

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

มือกล

ROBOTIC HAND



โดย

นายนรินทร์ ตันตราธิษฐาน
นายพงศธร มณีวัฒนา
นายรัฐไกร รักชนม์
นายอชิป ราษฎร์ชัย

เลขหน้.....
เลขทะเบียน.....33958..
วัน, เดือน, ปี 23 ก.ย. 2542

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมระบบควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มือกล
ROBOTIC HAND

โดย

นายณรินทร์	ตันตราธิษฐาน	38014227
นายพงศธร	มณีวัฒนา	38014307
นายรัฐไกร	รักชนม์	38014415
นายอธิป	กาญจนา	38014614

อาจารย์ที่ปรึกษา

อาจารย์สุมิตร พนาอุดมทรัพย์

ปฏิญญาฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมระบบควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ ปีการศึกษา 2541

ภาควิชาวิศวกรรมระบบควบคุม


คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง มือกล

ROBOTIC HAND

ผู้จัดทำ

- | | | |
|---------------|--------------|----------|
| 1. นายนรินทร์ | ตันตราธิษฐาน | 38014227 |
| 2. นายพงศธร | มณีวัฒนา | 38014307 |
| 3. นายรัฐไกร | รักชนม์ | 38014415 |
| 4. นายอชิป | กาญจน์ชัย | 38014614 |


..... อาจารย์ที่ปรึกษา
(อาจารย์สุมิตร พนาอุดมทรัพย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มือกล

นรินทร์ ตันตราธิษฐาน

พงศธร มณีวัฒนา

รัฐไกร รักชนม์

อธิป ราญมีชัย

อ. สุมิตร พนาอุดมทรัพย์

อาจารย์ที่ปรึกษา

ปีการศึกษา 2541

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการศึกษา ออกแบบและจัดสร้างมือกลแบบมีหลายนิ้วมือ ที่สามารถกำหนดรูปแบบการเคลื่อนที่ของมือกลได้ด้วยคอมพิวเตอร์ รายงานฉบับนี้ได้นำเสนอการจัดสร้างส่วนประกอบทางกล เช่น แขน มือ นิ้วมือ และวงจรควบคุมการทำงานต่างๆ เช่น วงจรขับมอเตอร์ วงจรเชื่อมต่ออุปกรณ์ภายนอกกับคอมพิวเตอร์ จากนั้นประกอบขึ้นเป็นโครงสร้างมือกลและทำการควบคุมด้วยคอมพิวเตอร์

Abstract

This thesis concerns a design and construction for a multifingered robotic hand that can be controlled with a prototype movement by the computer. The report proposes mechanism of robotic hand such as fingers, hand, arm and electronic parts such as interface circuit and drive control hardware. All parts are assembled and controlled afterwards.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อ	I
สารบัญ	II
สารบัญรูปภาพ	IV
สารบัญตาราง	VI
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	2
2.1 ทฤษฎีพื้นฐานของการวิเคราะห์และควบคุมหุ่นยนต์	2
2.1.1 การแบ่งแยกประเภทหุ่นยนต์	2
2.1.2 ลักษณะเฉพาะของหุ่นยนต์	6
2.1.3 Direct Kinematics : สัมการแขน	10
2.1.4 Inverse Kinematics : การแก้สัมการแขน	25
2.2 หลักการในการควบคุม	27
2.3 โครงสร้างของระบบ	27
2.4 วงจรเชื่อมต่อระหว่างวงจรควบคุมกับคอมพิวเตอร์	28
2.5 หลักการทำงานของมอเตอร์ไฟฟ้ากระแสตรง	28
2.5.1 การแยกประเภทของมอเตอร์ไฟฟ้ากระแสตรง	29
2.5.2 พื้นฐานของระบบควบคุมมอเตอร์ไฟฟ้ากระแสตรง	30
2.5.3 ลักษณะการควบคุมของระบบมอเตอร์ไฟฟ้ากระแสตรง	31
บทที่ 3 โครงสร้างและการออกแบบส่วนอิเล็กทรอนิกส์	32
3.1 วงจรอินเทอร์เฟส	32
3.2 การส่งสัญญาณควบคุม	33
3.3 การรับสัญญาณข้อมูล	34
3.4 วงจรขับมอเตอร์	36
3.5 วงจรรวมควบคุมมือถุ	37
บทที่ 4 โครงสร้างและการออกแบบทางกล	39
4.1 การออกแบบโครงสร้างส่วนนี้	39
4.2 การออกแบบโครงสร้างส่วนมือ	39
4.3 การออกแบบโครงสร้างส่วนแขน	40

เอกสารนี้ 4.2 การออกแบบโครงสร้างส่วนมือ เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 4.3 การออกแบบโครงสร้างส่วนแขน ไม่ผ่านการคัดค้านลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การออกแบบโครงสร้างส่วนฐาน	40
บทที่ 5 สรุปและวิเคราะห์ผลการดำเนินงาน	51
5.1 ผลการดำเนินงาน	51
5.2 สรุปผลการดำเนินงาน	52
5.3 แนวทางในการพัฒนา	53
ภาคผนวก ก.	54
ภาคผนวก ข.	60
บรรณานุกรม	
กิตติกรรมประกาศ	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่ 2.1	หุ่นยนต์พิกัดคาร์ทีเซียน	3
รูปที่ 2.2	หุ่นยนต์พิกัดทรงกระบอกล	4
รูปที่ 2.3	หุ่นยนต์พิกัดทรงกลม	4
รูปที่ 2.4	หุ่นยนต์ SCARA	5
รูปที่ 2.5	หุ่นยนต์ข้อต่อหมุน	5
รูปที่ 2.6	ระยะที่เชื่อมถึงและสโตรคของหุ่นยนต์ทรงกระบอกล	7
รูปที่ 2.7	yaw-pitch-roll ของ tool	7
รูปที่ 2.8	ลำดับการหมุนแกน	8
รูปที่ 2.9	ระบบ YPR	9
รูปที่ 2.10	ระบบ RPY	9
รูปที่ 2.11	แขนกลที่มีลักษณะเป็น links	11
รูปที่ 2.12	โครงพิกัดฉากใน R^3	12
รูปที่ 2.13	พิกัดของ p เทียบกับโครงพิกัด X	14
รูปที่ 2.14	การหมุนพื้นฐานใน R^3	15
รูปที่ 2.15	การหมุนของ M รอบ f^1 ด้วยมุม ϕ	15
รูปที่ 2.16	yaw-pitch-roll ของ tool	17
รูปที่ 2.17	มุมข้อต่อ θ และระยะข้อต่อ d	21
รูปที่ 2.18	ความยาว link และมุมบิด	22
รูปที่ 2.19	ตำแหน่งและการหมุนของเครื่องมือในพิกัดฐาน	25
รูปที่ 2.20	Direct และ Inverse Kinematics	25
รูปที่ 2.21	บล็อกไดอะแกรมโดยรวมของระบบควบคุมการทำงานของมือกล	27
รูปที่ 2.22	วงจรควบคุมการทำงาน	28
รูปที่ 3.1	IBM PC system bus	32
รูปที่ 3.2	การปรับตั้ง DIP Switch	33
รูปที่ 3.3	ADC0808 และขาที่ใช้งาน	34
รูปที่ 3.4	วงจร Schmitt Trigger กำเนิดพัลส์ความถี่ 600 MHz	34
รูปที่ 3.5	วงจรรวมบนการ์ดอินเทอร์เฟซ	35
รูปที่ 3.6	วงจรรับคลื่นและควบคุมมอเตอร์	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 หมายความว่าใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.7	วงจรรวมควบคุมมือกล	38
รูปที่ 4.1	ลักษณะและโครงสร้างของนิ้วมือ	40
รูปที่ 4.2	ลักษณะและโครงสร้างของนิ้วมือด้านหน้า (Front view)	41
รูปที่ 4.3	ลักษณะและโครงสร้างของนิ้วมือด้านข้าง (Side view)	41
รูปที่ 4.4	ลักษณะของการเคลื่อนที่ของนิ้วมือ	42
รูปที่ 4.5	ลักษณะและโครงสร้างของมือ	42
รูปที่ 4.6	ลักษณะและโครงสร้างของแขน	43
รูปที่ 4.7	ลักษณะและโครงสร้างของส่วนฐาน	43
รูปที่ 4.8	ลักษณะและโครงสร้างโดยรวม	44
รูปที่ 4.9	แบบโครงสร้างของนิ้วชี้ , นิ้วกลาง , นิ้วนางและนิ้วก้อย	45
รูปที่ 4.10	แบบโครงสร้างด้านข้างและด้านบนของนิ้ว	46
รูปที่ 4.11	แบบโครงสร้างด้านข้างและด้านบนของนิ้วโป้ง	47
รูปที่ 4.12	แบบโครงสร้างของฝ่ามือ	48
รูปที่ 4.13	แบบโครงสร้างของแขน	49
รูปที่ 4.14	แบบโครงสร้างของฐาน	50
รูปที่ 5.1	พิกัด link ของหุ่นยนต์ที่สร้างขึ้น	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 2.1 ชนิดของข้อต่อหุ่นยนต์	2
ตารางที่ 2.2 ขอบเขตการทำงานของหุ่นยนต์ ที่ขึ้นอยู่กับแกนหลัก	3
ตารางที่ 2.3 คุณลักษณะของหุ่นยนต์ (Robot Characteristics)	6
ตารางที่ 2.4 การเคลื่อนที่แบบ yaw , pitch และ roll	8
ตารางที่ 2.5 Kinematics parameters	21
ตารางที่ 2.6 การถ่ายโอนโครงพิกัด k ไปยังโครงพิกัด k-1	23
ตารางที่ 5.1 Kinematics parameters ของหุ่นยนต์ในรูปที่ 6.1	51



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบันนี้ หุ่นยนต์ นับได้ว่ามีบทบาทสำคัญอย่างมากในอุตสาหกรรมหลายๆประเภท โดยทั่วไปแล้วถ้ากล่าวถึงหุ่นยนต์ คนทั่วไปมักจะคิดถึงเครื่องยนต์ที่มีรูปร่างลักษณะเหมือนมนุษย์ สามารถคิดและทำงานได้ด้วยตนเอง แต่ที่จริงแล้ว คำว่า Robotics ตามคำนิยามของ Robotics Industrial Association และ Robotics Institute of America นั้นหมายถึง เครื่องยนต์ที่ประกอบกันขึ้นมาเพื่อทำงานบางอย่างแทนมนุษย์ สามารถควบคุมได้โดยใช้โปรแกรมคอมพิวเตอร์ และสามารถที่จะเปลี่ยนการทำงานให้ทำงานอย่างอื่นได้

ปริญญาโทนี้เป็นโครงการสร้างมือกล (Robotic hand) ที่มีวัตถุประสงค์ของโครงการดังนี้คือ

1. เพื่อศึกษาทฤษฎีวิชา Robotics และสร้างต้นแบบมือกล
2. เพื่อศึกษาและควบคุมการทำงานของมือกลด้วยคอมพิวเตอร์
3. นำความรู้ที่ได้จากการศึกษา,ทดลองไปประยุกต์ใช้งานและวางแนวทางการพัฒนาต่อไป

โดยมีการระบุข้อกำหนด,เป้าหมายของโครงการมือกลนี้คือ สามารถกำหนดรูปแบบในการเคลื่อนที่ของแขน มือและนิ้วมือได้ด้วยคอมพิวเตอร์ โดยมีการส่งสัญญาณควบคุมวงจรมอเตอร์ มีการรับสัญญาณป้อนกลับจากเซ็นเซอร์ไปประมวลผลด้วยคอมพิวเตอร์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 ทฤษฎีพื้นฐานของการวิเคราะห์และควบคุมหุ่นยนต์ (Fundamental of Robotics Analysis and Control)

หุ่นยนต์ (Robot) คือ อุปกรณ์ทางกลซึ่งควบคุมด้วยซอฟต์แวร์ โดยใช้เซ็นเซอร์ตรวจจับ End-effectors ของหุ่นยนต์ที่เคลื่อนที่ตามโปรแกรมในพื้นที่การทำงาน (Workspace) เพื่อทำการเคลื่อนย้ายวัตถุ



2.1.1 การแบ่งแยกประเภทหุ่นยนต์ (Robot Classification)

1. เทคโนโลยีการขับเคลื่อน (Drive Technologies)

พิจารณาจากต้นกำลังที่ใช้ขับเคลื่อนของหุ่นยนต์ ที่นิยมมี 2 แบบคือ การขับเคลื่อนด้วยไฟฟ้าและขับเคลื่อนด้วยไฮดรอลิก แขนกลส่วนใหญ่ในปัจจุบันเป็นแบบขับเคลื่อนด้วยไฟฟ้า โดยใช้ดีซีเซอร์โวมอเตอร์ (DC Servo Motor) ดีซีสเต็ปมิ่งมอเตอร์ (DC Stepping Motor) ในการเคลื่อนย้ายวัตถุที่ต้องการความเร็วสูง มักนิยมใช้แบบขับเคลื่อนด้วยไฮดรอลิก

2. รูปทรงของขอบเขตการทำงาน (Work Envelope Geometries)

ขอบเขตการทำงานสุทธิ (Gross Work Envelope) หมายถึงขอบเขตในปริภูมิสามมิติที่ข้อมือของแขนกลสามารถเคลื่อนที่ไปยังตำแหน่งนั้นได้ เราจะเรียกแกนของ 3 ข้อต่อแรกของแขนกลว่าเป็นแกนหลัก (Major axe) ซึ่งใช้ในการพิจารณาดำเนินการของข้อมือ ส่วนแกนของข้อต่อที่เหลือเรียกว่า แกนรอง (minor axe) ใช้กำหนดลักษณะการวางตัวหรือหมุน (Orientation) ของ tool ดังนั้นรูปทรงของขอบเขตการทำงานจึงพิจารณาได้จาก ลำดับชนิดของข้อต่อที่ใช้ใน 3 แกนแรก ข้อต่อมีหลายแบบแต่ที่เป็นพื้นฐานและนิยมใช้มี 2 แบบ

ชนิด	เครื่องหมาย	สัญลักษณ์	การเคลื่อนที่
ข้อต่อหมุน (Revolute)	R		แบบหมุนรอบแกน
ข้อต่อเลื่อน (Prismatic)	P		เชิงเส้นตามแนวแกน

ตารางที่ 2.1 ชนิดของข้อต่อหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรวมกันของข้อต่อหมุน (R) และข้อต่อเลื่อน (P) ใน 3 แกนหลัก ก่อให้เกิดรูปทรงการเคลื่อนที่หลายรูปแบบ ดังตัวอย่างในตารางที่ 2.2 ซึ่งแสดงเฉพาะรูปแบบที่นิยมใช้ในการวิเคราะห์การเคลื่อนที่ของแขนกล ข้อต่อหมุน (R) จะวิเคราะห์ยากกว่าแบบข้อต่อเลื่อน (P) ดังนั้นยังมีข้อต่อหมุน มาก แขนกลก็จะต้องซับซ้อนมากขึ้น

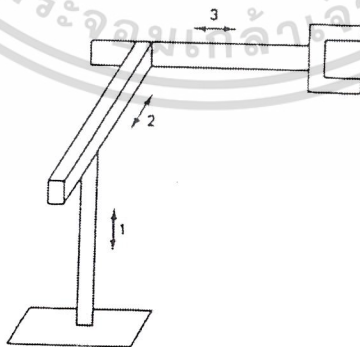
Robot	Axis 1	Axis 2	Axis 3	Total revolute
Cartesian	P	P	P	0
Cylindrical	R	P	P	1
Spherical	R	R	P	2
SCARA	R	R	P	2
Articulated	R	R	R	3

P = prismatic , R = revolute

ตารางที่ 2.2 ขอบเขตการทำงานของหุ่นยนต์ (Robot work envelopes) ที่ขึ้นอยู่กับแกนหลัก

ชนิดของหุ่นยนต์ แบ่งตามลักษณะขอบเขตการทำงานของหุ่นยนต์ (Robot work envelopes) ได้เป็น 5 ชนิดคือ

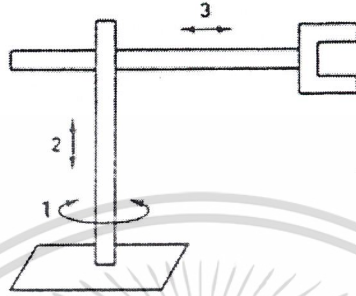
1. หุ่นยนต์พิกัดคาร์ทีเซียน (Cartesian-coordinate robot) หรือหุ่นยนต์พิกัดฉาก (Rectangular-coordinate robot) สัญลักษณ์ PPP ดังรูปที่ 2.1 โดยข้อมือจะเลื่อนขึ้น-ลง, เข้า-ออก และเดินหน้า-หลัง ทำให้พื้นที่การทำงานมีลักษณะเป็นกล่องสี่เหลี่ยมมุมฉาก



รูปที่ 2.1 หุ่นยนต์พิกัดคาร์ทีเซียน

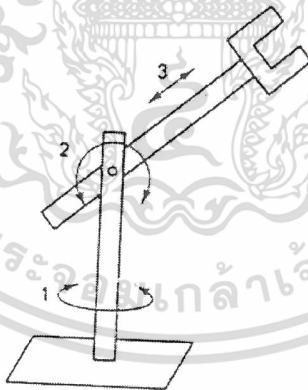
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หุ่นยนต์พิกัดทรงกระบอก (Cylindrical-coordinate robot) สัญลักษณ์ RPP ดังรูปที่ 2.2 สามารถเคลื่อนที่ขึ้นข้างบนได้ตามแกนตั้งที่เป็นแกนหลัก สามารถเคลื่อนที่เข้า-ออกตามแนวรัศมี และแขนหมุนรอบแกนตั้งฉากกับฐานได้ พื้นที่การทำงานจึงเป็นแบบทรงกระบอก



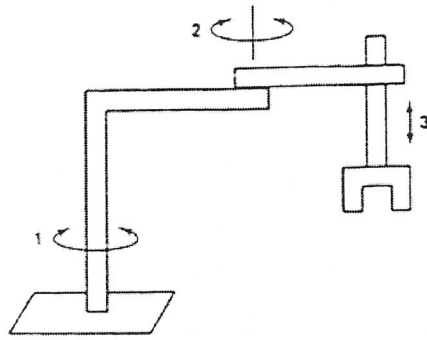
รูปที่ 2.2 หุ่นยนต์พิกัดทรงกระบอก

3. หุ่นยนต์พิกัดทรงกลม (Spherical-coordinate robot) สัญลักษณ์ RRP แสดงดังรูปที่ 2.3 ลักษณะการเคลื่อนที่ของแขนจะสามารถยกขึ้นลงได้ในแนวตั้งโดยยกทำมุมกับฐาน แขนสามารถหมุนได้รอบแกนแนวตั้งของฐาน พื้นที่การทำงานเป็นแบบทรงกลม



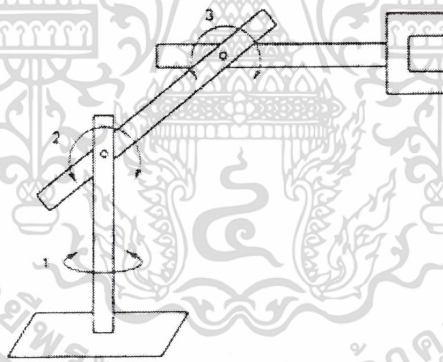
รูปที่ 2.3 หุ่นยนต์พิกัดทรงกลม

4. หุ่นยนต์ SCARA (Selective Compliance Assembly Robot Arm) ดังรูปที่ 2.4 มีลักษณะคล้ายหุ่นยนต์พิกัดทรงกลม มีสัญลักษณ์ RRP แต่แกนทั้ง 3 จะอยู่ในแนวตั้ง โดยข้อที่ 2 ทำให้แขนหมุนรอบแกนตั้งในแนวนอน เหมือนการหมุนของข้อแรก ภาพตัดขวางในแนวนอนของแขนที่เป็นเอกสารที่สวนได้ข้อที่ 2 นี้เป็นการใช้งานที่ควรศึกษาเป็นไปอย่างถูกต้องให้เข้าใจประโยชน์ด้านการค้า พื้นที่การทำงานค่อนข้างซับซ้อนขึ้นอยู่กับข้อจำกัดในการเคลื่อนที่ของสองแกนแรก ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 หุ่นยนต์ SCARA

5. หุ่นยนต์ข้อต่อหมุน (Articulated-coordinate robot) หรือเรียก Revolute robot สัญลักษณ์ RRR ทั้ง 3 ข้อเป็นแบบข้อต่อหมุน หุ่นยนต์แบบนี้มีลักษณะใกล้เคียงกับแขนของมนุษย์มากที่สุด มีข้อหมุนต่างๆ เหมือนกัน ดังนั้นพื้นที่การทำงานจึงสามารถที่จะทำงานได้ในทุกตำแหน่งในระยะความยาวของแขน



รูปที่ 2.5 หุ่นยนต์ข้อต่อหมุน

ข้อดี-ข้อเสียของหุ่นยนต์แต่ละชนิดนี้แตกต่างกันออกไป เพราะลักษณะทางกายภาพแตกต่างกัน แต่ถ้ามองในแง่ของการทำงานที่เป็นแบบซ้ำๆ ที่เดิมตลอด หุ่นยนต์พิกัดคาร์ทีเซียนจะสามารถทำงานได้ดีกว่า คือสามารถเคลื่อนที่ไปหาเป้าหมายโดยมีความผิดพลาดน้อยที่สุด แต่ถ้ามองในแง่การเข้าถึงวัตถุ ชนิดพิกัดทรงกลมและข้อต่อหมุนจะสามารถเข้าถึงวัตถุได้ดีกว่าชนิดอื่น ชนิดพิกัดทรงกระบอกมีข้อดีที่สามารถยกวัตถุได้มากกว่า ในงานทั่วไปแล้วใช้แบบชนิดพิกัดทรงกลมและพิกัดทรงกระบอก เพราะทั้ง 2 ชนิดสามารถที่จะทำงานเป็นแบบ Load และ Unload โดยมีการเคลื่อนที่ของแขนออกไปในด้านข้างได้ดีกว่าชนิดอื่นๆ

2.1.2 ลักษณะเฉพาะของหุ่นยนต์ (Robot Specification)

นอกจากเกณฑ์ต่างๆ ที่ใช้ในการแบ่งประเภทของหุ่นยนต์ดังกล่าวมาแล้ว ยังมีลักษณะบางอย่างที่ช่วยผู้ใช้ในการเลือกหุ่นยนต์ที่ต้องการ ลักษณะต่างๆ เหล่านี้แสดงดังตารางที่ 2.3

คุณลักษณะ (Characteristics)	หน่วย (Unit)
จำนวนแกน	-
ความสามารถในการยกน้ำหนัก	Kg
ความเร็วสูงสุด , เวลาครบรอบ	mm/sec
ระยะเอื้อมถึงและสโตรค	mm
การหมุนของเครื่องมือ	Deg
การซ้ำตำแหน่งเดิม	Mm
ความละเอียดและความแม่นยำ	Mm
สภาวะแวดล้อมของการทำงาน	-

ตารางที่ 2.3 คุณลักษณะของหุ่นยนต์ (Robot Characteristics)

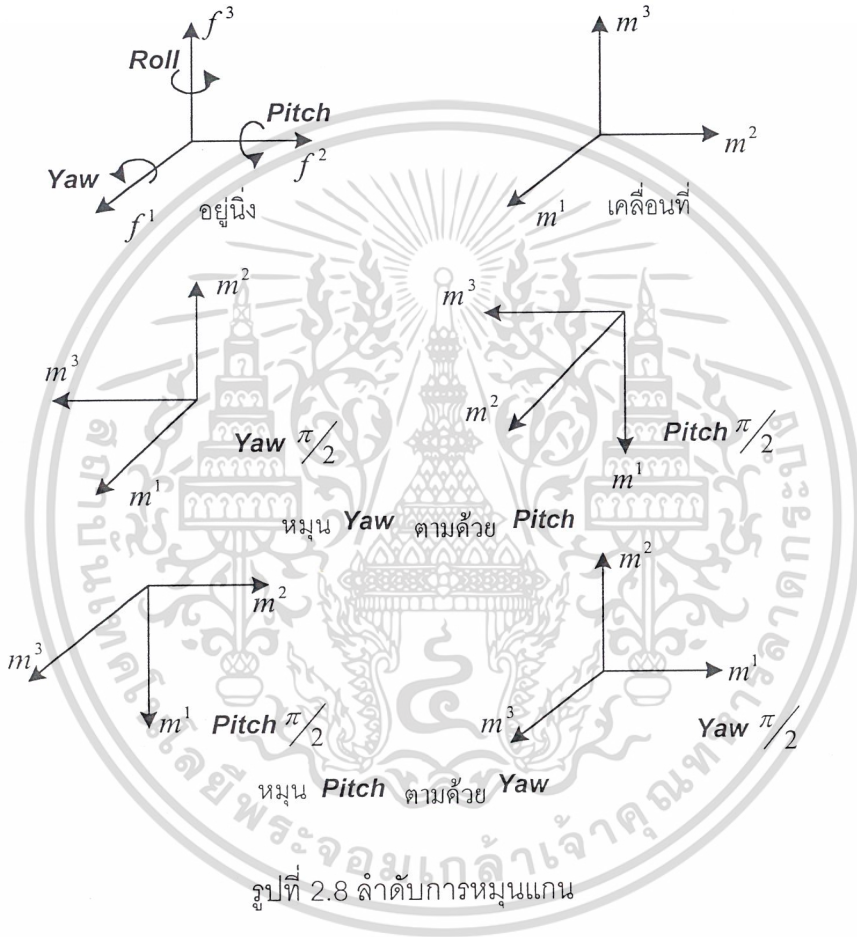
ระยะที่เอื้อมถึงและสโตรค (Reach and Stroke) เป็นการวัดขนาดของพื้นที่ทำงาน (Work Envelope) อย่างคร่าวๆ ระยะที่เอื้อมถึงในแนวนอน (Horizontal Reach) คือระยะทางตามแนวรัศมีที่มากที่สุดที่ข้อมือสามารถไปถึงโดยวัดจากแกนแนวตั้งที่ทำให้หุ่นยนต์หมุน สโตรคแนวนอน (Horizontal Stroke) คือระยะทางตามแนวรัศมีทั้งหมดที่ข้อมือสามารถไปได้ ดังนั้นระยะที่เอื้อมถึงในแนวนอน ลบด้วยสโตรคแนวนอน คือ ระยะทางตามแนวรัศมีที่น้อยที่สุดที่ข้อมือสามารถไปถึงได้โดยวัดจากแกนฐานซึ่งมีค่าเป็นบวกเสมอ จึงได้ว่า $Stroke \leq Reach$

ตัวอย่างเช่น ระยะที่เอื้อมถึงในแนวนอน (Horizontal Reach) ของหุ่นยนต์พิกัดทรงกระบอกก็คือรัศมีของทรงกระบอกอันนอกของพื้นที่ทำงาน ในขณะที่ สโตรคแนวนอน (Horizontal Stroke) คือผลต่างระหว่างรัศมีของทรงกระบอกอันในและอันนอก ดังแสดงในรูปที่ 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมุน tool รอบแกนข้อมือ f^2 , roll คือการหมุน tool รอบแกนข้อมือ f^3 ในแต่ละกรณีนี้มุมบวกเกิดจากการหมุนทวนเข็มนาฬิกาโดยมองจากปลายแกนเข้าหาจุดกำเนิด

ลำดับของการหมุน yaw, pitch, roll มีความสำคัญมาก เพราะมีผลต่อลักษณะการวางตัวของสุดท้ายของ tool ตัวอย่างเช่น หมุน yaw ไป 90° ตามด้วย pitch 90° จะมีลักษณะการวางตัวของสุดท้ายต่างกับการหมุน pitch ไป 90° แล้วตามด้วย yaw 90° เป็นต้น



