

การพัฒนา PLC และส่วนขยายด้วย OPEN PLC บน Arduino Mega 2560 เพื่อใช้งานแขนหุ่นยนต์ขนาดเล็ก

สมชาย เตียเจริญ¹ และ สมพร เตียเจริญ^{2*}

¹วิศวกร ภาควิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์ วิทยาลัยเทคโนโลยีอุตสาหกรรม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

²อาจารย์ ภาควิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์ วิทยาลัยเทคโนโลยีอุตสาหกรรม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

*somporn.t@cit.kmutnb.ac.th

บทคัดย่อ

งานวิจัยนี้นำเสนอการพัฒนาและสร้างระบบควบคุมลอจิกแบบโปรแกรมได้ (Programmable Logic Controller: PLC) โดยใช้บอร์ด Arduino Mega 2560 เป็นหน่วยประมวลผลหลัก เพื่อเป็นต้นแบบ PLC ต้นทุนต่ำสำหรับการเรียนรู้และการประยุกต์ใช้งานในระบบอุตสาหกรรมขนาดเล็ก การพัฒนาโปรแกรมควบคุมใช้ซอฟต์แวร์ OpenPLC Editor ในการออกแบบและเขียน Ladder Logic จากนั้นทำการอัปโหลดโปรแกรมลงสู่บอร์ดเพื่อควบคุมอินพุตและเอาต์พุตของระบบ ระบบถูกออกแบบให้สามารถสั่งงานแขนหุ่นยนต์ขนาดเล็กโดยใช้ Arduino Uno ทำให้ผู้ใช้งานสามารถควบคุมการทำงานของแขนหุ่นยนต์ขนาดเล็กได้ โดยมีการกำหนดตัวแปรที่อยู่ใน PLC ตามมาตรฐาน IEC 61131-3 ขึ้นใหม่บน OpenPLC เพื่อรับและส่งข้อมูลจาก Arduino Uno จากการทดสอบการทำงานพบว่าระบบสามารถทำงานได้ถูกต้องตามขอบเขตที่ออกแบบไว้ PLC มีอินพุตเอาต์พุตแบบดิจิทัล 54 ขาและแบบอนาล็อก 16 ขา สามารถเขียนโปรแกรมแลตเตอร์ลอจิก มีความเสถียรในระดับที่เหมาะสมสำหรับการใช้งานด้านการศึกษาและงานทดลอง สำหรับโปรแกรมพื้นฐานมีเวลาในการอ่านอินพุตประมวลผลแล้วส่งออกเอาต์พุตประมาณ 59.6 μ s และใช้เวลาเพิ่มขึ้นเมื่อประมวลผล Ladder Extension ที่พัฒนาเพิ่มเพื่อสั่งงานแขนหุ่นยนต์ งานวิจัยนี้จึงเป็นแนวทางหนึ่งในการพัฒนาระบบควบคุมอัตโนมัติที่มีต้นทุนต่ำ และสามารถนำไปต่อยอดเพื่อใช้งานในงานอุตสาหกรรมขนาดเล็กหรือการเรียนการสอนได้ในอนาคต

คำสำคัญ: โอเพ่นพีแอลซี แขนหุ่นยนต์ แลตเตอร์ลอจิก ระบบควบคุมอัตโนมัติที่มีต้นทุนต่ำ อาคูโอโนโอไมเก้า 2560

Development of a PLC and its extensions using OPEN PLC on an Arduino Mega 2560 to control a small robotic arm.

