

เอกสารทดลอง 2

HIL & PID Controller

Part I: Hardware-in-the-loop (HIL)

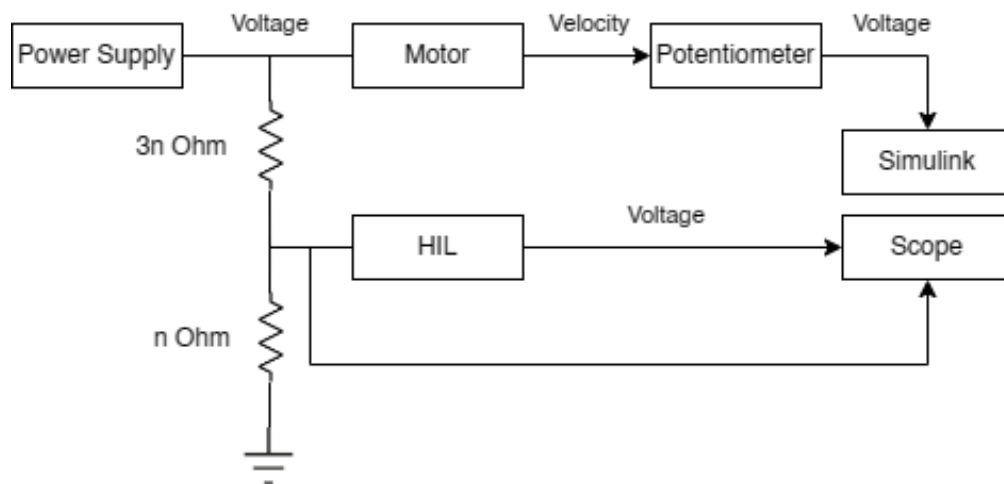
วัตถุประสงค์

เพื่อให้นักศึกษาสามารถจำลองพฤติกรรมการทำงานของระบบผ่าน Hardware-in-the-loop ได้

อุปกรณ์

1. RMX Board
2. Brushed DC motor.
3. Potentiometer
4. STM32 NUCLEO – F411RE
5. DC power supply

System Overview



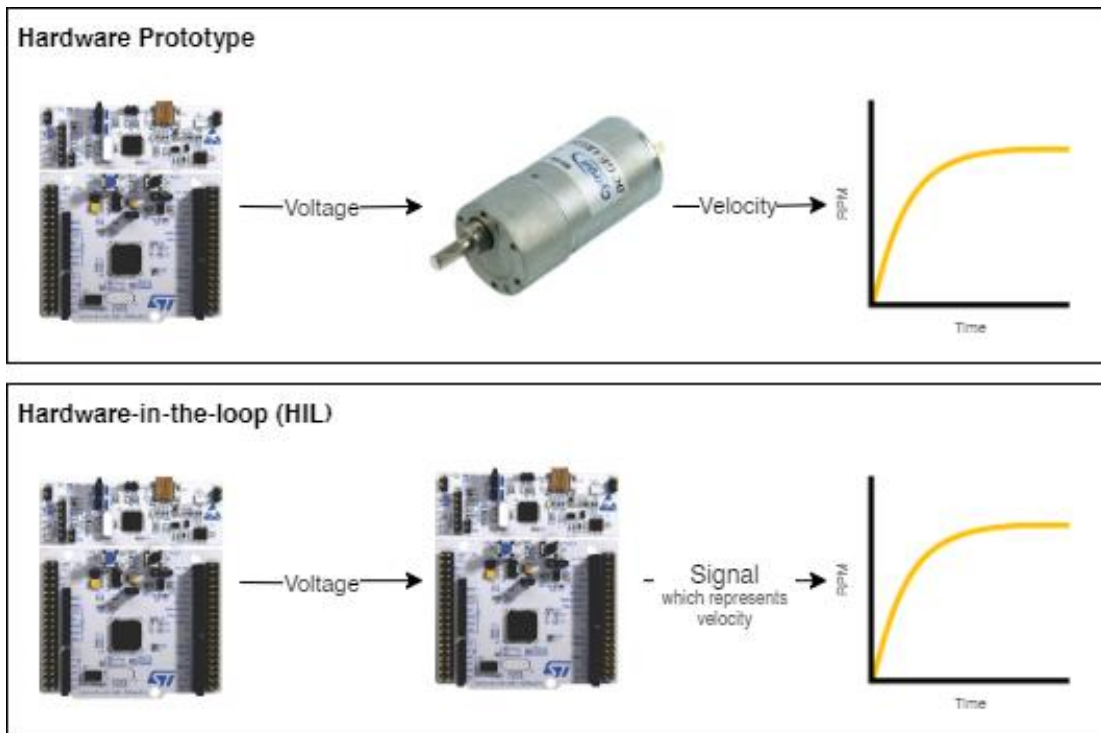
ภาพที่ 1 System Overview ของ Lab 2 HIL

ในส่วนนี้นักศึกษาจะต้องเขียน Hardware-in-the-loop (HIL) Firmware เพื่อจำลองพฤติกรรมของมอเตอร์ใส่ลงไปในบอร์ด STM32G474re จากนั้นจึงทดลองจ่ายไฟที่แรงดันต่าง ๆ จาก Power Supply แล้วจึงเปรียบเทียบความเร็วที่ได้จากมอเตอร์กับแรงดันที่ออกมาจาก HIL

Hardware-in-the-loop (HIL)

