รายงานสหกิจศึกษาฉบับสมบูรณ์

หุ่นยนต์ตรวจสอบหัวน็อตบนถังเครื่องซักผ้า
TUB VISION INSPECTION ROBOT

ธนกฤต โพธิ์ศรีนาค

ภาควิชาวิศวกรรมการวัดและควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

รายงานสหกิจศึกษาฉบับสมบูรณ์

หุ่นยนต์ตรวจสอบหัวน็อตบนถังเครื่องซักผ้า
TUB VISION INSPECTION ROBOT

ธนกฤต โพธิ์ศรีนาค

ภาควิชาวิศวกรรมการวัดและควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ชื่อโครงงาน	หุ่นยนต์ตรวจสอบน็อตบนถังเครื่องซักผ้า
นักศึกษา	นายธนกฤต โพธิ์ศรีนาค
ภาควิชา	วิศวกรรมการวัดและควบคุม
อาจารย์นิเทศ	ผู้ช่วยศาสตรจารย์ ดร.นพดล มณีรัตน์
ผู้นิเทศ	นายคมกฤช ทิพย์เกษร
สถานประกอบการ	บริษัท ไทยซัมซุง อิเล็กทรอนิกส์ จำกัด

บทคัดย่อ

โครงงานนี้เป็นการทำงานร่วมกันของระบบแมชชีนวิชันและหุ่นยนต์ เพื่อตรวจสอบคุณภาพของถังเครื่องซักผ้า ก่อนการประกอบเข้ากับชิ้นส่วนอื่นๆ เพื่อผลิตเป็นเครื่องซักผ้า และลดต้นทุนในการผลิตและตรวจสอบชิ้นงาน รวมทั้งช่วยลดระยะเวลาที่ใช้ในการตรวจสอบคุณภาพ โดยใช้โปรแกรมแลปวิว (LabVIEW) และฟังก์ชันวิชันแอสซิสแตนซ์ (Vision Assistant) ในการวิเคราะห์รูปภาพ ซึ่งตัวถังเครื่องซักผ้าที่นำมาวิเคราะห์มีลักษณะ และขนาดที่แตกต่างกันรวมทั้งจำนวนตำแหน่งที่ตรวจสอบแตกต่างกันในแต่ละรุ่น โดยระบบจะรับสัญญาณจาก PLC (Programmable Logic Controller) เพื่อเริ่มการทำงานและทำการบันทึกและวิเคราะห์ภาพ เพื่อคำนวณหาจุดศูนย์กลางและองศาการหมุนของตัวถังเครื่องซักผ้าอ้างอิงจากตำแหน่งของหุ่นยนต์ และทำการควบคุมหุ่นยนต์ให้เคลื่อนที่ไป ณ ตำแหน่งต่างๆ ที่ต้องการทำการตรวจสอบ แล้วจึงทำการบันทึกภาพหัวน็อต และประมวลผลเพื่อระบุว่ามีหัวน็อตครบถ้วนหรือไม่ ถ้าหากไม่พบน็อตระบบก็จะทำการคัดแยกตัวถังเครื่องซักผ้าชิ้นนั้นออกมาจากสายการผลิต เพื่อนำไปตรวจสอบแก้ไขต่อไป รวมทั้งระบบยังทำการบันทึกข้อมูลและรูปภาพเพื่อใช้ในการตรวจสอบย้อนหลัง และเป็นฐานข้อมูลเพื่อปรับปรุงสายการผลิตต่อไป

คำสำคัญ : หุ่นยนต์ตรวจสอบน็อตบนถังเครื่องซักผ้า, ระบบแมชชีนวิชัน

Project Title: Tub Vision Inspection Robot
Student: Mr.Thanakrit Phosrinak
Department: Instrumentation and Control Engineering
Advisor: Asst.Prof. Dr.Noppadol Maneerat
Mentor: Mr.Komkrit Thipgesorn
Company: Thai Samsung Electronics Co. Ltd.

ABSTRACT

This project is a collaboration between the machine vision system and robotics system to verify the quality of the washing machine tub before assembly with other parts and decrease production cost, decrease inspection time by using LabVIEW program and Vision Assistant function to analyze the image for calculate the position and analyze the picture of nut. The washing machine tub were analyzed with different styles and sizes, the number of checkpoints varies. The system receives signals from the PLC (Programable Logic Controller) to start and save the image. To calculate the center and degree of rotation of the washing machine tub. And the value is relative to the position of the robot and control the robot to move to every position to check. Make a picture of the nut that is processed with or without. If not, see the nut. The system will separate the washing machine tub for editing. The system also records and retrieves data and images for traceability and data updates to improve the production line.

Keywords: Tub Vision Inspection Robot, Machine Vision

กิตติกรรมประกาศ

ในการจัดทำโครงงานในครั้งนี้ สามารถทำให้สำเร็จลุล่วงไปด้วยดี เนื่องจากได้รับความช่วยเหลือจากผู้ช่วยศาสตราจารย์ ดร.นพดล มณีรัตน์ ที่มอบโอกาสในการเข้าร่วมโครงการสหกิจศึกษา คอยให้คำปรึกษา การสนับสนุน และความช่วยเหลือเป็นอย่างดี ตลอดจนการตรวจสอบความถูกต้องของรายงานจนทำให้รายงานฉบับนี้เสร็จสมบูรณ์ คณะผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบคุณทางบริษัท ไทยซัมซุง อิเล็กทรอนิกส์ จำกัด นายคมกฤช ทิพย์เกสร และขอขอบคุณพี่ๆ ทุกคนทั้งในแผนก FIT (Facility Innovation Technology) และแผนกอื่นๆ ที่คอยให้การช่วยเหลือสนับสนุนในเรื่องต่างๆ เช่น การให้คำปรึกษาและให้ข้อมูลที่เป็นประโยชน์ต่างๆ รวมทั้งทักษะในการแก้ไขปัญหา คอยดูแลติดตามความคืบหน้าของโครงงานอยู่เสมอ ทำให้โครงงานสามารถดำเนินไปและสำเร็จลุล่วงไปด้วยดี

สุดท้ายนี้ขอขอบคุณ เพื่อนร่วมงานทุกท่าน ที่คอยเป็นกำลังใจและให้การช่วยเหลือตลอดมาจนจบโครงงาน จึงขอขอบคุณไว้ ณ ที่นี้ หากมีข้อผิดพลาดประการใดให้ถือเป็นความบกพร่องของทางคณะผู้จัดทำ และขออภัยมา ณ ที่นี้ด้วย

ธนกฤต โพธิ์ศรีนาค

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญรูป VI
สารบัญตาราง VIII
บทที่ 1 บทนำ 1
1.1 ความเป็นมาและความสำคัญ 1
1.2 วัตถุประสงค์ของโครงงาน 2
1.3 ขอบเขตของโครงงาน 2
1.4 วิธีดำเนินโครงงาน 2
1.5 ประโยชน์ที่คาดว่าจะได้รับ 2
บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 3
2.1 วิธีการเลือกเซอร์กิตเบรกเกอร์ (Circuit Breaker) 3
2.2 โปรแกรม LabVIEW 4
2.3 Articulated Arm Robot (Revolute) 16
2.4 หลักการพื้นฐานในการสื่อสารข้อมูลในส่วนต่างๆ 20
2.5 หลักการพื้นฐานเกี่ยวกับการวิเคราะห์รูปภาพ 21
บทที่ 3 วิธีการดำเนินโครงงาน 24
3.1 การวางแผนการดำเนินโครงงาน 24
3.2 การศึกษาการทำงานในสายการผลิต และพื้นที่ในการติดตั้ง 25
3.3 การออกแบบโครงสร้างและชิ้นส่วนทางกล 27
3.4 การออกแบบโปรแกรม 30
3.5 การออกแบบระบบไฟฟ้า 35
บทที่ 4 ผลการดำเนินโครงงาน 36
4.1 โครงสร้างและระบบกลไก 36
4.2 ตู้ไฟที่ใช้ในการควบคุมการจ่ายไฟ 37
4.3 ผลการทดสอบการใช้งานจริง 38
บทที่ 5 สรุปผลการดำเนินโครงงานและข้อเสนอแนะ 50
5.1 ตำแหน่งของถังเครื่องซักผ้ามีความไม่แน่นอน 50
5.2 สายไฟบังจากกระบวนการผลิตก่อนหน้า (Wire Block) 51
5.3 ปัญหาระบบหาจุดศูนย์กลางถังเครื่องซักผ้าผิด (Settling Error) 52
5.4 การประมาลผลภาพไม่มีความแม่นยำ (Algorithm Error) 52
เอกสารอ้างอิง 55

สารบัญ (ต่อ)

หน้า

ภาคผนวก 56
ภาคผนวก ก. Wiring Diagram 57
ภาคผนวก ข. Programming Design-Flow Chart 58
ภาคผนวก ค. คู่มือการใช้งาน 61
ประวัติผู้เขียน 68

สารบัญรูป

รูปที่ หน้า

2.1 ตัวอย่างเครื่องมือวัดเสมือนที่สร้างจาก LabVIEW 4
2.2 หน้าจอการเขียนโปรแกรมและหน้าจอแสดงผล 6
2.3 Block Diagram ของ LabVIEW 7
2.4 Block Diagram เครื่องมือวัดที่สร้างจาก LabVIEW 8
2.5 Front Panel ของ LabVIEW 9
2.6 Object ที่อยู่บน Front Panel ของ LabVIEW 9
2.7 Controls Palette ที่ใช้ในการออกแบบ Front Panel 10
2.8 Tools Palette ที่ใช้ในการออกแบบ Front Panel 10
2.9 ตัวอย่าง Block Diagram 11
2.10 ตัวอย่าง Block Diagram Node 11
2.11 เครื่องมือ Tools Palette 12
2.12 ข้อมูลประเภท Numeric 13
2.13 ข้อมูลประเภท Boolean 14
2.14 การทำงานของ Dataflow Programming 15
2.15 ส่วนประกอบของระบบ DAQ 15
2.16 Articulated Arm Robot 16
2.17 แบบเคลื่อนที่ตามข้อหมุนของหุ่นยนต์ (Joint) 19
2.18 แบบเคลื่อนที่ตามแกน X, Y, Z (Trans) 19
2.19 แบบเคลื่อนที่ตามแกนของอุปกรณ์ที่ติดตั้ง (Tool) 20
3.1 การตรวจสอบน็อตโดยพนักงาน 25
3.2 พื้นที่ที่ใช้ในการติดตั้ง 25
3.3 ขนาดพื้นที่ที่ใช้ในการติดตั้ง 26
3.4 รูปแบบการทำงานของระบบ 26
3.5 ข้อมูลพื้นฐานของ Kawasaki Robot RS005L 27
3.6 ขนาดชิ้นส่วน และระยะการใช้งานของ Kawasaki Robot RS005L 27
3.7 รูปแบบหุ่นยนต์ตรวจสอบน็อตบนถังเครื่องซักผ้า 28
3.8 ลักษณะของชิ้นส่วนในการติดกล้องและไฟส่องสว่าง 28
3.9 รูปแบบโครงสร้างฐานวางหุ่นยนต์ 29
3.10 รูปแบบโครงสร้างรั้วป้องกันความปลอดภัย 29
3.11 Hex Washer 30
3.12 Hex Washer–Geometric Matching 30
3.13 Shotted Hex Washer 31
3.14 Shotted Hex Washer–Geometric Matching 31
3.15 Truss Head 31
3.16 Truss Head–Pattern Matching 32

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.17 Hex Nut	32
3.18 Hex Nut–Geometric Matching	32
3.19 หน้าต่าง Main Program	33
3.20 หน้าต่าง Setting	33
3.21 หน้าต่าง History	34
3.22 แผนผังวงจรไฟฟ้า	35
4.1 โครงสร้างทั้งหมดเมื่อนำมาติดตั้งในสายการผลิต	36
4.2 ชิ้นส่วนในการติดกล้องและไฟส่องสว่างที่เสร็จสมบูรณ์	36
4.3 การเดินสายไฟทั้งหมดในตู้ไฟ	37
4.4 ติดตั้งตู้ไฟและทดสอบการทำงาน	37
4.5 ทดสอบหาการบิดของถังเครื่องซักผ้า	39
4.6 ทดสอบหาจุดศูนย์กลางของถังเครื่องซักผ้า	39
4.7 กราฟเปรียบเทียบปริมาณ Error วันที่ 1	41
4.8 กราฟเปรียบเทียบปริมาณ Error วันที่ 2	42
4.9 กราฟเปรียบเทียบปริมาณ Error วันที่ 3	43
4.10 กราฟเปรียบเทียบปริมาณ Error วันที่ 4	44
4.11 กราฟเปรียบเทียบปริมาณ Error วันที่ 5	45
4.12 กราฟเปรียบเทียบปริมาณ Error วันที่ 6	46
4.13 กราฟเปรียบเทียบปริมาณ Error วันที่ 7	47
4.14 กราฟเปรียบเทียบปริมาณ Error วันที่ 8	48
4.15 กราฟเปรียบเทียบปริมาณ Error วันที่ 9	49
5.1 ก่อนทำการติดตั้ง Guide	50
5.2 หลังทำการติดตั้ง Guide	51
5.3 สายไฟบังหัวน็อต	51
5.4 ก่อนติด Cover กันแสง	52
5.5 หลังติด Cover เพื่อกันแสงรบกวนจากบริเวณรอบ	53
5.6 กราฟเปรียบเทียบปริมาณ Error หลังทำการแก้ไขในแต่ละวัน	53

สารบัญตาราง

ตารางที่ **หน้า**

2.1 เปรียบเทียบคำศัพท์ที่ใช้ในการเขียนโปรแกรม 13
2.2 ข้อมูลและขนาดการทำงานของหุ่นยนต์ 17
2.3 ข้อมูล E Controller 20
3.1 Tub Vision Inspection Robot Timeline 24
4.1 ทดสอบการตรวจจับหัวน็อต 38
4.2 บันทึกเวลาในการเคลื่อนที่ของหุ่นยนต์ 40
4.3 ผลการทดสอบวันที่ 1 41
4.4 ผลการทดสอบวันที่ 2 42
4.5 ผลการทดสอบวันที่ 3 43
4.6 ผลการทดสอบวันที่ 4 44
4.7 ผลการทดสอบวันที่ 5 45
4.8 ผลการทดสอบวันที่ 6 46
4.9 ผลการทดสอบวันที่ 7 47
4.10 ผลการทดสอบวันที่ 8 48
4.11 ผลการทดสอบวันที่ 9 49

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

ในยุคปัจจุบันเป็นยุคที่ภาคอุตสาหกรรมมีการเจริญเติบโตเป็นอย่างมาก ทำให้ผู้บริโภคมีตัวเลือกในการเลือกสินค้าที่หลากหลายและมีคุณภาพใกล้เคียงกัน ทำให้ในภาคการผลิตนั้นจำเป็นที่จะต้องควบคุมคุณภาพสินค้าให้มีความคงที่และเชื่อถือได้ อีกทั้งต้องพัฒนาวิธีในการผลิตเพื่อลดต้นทุน และลดเวลาในการผลิตให้น้อยกว่าผู้ผลิตรายอื่นในประเภทธุรกิจเดียวกัน เพื่อทำให้เกิดความได้เปรียบทางการค้า และรักษามาตรฐานของสินค้า เพื่อเพิ่มความน่าเชื่อถือของผู้บริโภค ดังนั้นในการผลิตชิ้นงานหรือเครื่องอุปโภคบริโภคจึงจำเป็นต้องมีหน่วยงานที่ทำหน้าที่ในการตรวจเช็ค เพื่อตรวจสอบคุณภาพของสินค้าก่อนที่จะมีการส่งสินค้าออกไปสู่ผู้บริโภค โดยในอดีตจำเป็นที่จะต้องใช้คนในการตรวจสอบคุณภาพของชิ้นงาน แต่ด้วยการแข่งขันในภาคอุตสาหกรรมทำให้การผลิตสินค้าต้องมีความรวดเร็ว และทำการผลิตแข่งกับเวลา ทำให้คนไม่สามารถที่จะทำงานได้เป็นเวลานาน และต่อเนื่อง อีกทั้งความเหนื่อยล้าของประสาทสัมผัสยังทำให้สมรรถภาพของคนลดลง จึงเป็นสาเหตุที่อาจทำให้ชิ้นงานที่ไม่ผ่านมาตรฐานหลุดออกไปสู่มือของผู้บริโภคได้ ซึ่งทำให้ส่งผลต่อความน่าเชื่อถือของผู้บริโภคต่อสินค้านั้น

ในภาคการผลิตยุคปัจจุบันจึงมีการนำเทคโนโลยีต่างๆ เข้ามาช่วยในการตรวจสอบคุณภาพของสินค้าแทนทรัพยากรบุคคล เนื่องจากในปัจจุบันนั้นเทคโนโลยีมีความแม่นยำสูง ใช้งานได้ต่อเนื่อง ทำงานได้อย่างรวดเร็ว และใช้ต้นทุนในระยะยาวที่คุ้มค่ากว่าการจ้างแรงงานคน เช่น การตรวจสอบรอยขีดข่วนในชิ้นงานขนาดเล็ก การตรวจสอบการขันสกรูในชิ้นงาน การตรวจสอบสีของชิ้นงาน การตรวจสอบการรั่วไหลของแก๊ส ฯลฯ ในงานเหล่านี้จึงจำเป็นต้องใช้เทคโนโลยีที่มีประสิทธิภาพสูงในการตรวจสอบคุณภาพของชิ้นงาน จึงได้เลือกนำระบบแมชชีนวิชัน (Machine Vision) มาเพื่อใช้ตรวจสอบการขันสกรูบนตัวถังของเครื่องซักผ้า เพื่อทำการคัดแยกชิ้นงานที่ผ่านมาตรฐานเข้าสู่สายการผลิตและคัดแยกของชิ้นงานที่ไม่ได้มาตรฐานออกจากสายการผลิต เพื่อนำไปปรับปรุงแก้ไขและนำกลับเข้าสู่สายการผลิตต่อไป

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อตรวจสอบคุณภาพของถังเครื่องซักผ้า
2. เพื่อลดระยะเวลาในกระบวนการผลิตเครื่องซักผ้า
3. เพื่อให้ง่ายต่อการนำถังเครื่องซักผ้ามาแก้ไขปรับปรุงและซ่อมแซม
4. เพื่อลดต้นทุนและเพิ่มประสิทธิภาพในการผลิตและการตรวจสอบคุณภาพ

1.3 ขอบเขตของโครงงาน

1. ออกแบบเครื่อง Tub Vision Inspection Robot แทนการใช้พนักงาน
2. ออกแบบโปรแกรมเพื่อใช้ในการตรวจสอบน็อตบนถังเครื่องซักผ้า
3. ลดระยะเวลาที่ใช้ในกระบวนการผลิต
4. เครื่อง Tub Vision Inspection Robot สามารถนำไปใช้ได้จริงในสายการผลิต

1.4 วิธีดำเนินโครงงาน

1. วางแผนการดำเนินงาน
2. ศึกษาการทำงานในสายการผลิต และตำแหน่งที่ต้องเข้าไปทำการติดตั้ง
3. ศึกษาและออกแบบโครงสร้างทางกล
4. ศึกษาและออกแบบโปรแกรม LabVIEW และโปรแกรมควบคุมหุ่นยนต์
5. ศึกษาและออกแบบวงจรไฟฟ้าควบคุม
6. จัดซื้อและประกอบ
7. ทดสอบการทำงาน
8. ติดตั้งในสายการผลิตและทดสอบการทำงานในพื้นที่จริง
9. จัดทำสรุปโครงงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถนำเครื่อง Tub Vision Inspection Robot ไปใช้แทนพนักงานได้จริง
2. รู้จักการวางแผนงาน และการแก้ไขปัญหาเฉพาะหน้า
3. ฝึกการทำงานร่วมกับผู้อื่น
4. สามารถนำประสบการณ์ในการทำงานไปต่อยอดได้ในอนาคต

บทที่ 2
แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 วิธีการเลือกเซอร์กิตเบรกเกอร์ (Circuit Breaker)

เหตุผลหลักที่จะติดตั้งเซอร์กิตเบรกเกอร์ก็เพื่อป้องกันระบบไฟฟ้าจากกระแสลัดวงจร และกระแสเกิน (Overload) ซึ่งไม่ว่าจะเป็นการติดตั้งใหม่หรือเปลี่ยนซ่อมบำรุงก็ตาม เซอร์กิตเบรกเกอร์นั้นใช้ในระบบตั้งแต่ในตู้คอนโทรล (Control Panel) ตู้คอนซูมเมอร์ยูนิท (Consumer Unit) สำหรับในบ้านพัก ตู้โหลดเซนเตอร์ (Load Center) ตู้สวิตช์บอร์ด (MDB), ตู้ควบคุมมอเตอร์

เซอร์กิตเบรกเกอร์ทุกประเภทจะใช้กับระบบไฟฟ้าแบบ 3 เฟส 4 สาย เป็นระบบที่ใช้ในเมืองไทย ซึ่งส่วนใหญ่แล้วคุณจะดึง 3 เฟสเพาเวอร์ ไปใช้ในอาคารพาณิชย์และโรงงานอุตสาหกรรม หรือ 1 เฟส (Single Phase) ไปใช้ในที่พักอาคาร

ในอาคารที่พักอาศัยที่ใช้แบบ 1 เฟส จะใช้เบรกเกอร์ลูกย่อยแบบ MCB ควบคู่กับตู้คอนซูมเมอร์ยูนิท แต่ในอาคารพาณิชย์กับโรงงานอุตสาหกรรมที่ใช้ไฟฟ้าแรงดันต่ำแบบไม่เกิน 690V และส่วนมากในประเทศไทยจะใช้อยู่ที่ 400V พวกเซอร์กิตเบรกเกอร์ในระบบนี้จะเป็นแบบ Molded Case Circuit Breaker (MCCB) หรือ Air Circuit Breaker (ACB) ที่ใส่ในตู้สวิตช์บอร์ด MDB (Main Distribution Board)

ในการที่จะเลือกเซอร์กิตเบรกเกอร์ให้ถูกต้องและเหมาะสมกับงานที่ใช้ จำเป็นต้องคำนึงถึง 2 ประเด็นด้วยกัน ดังนี้

2.1.1 จำนวน Pole

เป็นตัวบอกว่าเบรกเกอร์ที่ใช้นั้นเป็นชนิด 1 เฟส หรือ 3 เฟส

- 4 Pole หมายถึง เป็นเซอร์กิตเบรกเกอร์สำหรับระบบ 3 เฟส โดยป้องกันสาย Line และสาย Neutral เหมาะสำหรับระบบที่ต้องการความปลอดภัยสูง หากมีความผิดปกติของระบบไฟฟ้า เบรกเกอร์สามารถป้องกันได้ทั้ง 4 เส้น
- 3 Pole หมายถึง เป็นเซอร์กิตเบรกเกอร์สำหรับระบบ 3 เฟส โดยป้องกันแค่สาย Line อย่างเดียว 3 Pole จะใช้กันมากในอาคารพาณิชย์และโรงงานอุตสาหกรรม
- 2 Pole หมายถึง เป็นเซอร์กิตเบรกเกอร์สำหรับระบบ 1 เฟส โดยป้องกันสาย Line และสาย Neutral 2 Pole มักจะใช้มาเป็นเมนเบรกเกอร์ในตู้คอนซูมเมอร์ยูนิท มีทั้งที่เป็นเบรกเกอร์แบบ MCB และ MCCB
- 1 Pole หมายถึง เป็นเซอร์กิตเบรกเกอร์สำหรับระบบ 1 เฟส โดยป้องกันแค่สาย Line อย่างเดียว ส่วนใหญ่จะเป็นเบรกเกอร์ลูกย่อยที่ใช้ร่วมกับตู้คอนซูมเมอร์ยูนิท และมักใช้ในบ้านที่พักอาศัย

2.2.2 ค่าพิกัดกระแส (Breaking Capacity IC, Amp Trip AT, Amp Frame AF)

ซึ่งค่าพิกัดเป็นตัวบ่งบอกถึงความสามารถขีดจำกัดในการใช้งานของเซอร์กิตเบรกเกอร์ โดยค่าพิกัดที่ควรทราบมีดังนี้

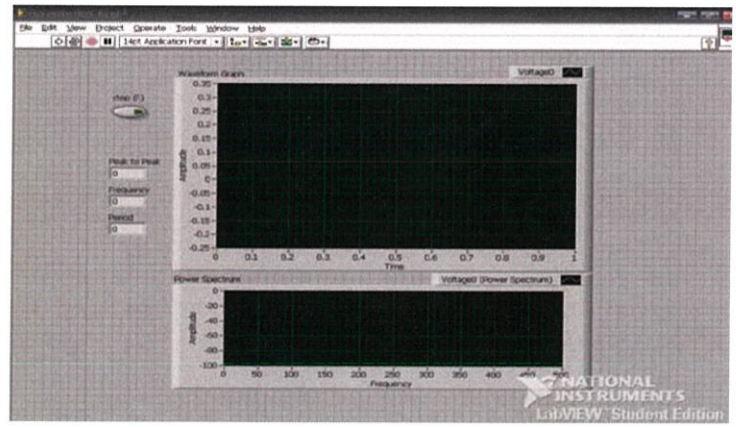
2.2.2.1 Interrupting Capacitive (IC) : พิกัดการทนกระแสลัดวงจรสูงสุดโดยปลอดภัยของเบรกเกอร์นั้นๆ มักแสดงในหน่วย kA

2.2.2.2 Amp Trip (AT) : ขนาดกระแสที่ใช้งาน เป็นตัวบอกให้รู้ว่าเบรกเกอร์ตัวนั้นสามารถทนต่อกระแสในภาวะปกติได้สูงสุดเท่าใด

2.2.2.3 Amp Frame (AF) : พิกัดกระแสโครง หมายถึงขนาดการทนกระแสของเปลือกหุ้มเป็นพิกัดการทนกระแสสูงสุดของเบรกเกอร์นั้นๆ เซอร์กิตเบรกเกอร์ที่มีขนาด AF เดียวกันจะมีขนาดมิติ (กว้างXยาวXสูง) เท่ากัน สามารถเปลี่ยนพิกัด Amp Trip ได้โดยที่ขนาด (มิติ) ของเบรกเกอร์ยังคงเท่าเดิม

2.2 โปรแกรม LabVIEW

LabVIEW ย่อมาจาก Laboratory Virtual Instrument Engineering Workbench โปรแกรมที่พัฒนาขึ้นโดยใช้ LabVIEW จะเรียกว่า Virtual Instrument หรือจะเรียกย่อๆ ว่า VI ซึ่งหมายถึงเครื่องมือวัดเสมือน ดังตัวอย่างจากรูปที่ 2.1 เป็น Oscilloscope ที่ได้ทำการสร้างขึ้นบนหน้าจอคอมพิวเตอร์



รูปที่ 2.1 ตัวอย่างเครื่องมือวัดเสมือนที่สร้างจาก LabVIEW

LabVIEW กำเนิดขึ้นในปี ค.ศ. 1983 โดยทางบริษัท National Instrument ได้เริ่มการค้นคว้า เพื่อหาวิธีการที่จะลดเวลาในการเขียนโปรแกรมเพื่อใช้ในงานด้านระบบเครื่องมือวัด ซึ่งเป็นจุดเริ่มของแนวความคิดการสร้าง LabVIEW หลังจากการใช้เวลาวิจัย 3 ปี ในปี ค.ศ. 1986 บริษัทได้ปล่อย LabVIEW Version 1 สู่ตลาดเพื่อใช้กับคอมพิวเตอร์ Macintosh เท่านั้น เพราะแม้ว่าเครื่อง Macintosh จะไม่เป็นที่ใช้อย่างกว้างขวางในงานด้านวิศวกรรม แต่ด้วยลักษณะการแสดงผลแบบกราฟฟิกของเครื่อง Macintosh ทำให้เหมาะสมกับการประยุกต์ใช้กับ LabVIEW สำหรับระบบปฏิบัติการอื่นที่ไม่ใช่ GUI นั้นยังไม่มีความเหมาะสมที่จะใช้กับ LabVIEW ดังนั้นสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคล (PC) ทาง NI ต้องรอจนกระทั่งระบบปฏิบัติการ Windows เกิดขึ้นเสียก่อน

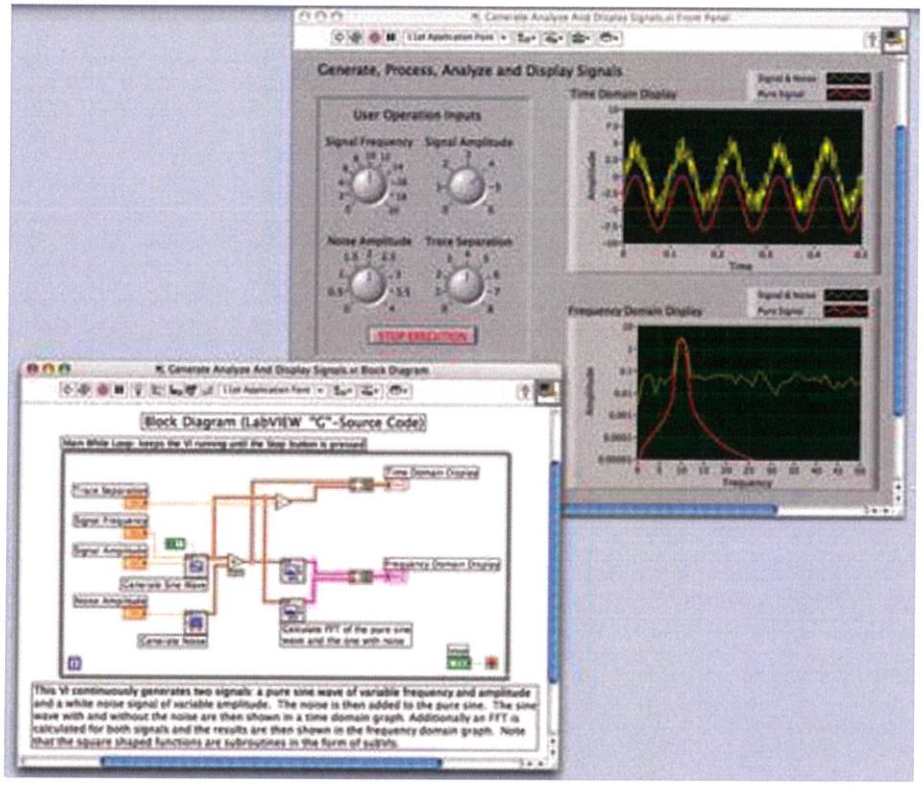
ในปี ค.ศ. 1990 ทาง NI ได้ประสบผลสำเร็จในการนำ LabVIEW Version 2 ออกสู่ตลาด โดยได้ปรับแก้และเขียนระบบควบคุมใหม่ทั้งหมดตามคำแนะนำของผู้ใช้งาน โดยเฉพาะการเขียน Compiler ที่ทำให้เวลาการทำงานของโปรแกรมรวดเร็วขึ้นทัดเทียมกับการเขียนด้วยภาษาขั้นพื้นฐาน เช่น C และต่อมาเมื่อเทคโนโลยีด้านระบบปฏิบัติของ PC มีความพร้อมที่จะใช้งานกับ GUI ทางบริษัทจึงได้ผลิต LabVIEW for Windows และ LabVIEW for SUN เข้าสู่ตลาดในปี ค.ศ. 1992

หลังจากนั้นบริษัทก็ได้พัฒนาโปรแกรม ให้เหมาะสมกับเทคโนโลยียิ่งขึ้นตามรูปแบบปฏิบัติการที่เปลี่ยนแปลงไป เช่น LabVIEW สำหรับ Windows NT, Windows 95 รวมถึงการสร้าง Version ใหม่ เพื่อจัดระบบและการเขียนโปรแกรมให้สะดวกมากขึ้น ตลอดจนสามารถเชื่อมต่อกับอุปกรณ์ต่างๆ มากขึ้นพร้อมทั้งสร้างฟังก์ชันต่างๆ เพื่อให้เหมาะสมกับการใช้งานมากขึ้น

