปร่างของ แผ่นจาน Encoder แสดงดังรูป 4.1



รูปที่ 4.1 แผ่นจาน เอ็นโค้ดเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

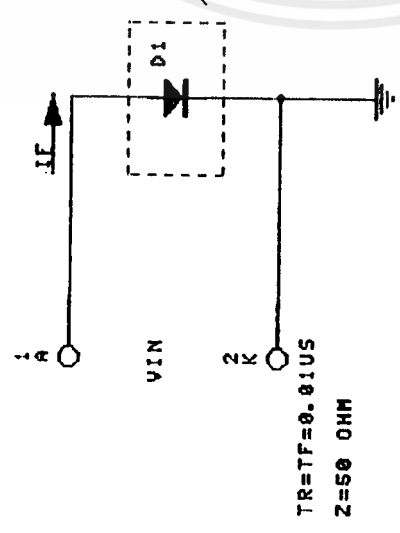
สาเหตุที่ออกแบบแผ่นจาน Encoder ให้เป็นรหัสแบบเกรย์ ก็เพราะว่าการเปลี่ยนแปลงตำแหน่ง ของจาน Encoder นั้น เปลี่ยนครั้งละ 1 Bit เสมอ คือ การเพิ่มขึ้นหรือลดลงของค่าประจำตำแหน่งนั้นๆ จะทำให้ Bit ของตำแหน่งนั้นเปลี่ยนแปลงเพียง 1 ตำแหน่ง เช่น การเปลี่ยนตำแหน่งจาก 3 ไปเป็นตำแหน่ง 4 ถ้าเป็นไบนารีไคต์ ก็จะเปลี่ยนตำแหน่งจาก 00000011 ไปเป็น 00000100 จะเห็นได้ว่า มีการเปลี่ยนพร้อมๆกันถึง 3 บิต ลักษณะเช่นนี้อาจจะทำให้การอ่านรหัสของตัวอ่านเกิดผิดพลาดได้มาก ถ้าเป็นเกรย์ไคต์ จะเปลี่ยนจาก 00000010 ไปเป็น 0000110 ซึ่งจะเห็นได้ว่าการเปลี่ยนจากบิตเดิมเพียงบิตเดียว การอ่านรหัส ของตัวตรวจจับ จึงมีโอกาสผิดพลาดน้อยมากเมื่อเทียบกับแผ่นจานที่สร้างด้วยรหัสไบนารี บล็อกไดอะแกรมของส่วนควบคุมล้อหน้าแสดงดังรูปที่ 4.3

จากบล็อกไดอะแกรม ชุดตรวจจับรหัส จะคอยอ่านตำแหน่งที่เปลี่ยนไป ของแผ่นจานรหัสซึ่งถูกยึดติดไว้กับเพลลาของ มอเตอร์ บอร์ดควบคุม Z-80180 จะอ่านรหัสจากชุดตรวจจับ เพื่อนำไปประมวลผล ในการตัดสินใจ ควบคุมการหมุนของมอเตอร์ ให้หมุนไปตามตำแหน่งที่ต้องการ นั่นคือ การตัดสินใจว่า จะเลี้ยวซ้าย หรือเลี้ยวขวา และเลี้ยวไปเป็นมุมกี่องศา การบังคับเลี้ยวสามารถระบุการเลี้ยว เป็นมุมที่มีความละเอียด ได้ถึง 1 องศา ชุด Power drive เป็นส่วนจ่ายแรงดันไฟตรงให้กับ มอเตอร์ Power drive นี้ ถูกควบคุม ด้วย ชุดควบคุม Control board

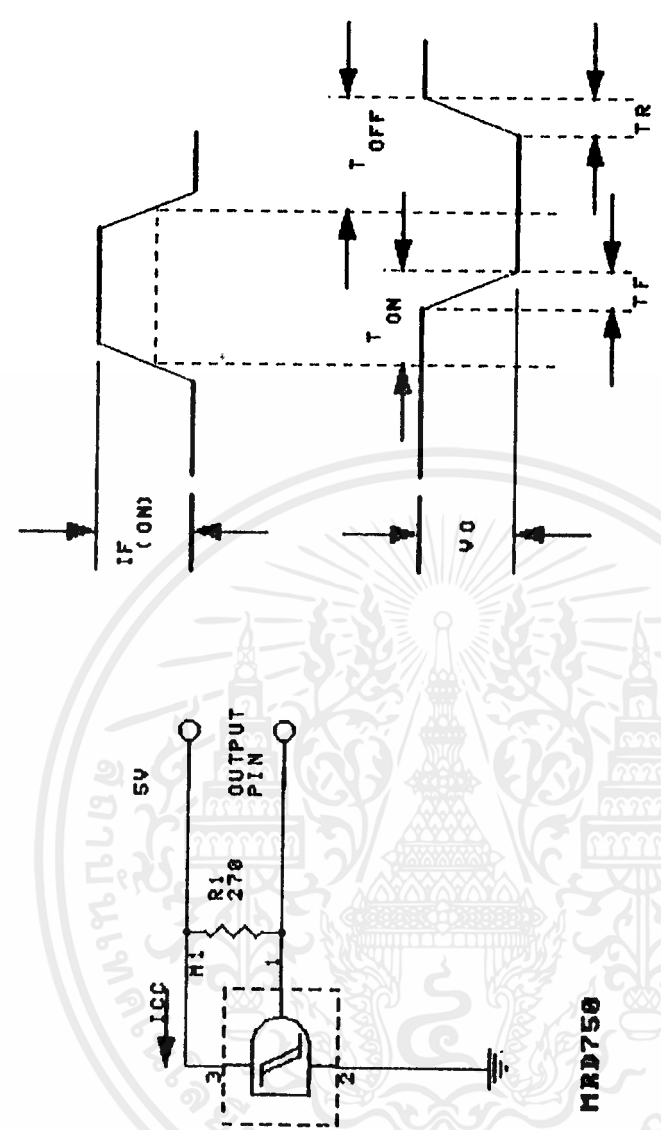
ข้อดีของการติดตั้งแผ่นเอ็นโคเดอร์ เข้ากับเพลลาของมอเตอร์โดยตรง ก็คือ สามารถลดข้อผิดพลาดของการเชื่อมต่อ (Coupling) ระหว่างอุปกรณ์ต่าง ๆ ลงได้มาก เพราะไม่ต้องนำรหัสที่อ่านได้มาผ่านวงจรแปลงสัญญาณจากสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล หรือไม่ต้องมีการแปลงจากความถี่เป็นแรงดัน แต่สามารถที่จะอ่านค่าตำแหน่งมุมต่างๆเป็นสัญญาณทางดิจิทัลในรูปของรหัสเกรย์แล้วใช้ซอฟต์แวร์แปลงเป็นรหัสไบนารี

4.1 optical Encoder

ใช้ Infrared LED เบอร์ MLED71 ของ MOTOROLA ซึ่งได้ออกแบบมาใช้กับงานทางด้าน Infrared Sensing โดยทั่วไป โดยใช้กับตัวตรวจจับแสงอินฟราเรด Photo Detector ตระกูล MRD700, MLED71 สามารถให้แสงอินฟราเรดที่มีความเข้มของแสงสูง มีความยาวคลื่นใช้งาน 940 นาโนเมตร เมื่อจ่ายกระแสให้ 60 mA มีย่านอุณหภูมิใช้งาน ที่ -40 ถึง +100 องศาเซลเซียส ตัวรับแสงอินฟราเรด เลือกใช้ เบอร์ MRD750 ซึ่งได้รวมเอาคุณสมบัติของวงจรชmittริกเกอร์ (Schmit-Trigger) ไว้ในตัวเอง สามารถกำจัดสัญญาณรบกวนต่าง ๆ ให้



รูปที่ 4.2 แสดงการทำงานของ MLED71 และ MLED750



Title	
Size	Number
A4	1993
Page	Sheet
1	1 of 1
Drawn By	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอาท์พุทที่เป็นรูปสี่เหลี่ยมที่สมบูรณ์ (Pure Square Wave) เอาท์พุทเป็นแบบคอลเล็คเตอร์เปิด (Open Collector) สามารถนำไปประยุกต์ใช้งานได้อย่างกว้างขวาง แรงดันเอาท์พุทอยู่ในย่าน 0-16 โวลต์ ใช้กับแหล่งจ่ายในย่าน 0-16 โวลต์ จ่ายกระแสเอาท์พุทได้ 50 mA มีย่านอุณหภูมิใช้งาน ที่ -40 ถึง +85 องศาเซลเซียส เมื่อเอาท์พุทเป็น ลอจิก "High" ที่แรงดันไฟเลี้ยง 5 โวลต์ สามารถจ่ายกระแสเอาท์พุท สูงสุด 5 mA

การทำงานของตัวรับส่งอินฟราเรด (Infrared) แสดงดังรูปที่ 4.2 เมื่อแสงอินฟราเรดจาก MLED71 ตกกระทบบนตัวรับแสงของ MRD750 MRD750 จะให้เอาท์พุทเป็นลอจิก "LOW" แต่ถ้ามีวัสดุทึบแสงมาบังแสงอินฟราเรดไว้ ไม่ให้แสงจาก MLED71 ส่องไปกระทบบน MRD750 ก็จะทำให้ เอาท์พุทเป็นลอจิก "HIGH"

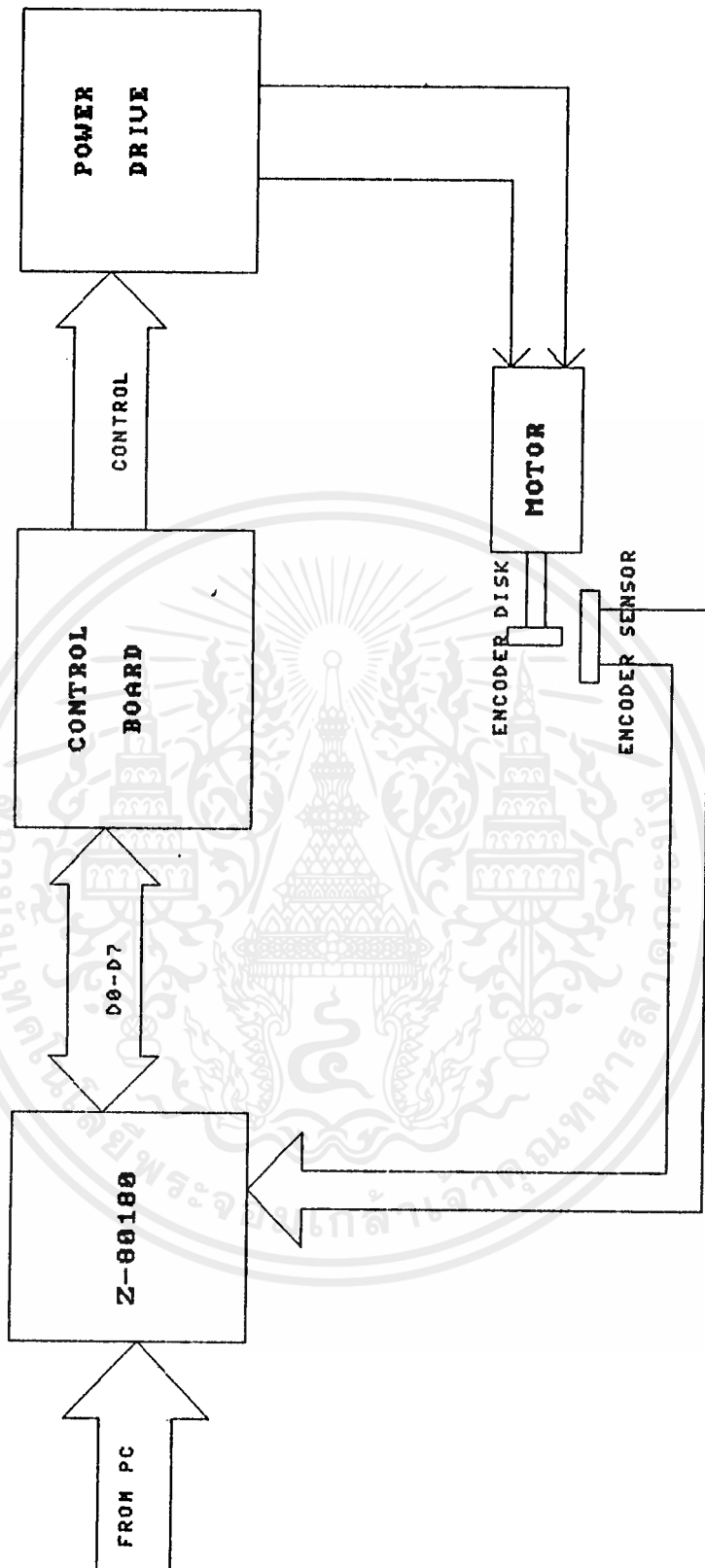
จากรูปที่ 4.2 เมื่อ MLED71 ได้รับแรงดันไบแอสตรง และมีกระแสไหลผ่านตัวมัน 50 mA MLED71 จะเปล่งแสง Infrared ออกมา เมื่อแสงไปตกกระทบบนตัวรับ MRD750 ซึ่งจ่ายไฟเลี้ยงไว้เรียบร้อยแล้ว MRD750 จะให้แรงดันเอาท์พุทที่ ขา 1 เป็น 0 โวลต์ ถ้าใช้แผ่นวัสดุทึบแสง ไปกั้น ไม่ให้แสง Infrared จากตัวส่ง MLED71 ผ่านมากระทบบนตัวรับ ตัวรับ MRD750 ก็จะทำให้แรงดันเอาท์พุท เป็น 5 โวลต์ ตัวรับ - ส่ง มีทั้งหมด 8 ชุด โดยวางเรียงให้ชิดกัน เพื่อให้ได้เอาท์พุท 8 บิต แบบขนาน ในขั้นนี้ จะวาง แผ่นจานรหัส ก็ ะหว่าง ตัวรับและตัวส่ง เมื่อส่วนที่ระบายความร้อนบนแผ่นจาน เคลื่อนที่เข้ามาใกล้ตัวรับ-ส่ง ก็จะได้ Logic HIGH ถ้าเป็นส่วนที่ไม่ระบายความร้อน ก็จะได้ Logic LOW เอาท์พุท 8 บิตที่ได้ก็จะขึ้นอยู่กับรหัสบนแผ่นจาน ซึ่งบอร์ด CPU Z-80180 สามารถอ่านโดยใช้คำสั่งอินพุท (Input) แบบขนานเข้าไปได้ทันที จึงทำให้การอ่านข้อมูลจากชุดตรวจจึงมีความเร็วสูงมาก

