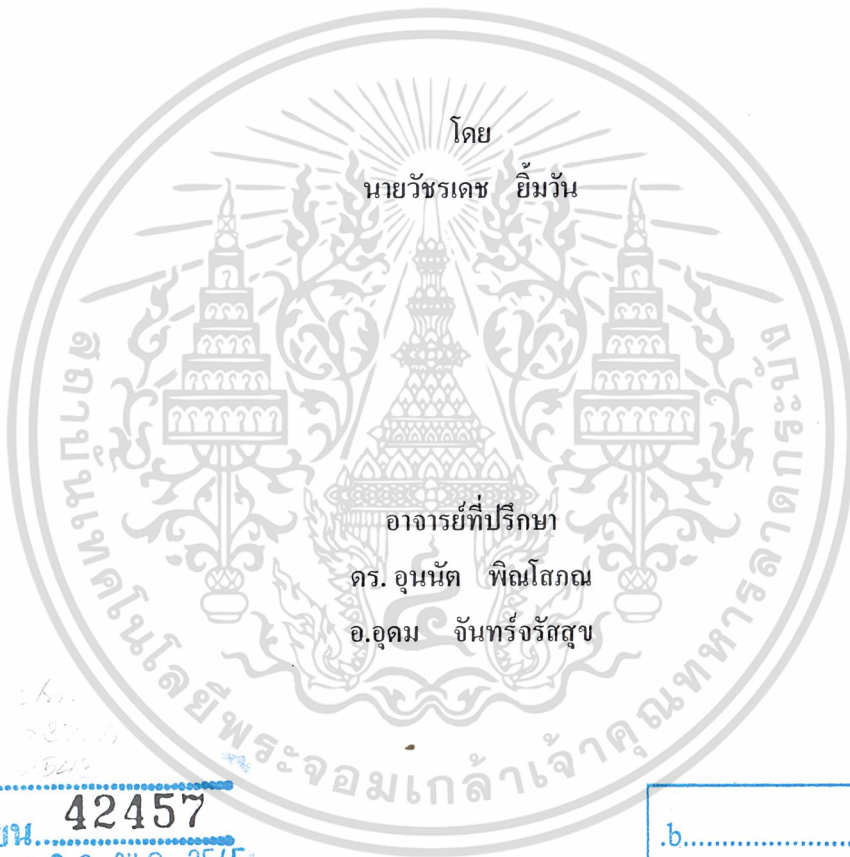




การควบคุมแขนกลด้วยคอมพิวเตอร์

Computer Controlled Robot



เลขที่.....
เลขทะเบียน..... 42457
วัน, เดือน, ปี..... 23 พ.ค. 2545

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6 1120415x

ปริญญาานิพนธ์ปีการศึกษา 2543

ภาควิชา วิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การควบคุมแขนกลด้วยคอมพิวเตอร์

COMPUTER CONTROLLED ROBOT

ผู้จัดทำ

1. นาย วัชรเดช ยี่มวัน รหัสประจำตัว 40010706



[Handwritten signature]

อาจารย์ที่ปรึกษา

(ดร. อุन्नัต พิณโสภณ)

[Handwritten signature]

อาจารย์ที่ปรึกษา

(อ.อุดม จันทร์จรัสสุข)

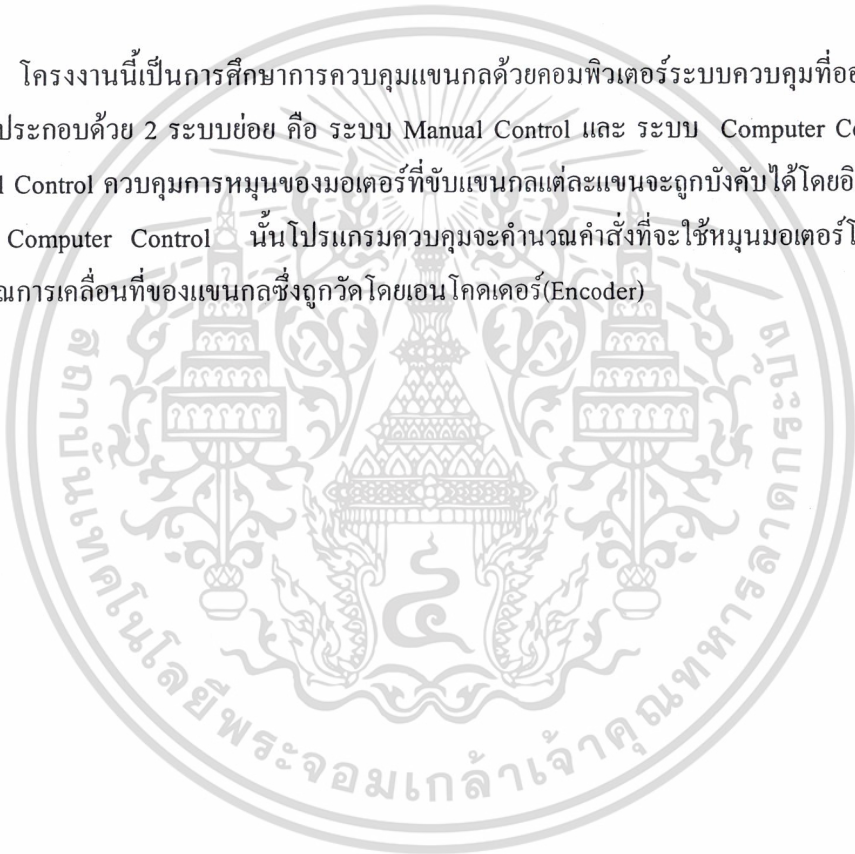
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมแขนกลด้วยคอมพิวเตอร์

นายวัชรเดช ยิ้มวัน 40010706
ดร.อนุนต์ พิณโสภณ อาจารย์ที่ปรึกษา
อ.อุดม จันทร์จรัสสุข อาจารย์ที่ปรึกษา
ปีการศึกษา 2543

บทคัดย่อ

โครงการนี้เป็นการศึกษาการควบคุมแขนกลด้วยคอมพิวเตอร์ระบบควบคุมที่ออกแบบในโครงการนี้ ประกอบด้วย 2 ระบบย่อย คือ ระบบ Manual Control และ ระบบ Computer Control ในระบบ Manual Control ควบคุมการหมุนของมอเตอร์ที่ขับเคลื่อนแต่ละแขนจะถูกบังคับได้โดยอิสระ สำหรับในระบบ Computer Control นั้น โปรแกรมควบคุมจะคำนวณคำสั่งที่จะใช้หมุนมอเตอร์โดยพิจารณาจากสัญญาณการเคลื่อนที่ของแขนกลซึ่งถูกวัดโดยเอนโคเดอร์(Encoder)



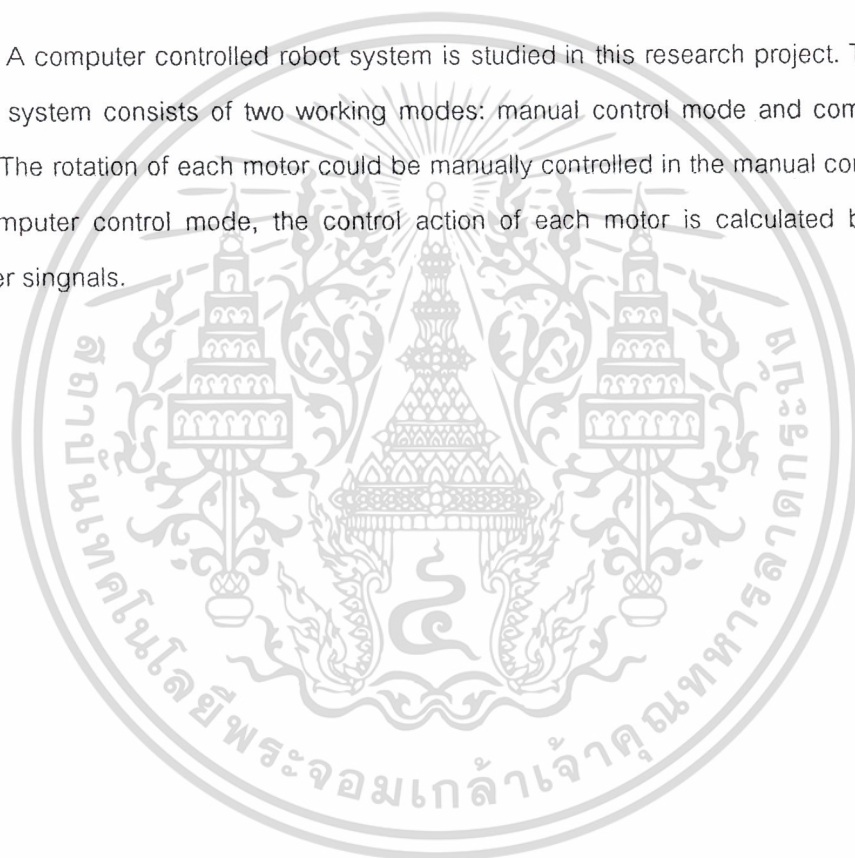
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Robots by Computer Control

Watcharadet	Yimwun	
Dr.Unnut	Pinsopon	Advisor
Udom	Janjarassuk	Advisor

Abstract

A computer controlled robot system is studied in this research project. The proposed control system consists of two working modes: manual control mode and computer control mode. The rotation of each motor could be manually controlled in the manual control mode. In the computer control mode, the control action of each motor is calculated based on the encoder signals.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
สารบัญ	III
สารบัญรูป	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของงานวิจัย	1
1.3 ขอบเขตงานวิจัย	2
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ส่วนประกอบของแขนกล	3
2.1 คุณลักษณะของแขนมนุษย์	4
2.2 การแบ่งประเภทของแขนกลอุตสาหกรรม	5
2.2.1 การแบ่งประเภทของแขนกลโดยอาศัยการเคลื่อนที่และโครงสร้างภายนอก	5
2.2.1.1 แขนกลที่มีการเคลื่อนไหวได้ในระบบพิกัดขั้ว	5
2.2.1.2 แขนกลที่มีการเคลื่อนไหวได้ในระบบพิกัดเชิงขั้ว	5
2.2.1.3 แขนกลที่มีลักษณะแขนเป็นข้อต่อ	6
2.2.1.4 แขนกลที่มีการเคลื่อนไหวได้ในระบบพิกัดคาร์ทีเซียน	7
2.2.2 การแบ่งประเภทของแขนกลโดยอาศัยทางเดินของแขนกล	7
2.3 การเคลื่อนที่ของแขนกล	8
2.3.1 การเคลื่อนที่ของแขน	8
2.3.2 การเคลื่อนที่ของมือ	8
2.4 End Effector	9
2.4.1 แบบมือจับ	9
2.4.2 มือที่เป็นอุปกรณ์	10
2.5 การสั่งงาน	11
2.5.1 การสั่งงานแบบ Manual	11
2.5.2 การสั่งงานแบบ Lead Through	12
2.5.3 การสั่งงานด้วยโปรแกรม	13
2.5.4 การประยุกต์ใช้งาน	13

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการเขียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
	14
บทที่ 3	21
2.6 ลักษณะการทำงานของแขนกล Scrobot ER3	21
ทฤษฎีและหลักการ	21
3.1 โครงสร้างและชนิดของมอเตอร์ไฟฟ้ากระแสตรง	22
3.1.1 มอเตอร์ไฟฟ้ากระแสตรงแบบเส้นแรงแม่เหล็กมีค่าคงที่	22
3.2 รายละเอียดของมอเตอร์ที่ใช้ในแขนกล Scrobot ER3	23
3.3 การควบคุมความเร็วรอบของมอเตอร์ไฟฟ้ากระแสตรง	23
3.3.1 หลักการทำงานของวงจรควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง	25
3.4 ตัวจับสัญญาณ (Sensor) และ เอนโคเดอร์ในระบบการควบคุมมอเตอร์	27
3.4.1 ความละเอียดของเอนโคเดอร์ที่นับการเพิ่มค่า	28
3.4.2 เอาท์พุทของเอนโคเดอร์	31
3.5 ทฤษฎีของ ไอซี 8255	31
3.5.1 โครงสร้างเบื้องต้นของ ไอซี 8255	31
3.5.2 การแบ่งพอร์ตของ ไอซี 8255	31
3.5.3 ขาต่าง ๆ ของ ไอซี 8255	32
3.5.4 รีจิสเตอร์ภายในของ ไอซี 8255	32
3.5.5 การใช้งาน ไอซี 8255	33
3.5.6 การกำหนดโหมดการทำงาน	33
3.6.1 การทำงานโหมด 0	33
3.6.2 การทำงานโหมด 1	35
3.6.3 การทำงานในโหมด 2	35
3.7 รายละเอียดเกี่ยวกับการ์ด ET-PC8255	39
บทที่ 4	39
ขั้นตอนการทำงาน	39
4.1 การทำงานของโปรแกรมควบคุมแขนกล Scrobot ER3	46
4.2 การเชื่อมต่อการใช้งานการ์ด ET-PC 8255	47
4.3 การส่งสัญญาณควบคุมความเร็วมอเตอร์และรับสัญญาณจากเอนโคเดอร์	51
4.4 สายสัญญาณจากแขนกล Scrobot ER3	53
บทที่ 5	53
บทวิจารณ์และสรุป	53
5.1 ปัญหาและแนวทางแก้ไข	53
5.1.1 ปัญหาทางด้านฮาร์ดแวร์และแนวทางแก้ไข	53
5.1.2 ปัญหาทางด้านซอฟต์แวร์และแนวทางแก้ไข	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
5.2 แนวทางการพัฒนา	53
5.3 บทสรุป	53
กิตติกรรมประกาศ	54
เอกสารอ้างอิง	55
ภาคผนวก ก. รูปแผนวงจรควบคุม	56
ภาคผนวก ข. รายละเอียดโปรแกรมควบคุม	58
ภาคผนวก ค. Data Set	77



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 1 แสดงลักษณะแขนกล Scorbob ER3 ที่ใช้ในโครงการนี้	1
รูปที่ 2.1 แสดงลักษณะของแขนกลในอุตสาหกรรม	3
รูปที่ 2.2 แสดงลักษณะการทำงานของแขนกลในอุตสาหกรรม	4
รูปที่ 2.3 แสดงแขนกลที่มีการเคลื่อนไหวในระบบพิกัดเชิงขั้ว	5
รูปที่ 2.4 แสดงแขนกลที่มีการเคลื่อนไหวในระบบพิกัดทรงกระบอก	6
รูปที่ 2.5 แสดงแขนกลที่มีการเคลื่อนไหวในลักษณะข้อต่อ	6
รูปที่ 2.6 แสดงแขนกลที่มีการเคลื่อนไหวในระบบพิกัดคาร์ทีเซียน	7
รูปที่ 2.7 แสดงลักษณะการเคลื่อนที่ของแขนกลแบบต่าง ๆ	8
รูปที่ 2.8 แสดงการเคลื่อนที่ของมือ	9
รูปที่ 2.9 แสดงลักษณะของมือจับแบบต่าง ๆ	10
รูปที่ 2.10 แสดงลักษณะของมือที่เป็นอุปกรณ์แบบต่าง ๆ	11
รูปที่ 2.11 แสดงลักษณะการสั่งงานแบบ Lead Through โดยอาศัย Robot Simulator	12
รูปที่ 2.12 แสดงลักษณะของแขนกล Scorbob ER3	14
รูปที่ 2.12 (ต่อ) แสดงลักษณะของแขนกล Scorbob ER3	14
รูปที่ 2.13 แสดงลักษณะ Link 1	15
รูปที่ 2.14 แสดงลักษณะ Link 2	15
รูปที่ 2.15 แสดงลักษณะ Link 3	16
รูปที่ 2.15 (ต่อ) แสดงลักษณะ Link 3	16
รูปที่ 2.16 แสดงลักษณะ Link 4	17
รูปที่ 2.17 แสดงลักษณะ Link 5	17
รูปที่ 2.18 แสดงลักษณะ Link 6	18
รูปที่ 2.19 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง Link 1 กับ Link 2	18
รูปที่ 2.20 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง Link 2 กับ Link 3	19
รูปที่ 2.21 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง Link 3 กับ Link 4	19
รูปที่ 2.22 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง Link 4 กับ Link 5	19
รูปที่ 2.23 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง Link 5 กับ Link 6	20
รูปที่ 2.24 แสดงการทำงาน Link 6	20
รูปที่ 3.1 แสดงมอเตอร์ไฟฟ้ากระแสตรงแบบเส้นแรงแม่เหล็กมีค่าคงที่	21
รูปที่ 3.2 แสดงรายละเอียดของวงจรควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง	23
รูปที่ 3.3 แสดงแผนผังเวลาแสดงการทำงานในส่วนพัลส์วิดธ์มอดูเลเตอร์	24

	หน้า
รูปที่ 3.4 แสดงระบบเอนโคเดอร์แบบโรตารี	25
รูปที่ 3.5 แสดงเอนโคเดอร์ที่นับการเพิ่มค่าแบบลิเนียร์	26
รูปที่ 3.6 แสดงตัวอย่างกลไกของออฟโตอินคริमेंท์	26
รูปที่ 3.7 แสดงตัวอย่างตัวจับสัญญาณแบบมีช่องปิดเปิดให้แสงผ่านได้ช่องเดียว และแบบมีหลายช่อง	26
รูปที่ 3.8 แสดงถึงผลของแสงที่เดินในแนวเดียวกันและแสงที่ตกกระทบ	27
รูปที่ 3.9 แสดงความสัมพันธ์ระหว่างความเป็นลิเนียร์กับความไม่ได้อินทรีย์กลาง	28
รูปที่ 3.10 (ก) แสดงตัวอย่างลูกคลื่นเอาท์พุทสี่เหลี่ยมของอุปกรณ์เอนโคเดอร์ ช่องเดียว(ไบโคเร็กซ์ัน)	29
(ข) แสดงสัญญาณเอนโคเดอร์ 2 ช่องมีมุมเฟสต่างกัน 90 องศา(สองทิศทาง)	29
รูปที่ 3.11 แสดงวงจรรับค่าจากเอนโคเดอร์โดยใช้ไอซี LM339	30
รูปที่ 3.12 แสดงการจัดเรียงขาของไอซี 8255	31
รูปที่ 3.14 แสดงลักษณะของการ์ด ET-PC8255	35
รูปที่ 3.15 แสดงค่าต่าง ๆ ที่ส่งให้พอร์ตควบคุมเพื่อกำหนดการทำงานของ 8255	37
รูปที่ 3.15 (ต่อ) แสดงค่าต่าง ๆ ที่ส่งให้พอร์ตควบคุมเพื่อกำหนดการทำงานของ 8255	38
รูปที่ 4.1 แสดงระบบควบคุมแขนกล Scorbob ER3 ด้วยคอมพิวเตอร์	39
รูปที่ 4.2 แสดงลักษณะของโปรแกรมควบคุมแขนกล Scorbob ER3	40
รูปที่ 4.3 แสดงลักษณะของโปรแกรมควบคุมแขนกล Scorbob ER3 ในโหมด AUTO	40
รูปที่ 4.4 แสดงลักษณะของโปรแกรมควบคุมแขนกล Scorbob ER3 ในโหมด Manual	41
รูปที่ 4.5 แสดงรูปที่เกิดจากฟังก์ชัน BASE(),LINK1(),LINK2()และLINK3()	44
รูปที่ 4.6 แสดงการเซ็ต DIP SW ของการ์ด PC8255	46
รูปที่ ก.1 แสดงวงจรควบคุมความเร็วมอเตอร์	57
รูปที่ ก.2 แสดงวงจรที่รับค่าจากเอนโคเดอร์	57
รูปที่ ก.3 แสดงบอร์ดควบคุมแขนกล Scorbob ER3	57

สารบัญตาราง

	หน้า
ตารางที่ 3.1 แสดงความหมายของสัญญาณตาม RD-WR-A1-A0	32
ตารางที่ 3.2 แสดงสถานะของขาต่าง ๆ ในพอร์ต C ในการทำงานโหมด 1	34
ตารางที่ 3.3 แสดงสถานะของขาต่าง ๆ ในพอร์ต C ในการทำงานโหมด 2	35
ตารางที่ 4.1 แสดงรหัสซึ่งเป็นผลจากการเปลี่ยนโหมดและความหมายของรหัส	42
ตารางที่ 4.2 แสดงความหมายของเลข 8 บิต	47
ตารางที่ 4.3 แสดงการใช้งานของพอร์ต PC8255	48
ตารางที่ 4.4 แสดงความหมายที่ได้จากสัญญาณที่ได้รับจากเอนโคเดอร์	50
ตารางที่ 4.5 แสดงสายสัญญาณจากตัวคอนเนกเตอร์ 50 ขาจากแขนกล Scorbot ER3	51



บทที่ 1

บทนำ

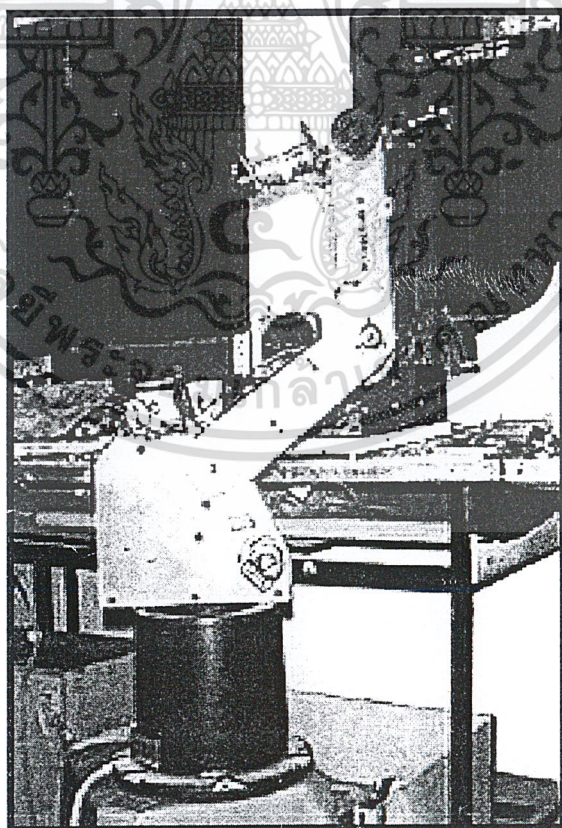
1.1 ความสำคัญและที่มา

ในปัจจุบันนี้วิทยาการและเทคโนโลยีมีความเจริญก้าวหน้าขึ้นจากในอดีตอย่างมาก มีการทำวิจัยและค้นคว้าเพื่อพัฒนาขีดความสามารถในการทำงานให้มากขึ้นและรวดเร็วยิ่งขึ้น ไม่ว่าจะเป็นโรงงานอุตสาหกรรมที่มีขนาดเท่าใดก็ตาม เพราะการพัฒนาความสามารถให้สูงขึ้นนั้นจะทำให้การทำงานในระบบสามารถทำงานได้ดีขึ้นหรือมีข้อผิดพลาดน้อยลง ในโรงงานอุตสาหกรรมนั้นมีเครื่องจักรอยู่หลายประเภท และมีแขนกล(Robot Arm) ที่ช่วยแก้ปัญหาการทำงานของคนงานที่ไม่สามารถจะทำงานบางอย่างได้และยังมีประสิทธิภาพสูงในการทำงาน

การใช้แขนกลมาช่วยในการผลิตในโรงงานอุตสาหกรรมนั้นจำเป็นจะต้องมีการควบคุมการทำงานให้สอดคล้องกับงานที่ทำดังนั้นจึงได้นำคอมพิวเตอร์(Computer) มาใช้ในการควบคุมการทำงาน

การใช้คอมพิวเตอร์ควบคุมการทำงานมีข้อดีหลายประการ เช่น สามารถแก้ไขกระบวนการทำงานได้โดยง่าย สามารถเชื่อมโยงการทำงานเป็นเครือข่ายได้ เป็นต้น

ดังนั้นในโครงการนี้ได้นำคอมพิวเตอร์มาใช้ควบคุมการทำงานแขนกล Scorbot ER3 ดังรูปข้างล่างนี้ โดยโปรแกรมควบคุมการทำงานของแขนกลนั้นถูกเขียนขึ้นด้วยภาษาซี



รูปที่ 1.1 แสดงแขนกล Scorbot ER3 ที่ใช้ในโครงการนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของงานวิจัย

1.2.1 ศึกษาการใช้คอมพิวเตอร์ควบคุมอุปกรณ์ต่าง ๆ ที่มีความซับซ้อนในการทำงาน

1.2.2 ศึกษาการใช้โปรแกรม 3 มิติจำลองการทำงานของแขนกล Scorbob ER3 เพื่อความสัมพันธ์การทำงานของข้อต่อต่าง ๆ เพื่อประกอบในการเขียนโปรแกรมควบคุม

1.3 ขอบเขตของงานวิจัย

โครงการนี้ได้เขียนโปรแกรมควบคุมแขนกล Scorbob ER3 เป็นแบบกราฟฟิกส์(Graphic)ในลักษณะ 2 มิติซึ่งการหมุนของข้อต่อในโปรแกรมจะมีลักษณะเดียวกับการหมุนของแขนกล Scorbob ER3 การควบคุมแขนกล Scorbob ER3 ไม่ได้เน้นเรื่องระยะเวลาในการเคลื่อนจากตำแหน่งหนึ่งไปยังอีกตำแหน่งหนึ่ง

1.4 วิธีการดำเนินงาน

โครงการนี้เริ่มต้นโดยทำการศึกษาทฤษฎีการควบคุมความเร็วของมอเตอร์ การรับสัญญาณจากเอนโคเดอร์(Encoder) การทำงานพื้นฐานของไอซี(IC)8255ดูรายละเอียดได้ในบทที่ 2 ส่วนในบทที่ 3 จะกล่าวรายละเอียดการเขียนโปรแกรมควบคุมและลักษณะการทำงานของแขนกล Scorbob ER3

บทที่ 2

ส่วนประกอบของแขนกล

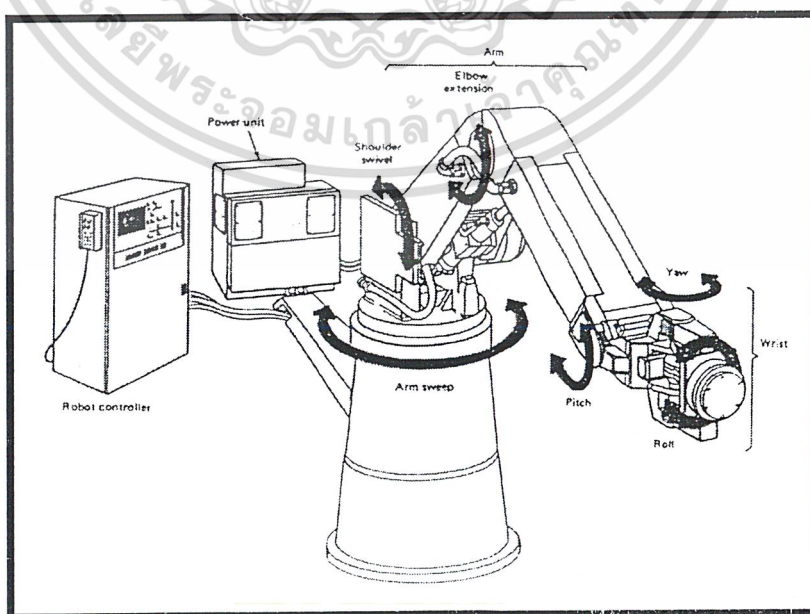
เมื่อก้าวถึงหุ่นยนต์คนส่วนใหญ่มักจะนึกถึงโครงเหล็กขนาดใหญ่ลักษณะคล้ายมนุษย์เคลื่อนตัวอย่างช้า ๆ ดังที่เห็นในภาพยนตร์แนววิทยาศาสตร์ทั่วไป แต่ในปัจจุบันได้ถูกพัฒนาขึ้นมาใช้งานในโรงงานอุตสาหกรรมขนาดใหญ่ มากขึ้นโดยอุตสาหกรรมเหล่านี้ครอบคลุมตั้งแต่โรงงานผลิตรถยนต์เครื่องจักร จนแม้แต่การผลิตไมโครคอมพิวเตอร์ก็ใช้หุ่นยนต์เข้ามาช่วยในการผลิตแต่หุ่นยนต์อุตสาหกรรมเหล่านี้มีลักษณะที่แตกต่างออกไป จากภาพพจน์เดิมที่นิยายวิทยาศาสตร์วาดภาพไว้มาก เพราะส่วนใหญ่จะมีลักษณะเป็นแขนและมือซึ่งใช้ทำงานตามคำสั่งเท่านั้น

