

RMXplorer

LAB2 : Actuator Report

สมาชิก

● นางสาวดีษย์ธร	สุทธาราเวศ	66340500019
● นายภาณุ	เอื้อสุตกิจ	66340500040
● นายภูริวัฒ	เกษมสุขไพบูล	66340500044

จุดประสงค์

- เพื่อศึกษาการทดลองด้วยกระบวนการทางวิทยาศาสตร์ ในการสืบเสาะพฤติกรรม ปรากฏการณ์ ทดลอง บันทึกผล การทดลอง สรุปผล และอภิปรายผลการทดลอง เพื่อเข้าใจหลักการทำงานของมอเตอร์ และอุปกรณ์ทั้งหมดที่เกี่ยวข้องได้ ตลอดจนใช้หลักการทำงานทางวิทยาศาสตร์ คณิตศาสตร์ สถิ และศาสตร์อื่นๆ ที่เกี่ยวข้อง ประยุกต์ใช้ร่วมกับโปรแกรม MATLAB เพื่อเก็บผลการทดลอง วิเคราะห์ผลการทดลอง วิเคราะห์ความเที่ยงตรง ความแม่นยำ ได้อย่างถูกต้อง และมีเหตุผลรองรับ ตรวจสอบความถูกต้องเทียบกับทฤษฎีที่น่าเชื่อถือ
- เพื่อศึกษาการรับรู้ปริมาณทางฟิสิกส์ของเซนเซอร์ทั้งหมดตั้งแต่ต้นจนจบกระบวนการ เช่น การอธิบายการวัด ความเร็วของ Brushless DC Motor จาก Hall Sensor (จากคลื่นสัญญาณไฟฟ้าแปลงเป็นความเร็วได้อย่างไร)
- เพื่อศึกษาการกำหนดตัวแปรในการทดลองอย่างถูกต้องและสมเหตุสมผล ไม่ว่าจะเป็น ตัวแปรต้น ตัวแปรตาม ตัวแปรควบคุม อธิบายจุดประสงค์การทดลอง อธิบายสมมติฐานให้สอดคล้องกับตัวแปรที่กำหนด นิยามเชิงปฏิบัติการ และมีทฤษฎีที่น่าเชื่อถือรองรับ เช่น ทฤษฎีทางฟิสิกส์ หรือข้อมูลจาก Datasheet
- เพื่อศึกษาวิธีออกแบบการทดลองเพื่อหาคำตอบ ให้สอดคล้องกับผลการเรียนรู้ย่อทั้งหมดได้ด้วยตนเองอย่างถูกต้องตามหลักวิทยาศาสตร์ รวมทั้ง บันทึกผล สรุปผล อภิปรายผล ตามข้อมูลที่บันทึกได้จริง มีกระบวนการทำข้อ อธิบายที่มากของผลการทดลองนั้นได้ โดยใช้อุปกรณ์ เครื่องมือ ไฟล์ Simulink, mlx ฯลฯ และชุดการทดลองพร้อมบอร์ด Microcontroller ที่ TA จัดเตรียมให้เบื้องต้น
- นักศึกษาต้องสามารถเขียนรายงาน สัญลักษณ์ และสมการทางวิทยาศาสตร์ ได้อย่างถูกต้อง ทั้งขนาด และรูปแบบ อักษร การเว้นช่องไฟ การเว้นขอบกระดาษ การเว้นระยะพิมพ์ ให้ได้ระยะที่เหมาะสมตามหลักสากล

การทดลองที่ 1

Brushed DC Motor

1.1 จุดประสงค์

- 1) เพื่อศึกษาหลักการทำงานของ DC Motor และความสามารถของ Motor-Torque Constant ของ DC Motor
- 2) เพื่อศึกษาความสัมพันธ์ของ Speed, Torque, Current, Power, %Efficiency เมื่อ Dynamic Load ที่กระทำต่อ DC Motor ผ่านการปรับ Torque ของ Dynamic Load เป็นไป และแรงดันไฟฟ้า Input จากการปรับกระแสไฟฟ้าของ DC Motor ณ Dynamic Load ที่ต่างกัน
- 3) เพื่อศึกษาลักษณะเฉพาะของมอเตอร์ (Motor Characteristics) ที่ได้จากการใช้มอเตอร์แบบแรงบิดไดนามิก (Dynamic Torque Motor) ในการสร้างโหลดที่แตกต่างกัน
- 4) เพื่อศึกษาความสัมพันธ์ระหว่างแรงดันไฟฟ้าและความเร็วของ DC Motor
- 5) เพื่อศึกษาคุณลักษณะเฉพาะของมอเตอร์ผ่านความสัมพันธ์ของ Speed, Torque, Power, %Efficiency และ Current ทั้งในเงื่อนไขแบบ No Load และ Full Load
- 6) เพื่อศึกษาหลักการทำงาน H-Bridge Drive Mode และการควบคุม DC Motor
- 7) เพื่อศึกษาระบบการ Signal Conditioning และ Signal Processing
- 8) เพื่อศึกษาการประยุกต์ใช้ MATLAB และ Simulink ในการสั่งการหรือรับค่าร่วมกับบอร์ด Nucleo STM32G474RE โดยใช้สัญญาณจาก Incremental Encoder, Hall Current Sensor เป็น Input และแสดงสัญญาณ Output จากการ Log สัญญาณ และผลเป็นกราฟจาก Data Inspector ใน MATLAB Simulink และให้เห็นว่าสัญญาณ Output แปรผันตามสัญญาณ Input แบบ Real Time โดยมี Output เป็นความเร็วเชิงมุมและกระแสไฟฟ้า ในหน่วย SI derived อุปกรณ์การทดลอง

1.2 สมมติฐาน

- 1) ลักษณะเฉพาะของมอเตอร์นี้ สามารถหาได้จากการใช้ค่าคงตัวดังต่อไปนี้ ω_{NL} , i_{ST} , v_{in} , τ_{ST} , i_{NL}
- 2) การใช้หลักการทำงาน H-Bridge Drive และการควบคุมมอเตอร์ DC ที่มีการเปลี่ยนแปลง จะทำให้กระแสไฟฟ้าและความเร็วของมอเตอร์เกิดการเปลี่ยนแปลง
- 3) เมื่อแรงบิดของ Dynamic Torque Motor เปลี่ยนแปลง จะทำให้กระแสไฟฟ้าและความเร็วของ Brushed DC Motor เปลี่ยนแปลงไป

1.3 ตัวแปร

1) การทดลองที่ 1 และ 2

ตัวแปรต้น : ขนาดของโหลด (Load)

ตัวแปรตาม : ลักษณะเฉพาะของมอเตอร์

ตัวแปรควบคุม : ทิศทางและแรงดันไฟฟ้าของมอเตอร์

2) การทดลองที่ 3

ตัวแปรต้น : แรงบิดของ Dynamic Torque Motor

ตัวแปรตาม : กระแสไฟฟ้าและแรงบิดของ Brushed DC Motor

ตัวแปรควบคุม : ทิศทางและแรงดันไฟฟ้าของ Brushed DC Motor

3) การทดลองที่ 4

ตัวแปรต้น : หลักการทำงาน H-Bridge Drive Mode และการควบคุม DC Motor

ตัวแปรตาม : กระแสไฟฟ้าและความเร็วของมอเตอร์

ตัวแปรควบคุม : ขนาดของ Load

1.4 นิยามศัพท์เฉพาะ

- 1.4.1 Motor-Torque Constant: ค่าความต้านทานในมอเตอร์ หรือเขียนแทนด้วย K_m มักใช้กับการคำนวณเพื่อหาค่าต่างๆ ในมอเตอร์
- 1.4.2 Efficiency: ตัวบ่งชี้ประสิทธิภาพของมอเตอร์ จากการแปลงพลังงานไฟฟ้าเป็นพลังงานกลสำหรับขับเคลื่อนโหลด มีค่าตั้งแต่ 0 ถึง 100 เปอร์เซ็นต์
- 1.4.3 Dynamic Load: โหลดที่มีการเปลี่ยนแปลงหรือแปรผันตลอดเวลา
- 1.4.4 Stall Torque: แรงบิดสูงสุดที่มอเตอร์สามารถสร้างได้เมื่อเพลาของมอเตอร์หยุดนิ่ง ($\omega = 0$)
- 1.4.5 No-Load Torque: แรงบิดต่ำสุดที่มอเตอร์ต้องใช้เพื่อหมุนเพลาของตัวเองในสภาพไม่มีโหลด
- 1.4.6 Duty Cycle: Duty Cycle: แสดงผลเป็นเปอร์เซ็นต์ โดยคำนวณจากอัตราส่วนของเวลาที่มอเตอร์ทำงาน (On Time) ต่อเวลาทั้งหมด (On Time + Off Time) ในหนึ่งรอบ
- 1.4.7 Rotor: ชิ้นส่วนมอเตอร์ที่หมุน

1.5 นิยามเชิงปฏิบัติการ

- 1.5.1 DC Motor คือ Brush DC Motor
- 1.5.2 Dynamic Load เกิดจากแรงบิดของ Dynamic Torque Motor

1.6 เอกสารและงานวิจัยที่เกี่ยวข้อง

- 1.6.1 ลักษณะเฉพาะของมอเตอร์ (Motor Characteristics)

Brushed DC Motor มีลักษณะเฉพาะที่หลากหลาย แต่มีลักษณะเฉพาะที่สำคัญดังนี้

- 1) Torque-Speed Characteristic (แรงบิด-ความเร็ว)

เมื่อ Load เพิ่มขึ้น (แรงบิดสูงขึ้น) ความเร็วรอบของมอเตอร์จะลดลง กราฟมีลักษณะเป็นเส้นตรงลดลงจากซ้ายไปขวา (Negative Slope)

จากกฎแรงดันของเคิร์ชhoff (Kirchhoff's Voltage Law หรือ KVL) และ กฎการเคลื่อนที่ของนิวตัน-อยเลอร์ (Newton-Euler's Law) จะได้สมการดังนี้

- 1.1) เมื่อค่า Torque สูงสุด (ความเร็วเข้ามุมมีค่าเท่ากับศูนย์ หรือ Stall Torque Condition)

$$\omega_{NL} = \frac{K_m v_{in}}{BR + K_m K_b}$$

- 1.2) เมื่อค่า Torque ต่ำสุด (ไม่มี Load หรือ No-Load Condition)

$$\tau_{ST} = \frac{K_m v_{in}}{R}$$

เมื่อ ω_{NL} คือความเร็วเข้ามุมขณะไม่มีโหลด/ความเร็วสูงสุดของมอเตอร์ มีหน่วยเป็น Radian/s

K_m คือค่าคงที่ของมอเตอร์ที่สัมพันธ์กับแรงบิด มีหน่วยเป็น N · m/A

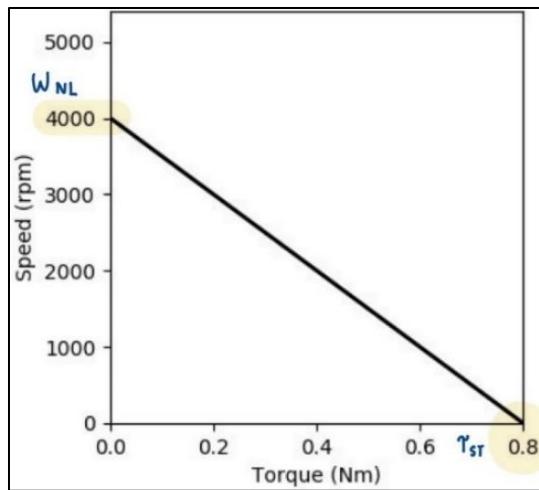
v_{in} คือแรงดันไฟฟ้าขาเข้า มีหน่วยเป็นโวลต์ (V)

B คือค่าความแม่เหล็กหรือแรงต้านการหมุน มีหน่วยเป็น N · m · s/Radian

R คือความต้านทานของชุดลวดในมอเตอร์ มีหน่วยเป็น โอม (Ω)

K_b คือค่าคงที่ของ Back-EMF มีหน่วยเป็น V · s/Radian

τ_{ST} คือแรงบิดสูงสุด มีหน่วยเป็น N · m



รูปที่ 1 แสดงกราฟแสดงความสัมพันธ์ระหว่าง Torque และ Speed

2) Torque-Current Characteristic (แรงบิด-กระแสไฟฟ้า)

เมื่อ Load เพิ่มขึ้น (แรงบิดสูงขึ้น) กระแสไฟฟ้าภายในมอเตอร์จะเพิ่มขึ้น กราฟมีลักษณะเป็นเส้นตรงเพิ่มขึ้นจากซ้ายไปขวา

จากกฎแรงดันของเคิร์ชhoff (Kirchhoff's Voltage Law หรือ KVL และ กฎการเคลื่อนที่ของนิวตัน-ออยเลอร์ (Newton-Euler's Law) จะได้สมการดังนี้

- 2.1) เมื่อค่า Torque สูงสุด (ความเร็วเชิงมุมมีค่าเท่ากับศูนย์ หรือ Stall Torque Condition)

$$i_{NL} = \frac{Bv_{in}}{BR + K_m K_b}$$

- 2.2) เมื่อค่า Torque ต่ำสุด (ไม่มี Load หรือ No-Load Condition)

$$i_{ST} = \frac{v_{in}}{R}$$

เมื่อ i_{NL} คือกระแสไฟฟ้าขณะไม่มีโหลด/ความเร็วสูงสุดของมอเตอร์ มีหน่วยเป็นแอมเปียร์ (A)

K_m คือค่าคงที่ของมอเตอร์ที่สัมพันธ์กับแรงบิด มีหน่วยเป็น ($N \cdot m/A$)

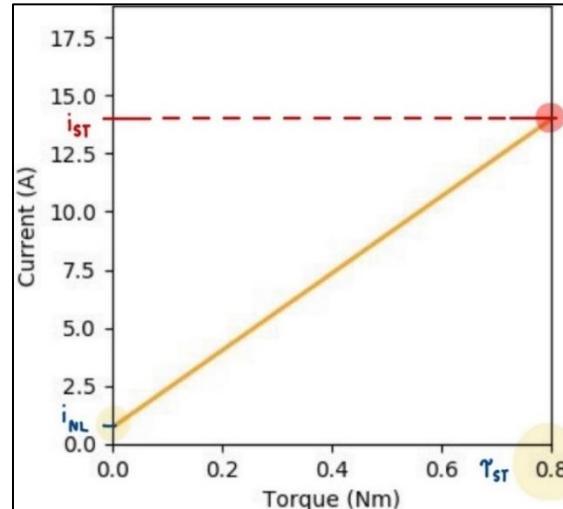
v_{in} คือแรงดันไฟฟ้าขาเข้า มีหน่วยเป็นโวลต์ (V)

B คือค่าความหนืดหรือแรงต้านการหมุน มีหน่วยเป็น $N \cdot m \cdot s/Radian$

R คือความต้านทานของขดลวดในมอเตอร์ มีหน่วยเป็นโอห์ม (Ω)

K_b คือค่าคงที่ของ Back-EMF มีหน่วยเป็น $V \cdot s/Radian$

i_{ST} คือกระแสไฟฟ้าในสภาวะเริ่มต้นหรือกระแสที่เกิดขึ้นเมื่อแรงบิดเริ่มต้น
มีหน่วยเป็นแอมป์ร์ (A)



รูปที่ 2 แสดงกราฟแสดงความสัมพันธ์ระหว่าง Torque และ Current

3) Power Characteristic (พลังงาน)

เมื่อ Load เพิ่มขึ้น (แรงบิดสูงขึ้น) พลังงานของมอเตอร์จะเพิ่มขึ้น โดยภาพมีลักษณะเป็นพาราโบลาคว่า ซึ่งมีพลังงานสูงสุดในช่วงหนึ่งเป็นจุดเปลี่ยนโค้ง และลดลงเมื่อแรงบิดของโหลดเพิ่มขึ้น เกินค่าที่เหมาะสมกับมอเตอร์

จากกฎแรงดันของเคิร์ชhoff (Kirchhoff's Voltage Law หรือ KVL) และ กฎการเคลื่อนที่ของนิวตัน-อยเลอร์ (Newton-Euler's Law) จะได้สมการดังนี้

3.1) สมการพลังงานทั่วไป

$$P = -\frac{\omega_{NL}}{\tau_{ST}}\tau_L^2 + \omega_{NL}\tau_L$$

- 3.2) เมื่อค่า Torque สูงสุด (ความเร็วเชิงมุมมีค่าเท่ากับศูนย์ หรือ Stall Torque Condition)

$$P_{max} = \frac{\tau_{ST}\omega_{NL}}{4}$$

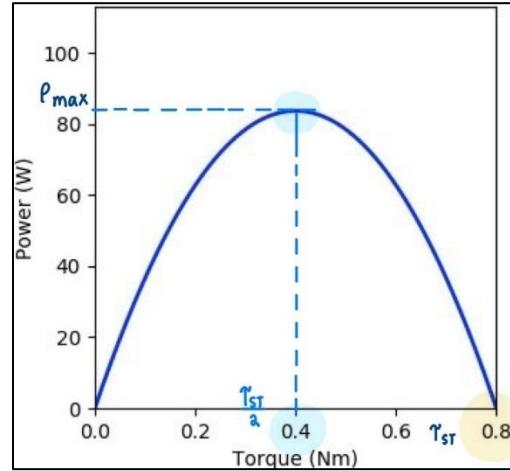
เมื่อ P คือกำลังงานของมอเตอร์ มีหน่วยเป็นวัตต์ (W)

ω_{NL} คือความเร็วเชิงมุมขณะไม่มีโหลด/ความเร็วสูงสุดของมอเตอร์ (Radian/s)

τ_{ST} คือแรงบิดสูงสุด มีหน่วยเป็น N · m

τ_L คือแรงบิดที่โหลด (Load Torque) มีหน่วยเป็น N · m

P_{max} คือกำลังงานสูงสุดของมอเตอร์ มีหน่วยเป็นวัตต์ (W)



รูปที่ 3 แสดงกราฟแสดงความสัมพันธ์ระหว่าง Torque และ Power

4) Efficiency Percentage (เบอร์เซ็นต์ประสิทธิภาพ)

คืออัตราส่วนระหว่าง พลังงานไฟฟ้าที่แปลงเป็นพลังงานกล (กำลังงานที่มอเตอร์ส่งออกไปใช้งาน) ต่อ พลังงานไฟฟ้าที่ป้อนเข้าให้กับมอเตอร์ ซึ่งแสดงในรูปของเบอร์เซ็นต์ โดยเป็นตัวบ่งชี้ประสิทธิภาพของมอเตอร์ จากการการแปลงพลังงานไฟฟ้าเป็นพลังงานกลสำหรับขับเคลื่อนโหลด グラฟของ Efficiency ต่อแรงบิดเป็นกราฟพาราโบลาเป็นทางขวา และมีสูตรการคำนวณดังนี้

$$\eta = \frac{P_{out}}{P_{in}} \times 100\% = \frac{-\frac{\omega_{NL}}{\tau_{ST}}\tau_L^2 + \omega_{NL}\tau_L}{\left(\frac{i_{st} - i_{NL}}{\tau_{ST}}\right)\tau_L v_{in} + i_{NL}v_{in}} \times 100\%$$

เมื่อ η คือเบอร์เซ็นต์ประสิทธิภาพของมอเตอร์ มีหน่วยเป็นเบอร์เซ็นต์ (%)

P_{out} คือกำลังงานเชิงกลของมอเตอร์ มีหน่วยเป็นวัตต์ (W)

P_{in} คือกำลังไฟฟ้าของมอเตอร์ มีหน่วยเป็นวัตต์ (W)

ω_{NL} คือความเร็วเชิงมุมขณะไม่มีโหลด/ความเร็วสูงสุดของมอเตอร์ มีหน่วยเป็น (Radian/s),

τ_{ST} คือแรงบิดสูงสุด มีหน่วยเป็น ($N \cdot m$)

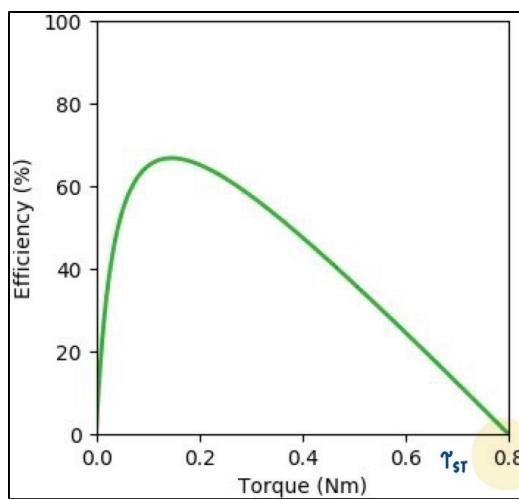
τ_L คือแรงบิดที่โหลด (Load Torque) มีหน่วยเป็น ($N \cdot m$)

P_{max} คือกำลังงานสูงสุดของมอเตอร์ มีหน่วยเป็น (W)

i_{st} คือกระแสไฟฟ้าในสภาวะเริ่มต้นหรือกระแสที่เกิดขึ้นเมื่อแรงบิดเริ่มต้น มีหน่วยเป็นแอมเปอร์ (A)

i_{NL} คือกระแสไฟฟ้าขณะไม่มีโหลด/ความเร็วสูงสุดของมอเตอร์ มีหน่วยเป็นแอมเปอร์ (A)

v_{in} คือแรงดันไฟฟ้าขาเข้า มีหน่วยเป็นโวลต์ (V)



รูปที่ 4 แสดง กราฟแสดงความสัมพันธ์ระหว่าง Torque และ Efficiency

1.6.2 Dynamic Torque Motor

Dynamic Torque Motor เป็นมอเตอร์ที่ออกแบบมาเพื่อให้สามารถควบคุมแรงบิด (Torque) ได้อย่างแม่นยำในระหว่างการทำงาน โดยเฉพาะในงานที่ต้องการการเปลี่ยนแปลงของแรงบิดในแบบเรียลไทม์ เช่น การเร่งความเร็ว การเบรก หรือการรักษาระดับความเร็วในสภาวะที่โหลดมีการเปลี่ยนแปลง โดยมีคุณสมบัติหลักดังนี้

1) สามารถควบคุมแรงบิดโดยตรง

มอเตอร์ชนิดนี้เน้นการควบคุมแรงบิดโดยตรงผ่านระบบอิเล็กทรอนิกส์หรือซอฟต์แวร์ควบคุม เช่น การใช้ตัวควบคุมแบบ Field-Oriented Control (FOC) หรือ Direct Torque Control (DTC) ที่ช่วยเพิ่มประสิทธิภาพและความแม่นยำในการปรับแรงบิดตามความต้องการ

2) ตอบสนองเร็ว

มีความสามารถในการตอบสนองต่อการเปลี่ยนแปลงของแรงบิดได้อย่างรวดเร็ว เหมาะสำหรับการใช้งานในระบบที่มีการเปลี่ยนแปลงโหลดหรือการทำงานที่ซับซ้อน เช่น แขนหุ่นยนต์ หรือเครื่องจักรกล

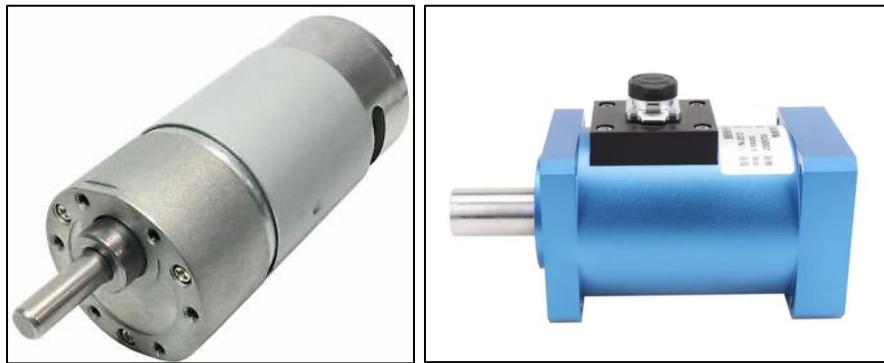
3) พลังงานและความแม่นยำสูง

Dynamic Torque Motor มักใช้ในการประยุกต์ที่ต้องการความแม่นยำสูงในการควบคุม เช่น การควบคุมการหมุนในระบบ Servo Motor, มอเตอร์ไฟฟ้าของยานพาหนะไฟฟ้า (EV), หรือในกระบวนการผลิตที่ต้องการความละเอียดสูง

4) การทำงานแบบไอนามิก

มอเตอร์สามารถปรับเปลี่ยนแรงบิดได้ตามความต้องการของงาน เช่น การเพิ่มแรงบิดชั่วขณะเพื่ออาชนัชแรงเฉือนของระบบ หรือการลดแรงบิดเพื่อประหยัดพลังงานในช่วงเวลาที่ไม่ต้องใช้แรงสูง

นอกจากนี้ยังสามารถนำไปใช้งานได้อย่างหลากหลาย เช่น อุตสาหกรรมการผลิต, ยานพาหนะไฟฟ้า, ระบบควบคุมการเคลื่อนที่ และงานวิจัยและการทดสอบ



รูปที่ 5 - 6 แสดงตัวอย่าง Dynamic Torque Motor และ Dynamic Torque Sensor

1.6.3 หลักการทำงาน H-Bridge Drive Mode และวิธีการควบคุม

- 1) H-Bridge ใช้สำหรับควบคุมมอเตอร์ DC โดยมี mode การทำงานหลัก 4 โหมด ได้แก่

1.1) Lock Anti-Phase Drive

ใช้ควบคุมความเร็วและทิศทางด้วยการสลับ PWM ระหว่างขั่วบวกและขั่วลบอย่างต่อเนื่อง และมีลักษณะเด่นคือการเปลี่ยนขั่วแรงตันอย่างรวดเร็ว ทำให้เหมาะสมสำหรับการควบคุมที่ต้องการการตอบสนองสูง

1.2) Sign-Magnitude Drive

ใช้ PWM เพื่อปรับความเร็วในขณะที่ควบคุมทิศทางผ่านการเปิดปิด MOSFET มักใช้วิธีนี้เพื่อความแม่นยำในการควบคุมความเร็ว

1.3) Asynchronous Sign-Magnitude Drive

คล้ายกับ Sign-Magnitude Drive แต่ใช้ Dead-Time เพื่อหลีกเลี่ยงการชนกันของกระแส (Short Circuit) ระหว่างการเปลี่ยนสถานะ สามารถลดการเกิดความร้อนใน MOSFET

1.4) Free (High Impedance)

สถานะที่ MOSFET อยู่ในสถานะเปิดทั้งหมด ทำให้มอเตอร์หมุนโดยอิสระ (Coast) เหมาะสำหรับการหยุดมอเตอร์แบบไม่มีแรงต้าน

2) H-Bridge Drive Mode มีวิธีการควบคุมดังนี้

2.1) การเปลี่ยนทิศทาง (Direction Control)

การเปลี่ยนทิศทางจะทำโดยควบคุมการเปิดปิดของ MOSFET ภายใน H-Bridge

เช่น เปิด MOSFET Q1 และ Q4 ทำให้มอเตอร์หมุนไปทางหน้า, เปิด MOSFET Q2 และ Q3 ทำให้มอเตอร์หมุนอยหลัง และต้องมี Dead-Time เพื่อป้องกัน Short Circuit

2.2) การควบคุมความเร็ว (Speed Control)

ใช้ PWM (Pulse Width Modulation) เพื่อควบคุมแรงดันเฉลี่ยที่ส่งไปยังมอเตอร์ ซึ่งความถี่ของ PWM มีผลต่อการตอบสนองของมอเตอร์ ดังนี้ เมื่อ ความถี่ต่ำ ทำให้ กระแสสูงขึ้น และ ประสิทธิภาพลดลง, ความถี่สูงทำให้กระแสต่ำและ ประสิทธิภาพสูงขึ้น

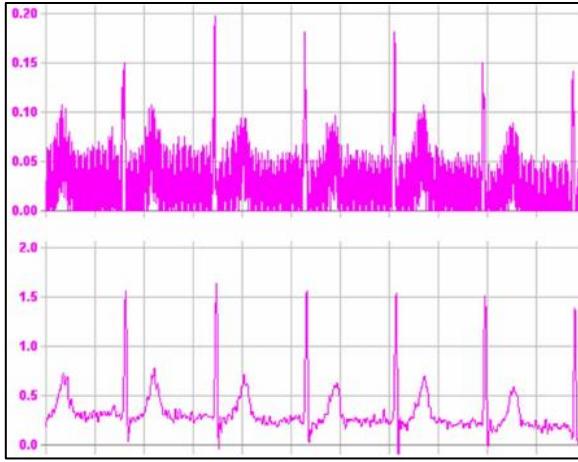
2.3) การเบรก (Braking)

ใช้ Slow Decay (เบรกแบบช้า) หรือ Fast Decay (เบรกแบบเร็ว) เพื่อควบคุมทิศทาง ซึ่ง Slow Decay ใช้แรงต้านจากการกระแสเยื่อแกลบผ่านมอเตอร์ ส่วน Fast Decay ใช้การตัดวงจรเพื่อหยุดมอเตอร์เร็วขึ้น

1.6.4 Signal Conditioning

Signal Conditioning คือกระบวนการเตรียมสัญญาณจากเซ็นเซอร์หรือแหล่งกำเนิด เพื่อให้เหมาะสมกับการนำไปประมวลผลต่อไป โดยมักประกอบด้วยขั้นตอนดังนี้

- 1) การขยายสัญญาณ (Amplification) : ใช้ขยายสัญญาณที่มีระดับแรงดันต่ำให้สูงขึ้น เช่น จาvmicelitolect เป็นโวลต์ มักใช้ Operational Amplifier (Op-Amp)
 - 2) การกรองสัญญาณ (Filtering) : กำจัดสัญญาณรบกวน (Noise) หรือความถี่ที่ไม่ต้องการ ตัวกรองมีหลายประเภท เช่น Low-pass, High-pass, Band-pass, Band-stop
 - 3) การปรับระดับสัญญาณ (Level Shifting) : ใช้ในกรณีที่สัญญาณมีค่าแรงดันที่ไม่เหมาะสม เช่น การเปลี่ยนช่วงแรงดันให้เข้ากับ ADC (Analog-to-Digital Converter)
 - 4) การแปลงรูปแบบสัญญาณ (Signal Conversion) : เช่น การแปลงสัญญาณกระแส (Current) เป็นแรงดัน (Voltage) หรือจากแรงดันเป็นดิจิทัล
 - 5) การปรับสมดุลสัญญาณ (Impedance Matching) : ทำให้ความต้านทานของแหล่งกำเนิดสัญญาณและตัวรับสัญญาณตรงกัน เพื่อลดการสูญเสียพลังงาน
- ยกตัวอย่างการใช้ Signal Conditioning เช่น การขยายสัญญาณจากเทอร์โมคัปเปิล (Thermocouple) ก่อนส่งเข้าสู่ไมโครคอนโทรลเลอร์



รูปที่ 7 แสดงตัวอย่างสัญญาณก่อนและหลังเข้ากระบวนการ Signal Conditioning

1.6.5 Signal Processing

Signal Processing คือกระบวนการประมวลผลและวิเคราะห์ข้อมูลที่ได้รับจากการวัดสัญญาณโดยอาจทำในรูปแบบ อนาล็อก (Analog) หรือ ดิจิทัล (Digital) เพื่อให้ได้ข้อมูลที่มีความหมายและสามารถใช้งานได้

- 1) การแปลงสัญญาณ (Signal Transformation) : Fourier Transform เพื่อแปลงสัญญาณจากโดเมนเวลา (Time Domain) ไปเป็นโดเมนความถี่ (Frequency Domain)
- 2) การกรองสัญญาณแบบดิจิทัล (Digital Filtering) : การกรองสัญญาณผ่านการคำนวณ เช่น FIR (Finite Impulse Response) หรือ IIR (Infinite Impulse Response) Filter
- 3) การลดสัญญาณรบกวน (Noise Reduction) : เช่น การใช้ Kalman Filter เพื่อคาดการณ์และลดผลกระทบจาก Noise
- 4) การตรวจจับเหตุการณ์ (Event Detection) : เช่น การตรวจจับจุดพีคของสัญญาณ ECG (คลื่นไฟฟ้าหัวใจ)
- 5) การปรับสเกลและการทำให้สัญญาณปกติ (Normalization) : การปรับระดับสัญญาณให้อยู่ในช่วงค่าที่กำหนด เช่น 0 ถึง 1 หรือ -1 ถึง 1

1.7 วิธีดำเนินการทดลอง

1.7.1 วัสดุอุปกรณ์

- 1) Incremental Encoder AMT103-V
- 2) Nidec Components Geared DC Geared Motor, 12VDC, 20 Ncm, 70 RPM, 6mm Shaft Diameter
- 3) Warner Electric Magnetic Particle Clutches MPB12
- 4) WCS1700 Hall Current Sensor
- 5) Cytron MDD20A Motor Driver
- 6) Nucleo STM32G474RE
- 7) MotorXplorer
- 8) สายจัมเปอร์

1.7.2 ขั้นตอนการดำเนินงาน การทดลองที่ 1 และ 2

คณะผู้จัดทำได้แบ่งการทดลองออกเป็นทั้งหมด 3 ช่วง ประกอบด้วย Calibration, Measurement และ Calculation โดยแต่ละช่วงจะมีขั้นตอนย่อยต่อไปนี้

1) Calibration

ในช่วงนี้จะเป็นการเตรียมเซ็นเซอร์ และอุปกรณ์ทดลองให้พร้อมสำหรับการเก็บผล โดยจะประกอบด้วยขั้นตอนย่อยต่อไปนี้

1.1) Current Sensor Calibration

ก่อนที่จะทำการรัดข้อมูลจาก Current sensor ควรดำเนินการใช้วิธี Linear Regression เพื่อปรับปรุงความแม่นยำในการประมาณค่ากระแสไฟฟ้า ในการทำ Linear Regression นั้น จำเป็นต้องเก็บค่าแรงดันไฟฟ้าที่ได้จาก Current sensor โดยในกรณีที่กระแสไฟฟ้าเท่ากับ 0 A ค่าแรงดันไฟฟ้าที่วัดได้คือ 2.05 V และในกรณีกระแสไฟฟ้าเดาๆ ทางคณะผู้จัดทำได้ใช้ Multimeter เป็นเครื่องมือวัดเปรียบเทียบ โดยทำการต่อ Multimeter แบบอนุกรมเพื่อวัดกระแสไฟฟ้าที่แหล่งกำเนิด

ในขณะที่มอเตอร์หมุนโดยไม่มีโหลด (No-Load) ซึ่งค่าแรงดันไฟฟ้าที่ได้จากเซ็นเซอร์คือ 2.11 V และค่ากระแสไฟฟ้าที่วัดได้จาก Multimeter คือ 0.92 A เมื่อนำข้อมูลดังกล่าวเข้าสมการ Linear Regression จะได้ค่าความชัน (m) เท่ากับ 15.33 และค่าเชิงเส้นตัดแกน (C) เท่ากับ -31.4265 ส่งผลให้สมการที่แสดงความสัมพันธ์ระหว่างกระแสไฟฟ้าที่ผ่านเซ็นเซอร์กับแรงดันไฟฟ้าที่เซ็นเซอร์ให้สามารถเขียนได้ดังนี้

$$I = 15.33(V) - 31.4265$$

เมื่อ เทียบจากสมการเส้นตรง $y = mx + c$
 I คือ กระแสไฟฟ้า มีหน่วยเป็นแอมป์ร์ (A)
 V คือ แรงดันไฟฟ้ามีหน่วยเป็นโวลต์ (V)

1.2) Loadcell Calibration

ก่อนดำเนินการวัดข้อมูลจาก Loadcell จะเป็นต้องสร้างสมการที่สามารถแปลงค่าแรงดันไฟฟ้าให้เป็นค่าน้ำหนักได้อย่างถูกต้อง ทั้งนี้เนื่องจากมอเตอร์ เมื่อผ่านการส่งกำลังผ่านก้านส่งกำลังแล้ว จะสร้างน้ำหนักที่น้อยกว่า 1 กิโลกรัม ทางคณะผู้จัดทำจึงได้ทำการปรับค่า R_g ให้เหมาะสมกับการวัด เพื่อให้ค่าแรงดันที่ได้จาก Loadcell มีลักษณะเป็นเส้นตรง (Linear) มากที่สุด

โดยทราบว่าแรงดันสูงสุดที่ Loadcell สามารถจ่ายได้คือ 2.5 V ทั้งนี้ คณะผู้จัดทำได้ปรับค่าแรงดันที่ออกจาก Loadcell ให้มีค่าอยู่ที่ 2.35 V เมื่อรับน้ำหนัก 990 กรัม และตรวจสอบค่าแรงดันที่วัดได้เมื่อไม่มีน้ำหนัก (0 กรัม) พบว่ามีค่า 0.26 V จากนั้นจึงนำค่าที่ได้เข้าสมการ Linear Regression เพื่อประมาณค่า

พบว่าค่าความชัน (m) เท่ากับ 473.68 และค่าความตัดแกน (c) เท่ากับ -123.1568 ทำให้สามารถเขียนสมการที่แสดงความสัมพันธ์ระหว่างน้ำหนักและแรงดันไฟฟ้าที่วัดได้จาก Loadcell ได้ดังนี้

$$\text{Mass} = 473.68(V) - 123.1568$$

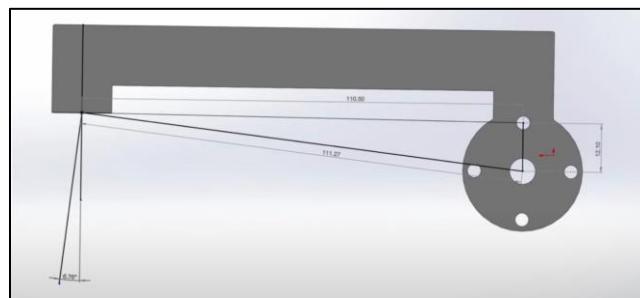
เมื่อ เทียบจากสมการเส้นตรง $y = mx + c$
 Mass คือน้ำหนักที่ตรวจจับได้
 V คือแรงดันไฟฟ้า มีหน่วยเป็นโวลต์ (V)

1.3) Torque Transmission Calculations

เนื่องจากการวัดค่า Torque จำเป็นต้องอาศัยก้านส่งกำลังที่ทำหน้าที่ถ่ายทอดแรงจากมอเตอร์ไปยังก้านส่งกำลังและกดลงบน Loadcell เมื่อผ่านกระบวนการส่งกำลัง จึงต้องทำการแปลงค่าจากน้ำหนักที่วัดได้จาก Loadcell ให้เป็นค่า Torque ซึ่งสามารถคำนวณได้โดยใช้สมการ $\tau = \vec{F} \times r$ โดยอาศัยความยาวของก้านส่งกำลัง ทำให้สามารถคำนวณค่า Torque ได้ด้วยสมการดังต่อไปนี้

$$\tau = \left(\frac{\text{mass}}{100} \times \cos(6.76) \right) \times 0.11127$$

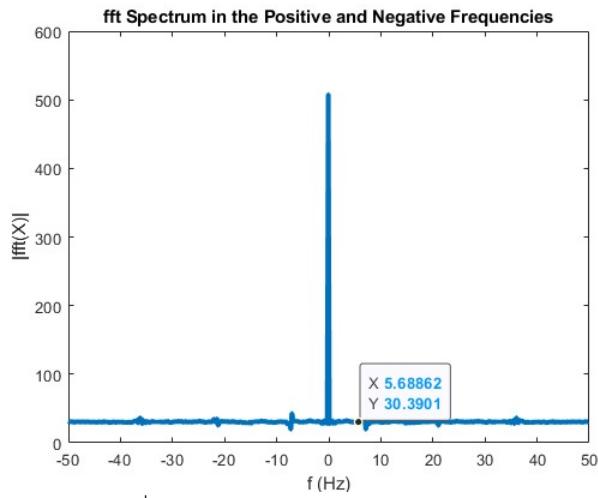
เมื่อ τ คือ แรงบิดที่เกิดขึ้น มีหน่วยเป็น $N \cdot m$
 mass คือน้ำหนักที่ตรวจจับได้ มีหน่วยเป็นกิโลกรัม (kg)



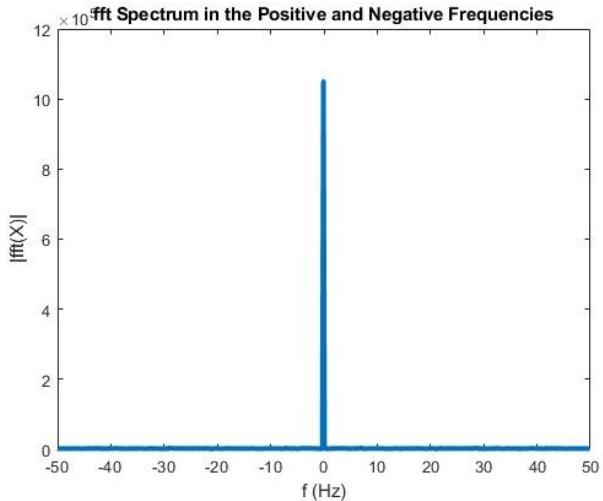
รูปที่ 8 แสดงความยาวของก้านส่งประจุในการคำนวณแรงบิด

1.4) Signal Conditioning

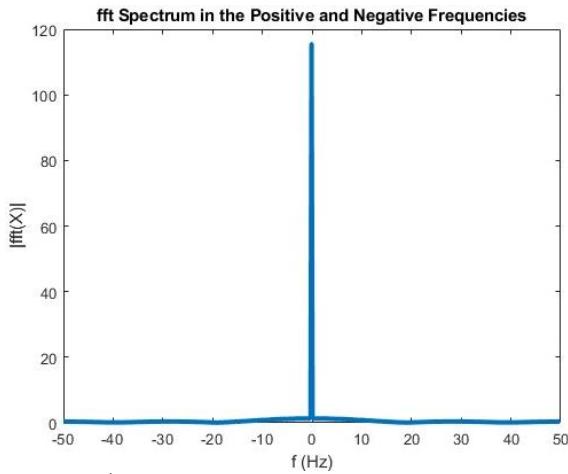
เนื่องจากข้อมูลที่ได้มีความผันผวนอันเกิดจากปัจจัยที่ไม่สามารถควบคุมได้ เช่น ความไม่ตรงแนว (Miss-alignment) และความไม่เป็นอุดมคติของเซ็นเซอร์ การกรองสัญญาณจึงจำเป็นต้องดำเนินการเพื่อแก้ไขความคลาดเคลื่อนของค่าที่วัดได้ โดยกระบวนการกรองสัญญาณสามารถแบ่งออกเป็นสองขั้นตอนย่อย ขั้นตอนแรกคือการใช้ FFT (Fast Fourier Transform) เพื่อวิเคราะห์และระบุช่วงความถี่ที่ไม่ต้องการ ซึ่งผลการวิเคราะห์มีรายละเอียดดังนี้



รูปที่ 9 แสดงผลการทำ FFT ของ Current Sensor



รูปที่ 10 แสดงผลการทำ FFT ของ Encoder



รูปที่ 11 แสดงผลการทำ FFT ของ Loadcell

จากการวิเคราะห์ด้วย FFT ทำให้สามารถระบุช่วงความถี่ที่ไม่ต้องการได้อย่างชัดเจน ซึ่งนำไปสู่การดำเนินการในขั้นตอนถัดไป นั่นคือกระบวนการกรองสัญญาณ (Filtering) จากผลการวิเคราะห์ด้วย FFT พบร่วมสัญญาณที่พิจารณาส่วนใหญ่ในช่วงความถี่ต่ำ ตั้งนั้น ทางคณะผู้จัดทำจึงเลือกใช้ Lowpass Filter เพื่อกรองความถี่ที่ไม่ต้องการออก

สำหรับสัญญาณจาก Current sensor ได้มีการเลือกใช้ Lowpass Filter ที่มีค่า Cut-Off Frequency เท่ากับ 5 Hz เพื่อกำจัดสัญญาณรบกวนออก อย่างไรก็ตาม สำหรับสัญญาณอื่นๆ จากการวิเคราะห์ด้วย FFT ไม่พบสัญญาณรบกวนที่เด่นชัด ทางคณะผู้จัดทำจึงเลือกใช้วิธี Running RMS ในการกรองค่าที่ไม่ต้องการแทนการใช้ Lowpass Filter

2) Measurement

ข้อมูลที่ทางคณะผู้จัดทำให้ความสนใจและดำเนินการเก็บรวบรวมมีทั้งหมด 4 ค่า โดยการ
เก็บข้อมูลแบ่งออกเป็น 2 รอบ ดังนี้

2.1) No-Load

เก็บข้อมูลขณะที่มอเตอร์หมุนโดยไม่ได้ต่อ Load ใดๆ โดยสามารถเก็บข้อมูล
ได้ 2 ค่าประกอบด้วย

(1) ω_{NL} ความเร็วเชิงมุมของมอเตอร์ ขณะมอเตอร์หมุนโดยไม่ได้ต่อ Load ใดๆ

(2) i_{NL} กระแสไฟฟ้า ขณะที่มอเตอร์หมุนโดยไม่ได้ต่อ Load ใดๆ

2.2) Stall

เก็บข้อมูลขณะที่มอเตอร์หมุนโดยใช้กำลังสูงสุด ซึ่งส่งผลให้มอเตอร์ไม่เกิดการขยับ
โดยสามารถเก็บข้อมูลได้ 2 ค่าดังต่อไปนี้

(1) τ_{ST} แรงบิดสูงสุดที่มอเตอร์สามารถส่งกำลังได้

(2) i_{ST} กระแสไฟฟ้า ขณะที่มอเตอร์ส่งกำลังสูงสุด

จากการเก็บข้อมูลดังกล่าวเพียงพอต่อการนำไปคำนวณในขั้นตอนถัดไป

3) Calculation

จากวัตถุประสงค์ของการทดลองที่ต้องการหา Motor Characteristic และความสัมพันธ์ของข้อมูลต่างๆ จึงจำเป็นต้องนำข้อมูลที่ได้รับมาทำการคำนวณต่อไป ซึ่งจะประกอบด้วยขั้นตอนการคำนวณดังต่อไปนี้

3.1) Torque Speed Curve

ความสัมพันธ์ระหว่างแรงบิดของมอเตอร์กับความเร็วเฉลี่ยของมอเตอร์ ตั้งแต่ No-Load Torque (Torque = 0 Nm) ถึง Stall Torque โดยสามารถใช้ข้อมูลจากการ Measurement มา Plot ลงกราฟด้วยสมการเส้นตรงได้เลย

3.2) Torque Current Curve

ความสัมพันธ์ระหว่างแรงบิดของมอเตอร์กับกระแสไฟฟ้าที่แหล่งกำเนิดมอเตอร์ ตั้งแต่ No-Load Torque (Torque = 0 Nm) ถึง Stall Torque โดยสามารถใช้ข้อมูลจากการ Measurement มา Plot ลงกราฟด้วยสมการเส้นตรงได้

3.3) Power Curve

ความสัมพันธ์ระหว่างแรงบิดของมอเตอร์กับกำลังที่มอเตอร์ใช้ ตั้งแต่ No-Load Torque (Torque = 0 Nm) ถึง Stall Torque โดยผ่านสมการดังต่อไปนี้

$$P = -\frac{\omega_{NL}}{\tau_{ST}} \tau_L^2 + \omega_{NL} \tau_L$$

ซึ่งสามารถนำข้อมูลที่ได้มาแทนในสมการได้เลย มีเพียง τ_L ที่จำเป็นต้องแทนค่าตั้งแต่ No-Load Torque (Torque = 0 Nm) ถึง Stall Torque โดยเพื่อความละเอียดทางคณนะผู้จัดทำได้เลือกใช้ค่าความละเอียดที่ 1000 ค่า

3.4) Efficiency Curve

ความสัมพันธ์ระหว่างแรงบิดของมอเตอร์กับประสิทธิภาพ ตั้งแต่ No-Load Torque (Torque = 0 Nm) ถึง Stall Torque โดยผ่านสมการดังต่อไปนี้

$$\eta = \frac{P_{out}}{P_{in}} = \frac{-\frac{\omega_{NL}}{\tau_{ST}} \tau_L^2 + \omega_{NL} \tau_L}{\left(\frac{i_{ST} - i_{NL}}{\tau_{ST}}\right) \tau_L V_{in} + i_{NL} V_{in}}$$