นอกจากนี้ยังสร้างโปรแกรมที่สามารถทำงานบนระบบปฏิบัติการอื่นที่ไม่ได้เขียนบนระบบปฏิบัติการนั้นได้ โดยเริ่มจาก LabVIEW 3 ในปี ค.ศ. 1993, LabVIEW 4 ในปี ค.ศ. 1996 และล่าสุด LabVIEW 2010 ในปี ค.ศ. 2010 ซึ่งเป็นโปรแกรมล่าสุด

โปรแกรม LabVIEW เป็นโปรแกรมที่สร้างเพื่อนำมาใช้ในด้านการวัด และเครื่องมือวัดสำหรับงานทางวิศวกรรม เป็นโปรแกรมที่สร้างเครื่องมือวัดเสมือนจริงในห้องปฏิบัติการทางวิศวกรรม ดังนั้นจุดประสงค์หลักของการทำงานของโปรแกรมนี้ก็คือ การจัดการในด้านการวัดและเครื่องมือวัดอย่างมีประสิทธิภาพ และในตัวของโปรแกรมจะประกอบไปด้วยฟังก์ชันที่ใช้ช่วยในการวัดมากมาย และแน่นอนที่สุดโปรแกรมนี้จะมีประโยชน์อย่างสูงเมื่อใช้ร่วมกับเครื่องมือวัดทางวิศวกรรมต่างๆ สิ่งที่ LabVIEW แตกต่างจากโปรแกรมอื่นอย่างเห็นได้ชัดที่สุดก็คือ LabVIEW นี้ เป็นโปรแกรมประเภท GUI (Graphic User Interface) โดยสมบูรณ์ นั่นคือไม่จำเป็นต้องเขียน Code หรือคำสั่งใดๆ ทั้งสิ้น และที่สำคัญลักษณะภาษาที่ใช้ในโปรแกรมนี้จะเรียกว่าเป็นภาษารูปภาพ หรือเรียกอีกอย่างว่าภาษา G (Graphical Language) ซึ่งจะแทนการเขียนโปรแกรมเป็นบรรทัดแบบภาษาพื้นฐาน เช่น C, BASIC หรือ FORTRAN ด้วยรูปภาพหรือสัญลักษณ์ทั้งหมด แม้ว่าในเบื้องต้นอาจจะสับสนอยู่บ้าง แต่เมื่อคุ้นเคยกับการใช้โปรแกรมนี้แล้ว จะพบว่า LabVIEW นี้มีความสะดวก และสามารถลดเวลาในการเขียนโปรแกรมลงไปได้มาก โดยเฉพาะในงานเขียนโปรแกรมคอมพิวเตอร์เพื่อเชื่อมต่อกับอุปกรณ์อื่นๆ เพื่อใช้ในการวัดและการควบคุม โดยจุดประสงค์หลักแล้ว บริษัท National Instrument ได้เริ่มพัฒนาโปรแกรมที่จะนำมาใช้กับระบบเครื่องมือวัด ที่มีความง่ายในการเขียนโปรแกรม และมีฟังก์ชันเพื่อจะช่วยในการวัดทางวิศวกรรมให้มากที่สุด เพราะด้วยความเป็นมาบริษัท National Instrument เริ่มจากการผลิตอุปกรณ์ที่ใช้กับการวัดทางวิศวกรรม ไม่ใช่บริษัทที่เริ่มต้นมาจากการผลิต Software เป็นหลัก ดังนั้นคงไม่ผิดนักสำหรับผู้ที่ต้องการจะใช้ประโยชน์สูงสุดจากโปรแกรม LabVIEW คือ ผู้ที่ต้องการจะนำข้อมูลจากภายนอกเครื่องคอมพิวเตอร์ เข้ามาในเครื่องเพื่อทำการวิเคราะห์ข้อมูล ประมวลค่า แสดงผล และในหลายกรณีใช้ในระบบควบคุมอัตโนมัติด้วยคอมพิวเตอร์ ข้อได้เปรียบสูงสุดของ LabVIEW คือ การพยายามทำให้เครื่องคอมพิวเตอร์เมื่อรวมกับ LabVIEW และอุปกรณ์เชื่อมต่อเพื่อการเก็บข้อมูล (Data Acquisition Card) แล้วสามารถเปลี่ยนเครื่องคอมพิวเตอร์ส่วนบุคคลให้กลายเป็นเครื่องมือวัดในหลายรูปแบบ ไม่ว่าจะเป็น Oscilloscope, Multi-meter, Function Generator, Strain Meter Thermometer หรือเครื่องมือวัดอื่นๆ ตามที่ต้องการ ทำให้สามารถใช้คอมพิวเตอร์ในการทำการวัดและเครื่องมือวัดได้อย่างกว้างขวาง นี้เองที่เป็นที่มาของชื่อเครื่องมือวัดเสมือนจริง (Virtual Instrument) และข้อได้เปรียบเหนือการใช้อุปกรณ์จริง

เหล่านั้นคือ Virtual Instrument สามารถปรับเปลี่ยนให้เหมาะสมกับการใช้งานของผู้ใช้แต่ละกลุ่มได้ โดยการเปลี่ยน VI ให้เป็นไปตามต้องการเป็นเรื่องที่ไม่ยุ่งยากนัก ดังรูปที่ 2.2



รูปที่ 2.2 หน้าจอการเขียนโปรแกรมและหน้าจอแสดงผล

ข้อดีอีกประการหนึ่งในการใช้คอมพิวเตอร์เป็นเครื่องมือวัดก็คือ สามารถใช้ทำเป็น Data Logger และ PLC (Programmable Logical Controlled) ได้พร้อมกัน ซึ่งโดยปกติแล้วระบบควบคุมมักจะไม่มีในเครื่องมือวัดจริงขั้นพื้นฐาน หรือ Data Logger แม้จะเก็บข้อมูลได้ แต่การสั่งการทำงานกับอุปกรณ์ตัวอื่น จะมีความยุ่งยากในการสั่งการมาก

สำหรับผู้ที่เคยใช้โปรแกรมประเภทที่ใช้ตัวหนังสือ หรือที่เรียกว่า Text Base ทั้งหลายคงจะทราบถึงความยุ่งยากในการจัดการกับตำแหน่งการส่งผ่านข้อมูลตามอุปกรณ์เชื่อมต่อ เช่น Port หรือ Card ต่างๆ รวมถึงการจัดวางตำแหน่งในหน่วยความจำเพื่อที่จะสามารถรวบรวมข้อมูลมาใช้ ในการคำนวณและเก็บข้อมูลให้ได้ประโยชน์สูงสุด ปัญหาเหล่านี้ได้รับการแก้ไขใน LabVIEW โดยได้มีการบรรจุโปรแกรมจำนวนมาก หรือ Libraries ไว้สำหรับจัดการกับปัญหาเหล่านั้น ไม่ว่าอุปกรณ์การเชื่อมต่อจะเป็น DAQ (Data Acquisition), GPIB (General Purpose Interface Bus หรือก่อนหน้านี้รู้จักกันในชื่อ Hewlett Packard Interface Bus, HP - IB), พอร์ตอนุกรม หรือ Serial Port เพื่อใช้ติดต่อกับอุปกรณ์ที่ส่งผ่านข้อมูลแบบอนุกรม (Serial Instrument) รวมถึงการวิเคราะห์ข้อมูล ที่ได้ด้วยวิธีการต่างๆ นอกจากนี้ใน Libraries เหล่านั้นยังได้บรรจุฟังก์ชันการทำงานที่สำคัญอีกหลายประการ เช่น Signal Generation, Signal Processing, Filters, สถิติ, พีชคณิต และคณิตศาสตร์อื่นๆ ดังนั้น LabVIEW จึงทำให้การวัดและการใช้เครื่องมือวัดกลายเป็นเรื่องง่ายลงไปมาก และทำให้คอมพิวเตอร์ส่วนบุคคลกลายเป็นเครื่องมือทางด้านการวัดหลายชนิดอยู่ในเครื่องเดียว

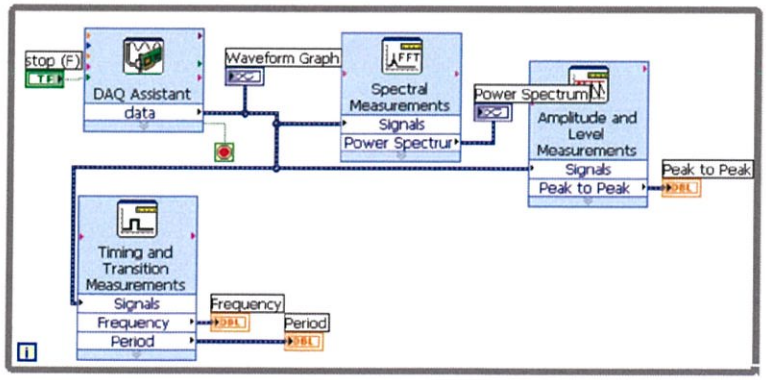
2.2.1 Data Flow and Programing

เนื่องจาก LabVIEW เป็นโปรแกรมที่ใช้รูปภาพ หรือสัญลักษณ์แทนการเขียนด้วยตัวอักษร เหมือนโปรแกรมปกติทั่วไปซึ่งข้อดีข้อแรกก็คือ การลดความผิดพลาดด้านการสะกดผิดหรือพิมพ์ผิดออกไป ข้อแตกต่างอีกประการหนึ่งที่สำคัญของการเขียนโปรแกรมแบบ G กับการเขียนด้วยตัวหนังสือก็คือ การเขียนด้วยภาษา G นี้เป็นการเขียนโดยใช้หลักการของ Data Flow ซึ่งเมื่อเริ่มส่งข้อมูลเข้าสู่โปรแกรม จะต้องกำหนดทิศทางไหลของข้อมูลว่าจะไปที่ส่วนใด ผ่านการประเมินผลและคำนวณในส่วนใดบ้าง และจะให้แสดงผลอย่างไร ซึ่งลักษณะการเขียนภาษา G หรือ Data Flow นี้จะมีลักษณะเหมือนกับการเขียน Block Diagram ซึ่งทำให้ผู้เขียนโปรแกรมสามารถให้ความสนใจกับการเคลื่อนที่และเปลี่ยนแปลงข้อมูลได้โดยไม่ต้องจดจำรูปแบบคำสั่งที่ยุ่งยาก

เนื่องจาก LabVIEW ใช้ลักษณะการเขียนแบบ Block Diagram ซึ่งวิศวกรส่วนใหญ่มีความคุ้นเคยอยู่แล้ว จึงเป็นการง่ายที่จะทำความเข้าใจและนำไปพัฒนาใช้ต่อไปได้ และถ้าหากจำได้ถึงขั้นตอนการเขียนโปรแกรมว่าก่อนที่จะเขียนโปรแกรม จะต้องเขียน Flow Chart ให้เสร็จสิ้นก่อน หลังจากตรวจสอบ Flow Chart เรียบร้อยแล้ว จึงนำไปเขียนโปรแกรม ซึ่งจะมีความสะดวกมากขึ้น ถ้าหากการเขียน Flow Chart ของ LabVIEW ก็คือการเขียนโปรแกรมนั่นเอง ซึ่งเป็นการลดขั้นตอนการทำงานลงไปได้เป็นอย่างมากแม้ว่าการเขียนโปรแกรมใน LabVIEW ไม่จำเป็นต้องมีความรู้ด้านการเขียนโปรแกรมใดๆ มาก่อนเลย แต่การมีความรู้ด้านการเขียนโปรแกรมหรือใช้โปรแกรมสำเร็จรูปอื่นๆ จะสามารถนำมาใช้ประโยชน์ได้เป็นอย่างดี

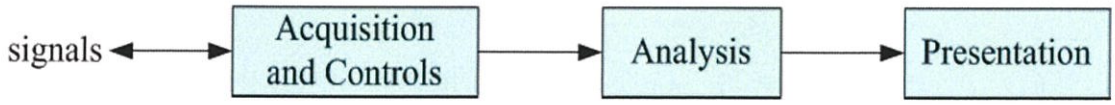
LabVIEW จะมี Front Panel ซึ่งเปรียบเสมือนได้กับสิ่งที่ผู้ใช้จะเห็นและควบคุมการทำงาน ผู้ใช้สามารถสร้างรูปแบบขึ้นเองได้อย่างรวดเร็วเพราะ LabVIEW มีส่วนประกอบต่างๆ ที่ใช้สำหรับออกแบบหน้าจอมากมาย เช่น จอแสดงผลแบบออฟิโลลโครบ, ปุ่มหมุน (Dial) และสวิตช์ เป็นต้น โดย LabVIEW จะแสดงผลและควบคุมการทำงานผ่านทางคอมพิวเตอร์

พื้นที่ส่วนเขียนโปรแกรมจะเรียกว่า Block Diagram เปรียบเสมือนกับ Hardware ภายในเครื่องมือวัด โดย LabVIEW จะเขียนโปรแกรมโดยอาศัยรูปภาพ ดังตัวอย่างในรูปที่ 2.3



รูปที่ 2.3 Block Diagram ของ LabVIEW

LabVIEW อาศัยหลักการทำงานของเครื่องมือวัดหรือการวัดคุม ทำให้ผู้ใช้สามารถออกแบบตามที่ผู้ใช้ต้องการดังรูปที่ 2.4 หลักการดังกล่าวแบ่งออกเป็น 3 ส่วนใหญ่ๆ คือ



รูปที่ 2.4 Block Diagram เครื่องมือวัดที่สร้างจาก LabVIEW

1. Acquisition ซึ่งเป็นส่วนที่รับข้อมูล (Input) จากสิ่งแวดล้อมภายนอกเข้าสู่ระบบในที่นี้คือคอมพิวเตอร์โดยข้อมูลที่เข้าสู่ระบบนี้อาจมาจากการ์ด DAQ (สำหรับสัญญาณทางไฟฟ้า)

2. Analysis หลังจากที่ได้รับข้อมูลแล้ว อาจจะผ่านฟังก์ชันในการวิเคราะห์ข้อมูล ซึ่งจะแสดงผลในรูปที่สื่อความหมายในสิ่งที่ผู้ใช้งานสามารถนำไปแสดงแทนสื่อที่วัดได้และใช้งานได้

3. Presentation คือ การแสดงผลในรูปแบบที่เป็นประโยชน์ต่อผู้ใช้งาน โดยอาจแสดงบนหน้าจอคอมพิวเตอร์ เช่น DMM (Digital Multimeter) แสดงผลเฉพาะที่วัดได้โดยไม่จำเป็นต้องรู้ความสำคัญกับเวลา หรือ Spectrum Analysis จะแสดงสัญญาณในรูปความถี่หรือการพิมพ์ออกมาเป็นรายงานหรือเก็บข้อมูลในฮาร์ดดิสก์

2.2.2 ส่วนประกอบต่างๆ ใน LabVIEW

โปรแกรมที่เขียนขึ้นมาโดย LabVIEW จะเรียกว่า Virtual Instrument (VI) เพราะลักษณะที่ปรากฏทางจอภาพเมื่อผู้ใช้ใช้งานจะเหมือนกับเครื่องมือหรืออุปกรณ์ทางวิศวกรรม ในขณะเดียวกันหลังฉากของอุปกรณ์เสมือนจริงเหล่านั้นจะเป็นการทำงานของฟังก์ชัน, Subroutines และโปรแกรมหลักเหมือนกับภาษาทั่วไป สำหรับ VI หนึ่งๆ จะประกอบด้วยส่วนประกอบ 3 ส่วน คือ

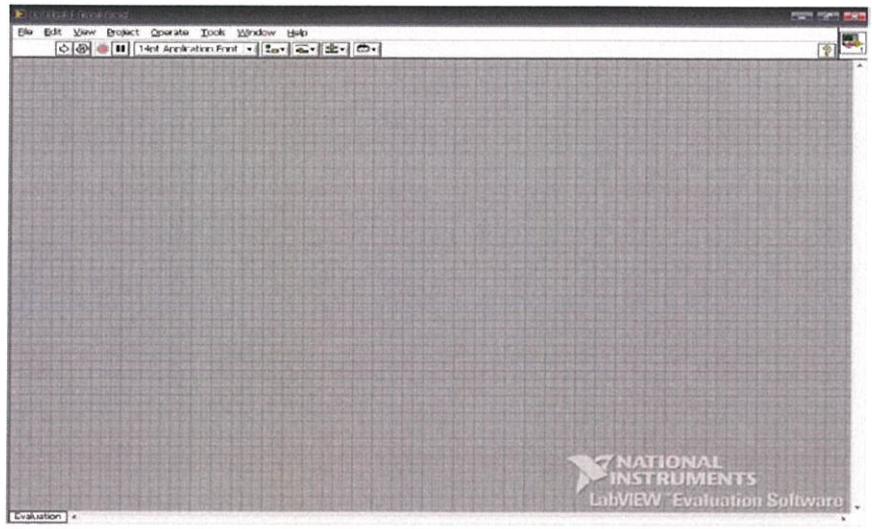
2.2.2.1 Front Panel

2.2.2.2 Block Diagram

2.2.2.3 Icon และ Connector

ทั้งสามส่วนนี้จะประกอบกันขึ้นมาเป็นอุปกรณ์เสมือนจริง ลักษณะและหน้าที่ของส่วนประกอบทั้ง 3 มีดังต่อไปนี้

2.2.2.1 Front Panel หรือหน้าปัทม์ จะเป็นส่วนที่ใช้สื่อความกันระหว่างผู้ใช้กับโปรแกรม (หรือที่นิยมเรียก User Interface) โดยทั่วไปจะมีลักษณะเหมือนกับหน้าปัทม์ของเครื่องมือหรือ อุปกรณ์ที่ใช้งานด้านการวัดทั่วๆ ไป โดยทั่วไปจะประกอบด้วย สวิตช์ปิด - เปิด, ปุ่ม, ปุ่มกด จอแสดงผลหรือแม้แต่ค่าที่ผู้ใช้สามารถกำหนด สำหรับผู้ที่คุ้นเคยกับการเขียนโปรแกรมประเภท Visual ทั้งหลายคงจะเข้าใจกันดีว่า Front Panel นี้จะเปรียบเสมือนเป็น GUI ของโปรแกรมหรือ VI ดังรูปที่ 2.5



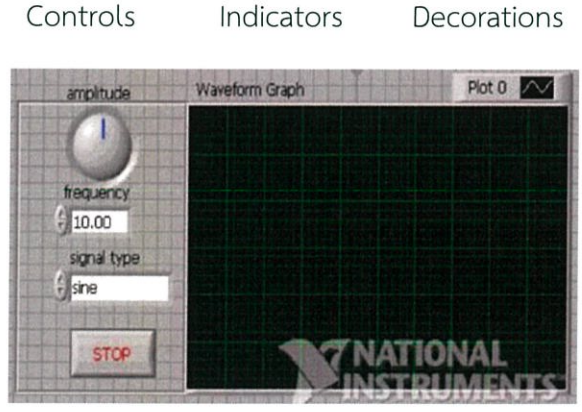
รูปที่ 2.5 Front Panel ของ LabVIEW

Object ที่อยู่บน Front Panel จะมีอยู่ 3 ประเภท คือ

1. Control คือ ประเภทที่รับค่าจากผู้ใช้ (Input) ซึ่งผู้ใช้สามารถพิมพ์ค่าลงไป หรือใช้เมาส์คลิกเพื่อเปลี่ยนแปลงค่าได้ เช่น ปุ่มหมุน ปุ่มเลื่อน สวิตช์ เป็นต้น

2. Indicators คือ ประเภทที่ใช้แสดงค่าต่างๆ เท่านั้น (Output) ผู้ใช้ไม่สามารถแก้ไขได้ เช่น กราฟ มิเตอร์ LED

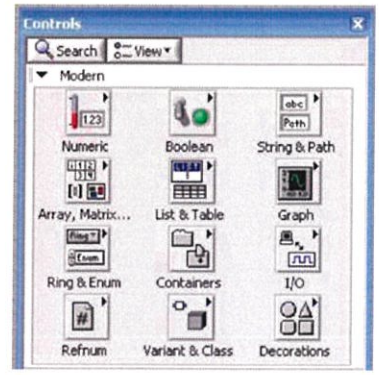
3. Decorations เป็น Object ที่ไม่เกี่ยวข้องกับโปรแกรม และ Code บน Block Diagram แต่มีไว้เพื่อความสวยงามเป็นระเบียบของ Front panel เท่านั้น ลักษณะของ Front Panel แสดงดังรูปที่ 2.6



รูปที่ 2.6 Object ที่อยู่บน Front Panel ของ LabVIEW

เครื่องมือที่ใช้ออกแบบ Front Panel

เครื่องมือที่ใช้ในการออกแบบ Front Panel จะประกอบไปด้วย Control Palette และ Tools Palette ซึ่ง LabVIEW มี Controls Palette ที่ใช้ในการออกแบบ Front Panel แสดงดังรูปที่ 2.7 ซึ่งเป็นส่วนที่ติดต่อกับผู้ใช้งาน (User Interface) โดยจะจัดเป็นกลุ่มต่างๆ เช่น กลุ่มของตัวเลข (Numeric) ซึ่งภายในกลุ่มจะมี Control และ Indicator ต่างๆ ที่เกี่ยวกับตัวเลข



รูปที่ 2.7 Controls Palette ที่ใช้ในการออกแบบ Front Panel

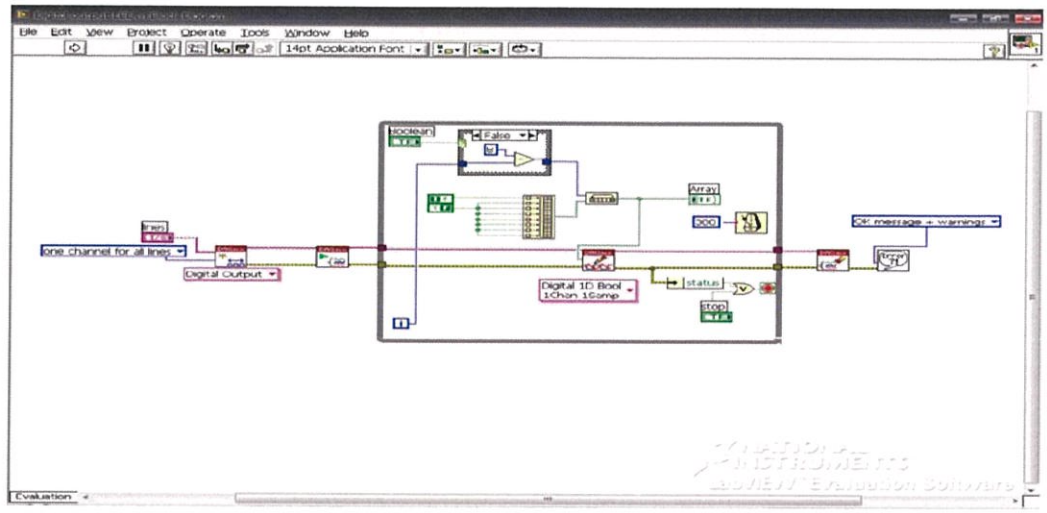
Tools Palette คือ เครื่องมือที่ใช้ในการพัฒนาโปรแกรม ซึ่งจะใช้ทั้งการออกแบบ Front Panel และ Block Diagram ในส่วนนี้จะกล่าวถึง Tools Palette สำหรับออกแบบ Front Panel ดังรูปที่ 2.8

รูปที่ 2.8 Tools Palette ที่ใช้ในการออกแบบ Front Panel

2.2.2.2 Block Diagram

เพื่อให้เกิดความเข้าใจง่ายขึ้น อาจมอง Block Diagram นี้เป็นเสมือนกับ Source Code หรือโปรแกรมของ LabVIEW ซึ่งปรากฏว่าอยู่ในรูปของภาษา G ซึ่ง Block Diagram ถือว่าเป็น Executable Program คือ สามารถที่จะทำงานได้ทันที และข้อดีอีกประการหนึ่งก็คือ LabVIEW จะมีการตรวจสอบความผิดพลาดของโปรแกรมตลอดเวลา ทำให้โปรแกรมจะทำงานได้ก็ต่อเมื่อไม่มีข้อผิดพลาดในโปรแกรมเท่านั้น โดยผู้ใช้สามารถที่จะดูรายละเอียดของความผิดพลาดแสดงให้เห็นได้ตลอดเวลา ทำให้การเขียนโปรแกรมนั้นง่ายขึ้นมาก

ส่วนประกอบภายใน Block Diagram นี้จะประกอบด้วย ฟังก์ชัน ค่าคงที่ โปรแกรมควบคุม การทำงานหรือโครงสร้าง จากนั้นในแต่ละส่วนเหล่านี้ซึ่งจะปรากฏในรูปของ Block จะได้รับการต่อสาย (Wire) สำหรับ Block ที่เหมาะสมเข้าด้วยกัน เพื่อกำหนดลักษณะการไหลของข้อมูลระหว่าง Block เหล่านั้น ทำให้ข้อมูลได้รับการประมวลผลตามที่ต้องการและแสดงผลออกมาดังรูปที่ 2.9



รูปที่ 2.9 ตัวอย่าง Block Diagram

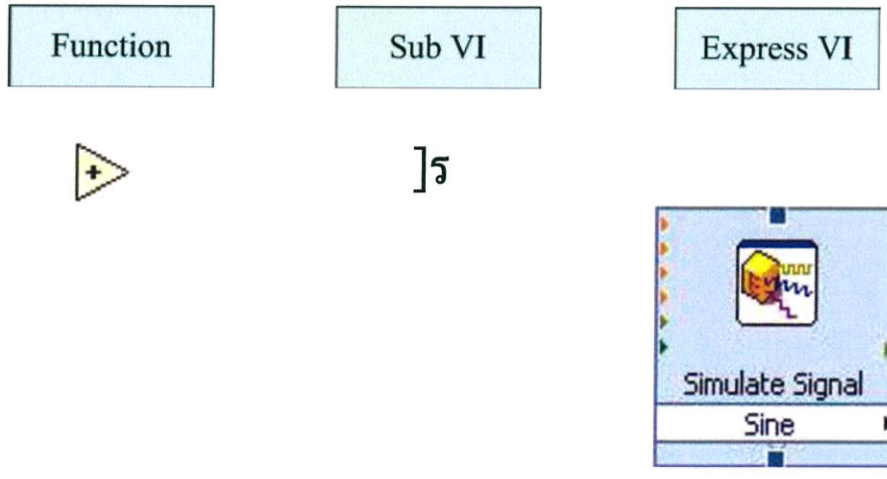
1. Block Diagram Node

Node คือ รูป Icon ที่อยู่บน Block Diagram ซึ่งมี Input และ/หรือ Output และจะทำงานตามหน้าที่เมื่อมีการรันโปรแกรม โดยแบ่งเป็น 3 ชนิดหลัก ได้แก่

1) Function คือ Node ที่มีหน้าที่พื้นฐานของคอมพิวเตอร์ ซึ่งไม่สามารถที่จะเจาะเข้าไปดูรายละเอียดภายในได้อีก เช่น การบวก การคูณ

2) Sub VIs หรือในภาษาทางซอฟต์แวร์อาจจะเรียกว่า Subroutine หรือ Subprogram คือ โปรแกรมย่อยที่ถูกเขียนขึ้นมาเพื่อถูกนำมาเรียกใช้ในอีกโปรแกรมหนึ่ง สามารถเปิดเข้าไปลูป Front Panel และ Block Diagram ได้เมื่อ Double Click ที่ Icon ของมัน

3) Express VIs เป็น Sub VIs ประเภทพิเศษ คือ เมื่อเลือก Express VI มาวางบน Block Diagram มันจะปรากฏหน้าต่าง Configuration ขึ้นมาเพื่อให้เข้าไปเปลี่ยนค่า Parameters ต่างๆ ตามต้องการ และเมื่อเปลี่ยนค่าเสร็จมันก็จะสร้างโค้ดไว้โดยอัตโนมัติตามที่ได้ตั้งค่าไว้ ซึ่งความสามารถของ Express VI นี้ทำให้ไม่จำเป็นต้องต่อสาย Input เลย เพราะ Parameter ทั้งหมดได้ถูกสร้างขึ้นมาแล้วถูกเก็บไว้ภายในเรียบร้อยแล้ว จึงทำให้การเขียน LabVIEW ง่ายและเร็วขึ้นมาก สังเกตง่ายๆ Express VI จะมี Icon ขนาดใหญ่ที่มีพื้นหลังเป็นสีฟ้า ดังรูปที่ 2.10



รูปที่ 2.10 ตัวอย่าง Block Diagram Node

2. เครื่องมือที่ใช้ในการเขียนโปรแกรมบน Block Diagram

LabVIEW ใช้ Functions Palette ซึ่งจะมี Function และ SubVI ต่าง ๆ ที่มีอยู่แล้วให้ผู้ใช้เลือกใช้ โดย Function และ SubVI จัดเป็นกลุ่มๆ เช่น Numeric Function จะมี Function ต่างๆ เกี่ยวกับตัวเลข เช่น บวก ลบ คูณ หาร

เปิด Tools Palette โดยการ Click ที่ Window » Show Tools Palette หรือกด Shift Right-Click แล้ว Click เลือก Tool ที่ต้องการใช้ (ถ้า Automatic Tools Selection เปิดอยู่แล้ว (LED) เป็นสีเขียว ให้ Click เพื่อเปิด Automatic Tool Selection ซึ่งมีเฉพาะใน Version 6.1 ขึ้นไป) Tool ที่ใช้ มีดังต่อไปนี้ ดังรูปที่ 2.11

รูปที่ 2.11 เครื่องมือ Tools Palette

1) Operation Tool ใช้ในการเปลี่ยนแปลงค่าหรือเลือกค่าคงที่ใน Block Diagram

2) Position/Size/Select ใช้ในการเลือก/เคลื่อนย้าย/จัดขนาดของสิ่งที่สร้างขึ้นบน Block Diagram

3) Edit Text Tool ใช้ในการแก้ไขข้อความที่เป็นตัวอักษร หรือเพิ่มข้อความลงบน Front Panel

4) Wiring Tool ใช้ในการโยงสาย (Wiring) ระหว่าง Terminal หรือ Node ซึ่งสายที่โยงนี้จะเป็นเส้นทางเดินของข้อมูล

2.2.2.3 Icon และ Connector

เปรียบเสมือนโปรแกรมย่อย Subroutine ในโปรแกรมปกติทั่วๆ ไป โดย Icon จะหมายถึง Block Diagram ตัวหนึ่งที่มีการส่งข้อมูลเข้าและออกผ่านทาง Connector ซึ่งใน LabVIEW จะเรียก Subroutine นี้ว่า SubVPU ข้อดีของการเขียนโปรแกรมด้วยภาษา G นี้ก็คือ สามารถสร้าง VI ทีละส่วนขึ้นมาให้ทำงานด้วยตัวเองได้อย่างอิสระ จากนั้นในภายหลังหากต้องการก็สามารถเขียนโปรแกรมอื่นขึ้นมาเพื่อเรียกใช้งาน VI ที่เคยสร้างขึ้นก่อนหน้านี้ทีละตัว ซึ่งทำให้ VI ที่เขียนขึ้นก่อนกลายเป็น SubVI การเขียนในลักษณะนี้เรียกว่า เขียนเป็น Module

สำหรับลักษณะทั่วไปของ Icon และ Connector จะแสดงในรูปต่อไปนี้ จะเห็นว่าเมื่อแสดงในรูปของ Connector จะพบว่ามีช่องต่อข้อมูลหรือที่เรียกว่า Terminal ปรากฏให้เห็น

คำศัพท์ต่างๆ ที่ใช้กันใน LabVIEW นี้ออกจะแตกต่างจากที่ใช้กันในภาษาการเขียนโปรแกรม ตัวหนังสือทั่วๆ ไปในหลายๆ ด้าน ดังนั้นเพื่อให้ผู้ที่เริ่มใช้ LabVIEW เข้าใจถึงศัพท์ต่างๆ ที่ใช้ในโปรแกรม จึงเปรียบเทียบศัพท์ใช้ใน LabVIEW กับโปรแกรมพื้นฐานทั่วๆ ไป ตามตารางที่ 2.1 ดังนี้