การออกแบบระบบควบคุมจำเป็นต้องพิจารณาปัจจัยต่าง ๆ อย่างละเอียดถี่ถ้วน ทั้งพฤติกรรมของระบบ ชีตจำกัดความสามารถของอุปกรณ์ต่าง ๆ และพฤติกรรมของระบบควบคุมที่เลือกใช้ การสร้าง Prototype โดยที่ยังไม่ทราบถึงพฤติกรรมที่แน่นอนของส่วนต่าง ๆ เหล่านี้จึงอาจทำให้อุปกรณ์เสียหายและทำให้เกิดความล่าช้าในการพัฒนาชิ้นงานได้ ดังนั้นในการออกแบบระบบควบคุมจึงนิยมออกแบบในระบบจำลอง (Simulation) ก่อน

นอกจากเราจะสามารถจำลองระบบทั้งหมดบนคอมพิวเตอร์ได้แล้ว (อย่างเช่นการจำลองผ่าน Simulink) เรายังสามารถจำลองระบบให้สมจริงขึ้นได้ผ่านการจำลองบน Microcontroller ซึ่งในการทดลองนี้จะให้นักศึกษาถอดพฤติกรรมของมอเตอร์ไปใส่ใน Microcontroller แทน ซึ่งจะทำให้นักศึกษาไม่จำเป็นต้องนำระบบควบคุมไปควบคุมระบบกับมอเตอร์จริง ๆ แต่นำมาควบคุม Microcontroller อีกตัวที่มีพฤติกรรมเหมือนกับมอเตอร์นั่นเอง



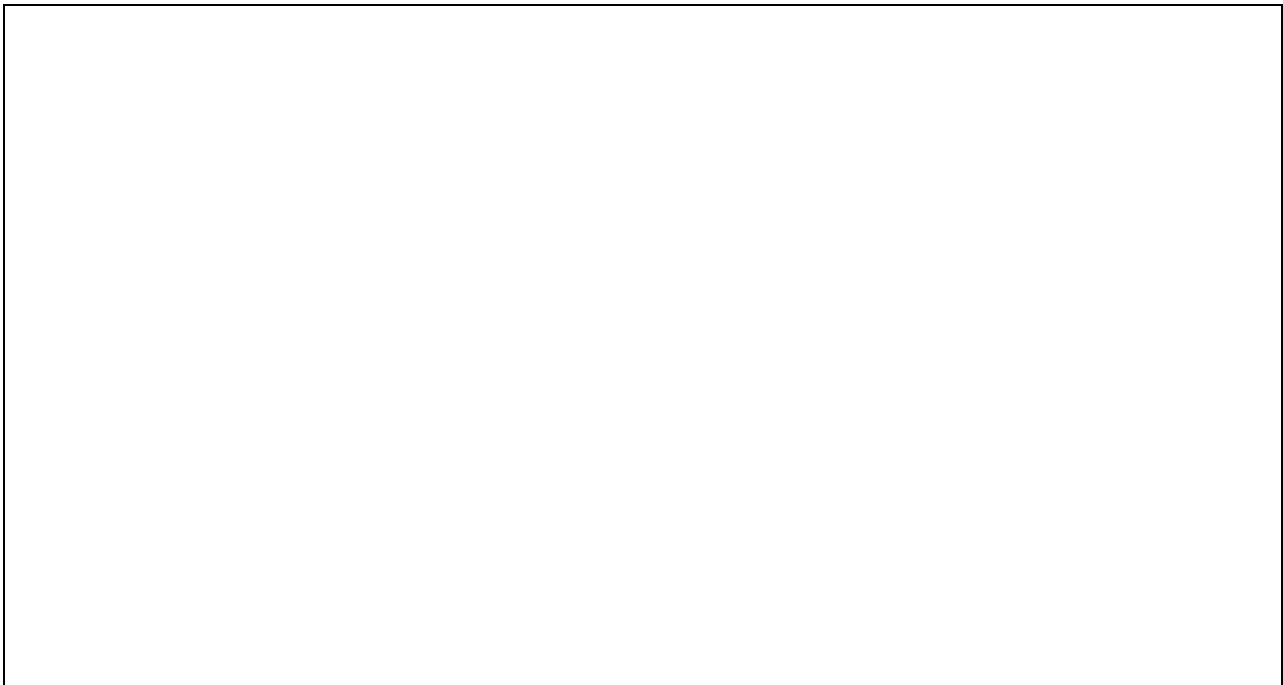
ภาพที่ 2 ระบบควบคุมแบบ Open Loop ใน Hardware-in-the-loop (HIL) เทียบกับการรันกับมอเตอร์จริง ๆ แบบทั่วไป

