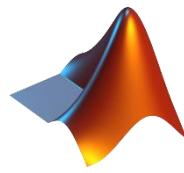


# RMXplorer



---

## LAB 1: Sensor

### Name

- นางสาว ณภัทร บัวพา 67340500011
- นาย นายนัดชัย สุขสราญ 67340500016
- นาย พุทธิพงศ์ หวานสินิท 67340500032

### Objectives

- เพื่อศึกษาหลักการทำงานของเซนเซอร์พื้นฐานที่ใช้ในการวัดเชิงกลและเชิงฟิสิกส์ ได้แก่ Potentiometer, Incremental Encoder, Magnetic Sensor (Hall Effect) และ Load Cell
- เพื่อเรียนรู้วิธีการแปลงค่าทางกายภาพ (เช่น ตำแหน่ง มุม การหมุน สนามแม่เหล็ก และแรง/น้ำหนัก) ให้เป็นสัญญาณทางไฟฟ้า (Analog Signal) ที่สามารถนำไปประมวลผลต่อได้
- เพื่อทำความเข้าใจการปรับสภาพสัญญาณ (Signal Conditioning) เช่น การขยายสัญญาณ (Amplification), การกรองสัญญาณ (Filtering) และการแปลงสัญญาณเป็นแบบ Digital เพื่อให้พร้อมใช้งานกับระบบควบคุมหรือไมโครคอนโทรลเลอร์
- เพื่อเปรียบเทียบลักษณะการตอบสนองของเซนเซอร์แต่ละชนิด รวมถึงวิเคราะห์ความสัมพันธ์ระหว่างค่าที่วัดได้กับค่าทางทฤษฎี เพื่อประเมินความถูกต้อง และความแม่นยำของอุปกรณ์
- เพื่อฝึกทักษะในการออกแบบวงจร, การต่ออุปกรณ์, การตั้งค่าระบบอ่านค่าใน Simulink หรือ STM32 และการวิเคราะห์ข้อมูลที่ได้จากการทดลองจริง
- เพื่อเสริมสร้างความเข้าใจเชิงระบบเกี่ยวกับการทำงานร่วมกันของ Sensor → Signal Conditioning → Analog-to-Digital Conversion → Data Processing ซึ่งเป็นพื้นฐานสำคัญของระบบวัดและควบคุมอัตโนมัติในงานวิศวกรรม

## 1. Potentiometer

### การทดลองที่ 1 การระบุชนิด และคุณสมบัติของ Potentiometer

#### จุดประสงค์

- เพื่อศึกษาหลักการทำงาน และชนิดของ Potentiometer
- เพื่อวิเคราะห์การตอบสนองของแรงดันไฟฟ้าข้ออกต่อระยะทางหรือองศาการหมุนของ Potentiometer แต่ละชนิดในชุดทดลอง

#### สมมติฐาน

แรงดันไฟฟ้าข้ออกของ Potentiometer จะแปรผันตามระยะทางหรือองศาการหมุน และ Linear Potentiometer จะให้ความสัมพันธ์ระหว่างองศาการหมุน/ระยะทาง กับแรงดันไฟฟ้าข้ออกเป็นเชิงเส้น ในขณะที่ Rotary Potentiometer จะให้ความสัมพันธ์เป็นแบบไม่ใช่เชิงเส้น

#### ตัวแปร

- ตัวแปรต้น:
  - องศาการหมุน และระยะทางเชิงเส้นของ Potentiometer
- ตัวแปรตาม:
  - แรงดันข้อออก  $V_{out}$  ของ INA125 (V)
- ตัวแปรควบคุม:
  - แรงดันไฟฟ้าเลี้ยงวงจร (3.3 V)
  - ชนิดของ Potentiometer ที่ใช้ในการทดสอบแต่ละครั้ง

#### เอกสารและงานวิจัยที่เกี่ยวข้อง

เอกสารของ Bourns อธิบายว่าการตอบสนองของ potentiometer ขึ้นอยู่กับลักษณะของการกระจายความต้านทานบนแทร็กภายใน โดย Type B ให้ความสัมพันธ์เชิงเส้นระหว่างแรงดันข้ออกกับตำแหน่งหมุน ขณะที่ Type A และ C มีการกระจายแบบลอการิทึมและกลับลอการิทึม จึงให้แรงดันที่เพิ่มหรือลดไม่สม่ำเสมอตามการหมุน เนื่องจากความต้านทานที่เปลี่ยนแปลงตามการหมุน

บทความของ RS-Online อธิบายหลักการของ potentiometer ว่าเป็นวงจรแบ่งแรงดันที่ปรับได้ ซึ่งแรงดันข้อออก แปรผันตามอัตราส่วนของค่าความต้านทานสองส่วนในตัวต้านทาน และสัดส่วนนี้จะเปลี่ยนไปเมื่อผู้ใช้หมุนแกน จึงสามารถใช้เป็นตัวตรวจจับตำแหน่ง (position sensor) ได้โดยตรง

Potentiometer สามารถต่อใช้งานได้สามรูปแบบหลัก ได้แก่ แบบแบ่งแรงดัน (voltage divider), แบบปรับกระแส (rheostat), และแบบควบคุมอุปกรณ์อื่น ซึ่งในงานวัดตำแหน่งนิยมใช้แบบ voltage divider เนื่องจากให้สัญญาณแรงดันแปรผันเชิงเส้นและต่อเข้ากับ ADC ของไมโครคอนโทรลเลอร์ได้โดยตรง

potentiometer ทำงานตามสมการแบ่งแรงดัน  $V_{out} = V_{in} \times \frac{R_2}{R_1+R_2}$  ซึ่งเป็นความสัมพันธ์เชิงเส้นระหว่างแรงดันอินพุตกับตำแหน่งการหมุน ดังนั้นหาก  $V_{in}$  คงที่ ค่าที่วัดได้จาก  $V_{out}$  จะเป็นตัวบ่งชี้ตำแหน่งของ wiper ได้อย่างแม่นยำ

ปัจจัยสำคัญที่มีผลต่อความแม่นยำของ potentiometer คือ tolerance ของค่าความต้านทาน, contact resistance ของ wiper และ linearity error ของแทร็ก โดยแนะนำให้เลือกอุปกรณ์ที่มีความคลาดเคลื่อนไม่เกิน  $\pm 5\%$  และใช้วงจรกรองสัญญาณเพื่อปรับปรุงเสถียรภาพในการวัด

#### ลักษณะของ Rotary Potentiometer type ต่างๆ

ประเภท	ลักษณะการตอบสนอง	ความสัมพันธ์ระหว่างการหมุน(%) กับแรงดัน(%)
Type A (Logarithmic)	การเปลี่ยนแรงดันช้าในช่วงตัน และเร็วขึ้นในช่วงปลาย	ไม่เชิงเส้นแบบ “โค้งนูนขึ้น”
Type B (Linear)	การเปลี่ยนแรงดันสม่ำเสมอตลอดช่วงการหมุน	เชิงเส้น (Linear)
Type C (Reverse Log)	การเปลี่ยนแรงดันเร็วในช่วงตัน และช้าลงในช่วงปลาย	ไม่เชิงเส้นแบบ “โค้งเว้า”

#### ลักษณะของ Linear Potentiometer type ต่างๆ

คุณสมบัติ	Type A	Type B
ลักษณะกราฟ	เส้นโค้งแบบลอการิทึม (ไม่เป็นเชิงเส้น)	เส้นตรง (เป็นเชิงเส้น)
การเปลี่ยนแปลงแรงดัน	ช่วงแรกของการหมุนแรงดันไฟฟ้าข้ามจากจะเปลี่ยนแปลงช้า และจะเปลี่ยนแปลงอย่างรวดเร็วในช่วงปลาย	แรงดันไฟฟ้าข้ามจากจะเปลี่ยนแปลงอย่างสม่ำเสมอตลอดช่วงการหมุน ( เช่น หมุนไป 50% แรงดันออก 50%)
การใช้งานหลัก	ใช้ในงานที่ต้องการการเปลี่ยนแปลงที่ละเอียดในช่วงแรก	ใช้ในงานที่ต้องการความแม่นยำในการวัดตำแหน่ง/องศา (Sensing Position) หรือใช้เป็น Voltage Divider ทั่วไป

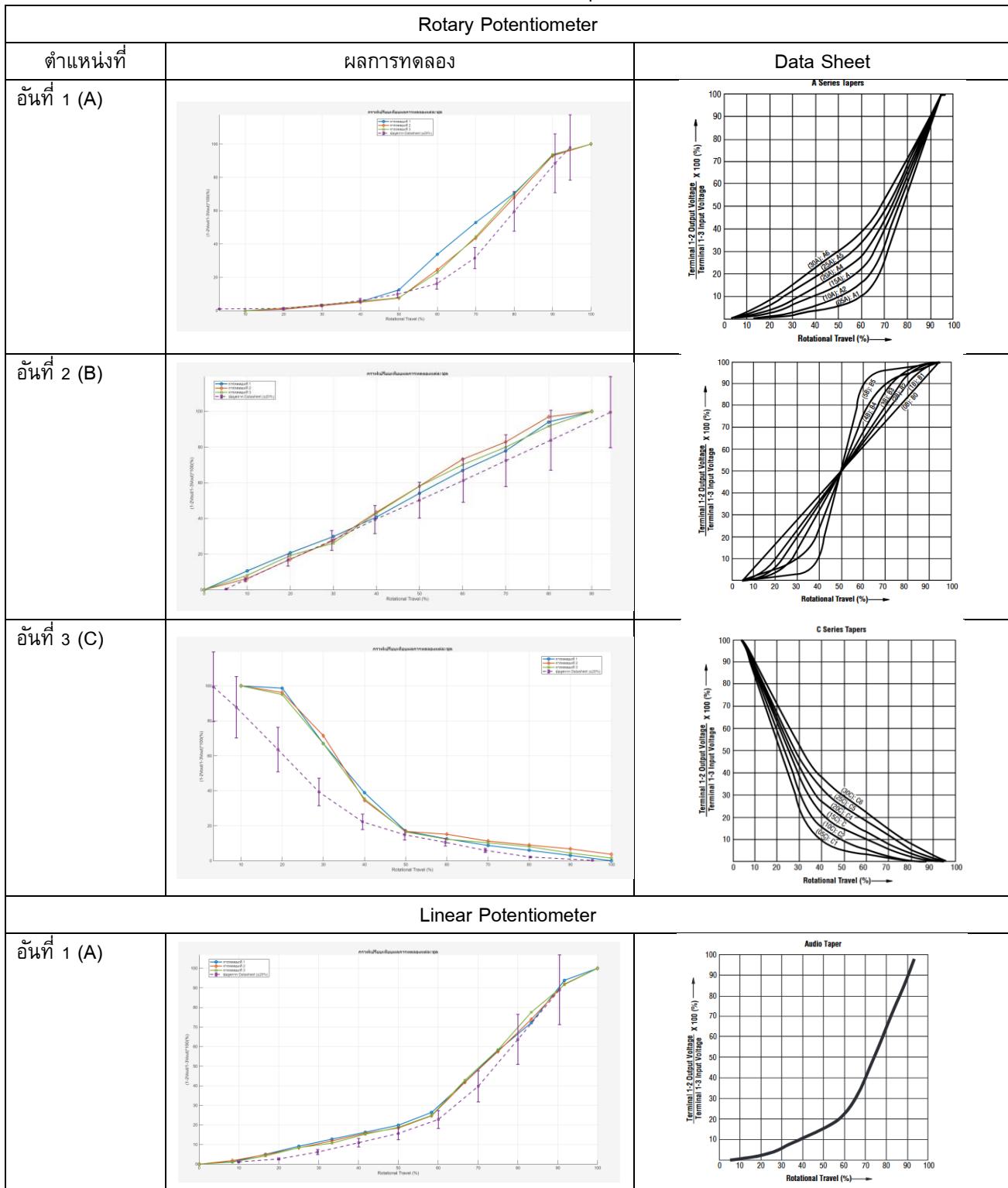
#### ขั้นตอนการดำเนินงาน

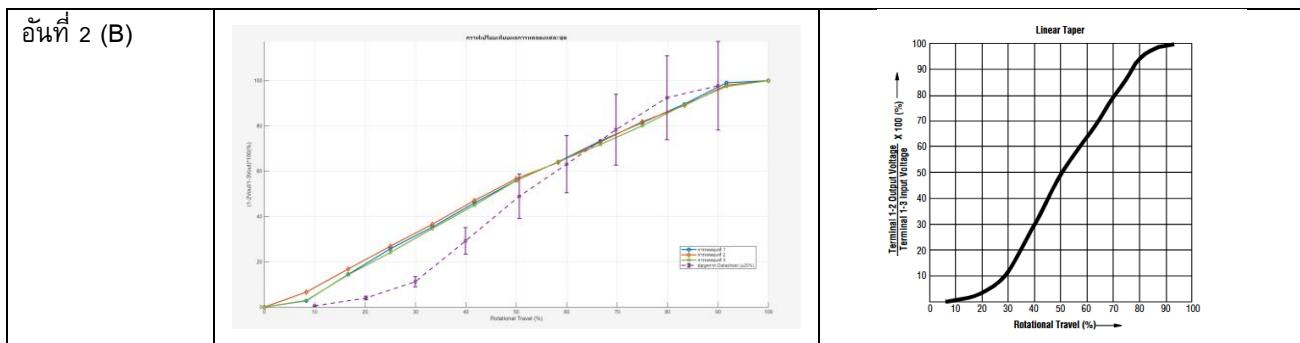
- หมุนปรับค่าไปทีละตำแหน่งของ Rotary Potentiometer อย่างสม่ำเสมอ โดยเพิ่มทีละ 10% โดยเริ่มจาก 0% ไปจนถึง 100%
- ทำการเก็บค่าจากการอ่านในโปรแกรม เมื่อปรับค่าแล้ว รอให้กราฟมีลักษณะคงที่ จึงจะสามารถนำค่านั้นไปใช้งานต่อได้
  - ทำซ้ำของแต่ละ Potentiometer อย่างน้อย 3 ครั้ง
  - บันทึกค่าแรงดันข้ามอก  $V_{out}$  ในแต่ละตำแหน่งที่หมุน
  - แสดงผลกราฟ  $V_{out} (%)$  เทียบกับการหมุน (%) เพื่อสังเกตความสัมพันธ์
  - จากนั้นทดลอง Linear Potentiometer เช่นเดียวกับ Rotary Potentiometer

7. การวิเคราะห์ผลการทดลองโดยการนำค่ามาทั้งหมด 100 ค่า จากช่วงที่กราฟมีการคงที่ ในแต่ละ ตำแหน่ง ทำการหาค่าเฉลี่ย และนำไปสร้างกราฟ  $V_{out}$  (%) เทียบกับการหมุน (%)

### ผลการทดลอง

จากการทดลองปรับตำแหน่งของ Potentiometer นำไปสร้างกราฟ  $V_{out}$  (%) เทียบกับการหมุน (%) ของการทดลอง 3 ครั้ง พร้อมเทียบกับกราฟ Type ต่างๆ เพื่อระบุว่าเป็นชนิดใด (เส้นประสม่วง)





## สรุปผลการทดลอง

จากตารางข้างต้นสามารถสรุปได้ว่า Rotary Potentiometer จำแนกงบน, กลาง และล่าง คือ Rotary Potentiometer type A, type B และ type C ตามลำดับ

สำหรับ Potentiometer แบบ Linear (Type B) พบร่วมกับแรงดันข้อออก ( $V_{out}$ ) มีความสัมพันธ์ตรงกับ จำแนกงบน ซึ่งสอดคล้องกับสมการแบ่งแรงดันทฤษฎี

$$V_{out} = V_{in} \times \frac{R_2}{R_1 + R_2}$$

สำหรับ Potentiometer แบบ A และ C มีความสัมพันธ์ไม่เป็นเชิงเส้นตามที่คาดไว้ ซึ่งเกิดจากการกระจายค่าความต้านทานของเส้นทางตัวต้านทานภายใน (resistive track) Type A มีการกระจายต้านทานแบบ log เพื่อให้เหมาะสมกับงานเสียง ส่วน Type C มีการกระจายแบบ reverse log เพื่อใช้ควบคุมสัญญาณลักษณะกลับกันกราฟที่ได้จากการทดลอง แสดงให้เห็นเส้นตรงที่มีทิศทางและรูปร่างสอดคล้องกับลักษณะของแต่ละประเภท โดยไม่มีการตัดยอดหรือสัญญาณผิดปกติ แสดงว่าอุปกรณ์ทำงานได้ถูกต้อง

Linear Potentiometer จำแนกงบน และล่าง คือ Linear Potentiometer type A และ type B ตามลำดับ

## อภิปรายผล

### ความแตกต่างของลักษณะการตอบสนองระหว่าง Type A, B, C

- Type B (Linear) แสดงลักษณะการตอบสนองตามสมการแบ่งแรงดันโดยตรง ทำให้เหมาะสมสำหรับการใช้งานทั่วไป เช่น ตัวแปรตำแหน่ง (position sensor) หรือ volume ที่ต้องการความสัมพันธ์เชิงเส้น
- Type A (Logarithmic) ให้ความละเอียดในการควบคุมมากในช่วงต้นของการหมุน และตอบสนองรวดเร็วในช่วงท้าย เหมาะกับการปรับระดับเสียง เพราะหมุนนุ่มนิ่มตอบสนองเชิงลอกการทิ่มต่อความดัง
- Type C (Reverse Log) ให้ผลตรงข้ามกับ Type A ใช้ในงานที่ต้องการการตอบสนองเร็วช่วงต้น และละเอียดช่วงปลาย เช่น การปรับความสว่าง LED แบบย้อน

### ความแม่นยำของการวัด

- ผลการวัดในช่วงกลางของการหมุน มีความเสถียรและตรงกับทฤษฎีมากที่สุด เพราะเป็นช่วงที่แรงดันเปลี่ยนสม่ำเสมอและค่าความต้านทานคงที่
- ความคลาดเคลื่อนปลายกราฟมักเกิดจากการหมุนไม่สุดทาง หรือตัว potentiometer มี dead zone เล็กน้อยที่ปลายขอบ

## ข้อเสนอแนะ

- เพิ่มการวัดค่าที่ตำแหน่งอยู่ ๆ (เช่น ทุก 5%) เพื่อให้กราฟความสัมพันธ์ละเอียดและตรวจสอบความเชิงเส้นได้ดียิ่งขึ้น
- ควรใช้ Potentiometer ที่มีค่าความต้านทานไม่สูงเกินไป เพื่อหลีกเลี่ยงปัญหา loading effect ที่ทำให้แรงดัน  $V_{out}$  ต่างจากทฤษฎี
- ควรวางแผนการเคลื่อนตัวหรือแรงกดที่ไม่สม่ำเสมอระหว่างการหมุน

## อ้างอิง

<https://eepower.com/resistor-guide/resistor-types/potentiometer-taper/#>

[https://www.bourns.com/docs/technical-documents/technical-library/corporate/OnlinePotentiometerHandbook.pdf?utm\\_source=chatgpt.com](https://www.bourns.com/docs/technical-documents/technical-library/corporate/OnlinePotentiometerHandbook.pdf?utm_source=chatgpt.com)

[https://www.rs-online.com/designspark/potentiometer-adjustable-voltage-divider-introduction?utm\\_source=chatgpt.com](https://www.rs-online.com/designspark/potentiometer-adjustable-voltage-divider-introduction?utm_source=chatgpt.com)

[https://www.keysight.com/used/us/en/knowledge/guides/potentiometer-guide?utm\\_source=chatgpt.com](https://www.keysight.com/used/us/en/knowledge/guides/potentiometer-guide?utm_source=chatgpt.com)

## การทดลองที่ 2 การจำลองวงจร Schmitt Trigger

### จุดประสงค์

- เพื่อเรียนรู้การออกแบบวงจร Schmitt Trigger เพื่อเปลี่ยนสัญญาณ Analog เป็น Digital
- เพื่อทดลองแสดงสัญญาณ Output ของ Potentiometer และ Schmitt Trigger ผ่าน Simulink
- เพื่อสังเกตการตอบสนอง และวิเคราะห์ผลการแปลงสัญญาณเชิง Analog–Digital

### สมมติฐาน

วงจร Schmitt Trigger ที่ออกแบบและจำลองใน Simulink จะสามารถแปลงสัญญาณ Analog Input ที่มีความแปรผันอย่างช้า ๆ จาก Potentiometer ให้เป็นสัญญาณ Digital Output ที่มีขอบเขตแรงดันไฟฟ้าข้าอกตามที่กำหนดได้อย่างถูกต้อง

### ตัวแปร

- ตัวแปรต้น:
  - องค์การหมุน และระยะทางเชิงเส้นของ Potentiometer
- ตัวแปรตาม:
  - แรงดันข้าอก  $V_{out}$  ของ INA125 (V)
  - สัญญาณ Digital Output จากวงจร Schmitt Trigger
- ตัวแปรควบคุม:
  - แรงดันไฟฟ้าเลี้ยงวงจร (3.3 V)
  - ชนิดของ Potentiometer ที่ใช้ในการทดสอบแต่ละครั้ง
  - ค่า Upper และ Lower Threshold ที่กำหนดสำหรับวงจร Schmitt Trigger

### เอกสารและงานวิจัยที่เกี่ยวข้อง

#### หลักการทำงานและการแปลงสัญญาณ

Schmitt Trigger ทำหน้าที่แปลงสัญญาณ Analog Input ที่อาจมีการเปลี่ยนแปลงอย่างช้า ๆ ให้เป็นสัญญาณ Digital Output และวงจรนี้แสดงให้เห็นได้ว่า Output ของระบบสามารถเปลี่ยนสถานะได้อย่างรวดเร็ว และชัดเจน (เป็น HIGH หรือ LOW) โดยแสดงให้เห็นว่าสัญญาณ Output แปรผันตามสัญญาณ Input แบบ Real Time

#### Hysteresis และ Thresholds

Hysteresis คือความแตกต่างระหว่างแรงดันไฟฟ้า Upper Threshold และ Lower Threshold ซึ่งเป็นคุณสมบัติเด่นของ Schmitt Trigger

#### การเปลี่ยนสถานะ

สัญญาณ Output จะเปลี่ยนสถานะจาก LOW เป็น HIGH เมื่อสัญญาณ Input สูงกว่า  $V_{Upper}$  และสัญญาณ Output จะเปลี่ยนสถานะจาก HIGH เป็น LOW เมื่อสัญญาณ Input ต่ำกว่า  $V_{Lower}$

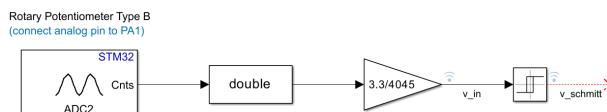
การมีช่วง Hysteresis ช่วยให้วงจร มีความทนทานต่อสัญญาณรบกวน หากไม่มี Hysteresis สัญญาณรบกวนขนาดเล็กที่อยู่ใกล้จุดเปลี่ยนสถานะเพียงจุดเดียว อาจทำให้ Output สลับสถานะไปมาอย่างรวดเร็ว แต่ Schmitt Trigger จะป้องกันปัญหานี้ได้ การออกแบบและกำหนดค่า  $V_{Upper}$  และ  $V_{Lower}$  ที่เหมาะสมใน Simulink จึงเป็นสิ่งสำคัญของการทดลองนี้ เพื่อให้การแปลงสัญญาณ Analog จาก Potentiometer เป็น Digital เป็นไปอย่างถูกต้องและมีเสถียรภาพ

### ขั้นตอนการดำเนินงาน

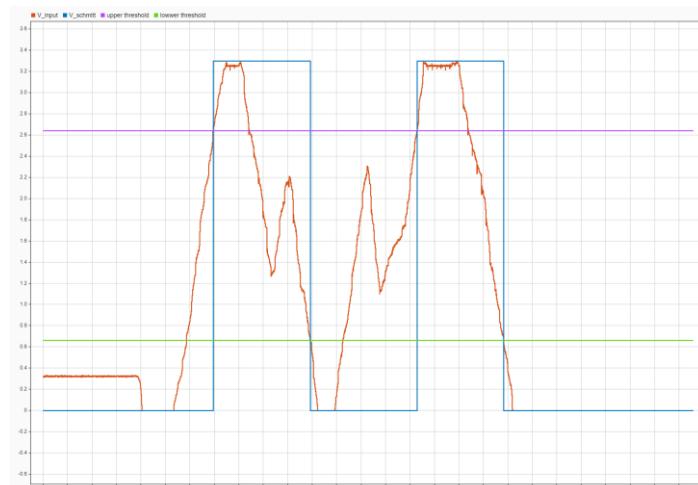
- สร้างวงจร Schmitt Trigger ใน Simulink เพื่อใช้ตรวจสอบสัญญาณจาก Potentiometer
- ตั้งค่า Upper Threshold ( $V_{UT}$ ) และ Lower Threshold ( $V_{LT}$ ) ให้เหมาะสมกับช่วงแรงดันที่ต้องการตรวจจับ
- จำลองการหมุน Potentiometer และสังเกตการเปลี่ยนสถานะของสัญญาณดิจิทัล (0 หรือ 1)
- ปรับค่า Threshold เพื่อให้การสลับสถานะเกิดในตำแหน่งที่ต้องการ
- แสดงกราฟแรงดันอินพุต (Analog) และสัญญาณเอาต์พุต (Digital) เพื่อวิเคราะห์การตอบสนองของวงจร

### ผลการทดลอง

ตั้งค่าวงจร Schmitt Trigger ให้  $V_{Lower}$  มีค่า 20% ของแรงดันไฟฟ้าเลี้ยงวงจร (3.3 V) และ  $V_{Upper}$  มีค่า 80% ของแรงดันไฟฟ้าเลี้ยงวงจร (3.3 V)



ได้ผลการทดลองดังภาพ โดยเส้นสีแดงแสดงผลการทดลองของวงจร Schmitt Trigger และสีฟ้าแสดงผลการทดลองของ  $V_{in}$



## สรุปผลการทดลอง

เมื่อใช้ Potentiometer ร่วมกับวงจร Schmitt Trigger สามารถแปลงสัญญาณจาก Analog เป็น Digital ได้สำเร็จ โดยมีการตั้งค่า Upper และ Lower Threshold ที่เหมาะสมทำให้การเปลี่ยนสถานะสัญญาณดิจิทัล (จาก 0 → 1 หรือ 1 → 0) เกิดขึ้นเฉพาะเมื่อแรงดันผ่านจุดที่กำหนด