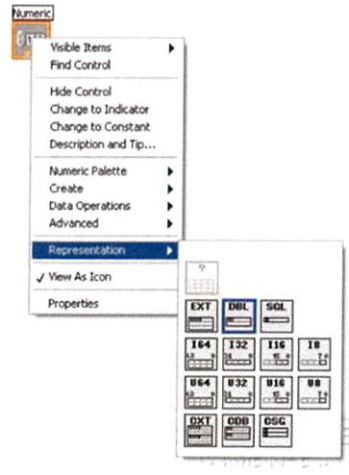
ตารางที่ 2.1 เปรียบเทียบคำศัพท์ที่ใช้ในการเขียนโปรแกรม

LabVIEW	โปรแกรมพื้นฐาน	หน้าที่
VI	Program	ตัวโปรแกรมหลัก
Function	Function	ฟังก์ชันสำเร็จรูปที่สร้างขึ้นมากับโปรแกรมนั้น เช่น sin, log เป็นต้น
Sub VI	Subroutine	โปรแกรมย่อยที่ถูกเรียกใช้โดยโปรแกรมหลัก
Front Panel	User Interface	ส่วนที่ติดต่อกับผู้ใช้
Block Diagram	Program Code	การเขียนตามขั้นตอนของที่แต่ละโปรแกรมกำหนดขึ้น

2.2.3 ประเภทของข้อมูล

ในการเขียนโปรแกรมทั่วๆ ไปจะต้องมีการประกาศตัวแปรก่อนที่จะใช้ตัวแปรนั้น แต่สำหรับโปรแกรม LabVIEW มันจะจัดการให้เองหมดโดยผู้ใช้ไม่จำเป็นต้องทำเอง เพียงแค่เลือกประเภทของข้อมูลที่มาวางบนโค้ดให้ถูกต้องเท่านั้น ประเภทของข้อมูลภายใน LabVIEW ก็มีหลายอย่างที่เหมือนกับโปรแกรมในภาษาอื่นๆ และยังมีอีกบางประเภทที่ใช้ใน LabVIEW เท่านั้น โปรแกรม LabVIEW แบ่งข้อมูลเป็น 6 ชนิด ดังนี้

2.2.3.1 Numeric คือ ข้อมูลประเภทตัวเลขมีทั้งจำนวนเต็ม ซึ่งใน Block Diagram จะเห็นเป็นสีน้ำเงิน และจำนวนทศนิยมจะเห็นเป็นสีส้ม และสามารถเปลี่ยนไปมาได้โดยการคลิกขวาที่ตัวเลขนั้น แล้วเลือก Representation และเลือกประเภทตัวเลขได้เลย แสดงดังรูปที่ 2.12

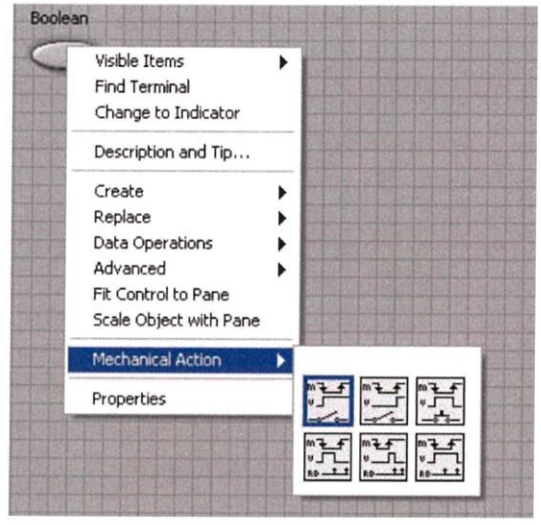


รูปที่ 2.12 ข้อมูลประเภท Numeric

2.2.3.2 Boolean คือ ข้อมูลประเภทที่มีสองค่าคือ True และ False บน Block Diagram จะแสดงข้อมูลเป็นสีเขียว และสำหรับ Front Panel ตัว Boolean จะมีลักษณะเป็นตัว Control หรือ สวิตช์ ถ้าเป็น Output ก็จะเป็น LED หรือหลอดไฟประเภทต่างๆ ดังรูปที่ 2.13

2.2.3.3 String คือ ข้อมูลประเภทที่เป็นตัวอักษร Icon จะแสดงเป็นสีชมพู สำหรับการแสดงผลจะมีอยู่ 4 แบบ คือ

- Normal Display คือ การแสดงปกติ
- Code Display คือ การแสดงแบบโค้ด มีประโยชน์สำหรับแสดงตัวอักษรที่ตาเปล่ามองไม่เห็น การเว้นวรรค แท็บ หรือการขึ้นบรรทัดใหม่
- Password Display คือ การแทนตัวอักษรด้วย *
- Hex Display คือ การแสดงผลเป็นรหัสเลขฐานสิบหก



รูปที่ 2.13 ข้อมูลประเภท Boolean

2.2.3.4 Enum คือ ข้อมูลประเภทแสดงให้ผู้ใช้เห็นเป็นตัวหนังสือ แต่ค่าจริงของมันคือตัวเลข ดังนั้นบน Block Diagram จึงมองเห็นข้อมูลประเภทนี้เป็นสีน้ำเงิน ซึ่งเหมือนกับจำนวนเต็ม

2.2.3.5 Dynamic เป็นข้อมูลที่อยู่ในรูปของ Waveform บน Block Diagram ถูกแสดงด้วยสีน้ำเงินเข้มซึ่งภายในจะประกอบด้วย Array ของเวฟฟอร์ม Time Stamp ชื่อของสัญญาณ ข้อมูลประเภท Dynamic นี้ส่วนใหญ่ใช้ใน Express VI จำพวกการอ่าน กำเนิด และวิเคราะห์สัญญาณ

2.2.3.6 Time Stamp เป็นข้อมูลที่ประกอบด้วยวันที่ และเวลาที่มีความละเอียดถึงมิลลิวินาที Time Stamp บน Block Diagram จะมีหน้าตาที่เป็นสีน้ำตาลเส้นหนา สามารถนำมาแปลงให้เป็น วันที่ เวลา แบบ String ได้

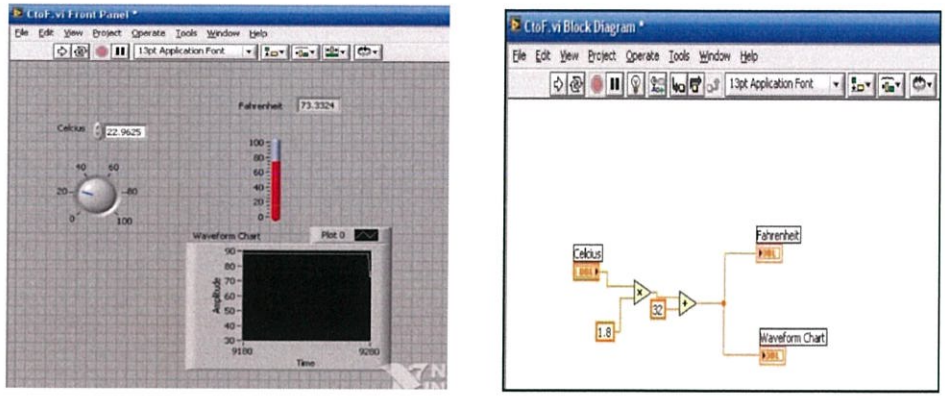
2.2.4 หลักการทำงานของ Dataflow Programming

ก่อนจะเริ่มพัฒนาโปรแกรมมาดูหลักการทำงานของโปรแกรมในรูปแบบ Dataflow และรูปแบบข้อมูล (Data Type) ซึ่งเป็นสิ่งสำคัญในการพัฒนาโปรแกรม

โปรแกรมที่เขียนขึ้นด้วย LabVIEW จะทำงานโดยอาศัยหลักการของ Dataflow ซึ่งมีหลักการดังต่อไปนี้

1. ฟังก์ชันหรือ SubVI จะทำงานเมื่อมีข้อมูล (Input)
2. เมื่อฟังก์ชันหรือ SubVI ทำงานเสร็จจะให้ข้อมูล (Output) ไปยังฟังก์ชันหรือ SubVI อื่นๆ ที่ต้องการข้อมูล ข้อมูลจะถูกส่งผ่านโดยสาย (Wire)

ในรูปที่ 2.14 เป็นตัวอย่างการเขียนโปรแกรม LabVIEW อุณหภูมิเป็นองศาฟาเรนไฮท์ โดยแสดงผลเป็นแบบตัวเลขและกราฟ



(a) โปรแกรมที่เขียนขึ้น (b) จอแสดงผลที่ต้องการ

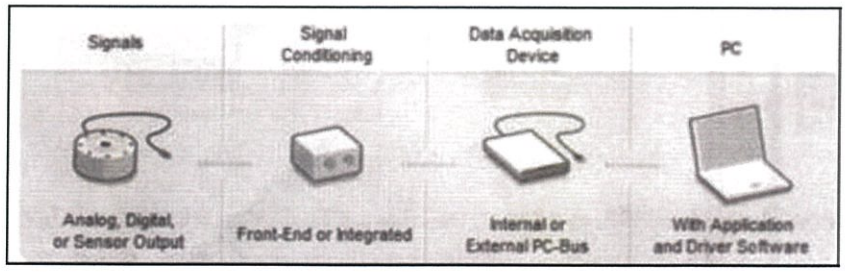
รูปที่ 2.14 การทำงานของ Dataflow Programming

จากรูปที่ 2.14 ขั้นตอนการทำงานของโปรแกรมจะเป็นดังนี้

1. ฟังก์ชันคูณ (Multiply) จะทำงานก่อนฟังก์ชันบวก เพราะฟังก์ชันคูณมีข้อมูล (Input) พร้อม แต่ฟังก์ชันบวกต้องรอ Output จากฟังก์ชันคูณ (หลักการ Dataflow ข้อ 1)
2. หลังจากฟังก์ชันคูณทำงานเสร็จ จะส่งผ่านข้อมูลไปยังฟังก์ชันบวก (หลักการ Dataflow ข้อที่ 2 และ 3)
3. ฟังก์ชันบวกทำงาน (หลักการข้อ 1) เพราะมีข้อมูลพร้อม
4. หลังจากฟังก์ชันบวกทำงาน จะส่งผลลัพธ์ไปให้ Terminal ที่ 2 คือ Fahrenheit และ Waveform Chart พร้อมกัน (หลักการข้อ 2 และ 3)

2.2.5 ภาพรวมของระบบ DAQ บน PC

ระบบ DAQ บน PC คือ กระบวนการในการอ่านค่าสัญญาณทางไฟฟ้าแล้วนำไปเก็บไว้ในหน่วยความจำบน PC เพื่อการวัดวิเคราะห์ จัดเก็บ หรือแสดงผลต่อไป ขึ้นอยู่กับวัตถุประสงค์ในการสร้าง VI ของแต่ละแอพพลิเคชัน ซึ่งระบบ PC จะมีส่วนประกอบอยู่ 4 ส่วนด้วยกัน คือ ส่วนของแหล่งสัญญาส่วนฮาร์ดแวร์ Signal Conditioning ส่วนอุปกรณ์ของซอฟต์แวร์ PC ดังรูปที่ 2.15

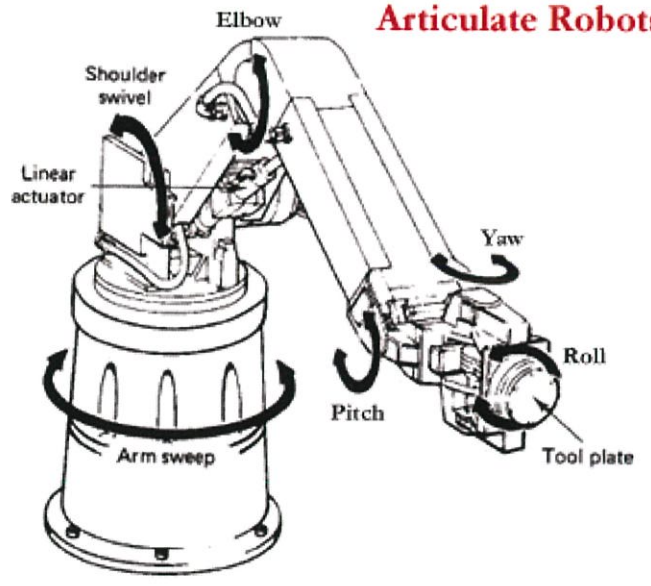


รูปที่ 2.15 ส่วนประกอบของระบบ DAQ

สัญญาณที่วัดอาจเป็นสัญญาณอนาลอก (+10V หรือ -10V หรือ 4-20 mA) หรือสัญญาณดิจิตอลหรืออาจจะมาจากเซนเซอร์ ซึ่งสัญญาณเหล่านี้จะถูกต่อสายเข้าไปในส่วนฮาร์ดแวร์อิเล็กทรอนิกส์ที่เรียกว่า Signal Conditioning ซึ่งจะทำหน้าที่ในการปรับปรุงสัญญาณให้เหมาะสมก่อนที่จะวัดด้วยฮาร์ดแวร์ DAQ ต่อไป เช่น ถ้าสัญญาณมีขนาดเล็กเกินไปก็จะต้องขยายให้ใหญ่ขึ้น หรือสัญญาณเข้ามาใหญ่เกินไปก็ต้องลดทอนสัญญาณให้เล็กลง เป็นต้น (ถ้าสัญญาณมีลักษณะเหมาะสมแล้วก็ไม่จำเป็นต้องนำ Signal Conditioning มาใช้ทุกครั้งเสมอไป) สำหรับฮาร์ดแวร์ DAQ แบบพื้นฐานจะสามารถวัดสัญญาณแรงดันไฟฟ้าอยู่ในช่วง +10V หรือ -10V เท่านั้น ซึ่งฮาร์ดแวร์นี้จะถูกติดตั้งบน PC ที่มีไดรฟ์เวอร์ และมีแอพพลิเคชั่นที่เขียนด้วยซอฟแวร์สำหรับพัฒนาระบบ เช่น LabVIEW

2.3 Articulated Arm Robot (Revolute)

หุ่นยนต์ที่มีลักษณะทุกแกนการเคลื่อนที่เป็นแบบหมุน (Revolute Joint) มีรูปแบบการเคลื่อนที่คล้ายกับแขนของมนุษย์ แต่มีองศาการหมุนที่มากกว่าคือ สามารถหมุนได้รอบแกนทุกแกน ดังรูปที่ 2.16



รูปที่ 2.16 Articulated Arm Robot
[**ที่มา :** https://www.applicadthai.com/business/sites/default/files/articles/Industrial_Robot_6.gif]

เนื่องจากการที่หุ่นยนต์สามารถหมุนได้ทุกแกน และพื้นที่การทำงานของหุ่นยนต์มีมาก ทำให้มีการเคลื่อนที่ในลักษณะท่าทางต่างๆ ได้มาก ทำให้สามารถเข้าถึงงานได้หลากหลายรูปแบบ ตั้งแต่งานที่มีพื้นที่การทำงานมากจนไปถึงงานที่มีพื้นที่การทำงานน้อยหรือแคบ มีระบบการควบคุมตำแหน่งที่ละเอียด ทำให้สามารถใช้งานกับงานที่ต้องการความละเอียดและความแม่นยำสูงได้

2.3.1 ข้อมูลของหุ่นยนต์

เนื่องจากหุ่นยนต์ที่นำมาใช้เป็นหุ่นยนต์เก่าของทางโรงงาน ทำให้จำเป็นต้องใช้หุ่นยนต์ Kawasaki รุ่น RS005L ซึ่งสามารถรับน้ำหนักสูงสุดที่ปลายแขนได้ 5 กิโลกรัม

2.3.2 ข้อมูลและขนาดการเคลื่อนที่ของหุ่นยนต์

เมื่อได้รุ่นของหุ่นยนต์ที่ใช้แล้ว จำเป็นต้องทราบขนาดและความเร็วของการเคลื่อนที่ของหุ่นยนต์ Kawasaki รุ่น RS005L มีข้อมูลและขนาดการเคลื่อนที่ดังตารางที่ 2.2

ตารางที่ 2.2 ข้อมูลและขนาดการทำงานของหุ่นยนต์
[**ที่มา :** https://robotics.kawasaki.com/en1/products/robots/small-medium-payloads/RS005L/]

RS005L SPECIFICATIONS

RS005L SPECIFICATIONS			
Degrees of Freedom	6 axes		
Payload	5 kg		
Horizontal Reach	903 mm		
Repeatability	±0.03 mm		
Work Envelope (degrees) & Maximum Speed (degrees/s)	Axis	Motion Range	Maximum Speed
	JT1	±180°	300°/s
	JT2	+135° ~ -80°	300°/s
	JT3	+118° ~ -172°	300°/s
	JT4	±360°	460°/s
	JT5	±145°	460°/s
	JT6	±360°	740°/s
Wrist Load Capacity	Axis	Max Torque	Moment of Inertia
	JT4	12.3 N•m	0.4 kg•m2
	JT5	12.3 N•m	0.4 kg•m2
	JT6	7.0 N•m	0.12 kg•m2
Mass	37 kg (excluding Options)		
Body Color	Kawasaki Standard, epoxy paint		
Installation	Floor, wall, ceiling		
Protection Classification	Wrist: IP67 Base: IP65 *Equivalent		
Controller	America		
	Europe	F60	
	Japan & Asia		

2.3.3 ภาษาและคำสั่งที่ใช้ในการเขียนหุ่นยนต์

การเขียนโปรแกรมเพื่อควบคุมการทำงานของหุ่นยนต์ Kawasaki จะใช้ภาษาที่เรียกว่า AS Language โดยการทำงานของโปรแกรมจะแบ่งเป็นส่วนหลักๆ 4 ส่วน และในทุกๆ ส่วนจะทำไปพร้อมๆ กัน 4 ส่วนหลักๆ ของโปรแกรม ได้แก่

2.3.3.1 Main Program : เป็นโปรแกรมหลักสำหรับเขียนให้หุ่นยนต์เคลื่อนที่ไปตามรูปแบบที่ต้องการ และคอยรับสัญญาณมาจาก PC Program

2.3.3.2 PC Program : เป็นโปรแกรมที่ทำงานไปพร้อมๆ กับ Main Program คอยรับสัญญาณมาจากอุปกรณ์ภายนอกผ่านทาง I/O และส่งสัญญาณต่อให้กับ Main Program

2.3.3.3 Teach Program : เป็นโปรแกรมที่เอาไว้สำหรับจดจำตำแหน่งที่หุ่นยนต์ต้องเคลื่อนที่ไป

2.3.3.4 Inter Panel Program : เป็นโปรแกรมสำหรับเขียนหน้าจอเพื่อสร้างเป็นจอแสดงผล หรือปุ่มกดเพื่อส่งสัญญาณไปให้ PC Program

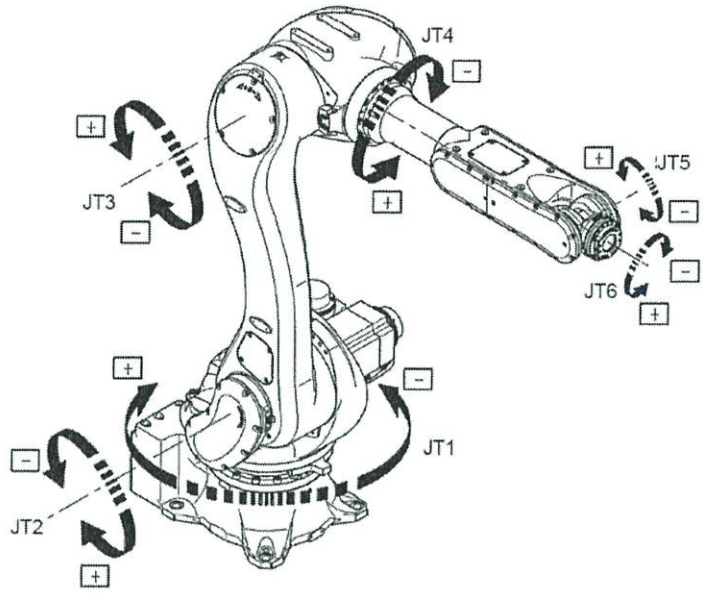
การเขียนโปรแกรมการเคลื่อนที่ของหุ่นยนต์ในโปรแกรมหลัก จะมีคำสั่งการเคลื่อนที่เฉพาะสำหรับหุ่นยนต์ ดังนี้

- BRAKE : สำหรับหยุดการเคลื่อนที่ของหุ่นยนต์
- BREAK : สำหรับหยุดโปรแกรมจนกว่าจะเคลื่อนที่เสร็จ
- DELAY : สำหรับหยุดหุ่นยนต์เป็นระยะเวลาหนึ่ง
- DRAW : สำหรับเคลื่อนที่หุ่นยนต์ไปตามระยะที่กำหนด
- IF : สำหรับตั้งเงื่อนไข (IF...END)
- ELSE : เป็นเงื่อนไขที่จะทำต่อจาก IF (IF...ELSE...END)
- END : ใช้สำหรับจบเงื่อนไข หรือจบโปรแกรม
- GOTO : เมื่อเจอ GOTO โปรแกรมจะข้ามไปตามที่กำหนดไว้
- HOME : หุ่นยนต์จะเคลื่อนที่ไปตำแหน่ง HOME
- JMOVE : หุ่นยนต์จะเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่งตามข้อหมุน
- LMOVE : หุ่นยนต์จะเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่งตามเส้นตรง
- RESET : สำหรับ RESET สัญญาณทั้งหมด
- SIG : สำหรับรับสัญญาณเข้าโปรแกรม
- SIGNAL : สำหรับส่งสัญญาณออก
- SPEED : สำหรับตั้งค่าความเร็วมีหน่วยเป็น %
- LOAD : สำหรับโหลดโปรแกรมเข้าคอนโทรลเลอร์
- SAVE : สำหรับโหลดโปรแกรมออกจากคอนโทรลเลอร์
- ABORT : สำหรับยกเลิกการทำงาน PC โปรแกรม
- KILL : สำหรับลบโปรแกรม
- PCEX : สำหรับเรียก PC โปรแกรมมาใช้งาน
- AS : สำหรับเข้าสู่ระบบหุ่นยนต์

2.3.4 การเคลื่อนที่ของหุ่นยนต์

การเคลื่อนที่ของหุ่นยนต์ Kawasaki มีอยู่ด้วยกัน 3 แบบ ได้แก่ แบบเคลื่อนที่ตามข้อหมุนของหุ่นยนต์ (Joint) แบบเคลื่อนที่ตามแกน X, Y, Z (Trans) และแบบเคลื่อนที่ตามแกนของอุปกรณ์ที่ติดตั้ง (Tool)

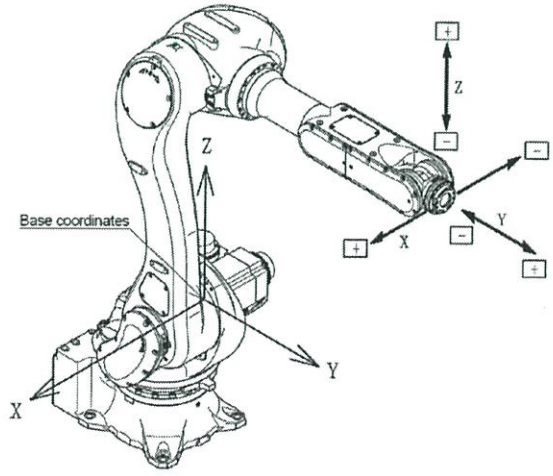
1. แบบเคลื่อนที่ตามข้อหมุนของหุ่นยนต์ (Joint) จะเคลื่อนที่ตามข้อต่างๆ ของหุ่นยนต์ ซึ่งจะมีอยู่ด้วยกัน 6 ข้อ ดังรูปที่ 2.17



รูปที่ 2.17 แบบเคลื่อนที่ตามข้อหมุนของหุ่นยนต์ (Joint)
[**ที่มา :** http://platforma.astor.com.pl/files/getfile/id/12053]

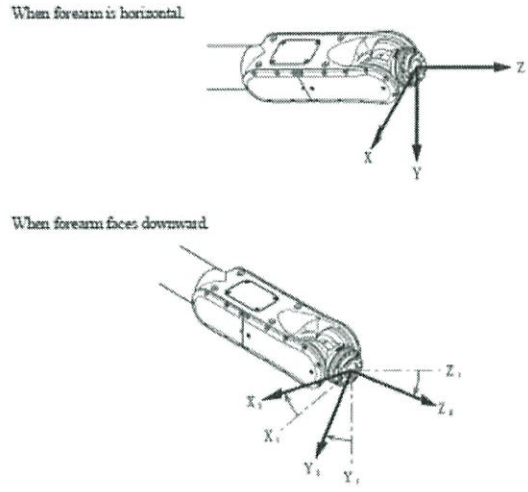
- JT1 คือ การหมุนรอบแกน 1 ของหุ่นยนต์
- JT2 คือ การหมุนรอบแกน 2 ของหุ่นยนต์
- JT3 คือ การหมุนรอบแกน 3 ของหุ่นยนต์
- JT4 คือ การหมุนรอบแกน 4 ของหุ่นยนต์
- JT5 คือ การหมุนรอบแกน 5 ของหุ่นยนต์
- JT6 คือ การหมุนรอบแกน 6 ของหุ่นยนต์

2. แบบเคลื่อนที่ตามแกน X, Y, Z (Trans) หุ่นยนต์จะเคลื่อนที่ตาม Base พื้นฐาน X, Y และ Z หรือก็คือ ยึดฐานของหุ่นยนต์เป็นหลัก และเคลื่อนที่ไปตามแนวแกน X, Y และ Z ดังรูปที่ 2.18



รูปที่ 2.18 แบบเคลื่อนที่ตามแกน X, Y, Z (Trans)
[**ที่มา :** http://platforma.astor.com.pl/files/getfile/id/12053]

3. แบบเคลื่อนที่ตามแกนของอุปกรณ์ที่ติดตั้ง (Tool) จะยึดอุปกรณ์ที่ติดตั้งอยู่บริเวณปลายของหุ่นยนต์เป็นหลัก เพื่อเคลื่อนที่ตามแนวแกน X, Y และ Z ดังรูปที่ 2.19



รูปที่ 2.19 แบบเคลื่อนที่ตามแกนของอุปกรณ์ที่ติดตั้ง (Tool)
[**ที่มา :** http://platforma.astor.com.pl/files/getfile/id/12053]

2.3.5 คอนโทรลเลอร์สำหรับหุ่นยนต์

ในการเคลื่อนที่ของหุ่นยนต์จะต้องมีคอนโทรลเลอร์ควบคุม สำหรับอัพโหลดโปรแกรมและเป็นตัวกลาง I/O ในการรับส่งข้อมูลสัญญาณระหว่างอุปกรณ์ภายนอกกับหุ่นยนต์

สำหรับหุ่นยนต์ Kawasaki รุ่น E74 มีข้อมูลอุปกรณ์ดังตารางที่ 2.3

ตารางที่ 2.3 ข้อมูล E Controller
[**ที่มา :** https://robotics.kawasaki.com/en1/products/controllers/e-controllers/]
E Controllers SPECIFICATIONS

E CONTROLLERS SPECIFICATIONS				
Japan & Asia		E73/74	E94	E10/12/14/20/22/23/24
Type of driving		Full digital servo system	Full digital servo system	Full digital servo system
Teaching method		Easy operation teaching or AS language programming	Easy operation teaching or As language programming	Easy operation teaching or As language programming
Teach pendant		LCD teach pendant	Color LCD teach pendant	Color LCD teach pendant
Memory capacity (MB)		8	8	8
I/O signals	**External operation**	Emergency stop, Hold etc.	Emergency stop, Hold etc.	Emergency stop, Hold etc.
	Input (Channels)	32 (max. 96)	32 (max. 96)	32 (max. 128)
	Output (Channels)	32 (max. 96)	32 (max. 96)	32 (max. 128)
Structure		Enclosed structure with indirect cooling system	Open structure with direct cooling system (Option: Enclosed structure)	Enclosed structure with indirect cooling system
Mass (kg)		30	40	180/180/180/95/95/110/110

2.4 หลักการพื้นฐานในการสื่อสารข้อมูลในส่วนต่างๆ

2.4.1 การสื่อสารระหว่างหุ่นยนต์กับ LabVIEW Program

ในการสื่อสารระหว่างหุ่นยนต์ และ LabVIEW Program สามารถสื่อสารได้แบบ TCP Protocol และ UDP Protocol เนื่องจากการตรวจสอบน็อต (Nut) เป็นงานที่ต้องใช้เวลาในการรับส่งข้อมูลที่ที่เร็วมาก เพื่อต้องการควบคุมหุ่นยนต์ให้ทำงานไปพร้อมกับระบบแมชชีนวิชัน (Machine

Vision) ให้มากที่สุด จึงเลือกใช้การสื่อสารแบบ UDP Protocol เพราะ UDP Protocol สามารถส่งข้อมูลได้เร็วกว่า TCP Protocol เนื่องจากไม่ต้องรอการสร้าง Connection ไม่ต้องรอการตรวจสอบข้อมูล นอกจากนี้ยังสามารถส่งข้อมูลให้มีปริมาณที่มากกว่าเพราะไม่มี Flow Control และ Congestion Control

เนื่องจาก TCP Protocol ไม่มีการตรวจสอบข้อมูลในการรับและส่งข้อมูล ได้มีการเขียนโปรแกรมสำหรับตรวจสอบความผิดพลาดของข้อมูล จึงทำให้ข้อมูลที่รับมาและส่งออกไปมีประสิทธิภาพที่สูงขึ้น

2.4.2 การจัดเก็บข้อมูลและการเรียกใช้ข้อมูล

ในกระบวนการทำงานจำเป็นต้องจัดเก็บข้อมูลตำแหน่งของน็อต (Nut) ในแต่ละโมเดล และจัดเก็บข้อมูลในการตรวจสอบ โดยจะถูกเก็บไว้ในไฟล์ .txt ในรูปแบบแถว (Row) และคอลัมน์ (Column) ซึ่งใน 1 แถวจะเป็นข้อมูลของแต่ละโมเดล ประกอบด้วยลักษณะของน็อต (Nut) ตำแหน่งในแต่ละตำแหน่งจะประกอบด้วย X Y Z ซึ่งจะจัดเก็บระยะห่างของน็อต (Nut) จนถึงจุดศูนย์กลางถังของเครื่องซักผ้า (Tub) เป็นหน่วยมิลลิเมตร

2.4.3 การสื่อสารระหว่างหุ่นยนต์กับ I/O ภายนอก

การเชื่อมต่อระหว่างหุ่นยนต์กับ I/O ภายนอก เป็นการเชื่อมต่อเพื่อส่งสัญญาณ และรับสัญญาณจากภายนอกเพื่อที่จะนำมาควบคุมในกระบวนการทำงาน ในการเชื่อมต่อนั้นสามารถเชื่อมต่อโดยตรงระหว่าง Board I/O ของหุ่นยนต์กับ I/O ภายนอกได้เลย

2.5 หลักการพื้นฐานเกี่ยวกับการวิเคราะห์รูปภาพ

2.5.1 หลักการในการเลือกกล้อง

ในการเลือกกล้องนั้น สามารถพิจารณาจากคุณลักษณะของกล้องต่องานที่ใช้ โดยแบบการพิจารณาออกเป็น 4 ด้าน ได้แก่ ความละเอียดของภาพ, Frame Rate, เลนส์, คุณสมบัติของภาพและชัตเตอร์ของกล้อง

1. ความละเอียดของภาพคือ จำนวน Pixel ที่ได้ต่อภาพ โดยจะแบ่งเป็น Pixel ตามแนวแกน X และแนวแกน Y ซึ่งจำนวน Pixel แกน X คูณกับจำนวน Pixel แกน Y มีค่าเท่ากับความละเอียดของภาพ โดยการพิจารณาความละเอียดนั้น จะพิจารณาจากวัตถุที่ต้องการถ่ายว่าต้องการภาพที่ชัดเจนมากแค่ไหน หรือมีลายละเอียดของวัตถุมากน้อยเพียงใด เพื่อให้จำนวน Pixel สามารถครอบคลุมได้ถึงรายละเอียดดังกล่าว ทั้งนี้การเลือกกล้องที่มีความละเอียดสูงอาจทำให้การประมวลผลของภาพช้าลง เนื่องจากมีจำนวน Pixel ที่มาก ทำให้ใช้เวลาประมวลผลช้ากว่ากล้องที่มีความละเอียดต่ำ