Somchai Tiacharoen¹ and Somporn Tiacharoen^{2*}

¹Engineer Electronics Engineering Technology College of Industrial Technology,
King Mongkut's University of Technology North Bangkok, Thailand

²Lecturer Electronics Engineering Technology College of Industrial Technology,
King Mongkut's University of Technology North Bangkok, Thailand

*somporn.t@cit.kmutnb.ac.th

Abstract

This research presents the development and construction of a Programmable Logic Controller (PLC) using an Arduino Mega 2560 board as the main processor. This serves as a low-cost PLC prototype for learning and application in small-scale industrial systems. The control program is designed and written using OpenPLC Editor software, with the ladder logic programmed into the board. The program is then uploaded to the board to control the system's inputs and outputs. The system is designed to control a small robotic arm using an Arduino Uno, allowing users to control the robot's operation. Variables in the PLC are redefined on OpenPLC according to the IEC 61131-3 standard to receive and send data from the Arduino Uno. Testing showed the system functioned correctly within the designed limits. The PLC has 54 digital input/output pins and 16 analog input/output pins. Ladder logic programming is possible, with a stability level suitable for educational and experimental applications. The basic program has an input read/process/output time of approximately 59.6 μ s, increasing when processing the developed ladder extensions for robotic arm control. This research represents a promising approach for developing low-cost automated control systems. And it can be further developed for use in small-scale industries or in education in the future.

Keywords: OpenPLC, Robot Arm, Ladder Logic, low-cost automated control, Arduino Mega 2560

1. บทนำ

ในปัจจุบันระบบควบคุมอัตโนมัติมีบทบาทสำคัญอย่างยิ่งในงานอุตสาหกรรม ไม่ว่าจะเป็นกระบวนการผลิต ระบบสายพานลำเลียง เครื่องจักรอัตโนมัติ หรือระบบควบคุมกระบวนการต่าง ๆ โดยอุปกรณ์ที่นิยมใช้งานอย่างแพร่หลายคือ Programmable Logic Controller (PLC) ซึ่งเป็นอุปกรณ์ควบคุมที่สามารถเขียนโปรแกรมเพื่อกำหนดลำดับการทำงานได้อย่างยืดหยุ่น มีความทนทาน และเหมาะสมกับสภาพแวดล้อมทางอุตสาหกรรม

อย่างไรก็ตาม PLC มีราคาค่อนข้างสูง ทำให้เป็นอุปสรรคต่อการศึกษาและการทดลองสำหรับผู้เรียนหรือสถานศึกษาที่มีงบประมาณจำกัด ดังนั้นจึงมีแนวคิดในการพัฒนาระบบ PLC ต้นทุนต่ำโดยใช้ไมโครคอนโทรลเลอร์ เช่น Arduino Mega

2560 มาเป็นหน่วยประมวลผลหลัก ร่วมกับซอฟต์แวร์ OpenPLC Editor ในการพัฒนาโปรแกรมควบคุมแบบ Ladder Logic ซึ่งเป็นรูปแบบการเขียนโปรแกรมที่นิยมใช้ในงานอุตสาหกรรม

นอกจากนี้ การประยุกต์ใช้ PLC เป็นตัวควบคุมในการสื่อสารข้อมูลกับแขนหุ่นยนต์ขนาดเล็ก ช่วยเพิ่มความสะดวกในการควบคุมการทำงานและตรวจสอบสถานะของแขนหุ่นยนต์ขนาดเล็กแบบเรียลไทม์ ทำให้ระบบมีความทันสมัยและสอดคล้องกับแนวคิดอุตสาหกรรม

2. ทฤษฎีที่สำคัญและงานวิจัยที่เกี่ยวข้อง

2.1. Programmable Logic Controller (PLC)

Programmable Logic Controller (PLC) คืออุปกรณ์ควบคุมทางอุตสาหกรรมที่ถูกออกแบบมาเพื่อใช้ควบคุมกระบวนการทำงานของเครื่องจักรหรือระบบอัตโนมัติ โดยสามารถเขียนโปรแกรมเพื่อกำหนดลำดับการทำงานได้ PLC ประกอบด้วยหน่วยประมวลผลกลาง (CPU), หน่วยความจำ, โมดูลอินพุต/เอาต์พุต (I/O Module) และแหล่งจ่ายไฟฟ้า

