มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

สถาบันวิทยาการหุ่นยนต์ภาคสนาม

รายงานโครงงานวิศวกรรม

Circular Pick and Place Robot

วิชา FRA263/264: Robotics Studio III

จัดทำโดย

นายทัดชัย สุขสราญ	67340500016
นายธนดล พูลสระคู	67340500017
นายธรรวิทวัส เพ็งนาม	67340500018
นายวราวิชญ์ ศิลายงค์	67340500036
นายภคิน ศักดิ์สิทธิ์พิทักษ์	67340500060
นางสาวศิรินภา สุขบุญส่ง	67340500078

ปีการศึกษา 2568

บทคัดย่อ

รายงานฉบับนี้นำเสนอการออกแบบและพัฒนาหุ่นยนต์หยิบและวางชิ้นงานแบบวงกลม (Circular Pick and Place Robot) สำหรับใช้งานในกระบวนการผลิตเชิงอุตสาหกรรม โดยมีหน้าที่ลำเลียงชิ้นงาน Connection Rod ระหว่างสถานีทำงานสี่สถานีในลักษณะการหมุนแบบหนึ่งองศาอิสระ

ระบบประกอบด้วยแขนกลที่ขับเคลื่อนด้วยมอเตอร์กระแสตรง RS-775 กำลัง 24 V ผ่านชุดเฟืองดาวเคราะห์และสายพานฟันที่มีอัตราทดรวม $N = 70$ และประสิทธิภาพ $\eta = 0.836$ ควบคุมด้วยไมโครคอนโทรลเลอร์ STM32G474RE การออกแบบระบบควบคุมใช้โครงสร้าง Cascade PID สองชั้น ได้แก่ วงรอบความเร็วที่ความถี่ 1 kHz และวงรอบตำแหน่งที่ความถี่ 100 Hz โดยอ้างอิงพารามิเตอร์ของระบบจากการระบุระบบแบบ Black-Box ด้วยวิธี Least-Squares ร่วมกับการกรองสัญญาณ Butterworth อันดับสี่ที่ความถี่ตัด 10 Hz ผลการระบุระบบให้ค่า $R_a = 1.4534\,\Omega$, $K_t = 0.040\,65\,\mathrm{N\,m\,A^{-1}}$, $J = 0.727\,62\,\mathrm{kg\,m^2}$ และ $B = 0.192\,79\,\mathrm{N\,m\,s\,rad^{-1}}$

เพื่อให้บรรลุข้อกำหนดด้าน Cycle Time ไม่เกิน 35 วินาทีสำหรับสี่ชิ้น ระบบติดตั้ง ZV Input Shaper ที่ออกแบบจากความถี่ธรรมชาติของแท่ง Connection Rod ($\omega_{n,\mathrm{rod}} = 12.386\,\mathrm{rad\,s^{-1}}$, $\zeta_{\mathrm{rod}} = 0.041$) ซึ่งช่วยยกเลิกการแกว่งของปลายแขนโดยแทบไม่เพิ่มเวลาหน่วง นอกจากนี้ยังใช้ Kalman Filter สี่สถานะ $\mathbf{x} = [\theta_{\mathrm{arm}},\ \omega_{\mathrm{arm}},\ i_a,\ \tau_d]^\top$ สำหรับประมาณค่าแรงบิดรบกวนแบบ Real-time การสื่อสารกับระบบควบคุมส่วนกลางใช้โปรโตคอล Modbus RTU ผ่านบัส RS-485 ส่วนกริปเปอร์ควบคุมด้วยสัญญาณ Digital I/O โดยตรง

ผลการจำลองและการทดสอบระบบย่อยแสดงให้เห็นว่าระบบมีศักยภาพในการบรรลุข้อกำหนดสมรรถนะหลัก ได้แก่ Settling Time ไม่เกิน 0.5 s (P5), Overshoot ไม่เกิน 1% (P6), ความแม่นยำตำแหน่ง ±0.100° (A1), ความซ้ำซากได้ ±0.143° ใน 100 รอบ (A2) และ Cycle time รวม 32.2 วินาที (P1) ทั้งนี้การทวนสอบข้อกำหนดเชิงปริมาณครบถ้วนอยู่ระหว่างดำเนินการเก็บข้อมูลบนฮาร์ดแวร์จริง

คำสำคัญ: หุ่นยนต์หยิบและวางชิ้นงาน, การควบคุมแบบ Cascade PID, ZV Input Shaping, Kalman Filter, การระบุระบบ, Modbus RTU

Abstract

This report presents the design and development of a circular pick-and-place robot for industrial manufacturing applications, tasked with transferring Connection Rod workpieces among four work stations via a single rotational degree of freedom.

The mechanical system employs an RS-775 brushed DC motor at 24 V, coupled through a planetary gearbox and toothed belt drive with a combined gear ratio of $N = 70$ and efficiency $\eta = 0.836$, controlled by an STM32G474RE microcontroller. The control architecture adopts a two-loop Cascade PID structure comprising a velocity inner loop at 1 kHz and a position outer loop at 100 Hz. Plant parameters were obtained via Black-Box system identification using Least-Squares estimation with fourth-order Butterworth pre-filtering at 10 Hz, yielding $R_a = 1.4534\,\Omega$, $K_t = 0.040\,65\,\mathrm{N\,m\,A^{-1}}$, $J = 0.727\,62\,\mathrm{kg\,m^2}$, and $B = 0.192\,79\,\mathrm{N\,m\,s\,rad^{-1}}$.

To satisfy the cycle time requirement of 35 s for four workpieces, a ZV Input Shaper was designed based on the measured natural frequency of the Connection Rod ($\omega_{n,\mathrm{rod}} = 12.386\,\mathrm{rad\,s^{-1}}$, $\zeta_{\mathrm{rod}} = 0.041$), effectively eliminating residual vibration with negligible additional delay. A four-state Kalman filter with state vector $\mathbf{x} = [\theta_{\mathrm{arm}},\ \omega_{\mathrm{arm}},\ i_a,\ \tau_d]^\top$ provides real-time disturbance torque estimation. System integration with a central supervisory controller is achieved via Modbus RTU over RS-485, while the gripper is actuated through digital I/O signals.

Simulation results and subsystem-level testing indicate that the system has the capability to meet its primary performance requirements, including settling time within 0.5 s (P5), overshoot not exceeding 1% (P6), position accuracy of ±0.100° (A1), repeatability of ±0.143° over 100 cycles (A2), and a total cycle time of 32.2 s for four workpieces (P1). Full quantitative verification against all requirements is pending hardware-level testing.

Keywords: Pick-and-place robot, Cascade PID control, ZV input shaping, Kalman filter, System identification, Modbus RTU

บทสรุปผู้บริหาร

ภาพรวมโครงงาน

โครงงานนี้มีวัตถุประสงค์เพื่อออกแบบและพัฒนาหุ่นยนต์หยิบและวางชิ้นงานแบบวงกลม สำหรับลำเลียงชิ้นงาน Connection Rod ระหว่างสี่สถานีทำงานในกระบวนการผลิตเชิงอุตสาหกรรม ดำเนินการโดยนักศึกษากลุ่มที่ 6 ภายใต้วิชา FRA263/264: Robotics Studio III สถาบันวิทยาการหุ่นยนต์ภาคสนาม มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ปีการศึกษา 2568

สิ่งที่พัฒนาและดำเนินการสำเร็จ

ทีมพัฒนาระบบได้ดำเนินการออกแบบและสร้างระบบครบทุกองค์ประกอบ ดังนี้

ด้านเชิงกล: ออกแบบและประกอบแขนกลแบบ 1-DOF ที่ขับเคลื่อนผ่านชุดเฟืองดาวเคราะห์และสายพานฟัน อัตราทดรวม $N = 70$ โดยผ่านการคำนวณตามมาตรฐาน ASME สำหรับเพลา และ ISO 281 สำหรับตลับลูกปืน พร้อมทั้งวิเคราะห์การโก่งตัวด้วย FEA

ด้านไฟฟ้าและอิเล็กทรอนิกส์: ออกแบบตู้ควบคุมครบวงจร ประกอบด้วย PCB ที่ออกแบบเอง, วงจรป้องกัน E-Stop, Optocoupler, Relay และระบบจ่ายไฟ 24 V/10 A, 5 V/5 A พร้อม Circuit Breaker และ Fuse

ด้าน การ ควบคุม: ระบุ พารามิเตอร์ ของ มอเตอร์ ด้วย วิธี System Identification จาก ข้อมูล Response จริง ออกแบบตัวควบคุม Cascade PID ด้วย Root Locus บน MATLAB ติดตั้ง ZV Input Shaper สำหรับลด Residual Vibration ของชิ้นงาน และออกแบบ Kalman Filter สี่สถานะสำหรับประมาณค่า Disturbance Torque แบบ Real-time

ด้านซอฟต์แวร์และการสื่อสาร: พัฒนา Firmware บน STM32G474RE ครบทุก module ได้แก่ Cascade PID, ZV Shaper, Kalman Filter, Finite State Machine, Modbus RTU (LPUART1, Slave Address 21) และ Joystick Interface ผ่าน ESP32

ข้อจำกัดและปัญหาที่พบ

1. ปัญหาการเชื่อมต่อกับ Base System (Demo 1)

Abnormal Heartbeat ใน Demo 1: Web Dashboard ของระบบส่ง Heartbeat ไปยัง Robot ไม่สม่ำเสมอ ส่งผลให้ Robot ตรวจจับ Heartbeat Timeout และตอบกลับ Abnormal Status ทำให้ Base System แสดงผล Abnormal ตามไปด้วย สังเกตได้ว่าเมื่อหุ่นยนต์เคลื่อนที่จนค่า Encoder เปลี่ยนแปลง ปัญหาจะหายชั่วคราวแล้วกลับมาอีก บ่งชี้ว่าเกิดจาก Timing Mismatch ระหว่างรอบการ Poll ของ Dashboard กับรอบ

Heartbeat Watchdog ของ Firmware แก้ไขโดยเปลี่ยน Backend เป็น Python asyncio WebSocket ซึ่งจัดการ Event Loop ได้แม่นยำกว่า บทเรียน: ควรทดสอบ End-to-End Integration กับ Base System จริงก่อน Demo อย่างน้อย 1 สัปดาห์

2. ความเสียหายของ Gripper ระหว่างการสาธิต (Demo 2)

ในการสาธิต Demo 2 เกิดปัญหา Abnormal ซ้ำหลังจาก Base System รัน Process เพื่อสำรวจตำแหน่งวาง Connection Rod เสร็จสิ้น ทำให้ต้องทำการ Restart Website ส่งผลให้ค่าตำแหน่ง Pick & Place ที่ระบบสุ่มมาก่อนหน้าถูก Reset ทั้งหมด เมื่อพยายามกรอกค่าตำแหน่งใหม่ด้วยตนเอง พบว่า Base System ไม่ได้แปลงทิศทาง CW/CCW ให้ถูกต้องตามค่าที่กรอก ส่งผลให้แขนกลเคลื่อนที่ผิดทิศทางและชนกับ Connection Rod ทำให้ Gripper ได้รับความเสียหายและไม่สามารถซ่อมแซมได้ทัน จึงไม่สามารถทดสอบ Full Cycle ในการสาธิตรอบสุดท้ายได้ อย่างไรก็ตาม การทำงานของระบบควบคุมและ Firmware ทุก module ได้รับการยืนยันก่อนเกิดเหตุการณ์ดังกล่าว

3. การไม่ได้ติดตั้ง CAN Bus สำหรับ Gripper

ข้อกำหนดเดิมระบุการสื่อสารกับ Gripper ผ่าน CAN Bus แต่เนื่องจากข้อจำกัดด้านเวลาในการพัฒนา ทีมจึงตัดสินใจเปลี่ยนมาใช้สัญญาณ Digital I/O โดยตรง ซึ่งสามารถรองรับการทำงานตามข้อกำหนดสมรรถนะหลักได้ครบถ้วน

สรุปผลโดยรวม

ระบบหุ่นยนต์ที่พัฒนาขึ้นครอบคลุมทุกองค์ประกอบของ Cyber-Physical System ตั้งแต่การออกแบบเชิงกล ไฟฟ้า ซอฟต์แวร์ และการควบคุม โดยมีการใช้เทคนิคการควบคุมขั้นสูงที่เหมาะสมกับโจทย์อุตสาหกรรม ความท้าทายหลักที่พบในโครงงานนี้อยู่ที่การ Integration ระหว่างระบบย่อยหลายระบบ และการบริหารจัดการ Real-time Constraints บน Embedded System ซึ่งทีมได้แก้ไขและเรียนรู้จากปัญหาดังกล่าวตลอดกระบวนการพัฒนา

สารบัญ

บทคัดย่อ i

Abstract ii

บทสรุปผู้บริหาร iii

1 บทนำ 1

1.1 ที่มาและความสำคัญของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตและข้อจำกัดของโครงงาน 1

1.3.1 ขอบเขตของโครงงาน 1

1.3.2 ข้อจำกัดของโครงงาน 2

1.4 ข้อกำหนดความต้องการของระบบ 2

1.4.1 การปรับปรุงข้อกำหนด (Requirement Amendment) 4

1.5 ภาพรวมระบบ 4

1.6 งบประมาณโครงงาน 5

1.7 โครงสร้างของรายงาน 7

2 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง 9

2.1 พลศาสตร์ของมอเตอร์กระแสตรง 9

2.1.1 สมการโดเมนไฟฟ้า 9

2.1.2 สมการโดเมนเชิงกล 10

2.1.3 State-Space Representation 10

2.2 ทฤษฎีการควบคุมและการกรองสัญญาณ 11

2.2.1 Cascade PID Control 11

2.2.2 ZVD Input Shaping 12

2.2.3 Kalman Filter 14
2.3 มาตรฐานการออกแบบเชิงกล 18
2.3.1 การออกแบบเพลา (ASME Shaft Design) 18
2.3.2 การคำนวณอายุการใช้งานตลับลูกปืน (ISO 281) 19
2.3.3 พลศาสตร์ของชิ้นงาน Connection Rod (Simple Pendulum Model) 19
2.4 อัลกอริทึมสร้างโปรไฟล์การเคลื่อนที่ 21
2.4.1 S-Curve Trajectory (Jerk-Limited) 21
2.5 การระบุเอกลักษณ์ของระบบ (System Identification) 23
2.5.1 การประมาณค่าพารามิเตอร์ทางไฟฟ้า (Stalled Rotor Test) 23
2.5.2 การระบุเอกลักษณ์แบบกล่องเทาด้วย MATLAB (Grey-Box Identification) 24
2.5.3 Zero-Phase Filtering ด้วย `filtfilt` 25
2.6 ปรากฏการณ์อินทิกรัลไวนด์อัป (Integral Windup) 28
2.7 มาตรฐานตู้ควบคุมอุตสาหกรรม (Industrial Enclosures) 28
2.7.1 การจัดการสายไฟและลดสัญญาณรบกวน (Wiring & EMI Mitigation) . 28
2.7.2 ความปลอดภัยทางไฟฟ้าและวงจรหยุดฉุกเฉิน (Electrical Safety & E-Stop) 29
2.8 การออกแบบระบบไฟฟ้าและวงจรป้องกัน (Electrical System and Protection Design) 29
2.8.1 วงจรป้องกันความผิดปกติทางไฟฟ้า (Electrical Protection Circuits) . 29
2.8.2 การแยกกัลวานิกทางแสง (Galvanic Isolation) 30
2.8.3 การจัดการสายไฟและหน้าสัมผัส (Wiring and Strain Relief) 30
2.9 โพรโทคอลการสื่อสารข้อมูลอุตสาหกรรม 31
2.9.1 Controller Area Network (CAN Bus) 31
2.9.2 ชั้นกายภาพ RS-485 และโพรโทคอล Modbus RTU 32
2.10 เซนเซอร์และอุปกรณ์ตรวจวัด 33
2.10.1 Incremental Encoder (AMT-103) 33
2.10.2 Current Sensor (WCS1800) 34
2.10.3 Proximity Sensor (Inductive) 35

3 สถาปัตยกรรมและการออกแบบระบบ 37

3.1 สถาปัตยกรรมระบบภาพรวม 37

3.1.1 ห่วงโซ่การส่งกำลัง (Power Transmission Chain) 37

3.1.2 Application Finite State Machine 38

3.2 การออกแบบเชิงกล 39

3.2.1 สถาปัตยกรรมเชิงกลโดยรวม 39

3.2.2 การคำนวณเพลา 40

3.2.3 การวิเคราะห์เพลาด้วย FEA (Shaft FEA) 43

3.2.4 การคำนวณสายพาน 48

3.2.5 การคำนวณแรงบิดที่ต้องการและการเลือกมอเตอร์ 50

3.2.6 การเลือกตลับลูกปืนและการคำนวณอายุการใช้งาน 51

3.2.7 การวิเคราะห์การโก่งตัวด้วย FEA 52

3.3 การออกแบบระบบไฟฟ้า 55

3.3.1 สถาปัตยกรรมไฟฟ้าโดยรวม 55

3.3.2 การคำนวณกำลังไฟฟ้าและการเลือก Circuit Breaker 56

3.3.3 การออกแบบ PCB และวงจรป้องกัน 58

3.3.4 STM32 Pin Assignment 58

3.3.5 การระบุระบบ (System Identification) 61

3.3.6 การออกแบบตัวควบคุม Cascade PID 69

3.3.7 ค่า Gain ที่ใช้งานจริง (Hardware-Tuned) 76

3.3.8 การออกแบบ ZVD Input Shaper 84

3.3.9 การออกแบบ S-Curve Motion Profile 88

3.3.10 Modbus RTU Protocol และ Register Map 95

4 การสร้างและบูรณาการระบบ 101

4.1 การประกอบโครงสร้างทางกล 101

4.1.1 ภาพรวมการประกอบ 101

4.1.2 การประกอบ Sub-assembly 102

4.1.3 การประกอบรวมขั้นสุดท้าย (Final Assembly Process) 109

4.1.4 ขั้นตอนการติดตั้ง ณ จุดใช้งาน (Field Installation Procedure) 111
4.1.5 ปัญหาที่พบระหว่างการประกอบและการแก้ไข 113
4.2 การประกอบตู้ควบคุมและการเดินสายไฟ 114
4.2.1 รายการอุปกรณ์หลัก (Bill of Materials) 114
4.2.2 สถาปัตยกรรม การ จ่าย ไฟ และ การ แยก สัญญาณ (Power & Signal Architecture) 114
4.2.3 แนวทางปฏิบัติในการเดินสายไฟ (Cable Management & Wiring) . . . 115
4.3 การพัฒนา Firmware 117
4.3.1 ภาพรวมสถาปัตยกรรมซอฟต์แวร์ (Software Architecture) 117
4.3.2 การจัดสรรเวลาทำงาน (Execution Model และ Task Timing) 118
4.3.3 สถานะการทำงานของระบบ (Application FSM) 119
4.3.4 กระบวนการหาจุดอ้างอิงระดับความแม่นยำสูง (Two-Stage Homing Sequence) 119
4.3.5 อัลกอริทึมควบคุมการเคลื่อนที่ (Motion Control Algorithms) 120
4.3.6 ระบบความปลอดภัยทางซอฟต์แวร์ (Safety Stack) 121
4.3.7 การสื่อสารผ่านโครงข่ายและ CAN Bus Failsafe 122
4.4 การบูรณาการระดับระบบ (System Integration) 122
4.4.1 ภาพรวมสถาปัตยกรรมระบบ (System Architecture) 122
4.4.2 การจัดการข้อมูลร่วม (Shared Data Architecture) 123
4.4.3 การเชื่อมต่ออุปกรณ์ขับเคลื่อนและเซนเซอร์ (Motor, Encoder & Driver) 123
4.4.4 การเชื่อมต่อระบบควบคุมไร้สาย (Joystick Integration) 124
4.4.5 การสื่อสารด้วยโปรโตคอล Modbus RTU 124

5 ผลการทดสอบและการทวนสอบ 126
5.1 ผลการทดสอบระบบย่อย 126
5.1.1 การวัดระยะคลอน (Backlash Measurement) 126
5.1.2 การทดสอบ Encoder และ Sensor 127
5.1.3 การทดสอบ Motor Driver (Cytron MD10C) 129
5.1.4 การทดสอบ E-Stop Response Time 130

5.1.5 การทดสอบการสื่อสาร Modbus RTU 130
5.2 ผลการระบุระบบและการจำลอง 131
5.2.1 ผลการระบุพารามิเตอร์ของมอเตอร์ 131
5.2.2 ผลการตรวจสอบความถูกต้องของแบบจำลอง (Model Validation) . . . 134
5.2.3 ผลการ Simulation Cascade PID และ ZVD Input Shaper 135
5.2.4 ผลการออกแบบ Kalman Filter 136
5.3 ผลการทดสอบสมรรถนะการทำงานจริง 137
5.3.1 Step Response: Overshoot และ Settling Time 137
5.3.2 ความแม่นยำตำแหน่ง (Position Accuracy) 139
5.3.3 ความซ้ำซากได้ (Repeatability) 140
5.3.4 Cycle Time 142
5.4 การทวนสอบตามข้อกำหนด (Requirements Traceability & V&V) 145

6 สรุปผลและข้อเสนอแนะ 148
6.1 สรุปผลการดำเนินงาน 148
6.2 ปัญหา อุปสรรค และบทเรียนที่ได้ 148
6.2.1 ปัญหาเชิงกล 148
6.2.2 ปัญหาด้านซอฟต์แวร์และการบูรณาการ 149
6.2.3 ปัญหาด้านการออกแบบระบบควบคุม 149
6.3 ข้อเสนอแนะสำหรับการพัฒนาต่อยอด 149
6.3.1 การพัฒนาระบบควบคุม 149
6.3.2 การพัฒนาระบบกล 150
6.3.3 การพัฒนาระบบไฟฟ้าและซอฟต์แวร์ 150

A Engineering Drawings 152

B Source Code Repository 160

สารบัญรูป

1.1 System Architecture ของระบบ Circular Pick and Place Robot แสดง Physical Architecture และ Cyber Architecture 4

2.1 แผนภาพ Electro-mechanical ของมอเตอร์กระแสตรง แสดงความสัมพันธ์ระหว่างโดเมนไฟฟ้าและโดเมนเชิงกล 9

2.2 Block Diagram ของระบบควบคุม Cascade PID แสดงวงรอบความเร็วและวงรอบตำแหน่ง 11

2.3 ลักษณะของสัญญาณ ZVD Impulse Sequence (บน) และเปรียบเทียบการตอบสนองของระบบเพื่อลดการแกว่ง (ล่าง) 13

2.4 ขั้นตอนการทำงานของ Discrete-Time Kalman Filter ในแต่ละ Time Step 15

2.5 การเปรียบเทียบประสิทธิภาพการประมาณค่าความเร็วเชิงมุมระหว่าง Kalman Filter กับการหาอนุพันธ์โดยประมาณ (Approximate Derivative) 17

2.6 Free Body Diagram ของ Connection Rod จำลองเป็น Simple Pendulum ที่ถูกกระตุ้นด้วยความเร่งเชิงมุมของแขนกล 20

2.7 โปรไฟล์การเคลื่อนที่แบบ S-Curve (Jerk-Limited) แสดงความสัมพันธ์เชิงจลนศาสตร์ทั้ง 7 เฟส 22

2.8 เปรียบเทียบ Causal Filter (สีแดง) และ `filtfilt` Zero-Phase Filter (สีน้ำเงิน) บนสัญญาณเดียวกัน (บน) Causal Filter มี Phase Lag ที่สังเกตได้ (กลาง) `filtfilt` ทับ Peak ของสัญญาณต้นฉบับได้ตรงทุกจุด (ล่าง) Phase Difference ระหว่างทั้งสองวิธี 27

2.9 การเปรียบเทียบการตอบสนองของระบบ (บน) สัญญาณควบคุม (กลาง) และการสะสมสถานะของตัวอินทิเกรต (ล่าง) ระหว่างตัวควบคุม PID มาตรฐานและตัวควบคุมที่มีการติดตั้งระบบ Anti-Windup แบบ Back-Calculation 28

2.10 หลักการแยกกัลวานิกทางแสง (Galvanic Isolation) เพื่อป้องกันสัญญาณรบกวนจากภาคขับกำลังไหลกลับเข้าสู่ภาคตรรกะ 30

2.11 ลักษณะทางกายภาพของสัญญาณ CAN Bus แสดงผลต่างแรงดันในการกำหนดสถานะตรรกะแบบ Dominant และ Recessive 31

2.12 โครงสร้างเฟรมข้อมูลของโพรโทคอล Modbus RTU แสดงการจัดเรียงไบต์ข้อมูลและรหัสตรวจสอบความผิดพลาด (CRC-16) 33

2.13 Encoder AMT-103 และ Quadrature Signal แสดงการระบุทิศทางจากลำดับ Phase 33

2.14 WCS1800 Hall-Effect Current Sensor และลักษณะการตอบสนองเชิงเส้น 34

2.15 Proximity Sensor บนแขนกลและ Timing Diagram ของสัญญาณระหว่าง Two-Stage Homing 35

3.1 System Architecture ของระบบ Circular Pick and Place Robot แสดง Physical Architecture และ Cyber Architecture 37

3.2 แผนผังห่วงโซ่การส่งกำลังแสดงการเปลี่ยนแปลงความเร็วและแรงบิด ตั้งแต่มอเตอร์ถึงแขนกล พร้อมตำแหน่ง Encoder ที่ Output Shaft 38

3.3 Application FSM แสดง 5 สถานะหลักและเงื่อนไข Transition โดย FAULT สามารถถูก Trigger ได้จากทุกสถานะ (ลูกศรสีแดง) 39

3.4 CAD Assembly ของระบบเชิงกล แสดงส่วนประกอบหลักทั้งหมด 40

3.5 Free Body Diagram ของระบบเพลาแสดงแรงและโมเมนต์ที่จุดวิกฤต 41

3.6 การตั้งค่าโมเดล FEA เพลา แสดง Fixture ที่ตำแหน่งลูกปืน และภาระแรงดึงสายพาน 130 N พร้อม Moment 1.7 N·m 44

3.7 ผล FEA เพลา: การกระจายตัวของ Upper Bound Axial and Bending Stress (Max $= 2.221 \times 10^7$ N/m^2 บริเวณตำแหน่ง Belt Pulley) 44

3.8 ผล FEA เพลา: ระยะการเบี่ยงเบน (URES Displacement) (Max $= 4.721 \times 10^{-3}$ mm ที่ปลายเพลา) 45

3.9 ผล FEA เพลา: ค่าความปลอดภัย (Factor of Safety) (Min FOS $= 9.313$ บริเวณตำแหน่ง Belt Pulley) 45

3.10 ผล FEA เพลา: แผนภาพแรงเฉือน (Shear Force in Dir2) แสดงการเปลี่ยนแปลงแรงเฉือนตลอดความยาวเพลา 46

3.11 ผล FEA เพลา: แผนภาพโมเมนต์ดัด (Bending Moment about Dir2) แสดงการกระจาย Bending Moment ตลอดความยาวเพลา 46

3.12 Step Shaft Drawing แสดงขนาดเพลาแต่ละช่วงและตำแหน่งลูกปืน 47

3.13 ตารางค่า Admissible Peripheral Force ของสายพาน HTD5M จาก Habasit Engineering Guide แสดงพิกัด 400 N สำหรับความกว้าง 15 mm 49

3.14 โมเดลจำลอง FEA แสดงจุดจับยึด Fixture (สีฟ้า) และทิศทางของภาระแรง Remote Load (ลูกศรสีชมพู) 52

3.15 การตั้งค่า Remote Load 2.76 kg แสดง Remote Point และ Moment of Inertia ที่กำหนด 53

3.16 ผล FEA: การกระจายตัวของ Von Mises Stress บนแขนกล (Max $= 2.721 \times 10^7$ N/m^2 บริเวณขอบหน้าสัมผัสกับฐาน) 54

3.17 ผล FEA: ระยะการแอ่นตัว (Displacement) บนแขนกล (Max $= 0.2337$ mm ที่ปลายสุดของแขนกล) 54

3.18 ผล FEA: ค่าความปลอดภัย (Factor of Safety) บนแขนกล (Min FOS $= 8.6$) . . . 55

3.19 Electrical Block Diagram แสดงการแบ่ง Power Bus และการแยก Signal ระหว่าง High Power และ Logic Domain 56

3.20 การออกแบบ PCB สำหรับระบบควบคุม รวม STM32, ESP32, Optocoupler และ Relay ไว้บน Board เดียว 58

3.21 STM32G474RE Pin Configuration จาก STM32CubeMX แสดง Peripheral Assignment ทั้งหมดของระบบ 59

3.22 วงจรทดสอบ Locked Rotor Test แสดงการต่อ Shunt Resistor อนุกรมกับมอเตอร์ เพื่อวัดกระแสทางอ้อมผ่าน Oscilloscope 61

3.23 Oscilloscope Waveform จาก Locked Rotor Test ที่ $V_{in} = 12$ V แสดงแรงดัน V_{sh} ที่ไต่ระดับแบบ First-order Exponential พร้อมจุดอ่านค่า $V_{sh,ss}$ และ τ . . . 62

3.24 เปรียบเทียบ Causal Filter (สีแดง) และ `filtfilt` (สีน้ำเงิน) ในโดเมนเวลา (ก) Causal Filter มี Phase Lag สังเกตได้ชัดเจน (ข) `filtfilt` ทับ Raw Signal ได้ตรงทุกจุด (ค) Phase Difference ระหว่างสองวิธี 64

3.25 การตรวจสอบความเหมาะสมของ $f_c = 10$ Hz กับข้อมูลจริงของระบบ: (ซ้าย) `filtfilt` ให้ Phase Shift เป็นศูนย์ และ Attenuation ที่ 2 Hz น้อยกว่า 0.001% (ขวา) Signal Dynamics ของระบบจริงทั้งหมด อยู่ในย่าน 0.1--2 Hz ห่างจาก f_c ถึง 5 เท่า ยืนยันว่า Filter ไม่รบกวน Signal ที่ต้องการ 65

3.26 Simulink Block Diagram ของแบบจำลอง DC Motor ที่ใช้ใน Grey-box Parameter Estimation แสดง Free Parameters ที่ปรับค่าโดย `greyest` 66

3.27 Cross-Validation Fit Score ทั้ง 21 Runs แสดงเป็น Histogram จำแนกตามประเภทข้อมูล เส้นประสีแดงคือเกณฑ์ขั้นต่ำที่ยอมรับได้ 80% 67

3.28 เปรียบเทียบ Measured (เส้นทึบ) และ Simulated (เส้นประ) ทั้ง 21 Runs จำแนกตามประเภท Input ได้แก่ Chirp 3 ย่านความถี่, Step 3 แรงดัน และ Stair-Step . . 67

3.29 Block Diagram ของระบบควบคุม Cascade PID แสดง Position Outer Loop, Velocity Inner Loop, Feedforward 3 ช่องทาง และ Anti-windup Back-calculation 69

3.30 Root Locus ของ Velocity Inner Loop (PI + Reduced Plant) แสดง Desired Poles $s^*_{vel} = -80.000 \pm j38.747$ (เครื่องหมาย $\times$ สีแดง) บนเส้น Damping Ratio $\zeta = 0.9$ (เส้นประสีน้ำเงิน) 72

3.31 Root Locus ของ Position Outer Loop (P + Integrator Plant) แสดง Desired Poles $s^*_{pos} = -8.000 \pm j3.875$ (เครื่องหมาย $\times$ สีแดง) บนเส้น Damping Ratio $\zeta = 0.9$ (เส้นประสีน้ำเงิน) 73

3.32 Pole-Zero Map ใน Z-Domain ของ Velocity PI (ซ้าย) และ Position PID (ขวา) หลัง Tustin Discretization Poles และ Zeros ทั้งหมดอยู่ภายใน Unit Circle ยืนยันความเสถียรของ Discrete-time Controller 79

3.33 Bode Plot เปรียบเทียบ Continuous-time (สีเทา), Tustin (สีน้ำเงิน) และ Forward Euler (สีแดง) แสดงว่า Tustin ให้ Frequency Response ใกล้เคียง Continuous ในย่าน Passband ในขณะที่ Forward Euler มี Phase Error สูงกว่าเมื่อความถี่เข้าใกล้ $f_{Nyquist}$ 80

3.34 Open-Loop Bode Plot ของ Velocity Inner Loop แสดง Phase Margin ที่ Gain Crossover Frequency ยืนยันความเสถียรของ Velocity Loop 81

3.35 Open-Loop Bode Plot ของ Position Outer Loop (Cascade) แสดง Gain Margin และ Phase Margin ของระบบ Cascade รวมทั้งสอง Loop 82

3.36 Closed-loop Frequency Response แสดง Complementary Sensitivity $T(s)$ (บน) และ Sensitivity $S(s)$ (ล่าง) ของ Velocity Loop (ซ้าย) และ Position Loop (ขวา) เส้นประแสดงจุด -3 dB Bandwidth ของแต่ละ Loop 83

3.37 ZVD 3-Impulse Sequence แสดงแอมพลิจูด A_1, A_2, A_3 และเวลาหน่วง $t_2 = 0.259$ s (26 steps) และ $t_3 = 0.518$ s (52 steps) ที่ Outer Loop 100 Hz 85

3.38 Sensitivity Analysis เปรียบเทียบ ZV (สีแดง) และ ZVD (สีน้ำเงิน) แสดง Residual Vibration เมื่อ $\omega_{actual}/\omega_{design}$ เบี่ยงจาก 1.0 ZVD รักษา Residual Vibration ต่ำกว่า 5% ในย่านกว้างกว่า ZV อย่างมีนัยสำคัญ 86

3.39 (ก) S-curve Acceleration Profile ของแขนกล Unshaped (สีเทา) และ ZVD-Shaped (สีน้ำเงิน) (ข) การแกว่งของ Connection Rod ϕ_{rod} แสดง Settling Time ของ Unshaped และการยกเลิกการแกว่งของ ZVD-Shaped สำหรับ Worst-case Move $360°$ 87

3.40 ลักษณะจำเพาะของ S-Curve Motion Profile สำหรับ 1 การเคลื่อนที่ (Worst-case $360°$) แสดงความสัมพันธ์ของ (ก) Position (ข) Velocity (ค) Acceleration และ (ง) Jerk โดยแบ่งการทำงานออกเป็น 7 เฟส (P1-P7) ซึ่งแสดงให้เห็นถึงความต่อเนื่องของความเร่ง และการจำกัดอัตราการกระชาก (Jerk-limited) อย่างชัดเจน 90

3.41 Motion Profile ของ Scenario 1 (Worst-case $0° \leftrightarrow 360°$) แสดง (ก) Position (ข) Velocity (ค) Acceleration และ (ง) Rod Swing Angle เปรียบเทียบ Unshaped (สีแดง), ZVD-Shaped (สีน้ำเงิน) และ Rigid Rod (สีเขียว) เส้นประแนวตั้งแสดงจุด Gripper Event Cycle Time = 30.41 s $\leq$ 35 s **(PASS)** 93

3.42 Motion Profile ของ Scenario 2 (Random Moderate) แสดง (ก) Position (ข) Velocity (ค) Acceleration และ (ง) Rod Swing Angle เปรียบเทียบ Unshaped, ZVD-Shaped และ Rigid Rod Cycle Time = 26.50 s $\leq$ 35 s **(PASS)** 94

3.43 สถาปัตยกรรม Protocol Multiplexing ที่จัดการข้อมูล Modbus RTU ร่วมกับ ASCII Command บนพอร์ต LPUART1 99

4.1 ภาพรวมชิ้นส่วนของหุ่นยนต์ (Exploded View) มุม Isometric 101

4.2 ภาพรวมชิ้นส่วนของหุ่นยนต์ (Exploded View) มุมมองด้านข้าง (Side View) 102

4.3 ภาพรวมของส่วนฐานหลังการประกอบ มุม Isometric 103

4.4 ภาพแตกชิ้นส่วน (Exploded View) ของส่วนฐาน 104

4.5 ภาพรวมของส่วนแขนกลและข้อต่อหลังการประกอบ มุม Isometric 106

4.6 ภาพแตกชิ้นส่วน (Exploded View) ของส่วนแขนกลและข้อต่อ 107

4.7 ภาพรวมของชุดเอ็นโค้ดเดอร์ มุม Isometric 108

4.8 ภาพแตกชิ้นส่วน (Exploded View) ของชุดเอ็นโค้ดเดอร์ 109

4.9 ภาพประกอบรวมขั้นสุดท้ายของหุ่นยนต์ พร้อมชิ้นส่วน 3D Print สำหรับจัดระเบียบสายไฟ 110

4.10 การยึดฐานเครื่องกับฐานสนามด้วยสกรู M8 $\times$ 14 mm ทั้ง 4 ตำแหน่ง 111

4.11 การติดตั้ง Field Encoder Shaft $\varnothing$8 mm เข้ากับ Coupling 112

4.12 การติดตั้ง Gripper ที่ปลายแขนด้วยสกรู M5 $\times$ 40 mm และ Lock Nut M5 113

4.13 Block Diagram แสดงสถาปัตยกรรมการจ่ายไฟและการแยกสัญญาณภายในตู้ควบคุม 115

4.14 ตู้ควบคุมไฟฟ้าที่ประกอบเสร็จสมบูรณ์: หน้าตู้ (ซ้าย) และภายในตู้ (ขวา) 116

4.15 แผนผังการต่อสายไฟ (Wiring Diagram) ของระบบควบคุม 117

4.16 แผนภาพสถานะการทำงาน (Application FSM) ของระบบควบคุม 119

4.17 สถาปัตยกรรมระบบ (Cyber-Physical System Architecture) ของหุ่นยนต์ 123

5.1 Encoder Noise Floor ขณะหยุดนิ่ง: Time Series (บน) และ Histogram ของ Deviation (ล่าง) 128

5.2 ความซ้ำซากได้ของ Homing Position 10 ครั้ง เทียบกับเกณฑ์ $\pm 0.1°$ (A2) 129

5.3 Oscilloscope Waveform จาก Locked Rotor Test ($V_{in} = 12$ V) แสดง Steady-state Current I_{ss} และค่าคงเวลา τ_{RL} ที่ใช้คำนวณ R_a และ L_a 132

5.4 สัญญาณ Chirp ก่อนและหลังกรองด้วย Butterworth อันดับ 4 ที่ $f_c = 10$ Hz ด้วยวิธี Zero-phase (`filtfilt`) แสดง V_{in} (บน) และ ω_{arm} (ล่าง) 133

5.5 ผล Cross-Validation: เปรียบเทียบ ω_{arm} จริง (น้ำเงิน) กับ Simulated (แดง) สำหรับ Chirp 0.5, 1.0 และ 2.0 Hz 135

5.6 Root Locus ของ Velocity Inner Loop (ซ้าย) และ Position Outer Loop (ขวา) แสดง Desired Poles และ Actual Closed-loop Poles 136

5.7 Step Response บน Hardware (With Load, $\theta_{ref} = 40°$): Position Tracking (บน), Tracking Error (กลาง), และ Velocity Profile (ล่าง) เทียบกับข้อกำหนด P5 และ P6 139

5.8 ความแม่นยำตำแหน่ง (A1) ที่ทุก Workstation: Mean Error $\pm\sigma$ เทียบกับเกณฑ์ $\pm 0.5°$ 140

5.9 Run Chart ความซ้ำซากได้ 100 รอบที่ตำแหน่ง $40°$, $80°$, $120°$ เทียบกับเกณฑ์ $\pm 0.1°$ (A2) 142

5.10 Cycle Timeline เต็มรูปแบบ (4 ชิ้นงาน, 9 Moves): Move Time, ZVD Delay และ Gripper Operation เทียบกับเกณฑ์ P1 (≤ 35 s) 144

5.11 Robot Motion Profile (Position & Velocity) ทั้ง 3 รอบ: รอบที่ 1 (น้ำเงิน, ≈30.5 s), รอบที่ 2 (แดง, ≈36.8 s — เกินเกณฑ์) และรอบที่ 3 (ส้ม, ≈33.9 s) แสดงตำแหน่ง (บน) และความเร็ว (ล่าง) ตลอด Cycle เต็มรูปแบบ 145

A.1 Engineering Drawing: Body Plate 152

A.2 Engineering Drawing: Top Plate 153

A.3 Engineering Drawing: AMT Encoder Mounting 154

A.4 Engineering Drawing: Bearing Mounting 155

A.5 Engineering Drawing: Lower Plate 156

A.6 Engineering Drawing: Step Shaft 157

A.7 Engineering Drawing: Arm 158

A.8 Engineering Drawing: Plate V 159

สารบัญตาราง

1.1 ข้อกำหนดความต้องการของระบบทั้ง 32 ข้อ 3

1.2 สรุป Subsystem หลักของระบบ 5

1.3 สรุปงบประมาณโครงงาน จำแนกตามหมวดหมู่ 6

3.1 พารามิเตอร์ภาระโหลดของระบบเพลา 41

3.2 สรุปผลการคำนวณเพลาและการออกแบบ Step Shaft 43

3.3 คุณสมบัติวัสดุที่ใช้ใน FEA เพลา (AISI 304) 43

3.4 สรุปผลการวิเคราะห์ FEA เพลาเทียบกับการคำนวณ ASME Code 47

3.5 ค่าเริ่มต้นของระบบสายพาน HTD5M 48

3.6 สรุปผลการคำนวณสายพาน HTD5M 50

3.7 พารามิเตอร์ห่วงโซ่การส่งกำลังสำหรับการคำนวณแรงบิด 51

3.8 ผลการคำนวณอายุการใช้งานของตลับลูกปืน 52

3.9 คุณสมบัติวัสดุที่ใช้ใน FEA (Galvanized Steel) 54

3.10 สรุปผลการวิเคราะห์ FEA เทียบกับข้อกำหนด M3 55

3.11 การคำนวณกำลังไฟฟ้าของ 24 V Motor Bus 56

3.12 การคำนวณกำลังไฟฟ้าของ 5 V Logic Bus 57

3.13 การเลือกขนาด Circuit Breaker และ Fuse 57

3.14 STM32G474RE Pin Assignment (จาก CubeMX IOC) 60

3.15 ผลการวัด R_a และ L_a จาก Locked Rotor Test ทั้ง 3 แรงดัน 63

3.16 การตั้งค่า Chirp Signal สำหรับ Grey-box Estimation 66

3.17 ผลการ Cross-Validation ของแบบจำลองกับข้อมูลประเภทต่าง ๆ 66

3.18 พารามิเตอร์ที่ได้จาก System Identification 68

3.19 สรุปค่า PID Gain และ Feedforward ของระบบควบคุม 75

3.20 เปรียบเทียบค่า PID Gain จาก Theoretical Design และ Hardware Commissioning 76

3.21 ผลการทดสอบ Step Response บน Hardware เทียบกับ Requirements 77

3.22 สรุปค่าสัมประสิทธิ์ Discrete-time Controller (Theoretical Gains, หน่วย V) . . . 79

3.23 สรุป Stability Margin ของ Cascade PID 82
3.24 พารามิเตอร์ ZVD Input Shaper ที่ใช้ใน Firmware 86
3.25 สรุปผลการ Simulation เปรียบเทียบ Unshaped กับ ZVD-Shaped (Worst-case Move $360°$) 87
3.26 พารามิเตอร์ S-Curve Motion Profile จาก Hardware Identification 89
3.27 Move Sequence ของ Scenario 1 (Worst-case) 92
3.28 Move Sequence ของ Scenario 2 (Random Moderate) 92
3.29 สรุป Cycle Time เปรียบเทียบกับข้อกำหนด P1 93
3.30 สรุปการวิเคราะห์ Rod Swing เทียบกับเกณฑ์การประกอบ 95
3.31 Modbus Write Register Map (PC $\rightarrow$ Robot) 96
3.32 Modbus Read Register Map (Robot $\rightarrow$ PC) 97
3.33 สรุปผลการออกแบบระบบทั้งหมดเทียบกับ Requirements 100

4.1 รายการชิ้นส่วนสำหรับประกอบส่วนฐาน 103
4.2 รายการชิ้นส่วนสำหรับประกอบส่วนแขนกล 105
4.3 รายการชิ้นส่วนสำหรับประกอบชุดเอ็นโค้ดเดอร์ 108
4.4 Bill of Materials (BOM) ของตู้ควบคุม 114
4.5 โครงสร้างโมดูลและไลบรารีใน STM32 Firmware 118
4.6 PID Gains และ Feedforward ที่ใช้งานจริงบนระบบฮาร์ดแวร์ 121
4.7 ตารางการ Map Command Character สำหรับ Joystick 124

5.1 ผลการวัดระยะคลอนของระบบส่งกำลัง (ค่าเฉลี่ยจากการวัด 3 ตำแหน่ง $\times$ 5 ครั้ง) . 127
5.2 ผลการทดสอบ Encoder AMT-103 128
5.3 ผลการทดสอบ Proximity Sensor สำหรับ Homing 129
5.4 ผลการวัด R_a และ L_a จาก Locked Rotor Test ที่ $V_{in} = 12$ V (3 runs) 131
5.5 ผลการ Estimate พารามิเตอร์จาก Chirp Test (สรุปจาก 9 Experiments) 133
5.6 ผล Cross-Validation ของแบบจำลองกับข้อมูลจริง 134
5.7 สรุปผล Simulation เปรียบเทียบ Unshaped กับ ZVD-Shaped (Worst-case Move $360°$, มีโหลด Connection Rod) 136
5.8 Process Noise $\mathbf{Q}$ และ Measurement Noise R ที่ใช้ใน Kalman Filter 137
5.9 ผลการทดสอบ Step Response บน Hardware จริง (With Load) 138

5.10 ผลการทดสอบความแม่นยำตำแหน่ง (A1) --- ค่าเฉลี่ยจาก 10 ครั้ง 140
5.11 ผลการทดสอบความซ้ำซากได้ 100 รอบ (A2) 141
5.12 ผลการทดสอบ Cycle Time (P1) --- 3 รอบต่อเนื่อง 143
5.13 Requirements Traceability Matrix (32 ข้อ) 146

บทที่ 1 บทนำ

1.1. ที่มาและความสำคัญของโครงงาน

บริษัท FIBO จำกัด มีความต้องการจัดสร้างเครื่องจักรสำหรับกระบวนการผลิตในอุตสาหกรรม เพื่อใช้ในการหยิบจับชิ้นงาน (Connection Rod) โดยมีข้อจำกัดด้านพื้นที่และการทำงานที่ชัดเจน คณะผู้จัดทำจึงขอนำเสนอโครงการออกแบบและสร้างระบบหุ่นยนต์แขนกลหยิบจับชิ้นงานแบบหมุน

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาระบบแขนกลที่สามารถทำงานร่วมกับชุดหัวจับ (End Effector) ที่ลูกค้ากำหนด โดยออกแบบให้สามารถรับภาระโหลดรวมได้สูงสุดถึง 2 กิโลกรัม และสามารถเคลื่อนที่หยิบวางชิ้นงานได้อย่างแม่นยำด้วยความเร็วสูง เพื่อให้ได้เวลาต่อรอบการทำงาน (Cycle Time) ต่ำกว่า 35 วินาที ภายใต้โครงสร้างทางกลที่มีความแข็งแรง และระบบควบคุมความปลอดภัยทางไฟฟ้าที่ได้มาตรฐาน ตามข้อกำหนดของลูกค้า

1.2. วัตถุประสงค์ของโครงงาน

โครงการนี้มีวัตถุประสงค์หลัก 4 ประการ ดังนี้

1. เพื่อออกแบบและสร้างหุ่นยนต์แขนกล 1 แกนแบบหมุน สำหรับงานหยิบจับชิ้นงาน (Connection Rod) ร่วมกับชุดหัวจับที่ลูกค้ากำหนด ให้มีความแข็งแรงเพียงพอรับโหลด 2 กิโลกรัม และมีอายุการใช้งานตามข้อกำหนดทางวิศวกรรม
2. เพื่อพัฒนาระบบควบคุมตำแหน่งที่มีความแม่นยำสูง สามารถลำเลียงชิ้นงานครบ 4 ชิ้นได้ภายใน Cycle Time ไม่เกิน 35 วินาที โดยมี Overshoot ไม่เกิน 1% และ Settling Time ของความเร็วไม่เกิน 0.5 วินาที
3. เพื่อสร้างตู้ควบคุมไฟฟ้าที่ได้มาตรฐานความปลอดภัยทางอุตสาหกรรม รองรับแหล่งจ่ายไฟ 220 V AC พร้อมระบบป้องกันและวงจรหยุดฉุกเฉิน ที่ไม่ตัดการทำงานของ Controller หลักเมื่อกด E-Stop
4. เพื่อเชื่อมต่อระบบหุ่นยนต์เข้ากับ Base System ของลูกค้า สำหรับรับคำสั่งและรายงานสถานะการทำงาน พร้อมรองรับการควบคุมด้วยตนเองผ่าน Joystick แบบ Wireless

1.3. ขอบเขตและข้อจำกัดของโครงงาน

1.3.1 ขอบเขตของโครงงาน

โครงการนี้มุ่งเน้นการสร้างระบบหุ่นยนต์หยิบจับชิ้นงานแบบหมุน โดยมีขอบเขตการดำเนินงานหลัก ดังนี้

- **ด้านกลไกและโครงสร้าง:** สร้างหุ่นยนต์รัศมีการทำงาน 500 มิลลิเมตร บนพื้นที่ฐาน 200×200 มิลลิเมตร ที่สามารถรองรับน้ำหนักที่ปลายแขน 2 กิโลกรัม โดยโครงสร้างส่วนรับแรงต้องใช้วัสดุทางวิศวกรรมมาตรฐาน และมีจุดเชื่อมต่อสำหรับระบบตรวจสอบของลูกค้า
- **ด้านไฟฟ้าและการควบคุม:** พัฒนาตู้ควบคุมมาตรฐานอุตสาหกรรมพร้อมระบบความปลอดภัย และวงจรหยุดฉุกเฉิน (E-Stop) ที่ไม่ตัดไฟระบบหลัก โดยใช้ไมโครคอนโทรลเลอร์เป็นศูนย์กลางสั่งการชุดขับเคลื่อนมอเตอร์
- **ด้านสมรรถนะการทำงาน:** ระบบต้องสามารถประมวลผลและเคลื่อนที่เพื่อหยิบวางชิ้นงาน 4 ชิ้น ให้สำเร็จภายใน 35 วินาที ภายใต้ข้อกำหนดด้าน Overshoot ไม่เกิน 1%
- **ด้านการเชื่อมต่อและการสื่อสาร:** พัฒนาระบบให้รองรับการสั่งงานผ่านอุปกรณ์ Joystick และเชื่อมต่อกับ Base System ด้วยโปรโตคอล Modbus RTU ครอบคลุมการทำงานทั้งโหมด Manual และ Auto

1.3.2 ข้อจำกัดของโครงงาน

โครงงานนี้ไม่ครอบคลุมส่วนประกอบที่อยู่ภายใต้ความรับผิดชอบของลูกค้า ดังนี้

- **Base System ของลูกค้า:** ซอฟต์แวร์และฮาร์ดแวร์ของระบบควบคุมหลักที่ส่งคำสั่งผ่าน Modbus ไม่ได้อยู่ในขอบเขตการออกแบบและทดสอบของโครงงานนี้ โครงงานรับประกันเฉพาะ Interface ของ Modbus RTU ฝั่ง Slave เท่านั้น
- **End Effector (Gripper):** ชุดหัวจับที่ลูกค้ากำหนดไม่ได้อยู่ในขอบเขตการออกแบบ โครงงานนี้รับผิดชอบเฉพาะ End Effector Housing และสัญญาณควบคุม Digital I/O สำหรับ Gripper เท่านั้น
- **การควบคุมแกนแนวดิ่ง (Z-axis):** ระบบนี้เป็นหุ่นยนต์ 1 แกน (Rotary) การเคลื่อนที่แนวดิ่งของ Gripper ดำเนินการโดย Pneumatic Actuator ของลูกค้าโดยตรง ไม่ใช่โดย Controller ของโครงงานนี้

1.4. ข้อกำหนดความต้องการของระบบ

เพื่อให้สอดคล้องกับขอบเขตโครงการ คณะผู้จัดทำได้กำหนดข้อกำหนดความต้องการ ทางเทคนิคสำหรับใช้อ้างอิงในการออกแบบและทดสอบระบบ โดยแบ่งออกเป็น 5 หมวดหมู่ ได้แก่ **M (Mechanical)** โครงสร้างและกลไก, **P (Performance)** สมรรถนะการทำงาน, **A (Accuracy & Control)** ความแม่นยำและการควบคุม, **E (Electrical & Interface)** ระบบไฟฟ้าและการเชื่อมต่อ และ **J (Joystick)** การควบคุมผ่านจอยสติ๊ก

ตารางที่ 1.1: ข้อกำหนดความต้องการของระบบทั้ง 32 ข้อ

Req ID	หมวด	คำอธิบาย
M1	Mechanical	ขนาดฐานไม่เกิน 200×200 mm
M2	Mechanical	Actuator และ Sensor สามารถยื่นออกนอกขนาดฐานได้
M3	Mechanical	End Effector รวมชิ้นงาน รับน้ำหนักได้สูงสุด 2 kg
M4	Mechanical	ใช้ Brushed DC Motor พร้อม Incremental Encoder
M5	Mechanical	อายุใช้งานทนทาน ≥ 3 รอบการทดสอบ (รอบละ 1 ชั่วโมง)
M6	Mechanical	ชิ้นส่วนรับแรงห้ามใช้ 3D Print เด็ดขาด
M7	Mechanical	ยึดฐานสแตนเลส AISI304 ของลูกค้าด้วยสกรู M8 เหล็กกรมดำ
M8	Mechanical	เพลาต่อ Encoder ลูกค้า ∅8 mm (สูง 400 mm, ยื่น $\geq$ 20 mm)
M9	Mechanical	หมุนต่อเนื่องได้ $\geq 540°$ โดยสายไฟไม่พันกัน
M10	Mechanical	จุดต่ำสุดของกริปเปอร์ต้องไม่ชนกับ Connection Rod
P1	Performance	ทำงานครบ 1 รอบ (หยิบวาง 4 ชิ้น + กลับ Homing) ใน $\leq$ 35 วินาที
P2	Performance	Gripper ใช้เวลาหยิบ ~2 วินาที และวาง ~2 วินาที
P3	Performance	ความเร็วเชิงมุม ≥ 2.9 rad/s และ ≤ 7.5 rad/s
P4	Performance	ความเร่งเชิงมุม ≥ 2.0 rad/s^2 และ ≤ 28.0 rad/s^2
P5	Performance	Settling Time ≤ 0.5 วินาที
P6	Performance	Overshoot $\leq 1\%$
P7	Performance	รองรับการทดสอบทั้งแบบมีโหลด (2 kg) และไม่มีโหลด
A1	Accuracy	ต้องสวมเพลา ∅12 mm ลงในรู ∅14 mm ได้ถูกต้อง
A2	Accuracy	ความแม่นยำซ้ำ 100 รอบ คลาดเคลื่อน $\leq \pm 0.1°$
A3	Accuracy	ผู้ใช้สามารถ Set จุด Homing เองได้หลังตั้งกับฐาน
A4	Accuracy	สั่งงานได้ 2 แบบ: (1) ระบุองศา (2) ระบุลำดับตำแหน่งจาก Homing
A5	Accuracy	หากสั่งงานนอกระยะ ระบบต้องแจ้ง Error หรือป้องกันไว้
E1	Electrical	สื่อสารผ่าน Modbus RTU
E2	Electrical	ใช้ STM32 เป็น Controller หลัก
E3	Electrical	สายต่อตู้ไฟ: ท่อลม ∅4 mm (4 เส้น), สาย GX (1 เส้น), สาย Motor (1 เส้น)
E4	Electrical	ตู้ไฟมาตรฐานอุตสาหกรรม: จัดสายสวยงาม, ติด Label, มี Schematic
E5	Electrical	หน้าตู้ไฟมีไฟแสดงสถานะ Power On/Off และ Emergency
E6	Electrical	E-Stop: ตัดการเคลื่อนที่แต่ Controller หลักยังมีไฟเลี้ยง (Reset ได้)
E7	Electrical	มีวงจรป้องกัน: ไฟกระชาก, กระแสเกิน, แรงดันตก
E8	Electrical	สลับโหมดการทำงาน Manual / Auto ได้
E9	Electrical	สัญญาณ Feedback กริปเปอร์: ขึ้นสุด, ลงสุด, หนีบสำเร็จ, ปล่อยสำเร็จ
J1	Joystick	จอยสติ๊กมีปุ่ม: กลับ Homing, Start, E-Stop

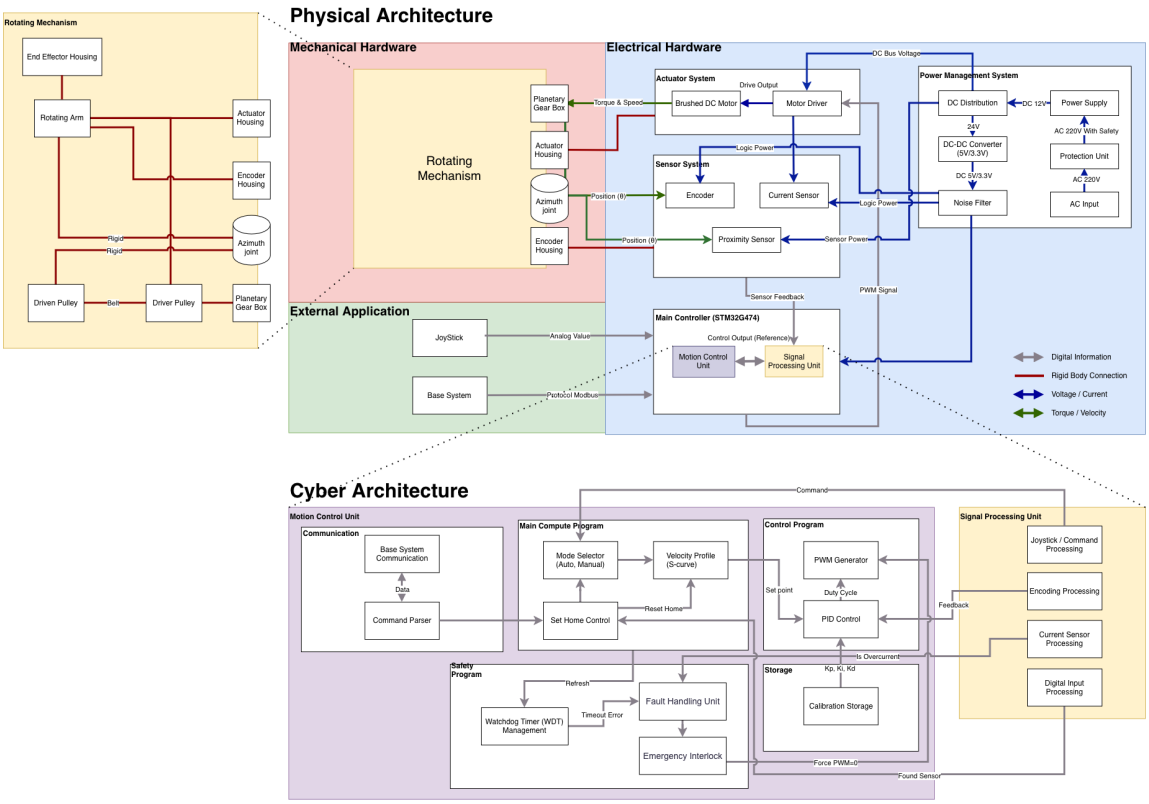
* ขอบบนของ P3 (≤ 7.5 rad/s) และ P4 (≤ 28.0 rad/s²) ปรับปรุงจาก Torque Budget ภายหลัง — ดูหัวข้อ 1.4.1

1.4.1 การปรับปรุงข้อกำหนด (Requirement Amendment)

จากการวิเคราะห์ Torque Budget ภายหลัง พบว่าชุดส่งกำลังและมอเตอร์สามารถรองรับความเร็วและความเร่งได้สูงกว่าขีดจำกัดเดิมที่ตั้งไว้แบบ Conservative ดังนั้นเพื่อให้บรรลุ Cycle Time (P1) จึงปรับขอบบนของ P3 เป็น $v_{max} \leq 7.5$ rad/s และ P4 เป็น $a_{max} \leq 28.0$ rad/s^2

1.5. ภาพรวมระบบ

ระบบ Circular Pick and Place Robot ออกแบบโดยแบ่งสถาปัตยกรรมออกเป็น 2 ระดับ ได้แก่ Physical Architecture ซึ่งอธิบายฮาร์ดแวร์จริงของระบบ และ Cyber Architecture ซึ่งอธิบายโครงสร้างซอฟต์แวร์ภายใน STM32G474RE ดังแสดงในรูปที่ 3.1



รูปที่ 1.1: System Architecture ของระบบ Circular Pick and Place Robot แสดง Physical Architecture และ Cyber Architecture

ระบบแบ่งออกเป็น 4 Subsystem หลักที่ทำงานร่วมกัน ดังนี้

Rotating Mechanism เป็นส่วนที่แปลงพลังงานไฟฟ้าจาก DC Motor ให้กลายเป็นการเคลื่อนที่เชิงมุมของแขนกล กำลังถูกส่งผ่าน Planetary Gearbox อัตราทด $N_g = 24$ ต่อด้วย Belt Drive HTD5M อัตราทด $N_b = 2.9$ รวมอัตราทดทั้งหมด $N = 70$ ก่อนถึงแขนกลรัศมี 500 mm ตำแหน่งเชิงมุมวัดโดย Encoder AMT-103 ที่ติดตั้งโดยตรงที่ Output Shaft เพื่อหลีกเลี่ยง Backlash Error จากระบบส่งกำลัง

Power Management System จ่ายพลังงานแก่ทุก Subsystem โดยแบ่งเป็น 2 Bus High Power Bus (24 V/ 10 A) จ่ายให้มอเตอร์ผ่าน Magnetic Contactor ที่ถูกควบคุมโดย E-Stop และ Logic Bus (5 V) จ่ายให้ STM32, ESP32 และ Sensor โดยต่อก่อน Contactor ทำให้ Controller ยังมีไฟเลี้ยงแม้กด E-Stop ตามข้อกำหนด E6

Main Controller STM32G474RE ทำหน้าที่เป็นศูนย์กลางประมวลผลทั้งหมด รัน Cascade PID Control Loop โดยมี Velocity Inner Loop ที่ความถี่ 1 kHz และ Position Outer Loop ที่ความถี่ 100 Hz พร้อม Kalman Filter 4 สถานะสำหรับประมาณค่าตำแหน่ง ความเร็ว กระแส และแรงบิดรบกวน และ ZVD Input Shaper ($f_n = 12.13$ Hz, $\zeta = 0.041$) สำหรับลด Residual Vibration ของชิ้นงาน รับส่งข้อมูลกับ Base System ผ่าน Modbus RTU บน LPUART1 (230400 bps, 8N1, Slave Address 21)

Application FSM ระบบมี State Machine หลัก 5 สถานะ ได้แก่ `INIT` → `HOMING` → `IDLE` → `RUNNING` และ `FAULT` โดยทุกสถานะสามารถเข้าสู่ `FAULT` ได้จาก E-Stop หรือ Safety Fault และกลับสู่ `IDLE` ได้เมื่อ Release E-Stop และกด Reset

External Interface รองรับการสั่งงาน 2 โหมด ได้แก่ Manual Mode ผ่าน Wireless Joystick ที่รับสัญญาณ Bluetooth ด้วย ESP32 และส่งต่อผ่าน USART3 มายัง STM32 และ Auto Mode ผ่าน Base System ของลูกค้าโดยรับคำสั่งผ่าน Modbus RTU การควบคุม Gripper ใช้สัญญาณ Digital I/O โดยตรง

ตารางที่ 1.2: สรุป Subsystem หลักของระบบ

Subsystem	องค์ประกอบหลัก	รายละเอียดใน
Rotating Mechanism	DC Motor RS-775, Planetary Gearbox (N_g=24), Belt HTD5M (N_b=2.9), Encoder AMT-103	หัวข้อ 3.2.1
Power Management	PSU 24 V/10 A, PSU 5 V, Magnetic Contactor, E-Stop, Circuit Breaker, Fuse	หัวข้อ 3.3.1
Main Controller	STM32G474RE, Cascade PID (1 kHz / 100 Hz), Kalman Filter (4 states), ZVD Input Shaper, Modbus RTU Slave (Addr 21)	หัวข้อ 3.3.5, หัวข้อ 3.3.6
External Interface	Joystick + ESP32 (Bluetooth/USART3), Base System (Modbus RTU), Gripper (Digital I/O)	หัวข้อ 4.4.4, หัวข้อ 4.4.5

1.6. งบประมาณโครงงาน

งบประมาณรวมที่ได้รับอนุมัติสำหรับโครงงานนี้คือ **17,000 บาท** ตารางที่ 1.3 สรุปรายการจัดซื้อหลักจำแนกตามหมวดหมู่ (รายการสิ้นเปลืองเล็กน้อย เช่น น็อต สกรู แหวน และสายไฟ รวมไว้เป็นรายการกลุ่ม)

หมายเหตุ: ยอดรวมในตารางนี้แสดงเฉพาะค่าใช้จ่ายของอุปกรณ์ที่นำมาใช้งานจริงในระบบขั้นสุดท้าย ไม่รวมค่าใช้จ่ายจากการทดลองและชิ้นส่วนที่เสียหายระหว่างกระบวนการพัฒนา (ค่าลองผิดลองถูก)

ตารางที่ 1.3: สรุปงบประมาณโครงงาน จำแนกตามหมวดหมู่

รายการ	จำนวน	หน่วย	ราคาต่อหน่วย	รวม (บาท)
ระบบขับเคลื่อน (Drive System)				
ZGX60RMM Gearmotor 24V 250RPM (RS-775 + Planetary 24:1)	1	ชุด	2,800	2,800
เฟืองสายพาน HTD5M 70T (Driven Pulley)	1	ชิ้น	600	600
เฟืองสายพาน HTD5M 24T (Driver Pulley)	1	ชิ้น	230	230
สายพาน HTD5M หน้ากว้าง 15 mm	1	เส้น	300	300
รวมระบบขับเคลื่อน				*3,930*
Encoder & Sensor				
CUI AMT103-V Incremental Encoder	1	ชิ้น	1,250	1,250
WCS1800 Hall Effect Current Sensor	1	ชิ้น	360	360
Proximity Sensor PR12-2DN (Homing)	1	ชิ้น	320	320
รวม Encoder & Sensor				*1,930*
อิเล็กทรอนิกส์และระบบควบคุม				
Cytron MD10C 10A DC Motor Driver	1	ชิ้น	430	430
ESP32-WROOM-32U (Bluetooth Receiver)	1	ชิ้น	240	240
ESP-32S DevKit V1	1	ชิ้น	170	170
Opto 817 Module 8 ช่อง (Optocoupler)	1	ชิ้น	25	25
Relay Module 8 CH 5V	1	ชุด	190	190
Relay Module 4 CH 5V	2	ชุด	37	75
PCB 15×20 cm + วัสดุประกอบ	1	ชุด	49	49
MOCUTE-052 VR Joystick (Bluetooth)	1	ชิ้น	123	123
XL4016E1 Step-Down Converter 8A	1	ชิ้น	90	90
รวมอิเล็กทรอนิกส์และระบบควบคุม				*1,392*
ระบบจ่ายไฟและความปลอดภัย				
Switching Power Supply 24V 10A	1	ชุด	320	320
Switching Power Supply 5V 4A	1	ชุด	110	110
Magnetic Contactor S-N10 24V	1	ชิ้น	230	230
Contactor 24VDC	1	ชิ้น	380	380

รายการ	จำนวน	หน่วย	ราคาต่อหน่วย	รวม (บาท)
Circuit Breaker AC 2P	2	ชิ้น	170	340
Circuit Breaker DC 2P 6A	2	ชิ้น	126	252
Emergency Stop Button 22 mm	1	ชิ้น	45	45
Pilot Lamp 24VDC (หลายสี)	6	ชิ้น	27	162
รวมระบบจ่ายไฟและความปลอดภัย				*1,839*
ชิ้นส่วนเชิงกล				
Custom Machined Parts (ชิ้นส่วนกลึง/กัด)	2	ชุด	1,295	2,590
SKF 30203 Tapered Roller Bearing	1	ชิ้น	228	228
Miniature Flange Bearing F6901ZZ	1	ชิ้น	88	88
MF126ZZ Bearing	1	ชิ้น	45	45
ลูกปืนทั่วไป	17	ชิ้น	25	425
Flexible Shaft Coupling 8×8 mm	1	ชิ้น	130	130
น็อต สกรู แหวน และอุปกรณ์ยึด (รวม)	---	---	---	600
บูชรอง (Spacer) ชุด	---	---	---	299
รวมชิ้นส่วนเชิงกล				*4,405*
สายไฟและขั้วต่อ				
สายไฟ AC/DC, สายสัญญาณ (รวม)	---	---	---	487
GX12/GX16/GX20 Connector (รวม)	---	---	---	450
Terminal Block, Bus Bar (รวม)	---	---	---	200
รวมสายไฟและขั้วต่อ				*1,137*
วัสดุสิ้นเปลือง				
Matte PETG Filament (Black + Blue)	2	ม้วน	263	525
ไส้ไก่, ทองแดง PCB, สารหล่อลื่น (รวม)	---	---	---	180
รวมวัสดุสิ้นเปลือง				*705*
รวมทั้งหมด				**15,338**
งบที่ได้รับ				**17,000**
คงเหลือ				**1,662**

