แขนกลมหัศจรรย์

MIRACULOUS AUTOMATIC ARM

ชิษณุพงศ์ ฤทธินนทชัย
ณัฐธยาน์ ทองยืน
ธนภัทร เตชะแสนศิริ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมแมคคาทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

แขนกลมหัศจรรย์

MIRACULOUS AUTOMATIC ARM

ชิษณุพงศ์ ฤทธินนทชัย

ณัฐธยาน์ ทองยืน

ธนภัทร เตชะแสนศิริ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมแมคคาทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MIRACULOUS AUTOMATIC ARM

CHITSANUPONG RIDTINONTACHAI

NATTHAYA THONGYUEN

TANAPAT TAECHASANSIRI

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN MECHATRONICS ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2016

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2559

ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง	แขนกลมหัศจรรย์		
	MIRACULOUS AUTOMATIC ARM		
ผู้จัดทำ	นายชิษณุพงศ์	ฤทธินนทชัย	56010317
	นางสาวณัฐธยาน์	ทองยืน	56010389
	นายธนภัทร	เตชะแสนศิริ	56010539

....อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แขนกลมหัศจรรย์

โดย

นายชิษณุพงศ์ ฤทธินนทชัย 56010317

นางสาวณัฐธยาน์ ทองยืน 56010389

นายธนภัทร เตชะแสนศิริ 56010539

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์

ปีการศึกษา 2559

บทคัดย่อ

ในปัจจุบันโรงงานอุตสาหกรรมได้มีการนำหุ่นยนต์หรือแขนกลเข้ามาทำงานในด้านต่างๆ แทนแรงงานมนุษย์ เช่น การทำงานด้านการประกอบชิ้นส่วนอิเล็กทรอนิกส์, การพ่นสีชิ้นงาน, การเคลื่อนย้ายชิ้นงาน เป็นต้น ซึ่งในส่วนของการทำงานด้านการพ่นสีชิ้นงานนั้น โรงงานจะใช้หุ่นยนต์ที่มีขนาดใหญ่หรือหุ่นยนต์ขนาดเล็กหลายๆ ตัวในการพ่นสีให้ครบทุกด้าน ด้วยเหตุนี้จึงเป็นแรงจูงใจให้ผู้พัฒนาได้คิดค้นวิจัย และสร้างแขนกลที่ใช้พ่นสีชิ้นงานได้รอบด้าน ซึ่งแขนกลนี้จะมีขนาดเล็กเพื่อลดพื้นที่การทำงานในโรงงานอุตสาหกรรมและแขนกลนี้ยังสามารถพ่นสีชิ้นงานได้รอบด้าน ซึ่งจะพ่นสีที่ด้านหน้า ด้านข้าง ด้านบน ด้านล่าง และด้านหลังของวัตถุ โดยแขนกลมีการทำงานคือ จะเริ่มพ่นสีจากด้านหน้าของวัตถุจนเต็ม แล้วจึงพ่นสีด้านอื่นๆ ต่อไปในส่วนของการควบคุมการทำงานของแขนกล ผู้พัฒนาได้เขียนโปรแกรมควบคุมการทำงานของแขนกลในตัว Arduino และแขนกลที่พัฒนาขึ้นมานี้สามารถนำไปประยุกต์ใช้ในการทำงานในโรงงานอุตสาหกรรมได้เป็นอย่างดีนั่นเอง เนื่องจากผู้พัฒนาได้ออกแบบแขนกลนี้ให้มีลักษณะการทำงานคล้ายคลึงกับแขนของมนุษย์มากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MIRACULOUS AUTOMATIC ARM

By

Mr. Chitsanupong Ridtinontachai 56010317

Ms. Natthaya Thongyuen 56010389

Mr. Tanapat Taechasansiri 56010539

Advisor

Asst.Prof.Sumit Panaudomsub

Academic Year 2016

ABSTRACT

Nowadays robot systems are crucial in manufacturing industries. Industrial factories use robot systems both robots and automatic arms in manufacturing process. Many factories reboot their manufacturing industry by replacing millions of workers with machines for electronic component assemblies, spraying and moving objects. In Spray painting part, they will use big industrial robots or many small-sized robots. Due to these situations, it inspires developers to invent and create a robotic arm that used for all round spraying. So we made a small-sized of robotic arm that we can save more space in the industrial factory. Moreover this robotic arm can paint every material in any dimension such as front, side, top back and bottom. First, robotic arm will start from front dimension then follow by side and top and last with back and bottom. Considering about control part, developers have designed a program to control a robotic arm by using Arduino. As a result, this robotic arm can be applied in industrial factories and will be used as capable as human's arm.

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ประสบความสำเร็จลุล่วงได้ดี เนื่องด้วยคำแนะนำ คำปรึกษา ความช่วยเหลือเป็นอย่างดีจาก ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์ ซึ่งเป็นอาจารย์ควบคุมปริญญานิพนธ์ ผู้จัดทำรู้สึกซาบซึ้ง และขอกราบขอบพระคุณท่านเป็นอย่างสูง

ขอขอบพระคุณคณาจารย์ทุกท่านที่ให้ความรู้ความเข้าใจในเนื้อหาวิชาตั้งแต่เริ่มเข้ารับการศึกษา เพื่อนำความรู้ที่ได้มาประยุกต์ใช้ในการทำปริญญานิพนธ์ฉบับนี้

ขอขอบพระคุณพี่ๆ ทุกท่านและเพื่อนๆ ทุกคนที่ให้คำแนะนำและคำปรึกษา ตลอดจนช่วยเหลือในการทำปริญญานิพนธ์ฉบับนี้

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา ของผู้จัดทำ ผู้มีพระคุณสูงสุด ผู้ให้โอกาสในการศึกษา ตลอดจนให้คำปรึกษาและความช่วยเหลือด้านต่างๆ และเป็นกำลังใจในการทำปริญญานิพนธ์ฉบับนี้ให้สำเร็จลุล่วงไปด้วยดี

ผู้จัดทำหวังเป็นอย่างยิ่งว่าปริญญานิพนธ์ฉบับนี้จะเป็นประโยชน์สูงสุดต่อผู้ที่สนใจ และหากเกิดข้อผิดพลาดประการใด ทางคณะผู้จัดทำต้องขออภัยมา ณ โอกาสนี้

ผู้จัดทำ
ชิษณุพงศ์ ฤทธินนทชัย
ณัฐธยาน์ ทองยืน
ธนภัทร เตชะแสนศิริ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูป VI

บทที่ 1 บทนำ 1

1.1 ความสำคัญและที่มาของปริญญานิพนธ์ 1

1.2 วัตถุประสงค์ของปริญญานิพนธ์ 2

1.3 ขอบเขตของโครงงาน 2

1.4 ประโยชน์ที่คาดว่าจะได้รับ 2

บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง 3

2.1 โครงสร้างพื้นฐานของแขนกล 3

2.1.1 ชิ้นส่วนกลไกต่างๆ ของหุ่นยนต์ 3

2.1.2 อุปกรณ์ขับเร้า 3

2.1.3 ตัวรับรู้ (Sensors) 3

2.1.4 ส่วนประมวลผลและควบคุม (Controller) 4

2.2 ข้อต่อ 5

2.3 ทฤษฎีพื้นฐานของแขนกล 6

2.3.1 ทฤษฎีจลนศาสตร์ของหุ่นยนต์ 7

2.4 สรีระของแขนกล 8

2.5 อัลกอลิทึมในการควบคุมแขนกล 8

2.6 Servo Motor 10

2.7 สรีระของแขนกล 12

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 3 หลักการออกแบบและโครงสร้างของระบบ	14
3.1 อุปกรณ์สำหรับการทำแขนกล	14
3.1.1 ไม้บัลซ่า	14
3.1.2 เซอร์โวมอเตอร์	14
3.2 การออกแบบแขนกล	16
3.3 คำสั่งในการควบคุมการหมุนของมอเตอร์	16
3.3.1 หลักการทำงานของ RC Servo Motor	16
3.3.2 สัญญาณ RC ในรูปแบบ PWM	17
บทที่ 4 การทดลองและผลการทดลอง	18
4.1 การทดลอง	18
4.1.1 การใช้ Arduino ในการสั่งให้เซอร์โวมอเตอร์ทำงาน	18
4.1.2 การหามุมการเคลื่อนที่ของเซอร์โวมอเตอร์	19
4.1.3 การทดลองสร้างโค้ดควบคุมด้วยโปรแกรม Arduino	20
4.2 ผลการทดลอง	22
บทที่ 5 บทวิจารณ์และสรุป	24
5.1 สรุปผลการทดลอง	24
5.2 ปัญหาที่พบและอุปสรรคที่พบ	24
5.3 แนวทางแก้ไข	25
เอกสารอ้างอิง	26
ภาคผนวก	27
ภาคผนวก ก โปรแกรมคำสั่งที่ใช้ในการเขียนให้ไมโครคอนโทรลเลอร์ให้ทำงาน	28

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 องค์ประกอบของหุ่นยนต์และความสัมพันธ์ระหว่างกัน	4
2.2 ข้อต่อชนิดต่างๆ	5
2.3 Block Diagram ของ Inverse Kinematics	9
2.4 ส่วนประกอบภายในของเซอร์โวมอเตอร์	11
2.5 ขนาดสัญญาณพัลส์ในการควบคุม	11
2.6 สัญญาของพัลส์แบบ PWM : Pulse Width Modulation	12
2.7 โครงสร้างของ Arduino	13
3.1 ไม้บัลซ่า	14
3.2 เซอร์โวมอเตอร์ที่ใช้ในการทำแขนกล	14
3.3 ขนาดของเซอร์โวมอเตอร์ที่ใช้ในการทำแขนกล	15
3.4 คุณสมบัติของเซอร์โวมอเตอร์	15
3.5 การทำงานของเซอร์โวมอเตอร์	16
3.6 การหมุนของเซอร์โวมอเตอร์	17
4.1 ลักษณะของแขนกลด้านหน้า	19
4.2 ลักษณะของแขนกลด้านหลัง	19
4.3 ออกแบบโค้ด Arduino ควบคุมให้เซอร์โวมอเตอร์ทำงาน	21
4.4 การพ่นสีของแขนกลที่ด้านหน้าของวัตถุ	22
4.5 การพ่นสีของแขนกลที่ด้านข้างของวัตถุ	22
4.6 การพ่นสีของแขนกลที่ด้านบนของวัตถุ	23
4.7 การพ่นสีของแขนกลที่ด้านหลังของวัตถุ	23

