เครื่องติดแผ่นกันกระแทกอัตโนมัติ

Automatic Sponge Machines

สรกฤษณ์ วิทยานันท์

รายงานสหกิจศึกษานี้เป็นส่วนหนึ่งในหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบการผลิต
วิทยาลัยนวัตกรรมการผลิตขั้นสูง
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องติดแผ่นกันกระแทกอัตโนมัติ

Automatic Sponge Machines

T147158

สรกฤษณ์ วิทยานันท์

สบหมู่.....
เลขทะเบียน 147158
วัน.เดือน.ปี 13 ก.ค. 2560

b 10284977
i.....

รายงานสหกิจศึกษานี้เป็นส่วนหนึ่งในหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต
สาขาวิชาวิศวกรรมระบบการผลิต
วิทยาลัยนวัตกรรมการจัดการข้อมูล
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Automatic Sponge Machines

SORAKRIT WITTHAYANAN

AN INTERNSHIP REPORT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN MANUFACTURING SYSTEM ENGINEERING

COLLEGE OF DATA STORAGE INNOVATION

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2015

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา	เครื่องติดแผ่นกันกระแทกอัตโนมัติ Auto Sponge
นักศึกษา	นายสรกฤษณ์ วิทยานันท์
รหัสนักศึกษา	55120036
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมระบบการผลิต
พ.ศ.	2558
อาจารย์ที่ปรึกษา	ดร. ฉัตรพล ภคศิริ

วิทยาลัยนวัตกรรมการจัดการข้อมูล สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้สหกิจศึกษาเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต ประจำปีการศึกษา

คณะกรรมการสอบ	ลายมือชื่อ
ดร. ฉัตรพล ภคศิริ	
ดร. อนรรฆพล แสนทน	
นาย มนัส สังข์เปีย	
นาย ภาษิต ธาตุสุวรรณ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา	เครื่องติดแผ่นกันกระแทกอัตโนมัติ
นักศึกษา	นายสรกฤษณ์ วิทยานันท์
รหัสนักศึกษา	55120036
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมระบบการผลิต
พ.ศ.	2558
อาจารย์นิเทศ	ดร. ฉัตรพล ภคศิริ
ผู้นิเทศงาน	นาย มนัส สังข์เปีย
ชื่อสถานประกอบการ	บริษัท แคล-คอมพ์ อิเล็คโทรนิกส์ (ประเทศไทย) จำกัด (มหาชน)

บทคัดย่อ

ปัจุบันบริษัทอุตสาหกรรมต่างๆมีการแข่งขันกันสูงในด้านการผลิตจึงต้องมีการพัฒนา สายการ ผลิตให้มีประสิทธิภาพโดยการเปลี่ยนระบบการผลิตให้เป็นระบบอัตโนมัติที่ช่วยในการลดต้นทุนและเพิ่มความสามารถในการผลิตมากยิ่งขึ้น ดังนั้นในรายงานนี้ได้มีจุดประสงค์ในการที่จะพัฒนาสายการผลิตการประกอบเอคเทอร์นอลฮาร์ดดิสก์(External Hardisk) ในขั้นตอนการติดแผ่นกันกระแทกบริเวณฝาเอคเทอร์นอลฮาร์ดดิสก์ที่ยังมีการติดโดยมนุษย์ จึงได้มีการประดิษฐ์เครื่องติดแผ่นกันกระแทกอัตโนมัติที่สามารถติดแผ่นกันกระแทกจากม้วนแผ่นกันกระแทกทั้งหมดได้ จึงได้มีการนาเสตปมอเตอร์(Step Motor) มาช่วยในการขับเคลื่อนแผ่นกันกระแทกให้ไปยังตาแหน่งของแผ่นกันกระแทกทั้งหมด 9 จุด โดยการทางานนั้นต้องอาศัยหลักการทางวิศวกรรม เช่น พีแอลซี(PLC), โปรแกรม Visual Basic C#, ระบบนิวเมติกส์(Pneumatic), การสื่อสารระหว่างพีแอลซีกับคอมพิวเตอร์, การสื่อสารระหว่างเสตปมอเตอร์กับคอมพิวเตอร์ เป็นต้น จากการปฏิบัติงานนั้นเครื่องติดแผ่นกันกระแทกสามารถปฏิบัติงานแทนมนุษย์ได้ซึ่งช่วยในการลดต้นทุนให้กับบริษัทและช่วยเพิ่มประสิทธิภาพให้กับระบบการผลิตมากยิ่งขึ้น

คำสำคัญ เครื่องจักรอัตโนมัติ พีแอลซี บรรจุภัณฑ์ เอกซ์เทอร์นอลฮาร์ดดิสก์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Automatic Sponge Machines
Student	Mr. Sorakrit Witthayanan
Student ID	55120036
Degree	Bachelor of Engineering
Program	Manufacturing System Engineering
Year	2015
Advisor	Dr. Chatrpol Pakasiri
Mentor	Mr. Manus Sangpia
Company	Cal-Comp Electronics (Thailand) Public Company Limited

Abstract

Currently companies across various industries are highly competitive in terms of production. Need to develop an efficient production line. By changing the system to produce a system that helps to reduce costs and increase the ability to produce even more. So, in this report, the purpose of which is to develop the production line assembly A. Terry methanol hard drive (External Harddisk) in the process of cushioning discs lid Eck Terry methanol hard disk also caught by humans. They have invented a machine that can automatically attach damper plate from coil cushioning all. I have adopted stage motor (Step Motor) helped to propel damper to the position of cushioning discs all nine points. The work that relies on engineering principles such as PLC (PLC), program. Visual Basic C #, systems, pneumatic Inc. (Pneumatic), communication between PLC and PC. , Communication between stage motor with a computer, from the practical work and stick cushioning discs can perform the work of man, which helps to reduce costs for companies and improves the efficiency of the production system.

KEYWORD Internship Automatic Sponge Machines External Harddisk PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การที่ข้าพเจ้าได้มาปฏิบัติงานสหกิจศึกษา ณ บริษัท แคล-คอมพ์ อิเล็คโทรนิกส์ (ประเทศไทย) จำกัด (มหาชน) ตั้งแต่วันที่ 20 กรกฎาคม 2558 ถึงวันที่ 27 พฤศจิกายน 2558 ส่งผลให้ข้าพเจ้าได้รับความรู้และประสบการณ์ต่างๆ ที่มีค่ามากมายสำหรับรายงานวิชาสหกิจศึกษาฉบับนี้สำเร็จลงได้ด้วยดีจากความร่วมมือและสนับสนุนจากหลายฝ่าย ดังนี้

1. คุณคงสิทธิ์ โจวกิจเจริญ (Managing Director) บริษัท แคล-คอมพ์ อิเล็คโทรนิกส์ (ประเทศไทย) จำกัด (มหาชน) ที่เห็นความสำคัญของระบบศึกษาแบบสหกิจศึกษาและได้ให้โอกาสที่มีคุณค่าอย่างยิ่งต่อข้าพเจ้า
2. คุณมนัส สังข์เปีย (Electronic Engineer) ซึ่งเป็น Mentor
3. คุณภาษิต ธาตุสุวรรณ (Electronic Engineer) ซึ่งเป็น Job supervisor
4. คุณนัยวัฒน์ เรือนก้อน (Electronic Engineer)
5. คุณจักรพงษ์ เจียรภัทรานนท์ (Electronic Engineer)
6. คุณวัชระ ใจอิ่ม (Process Engineer)

และบุคลากรท่านอื่นๆ ที่ไม่ได้กล่าวนามทุกท่านที่ได้ให้คำแนะนำและช่วยเหลือ

ข้าพเจ้าใคร่ขอขอบพระคุณผู้มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลเป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ตลอดจนให้การดูแลและให้ความเข้าใจเกี่ยวกับชีวิตของการทำงานจริงของข้าพเจ้า ขอขอบพระคุณไว้ ณ ที่นี้

นายสรกฤษณ์ วิทยานันท์

ผู้จัดทำรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ

ภาษาไทย..... I

บทคัดย่อ

ภาษาอังกฤษ..... II

กิตติกรรมประกาศ..... III

สารบัญ..... IV

สารบัญภาพ..... VII

สารบัญตาราง..... XI

บทที่ 1 บทนำ..... 1

1.1 ความเป็นมาและความสำคัญของปัญหา..... 1

1.2 วัตถุประสงค์ของการศึกษา..... 2

1.3 ขอบเขตของการศึกษา..... 2

1.4 ขั้นตอนของการศึกษา..... 2

1.5 ประวัติของสถานประกอบการ..... 2

บทที่ 2 ทฤษฎีและหลักการ..... 4

2.1 พีแอลซี (Programmable Logic Controller หรือ PLC) 4

- ชนิดของพีแอลซี..... 4
- ภาษาที่ใช้ในการเขียนโปรแกรมให้กับพีแอลซี..... 6
- อุปกรณ์สำหรับการโปรแกรม..... 8
- โครงสร้างของพีแอลซี..... 10
- หลักการเขียนแลดเดอร์ไดอะแกรมและคำสั่งพื้นฐาน..... 13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

- ความรู้พื้นฐานทางด้านดิจิตอล..... 25
- ASCII..... 27

2.2 เซ็นเซอร์และอุปกรณ์..... 28

- Photoelectric Sensor..... 28
- Proximity Sensor..... 30
- รีเลย์..... 31
- สวิตชิ่งเพาเวอร์ซัพพลาย..... 32
- โซลินอยด์วาล์ว..... 34
- อุปกรณ์สำหรับงานสูญญากาศ..... 37
- เซ็นเซอร์วัดความดัน..... 38
- เสต็ปมอเตอร์..... 38
- การใช้งานพอร์ตอนุกรม RS232..... 42

บทที่ 3 วิธีการดำเนินงาน..... 48

3.1 แนวทางการพัฒนาสายการผลิต..... 48

3.2 แผนผังกระบวนการทำงาน..... 48

3.3 การวาดเลเอาท์..... 50

3.4 การกำหนดเอาท์พุทและอินพุท..... 51

3.5 อุปกรณ์และการเดินสาย..... 53

3.6 ลักษณะการทำงานของเครื่อง Auto Sponge..... 57

3.7 การควบคุมเสต็ปมอเตอร์..... 61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 4 ผลการวิจัย 69

4.1 ผลการดำเนินงาน Auto Sponge ในเวอร์ชั่นต่างๆ..... 69

4.2 ประสิทธิผลโดยรวมของเครื่องจักร..... 75

บทที่ 5 วิเคราะห์ สรุป และข้อเสนอแนะ 80

5.1 วิเคราะห์..... 80

5.2 สรุป..... 82

5.3 ข้อเสนอแนะ..... 82

เอกสารอ้างอิง..... 83

ภาคผนวก..... 84

ประวัติผู้เขียน..... 93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้า

รูปที่ 2.1 พีแอลซี แบบบ Block..... 4
รูปที่ 2.2 พีแอลซี แบบบโมดูล..... 5
รูปที่ 2.3 การเขียนโปรแกรมโดยตัวป้อนโปรแกรมแบบมือถือ..... 8
รูปที่ 2.4 ภาพโปรแกรม KV STUDIO Ver.7..... 8
รูปที่ 2.5 โครงสร้างของ พีแอลซี..... 9
รูปที่ 2.6 ภาคอินพุต..... 10
รูปที่ 2.7 ภาคอินพุต..... 11
รูปที่ 2.8 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด..... 14
รูปที่ 2.9 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก..... 14
รูปที่ 2.10 แลดเดอร์ไดอะแกรม (Ladder Diagram) A..... 14
รูปที่ 2.11 แลดเดอร์ไดอะแกรม (Ladder Diagram) B..... 15
รูปที่ 2.12 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด..... 17
รูปที่ 2.13 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก..... 17
รูปที่ 2.14 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด..... 17
รูปที่ 2.15 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก..... 17
รูปที่ 2.16 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด..... 17
รูปที่ 2.17 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก..... 18
รูปที่ 2.18 การเขียนโปรแกรมแบบขนาน..... 18
รูปที่ 2.19 ตารางเลขฐานต่างๆ..... 26
รูปที่ 2.20 ตัวอย่างรหัส ASCII แปลเป็นเลขฐานอื่นๆ..... 27
รูปที่ 2.21 รูปตาราง ASCII แปลงเป็นเลขฐานอื่นๆ..... 27
รูปที่ 2.22 หลักการทำงานของ Photoelectric Sensor..... 27
รูปที่ 2.23 หลักการทำงานของ Retroreflective Mode..... 28
รูปที่ 2.24 หลักการทำงานของ Opposed Mode..... 28
รูปที่ 2.25 รูปร่างของเซนเซอร์แบบเหนี่ยวนำ..... 29
รูปที่ 2.26 รูปร่างเซนเซอร์ชนิดเก็บประจุ..... 30
รูปที่ 2.27 หลักการทำงานของรีเลย์..... 30
รูปที่ 2.28 โครงสร้างของ Switching Power Supply..... 32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.29 Switching Power Supply..... 32
รูปที่ 2.30 วาล์วควบคุม3 ทิศทาง2ตำแหน่ง ปกติปิด..... 33
รูปที่ 2.31 วาล์วควบคุม3 ทิศทาง2ตำแหน่ง ปกติปิด..... 33
รูปที่ 2.32 วาล์วควบคุม 5 ทิศทาง 2 ตำแหน่ง..... 34
รูป 2.33 วาล์วควบคุม 5 ทิศทาง 2 ตำแหน่ง..... 34
รูป 2.34 ตัวอย่างโซลินอยด์วาวล์..... 35
รูปที่ 2.35 หลักการทำงานตัวกำเนิดสูญญากาศ..... 35
รูปที่ 2.36 ตัวอย่างตัวกำเนิดสูญญากาศ..... 36
รูปที่ 2.37 ตัวอย่างเซ็นเซอร์วัดความดัน..... 36
รูปที่ 2.38 โครงสร้างของเสตปมอเตอร์..... 37
รูปที่ 2.39 หลักการทำงานของเสต็ปมอเตอร์..... 38
รูปที่ 2.40 ระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL..... 42
รูปที่ 2.41 ระดับสัญญาณของการสื่อสารแบบซิงโครนัส..... 43
รูปที่ 2.42 ระดับสัญญาณของการสื่อสารแบบอะซิงโครนัส..... 43
รูปที่ 3.1 แผนผังการทำงานระบบ PLC ของ Auto Sponge..... 46
รูปที่ 3.2 แผนผังการทำงานระบบ Stepping Motor ของ Auto Sponge..... 46
รูปที่ 3.3 เลเอาท์ของเครื่อง Auto Sponge..... 47
รูปที่ 3.4 การกำหนดอินพุทใน PLC..... 48
รูปที่ 3.5 การกำหนด Output ใน PLC..... 49
รูปที่ 3.6 การจัดวางตำแหน่งเซ็นเซอร์..... 53
รูปที่ 3.7 สถานะเริ่มต้นการทำงานของเครื่อง Auto Sponge..... 53
รูปที่ 3.8 ภาพจำลองการทำงานของเครื่องเมื่อเริ่มกดปุ่ม Start/Stop..... 54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.9 ภาพจำลองการทำงานของเครื่อง Auto Sponge..... 55

รูปที่ 3.10 ภาพจำลองการทำงานของเครื่อง Auto Sponge..... 56

รูปที่ 3.11 ภาพจำลองการทำงานของเครื่อง Auto Sponge..... 57

รูปที่ 3.12 การส่งผ่านข้อมูลของเครื่อง Auto Sponge Version 1.0..... 58

รูปที่ 3.13 โปรแกรมควบคุมการสื่อสารระหว่าง PLC และ

มอเตอร์สเตป ของเครื่อง Auto Sponge V 1.0..... 58

รูปที่ 3.14 การส่งผ่านข้อมูลข้อเครื่อง Auto Sponge Version 2.0..... 60

รูปที่ 3.15 เสปค I/O ของมอเตอร์สเตป..... 60

รูปที่ 3.16 I/O ของมอเตอร์เสตป..... 61

รูปที่ 3.17 การเลือกตำแหน่งที่ต้องการโดยการต่อ I/O..... 62

รูปที่ 3.18 ภาพโปรแกรม RS-Manager..... 62

รูปที่ 3.19 ภาพ I/O ของมอเตอร์สเตป..... 64

รูปที่ 4.1 Auto Sponge V.1.0..... 65

รูปที่ 4.2 แสดงให้เห็นไฟสีน้ำเงินของมอเตอร์เสตปว่าสามารถทำงานได้อย่างปกติ..... 66

รูปที่ 4.3 เครื่อง Auto Sponge V.1.0 ทำงานปกติที่ตำแหน่งต่างๆ..... 66

รูปที่ 4.4 โปรแกรมได้รับค่า ERROR จากมอเตอร์เสตป 67

รูปที่ 4.5 มอเตอร์สเตปมี Error สัญญาณไฟจะเป็นสีแดงและวิ่งกลับไปยังจุดเริ่มต้น..... 67

รูปที่ 4.6 การวัดสัญญาณมอเตอร์เสตปด้วยเครื่องสโคป..... 68

รูปที่ 4.7 ภาพจากเครื่องวัดสโคปในการวัดขณะที่มอเตอร์เสตปทำงานปกติ..... 68

รูปที่ 4.8 ภาพจากเครื่องวัดสโคปในการวัดขณะที่มอเตอร์เสตปทำงานผิดปกติ..... 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.9 การทดสอบเครื่อง Auto Sponge V.2.0 ก่อนนำเข้าสายการผลิตจริง..... 70

รูปที่ 4.10 การทดสอบเครื่อง Auto Sponge V.2.0 ก่อนนำเข้าสายการผลิตจริง..... 70

รูปที่ 4.11 ชิ้นงานที่ทำงานโดยเครื่อง Auto Sponge V.2.0..... 70

รูปที่ 4.12 การทำงานของกระบอกสูบที่ช่วยแก้ปัญหาเรื่องเซ็นเซอร์
แอมพลิฟายเออร์ตรวจจับไม่ได้..... 73

รูปที่ 4.13 การปรับโครงสร้างของหัวดูดกับเซ็นเซอร์แอมพลิฟลายเออร์ให้ตรงกันและ
รับความเร็วมอเตอร์ของการ Feed..... 73

รูปที่ 4.14 ทำการติดสักราชลงบนที่หนีบและแก้ไขโปรแกรม..... 74

รูปที่ 5.1 .ใช้มนุษย์ติดแผ่นกันกระแทก..... 76

รูปที่ 5.2 ใชเครื่อง Auto Sponge V.2.0 ติดแผ่นกันกระแทก..... 77

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 ข้อดีและข้อเสีย ของ พีแอลซี แบบ Block Type..... 4

ตารางที่ 2.2 ข้อดีและข้อเสียของ พีแอลซี ชนิดโมดูล..... 5

ตารางที่ 2.3 ลำดับการทำงานเสต็ปมอเตอร์แบบเวฟ..... 38

ตารางที่ 2.4 ลำดับการทำงานของเสต็ปมอเตอร์แบบ 2 เฟส..... 38

ตารางที่ 2.5 ลำดับการทำงานของเสต็ปมอเตอร์แบบครึ่งเสต็ป..... 39

ตารางที่ 2.6 I/O ของ RS232..... 41

ตารางที่ 3.1 อุปกรณ์ดำเนินงาน..... 49

ตารางที่ 4.1 การบันทึกผลการทำงานของ Auto Sponge V.2.0..... 72

ตารางที่ 4.2 การบันทึกผลการทำงานของ Auto Sponge V.2.0..... 74

ตารางที่ 5.1 Cycle Time ของมนุษย์ในการติดแผ่นกันกระแทก..... 77

ตารางที่ 5.2 Cycle Time ของเครื่อง Auto Sponge V.2.0..... 77

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันระบบควบคุมอัตโนมัติ(Automation) ทางด้านอุตสาหกรรม เข้ามามีบทบาทเป็นอย่างมากในการพัฒนาวิธีการผลิตที่ช่วยลดต้นทุน สามารถควบคุมคุณภาพของการผลิตได้ดีขึ้น เพิ่มกำลังการผลิต และทำให้การผลิตมีความน่าเชื่อถือสูง ปัจจุบันทุกบริษัทจึงมุ่งเน้นการพฒนาระบบสายงานการผลิตไปที่ระบบการผลิตแบบใช้เครื่องจักรอัตโนมัติ(Automation Machine)

ดังนั้นการผลิตในปัจจุบันนั้นต้องการความรวดเร็ว แม่นยำและการลดต้นทุนจึงจะเป็นการผลิตที่ดี ดังนั้นจากการที่ได้สังเกตุเห็นว่าสายการผลิตการประกอบเอคทานอลนั้นสามารถปรับปรุงและเพิ่มประสิทธิภาพได้ก็คือขั้นตอนในการติดแผ่นกันกระแทกที่จำเป็นต้องใช้มนุษย์ 1 คนในการปฏิบัติงาน ซึ่งประสบปัญหาดังนี้

1. มนุษย์ไม่สามารถทำงานได้ตลอดเวลาต้องมีการหยุดพัก
2. มนุษย์มีความแม่นยำในการทำงานต่ำเนื่องจากต้องแผ่นกันกระแทกมีขนาดเล็ก
3. เนื่องจากต้องใช้สายตาในการปฏิบัติงานอย่างสูงจึงอาจเป็นอันตรายต่อสายตามนุษย์ในอนาคตได้
4. มนุษย์มีค่าใช้จ่ายต่อเดือนจึงทำให้มีต้นทุนในการผลิตสูง

ซึ่งแนวทางการแก้ไขปัญหาคือการประดิษฐ์เครื่องติดแผ่นกันกระแทกอัตโนมัติโดยที่สามารถปฏิบัติงานแทนมนุษย์ได้

จากการศึกษาค้นคว้าเพื่อที่จะพัฒนาสายการผลิต จึงได้พบกับปัญหาต่างๆ จากนั้นจะนำเอาปัญหาที่พบมาเสนอให้กับทีมงานที่แผนก CCET Automation และได้ทำการออกแบบเครื่องเพื่อที่จะมาแก้ไขและปรับปรุงพัฒนาสายการผลิต โดยแบ่งการทำงานออกเป็น 2 กลุ่ม ดังนี้

กลุ่มที่ 1 (Team Mechanics) : ทำการออกแบบโครงสร้างภายนอกของเครื่อง ซึ่งชิ้นส่วนหลักของกลไกการทำงาน ทำจากเหล็กกล้า อะลูมิเนียมและ เหล็กกล้าไร้สนิม

กลุ่มที่ 2 (Team Electronics) : ทำการออกแบบวงจรทางไฟฟ้า และเขียนโปรแกรมการทำงานของเครื่องอัตโนมัติ ซึ่งส่วนใหญ่จะใช้ PLC ในการเขียนโปรแกรม (ได้รับมอบหมายให้ทำงานในกลุ่มที่ 2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของการศึกษา

1.2.1 เพื่อการพัฒนาสายการผลิต

1.2.2 เพื่อลดต้นทุนในการผลิต

1.2.3 เพื่อศึกษาระบบการทำงานของพีแอลซี

1.2.4 เพื่อศึกษาโปรแกรม Visual Basic C#

1.2.5 เพื่อศึกษาวงจรไฟฟ้าและเสตปมอเตอร์

1.2.6 เพื่อได้เรียนรู้ถึงสภาพปัญหา และวิธีการแก้ไขปัญหาที่เกิดขึ้นขณะปฏิบัติงาน

1.2.7 เพื่อนำทฤษฎีที่ศึกษามา นามาใช้กับงานจริง

1.3 ขอบเขตของการศึกษา

1.3.1 ลักษะณะของแผ่นกันกระแทก

1.3.2 การสื่อสารระหว่างเสตปมอเตอร์และคอมพิวเตอร์

1.3.3 การสื่อสารระหว่างพีแอลซีและคอมพิวเตอร์

1.3.4 ประสิทธิภาพของเครื่องจักร

1.4 ขั้นตอนของการศึกษา

1.4.1 ศึกษาทฤษฎี

1.4.2 วาดเลเอาท์และติดอุปกรณ์

1.4.3 เขียนลักษณะการทำงานของเครื่องจักร

1.4.4 ทำการเดินสายไฟและสายลม

1.4.5 เขียนโปรแกรมพีแอลซี

1.4.6 เขียนโปรแกรม Visual Basic C#

1.4.7 ทดสอบและวิเคราะห์ผลการทำงาน

1.4.8 สรุปผล

1.5 ประวัติของสถานประกอบการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5.1 รายละเอียดเกี่ยวกับบริษัท

แคล-คอมพ์ อิเล็คโทรนิกส์ (ประเทศไทย) จำกัด (มหาชน) บริษัท แคล-คอมพ์ อีเล็คโทรนิกส์ (ประเทศไทย) จำกัด มหาชน ได้ประกอบธุรกิจการผลิตและจัดจำหน่ายผลิตภัณฑ์อีเล็คโทรนิกส์ ในรูปแบบ Electronic Manufacturing Services (EMS) โดยได้แบ่งการผลิตออกเป็น 2 รูปแบบ คือการผลิตภายใต้เครื่องหมายการค้าของลูกค้าในรูปแบบของ Original Equipment Manufacturing (OEM) ซึ่งผลิตภัฑ์ต่างๆส่วนใหญ่จะได้รับการวิจัยและออกแบบจากลูกค้า โดยลูกค้าจะเป็นผู้ส่งสินค้าต้นแบบให้กับบริษัทเพื่อทำการผลิตให้กับลูกค้า และการผลิตภายใต้เครื่องหมายการค้าของลูกค้าในรูปแบบของ Original Design Manufacturing (ODM) ซึ่งผลิตภัฑ์ที่ผลิตให้กับลูกค้านั้นจะเป็นการวิจัยและออกแบบเองโดยบริษัท นอกจากนี้บริษัทยังเน้นการให้บริการแบบครบวงจรเพื่ออำนวยความสะดวกแก่ลูกค้า

1.5.2 เป้าหมายของบริษัท

1.5.2.1 ผลิตสินค้าคุณภาพทัดเทียมกับต่างประเทศ

1.5.2.2 สนองความต้องการของตลาดในประเทศและการส่งออก

1.5.2.3 ใช้วัตถุดิบในประเทศ

1.5.2.4 ส่งเสริมให้คนไทยมีงานทำ

1.5.2.5 สร้างความเจริญก้าวหน้าทางเศรษฐกิจ และสังคมของประเทศ

1.5.3 นโยบายของบริษัท

1.5.3.1 ใช้แรงงานท้องถิ่น

1.5.3.2 ให้ค่าตอบแทนที่เหมาะสม

1.5.3.3 ส่งเสริมให้พนักงานมีความก้าวหน้า

1.5.3.4 จัดสวัสดิการตามความเหมาะสม

1.5.3.5 พนักงานมีส่วนในความคิดริเริ่ม

1.5.3.6 สร้างความสัมพันธ์ระหว่างผู้บริหาร และพนักงาน

1.5.3.7 สนับสนุนให้เกิดความเป็นธรรมแก่ทุกฝ่าย

1.5.4 อุดมการณ์ของบริษัท

สามารถก้าวขึ้นเป็นผู้นำในธุรกิจ Electronics Manufacturing Service (EMS) อีกทั้งขยายฐานธุรกิจเข้าสู่อุตสาหกรรมแบบ Original Design Manufacturing (ODM) เพื่อเพิ่มคุณภาพสินค้าและสร้างความยั่งยืนในอนาคตให้กับธุรกิจได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 พีแอลซี (Programmable Logic Controller หรือ PLC) [1]

เป็นอุปกรณ์ที่คิดค้นขึ้นมาเพื่อใช้ควบคุมการทำงานของเครื่องจักรหรือระบบต่างๆแทนวงจรรีเลย์แบบเก่า ซึ่งวงจรรีเลย์มีข้อเสียคือ การเดินสายและการเปลี่ยนแปลงเงื่อนไขในการควบคุมมีความยุ่งยาก และเมื่อใช้งานไปนานๆ หน้าสัมผัสของรีเลย์จะเสื่อม ทำให้ขาดเสถียรภาพในการควบคุม ดังนั้นปัจจุบัน พีแอลซี จึงเข้ามาทดแทนวงจรรีเลย์ เพราะพีแอลซีใช้งานได้ง่ายกว่าสามารถต่อเข้ากับอุปกรณ์อินพุต/เอาต์พุตได้โดยตรง นอกจากนั้นเพียงแค่เขียนโปรแกรมควบคุมก็สามารถใช้งานได้ทันที ถ้าต้องการจะเปลี่ยนเงื่อนไขใหม่สามารถทำได้โดยการเปลี่ยนแปลงโปรแกรมเท่านั้น นอกจากนี้ พีแอลซี ยังสามารถใช้งานร่วมกับอุปกรณ์อื่นๆ เช่น เครื่องอ่านบาร์โค้ด, เครื่องพิมพ์ (Printer) และระบบ RFID เป็นต้น

ในปัจจุบันนอกจาก พีแอลซี จะใช้งานแบบเดี่ยว (Stand Alone) แล้ว ยังสามารถต่อ พีแอลซี หลายๆ ตัวเข้าด้วยกันเป็นเครือข่าย (Network) เพื่อควบคุมการทำงานของระบบให้มีประสิทธิภาพมากยิ่งขึ้นอีกด้วย จะเห็นได้ว่าการใช้งาน พีแอลซี มีความยืดหยุ่นมากกว่าการใช้งานวงจรรีเลย์แบบเก่าดังนั้นในปัจจุบันโรงงานอุตสาหกรรมต่างๆ จึงใช้ พีแอลซี เป็นหัวใจหลักในการควบคุมการทำงานของเครื่องจักรเราสามารถจำแนกประเภทของ พีแอลซี ตามลักษณะภายนอกได้เป็น 2 ชนิด คือ

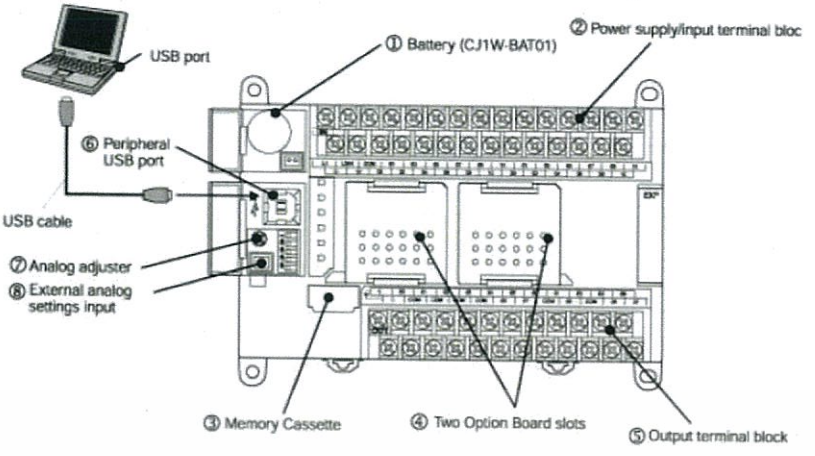
2.1.1 ชนิดของ พีแอลซี [2]

สามารถจำแนก พีแอลซี ตามโครงสร้างหรือลักษณะภายนอกได้เป็น 2 ชนิด คือ

2.1.1.1 พีแอลซี ชนิดบล็อก (Block Type พีแอลซี)

พีแอลซี ประเภทนี้ จะรวมส่วนประกอบทั้งหมดของ พีแอลซี อยู่ในบล็อกเดียวกัน ไม่ว่าจะเป็นตัวประมวลผล หน่วยความจำ ภาคอินพุต/เอาต์พุต และแหล่งจ่ายไฟ สามารถแสดงตัวอย่าง พีแอลซี แบบ Block Type ให้เห็นดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 พีแอลซี แบบบ Block

ตารางที่ 2.1 ข้อดีและข้อเสีย ของ พีแอลซี แบบ Block Type

ข้อดี	ข้อเสีย
1.มีขนาดเล็กสามารถติดตั้งได้ง่ายจึงเหมาะกับงาน ควบคุมขนาดเล็กๆ	1.การเพิ่มจำนวนอินพุต/เอาต์พุตสามารถเพิ่มได้น้อยกว่า พีแอลซี ชนิดโมดูล
2. สามารถใช้งานแทนวงจรรีเลย์ได้	2.เมื่ออินพุต/เอาต์พุตเสียจุดใดจุดหนึ่งต้องนำ พีแอลซี ออกไปทั้งชุดทำให้ระบบต้องหยุดทำงานชั่วระยะเวลาหนึ่ง
3. มีฟังก์ชันพิเศษ เช่นฟังก์ชันทางคณิตศาสตร์และฟังก์ชันอื่นๆ	3. มีฟังก์ชันให้เลือกใช้งานน้อยกว่า พีแอลซี ชนิดโมดูล
4.มีราคาถูกกว่าแบบแร็คหรือโมดูลในจำนวนอินพุต/เอาต์พุตที่เท่ากัน	

2.1.1.2 พีแอลซี ชนิดโมดูล (Modular Type พีแอลซี) หรือแร็ค (Rack Type พีแอลซี) พีแอลซี ชนิดนี้ส่วนประกอบแต่ละส่วนสามารถแยกออกจากกันเป็นโมดูล (Modules) เช่นภาคอินพุต/เอาต์พุต จะอยู่ในส่วนของโมดูลอินพุต/เอาต์พุต (Input / Output Units) ซึ่งสามารถเลือกใช้งานได้ว่าจะใช้โมดูลขนาดกี่อินพุต/เอาต์พุต ซึ่งมีให้เลือกใช้งานหลายรูปแบบอาจจะใช้เป็นอินพุตอย่างเดียวขนาด 8 /16 จุด หรือเป็นเอาต์พุตอย่างเดียวขนาด 4/8/12/16 จุด ขึ้นอยู่กับรุ่นของพีแอลซี ด้วยในส่วนของตัวประมวลผลและหน่วยความจำจะรวมอยู่ในซีพียูโมดูล (CPU Unit) สามารถเปลี่ยนขนาดของ CPU Unit ให้เหมาะสมตามความต้องการใช้งาน เช่น พีแอลซี รุ่น CS1 จะมี CPU ให้เลือกใช้งานหลายรุ่นเช่น รุ่น CS1G-CPU42H จะมีความแตกต่างกับ พีแอลซี รุ่น CS1HCPU65H (ทั้งสองรุ่นเป็น พีแอลซี ตระกูล CS1 เหมือนกัน) ตรงขนาดความจุของโปรแกรมและการรองรับจำนวนอินพุต/เอาต์พุต เป็นต้นส่วนประกอบต่างๆ ของ พีแอลซี ชนิดโมดูลที่กล่าวมาทั้งหมดนั้น เมื่อต้องการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะถูกนำมาต่อร่วมกัน บางรุ่นใช้เป็นคอนเนคเตอร์ในการเชื่อมต่อกันระหว่างยูนิต เช่น รุ่น CQM1 / CQM1H หรือ CJ1M/H/G แต่บางรุ่นใช้ Backplaneในการรวมยูนิตต่างๆ เข้าด้วยกัน เพื่อให้สามารถใช้งานร่วมกันได้สามารถยกตัวอย่าง พีแอลซี ชนิดโมดูลได้ดังแสดงรูปที่ 2.2

รูปที่ 2.2 พีแอลซี แบบบโมดูล

ตารางที่ 2.2 ข้อดีและข้อเสียของ พีแอลซี ชนิดโมดูล

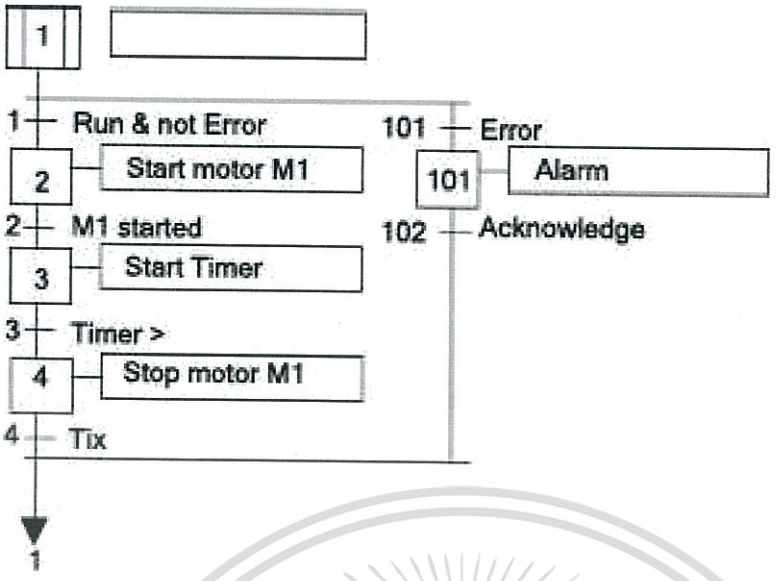
ข้อดี	ข้อเสีย
1. เพิ่มขยายระบบได้ง่ายเพียงแค่ติดตั้งโมดูลต่างๆ ที่ต้องการใช้งานลงไปบน Back plane	1. ราคาแพงเมื่อเทียบกับ พีแอลซี แบบ Block Type ที่มีจำนวนI/O เท่ากัน
2.สามารถขยายจำนวนอินพุต/เอาต์พุตได้มากกว่าแบบ Block Type	
3.อุปกรณ์อินพุต/เอาต์พุตเสียจุดใดจุดหนึ่งสามารถถอดเฉพาะโมดูลนั้นไปซ่อมทำให้ระบบสามารถทำการต่อได้	
4.มียูนิตและรูปแบบการติดต่อสื่อสารให้เลือกใช้งานมากกว่าแบบ Block Type	