การอ่านค่าแรงดันจาก ADC ใน Simulink สามารถแสดงผลแบบ Real-Time ได้ถูกต้องและต่อเนื่อง กราฟแรงดันที่ได้สอดคล้องกับพฤติกรรมของ Potentiometer ตามทฤษฎี

โดยสรุป การทดลองนี้ทำให้เข้าใจหลักการทำงานของ Potentiometer ในการเป็น Position Sensor รวมถึงการประยุกต์ใช้งานร่วมกับวงจร Schmitt Trigger เพื่อสร้างสัญญาณดิจิทัลที่เสถียรและเชื่อถือได้

## อภิปรายผล

หลักการแปลงสัญญาณ: วงจร Schmitt Trigger ถูกออกแบบมาเพื่อแปลงสัญญาณ Analog Input ( $V_{out}$  จาก Potentiometer) ที่อาจมีการเปลี่ยนแปลงอย่างช้าๆ หรือมีสัญญาณรบกวน ให้เป็นสัญญาณ Digital Output ที่มีความคมชัด (Square Wave)

การออกแบบและการกำหนด Threshold: วงจรถูกออกแบบใน Simulink โดยใช้บล็อกเบรี่ยบเทียบหรือบล็อกฟังก์ชัน และการกำหนด  $V_{Upper}$  และ  $V_{Lower}$  ในการทดลอง ได้กำหนดค่า Upper Threshold และ Lower Threshold ที่เหมาะสม เพราะ การกำหนด Hysteresis มีจุดประสงค์สำคัญเพื่อทำให้สัญญาณ Digital Output ไม่เกิดการสลับสถานะไปมา เมื่อสัญญาณ Input อยู่ใกล้กับจุด Threshold เพียงจุดเดียว

ผลการทดลองแสดงให้เห็นว่าสัญญาณ Digital Output ที่ได้จากการ Schmitt Trigger ใน Simulink แปร์ พันตามสัญญาณ Input แบบ Real Time เมื่อทำการหมุน Potentiometer. เมื่อ  $V_{in}$  สูงกว่า  $V_{Upper}$  Output จะเปลี่ยนเป็น High และเมื่อ  $V_{in}$  ต่ำกว่า  $V_{Lower}$  Output จะเปลี่ยนเป็น Low อย่างรวดเร็ว

จากการทดลอง สามารถยืนยันได้ว่า Potentiometer มีความสมพันธ์เชิงเส้นระหว่างตำแหน่งกับแรงดัน ข้าออก และสามารถใช้เป็นตัวตรวจจับตำแหน่งได้จริง เมื่อร่วมกับวงจร Schmitt Trigger จะช่วยให้สัญญาณ ดิจิทัลมีความเสถียรและลดผลกระทบจากสัญญาณรบกวน ซึ่งผลการทดลองทั้งหมดสอดคล้องกับหลักการทำงาน ทฤษฎีของวงจรแบ่งแรงดัน และวงจรเบรี่ยบเทียบแรงดันไฟฟ้า

## ข้อเสนอแนะ

- ปรับค่า Upper Threshold (VUT) และ Lower Threshold (VLT) ให้ห่างกันพอเหมาะสม เพื่อให้การเปลี่ยนสถานะของสัญญาณดิจิทัลชัดเจน และลดการสั่นของสัญญาณ
- หากต้องการตอบสนองไวขึ้น (เช่น ใช้กับ joystick หรือสวิตซ์ความเร็วสูง) สามารถปรับ threshold ให้แคบลง แต่ต้องควบคุม noise ให้ดีขึ้นด้วย

อ้างอิง

<https://www.ersaelectronics.com/blog/schmitt-trigger>

<https://testbook.com/electrical-engineering/schmitt-trigger>

<https://resourcespcb.cadence.com/blog/2021-schmitt-trigger-hysteresis-provides-noise-free-switching-and-output>

<https://www.allelcoelec.com/blog/schmitt-triggers-in-modern-electronics-understanding-their-role-and-capabilities.html>

## 2. Incremental Encoder

### การทดลองที่ 1 วิเคราะห์คุณสมบัติ และสัญญาณของ Encoder

#### จุดประสงค์

- อธิบายหลักการทำงานของ *Incremental Encoder* และสัญญาณ *Quadrature (A,B)*
- ตั้งค่า STM32 และ Simulink เพื่ออ่าน A และ B พร้อมแสดงผล มุม/ความเร็วแบบ *Real-Time*
- วัดหรือคำนวณ PPR และ Angular Resolution รวมถึงผลของ X1, X2 และ X4
- ออกแบบอัลกอริทึม Wrap-around/Unwrapping และ Homing
- วิเคราะห์ความคลาดเคลื่อน เปรียบเทียบผลกับทฤษฎี และสรุปข้อจำกัดของระบบ

#### สมมติฐาน

ค่า Pulses Per Revolution (PPR) ที่วัด และคำนวณได้ จะมีความสัมพันธ์โดยตรงกับค่า Angular Resolution (Degrees/Pulse) รูปแบบการอ่านค่าแบบ X4 จะให้ค่า Count ที่มีความละเอียดสูงกว่าการอ่านแบบ X1 และ X2

สัญญาณ A และ B ที่ได้จาก Encoder จะมีความสัมพันธ์เฟสต่างกัน  $90^\circ$  (Phase Relationship) ทิศทางการหมุน (Clockwise หรือ Counter-Clockwise) สามารถตรวจสอบได้จาก Phase Shift ของสัญญาณ A เทียบกับ B

ฟังก์ชัน Wrap-around ที่ออกแบบ และเขียนโปรแกรมไว้ใน Simulink จะสามารถแก้ไขปัญหาการนับค่าที่ผิดพลาด (Overflow) ที่เกิดจากการ Counter Period ทำให้ค่า Angular Position ที่แปลงออกมาจาก Raw Counts มีความต่อเนื่อง และถูกต้องแม่นยำ

ระบบที่ได้รับการออกแบบ Homing Sequence จะสามารถกำหนดจุดอ้างอิงเริ่มต้น (Home Position) ของ Encoder ได้อย่างแม่นยำ และเป็นอัตโนมัติ ทำให้การวัดตำแหน่งสัมบูรณ์ (Absolute Position) เริ่มต้นจากศูนย์ทุกครั้งที่รันระบบ

#### ตัวแปร

- ตัวแปรต้น:
  - ทิศทางการหมุน
  - ความเร็วในการหมุน
- ตัวแปรตาม:
  - Raw Counts
  - Angular Position
  - Angular Velocity
  - Phase Relationship ของสัญญาณ A และ B
- ตัวแปรควบคุม:
  - รูปแบบการอ่าน Quadrature (X1, X2, X4)

- Counter Period
- Pulses Per Revolution (PPR)

## เอกสารและงานวิจัยที่เกี่ยวข้อง

Incremental Encoder เป็นอุปกรณ์ตรวจจับการหมุน (Rotary Encoder) ที่ทำหน้าที่แปลงการเคลื่อนที่เชิงมุมของเพลาให้เป็นสัญญาณไฟฟ้าดิจิทัลในรูปของพัลส์ โดยแต่ละพัลส์จะเกิดขึ้นเมื่อเพลาหมุนผ่านหนึ่งตำแหน่งบนดิสก์ ซึ่งข้อมูลนี้สามารถนำไปใช้คำนวนหาตำแหน่งเชิงมุม, ความเร็วเชิงมุม และทิศทางการหมุนได้อย่างแม่นยำ

Rotary Encoder แบ่งออกเป็นแบบ Incremental และ Absolute โดยแบบ Incremental จะสร้างสัญญาณสองเฟลสคือ A และ B ซึ่งมีเฟสต่างกัน 90 องศา เพื่อให้สามารถระบุทิศทางการหมุนได้ทั้งแบบ CW (Clockwise) และ CCW (Counter-Clockwise) รวมทั้งสามารถเพิ่มความละเอียดในการอ่านค่าได้ด้วยการตรวจจับขอบสัญญาณแบบ X1, X2 และ X4 หลักการแปลงสัญญาณพัลส์ที่นับได้ (Raw Count) ให้เป็นค่าตำแหน่งเชิงมุม สัมพัทธ์ (Relative Position) และความเร็วเชิงมุม (Angular Velocity) โดยใช้สมการพื้นฐาน

$$\theta = \frac{2\pi N_{count}}{N_{ppr}} \text{ และ } \omega = \frac{\Delta\theta}{\Delta t}$$

ซึ่งเป็นหลักการสำคัญของระบบควบคุมการหมุนในไมโครคอนโทรลเลอร์ นอกจากนี้แนวทางในการวิเคราะห์ผลสัญญาณจาก Incremental Encoder ผ่านไมโครคอนโทรลเลอร์ โดยใช้โมดูล Input Capture (IC) หรือ Interrupt on Change (IOC) เพื่อจับขอบสัญญาณ และคำนวนความเร็วแบบเรียลไทม์ รวมถึงการจัดการโปรแกรมเพิ่มเติม เช่น Wrap-around เพื่อรีเซ็ตค่าการนับเมื่อหมุนครบหนึ่งรอบ และ Homing Sequence เพื่อกำหนดจุดอ้างอิงเริ่มต้นของการหมุน ซึ่งเป็นพื้นฐานสำคัญของระบบอัตโนมัติและการควบคุมตำแหน่ง

## ขั้นตอนการดำเนินงาน

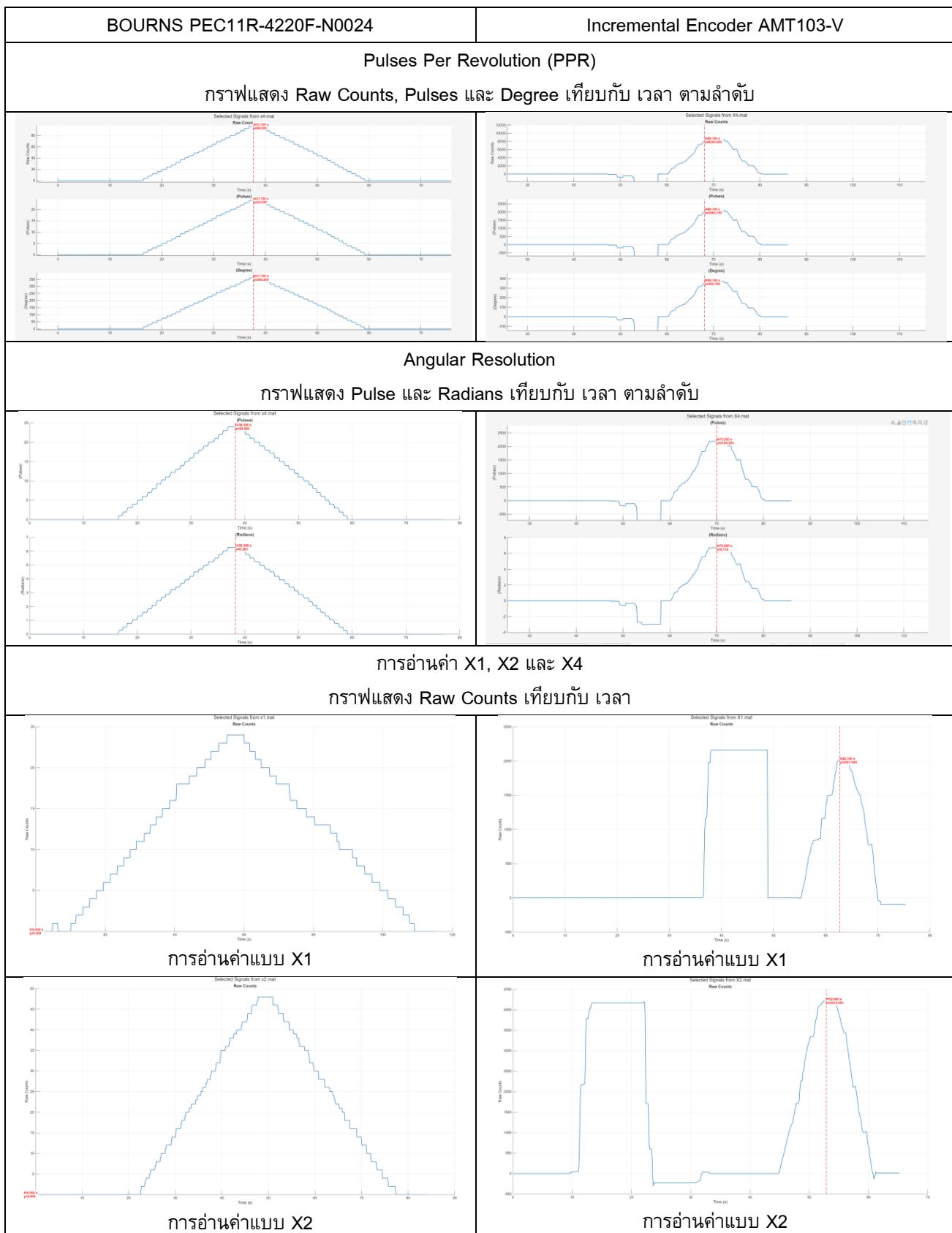
### การทดลอง

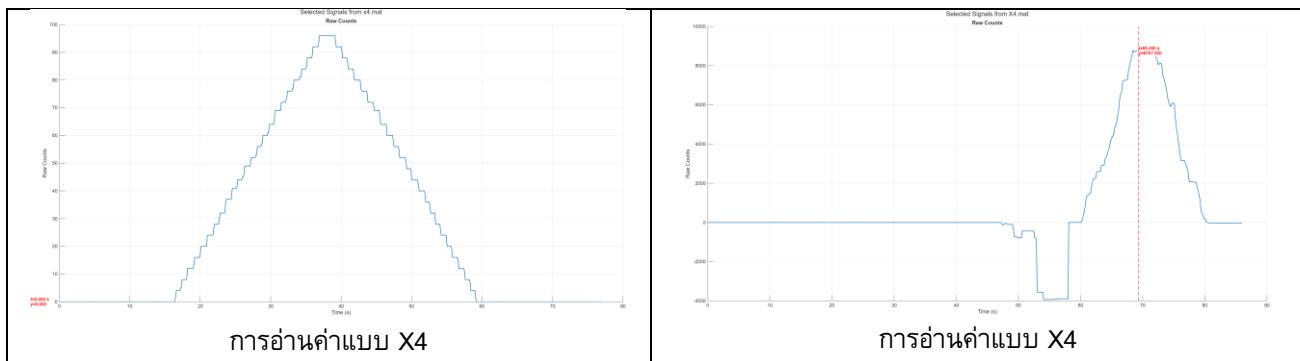
1. เลื่อน encoder มาที่จุด 0 องศาหรือจุดที่เราจะเริ่ม จากนั้นกด Homing
2. หมุน encoder 360 องศา แล้วหมุนกลับ ในทิศทาง ตรงกันข้าม พร้อมบันทึกผลการทดลอง  
การทดลองความเร็ว
  1. ทำการเอาไฟฟ้ามาติดกับตัว Encoder ด้วยเทป
  2. หมุนด้วยความเร็วที่ต่างกัน 3 ช่วงความเร็ว คือ ต่ำ, กลาง และสูง ซึ่งคือ 1, 3 และ 5 Nm ความเร็วที่ปรับได้จากไฟฟ้า
  3. หมุนด้วยความเร็วคงที่ 5 วินาที และบันทึกผลการทดลอง

### การทดลองหา PPR

1. เตรียมโปรแกรมคำนวน หา PPR จาก Raw count
2. ขีดรอยไว้ด้านบนเพื่อเป็นตำแหน่งอ้างอิง
3. หมุน Encoder ให้ได้ใกล้เคียง 360 องศา
4. บันทึกค่าที่ได้จาก Simulink
5. เปรียบเทียบค่าที่ได้ กับ Datasheet เพื่อระบุ PPR

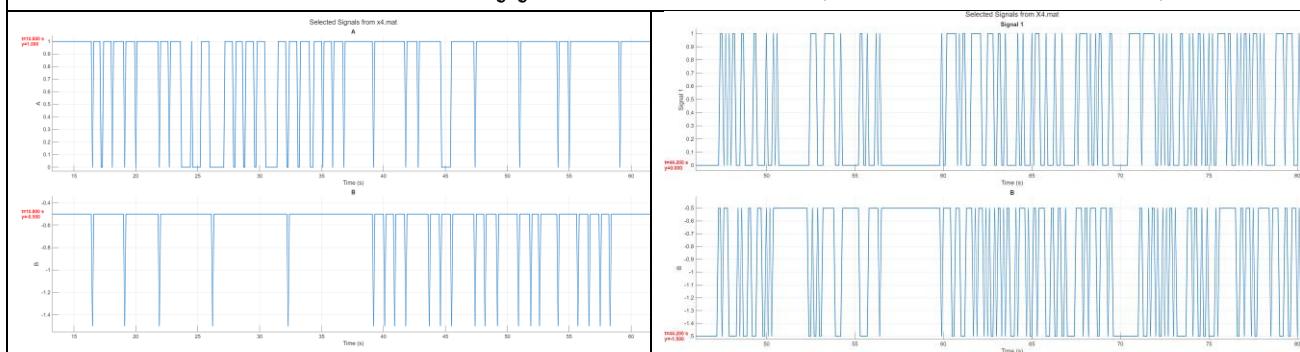
## ผลการทดลอง





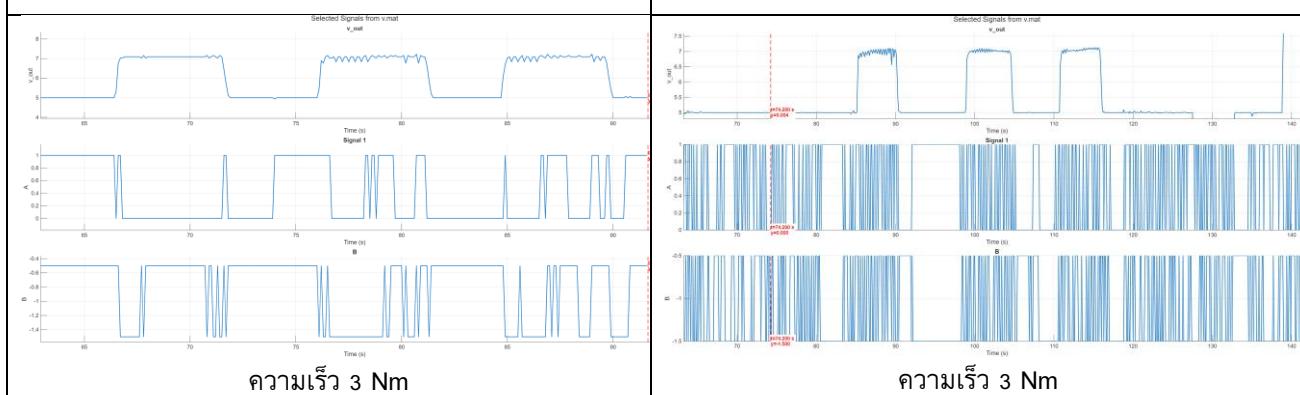
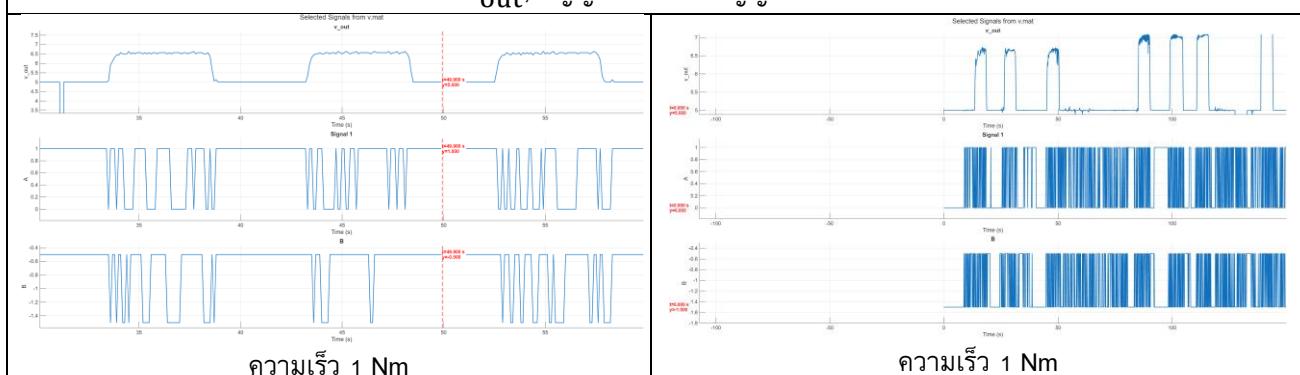
### Phase Relationship

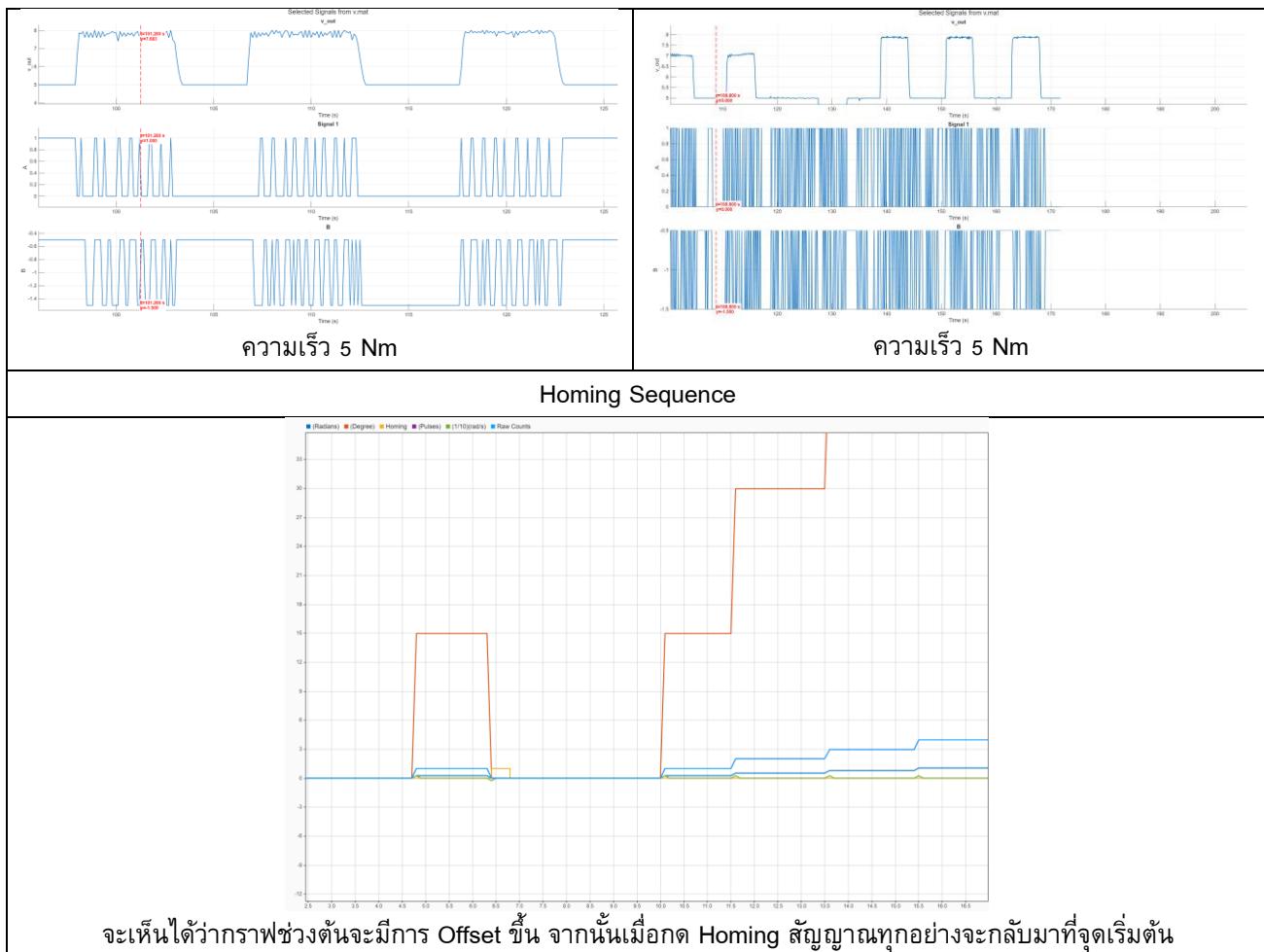
กราฟแสดงการเปลี่ยนแปลงของสัญญาณ A และ B เทียบกับเวลา (A คือ กราฟบน, B คือ กราฟล่าง)



ผลของความเร็วในการหมุนต่อคุณภาพสัญญาณได้ และสามารถเปรียบเทียบคุณภาพสัญญาณ

กราฟแสดง  $V_{out}$ , สัญญาณ A และ สัญญาณ B เทียบกับเวลา





## สรุปผลการทดลอง

สามารถยืนยันความแตกต่างของการอ่านค่า Quadrature ได้อย่างชัดเจน โดยการเปรียบเทียบกราฟ Raw Counts เทียบกับเวลาให้เห็นว่ารูปแบบ X4 ให้ค่า Raw Counts สูงสุด (ประมาณ 4 เท่าของ X1 และ 2 เท่าของ X2) ส่งผลให้ Angular Resolution ที่คำนวณได้มีความละเอียดสูงสุดตามทฤษฎี

การวิเคราะห์สัญญาณ กราฟ Phase Relationship ยืนยันว่าสัญญาณ A และ B มีความสัมพันธ์เพส ต่างกัน  $90^\circ$  ความสัมพันธ์นี้ทำให้สามารถตรวจสอบทิศทางการหมุน CW และ CCW ได้อย่างแม่นยำตลอดช่วงการทดลอง

ระบบสามารถแปลง Raw Counts ที่ได้จาก Hardware Counter ให้เป็นหน่วยวัดทางฟิสิกส์ที่ใช้งานได้จริง ได้แก่ Relative Position (Pulses), Angular Position (Radians) และ Angular Velocity (rad/s) ได้อย่างถูกต้องและต่อเนื่อง