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของปริญญานิพนธ์

เมื่อกล่าวถึงหุ่นยนต์ แต่ละคนอาจจะมีคำจำกัดความที่แตกต่างกันออกไปตามภาพที่เกิดขึ้นในความคิด ส่วนใหญ่คุ้นเคยกับหุ่นยนต์ที่มีลักษณะคล้ายมนุษย์จากภาพยนตร์ หรือนิยายต่างๆ ที่มีการเคลื่อนไหวและมีความชาญฉลาดใกล้เคียงกับมนุษย์ ในความเป็นจริงแล้วแม้ว่ามนุษย์จะมีนวัตกรรมด้านคอมพิวเตอร์ที่ถูกพัฒนาจนมีสมรรถนะสูงมากในปัจจุบันแล้ว แต่หุ่นยนต์ที่ทันสมัยมากที่สุดก็ยังเทียบไม่ได้กับมนุษย์ทั้งด้านระบบพลวัตและการประมวลผล อย่างไรก็ตามหุ่นยนต์ถือว่าเป็นเครื่องจักรกลที่มีประโยชน์ในการทำงานแทนมนุษย์ ด้วยเหตุนี้จึงจำกัดความอย่างง่ายกับหุ่นยนต์ว่าเป็น "เครื่องจักรกลที่ทำงานแทนคน" ซึ่งหุ่นยนต์จะมีความแตกต่างจากเครื่องจักรกลอัตโนมัติประเภทอื่นที่ควบคุมโดยคอมพิวเตอร์เหมือนกัน เช่น เครื่องซีเอ็นซีประเภทต่างๆ เพราะหุ่นยนต์มีความยืดหยุ่นในการทำงานมากกว่านั่นเอง เครื่องกัดซีเอ็นซีมีการทำงานที่ถูกจำกัดอยู่กับการกัดชิ้นงานในพิกัด 3 มิติ จึงไม่สามารถนำมาเคลื่อนย้ายสิ่งของได้ แต่แขนของหุ่นยนต์สามารถนำมาติดหัวกัดและเขียนโปรแกรมให้กัดชิ้นงานได้ หรือหากต้องการนำไปใช้งานเป็นตัวหยิบจับสิ่งของ ทำการเชื่อมโลหะ ก็เพียงแต่เปลี่ยนหัวเครื่องมือ และเขียนโปรแกรมใหม่เท่านั้น

วิทยาการหุ่นยนต์ (Robotics Science) เป็นศาสตร์ประยุกต์ที่เกิดขึ้นจากการประมวลความรู้จากหลายสาขาวิชาเข้าด้วยกันในการวิเคราะห์ และประดิษฐ์หุ่นยนต์ (Robot) สาขาวิชาที่สำคัญ เช่น สาขาวิชาวิศวกรรมเครื่องกล ซึ่งจะต้องทำการออกแบบและวิเคราะห์กลไกให้สามารถเคลื่อนไหวได้ดังที่ต้องการ สาขาวิชาวิศวกรรมไฟฟ้าที่เกี่ยวข้องกับวงจรต่างๆ ในระบบ เช่น วงจรควบคุม วงจรของตัวรับรู้ (Sensor) วงจรสัญญาณสื่อสาร สาขาวิชาวิศวกรรมคอมพิวเตอร์ที่จะพัฒนาโปรแกรมและขั้นตอนวิธี (Algorithm) ต่างๆ หรือแม้กระทั่งการพัฒนาแพลตฟอร์ม (Platform) เพื่อช่วยส่งเสริมงานพัฒนาโปรแกรมของหุ่นยนต์โดยเฉพาะ นอกจากนี้แล้วความรู้จากสาขาอื่นๆ ยังมีความจำเป็นในการพัฒนาหุ่นยนต์ที่ล้ำสมัย เช่น หุ่นยนต์ที่สามารถเปลี่ยนแปลงรูปได้ (Transformable Robot) จะต้องอาศัยความรู้ด้านวัสดุศาสตร์ ในการสังเคราะห์วัสดุฉลาด (Smart Material) ที่มีคุณสมบัติจำและควบคุมรูปร่างได้ และมีการตอบสนองที่รวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาการทำงานของแขนกล
2. เพื่อศึกษาการควบคุมการเคลื่อนที่ของแขนกล โดยการใช้เซอร์โวมอเตอร์
3. เพื่อศึกษาวิธีการทำงานของ Arduino

1.3 ขอบเขตของโครงงาน

1. ศึกษาการทำงานของแขนกล 5 แกน และ 6 แกน
2. ศึกษาควบคุมตำแหน่งของเซอร์โวมอเตอร์
3. เขียนโปรแกรมควบคุมการทำงานของแขนกล
4. การทดลอง บันทึก และสรุปผล

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้เกี่ยวกับการทำงานพื้นฐานของเซอร์โวมอเตอร์
2. ได้รับความรู้เกี่ยวกับการเคลื่อนที่ของแขนกล 5 แกน
3. สามารถนำโครงงานนี้ไปประยุกต์ด้านอื่นๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 โครงสร้างพื้นฐานของแขนกล

องค์ประกอบของแขนกลจะประกอบด้วยอุปกรณ์และชิ้นส่วนต่างๆ มากมาย ซึ่งแต่ละชนิดจะมีหน้าที่แตกต่างกันไปตามลักษณะและวัตถุประสงค์ของการใช้งาน การเลือกใช้จึงจำเป็นต้องอาศัยทั้งองค์ความรู้และประสบการณ์เพื่อให้ระบบกลไกสามารถทำงานได้อย่างมีประสิทธิภาพ รวดเร็ว คงทน และประหยัดพลังงาน แขนกลสามารถแบ่งส่วนประกอบหลักได้เป็น 4 ส่วน คือ

2.1.1 ชิ้นส่วนกลไกต่างๆ ของหุ่นยนต์

เป็นโครงสร้างทางกายภาพที่ทำงานตามการประมวลผลของคำสั่งควบคุม และสัญญาณ โครงสร้างนี้สามารถจำแนกได้เป็น 2 ส่วนหลัก ได้แก่ ส่วนขับเคลื่อน (Locomotion) ที่ทำให้แขนกลเคลื่อนที่ไปยังตำแหน่งต่างๆ ได้ อีกส่วนหลักที่สำคัญของโครงสร้างก็คือ ส่วนตัวกระทำ (Manipulation) ซึ่งจะทำให้หุ่นยนต์สามารถทำงานต่างๆ ได้ตามต้องการ กลไกในส่วนนี้มักจะเป็นแขนและมือของหุ่นยนต์ การออกแบบตัวแขนกลนั้นเป็นศาสตร์และศิลป์ที่นักออกแบบจะต้องประยุกต์ความรู้จากหลายแขนงวิชาและเทคโนโลยีที่มีอยู่ เพื่อให้บรรลุวัตถุประสงค์ของโครงงาน

2.1.2 อุปกรณ์ขับเร้า

อุปกรณ์ที่สามารถเปลี่ยนแปลงพลังงานไฟฟ้าที่ป้อนเข้ามาให้กลายเป็นการกระจัด การเคลื่อนที่หรือแรง เช่น มอเตอร์ไฟฟ้า ระบบนิวแมติกส์ และระบบไฮโดรลิกส์ ซึ่งเป็นส่วนที่ทำให้หุ่นยนต์เกิดการเคลื่อนที่และทำงาน ด้วยข้อจำกัดทางด้านเทคโนโลยีในปัจจุบัน ต้นกำลังในหุ่นยนต์จะใช้มอเตอร์เป็นส่วนใหญ่ และอาจจะมีกลไกซึ่งทำหน้าที่ลดทอนความเร็วและเพิ่มแรงขับให้เหมาะสมกับภาระ (Load) หุ่นยนต์อาจขับเคลื่อนด้วยแรงดันจากของไหล (Fluid Power) สำหรับงานที่มีภาระหนัก แต่ในอนาคตอาจจะมีนวัตกรรมใหม่ทางด้านต้นกำลัง เช่น การใช้วัสดุตระกูลพอลิเมอร์ (Polymer) ในการสร้างระบบตัวขับแบบกระจาย (Distributed Actuator System) เพื่อขับเคลื่อนหุ่นยนต์ ดังเช่นร่างกายมนุษย์ที่เคลื่อนไหวได้ด้วยกล้ามเนื้อ

2.1.3 ตัวรับรู้ (Sensors)

เป็นส่วนที่ทำหน้าที่ตรวจวัด (Acquire) ข้อมูลที่เกี่ยวข้องกับระบบและสิ่งแวดล้อมในแบบทันที (Real Time) เพื่อป้อนให้ส่วนประมวลผลและควบคุมต่อไป ข้อมูลอาจจะมีโครงสร้างที่ง่าย เช่น

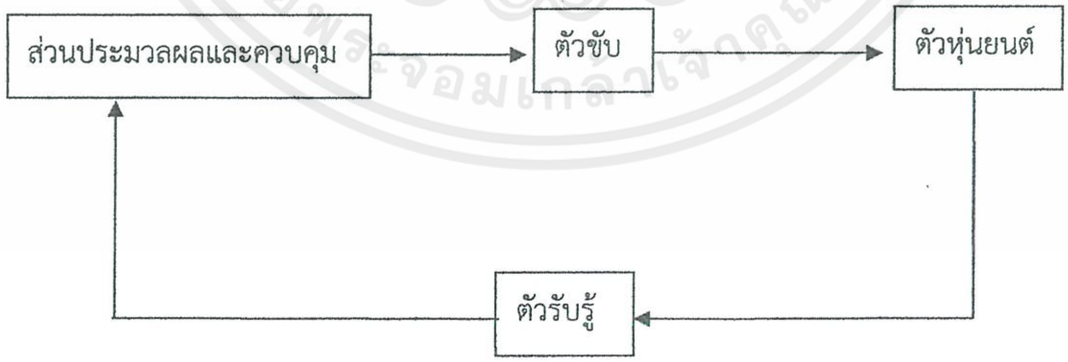
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งหรือความเร็วของตัวขับ หรืออาจจะเป็นข้อมูลที่มีขนาดใหญ่และซับซ้อน เช่น ข้อมูลที่ได้จากกล้อง ณ เวลาหนึ่ง อาจจะประกอบไปด้วยข้อมูลสีของจุดภาพ (Pixel) ในระนาบหนึ่งๆ และข้อมูลความลึก (Depth) ของพื้นผิวที่อยู่ในมุมมอง ณ ขณะเวลานั้น

2.1.4 ส่วนประมวลผลและควบคุม (Controller)

ทำหน้าที่คล้ายสมองของมนุษย์ กล่าวคือ จะรับข้อมูลจากตัวรับรู้ต่างๆ (Sensor Data Fusion) เพื่อนำมาประมวลผลและคำนวณสัญญาณควบคุม (Control Signal) ส่งต่อไปยังตัวขับ เช่น สมองกลที่ประดิษฐ์จากอุปกรณ์อิเลคทรอนิกส์ เครื่องควบคุมขนาดเล็ก คอมพิวเตอร์ชนิดแผงวงจรสำเร็จรูป เครื่องควบคุมเชิงตรรกะที่สามารถโปรแกรมได้ และคอมพิวเตอร์ส่วนบุคคล เพื่อให้แขนกลทำงานตามความต้องการ โดยทั่วไปแล้วคอมพิวเตอร์มักถูกนำมาใช้ในการดำเนินการ (Implement) สำหรับส่วนนี้ ปัจจุบันได้เริ่มมีงานวิจัยที่ใช้สมองของสัตว์เลี้ยงลูกด้วยนม เช่น ลิงและผู้พิการทางแขน เป็นส่วนประมวลผลและควบคุมหุ่นยนต์ หรือแขนเทียม (Bionic Arm)

