สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การออกแบบเครื่องควบคุมแบบตรรกและระบบโครงข่ายผ่านโฮสคอมพิวเตอร์

DESIGN OF PROGRAMMABLE LOGIC CONTROLLERS AND NETWORK SYSTEM THROUGH HOST COMPUTER

นาย ทวีพล ซื่อสัตย์

MR. TAWEEPOL SUESUT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2541

ISBN 974-622-251-1

เลขหมู่.....
เลขทะเบียน 30947
วัน, เดือน, ปี 4 ก.ย. 2541

ลิขสิทธิ์ของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DESIGN OF PROGRAMMABLE LOGIC CONTROLLERS AND NETWORK SYSTEM THROUGH HOST COMPUTER

TAWEEPOL SUESUT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

1998

ISBN 974-622-251-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การออกแบบเครื่องควบคุมแบบตรรกและระบบโครงข่ายผ่านโฮสคอมพิวเตอร์

DESIGN OF PROGRAMMABLE LOGIC CONTROLLERS AND NETWORK SYSTEM THROUGH HOST COMPUTER

ชื่อนักศึกษา นายทวีพล ซื่อสัตย์ **รหัสประจำตัว** 39061015

หลักสูตร วิศวกรรมศาสตรมหาบัณฑิต **สาขาวิชา** วิศวกรรมไฟฟ้า

อาจารย์ผู้ควบคุมวิทยานิพนธ์ ผศ.สุพรรณ กุลพาณิชย์

อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม ผศ.วิริยะ กองรัตน์

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
ผศ.สุพรรณ	กุลพาณิชย์	
ผศ.วิริยะ	กองรัตน์	
รศ.กิตติ	ตีรเศรษฐ	
รศ.พิพัฒน์	เลาหสงคราม	
ผศ.ประภาษ	อุคคกิมาพันธุ์	

ค่าระดับคะแนนที่ผ่านเป็นเอกฉันท์จากคณะกรรมการสอบ GOOD

วัน/เดือน/ปี ที่สอบ 10 กรกฎาคม 2541 เวลา 10.00 - 12.00 น.

สถานที่สอบ ห้องสอบวิทยานิพนธ์ คณะวิศวกรรมศาสตร์ ตึก 12 ชั้น ชั้น 4 ห้อง (E12-404)

บัณฑิตวิทยาลัยรับรองแล้ว

(รศ.ดร.มนัส สังวรศิลป์)

คณบดีบัณฑิตวิทยาลัย

วันที่ 17 **เดือน** กรกฎาคม **พ.ศ.** 2541

หมายเหตุ การวัดผลวิทยานิพนธ์ให้ใช้ค่าระดับคะแนนดังนี้

ค่าระดับคะแนน	ผลการศึกษา
O	Outstanding (ดีเยี่ยม)
G	Good (ดี)
P	Pass (ผ่าน)
F	Fail (ไม่ผ่าน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การออกแบบเครื่องควบคุมแบบตรรก และ ระบบโครงข่ายผ่านโฮส คอมพิวเตอร์
นักศึกษา	นาย ทวีพล ซื่อสัตย์
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ. สุพรรณ กุลพาณิชย์
อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม	ผศ. วิริยะ กองรัตน์
ระดับการศึกษา	วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง
พ.ศ.	2541

บทคัดย่อ

วิทยานิพนธ์นี้ นำเสนอการออกแบบเครื่องควบคุมแบบตรรกที่โปรแกรมได้ ทั้งในส่วนของฮาร์ดแวร์ ซอฟแวร์ และการประยุกต์ใช้งาน โดยทางด้านฮาร์ดแวร์ประกอบด้วยไมโครโปรเซสเซอร์เบอร์ Z84C11 เป็นตัวประมวลผลกลาง มีจำนวนอินพุท 24 จุด เอาท์พุท 12 จุด ชนิด OPTO ISOLATED มีแป้นคีย์สำหรับป้อนโปรแกรมภายในตัว และฟังก์ชั่นคำสั่งพิเศษ เช่น SHIFT REGISTER สำหรับช่วยในการเขียนโปรแกรมที่ทำงานเป็นลำดับ และมีฟังก์ชั่นการตรวจสอบการทำงานของเครื่องในขณะทำงานอยู่ เป็นต้น นอกจากนี้เครื่องควบคุมยังถูกออกแบบให้สามารถเชื่อมโยงกันได้เป็นระบบโครงข่ายที่มีการควบคุมสู่ศูนย์กลางผ่านโฮสคอมพิวเตอร์ โดยมีโปรแกรมซึ่งออกแบบมาเพื่อใช้สนับสนุนการทำงานของเครื่องควบคุม เช่น การติดตามผล การสังเกตการ การควบคุมระยะไกล การอ่านเขียนข้อมูล เพื่อจัดเก็บลงบนแฟ้มข้อมูลของเครื่องคอมพิวเตอร์ และมีฟังก์ชั่นอื่นๆ ที่จำเป็นเพื่ออำนวยความสะดวกในการใช้งานอีกจำนวนมาก ระบบโครงข่ายนี้ออกแบบการเชื่อมต่อสื่อสารตามมาตรฐาน RS 485 ระบบออกแบบให้เชื่อมโยงกันได้ 8 จุด และระยะทางในการติดต่อควบคุมได้นับร้อยเมตร ดังนั้นการควบคุมหรือแก้ไขข้อผิดพลาดของเครื่องควบคุมเครื่องจักรที่ถูกจัดวางในสถานที่ที่ต่างกัน และมีระยะทางไกลออกไปจึงสามารถกระทำได้อย่างสะดวกและไม่ยุ่งยากมากนัก ซึ่งเป็นการเพิ่มประสิทธิภาพในการทำงานให้รวดเร็วและครอบคุมทั้งระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Design of Programmable Logic Controllers and Network System through Host computer
Student	Mr. Taweepol Suesut
Thesis Advisor	Asst. Prof. Suphan Kulphanich
Thesis Co Advisor	Asst. Prof. Viriya Kongratana
Level of study	Master of Engineering in Electrical Engineering King Mongkut's Institute of Technology Ladkrabang
Year	1998

Abstract

This thesis presents a design of hardware, software and application of PLC using microprocessor No. Z84C11 as the Central Processing unit (CPU). The PLC consists of opto-isolated 24 inputs and 12 outputs , the boolean instruction program can be written down to PLC through build in keyboard, and the special function, such as shift register function , the monitoring function which used to observe the status of PLC operation and machine during the operation is also included. More over the centralized control network system using host computer to control the operations of many PLC in the same time is designed too. At the control station. The special program such as remote control function monitoring function, data read-write function is also designed in order to support the operation of each PLC at end terminal. The interface of this network is designed using standard RS 485 which can be used to link PLC up to 8 terminals, with the few hundred meters communication distance applicable using this network, the operation control, monitoring and error detection of each PLC and process or machine and control which place in the different area can be easily investigate and correction that is the higher efficiency of the overall system is achieve.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การเรียบเรียงวิทยานิพนธ์ฉบับนี้กว่าที่จะสำเร็จลุล่วงไปได้ ต้องประสบกับปัญหามากมาย ทั้งที่เกี่ยวข้องกับตัววิทยานิพนธ์ และที่เกี่ยวกับตัวผู้จัดทำเอง แต่ด้วยเพราะความช่วยเหลือและความอนุเคราะห์จากบุคคลหลายท่านด้วยกัน จึงทำให้ผู้จัดทำฝ่าฟันอุปสรรคเหล่านั้นไปได้ ซึ่งผู้จัดทำต้อง ขอขอบพระคุณมา ณ ที่นี้

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ สุพรรณ กุลพาณิชย์ อาจารย์ที่ปรึกษาที่เป็นมากกว่าอาจารย์ที่ปรึกษา คอยให้กำลังใจและคำปรึกษาที่ดีมากเหลือเกิน

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ วิริยะ กองรัตน์ ที่คอยช่วยให้ความรู้และคำแนะนำที่มีคุณค่าอย่างยิ่งในการทำวิทยานิพนธ์

ขอขอบพระคุณ รองศาสตราจารย์ กิตติ ตรีเศรษฐ ที่คอยช่วยให้คำแนะนำและกระตุ้นเตือนจนวิทยานิพนธ์สำเร็จตามเวลาอันสมควร

ขอขอบพระคุณท่านอาจารย์ทุกท่านในภาควิชาที่ให้ความรู้ คำแนะนำ คำปรึกษา และกำลังใจ อันมีคุณค่ามากสำหรับผู้จัดทำ

ขอขอบพระคุณ นาย สัมฤทธิ์ ไชยสมบัติ เพื่อนรักผู้ล่วงลับไปแล้ว ที่ทำให้ผู้จัดทำมีชีวิตอยู่จนถึงทุกวันนี้

ขอขอบคุณพี่ๆน้องๆและ เพื่อนๆ ทุกคนที่มีส่วนร่วมในให้กำลังใจ กำลังกาย จนทำให้การทำวิทยานิพนธ์ สำเร็จลุล่วงไปได้ด้วยดี