2.1.2 ภาษาที่ใช้ในการเขียนโปรแกรมให้กับ พีแอลซี

พีแอลซี แต่ละยี่ห้อจะใช้ภาษาในการเขียนโปรแกรมเพื่อสั่งให้ พีแอลซี ทำงานตามความต้องการแตกต่างกัน ซึ่งตามมาตรฐาน IEC1131-3 ได้แบ่งมาตรฐานภาษาต่างๆ ออกเป็น 5 แบบตามรูปที่แสดงข้างล่างนี้ ภาษาที่นิยมใช้มากที่สุดคือ Ladder Diagram เพราะเป็นภาษาที่ง่ายมีลักษณะคล้ายวงจรควบคุมแบบรีเลย์

1.) Sequential Flow Chart Language

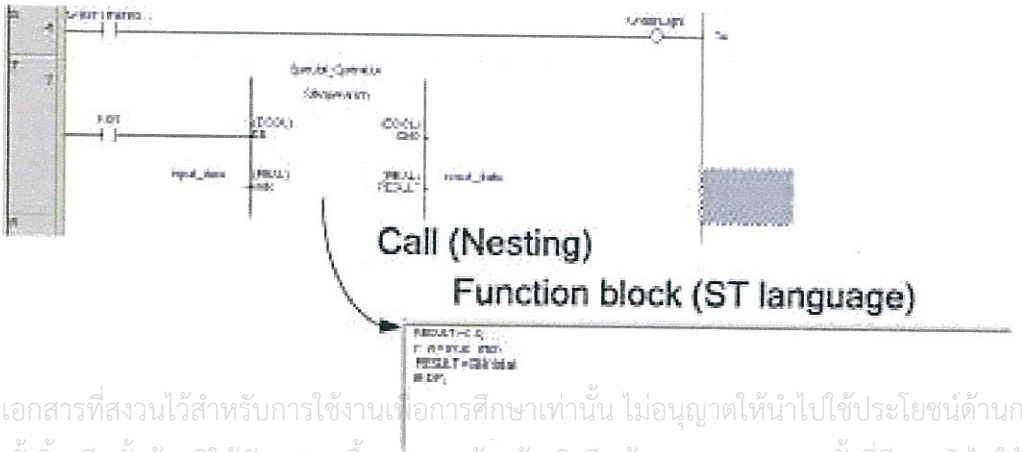
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



2.) Structure Text Language

```
D: = B*B - 4*A*C;
IF D < 0.0 THEN Nroots := 0;
ELSIF D  = 0.0 THEN
        Nroots:= 1;
        X1 := -B/(2.0*A);
ELSE Nroots := 2;
        X1 := (-B+sqrt(D)) / (2.0*A) ;
        X2 := (-B-sqrt(D)) / (2.0*A) ;
END_IF
```

3.) Function Block Diagram Language

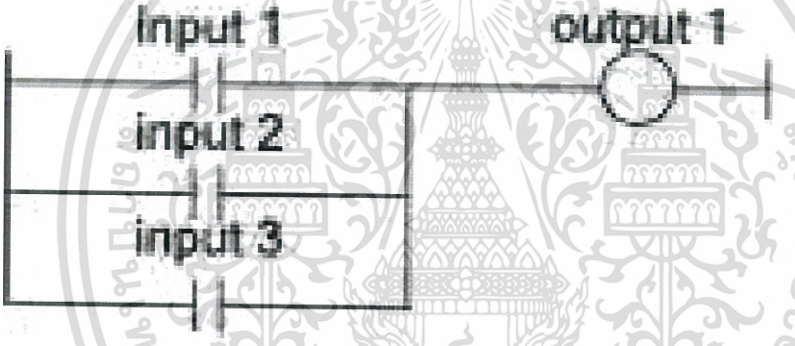


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.) Instruction List Language

```
Label:  LD      a1      (*result := a1*)
        ADD(    a2      (*delayed ADD, result := a2*)
        MUL(    a3      (*delayed MUL, result := a3*)
        SUB     a4      (*result := a3 - a4*)
        )               (*execute delayed MUL,*)
                        (*result := a1 + (a2*(a3 - a4) *a5)*)
        ADD     a6      (*a1 + (a2*(a3 - a4)*a5) + a6*)
        ST      res     (*store current result in res*)
```

5.) Ladder Diagram

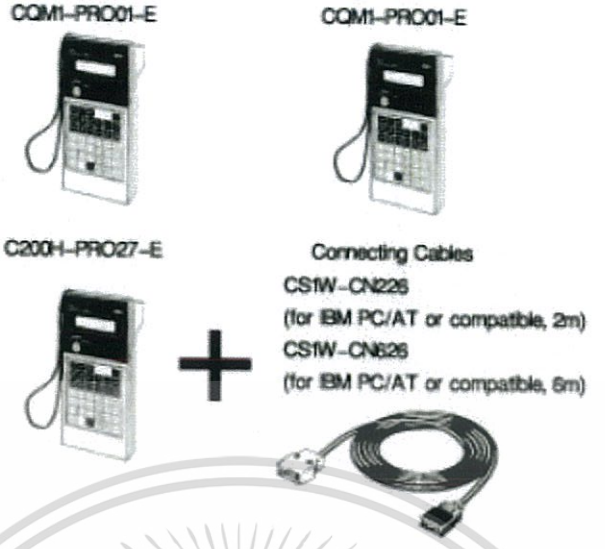


2.1.3 อุปกรณ์สำหรับการโปรแกรม [3]

การสั่งให้ พีแอลซี ทำงาน จะต้องป้อนโปรแกรมให้กับ พีแอลซี ก่อน ซึ่งอุปกรณ์ที่ใช้ในการป้อนโปรแกรมให้กับ พีแอลซี นั้น สามารถแบ่งได้เป็น 2 ประเภท

2.1.3.1 ตัวป้อนโปรแกรมแบบมือถือ (Hand Held Programmer) แต่ละยี่ห้อจะมีชื่อเรียกแตกต่างกัน เช่น OMRON จะเรียกว่า Programming Console เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



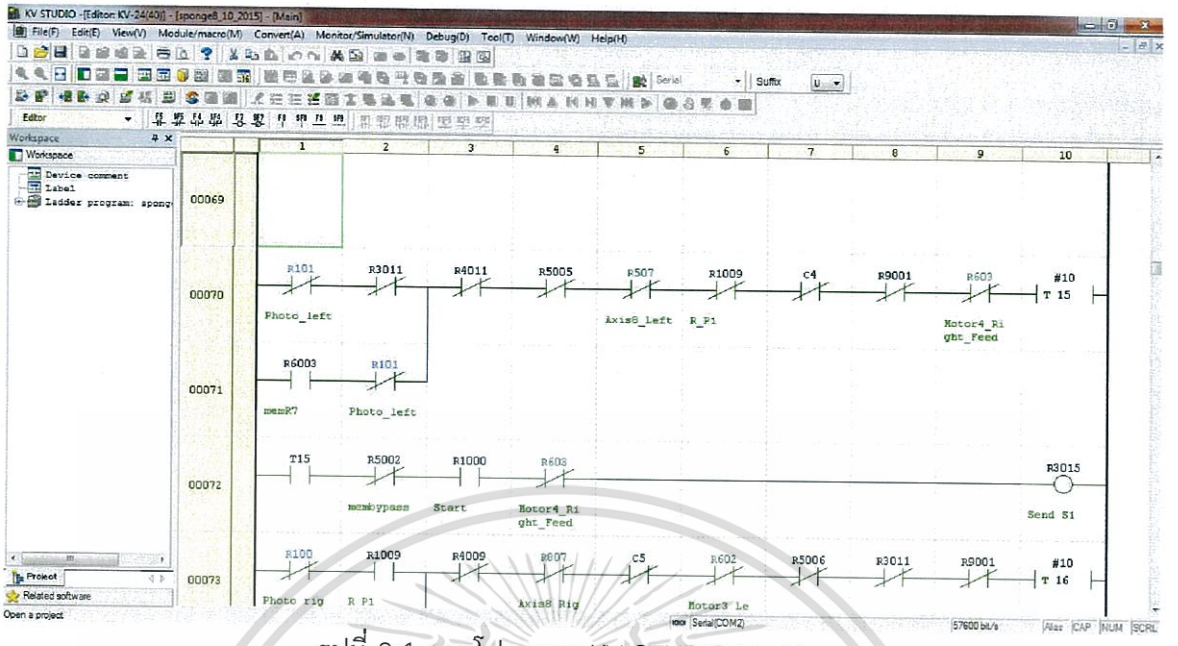
รูปที่ 2.3 การเขียนโปรแกรมโดยตัวป้อนโปรแกรมแบบมือถือ

การเขียนโปรแกรมให้กับ พีแอลซี โดยการใช้ Programming Console จะป้อนเป็นภาษา Statement List หรือ Mnemonic เช่น คำสั่ง LD, AND, OR ซึ่งเป็นคำสั่งพื้นฐาน สามารถเรียกใช้งานโดยการกดปุ่มที่อยู่บนตัว Programming Console นั้น แต่เมื่อต้องการใช้งานฟังก์ชันอื่นๆ ที่มีอยู่ใน พีแอลซี สามารถเรียกใช้งานได้โดยการกดปุ่มเรียกใช้คำสั่งพิเศษการใช้ Programming Console มีข้อดีตรงที่มีความสะดวกในการเคลื่อนย้าย และสามารถพกพาได้สะดวกเนื่องจากมีขนาดเล็ก แต่ก็มีข้อเสียคือในการใช้งานผู้ใช้ต้องศึกษาวิธีการใช้งานของอุปกรณ์เหล่านี้ว่ามีวิธีการกดอย่างไร ถึงจะสั่งงาน พีแอลซี ได้

2.1.3.2 คอมพิวเตอร์สามารถใช้ในการเขียนโปรแกรมให้กับ พีแอลซี ได้ โดยใช้งานร่วมกับซอฟต์แวร์ (Software) เฉพาะของ พีแอลซี ยี่ห้อนั้น เช่น พีแอลซี ของ Keyence จะใช้ซอฟต์แวร์ที่มีชื่อเรียกแตกต่างกันไป สามารถยกตัวอย่างได้เช่น

- KV STUDIO Ver.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



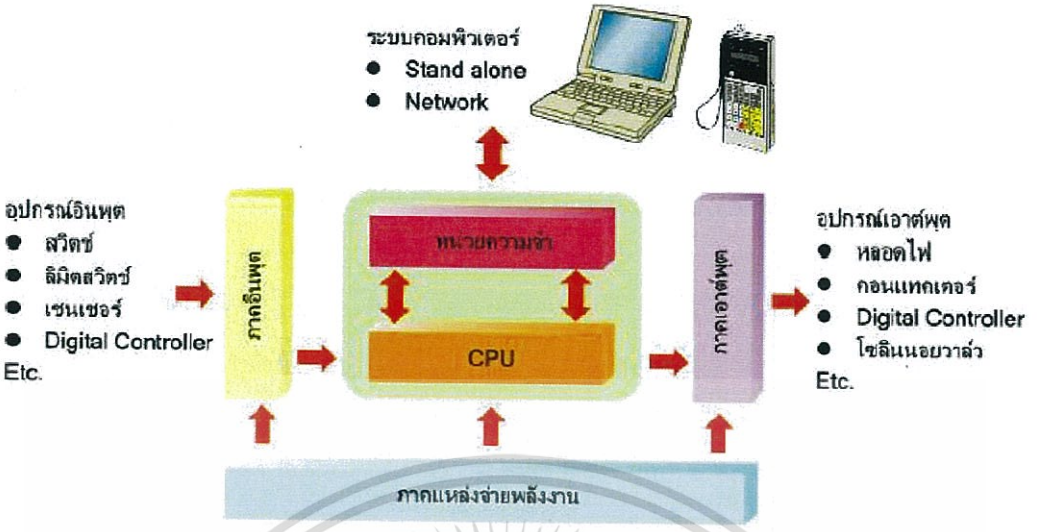
รูปที่ 2.4 ภาพโปรแกรม KV STUDIO Ver.7

ข้อดีของการใช้เครื่องคอมพิวเตอร์ในการป้อนโปรแกรมให้กับ พีแอลซี คือ ใช้งานง่าย เช่นในกรณีใช้ KV STUDIO Ver.7 ร่วมกับระบบปฏิบัติการ Window จากรูปที่ 2.4 ท่านจะเห็นว่าการเขียนโปรแกรมเป็นภาษา Ladder Diagram จะเป็นการนำสัญลักษณ์ต่างๆ เข้ามาใช้แทนการเขียนคำสั่ง ทำให้เข้าใจง่ายเพียงแค่คลิกเลือกสัญลักษณ์ต่างๆ จากส่วนของ Toolbar นอกจากนั้นยังมี Toolbar อื่นๆ ให้เลือกใช้งานซึ่งง่ายกว่าการใช้ Programming Console

2.1.4 โครงสร้างของ พีแอลซี

โครงสร้างภายในของ พีแอลซี แต่ละส่วนจะประกอบกันทำงานเป็นระบบควบคุมที่เรียกว่า พีแอลซี ซึ่งประกอบไปด้วยส่วนสำคัญดังรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 โครงสร้างของ พีแอลซี

2.1.4.1 ซีพียู (CPU; Central Process Unit)

ซีพียูหรือหน่วยประมวลผลกลาง ทำหน้าที่ประมวลผลการทำงานตามคำสั่งของส่วนต่างๆ ตามที่ได้รับมา ผลจากการประมวลผลก็จะถูกส่งออกไปส่วนต่างๆ ตามที่ระบุไว้ด้วยคำสั่งนั่นเอง ซีพียูจะใช้เวลาในการประมวลช้าหรือเร็ว ขึ้นอยู่กับการเลือกขนาดของซีพียู และความยาวของโปรแกรมที่เขียนด้วย

2.1.4.2 หน่วยความจำ (Memory Unit) หน่วยความจำเป็นอุปกรณ์ที่ใช้เก็บโปรแกรมและข้อมูลต่างๆ ของ พีแอลซี กรณีที่สั่งให้ พีแอลซี ทำงาน (RUN) มันจะนำเอาโปรแกรมและข้อมูลในหน่วยความจำมาประมวลผลการงาน สำหรับหน่วยความจำที่ใช้งานมีด้วยกัน 2 ชนิด คือ

- หน่วยความจำชั่วคราว (RAM: Random Access Memory) โปรแกรมและข้อมูลที่สร้างขึ้นโดยผู้ใช้จะถูกจัดเก็บในส่วนนี้ คุณสมบัติของRAM เมื่อไม่มีไฟเลี้ยงจะทำให้โปรแกรมและข้อมูลหายไปทันที ดังนั้นภายใน พีแอลซี จะพบว่าจะมีแบตเตอรี่สำรองข้อมูล (Backup Battery) เอาไว้สำรองข้อมูล (Backup Data) กรณีที่ไฟหลัก (Main Power Supply) ไม่จ่ายไฟให้กับ พีแอลซี ข้อควรระวังคือ ไม่ควรที่จะถอดแบตเตอรี่สำรอง (Backup Battery) กรณีที่ไม่มีไฟจ่ายให้ พีแอลซี
- หน่วยความจำถาวร (ROM: Read Only Memory)เป็นหน่วยความจำอีกชนิดหนึ่ง โดยที่ข้อมูลใน ROM ไม่จำเป็นต้องมีแบตเตอรี่สำรองข้อมูล แต่ก็มีปัญหาเรื่องเวลาในการเข้าถึงข้อมูล (Time Access) ช้ากว่า RAM จึงปรากฏให้ผู้ใช้เห็นว่า พีแอลซี จะมีหน่วยความจำใช้งานทั้ง RAM และ ROM ร่วมกันอยู่

2.1.4.3 ภาคอินพุต (Input Unit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคอินพุตของ พีแอลซี ทำหน้าที่รับสัญญาณอินพุตเข้ามาแปลงสัญญาณ ส่งเข้าไปภายใน พีแอลซี อุปกรณ์อินพุต (Input Device) ต่างๆ ที่นำมาต่อกับภาคอินพุตได้นั้น สามารถแสดงตัวอย่างได้ตามรูปที่แสดงดังนี้

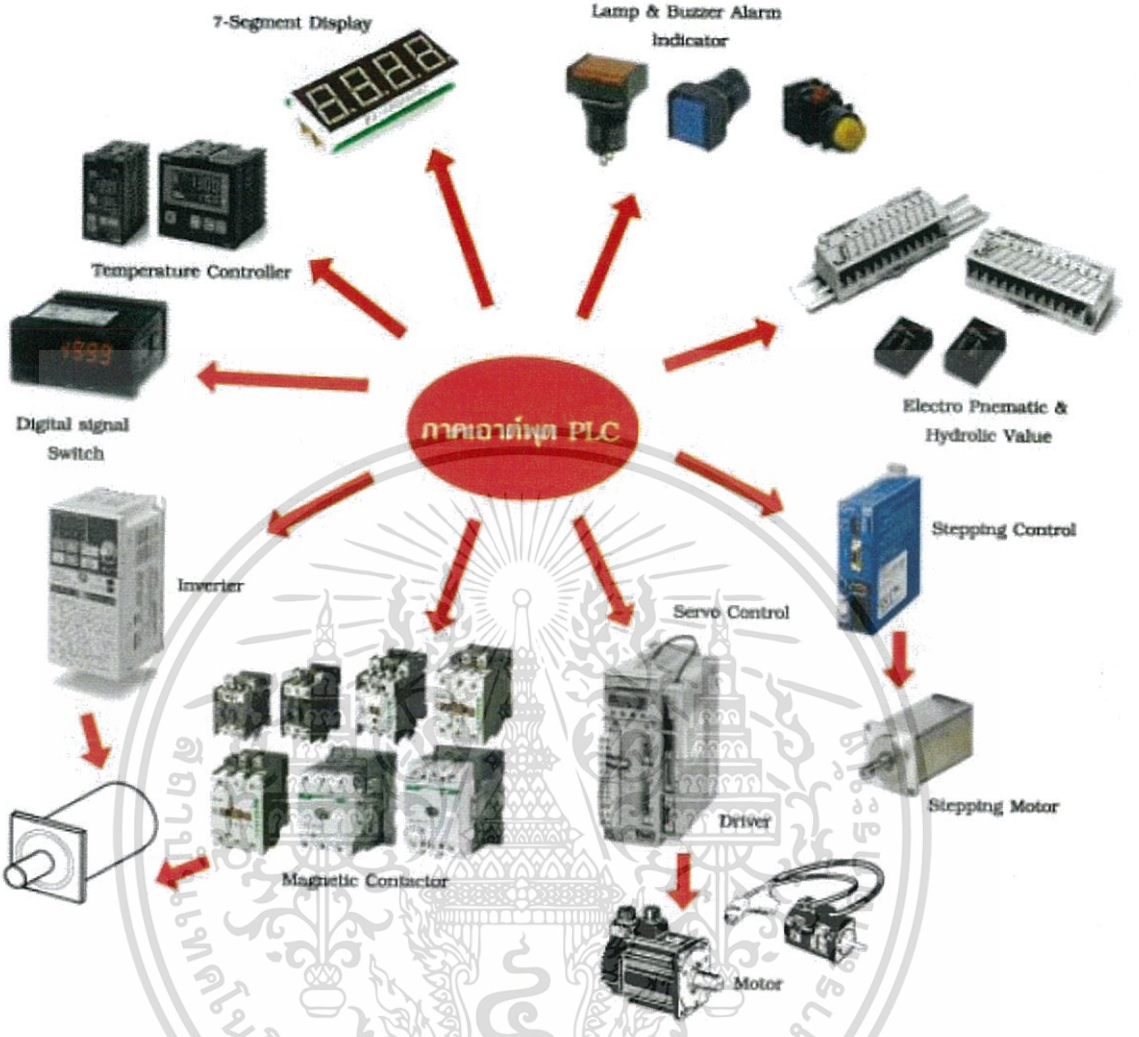
รูปที่ 2.6 ภาคอินพุต

อุปกรณ์ที่สามารถนำมาต่อกับภาคอินพุต พีแอลซี ได้จัดออกเป็นกลุ่มๆ ดังรูปที่ 2.6 โดยกลุ่มอุปกรณ์แต่ละกลุ่มจะมีวิธีต่อวงจรเข้าภาคอินพุต พีแอลซี แตกต่างกันออกไป เวลาใช้งานอุปกรณ์แต่ละกลุ่ม จำเป็นต้องศึกษาข้อมูลเพิ่มเติมของอุปกรณ์แต่ละชนิดก่อน เพื่อความเข้าใจขั้นตอนการทำงาน และสามารถต่อวงจรได้ถูกต้อง

2.1.4.4 ภาคเอาต์พุต (Output Unit)

ภาคเอาต์พุตของ พีแอลซี ทำหน้าที่ส่งสัญญาณออกไปขับโหลดชนิดต่างๆตามเงื่อนไขที่ได้โปรแกรมเอาไว้ ชนิดของโหลดที่สามารถนำมาต่อกับภาคเอาต์พุต สามารถแยกออกเป็นกลุ่มได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ภาคอินพุต

2.1.5 หลักการเขียนแลดเดอร์ไดอะแกรมและคำสั่งพื้นฐาน [4]

แลดเดอร์ไดอะแกรม (Ladder Diagram) จัดเป็นภาษาสัญลักษณ์ที่เทียบเคียงมาจากวงจรรีเลย์ สามารถดูตามโครงสร้างแล้วเข้าใจการทำงานได้ แต่เวลาที่ พีแอลซี ทำงานจะอาศัยชุดคำสั่ง (Instructions) ทำงานโดยวิธีการเขียนลงในส่วนหน่วยความจำข้อมูล ในหน่วยความจำนั้นจะจัดเก็บเป็นรหัส (Code) ไม่สามารถจัดเก็บในลักษณะของ Ladder Diagram ได้โดยตรง ดังนั้นผู้ใช้จึงจำเป็นต้องเข้าใจชุดคำสั่งเพราะชุดคำสั่งนั้นแปลงภาษามาจาก Ladder Diagram นั่นเอง

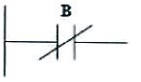
2.1.5.1 กลุ่มคำสั่งพื้นฐาน (Ladder Instruction & Output Control)

-การใช้คำสั่ง LOAD (LD), LOAD NOT (LD NOT)

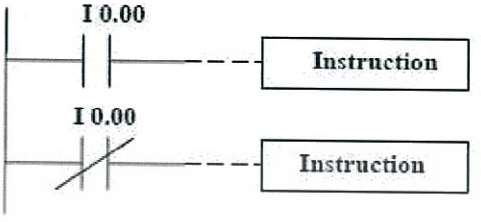
LOAD-LD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LOAD NOT-LD NOT



ตัวอย่างที่ 1 ชุดคำสั่งและการเขียน Ladder Diagram คำสั่ง LD และ LD NOT



Address	Instruction	Operands
00000	LD	0.00
00001	Instruction	
00002	LD NOT	0.00
00003	Instruction	

-การใช้คำสั่ง AND, AND NOT

AND-AND

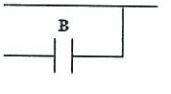
AND NOT-AND NOT

ตัวอย่างที่ 2 ชุดคำสั่งและการเขียน Ladder Diagram คำสั่ง AND, AND NOT

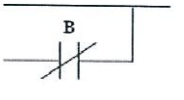
ddress	Instruction	Operands
00000	LD	0.00
00001	AND NOT	1.00
00002	AND	W 0.00
00003	Instruction	

-การใช้คำสั่ง OR, OR NOT

OR-OR

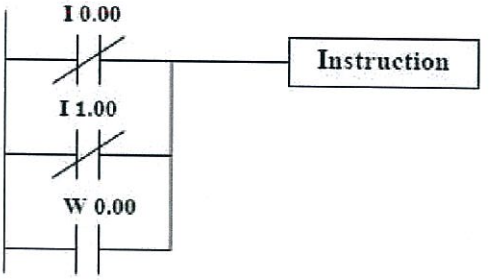


OR NOT-OR NOT



ตัวอย่างที่ 3 ชุดคำสั่งและการเขียน Ladder Diagram คำสั่ง OR, OR NOT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

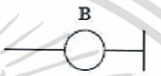


Address	Instruction	Operands
00000	LD NOT	0.00
00001	OR NOT	1.00
00002	OR	W 0.00
00003	Instruction	

-การใช้คำสั่ง OUT, OUT NOT

เป็นคำสั่งที่สั่งขับให้ OUTPUT ภายนอกทำงานหรือไม่ทำงานตามคำสั่ง

OUTPUT-OUT



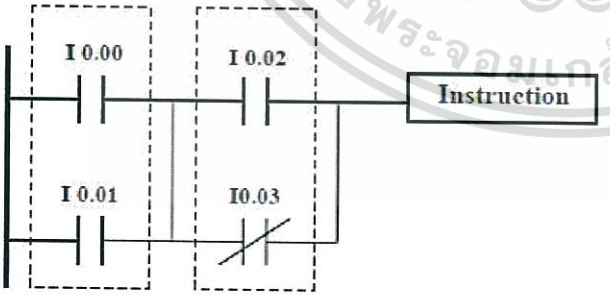
ตัวอย่างที่ 4 รูปแบบชุดคำสั่งจาก Ladder Diagram

Address	Instruction	Operands
00000	LD	0.00
00001	OUT	100.00

-การใช้คำสั่ง AND LOAD (AND LD), OR LOAD (OR LD)

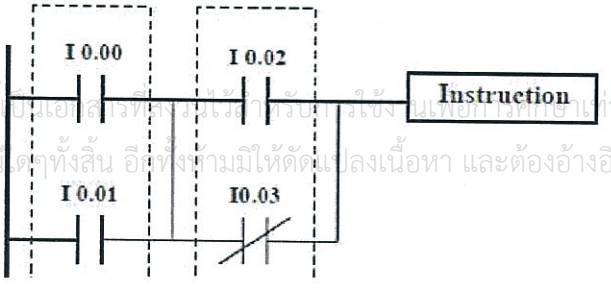
คำสั่งทั้งสองจะทำหน้าที่เชื่อมต่อกลุ่มแลดเดอร์ไดอะแกรม (Ladder Diagram) ในกรณีที่ต่ออนุกรม หรือขนานกันมากกว่า 1หน้าสัมผัส ซึ่งการใช้คำสั่ง AND หรือ OR นั้น จะกระทำทีละ 1 หน้าสัมผัสเท่านั้นจึงต้องใช้ AND LD หรือ OR LD

ตัวอย่างที่ 5 ชุดคำสั่งในรูปการเชื่อมแบบอนุกรมจะใช้คำสั่ง AND LD



Address	Instruction	Operands
00000	LD	0.00
00001	OR	0.01
00002	LD	0.02
00003	OR NOT	0.03
00004	AND LD	

ตัวอย่างที่ 6 ชุดคำสั่งในรูปการเชื่อมแบบขนานจะใช้คำสั่ง OR LD

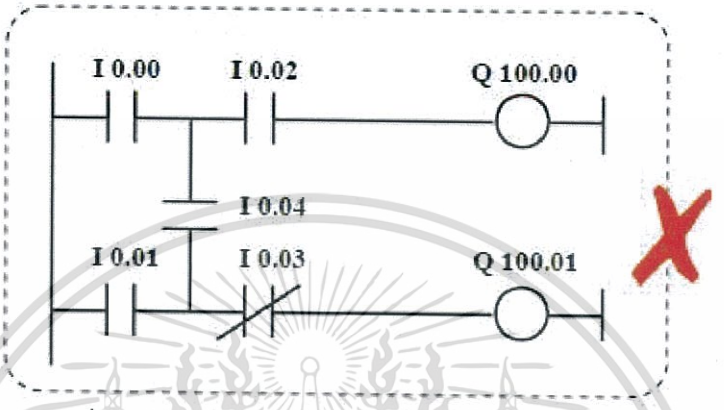


Address	Instruction	Operands
00000	LD	0.00
00001	AND	0.02
00002	LD	0.01
00003	AND NOT	0.03
00004	OR LD	- - -

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

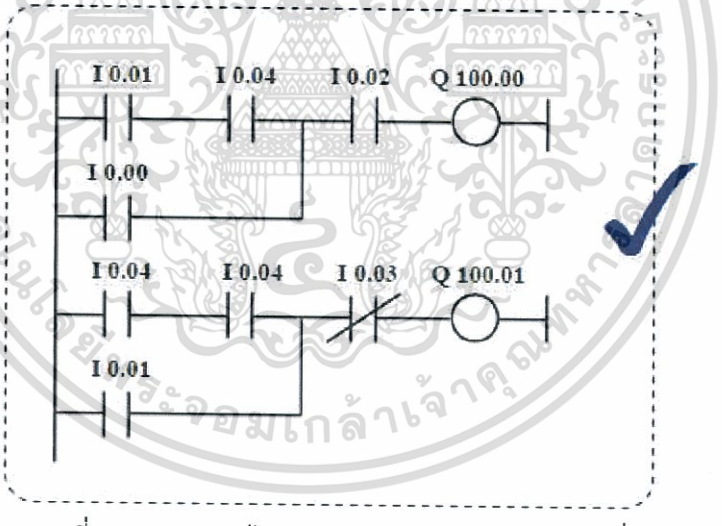
2.1.5.2 ข้อกำหนดในการเขียนแลดเดอร์ไดอะแกรม (Ladder Diagram)

-แลดเดอร์ไดอะแกรม (Ladder Diagram) ข้างล่าง จะไม่สามารถเขียนโปรแกรมได้ จำเป็นต้องแปลงชุด Ladder Diagram ก่อน



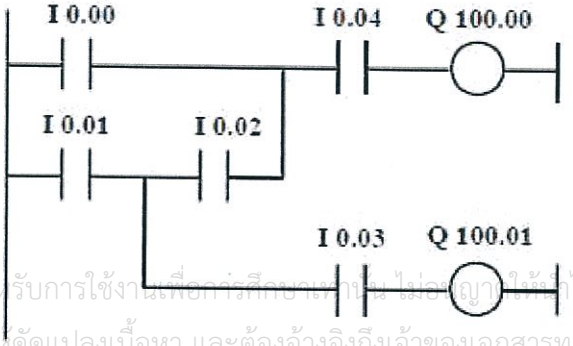
รูปที่ 2.8 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด

สามารถเขียนใหม่ได้ และวงจรทำงานเหมือนเดิม คือ



รูปที่ 2.9 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก

-แลดเดอร์ไดอะแกรม (Ladder Diagram) จะพิจารณาการทำงานจากซ้ายไปขวาเท่านั้น ดังตัวอย่างเช่น

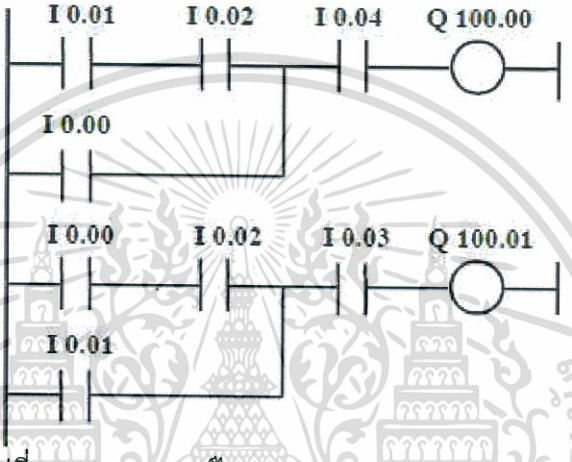


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

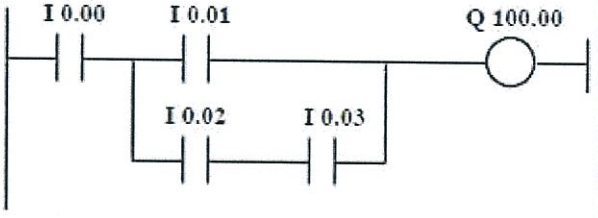
รูปที่ 2.10 แลดเดอร์ไดอะแกรม (Ladder Diagram) A

จากแลดเดอร์ไดอะแกรม (Ladder Diagram) A ถ้าหน้าสัมผัส 0.00, 0.02 และ 0.03มีสภาวะ “ON” ก็ไม่สามารถทำให้ เอาต์พุต 100.01 นั้น “ON” ได้เลย ดังนั้น ผู้ใช้จะต้องทำการจัดโปรแกรมเสียใหม่เพื่อให้การทำงานกระทำจากซ้ายไปขวาดังรูป แลดเดอร์ไดอะแกรม B



รูปที่ 2.11 แลดเดอร์ไดอะแกรม (Ladder Diagram) B

-จำนวนหน้าคอนแทคทั้ง NO และ NC ของอินพุต/เอาต์พุต, รีเลย์และไทม์เมอร์ (TIM)/เคาน์เตอร์ (CNT) จะนำมาเขียนโปรแกรมเป็นจำนวนเท่าใดก็ได้ตามความประสงค์ของผู้ใช้อย่างไรก็ตามการเขียนโปรแกรมที่ดีจะต้องพยายามประหยัดขนาดของโปรแกรมให้มากที่สุดเท่าที่จะสามารถทำได้ซึ่งจะเปรียบเทียบให้เห็นในแลดเดอร์ไดอะแกรม (Ladder Diagram) A และแลดเดอร์ไดอะแกรม (Ladder Diagram) B จะสังเกตเห็นได้ว่าการเขียนในแลดเดอร์ไดอะแกรม (Ladder Diagram) B จะประหยัดคำสั่งได้ 2 คำสั่ง ในขณะที่โปรแกรมทำงานได้เหมือนกัน

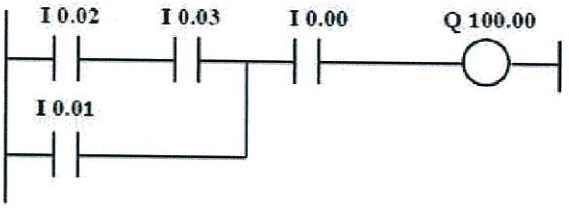


Address	Instruction	Operands
00000	LD	0.00
00001	LD	0.01
00002	LD	0.02
00003	AND	0.03
00004	OR LD	
00005	AND LD	
00006	OUT	100.00

แลดเดอร์ไดอะแกรม (Ladder Diagram) A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

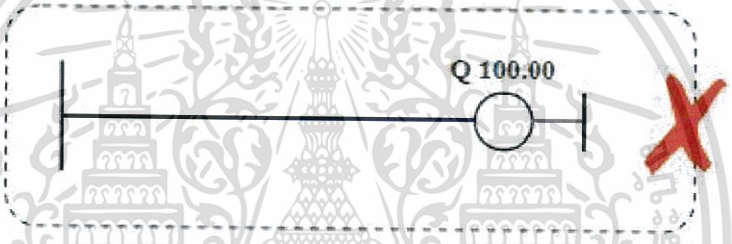
147158



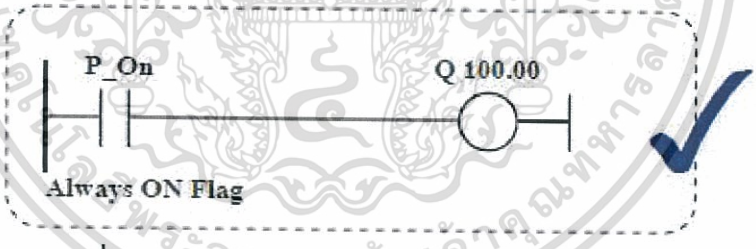
Address	Instruction	Operands
00000	LD	0.02
00001	AND	0.03
00002	OR	0.01
00003	AND	0.00
00004	OUT	100.00

แลดเดอร์ไดอะแกรม (Ladder Diagram) B

-เมื่อต้องการให้เอาต์พุต ON ตลอดเวลาเราจะใช้แฟลค (Flag) ที่เป็นแบบ “Always ON Flag” (CF113) มาเป็นตัวสร้างเงื่อนไขเพราะไม่สามารถต่อคอยล์เอาต์พุตได้โดยตรงกับ Bus Bar แต่ก็มีข้อยกเว้นเป็นบางคำสั่ง เช่น INTERLOCK CLEAR, JUMP END และ STEP



รูปที่ 2.12 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด

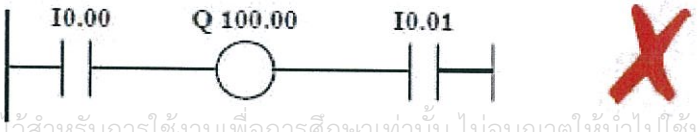


รูปที่ 2.13 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก

-จำนวนหน้าสัมผัสที่ใช้ในการต่ออนุกรม หรือขนานไม่มีขีดจำกัดจะใช้เท่าใดก็ได้ขึ้นอยู่กับความต้องการของผู้ใช้

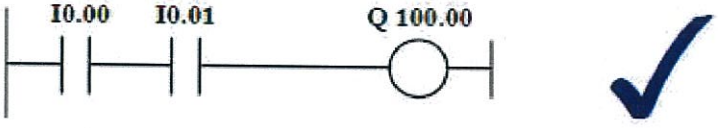
-เอาต์พุตทุกๆ ตัวจะมี Auxiliary Contact เพื่อใช้งานในโปรแกรมได้ และสามารถใช้จำนวนไม่จำกัด