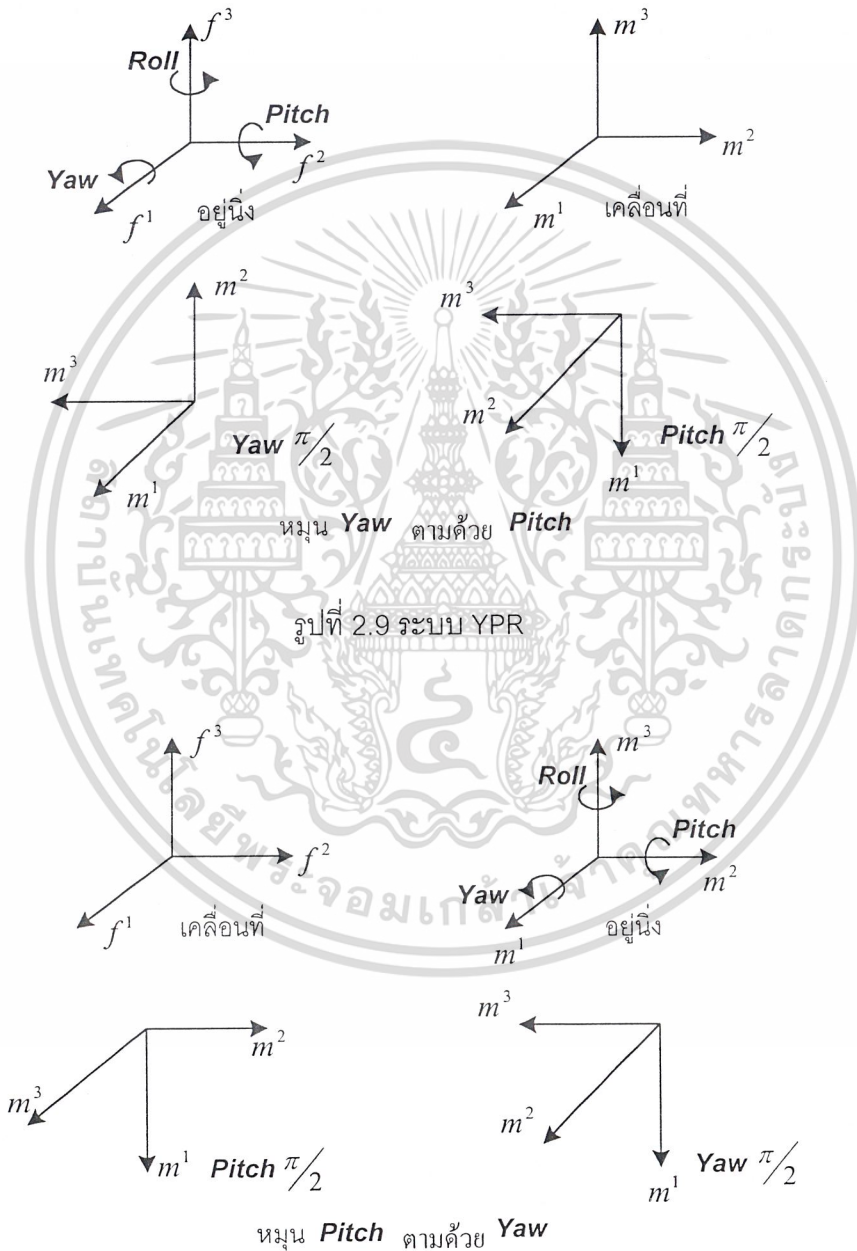
รูปที่ 2.8 ลำดับการหมุนแกน

การกำหนด YPR ตามลำดับการหมุน แสดงดังตารางที่ 2.4

ขั้นตอน	การอธิบาย	แกน
1	Yaw	f^1
2	Pitch	f^2
3	Roll	f^3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ตารางที่ 2.4 การเคลื่อนที่แบบ yaw, pitch และ roll
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกวิธีหนึ่งที่ใช้ระบบการหมุนคือ ระบบลำดับการหมุนย้อนลำดับกับ YPR และคิดเทียบกับ โครงพิกัดเคลื่อนที่ของ tool M แทนที่โครงพิกัดอยู่นิ่งของข้อมือ F ดังรูปที่ 2.10 โดยเริ่มแรก roll คือ การหมุนรอบแกน m^3 , pitch คือการหมุนรอบแกน m^2 , yaw คือการหมุนรอบแกน m^1 การคิด ลักษณะนี้เรียกว่าเป็นระบบ RPY จะให้ผลลัพธ์เหมือนกับแบบ YPR



รูปที่ 2.10 ระบบ RPY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความแม่นยำของการเคลื่อนที่ ความสำคัญประการหนึ่งของการสร้างหุ่นยนต์คือ ความแม่นยำในการทำงาน ซึ่งความแม่นยำขึ้นอยู่กับตัวแปร 3 อย่างคือ

1. Spatial Resolution คือช่วงการเคลื่อนที่ที่มีระยะทางสั้นที่สุดที่หุ่นยนต์แต่ละตัวสามารถที่จะทำได้ ซึ่ง Spatial Resolution นี้ขึ้นอยู่กับองค์ประกอบสำคัญ 2 ประการคือ

1.) ระบบการควบคุม (Control System) ระบบการควบคุมนี้จะรวมถึงการวัดสัญญาณป้อนกลับของหุ่นยนต์ด้วย ช่วงการเคลื่อนที่ที่ระบบการควบคุมสามารถที่จะทำได้ขึ้นอยู่กับหน่วยความจำหลักของคอมพิวเตอร์ ที่มีหน่วยความจำ 8 บิตจะสามารถแบ่งช่วงการเคลื่อนที่ได้เป็น 256 ช่วงคือ ช่วงของการเคลื่อนที่ที่คอมพิวเตอร์แบ่งได้มีค่าเท่ากับ 2^n เมื่อ n คือหน่วยความจำหลักของคอมพิวเตอร์

2.) ความคลาดเคลื่อนเชิงกล (Mechanical inaccuracy) ของหุ่นยนต์แต่ละตัวขึ้นอยู่กับลักษณะของข้อหมุน (joint) และข้อต่อ (link) และระบบต้นกำลังของหุ่นยนต์ตัวนั้นด้วย

2. Accuracy คือตัวที่แสดงถึงความสามารถของหุ่นยนต์ในการที่เคลื่อนที่เข้าใกล้จุดเป้าหมายตามที่เราระบุ Accuracy สามารถที่จะกำหนดให้อยู่ในเทอมของ Spatial Resolution ได้ทั้งนี้เพราะว่าการเคลื่อนที่ให้เข้าใกล้จุดเป้าหมาย ก็ต้องขึ้นกับช่วงของการเคลื่อนที่ที่มีความละเอียดมากน้อยเพียงใด ในการทำงานเราต้องวางจุดที่เราต้องการให้หุ่นยนต์ทำงานอยู่ระหว่างกลางของตำแหน่งของการเคลื่อนที่ของหุ่นยนต์ ทั้งนี้เพราะว่าความคลาดเคลื่อนเชิงกล มีผลต่อความแม่นยำของหุ่นยนต์ ความแม่นยำของหุ่นยนต์กำหนดให้เท่ากับครึ่งหนึ่งของระยะทางการเคลื่อนที่ที่สั้นที่สุดของหุ่นยนต์ที่สามารถทำได้ โดยขึ้นอยู่กับองค์ประกอบต่อไปนี้

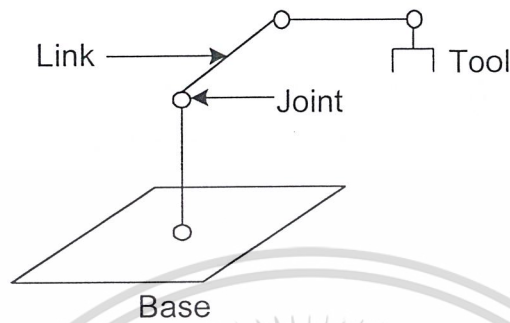
- 1.) พื้นที่การทำงานของหุ่นยนต์ ถ้าแขนทำงานในพื้นที่การทำงานจะมีความแม่นยำมากกว่าเมื่อแขนออกนอกพื้นที่การทำงาน
- 2.) วงรอบการทำงาน ถ้าวงรอบการทำงานเป็นวงรอบที่แน่นอนความแม่นยำจะมีมากขึ้น
- 3.) น้ำหนักที่ได้รับ ถ้าหุ่นยนต์ทำงานโดยการรับน้ำหนักมาก ๆ ความแม่นยำจะลดลง

3. Repeatability คือความสามารถของหุ่นยนต์ในการกลับมาทำงานซ้ำที่เดิม หลายๆ ครั้งได้ เนื่องจากสาเหตุต่างๆ เช่น Backlash ในเฟืองและความยืดหยุ่น (Flexibility) ของส่วนต่างๆ ซึ่งมักทำให้เกิดความผิดพลาดใน Repeatability

2.1.3 Direct Kinematics : The Arm Equation

แขนกลมีลักษณะเป็นวัตถุเกร็ง (Rigid body) เป็น link หลายๆ อันนำมาเชื่อมต่อเข้าด้วยกันโดยใช้ข้อต่อ ดังรูปที่ 2-11 ปลายข้างหนึ่งของ link ริมสุดจะติดแน่นกับฐานส่วนปลายของ link อีกอันที่อยู่คนละด้านจะเคลื่อนที่ได้อย่างอิสระ ปลายข้างที่เคลื่อนที่นี้จะต่อเข้ากับเครื่องมือ (Tool

or End-effector) ตามปกติมีข้อต่อ 2 ชนิดที่ใช้เชื่อมต่อ links คือ ข้อต่อหมุน (Revolute joint) และข้อต่อเลื่อน (Prismatic joint)



รูปที่ 2.11 แขนกลที่มีลักษณะเป็น links

จุดประสงค์คือ การควบคุมทั้งตำแหน่งและการหมุนของเครื่องมือในระนาบสามมิติ เพื่อโปรแกรมให้เครื่องมือเคลื่อนที่ไปตามเส้นทางที่กำหนดไว้และใช้ในการเคลื่อนย้ายวัตถุในพื้นที่ทำงาน โดยการโปรแกรมการเคลื่อนที่ของเครื่องมือนี้ จำเป็นจะต้องคำนวณความสัมพันธ์ระหว่างตัวแปรข้อต่อกับตำแหน่งและการหมุนของอุปกรณ์ ซึ่งเรียกว่า ปัญหา Direct Kinematics โดยนิยามดังนี้

ปัญหา Direct Kinematics คือการกำหนดเวกเตอร์ของตัวแปรข้อต่อ (Joint Variables) ของแขนกลโดยแยกตามลักษณะข้อต่อเป็น

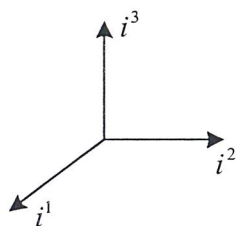
- ข้อต่อหมุน ตัวแปรได้แก่ ค่าองศาการหมุนรอบแกนข้อต่อ
- ข้อต่อเลื่อน ตัวแปรได้แก่ ระยะการเคลื่อนที่ไปตามแนวแกนข้อต่อ

แล้วทำการคำนวณหาตำแหน่ง (Position) และการหมุน (Orientation) ของเครื่องมือเทียบกับโครงฟิวกัดที่ติดอยู่กับฐานของแขนกล

Dot and Cross Products

เวกเตอร์ในปริภูมิ n มิติ R^n สามารถแทนด้วยลูกศรพุ่งออกจากจุดกำเนิดดังรูปที่ 2.12 ซึ่งเป็นกรณี $n = 3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.12 โครงงพิกัดฉากใน R^3

นิยาม 1 Dot Product ของ 2 เวกเตอร์ X และ Y ใน R^n เขียนเป็น $X \cdot Y$

$$X \cdot Y \cong \sum_{k=1}^n X_k Y_k$$

(สัญลักษณ์ \cong อ่านว่า นิยามเท่ากับ) ถ้าใช้การ Transpose ของเมตริกซ์ จะแสดงได้ในรูป.

$X \cdot Y = X^T Y$ โดย Dot product ใน R^n มีคุณสมบัติพื้นฐานดังนี้

ให้ $\{x, y, z\}$ เป็นเวกเตอร์ใน R^n และ $\{\alpha, \beta\}$ เป็นสเกลาร์ ดังนั้น

1. $x \cdot x \geq 0$
2. $x \cdot x = 0 \Leftrightarrow x = 0$
3. $x \cdot y = y \cdot x$
4. $(\alpha x + \beta y) \cdot z = \alpha(x \cdot z) + \beta(y \cdot z)$

นิยาม 2 การตั้งฉาก (Orthogonality) เวกเตอร์ x และ y ใน R^n จะตั้งฉากกันก็ต่อเมื่อ $x \cdot y = 0$.

นิยาม 3 Completeness เป็นเซตของเวกเตอร์ที่ตั้งฉากกัน $\{x^1, x^2, x^3, \dots, x^n\}$ ใน R^n เป็น Complete ก็ต่อเมื่อ

$$y \cdot x^k = 0 \text{ เมื่อ } 1 \leq k \leq n \Rightarrow y = 0$$

จำนวนเวกเตอร์ใน Complete Orthogonal Set เรียกว่ามิติ ดังนั้น R^3 ก็คือปริภูมิสามมิติที่มีรูปทั่วไปคือ R^n

นิยาม 4 ขนาด (Norm) ของเวกเตอร์ x ใน R^n เขียนเป็น $\|x\|$ นิยามได้ว่า

$$\|x\| \cong (x \cdot x)^{1/2} = \left(\sum_{k=1}^n x_k^2 \right)^{1/2}$$

โดยยูนิตเวกเตอร์ที่ตั้งฉากกัน เรียกว่า Orthogonal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหมุน (Orientation) ให้ x และ y เป็นเวกเตอร์ใน R^3 และ θ เป็นมุมจาก x ไป y ดัง
นั้น

$$x \cdot y = \|x\| \|y\| \cos \theta$$

ดังนั้น Dot Product จึงเป็นการวัดการหมุนระหว่างสองเวกเตอร์นั้น นอกจากนี้ยังมีการกระทำของเวกเตอร์อีกชนิดที่เป็นประโยชน์ในการวิเคราะห์แขนกลคือ Cross Product โดยนิยามได้ดังนี้

นิยาม 5 Cross Product ของเวกเตอร์ u และ v ใน R^3 คือเวกเตอร์ $w = u \times v$ ซึ่งตั้งฉากกับกฏมือขวาและ

$$w \cong \det \begin{bmatrix} i^1 & i^2 & i^3 \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{bmatrix} = \begin{bmatrix} (u_2 v_3 - u_3 v_2) i^1 \\ (u_3 v_1 - u_1 v_3) i^2 \\ (u_1 v_2 - u_2 v_1) i^3 \end{bmatrix}$$

โดยเวกเตอร์หนึ่งหน่วย $\{i^1, i^2, i^3\}$ เป็นเวกเตอร์ตามแนวแกนของโครงพิกัดฉากตามกฏมือขวา ดังในรูปที่ 2.12 โดย $i^1 = i^2 \times i^3$, $i^2 = i^3 \times i^1$ และ $i^3 = i^1 \times i^2$ ถ้า θ เป็นมุมจาก u ไป v ดังนั้น Cross Product

$$\|x \times y\| = \|u\| \|v\| \sin \theta$$

โครงพิกัด (Coordinate Frames)

นิยาม 1 พิกัด (Coordinates) ให้ p เป็นเวกเตอร์ใน R^n และ $X = \{x^1, x^2, x^3, \dots, x^n\}$ เป็น Complete Orthogonal Set ของ R^n

ดังนั้นพิกัดของ p เทียบกับ X เขียนเป็น $[p]^X$ และนิยามดังนี้

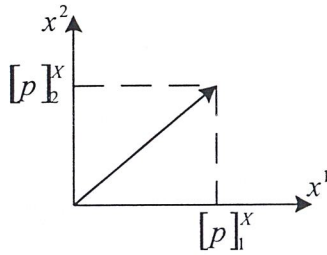
$$P = \sum_{k=1}^n [p]_k^X x^k$$

Complete Orthogonal Set X บางครั้งเรียกว่า โครงพิกัดหนึ่งหน่วยตั้งฉาก (Orthogonal coordinate frame) ซึ่งการใช้เวกเตอร์โครงพิกัดเป็นยูนิตเวกเตอร์ที่ตั้งฉากกันนี้ ทำให้หาพิกัดของเวกเตอร์อื่นเทียบกับโครงพิกัดนี้ได้ง่าย โดยใช้ทฤษฎีต่อไปนี้

โครงพิกัดฉาก (Orthogonal Coordinates) ให้ p เป็นเวกเตอร์ใน R^n และ $[p]^X$ เป็นพิกัดของ p เทียบกับโครงพิกัดฉาก $X = \{x^1, x^2, x^3, \dots, x^n\}$ ดังนั้น พิกัดที่ k ของ p เทียบกับ x คือ

$$[p]_k^X = p \cdot x^k : 1 \leq k \leq n$$

จะเห็นได้ว่า พิกัดที่ k ของ p เทียบกับโครงพิกัดฉาก X ก็คือ Dot Product ของ p กับสมาชิกตัวที่ k ของเซต X โดยสามารถแสดงความสัมพันธ์ระหว่างเวกเตอร์กับพิกัดได้ดังรูปที่ 2.13 ซึ่งเป็นปริภูมิ 2 มิติ โดย p คือ ผลบวกของภาพฉายตั้งฉากลงบนเวกเตอร์ x^1, x^2 ตามลำดับ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 พิกัดของ p เทียบกับโครงพิกัด X

คำตอบของปัญหา Direct Kinematics ในแขนกล เราต้องแสดงตำแหน่งและการหมุนของเครื่องมือที่เคลื่อนที่เทียบกับโครงพิกัดที่ติดกับฐานที่อยู่นิ่ง ดังนั้นจึงต้องมีการถ่ายโอนพิกัดจากเครื่องมือไปข้อมือ ข้อมือไปข้อศอก ข้อศอกไปไหล่ และต่อไปเรื่อยๆ การถ่ายโอนพิกัดแต่ละอันสามารถแสดงได้ด้วยเมตริกซ์

การถ่ายโอนพิกัด (Coordinate Transformations) ให้ $F = \{f^1, f^2, f^3, \dots, f^n\}$ และ $M = \{m^1, m^2, m^3, \dots, m^n\}$ เป็นโครงพิกัดของ R^n และ F เป็นโครงพิกัด Orthogonal ให้ A เป็นเมตริกซ์ขนาด $n \times n$ โดย $A_{kj} \cong f^k \cdot m^j$ เมื่อ $1 \leq k, j \leq n$ ดังนั้น สำหรับแต่ละจุด p ใน R^n

$$[p]^F = A[p]^M$$

ปัญหาของการถ่ายโอนพิกัด คือการให้พิกัดของ p เทียบกับโครงพิกัด M มาแล้วให้หาพิกัดของ p เทียบกับโครงพิกัด F เมตริกซ์ A เรียกว่า เมตริกซ์ถ่ายโอนพิกัด (Coordinate Transformation Matrix) เป็นคอลัมน์ที่ j ของ A จาก $a^j = [m^j]^F$ สำหรับ $1 \leq j \leq n$ หรือกล่าวได้ว่า คอลัมน์ที่ j ของเมตริกซ์ A คือพิกัดของเวกเตอร์ลำดับที่ j ของโครงพิกัดต้นกำเนิด M เทียบกับโครงพิกัดปลายทาง F

อินเวอร์สของการถ่ายโอนโครงพิกัด (Inverse Coordinate Transformation) ให้ F และ M เป็นโครงพิกัดฉาก Orthogonal ใน R^n มีจุดกำเนิดเดียวกัน และเป็นเมตริกซ์ถ่ายโอนพิกัดจาก M ไป F ดังนั้น เมตริกซ์ถ่ายโอนพิกัดจาก F ไป M คือ A^{-1} โดย

$$A^{-1} = A^T$$

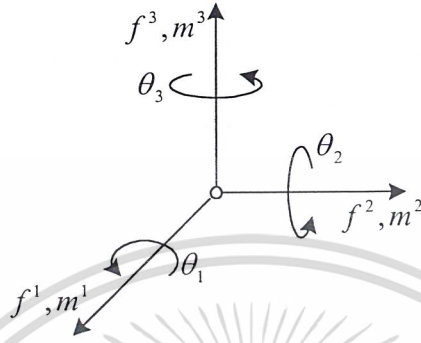
การหมุน (Rotation)

ในการระบุตำแหน่งและการหมุนของเครื่องมือในเทอมของพิกัดที่ติดอยู่กับฐานที่อยู่นิ่ง การถ่ายโอนพิกัดจึงเกี่ยวข้องกับทั้งการหมุนและการเคลื่อนที่เปลี่ยนตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

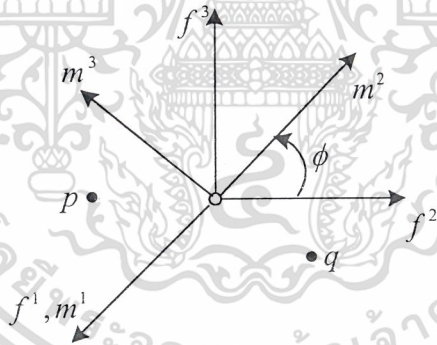
การหมุนพื้นฐาน (Fundamental Rotations)

ถ้าโครงพิกัดเคลื่อนที่ M รอบยูนิตเวกเตอร์อันหนึ่งของโครงพิกัดอยู่นิ่ง F ดังนั้นเมตริกซ์การถ่ายโอนพิกัดที่ได้เรียกว่า เมตริกซ์การหมุนพื้นฐาน ในปริภูมิ R^3 มีทางเป็นไปได้ 3 ทางดังแสดงในรูป 2.14



รูปที่ 2.14 การหมุนพื้นฐานใน R^3

ถ้าเราหมุนโครงพิกัดเคลื่อนที่ M รอบแกน f^1 ของโครงพิกัดอยู่นิ่ง F ให้ ϕ เป็น องศาการหมุนวัดตามกฎมือขวา ดังรูป 2.15



รูปที่ 2.15 การหมุนของ M รอบ f^1 ด้วยมุม ϕ

ให้ $R_1(\phi)$ เป็นเมตริกซ์การถ่ายโอนพิกัดเคลื่อนที่ M ไป F (จากการถ่ายโอนพิกัด) จะได้

$$R_1(\phi) = \begin{bmatrix} f^1 \cdot m^1 & f^1 \cdot m^2 & f^1 \cdot m^3 \\ f^2 \cdot m^1 & f^2 \cdot m^2 & f^2 \cdot m^3 \\ f^3 \cdot m^1 & f^3 \cdot m^2 & f^3 \cdot m^3 \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
เนื่องจากเราหมุนรอบแกน f^1 จากรูปที่ 2.15 จะได้ $f^1 = m^1$ แทนใน $R(\phi)$ และ

จากรูปที่ 2.15 เวกเตอร์ $\{f^1, f^2, f^3\}$ และ $\{m^1, m^2, m^3\}$ เป็นเซตของยูนิตเวกเตอร์ที่ตั้งฉากกัน ดังนั้นจะได้

$$R_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & f^2 \cdot m^2 & f^2 \cdot m^3 \\ 0 & f^3 \cdot m^2 & f^3 \cdot m^3 \end{bmatrix}$$

จากรูปที่ 2.15 เวกเตอร์ $\{f^2, f^3, m^2, m^3\}$ อยู่ในระนาบตั้งฉากกับ $f^1 = m^1$ จาก การหมุน (Orientation) จะได้ว่า Dot Product ของยูนิตเวกเตอร์เท่ากับค่า cosine ของมุม ระหว่าง 2 เวกเตอร์นั้น จากรูปที่ 2.15 มุมจาก f^2 ไป m^2 และ f^3 ไป m^3 เท่ากับ ϕ , มุมจาก f^2 ไป m^3 เท่ากับ $\pi/2 + \phi$ และมุมจาก f^3 ไป m^2 เท่ากับ $-\pi/2 + \phi$ ดังนั้นจะได้

$$R_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}$$

ในการทำงานเดียวกัน ถ้า $R_2(\phi)$ และ $R_3(\phi)$ แสดงการหมุนของโครงพิกัด M รอบ f^2 และ f^3 ของ F ตามลำดับจะได้

$$R_2(\phi) = \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix}$$

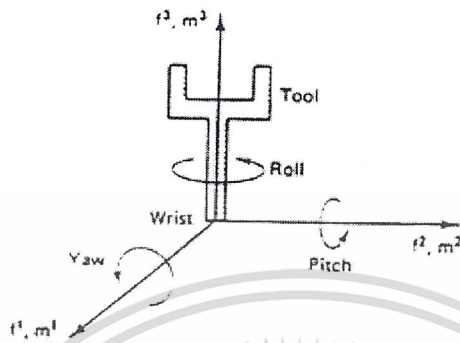
$$R_3(\phi) = \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

แถวที่ k และคอลัมน์ที่ k ของ $R_k(\phi)$ จะเหมือนกับแถวที่ k คอลัมน์ที่ k ของเมตริกซ์เอกลักษณ์ I ส่วน Submatrix 2 x 2 จะได้ว่าเทอมในแนวทแยงมุม จะเป็น $\cos \phi$ ส่วนเทอมนอกแนวทแยงมุมเป็น $\pm \sin \phi$ เมื่อ ϕ เป็นมุมการหมุนโดยเครื่องหมายของเทอมที่อยู่เหนือแนวทแยงมุมเป็น $(-1)^k$ สำหรับ $R_k(\phi)$ เช่น $R_1(\phi)$ จะได้ $-\sin \phi$ เป็นต้น

การหมุนผสม (Composite Rotations)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า เมื่อคุณเมตริกซ์การหมุนพื้นฐานเข้าด้วยกัน เมตริกซ์ผลลัพธ์ที่ได้จะแสดงถึงลำดับการหมุนไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งยังมีเหตุที่แปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้รอบยูนิตเวกเตอร์ เราเรียกเมตริกซ์ที่คูณกันนี้ว่า การหมุนผสม โดยการใช้การหมุนผสมนี้เราจะ

สามารถระบุการหมุนใดๆ ของเครื่องมือได้ พิจารณาภาพเครื่องมือแสดงในรูปที่ 2.16 โค้งพิกัดเคลื่อนที่ $M = \{m^1, m^2, m^3\}$ หมุนไปกับเครื่องมือ ในขณะที่โค้งพิกัดอยู่นิ่ง $F = \{f^1, f^2, f^3\}$ ติดอยู่ที่ส่วนปลายแขน การหมุนพื้นฐาน 3 อันคือ



รูปที่ 2.16 yaw-pitch-roll ของ tool

yaw เป็นการหมุนรอบแกน f^1

pitch เป็นการหมุนรอบแกน f^2

roll เป็นการหมุนรอบแกน f^3

การหมุนพื้นฐานแต่ละอันแทนด้วยเมตริกซ์ และเมตริกซ์ไม่มีคุณสมบัติการสลับที่การคูณ ดังนั้น ลำดับในการหมุนพื้นฐานจึงมีผลต่อการหมุนผสมที่ได้ นอกจากนั้นเมื่อมีการหมุนเกิดขึ้นแล้วครั้งหนึ่ง แกนของโค้งพิกัดทั้งสองก็จะไม่ทับกันอีกต่อไป

การหมุนครั้งต่อไปของเครื่องมือ จึงสามารถเป็นได้ทั้งการหมุนรอบยูนิตเวกเตอร์ของโค้งพิกัดอยู่นิ่ง F หรือโค้งพิกัดหมุน M ก็ได้ เพื่อป้องกันความสับสนในการหาเมตริกซ์การหมุนผสมจึงควรใช้อัลกอริธึมต่อไปนี้

อัลกอริธึมการหมุนผสม

1. ให้เมตริกซ์การหมุน $R = I$ ได้จากเมื่อเริ่มต้นโค้งพิกัดจาก F และ M ทับกันสนิท
2. ถ้าโค้งพิกัดเคลื่อนที่ M ถูกหมุนไป ϕ รอบยูนิตเวกเตอร์ที่ k ของโค้งพิกัดอยู่นิ่ง F ดังนั้นคุณเข้าทางซ้ายของ R ด้วย $R_k(\phi)$
3. ถ้าโค้งพิกัดเคลื่อนที่ M ถูกหมุนไป ϕ รอบยูนิตเวกเตอร์ที่ k ของมันเอง ดังนั้นคุณเข้าทางขวาของ R ด้วย $R_k(\phi)$
4. ถ้ามีการหมุนพื้นฐานหลายครั้งให้กลับไปขั้นตอน 2 มิฉะนั้นก็หยุด ผลลัพธ์ที่ได้เป็น

เมตริกซ์การหมุนผสม R ที่ใช้ถ่ายโอนพิกัด M ไป F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า เมื่อมีการทำแบบ yaw, pitch, roll เราสามารถหาเมตริกซ์ถ่ายโอนผสม yaw-pitch-roll ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ผลและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้โดยให้มุม yaw-pitch-roll แทนด้วยเวกเตอร์ θ ใน R^3 ดังต่อไปนี้

การถ่ายโอน Yaw-Pitch-Roll (Yaw-Pitch-Roll Transformation) ให้ $YPR(\theta)$ แทนเมตริกซ์การหมุนผลมที่ได้จากการหมุนโครงพิกัดเคลื่อนที่ $M = \{m^1, m^2, m^3\}$ ครั้งแรกรอบแกน f^1 ด้วยมุม yaw θ_1 ต่อมาหมุนรอบแกน f^2 ด้วยมุม pitch θ_2 และท้ายสุดหมุนรอบแกน f^3 ด้วยมุม roll θ_3 เมตริกซ์ผลม $YPR(\theta)$ ซึ่งถ่ายพิกัด M ไป F หาได้จาก

$$YPR(\theta) = \begin{bmatrix} C_2 C_3 & S_1 S_2 C_3 - C_1 S_3 & C_1 S_2 C_3 + S_1 S_3 \\ C_2 S_3 & S_1 S_2 S_3 + C_1 C_3 & C_1 S_2 S_3 - S_1 C_3 \\ -S_2 & S_1 C_2 & C_1 C_2 \end{bmatrix}$$

โดย $S_k \equiv \sin \theta_k$ และ $C_k \equiv \cos \theta_k$ และพิสูจน์โดยใช้หลักการอินทิเกรตและเมตริกซ์การหมุนพื้นฐานจะได้ $YPR(\theta) = R_3(\theta_3) R_2(\theta_2) R_1(\theta_1)$

การหมุนโครงเคลื่อนที่ M เทียบกับโครงอยู่นิ่ง F ปกติจะแบ่งได้เป็น 3 ชุดลำดับการหมุนพื้นฐานดังกล่าวถ่ายโอน YPR แต่ก็สามารถแสดงได้ด้วยการหมุนครั้งเดียวด้วยมุม ϕ รอบยูนิตเวกเตอร์ใดๆ u ซึ่งเรียกว่า แกนมุมเสมือน (Equivalent angle-axis)