ซึ่งสามารถนำข้อมูลที่ได้มาแทนในสมการได้เลย มีเพียง τ_L ที่จำเป็นต้องแทนค่าตั้งแต่ No-Load Torque (Torque = 0 Nm) ถึง Stall Torque โดยเพื่อความละเอียดทางคณะ ผู้จัดทำได้เลือกใช้ค่าความละเอียดที่ 1000 ค่า

3.5) Rated

ความเร็วเชิงมุม, แรงบิด, กระแสไฟฟ้า และกำลังงานของมอเตอร์ ขณะที่ Efficiency มีค่าสูงสุดโดยสามารถหาได้จากการหาแรงบิด ณ จุดสูงสุดของ Efficiency Curve (จุดที่ $m = 0$) จากนั้นก็สามารถที่จะใช้ τ_{Rated} เพื่อหาค่า Rated ของค่าอื่นๆจาก ความสัมพันธ์อื่นได้ เช่น กัน เมื่อ τ_{Rated} คือ แรงบิดของมอเตอร์ที่ประสิทธิภาพของมอเตอร์ สูงที่สุด

1.7.3 ขั้นตอนการดำเนินงาน การทดลองที่ 3

ก่อนจะทำการทดลองที่ 3 จะเป็นต้องทำขั้นตอนที่ 1 (Calibration) จากการทดลองที่ 1 และ 2 ก่อน การทดลองที่ 3 เป็นการวัดกระแสไฟบวกกับแรงบิดอ้างอิง โดยจะทำการต่อ Dynamic Load Motor เข้ากับตัว Motor Driver และควบคุมด้วยวิธี Torque Control หลังจากนั้นก็มาหมุนสวานทางกับมอเตอร์ที่พิจารณา และวัดกระแส ด้วยวิธีการนี้จะทำให้สามารถสร้างแรงบิดที่ต้านกับมอเตอร์ที่พิจารณาได้ด้วยขนาดของแรงบิดที่มีความหลากหลาย ในที่นี้ทางคณะผู้จัดทำได้เก็บทั้งหมด 10 ค่าจากทั้งหมด 3 รอบ โดยตั้งค่าแรงบิดที่ $0.02 - 0.2 \text{ N} \cdot \text{m}$ การเก็บค่าทั้งสามรอบนั้นวิธีการเก็บที่แตกต่างกันดังนี้

- 1) No-Load จะเป็นการวัดกระแสไฟฟ้าจากมอเตอร์ที่พิจารณา โดยที่ยังไม่ได้เชื่อมมอเตอร์ที่พิจารณา กับ Dynamic Load Motor โดยจะเก็บทั้งหมด 10 ค่า
- 2) No-Drive จะเป็นการวัดกระแสไฟฟ้าจากมอเตอร์ที่พิจารณา โดยที่เชื่อมมอเตอร์ที่พิจารณา กับ Dynamic Load Motor แต่จะไม่ทำการควบคุมใด หรือไม่ได้ต่อมอเตอร์เข้ากับบอร์ด Motor Drive นั้นเอง โดยจะเก็บทั้งหมด 10 ค่า
- 3) Dynamic Load จะเป็นการวัดกระแสไฟฟ้าจากมอเตอร์ที่พิจารณา โดยที่เชื่อมมอเตอร์ที่พิจารณา กับ Dynamic Load Motor และทำการต้านการหมุนของมอเตอร์ที่พิจารณาด้วยแรงบิดตั้งแต่ $0.02 - 0.2 \text{ N} \cdot \text{m}$ ผ่าน Dynamic Load Motor ด้วย Current Control

1.7.4 ขั้นตอนการดำเนินงาน การทดลองที่ 4

การทดลองที่ 4 ประกอบด้วย 2 ส่วนหลัก ได้แก่ ส่วนที่ 1 การศึกษาความสัมพันธ์ระหว่าง Duty Cycle, ความเร็วการหมุนของมอเตอร์ และกระแสไฟฟ้าที่แหล่งผ่านมอเตอร์ และส่วนที่ 2 การวิเคราะห์การทำงานของโหมดการควบคุมต่างๆ โดยใช้ H-bridge Drive

- 1) การทดลองเพื่อศึกษาความสัมพันธ์ระหว่าง Duty Cycle, ความเร็วการหมุนของมอเตอร์ และกระแสไฟฟ้าที่แหล่งผ่านมอเตอร์

การทดลองดำเนินการโดยควบคุมการทำงานของมอเตอร์ด้วย Duty Cycle ในช่วง 0 ถึง 100% ที่ความถี่ 1000 Hz และ 20000 Hz โดยทำการวัดค่าความเร็วรอบและกระแสไฟฟ้าที่แหล่งผ่านมอเตอร์ สำหรับความถี่ 20000 Hz จะทำการวัดเฉพาะกระแสไฟฟ้าที่แหล่งผ่านมอเตอร์เท่านั้น

- 2) การทดลองเพื่อศึกษาการทำงานของโหมดการควบคุมต่างๆ ด้วย H-bridge Drive

การทดลองนี้ประกอบด้วยการทดสอบการทำงานของ H-bridge Drive ใน 2 โหมด ได้แก่ Lock Anti-Phase Drive และ Sign-Magnitude Drive โดยทำการตรวจสอบการลดลงของกระแสไฟฟ้าหลังจากหยุดจ่ายแรงดันไฟฟ้าให้กับมอเตอร์ เพื่อสังเกตและเปรียบเทียบพฤติกรรม Fast Decay ในโหมด Lock Anti-Phase และ Slow Decay ในโหมด Sign-Magnitude

- 2.1) สำหรับโหมด Lock Anti-Phase Drive ให้จ่ายสัญญาณ PWM ไปยังช่อง DIR และจ่ายแรงดันไฟฟ้า 3.3 V ไปยังช่อง PWM ของ MD20 A
- 2.2) สำหรับโหมด Sign-Magnitude Drive ให้จ่ายสัญญาณ PWM ไปยังช่อง PWM และจ่ายแรงดันไฟฟ้า 3.3 V ไปยังช่อง DIR

1.8 ผลการทดลอง

จากสมมุติฐานจะสามารถสร้างการทดลองย่อยได้ทั้งหมด 4 การทดลอง การทดลองที่ 1 คือการทดลองเพื่อหาลักษณะเฉพาะของ Brushed DC Motor, การทดลองที่ 2 คือ การทดลองเพื่อศึกษาความสัมพันธ์ของ Brushed DC Motor, การทดลองที่ 3 คือ การทดลองเพื่อศึกษาหลักการทำงาน H-Bridge Drive Mode และการควบคุม DC Motor และ การทดลองที่ 4 คือ การทดลองเพื่อศึกษาความสัมพันธ์ระหว่างกระแสไฟฟ้ากับค่า Torque Setpoint ของ Dynamic Torque Motor

1.8.1 ผลการทดลองที่ 1 และ ผลการทดลองที่ 2

จากการทดลองที่จะสามารถหาค่าคงตัวที่จำเป็นต่อการคำนวณได้ดังต่อไปนี้

$$\omega_{NL} = 2200 \text{ RPM} = 230 \text{ Radian/s}$$

$$i_{NL} = 0.98 \text{ A}$$

$$\tau_{ST} = 0.2 \text{ N} \cdot \text{m}$$

$$i_{ST} = 5 \text{ A}$$

$$V_{in} = 12 \text{ V}$$

เมื่อ ω_{NL} คือ ความเร็วเชิงมุมขณะไม่มีโหลด หรือความเร็วสูงสุดของมอเตอร์ (Radian/s)

i_{NL} คือ กระแสไฟฟ้าขณะไม่มีโหลด หรือความเร็วสูงสุดของมอเตอร์ (A)

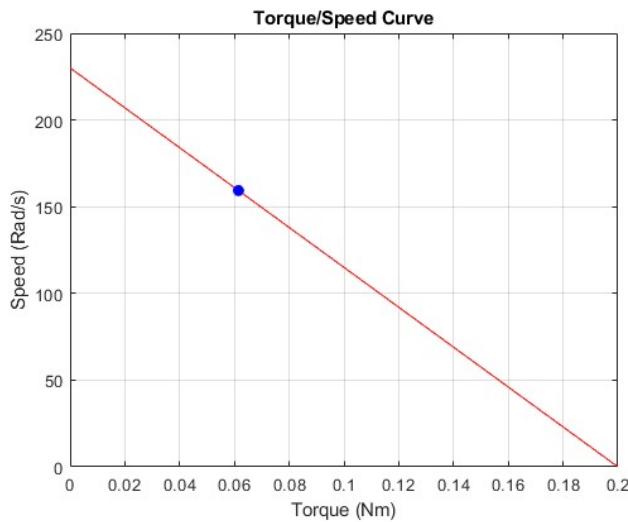
τ_{ST} คือ แรงบิดสูงสุด (N · m)

i_{ST} คือ กระแสไฟฟ้าในสภาวะเริ่มต้นหรือกระแสที่เกิดขึ้นเมื่อแรงบิดเริ่มต้น (A)

V_{in} คือ แรงดันไฟฟ้าขาเข้า (V)

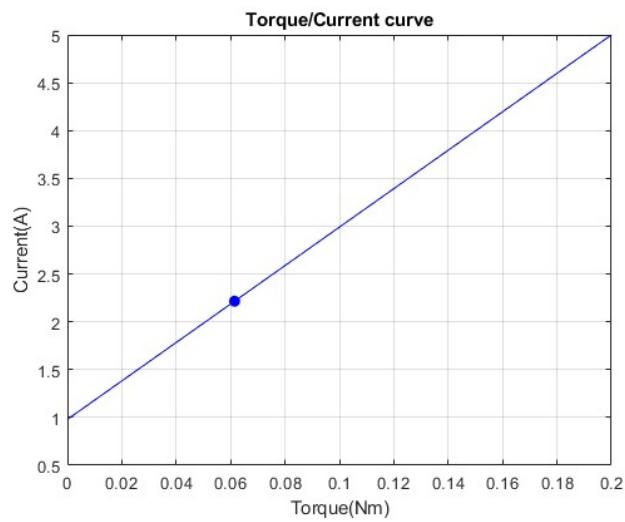
ซึ่งจากค่าคงตัวดังต่อไปนี้สามารถทำให้คณฑ์ผู้จัดทำการคำนวณและสร้างกราฟแสดงความสัมพันธ์ต่างๆ ได้ดังต่อไปนี้ โดยที่มีจุดสีฟ้าเป็นจุดที่ซึ่งให้เห็นถึงจุดที่มีค่า Efficiency สูงสุด

1) Torque Speed Curve



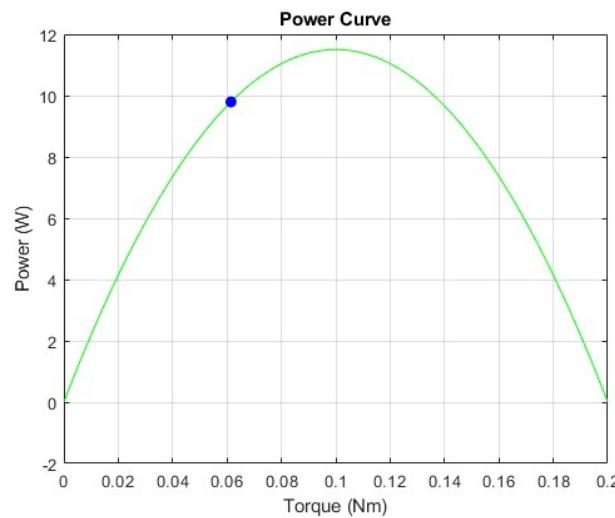
รูปที่ 12 แสดงกราฟแสดงความสัมพันธ์ระหว่างแรงบิดและความเร็วเชิงมุมของมอเตอร์

2) Torque Current Curve



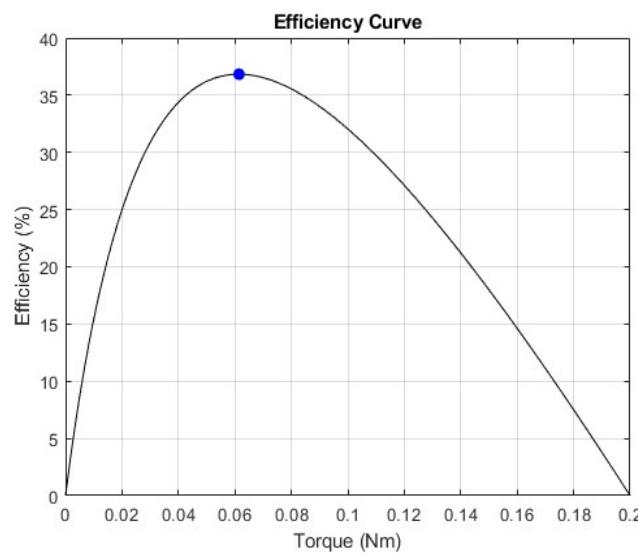
รูปที่ 13 กราฟแสดงความสัมพันธ์ระหว่างแรงบิดและกระแสไฟฟ้าของมอเตอร์

3) Power Curve



รูปที่ 14 แสดงกราฟแสดงความสัมพันธ์ระหว่างแรงบิดและกำลังงานของมอเตอร์

4) Efficiency Curve



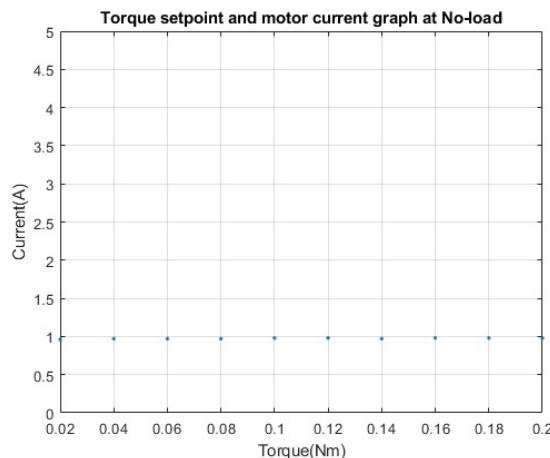
รูปที่ 15 แสดงกราฟแสดงความสัมพันธ์ระหว่างแรงบิดและประสิทธิภาพของมอเตอร์

หลังจากที่นำค่าคงตัวที่วัดได้มาพล็อตกราฟทำให้ได้ข้อสรุปของ Motor Characteristics ได้ดังต่อไปนี้

- Recommended/Rated Voltage : 12 V
- No-Load current @12V : 0.98 A
- No-Load speed @12V : 2200 RPM หรือ 230 Radian/s
- Stall current @12V : 5 A
- Stall Torque @12V : 0.2 N · m
- Rated current @12V : 2.21 A
- Rated speed @12V : 1519 RPM
- Rated Torque @12V : 0.0614 N · m
- Rated power @12V : 9.792 W
- Rated efficiency @12V : 36.83 %
- Shaft diameter 5 mm

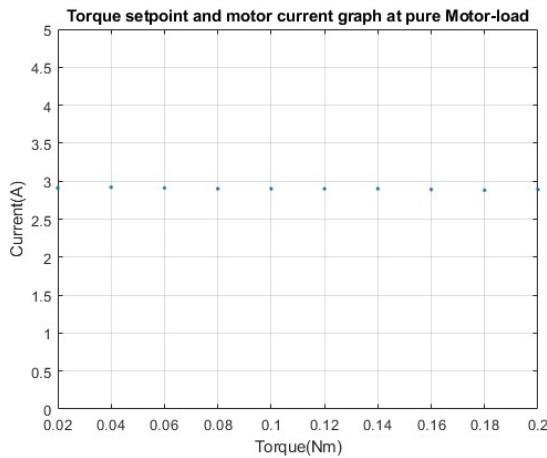
1.8.2 ผลการทดลองที่ 3

การทดลองนี้จะเป็นการสังเกตพฤติกรรมกระแสไฟฟ้าที่แหล่งจ่ายไฟมอเตอร์ขณะที่รับ Load ต่างๆ ผ่าน Dynamic Torque Motor โดยจะทำการทดลองทั้งหมดสามครั้ง ซึ่งสามารถสรุปผลได้ดังนี้



รูปที่ 16 แสดงความสัมพันธ์ระหว่างกระแสไฟฟ้าและแรงบิดของมอเตอร์เมื่อไม่มีโหลด (No-Load)

จากการทดลองครั้งที่ 1 พบร่วมกับการให้มอเตอร์หมุนโดยไม่ได้ต่อแหล่งพลังงานใดๆ ทำให้สามารถทดสอบ Dynamic Load Motor จะทำให้ค่ากระแสเมื่อไม่มี Load มากกว่าทำ



รูปที่ 17 แสดงความสัมพันธ์ระหว่างกระแสไฟฟ้าและแรงบิดของมอเตอร์เมื่อรับโหลดมอเตอร์เพียงอย่างเดียว
(Pure Motor-Load)

จากการทดลองครั้งที่ 2 พบร่วมกับการใช้แรงบิดในการทำให้ Rotor หมุน จากการที่คณะผู้จัดทำทราบแรงบิดสูงสุด ณ Stall Torque และกระแสขณะที่มอเตอร์จ่ายแรงบิดสูงสุดทำให้สามารถหาค่าคงตัวของมอเตอร์ออกมากได้ด้วยสมการดังต่อไปนี้

$$\tau = K_m i$$

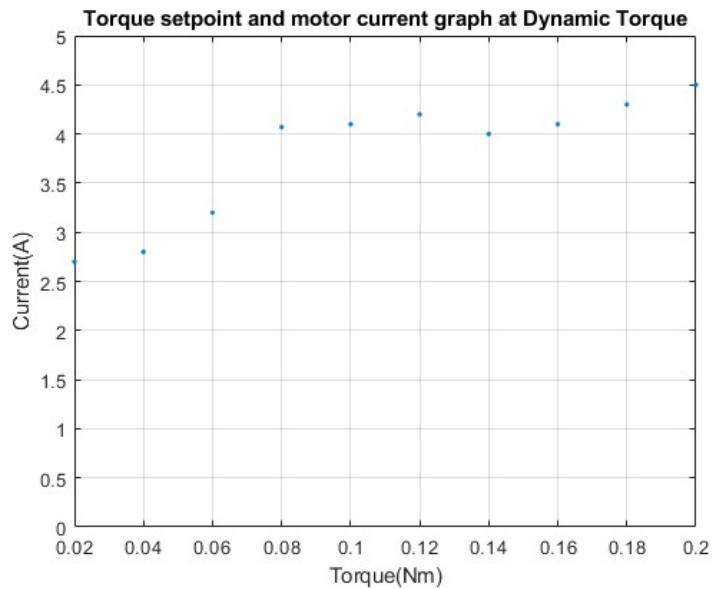
เมื่อ τ แทน แรงบิดสูงสุด ณ Stall Torque

i แทน กระแสไฟฟ้าที่มอเตอร์จ่ายแรงบิดสูงสุด และ

K_m แทนค่าคงตัวของมอเตอร์ จะได้

$$K_m = \frac{0.2}{5} = 0.04$$

และจากสมการเดียวกันทำให้คณะผู้จัดทำทราบว่า แรงบิดของ Dynamic Torque Motor ขณะไม่ได้ควบคุมหรือต่อ Motor Driver และ ขณะที่มีกระแสไฟฟ้าไหลผ่านมอเตอร์ที่พิจารณาที่ 2.8 A จะมีแรงบิดที่ $0.112 \text{ N} \cdot \text{m}$



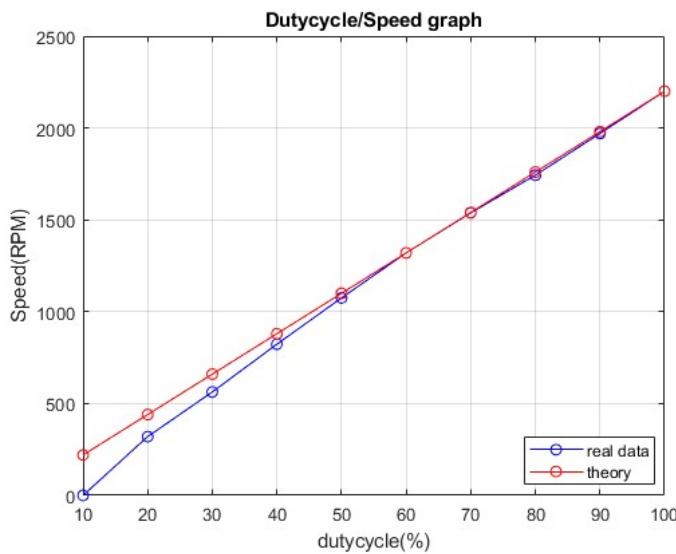
รูปที่ 18 แสดงความสัมพันธ์ระหว่างกระแสไฟฟ้าและแรงบิดของมอเตอร์เมื่อรับ Dynamic Load

จากการทดลองที่ 3 จะเป็นการสั่งการให้ Dynamic Load Motor ทำงานตามแรงบิดที่กำหนดเพื่อสังเกต พฤติกรรมของกระแสไฟฟ้าที่เหล่านมอเตอร์ ซึ่งสามารถสรุปได้ว่าเมื่อแรงบิดมากขึ้นกระแสที่ใช้ก็จะมากขึ้นตาม

1.8.3 ผลการทดลองที่ 4

การทดลองที่ 4 ได้ถูกแบ่งออกเป็น 2 การทดลองย่อย ดังนี้

- 1) การทดลองย่อยที่ 1 จะอธิบายเกี่ยวกับความสัมพันธ์ระหว่าง Duty Cycle, ความเร็วรอบการหมุนของมอเตอร์ และกระแสไฟฟ้าที่แหล่งกำเนิดมอเตอร์



รูปที่ 19 กราฟแสดงความสัมพันธ์ระหว่าง Duty Cycle และความเร็วรอบการหมุนของมอเตอร์

จากราฟทดลองหาความสัมพันธ์ระหว่าง Duty Cycle กับความเร็วรอบการหมุนของมอเตอร์พบว่า ความเร็วรอบและ Duty Cycle มีความสัมพันธ์กันใกล้เคียงเชิงเส้น โดยคนละผู้จัดทำทราบว่า Duty Cycle จะทำหน้าที่เสมือนลดระดับแรงดันโดยมีสูตรคำนวณดังนี้

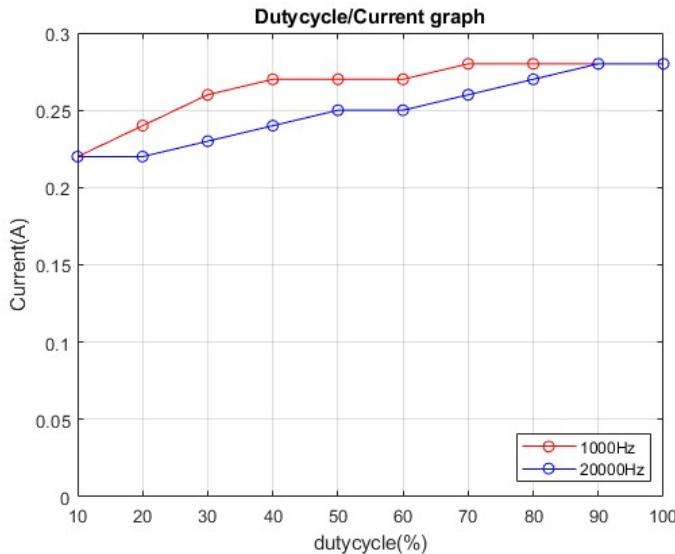
$$V_{\text{out}} = V_{\text{in}} \times \text{Duty Cycle}$$

ซึ่งจากสูตรคำนวณจะแสดงให้เห็นว่าระดับแรงดันไฟฟ้าจะลดลงตามขนาดของ Duty Cycle ซึ่งหากจะลดระดับแรงดันไฟฟ้าลดลงจะทำให้ความเร็วรอบการหมุนของมอเตอร์ลดลงตาม จากการวัดและวิเคราะห์ผลในการทดลองที่ 1 และ 2 พบว่า ณ แรงดัน 12 V มอเตอร์มีความเร็วรอบที่ 2200 RPM ซึ่งในสูตรคำนวณเดียวกันแต่เปลี่ยนจากแรงดันไฟฟ้าเป็นความเร็วรอบจะได้สูตรคำนวณดังนี้

$$\omega_{\text{out}} = \omega_{\text{NL}} \times \text{Duty Cycle}$$

ซึ่งสามารถสร้างกราฟได้ตามกราฟเส้นสีแดง และจากการทดลองพบว่าค่าจากทฤษฎีมีความใกล้เคียงค่าที่วัดได้เป็นอย่างมาก เพียงแต่จากการทดลองอาจจะมีการคลาดเคลื่อนจาก Load ที่กระทำต่อตัวมอเตอร์เพื่อใช้วัดความเร็วรอบทำให้การวัดนั้นคลาดเคลื่อนรวมถึงทำให้มอเตอร์ไม่

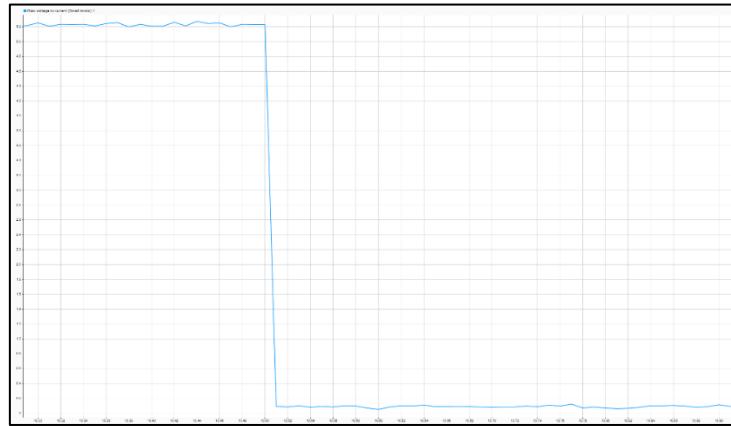
สามารถทำการทดลองได้ใกล้เคียงกับความเป็นอุดมคติ แต่ก็เพียงพอที่จะสรุปว่า Duty Cycle ส่งผลต่อความเร็วรอบของมอเตอร์ในลักษณะเชิงเส้น



รูปที่ 20 แสดงกราฟที่แสดงความสัมพันธ์ระหว่างกระแสไฟฟ้าและ Duty Cycle

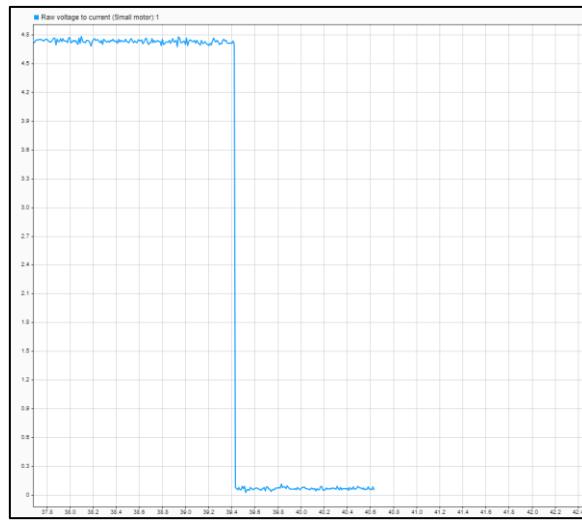
จากการทดลองระหว่าง Duty Cycle และกระแสไฟฟ้าที่แหล่งจ่ายไฟ ณ ความถี่ที่ 1000 Hz และ 20000 Hz จากทฤษฎี หากความถี่ของ PWM ต่ำจะส่งผลให้เกิดการใช้กระแสไฟฟ้ามากกว่า การใช้ PWM ที่ความถี่สูง ซึ่งจากการทดลองก็พบว่า การใช้ความถี่ที่สูงขึ้นมีส่วนทำให้กระแสไฟฟ้าที่แหล่งจ่ายไฟลดลงจากกราฟ แต่จากการทดลองไม่สามารถสรุปความเป็นเชิงเส้นของกระแสไฟฟ้าได้

- 2) การทดลองย่ออย่างที่ 2 จะอธิบายเกี่ยวกับพฤติกรรมของกระแสไฟฟ้าที่แหล่งผ่านมอเตอร์ ผ่าน mode การทำงานต่างๆของ Motor drive



รูปที่ 21 แสดงถึงกราฟกระแสที่ลดลงแบบ Slow Decay ในช่วง Off Time ในขณะที่ใช้โหมดการทำงานแบบ Sign Magnitude

Sign Magnitude (Slow Decay) การควบคุมมอเตอร์แบบ Sign Magnitude ใช้สายไฟทั้งหมด 2 เส้นในการควบคุม ห้อง PWM ที่ควบคุมในเรื่องของความเร็ว และ DIR ที่ควบคุมในเรื่องของทิศทาง ในส่วนของการลดลงของกระแสไฟฟ้าของ Slow Decay จากกระแสไฟฟ้าในช่วง Stall Torque ถึงช่วงที่ไม่ได้ทำการจ่าย PWM มีการใช้เวลาในการเปลี่ยนแปลงจาก 5 A ถึง 0 A เป็นเวลา 0.01 วินาที



รูปที่ 22 แสดงถึงกราฟกระแสที่ลดลงแบบ Fast Decay ในช่วง Off Time ในขณะที่ใช้โหมดการทำงานแบบ Lock Anti Phase

Lock Anti Phase Drive (Fast Decay) การควบคุมมอเตอร์ด้วยรูปแบบ Sign Magnitude ใช้สายไฟเพียงแค่หนึ่งเส้นเท่านั้น โดยหาก PWM มี Duty Cycle มากกว่า 50% จะทำให้มอเตอร์หมุนตามเข็มนาฬิกา หากมี Duty Cycle น้อยกว่า 50% จะหมุนทวนเข็มนาฬิกา และ หากมี Duty Cycle ที่ 50% ทำให้มอเตอร์ไม่เกิดการหมุน ในส่วนของการลดลงของกระแสไฟฟ้าของ Slow Decay จากกระแสไฟฟ้าในช่วง Stall Torque ถึงช่วงที่ไม่ได้ทำการจ่าย PWM มีการใช้เวลาในการเปลี่ยนแปลงจาก 5 A ถึง 0 A เป็นเวลา 0.005 วินาที ซึ่งเร็วกว่าการควบคุมแบบ Sign Magnitude ที่เป็น Slow Decay

1.9 สรุปผล

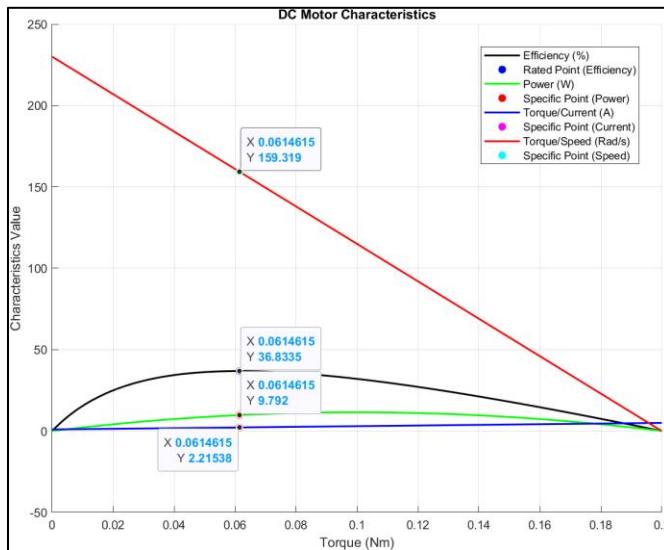
ในการทดลองที่ 1 และ 2 ทำให้คณะผู้จัดทำทราบค่าคงที่ที่ได้มาจากการทดลองได้แก่ ความเร็วของขณะไม่มีโหลด ($\omega_{No-Load}$), แรงบิดสูงสุด (τ_{Stall}), กระแสสูงสุด (i_{Stall}), กระแสขณะไม่มีโหลด ($i_{No-Load}$) และแรงดันไฟฟ้าที่จ่าย (V_{in}) ทำให้คณะผู้จัดทำสามารถที่จะหาความสัมพันธ์ระหว่าง แรงบิดต่อความเร็ว, แรงบิดต่อกระแส, กำลังไฟ และประสิทธิภาพการทำงานต่อได้จากการทดลองที่ 3 และ 4 ดังนี้ ในการทดลองที่ 3 ทำให้ทราบถึงการดึงกระแสของมอเตอร์เทียบกับแรงบิดที่มอเตอร์สามารถสร้างได้ ในการทดลองที่ 4 ทำให้ทราบถึงความสัมพันธ์ของความเร็วมอเตอร์ต่อกระแสไฟฟ้าที่แหล่งผ่านมอเตอร์ ซึ่งควบคุมผ่านความมากน้อยของร้อยละ Duty Cycle ผ่านสัญญาณในรูปแบบของ PWM และในการควบคุมโดยใช้ความถี่ที่สูงขึ้นจะสามารถลดปริมาณกระแสที่มอเตอร์ดึงไปใช้ได้ ซึ่งในการควบคุม Brush DC Motor นั้นสามารถแบ่งออกได้เป็นสองประเภทคือ Sign Magnitude และ Lock Anti Phase ซึ่งมีความแตกต่างกันตรงที่หากเป็นการควบคุมด้วย Sign Magnitude จะมีการใช้สายสัญญาณสองเส้นคือ PWM และ DIR ใน การควบคุม ซึ่งในช่วงเวลา Off Time มอเตอร์นั้นจะมีพกติกรรมในการค่อย ๆ ลดแรงดันภายในระบบลง (Slow Decay) แต่ใน mode Lock Anti Phase นั้นจะใช้สายสัญญาณเพียง 1 เส้นในการควบคุม ทิศทางและความเร็ว ซึ่งในช่วงเวลา Off Time มอเตอร์นั้นจะมีพกติกรรมในการลดแรงดันภายในระบบอย่างกระแทก (Fast Decay) ซึ่งจากการทำการทดลองทั้งหมดทำให้คณะผู้จัดทำสามารถสร้างกราฟ Motor Characteristic ได้ครบถ้วน

```

1 % Efficiency Curve
2 Torque1 = linspace(0, 0.2, 1000); % Torque from 0 to 0.2 Nm
3 n = ((-230/0.2)^(Torque1.^2) + 230*Torque1) ./ (((5-0.98)/0.2)*Torque1.^12 + 0.98*12)*100;
4
5 % Find the maximum efficiency and its corresponding torque
6 [maxEfficiency, idx] = max(n); % maxEfficiency is the highest value, idx is its index
7 ratedTorque = Torque1(idx); % Find the corresponding torque
8
9 % Power Curve
10 Torque2 = linspace(0, 0.2, 1000); % Torque from 0 to 0.2 Nm
11 P = -(230/0.2)^(Torque2.^2) + 230*Torque2;
12
13 % Find the Power at a specific Torque = 0.0614 Nm
14 specificTorque = 0.0614615; % Desired torque value
15 powerAtSpecificTorque = -(230/0.2)*(specificTorque.^2) + 230*specificTorque;
16
17 % Torque-Current Curve
18 xTorqueCurrent = [0, 0.2]; % Torque range
19 yCurrent = [0.98, 5]; % Current range
20 currentAtSpecificTorque = interp1(xTorqueCurrent, yCurrent, specificTorque, 'linear'); % Interpolation for current
21
22 % Torque-Speed Curve
23 xTorqueSpeed = [0, 0.2]; % Torque range
24 ySpeed = [230, 0]; % Speed range
25 speedAtSpecificTorque = interp1(xTorqueSpeed, ySpeed, specificTorque, 'linear'); % Interpolation for speed
26
27 % Combined Plot
28 figure;
29 hold on;
30
31 % Plot Efficiency Curve
32 plot(Torque1, n, 'black', 'LineWidth', 1.5, 'DisplayName', 'Efficiency (%)');
33 plot(ratedTorque, maxEfficiency, 'bo', 'MarkerFaceColor', 'b', 'DisplayName', 'Rated Point (Efficiency)');
34
35 % Plot Power Curve
36 plot(Torque2, P, 'g', 'LineWidth', 1.5, 'DisplayName', 'Power (W)');
37 plot(specificTorque, powerAtSpecificTorque, 'ro', 'MarkerFaceColor', 'r', 'DisplayName', 'Specific Point (Power)');
38
39 % Plot Torque-Current Curve
40 plot(xTorqueCurrent, yCurrent, '-b', 'LineWidth', 1.5, 'DisplayName', 'Torque/Current (A)');
41 plot(specificTorque, currentAtSpecificTorque, 'mo', 'MarkerFaceColor', 'm', 'DisplayName', 'Specific Point (Current)');
42
43 % Plot Torque-Speed Curve
44 plot(xTorqueSpeed, ySpeed, '-r', 'LineWidth', 1.5, 'DisplayName', 'Torque/Speed (Rad/s)');
45 plot(specificTorque, speedAtSpecificTorque, 'co', 'MarkerFaceColor', 'c', 'DisplayName', 'Specific Point (Speed)');
46
47 % Configure Labels, Title, and Legend
48 xlabel('Torque (Nm)');
49 ylabel('Characteristics');
50 title('DC Motor Characteristics');
51 grid on;
52 legend('Location', 'Best');
53
54 % Display Results
55 disp(['Rated Torque: ', num2str(ratedTorque), ' Nm']);
56 disp(['Maximum Efficiency: ', num2str(maxEfficiency), ' %']);
57 disp(['Power at Torque ', num2str(specificTorque), ' Nm: ', num2str(powerAtSpecificTorque), ' W']);
58 disp(['Current at Torque ', num2str(specificTorque), ' Nm: ', num2str(currentAtSpecificTorque), ' A']);
59 disp(['Speed at Torque ', num2str(specificTorque), ' Nm: ', num2str(speedAtSpecificTorque), ' Rad/s']);

```

รูปที่ 23 แสดงโค้ดสำหรับใช้สร้างกราฟ Motor Characteristic โดยระบุจุดที่มีประสิทธิภาพสูงสุด



รูปที่ 24 แสดงลักษณะ Motor Characteristic โดยระบุจุดที่มีประสิทธิภาพสูงสุด

1.10 อภิรายผล

จากการทดลองพบว่า มอเตอร์ทุกประเภทสามารถวิเคราะห์ลักษณะเฉพาะได้จากความสัมพันธ์ของตัวแปรสำคัญ เช่น แรงบิดกับความเร็ว (Torque/Speed), แรงบิดกับกระแส (Torque/Current), กำลังไฟฟ้า (Power) และประสิทธิภาพ (Efficiency) โดยก่อนที่จะคำนวณค่าความสัมพันธ์เหล่านี้ จะเป็นต้องกำหนดค่าลักษณะเฉพาะของมอเตอร์ ซึ่งประกอบด้วย 5 ค่า ได้แก่ ความเร็วรอบขณะไม่มีโหลด ($\omega_{No-Load}$), แรงบิดสูงสุด (T_{Stall}), กระแสสูงสุด (i_{Stall}), กระแสขณะไม่มีโหลด ($i_{No-Load}$) และแรงดันไฟฟ้าที่จ่าย (V_{in}) หลังจากได้ค่าลักษณะเฉพาะเหล่านี้แล้ว สามารถนำค่าดังกล่าวเข้าสมการเพื่อวิเคราะห์ค่าที่จุดแรงบิดใด ๆ ได้ ซึ่งการคำนวณนี้มีความสำคัญต่อการหาค่าประสิทธิภาพสูงสุดของมอเตอร์หรือค่า Rated ที่ต้องการ การทดลองยังแสดงให้เห็นถึงการเปลี่ยนแปลงของกระแสไฟฟ้าเมื่อแรงต้านเพิ่มขึ้น โดยพบว่าความสัมพันธ์ระหว่างแรงบิดและกระแสแม่ลักษณะใกล้เคียงกับเชิงเส้นกล่าวคือ เมื่อแรงบิดเพิ่มขึ้น มอเตอร์ต้องใช้กระแสมากขึ้น นอกจากนี้ยังสามารถคำนวณแรงบิดของมอเตอร์ได้จากกระแสที่ไหลผ่านมอเตอร์โดยใช้สมการ Lorentz Force – Motor's Motion ในส่วนของการควบคุมความเร็วของมอเตอร์นั้น สามารถทำได้ผ่านวิธีการ PWM (Pulse Width Modulation) โดยที่ความเร็วรอบของมอเตอร์แปรัตน์ตรงกับแรงดันไฟฟ้า ซึ่งแรงดันดังกล่าวถูกควบคุมผ่าน Duty Cycle ของ PWM กล่าวคือ หาก Duty Cycle สูง แรงดันเสมือนจะสูงขึ้น ส่งผลให้ความเร็วเพิ่มขึ้น ในทางกลับกัน หาก Duty Cycle ต่ำ แรงดันเสมือนจะลดลงและความเร็วลดลงตาม นอกจากนี้ การทดลองยังพบว่าหาก PWM มีความถี่สูง จะช่วยลดกระแสไฟฟ้าที่มอเตอร์ใช้ได้จริง สำหรับการควบคุมมอเตอร์แบบ Sign Magnitude และ Lock Anti-Phase พบรความแตกต่างที่น่าสนใจ โดยวิธี Sign Magnitude ใช้สายไฟ 2 เส้นในการควบคุม คือ PWM สำหรับรับความเร็ว และ DIR สำหรับเปลี่ยนทิศทาง เป็นวิธีที่ไม่ต้องจ่ายกระแสให้กับบอร์ดไดรฟ์ตลอดเวลาและลดกระแสลงอย่างช้า (Slow Decay) โดยใช้เวลา 0.01 วินาทีในการลดกระแสจาก 5V เหลือ 0V ขณะที่ Lock Anti-Phase ใช้สายไฟเพียงเส้นเดียวในการควบคุมทั้งความเร็ว และทิศทางผ่าน Duty Cycle ของ PWM แต่ต้องจ่ายกระแสให้บอร์ดไดรฟ์ตลอดเวลา และการลดกระแสเกิดขึ้นอย่างรวดเร็ว (Fast Decay) โดยใช้เวลาเพียง 0.05 วินาทีในการลดกระแสจาก 5V เหลือ 0V

1.11 ข้อเสนอแนะ

- 1) ปรับปรุงความแม่นยำในการวัดค่าพื้นฐานของมอเตอร์
 - ใช้เครื่องมือวัดที่มีความละเอียดและเที่ยงตรง เช่น อุปกรณ์วัดกระแสและแรงดันแบบดิจิทัลที่มีค่าความผิดพลาดต่ำ
 - ลดผลกระทบจากปัจจัยภายนอก เช่น การสั่นสะเทือนหรือความไม่ตรงแนว (Misalignment) ระหว่างมอเตอร์และโหลด
- 2) ลดการสูญเสียพลังงานในระบบทดลอง
 - พิจารณาใช้วัสดุและโครงสร้างที่ลดแรงเสียดทานของโหลด เช่น การหล่อลิ่นก้านส่งกำลัง
 - ควบคุมสภาพแวดล้อม เช่น ลดฝุ่นหรือสิ่งสกปรกที่อาจรบกวนการหมุนของมอเตอร์
- 3) พิจารณาผลกระทบจากค่าความถี่ PWM ที่สูงขึ้น
 - ทดสอบค่าความถี่ PWM ในช่วงต่างๆ ที่มากกว่า 20 kHz เพื่อตรวจสอบการลดกระแทกไฟฟ้าที่อาจเกิดขึ้นเพิ่มเติม
 - ใช้อุปกรณ์ตรวจจับการรบกวนแม่เหล็กไฟฟ้า (EMI) เพื่อประเมินผลกระทบของ PWM ความถี่สูงต่อระบบโดยรวม
- 4) ทดสอบมอเตอร์ในช่วงโหลดที่หลากหลายขึ้น
 - เพิ่มช่วงของแรงบิด (Torque) ระหว่าง 0 ถึง τ_{ST} ให้ละเอียดขึ้น เพื่อศึกษาความสัมพันธ์ที่ชัดเจนยิ่งขึ้น
 - ใช้ Dynamic Load ที่สามารถควบคุมแรงบิดได้แม่นยำ เช่น มอเตอร์ที่ควบคุมด้วย Torque Feedback
- 5) เปรียบเทียบโหมดควบคุมมอเตอร์เพิ่มเติม
 - ทดสอบโหมดการควบคุมอื่นๆ เช่น Asynchronous Sign-Magnitude Drive หรือ Current Mode Control เพื่อเปรียบเทียบประสิทธิภาพ
 - วิเคราะห์พฤติกรรมการทำงานของโหมดควบคุมในสถานการณ์พิเศษ เช่น การ starters หรือหยุดกะทันหัน
- 6) เพิ่มกระบวนการกรองสัญญาณที่เหมาะสม
 - ใช้ฟิลเตอร์ที่หลากหลาย เช่น Band-Pass หรือ Adaptive Filtering เพื่อจัดการสัญญาณรบกวนที่อาจเกิดขึ้นในระบบ
 - พิจารณาใช้เทคนิคลดสัญญาณรบกวนขั้นสูง เช่น Kalman Filter สำหรับข้อมูลที่ได้จากการเซ็นเซอร์
- 7) ปรับปรุงขั้นตอนการคำนวณ
 - ใช้ซอฟต์แวร์วิเคราะห์ข้อมูล เช่น MATLAB หรือ Python เพื่อช่วยในการคำนวณค่าและสร้างกราฟความสัมพันธ์
 - เพิ่มการวิเคราะห์จุดทำงานที่ประสิทธิภาพสูงสุด (Efficiency Peak) อย่างละเอียด เช่น การหาค่าที่เหมาะสมของ τ_{Rated} , i_{Rated} และ ω_{Rated}

8) ศึกษาผลกระทบจากการเสื่อมสภาพของมอเตอร์

- ทำการทดลองเปรียบเทียบระหว่างมอเตอร์ใหม่และมอเตอร์ที่ใช้งานมาแล้ว เพื่อประเมินผลกระทบของการสึกหรอ
- วางแผนการบำรุงรักษาเพื่อยืดอายุการใช้งานของมอเตอร์ในระบบทดลอง

9) ปรับปรุงระบบทดลองให้ครอบคลุมมากขึ้น

- เพิ่มการวัดพารามิเตอร์อื่นๆ เช่น อุณหภูมิของมอเตอร์ขณะทำงาน เพื่อประเมินผลกระทบต่อประสิทธิภาพ
- ใช้ระบบอัตโนมัติในการเก็บข้อมูล เช่น การบันทึกค่าด้วย Data Logger เพื่อลดความผิดพลาดจากมนุษย์

10) ขยายผลการทดลองไปยังมอเตอร์ประเภทอื่น

- ทดลองกับมอเตอร์ชนิดอื่น เช่น BLDC หรือ Stepper Motor เพื่อศึกษาความแตกต่างของลักษณะเฉพาะและพฤติกรรม
- เปรียบเทียบข้อดีและข้อเสียระหว่างมอเตอร์แต่ละประเภท เพื่อพัฒนาการใช้งานในสถานการณ์ต่างๆ

1.12 เอกสารอ้างอิง

- [1] N.R. Syed, "How to Read a DC Motor Datasheet," Available online [ມ.ປ.ປ.]:
<https://nrsyed.com/2018/01/21/how-to-read-a-dc-Motor-datasheet/>
- [2] DwyerOmega, "DC Motor Basics - Explained," Available online [Jan 22, 2020]:
<https://www.youtube.com/watch?v=d7y9xg-Bc18>
- [3] FitRox, "Understanding PWM Frequency in Motor Control," Available online [De 21, 2024]:
<https://fitrox.lnwshop.com/article/102/>
- [4] kruvisart SaraElect, "PWM Frequency Analysis for Motor Applications," Available online [Mar 21, 2024]: <https://www.youtube.com/watch?v=m8jpOX3h60k>
- [5] Grant Maloy Smith, "What is Signal Conditioning?" Available online [Nov 21, 2024]:
<https://dewesoft.com/blog/what-is-signal-Conditioning>
- [6] Kistler, "Signal Processing for Advanced Measurement Systems," Available online [ມ.ປ.ປ.]:
<https://www.kistler.com/INT/en/signal-processing/C00000155>
- [7] NamProvision, "บทความ: การทำงานของ PWM และการควบคุมมอเตอร์," Available online [Jun 6, 2020]: <https://www.blockdit.com/posts/5edb528e6b1c154a1ccbcd6>