2.2 ไมโครคอนโทรลเลอร์ และ Arduino Mega 2560



ภาพที่ 1 Arduino Mega 2560

ไมโครคอนโทรลเลอร์ (Microcontroller) คืออุปกรณ์อิเล็กทรอนิกส์ที่รวมหน่วยประมวลผล หน่วยความจำ และพอร์ตอินพุต/เอาต์พุตไว้ในชิปเดียวกัน เหมาะสำหรับงานควบคุมแบบฝังตัว (Embedded System)

Arduino Mega 2560 ใช้ชิป ATmega2560 มีจำนวนขา Digital I/O และ Analog Input มากกว่าบอร์ด Arduino รุ่นพื้นฐาน เหมาะสำหรับงานที่ต้องการอินพุตและเอาต์พุตจำนวนมาก จึงสามารถประยุกต์ใช้เป็นหน่วยประมวลผลหลักในการพัฒนา PLC ต้นทุนต่ำได้

2.3 OpenPLC และ Ladder Logic

OpenPLC Editor เป็นซอฟต์แวร์โอเพนซอร์สสำหรับพัฒนาโปรแกรมควบคุมตามมาตรฐาน IEC 61131-3 รองรับภาษาโปรแกรมหลายรูปแบบ เช่น Ladder Diagram (LD), Function Block Diagram (FBD) และ Structured Text (ST)

Ladder Logic (Ladder Diagram) เป็นภาษาการเขียนโปรแกรมที่เลียนแบบวงจรรีเลย์ไฟฟ้า ประกอบด้วยสัญลักษณ์หน้าสัมผัส (Contact) และคอยล์ (Coil) เหมาะสำหรับงานควบคุมลอจิกแบบดิจิทัล และเป็นภาษาที่นิยมใช้ในงานอุตสาหกรรม

3. วิธีดำเนินการ

3.1 หลักการทำงาน

บทความสร้าง PLC ด้วย Arduino Mega 2560 ที่พัฒนาขึ้น โดยใช้บอร์ด Arduino Mega 2560 เป็นหน่วยประมวลผลหลัก ร่วมกับซอฟต์แวร์ OpenPLC Editor ในการพัฒนาโปรแกรมควบคุมแบบ Ladder Logic และส่งข้อมูลการเคลื่อนที่ของแขนหุ่นยนต์ผ่านพอร์ตการสื่อสารแบบอนุกรม

หลักการทำงานของระบบเป็นไปตามแนวคิด PLC Scan Cycle ซึ่งประกอบด้วยขั้นตอนหลัก ได้แก่ การอ่านอินพุต การประมวลผลโปรแกรม และการกำหนดเอาต์พุต

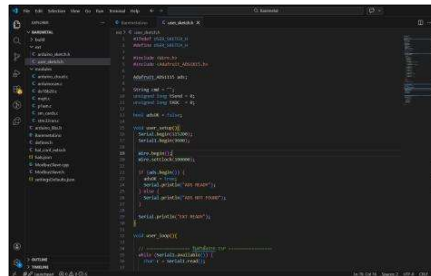
3.1.1 ส่วนการประมวลผลหลัก (Arduino Mega 2560)

บอร์ด Arduino Mega 2560 ทำหน้าที่เสมือน PLC โดยทำการประมวลผลคำสั่งตามโปรแกรม Ladder Logic ที่พัฒนาด้วย OpenPLC Editor ภายในขั้นตอนนี้ระบบจะทำงานตามหลักการ Scan Cycle คือ

- 1.อ่านค่าข้อมูลที่ได้รับ
- 2.ประมวลผลตามเงื่อนไขลอจิก
- 3.กำหนดสถานะเอาต์พุต

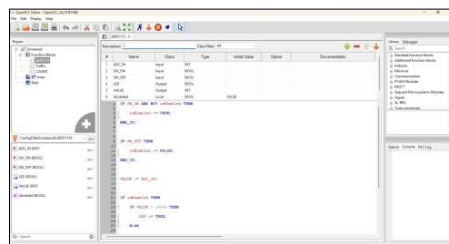
3.1.2 PLC Ladder Function Block และ Extension