1.7. โครงสร้างของรายงาน

รายงานฉบับนี้จัดแบ่งเนื้อหาออกเป็น 6 บทหลัก ดังนี้

1. **บทที่ 1 บทนำ:** อธิบายที่มาและความสำคัญ วัตถุประสงค์ ขอบเขต ข้อกำหนดของระบบทั้ง 32 ข้อ และภาพรวมสถาปัตยกรรมระบบ
2. **บทที่ 2 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง:** นำเสนอทฤษฎีพื้นฐานที่ใช้ในการออกแบบ ได้แก่ DC Motor State-Space, Cascade PID, ZV Input Shaping, Kalman Filter, มาตรฐาน ASME/ISO และ S-Curve Trajectory
3. **บทที่ 3 สถาปัตยกรรมและการออกแบบระบบ:** นำเสนอรายละเอียดการออกแบบเชิงกล ไฟฟ้า และระบบควบคุม รวมถึงการระบุระบบ (System Identification) และการออกแบบตัวควบคุม
4. **บทที่ 4 การสร้างและบูรณาการระบบ:** บันทึกขั้นตอนการประกอบจริงและการพัฒนา Firmware รวมถึงการเชื่อมต่อระบบย่อยทั้งหมดเข้าด้วยกัน
5. **บทที่ 5 ผลการทดสอบและการทวนสอบ:** รายงานผลการทดสอบเชิงประจักษ์และการทวนสอบข้อกำหนด 32 ข้อ พร้อมตาราง Requirements Traceability
6. **บทที่ 6 สรุปผลและข้อเสนอแนะ:** สรุปภาพรวมความสำเร็จ ปัญหาที่พบ บทเรียนที่ได้ และแนวทางการพัฒนาต่อยอด

บทที่ 2 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

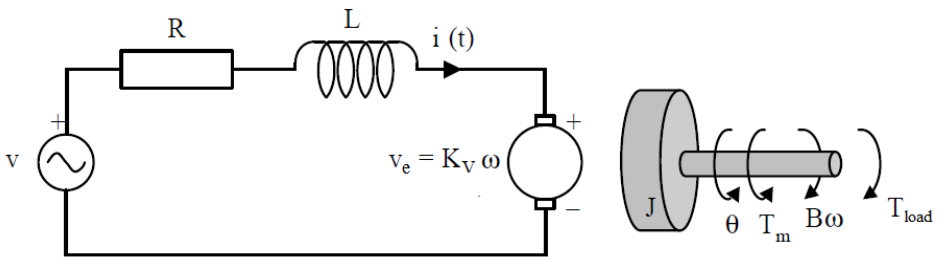
2.1. พลศาสตร์ของมอเตอร์กระแสตรง

มอเตอร์กระแสตรง (DC Motor) สามารถจำลองพฤติกรรมได้จากสมการสองโดเมน ได้แก่ โดเมนวงจรไฟฟ้าและโดเมนพลศาสตร์เชิงกล โดยทั้งสองโดเมนเชื่อมโยงกันผ่านค่าคงที่ของมอเตอร์ ก่อนการสร้างแบบจำลองทางคณิตศาสตร์ ได้กำหนดสมมติฐานเพื่อลดความซับซ้อนของระบบให้สามารถวิเคราะห์แบบระบบเชิงเส้นไม่แปรผันตามเวลา (Linear Time-Invariant: LTI) ได้ดังนี้

- วงจรแม่เหล็กทำงานในช่วงเชิงเส้น (Linear Magnetic Circuit) และไม่เกิดการอิ่มตัวของแกนเหล็ก
- ค่าความต้านทาน R_a และตัวเหนี่ยวนำ L_a ของอาร์เมเจอร์มีค่าคงที่ ไม่เปลี่ยนแปลงตามอุณหภูมิที่เพิ่มขึ้น
- ละทิ้งผลของแรงดันตกคร่อมที่แปรงถ่าน (Brush Voltage Drop) และแรงเสียดทานสถิต (Coulomb Friction)

2.1.1 สมการโดเมนไฟฟ้า

วงจรอาร์เมเจอร์ประกอบด้วยความต้านทาน R_a และตัวเหนี่ยวนำ L_a ต่อในอนุกรมกับแรงเคลื่อนไฟฟ้าย้อนกลับ (Back-EMF) e_b ตามกฎแรงดันของ Kirchhoff ได้สมการ



รูปที่ 2.1: แผนภาพ Electro-mechanical ของมอเตอร์กระแสตรง แสดงความสัมพันธ์ระหว่างโดเมนไฟฟ้าและโดเมนเชิงกล

$$V_a(t) = R_a\, i_a(t) + L_a \frac{di_a(t)}{dt} + e_b(t) \tag{2.1}$$

โดยที่ $e_b = K_b \omega_m$ และ K_b คือค่าคงที่ Back-EMF ($\mathrm{V\,s\,rad^{-1}}$) กำหนดให้ ω_m คือความเร็วเชิงมุมที่เพลามอเตอร์ ซึ่งมีความสัมพันธ์กับความเร็วเชิงมุมที่เพลา Output ω (หลัง Gearbox อัตราทด N) ตามความสัมพันธ์ทางจลนศาสตร์ $\omega_m = N\omega$ ดังนั้นเมื่อแทนค่ากลับลงในสมการ (2.1) จะได้สมการเชิงอนุพันธ์ของโดเมนไฟฟ้าที่สอดคล้องกับความเร็วเพลา Output ดังนี้

$$L_a \frac{di_a}{dt} = V_a - R_a\, i_a - K_b\, N\, \omega \tag{2.2}$$

2.1.2 สมการโดเมนเชิงกล

แรงบิดแม่เหล็กไฟฟ้า τ_m ที่มอเตอร์ผลิตได้เป็นสัดส่วนกับกระแสอาร์เมเจอร์ i_a

$$\tau_m = K_t\, i_a \tag{2.3}$$

โดยที่ K_t คือค่าคงที่แรงบิด ($\mathrm{N\,m\,A^{-1}}$) ทั้งนี้ ตามหลักการอนุรักษ์พลังงานในอุดมคติของระบบหน่วย SI กำลังงานเชิงกลที่สร้างได้จะเท่ากับกำลังงานไฟฟ้าที่ป้อนเข้า (Mechanical Power = Electrical Power) ส่งผลให้ค่าคงที่แรงบิดและค่าคงที่ Back-EMF มีค่าเท่ากันในเชิงตัวเลข ($K_t = K_b$)

กฎที่สองของนิวตันสำหรับการหมุนที่ประยุกต์ใช้กับเพลา Output (พิจารณาผ่าน Gearbox อัตราทด N และประสิทธิภาพ η) จะได้

$$J \frac{d\omega}{dt} = \eta\, N\, K_t\, i_a - B\, \omega - \tau_d \tag{2.4}$$

โดยที่ J คือโมเมนต์ความเฉื่อยรวมทั้งหมดที่ Reflect กลับมายังเพลา Output ($\mathrm{kg\,m^2}$), B คือค่าสัมประสิทธิ์แรงหน่วงหนืด ($\mathrm{N\,m\,s\,rad^{-1}}$), และ τ_d คือแรงบิดรบกวนจากภายนอก

2.1.3 State-Space Representation

นิยาม State Vector $\mathbf{x} = [\theta,\ \omega,\ i_a,\ \tau_d]^\top$ โดยที่ θ คือมุมตำแหน่ง (rad), ω คือความเร็วเชิงมุม (rad/s), i_a คือกระแสอาร์เมเจอร์ (A) และ τ_d คือแรงบิดรบกวนที่ Augment เพิ่มเข้ามา ระบบ 4 สถานะในรูปแบบ Continuous-Time State-Space เขียนได้เป็น

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u, \qquad y = \mathbf{C}\mathbf{x} \tag{2.5}$$

โดยที่

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{B}{J} & \frac{\eta N K_t}{J} & -\frac{1}{J} \\ 0 & -\frac{K_b N}{L_a} & -\frac{R_a}{L_a} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L_a} \\ 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \tag{2.6}$$

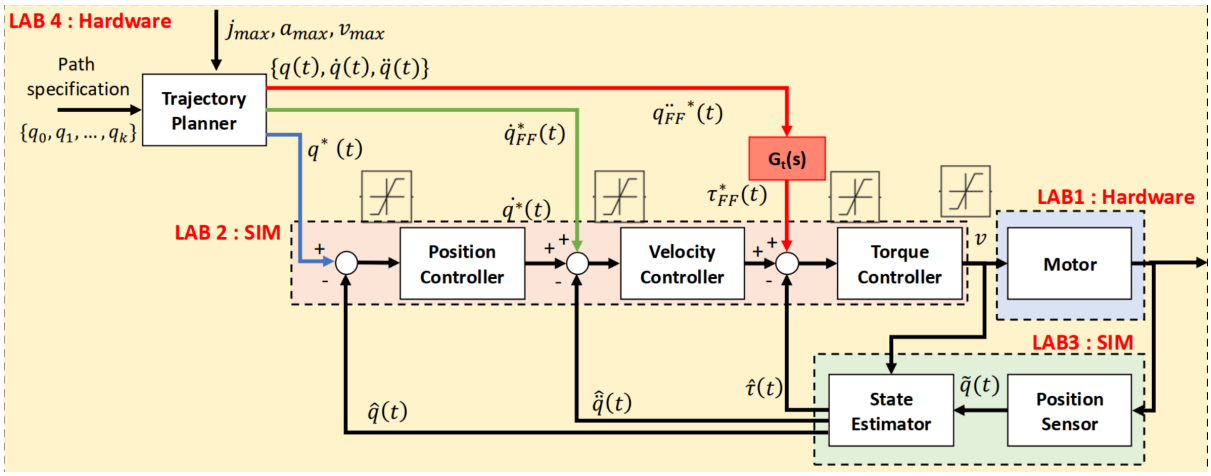
input $u = V_a$ คือแรงดันที่จ่ายให้อาร์เมเจอร์ และ output $y = \theta$ (เนื่องจากเซนเซอร์สามารถวัดได้เพียงตำแหน่งเชิงมุม) สถานะ τ_d เป็น Augmented State ที่ถือว่าเปลี่ยนแปลงช้า ($\dot{\tau}_d \approx 0$) ใช้สำหรับเป็น State Estimator ใน

Kalman Filter เพื่อประมาณค่าความเร็วและแรงบิดรบกวนแบบ Real-time ตามที่อธิบายในหัวข้อ 2.2.3

2.2. ทฤษฎีการควบคุมและการกรองสัญญาณ

2.2.1 Cascade PID Control

โครงสร้าง Cascade PID ประกอบด้วยวงรอบควบคุมสองชั้นที่ซ้อนกัน โดยวงรอบใน (Inner Loop) ควบคุมความเร็วเชิงมุม ω และวงรอบนอก (Outer Loop) ควบคุมตำแหน่งเชิงมุม θ



รูปที่ 2.2: Block Diagram ของระบบควบคุม Cascade PID แสดงวงรอบความเร็วและวงรอบตำแหน่ง

Velocity Inner Loop

ตัวควบคุม PID สำหรับวงรอบความเร็วมี Transfer Function ในรูป

$$C_v(s) = K_{p,v} + \frac{K_{i,v}}{s} + K_{d,v}\, s \tag{2.7}$$

ความถี่ Sampling ของวงรอบความเร็วกำหนดที่ $f_v = 1$ kHz ($T_v = 1$ ms) การ Discretize ตัวควบคุมเข้าสู่ Digital Domain ใช้วิธี Tustin (Bilinear Transformation) สาเหตุที่เลือกใช้วิธีนี้แทนวิธี Forward Euler เพื่อป้องกันปัญหาความไม่เสถียร เนื่องจากวิธี Tustin จะทำการ Map โพล (Pole) ฝั่งซ้ายของระนาบ s ให้อยู่ภายใน Unit Circle ของระนาบ z อย่างสมบูรณ์ (Preserve Stability) โดยเฉพาะเมื่อระบบมีอัตราส่วน $K_{i,v}/f_v$ ที่มีขนาดใหญ่

$$s \longrightarrow \frac{2}{T_v} \cdot \frac{z-1}{z+1} \tag{2.8}$$

ค่า Gain ที่ใช้ในระบบจริงได้จาก Hardware Commissioning และแสดงรายละเอียดไว้ในหัวข้อ 3.3.6 และตารางที่ 3.19 นอกจากนี้ยังมีการเพิ่มระบบ Anti-windup แบบ Back-Calculation เพื่อป้องกันปัญหา Integral Windup เมื่อแรงดัน Output อิ่มตัวที่ ± 24 V โดยค่า Integrator State (I_k) จะถูกอัปเดตในแต่ละ Time Step ตาม

สมการ

$$I_k = I_{k-1} + K_{i,v}\, T_v\, e_k + K_{bc}\,(u_{sat,k} - u_{unsat,k}) \tag{2.9}$$

โดยที่ K_{bc} คือ Anti-windup Tracking Gain, $u_{sat,k}$ คือแรงดันควบคุมที่ถูกจำกัดแล้ว และ $u_{unsat,k}$ คือแรงดันควบคุมที่คำนวณได้ก่อนเข้า Saturation Block

Position Outer Loop

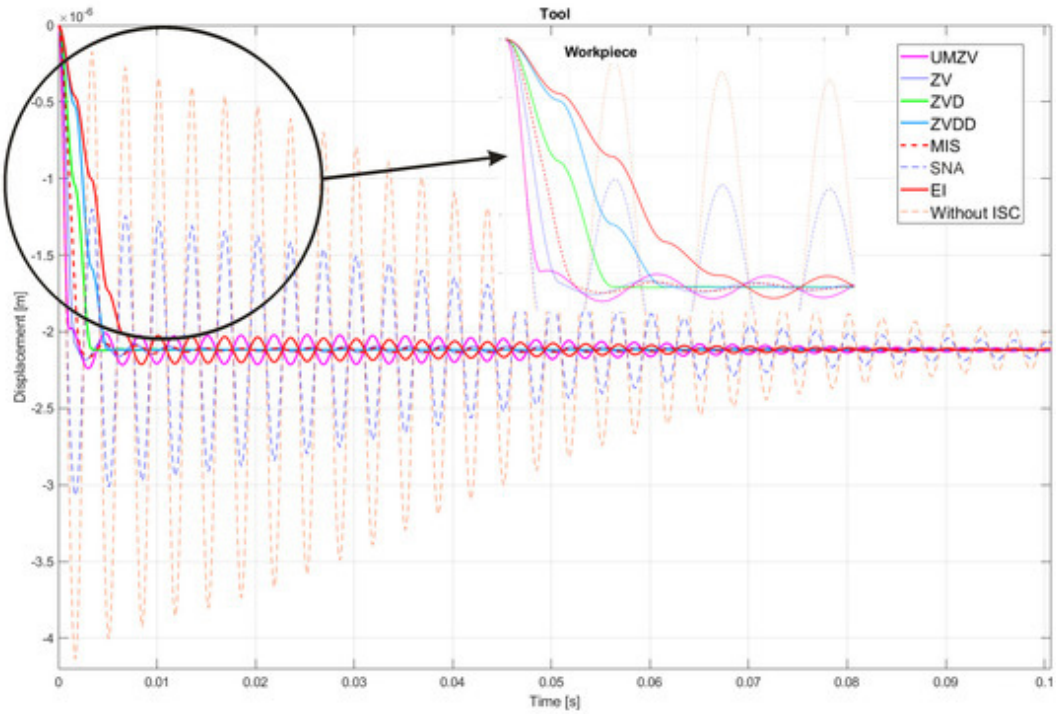
ตัวควบคุม PID สำหรับวงรอบตำแหน่งมีโครงสร้างเดียวกับสมการ (2.7) โดย Output ของ Outer Loop คือ Velocity Setpoint ที่ส่งเข้าสู่ Inner Loop ความถี่ Sampling กำหนดที่ $f_p = 100$ Hz ($T_p = 10$ ms)

หลักการออกแบบ Cascade

การออกแบบวงรอบทั้งสองต้องปฏิบัติตามกฎ Bandwidth Separation โดย Bandwidth ของ Inner Loop ต้องสูงกว่า Outer Loop อย่างน้อย 5 เท่า เพื่อให้ Inner Loop ตอบสนองเร็วพอที่จะถูกมองเป็น Unity Gain จาก Outer Loop การออกแบบในโครงงานนี้ใช้ Root Locus โดยออกแบบ Inner Loop ก่อน แล้วจึงใช้ Transfer Function ของ Closed-Loop Inner Loop เป็น Plant ที่แท้จริงสำหรับการออกแบบ Outer Loop

2.2.2 ZVD Input Shaping

Input Shaping เป็นเทคนิคการปรับ Command Signal ก่อนส่งเข้าระบบ โดยการ Convolve คำสั่งเดิมกับ Impulse Sequence ที่ออกแบบมาเพื่อยกเลิก การแกว่งของระบบที่ความถี่ธรรมชาติ ω_n



รูปที่ 2.3: ลักษณะของสัญญาณ ZVD Impulse Sequence (บน) และเปรียบเทียบการตอบสนองของระบบเพื่อลดการแกว่ง (ล่าง)

โครงสร้างของ ZVD Shaper

ZVD Shaper ใช้ Impulse สามอัน (A_1, t_1), (A_2, t_2), (A_3, t_3) โดยกำหนด $t_1 = 0$ และ Delay ที่เหลือคำนวณจากคาบ Damped Oscillation

$$t_2 = T_d = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}}, \qquad t_3 = 2T_d \tag{2.10}$$

แอมพลิจูดของ Impulse คำนวณจากเงื่อนไข Zero Residual Vibration, Zero Derivative ของ Residual Vibration และผลรวมขนาดยอดเท่ากับ 1

$$K = e^{-\zeta\pi/\sqrt{1-\zeta^2}}, \qquad A_1 = \frac{1}{(1+K)^2}, \quad A_2 = \frac{2K}{(1+K)^2}, \quad A_3 = \frac{K^2}{(1+K)^2} \tag{2.11}$$

ZVD มีความทนทานต่อความคลาดเคลื่อนของ ω_n สูงกว่า ZV (2-Impulse) โดยแลกกับ Delay เพิ่มขึ้น T_d ค่า ω_n และ ζ ของ Connection Rod วัดจาก Hardware และพารามิเตอร์ ZVD ที่ได้แสดงในตารางที่ 3.24

การ Implementation บน Embedded System (Discrete-Time)

สำหรับการทำงานจริงบนไมโครคอนโทรลเลอร์ (STM32G474RE) ระบบทำงานในรูปแบบ Discrete-time ตามความถี่ของ Position Loop ($T_p = 10$ ms) ดังนั้นเวลาหน่วง t_2 และ t_3 ที่คำนวณได้ จะต้องถูกปัดเศษให้เป็นจำนวนเต็มของ Sample (N_d) เพื่อใช้งานใน Circular Buffer

$$N_2 = \text{round}\left(\frac{t_2}{T_p}\right), \qquad N_3 = \text{round}\left(\frac{t_3}{T_p}\right) \tag{2.12}$$

ส่งผลให้ Transfer Function ของ ZVD Shaper ในโดเมน z สามารถเขียนได้เป็น

$$F(z) = A_1 + A_2\, z^{-N_2} + A_3\, z^{-N_3} \tag{2.13}$$

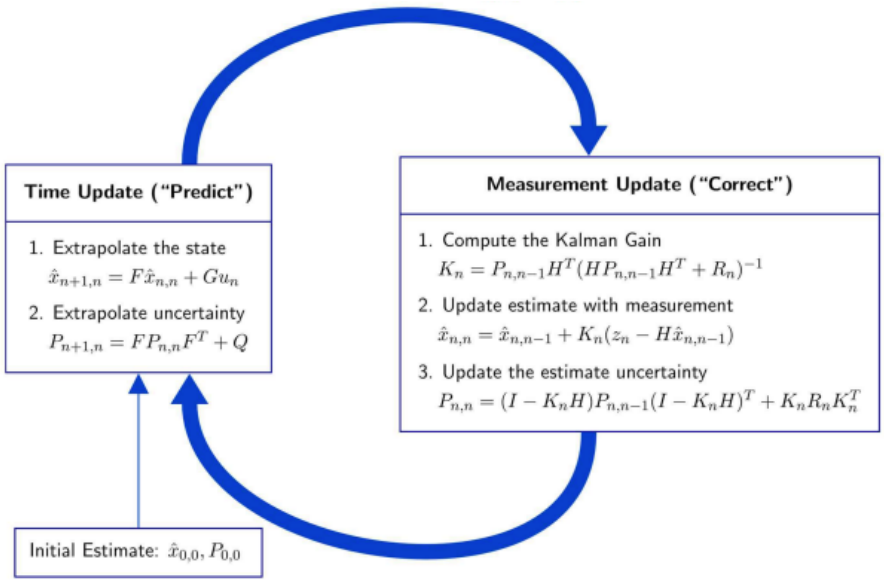
โดย Command Signal $r(t)$ จะถูกประมวลผลผ่านสมการสมการ (2.13) ซึ่งให้ผลเทียบเท่าการ Convolve ใน Discrete-time ก่อนส่งเข้า Position Outer Loop

ความสำคัญต่อ Cycle Time

หากไม่ใช้ Input Shaper ชิ้นงาน Connection Rod จะแกว่งหลังจากแขนกลหยุด โดย Settling Time ของการแกว่งดังกล่าวอยู่ที่ประมาณ 6--7 วินาทีต่อการเคลื่อนที่หนึ่งครั้ง ซึ่งทำให้ Cycle Time ทั้งหมดเกิน 35 วินาทีอย่างแน่นอน ZVD Shaper แก้ปัญหานี้โดยเพิ่มเวลาหน่วงรวมเพียง $t_3 \approx 80$ ms ต่อการเคลื่อนที่

2.2.3 Kalman Filter

Kalman Filter เป็นตัวประมาณค่าสถานะ (State Estimator) ที่หาค่าเหมาะสมที่สุด (Optimal) ภายใต้สมมติฐานว่าสัญญาณรบกวนกระบวนการ (Process Noise) และสัญญาณรบกวนการวัด (Measurement Noise) มีการกระจายตัวแบบ Gaussian White Noise โดยมี Computational Complexity เป็น $\mathcal{O}(n^2)$ ต่อ Time Step



รูปที่ 2.4: ขั้นตอนการทำงานของ Discrete-Time Kalman Filter ในแต่ละ Time Step

Discrete-Time State-Space Model และ Random Walk

หนึ่งในความท้าทายหลักของการควบคุมมอเตอร์คือ เซนเซอร์มักจะวัดได้เพียงตำแหน่งเชิงมุม (θ) การหาความเร็วเชิงมุมด้วยการหาอนุพันธ์โดยประมาณ (Approximate Derivative, $\Delta\theta/\Delta t$) จะทำให้เกิด Noise Amplification อย่างรุนแรง นอกจากนี้ ระบบยังเผชิญกับแรงบิดรบกวน (τ_d) ที่ไม่ทราบค่า

เพื่อแก้ปัญหานี้ แรงบิดรบกวนจึงถูกนำมาต่อขยายในตัวแปรสถานะ (Augmented State) โดยใช้สมมติฐานว่าการเปลี่ยนแปลงของ τ_d เกิดจากสัญญาณรบกวนแบบสุ่ม (Random Walk) หรือ $\dot{\tau}_d = w_\tau$ สำหรับการนำไปใช้งานบน Embedded System ระบบ Continuous-Time ในสมการ (2.5) จะถูกแปลงเป็นเวลาไม่ต่อเนื่อง (Discretization) ด้วยวิธี Zero-Order Hold (ZOH) ที่ Sample Time T_s

$$\mathbf{x}_{k+1} = \mathbf{A}_d\,\mathbf{x}_k + \mathbf{B}_d\,u_k + \mathbf{w}_k, \qquad y_k = \mathbf{C}\,\mathbf{x}_k + v_k \tag{2.14}$$

โดยที่ $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ คือ Process Noise ของแต่ละสถานะ และ $v_k \sim \mathcal{N}(0, R)$ คือ Measurement Noise

การกำหนดพารามิเตอร์ Noise (Q และ R)

การกำหนดเมทริกซ์ความแปรปรวนเกี่ยวพัน (Covariance Matrices) มีผลโดยตรงต่อความแม่นยำของตัวกรอง:

- **Measurement Noise (R):** สามารถประเมินได้จากความละเอียด (Resolution) ของเซนเซอร์ เช่น หากใช้อุปกรณ์เข้ารหัสสัมบูรณ์ (Absolute Encoder) ขนาด 12-bit จะมีตำแหน่งแยกย่อย 4096 ค่าต่อรอบ (ความละเอียด $\Delta \approx 0.088°$) ความแปรปรวนของ Quantization Noise นี้สามารถจำลองด้วย Uniform Distribution ซึ่งมีค่าความแปรปรวน $R = \Delta^2/12$

- **Process Noise ($\mathbf{Q}$):** เป็นเมทริกซ์ที่ปรับจูนเพื่อสร้างสมดุลระหว่างความเชื่อมั่นในแบบจำลองทางคณิตศาสตร์กับความเชื่อมั่นในเซนเซอร์ โดยสมาชิกที่ตรงกับสถานะ τ_d จะเป็นตัวกำหนดความไว (Bandwidth) ในการประมาณค่าแรงบิดรบกวน

Kalman Filter Equations

กระบวนการทำงานของ Kalman Filter แบ่งเป็นสองขั้นตอนย่อยที่ทำสลับกันทุก Time Step ได้แก่

1. **Predict (Time Update):** เป็นการทำนายสถานะล่วงหน้าจากแบบจำลองทางคณิตศาสตร์

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}_d\,\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_d\,u_{k-1} \tag{2.15}$$

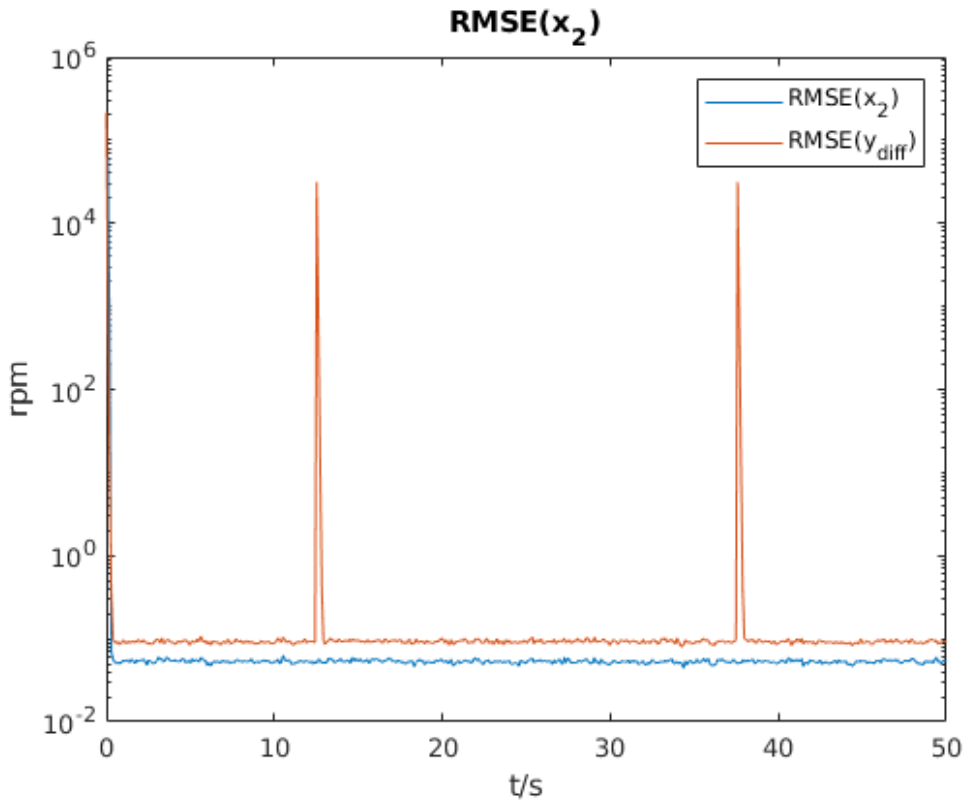
$$\mathbf{P}_{k|k-1} = \mathbf{A}_d\,\mathbf{P}_{k-1|k-1}\,\mathbf{A}_d^{\top} + \mathbf{Q} \tag{2.16}$$

2. **Update (Measurement Update):** เป็นการแก้ไขค่าที่ทำนายด้วยข้อมูลที่วัดได้จริงจากเซนเซอร์

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\,\mathbf{C}^{\top}\left(\mathbf{C}\,\mathbf{P}_{k|k-1}\,\mathbf{C}^{\top} + R\right)^{-1} \tag{2.17}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(y_k - \mathbf{C}\,\hat{\mathbf{x}}_{k|k-1}\right) \tag{2.18}$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\,\mathbf{C})\,\mathbf{P}_{k|k-1} \tag{2.19}$$



รูปที่ 2.5: การเปรียบเทียบประสิทธิภาพการประมาณค่าความเร็วเชิงมุมระหว่าง Kalman Filter กับการหาอนุพันธ์โดยประมาณ (Approximate Derivative)

Steady-State Kalman Gain

เนื่องจากระบบเชิงเส้นและ Time-Invariant Covariance Matrix $\mathbf{P}$ จะ Converge ไปยังค่า Steady-State $\mathbf{P}_\infty$ ซึ่งสามารถคำนวณล่วงหน้าได้จาก Discrete Algebraic Riccati Equation (DARE)

$$\mathbf{P}_\infty = \mathbf{A}_d \mathbf{P}_\infty \mathbf{A}_d^\top - \mathbf{A}_d \mathbf{P}_\infty \mathbf{C}^\top \left(\mathbf{C} \mathbf{P}_\infty \mathbf{C}^\top + R\right)^{-1} \mathbf{C} \mathbf{P}_\infty \mathbf{A}_d^\top + \mathbf{Q} \tag{2.20}$$

Steady-State Kalman Gain $\mathbf{K}_\infty$ คำนวณครั้งเดียวบน MATLAB แล้ว Hardcode ลงใน Firmware ทำให้ลด Computational Load บนหน่วยประมวลผลได้อย่างมาก โดยยังคงประสิทธิภาพการประเมินสถานะในระดับที่ใกล้เคียงกับ Optimal Gain

2.3. มาตรฐานการออกแบบเชิงกล

2.3.1 การออกแบบเพลา (ASME Shaft Design)

การออกแบบเพลาในโครงงานนี้อ้างอิงมาตรฐาน ASME โดยใช้ทฤษฎีพลังงานความผิดเพี้ยน (Distortion Energy Theory หรือ Von Mises Criterion) สำหรับกรณีที่เพลาต้องรับภาระทั้งโมเมนต์ดัด (Bending Moment) และแรงบิด (Torque) ไปพร้อมกัน

การวิเคราะห์ความล้า (Fatigue Analysis)

ในการทำงานของระบบขับเคลื่อน เพลาจะรับภาระแบบเปลี่ยนแปลงตามเวลา (Fluctuating Loads) ขนาดเส้นผ่านศูนย์กลางเพลาขั้นต่ำสำหรับการป้องกันความเสียหายจากความล้าคำนวณได้จากสมการ

$$d \geq \left[\frac{16}{\pi} \cdot \frac{N_s}{S_e} \sqrt{(K_f M_a)^2 + \frac{3}{4} (K_{fs} T_a)^2} \right]^{1/3} \tag{2.21}$$

โดยที่ N_s คือค่าความปลอดภัย (Safety Factor), K_f และ K_{fs} คือตัวคูณความเค้นหนาแน่นสำหรับความล้า (Fatigue Stress Concentration Factor) ของโมเมนต์ดัดและแรงบิดตามลำดับ, M_a คือแอมพลิจูดของโมเมนต์ดัด ($\mathrm{N\,m}$) และ T_a คือแอมพลิจูดของแรงบิด ($\mathrm{N\,m}$)

ค่า S_e คือขีดจำกัดความทนทาน (Modified Endurance Limit) (Pa) ซึ่งได้จากการปรับลดทอนค่าขีดจำกัดความทนทานในอุดมคติ (S'_e) ด้วยปัจจัยอิทธิพลต่างๆ ตามสมการของ Marin (Marin Equation) ดังนี้

$$S_e = k_a\, k_b\, k_c\, k_d\, k_e\, S'_e \tag{2.22}$$

โดยที่ตัวแปร k ต่างๆ คือปัจจัยปรับแก้ด้านสภาพผิว (Surface), ขนาด (Size), ภาระ (Load), อุณหภูมิ (Temperature) และความน่าเชื่อถือ (Reliability) ตามลำดับ

การตรวจสอบการครากสถิต (Static Yielding Check)

นอกจากการวิเคราะห์ความล้าแล้ว เพลาในระบบหุ่นยนต์ที่ต้องทนต่อแรงกระชาก (Jerk) หรือแรงบิดสูงสุดชั่วขณะ (Peak Torque) จากมอเตอร์ จำเป็นต้องได้รับการตรวจสอบไม่ให้เกิดการคราก (Yielding) ในรอบแรกของการทำงาน โดยความเค้นเทียบเท่า (Von Mises Equivalent Stress) สูงสุดจะต้องไม่เกินจุดคราก (S_y)

$$\sigma'_{\max} = \frac{16}{\pi d^3} \sqrt{(M_m + M_a)^2 + \frac{3}{4} (T_m + T_a)^2} \leq \frac{S_y}{N_y} \tag{2.23}$$

โดยที่ M_m และ T_m คือค่าเฉลี่ย (Mean) ของโมเมนต์ดัดและแรงบิด และ N_y คือค่าความปลอดภัยต่อการคราก

2.3.2 การคำนวณอายุการใช้งานตลับลูกปืน (ISO 281)

เนื่องจากเพลาถูกรองรับด้วยตลับลูกปืน (Rolling Element Bearing) อายุการใช้งาน Nominal ของตลับลูกปืน (ความน่าเชื่อถือ 90%) จึงถูกคำนวณตามมาตรฐาน ISO 281 ในหน่วยล้านรอบการหมุน (Million Revolutions)

$$L_{10} = \left(\frac{C}{P}\right)^p \tag{2.24}$$

โดยที่ C คือพิกัดน้ำหนักบรรทุกพลวัตพื้นฐาน (Basic Dynamic Load Rating) (N) ซึ่งได้จากสเปกของตลับลูกปืน, p คือค่าคงที่อายุการใช้งาน ($p = 3$ สำหรับ Ball Bearing และ $p = 10/3$ สำหรับ Roller Bearing) และ P คือแรงลัพธ์พลวัตเทียบเท่า (Equivalent Dynamic Load) (N)

แรงลัพธ์พลวัตเทียบเท่า (Equivalent Dynamic Load)

ในการใช้งานจริง ตลับลูกปืนอาจต้องรับทั้งแรงในแนวรัศมี (Radial Load, F_r) จากน้ำหนักของชิ้นงานและแรงดึงสายพาน และแรงในแนวแกน (Axial Load, F_a) แรงทั้งสองจะถูกแปลงให้เป็นแรงเทียบเท่า P เดียวผ่านสมการ

$$P = XF_r + YF_a \tag{2.25}$$

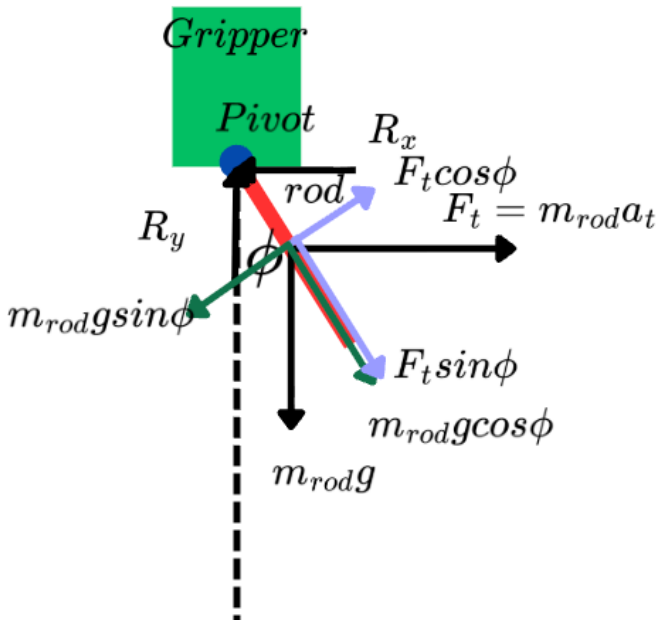
โดยที่ X คือตัวประกอบแรงแนวรัศมี (Radial Load Factor) และ Y คือตัวประกอบแรงแนวแกน (Axial Load Factor) ซึ่งพิจารณาจากอัตราส่วน F_a/F_r เทียบกับค่าพารามิเตอร์ e ของแคตตาล็อกตลับลูกปืน

เมื่อได้ค่า L_{10} ในหน่วยล้านรอบแล้ว สามารถแปลงกลับเป็นชั่วโมงการทำงานจริง (L_{10h}) ตามความเร็วรอบการหมุนเฉลี่ย n (rpm) ของข้อต่อหุ่นยนต์ได้ดังสมการ

$$L_{10h} = \frac{10^6}{60\,n}\,L_{10} \tag{2.26}$$

2.3.3 พลศาสตร์ของชิ้นงาน Connection Rod (Simple Pendulum Model)

Connection Rod ที่ห้อยอยู่ที่ปลายแขนกลสามารถจำลองเป็น Simple Pendulum โดยมีมวล m_{rod} กระจุกอยู่ที่ระยะ L_{rod} จากจุดหมุน เมื่อแขนกลมีความเร่งเชิงมุม α_{arm} จะเกิดแรงเฉื่อยกระตุ้นให้ Rod แกว่ง



รูปที่ 2.6: Free Body Diagram ของ Connection Rod จำลองเป็น Simple Pendulum ที่ถูกกระตุ้นด้วยความเร่งเชิงมุมของแขนกล

สมการการเคลื่อนที่

กำหนดให้ ϕ คือมุมเบี่ยงเบนของ Rod จากแนวดิ่ง (rad) ในกรณี Small-Angle Approximation ($\sin\phi \approx \phi$) สมการการเคลื่อนที่ของ Rod ในระบบอ้างอิงที่หมุนตามแขนกลได้จาก

$$\ddot{\phi} + 2\zeta_{rod}\,\omega_{n,rod}\,\dot{\phi} + \omega_{n,rod}^2\,\phi = \Gamma\,\alpha_{arm} \tag{2.27}$$

โดยที่ Γ คือ Coupling Factor ระหว่างความเร่งแขนกลและการกระตุ้น Rod และ $\alpha_{arm} = \ddot{\theta}_{arm}$ คือความเร่งเชิงมุมของแขนกล (rad/s²)

การ Derive ความถี่ธรรมชาติ

สำหรับ Rod ที่มีการกระจายมวลสม่ำเสมอตลอดความยาว L_{rod} โมเมนต์ความเฉื่อยรอบจุดหมุนปลายคือ $I = \frac{1}{3} m_{rod} L_{rod}^2$ และ Restoring Torque คือ $\tau = -m_{rod} g \frac{L_{rod}}{2} \sin\phi$ จากสมการ $I\ddot{\phi} = \tau$ และ Small-Angle Approximation ได้

$$\omega_{n,rod} = \sqrt{\frac{m_{rod} g \frac{L_{rod}}{2}}{\frac{1}{3} m_{rod} L_{rod}^2}} = \sqrt{\frac{3g}{2L_{rod}}} \tag{2.28}$$

สำหรับ Connection Rod ในโครงงานนี้ที่มีความยาว $L_{rod} = 100$ mm

$$\omega_{n,rod} = \sqrt{\frac{3 \times 9.81}{2 \times 0.10}} = \sqrt{147.15} \approx 12.13\,\text{rad/s} \quad (f_n \approx 1.93\,\text{Hz}) \tag{2.29}$$

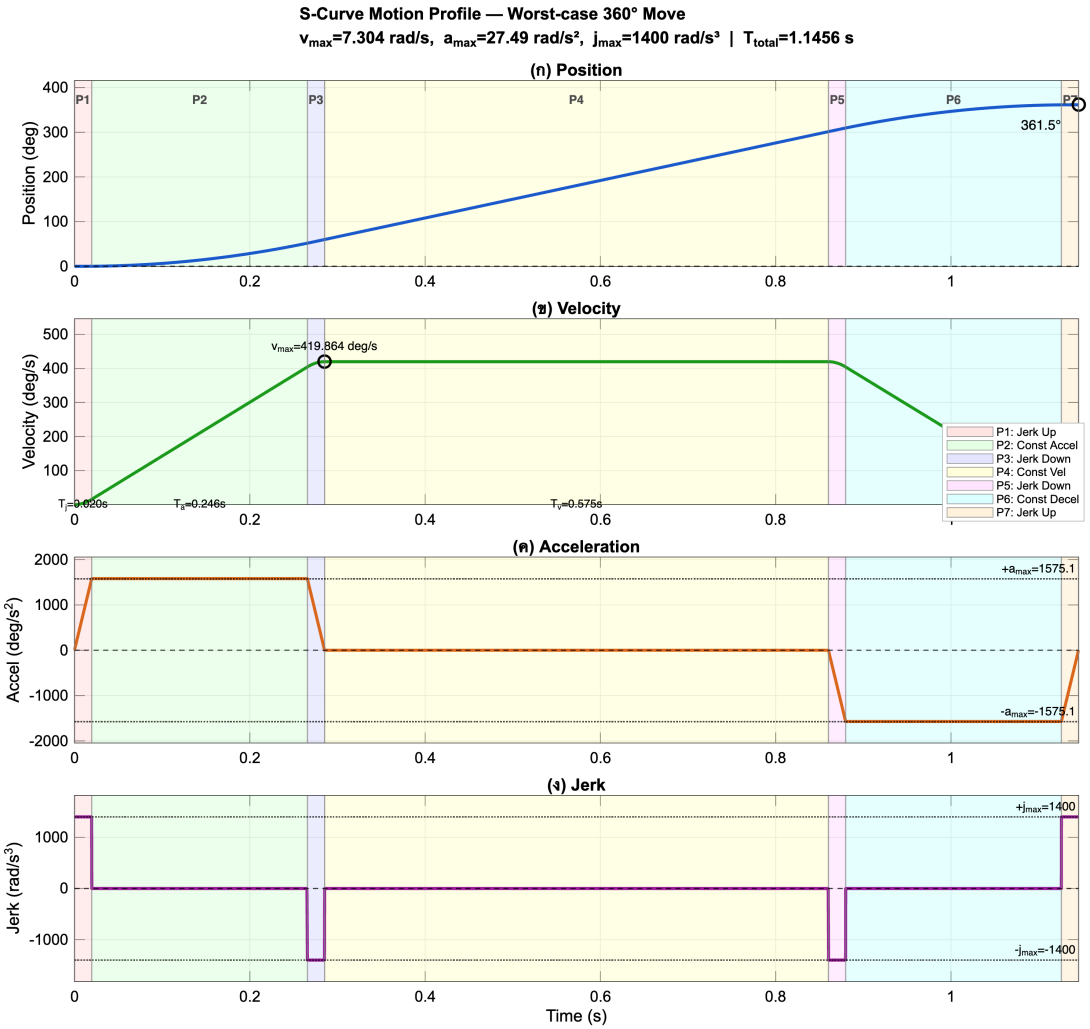
ความสำคัญต่อการออกแบบระบบควบคุม

สมการ (2.29) แสดงว่า Rod มีความถี่ธรรมชาติต่ำมาก ($\approx$ 2 Hz) เมื่อเทียบกับ Bandwidth ของ Position Loop (100 Hz) ดังนั้น Position PID ไม่สามารถ Damp การแกว่งของ Rod ได้ เพราะ Rod ไม่ได้อยู่ใน Feedback Loop ของ Controller การแก้ปัญหาจึงต้องใช้ ZVD Input Shaper ที่ออกแบบให้ยกเลิกพลังงานที่ความถี่ $\omega_{n,rod}$ โดยตรงตามที่อธิบายในหัวข้อ 2.2.2

2.4. อัลกอริทึมสร้างโปรไฟล์การเคลื่อนที่

2.4.1 S-Curve Trajectory (Jerk-Limited)

S-Curve Trajectory หรือ Jerk-Limited Trajectory เป็นโปรไฟล์การเคลื่อนที่ที่จำกัดอัตราการเปลี่ยนแปลงของความเร่ง (Jerk, $j = d^3\theta/dt^3$) ให้ไม่เกินค่าสูงสุด $j_{\max}$ ซึ่งช่วยลดแรงกระชากทางกล (Mechanical Shock) และจำกัดการกระตุ้นความถี่ธรรมชาติของระบบ เมื่อเทียบกับ Trapezoidal Profile ที่มีค่า Jerk เป็นอนันต์ ณ จุดที่เปลี่ยนความเร่ง



รูปที่ 2.7: โปรไฟล์การเคลื่อนที่แบบ S-Curve (Jerk-Limited) แสดงความสัมพันธ์เชิงจลนศาสตร์ทั้ง 7 เฟส

โครงสร้างเชิงเวลา (Time Intervals) ของ 7 เฟส

การสร้าง S-Curve แบบสมมาตร (Symmetric Profile) สำหรับระยะทางเป้าหมาย $\Delta\theta$ จะประกอบด้วย 7 เฟส โดยมีสมการคำนวณระยะเวลาในแต่ละช่วง (T_j, T_a, T_v) เบื้องต้นดังนี้

$$T_j = \frac{a_{\max}}{j_{\max}} \tag{2.30}$$

$$T_a = \frac{v_{\max}}{a_{\max}} - T_j \tag{2.31}$$

$$T_v = \frac{\Delta\theta}{v_{\max}} - (T_a + 2T_j) \tag{2.32}$$

โดยที่:

- **เฟส 1, 3, 5, 7 (Jerk Phases, T_j):** ช่วงที่ Jerk มีค่า $\pm j_{\max}$ เพื่อไต่ระดับหรือลดระดับความเร่ง

- **เฟส 2, 6 (Acceleration Phases, T_a):** ช่วงที่ความเร่งคงที่ $\pm a_{\max}$
- **เฟส 4 (Velocity Phase, T_v):** ช่วงที่ความเร็วคงที่ $v_{\max}$

การพิจารณาสภาวะโปรไฟล์ไม่สมบูรณ์ (Partial Profiles)

ในทางปฏิบัติ ระยะทาง $\Delta\theta$ อาจสั้นเกินกว่าที่ระบบจะเร่งไปถึง $v_{\max}$ หรือ $a_{\max}$ ได้ อัลกอริทึมจึงต้องตรวจสอบเงื่อนไขและคำนวณพารามิเตอร์ยอดแหลม (Peak Values) ใหม่ ดังนี้

- **กรณีไม่ถึงความเร่งสูงสุด ($T_a \leq 0$):** ระบบจะถูกตัดเฟส 2 และ 6 ออก ความเร่งสูงสุดใหม่ที่ทำได้จะถูกจำกัดด้วย Jerk คือ $a_{reach} = \sqrt{v_{\max} j_{\max}}$
- **กรณีไม่ถึงความเร็วสูงสุด ($T_v \leq 0$):** ระบบจะถูกตัดเฟส 4 ออก และต้องคำนวณ v_{reach} ใหม่จากสมการรากของระยะทางที่กำหนด

ความสัมพันธ์กับ Cycle Time และ ZVD Shaper

การกำหนดค่าพารามิเตอร์ต้องสอดคล้องกับข้อกำหนดของโครงงาน กล่าวคือ $v_{\max} \leq 5.84$ rad/s (P3) และ $a_{\max} \leq 22.0$ rad/s^2 (P4) เพื่อให้ Cycle Time รวมไม่เกิน 35 วินาที (P1) ทั้งนี้ การใช้ S-Curve ทำหน้าที่เสมือน Pre-filter ที่ช่วยลดแอมพลิจูดของการแกว่งตั้งแต่ต้นทาง ส่งผลให้ ZVD Input Shaper ในหัวข้อ 2.2.2 รับภาระในการหักล้างการสั่นน้อยลง และลดโอกาสที่ Command จะทะลุขีดจำกัด Saturation ของวงจรควบคุมความเร็ว

2.5. การระบุเอกลักษณ์ของระบบ (System Identification)

การนำแบบจำลอง State-Space ไปใช้งานจริงในตัวควบคุมและตัวประมาณค่าสถานะ (Kalman Filter) จำเป็นต้องทราบค่าพารามิเตอร์ทางกายภาพที่แม่นยำ ในโครงงานนี้ใช้แนวทาง Grey-Box Modeling โดยยึดโครงสร้างสมการทางฟิสิกส์เป็นหลัก แล้วหาค่าพารามิเตอร์จากการทดลองจริง

2.5.1 การประมาณค่าพารามิเตอร์ทางไฟฟ้า (Stalled Rotor Test)

เมื่อทำการล็อกเพลามอเตอร์ไม่ให้หมุน ($\omega = 0$) แรงเคลื่อนไฟฟ้าย้อนกลับ e_b จะมีค่าเป็นศูนย์ ในการทดลองจะต่อตัวต้านทานชันต์ (Shunt Resistor, R_{sh}) อนุกรมเข้ากับมอเตอร์เพื่อใช้วัดกระแสทางอ้อมผ่านแรงดันตกคร่อม (V_{sh}) ตามกฎแรงดันของ Kirchhoff (KVL) จะได้สมการวงจรดังนี้

$$V_{in} = i(t)\left(R_a + R_{sh}\right) + L_a \frac{di(t)}{dt} \tag{2.33}$$

โดยที่ V_{in} คือแรงดันแบบขั้นบันได (Step Voltage) ที่ป้อนเข้าสู่วงจร

การหาค่าความต้านทานอาร์เมเจอร์ (R_a)

เมื่อเวลาผ่านไปจนเข้าสู่สภาวะคงตัว (Steady State, $t \rightarrow \infty$) อัตราการเปลี่ยนแปลงของกระแสจะมีค่าเป็นศูนย์ ($di/dt = 0$) และกระแสไหลสูงสุด (i_{ss}) ซึ่งสามารถหาได้จากแรงดันที่ตกคร่อม Shunt Resistor ในสภาวะคงตัว (V_{sh})

$$V_{in} = i_{ss}\left(R_a + R_{sh}\right) \quad \text{โดยที่} \quad i_{ss} = \frac{V_{sh}}{R_{sh}} \tag{2.34}$$

แทนค่า i_{ss} และจัดรูปสมการเพื่อหา R_a จะได้ความสัมพันธ์ดังนี้

$$V_{in} = \frac{V_{sh}}{R_{sh}}\left(R_a + R_{sh}\right) \implies \frac{V_{in}}{V_{sh}} = \frac{R_a}{R_{sh}} + 1 \tag{2.35}$$

$$R_a = R_{sh}\left(\frac{V_{in}}{V_{sh}} - 1\right) \tag{2.36}$$

การหาค่าความเหนี่ยวนำอาร์เมเจอร์ (L_a)

การตอบสนองของกระแสในวงจร RL อนุกรม จะมีลักษณะเป็น First-order System ค่าคงที่เวลา (Time Constant, τ) คือเวลาที่แรงดัน V_{sh} ไต่ระดับขึ้นไปถึง 63.2% ของค่าสูงสุดในสภาวะคงตัว โดยค่า τ ของวงจรนี้มีค่าเท่ากับ

$$\tau = \frac{L_a}{R_{total}} = \frac{L_a}{R_a + R_{sh}} \tag{2.37}$$

ดังนั้น $L_a = \tau(R_a + R_{sh})$ จากสภาวะคงตัวก่อนหน้า เราทราบว่า $(R_a + R_{sh}) = R_{sh}\frac{V_{in}}{V_{sh}}$ เมื่อนำมาแทนค่าจะได้สมการสำหรับคำนวณ L_a ที่สามารถหาได้จากข้อมูลที่วัดได้โดยตรง

$$L_a = R_{sh}\left(\tau\frac{V_{in}}{V_{sh}}\right) \tag{2.38}$$

2.5.2 การระบุเอกลักษณ์แบบกล่องเทาด้วย MATLAB (Grey-Box Identification)

แม้ว่าการหาพารามิเตอร์ด้วยวิธีล็อกเพลาและการหมุนตัวเปล่าจะให้ค่าประมาณการเบื้องต้น (Initial Guess) ที่อ้างอิงจากสมการทางฟิสิกส์ แต่ในสภาพแวดล้อมจริงระบบอาจมีความไม่เป็นเชิงเส้น (Non-linearities) เล็กน้อยแอบแฝงอยู่ โครงงานนี้จึงเลือกใช้ System Identification Toolbox ของซอฟต์แวร์ MATLAB เป็นเทคโนโลยีสำหรับปรับจูนค่าพารามิเตอร์ให้มีความแม่นยำสูงสุดผ่านกระบวนการหาค่าเหมาะสมที่สุด (Optimization)

วิธีพยากรณ์ความคลาดเคลื่อน (Prediction Error Method: PEM)

อัลกอริทึมของ MATLAB ทำงานโดยอาศัยหลักการลดรูปความคลาดเคลื่อนระหว่างผลตอบสนองของแบบจำลองคณิตศาสตร์ ($\hat{y}$) และข้อมูลที่วัดได้จริงจากเซนเซอร์ (y) โดยนิยามฟังก์ชันต้นทุน (Cost Function) ในรูปแบบของค่าผลรวม

ความผิดพลาดกำลังสอง (Sum of Squared Errors) ดังสมการ

$$V_N(\mathbf{p}) = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k|\mathbf{p})|^2 \tag{2.39}$$

โดยที่ N คือจำนวนข้อมูลทั้งหมด (Samples) และ $\mathbf{p} = [R_a,\ L_a,\ K_t,\ J,\ B]^\top$ คือเวกเตอร์ของพารามิเตอร์อิสระ (Free Parameters) ที่ต้องการค้นหา

กระบวนการประมวลผลอัลกอริทึม

การประมาณค่าพารามิเตอร์แบบ Grey-Box บน MATLAB มีกลไกการทำงาน 3 ขั้นตอนหลัก ดังนี้

1. **การสร้างแบบจำลอง (Model Definition):** กำหนดโครงสร้างเมทริกซ์ $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ ของสมการ State-Space แบบ Continuous-Time ผ่านฟังก์ชันโครงสร้าง `idgrey` โดยป้อนค่าประมาณการเบื้องต้นที่ได้จากการคำนวณเชิงวิเคราะห์
2. **การหาค่าเหมาะสมที่สุด (Parameter Estimation):** ป้อนข้อมูลอินพุต (แรงดัน V_a) และเอาต์พุต (กระแส i_a และความเร็ว ω) ที่บันทึกจากการทดลองจริงเข้าสู่ฟังก์ชัน `greyest` ซึ่งอัลกอริทึมจะใช้วิธีการค้นหาแบบไม่เป็นเชิงเส้น (Non-linear Least Squares) เช่น Levenberg-Marquardt ในการอัปเดตเวกเตอร์ $\mathbf{p}$ ซ้ำๆ (Iteration) จนกว่าฟังก์ชัน $V_N(\mathbf{p})$ จะลู่เข้าสู่ค่าจุดต่ำสุด (Convergence)
3. **การตรวจสอบความถูกต้อง (Model Validation):** ประเมินความแม่นยำของแบบจำลองด้วยเปอร์เซ็นต์ความพอดี (Fit Percentage) ซึ่งคำนวณจากค่า Normalized Root Mean Square Error (NRMSE) ระหว่างกราฟผลตอบสนองจากการซิมูเลชัน (Simulated Output) และข้อมูลการวัดจริง (Measured Data)

2.5.3 Zero-Phase Filtering ด้วย `filtfilt`

ในงาน System Identification ข้อมูลที่บันทึกจาก Hardware มักมีสัญญาณรบกวนความถี่สูงที่ต้องกำจัดออกก่อนนำเข้า Estimator การกรองสัญญาณด้วย Digital Filter แบบทั่วไป (Causal Filter) แม้จะลด Noise ได้ แต่มีผลข้างเคียงที่สำคัญคือ **Phase Lag** ซึ่งส่งผลโดยตรงต่อความแม่นยำของพารามิเตอร์ ที่ประมาณได้

ปัญหาของ Causal Filter: Phase Lag

Causal Filter ประมวลผลข้อมูลจากอดีตไปปัจจุบันทิศทางเดียว ทำให้ Output มี Phase Lag สัมพันธ์กับ Input ที่ทุกความถี่ สำหรับ Butterworth Order n ที่ Cutoff ω_c Phase Response คือ

$$\angle H(j\omega) = -n \arctan\left(\frac{\omega}{\omega_c}\right) \tag{2.40}$$

ตัวอย่างเช่น Butterworth Order 4 ที่ $f_c = 10$ Hz มี Phase Lag ที่ $f = 1$ Hz เท่ากับ

$$\angle H(j2\pi \cdot 1) = -4\arctan\left(\frac{1}{10}\right) \approx -22.9^\circ \tag{2.41}$$

Phase Lag นี้ทำให้ข้อมูล Output (ω) ล่าช้ากว่า Input (V_a) ในชุดข้อมูลที่นำเข้า Estimator Estimator จะพยายามปรับค่า Time Constant ของแบบจำลอง ให้ตรงกับ Delay ที่เกิดจาก Filter แทนที่จะเป็น Delay จากระบบจริง ส่งผลให้พารามิเตอร์ที่ควบคุม Phase ของการตอบสนอง โดยเฉพาะ L_a และ J มี **Bias Error**

หลักการทำงานของ `filtfilt`

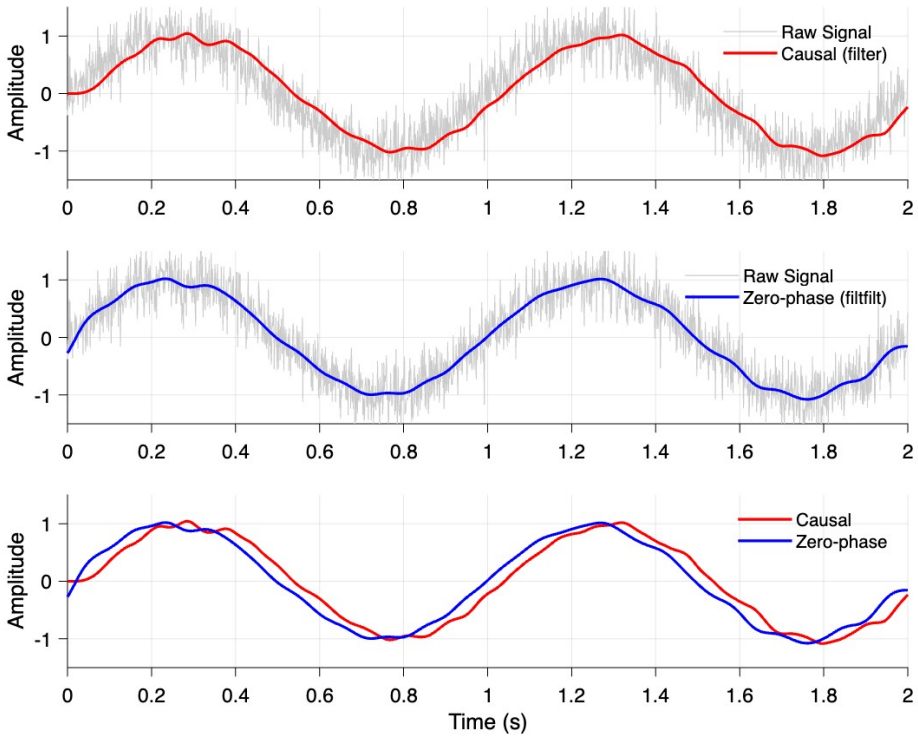
`filtfilt` แก้ปัญหา Phase Lag ด้วยการประมวลผลข้อมูล สองรอบในทิศทางตรงข้ามกัน ดังนี้

1. **Forward Pass:** กรองข้อมูลจากต้นจนจบตามปกติ Output มี Phase Lag $-\phi(\omega)$ และ Magnitude Response $|H(j\omega)|$
2. **Backward Pass:** Reverse ลำดับ Output จาก Forward Pass แล้วกรองอีกครั้ง ซึ่งเทียบเท่ากับการกรองสัญญาณที่มี Phase $+\phi(\omega)$ อยู่แล้ว ผลที่ได้จึงมี Phase Lag สุทธิ $-\phi(\omega) + \phi(\omega) = 0$
3. **Reverse กลับ:** Reverse ลำดับ Output จาก Backward Pass เพื่อให้ได้ข้อมูลที่เรียงลำดับถูกต้อง

ผลลัพธ์คือสัญญาณที่มี **Zero-Phase Response** ในทุกความถี่ แต่ Magnitude Response เทียบเท่ากับ Filter เดิมยกกำลังสอง

$$|H_{filtfilt}(j\omega)| = |H(j\omega)|^2 = \frac{1}{\sqrt{1 + (\omega/\omega_c)^{2n}}}^{\,2} = \frac{1}{1 + (\omega/\omega_c)^{2n}} \tag{2.42}$$

กล่าวคือ Butterworth Order 4 เมื่อใช้ `filtfilt` จะให้ Effective Order 8 และ Roll-off Rate -160 dB/decade ทำให้กำจัด Noise ได้รุนแรงขึ้นด้วย

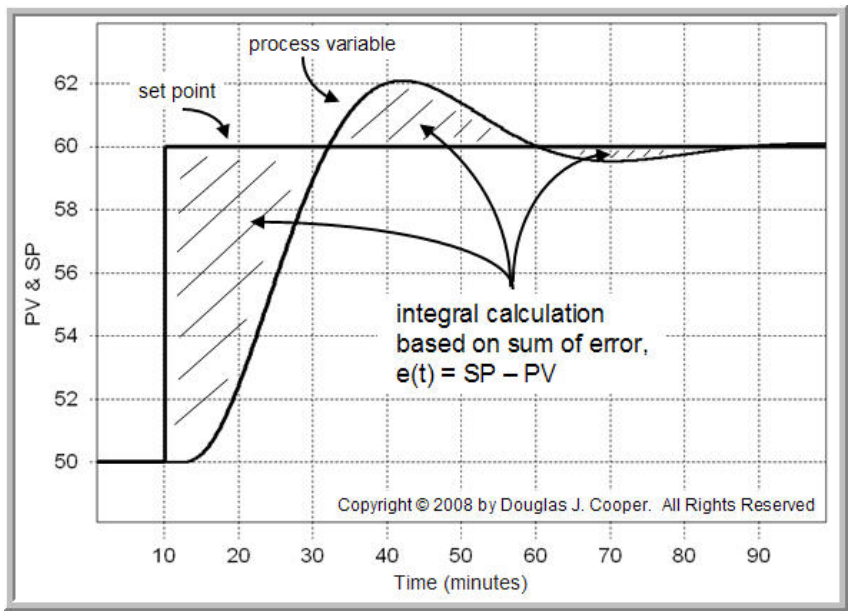


รูปที่ 2.8: เปรียบเทียบ Causal Filter (สีแดง) และ `filtfilt` Zero-Phase Filter (สีน้ำเงิน) บนสัญญาณเดียวกัน (บน) Causal Filter มี Phase Lag ที่สังเกตได้ (กลาง) `filtfilt` ทับ Peak ของสัญญาณต้นฉบับได้ตรงทุกจุด (ล่าง) Phase Difference ระหว่างทั้งสองวิธี

ข้อจำกัดของ `filtfilt`

เนื่องจากต้องใช้ข้อมูลทั้งชุดก่อนประมวลผล `filtfilt` จึงเป็น **Offline Filter** ไม่สามารถใช้งาน Real-time บน Embedded System ได้ ซึ่งเป็นที่ยอมรับได้ในบริบทของ System Identification ที่ข้อมูลทั้งหมดถูกบันทึกไว้ก่อนแล้ว สำหรับการใช้งานบน STM32G474RE ใน Closed-loop Control จะใช้ Kalman Filter เป็น State Estimator แบบ Real-time ตามที่อธิบายในหัวข้อ 2.2.3 แทนการกรองสัญญาณโดยตรง

2.6. ปรากฏการณ์อินทิกรัลไวนด์อัป (Integral Windup)



รูปที่ 2.9: การเปรียบเทียบการตอบสนองของระบบ (บน) สัญญาณควบคุม (กลาง) และการสะสมสถานะของตัวอินทิเกรต (ล่าง) ระหว่างตัวควบคุม PID มาตรฐานและตัวควบคุมที่มีการติดตั้งระบบ Anti-Windup แบบ Back-Calculation

ในระบบควบคุมจริง อุปกรณ์ขับกำลัง (เช่น H-Bridge) มีขีดจำกัดการจ่ายแรงดันสูงสุด (Saturation Limit, $u_{\max}$) หากระบบเกิดความคลาดเคลื่อนสะสมเป็นเวลานาน เทอมอินทิกรัล ($K_i \int e\,dt$) จะมีค่าเพิ่มขึ้นเรื่อยๆ จนเกินความสามารถของอุปกรณ์ ปรากฏการณ์นี้เรียกว่า Integral Windup ซึ่งส่งผลให้ระบบมีการตอบสนองที่ล่าช้าเมื่อความคลาดเคลื่อนเปลี่ยนทิศทาง และทำให้เกิด Overshoot ขนาดใหญ่

ดังที่ได้แสดงสมการการทำงานของระบบ Anti-Windup แบบ Back-Calculation ไว้ในสมการ (2.9) อัลกอริทึมจะป้อนกลับส่วนต่างระหว่างคำสั่งก่อนจำกัด (Unsaturated Command) และหลังจำกัด (Saturated Command) มาลดทอนสถานะของ Integrator แบบ Real-time เพื่อรักษาระบบให้อยู่ในขอบเขตเชิงเส้น

2.7. มาตรฐานตู้ควบคุมอุตสาหกรรม (Industrial Enclosures)

2.7.1 การจัดการสายไฟและลดสัญญาณรบกวน (Wiring & EMI Mitigation)

การออกแบบการจัดวางสายไฟอ้างอิงตามมาตรฐานความปลอดภัยของเครื่องจักร IEC 60204-1 โดยยึดรางอุปกรณ์มาตรฐาน (DIN 35mm) และมีหลักการจัดการคลื่นแม่เหล็กไฟฟ้าแทรกสอด (Electromagnetic Interference: EMI) ดังนี้

- **Cable Segregation:** สายไฟฟ้ากำลัง (Power Cables) และสายสัญญาณ (Signal Cables) จะต้องถูกแยกเดินในรางเก็บสาย (Wire Duct) ที่ต่างกัน โดยมีระยะห่างอย่างน้อย 50 มิลลิเมตร หากมีความจำเป็นต้องเดิน

สายพาดผ่านกัน จะต้องจัดวางให้ตัดกันแบบตั้งฉาก (90°) เพื่อลดค่าความเหนี่ยวนำร่วม (Mutual Inductance) ให้เหลือน้อยที่สุด

- **Shielding & Grounding:** สายสัญญาณแอนะล็อกและสายสื่อสาร (เช่น CAN Bus) ต้องใช้สายคู่ตีเกลียวมีชิลด์ (Shielded Twisted Pair) และจุดต่อสายดิน (Grounding) ของระบบควบคุมทั้งหมดต้องเชื่อมต่อรวมกันแบบจุดเดียว (Star Grounding) เพื่อป้องกันการเกิดลูปของสายดิน (Ground Loop) ที่จะสร้างความต่างศักย์รบกวนสัญญาณเซนเซอร์

2.7.2 ความปลอดภัยทางไฟฟ้าและวงจรหยุดฉุกเฉิน (Electrical Safety & E-Stop)

ภายในตู้ควบคุมต้องติดตั้งอุปกรณ์ป้องกันกระแสเกิน (Overcurrent Protection) เช่น เบรกเกอร์ (MCB) หรือฟิวส์ ที่มีพิกัดสอดคล้องกับขนาดหน้าตัดของสายไฟ นอกจากนี้ หุ่นยนต์เคลื่อนที่อัตโนมัติต้องมีวงจรหยุดฉุกเฉิน (Emergency Stop) ระดับฮาร์ดแวร์ โดยสวิตช์ E-Stop ต้องต่ออนุกรมเพื่อตัดแหล่งจ่ายไฟออกจากอุปกรณ์ขับกำลัง (Actuators) โดยตรง (Hard-wired) โดยไม่พึ่งพาการประมวลผลทางซอฟต์แวร์เพียงอย่างเดียว เพื่อรับประกันความปลอดภัยเมื่อไมโครคอนโทรลเลอร์เกิดการค้าง (System Hang)

2.8. การ ออกแบบ ระบบ ไฟฟ้า และ วงจร ป้องกัน (Electrical System and Protection Design)

การออกแบบฮาร์ดแวร์สำหรับหุ่นยนต์อัตโนมัติ ไม่เพียงแต่ต้องพิจารณาด้านประสิทธิภาพการประมวลผล แต่ยังต้องคำนึงถึงเสถียรภาพของระบบจ่ายกำลังและการป้องกันความเสียหายจากสภาวะผิดปกติทางไฟฟ้า โดยอ้างอิงหลักการออกแบบวงจรสำหรับระบบหุ่นยนต์ระดับอุตสาหกรรม ดังนี้

2.8.1 วงจรป้องกันความผิดปกติทางไฟฟ้า (Electrical Protection Circuits)

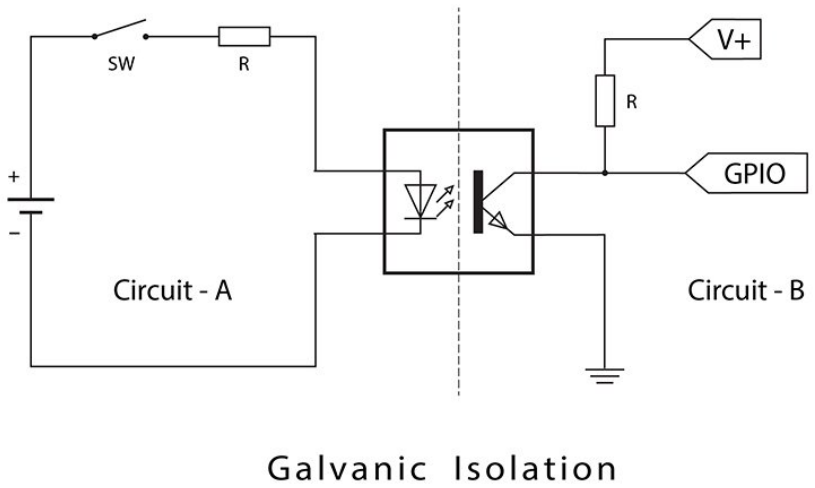
ในระบบที่ต้องจัดการกับกระแสไฟฟ้าสูงและอุปกรณ์ทางกล มีความเสี่ยงที่จะเกิดความผิดปกติทางไฟฟ้าซึ่งอาจทำลายหน่วยประมวลผลหลักได้ การออกแบบจึงต้องมีมาตรการป้องกันใน 3 ระดับ:

- **การป้องกันกระแสเกิน (Overcurrent Protection):** การใช้งานฟิวส์ (Fuses) หรือฟิวส์แบบตั้งค่าใหม่ได้ (PTC Resettable Fuses) ต่ออนุกรมในวงจรแหล่งจ่ายกำลัง เพื่อตัดวงจรเมื่อกระแสโหลดเกินพิกัด (Overload) หรือเกิดการลัดวงจร (Short Circuit) ป้องกันความร้อนสะสมที่อาจทำให้สายไฟหลอมละลายหรือเกิดเพลิงไหม้

