RMXplorer

LAB 1.4: Single Point Load Cell with INA125 Instrumentation Amplifier

สมาชิก

- นาย ตะวัน นพเกตุ 67340500014
- นาย นายพีรกานต์ ยู 67340500031
- นาย นายพุฒิพงศ์ วงษ์พงษ์ 67340500073

จุดประสงค์ (Objectives)

1. เพื่อศึกษาและอธิบายหลักการทำงานของระบบเครื่องชั่งดิจิทัล ซึ่งประกอบด้วย Load Cell, Strain Gauge, วงจร Wheatstone Bridge และ 2-Op-Amp Differential Amplifier ได้
2. เพื่อออกแบบวงจรขยายสัญญานโดยคำนวณหาอัตราขยาย (Gain) ที่เหมาะสม และทำความเข้าใจกระบวนการปรับปรุงคุณภาพสัญญาน (Signal Conditioning)
3. เพื่อสร้างแบบจำลองทางคณิตศาสตร์สำหรับสอบเทียบ (Calibration) ระบบ โดยแปลงค่า ADC ดิบ (Counts) ให้เป็นค่าน้ำหนักในหน่วยกิโลกรัม (kg)
4. เพื่อสร้างแบบจำลอง Simulink ที่สามารถประมวลผลและแสดงค่าน้ำหนักได้แบบ Real-time พร้อมฟังก์ชันตั้งศูนย์ (Tare)
5. เพื่อทดสอบและประเมิณค่าความแม่นยำ (Accuracy) และความเที่ยงตรง (Precision) ของเครื่องชั่งที่สร้างขึ้น โดยเปรียบเทียบกับเครื่องชั่งดิจิทัลมาตรฐาน และวิเคราะห์สาเหตุของความคลาดเคลื่อน

หลักการทำงานและทฤษฎีที่เกี่ยวข้อง

การสร้างเครื่องชั่งดิจิทัลโดยใช้ Load Cell อาศัยการทำงานร่วมกันขององค์ประกอบทางไฟฟ้าและเครื่องกลหลายส่วน เพื่อแปลงปริมาณทางกาภาพ (แรง/น้ำหนัก) ให้กลายเป็นสัญญานดิจิทัลที่สามารถประมวณผลได้ โดยมีหลักการทำงานในแต่ละส่วนดังนี้

1. **Strain Gauge และ Load Cell**

   Strain Gauge เป็นเซนเซอร์ที่ทำจากแผ่นฟอยล์โลหะบางๆ ที่มีความต้านทานไฟฟ้า เมื่อถูกแรงกระทำให้ยืดออก (Tension) ความต้านทานจะเพิ่มขึ้น เมื่อถูกบีบอัด (Compression) ความต้านทานจะลดลง หลักการนี้ถูกนำมาใช้ใน Load Cell ซึ่งเป็นโครงสร้างโลหะที่ออกแบบมาให้เกิดการบิดตัวอย่างแม่นยำเมื่อมีแรงมากระทำ โดยมีการติดตั้ง Strain Gauge ไว้ 4 ตัวในตำแหน่งที่คำนวณมาอย่างดี (2 ตัวในตำแหน่งที่เกิด Tension และอีก 2 ตัวในตำแหน่งที่เกิด Compression)

2. **วงจร Wheatstone Bridge**

   Strain Gauge ทั้ง 4 ตัวใน Load Cell ถูกนำมาต่อเข้าด้วยกันในรูปแบบของวงจร Wheatstone Bridge ซึ่งเป็นวงจรที่ไวต่อการเปลี่ยนแปลงความต้านทานที่น้อยมาก

   a. **สภาวะสมดุล (No Load):** เมื่อไม่มีน้ำหนักมากระทำ Strain Gauge ทั้ง 4 ตัวจะมีความต้านทานใกล้เคียงกัน ทำให้วงจรอยู่ในสภาวะสมดุล และความต่างศักย์ไฟฟ้าที่ขั้ว Output (A+ และ A-) จะมีค่าใกล้เคียง 0 โวลต์ (V)

   b. **สภาวะไม่สมดุล (Under Load):** เมื่อมีน้ำหนักมากระทำ Strain Gauge 2 ตัวจะมีความต้านทานเพิ่มขึ้น ในขณะที่อีก 2 ตัวมีความต้านทานลดลง ทำให้วงจรเสียสมดุลและเกิด สัญญานความต่างศักย์ไฟฟ้า (Differential Voltage) ขนาดเล็กมาก (ระดับมิลลิโวลต์-mV) ขึ้นที่ขั้ว Output โดยขนาดของแรงดันไฟฟ้าที่เกิดขึ้นนี้จะแปรผันตรงกับน้ำหนักที่กระทำต่อ Load Cell ซึ่งเป็นหลักการสำคัญในการวัดน้ำหนัก

3. **2-Op-Amp Differential Amplifier**

   เป็นวงจรขยายสัญญาณต่างศักย์ที่ใช้โอเปแอมป์สองตัว โดยแนวคิดหลักคือ การขยายอินพุตทั้งสองด้านในโหมดไม่กลับเฟส (non-inverting) ด้วย Gain ที่เท่าๆ กัน แล้วหักลบกัน

- โครงสร้างมาตรฐาน: แต่ละอินพุต (V_1, V_2) ป้อนเข้า op-amp ของตัวเอง (Gain A) จากนั้นนำเอาต์พุตทั้งสองมาหักลบเพื่อได้ $V_{out} \propto A\ (V_2 - V_1)$
- **สมการ Gain (กรณีค่าตัวต้านทานสมมาตร)** มักออกมาในรูปทั่วไป เช่น

$$V_{out} = \left(1 + \frac{R}{R_g}\right)(V_2 - V_1)$$

- **ข้อดี:** วงจรง่าย อุปกรณ์น้อย Gain ปรับได้
- **ข้อจำกัด:** ค่าความต้านทานต้องเข้ากันมาก มิฉะนั้นจะปล่อยสัญญาณโหมดร่วม (รบกวน/Off-set) เข้ามาได้ง่าย โดยประมาณ