วงจรเอ็นโค้ดเดอร์เซ็นเซอร์ แสดงดังรูปที่ 4.4

4.2 ชุดควบคุม มอเตอร์

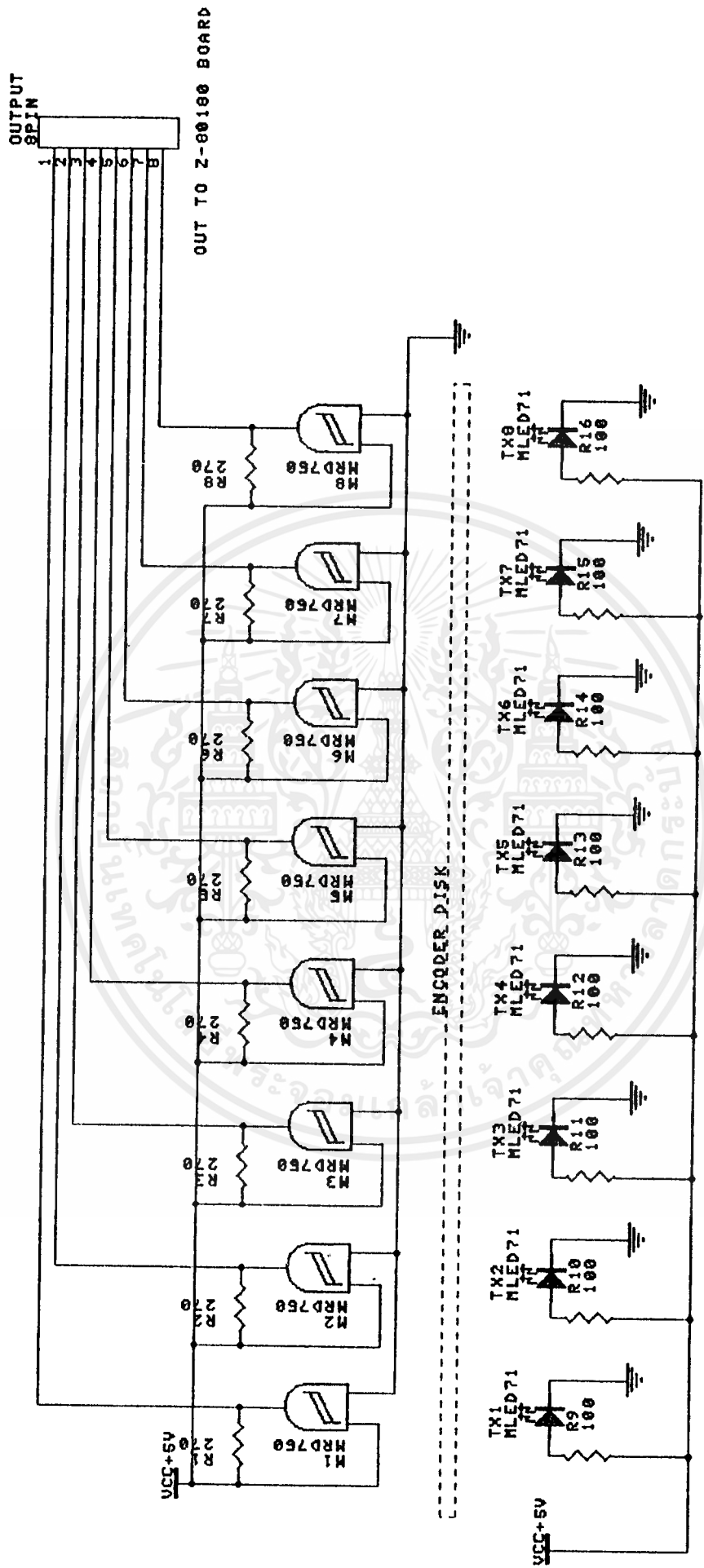
ใช้ชุดควบคุม เหมือนกับวงจรควบคุมการขับเคลื่อนล้อหลัง เป็นลักษณะการควบคุมแบบดิจิทัล (Digital Control of DC Motor) โดยใช้ ชิปไอซี เบอร์ L292 เป็นชุดควบคุมความเร็ว โดยใช้รหัสไบแนรี 5 บิต จาก CPU ซึ่งสามารถเปลี่ยนแปลงระดับความเร็วได้ และใช้บิตสถานะ (Sign Bit) ในการควบคุมทิศทางการหมุนของมอเตอร์

โดยปกติ ชิปไอซี เบอร์ L292 สามารถนำไปเป็นภาคขับกำลังของ DC Motor ได้โดยตรงอยู่แล้ว ซึ่งตามปกติจะใช้กับมอเตอร์ที่กินกระแสไม่เกิน 2 A แรงดันไฟกักไม่เกิน 36 Vdc แต่ในโครงการนี้ ใช้ DC Motor ที่กินกระแสสูงถึง 7.5 A ดังนั้นจึงได้ออกแบบภาคขับกำลังให้กับ DC Motor โดยใช้เพาเวอร์ทรานซิสเตอร์ (Power Transistor) มาต่อเป็นส่วนขับวงจรไฮบริดจ์ (High-Bridge) ร่วมกับชิปไอซีเบอร์ L292 อีกทั้งการควบคุมความเร็วโดยตรง วงจรแสดงดังรูปที่ 4.5



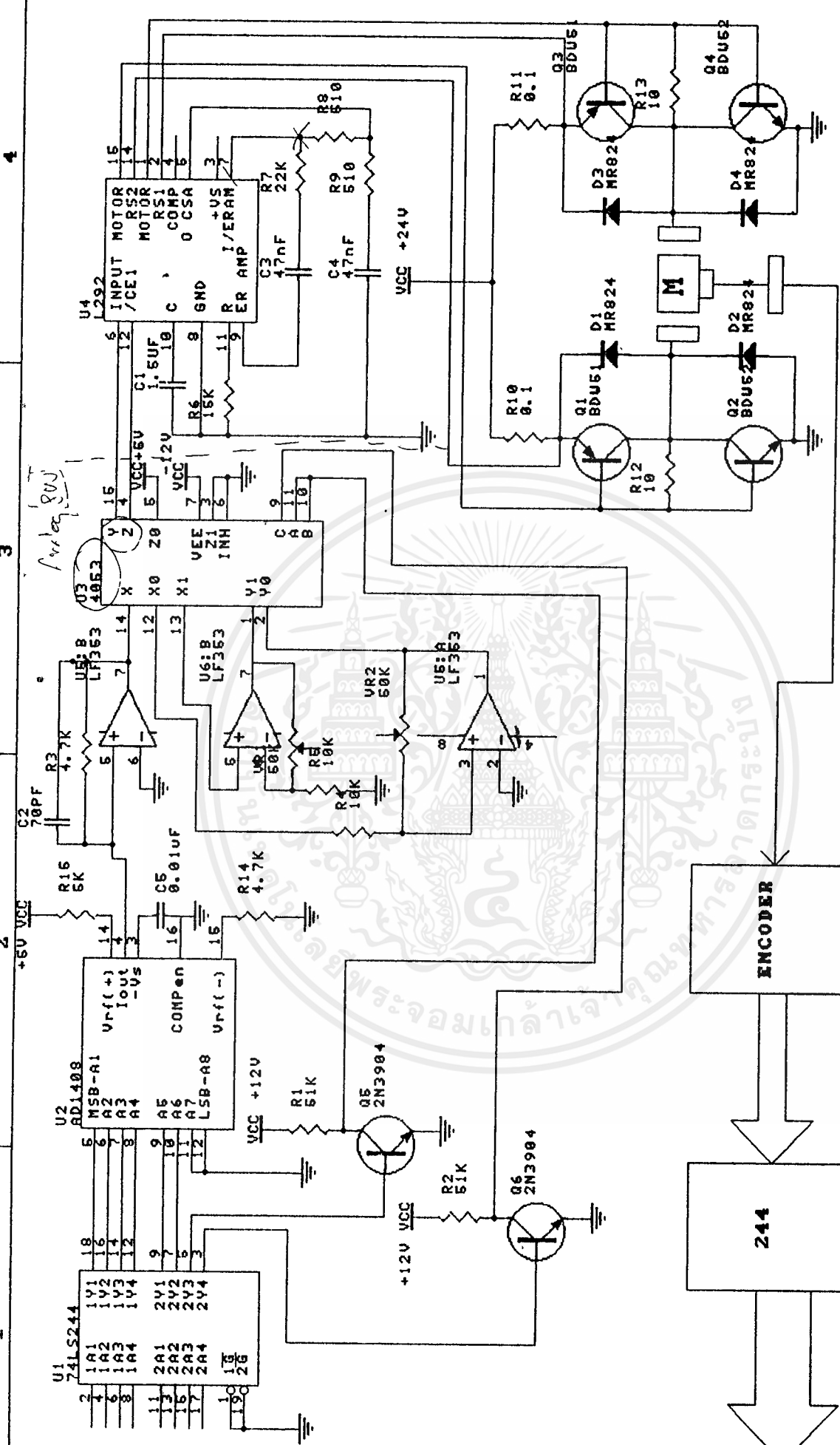
รูปที่ 4.3 แสดงบล็อกไดอะแกรมของระบบควบคุมล้อหน้า

Title F WHEEL CTRL BLOCK		
Size A4	Number	Revision
Date: 6-MAR-1988	File: F-WHEEL2/1	Sheet of
Drawn By:		4



รูปที่ 4.4 แสดงบล็อกวงจรเอนโคเดอร์ 8 บิต

TX & RX OF ENCODER		
Size	Number	Revision
A4		
Date: 1-NOV-1993		
Drawn By: SOMBON		



Title		PWMHLL CONTROL&DRIVE	
Size	Number	Revision	
A4		1	
Date: 6-NOV-1989	Sheet of	Drawn By:	
File: P04PHEM/1			

รูปที่ 4.5 แสดงวงจรควบคุมมอเตอร์

บทที่ 5 ระบบขับเคลื่อนล้อหลัง

5.1 ความเป็นมา

ระบบการขับเคลื่อนร륜ก่อนได้ใช้ระบบเฟสล็อกลูป (Phase Lock Loop) ซึ่งเป็นการสร้างความถี่อ้างอิงตามความเร็วที่ต้องการ และตรวจจับความถี่ที่ป้อนกลับเข้ามา ถ้าความถี่ที่ตรวจจับกับความถี่อ้างอิง มีเฟสตรงกัน เอาท์พุท (Output) ที่ไปควบคุม DC Motor ก็จะคงที่ ๆ ค่าหนึ่งเช่นที่ความเร็วสูงสุด output ที่ออกก็จะออก 5 Volt DC เป็นต้น ระบบนี้เราจะเรียกว่า Close Loop System ซึ่งข้อดีของระบบนี้ก็คือ สามารถควบคุมความเร็วหรือแรงบิดให้คงที่ได้ไม่ว่าสภาวะของภาระ (Load) จะมีมากหรือน้อย ในการพัฒนาจึงได้นำข้อดีในส่วนนี้มาใช้ แต่ระบบนี้ยังมีจุดบกพร่อง คือ ช่วงเวลาหน่วง (Delay time) ในการเข้าสู่สภาวะล็อก (Lock Loop) ค่อนข้างช้า ความถี่เนื้องานในการออกตัวยังไม่มี จุดบกพร่องต่างๆเหล่านี้ได้ถูกปรับปรุงแก้ไขซึ่งจะได้นำเสนอในหัวข้อต่อไป

จุดบกพร่องของรุ่นก่อนดังนี้

- ความเร็วในการเดินหน้าและถอยหลัง ไม่เท่ากัน
- การออกตัวไม่เนิ่นนวลมีการกระตุกเป็นช่วงๆ
- ระบบการป้อนกลับยังไม่สมบูรณ์

5.2 การพัฒนาระบบ

จากจุดบกพร่องที่ได้กล่าวมาในได้นำมาทดลองและวิจัยหาจุดที่ทำให้ระบบเป็นเช่นนั้นเราจึงพบว่าในการออกแบบวงจรควบคุมที่ใช้ IC MC4044 เป็นตัวตรวจจับเฟส ยังไม่สอดคล้องกับระบบที่ได้กำหนดไว้ในเอกสารข้อมูลจำเพาะของอุปกรณ์ (DATA SHEET) จึงส่งผลให้ระบบมีข้อผิดพลาดและขาดความสมบูรณ์ทางด้านการควบคุมไป เป็นผลทำให้ระบบดังต่อไปนี้เกิดขึ้นเพื่อแก้ไขจุดบกพร่องต่าง ๆ ที่ได้กล่าวมา

จากรูปที่ 5.2 แสดงบล็อกไดอะแกรมของระบบควบคุมล้อหลัง เป็นการแสดงระบบควบคุมล้อหลังทั้งหมดซึ่งได้เพิ่มเติมจากของเดิมเพื่อแก้จุดบกพร่องต่างๆ จะเห็นได้ว่า เราได้เพิ่ม CPU Z80180c ในการควบคุม DC motor โดยตรงซึ่งการควบคุมในลักษณะนี้จะทำให้ระบบใหญ่ของ Robot ตัวนี้ มีความสามารถทำงานในลักษณะเวลาจริง (Real Time) มากที่สุด ในแต่ละบล็อกไดอะแกรม ที่ได้เขียนไว้นี้ มีการเชื่อมต่อ (Interface) ซึ่งกันและกันได้ตลอดเวลา ดังจะเห็นในรูปที่ 5.3

5.3 อธิบายรายละเอียดการพัฒนา

จากรูปที่ 5.2 บล็อกไดอะแกรมระบบควบคุมล้อหลัง การติดต่อสื่อสารระหว่างคอมพิวเตอร์หลัก (CPU 80286) กับ CPU Z80180c จะเป็นการสื่อสารแบบมี Interrupt เพื่อให้ระบบสามารถทำงานได้อย่างเป็นอิสระ แยกการทำงานออกจากกัน เช่น Main Computer ต้องการสั่ง ให้ล้อหลังหมุนด้วยความเร็วสูงสุดเป็นระยะทาง 10 เมตร คอมพิวเตอร์หลัก ก็ต้องส่งสัญญาณขอ Interrupt ไปให้ Z80180c ก่อน และข้อมูล (Data) การควบคุมก็จะส่งตามมาโดยผ่านบัสข้อมูล (Data Bus) ในทางกลับกันเมื่อถึงระยะ 10 เมตรแล้ว Z80180c ก็ต้องบอกคอมพิวเตอร์หลักว่า ต่อมาถึงตำแหน่งที่สั่งแล้ว Z80180c ก็ต้องส่งสัญญาณ Interrupt ไปให้คอมพิวเตอร์หลัก ซึ่งลักษณะการทำงานเช่นนี้จะเป็นลักษณะ Real Time มากที่สุด

5.4 การตรวจจับความเร็วและระยะทาง

ในการพัฒนาระบบเราจะใช้ชิป Z80CTC เป็นตัวบัสสัญญาณการป้อนกลับจากการหมุนของล้อหลังซึ่งจะแปรผันโดยตรงกับความเร็ว โดยที่จะแบ่งการตรวจจับเป็น 2 ระบบ คือ แบบนับสัญญาณนาฬิกาอนุกรม กับแบบตรวจจับบิต (BIT) จากจาน Encoder Disk ที่ได้พัฒนาขึ้น ซึ่งแต่ละวิธีก็มีข้อดีของแต่ละวิธี แบบตรวจจับ BIT จะมีความเร็วในการอ่านค่าได้สูงกว่า