สถาบันทางหุ่นยนต์ของอเมริกา(ROBOTICS INSTITUTE OF AMERICA:RIA) ให้นิยามว่า

“เป็นมือจับ (manipulator) หลายหน้าที่การทำงาน ที่สามารถโปรแกรมใหม่ได้ ได้รับการออกแบบเพื่อการเคลื่อนย้ายวัสดุ ชิ้นส่วน เครื่องมือหรืออุปกรณ์พิเศษ โดยอาศัยการเคลื่อนที่ที่ได้รับการโปรแกรมไว้ใช้งานเฉพาะอย่าง”

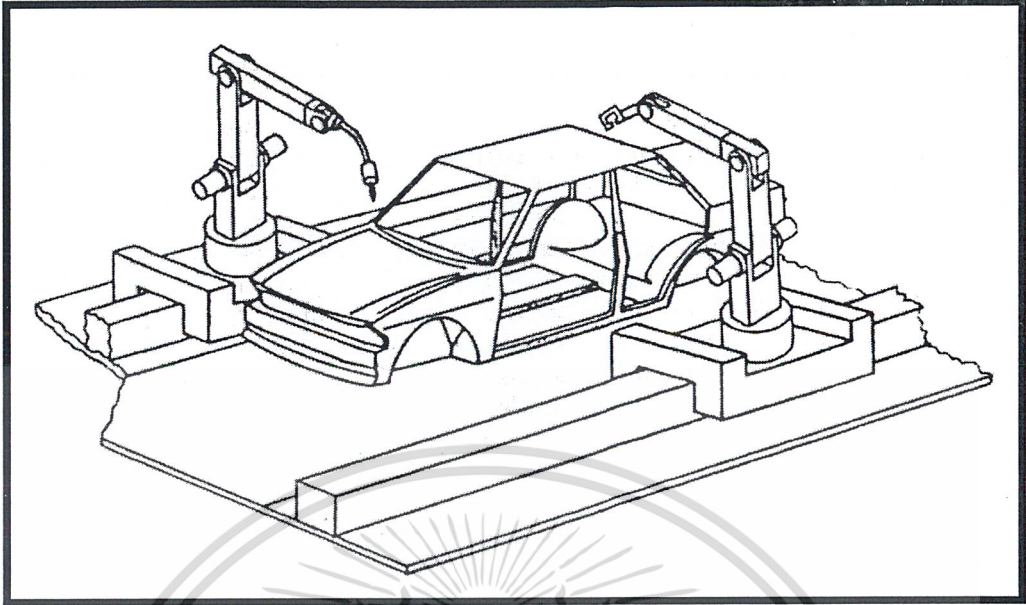
นิยามของสถาบันทางหุ่นยนต์ของอเมริกานี้ได้รับการยอมรับกันอย่างแพร่หลายในฐานะที่เป็นนิยามของหุ่นยนต์อุตสาหกรรม

ตัวอย่างของงานที่ใช้หุ่นยนต์ช่วยงานมีมากมายอาทิเช่น การประกอบชิ้นส่วน การพันสีการเชื่อม เป็นต้น โดยตำแหน่งที่หุ่นยนต์เคลื่อนที่ไปปฏิบัติงานนี้ยังคงมีตลอดเวลาและสามารถทำงานซ้ำ ๆ ได้หลาย ๆ ครั้ง โดยไม่มีการผิดพลาดจนกว่าจะมีการเปลี่ยนแปลงคำสั่ง การทำงานของหุ่นยนต์อุตสาหกรรมนี้ หลักการจะคล้ายกับการทำงานของอุปกรณ์ที่ควบคุมด้วยคำสั่งเชิงตัวเลข(numerical control machine) ในลักษณะของการทำงาน การ feedback เป็นต้น โดยมีแนวโน้มว่าจะมีการนำไปใช้ในอุตสาหกรรมขนาดเล็กเพิ่มขึ้นเพราะสามารถควบคุมได้ด้วย ไมโครคอมพิวเตอร์ทำให้การเปลี่ยนแปลงแก้ไขคำสั่งได้สะดวก



รูปที่ 2.1 แสดงลักษณะของแขนกลในอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดงลักษณะการทำงานของแขนกลในอุตสาหกรรม

2.1 คุณลักษณะของแขนมนุษย์

แขนกลเป็นการเลียนแบบการทำงานของแขนของมนุษย์ ดังนั้นจำเป็นต้องศึกษาโครงสร้างและลักษณะการเคลื่อนไหวของแขนมนุษย์ก่อน

แขนของมนุษย์ประกอบด้วยส่วน 2 ส่วนที่เห็นแยกกันอย่างชัด คือ ส่วนที่เป็นข้อมือ ซึ่งมีข้อต่อหลักอยู่ 3 ข้อต่อ และอีกส่วนคือแขน ซึ่งมีข้อต่อหลักอยู่ 2 ข้อต่อ (หัวไหล่และข้อศอก) ข้อมือของมนุษย์มีลักษณะดังต่อไปนี้

การหมุนข้อมือ (ROLL angle) สามารถหมุนได้ $-180 < \theta < 90 = 270$

การพับข้อมือ (PITCH angle) สามารถพับได้ $-90 < \theta < 50 = 140$

การบิดข้อมือ (YAW angle) สามารถบิดได้ $-45 < \theta < 15 = 60$

การเคลื่อนที่ของข้อมือทั้ง 3 แบบ มีการเคลื่อนที่ที่เป็นอิสระต่อกัน ดังนั้นเราจึงให้การเคลื่อนที่แต่ละแบบเป็น 1 องศาแห่งอิสระ

ส่วนที่สองคือ แขนประกอบด้วยข้อต่อหลัก 2 ข้อต่อ แต่มี 3 องศาแห่งอิสระคือ 2 องศาแห่งอิสระที่หัวไหล่และอีก 1 องศาแห่งอิสระที่ข้อศอก แต่หัวไหล่ของแขนกลจะมีเพียง 1 องศาแห่งอิสระ ลักษณะหนึ่งที่สำคัญของโครงสร้างของแขน คือ อัตราส่วนระหว่างความยาวของแขนท่อนบนกับแขนท่อนล่าง จะมีค่าราว ๆ 1.2 : 1 นั่นคือล่างจะต้องสั้นกว่าแขนท่อนบน ถ้าเราออกแบบให้แขนท่อนล่างยาวกว่าแขนท่อนบน จะทำให้ประสิทธิภาพการทำงานของแขนกลลดลง

2.2 การแบ่งประเภทของแขนกลอุตสาหกรรม

สามารถแบ่งประเภทแขนกลโดยพิจารณาจากพื้นฐานใหญ่ ๆ ดังนี้

1. โครงสร้างภายนอกและการเคลื่อนที่ของแขนกล
2. ทางเดินของแขนกลซึ่งขึ้นอยู่กับความควบคุมการเคลื่อนที่

2.2.1 การแบ่งประเภทของแขนกลโดยอาศัยการเคลื่อนที่และโครงสร้างภายนอก

แขนกลอุตสาหกรรม มีรูปร่างลักษณะแตกต่างกันไปแล้วแต่ลักษณะการใช้งานโดยขีดความสามารถในการเคลื่อนที่ของมือจะแตกต่างกันไปแต่โดยทั่วไปหากจะจัดแบ่งออกเป็นกลุ่ม ๆ แล้วก็คงจะแบ่งออกเป็น 4 กลุ่มคือ

2.2.1.1 แขนกลที่มีการเคลื่อนไหวได้ในระบบพิกัดขั้ว (polar coordinate configuration)

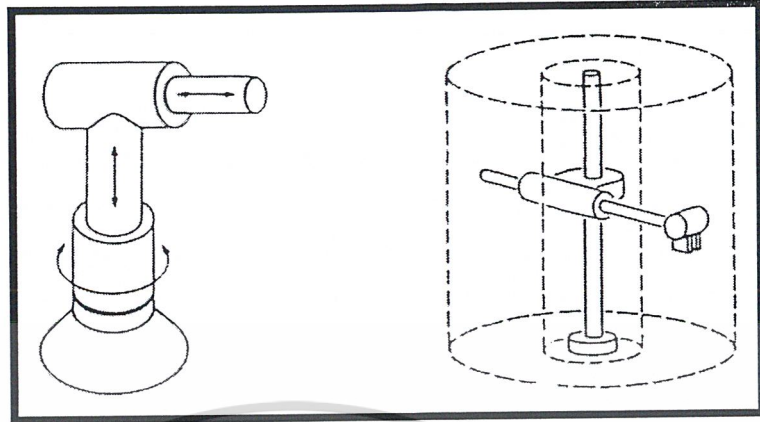
ขอบเขตการเคลื่อนไหวของแขนกลที่มีลักษณะเช่นนี้ อาจเรียกอีกอย่างหนึ่งว่าอยู่ใน ระบบพิกัดเชิงกลม (spherical coordinate configuration) ทั้งนี้เพราะลักษณะบริเวณที่แขนกลจะกวาดไปได้จะมีลักษณะเป็นรูปวงกลมดังรูปที่ 2.3



รูปที่ 2.3 แสดงแขนกลที่มีการเคลื่อนไหวในระบบพิกัดเชิงขั้ว

2.2.1.2 แขนกลที่มีเคลื่อนไหวได้ในระบบพิกัดเชิงขั้ว (cylindrical coordinate configuration)

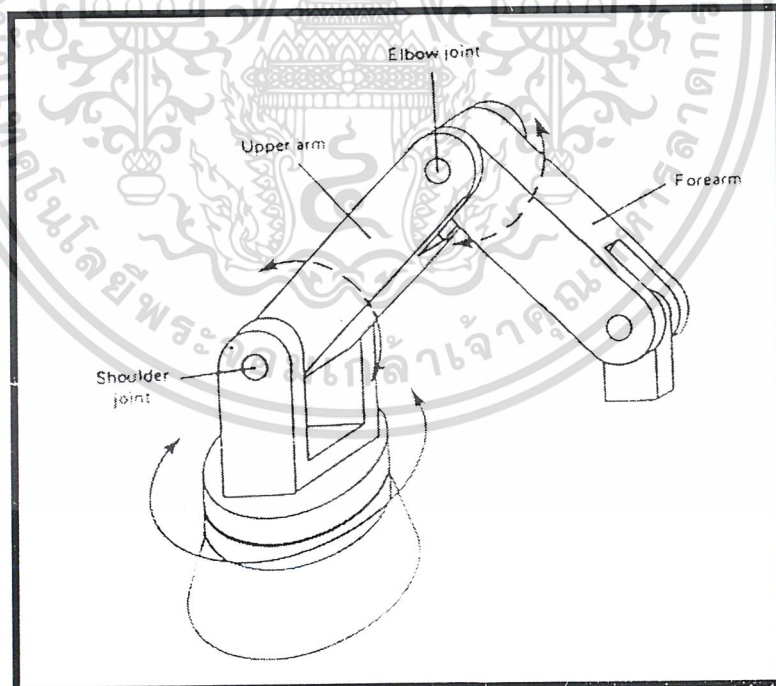
ขอบเขตของหุ่นยนต์ลักษณะนี้มีลักษณะการกวาดแขนเป็นรูปทรงกระบอกดังรูปที่ 2.4



รูปที่ 2.4 แสดงแขนกลที่มีการเคลื่อนไหวในระบบพิกัดทรงกระบอก

2.2.1.3 แขนกลที่มีลักษณะแขนเป็นข้อต่อ (joint arm configuration)

แขนกลประเภทนี้จะเลียนแบบลักษณะแขนของมนุษย์โดยส่วนต่าง ๆ สามารถยืดหดได้เช่นเดียวกับส่วนไหล่ ศอก แขน ข้อศอก และข้อมือ ทำให้การเคลื่อนที่คล่องตัวโดยจะมีรัศมีที่กวาดได้เป็นลักษณะรูปครึ่งทรงกลมดังรูปที่ 2.5

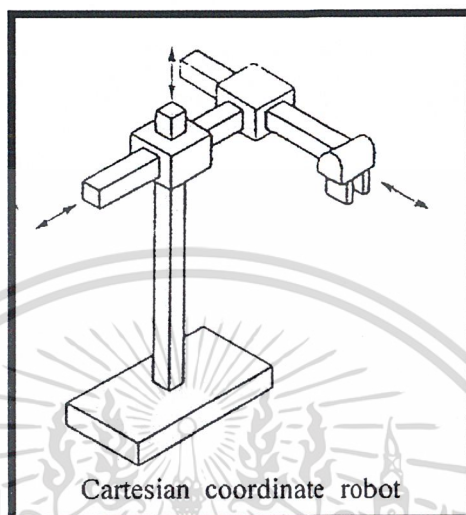


รูปที่ 2.5 แสดงแขนกลที่มีการเคลื่อนไหวในลักษณะข้อต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.4 แขนกลที่มีการเคลื่อนไหวได้ในระบบพิกัดคาร์ทีเซียน(cartesian coordinate robot)

แขนกลประเภทนี้จะเคลื่อนที่ได้ใน 3 แกนหลักคือ แกน x, แกน y, แกน z ของระบบพิกัดคาร์ทีเซียน ดังนั้นขอบเขตรัศมีการกวาดของมือจะเป็นรูปกรอบสี่เหลี่ยมดังรูปที่ 2.6



รูปที่ 2.6 แสดงแขนกลที่มีการเคลื่อนไหวในระบบพิกัดคาร์ทีเซียน

ทั้งนี้ไม่ว่าจะเป็นแขนกลในลักษณะใด งานหลักของแขนกลก็คือการจัดการกับเป้าหมายโดยจะใช้ส่วนที่อยู่ปลายสุดของแขนเป็นตัวจัดการ ส่วนนี้จะทำหน้าที่คล้ายมือของมนุษย์มีชื่อเสียเฉพาะว่า(end effector) ดังนั้นในการนำไปใช้งานจะต้องคำนึงความสามารถของส่วนนี้เป็นหลัก โดยปกติแล้วการเคลื่อนที่ของส่วนมือหรือ end effector นี้จะทำได้ใน 6 ระดับขั้นความเสรี(degree of freedom) ซึ่งจะทำให้คล้ายการทำงานของมือมนุษย์

2.2.2 การแบ่งประเภทของแขนกลโดยอาศัยทางเดินของแขนกล

การควบคุมทางเดินของแขนกลแบ่งเป็น 2 แบบใหญ่ ๆ คือ

1. การควบคุมแบบจุด(Point-to-point(PTP) control) แขนกลจะได้รับคำสั่งโปรแกรมให้หยุดที่จุดใดจุดหนึ่งและใช้จุดนั้นในการกำหนดจุดที่ต้องการให้หยุดจุดต่อไป แม้ว่าไม่สามารถควบคุมการเคลื่อนที่ระหว่างจุดให้อยู่ในระหว่างเส้นทางที่ต้องการ แต่การเคลื่อนที่ไปยังจุดที่กำหนดก็ยังคงต้องตามที่กำหนด ในทางตรงข้ามที่ควบคุมแบบต่อเนื่อง จะสามารถเคลื่อนที่ตามเส้นทางต่าง ๆ ได้อย่างถูกต้อง

2. การควบคุมแบบต่อเนื่อง(Continuous path(CP) control)

การควบคุมแบบจุดจะมีความแม่นยำสูงมากทั้งในแง่ความเที่ยงตรง และการทำงานช้า ๆ แม้ว่าการควบคุมแบบต่อเนื่องจะเคลื่อนไหวได้นุ่มนวลมากกว่าในเส้นทางที่ได้ระบุไว้ แต่ก็มีข้อเสียในด้านความเร็ว ซึ่งเป็นฟังก์ชันของจังหวะการเคลื่อนที่ที่คำนวณ โดยคอมพิวเตอร์ในลักษณะเวลาจริง(real time) ข้อเสียนี้จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้ความเร็วของการเคลื่อนที่แบบต่อเนื่องลดลงถึง 15-25% และประสิทธิภาพการทำงานจะลดลงเมื่อเปรียบเทียบกับแขนกลตัวเดียวกันที่ควบคุมแบบจุด

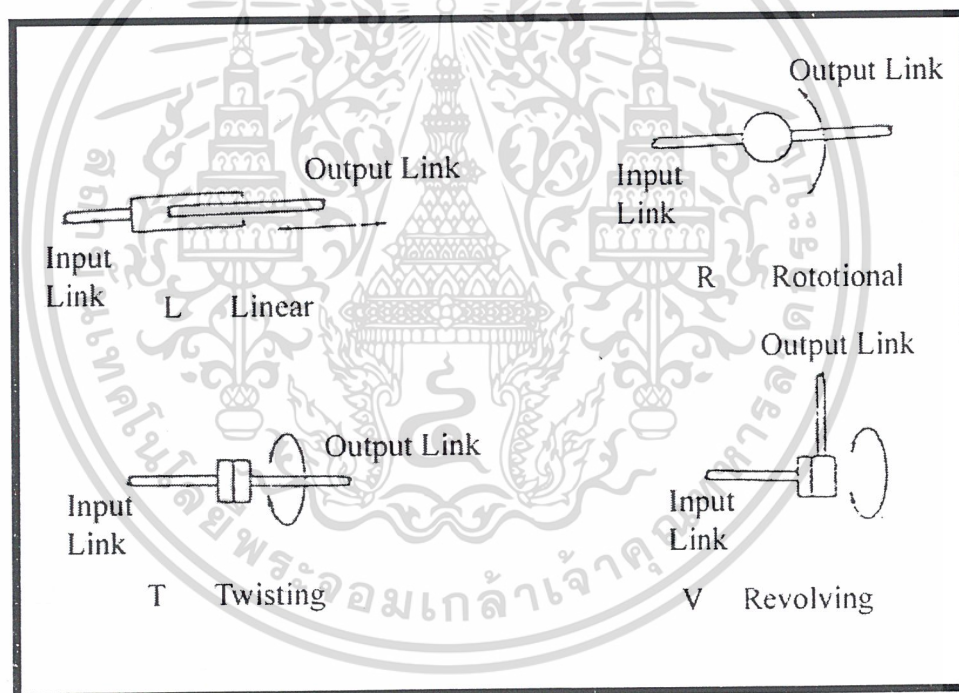
2.3 การเคลื่อนที่ของแขนกล

การเคลื่อนที่ของแขนกลสามารถแบ่งการเคลื่อนที่ออกได้ 2 ส่วนคือ

2.3.1 การเคลื่อนที่ของแขน

สามารถแบ่งตามลักษณะของข้อหมุนได้ดังนี้คือ

1. การเคลื่อนที่แบบเส้นตรง(Linear)
2. การเคลื่อนที่แบบหมุนรอบจุดหมุน(Rotational)
3. การเคลื่อนที่แบบบิดรอบจุดหมุน(Twisting)
4. การเคลื่อนที่แบบหมุนตั้งฉาก(Revolving)

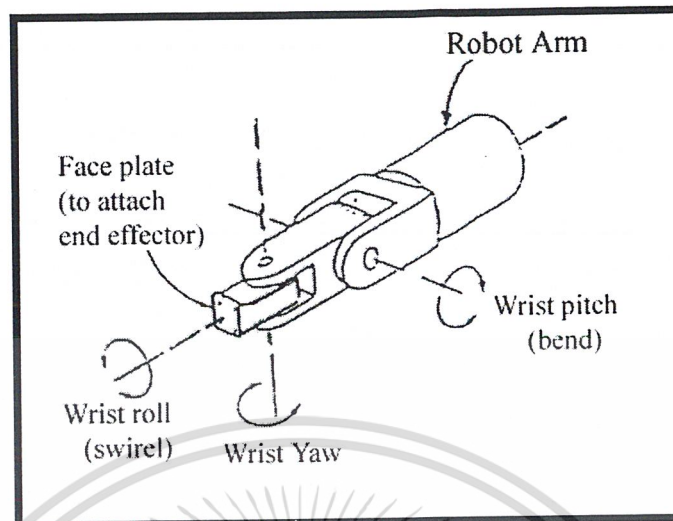


รูปที่ 2.7 แสดงลักษณะการเคลื่อนที่ของแขนกลแบบต่างๆ

2.3.2 การเคลื่อนที่ของมือ

1. หมุน(Roll) บางครั้งเรียกว่า Swival ข้อมือหมุนรอบแกนของแขน
2. บิด(Pitch) บางครั้งเรียกว่า Bend ข้อมือยกขึ้นลงในแนวตั้ง
3. ต่าย(Yaw) หมายถึงบิดไปมาทางซ้ายและขวาของแกนมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงการเคลื่อนที่ของมือ

2.4 End Effector

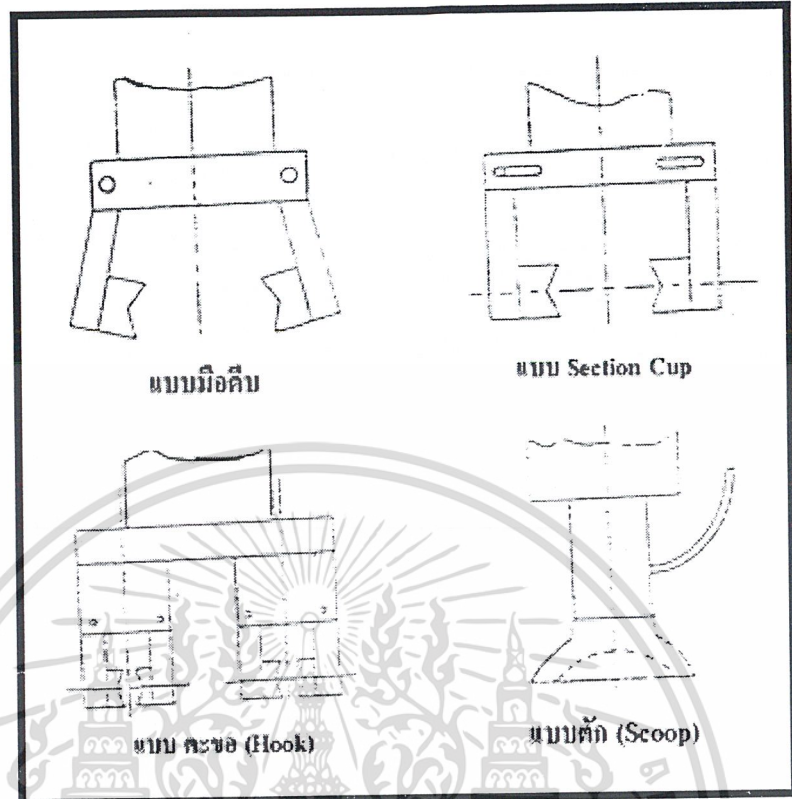
End Effector หรือ “มือ” ของแขนกลนี้เป็นอุปกรณ์ที่แขนกลใช้ทำงานจริง(ส่วนแขนนั้นใช้เพื่อเคลื่อนตำแหน่ง “มือ” นี้ให้ไปอยู่ในจุดที่ต้องการเท่านั้น) งานที่ส่วนนี้ทำก็มีอาทิเช่น การจับชิ้นส่วน การบัดกรีตามจุด การพ่นสี เป็นต้น ดังนั้นลักษณะของส่วนนี้จึงต่างกันออกไป สุดแต่แต่เจตนาการของผู้ออกแบบ ซึ่งต้องออกแบบให้เหมาะกับงานแต่ถ้าจะมีการแบ่งกันเป็นชนิดแล้ว “End Effector” นี้จะออกได้เป็นสองกลุ่มใหญ่ ๆ คือ แบบมือจับ(Gripper) แบบมือที่เป็นอุปกรณ์(Tool as End Effector)

2.4.1 แบบมือจับ

เป็นอุปกรณ์ที่ใช้จับชิ้นส่วนหรือเครื่องมือ โดยการออกแบบมีได้หลายลักษณะทั้งนี้ขึ้นอยู่กับการใช้งาน เช่น

1. แบบมือคีบ(อาศัยแรงเสียดทานในการจับ) ใช้งานได้กับวัสดุทุกประเภทแต่หากเป็นวัสดุบอบบางอาจต้องมีอุปกรณ์อื่นช่วยเสริม
- 2.แบบ Section Cup อาศัยคุณสมบัติการทำให้เกิดสภาพสุญญากาศเป็นตัวดึงชิ้นงานใช้ได้กับวัตถุผิวเรียบเท่านั้น
3. แบบตะขอ(Hook) สำหรับเกี่ยวของ เช่น ของที่อยู่บนสายพาน เป็นต้น
4. แบบตัก (Scoop) สำหรับการลำเลียงของที่เป็นของเหลวหรือฝุ่นผงตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงลักษณะของมือจับแบบต่าง ๆ

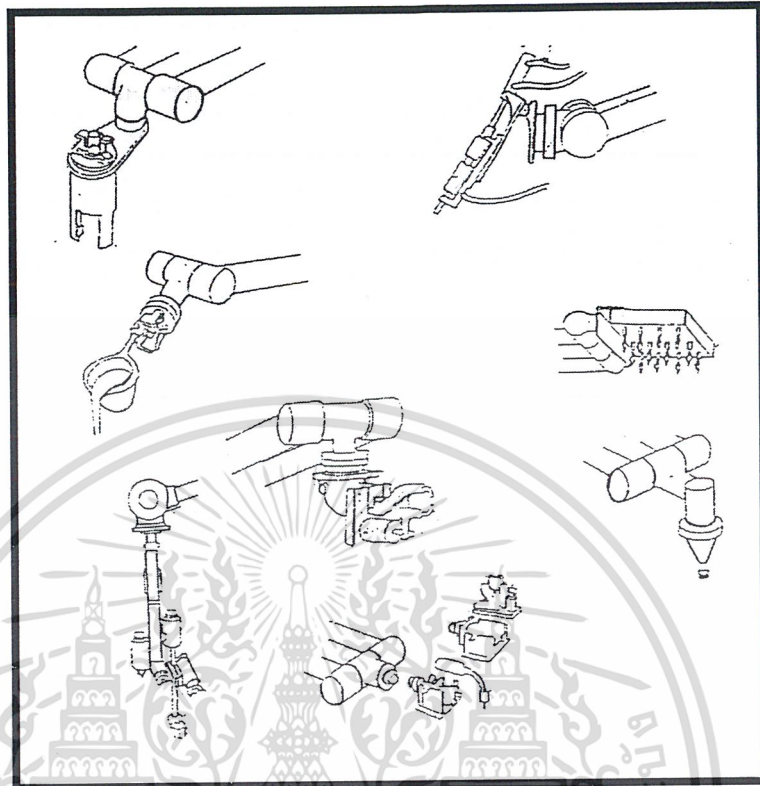
2.4.2 มือที่เป็นอุปกรณ์ (Tool as End Effector)

เนื่องจากแบบมือจับ มีข้อจำกัดใช้ได้เฉพาะกับงานหยิบวัสดุเท่านั้น แต่การใช้งานของแขนกลมีขอบเขตกว้างขวาง จึงได้มีการออกแบบอุปกรณ์ใช้งานติดเข้าไปในส่วนปลายหรือส่วนมือนี้ เพื่อใช้ในการทำงาน โดยอาจจะเป็นอุปกรณ์ที่ติดตายหรืออาจถอดเปลี่ยนได้ ก็แล้วแต่การออกแบบ ตัวอย่างของมือที่เป็นอุปกรณ์นี้มีมากมาย อาทิเช่น

1. ตัดอุปกรณ์เชื่อมเฉพาะจุด (Spot Welding Gun)
2. ตัดอุปกรณ์พ่นสี (Spray Paint Gun)
3. ตัดอุปกรณ์สว่าน (Drilling Spindle)

เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แสดงลักษณะของมือที่เป็นอุปกรณ์แบบต่าง ๆ