การทดลองที่ 2

Stepper Motor

2.1 จุดประสงค์

- 2.1.1 เพื่อเข้าใจหลักการทำงานและส่วนประกอบของ Stepper Motor
- 2.1.2 เพื่อเข้าใจความสัมพันธ์ระหว่างความถี่ของสัญญาณ และความเร็วของ Stepper Motor
- 2.1.3 เพื่อเข้าใจและประยุกต์ใช้รูปแบบการ Drive ทั้ง Full-Step, Half-Step, และ Micro-Step Drive และเปรียบเทียบผลกระทบที่มีต่อความแม่นยำในการควบคุมตำแหน่ง ความเร็ว และแรงบิดของ Stepper Motor
- 2.1.4 เพื่อเข้าใจประเภทของ Stepper Motor Rotor และเปรียบเทียบคุณสมบัติ และสามารถเลือก Rotor Type ที่เหมาะสมกับการใช้งาน
- 2.1.5 เพื่อเข้าใจวิธีการขับเคลื่อน Stepper Drive Type ต่างๆ และวิเคราะห์ข้อดีและข้อเสียของ Stepper Drive Type และสามารถเลือก Stepper Drive Type ให้เหมาะสมกับระบบควบคุม
- 2.1.6 เพื่อเข้าใจถึงปัจจัยที่ทำให้เกิดการ Loss Step ใน Stepper Motor และผลกระทบที่มีต่อประสิทธิภาพของมอเตอร์ พร้อมแนวทางการป้องกันและแก้ไข
- 2.1.7 เพื่อเข้าใจ Stepper Motor Characteristic และ Stepper Motor Specification
- 2.1.8 เพื่อเข้าใจการควบคุม Stepper Motor ด้วย MATLAB และ Simulink โดยอาศัยบอร์ด Nucleo STM32G474RE เป็นตัวประมวลผลหลัก และใช้สัญญาณจาก Incremental Encoder เป็น Input พร้อมแสดงผลแบบ Real Time ด้วย Data Inspector ในรูปแบบกราฟแสดงความเร็วเชิงมุมและกระแสไฟฟ้า

2.2 สมมติฐาน

2.2.1 การทดลองที่ 1

- 1) ความเร็วเชิงมุขของ Stepper Motor จะเปลี่ยนแปลงโดยแปรผันตรงกับความถี่ของสัญญาณควบคุมที่จ่ายเข้าสู่ Stepper Motor
- 2) การเพิ่มความถี่ของสัญญาณควบคุมจะทำให้ความเร็วของ Stepper Motor เพิ่มขึ้นในลักษณะเชิงเส้นในช่วงแรก แต่เมื่อถึงค่าความถี่สูงสุดที่ Stepper Motor สามารถตอบสนองได้ (Saturation Point) ความเร็วจะคงที่และไม่เพิ่มขึ้นต่อไป แม้ว่าความถี่ของสัญญาณควบคุมจะเพิ่มขึ้น
- 3) การ Loss Step ใน Stepper Motor จะเกิดขึ้นเมื่อแรงบิดของมอเตอร์ไม่เพียงพอ และภาระโหลดที่สูงเกินไป หรือ Acceleration Ramp ที่สูงเกินไป

2.2.2 การทดลองที่ 2

- 1) การ Drive Stepper Motor ในรูปแบบ Micro Step ช่วยลดแรงบิดกระเพื่อม (Torque Ripple) และทำให้การเคลื่อนที่ของมอเตอร์ราบรื่นกว่าการ Drive แบบ Full-Step หรือ Half-Step
- 2) Stepper Drive Type แบบ Chopper Drive จะมีประสิทธิภาพสูงกว่าการใช้ L/R Drive โดยเฉพาะในงานที่ต้องการแรงบิดและความเร็วสูง

2.3 ตัวแปร

2.3.1 การทดลองที่ 1

ตัวแปรต้น: ความถี่ของสัญญาณควบคุมที่จ่ายเข้าสู่ Stepper Motor, แรงบิดของมอเตอร์, ภาระโหลด, และ Acceleration Ramp

ตัวแปรตาม: ความเร็วเชิงมุขของ Stepper Motor และการเกิด Loss Step

ตัวแปรควบคุม: แรงดันไฟฟ้าที่จ่ายให้มอเตอร์, ประเภทของ Stepper Motor และอุณหภูมิของระบบ

2.3.2 การทดลองที่ 2

ตัวแปรต้น: รูปแบบการ Drive (Micro Step, Full-Step, Half-Step) และประเภทของ Stepper Drive (Chopper Drive, L/R Drive)

ตัวแปรตาม: แรงบิดกระเพื่อม (Torque Ripple), ความราบรื่นของการเคลื่อนที่, และประสิทธิภาพของมอเตอร์ในเรื่องแรงบิดและความเร็ว

ตัวแปรควบคุม: แรงดันไฟฟ้าที่จ่ายให้มอเตอร์, ความถี่ของสัญญาณควบคุม, ภาระโหลด, และประเภทของ Stepper Motor

2.4 นิยามศัพท์เฉพาะ

- 2.4.1 Stepper Motor: มอเตอร์ไฟฟ้าที่ขับเคลื่อนด้วยพัลส์ไฟฟ้า โดยการหมุนแต่ละครั้งจะถูกแบ่งออกเป็นขั้นตอนที่มีมุมคงที่ เช่น 1.8° หรือ 0.9° ขึ้นอยู่กับชนิดของมอเตอร์ ซึ่งช่วยให้สามารถควบคุมตำแหน่งได้อย่างแม่นยำ
- 2.4.2 Step Loss: การสูญเสียขั้นตอนในการหมุนของ Stepper Motor อันเนื่องมาจากการไม่สามารถตอบสนองต่อสัญญาณพัลส์ได้อย่างถูกต้อง สาเหตุอาจเกิดจากการเร่งความเร็วเกินกำหนดหรือโหลดที่หนักเกินไป
- 2.4.3 Micro Step Resolution: ความละเอียดในการควบคุมการหมุนของ Stepper Motor โดยการแบ่งขั้นตอนการหมุนย่อยลงไปในระดับที่เล็กกว่า Full Step ซึ่งช่วยเพิ่มความราบรื่นและความแม่นยำ
- 2.4.4 Pullout Torque: แรงบิดสูงสุดที่ Stepper Motor สามารถสร้างได้เมื่อหมุนที่ความเร็วหนึ่ง ๆ หากโหลดเกินค่าแรงบิดนี้ มอเตอร์จะสูญเสียขั้นตอนการหมุน
- 2.4.5 Butterworth Low Pass Filter: ตัวกรองสัญญาณที่ออกแบบมาให้ลดความแรงของสัญญาณความถี่สูงขณะเดียวกันยังคงรักษาความแรงของสัญญาณในความถี่ต่ำอย่างสม่ำเสมอ เพื่อลดสัญญาณรบกวน
- 2.4.6 Nyquist-Shannon Sampling Theorem: ทฤษฎีการสูมตัวอย่างที่ระบุว่าความถี่การสูมตัวอย่างต้องมีค่ามากกว่าสองเท่าของความถี่สูงสุดในสัญญาณ เพื่อป้องกันการเกิด Aliasing
- 2.4.7 Quantization Error: ความคลาดเคลื่อนที่เกิดขึ้นระหว่างค่าของสัญญาณ Analog จริงกับค่า Digital ที่ใกล้เคียงที่สุดในกระบวนการแปลงสัญญาณ
- 2.4.8 Encode: อุปกรณ์ที่แปลงการเคลื่อนที่เชิงกลของมอเตอร์เป็นสัญญาณไฟฟ้าเพื่อตรวจจับตำแหน่งความเร็ว หรือทิศทาง
- 2.4.9 Signal Conditioning: กระบวนการปรับแต่งสัญญาณให้เหมาะสมสำหรับการประมวลผลหรือการใช้งาน เช่น การกรอง การขยายสัญญาณ หรือการแปลงสัญญาณ
- 2.4.10 Real-Time Clock (RTC): นาฬิกาที่ให้ค่าของเวลาในปัจจุบันที่แม่นยำและสอดคล้องกับเวลาจริง ใช้สำหรับการอ้างอิงเวลาที่ถูกต้องในระบบควบคุม
- 2.4.11 Trapezoidal: รูปแบบการเร่งหรือชะลอความเร็วของ Stepper Motor โดยความเร็วจะเพิ่มขึ้นอย่างคงที่ในช่วง Ramp-Up จนถึงความเร็วคงที่ (Constant Speed) และลดลงอย่างคงที่ในช่วง Ramp-Down ซึ่งช่วยให้การเคลื่อนที่ราบรื่นขึ้น
- 2.4.12 Load Inertia: ความเรื้อรังของโหลดที่เข้มต่อกับมอเตอร์ ซึ่งส่งผลต่อการเริ่มต้นและหยุดการหมุนของ Stepper Motor โดยโหลดที่มีความเรื้อรังจะต้องการแรงบิดมากขึ้นในการเปลี่ยนแปลงความเร็ว

- 2.4.13 Frictional Load: แรงเสียดทานที่เกิดขึ้นจากการทำงานของระบบ เช่น แรงเสียดทานระหว่างแกน มอเตอร์และโหลด ซึ่งส่งผลต่อประสิทธิภาพการทำงานของ Stepper Motor และความแม่นยำ
- 2.4.14 จุดแท็ปกลาง : จุดเชื่อมต่อกลางของชุดลวดใน Stepper Motor แบบ Unipolar ซึ่งช่วยให้การจ่ายกระแสในทิศทางเดียวสามารถสร้างสนามแม่เหล็กที่เปลี่ยนข้ามได้
- 2.4.15 SNR (Signal-to-Noise Ratio): อัตราส่วนระหว่างพลังงานของสัญญาณที่สนใจ (Signal) กับพลังงานของสัญญาณรบกวน (Noise) ซึ่งค่าที่สูงขึ้นบ่งชี้ว่าสัญญาณมีคุณภาพและความแม่นยำที่ดีขึ้น
- 2.4.16 Analog-to-Digital Converter (ADC): อุปกรณ์ที่แปลงสัญญาณอะนาล็อก เช่น แรงดันไฟฟ้า ให้เป็นสัญญาณดิจิทัลที่สามารถประมวลผลหรือจัดเก็บในระบบดิจิทัลได้
- 2.4.17 Saturation Point: จุดที่แรงดันหรือกระแสไฟฟ้าที่ขยายออกมากถึงค่ามากที่สุดที่วงจรหรืออุปกรณ์สามารถรองรับได้ หากเกินค่าดังกล่าว การทำงานจะไม่เป็นไปตามปกติ
- 2.4.18 Acceleration Ramp: การเพิ่มความเร็วของ Stepper Motor อย่างค่อยเป็นค่อยไปตามช่วงเวลา เพื่อหลีกเลี่ยงปัญหา Step Loss ที่อาจเกิดจากการเร่งเร็วเกินไป
- 2.4.19 Stepper Drive: ระบบควบคุมที่จ่ายสัญญาณไฟฟ้าไปยัง Stepper Motor เพื่อกำหนดทิศทางและความเร็วของการหมุน
- 2.4.20 Chopper Drive: เทคนิคการควบคุมกระแสใน Stepper Motor โดยการเปิด-ปิดกระแสไฟฟ้าด้วยความถี่สูง ช่วยลดการสูญเสียพลังงานและเพิ่มประสิทธิภาพของมอเตอร์
- 2.4.21 L/R Drive: ระบบควบคุมกระแสใน Stepper Motor ที่ใช้ความต้านทาน (R) และความเหนี่ยวนำ (L) ของชุดลวดในการจำกัดกระแสไฟฟ้า ทำให้เหมาะสมสำหรับการควบคุมความเร็วต่ำ
- 2.4.22 Rotary Switch ในโปรแกรม MATLAB Simulink: ตัวเลือกแบบหมุนที่ใช้ในโปรแกรม MATLAB Simulink สำหรับเลือกค่าหรือพารามิเตอร์ต่าง ๆ ในการควบคุมหรือการตั้งค่าระบบ
- 2.4.23 Block Constant: บล็อกในโปรแกรม MATLAB Simulink ที่ใช้กำหนดค่าคงที่ซึ่งจะถูกส่งไปยังบล็อกหรือโมดูลอื่น ๆ เพื่อใช้ในการประมวลผลหรือควบคุม
- 2.4.24 Fast Fourier Transform (FFT) :กระบวนการทางคณิตศาสตร์ที่แปลงสัญญาณในโดเมนเวลาให้เป็นโดเมนความถี่ เพื่อวิเคราะห์ส่วนประกอบความถี่ของสัญญาณ
- 2.4.25 Block Varying Low Pass Filter: บล็อกในโปรแกรม MATLAB Simulink ที่ใช้กรองสัญญาณความถี่สูงออกไป โดยสามารถปรับค่าพารามิเตอร์ได้ตามการเปลี่ยนแปลงของสัญญาณอินพุต
- 2.4.26 Cut Off Frequency :ค่าความถี่ที่กำหนดเป็นขอบเขตของตัวกรองสัญญาณ เช่น Lowpass Filter จะลดทอนความถี่ที่สูงกว่าค่าที่กำหนดนี้

2.4.27 MATLAB Function Block: บล็อกในโปรแกรม MATLAB Simulink ที่อนุญาตให้เขียนโค้ด MATLAB แบบกำหนดเอง เพื่อใช้ในการคำนวณหรือควบคุมในระบบ Simulink

2.5 นิยามเชิงปฏิบัติการ

- 2.5.1 Step Loss ของ Stepper Motor: กระบวนการที่ Stepper Motor สูญเสียความสามารถในการหมุนตามขั้นตอนที่กำหนด เนื่องจากความถี่ของสัญญาณควบคุมสูงเกินไปหรือแรงเสียดทานในระบบสูงจน Rotor ไม่สามารถหมุนตามได้ทัน โดยวัดจากความแตกต่างระหว่างมุมการหมุนที่ควรได้กับมุมการหมุนจริงของ Rotor
- 2.5.2 Micro Step Resolution: ค่าความละเอียดในการหมุนของ Stepper Motor ซึ่งแบ่งขั้นตอนการหมุนให้เล็กลง เช่น 1/2 Step, 1/4 Step, หรือ 1/32 Step ใช้กำหนดโดย Pin M0, M1, และ M2 ของ Stepper Driver เพื่อเพิ่มความแม่นยำในการหมุน
- 2.5.3 Stepper Frequency: ความถี่ของสัญญาณพัลส์ที่ส่งไปยัง Stepper Motor ซึ่งส่งผลต่อความเร็วรอบของมอเตอร์โดยตรง โดยค่าความถี่ที่สูงขึ้นจะเพิ่มความเร็วในการหมุน
- 2.5.4 Trapezoidal: รูปแบบการเร่งหรือชะลอความเร็วใน Stepper Motor โดยแบ่งการเพิ่มและลดความเร็วออกเป็น 3 ช่วง ได้แก่ Ramp-Up (เร่งความเร็ว), Constant Speed (ความเร็วคงที่), และ Ramp-Down (ลดความเร็ว) เพื่อให้มอเตอร์หมุนได้ราบรื่นและป้องกันการสูญเสียขั้นตอน
- 2.5.5 Load Inertia: ค่าความเรื้อรังของโหลดที่เข้ามาร่วมกับ Stepper Motor ซึ่งกำหนดลักษณะการตอบสนองของมอเตอร์ในการเริ่มต้นหรือหยุดการหมุน โดยมีผลต่อความแม่นยำและแรงบิดที่ต้องการ
- 2.5.6 Frictional Load: แรงเสียดทานระหว่างส่วนที่เคลื่อนที่ของมอเตอร์และโหลดที่เข้ามาร่วม ทำให้เกิดการสูญเสียพลังงานและส่งผลต่อการทำงานของ Stepper Motor เช่น ความแม่นยำและความราบรื่นในการหมุน
- 2.5.7 จุดแท้ที่ปอกลา : ตำแหน่งของการเข้ามาร่วมกับโหลดของ Stepper Motor แบบ Unipolar ที่ใช้เปลี่ยนทิศทางกระแสไฟฟ้าภายในชุดโหลด โดยไม่ต้องเปลี่ยนขั้วของแหล่งจ่ายไฟ
- 2.5.8 SNR (Signal-to-Noise Ratio): อัตราส่วนระหว่างความแรงของสัญญาณที่สนใจกับความแรงของสัญญาณรบกวน โดยค่าที่สูงขึ้นบ่งชี้ว่ามีสัญญาณรบกวนต่ำและข้อมูลมีคุณภาพดี
- 2.5.9 Analog-to-Digital Converter (ADC) : อุปกรณ์ที่แปลงสัญญาณอะนาล็อก เช่น แรงดันไฟฟ้าให้เป็นสัญญาณดิจิทัล เพื่อให้สามารถประมวลผลหรือจัดเก็บข้อมูลในระบบคอมพิวเตอร์
- 2.5.10 Saturation Point: ค่าขีดจำกัดที่ Stepper Motor หรือวงจรไม่สามารถตอบสนองต่อแรงดันหรือกระแสที่เพิ่มขึ้นได้อีกต่อไป ทำให้เกิดการทำงานผิดปกติ
- 2.5.11 Acceleration Ramp : การเพิ่มความเร็วของ Stepper Motor แบบค่อยเป็นค่อยไปตามช่วงเวลา เพื่อป้องกันการสูญเสียขั้นตอน (Step Loss) และช่วยให้มอเตอร์หมุนได้อย่างราบรื่น

- 2.5.12 Stepper Drive: ระบบที่จ่ายสัญญาณไฟฟ้าให้ Stepper Motor เพื่อควบคุมทิศทาง ความเร็ว และการทำงานของมอเตอร์อย่างแม่นยำ
- 2.5.13 Chopper Drive: เทคนิคการควบคุมกระแสใน Stepper Motor โดยใช้การเปิด-ปิดกระแสตัววยความเร็วสูง เพื่อรักษากระแสตัวบกกระแสที่เหมาะสม ลดการสูญเสียพลังงาน และเพิ่มประสิทธิภาพ
- 2.5.14 L/R Drive: วิธีการควบคุมกระแสใน Stepper Motor โดยใช้คุณสมบัติของความต้านทาน (R) และความเนื่ิยวนำ (L) ในขดลวด เพื่อกำหนดการทำงานของมอเตอร์
- 2.5.15 Rotary Switch ในโปรแกรม MATLAB Simulink: เครื่องมือที่ใช้ใน Simulink สำหรับการเลือกค่าหรือพารามิเตอร์ต่าง ๆ เช่น การกำหนดความละเอียดของ Micro Step หรือการควบคุมทิศทางการหมุน
- 2.5.16 Incremental Encoder: อุปกรณ์ที่ใช้ตรวจสอบจับมุมการหมุนของ Stepper Motor โดยวัดสัญญาณพัลส์ที่สร้างขึ้นจากการหมุน และแปลงเป็นข้อมูลเชิงมุมในรูปดิจิทัล
- 2.5.17 Low Pass Filter: เครื่องมือสำหรับกรองสัญญาณรบกวนความถี่สูงออกจากสัญญาณที่วัดได้ โดยปล่อยให้เฉพาะความถี่ต่ำที่สนใจผ่านไป เพื่อปรับปรุงความแม่นยำของข้อมูล
- 2.5.18 Block Constant: ส่วนประกอบใน MATLAB Simulink ที่ใช้กำหนดค่าคงที่เพื่อส่งต่อให้บล็อกอื่นในระบบ เช่น การกำหนดความถี่หรือค่ากระแส
- 2.5.19 Fast Fourier Transform (FFT): กระบวนการวิเคราะห์สัญญาณในโดเมนเวลาเพื่อแยกส่วนประกอบความถี่ต่าง ๆ ทำให้สามารถระบุความถี่ที่ต้องการหรือตัดความถี่รบกวนออกได้
- 2.5.20 Block Varying Low Pass Filter: บล็อกใน Simulink ที่ออกแบบให้กรองความถี่สูงหรือความถี่ที่ไม่ต้องการออก โดยสามารถปรับค่าการกรองได้ตามลักษณะของสัญญาณ
- 2.5.21 Real Time Block: บล็อกใน Simulink ที่ใช้สำหรับปรับการทำงานของระบบให้ตรงกับเวลาจริง เพื่อให้ค่าที่วัดได้สะท้อนสถานการณ์จริงในหน่วยวินาที
- 2.5.22 Cut Off Frequency: ค่าความถี่ขอบเขตของตัวกรองที่กำหนดว่าความถี่ที่สูงกว่านี้จะถูกลดthonหรือตัดออก เช่น ใช้ใน Low Pass Filter เพื่อลดสัญญาณรบกวน
- 2.5.23 Butterworth Low Pass Filter: ตัวกรองสัญญาณแบบ Low Pass ที่ออกแบบให้ความถี่ที่ต่ำกว่าจุดตัด (Cut Off Frequency) ผ่านได้อย่างราบรื่น และลดความถี่สูงอย่างมีประสิทธิภาพ
- 2.5.24 MATLAB Function Block: บล็อกใน Simulink ที่ใช้เขียนโค้ด MATLAB แบบกำหนดเองเพื่อประมวลผลข้อมูลหรือคำนวนค่าต่าง ๆ ที่จำเป็นในระบบ
- 2.5.25 Fast Fourier Transform (FFT): กระบวนการแปลงข้อมูลสัญญาณในโดเมนเวลาให้เป็นโดเมนความถี่ เพื่อวิเคราะห์และแยกความถี่ที่ต้องการออกจากความถี่รบกวน

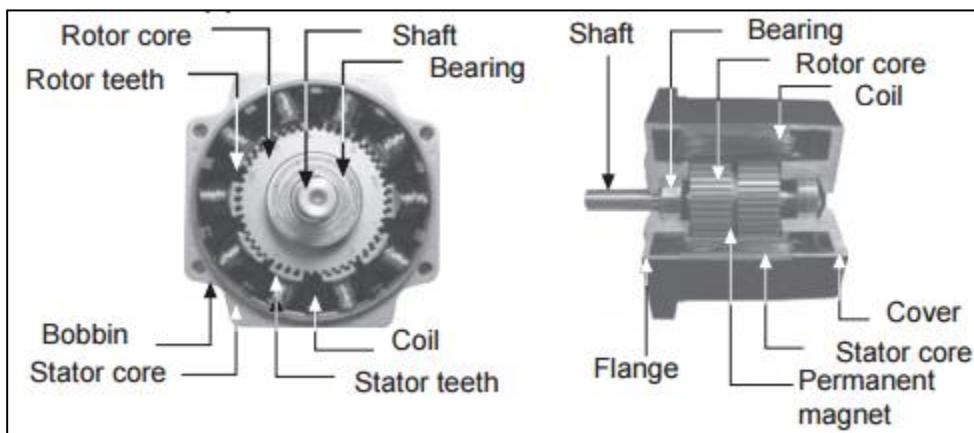
- 2.5.26 Step Loss Calculation: กระบวนการคำนวณการสูญเสียขั้นตอนของ Stepper Motor โดยใช้ความเร็ว เชิงมุมและมุมการหมุนที่วัดได้จาก Incremental Encoder เพื่อบวกค่าที่ควรจะเป็น
- 2.5.27 Rotary Switch ในโปรแกรม MATLAB Simulink: เครื่องมือใน Simulink สำหรับเลือกค่าหรือ พารามิเตอร์ เช่น ความละเอียดของ Micro Step โดยการหมุนสวิตช์เพื่อกำหนดค่าที่ต้องการ
- 2.5.28 Total Loss: จำนวนขั้นตอนที่สูญเสียไปทั้งหมดในการหมุนของ Stepper Motor
- 2.5.29 RadLoss: จำนวนมุมที่สูญเสียไปในหน่วยเรเดียนต่อการหมุนหนึ่งรอบ

2.6 เอกสารและงานวิจัยที่เกี่ยวข้อง

2.6.1 หลักการทำงานของ Stepper Motor

Stepper Motor เป็นมอเตอร์ไฟฟ้าที่ขับเคลื่อนด้วย Pulse ซึ่งเป็นมอเตอร์ชนิดหนึ่งที่สามารถหมุนในทิศทางที่แน่นอนได้ในทุก ๆ ขั้นตอน โดยโครงสร้างภายในนั้นจะประกอบไปด้วยขั้วแม่เหล็กบน Stator ทามาจากแผ่นเหล็กง่วงหวาน จะมีชี้นอกรมาประกอบกันเป็นชั้น ๆ โดยแต่ละชี้นที่ยื่นออกมาจะมีขดลวด (Coil) พันอยู่ Stepper Motor นั้นจะไม่สามารถขับเคลื่อนหรือทำงานเองได้ จำเป็นต้องมีวงจร อิเล็กทรอนิกส์ที่ใช้ในการสร้างสัญญาณหรือจ่ายพลังสืบไปให้กับ Stepping Motor Drive การสร้างสัญญาณนั้นจะเป็นต้องสร้างและเรียงลำดับของสัญญาณด้วยและอิกสิ่งที่สำคัญคือการคูณตำแหน่งของสายที่ทำการต่อเข้ากับ Stepper Motor มอเตอร์ชนิดนี้หมุนในหน่วยการหมุนที่เรียกว่า "Step" โดยแต่ละการหมุนจะมีมุมการหมุนคงที่ เช่น 1.8° , 0.9° ขึ้นอยู่กับชนิดของมอเตอร์

2.6.2 ส่วนประกอบของ Stepper Motor



รูปที่ 25 แสดงส่วนประกอบของ Stepper Motor

- 1) Rotor Core (แกนหมุนหลักของ Rotor) เป็นส่วนที่อยู่ตรงกลางของมอเตอร์ มีหน้าที่รับสนามแม่เหล็กที่เกิดขึ้นจากขดลวดและหมุนไปตามทิศทางของสนามแม่เหล็กนั้น
- 2) Rotor Teeth (ฟันของ Rotor) ส่วนของฟันบน Rotor ที่ช่วยให้มอเตอร์หมุนไปทีละ "Step" อย่างแม่นยำตามตำแหน่งของสนามแม่เหล็ก
- 3) Shaft (แกนหมุน) เป็นแกนโลหะที่เชื่อมต่อกับ Rotor Core ทำหน้าที่ส่งแรงหมุนจากมอเตอร์ไปยังภาระงานหรืออุปกรณ์อื่น
- 4) Bearing (ลูกปืน) รองรับการหมุนของแกนหมุน (Shaft) ลดแรงเสียดทานระหว่างการหมุน ช่วยให้การหมุนมีความราบรื่น

- 5) Coil (ขดลวด) เป็นขดลวดแม่เหล็กไฟฟ้าที่สร้างสนามแม่เหล็กขึ้นเมื่อมีกระแสไฟฟ้าไหลผ่าน ทำให้ Rotor ถูกดูดหรือดึงให้หมุนไปตามลำดับ
- 6) Stator Core (แกน Stator) โครงสร้างหลักของ Stator ที่ทำจากเหล็กแม่เหล็ก ชั้งรองรับขดลวดและช่วยให้สนามแม่เหล็กมีพิเศษทางที่เหมาะสม
- 7) Stator Teeth (ฟันของ Stator) ฟันของ Stator ทำหน้าที่จับตัวแทนของฟัน Rotor ทำให้การหมุนเป็นไปตามมุมการ Step อย่างถูกต้อง
- 8) Bobbin (แกนพลาสติกของขดลวด) เป็นวัสดุที่รองรับขดลวด (Coil) ช่วยให้ขดลวดจัดเรียงตัวได้ดีและป้องกันความเสียหายจากความร้อน
- 9) Permanent Magnet (แม่เหล็กถาวร) ใช้ในมอเตอร์แบบ Permanent Magnet Stepper Motor ทำหน้าที่สร้างสนามแม่เหล็กถาวรที่ส่งผลต่อการหมุนของ Rotor
- 10) Cover (ฝาครอบ) เป็นฝาครอบด้านนอกของมอเตอร์ที่ช่วยป้องกันชิ้นส่วนภายในจากสิ่งแผลกปลอม และให้ความแข็งแรง
- 11) Flange (แพลน) แพลนยึดเป็นจุดสำคัญที่มอเตอร์เข้ากับโครงสร้างภายนอก เช่น แท่นรองรับหรือเครื่องจักร

2.6.3 การ Loss Step

การทำงานผิดปกติของ Stepper Motor มักเกิดจากการสูญเสียขั้นตอน (Step Loss) ซึ่งทำให้มอเตอร์กระโดดข้ามขั้นตอนหรือหยุดทำงาน โดยสาเหตุอาจเกิดจากตัวมอเตอร์เองหรือระบบควบคุม อิเล็กทรอนิกส์ การระบุปัญหาอาจเป็นเรื่องยาก เนื่องจากการอ่านค่ากระแสไฟฟ้าของมอเตอร์นั้นไม่ค่อยมีประโยชน์ และมอเตอร์จำนวนมากไม่มีอุปกรณ์ป้อนกลับ (Feedback) เพื่อตรวจจับพฤติกรรมการทำงาน

1) การเลือก Stepper Motor ไม่เหมาะสมกับการใช้งาน

ทำให้เกิด Loss Step เกิดได้หากพารามิเตอร์ของมอเตอร์ไม่เหมาะสมกับความต้องการใช้งาน ฉะนั้นควรพิจารณาแก้ไขด้วย

- คำนวณแรงบิดและความเร็วที่ต้องการในสถานการณ์ที่เลวร้ายที่สุด (Worst-case Scenario)
- เว้นระยะความปลดภัยอย่างน้อย 30% บนกราฟ แรงบิด-ความเร็ว (Pull-out Torque) เพื่อให้มอเตอร์สามารถรับภาระได้แม้เกิดแรงกระชาก
- ตรวจสอบว่าไม่มีปัจจัยภายนอก เช่น โหลดติดขัด ที่อาจทำให้มอเตอร์หยุดทำงาน

2) ปัญหาในโหมด Start-Stop

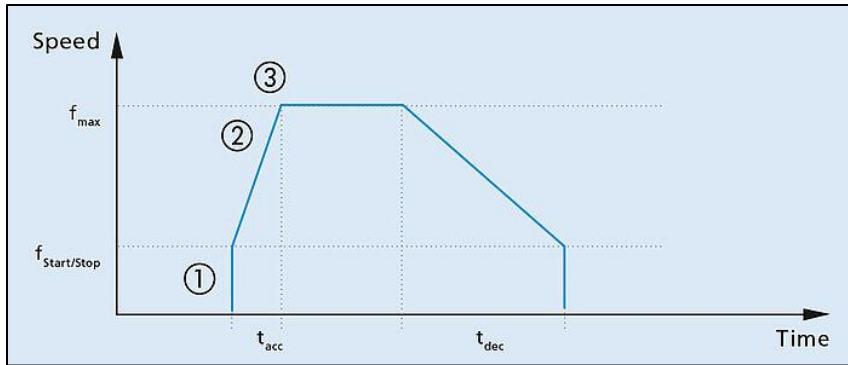
ในโหมด Start-Stop (เปิด-ปิดทันที) Stepper Motor ต้องสร้างแรงบิดให้มากพอเพื่อเข้าชนะ แรงเฉียบและแรงเสียดทานของโหลดภายในก้าวแรก หากไม่สามารถทำได้ มอเตอร์จะไม่เริ่มหมุน มีสาเหตุและวิธีแก้ไขดังนี้

- โหลดหนักเกินไป → ใช้มอเตอร์ที่มีกำลังสูงกว่า
- ความถี่สัญญาณสูงเกินไป → ลดความถี่ลง เพื่อให้มอเตอร์สามารถหมุนได้ทัน
- กระแสไฟฟ้าต่ำเกินไป → เพิ่มกระแสไฟเพียงพอ โดยเฉพาะในช่วงเริ่มต้นการหมุน

3) การเร่งความเร็วแบบ Trapezoidal

เมื่อมอเตอร์ต้องเร่งความเร็วขึ้นสู่ความเร็วสูงสุดแบบ Trapezoidal การเร่งที่รวดเร็วเกินไปอาจทำให้มอเตอร์หมุนไม่ทันและเกิด Step Loss มีสาเหตุและวิธีแก้ไขดังนี้

- การเร่งเร็วเกินไป → ลดความลาดชันของปรีไฟล์การเร่งความเร็ว
- เกิดการสั่นพ้อง (Resonance) → ทำ Micro Step, เพิ่มการ Damping เชิงกล
- แหล่งจ่ายไฟไม่เหมาะสม → ตรวจสอบแรงดันและกระแสไฟให้ตรงสเปกของมอเตอร์



รูปที่ 26 แสดงการเร่งแบบ Trapezoidal

4) การผิดพลาดจากการสลับเฟสไฟฟ้าภายนอก (External Commutation)

ข้อผิดพลาดที่เกิดขึ้นในการสลับเฟสไฟฟ้า (Commutation) ของ Stepper Motor ที่ถูกควบคุมโดยอุปกรณ์ภายนอก เช่น Driver หรือวงจรควบคุมไฟฟ้า ซึ่งการทำงานผิดพลาดนี้ส่งผลให้สัญญาณไฟฟ้าที่จ่ายเข้าสู่ดลัดมอเตอร์ไม่เป็นไปตามลำดับที่ถูกต้อง ทำให้มอเตอร์เกิด Step Loss ซึ่งมักมีลักษณะดังนี้

- เกิด Step Loss เป็นจำนวนที่ไม่เป็นตัวเลขของ 4 เช่น 8, 12 หรือ 16 ขั้น (ปกติ Stepper Motor จะทำงานเป็นรอบที่เกี่ยวข้องกับการสลับเฟสของสัญญาณไฟฟ้า)
- สาเหตุหลักมักมาจากการควบคุมที่ผิดพลาดในลำดับของสัญญาณไฟฟ้าที่ส่งไปยังชุดลวด (ผิดเฟส, คลาดเคลื่อนในการส่งสัญญาณ)
- มักไม่ได้เกิดจากตัวมอเตอร์ แต่เกิดจากอุปกรณ์ควบคุมหรือการตั้งค่าเฟสไฟฟ้าไม่ถูกต้อง

5) Step Loss จากภายนอก

การสูญเสียขั้นตอนที่ไม่ได้เกิดจากปัญหาภายในของ Stepper Motor หรือระบบควบคุม แต่เกิดจากเหตุการณ์หรือสภาพแวดล้อมภายนอกที่ส่งผลกระทบต่อการทำงานของมอเตอร์โดยตรง ซึ่งปัญหาเหล่านี้สามารถระบุได้ค่อนข้างง่ายจากลักษณะของการทำงานผิดปกติของมอเตอร์ ซึ่งมักมีสาเหตุมาจากการ

- โหลดเปลี่ยนแปลงกระแทก : การเพิ่มหรือลดภาระโหลดที่ไม่คาดคิด ทำให้แรงบิดที่มอเตอร์ต้องสร้างขึ้นไม่สอดคล้องกับความต้องการของระบบ
- การติดขัดของกลไกหรือระบบเครื่องกล : อุปกรณ์ที่เชื่อมต่อกับมอเตอร์ เช่น สายพาน, รยางค์, เพียง มีสิ่งกีดขวางหรือเกิดการเสียดสี ทำให้มอเตอร์หยุดการหมุน
- แรงกระแทกหรือการสั่นสะเทือนจากภายนอก : การสั่นสะเทือนที่เกิดขึ้นจากเครื่องจักรอื่นภายในระบบ หรือแรงกระแทกจากการทำงาน ส่งผลให้มอเตอร์สูญเสียการควบคุมและข้ามขั้นตอน
- สภาพแวดล้อมที่ไม่เหมาะสม : ความร้อนสูงเกินไป, ความชื้น, หรือการจ่ายไฟฟ้าที่ไม่เสถียร ทำให้มอเตอร์ทำงานผิดพลาด

2.6.4 การสั่งความเร็วด้วยความเร่งของ Stepper Motor

การสั่งความเร็วด้วยความเร่ง (Acceleration Control) สำหรับ Stepper Motor เป็นเทคนิคที่สำคัญเพื่อให้มอเตอร์ทำงานได้อย่างราบรื่นและป้องกันปัญหา Step Loss โดยเฉพาะในกรณีที่ต้องการเปลี่ยนแปลงความเร็วจากจุดหยุดนิ่งไปจนถึงความเร็วสูงสุด

หลักการของการสั่งความเร็วด้วยความเร่ง ไม่สามารถเปลี่ยนความเร็วจากศูนย์ไปยังความเร็วสูงสุดได้ในทันที เนื่องจากมอเตอร์มีข้อจำกัดจากแรงดึงดูดของโหลดและแรงเสียดทานที่ต้องเอาชนะ การเปลี่ยนความเร็วอย่างฉับพลันจะทำให้เกิดปัญหา Step Loss ซึ่งทำให้มอเตอร์สูญเสียความแม่นยำ

เพื่อแก้ไขปัญหานี้ การควบคุมความเร่ง จึงถูกนำมาใช้ โดยทำการเพิ่มความเร็วทีละน้อยจากความเร็วต่ำไปยังความเร็วสูงสุดในลักษณะ Ramp-up และลดความเร็วลงในลักษณะ Ramp-down เมื่อหยุดการทำงาน โดยรูปแบบการสั่งความเร็วด้วยความเร่งมีดังนี้

- 1) Linear Acceleration (การเร่งเชิงเส้น) การเพิ่มความเร็วอย่างคงที่ทีละขั้นตามระยะเวลาที่กำหนด 譬如ความเร็วเป็นเส้นตรง (Ramp-up) ไปยังจุดความเร็วสูงสุด และ Ramp-down เมื่อหยุดทำงาน
- 2) Trapezoidal Velocity Profile การเร่งความเร็วจากศูนย์ไปยังจุดสูงสุด จากนั้นรักษาความเร็วคงที่ และชะลอความเร็วลง ช่วยให้การทำงานราบรื่น และลดความเสี่ยงของ Step Loss
- 3) S-Curve Acceleration (การเร่งแบบโค้ง S) การเพิ่มความเร็วอย่างนุ่มนวลเป็นเส้นโค้ง (ไม่ใช่เส้นตรง) ช่วยลดแรงกระชากและแรงสั่นสะเทือน

ตารางที่ 1 แสดงการเปรียบเทียบข้อดีและข้อเสียของวิธีการสั่งความเร็วตัว�ความเร่งของ Stepper Motor

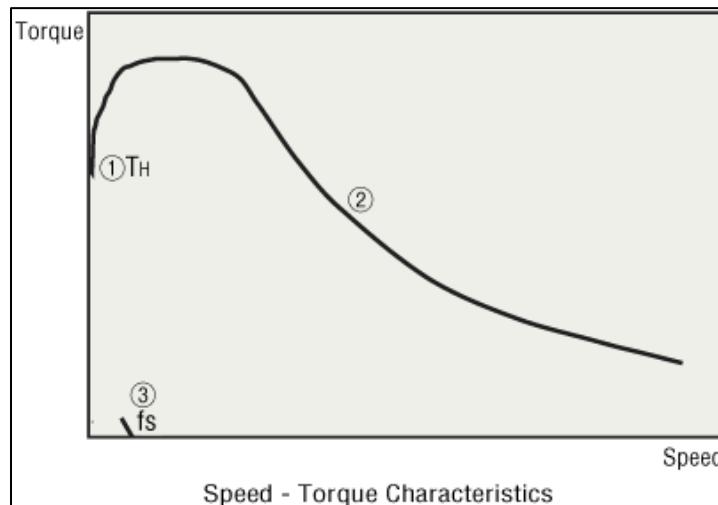
รูปแบบการเร่งความเร็ว	ข้อดี	ข้อเสีย
Linear Acceleration	<ul style="list-style-type: none"> - โครงสร้างโปรแกรมง่าย เข้าใจง่าย - การเร่งคงที่ ทำให้ควบคุมได้ง่าย 	<ul style="list-style-type: none"> - อาจเกิดแรงกระชากในช่วงเริ่มต้นและหยุดแรงสั่นสะเทือนยังคงมีอยู่
Trapezoidal Velocity Profile	<ul style="list-style-type: none"> - ราบรื่นขึ้นกว่า Linear - ลดการกระชากในช่วงรักษาความเร็วคงที่ - ควบคุมได้แม่นยำ 	<ul style="list-style-type: none"> - ช่วงเร่งและลดความเร็วอาจยังมีการสั่นสะเทือนเล็กน้อยในบางกรณี
S-Curve Acceleration	<ul style="list-style-type: none"> - ราบรื่นที่สุด ลดแรงกระชากและการสั่นสะเทือนได้ดี - ช่วยยืดอายุการใช้งานของมอเตอร์ - เหมาะสำหรับโหลดที่ต้องการความแม่นยำสูง- 	<ul style="list-style-type: none"> - ซับซ้อนในการออกแบบและควบคุม - ต้องการทรัพยากรการประมวลผลมาก

2.6.5 Stepper Motor Characteristics

คุณลักษณะของ Stepper Motor สามารถจำแนกออกอย่างกว้าง ๆ ได้ 2 ประเภท ดังนี้

1) Dynamic Characteristics

1.1) Speed - Torque Characteristics



รูปที่ 27 แสดงความสัมพันธ์ระหว่างแรงบิดและความเร็วของ Stepper Motor

ความสัมพันธ์ระหว่างความเร็วและแรงบิดนี้ถูกกำหนดโดยมอเตอร์และ Driver ซึ่งได้รับผลกระทบอย่างมากจากชนิดของ Driver ที่ใช้

1.1.1) Maximum Holding Torque (TH: Holding Torque)

: แรงบิดสูงสุด (แรงยืด) ที่ Stepper Motor มีเมื่อยื่นสถานะเปิดไฟ (กระแสไฟฟ้าตามค่าพิกัด) และหยุดนิ่ง

1.1.2) Pullout Torque

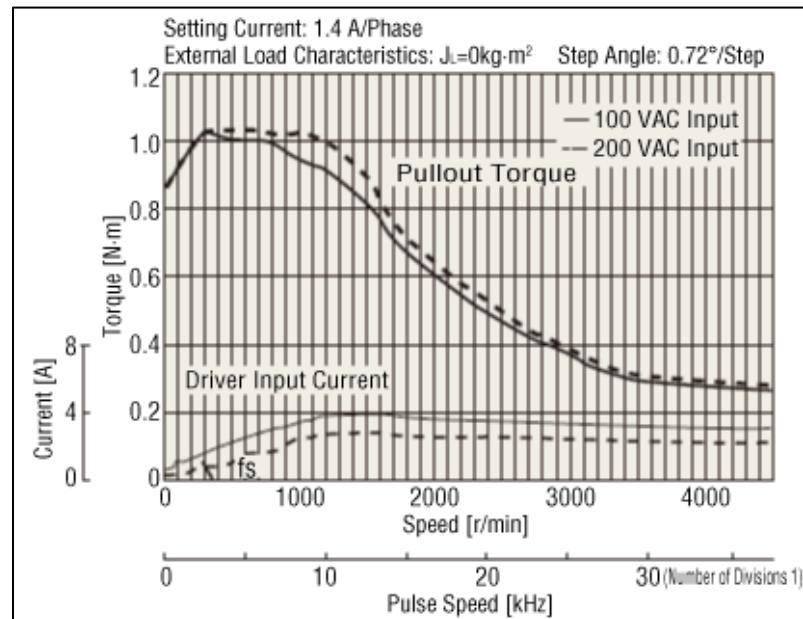
: แรงบิดสูงสุดในทันทีที่สามารถสร้างได้ในแต่ละความเร็วรอบ เมื่อเลือกมอเตอร์ควรตรวจสอบให้แน่ใจว่าแรงบิดที่ต้องการอยู่ภายในกราฟคุณลักษณะนี้

1.1.3) Maximum Starting Frequency

: ความถี่พัลส์สูงสุดที่มอเตอร์สามารถเริ่มและหยุดได้ทันที (โดยไม่ต้องใช้เวลาเร่ง/ลดความเร็ว) เมื่อไม่มีแรงเสียดทานและความเรื้อยโหลด (Load Inertia) บนมอเตอร์ซึ่งหากต้องการขับมอเตอร์ที่ความถี่พัลส์สูงกว่านี้ จะต้องเพิ่มเวลาเร่ง/ลดความเร็วโดยค่านี้จะลดลงเมื่อมี Load Inertia บnmอเตอร์

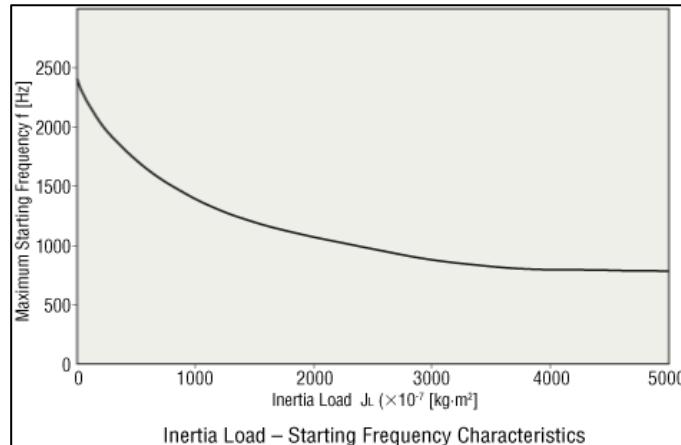
1.1.4) Maximum Response Frequency

: ความถี่พัลส์สูงสุดที่ Stepper Motor สามารถทำงานได้ด้วยการเร่ง/ลดความเร็วที่ลงน้อย เมื่อไม่มีแรงเสียดทานและ Load Inertia



รูปที่ 28 แสดงความสัมพันธ์ระหว่างกระแสและความเร็วของพัลส์ใน Stepper Motor

1.2) Load Inertia - Starting Frequency Characteristics



รูปที่ 29 แสดงความสัมพันธ์ระหว่างความถี่สูงสุดและความเมื่อยโหลดใน Stepper Motor

เนื่องจาก Rotor ของ Stepper Motor และ Load มีความเมื่อย ทำให้เกิดความล่าช้าหรือความเร่งล้าหน้าของแกนมอเตอร์ระหว่างการเริ่มต้นและหยุดในทันที ค่านี้จะเปลี่ยนแปลงไปตามความถี่พัลส์ แต่หากเกินค่าที่กำหนด มอเตอร์จะไม่สามารถติดตามความถี่พัลส์ได้และเกิด Step Loss ตามมา ซึ่งความถี่พัลส์ก่อนที่ระบบจะสูญเสียขั้นตอนเรียกว่า ความถี่เริ่มต้น (Starting Frequency)

ความสัมพันธ์ของความถี่เริ่มต้นสูงสุดและความเมื่อยโหลดสามารถประมาณได้ด้วยสมการต่อไปนี้

$$f = \frac{f_s}{1 + \sqrt{\frac{J_L}{J_o}}}$$

เมื่อ f_s คือความถี่เริ่มต้นสูงสุดของมอเตอร์เพียงอย่างเดียว มีหน่วยเป็น เฮิร์ท (Hz)

f คือความถี่เริ่มต้นสูงสุดเมื่อมี Load Inertia มีหน่วยเป็น เฮิร์ท (Hz)

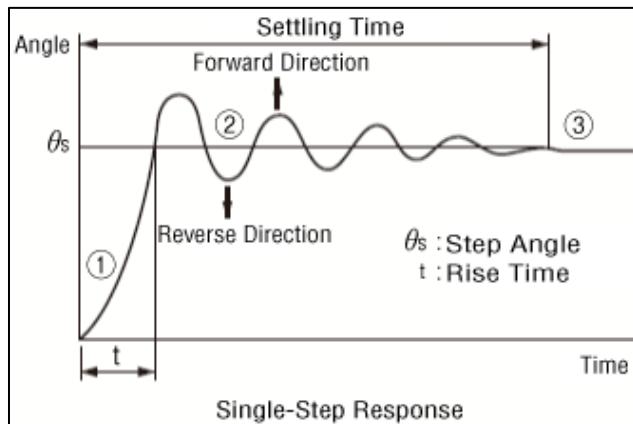
J_o คือความเมื่อยของ Rotor มีหน่วยเป็นนิวตัน (N) หรือ $\text{kg}\cdot\text{m}^2$

J_L คือความเมื่อยของโหลด มีหน่วยเป็นนิวตัน (N) หรือ $\text{kg}\cdot\text{m}^2$

1.3) Vibration Characteristics

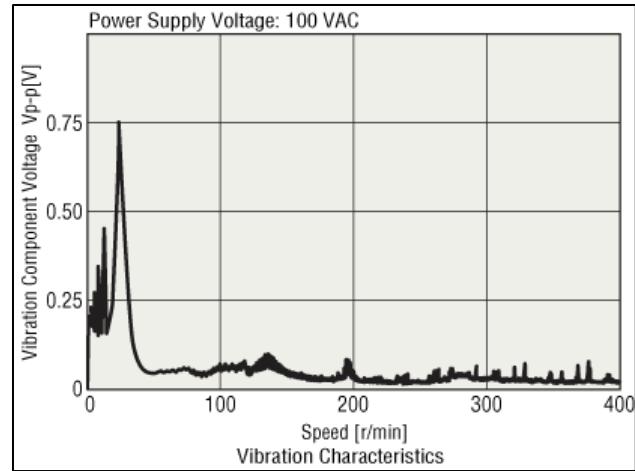
Stepper Motor หมุนในลักษณะการเคลื่อนที่เป็นขั้น (Step) ต่อเนื่อง การตอบสนองของแต่ละขั้นตอนซึ่งแสดงในรูปแบบการเคลื่อนที่เป็นขั้นตอน เรียกว่า Single-Step Response คือ