จาก Lab 1 : System Identification เราทราบว่า Transfer Function ของมอเตอร์ในการควบคุมความเร็วจากการจ่ายแรงดันไฟฟ้าเป็นระบบ Second Order ดังสมการ

$$G_{\theta}(s) = \frac{\Omega(s)}{V_{in}(s)} = \frac{\frac{K_t}{JL}}{s^2 + \left(\frac{R}{L} + \frac{B}{J}\right)s + \left(\frac{RB + K_t K_e}{JL}\right)}$$

เมื่อ	R	คือ ค่าความต้านทานภายใน (Resistance)
	L	คือ ค่าสภาพเหนี่ยวนำ (Inductance)
	B	คือ ค่าความหน่วง (Damp coefficient)
	J	คือ ค่าโมเมนต์ความเฉื่อย (Moment of Inertia)
	K_t	คือ สัมประสิทธิ์แรงบิด (Torque constant)
	K_e	คือ สัมประสิทธิ์แรงดันไฟฟ้าย้อนกลับ (Back-EMF constant)

เนื่องจากการประมวลผลใน Microcontroller เป็นแบบ Discrete การที่จะนำ Transfer Function ของมอเตอร์ไปใส่ใน Microcontroller จึงจำเป็นต้องแปลงสมการดังกล่าวให้เป็น Difference equation ซึ่งสามารถทำได้โดยการประมาณแบบกลับหลัง (Backward Estimation) ได้ดังนี้



เมื่อนำสมการดังกล่าวไปใส่ใน Firmware ของ Microcontroller โดยรับ Input มาจาก ADC จะได้ว่า Microcontroller ดังกล่าวสามารถแสดงพฤติกรรมแบบเดียวกับมอเตอร์ได้เมื่อจ่ายไฟเข้าไปให้ ADC

วิธีการทดลองและคำชี้แจง

1. ให้นักศึกษาพิจารณาหา Difference Equation ของระบบเพื่อนำไปเขียนใน Firmware ลงบนบอร์ด STM32G474re สำหรับการทำให้ HIL โดยที่ Firmware ดังกล่าวจะต้องรับแรงดันช่วง 0 – 12 V (ใช้ Voltage Divider แบ่งให้แรงดันที่เข้า ADC อยู่ในช่วง 0 – 3.3 V) และจะต้อง Output ออกมาให้อยู่ในช่วง 0 – 3.3 V ผ่าน DAC โดยเทียบ 1 V เท่ากับ 100 rad/s และ Offset ให้ 0 rad/s อยู่ที่ 1.65 V เพื่อให้เห็นทั้งความเร็วที่เป็นบวกและลบ
2. เปรียบเทียบลักษณะของกราฟของ Output ที่วัดได้จาก Oscilloscope ที่ได้มากับความเร็วจริง ๆ ของมอเตอร์ที่แรงดัน 5 V, 8 V และ 12 V อภิปรายผลลัพธ์ที่ได้และสรุปผลการทดลอง