โดยทั่วไปแล้วองค์ประกอบทั้ง 4 ส่วนนี้ มักจะมีความสัมพันธ์กัน ระบบหุ่นยนต์จะรับคำสั่ง ซึ่งอาจจะเป็นคำสั่งขั้นสูง เช่น การสื่อสารของมนุษย์ ส่วนประมวลผลและควบคุมจะทำหน้าที่แปลงคำสั่งเหล่านั้นเป็นคำสั่งปฐมฐาน (Primitive Command) และนำไปประมวลกับข้อมูลที่ได้รับจากตัวรับรู้ เพื่อที่จะคำนวณสัญญาณควบคุมตัวขับ และให้หุ่นยนต์ทำงานตามคำสั่งต่อไป ในอนาคตอาจจะมีการรวมหน่วย (Integration) ขององค์ประกอบทั้ง 4 ส่วนของหุ่นยนต์ที่ไม่สามารถแบ่งแยกได้อย่างชัดเจนอีกต่อไป ประกอบกับการพัฒนาการที่ก้าวกระโดดในงานวิจัยด้านความฉลาดเทียม (Artificial Intelligence) ทำให้หุ่นยนต์มีคุณลักษณะและความสามารถเข้าใกล้มนุษย์มากขึ้นทุกขณะ ดังในบทละครหรือนวนิยาย ดังรูปที่ 2.1

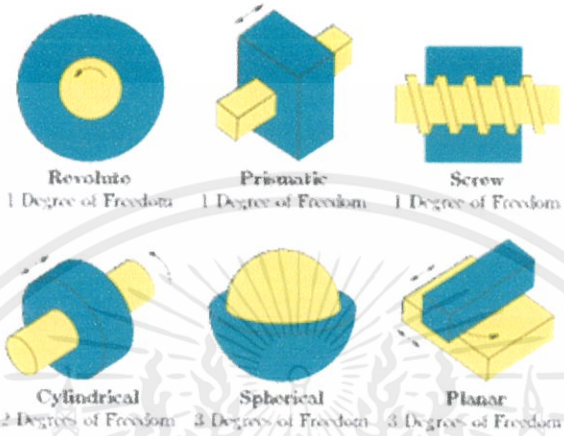


รูปที่ 2.1 องค์ประกอบของหุ่นยนต์และความสัมพันธ์ระหว่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ข้อต่อ

แขนกลคือ หุ่นยนต์ที่ประกอบไปด้วยท่อนแขน (Link) ที่นำมาประกอบกันด้วยข้อต่อ (Joint) ข้อต่อมีหลายแบบ แต่ละแบบก็จะอนุญาตให้เกิดการเคลื่อนที่ของท่อนแขนที่แตกต่างกันไป ดังรูปที่ 2.2 ลองพิจารณาว่าข้อต่อแต่ละแบบบังคับการเคลื่อนที่ของท่อนแขนสองท่อนที่มาเชื่อมกันอย่างไร



รูปที่ 2.2 ข้อต่อชนิดต่างๆ

ในการสร้างแขนกลโดยทั่วไป ข้อต่อที่นิยมใช้มากที่สุดคือข้อต่อแบบหมุน (Revolute Joint) และข้อต่อแบบเลื่อน (Prismatic Joint) สำหรับข้อต่อแบบหมุนท่อนแขนสองท่อนถูกยึดติดกันที่จุดหมุนซึ่งอยู่บนท่อนแขน โดยแต่ละท่อนสามารถหมุนได้รอบจุดหมุนนี้ จะสามารถบอกตำแหน่งของท่อนแขนสองท่อนที่สัมพันธ์กันด้วยมุมที่ท่อนแขนหมุนไป ส่วนข้อต่อแบบเลื่อนนั้นท่อนแขนสองท่อนติดอยู่ด้วยกันในลักษณะเดียวกันกับเสาอากาศวิทยุรถยนต์ที่ยืดหดได้ โดยท่อนแขนแต่ละท่อนสามารถเลื่อนเข้าออกได้ในหนึ่งทิศทาง จึงสามารถระบุตำแหน่งที่สัมพันธ์กันของท่อนแขนสองท่อนได้จากระยะเลื่อนเข้าออกดังกล่าว จะเห็นได้ว่าข้อต่อแบบเลื่อนมีระดับการเคลื่อนที่เสรีเป็นหนึ่ง ซึ่งเรียกตัวแปรที่กำหนดการเคลื่อนที่นี้ซึ่ง ได้แก่ มุมหมุนของข้อต่อแบบหมุน และระยะเลื่อนของข้อต่อแบบเลื่อนว่าเป็นพารามิเตอร์ของข้อต่อ การมีระดับเสรีของการเคลื่อนที่เป็นหนึ่งทำให้ง่ายในการออกแบบและวิเคราะห์ ข้อต่อทั้งสองแบบจึงถูกนำมาใช้งานมากที่สุดในการสร้างแขนกล โดยแขนกลที่มีระดับเสรีสูงๆ ก็สามารถสร้างขึ้นได้ โดยการประกอบท่อนแขนหลายๆ ท่อนด้วยข้อต่อสองแบบนี้

แขนกลทำงานด้วยการเคลื่อนที่ของท่อนแขนที่สัมพันธ์กันเพื่อให้ปลายแขน (End Effector) ไปอยู่ในตำแหน่งและทิศทางที่เหมาะสม เพื่อเครื่องมือที่ติดอยู่ที่ปลายแขนจะได้ทำงานตามที่ต้องการได้อย่างสะดวก และมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำนวณว่าปลายแขนจะอยู่ที่ตำแหน่งและทิศทางใดจึงเป็นเรื่องสำคัญ การคำนวณดังกล่าวอาศัยการกำหนดให้ท่อนแขนแต่ละท่อนมีพิกัดส่วนตัวที่เรียกว่าเฟรม เฟรมประกอบไปด้วยจุดกำเนิดและเวกเตอร์แกน โดยเฟรมที่กล่าวถึงจะอยู่ติดแน่นกับท่อนแขนที่เป็นเจ้าของเสมอ หรืออีกนัยหนึ่งก็คือ แต่ละท่อนแขนจะยังอยู่นิ่ง เมื่อเทียบกับเฟรมของมัน สำหรับท่อนแขนที่เกิดจากการเรียงต่อกันไปนิยมเรียกท่อนแขนที่อยู่นิ่งยึดติดกับพื้นว่า ฐาน (Base) และเรียกท่อนถัดมาตามชื่อส่วนของแขนได้แก่ ไหล่ (Shoulder) ข้อศอก (Elbow) แขนท่อนบน (Forearm) และข้อมือ (Wrist) เป็นต้น ตำแหน่งและทิศทางการวางตัวของท่อนแขนหนึ่งๆ เมื่อเทียบกับเฟรมของฐานจึงขึ้นอยู่กับ ตำแหน่งและทิศทางของท่อนแขนก่อนๆ ด้วย จึงสามารถคำนวณตำแหน่งและทิศทางของปลายแขนได้ด้วยการใช้การแปลงเอกพันธ์ที่ได้ศึกษาไปแล้ว โดยทำการคูณเมทริกซ์การแปลงแบบซ้ายไปขวาพิจารณาจากฐานไปยังส่วนปลายแขน

2.3 ทฤษฎีพื้นฐานของแขนกล

แขนกล (Manipulator) คือ ข้อต่อต่างๆ ที่ต่อกันเพื่อเลียนแบบการเคลื่อนที่ของแขนได้อย่างอิสระหรือเปรียบได้กับแขนของมนุษย์ ซึ่งมีส่วนประกอบหลักทางกลที่มีความสำคัญในการวิเคราะห์แขนหุ่นยนต์ เริ่มจากฐาน (Base) ที่ยึดติดกับผนัง พื้น เพดาน หรือหากต้องการเพิ่มขอบเขตการทำงานก็สามารถเพิ่มการติดตั้งล้อ หรือรางเลื่อนก็ได้ ข้อต่อ (Joints) คือ ตัวขับให้แขนเคลื่อนที่ได้ ซึ่งจะเปรียบได้กับอวัยวะของมนุษย์คือ ไหล่ ข้อศอก ข้อมือ เป็นต้น ส่วนที่เชื่อมต่อระหว่างข้อต่อเรียกว่าก้านต่อ (Links) เปรียบได้กับท่อนขนมนุษย์ และส่วนสัมผัสกับชิ้นงานเรียกว่า ตัวทำงานส่วนปลาย (End Effector) หรือเครื่องมือปลายแขน (End of Arm Tooling) เป็นส่วนที่ติดตั้งเครื่องมือที่มีความแตกต่างกันไปตามลักษณะงาน ตัวอย่างเช่น หัวเชื่อมไฟฟ้าสำหรับงานเชื่อมโลหะ มือจับ (Gripper) สำหรับงานเคลื่อนย้ายวัสดุ หัวฉีดสำหรับพ่นสี เครื่องมือกัด หรือหินเจียร โดยมีมอเตอร์เป็นตัวควบคุมการเคลื่อนที่ของแต่ละข้อต่อ แขนกลถูกออกแบบให้เคลื่อนที่โดยการเลียนแบบการเคลื่อนไหวของมนุษย์ เพื่อให้สามารถทำงานแทนมนุษย์ได้โดยผ่านการควบคุมจากโปรแกรมคอมพิวเตอร์อีกทีหนึ่ง แขนกลนั้นใช้ในงานอุตสาหกรรมต่างๆ เช่น งานเชื่อมโลหะ งานพ่นสี งานจับวางอุปกรณ์ตามตำแหน่ง และงานประกอบผลิตภัณฑ์ เช่น การวางตำแหน่งของตัวไอซีลงบนแผ่นวงจร หรือการประกอบชิ้นส่วนรถยนต์ ซึ่งปัจจุบันบางส่วนของงาน หรือทั้งหมดของกระบวนการจะมีการทำงานโดยแขนกลทั้งสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

องค์ประกอบของระบบแขนกลที่ดูซับซ้อนสามารถแบ่งออกได้อย่างชัดเจนได้เป็น 3 กลุ่ม ได้แก่

1. ฮาร์ดแวร์
2. คอมพิวเตอร์หรือคอนโทรลเลอร์
3. โปรแกรมควบคุม

2.3.1 ทฤษฎีจลนศาสตร์ของหุ่นยนต์

จลนศาสตร์ หรือที่เรียกว่า Kinematics เป็นการศึกษาเกี่ยวกับตำแหน่ง (Position) ทิศทางการหมุน (Orientation) และการเคลื่อนที่ทั้งเชิงเส้น (Translational Motion) และเชิงมุม (Rotational Motion) โดยมิได้คำนึงถึงปัจจัยที่ก่อให้เกิดการเคลื่อนที่ โดยพื้นฐานแล้วหุ่นยนต์เป็นระบบพลศาสตร์ที่ต้องเกี่ยวข้องกับการเคลื่อนที่ผ่านตัวขับ และเพื่อให้หุ่นยนต์สามารถทำงานได้หลากหลาย มันจะต้องสามารถเคลื่อนที่ในปริภูมิสามมิติได้ นี่จึงเป็นสาเหตุที่ทำให้การศึกษาเกี่ยวกับการเคลื่อนที่ของหุ่นยนต์นั้น มีความจำเป็นที่จะต้องเริ่มต้นจากการศึกษาจลนศาสตร์ของปริภูมิสามมิติ ทั้งนี้การเคลื่อนที่ในปริภูมิสองมิติที่คุ้นเคยกันดีถือเป็นเพียงกรณีเฉพาะเท่านั้น ซึ่งจลนศาสตร์สำหรับหุ่นยนต์นั้นมีการพิจารณาอยู่ 2 แบบด้วยกัน คือ Forward Kinematics และ Inverse Kinematics