5.5 ทฤษฎีการหาระยะทาง

ข้อมูลทางกายภาพ

$$1 \text{ รอบของล้อหลัง} = 79 \text{ Cm} = 256 \text{ Clock}$$

จากข้อมูลทางกายภาพ ถ้าต้องการให้ Robot วิ่งไปเก็บระยะทาง 10 เมตร จะคำนวณได้ดังนี้

$$\begin{aligned} \text{Clock} &= (1000 \times 256) / 79 \\ &= 3240.6 \end{aligned}$$

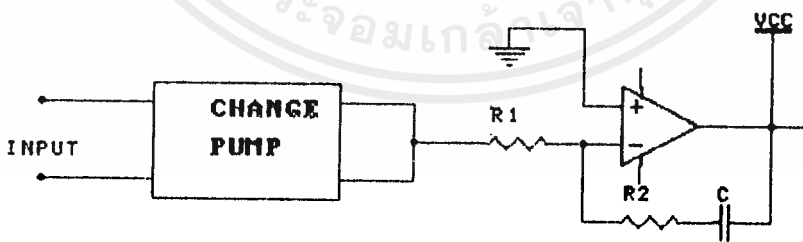
$$\text{จำนวนรอบ} = 12.65 \text{ รอบ}$$

เพราะฉะนั้นจะต้องให้ Z80CTC นับ Clock ไปเก็บจำนวน 3240.6 ANS

5.6 ทฤษฎีการตรวจจับความเร็ว

เราจะใช้ RTC (REAL TIME CLOCK) เป็นฐานเวลาการตรวจจับมีหน่วยเป็น m/s โดยมีหลักคิดอยู่ว่า ใน 1 วินาที Z80CTC นับ Clock ได้กี่ลูก และนำไปคำนวณ เช่น

$$79 \text{ Cm} = 256 \text{ clock เป็นต้น}$$



รูปที่ 5.1 แสดงการแก้การกระตุกและแรงบิดไม่คงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากวงจรเดิมอุปกรณ์บางตัวไม่อยู่ในเกณฑ์ของ Data Sheet ซึ่งส่งผลให้ค่า Damping t ในการเข้าระบบเร็วเกินไปซึ่งผลตอบสนองของ DC motor ตอบสนองไม่ทัน

สูตรการวิเคราะห์

$$Q_o(s)/Q_i(s) = K_o \cdot K_r \cdot K_v / (S + K_o \cdot K_r \cdot K_v / N) \quad (1)$$

$$K_r = 1 + T_1 \cdot S / T_2 \cdot S \quad (2)$$

$$T_1 = R_2 \cdot C$$

$$T_2 = R_1 \cdot C$$

Then..

$$Q_o(s)/Q_i(s) = N(1+T_1 \cdot S) / ((S^2 \cdot N \cdot T_2 / K_o \cdot K_v) + T_1 \cdot S + 1)$$

$$\gamma_n = \text{loop bandwidth or natural frequency}$$

$$\zeta = \text{damping factor}$$

$$\gamma_n = \sqrt{(K_o \cdot K_v / N \cdot T_2)}$$

$$\zeta = (\sqrt{K_o \cdot K_v / N \cdot T_2}) \cdot T_1 / 2$$

Then..

$$Q_o(s)/Q_i(s) = N(1+T_1 \cdot S) / (S^2 / \gamma_n^2 + 2 \cdot \zeta \cdot S / \gamma_n + 1)$$

$$T_2 = K_o \cdot K_v / N \cdot \gamma_n^2$$

$$T_1 = 2 / \gamma_n$$

$$R_1 = K_o \cdot K_v / N \cdot \gamma_n^2 \cdot C$$

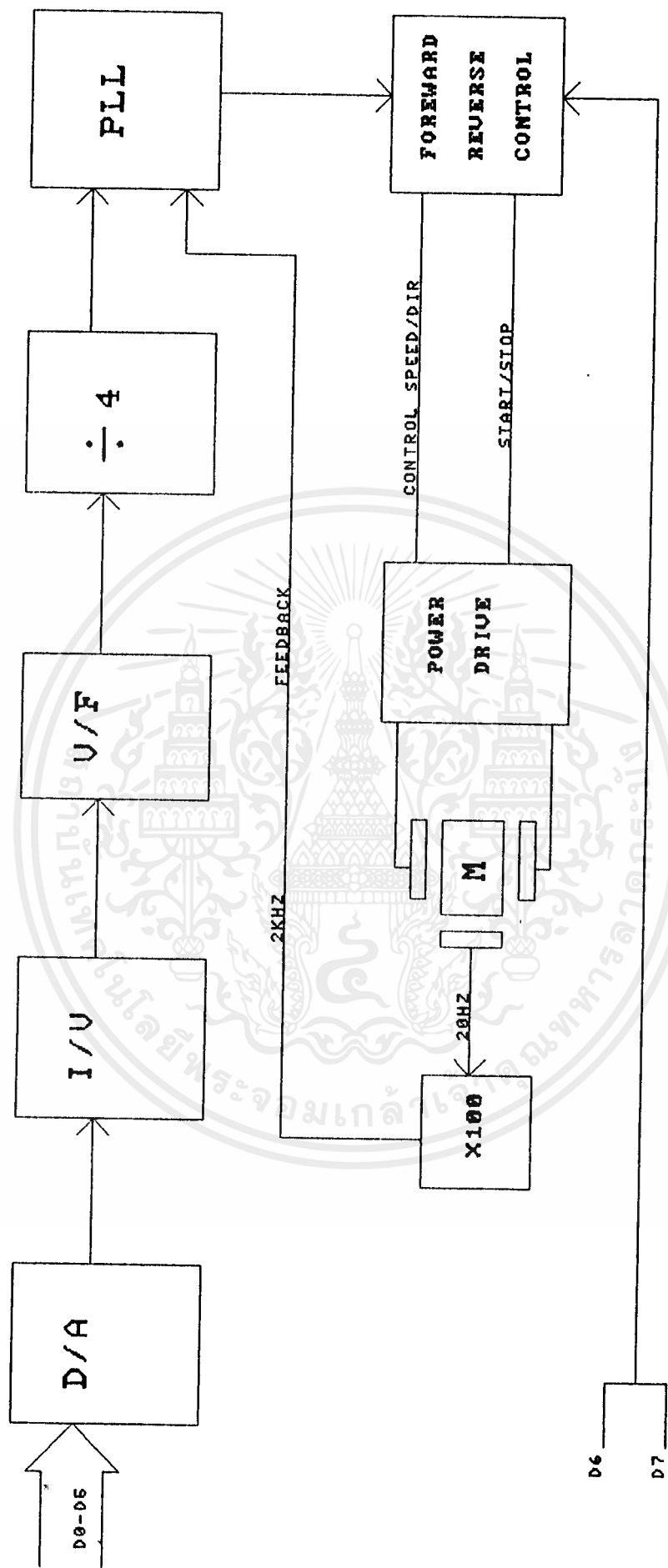
$$R_2 = 2 / \gamma_n \cdot C$$

$$R_2 > 50 \text{ ohm}$$

$$R_2 / R_1 \leq 10$$

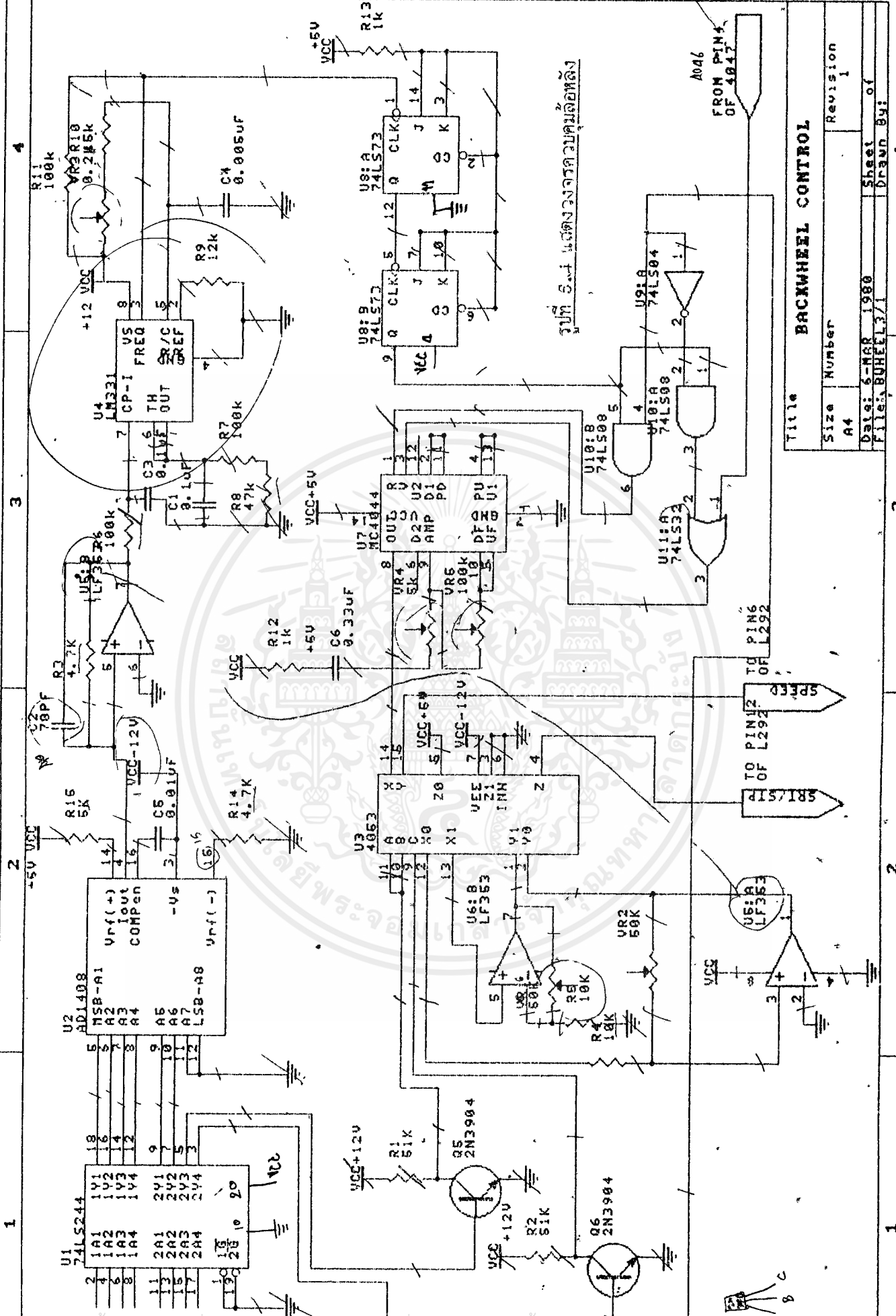
$$1k < R_1 < 5k \text{ ohm}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



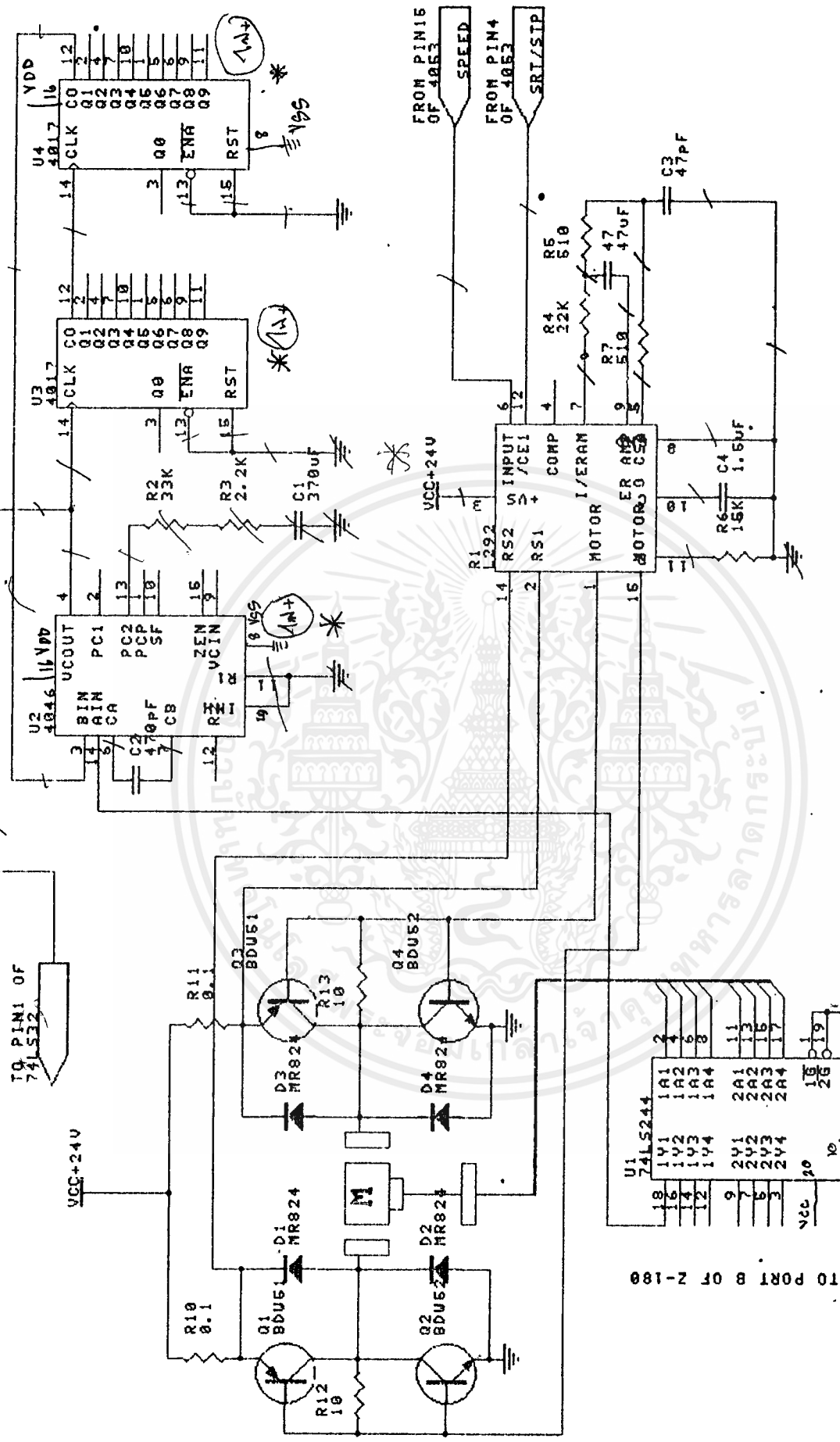
รูปที่ 5.3 แสดงบล็อกโตะแอมของระบบควบคุมมอเตอร์