ที่สำคัญมากที่สุดก็คือ คุณพ่อ คุณแม่ และ คุณยาย ที่รักและเคารพที่คอยเป็นกำลังใจอย่างดีเยี่ยมจนงานสำเร็จไปได้

ทวีพล ซื่อสัตย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... I

บทคัดย่อภาษาอังกฤษ..... II

กิตติกรรมประกาศ..... III

สารบัญ..... IV

สารบัญตาราง..... VII

สารบัญภาพ..... VIII

บทที่

1 บทนำ..... 1

2 ทฤษฎีและหลักการพื้นฐานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้..... 2

2.1 ที่มาและวิวัฒนาการของเครื่องควบคุมแบบตรรกที่โปรแกรมได้..... 2

2.1.1 วิวัฒนาการของเครื่องควบคุม PLC/PC..... 2

2.1.2 สรุปข้อดีของเครื่องควบคุม PLC/PC..... 3

2.2 โครงสร้างและหลักการทำงานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้..... 4

2.2.1 หน่วยประมวลผลกลาง..... 5

2.2.2 หน่วยอินพุทและเอาท์พุท..... 9

2.2.3 หน่วยป้อนโปรแกรม..... 12

2.3 ลักษณะการโปรแกรมให้กับเครื่องควบคุมแบบตรรกที่โปรแกรมได้..... 14

2.3.1 โปรแกรมภาษาแลดเดอร์ไดอะแกรม..... 15

2.3.2 โปรแกรมภาษาคำสั่งบูลีน..... 17

3 เครื่องควบคุมแบบตรรกที่โปรแกรมได้กับการติดต่อสื่อสาร..... 18

3.1 การติดต่อสื่อสารข้อมูลทั่วไป..... 18

3.1.1 วิธีการสื่อสารข้อมูล..... 18

3.1.2 ข้อตกลงในการติดต่อสื่อสาร..... 21

3.1.3 การควบคุมความผิดพลาดในการส่งข้อมูล..... 22

3.1.4 อัตราการส่งข้อมูล..... 23

3.1.5 มาตรฐานในการสื่อสารข้อมูลแบบอนุกรม..... 24

3.2 รูปแบบของการติดต่อสื่อสารของเครื่องควบคุมแบบตรรกที่โปรแกรมได้..... 27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่		หน้า
	3.2.1 การต่อวงจรสื่อสารแบบจุดต่อจุด	27
	3.2.2 การต่อวงจรสื่อสารแบบหลายจุด	28
	3.2.3 การต่อวงจรสื่อสารแบบเครือข่ายท้องถิ่น(LAN)	31
	3.3 ข้อตกลงในการสื่อสารของเครื่องควบคุมแบบตรรกที่โปรแกรมได้	32
	3.3.1 รูปแบบของบล็อก(Block Format)	32
	3.2.1 การคำนวณ FCS	33
4	การออกแบบและพัฒนาเครื่องควบคุมแบบตรรกที่โปรแกรมได้	35
	4.1 คุณสมบัติของเครื่องควบคุมแบบตรรกที่โปรแกรมได้	35
	4.1.1 ทางด้านฮาร์ดแวร์	35
	4.1.2 ทางด้านซอฟแวร์	35
	4.2 การออกแบบวงจร	36
	4.2.1 หน่วยประมวลผลกลาง	36
	4.2.2 หน่วยความจำ	37
	4.2.3 หน่วย อินพุทและเอาท์พุท	37
	4.2.4 หน่วยแสดงผลและรับข้อมูลกับผู้ใช้	39
	4.2.5 หน่วยสร้างฐานเวลา	41
	4.3 ระบบปฏิบัติการ	42
	4.3.1 การแบ่งพื้นที่หน่วยความจำ	42
	4.3.2 โปรแกรมผู้ใช้และคำสั่งสำหรับเครื่องควบคุม	44
	4.3.3 การจัดการข้อมูลของโปรแกรมผู้ใช้	45
	4.3.4 การแปลชุดคำสั่งจากโปรแกรมผู้ใช้เป็นภาษาเครื่อง	48
	4.3.5 การทำงานของฟังก์ชั่นควบคุม	49
5	การออกแบบระบบโครงข่าย สำหรับเครื่องควบคุมแบบตรรกที่โปรแกรมได้	54
	5.1 การออกแบบทางด้านฮาร์ดแวร์	57
	5.2 การออกแบบทางด้านซอฟแวร์	57
	5.3 ข้อตกลงในการติดต่อสื่อสาร	57
6	การออกแบบโปรแกรมสนับสนุนการทำงานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่		หน้า
	6.1 คุณสมบัติทั่วไปของโปรแกรมสนับสนุนการทำงาน.....	64
	6.2 การออกแบบโปรแกรม เพื่อการตรวจสอบสภาวะ การทำงานของเครื่องควบคุมที่อยู่ในโครงข่าย.....	64
	6.3 การออกแบบโปรแกรม เพื่อการถ่ายเทข้อมูลระหว่างเครื่องควบคุมPLC/PC กับคอมพิวเตอร์.....	69
	6.4 การออกแบบโปรแกรม เพื่อการจัดการแฟ้มข้อมูล.....	72
	6.4.1 การบันทึกแฟ้มข้อมูล.....	72
	6.4.2 การเปิดแฟ้มข้อมูล.....	73
	6.4.3 การพิมพ์แฟ้มข้อมูลออกทางเครื่องพิมพ์.....	74
	6.5 การออกแบบโปรแกรมเพื่อการเชื่อมโยงโครงข่ายผ่านโฮสคอมพิวเตอร์.....	75
7	การประยุกต์ การใช้งานและผลการทดสอบ	
	7.1 การประยุกต์ใช้งานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้.....	76
	7.1.1 ตัวอย่างการควบคุมที่ 1.....	76
	7.1.2 ตัวอย่างการควบคุมที่ 2.....	81
	7.2 การประยุกต์ใช้งานของเครื่องควบคุมร่วมกับโปรแกรมสนับสนุนการทำงาน	85
	7.2.1 หน้าที่ของปุ่มคำสั่งในโปรแกรม	86
	7.2.2 การทดสอบการเชื่อมโยงข้อมูลโครงข่ายผ่านโฮสคอมพิวเตอร์.....	93
8	บทสรุป.....	96
บรรณานุกรม.....		97
ภาคผนวก.....		98
ภาคผนวก ก	ผลงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์ที่ได้รับการตีพิมพ์.....	99
	บทความวิจัยในการประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 19 เรื่อง การออกแบบโปรแกรมโฮสคอมพิวเตอร์กับครื่องควบคุมแบบตรรกจำนวนหลายจุด	100
ภาคผนวก ข	รายละเอียดข้อมูลของไอซีเบอร์ Z84C11 เบอร์ 75176	110
ภาคผนวก ค	ตัวอย่างการพิมพ์ชุดคำสั่งบูลีน.....	118
ประวัติผู้เขียน.....		122