การจัดการ Position Control การออกแบน และเขียนโปรแกรมสำหรับจัดการ Wrap-around และ Homing Sequence ได้สำเร็จ พังก์ชันเหล่านี้มีความสำคัญในการทำให้ค่า Angular Position มีความต่อเนื่อง (ไม่มีการกระโดดของค่าเมื่อเกิด Overflow) และสามารถกำหนด Home Position (ตำแหน่ง 0) ได้อย่างเป็นมาตรฐาน

## อภิปรายผล

## 1. การวิเคราะห์คุณสมบัติและการประมวลผลสัญญาณ

### หลักการทำงานของ **Encoder** และสัญญาณ **Quadrature**

ผลการทดลองจากการหมุนในทิศทาง CW และ CCW ยืนยันว่า Encoder ใช้ความต่างเฟส  $90^\circ$  ระหว่าง A และ B ในการระบุทิศทางหากสัญญาณ A นำหน้า B แสดงว่าเป็นการหมุนในทิศทางหนึ่ง และในทางกลับกัน การที่สัญญาณ B นำหน้า A แสดงว่าเป็นการหมุนในทิศทางตรงกันข้าม

### การอธิบายความแตกต่างของ X1, X2, X4

กราฟเปรียบเทียบ Raw Counts แสดงให้เห็นอย่างชัดเจนว่า Resolution ขึ้นอยู่กับจำนวนของสัญญาณที่ Hardware Counter นับได้ X4 ทำการนับทั้ง 4 ขอบ (Rising/Falling Edge ของ A และ B) ในขณะที่ X1 นับเพียง 1 ขอบต่อ Pulse ความแตกต่างนี้มีความสำคัญในการเลือก Reading Mode ที่เหมาะสมสำหรับงานที่ต้องการความแม่นยำสูง

### การแปลงหน่วยทางฟิสิกส์

การแปลง Raw Counts ไปเป็น Angular Position และ Angular Velocity แสดงให้เห็นถึงการประยุกต์ใช้ Angular Resolution ที่คำนวนได้ การที่ระบบสามารถคำนวน Angular Velocity ได้อย่างต่อเนื่องหมายถึงความเข้าใจในหลักการ Kinematics ที่ใช้ในการควบคุมระบบ

## 2. การวิเคราะห์ผลของความเร็วต่อคุณภาพสัญญาณ

ผลการทดลองแสดงให้เห็นความสัมพันธ์ระหว่างความเร็วในการหมุนกับคุณภาพของสัญญาณ A และ B ที่ความเร็วต่ำ ( $1 \text{ Nm}$ ) สัญญาณมีความคมชัดสูง (Clean Edges) ที่ความเร็วสูง ( $3$  และ  $5 \text{ Nm}$ ) สัญญาณเริ่มแสดงความผิดเพี้ยน มีสัญญาณรบกวนมากขึ้น และขอบสัญญาณไม่คมชัดเท่าที่ควร ปัญหานี้เรียกว่า Signal Degradation

สัญญาณที่ผิดเพี้ยนหรือมีความถี่สูงเกินไปมีความเสี่ยงที่จะทำให้ Hardware Counter ไม่สามารถนับขอบสัญญาณได้ทันเวลา (Missed Counts) การเกิด Missed Counts ส่งผลโดยตรงต่อความถูกต้องของ Raw Counts ซึ่งจะทำให้ค่า Angular Position และ Angular Velocity ที่แปลงออกมามีความคลาดเคลื่อนอย่างถาวร

## 3. การจัดการ Position Management ใน Simulink

การออกแบบฟังก์ชัน Wrap-around มีความสำคัญอย่างยิ่งในการจัดการ Overflow ที่เกิดขึ้นเมื่อ Raw Counts ถึงค่า Counter Period ที่ตั้งไว้ ฟังก์ชันนี้ทำหน้าที่ปรับค่า Counts อย่างต่อเนื่องในซอฟต์แวร์ เพื่อรักษาความสมบูรณ์ของข้อมูลตำแหน่งเชิงมุม

การออกแบบ Homing Sequence ทำให้ระบบมีความสามารถในการทำงานแบบ Absolute Position โดยการกำหนดจุด 0 ที่ตำแหน่งอ้างอิง

## ข้อเสนอแนะ

- การจัดการ Noise และ Velocity สูง ควรมีการพัฒนา Wrap-around Function ให้รวมถึงการตรวจจับและแก้ไข Missed Counts ที่อาจเกิดขึ้นที่ความเร็วสูง โดยใช้ Digital Filtering ใน Simulink เพื่อปรับปรุงคุณภาพสัญญาณ A และ B ก่อนเข้าสู่ QEI
- ควรมีการเปรียบเทียบ Angular Velocity ที่คำนวนได้กับอุปกรณ์วัดความเร็วเชิงมุมมาตรฐาน (เช่น Tachometer) เพื่อหา Percent Error ในการวัด Velocity ของ Encoder ทั้งสองรุ่น

## อ้างอิง

<https://www.quantumdev.com/resource-library/incremental-encoder-basics/>

<https://www.motioncontrolltips.com/faq-what-do-x1-x2-and-x4-position-encoding-mean-for-incremental-encoders/>

[https://doc.synapticon.com/node/system\\_integration/referencing\\_systems.htm?TocPath=Resources%7CSystem%20integration%20guidelines%7C](https://doc.synapticon.com/node/system_integration/referencing_systems.htm?TocPath=Resources%7CSystem%20integration%20guidelines%7C) 13

[https://doc.synapticon.com/node/system\\_integration/referencing\\_systems.htm?TocPath=Resources%7CSystem%20integration%20guidelines%7C](https://doc.synapticon.com/node/system_integration/referencing_systems.htm?TocPath=Resources%7CSystem%20integration%20guidelines%7C) 13

### 3. Ratiometric Linear Hall Effect Magnetic Sensor

การทดลองที่ 1 ศึกษาลักษณะการตอบสนองของ Magnetic Sensor และผลของ Magnetic Shielding

#### จุดประสงค์

- เพื่อศึกษาและอธิบาย หลักการทำงานของ Magnetic Sensor (Hall Effect)
- เพื่อคำนวณหาค่า Magnetic Flux Density จากแรงดันไฟฟ้าที่วัดได้
- เพื่ออธิบายลักษณะสัญญาณ Output ของ Magnetic Sensor เมื่อ Magnetic Flux Density เปลี่ยนแปลงไป
- เพื่อวิเคราะห์ ผลของ Magnetic Shielding และอธิบายความสัมพันธ์ระหว่าง Magnetic Field Shielding และ Magnetic Flux Density
- เพื่อแสดงให้เห็นว่าสัญญาณ Output แปรผันตามสัญญาณ Input แบบ Real Time โดยมี Output เป็น Raw Signal หรือ Magnetic Flux Density ในหน่วย SI Derived

#### สมมติฐาน

แรงดันไฟฟ้าข้าอกองของ Magnetic Sensor จะแปรผันตาม Magnetic Flux Density และจะให้ค่า Output ที่แตกต่างกันเมื่อวัดด้วยขั้วแม่เหล็กที่ต่างกัน (เหนือ/ใต้)

สัญญาณ Output ของเซนเซอร์จะมีความสัมพันธ์ เป็นเชิงเส้น (Linearity) กับระยะห่างจากแม่เหล็กในสภาวะที่ไม่มี Shielding

การใช้ Magnetic Field Shielding จะส่งผลให้ค่า Magnetic Flux Density ที่วัดได้ ลดลงอย่างมีนัยสำคัญ เมื่อเทียบกับสภาวะที่ไม่มี Shield เนื่องจาก Shield ทำหน้าที่ป้องกันหรือเบี่ยงเบนสนามแม่เหล็กไม่ให้เข้าถึงตัวเซนเซอร์โดยตรง

#### ตัวแปร

- ตัวแปรต้น:
  - ระยะห่างระหว่างแม่เหล็กและเซนเซอร์ (mm)
- ตัวแปรตาม:
  - ค่าแรงดันเอาต์พุต ( $V_{out}$ , mV) และ Magnetic Flux Density (B, mT)
- ตัวแปรควบคุม:
  - ทิศทางขั้วแม่เหล็ก (เหนือ/ใต้)
  - อุณหภูมิขณะทดลอง
  - แรงดันไฟฟ้าเข้า 3.3V
  - การมี และไม่มี Shield

## เอกสารและงานวิจัยที่เกี่ยวข้อง

### หลักการ Hall Effect และการแปลงสัญญาณ

Magnetic Sensor รุ่น DRV5055 ทำงานบนหลักการ Hall Effect โดยเมื่อกระแสไฟฟ้าไหลผ่านวัสดุตัวนำ และมีสนามแม่เหล็กดึงจากมาต่อกันจะเกิดแรงดันไฟฟ้าคร่อมตัวนำในทิศทางตั้งฉากกับทิศกระแสและสนามแม่เหล็ก เรียกว่า Hall Voltage

Ratiometric Linear Sensor DRV5055 เป็นชนิด Ratiometric Linear หมายความว่า แรงดันไฟฟ้า Output ที่ได้จะเป็นเชิงเส้น ตามความเข้มของสนามแม่เหล็ก (Magnetic Flux Density) ที่วัดได้ นอกจากนี้ Output เป็นสัดส่วน (Ratiometric) กับแรงดันไฟฟ้าเลี้ยงวงจรด้วย

สมการคำนวณ Magnetic Flux Density สำหรับ Hall Voltage ที่เป็น Output จากเซนเซอร์ ( $V_{out}$ ) สามารถใช้คำนวณหาค่า Magnetic Flux Density ( $B$ ) ในหน่วย mT ได้ โดยมีความสัมพันธ์กับค่าความไวของเซนเซอร์ที่อุณหภูมิที่  $25^{\circ}\text{C}$  (Sensitivity), แรงดันไฟฟ้าครั้งหนึ่งของแรงดันไฟฟ้าขาเข้า ( $V_Q$ ), ค่าสัมประสิทธิ์ของอุณหภูมิ ( $S_{TC}$ ) ( $\frac{\%}{^{\circ}\text{C}}$ ) และอุณหภูมิแวดล้อม ( $T_A$ ) ดังสมการ

$$V_{out} = V_Q + B \times (\text{Sensitivity}_{25^{\circ}\text{C}} \times (1 + S_{TC}(T_A - 25^{\circ}\text{C})))$$

### การตอบสนองต่อข้าแม่เหล็กและ Magnetic Shielding

เซนเซอร์ DRV5055 เป็น Linear Sensor ที่สามารถตอบสนองต่อข้าแม่เหล็กที่แตกต่างกัน เมื่อแม่เหล็กข้ามหนึ่ง ( เช่น ข้าวเหนียว ) เข้าใกล้ เซนเซอร์จะให้แรงดันไฟฟ้า Output ที่สูงกว่า ค่าแรงดันไฟฟ้า Quiescent ( $V_Q$ ) และเมื่อแม่เหล็กอีกข้ามหนึ่ง ( เช่น ข้าวไถ ) เข้าใกล้ เซนเซอร์จะให้แรงดันไฟฟ้า Output ที่ต่ำกว่า ค่า  $V_Q$

การเปรียบเทียบผลการทดลองระหว่างข้าวเหนียวและข้าวไถ จึงมีความสำคัญในการวิเคราะห์ลักษณะสัญญาณ Output เมื่อ Magnetic Flux Density เปลี่ยนแปลงไป

### Magnetic Field Shielding

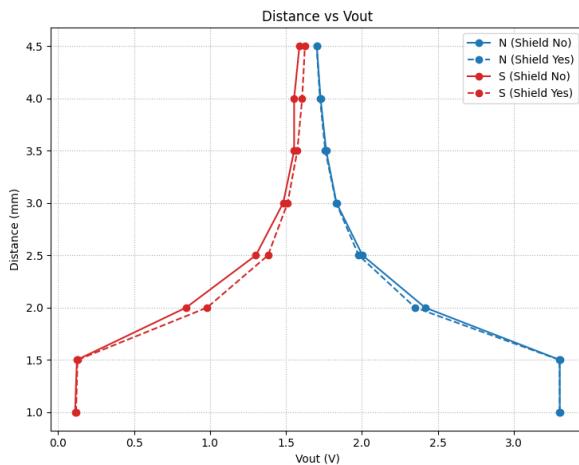
วัสดุที่ใช้ในการทำ Shielding ( การป้องกันสนามแม่เหล็ก ) มักเป็นวัสดุที่ยอมให้สนามแม่เหล็กไหลผ่านได้ง่าย และเมื่อวาง Shield ไว้ระหว่างแหล่งกำเนิดแม่เหล็ก และเซนเซอร์ Shield จะทำหน้าที่เบี่ยงเบน (Divert) เส้นแรงแม่เหล็กให้ไหลผ่านตัวเองแทนที่จะผ่านเซนเซอร์

## ขั้นตอนการดำเนินงาน

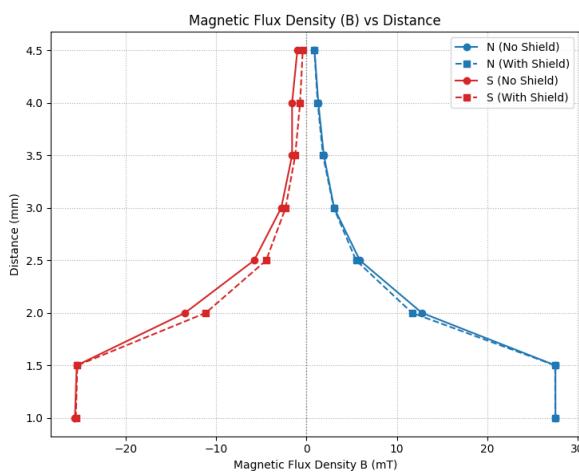
- ต่อวงจร Magnetic Sensor เข้ากับ ADC และระบบ Simulink เพื่อเก็บข้อมูลแรงดันแบบ Real Time
- จัดตำแหน่งแม่เหล็กข้าว N และ S ให้อยู่ห่างจากเซนเซอร์ตามค่าที่กำหนด (1.0–4.5 mm)
- วัดค่าแรงดันเอาต์พุต ( $V_{out}$ ) ด้วยมัลติมิเตอร์ และบันทึกสัญญาณ ADC ผ่าน Simulink พร้อมกัน
- ทำซ้ำอย่างน้อย 3 ครั้งในแต่ละกรณี (ข้าวเหนียวมี Shield, ข้าวเหนียวไม่มี Shield, ข้าวไถมี Shield, ข้าวไถไม่มี Shield)
- คำนวณค่าเฉลี่ยของ  $V_{out}$  และแปลงเป็น  $B$  (mT)
- สร้างกราฟความสัมพันธ์  $V_{out}$ ,  $B$  และ Distance
- วิเคราะห์ความแตกต่างของสัญญาณเมื่อมี และไม่มี Shield พร้อมประเมินความเป็นเชิงเส้น (Linear Regression,  $R^2$ )

## ผลการทดลอง

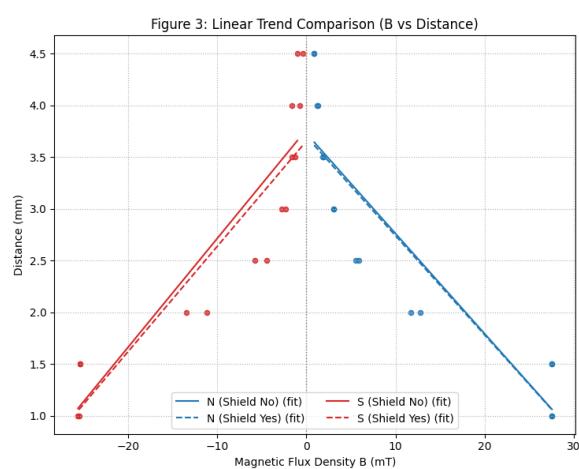
นำผลการทดลองมาสร้างกราฟ แยกกรณีขั้วเหนือ และขั้วใต้ มี และไม่มี Shield ดังนี้ กราฟที่ 1 แสดงการเปรียบเทียบระหว่างระยะทาง และ  $V_{out}$



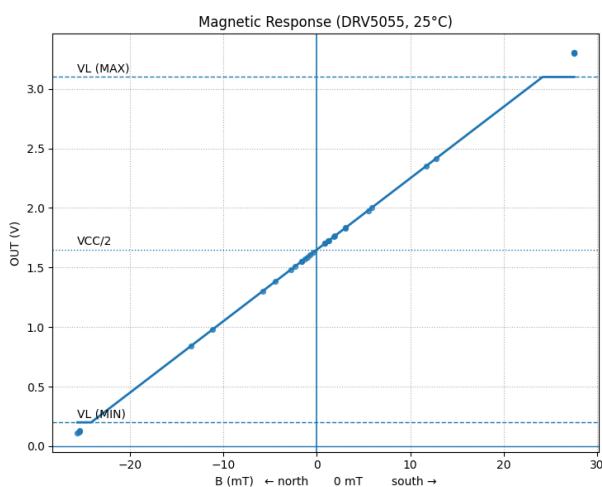
กราฟที่ 2 ความสัมพันธ์ระหว่าง Magnetic Flux Density (B) และระยะทาง



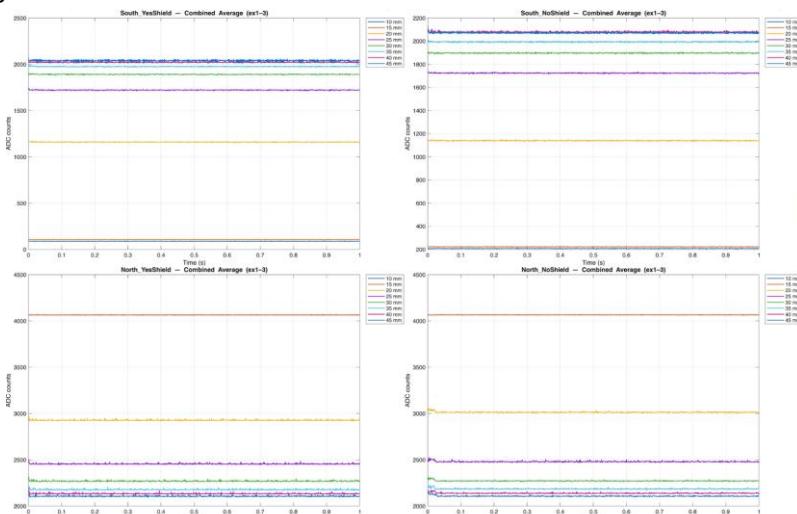
กราฟที่ 3 แสดงการเปรียบเทียบเส้นแนว Linear ของแต่ละกรณี



กราฟที่ 4 แสดงความสัมพันธ์ระหว่าง Magnetic Flux Density (B) กับ  $V_{out}$



กราฟที่ 5 แสดงสัญญาณ Output แบบ Real Time จาก Simulink (ADC) แสดง Raw Signal กับ เวลา

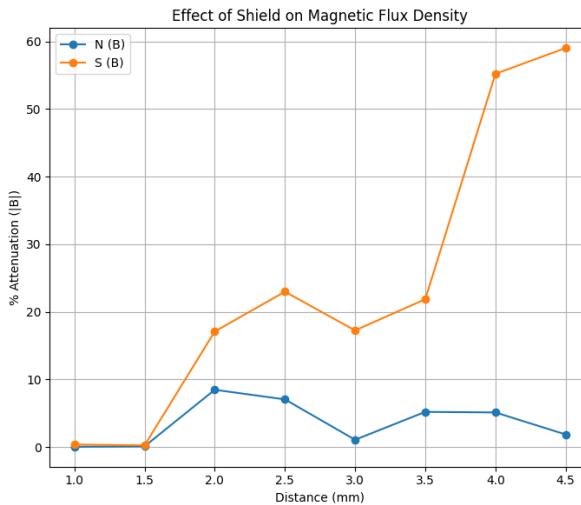


#### ลักษณะการตอบสนองของแม่เหล็ก

ข้าวเหนือ ยิ่งเข้าใกล้เซนเซอร์  $V_{out}$  ก็ยิ่งเพิ่มขึ้น ตัวอย่างจากกราฟที่ 1 พบว่าที่ Distance = 1.00 mm ค่า  $V_{out}$  ที่ได้ออกมาเป็น 3.3V ซึ่งก็คือ  $V_{out}$  สูงสุดของบอร์ด STM32 ข้าวใต้ จะตรงกันข้ามกัน ยิ่งเข้าใกล้เซนเซอร์  $V_{out}$  ก็จะยิ่งลดลง และจากกราฟที่ 2 พบว่ายิ่ง Distance ใกล้ขึ้น ค่า B และ  $V_{out}$  ลุยก้าว 0 หมายความว่าระยะห่างมีผลต่อการตรวจจับของเซนเซอร์ ที่ตำแหน่งสมดุล  $B = 0$  มี  $V_{out} \approx 1.65V$  ซึ่งตรงกับ  $V_Q$  (Quiescent Voltage) ตาม Datasheet ของ DRV5055A1/Z1 ที่  $T_A = 25^\circ C$  เมื่อเทียบ ความสัมพันธ์ระหว่าง Magnetic Flux Density (B) กับ  $V_{out}$  ดังกราฟที่ 4 พบว่าเราสามารถระบุข้าวได้จาก  $V_{out}$  เทียบกับ B เมื่อ spanning แม่เหล็ก ข้าวใต้ ( $B > 0$ ) เข้าใกล้เซนเซอร์ ค่า  $V_{out}$  เพิ่มขึ้นเกิน 1.65V เมื่อ spanning แม่เหล็กข้าวเหนือ ( $B < 0$ ) เข้าใกล้เซนเซอร์ ค่า  $V_{out}$  ลดลงต่ำกว่า 1.65V โดยค่า  $V_{out}$  จะเปลี่ยนแปลงในช่วงประมาณ 0.2 – 3.1V ซึ่งเป็นช่วง Linear Range ของเซนเซอร์ตาม Datasheet จุดตัดแกนที่ 1.65V ยืนยันว่า เซนเซอร์ DRV5055A1/Z1 ทำงานถูกต้องและมีความเป็น linear สูงในช่วง  $\pm 22$  mT (จาก Datasheet)

## ผลการทดลองของ Magnetic Shielding

Magnetic Flux Density ( $B$ ) และ ค่าแรงดันเอาต์พุต ( $V_{out}$ ) ที่วัดได้ ในกรณีที่มี Shield จะต่ำกว่ากรณีไม่มี Shield เล็กน้อย โดยมีแนวโน้มลดลงเด่นชัดขึ้นเมื่อสนามแม่เหล็กมีความเข้มสูงหรือตำแหน่งอยู่ใกล้เซ็นเซอร์ ตัวอย่างเช่น ที่ระยะ 2.0 mm ข้าวเหนือที่ไม่มี Sheild = 12.77 mT และมี Sheild = 11.68 mT % ลดลงประมาณ 8.6% ที่ระยะ 2.0 mm ข้าวใต้ที่ไม่มี Sheild = 13.45 mT และมี Sheild = 11.15 mT % ลดลงประมาณ 17.1%



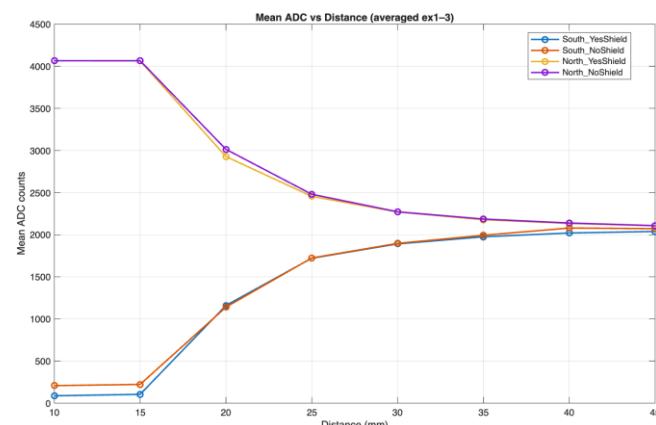
การมี Shield ทำให้สนามแม่เหล็กที่ถึงเซ็นเซอร์ลดลง เนื่องจากแผ่น Shield ทำหน้าที่ดูดซับและเบี้ยงเบนเส้นแรงแม่เหล็กบางส่วน ทำให้ความหนาแน่นของสนามแม่เหล็ก ( $B$ ) ที่เซ็นเซอร์รับได้ลดลง

ผลของ Shield มีความสำคัญเฉพาะบริเวณระยะใกล้ที่ระยะ 1–2.5 mm ซึ่งเป็นช่วงที่  $B$  มีค่าสูง สนามแม่เหล็กถูกดูดซับได้มากกว่า เมื่อระยะเพิ่มขึ้น ( $> 3 \text{ mm}$ ) สนามแม่เหล็กลดลง Shield จึงแทบไม่ส่งผลกระทบ

แนวโน้มของ  $B$  และ  $V_{out}$  ยังคงเป็นเชิงเส้นแม่มี Shield แสดงว่า Shield ไม่ได้รบกวนลักษณะการตอบสนองเชิงเส้นของเซ็นเซอร์ DRV5055 แต่เพียงลดค่าความเข้มของ  $B$  โดยรวมเท่านั้น

## การบันทึกสัญญาณ Real Time (Simulink Output)

กราฟแสดงการเปรียบเทียบระหว่างค่าเฉลี่ย ADC จากการทดลองช้า 3 ครั้ง เทียบกับระยะทาง



สัญญาณ ADC ที่บันทึกจาก Simulink แสดงรูปคลื่น Raw Signal เทียบกับ Time ที่สอดคล้องกับการเคลื่อนที่ของแม่เหล็กแบบเรียลไทม์ เส้นสัญญาณมีความต่อเนื่อง และเสถียร (noise ต่ำ) และระดับสัญญาณ

แยกแยะได้ชัดตามระยะห่างของแม่เหล็กในแต่ละกรณีทดลอง (ข้าวเหนือ/ข้าวใต้, มี/ไม่มี Shield) เมื่อเปรียบเทียบค่าแรงดันที่คำนวณจาก ADC กับค่าที่วัดจริงพบว่า ต่างกันไม่เกิน 0.02 V แสดงว่าแบบจำลองการวัด และการแปลง ADC เป็นแรงดันมีความถูกต้องเพียงพอ สามารถยืนยันว่า เช่นเซอร์ตตอบสนองแบบ Real-Time และวงจร ADC อ่านค่าได้ถูกต้อง รวมถึงสามารถตรวจจับความต่างของทิศทาง และผลของการทำ Shield ได้อย่างมีนัยสำคัญ