พิกัดโฮโมจีเนียส (Homogeneous Coordinates)

การหมุน (Rotation) ใช้อธิบายลักษณะการวางตัว (Orientation) ของเครื่องมือ แต่ในการอธิบายตำแหน่งของเครื่องให้สัมพันธ์กับโครงพิกัดที่อยู่กับฐานของแขนกลนั้น จำเป็นต้องใช้การถ่ายโอนอีกแบบหนึ่งที่เรียกว่า การเลื่อนตำแหน่ง (Translation) การเลื่อนตำแหน่งนั้นมีข้อแตกต่างจากการหมุนที่ลำดับปฏิบัติการหนึ่งคือ ในการหมุนจุดกำเนิดของโครงพิกัดที่ถูกหมุนเป็นตำแหน่งเดียวกับจุดกำเนิดของโครงพิกัดดั้งเดิม เราจึงสามารถแสดงการหมุนในปริภูมิสามมิติได้ด้วยเมตริกซ์ 3×3 แต่จุดกำเนิดของโครงพิกัดที่เลื่อนตำแหน่งไปนั้นไม่เป็นจุดเดียวกับจุดกำเนิดของโครงพิกัดดั้งเดิม ดังนั้นจึงไม่สามารถแสดงการเลื่อนตำแหน่งใน R^3 ด้วยเมตริกซ์ 3×3 ได้ ในกรณีนี้จะใช้ปริภูมิ 4 มิติ เพื่อช่วยในการนิยามพิกัดโฮโมจีเนียส

พิกัดโฮโมจีเนียส (Homogeneous Coordinates) ให้ q เป็นจุดใน R^3 และ F เป็นโครงพิกัดจากสำหรับ R^3 ถ้า σ เป็นสเกลาร์เวกเตอร์ที่ไม่เป็นศูนย์ ดังนั้นพิกัดโฮโมจีเนียสของ q เทียบกับ F เขียนเป็น $[q]^F$ นิยามได้ว่า

$$[q]^F \equiv \sigma [q_1, q_2, q_3, 1]^T$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น พิกัดโฮโมจีเนียสของจุด q ใน R^3 แสดงได้ด้วยเวกเตอร์ $[q]^F$ ในปริภูมิ 4 มิติ R^4 ในการแปลงพิกัดโฮโมจีเนียส 4 มิติกลับเป็นเวกเตอร์ 3 มิติดั้งเดิม เราใช้ $q = H_\sigma [q]^F$ โดย H_σ เป็นเมตริกแปลงพิกัดโฮโมจีเนียสขนาด 3×4 (Homogeneous coordinate conversion matrix)

$$H_\sigma = \frac{1}{\sigma} [I, 0]$$

จะสังเกตได้ว่า พิกัดโฮโมจีเนียส $[q]^F$ มีได้หลายค่า ขึ้นอยู่กับสเกลาร์แฟกเตอร์ $\sigma \neq 0$ ที่ใช้ แต่จะให้ผลเป็นเวกเตอร์ 3 มิติ q เดียวกัน ในหุ่นยนต์มักใช้ $\sigma = 1$ ในกรณีนี้เราหาพิกัดโฮโมจีเนียสจากพิกัด 3 มิติได้โดยเติม 1 ให้เป็นสมาชิกตัวที่ 4 ในทำนองเดียวกันเราจะแปลงพิกัดโฮโมจีเนียสจากพิกัด 4 มิติกลับเป็น 3 มิติ ได้ด้วยตัวสมาชิกตัวที่ 4 ทั้ง

ถ้าจุดในปริภูมิ 3 มิติ ถูกแสดงในเทอมของพิกัดโฮโมจีเนียส และเราต้องการเปลี่ยนโครงพิกัดหนึ่งไปยังโครงพิกัดอื่น เราจะใช้เมตริกซ์ถ่ายโอนโฮโมจีเนียส (Homogeneous transformation matrix) ขนาด 4×4 โดยปกติเมตริกซ์ T นี้แปรได้เป็นเมตริกซ์ย่อย 4 ส่วนคือ

$$T \cong \begin{bmatrix} R & p \\ \eta^T & \sigma \end{bmatrix}$$

โดย R คือ Rotation matrix แสดงการหมุนของโครงพิกัดเคลื่อนที่เทียบกับโครงพิกัดอยู่นิ่ง p คือ Translation vector แสดงตำแหน่งจุดกำเนิดของโครงพิกัดเคลื่อนที่เทียบกับโครงพิกัดอยู่นิ่ง σ คือ Scalar factor ที่ไม่เป็นศูนย์ ปกติให้เท่ากับหนึ่ง η^T คือ Perspective vector ปกติเท่ากับศูนย์

ในเทอมของแกนกล p แสดงตำแหน่งของปลายเครื่องมือ, R แสดงการหมุนของเครื่องมือ, η แสดงจุดในการมองของอุปกรณ์ เช่น กล้องจับภาพ ในกรณีนี้จะให้ค่าที่ไม่เป็นศูนย์, σ ปกติใช้หนึ่ง เป็นค่ามาตรฐาน

การเลื่อนตำแหน่งและการหมุน (Translations and Rotations)

การกระทำพื้นฐานของการเลื่อนตำแหน่งและการหมุน จัดว่าเป็นกรณีพิเศษในเมตริกซ์ถ่ายโอนโฮโมจีเนียส 4×4 เช่น กรณีการหมุนสมมติให้ F และ M เป็นโครงพิกัดฉากที่เริ่มต้นทับกันสนิท ถ้าหมุน M ไป ϕ รอบเวกเตอร์ลำดับที่ k ของ F ดังนั้น จะแสดงในเทอมของพิกัดโฮโมจีเนียส เป็นเมตริกซ์ 4×4 , $Rot(\phi, k)$ ได้ว่า

$$Rot(\phi, k) \cong \begin{bmatrix} & & & 0 \\ & R_k(\phi) & & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & & 1 \end{bmatrix} \quad 1 \leq k \leq 3$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย $R_k(\phi)$ เป็นเมตริกซ์การหมุนพื้นฐานลำดับที่ k เราเรียก $\text{Rot}(\phi, k)$ ว่าเป็นเมตริกซ์การหมุนโฮโมจีเนียสพื้นฐานลำดับที่ k เมตริกซ์การหมุนผสมจะสามารถสร้างขึ้นได้ในทำนองเดียวกับ อัลกอริทึมการหมุนผสม จะสังเกตเห็นว่า เวกเตอร์ตำแหน่ง p ในแต่ละกรณีเป็นศูนย์

พิกัดโฮโมจีเนียสยังมีประโยชน์ในการแสดงการเลื่อนตำแหน่งได้ เช่น ให้ F และ M เป็นโครงพิกัดฉากที่ทับกันสนิท สมมติให้เลื่อนจุดกำเนิดของโครงพิกัดเคลื่อนที่ M ไป p_k ตามแนวยูนิตเวกเตอร์ลำดับที่ k ของ F เมื่อ $1 \leq k \leq 3$ ดังนั้นในเทอมของพิกัดโฮโมจีเนียสสามารถแสดงได้ด้วยเมตริกซ์ 4×4 , $\text{Tran}(p)$ เป็น

$$\text{Tran}(p) \cong \left[\begin{array}{ccc|c} 1 & 0 & 0 & p_1 \\ 0 & 1 & 0 & p_2 \\ 0 & 0 & 1 & p_3 \\ \hline 0 & 0 & 0 & 1 \end{array} \right]$$

เราเรียก $\text{Tran}(p)$ ว่าเป็นเมตริกซ์เลื่อนตำแหน่งโฮโมจีเนียสพื้นฐาน (Fundamental homogeneous translation matrix) จะสังเกตเห็นว่าเวกเตอร์ตำแหน่งใน p ในขณะที่เมตริกซ์การหมุน 3×3 จะเป็นเมตริกซ์เอกลักษณ์ I

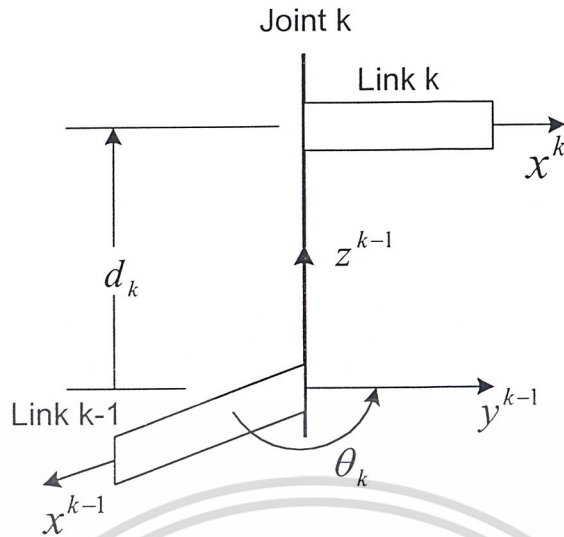
พิกัดลิงค์ (Link Coordinate)

ดังได้กล่าวแล้วว่า แขนกลประกอบด้วย link หลายๆ อันมาเชื่อมกันด้วยข้อต่อ จุดประสงค์ของหัวข้อนี้คือ การกำหนดโครงพิกัดลงบนแต่ละ link จากนั้นจึงทำการหาสมการแขน (Arm equation) ที่แสดงการเคลื่อนที่ของ link ของแขนกล โดยขั้นแรกจะศึกษาพารามิเตอร์ที่ใช้ในการออกแบบตัวแขนกลก่อน

Kinematics Parameters

ตำแหน่งและการหมุนสัมพัทธ์ของ link ที่อยู่ติดกัน แสดงได้ด้วย joint parameters 2 ตัว ดังแสดงในรูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.17 มุมข้อต่อ θ และระยะข้อต่อ d

จากรูปข้อต่อ k เชื่อม link $k-1$ พารามิเตอร์ที่เกี่ยวข้องกับข้อต่อ k ถูกนิยามเทียบกับแกน z^{k-1} ซึ่งอยู่ในแนวเดียวกับแกนของข้อต่อ k joint parameter ตัวแรกเรียกว่า มุมข้อต่อ (joint angle) θ_k ซึ่งเป็นการหมุนรอบ z^{k-1} เพื่อให้แกน x^{k-1} ขนานกับแกน x^k joint parameter ตัวที่สอง เรียกว่า ระยะข้อต่อ (joint distance) ; d_k ซึ่งเป็นการเลื่อนตามแนว z^{k-1} เพื่อให้แกน x^{k-1} ตัดกับแกน x^k ดังนั้น θ_k คือการหมุนรอบแกนข้อต่อ k และ d_k คือการเลื่อนไปตามแกนของข้อต่อ k ตามปกติแต่ละข้อจะมี joint parameter หนึ่งตัวมีค่าคงที่ และมีอีกค่าเปลี่ยนแปลงได้ ซึ่งขึ้นอยู่กับชนิดของข้อต่อ ดังแสดงในตารางที่ 2.5 สำหรับข้อต่อหมุนมุมข้อต่อ θ_k จะเปลี่ยนค่าได้ และระยะข้อต่อ d_k มีค่าคงที่ ดังนั้น link จะมีการหมุนสัมพันธ์กับรอบแกนของข้อต่อ k สำหรับข้อต่อเลื่อน ระยะข้อต่อ d_k จะเปลี่ยนค่าได้ และมุมข้อต่อ θ_k จะคงที่ ดังนั้น link จะมีการเลื่อนตำแหน่งสัมพันธ์กันไปตามแนวแกนของข้อต่อ k

Arm parameter	สัญลักษณ์	ข้อต่อหมุน (R)	ข้อต่อเลื่อน (P)
มุมข้อต่อ (Joint angle)	θ	ตัวแปร	คงที่
ระยะข้อต่อ (Joint distance)	d	คงที่	ตัวแปร
ความยาว link (Link length)	a	คงที่	คงที่
มุมบิด link (Link twist angle)	α	คงที่	คงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ [ตารางที่ 2.5 Kinematics parameters](#) มีอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อต่อจะอยู่ระหว่าง link 2 อันที่อยู่ติดกัน ในทางกลับกันจึงมี link 1 อันที่อยู่ระหว่างข้อต่อ 2 อัน ตำแหน่งและการหมุนสัมพัทธ์ของแกนของข้อต่อทั้ง 2 แสดงได้ด้วย link parameters ดังแสดงในรูปที่ 2.18 โดย link k เชื่อมข้อต่อ k กับข้อต่อ k+1 พารามิเตอร์ที่เกี่ยวข้องกับ link k ถูกนิยามเทียบกับ x^k ซึ่งเป็น common normal (ตั้งฉาก) ระหว่างแกนของข้อต่อ k และ k+1 link parameter ตัวแรกเรียกว่า ความยาว link (link length) a_k ซึ่งเป็นการเลื่อนตำแหน่งไปตามแนว x^k เพื่อให้แกน z^{k-1} ตัดกับ z^k link parameter ตัวที่สองเรียกว่า มุมบิด link (link twist angle) α_k ซึ่งเป็นการหมุนรอบ x^k เพื่อให้แกน z^{k-1} ขนานกับ z^k



รูปที่ 2.18 ความยาว link (link length) a_k และมุมบิด link (link twist angle) α_k

ตามปกติ link parameter ทั้งสองจะมีค่าคงที่ และถูกระบุเป็นส่วนหนึ่งของการออกแบบทางกล สำหรับหุ่นยนต์อุตสาหกรรม มุมบิด link (link twist angle) α_k มักเป็นจำนวนเท่าของ $\pi/2$ เจริญ บางครั้งแกนของข้อต่อ k และ k+1 จะตัดกันทำให้ความยาว link เป็นศูนย์ ตัวอย่างเช่นข้อต่อมือทรงกลม (spherical wrist) ของแขนกลที่มี n-3 แกนสุดท้ายตัดกัน นอกจากนี้หุ่นยนต์อาจถูกออกแบบให้มี kinematics parameters หลายๆ ตัวที่มีค่าคงที่และเท่ากับศูนย์ ซึ่งเรียกว่า แขนกลแบบ kinematically simple manipulators

สำหรับแขนกล n แกน ในแต่ละแกน พารามิเตอร์ 3 ตัวจะมีค่าคงที่ และขึ้นอยู่กับารออกแบบทางกล ในขณะที่พารามิเตอร์ตัวที่ 4 จะเป็นตัวแปรข้อต่อ (joint variable) ดังในตารางที่ 2.5 สำหรับหุ่นยนต์พิกัดฉาก ตัวแปรข้อต่อ 3 ตัวแรก จะเป็นระยะข้อต่อทั้งหมด ในขณะที่หุ่นยนต์ข้อต่อหมุน ตัวแปรข้อต่อ 3 ตัวแรก จะเป็นมุมข้อต่อทั้งหมด ส่วนหุ่นยนต์พิกัดทรงกระบอก, ทรงกลม และ SCARA ตัวแปร 3 ตัวแรก จะมีทั้งมุมข้อต่อและระยะข้อต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการแขน (The Arm Equation)

เมื่อกำหนดโครงฟัดลงบน link ได้แล้ว ก็ทำการย้ายโครงฟัด k ไปยังโครงฟัด $k-1$ โดยใช้เมตริกซ์การถ่ายโอนฟัดโฮโมจีเนียส เมื่อนำเมตริกซ์ที่ได้มาคูณกันทั้งหมด จะได้เมตริกซ์ที่แสดงการย้ายฟัดของเครื่องมือเทียบกับฟัดฐาน โดยเมตริกซ์ที่ได้นี้เรียกว่า เมตริกซ์แขน (Arm Matrix)

เมตริกซ์แขน (Arm Matrix)

การสร้างเมตริกซ์การถ่ายโอนฟัดโฮโมจีเนียสจากโครงฟัด k ไปยังโครงฟัด $k-1$ จะมี 4 ขั้นตอน แต่ละขั้นตอนจะเกี่ยวข้องกับ kinematics parameter หนึ่งตัว ในการพิจารณา เราจะทำการหมุนและเลื่อนตำแหน่งโครงฟัด $k-1$ จนทาบสนิทกับโครงฟัด k โดยกระทำตามขั้นตอนดังตารางที่ 2.6

ขั้นตอน (Operation)	คำอธิบาย (Description)
1	หมุน L_{k-1} รอบ z^{k-1} เป็นมุม θ_k
2	เลื่อน L_{k-1} ไปตามแนว z^{k-1} เป็นระยะ d_k
3	เลื่อน L_{k-1} ไปตามแนว x^{k-1} เป็นระยะ a_k
4	หมุน L_{k-1} รอบ x^{k-1} เป็นมุม α_k

ตารางที่ 2.6 การถ่ายโอนโครงฟัด k ไปยังโครงฟัด $k-1$

จากรูปที่ 2.17 จะได้ว่าขั้นตอนที่ 4 ทำให้แกน x^{k-1} ขนานกับแกน x^k ขั้นตอนที่ 2 แกน x^{k-1} จะทับกับ x^k ต่อมา จากรูปที่ 2.18 จะได้ว่าขั้นตอนที่ 3 ทำให้จุดกำเนิดของโครง L_{k-1} ทับกับ L_k ขั้นตอนที่ 4 ทำให้แกน z^{k-1} ทับกับ z^k ดังนั้นโครง L_{k-1} และ L_k ทับกันพอดี

การถ่ายโอนโครงฟัด (Link-Coordinate Transformation) ให้ $\{L_0, L_1, \dots, L_n\}$ เป็นเซตของโครงฟัด $[q]^k$ และ $[q]^{k-1}$ เป็นฟัดโฮโมจีเนียสของจุด q เทียบกับโครง L_k และ L_{k-1} สำหรับ $1 \leq k \leq n$ จะได้ $[q]^{k-1} = T_{k-1}^k [q]^k$, T_{k-1}^k คือการถ่ายโอนโครงฟัด k ไปยัง $k-1$

$$T_{k-1}^k = \begin{bmatrix} C\theta_k & -C\alpha_k S\theta_k & S\alpha_k S\theta_k & aC\theta_k \\ S\theta_k & C\alpha_k C\theta_k & -S\alpha_k C\theta_k & aS\theta_k \\ 0 & S\alpha_k & C\alpha_k & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังได้กล่าวมาแล้วว่า Kinematics parameters 3 ตัวที่ปรากฏใน T_{k-1}^k เป็นค่าคงที่ ตัวที่เหลือจะเป็นตัวแปรข้อต่อ ซึ่งอาจเป็น θ_k (สำหรับข้อต่อหมุน) หรือ d_k (สำหรับข้อต่อเลื่อน) จึงมีการกำหนดพารามิเตอร์ตัวหนึ่งขึ้นมาเรียกว่า joint type parameter โดย

$$\xi_k \equiv \begin{cases} 1 & \text{joint } k \text{ revolute} \\ 0 & \text{joint } k \text{ prismatic} \end{cases}$$

โดยใช้ ξ_k เราจะได้ตัวแปรข้อต่อลำดับที่ k : q_k ว่า

$$q_k \equiv \xi_k \theta_k + (1 - \xi_k) d_k$$

ตามปกติเมตริกซ์ถ่ายโอนโคจรพิกัดโฮโมจีเนียส T_{k-1}^k จะเป็นฟังก์ชันของ q_k สำหรับ $1 \leq k \leq n$ ในการแก้ปัญหา Direct kinematics เราต้องพิจารณาดำแหน่งและการหมุนของเครื่องมือเทียบกับโคจรพิกัดที่ติดอยู่กับฐาน การถ่ายโอนจากพิกัดเครื่องมือไปยังพิกัดฐานจะเริ่มจากปลายเครื่องมือถ่ายโอนย้อนกลับไปที่ละโคจรจนถึงฐาน ถ้า T_{base}^{tool} แสดงการถ่ายโอนจากพิกัดปลายเครื่องมือ (link n) ไปยังพิกัดฐาน (link 0) ดังนั้น

$$T_{base}^{tool}(q) = T_0^1(q_1) T_1^2(q_2) \dots T_{n-1}^n(q_n) = T_0^n(q)$$

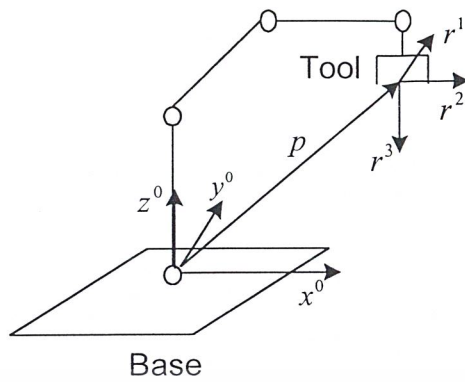
สมการแขน (The Arm Equation)

จาก T_{base}^{tool} เมื่อแทนค่าลงไป เราจะได้สมการแขนดังนี้

$$T_{base}^{tool}(q) = \begin{bmatrix} R(q) & p(q) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

โดย $R(q)$ เป็นเมตริกซ์ย่อย 3×3 แสดงการหมุนของเครื่องมือ คอลัมน์ทั้ง 3 ของ R แสดงทิศทางของยูนิตเวกเตอร์ $\{r^1, r^2, r^3\}$ ของโคจรเครื่องมือเทียบกับโคจรฐาน, $p(q)$ เป็นเมตริกซ์ย่อย 3×3 แสดงตำแหน่งของปลายเครื่องมือเป็นพิกัดของปลายเครื่องมือเทียบกับโคจรฐาน ดังแสดงในรูปที่ 2.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



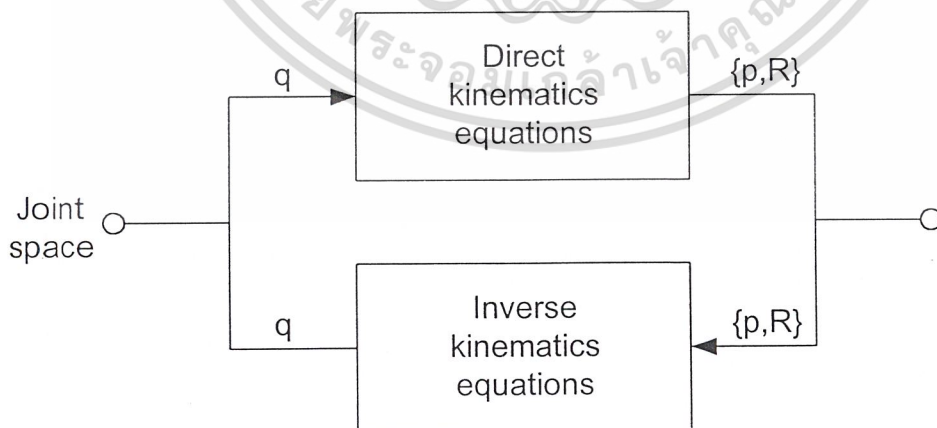
รูปที่ 2.19 ตำแหน่งและการหมุนของเครื่องมือในพิกัดฐาน

2.1.4 Inverse Kinematics : การแก้สมการแขน (Solving the Arm Equation)

ในหัวข้อที่แล้วเราได้ทำการหาตำแหน่งและการหมุนของเครื่องมือจากตัวแปรข้อต่อที่กำหนดให้ แต่ในหัวข้อนี้เราจะพิจารณาย้อนกลับ คือ การหาตัวแปรข้อต่อจากตำแหน่งและการหมุนของเครื่องมือที่กำหนดให้ ปัญหา Inverse Kinematics มีความสำคัญและนำไปใช้งานจริงในแขนกล ไม่มีขั้นตอนการแก้ปัญหาที่แน่นอนและมักมีหลายคำตอบที่ได้

คุณสมบัติทั่วไปของคำตอบ (General Properties of Solutions)

Inverse kinematics ดังรูปที่ 2.20 เป็นการกำหนดตำแหน่ง p และการหมุน R ของเครื่องมือเทียบกับโครงพิกัดอ้างอิงแล้วทำการคำนวณหาตัวแปรข้อต่อ q ที่สอดคล้องกับสมการแขน



รูปที่ 2.20 Direct และ Inverse kinematics

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การมีอยู่ของคำตอบ (Existence of Solutions)

พิจารณาในสถานะที่คำตอบของ Inverse kinematics มีจริง คือ ตำแหน่งปลายเครื่องมือ p อยู่ภายในขอบเขตการทำงานและการหมุนของเครื่องมือไม่เกินขีดจำกัดของตัวแปรข้อต่อ จากสมการแขน T_{base}^{tool}

$$T_{base}^{tool}(q) = \begin{bmatrix} R_{11} & R_{12} & R_{13} & p_1 \\ R_{21} & R_{22} & R_{23} & p_2 \\ R_{31} & R_{32} & R_{33} & p_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

เนื่องจากบรรทัดสุดท้ายของ T_{base}^{tool} มีค่าคงที่ ดังนั้นสมการแขนจึงเป็นระบบสมการพีชคณิตเชิงเส้น 12 สมการ มี q เป็นตัวแปร n ตัว แต่ทั้ง 12 สมการไม่เป็นอิสระซึ่งกันและกัน

$$R = \{r^1, r^2, r^3\} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

เมื่อ r^1, r^2, r^3 คือเวกเตอร์ในแนวแกนของโครงฟัดที่ปลายเครื่องมือเทียบกับโครงฟัดฐาน

R_{11} คือองค์ประกอบของ r^1 ในแนวแกน x ของโครงฟัดฐานอ้างอิง (f^1)

R_{21} คือองค์ประกอบของ r^2 ในแนวแกน y ของโครงฟัดฐานอ้างอิง (f^2)

R_{31} คือองค์ประกอบของ r^3 ในแนวแกน z ของโครงฟัดฐานอ้างอิง (f^3)

จาก $R^{-1} = R^T$ จะได้ว่า $R^T R = I$ เมื่อพิจารณาผลคูณของเทอมในแนวอนเส้นทแยงมุมของเมตริกซ์ จึงสรุปได้ว่า r^1, r^2, r^3 ตั้งฉากซึ่งกันและกัน

$$r^1 \cdot r^2 = 0$$

$$r^1 \cdot r^3 = 0$$

$$r^2 \cdot r^3 = 0$$

และพิจารณาผลคูณของเทอมในแนวเส้นทแยงมุม ได้ว่า r^1, r^2, r^3 เป็นยูนิตเวกเตอร์ จะได้ว่า $\|r^k\| = 1$ เมื่อ $1 \leq k \leq 3$

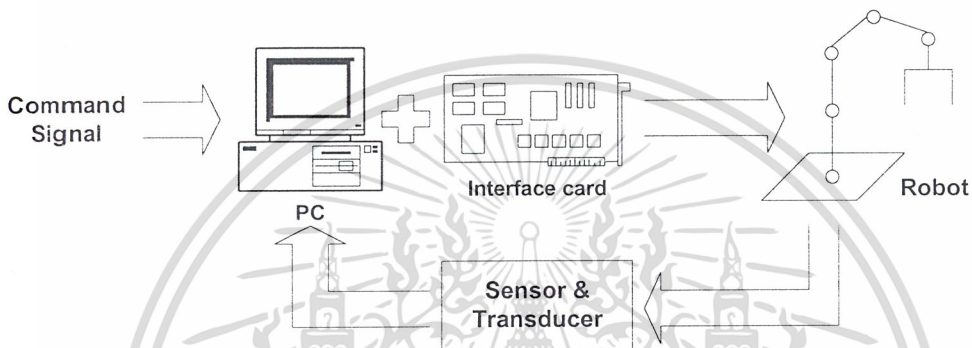
ดังนั้นจะได้ความสัมพันธ์มา 6 สมการ จากระบบสมการ 12 สมการจึงมีเพียง 6 สมการที่เป็นอิสระต่อกัน ถ้าเราต้องการคำตอบทั่วไปจึงควรมีตัวแปรข้อต่อ q อย่างน้อยที่สุด 6 ตัว, $q \geq 6$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 หลักการในการควบคุม

ในการควบคุมระบบการเคลื่อนที่ของมือกลให้ได้ตามต้องการนั้น การควบคุมที่เป็นสิ่งสำคัญที่สุดคือ การควบคุมการหมุนของมอเตอร์ ซึ่งเป็นอุปกรณ์ที่ทำให้ข้อนิ้วมือแต่ละข้อเคลื่อนที่ตามคำสั่งควบคุม

การควบคุมการทำงาน (การหมุน) ของมอเตอร์นั้นเป็นการควบคุมแบบดิจิทัล (Digital control) ซึ่งประกอบไปด้วย คอมพิวเตอร์, ชุดควบคุม, มอเตอร์และเซ็นเซอร์ ดังบล็อกไดอะแกรมรูปที่ 2.21



รูปที่ 2.21 แสดงบล็อกไดอะแกรมโดยรวมของระบบควบคุมการทำงานของมือกล

2.3 โครงสร้างของระบบ

สามารถแบ่งโครงสร้างของระบบออกเป็น 3 ส่วนดังนี้

1) ส่วนตรวจจับ (Sensor & Encoder)

เป็นส่วนประกอบที่สำคัญในระบบที่มีการป้อนกลับ Sensor & Encoder ที่ใช้กันมากในระบบควบคุมเช่น Potentiometer, Optical encoder, Resolver, Strain gauge เป็นต้น โดยสัญญาณอนาล็อกจากส่วนตรวจจับนี้จะถูกแปลงเป็นสัญญาณดิจิทัล หลังจากนั้นจะถูกส่งไปยังหน่วยประมวลผลกลางต่อไป