ส่วนนี้เป็นฟังก์ชันเสริมของ PLC และลอจิกควบคุมต่าง ๆ ซึ่งถูกเรียกใช้ระหว่างการประมวลผล เพื่อกำหนดเงื่อนไขการทำงานของระบบให้เป็นไปตามที่ออกแบบไว้



ภาพที่ 2 Extension ของระบบ

ส่วน Extension เป็นโปรแกรมที่ทำงานอยู่บนบอร์ด Arduino Mega 2560 ซึ่งทำหน้าที่ขยายความสามารถของระบบ PLC ให้สามารถรับ - ส่งข้อมูลกับอุปกรณ์ภายนอกและประมวลผลข้อมูลเพิ่มเติมได้



ภาพที่ 3 การออกแบบและกำหนดเงื่อนไข Function Block

ส่วน PLC Function Block ใช้เพิ่มเติม เพื่อใช้ร่วมกับ Ladder Logic โดยใช้ภาษา Structured Text (ST) ในการพัฒนาอัลกอริทึมควบคุมเฉพาะของระบบ ซึ่งช่วยให้สามารถสร้างฟังก์ชันที่มีความซับซ้อนมากขึ้นกว่าการใช้ Ladder Diagram เพียงอย่างเดียว Function Block ที่พัฒนาขึ้นมา สามารถนำไปใช้งานภายใน Ladder Program ได้เหมือนบล็อกมาตรฐานของ PLC เช่น

- 1.การประมวลผลค่าจากเซนเซอร์
- 2.การคำนวณข้อมูล
- 3.การกำหนดเงื่อนไขลอจิกเฉพาะของระบบ

เมื่อโปรแกรมถูกคอมไพล์ ระบบของ OpenPLC จะทำหน้าที่แปลงโปรแกรม Ladder Logic และ Structured Text ให้เป็นโค้ดที่สามารถทำงานบนฮาร์ดแวร์ PLC ได้ โดยในบทความนี้ใช้บอร์ด Arduino Mega 2560 เป็นหน่วยประมวลผลหลักของระบบ

3.1.3 ส่วนการแสดงผลและควบคุมอุปกรณ์ (Output)

เมื่อประมวลผลเสร็จสิ้น ระบบจะส่งสัญญาณควบคุมไปยังอุปกรณ์ปลายทาง เช่น หลอดไฟ ไฟจราจร หรืออุปกรณ์ไฟฟ้าอื่น ๆ เพื่อให้ทำงานตามเงื่อนไขที่กำหนด

3.2 การออกแบบระบบ

3.2.1 การกำหนดขอบเขตของระบบ

1. รับสัญญาณอินพุตดิจิทัล
2. ประมวลผลตามโปรแกรม Ladder Logic
3. ควบคุมเอาต์พุต
4. แสดงผลสถานะผ่าน output
5. รองรับการสั่งงานแบบเรียลไทม์ผ่าน Serial

3.2.2 การออกแบบโครงสร้างระบบ

โครงสร้างระบบแบ่งออกเป็น 3 ส่วนหลัก

1. ส่วนควบคุมหลัก (PLC Unit) ใช้ Arduino Mega 2560 ทำหน้าที่ประมวลผลลอจิก
2. ส่วนการสื่อสาร (Communication Unit) ใช้พอร์ตอนุกรมรับ - ส่งข้อมูลระหว่าง PLC และ Arduino Uno
3. ส่วนการแสดงผลและควบคุมใช้สำหรับแสดงสถานะอินพุต/เอาต์พุตและรับคำสั่งจากผู้ใช้งาน

ลำดับการทำงานของระบบเป็นไปตามหลักการ Scan Cycle ของ PLC ได้แก่

1. อ่านค่าจากอินพุต
2. ประมวลผลตามโปรแกรม
3. ส่งค่าผลลัพธ์ไปยังเอาต์พุต
4. ส่งข้อมูลสถานะไปยังแขนหุ่นยนต์ขนาดเล็ก