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2-1 อุปกรณ์อินพุท/เอาท์พุทแบบค่าสภาวะลอจิก	10
2-2 อุปกรณ์อินพุท/เอาท์พุทแบบค่าตัวเลข	11
2-3 อุปกรณ์อินพุท/เอาท์พุทแบบค่าสภาวะหลายบิท	11
2-4 ตัวอย่างการใช้ฟังก์ชั่นลอจิกและสัญลักษณ์พื้นฐานการเขียนแลดเดอร์ไดอะแกรม	15
2-5 โปรแกรมชุดคำสั่งบูลีน	17
3-1 แสดงค่าเปรียบเทียบคุณสมบัติทางไฟฟ้าของ RS232C, RS422 และ RS485	26
4-1 แสดงการแบ่งพื้นที่ใช้งานของหน่วยความจำ	42
4-2 แสดงตารางอินพุทเอาท์พุทและรีเลย์ภายใน	43
4-3 แสดงความสัมพันธ์ของสมการบูลีนกับชุดคำสั่งบูลีน	44
4-4 แสดงการเก็บข้อมูลของโปรแกรมผู้ใช้ในหน่วยความจำ	46
4-5 แสดงตารางคำสั่งที่ได้ออกแบบไว้	46
4-6 การจัดเก็บรหัสจากโปรแกรมคำสั่งบูลีน	47
4-7 แสดงการแปลชุดคำสั่งเป็นภาษาเครื่อง	48
7-1 แสดงการกำหนดอินพุทเอาท์พุทของชุดสายพานลำเลียงคัดเลือกกล่อง	78
7-3 แสดงการกำหนดอินพุทเอาท์พุทของชุดสายพานลำเลียงยกกล่องอัตโนมัติ	82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่	หน้า
2-1 บล็อกไดอะแกรมของPLC/PC	5
2-2 บล็อกไดอะแกรมของ CPU	5
2-3 การ SCAN ของ CPU	6
2-4 ข้อมูลเป็นบิทไบท์และเวร์ด	7
2-5 หน่วยความจำของ PLC/PC	8
2-6 หน่วยความจำสำหรับผู้ใช้	8
2-7 บล็อกไดอะแกรมของหน่วยอินพุทเอาท์พุท	9
2-8 เครื่องป้อนโปรแกรมแบบ CRT	13
2-9 เครื่องป้อนโปรแกรมขนาดเล็กแบบพกพา	13
2-10 วงจรรีเลย์ควบคุมการหมุนกลับทางของมอเตอร์แบบ JOGGING	16
2-11 แลดเดอร์ไดอะแกรมควบคุมการหมุนกลับทางของมอเตอร์แบบ JOGGING	16
2-12 แสดงการแบ่งแลดเดอร์ไดอะแกรมออกเป็นรังค์	16
3-1 แสดงความสัมพันธ์ของส่วนประกอบในการสื่อสารข้อมูล	18
3-2 การส่งแบบทิศทางเดียว	19
3-3 การส่งแบบฮาร์ฟดูเพล็กซ์	19
3-4 การส่งแบบสองทิศทาง	19
3-5 การสื่อสารข้อมูลแบบขนาน	20
3-6 การสื่อสารข้อมูลแบบอนุกรม	20
3-7 แสดงโครงสร้างของ PACKET	22
3-8 แสดงรูปแบบการตรวจสอบด้วย Parity Bit	23
3-9 แสดงโครงสร้างของการสื่อสารแบบ RS 232 C	24
3-10 แสดงโครงสร้างของการสื่อสารแบบ RS 422	25
3-11 แสดงโครงสร้างของการสื่อสารแบบ RS 485	26
3-12 ระบบการติดต่อแบบจุดต่อจุด	28
3-13 การเชื่อมโยงแบบอินพุทและเอาท์พุทระยะไกล	28
3-14 แสดงพื้นที่ของตัวควบคุมหลักขณะทำงานปกติ	28
3-15 แสดงพื้นที่ของตัวควบคุมหลักขณะต่อร่วมกับชุดอินพุท/เอาท์พุทระยะไกล	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
3-16 การต่อแบบ Serial Link	29
3-17 การต่อแบบ Parallel Link	30
3-18 การแบ่งพื้นที่สำหรับระบบ PC Link	30
3-19 ระบบการติดต่อสื่อสารแบบเครือข่ายท้องถิ่น	31
3-20 แสดงระบบโครงข่ายหลายจุดแบบ RS 485	31
3-21 แสดงการติดต่อระหว่างคอมพิวเตอร์กับเครื่องควบคุมPLC/PC	32
3-22 แสดงรูปแบบของบล็อก	32
3-23 แสดงรูปแบบของบล็อกและการคำนวณ FCS	33
4-1 เครื่องควบคุมPLC/PC ที่ออกแบบขึ้นมา	36
4-2 แสดงวงจรหน่วยประมวลผลประกอบกับหน่วยความจำ	37
4-3 แสดงวงจรหน่วยอินพุท	38
4-4 แสดงวงจรหน่วยเอาท์พุท	39
4-5 แสดงวงจรหน่วยแสดงผลและรับข้อมูลของผู้ใช้	40
4-6 แสดงชุดคีย์ป้อนโปรแกรมสำหรับผู้ใช้	41
4-7 แสดงให้เห็นภายในของเครื่องควบคุม PLC/PC	41
4-8 แสดงแผนผังการทำงานในวงรอบ	45
5-1 แสดงระบบโครงข่ายผ่านโฮสคอมพิวเตอร์ที่ทำการออกแบบ	54
5-2 แสดงส่วนของการติดต่อสื่อสารอนุกรม	55
5-3 แสดงวงจรสื่อสารข้อมูลอนุกรม	55
5-4 อุปกรณ์แปลงสัญญาณจาก RS 232 C เป็น RS 485	56
5-5 วงจรแปลงสัญญาณจาก RS 232 C เป็น RS 485	56
5-6 แสดงรูปแบบของบล็อกที่ออกแบบ	57
6-1 แสดงแผนผังการคำนวณ FCS	65
6-2 แสดงแผนผังการควบคุมการทำงานของอุปกรณ์แปลงสัญญาณจาก RS 232 C เป็น RS 485	66
6-3 แสดงแผนผังการตรวจสอบสภาวะการทำงาน	67
6-4 แสดงแผนผังการกำหนดค่า (Force)	68
6-5 แสดงแผนผังของการถ่ายเทข้อมูลจากเครื่องควบคุมPLC ไปยัง คอมพิวเตอร์	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
6-6 แสดงแผนผังของการถ่ายเทข้อมูลจาก คอมพิวเตอร์ ไปยังเครื่องควบคุมPLC	71
6-7 แสดงแผนผังของการบันทึกข้อมูลลงแฟ้มข้อมูล.....	72
6-8 แสดงแผนผังของการเปิดแฟ้มข้อมูล.....	73
6-9 แสดงแผนผังของการพิมพ์แฟ้มข้อมูล.....	74
6-10 แสดงแผนผังของการเชื่อมโยงโครงข่ายผ่านโฮสคอมพิวเตอร์.....	75
7-1 แสดงชุดสายพานลำเลียงคัดเลือกกล่อง(ด้านบน).....	76
7-2 แสดงชุดสายพานลำเลียงคัดเลือกกล่อง(ด้านข้าง).....	77
7-3 แสดงโปรแกรมแลดเดอร์ของชุดสายพานลำเลียงคัดเลือกกล่อง.....	79
7-4 แสดงวงจรการเดินสายไฟของชุดสายพานลำเลียงคัดเลือกกล่อง.....	80
7-5 แสดงการเดินสายระหว่างเครื่องควบคุมกับชุดสายพานลำเลียงคัดเลือกกล่อง.....	80
7-6 แสดงชุดสายพานลำเลียงยกกล่องอัตโนมัติ(ด้านบน).....	81
7-7 แสดงชุดสายพานลำเลียงยกกล่องอัตโนมัติ(ด้านข้าง).....	81
7-8 แสดงโปรแกรมแลดเดอร์ของชุดสายพานลำเลียงยกกล่องอัตโนมัติ.....	83
7-9 แสดงวงจรการเดินสายไฟของชุดสายพานลำเลียงยกกล่องอัตโนมัติ.....	84
7-10 แสดงการเดินสายระหว่างเครื่องควบคุมกับชุดสายพานลำเลียงยกกล่องอัตโนมัติ.....	84
7-11แสดงการทดสอบระบบโครงข่ายผ่านโฮสคอมพิวเตอร์.....	85
7-12 แสดงสายนำสัญญาณความยาว 100 เมตร.....	85
7-13 แสดงการเริ่มต้นเข้าสู่โปรแกรม.....	86
7-14 แสดงหน้าที่ของปุ่มคำสั่งในโปรแกรม.....	86
7-15 การเขียนโปรแกรมจาก Tool Box.....	87
7-16 แสดงชุดคำสั่งบูลีนบน Editor.....	88
7-17 การบันทึกโปรแกรมชุดคำสั่งบูลีนลงหน่วยความจำ.....	88
7-18 แสดงการเปิดไฟล์ข้อมูลจากหน่วยความจำ.....	89
7-19 แสดงการถ่ายเทโปรแกรมจากคอมพิวเตอร์ไปยังเครื่องควบคุม.....	89
7-20 แสดงการถ่ายเทโปรแกรมจากเครื่องควบคุมไปยังคอมพิวเตอร์.....	90
7-21 แสดงการเลือกตรวจสอบสภาวะ.....	91
7-22 แสดงการตรวจสอบสภาวะของอินพุทเอาท์พุท.....	91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
7-23 แสดงการตรวจสอบสภาวะของตัวตั้งเวลา.....	92
7-24 การเปลี่ยนแปลงค่าของตัวตั้งเวลา.....	92
7-25 แสดงผังการทำงานร่วมกันของชุดสายพานทั้ง 2 ชุด.....	93
7-26 แสดงการเชื่อมต่อทางฮาร์ดแวร์ของชุดจำลองทั้ง 2 ชุด.....	94
7-27 โปรแกรมแลดเดอร์ที่แทรกในเครื่องควบคุมชุดที่ 1.....	94
7-28 โปรแกรมแลดเดอร์ที่แทรกในเครื่องควบคุมชุดที่ 2.....	94
7-29 แสดงโปรแกรมการเชื่อมโยงข้อมูลโดยโฮสคอมพิวเตอร์.....	95
7-30 แสดงโปรแกรมการเชื่อมโยงข้อมูลโดยโฮสคอมพิวเตอร์.....	95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

เครื่องควบคุมแบบตรรกที่โปรแกรมได้ เป็นอุปกรณ์ที่ใช้ในการควบคุมเครื่องจักรอัตโนมัติในโรงงานอุตสาหกรรม ซึ่งปัจจุบันมีการนำมาใช้งานกับอย่างแพร่หลาย เป็นอุปกรณ์ที่พัฒนาและออกแบบโดยไมโครโปรเซสเซอร์ มาแทนระบบการควบคุมแบบเก่าซึ่งใช้อุปกรณ์ไฟฟ้าเชิงกล เช่น รีเลย์ (Relay) ตัวตั้งเวลา(Timer) ตัวนับ(Counter) เป็นต้น เนื่องจากเทคโนโลยีทางด้านไมโครโปรเซสเซอร์มีความเจริญก้าวหน้ามาก ดังนั้นการนำเทคโนโลยีดังกล่าวนี้มาพัฒนาเพื่อควบคุมเครื่องจักรในโรงงานอุตสาหกรรม จึงมีความน่าเชื่อถือสูง และมีข้อดีกว่าระบบเก่าหลายประการ