- 1.3.1) เมื่อส่งพัลส์เพียงครั้งเดียวไปยัง Stepper Motor ที่หยุดนิ่ง มอเตอร์จะเร่งไปยังมุมของขั้นตอนถัดไป
- 1.3.2) มอเตอร์ที่เร่งความเร็วจะผ่านมุมของขั้นตอนนั้น เกิดการเร่งล้าหน้า (Overshoot) และถูกดึงกลับในทิศทางตรงกันข้าม
- 1.3.3) หลังจากเกิดการสั่นแบบลดลง (Damped Oscillation) มอเตอร์จะหยุดที่ตำแหน่งของมุมขั้นตอนที่กำหนด



รูปที่ 30 แสดงการสั่นสะเทือนที่เกิดจาก Single-Step Response

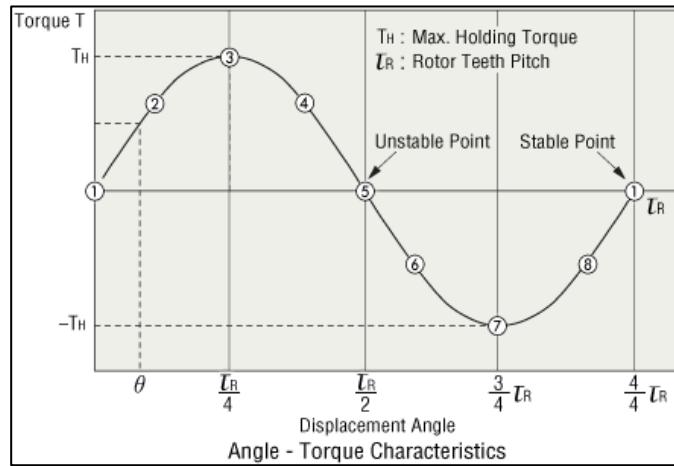
ซึ่ง Single-Step Response เป็นสาเหตุของการสั่นสะเทือนที่ความเร็วต่ำ และคุณลักษณะการสั่นสะเทือนนี้มักแสดงถึงระดับความรุนแรงของการสั่นสะเทือนในขณะที่ Stepper Motor กำลังหมุน ฉะนั้นหากการสั่นสะเทือนมีน้อยลง จะช่วยให้การหมุนราบรื่นมากขึ้น



รูปที่ 31 แสดง Vibration Characteristics

2) Static Characteristics

2.1) Angle - Torque Characteristics



รูปที่ 32 แสดงความสัมพันธ์ระหว่างแรงบิดและการทำมุมของ Stepper Motor

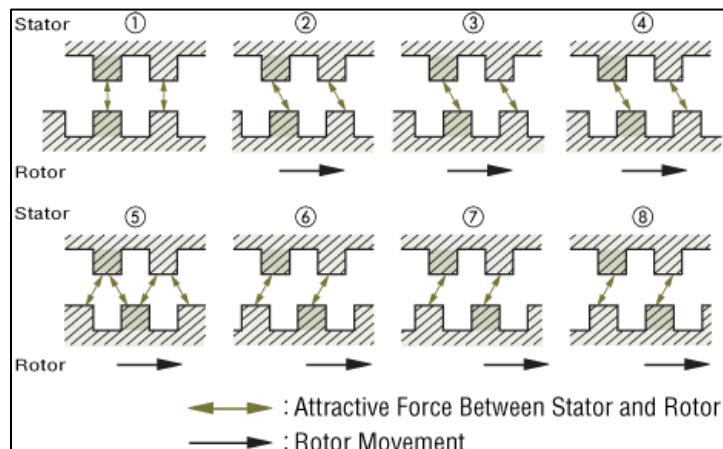
ความสัมพันธ์ระหว่างมุมการหมุนของ Rotor และแรงบิดที่เกิดขึ้นเมื่อมอเตอร์ถูกกระตุ้นด้วยกระแสไฟฟ้าตามค่าพิกัด และมีแรงบิดภายนอกกระทำต่อแกนหมุนของมอเตอร์ซึ่งสามารถอธิบายได้ดังนี้

2.1.1) จุดสมดุลและแรงบิด

เป็นจุดที่ฟันของ Stator และ Rotor อยู่ในตำแหน่งที่ตรงกันอย่างสมบูรณ์ (ฟันซ้อนกัน) ซึ่งเป็นตำแหน่งที่เสถียรสูงสุด และ Rotor จะหยุดอยู่ที่ตำแหน่งนี้เมื่อไม่แรงภายนอกกระทำ หากมีแรงบิดภายนอกกระทำให้ Rotor เบี่ยงออกจากจุดนี้ แรงบิด T(+) จะดึง Rotor กลับสู่จุดสมดุล

2.1.2) Unstable Point (จุดไม่เสถียร)

เป็นจุดที่ฟันของ Stator และ Rotor เยื่องกันครึ่งหนึ่งของระยะพิทช์ (1/2 pitch) ซึ่งเป็นจุดที่ไม่เสถียร หากมีแรงภายนอกเล็กน้อยกระทำต่อ Rotor จะถูกผลักไปยังจุดสมดุลข้างใดข้างหนึ่งทันที



รูปที่ 33 แสดง Rotor Movement ผ่านจุด Stable และ Unstable

2.2) Maximum Holding Torque (TH)

เมื่อแรงภายนอกที่กระทำต่อ Rotor เพิ่มขึ้น จะมีจุดหนึ่งที่แรงบิดถึงค่าสูงสุด ที่เรียกว่า Maximum Holding Torque (TH) ซึ่งหากแรงภายนอกเพิ่มขึ้นจนเกินจุดนี้ แรงบิดที่เกิดขึ้นจะทำให้ Rotor หมุนไปยังจุดสมดุลถัดไปและหยุดที่ตำแหน่งนั้น

2.3) ความแม่นยำเชิงมุม (Angular Accuracy)

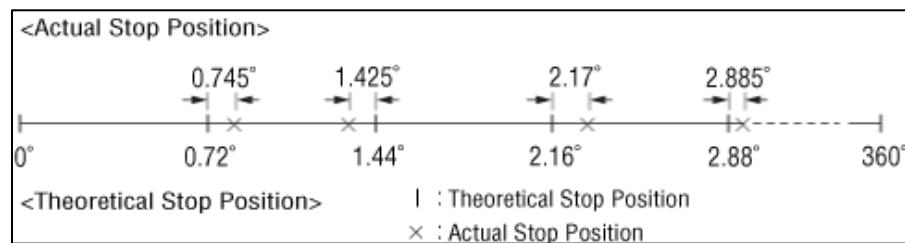
ความแม่นยำเชิงมุมของ Stepper Motor อยู่ที่ ± 3 Arcmin (0.05°) ภายใต้สภาวะไม่มี荷ลด ซึ่งความคลาดเคลื่อนนี้อาจเกิดจาก

2.3.1) ความแม่นยำเชิงกลของ Stator และ Rotor

2.3.2) ความไม่สม่ำเสมอของความต้านทานในชุดลวดของ Stator

2.3.3) Stop Position Accuracy (ความแม่นยำของตำแหน่งหยุด)

: ความแตกต่างระหว่างตำแหน่งหยุดตามทฤษฎีและตำแหน่งหยุดจริงของ Rotor ซึ่งวัดเป็นค่าสูงสุด (+) และค่าสูงสุด (-) ที่เกิดขึ้นเมื่อหมุนมาเต็มที่ลีบขั้นรอบ 360°

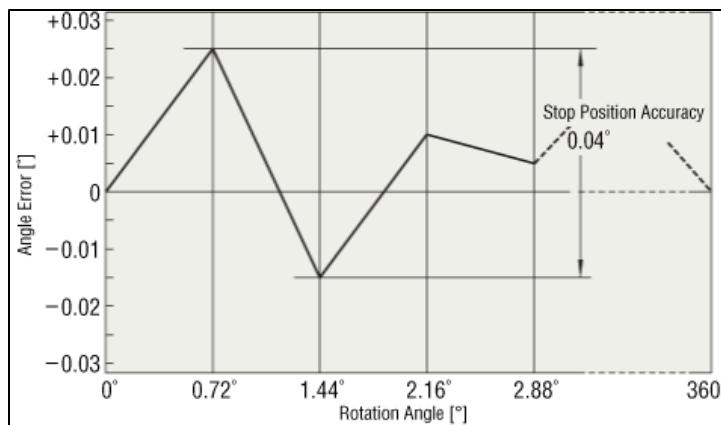


รูปที่ 34 แสดงความแตกต่างระหว่างตำแหน่งหยุดตามทฤษฎีและตำแหน่งหยุดจริงของ Rotor

2.4) อิทธิพลของ Frictional Load ต่อความแม่นยำเชิงมุม

ในการใช้งานจริง Frictional Load จะส่งผลให้เกิดการเบี่ยงเบนของมุน Rotor จากตำแหน่งที่คาดหวัง หากแรงเสียดทานมีค่าคงที่ การเบี่ยงเบนของมุนจะมีค่าคงที่ในกรณีที่มอเตอร์หมุนไปในทิศทางเดียว (Unidirectional Operation)

แต่ในกรณีหมุนไป-กลับ (Bidirectional Operation) การเบี่ยงเบนของมุนจะมีค่าเป็นสองเท่าของการเบี่ยงเบนปกติ ทำให้เมื่อต้องการความแม่นยำในการหยุดสูงสุด ควรกำหนดตำแหน่งหยุดจากทิศทางเดียวเท่านั้น เพื่อให้การเบี่ยงเบนเกิดขึ้นอย่างสม่ำเสมอ



รูปที่ 35 แสดงถึง Step Loss ต่อองศาที่เปลี่ยนแปลงไปตามจังหวะ

2.6.6 Stepper Motor Specification

1) Stability and Resolution in Control (ความเสถียรและความละเอียดในการควบคุม)

ความเสถียรในการควบคุมจำเป็นต้องใช้จำนวน Micro Step ที่สูง เพื่อช่วยให้ควบคุมตำแหน่งได้แม่นยำ แม้ในขั้นตอนที่เล็กมาก ซึ่งความละเอียดในการควบคุมจะช่วยลดข้อผิดพลาดและเพิ่มความแม่นยำในการทำงาน

2) Repeatability (ความสามารถในการทำซ้ำ)

Stepper Motor มีความแม่นยำอยู่ที่ประมาณ 3–5% ของแต่ละ Step โดยข้อผิดพลาดนี้ไม่จะสะสมจาก Step หนึ่งไปยัง Step ถัดไป ซึ่งความสามารถในการทำซ้ำขึ้นอยู่กับระบบอิเล็กทรอนิกส์ของ Driver, Load ที่มอเตอร์ต้องรับ และปัจจัยภายนอก เช่น การสั่นสะเทือน ความร้อน และความชื้น

3) Velocity (ความเร็ว)

ในการใช้งานมักต้องการความเร็วแตกต่างกันไป เช่น การเคลื่อนที่อย่างรวดเร็วสำหรับการกำหนดตำแหน่ง, การเคลื่อนที่ช้าในระดับ 100 $\mu\text{m/s}$ ถึงต่ำกว่า 100 nm/s สำหรับงานที่ต้องการความละเอียดสูง

4) Step Angle (มุมหมุนต่อขั้น)

หมายถึงการหมุนเชิงมุมของแกนมอเตอร์เมื่อได้รับพัลส์ไฟฟ้า 1 ครั้ง มักแสดงเป็นองศา เช่น 1.8° , 0.9° , 0.72° , หรือ 0.36° ตัวอย่าง มอเตอร์ที่มีมุมหมุนต่อขั้น 0.9° จะหมุนครบ 360° ด้วย 400 ขั้น

5) Step Size (ขนาดการหมุนต่อขั้น)

หมายถึงระยะทางเชิงเส้นที่แกนมอเตอร์เคลื่อนที่ต่อ 1 ขั้น กำหนดโดยมุมหมุนต่อขั้นและลักษณะทางกลของมอเตอร์ เช่น ระบบเกียร์หรือกลไกเชื่อมต่อ

6) Motor Flange Size (ขนาดหน้าแปลนมอเตอร์)

ใช้มาตรฐาน NEMA (National Electrical Manufacturers Association) เช่น NEMA 14 ถึง NEMA 56 ตัวอย่าง NEMA 17 มีขนาดหน้าแปลนประมาณ 43.2 มม. และมีรูติดตั้งแบบสี่เหลี่ยมจัตุรัส

7) Motor Connection Type (ประเภทการเชื่อมต่อ)

M Connector (เช่น M12) ตัวเชื่อมต่อทรงกลมที่นิยมในงานอุตสาหกรรม RJ45 Connector ตัวเชื่อมต่อที่ใช้ในระบบเครือข่าย Ethernet

8) Pullout Torque (แรงบิดดึงออก)

หมายถึงแรงบิดสูงสุดที่มอเตอร์สามารถสร้างได้ที่ความเร็วนี้ ๆ ควรเปรียบเทียบความต้องการแรงบิดกับกราฟ Speed-Torque Curve ของผลิตภัณฑ์

9) Maximum Holding Torque (แรงบิดหยุดสูงสุด)

หมายถึงแรงบิดสูงสุดที่มอเตอร์สามารถยึดตำแหน่งได้เมื่ออยู่ในสถานะหยุดนิ่งและได้รับพลังงาน

10) Encoder (ตัวเข้ารหัสตำแหน่ง)

ใช้ในการแปลงการเคลื่อนที่เชิงกลเป็นสัญญาณไฟฟ้าเพื่อให้ระบบควบคุมติดตามตำแหน่ง ความเร็ว และทิศทางแบบเรียลไทม์ Encoder Resolution (CPR/PPR): จำนวนพัลส์ที่สร้างได้ต่อการหมุน 1 รอบเต็มของแกนมอเตอร์

11) Current per Phase (กระแสต่อเฟส)

หมายถึงปริมาณกระแสไฟฟ้าที่ไหลในแต่ละชุดลวดของเฟسمอเตอร์ กระแสที่สูงขึ้นให้แรงบิดที่สูงขึ้น แต่เพิ่มการใช้พลังงานและการเกิดความร้อน

12) Number of Phases (จำนวนเฟส)

มอเตอร์มี 2 แบบหลัก 2-Phase Stepper Motor: ชุดลวด 2 ชุด (Phase A และ Phase B) สำหรับ 4-Phase Stepper Motor: ชุดลวด 4 ชุด (Phase A, B, C, D) ให้การเคลื่อนที่ราบรื่น และความละเอียดสูงกว่า.

13) Inductance (ความเหนี่ยวแน่น)

ค่าความเหนี่ยวแน่นต่อเฟسمีผลต่อความเร็วและแรงบิดที่มอเตอร์สามารถทำได้ ความเหนี่ยวแน่นที่สูงขึ้นช่วยให้การทำงานราบรื่นขึ้นในความเร็วสูง

14) Motor Length (ความยาวของมอเตอร์)

หมายถึงขนาดทางกายภาพของมอเตอร์ในแนวแกนหมุน (หน่วย: มม.).

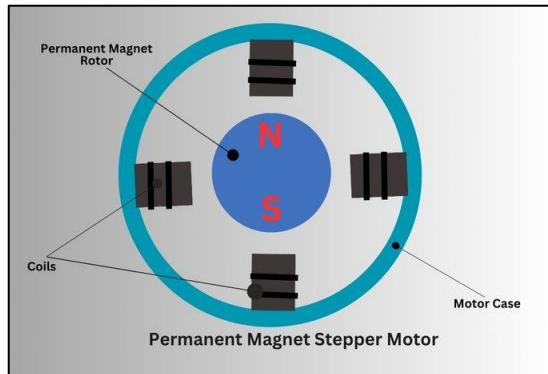
15) Resistance and Rated Voltage (ความต้านทานและแรงดันไฟฟ้าพิกัด)

ระบุค่าความต้านทานต่อเฟสและแรงดันที่จ่ายเมื่อมอเตอร์หยุดนิ่ง ควรเลือกแรงดันไฟฟ้าที่เหมาะสมกับแหล่งจ่ายไฟและ Driver

2.6.7 Stepper Motor Rotor Type

Rotor ใน Stepper Motor เป็นส่วนสำคัญที่หมุนและรับแรงจากสนามแม่เหล็กที่สร้างโดย Stator โดยประเภทของ Rotor ที่ใช้ใน Stepper Motor สามารถแบ่งออกเป็น 3 ประเภทหลัก โดยแต่ละประเภทมีคุณสมบัติและการใช้งานเฉพาะดังนี้

1) Permanent Magnet (PM) Stepper Motor



รูปที่ 36 แสดงส่วนประกอบของ Permanent Magnet (PM) Stepper Motor

Stepper Motor ชนิด Permanent Magnet (PM) เป็นมอเตอร์ไฟฟ้าที่แปลงพลังไฟฟ้าให้เป็นการเคลื่อนที่เชิงกลอย่างแม่นยำผ่านสนามแม่เหล็ก โดย Rotor ของมอเตอร์ชนิดนี้ฝังแม่เหล็กถาวรไว้ ซึ่งแม่เหล็กเหล่านี้จะสร้างสนามแม่เหล็กที่ทำปฏิกิริยากับสนามแม่เหล็กจาก Stator ทำให้มอเตอร์เคลื่อนที่ในมุมที่กำหนด

ข้อดีคือ PM Stepper Motor สามารถกำหนดตำแหน่งได้อย่างแม่นยำโดยไม่ต้องใช้อุปกรณ์ป้อนกลับ (Feedback Device) ซึ่งช่วยลดความซับซ้อนของการออกแบบและเพิ่มความน่าเชื่อถือโดยลดข้อผิดพลาดที่เกิดจากเซนเซอร์ และให้แรงบิดสูงที่ความเร็วต่ำ เนื่องจากใช้แม่เหล็กถาวรในการสร้างสนามแม่เหล็ก ทำให้มอเตอร์ชนิดนี้สามารถรักษาแรงบิดสูงได้แม้ในความเร็วต่ำทำให้เหมาะสมสำหรับการใช้งานที่ต้องการความแม่นยำและแรงบิดสูงที่ความเร็วต่ำ เช่น หุ่นยนต์, เครื่องจักรอัตโนมัติ, และอุปกรณ์ทางการแพทย์

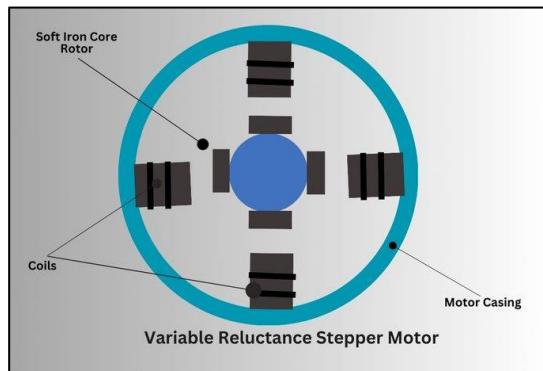
หลักการทำงาน PM Stepper Motor ทำงานบนหลักการแม่เหล็กไฟฟ้า โดยประกอบด้วย 2 ส่วนหลัก

Stator : ส่วนที่อยู่กับที่ ประกอบด้วยขดลวดแม่เหล็กไฟฟ้าจัดเรียงในรูปแบบเฉพาะ

Rotor : ส่วนที่หมุน ประกอบด้วยแม่เหล็กถาวรพร้อมพื้นผิวแบบมีช่อง

เมื่อต่อกระแสไฟฟ้ากับขดลวดใน Stator จะเกิดสนามแม่เหล็ก ซึ่งทำปฏิกิริยากับแม่เหล็กถาวรใน Rotor ทำให้เกิดการหมุน ซึ่งการควบคุมทิศทางการหมุนจะขึ้นอยู่กับลำดับการจ่ายกระแสไปยังขดลวดของ Stator

2) Variable Reluctance (VR) Stepper Motor

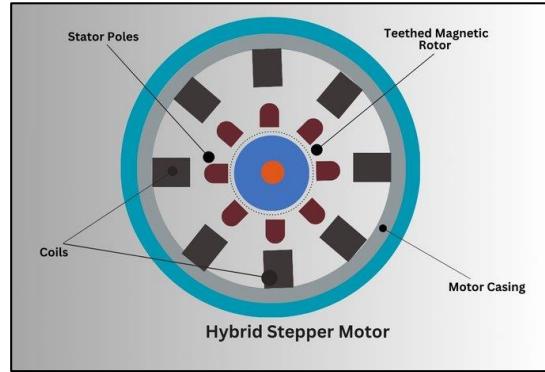


รูปที่ 37 แสดงส่วนประกอบ Variable Reluctance (VR) Stepper Motor

Stepper Motor ชนิด Variable Reluctance (VR) หรือ Reluctance Stepper Motor มี Rotor ทำจากเหล็กอ่อน (Soft Iron) และ Stator ที่มีฟันพร้อมขดลวดในแต่ละฟัน ข้อดีคือมีแรงบิดต่ำขณะไม่หมุน จากการออกแบบที่ช่วยลดแรงต้านเมื่อ Rotor เริ่มหมุน ทำให้มอเตอร์เคลื่อนที่ได้ราบรื่น ลดการสั่นสะเทือนและเสียงรบกวน รวมไปถึงให้แรงบิดสูงที่ความเร็วต่ำ การติดต่อระหว่างฟันของ Rotor และ Stator ช่วยสร้างแรงบิดสูงแม้ในความเร็วต่ำ VR Stepper Motor ใช้ในงานที่ต้องการกำลังสูง เช่น เครื่องจักรอุตสาหกรรม, หุ่นยนต์, และเครื่อง CNC ที่ต้องการการกำหนดตำแหน่งที่แม่นยำและการเคลื่อนที่ราบรื่น

หลักการทำงาน VR Stepper Motor ทำงานบนหลักการของความต้านทานแม่เหล็ก (Magnetic Reluctance) โดยขดลวดใน Stator สร้างสนามแม่เหล็กที่ดึงดูดฟันของ Rotor ให้หมุน โดยการควบคุมการหมุนสามารถทำได้ทั้งแบบ Unipolar Drives (จ่ายไฟทางเดียว) และ Bipolar Drives (จ่ายไฟสองทาง)

3) Hybrid Synchronous (HS) Stepper Motor



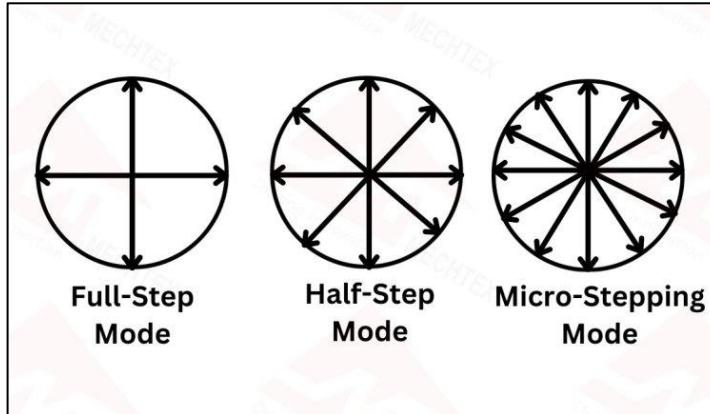
รูปที่ 38 แสดงส่วนประกอบ Hybrid Synchronous (HS) Stepper Motor

Stepper Motor ชนิด Hybrid Synchronous (HS) หรือ Hybrid Stepper Motor เป็นมอเตอร์ที่ผสมผสานคุณสมบัติของ PM และ VR Stepper Motor โดย Rotor มีทั้งแม่เหล็กถาวรและฟันที่ช่วยเพิ่มความแม่นยำ

ข้อดีคือสามารถให้แรงบิดสูงที่ความเร็วต่ำ เนื่องจากการโต้ตอบระหว่างแม่เหล็กถาวรในRotor และขดลวดใน Stator ช่วยให้เกิดแรงบิดสูงและแรงบิดหยุด (Holding Torque) ที่มั่นคง และให้ความแม่นยำในการกำหนดตำแหน่ง เพราะ Hybrid Stepper Motor สามารถทำงานได้ในระบบวงปิด (Closed-Loop System) โดยไม่ต้องใช้อุปกรณ์ป้อนกลับ และมีความละเอียดในการหมุนที่ 0.9° ถึง 1.8° หมายความว่าการนำไปใช้งาน Hybrid Stepper Motor ใช้ในงานที่ต้องการความแม่นยำสูง เช่น เครื่องพิมพ์ 3D, เครื่อง CNC, และอุปกรณ์ทางการแพทย์

หลักการทำงานคือ เมื่อขดลวดใน Stator ถูกกระแสไฟฟ้า จะเกิดสนามแม่เหล็กที่ดึงดูดฟันของ Rotor ฟันของ Rotor ช่วยเพิ่มความละเอียดในการกำหนดตำแหน่ง ทำให้ Hybrid Stepper Motor มีความสามารถในการเคลื่อนที่ที่นิ่มนวลและแม่นยำกว่า PM และ VR Stepper Motor

2.6.8 Stepper Drive Mode



รูปที่ 39 แสดงประเภทของโหมดการ Drive Stepper Motor

1) Full Step Mode

โหมด Full Step เป็นโหมดพื้นฐานที่ใช้ในการควบคุมการเคลื่อนที่ของมอเตอร์ ในโหมดนี้ มอเตอร์จะหมุนหนึ่งขั้นเต็ม (Full Step) สำหรับทุกพลัสด้าที่ส่งไปยังขดลวด ใน Stepper Motor แบบตั้งเดิม แต่ละเฟสจะประกอบด้วยขดลวด 2 ชุด เพื่อให้มอเตอร์เคลื่อนที่หนึ่งขั้นเต็มในมอเตอร์ ประเภทนี้ ขดลวดหนึ่งในแต่ละเฟสจะถูกกระตุ้นทีละลำดับอย่างต่อเนื่อง โดยการควบคุมลำดับการ กระตุ้นและระยะเวลาของแต่ละพลัสด้า มอเตอร์สามารถหมุนได้ทั้งในทิศตามเข็มนาฬิกา (Clockwise) หรือ ทวนเข็มนาฬิกา (Counterclockwise) ในโหมด Full Step Rotor จะหมุนทีมุมขั้น (Step Angle) ที่ กำหนดไว้สำหรับแต่ละพลัสด้าที่ส่งไปยังขดลวด โหมดนี้ช่วยให้มอเตอร์สร้างแรงบิดที่ดีกว่า แต่การเปลี่ยน จากขั้นหนึ่งไปยังอีกขั้นอาจทำให้เกิดเสียงรบกวนและการสั่นสะเทือนในบางแอปพลิเคชัน

2) Half Step Mode

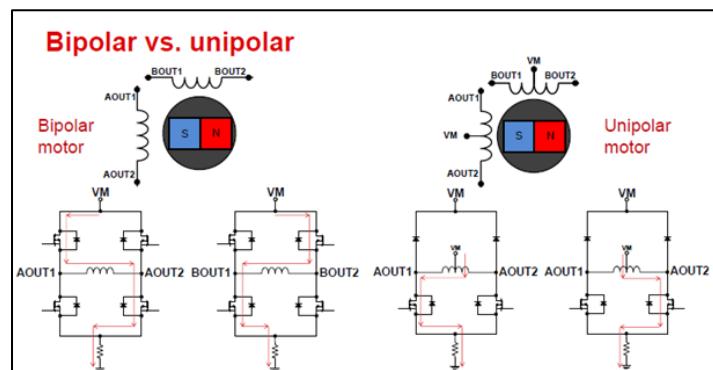
โหมด Half-Step เป็นโหมดที่อยู่ระหว่าง Full-Step Mode และ Micro Step Mode ในโหมดนี้ มอเตอร์จะหมุนด้วยขนาดของขั้นที่เล็กลงครึ่งหนึ่งเมื่อเทียบกับขั้นในโหมด Full-Step ใน Stepper Motor แบบสองเฟส (Two-Phase Stepper Motor) ชุดลวดจะถูกกระตุ้นในลำดับที่เริ่มจากการกระตุ้น ชุดลวดเพียงชุดเดียว (Single Coil) และต่อด้วยการกระตุ้นชุดลวดทั้งสองชุดพร้อมกัน (Dual Coils) แต่ ในระดับกระแสไฟฟ้าที่ต่ำกว่า ซึ่งผลลัพธ์คือมอเตอร์จะก้าวทีละขั้นที่เล็กกว่าขั้นในโหมด Full-Step โดย ในโหมด Half-Step มอเตอร์จะมีมุมก้าว (Step Angle) ลดลงครึ่งหนึ่งสำหรับทุกพลังที่ส่งเข้าไป ทำให้ จำนวนขั้นเพิ่มขึ้นเป็นสองเท่าเมื่อเทียบกับโหมด Full-Step แต่การทำงานจะราบรื่นขึ้นและมีความ ละเอียดสูงขึ้น โหมดนี้ช่วยปรับปรุงการเคลื่อนที่ของมอเตอร์ เนื่องจากขั้นที่เล็กลงช่วยให้ควบคุมตำแหน่ง ของมอเตอร์ได้อย่างแม่นยำขึ้น และลดเสียงรบกวนและการสั่นสะเทือนที่อาจเกิดขึ้น

3) Micro Step Mode

โหมด Micro Step เป็นโหมดที่ใช้เพื่อควบคุมการเคลื่อนที่ของมอเตอร์ให้แม่นยำมากที่สุด ใน โหมดนี้แต่ละขั้น (Step) จะถูกแบ่งย่อยออกเป็น Micro Step ที่เล็กกว่าอย่างมาก เมื่อเปรียบเทียบกับ โหมด Full-Step และ Half-Step โหมด Micro Step ช่วยทำให้ควบคุมการเคลื่อนที่ของมอเตอร์ได้ดีกว่า เพราะการกระตุ้นชุดลวดจะไม่ได้ทำแบบเต็มที่หรือสลับไปมาระหว่างชุดลวดเหมือนในโหมด Full-Step หรือ Half-Step แต่จะใช้วิธีปรับระดับกระแสไฟฟ้าในชุดลวดของมอเตอร์อย่างแม่นยำ การเปลี่ยนแปลง ระดับกระแสไฟนี้ช่วยให้ Rotor สามารถหยุดอยู่ที่ตำแหน่งระหว่างขั้นเต็มได้ง่าย โดยจำนวน Micro Steps ระหว่างขั้นเต็มของมอเตอร์อาจมีได้หลายสิบหรือมากกว่า ทำให้เหมาะสมสำหรับการใช้งานที่ ต้องการความแม่นยำสูงสุด การสั่นสะเทือนต่ำ และการทำงานที่เงียบ เช่น เครื่อง CNC, เครื่องพิมพ์ 3D และระบบอัตโนมัติอื่น ๆ ที่ต้องการควบคุมที่ละเอียดและราบรื่น

2.6.9 Stepper Drive Type

Stepper Motor แบบ Unipolar และ Bipolar สามารถทำงานได้โดยใช้แหล่งจ่ายไฟแบบ Unipolar (Single Rail) ทั้งสองแบบสามารถหมุนได้ทั้งสองทิศทางโดยใช้แหล่งจ่ายไฟตัวเดียวทั้งนี้ โดย Driver ของ Stepper Motor จะควบคุมตำแหน่งของ Rotor ด้วยการกระแสต้นขนาดของมอเตอร์ในลำดับที่กำหนด ตัวอย่างเช่น Microcontroller จะส่งพัลส์ไปยัง Driver มอเตอร์เพื่อบรรบว่า Rotor ควรเคลื่อนที่ไปยังตำแหน่งถัดไป เมื่อ Rotor ได้รับพัลส์ มันจะส่งพลังงานไปยังขดลวดในเฟสแรก และเมื่อได้รับพัลส์ถัดไป มันจะส่งพลังงานไปยังขดลวดในเฟสถัดไปเพื่อให้ Rotor หมุนต่อไป หากไม่มีครอคอกโนรอลเลอร์หยุดส่งพัลส์ Rotor จะยังคงหยุดนิ่งและอยู่ในแนวเดียวกับสนามแม่เหล็กของขดลวดเฟสที่ได้รับพลังงาน



รูปที่ 40 แสดงข้าของ Stepper Motor สามารถจัดเรียงและเดินสายได้ในรูปแบบ Bipolar และ Unipolar โดยแต่ละแบบต้องใช้ MOSFET ที่แตกต่างกัน

สำหรับ Stepper Motor แบบ H-Bridge และแม่เหล็กถาวร มีทักษะการตั้งค่าขดลวดแบบ Bipolar และ Unipolar มอเตอร์แบบ Bipolar ใช้งาน H-Bridge ซึ่งช่วยให้กระแสไฟฟ้าไหลในทิศทางเดียวกัน ขดลวด ทิศทางของกระแสไฟฟ้าจะเป็นตัวกำหนดขั้วของสนามแม่เหล็กที่สร้างขึ้นโดยขดลวด ในขณะที่ มอเตอร์แบบ Bipolar มีขดลวดเพียงชุดเดียวต่อเฟส ทำให้วางจารขับเคลื่อนขับซ้อนขึ้น เนื่องจากต้องกลับทิศทางกระแสไฟเพื่อเปลี่ยนขั้วของสนามแม่เหล็ก มอเตอร์แบบ Unipolar ต้องมี Transistor เพียง 4 ตัวในรูปแบบ Half-Bridge เพื่อควบคุม ขณะที่มอเตอร์แบบ Bipolar ต้องมี Transistor ถึง 8 ตัวที่จัดเรียงเป็นสอง H-Bridge เพื่อควบคุม

มอเตอร์แบบ Unipolar ต้องการเพียง FET แบบด้านเดียวหรือด้านสูง เพื่อกระตุ้นกระแสไฟฟ้าในทิศทางเดียวผ่านชุดลวด โดยชุดลวดในแต่ละเฟสจะมีจุดแท็บกลางซึ่งสามารถเชื่อมต่อกับกราว์ดหรือแหล่งจ่ายไฟได้ แทนที่จะเปลี่ยนทิศทางกระแสเพื่อเปลี่ยนขั้วของสนามแม่เหล็ก ชุดลวดสำหรับเฟสหนึ่งจะถูกพันในทิศทางตรงข้ามกันเพื่อสร้างสนามแม่เหล็กที่เปลี่ยนขั้วได้ วิธีนี้ช่วยให้สามารถเปลี่ยนทิศทางของมอเตอร์ได้โดยไม่ต้องกลับทิศทางของกระแสไฟฟ้า ทำให้สามารถทำงานได้แม้จะใช้แหล่งจ่ายไฟแบบขั้วเดียว โดยทั่วไป การเดินสายของมอเตอร์ Unipolar มีสามสายต่อเฟส ทำให้มอเตอร์สองเฟสมาตรฐานมีสายทั้งหมด 6 สาย และมอเตอร์ Bipolar สามารถถูกขับเหมือนมอเตอร์ Unipolar ได้หากจุดแท็บกลางไม่ได้เชื่อมต่อและชุดลวดถูกเชื่อมต่อกับ Driver มอเตอร์ Bipolar

ความแตกต่างที่สำคัญระหว่าง Stepper Motor แบบ Unipolar และ Bipolar คือการมีหรือไม่มีสายแท็บกลางซึ่งแบ่งชุดลวดที่พันเป็นสองส่วน การแบ่งนี้สามารถทำได้ด้วยสายเชื่อมต่อเส้นเดียวหรือสองเส้นสำหรับแต่ละปลายชุดลวด การตัดแท็บกลางออกจะเปลี่ยนการตั้งค่า Unipolar เป็นการเชื่อมต่อแบบอนุกรม Bipolar

ในทางปฏิบัติ สีของสายไฟของมอเตอร์มักถูกกำหนดเป็นมาตรฐานในอุตสาหกรรมและมีความสม่ำเสมอในกลุ่มผลิตภัณฑ์ของผู้ผลิตรายเดียวกัน ด้วยเหตุนี้ แผนผังการเดินสายส่วนใหญ่มักให้สีแทนการระบุหมายเลข แต่การเลือกโครงสร้างการตั้งค่าแบบ Unipolar หรือ Bipolar ไม่ได้ส่งผลกระทบต่อการเดินสายเพียงอย่างเดียว แต่ยังส่งผลกระทบต่อคุณลักษณะทางไฟฟ้าของชุดลวดมอเตอร์ เช่น แรงดันไฟฟ้า ความต้านทาน ความเหนี่ยวแน่น ความเร็ว อัตราเร่ง และแรงบิด ซึ่งต้องพิจารณาอย่างตามการใช้งานที่ต้องการ

ตารางที่ 2 แสดงข้อแตกต่างระหว่าง Stepper Motor แบบ Unipolar และ Bipolar

คุณสมบัติ	Unipolar	Bipolar
Voltage Drive	ใช้งานได้ด้วย Transistor 4 ตัว	ใช้งานได้ด้วย Transistor 8 ตัว
Current Drive	ซับซ้อน	ง่ายขึ้นเมื่อมีชิปและสามารถควบคุมในโหมด Micro Step ได้
Amount of Copper Powered	ใช้ชุดลวดเพียงครึ่งเดียว	ใช้ชุดลวดเต็ม
Torque for Same Joule Losses	แรงบิดน้อยกว่าเมื่อเทียบกับการสูญเสียพลังงานเท่ากัน	แรงบิดมากกว่า 40%
Voltage Drive Speed	แรงบิดอาจต่ำที่ความเร็วต่ำและสูงขึ้นที่ความเร็วสูง	แรงบิดสูงขึ้นที่ความเร็วต่ำ จำกัดด้วยค่าแรงบิดสูงสุดที่กำหนดได้
Current Drive Speed	ซับซ้อนและไม่ค่อยถูกใช้งาน	ใช้ประโยชน์จากการควบคุมกระแสได้เต็มที่
Coil Connection	ไม่มีตัวเลือก	เข้มต่อแบบขานหรืออนุกรม
Wires	มี 6 หรือ 8 สาย	มี 4 หรือ 8 สาย
Variability	สามารถปรับให้เป็นโหมดไบโพลาร์ได้หากมีสาย 8 สาย	ใช้งานได้ในหลายรูปแบบ

2.6.10 Signal Conditioning และ Signal Processing

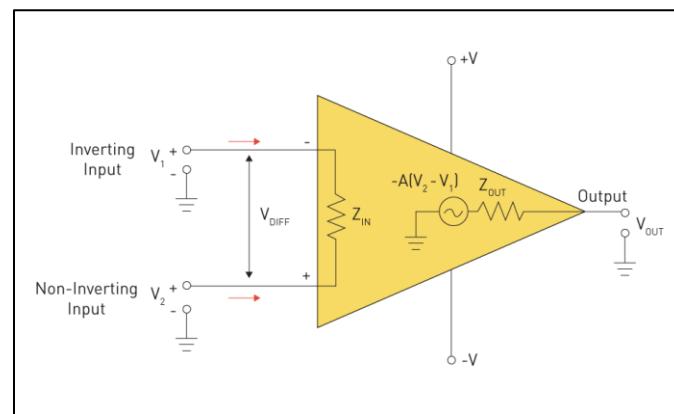
1) การขยายสัญญาณ (Amplification)

ในเซนเซอร์หลายประเภท โดยเฉพาะเซนเซอร์แบบ Passive มักจะส่งสัญญาณที่มีความเข้มต่ำ เช่น สัญญาณที่มีค่าในระดับ picoamperes (μA) ถึง microvolts (μV) ซึ่งไม่เพียงพอสำหรับการประมวลผลด้วยระบบอิเล็กทรอนิกส์มาตรฐาน เช่น Modulator ความถี่ (Frequency Modulators) หรือ วงจรแปลงสัญญาณแบบ Analog เป็น Digital (ADC) ด้วยเหตุนี้ การขยายสัญญาณของ Output เซนเซอร์ จึงมีความจำเป็น เพื่อให้สัญญาณมีความแรงเพียงพอสำหรับการอ่าน การประมวลผล หรือการแสดงผลโดย อุปกรณ์หรือระบบอื่น ๆ

1.1) สามารถแยกประเภทของวงจรขยายสัญญาณได้ 2 ประเภทคือ

- Operational Amplifiers (Op-Amps)

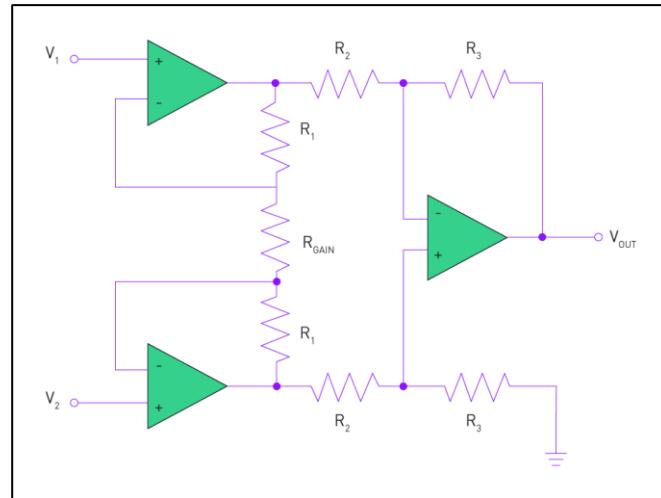
Op-Amps เป็นอุปกรณ์ที่มีความยืดหยุ่นและสามารถใช้งานได้หลากหลาย เช่น การเป็นตัวติดตามแรงดันไฟฟ้า (Voltage Follower) หรือวงจรเชิงช้อนที่มีการรวมค่า (Integration) หรือ การเปรียบเทียบค่า (Differentiation) Op-Amps มี Impedance อินพุตที่สูงและ Impedance เอาต์พุตต่ำ ทำให้เป็นส่วนประกอบที่สำคัญในอุปกรณ์อิเล็กทรอนิกส์สมัยใหม่และมีบทบาทสำคัญในกระบวนการปรับสัญญาณ (Signal Conditioning)



รูปที่ 41 แสดงวงจร Amplifiers แบบ Inverting และ Non-Inverting

- Instrumentation Amplifiers

Instrumentation Amplifiers ถูกออกแบบมาสำหรับการขยายสัญญาณที่แตกต่างกันในกรณีที่มีแรงดันแบบ Common-Mode สูง ตัวขยายประเภทนี้เหมาะสำหรับการประมวลผลสัญญาณจากเซนเซอร์ในสภาพแวดล้อมที่มีสัญญาณรบกวน (Noise)



รูปที่ 42 แสดงวงจร Instrumentation Amplifier

1.2) การพิจารณาอัตราการขยายสัญญาณและผลกระทบจากสัญญาณรบกวน

- อัตราการขยายสัญญาณ (Gain)

อัตราการขยายกำหนดระดับการขยายสัญญาณ หากอัตราการขยายสูงเกินไปจนแรงดันເຄົ້າພຸດມາກກວ່າแรงดันແຫລ່ງຈ່າຍໄຟ ຈະທຳໃໝ່ຈະຮຍາຍເຂົ້າສູ່ສ່ານະອົມຕ້ວ (Saturation) ດັ່ງນັ້ນຈີງຄວບຮັບອັດຕາການຮຍາຍໃຫ້ເໜາະສົມກັບແຮງດັນແຫລ່ງຈ່າຍໄຟແລະຊ່ວງອືນພົດທີ່ຄາດກາຮັນໄວ້

- ผลกระทบจากสัญญาณรบกวน (Noise)

ຈະຮຍາຍສัญญาณທຸກໆນິດເພີ່ມສัญญาณຮບກວນໃນຮະດັບໜຶ່ງ ການຮຍາຍສัญญาณທີ່ສູງນາກຈະເພີ່ມຮະດັບສัญญาณຮບກວນຈຸນຮບກວນສัญญาณທີ່ສົນໃຈ ການອົກແບບວຽກຮຍາຍສัญญาณຈຶ່ງຕ້ອງຄຳນິ້ງເລື່ອຕາສ່ວນສัญญาณຕ່ອງສัญญาณຮບກວນ (SNR) ແລະຄ່າຄວາມເພື່ອນຂອງສัญญาณ (Noise Figure)

2) การกรองสัญญาณ (Filtering)

การกรองสัญญาณมีบทบาทสำคัญในกระบวนการปรับแต่งสัญญาณ (Signal Conditioning) หน้าที่หลักของมันคือการลดTHONหรือกำจัดส่วนประกอบที่ไม่ต้องการ (ส่วนใหญ่คือสัญญาณรบกวน) ขณะเดียวกันก็อนุญาตให้ส่วนประกอบที่ต้องการสามารถผ่านไปได้ การออกแบบตัวกรองที่ดีสามารถปรับปรุงความแม่นยำและความน่าเชื่อถือของเอกสารพุตจากเซนเซอร์ได้อย่างมาก

2.1) ความสำคัญของการลดสัญญาณรบกวน

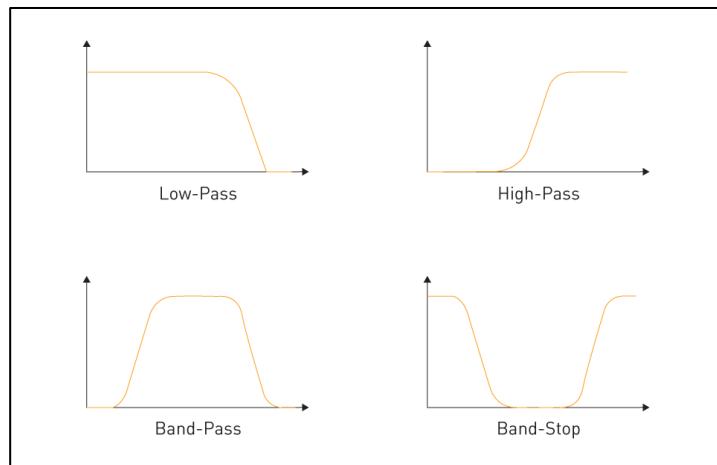
- เพิ่มคุณภาพของสัญญาณ

การปรับปรุงอัตราส่วน SNR ช่วยให้ข้อมูลที่ได้มีความแม่นยำและเป็นตัวแทนของปรากฏการณ์ที่กำลังตรวจวัดได้ดีขึ้น

- ป้องกันการแปลงผิดพลาดในระบบดิจิทัล (Aliasing)

การแปลงผิดพลาดเกิดขึ้นเมื่อความถี่สูงปรากฏเป็นความถี่ต่ำในระบบดิจิทัล การกรองช่วยให้เฉพาะความถี่ที่เหมาะสมผ่านไปยังขั้นตอนการแปลงสัญญาณ

2.2) การแบ่งประเภทของตัวกรอง



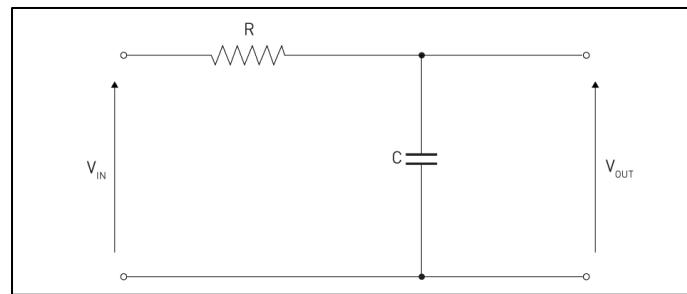
รูปที่ 43 แสดงประเภทของการ Filter สัญญาณและข้อมูล

- (1) Lowpass Filter: ลดทอนความถี่ที่สูงกว่าค่าที่กำหนดและอนุญาตให้ความถี่ที่ต่ำกว่าผ่าน
หมายเหตุ: สำหรับการกำจัดสัญญาณรบกวนความถี่สูง
- (2) High-Pass Filter: ลดทอนสัญญาณที่มีความถี่ต่ำกว่าค่าที่กำหนดและอนุญาตให้สัญญาณ
ความถี่สูงผ่าน หมายเหตุ: สำหรับการกำจัดส่วนประกอบที่เปลี่ยนแปลงช้า
- (3) Band-Pass Filter: อนุญาตเฉพาะความถี่ในช่วงที่กำหนดให้ผ่าน หมายเหตุ: สำหรับการแยก
ความถี่ที่สนใจ
- (4) Notch Filter: ลดทอนสัญญาณในช่วงความถี่แคบ ๆ เช่น สัญญาณรบกวนจากสายไฟฟ้าที่
ความถี่ 50/60 Hz