2) หน่วยประมวลผลกลางและควบคุม

ได้แก่คอมพิวเตอร์และการ์ดอินเทอร์เฟซที่ออกแบบขึ้นเพื่อใช้ในระบบควบคุมการเคลื่อนที่ของมือกล โดยใช้ ISA Card เป็นตัวเชื่อมต่อกับระบบบัสของคอมพิวเตอร์ มีวงจรควบคุมต่างๆ เช่น วงจรดีโคด I/O แอดเดรสที่จะใช้, วงจรที่ทำหน้าที่รับและส่งข้อมูล เป็นต้น

3) วงจรขับมอเตอร์

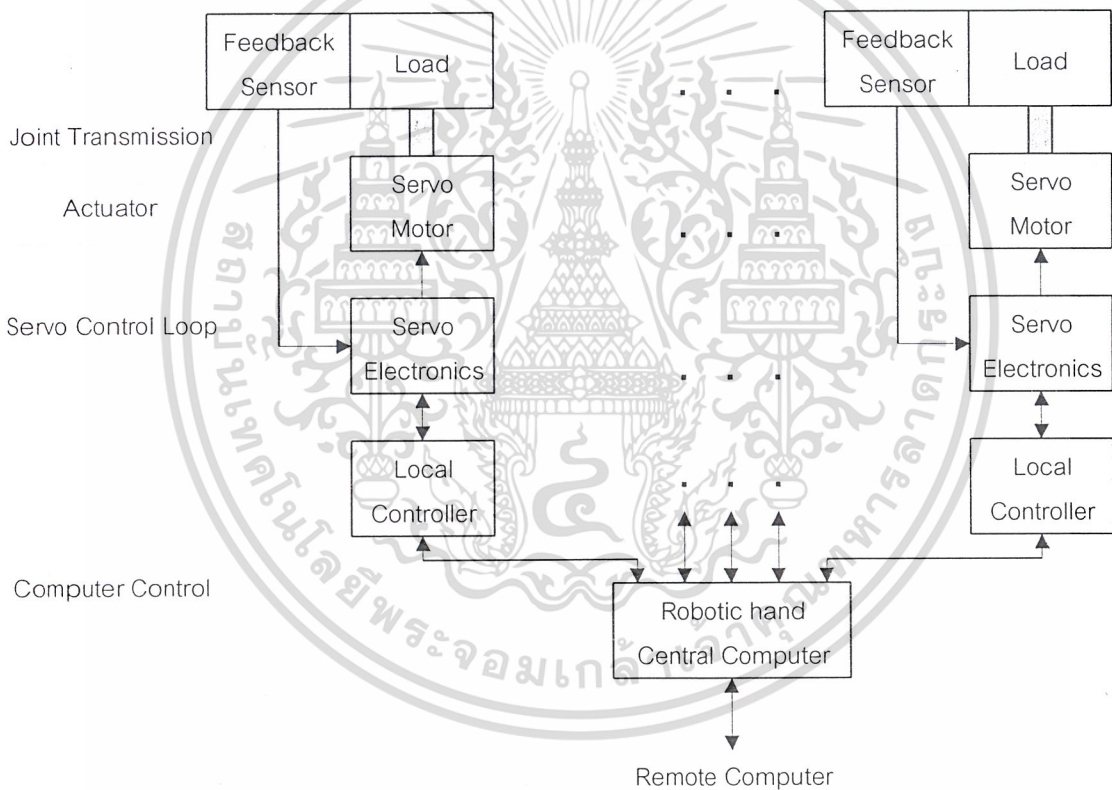
เป็นวงจรที่ควบคุมการหมุนของมอเตอร์ซึ่งเป็นอุปกรณ์ที่ทำให้มือกลเกิดการเคลื่อนไหว มอเตอร์จะถูกติดตั้งที่ข้อต่อต่างๆของนิ้วมือแต่ละนิ้วเพื่อควบคุมการเคลื่อนที่ของข้อนิ้วมือ ซึ่งรายการค่าไม่ถูกรวบรวมขึ้น อีกทั้งยังเป็นให้ข้อมูลป้อนกลับของแรงบิดของมอเตอร์ทุกครั้งที่มีการนำไปใช้

ละเอียดของวงจรขับมอเตอร์และโครงสร้างทางกลของมือกลจะกล่าวในบทต่อไป

2.4 วงจรเชื่อมต่อระหว่างวงจรควบคุมกับคอมพิวเตอร์ (Interface Card)

Interface card เป็นการ์ดวงจรที่ออกแบบขึ้นเพื่อใช้ในระบบควบคุมการเคลื่อนที่ของมือกล โดยวงจรเชื่อมต่อนี้จะทำหน้าที่เป็นเหมือนพอร์ต (Port) ,บัฟเฟอร์ของสัญญาณข้อมูล (Data Buffer) และบัฟเฟอร์ของสัญญาณควบคุม (Signal Buffer) ซึ่งช่วยให้คอมพิวเตอร์สามารถติดต่อ, ส่งผ่านและรับข้อมูลกับวงจรควบคุมมอเตอร์, เซ็นเซอร์และทรานส์ดิวเซอร์ได้

การควบคุมอุปกรณ์ภายนอกโดยใช้เครื่องคอมพิวเตอร์นั้นจะต้องทำให้เครื่องคอมพิวเตอร์ติดต่อกับอุปกรณ์ภายนอกให้ได้เสียก่อน แต่การที่จะทำเช่นนั้นได้ต้องผ่าน พอร์ต (Port) ซึ่งมีหลายลักษณะด้วยกัน โดยในการ์ดนี้จะแบ่งเป็น 2 ส่วนใหญ่ด้วยกัน ซึ่งจะเป็นส่วนของวงจร Decode และส่วนของ Port ซึ่งจะอธิบายรายละเอียดในบทต่อไป



รูปที่ 2.22 แสดงวงจรควบคุมการทำงาน Robotics Hand

2.5 หลักการทำงานของมอเตอร์ไฟฟ้ากระแสตรง

มอเตอร์ไฟฟ้ากระแสตรงเป็นทรานส์ดิวเซอร์แรงบิด ซึ่งมีการออกแบบให้มีคุณลักษณะพิเศษ คือ แรงบิดของเพลามอเตอร์ไฟฟ้ากระแสตรงกับกระแสอาร์เมเจอร์ แรงบิดของเพลามอเตอร์จะได้จากผลระหว่างสนามแม่เหล็กและขดลวดตัวนำ หลักการในที่นี้กระแสที่ไหลในขดลวดเป็นเอกซอสาร์ที่ส่งผ่านไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำเอาไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลวดตัวนำจะสร้างฟลักซ์ที่ประกอบด้วยเส้นแรงแม่เหล็ก และขอลวดตัวนำเหล่านั้นอยู่ห่างจากศูนย์กลางการหมุนเท่ากับ r ความสัมพันธ์ระหว่างแรงบิดของเพลลาและกระแสเท่ากับ

$$T = K \phi I$$

เมื่อ T คือแรงบิดของเพลลา มีหน่วยเป็น นิวตัน-เมตร

ϕ คือเส้นแรงแม่เหล็ก มีหน่วยเป็น เวเบอร์

I คือกระแส มีหน่วยเป็น แอมแปร์

K คือตัวคงที่

ดังนั้นแรงบิดของเพลลาจะเป็นสัดส่วนโดยตรงกับผลคูณของเส้นแรงแม่เหล็ก และกระแสเมื่อขดลวดตัวนำเคลื่อนที่ในสนามแม่เหล็กก็จะทำให้เกิดโวลต์เตจตกคร่อมตัวมันเอง โวลต์เตจนี้จะเป็นสัดส่วนกับความเร็วของเพลลาของมอเตอร์และด้านารไหลของกระแส ความสัมพันธ์ระหว่างโวลต์เตจย้อนกลับนี้และความเร็วของเพลลามอเตอร์ คือ

$$E = K \phi W$$

เมื่อ E คือโวลต์เตจย้อนกลับ emf มีหน่วยเป็น โวลต์

ϕ คือเส้นแรงแม่เหล็ก มีหน่วยเป็น เวเบอร์

W คือความเร็วของมอเตอร์ มีหน่วยเป็น เรเดียน/วินาที

2.5.1 การแยกประเภทของมอเตอร์ไฟฟ้ากระแสตรง

มอเตอร์ไฟฟ้ากระแสตรงสามารถแบ่งออกได้เป็นหลายประเภทขึ้นอยู่กับพื้นฐานการออกแบบโครงสร้างของอาร์เมเจอร์ การแบ่งประเภทตามลักษณะการจ่ายสนามแม่เหล็กแยกออกได้เป็น 2 แบบ คือ

1. มอเตอร์ไฟฟ้ากระแสตรงแบบปรับเส้นแรงแม่เหล็กได้
2. มอเตอร์ไฟฟ้ากระแสตรงแบบเส้นแรงแม่เหล็กมีค่าคงที่

ถ้าเราจะพิจารณาแยกประเภทตามลักษณะการออกแบบโครงสร้างอาร์เมเจอร์สามารถแยกออกได้เป็น 3 แบบ คือ

1. มอเตอร์ไฟฟ้ากระแสตรงแบบอาร์เมเจอร์เป็นแกนเหล็ก
2. มอเตอร์ไฟฟ้ากระแสตรงแบบอาร์เมเจอร์ที่มีขดลวดพันอยู่บนพื้นผิว
3. มอเตอร์ไฟฟ้ากระแสตรงแบบอาร์เมเจอร์เป็นขดลวดหมุน

นอกจากนี้ยังมีมอเตอร์ไฟฟ้ากระแสตรงชนิดพิเศษอีกแบบหนึ่งคือ แบบไม่มีแปรงถ่าน ซึ่ง

มีหลักการทางเทคโนโลยีเหมือนกับมอเตอร์ไฟฟ้ากระแสตรงชนิดมีแปรงถ่านยกเว้น การคอมมิวเต

ชั่นกระทำโดยเทคนิคทางอิเล็กทรอนิกส์แทนที่จะกระทำโดยวิธีการทางเชิงกล ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 พื้นฐานของระบบควบคุมดิซีมอเตอร์

ส่วนประกอบพื้นฐานของระบบคอนโทรลดิซีมอเตอร์ประกอบด้วยบล็อกที่สำคัญ 4 บล็อก คือ

1. ตัวคอนโทรลเลอร์
2. วงจรไดรเวอร์หรือเพาเวอร์แอมพลิไฟ
3. ฟีดแบ็คทรานสดิวเซอร์หรือเอนโคดเดอร์
4. ดิซีมอเตอร์และโหลด

ตัวคอนโทรลเลอร์ เป็นส่วนของระบบที่ทำให้เกิดสัญญาณคอนโทรลไปยังคั้งดิซีมอเตอร์และโหลด คอนโทรลเลอร์ที่ให้สัญญาณคอนโทรลเป็นสัญญาณอนาลอก เราเรียกว่าอนาลอกคอนโทรลเลอร์ ส่วนคอนโทรลเลอร์ที่ให้สัญญาณคอนโทรลเป็นสัญญาณดิจิตอล เราเรียกว่าดิจิตอลคอนโทรลเลอร์

วงจรไดรเวอร์ เป็นส่วนประกอบของระบบที่อยู่ระหว่างตัวคอนโทรลเลอร์กับดิซีมอเตอร์และโหลดมีหน้าที่ปรับรูปและขยายสัญญาณให้เหมาะสมก่อนที่จะป้อนเข้าไปขับดิซีมอเตอร์และโหลด วงจรไดรเวอร์ส่วนใหญ่ ได้แก่ เพาเวอร์แอมพลิไฟซึ่งอาจแบ่งย่อยออกเป็นลิเนียร์เพาเวอร์แอมพลิไฟและพัลส์วิดโมดูเลชั่นแอมพลิไฟซึ่ง

ฟีดแบ็คทรานสดิวเซอร์หรือเอนโคดเดอร์ เป็นสิ่งประดิษฐ์ที่ใช้รับรู้หรือดีเทคสัญญาณเอาท์พุทที่ต้องการ โดยไม่มีผลของการโหลดดิ่ง (loading) สัญญาณที่ดีเทคได้นี้จะป้อนกลับไปเปรียบกับสัญญาณอ้างอิงทำให้ได้สัญญาณเออร์เรอร์ ฟีดแบ็คทรานสดิวเซอร์แบ่งออกได้เป็น 2 แบบ คือ อนาลอกทรานสดิวเซอร์ คือสิ่งประดิษฐ์ใช้เปลี่ยนพลังงานรูปหนึ่งให้เป็นสัญญาณอนาลอก ได้แก่ พวุกทาโคเยนเนอเรเตอร์ โพเทนทิโอมิเตอร์และชิ่งโคร เป็นต้น ส่วนฟีดแบ็คทรานสดิวเซอร์อีกแบบหนึ่งคือ ดิจิตอลทรานสดิวเซอร์ เป็นสิ่งประดิษฐ์ที่ใช้เปลี่ยนพลังงานรูปหนึ่งให้เป็นสัญญาณดิจิตอล ได้แก่ พวุกอินคริเม้นท์เอนโคดเดอร์รีโซลเวอร์ แมกนิติคพิคอัพ เป็นต้น

ดิซีมอเตอร์และโหลด คือระบบที่ถูกคอนโทรลหรือส่วนที่ออกแรงทำงานซึ่งจะเป็นเครื่องจักรกล (ดิซีมอเตอร์) หรืออะไรก็ตามที่ให้ตัวแปร ดิซีมอเตอร์ในที่นี่เป็นแบบแม่เหล็กถาวรที่มีคุณสมบัติการทำงานสูง มีอาร์เมเจอร์อินดักเต้นซ์และแรงเฉื่อยของโรเตอร์ต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 ลักษณะการควบคุมของระบบดีซีเอ็มเอเตอร์

ระบบการคอนโทรลดีซีเอ็มเอเตอร์สามารถที่จะจำแนกลักษณะการคอนโทรลออกได้เป็น 2 แบบ คือระบบนาลอกคอนโทรลและระบบดิจิตอลคอนโทรล ในโครงการนี้ใช้ลักษณะการควบคุมแบบดิจิตอล จึงขออธิบายแต่ระบบดิจิตอลคอนโทรล

ระบบดิจิตอลคอนโทรล

ระบบดิจิตอลคอนโทรล คือ ระบบที่คอนโทรลเลอร์มีพีดีแบริคทรานสดิวเซอร์เป็นดิจิตอลทรานสดิวเซอร์ซึ่งสามารถวัดค่าไดนามิกวาริเอเบิลออกเป็นสัญญาณดิจิตอลหรือในรูปของสัญญาณแอนโคดดิ้งของไบนารี คือ สัญญาณเอาท์พุทของดิจิตอลทรานสดิวเซอร์จะเป็นไค้ดไบนารีที่ป้อนกลับไปยังตัวดิจิตอลคอนโทรลเลอร์เพื่อคอนโทรลให้คุณสมบัติการทำงานของระบบเป็นไปตามที่ต้องการ

ตัวอย่างการประยุกต์ใช้ในงานอุตสาหกรรมของระบบดิจิตอลดีซีเอ็มเอเตอร์คอนโทรลมีดังนี้

- เครื่องจักรกล NC
- หุ่นยนต์ที่ใช้ในอุตสาหกรรม
- ระบบการป้อนกระดาษของไลน์ปริ้นเตอร์
- อินกรีเมนต์เทปเรคคอร์ดเดอร์
- เครื่องอ่านและปรูเทป
- เครื่องอ่านและปรูการ์ด
- ระบบการขับเคลื่อนเทปในเครื่องคอมพิวเตอร์
- คอนโทรลแคเรจในเวิร์ดโพรเซสเซอร์
- คอนโทรลล้อพิมพ์ในเวิร์ดโพรเซสเซอร์
- แร็นดัมแอ็คเซสดีสค์เม็มโมรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

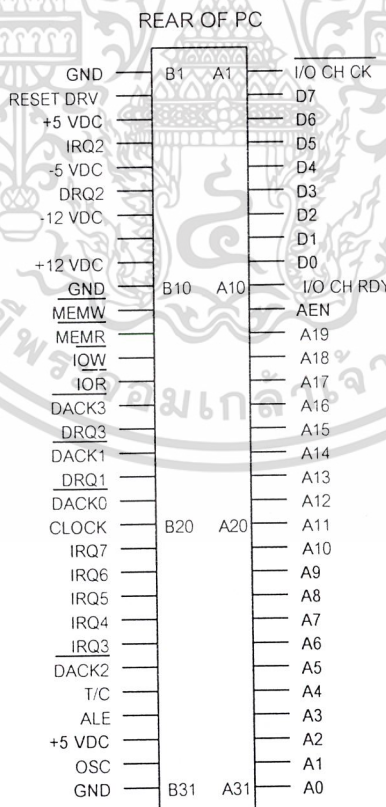
บทที่ 3

โครงสร้างและการออกแบบส่วนอิเล็กทรอนิกส์

3.1 วงจรอินเทอร์เฟส

การควบคุมอุปกรณ์ภายนอกโดยใช้เครื่องคอมพิวเตอร์นั้นจะต้องทำให้เครื่องคอมพิวเตอร์ติดต่อกับอุปกรณ์ภายนอกให้ได้เสียก่อน แต่การที่จะทำเช่นนั้นได้ต้องผ่าน พอร์ต (Port) ซึ่งมีหลายลักษณะด้วยกัน แต่ในที่นี้จะใช้ไอซีเบอร์ 74LS374 เป็นพอร์ตขนานและทำมาในรูปของการ์ด โดยในการ์ดนี้จะแบ่งเป็น 2 ส่วนใหญ่ด้วยกัน ซึ่งจะเป็นส่วนของวงจร Decode และส่วนของ Port ซึ่งจะอธิบายในรายละเอียดต่อไป

ที่เครื่องคอมพิวเตอร์นั้นจะมีช่องเสียบเรียกว่า SLOT PC โดยการเอาการ์ดมาเสียบที่ SLOT นี้ เพื่อเป็นการเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์ภายนอก โดยขาต่างของ SLOT (IBM PC system bus) แสดงดังรูปที่ 3.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามรูปที่ 3.1 แสดง IBM PC system bus ขาของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณที่ออกจาก IBM PC system bus ที่ใช้กับการ์ด มีดังนี้

A0 - A9 เป็นแอดเดรสของระบบ ที่ใช้ติดต่อกับหน่วยความจำและอุปกรณ์อินพุท/เอาต์พุท

D0 - D7 เป็นสัญญาณข้อมูลขนาด 8 บิต ที่ใช้ติดต่อกับหน่วยความจำไมโครโพรเซสเซอร์

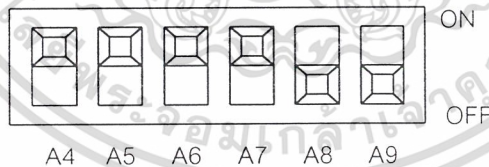
IOW เป็นสัญญาณเขียนข้อมูลลงบนอุปกรณ์อินพุท/เอาต์พุท สัญญาณนี้ควบคุมโดยไมโครโพรเซสเซอร์ และแอกทีฟที่ "0"

IOR เป็นสัญญาณอ่านข้อมูลจากอุปกรณ์อินพุท/เอาต์พุท ควบคุมโดยไมโครโพรเซสเซอร์ และแอกทีฟที่ "0"

AEN อีนาเบิลแอดเดรส (เป็นเอาต์พุท)

เมื่อรู้สัญญาณต่างๆที่ออกมาจากเครื่องคอมพิวเตอร์แล้ว ก็มาดูรายละเอียดของวงจรถนการด์บ้าง ซึ่งมี 2 ส่วนดังที่กล่าวมาแล้วในตอนต้น แต่ทั้ง 2 ส่วนจะทำงานร่วมกัน ส่วนของวงจรถน Decode ซึ่งในการ์ดนี้ประกอบด้วยไอซีเบอร์ 74LS688 (8-bit Equal to Comparator) และ DIP Switch เพื่อที่จะสามารถปรับ DIP Switch ตั้งตำแหน่งพอร์ทของการ์ดได้ โดยเพียงแค่ปรับที่ DIP Switch เท่านั้น สิ่งที่ต้องระวังคือ ในการปรับ DIP Switch นั้น จะต้องไม่ให้ตรงกับตำแหน่งพอร์ทที่เครื่องคอมพิวเตอร์ใช้อยู่แล้ว โดย I/O Port Address สำหรับ Prototype card มีช่วงตั้งแต่ 300H ถึง 31FH

การ Decode Address 300H = 0011 0000 XXXX B ดังนั้นจึงปรับตั้ง DIP Switch สำหรับ Decode Address ดังรูปที่ 3.2



รูปที่ 3.2 แสดงการปรับตั้ง DIP Switch

3.2 การส่งสัญญาณควบคุม

สัญญาณควบคุมหรือข้อมูลที่ได้จากการประมวลผลโดยคอมพิวเตอร์ จะถูกส่งผ่านออกไปยังบัสข้อมูล (Data Bus) พอร์ท 300 ถึง พอร์ท 303 โดยมี IC เบอร์ 74LS374 เป็นฟลิปฟลอป ซึ่งจะทำงานก็ต่อเมื่อมีสัญญาณเลือกจาก IC 74LS138 และสัญญาณ IOW ดังนั้นเมื่อมีสัญญาณ

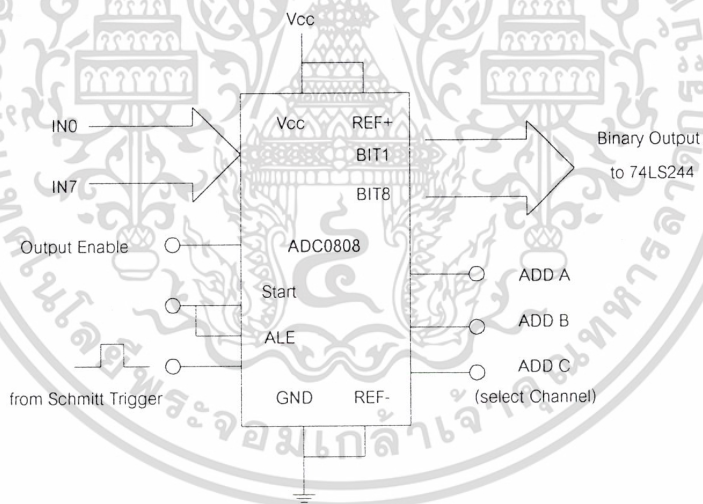
เลือกจาก IC 74LS138 และสัญญาณ IOW ไปยัง 74LS374 ตัวใด ก็จะทำให้เกิดการส่งข้อมูลจากไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัสข้อมูล ออกไปยังพอร์ทนั้นๆ สัญญาณข้อมูลนั้นถูกส่งไปยังวงจรขั้วมอเตอร์ เพื่อควบคุมการเคลื่อนที่ของมือกลต่อไป

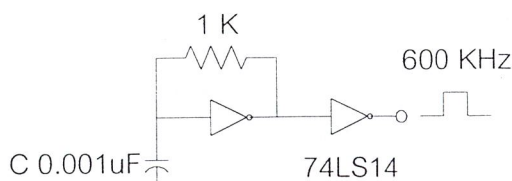
3.3 การรับสัญญาณข้อมูล

การรับสัญญาณข้อมูลหรือสัญญาณป้อนกลับ (Feedback signals) ใช้ IC ADC0808 (Analog to Digital Converter) ทำหน้าที่แปลงสัญญาณ analog ที่รับจากเซ็นเซอร์ต่างๆ ให้เป็นสัญญาณ digital เพื่อนำไปประมวลผลด้วยคอมพิวเตอร์ มีขั้นตอนการทำงานคือ เมื่อมีสัญญาณเลือกจาก 74LS138 และ IOR ไปยัง 74LS244 และสัญญาณควบคุมออกจากพอร์ท 303 ผ่านทาง 74LS373 ไปยัง ADC0808 เพื่อเลือก channel จะทำให้ ADC0808 ทำการแปลงข้อมูล analog เป็นข้อมูล digital และข้อมูลที่แปลงแล้วนั้นจะถูกส่งผ่าน Bus driver IC 74LS244 เข้าสู่บัสข้อมูล (Data Bus) และนำไปประมวลผลต่อไป

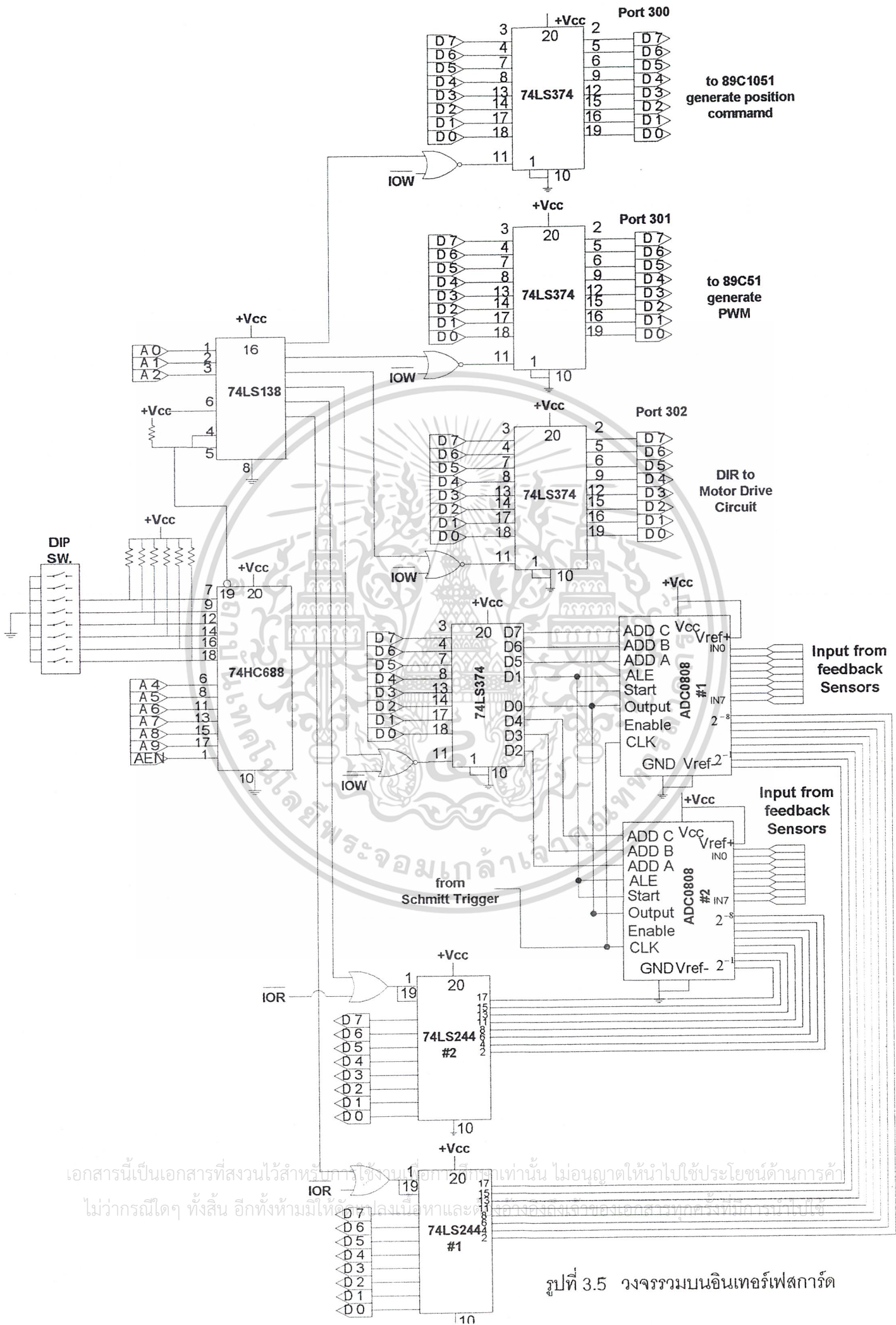
วงจร Schmitt Trigger ดังรูปที่ 3.4 เป็นวงจรที่สร้างสัญญาณพัลส์ให้แก่ ADC0808 เพื่อนำไปใช้เป็นคาบของการสุ่มสัญญาณ (Sampling period)



รูปที่ 3.3 แสดง ADC0808 และขาที่ใช้งาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ลิขสิทธิ์นี้สงวนไว้ให้ด้วย และจะสงวนไว้ให้ด้วย
รูปที่ 3.4 แสดงวงจร Schmitt Trigger กำหนดพัลส์ความถี่ 600 MHz ซึ่งมีการนำไปใช้