วิทยานิพนธ์นี้นำเสนอการออกแบบเครื่องควบคุมแบบตรรกที่โปรแกรมได้ โดยใช้ไมโครโปรเซสเซอร์ เบอร์ Z84C11 เป็นตัวประมวลผลกลางประกอบด้วย อินพุท 24 จุด เอาท์พุท 12จุด มีแป้นคีย์สำหรับป้อนโปรแกรมภายในตัว และฟังก์ชันคำสั่งพิเศษในการตรวจสอบสภาวะขณะที่เครื่องควบคุมทำงานอยู่ เป็นต้น นอกจากนี้ เครื่องควบคุมยังถูกออกแบบ ให้สามารถเชื่อมโยงกันได้เป็นระบบโครงข่ายที่มีการควบคุมสู่ศูนย์กลางผ่านโฮสคอมพิวเตอร์ และมีซอฟท์แวร์บนคอมพิวเตอร์ทำงานภายใต้ระบบจัดการของไมโครซอฟวินโดว์95 ซึ่งออกแบบมาเพื่อใช้สนับสนุนการทำงานของเครื่องควบคุม เช่น การติดตามผล การสั่งการควบคุมระยะไกล การอ่านเขียนข้อมูลเพื่อจัดเก็บลงบนแฟ้มข้อมูลของเครื่องคอมพิวเตอร์ และมีฟังก์ชันอื่นๆ ที่จำเป็น เพื่ออำนวยความสะดวกในการใช้งาน

ระบบโครงข่ายนี้ออกแบบการเชื่อมต่อสื่อสารตามมาตรฐาน RS 485 ระบบถูกออกแบบให้เชื่อมโยงกันได้ 8 ชุด และระยะทางในการติดต่อควบคุมได้ไกลนับร้อยเมตร จึงเป็นผลดีต่อการควบคุมหรือแก้ไขข้อผิดพลาดของเครื่องควบคุมเครื่องจักรที่ถูกจัดวางในสถานที่ที่ต่างกัน ซึ่งเป็นการเพิ่มประสิทธิภาพในการทำงานให้รวดเร็ว และครอบคุมทั้งระบบ

สำหรับภายวิทยานิพนธ์ในบทที่ 2 จะได้กล่าว ถึงทฤษฎี และ หลักการพื้นฐานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้ บทที่3 เป็นการติดต่อสื่อสารของเครื่องควบคุมกับอุปกรณ์ภายนอก บทที่4 เป็นการออกแบบและพัฒนาตัวเครื่องควบคุมแบบตรรกที่โปรแกรมได้ บทที่ 5 เป็นการออกแบบระบบโครงข่ายสำหรับเครื่องควบคุม บทที่6 เป็นการออกแบบโปรแกรมสนับสนุนการทำงานของเครื่องควบคุม บทที่ 7 เป็นการประยุกต์การใช้งานและผลการทดสอบ เครื่องควบคุม และระบบโครงข่ายร่วมกับ โปรแกรมสนับสนุนการทำงานส่วน บทที่ 8 จะเป็นการสรุปและวิจารณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการพื้นฐานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้

2.1 ที่มาและวิวัฒนาการของเครื่องควบคุมแบบตรรกที่โปรแกรมได้

การควบคุมแบบลำดับขั้น (Sequence Control) เป็นการควบคุมแบบหนึ่งที่ต้องการให้เครื่องจักรหรือระบบการทำงานตามช่วงเวลา ตามลำดับขั้นตอน ตลอดจนตามเงื่อนไขที่ได้กำหนดไว้โดยมีลักษณะของการควบคุมเป็นแบบ "ON" หรือ "OFF" การควบคุมแบบลำดับขั้นหรือแบบซีเควนซ์นี้จะพบเห็นอยู่เสมอในงานอุตสาหกรรมแทบทุกชนิด ตัวอย่างของการควบคุมแบบนี้ได้แก่ ระบบควบคุมเครื่องจักร ระบบล้างรถอัตโนมัติ ระบบป้อนวัสดุ ระบบการผสมวัสดุ ระบบควบคุมลิฟท์ เป็นต้น

ในอดีตที่ผ่านมา ระบบควบคุมแบบซีเควนซ์จะใช้อุปกรณ์ไฟฟ้า เชิงกล(Electromechanical-Device) เช่น รีเลย์แม่เหล็กไฟฟ้า (Relay) ตัวตั้งเวลา (Timer) ตัวนับ (Counter) มาประกอบกันเป็นวงจรควบคุมเพื่อให้เครื่องจักรหรือระบบกระบวนการทำงานตามช่วงเวลา ตามลำดับขั้นตอน และตามเงื่อนไขที่วิศวกรหรือผู้ออกแบบระบบกำหนดได้ อย่างไรก็ตามระบบควบคุมแบบซีเควนซ์ที่ใช้อุปกรณ์ไฟฟ้าเชิงกลที่กล่าวมานี้ มีข้อเสียมากคือขนาดของวงจรควบคุมจะมีขนาดใหญ่สิ้นเปลืองทั้งเนื้อที่และพลังงานสูง ราคาแพง ไม่สามารถจะใช้กับระบบควบคุมที่มีขั้นตอนการทำงานยุ่งยากซับซ้อน เมื่อมีปัญหาเกิดขึ้นในวงจรควบคุมก็ตรวจสอบแก้ไขยาก การขยายระบบทำได้ยาก และที่สำคัญคือถ้าต้องการเปลี่ยนแปลงลำดับขั้นตอนหรือเงื่อนไขของการทำงาน วงจรควบคุมแบบรีเลย์ที่มีอยู่เดิม จะเปลี่ยนแปลงเพื่อใช้กับงานใหม่ได้ยาก ต่อมาเมื่ออุปกรณ์สารกึ่งตัวนำ (Solid State Device) ได้แพร่หลาย และก้าวหน้าขึ้น จึงได้มีการนำเอาเกท (Gate) ต่างๆ มาประกอบกันเป็นวงจรควบคุมแทนรีเลย์ และสามารถลดข้อเสียของระบบเดิมลงไปได้มาก เช่น วงจรควบคุมแทนรีเลย์ และสามารถลดข้อเสียการขยายระบบทำได้ง่ายขึ้น และยังสามารถต่อเข้ากับคอมพิวเตอร์ เพื่อการเก็บข้อมูลและอื่นๆ ได้เป็นต้น แต่ก็ยังไม่สามารถจะลดข้อเสียบางอย่างที่สำคัญลงได้ นั่นคือการตรวจสอบแก้ไข เมื่อมีปัญหาก็ยังทำได้ยาก และไม่สามารถจะเปลี่ยนแปลงลำดับขั้นตอน หรือเงื่อนไขของการทำงาน จึงได้มีการสร้างเครื่องควบคุมที่สามารถกำหนดโปรแกรมการทำงาน (Programmable Controller) ขึ้นมาเพื่อให้นำมาประยุกต์ใช้กับระบบควบคุมแบบซีเควนซ์ได้ทุกระบบโดยไม่ต้องเปลี่ยนแปลงแก้ไขวงจรควบคุมแต่อย่างใด

2.1.1 วิวัฒนาการของเครื่องควบคุมแบบตรรกที่โปรแกรมได้ PLC หรือ Programmable-Logic Controllers และ PC หรือ Programmable Controller เป็นเครื่องควบคุมที่สามารถกำหนดโปรแกรมการทำงานได้ PLC/PC ที่ใช้ไมโครโปรเซสเซอร์นอกจากจะใช้ในการควบคุมเครื่องจักร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปแล้ว ยังมีการพัฒนาให้มีความสามารถและขอบเขตของงานได้กว้างขวางขึ้น เช่น มีฟังก์ชันทางคณิตศาสตร์เพิ่มขึ้น ทำให้การควบคุมเป็นได้ทั้งแบบ ON-OFF หรือแบบอนาล็อก เช่น PID (Proportional +Integral +Derivative) สามารถต่อเข้ากับอุปกรณ์วัดและควบคุม (Instrumentation) อื่นๆ จำนวนของอินพุท / เอาท์พุท(Input/Output) ขยายให้มากขึ้นได้ หรือการใช้อินพุท/เอาท์พุทแบบรีโมท(Remote) เพื่อลดการเดินสาย การติดต่อสื่อสารข้อมูลกับคอมพิวเตอร์ตลอดจนอุปกรณ์พิเศษอื่น ๆ เช่น เครื่องอ่านรหัสแถบเพื่อจำแนกหรือ จัดทำฐานข้อมูล การเชื่อมโยงข้อมูลลักษณะโครงข่ายท้องถิ่น (LAN) เป็นต้น หน่วยความจำก็ขยายได้มากทำให้ใช้กับระบบกระบวนการใหญ่ๆได้ และมีการจัดการเกี่ยวกับข้อมูลได้ตามต้องการ