4. **ความสัมพันธ์ระหว่าง Gain และความต้านทาน (RG)**

สำหรับ IC INA125 อัตราขยาย (Gain) ของวงจรสามารถกำหนดได้ง่ายด้วยตัวต้านทานภายนอกเพียงตัวเดียว (RG) ตามสมการที่ระบุใน Datasheet: $G = 4 + (60{,}000/RG)$ จากสมการนี้ จะเห็นได้ว่าค่า RG มีความสัมพันธ์แบบ แปรผกผัน (Inverse Relationship) กับ Gain กล่าวคือ:

- เมื่อ **ลด** ค่า RG -> อัตราขยาย G จะ สูงขึ้น (ระบบมีความไวต่อแรงกดมากขึ้น)
- เมื่อ **เพิ่ม** ค่า RG -> อัตราขยาย G จะ ต่ำลง

ในการทดลองนี้ ได้มีการคำนวณและเลือกใช้ค่า RG ประมาณ 432 โอห์ม เพื่อให้ได้ Gain ประมาณ 143 ซึ่งเป็นค่าที่เหมาะสมในการขยายสัญญาณตลอดช่วงการวัด 0-10 kg โดยไม่ทำให้เกิด Saturation ที่แรงดัน Output

การตั้งค่าฮาร์ดแวร์และการคำนวณ Gain

เพื่อให้ระบบสามารถวัดน้ำหนักได้ตลอดช่วง 0-10 kg โดยไม่เกิดปรากฏการณ์ Saturation (การอิ่มตัวของสัญญาน) จึงจำเป็นต้องคำนวณหาอัตราขยาย (Gain) ที่เหมาะสมสำหรับวงจร Instrumentation Amplifier (INA125) ซึ่งสามารถทำได้สองวิธี ได้แก่:

1. **การคาดเดาจาก Datasheet**

วิธีนี้เป็นการคำนวณเบื้องต้นโดยอาศัยค่าคุณสมบัติมาตราฐานที่ระบุไว้ใน Datasheet ของ Load Cell รุ่น YZC-131A จะมีค่า Rated Ouput อยู่ที่ 1.0 mV/V ซึ่งหมายความว่าไฟเลี้ยง 5V และน้ำหนัก

เต็มพิกัด 10 kg จะให้สัญญาน V_{in} สูงสุดประมาณ 5 mV ($1.0\ mV * 5$) เมื่อนำไปคำนวณหา Gain ที่จะขยายสัญญาน 5 mV ไปเป็น 3.0 V จะได้ Gain เท่ากับ 600 และคำนวณค่า R_G ได้ประมาณ 100 Ω วิธีนี้รวดเร็วแต่มีความไม่แน่นอนค่อนข้างสูง เนื่องจากค่า Rated Output ของ Load Cell แต่ละตัวอาจไม่ตรงกับ Datasheet เสมอไป (สาเหตุอาจมาจากหลายปัจจัย เช่น ความไม่สมบูรณ์ของวัสดุ ความคลาดเคลื่อนในการผลิต เป็นต้น)

2. **การวัดค่าสัญญาน Input จริง**

วิธีนี้ให้ความแม่นยำมากกว่าวิธีที่ 1 จึงเป็นวิธีที่เลือกใช้ในการทดลองนี้ โดยเป็นการวัดสัญญานไฟฟ้าที่เกิดขึ้นจริงจาก Load Cell โดยตรง แล้วนำไปคำนวณหาค่า Gain ที่เหมาะสมที่สุดสำหรับอุปกรณ์ชุดนี้ โดยมีขั้นตอนดังนี้:

a. วัดความต่างศักย์ไฟฟ้า (V_{in}) ที่ขั้ว Input (V_{in+} เทียบกับ V_{in-}) ของ INA125 โดยตรงด้วยมัลติมิเตอร์
b. ผลการวัดที่น้ำหนัก 0 kg (V_{in_zero}) คือ 0 mV และที่น้ำหนัก 1 kg (V_{in_ref}) คือ 2.1 mV
c. จากข้อมูลนี้ สามารถประมาณการ V_{in} สูงสุดที่ 10 kg ได้เท่ากับ 21 mV ($2.1\ mV * 10$)
d. ตั้งเป้าหมายแรงดัน Output สูงสุด (V_{out_max}) ไว้ที่ 3.0 V เพื่อให้มี Safety Margin ก่อนถึงขีดจำกัด 3.3 V ของ ADC จะสามารถคำนวณ Gain ที่ต้องการได้จาก:

$$Gain = \frac{V_{out_max}}{V_{in_max}} = \frac{3.0\ V}{0.021\ V} \approx 143$$

e. นำค่า Gain ที่ได้มาคำนวณหาค่าความต้านทาน R_G ที่ต้องใช้ จากสมการของ INA125:

$$R_G = \frac{60{,}000}{Gain - 4} = \frac{60{,}000}{143 - 4} \approx 432\ \Omega$$

การทดลองที่ 1: การสอบเทียบระบบ (System Calibration)

1. จุดประสงค์

1.1. เพื่อสร้างแบบจำลองคณิตศาสตร์ (Calibration Equation) ที่สามารถแปลงค่า ADC ดิบ (Counts) ให้เป็นค่าน้ำหนักจริง (กิโลกรัม) ได้อย่างแม่นยำ

1.2. เพื่อประเมิณความเป็นเชิงเส้น (Linearity) ของการตอบสนองของระบบ Load Cell

2. สมมติฐาน

2.1. ความสัมพันธ์ระหว่างค่า ADC ที่อ่านได้และน้ำหนักจริง สามารถประมาณได้ด้วยสมการเส้นตรง ($y = mx + c$)

3. ตัวแปร

3.1. **ตัวแปรต้น:** น้ำหนักมาตราฐาน (0-10kg)

3.2. **ตัวแปรตาม:** ค่า ADC ที่อ่านได้ (0-4095)