2. Frame Rate คือ จำนวนภาพที่กล้องสามารถถ่ายได้ต่อหนึ่งวินาที เช่น กล้องที่มี 14 Frame Rate แสดงว่ากล้องชนิดนี้สามารถถ่ายภาพได้ 14 ภาพใน 1 วินาที โดยการพิจารณานั้นจำเป็นต้องพิจารณาตัวเลขของ Frame Rate ต่อความไวของวัตถุที่ต้องการถ่ายภาพ หรือความเร็วของกล้องขณะเคลื่อนที่ในการถ่ายภาพวัตถุ โดยหากต้องการถ่ายวัตถุที่มีความเร็วสูง หรือกล้องเคลื่อนที่ด้วยความเร็วสูงจะต้องใช้กล้องที่มี Frame Rate สูงตามไปด้วย และในทางกลับกันหากต้องการถ่ายวัตถุที่เคลื่อนที่ช้า หรือหยุดนิ่งก็สามารถใช้กล้องที่มี Frame Rate ต่ำได้ ทั้งนี้ค่า

Frame Rate ยังสามารถคำนวนระยะเวลาที่กล้องจำเป็นต้องหยุด เพื่อถ่ายภาพได้อีกด้วย ถ้าหากทำให้กล้องหยุดด้วยระยะเวลาน้อยเกินไป อาจส่งผลให้ภาพที่ได้ไม่มีคุณภาพและไม่สามารถใช้งานได้

3. เลนส์คือ ชิ้นส่วนที่ประกอบด้วยกระจกเพื่อใช้ในการกระเจิงแสงให้ตกสู่แผ่นรับภาพ โดยเลนส์จะแบ่งออกเป็น 6 ชนิด ได้แก่

(1) เลนส์ปกติ (Normal Lens, Standard Lens) เป็นเลนส์ที่สามารถรับภาพได้ใกล้เคียงกับดวงตาของมนุษย์

(2) เลนส์มุมกว้าง (Wide Field lens, Wide Angle Lens, Short Lens) เป็นเลนส์ที่มีความยาวโฟกัสน้อยกว่าปกติ ใช้เพื่อถ่ายภาพในมุมกว้าง และภาพที่ได้อาจเกิดการโค้งเอนมากกว่าเลนส์ชนิดอื่น

(3) เลนส์มุมแคบ (Narrow Angle Lens) หรือเทเลโฟโต้ (Telephoto Lens) เลนส์ชนิดนี้มีความยาวโฟกัสมากกว่าปกติ โดยภาพที่ถ่ายได้จะมีขนาดขยายใหญ่กว่าวัตถุจริง

(4) เลนส์ตาปลา (Fisheye Lens) เป็นเลนส์ที่มีลักษณะคล้ายตาของปลาที่ว่ายอยู่ในน้ำ ทำมุมระยะถ่ายภาพได้มากถึง 180 องศา

(5) เลนส์ซูม (Zoom Lens) เป็นเลนส์ที่สามารถเปลี่ยนความยาวโฟกัสได้หลายค่าในตัวเดียวกัน คือ เป็นทั้งเลนส์มุมกว้าง เลนส์มาตรฐาน และเลนส์ถ่ายไกลในตัวเดียวกัน

(6) เลนส์มาโคร (Macro Lens) เป็นเลนส์ที่ใช้ถ่ายวัตถุขนาดเล็กมากๆ ขยายให้ใหญ่ขึ้นได้คล้ายกับเลนส์ถ่ายใกล้ แต่เลนส์มาโครนี้เป็นเลนส์ที่สามารถถ่ายภาพวัตถุที่อยู่ใกล้มาก ประมาณ 1-1.5 ฟุต

โดยในการคำนวนระยะโฟกัส จะคำนวนจากสมการที่ (2.1)

$$\frac{1}{F} = \frac{1}{U} + \frac{1}{V} \qquad (2.1)$$

โดยที่ F คือ ระยะโฟกัส
U คือ ระยะห่างของวัตถุ
V คือ ระยะห่างของแผ่นรับภาพ

ซึ่งส่วนใหญ่เลนส์ทั่วไปสามารถปรับระยะโฟกัสได้ ดังนั้นควรเลือกเลนส์ให้เหมาะกับระยะห่างของวัตถุที่จะทำการถ่าย นอกจากการปรับระยะโฟกัสแล้ว เลนส์ยังสามารถปรับรูรับแสงได้อีกด้วย โดยรูรับแสงจะส่งผลทำให้ภาพสว่างเมื่อเปิดกว้าง และภาพมืดลงเมื่อปรับให้รูรับแสงมีขนาดเล็กลง

4. คุณสมบัติของภาพจะพิจารณาจากวัตถุที่ต้องการถ่ายว่าต้องการภาพที่มีคุณลักษณะของสีหรือไม่ โดยกล้องนั้นส่วนใหญ่จะแบ่งออกเป็น 2 ประเภทคือ กล้องสี และกล้องขาว-ดำ

(1) กล้องสีคือ กล้องที่สามารถถ่ายภาพสีได้ โดยในแต่ละ Pixel ของภาพ จะประกอบด้วยข้อมูลจำนวน 32 Bits แบ่งเป็น ค่าแม่สี 8 Bits จำนวน 3 Channel (RGB, HSV, CYMK, etc.) และค่าเวกเตอร์ของ Pixel อีก 8 Bits โดยแต่ละช่วงจะประกอบไปด้วยข้อมูลในระยะช่วงที่มีค่า 0–255

(2) กล้องขาว-ดำคือ กล้องที่ไม่สามารถถ่ายภาพสีได้ โดยในแต่ละ Pixel ของภาพจะประกอบด้วยข้อมูลจำนวน 8 Bits ประกอบไปด้วยข้อมูลในระยะช่วงที่มีค่า 0–255 ซึ่งในการประมวลผลนั้น กล้องสีจะใช้ระยะเวลาที่นานกว่า เนื่องจากภาพที่ได้มีขนาดข้อมูลที่ใหญ่กว่า ทำให้ระยะเวลาในการประมวลผลต่อ Pixel นั้นกินเวลามากกว่าภาพที่ได้จากกล้องขาว-ดำ

5. ชัตเตอร์ของกล้องคือ ชนิดของแผ่นรับภาพของกล้อง โดยแบ่งเป็น 2 ชนิด ได้แก่ Rolling Shutter และ Global Shutter

(1) Rolling Shutter คือ แผ่นรับภาพที่มีกลไกการบันทึกภาพแบบทีละ Pixel จากซ้ายไปขวา บนลงล่าง โดยเป็นระบบ Shutter ที่พบได้ทั่วไป ใช้บันทึกภาพส่วนใหญ่ได้ แต่ไม่สามารถบันทึกภาพที่วัตถุที่ความเร็วสูงได้ ซึ่งอาจจะทำให้ภาพที่ได้ไม่มีคุณภาพ หรือเบลอได้

(2) Global Shutter คือ แผ่นรับภาพที่สามารถบันทึกภาพได้ทุก Pixel ในเวลาเดียวกัน ซึ่งเหมาะแก่การบันทึกภาพที่วัตถุเคลื่อนที่ด้วยความเร็วสูงมาก

บทที่ 3
วิธีการดำเนินโครงงาน

การดำเนินโครงงานหุ่นยนต์ตรวจสอบน็อตบนถังเครื่องซักผ้า (Tub Vision Inspection Robot) เริ่มดำเนินโครงงานตั้งแต่ การศึกษาพื้นที่การทำงาน การออกแบบโครงสร้างทางกล การออกแบบโปรแกรม การออกแบบวงจรไฟฟ้า การสั่งซื้อและการประกอบ หลังจากการประกอบและเขียนโปรแกรมจึงเริ่มทำการทดสอบก่อนจะนำไปทำการติดตั้งในสายการผลิต

3.1 การวางแผนการดำเนินโครงงาน

ในการจัดทำโครงงานจะต้องมีการวางแผนงานเป็นขั้นตอน และมีการจัดลำดับเวลาในงานแต่ละส่วนอย่างเหมาะสม เพื่อให้สามารถดำเนินได้อย่างเป็นระบบ โดยแผนงานที่วางไว้ เป็นช่วงเวลาที่ดำเนินโครงงานสหกิจศึกษา ณ บริษัท ไทยซัมซุง อิเล็กทรอนิกส์ จำกัด คือ ช่วงเวลาระหว่างวันที่ 5 มิถุนายน 2560 ถึงวันที่ 1 ธันวาคม 2560 แสดงแผนการดำเนินงานดังตารางที่ 3.1

ตารางที่ 3.1 Tub Vision Inspection Robot Timeline

แผนการดำเนินงาน	มิถุนายน				กรกฎาคม				สิงหาคม				กันยายน				ตุลาคม				พฤศจิกายน			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
สำรวจพื้นที่การทำงานและสอบถามความต้องการของทางโรงงาน																								
ออกแบบการทำงานของระบบทั้งหมด																								
ออกแบบระบบแมคคานิค																								
ออกแบบการติดตั้งระบบไฟฟ้า																								
ออกแบบโปรแกรม																								
สั่งซื้ออุปกรณ์																								
ประกอบชิ้นส่วนแมคคานิค																								
เดินระบบไฟฟ้า																								
เขียนโปรแกรม																								
ทดสอบระบบ																								
ติดตั้ง																								
ทดสอบระบบ ณ พื้นที่ติดตั้ง และปรับแก้ครั้งสุดท้าย																								

จากตารางที่ 3.1 จะเห็นว่าช่วงเวลาที่เริ่มการทำโครงงาน Tub Vision Inspection Robot นั้นเริ่มในช่วงเดือนสิงหาคม เนื่องจากในช่วงของเดือนมิถุนายนถึงเดือนกรกฎาคมมีการเปลี่ยนหัวข้อโครงงานอยู่หลายครั้ง จึงเป็นสาเหตุให้เริ่มต้นโครงงานนี้ในช่วงเดือนสิงหาคม

3.2 การศึกษาการทำงานในสายการผลิต และพื้นที่ในการติดตั้ง

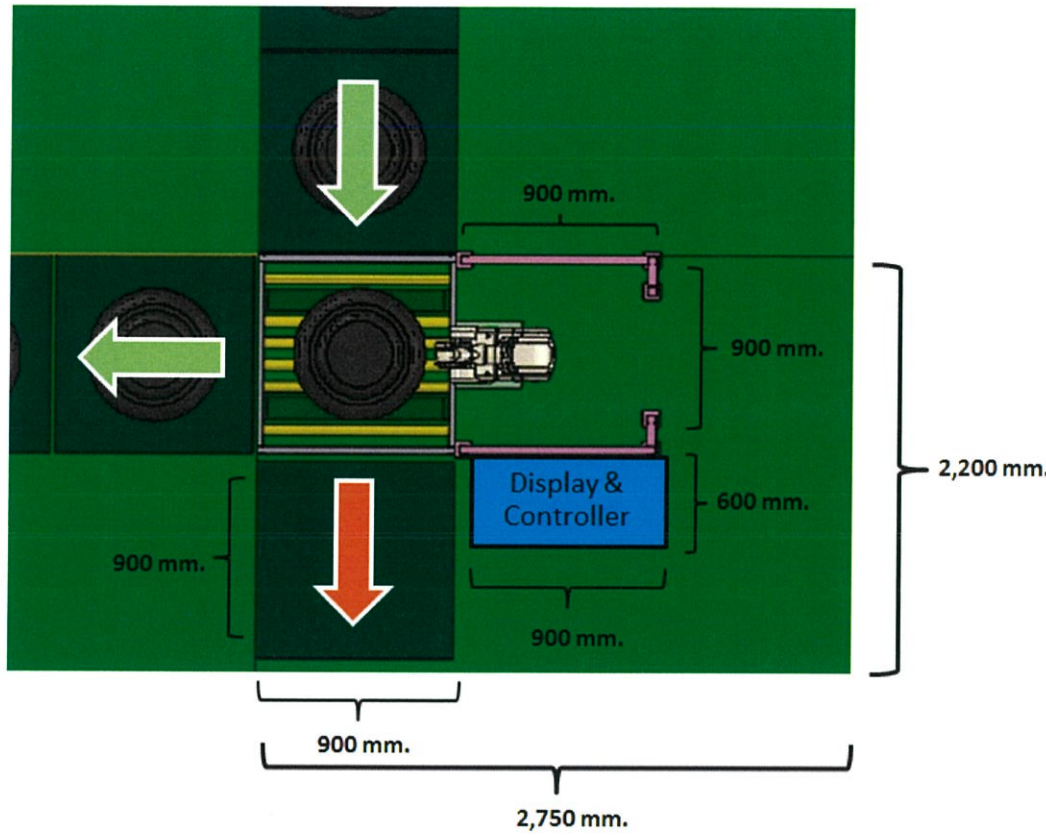
ลักษณะในการทำงานของจุดตรวจสอบน็อต เป็นการทำงานโดยพนักงานจะยืนประจำจุดอยู่ในกระบวนการผลิตก่อนหน้า จะเป็นการทำงานในส่วนของการประกอบยึดชิ้นส่วนบนตัวถังเครื่องซักผ้าโดยการขันน็อตยึด พนักงานงานที่ทำหน้าที่ในการตรวจสอบ จะต้องยืนตรวจน็อตที่ถูกขันยึดมาบนถังเครื่องซักผ้าทุกถัง ดังรูปที่ 3.1 ซึ่งจำนวนในการผลิตต่อวันแน่นอนว่ามีจำนวนที่มาก ทำให้พนักงานที่รับหน้าที่ในการตรวจสอบนั้นเกิดความล้าทางสายตาได้ ซึ่งอาจเป็นเหตุให้การตรวจสอบในบางครั้งอาจมีความผิดพลาดเกิดขึ้น และตัวพนักงานจะต้องโน้มตัวเข้าไปในสายการผลิตซึ่งอาจก่อให้เกิดอันตรายได้ โครงงานนี้จึงได้มีการออกแบบเครื่องจักรมาทำงานแทนพนักงานในขั้นตอนการตรวจสอบและสายการผลิตสามารถผลิตได้อย่างต่อเนื่อง

รูปที่ 3.1 การตรวจสอบน็อตโดยพนักงาน

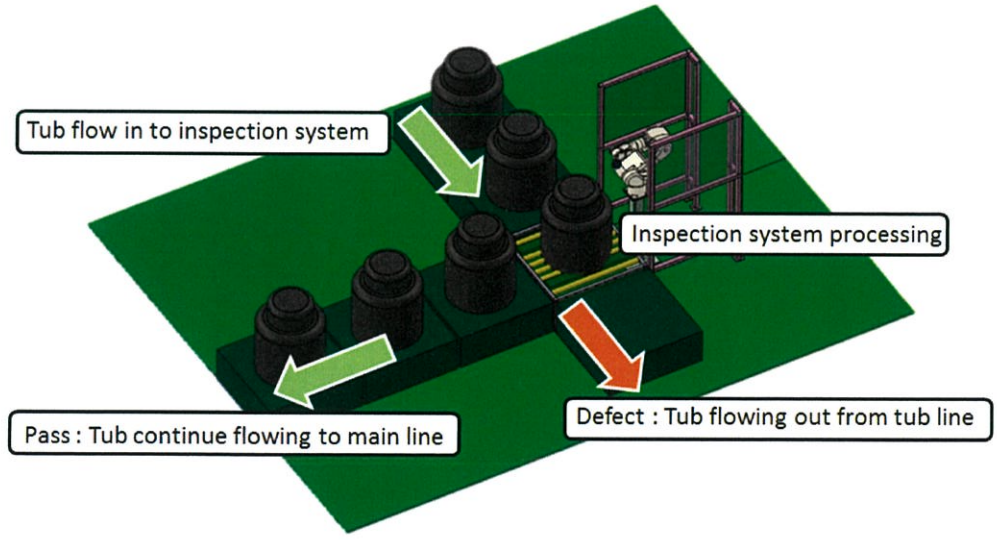
นอกจากการศึกษาการทำงานในสายการผลิตเพื่อทราบปัญหาที่พบแล้ว ยังต้องมีการศึกษาสภาพพื้นที่ และทำการวัดขนาดพื้นที่ที่ใช้ในการติดตั้งในรูปที่ 3.2 เพื่อใช้เป็นข้อมูลในการออกแบบโครงสร้างอื่นๆ ต่อไป

รูปที่ 3.2 พื้นที่ที่ใช้ในการติดตั้ง

เมื่อทำการศึกษาพื้นที่ที่ใช้ติดตั้งและทำการวัดขนาดต่างๆ เรียบร้อยแล้ว จึงทำการออกแบบพื้นที่ในการติดตั้ง และกระบวนการทำงานของระบบในลักษณะใดจึงจะเหมาะสม ซึ่งทำการออกแบบมาในลักษณะดังรูปที่ 3.3 และ รูปที่ 3.4



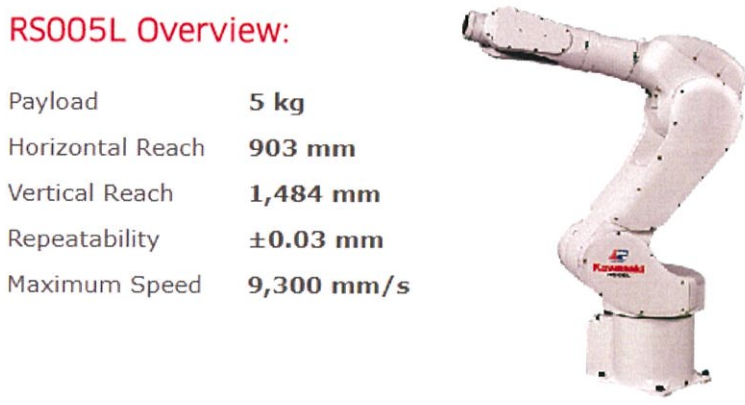
รูปที่ 3.3 ขนาดพื้นที่ที่ใช้ในการติดตั้ง



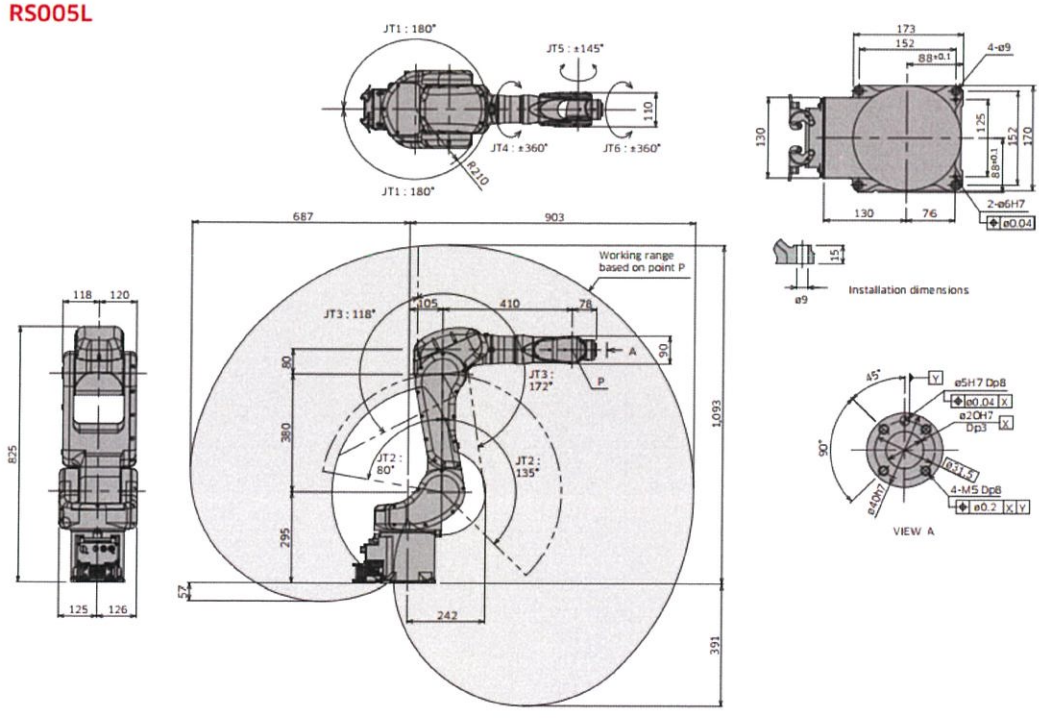
รูปที่ 3.4 รูปแบบการทำงานของระบบ

3.3 การออกแบบโครงสร้างและชิ้นส่วนทางกล

ในการออกแบบโครงสร้างและชิ้นส่วนทางกลของหุ่นยนต์ตรวจสอบน็อตบนถังเครื่องซักผ้า ต้องมีการอ้างอิงขนาดของหุ่นยนต์ที่ใช้ หุ่นยนต์ที่นำมาใช้เป็นหุ่นยนต์ของยี่ห้อ Kawasaki รุ่น RS005L โดยในการเลือกหุ่นยนต์ที่ใช้ต้องมีระยะเอื้อมของแขนหุ่นยนต์เพียงพอ และสามารถรับน้ำหนักของอุปกรณ์ที่นำมาติดตั้งได้ พื้นที่ที่หุ่นยนต์ต้องเอื้อมแขนเข้าไปนั้น เป็นคอนเวเยอร์แบบสายพาน (Belt Conveyor) ขนาดกว้าง 900 มิลลิเมตร × ยาว 900 มิลลิเมตร ซึ่งระยะของ Kawasaki Robot RS005L นั้นอยู่ที่ 903 มิลลิเมตร และสามารถรับน้ำหนักได้อยู่ที่ 5 กิโลกรัม ดังรูปที่ 3.5 และรูปที่ 3.6



รูปที่ 3.5 ข้อมูลพื้นฐานของ Kawasaki Robot RS005L



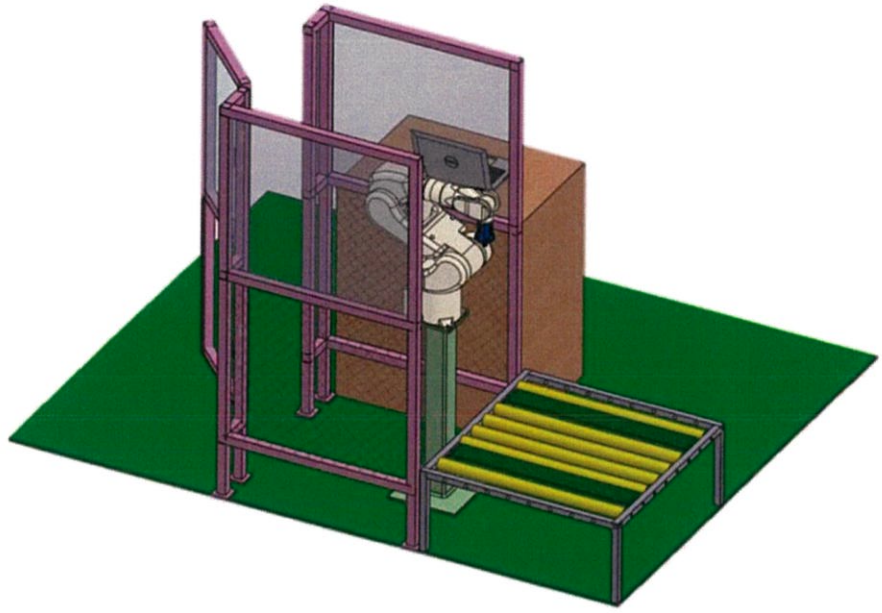
รูปที่ 3.6 ขนาดชิ้นส่วน และระยะการใช้งานของ Kawasaki Robot RS005L

หลังจากทำการเลือกหุ่นยนต์ที่ใช้แล้วก็จะนำขนาดชิ้นส่วนของหุ่นยนต์ ที่ใช้มาเป็นข้อมูลในการออกแบบโครงสร้างและชิ้นส่วนทางกลต่างๆ ดังรูปที่ 3.7 โดยมีชิ้นส่วนต่างๆ ดังนี้

3.3.1 ชิ้นส่วนในการติดกล้องและไฟส่องสว่าง

3.3.2 โครงสร้างฐานวางหุ่นยนต์

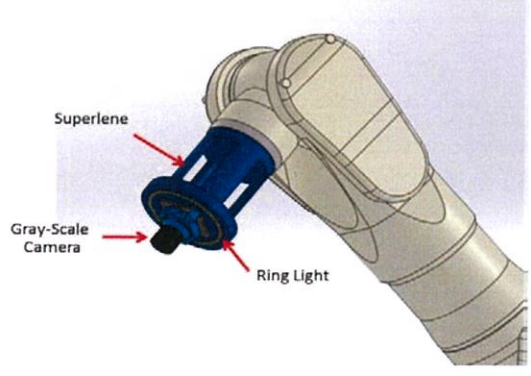
3.3.3 โครงสร้างรั้วกั้นความปลอดภัย



รูปที่ 3.7 รูปแบบหุ่นยนต์ตรวจสอบน็อตบนถังเครื่องซักผ้า

3.3.1 ชิ้นส่วนในการติดกล้องและไฟส่องสว่าง

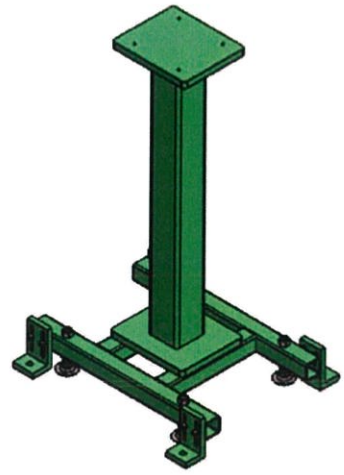
ชิ้นส่วนในการติดกล้องและไฟส่องสว่าง จะนำไปติดบริเวณปลายของข้อต่อที่ 6 ของหุ่นยนต์ โครงสร้างทำจากแผ่นอลูมิเนียม หนา 10 มิลลิเมตร มาขึ้นรูปเป็นชิ้นส่วนต่างๆ แล้วนำมาประกอบกัน โดยขนาดจะต้องดูจากขนาดของปลายข้อต่อที่ 6 ของหุ่นยนต์ จากรูปที่ 3.6 ซึ่งจะขนาดและตำแหน่งของรูสกรูที่ใช้ในการยึดชิ้นส่วน นอกจากจะต้องอ้างอิงขนาดจากหุ่นยนต์แล้ว ต้องอ้างอิงขนาดกล้องที่นำมาใช้ด้วย และข้อจำกัดในการออกแบบก็คือต้องให้กล้องอยู่ตรงส่วนกลางพอดี ดังรูปที่ 3.8



รูปที่ 3.8 ลักษณะของชิ้นส่วนในการติดกล้องและไฟส่องสว่าง

3.3.2 โครงสร้างฐานวางหุ่นยนต์

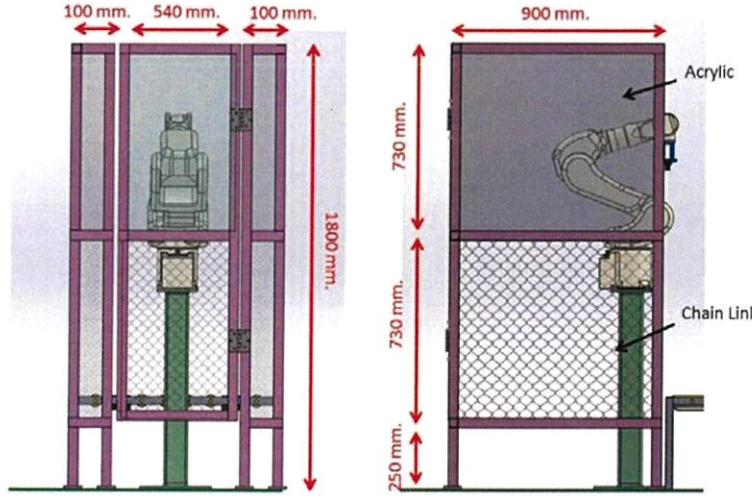
การออกแบบในส่วนของฐานวางหุ่นยนต์ ความสูงรวมของฐานคิดจาก ความสูงของคอนเวเยอร์ (Conveyor) รวมกับความสูงของถังเครื่องซักผ้าโมเดลใหญ่สุด เพื่อให้หุ่นยนต์สามารถทำงานได้กับทุกโมเดล และส่วนฐานล่างต้องมีความกว้างที่เหมาะสมเพื่อให้สามารถรับแรงฉุดจากการเคลื่อนที่ และหยุดอย่างรวดเร็วของหุ่นยนต์ได้ ซึ่งส่วนของฐานล่างจะมีการฝังหมุดยึดกับพื้นเพื่อความแข็งแรง ดังรูปที่ 3.9



รูปที่ 3.9 รูปแบบโครงสร้างฐานวางหุ่นยนต์

3.3.3 โครงสร้างรั้วป้องกันความปลอดภัย

รั้วป้องกันความปลอดภัยนี้ ทำเพื่อป้องกันไม่ให้เกิดอันตรายต่อพนักงาน ที่ทำงานอยู่บริเวณใกล้เคียงขณะที่หุ่นยนต์กำลังทำงานอยู่ ในกรณีที่เกิดเหตุการณ์ไม่คาดคิดขึ้น โดยทำการออกแบบให้ครอบคลุมพื้นที่การทำงานของหุ่นยนต์คือ ระยะเท่ากับความกว้างคอนเวเยอร์ (Conveyor) 900 มิลลิเมตร ความสูงตามข้อจำกัดของโรงงานต้องสูงไม่เกิน 1800 มิลลิเมตร ดังรูปที่ 3.10



รูปที่ 3.10 รูปแบบโครงสร้างรั้วป้องกันความปลอดภัย

3.4 การออกแบบโปรแกรม

ในส่วนของการออกแบบจะแบ่งแยกออกเป็น 3 ส่วนหลักๆ ดังนี้

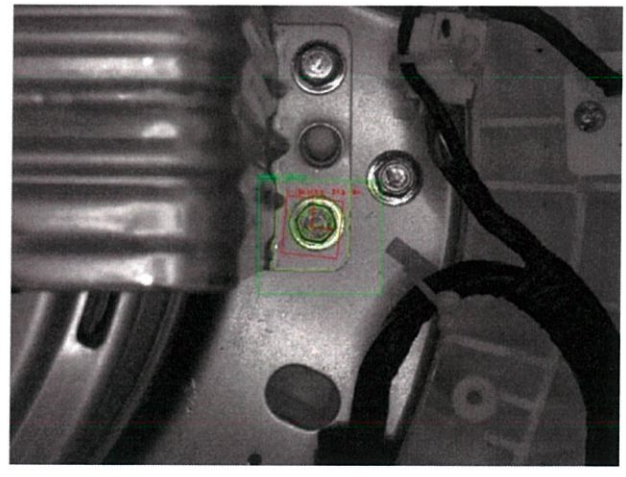
3.4.1 Vision Assistant

โปรแกรมในส่วนนี้จะใช้รูปภาพที่ผู้ใช้โปรแกรมบันทึกค่านำมาเป็นรูปภาพต้นแบบเพื่อหาหัวสกรู (Screw) ชนิดต่างๆ โดยถึงแม้ว่ากล้องที่มีจะเป็นกล้องสี แต่จากข้อกำหนดของโรงงานนั้นทำให้มีเวลาจำกัด จึงจำเป็นต้องใช้ภาพ Grayscale เนื่องจากแต่ละ Pixel มีขนาดเพียงแค่ 8 Bits ซึ่งแตกต่างจากภาพสี ซึ่งแต่ละ Pixel จะประกอบไปด้วย Red, Green, Blue และ Alpha ซึ่งมีขนาด 32 Bits/Pixel และได้ทำการ Resampling ภาพที่ได้จากขนาด 5 ล้านพิกเซล ให้เหลือเพียง 1 ล้านพิกเซล เพื่อลดระยะเวลาในการประมวลผล โดยมีวิธีวิเคราะห์สกรู (Screw) ชนิดต่างๆ ดังนี้