Title			ADC & DAC MOTOR CTRL	
Size	Number	Revision		
A4				
Date:	6-MAR-1988		Sheet of	
File:	8-UHELL2/1		Drawn By:	



รูปที่ 5.1 แสดงวงจรควบคุมล้อหลัง

Title BACKWHEEL CONTROL		
Size	Number	Revision
A4		1
Date:	6-MAR-1980	Sheet of
File:	BUWHEEL3/1	Drawn By:



BACKWELL DRIVER		
Title	Size	Number
	A4	
Revision		
1		
Date: 6-MAR-1980	Sheet of	
File: BUHELL4/1	Drawn By:	

รูปที่ 5.5 แสดงวงจรภาคขับกำลังมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังขอให้อ่านเงื่อนไขและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การติดต่อสื่อสารข้อมูลของระบบ

ระบบหุ่นยนต์เคลื่อนที่ได้เอง (Mobile Robot System) ที่ได้ออกแบบในระยະที่ 3 นี้ เป็นระบบที่ออกแบบให้มีการทำงานในโหมด Real Time มากที่สุด ซึ่งจากบล็อกไดอะแกรมรูปที่ 2.2 ในบทที่ 2 จะเห็นว่าการทำงานของ CPU แต่ละตัวสามารถทำงานไปพร้อมกันในลักษณะขนาน (Parallel Processing) และเราจะสังเกตได้ว่าการทำงานในระบบ Real Time นี้มีหัวใจสำคัญอยู่ที่การสื่อสารข้อมูล เมื่อ CPU ลูก (Slave CPU) ตัวใดต้องการรับข้อมูลหรือต้องการส่งข้อมูลให้ CPU แม่ (Master CPU) หรือ Master CPU ต้องการติดต่อกับ Slave CPU ตัวใด ก็ส่งสัญญาณ Interrupt นอกให้ CPU ที่ต้องการติดต่อด้วยรับรู้ เพื่อดำเนินการรับหรือส่งข้อมูลต่อไป จากหลักการนี้จึงทำให้ Mobile Robot นี้ทำงานเข้าใกล้ระบบ Real Time มากที่สุด

จากที่ได้กล่าวแล้วว่า การสื่อสารข้อมูลเป็นหัวใจสำคัญของระบบ Real Time นี้ ซึ่งแน่นอนที่สุด นอกจากอุปกรณ์อื่น ๆ เช่น ความเร็วของ CPU, การตอบสนองของฮาร์ดแวร์เบื้องต้น ฯลฯ เหล่านี้จะเป็นส่วนที่ทำให้ระบบทำงานใกล้เคียงกับ Real Time แล้ว ตัวกลางในการติดต่อของระบบก็เป็นส่วนที่ขาดเสียมิได้ ซึ่งเราจะขออธิบายลงไปในส่วนที่เป็นฮาร์ดแวร์ของระบบ

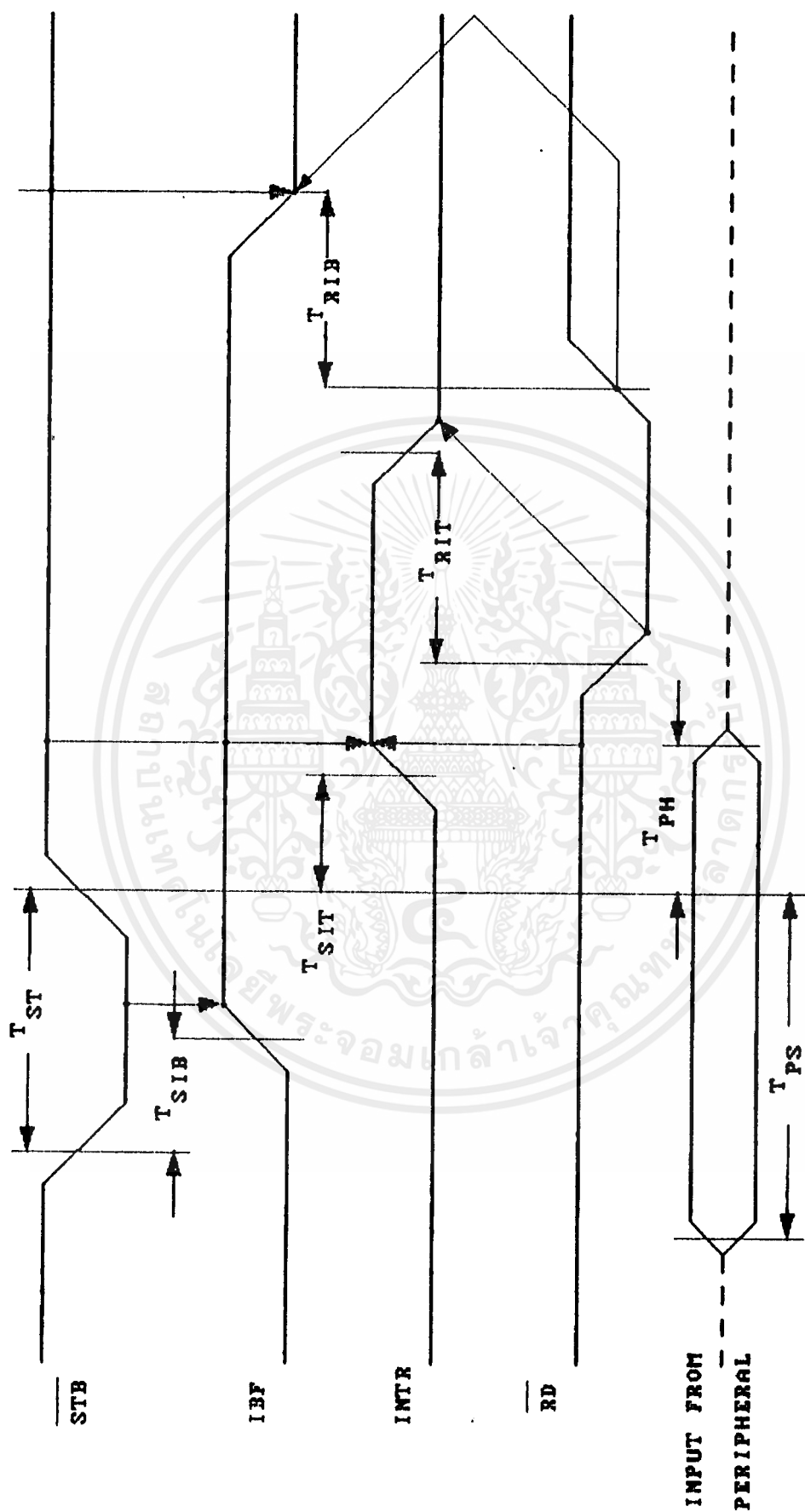
เราใช้ฮาร์ดแวร์ในโหมดแฮนด์เชก (Hand Shake) เป็นตัวกลางการสื่อสารข้อมูลในระบบ โดยเขียนออกมาเป็นผังวงจร (Schematic Diagram) ดังนี้

6.1 การใช้งาน 8255 ใน Mode 1 (Hand Shake)

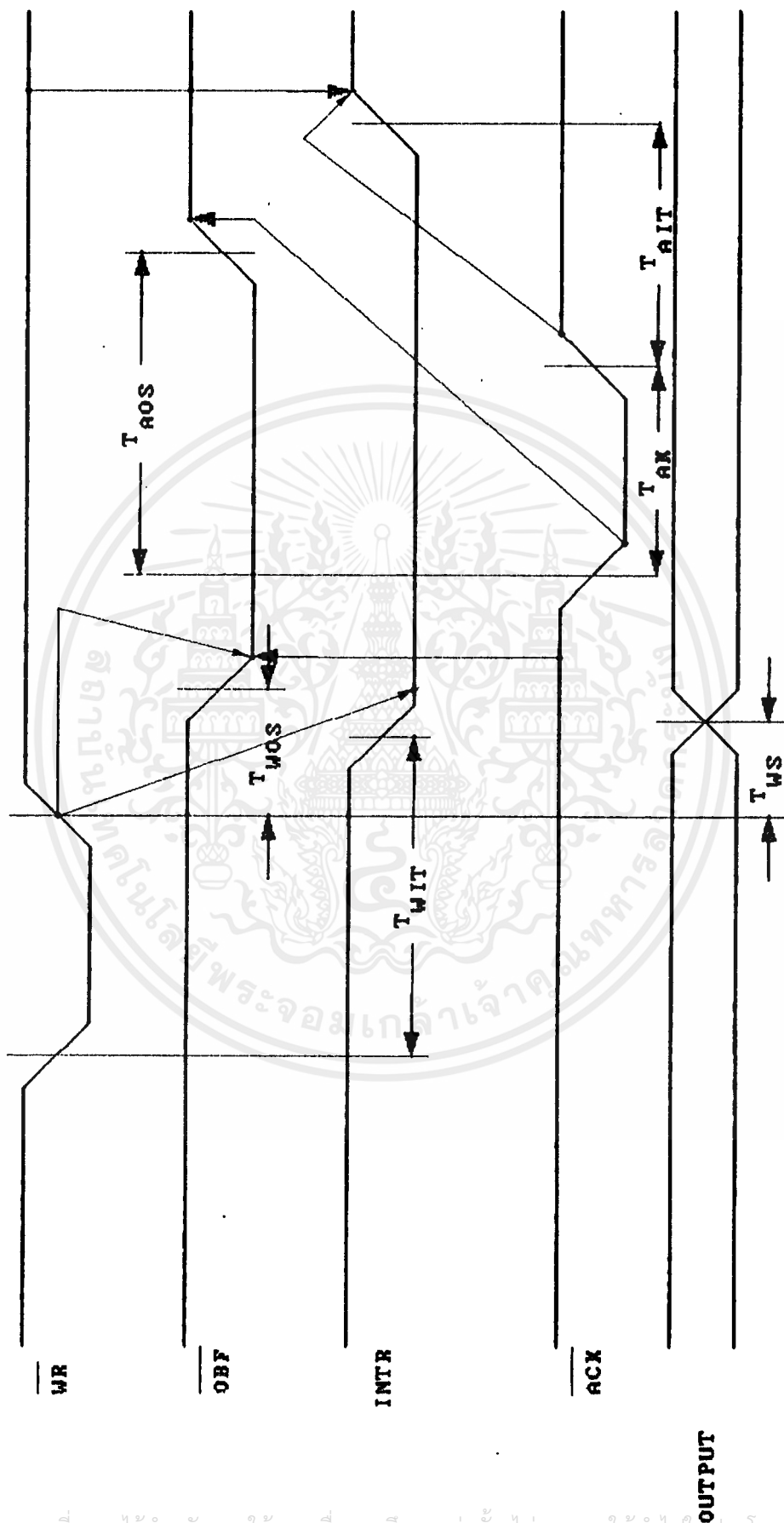
จากรูปที่ 6.1 เป็นฮาร์ดแวร์ Interface ระหว่าง Master CPU และ Slave CPU โดยมี 8255 เป็นตัวกลาง และเราใช้ 8255 ใน Mode 1 (Hand Shake) ซึ่งมีหลักการทำงานดังต่อไปนี้

Mode 1 : อินพุต/เอาต์พุตมาตรฐาน (Standard I/O) ในโหมดนี้เป็นโหมด Handshaking และ Interrupt Driven โดยที่พอร์ท A,B เป็นพอร์ทข้อมูล (Data Port) และพอร์ท C เป็นพอร์ทสถานะ (Status port) ซึ่งเราโปรแกรมให้ 8255 สามารถทำงานในโหมดต่างๆนี้ สามารถทำได้โดยการตั้ง คอนโทรลเวิร์ด (Control Word) และ set bit การทำงานของพอร์ท C ซึ่งเป็นพอร์ท แสดงสถานะของพอร์ท A และ B สามารถอธิบายโดยใช้ไดอะแกรมเวลา (Timing Diagram)

รูปที่ 6.2 Mode 1 : input port timing



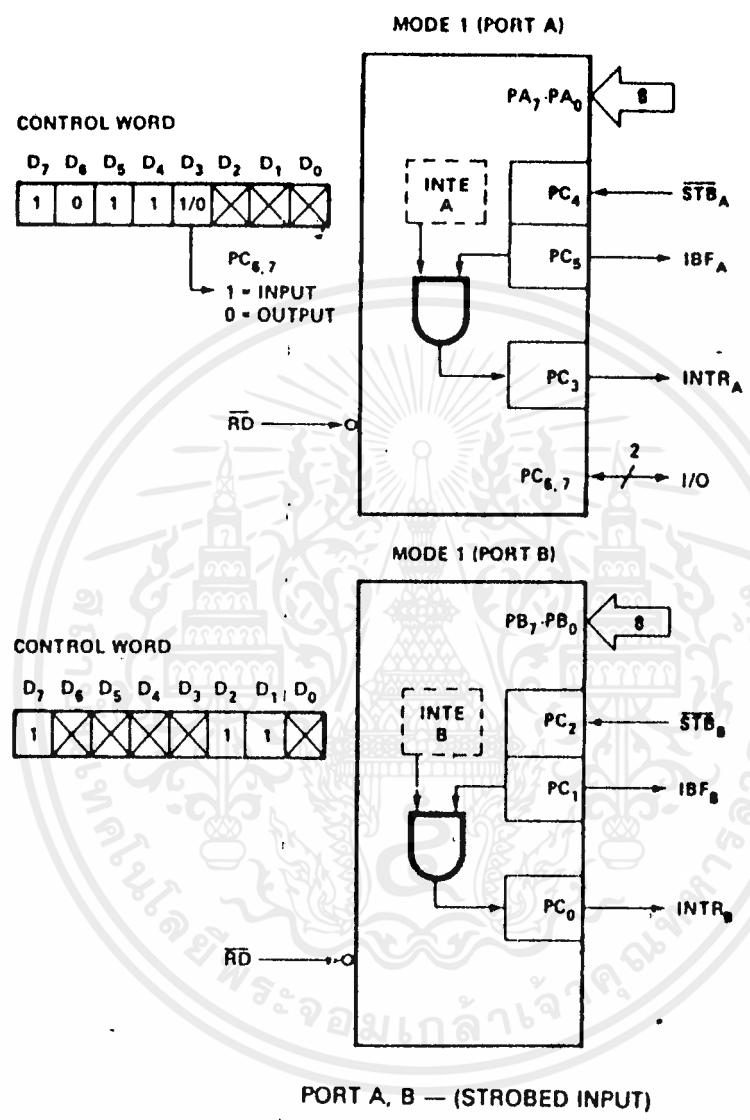
รูปที่ 6.2 Mode 1 : input port timing



รูปที่ 6.3 Mode 2 : output port timing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