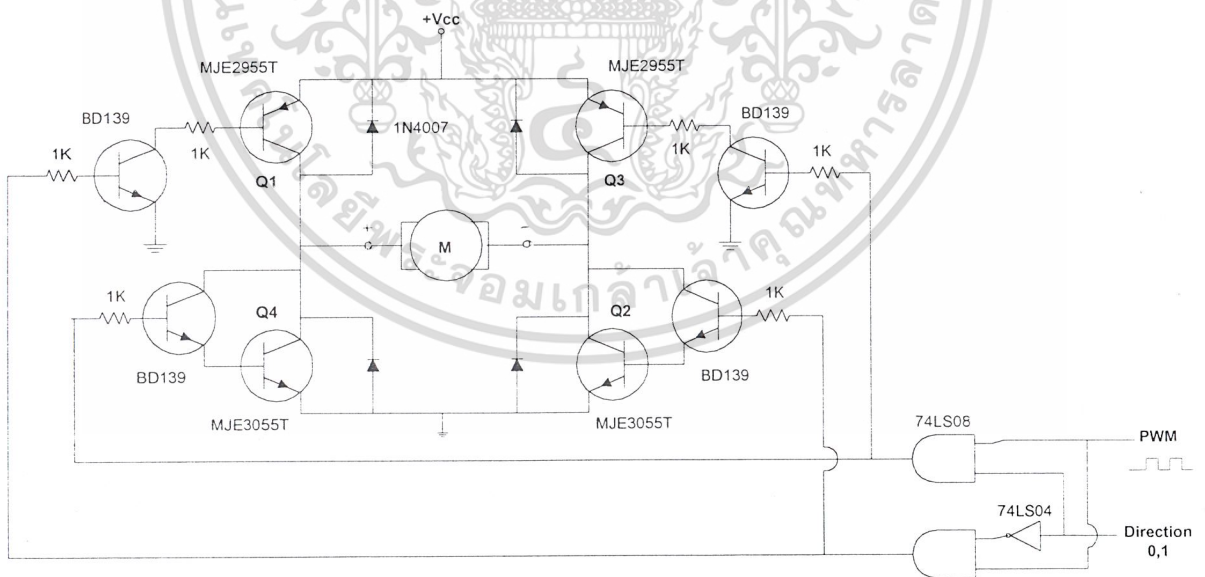
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและข้อมูลของเอกสารทุกครั้งที่มีการแก้ไข

รูปที่ 3.5 วงจรรวมบนอินเทอร์เฟสการ์ด

3.4 วงจรขับมอเตอร์

จากรูปที่ 3.5 วงจรขับเคลื่อนและควบคุมทิศทางมอเตอร์เป็นอินเวอร์เตอร์แบบบริดจ์เต็มเฟสเดียว (Single Phase Full-Bridge Inverters) ประกอบด้วยทรานซิสเตอร์ MJE2955T 2 ตัว เมื่อทรานซิสเตอร์ Q1 , Q2 ON ในเวลาเดียวกันด้วยสัญญาณพัลส์และทิศทางจากคอมพิวเตอร์ จะทำให้เกิดแรงดันอินพุท V_{cc} ต่อกะโรมอเตอร์ ถ้าทรานซิสเตอร์ Q3 , Q4 ON ในเวลาเดียวกัน แรงดันต่อกะโรมอเตอร์จะกลับขั้วเป็น $-V_{cc}$ ทำให้มอเตอร์หมุนทิศทางตรงกันข้ามกับในกรณีที่ทรานซิสเตอร์ Q1 , Q2 ON

สัญญาณควบคุมทิศทางจากคอมพิวเตอร์ จะส่งผ่านมายังแอนด์เกต 74LS08 ทำให้เกิดเป็นสัญญาณควบคุมการสวิตช์โดยการ ON Q1,Q2 ในเวลาเดียวกัน หรือ ON Q3,Q4 ในเวลาเดียวกัน โดยถ้าส่งลอจิก "0" ออกมาจะทำให้ Q1,Q2 ON ในเวลาเดียวกัน ถ้าส่งลอจิก "1" ออกมาจะทำให้ Q3,Q4 ON ในเวลาเดียวกัน ทำให้มอเตอร์หมุนกลับทิศทางกัน ส่วนไดโอดป้องกันกลับ 1N4007 นั้น (Feedback Diode) ต่อเพื่อลดการกระชากของกระแส ป้องกันความเสียหายที่จะเกิดขึ้นแก่อุปกรณ์ สำหรับความเร็วในการหมุนมอเตอร์จะควบคุมโดยสัญญาณพัลส์ PWM ที่ส่งมาจากคอมพิวเตอร์ ถ้าพัลส์มีช่วง ON มาก (Duty cycle มาก) มอเตอร์จะหมุนเร็ว ถ้าพัลส์มีช่วง ON น้อย (Duty cycle น้อย) มอเตอร์จะหมุนช้าลง



รูปที่ 3.6 วงจรขับเคลื่อนและควบคุมทิศทางมอเตอร์

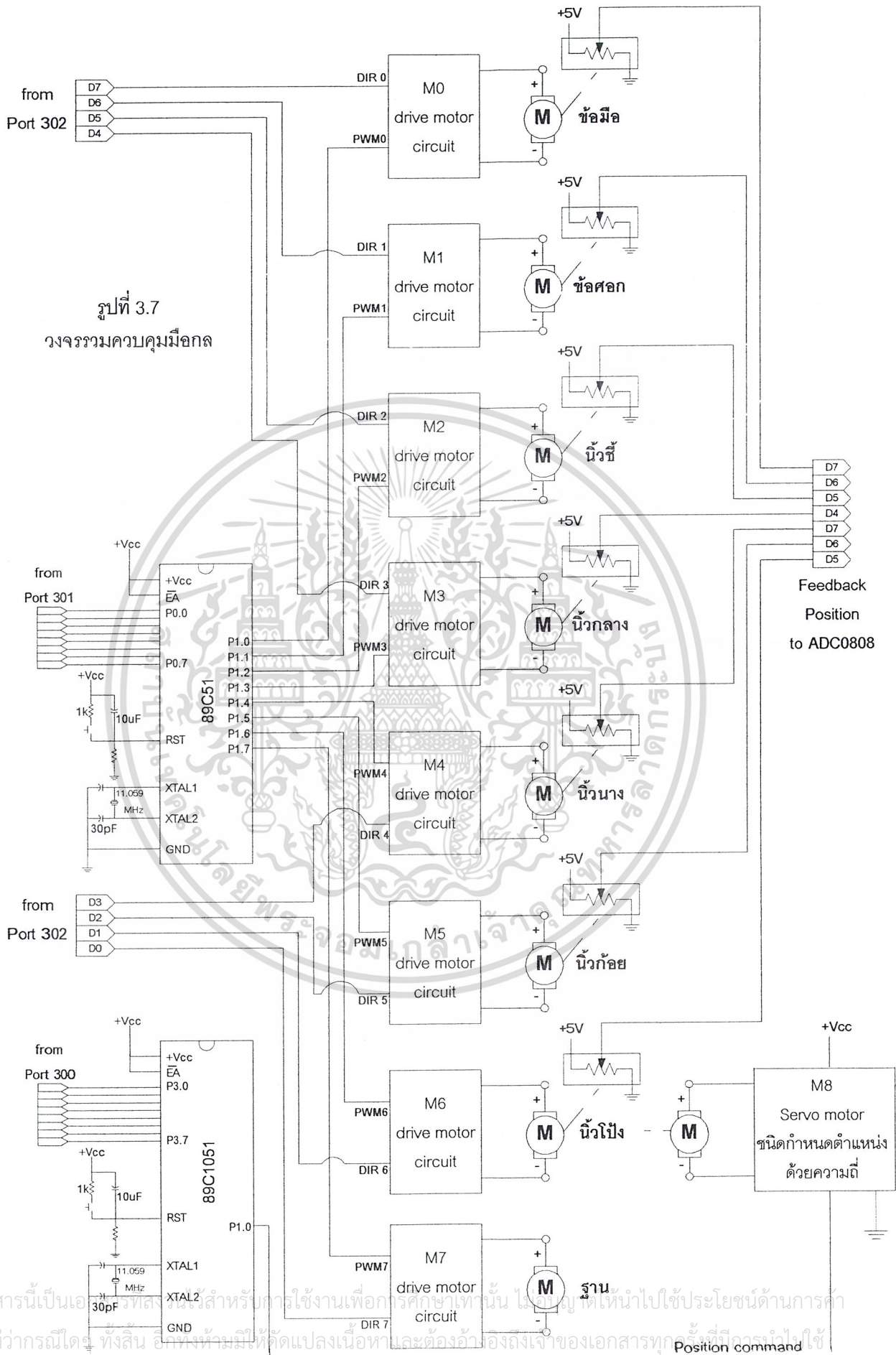
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 วงจรรวมควบคุมมือกล

ในการออกแบบมือกลครั้งนี้จะใช้มอเตอร์ทั้งหมด 9 ตัว โดยเป็นดีซีมอเตอร์ 8 ตัว และ เซอร์โวมอเตอร์ 1 ตัว โดยดีซีมอเตอร์ทั้ง 8 ตัวนี้จะใช้ในภาควงควบคุมการเคลื่อนที่ของชิ้นส่วนมือกล คือ ข้อมือ ข้อศอก นิ้วชี้ นิ้วกลาง นิ้วนาง นิ้วก้อย นิ้วหัวแม่มือ และฐาน ส่วนเซอร์โวมอเตอร์ใช้ขับเคลื่อนให้นิ้วหัวแม่มือสามารถพับเข้าหาฝ่ามือได้ โดยเราจะควบคุมดีซีมอเตอร์ผ่านวงจรขับมอเตอร์แบบ full bridge ดังรูปที่ 3.6 โดยเราจะส่งค่าทิศทางว่าจะให้มอเตอร์หมุนทางใดจาก port302 ของคอมพิวเตอร์และพัลส์จากไมโครคอนโทรลเลอร์ 89C51 ซึ่งไมโครคอนโทรลเลอร์ก็จะรับข้อมูลจาก port301 ของคอมพิวเตอร์ ซึ่งค่าที่รับมานี้จะเป็นตัวกำหนดว่าจะสร้างพัลส์ด้วยตัวตีไซเคิลเท่าไร และส่งไปยังวงจรขับมอเตอร์ของชุดใด

ส่วนไมโครคอนโทรลเลอร์ 89C1051 จะรับค่า port300 ของคอมพิวเตอร์ เพื่อนำมาสร้างพัลส์ที่สามารถเปลี่ยนความถี่ได้ให้แก่เซอร์โวมอเตอร์ในการพับส่วนของนิ้วโป้งเข้าหาฝ่ามือ ซึ่งตรงส่วนนี้เราไม่ต้องมี feedback เพราะเซอร์โวมอเตอร์กำหนดตำแหน่งด้วยความถี่ ซึ่งเป็นตำแหน่งที่แน่นอน ดังนั้นเราจึงสามารถควบคุมตำแหน่งของการเคลื่อนที่ได้โดยเปลี่ยนความถี่ที่ป้อนเข้าขาสัญญาณควบคุมของเซอร์โวมอเตอร์ ส่วนชิ้นส่วนอื่นๆเราต้องมี feedback เพื่อใช้ในการควบคุม โดยเราเลือกใช้ความต้านทางแบบปรับค่าได้ตามแนวเส้นตรงในการ feedback ตำแหน่งการเคลื่อนที่ส่งไปยัง ADC0808 ของการ์ดอินเทอร์เฟส เพื่อนำไปประมวลผลและควบคุมตำแหน่งของมือกลใน mode อัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7
วงจรรวมควบคุมมอเตอร์

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนสำหรับใช้ในการเรียนการสอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีวางนั้งไปใช้

Position command

บทที่ 4

โครงสร้างและการออกแบบทางกล

ในการออกแบบ Robotic Hand แบ่งออกเป็น 4 ส่วน คือ

- 1.) ส่วนของนิ้ว
- 2.) ส่วนของฝ่ามือ
- 3.) ส่วนของแขน
- 4.) ส่วนของฐาน

ซึ่งในแต่ละส่วนมีรายละเอียดในการออกแบบที่สำคัญ ๆ ดังนี้

1.) การออกแบบโครงสร้างส่วนนิ้ว

การออกแบบโครงสร้างส่วนนิ้วจะใช้อลูมิเนียมขนาดเส้นผ่านศูนย์กลาง 1 นิ้ว มาทำการกลึงด้วยเครื่อง CNC เป็นรูปนิ้ว โดยจะมีขนาดแตกต่างกันในแต่ละนิ้ว ซึ่งจะกลึงให้มีลักษณะและรูปร่างเหมือนนิ้วมือจริง แบ่งเป็นข้อ ๆ รวมทั้งสิ้น 14 ข้อ ซึ่งแบ่งเป็น นิ้วหัวแม่มือ นิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย จะมีนิ้วละ 3 ข้อ ส่วนนิ้วหัวแม่มือ มี 2 ข้อ หลังจากกลึงด้วยเครื่อง CNC แล้ว จะนำข้อของแต่ละนิ้วไปกัด เพื่อทำร่องสำหรับใส่สปริง และสลิง สำหรับใช้ดึงนิ้ว และดันนิ้ว หลังจากกัดแล้วจะทำการตกแต่งอีกครั้งหนึ่ง ข้อต่อแต่ละข้อจะมีสลักข้อต่อ แล้วนำมาประกอบกันขึ้นเป็นส่วนของนิ้ว และนำไปต่อเข้ากับส่วนของมือ นิ้วแต่ละนิ้วจะถูกดึงให้งอตัว และคลายตัวโดยใช้ซีมอเตอร์ในการดึงนิ้วนิ้วละ 1 ตัว รวมทั้งสิ้น 5 ตัว นิ้วมือจะมีลักษณะดังรูปที่ 4.1, 4.2 และ 4.3

2.) การออกแบบในส่วนของมือ

การออกแบบในส่วนนี้จะแบ่งเป็น 2 ส่วน คือ

- ฝ่ามือ ที่ประกอบด้วย ด้ามนิ้ว มี 3 ข้อ ทั้ง 4 นิ้ว ซึ่งโครงสร้างในส่วนนี้ จะใช้อลูมิเนียมเหลี่ยมขนาด $\frac{1}{4}$ นิ้ว มาขึ้นรูปและเชื่อมเป็นรูปฝ่ามือ ซึ่งในส่วนนี้จะประกอบด้วยข้อต่อของนิ้วทั้ง 4 นิ้วที่มีนิ้วละ 3 ข้อ ที่สำหรับยึดสปริงและมอเตอร์สำหรับขับเคลื่อนของฝ่ามือ ส่วนของนิ้วหัวแม่มือ

- ฝ่ามือที่เป็นส่วนประกอบของนิ้วหัวแม่มือ โครงสร้างในส่วนนี้จะใช้อลูมิเนียมแผ่นหนา 2 มิลลิเมตร นำมาพับขึ้นรูปเป็นรูปฝ่ามือ ในส่วนของนิ้วหัวแม่มือจะมีการติดตั้ง

เอกสารนี้เป็นเอกสารที่ใช้ในการดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
ไม่ว่าการนี้ได้นำรูปที่ 4.5

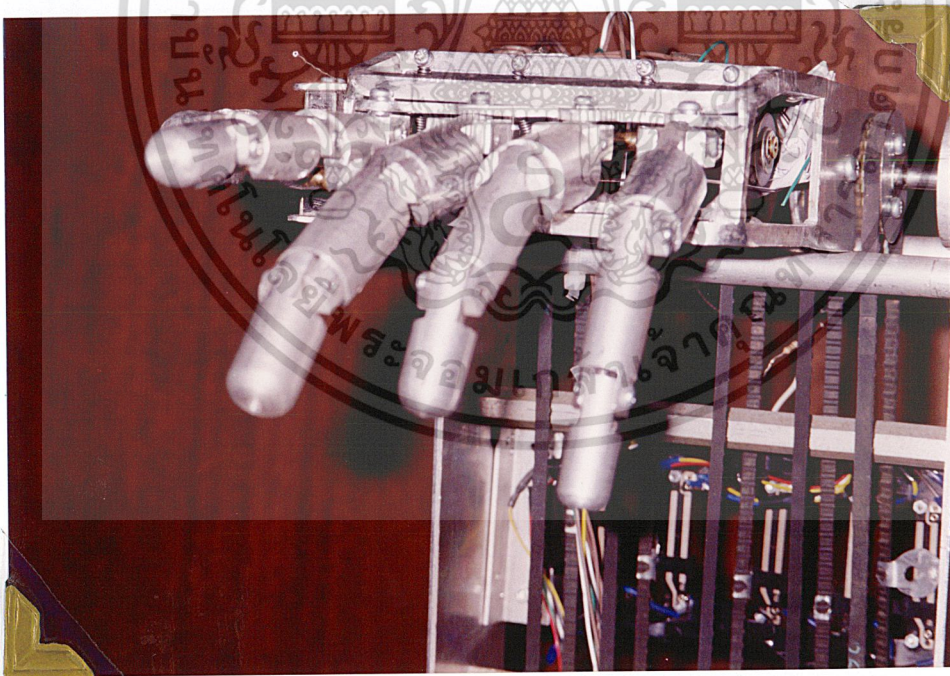
3.) การออกแบบโครงสร้างในส่วนของแขน

การออกแบบในนี้จะใช้อลูมิเนียมแผ่นขนาด 3 มิลลิเมตร นำมาตัดทำเป็นส่วนของแขน ซึ่งจะมีลักษณะรูปร่างดังรูปที่ 4.6 โดยในส่วนนี้จะมีการติดตั้งเฟืองและสายพานในการส่งกำลังจากมอเตอร์ไปดึงและคลายนิ้วให้อัตโนมัติ

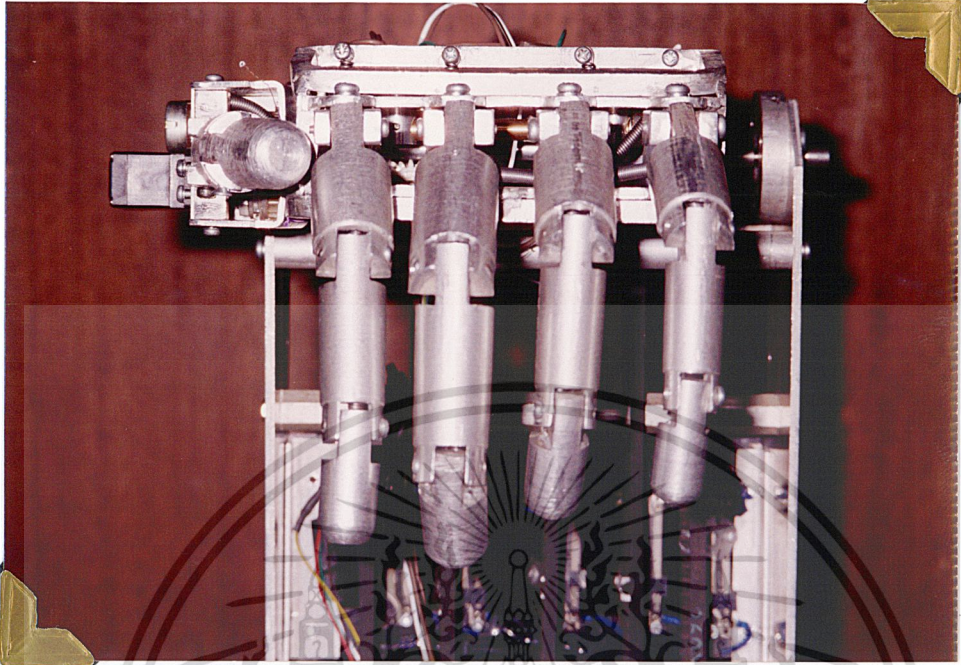
4.) การออกแบบในส่วนของฐาน

ส่วนฐานได้มีการออกแบบโครงสร้างดังรูป 4.8 มีโครงสร้างเป็นเหล็กซึ่งมีน้ำหนักมากพอสมควร เพื่อให้สามารถรองรับน้ำหนักของแขนและนิ้ว ตลอดจนแรงเหวี่ยงได้ มีการติดตั้งเฟืองและลูกปืนแบร์ริง ที่ส่วนของฐานเพื่อให้การหมุนของแขนมีความอิสระยิ่งขึ้น ซึ่งเป็นการลดแรงบิดของดีซีมอเตอร์ตัวที่อยู่ตำแหน่งฐาน

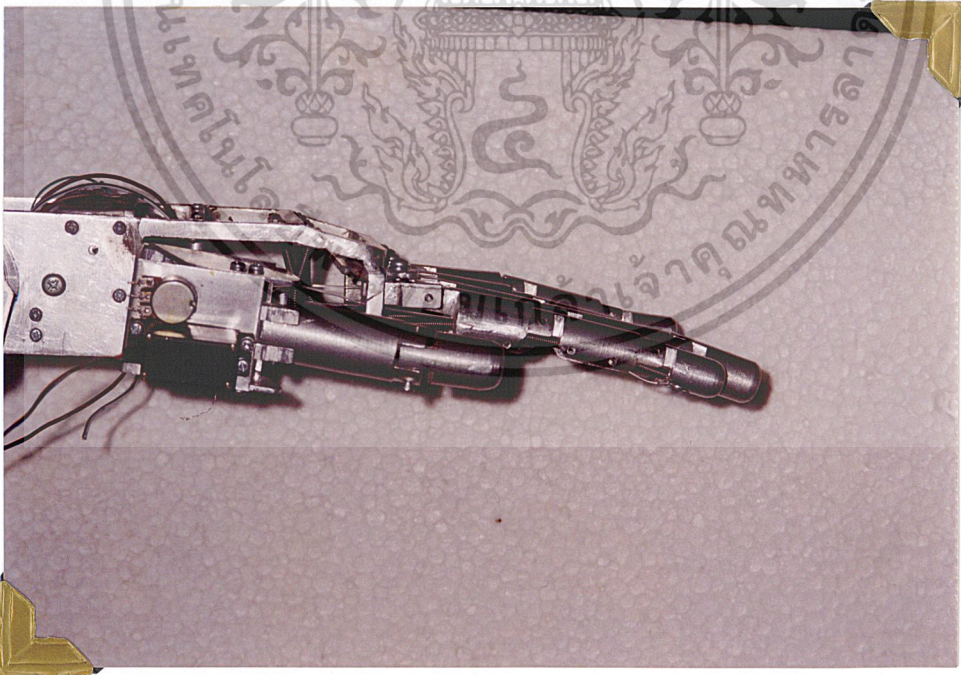
ส่วนฐานของแขนซึ่งอยู่ถัดจากฐานส่วนพื้น มีการติดตั้งมอเตอร์ที่ใช้ในการขยับนิ้ว ข้อมือ และข้อต่อของแขนที่ติดกับฐาน โครงสร้างส่วนนี้ทำจากอลูมิเนียมแผ่นหนา 3 มิลลิเมตร ในส่วนนี้จะมีมอเตอร์ติดตั้งอยู่ทั้งหมด 7 ตัวซึ่งเป็นมอเตอร์ขนาดต่าง ๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

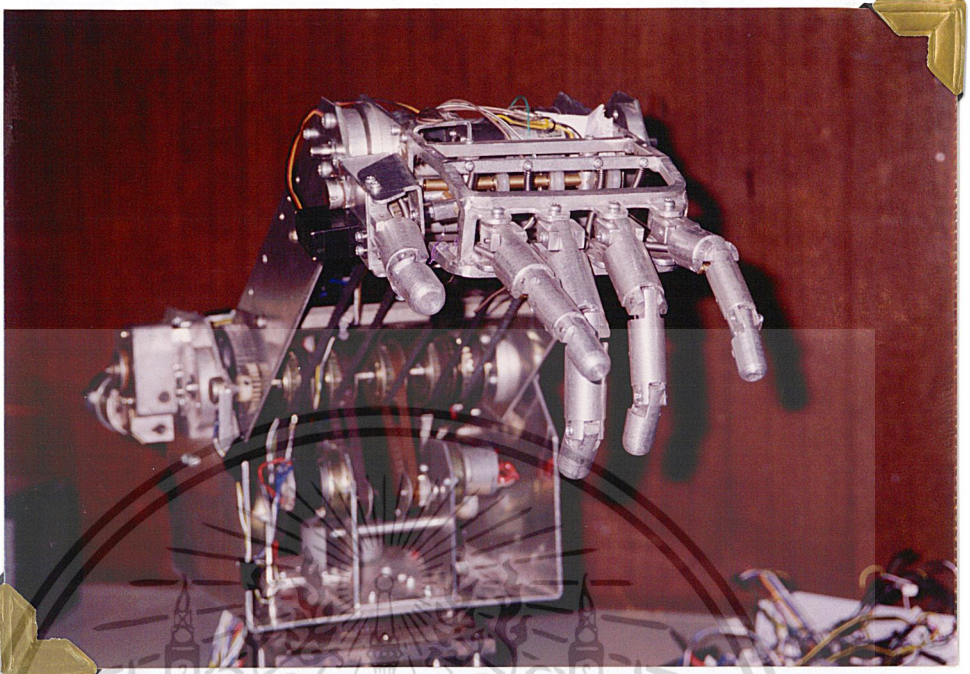


รูปที่ 4.2 ลักษณะและโครงสร้างของนิ้วมือด้านหน้า (Front view)

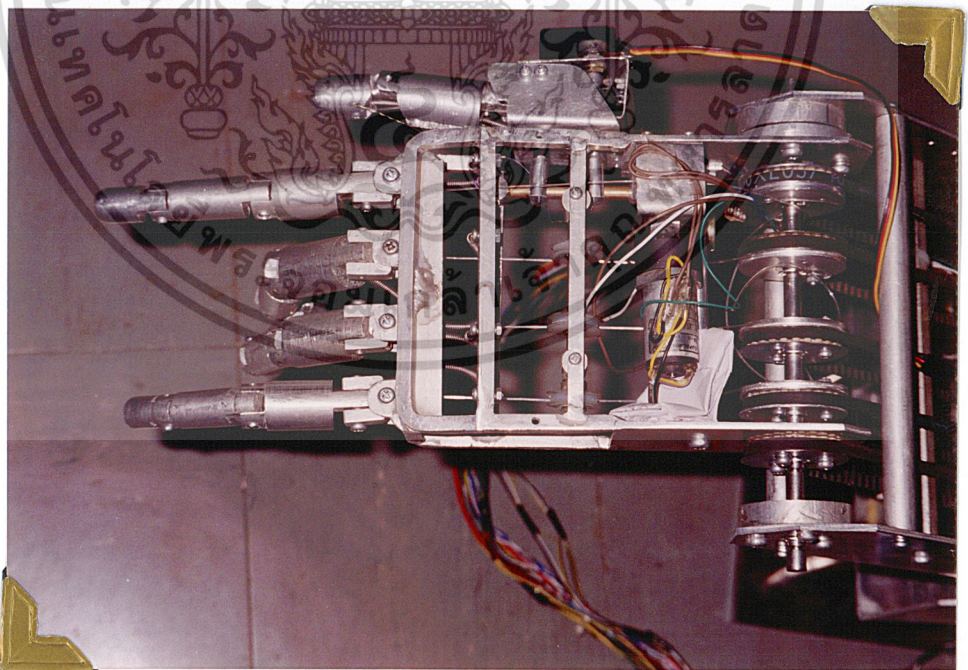


รูปที่ 4.3 ลักษณะและโครงสร้างของนิ้วมือด้านข้าง (Side view)

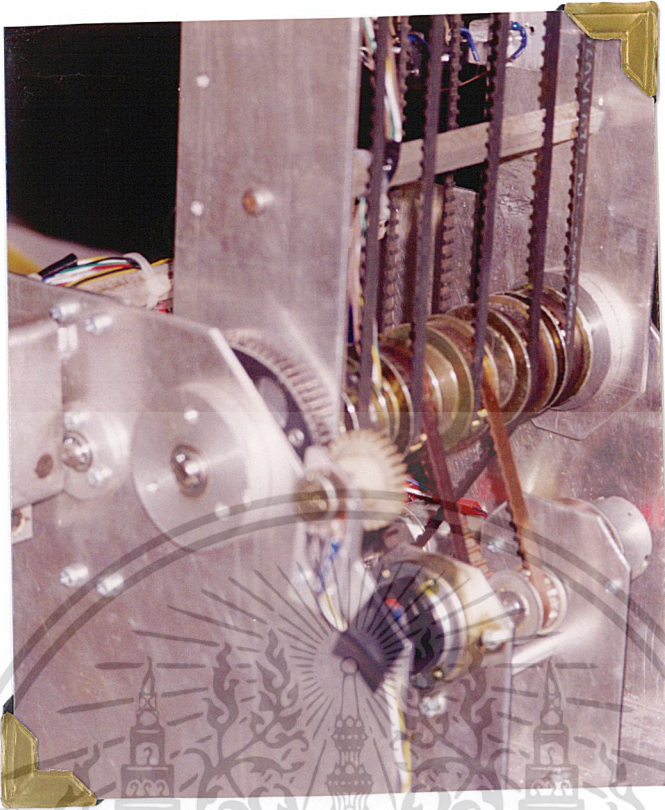
เอกสารนี้เป็นเอกสารที่ 4.3 ลักษณะและโครงสร้างของนิ้วมือด้านข้าง (Side view) ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