3.4.1.1 Hex Washer ดังรูปที่ 3.11

รูปที่ 3.11 Hex Washer

โดยหัวสกรู (Screw) ชนิดนี้หลังจากที่ได้ทำการบันทึกค่ารูปภาพเป็นที่เรียบร้อย ได้ทำการสร้าง Template ผ่าน Epsilon Filter หลังจากนั้นจึงได้ทำการปรับค่า Threshold เพื่อให้เห็นขอบเขตของเส้นตัดภาพที่ชัดเจน หลังจากนั้นจึงได้ทำการบันทึก Geometric Matching กล่าวคือการบันทึกรูปแบบลักษณะของเส้น แล้วจึงนำไปตรวจสอบกับรูปภาพ จะเช็คว่ามีเส้นรอบรูปใกล้เคียงกันมากน้อยเพียงใด โดยค่า Threshold ที่ใช้ คือ 45 จาก 0–255 ในโหมด Grayscale และปรับค่าความแม่นยำ (Accuracy) ไว้ตามความเหมาะสมของพื้นที่ที่จะตรวจสอบ ดังรูปที่ 3.12

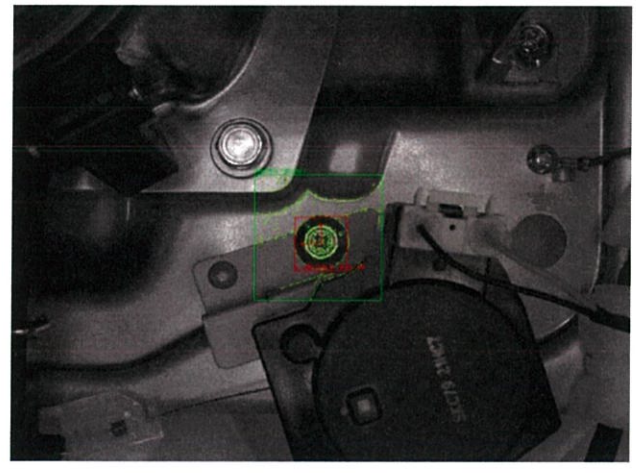


รูปที่ 3.12 Hex Washer–Geometric Matching

3.4.1.2 Shotted Hex Washer ดังรูปที่ 3.13

รูปที่ 3.13 Shotted Hex Washer

โดยหัวสกรู (Screw) ชนิดนี้มีความคล้ายกับ Hex Washer แต่จะมีความแตกต่างกันของรอยบากด้านใน ซึ่งสามารถใช้อัลกอริทึม (Algorithm) เดียวกับ Hex Washer ได้เพียงแต่ว่าต้องปรับ Threshold ให้สูงขึ้น เพื่อลดความผิดพลาดในการประมวลผล โดยค่า Threshold ที่ใช้คือ 75 และใช้ Geometric Matching โดยไม่ผ่าน Filter ใดๆ ดังรูปที่ 3.14

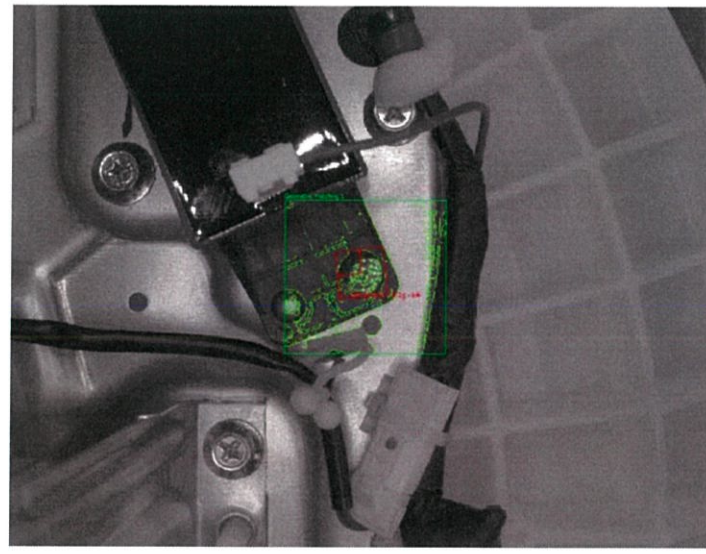


รูปที่ 3.14 Shotted Hex Washer–Geometric Matching

3.4.1.3 Truss Head ดังรูปที่ 3.15

รูปที่ 3.15 Truss Head

โดยการจำแนกหัวสกรูนั้นได้ใช้ Exponential Filter เพื่อทำให้รูปภาพสว่างขึ้น เนื่องจากหัวสกรู (Screw) มีขนาดเล็กกว่า Bolt ทั้งสองชนิด ทำให้มีระยะห่างจาก Light Ring ที่มากกว่า อาจจะทำให้แสงส่องไม่ถึงดีนัก หลังจากนั้นได้ทำการบันทึก Template ของสกรู ผ่าน Pattern Matching ที่ทำการวิเคราะห์ระดับสีที่ใกล้เคียงกันของ Template บนรูปภาพที่สนใจ และทำการตั้ง Rotation Matching ไว้ที่ 90 อาศา เนื่องจากรอยบากบนหัวสกรูนั้นมีค่าห่างกัน 90 องศาทั้ง 4 ด้าน ดังรูปที่ 3.16

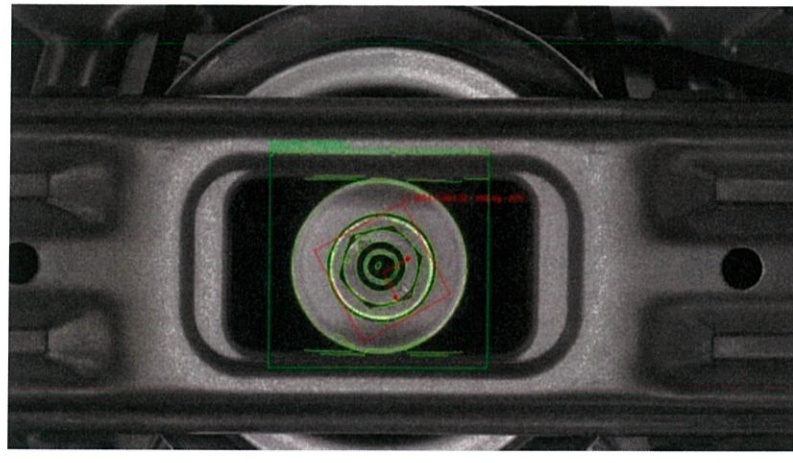


รูปที่ 3.16 Truss Head–Pattern Matching

3.4.1.4 Hex Nut ดังรูปที่ 3.17

รูปที่ 3.17 Hex Nut

เนื่องจาก Hex Nut นั้นมีเพียงจุดเดียว และตัวหัว Hex Nut นั้น มีความแตกต่างอย่างมาก กับหัวน็อตชนิดอื่นๆ จึงสามารถใช้ Geometric Matching ในการหาได้เลย โดยทำการตั้ง Rotation Matching ไว้ที่ 60 อาศา เนื่องจาก Bolt มี 6 ด้าน ด้านละ 60 องศา และปรับค่า Threshold ไว้ที่ 90 เนื่องจากพื้นที่จุดสนใจอยู่ใกล้ Light Ring ทำให้ภาพที่ได้นั้นสว่าง ดังรูปที่ 3.18



รูปที่ 3.18 Hex Nut–Geometric Matching

3.4.2 LabVIEW Program

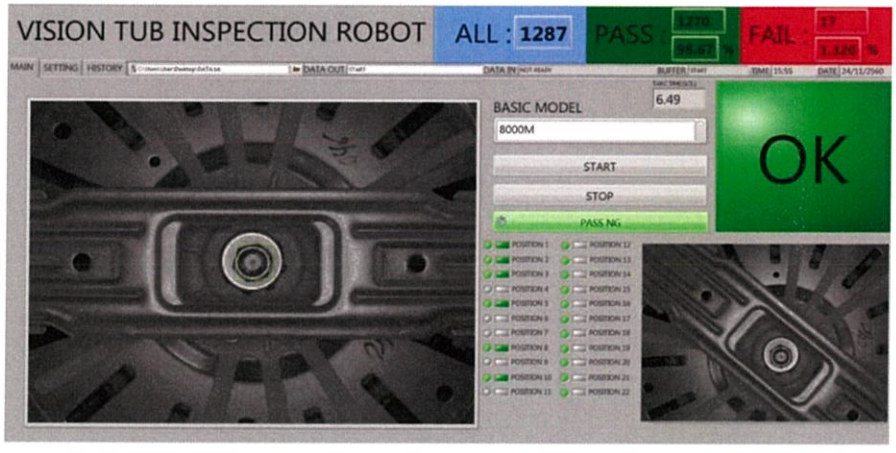
ส่วนนี้จะทำหน้าที่เป็นส่วนประมวลผลหลักของระบบทั้งหมด ซึ่งมีหน้าที่ต่างๆ ดังนี้

1. เป็นส่วนที่เอาไว้แสดงผลให้ผู้ใช้ได้ทราบสถานะต่างๆ ดังรูปที่ 3.19 มีหน้าที่ต่างๆ ได้แก่
   - แสดงภาพที่มาจากกล้องแบบ Real Time
   - แสดงผลว่าผ่านมาตรฐาน (OK) และไม่ผ่านมาตรฐาน (NG) ในแต่ละตำแหน่ง
   - แสดงสถานะการทำงานที่ผ่านมาทั้งหมด รวมถึงจำนวนทั้งหมดที่ผ่านมาตรฐาน (OK) และไม่ผ่านมาตรฐาน (NG)
   - แสดงจุดที่เจอน็อต (Nut) ของแต่ละชิ้นงาน

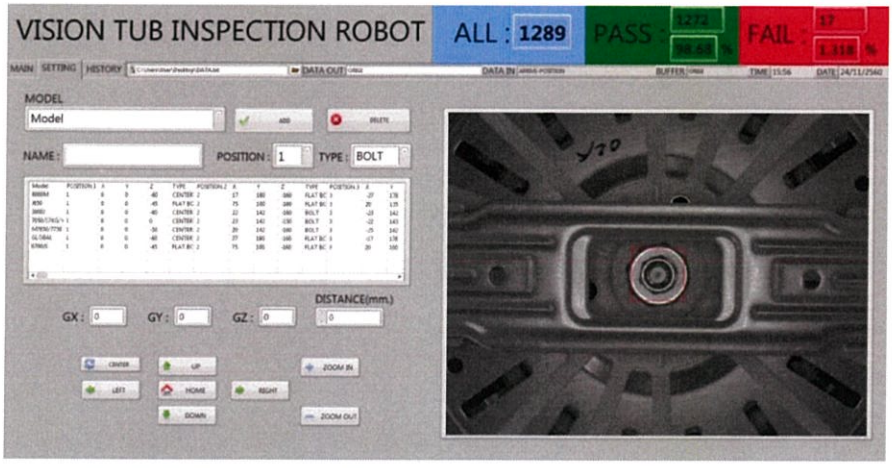
2. ส่วนบันทึกข้อมูลจะทำการจัดเก็บข้อมูลของแต่ละตำแหน่งในแต่ละโมเดล และจับเก็บสถานะของชิ้นงานในแต่ละช่วงเวลานั้นๆ โดยจะทำการเก็บไว้ใน ดังรูปที่ 3.20 และรูปที่ 3.21

3. เป็นส่วนรับข้อมูลและส่งข้อมูลจากหุ่นยนต์ โดยข้อมูลนั้นจะเป็นแบบ String

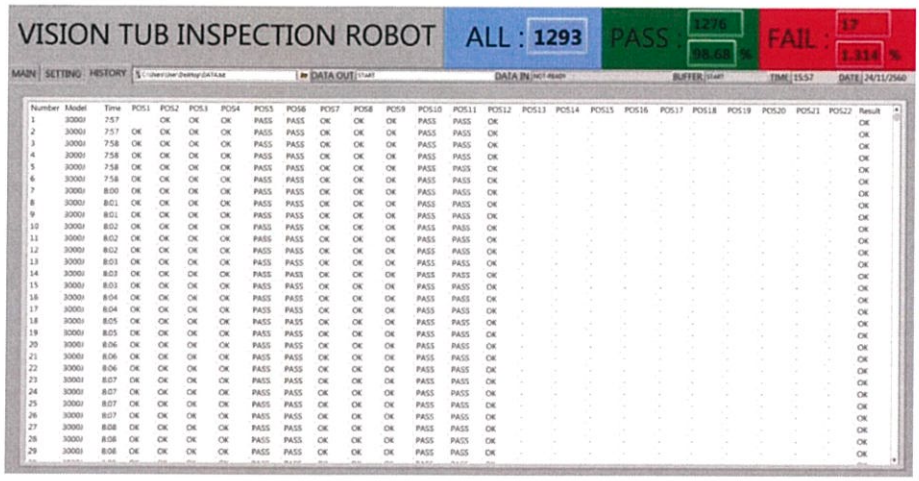
4. ส่วนตรวจสอบข้อมูล เนื่องจากการใช้ UDP Protocol ทำให้ข้อมูลที่รับและส่งข้อมูลไม่ได้ถูกตรวจสอบ จึงได้เขียนโปรแกรมไว้ตรวจสอบการรับส่งข้อมูลไว้ใน LabVIEW Program โดยเขียนว่าในทุกๆ การทำงานถ้าส่งข้อมูลออกไปให้หุ่นยนต์ ตัวหุ่นยนต์จะส่งข้อมูลกลับมาหา LabVIEW Program ถ้าไม่ส่งมาก็จะส่งข้อมูลซ้ำแต่ถ้าส่งมาแล้วก็จะทำในส่วนถัดไป



รูปที่ 3.19 หน้าต่าง Main Program



รูปที่ 3.20 หน้าต่าง Setting



รูปที่ 3.21 หน้าต่าง History

3.4.3 Program Robot

ในการเขียนโปรแกรมควบคุมหุ่นยนต์ Kawasaki นั้น จะใช้ AS Language ซึ่งมีพื้นฐานใกล้เคียงกับภาษา C ผ่าน IDE ของ Notepad++ นำมาบันทึกเป็นไฟล์ .pg และทำการ Compile/Upload ผ่าน KR Term ที่ทำการเชื่อมต่อกับ Robot Controller อยู่ ผ่านสาย Ethernet ซึ่งในโปรแกรมได้เขียนโปรแกรมไว้เป็น 4 ส่วนหลักๆ ดังนี้

3.4.3.1 Auto-Start Program

เป็นส่วนที่ไว้สำหรับตรวจสอบสถานะต่างๆ ของโปรแกรมไม่ว่าจะเป็นสถานะการเปิดปิดของสวิตช์ สถานะของเซนเซอร์ หรือหน้าที่ในการตรวจสอบ และรับ-ส่ง Protocol กันระหว่างหุ่นยนต์และโปรแกรม LabVIEW โดยโปรแกรมส่วนนี้เมื่อเช็คสถานะได้แล้วจะทำการเปิด-ปิดสถานะของ Internal Relay ใน Robot Controller และวนเป็นลูปต่อๆ ไป โดยส่วนใหญ่มักไม่เขียนให้โปรแกรมส่วนนี้ติดลูป หรือหยุดรอสัญญาณอื่นๆ เนื่องจากอาจจะทำให้เกิดความล่าช้าในการเช็คสถานะและอาจก่อให้เกิดปัญหากับโปรแกรมส่วนอื่นๆ ได้

3.4.3.2 Main Program

หลังจากส่วน Auto-Start ได้ทำการเปิด-ปิดสถานะ Internal Relay แล้ว โปรแกรม Main ส่วนนี้จะทำการเช็คสถานะดังกล่าวแล้วทำการเริ่มทำงานเป็นขั้นตอนที่เขียนไว้ โดยส่วนใหญ่จะนำขั้นตอนการเคลื่อนที่ของหุ่นยนต์มาใส่ในส่วนนี้ เพื่อการตรวจเช็คที่ง่ายและสามารถสั่งให้รอได้โดยไม่ส่งผลกับสถานะของโปรแกรมอื่นๆ

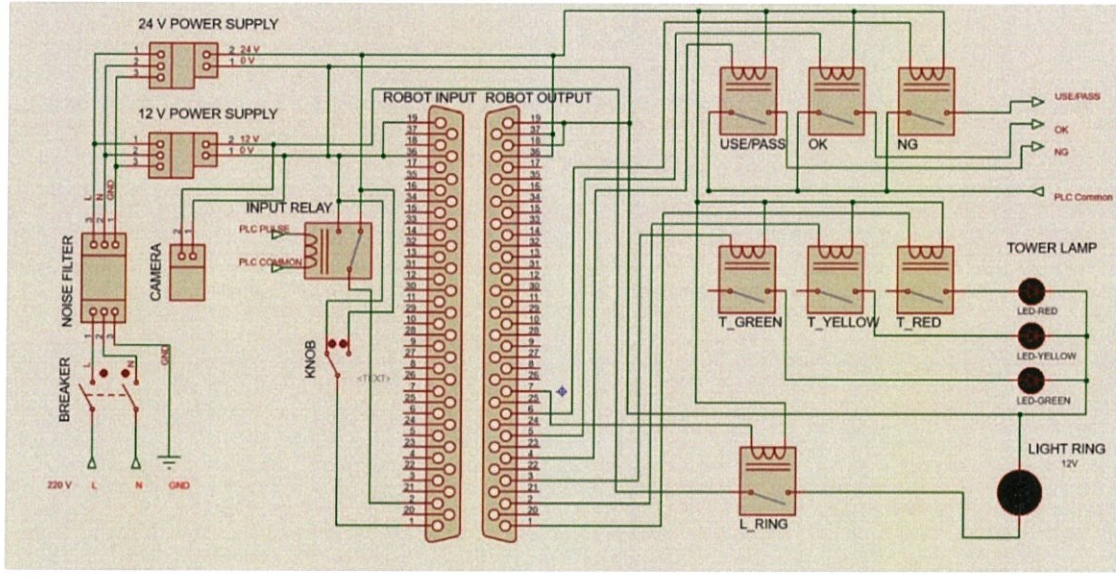
3.4.3.3 Send & Receive Program

โปรแกรมส่วนนี้คือ การประกาศ UDP Communication เพื่อทำการเชื่อมต่อระหว่างComputer และหุ่นยนต์ โดยจะเป็นโปรแกรมแยกส่วนออกจากโปรแกรมส่วน Auto-Start และ Main Program แต่จะคอยการเรียกใช้งานจากโปรแกรมส่วนอื่นๆ เพื่อรับและส่งข้อมูลในรูปแบบของ String ระหว่างกัน โดยในบางกรณีสามารถเขียนลำดับในการรับ-ส่งข้อมูลไว้เพื่อเช็ค State ในกรณีที่เกิด Error ได้อีกด้วย

3.5 การออกแบบระบบไฟฟ้า

ในการออกแบบวงจรไฟฟ้าโครงงานนี้ใช้หลักการความปลอดภัย และความเรียบง่ายของระบบไฟฟ้า ซึ่งอุปกรณ์ส่วนใหญ่ที่ใช้เป็นของที่ทางโรงงานมีอะไหล่สำรอง (Spare Part) และใช้งานเป็นประจำ

เมื่อทำการออกแบบการทำงานของระบบเสร็จสิ้นทั้งหมด จึงได้กำหนดอุปกรณ์ในที่ต้องใช้ในการทำงานของระบบ วงจรไฟฟ้านี้มีไว้สำหรับเป็นวงจรควบคุมในการจ่ายไฟฟ้าไปยังอุปกรณ์ต่างๆ และเป็นตัวกลางในการรับ Input จากอุปกรณ์ภายนอก ผ่าน Relay และส่งสัญญาณเข้าสู่ Input ของหุ่นยนต์ นอกจากนั้นยังเป็นตัวรับสัญญาณ Output จากหุ่นยนต์ ผ่าน Relay เพื่อไปสั่งงานอุปกรณ์อื่นๆ ต่อไป ดังรูปที่ 3.22



รูปที่ 3.22 แผนผังวงจรไฟฟ้า

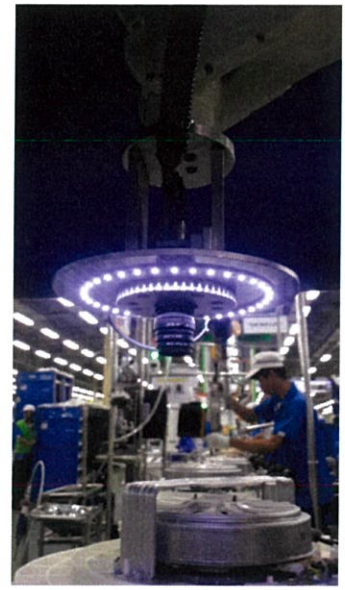
หลังจากทำการออกแบบวงจรไฟฟ้า และกำหนดอุปกรณ์ที่ใช้เสร็จสิ้น ลำดับต่อไปจะเป็นการจัดวางตำแหน่งอุปกรณ์ในตู้ไฟ เพื่อเตรียมการนำตู้ไฟไปติดตั้งในสายการผลิต โดยมีการเดินสายไฟต่างๆ ในรางสายไฟ (Wire Duct) และเข้าหางปลาในการต่อจุดเชื่อม (Terminal) ทุกจุด

บทที่ 4
ผลการดำเนินโครงงาน

4.1 โครงสร้างและระบบกลไก

การประกอบโครงสร้างต่างๆ ประกอบจนเสร็จสมบูรณ์ 100 เปอร์เซ็นต์ ทำการติดตั้งและทดสอบการใช้งานจริง ดังรูปที่ 4.1 และรูปที่ 4.2 โดยการทดสอบในเรื่องของการทำงานตามขั้นตอนที่ได้ออกแบบไว้ ความแม่นยำของระบบ และการตรวจสอบข้อผิดพลาดต่างๆ ของระบบ

รูปที่ 4.1 โครงสร้างทั้งหมดเมื่อนำมาติดตั้งในสายการผลิต



รูปที่ 4.2 ชิ้นส่วนในการติดกล้องและไฟส่องสว่างที่เสร็จสมบูรณ์

4.2 ตู้ไฟที่ใช้ในการควบคุมการจ่ายไฟฟ้า

ในการประกอบทางไฟฟ้าเสร็จสมบูรณ์ 100 เปอร์เซ็นต์ นำไปติดตั้ง เชื่อมต่อกับเซนเซอร์และอุปกรณ์ต่างๆ จากนั้นทำการทดสอบการใช้งาน ซึ่งผลที่ได้สามารถทำงานได้ตามเป้าหมาย ดังรูปที่ 4.3 และรูปที่ 4.4

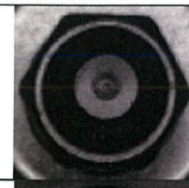
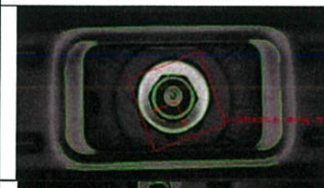
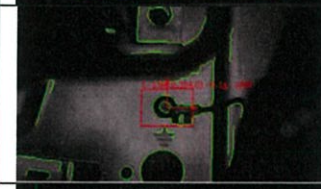
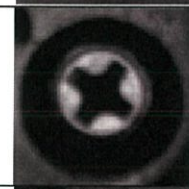
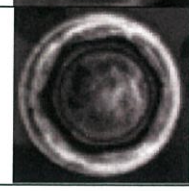
รูปที่ 4.3 การเดินสายไฟทั้งหมดในตู้ไฟ

รูปที่ 4.4 ติดตั้งตู้ไฟและทดสอบการทำงาน

4.3 ผลการทดสอบการใช้งานจริง

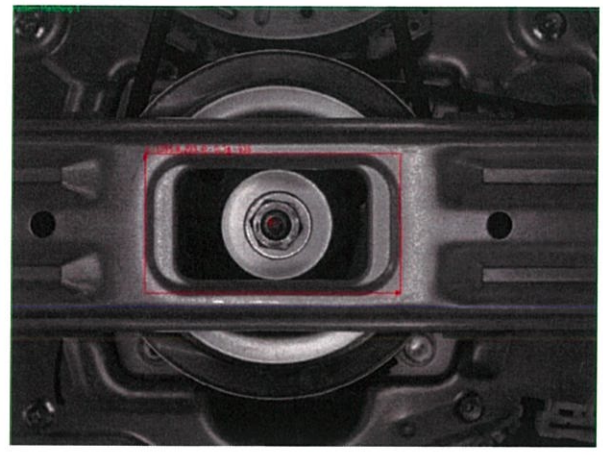
4.3.1 ทดสอบการตรวจจับหัวน็อตชนิดต่างๆ

ตารางที่ 4.1 ทดสอบการตรวจจับหัวน็อต

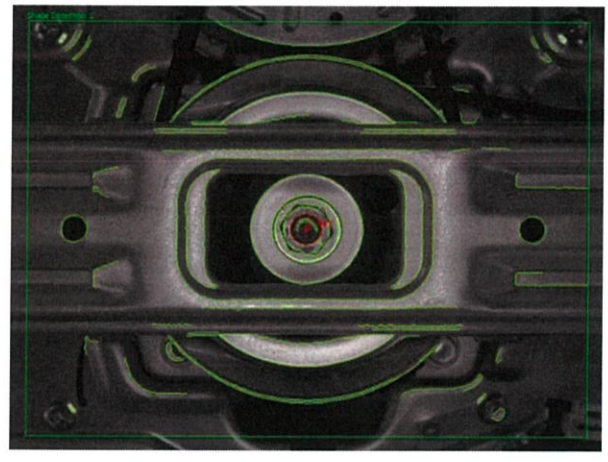
ประเภทของน็อต		ภาพที่ได้จากกล้อง	ภาพที่ตรวจจับได้
HEX NUT			
TRUSS HEAD			
SHOTTED HEX WASHER			
HEX WASHER			

4.3.2 ทดสอบการหามุมบิดของถังเครื่องซักผ้า หาจุดศูนย์กลาง และองศาการหมุนของกล้อง

หลังจากการหาจุดศูนย์กลางและองศาของ Tub แล้วจะนำค่าที่ได้มาทำการสั่งเคลื่อนที่หุ่นยนต์ โดยค่าองศาที่ได้ จะนำมา Mod ด้วย 90 และนำค่าที่เหลือไปสั่งบังคับ Joint 6 ของหุ่นยนต์ผ่านคำสั่ง Drive (“Number of Joint”), (“Angle”) ส่วนค่า X และ Y ที่ได้จะนำมาคูณกับ Scale อัตราส่วนของระยะทางในหน่วยมิลลิเมตรต่อพิกเซล แล้วจึงสั่งให้หุ่นยนต์เคลื่อนที่ด้วยคำสั่ง TDRAW (“X”), (“Y”), (“Z”) โดยต้องทำการบิดองศาของ Joint 6 ก่อนเนื่องจากการบิด Joint 6 จะเป็นการบิดองศาของ Tool ด้วย แล้วจึงส่งคำสั่ง TDRAW ออกไปเพื่อเป็นการบอกให้หุ่นเคลื่อนที่ Shift จากตำแหน่ง X, Y, Z, O, A, T ของ Tool ไม่เช่นนั้นจะเป็นการสั่งให้หุ่นยนต์ Shift จาก Base Coordinate แทนซึ่งจะทำให้หุ่นยนต์เคลื่อนที่ไปผิดตำแหน่ง ซึ่งในการหามุม และจุดศูนย์กลาง มีลักษณะดังรูปที่ 4.5 และรูปที่ 4.6



รูปที่ 4.5 ทดสอบหาการบิดของถังเครื่องซักผ้า



รูปที่ 4.6 ทดสอบหาจุดศูนย์กลางของถังเครื่องซักผ้า

4.3.3 ทดสอบระยะเวลาในการเคลื่อนที่หุ่นยนต์ไปยังตำแหน่งต่างๆ

ในการทำงานจริงให้ผู้ใช้งานบันทึกค่าของหัวน็อตที่ถัดไปจากจุดศูนย์กลางของถังเครื่องซักผ้า ดังนั้นจึงมีระยะห่างระหว่างจุดศูนย์กลางถังซักผ้า และหัวน็อตตำแหน่งต่างๆ แต่ในการใช้งานจริงต้องการให้หุ่นยนต์ทำการเคลื่อนที่จากตำแหน่งที่ 1 สู่ตำแหน่งที่ 2 โดยไม่ต้องกลับสู่จุดศูนย์กลาง จึงได้ทำการใช้ค่าข้อมูลของระยะห่างระหว่าง จุดศูนย์กลางถังซักผ้า และหัวน็อตตำแหน่ง 1 และจุดศูนย์กลางถังซักผ้า และหัวน็อตตำแหน่ง 2 เพื่อให้ได้ระยะห่างระหว่าง หัวน็อตตำแหน่ง 1 และหัวน็อตตำแหน่ง 2 โดยสามารถคำนวณได้จากการที่เอาค่า X, Y, Z ของจุดที่อยู่ ลบด้วย ค่า X, Y, Z ของจุดที่จะไป

ตารางที่ 4.2 บันทึกเวลาในการเคลื่อนที่ของหุ่นยนต์

LOAD IN: 4 Second	**Robot Speed : 100%**
LOAD OUT: 2 Second	

POINT	Processing Time (SEC)	Tack Time (SEC)
22	15.67	21.67
21	14.9	20.9
20	14.34	20.34
19	13.65	19.65
18	12.99	18.99
17	12.77	18.77
16	11.67	17.67
15	10.94	16.94
14	10.31	16.31
13	9.93	15.93
12	9.01	15.01
11	8.69	14.69
10	7.95	13.95
9	7.32	13.32
8	6.66	12.66
7	5.85	11.85
6	5.08	11.08
5	4.82	10.82
4	4.65	10.65
3	4.36	10.36
2	3.88	9.88
1	2.77	8.77

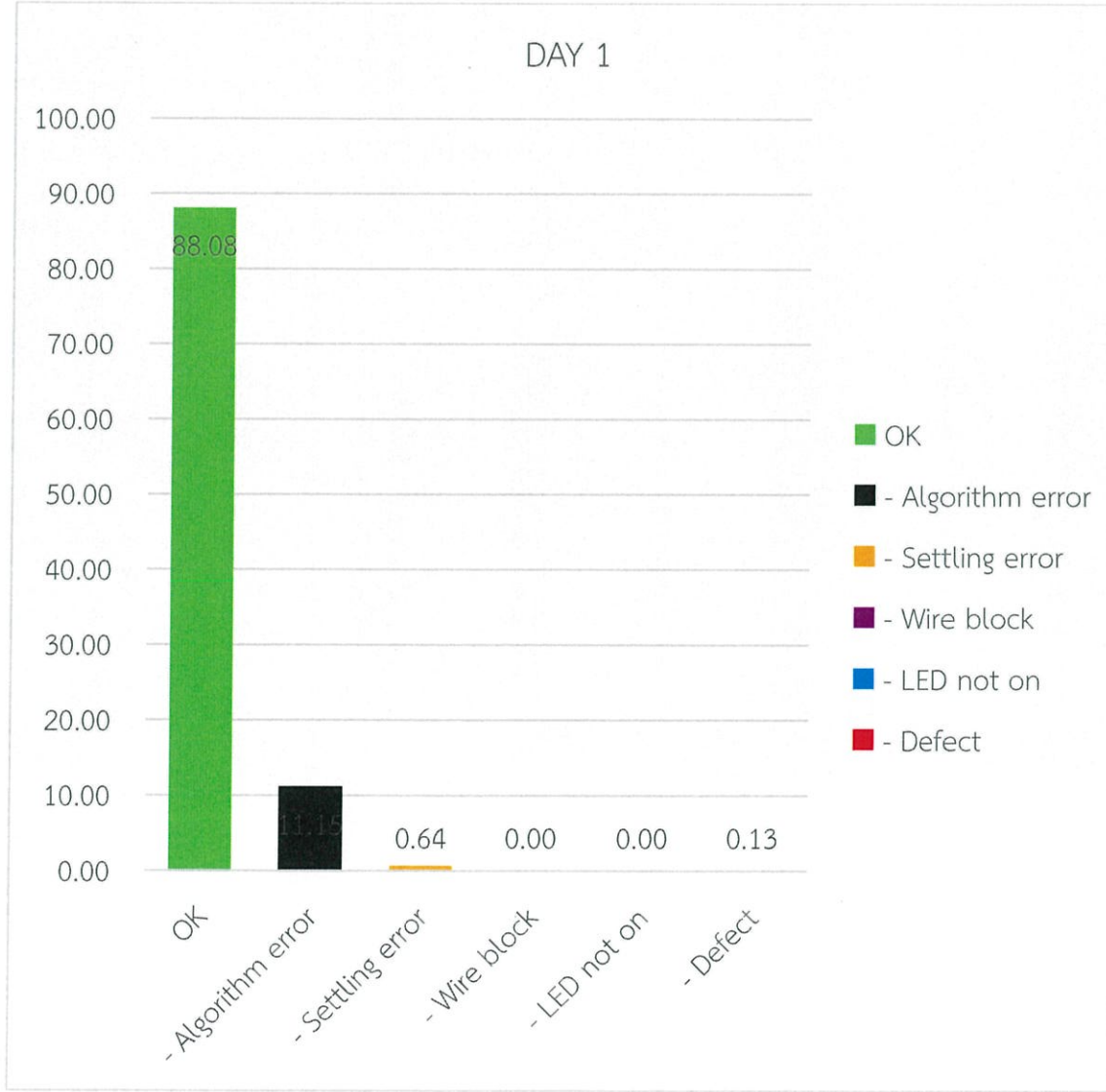
4.3.4 ทดสอบความแม่นยำของระบบ

ในการทดสอบความแม่นยำในการตรวจจับหัวน็อต หลังการจากทำการติดตั้งได้พบว่าระบบสามารถตรวจจับได้ แต่ยังพบปัญหาซึ่งเกิดจากหลายสาเหตุ ได้แก่ Algorithm Error, Settling Error, Wire Block, Light Source Error, Light Noise และ Defect