-ไม่สามารถเขียนโปรแกรมให้หน้าสัมผัสอยู่ตำแหน่งหลังจากคอยล์ได้



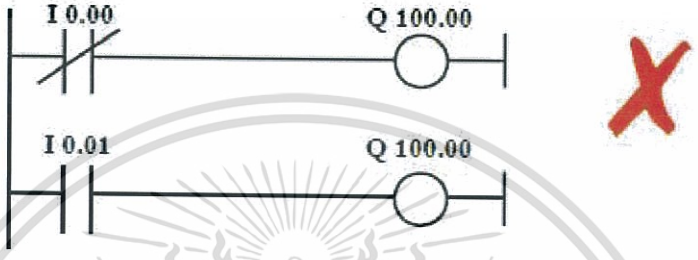
รูปที่ 2.14 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

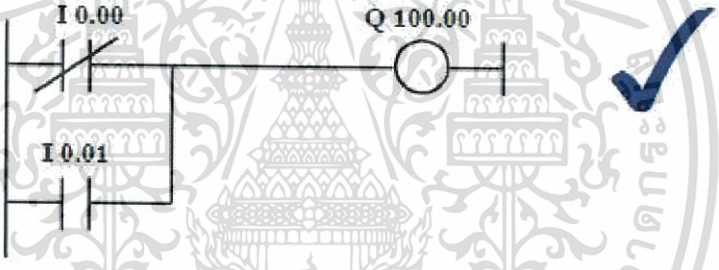


รูปที่ 2.15 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก

-ไม่สามารถเขียนโปรแกรมให้มีเอาต์พุตเบอร์เดียวกันซ้ำหลายๆ ครั้งได้ ต้องจัดรูปเสียใหม่

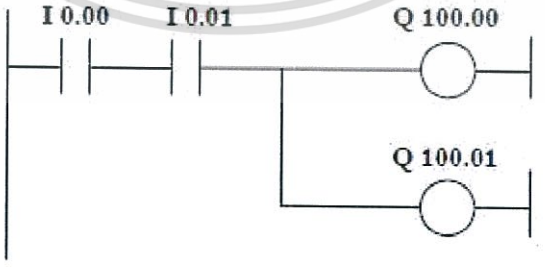


รูปที่ 2.16 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ผิด



รูปที่ 2.17 แลดเดอร์ไดอะแกรม (Ladder Diagram) ที่ถูก

-เอาต์พุตคอยล์ สามารถเขียนโปรแกรมให้ต่อขนานได้เลย กรณีรับเงื่อนไขของหน้าสัมผัสชุดเดียวกัน



รูปที่ 2.18 การเขียนโปรแกรมแบบขนาน

-พีแอลซี จะเริ่มประมวลผลโปรแกรมจาก Address แรกสุดจนกระทั่งถึงคำสั่ง END ตำแหน่งแรก โดยที่คำสั่ง END อาจจะมีหลายตำแหน่งในโปรแกรมที่เป็นเช่นนี้เพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

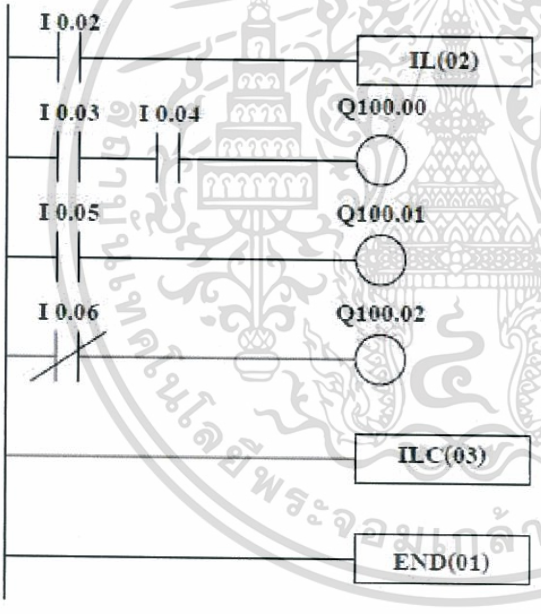
จุดประสงค์สำหรับการทดสอบโปรแกรม กรณีที่ต้องการแยกโปรแกรมออกเป็นส่วนๆ เพื่อให้ง่ายต่อการตรวจสอบและแก้ไขโปรแกรม

2.1.5.3 กลุ่มคำสั่ง Program Control Instruction

-การใช้คำสั่ง IL(02), ILC(03)คำสั่ง IL และ ILC

จะต้องใช้ร่วมกันคือ ถ้าเริ่มต้นมีการใช้คำสั่งด้วย IL เมื่อใดแล้วถ้าต้องการสิ้นสุดการทำงานต้องจบด้วย ILC เงื่อนไขของคำสั่งคือ คอนแทคตรงหน้าส่วนของ IL มีสภาวะ “ON” จะทำให้โปรแกรมที่อยู่ระหว่าง IL และ ILC ทำงานเป็นปกติ แต่ถ้าคอนแทคตำแหน่งดังกล่าวมีสภาวะ “OFF” จะทำให้การทำงานของโปรแกรมระหว่าง IL และ ILC ไม่ทำงาน ในขณะเดียวกัน Output Coil ในช่วงนั้นจะมีสภาวะ “OFF” ด้วย

ตัวอย่างที่ 7 การใช้คำสั่ง

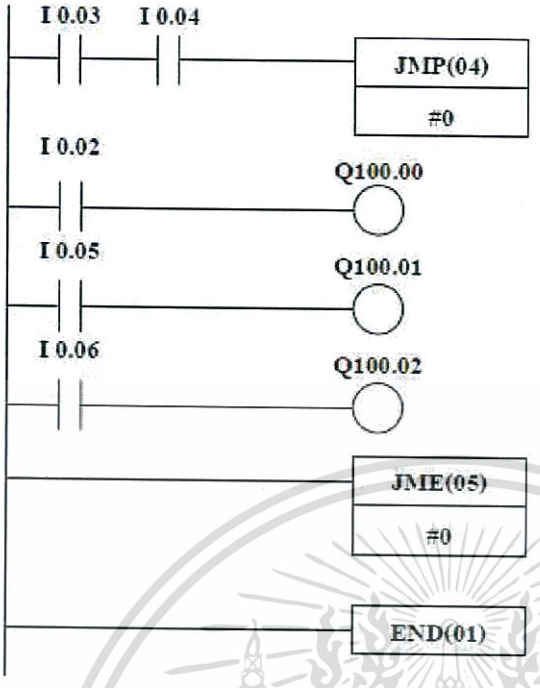


Address	Instruction	Data
00000	LD	0.02
00001	IL (02)	-
00002	LD	0.03
00003	AND	0.04
00004	OUT	100.00
00005	LD	0.05
00006	OUT	100.01
00007	LD NOT	0.06
00008	OUT	100.02
00009	ILC(03)	-
00010	END(01)	-

-การใช้คำสั่ง JMP (04) และ JME (05)

การใช้งานของคำสั่งคู่นี้จะต้องใช้งานคู่กัน เงื่อนไขต่างๆ ที่อยู่ระหว่างคำสั่ง JMP และ JME จะมีเงื่อนไขการทำงานเป็นปกติ ในกรณีที่ชุดของคอนแทคตรงส่วนหน้าของ JMP มีสภาวะเป็น “ON” แต่ถ้าชุดคอนแทคดังกล่าวมีสภาวะเป็น “OFF” เมื่อใด Output, Timer, Counter, Keep ที่อยู่ระหว่างคำสั่งดังกล่าวจะยังคงค้างสภาวะเอาไว้เช่นเดิม และจะมีการเปลี่ยนแปลงอีกครั้งถ้าชุดของคอนแทคมีสภาวะ “ON” เราใช้ JUMP 00 ได้หลายครั้งตามต้องการ แต่ JUMP 01 ถึง 49สามารถใช้ได้เพียงครั้งเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Address	Instruction	Operand
0000	LD	0.02
0001	AND	0.03
0002	JMP(04)	#0
0003	LD	0.04
0004	OUT	100.00
0005	LD	0.05
0006	OUT	100.01
0007	LD	0.06
0008	OUT	100.02
0009	JME(05)	#0
0010	END(01)	

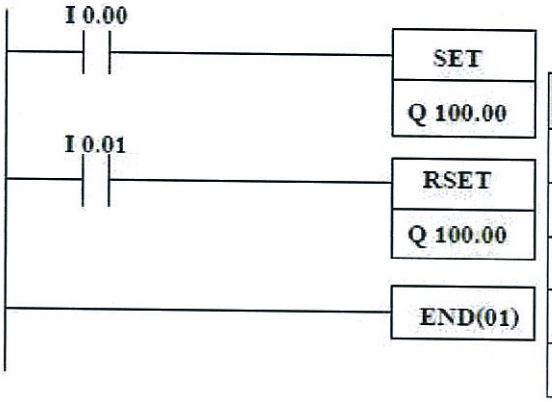
2.1.5.4 คำสั่งในกลุ่ม Bit Control Instruction

-การใช้คำสั่งเซต (SET) และรีเซต (RESET)

คำสั่ง SET จะทำให้บิตที่ถูกสั่ง “ON” และคงค้างอยู่จนกว่าจะมีคำสั่ง RSET ที่บิตเดียวกัน บิตนั้นจึงจะ “OFF”

ตัวอย่างที่ 8 ต้องการให้หลอดไฟที่เอาต์พุต 100.00 ติดตลอดเวลา หลังจาก ON อินพุต 000.00 แล้วครั้งเดียวโดยไม่ต้อง Hold คำสั่งอินพุต จนกว่าจะมีการ Reset ที่อินพุต 000.01 หลอดไฟจะดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



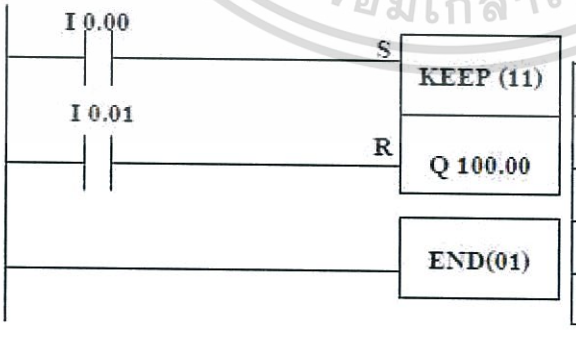
Address	Instruction	Operands
00000	LD	0.00
00001	SET	100.00
0002	LD	0.01
00003	RSET	100.00
00004	END	

-การทำงานของคำสั่ง KEEP

จะเหมือนกับคำสั่ง SET และ RESET เพียงแต่จับขาSET/RESET ให้รวมอยู่ในตัวเดียวกัน เพื่อให้ผู้ใช้งานสามารถเลือกใช้โปรแกรมได้สะดวกตามความเหมาะสม

เมื่อขา S มีสถานะ “ON” บิตที่ B จะทำงานจนกว่าขา R จะมีสถานะ “ON” บิต B ถึงจะเลิกทำงาน

ตัวอย่างที่ 5.11 ต้องการให้เอาต์พุต 100.00 เปลี่ยนเป็น “ON” ตลอดเวลาโดยการ ON อินพุต 0.00 ไม่ว่าจะ “OFF” แล้วก็ตามจนกว่าอินพุต 0.01 จะ ON (RESET)

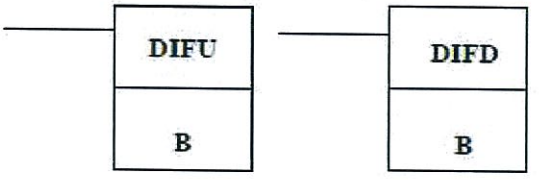


Address	Instruction	Operands
00000	LD	0.00
00001	LD	0.01
00002	KEEP	100.00
00003	END	

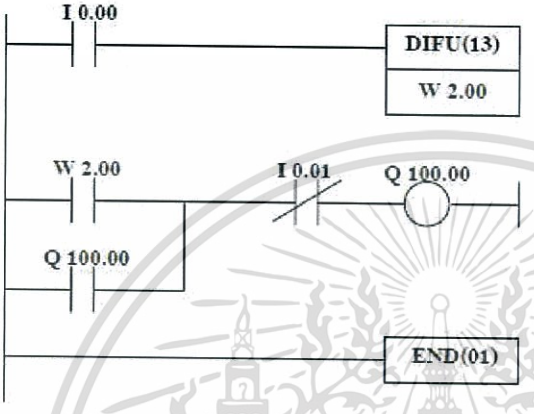
-การใช้คำสั่ง DIFFERENTIATE UP และ DOWN–DIFU(13), DIFD(14)

คำสั่ง DIFU(13) และ DIFD(14) จะเป็นคำสั่งที่ทำงานเพียงขอบขาขึ้น หรือขอบขาลงของสัญญาณอินพุตเท่านั้น และจะทำงานเพียงช่วงเวลา One Cycle เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ตัวอย่างที่ 9 ต้องการให้อินพุต 0.00 ทำงานเพื่อ ON บิต W2.00 ซึ่งจะทำให้เอาต์พุตหลอดไฟ 100.00 ให้ติดได้โดยอินพุต 0.01 เป็นตัวสั่ง OFF



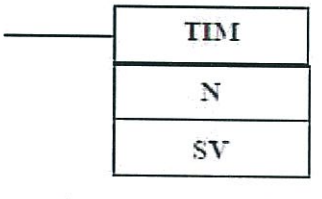
Address	Instruction	Operands
00000	LD	0.00
00001	DIFU	W2.00
0002	LD	W2.00
00003	OR	100.00
00004	AND NOT	
00005	OUT	100.00
00006	END	

2.1.5.5 กลุ่มคำสั่ง Timer/Counter

สำหรับ พีแอลซี บางรุ่น Timer และ Counter จะใช้พื้นที่เดียวกันซึ่งเรียกใช้ได้ทั้งหม256 ตัวตั้งแต่ตัวที่ 000 ถึง 255 ภายใน 256 ตัวนี้สามารถกำหนดให้เป็น Timer หรือ Counter ก็ได้โดยที่หากตัวใดถูกกำหนดให้เป็น Timer แล้วจะนำไปใช้เป็น Counter อีกไม่ได้ ดังนั้นต้องดู Manual ของพีแอลซี รุ่นนั้นประกอบด้วย ถ้า Timer/Counter อยู่ในพื้นที่เดียวกัน จะไม่สามารถใช้เบอร์เดียวกันได้แต่ พีแอลซี บางรุ่น Timer/Counter จะอยู่คนละพื้นที่ ดังนั้นจึงสามารถใช้ Timer และ Counterเบอร์เดียวกันได้ เช่น T000 และ C000สำหรับคำสั่งในกลุ่ม Timer/Counter มีหลายคำสั่ง ในที่นี้จะยกตัวอย่างการใช้งานคำสั่งTimer/Counter แบบพื้นฐานคือ คำสั่ง TIM และ CNT ดังนี้

-การใช้คำสั่ง TIMER: TIM

ใช้ในการจับเวลา, ตั้งเวลา โดยพื้นฐานแล้วต้องเข้าไปกำหนดค่า 2 ค่าคือ N และSV ตามตัวอย่างข้างล่าง



N = Timer Number (เบอร์ 0000 - 4095) เลือกว่าจะใช้ Timer ตัวที่เท่าใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

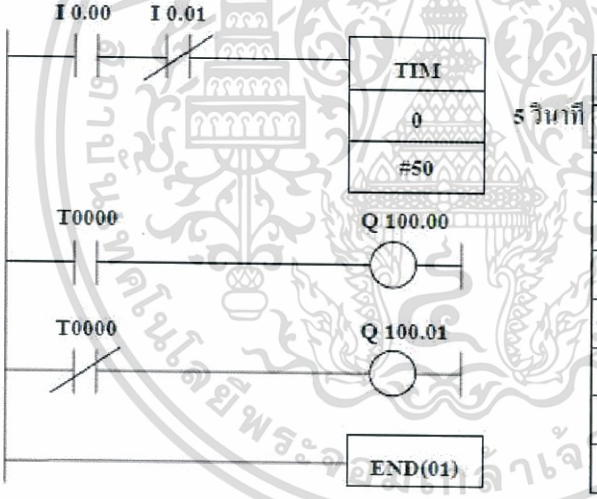
SV = Set Value ค่าตั้งเวลา ใช้กำหนดว่าจะให้ Timer ตั้งเวลานานเท่าใด ซึ่ง SV ที่ตั้งนั้น จะถูกคูณด้วย 0.1 เพื่อแปลงเป็นระยะเวลาจริง ซึ่งสามารถ

1.กำหนด SV เป็นค่าคงที่ #0000-9999 (000.0-999.9 วินาที คูณด้วย 0.1วินาที)

2.กำหนด SV เป็น แอดเดรส CIO,A,T,C,IR,DR,D,TK,H,Wโดยใส่ค่าตั้งเวลาที่เป็นค่าคงที่ 0000-9999 ไว้ใน แอดเดรส ที่อ้างถึงอีกทีหนึ่ง(ค่าที่กำหนดจะคูณด้วย 0.1 วินาทีเช่นเดียวกับการกำหนดแบบค่าคงที่)

เมื่อมีสัญญาณสั่งให้ Timer ทำงาน (Contact B มีสถานะ “ON”) คำสั่ง Timer จะเริ่มนับเวลาตามค่าที่ตั้งไว้ใน Timer เมื่อนับครบเวลา หน้า Contact ของ Timer ตัวนั้นๆ ก็จะ “ON” แต่ถ้าสัญญาณที่สั่งให้ Timer ทำงานหายไป (Contact B มีสถานะ OFF) Timer จะถูก Reset

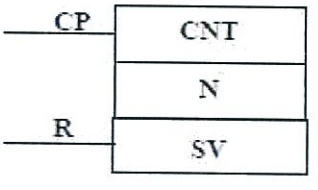
ตัวอย่างที่ 10 การใช้งานของคำสั่ง Timer เมื่ออินพุต 0.00 ทำงาน (ON) ค้างเป็นเวลา 5 วินาที เอาต์พุต 100.00 จะ ON และ เอาต์พุต 100.01 จะ OFF



Address	Instruction	Operands
00000	LD	0.00
00001	AND NOT	0.01
00002	TIM 000	#050
00003	LD	TIM0000
00004	OUT	100.00
00005	LD NOT	TIM0000
00006	OUT	100.01
00007	END (01)	

-การใช้คำสั่ง COUNTER - CNT

เป็นคำสั่งที่ใช้นับจำนวนครั้งของสัญญาณ อินพุต ที่ ON แต่ละครั้ง ซึ่งเป็นคำสั่งที่นับลงจากค่าที่ตั้งไว้ (Set Value)



N = Counter Number (เบอร์ 000- 4095) เลือกว่าจะใช้ Counter ตัวที่เท่าใด

SV = Set Value ค่าตั้งจำนวนนับ ใช้กำหนดว่าจะให้ Counter นับสัญญาณอินพุตเป็นจำนวนกี่ครั้ง หน้า Contact เอาต์พุตของ Counter จึงจะเริ่มทำงานซึ่งสามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

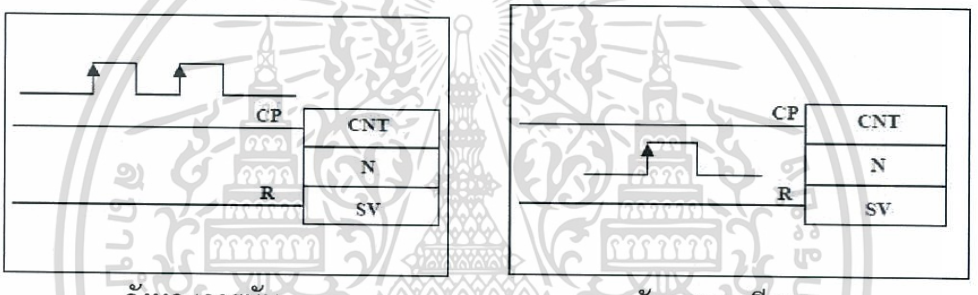
1. กำหนด SV เป็นค่าคงที่ #0000-9999

ค่าคงที่ 0000-9999 ไว้ในแอดเดรส ที่อ้างถึงอีกทีหนึ่ง

CP = ขานับ เมื่อมีสัญญาณอินพุตในช่วงที่เปลี่ยนสถานะจาก OFF เป็น ONเข้ามาที่ขานี้ Counter จะนับถอยหลังลง 1

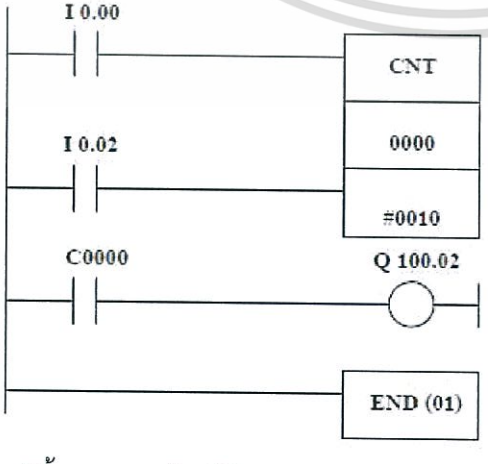
R = ขา Reset เมื่อมีสัญญาณอินพุตเข้ามาที่ขานี้ เอาต์พุตของ Counter จะหยุดทำงานและค่านับของ Counter จะถูก Reset กลับไปเท่ากับค่าตั้งจำนวนนับ (SV)

ตัวอย่างที่ 11 แสดงการใช้คำสั่ง Counter สัญญาณที่ส่งเข้าที่ขาสัญญาณนับจะทำให้ Counter นับถอยหลังเมื่อเทียบกับ SV ตัว Counter จะเริ่มนับที่ขอบขาขึ้นของสัญญาณอินพุตและจะนับใหม่เมื่อสัญญาณเปลี่ยนเป็น OFF หรือ '0' แล้วกลับมา ON อีกครั้ง และจะทำงานเช่นนี้ไปจนครบค่าที่ตั้งไว้ (SV) เอาต์พุตจึง ON



จังหวะการนับ จังหวะการรีเซต

ตัวอย่างที่ 12 การใช้งานของคำสั่ง Counter เมื่อ อินพุต 0.00 ทำงาน (ON) 1 ครั้งCounter จะนับ 1 ครั้ง ถ้าอินพุต 0.00 ทำงาน (ON) ครบ 10 ครั้ง จะทำให้คำสั่ง Counter ทำงานพร้อมกับContact ของ Counter (CNT0000) จะทำงานด้วย และจะถูก Reset ด้วยอินพุต 0.02



Address	Instruction	Operands
00000	LD	0.00
00001	LD	0.02
00002	CNT 001	#0010
00003	LD	CNT0000
00004	OUT	100.02
00005	END (01)	

2.1.6 ความรู้พื้นฐานทางด้านดิจิตอล [5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากพื้นฐานของ พีแอลซี นั้นมาจากการทำงานของวงจรรีเลย์ ซึ่งมีสภาวะการทำงานแบบลอจิก (0 และ 1) ซึ่งก็คือค่าทางดิจิตอลนั่นเอง ดังนั้นก่อนการใช้งาน พีแอลซี จะต้องมีความรู้พื้นฐานในเรื่องของเลขฐานและวิธีการแปลงเลขฐานก่อน เพื่อจะเข้าใจการทำงานที่แท้จริงและสามารถใช้งาน พีแอลซี ได้ดี

2.1.6.1 ระบบเลขฐาน (Number System)

ระบบเลขฐาน จัดเป็นระบบตัวเลขที่ใช้งานอยู่ใน พีแอลซี ในบทนี้จะยกตัวอย่างเฉพาะการใช้งานระบบเลขฐานสอง, เลขฐานสิบ และเลขฐานสิบหกเท่านั้น

- ระบบเลขฐานสอง (Binary: BIN) มีตัวเลขที่ไม่ซ้ำกันอยู่ทั้งหมด 2 ตัว คือ 0 และ 1
- ระบบเลขฐาน BCD (Binary Code Decimal: BCD) มีตัวเลขที่ไม่ซ้ำกันอยู่ทั้งหมด 10ตัว คือ 0 1 2 3 4 5 6 7 8 9 หรือเรียกอีกอย่างหนึ่งว่า BCD code
- ระบบเลขฐานสิบหก (Hexadecimal: HEX) มีตัวเลขที่ไม่ซ้ำกันอยู่ทั้งหมด 16 ตัว คือ 0 1 2 3 4 5 6 7 8 9 A B C D E F (ตัวอักษร 6 ตัว แทน ตัวเลข 10 –15)

HEX	BCD	FOUR DIGIT BINARY			
		$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$
0	0	0	0	0	0
1	1	0	0	0	1
2	2	0	0	1	0
3	3	0	0	1	1
4	4	0	1	0	0
5	5	0	1	0	1
6	6	0	1	1	0
7	7	0	1	1	1
8	8	1	0	0	0
9	9	1	0	0	1
A	-	1	0	1	0
B	-	1	0	1	1
C	-	1	1	0	0
D	-	1	1	0	1
E	-	1	1	1	0
F	-	1	1	1	1

รูปที่ 2.19 ตารางเลขฐานต่างๆ

หมายเหตุ BIN (Binary) = ระบบเลขฐานสอง HEX (Hexadecimal) = ระบบเลขฐานสิบหก

BCD (Binary Code Decimal) = ระบบเลขฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 ASCII

เดิมการแทนรหัสฐานสองด้วยพยัญชนะในภาษาต่างๆเป็นการกำหนดกันเองขึ้นอยู่กับว่าใครพัฒนาขึ้นมาทำให้การส่งผ่านข้อมูลระหว่างเครื่องคอมพิวเตอร์ต่างชนิดไม่สามารถส่งผ่านกันได้เพราะใช้รหัสในเลขฐานสองไม่เหมือนกันดังนั้นเพื่อที่จะให้สื่อสารกันได้จึงจำเป็นต้องมีการกำหนดมาตรฐานของรหัสขึ้นสำหรับภาษาอังกฤษเรียกว่ารหัสแอสกี้(American Standard Code Interchange ,ASCII)

American Standard Code For Information Interchange (ASCII) หรือ แอส-กี้ เป็นรหัสที่พัฒนาขึ้นโดยสถาบันมาตรฐานแห่งชาติสหรัฐอเมริกา(American National Standard Institute: ANSI อ่านว่า แอน-ซาย) เรียกว่า ASCII Code ซึ่งเป็นที่นิยมในกลุ่มผู้สร้างเครื่องคอมพิวเตอร์ทั่วไป

รหัสนี้ได้มาจากรหัสขององค์กรมาตรฐานระหว่างประเทศ (International Standardization Organization: ISO) ขนาด 7 บิท ซึ่งสามารถสร้างรหัสที่แตกต่างกันได้ถึง 128รหัส (ตั้งแต่ 000 0000 ถึง 111 1111) โดยกำหนดให้ 32 รหัสแรกเป็น 000 0000 ถึง 0011111 ทำหน้าที่เป็นสั่งควบคุม เช่น รหัส 000 1010 แทนการเลื่อนบรรทัด (Line Feed)ในเครื่องพิมพ์ เป็นต้น และอีก 96 รหัสถัดไป (32-95) ใช้แทนอักษรและสัญลักษณ์พิเศษอื่นรหัส ASCII ใช้วิธีการกำหนดการแทนรหัสเป็นเลขฐานสิบ ทำให้ง่ายต่อการจำและใช้งาน นอกจากนั้นยังสามารถเขียนมนรูปของเลขฐานสิบหกได้ด้วย ดังนั้น ASCII Code จึงเป็นรหัสที่เขียนได้ 3 แบบ เช่นอักษร A สามารถแทนเป็นรหัสได้ดังนี้

สัญลักษณ์	เลขฐานสิบ	เลขฐานสอง	เลขฐานสิบหก
A	65	100 0001	41

รูป 2.20 ตัวอย่างรหัส ASCII แปลเป็นเลขฐานอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

รูป 2.21 รูปตาราง ASCII แปลงเป็นเลขฐานอื่นๆ

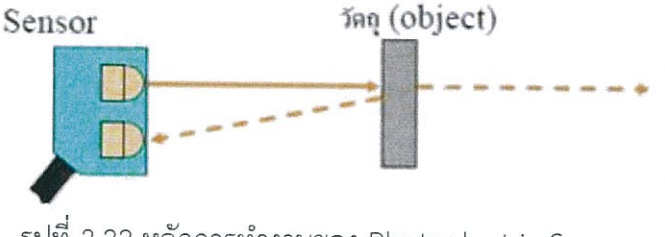
2.2 เซ็นเซอร์และอุปกรณ์ [6]

เซ็นเซอร์เป็นอุปกรณ์สำคัญที่ใช้งานอุตสาหกรรมในระบบการควบคุมแบบอัตโนมัติซึ่งสามารถแบ่งแยกตามลักษณะการใช้งานและคุณสมบัติที่ได้ดังนี้

2.2.1 Photoelectric Sensors [7]

คืออุปกรณ์ตรวจจับด้วยแสง คือการควบคุมแสงที่ใช้ในกระบวนการผลิตอัตโนมัติต่างๆ โดยทำงานตรวจจับแสงที่มองเห็นหรือแสงที่มองไม่เห็น และตอบสนองการทำงานตามการเปลี่ยนแปลงความเข้มของแสงที่ได้รับ โดย Photoelectric Sensors สามารถแบ่งได้หลักๆ 3 ประเภท

2.2.1.1 Diffuse Mode (สะท้อนวัตถุโดยตรง)

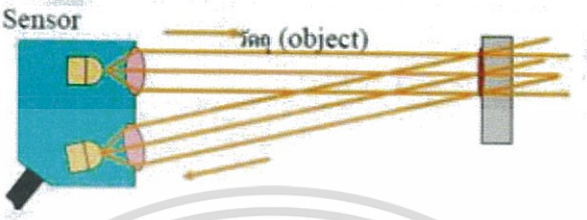


รูปที่ 2.22 หลักการทำงานของ Photoelectric Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น Sensor ที่อาศัยหลักการยิงแสงไปที่วัตถุ แล้วสะท้อนกลับมา ซึ่ง Sensor ลักษณะนี้นิยมใช้งานโดยทั่วไป เนื่องจากใช้พื้นที่ติดตั้งน้อย เพราะใช้ผิววัตถุที่ตรวจจับเป็นตัวสะท้อนแสงกลับมา

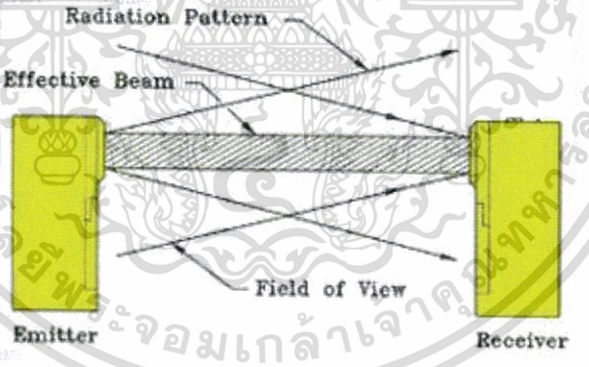
2.2.1.2 Retroreflective Mode (สะท้อนวัตถุโดยตรงแบบจำกัดลำแสง)



รูปที่ 2.23 หลักการทำงานของ Retroreflective Mode

เป็น Sensor ที่ต้องอาศัยแผ่นสะท้อน หรือที่เราเรียกว่า Reflect เป็นตัวสะท้อนแสงกลับมา ซึ่ง Sensorลักษณะนี้ สามารถนำไปใช้งานได้ดีในบริเวณที่มีการจำกัดพื้นที่การติดตั้ง นอกจากนี้แผ่นสะท้อน ยังส่งผลทำให้ระยะการตรวจจับวัตถุสามารถทำได้ไกลขึ้น

2.2.1.3 Opposed Mode (มีตัวส่งและตัวรับแยกกัน)



รูปที่ 2.24 หลักการทำงานของ Opposed Mode

เป็น Sensor แบบที่ใช้ตัวส่งและตัวรับ เนื่องจาก Sensor ลักษณะนี้ มีทั้งตัวส่งและตัวรับ ดังนั้นจึงทำให้ระยะการตรวจจับวัตถุสามารถตรวจจับได้ระยะไกลมากขึ้น นอกจากนี้ในยังสามารถนำไปใช้งานในสภาพแวดล้อมที่มีฝุ่นมากกว่าปกติได้จากประเภทของ Photoelectric Sensor ทั้ง 3 ประเภทข้างต้น ทำให้ในเบื้องต้น เราสามารถตัดสินใจเลือกใช้รูปแบบการติดตั้งได้อย่างเหมาะสม อย่างไรก็ตาม ยังมีปัจจัยอื่นๆ ที่สำคัญ ดังนี้

- สี , พื้นผิวของวัตถุที่ตรวจจับ (Color , Surface) สีต่างๆกัน มีผลต่อการดูดซับของแสง ลักษณะพื้นผิว เช่น ผิวมันวาว ผิวขรุขระ มีผลต่อการสะท้อนของแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความโตของลำแสง (Beam Pattern) ถ้าความโตของลำแสง มีขนาดใหญ่กว่าวัตถุที่ตรวจจับ ก็จะทำให้แสงนั้นยิงผ่านวัตถุไป บางครั้งทำให้ไม่สามารถตรวจจับได้ ดังนั้นความโตของลำแสง จึงต้องมีขนาดเล็กกว่าวัตถุที่ต้องการตรวจจับ

- ความแตกต่างของสี (Contrast) เนื่องจาก Photoelectric Sensors จะมองเห็นวัตถุที่ต้องการตรวจจับเป็น Gray Scale หรือ เป็นสีเทานั่นเอง ดังนั้น Photo Sensors จะตรวจจับวัตถุที่มีค่า Contrast สูงได้ดีกว่า วัตถุที่มีค่า Contrast ต่ำ

2.2.2 Proximity Sensor

คือ เซนเซอร์ชนิดหนึ่งที่สามารถทำงานโดยไม่ต้องสัมผัสกับชิ้นงานหรือวัตถุภายนอก โดยลักษณะของการทำงานอาจจะส่งหรือรับพลังงานรูปแบบใดรูปแบบหนึ่งดังต่อไปนี้ คือ สนามแม่เหล็ก สนามไฟฟ้า แสง เสียง และ สัญญาณลม ส่วนการนำเซนเซอร์ประเภทนี้ไปใช้งานนั้น ส่วนใหญ่จะใช้กับงานตรวจจับ ตำแหน่ง ระดับ ขนาด และรูปร่าง ซึ่งโดยปกติแล้วจะนำมาใช้แทนลิมิตสวิตช์ (Limit Switch) เนื่องด้วยสาเหตุของอายุการใช้งานและความเร็วในการตรวจจับวัตถุเป้าหมายทำได้ดีกว่าอุปกรณ์ประเภทสวิตช์ซึ่งอาศัยหน้าสัมผัสทางกล

คุณสมบัติเด่น

- สามารถตรวจจับได้โดยไม่มีการสัมผัส
- สามารถใช้งานได้ดีในสภาพแวดล้อมที่เลวร้าย
- ตรวจจับด้วยความแม่นยำ
- ตอบสนองต่อการทำงานได้รวดเร็วกว่า
- สามารถแยกการตรวจจับวัตถุที่เป็นโลหะ อโลหะและแม่เหล็กได้
- อายุการใช้งานยาวนาน
- จะมีระยะการตรวจจับวัตถุโดยทั่วๆไปอยู่ระหว่า4-40mmขึ้นอยู่กับขนาดและชนิดของ sensors

ประเภทของ Proximity switch type

2.2.2.1 เซนเซอร์แบบเหนี่ยวนำ (Inductive Sensor)

เป็นเซนเซอร์ที่ทำงานโดยอาศัยหลักการเปลี่ยนแปลงค่าความเหนี่ยวนำของขดลวด ซึ่งการเปลี่ยนแปลงดังกล่าวจะมีผลต่อชิ้นงานหรือวัตถุที่เป็นโลหะเท่านั้น ข้อเด่นของเซนเซอร์ชนิดนี้ คือ ทนทานสามารถทำงานได้ในช่วงอุณหภูมิที่กว้าง (wide temperature ranges)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถทำงานในสภาวะที่มีการรบกวนทางแสง (Optical) และเสียง (Acoustic) ซึ่งเทียบเท่ากับชนิดเก็บประจุ

รูปที่ 2.25 รูปร่างของเซนเซอร์แบบเหนี่ยวนำ

2.2.2.2 เซนเซอร์ชนิดเก็บประจุ (Capacitive Sensor)

โครงสร้างพื้นฐาน ของ Capacitive Proximity Switch จะมีลักษณะคล้ายกับแบบ Inductive Proximity Switch จะมีส่วนต่างกันที่หัวตรวจจับ (Active Electrode) ซึ่งจะใช้หลักการเปลี่ยนแปลงของค่าคาปาซิแตนซ์ (Capacitance) capacitive proximity sensor จะสร้าง สนามไฟฟ้าสถิตย์ (electrostatic) มาแทนที่จะเป็นสนามแม่เหล็กไฟฟ้าจึงทำให้ capacitive proximity sensor นี้สามารถที่จะตรวจจับวัตถุที่เป็นทั้งโลหะและอโลหะได้ ซึ่งถือเป็นข้อได้เปรียบของเซนเซอร์ประเภทนี้

รูปที่ 2.26 รูปร่างเซนเซอร์ชนิดเก็บประจุ

2.2.3 รีเลย์ (Relays) [8]