3.3. **ตัวแปรควบคุม:** อัตราขยาย (Gain) ของวงจร, อุณหภูมิห้อง

4. ขั้นตอนการดำเนินงาน

4.1. ตั้งค่าฮาร์ดแวร์โดยปรับค่า RG ของ INA125 ไว้ที่ประมาณ 432 Ω เพื่อให้ได้ Gain ที่เหมาะสม

4.2. สร้างแบบจำลอง Simulink เพื่ออ่านค่า ADC จาก Pin PA0 พร้อมใช้ฟิลเตอร์ Moving Average

4.3. ทำการเก็บข้อมูลโดยวางน้ำหนักมาตรฐานตั้งแต่ 0kg ถึง 10kg ทีละระดับ และบันทึกค่า ADC ที่อ่านได้ในแต่ละระดับ

4.4. นำชุดข้อมูล (ADC Value vs True Weight) ไปประมวณผลใน MATLAB โดยใช้คำสั่ง polyfit (degree 1) เพื่อหาสมการเส้นตรงและค่า R-Squared

5. **ผลการทดลอง**

ตารางที่ 1 ข้อมูล ADC Value ที่ได้ในแต่ละน้ำหนัก (ดูในภาคผนวก)

สมการสอบเทียบที่ได้คือ: $Weight(kg) = 5.30 * ADC - 218.29$

ค่า R-squared ที่คำนวณได้คือ **0.995850**

กราฟที่ 1 กราฟ Calibration Curve (Linear Fit) (ดูในภาคผนวก)

6. สรุปผลการทดลอง

6.1. สามารถสร้างสมการสอบเทียบแบบเส้นตรงเพื่อแปลงค่า ADC เป็นน้ำหนักได้สำเร็จ โดยค่า R-squared ที่สูง (0.995850) บ่งชี้ว่าระบบมีการตอบสนองที่เป็นเชิงเส้นสูง

7. อภิปรายผล

7.1. ค่า R-squared ที่เข้าใกล้ 1 ยืนยันสมมติฐานที่ว่าความสัมพันธ์ระหว่าง ADC และน้ำหนักเป็นเส้นตรงได้เป็นอย่างดี อย่างไรก็ตาม การที่ค่าไม่เป็น 1 พอดี แสดงให้เห็นถึงความไม่เป็นเชิงเส้น (Non-linearity)

เล็กน้อย จากการมี Noise ปะปนอยู่ในข้อมูล เช่น อุณหภูมิที่ไม่คงที่ การสั่นสะเทือน แหล่งจ่ายไฟไม่เสถียร เป็นต้น ซึ่งอาจส่งผลต่อความแม่นยำในการทำนายผล

การทดลองที่ 2: การประเมิณความแม่นยำของแบบจำลองการสอบเทียบ (Calibration Model Verification)

1. **จุดประสงค์:**
   1.1. เพื่อประเมินความแม่นยำ (Accuracy) ของแบบจำลองการสอบเทียบแบบเส้นตรง (Linear Model) ที่สร้างขึ้นในการทดลองที่ 1
   1.2. เพื่อวิเคราะห์ขนาดและแนวโน้มของความคลาดเคลื่อน (Error) ที่เกิดขึ้นตลอดช่วงการวัด
   1.3. เพื่อวิเคราะห์สาเหตุของความคลาดเคลื่อนที่เกิดขึ้น
2. **สมมติฐาน:**
   2.1. แบบจำลองเส้นตรงที่สร้างขึ้น จะสามารถทำนายค่าน้ำหนักได้ใกล้เคียงกับค่าจริง แต่จะมีความคลาดเคลื่อนเกิดขึ้นเนื่องจากความไม่เป็นเชิงเส้น (Non-linearity) ของเซ็นเซอร์และสัญญาณรบกวน (Noise)
   2.2. ความคลาดเคลื่อน (% Error) จะไม่คงที่ แต่จะมีการเปลี่ยนแปลงไปในแต่ละช่วงน้ำหนัก
3. **ตัวแปร:**
   3.1. **ตัวแปรต้น:** ค่าน้ำหนักจริงจากเครื่องชั่งดิจิทัล (Digtital Weight)
   3.2. **ตัวแปรตาม:** ค่าน้ำหนักที่ทำนายโดยแบบจำลอง (Predicted Weight), ค่าความคลาดเคลื่อน (% Error)
   3.3. **ตัวแปรควบคุม:** แบบจำลองทางคณิตศาสตร์ (ใช้สมการเส้นตรงจากการทดลองที่ 1), อัตราขยาย (Gain) ของวงจร, อุณหภูมิห้อง
4. **ขั้นตอนการดำเนินงาน:**
   4.1. สร้างแบบจำลอง Simulink ที่สมบูรณ์โดยใช้สมการเส้นตรง $Weight(kg) = 5.30 * ADC - 218.29$ ที่ได้จากการทดลองที่ 1
   4.2. ทำการทดลองเก็บข้อมูลชุดใหม่โดยการชั่งน้ำหนักมาตรฐานตั้งแต่ 0kg ถึง 10kg ทีละระดับ (ระดับละ 0.5kg) และบันทึก ค่าน้ำหนักที่วัดได้ จาก Display ของ Simulink

4.3. สร้างตารางเปรียบเทียบระหว่าง ค่าน้ำหนักจริง กับ ค่าน้ำหนักที่วัดได้ (ชุดใหม่) และคำนวณหาค่า Absolute Error และ Percentage Error

4.4. คำนวณหาค่า Percentage Error เฉลี่ย

รูปที่ 1 Simulink Setup สำหรับการ Calibrated (ดูในภาคผนวก)

5. ผลการทดลอง:

ตารางที่ 2 Verification (Linear Fit บนข้อมูลใหม่) (ดูในภาคผนวก)

ความคลาดเคลื่อนเฉลี่ย (Average Percentage Error) ที่คำนวณได้คือ **4.46%**

กราฟที่ 2 Verification: Measured vs. True (Linear Fit) (ดูในภาคผนวก)

