

แขนกลตัดโลหะด้วยพลาสมา

ROBOTIC ARM WITH PLASMA CUTTER

โดย

พิชชากร บุญจั้งหาร

สำหรับ นิมิตอนทอง

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมแมคคาทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2565

ภาควิชา วิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แขนกลตัดโลหะด้วยพลาสมา

ROBOTIC ARM WITH PLASMA CUTTER

ผู้จัดทำ นายพิชชากร บุญจันทาร รหัสประจำตัว 62010627

นายสหรัฐ ฉิมดอนทอง รหัสประจำตัว 62010923

ปริญญานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว



(ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	แขนกลตัดโลหะด้วยพลาสมา
นักศึกษา	นายพิชชากร บุญจันทาร รหัสประจำตัว 62010627
	นายสหรัถ ฉิมดอนทอง รหัสประจำตัว 62010923
ปริญญา	วิศวกรรมศาสตรบัณฑิต
ภาควิชา	วิศวกรรมแมคคาทรอนิกส์
ปีการศึกษา	2565
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้จัดทำขึ้นเพื่อศึกษาทฤษฎีและการออกแบบการสร้างแขนกล รวมถึงการเคลื่อนที่ของแขนกลเพื่อนำไปประยุกต์ในการใช้งานตัดโลหะด้วยพลาสมาและการทำงานอื่นในวงการอุตสาหกรรมอื่นๆ โดยใช้การจำลองการเคลื่อนที่ของแขนกลจากโปรแกรม MATLAB แล้วส่งโปรแกรมไปยังไมโครคอนโทรลเลอร์เพื่อนำค่าที่ได้จากตัวencoderมาคำนวณในโปรแกรมเพื่อปรับค่าการเคลื่อนที่ของแขนกลในระบบจริง

Thesis Title	ROBOTIC ARM WITH PLASMA CUTTER	
นักศึกษา	Pitchakorn Boonjungharn	Student ID 62010627
	Saharad Chimdonthong	Student ID 62010923
Degree	Bachelor of Engineering	
Program	Mechatronics Engineering	
Year	2023	
Thesis Advisor	Assistant Professor Sumit Phanaudomsab	

ABSTRACT

The following thesis will be made to study the theory and construction of the robotic arm and its components. In addition, it also compares plasma cutting and other operations to other industries as well. Simulate a model of a robotic arm from MATLAB and send the program to the microcontroller to test these values. Encode the encoder to calculate in the program. Adjustment of the robotic arm for reality

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์นี้ สามารถสำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับความช่วยเหลือสนับสนุนอย่างมากมายจาก ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์ อาจารย์ที่ปรึกษา ที่ให้คำแนะนำและความรู้ รวมถึงให้คำปรึกษาและแก้ไขปัญหาต่างจนสามารถแก้ไขอุปสรรคต่างๆให้ผ่านไปได้ด้วยดี

ขอขอบคุณคณาจารย์ทุกท่านที่ประสิทธิประสาทวิชาความรู้และคำแนะนำในด้านต่างๆ จนช่วยเหลือให้โครงการนี้ประสบความสำเร็จลุล่วงได้ดี

ขอขอบคุณบุพการีรวมถึงญาติผู้ใหญ่ที่คอยเป็นกำลังใจและแรงผลักดันในการจัดทำโครงการนี้

ขอบคุณเพื่อนๆ ที่ให้กำลังใจและคำแนะนำต่างๆ ตลอดจนอยู่เป็นเพื่อนกันจนสำเร็จการศึกษา

สุดท้ายนี้คณะผู้จัดทำขอขอบคุณที่ได้จากศึกษาค้นคว้าในครั้งนี้อย่างทุกท่านที่มีส่วนร่วมในโครงการ คณะผู้จัดทำมีความซาบซึ้งในความกรุณาอันดีจากทุกท่านที่ได้กล่าวมาและไม่ได้กล่าวไว้ และหากมีความผิดพลาดประการใดขออภัยไว้ ณ โอกาสนี้

พิชชากร บุญจันทาร

สำหรับ ฉิมดอนทอง

สารบัญ

	หน้า
บทคัดย่อ.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูป.....	VIII
สารบัญตาราง.....	XI
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมติฐานของการศึกษา.....	2
1.4 ขอบเขตการวิจัย.....	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	3
บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง.....	4
2.1 การทำงานของแขนกลหรือหุ่นยนต์ผู้ช่วย (Robotic Manipulator).....	4
2.1.1 ต้นกำเนิดการขับเคลื่อน.....	5
2.1.2 รูปแบบการเคลื่อนที่ในแขนกล.....	5
2.1.3 วิธีการควบคุมการเคลื่อนที่.....	11

สารบัญ(ต่อ)

	หน้า
2.2 จลนศาสตร์ทางตรงของแขนกล (Direct Kinematics).....	11
2.2.1 การdot productและcross product.....	12
2.2.2 การกำหนดพิกัดจุดในระนาบสามมิติ.....	14
2.2.3 พื้นฐานในการหมุนระนาบในระบบสามมิติ.....	14
2.2.4 การกำหนดจุดพร้อมระนาบในระบบสามมิติ.....	16
2.2.5 การเชื่อมต่อตำแหน่ง(Link coordinates).....	17
2.2.5.1 ค่าพารามิเตอร์จลนศาสตร์ (Kinematic Parameters).....	17
2.2.5.2 การจำลองโครงสร้างแขนกล.....	19
2.2.6 สมการแขนกล.....	20
2.2.6.1 สมการเมทริกซ์ของแขนกล.....	20
2.2.6.2 การสร้างสมการแขนกล.....	21
2.3 จลนศาสตร์ผกผันของแขนกล (Inverse Kinematics).....	22
2.3.1 ปัญหาจลนศาสตร์ผกผันของแขนกล.....	23
2.3.2 ตัวอย่างการหาจลนศาสตร์ผกผันของแขนกล.....	24
2.4 โปรแกรม MATLAB.....	28
2.5 โปรแกรม Arduino IDE.....	29

สารบัญ(ต่อ)

	หน้า
2.6 มอเตอร์ไฟฟ้ากระแสตรง.....	29
2.7 ไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033).....	30
2.8 ตัวขับมอเตอร์.....	32
2.9 เอ็นโคเดอร์ (Encoder).....	33
บทที่ 3 หลักการออกแบบและขั้นตอนการทำงาน.....	34
3.1 แบบจำลองแขนกลตัดโลหะด้วยพลาสมา.....	34
3.2 การต่อวงจรควบคุมแขนกลตัดโลหะด้วยพลาสมา.....	35
3.3 โปรแกรมการควบคุมแขนกลตัดโลหะด้วยพลาสมา.....	38
บทที่ 4 การทดลองและผลการทดลอง.....	40
4.1 การทดลองการแปลงค่าจาก Omron e6b2-cwz6c Encoder.....	40
4.2 การทดลองการเคลื่อนที่ของแขนกล.....	41
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง.....	44
5.1 สรุปผลการทดลอง.....	44
5.2 ปัญหาและอุปสรรคที่เกิดขึ้น.....	44
5.3 แนวทางในการแก้ไขปัญหาที่เกิดขึ้น.....	44
5.4 ข้อเสนอแนะในการค้นคว้าและพัฒนาต่อ.....	45

สารบัญ(ต่อ)

	หน้า
เอกสารอ้างอิง.....	46
ภาคผนวก.....	47
ภาคผนวก ก เอกสารคู่มือแบบจำลองแขนกล SCORBOT-ER 5Plus.....	48
ภาคผนวก ข เอกสารคู่มือไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board.....	52
ภาคผนวก ค เอกสารคู่มือ MDD10A Dual Channel 10A DC Motor Driver.....	58
ภาคผนวก ง เอกสารคู่มือ Omron e6b2-cwz6c Encoder.....	60
ประวัติผู้เขียน.....	63

สารบัญรูป

รูปที่	หน้า
รูปที่ 2-1 รูปตัวอย่างของแขนกล.....	4
รูปที่ 2-2 แขนกลคาร์ทีเซียน.....	7
รูปที่ 2-3 แขนกลพิกัดทรงกระบอก.....	8
รูปที่ 2-4 แขนกลพิกัดทรงกลม.....	9
รูปที่ 2-5 แขนกลสคาร่า(SCARA robot).....	10
รูปที่ 2-6 แขนกลชนิดข้อต่อแขน.....	11
รูปที่ 2-7 รูปจำลองของแขนกล.....	12
รูปที่ 2-8 เวกเตอร์ในมิติ n ที่ระนาบ R^n	12
รูปที่ 2-9 ระนาบเวกเตอร์ที่หมุนรอบเวกเตอร์ที่ตั้งฉากกับระนาบเวกเตอร์.....	14
รูปที่ 2-10 ระนาบเวกเตอร์ m_2, m_3	15
รูปที่ 2-11 ข้อต่อ k ทำมุม Θ และระยะห่าง dk กับข้อต่อ $k-1$	18
รูปที่ 2-12 เวกเตอร์ผลลัพธ์ของtoolเมื่อเทียบกับฐาน.....	22
รูปที่ 2-13 ค่าตัวแปร ข้อต่อ ของแขนกลในแต่ละ ข้อต่อ.....	23
รูปที่ 2-14 แขนกลสคาร่าสี่แกน โมเดลรุ่น Epson LS6-B SCARA Robot.....	24
รูปที่ 2-15 การเชื่อมต่อตำแหน่ง(Link coordinates)ของแขนกลสคาร่าสี่แกน โมเดลรุ่น Epson LS6-B SCARA Robot.....	25

สารบัญรูป(ต่อ)

รูปที่	หน้า
รูปที่ 2-16 หน้าต่างโปรแกรม MATLAB R2020b ที่ใช้ใน การเขียนโปรแกรมคำนวณสมการแขนกล.....	28
รูปที่ 2-17 หน้าต่างโปรแกรม Arduino IDE.....	29
รูปที่ 2-18 ลักษณะภายในมอเตอร์ไฟฟ้ากระแสตรง.....	30
รูปที่ 2-19 ไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033).....	31
รูปที่ 2-20 รูปตัวขับเคลื่อนมอเตอร์ MDD10A Dual Channel 10A DC Motor Driver.....	32
รูปที่ 2-21 รูปตัวเอ็นโคเดอร์ Omron e6b2-cwz6c.....	33
รูปที่ 3-1 โมเดลแขนกล SCORBOT-ER 5Plus.....	34
รูปที่ 3-2 แบบจำลองแขนกลตัดโลหะด้วยพลาสติก.....	35
รูปที่ 3-3 วงจรการควบคุมแขนกลตัดโลหะด้วยพลาสติก.....	36
รูปที่ 3-4 MDD10A Dual Channel 10A DC Motor Driver.....	37
รูปที่ 3-5 สายสัญญาณจาก Omron e6b2-cwz6c Encoder มาต่อเข้ากับ ไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033).....	37
รูปที่ 3-6 รูปโปรแกรมสมการแขนกลในโปรแกรมMATLAB.....	38

สารบัญรูป(ต่อ)

รูปที่	หน้า
รูปที่ 3-7 โปรแกรมเปรียบเทียบค่าจาก Omron e6b2-cwz6c Encoder กับค่าที่คำนวณจากโปรแกรมMATLAB.....	39
รูปที่ 4-1 โปรแกรมค่าการเปลี่ยนแปลงในตัว Omron e6b2-cwz6c Encoder.....	40
รูปที่ 4-2 การต่อวงจรเพื่อสังเกตค่าการเปลี่ยนแปลงในตัว Omron e6b2-cwz6c Encoder.....	40
รูปที่ 4-3 รูปการณ์เคลื่อนที่เป็นเส้นตรงของแกนกลในระบบจำลอง.....	41
รูปที่ 4-4 รูปการณ์เคลื่อนที่เป็นเส้นตรงของแกนกลในระบบจริง.....	42
รูปที่ 4-5 รูปการณ์เคลื่อนที่เป็นเส้นวงกลมของแกนกลในระบบจำลอง.....	42
รูปที่ 4-6 รูปการณ์เคลื่อนที่เป็นเส้นวงกลมของแกนกลในระบบจริง.....	43

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 2-1 ประเภท ข้อต่อ ในแกนกล.....	6
ตารางที่ 2-2 รูปแบบการเคลื่อนที่ของแกนกล.....	7
ตารางที่ 2-3 พารามิเตอร์ของแกนกล.....	18
ตารางที่ 2-4 ขั้นตอนในการแปลงระนาบ $k-1$ ไป k	20
ตารางที่ 2-5 ค่าพารามิเตอร์จลนศาสตร์.....	25



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันนี้ จะเริ่มเห็นการพัฒนาทางด้านหุ่นยนต์มากขึ้นและมีบทบาทสำคัญในอุตสาหกรรมหลายๆประเภท. โดยทั่วไปถ้ากล่าวถึงหุ่นยนต์คนทั่วไปจะคิดถึงเรื่องลักษณะหุ่นยนต์ที่มีความเหมือนมนุษย์สามารถคิดและทำงานได้ด้วยตัวเอง แต่ในความจริง คำว่า หุ่นยนต์ ตามนิยามของ Robotics Industrial Association นั้น หมายถึง เครื่องยนต์ที่ประกอบกันขึ้นมาเพื่อทำงานบางอย่างแทนมนุษย์สามารถควบคุมผ่านโปรแกรมคอมพิวเตอร์และกำหนดการทำงานของหุ่นยนต์ได้ ในปฏิญญาฉบับนี้จะขอยกตัวอย่างหุ่นยนต์ที่ถูกเรียกว่า แขนกล

ในยุคอุตสาหกรรม4.0 ได้มีการนำแขนกลมาทำงานในอุตสาหกรรมต่างๆเป็นจำนวนมาก ซึ่งอาจจะเป็นงานที่มนุษย์เราไม่สามารถทำได้ ยกตัวอย่างประเภทการทำงานเช่น การใช้แรงงานที่ต้องใช้แรงมหาศาล, งานที่ทำเป็นจำนวนมากๆหรืองานที่ใช้เวลาในการทำงานมากๆ เป็นต้น. โดยในประเทศไทยได้มีการนำแขนกลมาใช้ในงานอุตสาหกรรมกันมากขึ้น ซึ่งแขนกลส่วนใหญ่จะมีการสั่งซื้อจากต่างประเทศกันมาก ในงบต้นทุนนั้นจะมีราคาสูงทำให้ต้นทุนในอุตสาหกรรมนั้นสูงตามไปด้วย ดังนั้นเราจึงควรมีการค้นคว้าและขยายการศึกษาเกี่ยวกับแขนกลในประเทศให้มากขึ้น เพื่อให้มีการผลิตแขนกลเป็นของตัวเองภายในประเทศซึ่งจะช่วยลดต้นทุนในอุตสาหกรรมภายในประเทศ. และเนื่องจากในยุคอุตสาหกรรมปัจจุบันได้เริ่มมีการตัดโลหะกันมากขึ้น แต่ต้องการที่จะใช้ต้นทุนในการตัดที่น้อยและใช้งานได้หลากหลาย นอกจากนี้ในการตัดโลหะด้วยพลาสมานั้นหากให้มนุษย์ทำงาน อาจทำให้เกิดความผิดพลาดในการทำงานซึ่งเกิดจากคน(Human error)ได้ จึงเลือกที่จะใช้ แขนกลตัดโลหะด้วยพลาสมาเป็นหัวข้อในการทำโครงงานครั้งนี้ เพื่อที่จะศึกษาพื้นฐานหลักการในการทำงานของแขนกลโดยให้แขนกลควบคุมการตัดโลหะด้วยพลาสมาและสามารถตัดโลหะได้อย่างอิสระ อีกทั้งยังสามารถนำการทำงานของแขนกลไปประยุกต์ใช้กับการทำงานอื่นๆได้รวมถึงเพิ่มความสามารถในการทำงานที่ซับซ้อนขึ้นต่อไปในอนาคต

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1. เพื่อศึกษาทฤษฎีในการทำงานของแขนกลที่ใช้ในอุตสาหกรรม
2. เพื่อศึกษาการออกแบบและดัดแปลงวัสดุอุปกรณ์ในการสร้างแขนกล
3. เพื่อออกแบบโปรแกรมในการควบคุมการทำงานแขนกลตัดโลหะด้วยพลาสมา
4. เพื่อเป็นต้นแบบในการนำไปประยุกต์ใช้กับการทำงานอื่นๆ

1.3 สมมุติฐานของการศึกษา

การทำโครงการนี้เริ่มจากการศึกษาการสร้างแขนกลจำลองจากทฤษฎีในการทำงานของแขนกล โดยใช้โปรแกรม MATLAB เพื่อจำลองการเคลื่อนที่ของแขนกลในระบบจำลองโดยให้แสดงออกมาในรูปแบบกราฟเชิงเส้น และยังสามารถออกแบบการควบคุมการเคลื่อนที่ตามการออกแบบในโปรแกรมได้

เมื่อออกแบบการจำลองการเคลื่อนที่ของแขนกลในระบบจำลองเรียบร้อยแล้ว จึงเริ่มออกแบบลักษณะแขนกล โดยจะใช้โปรแกรม Solid Work เป็นโปรแกรมในการสร้างโมเดลแขนกลขึ้นมาตามที่ต้องการเพื่อให้เห็นรูปการทำงานของแขนกล จากนั้นนำมาวัสดุอุปกรณ์ต่างๆมาประกอบกันเป็นแขนกลที่ได้ออกแบบไว้

เมื่อประกอบแขนกลเรียบร้อยแล้วจะทดสอบการทำงานเบื้องต้นโดยออกแบบการจำลองการเคลื่อนที่ของแขนกลในระบบจำลองอย่างง่าย แล้วนำมาใส่ในไมโครคอนโทรลเลอร์โดยผู้จัดทำเลือกใช้ ตัวไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development เพื่อให้ตัวไมโครคอนโทรลเลอร์นั้น นำค่าที่อ่านได้จากเอ็นโค้ดเดอร์แบบแกนหมุน(Rotary Encoder)ที่ติดตั้งไว้ที่แขนกลมาคำนวณเพื่อตั้งค่าการหมุนของมอเตอร์ในแขนกลให้ทำงาน เมื่อทำการอัปโหลดโปรแกรมจำลองการเคลื่อนที่ของแขนกลในระบบจำลองอย่างง่ายลงในตัวไมโครคอนโทรลเลอร์เพื่อให้แขนกลในระบบจริงสามารถเคลื่อนที่ตามแขนกลในระบบจำลอง

1.4 ขอบเขตการวิจัย

การสร้างโปรแกรมระบบจำลองโดยใช้โปรแกรมMATLAB สร้างสมการแขนกลเพื่อควบคุมในระบบจำลอง โดยออกแบบให้มีการเคลื่อนที่อย่างง่ายเพื่อเป็นการทดสอบว่าสามารถนำมาปรับใช้ให้แขนกลเคลื่อนที่ได้อย่างอิสระตามที่ออกแบบในโปรแกรมหรือไม่ และนำมาทดลองการเคลื่อนที่กับแขนกลในระบบจริง เพื่อสังเกตว่าการเคลื่อนที่ของแขนกลในระบบจริงสามารถเคลื่อนที่ตรงกับการเคลื่อนที่ของแขนกลในระบบจำลองหรือไม่

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เพื่อศึกษาทฤษฎีการเคลื่อนที่ของแขนกลในระบบจำลอง
2. เพื่อศึกษาวิธีการใช้โปรแกรมMATLAB
3. เพื่อศึกษาวิธีการออกแบบและสร้างสรรค์ในการสร้างแบบจำลองแขนกลจริง
4. เพื่อสร้างต้นแบบของแขนกลที่คิดค้นขึ้นด้วยตัวเอง

บทที่ 2

ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1 การทำงานของแขนกลหรือหุ่นยนต์ผู้ช่วย (Robotic Manipulator)



รูปที่ 2-1 รูปตัวอย่างของแขนกล

คำนิยามของคำว่า “หุ่นยนต์” สามารถสื่อออกมาได้หลายหลายความหมายตามที่ผู้คนเข้าใจ ขึ้นอยู่กับบริบทของมัน. ในเนื้อหานี้จะหมายถึงหุ่นยนต์ที่ใช้ในอุตสาหกรรมหรือที่เรียกว่า “หุ่นยนต์ผู้ช่วย (Robotic Manipulator)” หรือ “แขนกล” ตามตัวอย่างในรูปที่ 1-1. แขนกลนี้คืออุปกรณ์เชิงกลที่สามารถตั้งค่าโปรแกรมได้และทำงานได้หลากหลายช่วยในการเคลื่อนย้ายวัสดุ, ชิ้นส่วน, วัตถุ, หรือเครื่องมือโดยการเคลื่อนไหวที่ตั้งโปรแกรมไว้เพื่อทำงานต่างๆ. หุ่นยนต์ผู้ช่วย(Robotic Manipulator) สามารถเคลื่อนย้ายหรือจับต้องวัตถุโดยอัตโนมัติซึ่งขึ้นอยู่กับองศาอิสระ (degrees of freedom) หรือก็คือแกนในการหมุนซึ่งสัมพันธ์กับจำนวนมอเตอร์ที่ใช้ในการเคลื่อนไหวของแขนกล แขนกลในอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนใหญ่จะมี4-6แกนหมุน แขนกลที่เจอได้มากที่สุดโดยทั่วไปจะเป็นแขนกล6แกนเพราะมีการเคลื่อนไหวที่คล้ายคลึงกับแขนมนุษย์มากที่สุด แขนกลนี้สามารถมองเป็นเส้นที่เชื่อมต่อกันด้วยข้อต่อที่เคลื่อนที่ได้อิสระ. เส้นที่เชื่อมกัน(Link)จะมีลักษณะเลียนแบบโครงสร้างร่างกายมนุษย์ที่เป็น อก แขน ในขณะที่ข้อต่อ(ข้อต่อ)จะมีลักษณะเลียนแบบ หัวไหล่ ข้อศอก และข้อมือ. ที่ปลายของแขนกลเป็นเครื่องมือที่กำหนดหน้าที่ของตัวแขนกล(end-effector) เรียกส่วนนี้ว่า tool, gripperหรือhand.