1. Forward Kinematics คือ การวิเคราะห์การเคลื่อนที่และทิศทางของตัวทำงานส่วนปลาย เมื่อทราบค่าตำแหน่งข้อต่อของหุ่นยนต์ ซึ่งเป็นการคำนวณตามลำดับจากข้อต่อไปยังส่วนปลายของแขนหุ่นยนต์ สามารถเขียนสมการการเคลื่อนที่แบบ Forward Kinematics โดย Forward Kinamatics จะทำการคำนวณหาตำแหน่งที่ปลายของแขนกล (End Effectors) โดยคำนวณจากสมการการเคลื่อนที่ของหุ่นยนต์ ซึ่งคำตอบที่ออกมาจะอยู่ในรูปของจุด (X, Y, Z) หรือระบบคาร์ทีเซียน โดยต้องทราบตัวแปรที่ข้อต่อของแต่ละข้อต่อ (Joint Variables) กรณีข้อต่อแบบหมุนตัวแปรที่ต้องทราบค่า คือ มุม θ ที่ทราบค่าในแต่ละข้อต่อ แล้วจึงนำไปคำนวณหาค่าพิกัดที่ปลายของแขนกล

2. Inverse Kinematics คือ การวิเคราะห์หาตำแหน่งข้อต่อต่างๆ ของหุ่นยนต์ เมื่อรู้ค่าตำแหน่งและทิศทางของตัวทำงานส่วนปลาย (ตำแหน่งเป้าหมาย) ซึ่งเป็นการคำนวณย้อนกลับจากตัวทำงานส่วนปลายมายังข้อต่อต่างๆ Forward Kinematics เป็นการคำนวณหาตำแหน่ง และทิศทางของมือหุ่นยนต์จากค่าของมุมข้อต่อและขนาดของลิงค์ ซึ่งเป็นการแทนค่าตัวแปร (มุมข้อต่อ) เพื่อหาตำแหน่งของหุ่นยนต์ ในทางกลับกัน Inverse Kinematics เป็นการคำนวณย้อนกลับ จากการระบุตำแหน่งและทิศทางของมือหุ่นยนต์ที่ต้องการ แล้วหาค่ามุมของทุกข้อต่อที่ต้องหมุนไป หมายความว่า ต้องหาค่ามุม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดักแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อต่อจากตำแหน่งเป้าหมายของหุ่นยนต์ ในทางปฏิบัติแล้ว Inverse Kinematics มีความสำคัญมากกว่า เพราะส่วนควบคุมต้องคำนวณค่ามุมของแต่ละข้อต่อจากข้อมูลตำแหน่งเป้าหมาย และสั่งการให้หุ่นยนต์เคลื่อนที่ไป สมการที่ใช้บอกตำแหน่งและทิศทางของหุ่นยนต์นั้น ขึ้นกับโครงสร้างและประเภทของหุ่นยนต์ หุ่นยนต์ที่มีลักษณะต่างกันก็จะมีสมการกับตำแหน่งและทิศทางที่ต่างกันออกไป สำหรับการคำนวณ Forward Kinematics ทำได้โดยการแทนค่าตัวแปรของข้อต่อและลิงค์ลงในชุดสมการของหุ่นยนต์ เพื่อหาค่าตำแหน่งและทิศทาง แต่การคำนวณ Inverse Kinematics ค่าตัวแปรของข้อต่อนั้นเป็นผลลัพธ์ที่ได้จากการแก้สมการ เมื่อแทนค่าตำแหน่งและทิศทางของตัวทำงานส่วนปลายลงไปในสมการของหุ่นยนต์ กรอบอ้างอิงได้ถูกกำหนดไว้ที่ข้อต่อและตัวทำงานส่วนปลาย เพื่อระบุตำแหน่งและทิศทาง ทำให้ง่ายต่อการคำนวณและสามารถอ้างอิงเทียบกับกรอบอ้างสมบูรณ์ได้

2.4 สรีระของแขนกล

องค์ประกอบพื้นฐานที่ครอบคลุมการเคลื่อนที่ของแขนกลคือ การออกแบบของรูปร่างและความซับซ้อนของโปรแกรมควบคุมแขนกลถูกออกแบบมาให้มีรูปแบบและขนาดต่างๆ มากมาย และโปรแกรมควบคุมเป็นส่วนทำให้เกิดผลสูงสุดกับความต้องการ ในแต่ละลักษณะเฉพาะของแขนกลในสภาพแวดล้อม 3 มิติ ปัจจุบันนี้มีวิธีอันหลากหลายที่จะออกแบบแขนกลให้ขยับไปถึงยังทุกจุด ในขอบเขตการทำงานของตัวเองโดยจะต้องมีมอเตอร์ 1 ตัวต่อหนึ่งองศาอิสระ (Degree of Freedom) อย่างน้อยที่สุดต้องมีมอเตอร์ 3 ตัว เพื่อทำงานในตำแหน่งสามมิติ (ระนาบของแกน X, Y และ Z) และอีก 3 ตัว เพื่อการหมุนมือ (Roll, Pitch และ Yaw)

2.5 อัลกอลิทึมในการควบคุมแขนกล

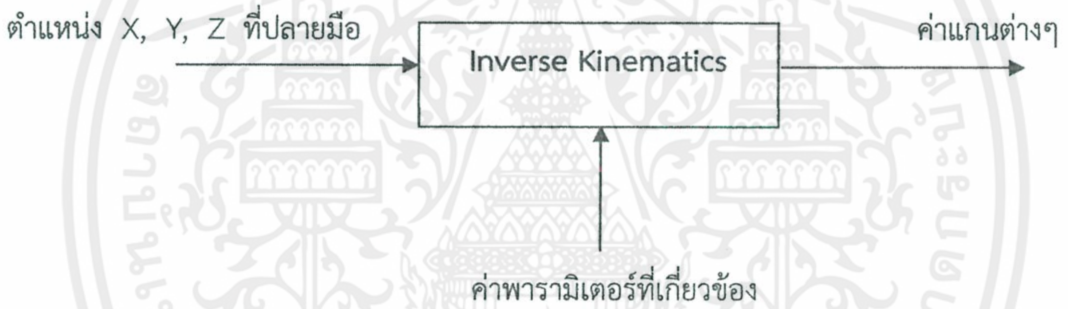
สิ่งหนึ่งที่บ่งบอกถึงประสิทธิภาพในการทำงานของแขนกลคือ ความแม่นยำและเที่ยงตรง ซึ่งการเคลื่อนไหวของการทำงานของแขนกลจะประกอบด้วย การเคลื่อนไหวของ ฐาน ไหล่ ข้อศอก และข้อมือ เป็นต้น การเคลื่อนไหวตามส่วนต่างๆ นี้ จะส่งผลให้มือเคลื่อนที่ไปตามตำแหน่งต่างๆ ที่ปฏิบัติงาน ดังนั้นถ้าส่วนต่างๆ นี้มีการควบคุมที่ไม่ดีพอ จะส่งผลถึงความแม่นยำและความเที่ยงตรงในการทำงานนั้น ดังนั้นมือของแขนกลจะเคลื่อนที่ไปตามตำแหน่งใดๆ นั้น จะขึ้นอยู่กับข้อต่อต่างๆ ที่มีการเคลื่อนไปในตำแหน่งต่างๆ ที่เหมาะสมและสอดคล้องกันด้วย ซึ่งการควบคุมก็ต้องควบคุมตำแหน่งของข้อต่อต่างๆ ให้เหมาะสมกัน โดยสามารถแสดงได้ด้วยสมการคณิตศาสตร์ สำหรับสมการที่จะกล่าวถึงในที่นี้มี 2 ลักษณะด้วยกันคือ สมการ Direct Kinematics และสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Inverse Kinematics ซึ่งสมการทั้ง 2 ลักษณะจะแสดงความสัมพันธ์ของข้อต่อต่างๆ และตำแหน่งปลายมือของแขนกล

Degree of Freedom ของแขนกล หรือ DOF หมายถึงจำนวนข้อต่อ หรือจุดต่อที่เคลื่อนที่เป็นอิสระต่อกัน ส่วนอีกคำหนึ่งคือ Link หมายถึงสิ่งที่เชื่อมต่อระหว่างข้อต่อกับข้อต่อ ซึ่งข้อต่อนั้นอาจจะเป็นการเคลื่อนที่แบบ R หรือแบบ P ก็ได้

ความหมายของสมการ Inverse Kinematics จะมีความหมายที่ตรงข้ามกับสมการ Direct Kinematics คือ จากเดิมป้อนค่าของมุมที่ข้อต่อต่างๆ เข้าไป โดยผลลัพธ์ที่ได้จะได้ค่าเป็นตำแหน่ง X, Y และ Z แต่สมการ Inverse Kinematics นี้ จะเป็นการป้อนตำแหน่ง X, Y และ Z ของปลายมือเข้าไปแล้ว จะได้ผลลัพธ์เป็นมุมของแต่ละข้อต่อว่าต้องเคลื่อนไปกี่องศาจากตำแหน่ง Home ปลายมือจึงจะไปอยู่ตามตำแหน่งที่ต้องการ ดังรูปที่ 2.3



รูปที่ 2.3 Block Diagram ของ Inverse Kinematics

โดยค่าพารามิเตอร์ต่างๆ ยังคงใช้ค่าเหมือนเดิมทุกประการ จากความหมายของสมการ Inverse Kinematics เมื่อป้อนตำแหน่ง X, Y และ Z ที่ปลายมือและค่าพารามิเตอร์ที่เกี่ยวข้อง ผลลัพธ์ที่ได้จะเป็นมุมของแกนต่างๆ คือ X, Y และ Z

สำหรับการประยุกต์ใช้งานอัลกอลิทึมของสมการ Inverse Kinematics อันดับแรกที่เห็นได้ชัด คือ ควบคุมการจับวัตถุที่ตำแหน่ง X, Y และ Z ต้นทาง ให้นำไปวางที่ตำแหน่ง X, Y และ Z ปลายทาง โดยคอมพิวเตอร์จะเป็นตัวคำนวณค่าของมุมที่ข้อต่อต่างๆ เมื่อคำนวณเสร็จ คอมพิวเตอร์จะส่งค่ามุมต่างๆ ไปให้กับชุดขับเคลื่อน โดยชุดขับเคลื่อนก็จะทำการขับเคลื่อนข้อต่อต่างๆ ไปตามมุมที่ได้มาจากอัลกอลิทึมของสมการ Inverse Kinematics ผลลัพธ์ที่ได้ก็คือ ปลายมือจะอยู่ในตำแหน่งที่ต้องการ จากนั้นก็เป็นหน้าที่ของผู้เขียนโปรแกรมควบคุมว่าจะให้ทำอย่างไรต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

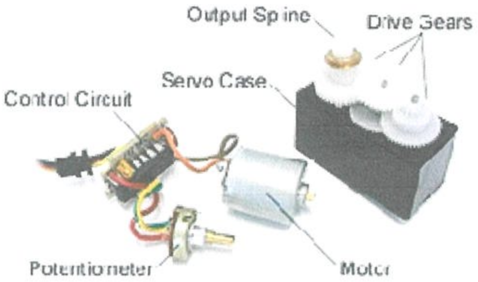
จากอัลกอลิทึมของสมการ Inverse Kinematics ผลลัพธ์ที่ได้ก็คือ ปลายมือจะอยู่ในตำแหน่งที่ต้องการ จากนั้นก็เป็นหน้าที่ของผู้เขียนโปรแกรมควบคุมว่าจะให้ทำอย่างไรต่อไป