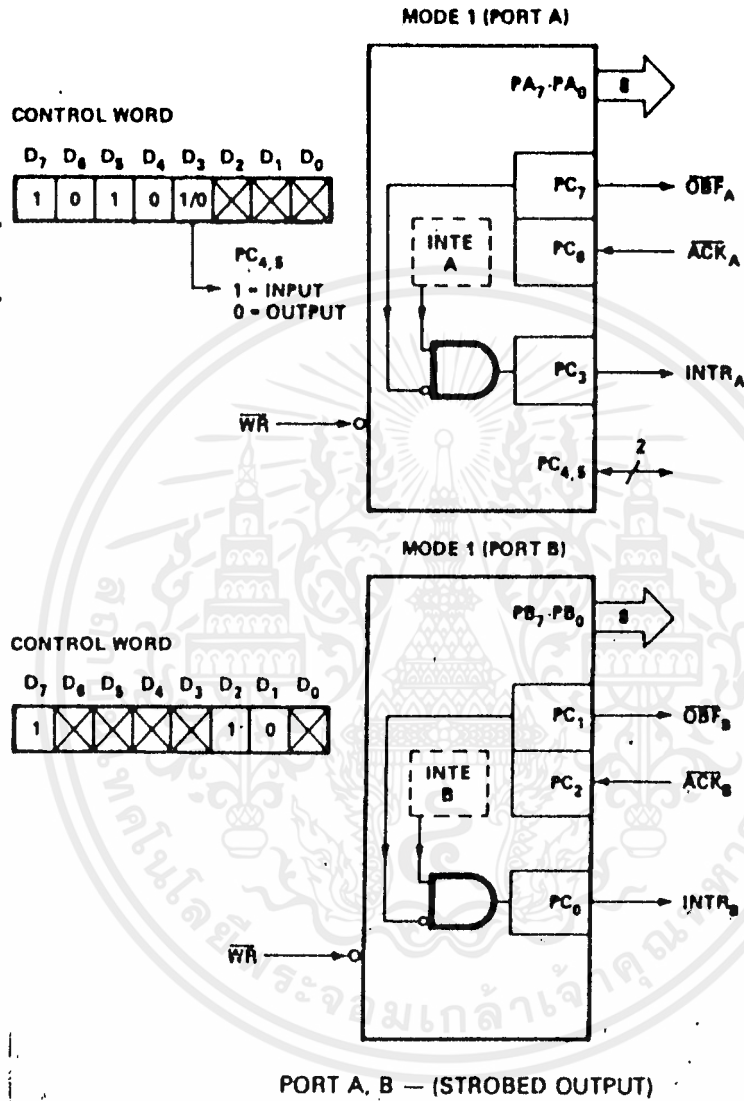
6.2 Control Word



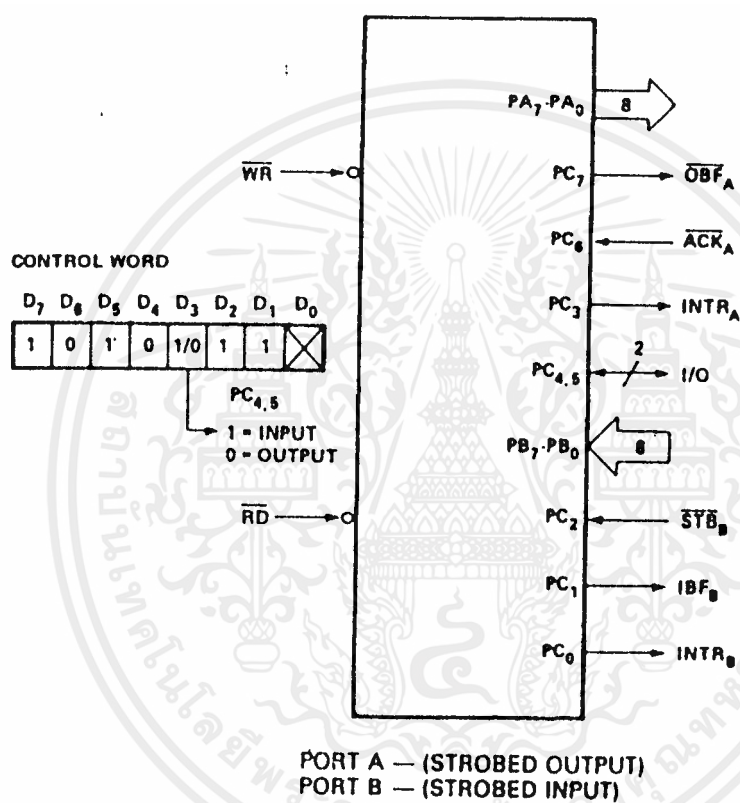
รูปที่ 6.4 (ก)

รูปที่ 6.4 คอนโทรลเวิร์ด ของ 8255 โมด 1

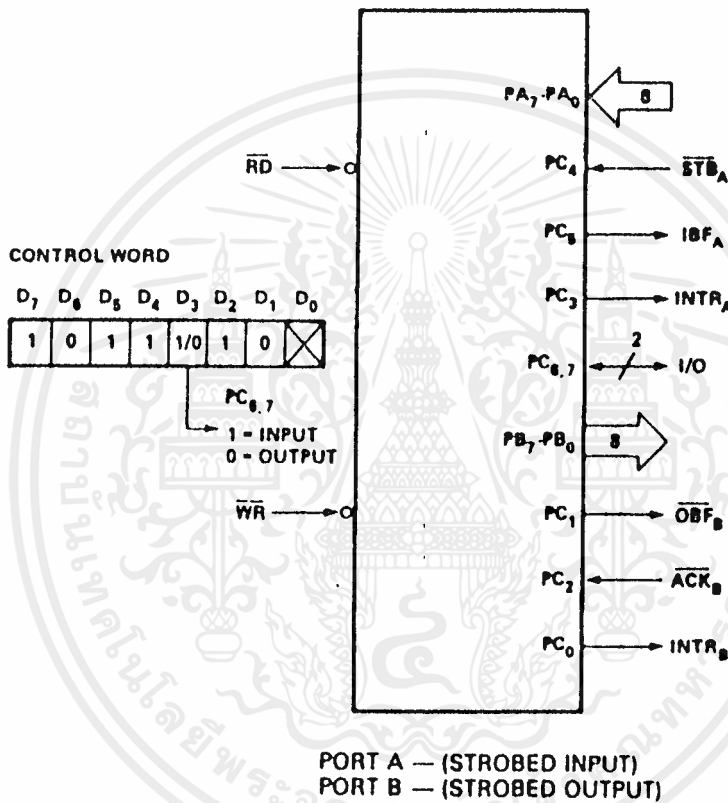
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 (ข).



รูปที่ 6.4 (ค)

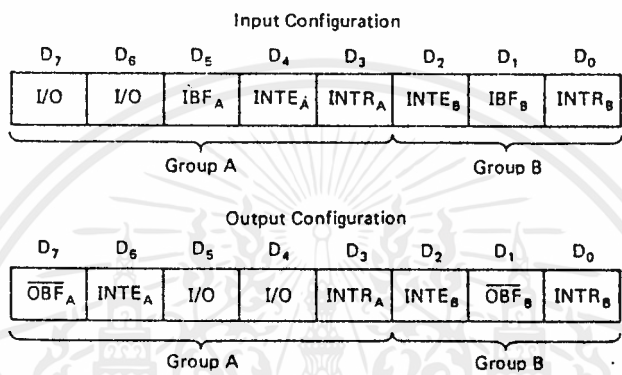


รูปที่ 6.4 (ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 Polling versus interrupts

8255 Mode 1 นี้สามารถที่จะโปรแกรมให้มีการ Interrupt ได้เช่น พอร์ต A มีข้อมูลเข้ามา Interrupt ก็จะแอกทีฟ (Active) ตามที่เรากำหนดได้



รูปที่ 6.5 รูปแบบ Status Word ของ 8255 โมด 1

6.4 รูปแบบการ Set bit

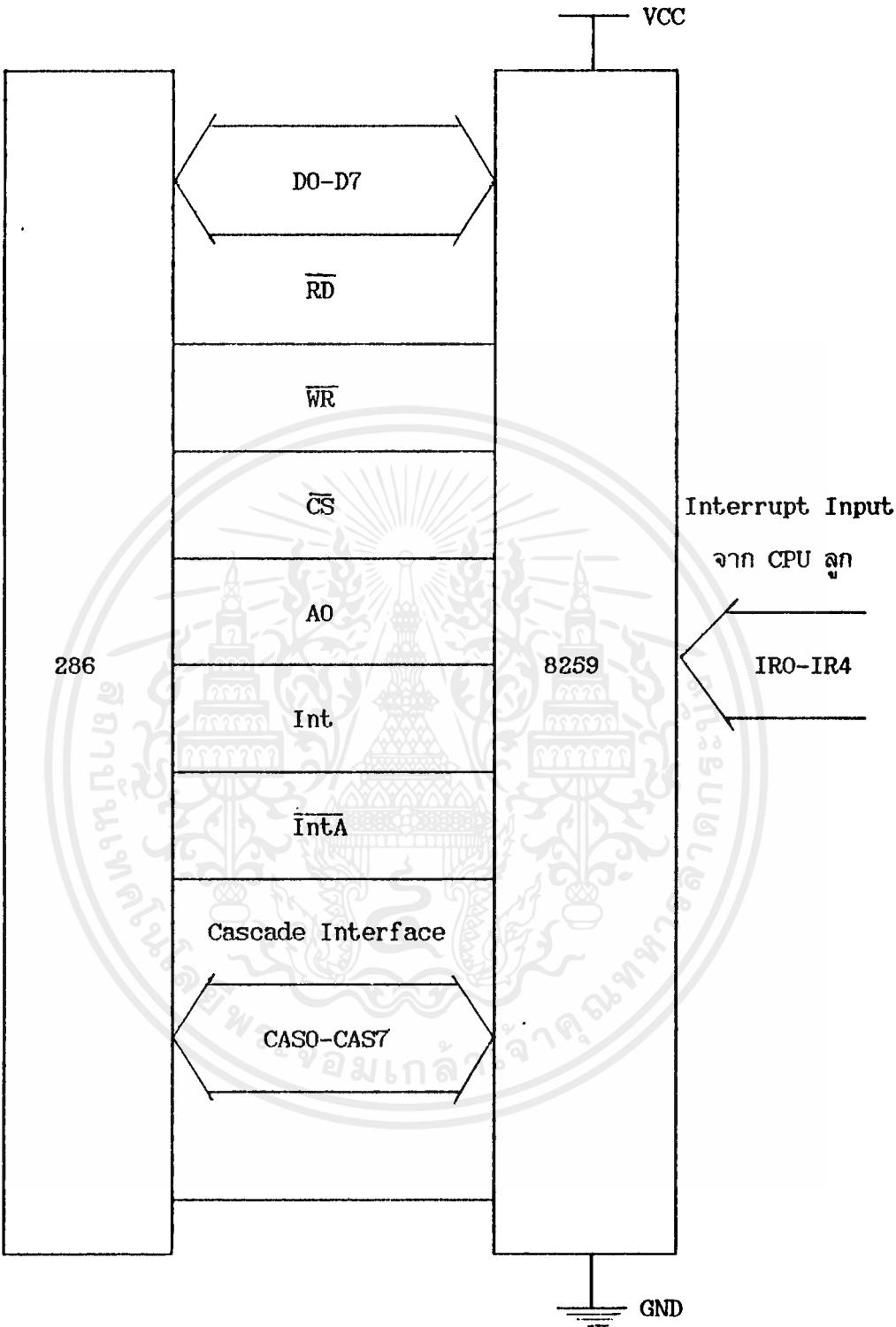
7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

- 0 set flag
- 1 set mode

3	2	1	set bit
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

6.5 โครงสร้างการจัดลำดับการอินเตอร์รัพท์

ในส่วนของฮาร์ดแวร์จะใช้ 8259 เป็นตัวควบคุมการจัดลำดับการอินเตอร์รัพท์ ซึ่งมีพอร์ทแอดเดรส (Port Address) อยู่ที่ 20H และ 21H สามารถแสดงวงจรทางฮาร์ดแวร์ได้ดังนี้



รูปที่ 6.6 การต่อฮาร์ดแวร์อินเตอร์รัพท์

Interrupt ของ Master CPU (ที่ใช้ในระบบ Mobile Robot) ที่ส่งมาจาก Slave CPU
ทั้ง 3 ตัวเราจะใช้ COM2,LPT1 และ LPT2 โดยมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IRQ 03 — COM2 ใช้กับอินเทอร์รัพท์ระบบอัลตราโซนิก

IRQ 05 — LPT2 ใช้กับอินเทอร์รัพท์ระบบล้อหลัง

IRQ 07 — LPT1 ใช้กับอินเทอร์รัพท์ระบบล้อหน้า

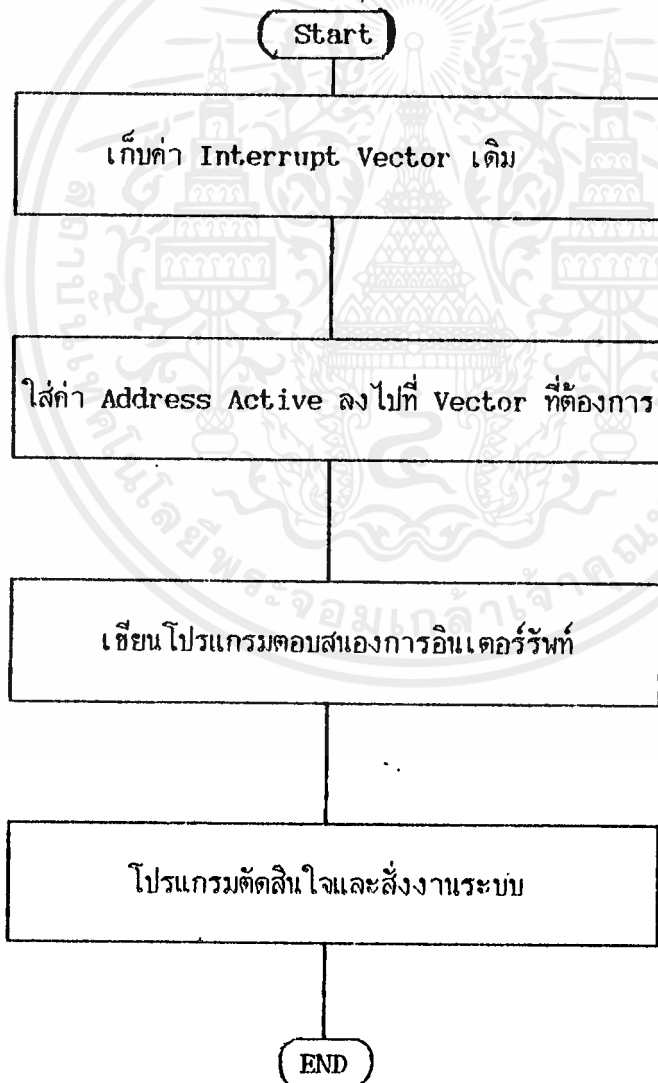
อินเทอร์รัพท์เวคเตอร์ COM2 — 0Bh

LPT1 — 0Fh

LPT2 — 0Dh

6.6 การอินเทอร์รัพท์ทางซอฟต์แวร์ (Software Interrupt)