แนวคิดในการทำงานของแขนกลจะแบ่งออกเป็นหัวข้อเพื่อช่วยในการจำแนกประเภทการทำงานของแขนกล ซึ่งใช้เกณฑ์ในการแยกหมวดหมู่ดังนี้

-ต้นกำลังการขับเคลื่อน

-รูปแบบการเคลื่อนที่ในแขนกล

-วิธีการควบคุมการเคลื่อนที่



2.1.1 ต้นกำลังการขับเคลื่อน

การทำงานของแขนกลนั้นจะต้องคำนึงถึงต้นกำลังในการทำให้ข้อต่อ(ข้อต่อ)ของแขนกลเคลื่อนไหว. ต้นกำลังที่นิยมใช้มี2แบบคือกำลังไฟฟ้าและระบบไฮดรอลิก. หน่วยงานส่วนใหญ่ในปัจจุบันใช้กำลังไฟฟ้าในรูปแบบของ DC servomotors และDC stepper motors. ในงานที่ใช้เป็นเวลานานต่อเนื่องเช่นการหลอมเหล็กหรือการประกอบชิ้นส่วนโดยอัตโนมัติ จะใช้ระบบไฮดรอลิกแทนซึ่งข้อเสียของการใช้ระบบไฮดรอลิกคือการทำความสะดวกได้ยาก. ทั้งสองแบบนี้จะใช้กับระบบนิวเมติกหรือเครื่องมือที่กำหนดหน้าที่ของตัวแขนกล โดยเฉพาะถ้าเป็นgripper(แท่งหนีบจับ)ที่มีการเคลื่อนไหวแบบง่าย.

2.1.2 รูปแบบการเคลื่อนที่ในแขนกล

เครื่องมือที่กำหนดหน้าที่ของตัวแขนกล(end-effector)หรือtoolของแขนกลโดยทั่วไปจะติดตั้งที่ปลายแขนของแขนกล. ในการทำงานของแขนกลนั้นจะมองให้อยู่ในรูปแบบสามมิติที่มองตำแหน่งต่างๆ จากฐานถึงปลายแขน. โดยจะกำหนดจุดที่มีการเปลี่ยนตำแหน่งว่า ข้อต่อ ซึ่งแบ่งออกเป็น 2 แบบตามตารางที่2-1

ตารางที่ 2-1 ประเภท ข้อต่อ ในแขนกล

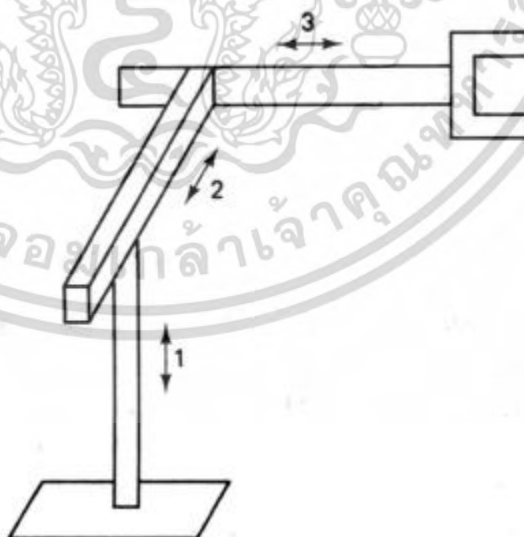
ประเภท ข้อต่อ	ตัวย่อ	สัญลักษณ์	คำอธิบาย
Revolute	R		การเคลื่อนที่แบบหมุนรอบแกนหมุน
Prismatic	P		การเคลื่อนที่แบบเชิงเส้นตามแกนหมุน

Revolute Joint คือ จุดที่แสดงการเคลื่อนที่แบบหมุน ซึ่งเป็นประเภทการเคลื่อนที่ของแขนกลที่พบเป็นส่วนใหญ่ อีกแบบหนึ่งคือ Prismatic Joint คือ จุดที่แสดงการเคลื่อนที่ในแนวเชิงเส้น. เมื่อนำมารวมกันจะได้รูปแบบการเคลื่อนที่ของแขนกลในรูปแบบต่างๆ ตารางที่ 2-2 ซึ่งจะแสดงให้เห็นว่าการเคลื่อนที่ของแขนกลนั้นมีความอิสระมากแค่ไหน ในรูปแบบการเคลื่อนที่ของแขนกลนั้น Prismatic Joint จะมีความง่ายในการควบคุมมากกว่า Revolute Joint รูปแบบการเคลื่อนที่แขนกล Articulated นั้นมีแกนหมุนทั้งหมดเป็นแบบ Revolute Joint ซึ่งจะทำให้การเคลื่อนที่ของแขนกลนั้นมีความซับซ้อนมากที่สุด

ตารางที่ 2-2 รูปแบบการเคลื่อนที่ของแขนกล

แขนกล	แกนหมุนที่ 1	แกนหมุนที่ 2	แกนหมุนที่ 3
แขนกลคาร์ทีเซียน	P	P	P
แขนกลพิกัดทรงกระบอก	R	P	P
แขนกลพิกัดทรงกลม	R	R	P
แขนกลสคาร์่า(SCARA robot)	R	R	P
แขนกลชนิดข้อต่อแขน	R	R	R

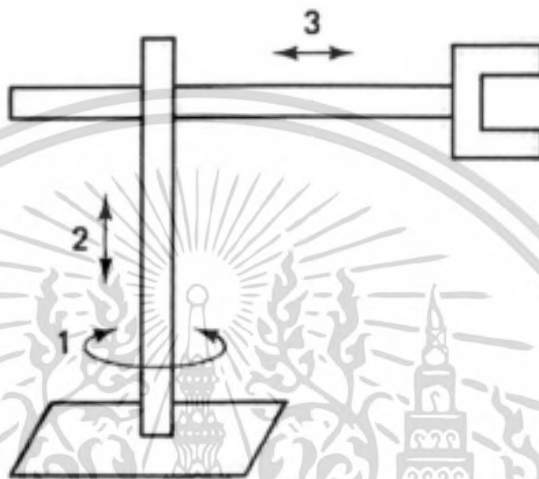
แขนกลที่มีการเคลื่อนที่ที่ง่ายที่สุดคือแบบคาร์ทีเซียน(Cartesian) รูปแบบการเคลื่อนที่ของแขนกลนี้จะมีข้อต่อแบบ Prismatic Joint ซึ่งมีลักษณะการเคลื่อนที่ตามรูปที่ 2-2 จะเห็นได้ว่าแขนมีการเคลื่อนที่ในแนวเชิงเส้นขึ้นลง, ซ้าย-ขวา, เข้า-ออก ซึ่งจะมีขอบเขตการทำงานในรูปทรงสี่เหลี่ยมผืนผ้า



รูปที่ 2-2 แขนกลคาร์ทีเซียน

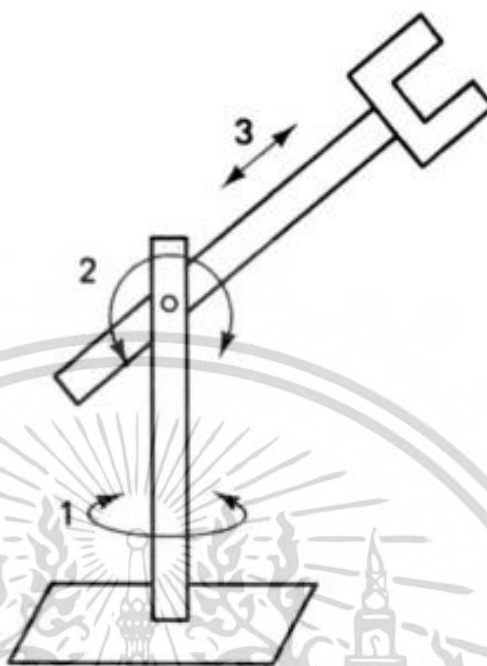
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเปลี่ยนแกนหมุนแรกจาก Prismatic Joint เป็น Revolute Joint จะมีรูปแบบการเคลื่อนที่แกนกลที่เรียกว่า พิกัดทรงกระบอก ตามรูปที่ 2-3 แกนหมุนแรกเป็น Revolute Joint จะหมุนรอบฐานของแกนกล เป็นรูปวงกลม Prismatic Joint ที่แกนหมุนอื่นจะเคลื่อนที่ขึ้น-ลง, เข้า-ออก รูปแบบการเคลื่อนที่นี้จะเริ่มมีการเคลื่อนที่ในรูปแบบเชิงมุม ซึ่งจะมีขอบเขตการทำงานในรูปทรงกระบอกในแนวตั้ง



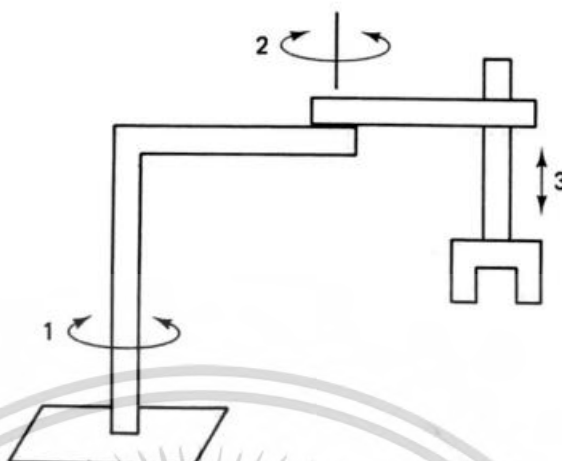
รูปที่ 2-3 แกนกลพิกัดทรงกระบอก

ถ้าเปลี่ยนแกนหมุนที่สองจาก Prismatic Joint เป็น Revolute Joint จะมีรูปแบบการเคลื่อนที่แกนกลที่เรียกว่า พิกัดทรงกลม ตามรูปที่ 2-4 แกนหมุนแรกเป็น Revolute Joint จะหมุนรอบฐานของแกนกลเป็นรูปวงกลม แกนหมุนที่สองเป็น Revolute Joint จะหมุนขึ้น-ลงตามแกนหมุนของแกนกลแนวนอน Prismatic Joint ที่แกนหมุนอื่นจะเคลื่อนที่เข้า-ออก รูปแบบการเคลื่อนที่นี้จะมีการเคลื่อนที่ในรูปทรงกลม ซึ่งจะมีขอบเขตการทำงานในรูปครึ่งทรงช่วงบนของแกนกล



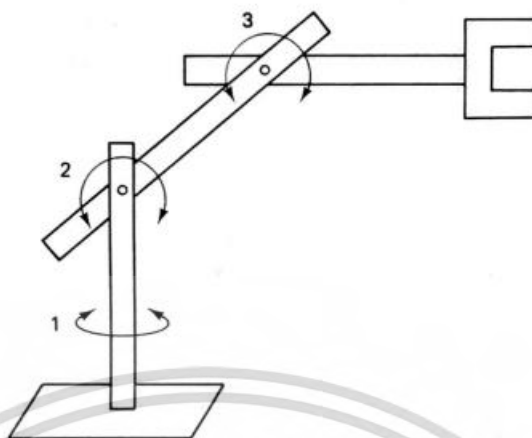
รูปที่ 2-4 แขนกลพิกัดทรงกลม

รูปแบบการเคลื่อนที่ต่อไปคือ สคาร่า(SCARA) ซึ่งจะมีความคล้ายคลึงกับแขนกลพิกัดทรงกลม แต่แกนหมุนของแขนกลนี้จะอยู่ในแนวตั้งตามรูปที่2-5 แกนหมุนแรกเป็น Revolute Joint จะหมุนรอบฐานของแขนกลเป็นรูปวงกลม แกนหมุนที่สองเป็น Revolute Joint จะหมุนซ้าย-ขวาตามแกนหมุนของแขนกลแนวตั้ง ซึ่งสองแกนหมุนนี้จะเคลื่อนที่ในระนาบแนวนอน Prismatic Joint ที่แกนหมุนอื่นจะเคลื่อนที่ขึ้น-ลงในแนวตั้ง รูปแบบการเคลื่อนที่นี้จะมีการเคลื่อนที่ที่ซับซ้อนมากขึ้น ซึ่งจะมีขอบเขตการทำงานที่ขึ้นอยู่กับการออกแบบ



รูปที่ 2-5 แขนกลสคาร์่า(SCARA robot)

ถ้าเปลี่ยนแกนหมุนที่สามจาก Prismatic Joint เป็น Revolute Joint จะมีการเคลื่อนที่ของแขนกลที่เรียกว่า แขนกลชนิดข้อต่อแขน ตามรูปที่ 2-6 แขนกลชนิดข้อต่อแขนมีความคล้ายคลึงกับแขนกลคาร์ทีเซียนที่ทุกแกนหมุนเหมือนกันแต่เป็น Revolute Joint แขนกลชนิดข้อต่อจะมีลักษณะใกล้เคียงกับโครงสร้างของแขนมนุษย์ แกนหมุนแรกเป็น Revolute Joint จะหมุนรอบฐานของแขนกลเป็นรูปวงกลม แกนหมุนที่สองและแกนหมุนที่สามเป็น Revolute Joint จะหมุนขึ้น-ลงตามแกนหมุนของแขนกลแนวนอน รูปแบบการเคลื่อนที่นี้จะมีการเคลื่อนที่ที่ซับซ้อนมากขึ้น โดยส่วนตัดมุมมองด้านข้างจะเป็นรูปพระจันทร์เสี้ยว



รูปที่ 2-6 แขนกลชนิดข้อต่อแขน

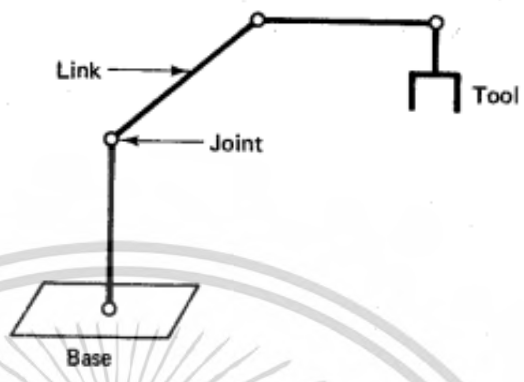
2.1.3 วิธีการควบคุมการเคลื่อนที่

วิธีการควบคุมการเคลื่อนที่ในการกำหนดตำแหน่งของ end-effector แบ่งเป็น 2 ประเภท ประเภทแรกคือการเคลื่อนที่จากจุดไปจุด ซึ่งจุดที่ว่า จะเคลื่อนที่เป็นลำดับจากจุดหนึ่งไปจุดหนึ่ง แต่ไม่สามารถเคลื่อนที่ระหว่างทางจากจุดไปจุด

ประเภทที่ 2 คือการเคลื่อนที่แบบต่อเนื่อง tool จะเคลื่อนที่ต่อเนื่องไปตามระนาบสามมิติและความเร็วตามเส้นทางที่กำหนด ซึ่งมีความซับซ้อนอย่างมาก ยกตัวอย่างที่มีการเคลื่อนที่แบบต่อเนื่องเช่น การพ่นสี, การเชื่อมอาร์ค

2.2 จลนศาสตร์ทางตรงของแขนกล (Direct Kinematics)

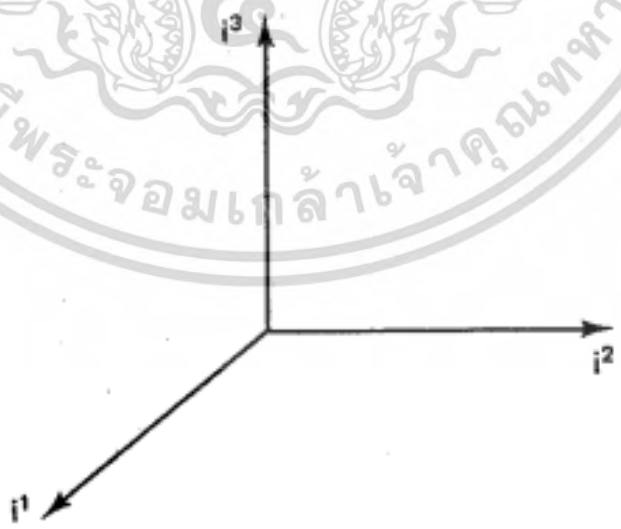
เราสามารถจำลองแขนกลให้เป็นเส้นที่เชื่อมต่อกันเรียกว่า link ซึ่งเชื่อมต่อกับอีก link โดย ข้อต่อ ตามรูปที่ 2-7 ปลายของแขนกลด้านหนึ่งจะยึดติดกับฐาน (base) อีกด้านหนึ่งจะเคลื่อนที่ได้อิสระซึ่งจะมีระนาบของ tool เราจำลองแขนกลเพื่อให้ควบคุมตำแหน่งของ tool ในระนาบสามมิติ เพื่อให้เคลื่อนที่ไปตามทิศทางที่เราต้องในพื้นที่การทำงานได้ ในการออกแบบโปรแกรมการเคลื่อนที่ของ tool เราต้องหาสมการที่สัมพันธ์กันระหว่างตัวแปรข้อต่อและตำแหน่งของ tool เรียกว่า จลนศาสตร์ทางตรงของแขนกล (Direct Kinematics) โดยการกำหนดค่าตัวแปรของ ข้อต่อ เพื่อหาค่าตำแหน่งของ tool



รูปที่ 2-7 รูปจำลองของแขนกล

2.2.1 การ dot product และ cross product

กำหนดให้เวกเตอร์ในมิติ n ที่ระนาบ R^n เป็นลูกศรที่พุ่งออกมาจากจุดกำเนิดตามรูปที่ 2-8 เมื่อ $n=3$ พิกัดของเวกเตอร์มีทิศทางเดียวกับหัวลูกศร



รูปที่ 2-8 เวกเตอร์ในมิติ n ที่ระนาบ R^n

การ dot product เวกเตอร์ x และ y ในระนาบ R^n จะได้ $x \cdot y$ และถูกนิยามเป็น

$$x \cdot y \triangleq \sum_{k=1}^n x_k y_k$$

เมื่อได้ค่า dot product แล้ว จะบ่งบอกถึงทิศทางของเวกเตอร์ทั้งสองที่กระทำต่อกันในระนาบเดียวกัน โดยใช้สมการนี้

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

ถ้าได้ค่าเท่ากับ 1 เวกเตอร์จะมีทิศทางเดียวกัน หากมีค่าเท่ากับ 0 เวกเตอร์จะมีทิศทางตั้งฉากกัน