## สรุปผลการทดลอง

1. ทิศทางของสนามแม่เหล็ก (ข้าวเหนือ และใต้) มีผลโดยตรงต่อแรงดันเอาต์พุต ( $V_{out}$ ) ข้าวเหนือ ทำให้แรงดันลดลงต่ำกว่า 1.65V และข้าวใต้ ทำให้แรงดันเพิ่มขึ้นสูงกว่า 1.65V ซึ่งตรงกับลักษณะการตอบสนองของ Hall Effect sensor ตามทฤษฎี และข้อมูลใน Datasheet
2. ระยะห่างระหว่างแม่เหล็กกับเซนเซอร์มีผลอย่างชัดเจน เมื่อระยะเพิ่มขึ้น ค่าความหนาแน่นของสนามแม่เหล็ก และ  $V_{out}$  จะลดลงอย่างต่อเนื่อง โดยเฉพาะในช่วง 1–3 mm ที่สนามมีความเข้มสูง
3. การมี Magnetic Shield ส่งผลลดค่า B และ  $V_{out}$  ในระยะใกล้ (1–2.5 mm) ค่า B ลดลงเฉลี่ย 8–20% ขึ้นอยู่กับข้าวแม่เหล็ก แต่ไม่มีผลชัดเจนในระยะเกิน 3 mm Shield จึงทำให้ที่ดูดซับ และเบี่ยงเบนเส้นแรงแม่เหล็กบางส่วนแต่ไม่ทำให้การตอบสนองสูญเสียความเป็นเชิงเส้น
4. ความสัมพันธ์ระหว่าง B กับ Distance เป็นเชิงเส้นเฉพาะช่วง 2–3.5 mm โดยมี  $R^2$  ประมาณ 0.8 แสดงว่าในช่วงใกล้เกินไปเซนเซอร์เข้าสู่ภาวะอิมตัว และในระยะใกล้กว่า 4 mm สัญญาณรับกวนมีผลต่อค่า B มากขึ้น
5. ผลจาก Simulink (ADC Output) แสดงสัญญาณแบบ Real-Time ที่ต่อเนื่อง และสอดคล้องกับการเคลื่อนของแม่เหล็ก ค่า  $V_{out}$  ที่คำนวณจาก ADC มีความคลาดเคลื่อนจากค่าจริงไม่เกิน 0.02 V แสดงว่าแบบจำลองการแปลง ADC → Voltage มีความถูกต้องเพียงพอ

โดยรวม เซนเซอร์ DRV5055A1/Z1 สามารถตรวจจับทิศทางและความเข้มของสนามแม่เหล็กได้อย่างแม่นยำ มีความเป็นเชิงเส้นสูงในช่วงทำงาน และตอบสนองแบบ Real-Time ได้ดี

## อภิปรายผล (วิเคราะห์สิ่งที่เกิดขึ้นในการทดลอง)

ผลการวัดสอดคล้องกับหลักการของ Hall Effect ซึ่งแรงดันข้ามอย่างเซนเซอร์จะเพิ่มขึ้นหรือลดลงตามทิศทางของสนามแม่เหล็กที่กระทำต่อเซนเซอร์ ค่า  $V_{out}$  ที่สูงกว่า  $V_Q$  บ่งบอกถึงทิศทางฟลักซ์เข้าสู่ด้านทิศเหนือในขณะที่ค่าต่ำกว่า  $V_Q$  หมายถึงทิศทางจากทิศใต้ ซึ่งตรงตามทฤษฎีใน datasheet ของ DRV5055

แผ่น shield มีผลลดทอนค่า B ได้จริง โดยค่าที่ลดลงมากที่สุดจะเกิดเมื่อวางแผ่นใกล้แม่เหล็กมากที่สุด สาเหตุที่ shield ลดทอนฟลักซ์ได้ เพราะวัสดุมี magnetic permeability สูง ทำให้เส้นแรงแม่เหล็กเปลี่ยนทิศทางและกระจายไปตามแผ่นโลหะแทนที่จะเข้าสู่เซนเซอร์

ช่วงแรงดัน  $V_{out}$  เปลี่ยนแปลงกับ B มีลักษณะเชิงเส้นภายในขอบเขต  $\pm 22$  mT ตามスペค แต่เมื่อแม่เหล็กอยู่ใกล้มาก กราฟเริ่มอิมตัว เนื่องจากการจำกัดของวงจรขยายภายใน IC

Magnetic Sensor เช่น DRV5055 สามารถนำไปใช้ในงาน position sensing, rotational speed measurement หรือ contactless switch โดยใช้คุณสมบัติในการตอบสนองต่อข้าวแม่เหล็กและการลดทอนจากวัสดุ shield

## ข้อเสนอแนะ

- เพิ่มระบบกันสั่นระหว่างแม่เหล็กกับเซนเซอร์
- ใช้การกรองสัญญาณแบบ Butterworth 3rd Order เพื่อลด noise ได้ดียิ่งขึ้น
- ใช้ Shield ความหนาหากลายเพื่อศึกษาความสัมพันธ์เชิงลึก
- การใช้แรงดันไฟฟ้าขาเข้าคงที่ และตรวจสอบค่า offset ก่อนเริ่มทดลอง
- ใช้สัญญาณ Real Time จาก Simulink เพื่อวิเคราะห์ dynamic response เพิ่มเติม

## อ้างอิง

<https://www.ti.com/product/DRV5055>

<https://www.google.com/search?q=https://www.electronicshub.org/hall-effect-sensor/>

<https://www.sciencedirect.com/topics/engineering/magnetic-flux-density>

<https://www.google.com/search?q=https://www.edn.com/magnetic-shielding-techniques-in-electronics/>

## 4. Load Cell

### การทดลองที่ 1 การเปรียบเทียบค่าที่ได้กับเครื่องชั่ง Digital

#### จุดประสงค์

- เพื่อศึกษาและอธิบายหลักการทำงานของ Load Cell, วงจร Wheatstone Bridge, Strain Gauge และ 2 Op-amps Differential Amplifier (INA125) ได้อย่างถูกต้อง
- เพื่อคำนวณหาค่า Gain ที่เหมาะสม สำหรับ INA125 และอธิบายความสัมพันธ์ระหว่างค่า Gain กับค่าความต้านทานที่ใช้ปรับ Gain
- เพื่ออธิบายกระบวนการ Signal Conditioning และ Calibration ทั้งหมด รวมถึงทำการทดลองสอบเทียบเพื่อแปลงสัญญาณ Output เป็นค่าหน้างานในหน่วย SI Derived แบบ Real Time
- เพื่อวิเคราะห์ปรากฏการณ์ Saturation ที่เกิดขึ้นในระบบขยายสัญญาณ และอธิบายสาเหตุของความคลาดเคลื่อนที่ทำให้ค่าที่วัดได้ไม่ตรงกับเครื่องชั่ง Digital
- เพื่อแสดงผลการเปรียบเทียบค่าที่ได้กับเครื่องชั่ง Digital อย่างชัดเจนและมีความสมเหตุสมผลของผลการทดลอง

#### สมมติฐาน

ระบบขยายสัญญาณโดยใช้ INA125 ที่มีการคำนวณ และปรับค่า Gain ที่เหมาะสม จะสามารถแปลงสัญญาณจาก Load Cell ให้อยู่ในyanการวัดของ ADC ได้อย่างมีประสิทธิภาพ การดำเนินการ Calibration อย่างถูกต้องจะทำให้สามารถแสดงสัญญาณ Output เป็นค่าหน้างานในหน่วย SI Derived ได้แบบ Real Time ด้วยความคลาดเคลื่อนที่สามารถอธิบายแหล่งที่มาได้เมื่อเปรียบเทียบกับเครื่องชั่ง Digital

#### ตัวแปร

- ตัวแปรต้น:
  - น้ำหนักที่กดบน Load Cell (kg)
- ตัวแปรตาม:
  - แรงดันขาออก  $V_{out}$  ของ INA125 ที่น้ำหนักที่เปลี่ยนแปลงไป
- ตัวแปรควบคุม:
  - แรงดันไฟฟ้าเลี้ยงวงจร
  - ตำแหน่งการวางถุงทราย
  - ค่าความต้านทาน  $R_G$

#### เอกสารและงานวิจัยที่เกี่ยวข้อง

ส่วนนี้จะกล่าวถึงทฤษฎีพื้นฐานของ Load Cell, Strain Gauge, วงจร Wheatstone Bridge และ Instrumentation Amplifier (INA125) ซึ่งเป็นความรู้ที่จำเป็นในการวิเคราะห์ผลการทดลองและการทำงานของระบบโดยรวม

## หลักการทำงานของ Load Cell และ Strain Gauge

- Load Cell เป็นอุปกรณ์ที่ทำหน้าที่แปลงแรง หรือน้ำหนักที่มีการกระทำ ให้เป็นสัญญาณไฟฟ้าที่วัดได้
- การทำงานอาศัยหลักการของ Strain Gauge ซึ่งเป็นตัวด้านท่านที่ค่าความต้านทานจะเปลี่ยนแปลงไปตามสัดส่วนของแรงที่มีการกระทำต่อโครงสร้างของ Load Cell ที่เกิดการเสียรูป (ยืดหรือหด)
- การเปลี่ยนแปลงความต้านทานนี้เป็นไปในสัดส่วนที่น้อยมาก.

## วงจร Wheatstone Bridge (Full Bridge Configuration)

วงจร Wheatstone Bridge ทำหน้าที่แปลงการเปลี่ยนแปลงความต้านทานที่เล็กน้อยของ Strain Gauge ให้เป็นสัญญาณแรงดันไฟฟ้าที่สามารถวัดได้ เมื่อไม่มีแรงมากระทำ วงจรส่ายู่ในสภาวะสมดุล และให้แรงดันไฟฟ้า Output เป็น 0V

### ข้อดีของ Full Bridge Configuration:

- ให้ความไวในการวัดสูงสุด (Highest Measurement Sensitivity) เนื่องจากมี Strain Gauge ที่ทำงานอยู่ (Active Gages) ทั้งสี่แขนของวงจร
- มีความสามารถในการชดเชยอุณหภูมิ ที่ดีที่สุด เพราะการเปลี่ยนแปลงอุณหภูมิจะส่งผลต่อ Strain Gauge ทั้งสี่ตัวเท่าๆ กัน ทำให้ข้อผิดพลาดที่เกิดจากอุณหภูมิถูกหักล้างไปโดยธรรมชาติของวงจร
- วงจร Full Bridge ให้ความสมพันธ์ระหว่างแรงที่กระทำกับแรงดันไฟฟ้า Output เป็นแบบ เชิงเส้น (Linearity) โดยตรง (ไม่ต้องมีการประมาณค่า) ซึ่งให้ความแม่นยำสูงกว่างจร Quarter Bridge หรือ Half Bridge

## Instrumentation Amplifier (INA125) และการขยายสัญญาณ

- สัญญาณแรงดันไฟฟ้าข้ออกที่ได้จาก Load Cell มีค่าต่ำมาก (โดยทั่วไปอยู่ในหน่วยมิลลิโวลต์) ดังนั้น จึงจำเป็นต้องใช้ Instrumentation Amplifier ในการขยายสัญญาณ
- INA125 เป็น Instrumentation Amplifier ที่มีความแม่นยำสูง และมีโครงสร้างแบบ 2 Op-amps Differential Amplifier ซึ่งเหมาะสมสำหรับการวัดสัญญาณ Differential ขนาดเล็กจาก Load Cell
- 4. การกำหนดค่า Gain ใช้ตัวด้านท่านภายนอกเพียงตัวเดียว ( $R_G$ ) ในการกำหนดค่า Gain ซึ่งสามารถปรับได้ตั้งแต่ 4 ถึง 10,000 สมการที่ใช้ในการคำนวณหาค่า Gain ( $G$ ) คือ  $G = 4 + \frac{60k\Omega}{R_G}$
- 5. การคำนวณหา  $R_G$  ที่เหมาะสมเป็นสิ่งสำคัญ เพื่อให้มั่นใจว่าสัญญาณที่ขยายแล้วจะไม่เกินขีดจำกัดของแหล่งจ่ายไฟ และอยู่ในย่านที่ ADC ของบอร์ด STM32G474RE สามารถอ่านค่าได้อย่างเต็มประสิทธิภาพโดยไม่เกิด Saturation

## ขั้นตอนการดำเนินงาน

1. ปรับค่า  $R_G$  ให้ตรงตามทฤษฎี จากนั้นเก็บผลการทดลอง
2. ปรับค่า  $R_G$  ด้วยตนเอง เพื่อให้เกิดการ Saturate ที่น้ำหนัก 10 kg (ถุงทราย 20 ถุง)
3. เริ่มทำการวางแผนทรายที่ละ 1 ถุง พร้อมเก็บค่า  $V_{out}$  ของแต่ละน้ำหนัก
4. ทำการทดลองซ้ำทั้งหมด 5 รอบ จากนั้นหาค่าเฉลี่ย และนำมารวเคราะห์ข้อมูล

## ผลการทดลอง

นิยามคำว่า Gain ที่เหมาะสม คือสามารถถ่วงได้ 10 kg พอดีโดย  $V_{out}$  จะต้องไม่ Saturate ที่ค่าใดค่าหนึ่ง จากการทดลองคันப์ว่า  $V_{out}$  สูงสุดที่สามารถออกมากได้คือประมาณ 2.5 V จากการคำนวณตามทฤษฎี

$$V_{out} = (V_{in}^+ - V_{in}^-)G$$

ทดลองใส่น้ำหนัก 10 kg และวัด  $V_{in}^+$ ,  $V_{in}^-$  และ  $V_{out}$  จะได้

$$2.500 = (1.220 - 1.215)G$$

จะได้  $G = 500$  นำ  $G$  ที่ได้นำไปคำนวณ  $R_G$

$$G = 4 + \frac{60\text{k}\Omega}{R_G}$$

$$500 = 4 + \frac{60\text{k}\Omega}{R_G}$$

$$R_G = 120.96 \Omega$$

จากนั้นคำนวณหา ADC ที่ควรเพิ่มขึ้นต่อน้ำหนักถุง เนื่องจากเราใช้ STM32 Output ออกมาเป็น 12 bit ซึ่งก็คือ 0-4095 bytes แรงดันสูงสุด 3.3 V และเนื่องจากค่า  $V_{out}$  คือ 2.5 V เพราะฉะนั้น ADC Max ที่ควรจะได้ คือ

$$ADC = \frac{V_{out}}{3.3} \times 4095$$

$$ADC = \frac{2.5}{3.3} \times 4095$$

$$ADC \text{ Max} = 3102.27 \text{ byte}$$

ค่าที่ควรจะได้ ที่ 20 ถุง (ถุงละ 500 กรัม) คือ  $3102.27 \div 20 = 155.11$  ต่อถุง แต่เมื่อนำไปทดลองด้วยการปรับ  $R_G$  เป็น 120.96 พบทว่าที่ 1 ถุง ADC ไม่ได้เพิ่มขึ้น 155.11 ต่อถุง จึงปรับค่า Trim pot เป็นค่าที่ถูกต้องโดยการปรับด้วยมือ ใส่น้ำหนักที่ 10 kg เมื่อเอารอก 1 ถุงควรจะลด 155.11 พอดี สรุปค่าที่ได้คือ ประมาณ 103Ω และ Gain จะได้ประมาณ 586.52

เพราะฉะนั้นที่น้ำหนัก 10 kg =  $155 \times 10 = 1550$  + Offset 30 = 1580 และมีค่า Saturate ที่ประมาณ 3102.27 bytes พอดี สรุปว่า Gain ที่เหมาะสมคือ 586.52 นำ Gain ที่ได้ไปทดลองซึ่งน้ำหนักเพิ่มทีละ 0.5 kg ทดลองซ้ำน้ำหนักละ 5 รอบ วัด  $V_{in}^+$ ,  $V_{in}^-$  และ  $V_{out}$  บันทึกค่าที่อ่านได้จาก ADC เพื่อเปรียบเทียบ

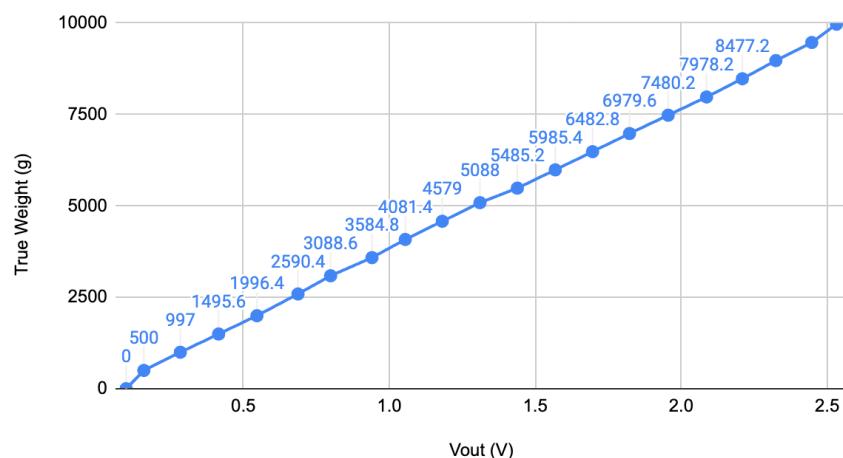
ตารางสรุปค่าเฉลี่ยที่ได้จากการทดลองซ้ำ 5 ครั้ง

จำนวนถุง (ถุง)	น้ำหนักที่ควรเป็น (g)	น้ำหนักที่ซึ่งได้จริง (g)	ค่าเฉลี่ย $V_{out}$ (V)
0	0	0	0.0986
1	500	500	0.1592
2	1000	997	0.2842
3	1500	1495.6	0.4154
4	2000	1996.4	0.5466
5	2500	2590.4	0.6872

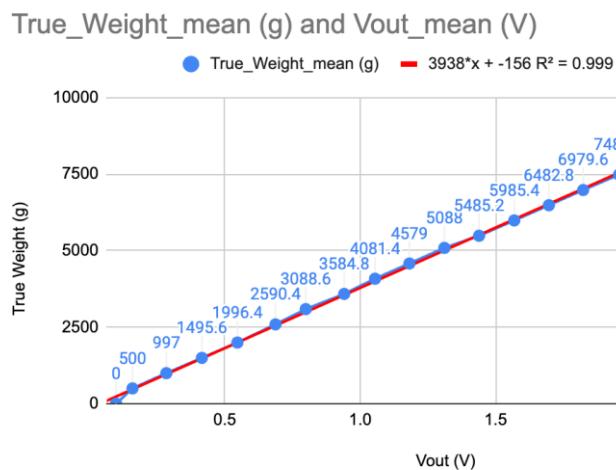
6	3000	3088.6	0.799
7	3500	3584.8	0.9408
8	4000	4081.4	1.0548
9	4500	4579	1.1816
10	5000	5088	1.3108
11	5500	5485.2	1.4388
12	6000	5985.4	1.5688
13	6500	6482.8	1.6966
14	7000	6979.6	1.8232
15	7500	7480.2	1.9556
16	8000	7978.2	2.087
17	8500	8477.2	2.2098
18	9000	8973.6	2.3246
19	9500	9469.2	2.4476
20	10000	9969.4	2.5322

เมื่อนำมาสร้างกราฟแสดงน้ำหนักจริง เทียบกับ  $V_{out}$  จะได้ดังนี้

True\_Weight\_mean (g) and Vout\_mean (V)



จากนั้นนำมาสร้าง Trend Line จะได้กราฟดังนี้

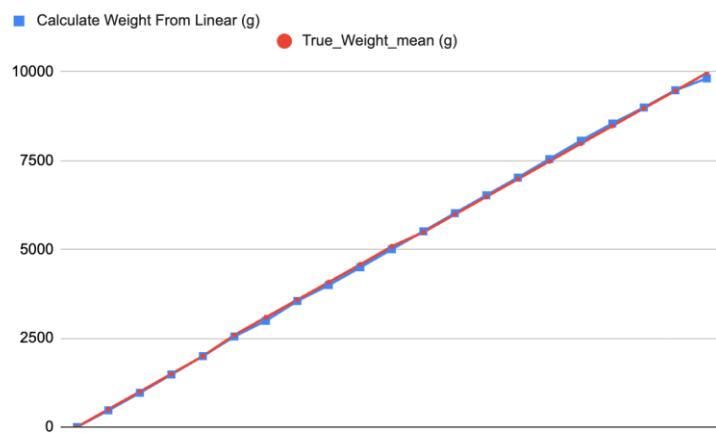


นำสมการที่ได้้นำมาคำนวณเพื่อหาค่าน้ำหนัก ดังตารางต่อไปนี้

น้ำหนักที่ควรจะเป็น (g)	ค่าเฉลี่ย V <sub>out</sub> (V)	น้ำหนักที่คำนวณได้จากการเชิงเส้น $W = 3938 \times V_{out} \pm 156$ (g)	ค่าเฉลี่ยน้ำหนักที่วัดได้จริง (g)	Error %
0	0.0986	0	0	0%
500	0.1592	470.9296	500	5.81%
1000	0.2842	963.1796	997	3.39%
1500	0.4154	1479.8452	1495.6	1.05%
2000	0.5466	1996.5108	1996.4	0.01%
2500	0.6872	2550.1936	2590.4	1.55%
3000	0.799	2990.462	3088.6	3.18%
3500	0.9408	3548.8704	3584.8	1.00%
4000	1.0548	3997.8024	4081.4	2.05%
4500	1.1816	4497.1408	4579	1.79%
5000	1.3108	5005.9304	5088	1.61%
5500	1.4388	5509.9944	5485.2	0.45%
6000	1.5688	6021.9344	6482.8	0.61%
6500	1.6966	6525.2108	6979.6	0.65%
7000	1.8232	7023.7616	8477.2	0.63%

7500	1.9556	7545.1528	7480.2	0.87%
8000	2.087	8062.606	7978.2	1.06%
8500	2.2098	8546.1924	8477.2	0.81%
9000	2.3246	8998.2748	8973.6	0.27%
9500	2.4476	9482.6488	9469.2	0.14%
10000	2.5322	9815.8036	9969.4	1.54%

กราฟแสดงการเปรียบเทียบน้ำหนักที่ซึ้งได้จริง และน้ำหนักที่คำนวณผ่านสมการเส้นตรง

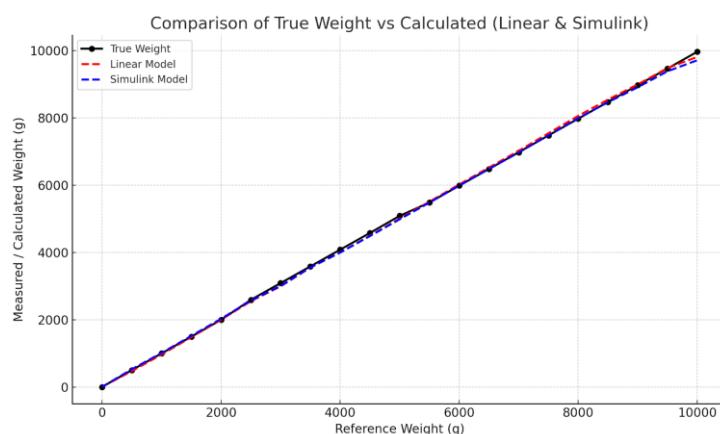


จากนี้วิเคราะห์ Linearity ของ Load cell จากค่าที่อ่านได้จาก Simulink เนื่องจากข้อมูลที่ได้ไม่สามารถนำไปคำนวณหาสมการเส้นตรงให้เราจึงต้องทำการ preprocess ข้อมูล โดยการกรองสัญญาณ แบ่งสัญญาณ เป็น 21 ช่วง เนื่องจากแต่ละช่วง รวมไฟล์ด้วยค่ามัธยฐาน จากนั้น Fit เส้นตรง  $W = (a \times ADC) + b$  และ ตรวจสอบเส้น ( $R^2$ ) ได้ดังตารางต่อไปนี้

น้ำหนักที่ควรจะเป็น (g)	ค่าเฉลี่ยน้ำหนักที่วัดได้จริง (g)	น้ำหนักที่คำนวณได้จากสมการเชิงเส้น $W = 3938 \times V_{out} \pm 156$ (g)	น้ำหนักที่คำนวณได้จาก Simulink $W = 3.121 \times ADC - 90.3$ (g)	Error % Linear	Error % Simulink
0	0	0	0	0	0
500	500	470.9296	526.259792	5.81%	5.25%
1000	997	963.1796	1010.38307	3.39%	1.34%
1500	1495.6	1479.8452	1518.488112	1.05%	1.53%
2000	1996.4	1996.5108	2026.61188	0.01%	1.51%
2500	2590.4	2550.1936	2571.145234	1.55%	0.74%

3000	3088.6	2990.462	3004.121564	3.18%	2.74%
3500	3584.8	3548.8704	3553.298966	1.00%	0.88%
4000	4081.4	3997.8024	3994.80811	2.05%	2.12%
4500	4579	4497.1408	4485.884976	1.79%	2.03%
5000	5088	5005.9304	4986.262422	1.61%	2.00%
5500	5485.2	5509.9944	5482.002062	0.45%	0.06%
6000	5985.4	6021.9344	5985.469298	0.61%	0.00%
6500	6482.8	6525.2108	6480.428688	0.65%	0.04%
7000	6979.6	7023.7616	6970.725304	0.63%	0.13%
7500	7480.2	7545.1528	7483.499362	0.87%	0.04%
8000	7978.2	8062.606	7992.397138	1.06%	0.18%
8500	8477.2	8546.1924	8467.987602	0.81%	0.11%
9000	8973.6	8998.2748	8912.586536	0.27%	0.68%
9500	9469.2	9482.6488	9388.951008	0.14%	0.85%
10000	9969.4	9815.8036	9716.593588	1.54%	2.54%

