

การค้นหาวัตถุด้วยกล้อง Smart Camera ในระนาบแบบ 2 มิติบนเครื่อง
CNC และจำแนกไปยังพื้นที่จัดเก็บแบบอัตโนมัติด้วยเครื่องควบคุม PLC และ
รายงานตำแหน่งวัตถุด้วยจอ HMI

AUTOMATIC OBJECT SEARCHING WITH A SMART CAMERA 2D
PLANE by CNC M/C, AUTOMATIC CLASSIFICATION TO STORAGE
by PLC AND REPORT OBJECT POSITION WITH HMI MONITOR



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การค้นหาวัตถุด้วยกล้อง Smart Camera ในระนาบแบบ 2 มิติบนเครื่อง
CNC และจำแนกไปยังพื้นที่จัดเก็บแบบอัตโนมัติด้วยเครื่องควบคุม PLC และ
รายงานตำแหน่งวัตถุด้วยจอ HMI

AUTOMATIC OBJECT SEARCHING WITH A SMART CAMERA 2D
PLANE by CNC M/C, AUTOMATIC CLASSIFICATION TO STORAGE
by PLC AND REPORT OBJECT POSITION WITH HMI MONITOR



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AUTOMATIC OBJECT SEARCHING WITH A SMART CAMERA 2D
PLANE by CNC M/C, AUTOMATIC CLASSIFICATION TO STORAGE
by PLC AND REPORT OBJECT POSITION WITH HMI MONITOR



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2564
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาานิพนธ์

.....

หัวข้อปริญญาานิพนธ์ การค้นหาวัตถุด้วยกล้อง Smart Camera ในระนาบแบบ 2 มิติบนเครื่อง
CNC และจำแนกไปยังพื้นที่จัดเก็บแบบอัตโนมัติด้วยเครื่องควบคุม PLC
และรายงานตำแหน่งวัตถุด้วยจอ HMI
AUTOMATIC OBJECT SEARCHING WITH A SMART CAMERA 2D
PLANE by CNC M/C, AUTOMATIC CLASSIFICATION TO STORAGE
by PLC AND REPORT OBJECT POSITION WITH HMI MONITOR

นักศึกษาผู้จัดทำ พิชญุตม์ ดันชกกิจ รหัสนักศึกษา 61010742
 พัฒน์ศก ศรีสกุล รหัสนักศึกษา 61010720
 อธิวิน ถาวรวงษ์ รหัสนักศึกษา 61011222

ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมการวัดคุม
ปีการศึกษา 2564

อาจารย์ผู้ควบคุมปริญญาานิพนธ์	ลายมือชื่อ
รศ.ดร.สุพรรณ กุลพานิชย์	
รศ.ดร.ทวีพล ชื้อสัตย์	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ การค้นหาวัตถุด้วยกล้อง Smart Camera ในระนาบแบบ 2 มิติบนเครื่อง CNC และจำแนกไปยังพื้นที่จัดเก็บแบบอัตโนมัติด้วยเครื่องควบคุม PLC และรายงานตำแหน่งวัตถุด้วยจอ HMI

AUTOMATIC OBJECT SEARCHING WITH A SMART CAMERA 2D PLANE by CNC M/C, AUTOMATIC CLASSIFICATION TO STORAGE by PLC AND REPORT OBJECT POSITION WITH HMI MONITOR

นักศึกษาผู้จัดทำ	พิชญุตม์	ต้นชกกิจ	รหัสนักศึกษา	61010742
	พัฒนัสก	ศรีสกุล	รหัสนักศึกษา	61010720
	อชวิน	ถาวรวงษ์	รหัสนักศึกษา	61011222
ปริญญา	วิศวกรรมศาสตรบัณฑิต			
สาขาวิชา	วิศวกรรมการวัดคุม			
ปีการศึกษา	2564			

บทคัดย่อ

งานวิจัยฉบับนี้เป็นการต่อยอดจากงานวิจัย เรื่องการแสกนวัตถุด้วยกล้อง smart camera ในระนาบแบบ 2 มิติ บนเครื่อง x – y machine ซึ่งแต่เดิมเป็นการควบคุมด้วยมือแบบตำแหน่งต่อตำแหน่งจึงได้พัฒนาโครงการให้เป็นการทำงานอัตโนมัติทั้งหมด โดยเริ่มต้นจากกล้อง pixy ติดตั้งที่มุมสูงเพื่ออ่านตำแหน่งของวัตถุทั้งหมดที่อยู่บนพื้นที่ที่กำหนด เพื่อส่งข้อมูลให้กับเครื่องควบคุม PLC เพื่อให้เครื่องควบคุม PLC ทำการควบคุม Servo motor ให้เคลื่อนที่พากล้อง Smart camera ไปยังตำแหน่งที่กล้อง pixy อ่านได้ หลังจากนั้นกล้อง smart camera จะทำการวิเคราะห์เปรียบเทียบกับรูปทรงเพื่อคัดแยกและนำไปจัดเก็บ ในพื้นที่ที่จัดเตรียมไว้ และไปวิเคราะห์วัตถุชิ้นต่อไปที่อยู่บนพื้นที่ที่เหลืออยู่

โดยจุดเด่นของงานวิจัยนี้ คือ โปรแกรมจะทำการคำนวณจัดลำดับการเคลื่อนที่ไปยังวัตถุที่อยู่ใกล้ที่สุดจากจุดที่อ้างอิงไว้ รวมถึงสามารถทำการวิเคราะห์เพื่อคัดแยกวัตถุ ได้ 2 ลักษณะ คือคัดแยกสีของวัตถุ และคัดแยกรูปทรงทางเรขาคณิตที่มีลักษณะเหมือนกัน เพื่อนำวัตถุเหล่านั้นไปเก็บยังพื้นที่ที่จัดเตรียมไว้ โดยมีความรวดเร็วและความแม่นยำในการจัดเก็บวัตถุมากกว่า 1 ใน 4 ของการควบคุมแบบมือ อีกทั้งยังสามารถควบคุมการทำงาน และแสดงผลผ่านหน้าจอ HMI ได้ จึงทำให้ง่ายต่อการใช้งาน

Thesis Title	AUTOMATIC OBJECT SEARCHING WITH A SMART CAMERA 2D PLANE by CNC M/C, AUTOMATIC CLASSIFICATION TO STORAGE by PLC AND REPORT OBJECT POSITION WITH HMI MONITOR	
Authors	Mr. Pichayut Tantakij	
	Mr. Patsok Srisakul	
	Mr. Atchawin Thawornwong	
Thesis Advisor	Assoc. Prof. Dr. Suphan Gulpanich	
	Assoc. Prof. Dr. Taweepol Suesut	
Year	2021	

Abstract

This research is a continuation of research. From the subject of scanning objects with a smart camera in a 2D plane on the x-y machine, which was originally a position-by-position manual control, the project has been developed to be fully automated. It starts with a pixy camera mounted at an elevated angle to read the position of all objects in a given area. to send data to the PLC controller. In order for the PLC controller to control the servo motor to move the smart camera to a position that the pixy camera can read the smart camera then analyzes the contours for separation and storage in the area provided and then analyzes the next object in the remaining area.

The highlight of this research is that the program will calculate the sequence of motion to the nearest object from the reference point. As well as being able to analyze to separate objects into 2 types, first separating object color, and then sorting out geometric shapes that look the same to bring those objects to the designated area with the speed and accuracy of object collection more than one-fourth that of manual control. You can also control the work and display through the HMI screen, so it is easy to use.

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลุล่วงลงได้ด้วยดีเนื่องจากได้รับความช่วยเหลือจากหลายฝ่าย โดยเฉพาะอย่างยิ่งทางคณะผู้จัดทำขอขอบพระคุณ รศ.ดร.สุพรรณ กุลพานิชย์ ที่ให้ความอนุเคราะห์และความกรุณาในการรับหน้าที่เป็นอาจารย์ที่ปรึกษา อีกทั้งยังมอบความรู้ คำแนะนำ ให้คำปรึกษาในการแก้ไขและปรับปรุงปริญญานิพนธ์ให้มีความสมบูรณ์และมีคุณภาพมากยิ่งขึ้น ตลอดจนอำนวยความสะดวกในด้านอุปกรณ์และสถานที่ และขอขอบพระคุณ รศ.ดร.ทวีพล ชี้อัสต์ย์ ที่ช่วยอำนวยความสะดวกในส่วนของกล้อง Smart Camera, ตัวโปรแกรมและวิธีการเขียนโปรแกรม NI Vision Builder และให้คำแนะนำในระหว่างการดำเนินงาน

ทางคณะผู้จัดทำขอขอบพระคุณคณะกรรมการสอบปริญญานิพนธ์ทุกท่าน ที่ได้ให้คำแนะนำและข้อแก้ไขเพิ่มเติมตลอดภาคการศึกษาที่ผ่านมา ขอขอบคุณคณะวิศวกรรมศาสตร์ และภาควิชาวิศวกรรมการวัดและควบคุม ที่อำนวยความสะดวกในด้านสถานที่และระเบียบขั้นตอนการดำเนินงานต่างๆ

สุดท้ายนี้ทางคณะผู้จัดทำต้องขอขอบคุณ ผู้เขียนเอกสารอ้างอิงและเว็บไซต์ต่างๆ ที่คณะผู้จัดทำนำมาใช้ประกอบการศึกษา และอ้างอิง รวมทั้งผู้มีส่วนร่วมในการจัดทำปริญญานิพนธ์ทุกท่านที่ให้ความช่วยเหลือแก่คณะผู้จัดทำไว้ ณ ที่นี้ด้วย

คณะผู้จัดทำ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....	3
2.1 Programmable Logic Controller (PLC).....	3
2.2 Human Machine Interface (HMI).....	5
2.3 กล้อง Pixy.....	5
2.4 PixyMon.....	6
2.5 Arduino UNO.....	7
2.6 Arduino IDE.....	8
2.7 Servo Motor.....	9
2.8 Machine Vision.....	11
2.9 การสื่อสารผ่านพอร์ตอนุกรม RS-232.....	14
2.10 การสื่อสาร Omron Host Link Protocol.....	17
2.11 NI Vision Builder AI.....	18
2.12 Image Processing.....	19
2.13 OPC Server.....	20
2.14 อัลกอริทึม K-Nearest Neighbors.....	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.15 การจัดเรียงข้อมูล.....	22
2.16 Solenoid Valve.....	22
2.17 Vacuum Ejector.....	23
2.18 การจัดเรียงข้อมูล.....	23
บทที่ 3 วิธีดำเนินงาน.....	24
3.1 อุปกรณ์.....	24
3.2 ซอฟต์แวร์.....	24
3.3 แผนผังการดำเนินงาน.....	25
3.4 วิธีการดำเนินงาน.....	25
3.4.1 การเคลื่อนที่.....	25
3.4.2 การวิเคราะห์และจำแนก.....	31
3.4.3 การออกแบบหน้าจอควบคุมและแสดงผลผ่าน HMI.....	36
3.4.4 การนำวัตถุไปจัดเก็บ.....	39
3.4.5 Function Box.....	49
บทที่ 4 ผลการทดลอง.....	55
4.1 ผลการทดลองเกี่ยวกับการหาตำแหน่งและสีของวัตถุ.....	55
4.1.1 การหาตำแหน่ง.....	55
4.1.2 การตรวจจับสีของวัตถุ.....	55
4.2 ผลการทดลองเกี่ยวกับการเคลื่อนที่.....	56
4.2.1 การควบคุมโดยใช้ Joy Stick.....	56
4.2.2 การควบคุมแบบอัตโนมัติ.....	57
4.3 ผลการทดลองเกี่ยวกับระบบการวิเคราะห์และจำแนกวัตถุด้วยรูปร่าง.....	57
4.4 ผลการทดลองเกี่ยวกับการจัดเก็บวัตถุ.....	60
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ.....	65
5.1 สรุปผลการทดลอง.....	65
5.2 ปัญหาที่พบในระหว่างดำเนินงาน.....	66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
5.3 แนวทางในการพัฒนา.....	66
บรรณานุกรม.....	67
ภาคผนวก ก.....	69



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
4.1 ตารางแสดงผลการทดสอบความถูกต้องในการวิเคราะห์.....56 และจำแนกวัตถุด้วยสี	
4.2 ตารางแสดงผลการทดสอบความถูกต้องในการวิเคราะห์..... 59 และจำแนกวัตถุด้วยรูปร่าง	
4.3 แสดงผลการทดลองเกี่ยวกับการจัดเก็บวัตถุ..... 60	
4.4 ตารางเปรียบเทียบเวลาที่ใช้ในการเคลื่อนที่ของโหมด Manual และโหมด Auto.....61	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 ส่วนประกอบของ PLC.....	4
2.2 PLC CP1H – XA ของบริษัท Omron.....	4
2.3 หน้าจอ Omron HMI NB7W-TW00B.....	5
2.4 กล้อง Pixy.....	6
2.5 กล้อง Pixy2.....	6
2.6 โปรแกรม PixyMon.....	7
2.7 แสดงบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO.....	8
2.8 โปรแกรม Arduino IDE.....	9
2.9 แสดงลักษณะของ Servo Motor.....	10
2.10 แสดงลักษณะของ Servo Drive.....	10
2.11 แสดงลักษณะกล้อง NI Smart Camera ISC-1732.....	12
2.12 แสดงลักษณะ NI Smart Camera I/O Accessory.....	13
2.13 แสดงส่วนประกอบ NI Smart Camera.....	13
2.14 แสดงตัวอย่างการสื่อสารแบบ RS-232.....	14
2.15 แสดงคุณสมบัติตำแหน่งขาสัญญาณต่างๆของ RS-232.....	15
2.16 ระดับสัญญาณระหว่าง RS-232 และ TTL.....	16
2.17 แสดงการเชื่อมต่อไอซี MAX232.....	16
2.18 แสดงรูปแบบการสื่อสารของ Omron Host link Protocol.....	17
2.19 แสดงการเขียน state diagram.....	18
2.20 หน้าต่างโปรแกรม NI Vision Builder AI.....	18
2.21 แสดงการใช้งานเงื่อนไข Transition.....	19
2.22 แสดงการเชื่อมต่อของ OPC Server.....	20
2.23 การพิจารณา K-Nearest Neighbors.....	21
2.24 การพิจารณา K-Nearest Neighbors (ต่อ).....	21
2.25 การพิจารณา K-Nearest Neighbors (ต่อ).....	21
2.26 Solenoid Valve.....	22
2.27 Vacuum Ejector.....	23
3.1 แผนผังแสดงการทำงาน.....	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.2 Function Block สำหรับการควบคุมการเคลื่อนที่ด้วย Joy stick.....	26
3.3 Structured Text สำหรับการเคลื่อนที่โดยใช้ Joy stick.....	26
3.4 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่โดยใช้ Joy Stick ในแนวแกน X.....	27
3.5 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่โดยใช้ Joy Stick ในแนวแกน Y.....	27
3.6 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่โดยใช้ Joy Stick ในแนวแกน Z.....	27
3.7 Function Block สำหรับการคำนวณค่า Pulse ในแกน X และ Y ให้ไปในตำแหน่ง.....	28
ที่เราต้องการ	
3.8 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่ในแกน X และ Y แบบ Auto.....	28
3.9 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการกำหนดค่าตัวแปรต่างๆ.....	29
3.10 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการจัดเรียงตำแหน่งของวัตถุ.....	30
3.11 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของฟังก์ชันการส่งข้อมูลตำแหน่งไปยัง.....	30
PLC	
3.12 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของฟังก์ชันการส่งข้อมูลตำแหน่งไปยัง.....	31
PLC (ต่อ)	
3.13 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการเรียกใช้ฟังก์ชันการรับ - ส่งค่า.....	31
3.14 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการรับค่าสีจากกล้อง Pixy2 camera.....	32
3.15 หน้าต่างโปรแกรม PixyMon V2 ในขณะจำแนกสีของวัตถุ.....	32
3.16 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของฟังก์ชันการส่งข้อมูลสีไปยังPLC.....	33
3.17 โปรแกรม NI Vision Builder AI.....	33
3.18 ตรวจสอบวัตถุชนิดวงกลม.....	34
3.19 ตรวจสอบวัตถุชนิดสี่เหลี่ยมจัตุรัส.....	34
3.20 ตรวจสอบวัตถุชนิดสี่เหลี่ยมข้าวหลามตัด.....	34
3.21 ตรวจสอบวัตถุชนิดแปดเหลี่ยม.....	34
3.22 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดวงกลมจะส่งเป็น 0 0.....	35
3.23 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดสี่เหลี่ยมจัตุรัสจะส่งเป็น 0 1.....	35
3.24 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดสี่เหลี่ยมข้าวหลามตัดจะส่งเป็น 1 0.....	35
3.25 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดแปดเหลี่ยมจะส่งเป็น 1 1.....	36
3.26 หน้าต่างใน HMI Screen.....	36

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.27 หน้าต่างในโหมดแมนวอล.....	37
3.28 หน้าต่างในโหมดอัตโนมัติ.....	37
3.29 หน้าต่างในโหมดการจำแนกวัตถุด้วยสี.....	38
3.30 หน้าต่างในโหมดการจำแนกวัตถุด้วยรูปร่าง.....	38
3.31 Ladder ที่ใช้ในการรับค่าจาก Arduino.....	39
3.32 Function Box ที่ใช้ในการรับค่าจำนวนของวัตถุบนระนาบ.....	39
3.33 Function ที่ใช้ในการ Set stack.....	40
3.34 Function ที่ใช้ในการ PUSH ค่า.....	40
3.35 Ladder ที่ใช้ในการรับตำแหน่งของวัตถุ.....	41
3.36 Function First In First Out.....	41
3.37 Function Block Pulse to object.....	42
3.38 Ladder ในการเคลื่อนที่ของแกน z.....	43
3.39 (ก) Function Block ของการแยกวัตถุด้วยสี.....	44
3.39 (ข) Function Box ของการแยกวัตถุด้วยรูปร่าง.....	44
3.40 Ladder ของการเก็บค่า.....	45
3.41 Ladder ของการเริ่มระบบการทำงานใหม่.....	45
3.42 Function Box ของการเคลื่อนที่ของ servo motor.....	46
3.43 Function Box ของการคำนวณความเร็วของการเคลื่อนที่.....	46
3.44 Function Box ของการคำนวณความเร็วของการเคลื่อนที่ (ต่อ).....	47
3.45 Function Box ของการคำนวณความเร็วของการเคลื่อนที่ (ต่อ).....	48
3.46 Function Box number และ keep.....	49
3.47 Function Box Pulse_To_Object.....	49
3.48 Function Box Pulse_To_Object (ต่อ).....	50
3.49 Function Box compareshape.....	50
3.50 Function Box Find_Obj_autoZ.....	51
3.51 Function Box Color_compare.....	51
3.52 Function Box Color_compare (ต่อ).....	52
3.53 Function Box Find_Obj_auto2.....	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.54 Function Box Find_Obj_auto2 (ต่อ).....	53
3.55 Function Box Scale.....	53
3.56 Function Box Scale (ต่อ).....	54
4.1 ผลการตรวจจับตำแหน่งของวัตถุ.....	55
4.2 ผลการตรวจจับสีของวัตถุ.....	55
4.3 แสดงผลการควบคุมโดยใช้ Joy Stick.....	56
4.4 แสดงผลการควบคุมแบบอัตโนมัติ.....	57
4.5 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และ แสดงผลเป็นวงกลม.....	57
4.6 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และ แสดงผลเป็นแปดเหลี่ยม.....	58
4.7 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และ แสดงผลเป็นสี่เหลี่ยมจัตุรัส.....	58
4.8 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และ แสดงผลเป็นสี่เหลี่ยมข้าวหลามตัด.....	59
4.9 กราฟแสดงความแม่นยำของหัวจับวัตถุ และขนาดของหัวจับวัตถุ.....	60
4.10 กราฟการเปรียบเทียบเวลาการเคลื่อนที่ในตำแหน่งที่ 1.....	62
4.11 กราฟการเปรียบเทียบเวลาการเคลื่อนที่ในตำแหน่งที่ 2.....	62
4.12 กราฟการเปรียบเทียบเวลาการเคลื่อนที่ในตำแหน่งที่ 3.....	63
4.13 กราฟการเปรียบเทียบเวลาการเคลื่อนที่เก็บวัตถุ 3 ตำแหน่ง.....	63
4.14 กราฟการเคลื่อนที่เก็บวัตถุที่ตำแหน่งใดๆ 5 ชั้น ในโหมดการคัดแยกสีเปรียบเทียบกับ..... เวลา	64
4.15 กราฟการเคลื่อนที่เก็บวัตถุที่ตำแหน่งใดๆ 5 ชั้น ในโหมดการคัดแยกรูปร่างของวัตถุ..... เปรียบเทียบกับเวลา	64

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

ปัจจุบันอุตสาหกรรมการผลิตมีจุดมุ่งหมายที่จะเพิ่มคุณภาพ ปริมาณของการผลิตสินค้าและลดต้นทุนการผลิตโดยมีการทำให้เป็นระบบอัตโนมัติมากยิ่งขึ้น และจากความต้องการดังกล่าวส่งผลให้เทคโนโลยีการประมวลผลภาพ (Image Processing) เริ่มเข้ามามีบทบาทในอุตสาหกรรมต่างๆ ในการตรวจสอบความถูกต้อง หรือการนับจำนวนชิ้นงานมากยิ่งขึ้น จึงเกิดแนวคิดในการนำเครื่อง CNC 2 ระนาบ กับกล้อง Pixy2 และกล้อง Smart Camera ที่ใช้ในการประมวลผลภาพ มาประยุกต์ใช้งาน ร่วมกับการเขียนโปรแกรมควบคุม PLC เพื่อพัฒนาเทคโนโลยีที่ก่อให้เกิดประโยชน์ในทาง อุตสาหกรรมต่างๆ ในหลากหลายรูปแบบ เช่น อุตสาหกรรมการผลิตชิ้นส่วนรถยนต์ อุตสาหกรรมการผลิตและอุตสาหกรรมอื่นๆ ที่ต้องมีการตรวจสอบคุณภาพการผลิต เป็นต้น ผู้จัดทำโครงการจึงนำแนวคิดดังกล่าวมาพัฒนาต่อยอดโดยมีจุดมุ่งหมายเพื่อทำให้การเก็บข้อมูลและการตรวจสอบวัตถุ สามารถทำงานได้อย่างอัตโนมัติ อีกทั้งยังสามารถลดต้นทุน และระยะเวลาที่ใช้ในการตรวจสอบ ชิ้นงาน และควบคุมการทำงานและแสดงผลผ่านจอ HMI ซึ่งทำให้ควบคุมการทำงานมีประสิทธิภาพมากยิ่งขึ้น และจะส่งผลให้ประสิทธิภาพในการตรวจสอบคุณภาพการผลิตเพิ่มสูงขึ้น และสอดคล้อง กับความต้องการของอุตสาหกรรมในปัจจุบัน

โครงการวิจัยฉบับนี้เป็นการต่อยอดจากงานวิจัยเรื่องการแสกนวัตถุด้วยกล้อง smart camera ในระนาบแบบ 2 มิติ บนเครื่อง x - y machine ซึ่งแต่เดิมเป็นการตรวจจับวัตถุด้วยการ ควบคุมด้วยมือ (แบบ Manual) ดังนั้นผู้จัดทำจึงพัฒนาให้สมบูรณ์ยิ่งขึ้นโดยขยายขอบเขตงานให้มี การตรวจจับแบบอัตโนมัติโดยใช้ Smart Camera ในการจำแนกวัตถุด้วยรูปร่าง และใช้กล้อง Pixy2 ในการจำแนกวัตถุด้วยสี โดยเครื่องควบคุม PLC Omron CP1H ควบคุมการเคลื่อนที่ของ Servo motor ที่มีกล้องติดตั้งเพื่อใช้ในการจำแนกชนิดของวัตถุ จากนั้นจะทำการเคลื่อนย้ายวัตถุนั้นๆ ไป จัดเก็บยังตำแหน่งที่ตั้งของวัตถุนั้นๆ โดยมีการแสดงผลผ่านทางจอ HMI

1.2 วัตถุประสงค์ของโครงการ

เพื่อพัฒนาต่อยอดระบบตรวจจับและจำแนกวัตถุโดยใช้โดยใช้ Smart Camera และกล้อง Pixy2 ในการจำแนกวัตถุด้วยสีและรูปร่าง ตามลำดับ โดยมีตัวควบคุม PLC ในการควบคุม servo motor ไปยังตำแหน่งของวัตถุนั้นๆ รวมถึงใช้ในการนำวัตถุไปจัดเก็บยังตำแหน่งที่กำหนดไว้ พร้อมทั้ง ควบคุมและแสดงผลผ่านหน้าจอ HMI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงการ

งานวิจัยฉบับนี้เป็นการต่อยอดงานวิจัยที่ผ่านมาซึ่งแต่เดิมการทำงานเป็นแบบ Manual และในรุ่นต่อมาเพื่อให้โครงการสมบูรณ์ยิ่งขึ้นจึงขยายขอบเขตงานให้มีการทำงานเป็นแบบอัตโนมัติ โดยในโหมดการทำงานแบบอัตโนมัติจะแบ่งการจำแนกออกเป็น 2 แบบ ได้แก่ แบบจำแนกด้วยสีวัตถุ และจำแนกด้วยรูปร่างวัตถุ ทั้งนี้ในการจำแนกด้วยสีและรูปร่างจะใช้โปรแกรม PixyMon และ NI Vision Builder AI เข้ามาใช้งานร่วมด้วยตามลำดับ โดยขอบเขตของงานสามารถแบ่งได้ดังนี้

สถานที่ปฏิบัติงาน : ห้อง SCADA LABORATORY ตึกภาควิชาวิศวกรรมการวัดและควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ระยะเวลาในการปฏิบัติงาน : 1 กันยายน 2564 – 10 พฤษภาคม 2565

ขอบเขตในการจัดทำโครงการ

1. การจำแนกวัตถุ
 - สามารถจำแนกสีของวัตถุด้วยกล้อง Pixy2 และสามารถจำแนกรูปร่างของวัตถุด้วย Smart Camera
2. การจัดเก็บและจดจำวัตถุลงบนฐานข้อมูล
 - การควบคุมหัวจับวัตถุและกล้อง Smart Camera ไปยังตำแหน่งของวัตถุ โดยมีกล้อง Pixy2 ที่ทำงานร่วมกับบอร์ด Arduino UNO R3 คอยระบุแหล่งวัตถุให้กับเครื่องควบคุม PLC Omron CP1H
 - จัดเก็บข้อมูลวัตถุด้วยกล้อง Smart Camera ที่ติดตั้งในแนวแกน Z โดยสแกนวัตถุที่วางบนระนาบ X-Y ของเครื่อง CNC เพื่อจดจำลักษณะของวัตถุและเก็บไว้เป็นฐานข้อมูลบนเครื่องคอมพิวเตอร์ ด้วยโปรแกรม NI Vision Builder AI
 - จัดเก็บข้อมูลตำแหน่ง สี และรูปร่างของวัตถุลงใน DATA Memory ถูกกำหนดบนเครื่องควบคุม PLC
3. สามารถควบคุมและแสดงผลข้อมูลของวัตถุที่ถูกจัดเก็บบนหน้าจอ HMI

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถใช้โปรแกรม NI Vision Builder และกล้อง Smart Camera ในการจำแนกรูปร่างของวัตถุโดยใช้กล้อง Pixy2 Camera ในการระบุตำแหน่งและสีของวัตถุได้
2. สามารถบันทึกข้อมูลตำแหน่ง และชนิดวัตถุ รวมทั้งค่าการเคลื่อนที่ไปยังวัตถุดังกล่าว ลงในฐานข้อมูล และจัดทำออกมาเป็นรายงานได้
3. สามารถพัฒนาการเคลื่อนที่ของเครื่อง CNC 2 ระนาบ ให้สามารถเคลื่อนที่หัวจับวัตถุและกล้อง Smart Camera ไปยังตำแหน่งที่มีวัตถุได้ดียิ่งขึ้น
4. สามารถจำแนกและจัดเก็บวัตถุที่ตรวจจับไปไว้ยังตำแหน่งที่ถูกต้องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 Programmable Logic Controller (PLC)

2.1.1 Programmable Logic Controller คืออะไร?

โปรแกรมเมเบิลลอจิกคอนโทรลเลอร์ (Programmable Logic Control : PLC) เป็นอุปกรณ์ควบคุมการทำงานของเครื่องจักรหรือกระบวนการทำงานต่างๆ โดยภายในมี Microprocessor เป็นมันสมองสิ่งสำคัญ PLC จะมีส่วนที่เป็นอินพุตและเอาต์พุตที่สามารถต่อออกไปใช้งานได้ทันที ตัวตรวจวัดหรือสวิตช์ต่างๆ จะต่อเข้ากับอินพุต ส่วนเอาต์พุตจะใช้ต่อออกไปควบคุมการทำงานของอุปกรณ์หรือเครื่องจักรที่เป็นเป้าหมาย เราสามารถสร้างวงจรหรือแบบของการควบคุมได้โดยการป้อนเป็นโปรแกรมคำสั่งเข้าไปใน PLC นอกจากนี้ยังสามารถใช้งานร่วมกับอุปกรณ์อื่นเช่นเครื่องอ่านบาร์โค้ด (Barcode Reader) เครื่องพิมพ์ (Printer) ซึ่งในปัจจุบันนอกจากเครื่อง PLC จะใช้งานแบบเดี่ยว (Stand alone) แล้วยังสามารถต่อ PLC หลายๆ ตัวเข้าด้วยกัน (Network) เพื่อควบคุมการทำงานของระบบให้มีประสิทธิภาพมากยิ่งขึ้น

PLC เป็นอุปกรณ์คอมพิวเตอร์สำหรับใช้ในงานอุตสาหกรรม PLC ประกอบด้วยหน่วยประมวลผลกลาง หน่วยความจำ หน่วยรับข้อมูล หน่วยส่งข้อมูล และหน่วยป้อนโปรแกรม PLC ขนาดเล็กส่วนประกอบทั้งหมดของ PLC จะรวมกันเป็นเครื่องเดียว แต่ถ้าเป็นขนาดใหญ่สามารถแยกออกเป็นส่วนประกอบย่อยๆ ได้

หน่วยความจำของ PLC ประกอบด้วย หน่วยความจำชนิด RAM และ ROM หน่วยความจำชนิด RAM ทำหน้าที่เก็บโปรแกรมของผู้ใช้และข้อมูลสำหรับใช้ในการปฏิบัติงานของ PLC ส่วน ROM ทำหน้าที่เก็บโปรแกรมสำหรับใช้ในการปฏิบัติงานของ PLC ตามโปรแกรมของผู้ใช้ ROM ย่อมาจาก Read Only Memory สามารถโปรแกรมได้แต่ลบไม่ได้ ถ้าชำรุดแล้วซ่อมไม่ได้

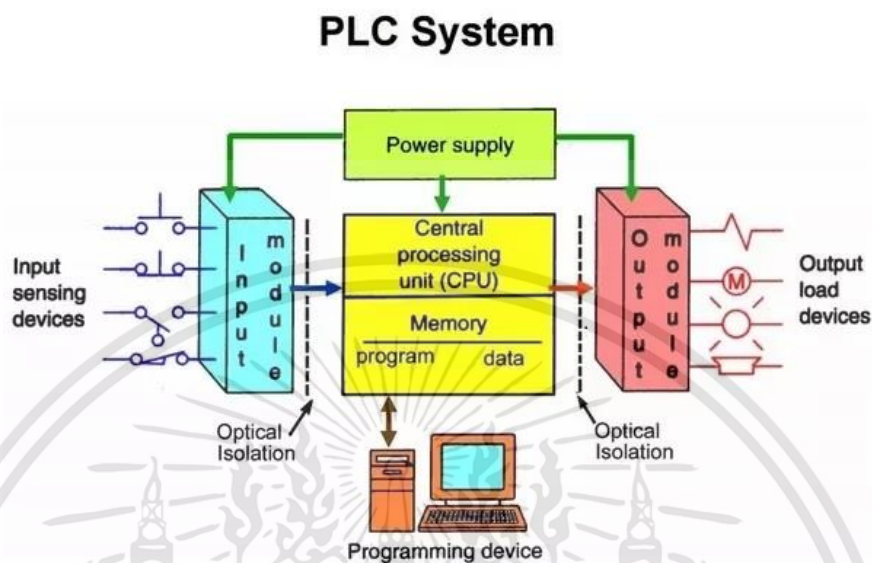
1. RAM (Random Access Memory) หน่วยความจำประเภทนี้จะมีแบตเตอรี่เล็กๆ ต่อไว้ เพื่อใช้เลี้ยงข้อมูลเมื่อเกิดไฟดับ การอ่านและเขียนโปรแกรมลงใน RAM ทำได้ง่ายมาก จึงเหมาะกับการใช้งานในระยะทดลองเครื่องที่มีการเปลี่ยนแปลงแก้ไขโปรแกรมบ่อยๆ

2. EPROM (Erasable Programmable Read Only Memory) หน่วยความจำชนิด EPROM นี้จะต้องใช้เครื่องมือพิเศษในการเขียนโปรแกรม การลบโปรแกรมทำได้โดยใช้แสงอัลตราไวโอเล็ตหรือตากแดดร้อนๆ นานๆ มีข้อดีตรงที่โปรแกรมจะไม่สูญหายแม้ไฟดับ จึงเหมาะกับการใช้งานที่ไม่ต้องเปลี่ยนโปรแกรม

3. EEPROM (Electrical Erasable Programmable Read Only Memory) หน่วยความจำชนิดนี้ไม่ต้องใช้เครื่องมือพิเศษในการเขียนและลบโปรแกรม โดยใช้วิธีการทางไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

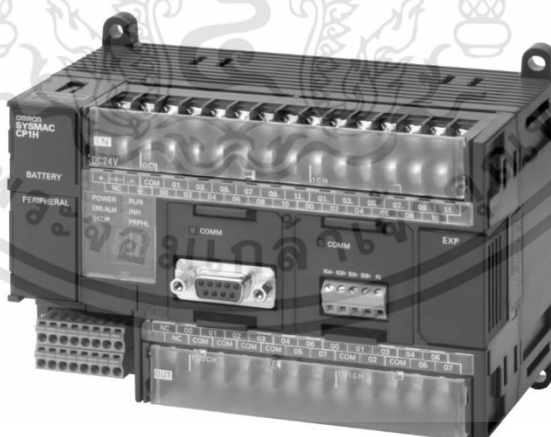
เหมือนกับ RAM นอกจากนั้นก็ไม่จำเป็นต้องมีแบตเตอรี่สำรองไฟเมื่อไฟดับ ราคาจะแพงกว่า แต่จะรวมคุณสมบัติที่ดีของทั้ง RAM และ EPROM เอาไว้ด้วยกัน



รูปที่ 2.1 ส่วนประกอบของ PLC

(<https://qph.fs.quoracdn.net/main-qimg-f6097b941be3f63343f1062a78f760a6.webp>)

2.1.2 Programmable Logic Controller : Omron CP1H – XA



รูปที่ 2.2 PLC CP1H – XA ของบริษัท Omron

โดย PLC ที่เลือกใช้ในโครงการนี้จะมีคุณสมบัติดังตารางในภาคผนวก ก หน้า 70-71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 Human Machine Interface (HMI)

2.3.1 Human Machine Interface (HMI) คืออะไร?

Human Machine Interface (HMI) คือ การใช้งานร่วมกันระหว่าง PLC Programming กับเครื่องคอมพิวเตอร์ จึงเรียกว่า HMI (Human Machine Interface) โดยนำคอมพิวเตอร์มาเป็นอุปกรณ์ที่ใช้ในการติดต่อระหว่างผู้ใช้งานกับเครื่องจักร เพื่อควบคุมและเป็นจอแสดงผล HMI เกิดจากความต้องการของผู้ใช้งานที่ต้องการเข้าไปควบคุมระบบที่ PLC เป็นตัวควบคุมอยู่ โดย HMI นั้น จะเป็นการนำข้อมูลจาก PLC ส่งผ่านโครงข่ายของการสื่อสารแบบต่างๆ และทำการรวบรวมข้อมูลในรูปแบบต่างๆ เข้าด้วยกัน

2.3.2 Human Machine Interface (HMI) : NB7W-TW00B

NB7W-TW00B เป็นหน้าจอ HMI ของบริษัท Omron โดยมีคุณสมบัติ คือ หน้าจอขนาด 7 นิ้ว TFT LCD มีความละเอียด 800 x 480 dots



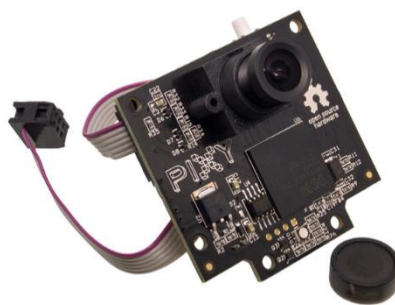
รูปที่ 2.3 หน้าจอ Omron HMI NB7W-TW00B

2.3 กล้อง Pixy

2.3.1 กล้อง Pixy คืออะไร ?