หากมีค่าเท่ากับ -1 เวกเตอร์จะมีทิศทางตรงข้ามกัน

การ cross product เวกเตอร์ u และ v ในระนาบ R^3 จะได้ $w = u \times v$ และถูกนิยามเป็น

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

เมื่อได้ค่า cross product แล้ว จะบ่งบอกถึงเวกเตอร์ที่ตั้งฉากกันบนระนาบระหว่างเวกเตอร์ทั้งสองที่กระทำต่อกันในระนาบเดียวกันในทิศทางตามมือขวาโดยใช้สมการนี้

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

2.2.2 การกำหนดพิกัดจุดในระนาบสามมิติ

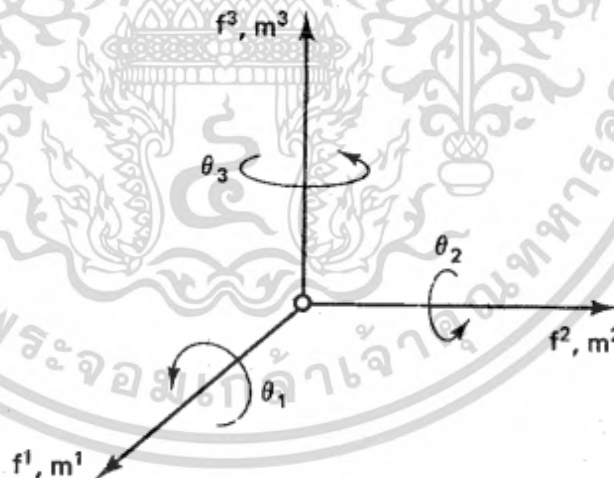
ในการสร้างสมการแกนกลขึ้นมาจะต้องกำหนดค่าตำแหน่งและระนาบของข้อต่อเป็นหลักซึ่งในการกำหนดค่านี้จะใช้ระบบเมทริกซ์ในการกำหนดค่าตำแหน่งและระนาบของข้อต่อต่างๆ โดยให้แต่ละระนาบเวกเตอร์คู่กับตำแหน่งในแต่ละระนาบจะได้

$$[p]^M = [p \cdot m^1, p \cdot m^2, p \cdot m^3]^T$$

ซึ่งหมายความว่าตำแหน่ง p ที่อยู่บนระนาบเวกเตอร์ m

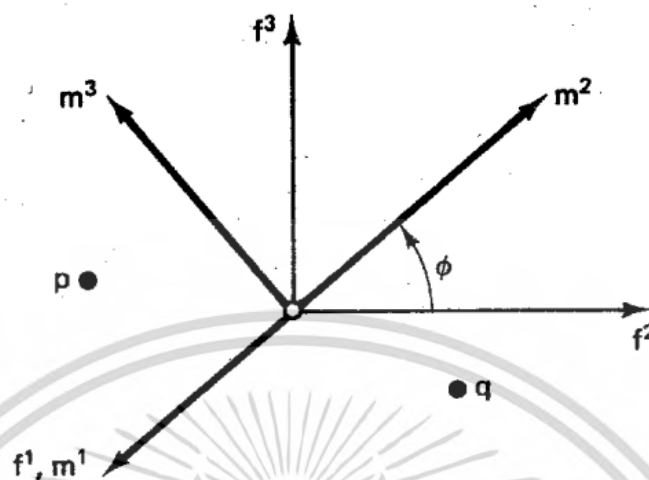
2.2.3 พื้นฐานในการหมุนระนาบในระบบสามมิติ

เมื่อกำหนดให้ตำแหน่งของจุด ๆ หนึ่งไม่มีการเคลื่อนที่และทำการหมุนสองเวกเตอร์ที่ทำระนาบต่อกันรอบเวกเตอร์อีกเวกเตอร์ที่ตั้งฉาก จะทำให้เกิดการหมุนได้ 3 แบบตามรูปที่ 2-9



รูปที่ 2-9 ระนาบเวกเตอร์ที่หมุนรอบเวกเตอร์ที่ตั้งฉากกับระนาบเวกเตอร์

สมมติถ้ากำหนดให้เวกเตอร์ f_1 อยู่กับที่ แล้วทำการหมุนระนาบเวกเตอร์ f_2, f_3 กลายเป็นระนาบเวกเตอร์ m_2, m_3 ตามกฎมือขวาทำมุม ϕ จะได้ตามรูปที่ 2-10

รูปที่ 2-10 ระนาบเวกเตอร์ m_2, m_3

เมื่อเราทำการหมุนระนาบเวกเตอร์ f_2, f_3 แล้วจะพบว่า เวกเตอร์ $f_1 = m_1$ จากนั้นเราทำการ dot product เวกเตอร์เทียบกันทั้งหมดให้อยู่ในรูปของเมทริกซ์จะได้

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

ซึ่งเมื่อทำการ dot product เวกเตอร์เทียบกันทั้งหมดแล้วจะได้สมการระนาบเวกเตอร์นี้

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

หากเวกเตอร์ใดๆจากสมการข้างต้นทำมุมกับ ϕ จะได้สมการดังต่อไปนี้โดยขึ้นอยู่กับว่าหมุนรอบแกนเวกเตอร์ไหน

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

หมุนรอบแกนเวกเตอร์ f1

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

หมุนรอบแกนเวกเตอร์ f2

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

หมุนรอบแกนเวกเตอร์ f3

2.2.4 การกำหนดจุดพร้อมระนาบในระบบสามมิติ

จากหัวข้อเรื่องการกำหนดพิกัดจุดในระนาบสามมิติและพื้นฐานในการหมุนระนาบในระบบสามมิติ ทำให้เราสามารถจำลองตำแหน่งของจุด ๆ หนึ่งซึ่งมีระนาบเป็นของตัวเองได้ โดยจัดให้อยู่ในรูปเมทริกซ์ 4*4 ดังนี้

$$T \triangleq \begin{bmatrix} \text{rotation} & \text{translation} \\ R & p \\ \text{perspective} & \text{scale} \\ \eta^T & \sigma \end{bmatrix}$$

R คือ ระบายของจุดเทียบกับจุดอ้างอิงก่อนหน้า

P คือ ตำแหน่งของจุดเทียบกับจุดอ้างอิงก่อนหน้า

η^T คือ ค่ามุมมองในหลายมิติ ซึ่งในโครงงานนี้ใช้ค่าเท่ากับ 0

σ คือ ตัวประกอบสเกลในระบบสามมิติ ซึ่งส่วนใหญ่กำหนดให้เท่ากับ 1

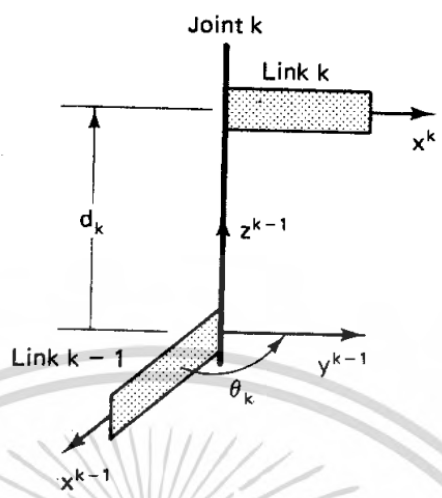
จากนั้นเราก็จะสามารถกำหนดลักษณะการเคลื่อนที่ของจุด ๆ หนึ่งในระบบสามมิติได้โดยใช้วิธีทางคณิตศาสตร์ในเรื่องเมทริกซ์มาใช้คำนวณหาตำแหน่งและระบายของจุด ๆ นั้น

2.2.5 การเชื่อมต่อตำแหน่ง(Link coordinates)

การจำลองแขนกลในระบบสามมิติ สามารถมองแขนกลให้อยู่ในรูปของเส้นที่เชื่อมกันระหว่างจุดที่เป็น ข้อต่อ เพื่อกำหนดตำแหน่งและระบายของข้อต่อต่างๆอย่างเป็นระบบ

2.2.5.1 ค่าพารามิเตอร์จลนศาสตร์(Kinematic Parameters)

ความสัมพันธ์ตำแหน่งและระบายของlinkทั้งสองสามารถบอกโดยใช้พารามิเตอร์ร่วมสองตัวตามรูปที่ 2-10



รูปที่ 2-11 ข้อต่อ k ทำมุม Θ และระยะห่าง d_k กับข้อต่อ k-1

ข้อต่อ k เชื่อมต่อจาก Link k-1 ไป Link k พารามิเตอร์ ข้อต่อ k จะคำนึงถึงแกน z^{k-1} ซึ่งสอดคล้องกับแกนหมุน ข้อต่อ k พารามิเตอร์ตัวแรกคือ Θ หรือ ข้อต่อ angle มันคือการหมุนรอบแกน z^{k-1} เพื่อให้แกน x^{k-1} ขนานกับแกน x^k พารามิเตอร์ตัวที่สองคือ d หรือ ข้อต่อ distance มันคือการเคลื่อนที่ไปตามแกน z^{k-1} เพื่อให้แกน x^{k-1} ตัดกับแกน x^k ซึ่งค่าพารามิเตอร์นี้ จะมีกรณีค่าหนึ่งคงที่และอีกค่าหนึ่งปรับค่าได้ พารามิเตอร์ของข้อต่อขึ้นอยู่กับชนิดของข้อต่อ จากตารางที่ 2-3 ซึ่งในการเขียนพารามิเตอร์จะต้องเขียนไล่ลำดับตามนี้ ข้อต่อ angle, ข้อต่อ distance, Link length, Link twist angle

ตารางที่ 2-3 พารามิเตอร์ของแขนกล

พารามิเตอร์แขน	สัญลักษณ์	Revolute ข้อต่อ(R)	Prismatic Joint(P)
ข้อต่อ angle	Θ	ปรับค่าได้	คงที่
ข้อต่อ distance	d	คงที่	ปรับค่าได้
Link length	a	คงที่	คงที่
Link twist angle	α	คงที่	คงที่

2.2.5.2 การจำลองโครงสร้างแขนกล

เราจะใช้วิธีการจำลองโครงสร้างแขนกลของ DenavitและHartenberg ซึ่งมีวิธีการดังนี้

1. นับ ข้อต่อ จาก1ถึงn โดยเริ่มจากฐานไปยังtool
2. กำหนดระนาบเริ่มต้นโดยใช้มือขวาหาฐานของแขนกล เพื่อให้แกน z^0 ตรงกับแกนของข้อต่อ 1ให้ $k=1$
3. ปรับให้ z^k ตรงกับแกนของข้อต่อ $k+1$
4. หาจุดกำเนิดของ L_k ที่เป็นจุดตัดของ z^k กับ z^{k-1} ถ้าไม่มี ใช้จุดตัดของ z^k กับค่าที่อยู่ระหว่าง z^k กับ z^{k-1}
5. เลือกค่า x^k ที่ตั้งฉาก z^k และ z^{k-1} ถ้าทั้งสองแกนขนานกัน ให้เลือกค่า x^k ที่ห่างจากแกน z^{k-1}
6. เลือกค่า y^k โดยใช้กฎมือขวาหาระนาบ L_k
7. ให้ $k=k+1$ ถ้า $k<n$. ให้ทำข้อที่3ใหม่ ถ้าไม่ใช่ไปข้อต่อไป
8. กำหนดจุดกำเนิดระนาบ L_k ปรับแกน z^k เป็นเวกเตอร์เข้าใกล้ z^k เป็นเวกเตอร์เลื่อน z^k เป็นเวกเตอร์toolให้ $k=1$
9. หาจุดตัดของ b^k กับ แกน z^k และ z^{k-1} ถ้าไม่มี ใช้จุดตัด x^k กับค่าระหว่างแกน x^k กับ z^{k-1}
10. คำนวณ Θ_k เป็นมุมการหมุนจาก x^{k-1} ไป x^k ที่หมุนรอบแกน z^{k-1}
11. คำนวณค่า d_k เป็นระยะห่างจากระนาบ L_{k-1} ไปยังจุด b^k ซึ่งวัดตามแนวแกน z^{k-1}
12. คำนวณค่า a_k เป็นระยะห่างจากจุด b^k ไปยัง ระนาบ L_k ซึ่งวัดตามแนวแกน x^k
13. คำนวณค่า α_k เป็นมุมการหมุนจาก z^{k-1} ไป z^k ที่หมุนรอบแกน x^k
14. ให้ $k=k+1$ ถ้าไม่น้อยกว่าหรือเท่ากับn ให้ทำข้อที่3ใหม่

2.2.6 สมการแขนกล

เมื่อเราทำการจำลองการเชื่อมต่อตำแหน่งของแขนกลด้วยวิธีของ DenavitและHartenbergแล้ว เราสามารถแปลงระนาบ k ไปเป็นระนาบ $k-1$ โดยการทำการแปลงพิกัดเมทริกซ์ โดยการคูณเมทริกซ์ที่แปลงพิกัดระนาบแล้วเข้าด้วยกัน เราก็จะได้เมทริกซ์ที่แปลงพิกัดระนาบซึ่งอยู่บนระนาบเดียวกับระนาบของฐาน

2.2.6.1 สมการเมทริกซ์ของแขนกล

ขั้นตอนในการแปลงพิกัดเมทริกซ์จากระนาบ k ไปเป็นระนาบ $k-1$ มี 4 ขั้นตอน ซึ่งแต่ละขั้นตอนจะใช้ค่าพารามิเตอร์จากตารางที่ 2-3 ในการคำนวณแปลงพิกัดเมทริกซ์ เพื่อให้ระนาบ $k-1$ ตรงกับ ระนาบ k จากวิธีของ DenavitและHartenberg ซึ่งทำให้แปลงพิกัดระนาบตามตารางที่ 2-4

ตารางที่ 2-4 ขั้นตอนในการแปลงระนาบ $k-1$ ไป k

ลำดับขั้นตอน	คำอธิบาย
1	หมุนระนาบ L_{k-1} รอบแกน z^{k-1} เป็นมุม Θ_k
2	เคลื่อนระนาบ L_{k-1} ตามแกน z^{k-1} เป็นระยะทาง d_k
3	เคลื่อนระนาบ L_{k-1} ตามแกน x^{k-1} เป็นระยะทาง a_k
4	หมุนระนาบ L_{k-1} รอบแกน x^{k-1} เป็นมุม α_k

จากนั้นนำสมการแขนกลที่ได้จากการแปลงระนาบมาคูณเมทริกซ์เพื่อหาตำแหน่ง ข้อต่อ โดยเทียบกับระนาบของฐานได้สมการเมทริกซ์นี้

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

ในการแก้สมการแขนกล เราต้องคำนวณหาตำแหน่งและระนาบของtoolที่เทียบกับระนาบของฐาน การแปลงพิกัดระนาบของtoolไปเป็นระนาบของฐาน ต้องคำนวณจากtoolย้อนกลับมาที่ฐาน ที่ระนาบบนแขนกลโดยจะได้สมการดังนี้

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

ซึ่งเรายังสามารถแบ่งการคำนวณในการแปลงพิกัดระนาบได้ตามนี้

$$T_{\text{base}}^{\text{wrist}}(q) = T_0^1(q_1)T_1^2(q_2)T_2^3(q_3) = T_0^3(q) \quad \text{แปลงพิกัดระนาบจากข้อต่อมาฐาน}$$

$$T_{\text{wrist}}^{\text{tool}}(q) = T_3^4(q_4)T_4^5(q_5) \cdots T_{n-1}^n(q_n) = T_3^n(q) \quad \text{แปลงพิกัดระนาบจากtoolมาข้อต่อ}$$

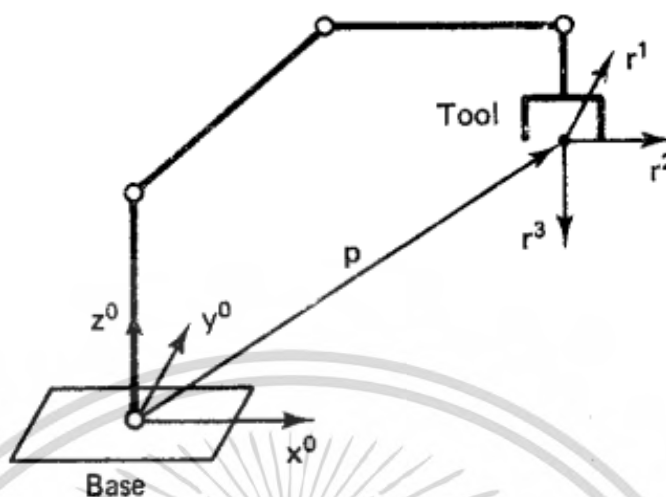
$$T_{\text{base}}^{\text{tool}}(q) = T_{\text{base}}^{\text{wrist}}(q_1, q_2, q_3)T_{\text{wrist}}^{\text{tool}}(q_4, q_5, \dots, q_n) \quad \text{แปลงพิกัดระนาบจากtoolมาข้อต่อมาฐาน}$$

2.2.6.2 การสร้างสมการแขนกล

จากการแปลงพิกัดระนาบเราจะได้สมการเมทริกซ์แขนกลซึ่งเป็นเมทริกซ์ 4×4 ได้ดังนี้

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

โดยเมทริกซ์ย่อย 3×3 มุมบนด้านซ้าย $R(q)$ จะบอกทิศทางระนาบของtool เมื่อเทียบกับระนาบของฐาน เมทริกซ์ย่อย 3×1 จะบอกตำแหน่งของtool เมื่อเทียบกับตำแหน่งของฐาน จากสมการข้างต้นที่กล่าวไป จะทำให้เกิดเวกเตอร์ผลลัพธ์ของtoolเมื่อเทียบกับฐานตามรูปที่2-11



รูปที่ 2-12 เวกเตอร์ผลลัพธ์ของtoolเมื่อเทียบกับฐาน

2.3 จลนศาสตร์ผกผันของแขนกล (Inverse Kinematics)

จากการหาจลนศาสตร์ทางตรงของแขนกล(Direct Kinematics)ซึ่งเป็นการกำหนดค่าตัวแปรของ ข้อต่อ เพื่อหาค่าตำแหน่งของ tool ในหัวข้อจลนศาสตร์ผกผันของแขนกล (Inverse Kinematics) จะเป็นการกำหนดค่าตำแหน่งของ tool เพื่อหาค่าตัวแปร ข้อต่อ ของแขนกลโดยยึดตำแหน่งฐานเป็นหลัก สาเหตุที่ต้องใช้จลนศาสตร์ผกผันของแขนกล (Inverse Kinematics)คือการทำงานในอุตสาหกรรมหรือปฏิบัติการต่างๆ จะใช้ตำแหน่งtoolของแขนกลในการทำงาน เมื่อเราได้ค่าตำแหน่ง toolของแขนกลจะนำค่าเหล่านั้นไปหาค่าตัวแปร ข้อต่อ ของแขนกลเพื่อควบคุมการเคลื่อนที่ของแขนกล

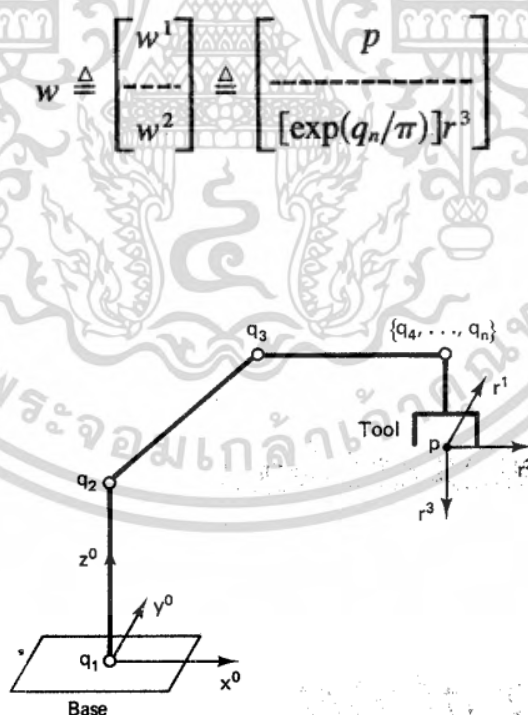
ในการหาสมการในการกำหนดค่าตำแหน่งของ tool เพื่อหาค่าตัวแปร ข้อต่อ ของแขนกลนั้น มีความยากเพราะในสมการแขนกลแต่ละแขน การทำงานแต่ละอย่าง จะมีความแตกต่างกันมาก จึงได้มีการจำลองตำแหน่งและระนาบtoolของแขนกลขึ้นมาเรียกว่า เวกเตอร์กำหนดค่าtool