2.3) การแบ่งการกรองแบบ Analog และ Digital

- Analog Filtering

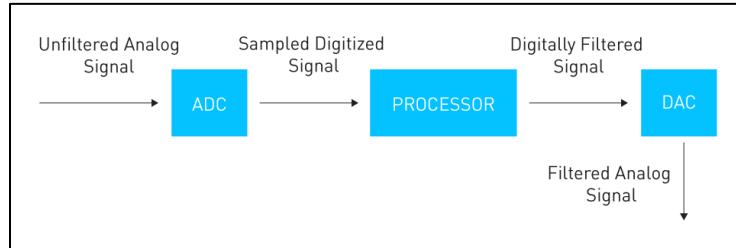
ใช้อุปกรณ์เชิงรุก เช่น Op-Amps หรืออุปกรณ์ Passive เช่น ตัวต้านทาน ตัวเก็บประจุ และตัวเหนี่ยวนำ ตัวกรอง Analog ตอบสนองได้รวดเร็วแต่มีความไวต่อการเปลี่ยนแปลงของอุปกรณ์



รูปที่ 44 แสดงการทำ Analog Lowpass Filter

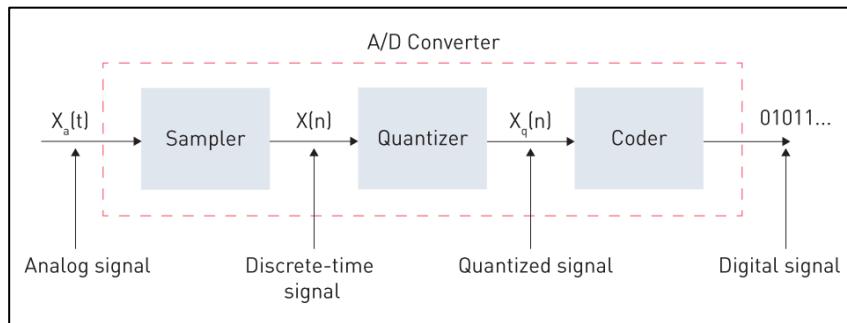
- Digital Filtering:

ใช้กับสัญญาณที่แปลงเป็น Digital แล้ว อัลกอริทึมช่วยให้ตัวกรองดิจิทัลสามารถสร้างรูปร่างการกรองที่แม่นยำ แต่มีความล่าช้าในกระบวนการประมวลผล



รูปที่ 45 แสดงการทำ Digital Filtering

3) Analog-to-Digital Conversion (ADC)



รูปที่ 46 แสดงหลักการอย่างง่ายของ Analog-to-Digital Converter

ในอุปกรณ์อิเล็กทรอนิกส์สมัยใหม่ การแปลงสัญญาณจากโดเมน Analog ไปสู่โดเมน Digital เป็นกระบวนการที่สำคัญ โดยเฉพาะอย่างยิ่งในการทำงานที่เกี่ยวกับเซนเซอร์ กระบวนการนี้ดำเนินการโดยเครื่องมือที่เรียกว่า Analog-to-Digital Converter (ADC) ซึ่งมีหน้าที่แปลงแรงดัน Analog ให้กลายเป็นค่า Digital เพื่อให้ง่ายต่อการประมวลผลหรือจัดเก็บข้อมูลในรูปแบบ Digital ในภายหลัง

3.1) หลักการของการconvonไทช์ (Quantization) และการสุ่มตัวอย่าง (Sampling)

- Sampling (การสุ่มตัวอย่าง)

สัญญาณ Analog ที่ต่อเนื่องตามเวลา จะถูกสุ่มตัวอย่างในช่วงเวลาที่กำหนด โดยอ้างอิงตามทฤษฎี Nyquist-Shannon Sampling ซึ่งระบุว่าความถี่ของการสุ่มตัวอย่าง (f_s) จะต้องมีค่ามากกว่าหรือเท่ากับสองเท่าของความถี่สูงสุดในสัญญาณ เพื่อป้องกันการเกิด Aliasing (การซ้อนทับของความถี่)

- Quantization (การconvonไทช์)

แต่ละตัวอย่างจะถูกกำหนดค่าเป็นค่าดิจิทัลที่จำกัดจำนวนระดับ Digital (ขึ้นอยู่กับความละเอียดของ ADC) กระบวนการนี้จะสร้างข้อผิดพลาดที่เรียกว่า Quantization Error ซึ่งเกิดจากความแตกต่างระหว่างค่าที่แท้จริงของสัญญาณ Analog กับค่า Digital ที่ใกล้เคียงที่สุด

- Encoding (การเข้ารหัส)

หลังจากสุ่มตัวอย่างและconvonไทช์แล้ว สัญญาณจะถูกเข้ารหัสเป็นรหัส Binary โดยจำนวนบิตที่ใช้ในกระบวนการเข้ารหัสจะกำหนดความละเอียด เช่น การใช้ 8 บิต สามารถแสดงระดับการconvonไทช์ได้ 256 ระดับ (2^8) โดยแต่ละตัวอย่างจะมีรหัสใบหน้าที่กำหนดไว้

3.2) การแบ่งประเภทของ ADC

(1) Successive Approximation ADC

ADC ประเภทนี้ใช้ Successive Approximation Register (SAR) และ Comparator ในการหาค่า Input สัญญาณแบบขั้นตอน ซึ่งมีความสมดุลระหว่าง ประสิทธิภาพและความเร็ว

(2) Delta-Sigma ADC ($\Delta\Sigma$ ADC)

เหมาะกับงานที่ต้องการความละเอียดสูง เช่น การประมวลผลเสียง โดยอาศัย เทคนิค Nose Shaping เพื่อลดสัญญาณรบกวนในช่วงความถี่ที่ต้องการ

(3) Flash ADC (Parallel ADC)

เป็น ADC ที่เร็วที่สุด โดยสามารถแปลงสัญญาณ Analog เป็น Digital ใน ขั้นตอนเดียว โดยใช้ Comparator หลายตัวเพื่อตรวจหาค่า Digital แต่ต้องใช้พลังงาน และความซับซ้อนสูง

3.3) ปัจจัยที่มีผลต่อการเลือก ADC

- Resolution (ความละเอียด) : กำหนดจำนวนระดับดิจิทัลที่ ADC สามารถสร้างได้ ยิ่ง ความละเอียดสูง ความแม่นยำในการแปลงยิ่งดีขึ้น แต่จะเพิ่มเวลาในการแปลงและการใช้ พลังงาน
- Speed (Sampling Rate): ความถี่การสุ่มตัวอย่างมีความสำคัญต่อแอปพลิเคชันต่าง ๆ เช่น การประมวลผลเสียงอาจต้องใช้ความถี่ระดับกิโลเฮิรตซ์ แต่การสื่อสารวิทยุอาจต้องใช้ ความถี่ระดับเมกะเฮิรตซ์หรือสูงกว่า
- Power Consumption: สำหรับอุปกรณ์ที่ใช้แบตเตอรี่หรืออุปกรณ์พกพา ความประหยัด พลังงานเป็นสิ่งสำคัญ ADC บางรุ่นถูกออกแบบมาให้ทำงานในโหมดพลังงานต่ำ
- Accuracy and Linearity: ความแม่นยำและความเป็นเส้นตรงสำคัญต่อการประยุกต์ที่ ต้องการความละเอียด เช่น การวัดค่าทางวิทยาศาสตร์

4) Linearization

ในเซนเซอร์บางประเภท Output ที่ไม่เป็นเชิงเส้น (Non-Linear Output) เป็นปัญหาที่ต้องจัดการ การ Linearization เป็นกระบวนการแก้ให้ Output เป็นเชิงเส้น ทำให้การประมวลผลและการวิเคราะห์ข้อมูลทำได้ง่ายขึ้น

โดยสามารถระบุของความไม่เป็นเชิงเส้นได้อาจมาจากการที่ Output ของเซนเซอร์บางประเภท เช่น Thermistor หรือเซ็นเซอร์สนามแม่เหล็ก ที่ไม่เป็นเชิงเส้นในช่วงการวัดทั้งหมด การทำ Linearization จึงจำเป็นสำหรับการวัดที่แม่นยำและเพื่อใช้งานเซนเซอร์ได้อย่างเต็มศักยภาพ

4.1) เทคนิคการทำ Linearization

(1) วงจรอิเล็กทรอนิกส์ (Electronic Circuits)

- ใช้ Operational Amplifiers เพื่อสร้าง Linearized Output
- ใช้เครือข่ายการทดเชย (Compensating Networks) เช่น ตัวด้านท่านและตัวเก็บประจุ

(2) ซอฟต์แวร์ (Software Methods)

- การใช้ตารางเพื่อจับคู่ Output ที่ไม่เป็นเส้นตรงกับค่าที่เป็นเส้นตรง
- Polynomial Correction หรือการใช้สมการทางคณิตศาสตร์เพื่อแก้ไข Output แบบเรียลไทม์
- Curve Fitting Algorithms: ใช้อัลกอริทึมเพื่อสร้างกราฟที่เหมาะสมที่สุดสำหรับเซนเซอร์

2.7 วิธีดำเนินการทดลอง

2.7.1 การทดลองที่ 1

ตั้งค่าระบบควบคุม Stepper Motor โดยใช้ Stepper Driver และ Controller เพื่อจ่ายสัญญาณควบคุมที่มีความถี่ต่างกันให้กับ Stepper Motor ทดสอบการเปลี่ยนแปลงความเร็วเชิงมุมของมอเตอร์โดยการบันทึกค่าความเร็วที่วัดได้เมื่อปรับความถี่ของสัญญาณควบคุมในช่วงที่เพิ่มขึ้นอย่างต่อเนื่องจนถึงจุดอิ่มตัว (Saturation Point) จากนั้นใช้ Encoder ในการวัดองศาที่มอเตอร์หมุนไปจริงเทียบกับองศาที่ควรหมุนเพื่อหา Loss Step ในสภาวะที่เพิ่มกระแสโหลดหรือตั้งค่า Acceleration Ramp ให้สูงขึ้นเพื่อวิเคราะห์ปัจจัยที่ทำให้มอเตอร์เกิดการสูญเสียขั้นตอนการหมุน (Loss Step)

2.7.2 การทดลองที่ 2

ตั้งค่าการควบคุม Stepper Motor ด้วยการเลือกใช้งานรูปแบบ Drive ที่แตกต่างกัน ได้แก่ Full-Step, Half-Step และ Micro Step ทดสอบประสิทธิภาพโดยสังเกตความราบรื่นของการหมุนและวิเคราะห์แรงบิดกระเพื่อม (Torque Ripple) ในแต่ละรูปแบบการ Drive ต่อมาเปรียบเทียบประสิทธิภาพระหว่าง Stepper Drive แบบ Chopper Drive และ L/R Drive โดยวัดแรงบิดและความเร็วของมอเตอร์ในสภาวะที่กำหนดค่าแรงดันไฟฟ้าและความถี่ของสัญญาณควบคุมเหมือนกันสำหรับทั้งสอง Drive Mode

2.8 วัสดุอุปกรณ์

- 2.8.1 RS PRO Hybrid, Permanent Magnet Stepper Motor, 0.22Nm Torque, 2.8 V, 1.8°, 42.3 x 42.3mm Frame, 5mm Shaft จำนวน 1 อัน
- 2.8.2 Incremental Encoder AMT103-V จำนวน 1 อัน
- 2.8.3 WCS1700 Hall Current Sensor จำนวน 1 อัน
- 2.8.4 Cytron MDD20A Motor Driver จำนวน 1 อัน
- 2.8.5 Nucleo STM32G474RE 1 อัน
- 2.8.6 สายอัปโหลด จำนวน 1 ชุด
- 2.8.7 MotorXplorer จำนวน 1 ชุด (ฐานสามารถบรรจุบอร์ดควบคุม, Breadboard, 3D-Print ใช้สำหรับการประกอบกับ Stepper Motor)

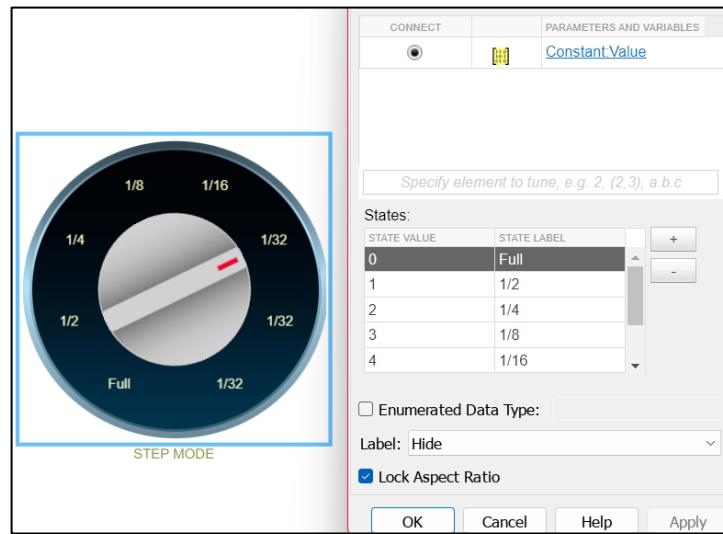
2.9 ขั้นตอนการดำเนินงาน

- 2.9.1 การทดลองหา Step Loss ของมอเตอร์เริ่มจากการที่คณะผู้ทดลองออกแบบการทดลองหา Step Loss ของ Stepper Motor ด้วยการเลือก Resolution ต่าง ๆ เช่น Full Step, Half Step และ Micro Step โดยการที่คณะผู้จัดทำได้ส่งเลือกค่าผ่านโปรแกรม Simulink ที่มีการเชื่อมต่อแบบ UART ผ่านเข้าหา Nucleo STM32G474RE ซึ่งสามารถเลือกค่าในการสั่ง Pin M0 M1 และ M2 ของบอร์ด Drive DRV8825 ได้ด้วยสลับความเป็นไปได้ทั้งหมด 8 แบบ

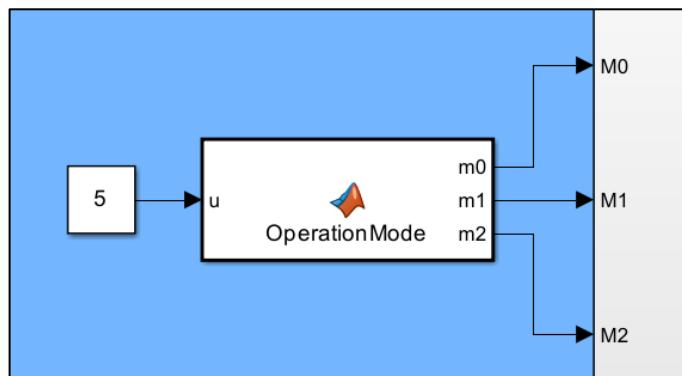
M0	M1	M2	Microstep Resolution
0	0	0	Full Step
1	0	0	Half Step
0	1	0	1/4 Step
1	1	0	1/8 Step
0	0	1	1/16 Step
1	0	1	1/32 Step
0	1	1	1/32 Step
1	1	1	1/32 Step

รูปที่ 47 แสดงถึงการเลือก Micro Step Resolution ด้วยการสั่งงานผ่านตัวรับ M0, M1 และ M2

ทางคณะผู้จัดทำจึงทำการเขียนฟังก์ชันเพื่อให้ง่ายต่อการเลือก Micro Step Resolution ในโปรแกรม Simulink ผ่านการใช้งาน Block MATLAB Function, Constant และ Rotary Switch



รูปที่ 48 แสดงให้เห็นถึงการใช้งาน Rotary Switch ในโปรแกรม MATLAB Simulink เพื่อควบคุม Micro Step Resolution ผ่านการควบคุมค่าของ Block Constant



รูปที่ 49 แสดงให้เห็นถึงการใช้ Constant Block ต่อเชื่อมเข้ากับ MATLAB Function Block เพื่อควบคุม Micro Step Resolution

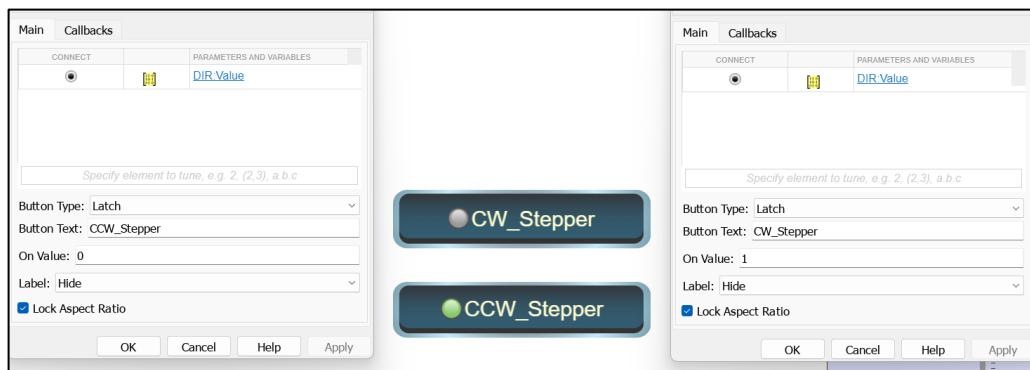
```

1 function [m0, m1, m2] = fcn(u)
2 % Initialize outputs
3 m0 = 0;
4 m1 = 0;
5 m2 = 0;
6 % Determine outputs based on input
7 if u == 0
8     m0 = 0; m1 = 0; m2 = 0;
9 elseif u == 1
10    m0 = 1; m1 = 0; m2 = 0;
11 elseif u == 2
12    m0 = 0; m1 = 1; m2 = 0;
13 elseif u == 3
14    m0 = 1; m1 = 1; m2 = 0;
15 elseif u == 4
16    m0 = 0; m1 = 0; m2 = 1;
17 elseif u == 5
18    m0 = 1; m1 = 0; m2 = 1;
19 elseif u == 6
20    m0 = 0; m1 = 1; m2 = 1;
21 elseif u == 7
22    m0 = 1; m1 = 1; m2 = 1;
23 end
24
25

```

รูปที่ 50 แสดงให้เห็นถึงโค้ดรายใน MATLAB Function Block ที่ใช้ในการควบคุม Micro Step Resolution โดยอิงค่า การนำเข้าข้อมูลมาจาก Constant Block

ตัวมาผู้จัดทำจึงได้สร้างปุ่มเพื่อใช้สำหรับการเลือกทิศทางการหมุนของ Stepper Motor ด้วยการ สั่ง Pin DIR บนบอร์ด DRV8825 ผ่านการสื่อสาร UART บนโปรแกรม MATLAB Simulink ไปยัง Nucleo STM32G474RE

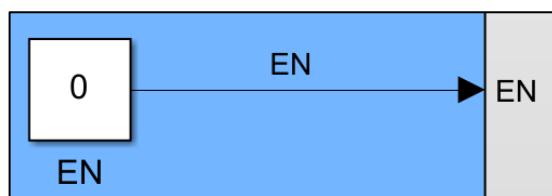


รูปที่ 51 แสดงถึงการใช้งาน Button และการตั้งค่าภายในร่วมกับ Block Constant เพื่อใช้สำหรับการควบคุมทิศทาง



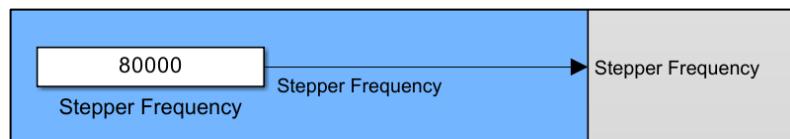
รูปที่ 52 แสดงการเชื่อมต่อ Constant Block กับ DIR ในโปรแกรม MATLAB Simulink เพื่อสั่งงานทิศทางการหมุน

การสั่งงานการเปิดปิดของ Stepper Motor นั้นสามารถทำได้ด้วยการสั่ง Pin EN ซึ่งย่อมาจาก Enable มีหน้าที่สำหรับการควบคุมการเปิดปิดของมอเตอร์ว่าจะทำงานหรือไม่ โดยที่การสั่งให้ Stepper Motor ทำงานนั้นจะต้องสั่งให้เป็น 0 เพราะว่า Pin นี้มีการทำงานแบบ Active Low หากสั่งให้ Pin นี้เป็น 1 (High) Pin นี้จะไม่ทำงานตามที่ได้ถูกสั่งไว้



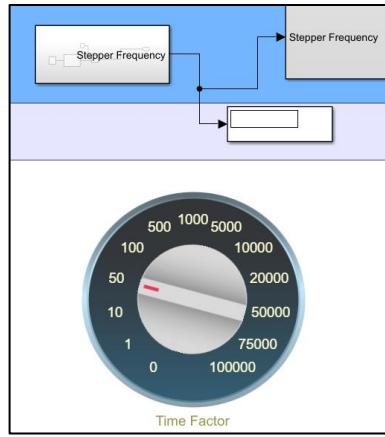
รูปที่ 53 แสดงการเชื่อมต่อ Constant Block กับ EN ในโปรแกรม MATLAB Simulink เพื่อสั่งงานเปิดปิดการทำงานของ Stepper Motor

คณะผู้จัดทำสามารถใส่ค่าความถี่เข้าไปในช่อง Stepper Frequency ซึ่งความถี่ที่สูงขึ้นจะส่งผลโดยตรงต่อความเร็วรอบที่เปลี่ยนแปลงไปอย่างมีนัยสำคัญ

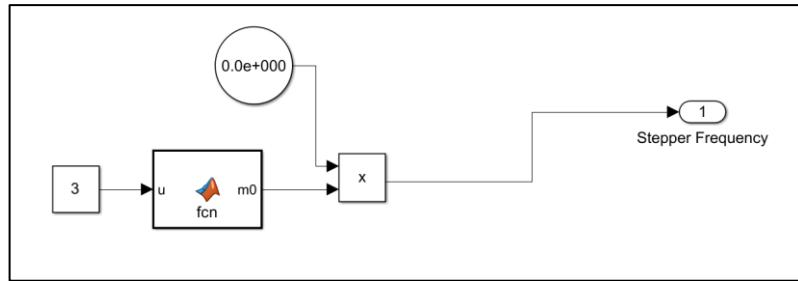


รูปที่ 54 แสดงการเชื่อมต่อ Constant Block กับช่อง Stepper Frequency ในโปรแกรม MATLAB Simulink

การเชื่อมต่อ Constant Block เพื่อควบคุมความถี่ที่เข้าในระบบนั้นยกต่อการเปลี่ยนแปลงค่าและยกต่อการที่จะควบคุมการเพิ่มความเร็ว ทางคณะผู้จัดได้ออกใช้เวลามาตรฐานกับตัวแปรคงที่ซึ่งสามารถทำให้ผู้จัดทำเลือกความเร็วในการของความถี่



รูปที่ 55 แสดงถึงการใช้ Rotary Switch ในและ Subsystem ในการควบคุมความเร่งของความถี่



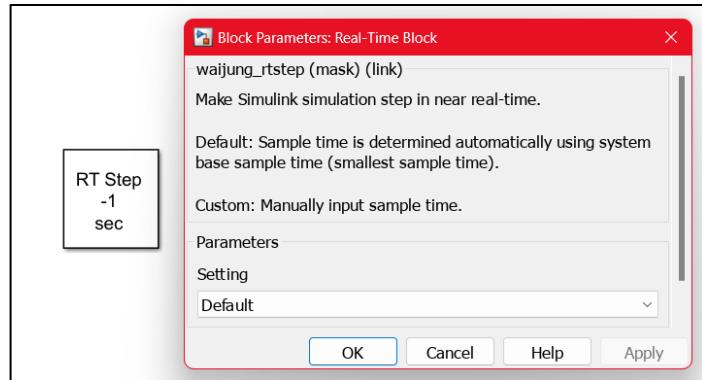
รูปที่ 56 แสดงแผนผังภายใน Simulink โดยใช้ Clock เป็นตัวแปรเวลาเข้ามาเพื่อคุณกับค่าคงที่ใน MATLAB Function

```

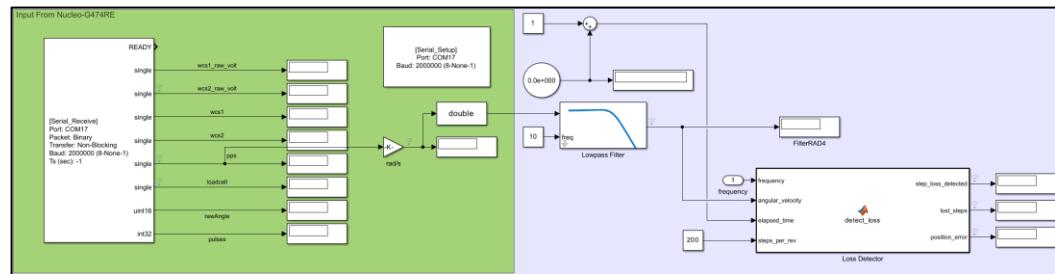
function m0 = fcn(u)
    % Initialize outputs
    m0 = 0;
    % Determine outputs based on input u
    if u == 0
        m0 = 0;
    elseif u == 1
        m0 = 1;
    elseif u == 2
        m0 = 10;
    elseif u == 3
        m0 = 50;
    elseif u == 4
        m0 = 100;
    elseif u == 5
        m0 = 500;
    elseif u == 6
        m0 = 1000;
    elseif u == 7
        m0 = 5000;
    elseif u == 8
        m0 = 10000;
    elseif u == 9
        m0 = 20000;
    elseif u == 10
        m0 = 50000;
    elseif u == 11
        m0 = 75000;
    elseif u == 12
        m0 = 100000;
    end

```

รูปที่ 57 แสดงถึงโค้ดภายในของ Block MATLAB Function เพื่อใช้ในการควบคุมตัวแปรที่ใช้คุณกับเวลา
แต่การที่ใช้เวลาที่มาจากการ MotorXplorer ที่ได้มานั้นจะไม่ตรงกับเวลาของโลกแท้ๆ ความ
เป็นจริง จึงต้องใช้ Real Time Block จาก Library waijung ซึ่งสามารถทำให้คณะผู้จัดทำได้ค่าเวลาอ้างอิง
และตรงจากความเป็นจริงมากที่สุดซึ่งจะได้ออกมาในหน่วยวินาที

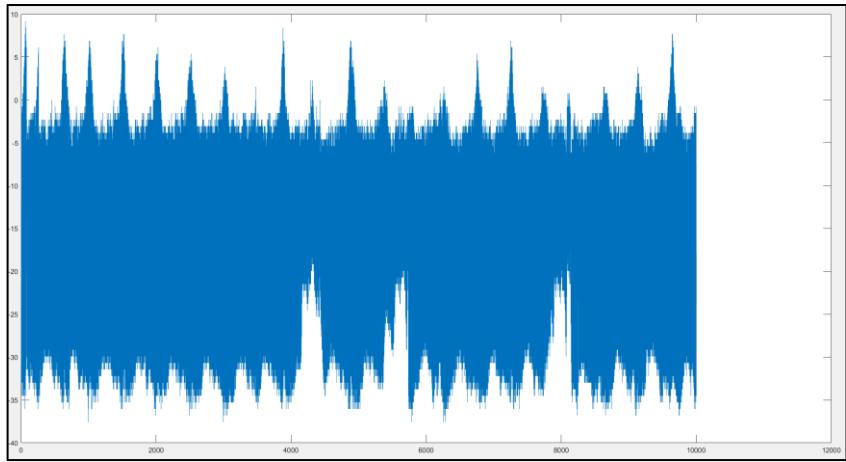


รูปที่ 58 แสดงถึงภาพของ Real Time Block และการตั้งค่า Parameter ภายใน



รูปที่ 59 แสดงถึงการวาง Simulink Block เพื่อที่จะตรวจสอบค่าที่วัดได้จากเซนเซอร์ต่าง ๆ

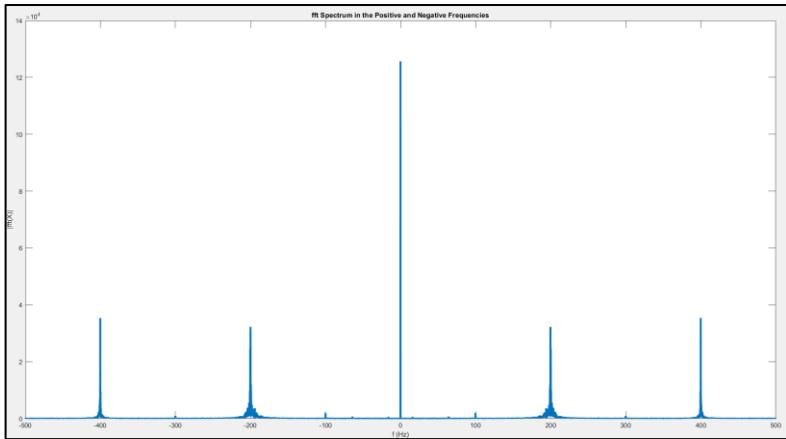
คณะผู้จัดทำได้ทำการกรองค่าความเร็วเชิงมุ่งจาก AMT103-V Incremental Encoder แต่สัญญาณที่ได้มาันี้มีสัญญาณอิ่นรบกวนอยู่ คณะผู้จัดทำจึงต้องใช้หลักการ Signal Conditioning ในส่วนของการกรองสัญญาณ (Filtering) ในการกรองคลื่นความถี่ที่ไม่เกี่ยวข้องออกซึ่งจะใช้ Low Pass Filter ใน การกรอง Noise หรือคลื่นสัญญาณรบกวนที่ได้มากจาก Encoder ให้น้อยมากที่สุดเท่าที่จะเป็นไปได้



รูปที่ 60 แสดงถึงกราฟของสัญญาณที่ได้จาก Encoder ผ่านการหมุนของ Stepper Motor คณะผู้จัดทำจะได้วิเคราะห์ช่วงสัญญาณเพื่อหาช่วงความถี่ที่เกิดขึ้นด้วยการใช้หลักการ Fast Fourier Transform ซึ่งถูกนำมาใช้งานร่วมกับโปรแกรม MATLAB เพื่อแยกหาช่วงของสัญญาณที่ควรถูกนำไปใช้

```
>> Fs = 1000; % Sampling frequency
T = 1/Fs; % Sampling period
L = 10001; % Length of signal
value = data.Data;
time = (linspace(0,10, 10001))';
y = fft(value);
plot(Fs/L*(-L/2:L/2-1),abs(fftshift(y)), "LineWidth",3)
title("fft Spectrum in the Positive and Negative Frequencies")
xlabel("f (Hz)")
ylabel ("|fft(X)|")
```

รูปที่ 61 แสดงถึง Code MATLAB ที่ใช้ในการแยกความแรงของสัญญาณตามความถี่ด้วย FFT



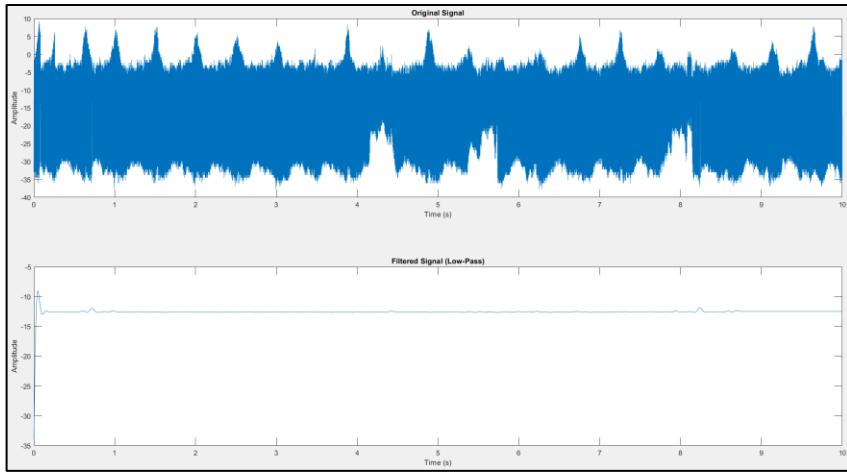
รูปที่ 62 แสดงถึงการกราฟความสัมพันธ์ระหว่างความแรงของสัญญาณต่อความถี่
จากรูปที่ 62 แสดงให้คณะผู้จัดทำเห็นได้อย่างชัดเจนว่าสัญญาณที่คณะผู้จัดทำต้องการนั่นอยู่ที่
ประมาณ 0Hz และมีสัญญาณรบกวนหนักช่วง 200Hz กับ 400Hz ตามลำดับ คณะผู้จัดทำจึงเลือกใช้ Cut
Off Frequency ให้อยู่ที่ 10Hz เพื่อที่จะกรองค่าสัญญาณรบกวนที่มีความรุนแรงออกไปทั้งหมด ซึ่งทาง
คณะผู้จัดทำจึงได้ทดสอบความสามารถในการลดการรบกวนของ Noise ด้วยการใช้ Butterworth Low
Pass Filter

```

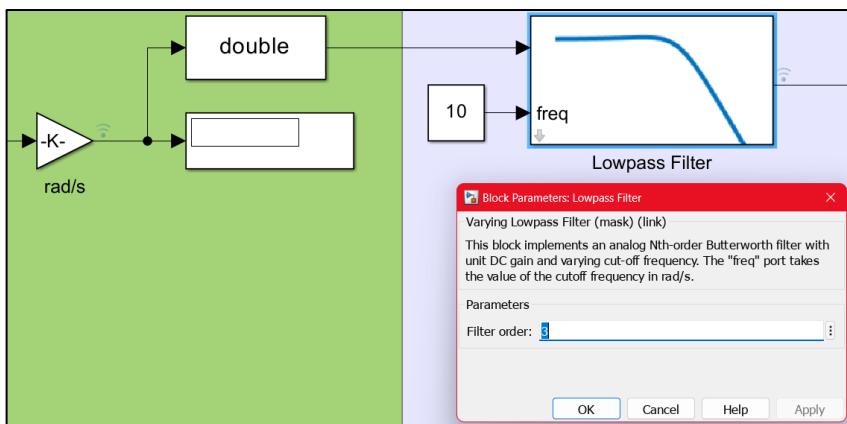
>> fs = 1000;
time = 0:0.001:10;
cutoff_frequency = 10;
[b, a] = butter(4, cutoff_frequency/(fs/2), 'low');
filtered_signal = filtfilt(b, a, signal);
figure;
subplot(2,1,1);
plot(time, signal);
title('Original Signal');
xlabel('Time (s)');
ylabel('Amplitude');
subplot(2,1,2);
plot(time, filtered_signal);
title('Filtered Signal (Low-Pass)');
xlabel('Time (s)');
ylabel('Amplitude');

```

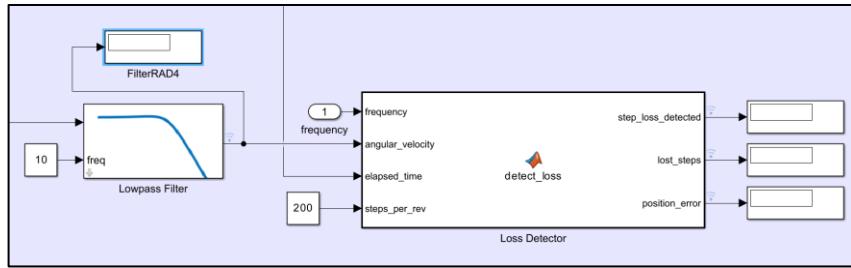
รูปที่ 63 แสดงถึงโค้ด Butterworth Low Pass Filter ที่ใช้สำหรับการกรองสัญญาณรบกวน



รูปที่ 64 แสดงถึงการเปรียบเทียบระหว่างสัญญาณที่ยังไม่ได้ถูกกรองต่อสัญญาณที่ถูกกรองแล้ว
คณะผู้จัดทำจึงได้ทำการประยุกต์ใช้ค่า Cut Off Frequency และ Butterworth Low Pass
Filter กับ Simulink เพื่ogrung สัญญาณrgb กวนจาก Encoder ออก



รูปที่ 65 แสดงถึงการประยุกต์ใช้ Block Varying Low Pass Filter และ Constant ในการกรองข้อมูล
ทางคณะผู้จัดทำจึงได้ทำการตรวจจับหา Step Loss ของมอเตอร์จากการหมุนหนึ่งรอบโดยการใช้
ความเร็วเขิงมุ่งมาคิดคำนวณย้อนกลับผ่านการใช้ MATLAB Function Block



รูปที่ 66 แสดงถึงการใช้ MATLAB Function Block เพื่อคิดคำนวณ Step Loss

```

1 function [step_loss_detected, lost_steps, position_error] = detect_loss(frequency, angular_velocity, elapsed_time, steps_per_rev)
2 % Inputs:
3 % frequency: Drive frequency of the motor (Hz)
4 % angular_velocity: Measured angular velocity (rad/s)
5 % elapsed_time: Time since the start (s)
6 % steps_per_rev: Number of steps per revolution of the motor (integer)
7 % Outputs:
8 % step_loss_detected: Boolean (1 if step loss is detected, 0 otherwise)
9 % lost_steps: Total accumulated lost steps
10 % position_error: Difference between commanded and actual position (radians)
11 % expected_step: Commanded step count based on drive frequency
12 persistent accumulated_lost_steps prev_elapsed_time total_expected_steps;
13 if isempty(accumulated_lost_steps)
14     accumulated_lost_steps = 0; % Initialize lost steps
15     prev_elapsed_time = 0; % Initialize previous elapsed time
16     total_expected_steps = 0; % Initialize total expected steps
17 end
18 step_angle = 2 * pi / steps_per_rev;
19 time_increment = elapsed_time - prev_elapsed_time;
20 prev_elapsed_time = elapsed_time;
21 commanded_steps_increment = frequency * time_increment;
22 total_expected_steps = total_expected_steps + commanded_steps_increment;
23 expected_step = (total_expected_steps);
24 commanded_position = total_expected_steps * step_angle;
25 actual_position = angular_velocity * elapsed_time;
26 position_error = commanded_position - actual_position;
27 incremental_lost_steps = round(position_error / step_angle);
28 accumulated_lost_steps = accumulated_lost_steps + incremental_lost_steps;
29 lost_steps = accumulated_lost_steps;
30 if abs(position_error) > (0.5 * step_angle)
31     step_loss_detected = 1;
32 else
33     step_loss_detected = 0;
34 end
35 end

```

รูปที่ 67 แสดงถึงโค้ดภาษาใน MATLAB Function Block ที่ใช้สำหรับการคำนวณหา Step Loss

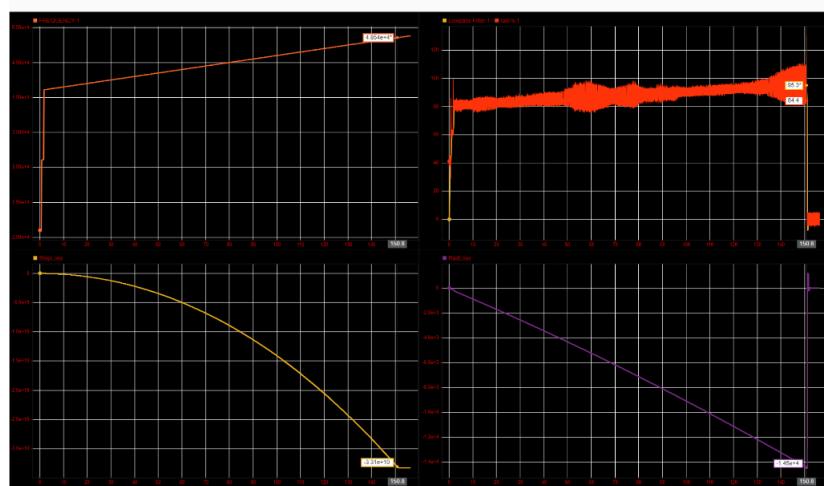
2.10 ผลการทดลอง

2.10.1 การทดลองหา Step Loss และการเปลี่ยนแปลง Micro Step Resolution

ทางคณะผู้จัดทำได้ทำการทดสอบว่าหาก Stepper Motor ถูกสั่งให้หมุนด้วยความเร็วสูงในระยะเวลาอันสั้น หรือว่าเร่งให้หมุนเร็วเกินไป Stepper Motor จะเกิด Step Loss อย่างสมบูรณ์ หรือว่าคือ Stepper Motor จะไม่หมุนตามที่สั่งได้อีกต่อไปแล้ว เพราะความถี่ที่ใช้ในการสั่งนั้นสูงเกินไป ผู้ทดลองจึงพยายามใช้ความเร่งที่มีค่าที่เพิ่มขึ้นอย่างช้า ๆ และให้เหมาะสมกับ Micro Step Resolution ต่อ ๆ กัน



รูปที่ 68 แสดงถึงกราฟความถี่, ความเร็วเชิงมุม(ทั้งก่อนและหลัง Lowpass Filter), TotalLoss (Step Loss ทั้งหมด) และ RadLoss (Radian ที่หายไปต่อการหมุน 1 รอบ) ในทิศทางตามเข็มนาฬิกา



รูปที่ 69 แสดงถึงกราฟความถี่, ความเร็วเชิงมุม (ทั้งก่อนและหลัง Lowpass Filter), TotalLoss (Step Loss ทั้งหมด) และ RadLoss (Radian ที่หายไปต่อการหมุน 1 รอบ) ในทิศทางทวนเข็มนาฬิกา

ทางคณะผู้จัดจึงทำการทดลองการเปลี่ยน Micro Step Resolution และการเลือกทิศทางการหมุนที่แตกต่างกันจากนั้นจึงบันทึกผลการทดลองเพื่อหาแนวโน้มและจุดความเชื่อมโยง

CCW					
Mode	Loss Frequency	Max Rad	Total Step Loss	Rad Loss per Rev	
1	8886.1	-139.6	7.28E+11	4.86E+04	
1/2	7869.5	-123.9	1.32E+10	5.51E+03	
1/4	1.65E+04	-129.9	1.05E+11	3.26E+04	
1/8	1.77E+04	-69.3	4.64E+10	1.61E+04	
1/16	4.85E+04	-95.2	6.34E+11	7.16E+04	
1/32	1.11E+05	-108.5	4.80E+10	2.17E+04	

รูปที่ 70 แสดงถึงตารางความถี่, ความเร็วเชิงมุมหลัง Lowpass Filter, TotalLoss (Step Loss ทั้งหมด) และ RadLoss (Radian ที่หายไปต่อการหมุน 1 รอบ) ในทิศทางทวนเข็มนาฬิกา

CW					
Mode	Loss Frequency	Max Rad	Total Step Loss	Rad Loss per Rev	
1	2973.6	124.3	-4.54E+10	-1.85E+04	
1/2	1.23E+04	193.1	-1.10E+11	-4.38E+04	
1/4	1.65E+04	129.5	-1.37E+11	-4.03E+04	
1/8	1.77E+04	69.3	-1.73E+10	-9.31E+03	
1/16	4.85E+04	84.4	-3.31E+10	-1.45E+04	
1/32	9.80E+04	96.2	-2.98E+10	-1.36E+04	

รูปที่ 71 แสดงถึงตารางความถี่, ความเร็วเชิงมุมหลัง Lowpass Filter, TotalLoss (Step Loss ทั้งหมด) และ RadLoss (Radian ที่หายไปต่อการหมุน 1 รอบ) ในทิศทางตามเข็มนาฬิกา

จากรูปที่ 70 และ 71 จะสังเกตเห็นได้อย่างชัดเจนว่าในโหมดที่ 1 ที่มี Micro Step Resolution ละเอียดน้อยที่สุด จะเข้าสู่ช่วง Step Loss สมบูรณ์ได้ด้วยความถี่ที่น้อยที่สุด ซึ่งตรงข้ามกับโหมดที่ 1/32 ที่มี Micro Step Resolution ละเอียดสูงสุด จะเห็นได้ว่าเข้าสู่ช่วง Step Loss สมบูรณ์ได้ด้วยความถี่ที่สูงที่สุด ไม่ว่าจะเป็นช่วงการหมุนทวนเข็มหรือตามเข็มก็ตาม

2.11 สรุปผล

จากการที่คณะผู้จัดทำได้ทำการทดลองเก็บข้อมูลมาจำนวนหลายครั้งทำให้ทราบได้ว่า ไม่สามารถที่จะทำการสรุปผลให้มีความชัดเจนได้อันเนื่องมาจากแรงเสียดทานที่เกิดขึ้นในตัว Instrumental Bearing บนบอร์ด MotorXplorer และความ Misalignment ที่ทำให้แรงเสียดทานนั้นเพิ่มขึ้นและไม่ได้มีความคงที่มากเท่าที่ควรจึงเป็นเหตุให้ไม่สามารถสรุปการทดลองได้ทั้งหมด ยกเว้นเพียงแต่แนวโน้มความน่าจะเป็นในเรื่องของ การที่ Stepper Motor ลึกลงๆ Step Loss อย่างสมบูรณ์ก็ต่อเมื่อความถี่นั้นสูงเกินไปหรือมีแรงเสียดทานเข้ามากระทำในระบบมากเกินไปจนทำให้ Rotor ภายใน Stepper Motor ไม่สามารถหมุนอังอิจตามความถี่นั้นได้ทัน โดยที่ถ้าหากยังเลือกใช้โหมดการทำงานที่มีความละเอียดสูงมากเท่าใดก็จะยังทำให้ความถี่ที่จะเกิดการ Loss Step แบบสมบูรณ์สูงมากขึ้นเท่านั้น

2.12 อภิปรายผล

จากการทดลองที่คณะผู้จัดทำได้ทำการเก็บข้อมูล พบว่า Step Loss ของ Stepper Motor เกิดขึ้นในกรณีที่ความถี่ของสัญญาณควบคุมสูงเกินไป หรือเกิดแรงเสียดทานในระบบมากเกินไป ทำให้ Rotor ไม่สามารถหมุนได้ตามที่กำหนด โดยสามารถสังเกตได้ว่าในโหมด Micro Step Resolution ที่มีค่าความละเอียดสูง เช่น 1/32 Step จะสามารถรับความถี่สูงได้มากกว่าโหมดที่มีความละเอียดต่ำ เช่น Full Step แต่การตั้งค่าความละเอียดสูงอาจทำให้ความเร็วในการหมุนลดลง จึงควรเลือก Micro Step Resolution ให้เหมาะสมกับการใช้งาน ที่ต้องการทั้งความเร็วและแรงบิด ในกรณีที่เกิด Step Loss อย่างสมบูรณ์ จะพบว่าความถี่ที่ใช้ควบคุมสูงกว่าความถี่ที่ Rotor สามารถตอบสนองได้ทัน ทำให้เกิดการหมุนที่ไม่สัมพันธ์กับคำสั่งที่ส่งไป นอกจากนี้ แรงเสียดทานภายในระบบ เช่น Frictional Load จาก Bearing หรือความ Misalignment ของระบบเครื่องกล ยังเป็นปัจจัยสำคัญที่ทำให้เกิดความไม่เสถียรและลดความแม่นยำของการทดลอง

การใช้ Low Pass Filter เพื่อกรองสัญญาณรบกวนจาก Encoder พบว่าสามารถช่วยเพิ่มความแม่นยำของข้อมูลได้ การวิเคราะห์สัญญาณด้วย FFT แสดงให้เห็นว่า Noise ที่เกิดขึ้นมีความถี่สูงกว่า 200 Hz ซึ่งถูกกรองออกด้วย Butterworth Low Pass Filter ที่กำหนด Cut Off Frequency ไว้ที่ 10 Hz ช่วยลดสัญญาณรบกวนได้อย่างชัดเจน อย่างไรก็ตาม ความไม่สม่ำเสมอของแรงเสียดทานจาก Instrumental Bearing และความ Misalignment ของระบบ ส่งผลกระทบต่อผลการทดลอง ทำให้ไม่สามารถสรุปค่าที่แน่นอนได้อย่างสมบูรณ์ ผลที่ได้เป็นเพียงแนวโน้มและข้อสังเกตเบื้องต้น ซึ่งควรมีการควบคุมปัจจัยเหล่านี้ให้เกิดขึ้น้อยที่สุดให้ได้ในอนาคต

2.13 ข้อเสนอแนะ

1) การปรับปรุงระบบกลไก

- ควรตรวจสอบและปรับแกนการหมุนของ Stepper Motor ให้มีความสม่ำเสมอ และลดปัญหา Misalignment เพื่อควบคุมแรงเสียดทานให้คงที่มากขึ้น
- ใช้ Bearing คุณภาพสูงเพื่อช่วยลดแรงเสียดทานและเพิ่มความราบรื่นในการหมุน