สมการสอบเทียบที่ได้คือ: $Weight(kg) = 5.30 * ADC - 218.29$

ค่า R-squared ที่คำนวณได้คือ **0.995850**

กราฟที่ 3 Calibration Curve (Linear Fit) (ดูในภาคผนวก)

6. สรุปผลการทดลอง:

6.1. เมื่อนำแบบจำลองเส้นตรงไปทดสอบกับชุดข้อมูลใหม่ พบว่ายังคงสามารถทำนายค่าน้ำหนักได้ โดยมีความคลาดเคลื่อนเฉลี่ย 4.46%

7. อภิปรายผล:

7.1. ผลการทดลองยืนยันว่าแบบจำลองเส้นตรงที่สร้างขึ้นมีความสามารถในการ Generalize (จำลองค่า) กับข้อมูลใหม่ได้ในระดับหนึ่ง

7.2. ความคลาดเคลื่อน 4.46% ที่เกิดขึ้น มีปัจจัยที่แบบจำลองเส้นตรงไม่สามารถอธิบายได้ครบถ้วน ได้แก่ Non-linearity ของ Load Cell และ Random Error/Drift ที่เกิดขึ้นแตกต่างกันในแต่ละรอบการทดลอง ซึ่งเห็นได้จากแนวโน้มของ Error ที่ยังคงมีค่าสูงในช่วงน้ำหนักน้อยและเปลี่ยนแปลงไปตลอดช่วงการวัด

การทดลองที่ 3: การพัฒนาระบบด้วยฟังก์ชัน Tare และการปรับปรุงแบบจำลอง (System Enhancement)

1. **จุดประสงค์:**
   1.1. เพื่อออกแบบและนำฟังก์ชัน Tare (การตั้งศูนย์) ไปใช้ในแบบจำลอง Simulink เพื่อแก้ไขปัญหา Systematic Offset Error
   1.2. เพื่อทดลองปรับปรุงความแม่นยำโดยใช้แบบจำลองการสอบเทียบที่ซับซ้อนขึ้น (Polynomial Regression) เพื่อชดเชย Non-linearity
2. **สมมติฐาน:**
   2.1. การเพิ่มฟังก์ชัน Tare จะทำให้ค่าที่วัดได้ที่น้ำหนัก 0 กรัม มีค่าใกล้เคียง 0 มากขึ้น
   2.2. การใช้แบบจำลอง Polynomial Regression จะให้ค่า R-squared ที่สูงขึ้น และลดค่าความคลาดเคลื่อนเฉลี่ย (% Error) ลง เมื่อเทียบกับแบบจำลองเส้นตรง
3. **ตัวแปร:**
   3.1. **ตัวแปรต้น:** แบบจำลองทางคณิตศาสตร์ที่ใช้ (Linear vs. Polynomial), การมี/ไม่มีฟังก์ชัน Tare
   3.2. **ตัวแปรตาม:** ค่า R-squared, ค่าความคลาดเคลื่อนเฉลี่ย (% Error)
   3.3. **ตัวแปรควบคุม:** อัตราขยาย (Gain) ของวงจร, อุณหภูมิห้อง
4. **ขั้นตอนการดำเนินงาน:**
   4.1. **การทำ Tare:** สร้างแบบจำลอง Simulink ที่สมบูรณ์ โดยเพิ่ม MATLAB Function ที่มี Logic การ Tare แบบกดปุ่ม (ใช้ Digital Port Read จาก PC13) เข้าไป เพื่อให้สามารถหักล้างค่า Offset ณ จุดเริ่มต้นได้
   4.2. **การปรับปรุงแบบจำลอง:** นำข้อมูล Calibration เดิมไปประมวลผลใหม่ด้วย polyfit (degree 2) เพื่อหาสมการพหุนาม $y = ax^2 + bx + c$
   4.3. **การประเมินผล:** นำสมการพหุนามที่ได้ไปทดสอบกับข้อมูล Calibration ชุดข้อมูลใหม่เพื่อคำนวณหา % Error เฉลี่ย และเปรียบเทียบกับผลของแบบจำลองเส้นตรง
5. **ผลการทดลอง:**

5.1. Tare Function:

รูปที่ 2 โมเดล Simulink ฟังก์ชัน Tare และปุ่มกด (ดูในภาคผนวก)

ผลการทดลองพบว่าเมื่อกดปุ่ม Tare ระบบสามารถตั้งค่าศูนย์ (0kg) ได้สำเร็จ

5.2. Polynomial Model:

สมการที่ได้คือ $Weight(kg) = -0.00065 * ADC^2 + 6.52 * ADC - 620.32$ โดยมีค่า R^2 สูงขึ้นเป็น **0.999057**

เมื่อนำไปทดสอบกับข้อมูลใหม่ พบว่ามีค่าความคลาดเคลื่อนเฉลี่ยลดลงเหลือ **3.32%**

ตารางที่ 3 Verification (Polynomial Fit on Calibration Data) (ดูในภาคผนวก)

กราฟที่ 4 Calibration Curve (Polynomial Fit) (ดูในภาคผนวก)

6. สรุปผลการทดลอง:

6.1. การเพิ่มฟังก์ชัน Tare สามารถแก้ปัญหา Offset Error ได้

6.2. การใช้แบบจำลอง Polynomial สามารถลดความคลาดเคลื่อนเฉลี่ยลงได้จาก 4.46% เหลือ 3.32% ซึ่งเป็นการปรับปรุงที่มีนัยสำคัญ

7. อภิปรายผล:

7.1. การที่แบบจำลอง Polynomial ให้ผลลัพธ์ที่มีความแม่นยำสูงกว่า สนับสนุนสมมติฐานที่ว่า Load Cell มีคุณลักษณะแบบไม่เป็นเชิงเส้น (Non-linearity) แฝงอยู่เล็กน้อย และแบบจำลองที่ซับซ้อนมากกว่าสามารถอธิบายพฤติกรรมดังกล่าวได้ดียิ่งขึ้น อย่างไรก็ตาม การที่ยังคงมีความคลาดเคลื่อนเหลืออยู่สะท้อนให้เห็นถึงข้อจำกัดของระบบวัด ซึ่งมีสาเหตุหลักมาจากสัญญานรบกวน (Noise) และความไม่แน่นอนอื่นๆ ในการกระบวนการวัด