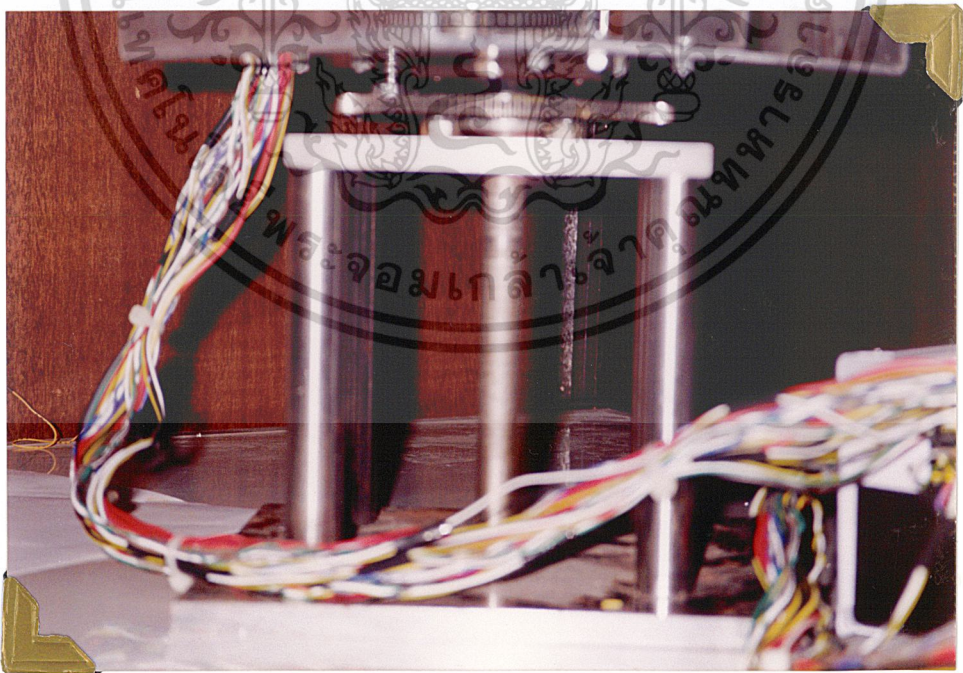
รูปที่ 4.4 ลักษณะของการเคลื่อนที่ของนิ้วมือ



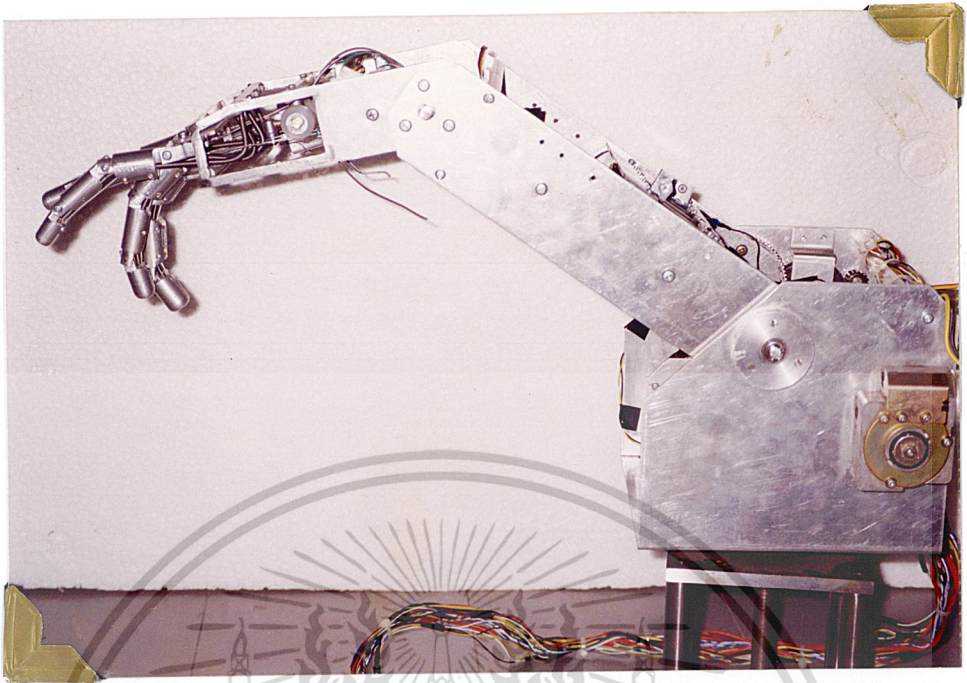
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.5 ลักษณะและโครงสร้างของมือ นุญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 ลักษณะและโครงสร้างของแขน

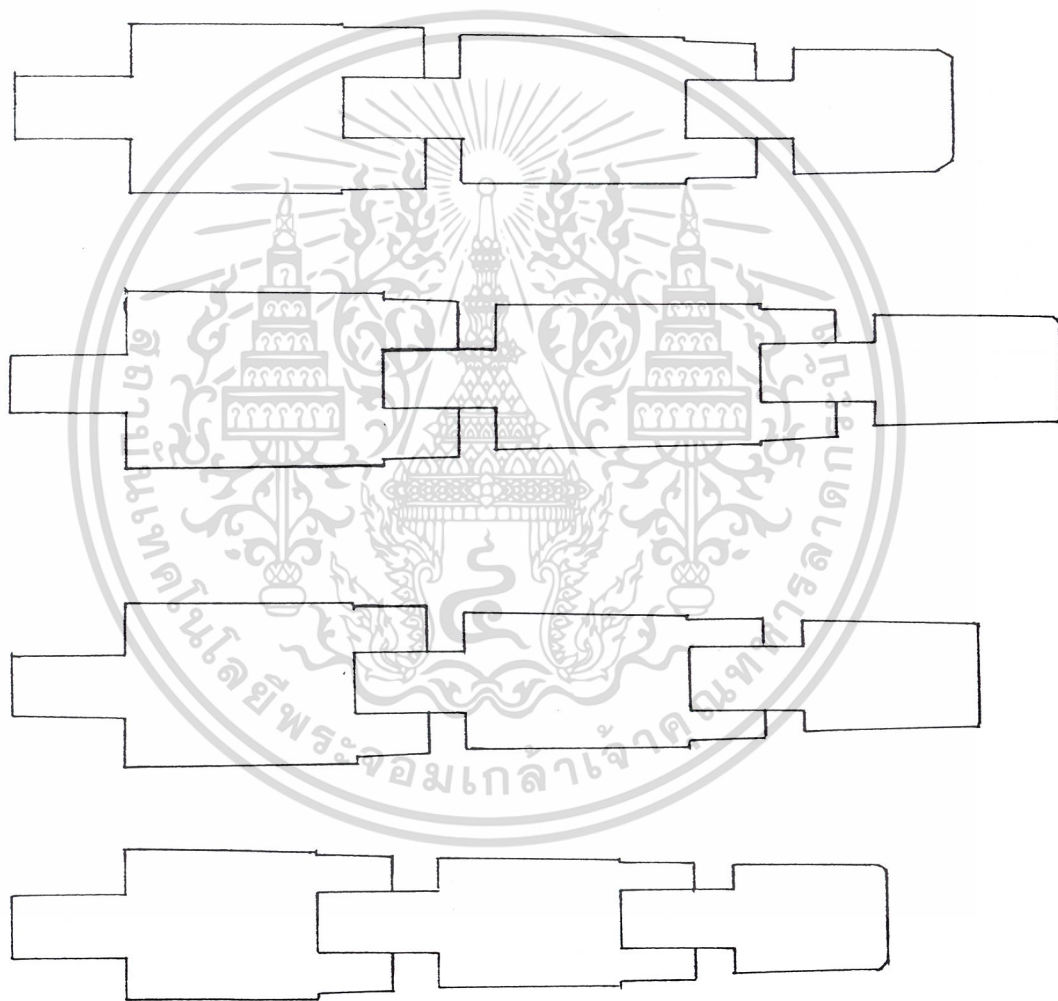


เอกสารนี้เป็นเอกสารที่สงวนรูปที่ 4.7 ลักษณะและโครงสร้างของส่วนฐาน ภาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



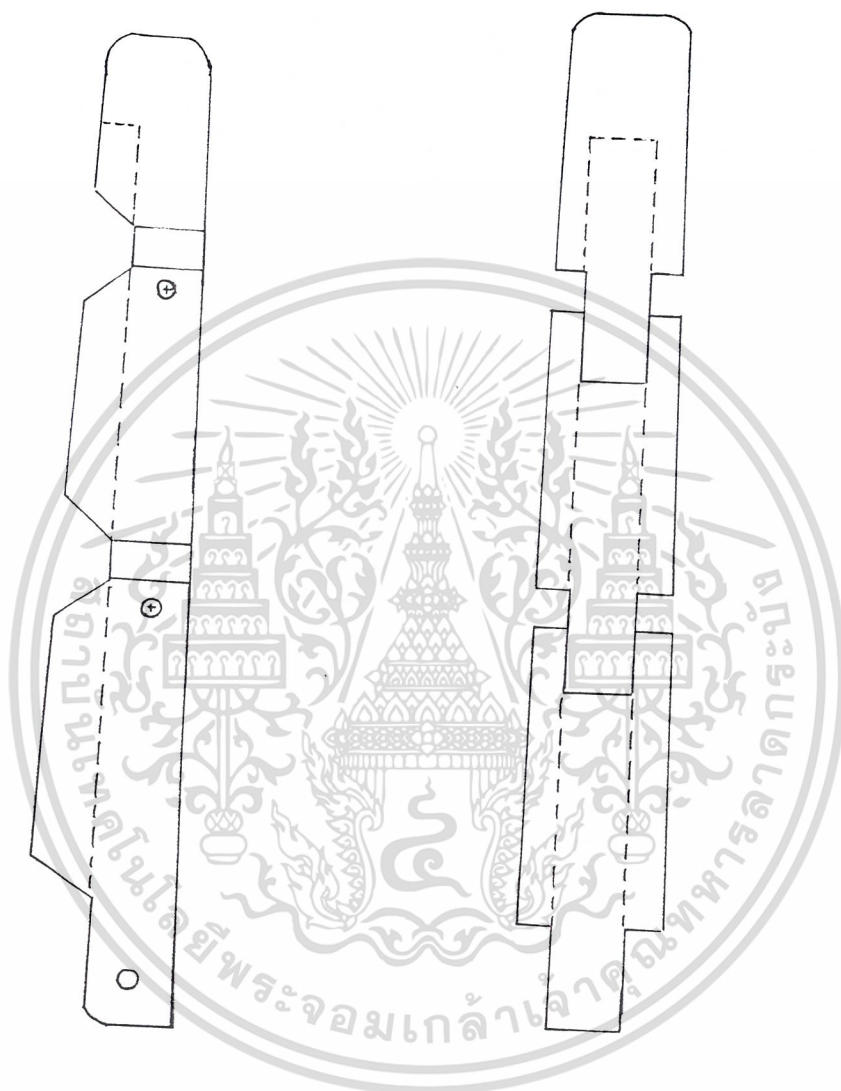
รูปที่ 4.8 ลักษณะและโครงสร้างโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



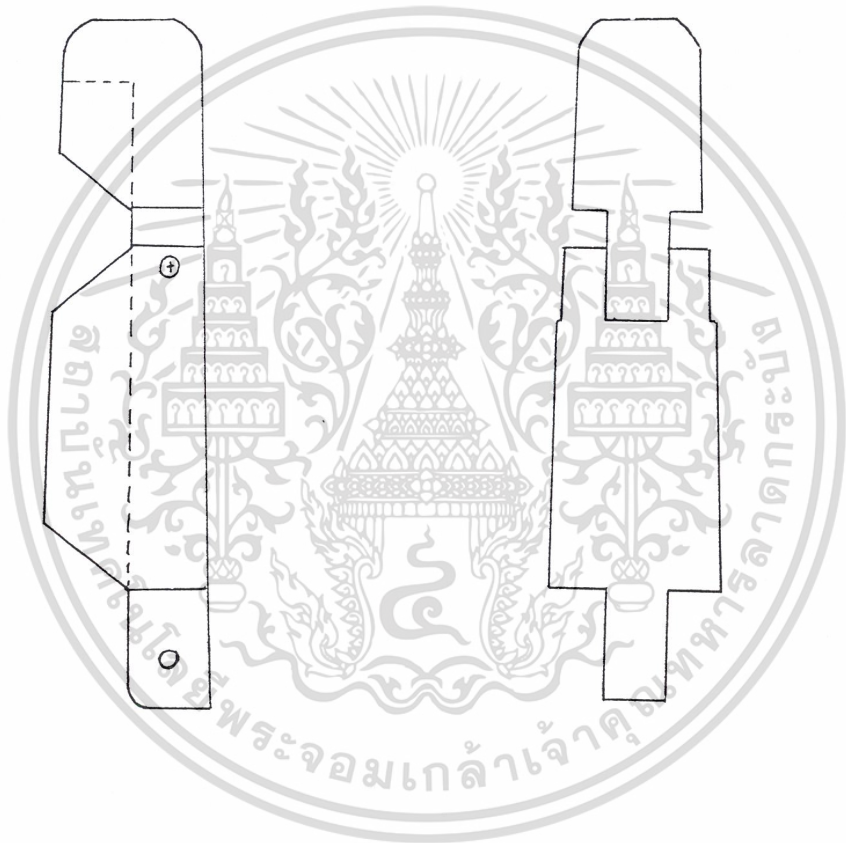
รูปที่ 4.9 แบบโครงสร้างของนั้วซี่, นั้วกลาง, นั้วนางและนั้วก้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



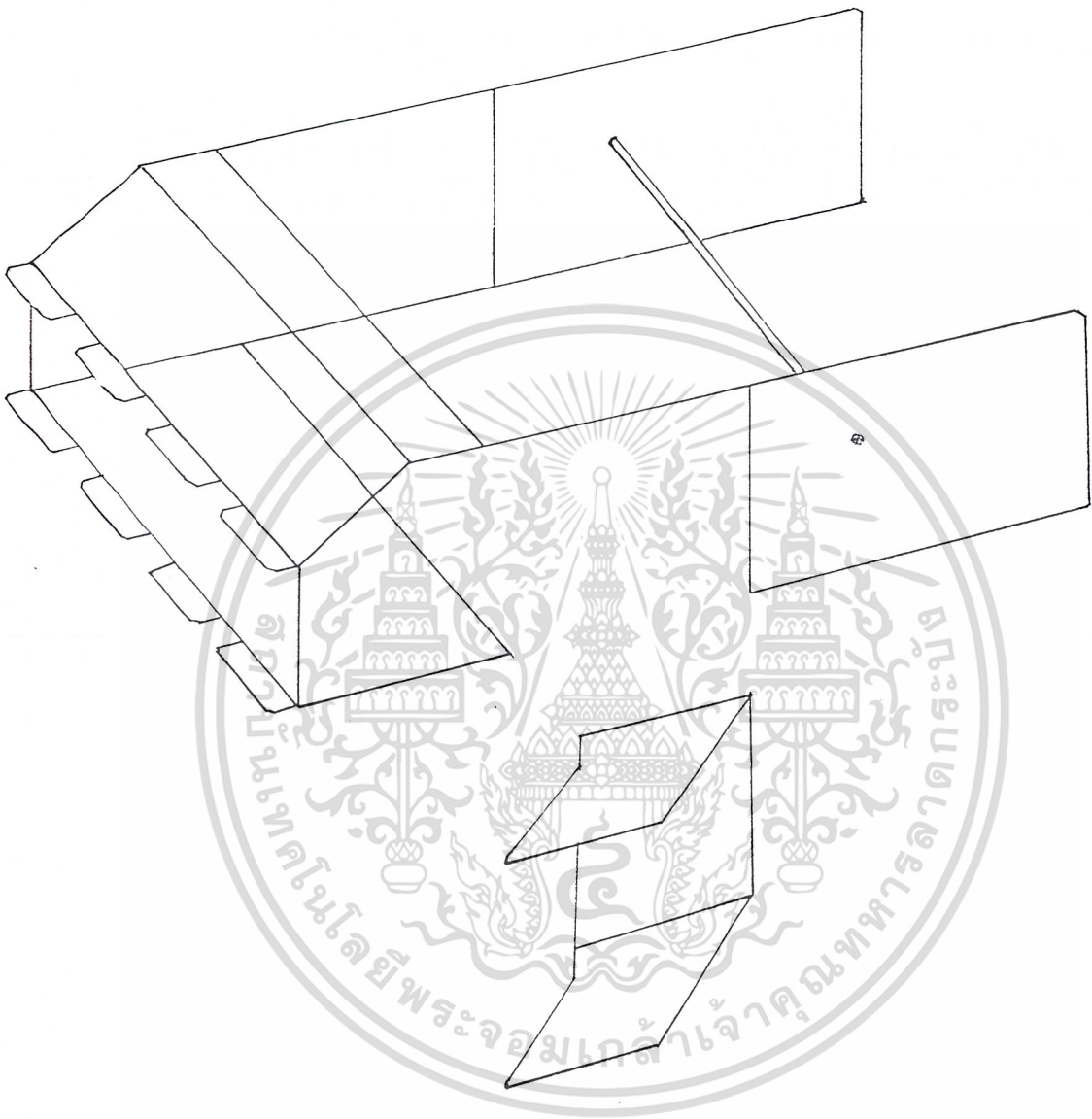
รูปที่ 4.10 แบบโครงสร้างด้านข้างและด้านบนของนี้้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



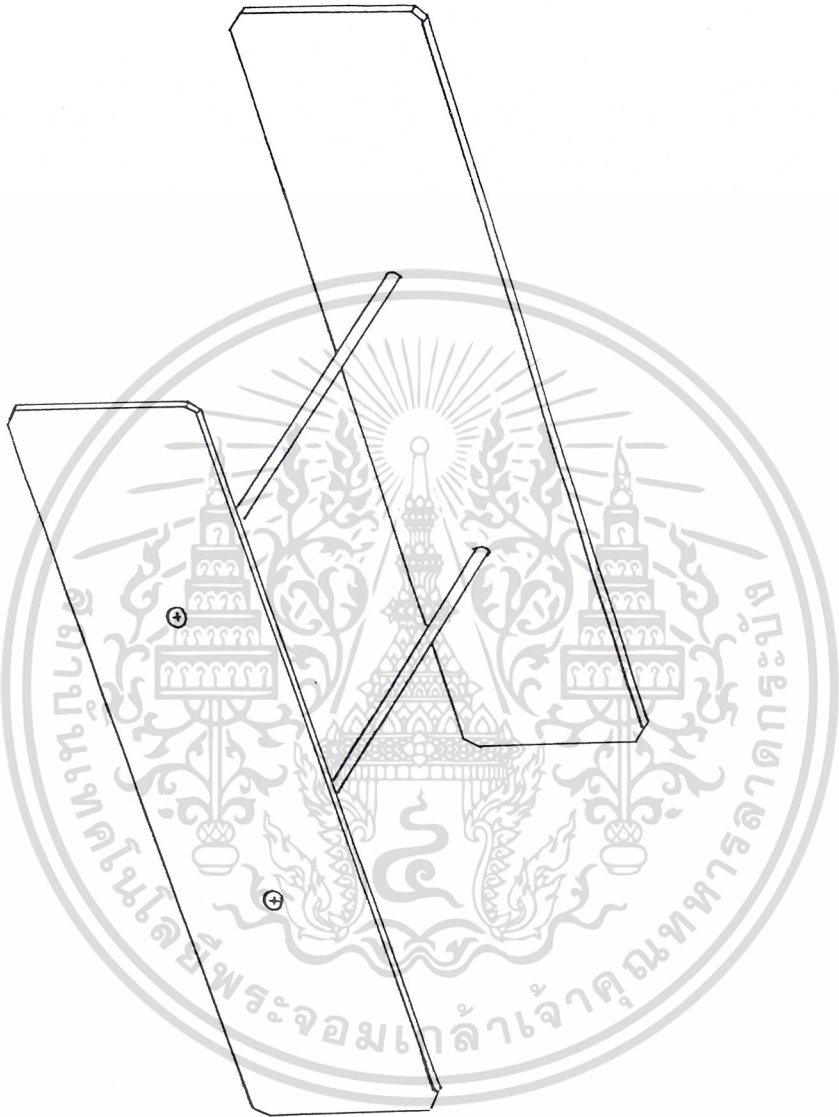
รูปที่ 4.11 แบบโครงสร้างด้านข้างและด้านบนของนิ้วโป้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



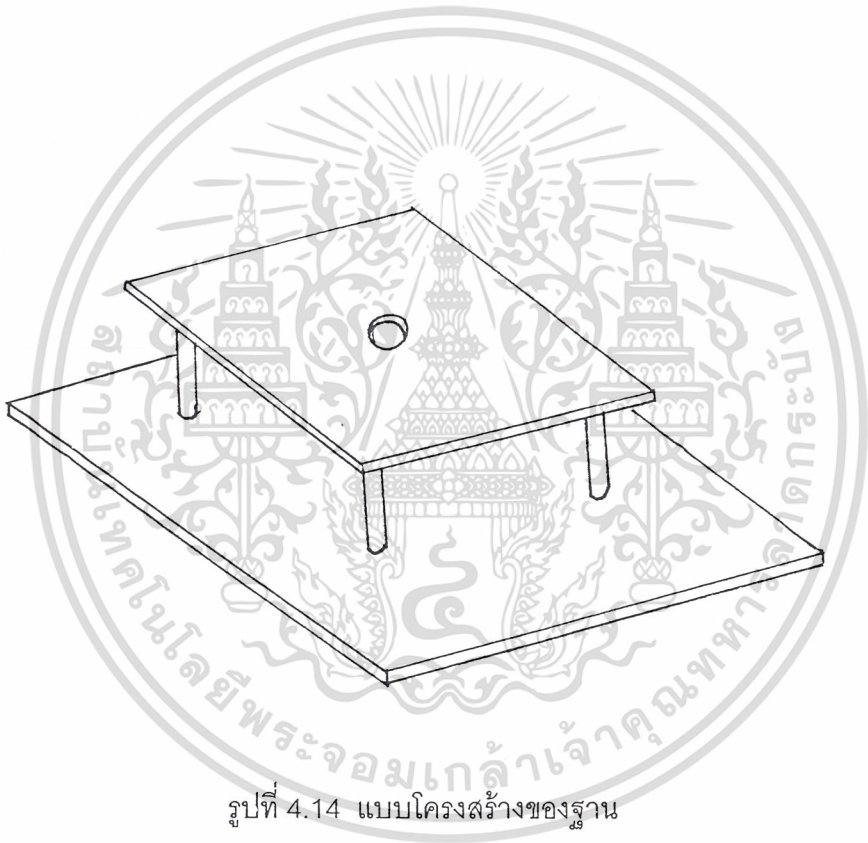
รูปที่ 4.12 แบบโครงสร้างของฝามือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แบบโครงสร้างของแ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 แบบโครงสร้างของฐาน

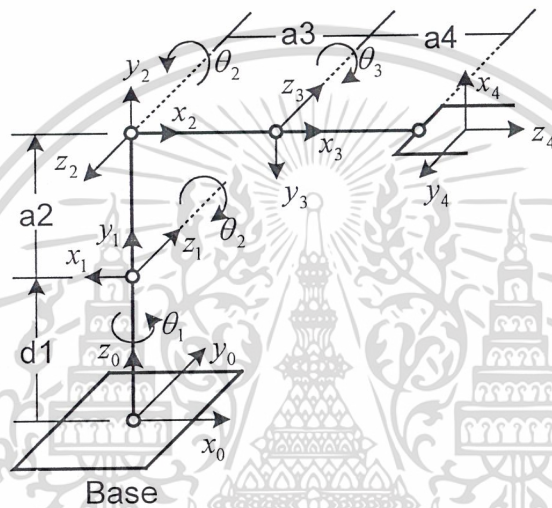
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิเคราะห์ผลการดำเนินงาน

5.1 ผลการดำเนินงาน

สมการแขน (The Arm Equation) ของหุ่นยนต์ที่สร้างขึ้น



รูปที่ 5.1 พิกัด link ของหุ่นยนต์ที่สร้างขึ้น

จากไดอะแกรมโครงพิกัดในรูปที่ 5.1 ได้ Kinematics parameters ดังตารางที่ 5.1

Axis	θ	d	a	α	Home
1	θ_1	d_1	0	$\pi/2$	π
2	θ_2	0	a_2	π	π
3	θ_3	0	a_3	π	0
4	θ_4	0	a_4	$-\pi/2$	$-\pi/2$

ตารางที่ 5.1 Kinematics parameters ของหุ่นยนต์ในรูปที่ 5.1

และได้เมตริกซ์แขนดังนี้
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งยังมีโทษตามกฎหมายที่เกี่ยวข้องหากมีการนำไปใช้

$$T_{base}^{tool}(\theta) = T_0^1(\theta_1)T_1^2(\theta_2)T_2^3(\theta_3)T_3^4(\theta_4) = T_0^4(\theta)$$

$$T_0^4 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_2 & S_2 & 0 & a_2 C_2 \\ S_2 & -C_2 & 0 & a_2 S_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_3 & S_3 & 0 & a_3 C_3 \\ S_3 & -C_3 & 0 & a_3 S_3 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_4 & 0 & -S_4 & a_4 C_4 \\ S_4 & 0 & C_4 & a_4 S_4 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C_1 C_2 & C_1 S_2 & S_1 & a_2 C_1 C_2 \\ S_1 C_2 & S_1 S_2 & C_1 & a_2 S_1 C_2 \\ S_2 & -C_2 & 0 & d_1 + a_2 S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_3 & S_3 & 0 & a_3 C_3 \\ S_3 & -C_3 & 0 & a_3 S_3 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_4 & 0 & -S_4 & a_4 C_4 \\ S_4 & 0 & C_4 & a_4 S_4 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C_1 C_{2-3} & C_1 S_{2-3} & S_1 & a_3 C_1 C_{2-3} + a_2 C_1 C_2 \\ S_1 C_{2-3} & S_1 S_{2-3} & -C_1 & a_3 S_1 C_{2-3} + a_2 S_1 C_2 \\ S_{2-3} & C_{2-3} & 0 & a_3 S_{2-3} + a_2 S_2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_4 & 0 & -S_4 & a_4 C_4 \\ S_4 & 0 & C_4 & a_4 S_4 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C_1 C_{2-3-4} & -S_1 & C_1 S_{2-3-4} & a_4 C_1 C_{2-3-4} + a_3 C_1 C_{2-3} + a_2 C_1 C_2 \\ S_1 C_{2-3-4} & C_1 & S_1 S_{2-3-4} & a_4 S_1 C_{2-3-4} + a_3 S_1 C_{2-3} + a_2 S_1 C_2 \\ S_{2-3-4} & 0 & C_{2-3-4} & a_4 S_{2-3-4} + a_3 S_{2-3} + a_2 S_2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

เมื่อ $C_{i+j} = \cos(\theta_i + \theta_j)$ และ $S_{i+j} = \sin(\theta_i + \theta_j)$

5.2 สรุปผลการดำเนินงาน

จากการศึกษา ทดลองและจัดสร้าง โครงงานนี้แบ่งได้เป็น 3 ส่วน คือ ส่วนโครงสร้างและการออกแบบทางกล ส่วนโครงสร้างและการออกแบบทางวงจรีเลคทรอนิกส์ และส่วนโปรแกรมควบคุมการทำงาน, รูปแบบการเคลื่อนที่ของมือกล

ส่วนโครงสร้างทางกล ผู้จัดทำได้รับความอนุเคราะห์จากภาควิชาวิศวกรรมเครื่องกล ในการใช้เครื่องมือและอุปกรณ์เพื่อจัดสร้างส่วนประกอบทางกลต่าง ๆ ดังนี้

1. นี๊วมือ ใช้ CNC กลึงเป็นขั้วนี๊วมือแต่ละขั้วแล้วนำมากัดร่องด้วยเครื่องกัด เจาะรูตีฟเกลียวเพื่อใช้ในการยึดข้อต่อเข้าด้วยกัน ปัญหาที่ประสบคือ ขาดความชำนาญในการใช้เครื่องมือต่างๆ เช่น CNC, เครื่องกัด, เครื่องกลึง ทำให้ทำงานล่าช้า และปัญหาอีกประการคือ ไม่สามารถหาสปริงที่มีขนาดเหมาะสมมาใช้กับนี๊วได้

เอกสารนี้เื่อ 2. ฝ่ามือ, ที่แขนและฐาน มีโครงสร้างที่ไม่ซับซ้อน การทำงานในส่วนนี้จึงไม่ประสบปัญหาการคำนวณมากนักใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ส่วนประกอบอื่นๆ เช่น เฟือง, สายพาน มักจะหาขนาดที่ต้องการไม่ได้และหาซื้อได้ยาก

ส่วนโครงสร้างและการออกแบบทางวงจรรีเลย์ทรอนิกส์ ประกอบไปด้วย

1. มอเตอร์กระแสตรง ซึ่งเป็นปัญหาสำคัญเนื่องจากไม่สามารถหามอเตอร์แบบที่ต้องการได้ คือ มอเตอร์ที่ติดเอ็นโคดเดอร์ในตัว และปัญหาอีกประการหนึ่งคือ หาซื้อมอเตอร์ได้ยากเนื่องจากทางกลุ่มได้ใช้มอเตอร์ของเก่าเพื่อลดงบประมาณและประหยัด (ถ้าเป็นมอเตอร์ใหม่จะมีราคาสูงมาก)

2. เซ็นเซอร์และเอ็นโคดเดอร์ต่าง ๆ เนื่องจากอุปกรณ์บางอย่างมีราคาสูงมาก เช่น Pressure sensor, Potentiometer ไม่สามารถซื้อมาได้ จึงใช้อุปกรณ์ที่มีราคาถูกแทน เช่น R ปรับค่าได้ เป็นต้น

3. วงจรควบคุมต่าง ๆ มักประสบปัญหาการสั้ดวงจรเนื่องจากการต่อวงจรที่ผิดพลาด ทำให้การทำงานล่าช้า