2.3.1 ปัญหาจลนศาสตร์ผกผันของแขนกล

กุญแจสำคัญในการหาเวกเตอร์กำหนดค่า tool เพื่อหาค่าตัวแปร ข้อต่อ ของแขนกลคือการเชื่อมต่อตำแหน่ง(Link coordinates)ของแขนกล ในการกำหนดค่าตัวแปร ข้อต่อ ของแขนกลจะต้องรู้รูปแบบของ ข้อต่อ ของแขนกลในแต่ละจุดและการแปลงพิกัดระนาบสมการเมทริกซ์แขนกลนี้

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

จึงจะสามารถกำหนดค่าตัวแปร ข้อต่อ ของแขนกลให้เคลื่อนที่ตามที่กำหนดได้ ซึ่งกำหนดค่าตัวแปร ข้อต่อ ของแขนกลเป็น q_n ตามรูปที่ 2-12 ซึ่งเมื่อเราสามารถหาสมการแขนกลกำหนดค่าตัวแปรของ ข้อต่อ เพื่อหาค่าตำแหน่งของ tool ได้แล้ว เราจะนำมาแก้สมการแขนกลกำหนดค่าตำแหน่งของ tool เพื่อหาค่าตัวแปร ข้อต่อ ของแขนกลแทน ซึ่งเราจะหาเวกเตอร์กำหนดค่า tool ที่แสดงตำแหน่งและระนาบจากสมการแขนกลนั้น สมการเวกเตอร์กำหนดค่า tool มีนิยามดังนี้



รูปที่ 2-13 ค่าตัวแปร ข้อต่อ ของแขนกลในแต่ละ ข้อต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราได้สมการเวกเตอร์กำหนดค่า tool ซึ่งจะมีตัวแปร ข้อต่อ ของแขนกลแต่ละตัว ก็จะนำมาแก้สมการหาตัวแปรนั้นๆ

2.3.2 ตัวอย่างการหาจลนศาสตร์ผกผันของแขนกล

ในการแก้สมการจากค่าตำแหน่งของ tool เพื่อหาค่าตัวแปร ข้อต่อ ของแขนกลในแต่ละแขนกลนั้นจะแตกต่างกันไปตามการหาจลนศาสตร์ทางตรงของแขนกล(Direct Kinematics)นั้นๆ ซึ่งจะยกตัวอย่างเป็นแขนกลสกร่าสี่แกน โมเดลรุ่น Epson LS6-B SCARA Robot ตามรูปที่ 2-13

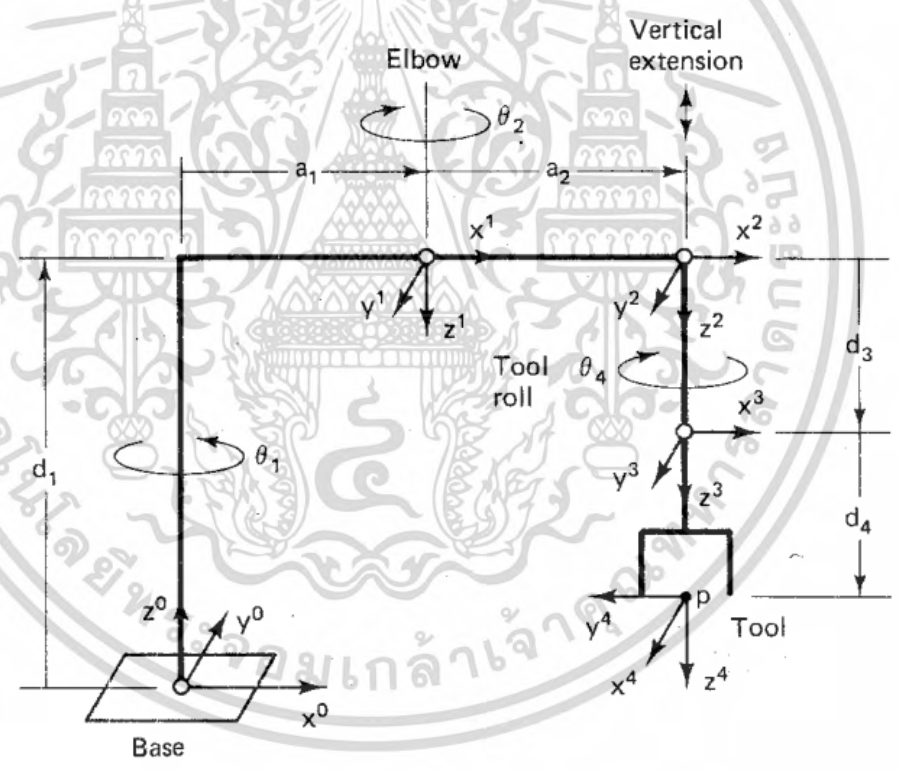


รูปที่ 2-14 แขนกลสกร่าสี่แกน โมเดลรุ่น Epson LS6-B SCARA Robot

เริ่มจากการหาค่าพารามิเตอร์จลนศาสตร์ตามตารางที่ 2-5 และการเชื่อมต่อตำแหน่ง(Link coordinates)ของแขนกลสกร่าสี่แกน โมเดลรุ่น Epson LS6-B SCARA Robot ได้ตามรูปที่ 2-14

ตารางที่ 2-5 ค่าพารามิเตอร์จลนศาสตร์

ลำดับแกน	θ	d	a	α
1	q_1	d_1	a_1	π
2	q_2	0	a_2	0
3	0	d_3	0	0
4	q_4	d_4	0	0



รูปที่ 2-15 การเชื่อมต่อตำแหน่ง(Link coordinates)ของแขนกลสคาร์่าสี่แกน โมเดลรุ่น Epson LS6-B SCARA Robot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นเรานำมาหาสมการแขนกลจลนศาสตร์ทางตรงของแขนกล(Direct Kinematics)ได้

$$\begin{aligned}
 T_{\text{base}}^{\text{tool}} &= T_0^1 T_1^2 T_2^3 T_3^4 \\
 &= \begin{bmatrix} C_1 & S_1 & 0 & a_1 C_1 \\ S_1 & -C_1 & 0 & a_1 S_1 \\ 0 & 0 & -1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_2 & -S_2 & 0 & a_2 C_2 \\ S_2 & C_2 & 0 & a_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_4 & -S_4 & 0 & 0 \\ S_4 & C_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} C_{1-2} & S_{1-2} & 0 & a_1 C_1 + a_2 C_{1-2} \\ S_{1-2} & -C_{1-2} & 0 & a_1 S_1 + a_2 S_{1-2} \\ 0 & 0 & -1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_4 & -S_4 & 0 & 0 \\ S_4 & C_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} C_{1-2} & S_{1-2} & 0 & a_1 C_1 + a_2 C_{1-2} \\ S_{1-2} & -C_{1-2} & 0 & a_1 S_1 + a_2 S_{1-2} \\ 0 & 0 & -1 & d_1 - q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_4 & -S_4 & 0 & 0 \\ S_4 & C_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-8-3)
 \end{aligned}$$

$$T_{\text{base}}^{\text{tool}} = \begin{bmatrix} C_{1-2-4} & S_{1-2-4} & 0 & a_1 C_1 + a_2 C_{1-2} \\ S_{1-2-4} & -C_{1-2-4} & 0 & a_1 S_1 + a_2 S_{1-2} \\ 0 & 0 & -1 & d_1 - q_3 - d_4 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

หาตัวแปร ข้อต่อ q_3 ได้จากสมการ

$$q_3 = d_1 - d_4 - w_3$$

หาตัวแปร ข้อต่อ q_2 ได้จากสมการ

$$w_1^2 + w_2^2 = a_1^2 + 2a_1 a_2 C_2 + a_2^2$$

$$q_2 = \pm \arccos \frac{w_1^2 + w_2^2 - a_1^2 - a_2^2}{2a_1 a_2}$$

หาตัวแปร ข้อต่อ q_1 ได้จากสมการ

$$(a_1 + a_2 C_2)C_1 + (a_2 S_2)S_1 = w_1$$

$$(-a_2 S_2)C_1 + (a_1 + a_2 C_2)S_1 = w_2$$

$$S_1 = \frac{a_2 S_2 w_1 + (a_1 + a_2 C_2)w_2}{(a_2 S_2)^2 + (a_1 + a_2 C_2)^2}$$

$$C_1 = \frac{(a_1 + a_2 C_2)w_1 - a_2 S_2 w_2}{(a_2 S_2)^2 + (a_1 + a_2 C_2)^2}$$

$$q_1 = \text{atan2} [a_2 S_2 w_1 + (a_1 + a_2 C_2)w_2, (a_1 + a_2 C_2)w_1 - a_2 S_2 w_2]$$

หาตัวแปร ข้อต่อ q_4 ได้จากระนาบการหมุน

$$R(q) = \begin{bmatrix} C_{1-2-4} & S_{1-2-4} & 0 \\ S_{1-2-4} & -C_{1-2-4} & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

โดยใช้ฟังก์ชันตรีโกณมิติ $\tan A = \sin A / \cos A$ ได้

$$q_{1-2-4} = \text{atan2} (R_{21}, R_{11})$$

$$q_4 = q_1 - q_2 - q_{1-2-4}$$

จากตัวแปร ข้อต่อ แต่ละค่าเราก็จะสามารถกำหนดการควบคุมแกนหมุนในแต่ละแกนได้จากตำแหน่ง tool ของแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 โปรแกรม MATLAB

MATLAB เป็นโปรแกรมคอมพิวเตอร์สมรรถนะสูงเพื่อใช้ในการคำนวณทางเทคนิค MATLAB ได้รวมการคำนวณ การเขียนโปรแกรมและการแสดงผลรวมกันอยู่ในตัวโปรแกรมเดียวได้อย่างมีประสิทธิภาพ และอยู่ในลักษณะที่ง่ายต่อการใช้งาน นอกจากนี้ลักษณะของการเขียนสมการในโปรแกรมก็จะเหมือนการเขียนสมการคณิตศาสตร์ที่เราคุ้นเคยดีอยู่แล้วงานที่ทั่วไปที่ใช้ MATLAB ก็เช่นการคำนวณค่าการคำนวณทั่วไป การสร้างแบบจำลองและการทดสอบแบบจำลอง การวิเคราะห์ข้อมูล การแสดงผลในรูปกราฟทั้งโดยทั่วไปและกราฟทางด้านทางวิทยาศาสตร์และวิศวกรรม สามารถสร้างโปรแกรมในลักษณะที่ติดต่อกับผู้ใช้ทางกราฟฟิกการทำงานของ MATLAB จะสามารถทำงานได้ทั้งในลักษณะของการติดต่อโดยตรง (Interactive) คือการเขียนคำสั่งเข้าไปทีละคำสั่ง เพื่อให้ MATLAB ประมวลผลไปเรื่อยๆ หรือสามารถที่จะรวบรวมชุดคำสั่งเหล่านั้นเป็นโปรแกรมก็ได้ข้อสำคัญอย่างหนึ่งของ MATLAB ก็คือข้อมูลทุกตัวจะถูกเก็บในลักษณะของ array คือในแต่ละตัวแปรจะได้รับการแบ่งเป็นส่วนย่อยเล็กๆขึ้น (หรือจะได้รับการแบ่งเป็น element นั้นเอง) ซึ่งการใช้ตัวแปรเป็น array ใน MATLAB นี้เราไม่จำเป็นที่จะต้องจง dimension เหมือนกับการเขียนโปรแกรมในภาษาขั้นต่ำทั่วไป ซึ่งทำให้เราสามารถที่จะแก้ปัญหาของตัวแปรที่อยู่ในลักษณะของ matrix และ vector ได้โดยง่าย ซึ่งทำให้เราลดเวลาการทำงานลงได้อย่างมากเมื่อเทียบกับการเขียนโปรแกรมโดย C หรือ Fortran

```

1 k1c
2 format shortg
3 c1c
4 s11 = [ 1 , 0 , 0 , 0 ];
5 s12 = [ 0 , 1 , 0 , 0 ];
6 s13 = [ 0 , 0 , 1 , 0 ];
7 s14 = [ 0 , 0 , 0 , 1 ];
8 mm_base1 = [s11;s12;s13;s14];
9 % s11 = [ 0 , 0 , 0 , 0 ; pi/2 ];
10 % s12 = [ 0 , 0 , 150 , 0 ];
11 % s13 = [ -pi/2 , 0 , 0 , pi/2 ];
12 % s14 = [ 0 , 200 , 0 , pi/2 ];
13 % s15 = [ pi/2 , 0 , 0 , -pi/2 ];
14 % s16 = [ 0 , 100 , 0 , 0 ];
15 % k4 = [ 111; 112; 113; 114; 115; 116 ];
16 s11 = [ 0 , 240 , 10 , -pi/2 ];
17 s12 = [ -pi/4 , 0 , 221 , 0 ];
18 s13 = [ pi/4 , 0 , 221 , 0 ];
19 s14 = [ -pi/2 , 0 , 0 , -pi/2 ];
20 s15 = [ -pi/2 , 110 , 0 , pi/2 ];
21 s16 = [ 0 , 135 , 0 , 0 ];
22 k5 = [s11;s12;s13;s14;s15;s16];
23 % s11 = [ 0 , 440 , 0 , -pi/2 ];
24 % s12 = [ 0 , 0 , 250 , 0 ];

```

รูปที่ 2-16 หน้าต่างโปรแกรม MATLAB R2020b ที่ใช้ในการเขียนโปรแกรมคำนวณสมการ

แขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 โปรแกรม Arduino IDE

โปรแกรม Arduino IDE เป็นโปรแกรมที่ใช้สำหรับการเขียนโปรแกรมควบคุมและสามารถทำการแปลงไฟล์ดังกล่าวเพื่อนำไปอัปโหลดลงยังบอร์ด Arduino ซึ่ง Arduino IDE จะทำหน้าที่ ติดต่อระหว่างคอมพิวเตอร์ ไม่ว่าจะเป็นระบบ Windows, Mac OS X หรือ Linux กับ บอร์ด Arduino



รูปที่ 2-17 หน้าต่างโปรแกรม Arduino IDE

2.6 มอเตอร์ไฟฟ้ากระแสตรง

มอเตอร์ไฟฟ้าเป็นอุปกรณ์ไฟฟ้าชนิดหนึ่งทำหน้าที่เปลี่ยนพลังงานไฟฟ้าเป็นพลังงานกล ซึ่งมีใช้งานกันแพร่หลายตั้งแต่อุปกรณ์ของเด็กเล่นเครื่องใช้ไฟฟ้าภายในบ้าน โรงงานอุตสาหกรรมขนาดเล็ก ขนาดกลาง และขนาดใหญ่ ซึ่งนำมาใช้งานในลักษณะเป็นเครื่องผ่อนแรง อุปกรณ์อำนวยความสะดวก เครื่องออกกำลังกาย และควบคุมการทำงานของเครื่องจักรกลในกระบวนการผลิตต่าง ๆ

ในโครงการนี้ได้เลือกใช้มอเตอร์ไฟฟ้ากระแสตรงเพราะเป็นที่นิยมใช้กันมาก เนื่องจากสามารถควบคุมความเร็วได้ดีกว่า มอเตอร์ไฟฟ้ากระแสสลับ เหมาะสำหรับการควบคุมความเร็ว ซึ่งมีความเที่ยงตรงและแม่นยำมาก ซึ่งที่มักพบเห็นโดยทั่วไปในเครื่องออกกำลังกาย ประเภทลู่วิ่งไฟฟ้า ระบบควบคุมสายพานการผลิตในโรงงานอุตสาหกรรม ซึ่งสามารถปรับความเร็วได้ตั้งแต่ความเร็วต่ำสุดจนถึงความเร็วสูงสุดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ยังนำมาใช้ในระบบขับเคลื่อนในรถยนต์ไฟฟ้า โรงงานทอผ้า โรงงานปั่นเส้นใยสังเคราะห์ และอื่น ๆ อีกมากมาย มีลักษณะภายในตามรูปที่ 2-16 หลักการทำงาน คือ เมื่อป้อนแรงดันไฟฟ้ากระแสตรงให้กับมอเตอร์กระแสส่วนหนึ่งจะผ่านแปรงถ่านผ่านคอมมิวเตเตอร์เข้าไปในขดลวดอาร์มาเจอร์สร้างสนามแม่เหล็กขึ้น กระแสไฟฟ้าอีกส่วนหนึ่งจะไหลเข้าไปในขดลวดเกิดสนามแม่เหล็ก 2 สนาม เกิดขั้วเหนือและขั้วใต้และตามคุณสมบัติของเส้นแรงแม่เหล็กจะเสริมกันเมื่ออยู่ในทิศทางเดียวกัน และจะหักล้างกันเมื่ออยู่ในทิศทางตรงกันข้าม ทำให้เกิดแรงบิดในตัวอาร์มาเจอร์ซึ่งวางอยู่บนแกนเพลลาเกิดการหมุนตัวขณะที่อาร์มาเจอร์หมุนเรียกว่า โรเตอร์(Rotor) ซึ่งการหมุนที่เกิดจากอำนาจของเส้นแรงแม่เหล็กทั้งสองทำปฏิกิริยาต่อกัน และทำให้ขดลวดอาร์มาเจอร์หรือโรเตอร์หมุน เป็นไปตามกฎซ้ายของเฟลมมิ่ง (Fleming left hand rule)



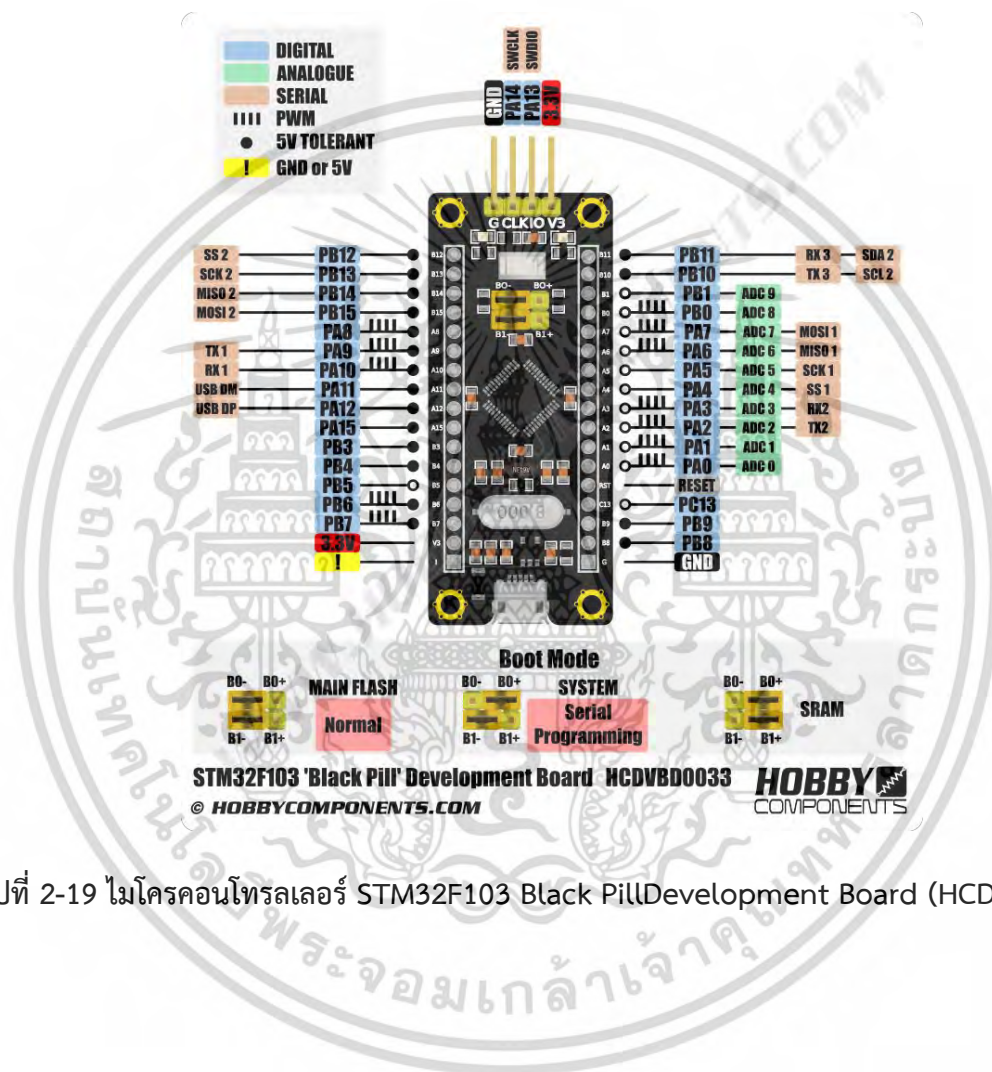
รูปที่ 2-18 ลักษณะภายในมอเตอร์ไฟฟ้ากระแสตรง