การทดลองที่ 4: การวิเคราะห์ปรากฏการณ์ Saturation

1. จุดประสงค์:

1.1. เพื่อศึกษาและวิเคราะห์พฤติกรรมของระบบเมื่อมีแรงกดกระทำเกินกว่าช่วงการวัดที่ออกแบบไว้

1.2. เพื่อพิสูจน์ให้เห็นถึงขีดจำกัดทางกายภาพของ Analog-to-Digital Converter (ADC)

2. สมมติฐาน (Hypothesis):

2.1. เมื่อแรงดันไฟฟ้าที่ Input ของ ADC มีค่าสูงเกินกว่าแรงดันอ้างอิง (V_{ref}) ค่าดิจิทัลที่ ADC แปลงได้ (ADC Counts) จะอิ่มตัวและคงที่ อยู่ที่ค่าสูงสุดที่เป็นไปได้ (4095) โดยจะไม่เพิ่มขึ้นอีกแม้ว่าแรงดัน Input จะเพิ่มขึ้นต่อไปก็ตาม

3. **ตัวแปร (Variables):**

   3.1. **ตัวแปรต้น:** แรงกดที่กระทำต่อ Load Cell (โดยจะเพิ่มขึ้นเรื่อยๆ จนเกินช่วงการวัด)

   3.2. **ตัวแปรตาม**: ค่า ADC ที่อ่านได้ (Counts)

   3.3. **ตัวแปรควบคุม:** อัตราขยาย (Gain) ของวงจรที่ถูกปรับให้สูงขึ้นชั่วคราว (โดยลดค่า RG เหลือ ~20 Ω)

4. **ขั้นตอนการดำเนินงาน:**

   4.1. **ปรับค่า Gain:** ปรับค่า RG ชั่วคราวให้มีค่าต่ำลง (ประมาณ 20 Ω) เพื่อเพิ่ม Gain ของวงจรให้สูงขึ้น เป็นการจำลองสภาวะ Overload

   4.2. **สร้างแบบจำลอง:** ใช้โมเดล ADC -> Data Type Conversion -> Moving Average -> Display เพื่ออ่านค่า ADC ดิบ

รูปที่ 3 Simulink Setup (ดูในภาคผนวก)

5. **ผลการทดลอง:**

รูปที่ 4 Simulink Display ADC Counts (0-4095) (ดูในภาคผนวก)

พบว่าค่า ADC จะเพิ่มขึ้นตามแรงกดจนถึงค่าสูงสุดคือ 4095 แล้วจะคงที่อยู่ ณ ค่านั้น ไม่ว่าจะมีแรงกดเพิ่มขึ้นอีกก็ตาม

6. **สรุปและอภิปรายผล:**

   6.1. ปรากฏการณ์นี้คือ Saturation (การอิ่มตัว) ซึ่งเกิดจากการที่แรงดันไฟฟ้าที่ออกจาก Amplifier มีค่าสูงเกินกว่าแรงดันอ้างอิง (3.3 V) ของ ADC ทำให้ ADC ไม่สามารถแยกแยะค่าที่สูงไปกว่านี้ได้อีก ซึ่งเป็นการแสดงให้เห็นถึงขีดจำกัดทางกายภาพของช่วงการวัดในระบบ

ภาคผนวก

1. ตารางข้อมูลการทดลอง

- **ตารางที่ 1**: ข้อมูล ADC Value ที่ได้ในแต่ละน้ำหนัก

True Weight (kg)	ADC Value
0	112
0.5	147
1	244
1.5	340
2	450
2.5	509
3	592
3.5	682
4	763
4.5	847
5	930
5.5	1016
6	1100
6.5	1185
7	1270
7.5	1356
8	1446
8.5	1532
9	1619
9.5	1707
10	1790

- **ตารางที่ 2**: Verification (Linear Fit บนข้อมูลใหม่)

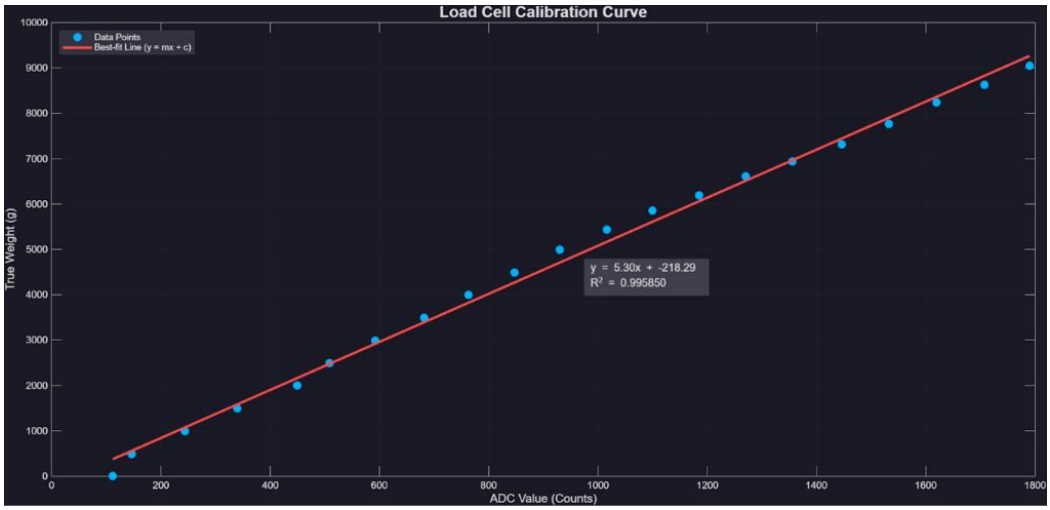
Digital Weight (kg)	Measured Weight (kg)	Absolute Error	% Error
0	0.189	189	N/A (Noise)
0.49	0.402	-88	17.96%
0.995	0.944	-51	5.13%
1.496	1.454	-42	2.81%
2.002	1.913	-89	4.45%
2.494	2.385	-109	4.37%
2.991	2.852	-139	4.65%
3.489	3.317	-172	4.93%
3.995	3.780	-215	5.38%
4.487	4.228	-259	5.77%
4.996	4.685	-311	6.22%
5.434	5.142	-292	5.37%
5.851	5.612	-239	4.08%
6.196	6.077	-119	1.92%
6.608	6.554	-54	0.82%
6.943	7.039	96	1.38%
7.320	7.508	188	2.57%
7.766	7.982	216	2.78%
8.242	8.425	183	2.22%
8.628	8.895	267	3.09%
9.050	9.348	298	3.29%