2.5 การสั่งงาน

การสั่งงานให้แขนกลทำงานตามที่กำหนดนี้ อาจทำได้หลายลักษณะ คือ

- 1.แบบ Manual
- 2.แบบ Lead Through
- 3.แบบการสั่งงานด้วยโปรแกรม

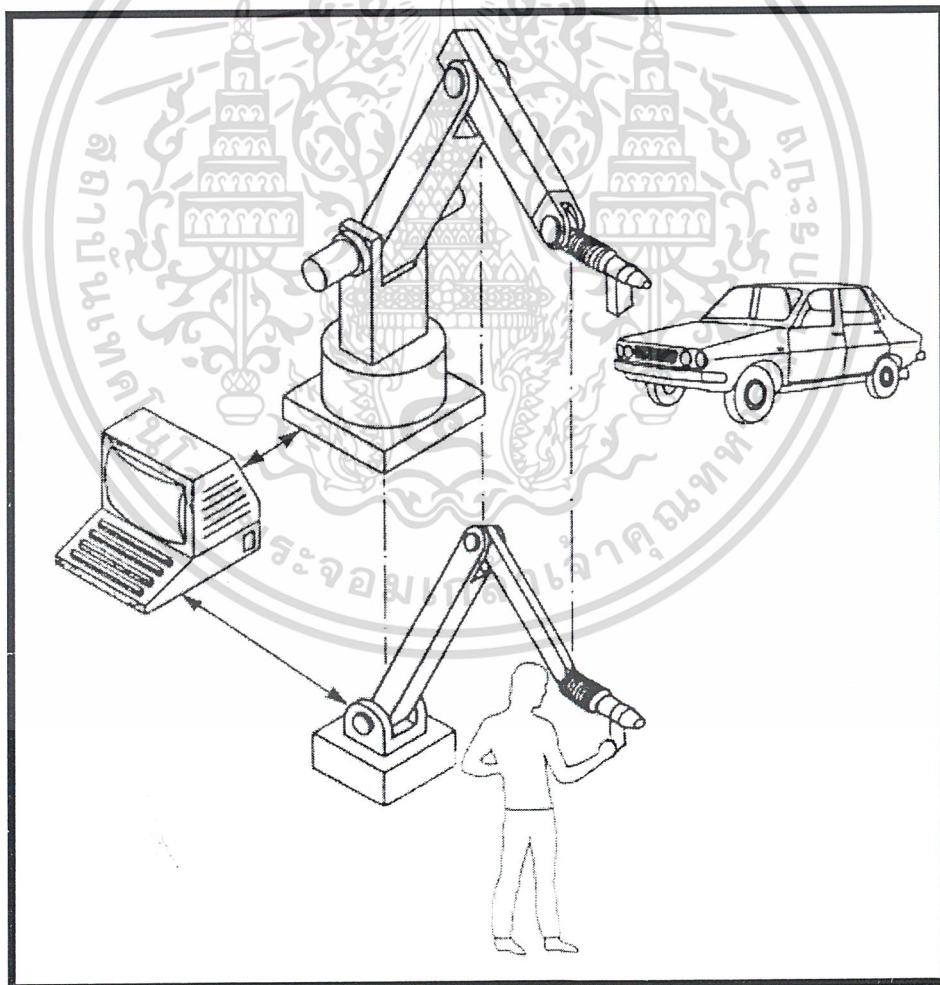
2.5.1 การสั่งงานแบบ Manual

การสั่งงานแบบนี้เป็นการสั่งงานที่ง่ายที่สุด และใช้กันมากในการสั่งงานให้แขนกล ทำงานแบบ จากจุดหนึ่งไปจุดหนึ่ง(Point - to - Point) โดยวิธีการจะเป็นการเคลื่อนแกนต่าง ๆ ไปยังตำแหน่งที่ต้องการ โดยการกดปุ่มบังคับ ดังนั้นลักษณะนี้จึงคล้ายกับการ Set up มากกว่าการโปรแกรม กล่าวคือมีการเคลื่อน ไปยังจุดเริ่มต้นแล้วก็กำหนดค่าไว้ จากนั้นก็เคลื่อนไปยังตำแหน่งต่อไป แล้วกำหนดค่าไว้ เป็นต้น โดย ค่าที่กำหนดไว้นี้จะเก็บไว้ในหน่วยความจำเรียกออกมาใช้ได้ภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 การสั่งงานแบบ Lead Through

การสั่งงานแบบนี้คล้าย ๆ กับการจับมือแขนกลให้เคลื่อนที่ไปตามแนวที่ต้องการ ในลักษณะที่เป็นการเคลื่อนที่แบบต่อเนื่องแล้วจึงเก็บค่าตำแหน่งต่าง ๆ ที่เคลื่อนที่ไป พร้อมกับความเร็วที่เคลื่อนที่แต่ในทางปฏิบัติไม่สามารถทำได้ เนื่องจากข้อจำกัดในแง่เทคนิค(อาทิเช่น ระบบเกียร์ เป็นต้น) ดังนั้นจึงมีผู้คิดค้นอุปกรณ์ซึ่งเลียนแบบแขนกลโดยสามารถเคลื่อนที่ไปยังตำแหน่งต่าง ๆ ได้ดังรูปที่ 2.11 อุปกรณ์นี้มีชื่อเรียกว่า Robot Simulator ซึ่งการทำงานก็ใช้หลักการข้างต้นกล่าวคือ ผู้สั่งงานจะเลื่อนไปยังตำแหน่งที่ต้องการ โดยมีแนวทางการเคลื่อนที่ที่กำหนดได้โดยตำแหน่งต่าง ๆ จะถูกเก็บไว้ในหน่วยความจำของคอมพิวเตอร์ ซึ่งพบว่าวิธีนี้เป็นการโปรแกรมโดยตรงวิธีหนึ่ง ซึ่งข้อเสียก็มีบ้างอาทิเช่น ความละเอียดไม่ดี ไม่สามารถกำหนดความเร็วได้ ต้องมีการซื้อ Simulator มาเพิ่มเข้ากับระบบซึ่งเป็นการสิ้นเปลืองและหากการสั่งการเกิดผิดพลาดอาจต้องทำใหม่ทั้งหมด เป็นต้น การสั่งงานชนิดนี้ใช้มากในงานด้านการพ่นสี ชิ้นส่วนรถยนต์ เป็นต้น



รูปที่ 2.11 แสดงลักษณะการสั่งงานแบบ Lead Through โดยอาศัย Robot Simulator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 การสั่งงานด้วยโปรแกรม

การสั่งงานให้แขนกลอุตสาหกรรมทำงานโดยการเรียกใช้โปรแกรมสั่งงานนี้ จะเป็นการให้ข้อมูลกับแขนกลเพื่อให้แขนกลเคลื่อนที่ไปยังตำแหน่งที่ต้องการและทำงานตามที่กำหนด โดยในปัจจุบัน ได้มีผู้เขียนภาษาคอมพิวเตอร์เพื่อสั่งให้หุ่นยนต์ทำงานมากมายหลายภาษา โดยโปรแกรมจะถูกเขียนบนเครื่องคอมพิวเตอร์แล้วส่งไปควบคุมแขนกลอีกต่อหนึ่ง

2.5.4 การประยุกต์การใช้งาน

ในการใช้แขนกลช่วยในงานอุตสาหกรรมนี้เนื่องจากราคาค่อนข้างแพง ดังนั้นการพิจารณาตัดสินใจใช้แขนกลเหล่านี้จึงต้องพิจารณาเฉพาะงานที่จำเป็น เช่น

1. งานที่อันตราย หรืองานที่มนุษย์ทำไม่สะดวก เช่น บริเวณที่มีรังสี สารพิษ เป็นต้น
2. งานที่ต้องทำซ้ำบ่อย ๆ ซึ่งหากมนุษย์ทำอาจผิดพลาดได้ง่าย เพราะเป็นงานที่น่าเบื่อ
3. ใช้ในงานที่ต้องการความสามารถเฉพาะ เช่น งานยกของหนัก งานต้องการความละเอียด

เป็นต้น

4. ใช้ในงานที่ต้องทำต่อเนื่อง

เป็นต้น

โดยลักษณะของงานที่ใช้มากที่สุดอาทิเช่น

1. งานหยิบจับชิ้นวัสดุ
2. งานเชื่อม ไม่ว่าจะเป็นแบบเชื่อมจุด (spot welding) หรือเชื่อมแบบต่อเนื่อง (arc welding)
3. การพันสารซึ่งอาจมีกลิ่น มีเสียงรบกวน เป็นอันตรายต่อสุขภาพ
4. การประกอบชิ้นงาน
5. งานตรวจเช็ค (Inspection)

ข้อดีของการใช้แขนกลก็มีมากมายอาทิเช่น

1. ความปลอดภัย
2. ความสม่ำเสมอของงานดี เพราะเมื่อโปรแกรมแล้วก็จะทำตามขั้นตอนด้วยความเร็วที่กำหนด
3. ใช้พลังงานน้อย เพราะการใช้แขนกลอาจไม่ต้องสนใจระบบถ่ายเทอากาศ เครื่องปรับอากาศก็

ไม่จำเป็น

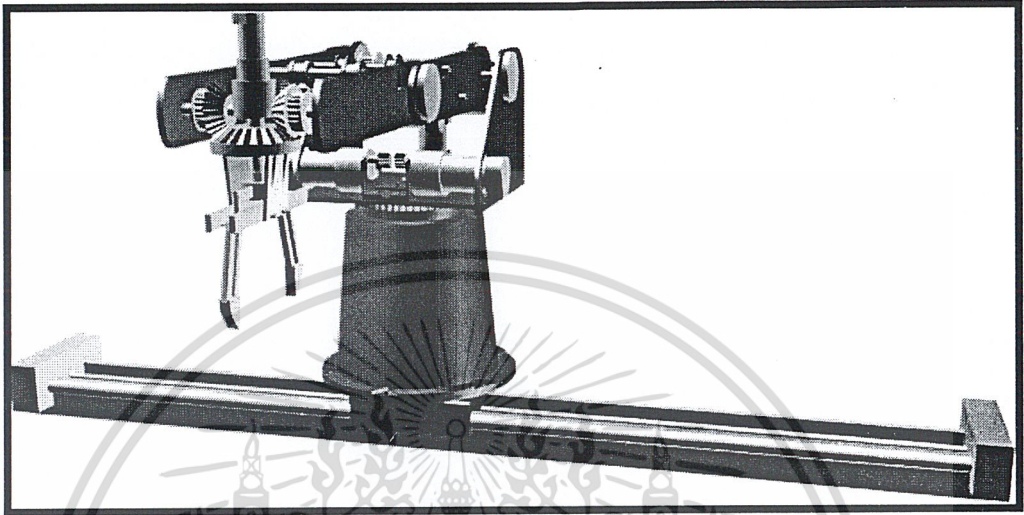
4. เพิ่มประสิทธิภาพการทำงาน เช่น งานเสร็จเร็วกว่าใช้แรงงานคน
5. ใช้วัสดุน้อย เช่นงานพันสี จะทำอย่างคงเส้นคงวา ทำให้ประหยัดสี

เป็นต้น

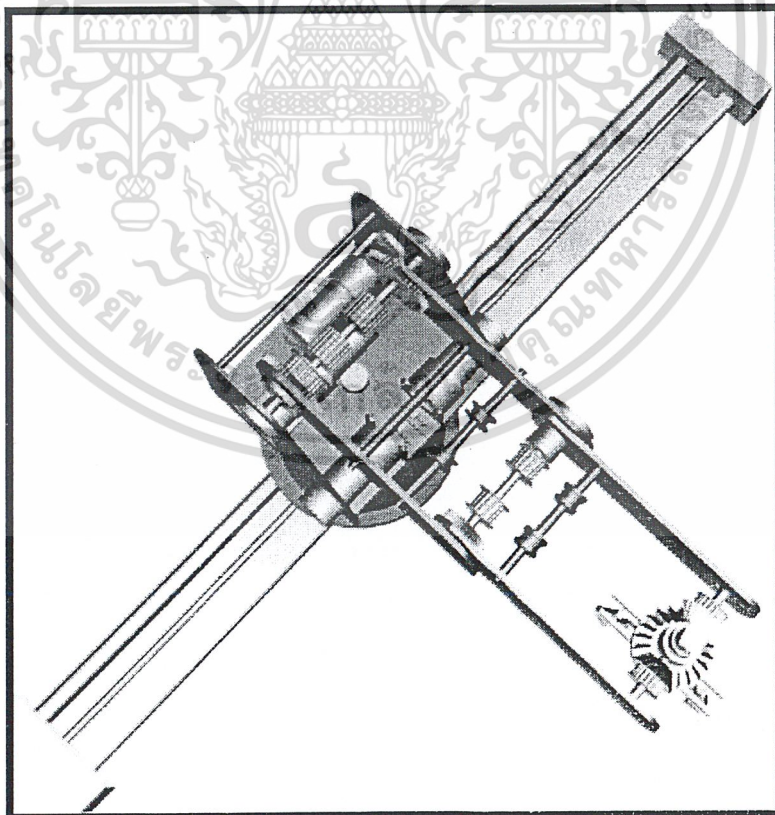
แต่ขอบเขตในการประยุกต์ใช้งานจริงแล้วขึ้นอยู่กับจินตนาการของผู้นำไปใช้งานมากกว่า

2.6 ลักษณะการทำงานของแขนกล Scorbot ER3

เนื่องจากโครงการนี้ได้นำแขนกล Scorbot ER3 มาใช้งานลักษณะต่าง ๆ ของแขนกล Scorbot ER3 สามารถดูได้จากรูปข้างล่าง

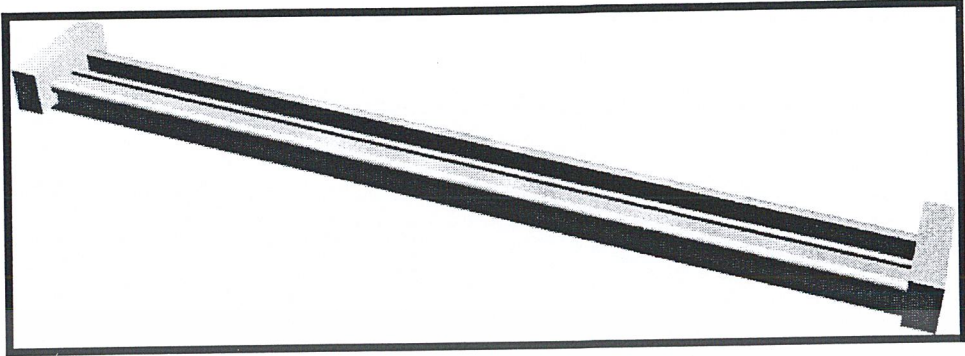


รูปที่ 2.12 แสดงลักษณะของแขนกล Scorbot ER3



รูปที่ 2.12(ต่อ)แสดงลักษณะของแขนกล Scorbot ER3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

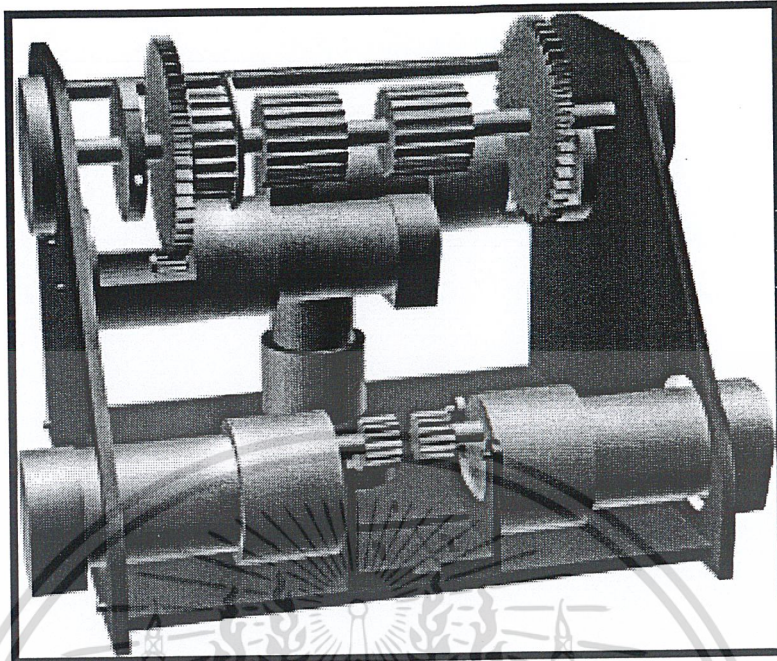


รูปที่ 2.13 แสดงลักษณะ link 1

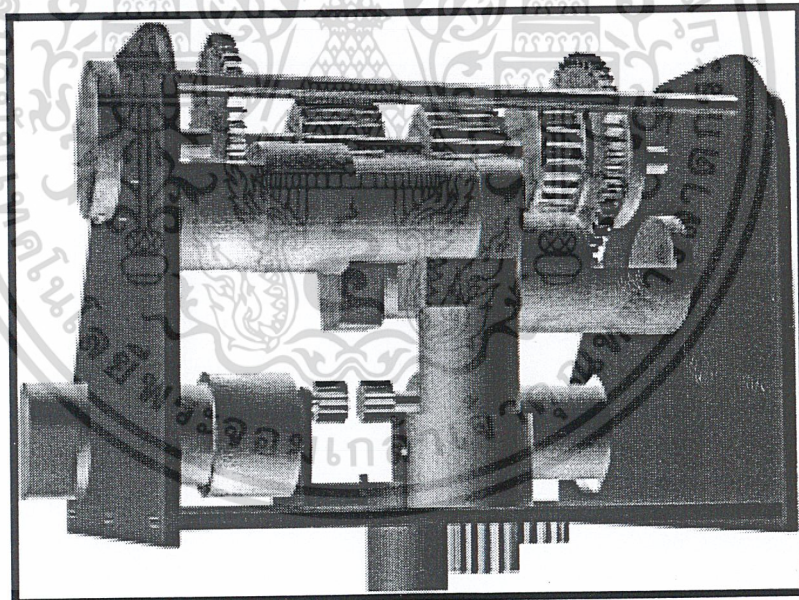


รูปที่ 2.14 แสดงลักษณะ link 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

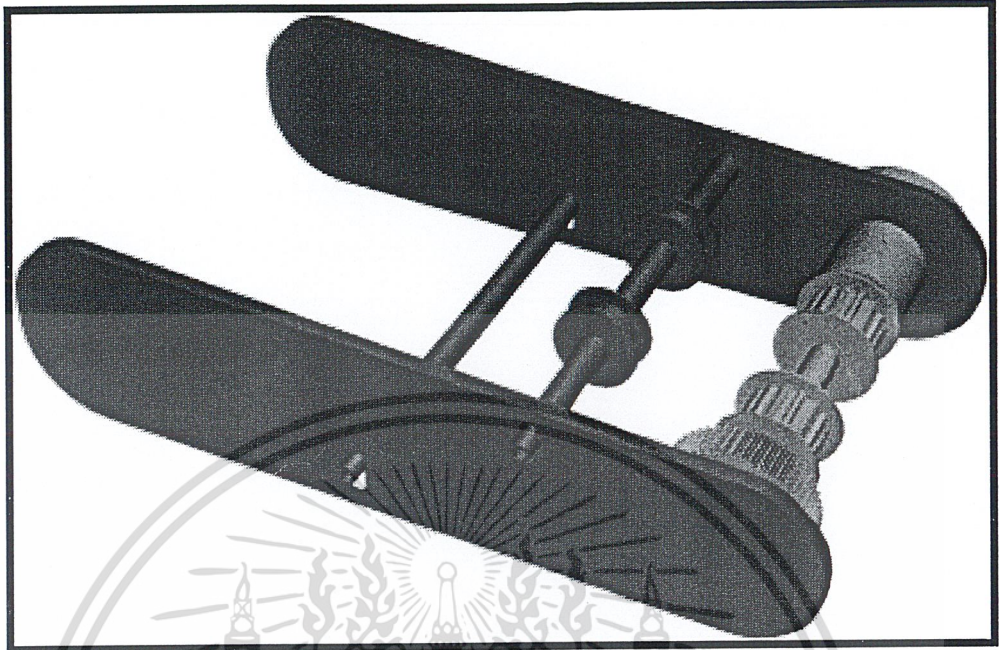


รูปที่ 2.15 แสดงลักษณะ link 3

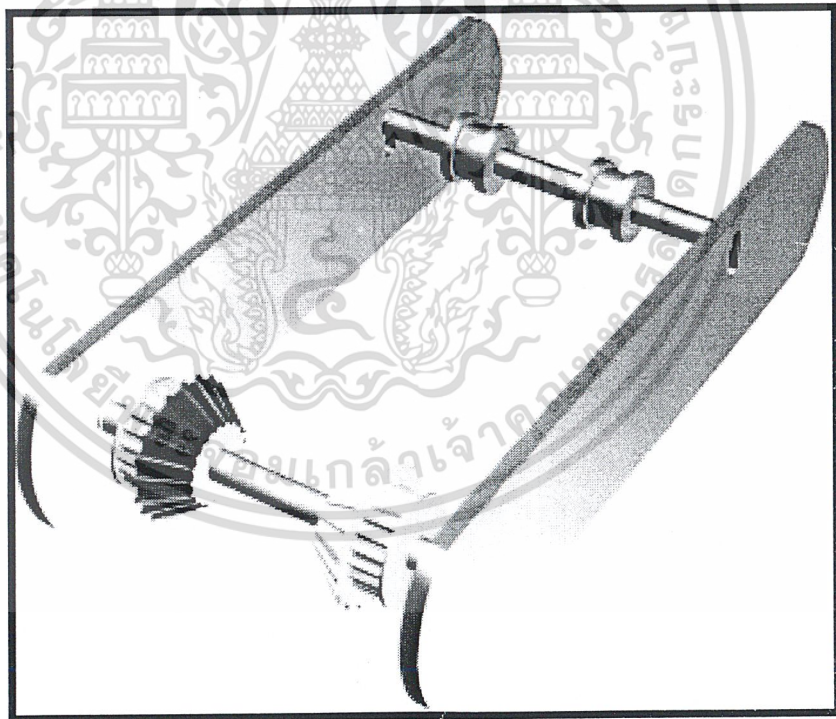


รูปที่ 2.15(ต่อ) แสดงลักษณะ link 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

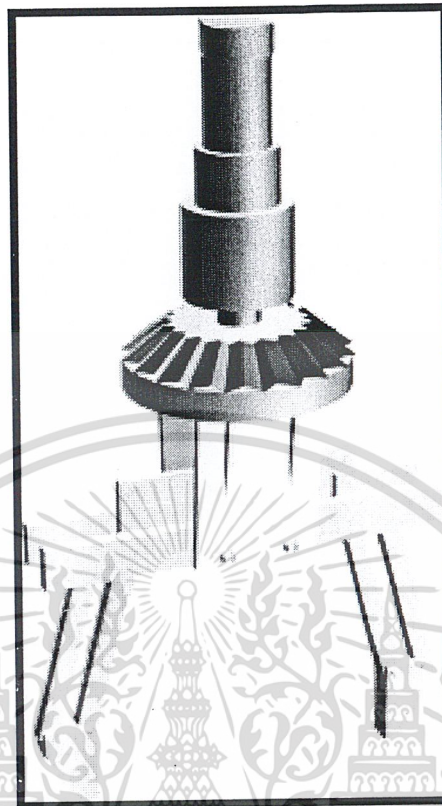


รูปที่ 2.16 แสดงลักษณะ link 4

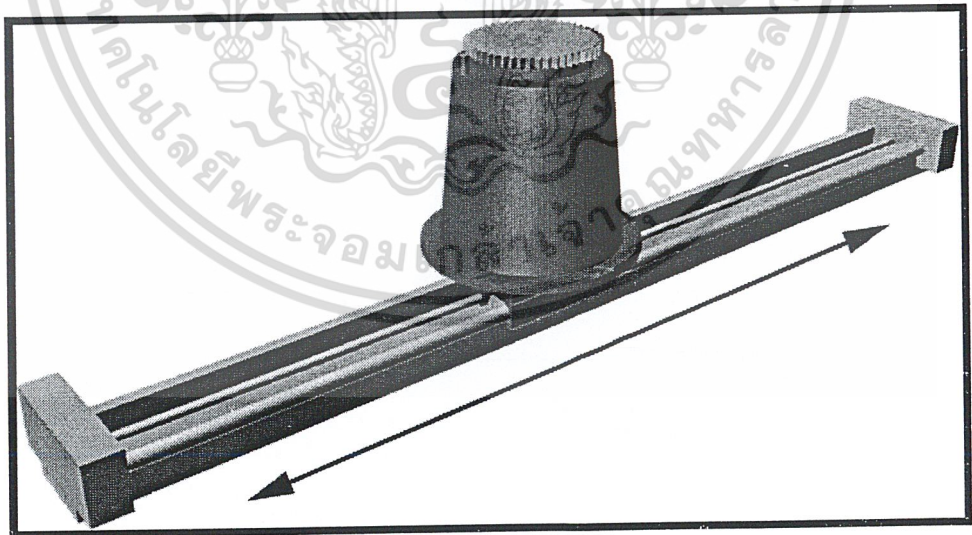


รูปที่ 2.17 แสดงลักษณะ link 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

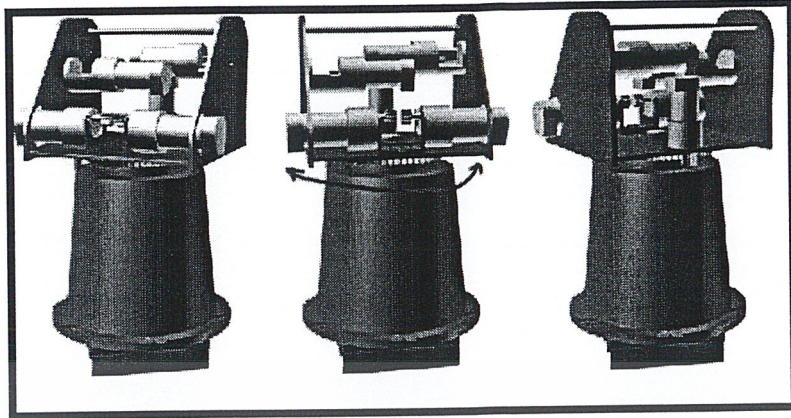


รูปที่ 2.18 แสดงลักษณะ link 6

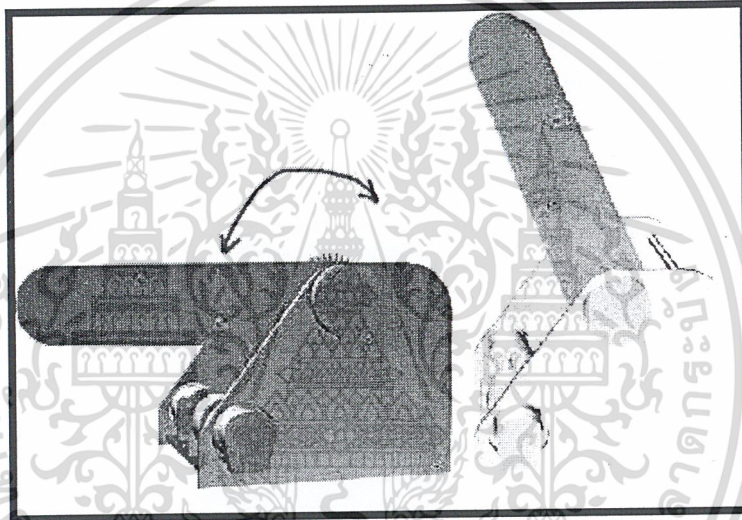


รูปที่ 2.19 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง link 1 กับ link 2

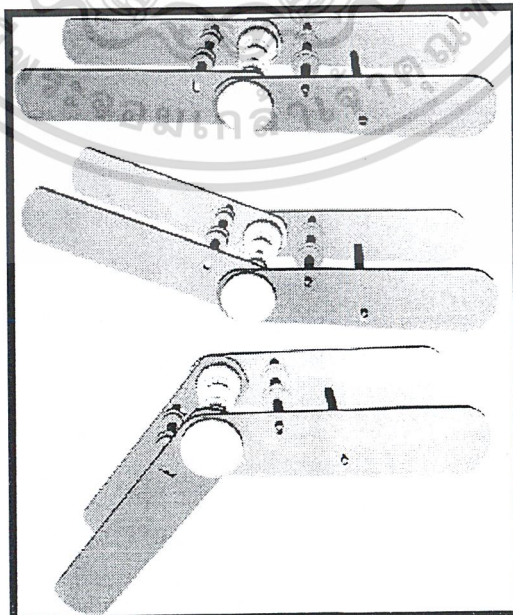
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง link 2 กับ link 3