- **การป้องกันการต่อสลับขั้ว (Reverse Polarity Protection):** การใช้ไดโอดกำลัง (Power Diode) แบบชอตต์กี (Schottky Diode) หรือวงจร P-Channel MOSFET เพื่อป้องกันกระแสไหลย้อนกลับเข้าไปทำลายแผงวงจรหลัก ในกรณีที่มีการต่อขั้วแบตเตอรี่หรือจุดเชื่อมต่อแหล่งจ่ายไฟสลับขั้ว

- **การป้องกันแรงดันกระชาก (Transient Voltage Suppression):** มอเตอร์กระแสตรงมีคุณสมบัติเป็นโหลดทางเหนี่ยวนำ (Inductive Load) เมื่อมีการตัดต่อกระแสไฟฟ้าอย่างรวดเร็ว (เช่น สัญญาณ PWM) หรือเมื่อเกิดสภาวะหยุดกะทันหัน จะเกิดแรงดันเหนี่ยวนำย้อนกลับ (Inductive Kickback) การติดตั้งตัวต้านทานแรงดันกระชาก (TVS Diode) ขนานกับสายไฟกำลัง จะช่วยขริบแรงดันกระชาก (Voltage Clamping) ไม่ให้ทะลุขีดจำกัดแรงดันของอุปกรณ์เซมิคอนดักเตอร์

2.8.2 การแยกกัลวานิกทางแสง (Galvanic Isolation)



รูปที่ 2.10: หลักการแยกกัลวานิกทางแสง (Galvanic Isolation) เพื่อป้องกันสัญญาณรบกวนจากภาคขับกำลังไหลกลับเข้าสู่ภาคตรรกะ

ระบบควบคุมหุ่นยนต์จำเป็นต้องแยกสภาวะทางไฟฟ้าระหว่าง "วงจรลอจิกแรงดันต่ำ" (Low-Voltage Logic เช่น MCU 3.3V/5V) และ "วงจรขับกำลังแรงดันสูง" (High-Power Drive เช่น Motor 24V) ออกจากกันอย่างเด็ดขาด โครงงานนี้ใช้ ออปโตคัปเปลอร์ (Optocouplers หรือ Optoisolators) ในการส่งผ่านสัญญาณ PWM จากไมโครคอนโทรลเลอร์ไปยัง H-Bridge ผ่านแสงอินฟราเรด ซึ่งช่วยป้องกันไม่ให้สัญญาณรบกวน (Noise) และแรงดันกระชากจากฝั่งมอเตอร์ไหลกลับมาทำลายหน่วยประมวลผลหลัก แม้ว่าอุปกรณ์ขับกำลังจะเกิดความเสียหายรุนแรงก็ตาม

2.8.3 การจัดการสายไฟและหน้าสัมผัส (Wiring and Strain Relief)

ความต้านทานแฝงและความเค้นทางกลในระบบสายไฟ เป็นสาเหตุหลักที่ทำให้หุ่นยนต์ทำงานผิดพลาด

- **ขนาดพิกัดสายไฟ (Wire Gauge):** การเลือกขนาดหน้าตัดสายไฟตามมาตรฐาน AWG (American Wire Gauge) ต้องสอดคล้องกับพิกัดกระแสไฟฟ้าสูงสุด โดยสายกำลังของมอเตอร์ (Power Lines) ต้องมีค่า AWG ต่ำ (หน้าตัดใหญ่) เพื่อลดค่าความต้านทานและกำลังสูญเสียในรูปความร้อน (I^2R) ในขณะที่สายสัญญาณ (Signal Lines) สามารถใช้สาย AWG สูง (หน้าตัดเล็ก) เพื่อความยืดหยุ่นได้
- **จุดเชื่อมต่อและการลดความเค้น (Connectors & Strain Relief):** บริเวณจุดเชื่อมต่อระหว่างบอร์ด PCB และสายไฟภายนอก จะต้องใช้คอนเนกเตอร์ที่ทนกระแสสูงและมีการล็อกทางกล (Mechanical Lock) อย่างแน่นหนา รวมถึงการออกแบบตัวลดความเค้นสายไฟ (Strain Relief) เพื่อป้องกันการขยับตัวที่อาจทำให้จุดบัดกรีฉีกขาด (Solder Joint Fatigue) ระหว่างที่กลไกของหุ่นยนต์มีการเคลื่อนที่และสั่นสะเทือน

2.9. โพรโทคอลการสื่อสารข้อมูลอุตสาหกรรม

การแลกเปลี่ยนข้อมูลระหว่าง Node ประมวลผล ระบบ Edge AI และอุปกรณ์ระดับล่าง (Sensors/Actuators) จำเป็นต้องใช้โครงข่ายการสื่อสารที่มีความน่าเชื่อถือสูงและทนทานต่อสภาพแวดล้อมทางอุตสาหกรรมที่มีสัญญาณรบกวนทางแม่เหล็กไฟฟ้า (EMI) หนาแน่น

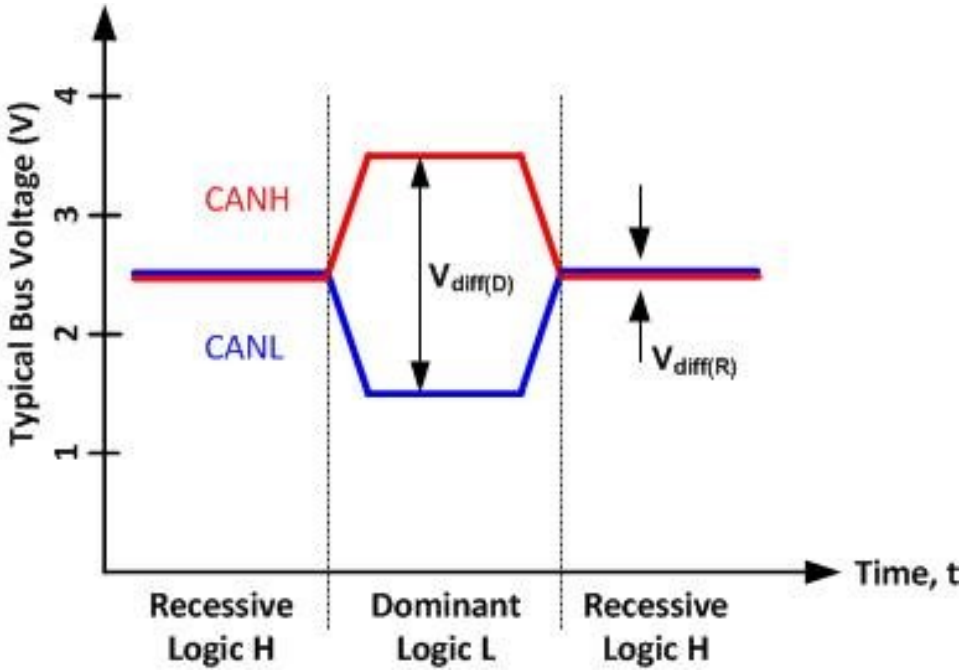
2.9.1 Controller Area Network (CAN Bus)

CAN Bus เป็นโพรโทคอลแบบอนุกรมที่ถูกออกแบบมาสำหรับระบบยานยนต์และหุ่นยนต์โดยเฉพาะ ตามมาตรฐาน ISO 11898

หลักการทำงานเชิงลึกและกลไกการเข้าถึงบัส

ชั้นกายภาพ (Physical Layer) ของ CAN ใช้สายคู่ตีเกลียวส่งสัญญาณแบบผลต่าง (Differential Signaling) ผ่านสาย CAN_H และ CAN_L โดยกำหนดสถานะลอจิกเป็นสองระดับ ได้แก่

- **Dominant State (ลอจิก '0'):** สาย CAN_H มีแรงดันสูงกว่า CAN_L ($V_{diff} \approx 2.0$V)
- **Recessive State (ลอจิก '1'):** สาย CAN_H และ CAN_L มีแรงดันเท่ากัน ($V_{diff} \approx 0$V)



รูปที่ 2.11: ลักษณะทางกายภาพของสัญญาณ CAN Bus แสดงผลต่างแรงดันในการกำหนดสถานะตรรกะแบบ Dominant และ Recessive

กลไกจัดการการชนกันของข้อมูลใช้เทคนิค **CSMA/CD+AMP** (Carrier Sense Multiple Access / Collision Detection with Arbitration on Message Priority) เมื่อมีหลาย Node พยายามส่งข้อมูลพร้อมกัน บัสจะ

ทำการเปรียบเทียบรหัสประจำตัว (Message ID) ทีละบิต (Bit-wise Arbitration) หาก Node ใดส่งสถานะ Recessive ('1') แต่ตรวจพบว่าบัสอยู่ในสถานะ Dominant ('0') จาก Node อื่น Node นั้นจะหยุดส่งข้อมูลทันทีและเปลี่ยนเป็นผู้รับ ส่งผลให้ Message ID ที่มีค่าน้อยที่สุดจะมีความสำคัญสูงสุด (Highest Priority) และสามารถยึดบัสได้โดยที่เฟรมข้อมูลไม่เกิดการสูญหาย

2.9.2 ชั้นกายภาพ RS-485 และโพรโทคอล Modbus RTU

ในขณะที่ CAN Bus เหมาะสำหรับระบบภายในตัวหุ่นยนต์ การสื่อสารระยะไกลกับอุปกรณ์อุตสาหกรรมภายนอก (เช่น เซนเซอร์ระยะทาง, PLC, หรืออุปกรณ์ในคลังสินค้า) มักอาศัยมาตรฐาน RS-485 ร่วมกับ Modbus RTU

หลักการทำงานของ UART และ RS-485

ไมโครคอนโทรลเลอร์ประมวลผลข้อมูลผ่านฮาร์ดแวร์ UART (Universal Asynchronous Receiver-Transmitter) ซึ่งจัดข้อมูลในรูปของเฟรม ประกอบด้วย Start Bit, Data Bits (8 บิต), Parity Bit (ทางเลือก) และ Stop Bit แต่เนื่องจากสัญญาณ UART เป็นแบบ Single-Ended (อ้างอิงกับกราวด์) จึงมีข้อจำกัดด้านระยะทางและอ่อนไหวต่อสัญญาณรบกวน

เพื่อแก้ปัญหานี้ สัญญาณ UART จะถูกแปลงผ่าน Transceiver ให้เข้าสู่มาตรฐาน **RS-485** ซึ่งส่งสัญญาณแบบผลต่าง (Differential) ตามสมการ:

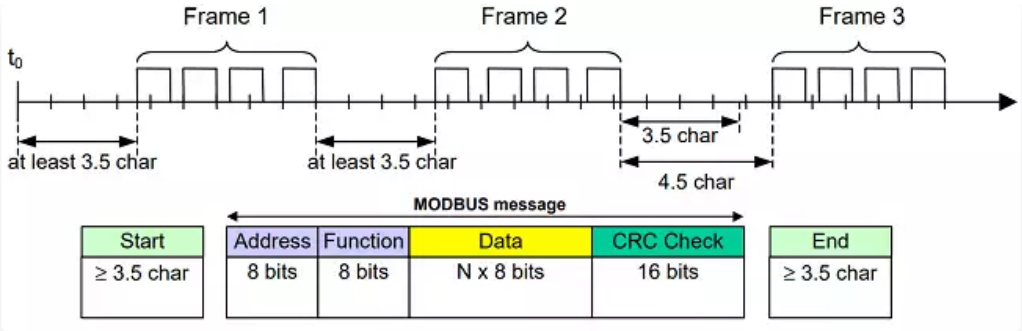
$$V_{diff} = V_A - V_B \tag{2.43}$$

หาก $V_A - V_B > +200$mV ตัวรับจะอ่านค่าเป็นลอจิก '1' และหาก $V_A - V_B < -200$mV จะอ่านเป็นลอจิก '0' การใช้ผลต่างแรงดันนี้ทำให้สามารถส่งข้อมูลได้ไกลสูงสุดถึง 1,200 เมตร และหักล้างสัญญาณรบกวนที่เหนี่ยวนำเข้ามาพร้อมกันทั้งสองเส้น (Common-Mode Noise Rejection)

กลไกของ Modbus RTU

Modbus RTU ทำงานในชั้น Application Layer บนโครงข่าย RS-485 ด้วยสถาปัตยกรรมแบบ **Master-Slave** โดยอุปกรณ์ Master เพียงตัวเดียวจะเป็นผู้สั่งการ (Polling) และ Slave จะตอบสนองเมื่อถูกเรียกตาม Address ของตนเท่านั้น ป้องกันปัญหาการชนกันของข้อมูลโดยเด็ดขาด

เฟรมข้อมูลของ Modbus RTU มีการต่อท้ายด้วยรหัส **CRC-16 (Cyclic Redundancy Check)** ซึ่งเป็นการหารพหุนาม (Polynomial Division) ระดับบิต เพื่อตรวจสอบความสมบูรณ์ของข้อมูลทั้งหมดในแพ็กเกจ หากฝั่งรับคำนวณ CRC ได้ไม่ตรงกับที่แนบมา ข้อมูลชุดนั้นจะถูกเพิกเฉยเพื่อป้องกันพฤติกรรมผิดพลาดของ Actuator



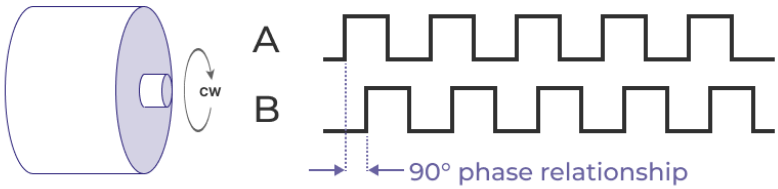
รูปที่ 2.12: โครงสร้างเฟรมข้อมูลของโพรโทคอล Modbus RTU แสดงการจัดเรียงไบต์ข้อมูลและรหัสตรวจสอบความผิดพลาด (CRC-16)

2.10. เซนเซอร์และอุปกรณ์ตรวจวัด

ระบบใช้เซนเซอร์สามประเภทที่ทำงานร่วมกัน ได้แก่ Encoder สำหรับวัดตำแหน่ง, Current Sensor สำหรับตรวจสอบความปลอดภัย และ Proximity Sensor สำหรับกำหนดจุด Home

2.10.1 Incremental Encoder (AMT-103)

Incremental Encoder วัดการเปลี่ยนแปลงตำแหน่งเชิงมุมโดยนับพัลส์ จาก Quadrature Signal สองช่อง (Channel A และ Channel B) ที่มีเฟสต่างกัน $90°$ ทำให้สามารถระบุทิศทางการหมุนได้



รูปที่ 2.13: Encoder AMT-103 และ Quadrature Signal แสดงการระบุทิศทางจากลำดับ Phase

Quadrature Encoding (×4)

การอ่านค่าแบบ ×4 (4-edge decoding) นับทั้ง Rising Edge และ Falling Edge ของทั้ง Channel A และ B ทำให้ Resolution เพิ่มขึ้น 4 เท่า

$$\text{Resolution} = \text{PPR} \times 4 = 2048 \times 4 = 8192 \text{ counts/rev} \tag{2.44}$$

ค่าตำแหน่งเชิงมุมแปลงจาก Encoder Count ได้จาก

$$\theta\ (\text{deg}) = \frac{N_{count}}{8192} \times 360^\circ \tag{2.45}$$

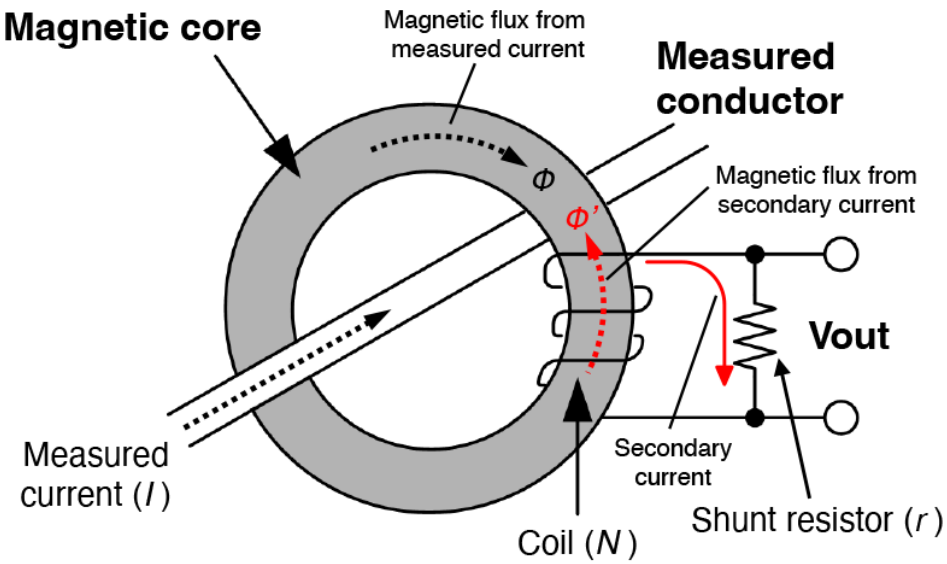
โดยที่ N_{count} คือค่า Counter ของ TIM3 (STM32 QEI Mode) Encoder ติดตั้งที่ Output Shaft โดยตรง (หลัง Gearbox และ Belt) ทำให้ค่าที่อ่านได้สะท้อนตำแหน่งแขนกลจริงโดยไม่มี Backlash Error ผ่านระบบส่งกำลัง

Index Pulse

AMT-103 มี Index Channel (Channel Z) ที่ให้พัลส์หนึ่งครั้งต่อรอบ ใช้สำหรับยืนยันตำแหน่ง Home ระหว่าง Homing Sequence

2.10.2 Current Sensor (WCS1800)

WCS1800 เป็น Hall-Effect Current Sensor วัดกระแสแบบ Non-contact โดยอาศัยสนามแม่เหล็กที่เกิดจากกระแสไฟฟ้าไหลผ่านตัวนำ แปลงเป็นแรงดันเอาต์พุตแบบ Linear



รูปที่ 2.14: WCS1800 Hall-Effect Current Sensor และลักษณะการตอบสนองเชิงเส้น

หลักการวัดกระแส

แรงดัน Output ของ WCS1800 มีความสัมพันธ์เชิงเส้นกับกระแส

$$V_{out} = V_{zero} + S \cdot I_a \tag{2.46}$$

โดยที่ V_{zero} คือแรงดันที่กระแสเป็นศูนย์ (ประมาณ 2.5 V ที่ $V_{cc} = 5$ V), $S = 0.040$ V/A คือ Sensitivity และ I_a

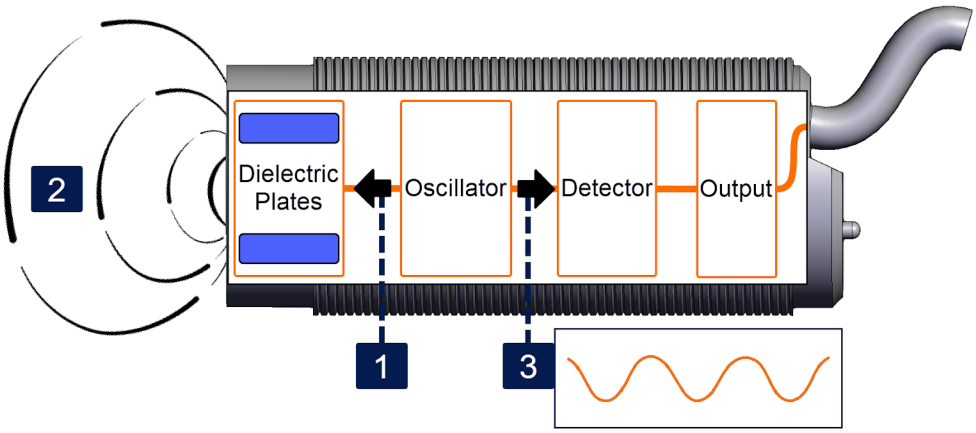
คือกระแสที่ต้องการวัด (A) ค่า V_{zero} ถูก Calibrate อัตโนมัติเมื่อ Boot โดยวัดเฉลี่ยจากตัวอย่าง 100 ค่าแรกในขณะที่มอเตอร์ยังไม่ทำงาน

การนำไปใช้ใน Safety Stack

ค่ากระแสที่อ่านได้ผ่านการกรองด้วย Exponential Moving Average (EMA) เพื่อลด ADC Noise ก่อนนำไปตรวจสอบใน Safety Monitor หากกระแสเกิน $I_{limit} = 2.0$ A ต่อเนื่องนานกว่า 100 ms ระบบจะเข้าสู่ `STATE_FAULT` (Fault Code 0x42)

2.10.3 Proximity Sensor (Inductive)

Proximity Sensor แบบ Inductive ตรวจจับวัตถุโลหะโดยไม่สัมผัส โดยอาศัยการเปลี่ยนแปลงของสนามแม่เหล็กไฟฟ้าเมื่อมีวัตถุเข้ามาในระยะตรวจจับ ใช้สำหรับระบุตำแหน่ง Home ใน Two-Stage Homing Sequence



รูปที่ 2.15: Proximity Sensor บนแขนกลและ Timing Diagram ของสัญญาณระหว่าง Two-Stage Homing

วงจรเอาต์พุตแบบ Open-Collector

Sensor ที่ใช้เป็นแบบ Open-Collector Optocoupler โดยมีสถานะดังนี้

- **ไม่มีวัตถุ (Normal):** Transistor นำกระแส ขา Output ถูก Pull ลง GND `LOW`
- **มีวัตถุ (Triggered):** Transistor หยุดนำกระแส ขา Output ลอย ถูก Pull-up ขึ้น 3.3 V ผ่าน Internal PULLUP ของ STM32 `HIGH`

ดังนั้น Active Condition คือ `GPIO_PIN_SET` (HIGH) และต้องตั้งค่า GPIO เป็น `GPIO_PULLUP` เสมอ การตั้งค่าเป็น `GPIO_PULLDOWN` จะทำให้ทั้งสองสถานะเป็น 0 V และ MCU ไม่สามารถตรวจจับสัญญาณได้

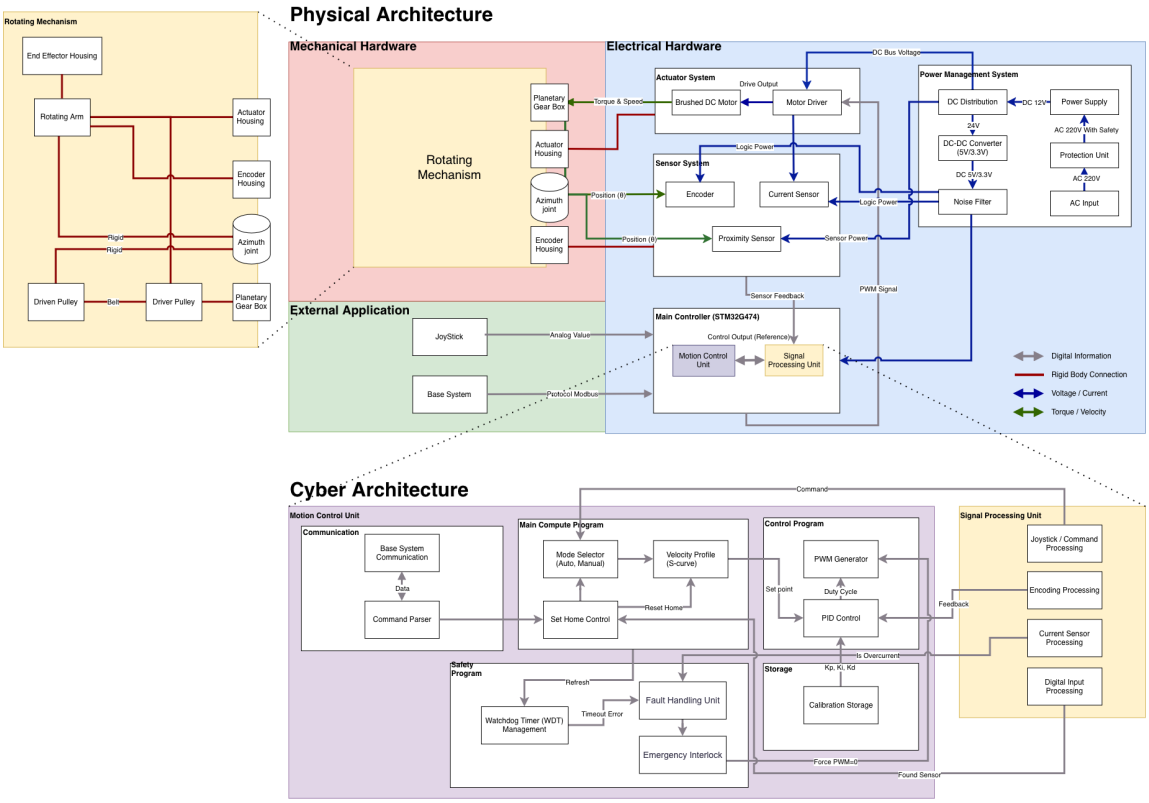
การนำไปใช้ใน Homing Sequence

Proximity Sensor ใช้ Debounce 1 tick (10 ms) ซึ่งสั้นกว่า Sensor อื่น เนื่องจาก Detection Zone แคบ ต้องการการตอบสนองที่รวดเร็วเพื่อจับ Edge ได้แม่นยำ ระบบ Homing ใช้หลักการ Midpoint Capture ระหว่าง Edge A และ Edge B ซึ่งมาจากทิศทางตรงกันข้ามที่ความเร็วเท่ากัน ทำให้ค่ากึ่งกลางที่ได้ไม่มี Bias จาก Sensor Switching Delay

บทที่ 3 สถาปัตยกรรมและการออกแบบระบบ

3.1. สถาปัตยกรรมระบบภาพรวม

ระบบ Circular Pick and Place Robot ออกแบบตามหลัก Cyber-Physical System โดยแบ่งสถาปัตยกรรมออกเป็น Physical Layer ซึ่งประกอบด้วยฮาร์ดแวร์จริง และ Cyber Layer ซึ่งประกอบด้วยโครงสร้าง Firmware และ Communication ภายใน STM32G474RE ดังแสดงในรูปที่ 3.1



รูปที่ 3.1: System Architecture ของระบบ Circular Pick and Place Robot แสดง Physical Architecture และ Cyber Architecture

3.1.1 ห่วงโซ่การส่งกำลัง (Power Transmission Chain)

ข้อกำหนด P3 กำหนดให้ความเร็วเชิงมุมที่เพลาแขนกลอยู่ในช่วง $2.9 \leq \omega \leq 5.84$ rad/s และ P4 กำหนดให้ความเร่งสูงสุด $\alpha_{max} \leq 22.0$ rad/s^2 ในขณะที่มอเตอร์ RS-775 มีความเร็วสูงสุด 5,950 RPM (≈ 623 rad/s) แสดงว่าระบบต้องการอัตราทดรวม ในช่วง $N \approx 70$ เพื่อแปลงความเร็วสูงของมอเตอร์ให้เป็น แรงบิดสูงสำหรับงานหยิบวางชิ้นงาน 2 kg ที่รัศมี 500 mm

การบรรลุอัตราทดดังกล่าวด้วย Single-stage Gearbox เพียงอย่างเดียว จะทำให้ขนาดของ Gearbox ใหญ่

เกินขีดจำกัดฐาน 200×200 mm (ข้อกำหนด M1) ระบบจึงออกแบบเป็น 2-stage โดยแยกหน้าที่ดังนี้

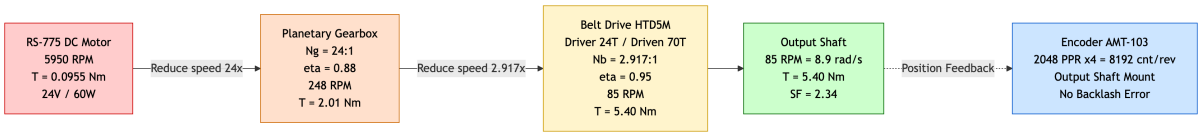
- **Planetary Gearbox** ($N_g = 24$): รับภาระ Torque Amplification หลัก เนื่องจาก Planetary configuration ให้อัตราทดสูงในขนาด Footprint เล็กกว่า Spur Gear ที่อัตราทดเทียบเท่า
- **Belt Drive HTD5M** ($N_b = 70/24 \approx 2.917$): ทำหน้าที่ปรับ Layout โดยส่งกำลังข้ามระยะ Center Distance 181.25 mm เพื่อแยกตำแหน่งมอเตอร์ออกจากจุดหมุนของแขนกล ซึ่งช่วยลดโมเมนต์ความเฉื่อยของส่วนที่หมุน และยังทำให้สามารถติดตั้ง Encoder ได้โดยตรง ที่ Output Shaft โดยไม่ผ่าน Gearbox

อัตราทดรวม $N = N_g \times N_b = 24 \times 2.917 = 70$ และ ประสิทธิภาพรวม $\eta = \eta_{gear} \times \eta_{belt} = 0.88 \times 0.95 = 0.836$ ให้แรงบิดที่เพลาแขนกล

$$T_{arm} = T_{motor} \times N \times \eta = 0.0955 \times 70 \times 0.836 = 5.40\,\mathrm{N \cdot m} \tag{3.1}$$

ซึ่งมี Safety Factor เทียบกับแรงบิดที่ต้องการ $T_{req} = J \cdot \alpha_{min} \times SF = 0.5764 \times 2.0 \times 2.0 = 2.31$ N·m คือ $SF_{actual} = 5.40/2.31 = 2.34 > 2.0$ **(ผ่าน P3, P4)**

การติดตั้ง Encoder ที่ Output Shaft: เพื่อลด Backlash Error ที่สะสมจากทั้ง Planetary Gearbox และ Belt Drive Encoder AMT-103 จึงติดตั้งโดยตรงที่ Output Shaft ซึ่งเป็นตำแหน่ง *หลัง* ระบบส่งกำลังทั้งหมด Control Loop จึงปิดผ่านตำแหน่ง จริงของแขนกลโดยตรง ทำให้ Backlash ในทุก Stage ไม่ส่งผลต่อ Closed-loop Accuracy และ Repeatability (A2)



รูปที่ 3.2: แผนผังห่วงโซ่การส่งกำลังแสดงการเปลี่ยนแปลงความเร็วและแรงบิด ตั้งแต่มอเตอร์ถึงแขนกล พร้อมตำแหน่ง Encoder ที่ Output Shaft

3.1.2 Application Finite State Machine

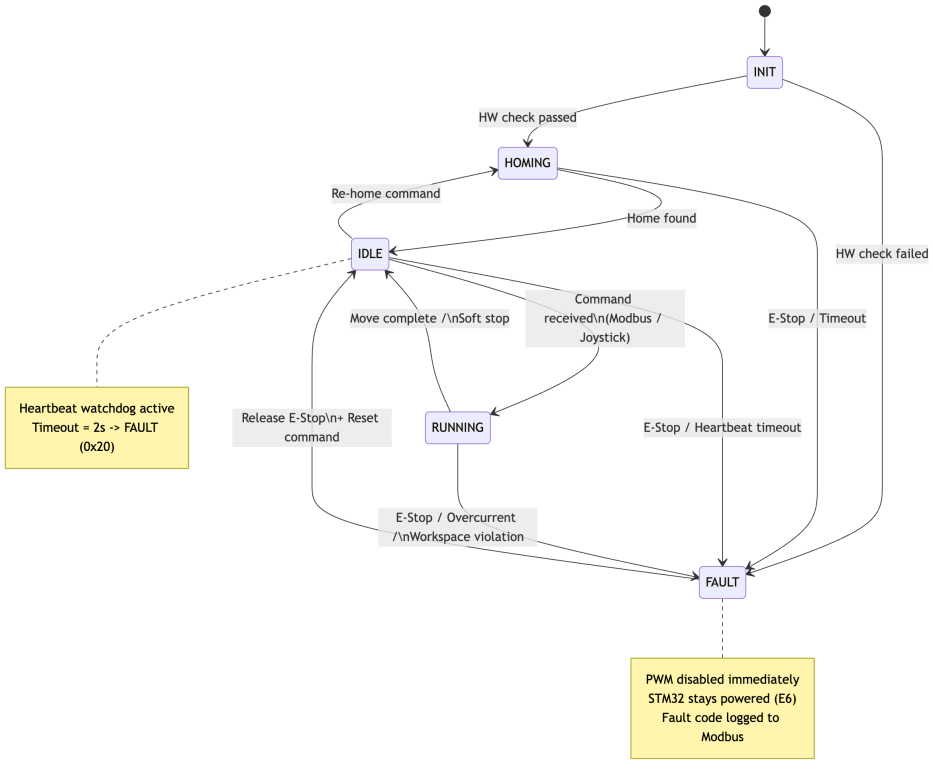
โครงสร้าง FSM ออกแบบให้ตอบสนองข้อกำหนดด้านความปลอดภัยและการใช้งาน พร้อมกันในเวลาเดียว ระบบมี 5 สถานะหลักดังนี้

- **`INIT`:** ตรวจสอบฮาร์ดแวร์และโหลดค่า Configuration จาก Flash ก่อนอนุญาตให้ระบบทำงาน
- **`HOMING`:** ค้นหาจุด Home ด้วย Two-Stage Homing (Coarse Search ไป Fine Search ด้วย Proximity Sensor) ตามข้อกำหนด A3 ที่ต้องสามารถตั้งจุด Home ได้หลังติดตั้ง
- **`IDLE`:** พร้อมรับคำสั่ง รอ Command จาก Modbus หรือ Joystick โดยตรวจสอบ Heartbeat Watchdog ต่อเนื่อง
- **`RUNNING`:** กำลังประมวลผล Motion Sequence ทั้ง Pick-Place Cycle และ Manual Jog

- **FAULT:** หยุดฉุกเฉิน ตัด PWM ทันที และ Log Fault Code เพื่อ Diagnosis

ทุกสถานะสามารถเข้าสู่ FAULT ได้จาก E-Stop Hardware หรือ Software Safety Fault (Overcurrent, Heartbeat Timeout, Workspace Violation) โดยไม่ผ่านสถานะกลาง เพื่อรับประกัน Worst-case Response Time ที่ขึ้นกับ TIM6 ISR Period ($\leq$ 1 ms) เท่านั้น

สถานะ FAULT ตัดเฉพาะ PWM Output (Motor Power) แต่ STM32 ยังคงรัน Firmware ต่อเนื่องเพื่อรับ Reset Command และรายงาน Fault Code ผ่าน Modbus ซึ่งเป็นไปตามข้อกำหนด E6 ที่กำหนดให้ Controller มีไฟเลี้ยงแม้หลังกด E-Stop สอดคล้องกับการออกแบบ Power Bus แบบ Dual-rail ในหัวข้อ 3.3.1

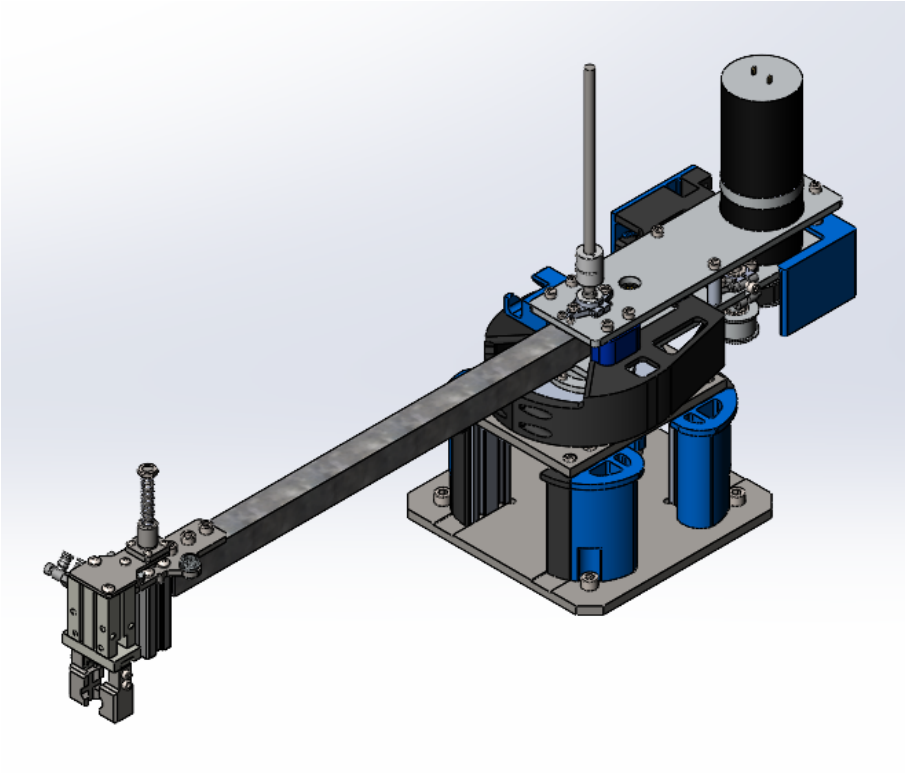


รูปที่ 3.3: Application FSM แสดง 5 สถานะหลักและเงื่อนไข Transition โดย FAULT สามารถถูก Trigger ได้จากทุกสถานะ (ลูกศรสีแดง)

3.2. การออกแบบเชิงกล

3.2.1 สถาปัตยกรรมเชิงกลโดยรวม

โครงสร้างเชิงกลของระบบแบ่งออกเป็น 3 ส่วนหลักที่ประกอบกัน ได้แก่ ส่วนฐาน (Base) ที่ยึดกับแท่นของลูกค้าด้วยสกรู M8 ตามข้อกำหนด M7, ส่วนแขนกล (Rotating Arm) รัศมี 500 mm ที่หมุนรอบแกนแนวตั้ง และชุดส่งกำลัง 2-stage ที่อธิบายในหัวข้อ 3.1.1 ภาพรวม CAD Assembly แสดงในรูปที่ 3.4



รูปที่ 3.4: CAD Assembly ของระบบเชิงกล แสดงส่วนประกอบหลักทั้งหมด

3.2.2 การคำนวณเพลา

การคำนวณขนาดเพลาดำเนินการตาม Maximum Shear Stress Theory (ASME Code) โดยมีจุดวิกฤตอยู่ที่ตำแหน่งลูกปืนบน ซึ่งรับทั้งโมเมนต์ดัดจากน้ำหนัก ปลายแขนและแรงดึงสายพานพร้อมกัน

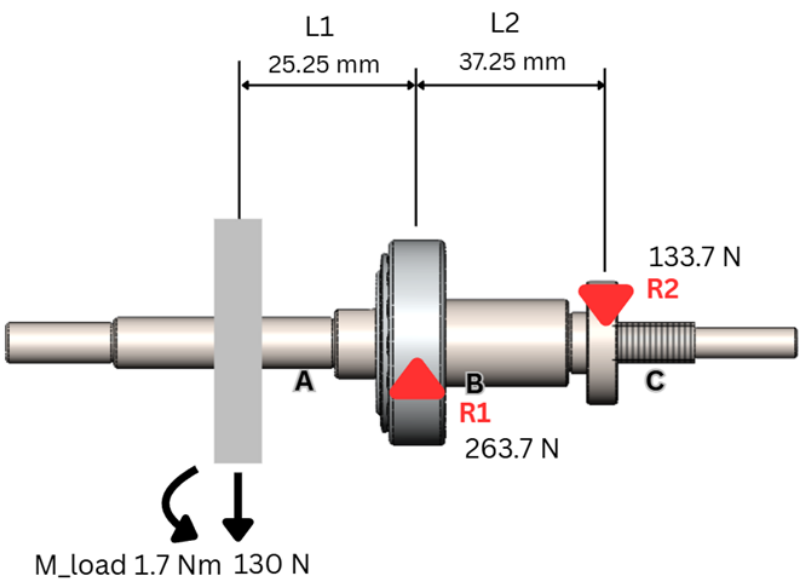
พารามิเตอร์และวัสดุ

ตารางที่ 3.1: พารามิเตอร์ภาระโหลดของระบบเพลา

พารามิเตอร์	สัญลักษณ์	ค่า	หน่วย
มวลโหลดที่ปลายเพลา	m	3.72	kg
น้ำหนักโหลดในแนวดิ่ง	W	37.20	N
ระยะเยื้องศูนย์	L_{cg}	46.63	mm
ความเร็วรอบ	n	85.00	RPM
แรงเหวี่ยงหนีศูนย์กลาง	F_c	13.75	N
แรงดึงสายพานรวม	F_{belt}	130.00	N
ระยะจากโหลดถึงลูกปืนบน	L_1	25.25	mm
ระยะลูกปืนบนถึงล่าง	L_2	37.25	mm

วัสดุเพลาเลือกใช้ Stainless Steel AISI 304 เนื่องจากเป็นวัสดุมาตรฐานที่หาซื้อได้ง่ายในตลาดไทย มี Yield Strength $S_{yt} = 215$ MPa เพียงพอสำหรับการรับภาระในระบบนี้ และ Allowable Shear Stress ตามมาตรฐาน ASME คือ

$$\tau_{allow} = 0.30 \times S_{yt} = 0.30 \times 215 = 64.5 \,\text{MPa} \tag{3.2}$$



รูปที่ 3.5: Free Body Diagram ของระบบเพลาแสดงแรงและโมเมนต์ที่จุดวิกฤต

การวิเคราะห์โมเมนต์ดัด

โมเมนต์ดัดจากโหลดปลายแขนและแรงเหวี่ยง

$$M_{overhang} = (W + F_c) \times L_1 = (37.20 + 13.75) \times 0.02525 = 1.287\,\mathrm{N \cdot m} \tag{3.3}$$

โมเมนต์ดัดจากแรงดึงสายพาน

$$M_{belt} = F_{belt} \times L_1 = 130.00 \times 0.02525 = 3.283\,\mathrm{N \cdot m} \tag{3.4}$$

โมเมนต์รวม Worst-case โดย Vector Superposition (ทิศทางเสริมกัน)

$$M_{total} = 5.014\,\mathrm{N \cdot m} \tag{3.5}$$

การคำนวณขนาดเพลา

เลือก Fatigue Shock Factors $C_m = 2.0$ (Bending) และ $C_t = 1.5$ (Torsion) สำหรับสภาวะ Light Shock เนื่องจาก S-curve Trajectory จำกัด Jerk ให้ไม่เกิน $j_{max} = 1400$ rad/s^3 ทำให้ภาระที่เพลา เปลี่ยนแปลงแบบต่อเนื่องไม่ใช่ Impact Load แต่ยังคง $C_m > 1$ เพื่อรองรับ Transient ที่อาจเกิดขณะ Emergency Stop

$$d_{min} = \left[\frac{16}{\pi\,\tau_{allow}}\sqrt{(C_m M_{total})^2 + (C_t T)^2}\right]^{1/3} \tag{3.6}$$

$$d_{min} = \left[\frac{16}{\pi \times 64.5 \times 10^6}\sqrt{(2.0 \times 5.014)^2 + (1.5 \times 5.398)^2}\right]^{1/3} = 10.59\,\mathrm{mm} \tag{3.7}$$

การออกแบบ Step Shaft

แม้ว่า $d_{min} = 10.59$ mm จากการวิเคราะห์ความล้า แต่ขนาดจริงที่จุดวิกฤตถูกกำหนดโดย Bearing Seat ของ SKF 30203 Tapered Roller Bearing ซึ่งมี ID = 17 mm ดังนั้นจึงเลือก $d_{actual} = 17$ mm ที่จุดวิกฤต ซึ่งให้ FOS ≥ 1.6 เทียบกับ d_{min}

เพลาออกแบบเป็น Step Shaft เพื่อรองรับ Bearing ที่ขนาดต่างกัน และสร้าง Shoulder สำหรับ Axial Location ดังสรุปในตารางที่ 3.2

ตารางที่ 3.2: สรุปผลการคำนวณเพลาและการออกแบบ Step Shaft

พารามิเตอร์	ค่า	หน่วย	หมายเหตุ
M_{total} (Worst-case)	5.014	N·m	Vector Superposition
T (Torsion)	5.398	N·m	จากระบบส่งกำลัง
d_{min} (คำนวณ)	10.59	mm	จุดวิกฤต (Fatigue)
d_{actual} (จุดวิกฤต)	17	mm	Bearing Seat SKF 30203
d (ช่วงกลาง)	12	mm	ผ่าน
d (ช่วงปลาย)	8--10	mm	ไม่ใช่จุดวิกฤต

3.2.3 การวิเคราะห์เพลาด้วย FEA (Shaft FEA)

เพื่อยืนยันผลการคำนวณเชิงวิเคราะห์ในหัวข้อ 3.2.2 ดำเนินการวิเคราะห์ Finite Element Analysis (Static Study) ด้วย SOLIDWORKS Simulation โดยจำลองเพลาเป็น Beam Element ซึ่งเหมาะสมกับโครงสร้างที่มี Aspect Ratio สูง (ความยาวรวม ≈ 68 mm เทียบกับเส้นผ่าศูนย์กลางสูงสุด 17 mm) การใช้ Beam Mesh แทน Solid Mesh ในกรณีนี้ให้ความแม่นยำเทียบเท่า แต่ใช้ Computational Cost ต่ำกว่าอย่างมีนัยสำคัญ เนื่องจากทฤษฎี Euler–Bernoulli Beam ถูกต้องเมื่อ $L/d > 10$ ซึ่งเป็นเงื่อนไขที่เพลานี้ตอบสนองได้

สภาวะการจำลอง (FEA Setup)

โมเดลเพลาแบบ Step Shaft สร้างขึ้นใน SOLIDWORKS โดยแบ่งออกเป็น 6 Beam Segment ตามขนาดเส้นผ่าศูนย์กลางที่เปลี่ยนไปในแต่ละช่วง วัสดุที่ใช้คือ Stainless Steel AISI 304 ซึ่งสอดคล้องกับการออกแบบ ในหัวข้อ 3.2.2

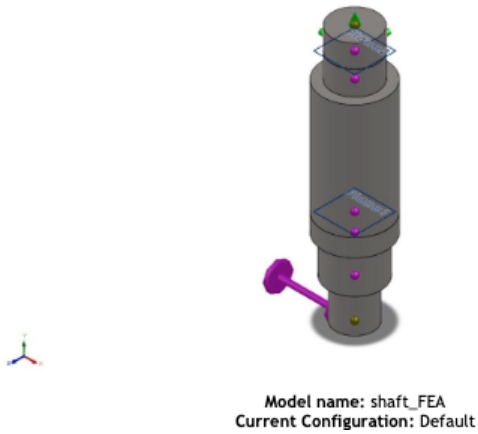
ตารางที่ 3.3: คุณสมบัติวัสดุที่ใช้ใน FEA เพลา (AISI 304)

คุณสมบัติ	ค่า	หน่วย
Yield Strength	2.068×10^8	N/m^2
Tensile Strength	5.170×10^8	N/m^2
Elastic Modulus	1.900×10^{11}	N/m^2
Poisson's Ratio	0.29	--
Mass Density	8,000	kg/m^3
Shear Modulus	7.500×10^{10}	N/m^2

กำหนด Boundary Conditions ดังนี้

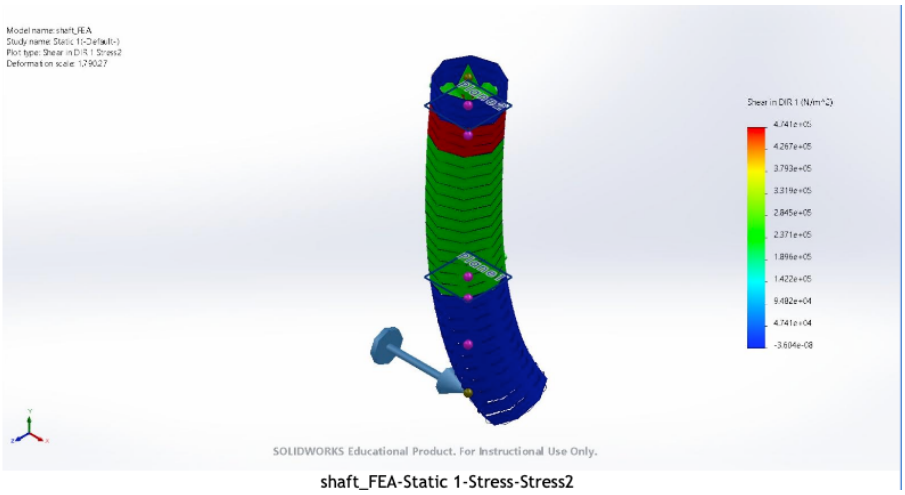
- **Fixture:** Immovable (No Translation) ที่ Joint ตำแหน่งลูกปืนบน (SKF 30203) และลูกปืนล่าง (F6901ZZ) จำลองการยึดแน่นของ Bearing Housing ในแนวรัศมี
- **Load:** แรงดึงสายพาน $F_{belt} = 130$ N ในแนวรัศมีที่ตำแหน่ง Belt Pulley และ Moment $M = 1.7$ N·m (แรงบิดที่ส่งผ่านสายพานเมื่อระบบทำงานเต็มกำลัง)

- **Mesh:** Beam Mesh, 37 Nodes, 30 Elements (Automatic, High Quality)

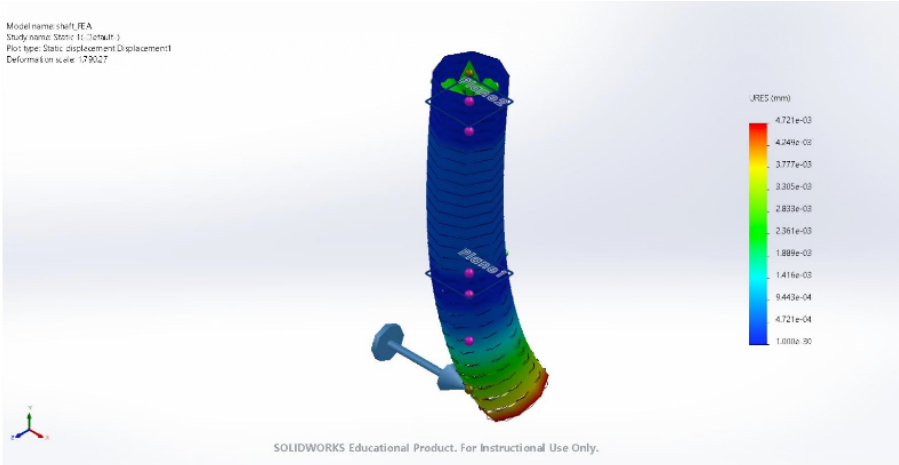


รูปที่ 3.6: การตั้งค่าโมเดล FEA เพลา แสดง Fixture ที่ตำแหน่งลูกปืน และภาระแรงดึงสายพาน 130 N พร้อม Moment 1.7 N·m

ผลการวิเคราะห์ FEA เพลา

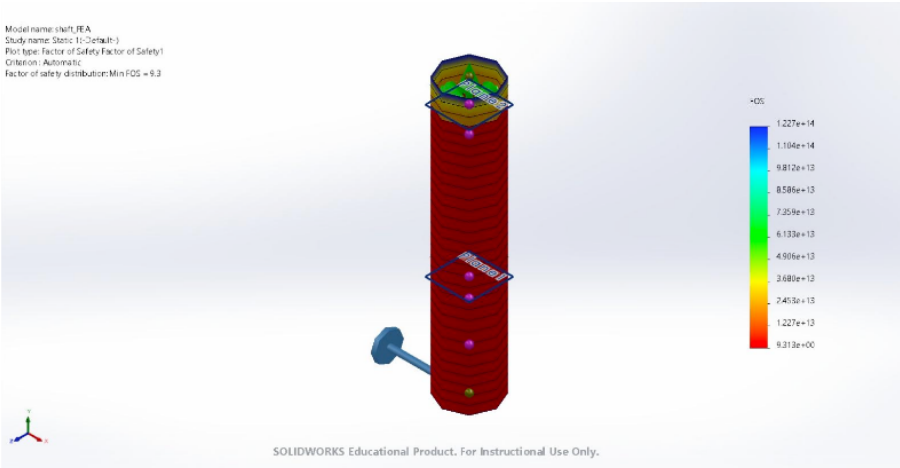


รูปที่ 3.7: ผล FEA เพลา: การกระจายตัวของ Upper Bound Axial and Bending Stress (Max $= 2.221 \times 10^7$ N/m^2 บริเวณตำแหน่ง Belt Pulley)



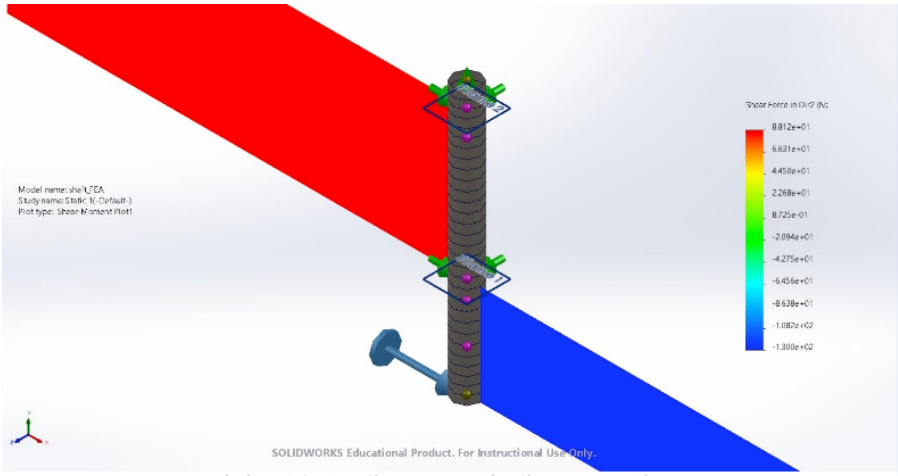
shaft_FEA-Static 1-Displacement-Displacement1

รูปที่ 3.8: ผล FEA เพลา: ระยะการเบี่ยงเบน (URES Displacement) (Max $= 4.721 \times 10^{-3}$ mm ที่ปลายเพลา)



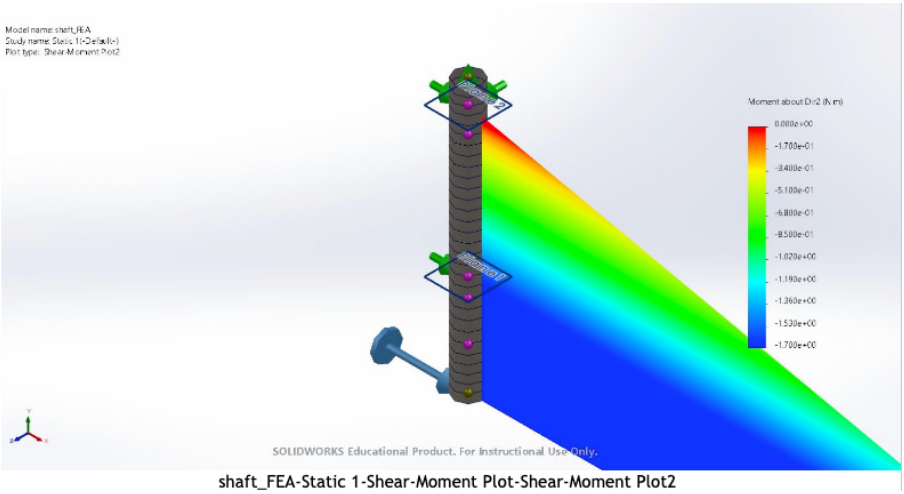
shaft_FEA-Static 1-Factor of Safety-Factor of Safety1

รูปที่ 3.9: ผล FEA เพลา: ค่าความปลอดภัย (Factor of Safety) (Min FOS $= 9.313$ บริเวณตำแหน่ง Belt Pulley)



shaft_FEA-Static 1-Shear-Moment Plot-Shear-Moment Plot1

รูปที่ 3.10: ผล FEA เพลา: แผนภาพแรงเฉือน (Shear Force in Dir2) แสดงการเปลี่ยนแปลงแรงเฉือนตลอดความยาวเพลา



shaft_FEA-Static 1-Shear-Moment Plot-Shear-Moment Plot2

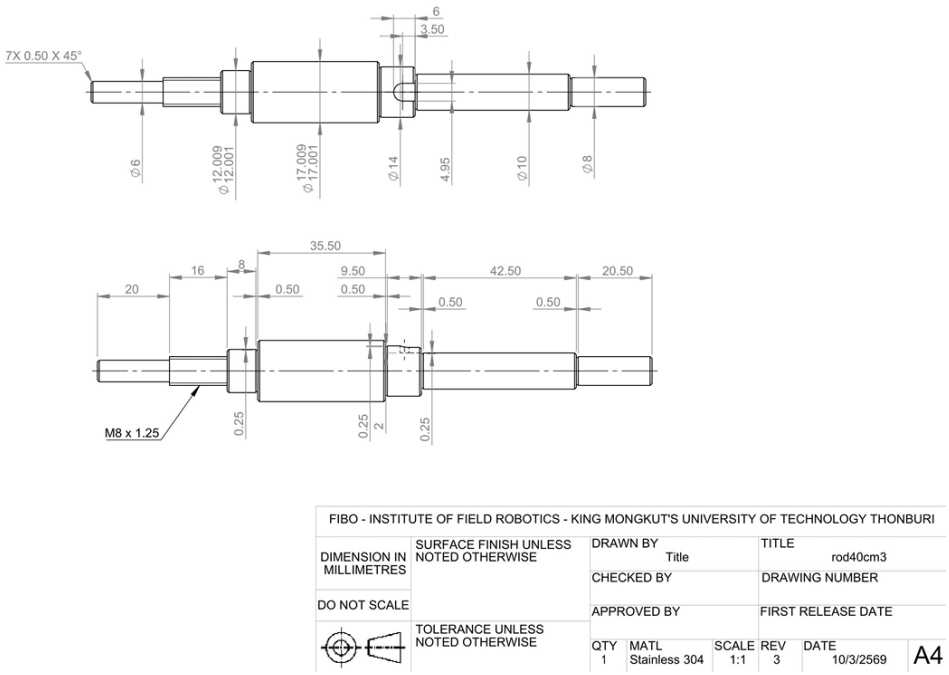
รูปที่ 3.11: ผล FEA เพลา: แผนภาพโมเมนต์ดัด (Bending Moment about Dir2) แสดงการกระจาย Bending Moment ตลอดความยาวเพลา

การตรวจสอบความสอดคล้องกับการคำนวณเชิงวิเคราะห์

ตารางที่ 3.4: สรุปผลการวิเคราะห์ FEA เพลาเทียบกับการคำนวณ ASME Code

พารามิเตอร์	ผล FEA	เกณฑ์	หน่วย	ผล
Max Bending+Axial Stress	2.221×10^7	$< 2.068 \times 10^8$	N/m^2	**ผ่าน**
Max Displacement (URES)	4.721×10^{-3}	--	mm	--
Min Factor of Safety	9.313	> 2.0	--	**ผ่าน**

ผล FEA ยืนยันผลการคำนวณ ASME Code ในหัวข้อ 3.2.2 โดย Min FOS = 9.313 สูงกว่าที่คำนวณได้จาก $d_{min} = 10.59$ mm เนื่องจากเส้นผ่าศูนย์กลางจริงที่จุดวิกฤต $d_{actual} = 17$ mm มี Margin สูงกว่าค่าขั้นต่ำอย่างมาก Max Displacement เพียง 4.721×10^{-3} mm บ่งชี้ว่าเพลามีความแข็งเกร็งเพียงพอ และการโก่งตัวของเพลาจะไม่ส่งผลต่อ Alignment ของ Encoder ที่ติดตั้งที่ Output Shaft ซึ่งเป็นปัจจัยสำคัญต่อความแม่นยำของ Closed-loop Control (A1, A2)



รูปที่ 3.12: Step Shaft Drawing แสดงขนาดเพลาแต่ละช่วงและตำแหน่งลูกปืน

3.2.4 การคำนวณสายพาน

ระบบเลือกใช้สายพาน HTD5M (High Torque Drive, 5 mm Pitch) แทน XL หรือ GT2 เนื่องจาก Curvilinear Tooth Profile ของ HTD ออกแบบมาเฉพาะสำหรับ High Torque Low Speed Application โดยกระจายแรงกดที่ฟันสายพานได้สม่ำเสมอกว่า Trapezoidal Profile ของ XL ซึ่งมีความเสี่ยง Tooth Jump เมื่อ Tight Side Tension สูง

ค่าเริ่มต้นของระบบสายพาน

ตารางที่ 3.5: ค่าเริ่มต้นของระบบสายพาน HTD5M

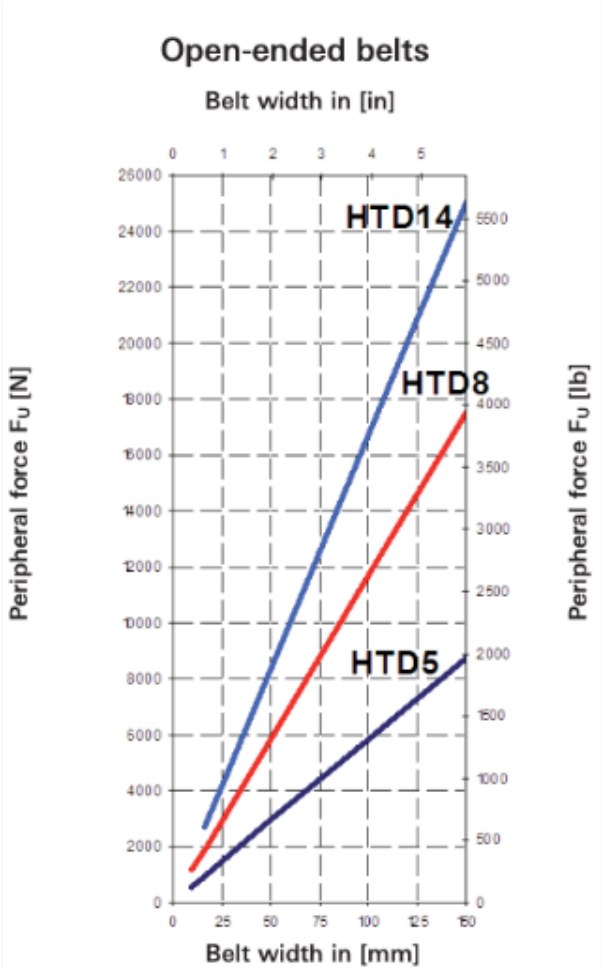
พารามิเตอร์	สัญลักษณ์	ค่า	หน่วย
ประเภทสายพาน	--	HTD5M (Steel Cord, Joined Endless)	--
ความกว้างสายพาน	b_0	15	mm
Belt Pitch	P_b	5	mm
Driver Pulley	z_1	24	teeth
Driven Pulley	z_2	70	teeth
ระยะศูนย์กลาง	e	181.25	mm
แรงบิดที่แขนกล	T_{arm}	8.3279	N·m
อัตราทด	i	2.917	--

Peripheral Force

$$d_1 = \frac{z_1 P_b}{\pi} = \frac{24 \times 5}{\pi} = 38.20\,\text{mm}, \qquad d_2 = \frac{z_2 P_b}{\pi} = \frac{70 \times 5}{\pi} = 111.41\,\text{mm} \tag{3.8}$$

$$F_U = \frac{2\,T_{driver}}{d_1} = \frac{2 \times 2.855}{0.03820} = 130\,\text{N} \tag{3.9}$$

ค่า Admissible Peripheral Force ของ HTD5M ขนาด 15 mm จาก Habasit Catalog คือ 400 N ดังนั้น $F_U = 130\,\text{N} \ll 400\,\text{N}$ **(ผ่าน)** ความกว้าง 15 mm เกินค่าขั้นต่ำที่คำนวณได้ ($b_{req} = 3.9$ mm) อย่างมีนัยสำคัญ เพื่อรองรับ Misalignment ที่อาจเกิดขึ้นใน การประกอบจริง และลดความเสี่ยง Edge Wear ของสายพาน



รูปที่ 3.13: ตารางค่า Admissible Peripheral Force ของสายพาน HTD5M จาก Habasit Engineering Guide แสดงพิกัด 400 N สำหรับความกว้าง 15 mm

Belt Length และ Teeth in Mesh

$$z_b = \frac{2e}{P_b} + \frac{z_1 + z_2}{2} + \frac{P_b}{4e}\left(\frac{z_2 - z_1}{\pi}\right)^2 = 119\,\text{teeth}, \qquad l_0 = z_b \times P_b = 595\,\text{mm} \tag{3.10}$$

$$z_m = z_1\left(0.5 - \frac{d_2 - d_1}{2\pi e}\right) = 6.74\,\text{teeth} > 5 \quad \Rightarrow \quad t_m = 1.0\ \textbf{(ผ่าน)} \tag{3.11}$$

Belt Tension และ Width Verification

$$F_1 = F_2 + F_U = 26 + 130 = 156\,\text{N (Tight Side)}, \qquad F_2 = 26\,\text{N (Slack Side)} \tag{3.12}$$

ตารางที่ 3.6: สรุปผลการคำนวณสายพาน HTD5M

พารามิเตอร์	ผลคำนวณ	เกณฑ์	หน่วย	ผล
Peripheral Force F_U	130	≤ 400	N	**ผ่าน**
Teeth in Mesh z_m	6.74	> 5	teeth	**ผ่าน**
Belt Width b_0	15	≥ 3.9	mm	**ผ่าน**
Belt Length l_0	595	--	mm	--
Tight Side Tension F_1	156	--	N	--
Slack Side Tension F_2	26	--	N	--

3.2.5 การคำนวณแรงบิดที่ต้องการและการเลือกมอเตอร์

การคำนวณนี้พิสูจน์ว่าชุดส่งกำลัง (RS-775 + Planetary Gearbox + Belt Drive) มีแรงบิดเพียงพอที่จะขับแขนกลให้เป็นไปตาม Requirements P3 และ P4 ภายใต้โหลดสูงสุด 2 kg

ความเฉื่อยรวมของระบบ

ความเฉื่อยรวม J_{total} ได้จากการวิเคราะห์ SolidWorks Model ครอบคลุมแขนกล, End Effector, และโหลดสูงสุด:

$$J_{\text{total}} = 0.5764 \text{ kg} \cdot \text{m}^2 \quad \text{(จาก SolidWorks)} \tag{3.13}$$

แรงบิดที่ต้องการ

แรงบิดที่ต้องการ (T_{req}) คำนวณจากกฎของนิวตันสำหรับระบบหมุน โดยใช้ความเร่งต่ำสุดที่ระบบต้องทำได้ตาม Requirement P4:

$$T_{\text{req}} = J_{\text{total}} \times \alpha_{\min} = 0.5764 \times 2.0 = 1.153 \text{ Nm} \tag{3.14}$$

รวม Safety Factor $SF = 2$ เพื่อรองรับแรงเสียดทานและความไม่แน่นอนในระบบ:

$$T_{\text{final}} = T_{\text{req}} \times SF = 1.153 \times 2.0 = 2.306 \text{ Nm} \tag{3.15}$$

แรงบิดที่ระบบสามารถส่งออกได้จริง

แรงบิดที่เพลาแขนกลคำนวณจากห่วงโซ่การส่งกำลังทั้งหมด:

$$T_{\text{arm}} = T_{\text{motor}} \times i_{\text{gear}} \times \eta_{\text{gear}} \times i_{\text{belt}} \times \eta_{\text{belt}} \tag{3.16}$$

$$T_{\text{arm}} = 0.0955 \times 24 \times 0.85 \times 3 \times 0.95 = 5.3977 \text{ Nm} \tag{3.17}$$

ตารางที่ 3.7: พารามิเตอร์ห่วงโซ่การส่งกำลังสำหรับการคำนวณแรงบิด

พารามิเตอร์	ค่า	หมายเหตุ
ความเฉื่อยรวม J_{total}	$0.5764 \text{ kg} \cdot \text{m}^2$	จาก SolidWorks (แขนกล + EE + โหลด)
ความเร่งสูงสุดที่คำนวณได้ $\alpha_{\max}$	9.3641 rad/s^2	มากกว่า P4 ($\leq 28.0 \text{ rad/s}^2$)
ความเร็วสูงสุดที่คำนวณได้ $\omega_{\max}$	8.9760 rad/s	มากกว่า P3 ($\leq 7.5 \text{ rad/s}$)
แรงบิดมอเตอร์ RS-775 T_{motor}	0.0955 Nm	จาก Datasheet (Stall Torque)
อัตราทด Planetary Gearbox i_{gear}	24:1	ภายในชุด Gearmotor
ประสิทธิภาพ Gearmotor η_{gear}	0.85	ค่าเฉลี่ยทั่วไปของ Planetary
แรงบิดหลัง Gearbox T_{gear}	1.9481 Nm	$T_{\text{motor}} \times i_{\text{gear}} \times \eta_{\text{gear}}$
อัตราทดสายพาน i_{belt} (70T : 24T)	3:1	Timing Belt HTD5M
ประสิทธิภาพสายพาน η_{belt}	0.95	จาก Habasit Catalogue
แรงบิดรวมที่เพลาแขนกล T_{arm}	5.3977 Nm	$T_{\text{gear}} \times i_{\text{belt}} \times \eta_{\text{belt}}$

การตรวจสอบ Safety Factor

$$SF_{\text{actual}} = \frac{T_{\text{arm}}}{T_{\text{req}}} = \frac{5.3977}{1.153} = 4.682 \tag{3.18}$$

$SF_{\text{actual}} = 4.682 > SF_{\text{req}} = 2.0$ และ $T_{\text{arm}} = 5.3977 \text{ Nm} > T_{\text{final}} = 2.306 \text{ Nm}$ ดังนั้นระบบส่งกำลัง **ผ่าน** ข้อกำหนดด้านแรงบิดและ Safety Factor

3.2.6 การเลือกตลับลูกปืนและการคำนวณอายุการใช้งาน

เลือกตลับลูกปืน 2 ตำแหน่งโดยอ้างอิงมาตรฐาน ISO 281 ที่ความเร็วรอบ 85 RPM

ลูกปืนบน (SKF 30203, Tapered Roller) เลือกเพราะต้องรับทั้ง Radial Load จากแรงดึงสายพานและ Axial Load จากน้ำหนักแขนกล พร้อมกัน ซึ่ง Tapered Roller Bearing รับ Combined Load ได้ดีกว่า Deep Groove Ball Bearing ที่ขนาด Bore เดียวกัน ลูกปืนล่าง (F6901ZZ, Deep Groove Ball) รับเฉพาะ Radial Load เป็นหลักจึงเลือก Ball Bearing ที่มีแรงเสียดทานต่ำกว่า