อัลกอลิทึมของสมการ Inverse Kinematics เป็นเพียงส่วนที่ช่วยคำนวณค่ามุม โดยที่แต่ละข้อต้องเคลื่อนที่ไปเพื่อให้มืออยู่ในตำแหน่งที่ต้องการ ส่วนที่สองของการประยุกต์ใช้งานของอัลกอลิทึมของสมการ Inverse Kinematics คือ เป็นการใช้ควบคุมกับกล้องในการมองภาพ กล้องจะทำหน้าที่ประมวลสัญญาณภาพที่รับเข้ามา แล้วจึงเปลี่ยนให้เป็นตำแหน่ง X, Y และ Z ของวัตถุแล้วส่งค่า X, Y และ Z ไปให้กับอัลกอลิทึมของสมการ Inverse Kinematics ซึ่งจะทำให้แขนกลสามารถทำงานได้เหมือนกับว่ามีดวงตาในการมองเห็นวัตถุ

2.6 Servo Motor

เซอร์โวมอเตอร์ (Servo Motor) เป็นมอเตอร์ที่มีการควบคุมการเคลื่อนที่ของมัน (State) ไม่ว่าจะเป็นระยะทาง, ความเร็ว และมุมการหมุน โดยใช้การควบคุมแบบป้อนกลับ (Feedback Control) ซึ่งเซอร์โวมอเตอร์เป็นอุปกรณ์ที่สามารถควบคุมเครื่องจักรกล หรือระบบการทำงานนั้นๆ ให้เป็นไปตามความต้องการ เช่น ควบคุมความเร็ว (Speed), ควบคุมแรงบิด (Torque), ควบคุมแรงตำแหน่ง (Position), ระยะทางในการเคลื่อนที่หมุน (Position Control) ของตัวมอเตอร์ได้ ซึ่งมอเตอร์ทั่วไปไม่สามารถควบคุมในลักษณะงานเบื้องต้นได้ โดยให้ผลลัพธ์ตามความต้องการที่มีความแม่นยำสูง ขนาดของ Servo Motor จะมีหน่วยในการบอกขนาดเป็นวัตต์ (Watt) Servo Motor ของ Panasonic จะมีขนาดตั้งแต่ 50W-15kW ทำให้ผู้ใช้งานมีความหลากหลายในการใช้งาน ดังรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



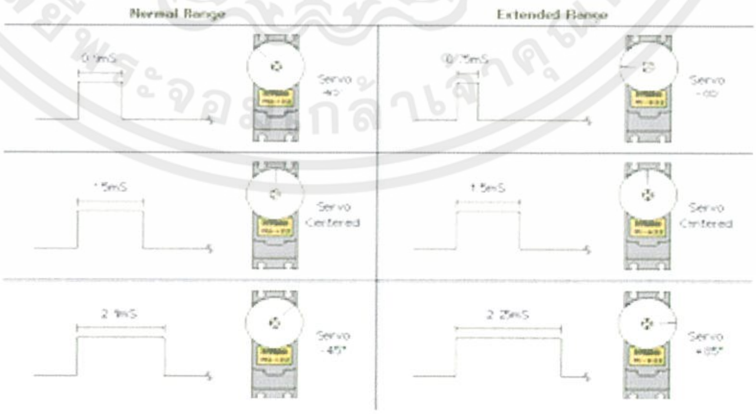
รูปที่ 2.4 ส่วนประกอบภายในของเซอร์โวมอเตอร์

การทำงานของเซอร์โวมอเตอร์ จะมีการทำงานโดยการหมุนของเซอร์โวมอเตอร์ จะถูกควบคุมด้วยพัลส์หรือลอจิก 1 (5 V) เป็นระยะเวลาที่กำหนดและหน่วงเวลาหรือส่งลอจิก 0 เป็นระยะเวลาคงที่ 20 มิลลิวินาทีสลับกันไป ส่งผลให้สามารถควบคุมการหมุนของเซอร์โวมอเตอร์ได้ โดยทิศทางการหมุนเป็นดังนี้

1. เซอร์โวมอเตอร์จะหมุนไปทางขวา (ตามเข็มนาฬิกา) เมื่อพัลส์บวกมีความกว้าง 1 มิลลิวินาที (1 ms) และพัลส์ลบ 20 มิลลิวินาที (20 ms)

2. เซอร์โวมอเตอร์จะหมุนไปทางซ้าย (ทวนเข็มนาฬิกา) เมื่อพัลส์บวกมีความกว้าง 2 มิลลิวินาที (2 ms) และพัลส์ลบ 20 มิลลิวินาที (20 ms)

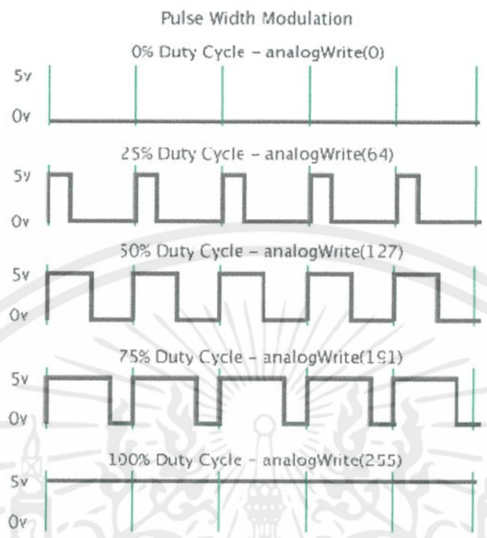
3. เซอร์โวมอเตอร์จะอยู่กึ่งกลาง (หยุดหมุน) เมื่อพัลส์บวกมีความกว้าง 1.5 มิลลิวินาที (1.5 ms) และพัลส์ลบ 20 มิลลิวินาที (20 ms) หากเซอร์โวมอเตอร์ไม่หยุดหมุน จะต้องปรับค่าความต้านทานของเซอร์โวมอเตอร์ใหม่ โดยช่องปรับจะอยู่ใกล้กับสายไฟของเซอร์โวมอเตอร์ ดังรูปที่ 2.5



รูปที่ 2.5 ขนาดสัญญาณพัลส์ในการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมความเร็วมอเตอร์ สามารถทำได้โดยใช้วิธีการลดระดับแรงดัน ซึ่งจะทำให้มอเตอร์หมุนช้าลงได้ และการใช้ความถี่ของสัญญาณไฟฟ้าเข้ามาควบคุมแบบพัลส์วิดท์มอดูเลชั่น (PWM : Pulse Width Modulation) ดังรูปที่ 2.6



รูปที่ 2.6 สัญญาณของพัลส์แบบ PWM : Pulse Width Modulation

พัลส์วิดท์มอดูเลชั่น เป็นการใช้ความกว้างของสัญญาณพัลส์ ในการกำหนดแรงดันเฉลี่ย ความถี่ของสัญญาณพัลส์จะไม่มีการเปลี่ยนแปลง โดยจะใช้ความกว้างของช่วงสัญญาณในการควบคุมแรงดัน ดิวตี้ไซเกิล (Duty Cycle) เป็นค่าของคาบช่วงเวลาทำงานของไซเกิล โดยคิดเป็นเปอร์เซ็นต์ หากแรงดันในการทำงานอยู่ที่ 5 โวลต์ ดิวตี้ไซเกิล 50 เปอร์เซ็นต์ จะมีแรงดันไฟฟ้าเฉลี่ยที่ 2.5 โวลต์ ดิวตี้ไซเกิล 75 เปอร์เซ็นต์ จะมีแรงดันไฟฟ้าเฉลี่ยที่ 3.75 โวลต์

2.7 โปรแกรมที่เกี่ยวข้อง

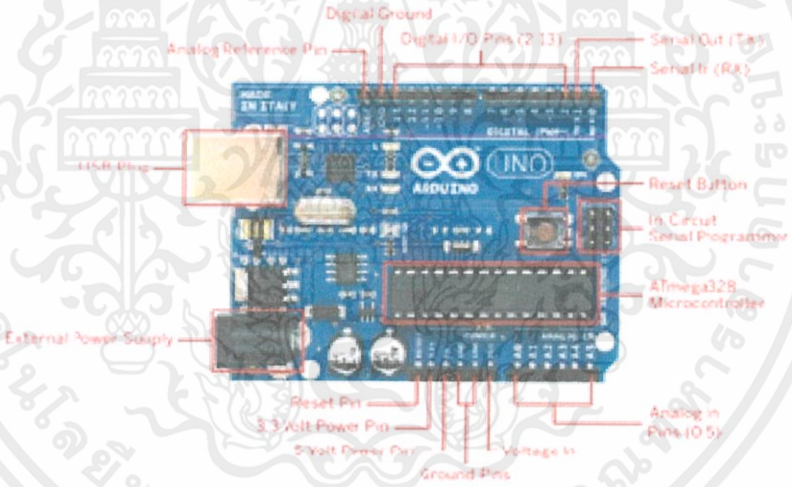
2.7.1 Arduino

Arduino เป็นภาษาอิตาลี เป็นชื่อที่สำหรับใช้เพื่อโครงการพัฒนาไมโครคอนโทรลเลอร์ตระกูลของ AVR (Alf (Egil Bogen) and Vegard (Wollan)'sRISC Processor) แบบรหัสเปิด (Open Source) ที่ได้รับการปรับปรุงมาจากอีกโครงการ “Wiring” ที่ใช้ ATmega128 ที่มีหน่วยความจุ และ I/O ค่อนข้างมาก และตัวถังของ ATmega128 เป็นอุปสรรคกับผู้เริ่มใช้งานในเบื้องต้นจึงไม่เป็นที่นิยม จากนั้นทีมงานของ Arduino ได้นำมาพัฒนาต่อยอดโดยให้สามารถใช้งาน AVR

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดเล็กได้ จึงทำให้วงจรของบอร์ดมีขนาดที่เล็กลง อีกทั้งยังใช้อุปกรณ์จำนวนน้อยอีกด้วย ทำให้ง่ายในการต่อวงจรและประหยัดในการสร้างบอร์ดมาก จึงทำให้ Arduino ได้รับความนิยมอย่างมาก

Arduino เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ AVR ขนาดเล็กเป็นตัวประมวลผล เหมาะสำหรับการศึกษาเรียนรู้ไมโครคอนโทรลเลอร์ และการนำไปประยุกต์ใช้งานในการควบคุมอุปกรณ์ทั้งอินพุตและเอาต์พุตต่างๆ ได้ โปรแกรมภาษาของ Arduino จะใช้รูปแบบของภาษา C++ Arduino นี้เป็นบอร์ดไมโครคอนโทรลเลอร์ ซึ่ง Arduino ที่มีการใช้งานจะเป็น Arduino Model : Arduino UNO R3 ใช้ชิพ ATmega328 รันที่ความถี่ 16 MHz หน่วยความจำแฟลช 32 KB แรม 2 KB บอร์ดใช้ไฟเลี้ยง 7 V ถึง 12 V มีระดับแรงดันไฟฟ้าในการทำงานและขาสัญญาณอยู่ที่ 5 V (TTL) มี Digital Input/Output 14 ขา (เป็น PWM ได้ 6 ขา) มี Analog Input 6 ขา, Serial UART 1 ชุด, I2C 1 ชุด และ SPI 1 ชุด เขียนโปรแกรมบนซอฟท์แวร์ Arduino IDE และโปรแกรมผ่านพอร์ต USB ดังรูปที่ 2.6



รูปที่ 2.6 โครงสร้างของ Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและโปรแกรมควบคุม