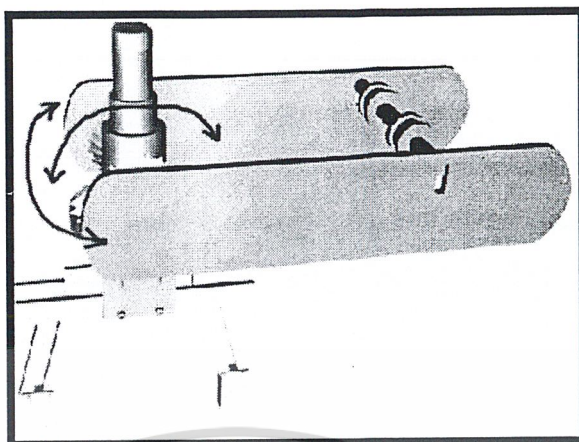


รูปที่ 2.21 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง link 3 กับ link 4

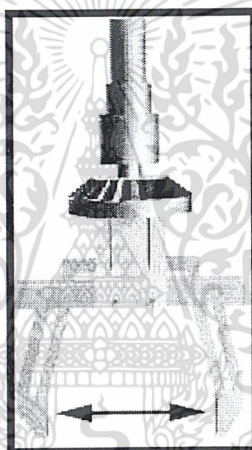


รูปที่ 2.22 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง link 4 กับ link 5

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 แสดงความสัมพันธ์การเคลื่อนที่ระหว่าง link 5 กับ link 6



รูปที่ 2.24 แสดงการทำงานของ link 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ทฤษฎีและหลักการ

เนื่องจากแขนกล Scorbot ER3 ใช้มอเตอร์กระแสตรงในการหมุนข้อต่อต่าง ๆ และมีเอนโคเดอร์ที่ทำหน้าที่ส่งสัญญาณรอบการหมุนของมอเตอร์ นอกจากนี้ยังมีระบบการควบคุมจากคอมพิวเตอร์ ดังนั้นจำเป็นต้องศึกษาการทำงานขององค์ประกอบต่าง ๆ เพื่อที่สามารถนำทฤษฎีเหล่านั้นมาใช้ควบคุมการทำงานของแขนกล Scorbot ER3 ได้

ดังนั้นในบทที่นี้จะกล่าวถึงทฤษฎี การควบคุมความเร็วของมอเตอร์ การทำงานของเอนโคเดอร์ และการทำงานของไอซี 8255 ที่ใช้ในการ์ด ET-PC 8255

3.1 โครงสร้างและชนิดของมอเตอร์ไฟฟ้ากระแสตรง

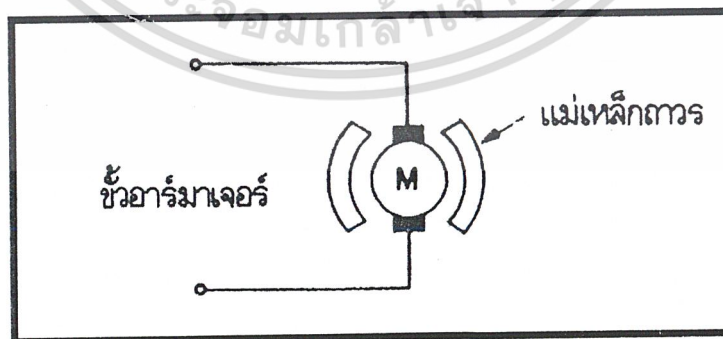
มอเตอร์ไฟฟ้ากระแสตรงประกอบด้วยส่วนต่าง ๆ ดังนี้คือ

1. ขั้วสนามแม่เหล็ก (Field Poles)
2. แกนเหล็กอาร์มีเจอร์ (Armature Core)
3. คอมมิวเตเตอร์ (Commutator)
4. แปรงถ่าน (Brush)

มอเตอร์ไฟฟ้ากระแสตรงถูกใช้อย่างกว้างขวางกับงานที่ต้องการความเร็วที่ปรับเปลี่ยนได้ (Adjustable Speed), การรักษาระดับความเร็ว (Speed Regulation) และงานที่ต้องเริ่มต้น (Starting) หรือหยุด (Braking) และกลับทิศทาง (Reversing) บ่อยครั้ง มอเตอร์ไฟฟ้ากระแสตรงที่ถูกใช้โดยทั่วไปแบ่งได้เป็น 2 ชนิดตามลักษณะการจ่ายสนามแม่เหล็ก คือ

1. มอเตอร์ไฟฟ้ากระแสตรงแบบปรับเส้นแรงแม่เหล็กได้
2. มอเตอร์ไฟฟ้ากระแสตรงแบบเส้นแรงแม่เหล็กมีค่าคงที่

ซึ่งในแขนกล Scorbot ER3 นี้ใช้มอเตอร์ไฟฟ้ากระแสตรงแบบเส้นแรงแม่เหล็กมีค่าคงที่ ดังรูปที่ 2.1



รูปที่ 3.1 แสดงมอเตอร์ไฟฟ้ากระแสตรงแบบเส้นแรงแม่เหล็กมีค่าคงที่

3.1.1 มอเตอร์ไฟฟ้ากระแสตรงแบบเส้นแรงแม่เหล็กมีค่าคงที่

ระบบการกระตุ้นฟิวด์ของมอเตอร์โดยทั่วไปในปัจจุบันมักใช้เป็นแบบ แม่เหล็กถาวร ดังรูปที่ 3.1 ในระบบนี้เส้นแรงของฟิวด์มีค่าคงที่ดังนั้น อัตราส่วนระหว่างกระแสอาร์มาเจอร์และแรงบิดจะมีค่าคงที่

$$T = K\phi I \quad (3.1)$$

$$E = K_c\omega \quad (3.2)$$

T = แรงบิดของเพลา มีหน่วยเป็นนิวตัน-เมตร

ϕ = เส้นแรงแม่เหล็ก มีหน่วยเป็นเวเบอร์

I = กระแสเป็นแอมแปร์

E = โวลต์เตจย้อนกลับ emf มีหน่วยเป็นโวลต์

ω = ความเร็วของมอเตอร์ มีหน่วยเป็นเรเดียน/วินาที

K_c = ค่าคงที่ของ โวลต์เตจย้อนกลับ

K = ค่าคงที่

ระบบนี้จะให้ความสัมพันธ์ระหว่าง กระแสอาร์มาเจอร์ แรงบิดและความเร็วอยู่ในลักษณะลิเนียร์ สมการทางไฟฟ้าของมอเตอร์ไฟฟ้ากระแสตรงแบบนี้ได้จากสมการ(3.3)

$$V = K_c\omega + L\frac{di}{dt} + Ri \quad (3.3)$$

เมื่อ

V = โวลต์เตจที่ป้อนให้กับมอเตอร์

L = อีคคักเต้นซ์ของอาร์มาเจอร์

R = ความต้านทานที่ขั้วของมอเตอร์

ข้อดีของมอเตอร์ไฟฟ้ากระแสตรงแบบฟิวด์แม่เหล็กถาวร ซึ่งเหนือกว่ามอเตอร์แบบมีโครงสร้างของฟิวด์ด้วยการพันของขดลวดคือ ไม่มีกำลังสูญเสียในฟิวด์ มีประสิทธิภาพสูงกว่าและมีขนาดเล็กกว่าเมื่อเทียบกับมอเตอร์ที่มีขนาดของกำลังม้าเท่ากันนอกจากนี้ความสัมพันธ์เชิงเส้นในสมการ(3.1) ยังให้ค่าของกระแสอาร์มาเจอร์ที่สูงกว่าดีซีมอเตอร์แบบฟิวด์เป็นขดลวด การประยุกต์ใช้งานเหมาะกับระบบที่ต้องการแรงบิดของโหลดสูง

3.2 รายละเอียดของมอเตอร์ที่ใช้ในแขนกล Scorbot ER3

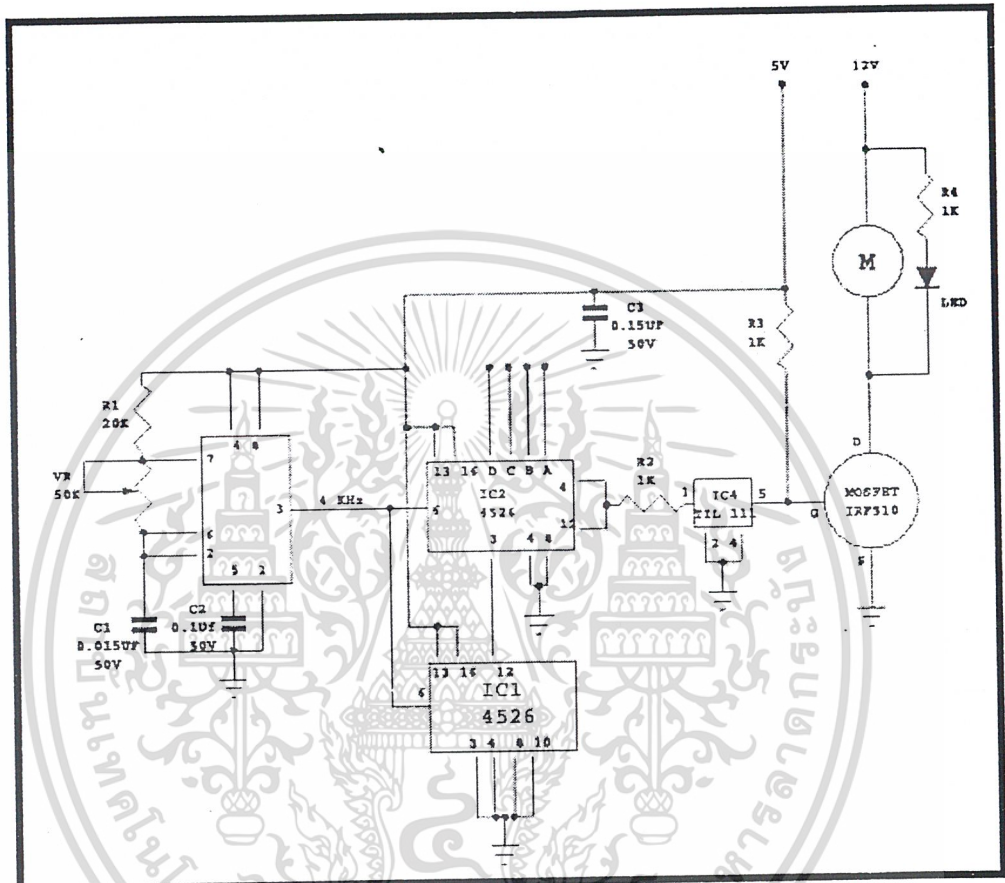
มีดังนี้ คือ

มอเตอร์ไฟฟ้ากระแสตรงมี เอน โคดเดอร์และเกียร์บล็อคมอเตอร์	127.7 : 1	จำนวน 5 ตัว
มอเตอร์ไฟฟ้ากระแสตรงมีเอน โคดเดอร์และเกียร์บล็อคมอเตอร์	19.5 : 1	จำนวน 1 ตัว
แรงดันไฟฟ้ากระแสตรงสูงสุดที่เข้าอาร์มาเจอร์	12	โวลต์(V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การควบคุมความเร็วรอบของมอเตอร์ไฟฟ้ากระแสตรง

การควบคุมความเร็วรอบของมอเตอร์ไฟฟ้ากระแสตรงจะใช้วิธีพัลส์วิดท์โมดูเลเตอร์ (PWM) วงจรที่ควบคุมความเร็วรอบของมอเตอร์ไฟฟ้ากระแสตรงโดยใช้หลักพัลส์วิดท์โมดูเลเตอร์ดูได้จากรูปที่ 3.2



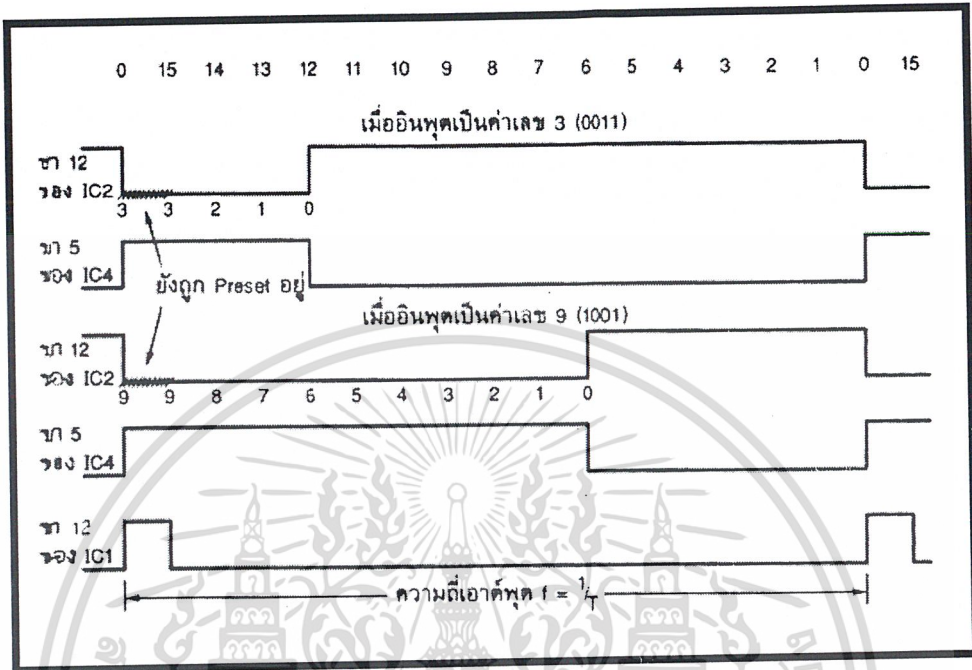
รูปที่ 3.2 แสดงรายละเอียดของวงจรควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง

3.3.1 หลักการทำงานของวงจรควบคุมความเร็วมอเตอร์ไฟฟ้ากระแสตรง

พิจารณาวงจรในรูปที่ 3.2 ส่วนหลักของการทำงานของวงจรมีคือ ไอซีเบอร์ 4526 ทั้งสองตัวซึ่งทำหน้าที่เป็นวงจรนับถอยหลังแบบโปรแกรมได้ เมื่อพิจารณาจากรูปที่ 3.2 จะเห็นว่า ไอซี 1 ถูกต่อให้นับแบบนับวนไปเรื่อย ๆ จาก 15 ถึง 0 เนื่องไม่มีการเซตค่าตั้งต้นให้นับและไม่ได้ถูกรีเซตเลยและเมื่อ ไอซี 1 จะจ่ายเอาต์พุตออกทางขา 3 ของ ไอซี2 (เป็นขา Preset) ซึ่งจะทำให้ ไอซี2 โหลดค่าที่ป้อนเข้ามาทางขา 2 ,14,11,5 และค่านี้จะถูกเก็บไว้เพื่อเป็นค่าตั้งต้นของการนับถอยหลัง นั่นก็หมายความว่าเมื่อ ไอซี1 นับลงมาถึงศูนย์แล้วจะทำให้ ไอซี2 เตรียมพร้อมที่จะนับถอยหลังโดยเริ่มนับจากค่าที่รับเข้ามาทาง 4 บิตอินพุตทั้ง ไอซี1 และ ไอซี2 จะนับลงมาพร้อม ๆ กันเรื่อย ๆ จนกระทั่งเมื่อ ไอซี2 นับลงมาถึงศูนย์มันจะส่งเอาต์พุตออกทางขา 12 และถูกป้อนเข้าขา 4 เพื่อ inhibit ตัวมันเองทำให้ ไอซี2 หยุดการทำงานทันที แต่ตามที่กล่าวแล้วข้างต้น ไอซี1 จะยังคงนับต่อไปจนกระทั่งถึงศูนย์ ก็จะทำให้ ไอซี2 โหลดค่าเข้ามาพร้อมที่จะเริ่มทำงานใหม่อีกครั้ง และก็จะเกิดกระบวนการนี้ซ้ำไปเรื่อย ๆ

เอกสารนี้เป็นเอกสารที่เผยแพร่ไว้ด้วยอำนาจหน้าที่ของข้าพเจ้าในฐานะที่ปฏิบัติงานในตำแหน่งดังกล่าว ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3 เป็นแผนผังเวลา (Timing Diagram) จะบอกลักษณะการทำงานของวงจร ไอซี1 เป็นตัวควบคุมให้ความถี่เอาต์พุตคงที่อยู่ตลอดเวลาและ ไอซี2 เป็นตัวกำหนดค่าควิตซ์ไซเคิลของความถี่เอาต์พุตนั้น



รูปที่ 3.3 แสดงแผนผังเวลาแสดงการทำงานในส่วนพัลส์สวิตช์มอดูเลเตอร์

เมื่อป้อนค่าเลข 0 ทางอินพุตของ ไอซี2 เอาต์พุตที่ขา 12 ของ ไอซี2 จะไม่ได้ค่าในขณะที่ยังมีสัญญาณ Preset เข้าทางขา 3 ซึ่งสัญญาณ Preset นี้จะมีความยาวเท่ากับ 1 คาบของการนับจาก 0 ไปเป็น 15 ดังในรูปที่ 3.3 จึงทำให้เมื่ออินพุตเป็นเลข 0 เราจะได้ค่าควิตซ์ไซเคิลเป็น 15/16 แทนที่จะเป็น 16/16

ไอซี3 เบอร์ 555 ใช้เป็นตัวกำเนิดสัญญาณนาฬิกาจ่ายให้แก่ ไอซี1 และ ไอซี2 ซึ่งถ้าต้องการความถี่ควบคุมมอเตอร์เป็น 250 รอบต่อวินาที(Hz) ก็ต้องปรับ VR ให้ ไอซี 3 ผลิตความถี่ออกมา 4 KHz เนื่องจาก ไอซี1 จะหารความถี่อินพุตด้วย 16 ซึ่งความถี่เอาต์พุตของ ไอซี3 หาได้จากสมการที่ (3.4)

$$f = \frac{1.44}{(R1 + 2VR)C1} \text{ Hz} \quad (3.4)$$

f = ความถี่เอาต์พุต

$R1$ = ค่าความต้าน

VR = ค่าความต้านของตัวทานด้านแบบปรับค่าได้

$C1$ = ค่าของตัวเก็บประจุ

ไอซี 4 เป็น Opto isolator เบอร์ TIL 111 ใช้เพื่อควบคุมการทำงานของมอเตอร์ ซึ่งมอเตอร์ตัวนี้จะเป็นตัวขับเคลื่อนมอเตอร์ เหตุที่ต้องใช้ ไอซี 4 ก็ป้องกันการเสียหายในกรณีที่มอเตอร์เสียลักษณะที่ขาดเรากับเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกิดข้อขัดกัน ความเสียหายจะได้หยุดแค่ ไอซี 4 เท่านั้น และทรานซิสเตอร์ภายใน ไอซี 4 ยังถูกต้องเป็น อินเวอร์เตอร์ เพื่อกลับสัญญาณเอาต์พุตของ ไอซี 2 ก่อนป้อนให้กับมอสเฟต ดังนั้นที่ขา 12 ของ ไอซี 2 และขา 5 ของ ไอซี 4 สัญญาณจะกลับเฟสกันอยู่ เช่นถ้าป้อนค่าความเร็วเป็น 0011 เข้าทาง ไอซี 2 ดังรูปที่ 3.2 ที่ขา 12 ของ ไอซี 2 จะมีค่าควิตซ์ไชเกิ้ลเป็น 12/16 แต่เอาต์พุตขา 5 ของ ไอซี 4 จะได้ค่าควิตซ์ไชเกิ้ลเป็น 4/16 ซึ่งเป็นค่าเดียวกับที่ใช้ขับมอเตอร์นั่นเอง

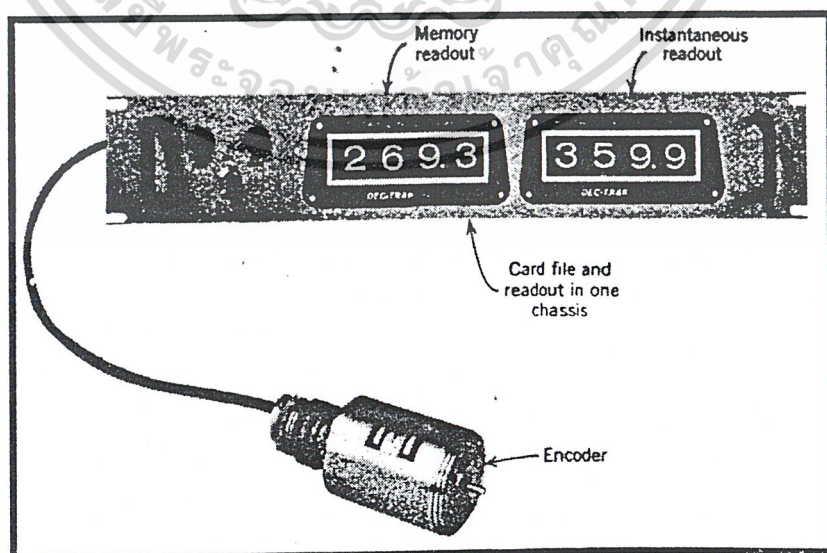
3.4 ตัวจับสัญญาณ (Sensor) และ เอนโคเดอร์ในระบบการควบคุมมอเตอร์

ตัวจับสัญญาณ และ เอนโคเดอร์เป็นส่วนประกอบที่สำคัญในระบบคอนโทรลที่มีการป้อนกลับในระบบคอนโทรลแบบเปิดลูป เอนโคเดอร์มักจะใช้สำหรับตรวจสอบคุณสมบัติการทำงานของระบบ ส่วนในระบบคอนโทรลแบบปิดลูป ตัวจับสัญญาณ และ เอนโคเดอร์ใช้เป็นตัวป้อนสัญญาณกลับเพื่อการคอนโทรล นอกจากนั้นส่วนประกอบเหล่านี้ยังใช้สำหรับพิสูจน์ความเหมือนกันของกระบวนการที่ไม่รู้ค่าหรือกระบวนการที่มีการเปลี่ยนแปลง

ตัวจับสัญญาณ และ เอนโคเดอร์ที่ใช้กันมากในระบบคอนโทรลเช่น โปเทนทิโอมิเตอร์ ออฟดีกอกอลเอนโคเดอร์ แมกนิติกอับทราสควิเซอร์จำพวกแม่เหล็กไฟฟ้า และรีโซลเวอร์

ตัวจับสัญญาณ และ เอนโคเดอร์ที่ใช้แขนกลนี้เป็นแบบ เอนโคเดอร์ที่นับการเพิ่มค่า (Increment Encoder)

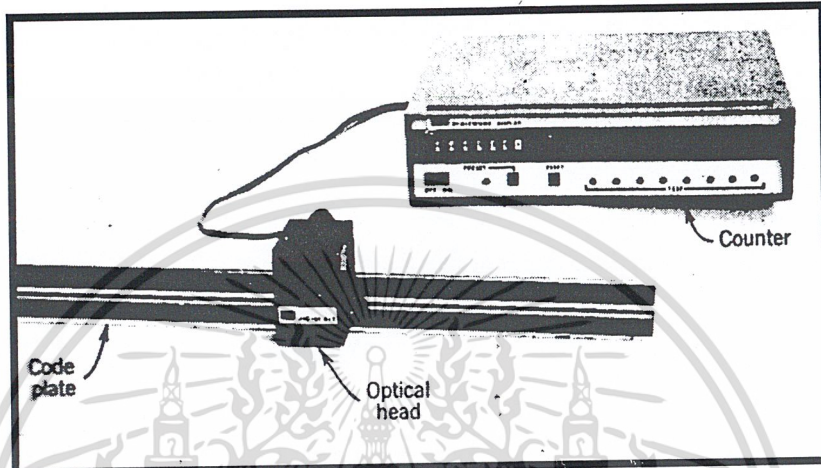
เอนโคเดอร์ที่นับการเพิ่มค่าใช้ในระบบการบังคับตำแหน่งหรือความเร็วของมอเตอร์และสร้างสัญญาณป้อนกลับโดยที่ตัว เอนโคเดอร์ จะสร้างสัญญาณลูกคลื่น(Pulse)ที่แปรผันตรงกับการหมุนของเพลลาซึ่งสามารถนำไปใช้ในการรับรู้ความเร็วของเพลลามอเตอร์ในรูปของอัตราจำนวนสัญญาณลูกคลื่นได้ในรูปที่ 3.4 และ 3.5 แสดงรูปลักษณะของ เอนโคเดอร์ที่นับการเพิ่มค่า แบบโรตารีและแบบลิเนียร์



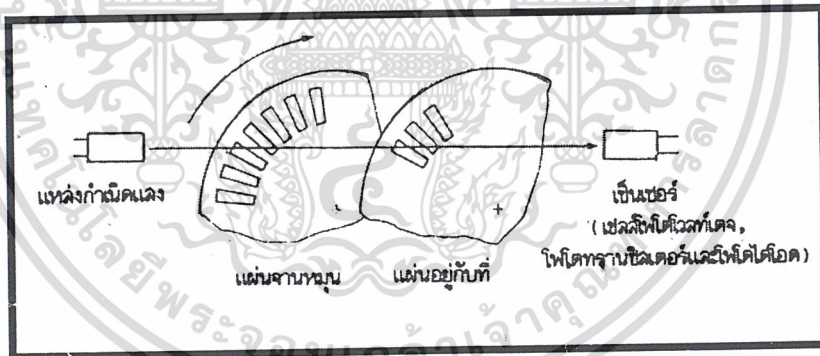
รูปที่ 3.4 แสดงระบบ เอนโคเดอร์แบบโรตารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

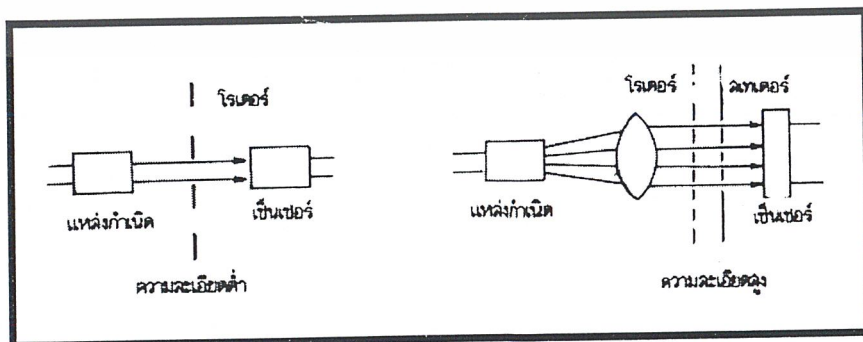
เอนโคเดอ์ที่นับการเพิ่มค่า ประกอบด้วยส่วนที่สำคัญคือตัวกำเนิดแสงจานหมุน (rotary disk) งานอยู่กับที่ และตัวจับสัญญาณ บนแผงงานทำเป็นช่องโดยรอบดังแสดงในรูปที่ 3.6 และบนแผ่นอยู่กับที่จะมีช่องสำหรับให้แสงผ่านตรงข้างไปยังตัวจับสัญญาณ ถ้าเป็นเอนโคเดอ์ที่ใช้วัดความเร็วค่าไม่ต้องมีแผ่นอยู่กับที่ก็ได้ ส่วนตัวกำเนิดแสงอาจจะเป็นหลอดไฟหรือ หลอด LED ก็ได้ดังแสดงในรูปที่ 3.7



รูปที่ 3.5 แสดงเอนโคเดอ์ที่นับการเพิ่มค่าแบบลิเนียร์



รูปที่ 3.6 แสดงตัวอย่างกลไกของออปโตอินคริमेंท์



รูปที่ 3.7 แสดงตัวอย่างตัวจับสัญญาณแบบมีช่องปิดเปิดให้แสงผ่านได้ช่องเดียว

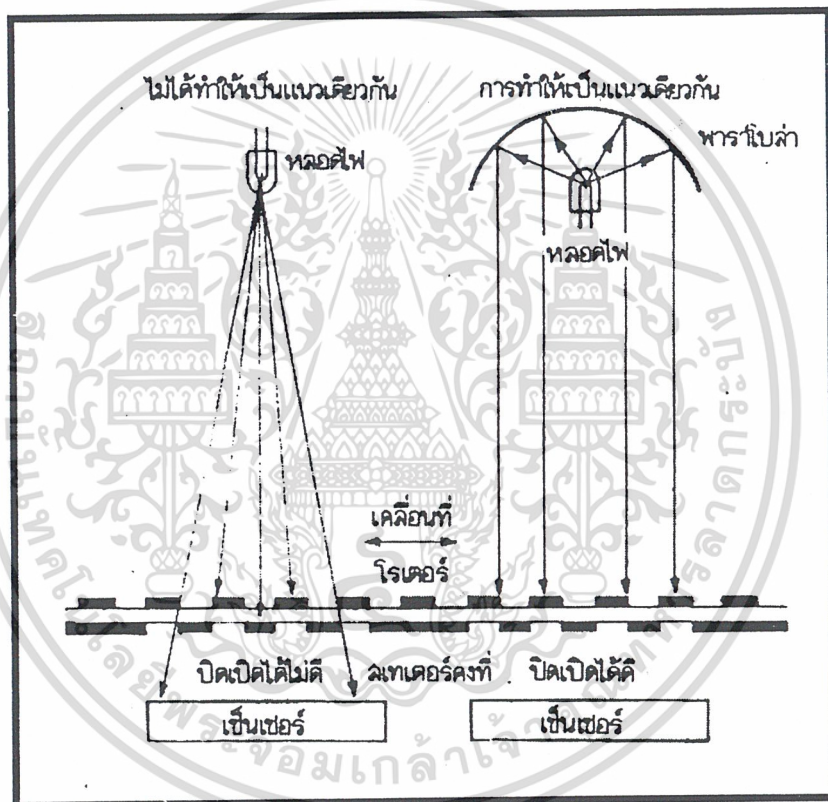
และแบบมีหลายช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1 ความละเอียดของเอนโคเดออร์ที่นับการเพิ่มค่า