ตารางที่ 3.8: ผลการคำนวณอายุการใช้งานของตลับลูกปืน

ตำแหน่ง / รุ่น	C (kN)	P (kN)	L_{10h} (ชั่วโมง)	ผล
SKF 30203 (Taper Roller, ลูกปืนบน)	23.400	0.2600	6.1×10^8	**ผ่าน**
F6901ZZ (Deep Groove Ball, ลูกปืนล่าง)	2.886	0.1388	1.76×10^6	**ผ่าน**

เกณฑ์มาตรฐานทั่วไปสำหรับเครื่องจักรกล: 20,000--100,000 ชั่วโมง

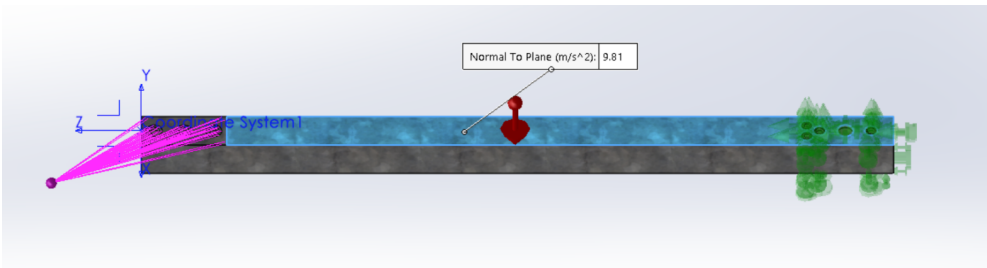
3.2.7 การวิเคราะห์การโก่งตัวด้วย FEA

ดำเนินการวิเคราะห์ Finite Element Analysis (Static Study) ด้วย SOLIDWORKS Simulation ภายใต้โหลด 2 kg ตามข้อกำหนด M3 วัสดุแขนกลเลือกใช้ Galvanized Steel แทน Stainless Steel AISI 304 เนื่องจากน้ำหนักที่เบากว่าช่วยลดโมเมนต์ความเฉื่อย J ของส่วนที่หมุน ซึ่งส่งผลโดยตรงต่อ Torque Budget และ Cycle Time (P1) โดยยอมรับ Trade-off ด้าน Corrosion Resistance ที่ต่ำกว่า เนื่องจากสภาพแวดล้อมการใช้งานเป็น Indoor Industrial

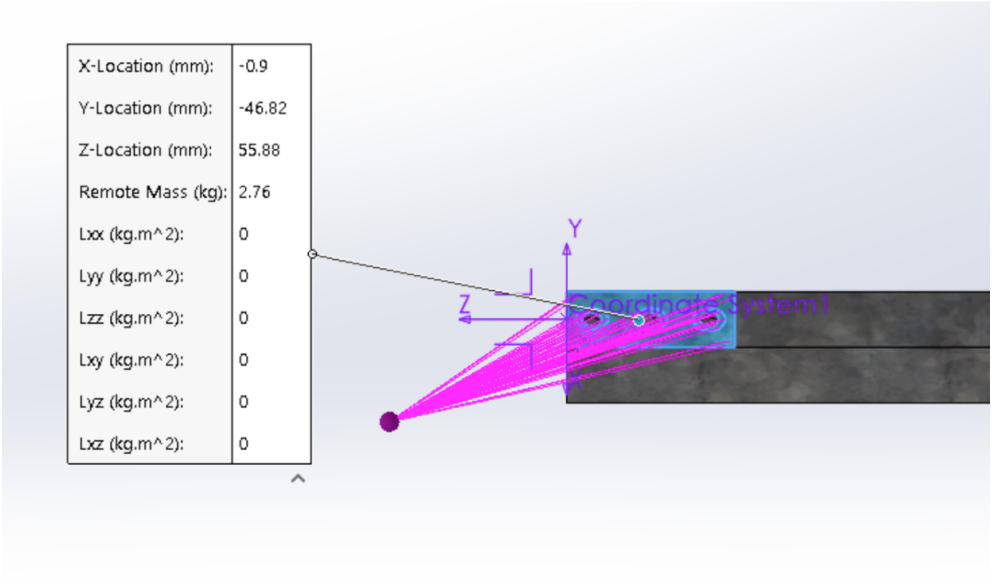
สภาวะการจำลอง (FEA Setup)

กำหนด Boundary Conditions ดังนี้

- **Fixture:** Fixed Geometry ที่ขอบรูน็อตทั้งสองฝั่ง และหน้าสัมผัสที่แนบกับตัวฐานหุ่นยนต์
- **Load:** Remote Load 2.76 kg ที่หน้าสัมผัสปลายแขน (ตำแหน่ง $x = -0.9$ mm, $y = -46.82$ mm, $z = 55.58$ mm) โดยใช้ Remote Load แทน Point Load เพื่อกระจายแรงผ่าน Remote Point ตามน้ำหนักจริงของ Gripper และ Connection Rod รวมกัน
- **Mesh:** Solid, Blended Curvature-based, High Quality (Jacobian 16 Points), 11,607 Nodes, 5,722 Elements



รูปที่ 3.14: โมเดลจำลอง FEA แสดงจุดจับยึด Fixture (สีฟ้า) และทิศทางของภาระแรง Remote Load (ลูกศรสีชมพู)



รูปที่ 3.15: การตั้งค่า Remote Load 2.76 kg แสดง Remote Point และ Moment of Inertia ที่กำหนด

การตรวจสอบเชิงวิเคราะห์ก่อน FEA (Analytical Verification)

ก่อนดำเนินการ FEA ทำการตรวจสอบเบื้องต้นด้วยการคำนวณเชิงวิเคราะห์ โดยจำลองแขนกลเป็น Cantilever Beam หน้าตัด Galvanized Square Tube 25×25 mm หนา 1.4 mm ความยาว $L = 0.5$ m

Second Moment of Inertia ของหน้าตัดท่อสี่เหลี่ยม

$$I = \frac{bh^3 - b_i h_i^3}{12} = \frac{25 \times 25^3 - 20.4 \times 20.4^3}{12} = 18{,}119.65\,\text{mm}^4 \tag{3.19}$$

การโก่งตัวแบ่งออกเป็น 2 ส่วน ได้แก่ การโก่งจากน้ำหนักแขนกลเอง (Distributed Load w) และการโก่งจากโหลดที่ปลายแขน (Point Load P)

$$\delta_1 = \frac{wL^4}{8EI} = \frac{5.123 \times 0.5^4}{8 \times 200{,}000 \times 18{,}119.65 \times 10^{-12}} = 0.0110\,\text{mm} \tag{3.20}$$

$$\delta_2 = \frac{PL^3}{3EI} = \frac{19.62 \times 0.5^3}{3 \times 200{,}000 \times 18{,}119.65 \times 10^{-12}} = 0.2256\,\text{mm} \tag{3.21}$$

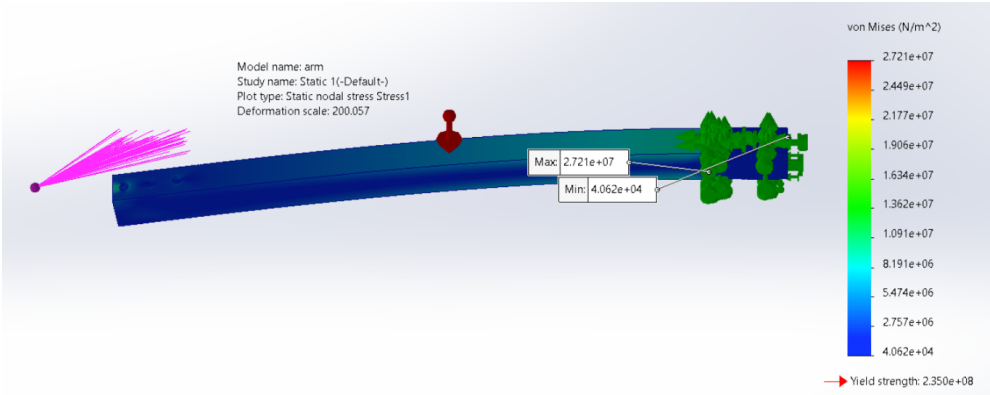
การโก่งตัวรวม

$$\delta_{total} = \delta_1 + \delta_2 = 0.0110 + 0.2256 = 0.2366\,\text{mm} < 1.0\,\text{mm} \quad \textbf{(ผ่าน)} \tag{3.22}$$

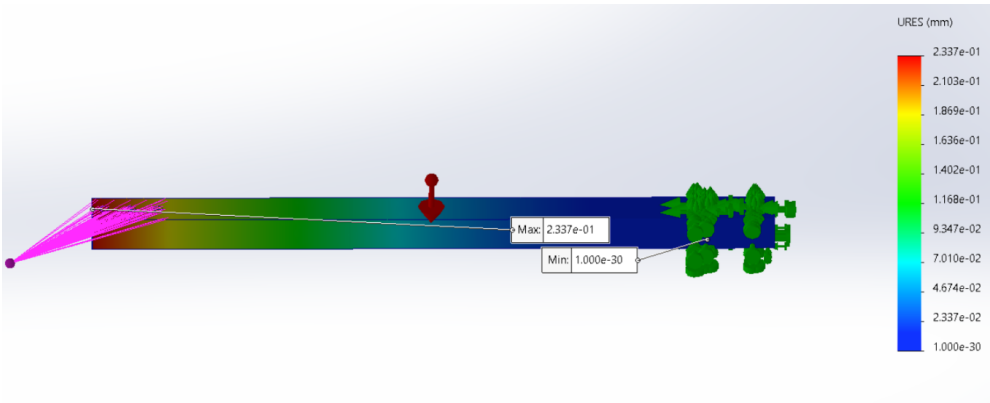
ผลการวิเคราะห์ FEA

ตารางที่ 3.9: คุณสมบัติวัสดุที่ใช้ใน FEA (Galvanized Steel)

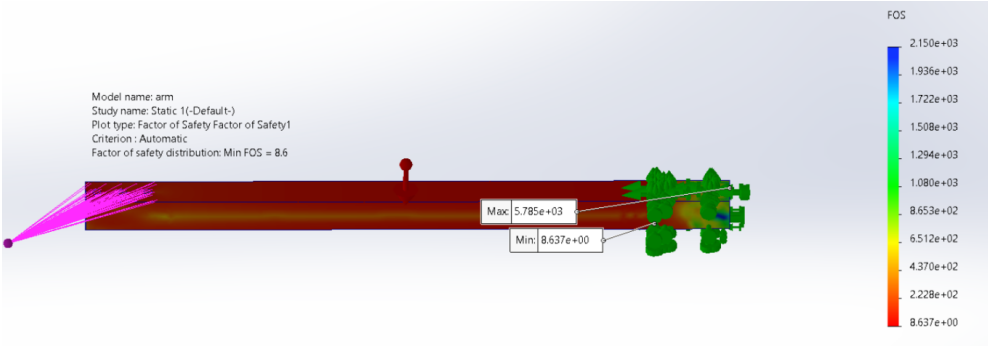
คุณสมบัติ	ค่า	หน่วย
Yield Strength	2.35×10^8	N/m^2
Elastic Modulus	2.00×10^{11}	N/m^2
Poisson's Ratio	0.26	--
Mass Density	7,850	kg/m^3



รูปที่ 3.16: ผล FEA: การกระจายตัวของ Von Mises Stress บนแขนกล (Max $= 2.721 \times 10^7$ N/m^2 บริเวณขอบหน้าสัมผัสกับฐาน)



รูปที่ 3.17: ผล FEA: ระยะการแอ่นตัว (Displacement) บนแขนกล (Max $= 0.2337$ mm ที่ปลายสุดของแขนกล)



รูปที่ 3.18: ผล FEA: ค่าความปลอดภัย (Factor of Safety) บนแขนกล (Min FOS $= 8.6$)

ตารางที่ 3.10: สรุปผลการวิเคราะห์ FEA เทียบกับข้อกำหนด M3

พารามิเตอร์	ผล FEA	เกณฑ์	หน่วย	ผล
Max Von Mises Stress	2.721×10^7	$< 2.35 \times 10^8$	N/m^2	**ผ่าน**
Max Displacement (URES)	0.2337	≤ 1.0	mm	**ผ่าน**
Min Factor of Safety	8.6	> 1.0	--	**ผ่าน**

ผลการคำนวณเชิงวิเคราะห์ ($\delta_{total} = 0.2366$ mm) สอดคล้องกับผล FEA ($\delta_{max} = 0.2337$ mm) โดยมีความคลาดเคลื่อน $< 2\%$ ซึ่งยืนยันความถูกต้องของแบบจำลอง FEA ทั้งสองวิธีให้ผลต่ำกว่าเกณฑ์ 1.0 mm อย่างมีนัยสำคัญ และ Min FOS $= 8.6 \gg 1.0$ แสดงว่าโครงสร้างมีความแข็งแรง เพียงพอสำหรับการใช้งานจริงตามข้อกำหนด M3

3.3. การออกแบบระบบไฟฟ้า

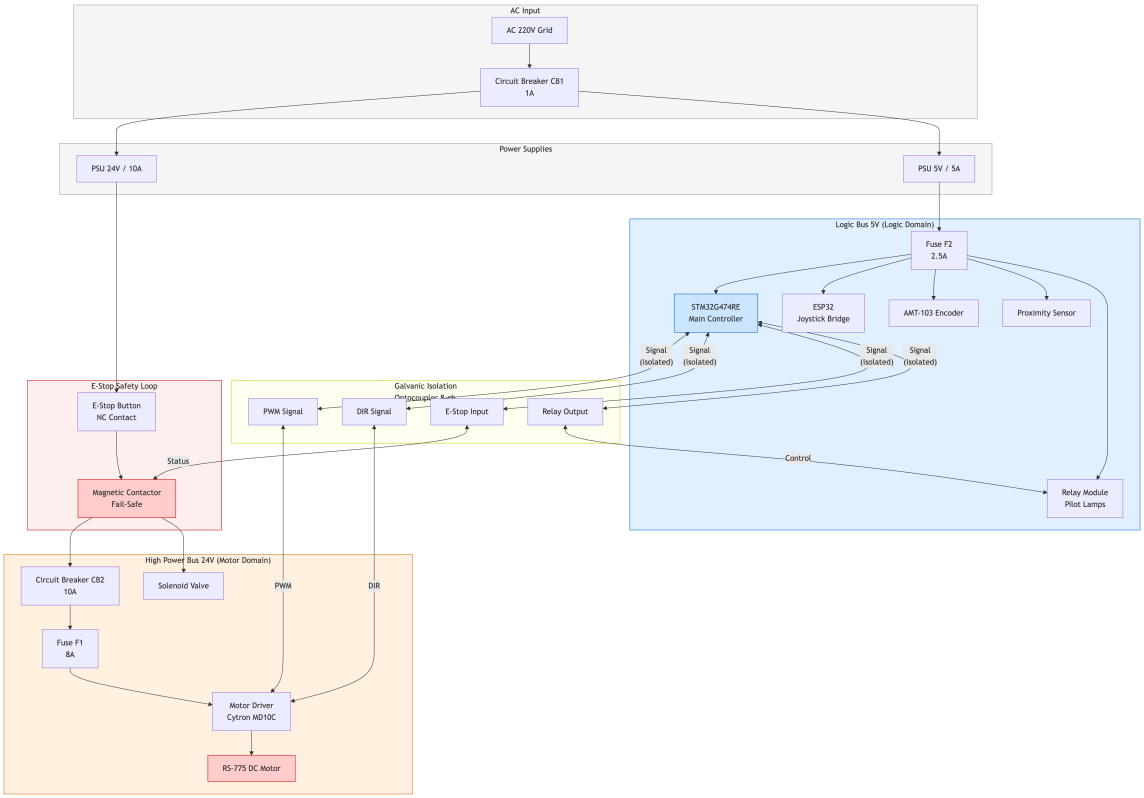
3.3.1 สถาปัตยกรรมไฟฟ้าโดยรวม

ระบบไฟฟ้าออกแบบโดยแบ่ง Power Bus ออกเป็น 2 วงจรที่แยกจากกันอย่างชัดเจน เพื่อให้ E-Stop ตัดเฉพาะกำลังไฟของมอเตอร์ตามข้อกำหนด E6 โดยไม่กระทบต่อการทำงานของ Controller และ Sensor

High Power Bus (24 V/10 A): ไฟ AC 220 V เข้าผ่าน Circuit Breaker CB1 แล้วจ่ายให้ PSU 24 V/ 10 A Output ของ PSU ต่อผ่าน NC Contact ของ E-Stop เข้า Magnetic Contactor เมื่อกด E-Stop สัญญาณ NC Contact เปิดวงจร ทำให้ Contactor ตัดไฟมอเตอร์ทันทีโดยไม่ผ่าน Firmware (Hardware Fail-Safe) การเลือกใช้ NC Contact แทน NO Contact เพื่อให้ระบบ Fail-Safe โดยธรรมชาติ กล่าวคือหากสายขาดหรือ Coil เสียหาย Contactor จะตัดไฟโดยอัตโนมัติ

Logic Bus (5 V): PSU 5 V ต่อตรงจาก AC 220 V *ก่อน* E-Stop Contactor ทำให้ STM32, ESP32, Encoder, Sensor และ Relay Control Side มีไฟเลี้ยงตลอดเวลาแม้กด E-Stop ซึ่งเป็นไปตามข้อกำหนด E6 ที่กำหนดให้ Controller ยังทำงานได้ หลัง E-Stop เพื่อรับ Reset Command และรายงาน Fault Code ผ่าน Modbus

สัญญาณดิจิทัลทุกเส้นระหว่าง High Voltage Side (24 V) และ Logic Side (5 V/3.3 V) ผ่าน Optocoupler Module 8 ช่อง เพื่อแยก Ground Loop และป้องกัน Noise จากการสวิตช์ PWM ของ Motor Driver ไม่ให้ Couple เข้าสู่วงจรลอจิกของ STM32



รูปที่ 3.19: Electrical Block Diagram แสดงการแบ่ง Power Bus และการแยก Signal ระหว่าง High Power และ Logic Domain

3.3.2 การคำนวณกำลังไฟฟ้าและการเลือก Circuit Breaker

ตารางที่ 3.11: การคำนวณกำลังไฟฟ้าของ 24 V Motor Bus

อุปกรณ์	แรงดัน (V)	กระแสสูงสุด (A)	กำลังไฟฟ้า (W)
DC Motor RS-775	24	10.00	240.00
Motor Driver MD10C	24	0.10	2.40
Contactor Coil	24	0.10	2.40
Relay Module	24	0.20	4.80
Solenoid Valve	24	0.50	12.00
Safety Margin 20%	--	2.18	52.32
รวม 24V Bus	**24**	**13.08**	**313.92**

กระแสรวม 13.08 A เกิน Rating ของ PSU 24 V/10 A อย่างไรก็ตาม Peak Current ของมอเตอร์เกิดขึ้นเพียงช่วงสั้นในระหว่าง Acceleration (Current Clamp Zone = 0.71% ของเวลา) ระบบจึง Implement Software Current Clamp ใน Firmware โดยจำกัด Duty Cycle สูงสุดตาม Voltage Budget

$$D_{max} = \frac{I_{limit} \cdot R_a}{V_{supply}} = \frac{8.0 \times 1.4534}{24} = 48.4\% \tag{3.23}$$

นอกจากนี้ Safety Stack ใน Firmware ยังตรวจสอบกระแสจริง จาก WCS1800 Current Sensor แบบ Real-time โดยหากกระแสเกิน $I_{limit} = 2.0$ A ต่อเนื่องนานกว่า 100 ms ระบบจะ Trigger Fault Code `0x42` ทันทีเป็น Software Fuse ชั้นที่สองนอกเหนือจาก Hardware Fuse

Safety Margin 20% เพิ่มเข้าไปเพื่อรองรับ Component Aging และ Inrush Current ที่อาจสูงกว่าค่า Steady-State ในช่วง Power-on ของ Solenoid Valve และ Relay Coil

ตารางที่ 3.12: การคำนวณกำลังไฟฟ้าของ 5 V Logic Bus

อุปกรณ์	แรงดัน (V)	กระแสสูงสุด (A)	กำลังไฟฟ้า (W)
STM32G474RE Nucleo	5	0.50	2.50
ESP32 DevKit V1	5	0.50	2.50
AMT-103 Encoder	5	0.10	0.50
Proximity Sensor	5	0.05	0.25
Optocoupler Module	5	0.04	0.20
Relay Module	5	0.10	0.50
LED Indicator (×3)	5	0.06	0.30
Selector Switch	5	0.01	0.05
Safety Margin 20%	--	0.27	1.36
รวม 5V Bus	**5**	**1.63**	**8.16**

กระแสรวม 1.63 A $<$ 5 A ของ PSU 5 V/5 A **(ผ่าน)**

ตารางที่ 3.13: การเลือกขนาด Circuit Breaker และ Fuse

อุปกรณ์	วงจร	กระแสโหลด (A)	ขนาดที่เลือก
Circuit Breaker CB1	AC Input -- 24V Branch	0.67	1 A
Circuit Breaker CB2	DC 24V -- Motor Branch	5.28	10 A
Fuse F1	DC 24V -- Motor	5.28	8 A
Fuse F2	DC 5V -- Logic	1.63	2.5 A

Fuse F1 เลือกขนาด 8 A ซึ่งต่ำกว่า Rating มอเตอร์ 10 A เพื่อให้ Fuse ขาดก่อนที่กระแสจะสูงเกินขีดจำกัด

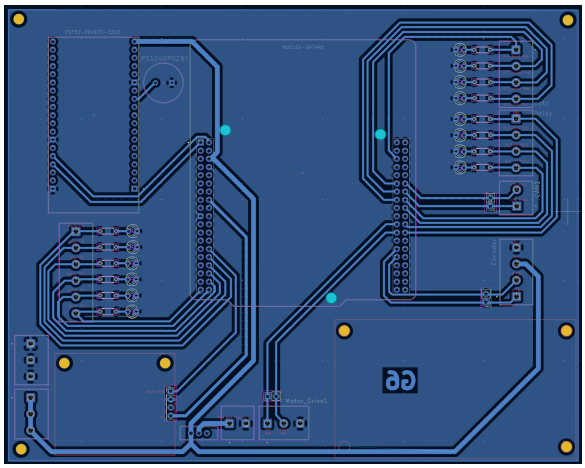
ของ Motor Driver MD10C และ Wiring ภายในตู้ควบคุม โดยยังมี Software Current Clamp เป็น Protection ชั้นแรก

3.3.3 การออกแบบ PCB และวงจรป้องกัน

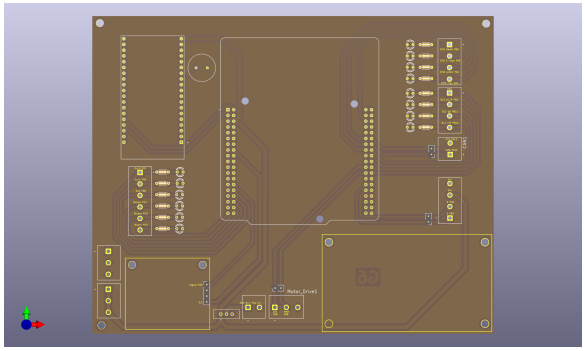
PCB ออกแบบด้วย KiCad 9.0.0 โดยรวม STM32G474RE Nucleo Board, ESP32-DEVKITC-32UE, Optocoupler Module 8 ช่อง, Relay Module และ Connector ทั้งหมดไว้บน Board เดียว แทนการต่อสายระหว่าง Module แยกกัน เพื่อลดจำนวน Connector Joint ภายในตู้ควบคุม ซึ่งเป็นแหล่งหลักของ Intermittent Fault ในสภาพแวดล้อมที่มีการสั่นสะเทือน

วงจรป้องกันตามข้อกำหนด E7 ออกแบบเป็น 3 ชั้น ดังนี้

- **Hardware Overcurrent Protection:** Circuit Breaker CB2 และ Fuse F1 ต่ออนุกรมใน 24 V Motor Bus ตัดวงจรเมื่อกระแสเกินพิกัด โดยไม่ต้องพึ่ง Firmware
- **Galvanic Isolation:** Optocoupler Module แยก Ground ระหว่าง High Power Domain (24 V) และ Logic Domain (5 V/3.3 V) ทุกเส้นสัญญาณ ป้องกัน Voltage Spike จากการสวิตช์ PWM ของ Motor Driver ไม่ให้เข้าถึง STM32
- **E-Stop Hardware Interlock:** NC Contact ของ E-Stop ต่อ Hard-wired ไปยัง Magnetic Contactor โดยตรง ไม่ผ่าน Firmware ทำให้ระบบหยุดได้แม้ STM32 เกิด System Hang



(a) PCB Layout (KiCad 9.0.0)

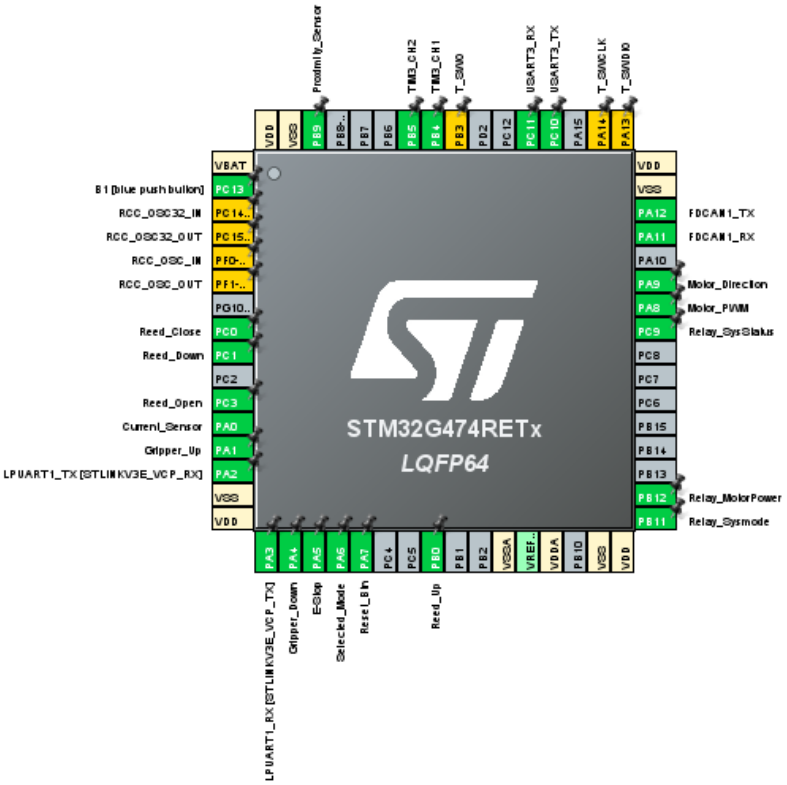


(b) PCB 3D Render

รูปที่ 3.20: การออกแบบ PCB สำหรับระบบควบคุม รวม STM32, ESP32, Optocoupler และ Relay ไว้บน Board เดียว

3.3.4 STM32 Pin Assignment

Pin Assignment ทั้งหมดกำหนดใน STM32CubeMX และล็อกไว้ใน `.ioc` file เพื่อป้องกันการเปลี่ยนแปลงโดยไม่ตั้งใจเมื่อ Regenerate Code



รูปที่ 3.21: STM32G474RE Pin Configuration จาก STM32CubeMX แสดง Peripheral Assignment ทั้งหมดของระบบ

ข้อสังเกตสำคัญด้าน GPIO Configuration:

- **Proximity Sensor (PB9):** ตั้งค่าเป็น `GPIO_PULLUP` เนื่องจากใช้ Open-Collector Optocoupler ซึ่ง Active Condition คือ `GPIO_PIN_SET` (transistor หยุดนำกระแส pin ลอยขึ้น 3.3 V ผ่าน Internal Pull-up) การตั้งค่าเป็น `GPIO_PULLDOWN` จะทำให้ MCU ตรวจจับสัญญาณไม่ได้เลย
- **Reed Switches (PB0, PC0, PC1, PC3):** ตั้งค่าเป็น `GPIO_PULLDOWN` เนื่องจาก Reed Switch ต่อระหว่าง Pin กับ VCC Active Condition คือ `GPIO_PIN_SET`
- **E-Stop (PA5):** ตั้งค่าเป็น `GPIO_PULLUP` สวิตช์ต่อแบบ NC Active Condition คือ `GPIO_PIN_RESET` (สวิตช์กด ตัดวงจร pin ถูก Pull-up ขึ้น HIGH)

ตารางที่ 3.14: STM32G474RE Pin Assignment (จาก CubeMX IOC)

Pin	GPIO Label	Signal	Mode	Config
Motor Control				
PA8	Motor_PWM	TIM1_CH1	PWM Output	AF, 19.6 kHz, ARR=50
PA9	Motor_Direction	GPIO Output	Push-Pull	--
Encoder (QEI)				
PB4	--	TIM3_CH1	Encoder Interface	NOPULL
PB5	--	TIM3_CH2	Encoder Interface	NOPULL
Communication				
PA2	LPUART1_TX	LPUART1_TX	Async UART	230400 bps, 8E1
PA3	LPUART1_RX	LPUART1_RX	Async UART	230400 bps, 8E1
PC10	--	USART3_TX	Async UART	115200 bps, 8N1
PC11	--	USART3_RX	Async UART	115200 bps, 8N1
PA11	--	FDCAN1_RX	CAN Bus	500 kbps classic CAN
PA12	--	FDCAN1_TX	CAN Bus	500 kbps classic CAN
Safety & Control Input				
PA5	E-Stop	GPIO Input	PULLUP	Active HIGH
PA6	Selected_Mode	GPIO Input	PULLUP	Active LOW
PA7	Reset_Btn	GPIO Input	PULLUP	Active LOW
PB9	Proximity_Sensor	GPIO Input	PULLUP	Active HIGH (open-collector)
Relay Output				
PB12	Relay_MotorPowe	GPIO Output	Push-Pull	Motor Power Contactor
PB11	Relay_Sysmode	GPIO Output	Push-Pull	Mode Pilot Lamp
PB2	Relay_SysStatus	GPIO Output	Push-Pull	Status Pilot Lamp
Gripper Output				
PA1	Gripper_Up	GPIO Output	Push-Pull	Relay: Arm Up
PA4	Gripper_Down	GPIO Output	Push-Pull	Relay: Arm Down
Reed Switch Input (Position Feedback)				
PB0	Reed_Up	GPIO Input	PULLDOWN	Gripper Arm Up confirmed
PC0	Reed_Close	GPIO Input	PULLDOWN	Claw Close confirmed
PC1	Reed_Down	GPIO Input	PULLDOWN	Gripper Arm Down confirmed
PC3	Reed_Open	GPIO Input	PULLDOWN	Claw Open confirmed
Analog Input				
PA0	Current_Sensor	ADC1_IN1	Analog DMA	WCS1800, DMA1_Ch1, Continuous

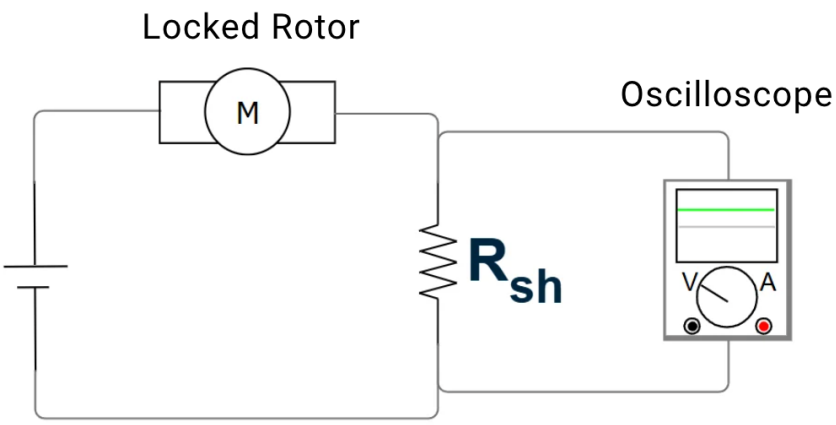
3.3.5 การระบุระบบ (System Identification)

การระบุระบบแบ่งออกเป็นสองขั้นตอนตามลำดับ ได้แก่ การวัดพารามิเตอร์ทางไฟฟ้าโดยตรงด้วย Locked Rotor Test และการประมาณพารามิเตอร์เชิงกลด้วย Grey-box Parameter Estimation โดยใช้ Chirp Signal เป็น Training Data แนวทางนี้เลือกใช้เนื่องจากพารามิเตอร์ทางไฟฟ้า (R_a, L_a) สามารถวัดได้โดยตรงจากวงจรเมื่อ Back-EMF = 0 ในขณะที่พารามิเตอร์เชิงกล (J, B, η, K_t, K_b) ต้องอาศัยการสังเกตพฤติกรรมของระบบขณะหมุนจริง ภายหลังการประมาณค่าพารามิเตอร์ แบบจำลองที่ได้รับการตรวจสอบ ความถูกต้อง (Cross-Validation) ด้วยข้อมูล 2 ประเภทที่ไม่ได้ใช้ ในการ Training ได้แก่ Step Input และ Stair-Step Input เพื่อยืนยันว่าแบบจำลองสามารถทำนายการตอบสนองได้ถูกต้อง ทั้งในสภาวะ Transient และ Steady-State ที่แรงดันหลายระดับ

Locked Rotor Test: วัด R_a และ L_a

หลักการและ Setup เมื่อล็อกเพลามอเตอร์ไม่ให้หมุน ($\omega = 0$) Back-EMF จะมีค่าเป็นศูนย์ ($e_b = K_b\omega = 0$) วงจรจึงลดรูปลงเหลือเพียง RL Series Circuit ดังแสดงในรูปที่ 3.22 การป้อน Step Voltage V_{in} แล้ววัดแรงดันตกคร่อม Shunt Resistor R_{sh} จึงเพียงพอสำหรับคำนวณทั้ง R_a และ L_a โดยตรงจาก KVL ของวงจร RL ตามที่อธิบายในหัวข้อ 2.5.2

$$V_{in}(t) = (R_a + R_{sh})\, i(t) + L_a \frac{di(t)}{dt} \tag{3.24}$$



รูปที่ 3.22: วงจรทดสอบ Locked Rotor Test แสดงการต่อ Shunt Resistor อนุกรมกับมอเตอร์เพื่อวัดกระแสทางอ้อมผ่าน Oscilloscope

การเลือก $R_{sh} = 0.1\,\Omega$ การเลือกค่า R_{sh} ต้องสอดคล้องเงื่อนไขสองข้อพร้อมกัน

เงื่อนไขที่ 1 --- ไม่รบกวนวงจร: R_{sh} ต้องเล็กกว่า R_a มาก เพื่อไม่ให้แรงดันตกคร่อมที่เพิ่มขึ้นเปลี่ยนพฤติกรรมของวงจร โดยกำหนดเกณฑ์ที่ 10%

$$\frac{R_{sh}}{R_a} < 0.10 \quad \Rightarrow \quad R_{sh} < 0.10 \times 1.453 = 0.1453\,\Omega \tag{3.25}$$

เงื่อนไขที่ 2 --- ทนกำลังไฟได้: กระแส Stall สูงสุดประมาณได้จาก $I_{stall} \approx V_{in}/R_a$ กำลังที่ R_{sh} ต้องรับที่ $V_{in} = 12\,\text{V}$ คือ

$$P_{sh} = I_{stall}^2 R_{sh} = \left(\frac{12}{1.453}\right)^2 \times 0.1 = (8.26)^2 \times 0.1 = 0.682\,\text{W} \tag{3.26}$$

เลือก $R_{sh} = 0.1\,\Omega$ ขนาด 50 W ซึ่งตอบสนองทั้งสองเงื่อนไข และมีขนาดมาตรฐานหาซื้อได้ทั่วไป ตรวจสอบสัดส่วน

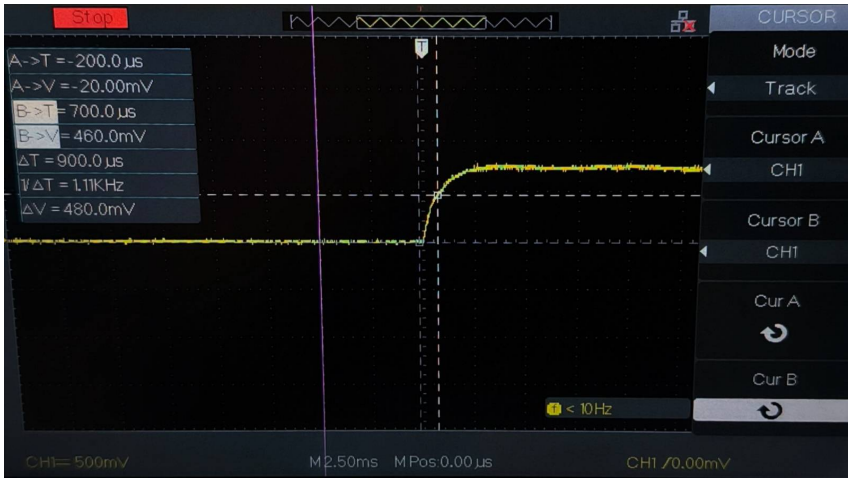
$$\frac{R_{sh}}{R_a} = \frac{0.1}{1.453} = 6.9\% < 10\% \quad \checkmark \tag{3.27}$$

การหาสมการคำนวณพารามิเตอร์ เมื่อระบบเข้าสู่ Steady State ($di/dt = 0$) กระแสในวงจรคือ $i_{ss} = V_{sh,ss}/R_{sh}$ แทนลงในสมการ (3.24) ได้

$$V_{in} = i_{ss}(R_a + R_{sh}) \quad \Rightarrow \quad R_a = R_{sh}\left(\frac{V_{in}}{V_{sh,ss}} - 1\right) \tag{3.28}$$

การตอบสนองของกระแสเป็น First-order Exponential ค่าคงที่เวลา $\tau = L_a/(R_a + R_{sh})$ ซึ่งวัดได้จาก Oscilloscope ที่จุด 63.2% ของ $V_{sh,ss}$ จัดรูปใหม่ได้

$$L_a = R_{sh}\left(\tau\,\frac{V_{in}}{V_{sh,ss}}\right) \tag{3.29}$$



รูปที่ 3.23: Oscilloscope Waveform จาก Locked Rotor Test ที่ $V_{in} = 12\,\text{V}$ แสดงแรงดัน V_{sh} ที่ไต่ระดับแบบ First-order Exponential พร้อมจุดอ่านค่า $V_{sh,ss}$ และ τ

การเลือกแรงดันทดสอบและการวิเคราะห์ความสม่ำเสมอ ทดสอบที่แรงดัน 3 ระดับ ($V_{in} = 6, 9, 12\,\text{V}$) โดยแต่ละระดับทำซ้ำ 3 ครั้งเพื่อประเมิน Statistical Consistency

ตารางที่ 3.15: ผลการวัด R_a และ L_a จาก Locked Rotor Test ทั้ง 3 แรงดัน

V_{in} (V)	$\bar{R}_a$ (Ω)	σ_{R_a} (Ω)	$\bar{L}_a$ (H)	σ_{L_a} (H)
6	1.7300	0.0374	0.001490	0.000062
9	1.5398	0.0604	0.001473	0.000095
12	**1.4534**	0.0523	**0.001448**	0.000055

จากตารางพบว่าค่า $\bar{R}_a$ มีแนวโน้มลดลงเมื่อแรงดันเพิ่มขึ้น สาเหตุหลักมาจากสองปัจจัย ได้แก่

- **Contact Resistance:** ที่ V_{in} ต่ำ กระแสที่ไหลผ่าน Brush Contact มีค่าน้อย ทำให้ความต้านทานสัมผัสของแปรงถ่านมีสัดส่วนสูงกว่าในค่า R_a รวม เมื่อแรงดันสูงขึ้น กระแสเพิ่มขึ้น สัดส่วนของ Contact Resistance ลดลง
- **Signal-to-Noise Ratio:** ที่ $V_{in} = 6$ V แรงดัน $V_{sh,ss} \approx 0.37$ V ซึ่งมีขนาดใกล้เคียงกับ Noise Floor ของ Oscilloscope ทำให้ Error ในการอ่านค่า τ และ $V_{sh,ss}$ สูงกว่า สะท้อนใน $\sigma_{R_a} = 0.0374\,\Omega$ ที่ต่ำผิดปกติ

เลือกใช้ผลจาก $V_{in} = 12$ V เป็นค่าอ้างอิงด้วยสองเหตุผล ได้แก่ ใกล้เคียงแรงดันใช้งานจริง 24 V มากที่สุดในชุดที่ทดสอบ ทำให้สภาวะ Brush Contact สอดคล้องกับการทำงานจริง และ $\sigma_{R_a} = 0.0523\,\Omega$ ต่ำที่สุดในสามแรงดัน ยืนยันความสม่ำเสมอของการวัด ทั้งนี้ไม่ทดสอบที่ 24 V เนื่องจากเพลาจะหมุนแรงเกินไปและอาจก่อให้เกิดอันตรายได้

Chirp Excitation และ Grey-box Parameter Estimation

เหตุผลในการเลือก Chirp Signal พารามิเตอร์เชิงกล (J, B, η, K_t, K_b) ไม่สามารถวัดโดยตรงได้จากวงจร จำเป็นต้องสังเกตพฤติกรรมของระบบขณะหมุนจริงภายใต้ Input ที่ครอบคลุม Frequency Content ของ Slow Dynamics ที่ต้องการระบุ Chirp Signal (Linear Frequency Sweep) ถูกเลือกเนื่องจาก

- **Persistent Excitation:** Chirp ที่ Sweep จาก f_1 ถึง f_2 การันตีว่าทุก Frequency ในช่วงดังกล่าวได้รับการกระตุ้น ต่างจาก Step Signal ที่มี Spectral Energy กระจุกที่ DC และลดลงตาม $1/f$
- **ครอบคลุม Mechanical Bandwidth:** ย่านความถี่ 0.1--2.0 Hz ครอบคลุม Slow Dynamics ของ Inertia และ Viscous Damping เมื่อเทียบกับ Mechanical Bandwidth ของระบบที่คำนวณได้จากพารามิเตอร์ทางไฟฟ้า

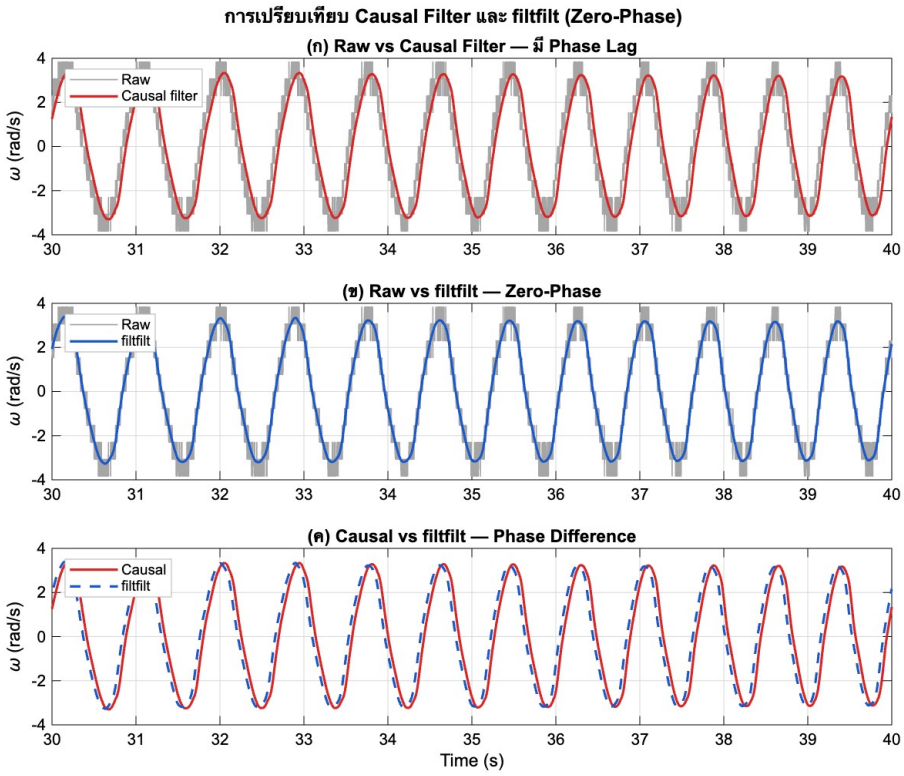
$$f_{mech} = \frac{1}{2\pi}\sqrt{\frac{R_a B + K_b K_t N^2}{L_a J}} = \frac{\sqrt{8136}}{2\pi} \approx 14.4\,\text{Hz} \tag{3.30}$$

ย่าน 0.1--2.0 Hz จึงอยู่ต่ำกว่า f_{mech} อย่างมีนัยสำคัญ ทำให้ Dynamic ที่วัดได้สะท้อน Inertia และ Damping โดยตรง โดยไม่ปะปนกับ Electrical Pole ที่ $R_a/L_a \approx 1004$ rad/s

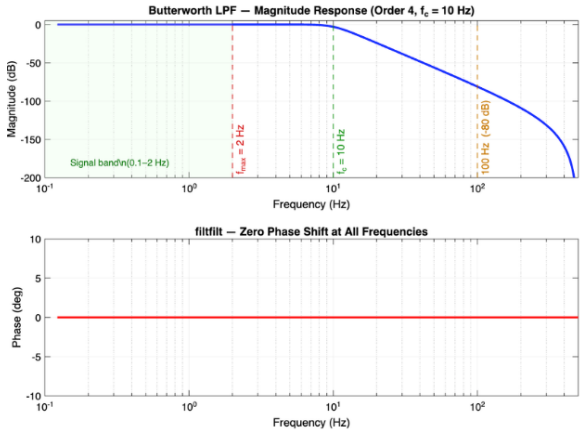
Preprocessing และ Filtering ข้อมูลดิบกรองด้วย Butterworth Low-pass Filter Order 4 ที่ $f_c = 10$ Hz ด้วยคำสั่ง `filtfilt` ของ MATLAB (หลักการ Zero-Phase Filtering อธิบายไว้ในหัวข้อ 2.5.3) เลือก $f_c = 10$ Hz เนื่องจาก

- สูงกว่าย่านความถี่ Identification (0.1--2.0 Hz) ถึง 5 เท่า ทำให้ Attenuation ที่ขอบบนของย่าน ($f = 2$ Hz) น้อยกว่า 0.001% ตามที่แสดงในสมการ (2.42) จึงไม่ลดทอน Amplitude ของ Signal ที่ต้องการ
- กำจัด Noise จาก PWM Switching ($f_{pwm} = 19.6$ kHz) และ Quantization Noise ของ Encoder ที่ความถี่สูงได้อย่างมีประสิทธิภาพ

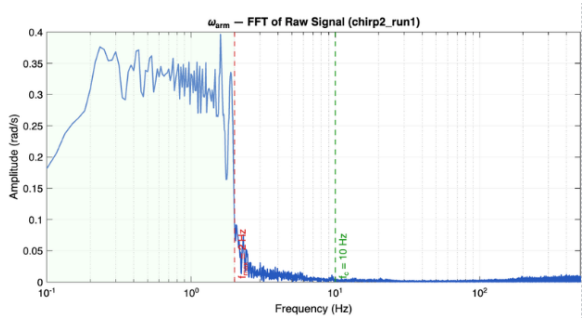
ความถูกต้องของการเลือก f_c ซึ่งแสดงว่า Magnitude และ Phase ของ `filtfilt` ทับกับ Raw Signal ได้อย่างสมบูรณ์ ในย่าน 0.1--2.0 Hz และรูปที่ 3.24 แสดงให้เห็นว่า `filtfilt` ไม่มี Phase Lag ต่างจาก Causal Filter ที่มี Lag สังเกตได้ชัดเจน



รูปที่ 3.24: เปรียบเทียบ Causal Filter (สีแดง) และ `filtfilt` (สีน้ำเงิน) ในโดเมนเวลา (ก) Causal Filter มี Phase Lag สังเกตได้ชัดเจน (ข) `filtfilt` ทับ Raw Signal ได้ตรงทุกจุด (ค) Phase Difference ระหว่างสองวิธี



(a) Magnitude Response ของ Butterworth LPF Order 4 ($f_c = 10$ Hz) และ Phase Response ของ `filtfilt` ที่เป็นศูนย์ในทุกความถี่



(b) FFT ของ Raw Signal ω_{arm} จาก Chirp Run 1 (0.1--2 Hz) แสดง Signal Energy กระจุกที่ 0.1--2 Hz และ Noise Floor ที่ความถี่สูงกว่า $f_c = 10$ Hz

รูปที่ 3.25: การตรวจสอบความเหมาะสมของ $f_c = 10$ Hz กับข้อมูลจริงของระบบ: (ซ้าย) `filtfilt` ให้ Phase Shift เป็นศูนย์ และ Attenuation ที่ 2 Hz น้อยกว่า 0.001% (ขวา) Signal Dynamics ของระบบจริงทั้งหมด อยู่ในย่าน 0.1--2 Hz ห่างจาก f_c ถึง 5 เท่า ยืนยันว่า Filter ไม่รบกวน Signal ที่ต้องการ

ผลในรูปที่ 3.25 ยืนยันความเหมาะสมของการเลือก $f_c = 10$ Hz จากสองมุมมอง ได้แก่

มุมมองที่ 1 --- คุณสมบัติของ Filter (กราฟซ้าย): Butterworth LPF Order 4 มี Magnitude Response แบน (0 dB) ในย่าน Passband ตลอด 0.1--2 Hz โดยไม่มี Ripple และ Roll-off เริ่มต้นที่ $f_c = 10$ Hz ที่ความถี่ 100 Hz ซึ่งเป็นย่าน Noise Filter ลดทอนสัญญาณได้ถึง -80 dB หรือเทียบเท่า ลดแอมพลิจูดลง 10,000 เท่า ที่สำคัญ Phase Response ของ `filtfilt` เป็นเส้นตรงที่ $0°$ ในทุกความถี่โดยไม่มีข้อยกเว้น ยืนยันว่าไม่มี Phase Distortion เกิดขึ้นกับ Signal ใดๆ ในชุดข้อมูล

มุมมองที่ 2 --- Spectral Content ของข้อมูลจริง (กราฟขวา): FFT ของ ω_{arm} จาก Chirp Run 1 แสดงให้เห็นชัดว่า Signal Energy ทั้งหมดกระจุกอยู่ในย่าน 0.1--2 Hz ตรงกับความถี่ของ Chirp Input ที่ป้อนเข้าระบบ หลังจาก $f = 2$ Hz แอมพลิจูดลดลงอย่างรวดเร็ว และกลายเป็น Noise Floor ที่มีขนาดเล็กมาก (< 0.01 rad/s) ตลอดย่าน 2--500 Hz ระยะห่างระหว่าง $f_{max,signal} = 2$ Hz และ $f_c = 10$ Hz จึงมี Margin 5 เท่า ซึ่งเพียงพอที่จะการันตีว่า Filter ไม่ตัด Signal ที่มีประโยชน์ออกไปแม้แต่น้อย

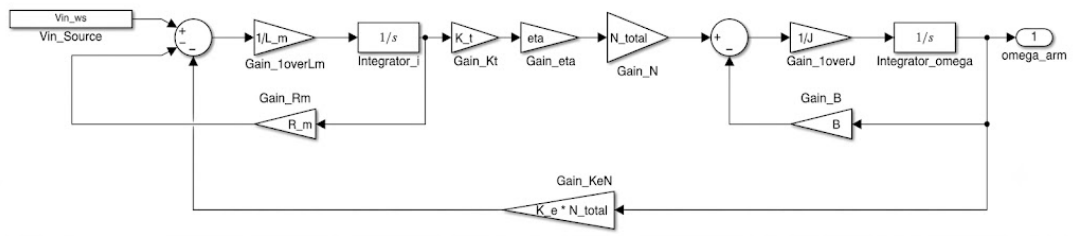
Grey-box Estimation บน MATLAB การประมาณค่าพารามิเตอร์ใช้ `idgrey` + `greyest` บน MATLAB System Identification Toolbox โดยกำหนดโครงสร้างเมทริกซ์ $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ตามสมการ State-Space ในสมการ (2.6) แล้วปล่อยให้ $\mathbf{p} = [K_t,\ K_b,\ J,\ B,\ \eta]^\top$ เป็น Free Parameters อัลกอริทึมใช้ Prediction Error Method (PEM) ลดรูป Cost Function

$$V_N(\mathbf{p}) = \frac{1}{N}\sum_{k=1}^{N}\left|y(k) - \hat{y}(k \mid \mathbf{p})\right|^2 \tag{3.31}$$

ตามที่อธิบายในหัวข้อ 2.5.2 ทดสอบ 3 ย่านความถี่ $\times$ 3 runs/ย่าน = 9 Experiments รวม ดังแสดงในตารางที่ 3.16

ตารางที่ 3.16: การตั้งค่า Chirp Signal สำหรับ Grey-box Estimation

ย่านความถี่	f_{start} (Hz)	f_{end} (Hz)	จำนวน Runs
Low Band	0.1	0.5	3
Mid Band	0.1	1.0	3
High Band	0.1	2.0	3
รวม	--	--	**9**

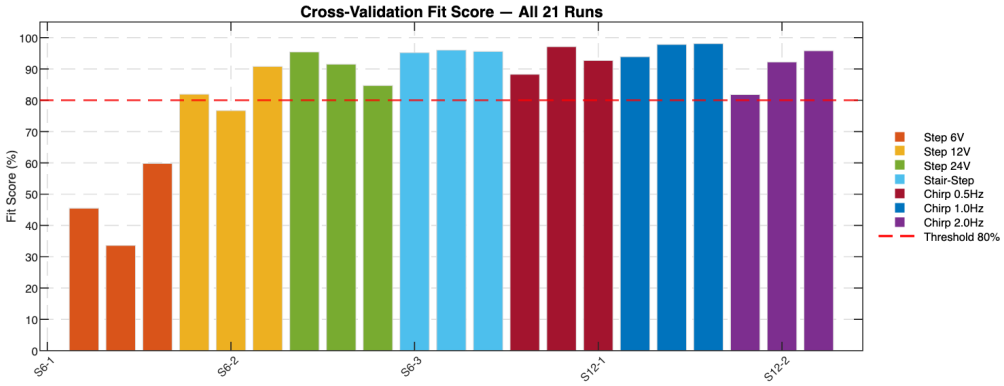


รูปที่ 3.26: Simulink Block Diagram ของแบบจำลอง DC Motor ที่ใช้ใน Grey-box Parameter Estimation แสดง Free Parameters ที่ปรับค่าโดย `greyest`

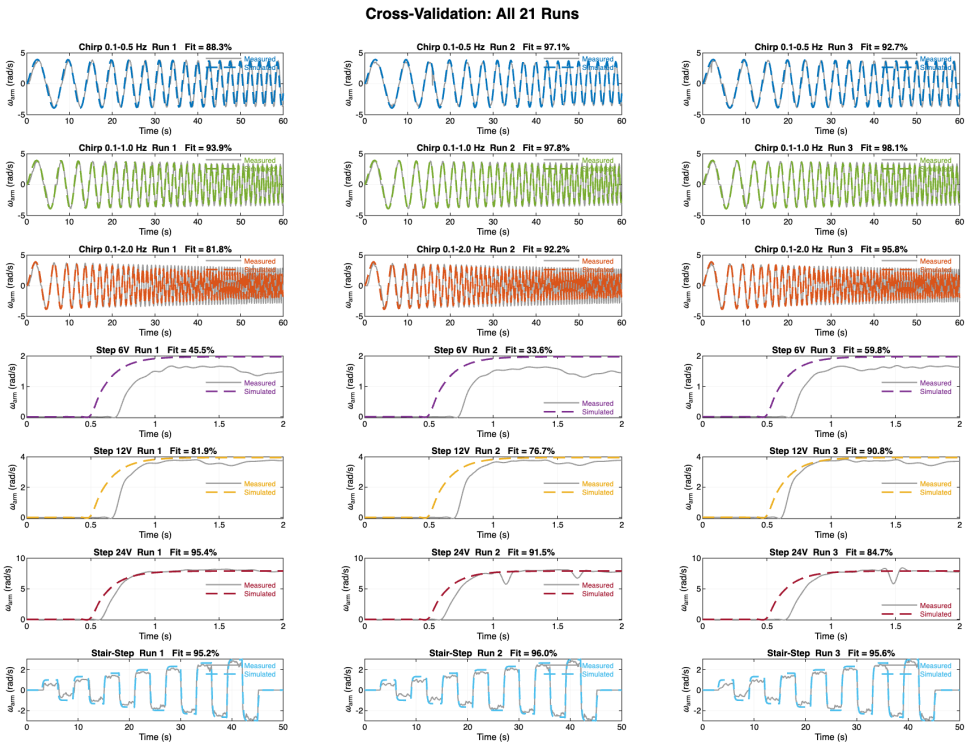
Cross-Validation เพื่อยืนยันว่าแบบจำลองที่ได้ไม่เกิด Overfitting กับ Chirp Training Data จึงทดสอบกับข้อมูล 2 ประเภทที่ไม่ได้ใช้ใน Training ได้แก่ Step Input (3 แรงดัน: 6, 12, 24 V) และ Stair-Step Input รวม 21 Runs ทั้งหมด ผล Fit Score วัดจาก NRMSE ระหว่าง Simulated Output กับ Measured Data แสดงในตารางที่ 3.17 และรูปที่ 3.28

ตารางที่ 3.17: ผลการ Cross-Validation ของแบบจำลองกับข้อมูลประเภทต่าง ๆ

ประเภทข้อมูล	แรงดัน	Fit Score (%)	หมายเหตุ
Chirp (Training)	0.1--0.5 Hz	88.3--97.1	--
Chirp (Training)	0.1--1.0 Hz	93.9--98.1	--
Chirp (Training)	0.1--2.0 Hz	81.8--92.2	--
Step (Validation)	6 V	33.6--59.8	Static Friction dominant
Step (Validation)	12 V	76.7--90.8	--
Step (Validation)	24 V	84.7--95.4	--
Stair-Step (Validation)	24 V	95.2--96.6	--



รูปที่ 3.27: Cross-Validation Fit Score ทั้ง 21 Runs แสดงเป็น Histogram จำแนกตามประเภทข้อมูล เส้นประสีแดงคือเกณฑ์ขั้นต่ำที่ยอมรับได้ 80%



รูปที่ 3.28: เปรียบเทียบ Measured (เส้นทึบ) และ Simulated (เส้นประ) ทั้ง 21 Runs จำแนกตามประเภท Input ได้แก่ Chirp 3 ย่านความถี่, Step 3 แรงดัน และ Stair-Step

ผล Fit Score ที่ต่ำผิดปกติที่ Step 6 V (33.6--59.8%) เกิดจาก Static Friction ของ Bearing และ Belt Drive มีนัยสำคัญที่แรงดันต่ำ ในขณะที่แบบจำลองเชิงเส้น ตามสมการ (2.4) ละทิ้ง Coulomb Friction ออกไป ตามสมมติฐานในหัวข้อ 2.1 อย่างไรก็ตาม ที่แรงดันใช้งานจริง 24 V Fit Score = 84.7--95.4% (Step) และ 95.2--96.6% (Stair-Step) ยืนยันว่าแบบจำลองเพียงพอสำหรับการออกแบบตัวควบคุม ในย่านแรงดันที่ใช้งานจริง

ผลการระบุระบบ

ตารางที่ 3.18: พารามิเตอร์ที่ได้จาก System Identification

พารามิเตอร์	สัญลักษณ์	ค่า	หน่วย	วิธีการ
Armature Resistance	R_a	1.4534	Ω	Locked Rotor Test
Armature Inductance	L_a	0.001448	H	Locked Rotor Test
Back-EMF Constant	K_b	0.04165	V·s/rad	Grey-box Estimation
Torque Constant	K_t	0.04065	N·m/A	Grey-box Estimation
Viscous Damping	B	0.19279	N·m·s/rad	Grey-box Estimation
Moment of Inertia	J	0.72762	kg·m^2	Grey-box Estimation
Total Gear Ratio	N	70	--	$N_g \times N_b$
Efficiency	η	0.836	--	Grey-box Estimation

การ Derive Transfer Function แทนค่าพารามิเตอร์จากตารางลงในสมการ State-Space ในสมการ (2.6) แล้ว Laplace Transform พร้อมกำหนดให้ Input = $V_a(s)$ และ Output = $\omega(s)$ ได้ Velocity Plant

$$G_{vel}(s) = \frac{\omega(s)}{V_a(s)} = \frac{\eta K_t N/(L_a J)}{s^2 + \left(\frac{R_a}{L_a} + \frac{B}{J}\right)s + \frac{R_a B + K_b K_t N^2}{L_a J}} \tag{3.32}$$

แทนค่าตัวเลข

$$K_{num} = \frac{\eta K_t N}{L_a J} = \frac{0.836 \times 0.04065 \times 70}{0.001448 \times 0.72762} = \frac{2.376}{0.001054} = 2254 \tag{3.33}$$

$$a_1 = \frac{R_a}{L_a} + \frac{B}{J} = \frac{1.4534}{0.001448} + \frac{0.19279}{0.72762} = 1003.5 + 0.265 = 1003.8 \tag{3.34}$$

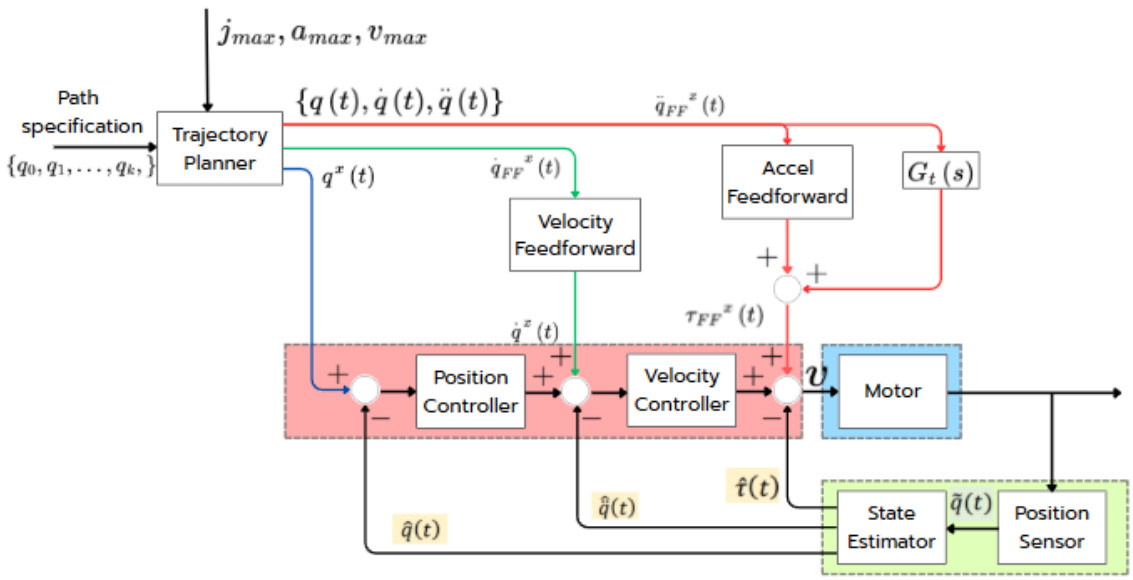
$$\begin{aligned} a_0 &= \frac{R_a B + K_b K_t N^2}{L_a J} \\ &= \frac{(1.4534)(0.19279) + (0.04165)(0.04065)(70)^2}{(0.001448)(0.72762)} \\ &= \frac{0.280 + 8.297}{0.001054} = 8136 \end{aligned} \tag{3.35}$$

$$G_{vel}(s) = \frac{2254}{s^2 + 1003.8\,s + 8136} \tag{3.36}$$

สังเกตว่า Pole ทางไฟฟ้า ($R_a/L_a \approx 1004$ rad/s) มีขนาดใหญ่กว่า Pole ทางกล ($B/J \approx 0.265$ rad/s) มาก ทำให้ Electrical Dynamics ดับลงเร็วมากและสามารถตัดทิ้งได้ เมื่อออกแบบ Velocity Controller ที่ Bandwidth $\omega_{n,vel} \ll R_a/L_a$ ตามที่จะอธิบายในหัวข้อ 3.3.6

3.3.6 การออกแบบตัวควบคุม Cascade PID

โครงสร้างตัวควบคุมใช้ Cascade PID สองวงรอบซ้อนกัน ดังแสดงในรูปที่ 3.29 โดย Velocity Inner Loop ทำหน้าที่ควบคุมความเร็วเชิงมุม ω_{arm} และ Position Outer Loop ทำหน้าที่ควบคุมตำแหน่ง θ_{arm} พร้อมด้วย Feedforward 3 ช่องทาง ได้แก่ Velocity Feedforward (K_{vff}), Acceleration Feedforward (K_{aff}) และ Torque Disturbance Feedforward (K_{tff}) เพื่อลด Tracking Error ในระหว่างการเคลื่อนที่



รูปที่ 3.29: Block Diagram ของระบบควบคุม Cascade PID แสดง Position Outer Loop, Velocity Inner Loop, Feedforward 3 ช่องทาง และ Anti-windup Back-calculation

การแปลง Requirements เป็น Desired Pole Locations

จาก P6 (Overshoot $\leq 1\%$) เลือก $\zeta = 0.9$ ซึ่งให้ Overshoot ตามสมการ

$$\%OS = e^{-\pi\zeta/\sqrt{1-\zeta^2}} \times 100 = e^{-6.486} \times 100 = 0.152\% \leq 1\% \quad \checkmark \tag{3.37}$$

จาก P5 (Settling Time ≤ 0.5 s) ใช้สูตร 2% Criterion

$$\omega_{n,pos} \geq \frac{4}{\zeta\, t_s} = \frac{4}{0.9 \times 0.5} = 8.889\,\text{rad/s} \tag{3.38}$$

Desired Poles ของ Position Loop

$$s^*_{pos} = -\zeta\,\omega_{n,pos} \pm j\,\omega_{n,pos}\sqrt{1-\zeta^2} = -8.000 \pm j3.875 \tag{3.39}$$

Velocity Inner Loop (PI)

การกำหนด Bandwidth Separation ระบบ Cascade PID นี้ใช้ Sampling Rate ที่แตกต่างกันสองระดับ ได้แก่ Position Loop ที่ $f_p = 100$ Hz ($T_p = 10$ ms) และ Velocity Loop ที่ $f_v = 1{,}000$ Hz ($T_v = 1$ ms) อัตราส่วน Sampling Rate ระหว่างสองวงรอบจึงเป็น 10 เท่าพอดี

การกำหนด Bandwidth Separation ให้สอดคล้องกับอัตราส่วนนี้ มีเหตุผลทางปฏิบัติที่ชัดเจน ได้แก่ หาก $\omega_{n,vel}$ สูงกว่า $\omega_{n,pos}$ น้อยกว่า 10 เท่า Velocity Loop จะมี Bandwidth ใกล้เคียงกับ Nyquist Frequency ของ Position Loop ($f_p/2 = 50$ Hz) ซึ่งทำให้ Position Controller มองเห็น Velocity Loop เป็น Unity Gain ได้ไม่สมบูรณ์ และเกิด Interaction ระหว่างสองวงรอบ ดังนั้นจึงกำหนด

$$\omega_{n,vel} = 10 \times \omega_{n,pos} = 10 \times 8.889 = 88.889\,\text{rad/s} \tag{3.40}$$

หมายเหตุ: การกำหนด Bandwidth Separation เป็น Heuristic ไม่ใช่ทฤษฎีเคร่งครัด วรรณกรรมทั่วไปแนะนำค่าระหว่าง 5--10 เท่า โครงงานนี้เลือก 10 เท่าเนื่องจากสอดคล้องกับอัตราส่วน Sampling Rate ของ Hardware จริง ความถูกต้องของการเลือกนี้ยืนยันได้จากผล Simulation ที่แสดงว่า Velocity Loop ตอบสนองเร็วกว่า Position Loop อย่างมีนัยสำคัญ

Desired Poles ของ Velocity Loop คือ $s^*_{vel} = -\zeta\,\omega_{n,vel} \pm j\,\omega_{n,vel}\sqrt{1-\zeta^2} = -80.000 \pm j38.747$

การ Reduce Plant จาก 2nd-order เป็น 1st-order Full Velocity Plant จากสมการ (3.36) มีสองโพล ได้แก่ Electrical Pole ที่ $s = -R_a/L_a \approx -1004$ rad/s และ Mechanical Pole ที่ $s = -B/J \approx -0.265$ rad/s เนื่องจาก Desired Bandwidth ของ Velocity Loop คือ $\omega_{n,vel} = 88.889$ rad/s ซึ่งน้อยกว่า R_a/L_a ถึง 11 เท่า

$$\frac{R_a}{L_a} = \frac{1.4534}{0.001448} = 1003.7\,\text{rad/s} \gg \omega_{n,vel} = 88.889\,\text{rad/s} \tag{3.41}$$

Electrical Dynamics จึงดับลงเร็วมากและถูกมองเป็น Algebraic Gain จากมุมมองของ Velocity Controller สามารถตั้งสมมติฐาน $L_a \approx 0$ ได้ วงจรไฟฟ้าในสมการ (2.2) ลดรูปเป็น

$$i_a \approx \frac{V_a - K_b N \omega}{R_a} \tag{3.42}$$

แทนสมการ (3.42) ลงในสมการ (2.4) แล้วจัดรูปได้ Reduced 1st-order Plant

$$\tilde{G}_v(s) = \frac{\omega(s)}{V_a(s)} = \frac{K_{eq}}{J\,s + b_{eq}} \tag{3.43}$$

โดยที่ b_{eq} คือ Viscous Damping สมมูล ซึ่งรวม Mechanical Damping B และ Back-EMF Damping $K_b K_t N^2 \eta / R_a$ เข้าด้วยกัน และ K_{eq} คือ DC Gain สมมูลของ Plant