2.7 ไมโครคอนโทรลเลอร์STM32

STM32 เป็นชิปที่สร้างโดย STMicroelectronics ภายในใช้ CPU ARM 32bit Cortex - M3 ซึ่งออกแบบมาเพื่อไมโครคอนโทรลเลอร์โดยเฉพาะ ความถี่ CPU สูงสุด 72MHz มีรอม 64KB และแรม 20KB มี GPIO ให้ใช้ 37 ขา ADC 12bit จำนวน 10 ช่อง รองรับบัส SPI I2C UART UART และ CAN ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงดันไฟฟ้าได้ตั้งแต่ 2V ถึง 3.6V STM32 แตกต่างจากชิปตระกูลอื่น ๆ ตรงที่ ไม่มี IDE และคอมไพเลอร์ เป็นของตัวเอง การใช้งานต้องดาวน์โหลด IDE และคอมไพเลอร์จากบริษัทอื่นมาใช้งาน ตัวอย่าง IDE และคอมไพเลอร์ที่นิยมใช้งานกัน ได้แก่ IAR Embedded Workbench และ ARMmbed

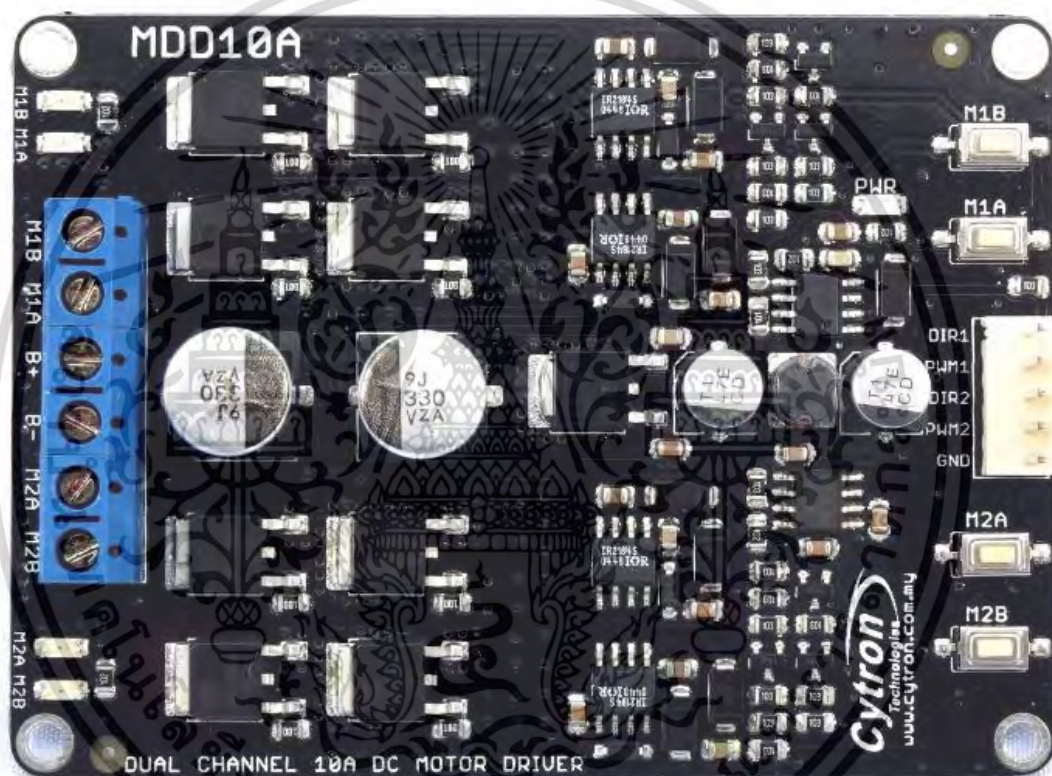


รูปที่ 2-19 ไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 ตัวขับมอเตอร์

บอร์ดที่มีช่องต่อแหล่งจ่ายไฟสำหรับมอเตอร์ มีช่องต่อมอเตอร์ และมีช่องต่อสัญญาณควบคุม เพื่อใช้ในการควบคุมมอเตอร์ไฟฟ้า ซึ่งมีช่องต่อสัญญาณควบคุมมีช่อง DIR(ทิศทาง) และช่อง PWM(ความเร็ว)



รูปที่ 2-20 รูปตัวขับมอเตอร์ MDD10A Dual Channel 10A DC Motor Driver

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 เอ็นโคเดอร์ (Encoder)

เอ็นโคเดอร์ (Encoder) เป็นเซ็นเซอร์ชนิดหนึ่ง ที่สามารถนำมาประยุกต์ใช้ในการวัดตำแหน่งหรือระยะขจัด (Position or Displacement Sensor) มีด้วยกัน 2 ลักษณะ คือ เป็นแบบเชิงเส้น (Linear Encoder) และเป็นแบบหมุน (Rotary Encoder) และในแต่ละลักษณะยังแบ่งออกเป็นแบบที่สัญญาณเอาต์พุตที่เกิดจากการวัดอยู่ในรูปแบบของรหัสเชิงตรรกะ (Digital coding signal) ซึ่งจะเรียกเอ็นโคเดอร์ในรูปแบบนี้ว่า “Absolute Encoder” ส่วนอีกลักษณะหนึ่งจะเป็นแบบที่ให้สัญญาณเอาต์พุตที่เกิดจากการวัดอยู่ในรูปแบบของพัลส์ (Pulse signal) ซึ่งมักจะถูกเรียกว่า “Incremental Encoder”



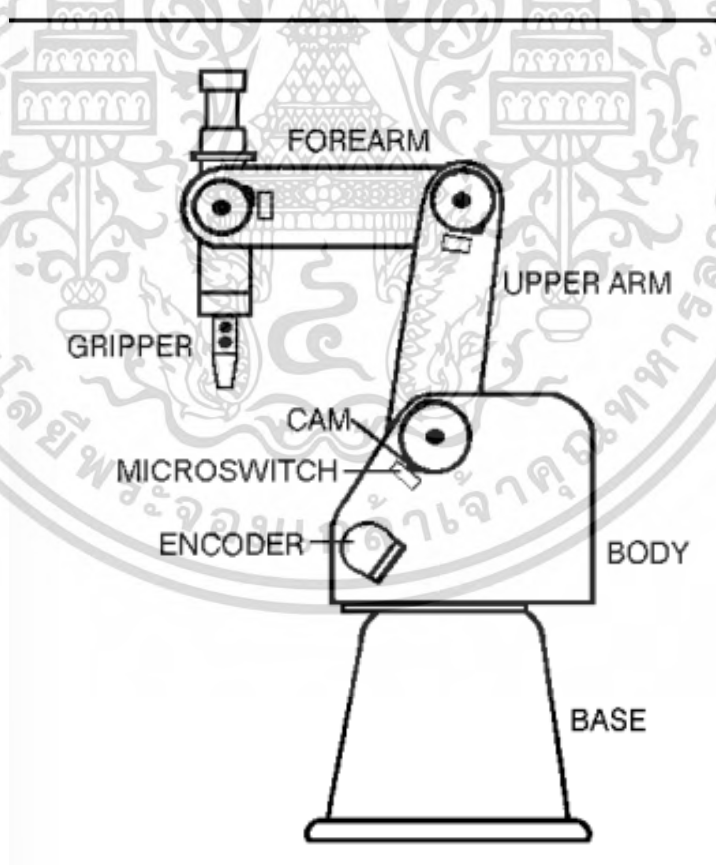
รูปที่ 2-20 รูปตัวเอ็นโคเดอร์ Omron e6b2-cwz6c

บทที่ 3

หลักการออกแบบและขั้นตอนการทำงาน

3.1 แบบจำลองแขนกลตัดโลหะด้วยพลาสติก

ในโครงการนี้ได้นำแบบจำลองแขนกล SCORBOT-ER 5Plus มาพัฒนาใหม่เป็นแขนกลตัดโลหะด้วยพลาสติกตามรูปที่ ซึ่งวัสดุจะใช้เป็นแผ่นอลูมิเนียมเป็นโครงสร้างหลักและใช้เส้นใยพลาสติก (PLA filament) ที่ใช้เครื่องพิมพ์สามมิติ (3D printer) เป็นส่วนโครงสร้างเสริมต่างๆ โดยส่วน Gripper ของแขนกล SCORBOT-ER 5Plus จะเปลี่ยนมาเป็นหัวตัดโลหะด้วยพลาสติก



รูปที่ 3-1 โมเดลแขนกล SCORBOT-ER 5Plus

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-2 แบบจำลองแขนกลตัดโลหะด้วยพลาสมา

แบบจำลองแขนกลตัดโลหะด้วยพลาสมานี้จะมีส่วนประกอบนอกตัวโครงสร้างหลักดังนี้

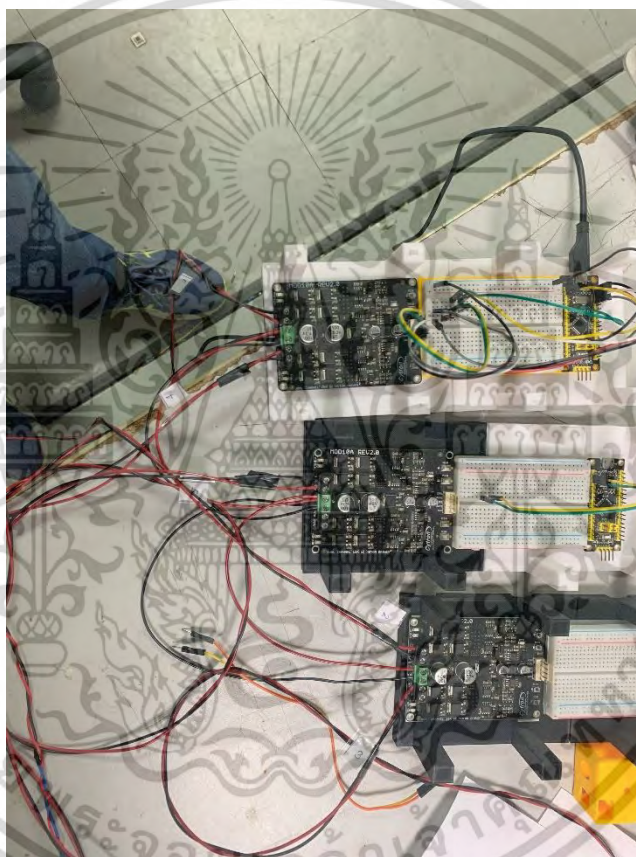
- Series GM9000 LO-COG Brush Commutated DC Gear motors
- ไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033)
- Omron e6b2-cwz6c Encoder
- MDD10A Dual Channel 10A DC Motor Driver
-

3.2 การต่อวงจรควบคุมแขนกลตัดโลหะด้วยพลาสมา

วงจรรควบคุมแขนกลตัดโลหะด้วยพลาสมานั้น จะใช้ไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033) ในการรับโปรแกรมคำสั่งจากคอมพิวเตอร์ที่ถูกแปลงเป็น RS232 จากโปรแกรมในคอมพิวเตอร์ ผ่านตัว USB TO TTL RS232 CH340G adapter จากนั้นนำ

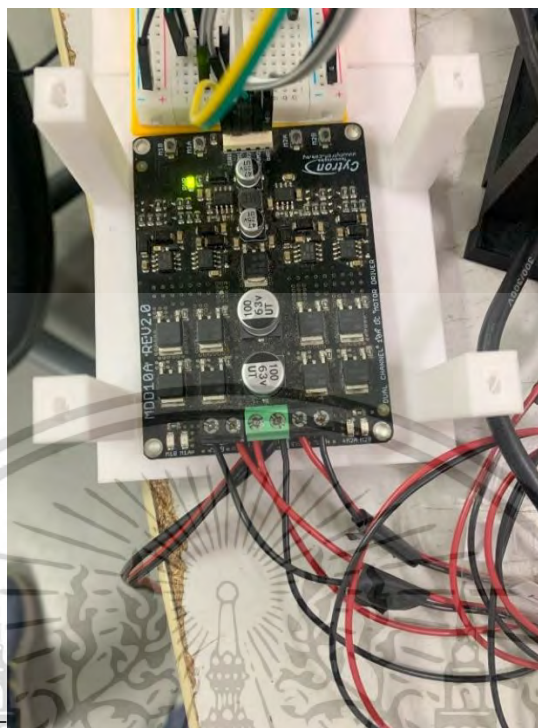
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สายสัญญาณของ Omron e6b2-cwz6c Encoder มาต่อเข้ากับไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033) เพื่อรับค่าตัวแปรข้อต่อของแกนกลมาคำนวณในตัวของไมโครคอนโทรลเลอร์ แล้วส่งสัญญาณควบคุมไปยัง MDD10A Dual Channel 10A DC Motor Driver เพื่อควบคุมให้มอเตอร์ทำงานในระบบจริงแทนข้อต่อของแกนกลในระบบจำลองโปรแกรม MATLAB

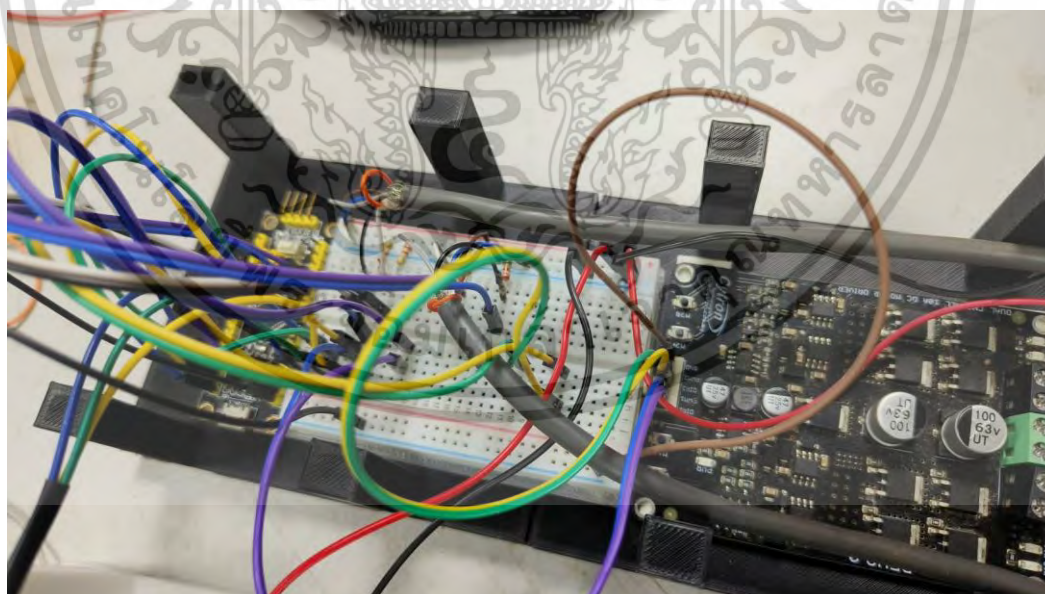


รูปที่ 3-3 วงจรการควบคุมแกนกลตัดโลหะด้วยพลาสติก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-4 MDD10A Dual Channel 10A DC Motor Driver



รูปที่ 3-5 สายสัญญาณจาก Omron e6b2-cwz6c Encoder มาต่อเข้ากับไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 โปรแกรมการควบคุมแขนกลตัดโลหะด้วยพลาสติก

สัญญาณที่ได้จาก Omron e6b2-cwz6c Encoder จะส่งไปยังไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033) เพื่อแปลงค่าเป็นตัวเลขแล้วแสดงค่าบวก-ลบเพื่อบอกทิศทางของการหมุน รวมถึงนับจำนวนรอบในการหมุน มาคำนวณในโปรแกรมMATLAB ซึ่งใช้หลักการจลนศาสตร์ผกผันของแขนกล (Inverse Kinematics) ในการคำนวณหาค่าตัวแปรข้อต่อในแขนกลจากค่าที่ได้จาก Omron e6b2-cwz6c Encoder โดยค่าที่ได้จะต้องแปลงค่าสัญญาณก่อนจากนั้นนำไปคำนวณในสมการแขนกล

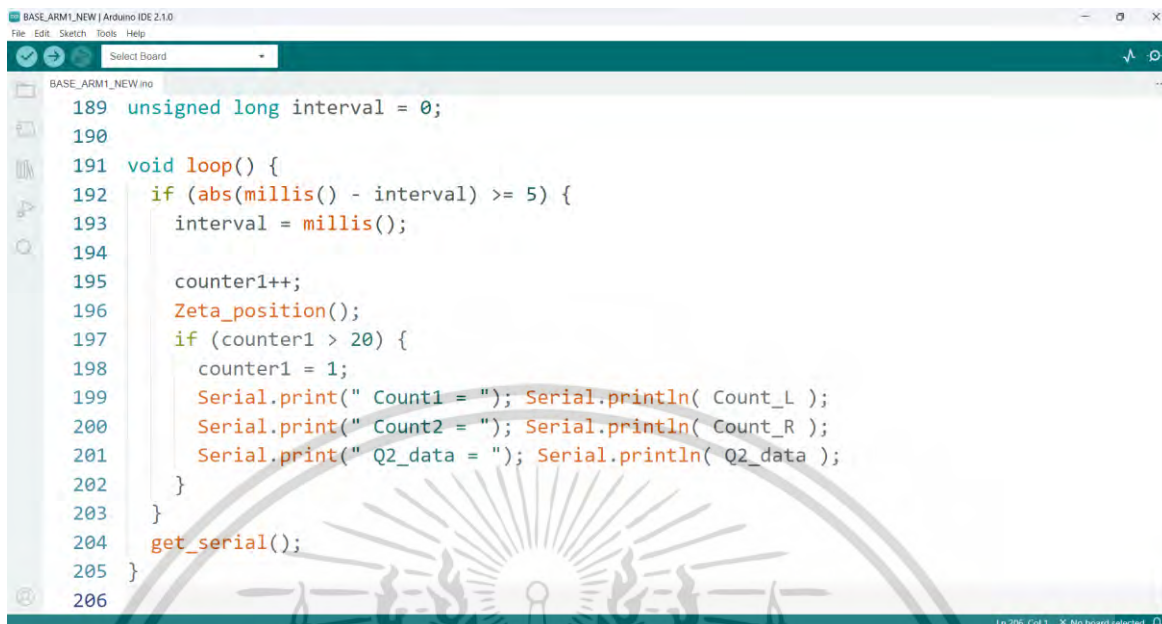
```

1  clc
2  cla
3
4  jj1 = [ 1 , 0 , 0 , 0 ];
5  jj2 = [ 0 , 1 , 0 , 0 ];
6  jj3 = [ 0 , 0 , 1 , 0 ];
7  jj4 = [ 0 , 0 , 0 , 1 ];
8  Base = [jj1;jj2;jj3;jj4];
9  jj1 = [ 0 , 349 , 16 , -pi/2 ];
10 jj2 = [ 0 , 0 , 221 , 0 ];
11 jj3 = [ 0 , 0 , 221 , 0 ];
12 jj4 = [ pi/2 , 0 , 0 , -pi/2 ];
13 jj5 = [ pi/2 , 100 , 0 , pi/2 ];
14 jj6 = [ 0 , 135 , 0 , 0 ];
15 ki = [jj1;jj2;jj3;jj4;jj5;jj6];
16 jj1 = [ 0 , 1 , 0 , 523.27 ];
17 jj2 = [ 1 , 0 , 0 , 0 ];
18 jj3 = [ 0 , 0 , -1 , 370.27 ];
19 jj4 = [ 0 , 0 , 0 , 1 ];
20 RP = [jj1;jj2;jj3;jj4];
21
22
23