เป็นโมดูลกล้องตรวจจับแยกแยะวัตถุด้วยสีที่โด่งดังและประสบความสำเร็จอย่างสูงในการระดมทุนผ่าน www.kickstarter.com โดยใช้โมดูล CCD ทำงานร่วมกับแผงวงจรประมวลผลภาพหรือ image processing โดย Pixy นับเป็นรุ่นที่ 5 ของโครงการ CMUCAM บางครั้งจึงเรียก Pixy ว่า CMUCAM5 โดยมีความเร็วในการตรวจจับภาพ 50 เฟรมต่อวินาที ตรวจจับสีที่ต้องการได้พร้อมกัน 7 สี และแยกแยะวัตถุได้ 100 ชิ้น สามารถเชื่อมต่อกับไมโครคอนโทรลเลอร์หรือบอร์ดสมองกลฝังตัวผ่านพอร์ต UART, บัส SPI, บัส I2C, พอร์ตดิจิทัล และ พอร์ตอะนาล็อก โดยกล้อง Pixy เป็นโครงการโอเพ่นซอร์สของ Charmlabs แห่งมหาวิทยาลัย Carnegie Mellon (CMU) สหรัฐอเมริกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 กล้อง Pixy

2.3.2 กล้อง Pixy2

Pixy2 CMUcam5 Sensor(EFDV686) เป็นโมดูลกล้องประมวลผลภาพแบบสำเร็จรูปโดยกล้อง Pixy2 CMUcam5 Sensor จะเป็นรุ่นอัปเดตจาก Pixy CMUcam5 Sensor ที่ได้ยกเลิกการผลิตไปแล้ว โดยเพิ่มความเร็วในการตรวจจับภาพเพิ่มเป็น 60 เฟรมต่อวินาที, มีขนาดโดยรวมเล็กลงตัวกล้องถูกออกแบบมาให้ง่ายต่อการใช้งานเหมาะกับบอร์ดไมโครคอนโทรลเลอร์ต่างๆ เช่น บอร์ด Arduino, บอร์ด Raspberry Pi หรือบอร์ดอื่นๆ ตัวกล้องมีขนาดเล็กลง ทำงานได้รวดเร็วขึ้น และมีความฉลาดมากขึ้น นอกจากความสามารถในการตรวจจับสีของวัตถุ ยังเพิ่มอัลกอริทึมในการตรวจจับและติดตามเส้นสำหรับใช้ในหุ่นยนต์เดินตามเส้น รวมทั้งสามารถตรวจจับป้ายเครื่องหมายคำสั่งบนผิวถนนได้อีกด้วย

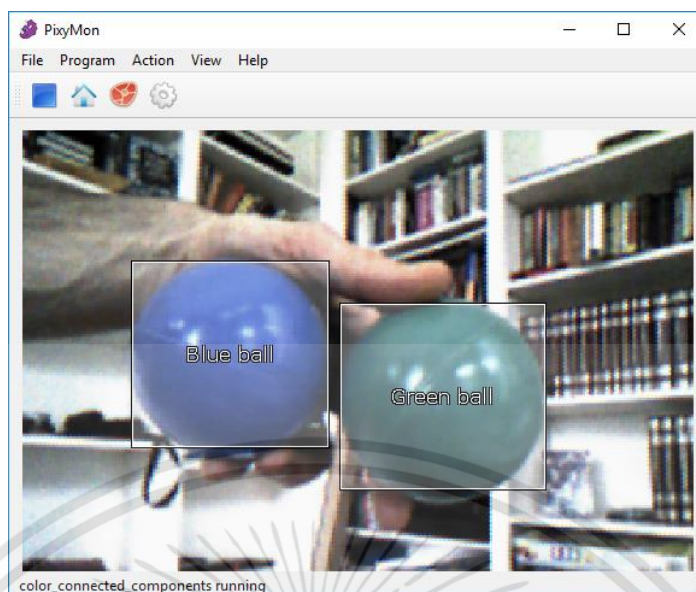


รูปที่ 2.5 กล้อง Pixy2

2.4 PixyMon

โปรแกรม PixyMon สามารถดาวน์โหลดได้เป็นฟรีแวร์ที่สามารถใช้ร่วมกับกล้อง Pixy ในการตั้งค่าการตรวจจับวัตถุได้ตามต้องการโดยกำหนดรูปแบบเป็น ‘Signature’ เช่น การตรวจจับสี, การตรวจจับเส้น, การติดตามตำแหน่งวัตถุ หรือตั้งค่าความสว่าง, ความไวในการตรวจจับวัตถุ, จำนวนวัตถุที่สามารถตรวจจับ นอกจากนี้ยังสามารถตั้งค่าพื้นฐานของกล้อง โดยโปรแกรม PixyMon สามารถใช้ได้ในระบบปฏิบัติการ Windows, Linux และ macOS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 โปรแกรม PixyMon

2.5 Arduino UNO

2.5.1 Arduino คืออะไร ?

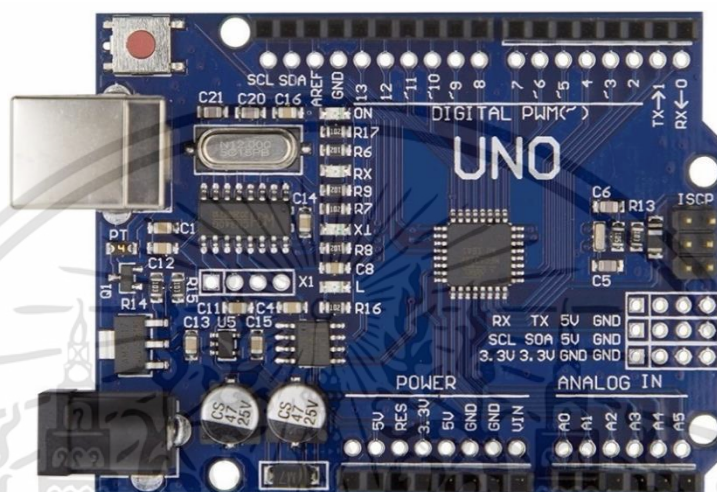
Arduino คือ โครงการที่นำชิปไอซีไมโครคอนโทรลเลอร์ตระกูลต่างๆ มาใช้ร่วมกันในภาษา C ซึ่งภาษา C นี้เป็นลักษณะเฉพาะ คือมีการเขียนไลบรารีของ Arduino ขึ้นมาเพื่อให้การสั่งงานไมโครคอนโทรลเลอร์ที่แตกต่างกัน สามารถใช้งานโค้ดตัวเดียวกันได้ โดยตัวโครงการได้ออกบอร์ดทดลองมาหลายรูปแบบ เพื่อใช้งานกับ IDE ของตนเอง สาเหตุหลักที่ทำให้ Arduino เป็นนิยมมาก เป็นเพราะซอฟต์แวร์ที่ใช้งานร่วมกันสามารถโหลดได้ฟรี และตัวบอร์ดทดลองยังถูกแจกแปลน ทำให้ผู้ผลิตจื๋นนำไปผลิตและขายออกตลาดมาในราคาที่ถูกมากๆ

Arduino นั้นได้ใช้ชิป AVR เป็นหลักใน Arduino แทบรุ่น สาเหตุมาจากไมโครคอนโทรลเลอร์ของตระกูล AVR นั้นมีความทันสมัย ในชิปในบางตัวสามารถเชื่อมต่อผ่าน USB ได้โดยตรง สามารถใช้กับคอมพิวเตอร์สมัยใหม่ได้เป็นอย่างดี และในไมโครคอนโทรลเลอร์ตระกูล AVR ยังมีส่วนของโปรแกรมพิเศษที่เรียกว่า Bootloader อยู่ในระดับต่ำกว่าส่วนโปรแกรมปกติ ซึ่งจะเป็นส่วนโปรแกรมที่จะถูกเรียกขึ้นมาก่อนการเรียกโปรแกรมปกติ ทำให้สามารถเขียนสั่งให้ทำงานใดๆก็ได้ ก่อนการเรียกโปรแกรมปกติ ทำให้ Arduino นั้นอาศัยส่วนโปรแกรมพิเศษนี้ในการทำให้ชิปสามารถโปรแกรมผ่านพอร์ตอนุกรมชนิด UART ได้ จึงทำให้การเขียนโปรแกรมลงไปในชิปใช้เพียง USB to UART ก็เพียงพอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 Arduino UNO

เป็นบอร์ด Arduino รุ่นแรกที่ยังคงมีขนาดประมาณ 68.6x53.4mm เป็นบอร์ดมาตรฐานที่นิยมใช้งานมากที่สุด เนื่องจากเป็นขนาดที่เหมาะสมสำหรับการเริ่มต้นเรียนรู้ Arduino และมี Shields ให้เลือกใช้งานได้มากกว่าบอร์ด Arduino รุ่นอื่นๆ ที่ออกแบบมาเฉพาะมากกว่า โดยบอร์ด Arduino UNO ได้มีการพัฒนาเรื่อยมา ตั้งแต่ R2 R3 และรุ่นย่อยที่เปลี่ยนชิปไอซีเป็นแบบ SMD



รูปที่ 2.7 แสดงบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO

2.6 Arduino IDE

เครื่องมือการเขียนโปรแกรมที่มีใช้งานได้กับ Arduino ได้ทุกรุ่น โดยภายในจะมีเครื่องมือที่จะเป็นสำหรับติดต่อ Arduino เช่น การค้นหา Arduino ที่ติดต่อกับเครื่องคอมพิวเตอร์ การเลือกรุ่น Arduino ที่ต่ออยู่เพื่อนตรวจสอบว่าขนาดของโปรแกรมที่เขียน หรือไลบรารีต่างๆ ซบพอร์ตกับ Arduino รุ่นนั้นๆ ไหม อีกทั้งยังมีโปรแกรมติดต่อผ่านซีเรียลโดยตรงสำหรับคอมพิวเตอร์

Arduino® Integrated Development Environment (IDE) เป็นโปรแกรมที่ออกแบบมาเพื่อให้ง่ายต่อการเขียนซอฟต์แวร์สำหรับแพลตฟอร์มโอเพ่นซอร์สนี้ แพลตฟอร์ม Arduino® เป็นแพลตฟอร์มอิเล็กทรอนิกส์ยอดนิยมที่ออกแบบมาเพื่อลดความซับซ้อนของกระบวนการออกแบบอุปกรณ์อิเล็กทรอนิกส์

```

real | Arduino 1.8.10
File Edit Sketch Tools Help
real
#include <SPI.h>
#include <Pixy2.h>
Pixy2 pixy;
int xpos[10];
int ypos[10];
void fxsent(int , int);
void fxrecieve();
void setup()
{
  Serial.begin(9600, SERIAL_8N1);
  pixy.init();
}

void loop()
{
  pixy.ccc.getBlocks();
  if(pixy.ccc.numBlocks)
  {
    for(int i=0;i<pixy.ccc.numBlocks;i++) //memory po
      ypos[i]=pixy.ccc.blocks[i].m_x; //swap bloc
    for(int i=0;i<pixy.ccc.numBlocks;i++) //swan bloc
      xpos[i]=pixy.ccc.blocks[i].m_y;
  }
}

Could not write preferences file: C:\Users\INGKARAT\AppData\Local\Arduino15
Arduino/Genuino Uno

```

รูปที่ 2.8 โปรแกรม Arduino IDE

2.7 Servo Motor

2.7.1 Servo Motor

Servo ในทางวิศวกรรมเป็นการควบคุมการขับเคลื่อน (servomechanism) คือ อุปกรณ์อิเล็กทรอนิกส์สำหรับควบคุมมอเตอร์หรือระบบขับเคลื่อนต่าง ๆ ด้วยการตรวจสอบสัญญาณจากระบบขับเคลื่อนและปรับค่าให้ถูกต้องตามที่กำหนดไว้อย่างต่อเนื่อง โดยปกติระบบ servo จะตรวจสอบข้อผิดพลาดจากสัญญาณตอบกลับเช่น ตำแหน่งทางกล ความเร็วหรือค่าผิดพลาดอื่น ๆ เช่น ข้อผิดพลาดจากระบบหน้าตาต่างของรถยนต์อาจไม่ได้รวมอยู่ในระบบขับเคลื่อน (none servomechanism)หรือระบบควบคุมเร็วซึ่งใช้การควบคุมแบบปิด (close-loop feedback) ถือว่าอยู่ในระบบขับเคลื่อนได้

ลักษณะของการควบคุมเซอร์โวมอเตอร์จะเป็นระบบควบคุมแบบลูปปิด (Closed loop control) ซึ่งประกอบด้วยโหมดการควบคุม 3 โหมดคือโหมดการควบคุมแรงบิด (Torque Control Mode) ซึ่งอยู่วงรอบหรือลูปสุด โหมดการควบคุมอัตราเร่ง (Velocity Control Mode) และโหมดการควบคุมตำแหน่ง (Position Control Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงลักษณะของ Servo Motor

2.7.2 Servo Drive

Servo Drive คือ อุปกรณ์อุตสาหกรรมที่ได้รับสัญญาณคำสั่งมาจาก Controller หรือ ระบบควบคุมสัญญาณ และส่งกระแสไฟฟ้าเพื่อเชื่อมต่อไปยัง Servo Drive เป็นตัวผ่านการส่ง การไปยัง [Servo Motor](#) ด้วยการใช้โปรแกรมของตัวงานนั้น ๆ ซึ่งจะทำให้การควบคุมการเคลื่อนที่ ความเร็วของการหมุน ระยะทางที่มอเตอร์หมุน และกำลังที่จะใช้หมุน (กรณีที่เครื่องจักรอุตสาหกรรม ต้องใช้กำลังการทำงานสูง) Servo Drive จะรับสัญญาณมาจาก Controller และส่งการให้กับ ตัว Servo Motor เคลื่อนที่ตามที่ Controller สั่งการมา เนื่องจาก Servo Drive จะเป็นตัวที่ปรับตั้ง ค่าของตัว Servo Motor ให้ทำงานตามรูปแบบของการควบคุมไม่ว่า จะเป็นการควบคุม ความเร็ว (Speed Control), แรงบิด (Torque) และ ตำแหน่ง(Position Control) ตัว Servo Drive จะเป็น ตัวกำหนดค่าตัวแปรหรือพารามิเตอร์ต่างๆ ให้กับตัว Servo Motor ให้ทำงานได้อย่างถูกต้องและ แม่นยำ



รูปที่ 2.10 แสดงลักษณะของ Servo Drive

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 Machine Vision

2.8.1 Machine Vision

ระบบแมชชีนวิชัน (Machine Vision) เป็นระบบที่นิยมใช้งานในภาคอุตสาหกรรมเป็นส่วนใหญ่ เนื่องจากสามารถนำระบบนี้มาใช้แทนการตรวจสอบสินค้าด้วยสายตาคน ซึ่งสายตาคนมีข้อจำกัดในเรื่องความเร็วและความถูกต้อง ดังนั้นแมชชีนวิชันจึงเป็นเครื่องมือควบคุมคุณภาพที่มีความน่าเชื่อถือและคุ้มค่าต่อการลงทุนของภาคอุตสาหกรรม โดยมีส่วนประกอบที่สำคัญดังนี้

2.8.1.1 แสงสว่าง (Lighting) เป็นสิ่งจำเป็นที่ทำให้กล้องสามารถเห็นสิ่งที่ต้องการเห็นอย่างชัดเจน ซึ่งปัจจุบันนิยมใช้หลอด LED เพราะแสงจะมีความเสถียรและไม่กระพริบ ซึ่งจะทำให้ซอฟต์แวร์สามารถประมวลผลได้ถูกต้องและแม่นยำขึ้น โดยในแต่ละงานเทคนิคในการใช้แสงจะแตกต่างกันออกไปตามลักษณะงาน เช่น สีของแสง (Color) รูปร่างของแสง (Shape) เป็นต้น อีกทั้งลักษณะการฉายแสงก็มีผลต่อการมองเห็นวัตถุ ได้แก่ Font Light, Back Light, Dome Light, Angle Light เป็นต้น

2.8.1.2 เลนส์ (Lens) มีผลต่อความละเอียดของภาพที่ถ่าย เนื่องจากเลนส์แต่ละอันจะมีค่าความละเอียดที่ต่างกัน

2.8.1.3 กล้อง (Camera) นิยมใช้กล้องแบบดิจิทัล ซึ่งการใช้งานกับระบบแมชชีนวิชันจะใช้กับการถ่ายภาพแบบพื้นที่ (Area scan) เป็นหลัก โดยกล้องจะมีการเชื่อมต่อกับเครื่องคอมพิวเตอร์หลายแบบให้เลือกใช้ ได้แก่

- GigE : เชื่อมต่อกับสายแลน (Lan) โดยมีการตั้งค่า ip address ให้กล้องกับเครื่องคอมพิวเตอร์เชื่อมต่อกันได้และความยาวสูงสุดของการเชื่อมต่อประมาณ 100 เมตร
- USB3: ง่ายและสะดวกกับการเชื่อมต่อคอมพิวเตอร์ และความยาวสูงสุดของการเชื่อมต่อประมาณ 5 เมตร

2.8.1.4 Frame Grabber คือ การ์ดที่ออกแบบมาเพื่อรองรับการเชื่อมต่อกับกล้องโดยเฉพาะ โดยมีทั้งสำหรับกล้อง Camera link, USB3 และ GigE เป็นต้น

2.8.1.5 การ์ดรับข้อมูลเข้าและข้อมูลออก (I/O card) จะเป็นการใช้งานการ์ด DAQ หรือการ์ดสำหรับสื่อสารกับอุปกรณ์ต่างๆ ซึ่งในการใช้งานจริง อาจต้องมีการอ่านค่าจากเซ็นเซอร์ (Sensor) รวมถึงการสั่งงานรีเลย์หรือมอเตอร์ตามผลลัพธ์ที่ได้จากการประมวลผลภาพ

2.8.1.6 ซอฟต์แวร์ (Software) เพื่อใช้ในการสั่งงานกล้องให้ดึงภาพและแสดงภาพ โดยจะใช้โปรแกรม NI-IMAQdx และ แอปพลิเคชัน (Application) ใช้สำหรับจัดการและประมวลผลภาพ ซึ่งเรียกว่า NI Vision Builder for Automated Inspection (VBAI)

2.8.2 Smart Camera

Smart Camera หรือ กล้องอัจฉริยะ เป็นระบบ Vision System ที่สามารถจับภาพ และสามารถนำภาพออกสู่แอปพลิเคชันจากภาพถ่ายพร้อมกับข้อมูลเฉพาะของรูปภาพ ตลอดจนสามารถสร้างการตัดสินใจที่สามารถใช้ในระบบอัจฉริยะและระบบอัตโนมัติ

Smart Camera มีระบบการมองเห็นแบบสแตนด์อโลนพร้อมเซ็นเซอร์รับภาพในตัว และประกอบด้วยอินเทอร์เฟซการสื่อสารที่จำเป็น เช่น Ethernet, I/O ที่จำเป็นต่อการเชื่อมต่อกับ PLC, Actuator, Relay, Pneumatic valves

2.8.3 NI 1732 Smart Camera

Smart Camera ISC-1732 เป็น real time target ที่มีราคาถูก สำหรับ machine vision และขับเคลื่อนโดยโปรเซสเซอร์ความเร็ว 400 MHz PowerPC และมีการผสมผสานระหว่างออนบอร์ดโปรเซสเซอร์กับเซ็นเซอร์รูปภาพ CCD ที่ทำให้ระบบการมองเห็น all in one กระจายได้ง่าย ตัว case ของ ISC-1732 ถูกออกแบบมาเพื่อการใช้งานในอุตสาหกรรม

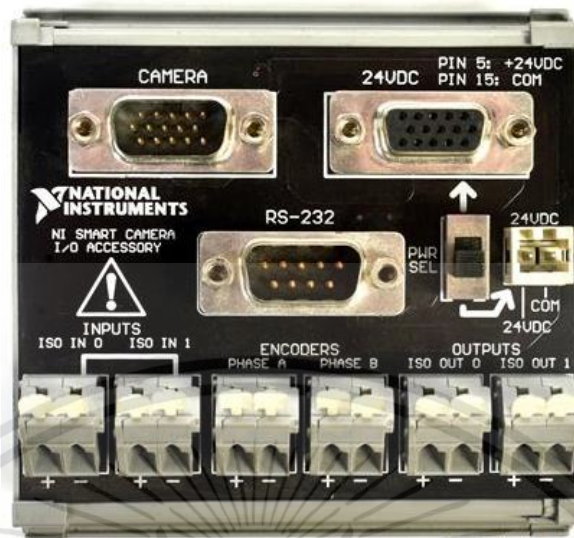
I/O ของกล้องประกอบด้วย 2 ดิจิตอลเอาต์พุตและอินพุต พอร์ต RS-232 และพอร์ต Gigabit Ethernet และสามารถตั้งค่า NI Smart Camera ด้วย ซอฟต์แวร์ Vision Builder for Automated Inspection หรือโปรแกรมให้กับกล้องด้วย LabVIEW Real-Time Module and the Vision Development Module

โดยคุณสมบัติของ NI 1732 Smart Camera สามารถดูได้จากภาคผนวกหน้าที่ 72-78 และคุณสมบัติของ I/O Accessory Board สามารถดูได้จากภาคผนวกหน้าที่ 79-80

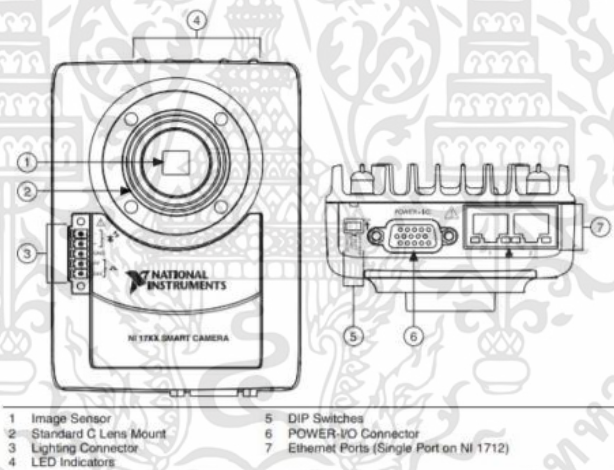


รูปที่ 2.11 แสดงลักษณะกล้อง NI Smart Camera ISC-1732

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แสดงลักษณะ NI Smart Camera I/O Accessory



- 1 Image Sensor
- 2 Standard C Lens Mount
- 3 Lighting Connector
- 4 LED Indicators
- 5 DIP Switches
- 6 POWER-IO Connector
- 7 Ethernet Ports (Single Port on NI 1712)

Table 1-1. NI Smart Camera Models

	Processor	Image Sensor	Lighting Strobe	Direct Drive Lighting Controller	Quadrature Encoder Support
NI 1712	400 MHz PowerPC	1/3 inch Sony ICX424AL CCD Monochrome 640 × 480 pixels (VGA)	No	No	No
NI 1732	400 MHz PowerPC	1/3 inch Sony ICX424AL CCD Monochrome 640 × 480 pixels (VGA)	Yes	No	Yes
NI 1752	600 MHz PowerPC	1/3 inch Sony ICX424AL CCD Monochrome 640 × 480 pixels (VGA)	Yes	Yes	Yes
NI 1754	600 MHz PowerPC	1/2 inch Sony ICX205AL CCD Monochrome 1,280 × 1,024 pixels (SXGA)	Yes	Yes	Yes

รูปที่ 2.13 แสดงส่วนประกอบ NI Smart Camera

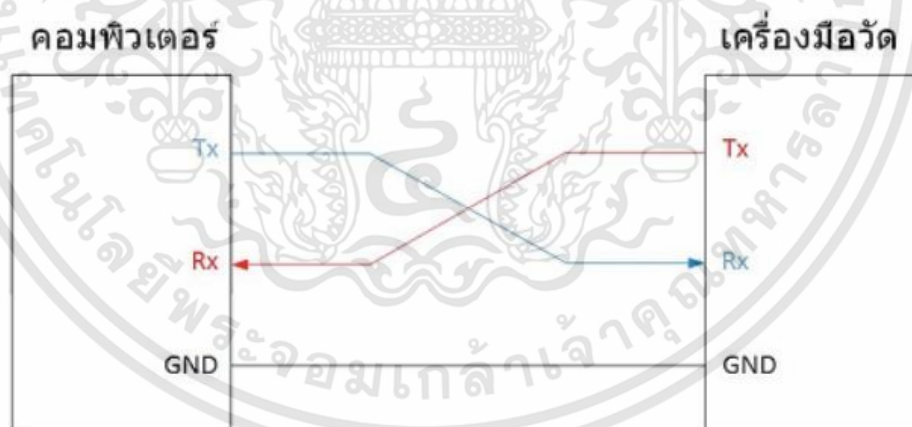
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 การสื่อสารผ่านพอร์ตอนุกรม RS-232

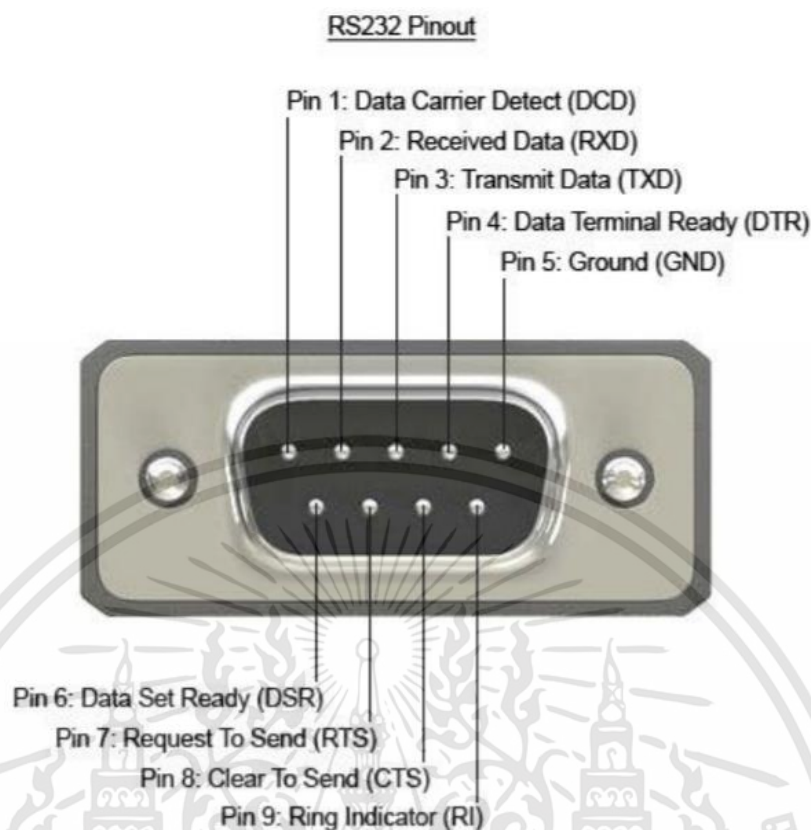
RS232 (Recommended Standard no. 232) คือมาตรฐานการสื่อสารข้อมูลดิจิทัลแบบอนุกรม (serial communication) ซึ่งถูกกำหนดขึ้นครั้งแรกในปี ค.ศ. 1960 โดย EIA (Electronic Industries Association) หรือ สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ของอเมริกา คือการกำหนดการเชื่อมต่อระหว่าง DTE(data terminal equipment) เช่น คอมพิวเตอร์ กับ DCE(data circuit-terminating equipment or data communication equipment) เช่น โมเด็ม เป็นต้น มาตรฐาน RS232 ถูกใช้กันอย่างแพร่หลายในการสื่อสารระดับอุตสาหกรรมแต่ในปัจจุบันได้มี USB ซึ่งเป็นมาตรฐานสื่อสารที่รับ/ส่งข้อมูลได้เร็วกว่าเข้ามาแทนที่

มาตรฐาน RS232 เป็นมาตรฐานที่รับ/ส่งข้อมูลแบบ Full duplex หรือจะให้พูดง่าย ๆ คือสามารถรับและส่งข้อมูลได้พร้อมกันทั้งคู่ในเวลาเดียวกัน โดยการรับ/ส่งข้อมูลนั้นจะใช้สายไฟทั้งหมด 3 เส้น ได้แก่