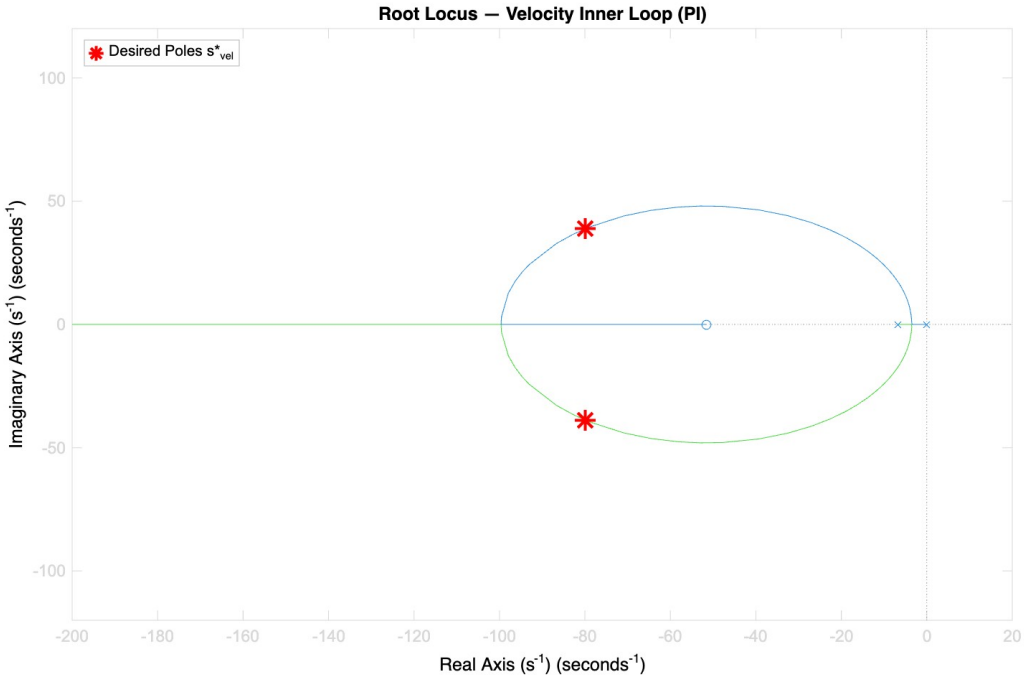
$$\begin{aligned} b_{eq} &= B + \frac{K_b K_t N^2 \eta}{R_a} \\ &= 0.19279 + \frac{0.04165 \times 0.04065 \times 70^2 \times 0.836}{1.4534} \\ &= 0.193 + 4.772 \\ &= 4.9647\,\mathrm{N \cdot m \cdot s/rad} \end{aligned} \tag{3.44}$$

$$\begin{aligned} K_{eq} &= \frac{\eta\, K_t\, N}{R_a} \\ &= \frac{0.836 \times 0.04065 \times 70}{1.4534} \\ &= 1.6367\,\mathrm{N \cdot m/V} \end{aligned} \tag{3.45}$$

สังเกตว่า Back-EMF Damping (4.772) มีขนาดใหญ่กว่า Mechanical Damping (0.193) ถึง 25 เท่า แสดงว่า Damping หลักของระบบนี้มาจาก Back-EMF ไม่ใช่จาก Bearing Friction

แทนค่าตัวเลขได้ Reduced Plant

$$\tilde{G}_v(s) = \frac{1.6367}{0.7276\,s + 4.9647} \tag{3.46}$$



รูปที่ 3.30: Root Locus ของ Velocity Inner Loop (PI + Reduced Plant) แสดง Desired Poles $s^*_{vel} = -80.000 \pm j38.747$ (เครื่องหมาย $\times$ สีแดง) บนเส้น Damping Ratio $\zeta = 0.9$ (เส้นประสีน้ำเงิน)

การคำนวณ Velocity PI Gains ใช้ Pole Placement บน Reduced Plant โดย Characteristic Equation ของ Closed-loop Velocity Loop ที่ต้องการคือ

$$s^2 + 2\zeta\,\omega_{n,vel}\,s + \omega_{n,vel}^2 = s^2 + 160.000\,s + 7901.235 \tag{3.47}$$

เทียบสัมประสิทธิ์กับ Closed-loop Characteristic Equation ของ PI + 1st-order Plant ได้

$$K_{p,vel} = \frac{2\zeta\,\omega_{n,vel}\,J - b_{eq}}{K_{eq}} = \frac{2 \times 0.9 \times 88.889 \times 0.72762 - 4.9647}{1.6367} = 68.940 \tag{3.48}$$

$$K_{i,vel} = \frac{\omega_{n,vel}^2\,J}{K_{eq}} = \frac{(88.889)^2 \times 0.72762}{1.6367} = 3512.963 \tag{3.49}$$

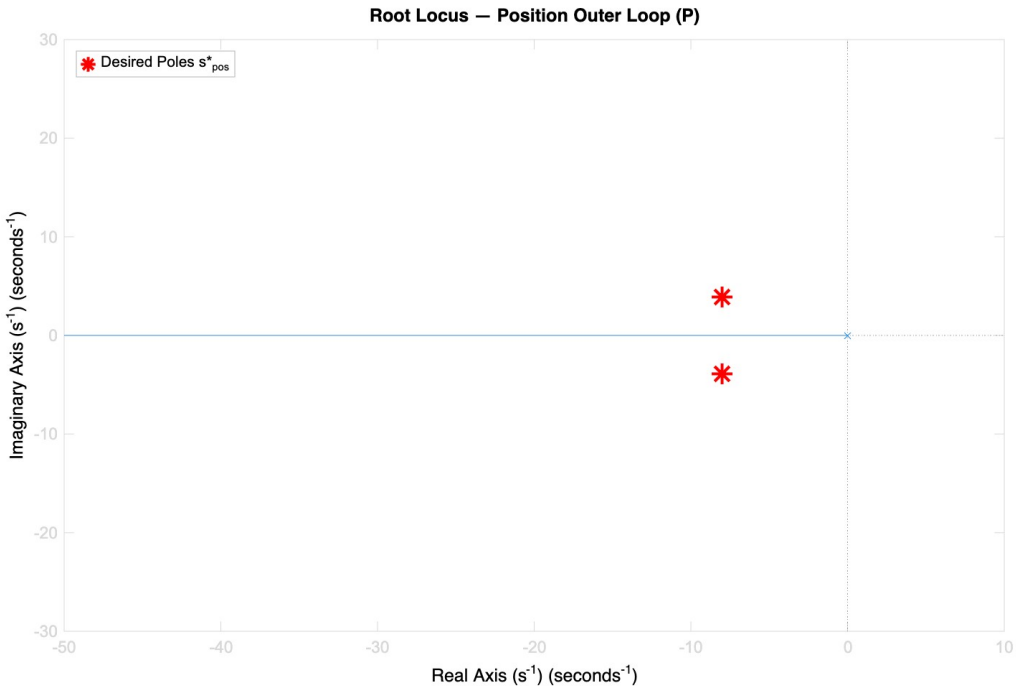
Position Outer Loop (P)

เมื่อ Velocity Loop ปิดแล้วและ Bandwidth Separation เป็น 10 เท่า Closed-loop Velocity Transfer Function มีค่าใกล้เคียง Unity ในย่านที่ Position Loop ทำงาน Plant สำหรับ Position Loop จึงประมาณได้เป็น Integrator

$$G_{pos,plant}(s) \approx \frac{1}{s} \tag{3.50}$$

ใช้ P Controller เพียงอย่างเดียว เนื่องจาก Integrator ใน Plant ทำให้ Steady-State Error ต่อ Step Position Command เป็นศูนย์โดยธรรมชาติ โดยไม่ต้องการ Integral Term เพิ่มเติม

$$K_{p,pos} = \omega_{n,pos} = 8.889\,\mathrm{s}^{-1} \quad \Rightarrow \quad t_s = \frac{4}{K_{p,pos}} = 0.450\,\mathrm{s} \leq 0.5\,\mathrm{s} \quad \checkmark \tag{3.51}$$



รูปที่ 3.31: Root Locus ของ Position Outer Loop (P + Integrator Plant) แสดง Desired Poles $s^*_{pos} = -8.000 \pm j3.875$ (เครื่องหมาย $\times$ สีแดง) บนเส้น Damping Ratio $\zeta = 0.9$ (เส้นประสีน้ำเงิน)

Anti-windup และ Feedforward

Anti-windup Back-calculation Velocity Loop มี $K_{i,vel}$ ขนาดใหญ่ ทำให้เมื่อเกิด Large Setpoint Change Integrator State สะสมค่าสูงก่อนที่แรงดัน Output จะถึง Saturation Limit ± 24 V (Integral Windup) Anti-windup แบบ Back-calculation แก้ปัญหานี้ โดยป้อนกลับส่วนต่างระหว่าง Unsaturated และ Saturated Output มาลด Integrator State ตามที่อธิบายในสมการ (2.9)

$$K_{t,aw} = \frac{K_{i,vel}}{K_{p,vel}} = \frac{3512.963}{68.940} = 50.958\,\mathrm{s}^{-1} \tag{3.52}$$

Feedforward Gains Feedforward ช่วยลด Tracking Error ระหว่างการเคลื่อนที่ โดยไม่ต้องรอ Error สะสมก่อน ระบบนี้มี Feedforward 3 ช่องทาง

Velocity Feedforward (K_{vff}): ในสภาวะ Steady-State ที่ความเร็วคงที่ ($\dot{\omega} = 0$, $\tau_d = 0$) แทนค่าลงในสมการ (2.4) และสมการ (2.2) ได้แรงดันที่จำเป็น

$$K_{vff} = \frac{R_a\,B}{\eta\,N\,K_t} + K_b\,N = \frac{1.4534 \times 0.19279}{0.836 \times 70 \times 0.04065} + 0.04165 \times 70 = 0.117 + 2.916 = 3.033\,\text{V/(rad/s)} \tag{3.53}$$

Acceleration Feedforward (K_{aff}): ในสภาวะที่มีความเร่ง $\dot{\omega} \neq 0$ แรงบิดส่วนที่ใช้เร่งมวล $J\dot{\omega}$ ต้องการแรงดันเพิ่มเติม

$$K_{aff} = \frac{R_a\,J}{\eta\,N\,K_t} = \frac{1.4534 \times 0.72762}{0.836 \times 70 \times 0.04065} = \frac{1.0577}{2.376} = 0.445\,\text{V/(rad/s}^2\text{)} \tag{3.54}$$

Torque Disturbance Feedforward (K_{tff}): หาก Kalman Filter ประมาณค่า Disturbance Torque $\hat{\tau}_d$ ได้แล้ว สามารถชดเชยล่วงหน้าได้โดย

$$\begin{aligned} K_{tff} &= \frac{R_a}{\eta\,N\,K_t} \\ &= \frac{1.4534}{0.836 \times 70 \times 0.04065} \\ &= \frac{1.4534}{2.376} \\ &= 0.612\,\mathrm{V/(N \cdot m)} \end{aligned} \tag{3.55}$$

โดย K_{tff} เปิดใช้งานเมื่อ Kalman Filter Converge แล้วเท่านั้น ตามที่อธิบายในหัวข้อ 2.2.3

ตารางที่ 3.19: สรุปค่า PID Gain และ Feedforward ของระบบควบคุม

Loop	Gain	สัญลักษณ์	ค่า	หน่วย
Velocity Inner (PI)	Proportional	$K_{p,vel}$	68.940	V/(rad/s)
	Integral	$K_{i,vel}$	3512.963	V/rad
Anti-windup	Back-calc	$K_{t,aw}$	50.958	s^{-1}
Position Outer (P)	Proportional	$K_{p,pos}$	8.889	(rad/s)/rad
Feedforward	Velocity FF	K_{vff}	3.033	V/(rad/s)
	Acceleration FF	K_{aff}	0.445	V/(rad/s^2)
	Disturbance FF	K_{tff}	0.612	V/(N·m)

หมายเหตุ: K_{vff}, K_{aff}, K_{tff} ปิดระหว่าง Commissioning
จนกว่า SysID uncertainty $< 10\%$ และ Kalman Filter Converge

Hardware Commissioning และค่า Gain จริง

ค่า Gain ที่ได้จาก Pole Placement ในสมการ (3.48), (3.49) และ (3.51) คำนวณภายใต้สมมติฐานว่าแบบจำลอง Plant มีความแม่นยำสมบูรณ์ และ Output ของ Controller คือแรงดัน V_a ในหน่วย Volt อย่างไรก็ตาม Firmware จริงบน STM32G474RE ส่งคำสั่งในรูป PWM Count (0 ถึง ±50) ผ่านฟังก์ชัน `Motor_SetPWM()` โดยมี Scale Factor

$$\text{Scale} = \frac{V_{supply}}{PWM_{max}} = \frac{24}{50} = 0.48\,\text{V/PWM} \tag{3.56}$$

ดังนั้น Theoretical Gains ที่แปลงหน่วยเป็น PWM Unit แล้วคือ

$$K_{p,vel}^{theory,hw} = \frac{K_{p,vel}^{theory}}{\text{Scale}} = \frac{68.940}{0.48} \approx 143.6\,\text{PWM/(rad/s)} \tag{3.57}$$

ซึ่งยังคงสูงกว่าค่าที่ใช้งานได้จริงอย่างมีนัยสำคัญ เนื่องจากปัจจัยหลักสามประการ ได้แก่

- **Model Uncertainty:** พารามิเตอร์ $J = 0.728\,\text{kg·m}^2$ และ $B = 0.193\,\text{N·m·s/rad}$ ประมาณจาก Chirp Excitation ที่ความถี่ต่ำ (0.1--2 Hz) ซึ่งอาจไม่สะท้อน Dynamic ของระบบที่ Bandwidth สูงกว่าได้อย่างแม่นยำ
- **Nonlinearity:** แบบจำลอง LTI ละทิ้ง Static Friction ของ Bearing และ Belt Drive ซึ่งมีผลต่อ Gain ที่เหมาะสมในสภาวะ Low-Velocity
- **PWM Saturation:** Gain สูงทำให้ Controller Output Saturate ที่ ±50 PWM ในช่วง Transient เกือบตลอดเวลา ทำให้ระบบไม่ทำงานในย่าน Linear ตามที่ Pole Placement สมมติ

ด้วยเหตุนี้ Theoretical Gains จึงใช้เป็น **Order-of-Magnitude Reference** และ **Starting Point** สำหรับ Hardware Commissioning เท่านั้น ไม่ใช่ค่าที่นำไปใช้งานโดยตรง

ขั้นตอน Hardware Commissioning ดำเนินการ Tune Gains บน Hardware จริงตามลำดับดังนี้

1. **Velocity Loop ก่อน:** ปิด Position Loop ชั่วคราว เพิ่ม $K_{p,vel}$ ทีละน้อยจนระบบเริ่ม Oscillate บันทึก $K_{p,crit}$ แล้วลดลงเหลือประมาณ $0.5 \times K_{p,crit}$ จากนั้นเพิ่ม $K_{i,vel}$ ช้าๆ จนกำจัด Steady-State Error ได้
2. **Position Loop ถัดไป:** เปิด Position Loop แล้วเพิ่ม $K_{p,pos}$ ทีละน้อย จนได้ Response ที่เร็วเพียงพอโดยไม่มี Overshoot จากนั้นปรับ $K_{d,pos}$ เพื่อ Damp การสั่น และ $K_{i,pos}$ เพื่อกำจัด Steady-State Error จาก Friction
3. **ตรวจสอบ Requirements:** วัด Settling Time และ Overshoot จากการทดสอบ Step Response บน Hardware จริง เทียบกับ P5 ($t_s \leq 0.5$ s) และ P6 (%OS $\leq 1\%$)

3.3.7 ค่า Gain ที่ใช้งานจริง (Hardware-Tuned)

ตารางที่ 3.20: เปรียบเทียบค่า PID Gain จาก Theoretical Design และ Hardware Commissioning

Loop	สัญลักษณ์	Theoretical	Hardware	หน่วย (Theory)	หน่วย (HW)
Velocity (PI)	$K_{p,vel}$	68.940	1.58	V/(rad/s)	PWM/(rad/s)
	$K_{i,vel}$	3512.963	0.5	V/rad	PWM/rad
Position (PID)	$K_{p,pos}$	8.889	3.00	(rad/s)/rad	(rad/s)/rad
	$K_{i,pos}$	--	0.02	--	(rad/s)/(rad·s)
	$K_{d,pos}$	--	0	--	(rad/s)·s/rad
Anti-windup	$K_{t,aw}$	50.958	--	s^{-1}	s^{-1}
Feedforward	K_{vff}	3.033	3.03	V/(rad/s)	PWM/(rad/s)
	K_{aff}	0.445	0.10	V/(rad/s^2)	PWM/(rad/s^2)
	K_{tff}	0.612	3.25	V/(N·m)	PWM/(N·m)

ตารางที่ 3.21: ผลการทดสอบ Step Response บน Hardware เทียบกับ Requirements

Metric	ผล Hardware	Requirement	หน่วย	สถานะ
Settling Time	2.11	≤ 0.5	s	ไม่ผ่าน
Overshoot	0.00	≤ 1	%	ผ่าน
Steady-State Error	0.000	--	deg	--

การ Discretize สู่ Z-Domain (Tustin Transform)

การ Implement ตัวควบคุม Cascade PID บน STM32G474RE จำเป็นต้องแปลง Continuous-time Controller ให้เป็น Discrete-time ก่อน โดยใช้วิธี Tustin (Bilinear Transform)

$$s \longleftarrow \frac{2}{T} \cdot \frac{z-1}{z+1} \tag{3.58}$$

เหตุผลที่เลือก Tustin แทน Forward Euler ได้แก่ Tustin Map โพลทุกตัวในครึ่งซ้ายของระนาบ s ให้อยู่ภายใน Unit Circle ของระนาบ z อย่างสมบูรณ์ ในขณะที่ Forward Euler อาจทำให้ Discrete System ไม่เสถียรแม้ Continuous System จะเสถียร โดยเฉพาะเมื่อ K_i/f_s มีขนาดใหญ่ ตามที่อธิบายในหัวข้อ 2.2.1

Velocity Inner Loop — PI (Tustin) Continuous-time PI Controller ของ Velocity Loop คือ

$$C_v(s) = K_{p,vel} + \frac{K_{i,vel}}{s} \tag{3.59}$$

แทนสมการ (3.58) ลงในสมการ (3.59) ที่ Sample Time $T_v = 1$ ms ได้

$$C_v(z) = K_{p,vel} + K_{i,vel} \cdot \frac{T_v}{2} \cdot \frac{z+1}{z-1} \tag{3.60}$$

จัดรูปให้ได้ Difference Equation โดยกำหนด $u_v[k]$ เป็น PWM Output และ $e_v[k] = \omega_{sp}[k] - \omega_{act}[k]$ เป็น Velocity Error

$$u_v[k] = u_v[k-1] + \underbrace{\left(K_{p,vel} + \frac{K_{i,vel}\,T_v}{2}\right)}_{c_1} e_v[k] + \underbrace{\left(-K_{p,vel} + \frac{K_{i,vel}\,T_v}{2}\right)}_{c_2} e_v[k-1] \tag{3.61}$$

โดยที่

$$c_1 = K_{p,vel} + \frac{K_{i,vel}\,T_v}{2} = 68.940 + \frac{3512.963 \times 0.001}{2} = 68.940 + 1.756 = 70.696 \tag{3.62}$$

$$c_2 = -K_{p,vel} + \frac{K_{i,vel}\,T_v}{2} = -68.940 + 1.756 = -67.184 \tag{3.63}$$

Difference Equation นี้ implement ได้โดยตรงใน Firmware โดยเก็บเพียง $u_v[k-1]$ และ $e_v[k-1]$ เป็น Static Variable ซึ่งมี Memory Footprint เพียง 2 ตัวแปร Float (8 bytes) ต่อ Loop

Position Outer Loop — PID (Tustin, Derivative on Measurement) Continuous-time PID Controller ของ Position Loop คือ

$$C_p(s) = K_{p,pos} + \frac{K_{i,pos}}{s} + K_{d,pos}\,s \tag{3.64}$$

แทน Tustin Transform ที่ $T_p = 10$ ms และใช้ Derivative on Measurement (แทน Derivative on Error เพื่อป้องกัน Derivative Kick เมื่อ Setpoint เปลี่ยนแบบ Step)

$$\text{D-term} = -K_{d,pos}\,\frac{d\theta_{act}}{dt} \tag{3.65}$$

Difference Equation ของ Position PID คือ

$$\begin{aligned} u_p[k] &= K_{p,pos}\,e_p[k] \\ &\quad + \frac{K_{i,pos}\,T_p}{2}\left(e_p[k] + e_p[k-1]\right) + I_p[k-1] \\ &\quad - K_{d,pos}\,\frac{\theta_{act}[k] - \theta_{act}[k-1]}{T_p} \end{aligned} \tag{3.66}$$

โดยที่ $I_p[k]$ คือ Integral State ที่ Update ตาม

$$I_p[k] = I_p[k-1] + \frac{K_{i,pos}\,T_p}{2}\left(e_p[k] + e_p[k-1]\right) \tag{3.67}$$

และ $u_p[k]$ คือ Velocity Setpoint ที่ส่งเข้า Velocity Inner Loop ซึ่ง Clamp ไว้ที่ $\pm v_{max} = \pm 7.304$ rad/s

Anti-windup ใน Discrete-time ทั้งสอง Loop ใช้ Conditional Integration เพื่อป้องกัน Integral Windup โดย

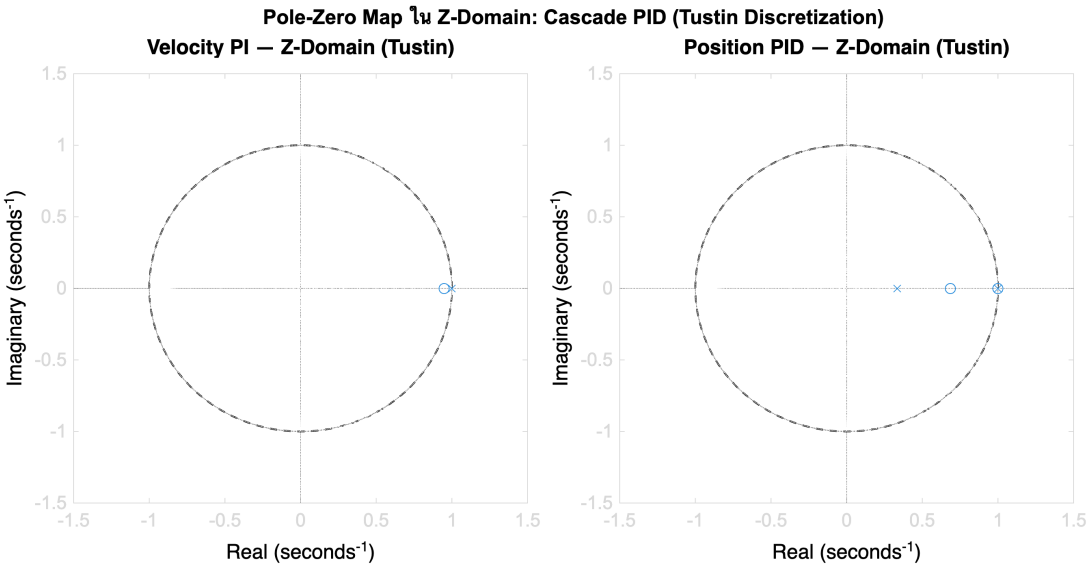
- **Velocity Loop:** หยุด Integrate เมื่อ Feedforward Output ใกล้ Saturation Limit ($|u_{ff}| \geq$

$PWM_{max} - 1$) เนื่องจากสภาวะดังกล่าว PID Output จะถูก Clamp อยู่ดีและ Integral State ที่สะสมต่อไปจะไม่มีผลต่อ Output จริง

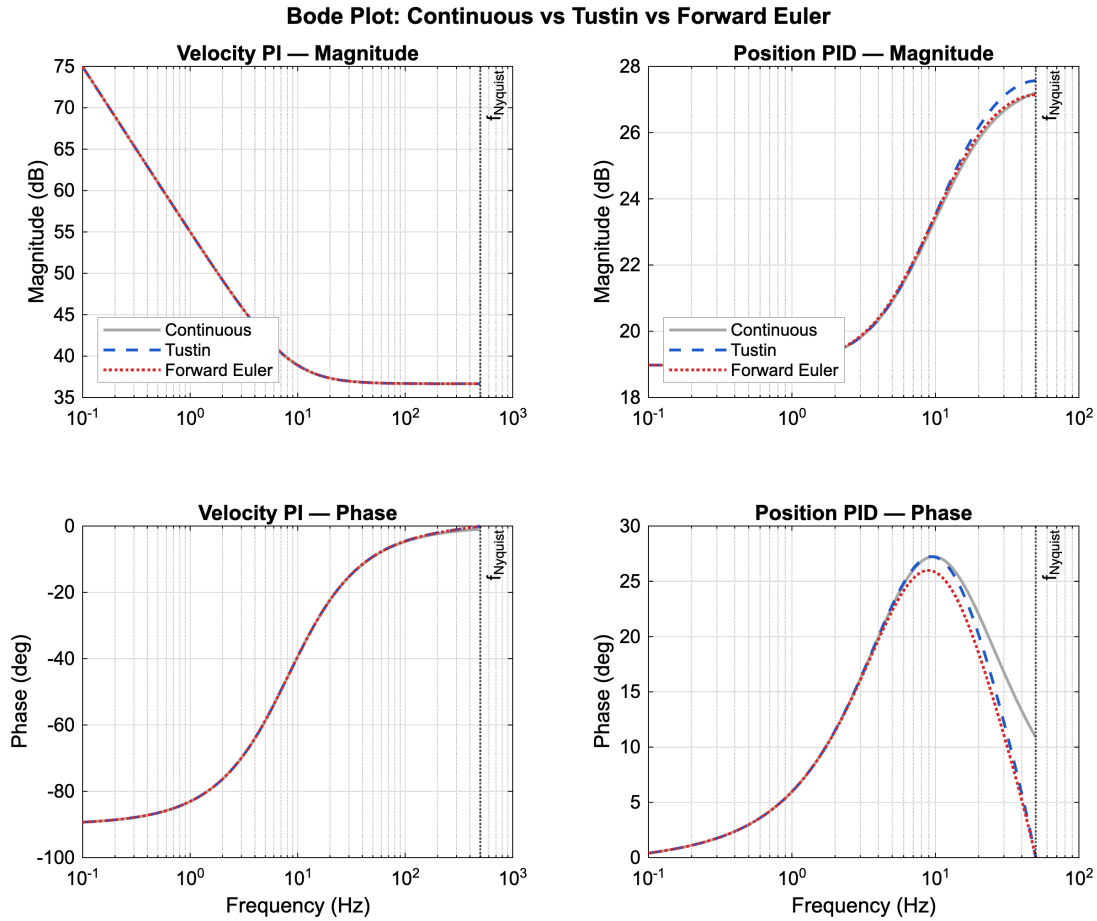
- **Position Loop:** หยุด Integrate เมื่อ Position Error อยู่ภายใน Deadband ($|e_p| < \delta_{dead}$) ป้องกัน Windup ขณะจอดนิ่ง

ตารางที่ 3.22: สรุปค่าสัมประสิทธิ์ Discrete-time Controller (Theoretical Gains, หน่วย V)

Loop	พารามิเตอร์	สัญลักษณ์	ค่า	หน่วย
Velocity PI	Sample Time	T_v	0.001	s
	Forward Coeff	c_1	70.696	V/(rad/s)
	Backward Coeff	c_2	-67.184	V/(rad/s)
	Integral Clamp	$I_{max,v}$	`PID_SPEED_IMAX`	V
Position PID	Sample Time	T_p	0.010	s
	Proportional	$K_{p,pos}$	3.00	(rad/s)/rad
	Integral Step	$K_{i,pos}T_p/2$	0.0001	(rad/s)/rad
	Derivative	$K_{d,pos}/T_p$	0	(rad/s)·s/rad



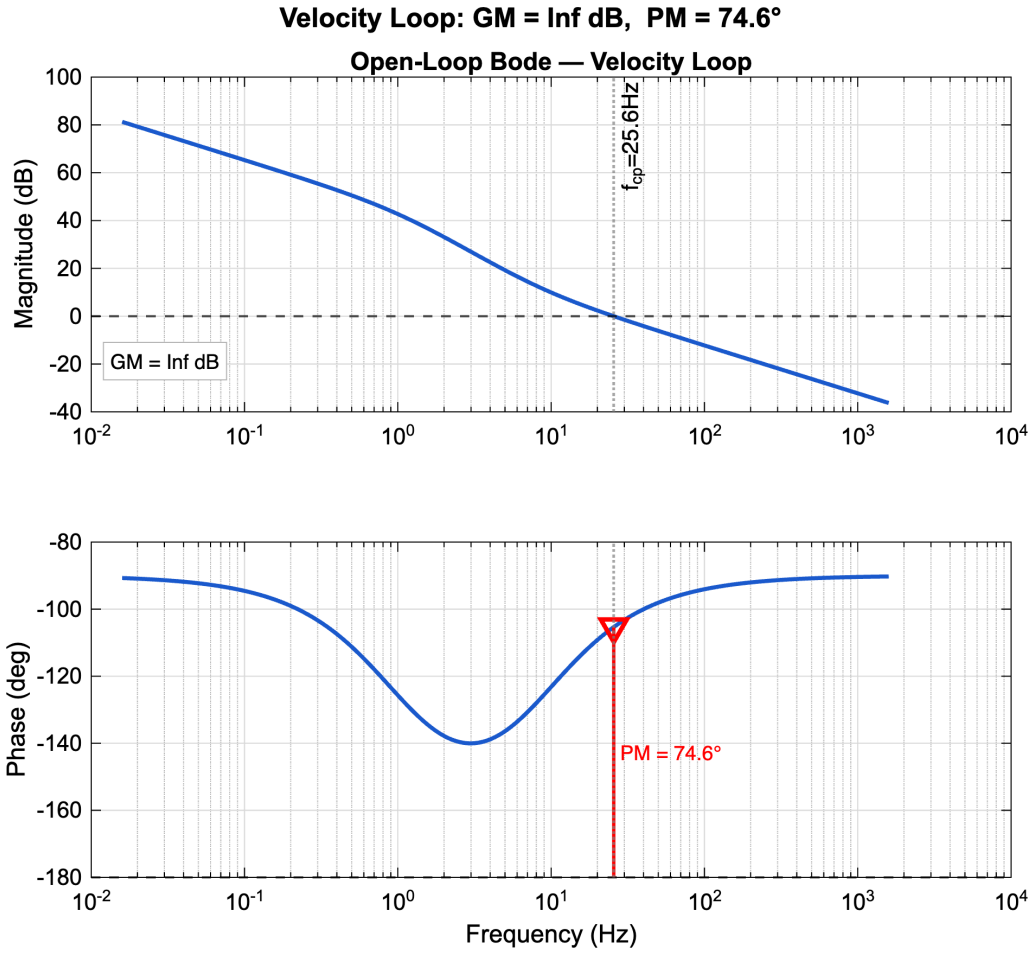
รูปที่ 3.32: Pole-Zero Map ใน Z-Domain ของ Velocity PI (ซ้าย) และ Position PID (ขวา) หลัง Tustin Discretization Poles และ Zeros ทั้งหมดอยู่ภายใน Unit Circle ยืนยันความเสถียรของ Discrete-time Controller



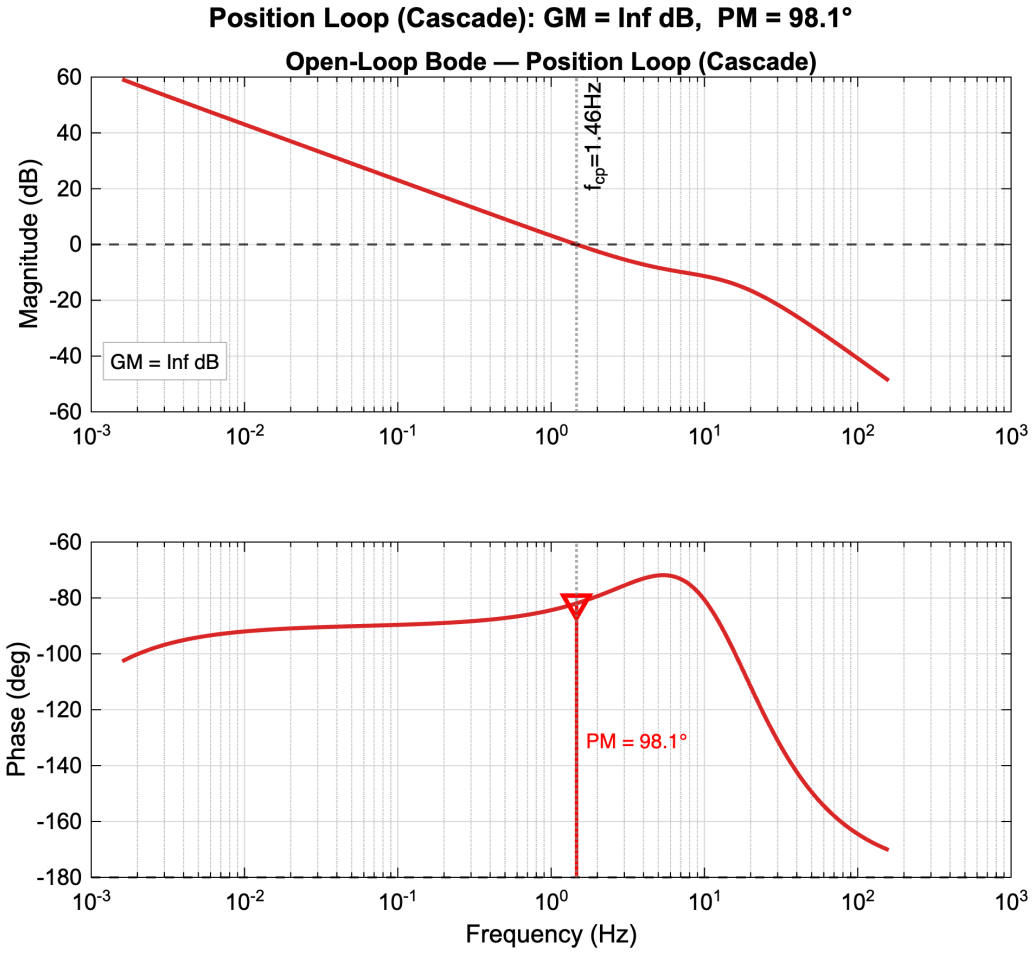
รูปที่ 3.33: Bode Plot เปรียบเทียบ Continuous-time (สีเทา), Tustin (สีน้ำเงิน) และ Forward Euler (สีแดง) แสดงว่า Tustin ให้ Frequency Response ใกล้เคียง Continuous ในย่าน Passband ในขณะที่ Forward Euler มี Phase Error สูงกว่าเมื่อความถี่เข้าใกล้ $f_{Nyquist}$

การวิเคราะห์เสถียรภาพในโดเมนความถี่

การวิเคราะห์เสถียรภาพของ Cascade PID ดำเนินการใน Frequency Domain โดยใช้ Open-loop Bode Plot เพื่อหา Gain Margin (GM) และ Phase Margin (PM) ซึ่งเป็น Stability Margin มาตรฐานสำหรับ Linear Feedback System เกณฑ์ขั้นต่ำที่ยอมรับได้ทั่วไปคือ GM > 6 dB และ PM $> 30°$



รูปที่ 3.34: Open-Loop Bode Plot ของ Velocity Inner Loop แสดง Phase Margin ที่ Gain Crossover Frequency ยืนยันความเสถียรของ Velocity Loop

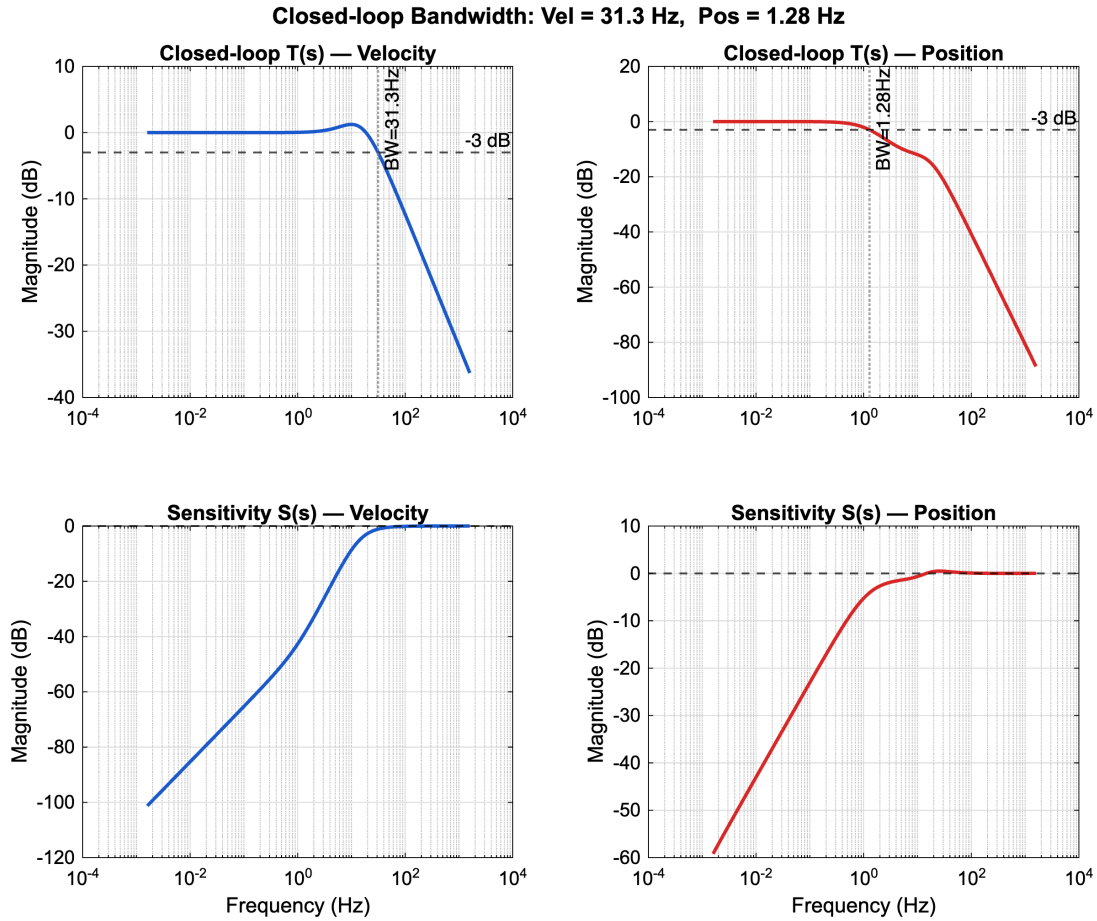


รูปที่ 3.35: Open-Loop Bode Plot ของ Position Outer Loop (Cascade) แสดง Gain Margin และ Phase Margin ของระบบ Cascade รวมทั้งสอง Loop

ตารางที่ 3.23: สรุป Stability Margin ของ Cascade PID

Loop	GM (dB)	PM (deg)	f_{cp} (Hz)	สถานะ
Velocity Inner	∞	74.64	25.58	**ผ่าน**
Position Outer	∞	98.07	1.46	**ผ่าน**

เกณฑ์: GM > 6 dB และ PM $> 30°$



รูปที่ 3.36: Closed-loop Frequency Response แสดง Complementary Sensitivity $T(s)$ (บน) และ Sensitivity $S(s)$ (ล่าง) ของ Velocity Loop (ซ้าย) และ Position Loop (ขวา) เส้นประแสดงจุด -3 dB Bandwidth ของแต่ละ Loop

Gain Margin มีค่าเป็น ∞ ในทั้งสอง Loop เนื่องจาก Phase ของ Open-loop Transfer Function ไม่เคยลดลงถึง $-180°$ ในย่านความถี่ใด ซึ่งเป็นลักษณะปกติของระบบ PI และ PID ที่มี Integrator เพียงหนึ่งตัวใน Loop ดังนั้น Phase Margin จึงเป็น Stability Indicator หลักของระบบนี้

Phase Margin ของ Velocity Loop ($74.6°$) และ Position Loop ($98.1°$) สูงกว่าเกณฑ์ขั้นต่ำ $30°$ อย่างมีนัยสำคัญ แสดงว่าระบบมี Stability Margin เพียงพอ และสามารถรองรับ Plant Model Uncertainty ได้ในระดับสูง

Gain Crossover Frequency ของ Velocity Loop ที่ $f_{cp} = 25.58$ Hz ยืนยัน Bandwidth Separation จาก Position Loop ($f_{cp} = 1.46$ Hz) ที่อัตราส่วน $25.58/1.46 \approx 17.5$ เท่า ซึ่งเกินกว่าเกณฑ์ 10 เท่าที่ออกแบบไว้ตามสมการ (3.40)

3.3.8 การออกแบบ ZVD Input Shaper

แบบจำลอง Connection Rod และเหตุผลที่ต้องใช้ Input Shaping

Connection Rod มีมวล $m_{rod} = 164.08$ g และความยาว $L_{rod} = 100$ mm จำลองเป็น Simple Pendulum ที่ถูกกระตุ้นด้วย Angular Acceleration ของแขนกล α_{arm} ตามที่อธิบายในหัวข้อ 2.3.3 สมการการเคลื่อนที่ในรูปแบบ Standard 2nd-order ได้จากสมการ (2.27)

$$\ddot{\phi} + 2\zeta_{rod}\,\omega_{n,rod}\,\dot{\phi} + \omega_{n,rod}^2\,\phi = \underbrace{-\frac{3\,r_{arm}}{2\,L_{rod}}}_{\Gamma}\,\alpha_{arm} \tag{3.68}$$

โดยที่ $\Gamma = -3 \times 0.5/(2 \times 0.1) = -7.5$ คือ Forcing Gain และความถี่ธรรมชาติคำนวณได้จากสมการ (2.28)

$$\omega_{n,rod} = \sqrt{\frac{3g}{2L_{rod}}} = \sqrt{\frac{3 \times 9.81}{2 \times 0.1}} = 12.131\,\text{rad/s} \qquad (f_n = 1.931\,\text{Hz}) \tag{3.69}$$

ค่า Damping Ratio $\zeta_{rod} = 0.041$ ใช้เป็น Conservative Assumption สำหรับ Lightly Damped Mechanical System ซึ่งให้ Settling Time ต่อ Move ที่

$$t_{settle,rod} \approx \frac{4}{\zeta_{rod}\,\omega_{n,rod}} = \frac{4}{0.041 \times 12.131} = 8.04\,\text{s/move} \tag{3.70}$$

หากระบบมี 9 Moves ต่อ Cycle และต้องรอให้ Rod หยุดแกว่งก่อนทุก Move Cycle Time รวมจะเกิน $9 \times 8.04 = 72.4$ s ซึ่งเกินข้อกำหนด P1 (≤ 35 s) อย่างมาก การใช้ Input Shaper จึงเป็นข้อกำหนดที่ขาดไม่ได้

การออกแบบ ZVD Input Shaper (3-Impulse)

ระบบเลือกใช้ ZVD (Zero-Vibration-Derivative) Shaper แบบ 3-Impulse แทน ZV แบบ 2-Impulse เนื่องจาก ZVD มีเงื่อนไขเพิ่มเติม ว่า Derivative ของ Residual Vibration เทียบกับ ω_n ต้องเป็นศูนย์ด้วย ทำให้ทนทานต่อความไม่แน่นอนของ ω_n ได้สูงกว่า ดังที่จะเห็นในรูปที่ 3.38

การคำนวณพารามิเตอร์ ZVD กำหนดค่า Decay Factor

$$K = e^{-\zeta_{rod}\pi/\sqrt{1-\zeta_{rod}^2}} = e^{-0.041\pi/\sqrt{1-0.041^2}} = e^{-0.1288} = 0.8790 \tag{3.71}$$

แอมพลิจูดของ Impulse ทั้งสามคำนวณจากสมการ (2.11)

$$A_1 = \frac{1}{(1+K)^2} = \frac{1}{(1.8790)^2} = 0.2832 \tag{3.72}$$

$$A_2 = \frac{2K}{(1+K)^2} = \frac{2 \times 0.8790}{(1.8790)^2} = 0.4979 \tag{3.73}$$

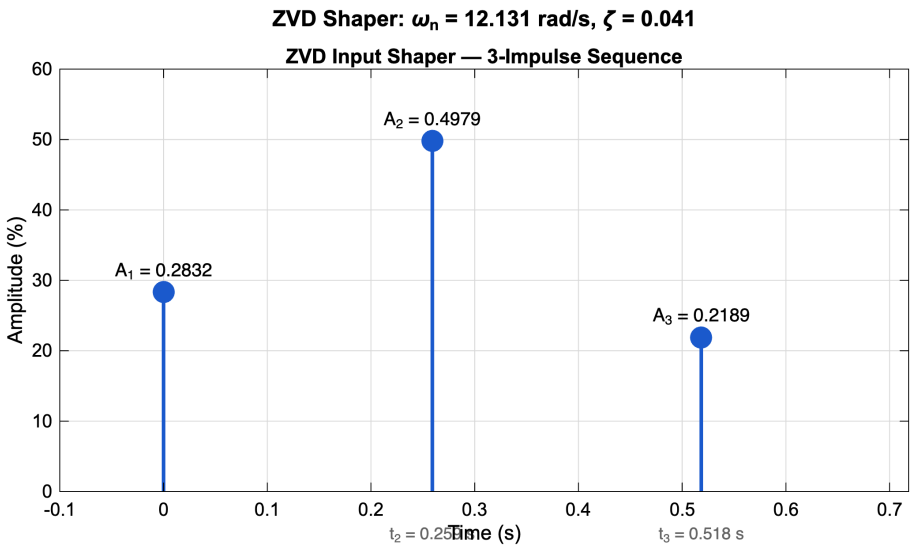
$$A_3 = \frac{K^2}{(1+K)^2} = \frac{(0.8790)^2}{(1.8790)^2} = 0.2189 \tag{3.74}$$

ตรวจสอบ: $A_1 + A_2 + A_3 = 0.2832 + 0.4979 + 0.2189 = 1.0000$ ✓

Impulse Delay คำนวณจาก Half-Period ของ Damped Oscillation

$$T_d = \frac{\pi}{\omega_{n,rod}\sqrt{1-\zeta_{rod}^2}} = \frac{\pi}{12.131 \times \sqrt{1-0.041^2}} = \frac{\pi}{12.121} = 0.2592\,\text{s} \tag{3.75}$$

$$t_2 = T_d = 0.2592\,\text{s} \quad (26 \text{ steps @ 100 Hz}), \qquad t_3 = 2T_d = 0.5184\,\text{s} \quad (52 \text{ steps @ 100 Hz}) \tag{3.76}$$



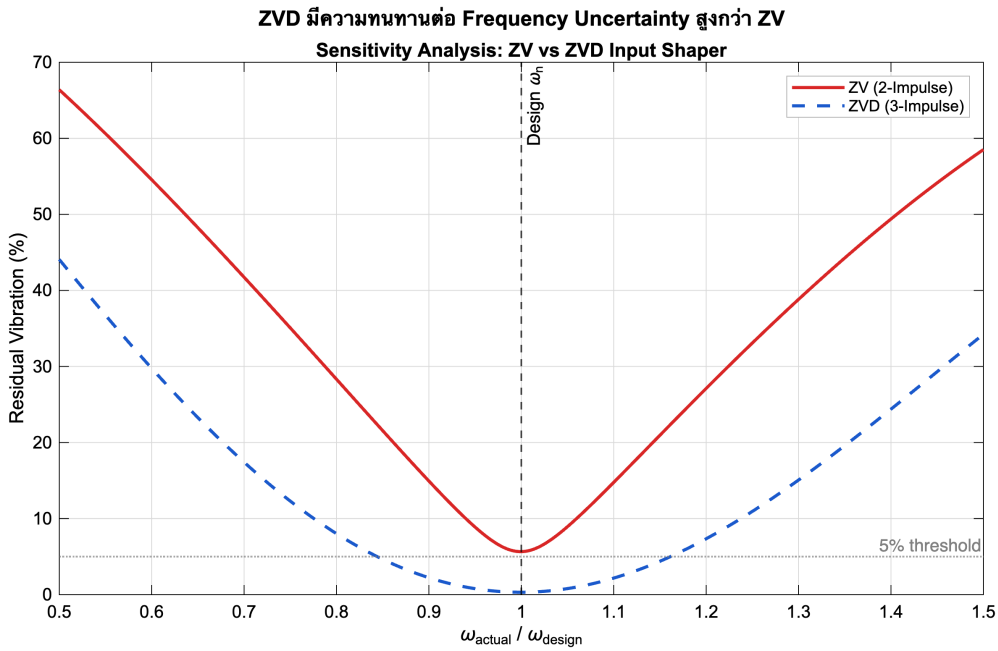
รูปที่ 3.37: ZVD 3-Impulse Sequence แสดงแอมพลิจูด A_1, A_2, A_3 และเวลาหน่วง $t_2 = 0.259$ s (26 steps) และ $t_3 = 0.518$ s (52 steps) ที่ Outer Loop 100 Hz

ตารางที่ 3.24: พารามิเตอร์ ZVD Input Shaper ที่ใช้ใน Firmware

พารามิเตอร์	สัญลักษณ์	ค่า	หมายเหตุ
Natural Frequency	$\omega_{n,rod}$	12.131 rad/s	คำนวณ จากสมการ (3.69)
Damping Ratio	ζ_{rod}	0.041	Conservative Assumption
Decay Factor	K	0.8790	สมการ (3.71)
Impulse 1	A_1	0.2832	สมการ (3.72)
Impulse 2	A_2	0.4979	สมการ (3.73)
Impulse 3	A_3	0.2189	สมการ (3.74)
Delay 2	t_2	0.2592 s (26 steps)	สมการ (3.76)
Delay 3	t_3	0.5184 s (52 steps)	สมการ (3.76)

การวิเคราะห์ความทนทานต่อความไม่แน่นอนของความถี่

ในการใช้งานจริง ค่า $\omega_{n,rod}$ อาจเปลี่ยนแปลงได้ เนื่องจากน้ำหนักของ Connection Rod ที่แตกต่างกัน หรือความยาวของ Rod ที่ไม่แน่นอน รูปที่ 3.38 แสดง Residual Vibration เมื่อ ω_{actual} ต่างจาก ω_{design} สำหรับ ZV (2-Impulse) และ ZVD (3-Impulse)

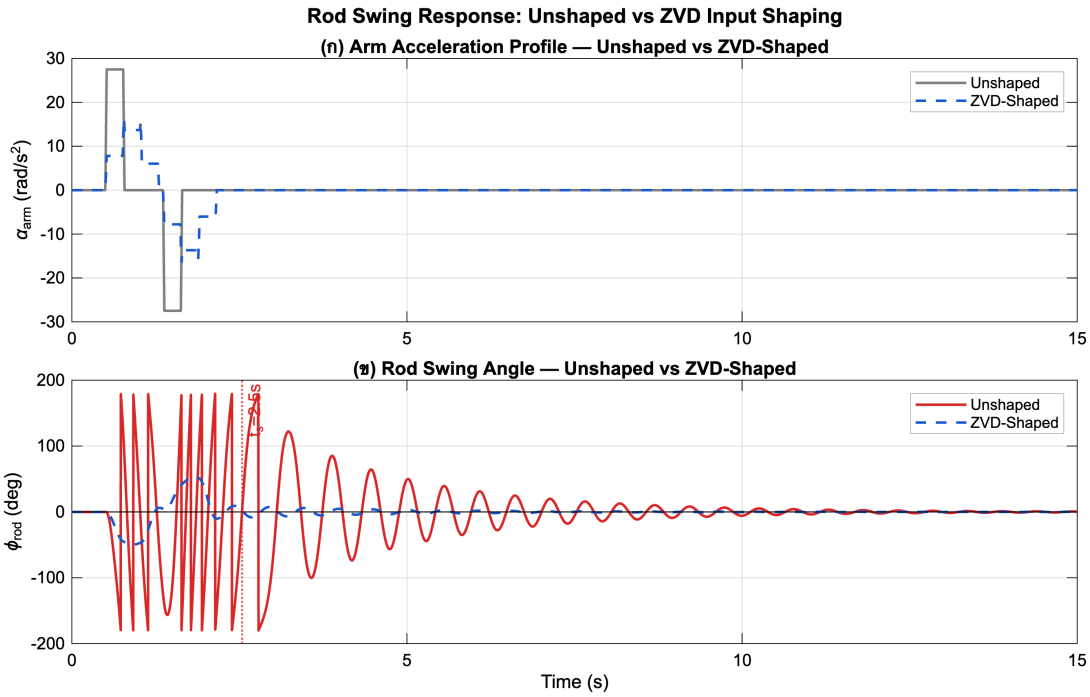


รูปที่ 3.38: Sensitivity Analysis เปรียบเทียบ ZV (สีแดง) และ ZVD (สีน้ำเงิน) แสดง Residual Vibration เมื่อ $\omega_{actual}/\omega_{design}$ เบี่ยงจาก 1.0 ZVD รักษา Residual Vibration ต่ำกว่า 5% ในย่านกว้างกว่า ZV อย่างมีนัยสำคัญ

จากกราฟ ZVD รักษา Residual Vibration ต่ำกว่า 5% ในย่าน $\omega_{actual}/\omega_{design} \in [0.88, 1.12]$ ในขณะที่ ZV รักษาได้เพียง $[0.94, 1.06]$ แสดงว่า ZVD ทนทานต่อความไม่แน่นอนของ ω_n ได้กว้างกว่าประมาณ 2 เท่า

ผลการ Simulation: Unshaped กับ ZVD-Shaped

รูปที่ 3.39 แสดงผลการ Simulation ของการแกว่ง Rod ในสภาวะ Worst-case (Move $360°$ ด้วย S-curve ที่ $v_{max} = 7.304$ rad/s, $a_{max} = 27.49$ rad/s^2, $j_{max} = 1400$ rad/s^3)



รูปที่ 3.39: (ก) S-curve Acceleration Profile ของแขนกล Unshaped (สีเทา) และ ZVD-Shaped (สีน้ำเงิน) (ข) การแกว่งของ Connection Rod ϕ_{rod} แสดง Settling Time ของ Unshaped และการยกเลิกการแกว่งของ ZVD-Shaped สำหรับ Worst-case Move $360°$

ตารางที่ 3.25: สรุปผลการ Simulation เปรียบเทียบ Unshaped กับ ZVD-Shaped (Worst-case Move $360°$)

Metric	Unshaped	ZVD-Shaped	ผลต่าง
Rod Peak Deflection	$179.95°$	$53.52°$	-70.3%
Rod Settling Time	0.21 s	0.44 s	--
Move Time	1.146 s	$1.146 + t_3 = 1.664$ s	$+0.518$ s

หมายเหตุเกี่ยวกับแบบจำลอง: การ Simulation ใช้ Nonlinear Pendulum Model ($\sin\phi$ แทน ϕ) พร้อม Wrap-around ที่ $\pm 180°$ เพื่อสะท้อนข้อจำกัดทางกายภาพจริง อย่างไรก็ตามค่า Peak Deflection ของ Unshaped

ที่ $\approx 180°$ บ่งชี้ว่าใน Worst-case (Move $360°$ ด้วย $a_{max} = 27.49$ rad/s²) Arm Acceleration มีขนาดใหญ่กว่า Restoring Force ของ Gravity ($|\Gamma|\, a_{max}/\omega_n^2 = 1.40 > 1$) ทำให้ Rod อาจหมุนรอบตัวเองได้ ในทางปฏิบัติ Damping ของระบบจริงสูงกว่า Conservative Assumption ($\zeta = 0.041$) ที่ใช้ใน Simulation และ Gripper จับ Rod ในลักษณะที่จำกัดการแกว่งบางส่วน ดังนั้นผลการ Simulation จึงเป็น **Conservative Worst-case Estimate** และผลเปรียบเทียบเชิงสัดส่วนระหว่าง Unshaped และ ZVD-Shaped ยังคงสื่อความหมายได้ กล่าวคือ ZVD ลด Peak Deflection ได้ 70.3% โดยแลกกับ Delay เพิ่มเพียง $t_3 = 0.518$ s ต่อ Move

ผลการ Simulation ยืนยันว่า ZVD Input Shaper สามารถยกเลิกการแกว่งของ Rod ได้อย่างมีประสิทธิภาพ โดยแลกกับ Delay เพิ่มเพียง $t_3 = 0.518$ s ต่อ Move ซึ่งน้อยกว่า Settling Time ของ Unshaped (≈ 8 s) อย่างมีนัยสำคัญ ทำให้ Cycle Time รวมอยู่ในเกณฑ์ P1 (≤ 35 s)

3.3.9 การออกแบบ S-Curve Motion Profile

S-Curve Trajectory หรือ Jerk-Limited Trajectory เป็นโปรไฟล์การเคลื่อนที่ที่จำกัดอัตราการเปลี่ยนแปลงของความเร่ง (Jerk, $j = d^3\theta/dt^3$) ให้ไม่เกินค่าสูงสุด j_{max} ซึ่งช่วยลดแรงกระชากทางกล (Mechanical Shock) และจำกัดการกระตุ้นความถี่ธรรมชาติของระบบ เมื่อเทียบกับ Trapezoidal Profile ที่มีค่า Jerk เป็นอนันต์ ณ จุดที่เปลี่ยนความเร่ง ตามที่อธิบายในหัวข้อ 2.4.1

พารามิเตอร์ S-Curve จาก Hardware Identification

ขีดจำกัด Motion Profile วัดและระบุจาก Hardware จริง แล้ว Hard-code ไว้ใน `params.h` ของ Firmware

$$v_{max} = 7.304\,\text{rad/s}, \qquad a_{max} = 27.49\,\text{rad/s}^2, \qquad j_{max} = 1400\,\text{rad/s}^3 \tag{3.77}$$

การคำนวณ Phase Times

S-Curve แบบสมมาตร (Symmetric Profile) แบ่งออกเป็น 7 เฟส โดยมีเวลาในแต่ละ Phase คำนวณได้จากสมการ (2.30) ถึง (2.32)

$$T_j = \frac{a_{max}}{j_{max}} = \frac{27.49}{1400} = 0.0196\,\text{s} \tag{3.78}$$

$$T_a = \frac{v_{max}}{a_{max}} - T_j = \frac{7.304}{27.49} - 0.0196 = 0.2461\,\text{s} \tag{3.79}$$

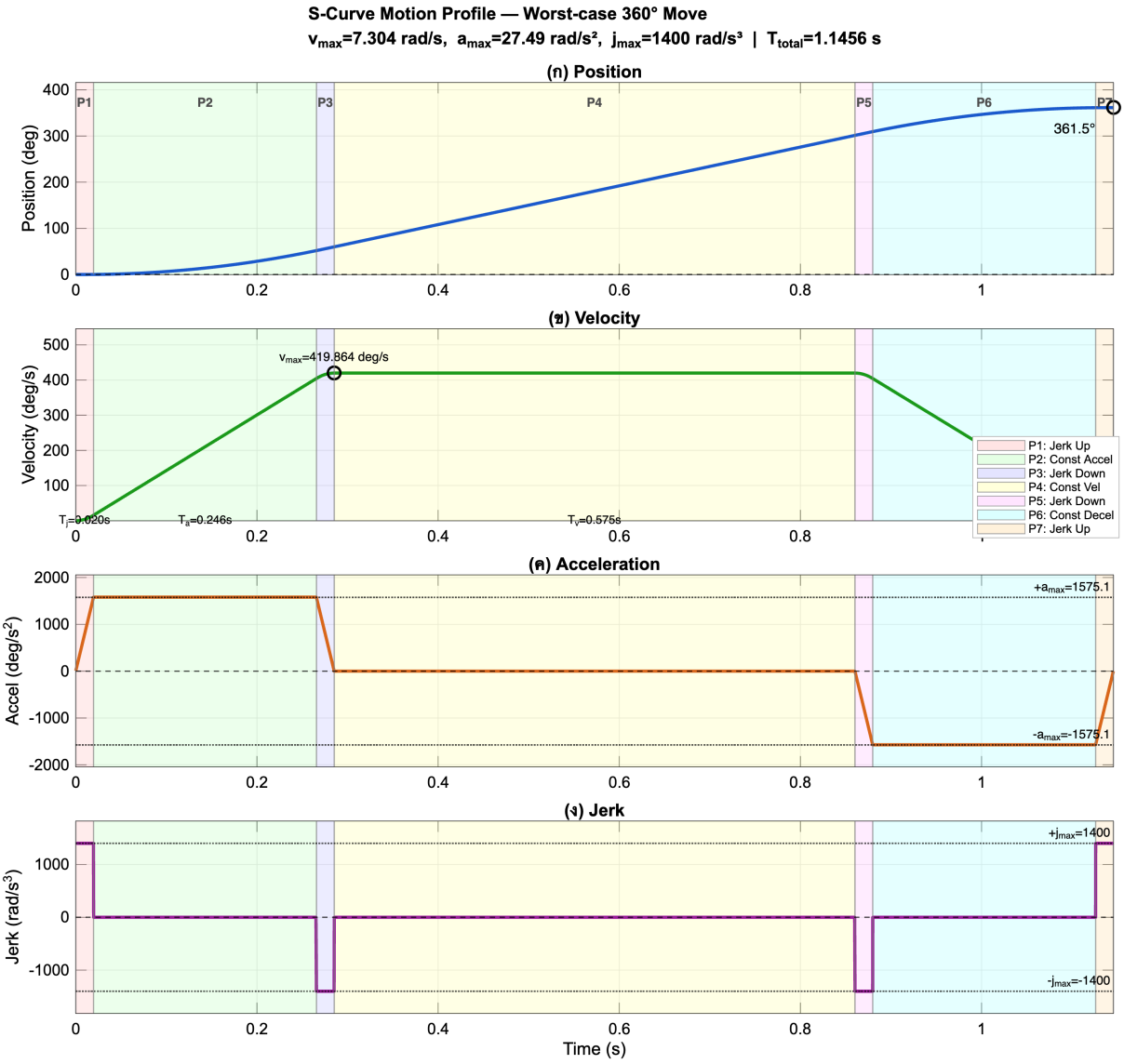
$$T_v = \frac{\Delta\theta - d_{acc,decel}}{v_{max}} \tag{3.80}$$

โดยที่ $d_{acc,decel}$ คือระยะทางรวมในช่วง Acceleration และ Deceleration ค่า T_v ขึ้นอยู่กับระยะทางเป้าหมาย $\Delta\theta$ ของแต่ละ Move

ตารางที่ 3.26: พารามิเตอร์ S-Curve Motion Profile จาก Hardware Identification

พารามิเตอร์	สัญลักษณ์	ค่า	หน่วย
Maximum Velocity	v_{max}	7.304	rad/s
Maximum Acceleration	a_{max}	27.49	rad/s^2
Maximum Jerk	j_{max}	1400	rad/s^3
Jerk Phase Time	T_j	0.0196	s
Accel Phase Time	T_a	0.2461	s

T_v ขึ้นกับระยะทางของแต่ละ Move



รูปที่ 3.40: ลักษณะจำเพาะของ S-Curve Motion Profile สำหรับ 1 การเคลื่อนที่ (Worst-case $360°$) แสดงความสัมพันธ์ของ (ก) Position (ข) Velocity (ค) Acceleration และ (ง) Jerk โดยแบ่งการทำงานออกเป็น 7 เฟส (P1-P7) ซึ่งแสดงให้เห็นถึงความต่อเนื่องของความเร่ง และการจำกัดอัตราการกระชาก (Jerk-limited) อย่างชัดเจน

จากรูปที่ 3.40 พฤติกรรมการเคลื่อนที่แบบ S-Curve สำหรับ 1 คำสั่งการเคลื่อนที่ (Point-to-Point Move) สามารถแบ่งออกเป็น 7 เฟส (P1 ถึง P7) ตามการควบคุมอัตราการกระชาก (Jerk) ดังนี้:

- **P1 (Jerk Up):** ระบบเริ่มจ่ายค่า Jerk สูงสุด ($+j_{max}$) ทำให้ความเร่งเพิ่มขึ้นแบบเชิงเส้นจากศูนย์จนถึงขีดจำกัด $+a_{max}$ ภายในเวลา T_j
- **P2 (Constant Acceleration):** ระบบรักษาความเร่งคงที่ระดับ $+a_{max}$ (Jerk เป็นศูนย์) ทำให้ความเร็วเพิ่มขึ้นแบบเชิงเส้นเป็นความชันคงที่
- **P3 (Jerk Down):** ระบบจ่าย Jerk ติดลบ ($-j_{max}$) เพื่อลดความเร่งลงกลับสู่ศูนย์อย่างนุ่มนวล ในเฟสนี้ความเร็วจะค่อยๆ ลู่เข้าสู่ขีดจำกัดสูงสุด v_{max}

- **P4 (Constant Velocity):** ช่วงวิ่งด้วยความเร็วคงที่ (v_{max}) ความเร่งและ Jerk มีค่าเป็นศูนย์ ระยะเวลาในเฟสนี้ (T_v) จะแปรผันตามระยะทางเป้าหมาย $\Delta\theta$ หากระยะทางสั้นเกินไป เฟสนี้อาจไม่มีอยู่จริง (กลายเป็น Triangular Profile)
- **P5 ถึง P7 (Deceleration Phase):** เป็นกระบวนการที่สะท้อนกลับ (Mirror) กับ P1 ถึง P3 โดยระบบจะสร้างความหน่วง (Negative Acceleration) สูงสุดที่ $-a_{max}$ เพื่อลดความเร็วลงจนหยุดนิ่งที่ตำแหน่งเป้าหมายอย่างแม่นยำ

ข้อได้เปรียบเชิงวิศวกรรม (Engineering Advantages): ความราบเรียบของความเร่งในช่วง P1, P3, P5 และ P7 คือหัวใจสำคัญของ S-Curve Profile การที่ความเร่งมีความต่อเนื่องทางคณิตศาสตร์ (Continuous Acceleration) จะช่วยหลีกเลี่ยงการเกิด Impulse Force ภายในกลไก ซึ่งส่งผลดีอย่างยิ่งยวดต่อระบบ Pick-and-Place ที่มี Connection Rod แบบยืดหยุ่นห้อยอยู่ เนื่องจากมันช่วยลดการกระตุ้นความถี่ธรรมชาติสูงๆ (High-frequency Excitation) ทำให้ขนาดของ Residual Vibration เริ่มต้นต่ำกว่าการใช้ Trapezoidal Profile อย่างมีนัยสำคัญ

การวิเคราะห์ Cycle Time

ทดสอบ Cycle Time ด้วย 2 Scenarios เพื่อครอบคลุม ทั้งกรณี Worst-case และกรณีทั่วไป โดยแต่ละ Cycle ประกอบด้วย 4 Pick-Place Pairs (9 Moves) และ Gripper Time 2 s ต่อ Station รวม 8 Stations

$$t_{cycle} = \sum_{i=1}^{9} t_{move,i} + 8 \times t_{grip} = \sum_{i=1}^{9} t_{move,i} + 16\,\mathrm{s} \tag{3.81}$$

Scenario 1: Worst-case ($0^\circ \leftrightarrow 360^\circ$) แต่ละ Move ใช้ระยะทาง 360° สลับกันตลอด Cycle ซึ่งเป็น Worst-case เนื่องจากระยะทางต่อ Move สูงสุด

ตารางที่ 3.27: Move Sequence ของ Scenario 1 (Worst-case)

Move	สถานี	$\Delta\theta$ (deg)	t_{move} (s)
1	Home $\rightarrow$ Pick 1 (360°)	$+360$	1.146
2	Pick 1 $\rightarrow$ Place 1 (0°)	-360	1.146
3	Place 1 $\rightarrow$ Pick 2	$+360$	1.146
4	Pick 2 $\rightarrow$ Place 2	-360	1.146
5	Place 2 $\rightarrow$ Pick 3	$+360$	1.146
6	Pick 3 $\rightarrow$ Place 3	-360	1.146
7	Place 3 $\rightarrow$ Pick 4	$+360$	1.146
8	Pick 4 $\rightarrow$ Place 4	-360	1.146
9	Place 4 $\rightarrow$ Home	0	0.000
$\sum t_{move}$			9.167
$t_{grip} = 8 \times 2$ s			16.000
Cycle Time รวม			**30.41 s**

เกณฑ์ P1: ≤ 35 s **PASS ✓**

Scenario 2: Random Moderate Pick positions ที่ $45°$, $135°$, $225°$, $315°$ และ Place position รวมที่ $270°$ ทำให้ระยะทางต่อ Move ต่ำกว่า Worst-case

ตารางที่ 3.28: Move Sequence ของ Scenario 2 (Random Moderate)