รีเลย์เป็นอุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในการตัด-ต่อวงจรคล้ายกับสวิตช์ โดยทั่วไปจะเป็นแบบ Electromagnetic Relay หรือเรียกว่าแบบหน้าสัมผัส ประกอบด้วยชุดหน้าสัมผัส (Contacts) ที่ต่อกับแท่งอาร์เมเจอร์ (Armature) และคอยล์ (Coil) ที่ถูกพันด้วยขดลวด เมื่อมีการจ่ายแรงดันไฟฟ้าให้กับคอยล์ (Energize) จะทำให้เกิดสนามแม่เหล็ก แท่งอาร์เมเจอร์ที่ต่อกับหน้าสัมผัสจะถูกดูด ทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้หน้าสัมผัสเปลี่ยนการเชื่อมต่อเป็นตรงกันข้าม กล่าวคือ ปกติเปิด (NO-Normally Open) เป็นปิด หรือปกติปิด (NC-Normally Closed) เป็นเปิด และเมื่อตัดไฟที่จ่ายให้คอยล์ (Deenergize) จะทำให้รีเลย์กลับสู่สถานะปกติ กล่าวคือ หน้าสัมผัสต่างๆ จะกลับสู่สภาวะแรกก่อนการจ่ายไฟด้วยแรงจากสปริง

รูปที่ 2.27 หลักการทำงานของรีเลย์

2.2.4 สวิตชิ่งเพาเวอร์ซัพพลาย (Switching Power Supply)

สวิตชิ่งเพาเวอร์ซัพพลาย (Switching Power Supply) คือแหล่งจ่ายไฟตรงคงค่าแรงดันแบบหนึ่ง และสามารถเปลี่ยนแรงดันไฟจากไปสลับโวลต์สูง ให้เป็นแรงดันไฟตรงค่าต่ำ เพื่อใช้ในงานอิเลคทรอนิกส์ได้เช่นเดียวกันแหล่งจ่ายไฟเชิงเส้น (Linear Power Supply) ถึงแม้เพาเวอร์ซัพพลายทั้งสองแบบจะต้องมีการใช้หม้อแปลงในการลดทอนแรงดันสูงให้เป็นแรงดันต่ำเช่นเดียวกัน แต่สวิตชิ่งเพาเวอร์ซัพพลายจะต้องการใช้หม้อแปลงที่มีขนาดเล็ก และน้ำหนักน้อย เมื่อเทียบกับแหล่งจ่ายไฟเชิงเส้น อีกทั้งสวิตชิ่งเพาเวอร์ซัพพลายยังมีประสิทธิภาพสูงกว่าอีกด้วย

ในปัจจุบันสวิตชิ่งเพาเวอร์ซัพพลาย (Switching Power Supply) ได้เข้ามามีบทบาทกับชีวิตอย่างมาก เครื่องใช้อิเลคทรอนิกส์ขนาดเล็กซึ่งต้องการแหล่งจ่ายไฟที่มีกำลังสูงแต่มีขนาดเล็ก เช่น เครื่องคอมพิวเตอร์ เครื่องโทรสาร และ โทรทัศน์ จำเป็นจะต้องใช้สวิตชิ่งเพาเวอร์ซัพพลาย แนวโน้มการนำสวิตชิ่งเพาเวอร์ซัพพลายมาใช้ในเครื่องใช้อิเลคทรอนิกส์ทุกประเภทจึงเป็นไปได้สูง การศึกษาหลักการทำงานและการออกแบบสวิตชิ่งเพาเวอร์ซัพพลายจึงเป็นสิ่งจำเป็นที่ไม่อาจหลีกเลี่ยงได้สำหรับผู้ที่เกี่ยวข้องกับงานอิเ ลคทรอนิกส์ทุกประเภท

- Switching Power Supply กับแหล่งจ่ายไฟเชิงเส้น

ข้อได้เปรียบของสวิตชิ่งเพาเวอร์ซัพพลายเมื่อเปรียบเทียบกับแหล่งจ่ายไฟเชิง เส้น คือ ประสิทธิภาพที่สูง ขนาดเล็ก และน้ำหนักเบากว่าแหล่งจ่ายไฟเชิงเส้น เนื่องจากแหล่งจ่ายไฟเชิงเส้นใช้หม้อแปลงความถี่ต่ำจึงมีขนาดใหญ่ และน้ำหนักมาก ขณะใช้งานจะมีแรงดันและกระแสผ่านตัวหม้อแปลงตลอดเวลา กำลังงานสูญเสียที่เกิดจากหม้อแปลงจึงมีค่าสูง การคงค่าแรงดันแหล่งจ่ายไฟเชิงเส้นส่วนมากจะใช้เพาเวอร์ทรานซิสเตอร์ต่ออนุ กรมที่เอาต์พุตเพื่อจ่ายกระแสและคงค่าแรงดัน กำลังงานสูญเสียในรูปความร้อนจะมีค่าสูงและต้องใช้แผ่นระบายความร้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

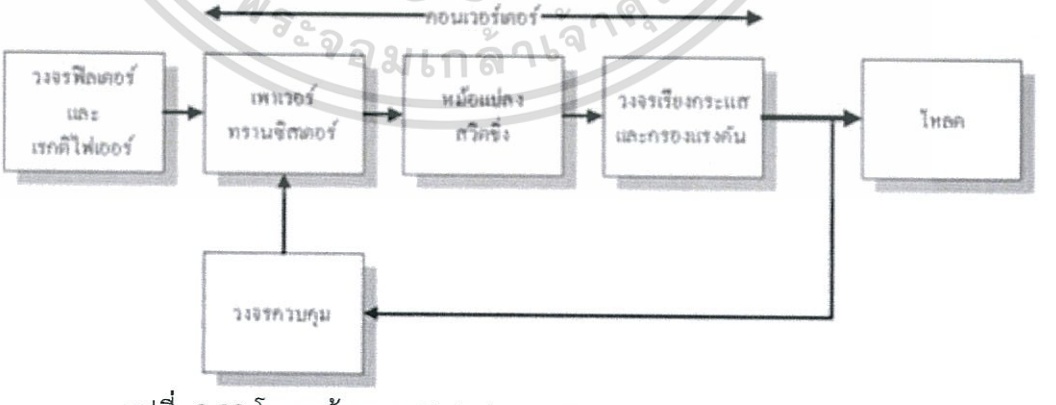
ขนาด ใหญ่ซึ่งกินเนื้อที่ เมื่อเพาเวอร์ซัพพลายต้องจ่ายกำลังงานสูงๆ จะทำให้มีขนาดใหญ่และมีน้ำหนักมาก ปกติแหล่งจ่ายไฟเชิงเส้นจะมีประสิทธิภาพประมาณ 30% หรืออาจทำได้สูงถึง 50% ในบางกรณี ซึ่งนับได้ว่าค่อนข้างต่ำเมื่อเปรียบเทียบกับสวิตชิ่งเพาเวอร์ซัพพลายซึ่งมีประสิทธิภาพในช่วง 65%-80%

สวิตชิ่งเพาเวอร์ซัพพลาย (Switching Power Supply) มีช่วงเวลาโคลสต์อัพประมาณ $20x10^{-3}$ ถึง $50x10^{-3}$ วินาที ในขณะที่แหล่งจ่ายไฟเชิงเส้นจะทำได้เพียงประมาณ $2x10^{-3}$ วินาที ซึ่งมีผลต่อการจัดหาแหล่งจ่ายไฟสำรองเพื่อป้องกันการหยุดทำงานของอุปกรณ์ที่ ใช้กับเพาเวอร์ซัพพลายเมื่อเกิดการหยุดจ่ายแรงดันไฟสลับ รวมทั้งสวิตชิ่งเพาเวอร์ซัพพลายสามารถทำงานได้ในช่วงแรงดันอินพุตค่อน ข้างกว้างจึงยังคงสามารถทำงานได้เมื่อเกิดกรณีแรงดันไฟตกอีกด้วย

อย่างไรก็ตาม สวิตชิ่งเพาเวอร์ซัพพลายจะมีเสถียรภาพในการทำงานที่ต่ำกว่า และก่อให้เกิดสัญญาณรบกวนได้สูงเมื่อเปรียบเทียบกับแหล่งจ่ายไฟเชิงเส้น รวมทั้งสวิตชิ่งเพาเวอร์ซัพพลายยังมีความซับซ้อนของวงจรมากกว่าและมีราคาสูง ที่กำลังงานต่ำๆ แหล่งจ่ายไฟเชิงเส้นจะประหยัดกว่าและให้ผลดีเท่าเทียมกัน ดังนั้นสวิตชิ่งเพาเวอร์ซัพพลายจึงมักนิยมใช้กันในงานที่ต้องการกำลังงานตั้งแต่20 วัตต์ขึ้นไปเท่านั้น

- หลักการทำงานของสวิตชิ่งเพาเวอร์ซัพพลาย

สวิตชิ่งเพาเวอร์ซัพพลาย (Switching Power Supply) โดยทั่วไปมีองค์ประกอบพื้นฐานที่คล้ายคลึงกัน และไม่ซับซ้อนมากนัก ดังแสดงในรูปที่ 2.28 หัวใจสำคัญของสวิตชิ่งเพาเวอร์ซัพพลายจะอยู่ที่คอนเวอร์เตอร์ เนื่องจากทำหน้าที่ทั้งลดทอนแรงดันและคงค่าแรงดันเอาต์พุตด้วยองค์ประกอบต่างๆ ทำงานตามลำดับดังนี้



รูปที่ 2.28 โครงสร้างของ Switching Power Supply

แรงดันไฟสลับค่าสูงจะผ่านเข้ามาทางวงจร RFI ฟิลเตอร์ เพื่อกรองสัญญาณรบกวนและแปลงเป็นไฟตรงค่าสูงด้วยวงจรเรกติไฟเออร์ เพาเวอร์ทรานซิสเตอร์จะทำงานเป็นเพาเวอร์คอนเวอร์เตอร์โดยการตัดต่อแรงดันเป็นช่วงๆ ที่ความถี่ประมาณ 20-200 KHzจากนั้นจะ

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผ่านไปยังหม้อแปลงสวิตชิ่งเพื่อลดแรงดันลง เอาต์พุตของหม้อแปลงจะต่อกับวงจรเรียงกระแส และกรองแรงดันให้เรียบ การคงค่าแรงดันจะทำได้โดยการป้อนกลับค่าแรงดันที่เอาต์พุตกลับมายังวงจรควบคุม เพื่อควบคุมให้เพาเวอร์ทรานซิสเตอร์นำกระแสมากขึ้นหรือน้อยลงตามการเปลี่ยนแปลงของแรงดันที่เอาต์พุต ซึ่งจะมีผลทำให้แรงดันเอาต์พุตคงที่ได้

ดังนั้น สวิตชิ่งเพาเวอร์ซัพพลาย เป็นแหล่งจ่ายไฟตรงที่มีประสิทธิภาพในการทำงานสูงกว่าและมีน้ำหนักเบากว่าเพาเวอร์ซัพพลายเชิงเส้น สวิตชิ่งเพาเวอร์ซัพพลายทำงานโดยแปลงแรงดันไฟสลับความถี่ต่ำจากอินพุตให้เป็นไฟตรง จากนั้นจึงเปลี่ยนกลับไปเป็นไฟสลับ (พัลส์) ที่ความถี่สูง แล้วส่งผ่านหม้อแปลงเพื่อลดแรงดันลง และผ่านวงจรเรียงกระแสและกรองแรงดันเพื่อให้ได้ไฟตรงอีกครั้งหนึ่ง สวิตชิ่งเพาเวอร์ซัพพลายประกอบด้วย 3 ส่วนใหญ่ คือ วงจรฟิลเตอร์และเรกติไฟเออร์ทำหน้าที่แปลงแรงดันไฟสลับเป็นไฟตรง คอนเวอร์เตอร์ ทำหน้าที่แปลงไฟตรงเป็นไฟสลับความถี่สูง และแปลงกลับเป็นไฟตรงโวลต์ต่ำ และวงจรควบคุมทำหน้าที่ควบคุมการทำงานของคอนเวอร์เตอร์ เพื่อให้ได้แรงดันเอาต์พุตตามต้องการ

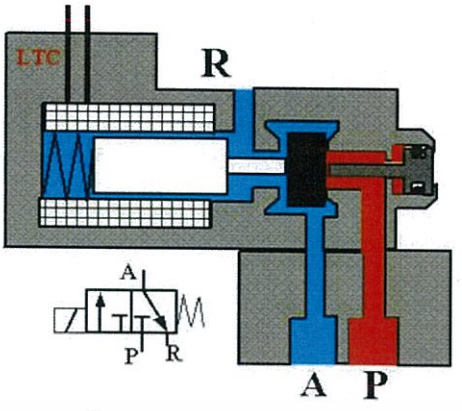
รูป 2.29 Switching Power Supply

2.2.5 โซลินอยด์วาล์ว (Solenoid valve) [9]

โซลินอยด์วาล์ว (Solenoid valve) คืออุปกรณ์สวิตช์ที่อาศัยหลักการทำงานของแม่เหล็กไฟฟ้าทำงานร่วมกับกลไกโดยใช้การป้อนไฟเป็นตัวกาหนดเงื่อนไขในการทางานควบคุมให้ลิ้นกลไกปิดหรือเปิดได้ อุปกรณ์ที่ใช้โซลีนอยด์วาล์วควบคุม ได้แก่วาล์วน้ำ เบรก และคลัตช์ เป็นต้น ประเภทของโซลินอยด์วาล์วลมที่ใช้กันทั่วไป

- วาล์วควบคุม3 ทิศทาง2ตำแหน่ง ปกติปิด เลื่อนวาล์วโดยโซลินอยด์ วาล์วเลื่อนกลับโดยสปริง

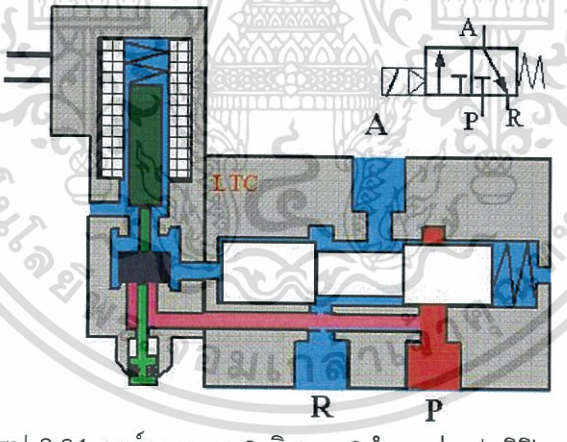
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.30 วาล์วควบคุม3 ทิศทาง2ตำแหน่ง ปกติปิด

- ปกติสปริงจะดันให้วาล์วปิดลมจากรูPไม่สามรถผ่านไปรูAได้
- เมื่อป้อนไฟให้โซลินอยด์ แกนจะถูกดึงให้เลื่อนไปทางซ้ายมือด้วยอำนาจของแม่เหล็กไฟฟ้า วาล์วจะเปิดให้ลมผ่านจากรูPไปรูA
- เมื่อตัดไฟออกจากโซลินอยด์ อำนาจแม่เหล็กของโซลินอยด์หมดไป สปริงจะดันแกนให้เลื่อนไปทางขวามือดันวาล์วให้ปิดรูPไว้ ลมจากรูAจะระบายออกที่รูR

- วาล์วควบคุม3 ทิศทาง2ตำแหน่ง ปกติปิด เลื่อนวาล์วโดยโซลินอยด์และลมดันช่วย วาล์วเลื่อนกลับโดยสปริง

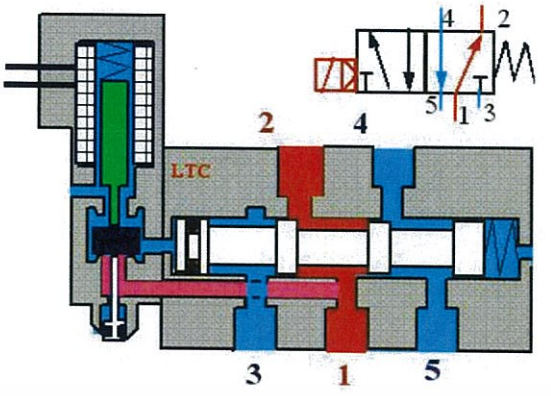


รูป 2.31 วาล์วควบคุม3 ทิศทาง2ตำแหน่ง ปกติปิด

- ปกติ สปริงจะดันให้วาล์วไพลอตปิด สปริงจะดันให้ลูกสูบเลื่อนไปทางซ้ายมือ ลมจากรูP ไม่สามรถผ่านไปรูAได้ รูAจะต่อกับรูR
- เมื่อป้อนไฟให้โซลินอยด์ แกนจะถูกดึงให้เปิดวาล์วไพลอต วาล์วไพลอตจะเปิดให้ลมไปดันลูกสูบให้เลื่อนไปด้านขวามือ เปิดให้ลมผ่านจากรูPไปยังรูA
- เมื่อตัดไฟออกจากโซลินอยด์ อำนาจแม่เหล็กของโซลินอยด์หมดไป สปริงจะดันให้แกนเลื่อนลงดันให้วาล์วไพลอตปิด สปริงจะดันให้ลูกสูบกลับตำแหน่งปกติ

- วาล์วควบคุม 5 ทิศทาง 2 ตำแหน่ง เลื่อนวาล์วโดยโซลินอยด์และลมดันช่วย วาล์วเลื่อนกลับโดยสปริง

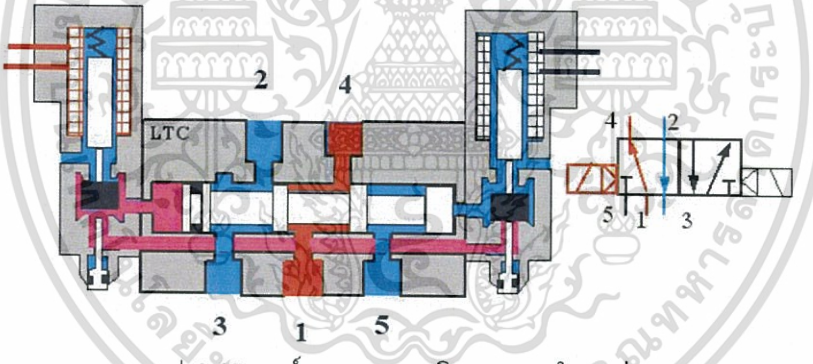
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.32 วาล์วควบคุม 5 ทิศทาง 2 ตำแหน่ง

- ปกติ ลมจากรู1ต่อไปยังรู2 ลมจากรู4ต่อไปยังรู5 ส่วนรู3อุดตัน
- เมื่อป้อนไฟฟ้าให้กับโซลินอยด์ วาล์วไพลอตจะเปิดให้ลมไปดันลูกสูบให้เลื่อนไปทางขวามือ ลมจากรู1จะต่อไปยังรู4 ส่วนลมจากรู2จะไหลไปยังรู3 ส่วนรู5อุดตัน
- เมื่อตัดไฟฟ้าออกจากโซลินอยด์ สปริงจะดันลูกลูบกลับตำแหน่งปกติ

- วาล์วควบคุม 5 ทิศทาง 2 ตำแหน่ง เลื่อนวาล์วโดยโซลีนอยด์ทั้ง 2 ข้าง



รูป 2.33 วาล์วควบคุม 5 ทิศทาง 2 ตำแหน่ง

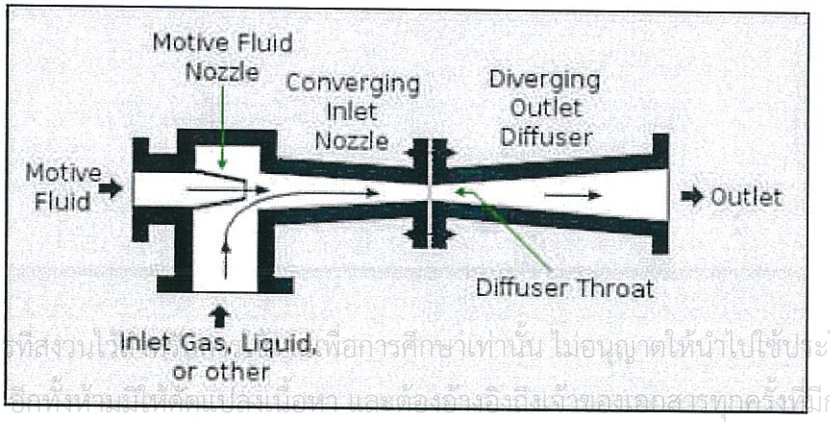
- การควบคุมวาล์วนี้ทำได้โดยการป้อนไฟฟ้าให้กับขดลวดโซลินอยด์ ดังภาพแสดงการทำงานขณะป้อนไฟฟ้าให้กับโซลินอยด์ด้านซ้ายมือ
- ถ้าป้อนไฟฟ้าให้กับโซลินอยด์ด้านขวามือ จะทำให้ลูกสูบเลื่อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.34 ตัวอย่างโซลินอยด์วาวล์

2.2.6 อุปกรณ์สำหรับงานสุญญากาศ(VACUUM EQUIPMENT)

เครื่อง/ตัวกำเนิดสุญญากาศ (Vacuum Ejector, Vacuum Generator) ตัวกำเนิดความดันสุญญากาศ Vacuum Ejector อาศัยหลักการที่ทำให้เกิดสุญญากาศตรงบริเวณคอคอดที่เป็นเสมือนท่อ 3 แยก Venturi ที่อยู่ภายในตัวกำเนิดความดันสุญญากาศตัว Ejector โดยบริเวณนี้เหมือนทาง 3 แพร่งในรูปข้างล่างลมเข้ามาทางรูที่ 1 และ เมื่อลมไหลเข้ามาเร็วในคอขาด ของตัวกำเนิดความดันสุญญากาศเช่น พื้นที่หน้าตัดที่ลมไหลผ่านลดลง 900% ความเร็วลมจะมากขึ้น 900% เช่นกัน พื้นที่ลดลง 9 เท่า ความเร็วลมที่เพิ่มขึ้น 9 เท่า ทำให้ลมที่ไหลผ่านที่ Venturi จะปั่นป่วนอย่างมาก (Turbulence) โดยลมจะไหลออกอย่างรวดเร็วไปสู่รูออกที่ 2 ซึ่งมีขนาดใหญ่ที่อยู่ตรงข้ามกับทางเข้าที่1 บริเวณทาง 3 แพร่ง ที่อยู่ภายใน Venturi ก็จะเกิดสุญญากาศ ดูดอากาศจากทางเข้าที่ 3 ท่อทางเข้าที่ 3 จะไปต่อกับลูกยางดูดจับชิ้นงาน ฯลฯ ในงานจับและวาง เครื่องกำเนิดความดันสุญญากาศที่ต้องการปริมาณลมดูดมากๆไม่เหมาะที่จะใช้ Jector ต้องใช้เป็นปั๊มสุญญากาศ ซึ่งมีทั้งแบบกระบอกสูบ แบบโรตารี่ แบบ Vane Type เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.35 หลักการทำงานตัวกำเนิดสูญญากาศ

รูปที่ 2.36 ตัวอย่างตัวกำเนิดสูญญากาศ

2.2.7 เซนเซอร์วัดความดัน (Pressure sensor)

เซนเซอร์วัดความดัน หรือ Pressure Sensor คืออุปกรณ์ที่สามารถวัดและควบคุมความดันที่มีหน่วยในการแสดงผลได้หลากหลายในตัวเดียว เช่น mbar, bar, kPa, kgf/cm2, psi, mmHg, inHg เป็นต้น และมีสัญญาณเอ๊าท์พุทแบบต่างๆให้เลือกใช้งาน เช่น NPN, PNP, 4-20 mA เป็นต้น เหมาะสำหรับลูกค้าทุกประเภทอุตสาหกรรม ไม่ว่าจะเป็นกลุ่มอุตสาหกรรมอิเล็กทรอนิกส์, อาหารและยา, เคมี, ยานยนต์ ฯลฯ

รูปที่ 2.37 ตัวอย่างเซ็นเซอร์วัดความดัน

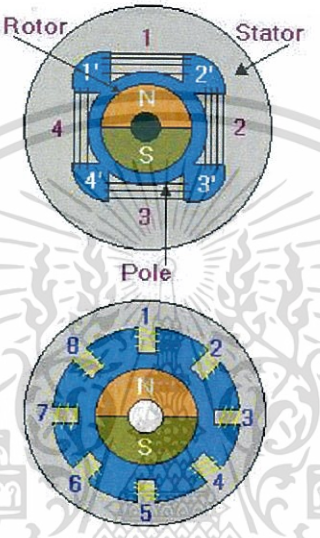
2.2.8 เสตปมอเตอร์ (Step Motor)

เป็นมอเตอร์ที่มีลักษณะเมื่อเราป้อนไฟฟ้าให้กับมอเตอร์ทำให้หมุนเพียงเล็กน้อยตามเส้นรอบวงและหยุด ซึ่งต่าง จากมอเตอร์ ทั่วไปที่จะหมุนทันทีและตลอดเวลาเมื่อป้อนแรงดันไฟฟ้าข้อดีของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

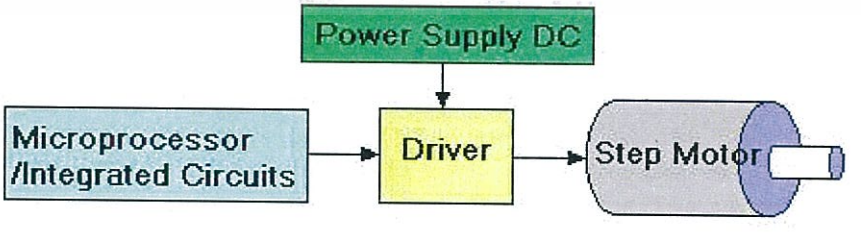
สเต็ปมอเตอร์ สามารถกำหนด ตำแหน่งของการหมุนด้วยตัวเลข(องศาหรือระยะทาง) ได้อย่างละเอียดโดย ใช้คอมพิวเตอร์หรือ ไมโครคอนโทรลเลอร์เป็น เครื่องกำหนดและจัดเก็บตัวเลข

โครงสร้างของขั้วแม่เหล็กบนสเตเตอร์ทำมาจากแผ่นเหล็กวงแหวนที่มีซี่ยื่นออกมาประกอบกันเป็นชั้นๆ โดยที่แต่ละซี่นั้นจะมีคอยล์(ขดลวด)พันสวมอยู่ เมื่อมีการป้อนกระแสผ่านคอยล์ทำให้เกิดสนามแม่ เหล็กไฟฟ้า(Electromagnetic) ดังรูป



รูปที่ 2.38 โครงสร้างของเสตปมอเตอร์

ลักษณะการนำไปใช้งาน สเต็ปมอเตอร์ ใช้งานลักษณะ ระบบเปิด(Open Loop System) คือ สเต็ปมอเตอร์สามาทำงานได้โดยไม่ต้องมีการ ป้อนค่าพารามิเตอร์กลับมา (Feed back) แต่ทุกวิธีที่ต้องการกำหนดตำแหน่งที่แน่นอนนั้น จะต้องการป้อนกลับไปยังระบบและตัวบอก ตำแหน่งว่าถูกต้องหรือผิดผลาดให้ทราบ



รูปที่ 2.39 หลักการทำงานของเสต็ปมอเตอร์

โดยแนวทางสเต็ปมอเตอร์เป็นอุปกรณ์จำพวกเชิงกลทางไฟฟ้า โดยมีกรุ๊ปของไบนารี่โวลท์เตทเป็นอินพุตและการเคลื่อนที่ แบบเชิงมุมเป็นเอาท์พุต หรือว่าหมุนทีละสเต็ปซึ่งอยู่ระหว่าง 0.1 - 30 องศา

เอกสารนี้มีลิขสิทธิ์ ห้ามนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ที่โครงสร้างของสเต็ปมอเตอร์ โดยตามสัญญาณ พัลซ์ที่จ่ายให้กับขดสเตเตอร์ทำให้เกิดแรงผลักแก่โรเตอร์หมุนไป สเต็ปมอเตอร์มีขดลวดหลายชุดในที่นี้เราเรียกว่า Phase (เฟส) ดังนั้นสัญญาณที่ต่อเนื่องเป็น Sequence(ซีเควน) ลักษณะของBinary(ไบนารี่) ซึ่งจะต้องไปผ่านวงจร Driver(ไดรเวอร์) ก็จะ ทำให้โรเตอร์หมุนไปอย่างต่อเนื่อง ดังรูปที่ 2.39

- การสั่งงานควบคุมการหมุนของสเต็ปมอเตอร์

การควบคุมและสั่งงานให้สเต็ปมอเตอร์ทำงาน ไปทีละสเต็ปสามารถทำได้โดยการจ่ายกำลังไฟไปยังขดลวด ในแต่ละขอบนสเตเตอร์ โดยการป้อนจะทำในลักษณะเป็นลำดับหรือเรียกว่าซีเควนเวียลในลูปที่ถูกต้อง ซึ่งจะแบ่งได้เป็น 3 รูปแแบ คือ แบบเวฟ(wave) แบบ 2 เฟส(2 phase) และแบบครึ่งสเต็ป (half step) ทั้ง 3 แบบนี้ก็จะมีข้อดีและข้อเสียต่างกันออกไป

- แบบเวฟ (wave)

จะเป็นการกระตุ้นแบบที่ง่ายที่สุด ซึ่งจะทำการกระตุ้นขดลวดทีละขดในเวลาหนึ่งๆเรียงกันไป ตัวอย่างเช่น ขดที่ 1 , 2 , 3 , 4 ,1, 2 , 3 , 4 เป็นลำดับอย่างนี้ หรือ ขด 1 , 4 , 3 , 2 , 1 , 4 , 3 , 2 เป็นลำดับกันไป ทั้งนี้ขึ้นอยู่กับทิศทางที่เราต้องให้มอเตอร์หมุนไป วงจรที่นำมากระตุ้นนั้นจะมีราคาค่อนข้างจะถูกกว่าและง่ายกว่า

Step No.	Phase 1	Phase 2	Phase 3	Phase 4
1	ON			
2		ON		
3			ON	
4				ON
5	ON			
6		ON		

ตารางที่ 2.3 ลำดับการทำงานเสต็ปมอเตอร์แบบเวฟ

- แบบ 2 เฟส(2 Phase)

นี้ก็จะคล้ายกับการกระตุ้นในแบบเวฟแต่จะต่างกันตรงที่ แบบ 2 เฟส จะกระตุ้นทีละ 2 ขด ที่อยู่ใกล้กันในเวลาเดียวกันและจะเรียงลำดับกันไป ตัวอย่างการกระตุ้นขดลวด ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12,23,34.41,12,23,34,41 เรียงลำดับกันไปเรื่อยๆ หรือจะเป็น 14,43,32,21,14,43,32,21

ข้อดี การที่จะเพิ่มจำนวนขดลวดที่ถูกกระตุ้นจะทำให้แรงบิดได้มากกว่า แบบเวฟ ซึ่งโรเตอร์จะหมุนด้วยแรง ดึงแบเต็มๆแรงจาก ทั้ง 2 ขดลวดที่กระตุ้นพร้อมกัน

ข้อเสีย แบบ 2 เฟส จะกระตุ้นขดลวดนั้นต้องใช้กำลังไฟมากขึ้นเป็น 2 เท่าของแบบเวฟ

Step No.	Phase 1	Phase 2	Phase 3	Phase 4
1	ON	ON		
2		ON	ON	
3			ON	ON
4	ON			ON
5	ON	ON		
6		ON	ON	

ตารางที่ 2.4 ลำดับการทำงานของเสต็ปมอเตอร์แบบ 2 เฟส

- แบบครึ่งสเต็ป

แบบนี้แบบรูปแบบผสมผสานของการกระตุ้นระหว่าง แบบเวฟ กับ แบบ 2 เฟส เพื่อให้จำนวนรอบของสเต็ปให้ มากขึ้นเป็น 2 เท่า ซึ่งในระบบนี้จะทำการกระตุ้นขดลวดเรียงกันไปเรื่อยๆเป็นลำดับ ดังจะยกตัวอย่างต่อไปนี้ 1,12,2,23,3,34,4,41,1,12,2,23,3,34,4,41,1 เป็นลำดับอยู่อย่างนี้เรื่อยไปครับ ถ้ากลับทิศทางการหมุนก็จะไดเป็น 1,41,4,43,3,32,2,21,1,41,4,43,3,32,2,21,1 เป็นลำดับ

ข้อดี การกระตุ้นแบบนี้จะให้แรงบิดที่เพิ่มมากขึ้น เนื่องจากช่วงสเต็ปที่มีระยะสั้นลงออีกประการหนึ่งแต่ละ สเต็ปเกิดแรงดึงจากขดลวด 2 ขดที่ถูกกระตุ้นพร้อมกันเป็นผลให้ค่าตำแหน่งความถูกต้องมากขึ้นไปด้วย

ข้อเสีย ก็คงจะเช่นเดียวกับแบบ 2 เฟสล่ะครับ ที่ต้องจ่ายกำลังไฟเป็น 2 เท่าของแบบเวฟหรือจะใช้เท่ากับแบบ 2เฟส นั้นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Step No.	Phase 1	Phase 2	Phase 3	Phase 4
1	ON	ON		
2	ON	ON		
3		ON	ON	
4			ON	
5			ON	ON
6				ON
7				ON
8	ON			
9	ON			
10	ON	ON		

ตารางที่ 2.5 ลำดับการทำงานของเสต็ปมอเตอร์แบบครึ่งเสต็ป

2.2.9 การใช้งานพอร์ตอนุกรม RS232

การสื่อสารแบบอนุกรมนับว่ามีความสำคัญต่อการใช้งานไมโครคอนโทรลเลอร์มาก เพราะสามารถใช้แป้นพิมพ์และจอภาพของคอมพิวเตอร์เป็นอินพุตและเอาต์พุตในการติดต่อหรือควบคุมไมโครคอนโทรล- เลอร์ด้วยสัญญาณอย่างน้อยเพียง 3 เส้นเท่านั้น คือ

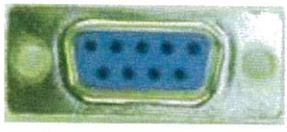
-สายส่งสัญญา

- สายรับสัญญาณ RX
- และสาย GND

โดยปกติพอร์ตอนุกรม RS-232C จะสามารถต่อสายได้ยาว 50 ฟุตโดยประมาณ ขึ้นอยู่กับชนิดของ สายสัญญาณ, ระยะทาง, และ ปริมาณ สัญญาณ รบกวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

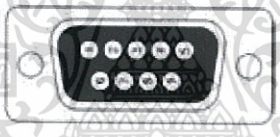
พอร์ตอนุกรมของ PC DB9 ตัวผู้ (Male)



พอร์ตอนุกรมของอุปกรณ์ภายนอกDB9ตัวเมีย (Female)

- พอร์ตอนุกรมของ PC จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวผู้ (Male)
- พอร์ตอนุกรม ของอุปกรณ์ภายนอก จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวเมีย (Female)

- แสดงการจัดขา ของคอนเน็กเตอร์ อนุกรมแบบ DB9 และหน้าที่การใช้งานต่างๆ



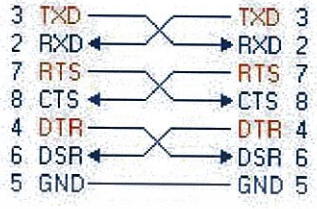
DB9 ตัวผู้ เมื่อมองจากด้านหลัง

Pin	Description	Type
1	Data Carrier Detect (DCD)	Input
2	Received Data (RXD)	Input
3	Transmitted Data (TXD)	Output
4	Data Terminal Ready (DTR)	Output
5	Signal Ground (GND)	Input
6	Data Set Ready (DSR)	Input
7	Request To Send (RTS)	Output
8	Clear to Send (CTS)	Input
9	Ring Indicator (RI)	Input

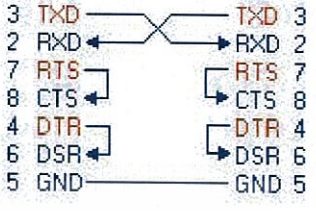
ตาราง 2.6 I/O ของ RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การเชื่อมต่ออุปกรณ์อุกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย DB9



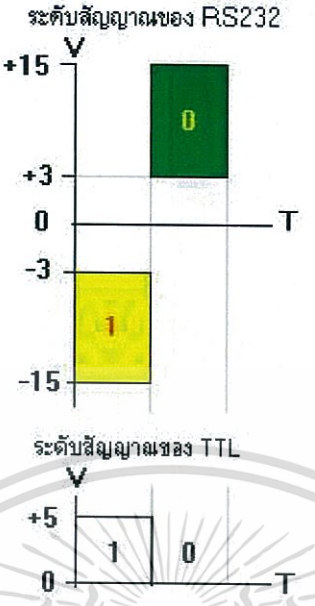
การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB9 แบบ 3 เส้น



การต่ออุปกรณ์ภายนอกผ่าน DB9 แบบ Null modem

- การทำงานของขาสัญญาณ DB9
  - TXD เป็นขาที่ใช้ส่งข้อมูล
  - RXD เป็นขาที่ใช้รับข้อมูล
  - DTR แสดงสภาวะพอร์ตว่าเปิดใช้งาน ,DSR ตรวจสอบว่าพอร์ต ที่ติดต่อด้วย เปิดอยู่หรือไม่
    - เมื่อเปิดพอร์ตอนุกรม ขา DTR จะ ON เพื่อให้อุปกรณ์ได้รับทราบว่าต้องการติดต่อด้วย
    - ในขณะเดียวกันก็จะตรวจสอบขา DSR ว่าอุปกรณ์พร้อมหรือไม่
  - RTS แสดงสภาวะพอร์ตว่าต้องการส่งข้อมูล ,CTS ตรวจสอบว่าพอร์ตที่ติดต่ออยู่ต้องการส่งข้อมูลหรือไม่
    - เมื่อต้องการส่งข้อมูลขา RTS จะ ON และจะส่งข้อมูลออกที่ขา TXD เมื่อส่งเสร็จก็จะ OFF
    - ในขณะเดียวกันก็จะตรวจสอบขา CTS ว่าอุปกรณ์ต้องการที่จะส่งข้อมูลหรือไม่
  - GND ขา ground
- ระดับสัญญาณของ RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.40 ระดับสัญญาณของ RS232C และระดับสัญญาณของ TTL