ในส่วนของ Software Interrupt เราสามารถเขียนเป็นโฟลวชาร์ตได้ดังนี้



ในการโปรแกรมส่วนของ Software Interrupt จะใช้ภาษา C เป็นตัวจัดระบบในส่วนนี้ โดยมีตัวอย่างดังนี้

```
old_intcom2 = getvect(0xb);
old_intlpt2 = getvect(0xf);
old_intlpt1 = getvect(0xd);
setvect(0xb, com2);
setvect(0xf, lpt2);
setvect(0xd, lpt1);
```

ตัวอย่าง โปรแกรมตอบสนองอินเตอร์รัพท์

```
void Interrupt lpt1(void)
{
    call Active Interrupt
}

void Active Interrupt(void)
{
}
```

การโปรแกรมในส่วนของการตอบสนองการอินเตอร์รัพท์ ให้ทำงานเร็วที่สุดและไม่ใช้ฮาร์ดแวร์ ในส่วนที่มีการอินเตอร์รัพท์ มาเกี่ยวข้องเพราะจะทำให้ CPU จัดลำดับการอินเตอร์รัพท์ตอบสนองต่อฮาร์ดแวร์ไม่ทัน อันจะส่งผลให้ CPU แอ้ง (Hang) ได้ ซึ่งต้องระวังในส่วนนี้

บทที่ 7

การทดลองการหลบหลีกสิ่งกีดขวางของหุ่นยนต์

7.1 ลักษณะการโปรแกรม

ในการเคลื่อนที่ของหุ่นยนต์ จะมีการอ่านข้อมูลระยะทางระหว่างสิ่งกีดขวางกับตัวหุ่นยนต์ โดยส่วนของอัลตราโซนิก จะทำหน้าที่อ่านข้อมูลระยะทางและส่งให้กับคอมพิวเตอร์หลัก เมื่อระยะทางระหว่างวัตถุกับหุ่นยนต์มีค่าเท่ากับค่าที่กำหนดเอาไว้ ที่ระยะห่าง ≤ 150 cm โปรแกรมระบบจะเข้าสู่โปรแกรมย่อยในการสั่งเลี้ยว เมื่อพบว่าวัตถุอยู่ในระยะ ≤ 150 cm หุ่นยนต์ ก็จะหยุดและวิเคราะห์ทิศทางการเลี้ยว โดยอ่านค่าระยะทางจากอัลตราโซนิก แชนแนลที่ 2-7 มาทำการวิเคราะห์หาทิศทางที่จะไป โดยปกติแล้วเราจะกำหนดไว้ว่า ถ้าไม่มีวัตถุ อยู่ทางซ้ายหรือทางขวาของวัตถุที่ตรวจจับได้ จะให้หุ่นยนต์หลบไปทางซ้ายเสมอ ตามรูปที่ 7.2 แต่ถ้ามีวัตถุอยู่ทางซ้ายหรือขวาของวัตถุที่ตรวจจับได้ หุ่นยนต์ก็จะหลบไปในทิศทางตรงกันข้ามเสมอ และถ้ามีวัตถุอยู่ทางซ้ายและขวา โปรแกรมจะสั่งให้หุ่นยนต์ถอยหลัง กลับสู่ตำแหน่งเริ่มต้นใหม่

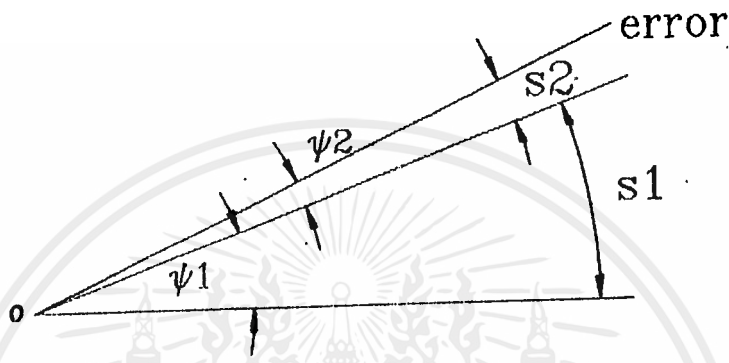
7.2 การนิเทศผลจากตำแหน่งเป้าหมาย

หลังจากที่ได้โปรแกรมการเคลื่อนที่ของหุ่นยนต์ตามแนวความคิด และทดสอบโดยการให้หุ่นยนต์เดินตามโปรแกรมพร้อมทั้งหลบหลีกวัตถุ และต้องกลับเข้าเส้นทางเดิม (เส้นประในรูป) โดยเดินตามโปรแกรมหลาย ๆ ครั้ง พบว่ายังมีความผิดพลาดซึ่งเกิดจากการอ่านค่าตำแหน่งล้อหน้า และล้อหลัง โดยเป็นเหตุผลทางไดนามิก (Dynamic) เช่นในการหลบหลีกสิ่งกีดขวางที่จะต้องมีการหยุด (Brake) ทำให้หุ่นยนต์ลื่นไถลไปจากจุดที่หยุดจริงที่คำนวณได้ ทำให้การเดินในครั้งต่อ ๆ มา มีความผิดพลาดเพิ่มขึ้นอีก โดยการเลี้ยวต่าง ๆ จะเคลื่อนที่แบบมีจุดหมุน (เป็นวงกลม) ดูรูปที่ 7.1

จากการทดลองยังพบว่า การเข้าหาจุดอ้างอิง (เส้นประ) ของหุ่นยนต์ จากการเลี้ยวหลบสิ่งกีดขวางไปทางซ้าย หุ่นยนต์สามารถเข้าหาเส้นอ้างอิงได้ถูกต้องมากกว่าการเข้าหาเส้นอ้างอิงจากการเลี้ยวหลบวัตถุไปทางขวา (ดูรูปที่ 7.2 เปรียบเทียบกับรูป 7.3)

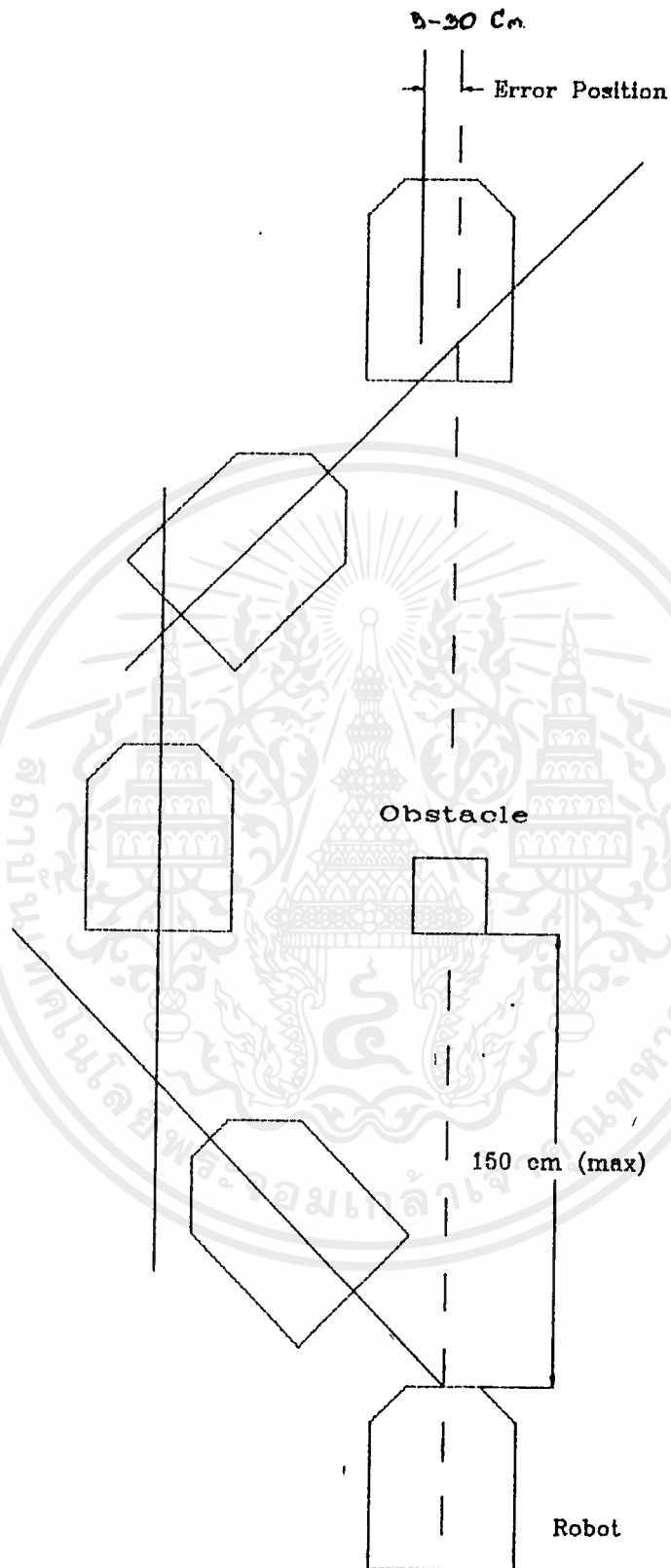
การทดลองยังได้ทำการโปรแกรมไว้ เมื่อหุ่นยนต์ตรวจพบวัตถุกีดขวางทั้ง 3 ด้าน คือ ด้านหน้า ด้านซ้าย และด้านขวา โปรแกรมก็จะสั่งให้หุ่นยนต์เดินถอยหลัง กลับไปยังตำแหน่งเริ่มต้นรูปที่ 7.4 ประกอบ

นอกจากนี้ ยังได้มีการทดลองให้หุ่นยนต์ ทลบทลึงสิ่งกีดขวาง ซึ่งวางไว้อย่างไม่เป็นระเบียบ พบว่า หุ่นยนต์ยังทลบทลึงได้ไม่ถี่เก้ มีการชนสิ่งกีดขวางในบางครั้ง เนื่องจากไม่ทราบขนาดของสิ่งกีดขวาง จึงไม่สามารถคำนวณมุมเลี้ยวที่เหมาะสมที่สามารถทลบทลึงวัตถุกีดขวางได้



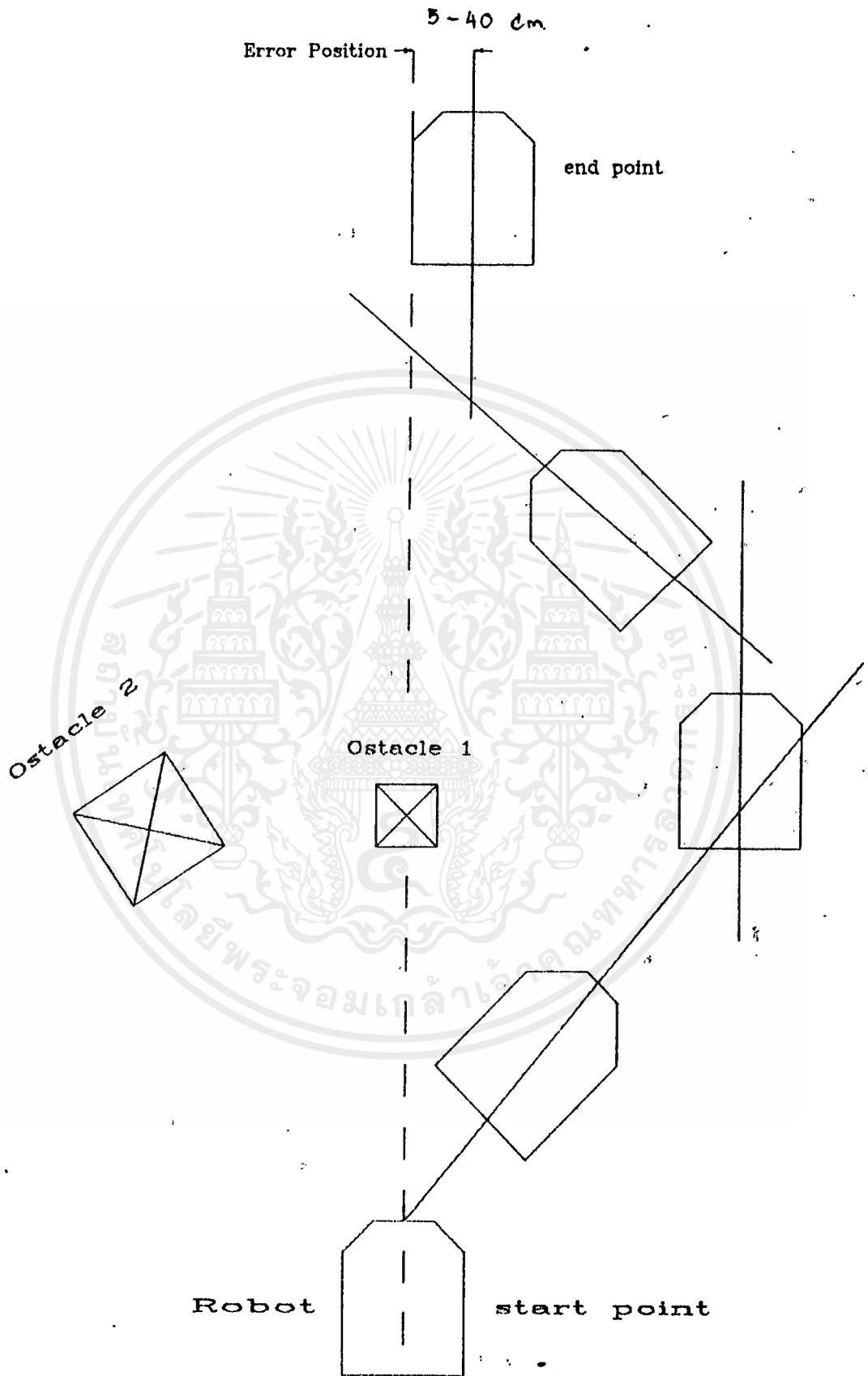
รูปที่ 7.1 แสดงจุดหมุนในการเลี้ยว

0	คือ	จุดหมุนในการเลี้ยว
1	คือ	มุมเลี้ยวของล้อหน้า
S1	คือ	ระยะทางในการเคลื่อนที่ของหุ่นยนต์
2	คือ	มุมเลี้ยวของล้อหน้าที่ผิดพลาด
S2	คือ	ระยะทางในการเคลื่อนที่ของหุ่นยนต์ที่ผิดพลาดไป