- Tx (Transmit data) คือ สายส่งข้อมูล ซึ่งสายเส้นนี้จะมีหน้าที่ในการส่งข้อมูลเท่านั้น
- Rx (Receive data) คือ สายรับข้อมูล ซึ่งสายเส้นนี้จะมีหน้าที่ในการรับข้อมูลเท่านั้น
- GND (Signal ground) คือ สายกราวด์ เป็นสายเทียบหรืออ้างอิงแรงดันไฟฟ้า 0V

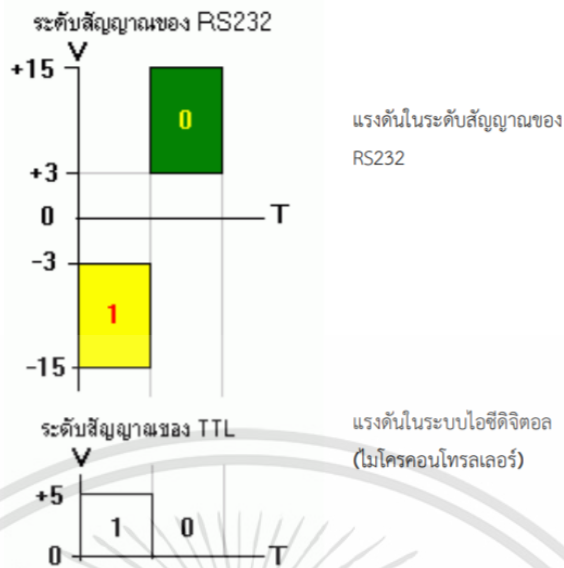


รูปที่ 2.14 แสดงตัวอย่างการสื่อสารแบบ RS-232



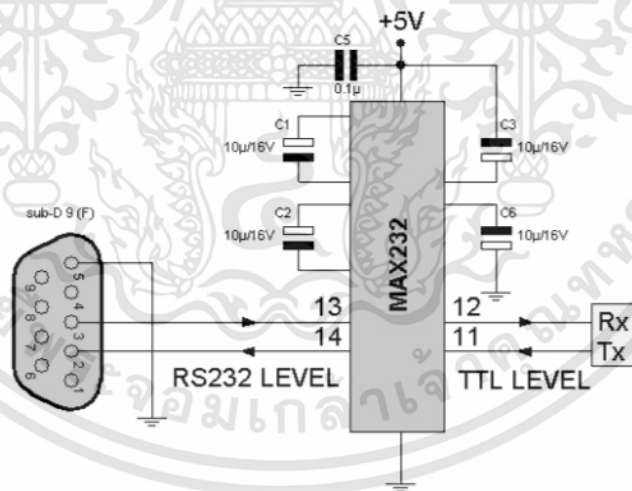
รูปที่ 2.15 แสดงคุณสมบัติตำแหน่งขาสัญญาณต่างๆของ RS-232

การกำหนดมาตรฐานการเชื่อมต่อแบบอนุกรม EIA RS-232 (x) เป็นมาตรฐานอุตสาหกรรม โดยคณะกรรมการสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association) ออกแบบมา เพื่อใช้ในการส่งข้อมูลอนุกรมแบบ อะซิงโครนัส 2 ทิศทาง เพื่อให้มีการใช้งานในการเชื่อมต่อที่สอดคล้องกัน ระหว่างอุปกรณ์คอมพิวเตอร์ต่างๆ การรับส่งสัญญาณจะมีระดับ สัญญาณ ตั้งแต่ 3 โวลต์ จนถึง 15 โวลต์ สำหรับลอจิก "0" และมีระดับแรงดันที่ -3 โวลต์ จนถึง -15 โวลต์ สำหรับลอจิก "1" ดังนั้นสังเกตได้ว่าจะมีระดับแรงดันที่ใช้ในสถานะลอจิก "0" และ ลอจิก "1" แตกต่างออกไปจาก ระบบไอซีดิจิทัลทั่วไป การต่อใช้งานกับวงจรไอซีดิจิทัลจึงต้องมีอุปกรณ์ที่ทำหน้าที่ปรับเปลี่ยนระดับ แรงดันจาก 3 - 15 โวลต์ ให้มีระดับแรงดัน 0 - 5 โวลต์ ในภาคการส่งข้อมูล ส่วนในภาคของการรับข้อมูล จะต้องเปลี่ยนระดับแรงดัน 0 - 5 โวลต์ จากไมโครคอนโทรลเลอร์ให้เป็นระดับแรงดันที่สูงกว่า +3 หรือต่ำกว่า - 3 โดยจะมีไอซีสำเร็จรูปพร้อมใช้งาน หรืออาจจะต่อวงจรจากทรานซิสเตอร์ก็ได้



รูปที่ 2.16 ระดับสัญญาณระหว่าง RS-232 และ TTL

ไอซี MAX232 เป็นไอซีที่มีความสามารถแปลงระดับสัญญาณระหว่าง TTL และ RS232 เพื่อใช้สื่อสารข้อมูลระหว่าง ไมโครคอนโทรลเลอร์ กับ อุปกรณ์ต่างๆเช่น PLC ในโครงงานนี้ได้ใช้ไอซี MAX232 ในการสื่อสารระหว่าง Arduino กับ PLC เช่นกัน



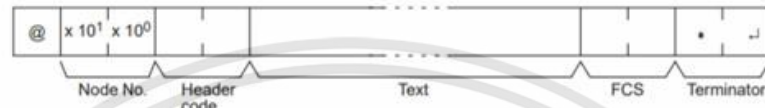
รูปที่ 2.17 แสดงการเชื่อมต่อไอซี MAX232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 การสื่อสาร Omron Host Link Protocol

Omron Host Link Protocol เป็นการสื่อสารระหว่างอุปกรณ์ของบริษัท Omron โดยมีพื้นฐานมาจาก ASCII ใช้สำหรับการสื่อสารผ่าน RS-232 หรือ RS-422 โดยมีการกำหนดรูปแบบการสื่อสารดังนี้

Command Frame Format When transmitting a command from the host computer, prepare the command data in the format shown below.



Normal Response Frame Format A normal response from the PC is returned in the format shown below. Prepare a program at the host so that the response data can be interpreted and processed.



Header code	PC mode			Name
	RUN	MON	PRG	
RR	Valid	Valid	Valid	IR/SR AREA READ
RL	Valid	Valid	Valid	LR AREA READ
RH	Valid	Valid	Valid	HR AREA READ
RC	Valid	Valid	Valid	TC PV READ
RG	Valid	Valid	Valid	TC STATUS READ
RD	Valid	Valid	Valid	DM AREA READ
RE	Valid	Valid	Valid	EM AREA READ
RJ	Valid	Valid	Valid	AR AREA READ
WR	Not valid	Valid	Valid	IR/SR AREA WRITE
WL	Not valid	Valid	Valid	LR AREA WRITE
WH	Not valid	Valid	Valid	HR AREA WRITE
WC	Not valid	Valid	Valid	TC PV WRITE
WG	Not valid	Valid	Valid	TC STATUS WRITE
WD	Not valid	Valid	Valid	DM AREA WRITE
WE	Not valid	Valid	Valid	EM AREA WRITE
WJ	Not valid	Valid	Valid	AR AREA WRITE

รูปที่ 2.18 แสดงรูปแบบการสื่อสารของ Omron Host Link Protocol (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 NI Vision Builder AI

2.11.1 โปรแกรม Ni Vision Builder AI คืออะไร?

เป็นซอฟต์แวร์ประยุกต์ช่วยพัฒนาและติดตั้งระบบตรวจสอบ machine vision ได้อย่างรวดเร็ว นอกจากนี้ยังสามารถตั้งค่ากล้อง และปรับแต่งการประมวลผลภาพได้เป็นร้อย ๆ อัลกอริทึม และสามารถเชื่อมต่อกับ automation hardware และสร้างผลลัพธ์ของการตรวจสอบได้

2.11.2 การทำงานของโปรแกรม

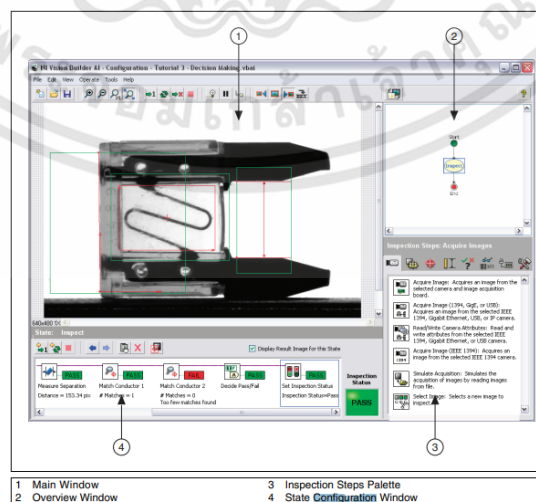
Vision Builder AI ใช้ state diagram และกำหนดการตรวจสอบด้วย state และ transition



รูปที่ 2.19 แสดงการเขียน state diagram

ภายใน state diagram แต่ละ state สามารถที่จะเชื่อมโยงไปยังหนึ่งหรือหลาย state หรือ cycle ได้ และสามารถกำหนดเงื่อนไขภายใน output ของ state เพื่อที่จะดำเนินการ state ต่อไปได้ โดย state diagram ทำงานต่อเนื่องจาก จุด Start ไปจนถึง จุด End

2.11.3 หน้าต่างโปรแกรม NI Vision Builder AI



รูปที่ 2.20 หน้าต่างโปรแกรม Ni Vision Builder AI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.3.1 Main windows จะแสดงการให้เห็นกระบวนการทำงานสำหรับการตรวจสอบ และใช้เพื่อกำหนด Region of Interest

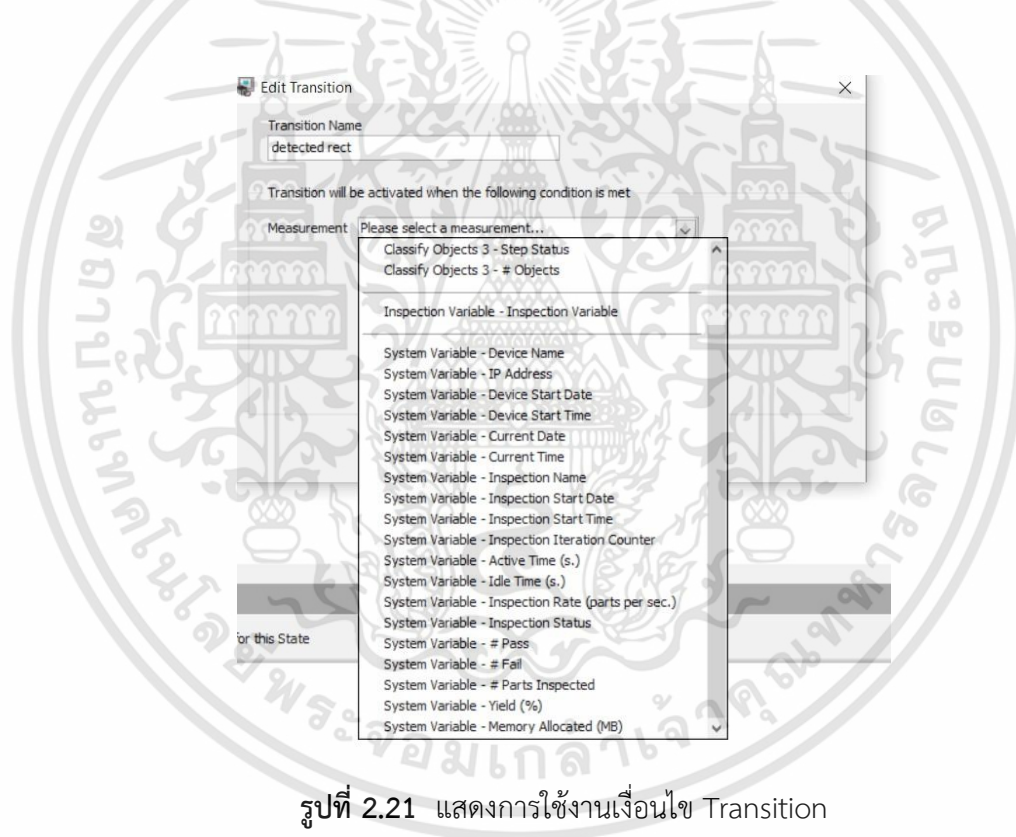
2.11.3.2 Overview Window แสดง state diagram

2.11.3.3 Inspection Steps palette เป็น list ของ step ที่จะใช้ในการตรวจสอบ ซึ่งการเลือกใช้จะขึ้นอยู่กับหน้างานและประเภทของของวัตถุ

2.11.3.4 State Configuration Window แสดงขั้นตอน (step) ของ state ขณะนั้น

2.11.4 Transitions

เป็นการวัด output จาก step ใน state ก่อนหน้า แล้วสามารถกำหนดเงื่อนไขได้ ถ้าตรงตามเงื่อนไขก็จะทำให้รันไปยัง state ต่อไปได้



รูปที่ 2.21 แสดงการใช้งานเงื่อนไข Transition

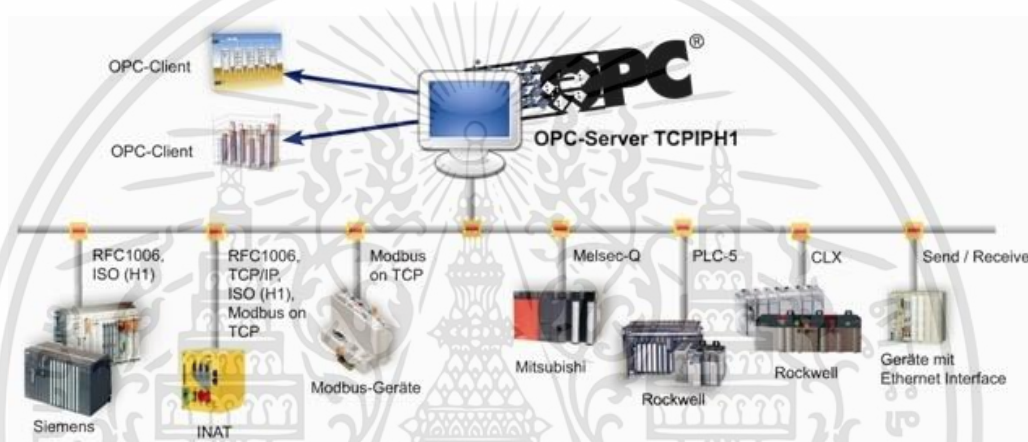
2.12 Image Processing

Image Processing เป็นวิธีการดำเนินการบางอย่างกับรูปภาพเพื่อให้ได้ภาพที่ได้รับการปรับปรุงหรือดึงข้อมูลที่เป็นประโยชน์บางอย่างออกมา เป็นการประมวลผลสัญญาณประเภทหนึ่งซึ่งอินพุตเป็นรูปภาพและเอาต์พุตอาจเป็นรูปภาพหรือลักษณะหรือคุณสมบัติที่เกี่ยวข้องกับรูปภาพนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 OPC Server

OPC ย่อมาจาก OLE For Process Control ซึ่งเป็นภาษาสำหรับการติดต่อสื่อสารระหว่างเครื่องจักรในอุตสาหกรรม เช่น HMI หรือ Remote Unit ต่างๆ ที่มีการใช้โปรโตคอลคนละแบบกัน เพื่อให้สามารถทำการสื่อสารกันได้ เปรียบเสมือนล่ามแปลภาษาของอุปกรณ์นั่นเอง ด้วยรูปแบบมาตรฐานของการสื่อสารด้วย OPC เป็นแบบเปิด ผู้ผลิตอุปกรณ์ เครื่องจักรต่างๆ จึงสามารถพัฒนาระบบสื่อสารให้เป็นไปตามมาตรฐาน เดียวกันโดยมีการสื่อสารระหว่างเซิร์ฟเวอร์ และไคลเอนต์ ทั้งนี้สามารถประยุกต์ใช้เกี่ยวกับการแลกเปลี่ยนข้อมูลจากอุปกรณ์คนละค่าย หรือ การรวบรวมข้อมูลจากอุปกรณ์ต่างๆ ที่มีมาตรฐานแตกต่างกันทำได้ง่ายขึ้น



รูปที่ 2.22 แสดงการเชื่อมต่อของ OPC Server

2.14 อัลกอริทึม K-Nearest Neighbors

K-Nearest Neighbors เป็นอัลกอริทึมใช้ในการจัดกลุ่ม โดยหาข้อมูลที่มีความคล้ายกันมากที่สุด โดยมีหลักการคือเปรียบเทียบข้อมูลที่สนใจกับข้อมูลอื่นว่ามีความคล้ายคลึงมากน้อยเพียงใดและถ้าหากข้อมูลที่เราสนใจมีความใกล้เคียงกับข้อมูลมากที่สุด ผลลัพธ์จะได้เป็นคำตอบของข้อมูลที่อยู่ใกล้ที่สุด ยกตัวอย่างเช่นระหว่างไวน์ขาวกับไวน์แดง ซึ่งจะมีองค์ประกอบทางเคมี 2 อย่างเรียกว่า Routine และ Myricetin พิจารณาด้วยการวัดระดับขององค์ประกอบทางเคมี



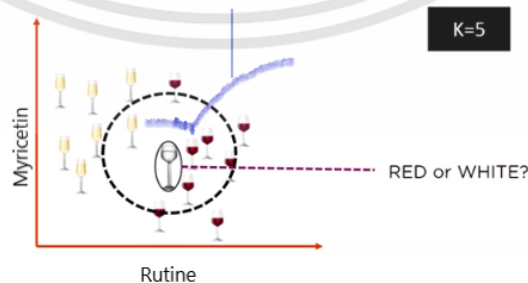
รูปที่ 2.23 การพิจารณา K-Nearest Neighbors

K สำหรับ K Nearest Neighbors คือค่าที่เปรียบเหมือนจำนวนเพื่อนบ้านใกล้ที่สุดที่ต้องการพิจารณาเช่นเรามีแก้วไวน์อันใหม่แล้วอยากจะทราบว่าเป็นไวน์ขาวหรือแดง



รูปที่ 2.24 การพิจารณา K-Nearest Neighbors (ต่อ)

ในขั้นตอนนี้เราก็พิจารณาเลือกเพื่อนบ้าน (k=5) และแก้วไวน์นี้จะถูกพิจารณาเป็นไวน์แดง เพราะ มีเพื่อนบ้านที่เป็นไวน์แดงจาก 4 ใน 5 หลักการเดียวกับการโหวต



รูปที่ 2.25 การพิจารณา K-Nearest Neighbors (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15 การจัดเรียงข้อมูล

การเรียงลำดับข้อมูลเป็นการจัดการข้อมูลที่กระทำกันมาในงานประยุกต์ต่างๆ มากมาย ยกตัวอย่างเช่น การนำข้อมูลนักศึกษา มาจัดเรียงลำดับตามรหัสนักศึกษา เพื่อนำไปใช้ในการพิมพ์ใบเซ็นชื่อเข้าสอบ หรือ การเรียงข้อมูลพนักงานตามรหัสของพนักงานเพื่อใช้ในการพิมพ์สลิปเงินเดือน เป็นต้นถ้ากำหนดให้ A เป็นลิสต์ขนาด n อิลิเมนต์ มี A1 ,A2 ,A3 ,... ,An เก็บอยู่ในหน่วยความจำ การจัดเรียงสามารถทำได้ n! แบบแต่โดยทั่วไปการจัดเรียงที่นิยม คือ การจัดเรียงลำดับข้อมูลจากน้อยไปมาก หรือจากมากไปหาน้อย เท่านั้น ซึ่งจะเลือกใช้อัลกอริทึมใดในการเรียงลำดับนั้น ขึ้นอยู่กับลักษณะของข้อมูลได้แก่ ขนาดข้อมูล และประสิทธิภาพการจัดเรียงของแต่ละอัลกอริทึม เมื่อเทียบกับปริมาณข้อมูล

2.16 Solenoid Valve

Solenoid Valve คือ อุปกรณ์ควบคุมการไหลของของไหล เช่น น้ำ อากาศ หรือก๊าซต่างๆ เป็นต้น ภายในวาล์วชนิดนี้เพิ่มส่วนประกอบขึ้นมาคือ ขดลวดพันแม่เหล็ก เป็นตัวกลางที่ใช้ร่วมกับพลังงานไฟฟ้า โดยการปล่อยกระแสไฟฟ้าผ่านขดลวดจะเกิดพลังงานสนามแม่เหล็ก ส่งผลการทำงานปิด-เปิด ช่องการไหลของตัววาล์ว โดย Solenoid Valve ที่เลือกใช้เป็นรุ่น model 4V210-08E4 เป็นแบบ 4/2, Pressure: 1.5-8 bar



รูปที่ 2.26 Solenoid Valve

2.17 Vacuum Ejector

Vacuum Ejector คือ ตัวกำเนิดสุญญากาศ โดยอาศัยหลักการ Venturi คือหลักการทำงานที่ใช้ความเร็วของลมอัด เคลื่อนที่ผ่านคอขวด และที่บริเวณคอขวดนี้จะเจาะรูเล็กๆไว้ เมื่อลมอัดเคลื่อนที่ผ่าน ความเร็วลมอัดจะเพิ่มขึ้น ความดันลมอัดลดลง จะทำให้เกิดสุญญากาศขึ้น นิยมใช้ต่อร่วมกับหัวดูด เหมาะสำหรับงานดูดจับปล่อยชิ้นงาน



รูปที่ 2.27 Vacuum Ejector

2.18 CX-Programmer

CX-Programmer เป็นโปรแกรมที่ใช้สำหรับเขียนคำสั่งและติดต่อระหว่าง PLC กับคอมพิวเตอร์โดยใช้ภาษาเฉพาะในการเขียนคำสั่งควบคุม เช่น Ladder, STL, FBC เป็นต้นซึ่งสามารถกำหนดสัญญาณ Input และ Output กำหนดตำแหน่งของหน่วยความจำ การตั้งค่าอุปกรณ์ต่างๆ นอกจากนี้ยังสามารถทำการจำลองการทำงานของคำสั่งได้อีกด้วย

บทที่ 3

วิธีดำเนินงาน

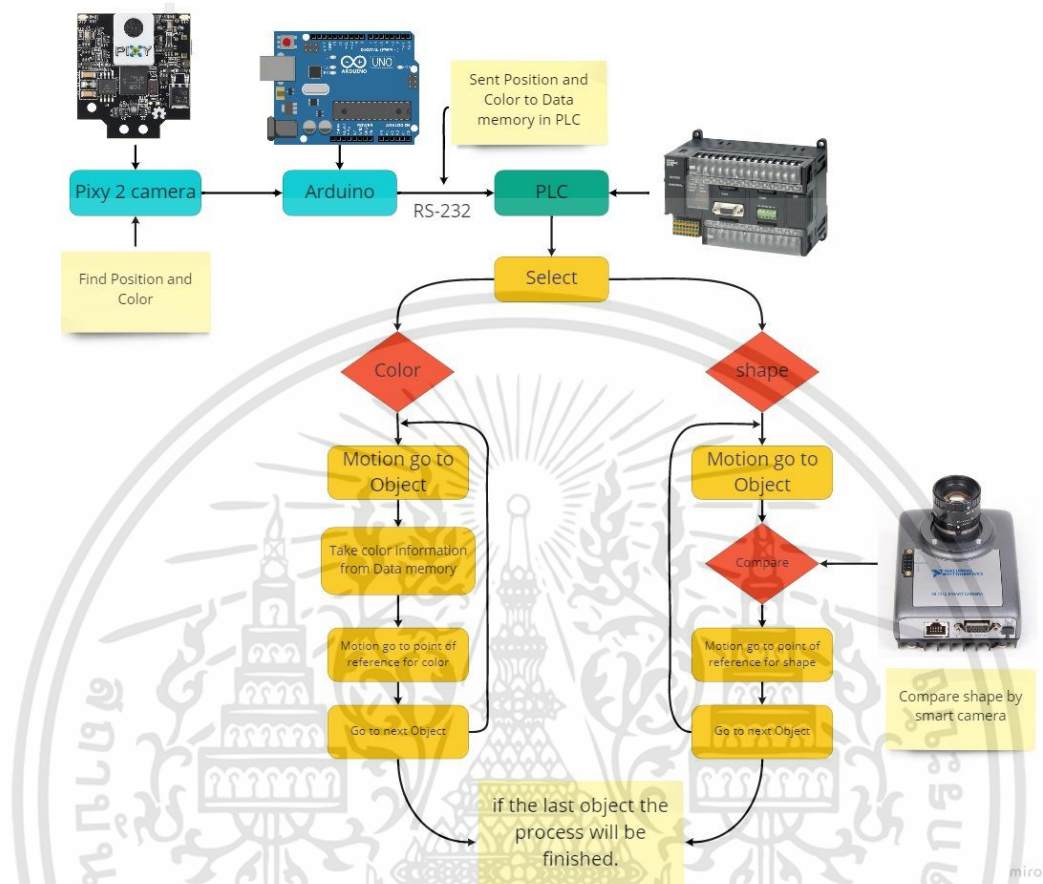
3.1 อุปกรณ์

3.1.1 AC Servo Motor	3 ตัว
3.1.2 PLC Omron CP1H	1 ตัว
3.1.3 Arduino Uno R3	1 ตัว
3.1.4 Smart camera NI 1732	1 ตัว
3.1.5 Servo Motor Drive	3 ตัว
3.1.6 RS-232 Shield	1 ชิ้น
3.1.7 Pixy2 camera	1 ตัว
3.1.8 Joy Strick	2 ตัว
3.1.9 สาย USB	
3.1.10 ชิ้นงานตัวอย่าง	
3.1.10.1 วงกลม	4 ชิ้น
3.1.10.2 สีเหลี่ยมจัตุรัส	4 ชิ้น
3.1.10.3 แปดเหลี่ยม	4 ชิ้น
3.1.10.4 สีเหลี่ยมข้าวหลามตัด	4 ชิ้น
3.1.11 Solenoid Valve + หัว Vacuum	1 ชุด

3.2 ซอฟต์แวร์

3.2.1 CX-Programmer
3.2.2 NB-Designer
3.2.3 PixyMon v2
3.2.4 NI Vision Builder AI
3.2.5 Arduino IDE

3.3 แผนผังการดำเนินงาน



รูปที่ 3.1 แผนผังแสดงการทำงาน

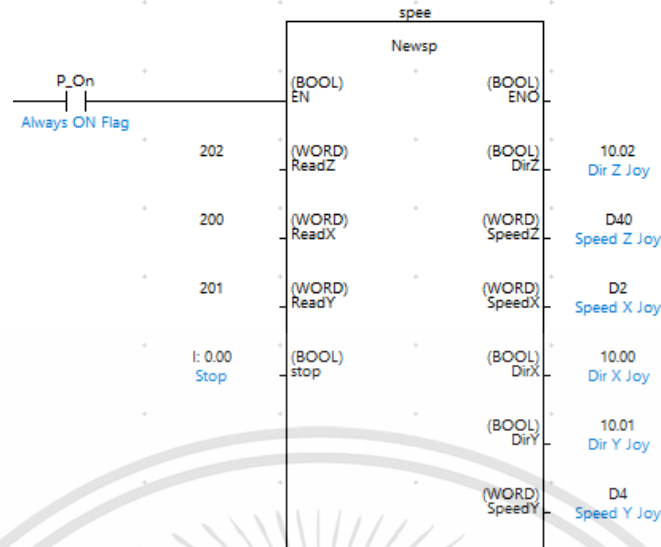
3.4 วิธีการดำเนินงาน

โครงการนี้มีการดำเนินงานแบ่งออกเป็น 4 ส่วน คือ การเคลื่อนที่, การวิเคราะห์และจำแนก, การนำวัตถุไปจัดเก็บ และการออกแบบหน้าจอควบคุมและแสดงผลผ่าน HMI

3.4.1 การเคลื่อนที่

3.4.1.1 ทำการเขียนโปรแกรมทดสอบการเคลื่อนที่ ของ AC Servo Motor โดยใช้ การเขียนโปรแกรมโดยใช้ภาษา STL ร่วม Ladder ในการเคลื่อนที่ของ AC Servo Motor ด้วย Joy Stick และโปรแกรมที่เขียนให้ AC Servo Motor เคลื่อนที่ไปยังตำแหน่งที่เรากำหนดบนระนาบ ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 Function Block สำหรับการควบคุมการเคลื่อนที่ด้วย Joy Stick

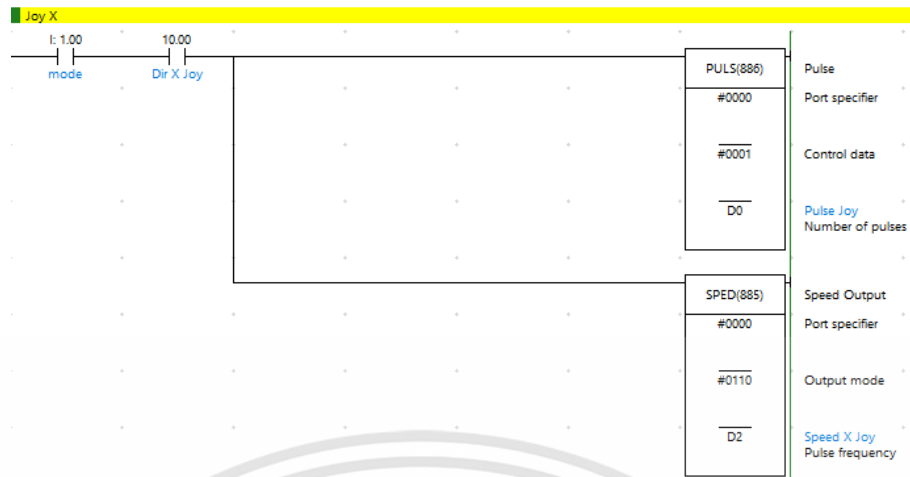
```

1  ADX :=WORD_to_INT(ReadX);
2  ADY :=WORD_to_INT(ReadY);
3  ADZ :=WORD_to_INT(ReadZ);
4  if (ADX > -200) and (ADX < 200)
5  then ADX :=0;
6  END_IF;
7  if (ADX < 0)
8  then ADX :=abs(ADX);
9  DirX :=false;
10 else
11  DirX :=true;
12 END_IF;
13 if (ADY > -200) and (ADY < 200)
14 then ADY :=0;
15 END_IF;
16 if (ADY < 0)
17 then ADY :=abs(ADY);
18 DirY :=false;
19 else
20 DirY :=true;
21 END_IF;
22 if (ADZ > -200) and (ADZ < 200)
23 then ADZ :=0;
24 END_IF;
25 if (ADZ < 0)
26 then ADZ :=abs(ADZ);
27 DirZ :=false;
28 else
29 DirZ :=true;
30 END_IF;
31 SpeedX :=INT_to_WORD(ADX*15);
32 SpeedY :=INT_to_WORD(ADY*15);
33 SpeedZ :=INT_to_WORD(ADZ*15);
34 if (stop = true)
35 then SpeedX :=INT_to_WORD(0);
36  SpeedY :=INT_to_WORD(0);
37  SpeedZ :=INT_to_WORD(0);
38 end_if ;

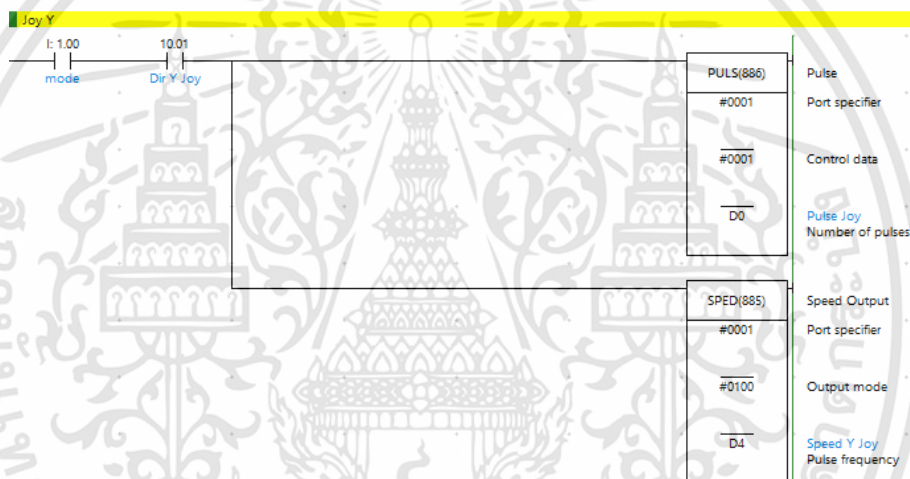
```