สัญญาณรบกวนที่เกิดขึ้น ในสายนำสัญญาณ มักจะมีแรงดันเป็นบวก เมื่อเทียบกับกราวน์

- เพื่อป้องกันสัญญาณรบกวนนี้ จึงออกแบบแรงดัน ของโลจิก "1" เป็นลบ คืออยู่ในช่วง -3V ถึง -15V ส่วนแรงดัน ของโลจิก "0" อยู่ในช่วง +3V ถึง +15V
- และเหตุที่ ระดับสัญญาณ ของ RS232 อยู่ในช่วง +15V ถึง -15V ก็เพื่อให้ต่อสายสัญญาณไปได้ไกลขึ้น

ดังนั้นจึงจำเป็นจะต้องมีวงจรเปลี่ยนระดับแรงดันของ RS232 มาเป็นระดับแรงดันของ TTL

- อัตราการส่งข้อมูล (Baud rate)

  - คือความเร็วของการรับ-ส่งข้อมูล เป็นจำนวนบิทต่อวินาทีเช่น 300, 1,200, 2,400, 4,800 , 9,600 ,14,400 ,19,200, 38,400 ,56,000 เป็นต้น

  - การเลือกอัตราการส่งข้อมูลขึ้นอยู่กับ ชนิดของสายสัญญาณ, ระยะทาง,และปริมาณสัญญาณรบกวน

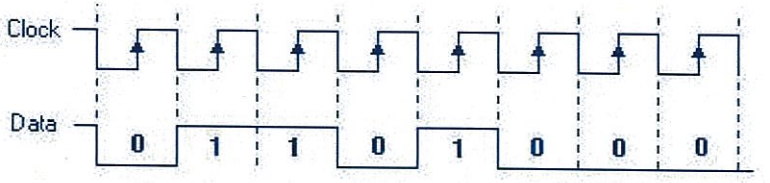
- รูปแบบการสื่อสารแบบอนุกรม

มีด้วยกันอยู่ 2 แบบ คือแบบซิงโครนัส (Synchronous) และแบบอะซิงโครนัส (Asynchronous)

- การสื่อสารแบบซิงโครนัส (Synchronous)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับส่งข้อมูลจะมีสัญญาณนาฬิกาซึ่งเป็นตัวกำหนดจังหวะเวลาการส่งข้อมูลร่วมอยู่ด้วยอีกเส้นหนึ่งใช้คู่กับสัญญาณข้อมูลตัวอย่างเช่น การส่งสัญญาณจากคีย์บอร์ด



รูปที่ 2.41 ระดับสัญญาณของการสื่อสารแบบซิงโครนัส

- การสื่อสารแบบอะซิงโครนัส (Asynchronous)

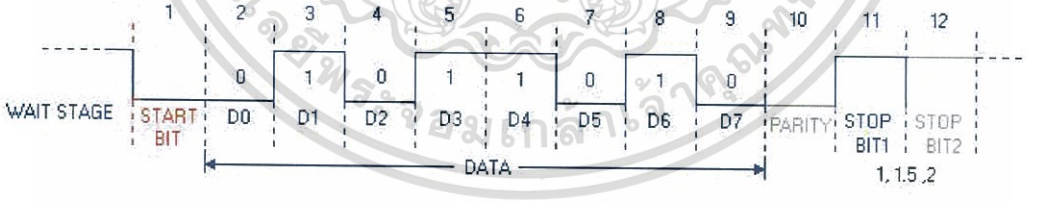
การรับส่งข้อมูลโดยที่ไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วยแต่จะใช้ให้ตัวส่งและตัวรับ มีอัตราส่งข้อมูล ที่เท่ากันรูปแบบข้อมูลแบบอะซิงโครนัส ประกอบด้วย 4 ส่วนคือ

1.บิตเริ่มต้น (Start bit) มีขนาด1 บิต

2.บิตข้อมูล (Data) มีขนาด 5,6,7 หรือ 8 บิต

3.บิตตรวจสอบพาริตี้ (Parity bit) มีขนาด 1 บิตหรือไม่มี

4.บิตหยุด (Stop bit) มีขนาด 1, 1.5, 2 บิต



รูปที่ 2.42 ระดับสัญญาณของการสื่อสารแบบอะซิงโครนัส

- เมื่อไม่มีการส่งข้อมูล ขา data จะมีสถานะเป็นโลจิก "1" หรือ สถานะหยุดรอ (Waiting stage)
- เมื่อเริ่มต้นส่งข้อมูลจะให้ขา data เป็นโลจิก "0" เป็นจำนวน 1 บิต เรียกว่าบิตเริ่มต้น (Start bit)
- จากนั้นก็จะเริ่มต้นส่งข้อมูล โดยส่งบิตต่ำไปก่อน (LSB)
- แล้วตามด้วยพาริตี้บิต (จะมีหรือไม่ก็ได้ ขึ้นอยู่กับการติดตั้งค่า ของทั้งสองฝ่าย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สุดท้ายตามด้วยโลจิก "1" อย่างน้อย 1 บิต (มีขนาด 1, 1.5, หรือ 2 บิต) เพื่อแสดงว่าสิ้นสุดข้อมูล

- การรับและส่งข้อมูลแบบอนุกรมยังแบ่งออกเป็นลักษณะการใช้งานได้ 3 แบบคือ
  1. แบบซิมเพลกซ์ (Simplex) เป็นการส่ง หรือรับข้อมูล แบบทิศทางเดียว เท่านั้น
  2. แบบฮาล์ฟดูเพลกซ์ (Half Duplex) เป็นการส่งและรับข้อมูลแบบสลับกัน

  คือเมื่อด้านหนึ่งส่ง อีกด้านหนึ่ง เป็นฝ่ายรับ สลับกัน ไม่สามารถรับ-ส่งในเวลาเดียวกันได้

  3. แบบฟลูดูเพลกซ์ (Full Duplex) สามารถรับ-ส่งข้อมูลในเวลาเดียวกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีดำเนินการวิจัย

3.1 แนวทางการพัฒนาสายการผลิต

โดยปกติแล้วการขั้นตอนการประกอบเอคทานอล มีด้วยกันหลายขั้นตอนและขั้นตอนที่ต้องการจะพัฒนาคือ ขั้นตอนในการติดแผ่นกันกระแทก(Sponge) เพราะว่าแผ่นกันกระแทกนั้นยังต้องทำการติดโดยใช้พนักงาน และแผ่นกันกระแทกมีขนาดเล็กและเรียงติดกันมาก ทำให้พนักงานปฏิบัติงานยากและเกิดการติดแผ่นกันกระแทกไม่มีความแม่นยำ ซึ่งขั้นตอนในการดำเนินการวิจัยนั้นมีด้วยกันดังนี้ คือ

3.1.1 ออกแบบแผนผังกระบวนการทำงาน

3.1.2 การวาดเลเอาท์และการจัดวางอุปกรณ์ของเครื่องจักร

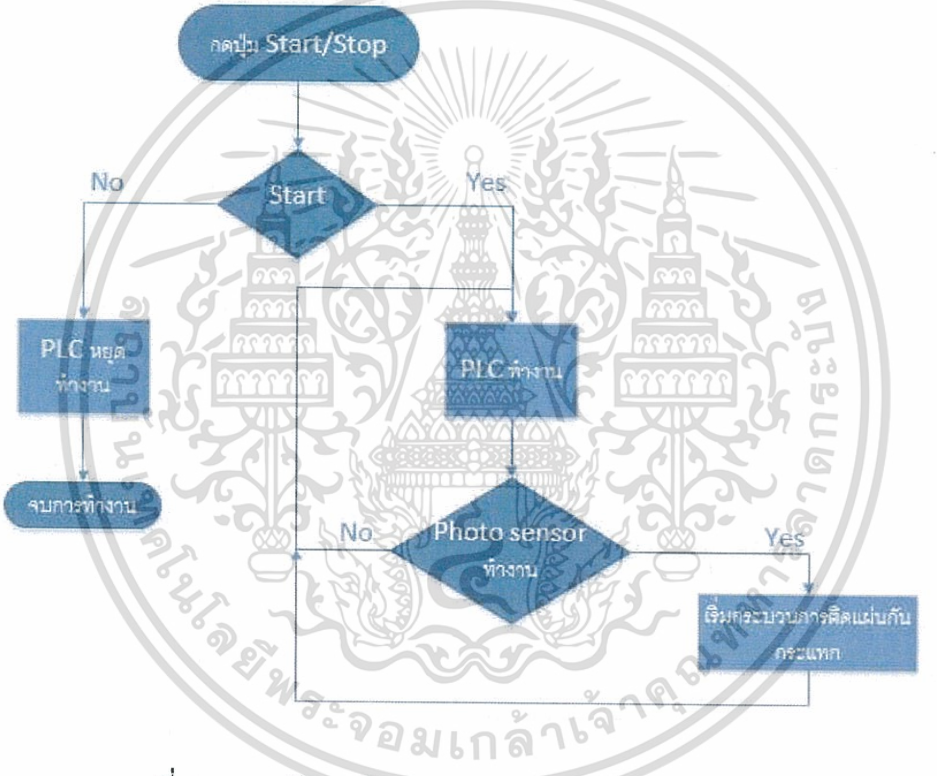
3.1.3 การควบคุมเสต็ปมอเตอร์

3.2 แผนผังกระบวนการทำงาน

ก่อนการประดิษฐ์เครื่องนั้นจะต้องมีการวางแผนและกระบวนการทำงานของเครื่องเพื่อความสะดวกและไม่สับสนกับงาน ดังรูป 3.1 จะเป็นแผนผังกระบวนการทำงานในส่วนของพีแอลซีเมื่อกดปุ่มเริ่มการทำงานจะมีเงื่อนไขโดยโปรแกรมจะเช็คว่าเครื่องเริ่มการทำงานจริงหรือไม่ จริงก็จะดำเนินงานต่อไป ถ้าไม่จริงก็จะหยุดการทำงานของเครื่อง โดยเมื่อทำงานต่อไปจะมีเงื่อนไขอีกทางหนึ่งคือ Photo Sensor ทำงานหรือไม่ ถ้าทำงานเครื่องก็จะเริ่มกระบวนการติดอผ่นกันกระแทก และวนกลับไปเช็ค Photo Sensor เป็นลูปไปเรื่อยๆจนกว่าจะมีการกดปุ่มหยุดการทำงาน แต่ถ้า Photo Sensor ไม่ทำงานก็จะวนกลับไปเช็ค Photo Sensor เช่นกัน และรูปที่ 3.2 จะเป็นแผนผัง

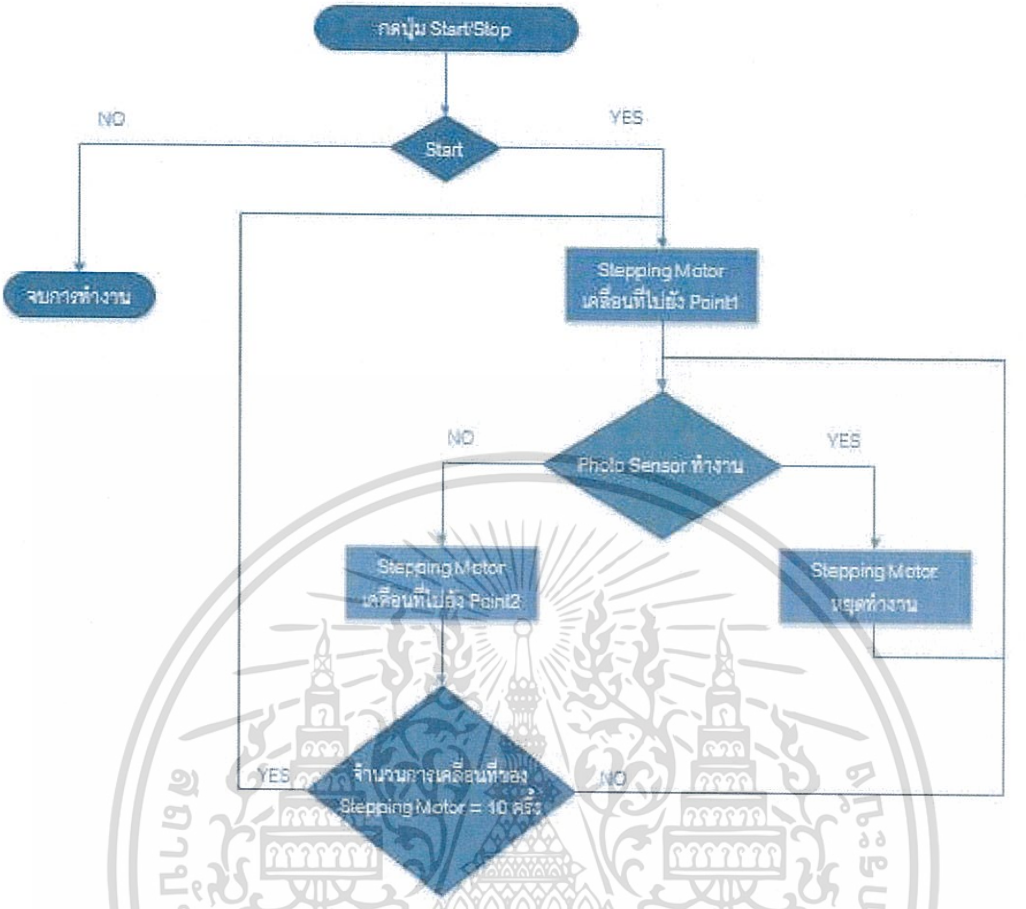
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการทำงานในส่วนของเสต็ปมอเตอร์โดยเมื่อกดปุ่มเริ่มการทำงานก็จะมีเงื่อนไขว่าเครื่องจักรได้กดปุ่มเริ่มการทำงานหรือไม่ ถ้าใช่เสต็ปมอเตอร์ก็จะเคลื่อนที่ไปยังตำแหน่งที่ 1 และจะมีเงื่อนไขเช็คว่า Photo Sensor ทำงานหรือไม่ ถ้าทำงานเสต็ปมอเตอร์ก็จะหยุดการเคลื่อนที่ แต่ถ้าไม่ทำงานในเวลาที่กำหนด เสต็ปมอเตอร์ก็จะเลื่อนไปตำแหน่งถัดไป และจะมีเงื่อนไขเช็คว่าเสต็ปมอเตอร์เคลื่อนที่ไป 10 ครั้งหรือยังถ้าครบแล้วมอเตอร์ก็จะวิ่งกลับไปที่ตำแหน่งที่ 1 และเริ่มการทำงานลูปใหม่อีกครั้ง แต่ถ้ายังไม่ครบ 10 ครั้งวนไปยังการเช็คสถานะการทำงานของ Photo Sensor และวนลูปไปเรื่อยๆ



รูปที่ 3.1 แผนผังการทำงานระบบ PLC ของ Auto Sponge

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

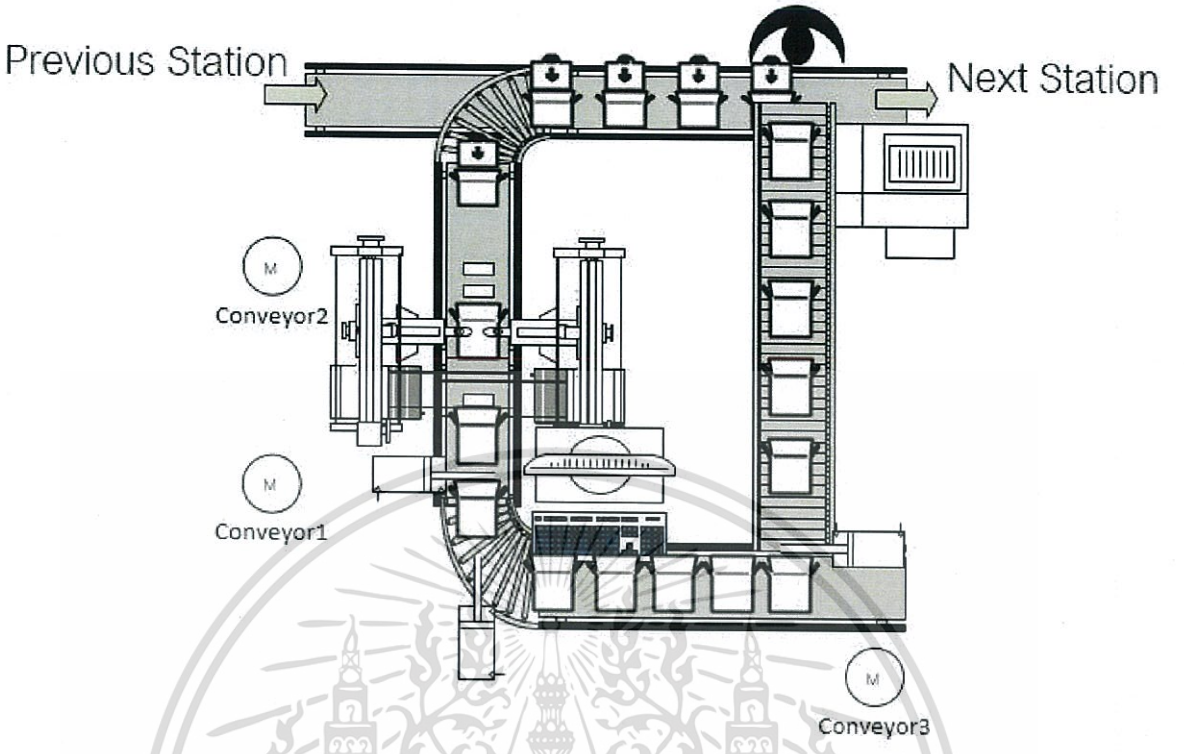


รูปที่ 3.2 แผนผังการทำงานระบบ Stepping Motor ของ Auto Sponge

3.3 การวาดเลเอาท์

เมื่อเราได้ทำการเขียนแผนผังระบบการทำงานของเครื่องจักรมาดังข้างต้นแล้ว ขั้นตอนในการปฏิบัติต่อไปก็คือจะต้องทำการวาดเลเอาท์ของเครื่องจักรเพื่อที่จะทำให้สะดวกต่อการเลือกอุปกรณ์และการจัดวางอุปกรณ์ให้เหมาะสมกับการทำงาน ดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 เลเอาท์ของเครื่อง Auto Sponge

3.4 การกำหนดเอาท์พุทและอินพุท

เมื่อทำการออกแบบลักษณะการทำงานและระบบของเครื่อง Auto Sponge แล้ว ขั้นตอนต่อไปทำให้เราสามารถคำนวณเอาท์พุทและอินพุทใน PLC โดยเราจะกำหนดเอาท์พุทและอินพุทจากอุปกรณ์ที่เราวางเลเอาท์ และออกแบบลักษณะการทำงานของเครื่องเอาไว้ เพื่อเป็นการประหยัดงบประมาณและสร้างเค้าโครงของงาน ในการคำนวณเอาท์พุทและอินพุทจึงมีประโยชน์อย่างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Batch monitor

Device	Current value	Display format	Set value	Contact	Comments
R000	-	1-bit BIN			SW_START
R001	-	1-bit BIN			By_Pass
R002	-	1-bit BIN			
R003	-	1-bit BIN			
R004	-	1-bit BIN			Axis3_out
R005	-	1-bit BIN			Axis3_in
R006	-	1-bit BIN			Axis6_Left_in
R007	-	1-bit BIN			Axis6_Left_out
R008	-	1-bit BIN			Axis7__Left_UP
R009	-	1-bit BIN			Axis7_Left_Down
R010	-	1-bit BIN			Axis7_Right_UP
R011	-	1-bit BIN			Axis7_Right_Down
R012	-	1-bit BIN			Axis6_Right_out
R013	-	1-bit BIN			Axis6_Right_in
R014	-	1-bit BIN			Presure_Left
R015	-	1-bit BIN			Presure_Right
R100	-	1-bit BIN			Photo_right
R101	-	1-bit BIN			Photo_left
R102	-	1-bit BIN			Sensor_1
R103	-	1-bit BIN			Sensor_2
R104	-	1-bit BIN			

Batch monitor

Device	Current value	Display format	Set value	Contact	Comments
R200	-	1-bit BIN			Sensor4
R201	-	1-bit BIN			Sensor3
R202	-	1-bit BIN			Sensor5
R203	-	1-bit BIN			

รูปที่ 3.4 การกำหนดอินพุทใน PLC

Batch monitor

Device	Current value	Display format	Set value	Contact	Comments
R500	-	1-bit BIN			Axis1
R501	-	1-bit BIN			Axis2
R502	-	1-bit BIN			Axis3
R503	-	1-bit BIN			Axis4
R504	-	1-bit BIN			Axis5
R505	-	1-bit BIN			Axis6_Left
R506	-	1-bit BIN			Axis7_Left
R507	-	1-bit BIN			Axis8_Left
R508	-	1-bit BIN			
R509	-	1-bit BIN			
R510	-	1-bit BIN			
R511	-	1-bit BIN			
R512	-	1-bit BIN			
R513	-	1-bit BIN			
R514	-	1-bit BIN			
R515	-	1-bit BIN			
R600	-	1-bit BIN			Motor1_con
R601	-	1-bit BIN			Motor2_con
R602	-	1-bit BIN			Motor3_Left_Feed
R603	-	1-bit BIN			Motor4_Right_Feed
R604	-	1-bit BIN			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

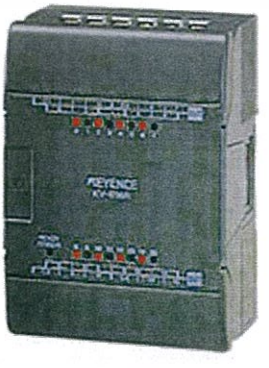
Batch monitor

Device	Current value	Display format	Set value	Contact	Comments
R702	-	1-bit BIN			Alarm_red
R703	-	1-bit BIN			Alarm_Green
R704	-	1-bit BIN			
R705	-	1-bit BIN			
R706	-	1-bit BIN			
R707	-	1-bit BIN			
R708	-	1-bit BIN			
R709	-	1-bit BIN			
R710	-	1-bit BIN			
R711	-	1-bit BIN			
R712	-	1-bit BIN			
R713	-	1-bit BIN			
R714	-	1-bit BIN			
R715	-	1-bit BIN			
R800	-	1-bit BIN			Ramp_ST
R801	-	1-bit BIN			Ramp_ByPass
R802	-	1-bit BIN			Vacume_Left
R803	-	1-bit BIN			Vacume_Right
R804	-	1-bit BIN			
R805	-	1-bit BIN			Axis6_Right
R806	-	1-bit BIN			Axis7_Right
R807	-	1-bit BIN			Axis8_Right
R808	-	1-bit BIN			
R809	-	1-bit BIN			

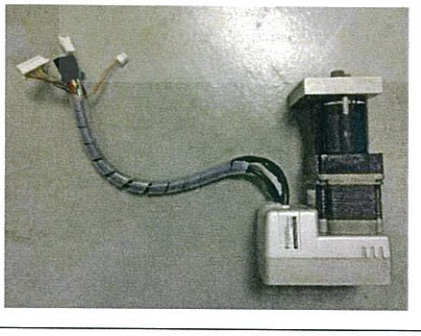
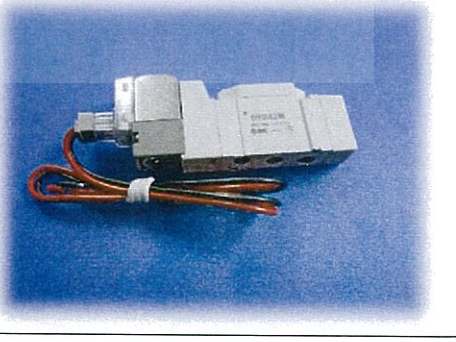
รูปที่ 3.5 การกำหนด Output ใน PLC

3.5 อุปกรณ์และการเดินสาย

จากที่ได้ทำการออกแบบเลเอาท์และกำหนด Input Output ของเครื่องจักรดังข้างต้นแล้ว จึงมีการจัดเตรียมอุปกรณ์ในการควบคุมเครื่องจักร ทำการเบิกอุปกรณ์ที่ต้องใช้ตามการออกแบบไว้ดังตาราง

- PLC (KV-24DR) 1 ตัว	- PLC (KV-E16X) 2 ตัว 
- PLC (KV-E16R) 2 ตัว	- PHOTOELETRIC SENSOR (FS-N11N) 2 ตัว

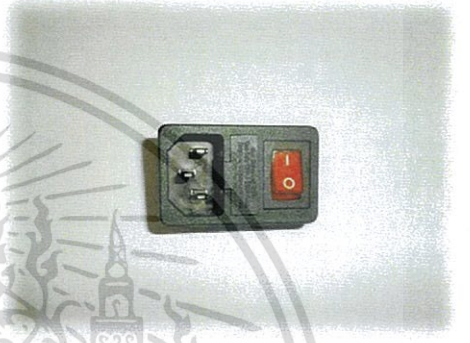
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<table>
<tr><td> </td><td> </td></tr>
<tr><td>- FIBER PHOTOELECTRIC (FU-35FZ) 2 ตัว
 </td><td>- PHOTOELECTRIC SENSOR (PZ-M11) 7 ตัว
 </td></tr>
<tr><td>- STEPPING MOTOR 2 ตัว
  </td><td>- SOLENOID VALVE (SY3120-5LZ-M5) 14 ตัว
  </td></tr>
<tr><td>- SOLENOID VALVE (VQ21A1-5Y-C6-F) 2 ตัว</td><td>- VACUUM PUMP 2 ตัว</td></tr>
</table>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- VACUUM PAD 2 ตัว

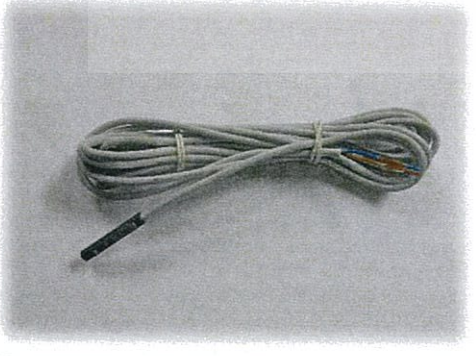
- PLUG +SWITCH +FUSE 1 ตัว



- SWITCHING POWER SUPPLY 3 ตัว

- PRESSURE SWITCH 2 ตัว

- REED SWITCH 6 ตัว

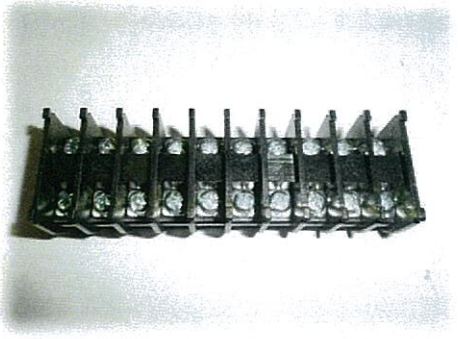
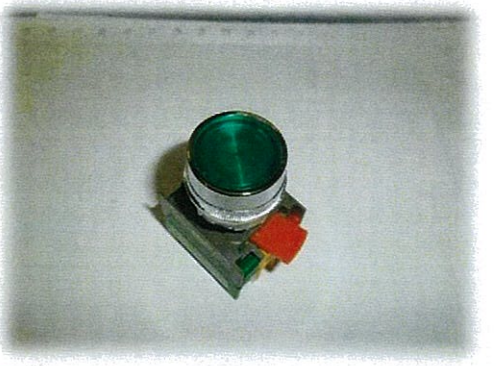
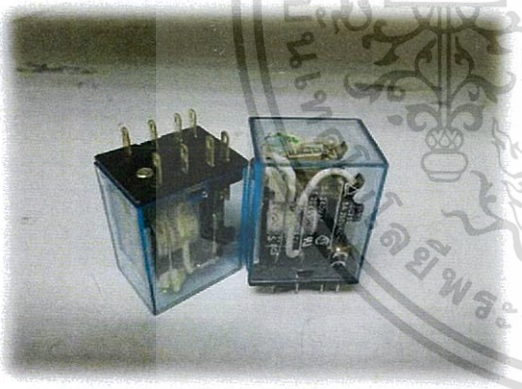


- TOGGLE SWITCH 1 ตัว

- TERMINAL 1 ตัว

- PUSH BOTTON 2 ตัว

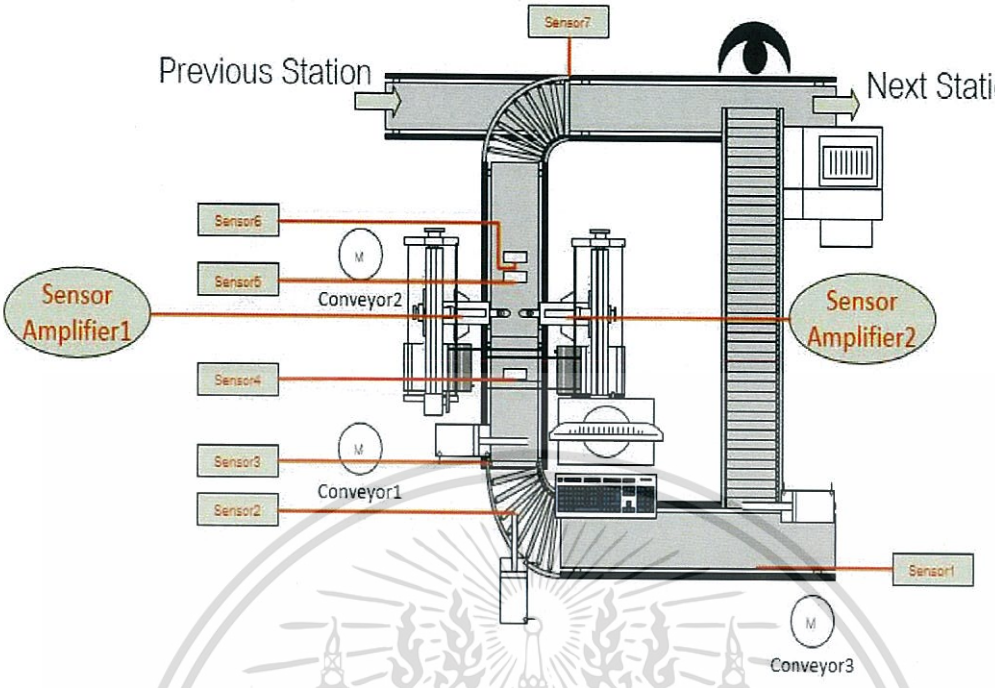
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	
- TOWER LIGHT LED 1 ตัว 	- MOTOR SUNTECH 5 ตัว
- RELAY 8 ตัว 	- RELAY - AGV CAR 8 ตัว

ตารางที่ 3.1 อุปกรณ์ดำเนินงาน

เมื่อทราบจำนวนของอุปกรณ์ดังกล่าวแล้ว จึงจัดวางอุปกรณ์เซ็นเซอร์ตามลักษณะการทำงานของเครื่องจักร ดังรูปที่ 3.6

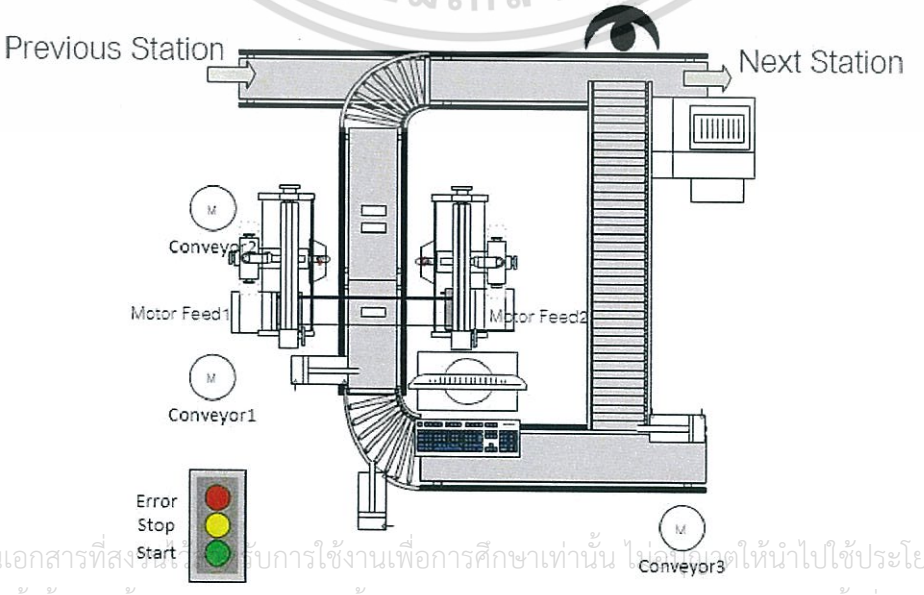
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 การจัดวางตำแหน่งเซ็นเซอร์

3.6 ลักษณะการทำงานของเครื่อง Auto Sponge

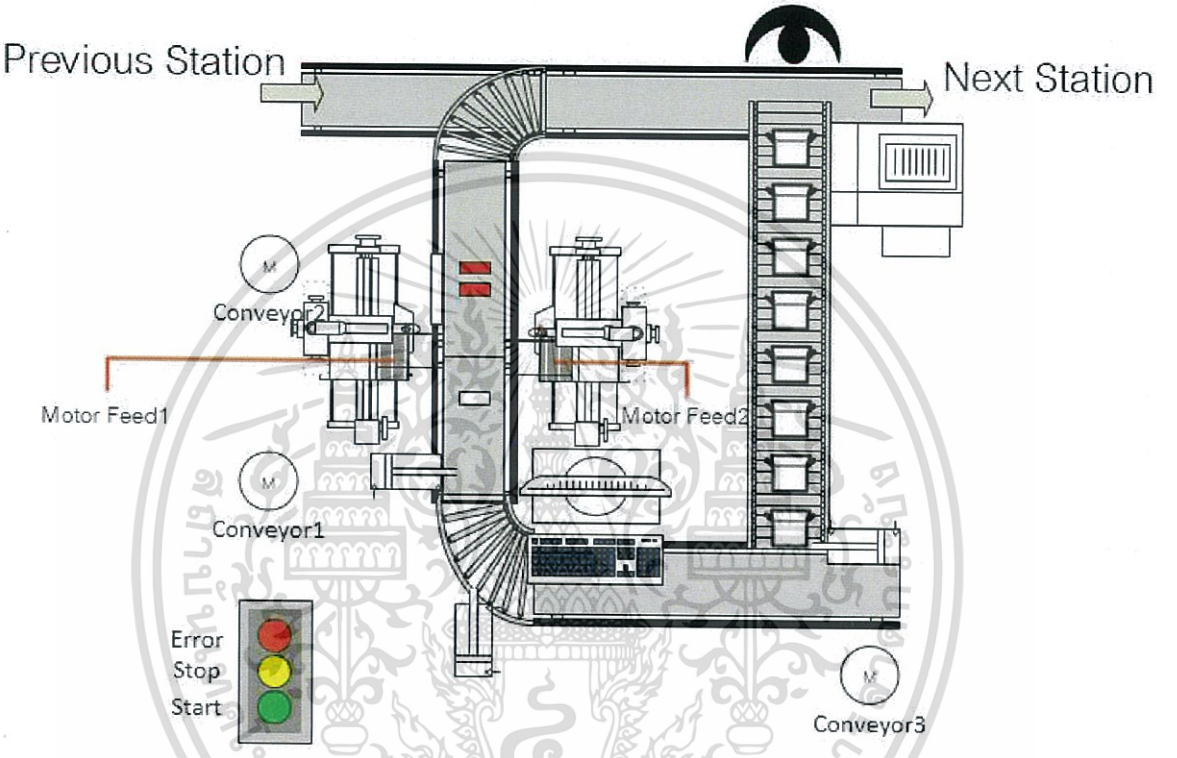
ในการดำเนินงานขั้นต่อไปหลักจากที่ติดตั้งอุปกรณ์เสร็จแล้ว คือการออกแบบลักษณะการทำงานของเครื่องจักรก่อนที่จะทำการเขียนโปรแกรม เพื่อความสะดวกและไม่สับสนในอนาคตโดยทำการออกแบบลักษณะการทำงานดังต่อไปนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.7 สถานะเริ่มต้นการทำงานของเครื่อง Auto Sponge