2.1.2 สรุปข้อดีของ PLC/PC

ข้อดีของ PLC/PC สามารถจำแนกออกได้เป็นข้อๆดังนี้

1. สิ้นเปลืองเนื้อที่น้อยเพราะมีขนาดเล็ก
2. สามารถจะใช้ควบคุมเครื่องจักรหรือระบบกระบวนการใดๆ ก็ได้ ถ้าเลือกขนาดของ PLC/PC ที่เหมาะสม
3. การเปลี่ยนลำดับขั้นตอนหรือเงื่อนไขของการทำงานก็ทำได้ตามต้องการ เพราะใช้หลักของการโปรแกรม
4. ตัวตั้งเวลาและตัวนับจะเป็นซอฟท์แวร์ ทำให้การกำหนดค่าต่างๆง่าย เปลี่ยนแปลงค่าได้ตลอดเวลาไม่ต้องมีฮาร์ดแวร์ร่วม และทำให้ราคาถูกลง
6. รีเลย์ภายใน(Internal relay) ก็เป็นซอฟท์แวร์เช่นเดียวกัน จึงลดค่าใช้จ่ายในการเดินสาย ลดฮาร์ดแวร์ และทำให้ขนาดเล็กลงด้วย
7. การติดตั้งทำ ได้ง่ายและสะดวก
8. การขยายระบบให้ใหญ่ขึ้นทำได้โดยง่าย
9. ราคาถูกกว่าระบบรีเลย์
10. ความน่าเชื่อถือ (Reliability) ดีเพราะเป็นอุปกรณ์สารกึ่งตัวนำ ไม่มีการเดินสายมาก ไม่มีปัญหาเกี่ยวกับหน้าสัมผัส (Contact) แบบรีเลย์
11. มีระบบตรวจสอบหาที่ผิดพลาดด้วยตัวเอง การตรวจสอบแก้ไขเมื่อมีปัญหาจึงทำได้เร็ว
12. ลดการเดินสายยาวๆและลดค่าใช้จ่ายในการเดินสายได้ เพราะมีอินพุท เอาท์พุทแบบรีโมท
13. การบำรุงรักษาทำได้ง่าย
14. เวลาในการทำงานเร็วกว่าระบบที่ใช้รีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. มีฟังก์ชันทางคณิตศาสตร์ได้แก่ บวก ลบ คูณ หาร และอื่นๆ ทำให้สามารถใช้สำหรับการควบคุมแบบ ON-OFF หรือแบบอนาล็อก เช่น PIDได้

16. สามารถเชื่อมต่อ กับอุปกรณ์การวัด เช่น เทอร์โมคัปเปิล (Thermocouple) และอื่นๆได้นอกเหนือจากอุปกรณ์ตรวจวัดที่เป็นสวิทซ์

17. ต่อเข้ากับคอมพิวเตอร์เพื่อวัตถุประสงค์ใดๆ เช่น การเก็บข้อมูลการกระจายการควบคุม(Distributed Control) เป็นต้น

18. การโปรแกรมทำได้หลายแบบ เช่น คำสั่งในรูปของแลดเดอร์ไดอะแกรม คำสั่งบูลีน คำสั่งในรูปบล็อก หรือคำสั่งภาษาเบสิค

19. ใช้ได้ในทุกสภาพแวดล้อมของงานอุตสาหกรรม

20. การโปรแกรมทำได้โดยใช้เครื่องป้อนโปรแกรม (Program Loader)หรือโปรแกรมลงบน CRT

จากที่กล่าวมานี้จะเห็นว่า PLC/PC เป็นเครื่องควบคุมที่มีประสิทธิภาพและมีประโยชน์อย่างยิ่งต่องานอุตสาหกรรมในปัจจุบันและในอนาคต

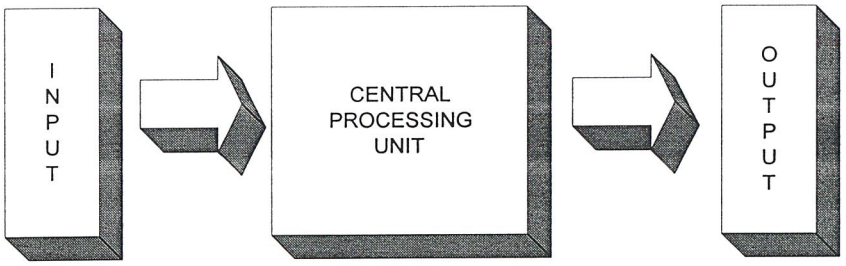
2.2 โครงสร้างและหลักการทำงานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้

เครื่องควบคุมแบบตรรกที่โปรแกรมได้หรือ PLC/PC เป็นอุปกรณ์สารกึ่งตัวนำที่ใช้สำหรับควบคุมเครื่องจักร หรือระบบกระบวนการให้ทำงานตามโปรแกรมคำสั่งของผู้ใช้ (User Program) และข้อมูลต่างๆ ที่ได้รับจากอินพุท/เอาท์พุทของ PLC/PC จะเป็นได้ทั้งการทำงานตามช่วงเวลาตามลำดับขั้นตอนฟังก์ชันทางคณิตศาสตร์ และอื่นๆ PLC/PC มีส่วนประกอบสำคัญ 2 ส่วนคือ

1. หน่วยประมวลผลกลางหรือ CPU (Central Processing Unit)
2. หน่วยอินพุท/เอาท์พุท (Input/Output Unit)

นอกจากส่วนประกอบทั้งสองแล้ว PLC/PC ยังประกอบด้วยหน่วยป้อนโปรแกรม ผู้ใช้สามารถติดต่อกับ PLC/PC หรือเปลี่ยนแปลงแก้ไขโปรแกรมคำสั่งตรวจสอบสภาพการทำงานของเครื่องจักร หรือระบบกระบวนการ ตลอดจนตรวจสอบสภาพการทำงานของ PLC/PC ได้ทางหน่วยป้อนโปรแกรม เพื่อป้อนหรือเปลี่ยนแปลงแก้ไขโปรแกรม คำสั่งตรวจสอบสภาพการทำงานของเครื่องจักร หรือระบบกระบวนการ ตลอดจนตรวจสอบสภาพการทำงานของ PLC/PC รูปที่ 2-1 แสดงส่วนประกอบที่สำคัญทั้ง 2 ส่วนของ PLC/PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-1 บล็อกไดอะแกรมของ PLC/PC

2.2.1 หน่วยประมวลผลกลาง

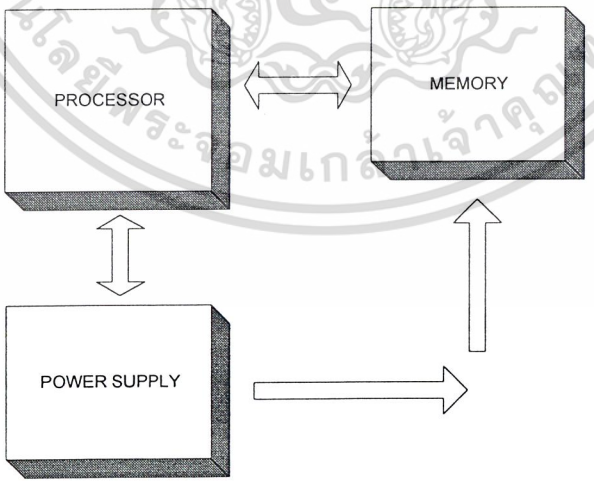
หน่วยประมวลผลกลางหรือ CPU ประกอบด้วย 3 ส่วนคือ

2.2.1.1 หน่วยประมวลผล (Processor)

2.2.1.2 หน่วยความจำ (Memory)

2.2.1.3 หน่วยจ่ายกำลัง (Power Supply)

รูปที่ 2-2 แสดงส่วนประกอบของ CPU โดยทั่วไป CPUจะใช้ไมโครโปรเซสเซอร์ซึ่งเป็นวงจรรวม (I.C. : Integrated Circuit) ที่มีความสามารถทั้งในการคำนวณทางคณิตศาสตร์ การจัดการข้อมูล และการตรวจสอบตัวเอง ในขณะที่วงจรควบคุมที่ใช้รีเลย์หรือไอซีพวกเกท ต่างๆ ไม่สามารถจะทำได้ PLC/PC จึงมีข้อดีและประโยชน์มากกว่าระบบแบบเก่าอย่างเห็นได้ชัด



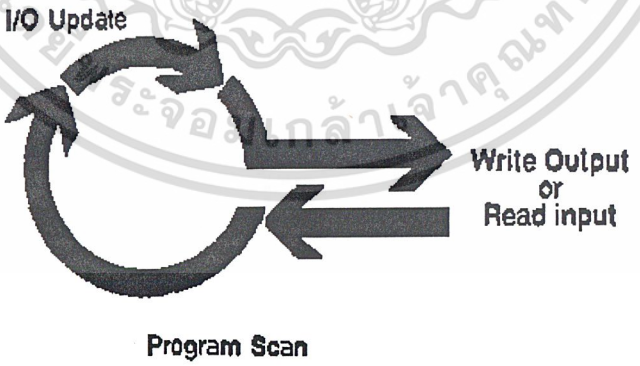
รูปที่ 2-2 บล็อกไดอะแกรมของ CPU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.1 หน่วยประมวลผล