รูปที่ 3.3 Structured Text สำหรับการเคลื่อนที่โดยใช้ Joy Stick

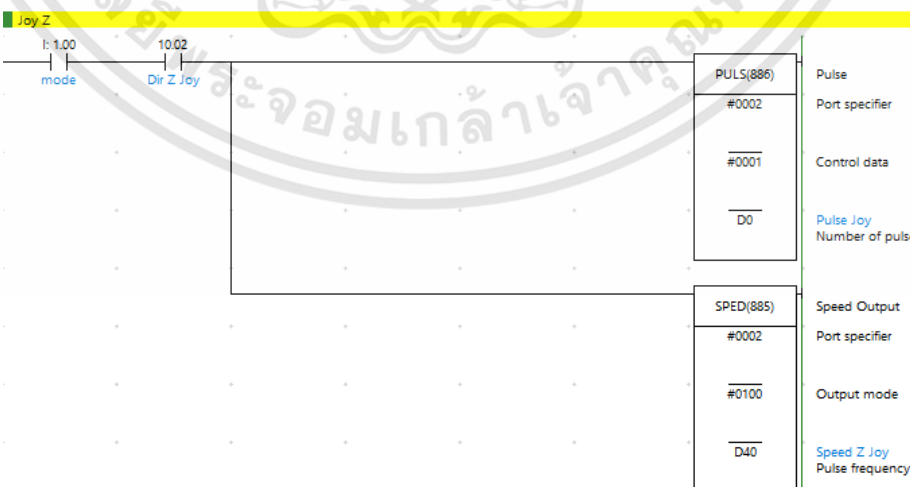
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่โดยใช้ Joy Stick ในแนวแกน X

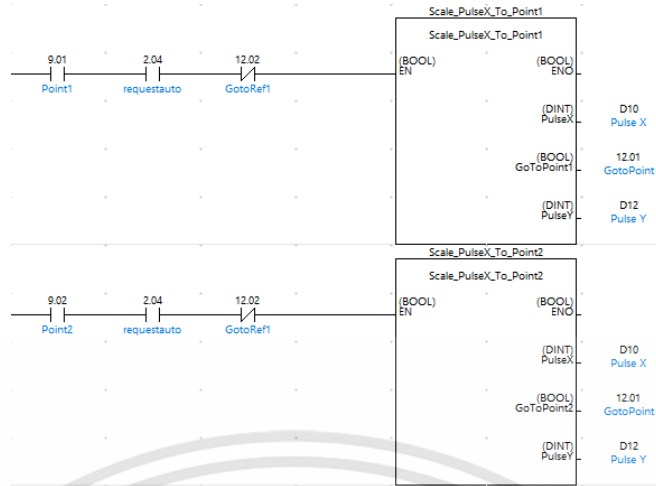


รูปที่ 3.5 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่โดยใช้ Joy Stick ในแนวแกน Y

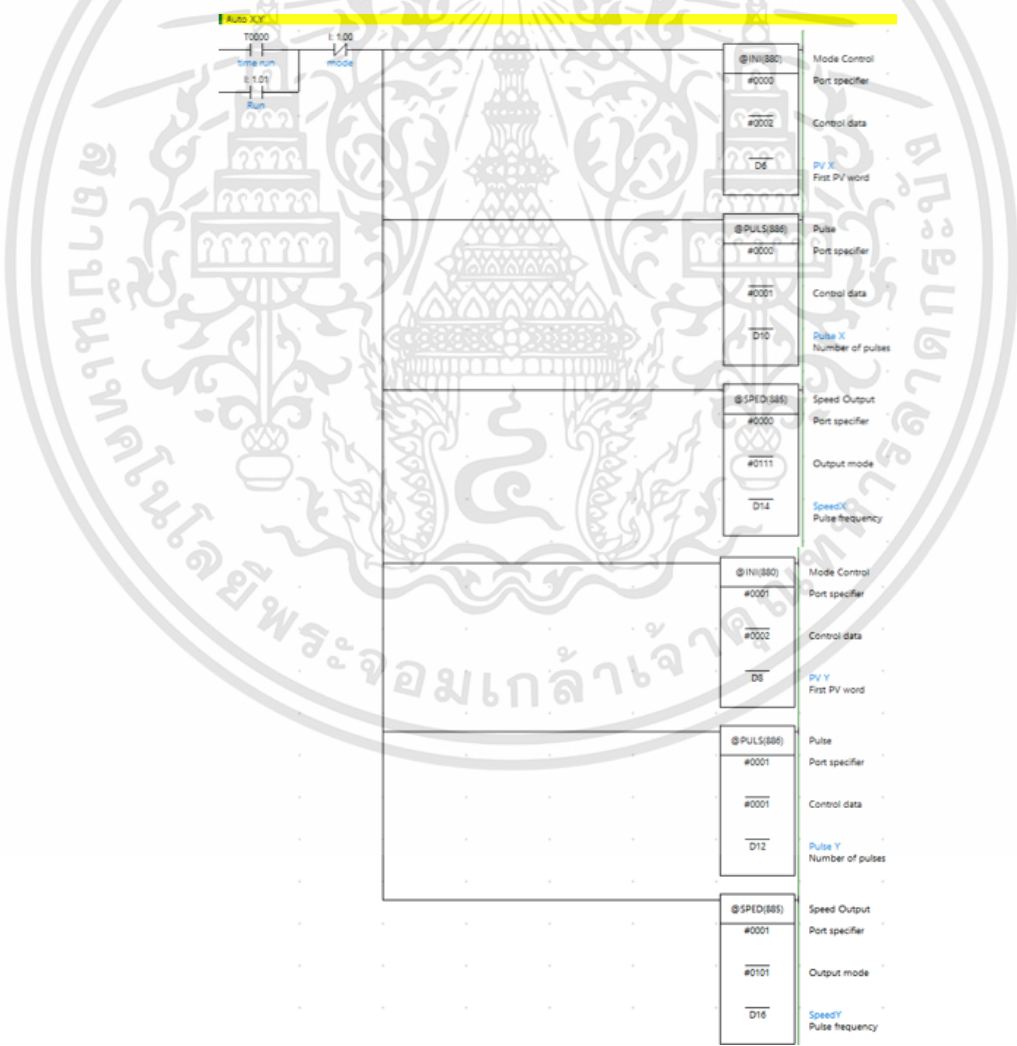


รูปที่ 3.6 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่โดยใช้ Joy Stick ในแนวแกน Z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 Function Block สำหรับการคำนวณค่า Pulse ในแกน X และ Y ให้ไปในตำแหน่งที่เราต้องการ



รูปที่ 3.8 Ladder Diagram สำหรับการควบคุมการเคลื่อนที่ในแกน X และ Y แบบ Auto

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1.3 ศึกษาการใช้งานโปรแกรม PixyMon V2 การตรวจจับตำแหน่งวัตถุ โดยการนำวัตถุวางไว้บนระนาบและทำการตั้งค่าให้โปรแกรมสามารถตรวจจับตำแหน่งวัตถุได้

3.4.1.4 ศึกษาการใช้งาน Pixy2 Library เพื่อสั่งการ Arduino Uno R3 กับกล้อง Pixy camera ซึ่งทำการเขียนโปรแกรมผ่าน Arduino IDE

3.4.1.5 ศึกษาการใช้งานการส่งค่าตำแหน่งจาก Arduino Uno R3 ไปยัง PLC Omron CP1H ซึ่งส่งข้อมูล Omron Host Link Protocol โดยใช้สายสัญญาณ RS-232

3.4.1.6 นำ Arduino UNO R3 เชื่อมต่อกับ PLC Omron CP1H โดยใช้สายสัญญาณ RS-232 และส่งข้อมูลตำแหน่งของวัตถุที่ได้รับจากกล้อง Pixy camera ไปยัง PLC Omron CP1H จากนั้นเขียนโปรแกรมควบคุมการเคลื่อนที่ของ AC Servo Motor เพื่อควบคุมให้เคลื่อนที่ไปหาวัตถุแบบอัตโนมัติ โดยไม่ต้องใช้ Joy Stick และทดสอบการทำงาน

```
#include <SPI.h>
#include <Pixy2.h>
#include <avr/wdt.h>;
Pixy2 pixy;
int tempx, tempy, i;
int tempc ;
int xpos[10];
int ypos[10];
int cpos[10];
void fxsent();
void fxrecieve();
void fxrecievenumblock();
void fxrecievereset();
byte FCA2;

int m = 1;
int p = 1;
int n = 1;
String KKK;
String FCSStr2;
byte FCS2;
String message2;
String BBB[10] ;
String CCC[10] ;
String DDD[10] ;
String XPOSX ;
String YPOSY ;
String CPOSC ;
String XMAX;
String YMAX;
String XMIN;
String YMIN;
```

รูปที่ 3.9 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการกำหนดค่าตัวแปรต่างๆ

```

if (pixy.ooc.numBlocks)
{
  for (int i=0;i<pixy.ooc.numBlocks;i++) //memory position in array
    ypos[i]=pixy.ooc.blocks[i].m_x; //swap boos pixy & plane are inverse

  for (int i=0;i<pixy.ooc.numBlocks;i++) //swap boos pixy & plane are inverse
    xpos[i]=pixy.ooc.blocks[i].m_y;

  for (int i=0;i<pixy.ooc.numBlocks;i++)
    cpos[i]=pixy.ooc.blocks[i].m_signature ;

  for (int i=0;i<(pixy.ooc.numBlocks);++i)
  {
    for (int j=i+1;j<(pixy.ooc.numBlocks);++j)
    {
      if (xpos[i] < xpos[j] ) //swap algorithm to find the nearest object
      {
        tempx = xpos[i];
        xpos[i] = xpos[j];
        xpos[j] = tempx;

        tempy = ypos[i];
        ypos[i] = ypos[j];
        ypos[j] = tempy;

        tempc = cpos[i];
        cpos[i] = cpos[j];
        cpos[j] = tempc;
      }
    }
  }
}

```

รูปที่ 3.10 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการจัดเรียงตำแหน่งของวัตถุ

```

void fixsent()
{
  String message;
  String FCSstr;
  byte FCS;
  int polo = 1;
  String JY ;

  JY = String(polo,HEX);

  if (JY.length() == 1) JY = "000" + JY;
  if (JY.length() == 2) JY = "00" + JY;
  if (JY.length() == 3) JY = "0" + JY;

  message = "@00WD1005" + XPOSX;
  FCS = message[0]; //0

  for (byte j=1; j <= message.length()-1; j++)
  {
    FCS = FCS ^ message[j];
  }
  FCSstr = String(FCS,HEX);
  FCSstr.toUpperCase();
  if (FCSstr.length() == 1) FCSstr = "0" + FCSstr;
  message = message + FCSstr + "*" + char(13);
  for (int i=0;i<10;i++)
  {
    Serial.println(message); //repeat to confirm position :]
  }
}

```

รูปที่ 3.11 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของฟังก์ชันการส่งข้อมูลตำแหน่งไปยัง PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

245 message = "@00WD1000" + KKK + b;
246 FCS = message[0]; //@
247
248 for (byte j=1; j <= message.length()-1; j++)
249 {
250   FCS = FCS ^ message[j];
251 }
252 FCSStr = String(FCS,HEX);
253 FCSStr.toUpperCase();
254 if (FCSStr.length() == 1) FCSStr = "0" + FCSStr;
255 message = message + FCSStr + "*" + char(13);
256 for (int i=0;i<10;i++)
257 {
258   Serial.println(message); //repeat to confirm position :)
259 }
260
261 /////////////////////////////////////////////////////////////////// X AXIS ABOVE
262

```

รูปที่ 3.12 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของฟังก์ชันการส่งข้อมูลตำแหน่งไปยัง PLC (ต่อ)

```

while (pixy.ccc.numBlocks > 0)
{
  fxrecievenumblock();

  KKK = String (pixy.ccc.numBlocks);

  if (KKK.length() == 1) KKK = "000" + KKK;
  if (KKK.length() == 2) KKK = "00" + KKK;
  if (KKK.length() == 3) KKK = "0" + KKK;

  message2 = "@00WD1000" + KKK ;
  FCS2 = message2[0]; //@

  for (byte j=1; j <= message2.length()-1; j++)
  {
    FCS2 = FCS2 ^ message2[j];
  }
  FCSStr2 = String(FCS2,HEX);
  FCSStr2.toUpperCase();
  if (FCSStr2.length() == 1) FCSStr2 = "0" + FCSStr2;
  message2 = message2 + FCSStr2 + "*" + char(13);
  for (int i=0;i<10;i++)
  {
    Serial.println(message2); //repeat to confirm position :)
  }
  fxrecieve();

  fxsent();

  fxrecievereset();

  ArduinoReset();
}

```

รูปที่ 3.13 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการเรียกใช้ฟังก์ชันการรับ – ส่งค่า

3.4.2 การวิเคราะห์และจำแนก

3.4.2.1 เขียนโปรแกรมให้กล่อง Pixy2 เพื่อส่งค่าสีของวัตถุต่างๆ ให้กับ PLC Omron CP1H โดยใช้โปรแกรม Arduino IDE ในการเขียนโปรแกรมโดยกำหนดสีของวัตถุเป็น 4 สี คือ สีแดง สีเขียว สีฟ้า และสีเหลือง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (pixy.ooo.numBlocks)
{
  for (int i=0;i<pixy.ooo.numBlocks;i++) //memory position in array
  ypoes[i]=pixy.ooo.blocks[i].m_x; //swap boos pixy & plane are inverse

  for (int i=0;i<pixy.ooo.numBlocks;i++)
  xpos[i]=pixy.ooo.blocks[i].m_y; //swap boos pixy & plane are inverse

  for (int i=0;i<pixy.ooo.numBlocks;i++)
  cpos[i]=pixy.ooo.blocks[i].m_signature ;

  for (int i=0;i<(pixy.ooo.numBlocks);++i)
  {
    for (int j=i+1;j<(pixy.ooo.numBlocks);++j)
    {
      if (xpos[i] < xpos[j] ) //swap algorithm to find the nearest object
      {
        tempx = xpos[i];
        xpos[i] = xpos[j];
        xpos[j] = tempx;

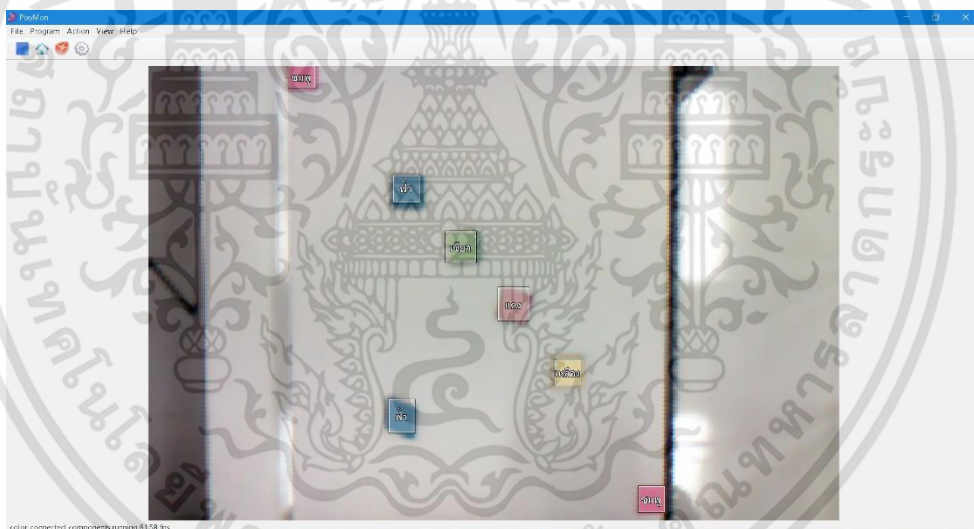
        tempy = ypoes[i];
        ypoes[i] = ypoes[j];
        ypoes[j] = tempy;

        tempc = cpos[i];
        cpos[i] = cpos[j];
        cpos[j] = tempc;
      }
    }
  }
}
}

```

รูปที่ 3.14 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของการรับค่าสีจากกล้อง Pixy2 camera

3.4.2.2 ทดสอบการทำงานของโปรแกรมจำแนกสีของวัตถุ โดยใช้วัตถุทั้งหมด 4 สี



รูปที่ 3.15 หน้าต่างโปรแกรม PixyMon V2 ในขณะที่จำแนกสีของวัตถุ

3.4.2.3 เขียนโปรแกรมเพื่อกำหนดให้ Arduino IDE มีการส่งข้อมูลการจำแนกสี ไปยัง PLC Omron CP1H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