```

รูปที่ 3-6 รูปโปรแกรมสมการแขนกลในโปรแกรมMATLAB

เมื่อได้ค่าที่ต้องการจากโปรแกรมMATLABแล้ว จะนำค่าจาก Omron e6b2-cwz6c Encoder มาเปรียบเทียบ เพื่อให้ได้ค่าที่ตรงกับที่คำนวณจากโปรแกรมMATLAB ถ้าหากค่าที่เปรียบเทียบไม่ตรงกัน จะสั่งให้มอเตอร์ทำงานเพื่อให้ค่าที่ได้จาก Omron e6b2-cwz6c Encoder ตรงกับค่าที่คำนวณจากโปรแกรมMATLAB



```

BASE_ARM1_NEW | Arduino IDE 2.1.0
File Edit Sketch Tools Help
Select Board
BASE_ARM1_NEW/ino
189 unsigned long interval = 0;
190
191 void loop() {
192   if (abs(millis() - interval) >= 5) {
193     interval = millis();
194
195     counter1++;
196     Zeta_position();
197     if (counter1 > 20) {
198       counter1 = 1;
199       Serial.print(" Count1 = "); Serial.println( Count_L );
200       Serial.print(" Count2 = "); Serial.println( Count_R );
201       Serial.print(" Q2_data = "); Serial.println( Q2_data );
202     }
203   }
204   get_serial();
205 }
206
Ln 206, Col 1 X No board selected

```

รูปที่ 3-7 โปรแกรมเปรียบเทียบค่าจาก Omron e6b2-cwz6c Encoder กับค่าที่คำนวณจากโปรแกรมMATLAB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองการแปลงค่าจาก Omron e6b2-cwz6c Encoder

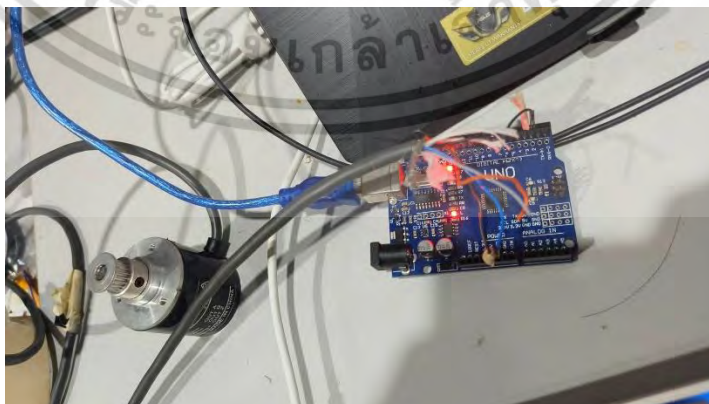
การทดลองการแปลงค่าจาก Omron e6b2-cwz6c Encoder โดยการนำสายสัญญาณของตัว Omron e6b2-cwz6c Encoder มาต่อเข้ากับไมโครคอนโทรลเลอร์ Arduino UNO เพื่อหาค่าการเปลี่ยนแปลงในตัว Omron e6b2-cwz6c Encoder

```

Sketch #41a (Arduino IDE 2.12)
File Edit Sketch Tools Help
Arduino IDE
encoder_mblisa.ino
1 volatile unsigned int temp, counter = 0; // Initial value of counter
2
3 void setup() {
4   Serial.begin(9600);
5   pinMode(2, INPUT_PULLUP); // Input of the encoder pin 2
6   pinMode(3, INPUT_PULLUP); // Input of the encoder pin 3
7   pinMode(10, OUTPUT);
8   // Pin 10 will be used as output pin for the Arduino IDE
9   digitalWrite(10, HIGH);
10  // Pin 10 will be used as output pin for the Arduino IDE
11  digitalWrite(10, LOW);
12  // Pin 10 will be used as output pin for the Arduino IDE
13  // Pin 10 will be used as output pin for the Arduino IDE
14  // Pin 10 will be used as output pin for the Arduino IDE
15  // Pin 10 will be used as output pin for the Arduino IDE
16  // Pin 10 will be used as output pin for the Arduino IDE
17  // Pin 10 will be used as output pin for the Arduino IDE
18  // Pin 10 will be used as output pin for the Arduino IDE
19  // Pin 10 will be used as output pin for the Arduino IDE
20  // Pin 10 will be used as output pin for the Arduino IDE
21  // Pin 10 will be used as output pin for the Arduino IDE
22  // Pin 10 will be used as output pin for the Arduino IDE
23  // Pin 10 will be used as output pin for the Arduino IDE
24  // Pin 10 will be used as output pin for the Arduino IDE
25  // Pin 10 will be used as output pin for the Arduino IDE
26  // Pin 10 will be used as output pin for the Arduino IDE
27  // Pin 10 will be used as output pin for the Arduino IDE
28  // Pin 10 will be used as output pin for the Arduino IDE
29  // Pin 10 will be used as output pin for the Arduino IDE
30  // Pin 10 will be used as output pin for the Arduino IDE
31  // Pin 10 will be used as output pin for the Arduino IDE
32  // Pin 10 will be used as output pin for the Arduino IDE
33  // Pin 10 will be used as output pin for the Arduino IDE
34  // Pin 10 will be used as output pin for the Arduino IDE
35  // Pin 10 will be used as output pin for the Arduino IDE
36  // Pin 10 will be used as output pin for the Arduino IDE
37  // Pin 10 will be used as output pin for the Arduino IDE
38  // Pin 10 will be used as output pin for the Arduino IDE
39  // Pin 10 will be used as output pin for the Arduino IDE
40  // Pin 10 will be used as output pin for the Arduino IDE
41  // Pin 10 will be used as output pin for the Arduino IDE
42  // Pin 10 will be used as output pin for the Arduino IDE
43  // Pin 10 will be used as output pin for the Arduino IDE
44  // Pin 10 will be used as output pin for the Arduino IDE
45  // Pin 10 will be used as output pin for the Arduino IDE
46  // Pin 10 will be used as output pin for the Arduino IDE
47  // Pin 10 will be used as output pin for the Arduino IDE
48  // Pin 10 will be used as output pin for the Arduino IDE
49  // Pin 10 will be used as output pin for the Arduino IDE
50  // Pin 10 will be used as output pin for the Arduino IDE
51  // Pin 10 will be used as output pin for the Arduino IDE
52  // Pin 10 will be used as output pin for the Arduino IDE
53  // Pin 10 will be used as output pin for the Arduino IDE
54  // Pin 10 will be used as output pin for the Arduino IDE
55  // Pin 10 will be used as output pin for the Arduino IDE
56  // Pin 10 will be used as output pin for the Arduino IDE
57  // Pin 10 will be used as output pin for the Arduino IDE
58  // Pin 10 will be used as output pin for the Arduino IDE
59  // Pin 10 will be used as output pin for the Arduino IDE
60  // Pin 10 will be used as output pin for the Arduino IDE
61  // Pin 10 will be used as output pin for the Arduino IDE
62  // Pin 10 will be used as output pin for the Arduino IDE
63  // Pin 10 will be used as output pin for the Arduino IDE
64  // Pin 10 will be used as output pin for the Arduino IDE
65  // Pin 10 will be used as output pin for the Arduino IDE
66  // Pin 10 will be used as output pin for the Arduino IDE
67  // Pin 10 will be used as output pin for the Arduino IDE
68  // Pin 10 will be used as output pin for the Arduino IDE
69  // Pin 10 will be used as output pin for the Arduino IDE
70  // Pin 10 will be used as output pin for the Arduino IDE
71  // Pin 10 will be used as output pin for the Arduino IDE
72  // Pin 10 will be used as output pin for the Arduino IDE
73  // Pin 10 will be used as output pin for the Arduino IDE
74  // Pin 10 will be used as output pin for the Arduino IDE
75  // Pin 10 will be used as output pin for the Arduino IDE
76  // Pin 10 will be used as output pin for the Arduino IDE
77  // Pin 10 will be used as output pin for the Arduino IDE
78  // Pin 10 will be used as output pin for the Arduino IDE
79  // Pin 10 will be used as output pin for the Arduino IDE
80  // Pin 10 will be used as output pin for the Arduino IDE
81  // Pin 10 will be used as output pin for the Arduino IDE
82  // Pin 10 will be used as output pin for the Arduino IDE
83  // Pin 10 will be used as output pin for the Arduino IDE
84  // Pin 10 will be used as output pin for the Arduino IDE
85  // Pin 10 will be used as output pin for the Arduino IDE
86  // Pin 10 will be used as output pin for the Arduino IDE
87  // Pin 10 will be used as output pin for the Arduino IDE
88  // Pin 10 will be used as output pin for the Arduino IDE
89  // Pin 10 will be used as output pin for the Arduino IDE
90  // Pin 10 will be used as output pin for the Arduino IDE
91  // Pin 10 will be used as output pin for the Arduino IDE
92  // Pin 10 will be used as output pin for the Arduino IDE
93  // Pin 10 will be used as output pin for the Arduino IDE
94  // Pin 10 will be used as output pin for the Arduino IDE
95  // Pin 10 will be used as output pin for the Arduino IDE
96  // Pin 10 will be used as output pin for the Arduino IDE
97  // Pin 10 will be used as output pin for the Arduino IDE
98  // Pin 10 will be used as output pin for the Arduino IDE
99  // Pin 10 will be used as output pin for the Arduino IDE
100 // Pin 10 will be used as output pin for the Arduino IDE

```

รูปที่ 4-1 โปรแกรมค่าการเปลี่ยนแปลงในตัว Omron e6b2-cwz6c Encoder

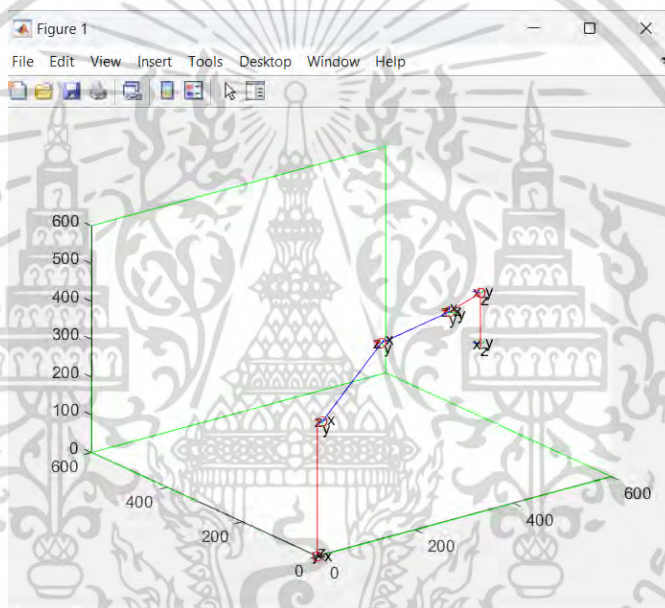


รูปที่ 4-2 การต่อวงจรเพื่อสังเกตค่าการเปลี่ยนแปลงในตัว Omron e6b2-cwz6c Encoder

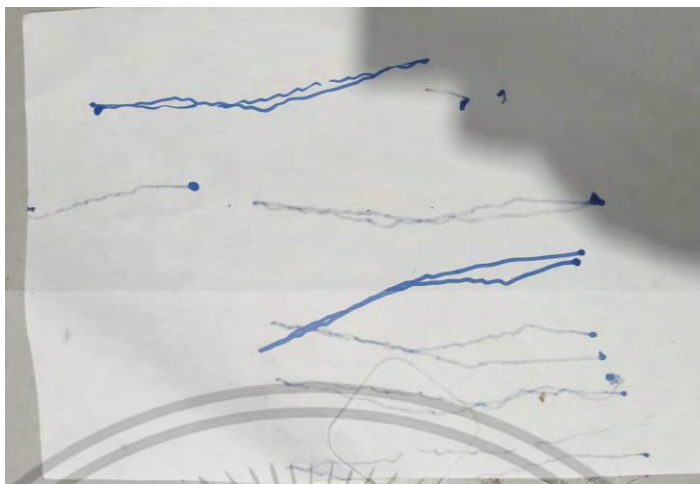
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองการเคลื่อนที่ของแขนกล

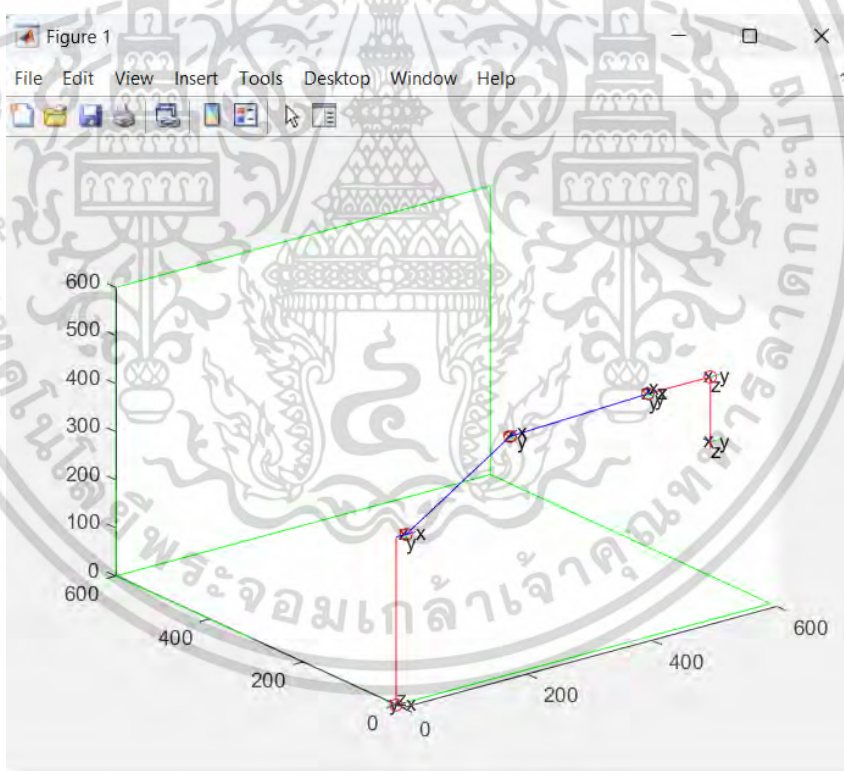
การทดลองการเคลื่อนที่ของแขนกลโดยใช้โปรแกรมการเคลื่อนที่ของแขนกลที่ได้จากระบบจำลองในโปรแกรม MATLAB มาใช้กับระบบจริงโดยใช้ค่าที่ได้จาก Omron e6b2-cwz6c Encoder มาคำนวณเพื่อเปรียบเทียบกัน ถ้าค่าที่ได้ไม่ตรงกับค่าจาก Omron e6b2-cwz6c Encoder ไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board (HCDVBD0033) จะสั่งให้มอเตอร์ทำงานเพื่อให้ค่าที่ได้จาก Omron e6b2-cwz6c Encoder ตรงกับค่าที่คำนวณจากโปรแกรม MATLAB



รูปที่ 4-3 รูปการณ์เคลื่อนที่เป็นเส้นตรงของแขนกลในระบบจำลอง

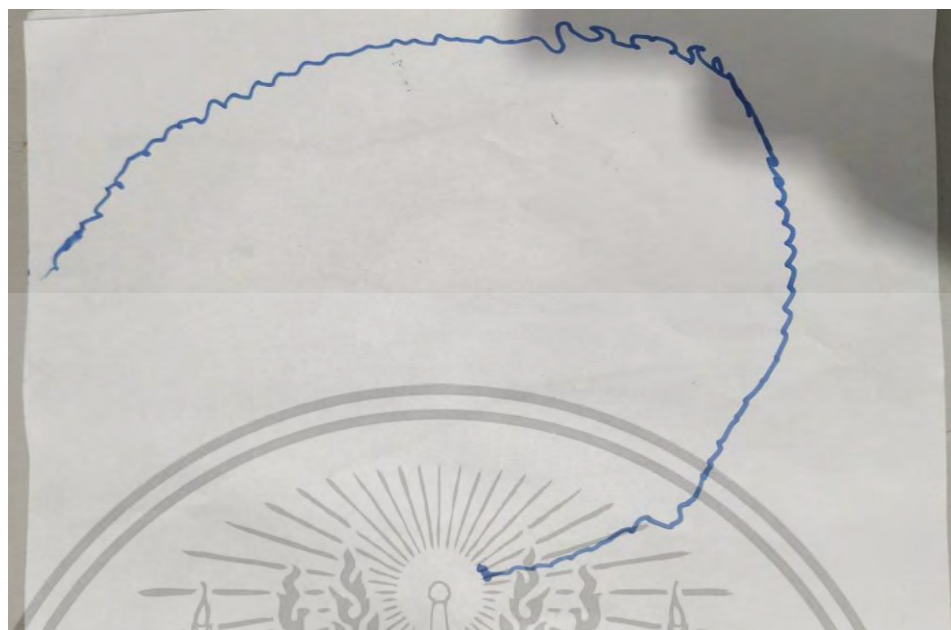


รูปที่ 4-4 รูปการเคลื่อนที่เป็นเส้นตรงของแขนกลในระบบจริง



รูปที่ 4-5 รูปการเคลื่อนที่เป็นเส้นวงกลมของแขนกลในระบบจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-6 รูปการเคลื่อนที่เป็นเส้นวงกลมของแขนกลในระบบจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิจารณ์ผลการทดลอง

5.1 สรุปผลการทดลอง

จากผลการทดลองการแปลงค่าจาก Omron e6b2-cwz6c Encoder พบว่า ค่าที่อ่านได้มีความแม่นยำเมื่อมีการเปลี่ยนแปลงของตัว Omron e6b2-cwz6c Encoder ส่วนการทดลองการเคลื่อนที่ของแขนกลพบว่าการเคลื่อนที่ของแขนกลในระบบจริงมีความคลาดเคลื่อนจากการเคลื่อนที่ของแขนกลในระบบจำลองอยู่มากและมีการสั่นคลอนจากการปรับค่าของ Omron e6b2-cwz6c Encoder

5.2 ปัญหาและอุปสรรคที่เกิดขึ้น

1. ปัญหาการรับค่าจาก Omron e6b2-cwz6c Encoder มีการสั่นคลอนตลอดเวลา ทำให้ค่าที่ได้มีโอกาสในการคลาดเคลื่อนจากระบบจริง
2. แขนกลมีน้ำหนักค่อนข้างมาก Series GM9000 LO-COG Brush Commutated DC Gear motors ไม่มีแรงพอที่จะรับน้ำหนักไหวเมื่อมีการเคลื่อนที่ที่ซับซ้อนหรือใช้ระยะเวลาานาน
3. แบบจำลองแขนกลในระบบจริงที่นำมาใช้มีอายุการใช้งานที่นานแล้ว ทำให้ข้อต่อในบางจุดไม่ประกบกัน มีผลทำให้การเคลื่อนที่ของแขนกลในระบบจริงไม่สอดคล้องกับการเคลื่อนที่ของแขนกลในระบบจำลอง

5.3 แนวทางในการแก้ไขปัญหาที่เกิดขึ้น

1. ทำการปรับค่าที่เกี่ยวข้องกับการสั่นคลอนในตัว Omron e6b2-cwz6c Encoder ให้อยู่ในสภาพที่สั่นคลอนน้อยที่สุด

2. เลือกใช้วัสดุในการทำโครงสร้างของแขนกลที่มีน้ำหนักเบาและทนทานมากขึ้น หรือ เลือกใช้มอเตอร์ที่มีกำลังในการขับเคลื่อนที่สูงขึ้น

3. ออกแบบจำลองแขนกลในระบบจริงที่มีรูปร่างมั่นคงและน้ำหนักเบา และเลือกใช้ข้อต่อที่ประกบพอดีเพื่อให้การเคลื่อนที่ของแขนกลในระบบจริงเสียดคล้องกับการเคลื่อนที่ของแขนกลในระบบจำลอง

5.4 ข้อเสนอแนะในการค้นคว้าและพัฒนาต่อ

แขนกลตัดโลหะด้วยพลาสมานี้เป็นต้นแบบจากการศึกษาพื้นฐานในการควบคุมการเคลื่อนที่ของแขนกล จึงสามารถสร้างสมการแขนกลในรูปแบบลักษณะอื่นๆได้ ซึ่งขึ้นอยู่กับรูปแบบลักษณะของแขนกลนั้นๆ หรือสามารถพัฒนาวัสดุอุปกรณ์ต่างในแขนกลก็ได้ เพื่อให้แขนกลทำงานได้อย่างมีประสิทธิภาพมากยิ่งขึ้น ทั้งนี้ยังคงจำเป็นที่จะต้องเขียนโปรแกรมเพื่อคำนึงถึงความปลอดภัย และพัฒนาให้แขนกลมีความสามารถในการควบคุมผ่านระบบสื่อสารทางไกลเพื่อที่จะควบคุมจากทุกที่ได้

เอกสารอ้างอิง

- [1] โศรฎา แซ่กการ, กนัฎร ชานึประศาสน์. การใช้งานMATLABสำหรับงานทางวิศวกรรม. [Online]. Available https://pirun.ku.ac.th/~fengslj/02212471/suppl1/matlab_thai.pdf
- [2] สุวิทย์ เมาะราคี. การโปรแกรมมึง Arduino ด้วย MATLAB & Simulink. กรุงเทพฯ: ซีเอ็ด ยูเคชัน, 2564.
- [3] สัฎฎา ลัษณะ. มอเตอรึไฟฟ้ากระแสตรง. [Online]. Available <https://www.prd.go.th/th/file/get/file/20210729d41d8cd98f00b204e9800998ecf8427e140258.pdf>
- [4] การใช้งาน STM32 ร่วมกับ Arduino เบื้องต้น. [Online]. Available <https://www.artronshop.co.th/article/43/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99-stm32-%E0%B8%A3%E0%B9%88%E0%B8%A7%E0%B8%A1%E0%B8%81%E0%B8%B1%E0%B8%9A-arduino-%E0%B9%80%E0%B8%9A%E0%B8%B7%E0%B9%89%E0%B8%AD%E0%B8%87%E0%B8%95%E0%B9%89%E0%B8%99>
- [5] เอ็นโคเดอรึ. [Online]. Available. http://www.kbmeng.com/media/wysiwyg/pdf/7_.pdf
- [6] Robert J. Schilling. **Fundamentals of Robotics Analysis & Control**. [Online]. Available https://kupdf.net/download/robert-j-schilling-fundamentals-of-robotics-pdf_5b0a1ee9e2b6f5d60be76100_pdf



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

เอกสารคู่มือแบบจำลองแขนกล SCORBOT-ER 5Plus

Specifications

The following table details the robot arm specifications.