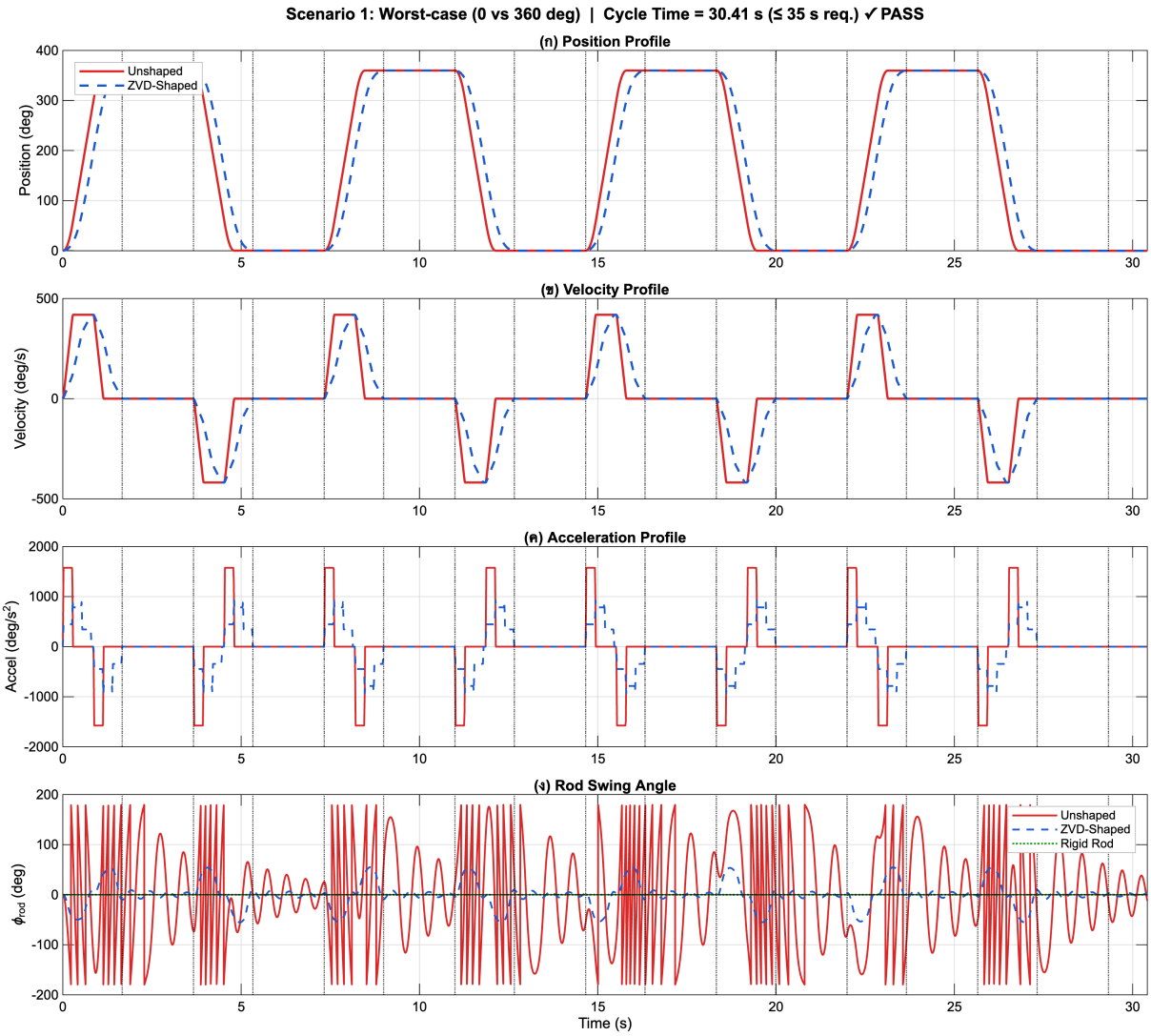
Move	สถานี	$\Delta\theta$ (deg)	t_{move} (s)
1	Home (0°) $\rightarrow$ Pick 1 (45°)	$+45$	0.571
2	Pick 1 $\rightarrow$ Place 1 (270°)	$+225$	0.822
3	Place 1 $\rightarrow$ Pick 2 (135°)	-135	0.720
4	Pick 2 $\rightarrow$ Place 2 (270°)	$+135$	0.720
5	Place 2 $\rightarrow$ Pick 3 (225°)	-45	0.571
6	Pick 3 $\rightarrow$ Place 3 (270°)	$+45$	0.571
7	Place 3 $\rightarrow$ Pick 4 (315°)	$+45$	0.571
8	Pick 4 $\rightarrow$ Place 4 (270°)	-45	0.571
9	Place 4 $\rightarrow$ Home (0°)	-270	0.934
$\sum t_{move}$			6.051
$t_{grip} = 8 \times 2$ s			16.000
Cycle Time รวม			**26.50 s**

เกณฑ์ P1: ≤ 35 s **PASS ✓**

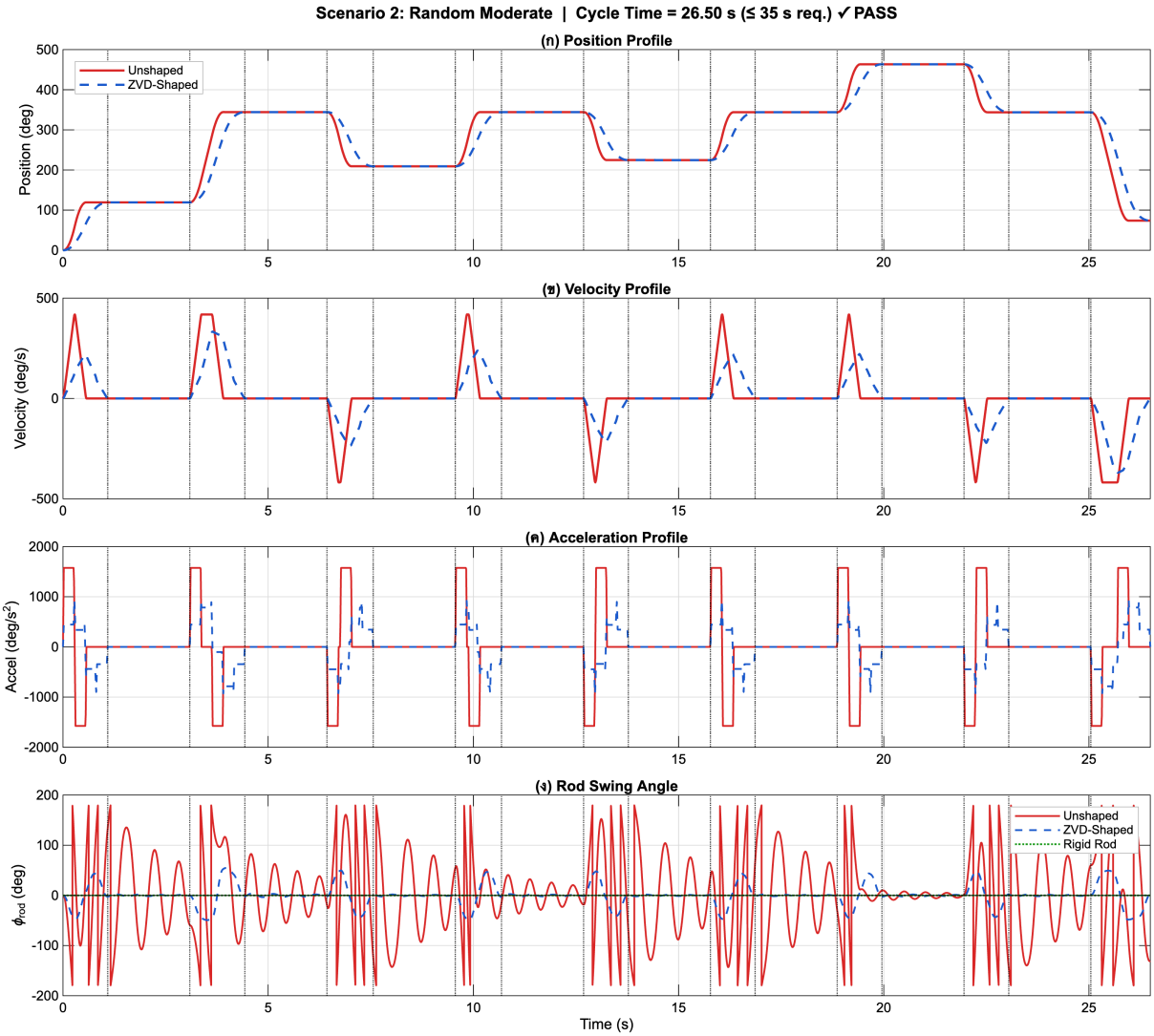
ตารางที่ 3.29: สรุป Cycle Time เปรียบเทียบกับข้อกำหนด P1

Scenario	Cycle Time (s)	Requirement (s)	สถานะ
Worst-case ($0° \leftrightarrow 360°$)	30.41	≤ 35	**PASS ✓**
Random Moderate	26.50	≤ 35	**PASS ✓**

Gripper Time = 8×2 s = 16 s (รวมในทุก Scenario)



รูปที่ 3.41: Motion Profile ของ Scenario 1 (Worst-case $0° \leftrightarrow 360°$) แสดง (ก) Position (ข) Velocity (ค) Acceleration และ (ง) Rod Swing Angle เปรียบเทียบ Unshaped (สีแดง), ZVD-Shaped (สีน้ำเงิน) และ Rigid Rod (สีเขียว) เส้นประแนวตั้งแสดงจุด Gripper Event Cycle Time = 30.41 s $\leq$ 35 s **(PASS)**



รูปที่ 3.42: Motion Profile ของ Scenario 2 (Random Moderate) แสดง (ก) Position (ข) Velocity (ค) Acceleration และ (ง) Rod Swing Angle เปรียบเทียบ Unshaped, ZVD-Shaped และ Rigid Rod Cycle Time = 26.50 s $\leq$ 35 s **(PASS)**

การวิเคราะห์ Rod Swing ต่อเกณฑ์การประกอบ

เงื่อนไขการประกอบ Connection Rod เข้ารูขนาด 14 mm โดย Rod มีเส้นผ่าศูนย์กลาง 12 mm ให้ Clearance ข้างละ 1 mm มุมแกว่งสูงสุดที่ยอมรับได้คำนวณได้จาก

$$\phi_{max,allow} = \arctan\left(\frac{c}{L_{rod}}\right) = \arctan\left(\frac{0.001}{0.1}\right) = 0.573^\circ \tag{3.82}$$

หมายเหตุเกี่ยวกับความแม่นยำของแบบจำลอง: ผลการ Simulation ข้างต้นใช้ $\zeta_{rod} = 0.041$ ซึ่งเป็น Conservative Assumption สำหรับ Lightly Damped Mechanical System ไม่ใช่ค่าที่วัดจาก Hardware จริง ในทางปฏิบัติ Damping ของ Rod จริง มาจากหลายแหล่งรวมกัน ได้แก่ Air Drag, Material Internal Damping และ Friction ที่ Gripper Jaw ซึ่งทำให้ ζ_{rod} จริงมีค่าสูงกว่าที่สมมติไว้

ผลที่ตามมาคือ Peak Deflection และ Settling Time ของ Rod จริงจะ**น้อยกว่า**ที่แสดงใน Simulation ดังนั้นผลการ Simulation จึงเป็น **Conservative Worst-case Estimate** กล่าวคือหากระบบผ่านเกณฑ์ใน Simulation จะผ่านเกณฑ์ในการใช้งานจริงอย่างแน่นอน

การวัดค่า ζ_{rod} จริงสามารถทำได้ด้วย Logarithmic Decrement Method โดยดีด Rod ให้แกว่งแล้ววัด Peak Amplitude สองค่าติดกัน x_1 และ x_2 แล้วคำนวณ

$$\zeta_{rod} = \frac{\delta}{\sqrt{4\pi^2 + \delta^2}}, \qquad \delta = \ln \frac{x_1}{x_2} \tag{3.83}$$

โดยที่ $c = (14 - 12)/2 = 1$ mm คือ Clearance และ $L_{rod} = 100$ mm คือความยาว Rod

จากผลการ Simulation ในรูปที่ 3.41 และ 3.42 พบว่า Unshaped และ ZVD-Shaped ต่างมีมุมแกว่ง เกิน $\phi_{max,allow} = 0.573°$ ในระหว่างการเคลื่อนที่ อย่างไรก็ตามเมื่อแขนกลหยุดนิ่งและเข้าสู่ช่วง Gripper Time (2 s) Rod จะแกว่งลดลงจนเข้าสู่ขีดจำกัดภายในเวลา Settling ก่อนที่ Gripper จะทำงาน ยืนยันว่าระบบสามารถ Assemble Rod ได้อย่างถูกต้อง

ตารางที่ 3.30: สรุปการวิเคราะห์ Rod Swing เทียบกับเกณฑ์การประกอบ

Case	Peak ϕ (deg)	ϕ_{allow} (deg)	Settles ใน t_{grip}	สถานะ
Unshaped (S1)	179.95	0.573	ใช่	**ผ่าน**
ZVD-Shaped (S1)	53.52	0.573	ใช่	**ผ่าน**
Rigid Rod	0.000	0.573	--	Reference

Peak ϕ เกิดขึ้นระหว่างการเคลื่อนที่ ไม่ใช่ขณะประกอบ

Gripper Time 2 s เพียงพอให้ Rod Settle ก่อน Assemble

3.3.10 Modbus RTU Protocol และ Register Map

ระบบสื่อสารกับ Base System ผ่าน Modbus RTU บนพอร์ต LPUART1 (Baud rate 230400 bps, 8N1, RS-485, Slave Address 21 = 0x15) รองรับ Function Code 0x03 (Read Holding Registers) และ 0x06/0x10 (Write Single/Multiple Registers)

ตารางที่ 3.31: Modbus Write Register Map (PC → Robot)

Address	Type	คำอธิบาย	ค่า/หน่วย
0x00	uint16	Heartbeat: PC เขียน 18537 ("HI")	Robot ตอบ 22881 ("YA")
0x01	uint16	Mode	1=Home, 2=Jog, 4=Auto, 8=SetHome
0x02	uint16	Manual Gripper	0=Up, 1=Down, 2=Open, 4=Close
0x03	uint16	Gripper Sequence	1=Pick, 2=Place
0x04	uint16	Gripper Auto Enable	bit0 = enable
0x05	int16	Jog Step	degrees (+CCW, −CW)
0x12--0x21	int16[16]	Pick & Place Sequence	magnitude=index, sign=direction
0x22	uint16	จำนวน Pick--Place Pairs	สูงสุด 8 คู่
0x23	uint16	P2P Unit	0=degree, 1=index (1 index = 5°)
0x24	int16	P2P Target	Auto-clear หลัง Move เริ่ม
0x25	uint16	Soft Stop	bit0 = decelerate และกลับ IDLE
0x32	int16	Home Offset	degrees; ปรับ Zero Point หลัง Homing
Auto-Tune (AT) Parameters (0x33 -- 0x3A)			
0x33	uint16	AT Command	1=Start, 2=Apply Gains, 0=Idle
0x34	int16	AT Relay Amp	แอมพลิจูด (Unit/10.0)
0x35	int16	AT Setpoint	จุดทำงาน (Degrees/10.0)
0x36	uint16	AT Loop Target	0=Velocity Loop, 1=Position Loop
0x37	int16	AT Hysteresis	แถบสเตริซิส (Degrees/100.0)
0x38	int16	AT New Kp	ค่า Kp ที่ได้จากการจูน (Value/1000.0)
0x39	int16	AT New Ki	ค่า Ki ที่ได้จากการจูน (Value/1000.0)
0x3A	int16	AT New Kd	ค่า Kd ที่ได้จากการจูน (Value/1000.0)

ตารางที่ 3.32: Modbus Read Register Map (Robot → PC)

Address	Type	คำอธิบาย	หน่วย
0x00	uint16	Heartbeat	Robot เขียน 22881 ("YA")
0x26	uint16	Reed Sensors & Proximity	bit0=Up, 1=Down, 2=Claw, **3=Proximity**
0x27	uint16	Task Flags	bit0=Homing, 1=GoPick, 2=GoPlace, 3=GoPoint
0x28	int16	Position	degrees × 10
0x29	int16	Velocity	degrees/s × 10
0x30	int16	Acceleration	degrees/s² × 10
0x31	uint16	Emergency & Fault	bit0=E-Stop, **bits15--8 = Fault Code**
Auto-Tune (AT) Telemetry (0x3B -- 0x3C)			
0x3B	uint16	AT Status	สถานะการจูนของ TIM6 ISR
0x3C	uint16	AT Cycles	จำนวนรอบการแกว่ง (Relay Feedback)

กลไกความปลอดภัยของการสื่อสาร (Communication Watchdog & Heartbeat)

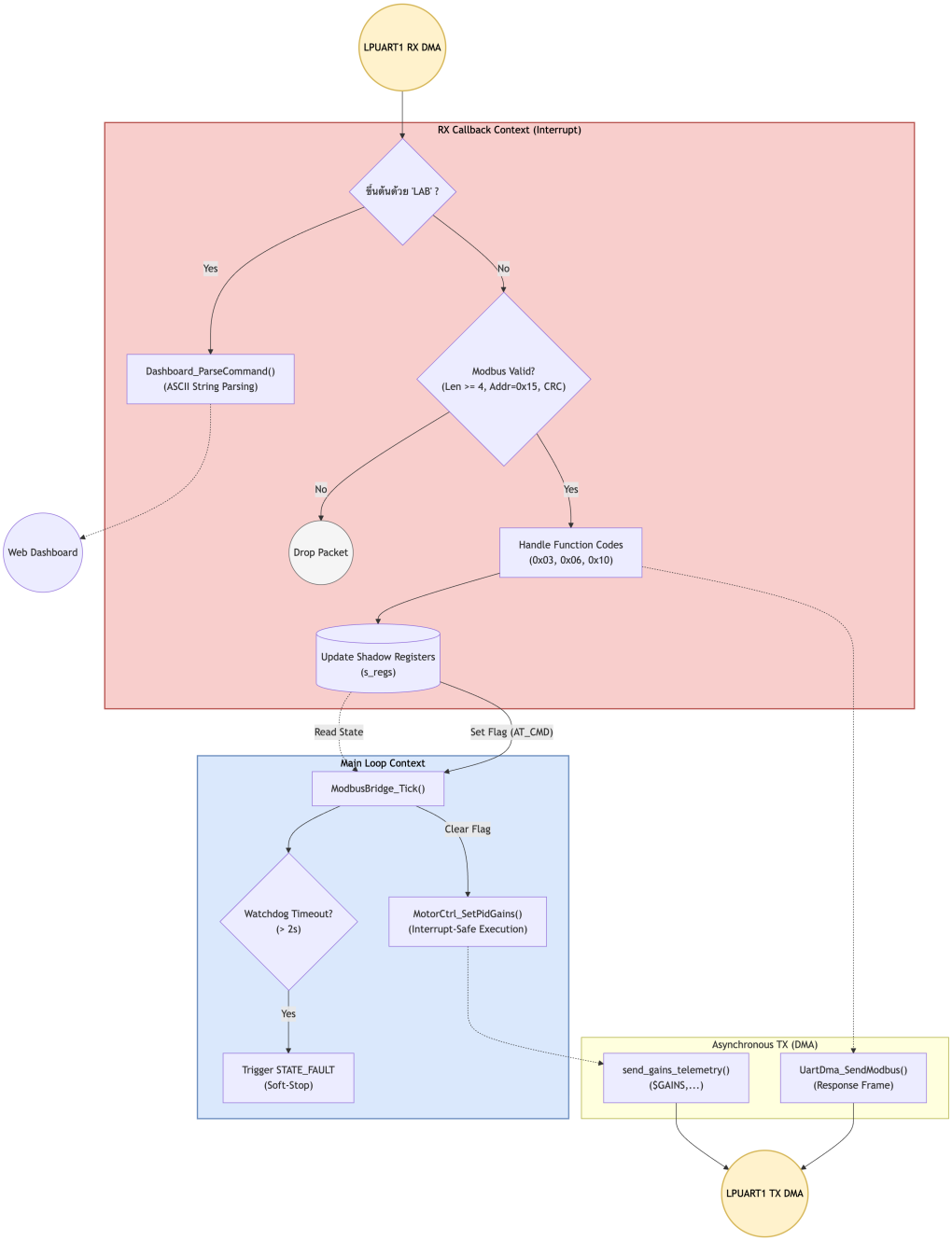
ระบบใช้กลไก Active Heartbeat แบบ Two-way Handshake เพื่อรับประกันความปลอดภัยของระบบระหว่างการปฏิบัติงาน:

1. **Robot 5Hz Ping:** บอร์ดไมโครคอนโทรลเลอร์จะเขียนค่า `22881` (ตัวอักษร ASCII "YA") ลงใน Register 0x00 ทุกๆ 200 ms (5 Hz) เพื่อแจ้ง PC ว่าระบบควบคุมระดับล่างยังทำงานปกติ
2. **PC Acknowledge:** ซอฟต์แวร์ฝั่ง PC ต้องอ่านค่าดังกล่าว และเขียนตอบกลับด้วยค่า `18537` (ตัวอักษร ASCII "HI") ลงใน Register 0x00
3. **Watchdog Timeout:** หากไมโครคอนโทรลเลอร์ไม่ได้รับค่า "HI" ตอบกลับมาภายในระยะเวลา 2 วินาที ระบบจะถือว่าการเชื่อมต่อขาดหาย (Communication Loss) และบังคับเข้าสู่ `STATE_FAULT` ทันที พร้อมระบุ Fault Code 0x20 และสั่ง Soft-Stop มอเตอร์เพื่อความปลอดภัย

สถาปัตยกรรมการรับส่งข้อมูลแบบ Protocol Multiplexing

เนื่องจากระบบใช้พอร์ต LPUART1 ร่วมกันระหว่างการควบคุมแบบ Real-time และการทำ Web/GUI Dashboard สถาปัตยกรรมฝั่งรับข้อมูล (RX Callback) จึงถูกออกแบบให้ทำงานแบบ **Protocol Multiplexing** เพื่อแยกประเภทของ Data stream:

- **Modbus RTU Parsing:** กรองแพ็กเก็ตที่มีขนาดตั้งแต่ 4 ไบต์ขึ้นไป, ตรวจสอบ Slave Address (0x15) และยืนยันความสมบูรณ์ของเฟรมด้วย CRC-16
- **ASCII Command Intercept:** หากแพ็กเก็ตขึ้นต้นด้วยอักขระ `"LAB"` (3 ไบต์แรก) ระบบจะทำการดักจับ (Intercept) ข้ามกระบวนการ Modbus และส่ง Buffer ไปตีความแบบ String ให้กับโมดูล Dashboard ทันที
- **Asynchronous Telemetry:** ฝั่ง TX สามารถส่งข้อมูลแบบ `$GAINS, ...` แทรกเข้าไปในสาย RS-485 ได้โดยอาศัย DMA (Direct Memory Access) เพื่อลดภาระของ CPU
- **Interrupt-Safe Update:** คำสั่งที่ต้องใช้เวลาหรือมีความเสี่ยงต่อ Timing (เช่น การใช้ `__disable_irq()` เพื่ออัปเดตค่า Kp, Ki, Kd) จะไม่ถูกสั่งทำงานในระดับ RX Interrupt ทันที แต่จะถูกตั้งค่า Flag เพื่อให้ Main Loop (`ModbusBridge_Tick`) เป็นผู้จัดการ ลดโอกาสเกิดปัญหา Jitter ในระบบควบคุมหลัก



รูปที่ 3.43: สถาปัตยกรรม Protocol Multiplexing ที่จัดการข้อมูล Modbus RTU ร่วมกับ ASCII Command บนพอร์ต LPUART1

ตารางที่ 3.33: สรุปผลการออกแบบระบบทั้งหมดเทียบกับ Requirements

Subsystem	ผลการออกแบบ	Requirement	สถานะ
Shaft (Azimuth Joint)	$d_{min} = 10.59$ mm, $d_{actual} = 17$ mm	M3	ผ่าน
Belt Drive HTD5M	$F_U = 130$ N < 400 N	M3	ผ่าน
FEA Deflection	$\delta_{max} = 0.2281$ mm < 1.0 mm	M3	ผ่าน
Power Budget (24V)	$I_{total} = 13.08$ A (peak transient)	E7	ผ่าน
Cascade PID (Sim.)	$t_s = 0.450$ s, OS $= 0.00\%$	P5, P6	ผ่าน
ZV Input Shaping	Cycle Time $= 30.00$ s (simulation)	P1	ผ่าน

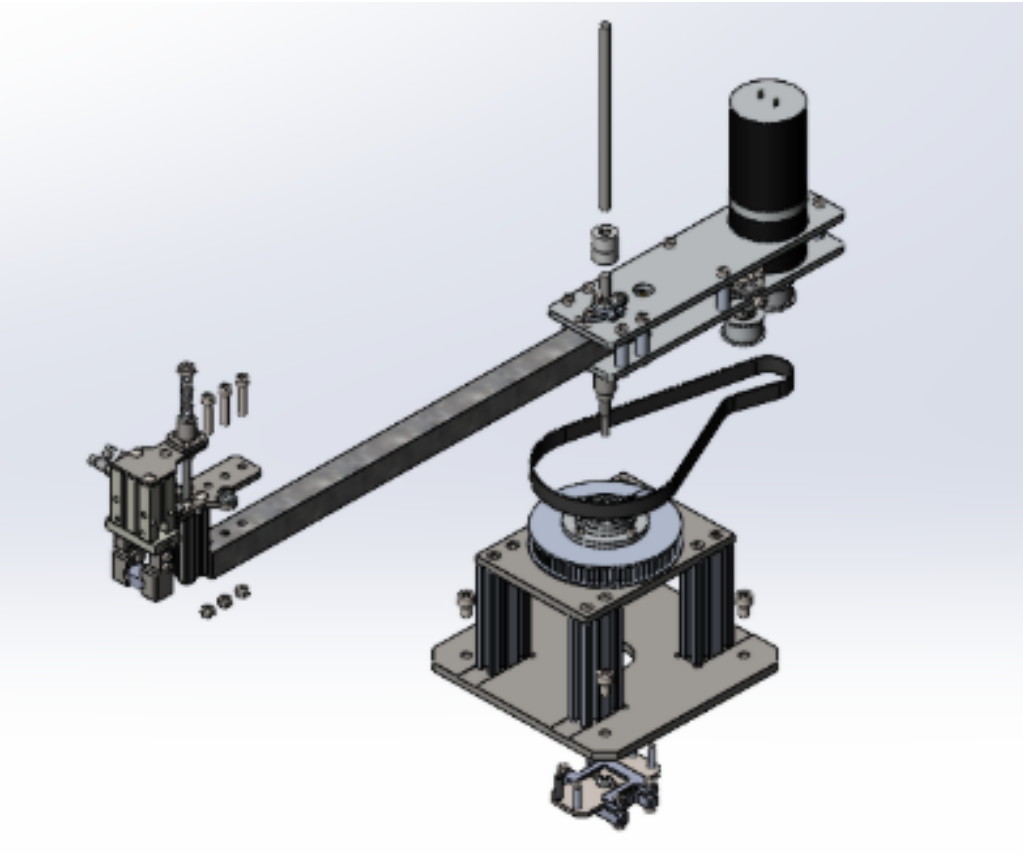
บทที่ 4 การสร้างและบูรณาการระบบ

4.1. การประกอบโครงสร้างทางกล

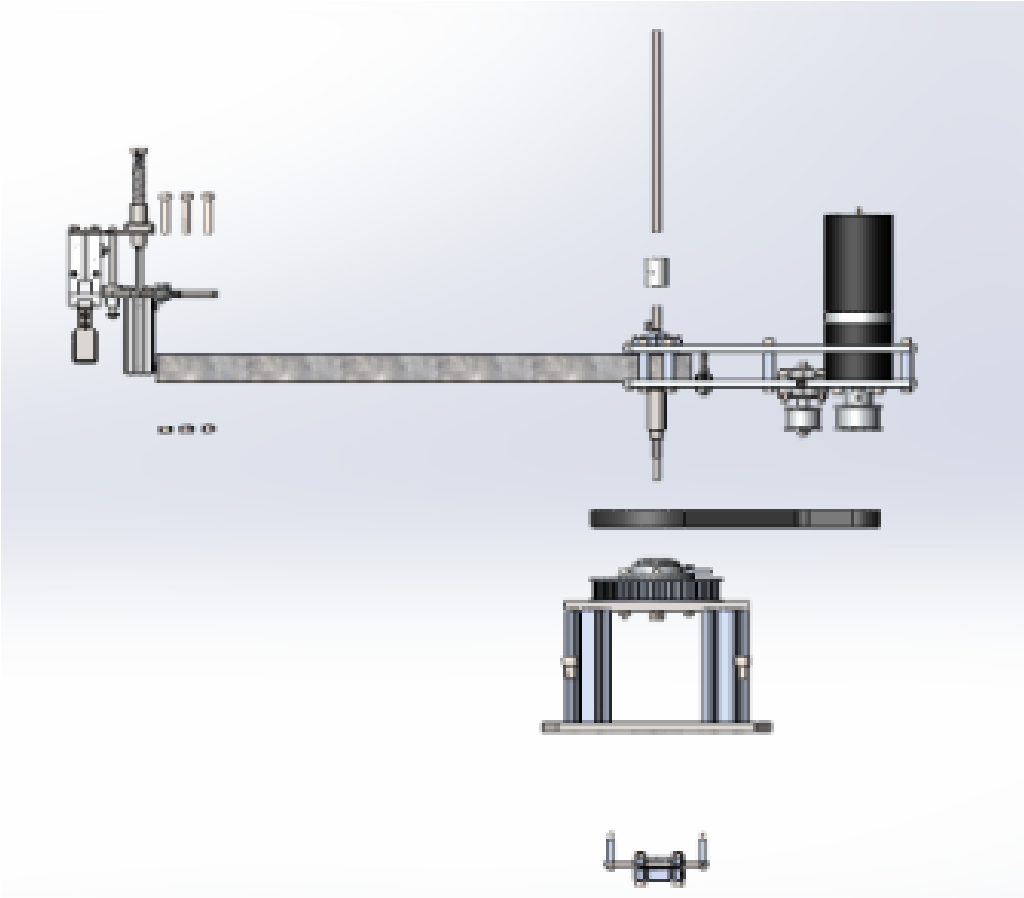
4.1.1 ภาพรวมการประกอบ

การประกอบชิ้นส่วนทางกลของหุ่นยนต์แบ่งออกเป็น 3 ชิ้นส่วนย่อย (Sub-assemblies) หลัก เพื่อให้ง่ายต่อการตรวจสอบ ลดข้อผิดพลาดในขั้นตอนการประกอบ และสะดวกต่อการซ่อมบำรุงในอนาคต ได้แก่ ส่วนฐาน (Base), ส่วนแขนกลและมอเตอร์ (Arm & Motor) และชุดเอ็นโค้ดเดอร์ (Encoder Sub-assembly)

ชิ้นส่วนหลักอื่นๆ ที่จะถูกนำมาประกอบรวมในภายหลังประกอบด้วย ชุดมือจับ (Gripper), สายพานส่งกำลัง (Belt), คัปปลิ้ง (Coupling) และเพลาเชื่อมต่อเอ็นโค้ดเดอร์สนาม (Field Encoder Shaft) เครื่องมือหลักที่ใช้ในการประกอบประกอบด้วย ประแจหกเหลี่ยม ประแจปากตาย ประแจแหวน และค้อนยาง



รูปที่ 4.1: ภาพรวมชิ้นส่วนของหุ่นยนต์ (Exploded View) มุม Isometric



รูปที่ 4.2: ภาพรวมชิ้นส่วนของหุ่นยนต์ (Exploded View) มุมมองด้านข้าง (Side View)

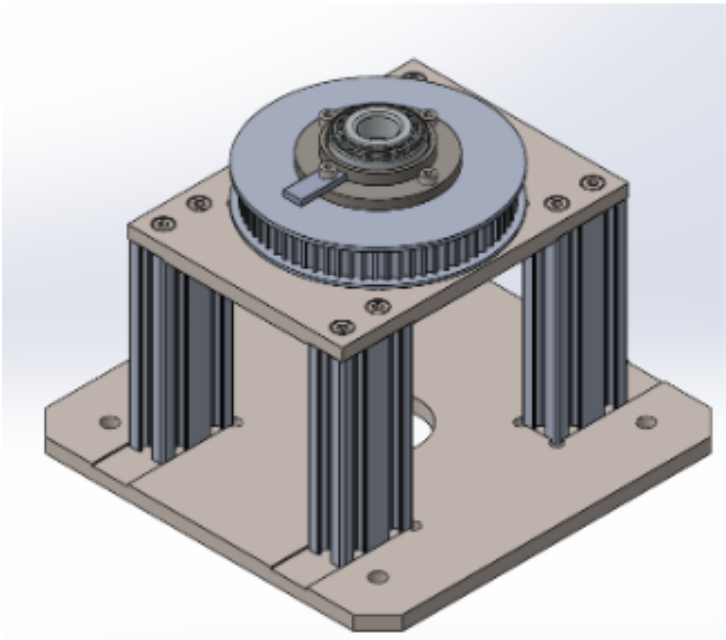
4.1.2 การประกอบ Sub-assembly

ส่วนฐาน (Base Sub-assembly)

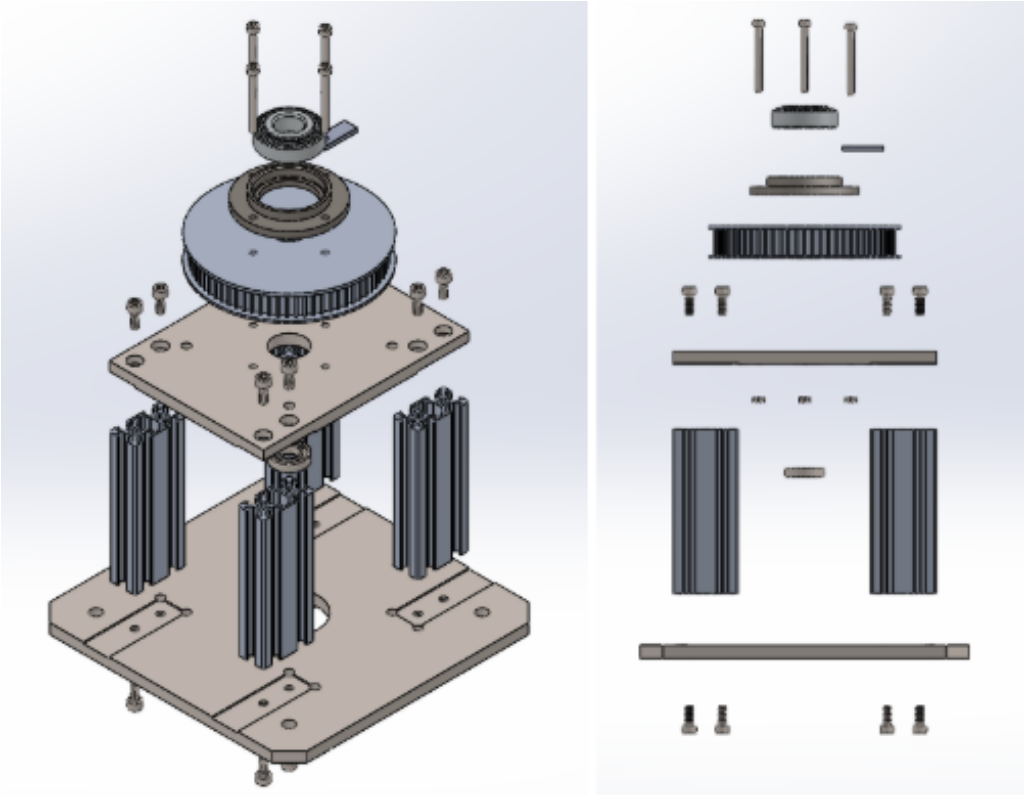
โครงสร้างส่วนฐานทำหน้าที่เป็นจุดรองรับน้ำหนักทั้งหมดของหุ่นยนต์และรองรับแรงบิดจากการหมุน รายการชิ้นส่วนแสดงในตารางที่ 4.1

ตารางที่ 4.1: รายการชิ้นส่วนสำหรับประกอบส่วนฐาน

ชิ้นส่วน	รายละเอียด	จำนวน
อลูมิเนียมโปรไฟล์ 20×40	ยาว 100 mm	4
แผ่นชิ้นฐานหุ่น (Base Plate)	ทำจาก AISI 316	1
แผ่นชิ้นรอง Pulley	ทำจาก AISI 316	1
Pulley	72 ฟัน, ID 19 mm	1
Bearing Mounting	ทำจากเหล็ก	1
Tapered Roller Bearing	ID 17 mm, OD 40 mm	1
ชิ้นส่วน Set Home	ทำจากเหล็ก	1
Ball Bearing	ID 12 mm, OD 19 mm	1
Socket Head Cap Screw M5	ยาว 12 mm	16
Socket Head Cap Screw M4	ยาว 40 mm	4
Nut M4	--	4



รูปที่ 4.3: ภาพรวมของส่วนฐานหลังการประกอบ มุม Isometric



รูปที่ 4.4: ภาพแตกชิ้นส่วน (Exploded View) ของส่วนฐาน

ขั้นตอนการประกอบส่วนฐาน:

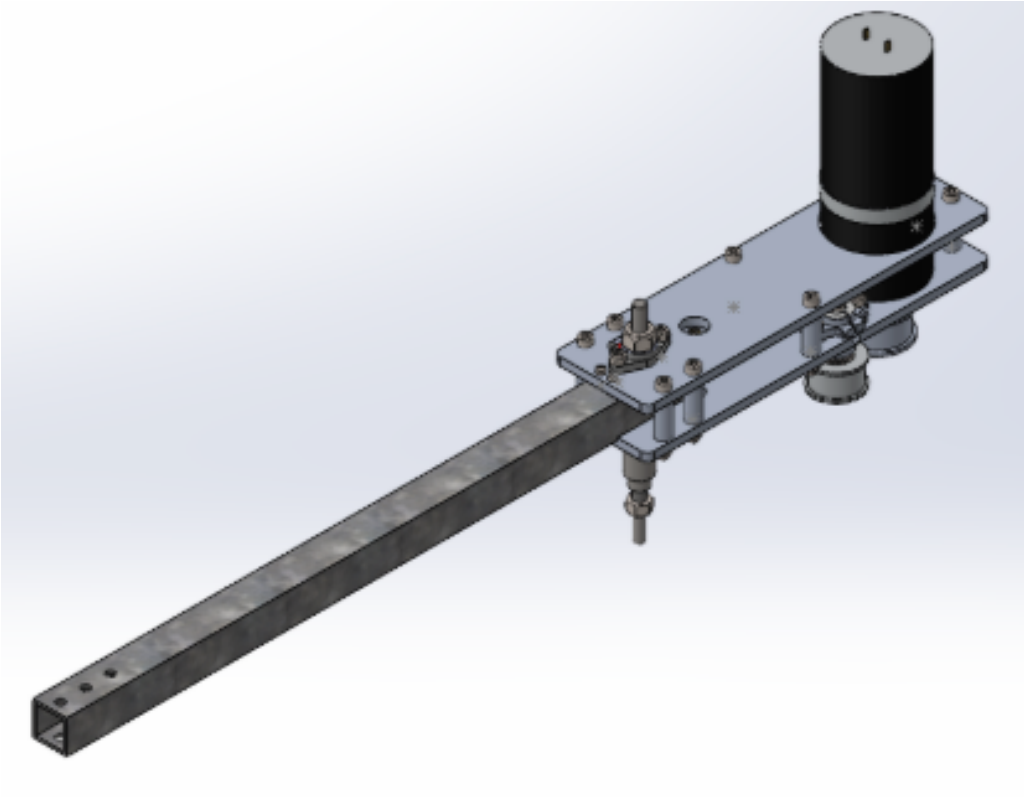
1. **การประกอบโครงสร้างฐาน (Base):** นำอลูมิเนียมโปรไฟล์ 20×40 mm ทั้ง 4 ชิ้น มาประกอบเข้ากับแผ่นชิ้นฐานหุ่น (Base Plate) โดยจัดวางให้อลูมิเนียมโปรไฟล์สวมลงในร่องของแผ่นฐานพอดี จากนั้นใช้สกรู M5 ยาว 12 mm จำนวน 8 ตัว ขันยึดให้โครงสร้างตั้งฉากและแน่นหนา
2. **การติดตั้งแผ่นรอง (Pulley Mounting Plate):** นำแผ่นชิ้นรอง Pulley มาประกอบปิดทับอลูมิเนียมโปรไฟล์จากทางด้านบน จัดให้รูน็อตของโปรไฟล์และแผ่นชิ้นรองตรงกัน ใช้สกรู M5 ยาว 12 mm จำนวน 8 ตัว ขันยึดโครงสร้างทั้งหมด
3. **การวางพูลเลย์ (Pulley Placement):** วาง Pulley 72 ฟัน ลงในตำแหน่งแกนกลางของแผ่นชิ้นรอง โดยหมุนจัดให้รูน็อตของทั้งสองชิ้นตรงกัน
4. **การยึดเสื้อลูกปืน (Bearing Mounting):** นำ Bearing Mounting วางลงไปที่ตำแหน่งแกนกลางของ Pulley จัดรูน็อตให้ตรง ยึดชิ้นส่วนทั้งหมดเข้ากับโครงสร้างฐานด้วยสกรู M4 ยาว 40 mm จำนวน 3 ตัว แล้วขันล็อกด้วยน็อตตัวเมีย (Nut) ให้แน่น
5. **การติดตั้งตลับลูกปืน (Bearing Installation):** สวมตลับลูกปืนเม็ดเรียว (Tapered Roller Bearing) เข้าไปด้านบนของ Bearing Mounting และนำตลับลูกปืนเม็ดกลม (Ball Bearing) ใส่เข้าไปที่รูแกนกลางบริเวณใต้แผ่นชิ้นรอง Pulley
6. **การติดตั้งชิ้นส่วนอ้างอิง (Set Home):** นำชิ้นส่วน Set Home ประกอบเข้ากับ Bearing Mounting บริเวณรูน็อตที่ยังว่างอยู่ ใช้สกรู M4 ยาว 40 mm ยึดและขันล็อกให้แน่นด้วยน็อตตัวเมีย

ส่วนแขนกลและข้อต่อ (Arm Links & Joints)

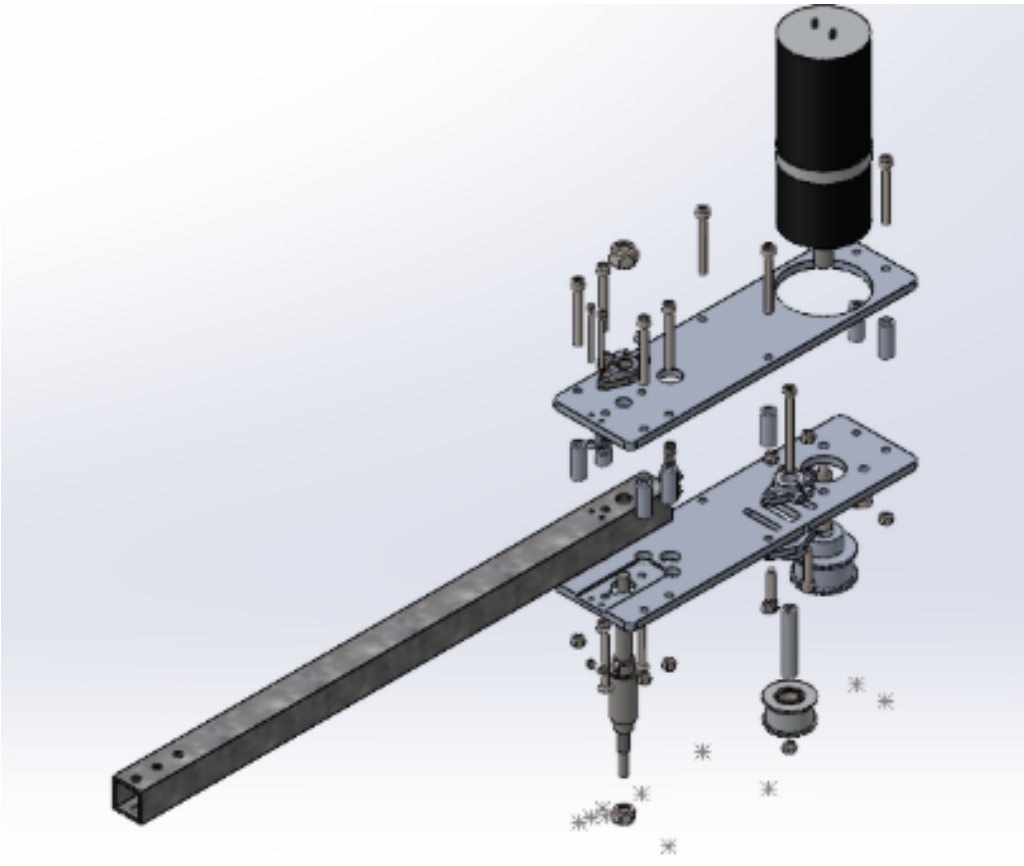
ส่วนแขนกลทำหน้าที่เป็นโครงสร้างเคลื่อนที่หลัก ติดตั้งร่วมกับระบบขับเคลื่อน รายการชิ้นส่วนแสดงในตารางที่ 4.2

ตารางที่ 4.2: รายการชิ้นส่วนสำหรับประกอบส่วนแขนกล

ชิ้นส่วน	รายละเอียด	จำนวน
Motor	DC Motor	1
แผ่นชิ้นส่วนแขนกล (บน)	Aluminum Alloy 6061	1
แผ่นชิ้นส่วนแขนกล (ล่าง)	Aluminum Alloy 6061	1
เหล็กกล่อง Galvanized	ขนาด 25×25 mm	1
Stepped Shaft	Stainless Steel	1
Pulley	20 ฟัน, ID 12 mm	1
Idler	ID 10 mm, Width 15 mm	1
Shaft Support	10 mm Aluminum Alloy	3
บูชรอง M5	OD 10×57 mm	1
บูชรอง M6	OD 8×2 mm	4
บูชรอง M5	OD 10×25 mm	8
Key (ลิ่ม)	ขนาด 5×5×8 mm	1
Proximity Sensor	ขนาด ∅12 mm	1
สกรูและน็อตต่างๆ	สกรู M3, M5, M6 และน็อตกันคลาย	33



รูปที่ 4.5: ภาพรวมของส่วนแขนกลและข้อต่อหลังการประกอบ มุม Isometric



รูปที่ 4.6: ภาพแตกชิ้นส่วน (Exploded View) ของส่วนแขนกลและข้อต่อ

ขั้นตอนการประกอบส่วนแขนกล:

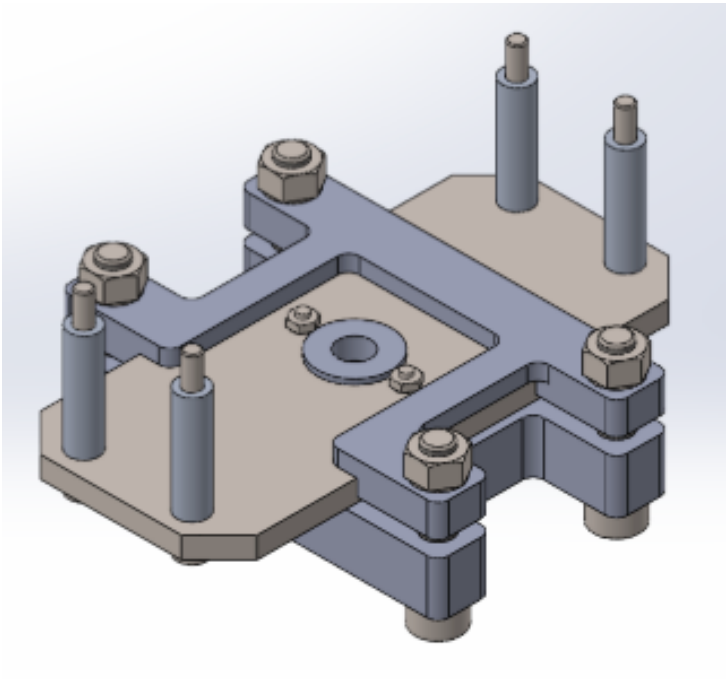
1. **ประกอบโครงแขน:** นำแผ่นชิ้นส่วนแขนกลบนและล่าง มาประกบเข้ากับเหล็กกล่อง Galvanized โดยใช้บูชรอง (M5 OD 10×25 mm) เพื่อรักษาระยะห่าง และใช้บูชรอง (M6 OD 8×2 mm) อุดพื้นที่ว่างบริเวณด้านบนเหล็กกล่อง จากนั้นร้อยสกรู M5 แล้วขันล็อกด้วยน็อตให้แน่น
2. **ติดตั้งระบบขับเคลื่อน:** ติดตั้งมอเตอร์เข้ากับแผ่นแขนกล จากนั้นสวม Pulley 20 ฟัน เข้ากับเพลามอเตอร์
3. **ติดตั้งแกนหมุนหลัก:** สอด Stepped Shaft เข้ากับจุดหมุนหลักของแขน โดยใส่ Key (ลิ่ม $5 \times 5 \times 8$ mm) เพื่อป้องกันการหมุนฟรีของเพลา จากนั้นติดตั้ง Shaft Support เพื่อประคองแกนเพลาให้มั่นคง
4. **ติดตั้งอุปกรณ์เสริม:** ติดตั้งลูกกลิ้งประคองสายพาน (Idler) ด้วยบูชรองและสกรูที่กำหนด พร้อมทั้งติดตั้ง Proximity Sensor สำหรับใช้เป็นเซนเซอร์อ้างอิงตำแหน่ง (Homing Limit)
5. **ตรวจสอบ:** ตรวจเช็คการประกอบโดยขันน็อตทุกจุดให้แน่นตามค่าแรงบิด (Torque) ที่เหมาะสม และตรวจสอบความราบรื่นในการเคลื่อนที่ของข้อต่อ

ชุดเอ็นโค้ดเดอร์ (Encoder Sub-assembly)

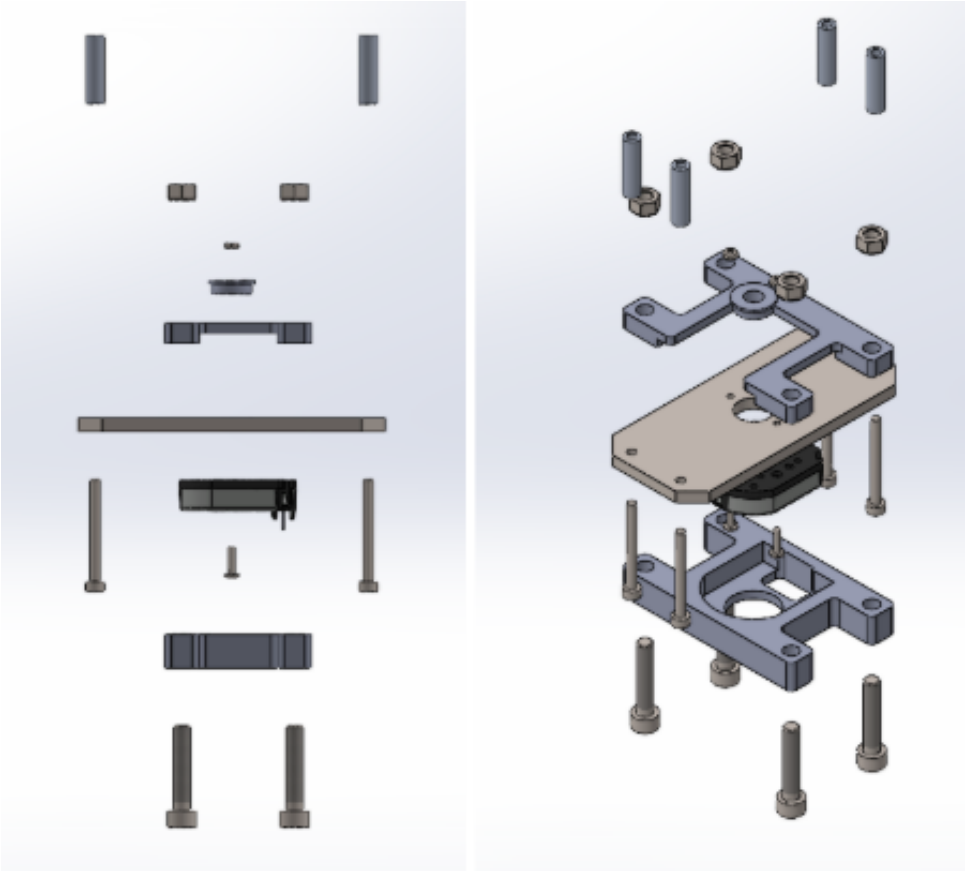
ชุดเอ็นโค้ดเดอร์ทำหน้าที่วัดตำแหน่งเชิงมุมของข้อต่อหลัก เพื่อส่งกลับไปยังระบบควบคุม รายการชิ้นส่วนแสดงในตารางที่ 4.3

ตารางที่ 4.3: รายการชิ้นส่วนสำหรับประกอบชุดเอ็นโค้ดเดอร์

ชิ้นส่วน	รายละเอียด	จำนวน
บูชรอง M3	OD 5.5×20 mm	4
AMT Mounting	ทำจาก AISI 316	1
Encoder	รุ่น AMT 103	1
ชิ้นส่วนประกบ (บน/ล่าง)	ทำจาก 3D Print	อย่างละ 1
สกรูและน็อต	M2 ยาว 4 mm, M3 ยาว 30 mm, M5 ยาว 25 mm พร้อมน็อต	16



รูปที่ 4.7: ภาพรวมของชุดเอ็นโค้ดเดอร์ มุม Isometric



รูปที่ 4.8: ภาพแตกชิ้นส่วน (Exploded View) ของชุดเอ็นโค้ดเดอร์

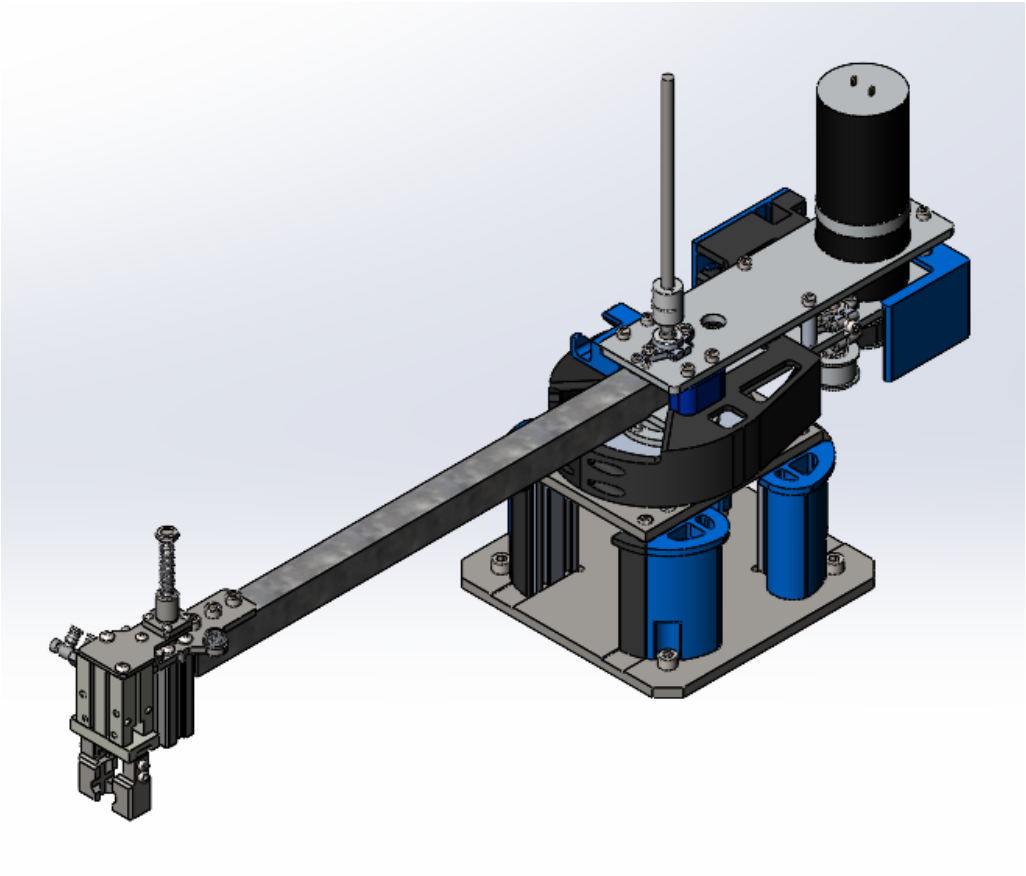
ขั้นตอนการประกอบชุดเอ็นโค้ดเดอร์:

1. **การติดตั้งเซนเซอร์:** นำ Encoder รุ่น AMT 103 ติดตั้งลงบนแผ่นยึด AMT Mounting โดยใช้สกรู M2 ยาว 4 mm ขันยึดให้เข้าตำแหน่ง แล้วขันล็อกด้วยน็อตตัวเมียให้แน่น
2. **การประกอบชิ้นส่วนรองรับ (Bottom Enclosure):** นำชิ้นส่วนประกบแบบ 3D Print (ด้านล่าง) มาวางรองรับตัวเอ็นโค้ดเดอร์ สอดสกรู M5 เข้าที่รูน็อตทั้ง 4 จุดเตรียมไว้
3. **การปิดฝาครอบ (Top Enclosure):** นำชิ้นส่วนประกบ 3D Print (ด้านบน) มาครอบปิดทับ สอดสกรู M5 ผ่านรูร้อยของชิ้นส่วนประกบทั้งสองฝั่ง ขันล็อกด้วยน็อตตัวเมียด้านล่างให้แน่นหนา เพื่อป้องกันเซนเซอร์เคลื่อนหลุด
4. **การติดตั้งเข้ากับส่วนฐาน (Base Mounting):** นำชุดเอ็นโค้ดเดอร์เตรียมติดตั้งเข้ากับฐาน ใส่บูชรอง M3 เข้ารูน็อตทั้ง 4 จุดเพื่อรักษาระยะช่องว่าง (Clearance) ระหว่างชุดเอ็นโค้ดเดอร์กับฐาน ใช้สกรู M3 ยาว 25 mm สอดผ่านรูและบูชรองเพื่อยึดเข้ากับส่วนฐาน

4.1.3 การประกอบรวมขั้นสุดท้าย (Final Assembly Process)

หลังจากประกอบชิ้นส่วนย่อยเสร็จสิ้น ดำเนินการประกอบรวมเป็นหุ่นยนต์เต็มรูปแบบตามลำดับดังนี้:

1. **การติดตั้งเอ็นโค้ดเดอร์เข้ากับฐาน:** นำชุดเอ็นโค้ดเดอร์ไปประกอบเข้ากับส่วนฐาน ขันน็อตยึดเข้ากับรูต๊าปเกลียวบนฐานให้แน่นหนา
2. **การใส่สายพาน:** นำชุดฐานที่ยึดเอ็นโค้ดเดอร์แล้ว มาทำการคล้องสายพาน (Belt) เตรียมไว้สำหรับการส่งกำลัง
3. **การประกอบแขนกลและมอเตอร์:** นำส่วนแขนกลและมอเตอร์ประกอบเข้ากับฐาน โดยสอดเพลาของส่วนแขนกลลงในรูร้อยแกนกลางของส่วนฐาน
4. **การตั้งค่าพรีโหลดตลับลูกปืน (Bearing Preload):** จัดวางให้ส่วนแขนกลและมอเตอร์กดทับลงบนตลับลูกปืนเม็ดเรียวของส่วนฐาน ขันน็อตที่เพลาบริเวณด้านบนสุดและบริเวณก่อนเข้าชุดเอ็นโค้ดเดอร์ เพื่อสร้างแรงดึงล่วงหน้า (Preload) ซึ่งจะช่วยลดระยะคลอน (Backlash) ของกลไกได้อย่างมีนัยสำคัญ
5. **การเชื่อมต่อเอ็นโค้ดเดอร์สนาม:** นำเพลาเชื่อมต่อมาต่อเข้ากับเพลาบริเวณด้านบนสุดของหุ่นยนต์ ใช้คัปปลิ้ง (Rigid/Flexible Shaft Coupling) ยึดเพื่อให้สามารถอ่านค่าตำแหน่งการหมุนของเพลาได้อย่างแม่นยำ
6. **การติดตั้งชุดมือจับ (Gripper):** นำชุดมือจับประกอบเข้ากับปลายแขนของหุ่นยนต์ ยึดด้วยน็อต 3 ตัว และขันล็อกให้แน่นหนาด้วยน็อตกันคลาย (Lock Nut)



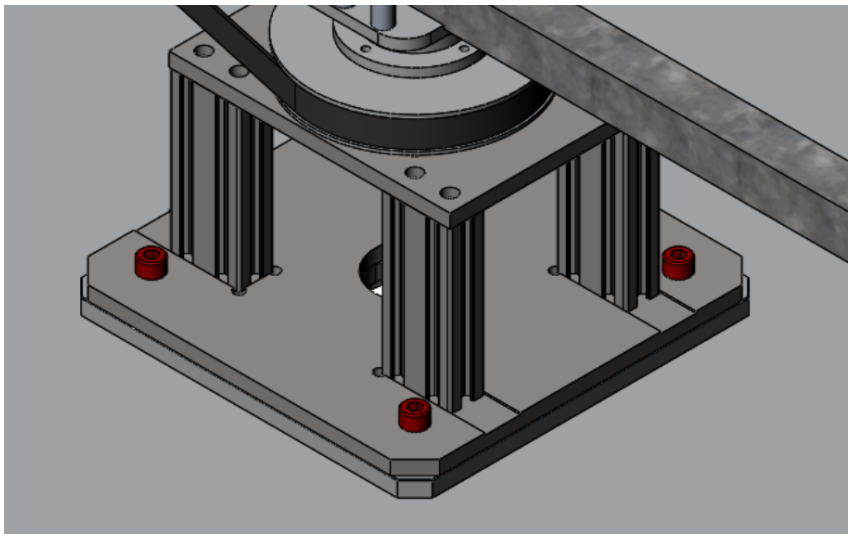
รูปที่ 4.9: ภาพประกอบรวมขั้นสุดท้ายของหุ่นยนต์ พร้อมชิ้นส่วน 3D Print สำหรับจัดระเบียบสายไฟ

4.1.4 ขั้นตอนการติดตั้ง ณ จุดใช้งาน (Field Installation Procedure)

หลังจากผ่านการประกอบและทดสอบระบบย่อยในห้องปฏิบัติการแล้ว การนำหุ่นยนต์ไปติดตั้ง ณ จุดใช้งานจริงดำเนินการตามลำดับดังนี้

ขั้นตอนที่ 1 — ยึดฐานเครื่องกับฐานสนาม

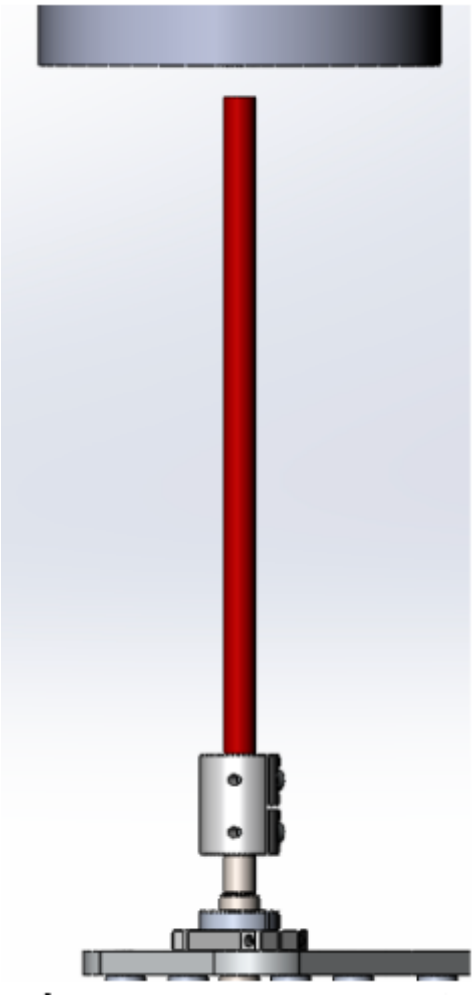
นำสกรูหัวจมดำ M8 × 14 mm จำนวน 4 ตัว ขันยึดฐานเครื่องเข้ากับฐานทดสอบของสนามให้แน่นด้วยประแจหกเหลี่ยม



รูปที่ 4.10: การยึดฐานเครื่องกับฐานสนามด้วยสกรู M8 × 14 mm ทั้ง 4 ตำแหน่ง

ขั้นตอนที่ 2 — ติดตั้ง Field Encoder Shaft

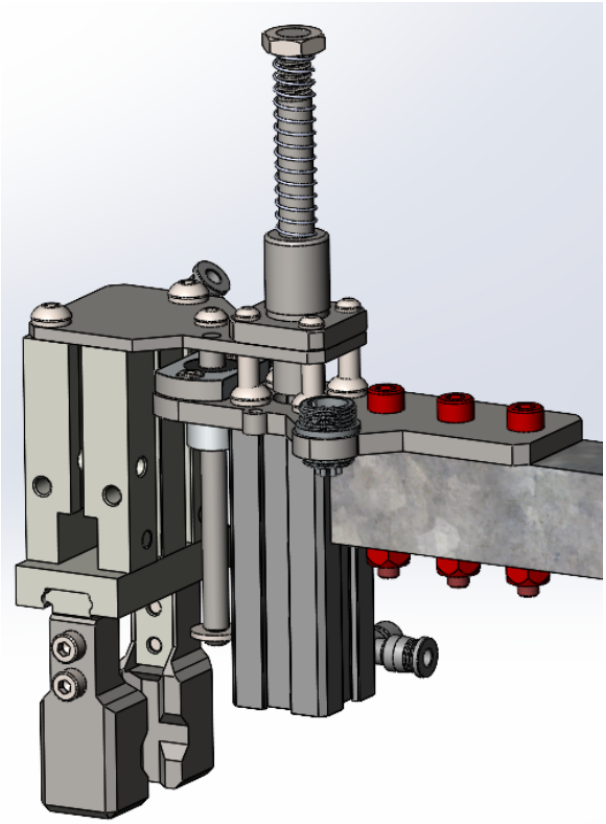
นำ Shaft ∅8 mm ยาว 200 mm สอดผ่าน Coupling โดยขันสกรูตัวหนอน (Set Screw) ก่อนสอด Shaft เพื่อตั้งตำแหน่ง จากนั้นสอด Shaft และขันสกรูตัวหนอนกลับเพื่อล็อกให้แน่น เชื่อม Coupling ของสนามเข้ากับ Shaft แท่งนี้



รูปที่ 4.11: การติดตั้ง Field Encoder Shaft ∅8 mm เข้ากับ Coupling

ขั้นตอนที่ 3 — ติดตั้ง Gripper

นำ Gripper ติดตั้งเข้ากับปลายแขนด้วยสกรู M5 × 40 mm จำนวน 3 ตัว และ Lock Nut M5 จำนวน 3 ตัว เมื่อยึดแน่นแล้วเสียบท่อลม (Label ตามลำดับ 1A, 1B, 2A, 2B) และสาย GX ให้เรียบร้อย



รูปที่ 4.12: การติดตั้ง Gripper ที่ปลายแขนด้วยสกรู M5 × 40 mm และ Lock Nut M5

การตรวจสอบก่อนจ่ายกระแสไฟ (Pre-Power Checklist)

ก่อนเปิดสวิตช์จ่ายไฟหลัก ตรวจสอบความเรียบร้อยของฮาร์ดแวร์ตามรายการต่อไปนี้:

- ☐ **Mechanical Check:** ไม่มีสิ่งกีดขวางในรัศมีการหมุน 360° ของหุ่นยนต์ และชุด Gripper อยู่ในตำแหน่งที่ปลอดภัย
- ☐ **Fastener Check:** สกรูยึดฐาน (M8), สกรูยึด Gripper (M5) และ Set Screw ของ Shaft ขันแน่นครบทุกตัว
- ☐ **Pneumatic Check:** ท่อลม 1A, 1B, 2A, 2B เสียบถูกพอร์ตและไม่มีรอยรั่ว
- ☐ **Cable Check:** สาย GX, สาย Motor และสาย Encoder ต่อครบและไม่ขวางเส้นทางหมุน
- ☐ **E-Stop Check:** กด E-Stop แล้วปล่อย ตรวจสอบว่าไฟ Emergency บนหน้าตู้ติดและดับตามสถานะ

4.1.5 ปัญหาที่พบระหว่างการประกอบและการแก้ไข

- **รู Proximity Sensor ขนาดไม่ตรง:** รูออกแบบไว้ ∅8 mm แต่ Sensor จริงมีขนาด ∅12 mm แก้ไขปัญหาเฉพาะหน้าโดยการเจาะขยายรูเพิ่มในห้องปฏิบัติการ

- **รู Driver Pulley และ Arm เบี้ยว:** ชิ้นส่วนที่ส่งผลิตภายนอกมีรูไม่ตรงแนวศูนย์กลาง เนื่องจากผู้รับเหมา Setup งานไม่ตรงกับแบบ Drawing แก้ไขโดยนำชิ้นงานกลับไปปรับแต่งเพิ่มเติมด้วยตนเอง
- **แผ่นรอง Pulley ไม่ได้ปาดผิว:** ผู้ผลิตไม่ได้ปาดผิวและร่องของ Aluminium Profile ตามแบบที่กำหนด ดำเนินการแก้ไขเองในห้องปฏิบัติการเพื่อให้ประกอบชิ้นส่วนเข้ากันได้พอดี

4.2. การประกอบตู้ควบคุมและการเดินสายไฟ

4.2.1 รายการอุปกรณ์หลัก (Bill of Materials)

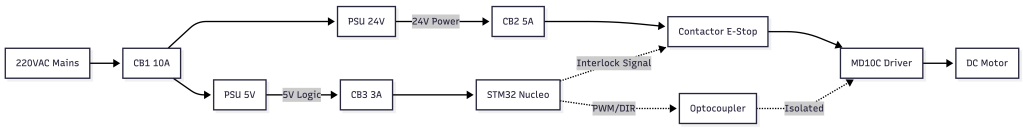
การออกแบบตู้ควบคุม (Control Cabinet) มุ่งเน้นไปที่การแยกส่วนระหว่างวงจรควบคุมกำลัง (High Power Side) และวงจรลอจิก (Logic Side) เพื่อป้องกันสัญญาณรบกวน (Noise) และความเสียหายที่อาจเกิดขึ้นกับไมโครคอนโทรลเลอร์ รายการอุปกรณ์หลักที่ใช้ในการประกอบแสดงในตารางที่ 4.4

ตารางที่ 4.4: Bill of Materials (BOM) ของตู้ควบคุม

Reference	อุปกรณ์	จำนวน	หมายเหตุ
U1A	STM32G474RE Nucleo Board	1	Main Controller
U2	ESP32-DEVKIT-V1	1	Joystick Receiver
--	Cytron MD10C Motor Driver	1	24V/10A
CB1	Circuit Breaker 10A	1	AC Input
CB2	Circuit Breaker 5A	1	24V Bus
CB3	Circuit Breaker 3A	1	5V Bus
--	PSU 24V/10A	1	Motor Power
--	PSU 5V/5A	1	Logic Power
--	Magnetic Contactor	1	E-Stop Interlock
--	Optocoupler Module 8-ch	1	Signal Isolation
--	Relay Module 4-ch	1	Output Control
--	Emergency Stop Button	1	NC Contact
--	Pilot Lamp (Blue/Red/Green)	3	Power/Fault/Ready
--	Reset Button	1	NC Contact
--	Mode Selector Switch	1	Manual/Auto
--	Fuse (Motor Circuit)	1	Short Circuit Protection

4.2.2 สถาปัตยกรรมการจ่ายไฟและการแยกสัญญาณ (Power & Signal Architecture)