2) การปรับปรุงการตั้งค่าทางไฟฟ้า

- ทดสอบการตั้งค่า Micro Step Resolution ในช่วงความละเอียดที่หลากหลาย เพื่อหาจุดที่เหมาะสมระหว่างความแม่นยำและแรงบิด
- ใช้ Driver ที่มีประสิทธิภาพสูงกว่า เช่น Chopper Drive เพื่อช่วยควบคุมกระแสไฟฟ้าได้อย่างแม่นยำ

3) การเก็บข้อมูลที่ละเอียดมากขึ้น

- เพิ่มเซนเซอร์เพื่อตรวจจับข้อมูลเพิ่มเติม เช่น Torque Sensor เพื่อวิเคราะห์แรงบิดที่ส่งผลต่อ Step Loss
- ใช้ Sampling Rate สูงขึ้นใน ADC เพื่อให้ได้ข้อมูลที่แม่นยำและสมบูรณ์มากขึ้น

4) การวิเคราะห์เพิ่มเติม

- ใช้เทคนิคการกรองแบบ Adaptive Filtering เพื่อปรับแต่งการกรองสัญญาณให้เหมาะสมกับ Noise ที่เปลี่ยนแปลง
- ทดสอบการปรับ Acceleration Ramp และ Deceleration Ramp ในลักษณะ S-Curve เพื่อช่วยลดแรงกระชากและเพิ่มความราบรื่นในการหมุน

5) การทดลองในสภาพแวดล้อมที่ควบคุมได้

- ทดสอบในสภาพแวดล้อมที่ไม่มีแรงสั่นสะเทือนและแรงกระแทกจากภายนอก เพื่อลดปัจจัยรบกวน

2.14 เอกสารอ้างอิง

- [1] Factomart.com, หลักการทำงานของ Stepping Motor, Available online [ม.ป.ป.] :
<https://mall.factomart.com/principle-of-Stepping-Motor/>
- [2] FAULHABER, How to recognize and prevent Step losses with Stepper Motors, Available online [ม.ป.ป.] : <https://www.faulhaber.com/en/know-how/tutorials/Stepper-Motor-tutorial-how-to-prevent-Step-losses-with-Stepper-Motors/>
- [3] OrientalMotor, Stepper Motor Basic Characteristics, Available online [ม.ป.ป.] :
<https://www.orientalMotor.com.sg/sg/tech/reference/Stepping-Motor03>
- [4] Raj Borkar, Mechtex, Types of Stepper Motor, Available online [ม.ป.ป.] :
<https://mechtex.com/blog/types-of-Stepper-Motor>
- [5] Mouser, Unipolar vs. Bipolar Drive for Stepper Motors, Available online [Oct 3, 2024] :
<https://www.mouser.co.cr/blog/unipolar-vs-bipolar-drive-for-Stepper-Motors?srsltid=AfmBOoqj8K667NO3hn2oKjPG8vbwohyUXNGdyWXdqmcD-FA11kcebFa0>
- [6] Monolithic Power System, Basics of Signal Conditioning, Available online [ม.ป.ป.] :
https://www.monolithicpower.com/jp/learning/mpscholar/sensors/basics-of-sensor-operation/basics-Conditioning?srsltid=AfmBOorO7MG3HVRWJjM_YOjz7wJ95i7_neDSBOvx6zfiNXenwgWJOFht

การทดลองที่ 3

Brushless DC Motor

3.1 จุดประสงค์

- 3.1.1 เพื่อศึกษาหลักการทำงานของ Brushless DC Motor
- 3.1.2 เพื่อศึกษาวิธีนักศึกษาต้องเข้าใจวิธีการตั้งค่า Motor ในโปรแกรม Motor Workbench 6.3.2
- 3.1.3 เพื่อเข้าใจวิธีการตั้งค่า PWM Generation Frequency และ Speed Sensing
- 3.1.4 เพื่อเข้าใจ Electrical parameters, Mechanical parameters, และ Motor magnetic Structure ของ BLDC Motor
- 3.1.5 เพื่อเข้าใจ BLDC Motor control types ทั้งในรูปแบบ Sensorless Control และ Sensor-Based Control
 - 3.1.6 เพื่อเข้าใจและอธิบายกราฟสัญญาณของ BLDC Motor ทั้ง 3 Phases จากพฤติกรรมที่สังเกตว่ามาจากการควบคุมแบบไหน และสาเหตุที่กราฟมีลักษณะเช่นนี้โดยละเอียด และวิเคราะห์โดยใช้ผลการเรียนรู้ย่อที่เกี่ยวข้องมาอธิบายและอภิปรายในผลการทดลอง ครอบคลุมถึง
 - 1) Brushless DC (BLDC) Motor
 - 2) Trapezoidal Back-EMF
 - 3) Permanent Magnet Synchronous Machine (PMSM)
 - 4) Trapezoidal Control o Field-Oriented Control (FOC)
 - 5) Pulse Width Modulation (PWM)
 - 6) Commutation Logic o Six-Step Commutation o Space Vector Modulation (SVM)
 - 7) Sensorless Control
 - 8) Hall Effect Sensors
- โดยสามารถนำผลการทดลองไปเปรียบเทียบกับพฤติกรรมการควบคุม PMSM แบบ FOC ด้วยการสืบค้นจากแหล่งข้อมูลภายนอก เช่น MATLAB
- 3.1.7 เพื่อศึกษาวิธีการคำนวณหาความเร็วของ BLDC Motor จาก Frequency ของสัญญาณที่อ่านได้จาก Oscilloscope
- 3.1.8 เพื่อศึกษาวิธีการใช้งาน Oscilloscope 4 Channels ในการจับสัญญาณของ BLDC Motor ทั้ง 3 Phases

3.2 สมมติฐาน

3.2.1 การทดลองที่ 1

- 1) ความเร็วของมอเตอร์ (RPM) จะแปรผันตรงกับความถี่ของสัญญาณที่วัดได้จากเซนเซอร์
- 2) การเพิ่มค่าความถี่ของ PWM จะส่งผลต่อความเร็วของมอเตอร์ โดยตรง และความสัมพันธ์จะอยู่ในช่วงเชิงเส้นจนถึงจุดอิ่มตัว (Saturation Point)

3.2.2 การทดลองที่ 2

- 1) การควบคุมแบบ Field Oriented Control (FOC) จะให้แรงบิดที่เสถียรและการตอบสนองที่ดีกว่าการควบคุมแบบ Trapezoidal Control หรือ Sinusoidal Control
- 2) การควบคุมแบบ Sensor-Based (Hall Sensor) จะมีความแม่นยำในการรัดความเร็วและตำแหน่งมากกว่าการควบคุมแบบ Sensorless (Back-EMF)

3.3 ตัวแปร

3.3.1 การทดลองที่ 1

- ตัวแปรต้น : ความถี่ของสัญญาณ (Hz)
ตัวแปรตาม : ความเร็วรอบมอเตอร์ (RPM)
ตัวแปรควบคุม : แรงดันไฟฟ้าขาเข้า, โหลดบนมอเตอร์

3.3.2 การทดลองที่ 2

- ตัวแปรต้น : เทคนิคการควบคุมแบบ FOC, Trapezoidal และ Sinusoidal
ตัวแปรตาม : ความเสถียรของแรงบิด, ประสิทธิภาพการควบคุม
ตัวแปรควบคุม : แรงดันไฟฟ้าขาเข้า, ความเร็วที่ตั้งไว้ (Setpoint)

3.4 นิยามศัพท์เฉพาะ

- 3.4.1 Brushless DC Motor (BLDC Motor): มอเตอร์ไฟฟ้ากระแสสลับที่ไม่มีแปรงถ่าน ใช้แม่เหล็กถาวรและวงจรริเล็กทรอนิกส์สำหรับการควบคุมการทำงาน มักถูกใช้ในแอปพลิเคชันที่ต้องการการควบคุมความเร็วที่แม่นยำ
- 3.4.2 Hall Sensor: เซนเซอร์ที่ใช้หลักการของ Hall Effect เพื่อตรวจจับตำแหน่งของแม่เหล็กในมอเตอร์ ใช้สำหรับดัดความเร็วและตำแหน่งของ Rotor ใน BLDC Motor
- 3.4.3 Back-EMF (Electromotive Force): แรงดันไฟฟ้าที่เกิดขึ้นในชุดลวดของมอเตอร์อันเนื่องมาจากการหมุนของ Rotor ซึ่งใช้สำหรับการควบคุมแบบ Sensorless
- 3.4.4 Pulse Width Modulation (PWM): เทคนิคการควบคุมแรงดันไฟฟ้าโดยการสลับสัญญาณเพื่อควบคุมความเร็วและแรงบิดของมอเตอร์
- 3.4.5 Field Oriented Control (FOC): เทคนิคการควบคุมมอเตอร์ขั้นสูงโดยการใช้ Space Vector Modulation (SVM) เพื่อเพิ่มประสิทธิภาพการควบคุมกระแสไฟฟ้าและแรงบิด
- 3.4.6 Sensorless Control: การควบคุมมอเตอร์โดยไม่ใช้เซนเซอร์ในการตรวจจับตำแหน่ง แต่ใช้ข้อมูลจาก Back-EMF แทน
- 3.4.7 Trapezoidal Control: การควบคุม BLDC Motor ด้วยคลื่นสัญญาณรูปสี่เหลี่ยมคงหู หมายความว่า สำหรับการควบคุมที่ไม่ซับซ้อนแต่ประสิทธิภาพการควบคุมต่ำกว่าวิธีอื่น
- 3.4.8 Sinusoidal Control: การควบคุมมอเตอร์โดยการป้อนคลื่นสัญญาณรูปไซน์ เพื่อการทำงานที่นุ่มนวลขึ้น แต่ซับซ้อนกว่าการควบคุมแบบ Trapezoidal
- 3.4.9 6-Step Commutation: วิธีการสลับกระแสไฟฟ้าในชุดลวดของมอเตอร์ BLDC โดยแบ่งหนึ่งรอบการหมุนออกเป็น 6 ขั้นตอน เพื่อให้ Rotor หมุนไปตามลำดับที่กำหนด ใช้ตรวจสอบความสมดุลของความเร็ว และแรงบิดในมอเตอร์ BLDC
- 3.4.10 PWM (Pulse Width Modulation): เทคนิคการควบคุมกระแสไฟฟ้าที่จ่ายให้กับมอเตอร์โดยปรับ Duty Cycle ของสัญญาณเพื่อควบคุมแรงดันไฟฟ้า ใช้ในการทดลองเพื่อปรับความเร็วและแรงบิดของมอเตอร์
- 3.4.11 Torque Ripple: การสั่นสะเทือนของแรงบิดที่เกิดขึ้นในมอเตอร์ระหว่างการหมุน สังเกตได้ในกระบวนการควบคุมแบบ Trapezoidal หรือ 6-Step
- 3.4.12 Zero Crossing Point: จุดที่แรงเคลื่อนไฟฟ้า Back-EMF เปลี่ยนค่าจากบวกเป็นลบหรือกลับกัน ใช้สำหรับการคำนวณตำแหน่ง Rotor ในการควบคุมแบบ Sensorless

3.4.13 RMS (Root Mean Square): ค่ารากกำลังสองเฉลี่ย ใช้สำหรับวิเคราะห์ค่าความเร็วรอบเฉลี่ย (Mechanical Speed) ในการทดลองเพื่อให้ได้ค่าที่เสถียรจากข้อมูลที่ผันผวน

$$RMS = \sqrt{\frac{1}{n} \sum_{i=0}^n X_i^2}$$

เมื่อ RMS คือค่าเฉลี่ยของกำลังสอง

n คือจำนวนข้อมูลทั้งหมด

X_i คือข้อมูล Raw Data ตัวที่ i โดยที่ $i = 1, 2, 3, \dots, n$

3.4.14 Field Oriented Control (FOC): วิธีการควบคุมมอเตอร์ที่รักษามุ่งระหว่างสนามแม่เหล็กของ Rotor และ Stator ให้อยู่ที่ 90 องศา เพื่อให้ได้แรงบิดและประสิทธิภาพสูงสุด หมายความว่าสามารถควบคุมมอเตอร์ BLDC ที่ต้องการความแม่นยำสูง

3.4.15 Hall Effect Sensor: เซนเซอร์ที่ตรวจจับการเปลี่ยนแปลงของสนามแม่เหล็กใน Rotor เพื่อบอกตำแหน่ง ใช้ในการควบคุมแบบ Sensor-Based

3.4.16 Sensorless Control: การควบคุมมอเตอร์โดยไม่ใช้เซนเซอร์ภายนอก แต่ใช้ Back-EMF หรือ แรงดันไฟฟ้าเพื่อคำนวณตำแหน่ง Rotor หมายความว่าสามารถลดต้นทุนและความซับซ้อน.

3.4.17 Reference Speed: ความเร็วเป้าหมายที่ระบบควบคุมสั่งให้มอเตอร์หมุน ใช้เปรียบเทียบกับ Mechanical Speed เพื่อวิเคราะห์ความแม่นยำ

3.4.18 Target Speed: ความเร็วที่ผู้ใช้งานกำหนดเพื่อให้มอเตอร์หมุนตามคำสั่งในโปรแกรม Motor Pilot

3.4.19 Mechanical Speed: ความเร็วรอบจริงของมอเตอร์ที่วัดได้ผ่านการตรวจจับหรือการคำนวณ สังเกตได้จากการทดลองใน Motor Pilot หรือ Oscilloscope

3.4.20 Phase Shift: ความต่างเฟสระหว่างสัญญาณของแต่ละเฟสในมอเตอร์ BLDC ใช้ตรวจสอบความสมพันธ์ ระหว่างเฟส U, V, W ในการควบคุมแบบ 6-Step

3.4.21 Trapezoidal Control: วิธีการควบคุมมอเตอร์ที่ใช้รูปคลื่นแรงดัน Back-EMF แบบ Trapezoidal หมายความว่า สำหรับการควบคุมมอเตอร์ BLDC ด้วยต้นทุนต่ำ

- 3.4.22 Sinusoidal Control: วิธีการควบคุมมอเตอร์ที่ใช้รูปคลื่นแรงดัน Back-EMF แบบ Sinusoidal เพื่อการควบคุมแรงบิดที่ราบรื่น
- 3.4.23 Oscilloscope: เครื่องมือสำหรับวัดและแสดงสัญญาณไฟฟ้าในรูปแบบกราฟ ใช้ในการตรวจสอบสัญญาณ PWM, Back-EMF และรูปคลื่น 6-Step ของมอเตอร์ BLDC
- 3.4.24 Cheat Engine: Software Open Source ออกแบบมาเพื่อแก้ไขค่าต่าง ๆ ในหน่วยความจำ (Memory Address) ของโปรแกรมที่กำลังทำงานอยู่ ใช้สำหรับการตัดแปลงเกมหรือโปรแกรมเพื่อเปลี่ยนแปลงตัวแปรต่าง ๆ เช่น ตัวเลข, ค่าเงิน, คะแนน หรือสถานะของผู้เล่น

3.5 นิยามเชิงปฏิบัติการ

- 3.5.1 BLDC Motor Speed: มอเตอร์กระแสตรงแบบไร้แปรงถ่านที่ใช้ Rotor เป็นแม่เหล็ก固定 และ Stator เป็นชุดลวดแม่เหล็กไฟฟ้า ใช้ในการทดลองเพื่อศึกษาและควบคุมความเร็วรอบและแรงบิดผ่านกระบวนการควบคุมด้วย X-NUCLEO-IHM08M1 และ STM32G474RE
- 3.5.2 Back-EMF Measurement: แรงดันไฟฟ้าที่เกิดจากการหมุนของ Rotor BLDC Motor โดยใช้ในการทดลอง Sensorless Control เพื่อคำนวณตำแหน่ง Rotor และตรวจสอบความสัมพันธ์ระหว่างความถี่และความเร็วรอบ
- 3.5.3 PWM Frequency: เทคนิคการควบคุมพลังงานโดยปรับ Duty Cycle ของสัญญาณ ใช้ในการทดลองเพื่อควบคุมแรงดันไฟฟ้าให้เหมาะสมกับความเร็วรอบเป้าหมาย (Target Speed) และตรวจสอบ Duty Cycle ด้วย Oscilloscope
- 3.5.4 Field Oriented Control (FOC) ใน BLDC Motor: การตั้งค่ากระแสไฟฟ้าและการเปลี่ยนเวกเตอร์ด้วย SVM เพื่อควบคุมแรงบิดของ BLDC Motor ในการทดลองนี้จะบันทึกค่าแรงบิดและพิจารณาประสิทธิภาพของมอเตอร์ภายใต้การควบคุมแบบ FOC
- 3.5.5 Sensor-Based Control และ Sensorless Control: กำหนดการใช้งาน Hall Sensor สำหรับ Sensor-Based Control และการใช้ Back-EMF สำหรับ Sensorless Control เพื่อสังเกตการตอบสนองของ BLDC Motor ภายใต้การควบคุมทั้งสองแบบ
- 3.5.6 Zero Crossing Point: จุดที่แรงดัน Back-EMF เปลี่ยนจากบวกเป็นลบหรือกลับกัน ใช้ในการทดลองควบคุมแบบ Sensorless เพื่อกำหนดตำแหน่ง Rotor และวิเคราะห์ความสัมพันธ์ของการหมุนกับสัญญาณ Back-EMF.

3.5.7 การวัดตำแหน่ง Rotor ด้วย Op-Amp และ Zero Crossing: การใช้อุปกรณ์ Op-Amp และตรวจจับ Zero Crossing ของสัญญาณเพื่อระบุตำแหน่งของ Rotor ใน BLDC Motor สำหรับการควบคุมแบบ Sensorless

3.5.8 6-Step Commutation: วิธีการสลับกระแสไฟฟ้าในขั้นตอนของการทดลองเพื่อศึกษาความสัมพันธ์ระหว่างความเร็วและแรงบิดของมอเตอร์รวมถึงวิเคราะห์สัญญาณด้วย Oscilloscop

3.5.9 การคำนวณ RPM จาก 6-Step Frequency

$$RPM = \frac{f \times 60}{Number\ of\ Pole\ Pairs}$$

โดย RPM คือ ความเร็วในการหมุนของ BLDC มีหน่วยเป็น Revolution per Minute (RPM)

f คือ ความถี่ มีหน่วยเป็น Hertz (Hz)

3.5.10 Torque Ripple: การสั่นสะเทือนของแรงบิดที่เกิดจากการควบคุมแบบ 6-Step สังเกตได้ในช่วงการเปลี่ยนเฟส (Commutation) ในมอเตอร์ BLDC ระหว่างการทดลอง

3.5.11 Hall Effect Sensor: เชznเซอร์ที่ตรวจจับตำแหน่ง Rotor โดยวัดสนามแม่เหล็ก ใช้ในการทดลองแบบ Sensor-Based เพื่อเปรียบเทียบความแม่นยำกับวิธี Sensorless

3.5.12 Sensorless Control: การควบคุมมอเตอร์โดยใช้ Back-EMF แทนการใช้เซนเซอร์ภายนอก ใช้ในการทดลองเพื่อลดความซับซ้อนและเปรียบเทียบประสิทธิภาพกับการควบคุมแบบ Sensor-Based.

3.5.13 Reference Speed: ความเร็วเป้าหมายที่ระบบควบคุมกำหนดให้มอเตอร์หมุน โดยใช้วัดความสัมพันธ์ระหว่างค่าสั่งควบคุมกับความเร็วจริงที่ได้ (Mechanical Speed) ในการทดลอง

3.5.14 Target Speed: ความเร็วที่กำหนดในโปรแกรม Motor Pilot เพื่อทดสอบความสัมพันธ์ระหว่างค่าสั่งควบคุมกับผลลัพธ์ที่เกิดขึ้นจริง

3.5.15 Mechanical Speed: ความเร็วรอบจริงของมอเตอร์ที่ตรวจวัดได้จากการทดลอง ใช้ในการวิเคราะห์ความแม่นยำของค่าสั่ง Target Speed และ Reference Speed.

3.5.16 Phase Shift: ความต่างเฟสของสัญญาณไฟฟ้าระหว่างเฟส U, V, W ในมอเตอร์ BLDC ใช้ในการวิเคราะห์การทำงานของการควบคุมแบบ 6-Step ผ่าน Oscilloscope

3.5.17 Trapezoidal Control: เทคนิคการควบคุมมอเตอร์ BLDC โดยใช้ Back-EMF ที่มีลักษณะ Trapezoidal ใช้ในการทดลองเพื่อวิเคราะห์ข้อดีและข้อเสียของการควบคุมแบบ 6-Step

3.5.18 Sinusoidal Control: เทคนิคการควบคุมมอเตอร์ BLDC ที่ใช้ Back-EMF แบบ Sinusoidal เพื่อการควบคุมที่ราบรื่น ใช้ในการเปรียบเทียบผลลัพธ์กับ Trapezoidal Control

3.5.19 Oscilloscope: อุปกรณ์ที่ใช้ตรวจสอบและวัดสัญญาณไฟฟ้าระหว่างการทดลอง เช่น Back-EMF, PWM และรูปแบบคลื่น 6-Step เพื่อวิเคราะห์ความสัมพันธ์ระหว่างสัญญาณกับพฤติกรรมมอเตอร์

3.5.20 Cheat Engine: Software ที่ใช้ในการอ่านและบันทึกค่าที่เก็บอยู่ในหน่วยความจำ (Memory Address) ของโปรแกรม Motor Pilot เพื่อดึงข้อมูลความเร็วรอบ (Mechanical Speed) ที่ไม่สามารถ Export ได้โดยตรงจากโปรแกรม Motor Pilot ใช้ Lua Script ในการเก็บข้อมูลและส่งออกในรูปแบบ Text File เพื่อการวิเคราะห์ในโปรแกรมอื่น เช่น Excel

3.6 เอกสารและงานวิจัยที่เกี่ยวข้อง

3.6.1 Datasheet

- 1) Low-Voltage BLDC Motor Driver expansion board based on STL220N6F7 for STM32

Nucleo

X-NUCLEO-IHM08M1 เป็นบอร์ดขยายสำหรับขั้บมอเตอร์กระแสตรงแบบไร้แปรงถ่าน (Brushless DC Motor) แบบ 3 เฟส ซึ่งใช้ STL220N6F7 STripFET™ F7 Power MOSFET เป็นส่วนประกอบหลักในการขับเคลื่อน ออกแบบมาให้ใช้งานร่วมกับ STM32 Nucleo โดยบอร์ดนี้ช่วยให้สามารถควบคุมมอเตอร์กระแสตรงแบบไร้แปรงถ่านได้ง่ายขึ้นใน STM32 Nucleo และรองรับการเชื่อมต่อเพิ่มเติมกับบอร์ดอื่น ๆ ผ่าน ST morpho connector รวมถึงสามารถติดตั้งตัวเชื่อมต่อแบบ Arduino™ UNO R3

IC Driver ที่ใช้ในบอร์ดนี้คือ L6398 ซึ่งเป็นชิปควบคุมแบบครึ่งสะพาน (Half-bridge Gate Driver) สำหรับ N-channel Power MOSFET โดยการผสานการทำงานระหว่าง L6398 Gate Driver และ STL220N6F7 Power MOSFET จะสร้างแพลตฟอร์มกำลังสูงสำหรับมอเตอร์ BLDC ในขณะที่ส่วนติดต่อทัลที่ควบคุมโดย STM32 Nucleo Board จะรองรับการควบคุมด้วยอัลกอริทึมแบบ 6-Step หรือ Field-Oriented Control (FOC) ซึ่งสามารถเลือกใช้งานได้ผ่าน Firmware

3.6.2 หลักการทำงานของ BLDC Motor (Brushless DC Motor)

BLDC Motor (Brushless DC Motor) ทำงานโดยอาศัย Lorentz Force Law ซึ่งเกี่ยวข้องกับ การติดต่อบรระหว่างสนามแม่เหล็กไฟฟ้าของส่วนประกอบต่างๆ เพื่อสร้างการเคลื่อนไหวหมุนของมอเตอร์ โดยขึ้นอยู่กับสองส่วนหลัก คือ Rotor และ Stator

- 1) Stator: เป็นส่วนที่อยู่นิ่งของมอเตอร์ ซึ่งประกอบด้วยชุดลวดแม่เหล็กไฟฟ้าที่เรียกว่า Stator Winding ซึ่งจัดเรียงในรูปแบบเฉพาะ ซึ่งเมื่อกระแสไฟฟ้าผ่านชุดลวดใน Stator จะเกิดสนามแม่เหล็กขึ้น สนามแม่เหล็กนี้จะหมุนไปตามทิศทางที่กำหนดโดยการสลับทิศทางของกระแสไฟฟ้าในชุดลวด ซึ่งจะทำให้ Rotor ซึ่งเป็นแม่เหล็กถาวรใน BLDC Motor หมุนไปตามทิศทางของสนามแม่เหล็ก
- 2) Rotor: เป็นส่วนที่หมุนของมอเตอร์ ซึ่งประกอบด้วยแม่เหล็กถาวรที่มีขั้วแม่เหล็กหนึบและได้จัดเรียงตามลำดับ เมื่อสนามแม่เหล็กจาก Stator โต้ตอบกับสนามแม่เหล็กจาก Rotor การติดต่อนี้จะทำให้ Rotor หมุนไปตามทิศทางของสนามแม่เหล็ก โดยการสลับทิศทางกระแสใน Stator จะช่วยทำให้ Rotor หมุนต่อเนื่อง
- 3) การควบคุมการหมุน: การควบคุม BLDC Motor จะขึ้นอยู่กับการจัดการกับกระแสไฟฟ้าที่ให้ไปยัง Stator Winding ซึ่งเป็นส่วนที่สร้างสนามแม่เหล็กที่ใช้ในการหมุน Rotor โดยที่กระแสไฟฟ้าที่ผ่านไปยัง Stator Winding ต้องมีการสลับทิศทางตามตำแหน่งของ Rotor ซึ่งการควบคุมนี้สามารถทำได้ทั้งในแบบ Sensor-Based และ Sensorless ขึ้นอยู่กับการใช้งานและลักษณะของระบบที่ต้องการ ซึ่ง Controller Board จะทำหน้าที่ควบคุมการสลับกระแสไฟฟ้าใน Stator Winding ตามลำดับที่เหมาะสม เพื่อสร้างสนามแม่เหล็กหมุนที่ใช้ในการหมุน Rotor การสลับกระแสนี้จะต้องทำในลำดับที่สอดคล้องกับการหมุนของ Rotor ซึ่งสามารถทำได้ผ่านการตรวจจับตำแหน่งของ Rotor จากเซนเซอร์ หรือการใช้ Back-EMF และความเร็วของ BLDC Motor จะถูกควบคุมโดยการปรับ PWM (Pulse Width Modulation) ที่ใช้ในการสลับกระแสไฟฟ้าใน Stator Winding ขณะที่แรงบิดจะถูกควบคุมโดย การปรับกระแสไฟฟ้าที่ให้ไปยัง Stator Winding

3.6.3 Sensor-Based vs Sensorless Motor Controllers

1) Sensor-Based Motor Controllers

ใช้เซนเซอร์ในการให้ข้อมูลย้อนกลับ (Feedback) เกี่ยวกับตำแหน่งและความเร็วของมอเตอร์ โดยเซนเซอร์ที่นิยมใช้หลัก ๆ คือ Encoder หรือ Hall Effect Sensor ทำหน้าที่ตรวจจับตำแหน่งของ Rotor และส่งข้อมูลดังกล่าวกลับไปยังตัวควบคุม (Controller)

ตารางที่ 3 แสดงการเปรียบเทียบเซนเซอร์ใน Sensor-Based Motor Controllers

อุปกรณ์	คำอธิบาย	ข้อดี	ข้อเสีย
Hall Effect Sensor	ตรวจจับสนามแม่เหล็ก	เรียบง่าย, ราคาถูก, เชื่อถือได้	ความแม่นยำต่ำกว่า ประเภทอื่น
Rotary Encoder	วัดการหมุนของแกน	แม่นยำ, มีความละเอียด สูง, ทำงานที่ความเร็วสูง	ราคาแพง, ซับซ้อน
Potentiometer	วัดตำแหน่งเชิงมุม	เรียบง่าย, ราคาถูก	ความแม่นยำและ ความละเอียดต่ำ
Resolver	วัดตำแหน่งเชิงมุมและ ความเร็ว	แม่นยำ, มีความละเอียด สูง, ทำงานที่ความเร็วสูง	ราคาแพง, ซับซ้อน
Synchro	วัดตำแหน่งเชิงมุม	แม่นยำ, มีความละเอียด สูง, ทำงานที่ความเร็วสูง	ราคาแพง, ซับซ้อน
Optical Encoder	วัดการหมุนของแกน	แม่นยำ, มีความละเอียด สูง, ราคาประหยัด	ทำงานที่ความเร็วสูง ไม่ได้เท่าที่ควร
Incremental Encoder	วัดจำนวนพัลส์ที่สร้างต่อ การหมุนหนึ่งรอบ	แม่นยำ, มีความละเอียด สูง, ราคาประหยัด	ทำงานที่ความเร็วสูง ไม่ได้เท่าตัว Absolute Encoder
Absolute Encoder	วัดตำแหน่งเชิงสัมบูรณ์ ของแกน	แม่นยำ, มีความละเอียด สูง, ทำงานที่ความเร็วสูง	ราคาแพง, ซับซ้อน

1) Sensorless Motor Controllers

แตกต่างจากระบบควบคุมมอเตอร์แบบ Sensor-Based และระบบควบคุมมอเตอร์แบบ Sensorless ทำงานโดยไม่ต้องใช้เซนเซอร์ภายนอกในการให้ข้อมูลย้อนกลับเกี่ยวกับตำแหน่งหรือความเร็วของมอเตอร์ แต่จะอาศัยแรงดันไฟฟ้า Back-EMF หรือคุณสมบัติภายในอิเล็กทรอนิกส์ ของมอเตอร์ในการประมาณตำแหน่งของ Rotor และควบคุมการทำงานของมอเตอร์

สามารถทำการทดลองได้ด้วย การควบคุมการสลับกระแส (Commutation Control) โดยจะวัดกระแสของเฟสของมอเตอร์อย่างต่อเนื่องและวัดแรงดันไฟฟ้าจ่ายเป็นระยะ จากนั้นอุปกรณ์จะใช้ข้อมูลเหล่านี้ในการคำนวณค่า Back-EMF และตำแหน่งของมอเตอร์ ความเร็วของมอเตอร์ถูกกำหนดจากจำนวนครั้งที่ Back-EMF ของเฟสหนึ่งตัดผ่านจุดศูนย์ (Zero Crossing) ในช่วงเวลาหนึ่ง

ตารางที่ 4 แสดงการเปรียบเทียบเทคนิคใน Sensorless Motor Controllers

เทคนิค	คำอธิบาย	ข้อดี	ข้อเสีย
Back-EMF (BEMF)	ใช้ Back-EMF ที่มอเตอร์สร้างขึ้นเพื่อประมาณตำแหน่งของ Rotor	เรียบง่าย, ติดตั้งได้ง่ายไม่ต้องใช้เซนเซอร์ภายนอก	ความแม่นยำลดลงเมื่อมอเตอร์ทำงานที่ความเร็วต่ำ/แรงบิดสูง
Space Vector Modulation (SVM)	ใช้อัลกอริทึม SVM ในการประมาณตำแหน่ง Rotor จากกระแสและแรงดันไฟฟ้าของมอเตอร์	แม่นยักษ์กว่า Back-EMF ในช่วงความเร็วต่ำและแรงบิดสูง	ต้องการฮาร์ดแวร์และซอฟต์แวร์ที่ซับซ้อนกว่า
Kalman Filter	ใช้ฟิลเตอร์ Kalman ในการประมาณตำแหน่ง Rotor จากกระแสและแรงดันไฟฟ้าของมอเตอร์	แม่นยักษ์กว่า SVM ในช่วงความเร็วต่ำและแรงบิดสูง	ต้องการฮาร์ดแวร์และซอฟต์แวร์ที่ซับซ้อนกว่า
Direct Torque Control (DTC)	ใช้อัลกอริทึม DTC เพื่อควบคุมแรงบิดของมอเตอร์โดยตรง โดยไม่ต้องประมาณตำแหน่งของ Rotor	ตอบสนองรวดเร็วมาก เหมาะสำหรับการใช้งานที่ต้องการประสิทธิภาพสูง	ต้องการฮาร์ดแวร์และซอฟต์แวร์ที่ซับซ้อนกว่า

2) เปรียบเทียบข้อดี-ข้อเสียของ Sensor-Based และ Sensorless Motor Controllers

ตารางที่ 5 แสดงความแตกต่างระหว่าง Sensor-Based และ Sensorless Controllers

คุณสมบัติ	Sensor-Based	Sensorless
ความซับซ้อนของระบบ	ซับซ้อนมากกว่า เพราะต้องติดตั้งเซนเซอร์และเดินสายเพิ่มเติม แต่เทคโนโลยีช่วยลดความซับซ้อนของการออกแบบ	ออกแบบง่ายกว่า มีส่วนประกอบน้อยกว่าและไม่ต้องเดินสายสำหรับเซนเซอร์ บำรุงรักษาง่าย
ต้นทุน	ราคาสูงกว่า เนื่องจากต้องใช้เซนเซอร์ต้นทุนขึ้นอยู่กับชนิดและคุณภาพของเซนเซอร์	ต้นทุนต่ำกว่า เนื่องจากไม่ต้องใช้เซนเซอร์ภายนอก หมายความสำหรับงานที่ต้องการลดค่าใช้จ่าย
ความแม่นยำ	มีประสิทธิภาพและความแม่นยำสูง ควบคุมได้ในช่วงการทำงานที่กว้าง เหมาะสมกับงานที่ต้องการความแม่นยำสูง	ประสิทธิภาพน่าพอใจสำหรับงานทั่วไป แต่มีความแม่นยำน้อยกว่าในความเร็วต่อ หรือการเปลี่ยนแปลงโหลดอย่างรวดเร็ว
การเริ่มต้นการทำงาน	การเริ่มต้นทำงานราบรื่นและควบคุมได้แม่นยำ แม้ในความเร็วต่ำ	มีข้อจำกัดในการเริ่มต้นทำงานและควบคุมที่ความเร็วต่ำ เนื่องจากต้องพึ่งพาการคำนวณจาก Back-EMF
การควบคุมความเร็วและแรงบิด	ควบคุมความเร็วและแรงบิดได้แม่นยำ ในช่วงการทำงานกว้าง หมายความสำหรับงานที่ต้องการการควบคุมที่แม่นยำ เช่น CNC	ควบคุมความเร็วและแรงบิดได้ในระดับน่าพอใจ แต่มีข้อจำกัดในความเร็วต่อและ การเปลี่ยนแปลงโหลดอย่างรวดเร็ว
ทนทานต่อสิ่งแวดล้อม	เซนเซอร์ไวต่อสิ่งแวดล้อม เช่น ฝุ่นและความชื้น	ทนต่อสภาพแวดล้อมได้ดี เช่น อุณหภูมิและความชื้นสูง
การนำไปใช้งาน	หมายความสำหรับงานที่ต้องการความแม่นยำสูง เช่น หุ่นยนต์ เครื่องจักร CNC ยานยนต์ไฟฟ้าประสิทธิภาพสูง	หมายความสำหรับงานที่เน้นต้นทุนต่ำและความเรียบง่าย เช่น อุปกรณ์อิเล็กทรอนิกส์ ระบบอัตโนมัติในบ้าน ยานยนต์ทั่วไป

3.6.4 การจัดการ BLDC Motor Feedbacks

1) Sensorless Control Feedback

ในการควบคุมแบบ Sensorless ของ BLDC Motor คณะผู้จัดทำจะใช้ Back-EMF ที่เกิดจาก การหมุนของ Rotor แทนการใช้เซนเซอร์ภายนอก เช่น Hall Sensors เพื่อรับตำแหน่งของ Rotor Back-EMF ที่เกิดจากการหมุนของ Rotor สามารถบ่งชี้ตำแหน่งของ Rotor ได้โดยไม่ต้องใช้เซนเซอร์ภายนอก. การใช้ Zero Crossing ช่วยในการตรวจสอบตำแหน่งของ Rotor เนื่องจากเมื่อ Back-EMF ถึง จุด Zero Crossing จะเป็นจุดที่สามารถบ่งชี้ตำแหน่งที่สำคัญของ Rotor ในแต่ละรอบการหมุน

ในระบบควบคุม Sensorless, Op-Amp (Operational Amplifier) จะถูกใช้เพื่อขยาย สัญญาณ Back-EMF ที่ได้รับจากมอเตอร์ เพื่อให้สามารถตรวจจับ Zero Crossing ได้อย่างแม่นยำ. โดย Op-Amp จะช่วยในการเสริมความชัดเจนของสัญญาณ Back-EMF เพื่อให้สามารถตรวจจับจุดที่ สัญญาณข้ามศูนย์และใช้ข้อมูลนี้ในการคำนวณตำแหน่งของ Rotor การขยายสัญญาณนี้จะช่วยให้ controller สามารถสับเปลี่ยนไฟฟ้าใน Stator Winding ตามลำดับที่ถูกต้องและทำให้ Rotor หมุน ได้อย่างแม่นยำและมีประสิทธิภาพ

การใช้ Zero Crossing และ Op-Amp ในระบบควบคุมแบบ Sensorless นี้ช่วยให้ระบบ สามารถทำงานได้โดยไม่ต้องพึ่งพาเซนเซอร์ภายนอก เช่น Hall Sensors และสามารถลดค่าใช้จ่ายและ ความซับซ้อนในระบบได้ในขณะเดียวกัน

2) Sensor-Based Control Feedback

ในการควบคุมแบบ Sensor-Based ของ BLDC Motor, Hall Effect Sensors จะถูกติดตั้งในมอเตอร์เพื่อให้ข้อมูลตำแหน่งของ Rotor โดยตรง. Hall Sensors สามารถตรวจจับการเปลี่ยนแปลงของสนามแม่เหล็กเมื่อ Rotor หมุน และส่งสัญญาณออกมายังรูปแบบดิจิตอลที่บ่งชี้ตำแหน่งของ Rotor. ความละเอียดของข้อมูลตำแหน่งจะขึ้นอยู่กับจำนวนของ Hall Sensors ที่ใช้งาน ซึ่งในมอเตอร์ 3-Phase นักจะใช้ 3 Hall Sensors ซึ่งจะส่งสัญญาณออกมายังรูปแบบที่สามารถระบุตำแหน่งของ Rotor ใน 6 ช่วง (หรือ 6 ขั้นตอนการหมุน)

ข้อมูลจาก Hall Effect Sensors จะถูกส่งไปยัง ESC (Electronic Speed Controller) หรือ MCU (Microcontroller Unit) เพื่อประมวลผล ESC หรือ MCU จะใช้ข้อมูลนี้ในการควบคุมกระแสไฟฟ้าที่ถูกจ่ายให้กับ Stator Winding ตามลำดับที่ถูกต้องเพื่อให้ Rotor หมุนในทิศทางที่ต้องการ

ในการควบคุมนี้ Op-Amp (Operational Amplifier) จะถูกใช้ในการแปลงสัญญาณจาก Hall Sensors ให้เป็นสัญญาณที่เหมาะสมสำหรับการประมวลผลใน ESC หรือ MCU Op-Amp จะทำหน้าที่ขยายหรือปรับรูปแบบของสัญญาณจาก Hall Sensors ให้มีความเสถียรและเหมาะสมกับการนำไปใช้ในการควบคุม Stator Winding โดยการทำงานร่วมกันของ Hall Sensors, Op-Amp, และ ESC/MCU ทำให้สามารถควบคุมการหมุนของ BLDC Motor ได้อย่างแม่นยำและมีประสิทธิภาพ

3.6.5 มอเตอร์ที่ใช้การควบคุม FOC รวมถึงมอเตอร์ที่ควบคุมด้วยวิธี Sinusoidal มักถูกเรียกว่า Permanent Magnet Synchronous Motors (PMSM)

3.6.6 BLDC Motor Control Techniques

ด้วยเทคโนโลยีมอเตอร์แบบ Brushless Motor และมอเตอร์ Permanent Magnet Synchronous (PMSM) มีคุณสมบัติที่ไม่มีประภารถ่านเหมือนกัน ทั้งสองเป็นมอเตอร์แบบซิงโครนัส และมีความแตกต่างกันเพียงเล็กน้อย ความแตกต่างนั้นอยู่ที่รูปแบบของ Back-EMF โดยมอเตอร์ BLDC มี Back-EMF แบบ Trapezoidal ในขณะที่ PMSM มี Back-EMF แบบ Sinusoidal แต่ในความเป็นจริง มอเตอร์ไม่สามารถสร้าง Back-EMF แบบ Trapezoidal ได้อย่างสมบูรณ์แบบ แต่มักมีลักษณะใกล้เคียงกับ Sinusoidal ดังนั้นในบางครั้ง ประเภทของมอเตอร์อาจไม่สำคัญเท่ากับวิธีการควบคุม (Commutation Method) ซึ่งมีความสำคัญมากกว่า

1) Trapezoidal

เป็นวิธีการสลับกระแสในมอเตอร์ที่พบบ่อยที่สุด เพราะเป็นเทคนิคที่ง่ายที่สุด ในแต่ละขั้นตอน จะมีการจ่ายกระแสไฟฟ้าให้กับชุดลวด 2 เฟส ในขณะที่ชุดลวดอีกเฟสหนึ่งจะถูกตั้งไว้ในสถานะ "ลอย" (Float) สำหรับมอเตอร์ที่ใช้การควบคุมแบบ Trapezoidal การสลับกระแสในชุดลวด 3 เฟสต้องเกิดขึ้นตามลำดับที่กำหนดไว้อย่างแม่นยำเพื่อให้มอเตอร์ BLDC ทำงานได้อย่างมีประสิทธิภาพ ลำดับการสลับกระแสแสดงด้วยตำแหน่งสัมพัทธ์ของคู่แม่เหล็กในRotorและชุดลวดของ Stator มอเตอร์ BLDC แบบ 3 เฟสต้องการลำดับการสลับกระแส 6 ขั้นตอนเพื่อให้ครบหนึ่งรอบไฟฟ้า ความสัมพันธ์ระหว่างการหมุนทางกลและรอบไฟฟ้าขึ้นอยู่กับจำนวนคู่แม่เหล็กที่ติดอยู่กับRotor

Trapezoidal เป็นวิธีที่อาศัยรูปคลื่นแรงเคลื่อนไฟฟ้าด้านหลัง (Back-EMF) ที่มีลักษณะเป็น Trapezoidal และจ่ายกระแสให้ตรงกับลักษณะ Back-EMF ดังกล่าว ซึ่งการควบคุมนี้มักเรียกว่า “Six-Step Commutation” เนื่องจากมีกระแสขับเคลื่อน 6 ขั้นตอนที่ใช้ในการหมุน Rotor ครบหนึ่งรอบ ในแต่ละขั้นตอนมีเพียง 2 เฟสที่ถูกจ่ายตู้นในขณะที่เฟสที่สามลอย (Float) ทำให้ลดการสูญเสียพลังงานจากการสลับกระแส (Switching Loss) อัลกอริทึมของวิธีนี้ง่ายและเหมาะสมสำหรับการใช้งานที่ไม่ต้องการความแม่นยำสูง แต่จะเกิด Torque Ripple ขึ้นในมอเตอร์ทุกครั้งที่มีการสลับกระแส (ทุก ๆ 60 องศา)

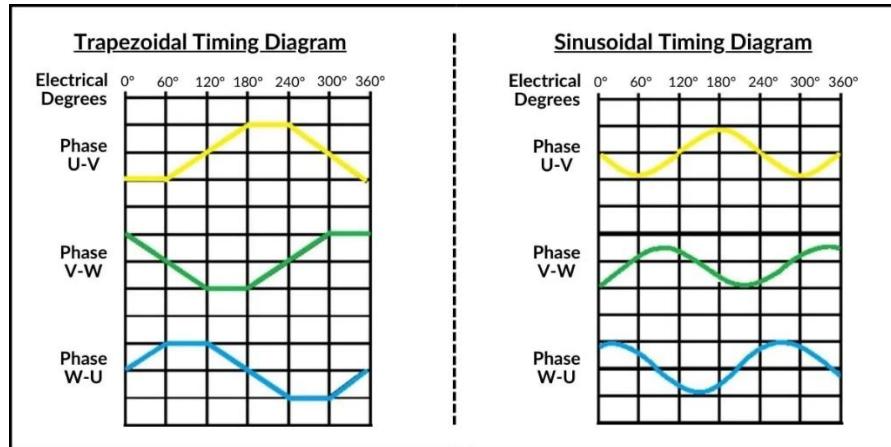
2) Sinusoidal

การควบคุมแบบ Sinusoidal จะจ่ายกระแสไฟฟ้าให้กับขดลวดของ Stator ทั้ง 3 เพส โดยกระแสที่ใช้จะมีลักษณะเป็นคลื่น Sinusoidal ซึ่งไฟฟ้าของกระแสแต่ละเฟสจะมีมุมต่างกัน 120° การจ่ายกระแสแบบนี้ช่วยให้การส่งแรงบิดมีความราบรื่นมากขึ้น ซึ่งไม่สามารถทำได้ด้วยการควบคุมแบบ Trapezoidal แต่แม้ว่าจะมีประสิทธิภาพมากในความเร็วต่ำ เมื่อมอเตอร์หมุนที่ความเร็วสูงสุด ประสิทธิภาพของวิธีนี้มักลดลง เนื่องจากตัวควบคุมกระแส (Current Loop Controllers) จำเป็นต้องติดตามสัญญาณ Sinusoidal ที่มีความถี่สูงขึ้นเรื่อยๆ ซึ่งต้องใช้การประมวลผลที่ซับซ้อนมากขึ้น แต่หากต้องการให้ได้การควบคุมที่แม่นยำและมีประสิทธิภาพที่ความเร็วต่ำ ควรเลือกใช้งานควบคู่กับเซนเซอร์อย่าง Hall Effect Sensor, Encoder อุปกรณ์ที่คล้ายกันจึงแทนการควบคุมแบบ Sinusoidal มุ่งเน้นที่การขัด Torque Ripple และการทำงานที่ราบรื่นมากขึ้น โดยใช้รูปคลื่น Back-EMF และกระแสขับเคลื่อนในลักษณะ Sinusoidal ซึ่งกระแสในแต่ละเฟสจะถูกเลื่อนเฟสออกจากกัน 120° วิธีนี้ใช้ได้ดีกับมอเตอร์ PMSM เนื่องจาก Back-EMF มีรูปร่างเป็น Sinusoidal ตามธรรมชาติ ในขณะที่มอเตอร์ BLDC ที่มี Back-EMF เป็น Trapezoidal ในทางทฤษฎี แต่อย่างไรก็ตาม BLDC ก็สามารถใช้งานวิธีนี้ได้ เช่นกัน เพราะความหน่วงนำของมอเตอร์ทำให้ Back-EMF เรียบขึ้นจนมีลักษณะใกล้เคียงกับ Sinusoidal ได้ การควบคุมแบบ Sinusoidal ช่วยเพิ่มประสิทธิภาพ ลดการสั่นสะเทือนและเสียงรบกวน แต่มีข้อเสียคือ การควบคุมที่ความเร็วสูงมีความซับซ้อนมากขึ้น เนื่องจากตัวควบคุมต้องติดตามกระแสที่เปลี่ยนแปลงตามความถี่ที่เพิ่มขึ้น และเกิด Switching Loss ที่สูงกว่า Trapezoidal Control นอกจากนี้ วิธีนี้ต้องการการตรวจสอบตำแหน่ง Rotor ที่แม่นยำ เช่น Encoder หรือ Resolver เพื่อให้ทำงานได้อย่างเต็มประสิทธิภาพ

3) Field Oriented Control (FOC)

การควบคุมแบบ Field Oriented Control หรือ FOC เป็นวิธีการควบคุมที่อาศัยการวัดและปรับกระแสในขดลวดของ Stator (Stator Currents) เพื่อให้มุ่งระหว่างสนามแม่เหล็กของ Rotor และ Stator (Rotor and Stator Magnetic Field) คงที่ที่ 90° ตลอดเวลา ซึ่งตำแหน่งนี้เป็นจุดที่ให้แรงบิดสูงสุด (Maximum Torque Output) และมีประสิทธิภาพสูงสุด โดยในความเร็วสูง FOC มีประสิทธิภาพมากกว่าการควบคุมแบบ Sinusoidal และมีความสามารถในการตอบสนองต่อโหลดที่เปลี่ยนแปลงอย่างรวดเร็วได้ดีกว่าวิธีการควบคุมอื่น ๆ นอกจากนี้ FOC ยังช่วยลด Torque Ripple ให้เหลือเกือบศูนย์ ทำให้การควบคุมมอเตอร์มีความราบรื่นและแม่นยำ ทั้งในช่วงความเร็วต่ำและความเร็วสูง โดยมอเตอร์ที่ใช้การควบคุม FOC มักเป็น PMSM