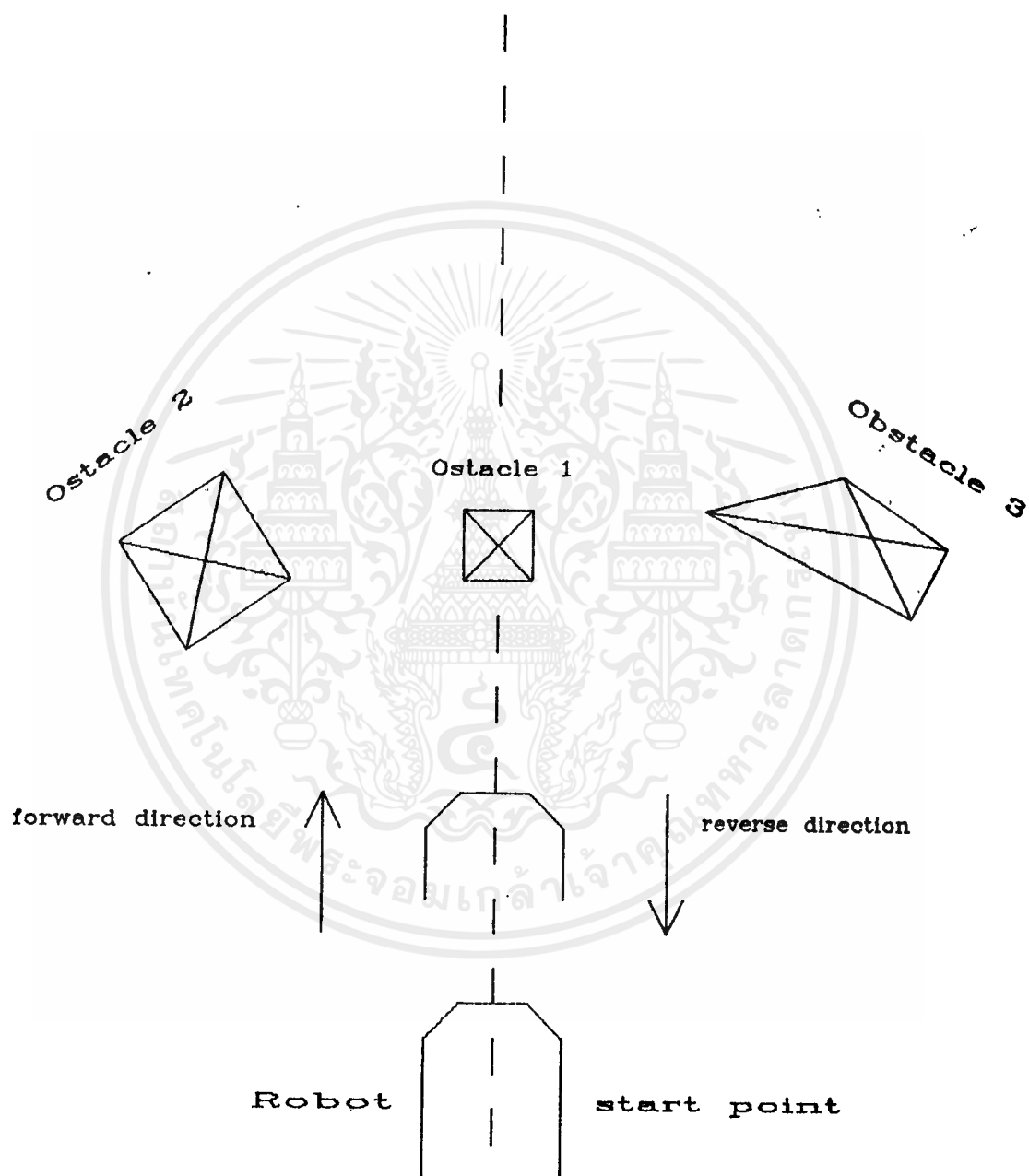
รูปที่ 7.2 การหลบหลีกของหุ่นยนต์เมื่อพบสิ่งกีดขวางด้านหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



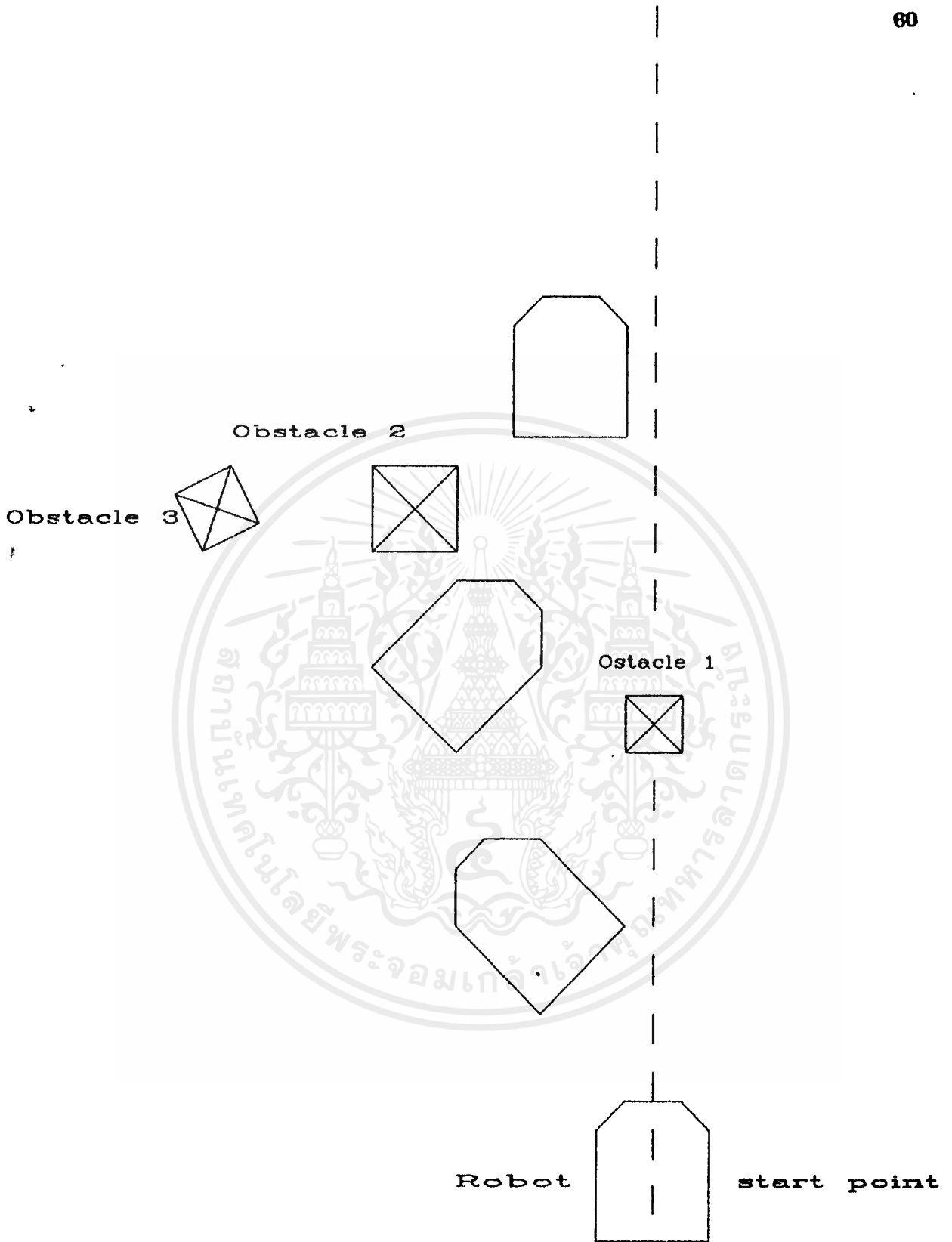
รูปที่ 7.3 การหลบหลีกของหุ่นยนต์เมื่อพบสิ่งกีดขวางด้านหน้าและด้านซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.4 การหลบหลีกของหุ่นยนต์เมื่อพบสิ่งกีดขวางด้านหน้า, ด้านซ้ายและด้านขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.5 การหลบหลีกของหุ่นยนต์เมื่อพบสิ่งกีดขวางที่วางไม่เป็นระเบียบ

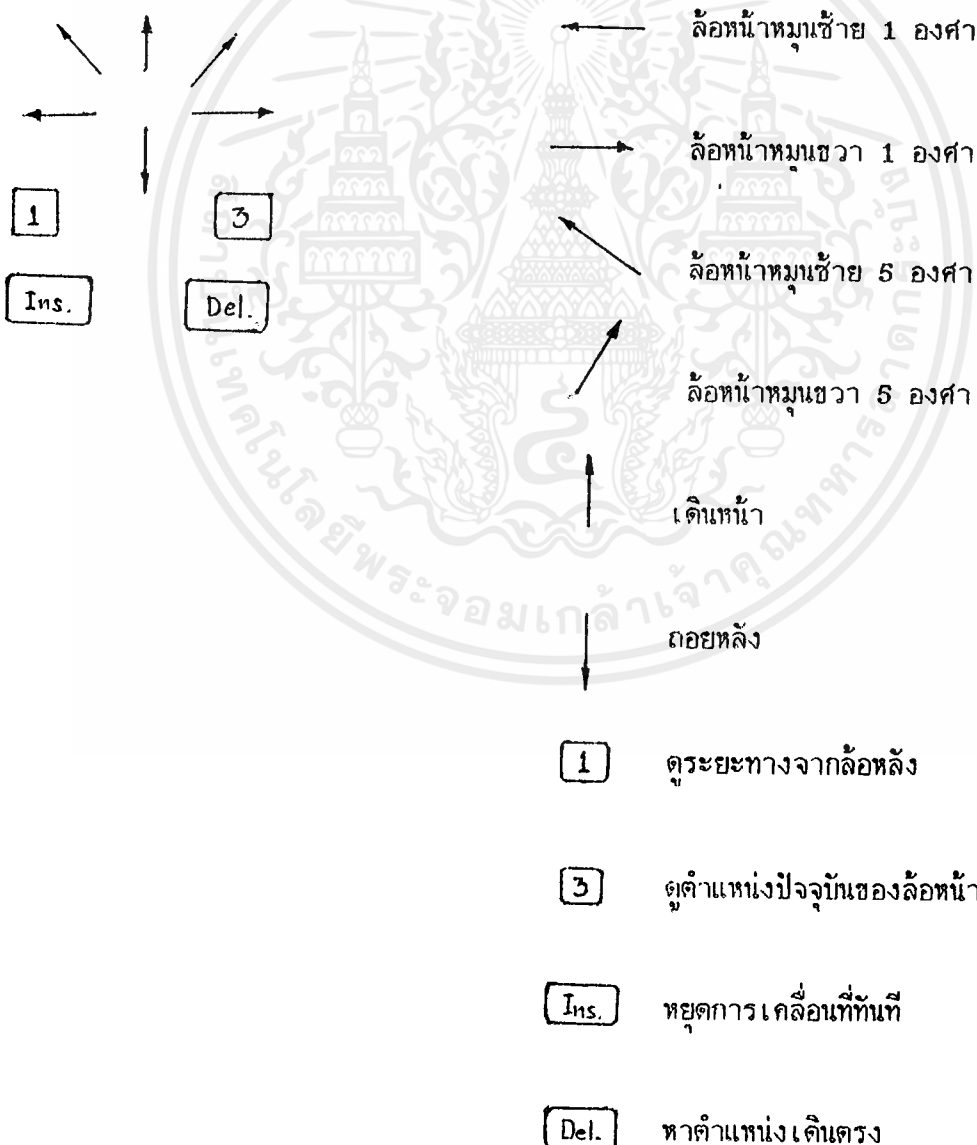
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3 การโปรแกรมใช้งาน

ฟังก์ชันการควบคุมหุ่นยนต์ ซึ่งเขียนโปรแกรมรับการควบคุมการเคลื่อนที่ของหุ่นยนต์ด้วย คีย์บอร์ด และ ได้เขียนเป็น include file เป็นภาษาซีไว้ ซึ่งผู้พัฒนาในระยะต่อไปสามารถนำ ฟังก์ชันเหล่านี้มาใช้งานได้ โดยไม่ต้องเขียนใหม่

โปรแกรมควบคุมการเคลื่อนที่ของหุ่นยนต์ มีชื่อโปรแกรม ว่า ROBOT19.EXE ซึ่งรับคำสั่ง ควบคุมทางคีย์บอร์ดโดยตรง ดังมีรายละเอียดต่อไปนี้

7.4 ฟังก์ชันการควบคุม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- F1 ออกจากระบบควบคุม
- F2 Reset CTC (Counter)
- F3 Initial CTC (Counter)
- F4 Start CTC
- F5 Initial ระบบควบคุมล้อยหลัง
- F6 หยุดระบบควบคุมล้อยหลังทั้งหมด
- F7 loop ช้าย Active
- F8 loop ขวา Active
- F9 แสดงผลความเร็วตัวรถ
- F10 หุ่นยนต์เคลื่อนที่ได้เองอัตโนมัติ
- "S" แสดงผลระยะทางของ Sensor ที่ตรวจจับได้
- "G" อ่านค่า Sensor จากทุกเซนแนล



บทที่ 8

สรุปผลการทำงานและการทดลอง

8.1 สรุปผลการทำงานของระบบนำร่อง

1. เราใช้ CPU Z80180 หนึ่งตัวควบคุมการทำงานของระบบนำร่อง จึงทำให้เกิดความเป็นอิสระในการคำนวณระยะทางของวัตถุที่ขวาง และไม่เ็นภาระกับ CPU แม่
2. ผลที่คำนวณ ได้อาจมีการผิดพลาดเนื่องมาจากนอยส์จากภายนอกและปัญหาครอสทอล์ค (crosstalk)

8.2 ข้อดีของระบบนำร่องโดยใช้อุลตราโซนิก

1. ต้นทุนต่ำในการสร้างชุดเซนเซอร์อัลตราโซนิกและสามารถสร้างได้ง่าย
2. ความเร็วในการตอบสนองเร็วกว่าเซนเซอร์แบบอื่น ๆ เช่น เซนเซอร์แบบพรีอักษิมีตี้

8.3 ข้อบกพร่องของระบบนำร่องโดยใช้อุลตราโซนิก

1. ถ้าทิศทางในการบีมของอัลตราโซนิกไม่ดีจะทำให้ความเที่ยงตรงในการคำนวณตำแหน่งของวัตถุผิดพลาด แต่ทั้งนี้ขึ้นอยู่กับระยะทางของวัตถุที่ขวางกับแกนของการบีม
2. การอ่านระยะทางที่ผิดพลาดบ่อยครั้ง ซึ่งมีสาเหตุมาจากนอยส์ภายนอกและสเตรย์รีเฟล็กชัน (Stray reflections) จากเซนเซอร์ตัวอื่น ๆ (crosstalk)
3. จะเกิดการสะท้อนของบีมในลักษณะเช่นเดียวกันกับแสงสะท้อนกระจกเงาขึ้น เมื่อมุมระหว่างเซนเซอร์และผิวหน้าวัตถุแคบเกินไป
4. เกิดการรบกวนเช่นแนลอื่นที่มีความถี่ตรงกัน การสะท้อนคลื่นกลับเข้ามายังตัวรับของตัวส่งที่มีความถี่ตรงกัน ทำให้ตัวรับของแนลอื่นรับคลื่นได้ และมีการวิเคราะห์วัตถุที่ผิดพลาด