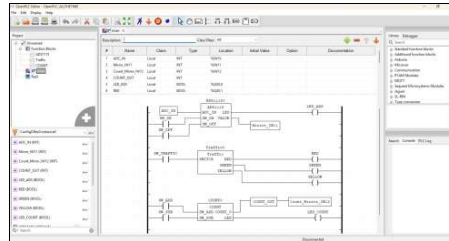
3.3 การออกแบบและติดตั้งฮาร์ดแวร์

1. ติดตั้ง Arduino Mega 2560 เป็นหน่วยควบคุมหลัก
2. เชื่อมต่อโมดูลสวิทช์หรือ LED เข้ากับขา Digital Input/Output
3. เชื่อมต่อแขนหุ่นยนต์ขนาดเล็กผ่านการสื่อสารพอร์ตอนุกรม
4. จัดเตรียมแหล่งจ่ายไฟให้เหมาะสมกับแต่ละอุปกรณ์
5. ตรวจสอบการต่อวงจรให้ถูกต้อง

3.4 การพัฒนาโปรแกรมควบคุม

3.4.1 การพัฒนาโปรแกรม PLC

- 1.ออกแบบ Ladder Diagram ใน OpenPLC Editor
- 2.กำหนดอินพุตและเอาต์พุตตามผังงาน
- 3.คอมไพล์และอัปโหลดโปรแกรมลงบอร์ด



ภาพที่ 4 Ladder Logic

3.4.2 การพัฒนาระบบสื่อสารและควบคุมแขนหุ่นยนต์

- 1.เขียนโปรแกรมใน PLC เพื่อรับ-ส่งข้อมูล
- 2.ออกแบบส่วน Extension ให้ควบคุมแขนหุ่นยนต์และแสดงสถานะที่เอาต์พุต
- 3.ทดสอบสั่งงานจาก PLC ไปยังแขนหุ่นยนต์
- 4.ตรวจสอบความถูกต้องของข้อมูลแบบเรียลไทม์

4. ผลการทดลอง

การทดลองนี้มีวัตถุประสงค์เพื่อทดสอบประสิทธิภาพการทำงานของระบบ PLC ที่พัฒนาขึ้นโดยใช้ Arduino Mega 2560 ร่วมกับ OpenPLC Editor และใช้ Extension ในการสื่อสารข้อมูลกับระบบแขนหุ่นยนต์

การทดสอบแบ่งออกเป็น 3 ส่วนหลัก ได้แก่

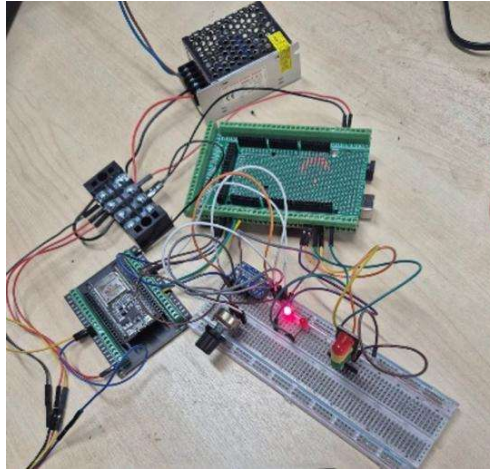
1. การทดสอบการทำงานของลอจิกควบคุม
2. การทดสอบการสื่อสารข้อมูล
3. การทดสอบความเสถียรของระบบ

4.1 การทดสอบอินพุตและเอาต์พุตดิจิทัล

ทำการป้อนสัญญาณอินพุต และตรวจสอบการตอบสนองของเอาต์พุต เช่น การสั่งงานหลอดไฟแสดงสถานะผลการ

ทดลอง:

- 1.ระบบสามารถอ่านค่าอินพุตได้ถูกต้อง
- 2.เอาต์พุตตอบสนองตามเงื่อนไข Ladder Logic ที่กำหนด
- 3.ไม่พบความผิดพลาดในการประมวลผลลอจิกพื้นฐาน