อย่างไรก็ตาม FOC สามารถใช้งานในรูปแบบ Sensorless ได้ โดยคำนวณตำแหน่งของ Rotor จากแรงดัน Back-EMF แต่ ณ จังหวะการเริ่มต้นการทำงานมักตรวจสอบแรงดัน Back-EMF ได้ยาก ทำให้ต้องพึงพาวิธีการควบคุมแบบ Open Loop ในช่วงเริ่มต้นจนกว่า Back-EMF จะเพียงพอสำหรับการตรวจจับตำแหน่งโดยไมโครคอนโทรลเลอร์



รูปที่ 72 แสดงการเปรียบเทียบระหว่าง Trapezoidal กับ Sinusoidal

ตารางที่ 6 แสดงการเปรียบเทียบ Control Techniques ระหว่าง Trapezoidal และ Sinusoidal

Parameters	Trapezoidal	Sinusoidal	คำอธิบาย
Switching Loss	ต่ำ	สูง	ใน Trapezoidal จะมีกระແສไฟฟ้าเข้าเพียง 2 เฟส ในแต่ละครั้ง ในขณะที่ Sinusoidal ใช้กระແສไฟฟ้าเข้า 3 เฟสพร้อมกัน ซึ่งเพิ่มการสูญเสียพลังงาน
Implementation Complexity	ต่ำ	สูง	การออกแบบให้กระແສไฟฟ้าเข้า 2 เฟสพร้อมกัน ในแบบ Trapezoidal ง่ายกว่าการออกแบบให้กระແສไฟฟ้าเข้า 3 เฟสพร้อมกันในแบบ Sinusoidal
Torque Ripple at Low Speed	สูง	ต่ำ	เนื่องจากการกระตุนเพียง 2 เฟสในแต่ละครั้งในแบบ Trapezoidal ทำให้แรงบิดตกลงเล็กน้อย ในช่วงเปลี่ยนเฟสและเกิด Torque Ripple สูง
Acoustic & Electrical Noise	สูง	ต่ำ	การควบคุมแบบ Trapezoidal ส่งผลให้เกิด Torque Ripple สูงขึ้น ซึ่งทำให้เกิดเสียงและสัญญาณรบกวนทางไฟฟ้ามากกว่า
Rotor Position Feedback	Hall Sensor	Encoder / Resolver	Trapezoidal ใช้ Hall Sensor ตรวจจับตำแหน่ง Rotor ขณะที่แบบ Sinusoidal ใช้ Encoder หรือ Resolver เพื่อตรวจจับตำแหน่งอย่างต่อเนื่อง

3.6.7 BLDC Motor Feedback

1) Sensorless Control Feedback

ในการควบคุมแบบ Sensorless คณะผู้จัดทำจะใช้สัญญาณ Back-EMF (Electromotive Force) ที่เกิดจากการหมุนของ Rotor ในมอเตอร์แทนการใช้เซ็นเซอร์ภายในอกเพื่อระบุตำแหน่งของ Rotor การใช้ Back-EMF นั้นจะช่วยให้คณะผู้จัดทำทราบตำแหน่งของ Rotor โดยไม่ต้องพึ่งพา Hall Sensor หรือเซ็นเซอร์ภายในอกอื่น ๆ โดยการใช้ Zero Crossing จะช่วยให้คณะผู้จัดทำทราบตำแหน่งของ Rotor ในแต่ละช่วงเวลา และใช้ Op-Amp ในการขยายสัญญาณ Back-EMF ที่ได้รับจากมอเตอร์ และใช้สัญญาณนี้ในการหาตำแหน่งของ Rotor

2) Sensor-Based Control Feedback

ในการควบคุมแบบ Sensor-Based จะมี Hall Effect Sensors ที่ติดตั้งอยู่ในมอเตอร์เพื่อให้ข้อมูลตำแหน่งของ Rotor ความละเอียดของข้อมูลตำแหน่งนี้จะขึ้นอยู่กับจำนวนของ Hall Sensor ที่ใช้งาน (มักจะมี 3 เซ็นเซอร์ในมอเตอร์ 3-Phase) ข้อมูลจาก Hall Sensor จะถูกส่งไปยัง ESC หรือ MCU เพื่อประมวลผลและใช้ในการควบคุมกระแสไฟฟ้าให้มอเตอร์หมุนในทิศทางที่ต้องการ และใช้ Op-Amp ในการแปลงสัญญาณจาก Hall Effect Sensors ให้เป็นสัญญาณที่เหมาะสมสำหรับการประมวลผลใน ESC หรือ MCU

3.6.8 ความสัมพันธ์ระหว่าง PWM และ Frequency Output

1) PWM ใน Sensor-Based

เมื่อ Hall Sensors ตรวจจับตำแหน่งของ Rotor ในแต่ละเฟสจากสนามแม่เหล็กที่เกิดจากแม่เหล็กภายใน Rotor ซึ่งข้อมูลจาก Hall Sensors จะถูกใช้ในการควบคุมการสลับเฟส (Commutation) ของมอเตอร์ให้เหมาะสมตามตำแหน่งของ Rotor ที่ตรวจจับได้

การสลับเฟสนี้ทำให้มอเตอร์สามารถหมุนได้อย่างราบรื่นและมีประสิทธิภาพ ในขณะเดียวกัน PWM จะถูกใช้ในการควบคุมความเร็วของมอเตอร์ โดยการปรับค่า Duty Cycle ของสัญญาณ PWM ซึ่งส่งผลให้มอเตอร์ได้รับพลังงานมากหรือน้อยตามที่กำหนด ซึ่งทำให้ความเร็วในการหมุนของมอเตอร์เปลี่ยนแปลงได้ตามต้องการ การใช้ Hall Effect ร่วมกับ PWM มีข้อดีที่ทำให้สามารถควบคุมมอเตอร์ได้อย่างแม่นยำ โดยไม่จำเป็นต้องใช้ Encoder ซึ่งลดค่าใช้จ่ายและความซับซ้อนในการออกแบบระบบ ในขณะเดียวกันก็ช่วยให้การควบคุมมอเตอร์สามารถทำได้ง่ายและมีประสิทธิภาพในระดับหนึ่ง แต่ข้อจำกัดของวิธีนี้คือความละเอียดในการควบคุมตำแหน่งของ Rotor ที่ต่ำกว่าเมื่อเทียบกับการใช้ Encoder เนื่องจาก Hall Sensors จะให้ข้อมูลตำแหน่งในจุดที่ค่อนข้างห่างกัน จึงอาจไม่เหมาะสมกับการใช้งานที่ต้องการความละเอียดสูง นอกจากนี้ หากใช้ PWM ความถี่สูง อาจมีปัญหาเก็บสัญญาณรบกวนจาก PWM ซึ่งอาจทำให้การตรวจจับสัญญาณจาก Hall Sensor ไม่แม่นยำ การกรองสัญญาณและการเลือกความถี่ PWM ที่เหมาะสมจึงเป็นสิ่งสำคัญในการออกแบบนี้เพื่อให้สามารถควบคุมมอเตอร์ได้อย่างแม่นยำและมีประสิทธิภาพในทุกช่วงความเร็วและการใช้งาน

2) PWM ใน Sensorless

Back-EMF เป็นแรงดันที่เกิดขึ้นในมอเตอร์ขณะหมุน จะใช้เป็นตัวบ่งชี้เพื่อให้ระบบควบคุมมอเตอร์สามารถปรับการทำงานให้เหมาะสมกับความเร็วที่ต้องการ ในขณะที่ PWM จะใช้ในการควบคุมการจ่ายพลังงานให้กับมอเตอร์ โดยการปรับค่า Duty Cycle ของ PWM ซึ่งส่งผลให้มอเตอร์ได้รับพลังงานมากหรือน้อยตามที่กำหนดและความเร็วในการหมุนของมอเตอร์ก็จะเปลี่ยนแปลงตามการปรับค่าเหล่านั้น

การใช้ Back-EMF ร่วมกับ PWM ช่วยให้สามารถควบคุมความเร็วได้อย่างแม่นยำโดยไม่จำเป็นต้องใช้เซนเซอร์เพิ่มเติม ในขณะเดียวกัน Back-EMF ยังสามารถใช้ในการควบคุมตำแหน่งของ Rotor ได้ แต่การใช้งานจริงอาจประสบปัญหาจากการตรวจจับ Back-EMF ในความเร็วต่ำเนื่องจาก Back-EMF ที่เกิดขึ้นจะมีขนาดเล็กมาก ทำให้การตรวจจับเป็นเรื่องยาก นอกจากนี้การใช้ PWM ความถี่สูงอาจทำให้เกิดสัญญาณรบกวนหรือ PWM Artifacts ที่ส่งผลกระทบต่อการตรวจจับ Zero-Crossing ของ Back-EMF จึงต้องมีการกรองสัญญาณเพื่อให้การตรวจจับสามารถทำได้แม่นยำและไม่เกิดความล่าช้าในการควบคุม การเลือกความถี่ PWM ที่เหมาะสมจึงเป็นสิ่งสำคัญเพื่อลดผลกระทบจากการรบกวนจากสัญญาณเหล่านี้ การกรองสัญญาณที่ดีและการซิงโครไนซ์ระหว่างการสลับเฟสและการตรวจจับ Zero-Crossing เป็นสิ่งจำเป็นในการใช้งาน Back-EMF PWM เพื่อให้สามารถควบคุมมอเตอร์ได้อย่างมีประสิทธิภาพ และแม่นยำ ทั้งนี้ระบบนี้มีข้อดีที่สามารถควบคุมมอเตอร์ได้โดยไม่ต้องใช้ Encoder และลดค่าใช้จ่าย แต่ก็มีข้อจำกัดในเรื่องของการตรวจจับในความเร็วต่ำและการซิงโครไนซ์ระหว่างการสลับเฟสและการตรวจจับ Back-EMF

ตารางที่ 7 แสดงความแตกต่างระหว่างการควบคุมด้วย PWM, Back-EMF และ Hall Sensor

ปัจจัย	PWM	Back-EMF	Hall Sensor
การควบคุมความเร็ว	ควบคุมความเร็วโดยการปรับ Duty Cycle ของ PWM	ควบคุมความเร็วจาก Back-EMF ที่ขึ้นอยู่กับความเร็วของมอเตอร์	ไม่เกี่ยวข้องโดยตรงกับความเร็ว
ความสัมพันธ์กับความถี่	มีผลต่อความละเอียดของ การควบคุมพลังงานและความเร็ว	ความถี่ไม่เกี่ยวข้อง	เกี่ยวข้องกับตำแหน่งของ Rotorและการสลับเฟส
การควบคุมตำแหน่ง	ใช้ PWM ในการควบคุมการจ่ายพลังงานที่แตกต่างกันเพื่อเปลี่ยนตำแหน่ง	Back-EMF สามารถใช้ในการคำนวณตำแหน่งได้จากการตรวจจับ Zero-Crossing	ใช้ Hall Sensor ในการตรวจจับตำแหน่งของ Rotorและการสลับเฟส
การตรวจจับสัญญาณรบกวน (Noise)	สัญญาณ PWM อาจรบกวน Back-EMF จึงต้องการกรอง	Back-EMF อาจได้รับผลกระทบจาก noise ของ PWM แต่จะมีการกรองหรือใช้การตรวจจับตามเวลา	สัญญาณ Hall อาจได้รับผลกระทบจาก noise ของ PWM แต่จะมีการกรองหรือใช้การตรวจจับตามเวลา
ความแม่นยำในการควบคุม	ขึ้นอยู่กับความถี่ PWM และการเลือก Duty Cycle	แม่นยำในการควบคุมเมื่อความเร็วต่ำเนื่องจาก Back-EMF มีขนาดเล็ก	มีความแม่นยำสูงในการควบคุมตำแหน่งของ Rotor โดยไม่ต้องคำนึงถึงความเร็ว
ข้อจำกัด	ใช้ความถี่สูงสามารถก่อให้เกิดสัญญาณรบกวน และ Artifacts	ควบคุมยากในความเร็วต่ำ และมีปัญหา กับสัญญาณรบกวนจาก PWM	อาจต้องใช้วงจรช่วย เช่น การกรองสัญญาณ หรือซอฟต์แวร์เสริม

3.7 วิธีดำเนินการทดลอง

ทดสอบ BLDC Motor รุ่น A2212/13T 1000KV โดยเชื่อมต่อกับ X-NUCLEO-IHM08M1 ซึ่งทำหน้าที่ขยายและแปลงสัญญาณทั้งขาเข้าและขาออก โดยขาเข้ารับสัญญาณจาก Nucleo STM32G474RE และขาออกส่งสัญญาณไปยังมอเตอร์ BLDC จากนั้นตั้งค่าผ่านโปรแกรม MotorControl Workbench เพื่อกำหนด Input/Output Configuration (IOC) ให้เหมาะสมสำหรับการใช้งานร่วมกับ X-NUCLEO-IHM08M1 เมื่อกำหนดค่าเรียบร้อยแล้วจึงอปป์โหลดการตั้งค่าเข้าสู่ Nucleo STM32G474RE หลังจากนั้นใช้โปรแกรม Motor Pilot เพื่อควบคุมการทำงานและติดตามความเร็วการหมุนของมอเตอร์ BLDC พร้อมทั้งทำการทดลองเก็บค่าสัญญาณที่ได้จาก X-NUCLEO-IHM08M1 ในการตั้งค่าความเร็วและทิศทางการหมุนที่แตกต่างกัน เพื่อเปรียบเทียบความสัมพันธ์ระหว่างความเร็วและสัญญาณที่อุปกรณ์ X-NUCLEO-IHM08M1 สุดท้ายนำข้อมูลที่ได้มาวิเคราะห์ในโปรแกรม Excel เพื่อคำนวณค่าความคลาดเคลื่อนและตรวจสอบว่าคำสั่งที่ส่งจากโปรแกรม Motor Pilot มีความสัมพันธ์กับสัญญาณที่วัดได้และความเร็วของมอเตอร์อย่างมีนัยสำคัญหรือไม่

3.8 วัสดุอุปกรณ์

- 3.8.1 BLDC Motor จำนวน 1 อัน
- 3.8.2 STMICROELECTRONICS X-NUCLEO-IHM08M1 จำนวน 1 อัน
- 3.8.3 Nucleo STM32G474RE 1 อัน
- 3.8.4 พرو์มสายอปป์โหลด จำนวน 1 เส้น
- 3.8.5 BLDCXplorer จำนวน 1 ชุด (ฐานสามารถบรรจุบอร์ดควบคุม, Breadboard, 3D-Print ใช้สำหรับการประกอบกับ BLDC Motor)

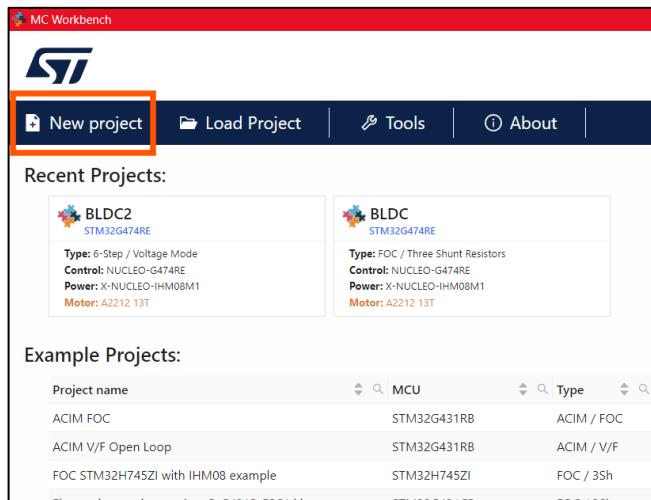
3.9 ขั้นตอนการดำเนินงาน

3.9.1 การเชื่อมต่อระหว่างบอร์ดและอุปกรณ์ต่าง ๆ

- 1) นำบอร์ด Nucleo STM32G474RE เชื่อมต่อเข้ากับ BLDCXplorer ในช่องที่ถูกวางไว้
- 2) นำบอร์ด X-NUCLEO-IHM08M1 เชื่อมต่อเข้ากับ Nucleo STM32G474RE
- 3) ตรวจสอบการเชื่อมต่อระหว่างสัญญาณขาออกของ X-NUCLEO-IHM08M1 และ BLDC Motor
- 4) ตรวจสอบแรงดันไฟเลี้ยงที่เข้ามาเลี้ยงบอร์ด X-NUCLEO-IHM08M1 และ Nucleo STM32G474RE ให้คงที่อยู่ที่ 12V และ 5V ตามลำดับ

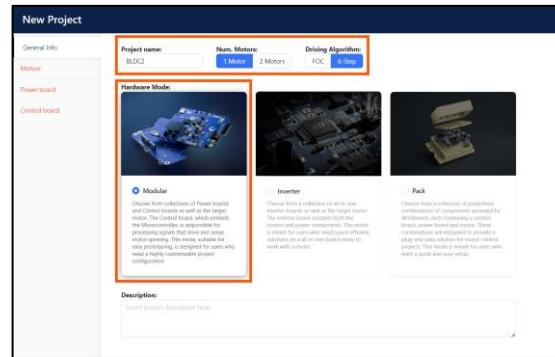
3.9.2 การตั้งค่าโปรแกรม MotorControl Workbench

- 1) สร้างไฟล์ใหม่เพื่อใช้สำหรับการควบคุม BLDC Motor A2212/13T 1000KV ผ่านการกดปุ่ม New project ในหน้าหลักของโปรแกรม



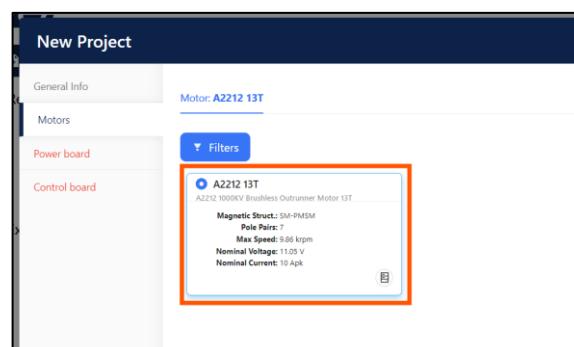
รูปที่ 73 แสดงถึงปุ่ม New project ในหน้าหลักของโปรแกรม MotorControl Workbench

- 2) ตั้งค่า Num. Motors., Driving Algorithm และ Hardware Module เป็น 1 Motor, 6-Step และ Modular ตามลำดับอ้างอิงจาก Output ที่คณะผู้จัดทำต้องการให้กับ BLDC Motor ที่เป็นแบบ Sensorless จำนวนหนึ่งตัว จานวนนึงจึงทำการตั้งค่าสำหรับไฟล์นี้



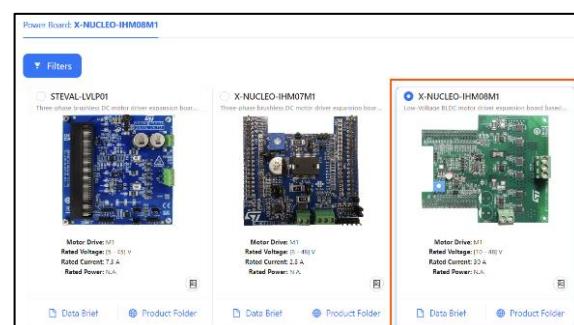
รูปที่ 74 แสดงการตั้งค่าในหน้า General Info ของโปรแกรม MotorControl Workbench

- 3) กำหนดค่ามอเตอร์ให้ตรงกับ BLDC Motor ที่มีชื่อในนี่คือ A2212/13T 1000KV



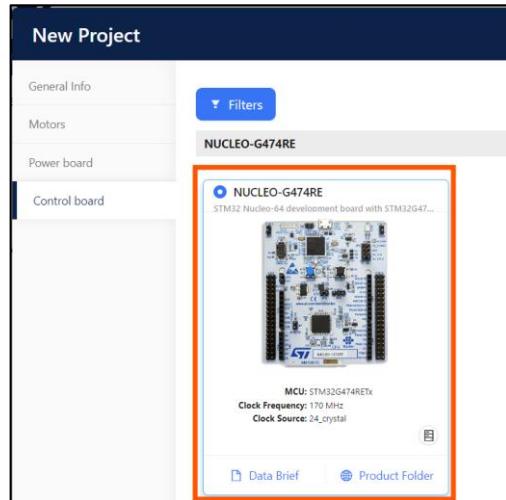
รูปที่ 75 แสดงถึงการกำหนดค่าในหน้า Motors ของโปรแกรม MotorControl Workbench

- 4) กำหนด Power board ให้ตรงกับอุปกรณ์ที่มีชื่อในนี่คือ X-NUCLEO-IHM08M1



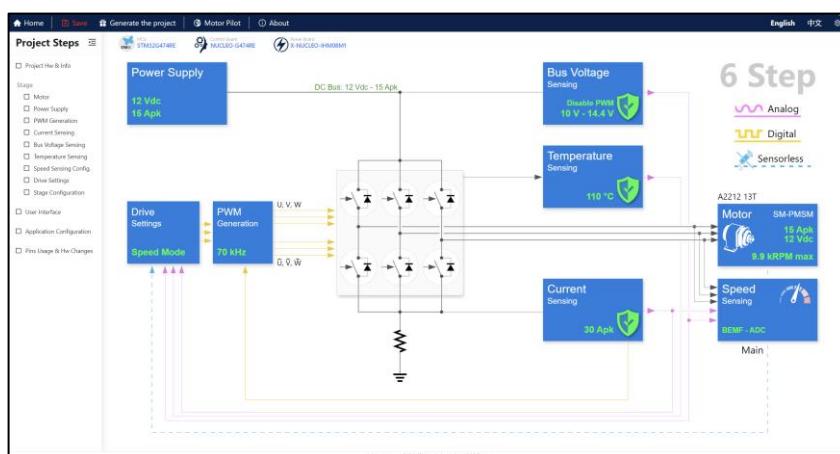
รูปที่ 76 แสดงถึงการกำหนดค่าในหน้า Power board ของโปรแกรม MotorControl Workbench

5) กำหนด Control board ให้ตรงกับอุปกรณ์ที่มีซึ่งในที่นี่คือ Nucleo-G474RE



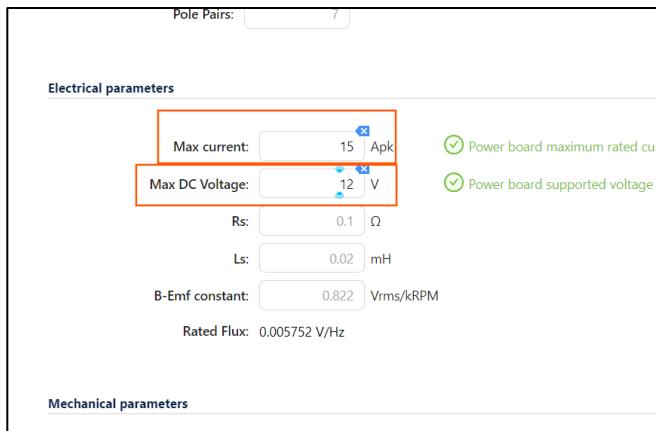
รูปที่ 77 แสดงถึงการกำหนดค่าในหน้า Control board ของโปรแกรม MotorControl Workbench

6) หลังจากการกำหนดบอร์ดให้ตรงกับอุปกรณ์ทั้งหมดที่คณผู้จัดทำมีแล้วโปรแกรมจะนำมาสู่หน้า
การตั้งค่าภายในโดยรวมอีกรอบหนึ่ง



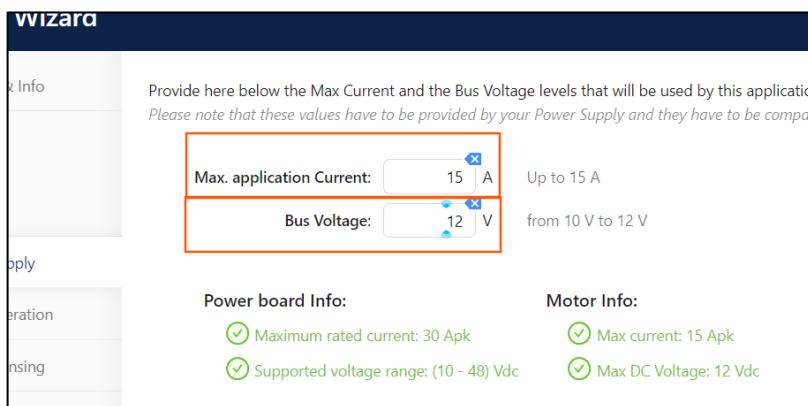
รูปที่ 78 แสดงถึงหน้าการตั้งค่าภายในของโปรแกรม MotorControl Workbench

ซึ่งผู้ใช้สามารถที่จะปรับการตั้งค่าเพื่อให้สามารถใช้งานได้กับมอเตอร์และบอร์ดอย่าง
ละเอียดอีกรอบหนึ่ง ซึ่งในขั้นตอนนี้คณผู้จัดทำจะปรับแรงดันและกระแสให้อยู่ที่ 12V และ 15A
(Peak) ตามลำดับ



รูปที่ 79 แสดงถึงการตั้งค่าแรงดันและกระแสสูงสุดภายในโปรแกรม MotorControl Workbench

- 7) หลังจากการตั้งค่าความสามารถของมอเตอร์แล้วคณะผู้จัดทำจึงมาปรับความสามารถของแหล่งจ่ายเพื่ออ้างอิงให้ตรงกับมอเตอร์ซึ่งในที่นี้คือการปรับกระแสสูงสุดและการปรับ Bus Voltage ให้อยู่ที่ 15A และ 12V ตามลำดับ



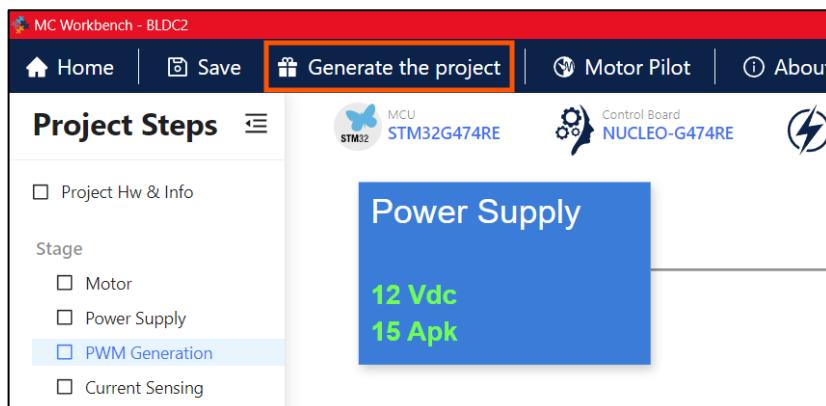
รูปที่ 80 แสดงถึงการตั้งค่าความสามารถของมอเตอร์ในโปรแกรม MotorControl Workbench

- 8) ภายหลังจึงทำการตั้งค่าความถี่ของการสร้างสัญญาณ PWM ให้อยู่ที่ 70000 Hz เพื่อความลื่นไหลของการควบคุม



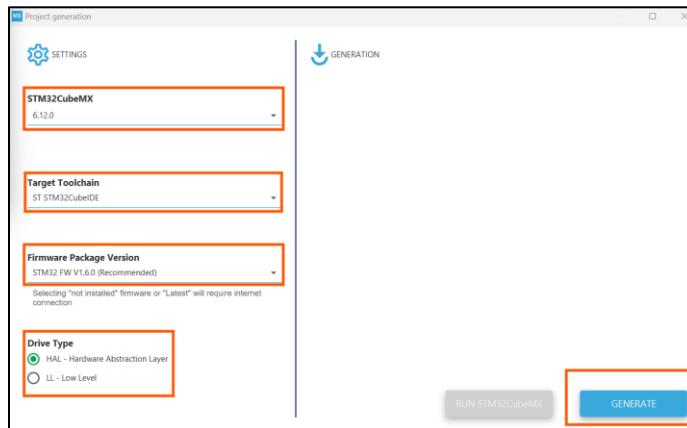
รูปที่ 81 แสดงถึงการตั้งค่าความถี่การสร้างสัญญาณ PWM ในโปรแกรม MotorControl Workbench

- 9) การสร้างคำสั่งสำหรับการควบคุมมอเตอร์และบอร์ด X-NUCLEO-IHM08M1 นั้นสามารถทำได้ด้วยการกดปุ่ม Generate the project จากนั้นจึงทำการตั้งค่าสภาพการสร้างทั้งเวอร์ชันและระบบที่จะใช้ส่งออกโค้ดให้เหมาะสมกับบอร์ดของคณะผู้จัดทำโดยจะตั้ง STM32CubeIDE เป็นเวอร์ชัน 6.12.0 ตั้ง Target Toolchain เป็น ST STM32CubeIDE ตั้ง Firmware Package Version เป็น STM32 FW V1.6.0 และตั้ง Drive Type เป็น HAL – Hardware abstraction layer

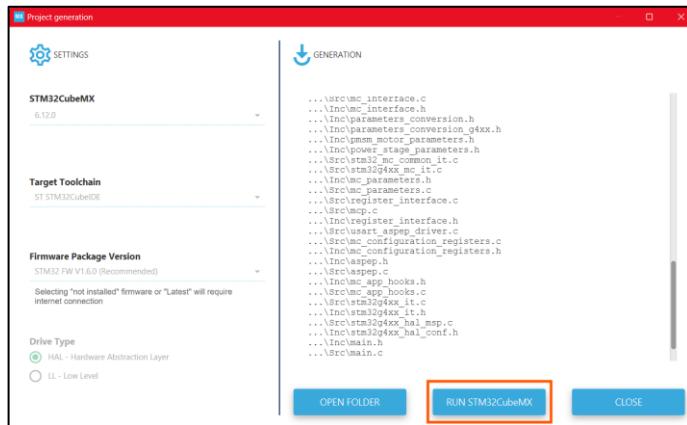


รูปที่ 82 แสดงถึงขั้นตอนการสร้างโค้ดจากโปรแกรม MotorControl Workbench

10) จากนั้นจึงกดปุ่ม Generate เพื่อทำการสร้าง Input/Output Configuration (IOC)

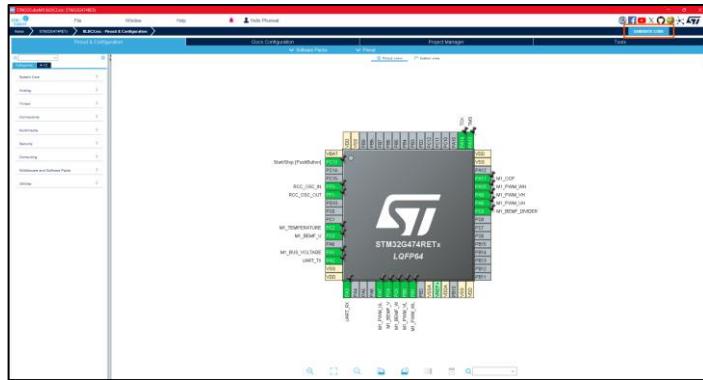


รูปที่ 83 แสดงถึงขั้นตอนการสร้างโค้ดจากโปรแกรม MotorControl Workbench

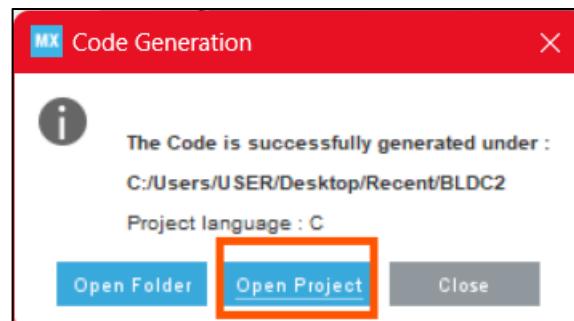


รูปที่ 84 แสดงถึงขั้นตอนการสร้าง IOC ในโปรแกรม MotorControl Workbench

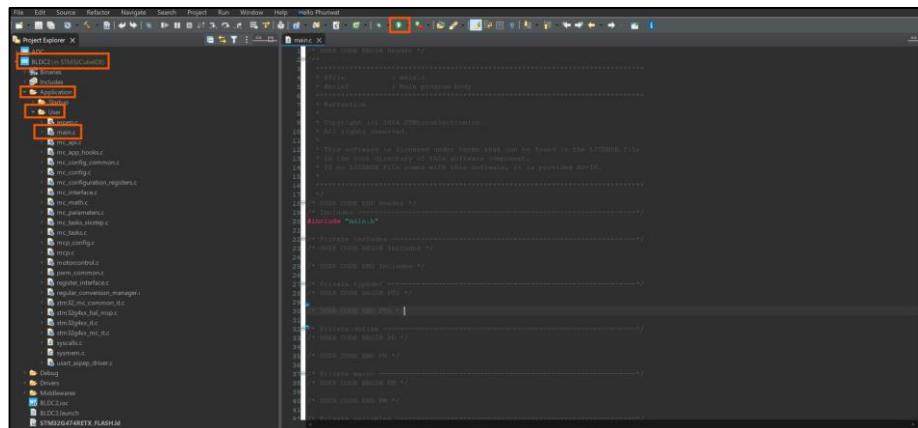
ภายหลังจากการสร้าง IOC ในโปรแกรม MotorControl Workbench เสร็จจะทำการกดปุ่ม RUN STM32CubeIDE ในโปรแกรม MotorControl Workbench เพื่อที่จะทำการเปิดหน้าต่าง UI สำหรับการตั้งค่า IOC และสร้างโค้ดในลำดับถัดมา จากการกดปุ่ม GENERATE CODE ตัวโปรแกรมจะทำการสร้างโค้ดอ้างอิงจาก IOC ที่ถูกต้องไว้ในโปรแกรม STM32CubeMX จากนั้นจึงทำการกดเปิดโค้ดที่สร้างขึ้นมาผ่านโปรแกรม STM32CubeIDE และหาเส้นทางไฟล์เพื่อเข้าถึงไฟล์ main.c และทำการอัปโหลดไฟล์นั้นเข้าสู่ STM32G474RE ผ่านการกดปุ่ม RUN ในขณะที่เปิดหน้า main.c ซึ่งสามารถหาไฟล์ main.c ได้จากการเข้า (Project Name) -> Application -> User -> main.c



รูปที่ 85 แสดงถึงหน้าการตั้งค่า IOC เพื่อที่จะสร้างโค้ดด้วยการกดปุ่ม GENERATE CODE



รูปที่ 86 แสดงถึงหน้าต่างหลังจาก Code Generation ในโปรแกรม STM32CubeMX

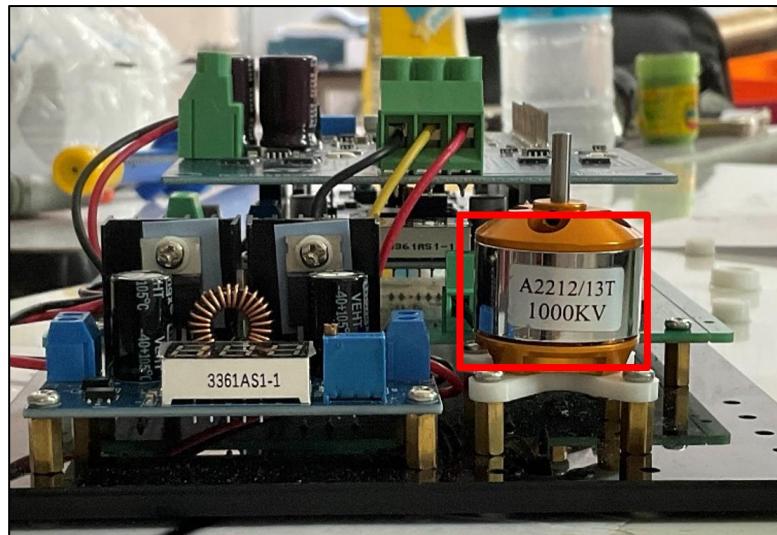


รูปที่ 87 แสดงถึงหน้าต่างโค้ดที่ถูกสร้างขึ้นมาในหน้าต่างของโปรแกรม STM32CubeIDE

3.9.3 การวัดและบันทึกข้อมูลความเร็วของ Motor Pilot

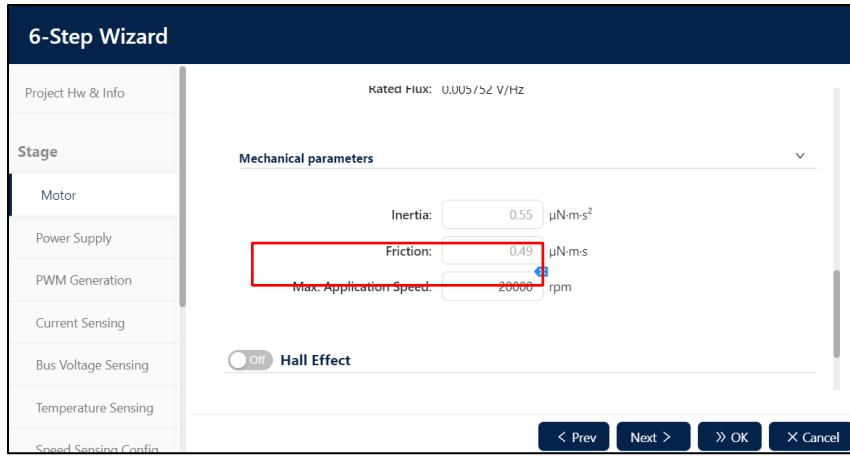
เมื่อตั้งค่าและจัดการกับการเริ่มต้นการทดลองได้เสร็จสิ้น ลำดับถัดไปคณะผู้จัดทำจึงได้เริ่มขั้นแรกในการสังเกตค่า Mechanical Speed ที่เที่ยงไป-มา มาจากโปรแกรม Motor Pilot นั่นคือการทดลองเปรียบเทียบค่า Target Speed (ความเร็วของที่ต้องการสั่ง), Reference Speed (ความเร็วของที่สั่งได้จริง) กับ Mechanical Speed (ความเร็วของจริงที่โปรแกรมอ่านได้แบบไม่คงที่)

- 1) การตรวจสอบ Max Speed ที่ BLDC Motor สามารถทำได้



รูปที่ 88 แสดงถึงชิ้นส่วน BLDC Motor ที่ถูกใช้งานบนบอร์ด MotorXplorer

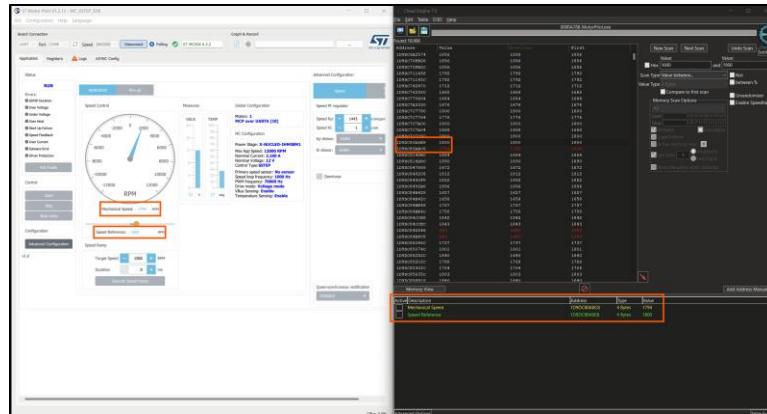
เนื่องจาก BLDC Motor สามารถให้ความเร็วได้ 1000KV มีความหมายว่ามอเตอร์สามารถให้ความเร็วได้ 1000 RPM ต่อ 1V นั่นแปลว่า หากคณะผู้จัดทำทำการทดลองด้วยการจ่ายแรงดันคงที่อยู่ที่ 12V จะทำให้มอเตอร์สามารถหมุนด้วยความเร็วสูงสุดอยู่ที่ 12000 RPM นั่นเอง แต่เพื่อตรวจสอบว่า BLDC Motor ตัวนี้สามารถหมุนได้ 12000 RPM จริงหรือไม่ คณะผู้จัดทำจึงตัดสินใจปรับการตั้งค่ามอเตอร์ในส่วนของ Max. Application Speed ในโปรแกรม MotorControl WorkBench 6.3.2 ให้เกินค่าสูงสุดที่คาดว่ามอเตอร์จะทำได้ โดยตัดสินใจใส่ให้เป็นค่าสูง ๆ ไม่เพื่อตรวจสอบค่าสูงสุดที่ BLDC Motor จะสามารถทำได้จริง ๆ คณะผู้จัดทำจึงได้กำหนดค่า Max. Application Speed ให้อยู่ที่ 20000 RPM



รูปที่ 89 แสดงการตั้งค่า Max. Application Speed ใน MotorControl WorkBench

2) การได้มาของค่า Mechanical Speed

ในการสั่งงาน BLDC Motor คงจะต้องทำให้ใช้โปรแกรม Motor Pilot เพื่อกำหนดค่าความเร็ว rob หรือ Target Speed ที่ต้องการ อย่างใกล้เคียง ในการทดลองจริง พบว่ามอเตอร์ BLDC ไม่สามารถหมุนได้ตรงกับค่าความเร็วที่ตั้งไว้ เนื่องจากปัจจัยหลายประการ เช่น ลักษณะการโหลด และพฤติกรรมเฉพาะของมอเตอร์ ทำให้ค่า Reference Speed ที่สั่งได้จริง ๆ มีความแตกต่างจากค่าที่กำหนด และเมื่อสั่งให้มอเตอร์หมุนด้วยความเร็วที่เท่ากับ Reference Speed แล้วนั้น โปรแกรมจึงสามารถคำนวณหา Mechanical Speed ได้ แต่เนื่องจากการที่จะได้มาซึ่งข้อมูลตัวเลขของค่า Mechanical Speed ที่เหวี่ยงไปมานั้น ไม่สามารถได้มาจากการ Export ค่าความเร็ว ทำให้คงจะต้องใช้โปรแกรมเสริมชื่อ Cheat Engine ในการอ่านตัวเลขผ่านจากการอ่าน Memory Address Value ของโปรแกรม Motor Pilot ที่ถูกเก็บไว้ใน Register Memory Address ทำให้สามารถสร้างกราฟพิสูจน์เทียบระหว่างกราฟในโปรแกรม Motor Pilot และค่าที่ได้จากการอ่าน Memory Address Value ของโปรแกรม Motor Pilot ได้ผ่านโปรแกรม Cheat Engine โดยเก็บค่าโดยใช้ Lua Script ในการดึงข้อมูลออกมาเป็นไฟล์ Text ซึ่งจะส่งการหมุนอ้างอิงจาก Reference Speed ตั้งแต่ 0 RPM ถึง 7998 RPM เพื่อทดสอบความตรงกันของข้อมูล ซึ่งจะใช้เวลาในการเก็บ 60 วินาทีและความถี่ในการเก็บอยู่ที่ 100 ms ต่อ 1 ค่าเทียบเท่ากับ 600 ค่าต่อ 60 วินาที



รูปที่ 90 แสดงถึงโปรแกรม Motor Pilot และ Cheat Engine ในขณะอ่านค่าความเร็วรอบ

```

Lua script: Cheat Table
File View
1 local address = 0x109DC80A6E0 -- Replace with the memory address
2 local file = io.open("output.txt", "w")
3 local TimeDomain = io.open("TimeDomain.txt", "a")
4 local ValueDomain = io.open("ValueDomain.txt", "a")
5 [[ If not file or not TimeDomain or not ValueDomain then .. ]]
6 [[ local logInterval = 1000 -- Log interval in milliseconds ]]
7 [[ local maxDuration = 60000 -- Maximum duration in milliseconds (60 seconds) ]]
8 [[ local timeStep = 0 -- Counter for time domain as integers ]]
9 [[ local elapsedTime = 0 -- Tracks elapsed time in milliseconds ]]
10 [[ local isRunning = false -- Logging status ]]
11 [[ function start() .. ]]
12 [[     if not isRunning then .. ]]
13 [[         isRunning = true ]]
14 [[         if elapsedTime > maxDuration then .. ]]
15 [[             stop() ]]
16 [[             print("Logging completed: 60 seconds.") ]]
17 [[             return ]]
18 [[         end ]]
19 [[         local value = readInteger(address) -- Or use readFloat/readByte depending on the type ]]
20 [[         if value then .. ]]
21 [[             file:write(os.date("%Y-%m-%d %H:%M:%S") .. ": " .. tostring(value) .. "\n") ]]
22 [[             file:flush() .. logInterval ]]
23 [[             TimeDomain:write(timeStep .. "\n") .. logInterval ]]
24 [[             ValueDomain:write(tostring(value) .. "\n") .. logInterval ]]
25 [[             TimeDomain:flush() .. logInterval ]]
26 [[             ValueDomain:flush() .. logInterval ]]
27 [[             timeStep = timeStep + 1 .. logInterval ]]
28 [[             elapsedTime = elapsedTime + logInterval ]]
29 [[         else .. ]]
30 [[             print("Error: Failed to read value at the address.") ]]
31 [[         end ]]
32 [[     end ]]
33 [[ end ]]
34 [[ local timer = createTimer(nil, false) ]]
35 [[ timer.Interval = logInterval ]]
36 [[ timer.OnTimer = log ]]
37 [[ function start() --start logging for 60 seconds .. ]]
38 [[ function stop() --stop logging manually .. ]]
39 [[ function clean() --clean the table and close files .. ]]
40 [[ function clear() --stop, clear, and close files .. ]]
41 [[ function close() --close the files .. ]]
42 ]]

```

รูปที่ 91 แสดง Lua Script ที่ใช้บันทึกความเร็วจาก Memory Address Value จาก Cheat Engine

3) การหาความเร็วรอบที่แท้จริง

เมื่อได้ค่าความเร็วรอบที่ได้จากการใช้โปรแกรม Cheat Engine มาแล้ว ลำดับถัดไปคุณผู้จัดทำจึงทดลองใส่ค่า Target Speed ตั้งแต่ 1000–13000 RPM โดยเพิ่มค่าความเร็วทุก ๆ 1000 RPM ซึ่งระยะห่างนี้ถูกกำหนดขึ้นเพื่อให้สามารถเห็นแนวโน้มการเปลี่ยนแปลงของค่าความเร็วได้อย่างชัดเจน และทำให้ผลลัพธ์สองค่าล้องกับข้อมูลเชิงตัวเลขที่เกิดขึ้นในแต่ละช่วงการวัดได้มากขึ้น โดยมีจุดประสงค์เพื่อตรวจสอบว่า Reference Speed ที่แท้จริงในแต่ละช่วงคือเท่าไร และวิจัยความสามารถนำค่าที่ได้มาหาความสัมพันธ์ระหว่างค่า Mechanical Speed ที่ผ่านการหา RMS แล้ว

3.9.4 การวัดและบันทึกข้อมูลจากการควบคุมมอเตอร์แบบ 6-Step

หลังจากที่คณผู้จัดทำได้สังเกตพฤติกรรมการหมุนของมอเตอร์ในระดับความเร็วที่ตั้งไว้ พบว่า ความเร็วที่ได้มีความสัมพันธ์โดยตรงกับการควบคุมความเร็วทั้ง PWM และ Reference Speed อย่างเรียบ ตาม เพื่อให้สามารถเข้าใจรายละเอียดของการควบคุมมอเตอร์แบบ 6-Step ซึ่งเป็นกระบวนการสำคัญที่ใช้ในการควบคุมการหมุนของ BLDC Motor ได้อย่างลึกซึ้ง

1) การอ่านค่าที่ได้จากสัญญาณผ่าน Oscilloscope

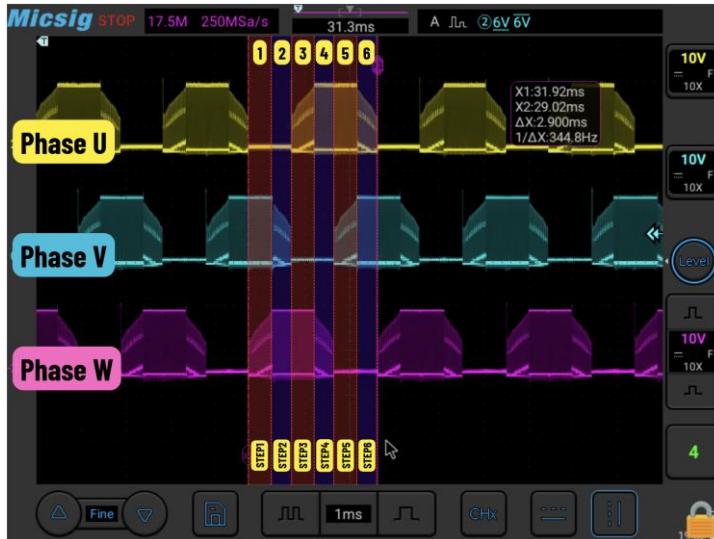
คณผู้จัดทำดำเนินการทดลองเพิ่มเติมโดยการสังเกตพฤติกรรมของสัญญาณไฟฟ้าที่เกิดขึ้นในกระบวนการควบคุมผ่าน Oscilloscope โดยการใช้ Oscilloscope เพื่อตรวจสอบ ลำดับการทำงานของ 6-Step, ลักษณะของคลื่นไฟฟ้าที่เกิดขึ้นในแต่ละเฟสของมอเตอร์, และ ผลกระทบที่เกิดจากการสลับเฟส(Commutation) ต่อความเร็วและแรงบิดของมอเตอร์ในช่วงเวลาต่างๆ