Part II: PID Controller

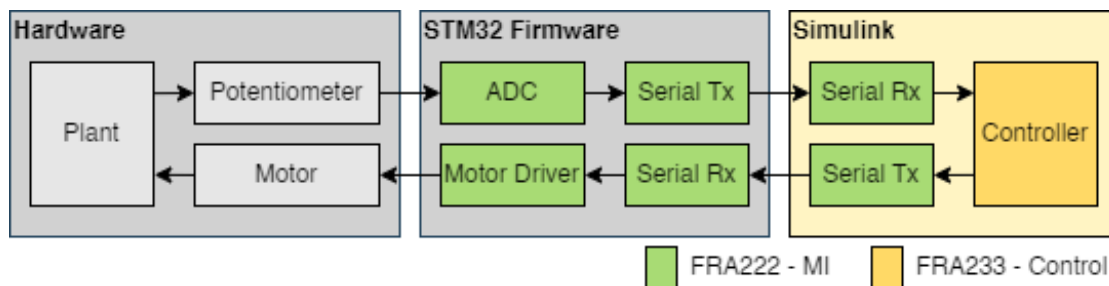
วัตถุประสงค์

เพื่อให้นักศึกษาเข้าใจหลักการทางคณิตศาสตร์ของ PID Controller รวมถึงเข้าใจพฤติกรรมที่เปลี่ยนไปของระบบเมื่อ Proportional Gain (K_p), Integral Gain (K_i) และ Derivative Gain (K_d) เปลี่ยนไป

อุปกรณ์

1. RMX Board
2. Brushed DC motor.
3. Potentiometer
4. STM32 NUCLEO – F411RE
5. DC power supply

System Overview



ในส่วนนี้ นักศึกษาจะต้องนำ Firmware ที่ใช้ในการอ่านตำแหน่งเชิงมุมของมอเตอร์จาก Potentiometer และการสั่งงานมอเตอร์ที่เขียนขึ้นในรายวิชา FRA222 Microcontroller Interface มาใช้เป็นสื่อกลางระหว่าง Controller และ Hardware โดย Simulink และ STM32 Firmware จะคุยกันผ่าน UART

PID Controller

PID Controller คือระบบควบคุมที่นำความคลาดเคลื่อน (Error) ซึ่งคำนวณได้จากการหาผลต่างระหว่างปริมาณที่ต้องการควบคุมกับปริมาณชนิดเดียวกันที่สามารถวัดได้จากระบบจริงมาคำนวณพจน์ต่าง ๆ ได้แก่ พจน์สัดส่วน (Proportional Term) พจน์ปริพันธ์ (Integral Term) และพจน์อนุพันธ์ (Derivative Term) แล้วจึงนำผลรวมของทั้ง 3 พจน์มารวมกันเพื่อใช้เป็น Input สำหรับระบบ (Plant) ดังสมการ

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de}{dt}$$

เมื่อ $u(t)$ คือ สัญญาณ Input ให้ระบบ
 $e(t)$ คือ ความคลาดเคลื่อนของปริมาณที่ต้องการควบคุม
 K_x คือ ค่า Gain ของพจน์ x เมื่อ $x \in \{p, i, d\}$

โดยที่หน้าที่ของพจน์ต่าง ๆ มีดังนี้

- **Proportional Term:** ขนาดของพจน์ถูกกำหนดด้วย K_p เป็นพจน์ที่แปรผันตรงกับ Error และเป็นพจน์ที่บอกว่าระบบจะเข้าสู่ Setpoint เร็วหรือช้า แต่หากเร็วเกินไป ระบบก็อาจจะไม่เสถียรได้
- **Integral Term:** ขนาดของพจน์ถูกกำหนดด้วย K_i เป็นพจน์ที่จะเก็บผลรวมของ Error ทุกช่วงเวลาเอาไว้ มีหน้าที่ทำให้ระบบเข้าสู่ Setpoint (ทำให้ Steady State Error น้อยที่สุด) แต่การกำหนดให้พจน์นี้มีขนาดใหญ่อาจทำให้ระบบ Overshoot มากขึ้น
- **Derivative Term:** ขนาดของพจน์ถูกกำหนดด้วย K_d เป็นพจน์ที่คำนวณจากความชันของ Error ในช่วงเวลานั้น ๆ มีหน้าที่ในการชะลอการเปลี่ยนแปลงของ Output ทำให้ Overshoot น้อยลง แต่เนื่องจากถูกคำนวณมาจากความชันของสัญญาณซึ่งรวมไปถึงสัญญาณรบกวนด้วย Derivative Term จึงเป็นพจน์ที่ไวต่อสัญญาณรบกวนมาก ๆ

ดังนั้น จากที่สมการกล่าวมาข้างต้นจึงสามารถหา Transfer Function ของ PID ได้ดังนี้



จาก Lab 1 : System Identification เราทราบว่า Transfer Function ของมอเตอร์ในการควบคุมตำแหน่งจากการจ่ายแรงดันไฟฟ้าเป็นระบบ Third Order ดังสมการ

$$G_{\theta}(s) = \frac{\theta(s)}{V_{in}(s)} = \frac{\frac{K_t}{JL}}{s^3 + \left(\frac{R}{L} + \frac{B}{J}\right)s^2 + \left(\frac{RB + K_t K_e}{JL}\right)s}$$