ติดตามผลการทำงานของระบบ เป็นระยะเวลา 9 วัน ได้ผลดังนี้

ตารางที่ 4.3 ผลการทดสอบวันที่ 1

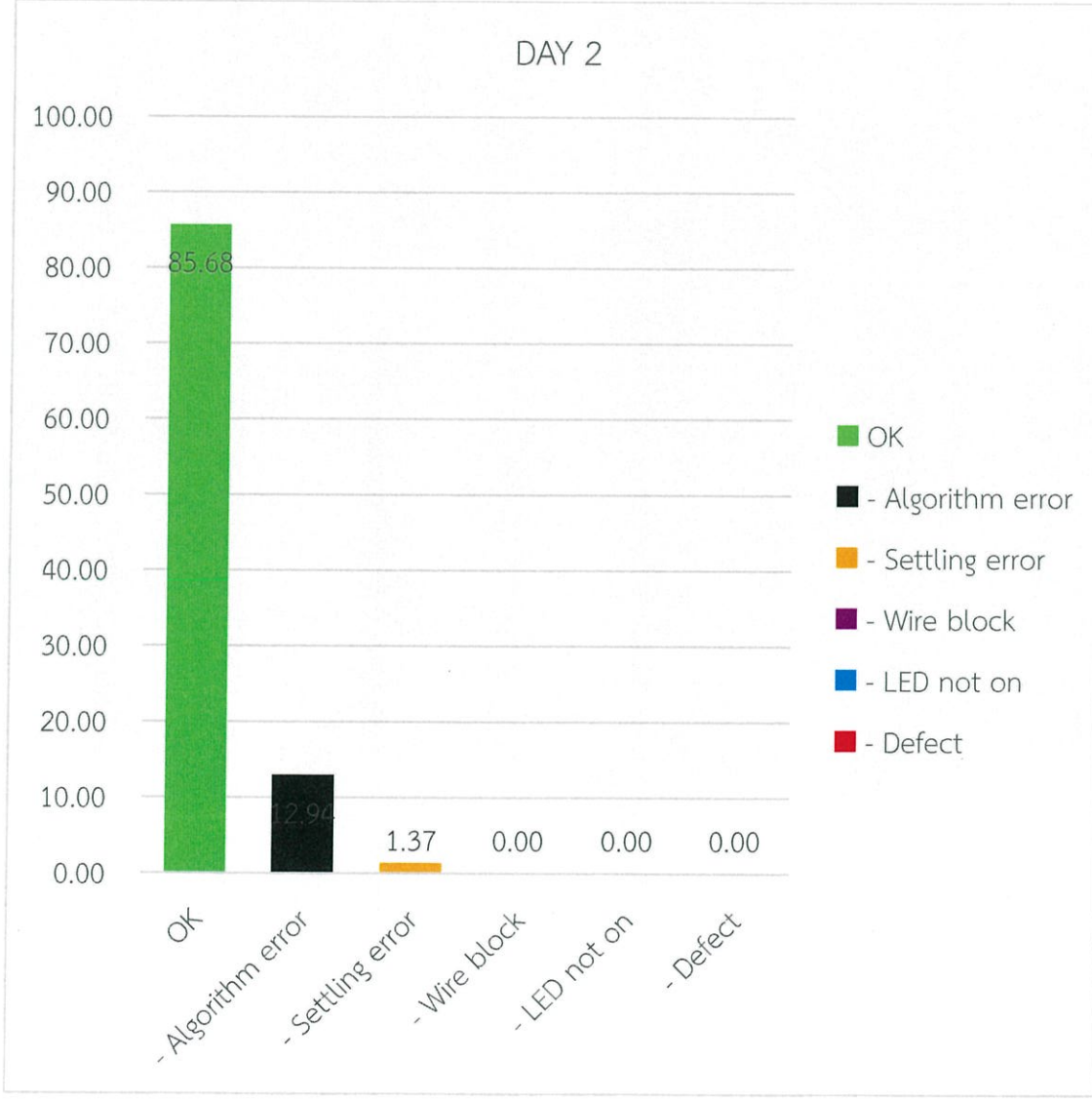
All	780	100.00	%
OK	687	88.08	%
NG	93	11.92	%
- Algorithm error	87	11.15	%
- Settling error	5	0.64	%
- Wire block	0	0.00	%
- LED not on	0	0.00	%
- Defect	1	0.13	%



รูปที่ 4.7 กราฟเปรียบเทียบปริมาณ Error วันที่ 1

ตารางที่ 4.4 ผลการทดสอบวันที่ 2

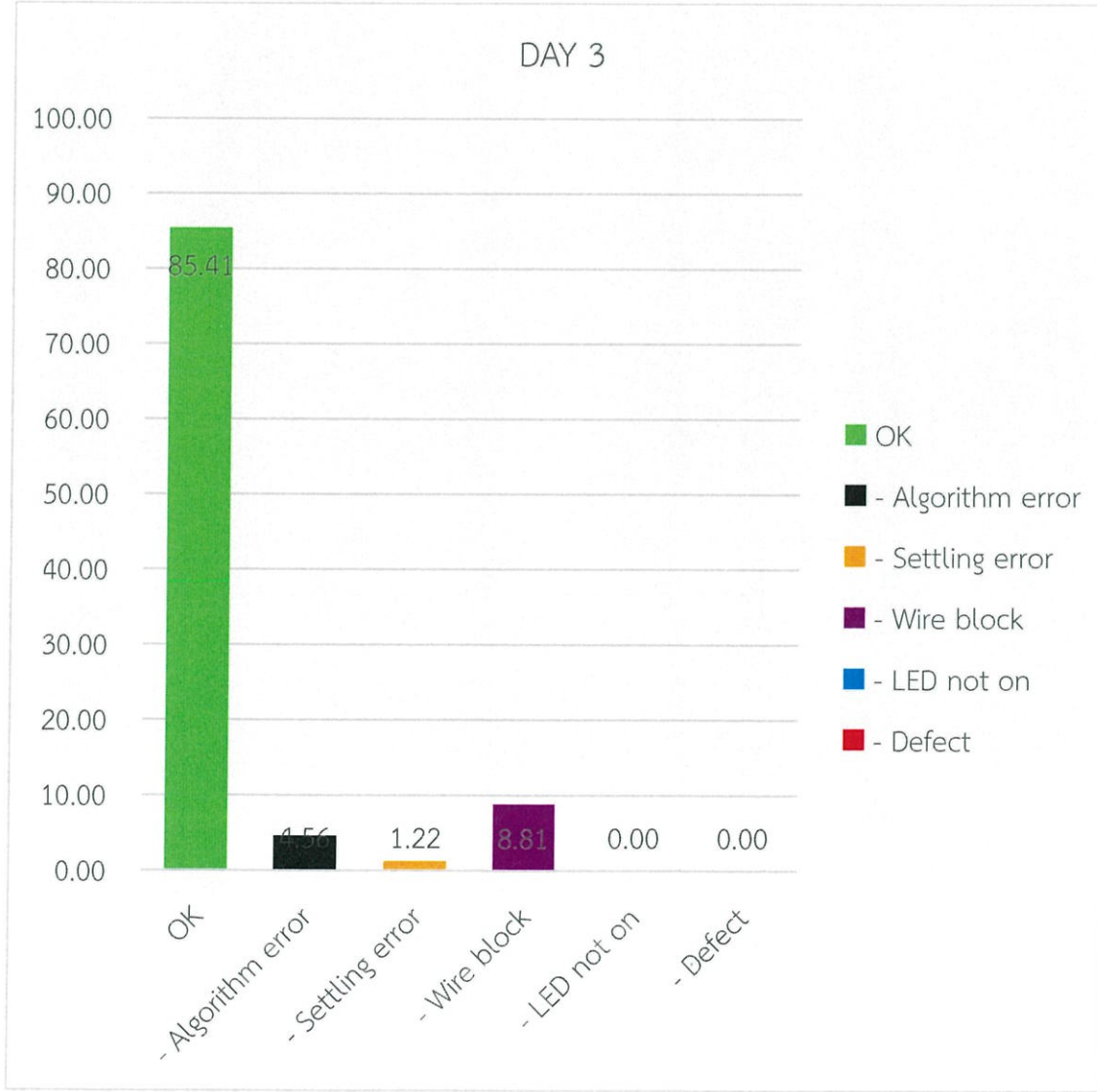
All	873	100.00	%
OK	748	85.68	%
NG	125	14.32	%
- Algorithm error	113	12.94	%
- Settling error	12	1.37	%
- Wire block	0	0.00	%
- LED not on	0	0.00	%
- Defect	0	0.00	%



รูปที่ 4.8 กราฟเปรียบเทียบปริมาณ Error วันที่ 2

ตารางที่ 4.5 ผลการทดสอบวันที่ 3

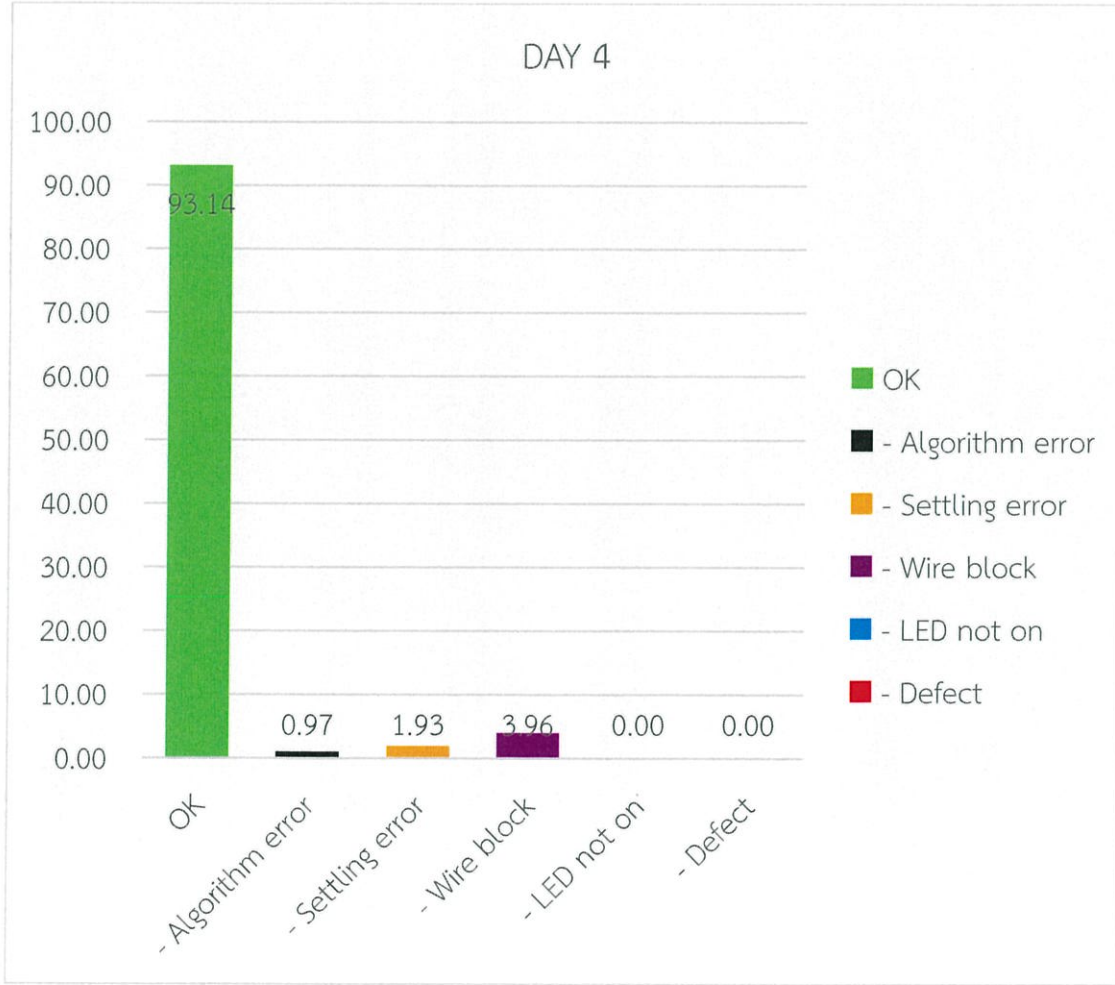
All	329	100.00	%
OK	281	85.41	%
NG	48	14.59	%
- Algorithm error	15	4.56	%
- Settling error	4	1.22	%
- Wire block	29	8.81	%
- LED not on	0	0.00	%
- Defect	0	0.00	%



รูปที่ 4.9 กราฟเปรียบเทียบปริมาณ Error วันที่ 3

ตารางที่ 4.6 ผลการทดสอบวันที่ 4

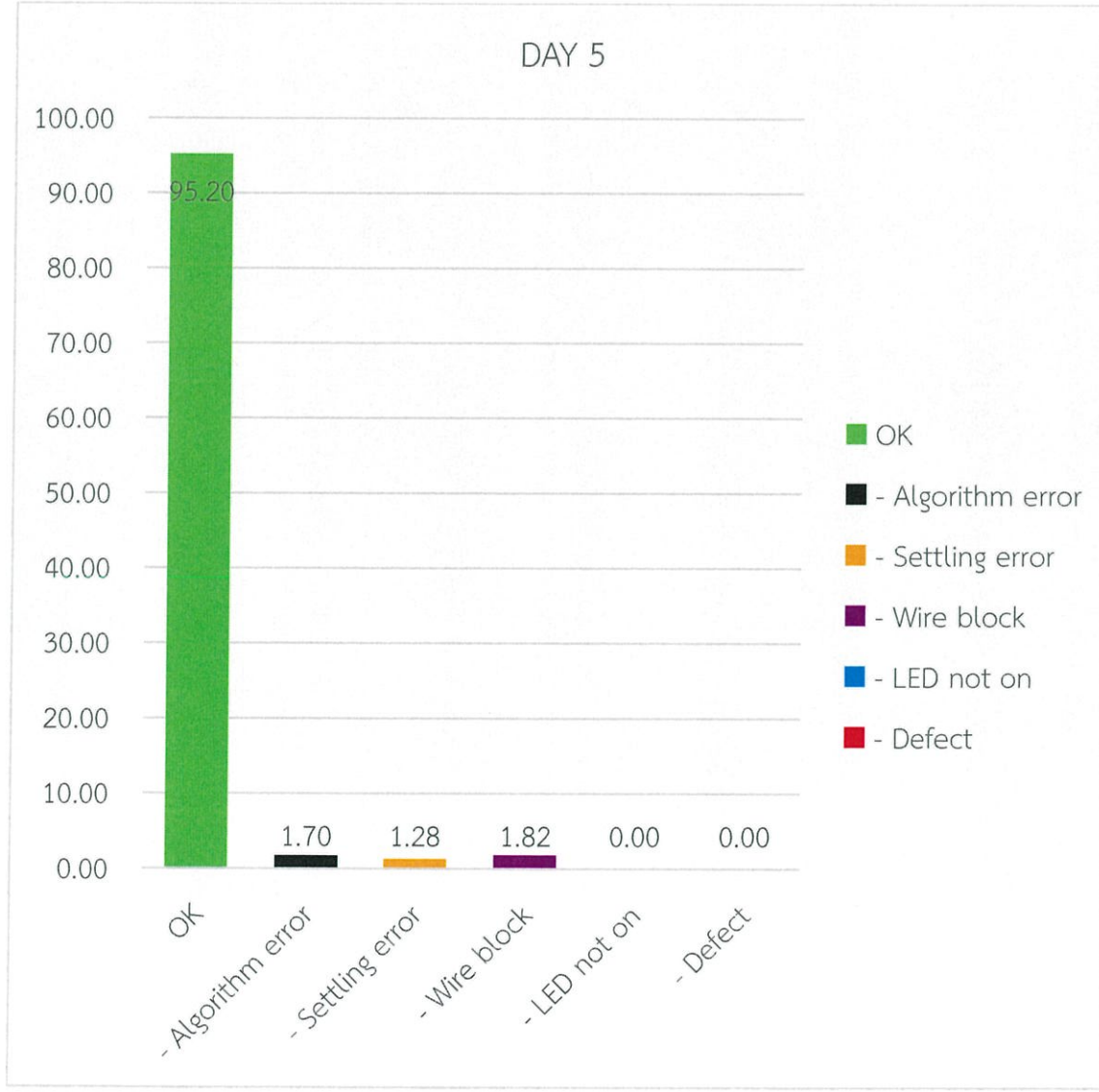
All	1137	100.00	%
OK	1059	93.14	%
NG	78	6.86	%
- Algorithm error	11	0.97	%
- Settling error	22	1.93	%
- Wire block	45	3.96	%
- LED not on	0	0.00	%
- Defect	0	0.00	%



รูปที่ 4.10 กราฟเปรียบเทียบปริมาณ Error วันที่ 4

ตารางที่ 4.7 ผลการทดสอบวันที่ 5

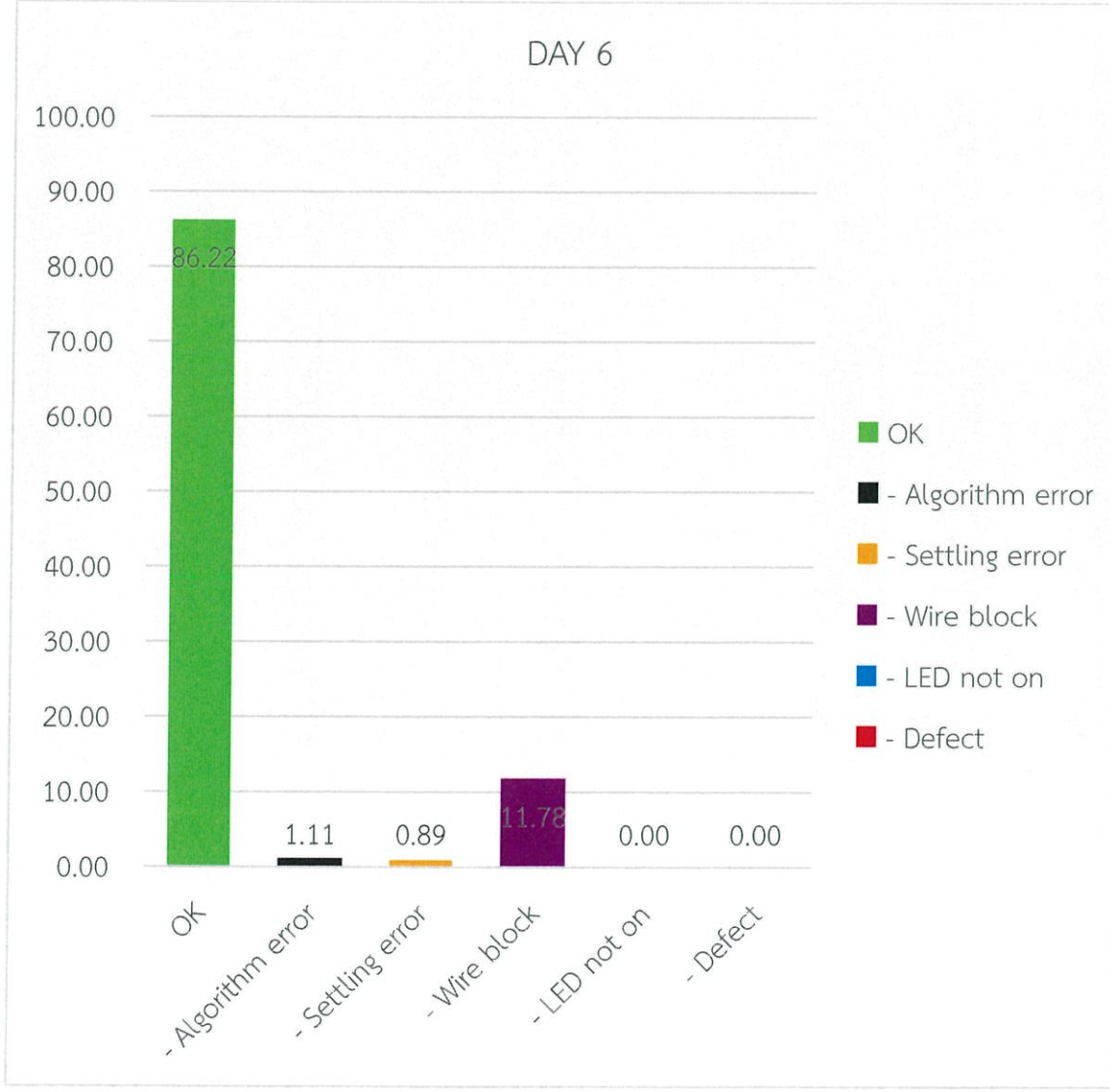
All	1646	100.00	%
OK	1567	95.20	%
NG	79	4.80	%
- Algorithm error	28	1.70	%
- Settling error	21	1.28	%
- Wire block	30	1.82	%
- LED not on	0	0.00	%
- Defect	0	0.00	%



รูปที่ 4.11 กราฟเปรียบเทียบปริมาณ Error วันที่ 5

ตารางที่ 4.8 ผลการทดสอบวันที่ 6

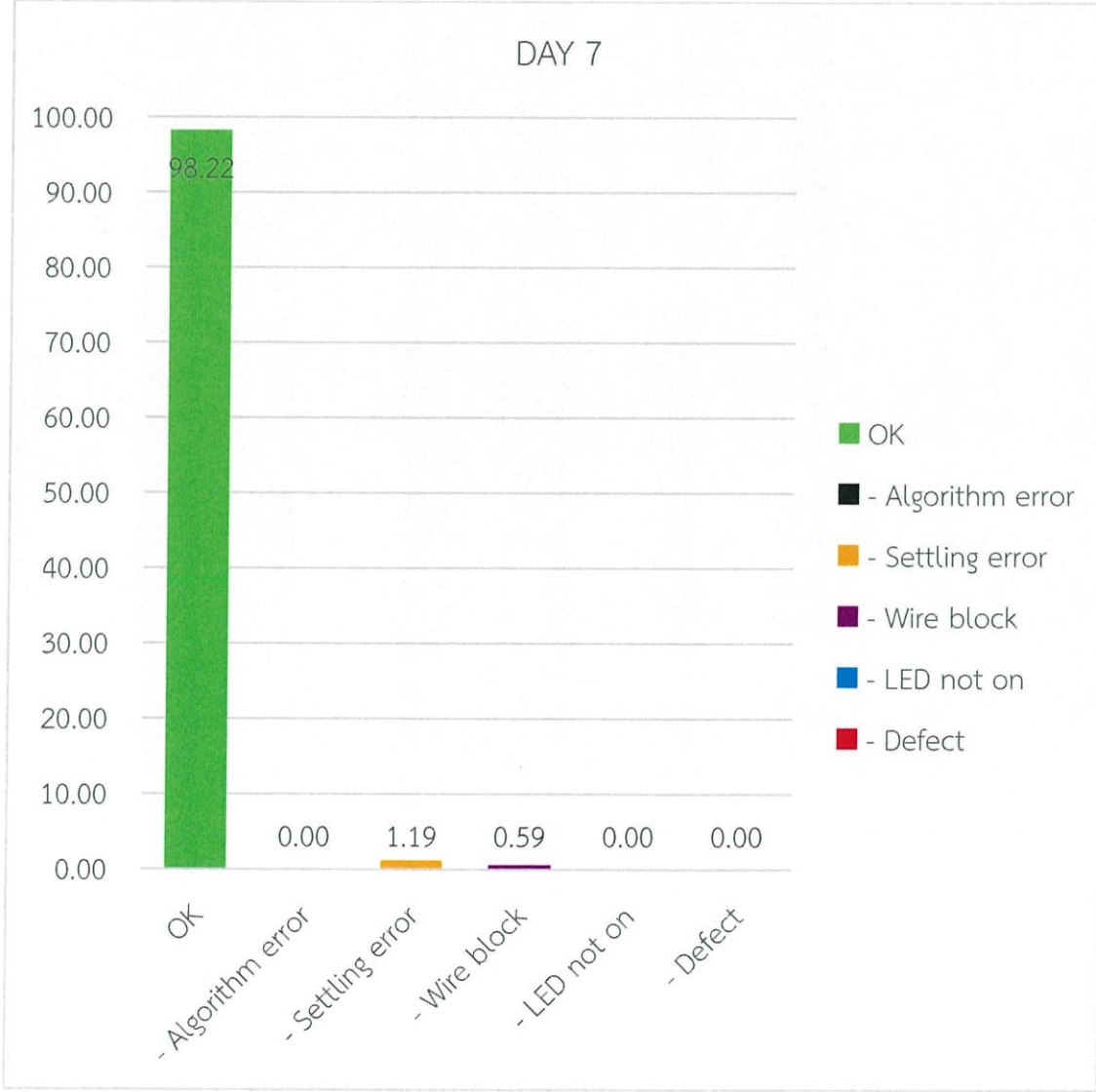
All	450	100.00	%
OK	388	86.22	%
NG	62	13.78	%
- Algorithm error	5	1.11	%
- Settling error	4	0.89	%
- Wire block	53	11.78	%
- LED not on	0	0.00	%
- Defect	0	0.00	%



รูปที่ 4.12 กราฟเปรียบเทียบปริมาณ Error วันที่ 6

ตารางที่ 4.9 ผลการทดสอบวันที่ 7

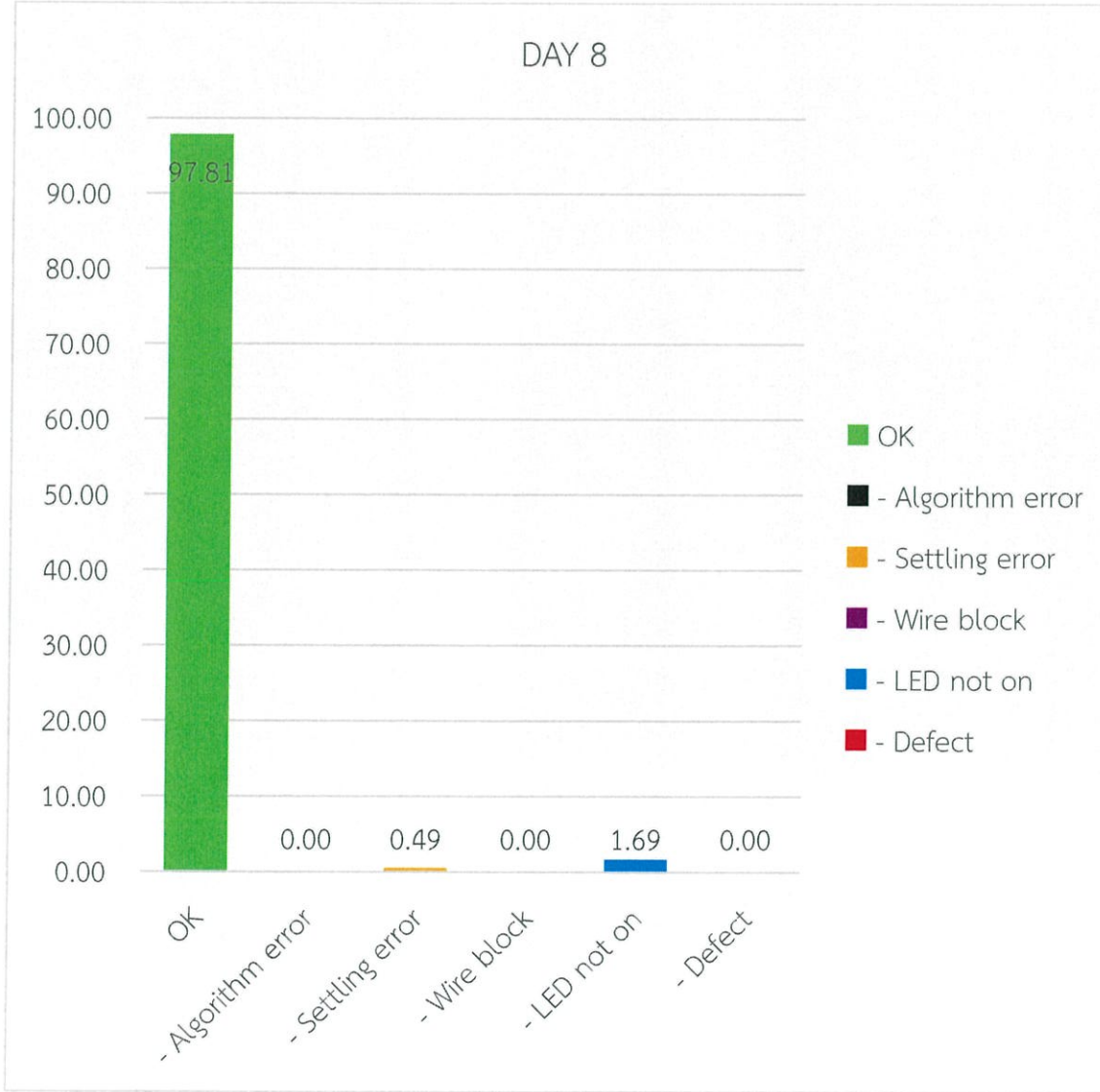
All	1514	100.00	%
OK	1487	98.22	%
NG	27	1.78	%
- Algorithm error	0	0.00	%
- Settling error	18	1.19	%
- Wire block	9	0.59	%
- LED not on	0	0.00	%
- Defect	0	0.00	%