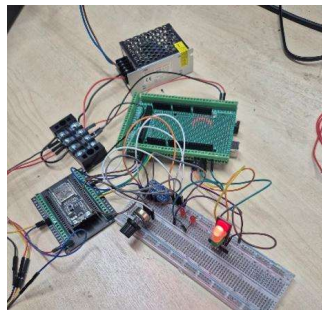
ภาพที่ 5 การทำงานของระบบทดสอบอินพุตและเอาต์พุตดิจิทัล

4.2 ผลการทดสอบส่วน Extension ให้ควบคุมแขนหุ่นยนต์และแสดงสถานะที่เอาต์พุต

ทดสอบการแสดงผลสถานะเอาต์พุตและการสั่งงานแขนหุ่นยนต์จาก PLC ด้วยโปรโตคอลการสื่อสาร RS232 ผลการ

ทดลอง:

- 1.LED สามารถแสดงสถานะได้แบบเรียลไทม์
- 2.คำสั่งจาก PLC ถูกส่งถึงแขนหุ่นยนต์ได้ถูกต้อง



ภาพที่ 6 การทำงานของระบบควบคุมแขนหุ่นยนต์

4.3 ผลการทดสอบความเสถียรของระบบ

ทำการทดสอบการทำงานต่อเนื่องเป็นระยะเวลา 4 - 6 ชั่วโมง ผลการทดลอง:

- 1.ระบบสามารถทำงานได้ต่อเนื่องโดยไม่เกิดการรีเซ็ตตัวเอง
- 2.ไม่พบปัญหาการค้างของโปรแกรม
- 3.การสื่อสารระหว่าง PLC และควบคุมแขนหุ่นยนต์มีความต่อเนื่อง

4.4 ผลประเมินจากการทดลองการใช้งาน

จากผลการทดลองการใช้งาน สร้าง PLC ด้วย Arduino Mega 2560 โดยใช้งานระบบ Programmable Logic Controller (PLC) ที่พัฒนาขึ้นโดยใช้ Arduino Mega 2560 ร่วมกับแขนหุ่นยนต์ พบว่าระบบสามารถทำงานได้อย่างมีประสิทธิภาพและสอดคล้องกับวัตถุประสงค์ของบทความ

4.5 การเปรียบเทียบไมโครคอนโทรลเลอร์ Arduino UNO และ Arduino Mega 2560 สำหรับงาน PLC

ในการพัฒนาระบบ Programmable Logic Controller (PLC) โดยใช้ไมโครคอนโทรลเลอร์เป็นหน่วยประมวลผลหลัก การเลือกบอร์ดที่เหมาะสมมีความสำคัญอย่างยิ่ง เนื่องจากมีผลต่อจำนวนอินพุต/เอาต์พุต ความสามารถในการ



ประมวลผล และความยืดหยุ่นในการพัฒนาโปรแกรม ในบทความนี้ได้ทำการพิจารณาเปรียบเทียบระหว่างบอร์ด Arduino UNO และ Arduino Mega 2560 ซึ่งมีรายละเอียดดังตารางต่อไปนี้

ตารางที่ 2 การเปรียบเทียบ Arduino UNO และ Arduino Mega 2560

หัวข้อ	Arduino UNO	Arduino Mega 2560
ไมโครคอนโทรลเลอร์	ATmega328P	ATmega2560
Digital I/O	14 ขา	54 ขา
Analog Input	6 ขา	16 ขา
UART (Serial)	1 ช่อง	4 ช่อง
หน่วยความจำ (Flash)	32 KB	256 KB
SRAM	2 KB	8 KB

ตารางที่ 3 การรองรับตัวแปร PLC ใน OpenPLC

ประเภทตัวแปร	Arduino UNO	Arduino Mega 2560
%IX (Digital Input)	รองรับจำกัด (14)	รองรับมาก (54)
%QX (Digital Output)	รองรับจำกัด (14)	รองรับมาก (54)
%IW (Word Input)	รองรับจำนวนน้อย (6)	รองรับมาก (16 ขึ้นไป)
%QW (Word Output)	จำกัด	รองรับได้มากกว่า