ความละเอียดของเอนโคเดออร์คือจำนวนคาบเวลาของสัญญาณเอาท์พุทต่อการหมุนของเพลลา 1 รอบ ซึ่งบวกเป็นจำนวนสัญญาณลูกคลื่นต่อรอบหรือจำนวนรอบต่อ 360 องศาหรือรอบต่อองศา เอนโคเดออร์ที่ใช้กันทั่ว ๆ ไปมีค่าความละเอียดตั้งแต่ 15 ถึง 10000 สัญญาณลูกคลื่นต่อรอบ

เนื่องจากแสงที่ออกจากแหล่งกำเนิดเป็นลำแสงเดี่ยว ถ้าเราต้องการให้แสงที่ผ่านช่องไปยังตัวจับสัญญาณเป็นเส้นตรงพร้อม ๆ กัน (collimation) ก็ทำได้โดยใช้เลนส์หรือ พาราโบลิกรีเฟลคเตอร์ดังแสดงในรูปที่ 3.8



รูปที่ 3.8 แสดงถึงผลของแสงที่เดินในแนวเดียวกันและแสงที่ตกกระทบ

จำนวนสัญญาณลูกคลื่นต่อ 1 รอบของสัญญาณที่เอนโคเดออร์สร้างออกมาจะเท่ากับจำนวนช่องว่างบนแผ่นจานหมุน และความกว้างของช่องว่างก็กับความกว้างของแถบที่ระหว่างช่องว่างจะเท่ากัน เพราะฉะนั้นเราสามารถคำนวณหาความกว้างของช่องว่าง (ω) ได้จากสมการที่ (3.5)

$$\omega = \pi\Phi / 2N \quad (3.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ

Φ = เส้นผ่าศูนย์กลางเฉลี่ยของเพ็ทเทิน

N = จำนวนของความละเอียดเป็นสัญญาณลูกคลื่นต่อรอบ

ω = ความกว้างของช่องว่าง

ค่าของตัวแปรของสมการ (3.5) นี้หาได้จากรูปที่ 3.9 ถ้าให้ D เป็นเส้นผ่าศูนย์กลางของแผ่นหมุนของเอนโคเดอ์ค่าประมาณที่ใกล้เคียงมากของค่าความกว้างของช่องว่าง แสดงได้ในสมการที่ (3.6)

$$\omega = 0.75 \pi \Phi / 2N$$

(3.6)



รูปที่ 3.9 แสดงความสัมพันธ์ระหว่างความเป็นลิเนียร์กับความไม่ได้อันยกลาง

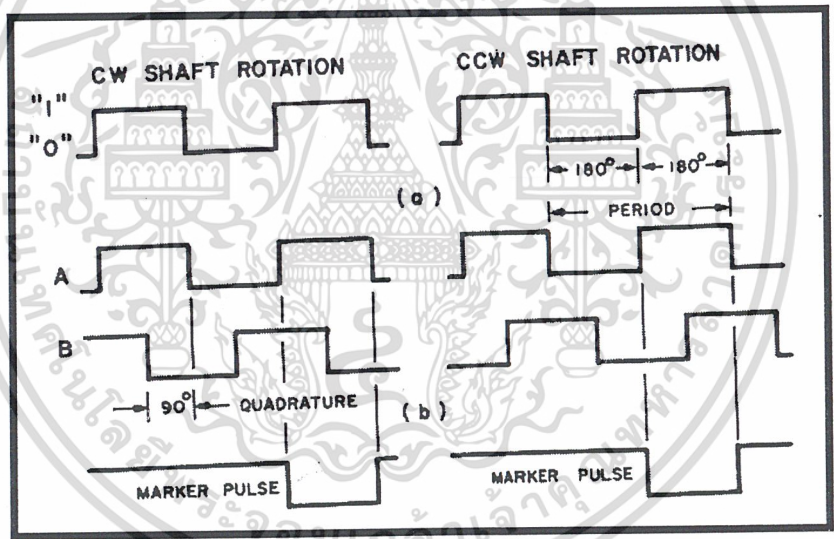
3.4.2 เอาท์พุทของเอนโคเดอ์

โดยทั่ว ๆ ไปแล้วสัญญาณเอาท์พุทที่ออกจากเอนโคเดอ์โดยตรงจะมีระดับไม่เพียงพอในการควบคุมหรือสำหรับการประมวลสัญญาณ ดังนั้นจึงต้องมีวงจรขยายและแปลงรูปร่างลูกคลื่นสัญญาณต่อไว้ในเอนโคเดอ์ด้วยเสมอ สัญญาณลูกคลื่นที่ได้จากตัวรับสัญญาณปกติแล้วจะเป็นรูปสัญญาณสามเหลี่ยมหรือรูปสัญญาณซายน์ขึ้นอยู่กับความละเอียดที่ต้องการ รูปสัญญาณเหล่านี้สามารถทำให้เป็นเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณรูปสี่เหลี่ยมได้โดยการต่อตัวคอมพาราเตอร์เข้ากับลิเนียร์แอมพลิไฟของเอนโคเดอร์ก็จะได้เอาท์พุทเป็นลูกคลื่นสี่เหลี่ยมตามต้องการ

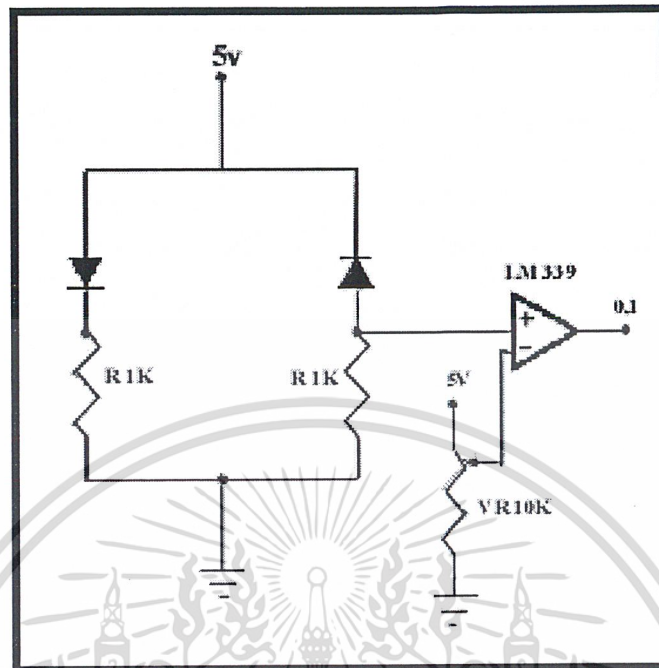
รูปที่ 3.10(ก) แสดงถึงลูกคลื่นเอาท์พุทสี่เหลี่ยมของเอนโคเดอร์ชนิด 1 ช่องไม่ว่าเพลจะหมุนในทิศทางใดก็ได้สัญญาณออกมาเหมือนกันจึงเหมาะที่จะใช้กับงานที่ไม่กำหนดทิศทางเท่านั้น ส่วนในรูปที่ 3.10(ข) แสดงสัญญาณ 2 ชุดที่ได้จากเอนโคเดอร์ชนิด 2 ช่องเฟสของสัญญาณ 2 ช่องนี้จะต่างกัน 90 องศาทางไฟฟ้าเราเรียกสัญญาณ 2 ช่องนี้ว่าเป็นควอดราเจอร์(Quadrature) กันซึ่งเหมาะที่ใช้ในการรับรู้ทิศทางการหมุนของเพลหรือใช้ควบคุมระบบที่ซับซ้อนอื่น ๆ จากสัญญาณในรูปที่ 3.10(ข) จะเห็นได้ว่าสัญญาณทั้ง 2 ช่องจะเริ่มจาก 0 ถึง 1 และ 1 ถึง 0 ขึ้นอยู่กับทิศทางการหมุนของแผ่นหมุนของเอนโคเดอร์

ในเอนโคเดอร์ที่นับการเพิ่มค่าบางชนิดจะมีสัญญาณลูกคลื่นแสดงถึงจำนวนรอบของการหมุนสำหรับใช้เป็นศูนย์ในการอ้างอิงสัญญาณลูกคลื่นที่ใช้แสดงจำนวนรอบนี้จะเกิดขึ้น 1 สัญญาณลูกคลื่นต่อ 1 รอบ โดยทั่วไปแล้วใช้บอกถึงตำแหน่งเชิงกลหรือใช้เป็นสัญญาณเคลียร์จำนวนที่นับไว้ในหน่วยเก็บข้อมูล



รูปที่ 3.10 (ก)แสดงตัวอย่างลูกคลื่นเอาท์พุทสี่เหลี่ยมของอุปกรณ์เอนโคเดอร์ช่องเดียว(ไบไตรีกซ์)

(ข)แสดงสัญญาณเอนโคเดอร์2 ช่องมีมุมเฟสต่างกัน 90 องศา(สองทิศทาง)



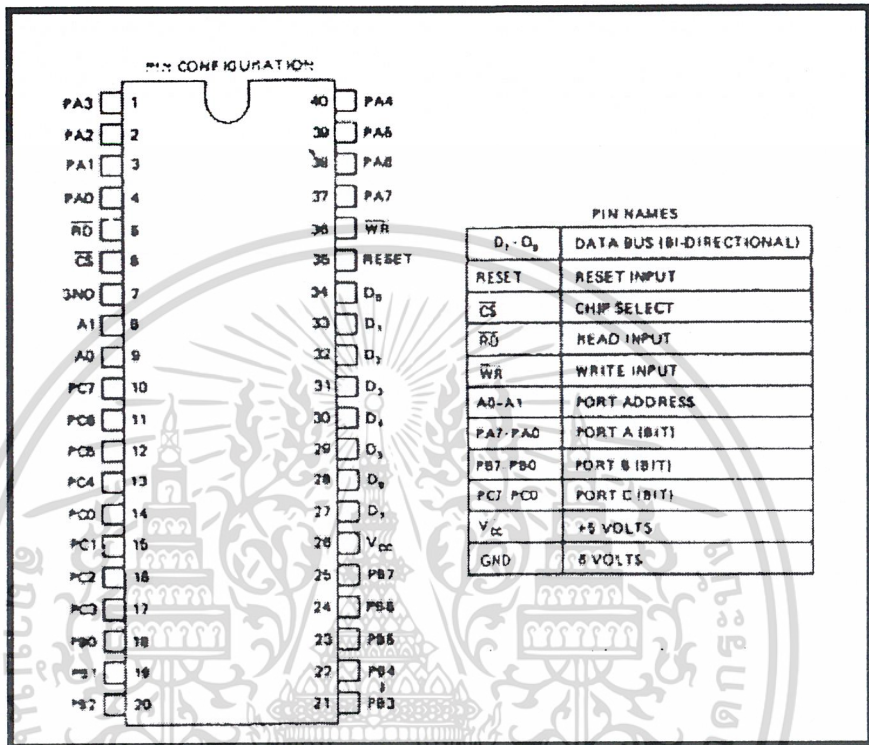
รูปที่ 3.11 แสดงวงจรรับค่าจากเอนโคเดอร์โดยใช้ไอซี LM339

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ทฤษฎีของไอซี8255

3.5.1 โครงสร้างเบื้องต้นของไอซี 8255

ไอซี 8255 เป็นชิป(Chip)ที่มีขา40 ขา สามารถต่อสัญญาณอินพุต(Input)หรือเอาต์พุต(Output)ได้ 3 พอร์ตการจักระเบียงของขาได้จากรูปที่ 3.12



รูปที่ 3.12 แสดงการจักระเบียงขาของไอซี 8255

3.5.2 การแบ่งพอร์ตของไอซี 8255

ไอซี 8255 แบ่งพอร์ตออกเป็น พอร์ต A PA7-PA0, พอร์ต B PB7-PB0, พอร์ต C ซึ่งพอร์ต C นี้แบ่งออกเป็นพอร์ต C บน PC7-PC4 และพอร์ต C ล่าง PC3-PC0 โดยพอร์ตทุกพอร์ตสามารถเป็นได้ทั้งพอร์ตอินพุตและพอร์ตเอาต์พุต

3.5.3 ขาต่าง ๆ ของ ไอซี8255

D0-D7 : เป็นขาสำหรับให้ข้อมูลผ่าน ต่อเข้ากับบัสของไมโครโปรเซสเซอร์

A0-A1 : (สัญญาณแอดเดรส(Address)) สัญญาณจากทั้งสองขาจะถอดรหัสเป็น 4 แอดเดรสเพื่อติดต่อกับพอร์ตและรีจิสเตอร์(Register)ภายใน

PA0-PA7 : เป็นสายสัญญาณที่เป็นพอร์ตของไอซี 8255 ที่เรียกว่าพอร์ต A

PB0-PB7 : เป็นสายสัญญาณที่เป็นพอร์ตของไอซี 8255 ที่เรียกว่าพอร์ต B

PC0-PC7 : เป็นสายสัญญาณที่เป็นพอร์ตของไอซี 8255 ที่เรียกว่าพอร์ต C

CS : (สัญญาณเลือกชิป)เป็นขาที่รับสัญญาณจากภายนอกเพื่อเลือกชิปเมื่อขานี้เป็น “0” จะทำให้ไอซี 8255 ต่อกับระบบบัสของไมโครโปรเซสเซอร์

RD : (สัญญาณอ่าน)เป็นสัญญาณอินพุตจากหน่วยประมวลผลกลางเมื่อสัญญาณนี้และสัญญาณ CS เป็น “0” จะให้หน่วยประมวลผลกลางอ่านข้อมูลจากบัส(Bus)

WR : (สัญญาณเขียน)เช่นเดียวกับสัญญาณ RD แต่เป็นการเขียนข้อมูล

RESET : (สัญญาณรีเซต)เป็นสัญญาณจากภายนอกที่ส่งเข้ามาเพื่อทำการรีเซต 8255 เมื่อได้รับสัญญาณนี้พอร์ททุกพอร์ทจะกลายเป็นพอร์ทอินพุต โดยขา CS,RD,WR,RESET จะแอกทีฟที่ “0” ทั้งหมด

3.5.4 รีจิสเตอร์ภายในของไอซี 8255

พอร์ทแต่ละพอร์ทของไอซี 8255 นั้นเปรียบเสมือนรีจิสเตอร์แต่ละตัวที่สามารถ เขียนหรืออ่านได้ ซึ่งจะถูกกำหนดด้วยแอดเดรสตามที่ตั้งไว้โดยรีจิสเตอร์แต่ละตัวจะได้รับการกำหนดควบคู่กับสัญญาณ RD และ WR เพื่อแสดงความหมายเช่น พอร์ท 10H เป็นพอร์ท A ซึ่งเมื่อเขียนที่พอร์ทนี้ จะเป็นการส่งเอาท์พุท สัญญาณจะมีความหมายต่าง ๆ กันตามแสดงต่อไปนี้ โดยเรียงสัญญาณตามลำดับจาก RD-WR-A1-A0 และความหมายของสัญญาณนั้น

ตารางที่ 3.1 แสดงความหมายของสัญญาณตาม RD-WR-A1-A0

RD	WR	A1	A0	ความหมาย
1	0	0	0	เขียนพอร์ท A ซึ่งเป็นข้อมูล
0	1	0	0	อ่านพอร์ท A ซึ่งเป็นข้อมูล
1	0	0	1	เขียนพอร์ท B ซึ่งเป็นข้อมูล
0	1	0	1	อ่านพอร์ท B ซึ่งเป็นข้อมูล
1	0	1	0	เขียนพอร์ท C ซึ่งเป็นข้อมูล
0	1	1	0	อ่านพอร์ท C ซึ่งเป็นข้อมูล
1	0	1	1	เขียนข้อมูล ซึ่งเป็นรหัสควบคุม
0	1	1	1	อ่านเข้ามา ไม่มีความหมายใด

3.5.5 การใช้งานไอซี 8255

การใช้งานไอซี 8255 นั้นจะต้องส่งรหัสควบคุมเข้าสู่พอร์ทควบคุมเพื่อควบคุมการทำงานของไอซี 8255 โดยการควบคุมการทำงานของไอซี 8255 มีหลายโหมด แต่ละโหมดก็จะแตกต่างกันไปแบ่งเป็น โหมด 0, โหมด 1 และ โหมด 2

3.5.6 การกำหนดโหมดการทำงาน

จะกำหนด โดยการส่งข้อมูลซึ่งเป็นรหัสควบคุมไปยังพอร์ตควบคุมของไอซี 8255 ซึ่งก็คือพอร์ต 30BH โดยแต่ละบิตของรหัสนี้จะมีความหมายของตนเอง โดยความหมายของแต่ละบิตมีดังนี้

- D7 : ถ้าบิตนี้เป็น “1” หมายถึงรหัสนี้มีผลต่อการเปลี่ยนแปลงการเซต(Set) โหมด
- D5-D6 : ใช้เลือกโหมดถ้าเป็น
00 หมายถึง โหมด 0
01 หมายถึง โหมด 1
1x หมายถึง โหมด 2
- D4 : ใช้กำหนดพอร์ต A โดยถ้าเป็น “0” หมายถึงกำหนดพอร์ต A เป็นเอาต์พุต ถ้าเป็น “1” หมายถึงกำหนดพอร์ต A เป็นอินพุต
- D3 : ใช้กำหนดพอร์ต C บน โดยถ้าเป็น “0” หมายถึงกำหนดพอร์ต C บน เป็นเอาต์พุต ถ้าเป็น “1” หมายถึงกำหนดพอร์ต C บนเป็นอินพุต
- D2 : เป็นการเลือกโหมด ถ้าเป็น “0” หมายถึงเป็นโหมด 0 ถ้าเป็น “1” หมายถึง การเลือกโหมด 1
- D1 : ใช้กำหนดพอร์ต B โดยถ้าเป็น “0” หมายถึงกำหนดพอร์ต B เป็นเอาต์พุต ถ้าเป็น “1” หมายถึงกำหนดพอร์ต B เป็นอินพุต
- D0 : ใช้กำหนดพอร์ต C ล่าง โดยถ้าเป็น “0” หมายถึงกำหนดพอร์ต C ล่าง เป็นเอาต์พุต ถ้าเป็น “1” หมายถึงกำหนดพอร์ต C ล่าง เป็นอินพุต
- การโปรแกรมไอซี 8255 จะทำได้โดยเอาต์พุตค่าที่ต้องการไปยังพอร์ตควบคุม เช่น ถ้าต้องการให้ทุกพอร์ตเป็นพอร์ตเอาต์พุตหมด และให้เป็น โหมด 0 ดังนั้น รหัสนี้จะเท่ากับ 10000000 หรือ 80H

3.6.1 การทำงานโหมด 0

ในโหมดนี้ทุกพอร์ตสามารถเป็น ได้ทั้งอินพุตและเอาต์พุต ทั้งพอร์ต A, พอร์ตB,พอร์ตC บน และพอร์ต C ล่าง ดังนั้นจึงมีรูปแบบทั้งหมด $2 \times 2 \times 2 \times 2 = 16$ แบบ

3.6.2 การทำงานโหมด 1

เป็นโหมดที่ทำให้อินพุตเอาต์พุตมีการตรวจสอบสัญญาณ(Handshaking)โดยการใช้พอร์ต A และพอร์ตB เป็นหลัก ส่วนพอร์ต C ใช้เป็นตัวตรวจสอบสัญญาณ โดย

พอร์ต C บน จะเป็นตัวตรวจสอบสัญญาณของพอร์ต A ในขณะที่

พอร์ต C ล่าง จะเป็นตัวตรวจสอบสัญญาณของพอร์ต B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อให้ไอซี 8255 เป็นโหมด 1 แล้ว พอร์ต C จะเป็นสัญญาณควบคุม โดยแต่ละบิต ของพอร์ต C จะมีค่าดังตารางที่ 3.2

ตารางที่ 3.2 แสดงสถานะของขาต่างๆ ในพอร์ต C ในการทำงานโหมด 1

ขา	กรณีอินพุต	กรณีเอาต์พุต
PC0	INTR-B	INTR-B
PC1	IBF-B	OBF-B
PC2	STB-B	ACK-B
PC3	INTR-A	INTR-A
PC4	STB-A	I/O
PC5	IBF-A	I/O
PC6	I/O	ACK-A
PC7	I/O	OBF-A

INTR : เป็นขาเอาต์พุตแอกทีฟที่ 1 จะแอกทีฟหลังจากที่ได้รับ STB แอกทีฟ 0 ช่วงที่ STB เปลี่ยนจากระดับต่ำไปสูงขา INTR จะแอกทีฟทันที(ต้องอินเวิล อินเวอร์รท์ ฟลิปฟลอปไว้ก่อน)โดยเซตที่ PC4 ถ้าใช้งานที่พอร์ต A และเซตที่ PC2 ถ้าใช้พอร์ต B)

IBF : เป็นขาเอาต์พุตแอกทีฟ(Active)ที่ 1 แสดงถึงอินพุตบัฟเฟอร์(Buffer) เต็ม(เมื่อหน่วยประมวลผลกลาง อ่านข้อมูลจากพอร์ตอินพุตไปแล้วจะทำให้ขานี้เปลี่ยนเป็นลอจิก(Logic) 0 คือว่างพร้อมที่จะรับข้อมูลใหม่)

OBF : เป็นขาเอาต์พุตแอกทีฟ 0 เมื่อหน่วยประมวลผลกลางส่งข้อมูลออกมาที่พอร์ตของไอซี 8255 แล้วขานี้จะเป็น 0 หลังจากช่วงขอบขาขึ้นของ WR และจะเปลี่ยนเป็น 1 ในช่วงขอบขาขึ้นของ ACK (ขา OBF ถ้ามีลอจิก 0 หมายถึงเอาต์พุตบัฟเฟอร์เต็ม, ถ้าเป็น 1 หมายถึงว่าง)

STB : เป็นขาอินพุตแอกทีฟที่ 0 เป็นขาที่อุปกรณ์ภายนอกส่งมากระตุ้นให้อินพุตพอร์ตเลขที่ข้อมูลไว้

ACK : เป็นขาอินพุตแอกทีฟที่ 0 เป็นสัญญาณตอบรับอุปกรณ์ภายนอกส่งมายังไอซี 8255 หลังจากให้อินพุตพอร์ตเก็บข้อมูลได้แล้ว

I/O : อินพุต,เอาต์พุต

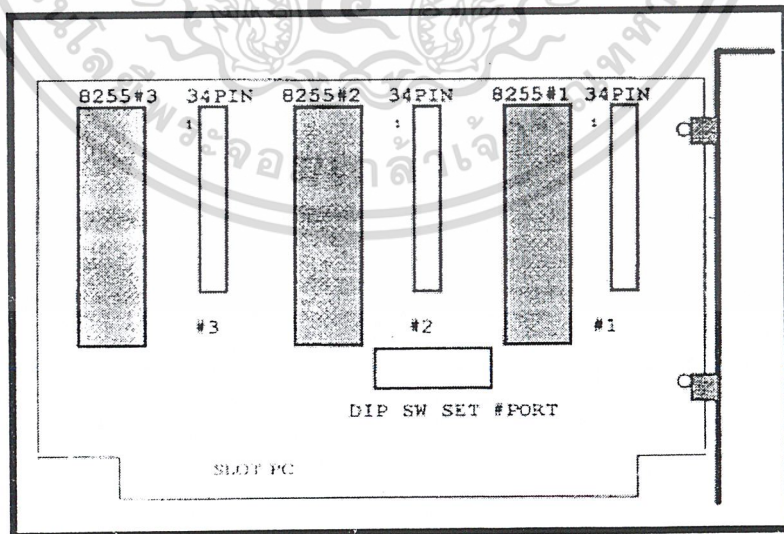
3.6.3 การทำงานในโหมด 2

การทำงานในโหมด 2 นั้นจะใช้พอร์ท A สามารถทำหน้าที่เป็นได้ทั้งพอร์ทอินพุตและพอร์ทเอาต์พุตโดยโครงสร้างของพอร์ท A ทั้งอินพุตและเอาต์พุตจะมี ตัวตรวจสอบสัญญาณทั้งคู่ในขณะที่พอร์ท C จำทำหน้าที่เป็นสัญญาณตรวจสอบโดยมีสัญญาณแต่ละขาดังต่อไปนี้ ตารางที่3.3 แสดงสถานะของขาต่าง ๆ ในพอร์ท C ในการทำงานโหมด 2

พอร์ท C	ความหมาย
PC0	I/O
PC1	I/O
PC2	I/O
PC3	INTR-A
PC4	STB-A
PC5	IBF-A
PC6	ACK-A
PC7	OBF-A

เมื่อโปรแกรมพอร์ท A เป็นโหมด 2 แล้วพอร์ท B จะต้องโปรแกรมเป็นโหมด 0 หรือโหมด 1 ก็ได้ซึ่งก็ทำงานแบบแยกอิสระ

3.7 รายละเอียดเกี่ยวกับการ์ด ET-PC8255



รูปที่ 3.14 แสดงลักษณะของการ์ด ET-PC8255

เป็นลักษณะของการ์ดอินเทอร์เฟซที่ใช้เชื่อมต่อกับเครื่องคอมพิวเตอร์เพื่อขยายระบบอินพุตและเอาต์พุต ให้ใช้งานได้มากยิ่งขึ้นซึ่งการ์ดนี้สามารถที่จะรับสัญญาณอินพุตและให้สัญญาณเอาต์พุตออกมา เอกสารนี้ในรูปดิจิทัล (Digital) มีจำนวนใช้งานเป็นอินพุตหรือเอาต์พุต จำนวน 9 พอร์ท (Port) หรือ 72 บิต (Bit) ค่าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