หน่วยประมวลผลของ CPU ทำหน้าที่ควบคุมและดูแลการทำงานของระบบทั้งหมด โดยรับข้อมูลจากหน่วยอินพุทมาทำการประมวลผลตามโปรแกรมคำสั่งของผู้ใช้ที่เก็บไว้ในหน่วยความจำ และส่งผลที่ได้ออกไปยังหน่วยเอาท์พุท ในปัจจุบันหน่วยประมวลผลของCPU จะใช้ไมโครโปรเซสเซอร์ขนาดตั้งแต่ 4 บิท 8 บิท หรือ 16 บิท จำนวน 1 ตัว หรือหลายๆ ตัวมาทำงานร่วมกัน ในกรณีที่ใช้ไมโครโปรเซสเซอร์หลายๆ ตัว จะมีข้อดีคือ การทำงานของระบบจะเร็วขึ้น ทำให้ PLC/PC มีขีดความสามารถสูงขึ้น เช่น การใช้ชุดเชื่อมต่อพิเศษที่มีไมโครโปรเซสเซอร์และหน่วยความจำภายใน จะทำให้การทำงานควบคุมดียิ่งขึ้นและเป็นอิสระไม่ขึ้นกับ CPU ตัวอย่างเช่น ชุดควบคุม PID ที่ใช้สำหรับควบคุมระบบแบบลูฟปิด (Closed-Loop Control) ซึ่งทำงานอิสระจาก CPU

การทำงานของ CPU 1 รอบ เริ่มจากการรับข้อมูลจากการรับข้อมูลจากหน่วยอินพุทเข้ามาประมวลผลตามโปรแกรมคำสั่งของผู้ใช้ ส่งผลที่ได้ไปยังหน่วยเอาท์พุท ตรวจสอบการทำงานของระบบทั้งหมดและติดต่อกับผู้ใช้ เรียกว่าการ SCAN โดยทั่วไปเวลา SCAN ของ CPU จะใช้เวลาประมาณ 0.001 ถึง 0.1 วินาที เวลา SCAN จะอ้างอิงถึงขนาดของหน่วยความจำที่เก็บโปรแกรมของผู้ใช้เสมอ เช่น เวลา SCAN เท่ากับ 0.01 วินาทีต่อโปรแกรมคำสั่งขนาด 1 กิโลไบท์ (K-byte) อย่างไรก็ตามเวลา SCAN จะขึ้นอยู่กับองค์ประกอบอื่นๆด้วย เช่นการใช้ชุดอินพุท/เอาท์พุทแบบรีโมทจะทำให้เวลา SCAN ของ CPU เพิ่มขึ้น



รูปที่ 2-3 การ SCAN ของ CPU

ช่วงเวลา SCAN ของ CPU จะทำให้ทราบถึงความสามารถในการตอบสนองต่อการเปลี่ยนแปลงของอินพุท/เอาท์พุท ของPLC/PC ว่ารวดเร็วเพียงใด เวลาSCANจึงเป็นองค์ประกอบที่สำคัญอย่างหนึ่งในการเลือกใช้ PLC/PC ให้เหมาะสมกับงาน เช่น ถ้าต้องการที่จะตรวจสอบสัญญาณอินพุทที่เปลี่ยนค่าสภาวะถึง 2 ครั้งภายใน 0.008 วินาที การใช้PLC/PC ที่มีช่วงเวลา SCAN เท่ากับ

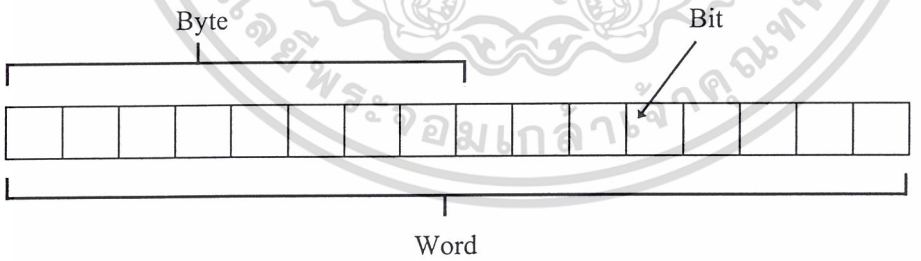
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCAN เท่ากับ 0.01 วินาที จะไม่สามารถตรวจสอบค่าสภาวะที่ถูกต้อง และอาจทำให้เครื่องจักรหรือกระบวนการที่ควบคุมทำงานผิดพลาดได้ นอกจากนี้ หน่วยประมวลผลใน CPU ยังทำหน้าที่ในการติดต่อกับระบบย่อย (Sub system) ตรวจสอบความผิดพลาดต่างๆ เช่น การทำงานของหน่วยประมวลผล หน่วยความจำ หรือสภาพของแบตเตอรีจ่ายกำลังสำรอง เป็นต้น

2.2.1.2 หน่วยความจำ

หน่วยความจำ เป็นที่เก็บโปรแกรมและข้อมูลที่หน่วยประมวลผลใช้ในการควบคุมการทำงานของ PLC/PC และการทำงานตามโปรแกรมคำสั่งของผู้ใช้ ขนาดของหน่วยความจำจะแบ่งออกเป็นบิทข้อมูลภายในหน่วยความจำ 1 บิท จะมีค่าสภาวะทาง ลอจิก “1” หรือ “0” ข้อมูลขนาด 8 บิท และ 16 บิท รวมกันเรียกว่าไบท์ (Byte) และ เวิร์ด(Word) ตามลำดับ

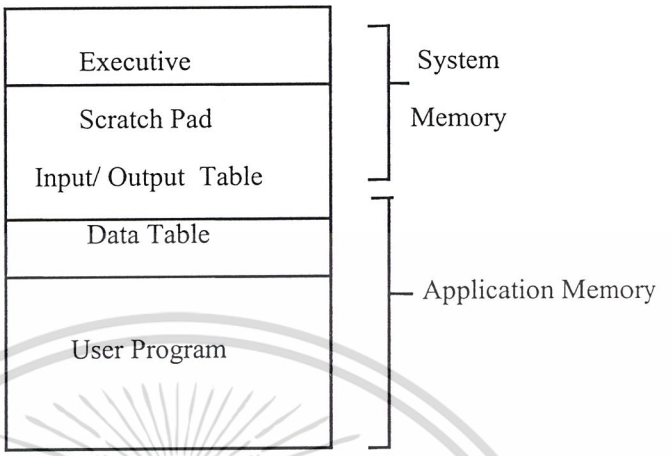
หน่วยความจำแบ่งออกได้เป็น 2 ประเภทใหญ่ๆ คือหน่วยความจำที่ข้อมูลสูญหายเมื่อไม่มีแหล่งจ่ายกำลัง(Volatile) และหน่วยความจำที่ข้อมูลคงอยู่แม้ไม่มีแหล่งจ่ายกำลัง (Non- Volatile) หรืออาจจะแบ่งเป็น 2 ชนิดใหญ่ๆ ตามคุณลักษณะคือ RAM (Random Access Memory) หรือ NOVRAM (Non-Volatile RAM) และ ROM(Read Only Memory) นอกจากนี้ ROM ยังจำแนกออกได้เป็น PROM(Programmable ROM) EPROM(Erasable Programmable ROM) EAROM (Electrically Alterable ROM) และ EEPROM (Electrically Erasable Programmable ROM)



รูปที่ 2-4 ข้อมูลเป็น บิท ไบท์ และเวิร์ด

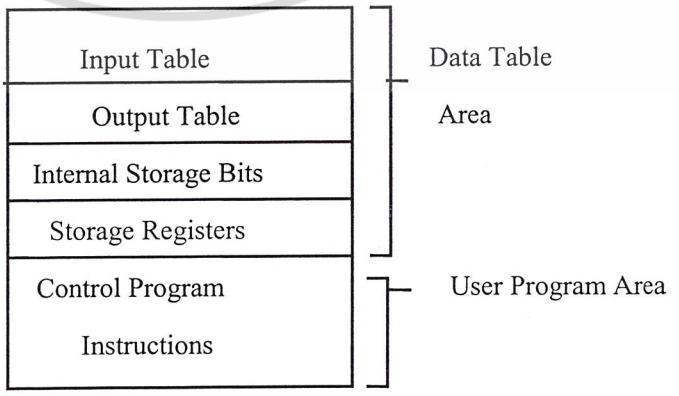
หน่วยความจำของ PLC/PC แบ่งออกเป็น 4 ส่วน ตามรูปที่ 2-5 เพื่อใช้เก็บข้อมูลและโปรแกรมต่างๆ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-5 หน่วยความจำของ PLC/PC