- **ตารางที่ 3:** Verification (Polynomial Fit on Calibration Data)

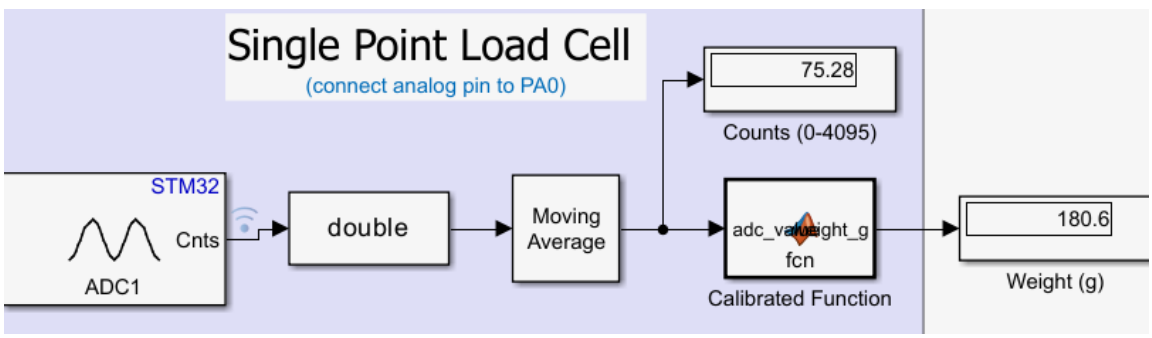
Digital Weight (kg)	"Measured Weight (kg) [New Data]"	Absolute Error	% Error
0	0.084	0.8	N/A (Noise)
0.490	0.511	21	4.29%
0.995	0.958	-37	-3.72%
1.496	1.449	-47	-3.14%
2.002	1.932	-70	-3.50%
2.494	2.582	88	3.53%
2.991	3.074	83	2.77%
3.489	3.393	-96	-2.75%
3.995	3.864	-131	-3.28%
4.487	4.355	-132	-2.94%
4.996	5.118	122	2.44%
5.434	5.594	160	2.94%
5.851	5.659	-192	-3.28%
6.196	6.025	-171	-2.76%
6.608	6.425	-183	-2.77%
6.943	7.187	244	3.51%
7.320	7.577	257	3.51%
7.766	7.524	-242	-3.12%
8.242	7.937	-305	-3.70%
8.628	8.252	-376	-4.36%
9.050	9.420	370	4.09%

2. รูปผลการทดลอง และแบบจำลอง Simulink

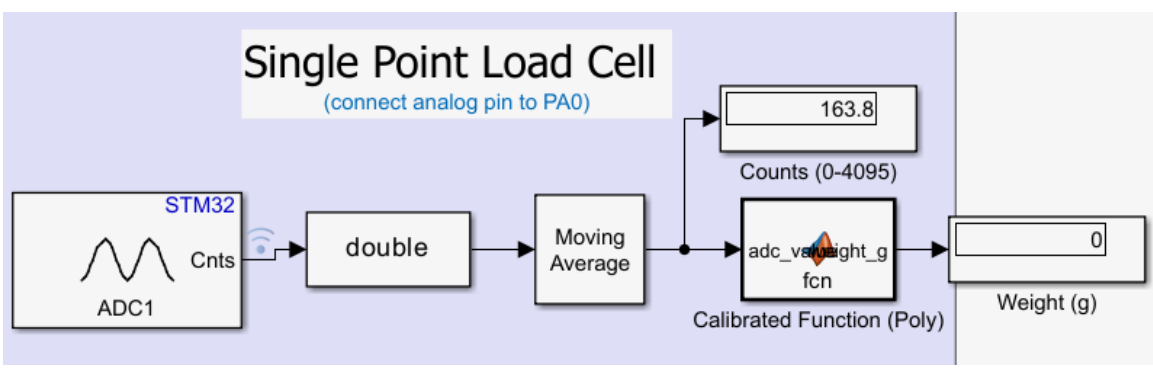
- **กราฟที่ 1:** กราฟ Calibration Curve (Linear Fit)