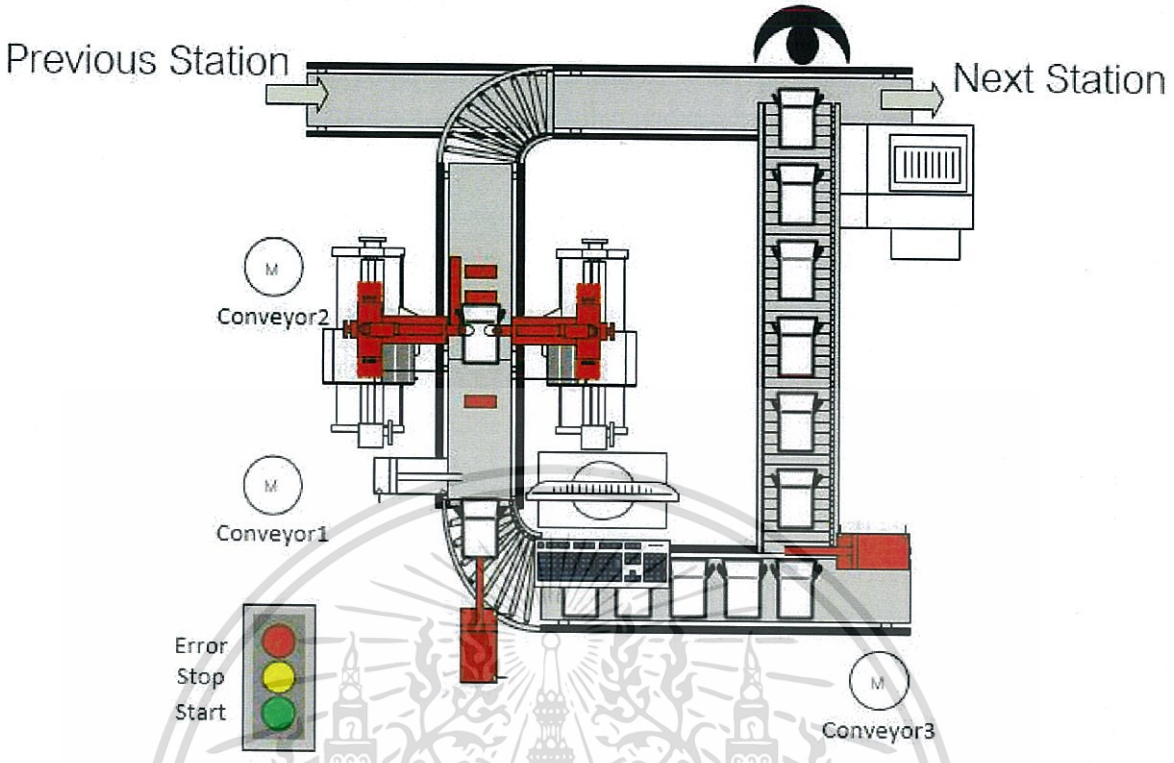
สถานะเริ่มต้นของเครื่องจะอยู่ดังรูปที่ 3.7 จะมีทาวเวอร์ไลท์เป็นตัวบอกสถานะการทำงานของเครื่อง Auto Sponge ดังนี้ สีเขียวบ่งบอกสถานะว่าเครื่องกำลังทำงาน สีเหลืองบ่งบอกสถานะว่าเครื่องหยุดทำงาน สีแดงบ่งบอกสถานะว่าเครื่องขัดข้องจะส่งสัญญาณเรียกผู้เชี่ยวชาญเข้ามาแก้ไข



รูปที่ 3.8 ภาพจำลองการทำงานของเครื่องเมื่อเริ่มกดปุ่ม Start/Stop

จากรูปที่ 3.8 เริ่มกดปุ่ม Start/Stop มอเตอร์เสตปทั้งสองด้านถูกสั่งให้เคลื่อนที่ไปยังตำแหน่งที่1 เมื่อมอเตอร์เสตปหยุดเคลื่อนที่แล้วมอเตอร์Feed ทั้งสองด้านเริ่มหมุนให้แผ่นกันกระแทกออกมาถึงเซนเซอร์แอมพลิฟายเออร์ เมื่อเซ็นเซอร์แอมพลิฟายเออร์ทำงานทำให้มอเตอร์ Feed ทั้งสองด้านหยุดหมุนแล้วตำแหน่งของเซ็นเซอร์แอมพลิฟายเออร์จะหดเข้าไปเพื่อทำให้มั่นใจว่าเวลามอเตอร์เสตปเคลื่อนที่ไปยังตำแหน่งต่างๆ จะสามารถตรวจจับแผ่นกันกระแทกได้อย่างแน่นอน ในขณะเดียวกันนั้นสตอปเปอร์สีแดงทั้งสองก็เริ่มทำงานเช่นเดียวกัน

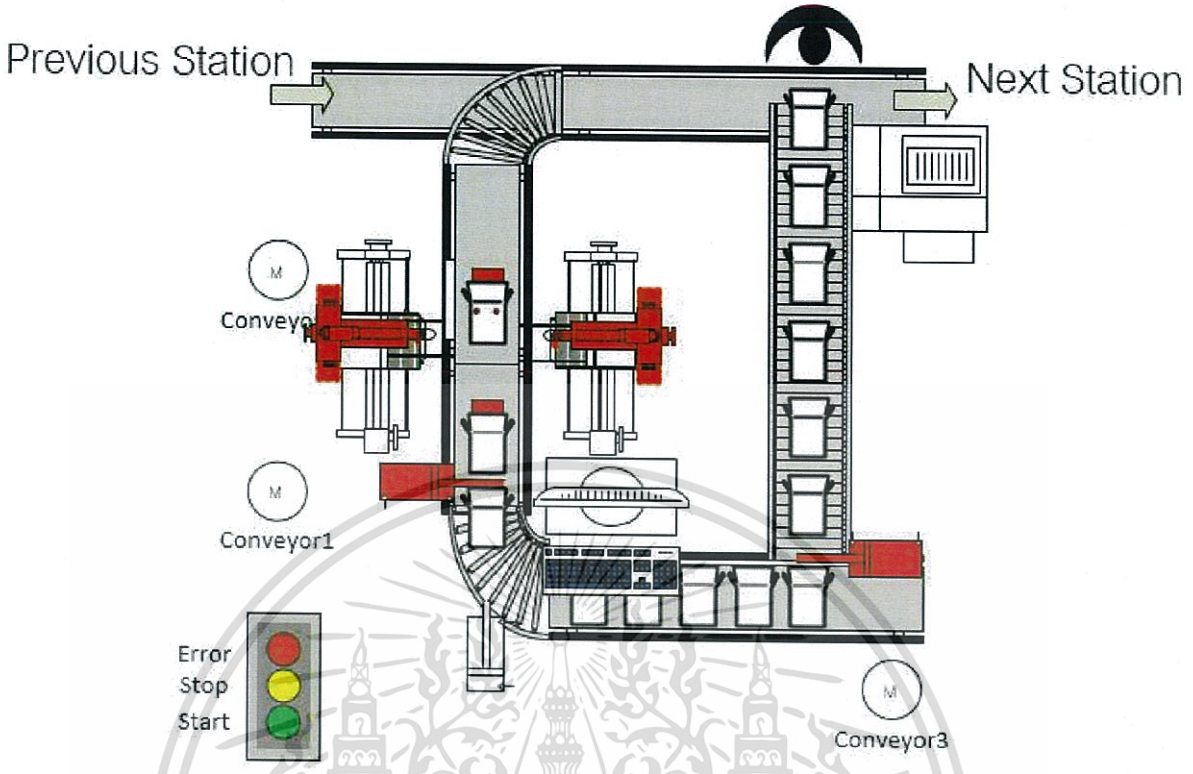
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 ภาพจำลองการทำงานของเครื่อง Auto Sponge

จากรูปที่ 3.9 เป็นภาพจำลองขณะที่ชิ้นงานที่ใส่โดยพนักงานเคลื่อนที่ด้วย Conveyor3 มาจนถึง Sensor2 ทำให้กระบอกสูบที่สองดันชิ้นงานเข้าไปที่ Conveyor1 และเมื่อถึง Sensor5 ทำงานเป็นเวลาที่ตั้งไว้จะทำให้ Stopper1 ทำงานเพื่อไม่ให้ชิ้นงานเข้าเกินหนึ่งชิ้นงานและConveyor2 จะหยุดหมุน ในขณะเดียวกันนั้นกระบอกสูบที่สี่ก็จะหนีบชิ้นงานเอาไว้ไม่ให้เคลื่อนที่และเครื่องก็ทำการดูดแผ่นกันกระแทกขึ้นมาครบทั้งสองด้านจึงจะติดลงบนชิ้นงาน(กรณีดูดชิ้นงานไม่ติดจะทำการดูดซ้ำจนถึงจำนวนครั้งที่กำหนดไว้และจะส่งสัญญาณให้ผู้เชี่ยวชาญเข้ามาแก้ไขต่อไป) และเมื่อมีชิ้นงานมารอเต็ม Conveyor3 จะใช้ Sensor1 ในการสั่งงาน กระบอกสูบที่หนึ่งทำงานเพื่อไม่ให้ชิ้นงานเข้ามาซ้อนกัน

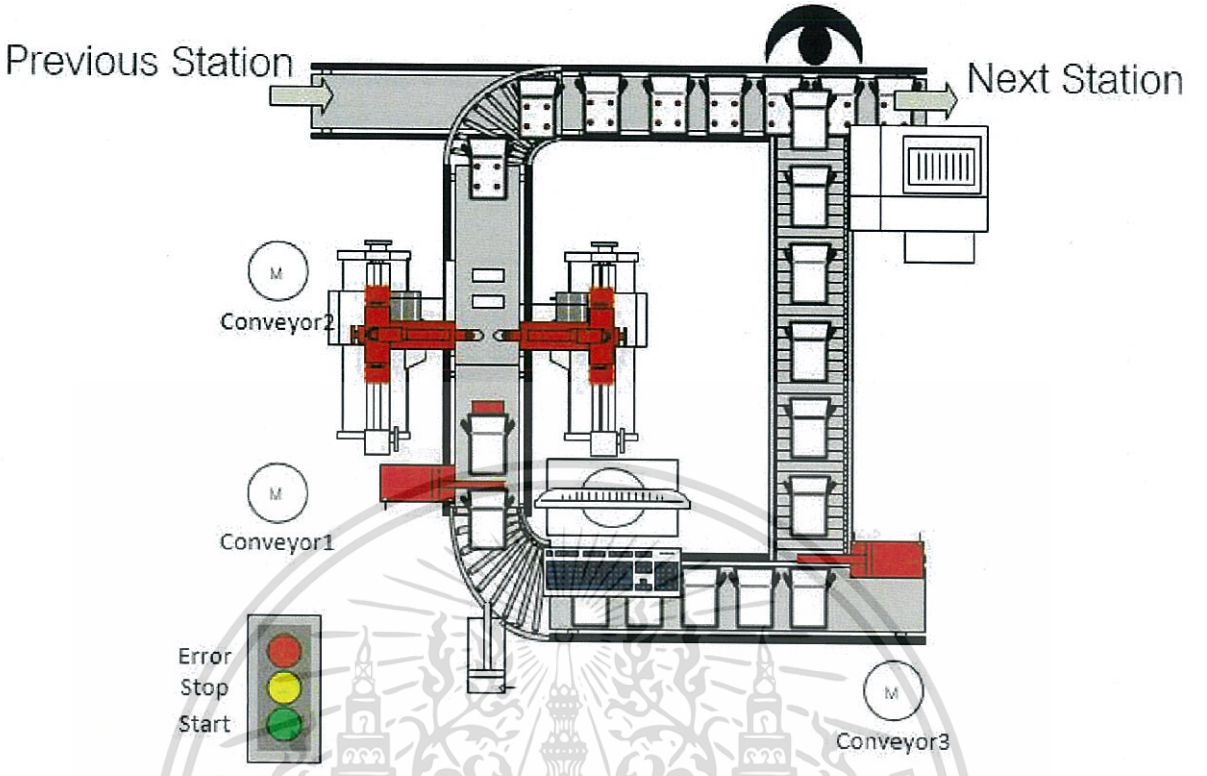
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 ภาพจำลองการทำงานของเครื่อง Auto Sponge

จากรูปที่ 3.10 เมื่อทำการติดแผ่นกันกระแทกลงบนชิ้นงานทั้งสองด้านแล้ว Stopper2 และ Stopper3 จะหยุดทำงานแล้ว Conveyor2 ก็จะหมุนทำให้ชิ้นงานเคลื่อนที่ไปยัง Sensor6 จากนั้น Stopper2 ก็จะหนีบชิ้นงานและ Conveyor2 ก็จะหยุดมันอีกครั้งเพื่อให้ติดแผ่นกันกระแทก และในขณะที่เซ็นเซอร์แอมพลิฟลายเออร์ตรวจจับแผ่นกันกระแทกไม่เจอเนื่องจากว่าดูดไปติดแล้วนั้น มอเตอร์เสตปก็ทำการเคลื่อนที่ไปยังตำแหน่งต่อไปและเมื่อตรวจจับเจอแผ่นกันกระแทกก็จะดูดแล้วนำไปติดที่ชิ้นงานอีกครั้ง จะสังเกตุว่าเมื่อ Stopper1 ทำงานจะทำให้ Sensor4 ทำงานจนครบเวลาที่กำหนดไว้แล้วกระบอกสูบที่สามทำงานเพื่อไม่ให้ชิ้นงานเข้าไปได้ และเมื่อ Sensor3 ทำงานจนครบเวลาที่กำหนดไว้แล้ว Conveyor3 ก็จะหยุดหมุนเพื่อไม่ให้กระบอกสูบที่สามพลักชิ้นงานชนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 ภาพจำลองการทำงานของเครื่อง Auto Sponge

จากรูปที่ 3.11 เมื่อติดแผ่นกันกระแทกบนชิ้นงานครบตามต้องการแล้ว Stopper2 และ Stopper4 จะหยุดทำงานและ Conveyor2 ก็จะหมุนให้ชิ้นงานออกไป เป็นอันจบขั้นตอนสำหรับชิ้นงานหนึ่งชิ้น และจากนั้น ถ้า Sensor5 และ Sensor6 ตรวจจับไม่เจอเป็นเวลาตามที่กำหนด Stopper3 และ Stopper4 ก็จะทำงานดังขั้นตอนเริ่มต้น เมื่อ Stopper ทั้งสองทำงานแล้ว Stopper1 ก็จะยอมปล่อยให้ชิ้นงานเข้าไปได้และเครื่องก็จะเริ่มทำขั้นตอนการติดแผ่นกันกระแทกใหม่

3.7 การควบคุมมอเตอร์เสตป

ในการที่จะทำการดูดแผ่นกันกระแทกให้ครบทุกจุดนั้นมีข้อจำกัดของตัวแผ่นกันกระแทกคือแผ่นกันกระแทกจะใช้เป็นม้วน โดยที่ 1 แถว จะมีแผ่นกันกระแทกอยู่ด้วยกัน 9 ตัว ดังนั้นจึงได้มีการนำเอาเสต็ปมอเตอร์เข้ามาช่วยในการเลื่อนตำแหน่งแผ่นกันกระแทกให้ไปยังจุดต่างๆและในการเขียน

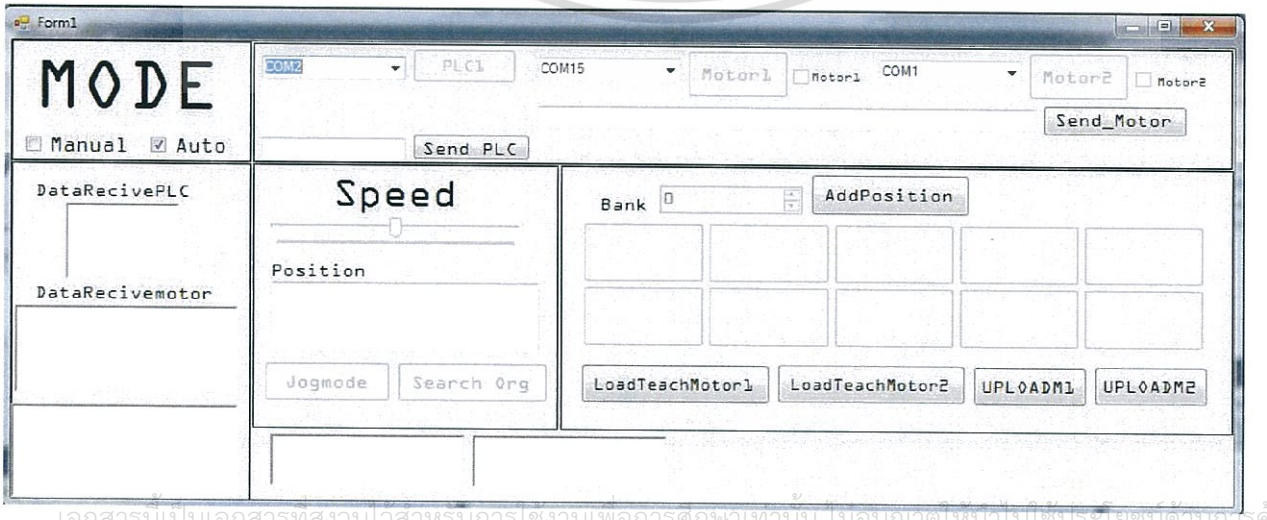
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมนั้นจะต้องมีการทำงานร่วมกันระหว่างเสต็ปมอเตอร์และพีแอลซี ดังนั้นการควบคุมเสต็ปมอเตอร์ด้วยพีแอลซีนั้นได้ทำการควบคุมด้วยกันสองวิธีคือ ใน Auto Sponge Version 1.0 ได้ใช้คอมพิวเตอร์เป็นตัวกลางในการควบคุมเสต็ปมอเตอร์ และใน Auto Sponge Version 2.0 ได้ใช้พีแอลซีในการควบคุมเสต็ปมอเตอร์ โดยทั้งสอง Version มีหลักการดังนี้

<u>Auto Sponge Version 1.0</u> ควบคุมโดยการใช้หลักการส่งผ่านข้อมูลผ่าน RS232 โดยที่มีคอมพิวเตอร์เป็นตัวกลางในการสื่อสารระหว่าง PLC และ มอเตอร์เสตป ดังรูปที่ 3.12

รูปที่ 3.12 การส่งผ่านข้อมูลของเครื่อง Auto Sponge Version 1.0

จากรูปที่ 3.12 คอมพิวเตอร์มีหน้าที่สำคัญในการเป็นตัวกลางในการสื่อสารระหว่าง PLC และ มอเตอร์เสตป ดังนั้นเพื่อให้ง่ายต่อผู้ใช้งานจึงต้องสร้างหน้าอินเตอร์เฟซสำหรับผู้ใช้งาน โดยใช้ภาษา C# ในการออกแบบดังรูปที่ 3.13



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.13 โปรแกรมควบคุมการสื่อสารระหว่าง PLC และ มอเตอร์สเตป ของเครื่อง Auto Sponge V 1.0

หลักการของโปรแกรม

คือการรับค่าส่งค่าด้วยรหัสฮัสกี้ผ่าน RS232 ดังนั้นในการส่งข้อมูลจาก PLC ไปยังคอมพิวเตอร์ใช้คำสั่งเปิด Port ส่ง ในการระบุพอร์ตที่จะส่งข้อมูลและส่งข้อมูลที่เป็นเลขฐานสิบหกด้วยคำสั่ง Data Memory ในที่นี้คือ S 1และ S2 และการรับข้อมูลจากคอมพิวเตอร์ของ PLC ใช้คำสั่งในการเปิดพอร์ตรับเมื่อคอมพิวเตอร์ส่งข้อมูลมาเป็นเลขฐานสิบหกก็จะเอาข้อมูลนั้นมาเทียบกับข้อมูลอ้างอิงที่ตั้งไว้ในที่นี้คือ P1 และ P2 ถ้าได้ข้อมูลที่ต้องการแล้วจึงสั่งการทำงานต่อไป

การส่งข้อมูลจากมอเตอร์สเตปไปยังคอมพิวเตอร์นั้นมอเตอร์เสตปจะทำการส่งข้อมูลมาตามการสั่งงานของคอนโทรลเลอร์ของมอเตอร์จึงใช้คอมพิวเตอร์ในการคัดกรองหาข้อมูลที่ต้องการและนำไปสั่งงานโปรแกรมต่อไป และการรับข้อมูลจากคอมพิวเตอร์ของมอเตอร์สเตปใช้คำสั่งในการเปิด Port ส่งจากคอมพิวเตอร์และส่งข้อมูลที่แปลงเป็นรหัสแอสกี้ไปยังมอเตอร์

โปรแกรมนี้สามารถเลือกโหมดได้สองแบบ คือ โหมดManualสามารถควบคุมมอเตอร์โดยการเลื่อน TrackBar โดยทิศทางและความเร็วของมอเตอร์ขึ้นอยู่กับค่าที่ TrackBar เลื่อนไป อีกทั้งสามาถเลือกจำนวนตำแหน่งที่ต้องการบันทึกและบันทึกตำแหน่งที่ต้องการลงในมอเตอร์สเตป โหมดAutoเป็นโหมดที่ควบคุมมอเตอร์สเตปได้ทั้งสองมอเตอร์สั่งงานโดย PLC มอเตอร์สเตปจะเลื่อนไปตามตำแหน่งที่เราบันทึกค่าลงไปในมอเตอร์ สามารถโหลดตำแหน่งเก่าที่เคยTeach ได้ ด้วยการกดปุ่ม LoadTeachMotor1 LoadTeachMotor2 และบันทึกค่าลงไปในมอเตอร์ใหม่ได้

หลักการทำงานของเสต็ปมอเตอร์ใน Auto Sponge V.1.0

เริ่มจากการเชื่อมต่อ PLC และมอเตอร์สเตปกับคอมพิวเตอร์แล้วจึงกำหนดตำแหน่งตามที่ต้องการด้วยโปรแกรมโหมด Manual จากนั้นบันทึกค่าลงมอเตอร์สเตปโดยการกดอัพโหลด เมื่อกดอัพโหลดเสร็จแล้วมอเตอร์จะกลับไปจุดเริ่มต้นของมอเตอร์ที่ตั้งไว้ และเปลี่ยนโปรแกรมเป็นโหมด Auto มอเตอร์จะเคลื่อนที่ได้ก็ต่อเมื่อได้รับค่า S1หรือS2(S1 เป็นการสั่งให้มอเตอร์สเตปตัวที่1 เคลื่อนที่ไปยังตำแหน่งที่ C# กำหนด และ S2 เป็นการสั่งให้มอเตอร์สเตปตัวที่2เคลื่อนที่ไปยังตำแหน่งที่ C# กำหนด) เมื่อมอเตอร์เสตปทั้งสองเคลื่อนที่ไปยังตำแหน่งนั้นๆเสร็จแล้ว ตัวมอเตอร์สเตปจะส่งค่าไปยัง Serial Port เข้า C# และแสดงผลออกมาทางโปรแกรมว่า End.1เมื่อได้ End.1 มา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C# จะคำนวณว่าเป็นมอเตอร์ตัวใดที่ส่งค่ามาจากนั้นจะส่ง P1 และ P2(P1 เป็นการรับค่ามาจากมอเตอร์ตัวที่1และP2เป็นการรับค่ามาจากมอเตอร์ตัวที่2)ไปยังPLC เพื่อทำขั้นตอนต่อไป ข้อจำกัดของ Auto Sponge V.1.1 คือมอเตอร์สเตปทั้งสองไม่สามารถวิ่งพร้อมกันได้(ต้องมีตัวใดตัวหนึ่งเร็วกว่าหรือช้ากว่า) เนื่องจากในการเขียนโปรแกรมนี้ได้ทำการเชื่อมต่อ Serial Port เพียง Port เดียวเท่านั้น จึงจำเป็นที่จะต้องเขียน Ladder .ของ PLC ให้ทำการส่ง S1 และ S2 สลับกัน

Auto Sponge Version 2.0 ควบคุมโดยการใช้การสื่อสารระหว่าง PLC และ มอเตอร์เสตป ดังรูปที่ 3.14

รูปที่ 3.14 การส่งผ่านข้อมูลข้อเครื่อง Auto Sponge Version 2.0

จากรูปที่ 3.14 มอเตอร์เสตปสามารถควบคุมด้วย Output ของ PLC วายริ่งไปที่ Input ของมอเตอร์เสตป โดย Auto Sponge เวอร์ชั่นนี้ใช้มอเตอร์เสตปของ MISUMI SINGLE-AXIS ROBOT CONTROLLER โดยคุณสมบัติและหลักการสั่งงานมอเตอร์มีดังนี้

หลักการทำงานของเสต็ปมอเตอร์ใน Auto Sponge V.2.0

จากรูปที่ 3.14 และ 3.15 สามารถควบคุมมอเตอร์สเตปผ่าน PLC โดยตรงได้ด้วยการวายริ่ง Output ของ PLC มาเข้ากับ Input ของมอเตอร์นั่นคือ A1-A20 หลักการในการจะขับเคลื่อนมอเตอร์สเตปไปยังตำแหน่งที่เราบันทึกไว้ดังต่อไปนี้

I/O Specification		Explanation
Parallel I/O	NPN	16 input points, 24VDC ± 10%, 5.1mA/point, plus common 16 output points, 24VDC ± 10%, 50mA/point, sync type
	PNP	16 input points, 24VDC ± 10%, 5.5mA/point, minus common 16 output points, 24VDC ± 10%, 50mA/point, source type
Serial I/O	CC-Link	CC-Link Ver.1.10 compatible, remote device station (1 station)
	DeviceNet	Number of channels occupied by DeviceNet slaves: Inputs: 6CH; Outputs: 6CH.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.15 เสปค I/O ของมอเตอร์สเตป

1.1 NPN and PNP type

Both the NPN and PNP types have 16 input points and 16 output points.

■ I/O signal table

No.	Signal Name	Description		No.	Signal Name	Description	
A1	+COM	I/O power input, positive common (24VDC ±10%)		B1	POUT0	Outputs	Point No. outputs
A2				B2	POUT1		
A3	NC	No connection		B3	POUT2		
A4	NC			B4	POUT3		
A5	PIN0	Inputs	Point No. select	B5	POUT4		
A6	PIN1			B6	POUT5		
A7	PIN2			B7	POUT6		
A8	PIN3			B8	POUT7		
A9	PIN4			B9	OUT0		Control outputs
A10	PIN5			B10	OUT1		
A11	PIN6			B11	OUT2		
A12	PIN7			B12	OUT3		
A13	JOG+		JOG movement (+ direction)	B13	BUSY		Operation-in-progress
A14	JOG-		JOG movement (- direction)	B14	END		Operation-end
A15	MANUAL		MANUAL mode	B15	/ALM		Alarm
A16	ORG		Return-to-origin	B16	SRV-S		Servo status
A17	/LOCK		Interlock	B17	NC	No connection	
A18	START		Start	B18	NC		
A19	RESET		Reset	B19	-COM	I/O power input, negative common (0V)	
A20	SERVO		Servo ON	B20			

รูปที่ 3.16 I/O ของมอเตอร์เสตป

จากรูปที่ 3.16 I/O ของตำแหน่ง มีตั้งแต่ PIN0-PIN7 ดังนั้นการที่จะเลื่อนไปตำแหน่งลำดับที่มากขึ้นจึงใช้หลักการเลขฐานสอง เช่น ต้องการตำแหน่งที่ 41 ดังนั้นจึงจ่ายไฟให้กับ PIN0 แปลงเป็นเลขฐานสองเท่ากับ 1 จ่ายไฟให้กับ PIN3 แปลงเป็ฯเลขฐานสองเท่ากับ 8 และจ่ายไฟให้กับ PIN5 แปลงเป็นเลขฐานสองได้ 32 รวมกันทั้งหมดเท่ากับ 41 เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIN0 to PIN7 (Point No. Select)

These inputs are read as 8-bit binary code Point Nos. when the START and TEACH commands are executed.

- **Input example**

PIN7 PIN0

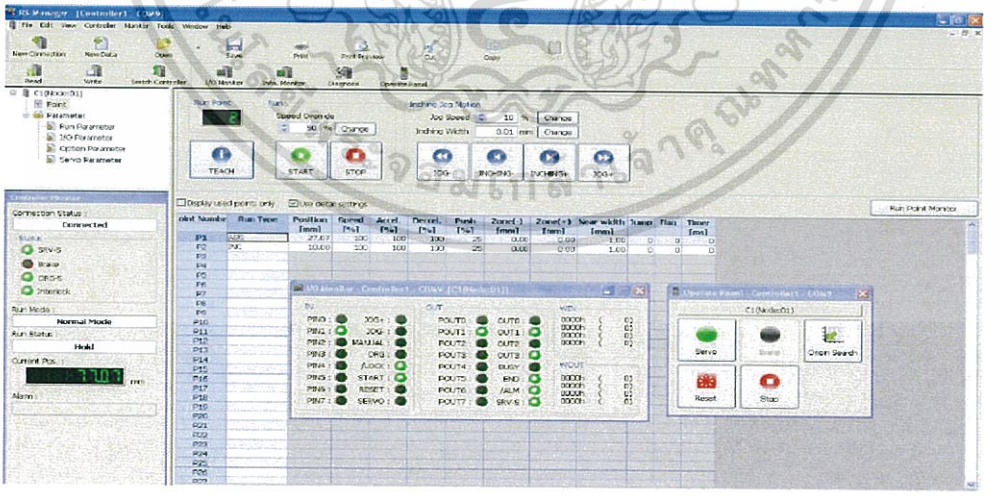
0	0	1	0	1	0	0	1

Sum at ON	Example	
2^0	1	
2^1	0	
2^2	0	
2^3	8	Total = 41 (Point No.41)
2^4	0	
2^5	32	
2^6	0	
2^7	0	

รูปที่ 3.17 การเลือกตำแหน่งที่ต้องการโดยการต่อ I/O

ในการ Teaching ตำแหน่งลงไปในมอเตอร์นั้น ได้ใช้โปรแกรม

ในการ โปรแกรมนี้เป็นโปรแกรมที่ควบคุมมอเตอร์ได้ผ่าน RS232 ซึ่งมีความสะดวกและแม่นยำสูง โดยหลักการของโปรแกรมนี้จะเป็นการส่ง command ผ่าน RS232 เข้าไปใน Controller ของมอเตอร์เสตป



รูปที่ 3.18 ภาพโปรแกรม RS-Manager

โปรแกรม RS-Manager สามารถบันทึกตำแหน่งได้เป็นจำนวนมากและการสั่งงานตำแหน่งนั้นๆ จำเป็นต้องคลิกที่ตำแหน่งที่ต้องการไว้แล้วกด START มอเตอร์ก็จะทำงานตามคำสั่งภายในตำแหน่งที่กำหนดไว้ โดยการกำหนดคำสั่งที่ใช้มี ดังนี้

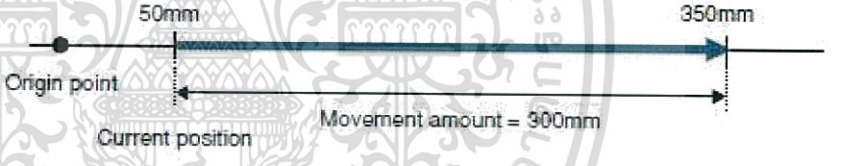
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ABS คือเมื่อทำงานครั้งหนึ่งแล้วจะวิ่งไปเท่ากับระยะที่กำหนดและเมื่อสั่งให้ทำงานอีกถ้าตำแหน่งเดิมคือจุดที่อยู่แล้วก็จะไม่วิ่ง เช่น กำหนดให้มอเตอร์ตำแหน่งเริ่มต้นที่ 50วิ่งไป 300 โดยคำสั่ง ABS มอเตอร์ก็จะวิ่งไปที่ตำแหน่ง 300 ดังรูป

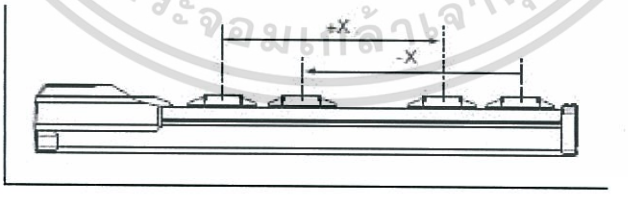
RUN type: ABS

- INC คือเมื่อทำงานครั้งหนึ่งแล้วจะวิ่งไปเท่ากับตำแหน่งเดิมบวกกับระยะที่กำหนดและเมื่อสั่งให้ทำงานอีก มอเตอร์ก็จะวิ่งไปที่ตำแหน่งเดิมบวกกับระยะที่กำหนด เช่น กำหนดให้มอเตอร์ตำแหน่งเริ่มต้นที่ 50 วิ่งไป 300 โดยคำสั่ง INC มอเตอร์ก็จะวิ่งไปที่ตำแหน่ง 350 ดังรูป

RUN type: INC



การกำหนดตำแหน่งให้มอเตอร์สเตปนั้น สามารถทำได้โดยใส่ค่าที่ Position ส่วนทิศทางก็ตามเครื่องหมาย คือ

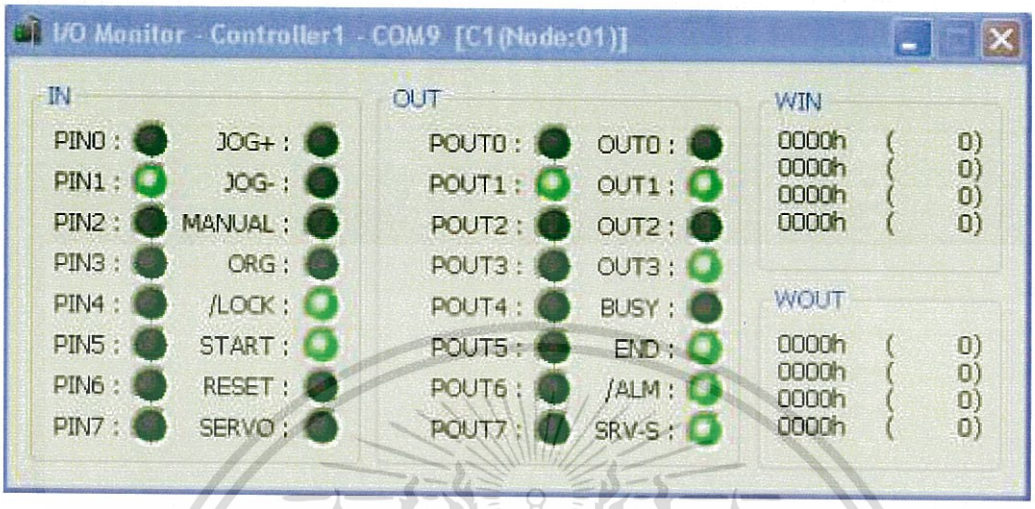


- เลขเป็นจำนวนบวก มอเตอร์เคลื่อนที่ไปในทิศทาง X+
- เลขเป็นจำนวนลบ มอเตอร์เคลื่อนที่ไปในทิศทาง X-

และสามาถกำหนดความเร็วและความเร่งในการเคลื่อนที่ของมอเตอร์ได้โดยใส่ค่าเป็นเปอร์เซ็นท์ที่ Speed และ Acceleration อีกทั้งยังมีโปรแกรมในการ Teaching โดยการเลือก JOG+ เคลื่อนที่ไปแนว X+ JOG- เคลื่อนที่ไปแนว X-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม RS-Manager ยังสามารถดู I/O ในขณะที่มอเตอร์ทำงานอยู่ได้ทำให้ง่ายต่อการวายริ่งเป็นอย่างมาก แสดงตัวอย่างในรูปที่ 3.19



รูปที่ 3.19 ภาพ I/O ของมอเตอร์สเตป

การทำงานของมอเตอร์ใน Auto Sponge V.2.0

ทำการเชื่อมต่อมอเตอร์สเตปทั้งสองกับคอมพิวเตอร์ด้วยโปรแกรม RS-Manager จากนั้นทำการกำหนดตำแหน่งแต่ละมอเตอร์โดย ตำแหน่งที่1 จะเป็นตำแหน่งที่สามารถตรวจจับแผ่นกันกระแทกตัวแรกได้ ใช้คำสั่งการเคลื่อนที่แบบ ABS ตำแหน่งที่2 จะใช้คำสั่งการเคลื่อนที่แบบ INC โดยใช้ระยะจากการใช้เวอร์เนียวัดจุดศูนย์กลางของแผ่นกันกระแทกแผ่นแรกจนถึงจุดศูนย์กลางของแผ่นกันกระแทกแผ่นที่สองจะได้ระยะห่างระหว่างแผ่นกันกระแทกที่เท่าๆกันเท่ากับ 10 mm จากนั้นกด Write ทำการเขียนข้อมูลลงในมอเตอร์สเตป มอเตอร์จะเคลื่อนทีได้ก็ต่อเมื่อ I/O PIN1 และ START ของมอเตอร์ทำงาน โดยจะทำงานได้นั้นก็จะได้รับคำสั่งจาก PLC Auto Sponge V.2.0 สามารถเคลื่อนที่พร้อมกันได้เนื่องจากโปรแกรมใช้การต่อ Serial Port แบบแยก Port กัน ข้อจำกัดของ Auto Sponge V.2.0 คือ เนื่องจากใช้การสื่อสารกันระหว่าง PLC กับ มอเตอร์สเตป กันโดยใช้ I/O ทำให้มี Output มาก จึงสิ้นเปลืองต่อการใช้ PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