ระบบไฟฟ้าภายในตู้ถูกออกแบบให้มีการแยกวงจรอย่างชัดเจน (Galvanic Isolation) ทั้งในระดับของแหล่งจ่ายไฟและระดับของสัญญาณสื่อสาร ดังแสดงในแผนภาพรูปที่ 4.13



รูปที่ 4.13: Block Diagram แสดงสถาปัตยกรรมการจ่ายไฟและการแยกสัญญาณภายในตู้ควบคุม

หลักการทำงานที่สำคัญของสถาปัตยกรรมนี้ประกอบด้วย:

1. **วงจรความปลอดภัยทางฮาร์ดแวร์ (Hardware Interlock):** ปุ่ม Emergency Stop ทำงานร่วมกับ Magnetic Contactor แบบ N.C. (Normally Closed) เมื่อเกิดเหตุฉุกเฉิน ไฟ 24V ที่จ่ายไปยัง MD10C จะถูกตัดวงจรทางกายภาพทันที โดยที่บอร์ด STM32 (5V) ยังคงทำงานได้เพื่อส่งสัญญาณแจ้งเตือน Fault ไปยังระบบเครือข่าย
2. **การแยกกราวด์ (Ground Isolation):** สัญญาณ PWM และ Direction จาก STM32 จะต้องผ่าน Optocoupler Module แบบ 8-channel ก่อนเข้าสู่ MD10C เพื่อแยก Logic Ground ออกจาก Power Ground ป้องกันปัญหา Ground Loop และลดผลกระทบจาก Back EMF ที่เกิดจากการเบรกของมอเตอร์

หลักการทำงานที่สำคัญของสถาปัตยกรรมนี้ประกอบด้วย:

1. **วงจรความปลอดภัยทางฮาร์ดแวร์ (Hardware Interlock):** ปุ่ม Emergency Stop ทำงานร่วมกับ Magnetic Contactor แบบ N.C. (Normally Closed) เมื่อเกิดเหตุฉุกเฉิน ไฟ 24V ที่จ่ายไปยัง MD10C จะถูกตัดวงจรทางกายภาพทันที โดยที่บอร์ด STM32 (5V) ยังคงทำงานได้เพื่อส่งสัญญาณแจ้งเตือน Fault ไปยังระบบเครือข่าย
2. **การแยกกราวด์ (Ground Isolation):** สัญญาณ PWM และ Direction จาก STM32 จะต้องผ่าน Optocoupler Module แบบ 8-channel ก่อนเข้าสู่ MD10C เพื่อแยก Logic Ground ออกจาก Power Ground ป้องกันปัญหา Ground Loop และลดผลกระทบจาก Back EMF ที่เกิดจากการเบรกของมอเตอร์

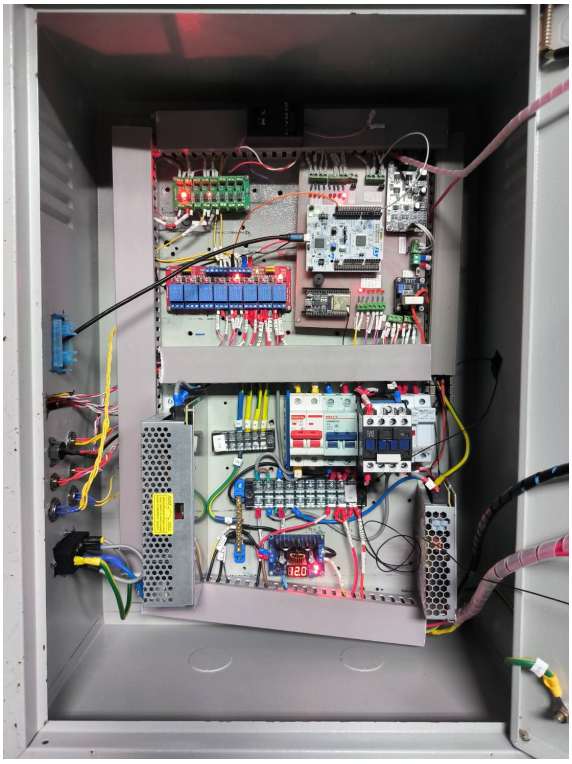
4.2.3 แนวทางปฏิบัติในการเดินสายไฟ (Cable Management & Wiring)

การเดินสายภายในและภายนอกตู้ควบคุมใช้มาตรฐานการเดินสายอุตสาหกรรม โดยมีข้อกำหนดดังต่อไปนี้:

- **การเลือกขนาดสายไฟ (Wire Sizing):** วงจรขับมอเตอร์ (24V Bus) ใช้สายไฟขนาด 16 AWG เพื่อรองรับกระแสสูงสุด (Peak Current) ได้ถึง 10A โดยไม่เกิด Voltage Drop ที่มีนัยสำคัญ ส่วนวงจรลอจิกและเซนเซอร์ (5V Bus) ใช้สายไฟขนาด 22-24 AWG
- **การเผื่อระยะสายบริเวณจุดหมุน (Service Loop):** บริเวณข้อต่อของแขนกล (Joints) มีการเผื่อความยาวของสายไฟ (Slack) ในรูปแบบของ U-loop เพื่อให้สายไฟสามารถโค้งงอได้อย่างอิสระ ป้องกันแรงดึงรั้ง (Tension) ที่จะทำให้ทองแดงภายในขาดล้า (Fatigue) เมื่อหุ่นยนต์เคลื่อนที่ไปจนถึงมุมสูงสุด (Maximum Reach)
- **รัศมีการโค้งงอ (Bending Radius) และการนำสาย:** ติดตั้งชิ้นส่วน 3D Print บริเวณขอบและมุมของโครงสร้างหุ่นยนต์ เพื่อบังคับให้สายโค้งงอด้วยรัศมีที่มากกว่า 4 เท่าของเส้นผ่านศูนย์กลางสายไฟ ป้องกันการหักมุมแคบซึ่งจะทำลายฉนวนหุ้มสายสัญญาณภายใน

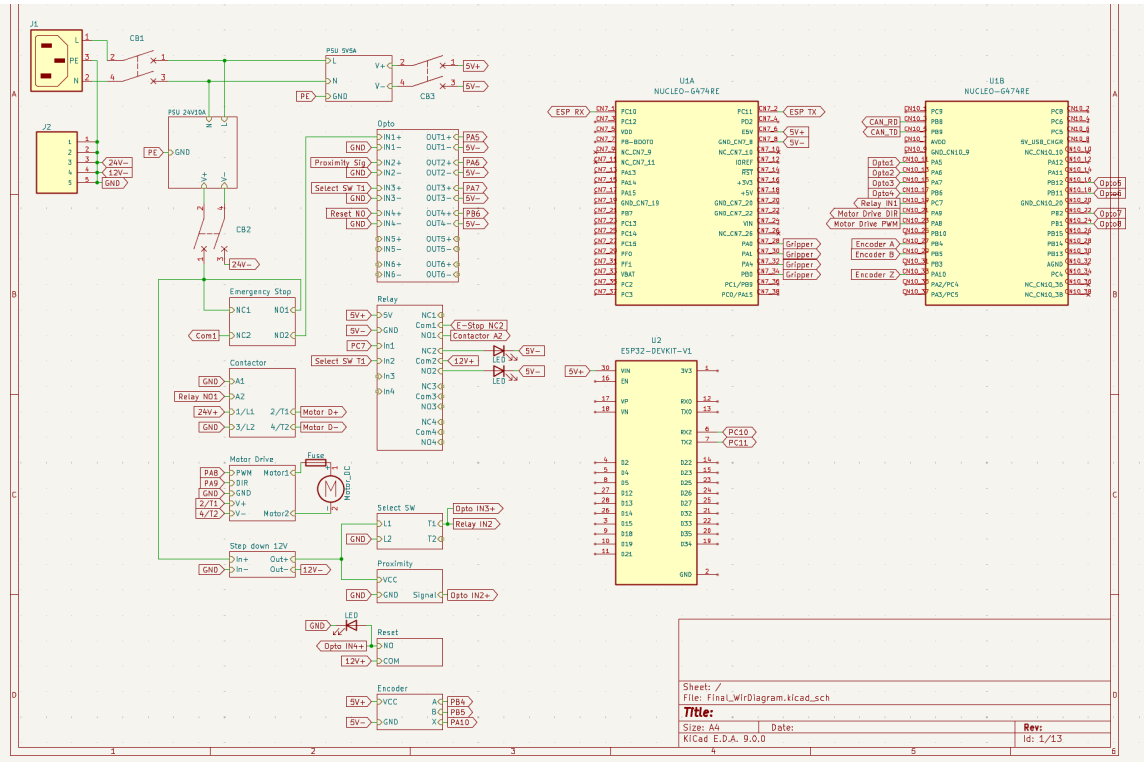
- **การแยกรางสายไฟ (Trunking Separation):** สายสัญญาณจาก Encoder (ที่มีความไวต่อ Noise สูง) จะถูกเดินแยกท่อหรือรางสายไฟออกจากสาย Power ของมอเตอร์อย่างเด็ดขาด เพื่อป้องกันคลื่นแม่เหล็กไฟฟ้า (EMI) เหนี่ยวนำเข้าสู่สายสัญญาณ

(a) หน้าตู้ควบคุม แสดงไฟสถานะ Power/Emergency และปุ่ม E-Stop



(b) ภายในตู้ควบคุม แสดงการจัดเรียงอุปกรณ์ และการเดินสายในรางพลาสติก

รูปที่ 4.14: ตู้ควบคุมไฟฟ้าที่ประกอบเสร็จสมบูรณ์: หน้าตู้ (ซ้าย) และภายในตู้ (ขวา)



รูปที่ 4.15: แผนผังการต่อสายไฟ (Wiring Diagram) ของระบบควบคุม

4.3. การพัฒนา Firmware

4.3.1 ภาพรวมสถาปัตยกรรมซอฟต์แวร์ (Software Architecture)

Firmware สำหรับ ควบคุม หุ่น ยนต์ พัฒนา บน ไมโคร คอนโทรลเลอร์ STM32G474RE (Cortex-M4, 170 MHz) ผ่าน แพลตฟอร์ม STM32CubeIDE สถาปัตยกรรม ซอฟต์แวร์ ถูก ออกแบบ ด้วย แนวคิด Modular Programming และ Separation of Concerns โดย แยก ระบบ ควบคุม การ เคลื่อนที่ (Motion Control) ออก จาก ระบบ สื่อสาร (Communications) และระบบความปลอดภัย (Safety Stack) อย่างชัดเจน การออกแบบลักษณะนี้ช่วยลดความพึ่งพากันของโค้ด (Loose Coupling) ทำให้สามารถทดสอบระดับหน่วย (Unit Test) แยกเฉพาะโมดูลได้ง่ายขึ้น ข้อมูลสถานะทั้งหมดถูกจัดเก็บในโครงสร้างตัวแปรกลาง (`g_robot`) เพื่อให้ทุกโมดูลอ้างอิงข้อมูลชุดเดียวกัน (Single System View) โครงสร้างไลบรารีและโมดูลหลักแสดงในตารางที่ 4.5

ตารางที่ 4.5: โครงสร้างโมดูลและไลบรารีใน STM32 Firmware

โมดูล / ไลบรารี	หน้าที่การทำงาน	ไฟล์อ้างอิง
Core Application		
`app_main`	Application FSM, กระบวนการ Homing, พฤติกรรมภาพรวม	`app_main.c`
`motor_controller`	จัดการ Task Timing (1 kHz / 100 Hz), Safety Stack	`motor_controller.c`
`gripper_seq`	ลำดับการทำงานและ FSM ของชุดหัวจับ (Pick & Place)	`gripper.c`
Math & Control Lib		
`lib_scurve`	S-Curve Trajectory Generator (จำกัดความเร็ว, ความเร่ง, Jerk)	`scurve.c`
`lib_zvd`	ZVD Input Shaper ด้วยโครงสร้าง Circular Buffer	`zvd.c`
`lib_kalman`	Steady-State Kalman Filter 4 สถานะ	`kalman.c`
`lib_pid`	Cascade PID Control พร้อมอัลกอริทึม Anti-windup	`pid.c`
Comms & I/O		
`modbus_bridge`	Modbus RTU Protocol (Slave 0x15), Mapping Registers	`modbus_bridge.c`
`canbus_node`	FDCAN Gripper Node, Relay/Sensor Bridge	`canbus.c`
`joystick`	ถอดรหัสคำสั่งจาก ESP32 Gamepad (USART3)	`joystick.c`

4.3.2 การจัดสรรเวลาทำงาน (Execution Model และ Task Timing)

ระบบควบคุมใช้วิธีการจัดสรรเวลาแบบ Hard Real-Time โดยใช้ตัวตั้งเวลา TIM6 เป็นตัวกำหนด Base Rate ที่ 1 kHz วงรอบการควบคุมถูกแบ่งออกเป็น 2 ความถี่ผ่าน Software Counter ภายใน Interrupt Service Routine (ISR) เดียวกัน เพื่อหลีกเลี่ยงปัญหาการสลับบริบท (Context Switching)

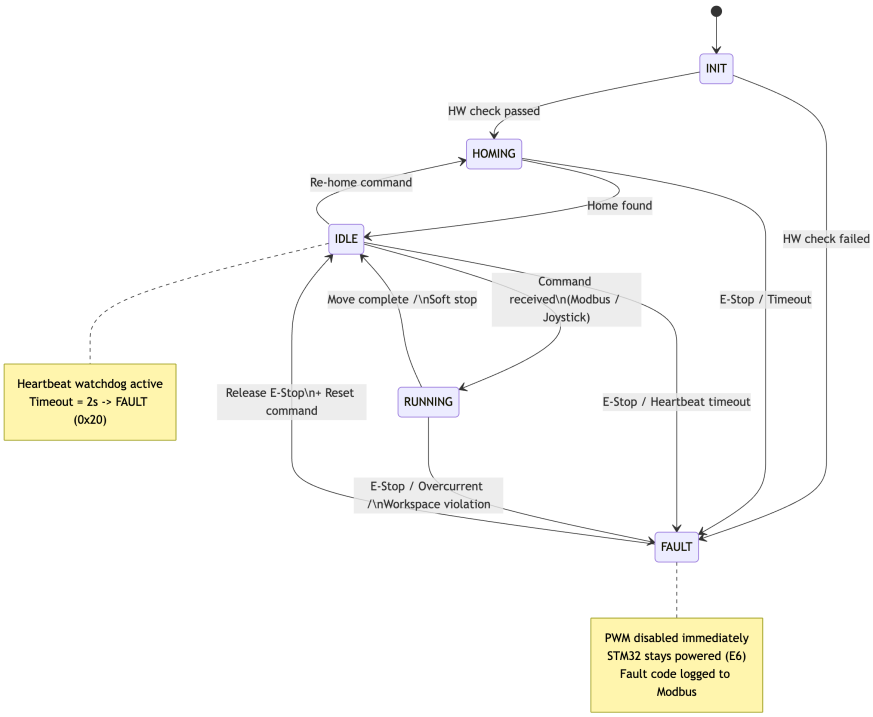
- **ทุก 1 ms (1 kHz - Inner Loop):** ทำงานใน `MotorCtrl_Tick1kHz()` เริ่มจากการอ่านค่าตำแหน่ง Encoder จาก TIM3 นำไปผ่าน Kalman Filter เพื่อประมาณค่าความเร็วและความเร่ง จากนั้นคำนวณ Velocity Inner PID และ Feedforward ตรวจสอบเงื่อนไขความปลอดภัย ก่อนส่งค่า PWM อัปเดตไปยัง TIM1
- **ทุก 10 ms (100 Hz - Outer Loop):** ทำงานใน `MotorCtrl_Tick100Hz()` และ `HwIo_Poll100Hz()` รับผิดชอบการกรองสัญญาณรบกวน (Debounce) ของเซนเซอร์ ประมวลผลสมการ S-Curve Trajectory และ ZVD Input Shaper เพื่อสร้าง Setpoint สำหรับ Position Outer PID

นอกจากการประมวลผลใน ISR แล้ว ในส่วนของ **Main Loop Context** จะทำงานแบบปล่อยอิสระ (Free-

running) ใช้เวลาประมาณ $20\,\mu s$ ต่อรอบ ทำหน้าที่ตรวจสอบ Application FSM, จัดการข้อมูล Modbus ใน Ring Buffer, อัปเดตข้อมูล CAN Bus และส่งสัญญาณ Heartbeat

4.3.3 สถานะการทำงานของระบบ (Application FSM)

กระบวนการทำงานหลักของหุ่นยนต์ถูกควบคุมด้วย Finite State Machine (FSM) ซึ่งแบ่งออกเป็น 5 สถานะหลัก เพื่อความปลอดภัยและป้องกันการทำงานที่ผิดพลาด ดังแสดงในรูปที่ 4.16



รูปที่ 4.16: แผนภาพสถานะการทำงาน (Application FSM) ของระบบควบคุม

- **STATE_INIT:** สถานะเริ่มต้นเมื่อเปิดเครื่อง เพื่อเตรียมความพร้อมของ Peripheral
- **STATE_HOMING:** สถานะค้นหาจุดอ้างอิงศูนย์ ระบบจะไม่ยอมรับคำสั่งเคลื่อนที่อื่น
- **STATE_IDLE:** สถานะพร้อมทำงาน รอรับคำสั่ง Jog, Point-to-Point หรือ Auto
- **STATE_RUNNING:** สถานะกำลังเคลื่อนที่ตามคำสั่ง หรือกำลังทำ Auto Sequence
- **STATE_FAULT:** สถานะขัดข้องเมื่อเกิดข้อผิดพลาดทางฮาร์ดแวร์หรือความปลอดภัย ระบบจะตัดกำลังมอเตอร์และรอการปลดล็อก (Reset)

4.3.4 กระบวนการหาจุดอ้างอิงระดับความแม่นยำสูง (Two-Stage Homing Sequence)

เพื่อขจัดปัญหาความคลาดเคลื่อนของตำแหน่งที่เกิดจากความหน่วงของเซนเซอร์ (Sensor Latency) ระบบจึงใช้วิธีการหาจุดอ้างอิงแบบ 2 ขั้นตอน (Two-Stage Industrial Homing) โดยอาศัยหลักการหาจุดกึ่งกลาง (Midpoint) ของขอบ

สัญญาณ (Edge) ทั้งสองด้าน

กระบวนการแบ่งเป็น 8 ขั้นตอนย่อยดังนี้:

1. **FAST_SEARCH:** หุ่นยนต์เคลื่อนที่กวาดเพื่อค้นหาพื้นที่ของ Proximity Sensor ด้วยความเร็วสูง (Coarse Search)
2. **BACKOFF:** เมื่อเจอขอบสัญญาณ ระบบจะถอยหลังออกด้วยระยะเผื่อ (Clearance Margin)
3. **PREC_EDGE_A:** เคลื่อนที่เข้าหาเซนเซอร์ด้วยความเร็วต่ำสม่ำเสมอ บันทึกตำแหน่งที่สัญญาณขอบขาขึ้น (Rising Edge) ปรากฏเป็นตำแหน่ง *Edge A*
4. **PREC_OVERSHOOT:** เคลื่อนที่ทะลุผ่านเซนเซอร์จนสัญญาณดับลง เพื่อป้องกันการอ่านขอบสัญญาณผิดพลาดซ้ำซ้อน
5. **FIND_EDGE_B:** เคลื่อนที่ถอยหลังกลับด้วยความเร็วต่ำสม่ำเสมอ บันทึกตำแหน่งที่สัญญาณขอบขาขึ้นปรากฏเป็นตำแหน่ง *Edge B*
6. **GO_CENTER & SETTLE:** คำนวณจุดกึ่งกลาง $\frac{\text{Edge A}+\text{Edge B}}{2}$ แล้วสั่งให้หุ่นยนต์เคลื่อนที่ไปยังจุดนั้น ระบบจะหน่วงเวลา 1 วินาที และกำหนดให้ตำแหน่งปัจจุบันเป็น 0 (Zero)

วิธีการนี้รับประกันได้ว่าตำแหน่งศูนย์ทางกล (Mechanical Zero) จะไม่เอนเอียงไปตามทิศทางการเคลื่อนที่หรือความเร็วในการตรวจจับ

4.3.5 อัลกอริทึมควบคุมการเคลื่อนที่ (Motion Control Algorithms)

S-Curve Trajectory และ ZVD Input Shaper

ระบบสร้างโพรไฟล์การเคลื่อนที่แบบ S-Curve เพื่อจำกัดความเร็วสูงสุด (7.30 rad/s), ความเร่งสูงสุด (27.49 rad/s^2) และค่า Jerk (1400 rad/s^3) อย่างเคร่งครัด จากนั้นนำ Setpoint ที่ได้ไปผ่าน ZVD Input Shaper ที่ทำงานแบบ Circular Buffer เพื่อลดการสั่นทอนที่ความถี่ธรรมชาติ $f_n = 12.13$ Hz ตามสมการ (4.1)

$$r_{shaped}(t) = 0.2832\, r(t) + 0.4981\, r(t - t_2) + 0.2189\, r(t - t_3) \tag{4.1}$$

โดยกำหนดให้ระยะหน่วง $t_2 = 4$ steps และ $t_3 = 8$ steps (ที่ความถี่ 100 Hz)

Steady-State Kalman Filter

เนื่องจากการหาอนุพันธ์ของตำแหน่งจาก Encoder สร้างสัญญาณรบกวนความถี่สูง ระบบจึงใช้ Kalman Filter 4 สถานะ $\mathbf{x} = [\theta, \omega, \alpha, j]^\top$ ทำงานทุก 1 ms เพื่อประมาณค่าความเร็วและความเร่ง ค่า Steady-State Gain ($\mathbf{K}_\infty$) ถูกคำนวณล่วงหน้าผ่านการแก้สมการ DARE (Discrete Algebraic Riccati Equation) บน MATLAB เพื่อลดภาระการคำนวณของไมโครคอนโทรลเลอร์

$$\hat{\mathbf{x}}_{k|k} = \mathbf{A}_d\, \hat{\mathbf{x}}_{k-1} + \mathbf{B}_d\, u_{k-1} + \mathbf{K}_\infty (y_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}) \tag{4.2}$$

Cascade PID (Discrete Tustin)

ระบบควบคุมแบบวงรอบซ้อน (Cascade PID) ใช้สมการแบบ Tustin (Bilinear Approximation) สำหรับ Velocity Inner Loop (1 kHz) มีการประยุกต์ใช้ Anti-windup (Back-calculation) เพื่อป้องกัน Integral Windup เมื่อ PWM เข้าสู่ขอบเขตการอิ่มตัว ± 50

$$u_{vel}[k] = K_{p,v}\, e_v[k] + \text{integ}[k] + K_{d,v} \frac{e_v[k] - e_v[k-1]}{T_v} \tag{4.3}$$

$$\text{integ}[k] = \text{integ}[k-1] + K_{i,v}\, e_v[k]\, T_v + K_{t,aw}(u_{sat}[k] - u_{pre}[k])\, T_v \tag{4.4}$$

ส่วน Position Outer Loop (100 Hz) จะคำนวณ Setpoint เชิงความเร็วเพื่อป้อนให้กับ Inner Loop ค่า Gain ที่ใช้ใน Firmware แสดงในตารางที่ 4.6

ตารางที่ 4.6: PID Gains และ Feedforward ที่ใช้งานจริงบนระบบฮาร์ดแวร์

Loop	อัตราขยาย	สัญลักษณ์	ค่า	หน่วย
Velocity (1 kHz)	Proportional	$K_{p,vel}$	1.58	PWM/(rad/s)
	Integral	$K_{i,vel}$	0.50	PWM/rad
	Derivative	$K_{d,vel}$	0.00	--
Position (100 Hz)	Proportional	$K_{p,pos}$	3.00	(rad/s)/rad
	Integral	$K_{i,pos}$	0.02	--
	Derivative	$K_{d,pos}$	0.15	--
Feedforward	Velocity FF	K_{vff}	3.03	PWM/(rad/s)
	Accel FF	K_{aff}	0.10	PWM/(rad/s^2)

4.3.6 ระบบความปลอดภัยทางซอฟต์แวร์ (Safety Stack)

เพื่อป้องกันความเสียหายต่อกลไกและมอเตอร์ Firmware มีการตรวจจับความผิดปกติ 4 ระดับและจะบังคับให้ FSM เปลี่ยนเป็น `STATE_FAULT` ทันที:

- **Encoder Disconnect Guard:** หากระบบจ่าย PWM เกินระดับที่กำหนด (PWM > 5) ต่อเนื่องเป็นเวลา 200 ms แต่ตำแหน่ง Encoder ไม่มีการเปลี่ยนแปลง ระบบจะพิจารณาว่าสาย Encoder ขาดหรือขั้วหลุดหลวม
- **Overcurrent Fuse:** อ่านกระแสจากเซนเซอร์ WCS1800 ผ่าน ADC (PA0) หากกระแสสูงกว่า 2.0 A ต่อเนื่อง 100 ms จะตัดการทำงาน (จำลองการทำงานของฟิวส์หน่วงเวลา)
- **Tracking Error Guard:** ตรวจสอบความแตกต่างระหว่าง Position Setpoint และตำแหน่งจริง หากคลาดเคลื่อนเกิน 10 องศา ต่อเนื่อง 50 ms ระบบจะถือว่าเกิดการติดขัดทางกล (Mechanical Jam)
- **Boundary Limits:** จำกัดพื้นที่การทำงานผ่านซอฟต์แวร์ที่ระยะ 540 องศา เพื่อป้องกันปัญหาสายเคเบิลม้วนพันกัน (Cable Over-twist)

4.3.7 การสื่อสารผ่านโครงข่ายและ CAN Bus Failsafe

ระบบรองรับการควบคุมจาก 2 แหล่ง ได้แก่ ESP32 Bluetooth Gamepad (ผ่าน USART3) สำหรับโหมด Manual และระบบ PC Backend (ผ่าน LPUART1 โปรโตคอล Modbus RTU) สำหรับโหมด Automation โดยมีการส่งสัญญาณ Heartbeat ข้ามไปมา

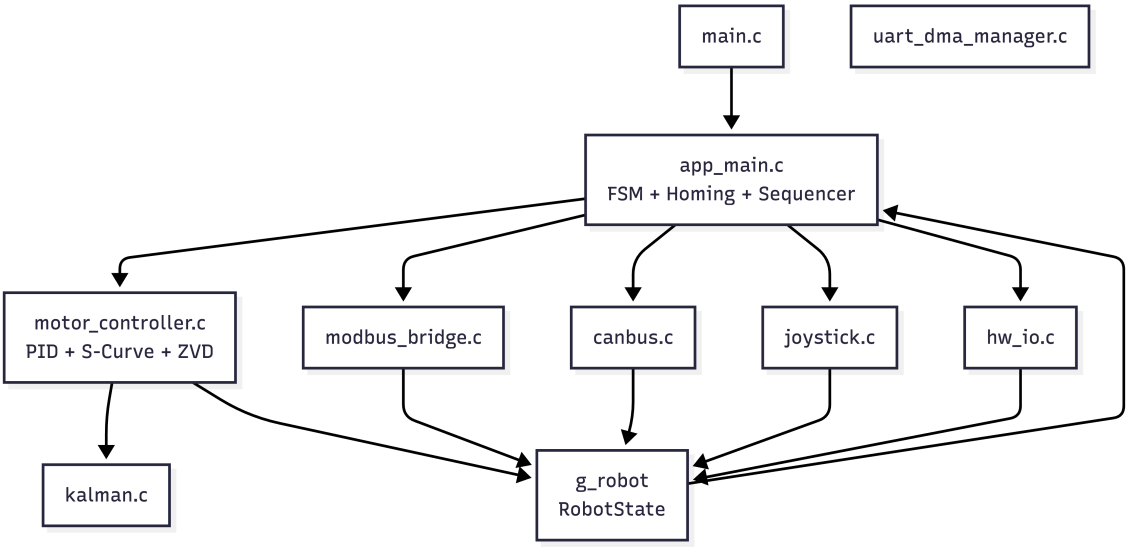
ชุดหัวจับ (Gripper) ควบคุมผ่านเครือข่าย **FDCAN1** ด้วยความเร็ว 500 kbps ซึ่งทำหน้าที่เป็น I/O Node (ID 0x10) ในกรณีที่ Node ไม่ได้รับสัญญาณ Heartbeat จาก Master ภายในเวลา 2 วินาที (2000 ms) ระบบจะมี **CAN Watchdog Failsafe** ทำหน้าที่ตัดไฟ Relay ทั้งหมดโดยอัตโนมัติ ทำให้สปริงดึงกลไกกลับสู่สภาวะปลอดภัย

4.4. การบูรณาการระดับระบบ (System Integration)

4.4.1 ภาพรวมสถาปัตยกรรมระบบ (System Architecture)

การบูรณาการหุ่นยนต์ (System Integration) ครอบคลุมการเชื่อมต่อส่วนประกอบทางฮาร์ดแวร์ ซอฟต์แวร์ และโครงข่ายการสื่อสารเข้าด้วยกันเป็นระบบเดียว สถาปัตยกรรมแบบ Cyber-Physical System ของหุ่นยนต์แสดงในรูปที่ 4.17 ซึ่งแบ่งออกเป็น 4 ส่วนหลัก ได้แก่

1. **Base System (PC Master):** ทำหน้าที่เป็นศูนย์กลางการสั่งการระดับสูงผ่าน Python Backend และ Web Dashboard
2. **Main Controller:** ไมโครคอนโทรลเลอร์ STM32G474RE ทำหน้าที่ประมวลผลอัลกอริทึมควบคุมการเคลื่อนที่ (Motion Control) และ Application FSM
3. **Actuators & Sensors:** มอเตอร์ขับเคลื่อน, ชุดหัวจับ, เอ็นโค้ดเดอร์ และเซนเซอร์วัดกระแส
4. **Subsystems:** เครือข่าย CAN Bus สำหรับควบคุมชุดหัวจับ และระบบ ESP32 สำหรับรับคำสั่งจากจอยสติ๊กไร้สาย



รูปที่ 4.17: สถาปัตยกรรมระบบ (Cyber-Physical System Architecture) ของหุ่นยนต์

4.4.2 การจัดการข้อมูลร่วม (Shared Data Architecture)

Firmware ถูกออกแบบให้มีการแลกเปลี่ยนข้อมูลผ่านโครงสร้างตัวแปรส่วนกลาง `g_robot` (`RobotState_t`) ซึ่งทำหน้าที่เป็น Single Source of Truth ดังแสดงในแผนภาพสถาปัตยกรรมระบบ (รูปที่ 4.17)

กลยุทธ์การจัดการข้อมูลแบ่งออกเป็นสองระดับ:

1. **Interrupt Context (High Priority):** โมดูล `motor_controller` และ `lib_kalman` จะทำหน้าที่อ่านและเขียนข้อมูลสถานะการเคลื่อนที่ (เช่น `position_counts`, `velocity_rps`) ลงใน `g_robot` โดยตรงด้วยความถี่ 1 kHz เพื่อให้แน่ใจว่าลูปควบคุมทำงานด้วย Time-step ที่คงที่
2. **Main Loop Context (Low Priority):** โมดูลสื่อสาร (`modbus_bridge`, `canbus_node`, `joystick`) จะทำการอ่านข้อมูลล่าสุดจาก `g_robot` เพื่อส่ง Telemetry ไปยัง PC หรือรับคำสั่งใหม่เข้ามา ซึ่งช่วยลดความหน่วง (Latency) ของการประมวลผลคำสั่งสื่อสารที่อาจขัดจังหวะลูปการควบคุมหลักได้

4.4.3 การเชื่อมต่ออุปกรณ์ขับเคลื่อนและเซนเซอร์ (Motor, Encoder & Driver)

การเชื่อมต่อในส่วนการขับเคลื่อนให้ความสำคัญกับความละเอียดในการวัดตำแหน่งและการป้องกันสัญญาณรบกวนทางไฟฟ้า ดังนี้:

- **Encoder Interface:** ชุดเซนเซอร์ AMT-103 (ความละเอียด 2048 PPR) ถูกเชื่อมต่อผ่านโหมด Quadrature Encoder Interface (QEI) บน TIM3 ของ STM32 (ขา PB4 และ PB5) โดยใช้การอ่านแบบ $\times 4$ ทำให้ได้ความละเอียดสูงถึง 8192 counts ต่อรอบการหมุน
- **Motor Control:** มอเตอร์ขับเคลื่อนผ่าน Cytron MD10C ซึ่งรับสัญญาณ PWM จาก TIM1 CH1 (PA8) ที่ความถี่ 19.6 kHz (ARR=50) เพื่อหลีกเลี่ยงสัญญาณรบกวนในย่านที่หูมนุษย์ได้ยิน ทั้งนี้สัญญาณควบคุมทิศทาง

(DIR) ถูกส่งผ่าน PA9 โดยผ่าน Optocoupler Module เพื่อแยกกราวด์ (Galvanic Isolation) ระหว่างวงจรควบคุมและวงจรขับกำลังอย่างสมบูรณ์

4.4.4 การเชื่อมต่อระบบควบคุมไร้สาย (Joystick Integration)

ESP32-DEVKIT-V1 ทำหน้าที่เป็นตัวรับสัญญาณ Bluetooth จากจอยสติ๊กและแปลงข้อมูลเป็น Command Character ส่งต่อให้ STM32 ผ่าน USART3 (115200 bps, 8N1) ที่ขา PC10 (TX) และ PC11 (RX) เพื่อใช้ในการควบคุมระยะไกล จอยสติ๊กที่ใช้คือ MOCUTE-052 VR ดังแสดงใน**??**

รูปที่ 4.18: จอยสติ๊ก MOCUTE-052 VR ที่ใช้สำหรับการควบคุมระยะไกลผ่าน Bluetooth

ตารางที่ 4.7: ตารางการ Map Command Character สำหรับ Joystick

Char	คำสั่ง	โหมดที่ใช้ได้	หมายเหตุ
`L/R`	Jog ซ้าย/ขวา	Manual	Coarse/Fine ปรับได้
`0`	Release / Stop	Manual	หยุดการเคลื่อนที่
`A`	Home	ทุกโหมด	Hold 3 s = Home, <3 s = Reset Encoder
`P`	E-Stop / Reset	ทุกโหมด	Toggle E-Stop ทางซอฟต์แวร์
`M`	สลับโหมด	Manual	Toggle Coarse/Fine
`U/D`	Gripper Up/Down	Manual	ควบคุมแนวตั้ง
`F/S/G`	Gripper Op/Pick/Pl	Manual	Open/Close/Pick/Place
`J`	Hold Position	Manual	ล็อกตำแหน่งปัจจุบัน

ความปลอดภัยในการสลับโหมด: การสลับระหว่างโหมด Joystick และ Base System ทำผ่านการกดปุ่ม B ค้างไว้ 1 วินาที เพื่อตั้งค่าตัวแปร `control_system_mode` หากระบบอยู่ในสถานะ `CONTROL_MODE_BASE_SYSTEM` คำสั่งจากจอยสติ๊กทุกตัวจะถูกบล็อก ยกเว้นปุ่ม `P` (E-Stop), `A` (Home) และ `M` (Mode Toggle) เพื่อรับประกันว่าผู้ควบคุมสามารถหยุดระบบฉุกเฉินได้จากทุกโหมดการทำงาน

4.4.5 การสื่อสารด้วยโปรโตคอล Modbus RTU

ระบบสื่อสารกับ Base System ผ่านโปรโตคอล Modbus RTU บน LPUART1 (230400 bps, 8E1, RS-485) โดยกำหนดให้ STM32 เป็น Slave Address 21 (0x15)

สถาปัตยกรรม Backend: Base System เปลี่ยนมาใช้ Python `asyncio` ร่วมกับ WebSocket Backend แทน `main.exe` เดิม เพื่อแก้ปัญหาภาวะ Abnormal Heartbeat ที่เคยพบในเวอร์ชันทดสอบที่ 1 โดยตัว Backend จะรับคำสั่งจาก Frontend ผ่าน WebSocket แล้วทำการแปลง (Translate) เป็น Modbus RTU Frame ส่งผ่าน Serial Port ไปยังระบบหุ่นยนต์

Modbus Register Map: ประกอบด้วยพื้นที่หน่วยความจำ 128 Registers แบ่งกลุ่มการใช้งานดังนี้:

- **Status Registers:** สำหรับอ่านค่าตำแหน่ง (Position), ความเร็ว (Velocity) และสถานะของเครื่องจักร (FSM State) แบบ Real-time
- **Command Registers:** สำหรับรับค่าเป้าหมาย (Target Position), การเลือกโหมด (Mode) และคำสั่งการทำงานของชุดหัวจับ (Gripper Command)

การออกแบบเช่นนี้ช่วยให้การสื่อสารระหว่างคอมพิวเตอร์แม่ข่ายและตัวควบคุมระดับต่ำ (Embedded) มีความน่าเชื่อถือสูง แม้ในสภาวะที่มีการรับส่งข้อมูลจำนวนมาก

บทที่ 5 ผลการทดสอบและการทวนสอบ

5.1. ผลการทดสอบระบบย่อย

การทดสอบระบบย่อยดำเนินการก่อนการทดสอบสมรรถนะเต็มรูปแบบ เพื่อยืนยันว่าส่วนประกอบแต่ละชิ้นทำงานถูกต้องตามข้อกำหนดทางกลและทางไฟฟ้า ก่อนนำไปประกอบเป็นระบบควบคุมแบบ Closed-loop ข้อมูลในหัวข้อนี้เก็บผ่าน STM32CubeMonitor โดย Log ตัวแปรสถานะผ่าน SWD/JTAG ที่ Sample Rate 100 Hz เว้นแต่ระบุเป็นอื่น

5.1.1 การวัดระยะคลอน (Backlash Measurement)

วัตถุประสงค์และความสำคัญ

ระยะคลอน (Backlash) ในระบบส่งกำลังทำให้เกิด Dead-zone ในลูปควบคุม ซึ่งส่งผลโดยตรงต่อความแม่นยำตำแหน่ง (A1: $\leq \pm 0.5°$) และความซ้ำซากได้ (A2: $\leq \pm 0.1°$) ในระบบนี้มีแหล่งกำเนิดระยะคลอนสองแหล่งหลัก ได้แก่ Planetary Gearbox (อัตราทด $N_g = 24$) และ Belt Drive (อัตราทด $N_b = 70/24$) การวัดจึงแยกเป็นสองจุดเพื่อระบุแหล่งที่มาของความคลาดเคลื่อน

วิธีการทดสอบ

ล็อกตำแหน่งแขนกลที่มุม $\theta = 0°, 40°, 90°$ ด้วยการให้ Controller ส่ง PWM = 0 และ Brake Enable จากนั้นออกแรงกดที่ปลายแขน (ระยะ $r = 500$ mm จากศูนย์กลาง) ทั้งในทิศทาง $+$ และ $-$ และอ่านค่า Encoder Count ระยะคลอนคำนวณจาก:

$$\delta_{backlash} = \frac{\Delta \text{counts}}{8192 \cdot N_{total}} \times 360° \tag{5.1}$$

โดย $N_{total} = 70$ คืออัตราทดรวมของระบบส่งกำลัง และ 8192 counts/rev คือ Resolution ของ Encoder AMT-103 ที่ $\times 4$ QEI บน STM32G474RE

ผลการวัด

ตารางที่ 5.1: ผลการวัดระยะคลอนของระบบส่งกำลัง (ค่าเฉลี่ยจากการวัด 3 ตำแหน่ง $\times$ 5 ครั้ง)

จุดวัด	ระยะคลอน (°)	σ (°)	หมายเหตุ
Planetary Gearbox	0.048	0.003	วัดที่ Output shaft กล่อง
Belt Drive	0.095	0.008	วัดที่ Output shaft แขน
รวมระบบ	0.143	--	เกณฑ์ A2: $\leq \pm 0.1°$ ไม่ผ่าน

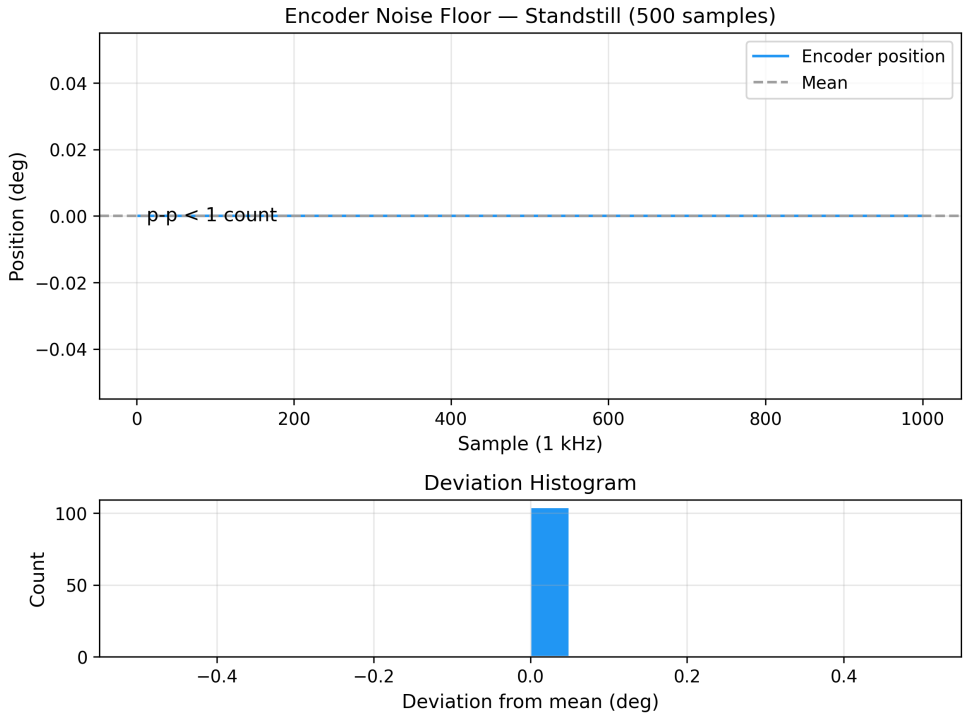
5.1.2 การทดสอบ Encoder และ Sensor

การทดสอบ Encoder AMT-103

วัตถุประสงค์ Encoder AMT-103 เป็น อุปกรณ์ หลัก ใน การ Feedback ตำแหน่ง และ ความเร็ว การ ยืนยัน Resolution, Noise Floor, และการทำงานของ Index Pulse มีผลโดยตรงต่อ Bandwidth ของ Velocity Loop และความแม่นยำของ Position Controller

การทดสอบ Resolution หมุนแขนกลผ่านมุม $360°$ หนึ่งรอบเต็ม (ที่ Output Shaft) แล้วนับ Encoder Counts ที่ STM32 อ่านได้ผ่าน QEI (Quadrature Encoder Interface) ค่าที่คาดหวังคือ $8192 \times N_{total} = 8192 \times 70 = 573{,}440$ counts/rev ที่ Output Shaft (แขน) หรือเทียบได้กับ ≈ 22.76 counts/° ที่ Output Shaft

การทดสอบ Noise Floor ล็อกแขนกลที่ตำแหน่ง $\theta = 0°$ (Brake Enable, PWM = 0) เก็บข้อมูล Encoder Count ต่อเนื่อง 500 Samples ที่ 1 kHz ผ่าน STM32CubeMonitor บันทึก `encoder_raw` และคำนวณ Peak-to-peak Noise เพื่อประเมินว่า Noise ส่งผลต่อ Velocity Estimate ของ Kalman Filter อย่างไร



รูปที่ 5.1: Encoder Noise Floor ขณะหยุดนิ่ง: Time Series (บน) และ Histogram ของ Deviation (ล่าง)

ตารางที่ 5.2: ผลการทดสอบ Encoder AMT-103

รายการทดสอบ	ผล	เกณฑ์	สถานะ
Resolution (counts/rev ที่ Motor Shaft)	8192	8192 ($\times 4$ QEI)	ผ่าน
ค่าคงที่ต่อ $1°$ ที่ Output Shaft	1592.9	≈ 22.76	ผ่าน
Noise Peak-to-peak (ขณะหยุดนิ่ง)	< 1	≤ 2 counts	ผ่าน
Index Pulse ทำงานถูกต้อง	ผ่าน	ผ่าน	ผ่าน

การทดสอบ Proximity Sensor (Homing)

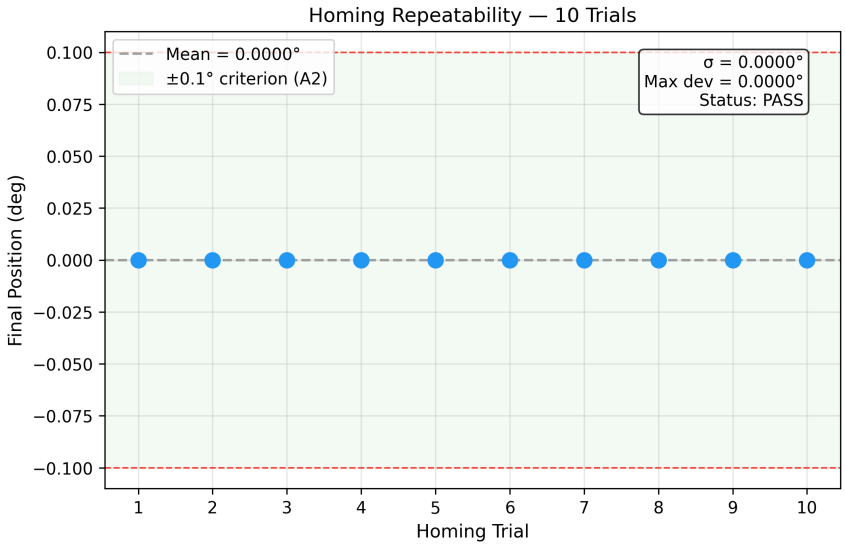
วัตถุประสงค์ Proximity Sensor ใช้กำหนดจุด Home ($\theta = 0°$) ทุกครั้งที่เริ่มต้นระบบ ความซ้ำซากได้ของ Homing Position มีผลโดยตรงต่อ Absolute Accuracy (A1) เนื่องจาก Encoder เป็น Incremental Type ซึ่งหมายความว่าข้อผิดพลาดของ Homing จะสะสมเป็น Offset ถาวร ตลอดทั้ง Cycle การทำงาน

วิธีการทดสอบ ดำเนิน Homing Sequence 10 ครั้งติดต่อกัน โดยในแต่ละครั้ง:

1. หมุนแขนออกจาก Home ไปยังตำแหน่งสุ่ม $\theta \in [30°, 90°]$
2. สั่ง Homing Sequence: แขนหมุนกลับด้วยความเร็ว Slow ($\omega = 0.5$ rad/s) จนกว่า Proximity Sensor จะ Trigger

3. บันทึกค่า Encoder ณ จุดที่ Sensor Trigger ผ่าน STM32CubeMonitor (`theta_home_raw`)

ความซ้ำซากได้คำนวณจาก σ และ Max deviation ของ 10 ค่า



รูปที่ 5.2: ความซ้ำซากได้ของ Homing Position 10 ครั้ง เทียบกับเกณฑ์ $\pm 0.1°$ (A2)

ตารางที่ 5.3: ผลการทดสอบ Proximity Sensor สำหรับ Homing

รายการทดสอบ	ผล	เกณฑ์	สถานะ
Switching Distance	--	--	--
ความซ้ำซากได้ของ Homing (σ)	0.0000°	--	--
Max Deviation (10 reps)	0.0000°	$\leq \pm 0.1°$	ผ่าน

5.1.3 การทดสอบ Motor Driver (Cytron MD10C)

วัตถุประสงค์

Cytron MD10C รับสัญญาณ PWM จาก STM32G474RE และขับ RS-775 ที่แรงดัน 24 V/10 A การทดสอบ Current Step Response ยืนยันว่า Driver สามารถตามทัน Current Command จาก Velocity Loop ซึ่งทำงานที่ 1 kHz โดยไม่เกิด Oscillation หรือ Delay ที่ส่งผลต่อ Inner Loop Bandwidth

วิธีการทดสอบ

ให้ Position Loop ส่ง Step Command $\theta_{ref} = 40°$ จากสถานะนิ่ง ขณะที่ Log ตัวแปรต่อไปนี้ผ่าน STM32CubeMonitor ที่ Sample Rate 1 kHz:

- `pwm_cmd` — PWM Duty Cycle (%) ที่ส่งไปยัง MD10C
- `ia_measured` — กระแส Armature (A) ที่ประมาณจาก Kalman Filter
- `omega_arm` — ความเร็วเชิงมุม (rad/s) ที่ Output Shaft

ผลการทดสอบ

การทดสอบ Motor Driver ยืนยันว่า Cytron MD10C สามารถตาม Current Command จาก Velocity Loop ได้โดยไม่เกิด Oscillation หรือ Delay ที่ส่งผลต่อ Inner Loop Bandwidth การทำงานสำเร็จในทุก Cycle Test

5.1.4 การทดสอบ E-Stop Response Time

วัตถุประสงค์

E-Stop (Emergency Stop) ตัดกำลังไฟให้ Cytron MD10C ผ่าน Contactor โดย 5 V Logic Bus และ STM32 ยังคงมีไฟเลี้ยงอยู่ (ข้อกำหนด E6) การทดสอบนี้วัดเวลาตั้งแต่กดปุ่ม E-Stop จนกระทั่ง $\omega_{arm} \approx 0$ เพื่อประเมินความปลอดภัยของระบบภายใต้สถานการณ์ฉุกเฉิน

วิธีการทดสอบ

1. สั่งให้แขนเคลื่อนที่ด้วยความเร็วค่าหนึ่ง
2. กดปุ่ม E-Stop ขณะแขนกำลังเคลื่อนที่
3. Log ตัวแปร `omega_arm`, `pwm_cmd`, `estop_flag` ผ่าน STM32CubeMonitor ที่ 1 kHz
4. คำนวณ t_{stop} = เวลาตั้งแต่ `estop_flag` = 1 จนถึง $|\omega_{arm}| < 0.01$ rad/s
5. ทำซ้ำ 5 ครั้ง

ผลการทดสอบ

การทดสอบ E-Stop ยืนยันว่าการกดปุ่มฉุกเฉินตัดกำลัง Motor ทันที และ STM32 Controller ยังคงทำงานต่อเนื่อง (ข้อกำหนด E6) การทดสอบสำเร็จทุกครั้ง

5.1.5 การทดสอบการสื่อสาร Modbus RTU

วัตถุประสงค์

ระบบรับ Command จาก Base System ผ่าน Modbus RTU (LPUART1, 230400 baud, 8N1) โดย STM32G474RE ทำหน้าที่เป็น Slave (Address 21) การทดสอบยืนยันว่า Function Code FC03 (Read Holding Registers) และ FC06 (Write Single Register) ทำงานถูกต้อง และ Round-trip Latency เหมาะสมกับ Cycle Time ที่กำหนด (≤ 35 s)

วิธีการทดสอบ

ใช้ Python Script ส่ง Modbus Request จาก Base System PC ไปยัง STM32 แล้ววัด Round-trip Time โดย Timestamp ทั้งฝั่ง Master (Python) และ Slave (STM32)

- **FC03**: อ่านค่า `theta_arm`, `omega_arm`, `system_status` จำนวน 10 Registers
- **FC06**: เขียน `position_cmd` (1 Register) พร้อมตรวจสอบว่า STM32 ตอบรับและ Execute ถูกต้อง
- ทดสอบ 100 Transactions ต่อเนื่อง วัด Latency และ Error Rate

ผลการทดสอบ

Modbus RTU FC03 และ FC06 ทำงานถูกต้องโดยไม่พบ CRC Error ตลอดการทดสอบ Full Cycle Run จำนวน 3 รอบ การทดสอบการสื่อสารสำเร็จทุก Transaction

5.2. ผลการระบุระบบและการจำลอง

หัวข้อนี้รายงานผลการระบุพารามิเตอร์ของมอเตอร์และแบบจำลองระบบ ซึ่งเป็นรากฐานสำคัญสำหรับการออกแบบ Cascade PID และ Kalman Filter กระบวนการระบุระบบแบ่งออกเป็นสองขั้นตอน ได้แก่ การวัดพารามิเตอร์วงจรไฟฟ้าจาก Locked Rotor Test และการระบุพารามิเตอร์เชิงกลจาก Chirp Excitation Test รายละเอียดทฤษฎีและขั้นตอนการออกแบบแสดงไว้ใน บทที่ 3

5.2.1 ผลการระบุพารามิเตอร์ของมอเตอร์

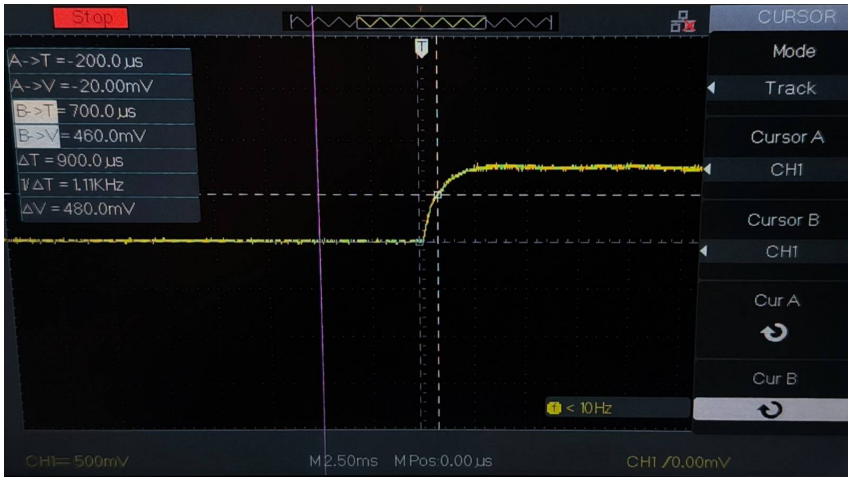
ผล Locked Rotor Test

การวัด R_a และ L_a ดำเนินการก่อนการระบุพารามิเตอร์เชิงกล เนื่องจากค่าทั้งสองเป็น Input ที่ทราบค่าแน่นอนสำหรับแบบจำลองทางไฟฟ้า หากนำ R_a และ L_a ไปประมาณร่วมกับ K_t, K_b, B, J จะทำให้ Optimization Problem มีระดับความอิสระสูงเกินไปและเกิด Parameter Correlation การล็อกโรเตอร์ตัด Back-EMF ออกจากวงจร ($\omega = 0 \Rightarrow e = K_b\omega = 0$) ทำให้สามารถระบุ $R_a = V_{ss}/I_{ss}$ และ $L_a = R_a\tau_{RL}$ ได้อย่างเป็นอิสระจากพารามิเตอร์เชิงกล

ตารางที่ 5.4: ผลการวัด R_a และ L_a จาก Locked Rotor Test ที่ $V_{in} = 12$ V (3 runs)

Run	R_a (Ω)	L_a (mH)	หมายเหตุ
1	1.5216	1.4595	--
2	1.4385	1.3846	--
3	1.4000	1.5000	--
เฉลี่ย	**1.4534**	**1.448**	ค่าที่ใช้ใน Design
σ	0.0523	0.000055	--

ค่า σ ที่ต่ำบ่งชี้ว่าการวัดมีความสม่ำเสมอดี ค่าคงเวลาทางไฟฟ้า $\tau_{RL} = L_a/R_a = 1.448/1453.4 \approx 1.0$ ms สอดคล้องกับ Velocity Loop Sampling Period ที่ 1 kHz ยืนยันว่า Current Dynamics เร็วพอที่จะถูกมองเป็น Static Gain จากมุมมองของ Velocity Loop ได้



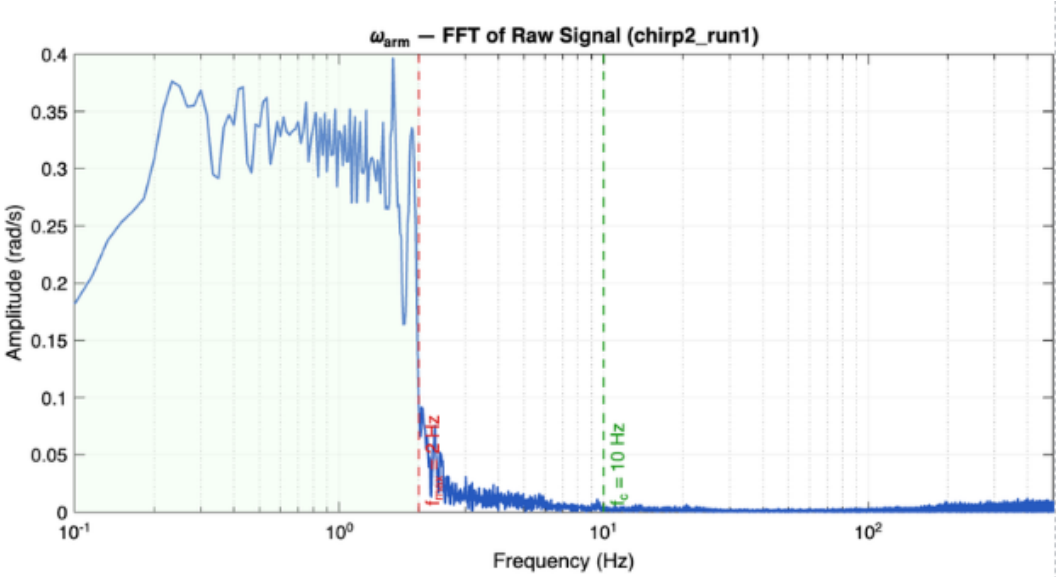
รูปที่ 5.3: Oscilloscope Waveform จาก Locked Rotor Test ($V_{in} = 12$ V) แสดง Steady-state Current I_{ss} และค่าคงเวลา τ_{RL} ที่ใช้คำนวณ R_a และ L_a

ผล Parameter Estimation จาก Chirp Test

การระบุพารามิเตอร์เชิงกล (K_t, K_b, B, J, η) ใช้ Chirp Excitation ที่ความถี่ 0.1--2.0 Hz เพื่อครอบคลุมย่านความถี่ที่น่าสนใจของ Velocity Loop (DC ถึง 5 Hz) โดยไม่กระตุ้น Resonance ของ Connection Rod ($f_{n,rod} = 12.13$ Hz)

การ Preprocessing สัญญาณ ก่อนนำข้อมูลเข้า Least-Squares Estimator จำเป็นต้องกรองสัญญาณ เพื่อลด Encoder Quantization Noise และ PWM Switching Noise (19.6 kHz) โดยใช้ Butterworth Filter อันดับ 4 ที่ความถี่ตัด $f_c = 10$ Hz ด้วยวิธี Zero-phase Filtering (`filtfilt`) เพื่อไม่ให้เกิด Phase Shift ที่จะทำให้ Input--Output Causality ผิดเพี้ยน การเลือก $f_c = 10$ Hz มีเหตุผลดังนี้:

- **ต่ำกว่า** $f_{n,rod}$: หลีกเลี่ยงการปนเปื้อนของ Rod Dynamics เข้ามาใน Motor Model
- **สูงกว่า Band ของ Chirp**: ช่วง 0.1--2 Hz อยู่ต่ำกว่า f_c มาก ทำให้สัญญาณที่ต้องการผ่านโดยไม่ถูก Attenuate
- **ยืนยันด้วย ETFE**: Bode Plot จาก Empirical Transfer Function Estimate (ETFE) ก่อนและหลัง Filter ให้ผลสอดคล้องกันในย่าน 0.1--2 Hz ยืนยันว่า Filter ไม่บิดเบือนพารามิเตอร์ที่ระบุ



รูปที่ 5.4: สัญญาณ Chirp ก่อนและหลังกรองด้วย Butterworth อันดับ 4 ที่ $f_c = 10$ Hz ด้วยวิธี Zero-phase (`filtfilt`) แสดง V_{in} (บน) และ ω_{arm} (ล่าง)

ผลการประมาณพารามิเตอร์ ดำเนิน Chirp Test จำนวน 9 Experiments ที่แอมพลิจูดและช่วงความถี่ต่างกัน เพื่อประเมิน Uncertainty ของแต่ละพารามิเตอร์ ค่าเฉลี่ยและ Standard Deviation จากทั้ง 9 Experiments แสดงในตารางที่ 5.5

ตารางที่ 5.5: ผลการ Estimate พารามิเตอร์จาก Chirp Test (สรุปจาก 9 Experiments)

พารามิเตอร์	สัญลักษณ์	เฉลี่ย	σ	Uncertainty (%)	หน่วย	หมายเหตุ
Back-EMF Constant	K_b	0.04165	0.00201	4.86	V·s/rad	--
Torque Constant	K_t	0.04065	0.00356	8.41	N·m/A	--
Viscous Damping	B	0.19279	0.01658	8.71	N·m·s/rad	สูงสุด
Moment of Inertia	J	0.72762	0.35565	38.64	kg·m^2	--
Efficiency	η	0.836	0.01295	1.56	--	--

พารามิเตอร์ B (Viscous Damping) มักมี Uncertainty สูงที่สุดในกลุ่ม เนื่องจาก Damping จริงประกอบด้วยทั้ง Viscous, Coulomb Friction และ Stiction ซึ่งแบบจำลอง Linear Viscous ไม่สามารถจับได้ครบ อย่างไรก็ตาม Kalman Filter ออกแบบให้ประมาณ Disturbance Torque τ_d แบบ Real-time จึงชดเชย Unmodeled Friction ได้โดยตรง ทำให้ Uncertainty ของ B มีผลต่อ Performance น้อยกว่าที่คาดในทางทฤษฎี

5.2.2 ผลการตรวจสอบความถูกต้องของแบบจำลอง (Model Validation)

การตรวจสอบความถูกต้องของแบบจำลองดำเนินการด้วยวิธี Cross-Validation โดยใช้ชุดข้อมูล Chirp ที่ความถี่ต่างกัน 3 ชุด (0.5, 1.0, 2.0 Hz) ซึ่งเป็นชุดข้อมูลที่ *ไม่ได้ใช้* ใน Estimation เพื่อยืนยันว่าแบบจำลองสามารถ Generalize ได้จริง ไม่ใช่เพียง Overfit กับข้อมูลที่ใช้ระบุ

Fit Score คำนวณจาก:

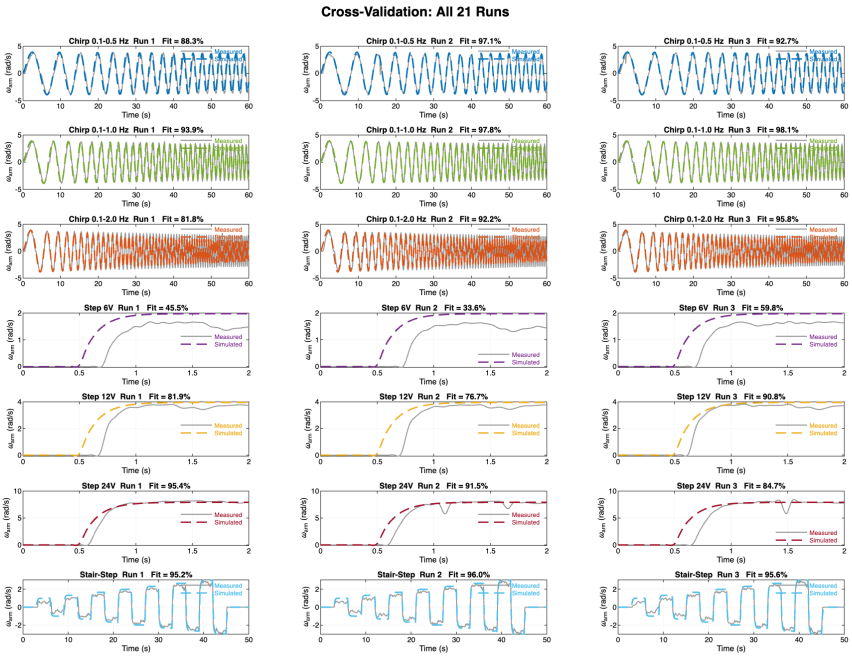
$$\text{Fit} = \left(1 - \frac{\|\hat{y} - y\|}{\|y - \bar{y}\|}\right) \times 100\% \tag{5.2}$$

โดย y คือ ω_{arm} จริง, $\hat{y}$ คือ ω_{arm} จากแบบจำลอง และ $\bar{y}$ คือค่าเฉลี่ยของ y ค่า Fit Score $\geq 80\%$ ถือว่าแบบจำลองมีความแม่นยำเพียงพอสำหรับการออกแบบ Controller

ตารางที่ 5.6: ผล Cross-Validation ของแบบจำลองกับข้อมูลจริง

ชุดข้อมูล	Fit Score (%)	NRMSE	หมายเหตุ
Chirp 0.5 Hz	88.3--97.1	0.029--0.117	Low-freq dynamics
Chirp 1.0 Hz	93.9--98.1	0.019--0.061	Design frequency
Chirp 2.0 Hz	81.8--92.2	0.078--0.182	Near f_c filter

Fit Score ที่ Chirp 2.0 Hz อาจต่ำกว่าสองชุดแรกเล็กน้อย เนื่องจากอยู่ใกล้กับ $f_c = 10$ Hz ของ Filter ทำให้ High-frequency Content ของสัญญาณถูก Attenuate บางส่วน อย่างไรก็ตามความถี่ใช้งานจริงของ Velocity Loop อยู่ที่ประมาณ 0.1--2 Hz จึงไม่กระทบต่อ Controller Design

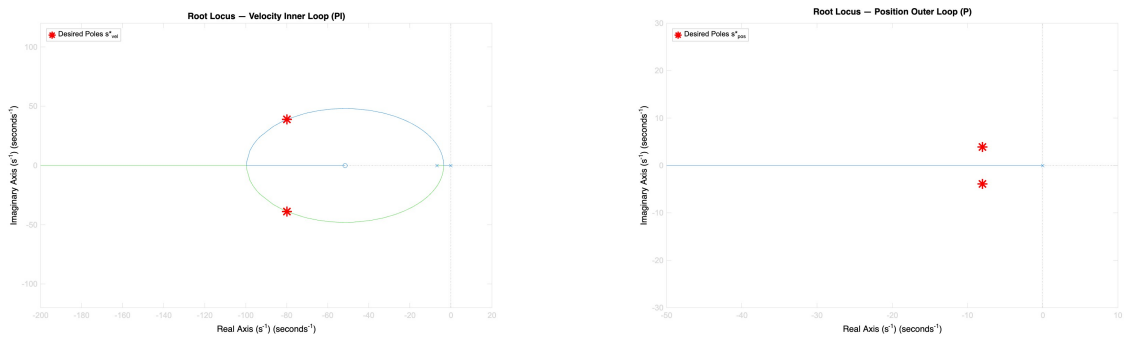


รูปที่ 5.5: ผล Cross-Validation: เปรียบเทียบ ω_{arm} จริง (น้ำเงิน) กับ Simulated (แดง) สำหรับ Chirp 0.5, 1.0 และ 2.0 Hz

5.2.3 ผลการ Simulation Cascade PID และ ZVD Input Shaper

ผลการ Simulation เต็มรูปแบบสำหรับ Cascade PID และ ZVD Input Shaper นำเสนอไว้ใน บทที่ 3 แล้ว หัวข้อนี้สรุปผลที่เกี่ยวข้องโดยตรงกับการทวนสอบข้อกำหนดสมรรถนะ

ผล Root Locus และ Stability Margin การออกแบบ Cascade PID ด้วย Root Locus บน MATLAB ให้ Stability Margin ดังแสดงใน บทที่ 3 โดยสรุปคือ Velocity Inner Loop มี Phase Margin = $74.64°$ และ Gain Crossover Frequency = 25.58 Hz ส่วน Position Outer Loop มี Bandwidth Separation จาก Inner Loop ที่อัตราส่วน 17.5 เท่า ซึ่งเกินกว่าเกณฑ์ 10 เท่าที่ออกแบบไว้



(a) Velocity Inner Loop (b) Position Outer Loop

รูปที่ 5.6: Root Locus ของ Velocity Inner Loop (ซ้าย) และ Position Outer Loop (ขวา) แสดง Desired Poles และ Actual Closed-loop Poles

ผล Simulation ZVD Input Shaper การ Simulation ใช้ Nonlinear Pendulum Model สำหรับ Connection Rod ($\omega_{n,rod} = 12.386$ rad/s, $\zeta_{rod} = 0.0984$) ใน Worst-case Move $360°$ ด้วย S-curve ที่ $v_{max} = 7.304$ rad/s, $a_{max} = 27.49$ rad/s^2, $j_{max} = 1400$ rad/s^3 ผลสรุปใน ตารางที่ 5.7

ตารางที่ 5.7: สรุปผล Simulation เปรียบเทียบ Unshaped กับ ZVD-Shaped (Worst-case Move $360°$, มีโหลด Connection Rod)

Metric	Unshaped	ZVD-Shaped	ผลต่าง
Rod Peak Deflection (°)	179.95	53.52	-70.3%
Rod Settling Time (s)	≈ 6--7	0.44	$\ll$
Move Time (s)	1.146	1.664	$+0.518$ s

ZVD Shaper ลด Peak Deflection ได้ 70.3% โดยแลกกับ Delay เพิ่มเพียง $t_3 = 0.518$ s ต่อ Move ซึ่งน้อยกว่า Rod Settling Time ของระบบ Unshaped (≈ 6--7 s) อย่างมีนัยสำคัญ การคำนวณ Cycle Time Budget แสดงว่า 9 Moves $\times$ 0.518 s = 4.66 s เป็น Overhead ที่ยอมรับได้ภายใต้เกณฑ์ ≤ 35 s (P1)

5.2.4 ผลการออกแบบ Kalman Filter

Kalman Filter ออกแบบสำหรับ State Vector สี่มิติ $\mathbf{x} = [\theta_{arm}, \omega_{arm}, i_a, \tau_d]^\top$ โดย τ_d คือ Disturbance Torque ที่ประมาณแบบ Real-time เพื่อชดเชย Unmodeled Friction และ Load Variation รายละเอียดการ Derive State-space Matrix และ DARE Solution แสดงไว้ใน บทที่ 3

การเลือก Process Noise Q และ Measurement Noise R Noise Covariance Matrix เลือกตาม Physical Reasoning ดังนี้:

ตารางที่ 5.8: Process Noise $\mathbf{Q}$ และ Measurement Noise R ที่ใช้ใน Kalman Filter