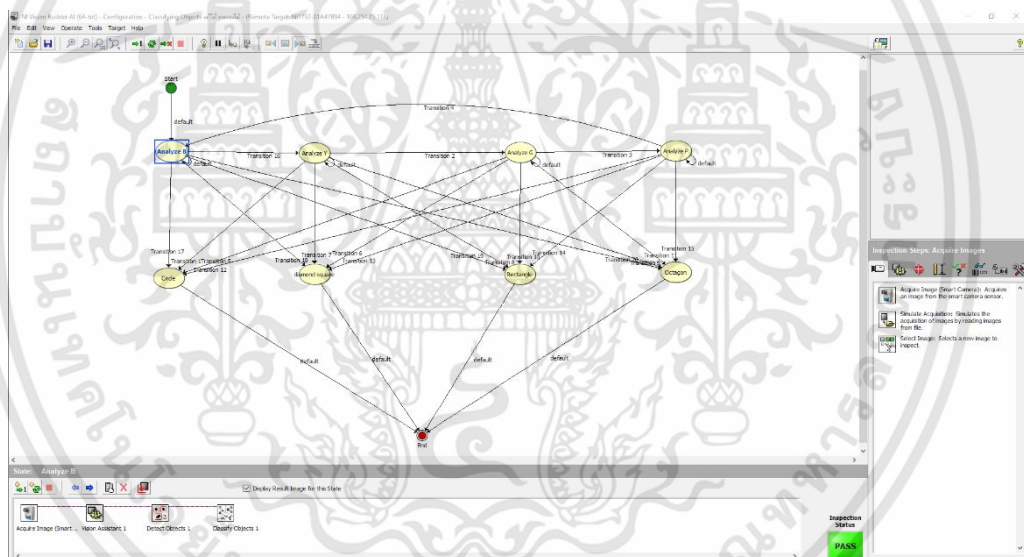
262     message = "@00WD1300" + KKK + c;
263     FCS = message[0]; //@
264
265     for (byte j=1; j <= message.length()-1; j++)
266     {
267         FCS = FCS ^ message[j];
268     }
269     FCSStr = String(FCS,HEX);
270     FCSStr.toUpperCase();
271     if (FCSStr.length() == 1) FCSStr = "0" + FCSStr;
272     message = message + FCSStr + "*" + char(13);
273     for(int i=0;i<10;i++)
274     {
275         Serial.println(message); //repeat to confirm position :)
276     }
277 }
278
279 }
280
281 //////////////////////////////////////// COLOR

```

รูปที่ 3.16 คำสั่งภายในโปรแกรม Arduino IDE ในส่วนของฟังก์ชันการส่งข้อมูลสี่ไปยัง PLC

3.4.2.4 ทดสอบการส่งข้อมูลของการจำแนกสีของ Pixy2 camera โดยทำการตรวจสอบหน่วยความจำของ PLC Omron CP1H

3.4.2.5 เขียนโปรแกรมให้กล้อง Smart camera NI 1732 ผ่านโปรแกรม NI Vision Builder AI



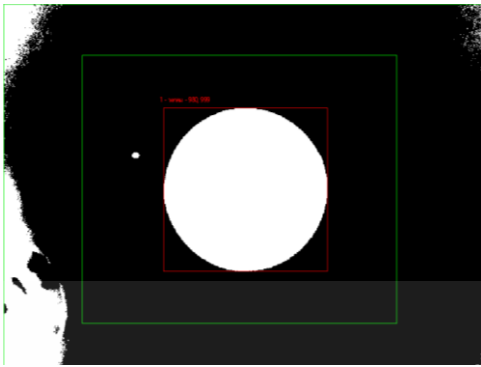
รูปที่ 3.17 โปรแกรม NI Vision Builder AI

3.4.2.6 กล้อง Pixy2 ตรวจจับตำแหน่งและสีของวัตถุบนระนาบของเครื่อง CNC

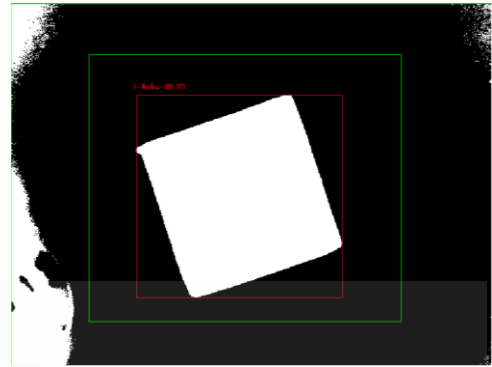
3.4.2.7 กล้อง Pixy2 จะส่งพิกัดจุด coordinate ไปให้ PLC (ส่วนของการเคลื่อนที่)

3.4.2.8 เมื่อกำลัง Smart camera NI 1732 ถูกเลื่อนมาอยู่ที่ตำแหน่งของวัตถุ ตัวกล้องจะตรวจจับและจำแนกชนิดของวัตถุ

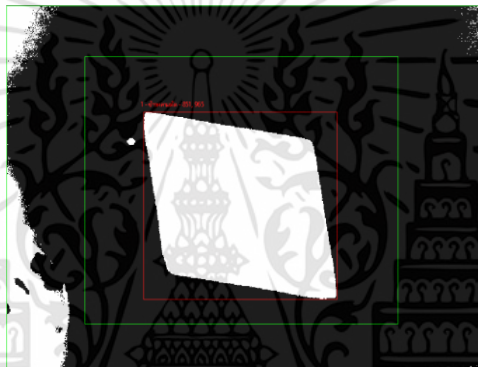
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 ตรวจสอบวัตถุชนิดวงกลม



รูปที่ 3.19 ตรวจสอบวัตถุชนิดสี่เหลี่ยมจัตุรัส



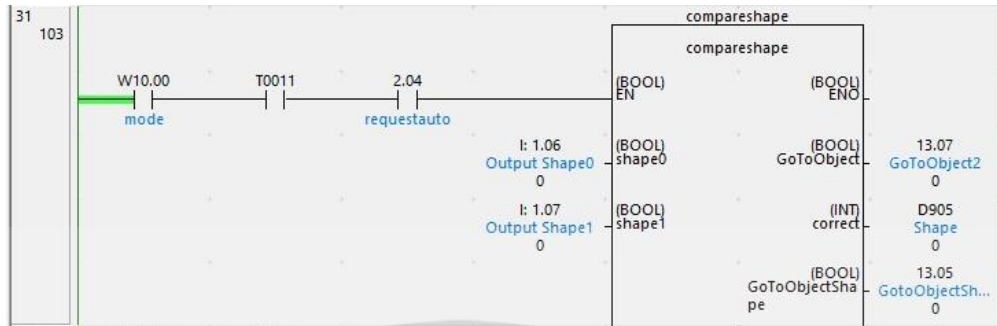
รูปที่ 3.20 ตรวจสอบวัตถุชนิดสี่เหลี่ยมข้าวหลามตัด



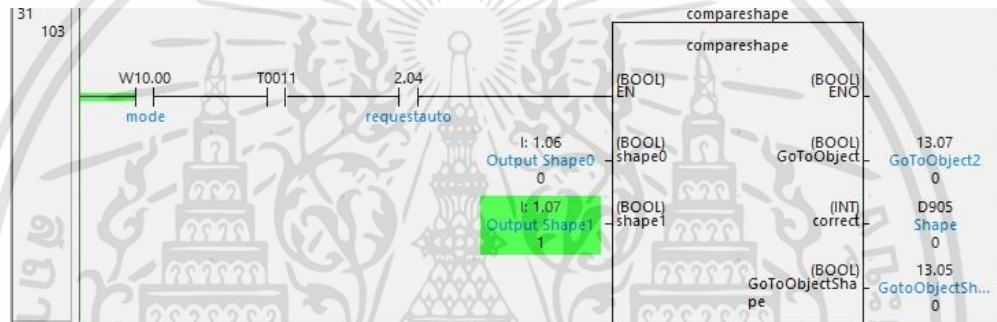
รูปที่ 3.21 ตรวจสอบวัตถุชนิดแปดเหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

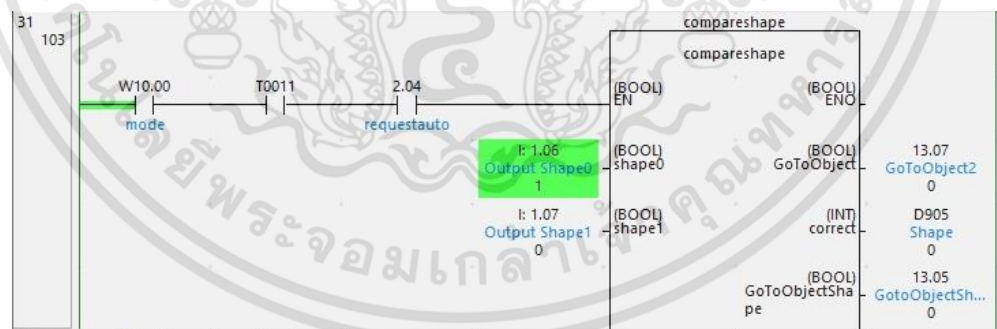
3.4.2.9 เมื่อกล้อง Smart camera NI 1732 ตรวจจับและจำแนกวัตถุสำเร็จ กล้องจะส่งข้อมูลชนิดของวัตถุไปให้ PLC ผ่าน I/O Accessory Board



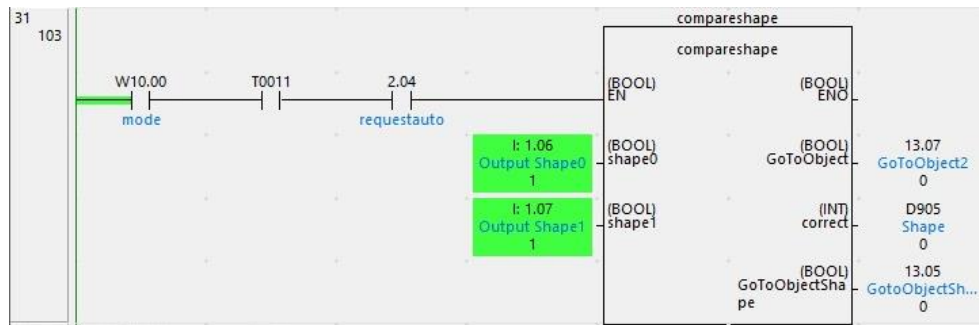
รูปที่ 3.22 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดวงกลมจะส่งเป็น 0 0



รูปที่ 3.23 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดสี่เหลี่ยมจัตุรัสจะส่งเป็น 0 1



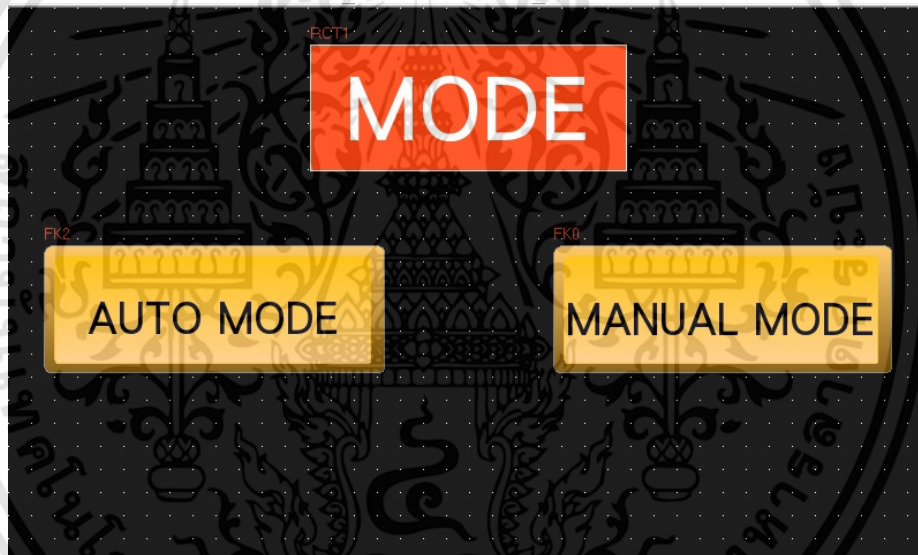
รูปที่ 3.24 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดสี่เหลี่ยมข้าวหลามตัดจะส่งเป็น 1 0



รูปที่ 3.25 การส่งข้อมูลจาก Smart camera ไปให้ PLC ชนิดแปดเหลี่ยมจะส่งเป็น 1 1

3.4.3 การออกแบบหน้าจอควบคุมและแสดงผลผ่าน HMI

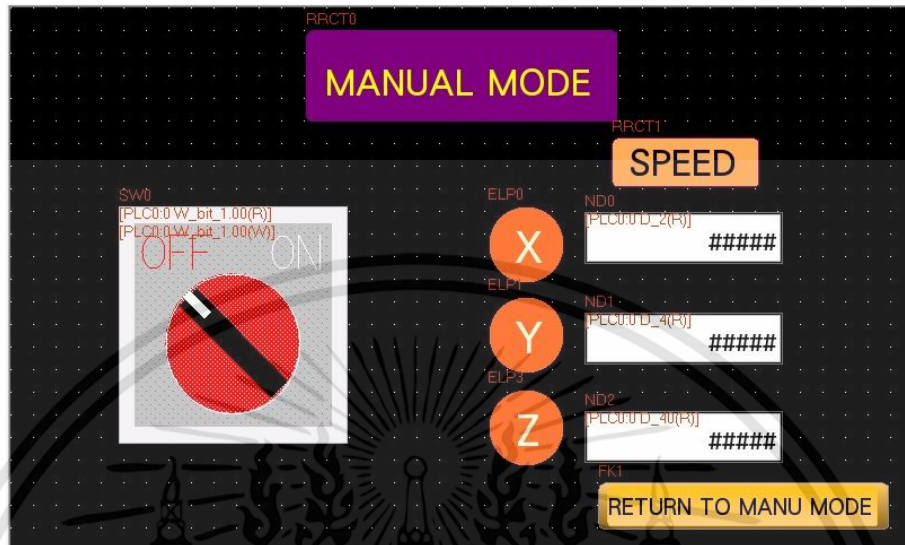
3.4.3.1 หน้าแรกเป็นการเลือกโหมดปฏิบัติการ โดยมีโหมดอัตโนมัติ โหมดแมน
นวลให้เลือกใช้



รูปที่ 3.26 หน้าต่างใน HMI Screen

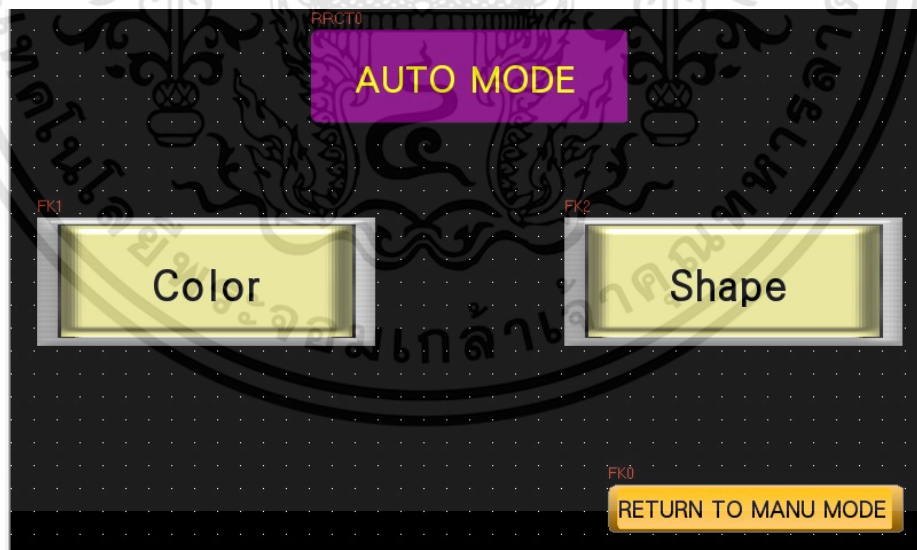
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3.2 หน้าต่างโหมดแมนนวล แสดงค่าความเร็วของแต่ละแกนที่ใช้ในการเคลื่อนที่



รูปที่ 3.27 หน้าต่างในโหมดแมนนวล

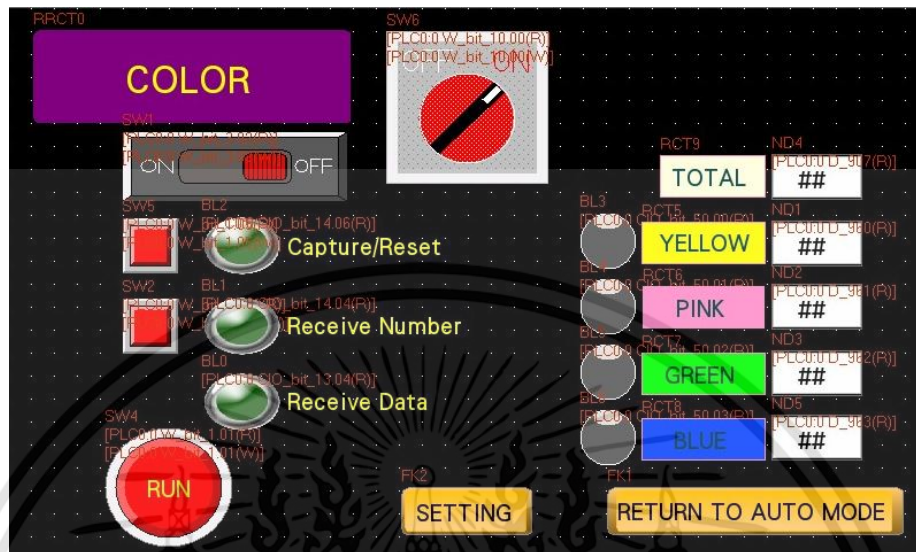
3.4.3.3 หน้าต่างโหมดอัตโนมัติ จะมีให้เลือกว่าจะใช้การจำแนกวัตถุแบบใด ได้แก่ การจำแนกแบบแยกสี และการจำแนกแบบรูปร่างของวัตถุ



รูปที่ 3.28 หน้าต่างในโหมดอัตโนมัติ

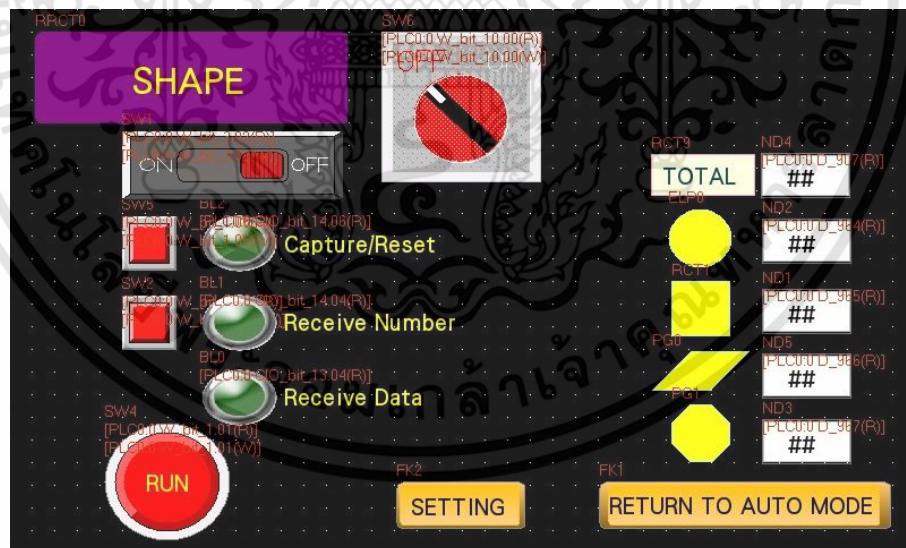
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3.4 หน้าต่างในโหมดการจำแนกวัตถุด้วยสี จะมีปุ่มเริ่มการทำงานที่อยู่ฝั่งซ้ายของจอ และช่องแสดงวัตถุที่จำแนกได้ทางด้านขวาของหน้าจอ



รูปที่ 3.29 หน้าต่างในโหมดการจำแนกวัตถุด้วยสี

3.4.3.5 หน้าต่างในโหมดการจำแนกวัตถุด้วยรูปร่าง จะมีปุ่มเริ่มการทำงานที่อยู่ฝั่งซ้ายของจอ และช่องแสดงวัตถุที่จำแนกได้ทางด้านขวาของหน้าจอ

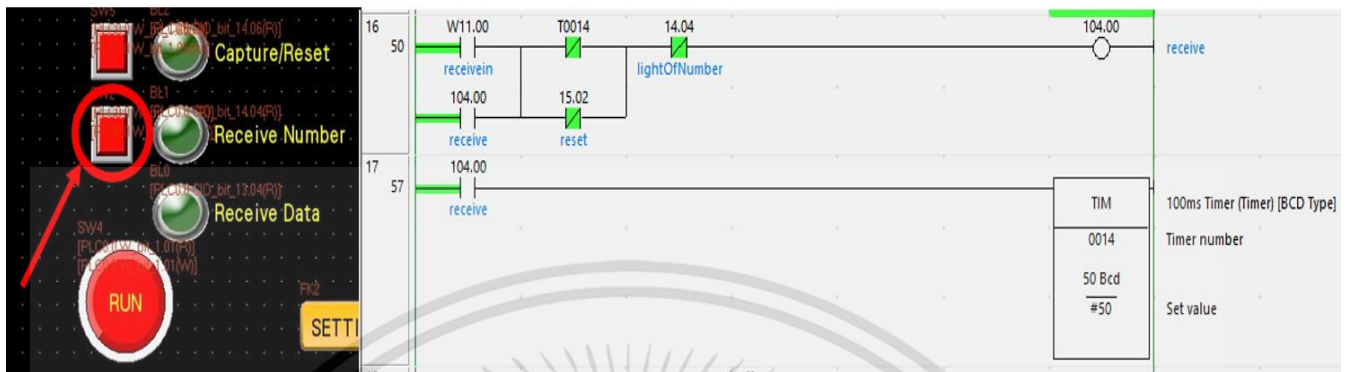


รูปที่ 3.30 หน้าต่างในโหมดการจำแนกวัตถุด้วยรูปร่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

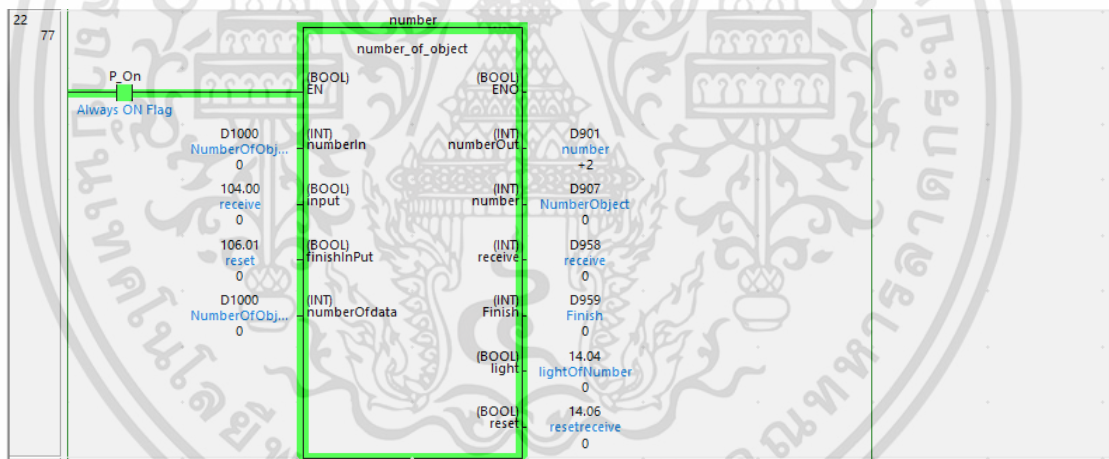
3.4.4 การนำวัตถุไปจัดเก็บ

3.4.4.1 เริ่มต้นด้วยการกดปุ่ม Receive Number ที่หน้าจอ HMI



รูปที่ 3.31 Ladder ที่ใช้ในการรับค่าจาก arduino

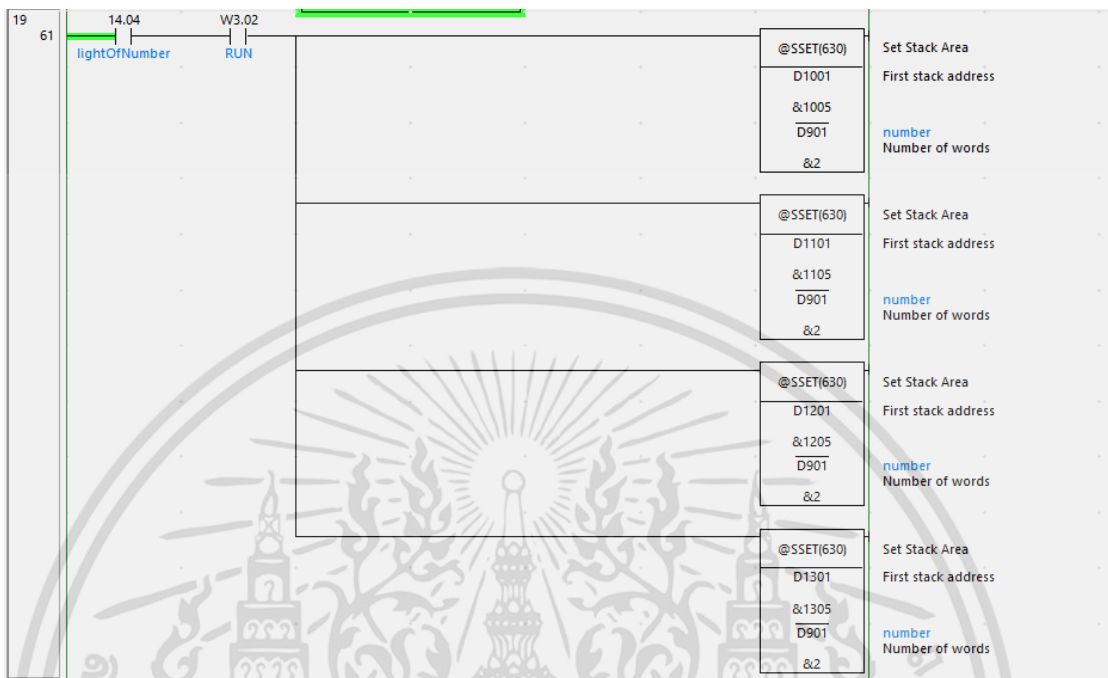
3.4.4.2 จากนั้น Function Block จากรูปด้านล่างจะเป็นตัวบอกว่า PLC ได้รับค่าจำนวนของวัตถุจนระนาบแล้วหรือยัง



รูปที่ 3.32 Function Block ที่ใช้ในการรับค่าจำนวนของวัตถุจนระนาบ

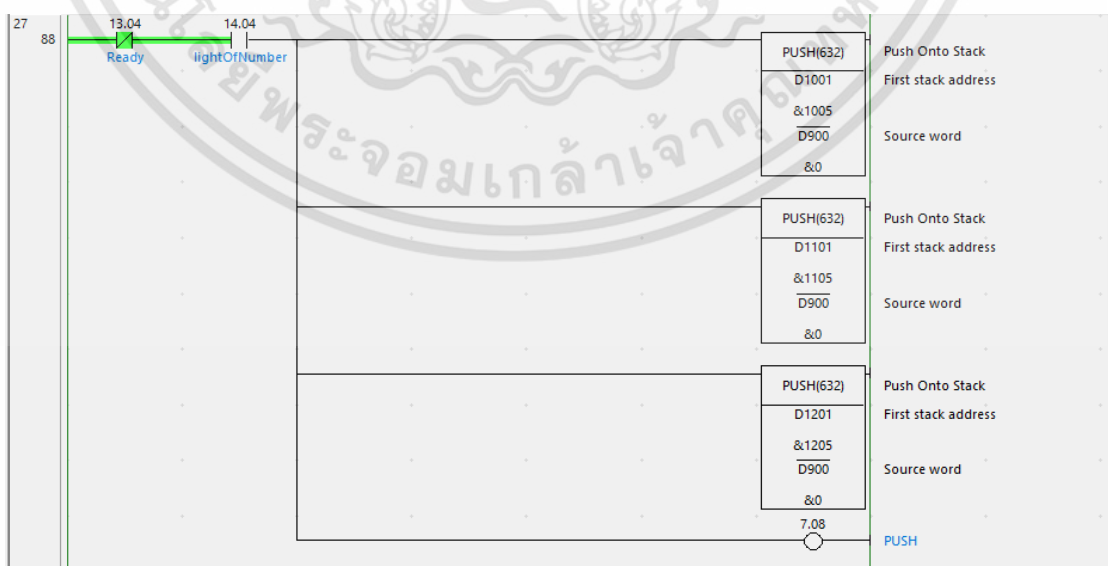
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4.3 หลังจาก PLC ได้รับค่าจำนวนของวัตถุแล้วจะทำการ Set Stack เพื่อจองพื้นที่ที่เท่ากับจำนวนของวัตถุที่ได้รับมา



รูปที่ 3.33 Function ที่ใช้ในการ Set stack

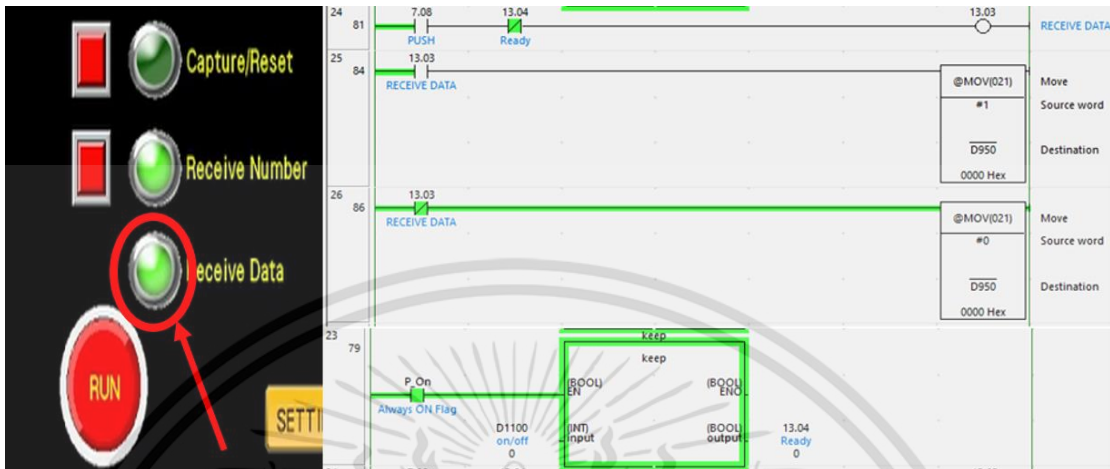
3.4.4.4 ในขณะที่ทำการ Set Stack โปรแกรมจะทำการ PUSH ค่า 0 เข้าไปยังพื้นที่ของ Stack ที่เราได้จองเอาไว้เพื่อเคลียร์ค่าให้เป็น 0 และ ทำให้สามารถใช้ Function First In First Out ในขั้นตอนต่อไปได้



รูปที่ 3.34 Function ที่ใช้ในการ PUSH ค่า

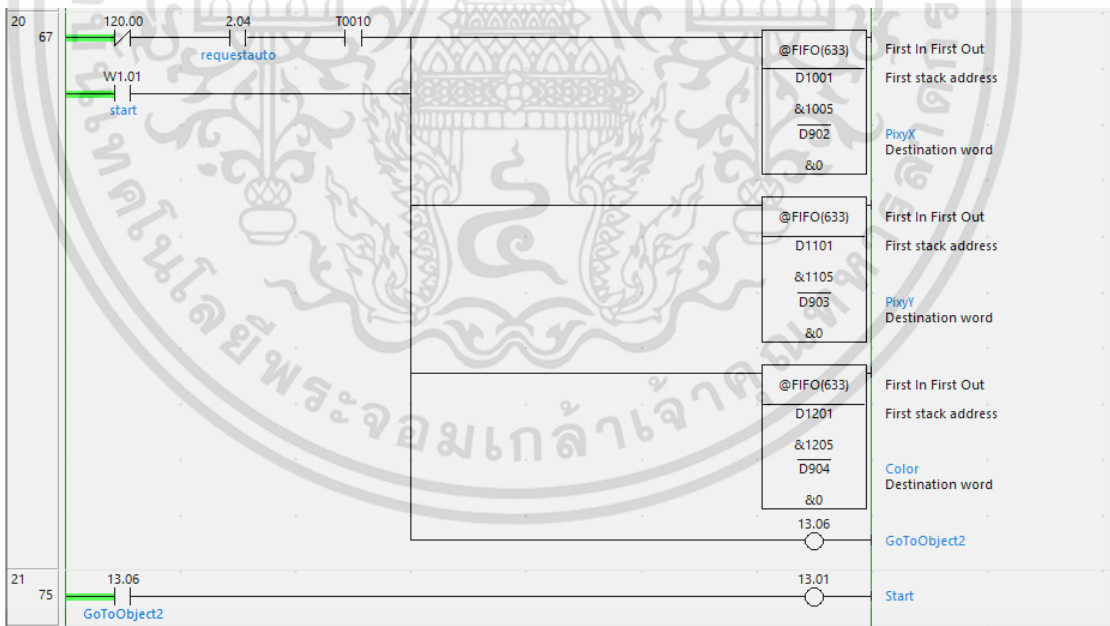
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4.5 เป็นการแจ้งให้ Arduino ส่งค่าตำแหน่งและสีของวัตถุมายังหน่วยความจำของ PLC และโชว์ไฟแสดงผลบนหน้าจอ HMI เมื่อได้รับตำแหน่งของวัตถุแล้ว



รูปที่ 3.35 Ladder ที่ใช้ในการรับตำแหน่งของวัตถุ

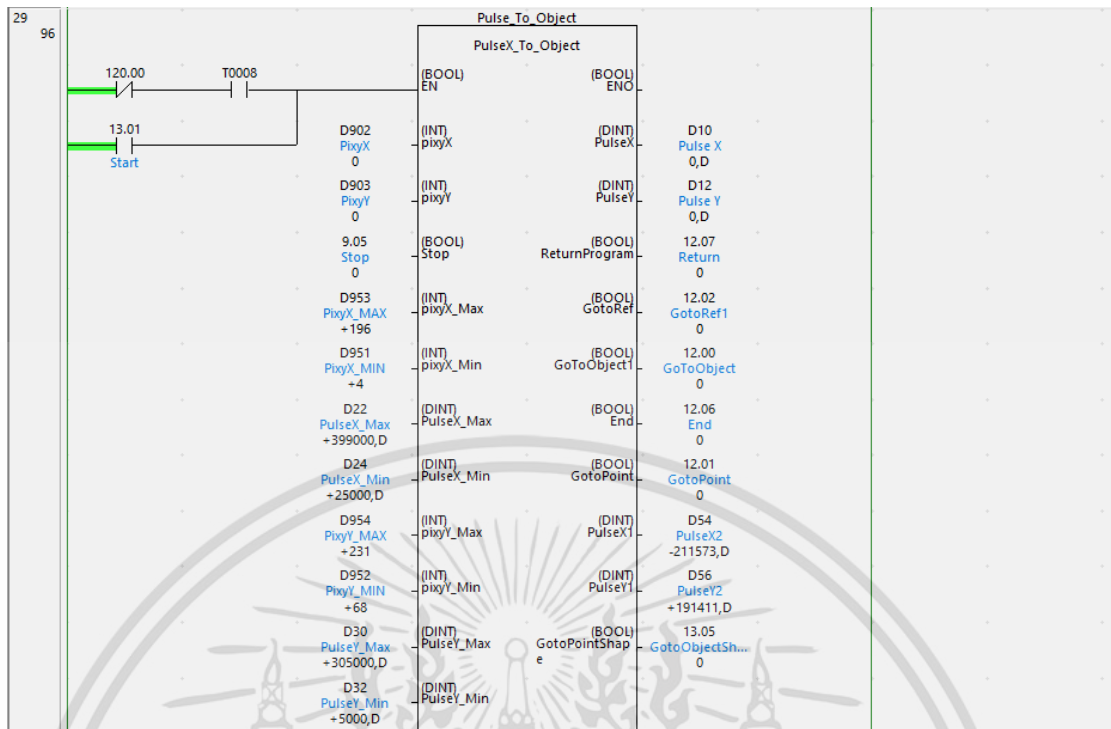
3.4.4.6 Function First In First Out จะนำค่าที่อยู่ใน Stack ตัวแรกออกมายังตำแหน่งที่เรากำหนด และนำข้อมูลในตำแหน่งต่อไปใน Stack ขึ้นมาแทนที่ในตำแหน่งแรกที่ถูกดึงไป และทำทุกครั้งที่ถูกกระตุ้นในแต่ละรอบ



รูปที่ 3.36 Function First In First Out

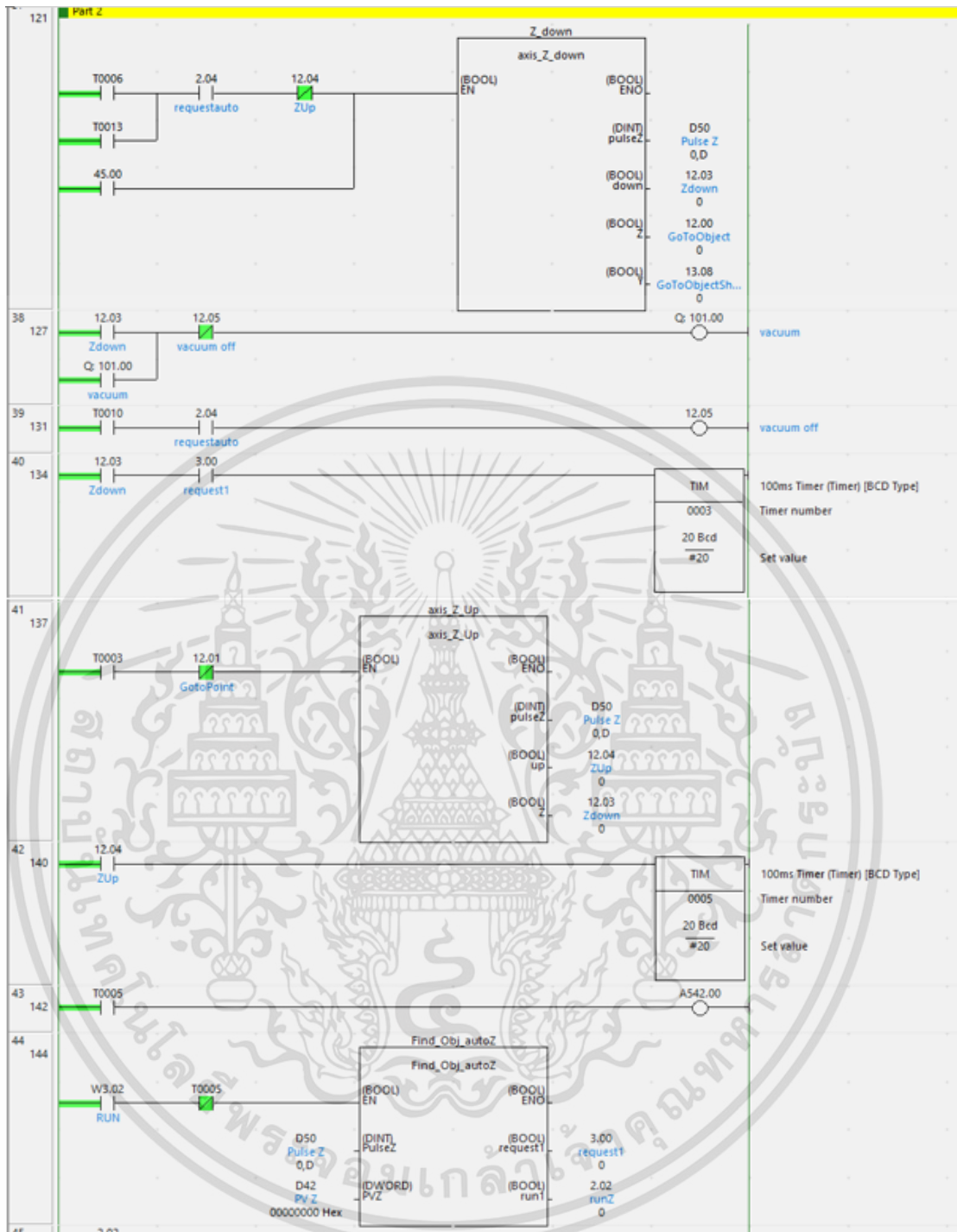
3.4.4.7 Function Block นี้จะเลือกโหมดการทำงานและนำค่าตำแหน่งของวัตถุที่ได้จาก Function First In First Out มาคำนวณเพื่อหา Pulse ทั้งสองแกน เพื่อให้เครื่องสามารถเคลื่อนที่ไปยังวัตถุชิ้นนั้นๆได้ ผ่านการทำงาน Function Block ของการเคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.37 Function Block Pulse to object

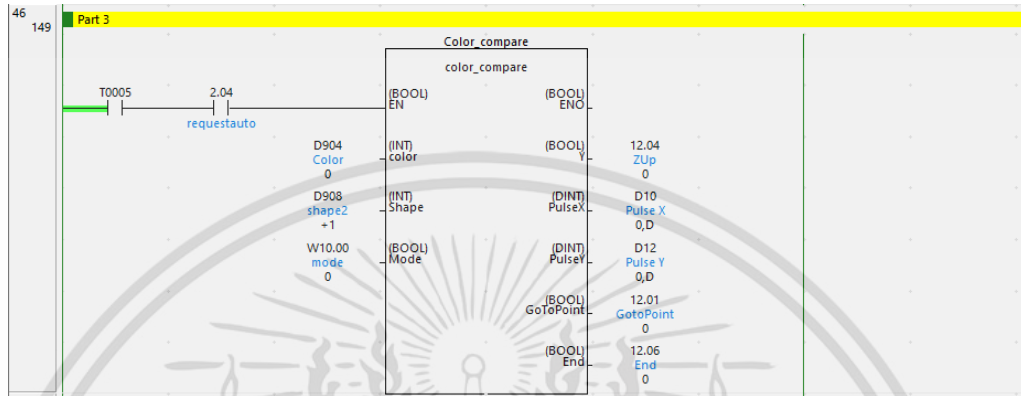
3.4.4.8 รูปด้านล่างคือกระบวนการเคลื่อนที่ในแกน Z จะเริ่มหลังจากจบการเคลื่อนที่ไปหาวัตถุ โดยจะเริ่มจากเปิดปั๊มลม และหัวจับจะเคลื่อนที่ลงไปจับวัตถุ และ เคลื่อนที่ขึ้นกลับมาที่เดิมและเข้าสู่กระบวนการต่อไป



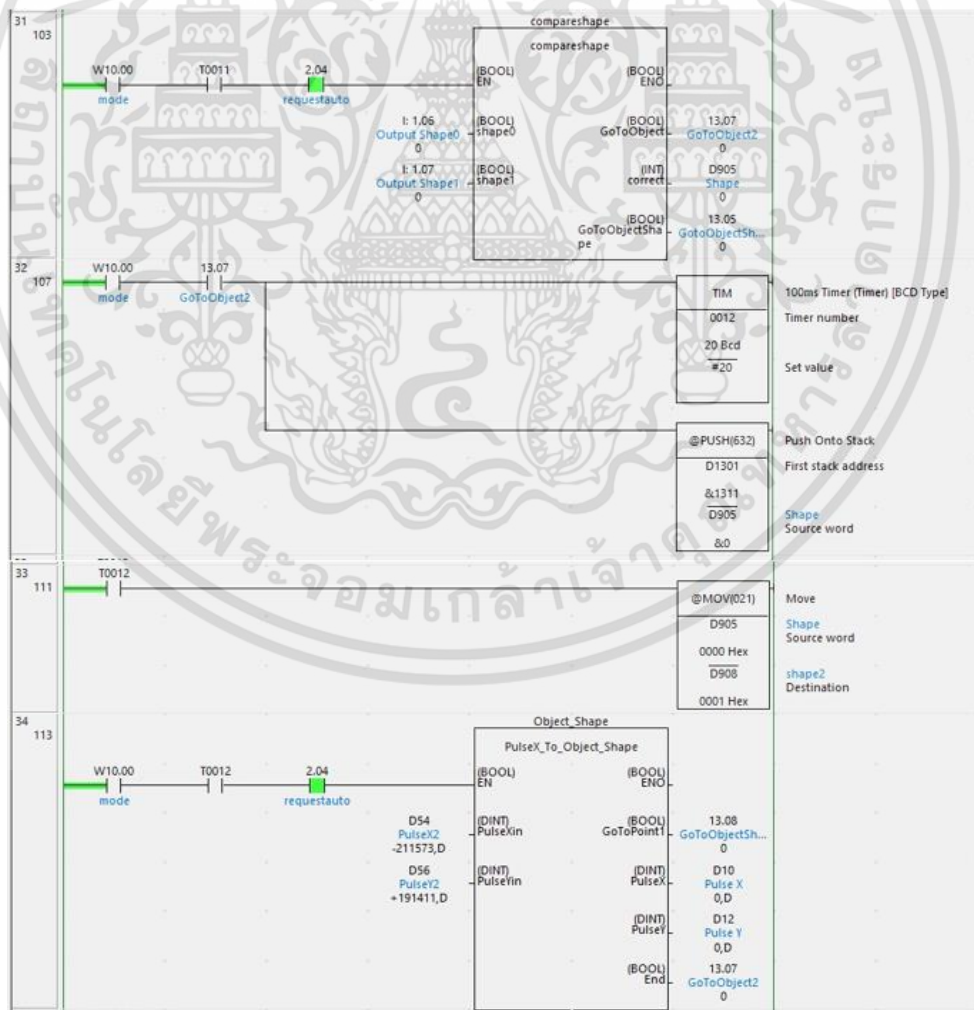
รูปที่ 3.38 Ladder ในการเคลื่อนที่ของแกน z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4.9 Function Box ในรูปที่ 3.39 (ก) เป็นการนำค่าสีของวัตถุมาคำนวณค่า Pulse เพื่อให้หัวจับเคลื่อนที่นำวัตถุไปเก็บตามตำแหน่งที่ถูกกำหนดไว้ ส่วนในรูปที่ 3.39 (ข) ถ้าเป็นการจำแนกวัตถุด้วยรูปร่าง โปรแกรมจะเข้ามาทำงานที่ส่วนนี้ก่อนที่จะไปทำงานที่ส่วนของการเคลื่อนที่ของแกน z



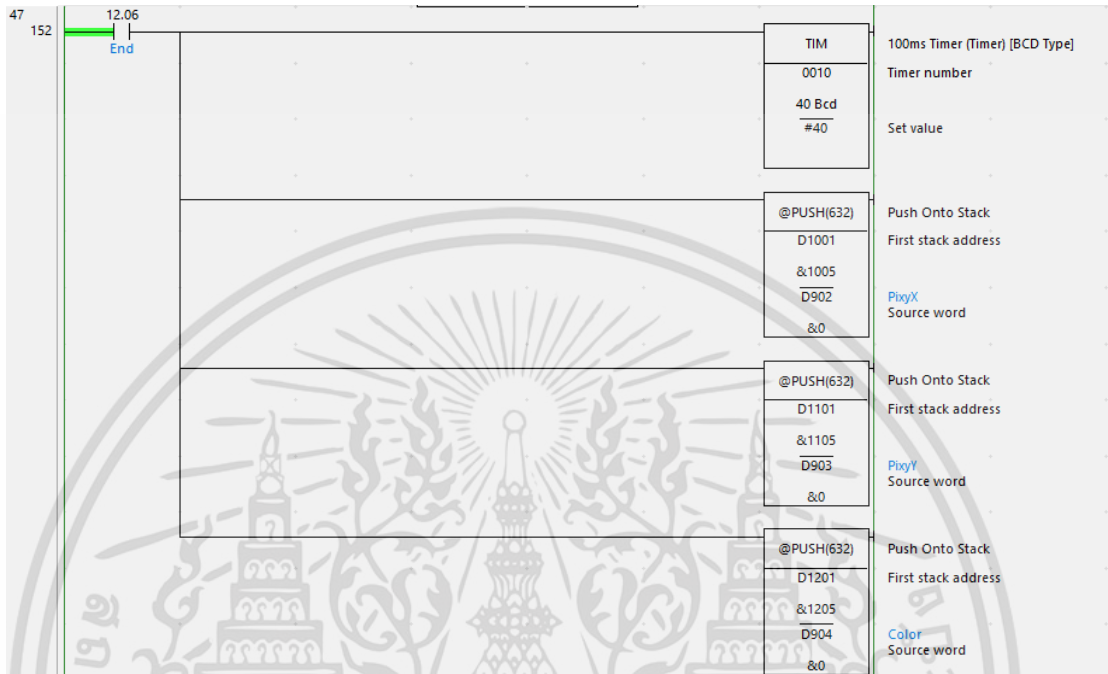
รูปที่ 3.39 Function Block ของการแยกวัตถุด้วยสี (ก)



รูปที่ 3.39 Function Block ของการแยกวัตถุด้วยรูปร่าง (ข)

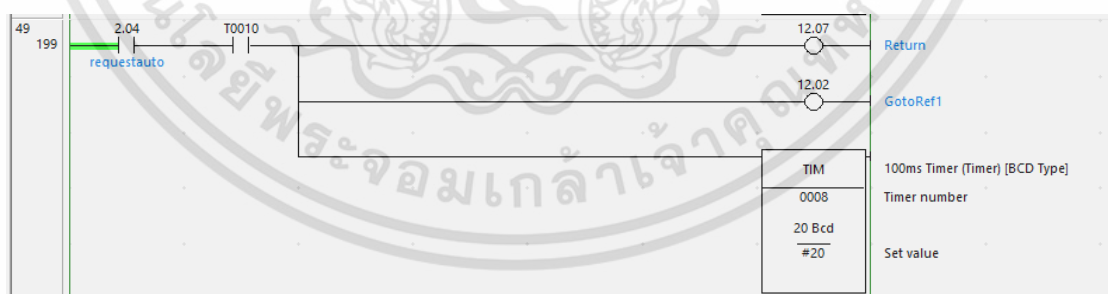
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4.10 Function นี้จะทำงานเมื่อจบกระบวนการการเคลื่อนที่เก็บวัตถุโดยจะนำค่าตำแหน่งของวัตถุทั้งสองแกน, สี และ รูปร่าง กลับเข้าไปเก็บยังพื้นที่ที่จองเอาไว้ (Stack) โดยเก็บที่ word ท้ายสุดของ Stack ตาม Data memory ที่กำหนดไว้



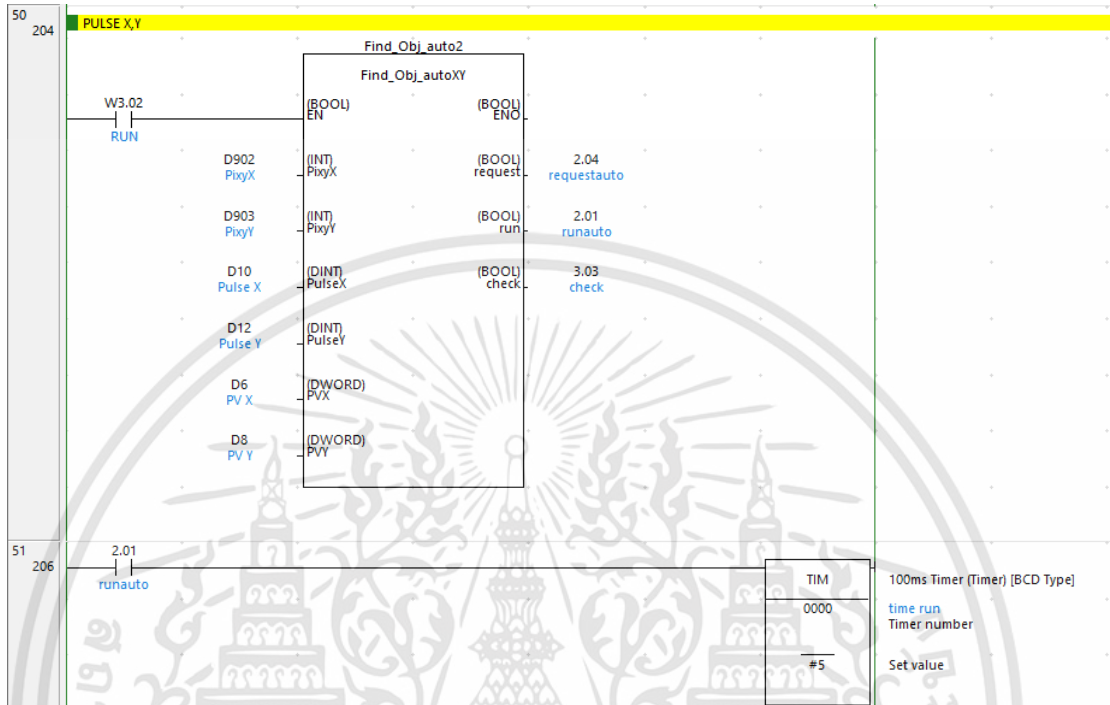
รูปที่ 3.40 Ladder ของการเก็บค่า

3.4.4.11 Function นี้จะทำการเริ่มระบบใหม่อีกครั้งตั้งแต่ต้นโปรแกรมของวัตถุขึ้นต่อไป



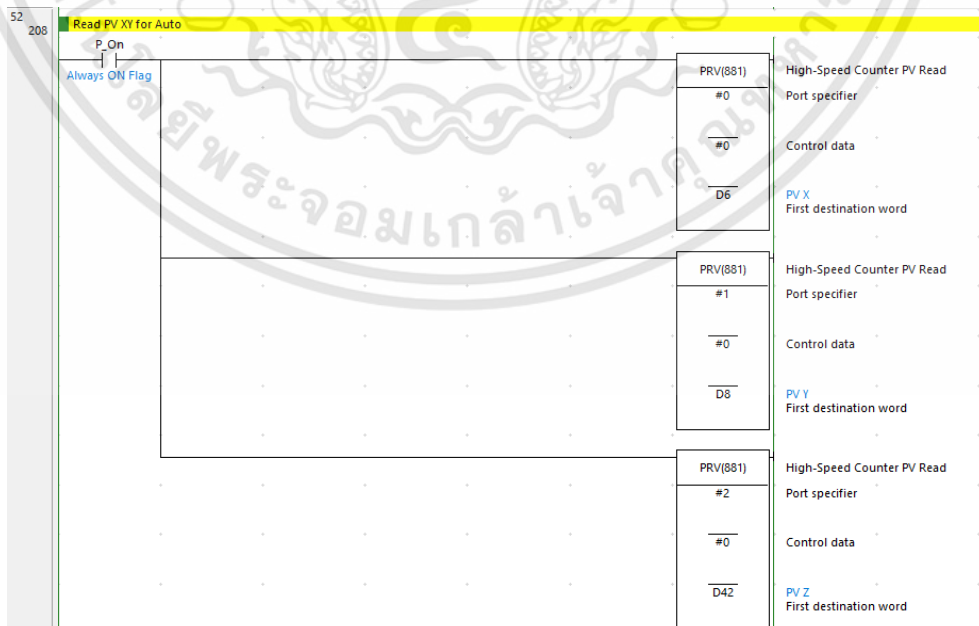
รูปที่ 3.41 Ladder ของการเริ่มระบบการทำงานใหม่

3.4.4.12 Function Block นี้จะเป็นตัวควบคุมการเคลื่อนที่ของ Servo แต่ละแกน ให้เคลื่อนที่ไปยังตำแหน่งหรือ Pulse ที่กำหนดและจะระบุว่าจะจับวัตถุเคลื่อนที่ไปถึงตำแหน่งที่กำหนดหรือยังเพื่อนำไปใช้ในส่วนของโปรแกรม



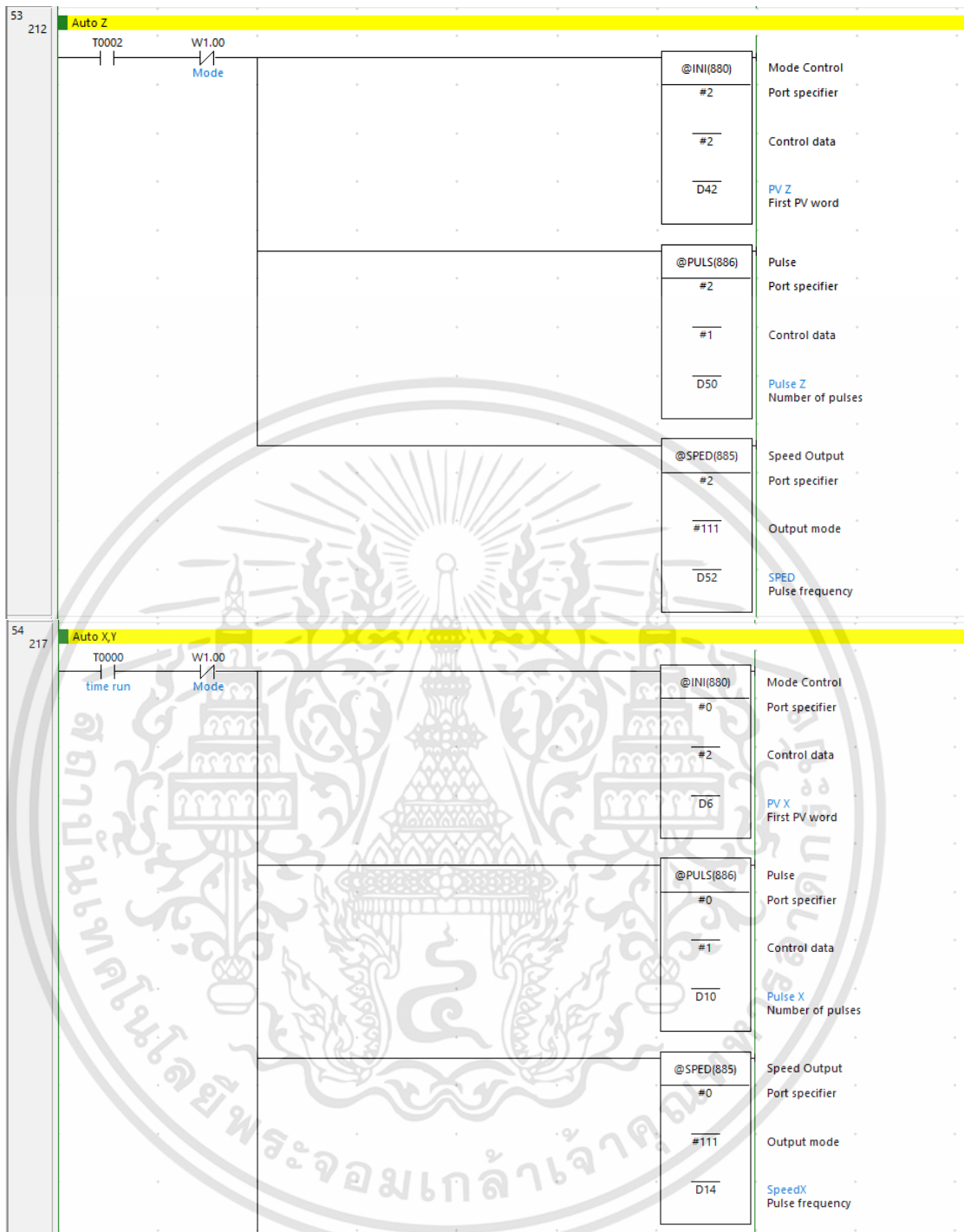
รูปที่ 3.42 Function Block ของการเคลื่อนที่ของ servo motor

3.4.4.13 Function Block เหล่านี้จะทำการคำนวณความเร็วของการเคลื่อนที่ของแต่ละแกน



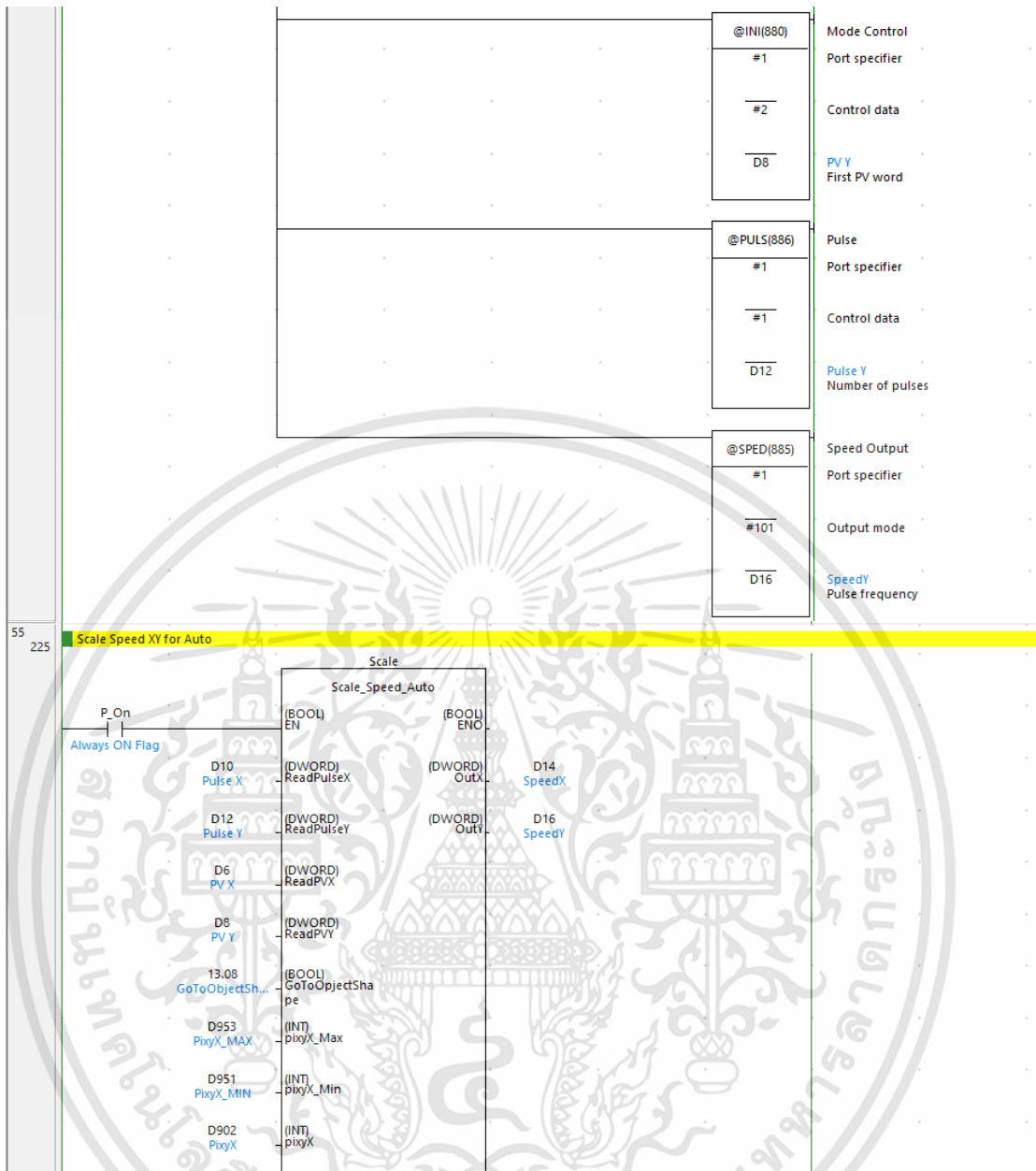
รูปที่ 3.43 Function Block ของการคำนวณความเร็วของการเคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.44 Function Block ของการคำนวณความเร็วของการเคลื่อนที่(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

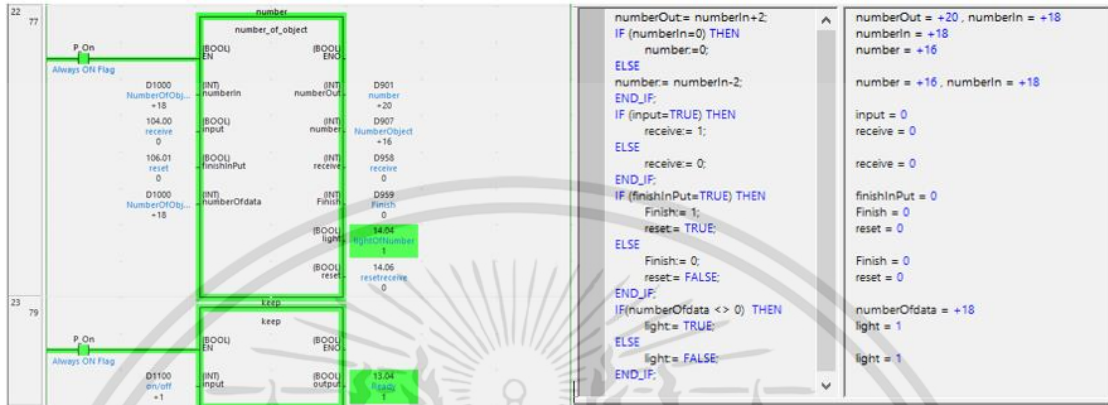


รูปที่ 3.45 Function Block ของการคำนวณความเร็วของการเคลื่อนที่(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

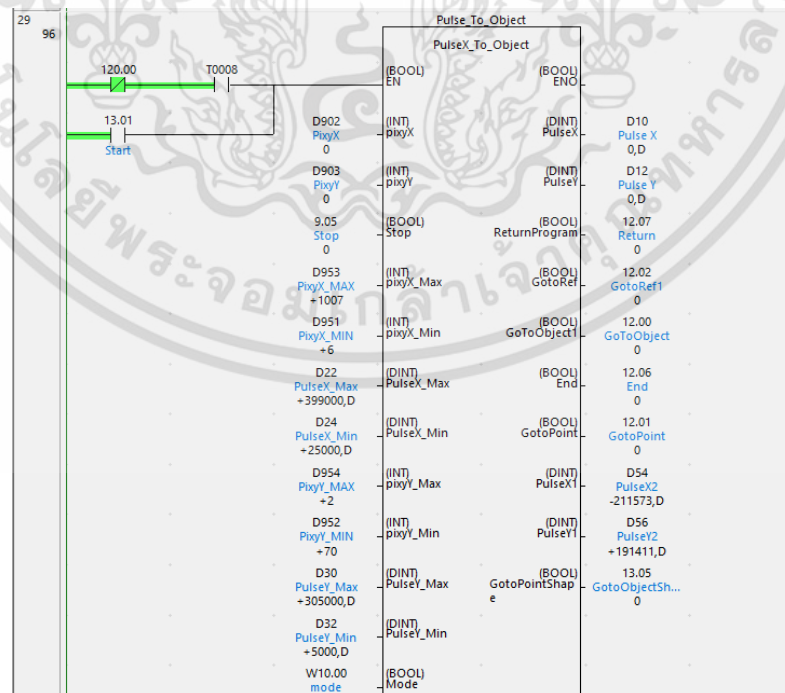
3.4.5 Function Block

3.4.5.1 รูปที่ 3.46 เป็น Function Block ที่ใช้ในการสั่งให้ Arduino ส่งค่าจำนวน และตำแหน่งของวัตถุมายัง PLC และสั่งให้จอแสดงผล HMI แสดงไฟว่าเครื่องพร้อมใช้งานหลังจากได้รับข้อมูลเหล่านั้นแล้ว



รูปที่ 3.46 Function Block number และ keep

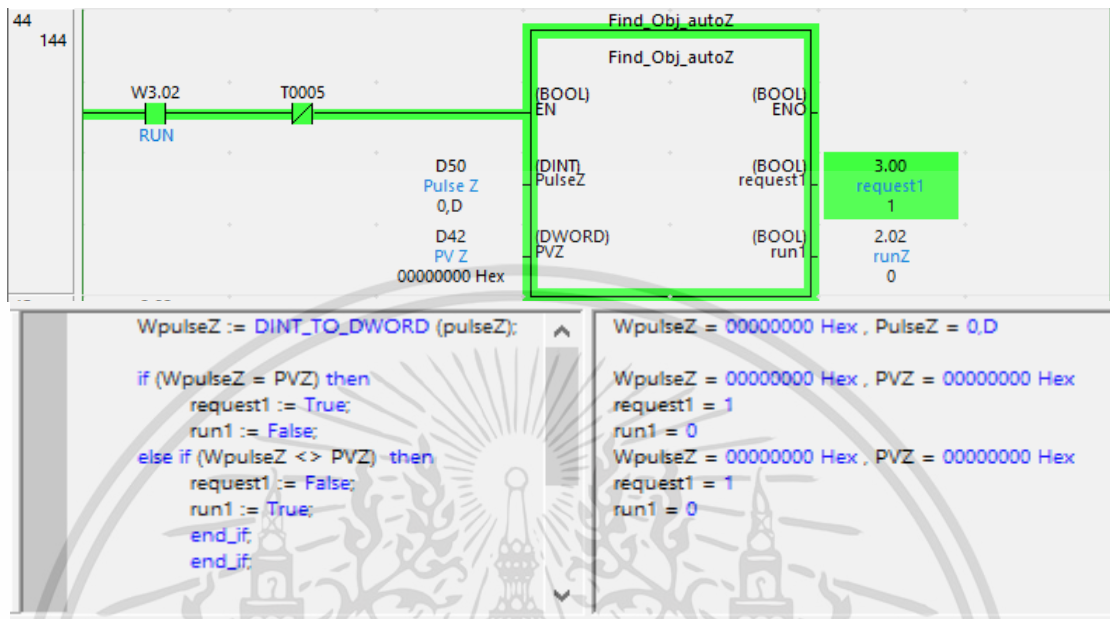
3.4.5.2 รูปที่ 3.47 เป็น Function Block ที่ใช้ในการนำค่าตำแหน่งของวัตถุมาทำการคำนวณหาค่า Pulse ในการเคลื่อนที่ไปยังตำแหน่งของวัตถุ และยังมีการแยกการคำนวณค่า Pulse ของแต่ละความยาวในแต่ละแกนอีกด้วยเพื่อความแม่นยำและถูกต้องของการเคลื่อนที่ (โดยเราจะแบ่งแกน X เป็น 7 โซน และแกน Y เป็น 6 โซน)



รูปที่ 3.47 Function Block Pulse_To_Object

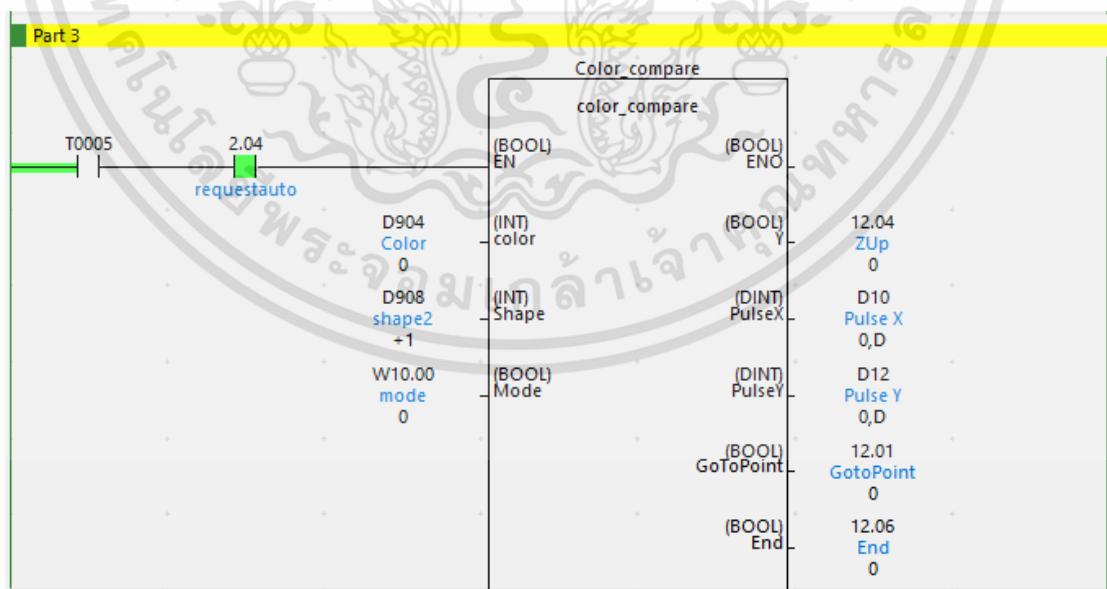
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.5.4 รูปที่ 3.50 เป็น Function Block ที่ใช้ในการควบคุมการเคลื่อนที่ในแกนตั้ง (แกน Z)



รูปที่ 3.50 Function Block Find_Obj_autoZ

3.4.5.5 รูปที่ 3.51 เป็น Function Block ที่ทำการให้ค่า Pulse สำหรับการเคลื่อนที่เพื่อนำวัตถุไปเก็บตามลักษณะสี หรือรูปร่างของวัตถุตามโหมดที่เลือก ตามพื้นที่ที่กำหนดไว้



รูปที่ 3.51 Function Block Color_compare

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

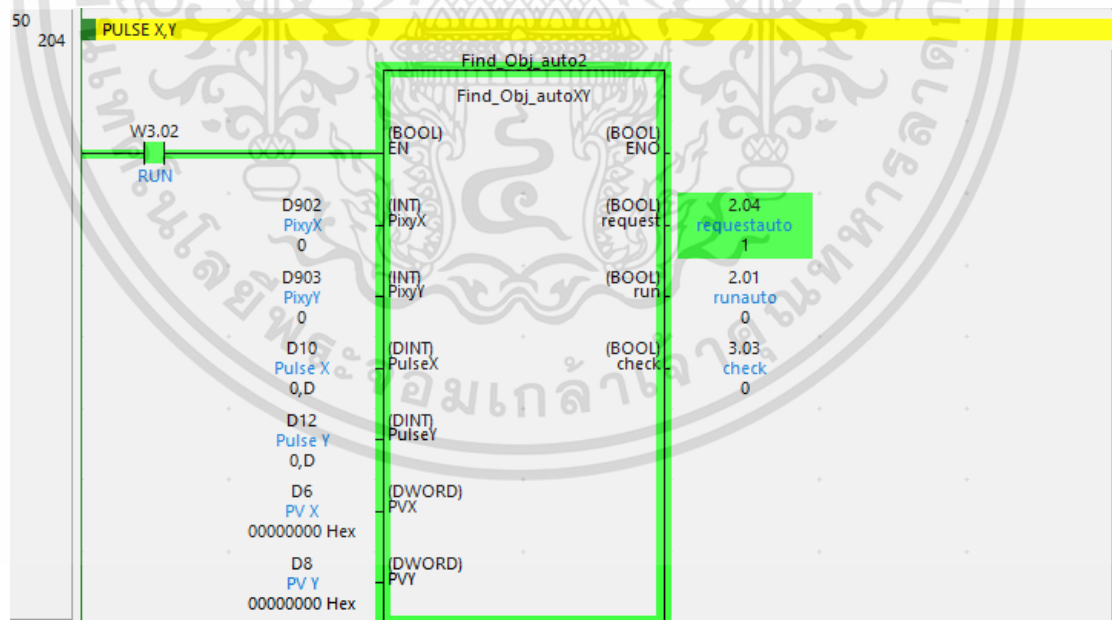
```