รูปที่ 92 - 93 แสดงตัวอย่างสัญญาณ 6-Step ของ BLDC ที่ Target Speed 10000 และ 2000 RPM

ตามลำดับ

เมื่อสามารถแยกวิเคราะห์ข้อมูลได้แล้ว คณะผู้จัดทำจึงดำเนินการเก็บค่าความถี่ของ 6-Step โดยใช้ Cursor บน Oscilloscope เพื่อวัดความกว้างของคลาบสัญญาณ (Period)



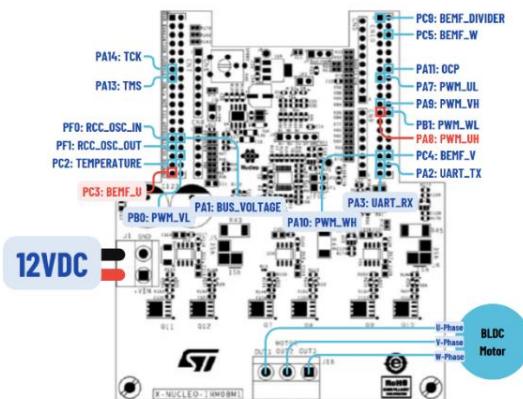
รูปที่ 94 แสดงถึงกราฟและช่วงสัญญาณแรงดันขาออกที่วัดได้จาก X-NUCLEO-IHM08M1 ซึ่งนำไปใช้หาคลาบเวลาของสัญญาณรูปแบบ 6-Step

โดยที่เริ่มวัดตั้งแต่การเริ่มการเปิด PWM ของเฟส V (กราฟเส้นสีฟ้า)ที่ได้จากการทำงานของมอเตอร์ จนนั้นนำค่าดังกล่าวไปคำนวณหาความเร็วรอบของมอเตอร์ผ่านสูตร คำนวณหาจาก 6-Step Frequency ที่แสดงความสัมพันธ์ระหว่างความถี่และความเร็ว

ในการเก็บข้อมูลจากหัวข้อที่ 3.9.3 ทำให้คณะผู้จัดทำทราบถึงช่วง Reference Speed ที่แท้จริงว่าควรเริ่มต้นการเก็บค่าที่ 996-12350 RPM และปรับ Target Speed ให้เพิ่มขึ้นทุก ๆ 1000 RPM (แต่ในความเป็นจริง Reference Speed จะเพิ่มขึ้นได้ใกล้เคียง Target Speed เท่านั้น) เพื่อให้คณะผู้จัดทำสามารถวิเคราะห์ความเร็วที่จะคำนวณได้จาก 6-Step มาแสดงความสัมพันธ์ได้ง่าย

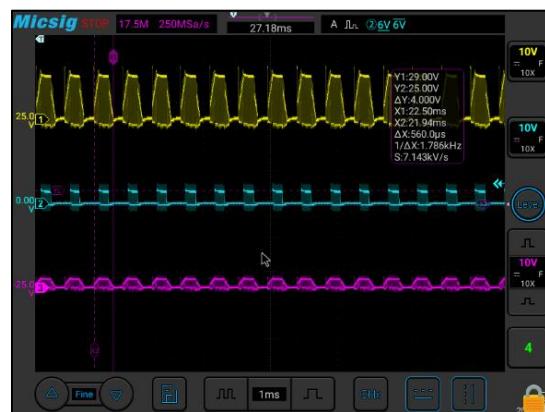
2) วิเคราะห์สัญญาณที่อยู่ภายในช่วง 1 คาบของการควบคุมแบบ 6-Step

เมื่อสังเกตสัญญาณในช่วง 1 คาบของการควบคุม 6-Step คงผู้จัดทำสามารถแยกองค์ประกอบของสัญญาณที่เกิดขึ้นได้อย่างชัดเจน ซึ่งประกอบไปด้วย สัญญาณ Back-EMF และสัญญาณ PWM ที่ใช้ควบคุมกระแสในแต่ละเฟสของมอเตอร์ BLDC ซึ่งการรวมกันของ Back-EMF และ PWM ส่งผลให้สามารถวิเคราะห์ลักษณะของสัญญาณในแต่ละช่วงการหมุนของมอเตอร์ได้ โดยสัญญาณที่ได้จาก Oscilloscope จะถูกนำมาวิเคราะห์ในเชิงลึก เพื่อทำความเข้าใจลำดับการทำงานของ 6-Step, ผลกระทบของการสลับเฟส และพัฒนาระบบการทำงานของมอเตอร์ในแต่ละช่วงเวลา



รูปที่ 95 แสดงถึงช่องสัญญาณสำหรับใช้ในการตรวจจับและซองที่ถูกกรับส่งค่าของบอร์ด X-NUCLEO-

IHM08M1



รูปที่ 96 แสดงตัวอย่างการแยกสัญญาณ 6-Step ของ BLDC Motor ที่ Target Speed 10000 RPM

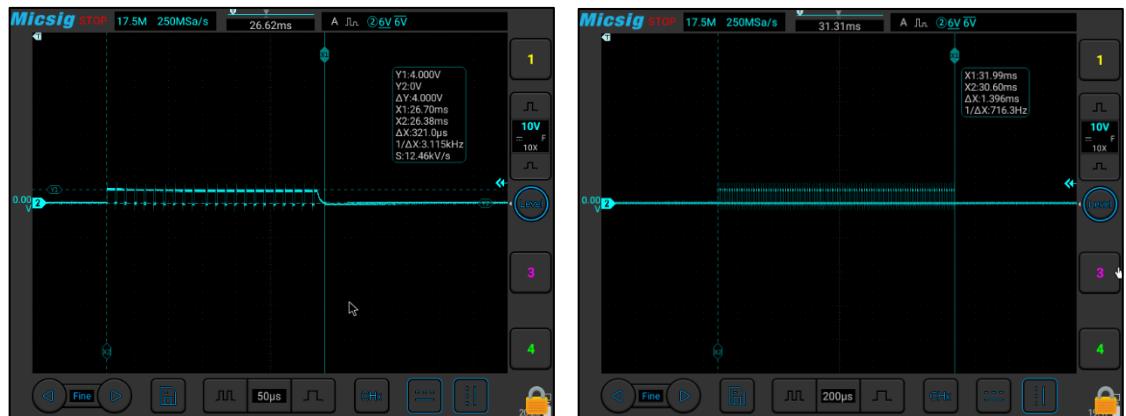


รูปที่ 97 แสดงตัวอย่างการแยกสัญญาณ 6-Step ของ BLDC Motor ที่ Target Speed 2000 RPM

2.1) วิเคราะห์สัญญาณ PWM ใน 6-Step

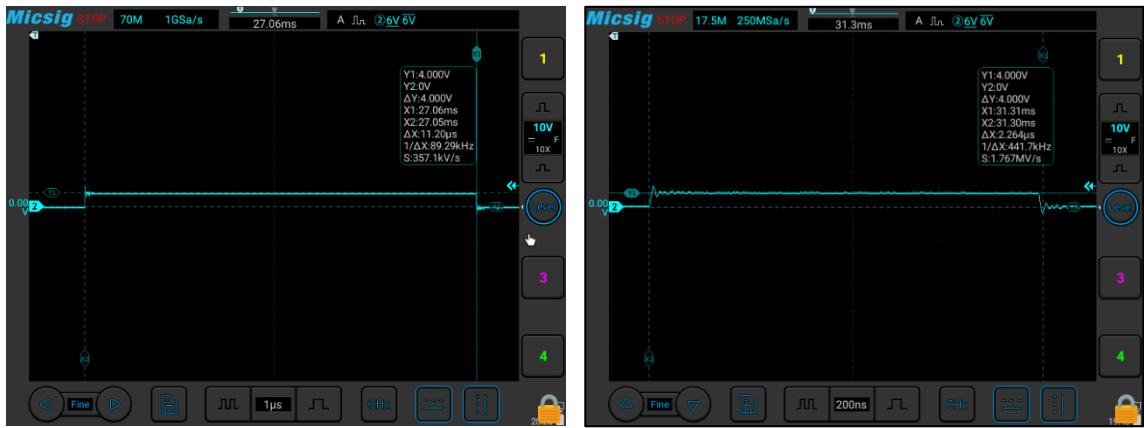
เมื่อสังเกตภายใต้สัญญาณ 6-Step จะพบสัญญาณ PWM ซึ่งถูกใช้เพื่อปรับปริมาณพลังงานที่จำเพื่อให้มอเตอร์ในแต่ละเฟส โดยการปรับค่า Duty Cycle (อัตราส่วนระหว่างช่วงสัญญาณ ON ต่อ OFF) เพื่อให้มอเตอร์ได้รับแรงดันที่เหมาะสม ซึ่งมากไปกว่าหนึ่น ในสัญญาณ PWM ที่ได้มาจะถูกประกอบไปด้วยสัญญาณ PWM ย่อย ๆ อีก ตั้ง

รูป



รูปที่ 98 - 99 แสดงตัวอย่าง PWM 6-Step ของ BLDC Motor ที่ความเร็ว 12000 และ 2000 RPM

ตามลำดับ

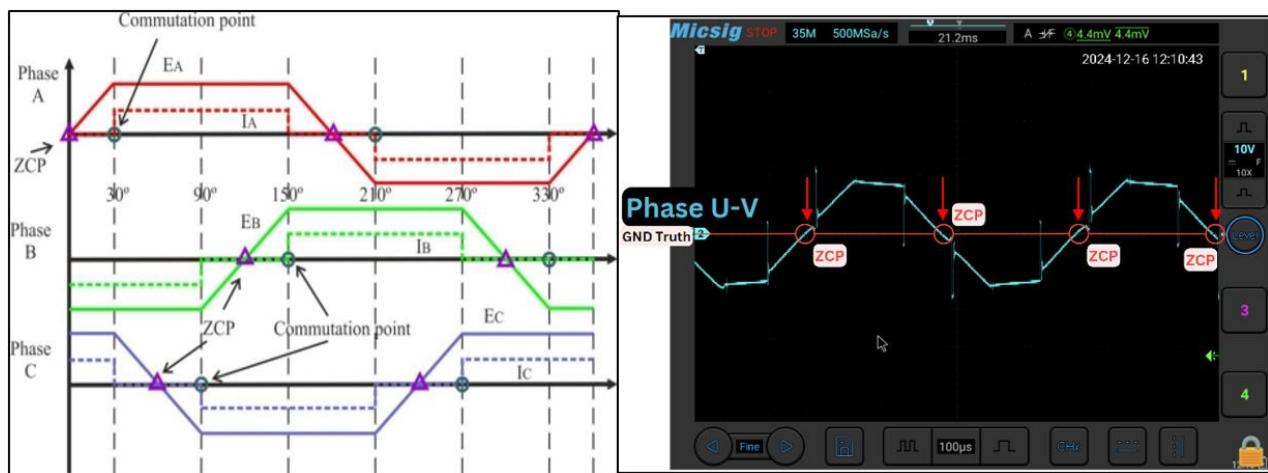


รูปที่ 100 - 101 แสดงตัวอย่าง PWM ย่อยใน PWM 6-Step ของ BLDC Motor ที่ความเร็ว 12000 และ 2000 RPM ตามลำดับ

2.2) วิเคราะห์สัญญาณ Back-EMF ใน 6-Step

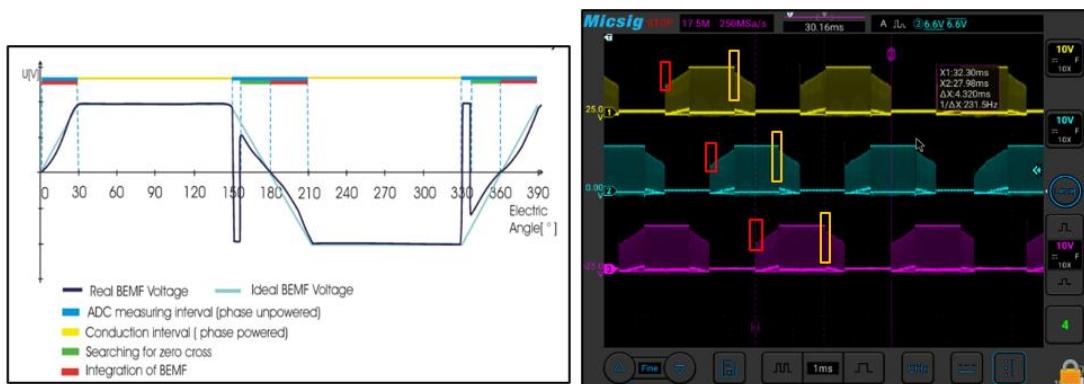
ในการวิเคราะห์สัญญาณ Back-EMF ที่เกี่ยวข้องกับการหมุนของการควบคุม 6-Step นั้น คณะผู้จัดทำข้อมูลตัวอย่างการวิเคราะห์สัญญาณของ BLDC Motor ขณะหมุนที่ Reference Speed ที่ 1998 RPM และเพื่อการวิเคราะห์และพิสูจน์สมมุติฐาน คณะผู้จัดทำสามารถแยกย่อยประเด็นได้ 3 ส่วนนั่นคือ

- Zero Crossing Point : คือจุดที่แรงเคลื่อนไฟฟ้า (Back-EMF) ของเฟสใดเฟสหนึ่งเปลี่ยนค่าจากบวกเป็นลบหรือจากลบเป็นบวก โดยคณะผู้จัดทำสังเกตได้จากการใช้ Oscilloscope ในการจับสัญญาณ Back-EMF ในช่วงการสลับเฟส U, V, และ W เทียบกับเฟสที่ต่างกัน เพื่อหาจุด Zero Crossing Point ทำให้คณะผู้จัดทำสามารถสังเกตได้ว่า จุด Zero Crossing Point นั้นคือจุดที่บ่งบอกจุดเริ่มต้น ตำแหน่งที่ Back-EMF ตัดกับจุดศูนย์ (Zero Crossing) เป็นตัวบ่งชี้ตำแหน่งของ Rotor ที่สามารถใช้สำหรับการควบคุมแบบ Sensorless ได้



รูปที่ 102 - 103 แสดงการหาจุด Zero Crossing Point ด้วยการวัดจุดตัดในสัญญาณ

- Spike : คือยอดแหลมที่พุ่งอย่างรวดเร็ว ปรากฏบนสัญญาณ Back-EMF ซึ่งเกิดขึ้น ในช่วงที่มีการสลับเฟสหรือเปลี่ยนกระแสในขดลวด สืบเนื่องมาจากกระแสคงค้างใน MOSFET ที่ใช้สิ่งการสลับเฟสใน BLDC Motor



รูปที่ 104 - 105 แสดงการจุดที่เกิดการ Spike ของสัญญาณ

- Back-EMF : ในภาพรวม ส่วนที่เป็นสัญญาณ Back-EMF จะอยู่ในลักษณะของ Trapezoidal อันเนื่องมาจากการควบคุมแบบ 6-Step



รูปที่ 106 แสดงส่วนที่เป็นสัญญาณ Back-EMF

3.10 ผลการทดลอง

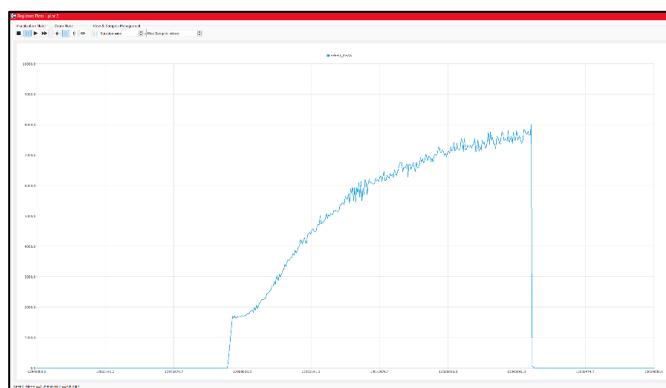
3.10.1 การวัดและบันทึกข้อมูล RPM จาก Motor Pilot

1) การตรวจสอบ Max Speed ที่ BLDC Motor สามารถทำได้

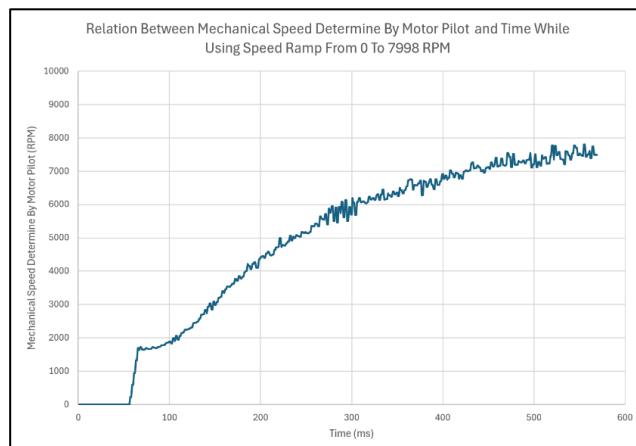
จากการทดลองหาความเร็วรอบสูงสุดที่ทำได้พบว่า BLDC Motor ตัวนี้ที่ 12V สามารถทำความเร็วสูงสุดอยู่ที่ 12350 RPM โดยประมาณ ซึ่งตรงกับค่า KV ที่มอเตอร์

2) การได้มาของค่า Mechanical Speed

จากการที่คนผู้จัดทำต้องการนำค่า Mechanical Speed ออกมานำมาเพื่อทำการทดลองเพื่อหาความสัมพันธ์ในลำดับอื่น ๆ ต่อไป แต่ภายในโปรแกรม Motor Pilot ไม่สามารถส่งออกค่าได้ ทำให้ต้องใช้โปรแกรมเสริมอย่าง Cheat Engine ทำให้ได้กราฟความเร็วรอบดังนี้



รูปที่ 107 แสดงถึงกราฟระหว่างความเร็วรอบและเวลาที่อ่านได้ผ่านโปรแกรม Motor Pilot

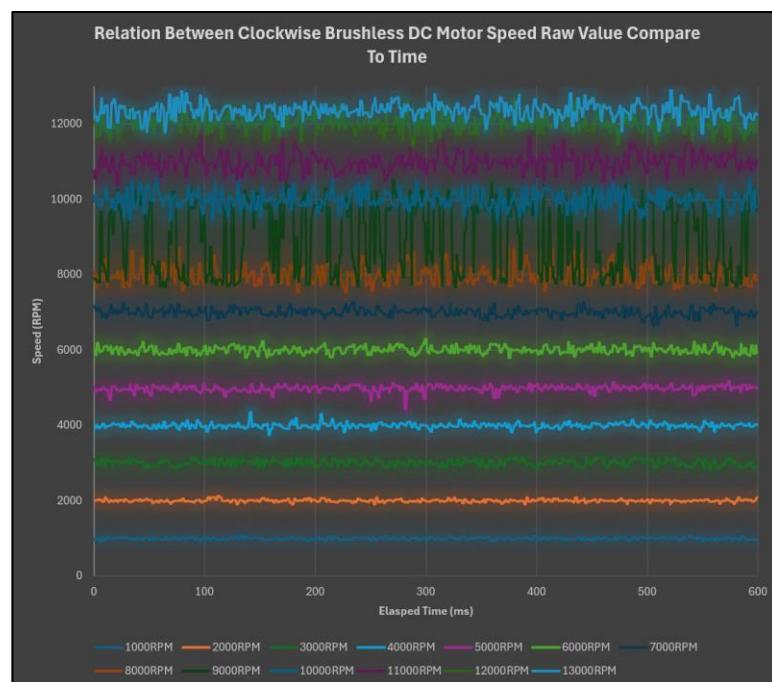


รูปที่ 108 แสดงถึงกราฟระหว่างความเร็วรอบและเวลาที่อ่านจาก Cheat Engine และสร้างกราฟผ่านโปรแกรม Excel

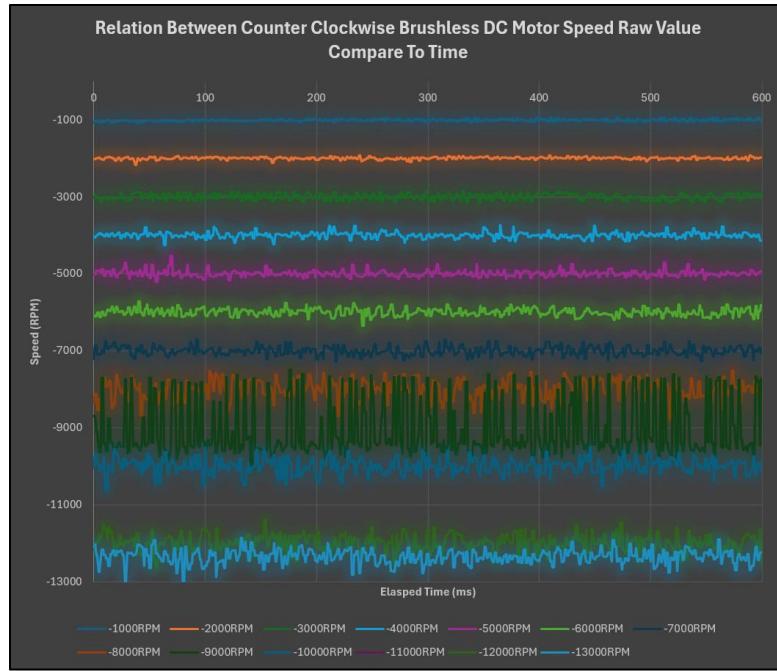
จากราฟในรูปที่ 97 และ 98 ทั้งสองจะแสดงให้เห็นว่ากราฟนั้นมีความเหมือนกันในช่วงเวลาหนึ่งซึ่งเป็นสิ่งที่สามารถพิสูจน์ได้ว่า โปรแกรม Cheat Engine สามารถที่จะอ่านค่าได้อย่างถูกต้อง

3) การหาความเร็วรอบที่แท้จริง

จากการใช้โปรแกรม Motor Pilot เพื่อเลือกสั่งงานมอเตอร์ด้วย Target Speed ทำให้คณผู้จัดทำเห็นถึงความสอดคล้องกันของข้อมูล ได้ดังนี้



รูปที่ 109 แสดงค่า Mechanical Speed ที่ส่ายไป-มาในช่วง Target Speed ที่จ่ายในทิศทางตามเข็ม

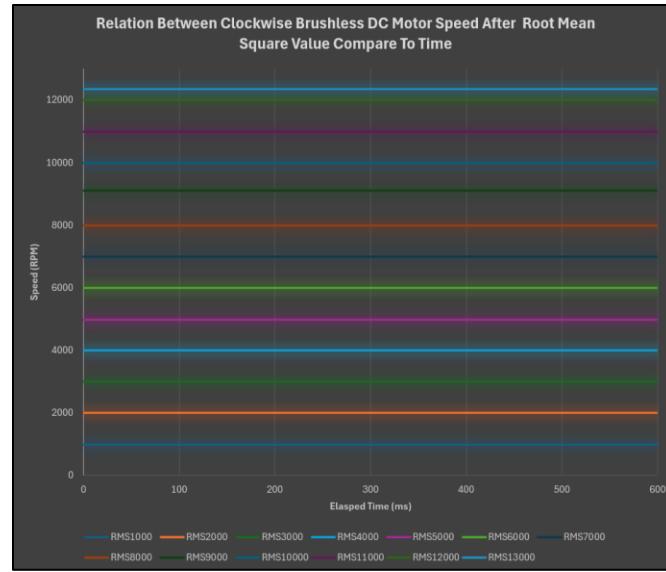


รูปที่ 110 แสดงค่า Mechanical Speed ที่ส่ายไป-มาในช่วง Target Speed ที่จ่ายในทิศทางทวนเข็ม

เนื่องจากค่า RPM ที่ได้มีความผันผวน ทำให้คณะผู้จัดทำเลือกใช้วิธีคำนวณ RMS เพื่อหาค่าประมาณในแต่ละช่วง โดยสรุปอุปกรณ์ได้ตารางดังนี้

RPM+													<- Error %
0.08325	0.069504	0.189094	0.071024	0.424074	0.019145	0.046562	0.022291	1.265239	0.092116	0.117719	0.016296	4.969577	<- Reference Speed
996	1998	3000	3996	4998	6000	6996	7998	9000	9996	10998	12000	12996	<- RMS Actual Speed
995.1708	1996.611	3005.673	3993.162	4976.805	6001.149	6999.257	7999.783	9113.871	9986.792	10985.05	12001.96	12350.15	

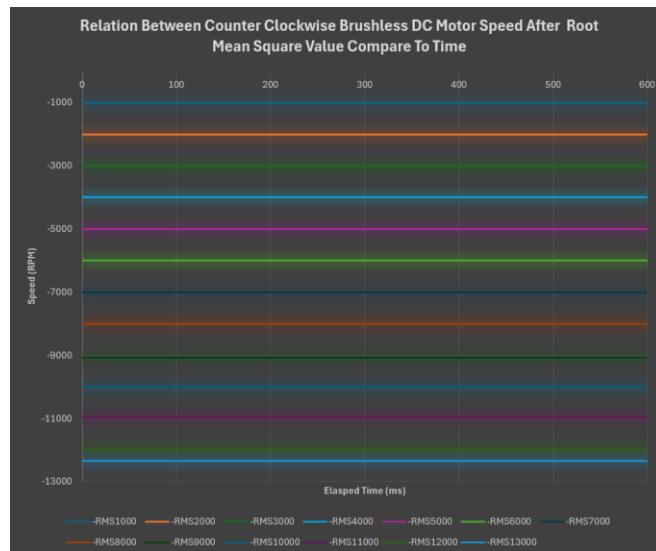
ตารางที่ 8 แสดงค่าความเร็วที่ได้จากการคำนวณด้วย RMS ในทิศทางตามเข็ม



รูปที่ 111 แสดงถึงความเร็วการหมุนของ BLDC ที่ผ่านการนำค่าผ่านการ Root Mean Square ในทิศทางตามเข็มนาฬิกา

ตารางที่ 9 แสดงค่าความเร็วที่ได้จากการคำนวนด้วย RMS ในทิศทางตามเข็ม

RPM-															<- Error %
1.02909	0.07849	0.10416	0.09567	0.11254	0.11508	0.07403	0.02384	0.74377	0.07576	0.28679	0.10765	5.01304	<- Reference Speed	<- RMS Actual Speed	
-996	-1998	-3000	-3996	-4998	-6000	-6996	-7998	-9000	-9996	-10998	-12000	-12996	<- RMS Actual Speed	<- RMS Actual Speed	
-1006.25	-1996.432	-2996.875	-3992.177	-4992.375	-5993.095	-7001.179	-7999.906	-9066.94	-9988.427	-10966.46	-11987.08	-12344.51	<- RMS Actual Speed	<- RMS Actual Speed	



รูปที่ 112 แสดงถึงความเร็วการหมุนของ BLDC ที่ผ่านการนำค่าผ่านการ Root Mean Square ในทิศทางทวนเข็มนาฬิกา

3.10.2 การวัดและบันทึกข้อมูลจากการควบคุมมอเตอร์แบบ 6-Step

เพื่อควบคุม BLDC ค่า RPM ของมอเตอร์จะมีความสัมพันธ์กับสัญญาณ PWM ที่ใช้ในการควบคุมมอเตอร์ การเข้าใจความสัมพันธ์นี้ช่วยให้คณะผู้จัดทำสามารถควบคุมความเร็วในการหมุนของมอเตอร์ได้อย่างแม่นยำและมีประสิทธิภาพ โดยข้อมูลที่เก็บได้สามารถสรุปเป็นหลักหลายส่วนดังนี้

1) การหา Phase Shift

คณะผู้จัดทำเคราะห์ลูกคลื่นจากสัญญาณได้ โดยสามารถคำนวณเพื่อหาองศาของสัญญาณที่เลื่อนได้ (Phase Shift) โดยคำนวนได้จาก

$$\text{Phase Shift} = \frac{\Delta T}{\text{Total Time}} \times 360^\circ$$

เมื่อ *Phase Shift* คือองศาของการเลื่อนเฟส

ΔT คือความแตกต่างของระยะเวลาในการทำงานของสองลูกคลื่น

Total Time คือระยะเวลาทั้งหมดของหนึ่งช่วงคลื่นสัญญาณ



รูปที่ 113 แสดงถึงระยะเวลาของลูกคลื่นสัญญาณและระยะเวลาทั้งหมดระหว่างเฟสของทั้งสองลูกคลื่นที่ Target Speed 1000

RPM

ทำให้มีวิเคราะห์จากรูปและการทำงานค่ากลับเข้าสู่สมการจะได้ค่าเพื่อคำนวณหา Phase Shift จึงได้ค่าดังนี้

$$Phase Shift = \frac{2.79ms}{8.38ms} \times 360^\circ$$

$$Phase Shift = 120^\circ$$

คณะผู้จัดทำจะสามารถสรุปได้ว่าความห่างกันของคลื่นสัญญาณทางไฟฟ้านั้นอยู่ที่ 120 องศา ระหว่างเฟสเมื่อนำมาเทียบเฟสต่อเฟสกัน

2) การหาความเร็วรอบจาก 6-Step Frequency

จากการทดลองทำให้เห็นความสอดคล้องและความใกล้เคียงของค่าความเร็วรอบที่ได้จากการคำนวณด้วยความถี่ของ 6-Step กับความเร็วรอบที่ได้จาก Mechanic Speed จาก Motor Pilot

ตารางที่ 12 แสดงตารางความสัมพันธ์ระหว่างความถี่ของ 6-Step และความเร็วรอบ

Reference RPM	Actual RPM	OSC_Measure_RPM	% Error REF_Actual	% Error_REF_OSC	% Error_Actual_OSC
-12996	-12344.50533	-12248.57143	5.277608589	6.102169349	0.777138456
-12000	-11987.08243	-12020	0.107762441	0.166389351	0.274608712
-10998	-10966.45857	-10994.28571	0.287617301	0.033783784	0.253747792
-9996	-9988.427059	-9891.428571	0.075817149	1.057192374	0.971108738
-9000	-9066.939567	-8521.142857	0.738281831	5.619635193	6.019635471
-7998	-7999.906491	-7699.714286	0.023831412	3.873984192	3.752446423
-6996	-7001.178936	-6950.571429	0.073972348	0.653594771	0.722842657
-6000	-5993.695292	-6008	0.115211048	0.133155792	0.248697996
-4998	-4992.37501	-4983.428571	0.112671625	0.292397661	0.179202053
-3996	-3992.177075	-3993.428571	0.095760401	0.0643915	0.031348715
-3000	-2998.875203	-3004	0.104268519	0.133155792	0.237740877
-1998	-1996.431697	-2005.714286	0.078555314	0.384615385	0.464959003
-996	-1006.24973	-987.7142857	1.018607024	0.83887764	1.842032267
996	995.1708296	1010.857143	0.083319408	1.469756925	1.576243278
1998	1996.611304	1987.428571	0.069552637	0.531914894	0.459915895
3000	3005.672827	2983.428571	0.188737326	0.555449148	0.740075729
3996	3993.161867	3980.857143	0.071074817	0.380391875	0.3081449
4998	4976.8048	4983.714286	0.425879667	0.286647939	0.138833764
6000	6001.148715	6008	0.019141586	0.133155792	0.114166225
6996	6999.257488	7027.142857	0.046540476	0.443179508	0.39840468
7998	7999.782858	7891.142857	0.022286334	1.354140266	1.358036875
9000	9113.871477	9060	1.249430358	0.662251656	0.591093227
9996	9986.792113	9742.857143	0.092200652	2.598240469	2.442575824
10998	10985.05327	10988.57143	0.1178577	0.085803432	0.032026788
12000	12001.95557	11905.71429	0.016293767	0.791936645	0.80188003
12996	12350.15375	12248.57143	5.229459225	6.102169349	0.822518643

จากการทดลองทำให้คณะผู้จัดทำสังเกตเห็นพฤติกรรมจากการควบคุม BLDC Motor ด้วย 6-Step ได้ว่า ความเร็ว robj จริง (Actual RPM) มีความสัมพันธ์เชิงเส้นกับค่าความถี่ที่ได้จาก 6-Step Frequency ที่ได้จากการคำนวณ (OSC_Measure_RPM) ในช่วงค่าความเร็วสูง แต่ในช่วงความเร็วกลางถึงต่ำ ค่า 6-Step Frequency มีการเปลี่ยนแปลงแบบไม่สม่ำเสมอ (เช่นค่าที่ 115.2 Hz และ 234.7 Hz) ซึ่งอาจเป็นผลจากการควบคุมที่ไม่เสถียรในช่วงความเร็วต่ำ ซึ่งความเบี่ยงเบนของ Actual RPM จาก Reference RPM มีแนวโน้มมากขึ้นในช่วงนี้ ซึ่งสอดคล้องกับพฤติกรรมของมอเตอร์ BLDC ที่ควบคุมด้วย 6-Step เนื่องจาก Torque Ripple และผลของ Back-EMF ที่ลดลงในความเร็วต่ำ รวมไปถึงแสดงให้เห็นถึงค่าความคลาดเคลื่อนของข้อมูลอย่าง ค่าความคลาดเคลื่อนระหว่าง Reference RPM และ Actual RPM (Error_REF_Actual), ค่าความคลาดเคลื่อนระหว่าง Reference RPM ต่อกำลัง ความเร็ว robj จาก 6-Step (Error_REF_OSC), ค่าความคลาดเคลื่อนระหว่าง Actual RPM ต่อกำลัง ความเร็ว robj จาก 6-Step และ (Error_Actual_OSC)

3) การหา PWM

ตารางที่ 13 แสดงค่าของ PWM ในแต่ละช่วงของความเร็ว robj

REF_RPM	Pulse Per PWM Section (ms)
13000	0.24
12000	0.245
11000	0.271
10000	0.321
9000	0.328
8000	0.378
7000	0.42
6000	0.469
5000	0.56
4000	0.736
3000	0.954
2000	1.396
1000	2.87
-1000	2.88
-2000	1.436
-3000	0.948
-4000	0.7
-5000	0.578
-6000	0.489
-7000	0.435
-8000	0.372
-9000	0.364
-10000	0.312
-11000	0.252
-12000	0.245
-13000	0.2436

ในการวิเคราะห์สัญญาณที่ได้จากการวัดด้วย Oscilloscope ทำให้ทราบว่าในหนึ่ง Step จะประกอบไปด้วยสัญญาณ PWM ภายใน จากผลการทดลองที่ได้ทำการวัดความสัมพันธ์ระหว่าง RPM ของ BLDC Motor กับพัฒนาระบบที่สัญญาณ PWM พบว่า เมื่อความเร็วรอบของมอเตอร์ (RPM) เพิ่มขึ้น ค่า Time Elapsed Pulse per PWM มีแนวโน้มเพิ่มขึ้นตามไปด้วย ขณะที่ Pulse Per PWM Section มีค่าลดลงอย่างชัดเจน ซึ่งเป็นไปตามหลักการทำงานของ PWM ในการควบคุมความเร็ว โดยความถี่ของสัญญาณ PWM เพิ่มขึ้นตามความเร็วรอบที่สูงขึ้น เช่น ที่ 1000 RPM ความถี่อยู่ที่ประมาณ 0.3484 Hz และเพิ่มขึ้นเป็น 4.0816 Hz เมื่อ RPM สูงถึง 12000 รอบต่อนาที การทดลองแสดงให้เห็น ว่าความถี่ของสัญญาณ PWM และช่วงเวลาของ Pulse สัมพันธ์ผกผันกัน

3.11 สรุปผล

3.11.1 การทดลองที่ 1

- 1) จากการทดลองพบว่า ความเร็วของมอเตอร์ (RPM) มีความสัมพันธ์เชิงเส้นกับ ความถี่ของสัญญาณที่วัดได้จากเซนเซอร์ ซึ่งสอดคล้องกับสมมติฐานที่ตั้งไว้
- 2) การเพิ่มค่าความถี่ของ PWM ส่งผลให้ความเร็วของมอเตอร์เพิ่มขึ้น โดยความสัมพันธ์นี้เป็นแบบเชิงเส้นจนถึงจุดอิมตัว (Saturation Point) ซึ่งหลังจากจุดนี้ ความเร็วของมอเตอร์ไม่เพิ่มขึ้นแม้ว่าจะเพิ่มค่าความถี่ของ PWM ต่อไป

3.11.2 การทดลองที่ 2

- 1) การควบคุมมอเตอร์แบบ Field Oriented Control (FOC) มีการศึกษาจากแหล่งข้อมูลต่าง ๆ ซึ่งพบว่า FOC ให้แรงบิดที่เสถียรและการตอบสนองที่ดีกว่า Trapezoidal Control และ Sinusoidal Control โดย FOC สามารถควบคุมมอเตอร์ได้ดีกว่าในแง่ของประสิทธิภาพการทำงานที่สูงขึ้น ทั้งในเรื่องของความแม่นยำในการควบคุมแรงบิดและการตอบสนองในสภาพที่มีการเปลี่ยนแปลงโอลด์หรือความเร็ว
- 2) การควบคุมแบบ Sensor-Based (Hall Sensor) จะมีความแม่นยำในการวัดตำแหน่งและความเร็วของมอเตอร์ที่สูงกว่าการควบคุมแบบ Sensorless (Back-EMF) โดยเฉพาะในสถานการณ์ที่มอเตอร์มีการเปลี่ยนแปลงความเร็วหรือการระการทำงานที่หลากหลาย การใช้ Hall Sensor ทำให้สามารถติดตามตำแหน่งของ Rotor ได้อย่างต่อเนื่องและแม่นยำ ซึ่งต่างจากการใช้สัญญาณ Back-EMF ที่อาจมีความแม่นยำน้อยกว่าในบางกรณีที่มอเตอร์ทำงานในสภาพที่มีความเร็วต่ำหรือการเปลี่ยนแปลงโอลด์อย่างรวดเร็ว

3.12 อกิจกรรม

3.12.1 การทดลองที่ 1

จากผลการทดลองพบว่า ความเร็วของมอเตอร์ (RPM) มีความสัมพันธ์เชิงเส้นกับ ความถี่ของ สัญญาณที่วัดได้จากเซนเซอร์ ซึ่งสอดคล้องกับสมมติฐานและสนับสนุนแนวคิดพื้นฐานเกี่ยวกับการทำงาน ของ BLDC Motor ที่ความเร็วของ Rotor ขึ้นอยู่กับความถี่ของสัญญาณควบคุม การตอบสนองของ มอเตอร์ต่อการปรับความถี่ PWM แสดงให้เห็นถึงความสัมพันธ์แบบเชิงเส้นจนถึงจุดอิมตัว (Saturation Point) ซึ่งเป็นช่วงที่ความเร็วไม่เพิ่มขึ้นแม้จะเพิ่มความถี่ของ PWM ต่อไป

ปรากฏการณ์นี้อาจเกิดจากข้อจำกัดของระบบ เช่น ความสามารถในการขับกระแสไฟฟ้าของ Controller Board หรือข้อจำกัดทางกลไกของมอเตอร์เอง ซึ่งหากต้องการหลีกเลี่ยงข้อจำกัดดังกล่าว ควรศึกษาหรือออกแบบระบบควบคุมเพิ่มเติม เช่น การเพิ่มแรงดันไฟฟ้าหรือการปรับปรุงประสิทธิภาพ ของ ESC

3.12.2 การทดลองที่ 2

แม้ไม่ได้มีการทดลองโดยตรง แต่จากข้อมูลที่ศึกษาและอ้างอิง พบว่า Field Oriented Control (FOC) มีประสิทธิภาพในการควบคุมที่ดีกว่า Trapezoidal Control และ Sinusoidal Control เนื่องจาก สามารถจัดการกับแรงบิดได้อย่างเสถียรในทุกช่วงความเร็ว และช่วยลดแรงบิดกระเพื่อม (Torque Ripple) ที่พบใน Trapezoidal Control การที่ FOC สามารถควบคุมแรงบิดได้แม่นยำขึ้นยังช่วยให้ มอเตอร์ตอบสนองได้ดีกว่าในสถานการณ์ที่มีการเปลี่ยนแปลงของโหลดหรือความเร็ว

สำหรับการควบคุมแบบ Sensor-Based การใช้ Hall Sensors ทำให้สามารถวัดตำแหน่งและ ความเร็วได้อย่างแม่นยำกว่า Sensorless Control ที่พึ่งพาการวัด Back-EMF โดยเฉพาะในช่วงความเร็ว ต่ำที่ Back-EMF มีขนาดเล็ก ซึ่งอาจทำให้เกิดความผิดพลาดในการตรวจจับตำแหน่งของ Rotor

3.13 ข้อเสนอแนะ

3.13.1 การปรับปรุงความแม่นยำในการควบคุมความเร็ว

- 1) ลดความผันผวนของความเร็ว (RPM) ในช่วงความเร็วต่ำ
 - ใช้เทคนิคการควบคุมที่แม่นยำขึ้น เช่น Field Oriented Control (FOC) เพื่อลด Torque Ripple และเพิ่มเสถียรภาพของแรงบิด โดยเฉพาะในช่วงความเร็วต่ำที่ Back-EMF มีขนาดเล็ก
 - เพิ่มความละเอียดในการควบคุม PWM โดยใช้ความถี่ PWM สูงขึ้น เพื่อลดการกระเพื่อมของสัญญาณแรงดันและปรับปรุงการทำงานของมอเตอร์
- 2) เพิ่มความแม่นยำของการวัดความเร็วรอบ (RPM)
 - ใช้เซนเซอร์เพิ่มเติม เช่น Hall Effect Sensors หรือ Encoder เพื่อวัดความเร็วและตำแหน่งของ Rotor ในช่วงความเร็วต่ำ
 - ปรับปรุงการตรวจจับ Zero Crossing Point โดยการกรองสัญญาณ Back-EMF ผ่านวงจร Lowpass Filter เพื่อลดผลกระทบจาก Noise

3.13.2 การวิเคราะห์สัญญาณที่ถูกต้องและแม่นยำขึ้น

- 1) การกรองสัญญาณ PWM และ Back-EMF
 - ใช้ Software Filtering หรือวงจรกรองสัญญาณ (เช่น RC Filter) เพื่อลด Switching Noise ที่แทรกในสัญญาณ Back-EMF
 - ปรับปรุงการตั้งค่า Oscilloscope ให้เหมาะสม เช่น เพิ่ม Sample Rate และปรับ Trigger ให้จับสัญญาณได้ชัดเจนขึ้น
- 2) วิเคราะห์ Zero Crossing Point อย่างละเอียด
 - ตรวจสอบการจับเวลา Zero Crossing Point ให้ถูกต้อง โดยเพิ่มจุดการวัดในแต่ละช่วงของคลื่น Back-EMF
 - ใช้ Comparator Circuit ช่วยตรวจจับ Zero Crossing Point ได้อย่างแม่นยำและรวดเร็วกว่า

3.13.3 ปรับปรุงวิธีการเก็บข้อมูล

1) ลดความผิดพลาดในการอ่านข้อมูล

- ใช้เครื่องมือวัดและซอฟต์แวร์ที่สามารถส่งออกข้อมูลได้โดยตรง เช่น โปรแกรมที่รองรับการ Export ข้อมูลจาก Motor Pilot เพื่อลดความซับซ้อนในการใช้โปรแกรมเสริมอย่าง Cheat Engine

2) เพิ่มความละเอียดในการวัดและบันทึกข้อมูล

- เก็บข้อมูลในช่วงเวลาที่ถี่ขึ้น เพื่อวิเคราะห์พฤติกรรมการทำงานของมอเตอร์ได้แม่นยำขึ้น
- เพิ่มจำนวน Sampling ต่อช่วงเวลาเพื่อลดผลกระทบจากความผันผวนของความเร็ว

3.13.4 การศึกษาเพิ่มเติม

1) เปรียบเทียบการควบคุมแบบ Sensor-Based และ Sensorless

- ทำการทดลองเปรียบเทียบประสิทธิภาพของการควบคุมแบบ Hall Sensor (Sensor-Based) กับการควบคุมแบบ Back-EMF (Sensorless) ในสถานการณ์ต่าง ๆ เช่น ความเร็วต่ำ ความเร็วสูง และการเปลี่ยนแปลงโหลด

2) ศึกษาผลกระทบจากการเปลี่ยนไฟฟ้า (Bus Voltage)

- ทดลองปรับแรงดันไฟฟ้าขาเข้า (เช่น 12V, 15V หรือ 24V) เพื่อศึกษาผลกระทบต่อความเร็วรอบสูงสุดและประสิทธิภาพของมอเตอร์

3) ศึกษาการควบคุมแบบ Field Oriented Control (FOC)

- พัฒนาระบบควบคุม FOC เพื่อลด Torque Ripple และเพิ่มประสิทธิภาพในการควบคุมความเร็วโดยเฉพาะในระบบที่ต้องการความแม่นยำสูง

3.13.5 การพัฒนาระบบควบคุมมอเตอร์เพิ่มเติม

1) ปรับปรุง PID Control ในการควบคุมความเร็วรอบของมอเตอร์ เพื่อเพิ่มความแม่นยำและลดความผันผวนจาก Target Speed

2) ทดสอบและปรับปรุงเฟิร์มแวร์ของ Controller Board เช่น X-NUCLEO-IHM08M1 ให้รองรับการควบคุมความเร็วในช่วงกว้างขึ้น

3.13.6 การปรับปรุงการทดลอง

- เพิ่มการวัดในทิศทางการหมุนที่แตกต่างกันเพื่อวิเคราะห์ความสมมาตรของมอเตอร์
- ทดลองใช้โหลดที่แตกต่างกัน เพื่อศึกษาผลกราฟทบทะబของโหลดต่อการทำงานของมอเตอร์ และความสัมพันธ์ของ Back-EMF กับ Torque

3.14 เอกสารอ้างอิง

- [1] Rajesh Manjibhai Pindoriya, Anshul kumar Mishra, Bharat Singh Rajpurohit, Rajeev Kumar, Analysis of Position and Speed Control of Sensorless BLDC Motor Using Zero Crossing Back-EMF Technique, Available online [Jul 2016] :
https://www.researchgate.net/publication/319175992_Analysis_of_Position_and_Speed_Control_of_Sensorless_BLDC_Motor_Using_Zero_Crossing_Back-EMF_Technique
- [2] Paridhi Wadhwani, Shivam Joshi, Trapezoidal & Sinusoidal: Two BLDC Motor Controls, Available online [March 21, 2024] : <https://bacancysystems.com/blog/trapezoidal-and-sinusoidal-bldc-motors>
- [3] SOLO Motor Controllers, Sensor vs Sensorless Motor Controllers: A Head-to-Head Comparison, Available online [ม.ป.ป.] : https://www.soloMotorcontrollers.com/blog/sensor-vs-sensorless/?srsltid=AfmBOooGJFOvVvKXLjujToEFw02tM7XVOUM3d_oRlifawyYadW1bWhz9
- [4] MathWorks, Six-Step Commutation of BLDC Motor Using Sensor Feedback, Available online [ม.ป.ป.] : <https://www.mathworks.com/help/mcb/gs/six-step-commutation-bldc-motor-using-position-sensor.html>
- [5] Marshall Brain, What Is a Brushless Motor?, Available online [Mar 6, 2024] :
<https://electronics.howstuffworks.com/brushless-motor.html>
- [6] ISL Products, An Introduction to Brushless (BLDC) Motor Control, Available online [ม.ป.ป.] :
<https://islproducts.com/design-note/brushless-motor-control-overview/>
- [7] DigiKey's North American Editors, Why and How to Sinusoidally Control Three-Phase Brushless DC Motors, Available online [Jan 1, 2017] : <https://www.digikey.co.th/th/articles/why-and-how-to-sinusoidally-control-three-phase-brushless-dc-motors>