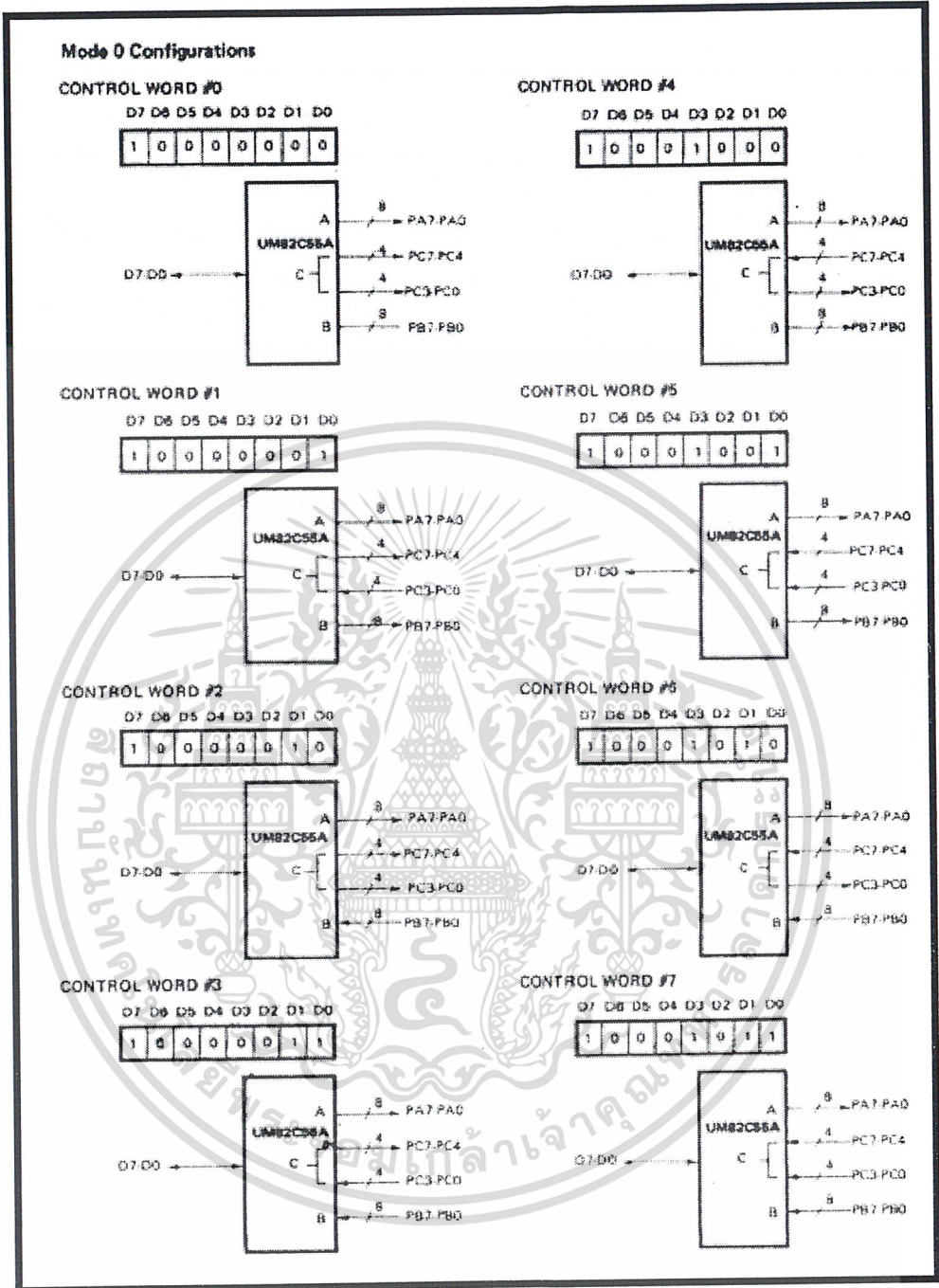
VO)ทำให้มีความอ่อนตัวในการนำไปประยุกต์ใช้งานในด้านต่าง ๆ ได้มากยิ่งขึ้นซึ่งการ์ดนี้ มีอุปกรณ์
ร่วมและมีจุดเด่น ๆ ดังนี้คือ

ไอซี 8255 จำนวน 3 ตัว ซึ่งสามารถที่จะใช้โปรแกรมให้เป็นได้ทั้งอินพุตและเอาต์พุตตามความต้องการ
ของผู้ใช้เองได้ทั้งหมด 9 พอร์ต (Port)

การใช้งาน 8255 สามารถโปรแกรมให้เป็นได้ทั้งอินพุตและเอาต์พุตได้ทั้ง 3 พอร์ต โดยสามารถ
กำหนดหน้าที่การใช้งานของแต่ละพอร์ตได้โดยการส่งสัญญาณค่าต่าง ๆ ไปที่พอร์ตควบคุม (Control
Port) ซึ่งมีรายละเอียดดังรูปที่ 3.15

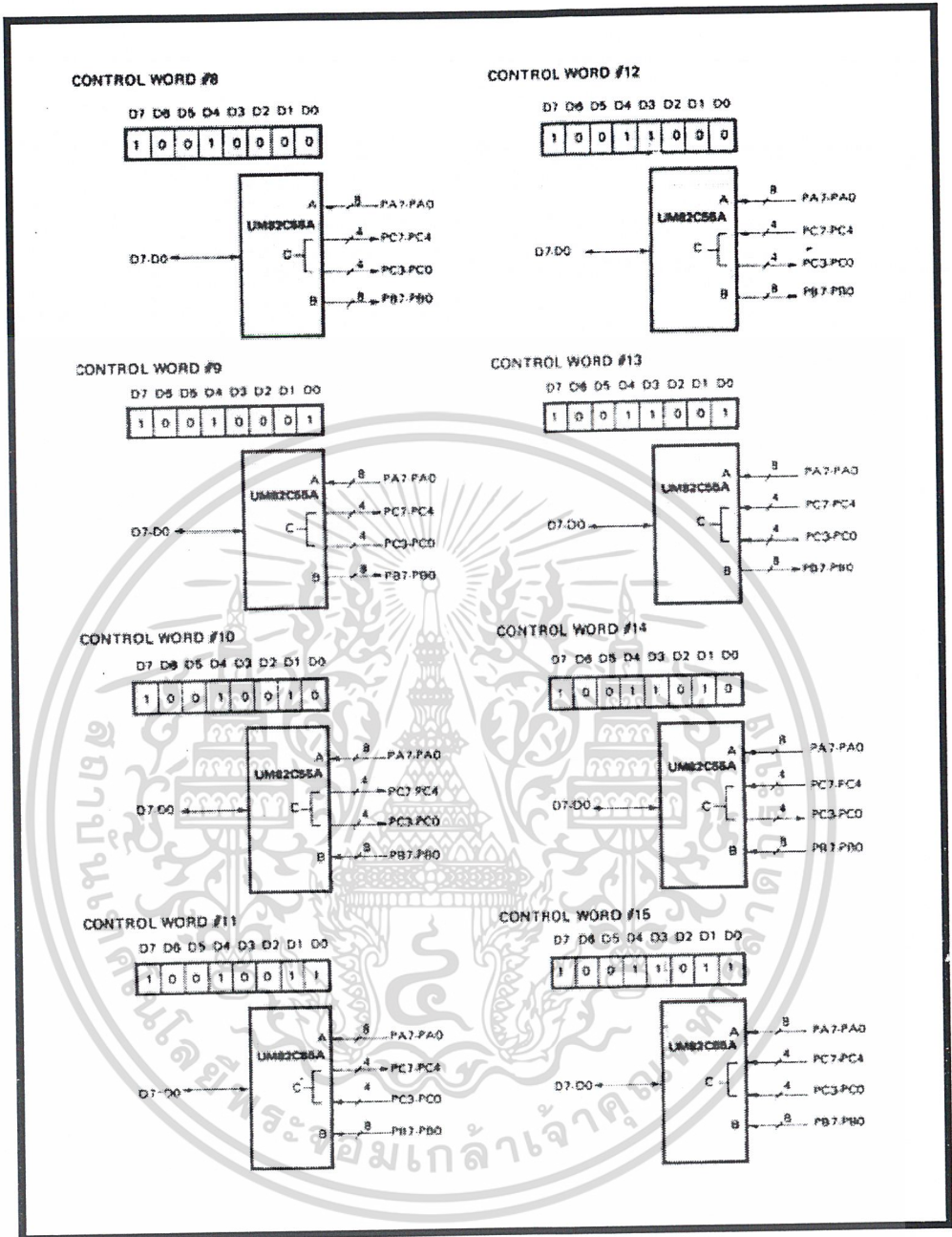


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 แสดงค่าต่าง ๆ ที่ส่งให้พอร์ทควบคุมเพื่อกำหนดการทำงานของ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



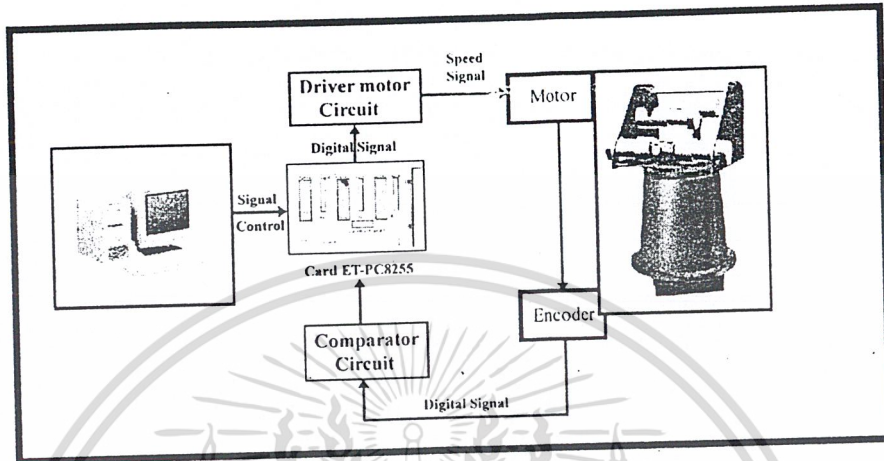
รูปที่ 3.15 (ต่อ) แสดงค่าต่าง ๆ ที่ส่งให้พอร์ทควบคุมเพื่อกำหนดการทำงานของ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ขั้นตอนการทำงาน

ระบบการควบคุมแขนกล Scorbot ER3 ด้วยคอมพิวเตอร์มีระบบการทำงานดังรูปที่ 4.1



รูปที่ 4.1 แสดงระบบควบคุมแขนกล Scorbot ER3 ด้วยคอมพิวเตอร์

การทำงานของระบบควบคุมคอมพิวเตอร์จะส่งสัญญาณควบคุมโดยผ่านการ์ด(Card) ET-PC 8255 เมื่อการ์ด ET-PC 8255 ได้รับสัญญาณแล้วจะส่งสัญญาณต่อไปยัง Control Interface Box แล้วส่งสัญญาณต่อไปยังมอเตอร์เมื่อมอเตอร์หมุนทำให้เกิดสัญญาณที่ Encoder สัญญาณที่ได้นี้จะถูกส่งต่อไปให้กับ Control Interface Box เพื่อขยายสัญญาณให้ได้ระดับที่ การ์ด ET-PC 8255 สามารถที่จะรับค่าได้เมื่อ การ์ด ET-PC 8255 ได้ทำการขยายสัญญาณแล้วสัญญาณนี้คอมพิวเตอร์จะใช้ในการควบคุมตำแหน่งของแขนกล Scorbot ER3 ต่อไป

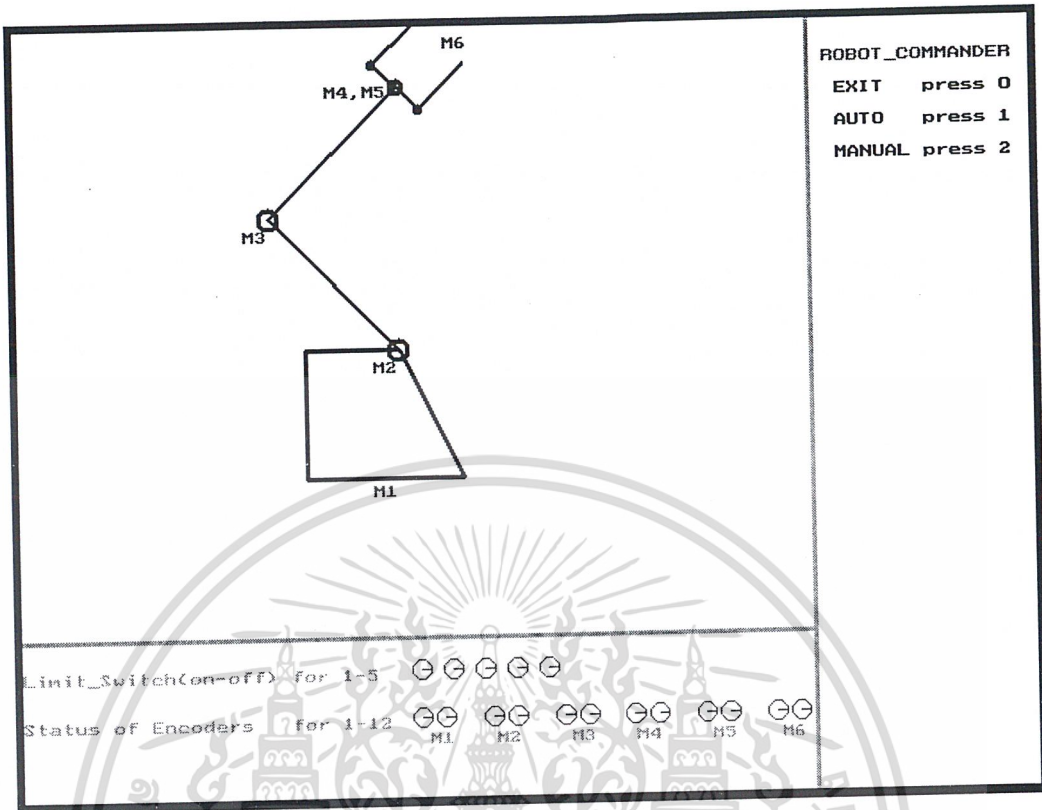
จากขั้นตอนการทำงานของระบบข้างบนสามารถแบ่งรายละเอียดเป็นส่วน ๆ ได้ดังนี้

- 1.คอมพิวเตอร์จะกล่าวถึงรายละเอียดของโปรแกรมที่ใช้ควบคุม
- 2.การ์ด ET-PC 8255 จะกล่าวถึงการเชื่อมต่อใช้งาน
- 3.Control Interface Box จะกล่าวถึงการส่งสัญญาณควบคุมความเร็วการหมุนของมอเตอร์และการรับสัญญาณจากเอนโคเดอร์
- 4.ในส่วนของแขนกล Scorbot ER3 จะกล่าวถึงลักษณะของแขนกล Scorbot ER3 สายสัญญาณจากแขนกล Scorbot ER

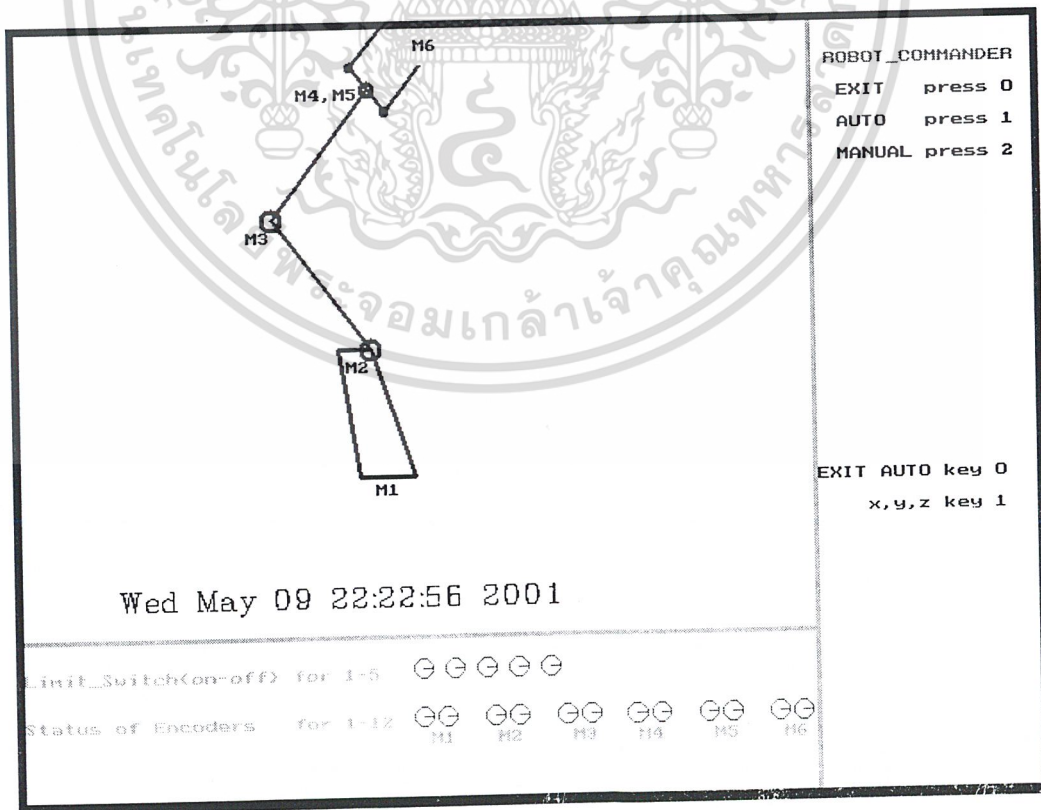
4.1 การทำงานของโปรแกรมควบคุมแขนกล Scorbot ER3

โปรแกรมควบคุมแขนกล Scorbot ER3 ถูกเขียนด้วยภาษาซี ลักษณะของโปรแกรมสามารถดูได้จากรูปที่ 4.2, 4.3 และ 4.4 โปรแกรมนี้สามารถควบคุมแขนกลได้ 2 แบบคือสามารถควบคุมการหมุนของมอเตอร์แต่ละตัวได้โดยตรงโดยผ่านทางคีย์บอร์ด(Key board)และสามารถรับค่าพิกัด x , y

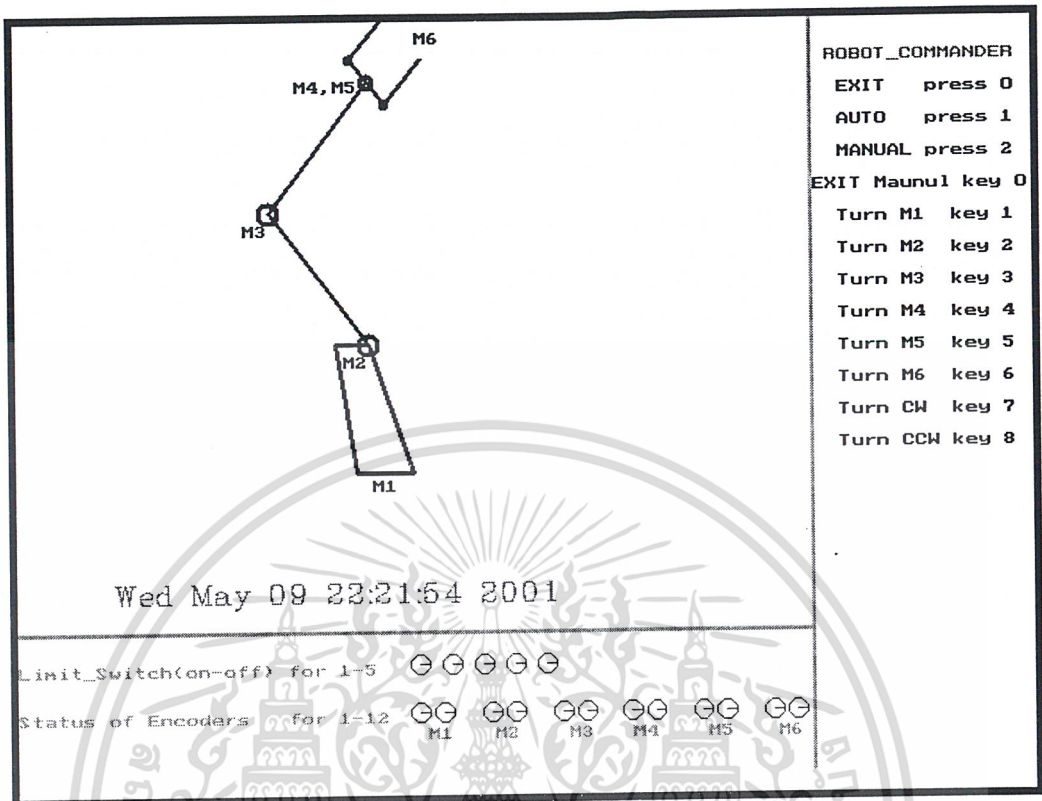
เอกสารนี้เป็นเอกสารลิขสิทธิ์ส่วนตัวสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงลักษณะของโปรแกรมควบคุมแขนกล Scrobot ER3



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงลักษณะของโปรแกรมควบคุมแขนกล Scrobot ER3 ในโหมด Manual

จากโปรแกรมควบคุมแขนกล Scrobot ER3 จะทำงานในโหมดกราฟิกส์ฟังก์ชันการทำงานในการเข้าโหมดการฝึกส์ตามโปรแกรมข้างล่าง

```
void CHECK_CARD()
{
  initgraph(&GrDriver,&GrMode,"D:\\TC\\bgi");
  GrError=graphresult();
  if (GrError!=0)
  {
    clrscr();
    cout<<"\a GRAHPIC ERROR : "<<grapherrormsg(GrError);
    getch();
    exit(GrError);
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน CHECK_CARD() จะทำการตรวจสอบระบบกราฟิกส์ของเครื่องคอมพิวเตอร์ที่ใช้งานอยู่ถ้า GrError มีค่าเท่ากับศูนย์สามารถเข้าใช้งานในโหมดกราฟิกส์ได้แต่ถ้ามีค่าไม่เท่ากับศูนย์จะไม่สามารถเข้าใช้งานได้ความหมายของค่าต่าง ๆ ที่ GrError ดูได้จากตารางที่ 4.1

ตารางที่ 4.1 แสดงรหัสซึ่งเป็นผลจากการเปลี่ยนโหมดและความหมายของรหัส

GrError	Grapherrormag	ความหมาย
0	No error	ไม่มีความผิด
-1	(BGI)graphics not installed	ไม่มี BGI
-2	Graphics hardware not detected	ไม่พบระบบกราฟิกส์
-3	Device driver file not found	ไม่พบไฟล์ไดรเวอร์
-4	Invalid device driver file	ไฟล์ไดรเวอร์ไม่ถูกต้อง
-5	Not enough memory to load driver	หน่วยความจำไม่พอที่จะรับไดรเวอร์
-6	Out of memory is scan fill	หน่วยความจำไม่พอใน scan fill
-7	Out of memory in flood fill	หน่วยความจำไม่พอใน flood fill
-8	Font file not found	ไม่พบฟอนต์ไฟล์
-9	Not enough memory to load font	หน่วยความจำไม่พอที่จะรับฟอนต์
-10	Invalid graphics mode for selected driver	เลือกโหมดจากไดรเวอร์ไม่ถูกต้อง
-11	Graphics error	การพิกซ์ไม่ถูกต้อง
-12	Graphics I/O error	I/Oของกราฟิกส์ไม่ถูกต้อง
-13	Invalid font file	ฟอนต์ไฟล์ไม่ถูกต้อง
-14	Invalid font number	เบอร์ฟอนต์ไม่ถูกต้อง
-15	Invalid device number	เบอร์ไดรเวอร์ไม่ถูกต้อง
-18	Invalid version number	เบอร์เวอร์ชันไม่ถูกต้อง

รูปของแกนกล Scorbot ER3 เกิดจากเส้นตรงหลายเส้นมาประกอบกันจนกลายเป็นรูปร่างแกนกล Scorbot ER3 ตามฟังก์ชันข้างล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void BASE()
{
    void CHECK_CARD();
    CHECK_CARD();
    setlinestyle(SOLID_LINE,0,3);
    setcolor(15);
    line(x8,Max_y-y8,x9,Max_y-y9);
    line(x9,Max_y-y9,x10,Max_y-y10);
    line(x10,Max_y-y10,x11,Max_y-y11);
    line(x11,Max_y-y11,x8,Max_y-y8);
    setcolor(14);
    circle(x10,Max_y-y10,6);
}
void LINK_1()
{
    setlinestyle(SOLID_LINE,0,3);
    setcolor(15);
    line(x1,Max_y-y1,x2,Max_y-y2);
    setcolor(14);
    circle(x2,Max_y-y2,6);
}
void LINK_2()
{
    setlinestyle(SOLID_LINE,0,3);
    setcolor(15);
    line(x2,Max_y-y2,x3,Max_y-y3);
    setcolor(14);
    circle(x3,Max_y-y3,4);
}
void LINK_3()
{
    setlinestyle(SOLID_LINE,0,3);

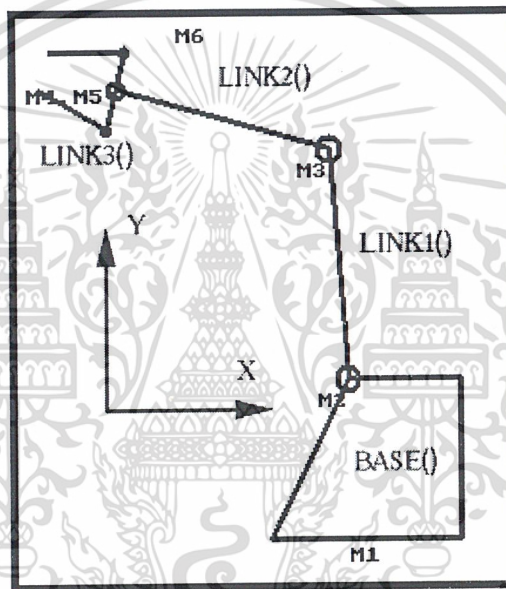
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setcolor(15);
line(x4,Max_y-y4,x5,Max_y-y5);
line(x4,Max_y-y4,x6,Max_y-y6);
line(x5,Max_y-y5,x7,Max_y-y7);
setcolor(14);
circle(x4,Max_y-y4,2);
circle(x5,Max_y-y5,2);
}

```



รูปที่4.5 แสดงรูปที่เกิดจากฟังก์ชัน BASE(),LINK1(),LINK2() และLINK()3

เมื่อได้รูปของแขนกล Scorbobot ER3 ขึ้นต่อไปต้องนำรูปเหล่านี้มาหมุนโดยให้ขนาดของรูปเท่าเดิมการหมุนนี้จะหมุนรอบแกน z และ y สมการการหมุนรอบแกน z คูได้จากสมการ(4.1), (4.2) การหมุนรอบแกน y ค่าของ y เท่าเดิมค่า x เปลี่ยนไปตามสมการ(4.3)

$$x' = x + L \cos \theta \tag{4.1}$$

$$y' = y + L \sin \theta \tag{4.2}$$

$$x' = x + L \cos \theta \cos \alpha \tag{4.3}$$

x = ค่าเริ่มต้นในแกน x

y = ค่าเริ่มต้นในแกน y

L = ความยาวของเส้นตรง ในที่นี้กำหนดให้ L=115

θ = มุมที่หมุนไปในแกน z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

α = มุมที่หมุนไปในแกน y

ฟังก์ชันการหมุนคู่ได้จากโปรแกรมข้างล่าง

```
void RO_BASE()
{
    rad0=((ang0)*Pi)/180;
    if((ang0>=0)&&(ang0<=90))
    {
        x8_n=x8+(L0*cos(1.57-rad0));
        x9_n=x9-(L0*cos(1.57-rad0));
        x10_n=x10-(L0*cos(1.57-rad0));
        x11_n=x11+(L0*cos(1.57-rad0));
    }
    else if((ang0>90)&&(ang0<=270))
    {
        x8_n=(-1*L0*cos(rad0))+x8+L0;
        x9_n=(L0*cos(rad0))+x9-L0;
        x10_n=(L0*cos(rad0))+x10-L0+40;
        x11_n=(-1*L0*cos(rad0))+x11+L0+40;
    }
    else if((ang0>270)&&(ang0<=360))
    {
        x8_n=x8+(L0*cos(rad0));
        x9_n=x9-(L0*cos(rad0));
        x10_n=x10-(L0*cos(rad0));
        x11_n=x11+(L0*cos(rad0));
    }
}

void RO_LINK_1()
{
    rad1=(ang1*Pi)/180;
    y2_n=y1+(115*sin(rad1));
    x2_n=x10_n+115*cos(rad1)*cos(rad0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อปรับ DIP SW เสร็จแล้วในโรงงานเซตให้ DIP SW มีค่าเท่ากับ 300H การทำงานจะทำในโหมด DOS

4.3 การส่งสัญญาณควบคุมความเร็วมอเตอร์และรับสัญญาณจากเอนโคเดอร์

การส่งสัญญาณควบคุมความเร็วมอเตอร์จะส่งเป็นเลข 8 บิตต่อมอเตอร์ 1 ตัวในแกนกล Scorbobot ER3 มี มอเตอร์ทั้งหมด 6 ตัว จะใช้บิตทั้งหมด 48 บิตดังนั้นต้องใช้พอร์ต: 2 พอร์ตของการ์ด PC 8255 ความหมายของเลข 8 บิตที่ใช้ควบคุม มอเตอร์และการใช้งานของพอร์ตดูได้จากตารางข้างล่าง

ตารางที่ 4.2 แสดงความหมายของเลข 8 บิต

เลข 8 บิต	ความหมาย
00001111	หมุนตามเข็มนาฬิกาหมุนด้วยความเร็วสูงสุด
00011111	หมุนตามเข็มนาฬิกาหมุนด้วยความเร็ว 15/16
00101111	หมุนตามเข็มนาฬิกาหมุนด้วยความเร็ว 14/16
11111111	ไม่หมุน
11110000	หมุนตามทวนนาฬิกาหมุนด้วยความเร็วสูงสุด
11110001	หมุนตามทวนนาฬิกาหมุนด้วยความเร็ว 15/16
11110010	หมุนตามทวนนาฬิกาหมุนด้วยความเร็ว 14/16
11111111	ไม่หมุน
นอกจากนี้ไม่ใช้งาน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 แสดงการใช้งานของพอร์ต PC8255

พอร์ตของ PC8255	การใช้งาน
พอร์ต ที่1	พอร์ต PA1 ใช้ควบคุมการหมุนของมอเตอร์ตัวที่ 1
	พอร์ต PB2 ใช้ควบคุมการหมุนของมอเตอร์ตัวที่ 2
	พอร์ต PC1 ใช้ควบคุมการหมุนของมอเตอร์ตัวที่ 3
พอร์ต ที่2	พอร์ต PA2 ใช้ควบคุมการหมุนของมอเตอร์ตัวที่ 4
	พอร์ต PB2 ใช้ควบคุมการหมุนของมอเตอร์ตัวที่ 5
	พอร์ต PC2 ใช้ควบคุมการหมุนของมอเตอร์ตัวที่ 6
พอร์ต ที่3	พอร์ต PA3 ใช้รับสัญญาณจากเอนโคเดอร์
	พอร์ต PB3 ใช้รับสัญญาณจากเอนโคเดอร์

ฟังก์ชันการใช้งานดูได้จากโปรแกรมข้างล่าง