1. โปรแกรมจัดการ (Execute program) เป็นโปรแกรมที่ทำหน้าควบคุม ดูแล และตรวจสอบการทำงานของ PLC/PC ทั้งหมด
2. ข้อมูลชั่วคราว (Processor Work Area หรือ Scratch pad) หน่วยความจำส่วนที่สองทำหน้าที่เก็บข้อมูลที่เกิดขึ้นระหว่าง PLC/PCทำงานตามโปรแกรมจัดการและโปรแกรมคำสั่งของผู้ใช้
3. ตารางข้อมูล (Data Table) หน่วยความจำส่วนที่สามทำหน้าที่เก็บ ค่าของอินพุท เอาท์พุท และตัวแปรต่างๆ จากการทำงานตามโปรแกรมคำสั่ง ของผู้ใช้ ทั้งที่เป็นค่าสภาวะทางลอจิกและตัวเลข
4. หน่วยความจำสำหรับเก็บโปรแกรมคำสั่งของผู้ใช้ (User Program Memory)



รูปที่ 2-6 หน่วยความจำสำหรับผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำของผู้ใช้จะแตกต่างกันตามขนาดของ PLC/PC ใน PLC หรือ PC ขนาดใหญ่ ขนาดของหน่วยความจำดังกล่าวจะสามารถเปลี่ยนแปลงขนาดได้ตามต้องการ การเลือกใช้ PLC/PC จึงต้องคำนึงถึงขีดความสามารถและขีดจำกัดต่างๆ ประกอบด้วย เช่น ขนาดโปรแกรมคำสั่งของผู้ใช้ จำนวนอินพุท/เอาท์พุทที่จะขยายได้สูงสุด ขนาดของหน่วยความจำภายในที่ใช้เก็บข้อมูล และฟังก์ชันพิเศษต่างๆ เช่น รีเลย์ภายใน ตัวตั้งเวลา และตัวนับ

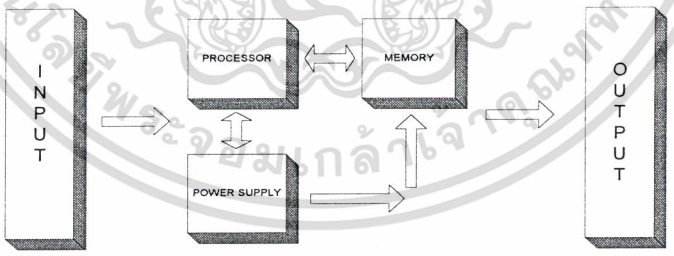
2.2.1.3 หน่วยจ่ายกำลัง

หน่วยจ่ายกำลังจะทำหน้าที่จ่ายและรักษาระดับแรงดันไฟฟ้ากระแสตรง (D.C Voltage) ให้กับหน่วยประมวลผล หน่วยความจำ และหน่วยอินพุท/เอาท์พุท ตามความต้องการ และทำหน้าที่เตือนให้หน่วยประมวลผลทราบเมื่อเกิดปัญหา

2.2.2 หน่วยอินพุท/เอาท์พุท

หน่วยอินพุททำหน้าที่เชื่อมต่อระหว่าง CPU กับอุปกรณ์ภายนอก โดยรับค่าสภาวะหรือปริมาณทางกายภาพต่างๆ จากอุปกรณ์ตรวจวัด (Sensor) ของเครื่องจักรหรือกระบวนการ เช่น ลิมิตสวิทซ์ (Limit Switch) พร็อกซิมิตี้สวิทซ์ (Proximity Switch) ตำแหน่งอุณหภูมิ ระดับแรงดันกระแสไฟ และอื่นๆ ส่งไปยัง CPU เพื่อประมวลผลตามโปรแกรมคำสั่งของผู้ใช้

หน่วยเอาท์พุททำหน้าที่รับค่าสภาวะหรือคำสั่งควบคุมที่ได้จาก CPU เพื่อส่งไปควบคุมอุปกรณ์ภายในเครื่องจักรหรือกระบวนการเช่น วาล์ว(Valve) มอเตอร์(Motor) ปั๊ม(Pump) และอื่นๆให้ทำงานตามค่าสภาวะหรือคำสั่งที่ CPU ต้องการ



รูปที่ 2-7 บล็อกไดอะแกรมของหน่วยอินพุท/เอาท์พุท

ดังที่ได้กล่าวมาแล้วว่าในช่วงแรก PLC/PC ถูกใช้แทนรีเลย์สำหรับการควบคุมแบบ ON-OFF หน่วยอินพุท/เอาท์พุทในขณะนั้นจึงเป็นแบบที่ใช้สำหรับค่าสภาวะลอจิกเท่านั้น แต่ต่อมาเมื่อPLC/PC ได้ถูกพัฒนาจนมีการควบคุมแบบอนาล็อกรวมอยู่ด้วย โดยใช้หน่วยอินพุท/เอาท์พุทที่ถูกพัฒนาขึ้นเพื่อ รับ-ส่งสัญญาณแบบอนาล็อกโดยเฉพาะ ปัจจุบันหน่วยอินพุทและเอาท์พุท ของ PLC/PC อาจแบ่งได้ดังนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.1 อินพุท/เอาท์พุทแบบค่าสภาวะลอจิก

อินพุท/เอาท์พุทแบบนี้จะใช้กันมากที่สุด ข้อมูลจะมีค่าสภาวะอยู่ 2 สภาวะคือ “1” หรือ “ ON ” หมายถึง สวิทซ์ หรือหน้าสัมผัสของรีเลย์แตะกัน และสภาวะ “0” หรือ “OFF” หมายถึง สวิทซ์หรือหน้าสัมผัสของรีเลย์เปิดออก

เนื่องจากระดับแรงดันไฟฟ้าทางด้านอินพุทมีหลายแบบ การเลือกใช้หน่วยอินพุท/เอาท์พุทแบบค่าสภาวะลอจิกจึงต้องเลือกใช้ให้ถูกต้อง ตารางที่ 2-1 แสดงให้เห็นตัวอย่างของอุปกรณ์อินพุท/เอาท์พุทแบบลอจิก

อุปกรณ์ทางด้านอินพุท	อุปกรณ์ทางด้านเอาท์พุท
Selector Switches	Alarms
Push Buttons	Control Relays
Photoelectric Eyes	Fans
Limit Switches	Lights
Circuit Breakers	Horns
Proximity Switches	Valves
Level Switches	Motor Starters
Motor Starter Contacts	Solenoid
Relay Contacts	

ตารางที่ 2-1 อุปกรณ์ อินพุท/เอาท์พุทแบบค่าสภาวะลอจิก

2.2.2.2 อินพุท/เอาท์พุทแบบค่าตัวเลข

หน่วยอินพุท/เอาท์พุทแบบค่าตัวเลขนี้จะมีอยู่ใน PLC/PC ตั้งแต่ขนาดกลางที่มีฟังก์ชั่นการคำนวณทางคณิตศาสตร์รวมอยู่ด้วย โดยทั่วไปแล้วหน่วยอินพุท/เอาท์พุทแบบค่าตัวเลขนี้จะแบ่งเป็น 2 ชนิดคือหน่วยอินพุท/เอาท์พุทสำหรับสัญญาณอนาล็อก และหน่วยอินพุท/เอาท์พุทสำหรับค่าสภาวะแบบหลายบิท

หน่วยอินพุทสำหรับสัญญาณอนาล็อกจะรับสัญญาณอินพุทจากอุปกรณ์ต่างๆ และหน่วยเอาท์พุทจะส่งสัญญาณอนาล็อกไปยังอุปกรณ์ต่างๆ ตาราง 2-2 แสดงถึงอุปกรณ์อินพุท/เอาท์พุทบางแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์อินพุทแบบอนาล็อก	อุปกรณ์เอาท์พุทแบบอนาล็อก
Temperature Transducers	Analog Valves and Actuators
Pressure Transducers	Chart Recorders
Load Cell Transducers	Electric Motor Drives
Humidity Transducers	Analog Meters
Flow Transducers	

ตารางที่ 2-2 อุปกรณ์อินพุท/เอาท์พุทแบบค่าตัวเลข

2.2.2.3 อินพุท/เอาท์พุทสำหรับค่าสภาวะแบบหลายบิท

หน่วยอินพุท/เอาท์พุทสำหรับค่าสภาวะแบบหลายบิท นี้จะทำงานเหมือนกับหน่วยอินพุท เอาท์พุท แบบค่าสภาวะลอจิกทุกอย่าง ต่างกันก็แต่เพียงการรับ-ส่งข้อมูล ซึ่งสามารถทำได้ครั้งละหลายๆ บิทในเวลาเดียวกัน เช่น อินพุท/เอาท์พุทแบบ BCD เป็นต้น ตารางที่2-3 แสดงถึงอุปกรณ์ที่ใช้ร่วมกับหน่วยอินพุท/เอาท์พุทสำหรับค่าสภาวะแบบหลายบิท

อุปกรณ์อินพุทแบบหลายบิท	อุปกรณ์เอาท์พุทแบบหลายบิท
Thumbwheel Switches	Seven - Segment Displays
Bar Codes Readers	Intelligent Display