รูปที่ 4.13 กราฟเปรียบเทียบปริมาณ Error วันที่ 7

ตารางที่ 4.10 ผลการทดสอบวันที่ 8

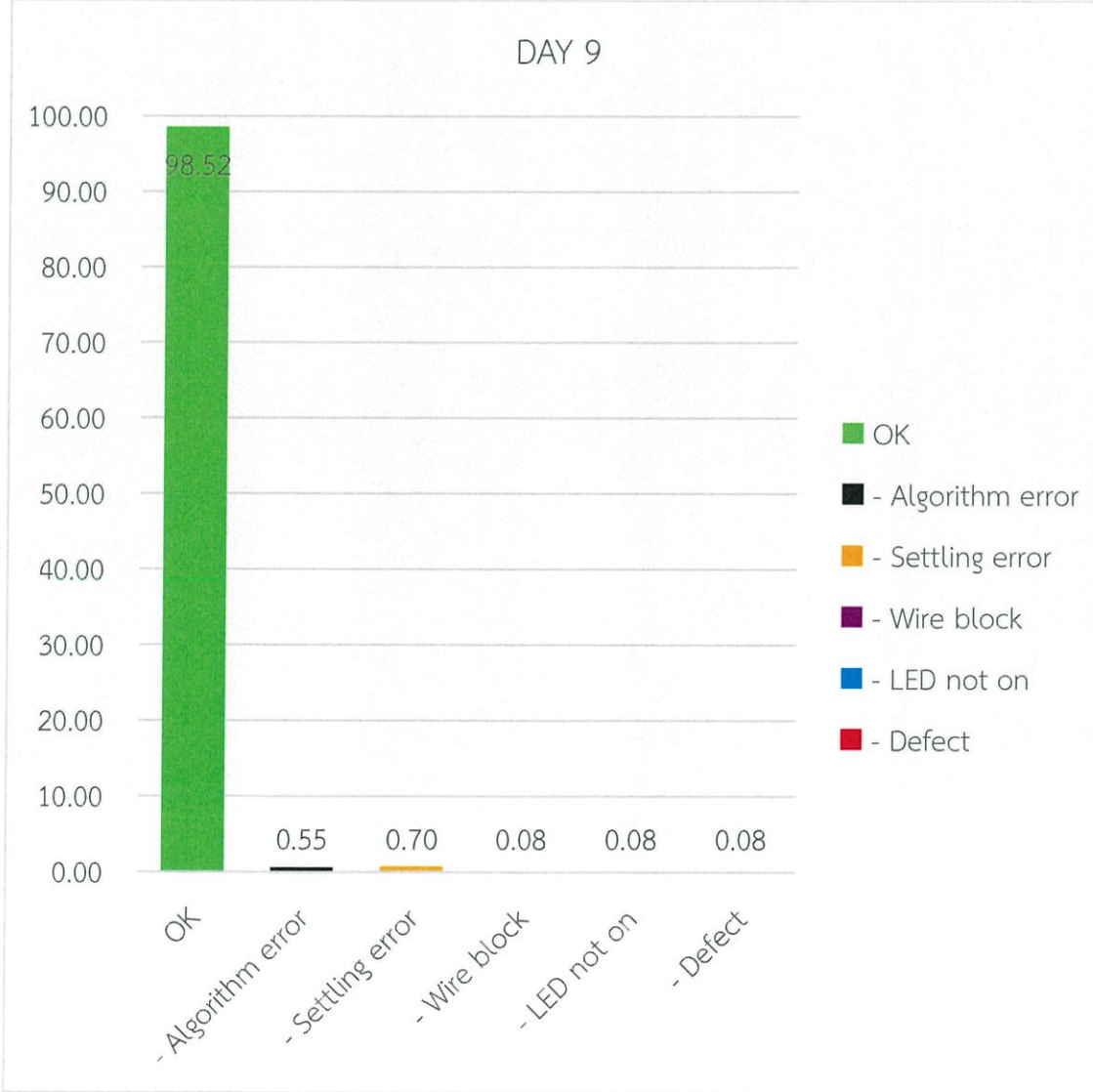
All	1417	100.00	%
OK	1386	97.81	%
NG	31	2.19	%
- Algorithm error	0	0.00	%
- Settling error	7	0.49	%
- Wire block	0	0.00	%
- LED not on	24	1.69	%
- Defect	0	0.00	%



รูปที่ 4.14 กราฟเปรียบเทียบปริมาณ Error วันที่ 8

ตารางที่ 4.11 ผลการทดสอบวันที่ 9

All	1280	100.00	%
OK	1261	98.52	%
NG	19	1.48	%
- Algorithm error	7	0.55	%
- Settling error	9	0.70	%
- Wire block	1	0.08	%
- LED not on	1	0.08	%
- Defect	1	0.08	%



รูปที่ 4.15 กราฟเปรียบเทียบปริมาณ Error วันที่ 9

บทที่ 5
สรุปผลการดำเนินโครงงานและข้อเสนอแนะ

จากการดำเนินงานในแต่ละขั้นตอน ทำให้พบปัญหาที่มีส่งผลในการวิเคราะห์รูปภาพหลายสาเหตุ ทั้งปัจจัยจากโปรแกรม และจากสภาพแวดล้อมบริเวณตำแหน่งติดตั้งงาน ในบทนี้กล่าวถึงปัญหาสำคัญที่มีผลต่อการวิเคราะห์ โดยวิธีแก้ไขปัญหามีทั้งที่สามารถดำเนินการแก้ไขให้เรียบร้อย และบางปัญหาที่มีการคิดแนวทางแก้ไขไว้แล้ว แต่ด้วยระยะเวลาที่มีจำกัดจึงไม่สามารถทำการแก้ไขได้ โดยปัญหาที่แก้ไขไปแล้ว ได้แก่ ตำแหน่งของถังซักมีความคลาดเคลื่อนสูง, Wire Blocked, Settling Error และ Algorithm Error โดยทำการแก้ไขโดยแยกตามปัญหาได้ ดังนี้

5.1 ตำแหน่งของถังเครื่องซักผ้ามีความไม่แน่นอน

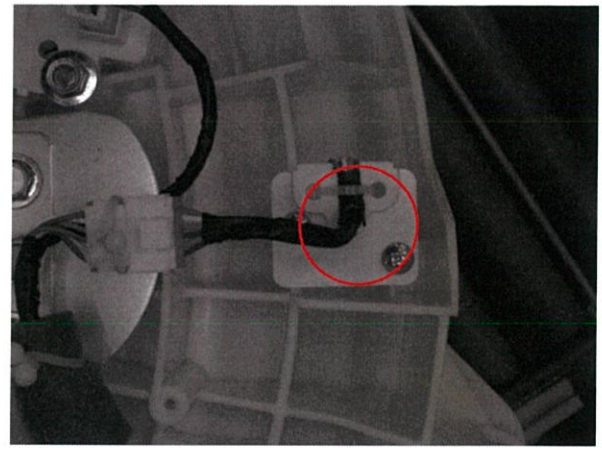
การเคลื่อนที่ของถังซักผ้าที่ผ่านทางสายพานลำเลียงนั้น จะต้องผ่านพนักงานที่ทำการยิงน็อตและสกรู เพื่อทำการยึดชิ้นส่วนมอเตอร์ ชุดเบรก สายดิน และชิ้นส่วนต่างๆ โดยในการทำงานนั้นพนักงานส่วนใหญ่จะทำการสัมผัสกับชิ้นงานโดยตรง ซึ่งอาจทำให้ตัวถังเครื่องซักผ้ามีการขยับและหมุนไปในองศาที่แตกต่างกัน ดังรูปที่ 5.1 ทำให้ถังเครื่องซักผ้าที่ถูกลำเลียงเข้ามาสู่ระบบนั้นเกิดความคลาดเคลื่อนที่มากเกินกว่าระบบจะรับได้ อีกทั้งสายการผลิตนี้ยังทำการผลิตเครื่องซักผ้าถึง 7 Model ด้วยกันทำให้ระบบตรวจสอบหัวน็อตนั้นเกิดความผิดพลาดได้ จึงจำเป็นต้องออกแบบ Guide ดังรูปที่ 5.2 เพื่อบังคับให้ทิศทางของถังเครื่องซักผ้ามีความคลาดเคลื่อนอยู่ในขอบเขตที่ระบบสามารถรับได้

รูปที่ 5.1 ก่อนทำการติดตั้ง Guide

รูปที่ 5.2 หลังทำการติดตั้ง Guide

5.2 สายไฟบังจากกระบวนการผลิตก่อนหน้า (Wire Block)

หลังจากทำการทดสอบระบบทำให้พบว่าในบางตำแหน่งที่ทำการตรวจสอบหัวน็อตนั้นมีสายไฟมาบังหัวน็อตในบางครั้ง ดังรูปที่ 5.3 ทำให้ระบบวิเคราะห์ผลผิดพลาดและนับถังซักผ้าชิ้นนั้นเป็นของไม่ผ่านคุณภาพ จึงได้ทำการบอกกล่าวกับหัวหน้าฝ่ายการผลิตให้บอกพนักงานที่ทำการผลิตให้ขยับสายไฟไม่ให้บังหัวน็อต



รูปที่ 5.3 สายไฟบังหัวน็อต

5.3 ปัญหาระบบหาจุดศูนย์กลางถังเครื่องซักผ้าผิด (Settling Error)

ในการตรวจสอบหัวน็อตนั้น จะทำการหาจุดศูนย์กลางของถังเครื่องซักผ้าก่อนแล้วจึงทำการเคลื่อนที่ และหมุน Tools ของหุ่นยนต์ให้มีองศาขนานกับถังเครื่องซักผ้า แล้วจึงเริ่มกระบวนการเคลื่อนที่ไปยังตำแหน่งต่างๆ เพื่อตรวจสอบหัวน็อตบนถังเครื่องซักผ้าในแนวแกน X Y และ Z ตามลำดับโดยอ้างอิงจากจุดศูนย์กลางถังเครื่องซักผ้า โดยปกติแล้วจะทำการวนลูปหาจุดศูนย์กลางเพียงแค่ครั้งเดียว แต่ในการทำงานจริงนั้นพบว่าระบบเกิดปัญหาในการคำนวณ และเคลื่อนที่เข้าสู่จุดศูนย์กลางถังเครื่องซักผ้าที่ผิดพลาด และเมื่อจุดศูนย์กลางผิดพลาดแล้วจึงทำให้ตำแหน่งทุกไตำแหน่งที่ต้องการจะทำการตรวจสอบเกิดความผิดพลาดไปด้วย จึงทำการแก้ไขโดยการเพิ่มลูปการตรวจสอบและคำนวณหาจุดศูนย์กลางให้มีจำนวนครั้งที่มากขึ้น ซึ่งส่งผลให้ค่าที่ได้จะมีการลู่เข้าสู่จุดศูนย์กลางซึ่งจะช่วยให้ระบบมีความแม่นยำที่สูงขึ้นตามไปด้วย

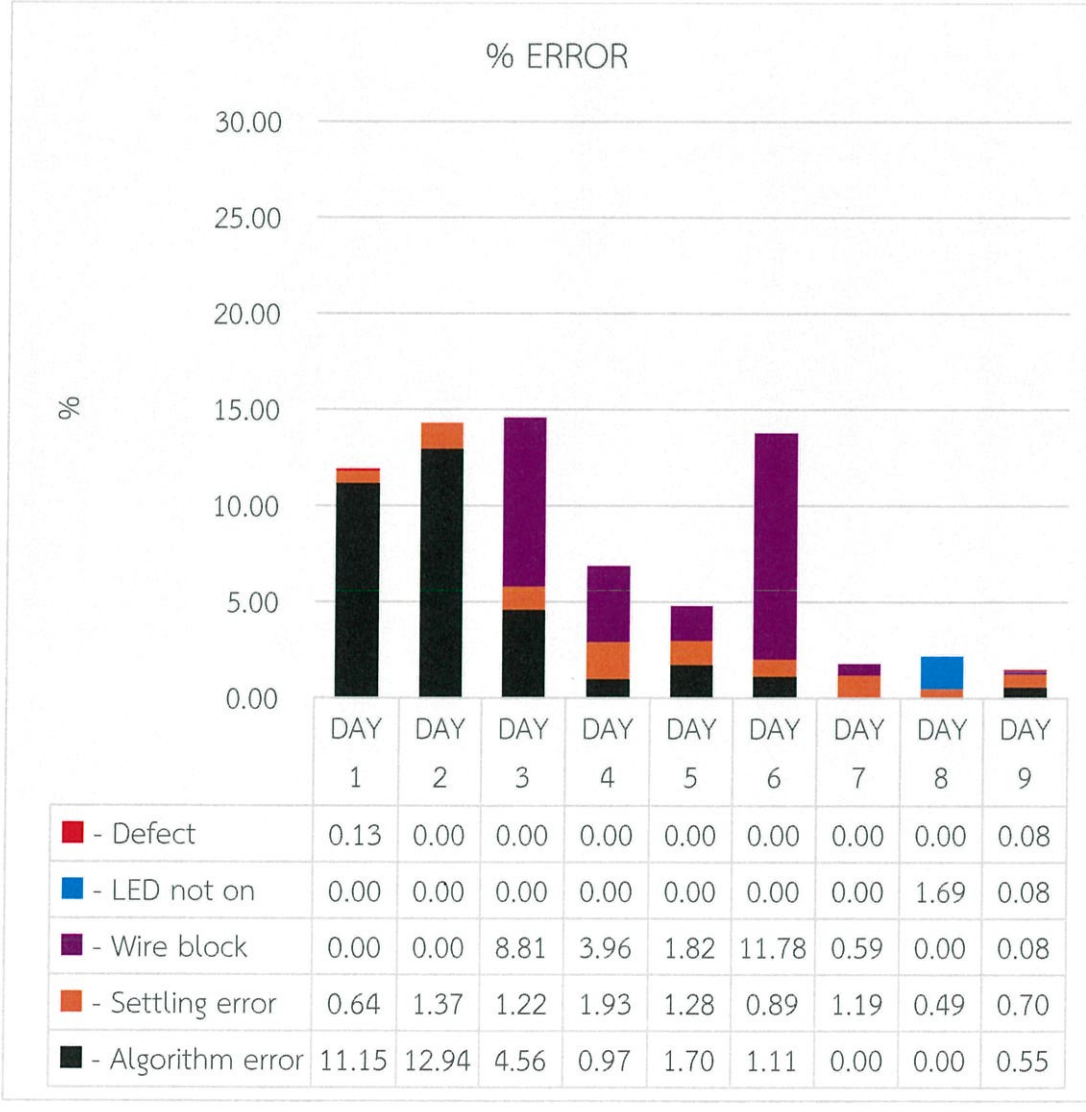
5.4 การประมวลผลภาพไม่มีความแม่นยำ (Algorithm Error)

เมื่อระบบเริ่มได้ถูกใช้งาน จึงได้ทำการบันทึกข้อมูลเพื่อนำมาประมวลผล แล้วพบว่าระบบการตรวจสอบน็อตนั้นได้ส่งผลที่ผิดพลาดออกมา (Fake Defect) โดยระบบได้ทำการตรวจสอบตำแหน่งที่มีหัวน็อต แต่กลับให้ผลว่าตำแหน่งดังกล่าวนั้นไม่พบหัวน็อต จึงทำการปรับค่า Threshold ของรูปภาพที่ใช้ในการ Sampling Contours ให้มีค่าน้อยลง เพื่อทำให้รูปภาพที่มี Pixel ที่มีค่า Gray Scale ต่ำนั้นถูก Sampling Contours ด้วยอีกทั้งยังได้ใช้ Filter Equalizer เพื่อทำให้ค่า Pixel มีความสมดุลมากขึ้น และได้ทำการปรับค่า Accuracy Score ให้น้อยลง เพื่อทำให้ระบบยอมรับ Contours ที่มีรูปแบบใกล้เคียงกับภาพต้นแบบให้มากขึ้น แต่อย่างไรก็ตามการแก้ไขปัญหาดังกล่าวก็ยังไม่ส่งผลให้ผลที่ได้มีค่าแม่นยำมากพอ จากสภาพแวดล้อมในรูปที่ 5.4 จึงได้ตัดสินใจสร้าง Cover ดังรูปที่ 5.5 เพื่อบังแสงที่กระเจิงเข้ามาสู่ถังเครื่องซักผ้า ซึ่งเป็น Noise รบกวนระบบตรวจสอบหัวน็อต แล้วจึงทำการเก็บข้อมูลใหม่อีกครั้ง

รูปที่ 5.4 ก่อนติด Cover กันแสง

รูปที่ 5.5 หลังติด Cover เพื่อกันแสงรบกวนจากบริเวณรอบ

หลังจากที่ทำการติดตั้งและแก้ไขปัญหาที่พบ ได้บันทึกผลทั้ง 9 วัน เพื่อเปรียบเทียบผลในแต่ละวันว่ามีการเปลี่ยนแปลงหลังจากการแก้ไขมากน้อยเพียงใด ซึ่งสามารถดูความเปลี่ยนแปลงได้จากรูปที่ 5.6



	DAY 1	DAY 2	DAY 3	DAY 4	DAY 5	DAY 6	DAY 7	DAY 8	DAY 9
- Defect	0.13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08
- LED not on	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.69	0.08
- Wire block	0.00	0.00	8.81	3.96	1.82	11.78	0.59	0.00	0.08
- Settling error	0.64	1.37	1.22	1.93	1.28	0.89	1.19	0.49	0.70
- Algorithm error	11.15	12.94	4.56	0.97	1.70	1.11	0.00	0.00	0.55

รูปที่ 5.6 กราฟเปรียบเทียบปริมาณ Error หลังทำการแก้ไขในแต่ละวัน

จากรูปที่ 5.6 จะพบว่าระบบสามารถตรวจสอบได้ดีขึ้น แต่ทว่าก็ยังมีข้อผิดพลาดที่เกิดจาก Algorithm ซึ่งควรทำการแก้ไขต่อไป แต่ได้เล็งเห็นแล้วว่า ซอฟต์แวร์แลปวิว (Software LabVIEW) มีประสิทธิภาพไม่เพียงพอ ที่จะทำให้ระบบการตรวจสอบนั้นมีความแม่นยำร้อยเปอร์เซ็นต์ เล็งเห็นว่าควรนำระบบปัญญาประดิษฐ์มาช่วยในการตรวจสอบ เพื่อให้ระบบดังกล่าวนั้นสร้างรูปแบบและเรียนรู้รูปร่างของหัวน็อตให้แม่นยำ เพื่อการประมวลผลที่มีประสิทธิภาพสูงสุด

นอกจากปัญหาข้างต้นแล้ว ยังต้องปรับปรุงในเรื่องของเวลาในการทำงานของระบบ เนื่องจากระยะเวลาในการทำงานในปัจจุบันนั้นมากกว่าที่ทางโรงงานนั้นต้องการ ซึ่งเกิดจากความสามารถในการจับภาพของกล้องที่นำมาใช้มีมี Frame Rate ที่ต่ำ จึงทำให้ต้องทำการหน่วงเวลาในการถ่ายภาพในแต่ละจุดทุกครั้ง นอกจากนั้นแล้วการประมวลผลของโปรแกรมมีความล่าช้าเนื่องจากความสามารถของคอมพิวเตอร์ไม่เพียงพอต่อการทำงานของระบบ จากปัญหาดังกล่าวผู้จัดทำจึงแนะนำให้ทำการเปลี่ยนกล้องให้มี Frame Rate ที่สูงขึ้น และคอมพิวเตอร์ให้มีคุณสมบัติที่มากขึ้นเพื่อรองรับกำลังผลิตที่จะขยายตัวเพิ่มขึ้นต่อไปในอนาคต

เอกสารอ้างอิง

[1] **การเลือกใช้ Circuit Breaker.** (Online). 27 ตุลาคม 2560.
Available : https://mall.factomart.com/circuit-breaker/how-to-select-a-circuit-breaker/

[2] **ทฤษฎี LabVIEW.** (Online). 28 ตุลาคม 2560.
Available : http://www.research-system.siam.edu /images/coop/DESIGN_AND_ CONSTRUCTION_OF_ELECTRICAL _MEASUREMENT_USING_LABVIEW___PROGRAM

[3] **หุ่นยนต์.** (Online). 28 ตุลาคม 2560.
Available : http://dusithost.dusit.ac.th/~juthawut_cha/download/ Lession%2010%20Robotics.pdf

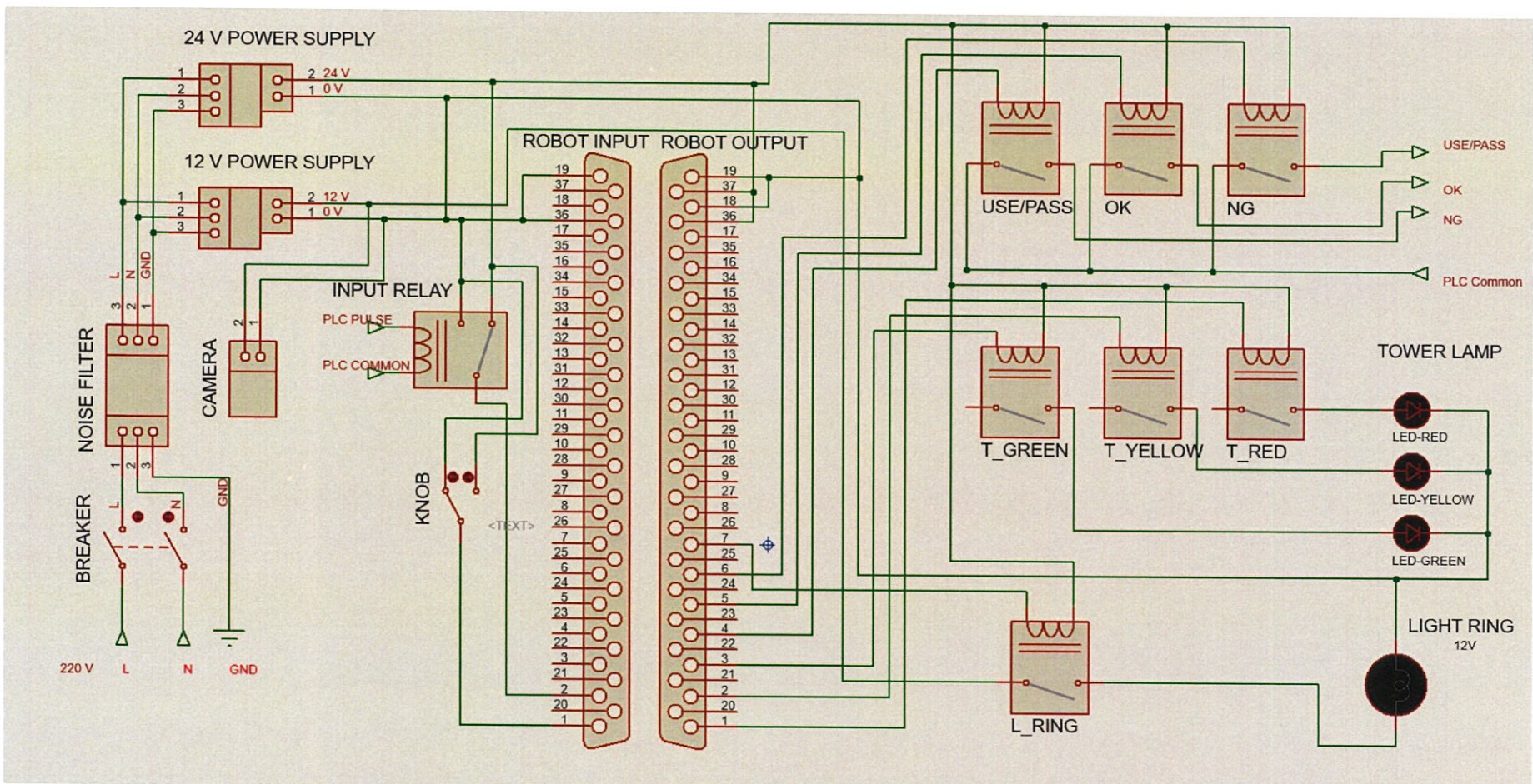
[4] **หลักการประมวลผลภาพ Digital Image processing.** (Online). 30 ตุลาคม 2560.
Available : http://www.vcharkarn.com/vcafe/180103

[5] **Industrial Robot Type.** (Online). 31 ตุลาคม 2560.
Available : https://www.applicadthai.com/articles/หุ่นยนต์อุตสาหกรรม-industrial-robot-type

ภาคผนวก

ภาคผนวก ก.

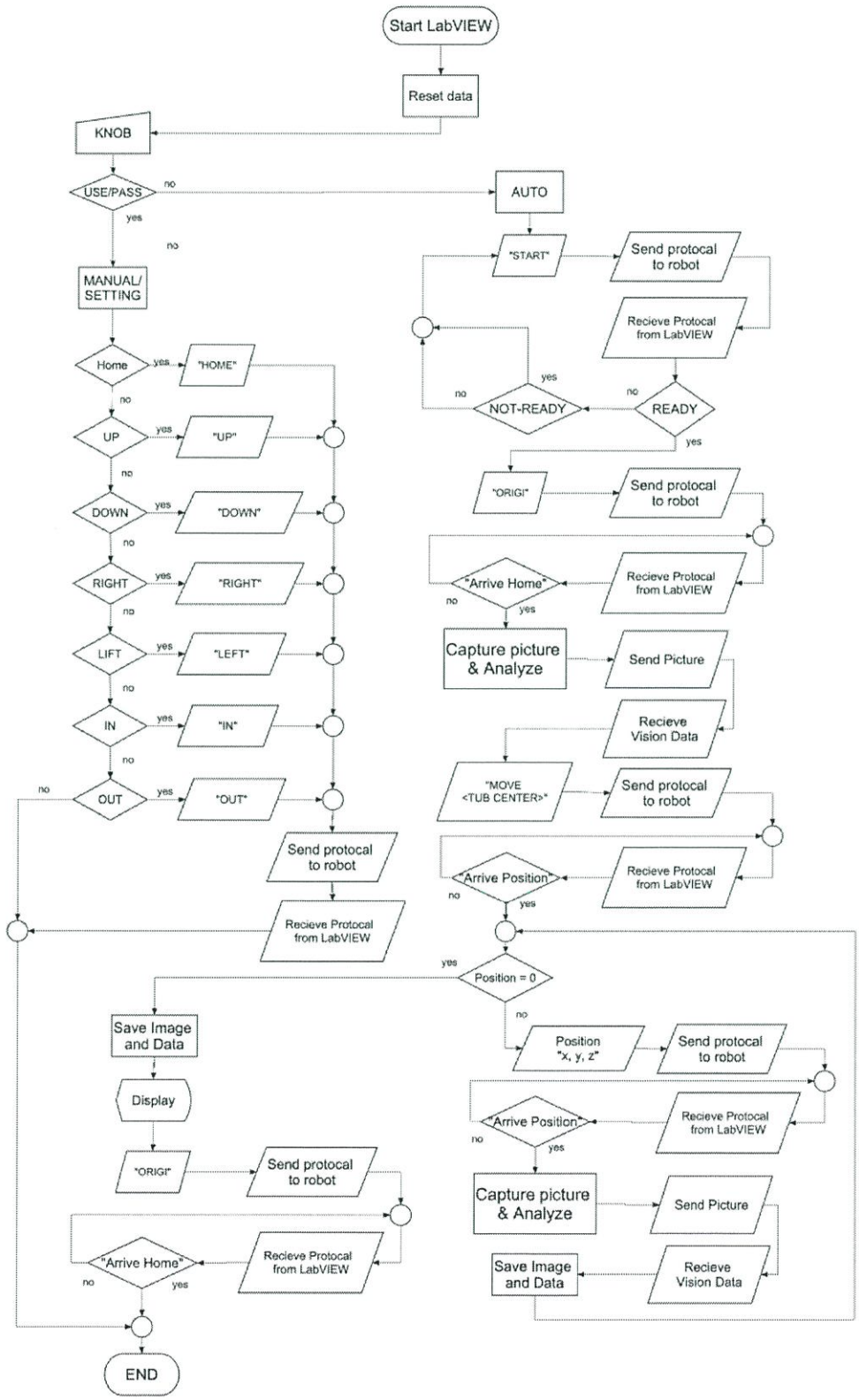
Wiring Diagram



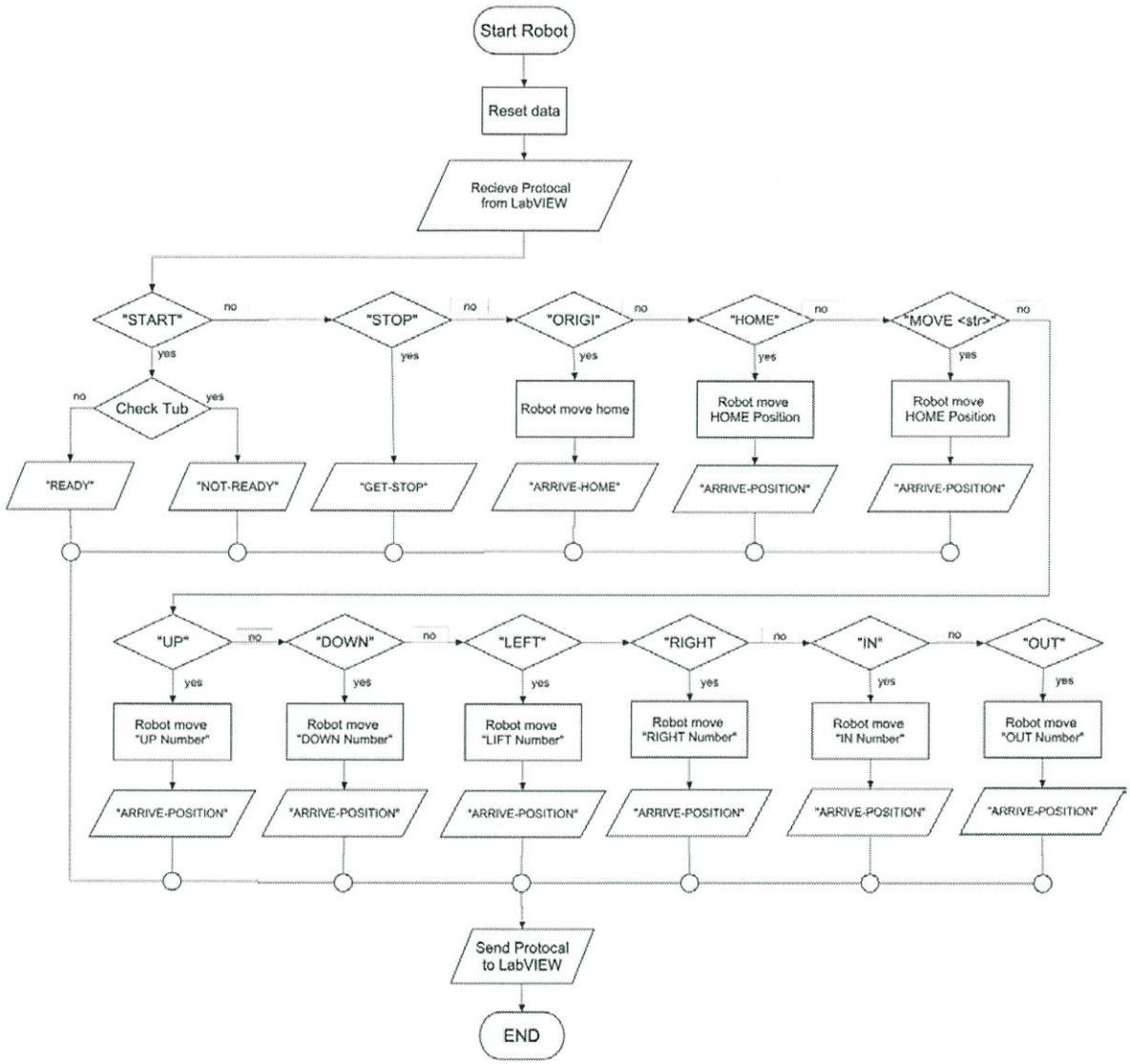
ภาคผนวก ข.