SCORBOT-ER Vplus Specifications	
Mechanical Structure	Vertical articulated
Number of Axes	5 axes plus servo gripper
Axis Movement	
Axis 1: Base rotation	310°
Axis 2: Shoulder rotation	+130° / -35°
Axis 3: Elbow rotation	±130°
Axis 4: Wrist pitch	±130°
Axis 5: Wrist roll	Unlimited (mechanically); ±570° (electrically)
Maximum Operating Radius	610mm (24.4")
End Effector	DC servo gripper, with optical encoder, parallel finger motion; Measurement of object's size/gripping force by means of gripper sensor and software,
Maximum Gripper Opening	75 mm (3") without rubber pads 65 mm (2.6") with rubber pads
Hard Home	Fixed position on each axis, found by means of microswitches
Feedback	Optical encoder on each axis
Actuators	12VDC servo motors
Motor Capacity (axes 1-6)	15 oz. in Peak Torque (stall) 70 W Power for Peak Torque
Gear Ratios	Motors 1, 2, 3: 127.1:1 Motors 4, 5: 65.5:1 Motor 6 (gripper) 19.5:1
Transmission	Gears, timing belts, lead screw
Maximum Payload	1 kg (2.2 lb.), including gripper
Position Repeatability	±0.5 mm (0.02") at TCP (tip of gripper)
Weight	11.5 kg (25 lb)
Maximum Path Velocity	600 mm/sec (23.6"/sec)
Ambient Operating Temperature	2°-40°C (36°-104°F)

Structure

The **SCORBOT-ER Vplus** is a vertical articulated robot, with five revolute joints. With gripper attached, the robot has six degrees of freedom. This design permits the end effector to be positioned and oriented arbitrarily within a large work space.

Figures 2-2 and 2-3 identify the joints and links of the mechanical arm.

The movements of the joints are described in the following table:

Axis No.	Joint Name	Motion	Motor No.
1	Base	Rotates the body.	1
2	Shoulder	Raises and lowers the upper arm.	2
3	Elbow	Raises and lowers the forearm.	3
4	Wrist Pitch	Raises and lowers the end effector (gripper).	4+5
5	Wrist Roll	Rotates the end effector (gripper).	4+5

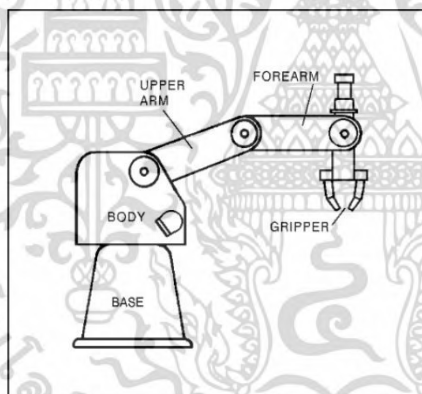


Figure 2-2: Robot Arm Links

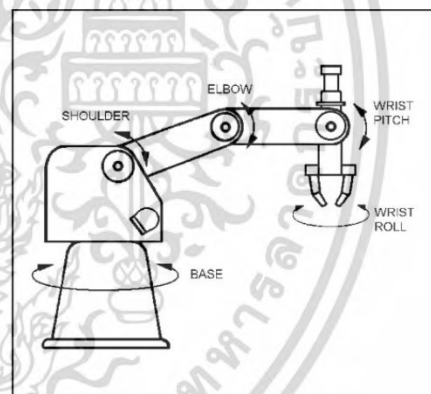


Figure 2-3: Robot Arm Joints

Work Envelope

The length of the links and the degree of rotation of the joints determine the robot's work envelope. Figures 2-4 and 2-5 show the dimensions and reach of the SCORBOT-ER Vplus.

The base of the robot is normally fixed to a stationary work surface. It may, however, be attached to a slidebase, resulting in an extended working range.

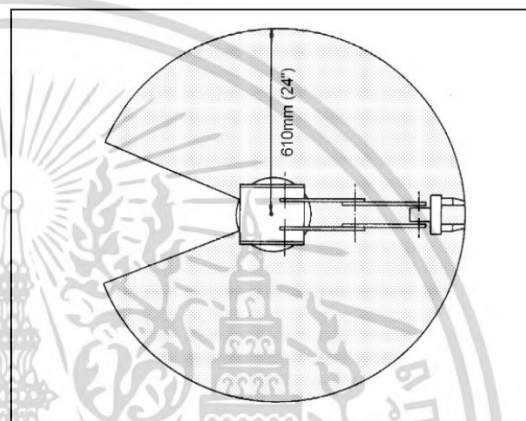


Figure 2-4: Operating Range (Top View)

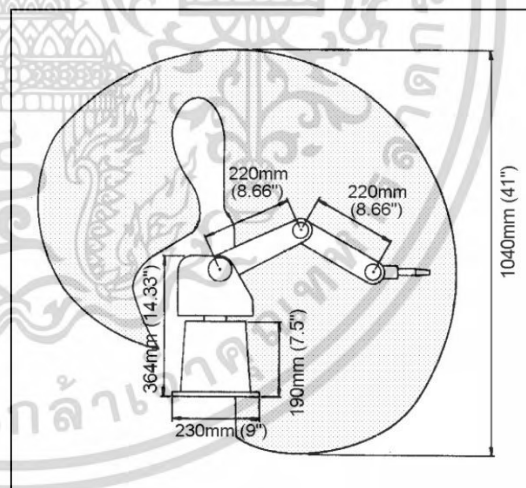


Figure 2-5: Operating Range (Side View)

Transmissions

Several kinds of transmissions are used to move the links of the robot arm.

- Spur gears move the base and shoulder axes.
- Pulleys and timing belts move the elbow axis.
- Pulleys and timing belts, and a bevel gear differential unit at the end of the arm move the wrist pitch and roll axes.
- A lead screw transmission opens and closes the gripper.

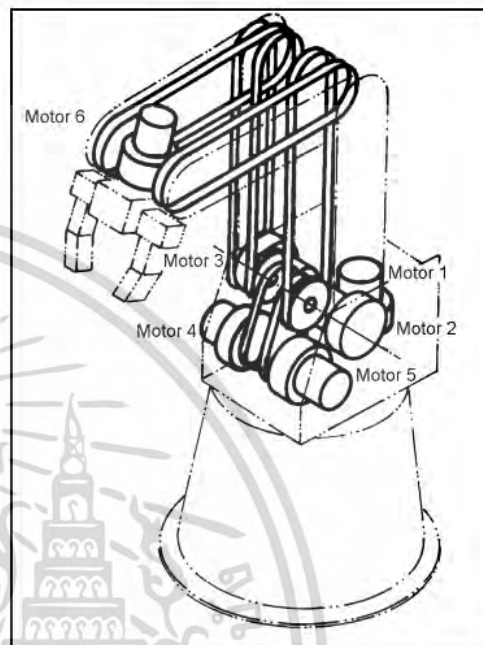


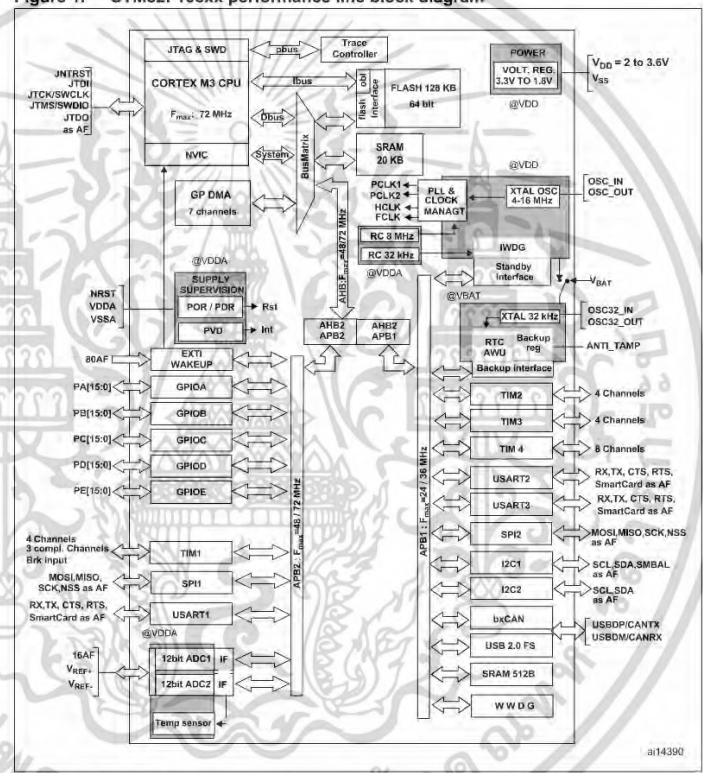
Figure 2-9: Transmissions

ภาคผนวก ข

เอกสารคู่มือไมโครคอนโทรลเลอร์ STM32F103 Black Pill Development Board

Description STM32F103xx

Figure 1. STM32F103xx performance line block diagram



1. $T_A = -40\text{ }^{\circ}\text{C}$ to $+105\text{ }^{\circ}\text{C}$ (junction temperature up to $125\text{ }^{\circ}\text{C}$).
2. AF = alternate function on I/O port pin.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 Pin descriptions

Figure 2. STM32F103xx performance line LQFP100 pinout

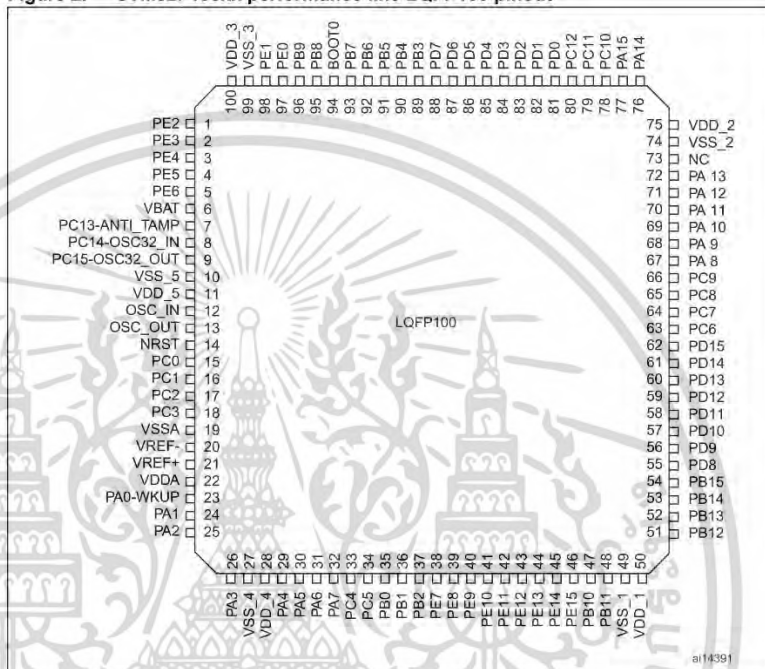


Table 3. Pin definitions (continued)

Pins				Pin name	Type ⁽¹⁾	I/O Level ⁽²⁾	Main function ⁽³⁾ (after reset)	Default alternate functions
BGA100	LQFP48	LQFP64	LQFP100					
G3	14	20	29	PA4/SPI1_NSS/ USART2_CK/ADC_IN4	I/O		PA4	SPI1_NSS ⁽⁶⁾ / USART2_CK ⁽⁶⁾ / ADC_IN4
H3	15	21	30	PA5/SPI1_SCK/ ADC_IN5	I/O		PA5	SPI1_SCK ⁽⁶⁾ / ADC_IN5
J3	16	22	31	PA6/SPI1_MISO/ ADC_IN6/TIM3_CH1	I/O		PA6	SPI1_MISO ⁽⁶⁾ / ADC_IN6/TIM3_CH1 ⁽⁶⁾
K3	17	23	32	PA7/SPI1_MOSI/ ADC_IN7/TIM3_CH2	I/O		PA7	SPI1_MOSI ⁽⁶⁾ / ADC_IN7/TIM3_CH2 ⁽⁶⁾
G4	-	24	33	PC4/ADC_IN14	I/O		PC4	ADC_IN14
H4	-	25	34	PC5/ADC_IN15	I/O		PC5	ADC_IN15
J4	18	26	35	PB0/ADC_IN8/ TIM3_CH3	I/O		PB0	ADC_IN8/TIM3_CH3 ⁽⁶⁾
K4	19	27	36	PB1/ADC_IN9/ TIM3_CH4	I/O		PB1	ADC_IN9/TIM3_CH4 ⁽⁶⁾
G5	20	28	37	PB2 / BOOT1	I/O	FT	PB2/BOOT1	
H5	-	-	38	PE7	I/O	FT	PE7	
J5	-	-	39	PE8	I/O	FT	PE8	
K5	-	-	40	PE9	I/O	FT	PE9	
G6	-	-	41	PE10	I/O	FT	PE10	
H6	-	-	42	PE11	I/O	FT	PE11	
J6	-	-	43	PE12	I/O	FT	PE12	
K6	-	-	44	PE13	I/O	FT	PE13	
G7	-	-	45	PE14	I/O	FT	PE14	
H7	-	-	46	PE15	I/O	FT	PE15	
J7	21	29	47	PB10/I2C2_SCL/ USART3_TX	I/O	FT	PB10	I2C2_SCL/USART3_TX ⁽⁵⁾⁽⁶⁾
K7	22	30	48	PB11/I2C2_SDA/ USART3_RX	I/O	FT	PB11	I2C2_SDA/ USART3_RX ⁽⁵⁾⁽⁶⁾
E7	23	31	49	V _{SS_1}	S		V _{SS_1}	
F7	24	32	50	V _{DD_1}	S		V _{DD_1}	
K8	25	33	51	PB12/SPI2_NSS / I2C2_SMBAL/ USART3_CK / TIM1_BKIN	I/O	FT	PB12	SPI2_NSS ⁽⁵⁾ / I2C2_SMBAL ⁽⁵⁾ / USART3_CK ⁽⁵⁾⁽⁶⁾ / TIM1_BKIN ⁽⁶⁾
J8	26	34	52	PB13/SPI2_SCK / USART3_CTS / TIM1_CH1N	I/O	FT	PB13	SPI2_SCK ⁽⁵⁾ / USART3_CTS ⁽⁵⁾⁽⁶⁾ / TIM1_CH1N ⁽⁶⁾
H8	27	35	53	PB14/SPI2_MISO / USART3_RTS / TIM1_CH2N	I/O	FT	PB14	SPI2_MISO ⁽⁵⁾ / USART3_RTS ⁽⁵⁾⁽⁶⁾ / TIM1_CH2N ⁽⁶⁾

Pin descriptions

STM32F103xx

Table 3. Pin definitions (continued)

BGA100	Pins			Pin name	Type ⁽¹⁾	I/O Level ⁽²⁾	Main function ⁽³⁾ (after reset)	Default alternate functions
	LQFP48	LQFP64	LQFP100					
G8	28	36	54	PB15/SPI2_MOSI/ TIM1_CH3N	I/O	FT	PB15	SPI2_MOSI ⁽⁵⁾ / TIM1_CH3N ⁽⁶⁾
K9	-	-	55	PD8	I/O	FT	PD8	
J9	-	-	56	PD9	I/O	FT	PD9	
H9	-	-	57	PD10	I/O	FT	PD10	
G9	-	-	58	PD11	I/O	FT	PD11	
K10	-	-	59	PD12	I/O	FT	PD12	
J10	-	-	60	PD13	I/O	FT	PD13	
H10	-	-	61	PD14	I/O	FT	PD14	
G10	-	-	62	PD15	I/O	FT	PD15	
F10	-	37	63	PC6	I/O	FT	PC6	
E10	-	38	64	PC7	I/O	FT	PC7	
F9	-	39	65	PC8	I/O	FT	PC8	
E9	-	40	66	PC9	I/O	FT	PC9	
D9	29	41	67	PA8/USART1_CK/ TIM1_CH1/MCO	I/O	FT	PA8	USART1_CK/ TIM1_CH1 ⁽⁶⁾ /MCO
C9	30	42	68	PA9/USART1_TX/ TIM1_CH2	I/O	FT	PA9	USART1_TX ⁽⁶⁾ / TIM1_CH2 ⁽⁶⁾
D10	31	43	69	PA10/USART1_RX/ TIM1_CH3	I/O	FT	PA10	USART1_RX ⁽⁶⁾ / TIM1_CH3 ⁽⁶⁾
C10	32	44	70	PA11 / USART1_CTS/ CANRX / USBDM/ TIM1_CH4	I/O	FT	PA11	USART1_CTS/ CANRX ⁽⁶⁾ / TIM1_CH4 ⁽⁶⁾ / USBDM
B10	33	45	71	PA12 / USART1_RTS/ CANTX / USBDP/ TIM1_ETR	I/O	FT	PA12	USART1_RTS/ CANTX ⁽⁶⁾ / TIM1_ETR ⁽⁶⁾ / USBDP
A10	34	46	72	PA13/JTMS/SWDIO	I/O	FT	JTMS/SWDIO	PA13
F8	-	-	73				Not connected	
E6	35	47	74	V _{SS_2}	S		V _{SS_2}	
F6	36	48	75	V _{DD_2}	S		V _{DD_2}	
A9	37	49	76	PA14/JTCK/SWCLK	I/O	FT	JTCK/SWCLK	PA14
A8	38	50	77	PA15/JTDI	I/O	FT	JTDI	PA15
B9	-	51	78	PC10	I/O	FT	PC10	
B8	-	52	79	PC11	I/O	FT	PC11	
C8	-	53	80	PC12	I/O	FT	PC12	



Table 3. Pin definitions (continued)

Pins				Pin name	Type ⁽¹⁾	I/O Level ⁽²⁾	Main function ⁽³⁾ (after reset)	Default alternate functions
BGA100	LQFP48	LQFP64	LQFP100					
D8	5	5	81	PD0	I/O	FT	OSC_IN ⁽⁷⁾	
E8	6	6	82	PD1	I/O	FT	OSC_OUT ⁽⁷⁾	
B7	-	54	83	PD2/TIM3_ETR	I/O	FT	PD2	TIM3_ETR
C7	-	-	84	PD3	I/O	FT	PD3	
D7	-	-	85	PD4	I/O	FT	PD4	
B6	-	-	86	PD5	I/O	FT	PD5	
C6	-	-	87	PD6	I/O	FT	PD6	
D6	-	-	88	PD7	I/O	FT	PD7	
A7	39	55	89	PB3/JTDO/TRACESWO	I/O	FT	JTDO	PB3/TRACESWO
A6	40	56	90	PB4/JNTRST	I/O	FT	JNTRST	PB4
C5	41	57	91	PB5/I2C1_SMBAL	I/O		PB5	I2C1_SMBAL
B5	42	58	92	PB6/I2C1_SCL/ TIM4_CH1	I/O	FT	PB6	I2C1_SCL ⁽⁶⁾ / TIM4_CH1 ⁽⁵⁾⁽⁶⁾
A5	43	59	93	PB7/I2C1_SDA/ TIM4_CH2	I/O	FT	PB7	I2C1_SDA ⁽⁶⁾ / TIM4_CH2 ⁽⁵⁾⁽⁶⁾
D5	44	60	94	BOOT0	I		BOOT0	
B4	45	61	95	PB8/TIM4_CH3	I/O	FT	PB8	TIM4_CH3 ⁽⁵⁾⁽⁶⁾
A4	46	62	96	PB9/TIM4_CH4	I/O	FT	PB9	TIM4_CH4 ⁽⁵⁾⁽⁶⁾
D4	-	-	97	PE0/TIM4_ETR	I/O	FT	PE0	TIM4_ETR ⁽⁵⁾
C4	-	-	98	PE1	I/O	FT	PE1	
E5	47	63	99	V _{SS_3}	S		V _{SS_3}	
F5	48	64	100	V _{DD_3}	S		V _{DD_3}	

1. I = input, O = output, S = supply, HiZ = high impedance.

2. FT = 5 V tolerant.

3. Function availability depends on the chosen device. Refer to *Table 2 on page 7*.

4. PC13, PC14 and PC15 are supplied through the power switch, and so their use in output mode is limited: they can be used only in output 2 MHz mode with a maximum load of 30 pF and only one pin can be put in output mode at a time.