3.1 อุปกรณ์สำหรับการทำแขนกล

3.1.1 ไม้บัลซ่า

ใช้ไม้บัลซ่าในการทำโครงสร้างของแขนกล เพราะไม้บัลซ่ามีน้ำหนักเบา ราคาถูกและมีความแข็งแรงในระดับหนึ่ง และไม้บัลซ่าสามารถสร้างภาระให้เซอร์โวได้น้อย ดังรูปที่ 3.1

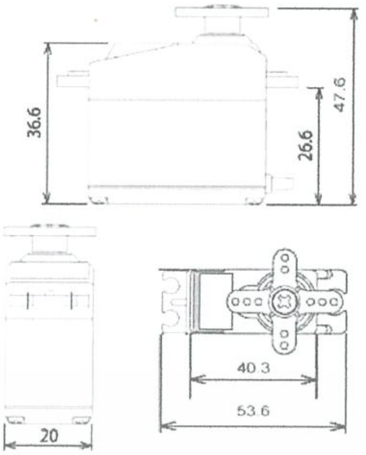
รูปที่ 3.1 ไม้บัลซ่า

3.1.2 เซอร์โวมอเตอร์

เลือกหาเซอร์โวมอเตอร์เพื่อใช้ในการทำแขนกล เพื่อที่จะมาออกแบบวงจรที่สามารถขับมอเตอร์ได้จะต้องเลือกมอเตอร์ที่บอกคุณสมบัติต่างๆ อย่างเช่น แรงบิดสูงสุด, ความเร็ว และแรงดันไฟฟ้าที่ใช้เป็นต้น ดังรูปที่ 3.2 ถึงรูปที่ 3.4

รูปที่ 3.2 เซอร์โวมอเตอร์ที่ใช้ในการทำแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ขนาดของเซอร์โวมอเตอร์ที่ใช้ในการทำแขนกล

TowerPro MG945 Servo

Specifications

Modulation:	Digital
Torque:	**4.8V:** 138.87 oz-in (10.00 kg-cm) **6.0V:** 166.65 oz-in (12.00 kg-cm)
Speed:	**4.8V:** 0.23 sec/60° **6.0V:** 0.20 sec/60°
Weight:	1.94 oz (55.0 g)
Dimensions:	**Length:** 1.60 in (40.7 mm) **Width:** 0.78 in (19.7 mm) **Height:** 1.69 in (42.9 mm)
Motor Type:	(add)
Gear Type:	Metal
Rotation/Support:	Dual Bearings
Rotational Range:	(add)
Pulse Cycle:	1 ms
Pulse Width:	(add)
Connector Type:	JR

รูปที่ 3.4 คุณสมบัติของเซอร์โวมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดักแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซอร์โวมอเตอร์ที่ใช้สำหรับเป็นข้อต่อของแขนกลนั้น มีการรับค่าสัญญาณแบบพัลส์ มีการทำงานที่ตำแหน่งของความดันไฟฟ้า 2 ค่าคือ ที่ความดันไฟฟ้า 4.8 โวลต์ จะสร้างแรงบิดและความเร็วได้ 10 kg-cm และ 0.23 sec/60° ตามลำดับ และที่ความดันไฟฟ้า 6 โวลต์ จะสร้างแรงบิดและความเร็วได้ 12 kg-cm และ 0.20 sec/60° ตามลำดับ

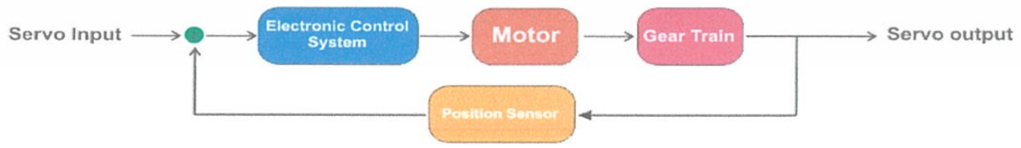
3.2 การออกแบบแขนกล

เมื่อได้เซอร์โวมอเตอร์ที่จะใช้ในการทดลองแล้ว จากนั้นนำเซอร์โวมอเตอร์ที่ได้ไปติดตั้งในข้อต่อแต่ละข้อของแขนกลเพื่อให้แขนกลขยับได้ตามที่ออกแบบไว้ โดยออกแบบให้แขนกลมีการเคลื่อนที่เหมือนแขนมนุษย์ โดยส่วนที่หนึ่งติดตั้งส่วนฐานไว้ในแนวราบ เพื่อให้แขนแกว่งได้เหมือนหัวไหล่ ส่วนที่สองติดตั้งไว้ในแนวดิ่ง เพื่อให้แขนขยับขึ้นลงได้เหมือนหัวไหล่ ส่วนที่สามติดตั้งไว้แนวตั้งเพื่อให้แขนกลขยับขึ้นลงได้เหมือนข้อศอก ส่วนที่สี่ติดตั้งแขนกลไว้แนวราบ เพื่อให้แขนกลหมุนได้เหมือนแขนมนุษย์ ส่วนที่ห้าติดตั้งไว้แนวตั้ง เพื่อให้แขนกลขยับขึ้นลงได้เหมือนข้อมือมนุษย์

3.3 คำสั่งในการควบคุมการหมุนของมอเตอร์

3.3.1 หลักการทำงานของ RC Servo Motor

เมื่อจ่ายสัญญาณพัลส์เข้ามายัง RC Servo Motor ส่วนวงจรควบคุม (Electronic Control System) ภายใน Servo จะทำการอ่านและประมวลผลค่าความกว้างของสัญญาณพัลส์ที่ส่งเข้ามาเพื่อแปลค่าเป็นตำแหน่งองศาที่ต้องการให้ Motor หมุนเคลื่อนที่ไปยังตำแหน่งนั้น แล้วส่งคำสั่งไปทำการควบคุมให้ Motor หมุนไปยังตำแหน่งที่ต้องการ โดยมี Position Sensor เป็นตัวเซ็นเซอร์คอยวัดค่ามุมที่ Motor กำลังหมุน เป็น Feedback กลับมาให้วงจรควบคุมเปรียบเทียบกับค่าอินพุตเพื่อควบคุมให้ได้ตำแหน่งที่ต้องการอย่างถูกต้องแม่นยำ ดังรูปที่ 3.5



รูปที่ 3.5 การทำงานของเซอร์โวมอเตอร์

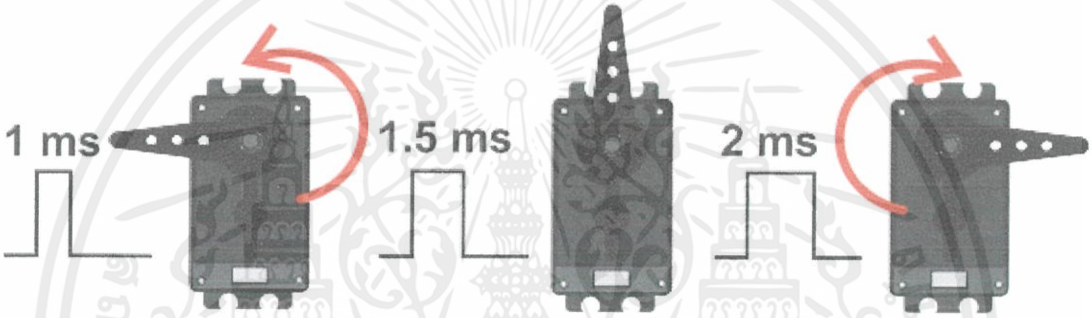
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 สัญญาณ RC ในรูปแบบ PWM

ตัว RC Servo Motor ออกแบบมาใช้สำหรับรับคำสั่งจาก Remote Control ที่ใช้ควบคุมของเล่นด้วยสัญญาณวิทยุต่างๆ เช่น เครื่องบินบังคับ รถถังบังคับ เรือบังคับ เป็นต้น ซึ่ง Remote จำพวกนี้ที่ภาครับจะแปลงความถี่วิทยุออกมาในรูปแบบสัญญาณ PWM (Pulse Width Modulation)

มุมหรือองศาจะขึ้นอยู่กับความกว้างของสัญญาณพัลส์ ซึ่งโดยส่วนมากความกว้างของพัลส์ที่ใช้ใน RC Servo Motor จะอยู่ในช่วง 1-2 ms หรือ 0.5-2.5 ms

ยกตัวอย่างเช่น หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1 ms ตัว Servo Motor จะหมุนไปทางด้ายซ้ายจนสุด ในทางกลับกันหากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 2 ms ตัว Servo Motor จะหมุนไปยังตำแหน่งขวาสุด แต่หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1.5 ms ตัว Servo Motor ก็จะหมุนมาอยู่ที่ตำแหน่งตรงกลางพอดี ดังรูปที่ 3.6



รูปที่ 3.6 การหมุนของเซอร์โวมอเตอร์

ดังนั้นสามารถกำหนดองศาการหมุนของ RC Servo Motor ได้โดยการเทียบค่า เช่น RC Servo Motor สามารถหมุนได้ 180 องศา โดยที่ 0 องศาใช้ความกว้างพัลส์เท่ากับ 1000 us ที่ 180 องศาความกว้างพัลส์เท่ากับ 2000 us เพราะฉะนั้นค่าที่เปลี่ยนไป 1 องศาจะใช้ความกว้างพัลส์ต่างกัน (2000-1000)/180 เท่ากับ 5.55 us

จากการหาค่าความกว้างพัลส์ที่มุม 1 องศาข้างต้น หากต้องกำหนดให้ RC Servo Motor หมุนไปที่มุม 45 องศา จะหาค่าพัลส์ที่ต้องการได้จาก 5.55 x 45 เท่ากับ 249.75 us แต่ที่มุม 0 องศาเริ่มที่ความกว้างพัลส์ 1ms หรือ 1000 us เพราะฉะนั้นความกว้างพัลส์ที่ใช้กำหนดให้ RC Servo Motor หมุนไปที่ 45 องศา คือ 1000 + 249.75 เท่ากับประมาณ 1250 us

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดักแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึง การทดลองการเคลื่อนที่ของโครงสร้างแขนกลเพื่อที่จะให้ได้ตำแหน่งการเคลื่อนที่ที่สามารถขยับและทำงานได้ตามวัตถุประสงค์ที่จะประยุกต์ใช้ซึ่งก็คือ การพ่นสี โดยจำเป็นต้องมีการออกแบบให้โครงสร้างมีความคล่องตัวในการเคลื่อนที่และมีน้ำหนักเบาที่สุด รวมไปถึงการทดลองเขียนโค้ดในโปรแกรม Arduino เพื่อนำโค้ดที่ได้มาบรรจุใส่ในบอร์ด Arduino เพื่อใช้ควบคุม Servo Motor ให้มีการทำงานของแขนกลในรูปแบบอัตโนมัติ