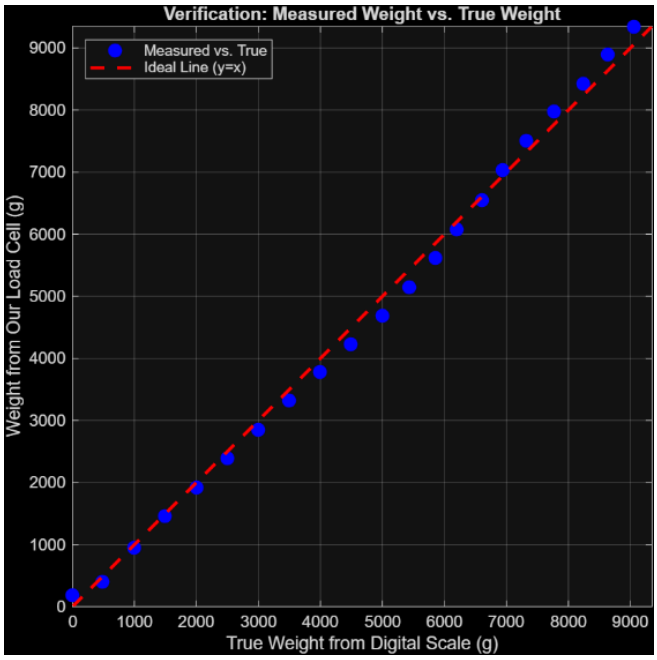


- **รูปที่ 1:** Simulink Setup สำหรับการ Calibrated

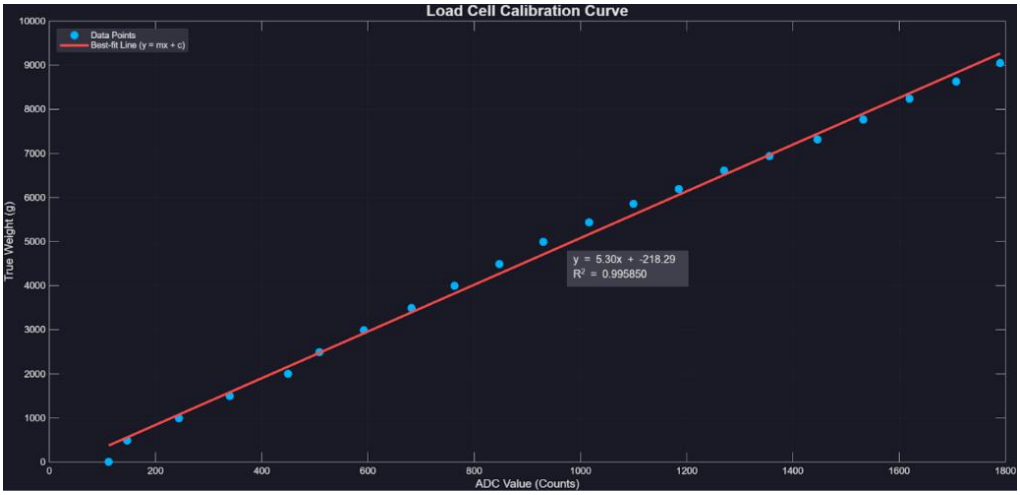


- **กราฟที่ 2:** Verification: Measured vs. True (Linear Fit)

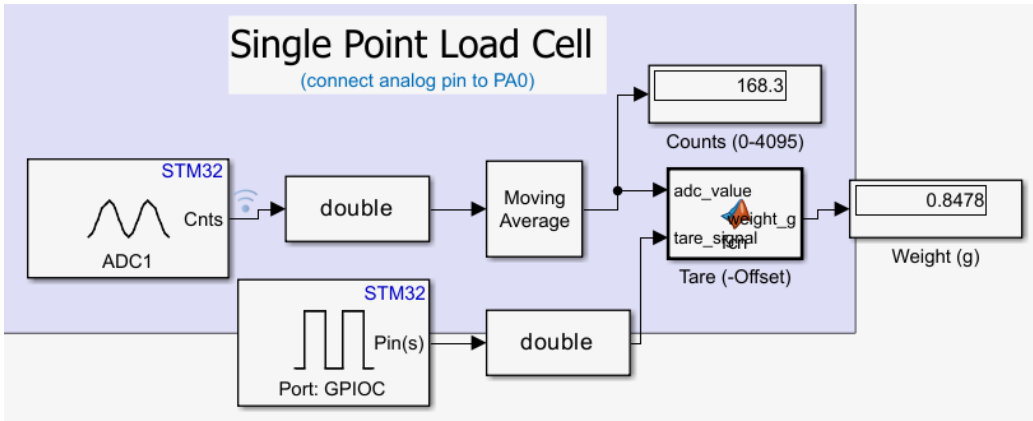




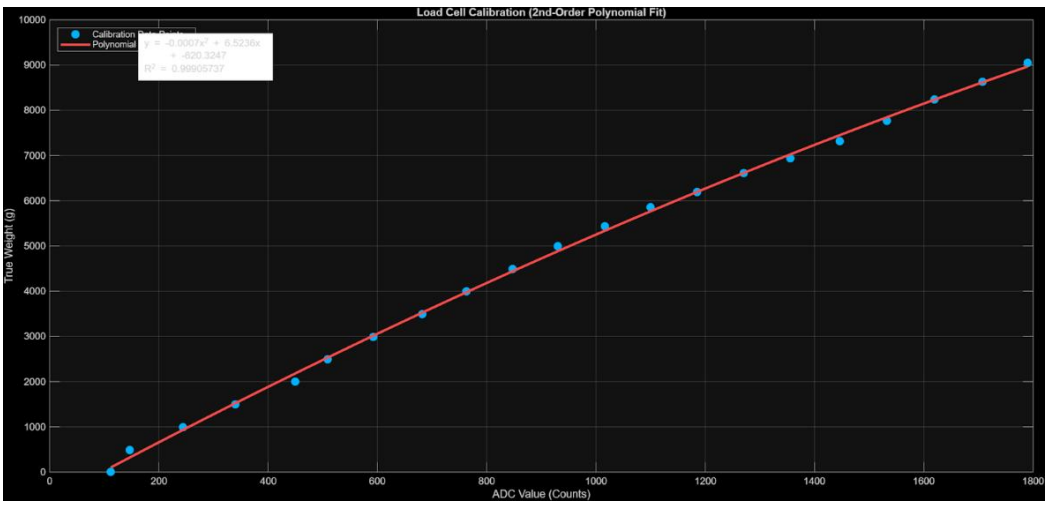
- **กราฟที่ 3:** Calibration Curve (Linear Fit)