IF (Mode = FALSE)THEN
IF(color = 1)THEN
PulseX := 0 ;
PulseY := 259000 ;
GoToPoint= TRUE;
End:= TRUE;
Y:= FALSE;
ELSE IF (color = 2)THEN
PulseX :=0 ;
PulseY :=190000 ;
GoToPoint= TRUE;
End:= TRUE;
Y:= FALSE;
ELSE IF (color = 3)THEN
PulseX := 0 ;
PulseY := 124000 ;
GoToPoint= TRUE;
End:= TRUE;
Y:= FALSE;
ELSE IF (color = 4)THEN
PulseX := 0 ;
PulseY := 54000 ;
GoToPoint= TRUE;
End:= TRUE;
Y:= FALSE;
END_IF;
END_IF;
END_IF;
END_IF;
Mode = 0
color = 0
PulseX = 0,D
PulseY = 0,D
GoToPoint = 0
End = 0
Y = 0
color = 0
PulseX = 0,D
PulseY = 0,D
GoToPoint = 0
End = 0
Y = 0
color = 0
PulseX = 0,D
PulseY = 0,D
GoToPoint = 0
End = 0
Y = 0
color = 0
PulseX = 0,D
PulseY = 0,D
GoToPoint = 0
End = 0
Y = 0

```

รูปที่ 3.52 Function Block Color_compare (ต่อ)

3.4.5.6 รูปที่ 3.53 เป็น Function Block ที่ใช้ในการควบคุมการเคลื่อนที่ในแกน X และแกน Y



รูปที่ 3.53 Function Block Find_Obj_auto2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

WpulseX := DINT_TO_DWORD (pulseX);
WpulseY := DINT_TO_DWORD (pulseY);

if ((WpulseX=PvX)AND(WpulseY=PvY)) THEN
  request := True;
  run := False;
ELSE IF((WpulseX=0)AND(PulseY=0))THEN
  request := False;
  run := True;
ELSE IF ((WpulseX <> PvX) AND (WpulseY <> PvY)) then
  request := False;
  run := True;
else if ((WpulseX <> PvX) OR (WpulseY <> PvY)) then
  request := False;
  run := True;
if (((WpulseX=PvX)AND(WpulseY<>PvY))) THEN
  request := True;
  run := False;
end_if;
end_if;
END_IF;
end_if;
END_IF;

```

```

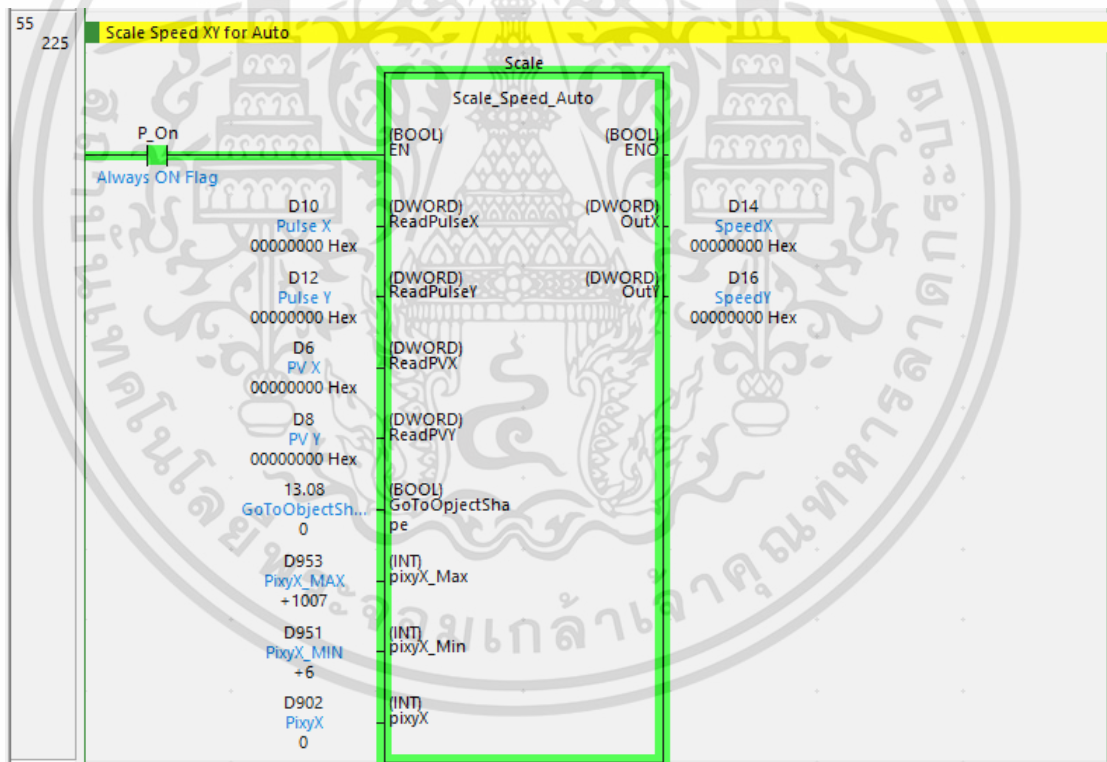
WpulseX = 00000000 Hex , PulseX = 0.D
WpulseY = 00000000 Hex , PulseY = 0.D

WpulseX = 00000000 Hex , PvX = 00000000 Hex , WpulseY = 00000000 Hex , PvY = 00000000 Hex
request = 1
run = 0
PulseX = 0.D , PulseY = 0.D
request = 1
run = 0
WpulseX = 00000000 Hex , PvX = 00000000 Hex , WpulseY = 00000000 Hex , PvY = 00000000 Hex
request = 1
run = 0
WpulseX = 00000000 Hex , PvX = 00000000 Hex , WpulseY = 00000000 Hex , PvY = 00000000 Hex
request = 1
run = 0
WpulseX = 00000000 Hex , PvX = 00000000 Hex , WpulseY = 00000000 Hex , PvY = 00000000 Hex
request = 1
run = 0

```

รูปที่ 3.54 Function Block Find_Obj_auto2 (ต่อ)

3.4.5.7 รูปที่ 3.55 เป็น Function Block ที่จะคำนวณความเร็วในการเคลื่อนที่ในแต่ละแกนตามเวลาที่กำหนด