ได้น้ำหนักจริงเทียบกับน้ำหนักที่คำนวณกับ Simulink ดังกราฟต่อไปนี้



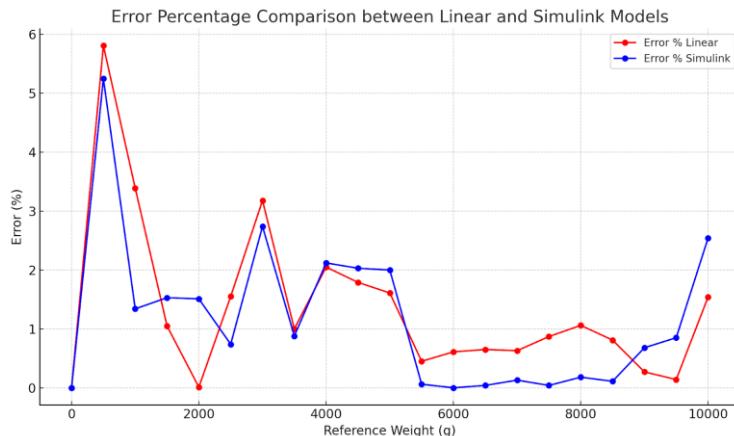
## สรุปผลการทดลอง

เมื่อทดลองปรับค่า Gain มากขึ้นคันพบความสัมพันธ์ว่า ยิ่ง Gain เยอะจะยิ่งทำให้ Saturate เร็วขึ้น เพราะฉะนั้นเราจึงต้องหา Gain ที่เหมาะสมที่จะสามารถวัดใน Range 10 kg ได้

จะได้สมการเส้นตรงสำหรับการวัดจริงของ Load cell คือ  $y = 3938x \pm 156$  โดยที่  $x$  คือ  $V_{out}$  (V)

และแกน Y คือ น้ำหนัก (g) ค่า  $R^2$  มีค่า 0.99 หมายความว่าข้อมูล Fit กับสมการเกือบจะสมบูรณ์แบบ ทำการกำหนด ค่าเริ่มต้นเป็น 0 จากเนื่องจากมีค่า off set เริ่มต้นทำให้มีอิทธิพลเป็นน้ำหนักมีค่าประมาณ 200g ซึ่งไม่ถูกต้องจึงกำหนดค่าเริ่มต้นเท่ากับ 0 เพื่อให้ตรงตามความเป็นจริง

จากการทดลองหา Linearity ของ Load cell ผ่านการวิเคราะห์ต่างๆ สำหรับการอ่านค่าจาก Simulink ได้สมการ คือ  $\text{Weight} = 3.121 \times \text{ADC} - 90.3$  และ ค่าที่อ่านได้จริงสมการ คือ  $\text{Weight} = 3938 \times V_{\text{out}} - 156$



จากภาพข้างต้นสามารถอธิบายได้ว่า

เกณฑ์	Linear	Simulink	สิ่งที่ดี
Mean Error %	1.41	1.35	Simulink
Max Error %	5.81	5.25	Simulink
Std Dev %	1.43	1.41	Simulink
Average Error %	1.36	1.18	Simulink
$R^2$	0.9996	0.9997	Simulink
เสถียรภาพในช่วงกลาง	ดี	ดีมาก	Simulink

## อภิปรายผล

จากการทดลอง Load Cell พบว่าแรงดันเอาต์พุต ( $V_{\text{out}}$ ) มีความสัมพันธ์เชิงเส้นกับน้ำหนักที่กระทำต่อเซนเซอร์ ซึ่งสอดคล้องกับหลักการของวงจร Wheatstone Bridge ที่ใช้แปลงการเปลี่ยนแปลงความต้านทานของ Strain Gauge ให้เป็นแรงดันไฟฟ้าที่สามารถขยาย และวัดได้

เมื่อทำการปรับค่า  $\text{Gain} = 586.52$  ซึ่งเป็นค่าที่เหมาะสมต่อช่วงการวัด 0–10 kg พบว่าแรงดันเอาต์พุต ไม่เข้าสู่ภาวะอิมตัว (Saturation) และให้สัญญาณที่คงที่และเป็นเชิงเส้นดี ผลจากการวัดจริงได้สมการเชิงเส้น  $\text{Weight} = 3938 \times V_{\text{out}} - 156$  โดยมีค่า  $R^2 = 0.99$  แสดงว่าความสัมพันธ์ระหว่างน้ำหนักกับแรงดันมีความเป็นเชิงเส้นสูงมาก ขณะที่ข้อมูลจากการจำลองด้วย Simulink (ADC Output) ได้สมการ  $\text{Weight} = 3.121 \times \text{ADC} -$

90.3 และมี  $R^2 = 0.9997$  ชี้งบอกได้ว่าสัญญาณของ ADC มีความถูกต้อง และสอดคล้องกับผลการวัดจริงอย่างมาก โดยมีค่าเฉลี่ยความคลาดเคลื่อนเพียง 1.18% และค่าสูงสุดไม่เกิน 5.25%

วิเคราะห์ความคลาดเคลื่อน Offset Error ที่น้ำหนัก 0 g พบว่า  $V_{out}$  ประมาณ 0.0986V ซึ่งไม่เป็น 0 แสดงว่าระบบมีการ Offset เกิดขึ้น ทำให้ต้องกำหนดค่าเริ่มต้นเป็น 0 เท่านั้น ในสมการเชิงเส้น สาเหตุอาจมาจากการไม่สมดุลของค่าความต้านทานใน Wheatstone Bridge, ค่า Input Offset Voltage ของ Op-Amp (INA125), แรงกดคงค้างใน Load Cell (Mechanical preload), การเปลี่ยนแปลงอุณหภูมิ และสัญญาณรบกวนระหว่างการวัด

ในช่วงน้ำหนัก 1–10 kg ค่า  $V_{out}$  และ ADC เพิ่มขึ้นอย่างสม่ำเสมอ โดยไม่พบ Saturation ก่อนถึงค่าการวัดสูงสุด แสดงว่า Gain และการปรับ  $R_g$  เหมาะสมกับช่วงการวัด สำหรับความแตกต่างระหว่างค่าที่วัดจริง และ Simulink แม้ค่า Simulink จะมีแนวโน้มสูงกว่าเล็กน้อย แต่ความสัมพันธ์โดยรวมใกล้เคียงกันมาก แสดงว่า แบบจำลอง ADC–Voltage มีความน่าเชื่อถือสูง

### ข้อเสนอแนะ

6. ใช้ *Shielded Cable* ลด noise จากสัญญาณ mV
7. ใช้การกรองสัญญาณแบบ *Butterworth 3rd Order* เพื่อลด noise ได้ดียิ่งขึ้น
8. ควรใช้แรงดันไฟฟ้าเข้าคองที่ และตรวจสอบ offset ก่อนเริ่มหาดลอง (*tare* ศูนย์น้ำหนัก)

### อ้างอิง (สีแลค Link)

<https://www.ti.com/lit/ds/symlink/ina125.pdf>

[https://resources.altium.com/p/wheatstone-bridges?utm\\_source=chatgpt.com](https://resources.altium.com/p/wheatstone-bridges?utm_source=chatgpt.com)

[https://www.analogictips.com/wheatstone-bridge-part-1-principles-and-basic-applications/?utm\\_source=chatgpt.com](https://www.analogictips.com/wheatstone-bridge-part-1-principles-and-basic-applications/?utm_source=chatgpt.com)

# **ภาคผนวก**

## Lab 1.1 Potentiometer

อุปกรณ์ที่ใช้ในการเก็บข้อมูลของ Linear potentiometer



ภาพด้านบนแสดงอุปกรณ์ที่ใช้ในการเก็บข้อมูลของ Linear Potentiometer ซึ่งติดตั้งอยู่บนฐานทดลอง PotenXplorer ร่วมกับบอร์ด STM32 Nucleo G474RE โดยเชื่อมต่อขา Signal → PA1 (ADC2) เพื่ออ่านค่าแรงดันไฟฟ้าแบบ Real-Time ผ่าน Simulink

ในชุดทดลองนี้ ใช้สเกลระยะทางจำนวน 13 ช่อง โดยแต่ละช่องมีระยะห่าง 1 เซนติเมตร รวมช่วงการวัดตั้งแต่ 0 – 12 เซนติเมตร

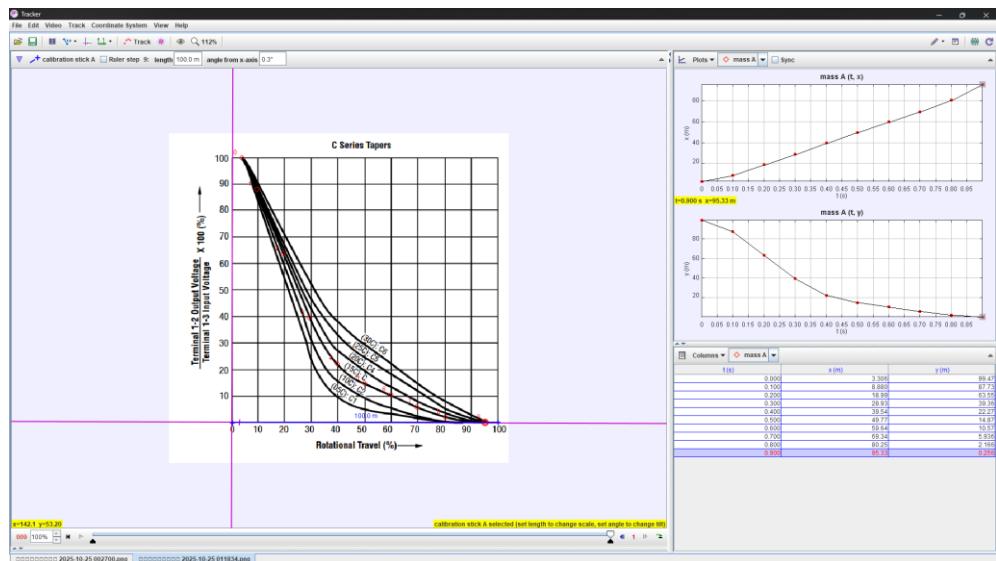
ผู้ทดลองทำการเลื่อนตัวเลื่อน (slider) ของ Potentiometer ที่ละช่องเพื่อบันทึกแรงดันไฟฟ้า ( $V_{out}$ ) ที่สัมพันธ์กับตำแหน่งทางกล

วิธีที่ใช้ในการได้ค่าของ Datasheet มา plot เปรียบเทียบกับค่าที่เราเก็บได้

การเปรียบเทียบค่าจากการทดลองกับข้อมูลจาก Datasheet เพื่อประเมินความเป็นเชิงเส้น (Linearity) ของ Linear Potentiometer ผู้ทดลองได้นำข้อมูลจาก Datasheet ของผู้ผลิต มาวิเคราะห์เปรียบเทียบกับค่าที่วัดได้จริง ขั้นตอนมีดังนี้:

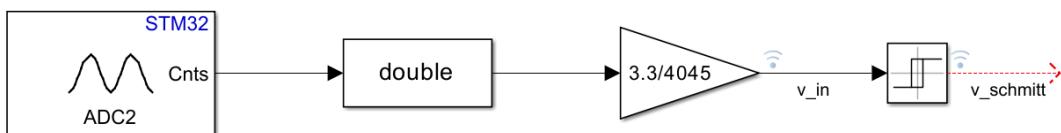
1. เปิดกราฟคุณลักษณะทางไฟฟ้า (Electrical Characteristic Curve) จาก Datasheet
2. ใช้โปรแกรม Tracker ทำการ Calibrate แกนกราฟ ให้ตรงกับหน่วยแรงดัน (V) และระยะทาง (%)
3. ทำการ Mark จุดตัวอย่าง จากเส้นโค้งมาตรฐานของ Datasheet
4. นำค่าที่ได้มาสร้างกราฟเปรียบเทียบกับค่าที่วัดจากการทดลองจริงใน MATLAB

ผลลัพธ์ทำให้สามารถวิเคราะห์ความแตกต่างระหว่างค่าที่วัด ได้กับค่าทางทฤษฎีได้อย่างชัดเจน และตรวจสอบความเป็นเชิงเส้นของ Linear Potentiometer แต่ละตำแหน่งได้อย่างถูกต้อง



## โปรแกรม Schmitt Trigger

Rotary Potentiometer Type B  
(connect analog pin to PA1)



วัตถุประสงค์ของโปรแกรมนี้คือการสร้างวงจรจำลองที่สามารถตรวจจับการเปลี่ยนแปลงของแรงดันไฟฟ้า จาก Potentiometer และแปลงให้อยู่ในรูปของสัญญาณดิจิทัล ตามหลักการของวงจร Schmitt Trigger ที่มี Hysteresis เพื่อความเสถียรของสัญญาณ Output

### ADC2 (STM32 Block)

- อ่านสัญญาณ Analog จาก Potentiometer ผ่านขา PA1 ของ STM32 ซึ่งเป็นขา ADC (Analog-to-Digital Converter)
- ค่าที่อ่านได้จะอยู่ในรูปของ Counts (0 – 4095) ซึ่งเป็นค่าดิจิทัล 12 บิต จาก ADC2

### Double Conversion Block

- แปลงค่าที่อ่านได้จากชนิดข้อมูล uint16 หรือ int ให้เป็น double เพื่อให้สามารถนำไปคำนวณทางคณิตศาสตร์ได้ใน Simulink
- ไม่มีการเปลี่ยนค่า เพียงแต่เปลี่ยนชนิดข้อมูล

### Gain Block (3.3 / 4045)

- ใช้สมการ

$$V_{in} = \frac{3.3}{4095} \times \text{Counts}$$

- เพื่อแปลงค่าจำนวน Counts ให้เป็นแรงดันไฟฟ้าในหน่วย โวลต์ ( V )
- ผลลัพธ์คือแรงดันจริงของ Potentiometer ในช่วง 0 – 3.3 V ซึ่งส่งออกเป็นตัวแปร v\_in

### Schmitt Trigger Subsystem

- บล็อกนี้คือหัวใจหลักของโปรแกรม ใช้หลักการของวงจร Comparator ที่มี Hysteresis เพื่อสร้างขอบเขตการเปลี่ยนสถานะ
- ภายในกำหนดค่า Upper Threshold (VUT) และ Lower Threshold (VLT) ดังนี้

$$V_{UT} = 0.8 \times 3.3 = 2.64V, V_{LT} = 0.2 \times 3.3 = 0.66V$$

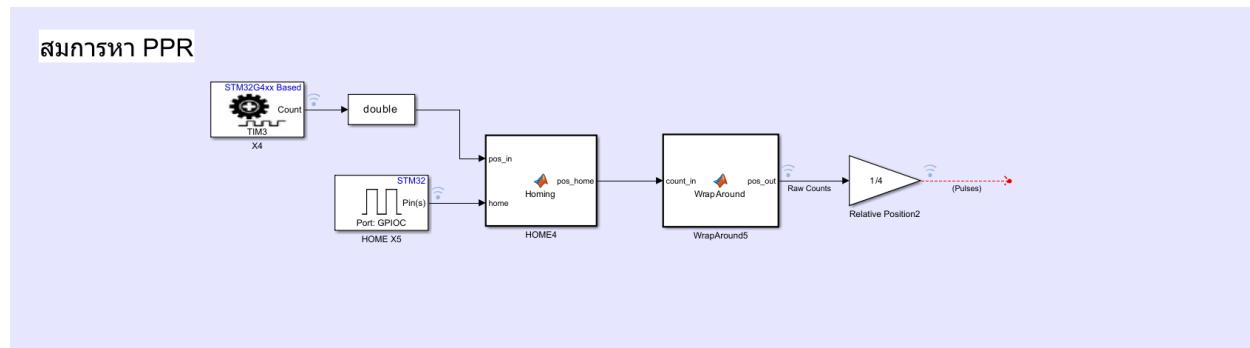
- เมื่อแรงดัน Vin สูงกว่า VUT สัญญาณ Output จะเปลี่ยนเป็น High (1 หรือ 3.3 V) และเมื่อแรงดัน Vin ต่ำกว่า VLT จะเปลี่ยนเป็น Low (0 V)
- ในช่วงที่ Vin อยู่ระหว่าง VLT–VUT สัญญาณจะคงสถานะเดิมไว้
- ส่งผลให้สัญญาณ v\_schmitt เป็นรูป Square Wave หรือ Digital Output ที่เสถียรและไม่ไวต่อ Noise

### หลักการทำงานโดยรวม

- ผู้ใช้หมุน Rotary Potentiometer เพื่อเปลี่ยนแรงดันขาออก Vout
- STM32 อ่านค่าแรงดันนี้เป็น ADC Counts (0 – 4095)
- Simulink คำนวณแปลงค่าเป็นแรงดันจริง (Vin) และส่งเข้าสู่ Schmitt Trigger Subsystem
- Schmitt Trigger ตรวจสอบระดับแรงดันเทียบกับ VUT และ VLT แล้วสร้างสัญญาณ Digital Output (v\_schmitt)
- ผลลัพธ์คือการแปลงสัญญาณ Analog → Digital แบบ Real-Time แสดงบน Scope หรือ Data Inspector

## Lab 1.2 Incremental Encoder

### โปรแกรมคำนวณ PPR (Pulses Per Revolution)



ภาพด้านบนแสดงแบบจำลองการทำงานของระบบอ่านค่าจาก Incremental Encoder เพื่อคำนวณจำนวนพัลส์ต่อรอบการหมุน (PPR) โดยใช้บอร์ด STM32G474RE ผ่านการสร้างแบบจำลองใน Simulink ซึ่งแสดงการรับสัญญาณ การประมวลผล และการแปลงให้อยู่ในหน่วยพัลส์อย่างเป็นระบบ

### หลักการทำงานโดยรวม

ระบบที่ทำหน้าที่ อ่านสัญญาณ Encoder จากแกนหมุน, ประมวลผลเพื่อหาตำแหน่งเชิงสัมพันธ์ (Relative Position) และคำนวณจำนวนพัลส์ต่อหนึ่งรอบ (PPR) โดยใช้การนับสัญญาณจาก Quadrature Encoder ผ่านช่องสัญญาณ A และ B ด้วยการอ่านแบบ X4 Mode เพื่อให้มีความละเอียดสูงสุด

### อธิบายการทำงานของแต่ละบล็อก

ลำดับ	ชื่อบล็อก	รายละเอียดการทำงาน
1	<b>TIM3 (STM32G4xx Based)</b>	เป็นบล็อกหลักที่ใช้ Timer 3 (TIM3) ของ STM32 ในโหมด Encoder Interface เพื่ออ่านค่าจำนวนพัลส์จากสัญญาณ Quadrature (ช่อง A และ B) ของ Encoder โดยการตั้งค่าในโหมด X4 ทำให้สามารถนับได้ทั้งข้อมูลขึ้นและขอลองของสัญญาณ A และ B (1 รอบ = 4 พัลส์) ส่งผลให้ความละเอียดเพิ่มขึ้น 4 เท่า
2	<b>Double Conversion Block</b>	ทำหน้าที่แปลงข้อมูลที่ได้จาก TIM3 ซึ่งเป็นชนิดข้อมูล Unsigned Integer (uint16) ให้เป็นชนิด Double Precision (double) เพื่อให้สามารถนำไปคำนวณหรือดำเนินการทางคณิตศาสตร์ได้ใน Simulink อย่างถูกต้อง
3	<b>HOME X5 (GPIO Input)</b>	เป็นสัญญาณเดิจิทัลจากสวิตซ์โฮม (Home Sensor) ที่ต่อเข้าขา GPIOC ของ STM32 โดยใช้สำหรับกำหนด “จุดเริ่มต้น” หรือ ตำแหน่งอ้างอิง (Home Position) ของการหมุน เมื่อสวิตซ์ถูกกดจะส่งสัญญาณลงจิก “1” กลับมาที่บล็อก Homing
4	<b>Homing Subsystem (HOME4)</b>	บล็อกนี้รับสัญญาณตำแหน่งปัจจุบัน (pos_in) และสัญญาณ Home (home) เพื่อตรวจสอบว่า มีการเข้าจุดอ้างอิงหรือไม่ เมื่อพบว่า Home Sensor ถูกทริกเกอร์ ระบบจะรีเซ็ตค่าตำแหน่งให้เป็นศูนย์ (pos_home = 0) เพื่อใช้เป็นจุดเริ่มต้นของการคำนวณพัลส์รอบต่อไป

ลำดับ	ชื่อบล็อก	รายละเอียดการทำงาน
5	<b>WrapAround Subsystem (WrapAround5)</b>	ใช้สำหรับจัดการค่าการนับที่เกิดการ “วนกลับ (overflow)” ของ Timer เมื่อค่าพัลส์นับเกินขอบเขต 16-bit หรือ 32-bit เช่น จาก 65535 → 0 บล็อกนี้จะตรวจสอบและปรับค่าตำแหน่ง (pos_out) ให้ต่อเนื่องโดยอัตโนมัติ ป้องกันไม่ให้เกิดการกระโดดของข้อมูล
6	<b>Gain Block (1/4)</b>	ทำหน้าที่แปลงค่าพัลส์ที่อ่านได้ในโหมด X4 ให้กลับมาเป็นค่าพัลส์จริงต่อหนึ่งรอบ โดยใช้สมการ

### สมการที่ใช้ในโปรแกรม

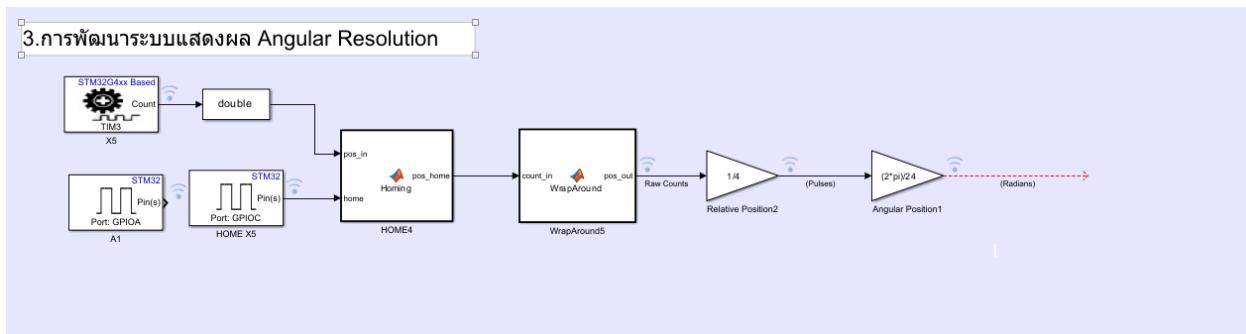
- การคำนวณจำนวนพัลส์ต่อรอบ (PPR):

$$PPR = \frac{\text{Raw Counts per Revolution}}{4}$$

- การคำนวณความละเอียดเชิงมุม (Angular Resolution):

$$\text{Resolution} = \frac{360^\circ}{PPR}$$

### โปรแกรมคำนวณและแสดงผล Angular Resolution



ภาพด้านบนแสดง แบบจำลองใน Simulink สำหรับคำนวณและแสดงผล Angular Resolution (ความละเอียดเชิงมุม) ของ Incremental Encoder โดยใช้ข้อมูลจากเอกสารพูดพัลส์ที่อ่านได้จริงจาก STM32 โปรแกรมนี้ ต่อยอดจากแบบจำลองการคำนวณ PPR โดยเพิ่มขั้นตอนการคำนวณตำแหน่งเชิงมุม (Angular Position) และ อัตราการหมุนเชิงมุม (Angular Velocity) ตามสมการพื้นฐานของ Encoder ดังนี้

### สมการ

- Resolution (ความละเอียดเชิงมุม):**

$$res_{\theta} = \frac{2\pi}{PPR} [\text{rad/pulse}]$$

ใช้หามุ่งเชิงรัศมีที่แทนด้วยหนึ่งพัลส์ของ Encoder

## 2. Position (ตำแหน่งเชิงมุม):

$$\theta = res_{\theta} \cdot n_{pulse} [\text{rad}]$$

แปลงจำนวนพัลส์ที่นับได้ให้เป็นมุมเชิงรัศมี

## 3. Velocity (ความเร็วเชิงมุม):

$$\omega = res_{\theta} \cdot \frac{d(n_{pulse})}{dt} [\text{rad/s}]$$

หาความเร็วเชิงมุมจากการเปลี่ยนแปลงของพัลส์ตามเวลา

## อธิบายการทำงานของแต่ละบล็อก

ลำดับ	ชื่อบล็อก	การทำงาน
1	<b>TIM3 (STM32G4xx Based)</b>	อ่านค่าพัลส์จากช่องสัญญาณ A และ B ของ Encoder ผ่าน Timer 3 ที่ตั้งค่าในโหมด Quadrature X4 ทำให้ได้ค่าความละเอียดสูงสุด
2	<b>Double Conversion Block</b>	แปลงข้อมูลจากชนิด uint16 ให้เป็น double เพื่อเตรียมพร้อมสำหรับการคำนวณทางคณิตศาสตร์ใน Simulink
3	<b>Homing Subsystem (HOME4)</b>	รับสัญญาณ home จาก Home Switch และใช้เพื่อตั้งค่าตำแหน่งเริ่มต้น (pos_home) เป็นศูนย์ทุกครั้งที่กลับมาที่ตำแหน่งอ้างอิง
4	<b>WrapAround Subsystem (WrapAround5)</b>	ตรวจสอบและแก้ไขค่าการนับที่ Overflow หรือ Underflow จาก Timer เพื่อให้สัญญาณตำแหน่งต่อเนื่องไม่กระโดด
5	<b>Gain Block (1/4)</b>	ใช้สมการ Relative Position = $\frac{\text{Raw Counts}}{4}$ เพื่อลดค่าการนับให้อยู่ในหน่วยพัลส์จริงต่อการหมุน 1 รอบ (ไม่รวมการคูณ X4)
6	<b>Gain Block (<math>2\pi / 24</math>)</b>	ใช้สมการ $\theta = res_{\theta} \cdot n_{pulse}$ โดยกำหนด PPR = 24 จึงได้อัตราส่วน $\frac{2\pi}{24} = 0.2618 \text{ rad/pulse}$ เพื่อแปลงจำนวนพัลส์ให้เป็นมุมเชิงรัศมี (Radians)
7	<b>Output Port (Angular Position1)</b>	แสดงผลตำแหน่งเชิงมุมแบบ Real-Time ในหน่วย [rad] โดยค่าจะเพิ่มขึ้นตามการหมุนของเพลา Encoder