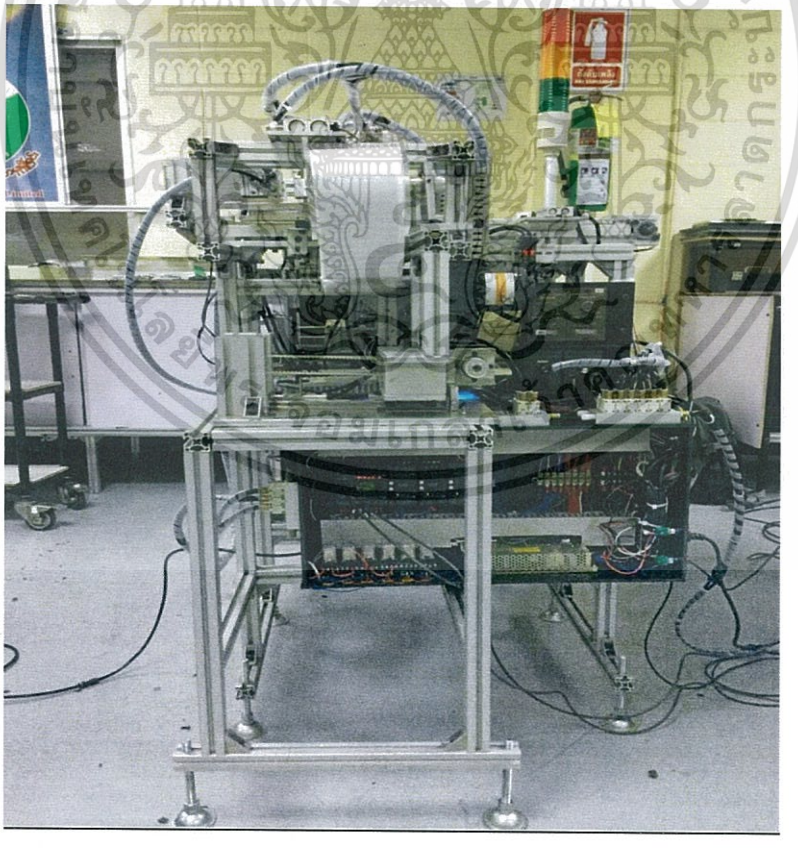
บทที่4

ผลการดำเดินงาน

4.1 ผลการดำเนินงาน Auto Sponge ในเวอร์ชั่นต่างๆ

จากการดำเนินงานดังข้างต้นมาแล้วนั้นในบทนี้จะเป็นการทดสอบเครื่องและการนำไปใช้งานในสายการผลิตจริง

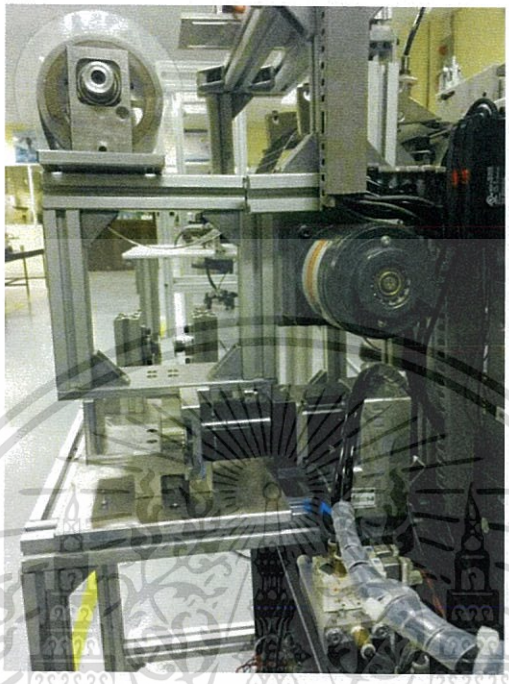
Auto Sponge V.1.0



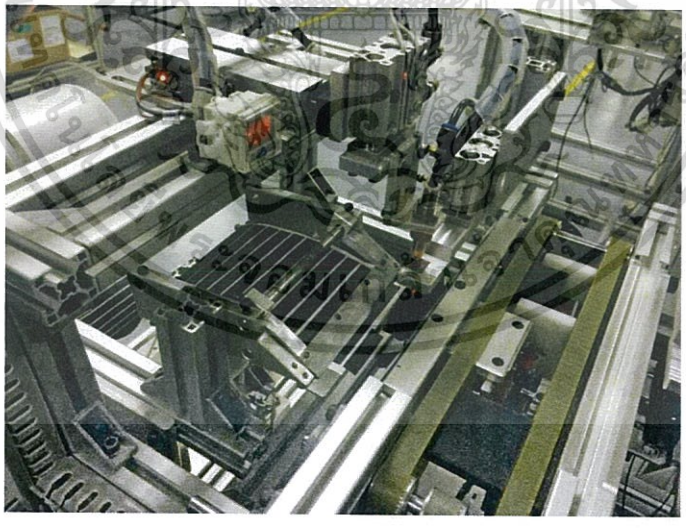
รูปที่ 4.1 Auto Sponge V.1.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการทดสอบก่อนทำเครื่องขึ้นไปใช้งานจริง เมื่อเปิดเครื่องมอเตอร์จะมีแสงสีน้ำเงินเป็นสถานะว่ามอเตอร์ทำงานเป็นปกติ และกดปุ่ม Start/Stop เครื่องก็สามารถทำงานได้ ดังรูป



รูปที่ 4.2 แสดงให้เห็นไฟสีน้ำเงินของมอเตอร์เสตปว่าสามารถทำงานได้อย่างปกติ

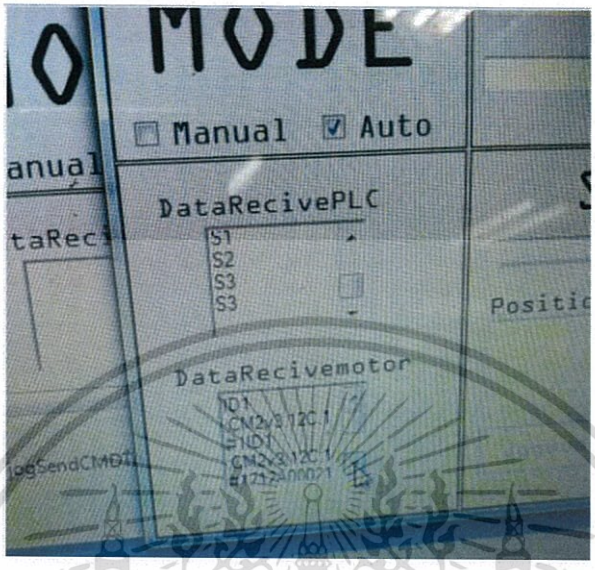


รูปที่ 4.3 เครื่อง Auto Sponge V.1.0 ทำงานปกติที่ตำแหน่งต่างๆ

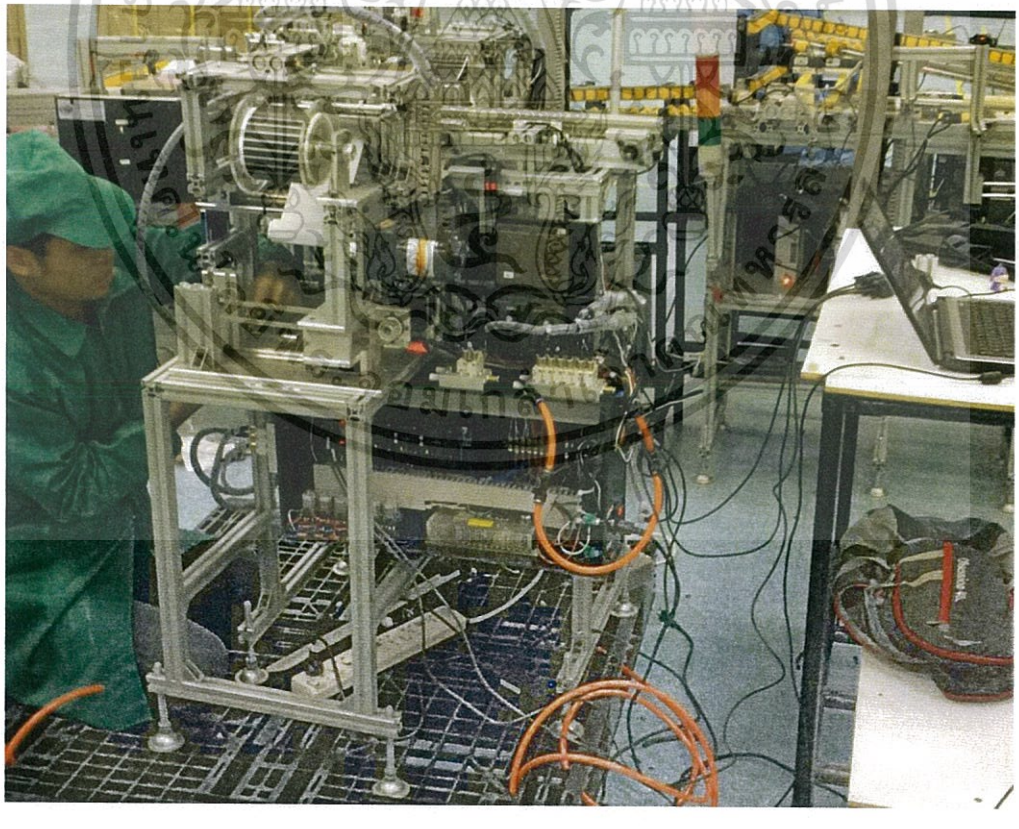
จากรูปที่ 4.3 เครื่อง Auto Sponge V.1.0 ทำการดูดแผ่นกันกระแทกที่ตำแหน่งต่างๆไปติดชิ้นงานได้จนแผ่นกันกระแทกหมดแถว จากนั้นมอเตอร์จะทำการ Feed แผ่นกันกระแทกออกมาและมอเตอร์สเตปจะวิ่งกลับในทิศทางตรงข้ามทำให้เครื่องทำงานด้วยเวลาที่รวดเร็วและแม่นยำ แต่เมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่อง Auto Sponge ทำงานไปได้สักระยะปรากฏว่ามอเตอร์สเตปส่งค่า ERROR กลับมา ทำให้มอเตอร์เลื่อนกลับไปยังจุดเริ่มต้นโดยไม่ทราบสาเหตุ



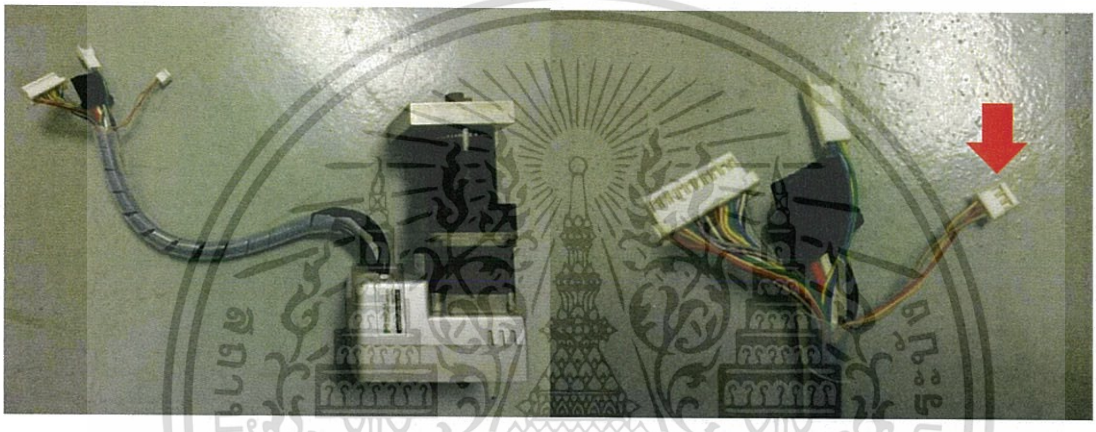
รูปที่ 4.4 โปรแกรมได้รับค่า ERROR จากมอเตอร์เสตป



รูปที่ 4.5 มอเตอร์สเตปมี Error สัญญาณไฟจะเป็นสีแดงและวิ่งกลับไปยังจุดเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

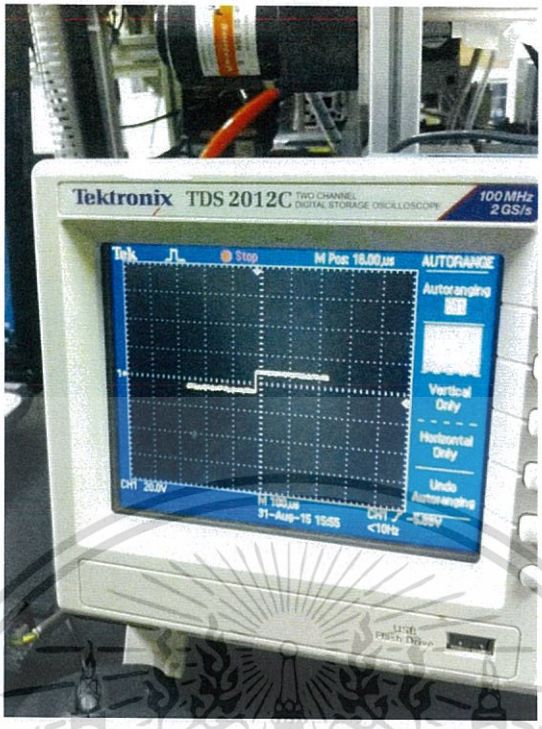
จากรูปที่ 4.4 และ 4.5 มอเตอร์เสตปจะมีปัญหาคือเมื่อทำงานไปในตำแหน่งๆต่างๆสักระยะหนึ่งแล้ว ตัวมอเตอร์จะข้อมูลกลับมาโดยเป็น ชื่อรุ่นของมอเตอร์ และจะวิ่งกลับไปจุดเริ่มต้นขณะที่ยังดูดแผ่นกันกระแทกไปไม่หมดและหยุดทำงาน โดยการจะให้มอเตอร์ส่งชื่อรุ่นของมอเตอร์กลับมานั้นจะต้องส่งคำสั่ง Command จากโปรแกรม C# แต่จากการตรวจสอบแล้วไม่พบอะไรผิดปกติ ดังนั้นได้ปรึกษากับพนักงานที่เชี่ยวชาญ ได้รับคำแนะนำว่าอาจเกิดคลื่นรบกวนโครนทอลเลอร์ภายในตัวมอเตอร์สเตปจึงได้ทำการใช้เครื่องวัดสัญญาณวัดที่การส่งข้อมูลผ่าน RS232ระหว่างมอเตอร์และคอมพิวเตอร์



รูปที่ 4.6 การวัดสัญญาณมอเตอร์เสตปด้วยเครื่องสโคป

การวัดสัญญาณที่ส่งผ่านระหว่างคอมพิวเตอร์และมอเตอร์สเตปทำได้โดยการวัดจากสายที่ใช้ในการเชื่อมต่อจากมอเตอร์สเตปไปยังคอมพิวเตอร์ผ่าน RS232 ดังลูกศรในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 ภาพจากเครื่องวัดสโคปในการวัดขณะที่มอเตอร์เสตปทำงานปกติ

รูปที่ 4.8 ภาพจากเครื่องวัดสโคปในการวัดขณะที่มอเตอร์เสตปทำงานผิดปกติ

จากรูปที่ 4.7 เริ่มต้นทำการวัดสัญญาณในขณะที่เครื่อง Auto Sponge V.1.0 ทำงานเป็นปกติจะเห็นได้ว่าสัญญาณของมอเตอร์เวลาเคลื่อนที่หรือทำงานปกตินั้น สัญญาณจะเป็น Pulse และรูปที่ 4.8 เป็นการวัดสัญญาณในขณะที่เครื่อง Auto Sponge V.1.0 ทำงานผิดปกติจะเห็นได้ว่า

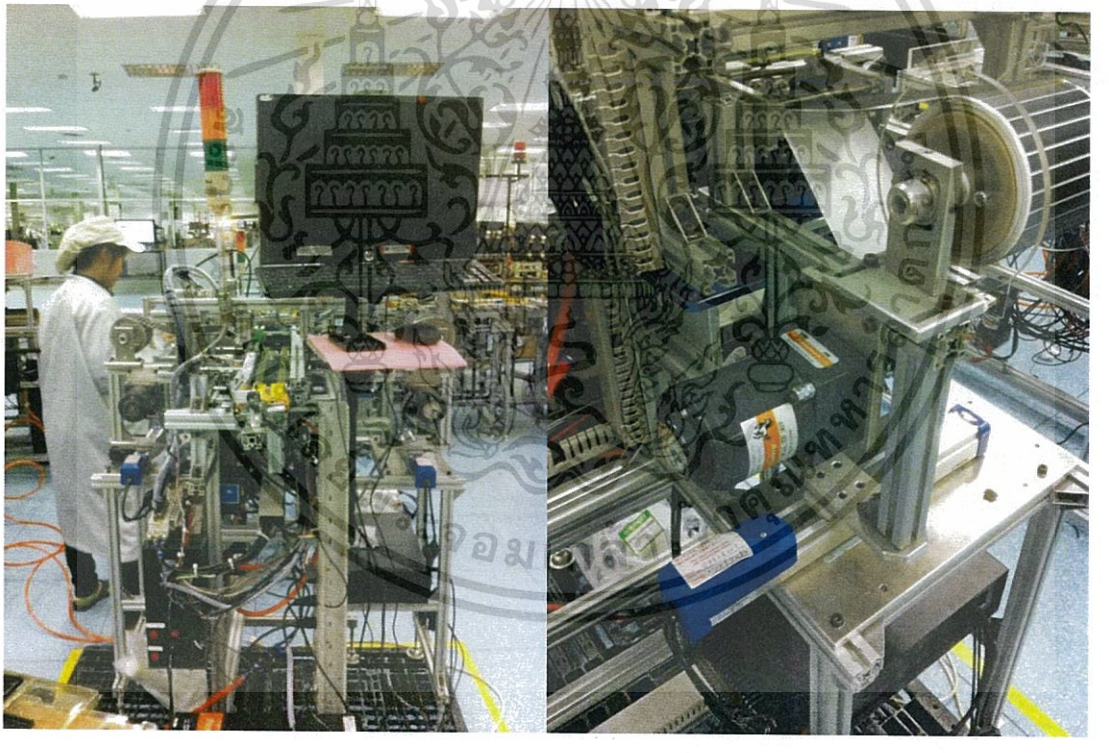
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณของมอเตอร์ในขณะที่มอเตอร์สเตปทำงานผิดปกติ ส่งค่าแปลกมาทำให้มอเตอร์เคลื่อนที่กลับไปยังจุดเริ่มต้นในขณะที่ยังทำงานอยู่ ดังนั้นจึงสรุปได้ว่าเกิดปัญหาที่สัญญาณรบกวนคอนโทรเลอร์ภายในมอเตอร์สเตป

Auto Sponge V.2.0

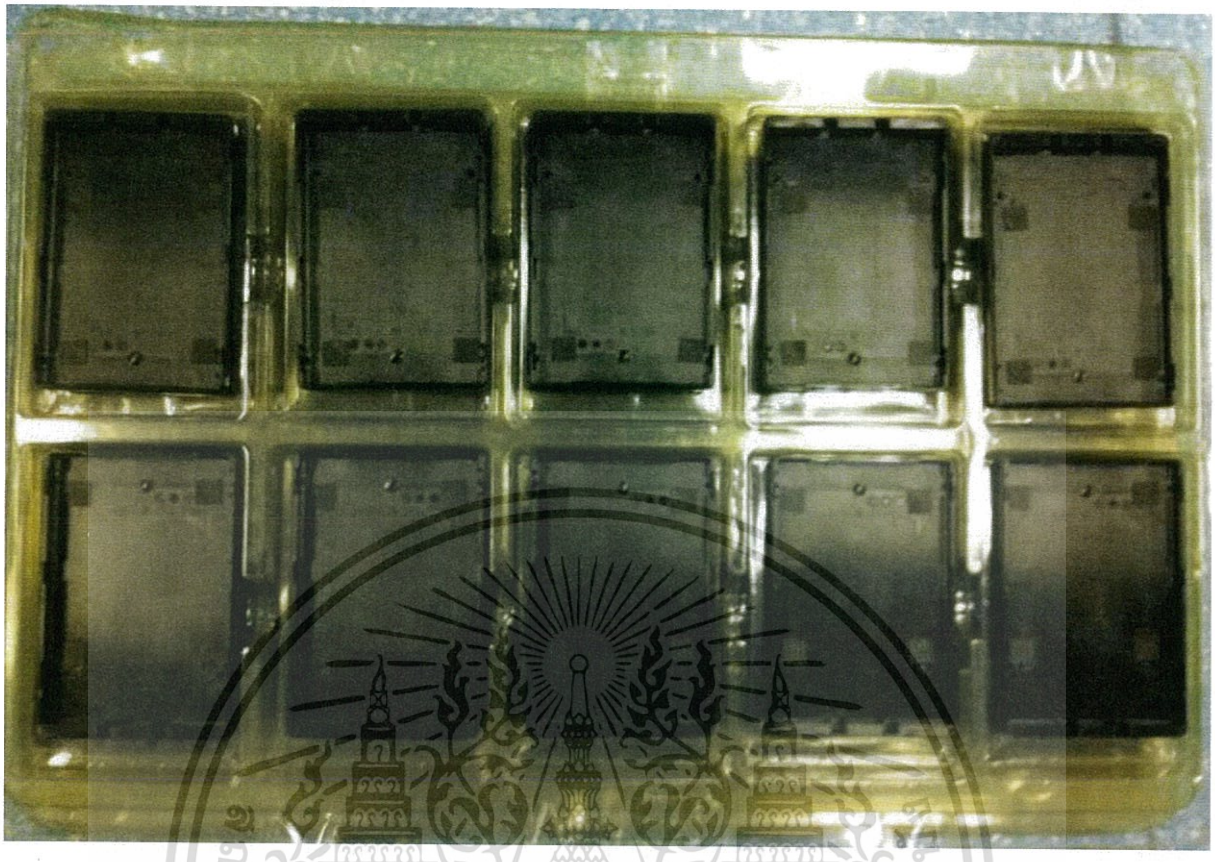
ดังนั้นจากการผลของเครื่อง Auto Sponge V.1.0 ไม่สามารถแก้ไขปัญหาสัญญาณรบกวนมอเตอร์ได้ จากคำปรึกษาของพนักงานที่ปรึกษาจึงได้ตัดสินใจในการเปลี่ยนมอเตอร์เสตปมอใช้มอเตอร์เสตปแบบโรบอทหนึ่งแกนและได้พัฒนาจนมาเป็น Auto Sponge V.2.0

เริ่มต้นด้วยการทดสอบเครื่อง Auto Sponge V.2.0 ก่อนนำเข้าสายการผลิตจริง เนื่องจากต้องแน่ใจว่าเครื่องจักรสามารถทำงานได้โดยไม่มีปัญหา ผลการทดสอบเป็นดังนี้



รูปที่ 4.9 และ 4.10 การทดสอบเครื่อง Auto Sponge V.2.0 ก่อนนำเข้าสายการผลิตจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 ชิ้นงานที่ทำงานโดยเครื่อง Auto Sponge V.2.0

4.2 ประสิทธิผลโดยรวมของเครื่องจักร

การวัดประสิทธิผลโดยรวมของเครื่องจักร(OEEหรือOverall Equipment Effectiveness) เป็นวิธีการที่ดีวิธีหนึ่งที่นอกจากทำให้รู้ประสิทธิผลของเครื่องจักรแล้วยังรู้ถึงสาเหตุของความสูญเสียที่เกิดขึ้นทั้งในภาพใหญ่ คือ สามารถแยกประเภทการสูญเสียและรายละเอียดของสาเหตุนั้น ทำให้สามารถที่จะปรับปรุง ลดความสูญเสียที่เกิดขึ้นได้อย่างถูกต้องและเป็นระบบ

เครื่องจักรที่ดีไม่ใช่เป็นเพียงแค่เครื่องจักรที่ไม่เสีย เปิดสวิตซ์เมื่อใดทำงานได้เมื่อนั้น หากแต่ต้องเป็นเครื่องจักรที่เปิดขึ้นมาแล้วทำงานได้อย่างเต็มประสิทธิภาพคือ เดินเครื่องได้เต็มกำลังความสามารถ แต่ถ้าเครื่องจักรใช้งานได้ตลอดเวลาและเดินเครื่องได้เต็มกำลัง แต่ชิ้นงานที่ผลิตออกมาไม่มีคุณภาพ ก็คงไม่มีประโยชน์อะไร ดังนั้นเรื่องคุณภาพของงานที่ออกมาจึงเป็นอีกปัจจัยหนึ่งที่จะใช้ในการพิจารณาเครื่องจักร และที่สำคัญเครื่องจักรที่ดีต้องใช้งานได้อย่างปลอดภัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำนวณ OEE

ประกอบด้วยผลคูณของ 3 Factor ดังนี้

OEE = อัตราเดินเครื่อง x ประสิทธิภาพเดินเครื่อง x อัตราคุณภาพ
(Availability) (Performance Efficiency) (Quality Rate)

ซึ่งเมื่อนำปัจจัยต่างๆ ที่ส่งผลกระทบต่อระบบการผลิต อันได้แก่ พนักงาน, เครื่องจักร และชิ้นงานที่ผลิต มาวิเคราะห์แล้ว จะทำให้ทราบได้ว่าเกิดอะไรขึ้นกับระบบการผลิตของเราบ้าง ซึ่ง OEE จะเป็นดัชนีที่ชี้ให้เห็นสภาพโดยรวมในระบบการผลิต

เกณฑ์มาตรฐานของ OEE

ประสิทธิผลโดยรวมของเครื่องจักรอุปกรณ์(Overall Equipment Effectiveness : OEE) ที่ตั้งเป็นมาตรฐานโดยทั่วไป

อัตราการเดินเครื่อง(Availability) = 90%
ประสิทธิภาพการเดินเครื่อง(Performance Efficiency) = 95%
อัตราคุณภาพ(Quality Rate) = 99%

ดังนั้น ประสิทธิผลโดยรวมของเครื่องจักรอุปกรณ์
(Overall Equipment Effectiveness : OEE) = 0.90 x 0.95x 0.99 x 100 = 85%

ค่าดังกล่าวมิใช่ค่าเป้าหมายที่บังคับใช้ สามารถกำหนดค่าเป้าหมายได้ความเหมาะสมของแต่ละโรงงาน

ดังนั้นในการทดสอบดังนี้ - ให้ผลิตชิ้นงานจำนวน 5000 ชิ้น

- เครื่องจักรต้องเดินเครื่อง 1380 นาที/วัน

ตารางที่ 4.1 การบันทึกผลการทำงานของ Auto Sponge V.2.0

ชิ้นงานที่ผลิตได้	ชิ้นงานที่เสียหาย	เวลาที่เครื่องจักรหยุดทำงาน
4800 ชิ้น	100 ชิ้น	ปัญหาทางแทคคานิค 100 นาที
-	-	ปัญหาทางโปรแกรม 30 นาที
-	-	รวม 130 นาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางดังกล่าว สามารถหาค่า

$$\text{อัตราการเดินเครื่องได้} = \frac{\text{เวลาเดินเครื่อง}}{\text{เวลาปฏิบัติงานสุทธิ}} = \frac{410 \text{ นาที}}{540 \text{ นาที}} = 0.76$$

$$\text{ประสิทธิภาพเดินเครื่อง} = \frac{\text{ผลผลิตจริง}}{\text{ผลผลิตเป้าหมาย}} = \frac{4800 \text{ ชิ้น}}{5000 \text{ ชิ้น}} = 0.96$$

$$\text{อัตราคุณภาพ} = \frac{\text{ผลผลิตที่เป็นของดี}}{\text{ผลผลิตจริง}} = \frac{4700 \text{ ชิ้น}}{4800 \text{ ชิ้น}} = 0.98$$

ดังนั้น ประสิทธิผลโดยรวมในการทดสอบเครื่องจักร

(Overall Equipment Effectiveness : OEE) = 0.76 x 0.96x 0.98 x 100 = 71.15 %

จะเห็นว่าประสิทธิผลโดยรวมของเครื่อง Auto Sponge V.2.0 ยังมีเปอร์เซ็นท์ที่ต่ำกว่าเกณฑ์ ดังนั้น จึงต้องมีการปรับปรุงก่อนนำเครื่องเข้าสายการผลิตจริงโดยมีหลักการ

<u>การปรับปรุงค่า OEE</u>

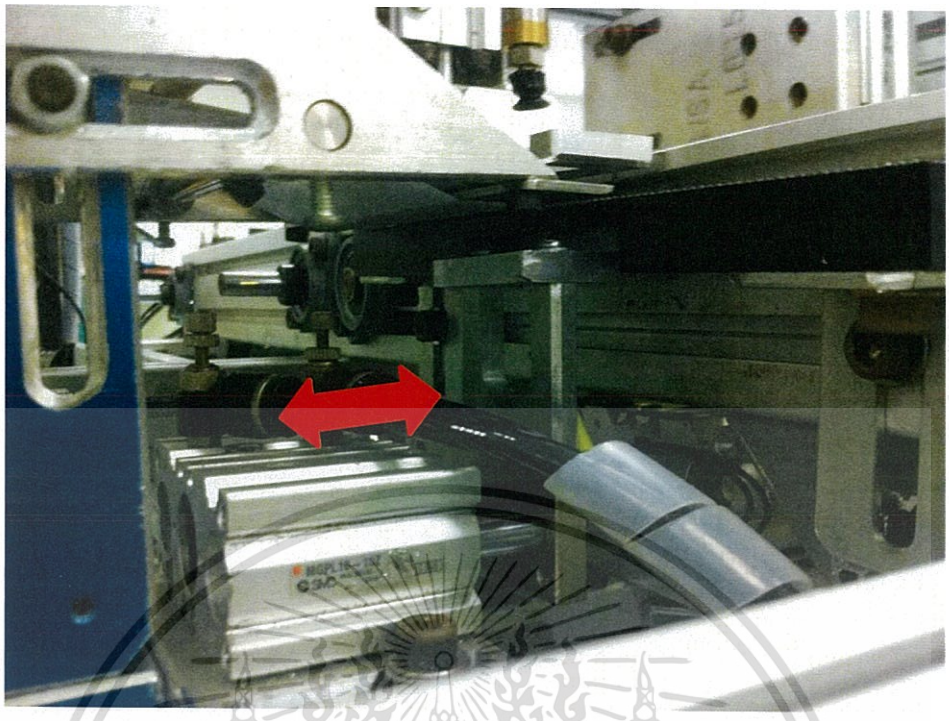
OEE เป็นค่าเปอร์เซ็นต์ที่มาจากการคูณกันระหว่าง อัตราการเดินเครื่องได้, ประสิทธิภาพเดินเครื่อง และ อัตราคุณภาพ ดังนั้นการปรับปรุงค่า OEE ก็คือการปรับปรุงค่าทั้งสามเหลานี้ตัวใดตัวหนึ่ง หรือสองตัว หรือทั้งสามตัว ขึ้นอยู่กับความจำเป็นเร่งด่วนหรือขึ้นอยู่กับนโยบายในขณะนั้น แต่ถ้าไม่มีความจำเป็นหรือนโยบายอะไรเป็นพิเศษ โดยปกติเราจะปรับปรุงค่าที่ต่ำที่สุดก่อน

ดังนั้นจึงทำการปรับปรุงจาก

1. อัตราการเดินเครื่องได้

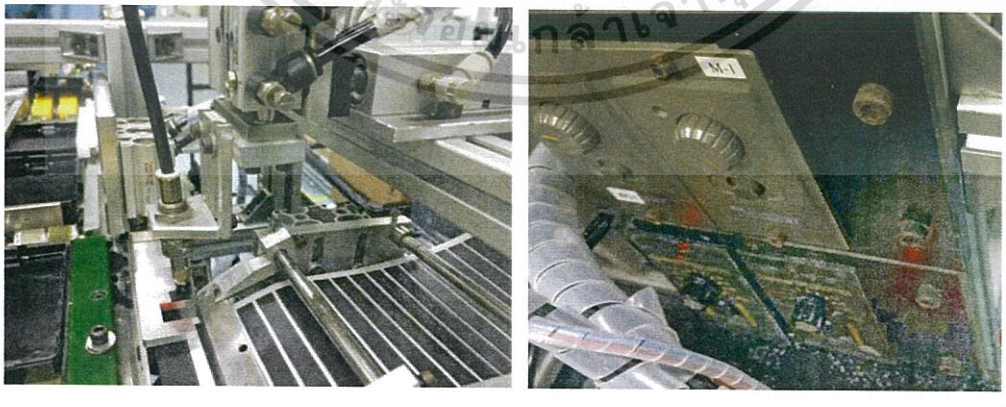
1.1 เนื่องจากที่เครื่องหยุดทำงานบ่อยเป็นเพราะเมื่อมอเตอร์ทำการ Feed แผ่นกันกระแทกออกมาเวลาเลื่อนไปที่ตำแหน่งต่อไปเซ็นเซอร์แอมพลิฟายเออร์จะไม่สามารถตรวจจับแผ่นกันกระแทกได้ทำให้มอเตอร์ Feed ในขณะที่แผ่นกันกระแทกยังถูกดูดออกไปไม่หมดและต้องหยุดการทำงานเพื่อจะเอาแผ่นกันกระแทกออกมา วิธีการแก้ปัญหาจึงได้ทำการแก้ไขแมคคานิคโดยการเพิ่มกระบอกสูบบริเวณเซ็นเซอร์แอมพลิฟายเออร์ จะสามารถแก้ไขปัญหานี้ได้โดยเมื่อทำมอเตอร์ทำการ Feed และตรวจจับได้โดยเซ็นเซอร์แอมพลิฟายเออร์กระบอกสูบที่ก็จะเลื่อนเข้าทำให้เซ็นเซอร์แอมพลิฟายเออร์ตรวจจับแผ่นกันกระแทกที่ถูก Feed ออกมาได้อย่างแน่นอนและเมื่อแผ่นกันกระแทกถูกดูดจนหมดแถวกระบอกสูบก็จะเลื่อนออกไปอีกครั้งเพื่อให้ Feed ได้ปริมาณที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 การทำงานของกระบอกสูบที่ช่วยแก้ปัญหาเรื่องเซ็นเซอร์แอมพลิฟายเออร์ตรวจจับไม่ได้

1.2 เนื่องจากการ Feed แผ่นกันกระแทกออกมานั้นจะต้อง Feed ในปริมาณที่เหมาะสมเพราะถ้าทำการ Feed มากเกินไปอาจจะทำให้หัวดูดไม่สามารถดูดแผ่นกันกระแทกได้เต็มแผ่นทำให้ไม่สามารถดูดแผ่นกันกระแทกไปติดที่ชิ้นงานได้จึงต้องมีการหยุดเครื่องและปรับการแก้ไขอยู่บ่อยๆ วิธีการแก้ปัญหานี้ได้ทำการติดตั้งให้เซ็นเซอร์แอมพลิฟายเออร์กับหัวดูดตรงกันและปรับความเร็วในการ Feed ด้วยที่ปรับความเร็วของมอเตอร์ให้ Feed แผ่นกันกระแทกออกมาในปริมาณที่พอดี

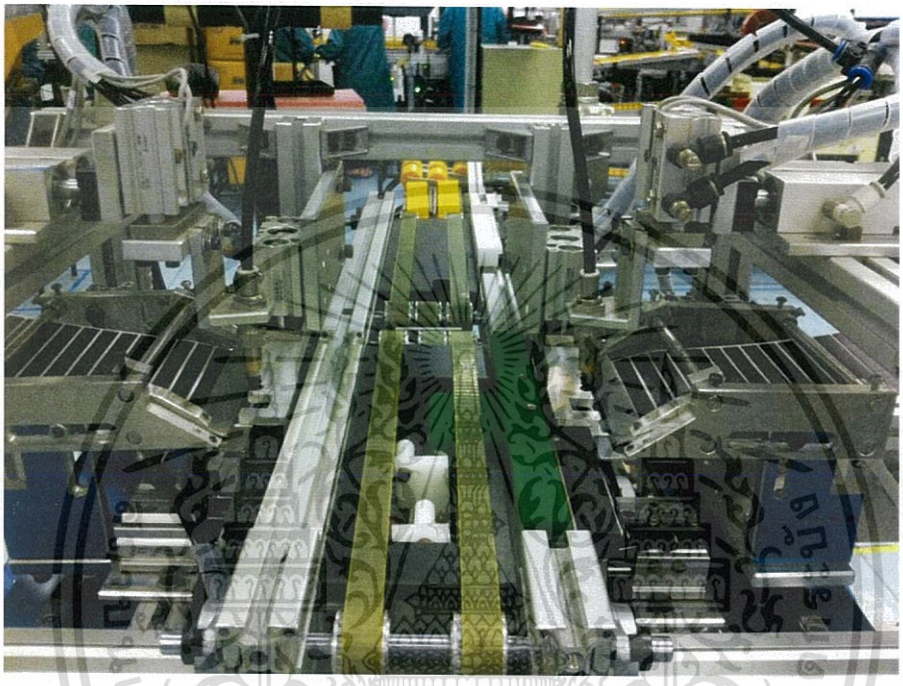


รูปที่ 4.13 การปรับโครงสร้างของหัวดูดกับเซ็นเซอร์แอมพลิฟลายเออร์ให้ตรงกันและปรับความเร็วมอเตอร์ของการ Feed

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. อัตราคุณภาพ

2.1 เนื่องจากชิ้นงานที่ผลิตได้มีความเสียหายมากจากเกิดรอยขีดข่วนกับชิ้นงานระหว่างที่กระบอกสูบหนีบชิ้นงานไว้เพื่อติดแผ่นกันกระแทก วิธีการแก้ไขปัญหาจึงทำได้โดยการติดสักราชลงบนที่หนีบและแก้ไขโปรแกรมใหม่โดยให้สายพานหยุดหมุนในขณะที่ชิ้นงานมารอติดแผ่นกันกระแทก