รูปที่ 3.55 Function Block Scale

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<pre> PulseX := DWORD_to_DINT(ReadPulseX); PulseY := DWORD_to_DINT(ReadPulseY); PVX := DWORD_to_DINT(ReadPVX); PVY := DWORD_to_DINT(ReadPVY); XXX:= ABS(pixyX_Max-pixyX_Min)/7; IF(GoToObjectShape = TRUE)THEN SpeedX := ABS(PulseX-PVX) / 2; SpeedY := ABS(PulseY-PVY) / 2; ELSE IF (GoToObjectShape=FALSE) THEN IF (pixyX_Min<pixyX)AND(pixyX<(XXX*3)) THEN SpeedX := ABS(PulseX-PVX) / 6; SpeedY := ABS(PulseY-PVY) / 6; ELSE IF ((XXX*3)<=pixyX)AND(pixyX<(XXX*5)) THEN SpeedX := ABS(PulseX-PVX) / 4; SpeedY := ABS(PulseY-PVY) / 4; ELSE IF ((XXX*5)<=pixyX)AND(pixyX<pixyX_Max) THEN SpeedX := ABS(PulseX-PVX) / 3; SpeedY := ABS(PulseY-PVY) / 3; ELSE SpeedX := ABS(PulseX-PVX) / 5; SpeedY := ABS(PulseY-PVY) / 5; END_IF; END_IF; END_IF; OutX := DINT_to_DWORD(SpeedX); OutY:=DINT_to_DWORD(SpeedY); </pre>	<pre> PulseX = 0.D , ReadPulseX = 00000000 Hex PulseY = 0.D , ReadPulseY = 00000000 Hex PVX = 0.D , ReadPVX = 00000000 Hex PVY = 0.D , ReadPVY = 00000000 Hex XXX = +143 , pixyX_Max = +1007 , pixyX_Min = +6 GoToObjectShape = 0 SpeedX = 0.D , PulseX = 0.D , PVX = 0.D SpeedY = 0.D , PulseY = 0.D , PVY = 0.D GoToObjectShape = 0 pixyX_Min = +6 , pixyX = 0 , XXX = +143 SpeedX = 0.D , PulseX = 0.D , PVX = 0.D SpeedY = 0.D , PulseY = 0.D , PVY = 0.D XXX = +143 , pixyX = 0 SpeedX = 0.D , PulseX = 0.D , PVX = 0.D SpeedY = 0.D , PulseY = 0.D , PVY = 0.D XXX = +143 , pixyX = 0 , pixyX_Max = +1007 SpeedX = 0.D , PulseX = 0.D , PVX = 0.D SpeedY = 0.D , PulseY = 0.D , PVY = 0.D SpeedX = 0.D , PulseX = 0.D , PVX = 0.D SpeedY = 0.D , PulseY = 0.D , PVY = 0.D OutX = 00000000 Hex , SpeedX = 0.D OutY = 00000000 Hex , SpeedY = 0.D </pre>
--	--

รูปที่ 3.56 Function Block Scale (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

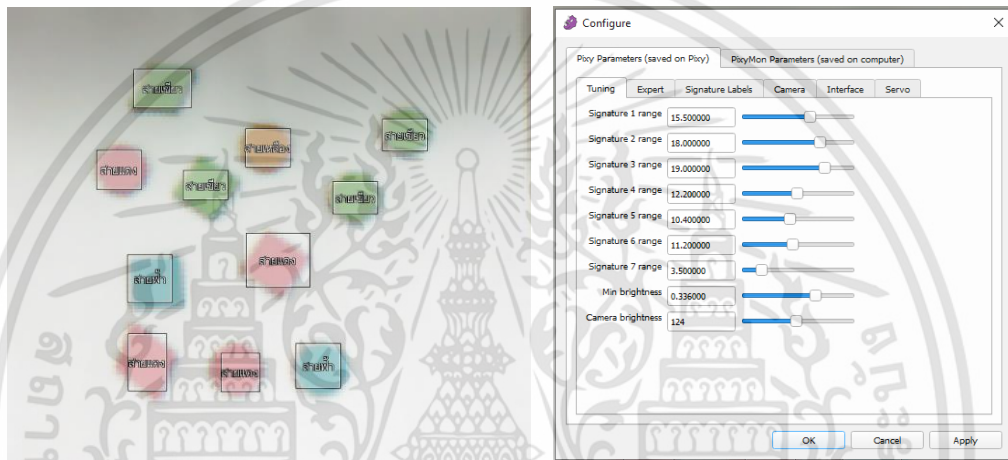
บทที่ 4

ผลการทดลอง

4.1 ผลการทดลองเกี่ยวกับการหาตำแหน่งและสีของวัตถุ

4.1.1 การหาตำแหน่ง

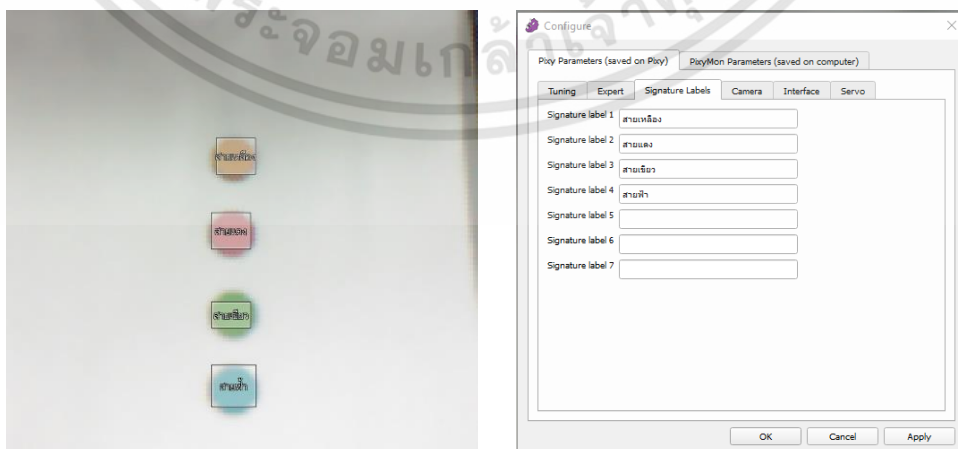
จากการเขียนโปรแกรมตรวจจับตำแหน่งของวัตถุ ในโปรแกรม PixyMon ได้ผลการทดลองดังรูปต่อไปนี้



รูปที่ 4.1 ผลการตรวจจับตำแหน่งของวัตถุ

4.1.2 การตรวจจับสีของวัตถุ

จากการเขียนโปรแกรมตรวจจับสีของวัตถุ ในโปรแกรม PixyMon ได้ผลการทดลองดังรูปต่อไปนี้



รูปที่ 4.2 ผลการตรวจจับสีของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

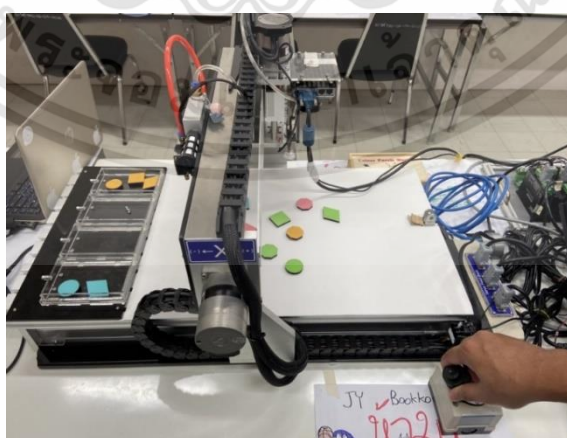
สีของวัตถุ	การทดลองครั้งที่	ผลการวิเคราะห์ จำแนกสี
เหลือง	1	เหลือง
	2	เหลือง
	3	เหลือง
ชมพู	1	ชมพู
	2	ชมพู
	3	ชมพู
เขียว	1	เขียว
	2	เขียวตัด
	3	เขียว
ฟ้า	1	ฟ้า
	2	ฟ้า
	3	ฟ้า

ตารางที่ 4.1 แสดงผลการทดสอบความถูกต้องในการวิเคราะห์และจำแนกวัตถุด้วยสี

4.2 ผลการทดลองเกี่ยวกับการเคลื่อนที่

4.2.1 การควบคุมโดยใช้ Joy Stick

จากการเขียนโปรแกรมควบคุมการเคลื่อนที่ของเซอร์โวมอเตอร์โดยใช้คั่นบังคับควบคุมด้วยมือเพื่อศึกษาการเคลื่อนที่ของ CNC ได้ผลการทดลองดังรูป



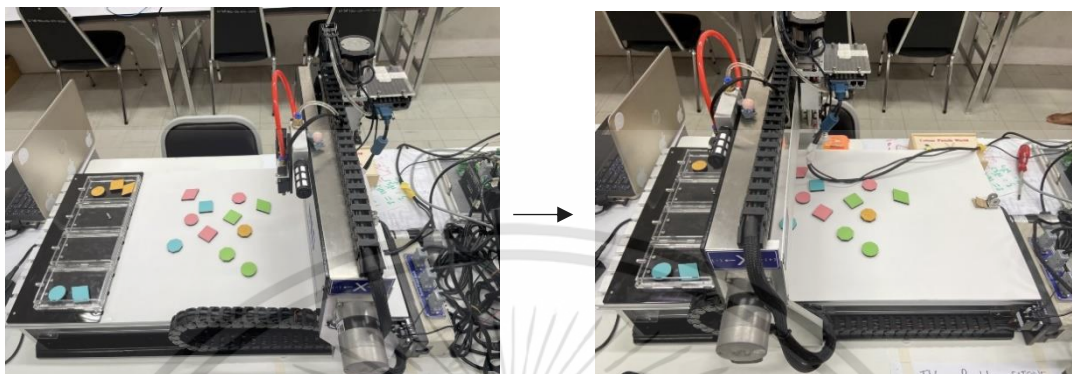
← ทิศทางการเคลื่อนที่

รูปที่ 4.3 แสดงผลการควบคุมโดยใช้ Joy Stick

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การควบคุมแบบอัตโนมัติ

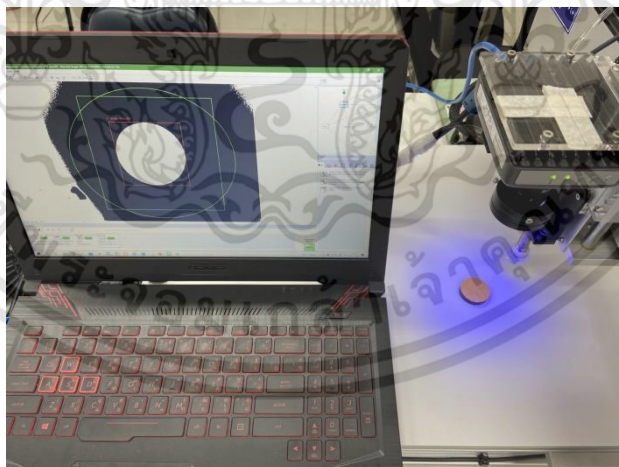
จากการเขียนโปรแกรมควบคุมการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ใช้การควบคุมแบบอัตโนมัติ โดยทำการเขียนโปรแกรมภายในให้ PLC



รูปที่ 4.4 แสดงผลการควบคุมแบบอัตโนมัติ

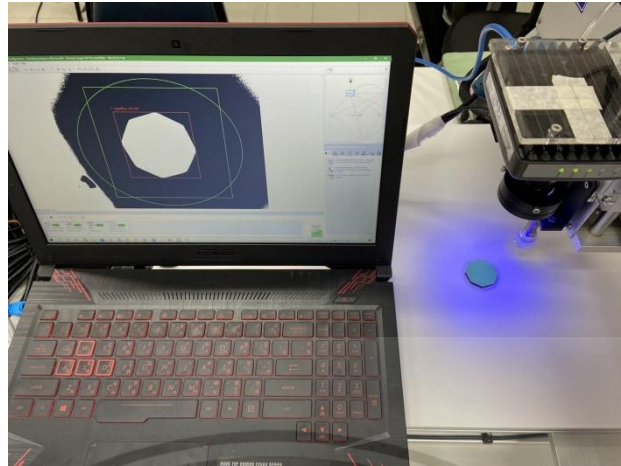
4.3 ผลการทดลองเกี่ยวกับระบบการวิเคราะห์และจำแนกวัตถุด้วยรูปร่าง

จากการเขียนโปรแกรมเพื่อวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder AI โดยใช้กล้อง Smart Camera NI 1732 ทำการจำแนกวัตถุและแสดงผลออกทางหน้าจอโปรแกรม NI Vision Builder AI ดังภาพด้านล่างต่อไปนี้

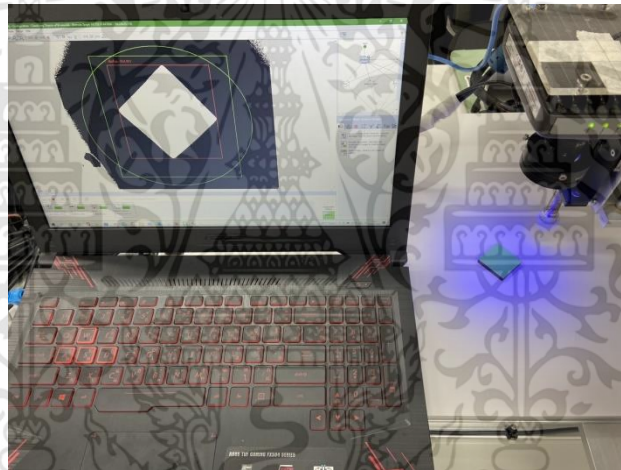


รูปที่ 4.5 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และแสดงผลเป็นวงกลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

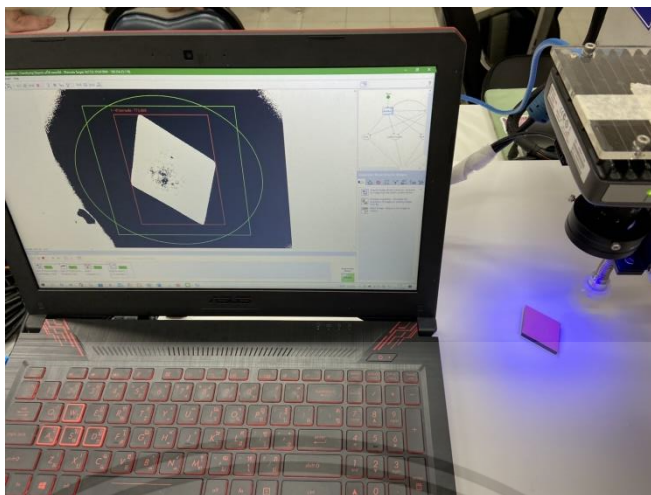


รูปที่ 4.6 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และแสดงผลเป็นแปดเหลี่ยม



รูปที่ 4.7 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และแสดงผลเป็นสี่เหลี่ยมจัตุรัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แสดงผลการวิเคราะห์ จำแนกวัตถุด้วยโปรแกรม NI Vision Builder และแสดงผลเป็น สีเหลี่ยมข้าวหลามตัด

เมื่อทำการทดลองโดยการนำวัตถุทรงกลม, สีเหลี่ยมจัตุรัส, สีเหลี่ยมข้าวหลามตัด และแปดเหลี่ยม ไปไว้ในกรอบภาพของ Smart Camera เพื่อทดสอบความถูกต้องในการวิเคราะห์ จำแนกวัตถุ และจะได้ผลการทดลองดังนี้

รูปทรงของวัตถุ	การทดลองครั้งที่	ผลการวิเคราะห์ จำแนกวัตถุ
วงกลม	1	วงกลม
	2	วงกลม
	3	วงกลม
สีเหลี่ยมจัตุรัส	1	สีเหลี่ยมจัตุรัส
	2	สีเหลี่ยมจัตุรัส
	3	สีเหลี่ยมจัตุรัส
สีเหลี่ยมข้าวหลามตัด	1	สีเหลี่ยมข้าวหลามตัด
	2	สีเหลี่ยมข้าวหลามตัด
	3	สีเหลี่ยมข้าวหลามตัด
แปดเหลี่ยม	1	แปดเหลี่ยม
	2	แปดเหลี่ยม
	3	แปดเหลี่ยม

ตารางที่ 4.2 แสดงผลการทดสอบความถูกต้องในการวิเคราะห์ จำแนกวัตถุ

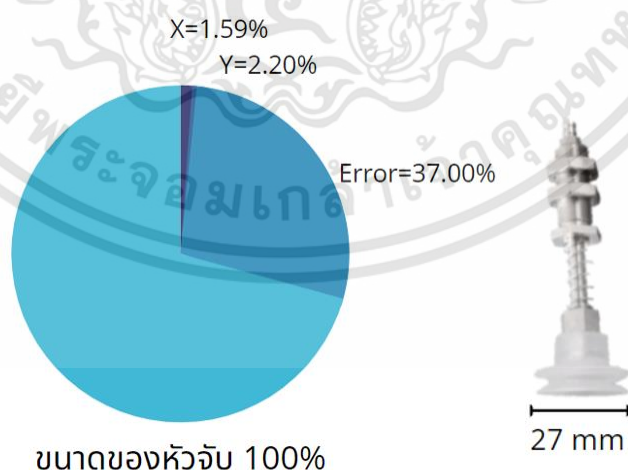
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ผลการทดลองเกี่ยวกับการจัดเก็บวัตถุ

จากการทดลองนำวัตถุไปจัดเก็บ ตำแหน่งละ 6 ครั้งจาก 7 ตำแหน่ง โดยถ้าผลรวมเปอร์เซ็นต์ความคลาดเคลื่อนรวมเกิน 37 % หรือ 10 มิลลิเมตร จะทำให้หัว vacuum ไม่สามารถดูดจับวัตถุขึ้นมาได้

ครั้งที่		1	2	3	4	5	6	ค่าเฉลี่ยของความคลาดเคลื่อน	ความกว้างหัวจับวัตถุ	คิดเป็นเปอร์เซ็นต์	
กลาง	X	1	0	1	0	0	1	0.50	27	1.85%	
	Y	0	0	0	0	1	0	0.17	27	0.62%	
รวม		1	0	1	0	1	1	0.67	54	1.23%	
มุมล่างขวา	X	0	0	0	0	1	1	0.33	27	1.23%	
	Y	0	0	0	0	0	0	0.00	27	0.00%	
รวม		0	0	0	0	1	1	0.33	54	0.62%	
บนซ้าย	X	2	3	0	0	0	0	0.83	27	3.09%	
	Y	0	1	3	3	2	2	1.83	27	6.79%	
รวม		2	4	3	3	2	2	2.67	54	4.94%	
บนขวา	X	0	0	0	0	0	0	0.00	27	0.00%	
	Y	0	0	3	0	0	0	0.50	27	1.85%	
รวม		0	0	3	0	0	0	0.50	54	0.93%	
มุมล่างซ้าย	X	0	0	0	2	2	1	0.83	27	3.09%	
	Y	1	0	0	0	0	0	0.17	27	0.62%	
รวม		1	0	0	2	2	1	1.00	54	1.85%	
ตำแหน่งใดๆ1	X	0	0	0	0	0	0	0.00	27	0.00%	
	Y	3	2	0	0	0	1	1.00	27	3.70%	
รวม		3	2	0	0	0	1	1.00	54	1.85%	
ตำแหน่งใดๆ2	X	0	1	0	1	1	0	0.50	27	1.85%	
	Y	0	1	0	1	1	0	0.50	27	1.85%	
รวม		0	2	0	2	2	0	1.00	54	1.85%	
									รวมแกนX	1.59%	จากการทดลอง 42 ครั้ง
									รวมแกนY	2.20%	จากการทดลอง 42 ครั้ง
									รวมสองแกน	1.90%	จากการทดลอง 42 ครั้ง

ตารางที่ 4.3 แสดงผลการทดลองเกี่ยวกับการจัดเก็บวัตถุ



รูปที่ 4.9 กราฟแสดงความแม่นยำของหัวจับวัตถุ และขนาดของหัวจับวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจากการทดลอง 42 ครั้ง

เส้นผ่านศูนย์กลางของหัวจับวัตถุ(100%)	= 27 mm
ค่าความคลาดเคลื่อนของหัวจับที่แกน X	= 1.59% (0.43 mm)
ค่าความคลาดเคลื่อนของหัวจับที่แกน Y	= 2.20% (0.59 mm)
ค่าความคลาดเคลื่อนของหัวจับที่ไม่สามารถจับวัตถุได้	=37.00% (10 mm)

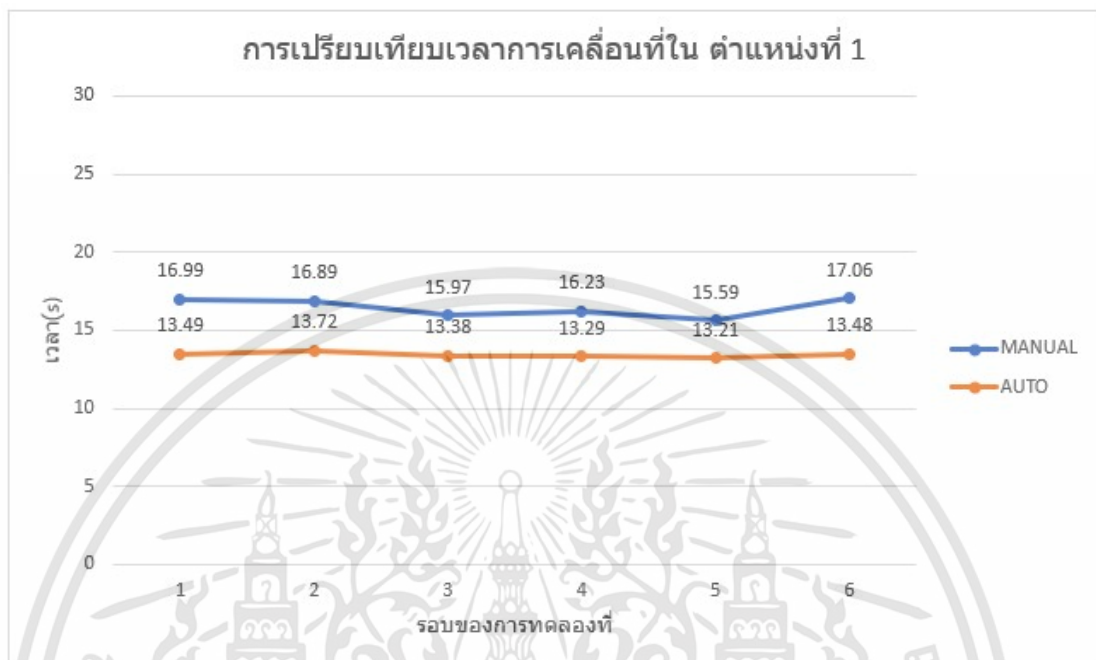
4.5 ผลการทดลองเกี่ยวกับการจัดเก็บวัตถุเทียบกับเวลา

ตำแหน่งและโหมดการเคลื่อนที่		1	2	3	4	5	6	avg	MAX	MIN	% การเปลี่ยนแปลง	
ตำแหน่งที่ 1	MANUAL	16.99	16.89	15.97	16.23	15.59	17.06	16.46	17.06	15.59	diff	3.03
	AUTO	13.49	13.72	13.38	13.29	13.21	13.48	13.43	13.72	13.21	diff percent	18.39%
ตำแหน่งที่ 2	MANUAL	18.39	22.35	18.53	21.51	18.51	19.05	19.72	22.35	18.39	diff	4.59
	AUTO	15.16	15.14	15.07	15.13	15.2	15.11	15.14	15.20	15.07	diff percent	23.26%
ตำแหน่งที่ 3	MANUAL	23.15	23.42	23.03	23.9	23.2	24.21	23.49	24.21	23.03	diff	4.23
	AUTO	19.32	19.3	19.16	19.21	19.27	19.26	19.25	19.32	19.16	diff percent	18.02%
3 objects	MANUAL	54.79	58.11	56.94	56.05	56.89	57.74	56.75	58.11	54.79	diff	10.05
	AUTO	46.81	46.82	46.48	46.9	46.52	46.7	46.71	46.90	46.48	diff percent	17.71%
5 objects	AUTO Color Mode	78.48	78.47	78.41	78.63	78.77	78.64	78.57	78.77	78.41	diff	0.36
											diff percent	0.46%
5 objects	AUTO Shape Mode	119.32	119.37	119.37	119.48	119.46	119.35	119.39	119.48	119.32	diff	0.16
											diff percent	0.13%

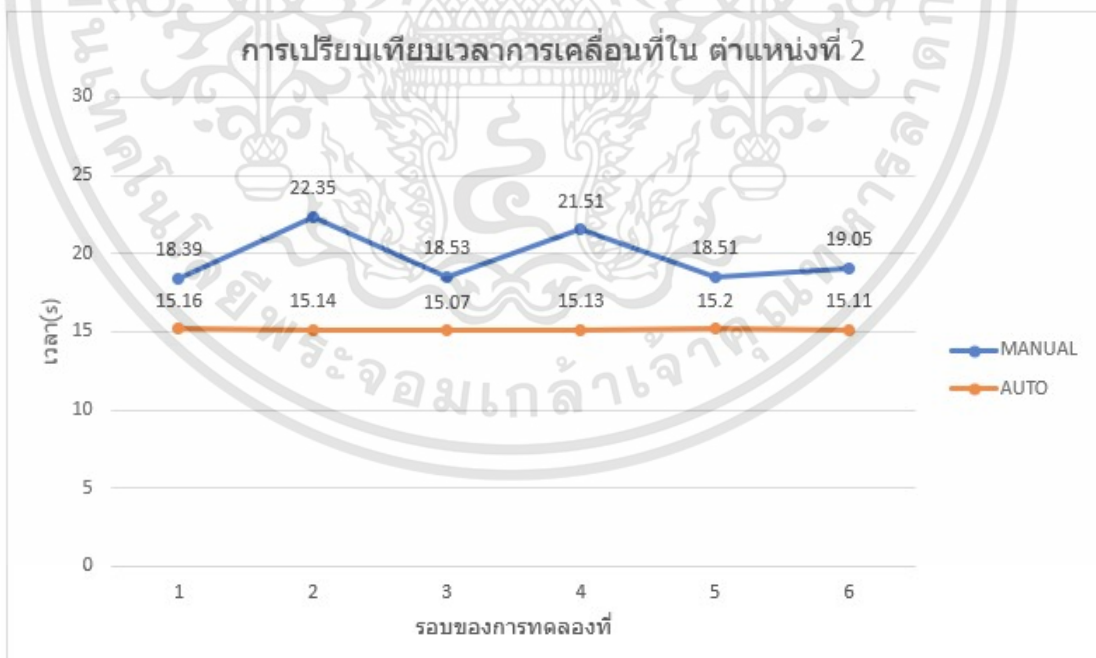
ตารางที่ 4.4 ตารางเปรียบเทียบเวลาที่ใช้ในการเคลื่อนที่ของโหมด Manual และโหมด Auto

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางด้านบนสามารถนำมาทำเป็นกราฟเพื่อแสดงการเปรียบเทียบการเคลื่อนที่ในรูปแบบต่างๆเทียบกับเวลาได้ดังต่อไปนี้

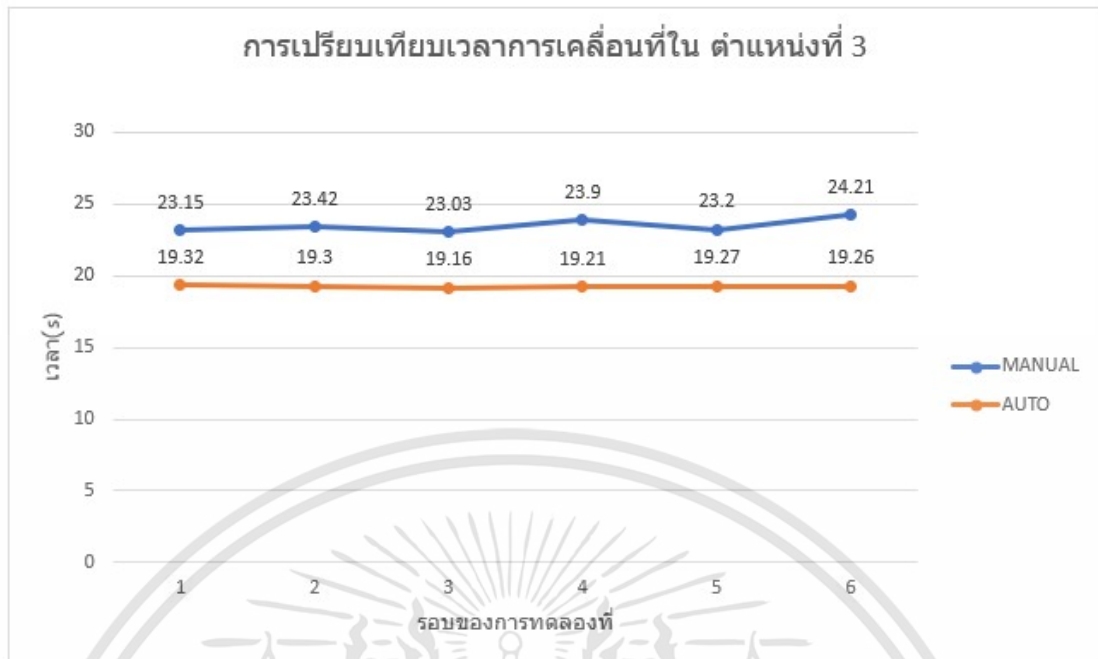


รูปที่ 4.10 กราฟการเปรียบเทียบเวลาการเคลื่อนที่ในตำแหน่งที่ 1

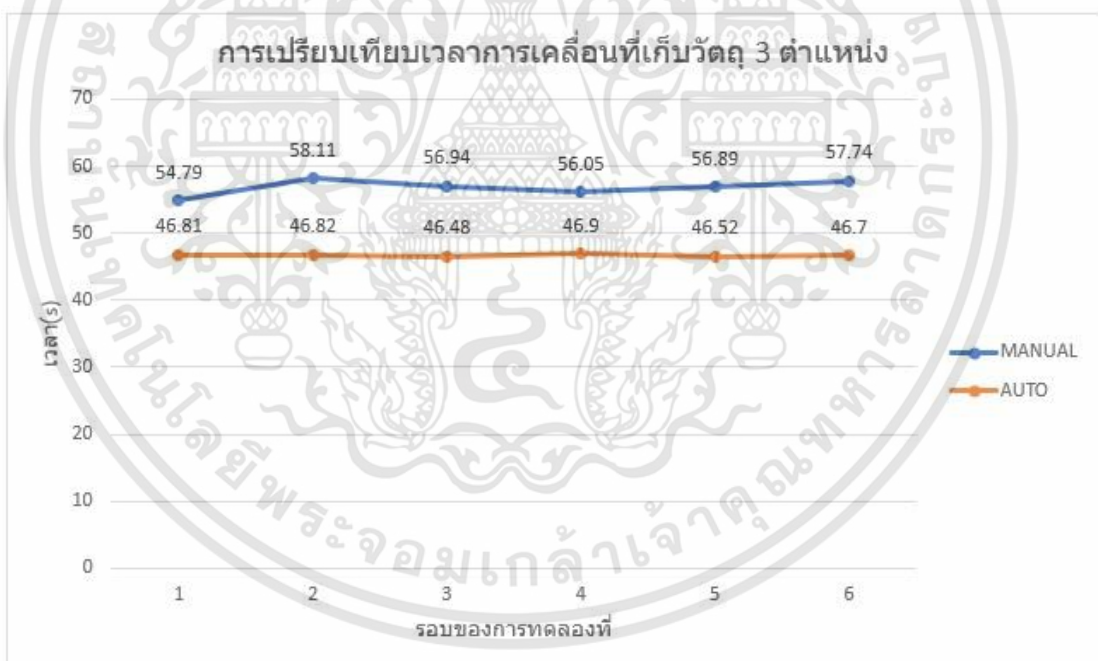


รูปที่ 4.11 กราฟการเปรียบเทียบเวลาการเคลื่อนที่ในตำแหน่งที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