```
#define PA1    0x300
#define PB1    0x301
#define PC1    0x302
#define CTRL1  0x303
#define PA2    0x304
#define PB2    0x305
#define PC2    0x306
#define CTRL2  0x307
```

output(CTRL1,0x80);จะอยู่ในฟังก์ชัน void main() เพื่อให้ทุกพอร์ตส่งสัญญาณได้อย่างเดียว

output(CTRL2,0x80);จะอยู่ในฟังก์ชัน void main()

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void STOP_M1()
{
    output(PA1,0xFF);//0xFF=11111111
}
```

จากฟังก์ชัน void STOP_M1() เป็นการส่งสัญญาณให้มอเตอร์ตัวที่ 1 หยุดหมุนมอเตอร์ตัวอื่น ๆ ก็ใช้หลักการเดียวกันแต่เปลี่ยนจาก PA1 เป็นอย่างอื่น ตามตารางที่ 4.3

```
void MOTOR(int port_no,unsigned char speed)
{
    output(port_no,speed);
}
```

speed เป็นส่งสัญญาณว่าจะให้หมุนเร็วเท่าไรหมุนทวนเข็มนาฬิกาหรือตามนาฬิกาโดยดูค่าตารางที่ 4.2 แต่ต้องเปลี่ยนเป็นเลขฐาน 16 ก่อน

การรับสัญญาณจากเอนโคคเตอร์จะมีทั้งหมด 12 บิต มอเตอร์ 1 ตัวใช้ 2 บิตเหตุที่ใช้ 2 บิตเพื่อจะบอกทิศทางการหมุนของมอเตอร์ว่าหมุนทวนเข็มนาฬิกาหรือตามเข็มนาฬิกาฟังก์ชันการรับสัญญาณจากเอนโคคเตอร์ดูได้จากโปรแกรมต่อไปนี้

```
#define PA3    0x308
#define PB3    0x309
#define CTRL3  0x30B
output(CTRL3,0x9B); จะอยู่ในฟังก์ชัน void main() เชื่อมให้ทุกพอร์รับสัญญาณได้อย่างเดียว
void IN_ENCODER()
{
    signal0= inport(PA3);
    signal1= inport(PB3);
}
```

ค่าที่ได้จากสัญญาณเอนโคคเตอร์จะเป็นเลขฐาน 10 ต้องแปลงเป็นเลขฐาน 2 ก่อนโดยใช้โปรแกรมข้างล่างนี้

```
int signal2[7],signal3[7];
while(signal0!=0)
{
    signal2[ii]=signal0%2;
    signal0=signal0/2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

}

ความหมายของการเก็บค่าสัญญาณจากเอนโคเดออร์โดยใช้อาร์เรย์ดูได้จากตารางที่ 4.4

ตารางที่ 4.4 แสดงความหมายที่ได้จากสัญญาณที่ได้รับจากเอนโคเดออร์

ลำดับที่	ความหมาย
Signal[0]	1 = เอนโคเดออร์ตัวที่ 1 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 1 ทำงาน
Signal[1]	1 = เอนโคเดออร์ตัวที่ 2 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 2 ทำงาน
Signal[2]	1 = เอนโคเดออร์ตัวที่ 3 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 3 ทำงาน
Signal[3]	1 = เอนโคเดออร์ตัวที่ 4 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 4 ทำงาน
Signal[4]	1 = เอนโคเดออร์ตัวที่ 5 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 5 ทำงาน
Signal[5]	1 = เอนโคเดออร์ตัวที่ 6 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 6 ทำงาน
Signal[6]	1 = เอนโคเดออร์ตัวที่ 7 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 7 ทำงาน
Signal[7]	1 = เอนโคเดออร์ตัวที่ 8 ไม่ทำงาน 0 = เอนโคเดออร์ตัวที่ 8 ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 สายสัญญาณจากแขนกล Scorbot ER3

สายสัญญาณจากแขนกล Scorbot ER3 ถูกรวมเข้าด้วยตัวคอนเนกเตอร์(Connector) 50 ขา

ตารางที่3.5 แสดงสายสัญญาณจากตัวคอนเนกเตอร์ 50 ขาจากแขนกล Scorbot ER3

หมายเลข	ตำแหน่ง	สายสี	จากอุปกรณ์
1	มอเตอร์ตัวที่ 2	เทา	เอนโคคเตอร์(4)
2	มอเตอร์ตัวที่ 1	น้ำตาล	เอนโคคเตอร์(4)
3	มอเตอร์ตัวที่ 5	เขียว	เอนโคคเตอร์(3)
4	มอเตอร์ตัวที่ 3	น้ำตาล	เอนโคคเตอร์(3)
5	มอเตอร์ตัวที่ 1	ขาว	เอนโคคเตอร์(3)
6	มอเตอร์ตัวที่ 5	ส้ม	ลิมิตสวิตช์
7	มอเตอร์ตัวที่ 2	เทา	ลิมิตสวิตช์
8	ลิงค์อันที่ 5	น้ำเงิน	ลิมิตสวิตช์
9	มอเตอร์ตัวที่ 5	เหลือง	เอนโคคเตอร์(2)
10	มอเตอร์ตัวที่ 3	เหลือง	เอนโคคเตอร์(2)
11	มอเตอร์ตัวที่ 1	เหลือง	เอนโคคเตอร์(2)
12	มอเตอร์ตัวที่ 6	เหลือง	ขั้วมอเตอร์
13	มอเตอร์ตัวที่ 5	ขาว	ขั้วมอเตอร์
14	มอเตอร์ตัวที่ 4	ส้ม	ขั้วมอเตอร์
15	มอเตอร์ตัวที่ 3	ขาว	ขั้วมอเตอร์
16	มอเตอร์ตัวที่ 2	ขาว	ขั้วมอเตอร์
17	มอเตอร์ตัวที่ 1	เทา	ขั้วมอเตอร์
18	มอเตอร์ตัวที่ 5	ฟ้า	เอนโคคเตอร์(4)
19	มอเตอร์ตัวที่ 6	เขียว	เอนโคคเตอร์(3)
20	มอเตอร์ตัวที่ 4	เขียว	เอนโคคเตอร์(3)
21	มอเตอร์ตัวที่ 2	ขาว	เอนโคคเตอร์(3)
22	ไม่ใช้งาน		
23	ใต้ฐานแขนกล	น้ำตาล	ลิมิตสวิตช์
24	ลิงค์อันที่ 4	ขาว	ลิมิตสวิตช์
25	มอเตอร์ตัวที่ 6	เหลือง	เอนโคคเตอร์(2)
26	มอเตอร์ตัวที่ 4	เหลือง	เอนโคคเตอร์(2)
27	มอเตอร์ตัวที่ 2	เหลือง	เอนโคคเตอร์(2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 (ต่อ) แสดงสายสัญญาณจากตัวคอนเนกเตอร์ 50 ขาจากแขนกล Scorbot ER3			
28	มอเตอร์ตัวที่ 6	ม่วงดำ	เอน โคคเคอร์(1)
*29	[ปากจับของแขนกล][มอเตอร์ตัวที่ 5]	[ส้ม][ขาว]	[ลิมิตสวิตช์][เอน โคคเคอร์(1)]
*30	[ปากจับของแขนกล][มอเตอร์ตัวที่ 4]	[น้ำเงิน][ขาว]	[ลิมิตสวิตช์][เอน โคคเคอร์(1)]
*31	[ลิงค์อันที่ 5][มอเตอร์ตัวที่ 3]	[ขาว][ขาว]	[ลิมิตสวิตช์][เอน โคคเคอร์(1)]
*32	[ลิงค์อันที่ 4][มอเตอร์ตัวที่ 2]	[ขาว][ขาว]	[ลิมิตสวิตช์][เอน โคคเคอร์(1)]
*33	[ลิงค์อันที่ 2][มอเตอร์ตัวที่ 1]	[ขาว][ขาว]	[ลิมิตสวิตช์][เอน โคคเคอร์(1)]
34	มอเตอร์ตัวที่ 6	น้ำตาล	เอน โคคเคอร์(4)
35	มอเตอร์ตัวที่ 4	ส้ม	เอน โคคเคอร์(4)
36	มอเตอร์ตัวที่ 3	เขียว	เอน โคคเคอร์(4)
37-44	ไม่ใช้งาน		
45	มอเตอร์ตัวที่ 6	เทา	ตรงข้ามมอเตอร์
46	มอเตอร์ตัวที่ 5	เทา	ตรงข้ามมอเตอร์
47	มอเตอร์ตัวที่ 4	ขาว	ตรงข้ามมอเตอร์
48	มอเตอร์ตัวที่ 3	ส้ม	ตรงข้ามมอเตอร์
49	มอเตอร์ตัวที่ 2	ขาว	ตรงข้ามมอเตอร์
50	มอเตอร์ตัวที่ 1	ขาว	ตรงข้ามมอเตอร์
หมายเหตุ	* ใช้สายไฟเส้นเดียวกัน		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทวิจารณ์และสรุป

5.1 ปัญหาและแนวทางแก้ไข

แบ่งเป็นปัญหาทางด้านฮาร์ดแวร์และซอฟต์แวร์

5.1.1 ปัญหาทางด้านฮาร์ดแวร์และแนวทางแก้ไข

จากการทดลองจะต้องมีการรับสัญญาณจากเอนโคเดอร์แต่เนื่องจากวงจรที่ใช้งานนั้นไม่สามารถรับสัญญาณได้เพราะรูปแบบของวงจรเอนโคเดอร์ในแขนกล Scorbol ER3 ได้นำขาบวกของเอนโคเดอร์ที่เป็นตัวส่งกับตัวรับ รวมเข้าด้วยกันแต่วงจรที่รับสัญญาณจากเอนโคเดอร์เอนโคเดอร์ที่เป็นตัวส่งกับตัวรับจะไม่นำขาบวกมารวมกัน วงจรเอนโคเดอร์ในแขนกล Scorbol ER 3 เป็นวงจรเฉพาะไม่สามารถนำมาใช้กับวงจรที่ใช้อยู่ซึ่งเป็นวงจรทั่วไปที่ใช้และเนื่องจากแขนกล Scorbol ER3 ไม่มีคู่มือมาให้ทำให้ไม่สามารถแก้ไขตรงนี้ได้ นอกจากจะทำลายวงจรใหม่ให้กับแขนกล Scorbol ER3

5.1.2 ปัญหาทางด้านซอฟต์แวร์และแนวทางแก้ไข

ในส่วนของโปรแกรมภาษาซีที่เขียนเป็นระบบกราฟิกส่วนนั้นจะเป็นการมองในลักษณะการเคลื่อนที่ของแขนกล Scorbol ER3 ใน 2 มิติถ้าสามารถเขียนเป็น 3 มิติได้ก็สามารถมองรายละเอียดการเคลื่อนที่ของแขนกล Scorbol ER3 ได้ดียิ่งขึ้น

5.2 แนวทางการพัฒนา

เนื่องจากโครงการนี้ได้ใช้คอมพิวเตอร์ควบคุมการทำงานของแขนกล Scorbol ER3 ซึ่งสามารถให้การควบคุมแขนกลแสดงผลเป็น 3 มิติได้ เหตุที่เป็น 3 มิติเพราะตอนออกแบบได้ใช้โปรแกรม 3 มิติมาช่วยในการออกแบบจึงได้เอาวัตถุ 3 มิติที่ออกแบบนั้นมาใช้ในการแสดงผลในการควบคุมซึ่งจะมีประโยชน์ต่อผู้ใช้งานคนอื่นที่สามารถเห็นรายละเอียดการทำงานได้ชัดเจน

5.3 บทสรุป

ในการใช้คอมพิวเตอร์เข้ามาควบคุมจะทำให้ระบบสามารถทำงานได้เป็นระบบมากขึ้นและง่ายต่อการควบคุม โดยจะต้องมีการสร้างสัญญาณอินเตอร์รัพต์ขึ้นมาเพื่อให้คอมพิวเตอร์กับฮาร์ดแวร์นั้นซิงโครไนซ์กัน ปริญญาโทฉบับนี้จะเป็นการศึกษาเกี่ยวกับการหาควบคุมแขนกล Scorbol ER3 โดยมีการแสดงผลในลักษณะ 2 มิติ ซึ่งทำให้ผู้ใช้งานสามารถมองลักษณะการเคลื่อนที่และปัญหาได้ง่ายขึ้นอีกทั้งยังเป็นการดึงดูความสนใจในการใช้งานด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้คงไม่อาจเสร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และร่วมมือจากหลาย ๆ ฝ่ายด้วยกัน บุคคลแรกที่ต้องกล่าวถึงเพราะเป็นส่วนสำคัญที่ทำให้วิทยานิพนธ์นี้เสร็จลงได้ก็คือ ดร.อุนนต์ พิณโสภณ และอ.อุดม จันทร์จรัสสุข อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความเอาใจใส่ แนะนำ และช่วยเหลือเสมอมา ซึ่งต้องขอขอบพระคุณเป็นอย่างมากและขอขอบชุมนุมอิเล็กทรอนิกส์ที่ให้ความเอื้อเฟื้อในการดำเนินงาน

และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูผู้เขียนมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมา ในทุก ๆ ด้านอันหาที่เปรียบมิได้ ข้าพเจ้าขอระลึกในพระคุณอันสุดประมาณ และขอกราบขอบพระคุณมา ณ ที่นี้

วัชรเดช ยี่มวัน

เอกสารอ้างอิง

- [1] อานาจ ทองแสน, “โปรแกรมและฐานข้อมูลสำหรับคอมพิวเตอร์ช่วยออกแบบและผลิต”,เทคโนโลยีคอมพิวเตอร์ช่วยออกแบบและผลิต, พิมพ์ครั้งที่ 1 , พ.ศ.2542
- [2] ประเสริฐ โรจน์สุธีวัฒน์, ”ดิจิทัลพลัสส์วิศวกรรมเครื่องกล”,เซมิคอนดักเตอร์ อิเล็กทรอนิกส์,ฉบับที่ 172, พ.ศ.2540
- [3] รศ.ดร. โยธิน เปรมปราณีรัชต์,วิเคราะห์และออกแบบระบบการควบคุมมอเตอร์,พิมพ์ครั้งที่1, พ.ศ.2528
- [4] รศ.สมยศ จุณณะปิยะ,การประยุกต์ใช้งานไมโครคอนโทรลเลอร์,พิมพ์ครั้งที่3,พ.ศ.2543
- [5] Mark W.Spong, M.Vidyasagar, Robot Dynamics and Control, Wiley, 1989
- [6] MOSHE SHOHAM,Fundamentals of Robotic,Eshed Robotec,1982



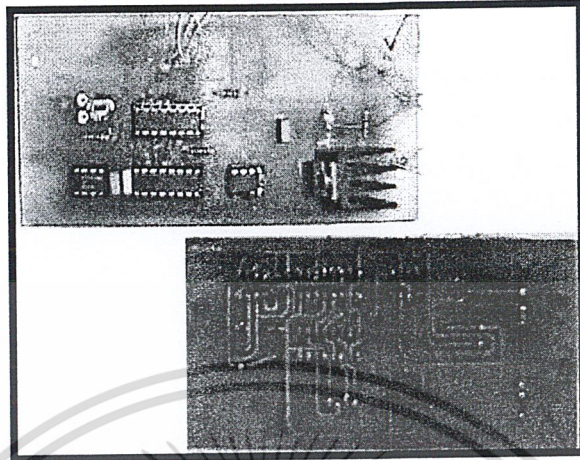
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



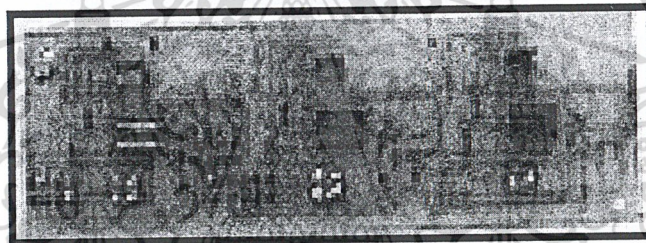
ภาคผนวก ก.

รูปแพ่งวงจรควบคุม

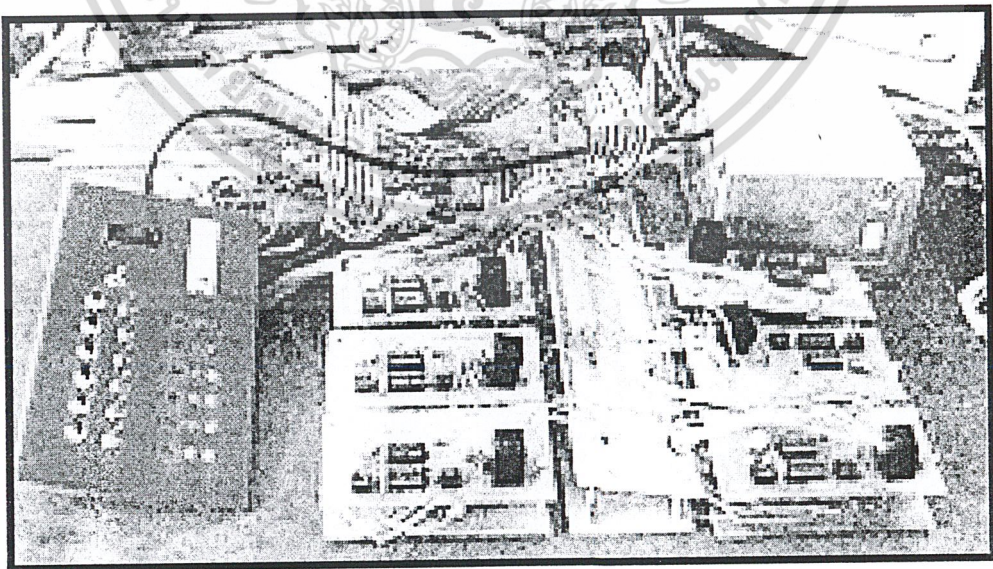
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.1 แสดงวงจรควบคุมความเร็วมอเตอร์



รูปที่ ก.2 แสดงวงจรที่รับค่าจากเอนโคเดอร์



รูปที่ ก.3 แสดงบอร์ดควบคุมแกนกล Scrobot ER3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข.
รายละเอียดโปรแกรมควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include<graphics.h>
#include<stdlib.h>
#include<conio.h>
#include<iostream.h>
#include<stdio.h>
#include<math.h>
#include<dos.h>
#include <time.h>

int      x1=160+80,x2=275+80,x3=390+80,x4=390+80,x5=390+80,x6=430+80,x7=430+80,
        y1=130-50,y2=130-50,y3=130-50,y4=150-50,y5=110-50,y6=150-50,y7=110-50,
        x8=100+80,x9=200+80,x10=160+80,x11=100+80,y8=50-50,y9=50-50,y10=130-
        50,y11=130-50,//base
        Max_y=280,i=1,ii=15,iii=14,x,y,z,start,fit_x,
        step=5,//angle turn when prees buttom
        s1,s2,s3,s4,s5,//limitswtich
        e1,e2,e3,e4,e5,e6,e7,e8,e9,e10,e11,e12,//Encoders
        on =4,color=15,colors=14,auto_color=14,clocks,color_clock,//red switch on
        L00=(x10-x11)/2,L0=(x9-x8)/2,xx1,xx2,
        ang0=0,//angle of Base + cw - ccw
        ang1=135,ang2=45,
        ang3=90,//angle rotate hand ccw cw
        ang3_1=0,
        ang4=0,//angle press release hand 0-25 degree
        limit_ang0=330,//+330 if > trun back
        limit_ang1=135,//+135 if > trun back
        limit_ang2=45,//+-45 trun back
        limit_ang3=360,//
        limit_ang4=25,// 0<=ang4<25
        ang0n,ang1n,ang2n,ang3n,ang4n;

long double  rad0,rad1,rad2,rad3,rad3_1,rad4,l3,x2_n,x3_n,x4_n,x5_n,x6_n,x7_n,
            x2_nn,x_n,y_n,Min_x3,
            rad1n,rad2n,search_x,search_y,
            x8_n,x9_n,x10_n,x11_n,
            y2_n,y3_n,y4_n,y5_n,y6_n,y7_n,A,B,C,

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Pi=3.14;

int GrDriver=DETECT,GrMode,GrError;

char ch,cho,time_man,round,
      choice='7',choices;//alway press buttom turn cw

time_t first, second;

void main()
{

```

```

clrscr();

void AUTO();

void MANUAL();

void BASE();

void LINK_1();
void LINK_2();
void LINK_3();
void BASEn();
void LINK_1n();
void LINK_2n();
void LINK_3n();
void RO_BASE();
void RO_LINK_1();
void RO_LINK_2();
void RO_LINK_3();
void LIMIT_SWITCH_1();
void LIMIT_SWITCH_2();
void LIMIT_SWITCH_3();
void LIMIT_SWITCH_4();
void LIMIT_SWITCH_5();

void ENCODER_1();
void ENCODER_2();
void ENCODER_3();
void ENCODER_4();
void ENCODER_5();
void ENCODER_6();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void ENCODER_7();
void ENCODER_8();
void ENCODER_9();
void ENCODER_10();
void ENCODER_11();
void ENCODER_12();
void CHECK_CARD();
CHECK_CARD();

RO_BASE();
BASEn();
RO_LINK_1();
LINK_1n();
RO_LINK_2();
LINK_2n();
RO_LINK_3();
LINK_3n();
setcolor(9);
line(0,380,500,380);//below show status encoders limit_switch
outtextxy(0,400,"Limit_Switch(on-off) for 1-5");
setlinestyle(SOLID_LINE,0,1);
setcolor(10);
LIMIT_SWITCH_1();
LIMIT_SWITCH_2();
LIMIT_SWITCH_3();
LIMIT_SWITCH_4();
LIMIT_SWITCH_5();
setcolor(9);
outtextxy(0,430,"Status of Encoders for 1-12");
setcolor(6);
outtextxy(255,440,"M1");
outtextxy(297,440,"M2");
outtextxy(344,440,"M3");
outtextxy(386,440,"M4");
outtextxy(434,440,"M5");

```

```

outtextxy(478,440,"M6");

setcolor(10);

ENCODER_10);
ENCODER_20);
ENCODER_30);
ENCODER_40);
ENCODER_50);
ENCODER_60);
ENCODER_70);
ENCODER_80);
ENCODER_90);
ENCODER_100);
ENCODER_110);
ENCODER_120);

/*****
outtextxy((x8_n+x9_n)/2-8,Max_y+5,"M1");
outtextxy(x10_n-16,Max_y-y10+8, "M2");
outtextxy(x2_n-16,Max_y-y2_n+8, "M3");
outtextxy(x3_n-45,Max_y-y3_n, "M4,M5");
outtextxy(x3_n+30,Max_y-y3_n-30, "M6");
setlinestyle(SOLID_LINE,0,3);

setcolor(9);

line(500,0,500,480);

setcolor(13);

outtextxy(510,20,"ROBOT_COMMANDER ");

setcolor(15);

outtextxy(510,40," EXIT  press 0 ");

outtextxy(510,60," AUTO  press 1 ");

outtextxy(510,80," MANUAL press 2 ");

while(ch!='0')
{

ch=getch();

switch(ch)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        case'1':AUTO();break;
        case'2':MANUAL();break;
    }
    settextstyle(8,0,1);
    first = time(NULL);
    time(&first);
    ctime(&first);
    setcolor(14);
    outtextxy(60,340,ctime(&first));
    settextstyle(0,0,1);
}
closegraph();
}
void CHECK_CARD()
{
    initgraph(&GrDriver,&GrMode,"F:\\WAT\\TC\\bgi");
    GrError=graphresult();
    if (GrError!=0)
    {
        clrscr();
        cout<<"\a GRAHPIC ERROR : "<<grapherrormsg(GrError);
        getch();
        exit(GrError);
    }
}

void BASE()
{
    void CHECK_CARD();
    CHECK_CARD();
    setlinestyle(SOLID_LINE,0,3);
    setcolor(15);
    line(x8,Max_y-8,x9,Max_y-9);
    line(x9,Max_y-9,x10,Max_y-10);
    line(x10,Max_y-10,x11,Max_y-11);
}

```

```

        line(x11,Max_y-y11,x8,Max_y-y8);
        setcolor(14);
        circle(x10,Max_y-y10,6);
    }
void LINK_10()
{
    setlinestyle(SOLID_LINE,0,3);
    setcolor(15);
    line(x1,Max_y-y1,x2,Max_y-y2);
    setcolor(14);
    circle(x2,Max_y-y2,6);
}
void LINK_20()
{
    setlinestyle(SOLID_LINE,0,3);
    setcolor(15);
    line(x2,Max_y-y2,x3,Max_y-y3);
    setcolor(14);
    circle(x3,Max_y-y3,4);
}
void LINK_30()
{
    setlinestyle(SOLID_LINE,0,3);
    setcolor(15);
    line(x4,Max_y-y4,x5,Max_y-y5);
    line(x4,Max_y-y4,x6,Max_y-y6);
    line(x5,Max_y-y5,x7,Max_y-y7);
    setcolor(14);
    circle(x4,Max_y-y4,2);
    circle(x5,Max_y-y5,2);
}
void RO_BASE()
{

```

```

    rad0=((ang0)*Pi)/180;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if((ang0>=0)&&(ang0<=90))
{
x8_n=x8+(L0*cos(1.57-rad0));
x9_n=x9-(L0*cos(1.57-rad0));
x10_n=x10-(L00*cos(1.57-rad0));
x11_n=x11+(L00*cos(1.57-rad0));
}
else if((ang0>90)&&(ang0<=270))
{
x8_n=(-1*L0*cos(rad0))+x8+L0;
x9_n=(L0*cos(rad0))+x9-L0;
x10_n=(L00*cos(rad0))+x10-L00+40;
x11_n=(-1*L00*cos(rad0))+x11+L00+40;
}
else if((ang0>270)&&(ang0<=360))
{
x8_n=x8+(L0*cos(rad0));
x9_n=x9-(L0*cos(rad0));
x10_n=x10-(L00*cos(rad0));
x11_n=x11+(L00*cos(rad0));
}
}
void RO_LINK_1()
{
rad1=(ang1*Pi)/180;
y2_n=y1+(115*sin(rad1));
x2_n=x10_n+115*cos(rad1)*cos(rad0);
}
void RO_LINK_2()
{
rad2=(ang2*Pi)/180;
x3_n=x2_n+(115*cos(rad2)*cos(rad0));
y3_n=y2_n+(115*sin(rad2));
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void RO_LINK_3()
{
    rad3=(ang3*Pi)/180;
    rad4=(ang4*Pi)/180;
    x4_n=x3_n+(20*cos(rad2+rad3)*(cos(rad0)));
    y4_n=y3_n+(20*sin(rad2+rad3));
    x5_n=x3_n-(20*cos(rad2+rad3)*(cos(rad0)));
    y5_n=y3_n-(20*sin(rad2+rad3));
    x6_n=x4_n+(40*cos(rad2-rad4)*(cos(rad0)));
    y6_n=y4_n+(40*sin(rad2-rad4));
    x7_n=x5_n+(40*cos(rad2+rad4)*(cos(rad0)));
    y7_n=y5_n+(40*sin(rad2+rad4));
}