## การอธิบายภาพรวมของการทำงาน

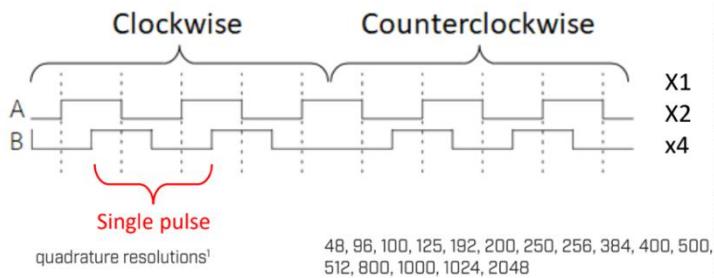
- เมื่อเพลารของ Encoder หมุน สัญญาณ A และ B ถูกส่งเข้า **Timer 3**
- Timer จะนับจำนวนพัลส์และส่งออกเป็นค่าตัวเลข (Raw Counts)
- ระบบ Homing จะรีเซ็ตตำแหน่งทุกครั้งเมื่อพบ Home Switch
- ค่าที่นับได้ถูกคำนวณกลับเป็นพัลส์จริง ( $1/4$ ) และคูณด้วยค่าคงที่  $2\pi/PPR$  เพื่อได้มุมเชิงรัศมี
- ผลลัพธ์ที่ได้สามารถนำไปคำนวณต่อเพื่อหาความเร็วเชิงมุม ( $\omega$ ) โดยใช้การหาอนุพันธ์ของตำแหน่งตามเวลา

## ภาพประกอบเชิงทฤษฎี

ภาพ Encoder ด้านบนอธิบายลักษณะของ **Quadrature Signal**

### Encoder – Position sensor

#### Basic computation of “incremental encoder”



$$\text{Resolution in rad/pulse: } res_\theta = \frac{2\pi}{PPR} \left[ \frac{\text{rad}}{\text{pulse}} \right]$$

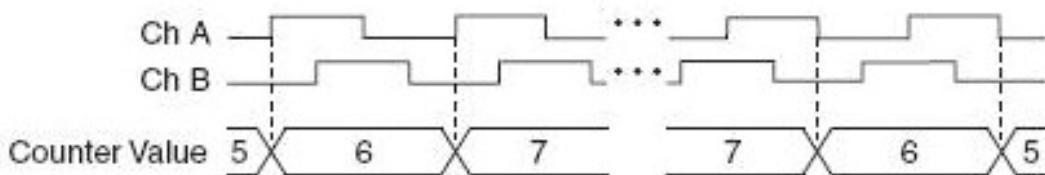
$$\text{Position in rad: } \theta = res_\theta \cdot n_{pulse} [\text{rad}]$$

$$\text{Velocity in rad/s: } \omega = res_\theta \cdot \frac{d}{dt} n_{pulse} \left[ \frac{\text{rad}}{\text{s}} \right]$$

- สัญญาณ A และ B มีเฟสต่างกัน  $90^\circ$
- เมื่อหมุนตามเข็มนาฬิกา (CW) → สัญญาณ A นำหน้า B
- เมื่อหมุนตามเข็มนาฬิกา (CCW) → สัญญาณ B นำหน้า A
- แต่ละรอบของสัญญาณเรียกว่า “1 Pulse” และเมื่ออ่านแบบ X4 จะได้ความละเอียด 4 เท่าของ PPR  
เดิม

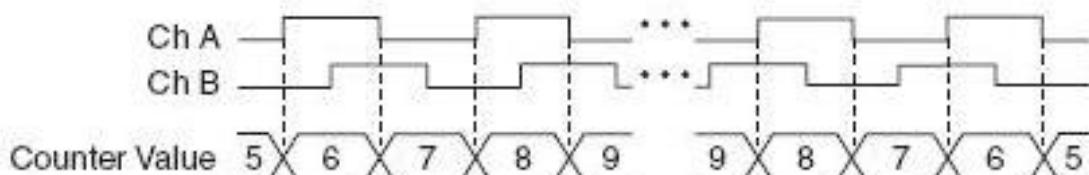
## ความแตกต่างของการอ่าน X1, X2 และ X4

### โหมด X1



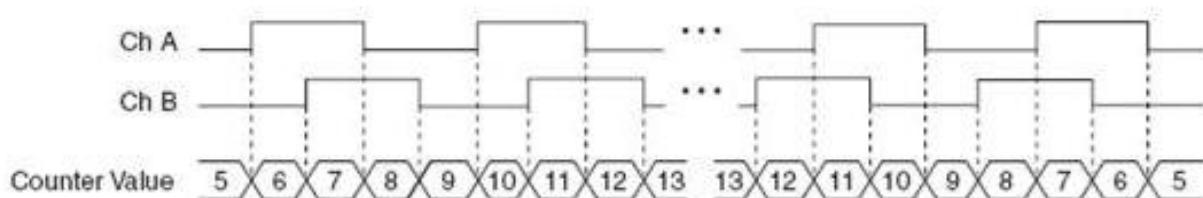
X1 สามารถอ่านได้จากการเปลี่ยนแปลงของสัญญาณ Output A หรือ Output B ก็ได้ สามารถอ่านได้จากการเปลี่ยนของสัญญาณ Output rising หรือ falling (ทำให้เราทราบองค์ที่ Encoder หมุน แต่ไม่รู้ทิศทางของการหมุน)

### โหมด X2



X2 สามารถอ่านได้จากการเปลี่ยนแปลงของสัญญาณ Output A จากการเปลี่ยนแปลงสัญญาณ Output (rising หรือ falling) พร้อมกับอ่านสัญญาณ Output B (HIGH หรือ LOW) (ทำให้เราทราบองค์ที่ Encoder หมุน และทราบทิศทางของการหมุน)

### โหมด X4

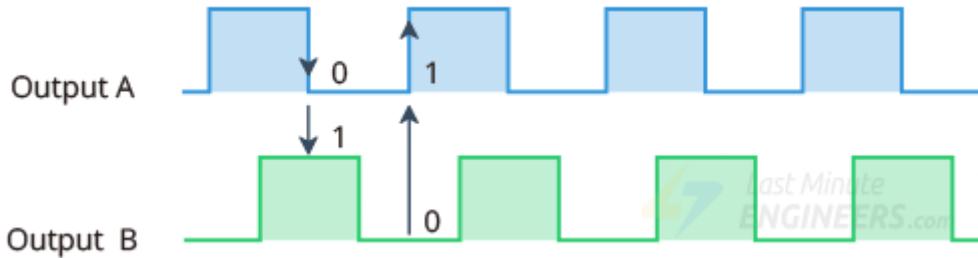


X4 สามารถอ่านได้จากการเปลี่ยนแปลงของสัญญาณ Output A และ Output B จากการเปลี่ยนแปลงสัญญาณ Output rising และ falling (ทำให้เราทราบระยะที่ Encoder หมุน และทราบทิศทางของการหมุน เพิ่มความละเอียดในการอ่านสัญญาณ)

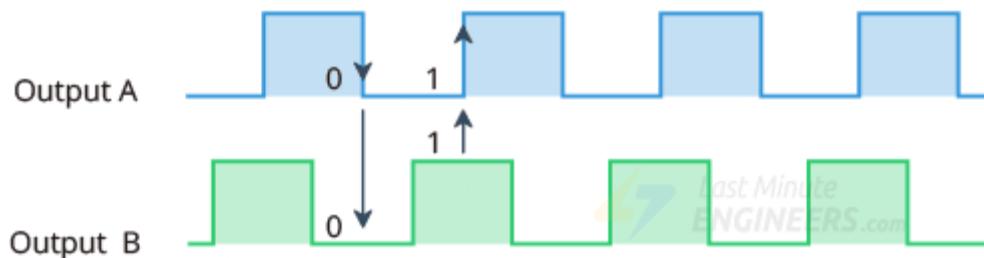
### ความแตกต่างของสัญญาณเมื่อหมุน CW และ CCW

Incremental Encoder ของเรา มี 2 สัญญาณ Output (A, B) ทำมุมห่างกัน 90 องศา ดังนั้นจึงทำให้เราทราบถึงทิศทางการหมุนของมอเตอร์ได้แน่นอน

เมื่อสัญญาณ A เปลี่ยนแปลงสถานะของสัญญาณ ถ้าสัญญาณ Pulse A ไม่เท่ากับ Pulse B ก็จะทำให้เราทราบถึงทิศทางการหมุนไปตามเข็มนาฬิกา

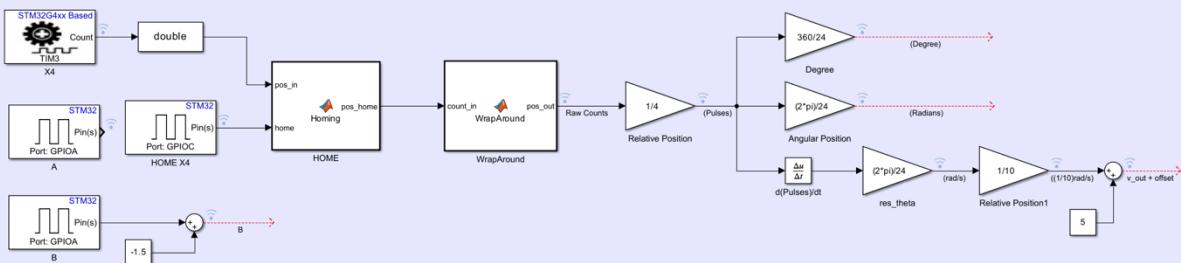


เมื่อสัญญาณ A เปลี่ยนแปลงสถานะของสัญญาณ ถ้าสัญญาณ Pulse A เท่ากับ Pulse B ก็จะทำให้เราทราบถึงทิศทางการหมุนไปทางเข็มนาฬิกา



### โปรแกรมพัฒนาระบบอ่านค่า Encoder และการแปลงหน่วยเชิงมุม

#### 3. การพัฒนาระบบอ่านค่า



ภาพด้านบนแสดงแบบจำลองใน Simulink สำหรับการพัฒนา ระบบอ่านค่า Encoder แบบ Real-Time โดยใช้บอร์ด STM32G474RE เพื่อแปลงข้อมูลดิจิทัลจาก Encoder ให้เป็นตำแหน่งเชิงมุม (Angular Position) และความเร็วเชิงมุม (Angular Velocity)

หลักการทำงานโดยรวม

- ระบบอ่านค่าพัลส์จาก Incremental Encoder (ช่อง A, B และ HOME) ผ่านบอร์ด STM32
- ประมวลผลสัญญาณให้อยู่ในรูปแบบต่อเนื่องโดยไม่สูญเสียข้อมูลเมื่อ Timer Overflow
- แปลงค่าจำนวนพัลส์ (Counts) ให้เป็น
  - Relative Position (หน่วย: Pulses)
  - Angular Position (หน่วย: Radians, Degrees)
  - Angular Velocity (หน่วย: rad/s)

4. เพิ่มพังก์ชันการซัดเซยอффเซต (Offset) เพื่อปรับศูนย์อ้างอิงของสัญญาณความเร็ว

### ตารางอธิบายการทำงานของแต่ละบล็อก

ลำดับ	ชื่อบล็อก	การทำงาน
1	<b>TIM3 (STM32G4xx Based)</b>	ใช้อ่านค่าจำนวน脉冲จาก <b>Quadrature Encoder (A และ B Channels)</b> ผ่าน <b>Timer 3</b> ซึ่งตั้งค่าในโหมด <b>X4</b> เพื่อให้ได้ความละเอียดสูงสุด โดยสามารถนับได้ทั้งข้อมูลขึ้นและขากลงของสัญญาณ A และ B
2	<b>Double Conversion</b>	แปลงข้อมูลจากชนิด <b>uint16</b> ที่ได้จาก TIM3 ให้เป็นชนิด <b>double</b> เพื่อสามารถนำไปคำนวณทางคณิตศาสตร์ใน Simulink ได้
3	<b>GPIO Blocks (A, B, HOME X4)</b>	รับสัญญาณจากขา GPIO ของ STM32 เพื่อใช้ตรวจสอบค่า Home ของ Encoder และสัญญาณ <b>Home</b> โดย <b>A/B:</b> ใช้ระบุทิศทางการหมุนและจำนวน脉冲. <b>HOME:</b> ใช้กำหนดตำแหน่งอ้างอิง (Home Position) สำหรับรีเซ็ตค่าเริ่มต้น
4	<b>Homing Subsystem (HOME)</b>	รับสัญญาณตำแหน่ง ( <b>pos_in</b> ) และสัญญาณ Home ( <b>home</b> ) เมื่อถึงตำแหน่ง Home ระบบจะรีเซ็ตตำแหน่งเป็นศูนย์ ( <b>pos_home = 0</b> ) เพื่อใช้เป็นจุดอ้างอิงของการวัด
5	<b>WrapAround Subsystem</b>	ตรวจสอบและแก้ไขปัญหา <b>Timer Overflow / Underflow</b> ของตัวนับ 16-bit (0–65535) เพื่อให้ค่าตำแหน่งต่อเนื่อง ไม่เกิดการกระโดดของข้อมูล
6	<b>Gain (1/4)</b>	แปลงค่าการนับจากโหมด X4 ให้กลับมาเป็นจำนวน脉冲จริงต่อรอบ โดยใช้สมการ: <b>Relative Position = Raw Counts / 4</b>
7	<b>Gain (360/24)</b>	แปลงจำนวน脉冲ให้เป็นหน่วยองศา (°) โดยใช้ค่า <b>PPR = 24:Degree = (360 / 24) × Pulses</b>
8	<b>Gain (2π/24)</b>	แปลงจำนวน脉冲ให้เป็นมุมเชิงรัศมี (Radians): <b>Angular Position = (2π / 24) × Pulses</b>
9	<b>Differentiator (d(Pulses)/dt)</b>	คำนวณอัตราการเปลี่ยนแปลงของ脉冲ต่อหน่วยเวลา ( $\Delta n/\Delta t$ ) เพื่อหาความเร็วเชิงมุม (Angular Velocity)
10	<b>Gain (2π/24)</b>	แปลงค่าความเร็วจาก “pulseต่อวินาที” เป็น “เรเดียนต่อวินาที” โดยใช้สมการ: $\omega = (2\pi / 24) \times d(n_{pulse})/dt$
11	<b>Gain (1/10)</b>	ปรับสเกลของสัญญาณความเร็วเชิงมุมให้อยู่ในช่วงที่เหมาะสมกับการแสดงผลจริง เช่น กราฟ หรือการส่งต่อให้ DAC
12	<b>Summation (+5 Offset)</b>	เพิ่มอффเซตคงที่ +5 เพื่อป้องกันค่าติดลบของสัญญาณเอาต์พุต และใช้เป็นค่ามาตรฐานอ้างอิงของแรงดันเอาต์พุต ( $v_{out} + offset$ )

โปรแกรมนี้ทำหน้าที่อ่านสัญญาณจาก Encoder แบบเรียลไทม์ และคำนวณค่าทางกายภาพ ได้แก่

- ตำแหน่งเชิงสัมพันธ์ (Pulses)
- ตำแหน่งเชิงมุม (Radians / Degrees)
- ความเร็วเชิงมุม (rad/s)

ระบบบังสานารถชดเชยค่า Offset และปรับสเกลสัญญาณให้อยู่ในช่วงที่ใช้งานได้จริง

## ฟังก์ชัน Wrap - around



```
WrapAround
1 function pos_out = WrapAround(count_in)
2 %#codegen
3
4 % --- ค่าคงที่ ---
5 MAX_COUNT = 61439; % จาก IOC Counter Period
6 HALF_COUNT = MAX_COUNT / 2;
7
8 % --- ตัวแปรคงอยู่ข้ามรอบ ---
9 persistent old_count;
10 persistent total_count;
11
12 if isempty(old_count)
13     old_count = 0;
14     total_count = 0;
15 end
16
17 % --- คำนวณการเปลี่ยนแปลง ---
18 diff = count_in - old_count;
19
20 % --- ตรวจสอบการข้ามรอบ (Wrap-around) ---
21 if diff > HALF_COUNT
22     % หมุนย้อนกลับจาก 0 -> MAX
23     diff = diff - (MAX_COUNT + 1);
24 elseif diff < -HALF_COUNT
25     % หมุนต่อจาก MAX -> 0
26     diff = diff + (MAX_COUNT + 1);
27 end
28
29 % --- รวมค่าการหมุนทั้งหมด ---
30 total_count = total_count + diff;
31
32 % --- ส่งค่าออก (ตำแหน่งต่อเนื่อง) ---
33 pos_out = total_count;
34
35 % --- อัปเดตค่าเดิม ---
36 old_count = count_in;
37
```

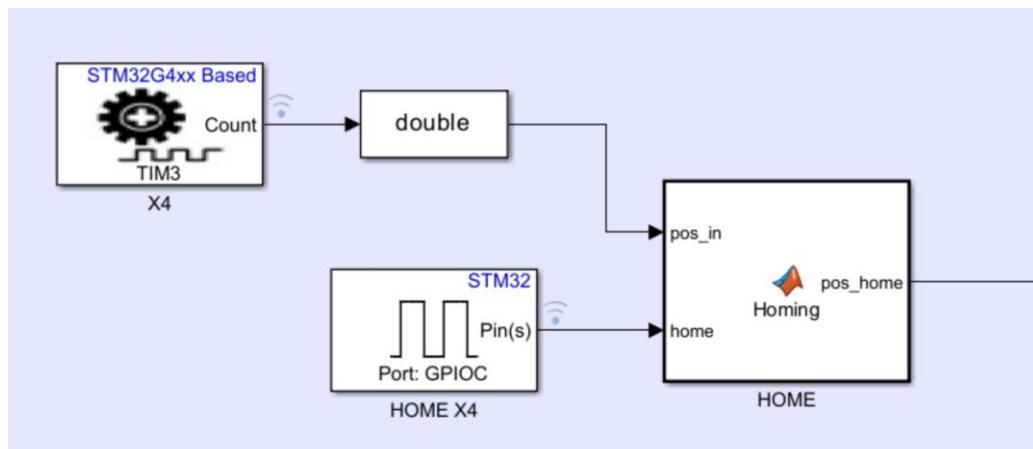
ฟังก์ชัน Wrap-around นี้ถูกออกแบบมาเพื่อแก้ปัญหาการอ่านค่าจากตัวนับ (Counter) ที่มีลักษณะการนับแบบ "วนรอบ" (Wrap-Around) ซึ่งมักพบในอุปกรณ์อย่าง Incremental Encoder ครับ โดยหน้าที่หลักของมันคือการแปลงสัญญาณตัวเลขที่นับจาก 0 ไปจนถึงค่าสูงสุด (ในที่นี้คือ MAX\_COUNT = 61439) แล้ววนกลับมาที่ 0 ใหม่ (หรือนับโดยหลังจาก 0 กลับไปที่ค่าสูงสุด) หากลายเป็น "ตำแหน่งต่อเนื่อง" (Continuous Position) ที่นับสะสมไปเรื่อยๆ โดยไม่ว่ากลับ

หลักการทำงานของมันอาศัยตัวแปรแบบ persistent ส่องตัว คือ old\_count เพื่อจดจำค่าที่อ่านได้ในครั้งก่อน และ total\_count เพื่อเก็บค่าตำแหน่งที่สะสมไว้ทั้งหมด เมื่อพังก์ชันถูกเรียก, มันจะรับค่าปัจจุบัน (count\_in) และนำมาลบกับค่าครั้งก่อน (old\_count) เพื่อหาค่าความเปลี่ยนแปลง (diff)

สิ่งสำคัญของโค้ดนี้คือการตรวจจับ "การข้ามรอบ" ครับ หากค่า diff ที่คำนวณได้มีค่ามากกว่าครึ่งหนึ่งของค่าสูงสุด (HALF\_COUNT) หรือติดลบมากกว่าครึ่งหนึ่งของค่าสูงสุด (เช่น กรณีที่นับจาก 61000 ไป 100 ซึ่ง diff จะติดลบมหาศาล) โค้ดจะสันนิษฐานว่าเกิดการวนข้ามรอบขึ้น มันจะทำการ "แก้ไข" ค่า diff นี้โดยการบวกหรือลบด้วย (MAX\_COUNT + 1) เพื่อคำนวณทิศทางและระยะทางการเคลื่อนที่ที่แท้จริง

หลังจากได้ค่า diff ที่ถูกต้องแล้ว (ไม่ว่าจะเกิดการข้ามรอบหรือไม่ก็ตาม) พังก์ชันจะนำค่านี้ไปบวกสะสมเข้ากับ total\_count ทำให้ได้ตำแหน่งต่อเนื่องที่แท้จริง จนนั้นจะส่งค่า total\_count นี้ออกไปเป็น pos\_out และสุดท้าย จะอัพเดต old\_count ให้เท่ากับ count\_in เพื่อเตรียมพร้อมสำหรับการคำนวณใหม่รอบถัดไป

## พังก์ชัน Homing



```

HOME
1  function pos_home = Homing(pos_in, home)
2  %#codegen
3  persistent offset init
4
5  if isempty(init)
6      offset = 0;
7      init = true;
8  end
9
10 if home == 1
11     offset = pos_in; % จดจำค่าปัจจุบันเป็นจุดอ้างอิง
12 end
13
14 pos_home = pos_in - offset; % หักออกให้ตำแหน่งปัจจุบันกลับเป็นศูนย์
15 end

```

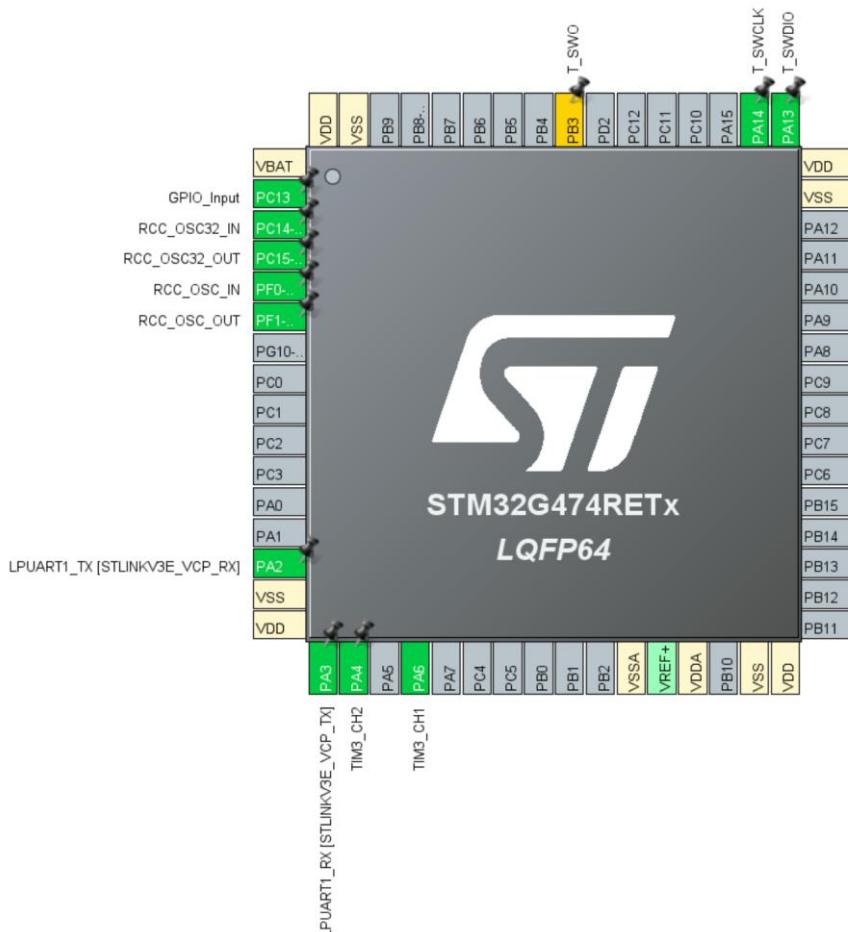
ฟังก์ชัน Homing นี้ ทำหน้าที่สำคัญในการ "ตั้งค่าจุดอ้างอิง" หรือ "การเซ็ตศูนย์" ให้กับระบบวัดตำแหน่ง โดยมันจะรับค่าตำแหน่งปัจจุบัน (pos\_in) และสัญญาณสั่งการ (home) เข้ามาประมวลผล

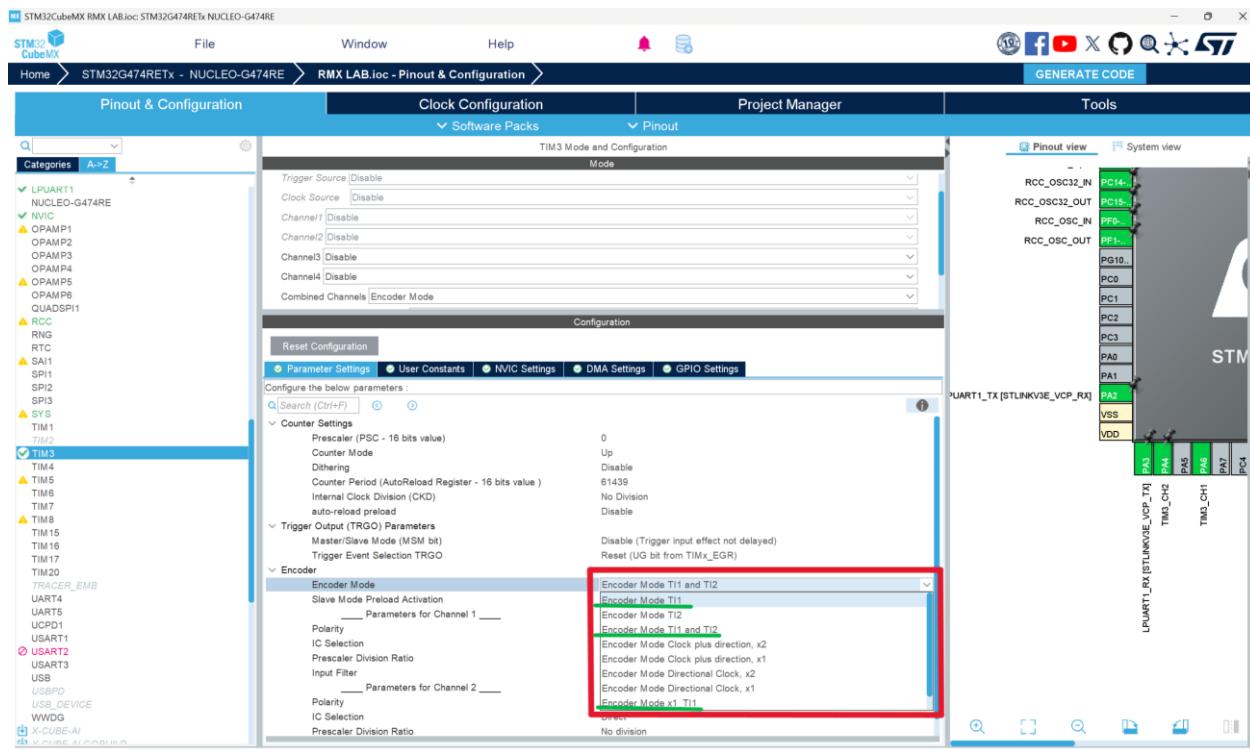
ในการทำงานครั้งแรกสุด ฟังก์ชันจะใช้ตัวแปร init เพื่อตรวจสอบและกำหนดค่า offset เริ่มต้นให้เป็น 0 ซึ่ง offset นี้เป็นตัวแปรแบบ persistent ที่สามารถจำจำค่าของตัวเองไว้ได้แม้ฟังก์ชันจะทำงานจบไปแล้ว