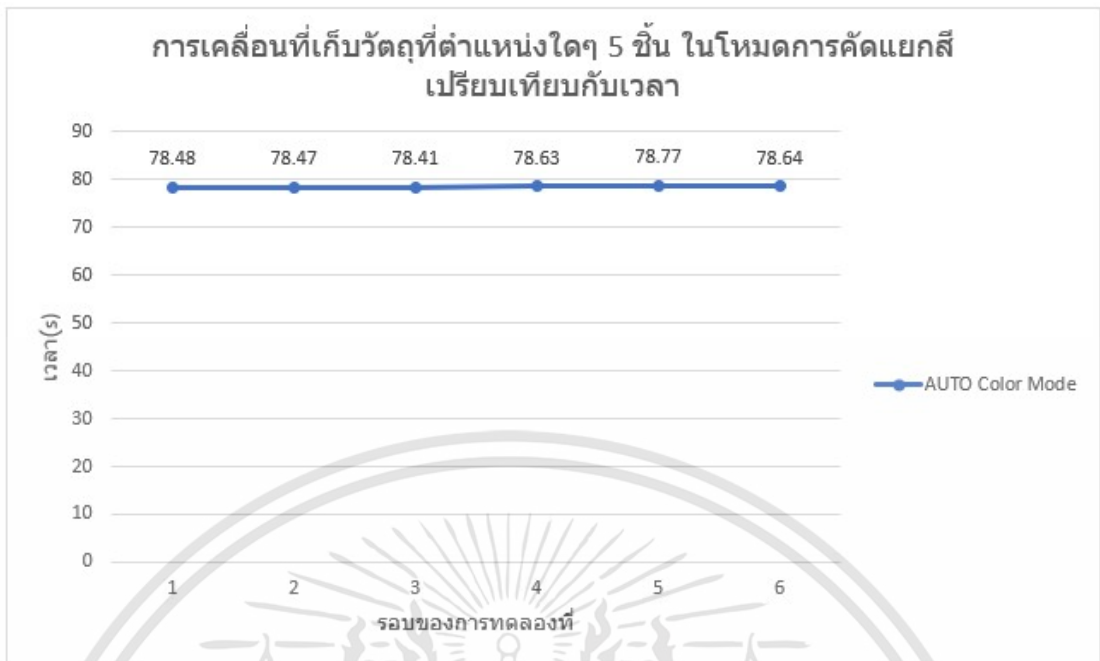


รูปที่ 4.12 กราฟการเปรียบเทียบเวลาการเคลื่อนที่ในตำแหน่งที่ 3



รูปที่ 4.13 กราฟการเปรียบเทียบเวลาการเคลื่อนที่เก็บวัตถุ 3 ตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 กราฟการเคลื่อนที่เก็บวัตถุที่ตำแหน่งใดๆ 5 ชั้น ในโหมดการคัดแยกสีเปรียบเทียบกับเวลา



รูปที่ 4.15 กราฟการเคลื่อนที่เก็บวัตถุที่ตำแหน่งใดๆ 5 ชั้น ในโหมดการคัดแยกรูปร่างของวัตถุเปรียบเทียบกับเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

จากผลการทดลองพบว่าส่วนของการเคลื่อนที่สามารถควบคุมการเคลื่อนที่ของหัวจับวัตถุได้ทั้งในโหมดการควบคุมแบบ Manual (การใช้ Joy Stick) และควบคุมแบบอัตโนมัติ โดยการควบคุมแบบอัตโนมัติสามารถนำค่าตำแหน่งและสีของวัตถุที่อยู่บนระนาบ โดยเรียงลำดับวัตถุจากระยะทางของวัตถุกับจุดที่กำหนดไว้ที่ถูกลำดับโดย Arduino board ส่งผ่าน Protocol RS-232 ไปเก็บยัง Data memory ที่กำหนดของ PLC ได้อย่างถูกต้อง และในรูปแบบการคัดแยกวัตถุด้วยสีตัวเครื่องสามารถทำการจำแนกสีของวัตถุและนำไปจัดเก็บตามพื้นที่ที่กำหนดได้อย่างถูกต้อง โดยแบ่งผลการจำแนกสีของวัตถุออกเป็น 4 สี ได้แก่ สีเหลือง สีชมพู สีเขียว และสีฟ้า ในขณะที่รูปแบบการคัดแยกวัตถุด้วยรูปร่าง สามารถทำการจำแนกรูปร่างของวัตถุด้วย Smart camera ได้อย่างถูกต้องและนำไปเก็บยังพื้นที่ ที่ตรงกับชนิดของวัตถุที่ทำการจำแนก โดยแบ่งผลการจำแนกรูปร่างของวัตถุออกเป็น 4 รูปร่าง ได้แก่ วงกลม สีเหลี่ยมจัตุรัส สีเหลี่ยมข้าวหลามตัด และแปดเหลี่ยม

ในส่วนสุดท้ายคือการออกแบบหน้าจอควบคุมและแสดงผลผ่านทางหน้าจอ HMI โดยในหน้าจอควบคุมนั้นจะมีโหมดการควบคุมให้เลือก ได้แก่ โหมดแมนนวล และโหมดอัตโนมัติ โดยในโหมดการควบคุมแบบอัตโนมัติจะมีการแบ่งเป็น 2 โหมดการควบคุม คือ รูปร่างของวัตถุ และสีของวัตถุ ภายในหน้าจอแสดงผล และในแต่ละโหมดจะมีการแสดงจำนวนของวัตถุ จำนวนของสีที่ตรวจพบ และจำนวนของรูปร่างของวัตถุที่ตรวจพบตามโหมดที่ได้เลือก โดยในการจัดทำหน้าจอควบคุมและแสดงผลแบบนี้จะทำให้สะดวกต่อการควบคุมและดูผลมากยิ่งขึ้น

โดยจากการทดลองทั้งหมดจะเห็นได้ว่าการเคลื่อนที่ของโหมด Auto ใช้เวลาในการเคลื่อนที่น้อยกว่าการเคลื่อนที่ในโหมด Manual ทุกระยะทางและจำนวนของวัตถุบนระนาบที่ได้ทำการทดลอง โดยมีความไวสูงสุดถึง 1 ใน 4 ของการเคลื่อนที่ในโหมด Manual ในตำแหน่งที่วัตถุอยู่บริเวณกึ่งกลางของพื้นที่ที่กำหนด โดยในโหมด Manual จะใช้ความเร็วในการเคลื่อนที่ 5.4 cm/s และจากการทำการทดลองซ้ำหลายๆครั้งในโหมด Auto พบว่าเวลาที่ใช้ในการเคลื่อนที่ในแต่ละครั้งใช้เวลาในการเคลื่อนที่มีความใกล้เคียงกันมาก ซึ่งมีความผิดพลาดเพียง 0.46 % ในโหมดการคัดแยกวัตถุด้วยสี และ 0.13% ในโหมดการคัดแยกวัตถุด้วยรูปร่าง ซึ่งคาดว่าความผิดพลาดเกิดจากความผิดพลาดของผู้ทดลอง ซึ่งค่าเหล่านี้อยู่ในเกณฑ์ที่รับได้

5.2 ปัญหาที่พบในระหว่างดำเนินการ

1. หากข้อมูลของการส่งค่าในแต่ละครั้งมากเกินไป จะทำให้ความจำของ Arduino เต็ม และจะทำให้การส่งค่านั้นผิดเพี้ยนไป
2. การสื่อสารของ Arduino กับ PLC ผ่านสาย Rs-232 นั้น มีความล่าช้า จึงทำให้การทำงานของโปรแกรมภายใน PLC เกิดความล่าช้าตามไปด้วย
3. หากเปิดใช้งาน servo drive เป็นเวลานาน จะทำให้ค่า PRV (Pulse) ที่อ่านได้เกิดการคลาดเคลื่อน ทำให้การคำนวณภายในโปรแกรมของ PLC เกิดความผิดพลาดได้ และ servo drive อาจมีอุณหภูมิที่สูงขึ้นได้หากเปิดใช้งานเป็นเวลานาน
4. ข้อจำกัดในการส่งค่าของ smart camera ผ่าน I/O board ให้ PLC นั้น สามารถส่งค่าได้เพียง 4 แบบ คือ 00, 01, 10 และ 11
5. ความแข็งแรงของเครื่อง คือ ขาที่ใช้จับกล้อง Pixy2 นั้นไม่มีความมั่นคง จึงทำให้กล้อง Pixy2 เกิดอาการสั่นไหวขณะที่ servo motor ทำงาน

5.3 แนวทางในการพัฒนา

จากงานวิจัยการค้นหาวัตถุด้วยกล้อง Smart Camera ในระนาบแบบ 2 มิติบนเครื่อง CNC และจำแนกไปยังพื้นที่จัดเก็บแบบอัตโนมัติด้วยเครื่องควบคุม PLC และรายงานตำแหน่งวัตถุด้วยจอ HMI สามารถนำไปพัฒนาต่อยอดได้ในการนำ IoT และ SCADA มาใช้ และหากต้องการเพิ่มความแม่นยำในการบอกตำแหน่งของวัตถุบนระนาบ ก็สามารถที่จะเปลี่ยนกล้อง Pixy2 ให้กลายเป็นกล้องที่ดีขึ้นกว่าเดิมมาใช้แทน

บรรณานุกรม

- [1] ดลสิทธิ์ ศิริเตียวศรี, ฤทธิรงค์ ดีสวาท และอิงครัต แหวนโชติช่วง, “การค้นหาวัดลุดด้วยกล้อง Smart Camera ในระนาบแบบ 2 มิติบนเครื่อง CNC และบันทึกตำแหน่งข้อมูลด้วย GENESIS32 SCADA”, ปรียญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมการวัดคุม, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2563
- [2] National Instruments, “NI 17xx Smart Camera User Manual”, February 2013
- [3] National Instruments, “NI Smart Camera I/O Accessory User Guide”, August 2008
- [4] National Instruments, “NI Vision Builder for Automated Inspection Tutorial”, September 2011
- [5] บริษัท ออมรอน อิเล็กทรอนิกส์ จำกัด, “คู่มือการติดตั้งและใช้งานโปรแกรมหน้าจอสัมผัส รุ่น NB” [ระบบออนไลน์] แหล่งที่มา : www.omron-ap.co.th
- [6] Omron, “Host Link Units System Manual”, May 1995
- [7] Omron, “CP1H CPU Unit Operation Manual”, May 2010
- [8] Omron, “Function Block/Structured Text Introduction Guide”, 2013
- [9] MISUMI, “โซลินอยด์วาล์ว วาล์วไฟฟ้าควบคุมปริมาตรการไหล”, [ระบบออนไลน์] แหล่งที่มา : https://th.misumi-ec.com/th/pr/recommend_category/solenoid_valves201904/ (6 ตุลาคม 2564)
- [10] RIGHT EQUIPMENT Co.,Ltd. , “ตัวสร้างสุญญากาศ (Vacuum Ejector)”, [ระบบออนไลน์] แหล่งที่มา : <http://www.right-eqm.com/14663795/ตัวสร้างสุญญากาศ-vacuum-ejector> (6 ตุลาคม 2564)
- [11] Pixy Documentation, “PixyMon v2 Reference”, 2018. [Online] Available : https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:pixymon_index (1 กันยายน 2564)
- [12] Pixycam, “Using Color Codes”, 2018. [Online] Available : https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:using_color_codes (1 กันยายน 2564)
- [13] DroneBot Workshop, “Pixy2 Camera – Object Recognition with Arduino & Raspberry Pi”, 2018. [Online] Available : https://dronebotworkshop.com/pixy2-camera/#Arduino_Library (1 กันยายน 2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[14] Pixy Documentation, “Arduino Library and API”, 2020. [Online] Available :
https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:arduino_api (1 กันยายน 2564)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP1H

General Specifications

Item	Type	AC power supply models	DC power supply models
	Model	CP1H-□□□-A	CP1H-□□□-D
Power supply		100 to 240 VAC 50/60 Hz	24 VDC
Operating voltage range		85 264 VAC	20.4 to 26.4 VDC (with 4 or more Expansion Units and Expansion I/O Units: 21.6 to 26.4 VDC)
Power consumption		100 VA max. (CP1H-□□□-A)(page 28)	50 W max. (CP1H-□□□-D)(page 28)
Inrush current (See note.)		100 to 120 VAC inputs: 20 A max. (for cold start at room temperature) 8 ms max. 200 to 240 VAC inputs: 40 A max. (for cold start at room temperature), 8 ms max.	30 A max. (for cold start at room temperature) 20 ms max.
External power supply		300 mA at 24 VDC	None
Insulation resistance		20 M Ω min. (at 500 VDC) between the external AC terminals and GR terminals	No insulation between primary and secondary for DC power supply
Dielectric strength		2,300 VAC at 50/60 Hz for 1 min between the external AC and GR terminals, leakage current: 5 mA max.	No insulation between primary and secondary for DC power supply
Noise immunity		Conforms to IEC 61000-4-4, 2 kV (power supply line)	
Vibration resistance		Conforms to JIS C60068-2-6, 10 to 57 Hz, 0.075-mm amplitude, 57 to 150 Hz, acceleration: 9.8 m/s ² in X, Y, and Z directions for 80 minutes each. Sweep time: 8 minutes \times 10 sweeps = total time of 80 minutes)	
Shock resistance		Conforms to JIS C60068-2-27, 147 m/s ² three times each in X, Y, and Z directions	
Ambient operating temperature		0 to 55°C	
Ambient humidity		10% to 90% (with no condensation)	
Ambient operating environment		No corrosive gas	
Ambient storage temperature		-20 to 75°C (Excluding battery.)	
Power holding time		10 ms min.	2 ms min.

Note: The above values are for a cold start at room temperature for an AC power supply, and for a cold start for a DC power supply.

- A thermistor (with low-temperature current suppression characteristics) is used in the inrush current control circuitry for the AC power supply. The thermistor will not be sufficiently cooled if the ambient temperature is high or if a hot start is performed when the power supply has been OFF for only a short time. In those cases the inrush current values may be higher (as much as two times higher) than those shown above. Always allow for this when selecting fuses and breakers for external circuits.
- A capacitor charge-type delay circuit is used in the inrush current control circuitry for the DC power supply. The capacitor will not be charged if a hot start is performed when the power supply has been OFF for only a short time, so in those cases the inrush current values may be higher (as much as two times higher) than those shown above.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP1H

Performance Specifications

Item	Type Models	CP1H-XA CPU Units	CP1H-X CPU Units	CP1H-Y CPU Units
		CP1H-XA□□□□□□	CP1H-X□□□□□□	CP1H-Y□□□□□□
Control method	Stored program method			
I/O control method	Cyclic scan with immediate refreshing			
Program language	Ladder diagram			
Function blocks	Maximum number of function block definitions: 128 Maximum number of instances: 256 Languages usable in function block definitions: Ladder diagrams, structured text (ST)			
Instruction length	1 to 7 steps per instruction			
Instructions	Approx. 500 (function codes: 3 digits)			
Instruction execution time	Basic instructions: 0.10 μs min. Special instructions: 0.15 μs min.			
Common processing time	0.7 ms			
Program capacity	20K steps			
Number of tasks	288 (32 cyclic tasks and 256 interrupt tasks)			
	Scheduled interrupt tasks	1 (interrupt task No. 2, fixed)		
	Input interrupt tasks	8 (interrupt task No. 140 to 147, fixed)	6 (interrupt task No. 140 to 145, fixed)	
		(Interrupt tasks can also be specified and executed for high-speed counter interrupts.)		
Maximum subroutine number	256			
Maximum jump number	256			
I/O areas (See note.)	Input bits	272bits (17 words) : CIO 0.00 to 16.15		
	Output bits	272bits (17 words) : CIO 100.00 to 116.16		
	Built-in Analog Inputs	CIO 200 to CIO 203		
	Built-in Analog Outputs	CIO 210 to CIO 211		
	Serial PLC Link Area	1,440 bits (90 words) : CIO 3100.00 to CIO 3189.15 (CIO 3100 to CIO 3189)		
Work bits	8,192 bits (512 words) : W0.00 to W511.15 (W0 to W511) CIO Area: 37,504 bits (2,344 words) : CIO 3800.00 to CIO 6143.15 (CIO 3800 to CIO 6143)			
TR Area	16 bits: TR0 to TR15			
Holding Area	8,192 bits (512 words) : H0.00 to H511.15 (H0 to H511)			
AR Area	Read-only (Write-prohibited) : 7168 bits (448 words) : A0.00 to A447.15 (A0 to A447) Read/Write: 8192 bits (512 words) : A448.00 to A959.15 (A448 to A959)			
Timers	4,096 bits: T0 to T4095			
Counters	4,096 bits: C0 to C4095			
DM Area	32 Kwords: D0 to D32767			
Data Register Area	16 registers (16 bits): DR0 to DR15			
Index Register Area	16 registers (32 bits): IR0 to IR15			
Task Flag Area	32 flags (32 bits): TK0000 to TK0031			
Trace Memory	4,000 words (500 samples for the trace data maximum of 31 bits and 6 words.)			
Memory Cassette	A special Memory Cassette (CP1W-ME05M) can be mounted. Note: Can be used for program backups and auto-booting.			
Clock function	Supported. Accuracy (monthly deviation): -4.5 min to -0.5 min (ambient temperature: 55°C), -2.0 min to +2.0 min (ambient temperature: 25°C), -2.5 min to +1.5 min (ambient temperature: 0°C)			
Communications functions	One built-in peripheral port (USB 1.1): For connecting Support Software only.			
	A maximum of two Serial Communications Option Boards can be mounted. A maximum of two Ethernet Option Boards can be mounted. When using CP1W-CIF41 Ver.1.0, one Ethernet Option Board can be mounted.			
Memory backup	Flash memory: User programs, parameters (such as the PLC Setup), comment data, and the entire DM Area can be saved to flash memory as initial values. Battery backup: The Holding Area, DM Area, and counter values (flags, PV) are backed up by a battery.			
Battery service life	5 years at 25°C. (Use the replacement battery within two years of manufacture.)			
Built-in input terminals	40 (24 inputs, 16 outputs)		20 (12 inputs, 8 outputs) Line-driver inputs: Two axes for phases A, B, and Z Line-driver outputs: Two axes for CW and CCW	
Number of connectable Expansion (I/O) Units	CP Expansion I/O Units: 7 max.; CJ-series Special I/O Units or CPU Bus Units: 2 max.			
Max. number of I/O points	320 (40 built in + 40 per Expansion (I/O) Unit × 7 Units)		300 (20 built in + 40 per Expansion (I/O) Unit × 7 Units)	
Item	Type Models	CP1H-XA CPU Units	CP1H-X CPU Units	CP1H-Y CPU Units
		CP1H-XA□□□□□□	CP1H-X□□□□□□	CP1H-Y□□□□□□
High-speed counters		4 inputs: Differential phases (4x), 50 kHz or Single-phase (pulse plus direction, up/down, increment), 100 kHz Value range: 32 bits, Linear mode or ring mode Interrupts: Target value comparison or range comparison		2 inputs: Differential phases (4x), 500 kHz or Single-phase, 1 MHz and 2 inputs: Differential phases (4x), 50 kHz or Single-phase (pulse plus direction, up/down, increment), 100 kHz Value range: 32 bits, Linear mode or ring mode Interrupts: Target value comparison or range comparison
	Pulse outputs (models with transistor outputs only)	Trapezoidal or S-curve acceleration and deceleration (Duty ratio: 50% fixed) 4 outputs, 1 Hz to 100 kHz (CCW/CW or pulse plus direction)		Trapezoidal or S-curve acceleration and deceleration (Duty ratio: 50% fixed) 2 outputs, 1 Hz to 1 MHz (CCW/CW or pulse plus direction) 2 outputs, 1 Hz to 100 kHz (CCW/CW or pulse plus direction)
	PWM outputs	Duty ratio: 0.0% to 100.0% (Unit: 0.1%) 2 outputs, 0.1 to 6553.5 Hz (Accuracy: ±5% at 1 kHz)		
Built-in analog I/O terminals	4 analog inputs and 2 analog outputs		None	
Analog control	1 (Setting range: 0 to 255)			
External analog input	1 input (Resolution: 1/256, Input range: 0 to 10 V), not isolated			

Note: The memory areas for CJ-series Special I/O Units and CPU Bus Units are allocated at the same as for the CJ-series. For details, refer to the CJ Series catalog (Cat. No. P052).

A

Specifications

The following specifications apply to the NI 1712/1732/1752/1754 Smart Camera. These specifications are typical at 25 °C, unless otherwise stated.

Power Requirements

Power consumption

NI 1712/1732 24 VDC, +20%/-15% (IEC 1311); 450 mA

NI 1752/1754

Direct Drive disabled 24 VDC, +20%/-15% (IEC 1311); 450 mA

Direct Drive enabled 24 VDC, +20%/-15% (IEC 1311); 800 mA

Reverse polarity protection Yes

Memory

SDRAM 256 MB

Nonvolatile program/data memory 512 MB

Image/data storage Unlimited using FTP or an Ethernet hard drive

Processor

NI 1712/1732 Freescale PowerQUICC II Pro 400 MHz

NI 1752/1754 Freescale PowerQUICC II Pro 600 MHz

VGA Sensor (NI 1712/1732/1752 Only)

Sensor Sony CCD ICX424AL

Active pixels (VGA)

Full scan 640 × 480

1/2 scan 640 × 240

1/4 scan 640 × 120

Binning (1 × 2) 640 × 240

Pixel size 7.4 μm × 7.4 μm

Pixel pitch for field of view calculation

Full scan, 1/2 scan, 1/4 scan 7.4 μm horizontal,

7.4 μm vertical

Binning (1 × 2) 7.4 μm horizontal,

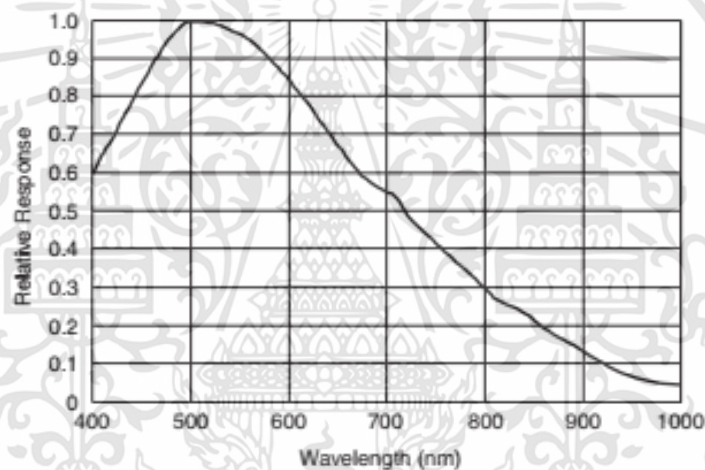
14.8 μm vertical

Appendix A Specifications

Maximum frame rate¹

Full scan	Up to 60 fps
1/2 scan	Up to 109 fps
1/4 scan	Up to 175 fps
Binning (1 × 2).....	Up to 114 fps
Optical format	1/3 in.
Sensor readout.....	Progressive scan
Bits per pixel	8 bits; 256 gray levels
Minimum exposure time	36.28 μs
Exposure time increment	31.2 μs
Spectral characteristics	Refer to Figure A-1

Figure A-1. VGA Spectral Response Curve



Gamma

1.0 fixed

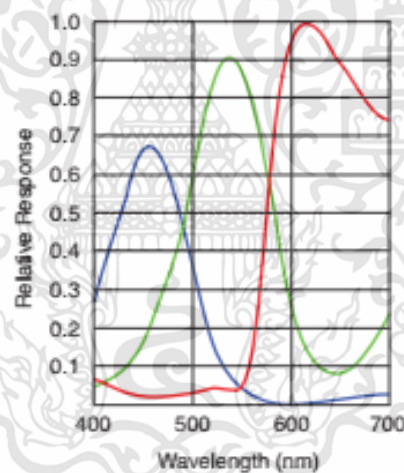
SXGA Sensor (NI 1754 Only)

Sensor	Sony CCD ICX205AL
Active pixels (SXGA)	
Full scan	1,280 × 1,024
1/2 scan	1,280 × 512
1/4 scan	1,280 × 256
Binning (1 × 2).....	1,280 × 512
Pixel size	4.65 μm × 4.65 μm
Pixel pitch	
Full scan, 1/2 scan, 1/4 scan	4.65 μm horizontal,

¹ Refer to the *Maximum Frame Rate* section of Chapter 6, *Image Acquisition*, for more information about calculating the maximum frame rate for your application.

Binning (1 × 2)	4.65 μm vertical 4.65 μm horizontal, 9.3 μm vertical
Maximum frame rate ¹	
Full scan	Up to 13 fps
1/2 scan	Up to 23 fps
1/4 scan	Up to 39 fps
Binning (1 × 2)	Up to 26 fps
Optical format	1/2 in.
Sensor readout	Progressive scan
Bits per pixel	8 bits; 256 gray levels
Minimum exposure time	76.68 μs
Exposure time increment	71.6 μs
Spectral characteristics	Refer to Figure A-2

Figure A-2. SXGA Sensor Spectral Response Curve



Gamma

1.0 fixed

Lighting

Direct Drive lighting controller (NI 1752/1754 Only)

Maximum current

500 mA continuous; 1 A strobed

Minimum current

50 mA

Light requirements

¹ Refer to the *Maximum Frame Rate* section of Chapter 6, *Image Acquisition*, for more information about calculating the maximum frame rate for your application.

Appendix A Specifications

Maximum voltage drop across LED+/LED- terminals	30 V, with $\pm 10\%$ input power supply 25 V, with $+20\%/-15\%$ input power supply
Minimum voltage drop across LED+/LED- terminals	7 V
Strobe frequency	Operating frame rate
Maximum strobe duty cycle	45%
5 V external strobe (NI 1732/1752/1754 Only)	
Polarity	Programmable
Strobe frequency	Operating frame rate
V _{OH} minimum	3.8 V
V _{OL} maximum	0.55 V
I _{OH} maximum	-12 mA
I _{OL} maximum	12 mA
24 V external strobe (NI 1732/1752/1754 Only)	
Polarity	Active high
Strobe frequency	Operating frame rate
ON state	
Voltage	Unregulated output drawn from the smart camera power supply
Current	16 mA, maximum
OFF state	
Voltage	Not driven
Current	Not applicable
Network	
Network interface	Ethernet
Ports	
NI 1712	1
NI 1732/1752/1754	2
Speed	
NI 1712	10; 100 Mbps
NI 1732/1752/1754	
Port 1	10; 100; 1,000 Mbps
Port 2	10; 100 Mbps
Duplex	Full, half
Speed autodetection	Yes
Duplex autodetection	Yes
Auto MDI/MDI-X correction	Yes
DHCP Support	Port 1 only

Serial

Baud rates	Up to 230.4 Kbps
Default baud rate	9,600 bps
Hardware flow control.....	No

Optically Isolated Inputs and Outputs

Isolated Inputs

Channels	
NI 1712	1
NI 1732/1752/1754.....	2
Input type	Sinking/sourcing, both inputs must have the same configuration
Digital logic levels	
OFF state	
Input current	0 mA to 0.1 mA
Input voltage	0 V to 1 V
ON state	
Input current	3 mA to 5.4 mA
Input voltage	20 V to 30 V
Minimum pulse width.....	1 ms

Isolated Outputs

Channels	
NI 1712	1
NI 1732/1752/1754.....	2
Output type	Sinking/sourcing, independently configurable
External load power supply range	19 V to 30 V
Output current.....	100 mA, maximum per channel
Max toggle rate.....	10 kHz

Quadrature Encoder (NI 1732/1752/1754 Only)

Encoder type	Differential, RS-422; phase A/phase B, no index
Max toggle rate	1 MHz

Physical Characteristics

Lens mount	C-mount
Camera housing	Painted die-cast aluminium

Appendix A Specifications

Dimensions	11.77 cm × 8.58 cm × 5.06 cm (4.63 in. × 3.38 in. × 1.99 in.)
Weight	525 g (18.52 oz)

Environmental

The NI Smart Camera is intended for indoor use only.

Operating temperature

Ambient temperature 0 to 45 °C

Maximum camera housing temperature ... 65 °C

Storage temperature -15 to 45 °C

Humidity 10% to 90% RH, noncondensing

IP rating..... 40

Pollution degree 2

Operating shock (IEC 60068-2-27) 50 g, 3 ms half sine, 18 shocks at 6 orientations;
30 g, 11 ms half sine, 18 shocks at 6 orientations

Operating vibration

Random (IEC 60068-2-34) 10 Hz to 500 Hz, 10 Grms, 100 min per axis

Swept sine (IEC 60068-2-6) 10 Hz to 500 Hz, 10 g

Approved at altitudes up to 2,000 m.

Safety

The NI Smart Camera meets the requirements of the following standards for safety and electrical equipment for measurement, control, and laboratory use:

- IEC 61010-1, EN 61010-1
- UL 61010-1, CSA 61010-1



Note For UL and other safety certifications, refer to the product label or visit ni.com/certification, search by model number or product line, and click the appropriate link in the Certification column.

Electromagnetic Compatibility

The NI Smart Camera meets the following standards of EMC for electrical equipment for measurement, control, and laboratory use:

- EN 61326 EMC requirements; Minimum Immunity
- EN 55011 Emissions; Group 1, Class A
- CE, C-Tick, ICES, and FCC Part 15 Emissions; Class A



Note For full EMC compliance, operate this device with shielded cabling.

CE Compliance

The NI Smart Camera meets the essential requirements of applicable European Directives, as amended for CE marking, as follows:

- 2006/95/EC; Low-Voltage Directive (safety)
- 2004/108/EC; Electromagnetic Compatibility Directive (EMC)



Note Refer to the Declaration of Conformity (DoC) for this product for any additional regulatory compliance information. To obtain the DoC for this product, visit ni.com/certification, search by model number or product line, and click the appropriate link in the Certification column.

Environmental Management

NI is committed to designing and manufacturing products in an environmentally responsible manner. NI recognizes that eliminating certain hazardous substances from our products is beneficial to the environment and to NI customers.

For additional environmental information, refer to the *Minimize Our Environmental Impact* web page at ni.com/environment. This page contains the environmental regulations and directives with which NI complies, as well as other environmental information not included in this document.

Waste Electrical and Electronic Equipment (WEEE)



EU Customers At the end of the product life cycle, all products *must* be sent to a WEEE recycling center. For more information about WEEE recycling centers, National Instruments WEEE initiatives, and compliance with WEEE Directive 2002/96/EC on Waste and Electronic Equipment, visit ni.com/environment/weee.

电子信息产品污染控制管理办法（中国 RoHS）



中国客户 National Instruments 符合中国电子信息产品中限制使用某些有害物质指令 (RoHS)。关于 National Instruments 中国 RoHS 合规性信息，请登录 ni.com/environment/rohs_china。(For information about China RoHS compliance, go to ni.com/environment/rohs_china.)

Specifications

This section lists the specifications of the NI Smart Camera I/O Accessory. These specifications are typical at 25 °C, unless otherwise noted.



Note Refer to the *NI 17xx Smart Camera User Manual* for specifications about signals.

Power Requirement

Supply voltage	24 VDC +20%/–15% (IEC 1311)
Current requirement	Refer to the specification for your NI Smart Camera

Physical

Dimensions	8.712 cm × 8.509 cm × 4.445 cm (3.43 in. × 3.35 in. × 1.75 in.)
Weight	98.1 g (3.46 oz)
Spring terminal blocks	6, 2 positions per block
Wire gauge	12–28 AWG wire
Camera connector	15-pin D-SUB male connector
RS-232 connector	9-pin D-SUB male connector
24VDC connector	15-pin D-SUB female connector

Environmental

The NI Smart Camera I/O Accessory is intended for indoor use only.

Operating temperature	0 to 55 °C
Storage temperature	–20 to 70 °C
Humidity	10 to 90%, noncondensing
Pollution Degree	2
Approved at altitudes up to 2,000 m.	

Safety

The NI Smart Camera I/O Accessory is designed to meet the requirements of the following standards of safety for electrical equipment for measurement, control, and laboratory use:

- IEC 61010-1, EN 61010-1
- UL 61010-1, CSA 61010-1



Note For UL and other safety considerations, refer to the product label, or visit ni.com/certification, search by model number or product line, and click the appropriate link in the Certification column.

CE Compliance

The NI Smart Camera I/O Accessory meets the essential requirements of applicable European Directives, as amended for CE marking, as follows:

- 2006/95/EC; Low-Voltage Directive (safety)
- 2004/108/EC; Electromagnetic Compatibility Directive (EMC)



Note Refer to the Declaration of Conformity (DoC) for this product for any additional regulatory compliance information. To obtain the DoC for this product, visit ni.com/certification, search by model number or product line, and click the appropriate link in the Certification column.

Compliance with EU Directives

Users in the European Union (EU) should refer to the Declaration of Conformity (DoC) for information¹ pertaining to the CE marking. Refer to the Declaration of Conformity (DoC) for this product for any additional regulatory compliance information. To obtain the DoC for this product, visit ni.com/certification, search by model number or product line, and click the appropriate link in the Certification column.

Environmental Management

National Instruments is committed to designing and manufacturing products in an environmentally responsible manner. NI recognizes that eliminating certain hazardous substances from our products is beneficial not only to the environment but also to NI customers.

For additional environmental information, refer to the *NI and the Environment* Web page at ni.com/environment. This page contains the environmental regulations and directives with which NI complies, as well as other environmental information not included in this document.

Waste Electrical and Electronic Equipment (WEEE)



EU Customers At the end of their life cycle, all products *must* be sent to a WEEE recycling center. For more information about WEEE recycling centers and National Instruments WEEE initiatives, visit ni.com/environment/weee.htm.

电子信息产品污染控制管理办法 (中国 RoHS)



中国客户 National Instruments 符合中国电子信息产品中限制使用某些有害物质指令 (RoHS)。关于 National Instruments 中国 RoHS 合规性信息, 请登录 ni.com/environment/rohs_china。(For information about China RoHS compliance, go to ni.com/environment/rohs_china.)

¹ The CE marking Declaration of Conformity contains important supplementary information and instructions for the user or installer.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้