```

```

void BASEn()
{
    setlinestyle(SOLID_LINE,0,3);
    setcolor(ii);
    line(x8_n,Max_y-y8,x9_n,Max_y-y9);
    line(x9_n,Max_y-y9,x10_n,Max_y-y10);
    line(x10_n,Max_y-y10,x11_n,Max_y-y11);
    line(x11_n,Max_y-y11,x8_n,Max_y-y8);
    setcolor(iii);
    circle(x10_n,Max_y-y10,6);
}

```

```

void LINK_1n()
{
    setlinestyle(SOLID_LINE,0,3);
    setcolor(ii);
    line(x10_n,Max_y-y10,x2_n,Max_y-y2_n);
    setcolor(iii);
    circle(x2_n,Max_y-y2_n,6);
}

```

```

void LINK_2n()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        setlinestyle(SOLID_LINE,0,3);
        setcolor(ii);
        line(x2_n,Max_y-y2_n,x3_n,Max_y-y3_n);
        setcolor(iii);
        circle(x3_n,Max_y-y3_n,4);
    }
void LINK_3n()

```

```

    {
        setlinestyle(SOLID_LINE,0,3);
        setcolor(ii);
        line(x4_n,Max_y-y4_n,x5_n,Max_y-y5_n);
        line(x4_n,Max_y-y4_n,x6_n,Max_y-y6_n);
        line(x5_n,Max_y-y5_n,x7_n,Max_y-y7_n);
        setcolor(iii);
        circle(x4_n,Max_y-y4_n,2);
        circle(x5_n,Max_y-y5_n,2);
    }

```

```

void LIMIT_SWITCH_1()
    {
        setfillstyle(1,0);
        pieslice(250,400,0,360,6);
    }

```

```

void LIMIT_SWITCH_2()
    {
        setfillstyle(1,s1);
        pieslice(270,400,0,360,6);
    }

```

```

void LIMIT_SWITCH_3()
    {
        setfillstyle(1,s1);
        pieslice(290,400,0,360,6);
    }

```

```

void LIMIT_SWITCH_4()

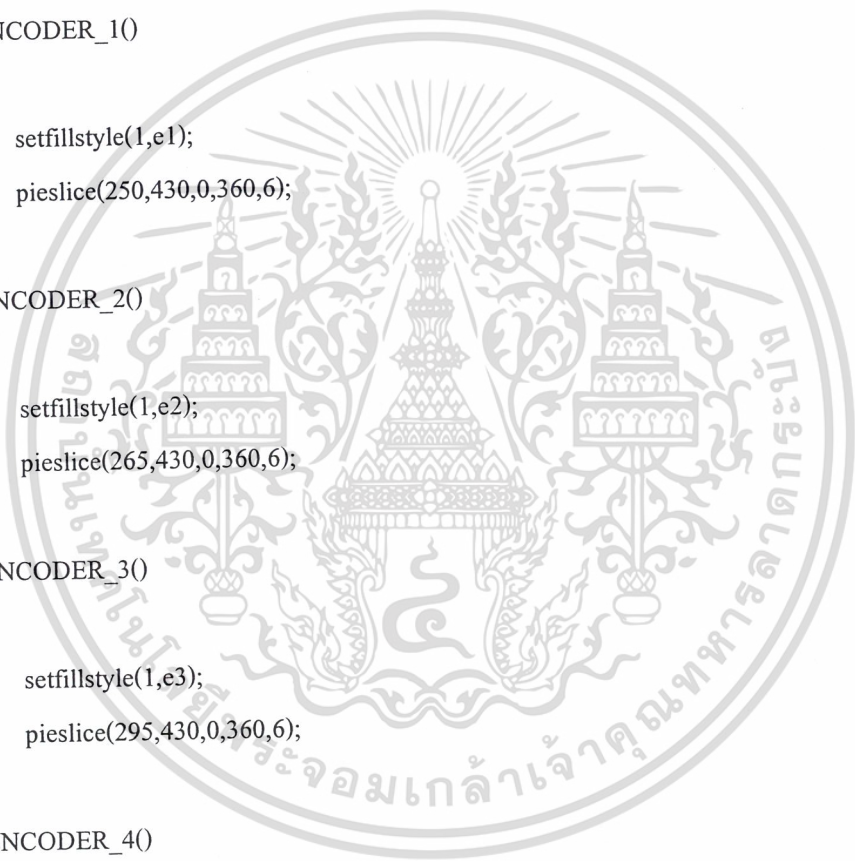
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        setfillstyle(1,s1);
        pieslice(310,400,0,360,6);
    }
void LIMIT_SWITCH_5()
    {
        setfillstyle(1,s1);
        pieslice(330,400,0,360,6);
    }
void ENCODER_1()
    {
        setfillstyle(1,e1);
        pieslice(250,430,0,360,6);
    }
void ENCODER_2()
    {
        setfillstyle(1,e2);
        pieslice(265,430,0,360,6);
    }
void ENCODER_3()
    {
        setfillstyle(1,e3);
        pieslice(295,430,0,360,6);
    }
void ENCODER_4()
    {
        setfillstyle(1,e4);
        pieslice(310,430,0,360,6);
    }
void ENCODER_5()
    {
        setfillstyle(1,e5);
        pieslice(340,430,0,360,6);
    }

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void ENCODER_6()
{
    setfillstyle(1,e6);
    pieslice(355,430,0,360,6);
}

void ENCODER_7()
{
    setfillstyle(1,e7);
    pieslice(385,430,0,360,6);
}

void ENCODER_8()
{
    setfillstyle(1,e8);
    pieslice(400,430,0,360,6);
}

void ENCODER_9()
{
    setfillstyle(1,e9);
    pieslice(430,430,0,360,6);
}

void ENCODER_10()
{
    setfillstyle(1,e10);
    pieslice(445,430,0,360,6);
}

void ENCODER_11()
{
    setfillstyle(1,e11);
    pieslice(475,430,0,360,6);
}

void ENCODER_12()
{
    setfillstyle(1,e12);
    pieslice(490,430,0,360,6);
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
```

```
void MANUAL()
```

```
{
```

```
void BASEn();
```

```
void LINK_1n();
```

```
void LINK_2n();
```

```
void LINK_3n();
```

```
void RO_BASE();
```

```
void RO_LINK_1();
```

```
void RO_LINK_2();
```

```
void RO_LINK_3();
```

```
void RO_BASEn();
```

```
void RO_LINK_1n();
```

```
void RO_LINK_2n();
```

```
void RO_LINK_3n();
```

```
setcolor(14);
```

```
outtextxy(502,100,"EXIT Maunul key 0 ");
```

```
setcolor(15);
```

```
outtextxy(510,120, " Turn M1 key 1");
```

```
outtextxy(510,140, " Turn M2 key 2");
```

```
outtextxy(510,160, " Turn M3 key 3");
```

```
outtextxy(510,180, " Turn M4 key 4");
```

```
outtextxy(510,200, " Turn M5 key 5");
```

```
outtextxy(510,220, " Turn M6 key 6");
```

```
outtextxy(510,240, " Turn CW key 7");
```

```
outtextxy(510,260, " Turn CCW key 8");
```

```
cho='7';
```

```
choice='7';
```

```
while(choice!='0')
```

```
{
```

```
setcolor(14);
```

```
if(cho=='8')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

clocks=clocks-step;
if((clocks<=0)&&(clocks>=-360))
{
    if (clocks===-360)
    {
        setcolor(0);
        outtextxy(400,40,&round);
        round=(char)(i+48);
        setcolor(14);
        outtextxy(400,40,&round);
        setcolor(0);
        outtextxy(408,40,"1");
        setcolor(14);
        clocks=0;
        i=i+1;
    }
    else;
    setcolor(i);
    if(i==10) i=1; else;
    pieslice(440,40,0,clocks,20);
}
}
else if(cho=='7')
{
    clocks=clocks+step;
    if((clocks>=0)&&(clocks<=360))
    {
        if (clocks===360)
        {
            setcolor(0);
            outtextxy(400,40,&round);
            round=(char)(i+48);
            setcolor(14);
            outtextxy(400,40,&round);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setcolor(0);
outtextxy(408,40,"1");
setcolor(14);
clocks=0;
i=i+1;
}
else;
setcolor(i);
if(i==10) i=1; else;
pieslice(440,40,0,clocks,20);
}
}
if(choice=='7')
cho='7';
else if(choice=='8')
cho='8';
choice=getch();
ang0n=ang0;
ang1n=ang1;
ang2n=ang2;
ang3n=ang3;
ang4n=ang4;
ii=0;iii=0;
setcolor(ii);
outtextxy((x8_n+x9_n)/2-8,Max_y+5,"M1");
outtextxy(x10_n-16,Max_y-y10+8, "M2");
outtextxy(x2_n-16,Max_y-y2_n+8, "M3");
outtextxy(x3_n-45,Max_y-y3_n, "M4,M5");
outtextxy(x3_n+30,Max_y-y3_n-30, "M6");
RO_BASEn();
RO_LINK_1n();
RO_LINK_2n();
RO_LINK_3n();

```

เอกสารนี้เป็นเอกสารทสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

BASEn();
LINK_1n();
LINK_2n();
LINK_3n();
switch(choice)
{
    case'0':setcolor(0);
        outtextxy(502,100,"EXIT Maunul key 0 ");
        outtextxy(510,120, " Turn M1 key 1");
        outtextxy(510,140, " Turn M2 key 2");
        outtextxy(510,160, " Turn M3 key 3");
        outtextxy(510,180, " Turn M4 key 4");
        outtextxy(510,200, " Turn M5 key 5");
        outtextxy(510,220, " Turn M6 key 6");
        outtextxy(510,240, " Turn CW key 7");
        outtextxy(510,260, " Turn CCW key 8");
        break;
    case'1':if(cho=='7') ang0=ang0+step; else if(cho=='8') ang0=ang0-step;break;
    case'2':if(cho=='7') ang1=ang1+step; else if(cho=='8') ang1=ang1-step;break;
    case'3':if(cho=='7') ang2=ang2+step; else if(cho=='8') ang2=ang2-step;break;
    case'4':if(cho=='7') ang3=ang3+step; else if(cho=='8') ang3=ang3-step;break;
    case'5':if(cho=='7') ang3=ang3-step; else if(cho=='8') ang3=ang3+step;break;
    case'6':if(cho=='7') ang4=ang4+step; else if(cho=='8') ang4=ang4-step;break;
    case'7':break;
    case'8':break;
    default:cout<<"\a";
}
settextstyle(8,0,1);
setcolor(0);
outtextxy(60,340,ctime(&second));
setcolor(13);
second=time(NULL);
time(&second);
ctime(&second);

```

```

outtextxy(60,340,ctime(&second));
settextstyle(0,0,1);
ii=15;iii=14;
RO_BASE();
RO_LINK_1();
RO_LINK_2();
RO_LINK_3();
BASEn();
LINK_1n();
LINK_2n();
LINK_3n();
setcolor(iii);
outtextxy((x8_n+x9_n)/2-8,Max_y+5,"M1");
outtextxy(x10_n-16,Max_y-y10+8, "M2");
outtextxy(x2_n-16,Max_y-y2_n+8, "M3");
outtextxy(x3_n-45,Max_y-y3_n, "M4,M5");
outtextxy(x3_n+30,Max_y-y3_n-30, "M6");
}
}
void AUTO()
{
void BASEn();
void LINK_1n();
void LINK_2n();
void LINK_3n();
void RO_BASE();
void RO_LINK_1();
void RO_LINK_2();
void RO_LINK_3();
void SEARCH();
void TEXT_AUTO();
setcolor(auto_color);
outtextxy(502,280,"EXIT AUTO key 0 ");
outtextxy(502,300," x,y,z key 1 ");

```

```

choices='7';
while(choices!='0')
{
    choices=getch();
    switch(choices)
    {
        case '0':setcolor(0);
            outtextxy(502,280,"EXIT AUTO key 0 ");
            outtextxy(502,300," x,y,z key 1 ");
            outtextxy(510,320," Wait update ");
            break;

        case '1':SEARCH();break;
    }
    RO_LINK_2();
    RO_LINK_3();
    LINK_2n();
    LINK_3n();
}
}
void SEARCH()
{
    setcolor(auto_color);
    outtextxy(510,320," Wait update ");
    /*cin>>x>>y;
    x=x-x10_n;
    y=y-y10;
    rad2=(((x*x)+(y*y)-2*(115*115))/(115*115));
    rad2=acos(rad2);
    ang2=(rad2*180)/Pi;
    for(xx1=0;xx1<=360;xx1++)
    {
        rad1n=(xx1*Pi)/180;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if((search_y>=y-5)&&(search_y<=y+5))
            angl=xx1;
    }
    cout<<angl<<endl<<ang2; */
}

/***** white to black */

void RO_BASEn()
{
    rad0=((ang0n)*Pi)/180;
    if((ang0n>=0)&&(ang0n<=90))
    {
        x8_n=x8+(L0*cos(1.57-rad0));
        x9_n=x9-(L0*cos(1.57-rad0));
        x10_n=x10-(L0*cos(1.57-rad0));
        x11_n=x11+(L0*cos(1.57-rad0));
    }
    else if((ang0n>90)&&(ang0n<=270))
    {
        x8_n=(-1*L0*cos(rad0))+x8+L0;
        x9_n=(L0*cos(rad0))+x9-L0;
        x10_n=(L0*cos(rad0))+x10-L0+40;
        x11_n=(-1*L0*cos(rad0))+x11+L0+40;
    }
    else if((ang0n>270)&&(ang0n<=360))
    {
        x8_n=x8+(L0*cos(rad0));
        x9_n=x9-(L0*cos(rad0));
        x10_n=x10-(L0*cos(rad0));
        x11_n=x11+(L0*cos(rad0));
    }
}

void RO_LINK_1n()
{

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        y2_n=y1+(115*sin(rad1));
        x2_n=x10_n+115*cos(rad1)*cos(rad0);
    }
void RO_LINK_2n()
{
    rad2=(ang2n*Pi)/180;
    x3_n=x2_n+(115*cos(rad2)*cos(rad0));
    y3_n=y2_n+(115*sin(rad2));
}
void RO_LINK_3n()
{
    rad3=(ang3n*Pi)/180;
    rad4=(ang4n*Pi)/180;
    x4_n=x3_n+(20*cos(rad2+rad3)*(cos(rad0)));
    y4_n=y3_n+(20*sin(rad2+rad3));
    x5_n=x3_n-(20*cos(rad2+rad3)*(cos(rad0)));
    y5_n=y3_n-(20*sin(rad2+rad3));
    x6_n=x4_n+(40*cos(rad2-rad4)*(cos(rad0)));
    y6_n=y4_n+(40*sin(rad2-rad4));
    x7_n=x5_n+(40*cos(rad2+rad4)*(cos(rad0)));
    y7_n=y5_n+(40*sin(rad2+rad4));
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก.

DATA SHEET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM139/LM239/LM339/LM2901/LM3302 Low Power Low Offset Voltage Quad Comparators

General Description

The LM139 series consists of four independent precision voltage comparators with an offset voltage specification as low as 2 mV max for all four comparators. These were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage. These comparators also have a unique characteristic in that the input common-mode voltage range includes ground, even though operated from a single power supply voltage.

Application areas include limit comparators, simple analog to digital converters; pulse, squarewave and time delay generators; wide range VCO; MOS clock timers; multivibrators and high voltage digital logic gates. The LM139 series was designed to directly interface with TTL and CMOS. When operated from both plus and minus power supplies, they will directly interface with MOS logic— where the low power drain of the LM339 is a distinct advantage over standard comparators.

Advantages

- High precision comparators
- Reduced V_{OS} drift over temperature

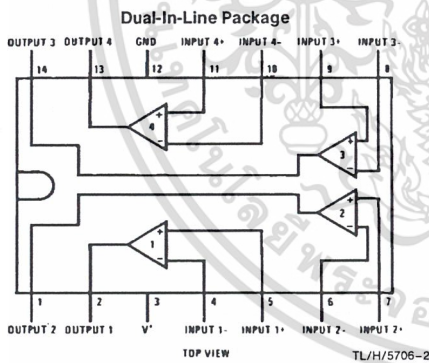
- Eliminates need for dual supplies
- Allows sensing near GND
- Compatible with all forms of logic
- Power drain suitable for battery operation

Features

- Wide supply voltage range
 - LM139 series, $2 V_{DC}$ to $36 V_{DC}$ or $\pm 1 V_{DC}$ to $\pm 18 V_{DC}$
 - LM139A series, LM2901 $2 V_{DC}$ to $28 V_{DC}$
 - LM3302 or $\pm 1 V_{DC}$ to $\pm 14 V_{DC}$
- Very low supply current drain (0.8 mA) — independent of supply voltage
- Low input biasing current 25 nA
- Low input offset current ± 5 nA
- Low input offset voltage ± 3 mV
- Input common-mode voltage range includes GND
- Differential input voltage range equal to the power supply voltage
- Low output saturation voltage 250 mV at 4 mA
- Output voltage compatible with TTL, DTL, ECL, MOS and CMOS logic systems

LM139/LM239/LM339/LM2901/LM3302
Low Power Low Offset Voltage Quad Comparators

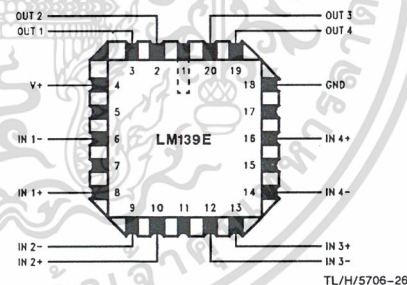
Connection Diagrams



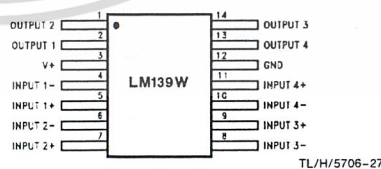
Order Number LM139J, LM139J/883*, LM139AJ,
LM139AJ/883**, LM239J, LM239AJ, LM339J,
See NS Package Number J14A
Order Number LM339AM, LM339M or LM2901M
See NS Package Number M14A
Order Number LM339N, LM339AN,
LM2901N or LM3302N
See NS Package Number N14A

*Available per JM38510/11201

**Available per SMD# 5962-8873901



Order Number LM139AE/883 or LM139E/883
See NS Package Number E20A



Order Number LM139AW/883 or LM139W/883*
See NS Package Number W14B

• Application Hints

The LM139 series are high gain, wide bandwidth devices which, like most comparators, can easily oscillate if the output lead is inadvertently allowed to capacitively couple to the inputs via stray capacitance. This shows up only during the output voltage transition intervals as the comparator changes states. Power supply bypassing is not required to solve this problem. Standard PC board layout is helpful as it reduces stray input-output coupling. Reducing this input resistors to $< 10\text{ k}\Omega$ reduces the feedback signal levels and finally, adding even a small amount (1 to 10 mV) of positive feedback (hysteresis) causes such a rapid transition that oscillations due to stray feedback are not possible. Simply socketing the IC and attaching resistors to the pins will cause input-output oscillations during the small transition intervals unless hysteresis is used. If the input signal is a pulse waveform, with relatively fast rise and fall times, hysteresis is not required.

All pins of any unused comparators should be grounded.

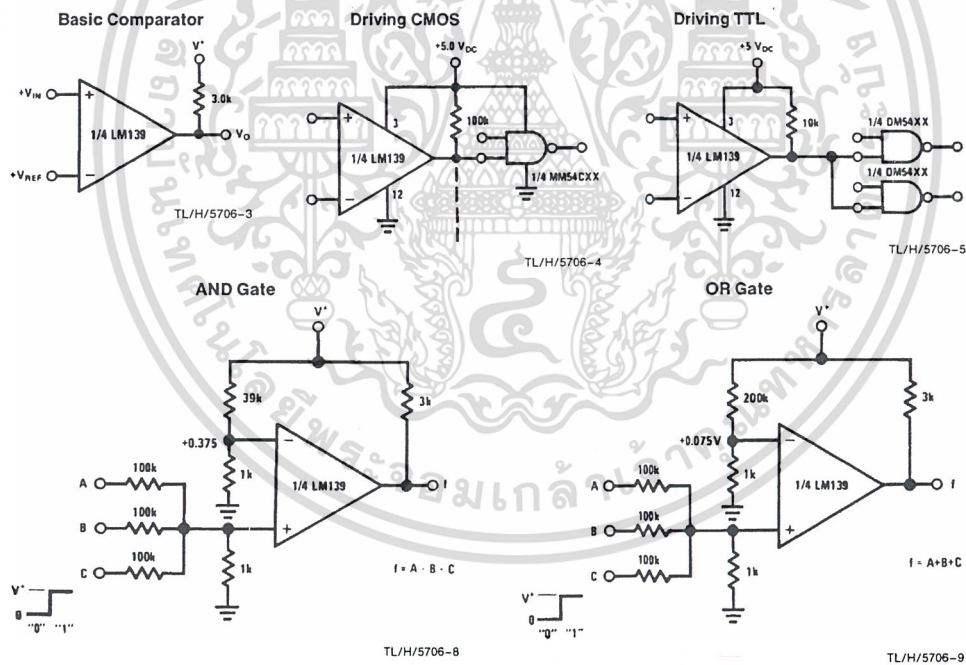
The bias network of the LM139 series establishes a drain current which is independent of the magnitude of the power supply voltage over the range of from $2 V_{DC}$ to $30 V_{DC}$.

It is usually unnecessary to use a bypass capacitor across the power supply line.

The differential input voltage may be larger than V^+ without damaging the device. Protection should be provided to prevent the input voltages from going negative more than $-0.3 V_{DC}$ (at 25°C). An input clamp diode can be used as shown in the applications section.

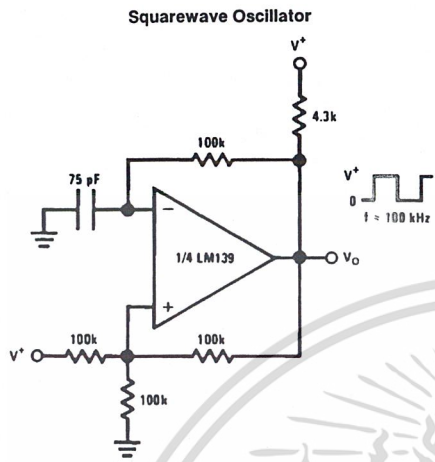
The output of the LM139 series is the uncommitted collector of a grounded-emitter NPN output transistor. Many collectors can be tied together to provide an output OR'ing function. An output pull-up resistor can be connected to any available power supply voltage within the permitted supply voltage range and there is no restriction on this voltage due to the magnitude of the voltage which is applied to the V^+ terminal of the LM139A package. The output can also be used as a simple SPST switch to ground (when a pull-up resistor is not used). The amount of current which the output device can sink is limited by the drive available (which is independent of V^+) and the β of this device. When the maximum current limit is reached (approximately 16 mA), the output transistor will come out of saturation and the output voltage will rise very rapidly. The output saturation voltage is limited by the approximately $60\Omega R_{SAT}$ of the output transistor. The low offset voltage of the output transistor (1 mV) allows the output to clamp essentially to ground level for small load currents.

Typical Applications ($V^+ = 5.0 V_{DC}$)

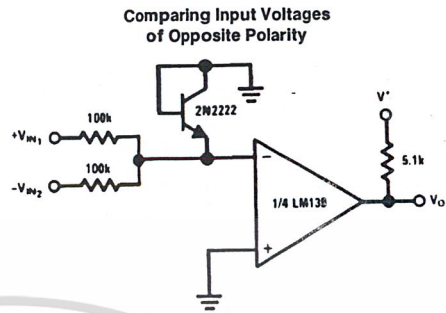


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

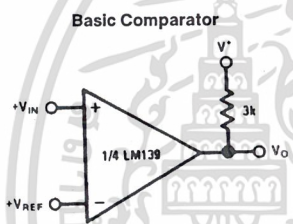
Typical Applications ($V^+ = 15\text{ V}_{\text{DC}}$) (Continued)



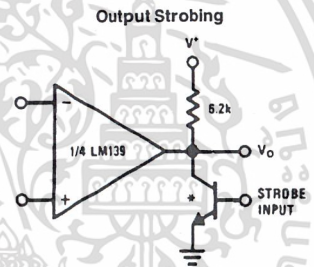
TL/H/5706-16



TL/H/5706-20

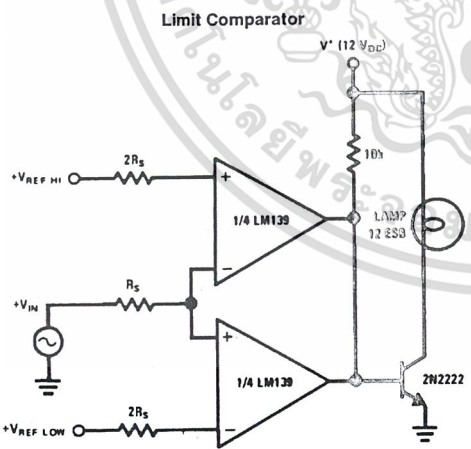


TL/H/5706-21

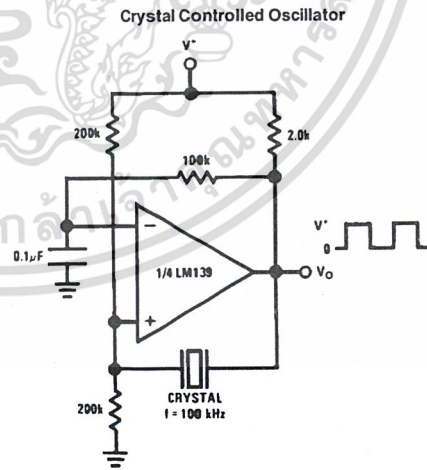


TL/H/5706-22

*Or open-collector logic gate without pull-up resistor



TL/H/5706-24

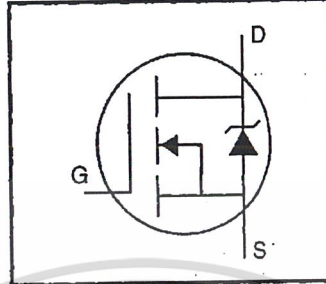


TL/H/5706-25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HEXFET® Power MOSFET

- Dynamic dv/dt Rating
- Repetitive Avalanche Rated
- 175°C Operating Temperature
- Fast Switching
- Ease of Paralleling
- Simple Drive Requirements



$$V_{DSS} = 100V$$

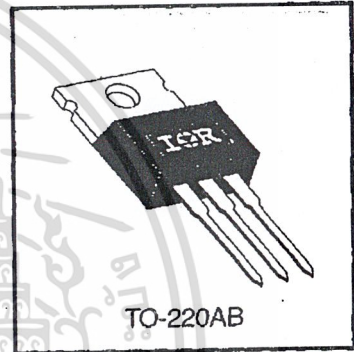
$$R_{DS(on)} = 0.54\Omega$$

$$I_D = 5.6A$$

Description

Third Generation HEXFETs from International Rectifier provide the designer with the best combination of fast switching, ruggedized device design, low on-resistance and cost-effectiveness.

The TO-220 package is universally preferred for all commercial-industrial applications at power dissipation levels to approximately 50 watts. The low thermal resistance and low package cost of the TO-220 contribute to its wide acceptance throughout the industry.



DATA SHEETS

Absolute Maximum Ratings

	Parameter	Max.	Units
$I_D @ T_C = 25^\circ C$	Continuous Drain Current, $V_{GS} @ 10 V$	5.6	A
$I_D @ T_C = 100^\circ C$	Continuous Drain Current, $V_{GS} @ 10 V$	4.0	
I_{DM}	Pulsed Drain Current ①	20	
$P_D @ T_C = 25^\circ C$	Power Dissipation	43	W
	Linear Derating Factor	0.29	W/°C
V_{GS}	Gate-to-Source Voltage	± 20	V
E_{AS}	Single Pulse Avalanche Energy ②	100	mJ
I_{AR}	Avalanche Current ①	5.6	A
E_{AR}	Repetitive Avalanche Energy ①	4.3	mJ
dv/dt	Peak Diode Recovery dv/dt ③	5.5	V/ns
T_J T_{STG}	Operating Junction and Storage Temperature Range	-55 to +175	°C
	Soldering Temperature, for 10 seconds	300 (1.6mm from case)	
	Mounting Torque, 6-32 or M3 screw	10 lbf·in (1.1 N·m)	

Thermal Resistance

	Parameter	Min.	Typ.	Max.	Units
$R_{\theta JC}$	Junction-to-Case	—	—	3.5	°C/W
$R_{\theta CS}$	Case-to-Sink, Flat, Greased Surface	—	0.50	—	
$R_{\theta JA}$	Junction-to-Ambient	—	—	62	