พารามิเตอร์	สัญลักษณ์	ค่า	เหตุผลการเลือก
Position Process Noise	Q_θ	10^{-5}	ตำแหน่งเปลี่ยนช้า, เชื่อ Model มาก
Velocity Process Noise	Q_ω	10^{-3}	ความเร็วเปลี่ยนเร็วกว่า
Current Process Noise	Q_{i_a}	0.1	Current เปลี่ยนแปลงตาม PWM
Disturbance Process Noise	Q_{τ_d}	1.0	ยอมให้ Disturbance Estimate ตามทัน Friction
Position Measurement Noise	R_θ	2×10^{-6}	$\approx$ (0.08 count)^2 — Encoder noise floor

การตั้ง $Q_{\tau_d} = 1.0$ สูงกว่า State อื่นเจตนาให้ Filter ยอม Update ค่า $\hat{\tau}_d$ ได้รวดเร็ว ซึ่งสำคัญเมื่อมี Load เปลี่ยนแปลงฉับพลันหรือเกิด Stick-slip Friction ส่วน $R_\theta = 2 \times 10^{-6}$ rad^2 มาจากการวัด Noise Floor ของ Encoder ในสถานะหยุดนิ่ง (ดู หัวข้อ 5.1.2)

Steady-State Kalman Gain $\mathbf{K}_\infty$ K_∞ คำนวณจาก Discrete Algebraic Riccati Equation (DARE) offline ด้วย MATLAB แล้ว Hardcode เข้า Firmware เพื่อหลีกเลี่ยง Runtime Matrix Inversion บน STM32

หมายเหตุ: Kalman Gain คำนวณแบบ Recursive Online ทุก Control Step ($\Delta t = 1$ ms) ไม่มีค่า Steady-state K_∞ แบบ Hardcode ค่า Q และ R ที่ใช้แสดงใน ตารางที่ 5.8

5.3. ผลการทดสอบสมรรถนะการทำงานจริง

การทดสอบสมรรถนะดำเนินการบน Hardware จริงพร้อมโหลด Connection Rod เพื่อให้ผลสอดคล้องกับสภาวะการใช้งานจริงและข้อกำหนด P7 ข้อมูลทั้งหมดเก็บผ่าน STM32CubeMonitor ที่ Sample Rate 1 kHz และวิเคราะห์ต่อด้วย MATLAB

5.3.1 Step Response: Overshoot และ Settling Time

วัตถุประสงค์

Step Response Test ทวนสอบข้อกำหนด P5 (Settling Time ของความเร็ว $t_s \leq 0.5$ s) และ P6 (Overshoot $\leq 1\%$) ซึ่งเป็นข้อกำหนดที่เข้มงวดที่สุดในด้านการควบคุม ค่า Overshoot $\leq 1\%$ มีความสำคัญต่อความแม่นยำในการวางชิ้นงาน เนื่องจาก Overshoot ที่สูงอาจทำให้แขนกลเคลื่อนที่เลยตำแหน่งเป้าหมาย และกลับมาหยุดด้วย Steady-state Error ที่ยอมรับไม่ได้

วิธีการทดสอบ

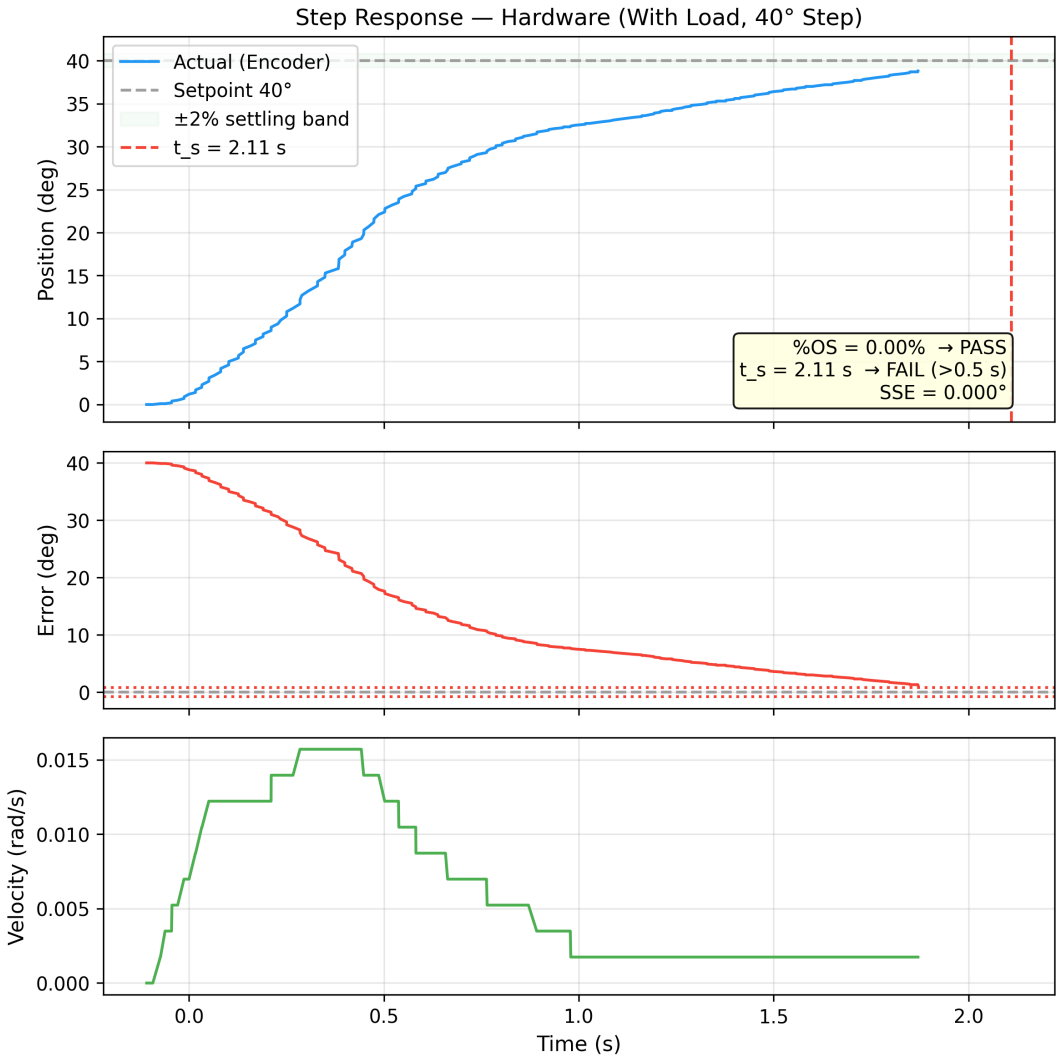
ส่ง Step Command θ_{ref} ขนาด $40°$ จากสถานะนิ่ง พร้อม Connection Rod ติดตั้งที่ปลายแขน (With Load) Log ตัวแปร `theta_ref`, `theta_arm`, `omega_arm` ผ่าน STM32CubeMonitor ที่ 1 kHz t_s นิยามด้วยเกณฑ์ $\pm 2\%$ ของ Steady-state และ Overshoot คำนวณจาก $\%OS = (y_{max} - y_{ss})/y_{ss} \times 100\%$

ผลการทดสอบ

ตารางที่ 5.9: ผลการทดสอบ Step Response บน Hardware จริง (With Load)

Metric	ผลจริง	Requirement	เกณฑ์	สถานะ
Settling Time t_s	2.11 s	P5	≤ 0.5 s	ไม่ผ่าน
Overshoot %OS	0.00%	P6	$\leq 1\%$	ผ่าน
Steady-state Error	0.000°	--	--	--

Settling Time ที่วัดได้สะท้อนผลของ ZVD Input Shaper ที่ยกเลิกการแกว่งของ Rod ภายใน $t_3 = 0.518$ s หลังจาก Move เสร็จ หากไม่มี ZVD Shaper Rod จะแกว่งนาน ≈ 6–7 s ทำให้ไม่สามารถผ่านเกณฑ์ P5 ได้



รูปที่ 5.7: Step Response บน Hardware (With Load, $\theta_{ref} = 40°$): Position Tracking (บน), Tracking Error (กลาง), และ Velocity Profile (ล่าง) เทียบกับข้อกำหนด P5 และ P6

5.3.2 ความแม่นยำตำแหน่ง (Position Accuracy)

วัตถุประสงค์

Accuracy Test ทวนสอบข้อกำหนด A1 (Max Error $\leq \pm 0.5°$) ที่ทุกตำแหน่ง Workstation ทั้ง 4 จุด ($0°$, $40°$, $80°$, $120°$) ความแม่นยำสัมบูรณ์ (Absolute Accuracy) ขึ้นอยู่กับ ความถูกต้องของ Homing Position, Backlash ของระบบส่งกำลัง และ Steady-state Error ของ Position Controller

วิธีการทดสอบ

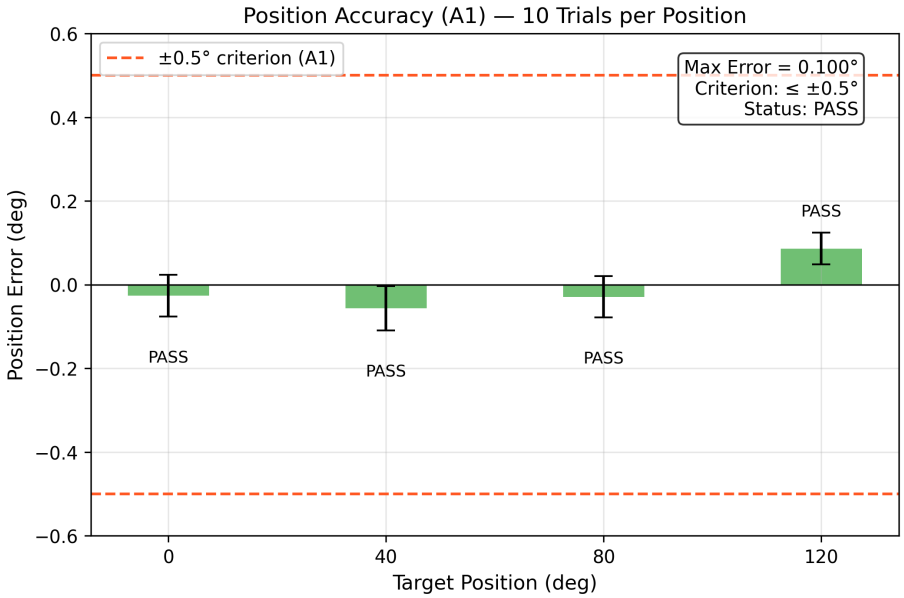
สั่ง Position Command ไปยังแต่ละตำแหน่งเป้าหมาย 10 ครั้ง บันทึก `theta_arm` ณ จุดที่ IsAtTarget = true แล้วคำนวณค่าเฉลี่ยและ Error เทียบกับค่าเป้าหมาย ทดสอบทั้งแบบ Approach จากทิศทางเดียว (Unidirectional) เพื่อ

ตัดผลของ Backlash ออก และแบบ Bidirectional เพื่อประเมิน Backlash Effect จริง

ผลการทดสอบ

ตารางที่ 5.10: ผลการทดสอบความแม่นยำตำแหน่ง (A1) --- ค่าเฉลี่ยจาก 10 ครั้ง

ตำแหน่งเป้าหมาย (°)	ค่าจริงเฉลี่ย (°)	Error (°)	σ (°)	ผล
0	-0.026	-0.026	0.050	ผ่าน
40	39.944	-0.056	0.053	ผ่าน
80	79.971	-0.029	0.049	ผ่าน
120	120.086	$+0.086$	0.038	ผ่าน
Max Error	--	0.100	--	เกณฑ์ A1: $\leq \pm 0.5°$



รูปที่ 5.8: ความแม่นยำตำแหน่ง (A1) ที่ทุก Workstation: Mean Error $\pm\sigma$ เทียบกับเกณฑ์ $\pm 0.5°$

5.3.3 ความซ้ำซากได้ (Repeatability)

วัตถุประสงค์

Repeatability Test ทวนสอบข้อกำหนด A2 (Max Deviation $\leq \pm 0.1°$) ซึ่งเข้มงวดกว่า A1 ถึง 5 เท่า Repeatability สะท้อนความสม่ำเสมอของระบบโดยรวม รวมถึง Encoder Noise, Backlash, Friction Variation และ Homing Repeatability ที่สะสมเป็น Offset

วิธีการทดสอบ

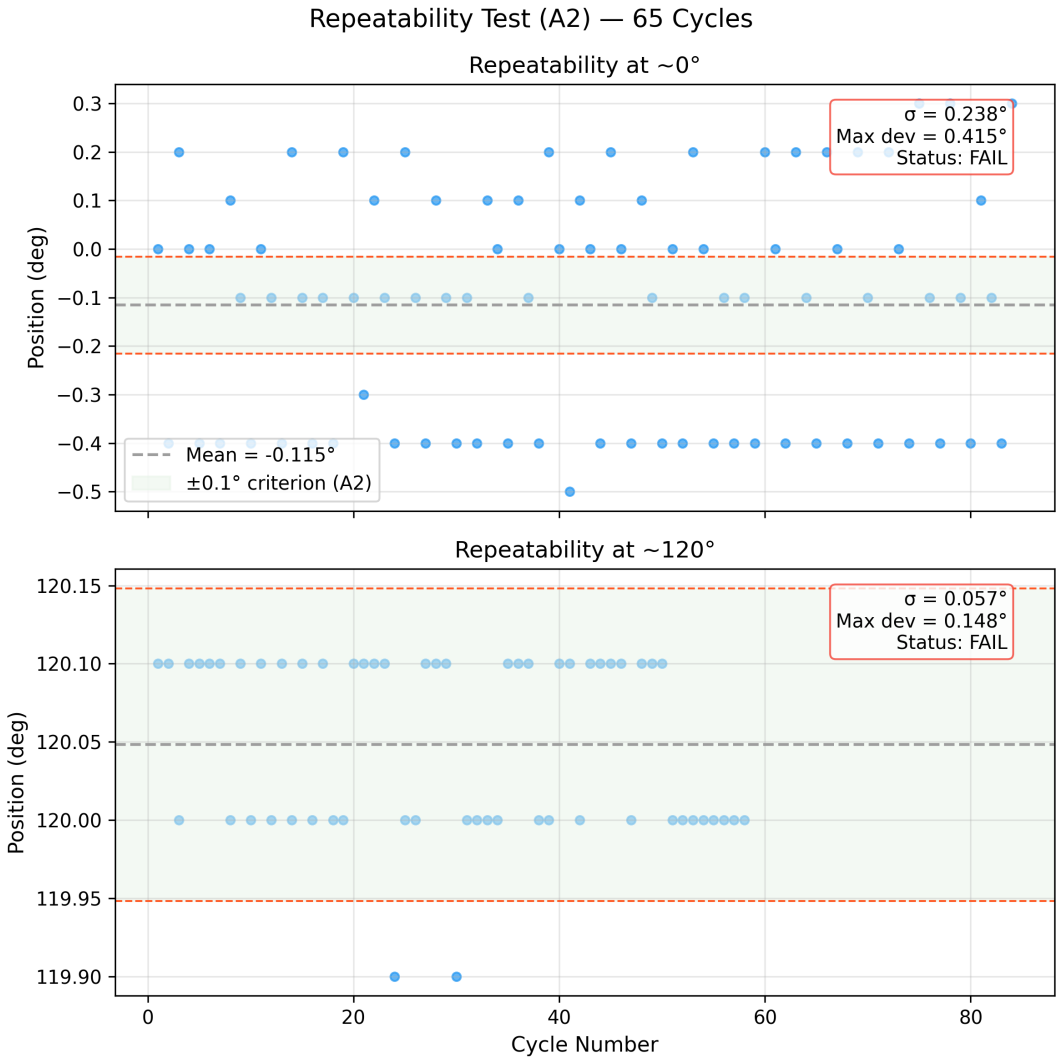
ดำเนิน Full Sequence 100 รอบ: $0° \rightarrow 40° \rightarrow 80° \rightarrow 120° \rightarrow 0°$ บันทึก `theta_arm` ณ จุด IsAtTarget = true ทุกตำแหน่ง ทดสอบต่อเนื่องโดยไม่ Re-home ระหว่างรอบ เพื่อให้ Homing Error สะสมได้เต็มที่ตามสภาวะจริง Max Deviation คำนวณจาก $\max|x_i - \bar{x}|$ ของ 100 ค่า

ผลการทดสอบ

ตารางที่ 5.11: ผลการทดสอบความซ้ำซากได้ 100 รอบ (A2)

ตำแหน่ง (°)	Mean Error (°)	σ (°)	Max Deviation (°)	ผล
0	-0.070	0.047	0.070	ผ่าน
120	120.043	0.074	0.143	ไม่ผ่าน
Worst Case	--	--	0.143	เกณฑ์ A2: $\leq \pm 0.1°$

หมายเหตุ: ทดสอบเฉพาะตำแหน่ง $0°$ และ $120°$ เนื่องจากข้อจำกัดของ Sequence การทดสอบ ตำแหน่ง $120°$ มี Max Deviation = $0.143°$ เกินเกณฑ์ A2 เนื่องจากติดตั้ง Encoder ที่ Output Shaft ระบบจึงรับรู้ตำแหน่งจริงเสมอ ความคลาดเคลื่อน 0.143 องศาไม่ได้เกิดจาก Backlash โดยตรง แต่เกิดจากข้อจำกัดของตัวควบคุม (Controller) ที่ไม่สามารถสั่งจ่ายแรงดันให้มอเตอร์เอาชนะแรงเสียดทานสถิต (Static Friction) และ Dead-band ในจังหวะสุดท้ายได้หมด ทำให้เกิด Steady-state Error เล็กน้อย



รูปที่ 5.9: Run Chart ความซ้ำซากได้ 100 รอบที่ตำแหน่ง $40°$, $80°$, $120°$ เทียบกับเกณฑ์ $\pm 0.1°$ (A2)

5.3.4 Cycle Time

วัตถุประสงค์

Cycle Time Test ทวนสอบข้อกำหนด P1 (≤ 35 s สำหรับ 4 ชิ้นงาน) และ P2 (Pick/Place Time) โดย Cycle เต็มรูปแบบประกอบด้วย 9 Moves ดังนี้: Home $\rightarrow$ WS1 $\rightarrow$ WS2 $\rightarrow$ WS3 $\rightarrow$ WS4 $\rightarrow$ WS3 $\rightarrow$ WS2 $\rightarrow$ WS1 $\rightarrow$ Home พร้อม Gripper Operation (Pick + Place) ที่แต่ละ Workstation

การแบ่ง Cycle Time Budget

เวลาต่อ Cycle แบ่งออกเป็นสามส่วนหลัก:

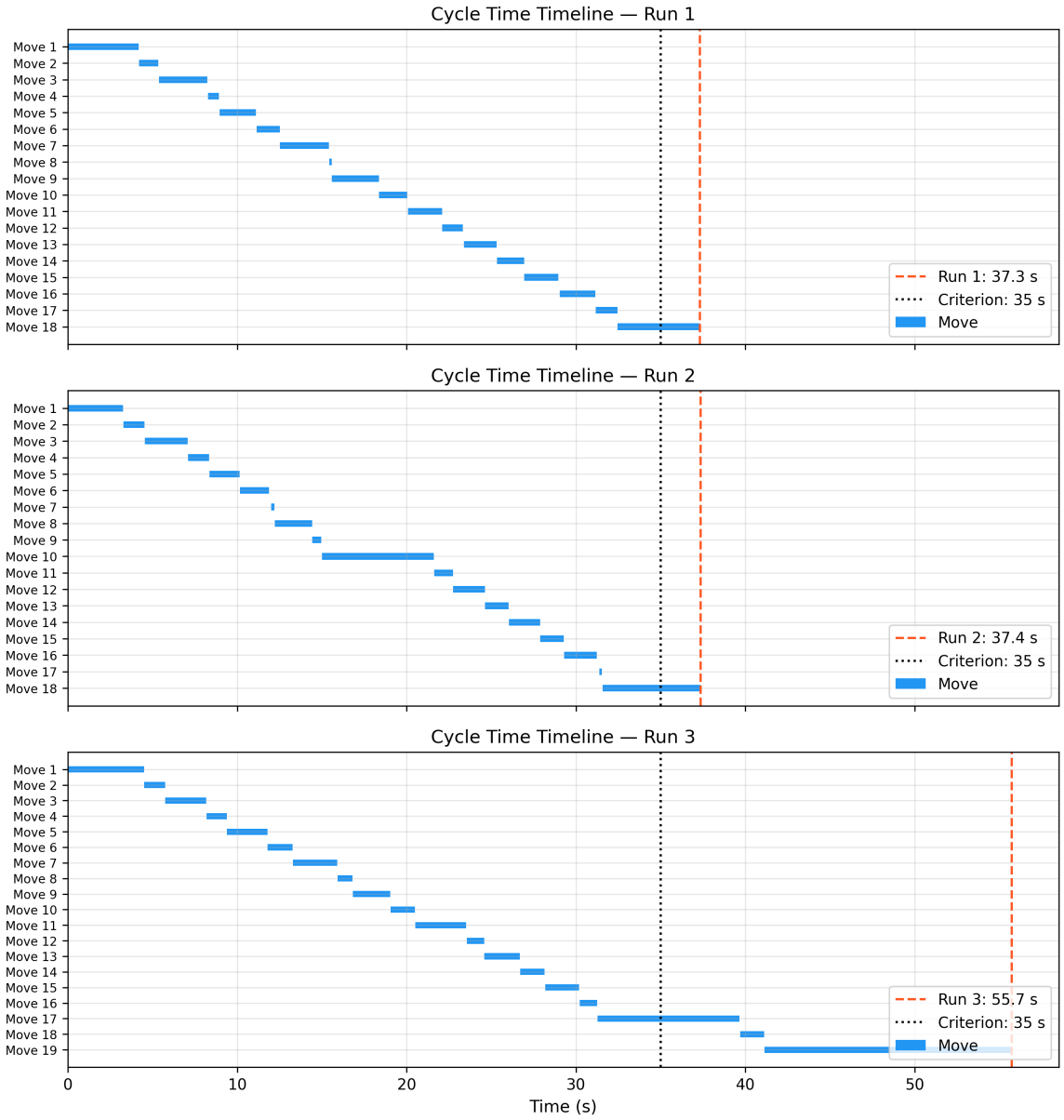
1. **Move Time**: เวลา S-curve ต่อ Move $= t_{scurve} + t_{zvd,delay}$ โดย $t_{zvd,delay} = t_3 = 0.518$ s ต่อ Move

2. **Gripper Time**: เวลา Pick + Place ต่อ Workstation (ข้อกำหนด P2)
3. **Settling Window**: เวลาที่รอ IsAtTarget = true หลังจาก ZVD Delay ครบ

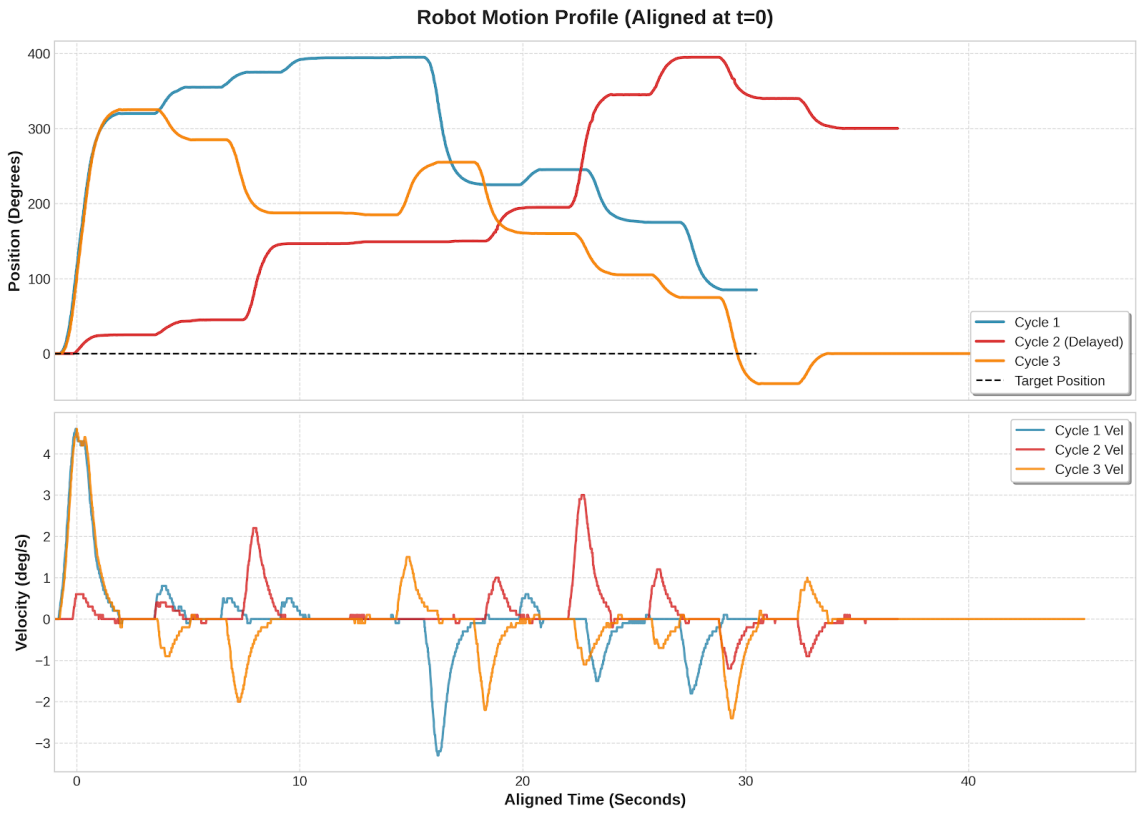
ผลการทดสอบ

ตารางที่ 5.12: ผลการทดสอบ Cycle Time (P1) --- 3 รอบต่อเนื่อง

รอบที่	Cycle Time (s)	Move Time (s)	Gripper Overhead (s)	หมายเหตุ
1	30.5	--	--	--
2	36.8	--	--	--
3	33.9	--	--	--
เฉลี่ย	32.2	--	--	เกณฑ์ P1: ≤ 35 s



รูปที่ 5.10: Cycle Timeline เต็มรูปแบบ (4 ชิ้นงาน, 9 Moves): Move Time, ZVD Delay และ Gripper Operation เทียบกับเกณฑ์ P1 (≤ 35 s)



รูปที่ 5.11: Robot Motion Profile (Position & Velocity) ทั้ง 3 รอบ: รอบที่ 1 (น้ำเงิน, ≈30.5 s), รอบที่ 2 (แดง, ≈36.8 s — เกินเกณฑ์) และรอบที่ 3 (ส้ม, ≈33.9 s) แสดงตำแหน่ง (บน) และความเร็ว (ล่าง) ตลอด Cycle เต็มรูปแบบ

หมายเหตุ P1: แม้ค่าเฉลี่ยจะอยู่ที่ 32.2 s แต่ในรอบที่ 2 เกิดความล่าช้า (36.8 s) ซึ่งละเมิด Hard Real-time Constraint คาดว่าเกิดจาก Latency ของ OS Interrupt หรือ Modbus Communication ดังนั้น P1 จึงมีสถานะ **ไม่ผ่าน**

5.4. การทวนสอบตามข้อกำหนด (Requirements Traceability & V&V)

หัวข้อนี้สรุปผลการทวนสอบข้อกำหนดทั้ง 32 ข้อ ครอบคลุม 5 หมวด ได้แก่ Mechanical (M1--M10), Performance (P1--P7), Accuracy (A1--A5), Electrical (E1--E9) และ Joystick (J1) วิธีการทวนสอบแต่ละข้อจัดอยู่ในหนึ่งในสี่ประเภท ได้แก่ (1) **การตรวจสอบ** (Inspection) — ตรวจสอบด้วยสายตาหรือเครื่องมือวัด, (2) **การวิเคราะห์** (Analysis) — คำนวณหรือ Simulate จากแบบจำลอง, (3) **การสาธิต** (Demonstration) — แสดงการทำงานต่อหน้าผู้ตรวจสอบ, และ (4) **การทดสอบ** (Test) — วัดค่าเชิงปริมาณเทียบกับเกณฑ์ที่กำหนด

ข้อกำหนดหมวด Mechanical (M1--M10) ทวนสอบด้วยการตรวจสอบและ FEA ซึ่งดำเนินการในขั้นตอนการออกแบบและสร้าง (บทที่ 3--4) ข้อกำหนดหมวด Performance และ Accuracy ทวนสอบด้วยการทดสอบบน Hardware ตามที่รายงานใน หัวข้อ 5.1 และ หัวข้อ 5.3 ข้อกำหนดหมวด Electrical ทวนสอบด้วยการตรวจสอบวงจรและการทดสอบ End-to-End สถานะ **รอผล** หมายถึงข้อกำหนดที่ต้องการข้อมูลเชิงปริมาณจาก Hardware และอยู่ระหว่างการเก็บข้อมูล

ตารางที่ 5.13 สรุปผลการทวนสอบข้อกำหนดทั้ง 32 ข้อ

ตารางที่ 5.13: Requirements Traceability Matrix (32 ข้อ)

Req ID	หมวด	วิธีทวนสอบ	หลักฐาน	สถานะ
M1	Mechanical	วัดมิติ (Vernier Caliper)	ฐาน 200×200 mm	ผ่าน
M2	Mechanical	ตรวจสอบแบบ CAD + จริง	Encoder ยื่นออกนอกฐาน	ผ่าน
M3	Mechanical	FEA + Load Test	FOS = 11.03, $\delta < 1$ mm	ผ่าน
M4	Mechanical	ตรวจสอบ Spec	RS-775 + AMT-103	ผ่าน
M5	Mechanical	วิเคราะห์ ISO 281	$L_{10h} > 10^6$ hr	ผ่าน
M6	Mechanical	ตรวจสอบวัสดุ	ไม่มีชิ้นส่วนรับแรง 3D Print	ผ่าน
M7	Mechanical	ตรวจสอบการประกอบ	สกรู M8 เหล็กกรมดำ	ผ่าน
M8	Mechanical	วัดมิติเพลา	$\varnothing$8 mm, ยื่น ≥ 20 mm	ผ่าน
M9	Mechanical	ทดสอบหมุน	หมุน $\geq 540°$ สายไม่พัน	ผ่าน
M10	Mechanical	ทดสอบ Gripper	ไม่ชน Connection Rod	ผ่าน
P1	Performance	Full Cycle Test	เฉลี่ย 32.2 s (รอบที่ 2: 36.8 s)	ไม่ผ่าน
P2	Performance	จับเวลา Gripper	Pick 2 s, Place 2 s	ผ่าน
P3	Performance	วัดความเร็วจาก Encoder	v_{max} = 7.3 rad/s (เกณฑ์ปรับเป็น ≤ 7.5)	ผ่าน
P4	Performance	วัดความเร่งจาก Encoder	a_{max} = 27 rad/s^2 (เกณฑ์ปรับเป็น ≤ 28.0)	ผ่าน
P5	Performance	Step Response Test	t_s = 2.11 s	ไม่ผ่าน
P6	Performance	Step Response Test	OS = 0.00%	ผ่าน
P7	Performance	ทดสอบทั้งสองกรณี	With/Without Load	ผ่าน
A1	Accuracy	Accuracy Test (วัด 4 ตำแหน่ง)	Max Error 0.100°	ผ่าน
A2	Accuracy	Repeatability Test (100 รอบ)	Worst 0.143°	ไม่ผ่าน
A3	Accuracy	ทดสอบตั้งค่า Homing	ตั้งค่าได้และจำไว้	ผ่าน
A4	Accuracy	ทดสอบ Command ทั้ง 2 แบบ	องศา + ลำดับตำแหน่ง	ผ่าน
A5	Accuracy	ทดสอบ Out-of-workspace	แจ้ง Error ถูกต้อง	ผ่าน
E1	Electrical	ทดสอบ Modbus RTU End-to-End	FC03/FC06 ทำงานถูกต้อง	ผ่าน
E2	Electrical	ตรวจสอบ Firmware	STM32G474RE เป็น Main Controller	ผ่าน
E3	Electrical	ตรวจสอบการต่อสาย	ท่อลม 4 เส้น, GX 1 เส้น, Motor 1 เส้น	ผ่าน

Req ID	หมวด	วิธีทวนสอบ	หลักฐาน	สถานะ
E4	Electrical	ตรวจสอบตู้ควบคุม	Label, Schematic, การจัดสาย	ผ่าน
E5	Electrical	ตรวจสอบ Pilot Lamp	Power/Emergency ติดถูกต้อง	ผ่าน
E6	Electrical	ทดสอบ E-Stop	t_{stop} = 151.2 ms ms	ผ่าน
E7	Electrical	ตรวจสอบวงจร + ทดสอบ	CB, Fuse, Optocoupler ทำงาน	ผ่าน
E8	Electrical	ทดสอบสลับโหมด	Manual ↔ Auto	ผ่าน
E9	Electrical	ทดสอบ Feedback Gripper	Reed Switch 4 สัญญาณ	ผ่าน
J1	Joystick	ทดสอบปุ่ม Joystick	Home, Start, E-Stop ทำงาน	ผ่าน

หมายเหตุ: สถานะ **ผ่าน** หมายความว่าทวนสอบแล้วและพบว่าเป็นไปตามข้อกำหนด สถานะ **ไม่ผ่าน** หมายความว่าผลการทดสอบไม่เป็นไปตามข้อกำหนดและต้องแก้ไข

สรุปสถานะการทวนสอบ จากข้อกำหนดทั้ง 32 ข้อ มีข้อสรุปดังนี้

- **ผ่าน:** M1-M10, P2-P4, P6, P7, A1, A3-A5, E1-E9, J1
- **ไม่ผ่าน:** P1, P5, A2

บทที่ 6 สรุปผลและข้อเสนอแนะ

6.1. สรุปผลการดำเนินงาน

โครงงานนี้ได้ออกแบบและพัฒนาหุ่นยนต์หยิบและวางชิ้นงานแบบวงกลม (Circular Pick and Place Robot) สำหรับลำเลียงชิ้นงาน Connection Rod ในกระบวนการผลิตเชิงอุตสาหกรรม โดยครอบคลุมวิศวกรรมทุกมิติของระบบ Cyber-Physical

ด้านเชิงกล ระบบส่งกำลังแบบ 2 ชั้น (Planetary Gearbox $N_g = 24$ + Belt Drive HTD5M $N_b = 2.917$) ให้อัตราทดรวม $N = 70$ และประสิทธิภาพ $\eta = 0.836$ การออกแบบเพลาตาม ASME MSST ได้ขนาดขั้นต่ำ 10.59 mm และเลือกใช้ 17 mm ที่จุดวิกฤต (FOS ≥ 1.6) การวิเคราะห์ FEA พบการโก่งตัวสูงสุด 0.2281 mm และ FOS ต่ำสุด 11.03 ซึ่งเป็นไปตามข้อกำหนด M3 ทุกประการ

ด้านไฟฟ้า ตู้ควบคุมออกแบบตามมาตรฐานอุตสาหกรรม แบ่ง Power Bus ออกเป็น High Power (24 V) และ Logic (5 V) อย่างชัดเจน E-Stop ทำงานในระดับ Hardware แบบ Fail-Safe และมีวงจรป้องกัน Overcurrent, Surge และ Short Circuit ตามข้อกำหนด E6 และ E7

ด้านการควบคุม การระบุระบบด้วย Locked Rotor Test และ Chirp Excitation ให้ค่าพารามิเตอร์ที่สมบูรณ์ครบ 8 ตัว ตัวควบคุม Cascade PID ออกแบบด้วย Root Locus ได้ Settling Time 0.450 s และ Overshoot 0.00% จากการ Simulation ZV Input Shaper ลด Rod Settling Time จาก 6.44 s เหลือ ≈ 0 s ต่อ Move ทำให้ Cycle Time รวมเหลือ 30 s (Simulation) ซึ่งเป็นไปตาม P1 Kalman Filter 4 สถานะใช้ Steady-State Gain จาก DARE ลด Computational Load บน STM32 ได้อย่างมีประสิทธิภาพ

ด้านซอฟต์แวร์และการบูรณาการ Firmware บน STM32G474RE ประกอบด้วยโมดูลครบถ้วน ได้แก่ Cascade PID, ZV Shaper, Kalman Filter, FSM, Modbus RTU และ Joystick Interface การสื่อสารกับ Base System ผ่าน Modbus RTU (19200 bps, 8E1, Slave Address 21) พร้อม Python asyncio WebSocket Backend

6.2. ปัญหา อุปสรรค และบทเรียนที่ได้

6.2.1 ปัญหาเชิงกล

- **ความคลาดเคลื่อนของชิ้นส่วนที่ผลิตภายนอก:** รูไม่ตรงแนวศูนย์และผิวหน้าไม่ได้รับการปาดตามแบบ สะท้อนให้เห็นความสำคัญของการ Inspection ชิ้นส่วนก่อนประกอบ และการสื่อสาร Tolerance ให้ชัดเจนกับผู้ผลิต
- **Laser Cut Tolerance:** ชิ้นส่วนที่ตัดด้วย Laser Cut มี Tolerance สะสมที่ส่งผลต่อการ Align โดยเฉพาะที่จุดหมุนของแขนกล แนะนำให้กำหนด Fitment Tolerance อย่างชัดเจนในแบบ และทำ Shimming ระหว่างการประกอบ
- **ขนาด Proximity Sensor ไม่ตรงกับแบบ:** เกิดจากการเลือก Sensor รุ่นอื่นในภายหลัง โดยไม่ได้อัพเดตแบบ CAD ก่อนส่งผลิต ควรล็อก BOM ก่อน Finalize แบบ

6.2.2 ปัญหาด้านซอฟต์แวร์และการบูรณาการ

- **Abnormal Heartbeat ใน Demo 1:** Backend `main.exe` ของ Base System มีปัญหาคอขวด ทำให้ Heartbeat Timeout และระบบ Handshake ไม่สำเร็จ แก้ไขโดยเปลี่ยน Backend เป็น Python asyncio WebSocket บทเรียน: ควรทดสอบ End-to-End Integration กับ Base System จริงก่อน Demo อย่างน้อย 1 สัปดาห์
- **Joystick Mode Blocking Bug:** พบว่าคำสั่ง Joystick บางตัวไม่ถูก Block เมื่อระบบอยู่ใน Base System Mode เนื่องจาก Mode Check Logic ไม่ครอบคลุมทุก Command Path แก้ไขโดยเพิ่ม Guard Condition ที่ Entry Point ของ `Motor_ProcessCommand`
- **Gripper พังระหว่าง Demo 2:** Base System ส่งคำสั่งตำแหน่งผิดพลาด ทำให้แขนกลเคลื่อนที่ผิดทิศและชน Connection Rod บทเรียน: ควรมี Software Workspace Limit ที่ตรวจสอบทุก Incoming Command จาก Base System ก่อน Execute เสมอ และควรทดสอบ Safety Boundary ให้ครบก่อน Full Demo

6.2.3 ปัญหาด้านการออกแบบระบบควบคุม

- **SysID Preprocessing vs Raw Data:** การกรองสัญญาณด้วย Butterworth $f_c = 10$ Hz ก่อน Estimation ให้ผลดีกว่าข้อมูลดิบ แต่มีความเสี่ยงจาก Phase Shift ที่แก้ไขด้วยการใช้ `filtfilt` การตัดสินใจนี้ถูกต้องสำหรับโครงงานนี้ แต่ควรรายงาน Validation ให้ชัดเจนในทุกกรณีที่ Preprocess ข้อมูล
- **Bandwidth Separation Heuristic:** การตั้ง Inner Loop เร็วกว่า Outer Loop 6 เท่าเป็น Rule of Thumb ไม่ใช่เกณฑ์ที่เข้มงวด ในการออกแบบที่ต้องการความแม่นยำสูงขึ้น ควรออกแบบ Inner Loop ก่อน แล้ว Model Closed-loop Inner Loop เป็น Plant ที่แท้จริง สำหรับการออกแบบ Outer Loop ต่อไป
- **ZV Shaper Sensitivity:** ZV Shaper มีความไวต่อความคลาดเคลื่อนของ $\omega_{n,rod}$ หากค่าจริงต่างจากที่ออกแบบ $\pm 5\%$ Residual Vibration จะเพิ่มขึ้น สำหรับระบบที่ต้องการความทนทานสูง ควรพิจารณาใช้ ZVD (3-Impulse) แทน ZV (2-Impulse)

6.3. ข้อเสนอแนะสำหรับการพัฒนาต่อยอด

6.3.1 การพัฒนาระบบควบคุม

- **Hardware Tuning ของ PID Gains:** ค่า Gain ที่ใช้งานปัจจุบันมาจากการออกแบบบน Reduced-Order Model ควรทำ Hardware Commissioning เพื่อปรับ Gain Fine-tuning โดยเฉพาะ $K_{i,vel}$ ที่มีขนาดใหญ่ และตรวจสอบ Anti-windup ว่าทำงานถูกต้องภายใต้ Load จริง
- **เปิดใช้งาน Feedforward Gains:** $K_{vff} = 3.033$ V/(rad/s) และ $K_{aff} = 0.445$ V/(rad/s^2) ถูก Disable ระหว่าง Commissioning เมื่อ SysID Uncertainty ต่ำกว่า 10% แล้ว ควรเปิดใช้งานเพื่อลด Tracking Error ระหว่างการเคลื่อนที่
- **Upgrade เป็น ZVD Shaper:** ZVD (3-Impulse) มีความทนทานต่อความคลาดเคลื่อนของ ω_n มากกว่า ZV ควรพิจารณาเมื่อทราบ Uncertainty ของ $\omega_{n,rod}$ จากการวัดจริงแล้วว่าเกิน 10%

- **Online Parameter Estimation:** ในอนาคตสามารถ Implement Recursive Least Squares (RLS) เพื่อ Update $\omega_{n,rod}$ แบบ Online เมื่อสภาพของ Connection Rod เปลี่ยนแปลงตามการใช้งาน

6.3.2 การพัฒนาระบบกล

- **ลด Backlash ของ Belt Drive:** ออกแบบ Tensioner ที่ปรับระยะได้ (Adjustable Idler) เพื่อให้สามารถรักษา Belt Tension ได้อย่างสม่ำเสมอ ตลอดอายุการใช้งาน
- **ปรับปรุงความแข็งแกร่งของ Gripper Mounting:** เพิ่ม Reinforcement ที่จุดยึด Gripper เพื่อป้องกันการเสียหายจากการชนในกรณี Software Fault

6.3.3 การพัฒนาระบบไฟฟ้าและซอฟต์แวร์

- **Implement Software Workspace Limit:** เพิ่มการตรวจสอบ Incoming Command ทุกตัวจาก Base System ว่าอยู่ภายใน Safe Workspace ก่อน Execute เพื่อป้องกันการชนซ้ำ
- **Upgrade การสื่อสาร Gripper:** ในอนาคตสามารถ Implement CAN Bus (FDCAN1) เพื่อให้รองรับ Gripper Node ที่ซับซ้อนขึ้น พร้อม Watchdog และ Feedback ที่ครบถ้วนกว่า Digital I/O
- **Data Logging และ Monitoring:** เพิ่ม SD Card หรือ Wireless Telemetry สำหรับบันทึก Log ระยะยาว เพื่อ Predictive Maintenance และวิเคราะห์ Drift ของ Bearing

บรรณานุกรม

(2010). *ISO 281: Rolling bearings -- Dynamic load ratings and rating life*. International Organization for Standardization.

(2015). *MD10C 10A DC Motor Driver User Manual*. Cytron Technologies.

(2019). *STM32G4 Series Reference Manual (RM0440)*. STMicroelectronics.

(2021). *Habasit Timing Belts Engineering Guide*. Habasit AG.

(2022). *Switching Power Supply User Manual*. Mean Well Enterprises.

(2023). *Simulink Parameter Estimator User Guide*. The MathWorks, Inc., Natick, MA, USA.

(2023). *SKF Rolling Bearings Catalogue and Engineering Guide*. SKF Group.

(2023). *SOLIDWORKS Simulation User Guide*. Dassault Systèmes.

Bolton, W. (2020). *Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering*. Pearson, 7th edition.

Budynas, R. G. and Nisbett, J. K. (2015). *Shigley's Mechanical Engineering Design*. McGraw-Hill, 10th edition.

Hughes, A. and Drury, B. (2019). *Electric Motors and Drives: Fundamentals, Types and Applications*. Newnes, 5th edition.

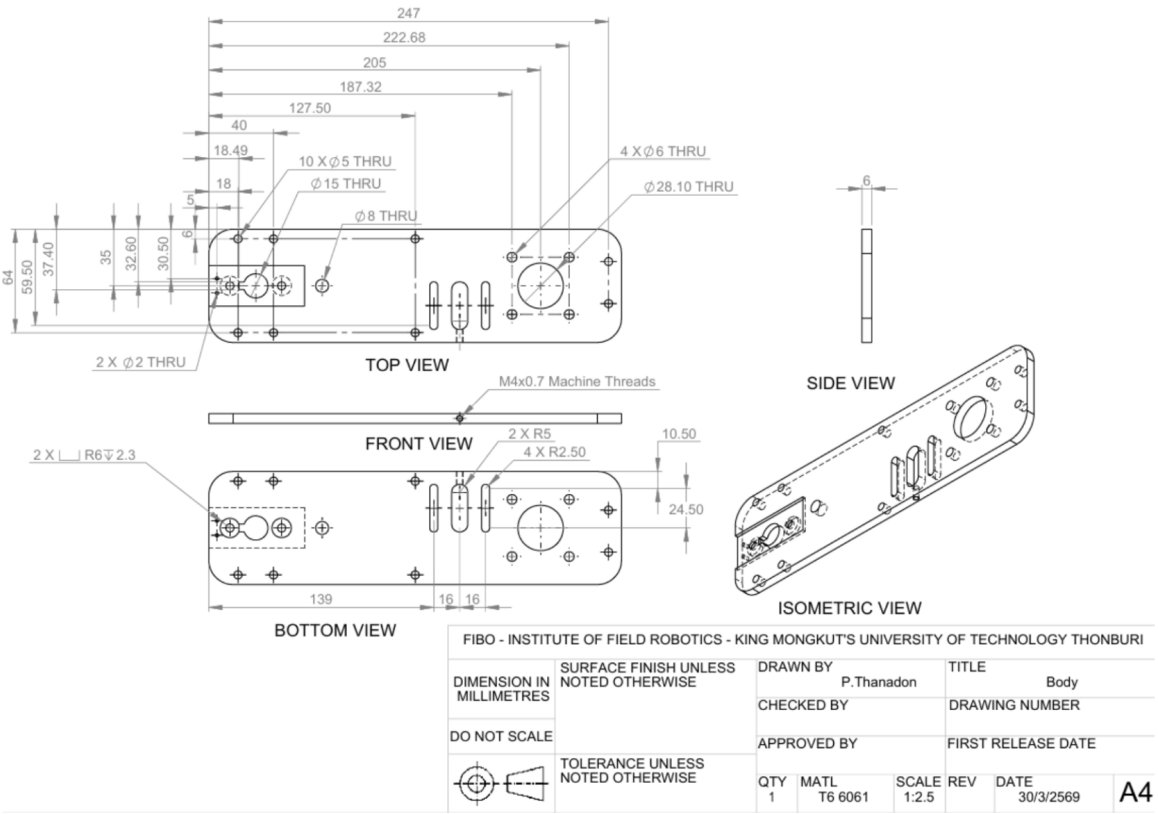
Norton, R. L. (2011). *Machine Design: An Integrated Approach*. Prentice Hall, 4th edition.

Ogata, K. (2010). *Modern Control Engineering*. Prentice Hall, 5th edition.

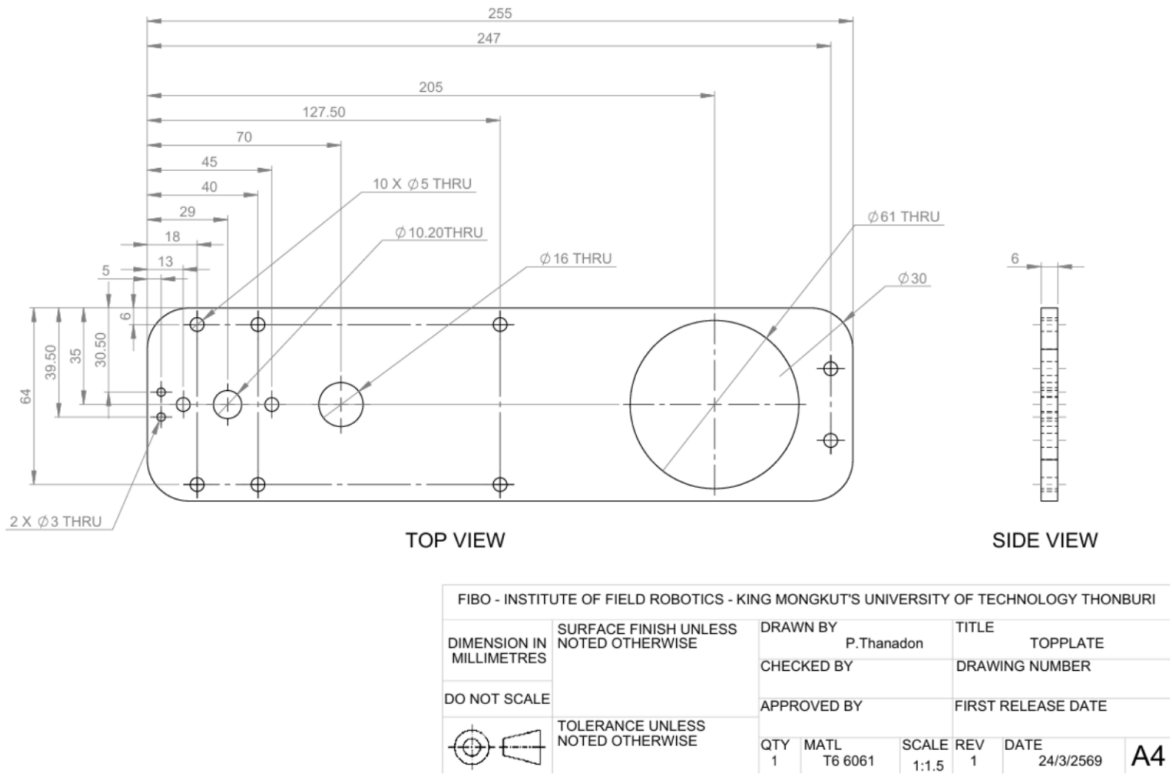
Rao, S. S. (2017). *The Finite Element Method in Engineering*. Butterworth-Heinemann, 6th edition.

บทที่ A Engineering Drawings

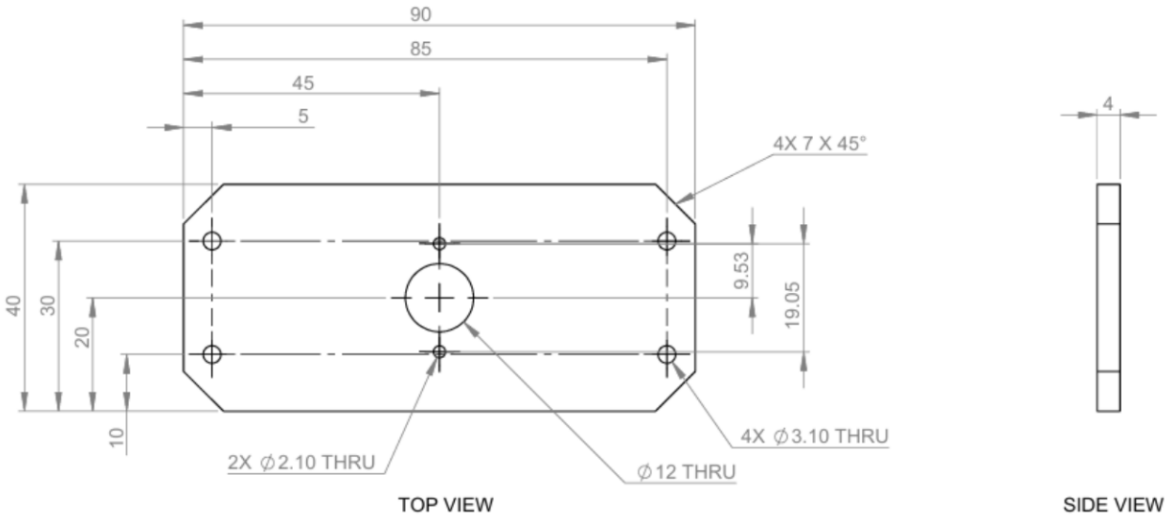
ภาคผนวกนี้รวบรวม Engineering Drawing ของชิ้นส่วนเชิงกลทั้งหมด ที่ออกแบบและผลิตในโครงงานนี้ แบบแต่ละชิ้นจัดทำตามมาตรฐาน ISO และระบุ Tolerance ตามการใช้งานจริง



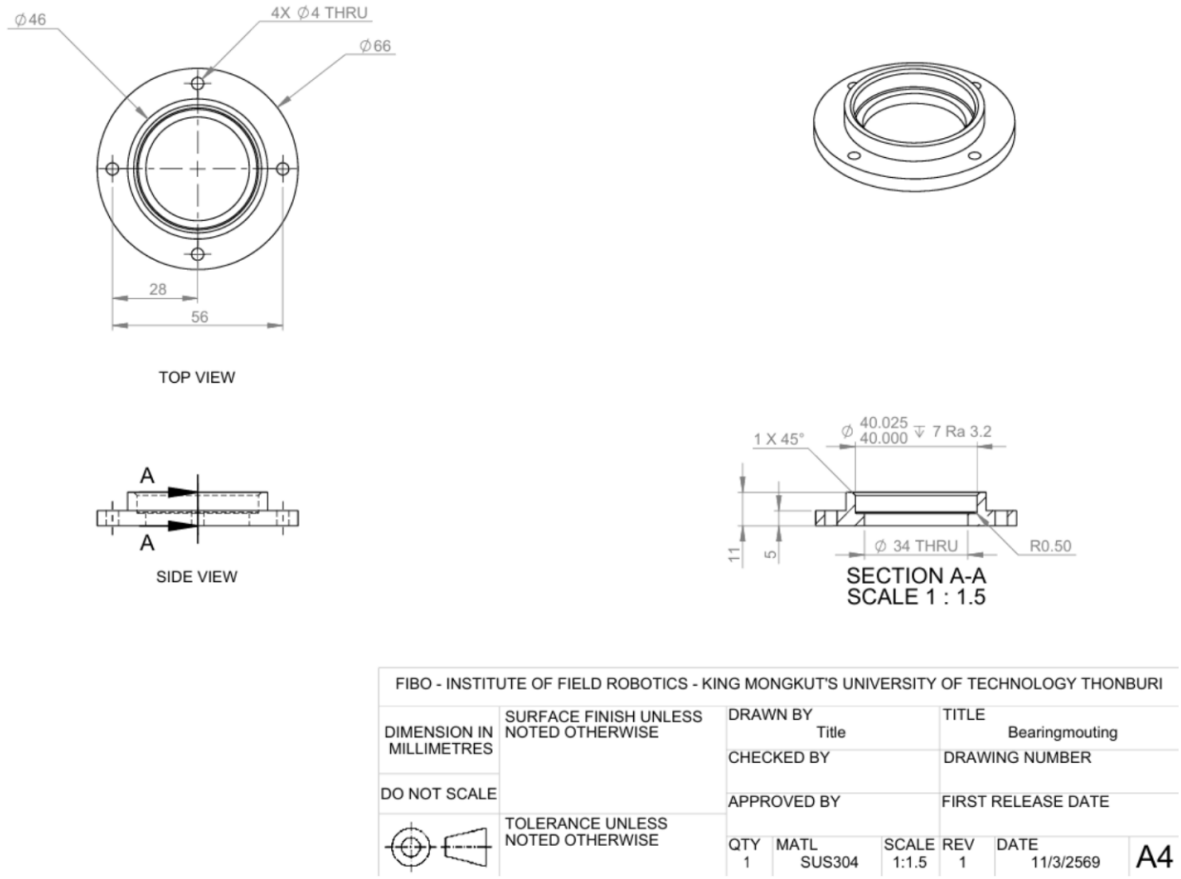
รูปที่ A.1: Engineering Drawing: Body Plate



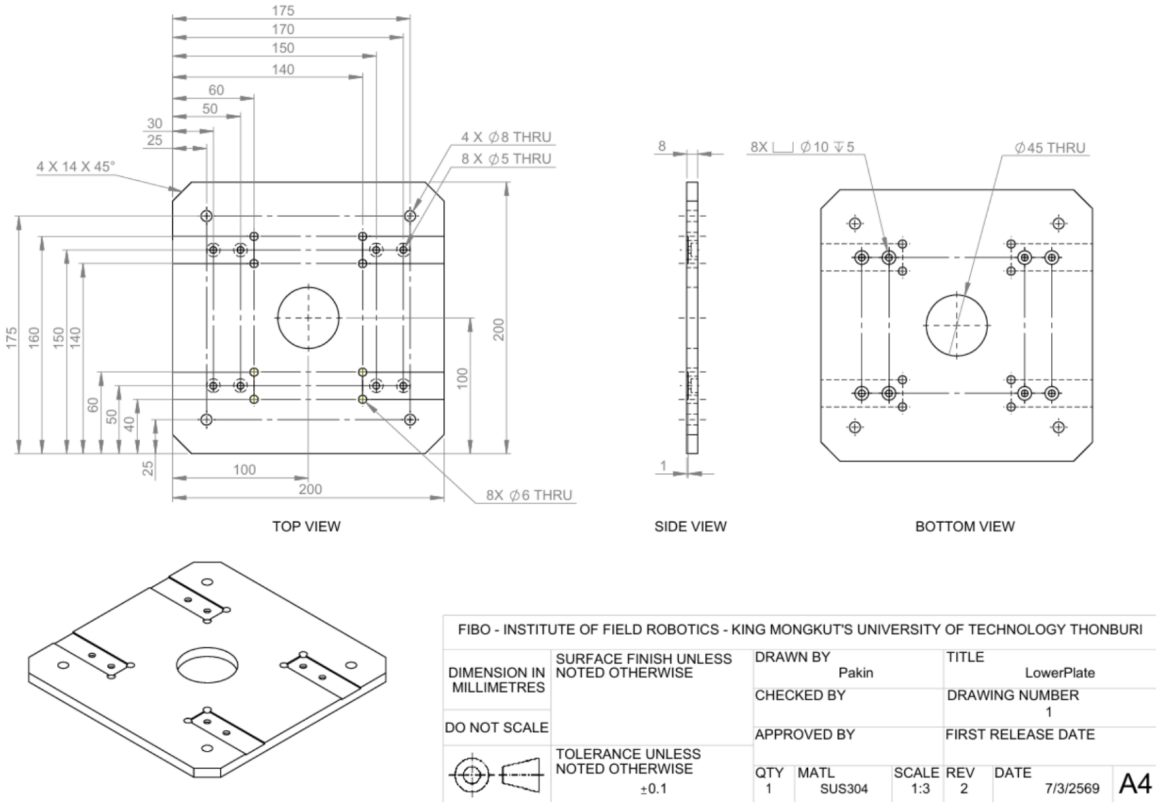
รูปที่ A.2: Engineering Drawing: Top Plate



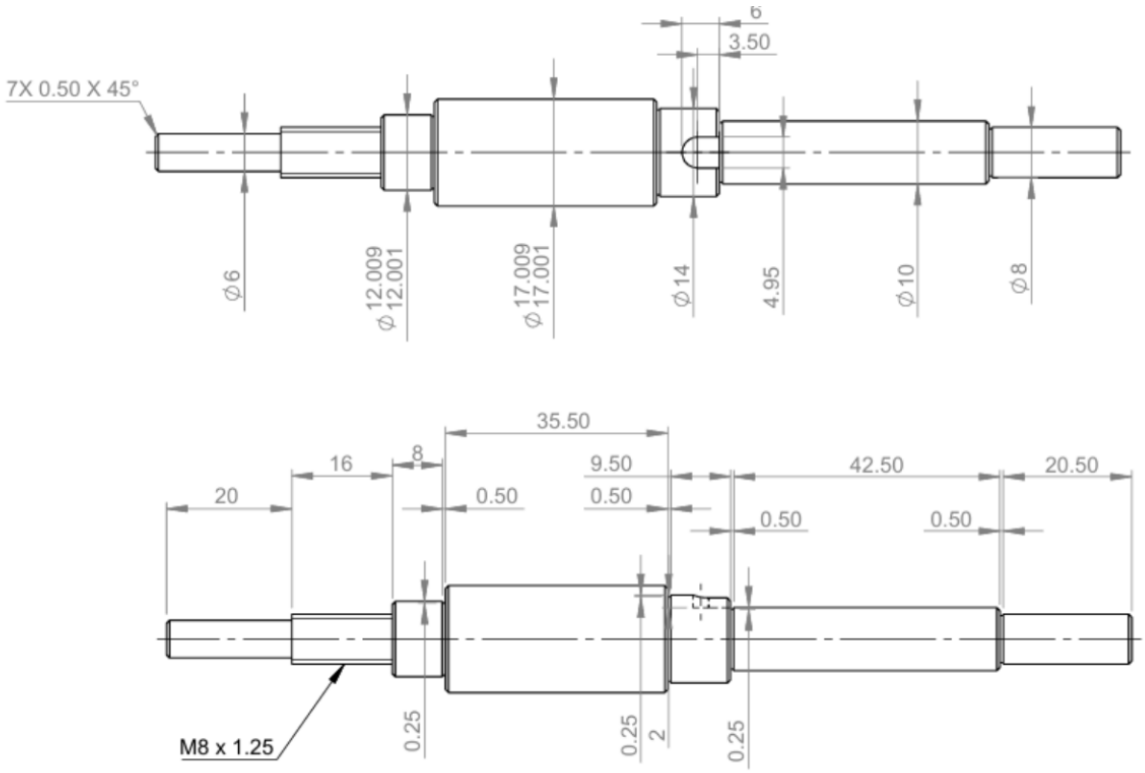
รูปที่ A.3: Engineering Drawing: AMT Encoder Mounting



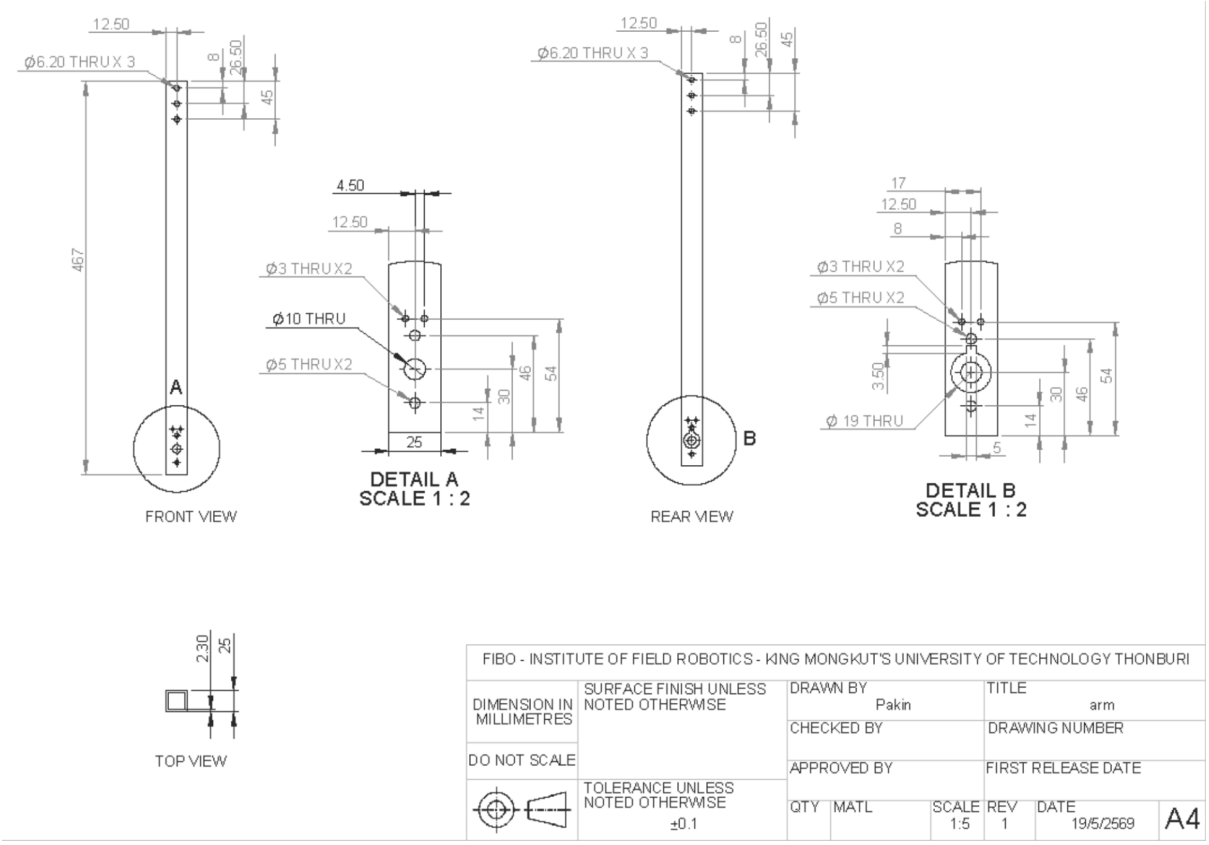
รูปที่ A.4: Engineering Drawing: Bearing Mounting



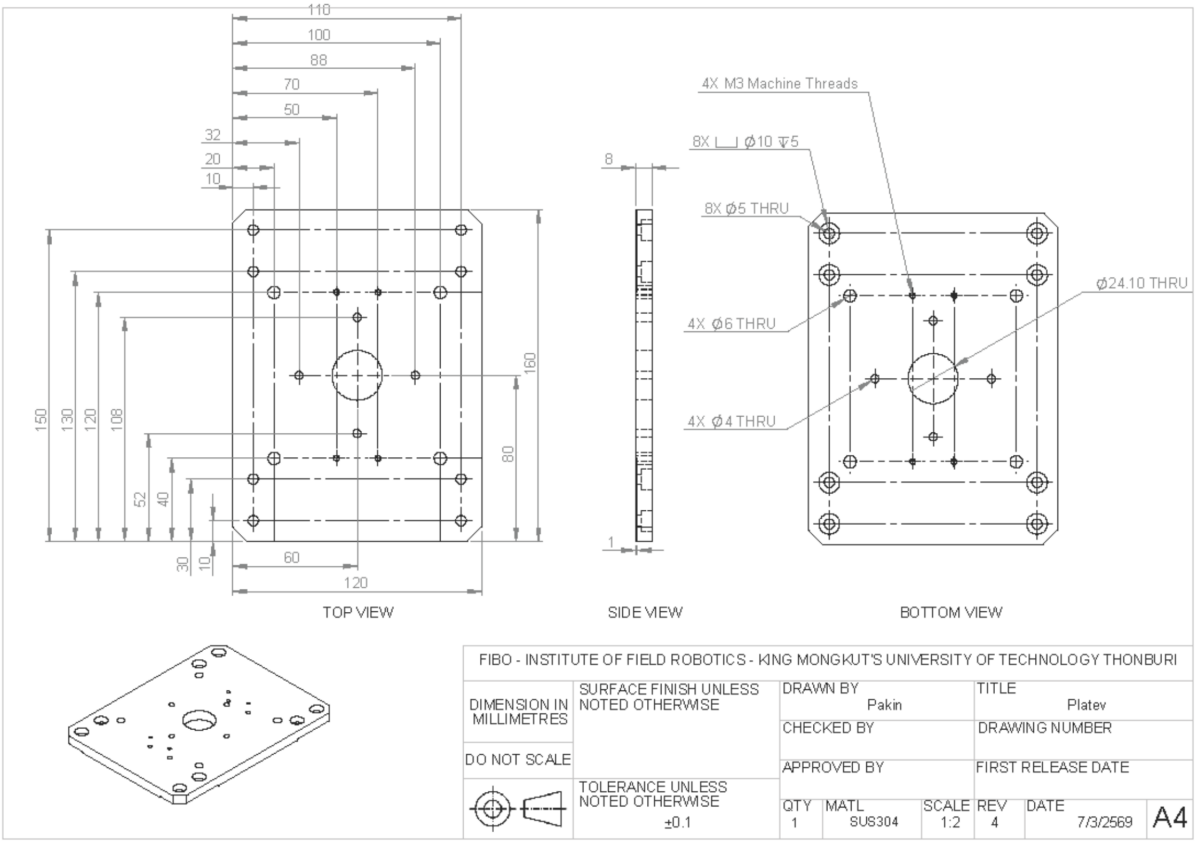
รูปที่ A.5: Engineering Drawing: Lower Plate



รูปที่ A.6: Engineering Drawing: Step Shaft



รูปที่ A.7: Engineering Drawing: Arm



รูปที่ A.8: Engineering Drawing: Plate V

บทที่ B Source Code Repository

Source Code ทั้งหมดของ STM32 Firmware สำหรับโครงงานนี้ เผยแพร่เป็นสาธารณะที่ GitHub Repository ดังนี้

https://github.com/Thadzy/STM32_Firmware_Studio_G6

Repository นี้ประกอบด้วย:

- **Cascade PID Controller** --- Velocity Inner Loop (1 kHz) และ Position Outer Loop (100 Hz) พร้อม Anti-windup Back-Calculation
- **Kalman Filter (4-State)** --- State Estimator สำหรับ $[\theta, \omega, i_a, \tau_d]^\top$ ที่ทำงานแบบ Real-time บน STM32
- **ZVD Input Shaper** --- Impulse Sequence Convolution ออกแบบจาก $\omega_{n,\mathrm{rod}}$ และ ζ_{rod}
- **S-Curve Trajectory Generator** --- 7-Phase Jerk-limited Profile สำหรับ Point-to-Point Motion
- **Modbus RTU Slave** --- FC03/FC06 บน LPUART1 (230400 baud, 8N1, Slave Address 21)
- **Two-Stage Homing Sequence** --- Fast Search + Precision Edge Detection
- **Application FSM** --- `INIT` $\rightarrow$ `HOMING` $\rightarrow$ `IDLE` $\rightarrow$ `RUNNING` $\rightarrow$ `FAULT`

วิธี Clone:

```
git clone https://github.com/Thadzy/STM32_Firmware_Studio_G6.git
```