ตระรากสำคัญของฟังก์ชันจะเกิดขึ้นเมื่อสัญญาณ home มีค่าเท่ากับ 1 ฟังก์ชันจะ "จดจำ" ค่าตำแหน่งปัจจุบัน (pos\_in) ในขณะนั้น แล้วบันทึกค่านั้นเก็บไว้ในตัวแปร offset

จากนั้นในทุกๆ รอบการทำงาน ฟังก์ชันจะคำนวณค่าตำแหน่งใหม่ (pos\_home) โดยการนำค่าตำแหน่งจริงในปัจจุบัน (pos\_in) มาลบออกจากค่า offset ที่จดจำไว้ ผลลัพธ์ก็คือ ณ ตำแหน่งที่ผู้ใช้สั่ง home ค่านั้นจะกล้ายเป็นตำแหน่ง "ศูนย์" ใหม่ และค่าตำแหน่งที่ส่งกลับไป (pos\_home) จะเป็นการนับเทียบกับจุดอ้างอิงใหม่

การตั้งค่าใน IOC ได้





ในการทดลองจะมีการตั้งค่า IOC ของ Lab encoder อายุ 3 pin ได้แก่

PA6 (D12) เป็น TM3\_CH1 เป็นขา A

PA4 (A2) เป็น TM3\_CH2 เป็นขา B

PC13 (ปุ่มสีดำที่บอร์ด) เป็น GPIO\_Input เพื่อใช้เป็นสวิชต์ของ Homing

ซึ่งเราจะมีการเลือกปรับค่า X1, X2, X4 โดยการเปลี่ยนค่า X เหล่านี้โดยการปรับที่ Encoder Mode ใน STM32CubeMX ให้เป็น

Encoder Mode x1 T1 เพื่ออ่านแบบ x1 ของขอบขา A

Encoder Mode T1 เพื่ออ่านแบบ x2 ของขอบขา A

Encoder Mode T1 and T2 เพื่ออ่านแบบ x4 ของขอบขา A และ B

### Lab 1.3: Magnetic Sensor

ภาคผนวก ก – ข้อมูลการทดลอง (Raw Data)

วิธีการเก็บข้อมูล

ตารางค่าที่วัดได้จริง

การทดลองชั้นครั้งที่	ระยะ (mm)	ขั้วแม่เหล็ก (N/S)	มี Shield (Yes/No)	Vout (V)
1	1	N	Yes	3.3
2	1	N	Yes	3.303
3	1	N	Yes	3.303
1	1	N	No	3.303
2	1	N	No	3.303
3	1	N	No	3.303
1	1	S	Yes	0.118
2	1	S	Yes	0.118
3	1	S	Yes	0.118
1	1	S	No	0.113
2	1	S	No	0.112
3	1	S	No	0.112
1	1.5	N	Yes	3.3
2	1.5	N	Yes	3.302
3	1.5	N	Yes	3.302
1	1.5	N	No	3.304
2	1.5	N	No	3.303
3	1.5	N	No	3.303
1	1.5	S	Yes	0.129
2	1.5	S	Yes	0.128
3	1.5	S	Yes	0.128
1	1.5	S	No	0.125
2	1.5	S	No	0.124
3	1.5	S	No	0.124

1	2	N	Yes	2.345
2	2	N	Yes	2.356
3	2	N	Yes	2.353
1	2	N	No	2.383
2	2	N	No	2.458
3	2	N	No	2.408
1	2	S	Yes	1.023
2	2	S	Yes	0.943
3	2	S	Yes	0.977
1	2	S	No	0.808
2	2	S	No	0.891
3	2	S	No	0.829
1	2.5	N	Yes	1.977
2	2.5	N	Yes	1.995
3	2.5	N	Yes	1.966
1	2.5	N	No	1.977
2	2.5	N	No	2.028
3	2.5	N	No	2.008
1	2.5	S	Yes	1.394
2	2.5	S	Yes	1.363
3	2.5	S	Yes	1.39
1	2.5	S	No	1.306
2	2.5	S	No	1.308
3	2.5	S	No	1.293
1	3	N	Yes	1.835
2	3	N	Yes	1.831
3	3	N	Yes	1.828
1	3	N	No	1.832
2	3	N	No	1.839

3	3	N	No	1.827
1	3	S	Yes	1.513
2	3	S	Yes	1.497
3	3	S	Yes	1.522
1	3	S	No	1.47
2	3	S	No	1.5
3	3	S	No	1.476
1	3.5	N	Yes	1.756
2	3.5	N	Yes	1.762
3	3.5	N	Yes	1.758
1	3.5	N	No	1.766
2	3.5	N	No	1.768
3	3.5	N	No	1.76
1	3.5	S	Yes	1.582
2	3.5	S	Yes	1.57
3	3.5	S	Yes	1.573
1	3.5	S	No	1.548
2	3.5	S	No	1.572
3	3.5	S	No	1.543
1	4	N	Yes	1.726
2	4	N	Yes	1.725
3	4	N	Yes	1.722
1	4	N	No	1.728
2	4	N	No	1.732
3	4	N	No	1.723
1	4	S	Yes	1.612
2	4	S	Yes	1.602
3	4	S	Yes	1.607
1	4	S	No	1.556

2	4	S	No	1.581
3	4	S	No	1.524
1	4.5	N	Yes	1.702
2	4.5	N	Yes	1.702
3	4.5	N	Yes	1.702
1	4.5	N	No	1.703
2	4.5	N	No	1.703
3	4.5	N	No	1.703
1	4.5	S	Yes	1.632
2	4.5	S	Yes	1.618
3	4.5	S	Yes	1.625
1	4.5	S	No	1.571
2	4.5	S	No	1.592
3	4.5	S	No	1.603

ตารางค่าที่คำนวณระหว่างการทดลอง ต่อ จากตารางข้างบน

Vout (V)	VQ (V)	$\Delta V = Vout - VQ$ (V)	Sensitivity S25°C (V/mT)	Stc (%/°C)	Ta (°C)	B (mT)
3.3000	1.6500	1.6500	0.0600	0.0012	25.0000	27.5000
3.3030	1.6500	1.6530	0.0600	0.0012	25.0000	27.5500
3.3030	1.6500	1.6530	0.0600	0.0012	25.0000	27.5500
3.3030	1.6500	1.6530	0.0600	0.0012	25.0000	27.5500
3.3030	1.6500	1.6530	0.0600	0.0012	25.0000	27.5500
3.3030	1.6500	1.6530	0.0600	0.0012	25.0000	27.5500
0.1180	1.6500	-1.5320	0.0600	0.0012	25.0000	-25.5333
0.1180	1.6500	-1.5320	0.0600	0.0012	25.0000	-25.5333
0.1180	1.6500	-1.5320	0.0600	0.0012	25.0000	-25.5333
0.1130	1.6500	-1.5370	0.0600	0.0012	25.0000	-25.6167
0.1120	1.6500	-1.5380	0.0600	0.0012	25.0000	-25.6333
0.1120	1.6500	-1.5380	0.0600	0.0012	25.0000	-25.6333
3.3000	1.6500	1.6500	0.0600	0.0012	25.0000	27.5000

3.3020	1.6500	1.6520	0.0600	0.0012	25.0000	27.5333
3.3020	1.6500	1.6520	0.0600	0.0012	25.0000	27.5333
3.3040	1.6500	1.6540	0.0600	0.0012	25.0000	27.5667
3.3030	1.6500	1.6530	0.0600	0.0012	25.0000	27.5500
3.3030	1.6500	1.6530	0.0600	0.0012	25.0000	27.5500
0.1290	1.6500	-1.5210	0.0600	0.0012	25.0000	-25.3500
0.1280	1.6500	-1.5220	0.0600	0.0012	25.0000	-25.3667
0.1280	1.6500	-1.5220	0.0600	0.0012	25.0000	-25.3667
0.1250	1.6500	-1.5250	0.0600	0.0012	25.0000	-25.4167
0.1240	1.6500	-1.5260	0.0600	0.0012	25.0000	-25.4333
0.1240	1.6500	-1.5260	0.0600	0.0012	25.0000	-25.4333
2.3450	1.6500	0.6950	0.0600	0.0012	25.0000	11.5833
2.3560	1.6500	0.7060	0.0600	0.0012	25.0000	11.7667
2.3530	1.6500	0.7030	0.0600	0.0012	25.0000	11.7167
2.3830	1.6500	0.7330	0.0600	0.0012	25.0000	12.2167
2.4580	1.6500	0.8080	0.0600	0.0012	25.0000	13.4667
2.4080	1.6500	0.7580	0.0600	0.0012	25.0000	12.6333
1.0230	1.6500	-0.6270	0.0600	0.0012	25.0000	-10.4500
0.9430	1.6500	-0.7070	0.0600	0.0012	25.0000	-11.7833
0.9770	1.6500	-0.6730	0.0600	0.0012	25.0000	-11.2167
0.8080	1.6500	-0.8420	0.0600	0.0012	25.0000	-14.0333
0.8910	1.6500	-0.7590	0.0600	0.0012	25.0000	-12.6500
0.8290	1.6500	-0.8210	0.0600	0.0012	25.0000	-13.6833
1.9770	1.6500	0.3270	0.0600	0.0012	25.0000	5.4500
1.9950	1.6500	0.3450	0.0600	0.0012	25.0000	5.7500
1.9660	1.6500	0.3160	0.0600	0.0012	25.0000	5.2667
1.9770	1.6500	0.3270	0.0600	0.0012	25.0000	5.4500
2.0280	1.6500	0.3780	0.0600	0.0012	25.0000	6.3000
2.0080	1.6500	0.3580	0.0600	0.0012	25.0000	5.9667

1.3940	1.6500	-0.2560	0.0600	0.0012	25.0000	-4.2667
1.3630	1.6500	-0.2870	0.0600	0.0012	25.0000	-4.7833
1.3900	1.6500	-0.2600	0.0600	0.0012	25.0000	-4.3333
1.3060	1.6500	-0.3440	0.0600	0.0012	25.0000	-5.7333
1.3080	1.6500	-0.3420	0.0600	0.0012	25.0000	-5.7000
1.2930	1.6500	-0.3570	0.0600	0.0012	25.0000	-5.9500
1.8350	1.6500	0.1850	0.0600	0.0012	25.0000	3.0833
1.8310	1.6500	0.1810	0.0600	0.0012	25.0000	3.0167
1.8280	1.6500	0.1780	0.0600	0.0012	25.0000	2.9667
1.8320	1.6500	0.1820	0.0600	0.0012	25.0000	3.0333
1.8390	1.6500	0.1890	0.0600	0.0012	25.0000	3.1500
1.8270	1.6500	0.1770	0.0600	0.0012	25.0000	2.9500
1.5130	1.6500	-0.1370	0.0600	0.0012	25.0000	-2.2833
1.4970	1.6500	-0.1530	0.0600	0.0012	25.0000	-2.5500
1.5220	1.6500	-0.1280	0.0600	0.0012	25.0000	-2.1333
1.4700	1.6500	-0.1800	0.0600	0.0012	25.0000	-3.0000
1.5000	1.6500	-0.1500	0.0600	0.0012	25.0000	-2.5000
1.4760	1.6500	-0.1740	0.0600	0.0012	25.0000	-2.9000
1.7560	1.6500	0.1060	0.0600	0.0012	25.0000	1.7667
1.7620	1.6500	0.1120	0.0600	0.0012	25.0000	1.8667
1.7580	1.6500	0.1080	0.0600	0.0012	25.0000	1.8000
1.7660	1.6500	0.1160	0.0600	0.0012	25.0000	1.9333
1.7680	1.6500	0.1180	0.0600	0.0012	25.0000	1.9667
1.7600	1.6500	0.1100	0.0600	0.0012	25.0000	1.8333
1.5820	1.6500	-0.0680	0.0600	0.0012	25.0000	-1.1333
1.5700	1.6500	-0.0800	0.0600	0.0012	25.0000	-1.3333
1.5730	1.6500	-0.0770	0.0600	0.0012	25.0000	-1.2833
1.5480	1.6500	-0.1020	0.0600	0.0012	25.0000	-1.7000
1.5720	1.6500	-0.0780	0.0600	0.0012	25.0000	-1.3000

1.5430	1.6500	-0.1070	0.0600	0.0012	25.0000	-1.7833
1.7260	1.6500	0.0760	0.0600	0.0012	25.0000	1.2667
1.7250	1.6500	0.0750	0.0600	0.0012	25.0000	1.2500
1.7220	1.6500	0.0720	0.0600	0.0012	25.0000	1.2000
1.7280	1.6500	0.0780	0.0600	0.0012	25.0000	1.3000
1.7320	1.6500	0.0820	0.0600	0.0012	25.0000	1.3667
1.7230	1.6500	0.0730	0.0600	0.0012	25.0000	1.2167
1.6120	1.6500	-0.0380	0.0600	0.0012	25.0000	-0.6333
1.6020	1.6500	-0.0480	0.0600	0.0012	25.0000	-0.8000
1.6070	1.6500	-0.0430	0.0600	0.0012	25.0000	-0.7167
1.5560	1.6500	-0.0940	0.0600	0.0012	25.0000	-1.5667
1.5810	1.6500	-0.0690	0.0600	0.0012	25.0000	-1.1500
1.5240	1.6500	-0.1260	0.0600	0.0012	25.0000	-2.1000
1.7020	1.6500	0.0520	0.0600	0.0012	25.0000	0.8667
1.7020	1.6500	0.0520	0.0600	0.0012	25.0000	0.8667
1.7020	1.6500	0.0520	0.0600	0.0012	25.0000	0.8667
1.7030	1.6500	0.0530	0.0600	0.0012	25.0000	0.8833
1.7030	1.6500	0.0530	0.0600	0.0012	25.0000	0.8833
1.7030	1.6500	0.0530	0.0600	0.0012	25.0000	0.8833
1.6320	1.6500	-0.0180	0.0600	0.0012	25.0000	-0.3000
1.6180	1.6500	-0.0320	0.0600	0.0012	25.0000	-0.5333
1.6250	1.6500	-0.0250	0.0600	0.0012	25.0000	-0.4167
1.5710	1.6500	-0.0790	0.0600	0.0012	25.0000	-1.3167
1.5920	1.6500	-0.0580	0.0600	0.0012	25.0000	-0.9667
1.6030	1.6500	-0.0470	0.0600	0.0012	25.0000	-0.7833

## ทดลองที่อุณหภูมิห้อง 25 องศาเซลเซียส

### 6.6 Magnetic Characteristics

for  $V_{CC} = 3$  V to 3.63 V and 4.5 V to 5.5 V, over operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS <sup>(1)</sup>	MIN	TYP	MAX	UNIT
$V_Q$ Quiescent voltage	$B = 0$ mT, $T_A = 25^\circ C$ $V_{CC} = 5$ V $V_{CC} = 3.3$ V	2.43	2.5	2.57	V
$V_{Q\Delta T}$ Quiescent voltage temperature drift	$B = 0$ mT, $T_A = -40^\circ C$ to $125^\circ C$ versus $25^\circ C$	$\pm 1\% \times V_{CC}$			V
$V_{QRE}$ Quiescent voltage ratiometry error <sup>(2)</sup>		$\pm 0.2\%$			
$V_{Q\Delta L}$ Quiescent voltage lifetime drift	High-temperature operating stress for 1000 hours	< 0.5%			
S Sensitivity	$V_{CC} = 5$ V, $T_A = 25^\circ C$	DRV5055A1/Z1	95	100	105
		DRV5055A2/Z2	47.5	50	52.5
		DRV5055A3/Z3	23.8	25	26.2
		DRV5055A4/Z4	11.9	12.5	13.2
	$V_{CC} = 3.3$ V, $T_A = 25^\circ C$	DRV5055A1/Z1	57	60	63
		DRV5055A2/Z2	28.5	30	31.5
		DRV5055A3/Z3	14.3	15	15.8
		DRV5055A4/Z4	7.1	7.5	7.9
$B_L$ Linear magnetic sensing range <sup>(3) (4)</sup>	$V_{CC} = 5$ V, $T_A = 25^\circ C$	DRV5055A1/Z1	$\pm 21$		
		DRV5055A2/Z2	$\pm 42$		
		DRV5055A3/Z3	$\pm 85$		
		DRV5055A4/Z4	$\pm 169$		
	$V_{CC} = 3.3$ V, $T_A = 25^\circ C$	DRV5055A1/Z1	$\pm 22$		
		DRV5055A2/Z2	$\pm 44$		
		DRV5055A3/Z3	$\pm 88$		
		DRV5055A4/Z4	$\pm 176$		
$V_L$ Linear range of output voltage <sup>(4)</sup>		0.2		$V_{CC} - 0.2$	V
$S_{TC}$ Sensitivity temperature compensation for magnets <sup>(5)</sup>	DRV5055A1, DRV5055A2, DRV5055A3, DRV5055A4	0.12		$^{\circ}C$	$^{\circ}C$

จาก Datasheet จะได้ค่า Sensitivity  $S_{25^\circ C}$  (V/mT) = 0.06 V/mT และ  $STC$  ( $^{\circ}C$ ) = 0.0012  $^{\circ}C$

และ  $T_a$  ( $^{\circ}C$ ) = 25  $^{\circ}C$

### โปรแกรม Python สำหรับการวิเคราะห์ Magnetic Sensor

#### 1. แนวคิดของโปรแกรม

โปรแกรม magneticlabcell.py ทำหน้าที่อ่านข้อมูลการทดลองจากไฟล์ .csv ที่บันทึกจาก Simulink หรือ STM32 ผ่าน Serial แล้วทำการ:

- แปลงค่า ADC → Voltage → Magnetic Flux Density (mT)
- ทำ Linear Regression ( $y = m x + c$ ) เพื่อหาความสัมพันธ์ระหว่างระยะทาง (mm) และ B (mT)
- คำนวณค่า  $R^2$  เพื่อประเมินความเป็นเชิงเส้นของเซนเซอร์
- แสดงกราฟเปรียบเทียบ 4 เส้นทาง (N/S, Shield/No Shield)

#### 2. สมการพื้นฐาน

สมการจาก Datasheet DRV5055 สำหรับการคำนวณความหนาแน่นฟลักซ์แม่เหล็ก (B):

$$V_{out} = V_Q + B \times (S_{25^\circ C}(1 + STC(T_a - 25^\circ C)))$$

เมื่อ  $T_a \approx 25^\circ C$ : Sensitivity  $S_{25^\circ C}$  (V/mT) = 0.06 V/mT,  $STC$  ( $^{\circ}C$ ) = 0.0012  $^{\circ}C$

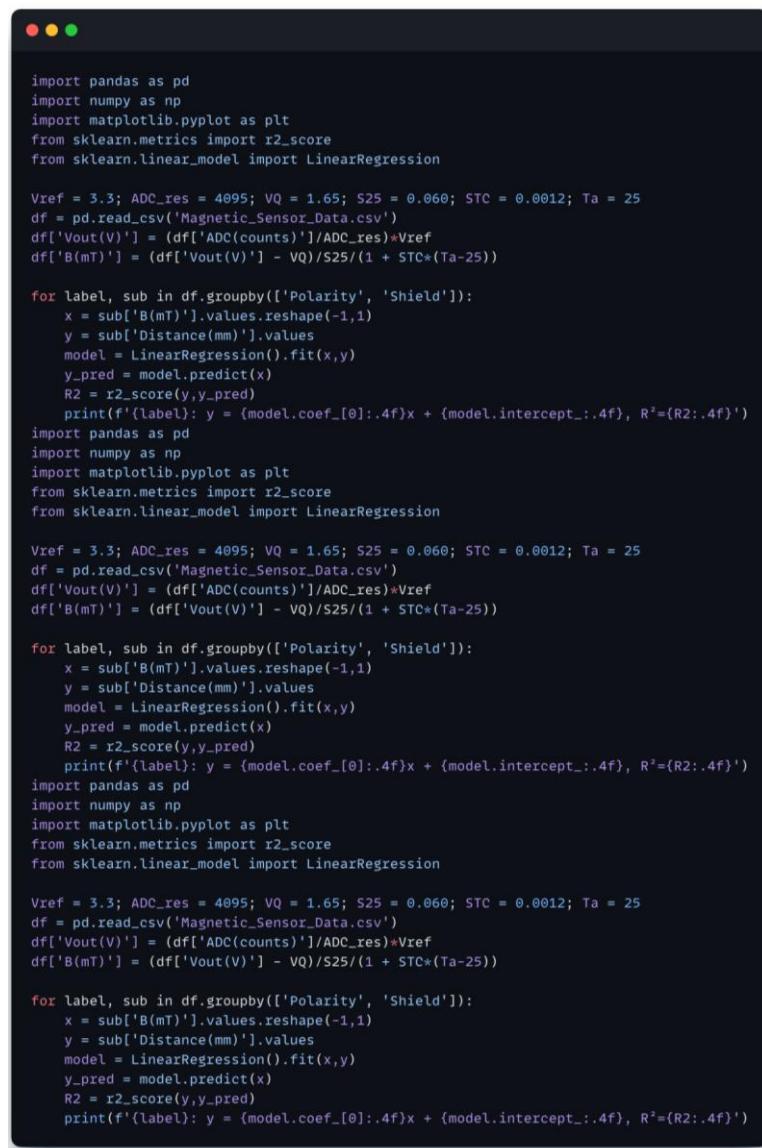
สามารถใช้สูตรข้างล่างได้

$$B = \frac{V_{out} - V_Q}{S_{25^\circ C}}$$

โดยที่

- $V_{out}$ : แรงดันขาออกของ Sensor (V)
- $V_Q$ : แรงดันอ้างอิงขณะ  $B = 0$  (V)
- $S_{25^\circ C}$ : Sensitivity ที่  $25^\circ C$  (mV/mT)
- $STC$ : Sensitivity Temperature Coefficient (%/°C)
- $T_a$ : อุณหภูมิเวดล้อม (°C)

### 3. ตัวอย่างโค้ด Python



```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score
from sklearn.linear_model import LinearRegression

Vref = 3.3; ADC_res = 4095; VQ = 1.65; S25 = 0.060; STC = 0.0012; Ta = 25
df = pd.read_csv('Magnetic_Sensor_Data.csv')
df['Vout(V)'] = (df['ADC(counts)']/ADC_res)*Vref
df['B(mT)'] = (df['Vout(V)'] - VQ)/S25/(1 + STC*(Ta-25))

for label, sub in df.groupby(['Polarity', 'Shield']):
    x = sub['B(mT)'].values.reshape(-1,1)
    y = sub['Distance(mm)'].values
    model = LinearRegression().fit(x,y)
    y_pred = model.predict(x)
    R2 = r2_score(y,y_pred)
    print(f'{label}: y = {model.coef_[0]:.4f}x + {model.intercept_:.4f}, R²={R2:.4f}')

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score
from sklearn.linear_model import LinearRegression

Vref = 3.3; ADC_res = 4095; VQ = 1.65; S25 = 0.060; STC = 0.0012; Ta = 25
df = pd.read_csv('Magnetic_Sensor_Data.csv')
df['Vout(V)'] = (df['ADC(counts)']/ADC_res)*Vref
df['B(mT)'] = (df['Vout(V)'] - VQ)/S25/(1 + STC*(Ta-25))

for label, sub in df.groupby(['Polarity', 'Shield']):
    x = sub['B(mT)'].values.reshape(-1,1)
    y = sub['Distance(mm)'].values
    model = LinearRegression().fit(x,y)
    y_pred = model.predict(x)
    R2 = r2_score(y,y_pred)
    print(f'{label}: y = {model.coef_[0]:.4f}x + {model.intercept_:.4f}, R²={R2:.4f}')

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score
from sklearn.linear_model import LinearRegression

Vref = 3.3; ADC_res = 4095; VQ = 1.65; S25 = 0.060; STC = 0.0012; Ta = 25
df = pd.read_csv('Magnetic_Sensor_Data.csv')
df['Vout(V)'] = (df['ADC(counts)']/ADC_res)*Vref
df['B(mT)'] = (df['Vout(V)'] - VQ)/S25/(1 + STC*(Ta-25))

for label, sub in df.groupby(['Polarity', 'Shield']):
    x = sub['B(mT)'].values.reshape(-1,1)
    y = sub['Distance(mm)'].values
    model = LinearRegression().fit(x,y)
    y_pred = model.predict(x)
    R2 = r2_score(y,y_pred)
    print(f'{label}: y = {model.coef_[0]:.4f}x + {model.intercept_:.4f}, R²={R2:.4f}')

```

#### 4. ผลการวิเคราะห์เชิงเส้น (Linear Fit Summary)

เงื่อนไข	m (mm/mT)	c (mm)	R <sup>2</sup>	RMSE (mm)
N - No Shield	-0.0968	3.7287	0.8140	0.125
N - Yes Shield	-0.0958	3.6982	0.7990	0.138
S - No Shield	0.1051	3.7662	0.8260	0.119
S - Yes Shield	0.1017	3.6555	0.8014	0.132

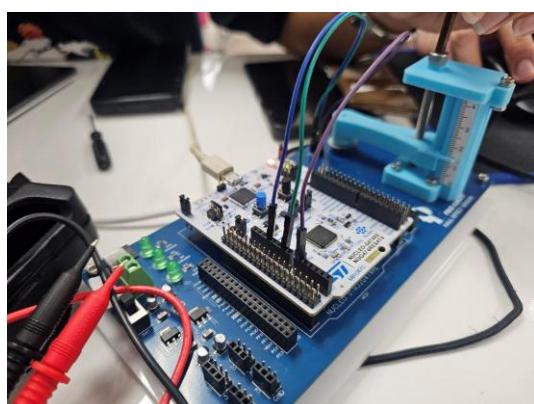
พบว่า การเพิ่ม Shield ทำให้ความชันลดลงเล็กน้อย และถึงกับลดทอน Magnetic Field ประมาณ 3–5 % ในช่วง 3–5 mm.