รูปที่ 4.14 ทำการติดสักราชลงบนที่หนีบและแก้ไขโปรแกรม

จากการปรับปรุงเครื่อง Auto Sponge V.2.0

ดังนั้นคำนวณค่า OEE หลังการปรับปรุงเครื่องจักร

- ให้ผลิตชิ้นงานจำนวน 5500 ชิ้น
- เครื่องจักรต้องเดินเครื่อง 540 นาที/วัน

ตารางที่ 4.2 การบันทึกผลการทำงานของ Auto Sponge V.2.0

ชิ้นงานที่ผลิตได้	ชิ้นงานที่เสียหาย	เวลาที่เครื่องจักรหยุดทำงาน
5000 ชิ้น	-	ปัญหาทางแทคคานิค 10 นาที
-	-	-
-	-	รวม 10 นาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางดังกล่าว สามารถหาค่า

$$\text{อัตราการเดินเครื่องได้} = \frac{\text{เวลาเดินเครื่อง}}{\text{เวลาปฏิบัติงานสุทธิ}} = \frac{530 \text{ นาที}}{540 \text{ นาที}} = 0.98$$

$$\text{ประสิทธิภาพเดินเครื่อง} = \frac{\text{ผลผลิตจริง}}{\text{ผลผลิตเป้าหมาย}} = \frac{5300 \text{ ชิ้น}}{5500 \text{ ชิ้น}} = 0.96$$

$$\text{อัตราคุณภาพ} = \frac{\text{ผลผลิตที่เป็นของดี}}{\text{ผลผลิตจริง}} = \frac{5500 \text{ ชิ้น}}{5500 \text{ ชิ้น}} = 1.00$$

ดังนั้น ประสิทธิผลโดยรวมหลังการปรับปรุงเครื่องจักร

(Overall Equipment Effectiveness : OEE) = 0.98 x 0.96x 1.00 x 100 = 94.08 %

จะเห็นว่าประสิทธิผลโดยรวมของเครื่อง Auto Sponge V.2.0 มีเปอร์เซ็นท์ที่ดีขึ้นมากและสูงกว่าเกณฑ์

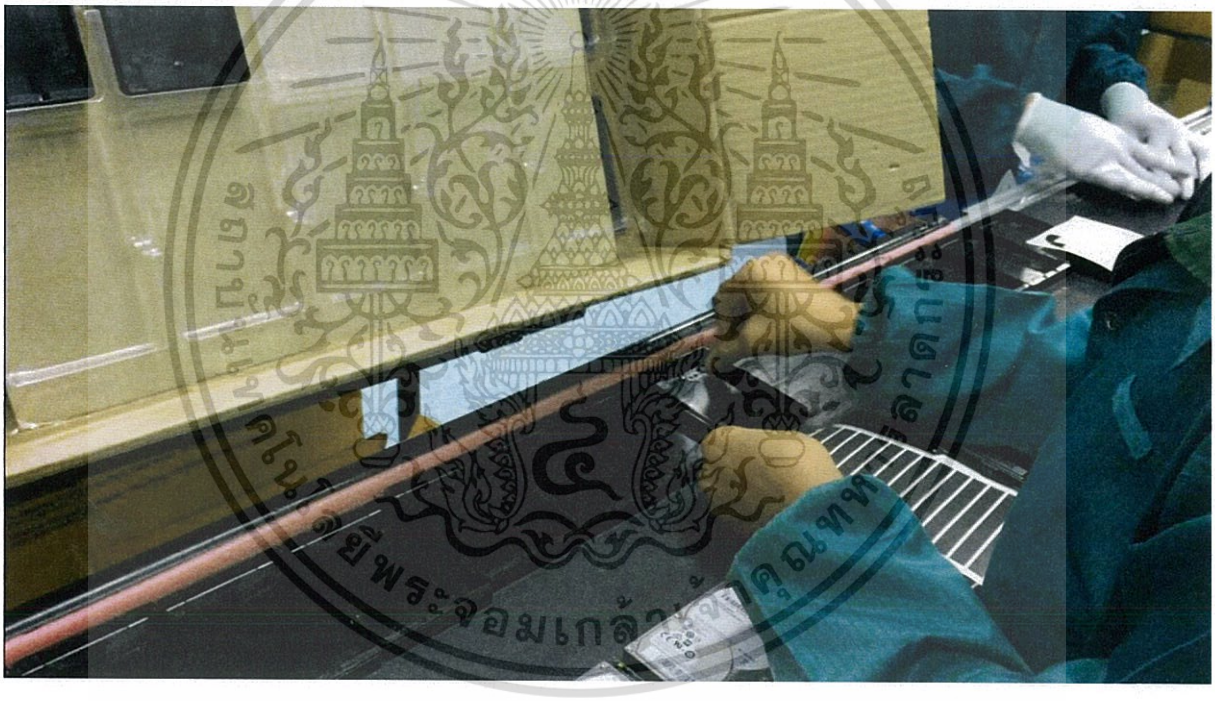
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

วิเคราะห์ สรุปและข้อเสนอแนะ

5.1 การวิเคราะห์

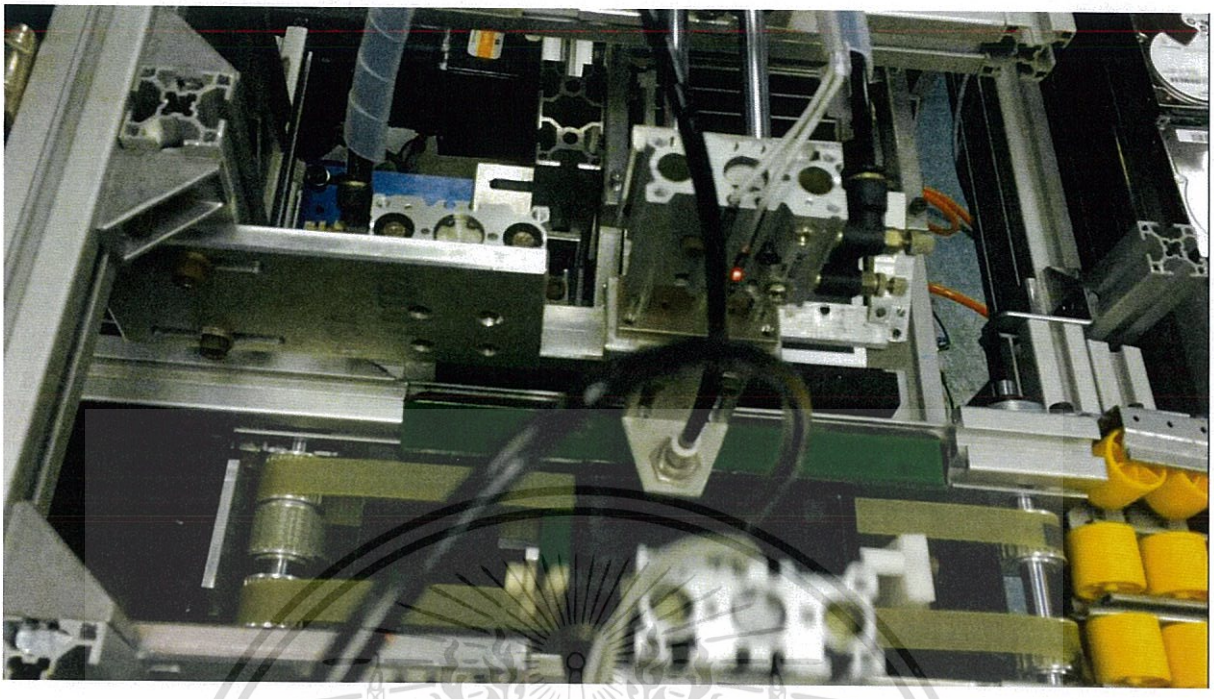
การติดแผ่นกันกระแทกโดยปกติแล้วจำเป็นต้องใช้พนักงาน 1 คน ซึ่งมีค่าแรง 300/วัน อีกทั้งมนุษย์ยังมีข้อจำกัดในการทำงาน เนื่องจากไม่สามารถทำงานได้ตลอด 24 ชม. และยังมีความเหนื่อยล้าใช้สายตาในการปฏิบัติงานอย่างสูง จึงอาจทำให้ในการปฏิบัติงานนั้นมีความผิดพลาดทำให้ชิ้นงานเสียหายและอาจเป็นปัญหาทางสายตาได้



รูปที่ 5.1 .ใช้มนุษย์ติดแผ่นกันกระแทก

ดังนั้นจากการที่สามารถนำเครื่อง Auto Sponge V.2.0 เข้ามาทดแทนมนุษย์ได้นั้นเป็นประโยชน์อย่างมากเนื่องจากเครื่อง Auto Sponge V.2.0 ไม่มีข้อจำกัดในการทำงานทางด้านเวลา ไม่มีความเหนื่อยล้าและสามาถปฏิบัติงานได้อย่างแม่นยำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ใชเครื่อง Auto Sponge V.2.0 ติดแผ่นกันกระแทก

จากวิธีการติดแผ่นกันกระแทกทั้งสองจึงสามารถเปรียบเทียบ Cycle Time ได้ดังนี้

ตารางที่ 5.1 Cycle Time ของมนุษย์ในการติดแผ่นกันกระแทก

จำนวนชิ้นงาน	Cycle Time
1	5 วินาที
12	60 วินาที

ตารางที่ 5.2 Cycle Time ของเครื่อง Auto Sponge V.2.0

จำนวนชิ้นงาน	Cycle Time
1	6 วินาที
12	60 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 สรุป

การพัฒนาสายการผลิตนั้น จำเป็นอย่างยิ่งที่จะต้องเข้าไปสังเกตุปัญหาภายในสายการผลิตว่ากระบวนการผลิตใดสามารถลดขั้นตอนหรือลดต้นทุนการผลิตลง ดังนั้นจึงเล็งเห็นว่ากระบวนการติดแผ่นกันกระแทกสามารถพัฒนาเครื่องจักรเข้ามาแทนคนได้จึงได้เกิดความคิดที่จะประดิษฐ์เครื่อง Auto Sponge โดยการใช้ความรู้ต่างๆทางด้านวิศวกรรม เช่น PLC ภาษา C# และนิวเมติกส์ เพื่อการพัฒนาเครื่องให้สามารถทำงานได้เทียบเท่ามนุษย์หรือดีกว่า

จากตารางการเปรียบเทียบ Cycle Time ระหว่างมนุษย์กับเครื่อง Auto Sponge V.2.0 จะเห็นได้ว่าเครื่อง Auto Sponge V.2.0 สามารถทำงานแทนมนุษย์ได้ อีกทั้งยังก้าวข้ามข้อจำกัดต่างๆที่มนุษย์ไม่สามารถทำได้และยังเป็นการลดต้นทุนการผลิตโดยการลดจำนวนคนลงไปได้ 1 คน และผลของประสิทธิภาพโดยรวมของเครื่องจักรหรือ OEE ยังมีค่าเท่ากับ 94.08% ดังนั้นจึงทำการนำเครื่อง Auto Sponge V.2.0 เข้าทำงานในสายการผลิตจริงและเครื่องก็สามารถปฏิบัติงานได้เป็นสัปดาห์โดยไม่มีปัญหาใด

5.3 ข้อเสนอแนะ

ปัญหาที่พบในการปฏิบัติงานนั้นคือ Auto Sponge V.1.0 เนื่องจากมอเตอร์สเตปที่นำมาใช้เกิดการ ERROR สาเหตุเกิดจากสัญญาณรบกวนที่ไม่สามารถแก้ไขได้จึงทำให้ต้องเปลี่ยนมอเตอร์และพัฒนามาเป็น Auto Sponge V.2.0 แต่ก็ยังมีปัญหาตรงที่เมื่อมอเตอร์เคลื่อนที่ไปตำแหน่งทุกจุดจนครบแล้วมอเตอร์ต้องเลื่อนกลับไปยังตำแหน่งที่ 1 แล้วจึงค่อย Feed แผ่นกันกระแทกออกมาอีกครั้งหนึ่ง ทำให้เครื่องจักรเสียเวลาในการทำงาน จึงมีข้อเสนอแนะว่าควรทำให้มอเตอร์ Feed ออกมายังตำแหน่งสุดท้ายแล้วเลื่อนกลับมาทีละสเตปเหมือนเดิม ซึ่งจะทำให้ประหยัดเวลา และมอเตอร์เสตปไม่ต้องเลื่อนกลับมา Feed ที่ตำแหน่งแรก ซึ่งสามารถแก้ไขได้ โดยการเพิ่ม PLC แต่ไม่สามารถทำแบบนี้ได้เพราะ Output ของ PLC ไม่เพียงพอ ถ้าเพิ่ม Output ได้ ก็จะสามารถแก้ไขปัญหาดังกล่าวได้ และอีกวิธีหนึ่งก็คือทำการเขียนโปรแกรมด้วยภาษา C# เองและใช้หลักการการส่งข้อมูลด้วยรหัสแอสกี้เพื่อส่ง Command ไปทำการควบคุมมอเตอร์สเตปได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] Darin. ความรู้พื้นฐานเกี่ยวกับเรื่อง PLC [อินเทอร์เน็ต]. กรุงเทพฯ : ภาควิชาครุศาสตร์เครื่องกล มหาลัยพระจอมเกล้าธนบุรี ; 2557 [เข้าถึงเมื่อ 16 พ.ย. 2558]. เข้าถึงได้จาก : http://www.tgcontrol.com/?p=3313.

[2] นายชลชัย เกงการเรือ, นายณัฐวุฒิ กาญจนลักษณ และ นางสาวมยุรี มานะ. เครื่องทำขนมโดนัทอัตโนมัติ [หลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต]. ชลบุรี: มหาวิทยาลัยบูรพา ; 2551. ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร มหาวิทยาลัยบูรพา.

[3] ทฤษฎีเกี่ยวกับ PLC, โปรแกรมเมเบิล ลอจิก คอนโทรลเลอร์, [อินเทอร์เน็ต], : [เข้าถึงเมื่อ 16 พ.ย. 2558]. เข้าถึงได้จาก : http://mte.kmutt.ac.th/elearning/Plc/unit_2.htm.

[4] คู่มือปฏิบัติการวิศวกรรมไฟฟ้าและคอมพิวเตอร์ การควบคุมด้วยพีแอลซี (PLC Control) [อินเทอร์เน็ต]. : ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยนเรศวร ; [เข้าถึงเมื่อ 16 พ.ย. 2558]. เข้าถึงได้จาก : www.ecpe.nu.ac.th/piyadanai/content/50_01/303407.../Lab5_PLC.doc.

[5] บทที่ 1, แนะนำให้รู้จัก PLC, [อินเทอร์เน็ต], : [เข้าถึงเมื่อ 16 พ.ย. 2558]. เข้าถึงได้จาก : http://www.tatc.ac.th/files/0902050883921_1106010774824.pdf.

[6] ผศ.ดร.นวภัทรา หนูนาค. เซนเซอร์ชนิดใช้แสง [อินเทอร์เน็ต]. ; 2555 [เข้าถึงเมื่อ 18 พ.ย. 2558]. เข้าถึงได้จาก : http://www.foodnetworksolution.com/wiki/word/7235/optical-sensor.

[7] Basics of Photoelectric Sensing, พื้นฐานของ Photoelectric, [อินเทอร์เน็ต]], : [เข้าถึงเมื่อ 18 พ.ย. 2558]. เข้าถึงได้จาก : http://www.compomax.co.th/product/basics-of-photoelectric-sensing/.

[8] งานไฟฟ้าและอิเล็กทรอนิกส์เบื้องต้น, รีเลย์, [อินเทอร์เน็ต], : [เข้าถึงเมื่อ 19 พ.ย. 2558]. เข้าถึงได้จาก : http://kpp.ac.th/elearning/elearning3/book-09.html

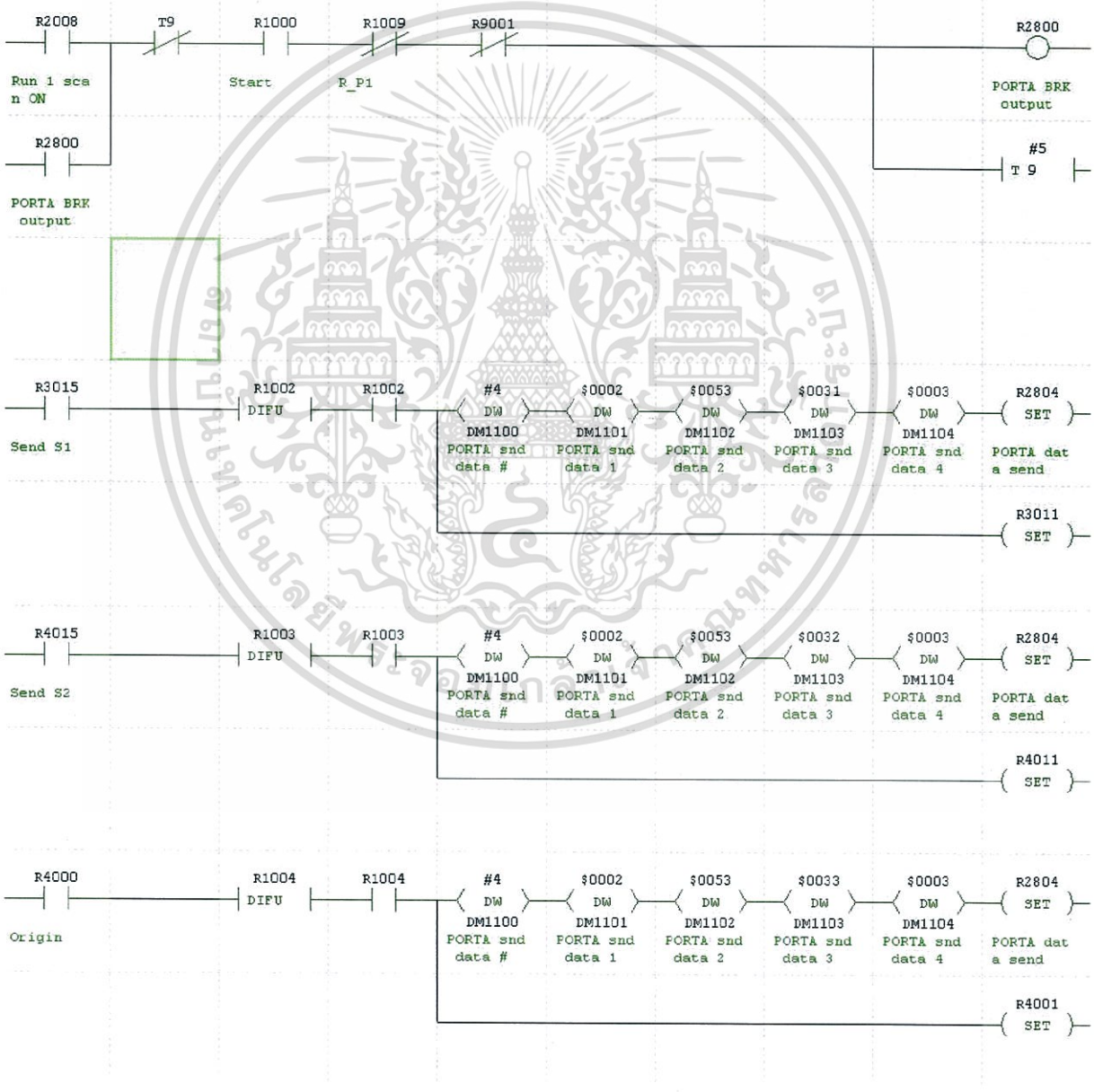
[9] บทที่ 3, วาลวควบคุมในระบบนิวแมติกส (Pneumatic Valves), [อินเทอร์เน็ต], : [เข้าถึงเมื่อ 17 พ.ย. 2558]. เข้าถึงได้จาก : utcc2.utcc.ac.th/engineer/learning/chalermchon_vis/.../Automation03.pdf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

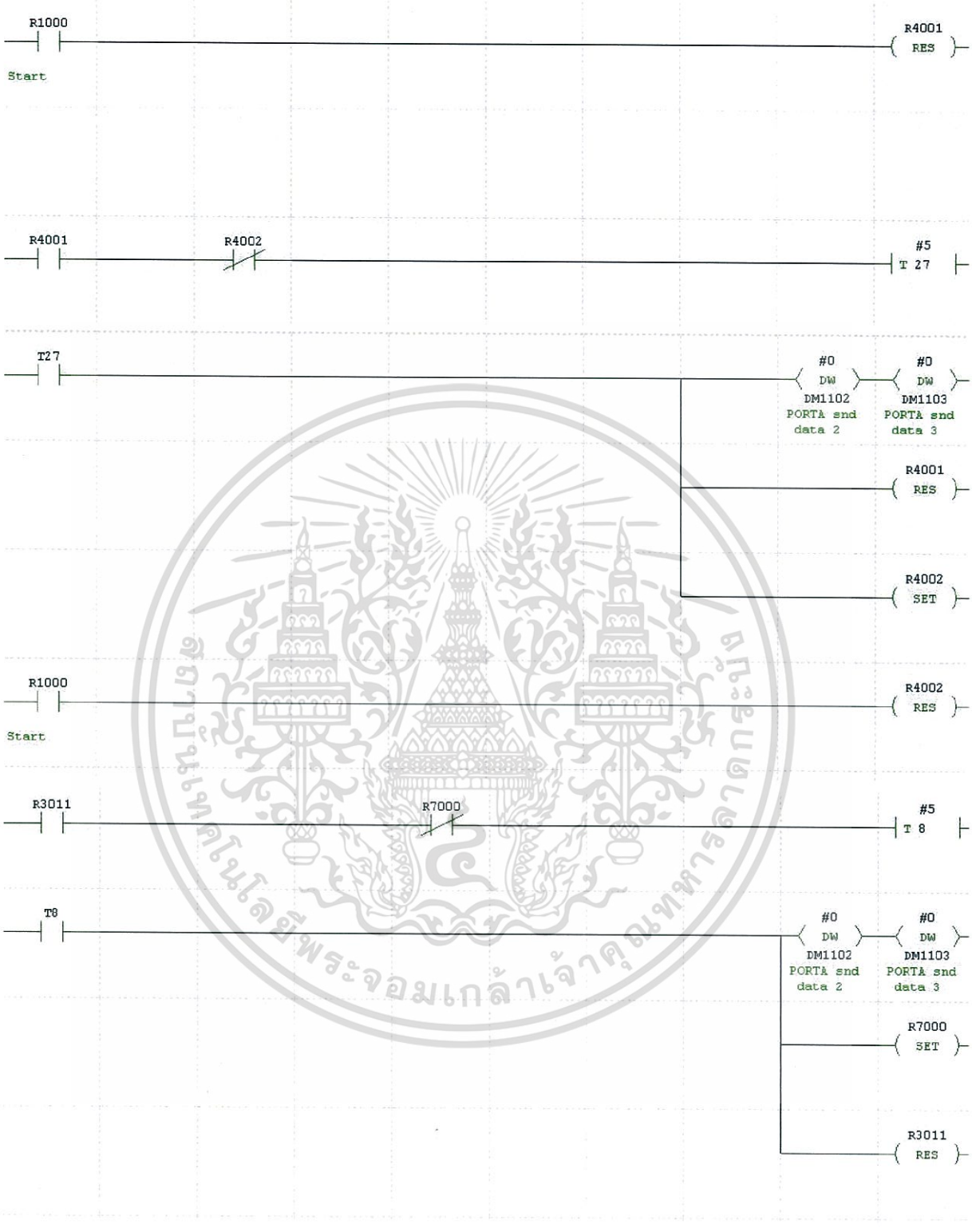
ภาคผนวก

แลดเดอร์การส่งข้อมูล

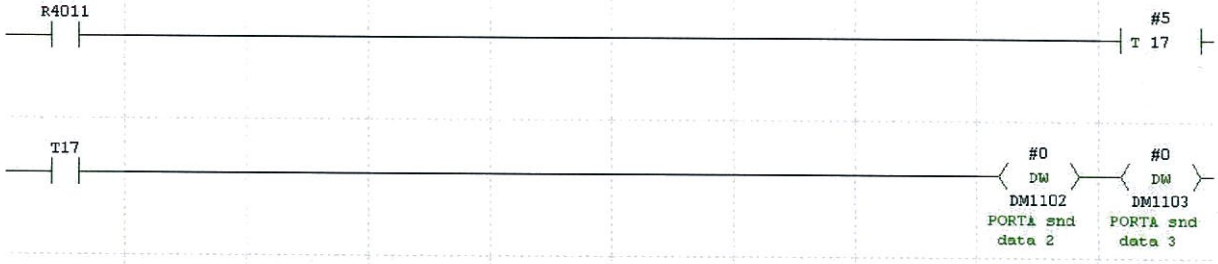
เป็นตัวอย่างการส่งข้อมูลจากพีแอลซีไปยังคอมพิวเตอร์โดยผ่าน RS232 โดยใช้รหัส เลขฐานสิบหก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

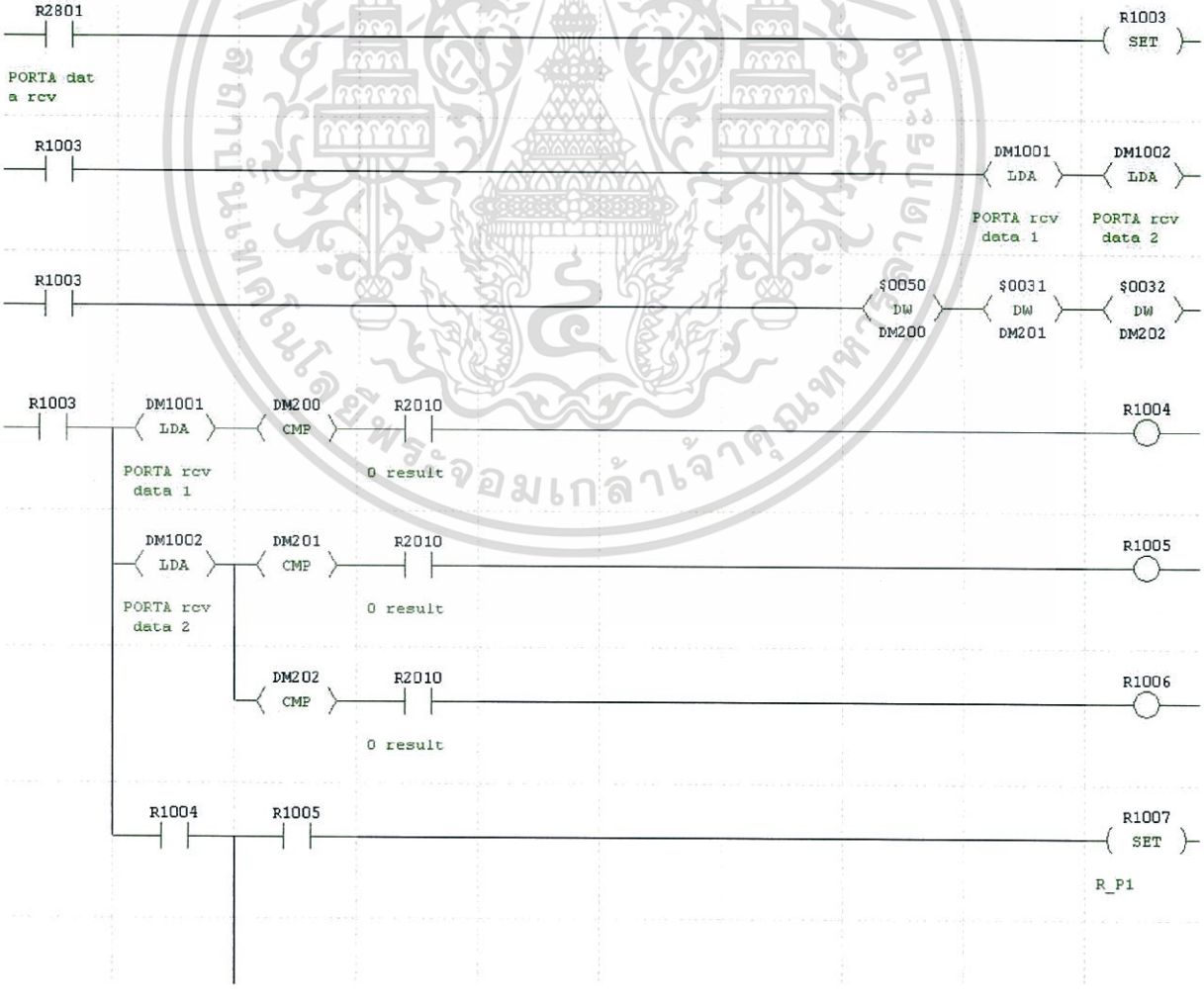


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

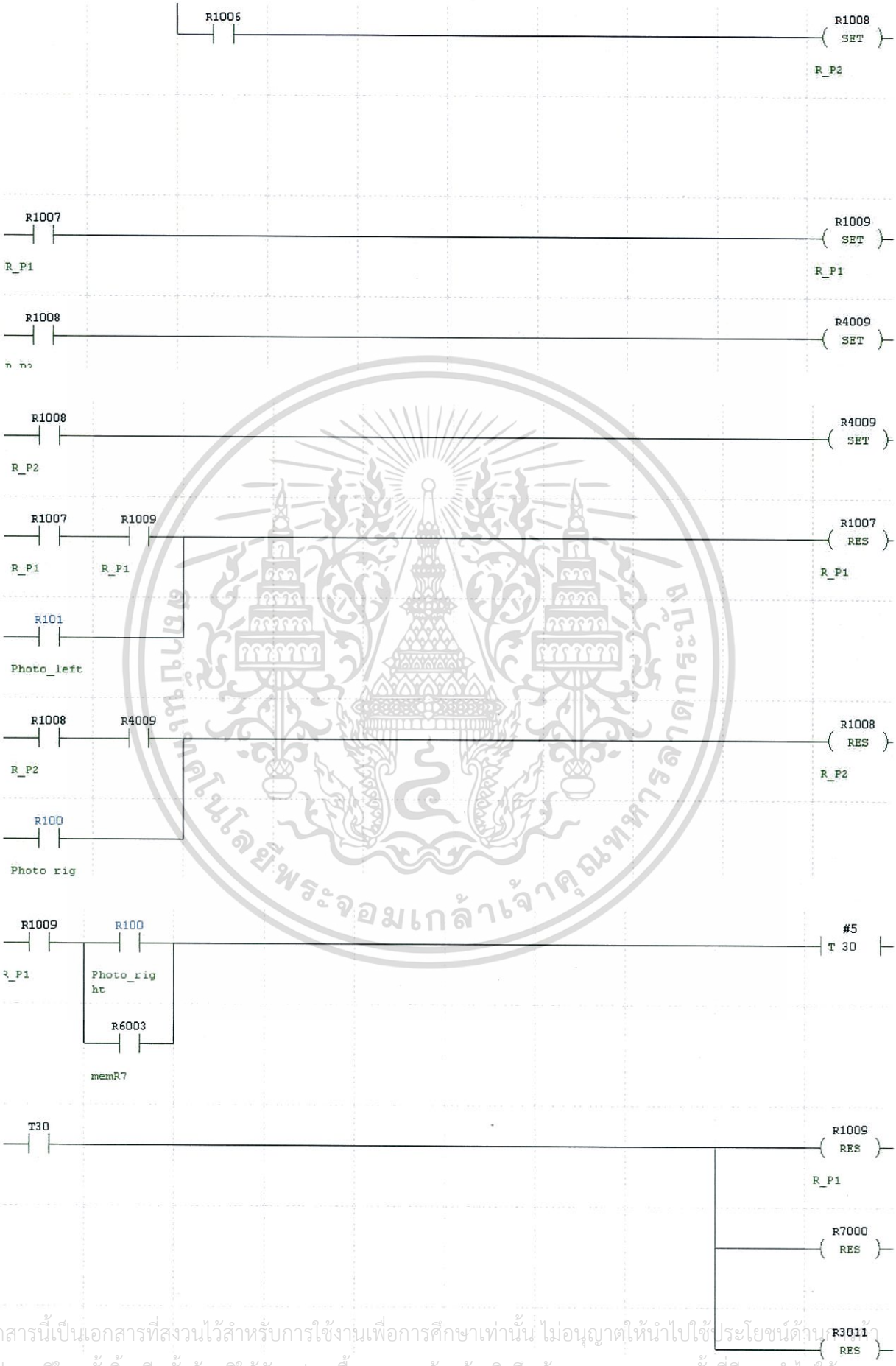


แลดเดอร์การรับข้อมูล

เป็นตัวอย่างการรับข้อมูลจากคอมพิวเตอร์โดยผ่าน RS232 โดยใช้รหัส เลขฐานสิบหก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดสำหรับการส่งข้อมูล

เป็นตัวอย่างการส่งข้อมูลจากคอมพิวเตอร์ไปยังพีแอลซีโดยผ่าน RS232 โดยใช้รหัส รหัส ASCII

```
private void button2_Click(object sender, EventArgs e)
{
    step = 0;
    count = 1;
    cmd = "";
    byte STX22 = 0x02;
    byte ETX22 = 0x03;
    string A = textBox1.Text.ToString();
    byte[] txDATA = System.Text.Encoding.ASCII.GetBytes(A);

    List<byte> completeProtocal = new List<byte>();
    completeProtocal.Add(STX22);
    completeProtocal.AddRange(txDATA);
    completeProtocal.Add(ETX22);
    if (ComPortPLC.IsOpen)
    {
        ComPortPLC.Write(completeProtocal.ToArray(), 0, completeProtocal.ToArray().Length);
    }
    textBox1.Clear();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดสำหรับการรับข้อมูล

เป็นตัวอย่างการรับข้อมูลจากพีแอลซีโดยผ่าน RS232 โดยใช้รหัส

```
string PLCData, PLCBuff = "";
private void Received2(object s, EventArgs e)
{
    PLCBuff += PLCData;
    char STX = (char)0x02;
    char ETX = (char)0x03;

    if (PLCBuff.IndexOf(ETX) > 0)
    {
        PLCBuff = PLCBuff.Substring(PLCBuff.IndexOf(STX) + 1, PLCBuff.IndexOf(ETX) - PLCBuff.IndexOf(STX) - 1);
        //richTextBox2.Text += PLCBuff + Environment.NewLine;
        richTextBox2.AppendText(PLCBuff + Environment.NewLine);
        richTextBox2.ScrollToCaret();


            if (PLCBuff == "S1" && CheckFinish == true)
            {

                AutoMode();


            }
            else  if (PLCBuff == "S2" && CheckFinish2 == true)
             {

                AutoMode2();
```

โค้ดสำหรับการส่งข้อมูล

เป็นตัวอย่างการส่งข้อมูลจากคอมพิวเตอร์ไปยังเสต็ปมอเตอร์โดยผ่าน RS232 โดยใช้รหัส

รหัส ASCII

```
private void button4_Click(object sender, EventArgs e)
{
    try
    {

        byte ETX33 = 0x0d;
        string A = textBox2.Text.ToString();
        string B = "?96.1";


        //byte cr = 0x0D;
        byte[] txDATA = Encoding.ASCII.GetBytes(A);
        byte[] txDATA2 = Encoding.ASCII.GetBytes(B);
        //byte[] txDATA = Encoding.ASCII.GetBytes(A);

        List<byte> completeProtocal = new List<byte>(txDATA);
        List<byte> completeProtocal2 = new List<byte>(txDATA2);


        completeProtocal.Add(ETX33);
        completeProtocal2.Add(ETX33);

        if (ComPortMotor.IsOpen)
        {
            ComPortMotor.Write(completeProtocal.ToArray(), 0, completeProtocal.ToArray().Length);
            ComPortMotor.Write(completeProtocal2.ToArray(), 0, completeProtocal2.ToArray().Length);
            //ComPort.Write(A);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
            if (ComPortMotor2.IsOpen)
            {
                ComPortMotor2.Write(completeProtocal.ToArray(), 0, completeProtocal.ToArray().Length);
                ComPortMotor2.Write(completeProtocal2.ToArray(), 0, completeProtocal2.ToArray().Length);
                //ComPort.Write(A);


            }
        }
        catch (Exception)
        {

        }
        //textBox1.Clear();
    }
```

โค้ดสำหรับการรับข้อมูล

เป็นตัวอย่างการรับข้อมูลจากเสต็ปมอเตอร์โดยผ่าน RS232

```
if (ComPortMotor.IsOpen == true)
{

    MotorBuff += MotorData;
    //point = point.Substring(1, point.Length - 2);
    pointmotor = MotorBuff;
    if (pointmotor.Contains('\r'))
    {


        string[] x = pointmotor.Split('\r', '\n');

        for (int i = 0; i < x.Length; i++)
        {
            if (x[i].IndexOf("Px.1") != -1)
            {
                string[] position = x[i].Split('=');
                if (position.Length > 1)
                    textBox4.Text = position[1];
            }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ-นามสกุล สรกฤษณ์ วิทยานันท์

วัน เดือน ปีเกิด 5 กันยายน 2537

ที่อยู่ 6/1 ม.8 ต.ซึ้ง อ.ขลุง จ.จันทบุรี 22110

E-mail: yeans045@gmail.com

Tel. 097-130-1328

ประวัติการศึกษา

พ.ศ. 2554 จบมัธยมศึกษาชั้นปีที่ 6 แผนกวิทยาศาสตร์และคณิตศาสตร์
โรงเรียนเบญจมราชูทิศ จันทบุรี

พ.ศ. 2558-ปัจจุบัน กำลังศึกษาหลักสูตร วิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมระบบการผลิต วิทยาลัยนวัตกรรมการผลิตขั้นสูง
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประวัติการทำงานและผลงานวิจัย

พ.ศ. 2558 นักศึกษาฝึกงานภาคฤดูร้อน/สหกิจศึกษา
แผนก Electronic Engineering (Automation)
บริษัท แคล-คอมพ์ อีเล็คโทรนิคส์ (ประเทศไทย) จำกัด (มหาชน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้