4.1 การทดลอง

4.1.1 การใช้ Arduino ในการสั่งให้เซอร์โวมอเตอร์ทำงาน

เพื่อศึกษาการเคลื่อนที่ของเซอร์โวมอเตอร์ในมุมต่างๆ

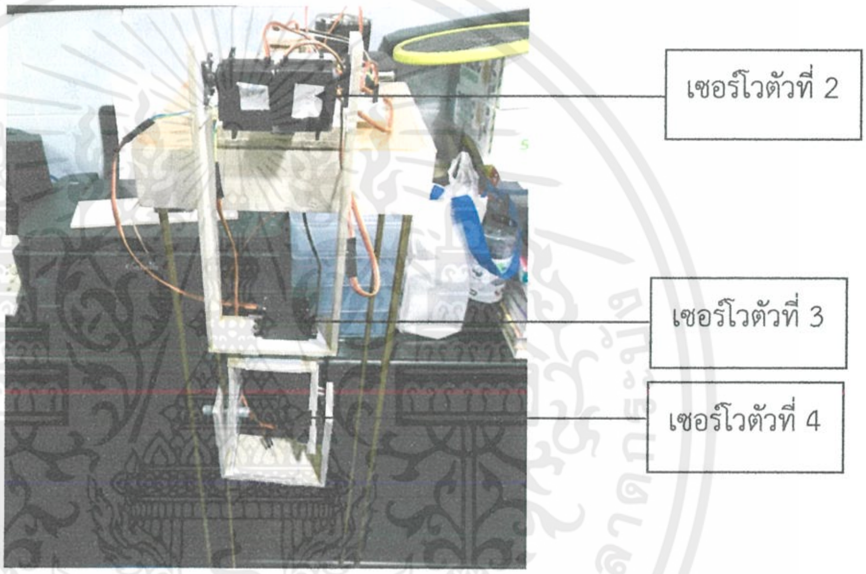
โค้ดตัวอย่างการควบคุมตำแหน่ง RC Servo Motor

```
#include <Servo.h>
 Servo myservo;
 void setup()
  {
    myservo.attach(9);
  }
  void loop()
  {      myservo.write(0);
         delay(1000);
         myservo.write(90);
         delay(1000);
         myservo.write(180);
         delay(1000);  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การหามุมการเคลื่อนที่ของเซอร์โวมอเตอร์

ในขั้นตอนนี้ จะกำหนดระยะห่างของวัตถุที่ต้องการพ่นสีกับปลายมือของแขนกล เพื่อนำค่าที่ได้มาคำนวณใน Inverse Kinematics เพื่อหาค่ามุมของเซอร์โวในการเคลื่อนไปยังตำแหน่งต่างๆ จากการคำนวณจะได้ว่า เซอร์โวตัวที่ 1 ต้องเคลื่อนที่เป็นมุม ±90° เพื่อยกแขนกลไปทางด้านขวา และด้านซ้ายเพื่อพ่นสีทางด้านข้างของวัตถุทั้ง 2 ข้าง ต่อมาเซอร์โวตัวที่ 2 จะเคลื่อนที่เป็นมุม ±90° เพื่อยกแขนกลขึ้นและลง ถัดมาเซอร์โวตัวที่ 3 จะเคลื่อนที่เป็นมุม ±90° เพื่อยกแขนกลให้ตั้งฉากกับพื้น และสุดท้ายเซอร์โวตัวที่ 4 จะเคลื่อนที่เป็นมุม ±90° เพื่อใช้พ่นสีทางซ้ายและขวา โดยในการทดลองจะใช้หลอดไฟแอลอีดี ในการจำลองการพ่นสีแทนการใช้สเปรย์พ่นสี ดังรูปที่ 4.1 และรูปที่ 4.2



รูปที่ 4.1 ลักษณะของแขนกลด้านหน้า

รูปที่ 4.2 ลักษณะของแขนกลด้านข้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

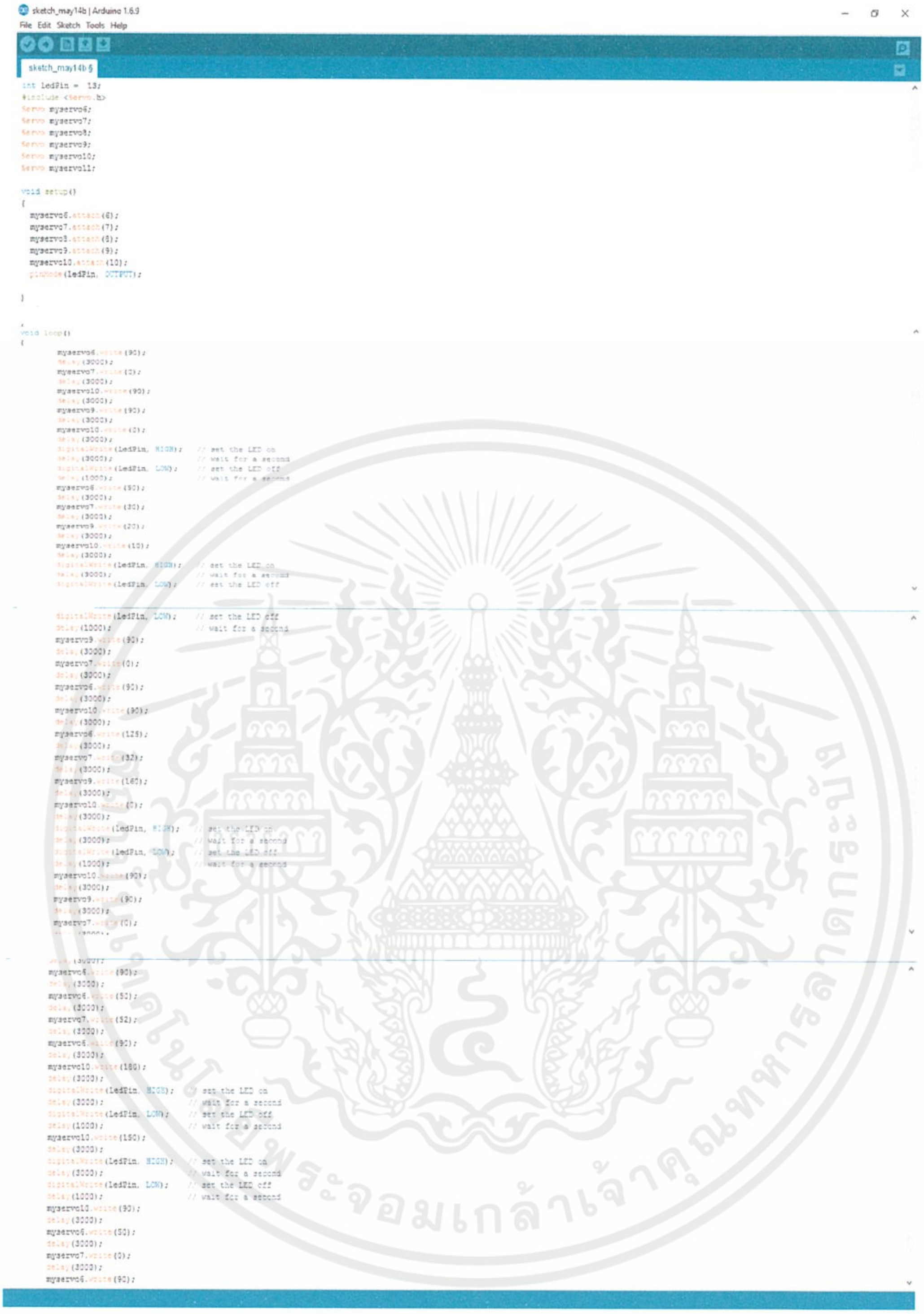
4.1.3 การทดลองสร้างโค้ดควบคุมด้วยโปรแกรม Arduino ในรูปแบบอัตโนมัติ

เริ่มต้นการทดลองด้วยการสร้าง Library ในแต่ละแบบที่จำเป็นต้องใช้เพราะโปรแกรมนี้สามารถสร้างโค้ดควบคุมและสั่งงานทาง Servo Motor ด้วยการบรรจุข้อมูลหรือโปรแกรมที่เขียนเข้าไปในบอร์ด Arduino และสามารถกำหนดค่าต่างๆ ในอุปกรณ์ได้อย่างละเอียด อีกทั้งตัวโปรแกรมยังสามารถวิเคราะห์สมการทางคณิตศาสตร์ และทำงานเป็นส่วนๆ ได้เหมือนสมองของมนุษย์ ทำการทดลองโดยใช้โปรแกรม Arduino โดยทำการสร้าง Library จากนั้นกำหนดค่าสัญญาณขาเข้าและขาออกที่โฟโต้บอร์ด และควบคุมให้เซอร์โวทำงานตามค่าของมุมที่คำนวณได้จากสมการ Inverse Kinematics

ตัวอย่างโค้ดที่ใช้ควบคุมเซอร์โวผ่าน Arduino

```
#include <Servo.h>
Servo myservo6 ;
Servo myservo7 ;
Servo myservo8 ;
Servo myservo9 ;
Servo myservo10 ;
Servo myservo11 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดักแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



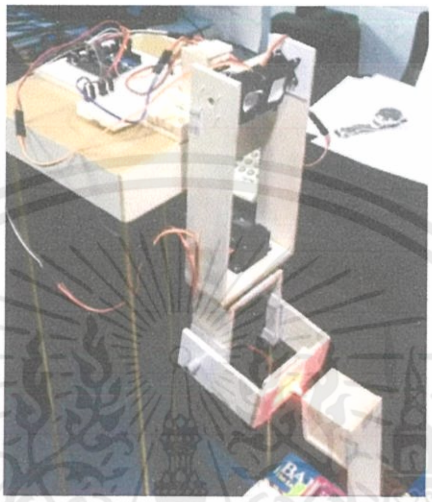

```
int ledPin =  13;
#include <Servo.h>
Servo myservo6;
Servo myservo7;
Servo myservo8;
Servo myservo9;
Servo myservo10;
Servo myservo11;

void setup()
{
  myservo6.attach(6);
  myservo7.attach(7);
  myservo8.attach(8);
  myservo9.attach(9);
  myservo10.attach(10);
  pinMode(ledPin, OUTPUT);

}

void loop()
{
        myservo6.write(90);
        delay(3000);
        myservo7.write(0);
        delay(3000);
        myservo10.write(90);
        delay(3000);
        myservo9.write(90);
        delay(3000);
        myservo10.write(0);
        delay(3000);
        digitalWrite(ledPin, HIGH);   // set the LED on
        delay(3000);                  // wait for a second
        digitalWrite(ledPin, LOW);    // set the LED off
        delay(1000);                  // wait for a second
        myservo6.write(90);
        delay(3000);
        myservo7.write(30);
        delay(3000);
        myservo9.write(20);
        delay(3000);
        myservo10.write(10);
        delay(3000);
        digitalWrite(ledPin, HIGH);   // set the LED on
        delay(3000);                  // wait for a second
        digitalWrite(ledPin, LOW);    // set the LED off

        digitalWrite(ledPin, LOW);    // set the LED off
        delay(1000);                  // wait for a second
        myservo9.write(90);
        delay(3000);
        myservo7.write(0);
        delay(3000);
        myservo6.write(90);
        delay(3000);
        myservo10.write(90);
        delay(3000);
        myservo6.write(125);
        delay(3000);
        myservo7.write(32);
        delay(3000);
        myservo9.write(160);
        delay(3000);
        myservo10.write(0);
        delay(3000);
        digitalWrite(ledPin, HIGH);   // set the LED on
        delay(3000);                  // wait for a second
        digitalWrite(ledPin, LOW);    // set the LED off
        delay(1000);                  // wait for a second
        myservo10.write(90);
        delay(3000);
        myservo9.write(90);
        delay(3000);
        myservo7.write(0);

        delay(3000);
        myservo6.write(90);
        delay(3000);
        myservo6.write(50);
        delay(3000);
        myservo7.write(52);
        delay(3000);
        myservo6.write(90);
        delay(3000);
        myservo10.write(180);
        delay(3000);
        digitalWrite(ledPin, HIGH);   // set the LED on
        delay(3000);                  // wait for a second
        digitalWrite(ledPin, LOW);    // set the LED off
        delay(1000);                  // wait for a second
        myservo10.write(150);
        delay(3000);
        digitalWrite(ledPin, HIGH);   // set the LED on
        delay(3000);                  // wait for a second
        digitalWrite(ledPin, LOW);    // set the LED off
        delay(1000);                  // wait for a second
        myservo10.write(90);
        delay(3000);
        myservo6.write(50);
        delay(3000);
        myservo7.write(0);
        delay(3000);
        myservo6.write(90);
```