8.4 แนวทางในการพัฒนาระบบนำร่อง

เพื่อลดปัญหาที่สรุปไว้ในข้อบกพร่องข้อ (2) เราอาจใช้วิธีการแสดงการมีอยู่ของวัตถุที่คิด ขวางด้วยวิธีการกำหนด certainty grid ซึ่งสามารถศึกษาวิธีการทำงานได้จาก Reference [12]

8.5 สรุปการผลการทำงานของระบบบังคับเลี้ยว

ในการพัฒนาระยะที่ 3 ระบบบังคับเลี้ยวได้สร้างชิ้นใหม่ทั้งหมดเนื่องจากระบบเก่านี้ใช้งานไม่ได้ไม่เท่าที่ควร จึงได้มีการออกแบบระบบใหม่ ให้สามารถใช้งานได้ดีกว่า โดยใช้ CPU (Z80180) มาควบคุมการทำงานของระบบโดยเฉพาะ และนอกจากนี้ยังได้ออกแบบเซนเซอร์ของระบบป้อนกลับชิ้นใหม่ซึ่งเป็นแบบ optical encoder ทำให้ระบบทำงานได้ประสิทธิภาพที่ดี การสื่อสารข้อมูลกับ CPU แม้ ใช้การอินเทอร์รัพท์ทำให้มีการตอบสนองของการเลี้ยวต่อการสั่งงานจาก CPU แม้เป็นไปในลักษณะ Real-Time มากขึ้น ผลการทดสอบพบว่า การควบคุมคำสั่งต่าง ๆ เป็นไปอย่างรวดเร็ว ไม่มีการหน่วงเวลาและการแกว่งขณะเลี้ยว ความละเอียดของมุมเลี้ยวมีความละเอียดถึง 1 องศา ต่อสแต็ปการเลี้ยว 1 สแต็ป ซึ่งนับว่ามีความละเอียดสูงกว่าการพัฒนาในขั้นที่ผ่านมา ซึ่งเลี้ยวได้เพียง 6 องศาต่อสแต็ปและมีการแกว่งก่อนเข้าตำแหน่งที่ต้องการ

8.6 ข้อเสนอแนะสำหรับการพัฒนาในขั้นต่อไป

จากการทดสอบพบว่า การหมุนของดีซีมอเตอร์ในทิศทางตามและทวนเข็มนาฬิกามีแรงบิดไม่เท่ากัน ทำให้การควบคุมการเลี้ยวมีข้อผิดพลาดอยู่บ้าง ถึงแม้ว่าจะได้พยายามแก้ไขข้อผิดพลาดด้วยการชดเชยทางซอฟต์แวร์แล้วก็ตาม คาดว่าสาเหตุอาจจะเกิดจากความผิดภายในมอเตอร์ (Motor's friction) ดังนั้นควรจะใช้มอเตอร์ตัวอื่นมาทดสอบ

อัตราการผลิตเฟืองระหว่างมอเตอร์กับเพลาทวนล้อ ซึ่งมีอัตราทด 1 : 40 นั้นยังน้อยเกินไป ถ้าสั่งมอเตอร์หมุนเร็วเกินไป การเลี้ยวก็จะเร็วตาม แต่ถ้าลดความเร็วของมอเตอร์ลงมามาก ๆ ก็จะทำให้แรงบิดของมอเตอร์น้อยลง จนไม่สามารถบังคับเลี้ยวได้ ถ้าเพิ่มอัตราทดเฟืองขึ้นไปอีก อาจจะเป็น 1 : 100 หรือมากกว่า จะทำให้การหมุนเลี้ยวช้าลงในขณะที่

แรงบิดยังคงที่ ซึ่งจะเป็ผลดีต่อการบังคับเลี้ยวมาก

การออกแบบ optical encoder นั้นได้ทำไว้แบบง่าย เพื่อต้องการลดต้นทุน ถ้าหากต้องการแบบที่มีคุณภาพและมีความละเอียดมาก ๆ ต้องซื้อจากโรงงานผู้ผลิต ซึ่งมีราคาแพงมาก เพราะฉะนั้น ถ้าเป็นไปได้ก็ควรจะใช้ optical encoder ที่ได้ออกแบบไว้นี้ แต่ควรระวังอย่าให้ฝุ่นละอองและคราบน้ำมันจับที่แผ่น encoder เนื่องจากจะทำให้เกิด error ได้

8.7 สรุปผลการทำงานของระบบควบคุมความเร็ว

ระบบควบคุมความเร็วล้อหลัง นั้นเปลี่ยนแปลงวงจรใหม่เกือบทั้งหมด คงไว้เฉพาะวงจรขับกำลัง (Power Drive) การพัฒนาในระยะที่ 3 ใช้ CPU 1 ตัวเพื่อควบคุมการทำงานของระบบควบคุมความเร็วทั้งหมด ทำให้ระบบทำงานได้อย่างอิสระ มีการประมวลผลเองภายในระบบ การติดต่อสื่อสารกับ Master CPU ใช้การอินเตอร์เวิร์ท ทำให้การตอบสนองต่อการเปลี่ยนแปลงทางแมคคานิคเป็นไปในลักษณะ Real-Time การสื่อสารข้อมูลเป็นแบบขนานซึ่งมีความเร็วมากกว่าระบบเดิมที่ใช้เป็นแบบอนุกรม

การควบคุมความเร็วและตำแหน่งของการพัฒนาในระยะที่ 3 นี้ นับว่าประสบความสำเร็จเป็นอย่างสูง การควบคุมความเร็วและระยะทางทำได้อย่างแม่นยำ โดยอาศัยระบบป้อนกลับจากเซ็นเซอร์แบบ optical encoder ที่ได้ออกแบบเองเช่นกัน และยังมีแรงบิดสูงมากสามารถบรรทุกโหลดที่มีขนาด 60 กิโลกรัมได้อย่างสบาย ๆ

อย่างไรก็ตามข้อบกพร่องของระบบควบคุมความเร็วก็ยังมีอยู่ การที่ระบบควบคุมความเร็วมอเตอร์ให้ระบบเฟสล็อกคัล Phase Locked Loop (PLL) ยังมีข้อจำกัดที่ว่าระบบยังใช้ช่วงเวลาในการล็อกเพื่อให้เข้าสู่ steady state นาน เป็นผลทำให้การเคลื่อนที่ของหุ่นยนต์ในระยะเริ่มต้นออกตัวได้ช้า เราสามารถแก้ไขได้โดยเปลี่ยนโอซีระบบ PLL ให้เป็นเบอร์ที่คอมพลีเม้นท์กันทั้งหมดคือ L291, L292 และ L293 แทนวงจรเดิมที่ใช้ L292 เพียงตัวเดียวร่วมกับวงจรที่จำลองการทำงานของ L291 และ L293

8.8 สรุป

การพัฒนาหุ่นยนต์เคลื่อนที่เองในระยะที่ 3 ได้แก้ไขข้อบกพร่องของการพัฒนาในระยะที่ 2 ได้ทั้งหมด ตลอดระยะเวลาที่ทำการพัฒนา แม้ว่าผู้พัฒนาจะแยกการพัฒนาออกเป็นส่วนๆ แต่ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พัฒนาาร่วมกันและมีการปรึกษาแลกเปลี่ยนความคิดเห็นซึ่งกันและกันอยู่เสมอ หัวข้อสำคัญที่ได้พัฒนาขึ้นมาในระยะที่ 3 ตามแนวทางการแก้ไขของผู้พัฒนาในระยะที่ 2 ได้เสนอแนะไว้มีดังนี้

1. มีการเพิ่มโปรเซสเซอร์ให้ระบบ โดยเพิ่มโปรเซสเซอร์เป็น 3 ตัว สำหรับควบคุมการทำงานของระบบทั้ง 3 และมีโปรเซสเซอร์อีกหนึ่งตัวคอยควบคุมการทำงานของระบบทั้ง 3 ยังผลให้การทำงานของระบบ มีประสิทธิภาพมากยิ่งขึ้น

2. มีการนำระบบ Real Time มาใช้กับระบบย่อยทุกระบบ เพื่อให้สามารถตอบสนองการเปลี่ยนแปลงทางแมคคานิคทันทั่วทั้ง

3. สามารถจัดปัญหาการสั่นกระตุกของล้อหน้าได้อย่างสิ้นเชิง

4. กำจัดรอยร้าวที่เกิดจากมอเตอร์ไม่ให้รบกวนการทำงานของระบบ

5. การป้อนกลับของระบบบังคับเลี้ยว จาก Optical Encoder เป็นแบบ Digital ทำให้การประมวลผลเป็นไปได้อย่างรวดเร็ว เพราะไม่ต้องมาผ่านกระบวนการแปลงสัญญาณแอนาล็อกมาเป็นสัญญาณ ดิจิตอล และยังสามารถลดข้อผิดพลาดในการแปลงสัญญาณได้อีกด้วย

6. ใช้อัลกอริทึม ในการทำ Path Planning ให้กับระบบเพื่อช่วยในการกำหนดเส้นทางการตัดสินใจหลบหลีกวัตถุกีดขวาง

8.9 ข้อเสนอแนะ

1. เนื่องจากการพัฒนาหุ่นยนต์เคลื่อนที่อัตโนมัติเป็นงานวิจัยที่ต้องอาศัยวิชาการหลายแขนง การวิจัย จึงน่าจะใช้กลุ่มผู้วิจัยที่มีความรู้ในสาขาวิชาเหล่านั้น และควรมีการวิจัยโดยกลุ่มผู้วิจัยกลุ่มนั้นอย่างต่อเนื่อง เพราะจะได้ไม่เสียเวลามาศึกษาาระบบ ที่สำคัญก็คือควรมีทุนอุดหนุนการวิจัยที่พอเพียง เนื่องจากเป็นงานวิจัยที่ต้องใช้เครื่องมือที่ทันสมัยและมีราคาค่อนข้างสูง

2. Master CPU ควรจะพัฒนาไปเป็นแบบ Card ที่มีขนาดเล็กเพราะจะเป็นการประหยัดพื้นที่ และลดน้ำหนักลงได้มาก

3. ควรใช้อุปกรณ์ตรวจจับวัตถุกีดขวางที่สามารถหาขนาด หรือ มิติของวัตถุได้ เพราะจะช่วยให้การหลบหลีกวัตถุของหุ่นยนต์ถูกต้องสมบูรณ์ยิ่งขึ้น

4. ควรเพิ่มอุปกรณ์ตรวจจับวัตถุกีดขวางให้สามารถค้นหาวัตถุกีดขวางได้รอบตัว เพื่อช่วยการหลบหลีก

5. ควรมีการนำระบบ AI(Artificial Intelligence) มาใช้กับการพัฒนาหุ่นยนต์เคลื่อนที่ได้เอง

กิตติกรรมประกาศ

ขอขอบพระคุณ อ.เกษตร์ ศิริสันติสัมฤทธิ์ เป็นอย่างสูง ที่ได้ให้การประสิทธิ์ประสาทวิชาความรู้ ให้คำแนะนำปรึกษาแก่คณะผู้จัดทำ ให้การสนับสนุนทางด้านอุปกรณ์ใช้งาน จนปริญญานิพนธ์สำเร็จลุล่วงไปด้วยดี ขอบคุนเฟืสาทที่อำนวยความสะดวกทางด้านอุปกรณ์ตลอดจนเครื่องมือต่างๆ ขอบคุนเพื่อนะที่ให้กำลังใจ ในการทำงาน สุดท้ายคณะผู้จัดทำ ขอกราบขอพระคุณ คุณพ่อ คุณแม่ ที่ให้ทุนในการทำปริญญานิพนธ์ ทั้งยังเป็นกำลังใจสำคัญในการทำงาน และการศึกษาครั้งนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] กมล วรรณโสภา , การพัฒนาหุ่นยนต์เคลื่อนที่ ชั้นที่ 2 , คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง กรุงเทพฯ , 2534
- [2] ชีเอ็ด ยูเคชั่น , คู่มือ/เทียบเบอร์ ไอซี TTL , บ.ชีเอ็ด ยูเคชั่น , 2531
- [3] ธาณินทร์ ถาวรศาสนวงศ์,ทินกร ตึก , การอินเตอร์เฟส IBM PC , สำนักพิมพ์ ฟิสิกส์ เซนเตอร์ การพิมพ์
- [4] ยืน ภู่วรรณ และคณะ , เทคโนโลยีไมโครคอมพิวเตอร์ 16 บิต , บ.ชีเอ็ด ยูเคชั่น จำกัด , 2531
- [5] วิบูลย์ ชื่นแขก , ไมโครโปรเซสเซอร์ , สำนักพิมพ์สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ
- [6] A.Meystel, Autonomous Mobile Robots Vehicles With Cognitive Control ,World Scientific Publishing ,1991
- [7] Avtar Singh,Walter A. Triebel , The 8086 and 80286 Microprocessors Hardware,Software,and Interfacing , Prentice Hall , 1990
- [8] Borland Inc., Turbo C reference manual , Borland Inc. ,
- [9] Gordon Mccomb, Robots Builder's (The) BONANZA 99 Inexpensive Robotic Projects , Mc Millan ,1989
- [10] Harry H. Poole, Fundamentals of Robotics Engineering , Van Nostrand Reinhold , 1989
- [11] James W. Coffron , Z80 Applications , SYBEX Inc. , America , 1983
- [12] J.Borenstein and Y.Koren , "Real-Time Ostacles Avoidance for Fast Mobile Robots" , IEEE Trans. on SYS,MAN, and CYBER. vol.19, no.5 , p1179-1187 , 1989

- [13] John Uffenbeck , Microcomputers and Microprocessors the 8080 , 8085 and Z-80 programming , interfacing , and troubleshooting , Prentice-Hall International Edition ,1991
- [14] Lance A. Leventhal , Z80 Assembly Language Programming , Osborne/McGraw-Hill , Berkeley,California , 1979
- [15] Michael Tischer , PC intern system programming , Abacus,1992
- [16] Motorola Inc., Motorola high-speed CMOS logic data Handbook, Motorola Inc.,1989
- [17] Russel Rector-George Alexy , The 8086 Book , Osborne/McGraw-Hill , Berkeley,California , 1980
- [18] Z-World , Z-World Z80 "C" User's Manual , Z-World , California , 1988