เมื่อ	R	คือ ค่าความต้านทานภายใน (Resistance)
	L	คือ ค่าสภาพเหนี่ยวนำ (Inductance)
	B	คือ ค่าความหน่วง (Damp coefficient)
	J	คือ ค่าโมเมนต์ความเฉื่อย (Moment of Inertia)
	K_t	คือ สัมประสิทธิ์แรงบิด (Torque constant)
	K_e	คือ สัมประสิทธิ์แรงดันไฟฟ้าย้อนกลับ (Back-EMF constant)

ซึ่งสามารถเขียน Block Diagram แสดงความสัมพันธ์ของ Transfer Function ของระบบทั้งสองและปริมาณที่เกี่ยวข้องได้ ดังนี้

ดังนั้นจาก Block Diagram จึงสามารถรวมออกมาเป็น Transfer Function ของระบบทั้งหมดได้ว่า

วิธีการทดลอง

1. ให้นักศึกษาเขียน Firmware ในการอ่านค่าตำแหน่งจาก Potentiometer และการสั่งงาน Motor โดยเขียนให้ Firmware ดังกล่าวสื่อสารกับ Simulink ด้วย UART (เลือก Configuration และวิธีการสื่อสารตามความเหมาะสม หรือตามที่เรียนมาในวิชา Microcontroller Interface)
2. จากตำแหน่งที่ได้รับมาจาก Microcontroller ให้นักศึกษาเขียน PID Controller ลงบน Simulink เพื่อควบคุมตำแหน่งของมอเตอร์ ลองปรับ K_p , K_i และ K_d เพื่อให้เห็นพฤติกรรมที่เปลี่ยนไปของมอเตอร์ (ค่า K_p , K_i และ K_d ที่ทำให้เห็นความแตกต่างจะแตกต่างกันตามคุณลักษณะของมอเตอร์แต่ละตัว)
3. กำหนดให้ Input Voltage ที่จ่ายให้มอเตอร์อยู่ในช่วง -12 V ถึง 12 V จงเลือก K_p , K_i และ K_d เพื่อควบคุมตำแหน่งของมอเตอร์จากตำแหน่งในช่วง 0 ถึง 2π ให้ไปอยู่ที่ตำแหน่ง π ภายใต้ข้อจำกัดทางกายภาพดังต่อไปนี้
 - a. Steady state error น้อยกว่า 1 องศา
 - b. Settling Time ของระบบอยู่ภายในเวลา 5 วินาที

คำชี้แจง

1. จากการทดลองให้เขียนสรุปผลพฤติกรรมที่เปลี่ยนแปลงไปจากการเปลี่ยนค่า K_p , K_i และ K_d ที่เกิดขึ้นบนระบบจริง เปรียบเทียบกับสิ่งที่คาดว่าจะเกิดขึ้นผ่านการคำนวณเพื่อเป็นหลักฐานมาสนับสนุนข้อมูลที่ให้มาในเอกสาร รวมทั้งสรุปข้อดี – ข้อเสียและหน้าที่ของแต่ละพจน์อีกครั้งจากผลการทดลองจริง
2. จงอธิบายวิธีการจูน K_p , K_i และ K_d เพื่อให้มอเตอร์อยู่ภายใต้ข้อจำกัด (ใส่มาในขั้นตอนการทดลอง)
3. จงแสดงการคำนวณให้เห็นว่าค่า K_p , K_i และ K_d ที่เลือกมาสามารถทำให้มอเตอร์อยู่ภายใต้ข้อจำกัดที่กำหนดไว้โดยอาศัย Transfer Function จาก Parameters ของมอเตอร์ที่ทำได้จาก Lab 1 (สามารถประมาณ Parameters บางตัวให้มีค่าเป็น 0 ได้ พร้อมเขียนเหตุผลประกอบ)

รายละเอียดการส่งงาน

ให้นักศึกษาส่งรายงาน (Lab report) ตั้งชื่อไฟล์ตามแบบฟอร์มด้านล่างเพียงไฟล์เดียว โดยให้ตัวแทนกลุ่มส่งเพียงไฟล์เดียวเท่านั้น (หากชื่อไฟล์ผิดหรือมีการส่งซ้ำกันจะหักจุดละ 1 คะแนน)

ชื่อไฟล์: FRA233_LAB#_xx_xx_xx_v#.pdf

โดยรายงานการทดลองประกอบด้วย

1. ชื่อการทดลอง
2. วัตถุประสงค์
3. สมมติฐาน
4. ขั้นตอนการทดลอง
5. วิเคราะห์ผลการทดลอง
6. อภิปรายและสรุปผล