5. Available only on devices with a Flash memory density equal or higher than 64 Kbytes.

6. This alternate function can be remapped by software to some other port pins (if available on the used package). For more details, refer to the Alternate function I/O and debug configuration section in the STM32F10xxx reference manual, UM0306, available from the STMicroelectronics website: www.st.com.

7. For the LQFP48 and LQFP64 packages, the pins number 5 and 6 are configured as OSC_IN/OSC_OUT after reset, however the functionality of PD0 and PD1 can be remapped by software on these pins.

5.2 Absolute maximum ratings

Stresses above the absolute maximum ratings listed in [Table 4: Voltage characteristics](#), [Table 5: Current characteristics](#), and [Table 6: Thermal characteristics](#) may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Table 4. Voltage characteristics

Symbol	Ratings	Min	Max	Unit
$V_{DD}-V_{SS}$	External 3.3 V supply voltage (including V_{DDA} and V_{DD}) ⁽¹⁾	-0.3	4.0	V
V_{IN}	Input voltage on five volt tolerant pin ⁽²⁾	$V_{SS}-0.3$	+5.5	
	Input voltage on any other pin ⁽²⁾	$V_{SS}-0.3$	$V_{DD}+0.3$	
$ \Delta V_{DDx} $	Variations between different power pins	50	50	mV
$ V_{SSx}-V_{SS} $	Variations between all the different ground pins	50	50	
$V_{ESD(HBM)}$	Electrostatic discharge voltage (human body model)	see Section 5.3.11: Absolute maximum ratings (electrical sensitivity)		

- All 3.3 V power (V_{DD} , V_{DDA}) and ground (V_{SS} , V_{SSA}) pins must always be connected to the external 3.3 V supply.
- $I_{NJ(PIN)}$ must never be exceeded (see [Table 5: Current characteristics](#)). This is implicitly insured if V_{IN} maximum is respected. If V_{IN} maximum cannot be respected, the injection current must be limited externally to the $I_{NJ(PIN)}$ value. A positive injection is induced by $V_{IN} > V_{DD}$ while a negative injection is induced by $V_{IN} < V_{SS}$.

Table 5. Current characteristics

Symbol	Ratings	Max.	Unit
I_{VDD}	Total current into V_{DD} power lines (source) ⁽¹⁾	150	mA
I_{VSS}	Total current out of V_{SS} ground lines (sink) ⁽¹⁾	150	
I_{IO}	Output current sunk by any I/O and control pin	25	
	Output current source by any I/Os and control pin	-25	
$I_{NJ(PIN)}$ ⁽²⁾⁽³⁾	Injected current on NRST pin	± 5	
	Injected current on HSE OSC_IN and LSE OSC_IN pins	± 5	
	Injected current on any other pin ⁽⁴⁾	± 5	
$\Sigma I_{NJ(PIN)}$ ⁽²⁾	Total injected current (sum of all I/O and control pins) ⁽⁴⁾	± 25	

- All 3.3 V power (V_{DD} , V_{DDA}) and ground (V_{SS} , V_{SSA}) pins must always be connected to the external 3.3 V supply.
- $I_{NJ(PIN)}$ must never be exceeded. This is implicitly insured if V_{IN} maximum is respected. If V_{IN} maximum cannot be respected, the injection current must be limited externally to the $I_{NJ(PIN)}$ value. A positive injection is induced by $V_{IN} > V_{DD}$ while a negative injection is induced by $V_{IN} < V_{SS}$.
- Negative injection disturbs the analog performance of the device. See note in [Section 5.3.17: 12-bit ADC characteristics](#).
- When several inputs are submitted to a current injection, the maximum $\Sigma I_{NJ(PIN)}$ is the absolute sum of the positive and negative injected currents (instantaneous values). These results are based on characterization with $\Sigma I_{NJ(PIN)}$ maximum current injection on four I/O port pins of the device.

ภาคผนวก ค

เอกสารคู่มือ MDD10A Dual Channel 10A DC Motor Driver

ROBOT . HEAD to TOE
Product User's Manual – [MDD10A](#)

3. PRODUCT SPECIFICATION AND LIMITATIONS

Absolute Maximum Rating

No.	Parameters	Min	Typical	Max	Unit
1	Power Input Voltage***	5	-	30	V
2	I_{MAX} (Maximum Continuous Motor Current)*	-	-	10	A
3	I_{PEAK} – (Peak Motor Current) **	-	-	30	A
4	V_{IOH} (Logic Input – High Level)	3	-	5.5	V
5	V_{IOL} (Logic Input – Low Level)	0	0	0.5	V
6	Maximum PWM Frequency	-	-	20	KHz

* Tested in room temperature at 25°C

** Must not exceed 10 seconds.

*** Rev2.0 has upgraded to support 30V maximum power input voltage

4. DIMENSION

6. Input

Pin No.	Pin Name	Description
1	GND	Ground
2	*PWM2	PWM input for speed control (Motor 2)
3	DIR2	Direction input (Motor 2)
4	*PWM1	PWM input for speed control (Motor 1)
5	DIR1	Direction input (Motor 1)

*Note that it is not for RC PWM

The truth table for the control logic for motor 1 and motor 2 are as follow:

PWM	DIR	Output A	Output B
Low	X(Don't care)	Low	Low
High	Low	High	Low
High	High	Low	High

7. Test Button M2A – When this button is pressed, current flows from output M2A to M2B and motor will turn CW (or CCW depending on the connection).
8. Test Button M2B – When this button is pressed, current flows from output M2B to M2A and motor will turn CCW (or CW depending on the connection).
9. Red LED M2B – Turns on when the output M2A is low and output M2B is high. Indicates the current flows from output M2B to M2A.
10. Red LED M2A – Turns on when the output M2A is high and output M2B is low. Indicates the current flows from output M2A to M2B.
11. Terminal Block – Connect to motor and power source.

Pin No	Pin Name	Description
1	Motor 1 Output B	Connect to motor 1 terminal B
2	Motor 1 Output A	Connect to motor 1 terminal A
3	POWER +	Positive Supply (positive terminal of battery)
4	POWER -	Negative Supply (negative terminal of battery)
5	Motor 2 Output A	Connect to motor 2 terminal A
6	Motor 2 Output B	Connect to motor 2 terminal B

ภาคผนวก ง

เอกสารคู่มือ Omron e6b2-cwz6c Encoder

Incremental 40-mm-dia. Rotary Encoder
E6B2-C

General-purpose Encoder with External Diameter of 40 mm

- Incremental model
- External diameter of 40 mm.
- Resolution of up to 2,000 ppr.





Ordering Information

Encoders [Refer to *Dimensions* on page 4.]

Power supply voltage	Output configuration	Resolution (pulses/rotation)	Model
5 to 24 VDC	NPN open-collector output	10, 20, 30, 40, 50, 60, 100, 200, 300, 360, 400, 500, 600	E6B2-CWZ6C (resolution) 0.5M Example: E6B2-CWZ6C 10P/R 0.5M
		720, 800, 1,000, 1,024 1,200, 1,500, 1,800, 2,000	
12 to 24 VDC	PNP open-collector output	100, 200, 360, 500, 600	E6B2-CWZ5B (resolution) 0.5M Example: E6B2-CWZ5B 100P/R 0.5M
		1,000 2,000	
5 to 12 VDC	Voltage output	10, 20, 30, 40, 50, 60, 100, 200, 300, 360, 400, 500, 600	E6B2-CWZ3E (resolution) 0.5M Example: E6B2-CWZ3E 10P/R 0.5M
		1,000 1,200, 1,500, 1,800, 2,000	
5 VDC	Line-driver output	10, 20, 30, 40, 50, 60, 100, 200, 300, 360, 400, 500, 600	E6B2-CWZ1X (resolution) 0.5M Example: E6B2-CWZ1X 10P/R 0.5M
		1,000, 1,024 1,200, 1,500, 1,800, 2,000	

Accessories (Order Separately) [Refer to *Dimensions* on *Rotary Encoder Accessories*.]

Name	Model	Remarks
Couplings	E69-C06B	Provided with the product.
	E69-C68B	Different end diameter
	E69-C610B	Different end diameter
	E69-C06M	Metal construction
Flanges	E69-FBA	---
	E69-FBA02	E69-2 Servo Mounting Bracket provided.
Servo Mounting Bracket	E69-2	---

Refer to *Accessories* for details.

E6B2-C

Ratings and Specifications

Item	Model	E6B2-CWZ6C	E6B2-CWZ5B	E6B2-CWZ3E	E6B2-CWZ1X
Power supply voltage		5 VDC -5% to 24 VDC +15%, ripple (p-p): 5% max.	12 VDC -10% to 24 VDC +15%, ripple (p-p): 5% max.	5 VDC -5% to 12 VDC +10%, ripple (p-p): 5% max.	5 VDC ±5%, ripple (p-p): 5% max.
Current consumption *1		80 mA max.	100 mA max.		160 mA max.
Resolution (pulses/rotation)		10, 20, 30, 40, 50, 60, 100, 200, 300, 360, 400, 500, 600, 720, 800, 1,000, 1,024, 1,200, 1,500, 1,800, 2,000	100, 200, 360, 500, 600, 1,000, 2,000	10, 20, 30, 40, 50, 60, 100, 200, 300, 360, 400, 500, 600, 1,000, 1,200, 1,500, 1,800, 2,000	10, 20, 30, 40, 50, 60, 100, 200, 300, 360, 400, 500, 600, 1,000, 1,024, 1,200, 1,500, 1,800, 2,000
Output phases		Phases A, B, and Z			Phases A, \bar{A} , B, \bar{B} , Z, and \bar{Z}
Phase difference between outputs		90°±45° between A and B (1/4 T ± 1/8 T)			
Output configuration		NPN open-collector output	PNP open-collector output	Voltage output (NPN output)	Line driver output *2
Output capacity		Applied voltage: 30 VDC max. Sink current: 35 mA max. Residual voltage: 0.4 V max. (at sink current of 35 mA)	Applied voltage: 30 VDC max. Source current: 35 mA max. Residual voltage: 0.4 V max. (at source current of 35 mA)	Output resistance: 2 kΩ Sink current: 20 mA max. Residual voltage: 0.4 V max. (at sink current of 20 mA)	AM26LS31 equivalent Output current High level: I _o = -20 mA Low level: I _s = 20 mA Output voltage: V _o = 2.5 V min. V _s = 0.5 V max.
Maximum response frequency *3		100 kHz	50 kHz	100 kHz	
Rise and fall times of output		1 μs max. (Control output voltage: 5 V, Load resistance: 1 kΩ, Cable length: 2 m max.)	1 μs max. (Cable length: 2 m max., Sink current: 10 mA)		0.1 μs max. (Cable length: 2 m max., I _o = -20 mA, I _s = 20 mA)
Starting torque		0.98 mN·m max.			
Moment of inertia		1×10 ⁻⁶ kg·m ² max.; 3×10 ⁻⁷ kg·m ² max. at 600 P/R max.			
Shaft load-ing	Radial	30 N			
	Thrust	20 N			
Maximum permissible speed		6,000 r/min			
Protection circuits		Power supply reverse polarity protection, Load short-circuit protection			
Ambient temperature range		Operating: -10 to 70°C (with no icing), Storage: -25 to 85°C (with no icing)			
Ambient humidity range		Operating/Storage: 35% to 85% (with no condensation)			
Insulation resistance		20 MΩ min. (at 500 VDC) between current-carrying parts and case			
Dielectric strength		500 VAC, 50/60 Hz for 1 min between current-carrying parts and case			
Vibration resistance		Destruction: 10 to 500 Hz, 150 m/s ² or 2-mm double amplitude for 11 min 3 times each in X, Y, and Z directions			
Shock resistance		Destruction: 1,000m/s ² 3 times each in X, Y, and Z directions			
Degree of protection		IEC 60529 IP50			
Connection method		Pre-wired Models (Standard cable length: 500 mm)			
Materials		Case: ABS, Main unit: Aluminum, Shaft: SUS420J2			
Weight (packed state)		Approx. 100 g			
Accessories		Coupling, Hexagonal wrench, Instruction manual			

*1. An inrush current of approximately 9 A will flow for approximately 0.3 ms when the power is turned ON.

*2. The line driver output is a data transmission circuit compatible with RS-422A and long-distance transmission is possible with a twisted-pair cable. The quality is equivalent to AM26LS31.

*3. The maximum electrical response speed is determined by the resolution and maximum response frequency as follows:

$$\text{Maximum electrical response speed (rpm)} = \frac{\text{Maximum response frequency}}{\text{Resolution}} \times 60$$

This means that the E6B2-C Rotary Encoder will not operate electrically if its speed exceeds the maximum electrical response speed.

E6B2-C

I/O Circuit Diagrams

Model/Output Circuits	Output mode	Connection																		
<p>E6B2-CWZ6C</p> <p>E6B2-CWZ5B</p>	<p>E6B2-CWZ6C NPN Open-collector Output Model E6B2-CWZ5B PNP Open-collector Output Model</p> <p>Direction of rotation: CW (as viewed from end of shaft) Direction of rotation: CCW (as viewed from end of shaft)</p> <p>Note: Phase A is $1/4 T \pm 1/8 T$ faster than phase B. The ONs in the above timing chart mean that the output transistor is ON and the OFFs mean that the output transistor is OFF.</p>	<table border="1"> <thead> <tr> <th>Color</th> <th>Terminal</th> </tr> </thead> <tbody> <tr> <td>Brown</td> <td>Power supply (+Vcc)</td> </tr> <tr> <td>Black</td> <td>Output phase A</td> </tr> <tr> <td>White</td> <td>Output phase B</td> </tr> <tr> <td>Orange</td> <td>Output phase Z</td> </tr> <tr> <td>Blue</td> <td>0 V (common)</td> </tr> </tbody> </table>	Color	Terminal	Brown	Power supply (+Vcc)	Black	Output phase A	White	Output phase B	Orange	Output phase Z	Blue	0 V (common)						
Color	Terminal																			
Brown	Power supply (+Vcc)																			
Black	Output phase A																			
White	Output phase B																			
Orange	Output phase Z																			
Blue	0 V (common)																			
<p>E6B2-CWZ3E</p>	<p>E6B2-CWZ3E Voltage Output Model</p> <p>Direction of rotation: CW (as viewed from end of shaft) Direction of rotation: CCW (as viewed from end of shaft)</p> <p>Note: Phase A is $1/4 T \pm 1/8 T$ faster than phase B. Note: Phase A is $1/4 T \pm 1/8 T$ slower than phase B.</p>																			
<p>E6B2-CWZ1X</p>	<p>E6B2-CWZ1X Line Driver Output Model</p> <p>Direction of rotation: CW (as viewed from end of shaft) Direction of rotation: CCW (as viewed from end of shaft)</p>	<table border="1"> <thead> <tr> <th>Color</th> <th>Terminal</th> </tr> </thead> <tbody> <tr> <td>Brown</td> <td>Power supply (+Vcc)</td> </tr> <tr> <td>Black</td> <td>Output phase A</td> </tr> <tr> <td>Black/red stripes</td> <td>Output phase A</td> </tr> <tr> <td>White</td> <td>Output phase B</td> </tr> <tr> <td>White/red stripes</td> <td>Output phase B</td> </tr> <tr> <td>Orange</td> <td>Output phase Z</td> </tr> <tr> <td>Orange/red stripes</td> <td>Output phase Z</td> </tr> <tr> <td>Blue</td> <td>0 V (common)</td> </tr> </tbody> </table> <p>Note: Receiver: AM26LS32 equivalent</p>	Color	Terminal	Brown	Power supply (+Vcc)	Black	Output phase A	Black/red stripes	Output phase A	White	Output phase B	White/red stripes	Output phase B	Orange	Output phase Z	Orange/red stripes	Output phase Z	Blue	0 V (common)
Color	Terminal																			
Brown	Power supply (+Vcc)																			
Black	Output phase A																			
Black/red stripes	Output phase A																			
White	Output phase B																			
White/red stripes	Output phase B																			
Orange	Output phase Z																			
Orange/red stripes	Output phase Z																			
Blue	0 V (common)																			

1. The shielded cable outer core (shield) is not connected to the inner area or to the case.
2. The phase A, phase B, and phase Z circuits are all identical.
3. Normally, connect GND to 0 V or to an external ground.

ประวัติผู้เขียน



นายพิชชากร บุญจันทร

เกิดวันที่ : 10 สิงหาคม พ.ศ.2543

ที่อยู่ : 22/20 หมู่ที่8 แขวงคูฝั่งเหนือ เขตหนองจอก จ.กรุงเทพมหานคร
10530

โทรศัพท์ : 098-270-7731

อีเมล : pitchakorn043@gmail.com

ประวัติการศึกษา

2559 – 2562 โรงเรียนเตรียมอุดมศึกษาน้อมเกล้า

สายการเรียน วิทยาศาสตร์ - คณิตศาสตร์

2562 - 2566

สาขาวิชาวิศวกรรมแมคคาทรอนิกส์ ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบัง

ฝึกงาน : Toyota Daihatsu Engineering & Manufacturing Co, Ltd
(มิถุนายน 2565 - กรกฎาคม 2565)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



นายสหรัถ ฉิมดอนทอง

เกิดวันที่ : 8 มกราคม พ.ศ.2544

ที่อยู่ : 85 ถ.ปิ่น แขวงสีลม เขตบางรัก จ.กรุงเทพมหานคร 10500

โทรศัพท์ : 085-384-9312

อีเมล : saharad.ch@gmail.com

ประวัติการศึกษา

2559 – 2562 โรงเรียนวัดสุทธิวราราม

สายการเรียน วิทยาศาสตร์ - คณิตศาสตร์

2562 - 2566

สาขาวิชาวิศวกรรมแมคคาทรอนิกส์ ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบัง

ฝึกงาน : บริษัท โลว์ อินสทรูเมนต์ จำกัด (มิถุนายน 2565 - กรกฎาคม 2565)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้