จากตารางที่ 2 ความสามารถด้านฮาร์ดแวร์และการรองรับระบบ PLC สามารถสรุปได้ว่า Arduino Mega 2560 มีความเหมาะสมในการนำมาใช้พัฒนา PLC มากกว่า Arduino UNO เนื่องจากมีจำนวน I/O ที่เพียงพอ รองรับตัวแปรในระบบได้มาก

5. สรุปผลการดำเนินงาน

บทความนี้มีวัตถุประสงค์เพื่อออกแบบและพัฒนาระบบ Programmable Logic Controller (PLC) ต้นทุนต่ำ โดยใช้บอร์ด Arduino Mega 2560 เป็นหน่วยประมวลผลหลัก และพัฒนาโปรแกรมควบคุมด้วย OpenPLC Editor ในรูปแบบ Ladder Logic เพื่อควบคุมอินพุตและเอาต์พุตของระบบ นอกจากนี้ ได้พัฒนาระบบแสดงผลและสั่งงานผ่านเซนเซอร์ โดยใช้พอร์ตอนุกรมเป็นตัวกลางในการรับ - ส่งข้อมูลระหว่าง PLC และเซนเซอร์ ผลการทดลองสามารถแสดงสถานะและควบคุมการทำงานของเซนเซอร์ได้อย่างถูกต้องตามเงื่อนไขที่กำหนด จากผลการทดลองพบว่า ระบบสามารถทำงานได้ตามขอบเขตที่ตั้งไว้ ทั้งในส่วนของ การควบคุมลอจิกพื้นฐาน เช่น การสั่งงานดิจิทัลเอาต์พุต การตั้งเวลา (Timer) และการนับจำนวน (Counter) รวมถึงการแสดงผลสถานะ ได้อย่างถูกต้องและมีความเสถียรในระดับที่เหมาะสมสำหรับงานด้านการศึกษา และการทดลอง ดังนั้น บทความนี้สามารถสรุปได้ว่าการประยุกต์ใช้ไมโครคอนโทรลเลอร์ในการพัฒนา PLC ต้นทุนต่ำมีความเป็นไปได้ และสามารถใช้เป็นสื่อการเรียนรู้เพื่อเสริมความเข้าใจเกี่ยวกับหลักการทำงานของ PLC และการเขียนโปรแกรม Ladder Logic ได้อย่างมีประสิทธิภาพ แผนพัฒนาต่อไปคือการเพิ่มระบบ HMI และการเชื่อมต่อ IoT

6. กิตติกรรมประกาศ

งานวิจัยนี้ได้รับทุนสนับสนุนจากวิทยาลัยเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ ตามสัญญาเลขที่ Res-CIT 534/2023



7. เอกสารอ้างอิง

- [1] Autonomy. (2026). OpenPLC Editor. [Online]. Retrieved March 3, 2026, from .
- [2] Arduino. (2026). Arduino Mega 2560 Rev3. [Online]. Retrieved March 3, 2026, from .
- [3] Saulius Niauronis. (2023). OpenPLC hardware speed performance comparison. Professional Studies: Theory and Practice, 12(27).
- [4] IoT Maker Thailand. (2026). What is ESP32 and basic applications. [Online]. Retrieved March 3, 2026, from . <https://www.iotmaker.in.th>
- [5] ThaiEasyElec. (2026). Introduction to Arduino and microcontrollers. [Online]. Retrieved March 3, 2026, from . <https://www.thaieasyelec.com>
- [6] PubNub. (2026). What is Ladder Logic used for? [Online]. Retrieved March 3, 2026, from . <https://www.pubnub.com/learn/glossary/ladder-logic>
- [7] Shanghai StarFive Semiconductor Co., Ltd. (2025). OpenPLC User Guide for VisionFive 2 and VisionFive 2 Lite, Version 1.2.
- [8] Curtis Cohenour. (2018). An Arduino-based programmable logic control (PLC) lab activity for undergraduate Engineering and Technology (ETM) students. American Society for Engineering Education.