5.3 แนวทางในการพัฒนา

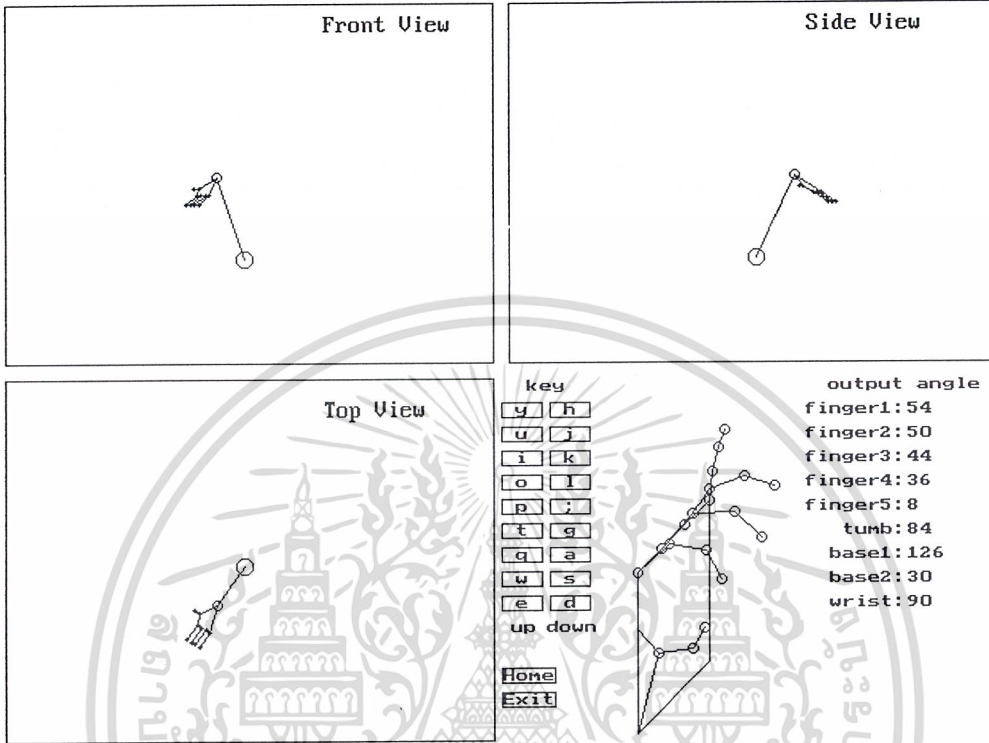
สามารถเพิ่มเติมส่วนประกอบต่าง ๆ ได้อีกเพื่อขยายความสามารถของแขนกลนี้ เช่น ติดตั้งอุปกรณ์ต่าง ๆ ไม่ว่าจะเป็น Pressure sensor, Encoder, Potentiometer เป็นต้น รวมทั้งเปลี่ยนแปลง, พัฒนาในส่วนวงจรรวมให้มีประสิทธิภาพมากขึ้น การพัฒนาในส่วนโปรแกรมควบคุมการทำงานก็เป็นสิ่งสำคัญอีกอย่างหนึ่งที่ควรได้รับการพัฒนามากขึ้นด้วย เพื่อให้มีออกมามีขีดความสามารถมากขึ้น เช่น จับและเคลื่อนย้ายสิ่งของ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

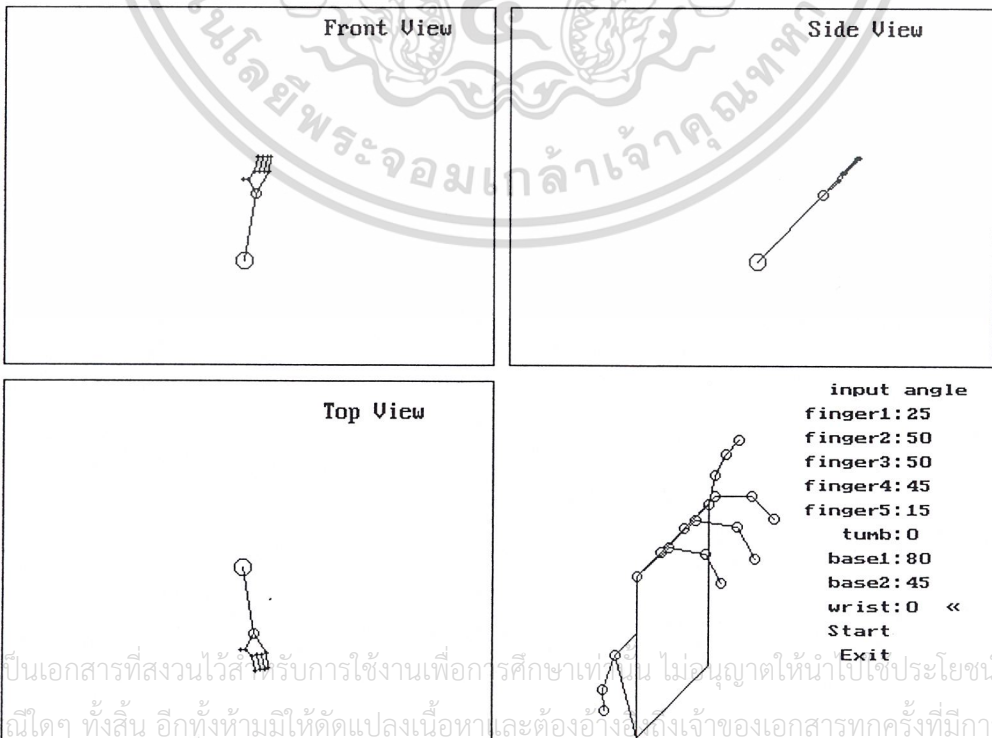


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
แนวความคิดในการออกแบบโปรแกรม



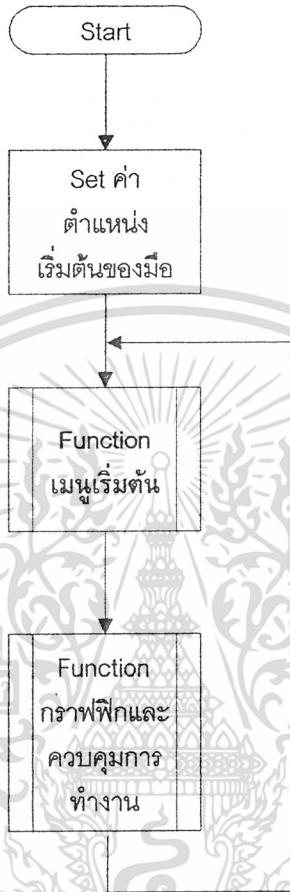
หน้าจอหลักของโปรแกรมโหมด Manual



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าจอหลักของโปรแกรมโหมด Automatic

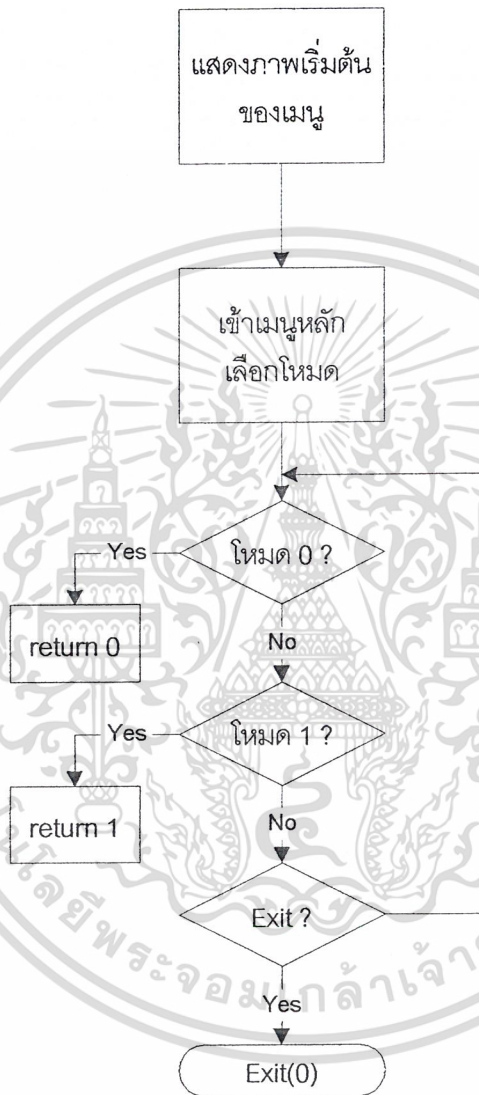
Main program



Flowchart แสดงส่วนของ Main program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

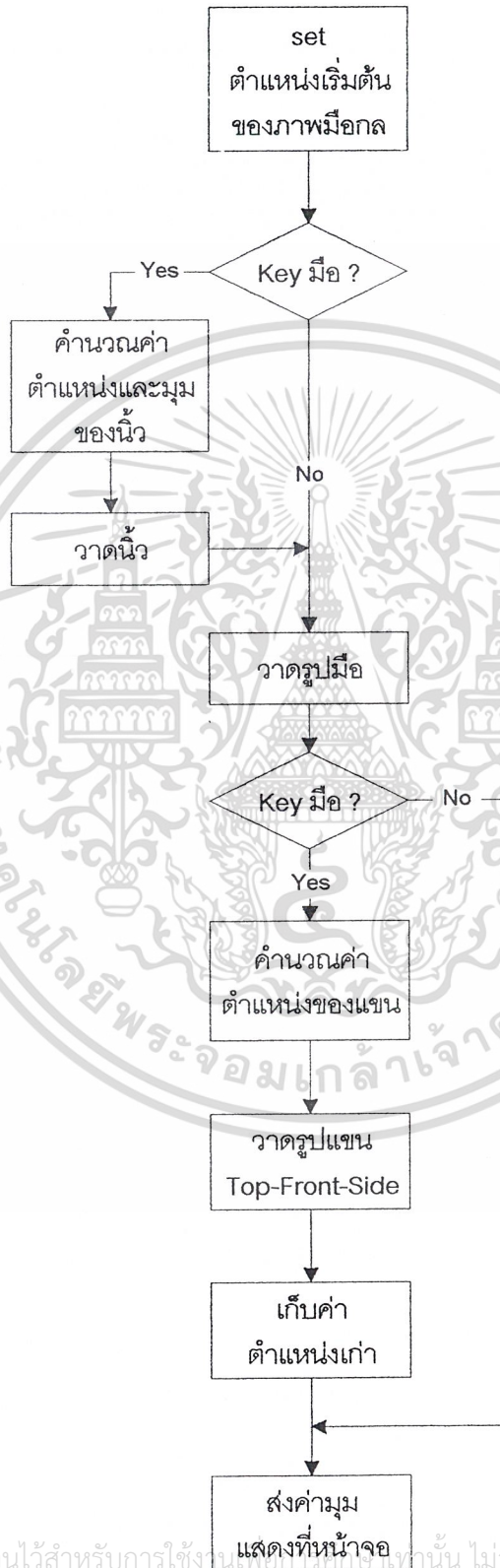
Function menu



Flowchart แสดงเมนูเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

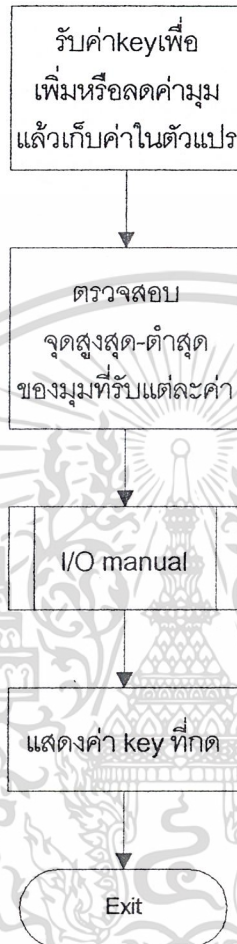
Function แสดงกราฟฟิก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flowchart แสดงกราฟฟิกของมือกล

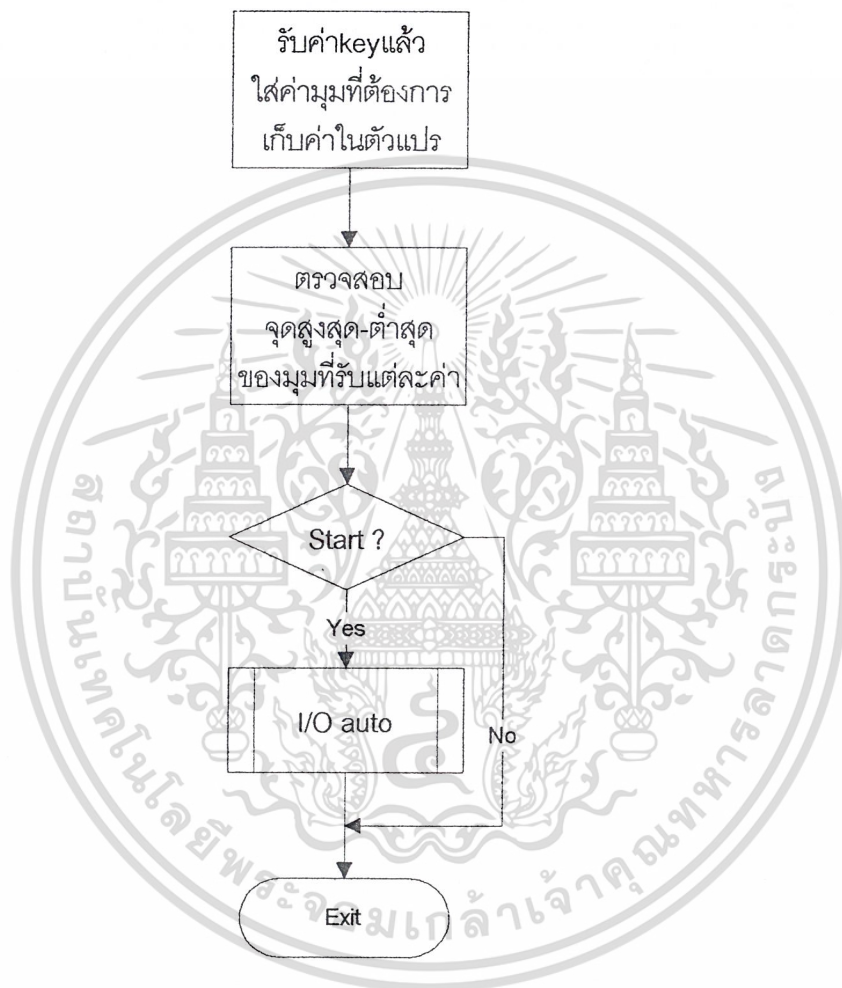
Manual mode



Flowchart แสดงส่วน
ของ Manual program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Automatic mode



Flowchart แสดงส่วน
ของ Automatic program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

โปรแกรมควบคุมการทำงาน

robot_h.cpp

```
#include <CONIO.H>
#include <MATH.H>
#include <GRAPHICS.H>
#include <STRING.H>
#include <IOSTREAM.H>
#include <IOMANIP.H>
#include <STDIO.H>
#include <STDLIB.H>
#include <DOS.H>

struct position{
    int x[5][3],y[5][3],old_x[5][3],old_y[5][3];
};

struct position2{
    int wristX,wristY,wristZ,old_wristX,old_wristY,old_wristZ,
        fingerX[5],fingerY[5],fingerZ[5],old_fingerX[5],old_fingerY[5],
        old_fingerZ[5],end_fingerX[5],end_fingerY[5],end_fingerZ[5],
        old_end_finX[5],old_end_finY[5],old_end_finZ[5];
};

struct long_finger{
    int j1[5],j2[5],j3[5];
};

struct base_TFS{
    int Px1,Py1,Px2,Py2,Px3,Py3;
};
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
extern int menu();

```

void draw_all(int*,int*,struct position,int*,struct long_finger,int);
void draw_finger(int*,int*,int*,int*,int);
void draw_hand(int*,int*,struct position);
void out_windows();
check_angle(int,int*);
getfb(int);
void draw_hand_TFS(struct base_TFS,struct position2);
float ang(int);
void io_auto(int*,int,int,int,int);
void io_manual(int*,int,int,int,int,int);
void set_drt(int,int);
void set_pwm(int,int);

main()
{
    struct position pos;
    struct long_finger fin;
    int driver,gmode,mode,finger_num;
    int angle[5]={2,0,0,0,0};
    int middle_x=420,
        middle_y=350;
    int base_finx[5]={middle_x-15,middle_x-15,middle_x,middle_x+15,middle_x+30},
        base_finy[5]={middle_y+50,middle_y+15,middle_y,middle_y-15,middle_y-30};
    int num;
    fin.j1[0]=20,fin.j1[1]=27,fin.j1[2]=30,fin.j1[3]=27,fin.j1[4]=18;
    fin.j2[0]=23,fin.j2[1]=24,fin.j2[2]=27,fin.j2[3]=24,fin.j2[4]=15;
    fin.j3[0]=14,fin.j3[1]=21,fin.j3[2]=24,fin.j3[3]=21,fin.j3[4]=12;
    for(finger_num=0;finger_num<=4;finger_num++)
        for(num=0;num<=2;num++){
            pos.old_x[finger_num][num]=base_finx[finger_num];
            pos.old_y[finger_num][num]=base_finy[finger_num];
        }
    while(1){
        mode=menu();

```

```

        outputb(0x301,0x00); //set duty 0 every bit

```

```

        delay(1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outportb(0x301,0x10);
delay(1);
outportb(0x301,0x20);
delay(1);
outportb(0x301,0x30);
delay(1);
outportb(0x301,0x40);
delay(1);
outportb(0x301,0x50);
delay(1);
outportb(0x301,0x60);
delay(1);
outportb(0x301,0x70); //set duty 0 every bit
outportb(0x300,2);
driver=DETECT;
initgraph(&driver,&gmode,"c:\tc\bgi");
out_windows();
if((mode==0)||(mode==1))
    draw_all(base_finx,base_finy,pos,angle,fin,mode);
else {closegraph();exit(0);}
closegraph();
}
}

```

```

void draw_all(int *base_finx,int *base_finy,struct position pos,
int *angle,struct long_finger fin,int mode)
{

```

```

const U_ARRO=72,
D_ARRO=80,
BACK_SPACE=8,
ENTER='\r',
M_F1=260,M_F2=275,M_F3=290,M_F4=305,M_F5=320,M_T=335,
M_B1=350,M_B2=365,M_W=380,M_ST=395,M_EX=410,
ch1=0x00,ch2=0x20,ch3=0x40,ch4=0x60,ch5=0x80,ch6=0xA0,ch7=0xC0,
ch8=0xE0,ch9=0x00,ch10=0x04,ch11=0x08,ch12=0x0C,ch13=0x10,ch14=0x14,
ch15=0x18,ch16=0x1C,

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

forward=1,backward=0,
motor0=0,motor1=1,motor2=2,motor3=3,
motor4=4,motor5=5,motor6=6,motor7=7;

```

```

struct position2 pos2;
struct base_TFS pos3;
char tp_ang[]=" ",*endptr;
int exit;
int num,finger_num,Larm=60,Lfin;
int Lhand[]={13,20,20,20,20};
int angle_base1=0,angle_base2=0,angle_wrist=0;
int angle_t=0;
char temp_outtext[5];
int ff[]={-6,-3,0,3,6};
int menu_a=260;//
int loop=0;
long temp_angle;
int ox=600,oy=600;char och[]="□ □ □ □ ";
char key='m';
int first_in=0;

pos3.Px1=(getmaxx()/4), //
pos3.Py1=((getmaxy()/2)-70), // set initial
pos3.Px2=(getmaxx()-(getmaxx()/4)), // base position
pos3.Py2=pos3.Py1, // in top,front
pos3.Px3=pos3.Px1,pos3.Py3=(getmaxy()-(getmaxy()/4)); // side view
pos2.old_wristX=pos2.old_wristY=pos2.old_wristZ=6; //
for(num=0;num<=4;num++){ //
pos2.old_fingerX[num]=pos2.old_fingerY[num]=pos2.old_fingerZ[num]=6; //
pos2.old_end_finX[num]=pos2.old_end_finY[num]=pos2.old_end_finZ[num]=6;//
}

do{

```

เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น ยกเว้นหากมีเหตุเห็นสมควรขออนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

||(key=='t')||(key=='g')||(key=='m')||key==ENTER)
    {
//begin if
//***** calculate position of fingers *****
    pos.x[0][0]=base_finx[0]-fin.j1[0]*cos(ang(angle_t+47));
    pos.y[0][0]=base_finy[0]+fin.j1[0]*sin(ang(angle_t+47));
    pos.x[0][1]=pos.x[0][0]-fin.j2[0]*cos(ang(angle_t+47+angle[0]));
    pos.y[0][1]=pos.y[0][0]+fin.j2[0]*sin(ang(angle_t+47+angle[0]));
    pos.x[0][2]=pos.x[0][1]-fin.j3[0]*cos(ang(angle_t+47+ang(angle[0])*2);
    pos.y[0][2]=pos.y[0][1]+fin.j3[0]*sin(ang(angle_t+47+ang(angle[0])*2);
    for(finger_num=1;finger_num<=4;finger_num++)
    {
        pos.x[finger_num][0]=base_finx[finger_num]-fin.j1[finger_num]
            *sin(ang(-angle[finger_num]));
        pos.y[finger_num][0]=base_finy[finger_num]-fin.j1[finger_num]
            *cos(ang(-angle[finger_num]));
        pos.x[finger_num][1]=pos.x[finger_num][0]-fin.j2[finger_num]
            *sin(ang(-angle[finger_num])*2);
        pos.y[finger_num][1]=pos.y[finger_num][0]-fin.j2[finger_num]
            *cos(ang(-angle[finger_num])*2);
        pos.x[finger_num][2]=pos.x[finger_num][1]-fin.j3[finger_num]
            *sin(ang(-angle[finger_num])*3);
        pos.y[finger_num][2]=pos.y[finger_num][1]-fin.j3[finger_num]
            *cos(ang(-angle[finger_num])*3);
    }
//***** draw finger *****
    setcolor(BLACK);
    for(finger_num=0;finger_num<=4;finger_num++)
    {
        draw_finger(base_finx,base_finy,&pos.old_x[0][0],&pos.old_y[0][0],
            finger_num);

        for(num=0;num<=2;num++)
        {
            pos.old_x[finger_num][num]=pos.x[finger_num][num];
            pos.old_y[finger_num][num]=pos.y[finger_num][num];
        }
    }
}

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

setcolor(WHITE);

for(finger_num=0;finger_num<=4;finger_num++)

draw_finger(base_finx,base_finy,&pos.x[0][0],&pos.y[0][0],

finger_num);

//***** draw hand *****

} //end if

draw_hand(base_finx,base_finy,pos);

if((key=='q')||(key=='a')||(key=='w')||(key=='s')||(key=='e')

||(key=='d')||(key=='m')||(key==ENTER))

{//begin if

//***** calculate position of arm *****

pos2.wristX=Larm*cos(ang(angle_base1))*cos(ang(angle_base2+90));

pos2.wristY=Larm*sin(ang(angle_base1))*cos(ang(angle_base2+90));

pos2.wristZ=Larm*sin(ang(angle_base2+90));

for(num=0;num<=4;num++){

pos2.fingerX[num]=Lhand[num]*cos(ang(angle_base1))*cos(ang

(angle_base2+90+angle_wrist))+pos2.wristX-ff[num]*sin(ang

(angle_base1));

pos2.fingerY[num]=Lhand[num]*sin(ang(angle_base1))*cos(ang

(angle_base2+90+angle_wrist))+pos2.wristY+ff[num]*cos(ang

(angle_base1));

pos2.fingerZ[num]=Lhand[num]*sin(ang(angle_base2+90+angle_wrist))

+pos2.wristZ;

}

for(num=0;num<=4;num++){

if(num==0){Lfin=0;ff[0]=-10;}

else{Lfin=13;ff[0]=-6;};

pos2.end_fingerX[num]=(Lhand[num]+Lfin)*cos(ang(angle_base1))*

cos(ang(angle_base2+90+angle_wrist))+pos2.wristX-ff[num]*

sin(ang(angle_base1));

pos2.end_fingerY[num]=(Lhand[num]+Lfin)*sin(ang(angle_base1))*

cos(ang(angle_base2+90+angle_wrist))+pos2.wristY+ff[num]*

cos(ang(angle_base1));

pos2.end_fingerZ[num]=(Lhand[num]+Lfin)*sin(ang(angle_base2+90+

angle_wrist))+pos2.wristZ;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }

//***** draw arm&hand TFS *****

    draw_hand_TFS(pos3,pos2);

    }//end if

//*****save old position*****

    pos2.old_wristX=pos2.wristX;           //
    pos2.old_wristY=pos2.wristY;           //
    pos2.old_wristZ=pos2.wristZ;           //

    for(num=0;num<=4;num++){               //save old position
        pos2.old_fingerX[num]=pos2.fingerX[num];           //
        pos2.old_fingerY[num]=pos2.fingerY[num];           //
        pos2.old_fingerZ[num]=pos2.fingerZ[num];           //
        pos2.old_end_finX[num]=pos2.end_fingerX[num];       //
        pos2.old_end_finY[num]=pos2.end_fingerY[num];       //
        pos2.old_end_finZ[num]=pos2.end_fingerZ[num];       //
    }

//*****output text*****

    if(mode==0)outtextxy(525,245,"output angle");
    if(mode==1)outtextxy(525,245,"input angle");
    outtextxy(510,M_F1,"finger1:");
    outtextxy(510,M_F2,"finger2:");
    outtextxy(510,M_F3,"finger3:");
    outtextxy(510,M_F4,"finger4:");
    outtextxy(510,M_F5,"finger5:");
    outtextxy(510,M_T," tumb:");
    outtextxy(510,M_B1," base1:");
    outtextxy(510,M_B2," base2:");
    outtextxy(510,M_W," wrist:");
    setcolor(BLUE);

    outtextxy(570,M_F1,"□□□□□");
    outtextxy(570,M_F2,"□□□□□");
    outtextxy(570,M_F3,"□□□□□");
    outtextxy(570,M_F4,"□□□□□");
    outtextxy(570,M_F5,"□□□□□");
    outtextxy(570,M_T,"□□□□□");
    outtextxy(570,M_B1,"□□□□□");

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเหตุเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outtextxy(570,M_B2,"□□□□□");
outtextxy(570,M_W,"□□□□□");
outtextxy(570,M_ST,"□□□□□");
outtextxy(570,M_EX,"□□□□□");
setcolor(WHITE);
if(mode==1)
{
    outtextxy(510,M_ST," Start ");
    outtextxy(510,M_EX," Exit ");
}
sprintf(temp_outtext,"%d",angle[0]);
outtextxy(575,M_F1,temp_outtext);
sprintf(temp_outtext,"%d",angle[1]);
outtextxy(575,M_F2,temp_outtext);
sprintf(temp_outtext,"%d",angle[2]);
outtextxy(575,M_F3,temp_outtext);
sprintf(temp_outtext,"%d",angle[3]);
outtextxy(575,M_F4,temp_outtext);
sprintf(temp_outtext,"%d",angle[4]);
outtextxy(575,M_F5,temp_outtext);
sprintf(temp_outtext,"%d",angle_t);
outtextxy(575,M_T,temp_outtext);
sprintf(temp_outtext,"%d",angle_base1);
outtextxy(575,M_B1,temp_outtext);
sprintf(temp_outtext,"%d",angle_base2);
outtextxy(575,M_B2,temp_outtext);
sprintf(temp_outtext,"%d",angle_wrist);
outtextxy(575,M_W,temp_outtext);

```

//***** automatic mode *****

```

if(mode==1){ //
    outtextxy(600,menu_a,"ฎ"); //
if(loop==0){ //
    switch(key=getch()){ //
        case U_ARRO:menu_a-=15;break; //
        case D_ARRO:menu_a+=15;break; //
        case ENTER;break; //

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        default:key=' ';          //
    }                            //

    if(menu_a<M_F1)menu_a=M_EX;    //
    if(menu_a>M_EX)menu_a=M_F1;    // mode automatic
}

if(key==ENTER){                  //
    setcolor(BLUE);outtextxy(575,menu_a,"□□□□");
    setcolor(WHITE);//moveto(575,menu_a); //
    for(num=0;num<=3;num++)tp_ang[num]=' ';
    if((loop==0)&&(menu_a!=M_EX))
    {
        num=0;
        do{
            if(menu_a==M_ST)exit=1;
            else if((tp_ang[num]=getch())==ENTER){
                tp_ang[num]=' ';exit=1;if(num==0)exit=2;//exit=2 for maintain old position
            }else exit=0;
            if(tp_ang[num]==BACK_SPACE){
                tp_ang[num]=' ';num--;tp_ang[num]=' ';
            }
            if((tp_ang[num]=='0')||(tp_ang[num]=='1')||(tp_ang[num]=='2')
                ||(tp_ang[num]=='3')||(tp_ang[num]=='4')||(tp_ang[num]=='
                '5')||(tp_ang[num]=='6')||(tp_ang[num]=='7')||(tp_ang[num]
                =='8')||(tp_ang[num]=='9'))num++;
            else tp_ang[num]=' ';
        }while((exit!=1)&&(exit!=2));
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

case'h':angle[0]=2;ox=350;oy=M_F1;och[3]=' ';break;
case'u':angle[1]+=1;ox=320;oy=M_F2;och[3]=' ';break;
case'j':angle[1]=1;ox=350;oy=M_F2;och[3]=' ';break;
case'i':angle[2]+=1;ox=320;oy=M_F3;och[3]=' ';break;
case'k':angle[2]=-1;ox=350;oy=M_F3;och[3]=' ';break;
case'o':angle[3]+=1;ox=320;oy=M_F4;och[3]=' ';break;
case'l':angle[3]=1;ox=350;oy=M_F4;och[3]=' ';break;
case'p':angle[4]+=1;ox=320;oy=M_F5;och[3]=' ';break;
case';':angle[4]=1;ox=350;oy=M_F5;och[3]=' ';break;

case'q':angle_base1+=1;ox=320;oy=M_B1;och[3]=' ';break;
case'a':angle_base1=-1;ox=350;oy=M_B1;och[3]=' ';break;
case'w':angle_base2+=1;ox=320;oy=M_B2;och[3]=' ';break;
case's':angle_base2=-1;ox=350;oy=M_B2;och[3]=' ';break;
case'e':angle_wrist+=1;ox=320;oy=M_W;och[3]=' ';break;
case'd':angle_wrist=-1;ox=350;oy=M_W;och[3]=' ';break;
case't':angle_t+=2;ox=320;oy=M_T;och[3]=' ';break;
case'g':angle_t-=2;ox=350;oy=M_T;och[3]=' ';break;
case'm':angle_base1=angle_base2=angle_wrist=angle[0]=angle[1]
        =angle[2]=angle[3]=angle[4]=angle_t=0;
        ox=320;oy=M_EX+15;och[3]='□';break;
case'x':ox=320;oy=M_EX+30;och[3]='□';break;
default:ox=600;oy=600;och[3]=' '; //
}

if(angle_base1<0)angle_base1=0; //
if(angle_base1>360)angle_base1=360; //
if(angle_base2<0)angle_base2=0; //
if(angle_base2>120)angle_base2=120; //
if(angle_wrist<0)angle_wrist=0; //
if(angle_wrist>90)angle_wrist=90; //check angle
if(angle_t<0)angle_t=0; //
if(angle_t>90)angle_t=90; //
if(angle[0]<0)angle[0]=0; //
if(angle[0]>60)angle[0]=60; //

for(finger_num=1;finger_num<=4;finger_num++) //

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น หากพบเห็นการละเมิดลิขสิทธิ์ของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(angle[finger_num]>90)angle[finger_num]=90; //
} //
io_manual(angle,angle_t,angle_base1,angle_base2,angle_wrist,key);