Programming Design–Flow Chart

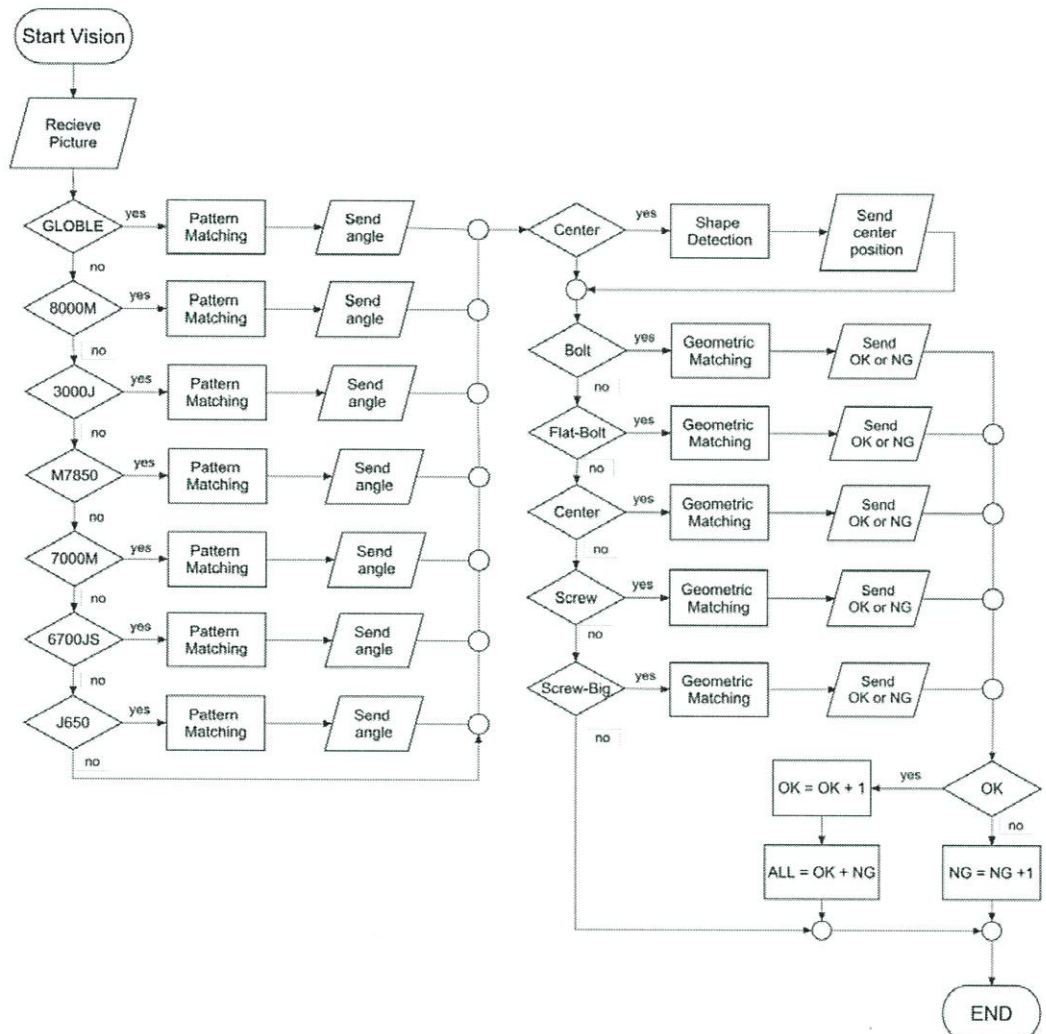
1. LabVIEW



2. Robot


Start Robot
Reset data
Recieve Protocal from LabVIEW
"START"
no
yes
Check Tub
no
yes
"READY"
"NOT-READY"
"STOP"
no
yes
"GET-STOP"
"ORIGI"
no
yes
Robot move home
"ARRIVE-HOME"
"HOME"
no
yes
Robot move HOME Position
"ARRIVE-POSITION"
"MOVE <str>"
no
yes
Robot move HOME Position
"ARRIVE-POSITION"
"UP"
no
yes
Robot move "UP Number"
"ARRIVE-POSITION"
"DOWN"
no
yes
Robot move "DOWN Number"
"ARRIVE-POSITION"
"LEFT"
no
yes
Robot move "LIFT Number"
"ARRIVE-POSITION"
"RIGHT
no
yes
Robot move "RIGHT Number"
"ARRIVE-POSITION"
"IN"
no
yes
Robot move "IN Number"
"ARRIVE-POSITION"
"OUT"
no
yes
Robot move "OUT Number"
"ARRIVE-POSITION"
Send Protocal to LabVIEW
END

3. Vision


Start Vision
Recieve Picture
GLOBLE
yes
Pattern Matching
Send angle
no
8000M
yes
Pattern Matching
Send angle
no
3000J
yes
Pattern Matching
Send angle
no
M7850
yes
Pattern Matching
Send angle
no
7000M
yes
Pattern Matching
Send angle
no
6700JS
yes
Pattern Matching
Send angle
no
J650
yes
Pattern Matching
Send angle
no
Center
yes
Shape Detection
Send center position
Bolt
yes
Geometric Matching
Send OK or NG
no
Flat-Bolt
yes
Geometric Matching
Send OK or NG
no
Center
yes
Geometric Matching
Send OK or NG
no
Screw
yes
Geometric Matching
Send OK or NG
no
Screw-Big
yes
Geometric Matching
Send OK or NG
no
OK
yes
OK = OK + 1
no
NG = NG +1
ALL = OK + NG
END

ภาคผนวก ค.
คู่มือการใช้งาน

1. การตรวจสอบความปลอดภัย

1. ควรทำการตรวจสอบความปลอดภัยขณะที่ปิดการทำงานของหุ่นยนต์
2. ควรตรวจสอบว่าชิ้นส่วนทุกชิ้นทำการยึดติดกันดีหรือไม่ สกรูหลวมหรือหลุดหรือไม่
3. ควรตรวจสอบสายไฟว่ามีการเกี่ยวหรือฉีกขาดตรงส่วนไหน
4. ควรตรวจสอบหน้ากล้องและเลนส์พร้อมทั้งทำความสะอาดในกรณีที่ฝุ่นเข้าไปเกาะ
5. ก่อนการเปิดใช้งานควรทำการปิดประตูนริภัยให้เรียบร้อย และตรวจสอบพื้นที่รอบๆ ก่อนการเปิดใช้งาน

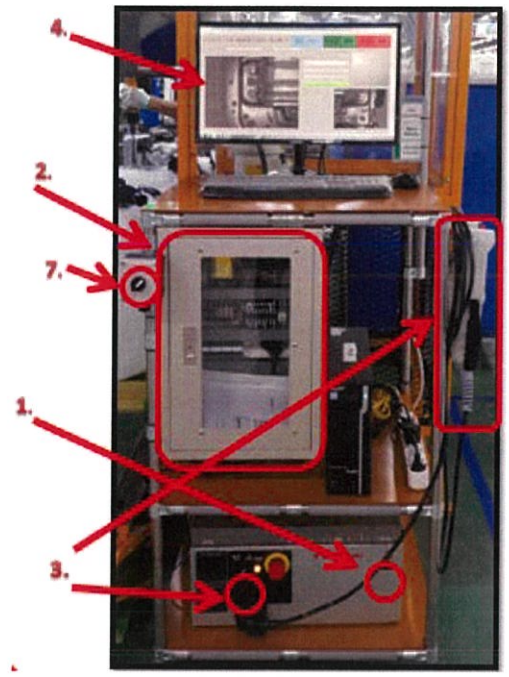
2.วัสดุอุปกรณ์

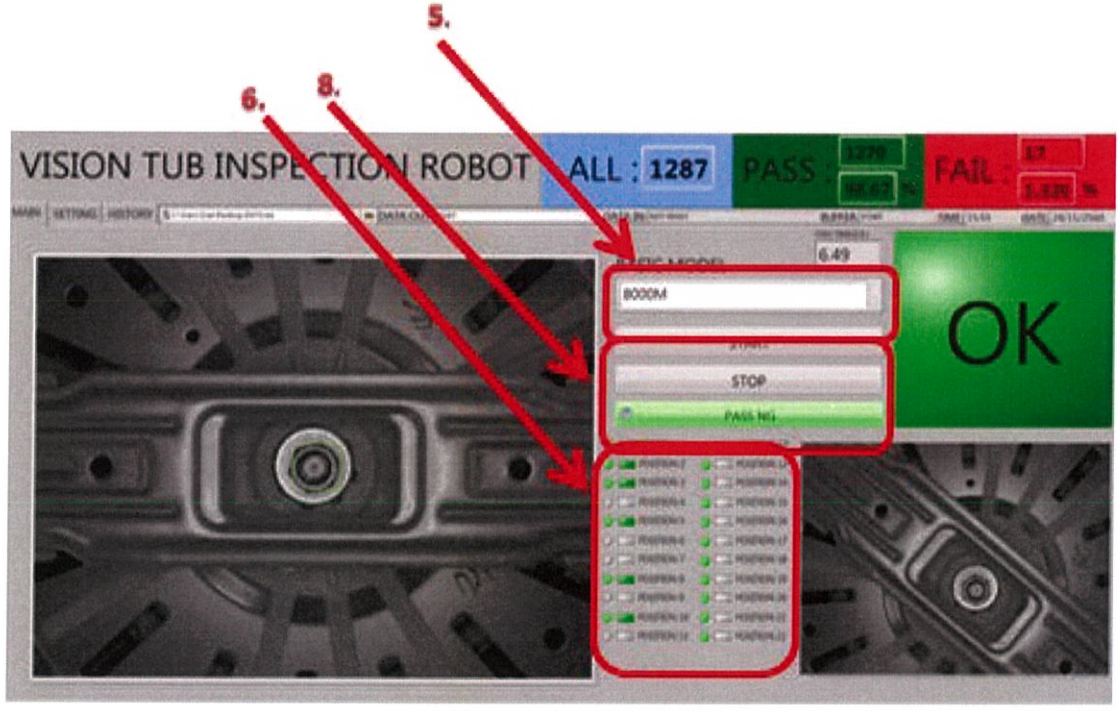
ตารางที่ ค-1 วัสดุอุปกรณ์ที่ใช้

Item	Q'ty	Unit	Material Cost
Robot Part			
Kawasaki Robot (RS005L)	1	set	480,000
Robot Base	1	pcs.	9,800
Robot Fence	1	set	15,000
Light Cover	1	pcs.	2,300
Camera Part			
Basler Camera (Aca 1300-75gm)	1	pcs.	23,200
Lense (M1214 - MP2)	1	pcs.	5,400
Ring Light (RL15-5028-24W)	1	pcs.	8,700
Camera Mount	1	set	1,000
Camera Power Cable	1	pcs.	1,800
Camera GigE Cable	1	pcs.	2,900
Computer Part			
PC (i7 Ram 16 GB)	1	set	57,580
LCD Display 24"	1	set	6,100
Mouse & Keyboard	1	set	0
PC Shelf	1	pcs.	4,000
GigE Hub	1	pcs.	1,500
Electronic Part			
Control Box #2	1	pcs.	400
Breaker (2P - 15A)	1	pcs.	450
Noise Filter (WYT - S10B 10A)	1	pcs.	550
Power Supply (24V - 4.2A)	1	pcs.	1,300
Power Supply (12V - 3A)	1	pcs.	300
Relay (24V - 5A)	2	pcs.	700
Relay Block (24V - 0.5A)	1	pcs.	3,500
Terminal I/O Link (40 PIN)	2	pcs.	4,000
Terminal (BA111T 16A)	10	pcs.	250
Terminal (TB2506 25A)	1	pcs.	50
I/O Cable (37 Pin M-F)	2	pcs.	2,400
Din Rail	3	pcs.	150
Wire Duct	1	pcs.	250
Wire (Red - 220V, Blue, Green-G	3	pcs.	2,850
Selector Switch	1	pcs.	250
Emergency Switch	1	pcs.	300
Tower Lamp	1	pcs.	1,500
Fence Safty Sensor	1	pcs.	750
Power Outlet	1	pcs.	350
Etc.			5,000

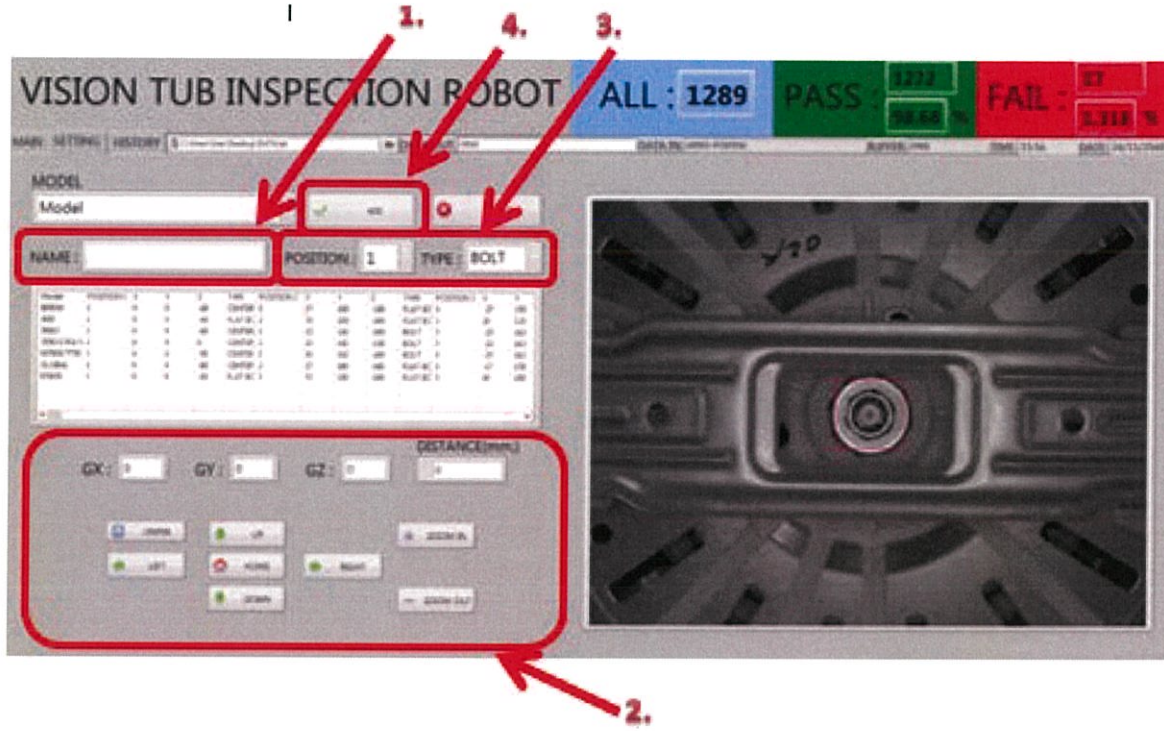
3. การใช้งานเครื่อง

1. เปิดสวิตช์โรบอท
2. เปิดสวิตช์ตู้ไฟ
3. เปิดโหมด Cycle Robot
4. เปิดโปรแกรมบนคอมพิวเตอร์ และ Run โปรแกรม
5. เลือกโมเดลที่จะใช้
6. เลือกจุดที่ต้องการจะทำการตรวจสอบ
7. บิดสวิตช์ USE/PASS
8. กดปุ่มStart





4. การเพิ่ม-ลบ Model



การเพิ่ม Model

1. ใส่ชื่อ Model ที่ต้องการ
2. เลื่อนไปยังตำแหน่งที่ต้องการ โดยการกรอกตัวเลขระยะทางในหน่วยมิลลิเมตรแล้วกดปุ่มเลือกทิศทาง (ควรเคลื่อนที่ไปให้หัวน๊อตอยู่ในกรอบสีแดง)
3. เลือกตำแหน่ง และชนิดของหัวน๊อต
4. กดปุ่ม Add (ควรสำเนาไฟล์ไว้บ่อยๆ)

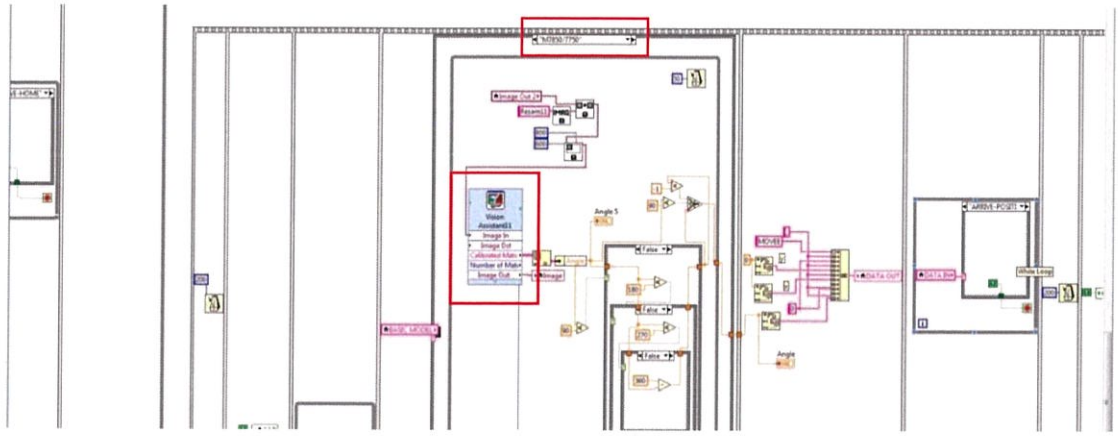
การลบ Model

1. เลือก Model บนแถบด้านบน
2. กดปุ่ม Delete

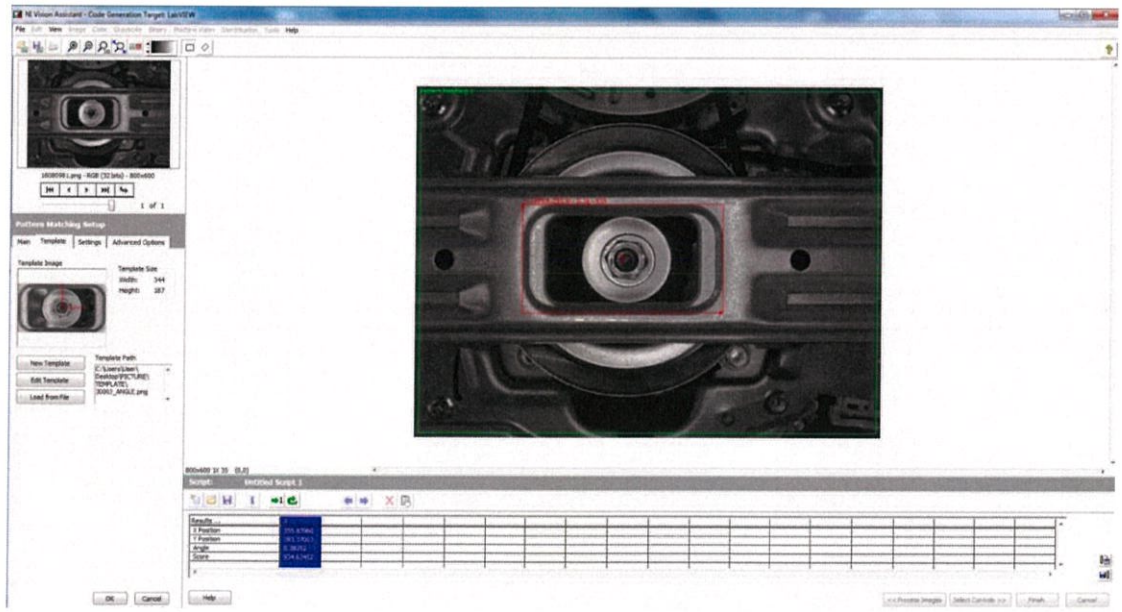
6. การตั้งค่าใน LabVIEW

Angle

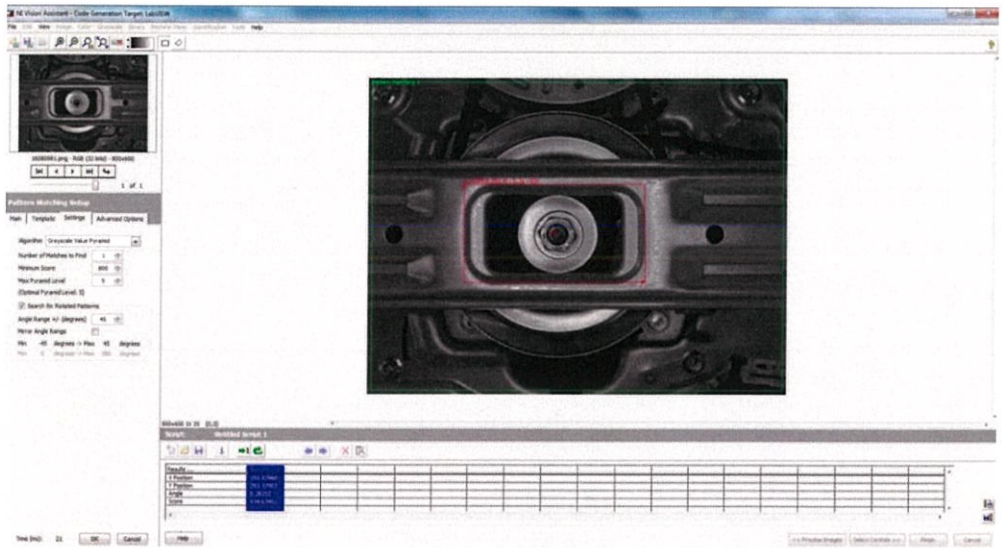
1. เลือก Vision Assistance ในโมเดลที่ต้องการแก้



2. สร้าง New Template บริเวณสี่เหลี่ยมตรงกลาง
3. ปรับปรุงเส้น Contour ให้เหลือเพียงเส้นที่ต้องการ

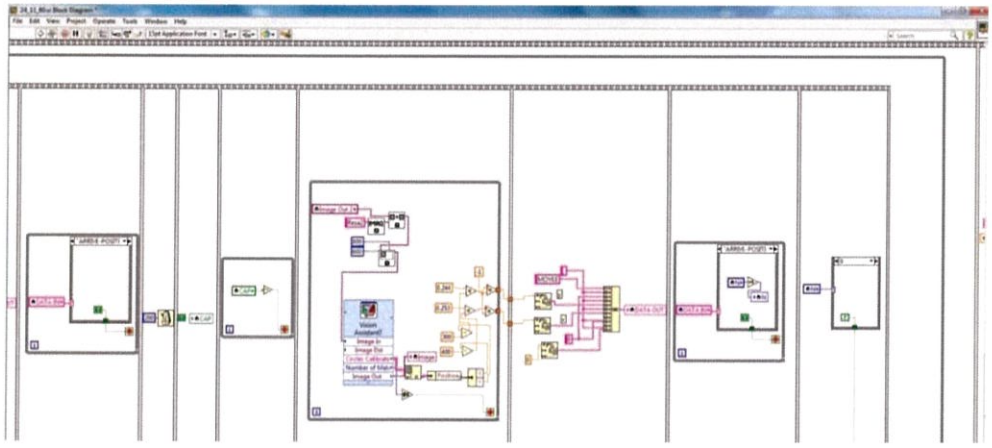


4. ปรับ Accuracy Score ให้เหมาะสม

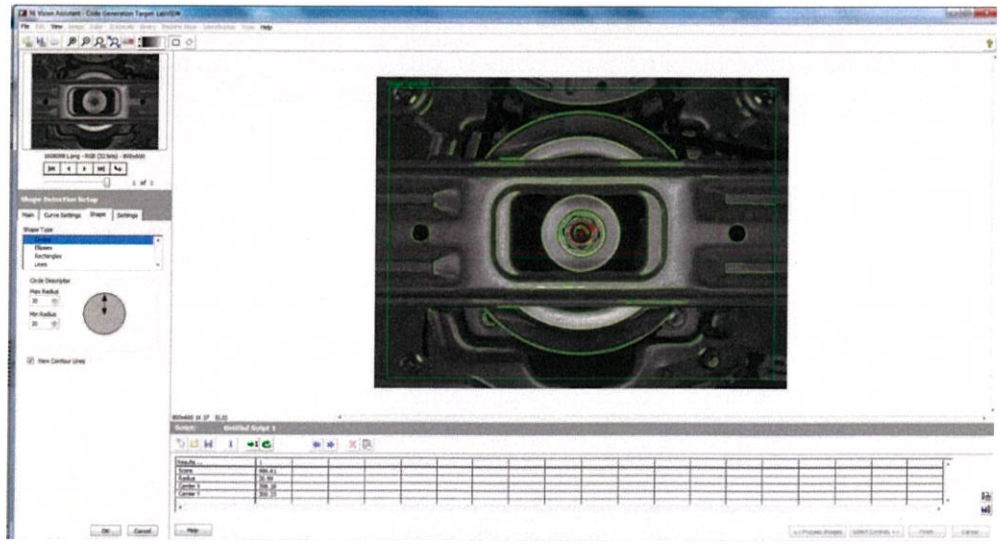


Center

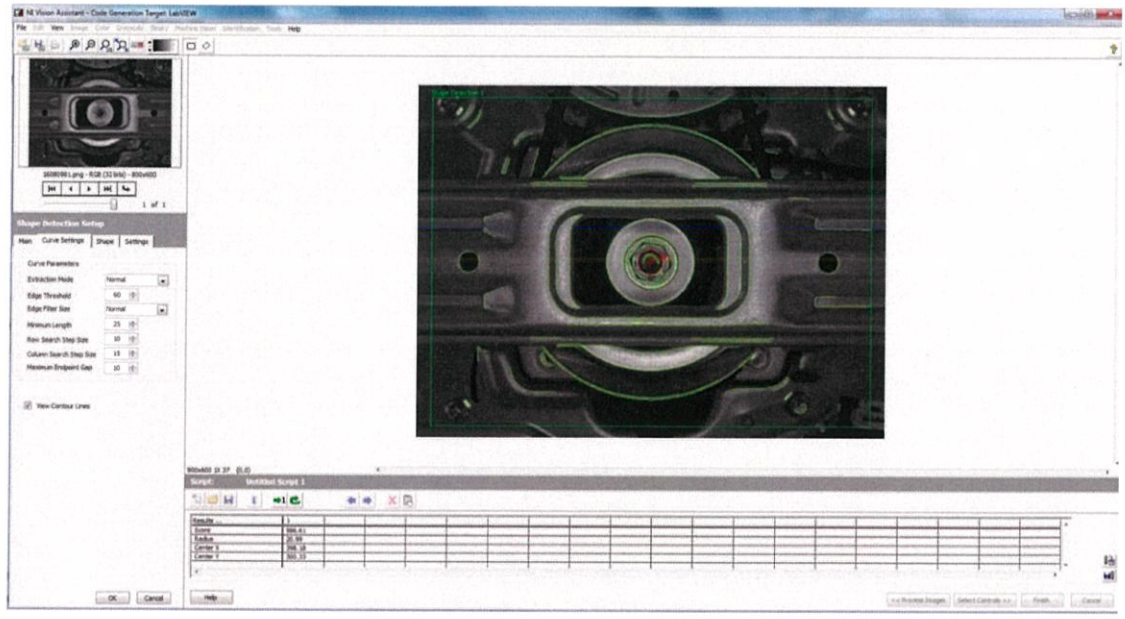
1. เลือก Vision Assistance



2. เลือก Shape Type ที่ต้องการ ในส่วนของ Center จำเป็นต้องใช้เป็น Circles

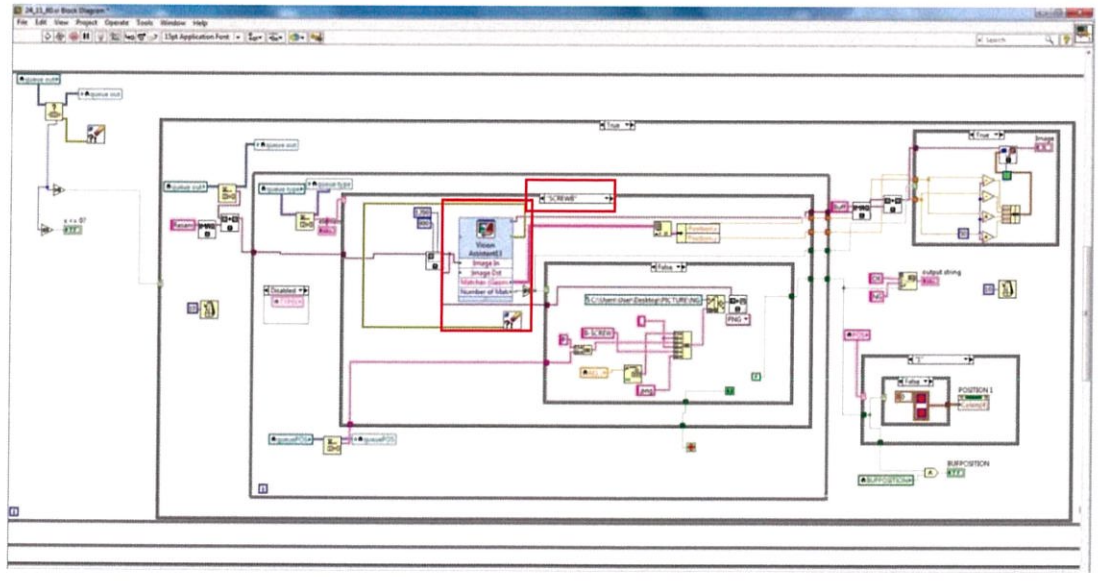


3. ปรับค่า Edge Threshold ให้เหมาะสม



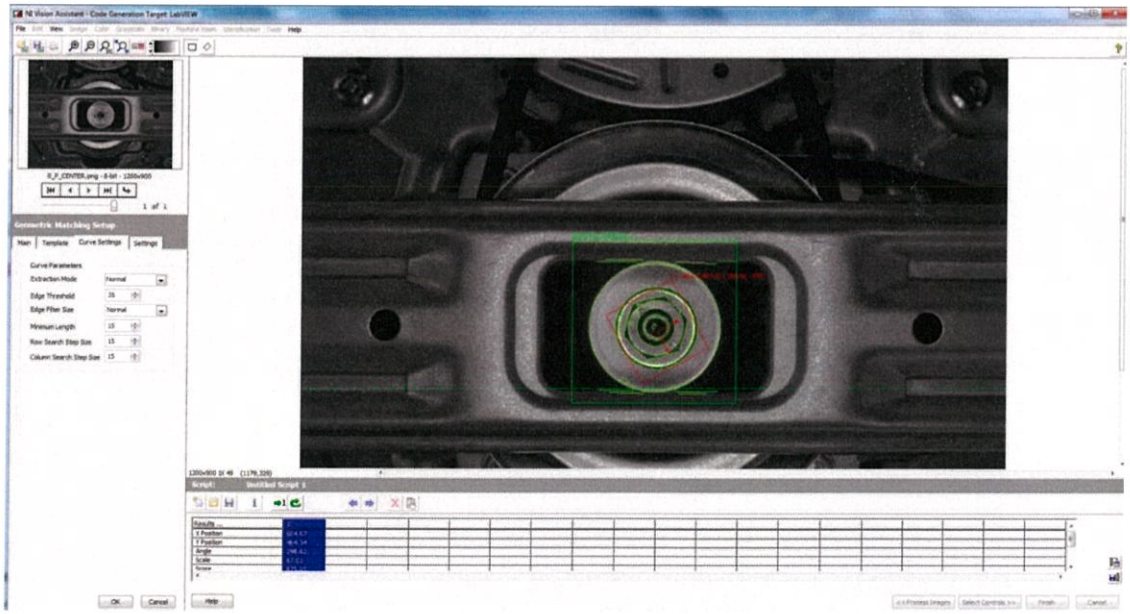
Pattern

1. เลือก Vision Assistance ในชนิดน็อตที่ต้องการแก้



2. สร้าง New Template บริเวณจุดที่สนใจ และปรับค่า Edge Threshold ให้เหมาะสม

3. ปรับค่า Edge Threshold, Minimum Length, Row Search Step Size, Column Search Step Size ให้เหมาะสม



ประวัติผู้เขียน

ชื่อ-นามสกุล	นายธนกฤต โพธิ์ศรีนาค
วัน เดือน ปีเกิด	5 เมษายน พุทธศักราช 2539
ที่อยู่ปัจจุบัน	36/1 หมู่ที่ 7 ต.ม่วงเตี้ย อ.วิเศษชัยชาญ จ.อ่างทอง 14110
เบอร์โทรศัพท์	092-9328922
E-mail	thanakrit.ph@outlook.com

ประวัติการศึกษา

พุทธศักราช 2546-2551	สำเร็จการศึกษาระดับชั้นประถมศึกษา จาก โรงเรียนอนุบาลวัดนางใน อ่างทอง
พุทธศักราช 2552-2554	สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้น สายการเรียนคณิตศาสตร์ วิทยาศาสตร์ ห้องเรียนพิเศษ จาก โรงเรียนอ่างทองปัทมโรจน์วิทยาคม อ่างทอง
พุทธศักราช 2555-2557	สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลาย สายการเรียนพิเศษวิทยาศาสตร์ พสวท. จาก โรงเรียนอ่างทองปัทมโรจน์วิทยาคม อ่างทอง
พุทธศักราช 2558-2561	ศึกษาระดับอุดมศึกษา หลักสูตรวิศวกรรมแมคคาทรอนิกส์ ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพมหานคร

ประวัติการทำงาน

พุทธศักราช 2560	ฝึกงานบริษัท ไทยซัมซุง อิเล็กทรอนิกส์ จำกัด แผนก Factory Innovation Technology (FIT)