รูปที่ 4.3 ออกแบบโค้ด Arduino ควบคุมให้เซอร์โวทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดลอง

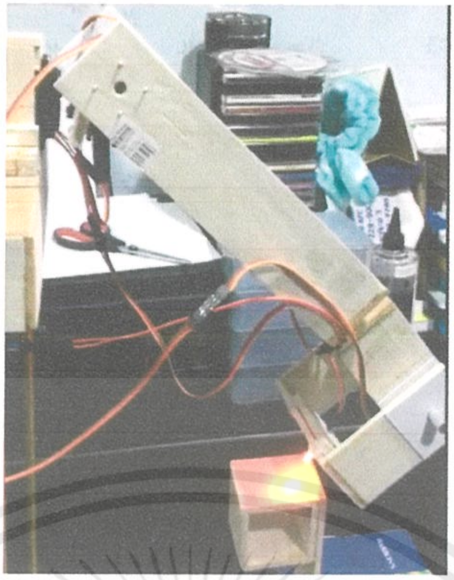
จากการทดลอง แขนกลสามารถพ่นสีได้อัตโนมัติ โดยจะเริ่มพ่นสีวัตถุจากตำแหน่งด้านหน้าของวัตถุก่อน แล้วจึงพ่นสีด้านข้างของวัตถุ จากนั้นจึงพ่นสีด้านบนของวัตถุ และสุดท้ายจะพ่นสีที่ด้านหลังของวัตถุ ดังรูปที่ 4.4 ถึงรูปที่ 4.7



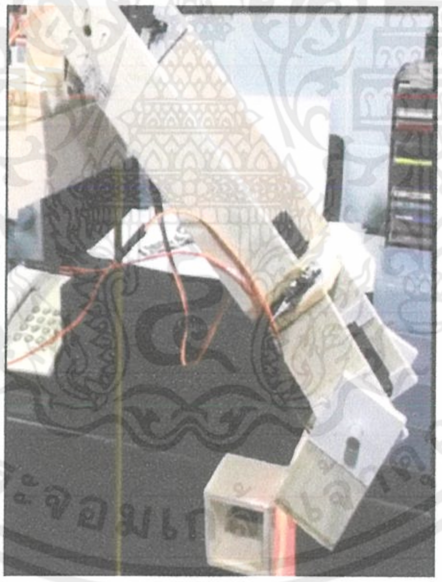
รูปที่ 4.4 การพ่นสีของแขนกลที่ด้านหน้าของวัตถุ

รูปที่ 4.5 การพ่นสีของแขนกลที่ด้านข้างของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 การพ่นสีของแขนกลที่ด้านบนของวัตถุ



รูปที่ 4.7 การพ่นสีของแขนกลที่ด้านหลังของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง

จากการทดลอง การทำงานของแขนกลของข้อต่อในแต่ละข้อต่อ และลักษณะการเคลื่อนไหวของข้อต่อในแต่ละช่วง โดยความยาวของแขนกลนี้มีโครงสร้างแขนของมนุษย์เป็นต้นแบบ และได้ใช้วัสดุที่มีน้ำหนักเบาในการทำโครงสร้างเพื่อลดภาระการทำงานของเซอร์โวมอเตอร์การควบคุมแขนกลด้วยโปรแกรม Arduino โดยใช้ Servo Motor เป็นอุปกรณ์ขับเคลื่อนโดยใช้ Feedback Control ทำให้สามารถควบคุมการเคลื่อนไหวของแขนกลได้เพื่อไปพ่นสีโดยจะใช้หลอดไฟ LED แทนการพ่นสีกับวัตถุจริงๆ โดยในการทดองกับวัตถุทรงลูกบาศก์จากทั้งหมด 10 ครั้งพบว่าผลลัพธ์เป็นไปตามต้องการคือพ่นสีได้ครบทุกด้านทั้ง 8 ครั้ง ซึ่งถือว่าแขนกลพ่นสีได้อย่างมีความอย่างแม่นยำค่อนข้างสูง และเป็นไปตามผลลัพธ์ที่ต้องการ

5.2 ปัญหาและอุปสรรคที่พบ

1. ปัญหา เนื่องจากการตัดวัสดุชิ้นงานทั้งหมดทำด้วยตัวเองจึงมีความคลาดเคลื่อน และมีความผิดพลาดอยู่บ้างเป็นต้นว่าตัดไม้บัลซ่าได้ไม่ตรงตามที่คำนวณ จากทฤษฎีจลศาสตร์ส่งผลให้ Servo Motor ไม่สามารถรับแรงได้ไหว

2. ปัญหา เนื่องจากสั่งซื้ออุปกรณ์ Servo Motor มาผิดชนิดส่งผลไม่สามารถใช้งานในส่วนของ Feedback Control ได้

3. ปัญหา เนื่องจากสายแพรที่ใช้ต่อกับโฟโต้บอร์ดมีการหลุดบ่อยส่งผลให้มีการชะงักในการเคลื่อนที่ของแขนกล

4. ปัญหา เนื่องจาก Servo Motor เป็น Servo Motor ชนิด High-torque ส่งผลให้เวลาแขนกลเคลื่อนไหวไม่สามารถคุม Torque ได้อยู่แขนกลจึงมีการเคลื่อนที่ที่ไม่ Smooth ในบางมุม

5. ปัญหา เนื่องจากโค้ด Arduino ที่เขียนไว้ในบางส่วนมี Bug ส่งผลให้เมื่อแขนกลทำงานเกิด Error ขึ้นหรือแขนกลค้างและหยุดทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดักแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางแก้ไข

1. ศึกษาการใช้ เลื่อยไฟฟ้า สว่านไฟฟ้า ไขควงไฟฟ้า และใช้อุปกรณ์ที่เกี่ยวกับเครื่องมือในการวัดทุกครั้งในการตรวจสอบเพื่อให้เกิดความแม่นยำ และจะมีการควบคุมที่มีความเสถียรมากยิ่งขึ้น

2. ศึกษาการใช้ Servo Motor และรุ่นของ Servo Motor ให้ดียิ่งขึ้นก่อนสั่งซื้อเพื่อมาใช้งานรวมถึงศึกษาการทำงานของ Motor ชนิดอื่นๆ ที่อาจนำมาใช้งานแทน Servo Motor ได้ให้มากยิ่งขึ้น

3. ในการแก้ไขปัญหาที่เกิดจากสายแพรหลุดขณะที่แขนกลทำงานได้ใช้ท่อหดเพื่อเก็บสายไฟให้เรียบร้อยลดปัญหาการที่สายแพรพันกันซึ่งส่งผลให้สายแพรหลุดจากโฟโต้บอร์ด

4. ในการแก้ไขปัญหาที่เกิดจากความไม่ Smooth ในการเคลื่อนที่ของแขนกลแก้ปัญหาด้วยการทำฐานที่มั่นคงให้มากยิ่งขึ้นและลดความเร็วในการเคลื่อนไหวของแขนกล

5. ในการแก้ไขปัญหาที่เกิดจากความผิดพลาดของโปรแกรม Arduino ส่งผลให้แขนกลค้างและหยุดทำงานได้ทำการแก้ปัญหาด้วยการ Run โค้ดที่เขียนไว้ใน Arduino ทีละส่วนๆ และหาส่วนที่ทำให้เกิดความผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] **“พื้นฐานของหุ่นยนต์”**, นายพงศ์แสน พิทักษ์วัชระ, 2557

[2] **“การวิเคราะห์และควบคุมหุ่นยนต์อุตสาหกรรม”**, นายวโรดม ตู้จินดา, 2559

[3] **“What is Arduino?”** [online.] Available:

https://www.arduino.cc

[4] **“การควบคุม servo motor ด้วย arduino”** [online.] Available:

http://thaieasyelec.com/article-wiki/review-product-article/

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดักแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โปรแกรมคำสั่งที่ใช้ในการเขียนให้ไมโครคอนโทลเลอร์ให้ทำงาน

โดยใช้โปรแกรม Atmel Studio 6.2

```
int ledPin =  13;
#include <Servo.h>
Servo myservo6;
Servo myservo7;
Servo myservo8;
Servo myservo9;
Servo myservo10;
Servo myservo11;

void setup()
{
  myservo6.attach(6);
  myservo7.attach(7);
  myservo8.attach(8);
  myservo9.attach(9);
  myservo10.attach(10);
  pinMode(ledPin, OUTPUT);

}
void loop()
{
        myservo6.write(90);
        delay(3000);
        myservo7.write(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดักแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
delay(3000);
myservo10.write(90);
delay(3000);
myservo9.write(90);
delay(3000);
myservo10.write(0);
delay(3000);
digitalWrite(ledPin, HIGH);   // set the LED on
delay(3000);                  // wait for a second
digitalWrite(ledPin, LOW);    // set the LED off
delay(1000);                  // wait for a second
myservo6.write(50);
delay(3000);
myservo7.write(30);
delay(3000);
myservo9.write(20);
delay(3000);
myservo10.write(10);
delay(3000);
digitalWrite(ledPin, HIGH);   // set the LED on
delay(3000);                  // wait for a second
digitalWrite(ledPin, LOW);    // set the LED off
delay(1000);                  // wait for a second
myservo9.write(90);
delay(3000);
myservo7.write(0);
delay(3000);
myservo6.write(90);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
delay(3000);
myservo10.write(90);
delay(3000);
myservo6.write(125);
delay(3000);
myservo7.write(32);
delay(3000);
myservo9.write(160);
delay(3000);
myservo10.write(0);
delay(3000);
digitalWrite(ledPin, HIGH);   // set the LED on
delay(3000);                  // wait for a second
digitalWrite(ledPin, LOW);    // set the LED off
delay(1000);                  // wait for a second
myservo10.write(90);
delay(3000);
myservo9.write(90);
delay(3000);
myservo7.write(0);
delay(3000);
myservo6.write(90);
delay(3000);
myservo6.write(50);
delay(3000);
myservo7.write(52);
delay(3000);
myservo6.write(90);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
delay(3000);
myservo10.write(180);
delay(3000);
digitalWrite(ledPin, HIGH);   // set the LED on
delay(3000);                  // wait for a second
digitalWrite(ledPin, LOW);    // set the LED off
delay(1000);                  // wait for a second
myservo10.write(150);
delay(3000);
digitalWrite(ledPin, HIGH);   // set the LED on
delay(3000);                  // wait for a second
digitalWrite(ledPin, LOW);    // set the LED off
delay(1000);                  // wait for a second
myservo10.write(90);
delay(3000);
myservo6.write(50);
delay(3000);
myservo7.write(0);
delay(3000);
myservo6.write(90);
delay(3000);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้