```

```

setcolor(GREEN);outtextxy(ox,oy,och);setcolor(WHITE);
rectangle(319,M_F1-1,344,M_F1+8);rectangle(349,M_F1-1,374,M_F1+8);
rectangle(319,M_F2-1,344,M_F2+8);rectangle(349,M_F2-1,374,M_F2+8);
rectangle(319,M_F3-1,344,M_F3+8);rectangle(349,M_F3-1,374,M_F3+8);
rectangle(319,M_F4-1,344,M_F4+8);rectangle(349,M_F4-1,374,M_F4+8);
rectangle(319,M_F5-1,344,M_F5+8);rectangle(349,M_F5-1,374,M_F5+8);
rectangle(319,M_T-1,344,M_T+8);rectangle(349,M_T-1,374,M_T+8);
rectangle(319,M_B1-1,344,M_B1+8);rectangle(349,M_B1-1,374,M_B1+8);
rectangle(319,M_B2-1,344,M_B2+8);rectangle(349,M_B2-1,374,M_B2+8);
rectangle(319,M_W-1,344,M_W+8);rectangle(349,M_W-1,374,M_W+8);
rectangle(319,M_EX+14,352,M_EX+23);
rectangle(319,M_EX+29,352,M_EX+38);
outtextxy(335,M_F1-15,"key");
outtextxy(325,M_W+15,"up");outtextxy(347,M_W+15,"down");
outtextxy(320,M_F1," y ");outtextxy(350,M_F1," h ");
outtextxy(320,M_F2," u ");outtextxy(350,M_F2," j ");
outtextxy(320,M_F3," i ");outtextxy(350,M_F3," k ");
outtextxy(320,M_F4," o ");outtextxy(350,M_F4," l ");
outtextxy(320,M_F5," p ");outtextxy(350,M_F5," ; ");
outtextxy(320,M_T," t ");outtextxy(350,M_T," g ");
outtextxy(320,M_B1," q ");outtextxy(350,M_B1," a ");
outtextxy(320,M_B2," w ");outtextxy(350,M_B2," s ");
outtextxy(320,M_W," e ");outtextxy(350,M_W," d ");
outtextxy(320,M_EX+30," E it");outtextxy(320,M_EX+15," Ho e");
setcolor(YELLOW);
outtextxy(320,M_EX+30," x");outtextxy(320,M_EX+15," m ");
setcolor(WHITE);
} //end manual mode

```

```

//*****

```

```

} while(key!='x');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void draw_finger(int *base_finx,int *base_finy,int *x,int *y,int finger_num)
{
    int num;
    moveto(*(base_finx+finger_num),*(base_finy+finger_num));
    for(num=0;num<=2;num++){
        if((finger_num!=0)||(num!=0))
            lineto(*(x+3*finger_num+num),*(y+3*finger_num+num));
        circle(*(x+3*finger_num+num),*(y+3*finger_num+num),3);
    }
    if((finger_num!=0))
        circle(*(base_finx+finger_num),*(base_finy+finger_num),3);
}

void draw_hand(int *base_finx,int *base_finy,struct position pos)
{
    static old_posx[6],old_posy[6];
    int old_poly[]={old_posx[0],old_posy[0],old_posx[1],old_posy[1],
        old_posx[2],old_posy[2],old_posx[3],old_posy[3],
        old_posx[4],old_posy[4],old_posx[5],old_posy[5],
        old_posx[0],old_posy[0]};

    setcolor(BLACK);
    drawpoly(7,old_poly);
    int poly[]={pos.x[0][0],pos.y[0][0],base_finx[0],base_finy[0],base_finx[1],
        base_finy[1],base_finx[4],base_finy[4],base_finx[4],
        base_finy[4]+100,base_finx[1],base_finy[1]+100,pos.x[0][0],
        pos.y[0][0]};

    setcolor(WHITE);
    drawpoly(7,poly);

    old_posx[0]=pos.x[0][0],old_posx[1]=base_finx[0],old_posx[2]=base_finx[1],
    old_posx[3]=base_finx[4],old_posx[4]=base_finx[4],old_posx[5]=base_finx[1];
    old_posy[0]=pos.y[0][0],old_posy[1]=base_finy[0],old_posy[2]=base_finy[1],
    old_posy[3]=base_finy[4],old_posy[4]=base_finy[4]+100,old_posy[5]=
        base_finy[1]+100;

    line(base_finx[1],base_finy[1]+100,base_finx[0],base_finy[0]);
}

```

เอกสาร void out_windows() นี้ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าวิธีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int a;
char b[3];
setbkcolor(BLUE);
rectangle(10,10,(getmaxx()/2)-5,(getmaxy()/2)-5);
rectangle((getmaxx()/2)+5,10,getmaxx()-10,(getmaxy()/2)-5);
rectangle(10,(getmaxy()/2)+5,((getmaxx()/2)-5),getmaxy()-10);
gotoxy(27, 2); cout << "Front View";
gotoxy(65, 2); cout << "Side View";
gotoxy(27,17); cout << "Top View";
}
void draw_hand_TFS(struct base_TFS pos3,struct position2 pos2)
{
    int num;
    setcolor(BLACK);
    circle(pos3.Px1,pos3.Py1,5);circle(pos3.Px2,pos3.Py2,5);
    circle(pos3.Px3,pos3.Py3,5);
    moveto(pos3.Px1,pos3.Py1);lineto(pos3.Px1-pos2.old_wristX,pos3.Py1-
pos2.old_wristZ);
    moveto(pos3.Px2,pos3.Py2);lineto(pos3.Px2-pos2.old_wristY,pos3.Py2-
pos2.old_wristZ);
    moveto(pos3.Px3,pos3.Py3);lineto(pos3.Px3-pos2.old_wristX,pos3.Py3-
pos2.old_wristY);
    circle(pos3.Px1-pos2.old_wristX,pos3.Py1-pos2.old_wristZ,3);
    circle(pos3.Px2-pos2.old_wristY,pos3.Py2-pos2.old_wristZ,3);
    circle(pos3.Px3-pos2.old_wristX,pos3.Py3-pos2.old_wristY,3);
    moveto(pos3.Px1-pos2.old_wristX,pos3.Py1-pos2.old_wristZ);
    for(num=0;num<=4;num++)lineto(pos3.Px1-pos2.old_fingerX[num],
    pos3.Py1-pos2.old_fingerZ[num]);
    lineto(pos3.Px1-pos2.old_wristX,pos3.Py1-pos2.old_wristZ);
    moveto(pos3.Px2-pos2.old_wristY,pos3.Py2-pos2.old_wristZ);
    for(num=0;num<=4;num++)lineto(pos3.Px2-pos2.old_fingerY[num],
    pos3.Py2-pos2.old_fingerZ[num]);
    lineto(pos3.Px2-pos2.old_wristY,pos3.Py2-pos2.old_wristZ);
    moveto(pos3.Px3-pos2.old_wristX,pos3.Py3-pos2.old_wristY);
    for(num=0;num<=4;num++)lineto(pos3.Px3-pos2.old_fingerX[num],
    pos3.Py3-pos2.old_fingerY[num]);
}

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ทางปัญญาของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นหากไม่มีเห็นแต่เพียงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lineto(pos3.Px3-pos2.old_wristX,pos3.Py3-pos2.old_wristY);
for(num=0;num<=4;num++){
moveto(pos3.Px1-pos2.old_fingerX[num],pos3.Py1-pos2.old_fingerZ[num]),
lineto(pos3.Px1-pos2.old_end_finX[num],pos3.Py1-pos2.old_end_finZ[num]);
moveto(pos3.Px2-pos2.old_fingerY[num],pos3.Py2-pos2.old_fingerZ[num]),
lineto(pos3.Px2-pos2.old_end_finY[num],pos3.Py2-pos2.old_end_finZ[num]);
moveto(pos3.Px3-pos2.old_fingerX[num],pos3.Py3-pos2.old_fingerY[num]),
lineto(pos3.Px3-pos2.old_end_finX[num],pos3.Py3-pos2.old_end_finY[num]);
circle(pos3.Px1-pos2.old_fingerX[num],pos3.Py1-pos2.old_fingerZ[num],1);
circle(pos3.Px2-pos2.old_fingerY[num],pos3.Py2-pos2.old_fingerZ[num],1);
circle(pos3.Px3-pos2.old_fingerX[num],pos3.Py3-pos2.old_fingerY[num],1);
circle(pos3.Px1-pos2.old_end_finX[num],pos3.Py1-pos2.old_end_finZ[num],1);
circle(pos3.Px2-pos2.old_end_finY[num],pos3.Py2-pos2.old_end_finZ[num],1);
circle(pos3.Px3-pos2.old_end_finX[num],pos3.Py3-pos2.old_end_finY[num],1);
}
setcolor(WHITE);
circle(pos3.Px1,pos3.Py1,5);circle(pos3.Px2,pos3.Py2,5);circle(pos3.Px3,pos3.Py3,5);
moveto(pos3.Px1,pos3.Py1);lineto(pos3.Px1-pos2.wristX,pos3.Py1-pos2.wristZ);
moveto(pos3.Px2,pos3.Py2);lineto(pos3.Px2-pos2.wristY,pos3.Py2-pos2.wristZ);
moveto(pos3.Px3,pos3.Py3);lineto(pos3.Px3-pos2.wristX,pos3.Py3-pos2.wristY);
circle(pos3.Px1-pos2.wristX,pos3.Py1-pos2.wristZ,3);
circle(pos3.Px2-pos2.wristY,pos3.Py2-pos2.wristZ,3);
circle(pos3.Px3-pos2.wristX,pos3.Py3-pos2.wristY,3);
moveto(pos3.Px1-pos2.wristX,pos3.Py1-pos2.wristZ);
for(num=0;num<=4;num++)lineto(pos3.Px1-pos2.fingerX[num],pos3.Py1-pos2.fingerZ
[num]);
lineto(pos3.Px1-pos2.wristX,pos3.Py1-pos2.wristZ);
moveto(pos3.Px2-pos2.wristY,pos3.Py2-pos2.wristZ);
for(num=0;num<=4;num++)lineto(pos3.Px2-pos2.fingerY[num],pos3.Py2-pos2.fingerZ
[num]);
lineto(pos3.Px2-pos2.wristY,pos3.Py2-pos2.wristZ);
moveto(pos3.Px3-pos2.wristX,pos3.Py3-pos2.wristY);
for(num=0;num<=4;num++)lineto(pos3.Px3-pos2.fingerX[num],pos3.Py3-pos2.fingerY
[num]);

```

เอกสารนี้เป็นเอกสารต้นฉบับที่จัดทำขึ้นเพื่อใช้ในการศึกษาวิจัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดก็ตาม หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

moveto(pos3.Px1-pos2.fingerX[num],pos3.Py1-pos2.fingerZ[num]),lineto(pos3.Px1-
pos2.end_fingerX[num],
pos3.Py1-pos2.end_fingerZ[num]);
moveto(pos3.Px2-pos2.fingerY[num],pos3.Py2-pos2.fingerZ[num]),lineto(pos3.Px2-
pos2.end_fingerY[num],
pos3.Py2-pos2.end_fingerZ[num]);
moveto(pos3.Px3-pos2.fingerX[num],pos3.Py3-pos2.fingerY[num]),lineto(pos3.Px3-
pos2.end_fingerX[num],
pos3.Py3-pos2.end_fingerY[num]);
circle(pos3.Px1-pos2.fingerX[num],pos3.Py1-pos2.fingerZ[num],1);
circle(pos3.Px2-pos2.fingerY[num],pos3.Py2-pos2.fingerZ[num],1);
circle(pos3.Px3-pos2.fingerX[num],pos3.Py3-pos2.fingerY[num],1);
circle(pos3.Px1-pos2.end_fingerX[num],pos3.Py1-pos2.end_fingerZ[num],1);
circle(pos3.Px2-pos2.end_fingerY[num],pos3.Py2-pos2.end_fingerZ[num],1);
circle(pos3.Px3-pos2.end_fingerX[num],pos3.Py3-pos2.end_fingerY[num],1);
}
}

```

```
float ang(int angle)
```

```
{
```

```
float pi = 3.141592654;
```

```
return(angle*2*pi/360);
```

```
}
```

```
check_angle(int temp_angle,int *angle)
```

```
{
```

```
if(temp_angle>*angle)(*angle)++;
```

```
if(temp_angle<*angle)(*angle)--;
```

```
if(temp_angle==*angle)return(0);
```

```
else return(1);
```

```
}
```

```
void io_auto(int *angle,int angle_t,int angle_base1,int angle_base2,
```

```
int angle_wrist)
```

```
{
```

```
int feedback;
```

```
outputb(0x300,2+angle[0]*238.00/60.00); //finger 1
```

เอกสาร set_pwm(2,10);set_drt(2,0);delay(3000);set_drt(2,1);delay(3000);set_pwm(2,0);ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าในรูปแบบใดก็ตาม ห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

void io_manual(int *angle,int angle_t,int angle_base1,int angle_base2,
               int angle_wrist,int key)
{
  outportb(0x300,2+(angle[0]*238.00/60.00));
  switch(key){
    case'u':set_pwm(4,15);set_drt(4,1);delay(100);
             set_pwm(4,0);break;
    case'j':set_pwm(4,10);set_drt(4,0);delay(100);set_pwm(4,0);break;
    case'i':set_pwm(5,15);set_drt(5,1);delay(50);set_pwm(5,0);break;
    case'k':set_pwm(5,10);set_drt(5,0);delay(50);set_pwm(5,0);break;
    case'o':set_pwm(6,15);set_drt(6,1);delay(50);set_pwm(6,0);break;
    case'l':set_pwm(6,10);set_drt(6,0);delay(50);set_pwm(6,0);break;
    case'p':set_pwm(7,15);set_drt(7,1);delay(50);set_pwm(7,0);break;
    case':':set_pwm(7,10);set_drt(7,0);delay(50);set_pwm(7,0);break;
    case'q':set_pwm(2,14);set_drt(2,1);delay(20);set_pwm(2,0);break;
    case'a':set_pwm(2,14);set_drt(2,0);delay(20);set_pwm(2,0);break;
    case'w':set_pwm(1,5);set_drt(1,1);delay(50);set_pwm(1,0);break;
    case's':set_pwm(1,13);set_drt(1,0);delay(50);set_pwm(1,0);break;
    case'e':set_pwm(2,10);set_drt(2,1);delay(10);set_pwm(2,0);break;
    case'd':set_pwm(2,10);set_drt(2,0);delay(10);set_pwm(2,0);break;
    case't':set_pwm(3,15);set_drt(3,1);delay(50);set_pwm(3,0);break;
    case'g':set_pwm(3,15);set_drt(3,0);delay(50);set_pwm(3,0);break;
    case'm':break;
  }
}

getfb(int channel)
{
  int n,temp;
  for(n=0;n<=2;n++){
    outportb(0x303,channel+2);
    outportb(0x303,channel+1);
    temp=inportb(0X304);
    delay(1);
  }
  return(temp);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
void set_dir(int motor,int direction)
{
    static int bit[8];
    int num=0;
    bit[motor]=direction;
    output(0x302,128*bit[7]+64*bit[6]+32*bit[5]+16*bit[4]+
        8*bit[3]+4*bit[2]+2*bit[1]+bit[0]);
}
void set_pwm(int motor,int duty)
{
    output(0x301,(motor*16)+(duty));
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Menu.cpp

```
// Program that demonstrates how to creates a drop-down menu.
```

```
// Uses extended codes and text-window functions
```

```
#include <conio.h> // for text-window functions
```

```
#include <string.h> // for strcmp(), etc.
```

```
#include <process.h> // for exit()
```

```
#include <dos.h>
```

```
#define TRUE 1
```

```
#define U_ARRO 72 // Up-arrow key
```

```
#define D_ARRO 80 // Down-arrow key
```

```
#define MENU_WIDTH 12 // width of menu
```

```
#define MAX_ITEMS 5 // number of menu items
```

```
char *menu_item[MAX_ITEMS] = // menu item names  
{ "MODE 0", "MODE 1", "About.", "Quit"};
```

```
void pre_display(void);
```

```
void display_menu(void); // prototypes
```

```
void draw_box(int left, int top, int width, int height);
```

```
char get_code(void);
```

```
void menu_action(char *item_name);
```

```
char key;
```

```
int vpos; // position of active item on menu (0 at top)
```

```
// Gets key-press from user, changes selected item, takes action.
```

```
int menu(void)
```

```
{
```

```
int a;
```

```
pre_display();
```

```
textbackground(BLUE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(a=0;a<=13;a++){
    window(40-a*3,13-a,40+a*3,13+a);
    delay(20);
    clrscr();
}
window(1,1,80,25);
clrscr();
_setcursortype(_NOCURSОР); // remove cursor
do // cycle forever
{
    display_menu(); // display the menu

    menu_action( menu_item[vpos] );
    switch (key=get_code() ) // get key press from user
    {
        case U_ARRO: // move highlight up
            vpos = (vpos>0) ? --vpos : MAX_ITEMS-2;
            break;

        case D_ARRO: // move highlight down
            vpos = (vpos<MAX_ITEMS-2) ? ++vpos : 0;
            break;

    } // end switch

    if(vpos==2)key=0;
} // end while
while(key!='\r');
if(vpos==0)return(0);
if(vpos==1)return(1);
} // end main()

```

// Displays box and menu items, highlights active item.

```
void display_menu()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 int j;
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

textbackground(CYAN);          // black on white
textcolor(BLACK);
draw_box(35, 7, MENU_WIDTH, MAX_ITEMS+1); // draw the box

for(j=0; j<MAX_ITEMS; j++)    // for each menu item
{
    if(j==vpos)                // if active item,
        textbackground(GREEN); // background
    gotoxy(38, j+8);           // display item name
    cputs( menu_item[j] );
    textbackground(CYAN);     // restore background
}
} // end display_menu()

// Draws a box with a single line around it.
void draw_box(int left, int top, int width, int height)
{
    int j;
    char string[81];           // utility string

    // define box-size window
    window(left, top, left+width-1, top+height-1);

    // Clear window to background color
    clrscr();

    // Now add line so chars on bottom line won't cause scroll
    window(left, top, left+width-1, top+height);

    // Draw top and bottom lines, with corners
    for(j=1; j<width-1; j++) // make string of
        string[j] = '\xC4'; // horizontal characters
    string[width] = '\0'; // end the string

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Insert upper left corner
string[0] = '\xDA';

// Insert upper right corner
string[width-1] = '\xBF';

gotoxy(1,1);      // display top line
cputs(string);

string[0] = '\xC0'; // insert lower left corner
string[width-1] = '\xD9'; // insert lower right corner
gotoxy(1, height); // display bottom line
cputs(string);

// draw vertical lines on left and right
for(j=2; j<height; j++) // for each row
{
    gotoxy(1, j); // on the left end,
    putchar('\xB3'); // draw vertical character
    gotoxy(width, j); // on the right end,
    putchar('\xB3'); // draw vertical character
}

// Restore window to full screen
window(1, 1, 80, 25);
} // end draw_box()

```

```

// Gets key-press from user, returns extended code

```

```

char get_code(void)

```

```

{

```

```

    char key;

```

```

    if( (key=getch()) == 0 ) // if extended code,

```

```

        return( getch() ); // return the code

```

```

    else if ( key == '\r' ) // if [Enter],

```

```

        return(key); // return it

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else          // if anything else,
    return(0);    // return 0
} // end get_code()

```

// Takes action, depending on name of menu item.

```

void menu_action(char *item_name)

```

```

{
int p_x,temp;

gotoxy(1,25);          // lower left
clrhol();              // clear line
if( strcmp(item_name, "MODE 0") == 0 )
    for(p_x=68;p_x>=1;p_x--)
    {
    delay(10);
    delline();
    gotoxy(p_x,25);
    cputs(" MANUAL MODE");
    if(kbhit())break;
    }
else if( strcmp(item_name, "MODE 1") == 0 )
    for(p_x=65;p_x>=1;p_x--)
    {
    delay(10);
    delline();
    gotoxy(p_x,25);
    cputs(" AUTOMATIC MODE");
    if(kbhit())break;
    }
else if( strcmp(item_name, "About..") == 0 )
    for(p_x=27;p_x>=1;p_x--)
    {
    delay(10);
    delline();

    gotoxy(p_x,25);

```

```

cputs(" Robotic Hand : NARIN, PONGSATORN, RATTAKRAI, A-TIP.");

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    if(kbhit())break;
}

// else if( strcmp(item_name, "Print") == 0 )
//  cputs(" Print");

else if( strcmp(item_name, "Quit") == 0 )
//  _setcursortype(_NORMALCURSOR);    // restore cursor
for(p_x=73;p_x>=1;p_x--)
{
delay(10);
delline();
gotoxy(p_x,25);
cputs(" Quit? ");
if(key=='\r')exit(0);
if(kbhit())break;
}
} // end menu_action

void pre_display()
{
int j;
textbackground(3);
clrscr();
_setcursortype(_NOCURSOR);
window(20,6,60,20);
textcolor(YELLOW+BLINK);
textbackground(3);
for (j=0; j<70; j++)
    cputs(" Robotic Hand ");
getch();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมสร้าง pulse width modulate

```
ORG 0000H

PWM1 EQU 20H
PWM2 EQU 21H
PWM3 EQU 22H
PWM4 EQU 23H
PWM5 EQU 24H
PWM6 EQU 25H
PWM7 EQU 26H
PWM8 EQU 27H
PWM9 EQU 28H
PWM10 EQU 29H
PWM11 EQU 30H
PWM12 EQU 31H

JMP START

ORG 000BH
JMP CREAT_PWM

START: MOV TMOD,#00100010B
MOV IE,#10001010B

MOV PWM1,#0D
MOV PWM2,#0D
MOV PWM3,#0D
MOV PWM4,#0D
MOV PWM5,#0D
MOV PWM6,#0D
MOV PWM7,#0D
MOV PWM8,#0D
MOV PWM9,#0D
MOV PWM10,#0D
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV    PWM11,#0D
MOV    PWM12,#0D

MOV    P1,#00H
MOV    P3,#00H

MOV    TL0,#0F0H
MOV    TH0,#0F0H
MOV    R0,#16D      ;SET 16 STEP PER PERIOD
MOV    R1,#0D      ;SET INITIAL %DUTY IS %
SETB   TR0
SET_DUTY: MOV    DPTR,#CHOSE_BIT
MOV    A,P0
ANL    A,#00001111B
MOV    R1,A
MOV    A,P0
ANL    A,#11110000B
RR     A
RR     A
RR     A
JMP    @A+DPTR
CHOSE_BIT: AJMP  BIT1
AJMP  BIT2
AJMP  BIT3
AJMP  BIT4
AJMP  BIT5
AJMP  BIT6
AJMP  BIT7
AJMP  BIT8
AJMP  BIT9
AJMP  BIT10
AJMP  BIT11
AJMP  BIT12
JMP   SET_DUTY
JMP   SET_DUTY
JMP   SET_DUTY

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                JMP     SET_DUTY

BIT1:    MOV     PWM1,R1
                                JMP     SET_DUTY
BIT2:    MOV     PWM2,R1
                                JMP     SET_DUTY
BIT3:    MOV     PWM3,R1
                                JMP     SET_DUTY
BIT4:    MOV     PWM4,R1
                                JMP     SET_DUTY
BIT5:    MOV     PWM5,R1
                                JMP     SET_DUTY
BIT6:    MOV     PWM6,R1
                                JMP     SET_DUTY
BIT7:    MOV     PWM7,R1
                                JMP     SET_DUTY
BIT8:    MOV     PWM8,R1
                                JMP     SET_DUTY
BIT9:    MOV     PWM9,R1
                                JMP     SET_DUTY
BIT10:   MOV     PWM10,R1
                                JMP     SET_DUTY
BIT11:   MOV     PWM11,R1
                                JMP     SET_DUTY
BIT12:   MOV     PWM12,R1
                                JMP     SET_DUTY

```

```

;*****

```

```

CREAT_PWM:  PUSH  ACC
            DEC   R0
            MOV  A,R0
            CJNE A,PWM1,C_PWM2
            SETB P1.0
C_PWM2:    CJNE A,PWM2,C_PWM3
            SETB P1.1
C_PWM3:    CJNE A,PWM3,C_PWM4
            SETB P1.2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

C_PWM4:    CJNE  A,PWM4,C_PWM5
           SETB  P1.3
C_PWM5:    CJNE  A,PWM5,C_PWM6
           SETB  P1.4
C_PWM6:    CJNE  A,PWM6,C_PWM7
           SETB  P1.5
C_PWM7:    CJNE  A,PWM7,C_PWM8
           SETB  P1.6
C_PWM8:    CJNE  A,PWM8,C_PWM9
           SETB  P1.7
C_PWM9:    CJNE  A,PWM9,C_PWM10
           SETB  P3.0
C_PWM10:   CJNE  A,PWM10,C_PWM11
           SETB  P3.1
C_PWM11:   CJNE  A,PWM11,C_PWM12
           SETB  P3.2
C_PWM12:   CJNE  A,PWM12,C_PWMA
           SETB  P3.3
C_PWMA:    JZ     C_PWMB
           SETB  TR0
           POP   ACC
           RETI
C_PWMB:    MOV   R0,#16D
           MOV   P1,#00H
           MOV   P3,#00H
           SETB  TR0
           POP   ACC
           RETI

```

END

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมกำเนิด Position command ควบคุมเซอร์โวมอเตอร์

```
ORG 0000H
JMP START

ORG 000BH ;timer 1 interrupt
JMP M_FREQ

START: MOV IE,#10000010B
MOV TMOD,#00000010B ;SET TIMER MODE 2
MOV TH0,#0F0H
MOV TL0,#0F0H
MOV P1,#00H
MOV P3,#01010101B
MOV R0,#10D
SETB TR0
LOOP: MOV TH0,P0
JMP LOOP

M_FREQ: CJNE R0,#1D,M_F2
SETB P1.0

M_F2: DJNZ R0,END_MF
MOV R0,#10D
MOV P1,#00H
SETB TR0
RETI

END_MF: SETB TR0
RETI
END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ผศ. โยธิน เปรมปราณีรัตน์, “วิเคราะห์และออกแบบระบบควบคุมมอเตอร์”, ๙๙๙., กันยายน ๒๕๒๖.
2. Pool, Harry H. “ Fundamental of Robotic Engineering “, Van Nostrand Reinhold, USA, 1989.
3. Rosheim, Mark E., “ Robot Evolution : The development of anthropotics “ , John Wiley & Sons, Inc., USA, 1994.
4. Rucco, S.R., “ Robot sensors and transducers “, Halsted Press, USA, 1987.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

สำหรับความก้าวหน้าในโครงการมือกลนี้ เกิดขึ้นได้ด้วยความกรุณาของ อาจารย์สุมิตร พนาอุดมทรัพย์ ที่ให้คำปรึกษาแนะนำที่เป็นประโยชน์ต่อโครงการ ขอขอบพระคุณอาจารย์ทุกท่าน ที่ได้ให้ความรู้ในด้านต่าง ๆ ขอขอบคุณภาควิชาชีพวิศวกรรมเครื่องกล ที่ให้ความเอื้อเฟื้ออุปกรณ์และ เครื่องมือมาโดยตลอด และขอขอบคุณเพื่อน ๆ ทุกคนที่ได้ให้กำลังใจและช่วยเหลือในการจัดทำ รายงานฉบับนี้ขึ้น

สุดท้ายนี้ขอขอบพระคุณบิดา มารดา ที่ได้ให้การสนับสนุน ส่งเสริมและให้กำลังใจมาโดย ตลอด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้