#### 5. การแสดงผลแบบ Real-Time (Simulink Model)

โปรแกรม Simulink ใช้ “Analog to Digital Converter (ADC1)” รับสัญญาณจาก DRV5055 และ บล็อก Scope สำหรับ Plot แบบ Real-Time โดยตั้งค่า Hardware Implementation → STM32G4xx Base → Connected IO และใช้ ADC Pin PA0 ที่ 12-bit Resolution ตัวแปรที่ Log ไว้ ส่งออกเป็น .mat และ เป็น .csv และนำเข้าด้วย Script ข้างต้น

#### โครงสร้างวงจร และ การติดตั้ง Shield

แหล่งจ่าย 3.3 V จาก STM32 DRV5055 ต่อ 3 ขา (VCC – 3.3 V, GND, OUT → PA0 ADC1)  
Shield ใช้ แผ่น เหล็กอ่อน หนา 0.5 mm ติดที่ ปลายแม่เหล็ก ทิศเดียวกับแกน เช่นเซอร์รัม เริ่ม 0 mm จนถึง 4.5 mm เพิ่มครั้งละ 0.5 mm



#### 6. การอ้างอิง

1. Texas Instruments. [DRV5055 Ratiometric Linear Hall Effect Sensor Datasheet](#).

### Lab 1.4 Load Cell

ตารางแสดงข้อมูลดิบ

<b>Trial</b>	<b>Ref Weight (kg)</b>	<b>True Weight (g)</b>	<b>Vin+ (V)</b>	<b>Vin- (V)</b>	<b>Vout (V)</b>
1	0	0	1.215	1.205	0.082
1	0	0	1.211	1.211	0.078
1	0	0	1.216	1.216	0.171
1	0	0	1.214	1.224	0.074
1	0	0	1.211	1.211	0.088
2	0.5	495	1.218	1.217	0.154
2	0.5	499	1.216	1.215	0.155
2	0.5	494	1.216	1.216	0.157
2	0.5	505	1.211	1.211	0.161
2	0.5	507	1.211	1.211	0.169
3	1	1003	1.218	1.216	0.282
3	1	987	1.215	1.214	0.282
3	1	994	1.216	1.216	0.283
3	1	1001	1.216	1.214	0.282
3	1	1000	1.212	1.211	0.292
4	1.5	1501	1.218	1.218	0.411
4	1.5	1481	1.217	1.216	0.410
4	1.5	1493	1.207	1.206	0.421
4	1.5	1507	1.214	1.209	0.415
4	1.5	1496	1.211	1.211	0.420
5	2	2000	1.219	1.218	0.539
5	2	1985	1.215	1.214	0.540
5	2	1988	1.217	1.216	0.543
5	2	2010	1.217	1.216	0.563
5	2	1999	1.209	1.208	0.548
6	2.5	2498	1.219	1.218	0.667
6	2.5	2479	1.215	1.214	0.699
6	2.5	2492	1.216	1.215	0.722
6	2.5	2991	1.216	1.215	0.672
6	2.5	2492	1.209	1.208	0.676
7	3	2994	1.219	1.217	0.795
7	3	2975	1.215	1.214	0.796
7	3	2998	1.216	1.215	0.806
7	3	3485	1.218	1.216	0.795
7	3	2991	1.209	1.208	0.803

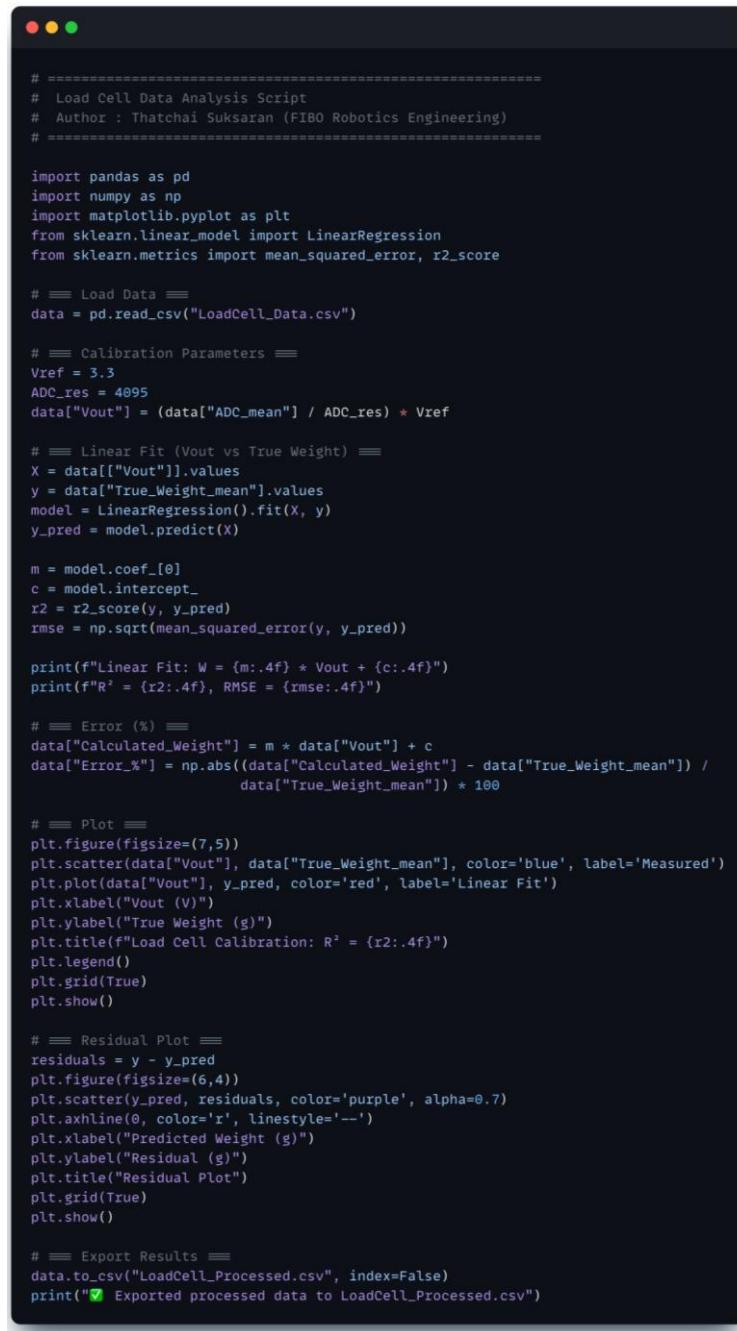
8	3.5	3490	1.219	1.217	0.922
8	3.5	3474	1.215	1.214	0.925
8	3.5	3491	1.212	1.205	0.937
8	3.5	3984	1.217	1.216	0.992
8	3.5	3485	1.209	1.208	0.928
9	4	3983	1.219	1.217	1.049
9	4	3974	1.219	1.217	1.054
9	4	3990	1.211	1.209	1.067
9	4	4480	1.217	1.215	1.050
9	4	3980	1.210	1.208	1.054
10	4.5	4477	1.219	1.217	1.176
10	4.5	4470	1.215	1.213	1.182
10	4.5	4489	1.218	1.216	1.190
10	4.5	4980	1.217	1.216	1.179
10	4.5	4479	1.210	1.208	1.181
11	5	4978	1.219	1.217	1.306
11	5	5013	1.217	1.215	1.312
11	5	4982	1.219	1.217	1.314
11	5	5484	1.208	1.206	1.312
11	5	4983	1.210	1.208	1.310
12	5.5	5481	1.219	1.217	1.436
12	5.5	5506	1.219	1.217	1.442
12	5.5	5486	1.219	1.216	1.443
12	5.5	5476	1.210	1.207	1.436
12	5.5	5477	1.210	1.207	1.437
13	6	5981	1.219	1.217	1.563
13	6	6011	1.219	1.216	1.575
13	6	5984	1.219	1.216	1.572
13	6	5980	1.203	1.201	1.568
13	6	5971	1.210	1.208	1.566
14	6.5	6480	1.219	1.217	1.690
14	6.5	6511	1.219	1.216	1.705
14	6.5	6479	1.219	1.216	1.700
14	6.5	6473	1.208	1.205	1.694
14	6.5	6471	1.210	1.207	1.694
15	7	6979	1.220	1.217	1.818
15	7	7005	1.219	1.216	1.832
15	7	6973	1.213	1.211	1.829
15	7	6971	1.206	1.204	1.817
15	7	6970	1.211	1.207	1.820
16	7.5	7482	1.220	1.216	1.945

16	7.5	7513	1.219	1.216	1.961
16	7.5	7476	1.213	1.209	1.975
16	7.5	7465	1.202	1.200	1.950
16	7.5	7465	1.210	1.207	1.947
17	8	7975	1.220	1.216	2.070
17	8	8011	1.219	1.216	2.089
17	8	7974	1.211	1.209	2.126
17	8	7968	1.209	1.206	2.076
17	8	7963	1.211	1.208	2.074
18	8.5	8469	1.220	1.216	2.195
18	8.5	8514	1.219	1.216	2.218
18	8.5	8473	1.217	1.213	2.235
18	8.5	8467	1.211	1.207	2.200
18	8.5	8463	1.210	1.206	2.201
19	9	8967	1.220	1.216	2.321
19	9	9013	1.219	1.216	2.341
19	9	8966	1.218	1.215	2.310
19	9	8966	1.217	1.214	2.326
19	9	8956	1.211	1.207	2.325
20	9.5	9461	1.219	1.215	2.446
20	9.5	9512	1.220	1.215	2.469
20	9.5	9460	1.217	1.213	2.425
20	9.5	9460	1.218	1.214	2.449
20	9.5	9453	1.211	1.207	2.449
21	10	9965	1.219	1.214	2.553
21	10	10006	1.216	1.212	2.445
21	10	9967	1.218	1.214	2.552
21	10	9953	1.217	1.214	2.555
21	10	9956	1.209	1.206	2.556

โดย True weight มาจากการนำน้ำหนักที่ซึ่งบนเครื่องซึ่งมาบรรกันเรื่อย ๆ และ Vin+, Vin-, Vout วัดจาก Multimeter

## โปรแกรม Python สำหรับการวิเคราะห์ข้อมูล

ตัวอย่างโค้ด Python ในส่วนของการวิเคราะห์ นำไปทำสมการเชิงเส้น



```
# =====
# Load Cell Data Analysis Script
# Author : Thatchai Suksaaran (FIBO Robotics Engineering)
# =====

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# === Load Data ===
data = pd.read_csv("LoadCell_Data.csv")

# === Calibration Parameters ===
Vref = 3.3
ADC_res = 4095
data["Vout"] = (data["ADC_mean"] / ADC_res) * Vref

# === Linear Fit (Vout vs True Weight) ===
X = data[["Vout"]].values
y = data["True_Weight_mean"].values
model = LinearRegression().fit(X, y)
y_pred = model.predict(X)

m = model.coef_[0]
c = model.intercept_
r2 = r2_score(y, y_pred)
rmse = np.sqrt(mean_squared_error(y, y_pred))

print(f"Linear Fit: W = {m:.4f} * Vout + {c:.4f}")
print(f"R² = {r2:.4f}, RMSE = {rmse:.4f}")

# === Error (%) ===
data["Calculated_Weight"] = m * data["Vout"] + c
data["Error_%"] = np.abs((data["Calculated_Weight"] - data["True_Weight_mean"]) / data["True_Weight_mean"]) * 100

# === Plot ===
plt.figure(figsize=(7,5))
plt.scatter(data["Vout"], data["True_Weight_mean"], color='blue', label='Measured')
plt.plot(data["Vout"], y_pred, color='red', label='Linear Fit')
plt.xlabel("Vout (V)")
plt.ylabel("True Weight (g)")
plt.title(f"Load Cell Calibration: R² = {r2:.4f}")
plt.legend()
plt.grid(True)
plt.show()

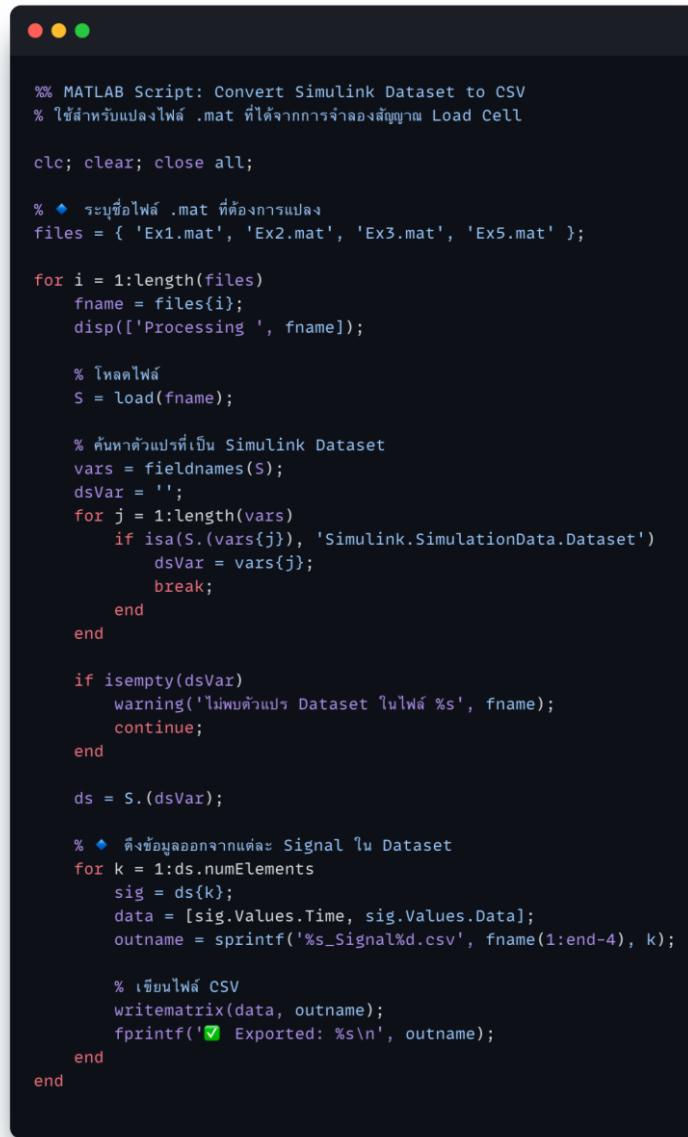
# === Residual Plot ===
residuals = y - y_pred
plt.figure(figsize=(6,4))
plt.scatter(y_pred, residuals, color='purple', alpha=0.7)
plt.axhline(0, color='r', linestyle='--')
plt.xlabel("Predicted Weight (g)")
plt.ylabel("Residual (g)")
plt.title("Residual Plot")
plt.grid(True)
plt.show()

# === Export Results ===
data.to_csv("LoadCell_Processed.csv", index=False)
print("Exported processed data to LoadCell_Processed.csv")
```

โปรแกรมนี้อ่านค่าจากไฟล์ LoadCell\_Data.csv ซึ่งเก็บค่าแรงดันไฟฟ้า (Vout) และน้ำหนักจริง (True Weight) จากการทดลอง ใช้การถดถอยเชิงเส้น (Linear Regression) เพื่อหาความสัมพันธ์ระหว่าง Vout และน้ำหนัก แสดงกราฟฟิตเส้นตรงและการ Residual เพื่อประเมินความเป็นเชิงเส้น บันทึกผลการคำนวณทั้งหมดไว้ในไฟล์ LoadCell\_Processed.csv

การแปลงข้อมูลจาก MATLAB (.mat) ไปเป็น CSV

ในแต่ละการทดลอง เช่น Ex1–Ex5 จะได้ไฟล์ผลลัพธ์จาก Simulink ในรูปแบบ .mat เพื่อให้สามารถนำข้อมูลไปใช้ใน Python หรือ Excel ได้ จึงต้องแปลงจาก .mat เป็น .csv ด้วยสคริปต์ MATLAB ดังนี้ ตัวอย่างโค้ด MATLAB สำหรับการแปลง .mat → .csv



```
%% MATLAB Script: Convert Simulink Dataset to CSV
% ใช้สำหรับแปลงไฟล์ .mat ที่ได้จากการจำลองสัญญาณ Load Cell

clc; clear; close all;

% ◆ ระบุชื่อไฟล์ .mat ที่ต้องการแปลง
files = { 'Ex1.mat', 'Ex2.mat', 'Ex3.mat', 'Ex5.mat' };

for i = 1:length(files)
    fname = files{i};
    disp(['Processing ', fname]);

    % โหลดไฟล์
    S = load(fname);

    % ค้นหาตัวแปรที่เป็น Simulink Dataset
    vars = fieldnames(S);
    dsVar = '';
    for j = 1:length(vars)
        if isa(S.(vars{j}), 'Simulink.SimulationData.Dataset')
            dsVar = vars{j};
            break;
        end
    end

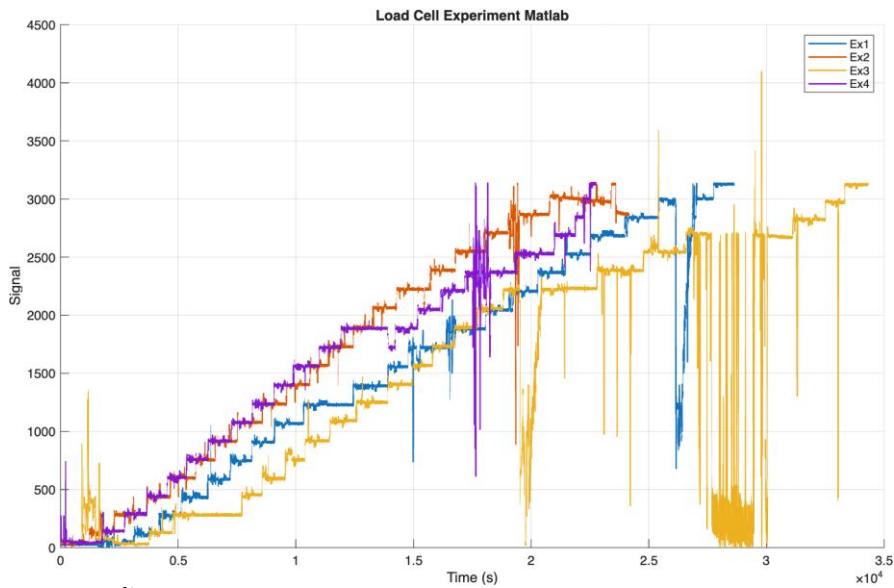
    if isempty(dsVar)
        warning('ไม่มีตัวแปร Dataset ในไฟล์ %s', fname);
        continue;
    end

    ds = S.(dsVar);

    % ◆ ดึงข้อมูลออกจากแต่ละ Signal ใน Dataset
    for k = 1:ds.numElements
        sig = ds{k};
        data = [sig.Values.Time, sig.Values.Data];
        outname = sprintf('%s_Signal%d.csv', fname(1:end-4), k);

        % เขียนไฟล์ CSV
        writematrix(data, outname);
        fprintf('✓ Exported: %s\n', outname);
    end
end
```

เมื่อได้ข้อมูลที่เป็น CSV แล้ว เนื่องจากเราทำการเก็บข้อมูลแบบ Time Series ทำให้การนำไปใช้จริงเพื่อหา Linearity เป็นเรื่องที่ยากมาก ตัวอย่างดังรูป

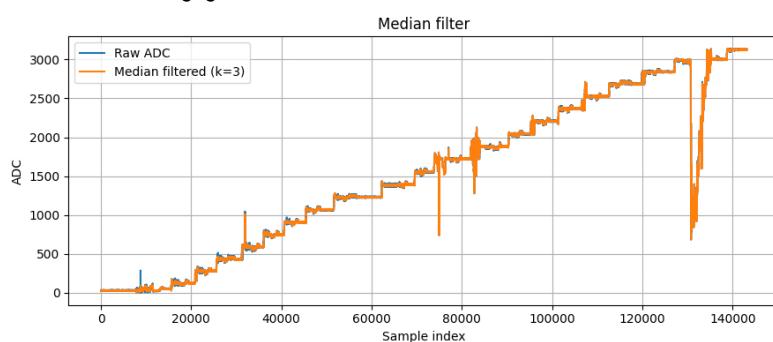


เพื่อจะนี้เราจะต้องทำการ preprocess ข้อมูลให้สามารถใช้งานได้โดยการ

### 1. กรองสัญญาณ (median filter) เพื่อลดสไปก์

เป้าหมาย: ทำให้สัญญาณเรียบขึ้นก่อนนำไปวิเคราะห์ แต่ยังคงค่ากลางของข้อมูลเดิมไว้

- ใช้ Median Filter เพื่อกำจัด “สัญญาณรบกวนเฉียบพลัน (spikes)” ที่เกิดจาก noise ในวงจร
- $y = \text{medfilt}(x, \text{kernel\_size}=3)$
- Kernel ขนาดเล็ก (3 จุด) เหมาะกับข้อมูล sampling ต่ำ ( $\sim 1 \text{ Hz}$ )
- Median filter จะไม่ทำให้สัญญาณล่าช้าเหมือน low-pass filter ทั่วไป



### 2. แบ่งสัญญาณเป็น 21 ช่วง (change-point detection)

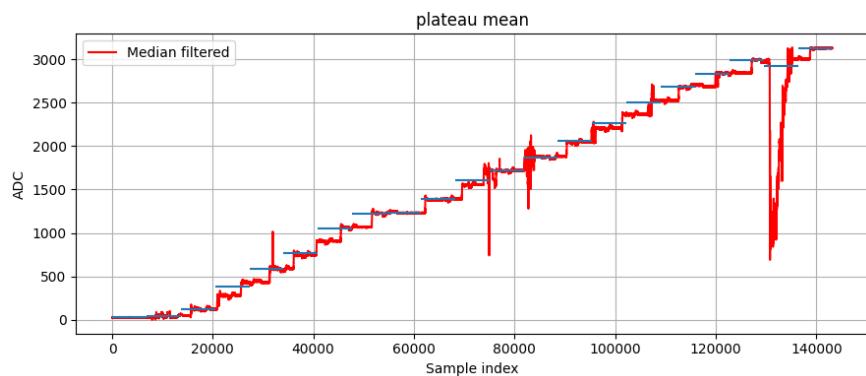
เป้าหมาย: ระบุขอบเขตของแต่ละสเต็ปน้ำหนักในสัญญาณที่ต่อเนื่องจาก Simulink

- การทดลองเพิ่มน้ำหนักทีละ 0.5 kg (รวม 21 สเต็ปตั้งแต่ 0–10 kg)
- ดังนั้นต้อง “แยกช่วงนิ่ง” แต่ละชั้นออกจากกัน
- โค้ดจะ
  - คำนวณความต่างของสัญญาณ  $\text{diff}(y)$
  - หาจุดที่สัญญาณเปลี่ยนแปลงมากที่สุด (change points)
  - เลือก 20 จุดที่ห่างกันพอสมควร เพื่อแบ่งเป็น 21 ช่วง
  - ถ้าหากสัญญาณเปลี่ยนไม่ชัด  $\rightarrow$  แบ่งช่วงเท่า ๆ กันโดยอัตโนมัติ

### 3. เฉลี่ยค่าท้ายของแต่ละช่วง (plateau mean)

เป้าหมาย: “ได้ค่าเฉลี่ยที่แทนค่าจริงของสัญญาณในแต่ละสเตป”

- ในแต่ละช่วงนำหน้า จะเฉลี่ยเฉพาะ “ส่วนท้าย” ของสัญญาณ เช่น 35% สุดท้าย
- $\text{mean} = \text{np.mean}(y[\text{int}((1 - \text{tail\_frac}) * L):])$
- โดย  $\text{tail\_frac} = 0.35$
- ตัดช่วงต้นที่นำหน้ากันไปนิ่งออก
- เก็บเฉพาะค่าที่สัญญาณนิ่งแล้วหลังนำหน้ากันตั้งตัว



f.

### 4. รวมข้อมูลไฟล์ด้วยค่ามัธยฐาน (Median Across Files)

เป้าหมาย: “ได้ค่าตัวแทน ADC ของแต่ละสเตปที่ “เชื่อถือได้มากที่สุด”

- ข้อมูลแต่ละไฟล์อาจมี noise หรือค่าคลาดเคลื่อนต่างกัน
- จึงรวมข้อมูลจากทุกไฟล์ด้วย ค่ามัธยฐาน (median) แทนค่าเฉลี่ย (mean)
- $\text{adc\_vec} = \text{np.nanmedian}(M, \text{axis}=0)$
- ทบทวนต่อ outlier จากไฟล์ใดไฟล์หนึ่ง
- ค่ามัธยฐานสะท้อนพฤติกรรมที่แท้จริงของระบบโดยรวม

### 5. พิทเส้นตรง $W = a \cdot \text{ADC} + b$ และตรวจเชิงเส้น ( $R^2$ )

จะได้สมการเส้นตรงของค่าที่วัดจริงและจาก ADC มาเปรียบเทียบกัน

ตารางสรุปค่าพารามิเตอร์

เงื่อนไข	สมการ	$R^2$	Mean Error (%)	RMSE (g)
Linear (Vout)	$W = 3938x - 156$	0.9996	1.41	45
Simulink (ADC)	$W = 3.121x - 90.3$	0.9997	1.18	42