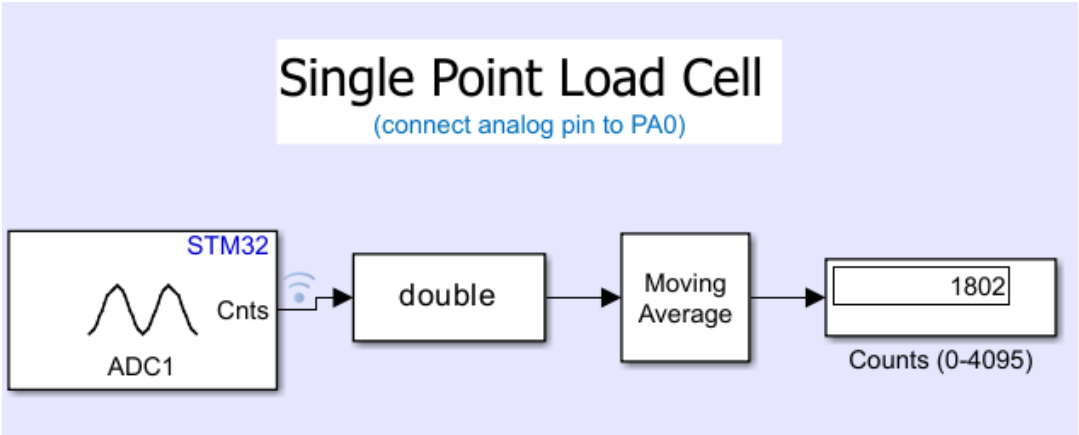
- **รูปที่ 2:** โมเดล Simulink ฟังก์ชัน Tare และปุ่มกด



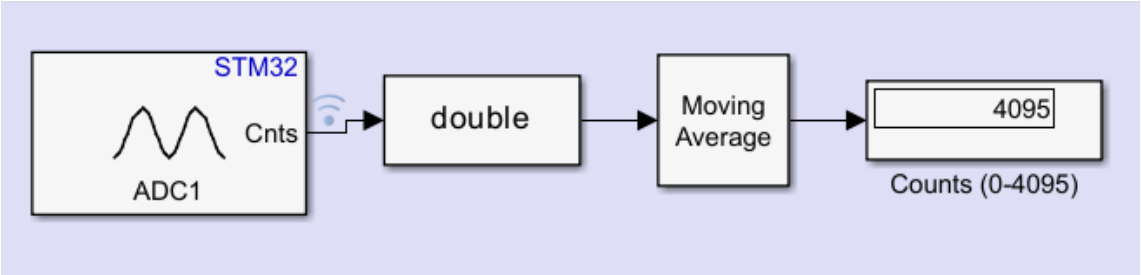
- **กราฟที่ 4:** Calibration Curve (Polynomial Fit)



- **รูปที่ 3:** Simulink Setup



- **รูปที่ 4:** Simulink Display ADC Counts (0-4095)



3. โค้ด MATLAB Function

- MATLAB Function (สำหรับ Simulink) — Tare แบบกดปุ่ม

```
function y = fcn(adc, tareBtn)

a = -0.000652; b = 6.523590; c = -620.324675;

raw = a*adc*adc + b*adc + c;

% Tare เมื่อกดปุ่ม (rising edge)
if (tareBtn==1) && (prevBtn==0)
    off = raw;
end
prevBtn = tareBtn;

y = raw - off;            % ชดเชย offset
if y < 0; y = 0; end      % กันค่าติดลบ
y = round(y);             % ต้องการจำนวนเต็ม
end
```

- สคริปต์คาลิเบรต (Polynomial Fit)

```
% === Calibration (deg-2) ===
clear; clc;

% กรณีอ่านจากตาราง/ไฟล์ ให้แทนค่าด้านล่าง
adc = [ ... ];            % ค่าที่อ่านได้ตอนคาลิเบรต
true_g = [ ... ];         % น้ำหนักจากเครื่องชั่งดิจิทัล (g)

p = polyfit(adc, true_g, 2);      % p = [a b c]
a=p(1); b=p(2); c=p(3);

pred = polyval(p, adc);
R2 = 1 - sum((true_g - pred).^2)/sum((true_g-mean(true_g)).^2);

fprintf('Weight(g) = %.6f*ADC^2 + %.6f*ADC + %.6f\n', a,b,c);
fprintf('R^2 = %.6f\n', R2);

figure; plot(adc, true_g,'o'); hold on;
x = linspace(min(adc), max(adc), 400);
plot(x, polyval(p,x), 'LineWidth',2);
xlabel('ADC (counts)'); ylabel('Weight (g)'); grid on;

% บันทึกค่าสมการไปใช้ใน Simulink (โหลดใช้ได้ทันที)
save calib_coeffs.mat a b c
```

- สคริปต์ตรวจสอบผล (Verification / %Error, Residuals)

```
% === Verification ===
clear; clc;

true_g = [ ... ];            % น้ำหนักจริงจาก Digital scale
calc_g = [ ... ];            % น้ำหนักที่วัดได้จากระบบเรา (หลังคาลิเบรต)
err_g  = calc_g - true_g;
den    = true_g; den(den==0) = NaN;
perr   = 100*abs(err_g)./den;

mape = mean(perr(~isnan(perr)));
fprintf('Average |%%Error| = %.2f%%\n', mape);

figure; plot(true_g, calc_g,'o'); hold on;
mx = max([true_g; calc_g]); plot([0 mx],[0 mx],'--'); axis equal;
xlabel('True (g)'); ylabel('Measured (g)'); grid on;

figure; plot(true_g, err_g,'o-'); yline(0,'--');
xlabel('True (g)'); ylabel('Residual (g)'); grid on;
```

เอกสารอ้างอิง

- Wikipedia – Load Cell หลักการ, การใช้บริดจ์, ขยายสัญญาณ
  https://th.wikipedia.org/wiki/%E0%B9%82%E0%B8%AB%E0%B8%A5%E0%B8%94%E0%B9%80%E0%B8%8B%E0%B8%A5
- Factomart - Load Cell คืออะไร?
  https://mall.factomart.com/what-is-load-cell/

- KMUTech - Strain Gauge (พื้นฐาน/โครงสร้าง/การเปลี่ยนค่าความต้านทาน)

  https://www.kmutech.com/strain-gauge/

- Micro-Tess - คู่มือ Wheatstone Bridge (ทำงาน/ข้อดีข้อจำกัด/การใช้งานกับเซนเซอร์)

  https://www.micro-tess.com/th/wheatstone-bridge/

- INA125 - Datasheet

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