ตารางที่ 2-3 อุปกรณ์อินพุท/เอาท์พุทสำหรับค่าสภาวะแบบหลายบิท

2.2.2.4 อินพุท/เอาท์พุทแบบพิเศษ

สำหรับ PLC/PCตั้งแต่ขนาดกลางขึ้นไปจะสามารถนำเอาหน่วยอินพุทเอาท์พุท แบบพิเศษมาต่อเพื่อใช้งานเฉพาะด้านได้ อินพุท/เอาท์พุทสำหรับงานเฉพาะด้านนั้นมีหลายอย่างดังต่อไปนี้

1. อินพุทที่เป็นเทอร์โมคัปเปิล (Thermocouple Input) หน่วยอินพุทสำหรับเทอร์โมคัปเปิลนี้จะทำหน้าที่ปรับระดับสัญญาณ (Signal Conditioning) ที่รับมาโดยตรงจากเทอร์โมคัปเปิลให้มีระดับที่เหมาะสม การทำงานของหน่วย อินพุทชนิดนี้จะเหมือนกับหน่วยอินพุทสัญญาณอนาล็อกจาก เทอร์โมคัปเปิลนี้จะถูกกรอง ขยาย และทำให้เป็นสัญญาณดิจิตอล จากนั้นถูกส่งไปยังหน่วยประมวลผลเพื่อการทำงานตามโปรแกรมคำสั่งของผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. อินพุทที่เป็นพัลซ์ หน่วยอินพุทสำหรับรับอินพุทที่เป็นพัลซ์นี้จะทำหน้าที่ตรวจจับสัญญาณที่มีการเปลี่ยนแปลงเร็วกว่า 1 ช่วงเวลาของการ SCAN ของCPU

3. อินพุท/เอาท์พุทที่เป็นรหัส ASCII หน่วยอินพุท/เอาท์พุทแบบนี้ใช้สำหรับรับ-ส่งข้อมูลที่เป็นตัวอักษรในรหัส ASCII

4. อินพุทที่เป็นสเตนเกจ (Strain Gage Input) หน่วยอินพุทสำหรับสัญญาณอนาล็อกจากสเตนเกจเหมือนกับหน่วยอินพุทสำหรับเทอร์โมคับเปิล แตกต่างกันที่วงจรปรับระดับสัญญาณ ซึ่งถูกออกแบบไว้รับสัญญาณจากสเตนเกจเท่านั้น

5. เอาท์พุทที่เป็นพัลซ์(Pulse Output) หน่วยเอาท์พุทนี้ทำหน้าที่ส่งพัลซ์ไปควบคุมการทำงานของสเตปปิ้งมอเตอร์ โดยเฉพาะ ตำแหน่งของมอเตอร์จะถูกกำหนดจากจำนวนของพัลซ์เอาท์พุทที่ถูกนับไว้ก่อน และคำสั่งให้หมุนไปในทิศทางทวนเข็มนาฬิกาหรือตามเข็มนาฬิกา

6. หน่วยเชื่อมต่อเซอร์โว (Servo Interface) หน่วยเชื่อมต่อเซอร์โว เป็นหน่วยอินพุทเอาท์พุทที่ถูกออกแบบขึ้นเพื่อตรวจสอบ และควบคุมตำแหน่งของเครื่องจักรโดยเฉพาะ เช่นการควบคุมในระบบ NC (Numerical Control)

7. หน่วย APM (Axis Positioning Module) หน่วย APM เป็นหน่วยเชื่อมต่อของ PC ที่ได้รับการออกแบบขึ้นเพื่อใช้สำหรับตรวจสอบ และควบคุมตำแหน่งของเครื่องจักร เช่น เดียวกับหน่วยเชื่อมต่อเซอร์โว แต่สามารถจะใช้ร่วมกับการควบคุมแบบซีเควนซ์ได้โดยที่การทำงานที่แยกเป็นอิสระจาก CPU

8. หน่วย PID เป็นหน่วยอินพุท/เอาท์พุทที่ได้รับการออกแบบเพื่อใช้ควบคุมในระบบลูฟปิดที่ต้องการฟังก์ชัน การควบคุมแบบ PID (Proportional ,Integral ,Derivative)

9. หน่วยเชื่อมต่อโครงข่าย (Network Interface Module) หน่วยเชื่อมต่อนี้ถูกออกแบบขึ้นเพื่อเพิ่มสมรรถนะการทำงานของ PC ให้สูงขึ้น โดยช่วยให้ PC หลายๆ ระบบเชื่อมต่อกันและติดต่อกันได้ หรือติดต่อกับระบบคอมพิวเตอร์เพื่อใช้เป็นหน่วยควบคุมหลัก

10. หน่วยอินพุท/ เอาท์พุทแบบรีโมท หน่วยอินพุท/ เอาท์พุทแบบรีโมท จะใช้กับ PC ขนาดใหญ่ที่มี อินพุท / เอาท์พุท จำนวนมากๆ และมีระบบย่อยๆ หลายระบบ เพื่อทำหน้าที่ตรวจสอบสภาวะของอุปกรณ์จากจุดต่างๆ หลายจุดในเวลาเดียวกัน ภายใต้การควบคุมของ CPU

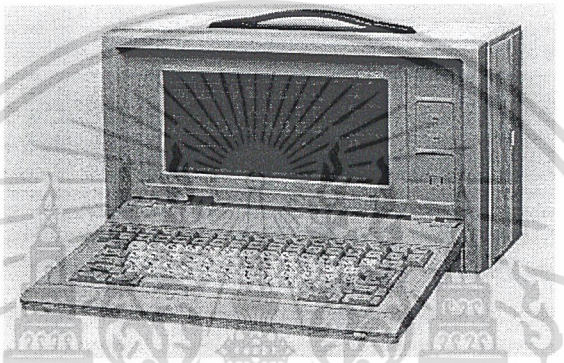
2.2.3 หน่วยป้อนโปรแกรม

หน่วยป้อนโปรแกรมของ PLC/PC ทำหน้าที่ติดต่อระหว่างผู้ใช้และ PLC/PC ทั้งระบบคือการป้อนโปรแกรมคำสั่งของผู้ใช้เข้าสู่หน่วยความจำของ CPU ตรวจสอบสภาวะการทำงาน รับ-ส่งข้อมูลระหว่าง CPU กับหน่วยป้อนโปรแกรม รับ- ส่งข้อมูลระหว่าง CPU กับเทป หรืออื่นๆ ตามฟังก์ชั่นการใช้งานที่มีอยู่ หน่วยป้อนโปรแกรมของ PLC/PC แบ่งออกได้หลายชนิดคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.1 เครื่องป้อนโปรแกรมแบบ CRT

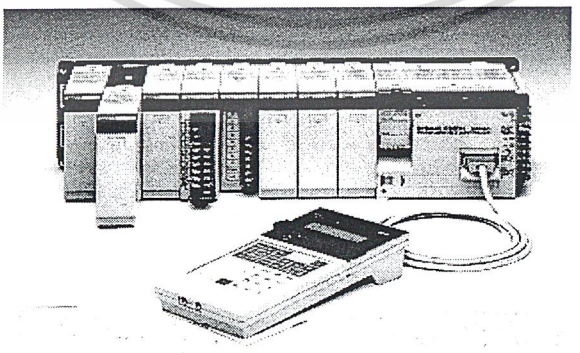
หน่วยป้อนโปรแกรม CRT ประกอบด้วยจอภาพและแป้นพิมพ์ เป็นเครื่องป้อนโปรแกรมแบบที่สะดวกที่สุดเครื่องป้อนโปรแกรมแบบนี้แบ่งออกเป็น 2 นิดคือแบบดัมบ์CRT(Dumb CRT) ซึ่งเป็นแบบที่ไม่มีหน่วยประมวลอยู่ภายในตัว การทำงานทุกอย่างจะถูกควบคุมจาก CPU ของ PLC/PC และแบบอินเทลลิเจนต์ CRT (Intelligent CRT) ซึ่งจะมีหน่วยประมวลอยู่ภายในตัว การทำงานเป็นอิสระจาก CPU จึงมีประสิทธิภาพดีกว่าแบบ ดัมบ์



รูปที่ 2-8 เครื่องป้อนโปรแกรมแบบ CRT

2.2.3.2 เครื่องป้อนโปรแกรมขนาดเล็ก (Mini Programmer)

เป็นเครื่องป้อนโปรแกรมขนาดเล็กที่สามารถพกติดตัวและเคลื่อนย้ายไปมาได้สะดวก แต่ประสิทธิภาพจะด้อยกว่าเครื่องป้อนโปรแกรมแบบ CRT ปกติแล้วจะใช้จอ LCD ในการแสดงผล เครื่องป้อนโปรแกรมแบบนี้จะแบ่งออกเป็น 2 ชนิดคือ แบบอินเทลลิเจนต์ และ แบบดัมบ์



รูปที่ 2-9 เครื่องป้อนโปรแกรมขนาดเล็กหรือพกพา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.3 เครื่องป้อนโปรแกรมลงในหน่วยหน่วยความจำของ PLC/PC หรือ เทป

เครื่องป้อนโปรแกรมแบบนี้สามารถใช้ป้อนหรือแก้ไขโปรแกรมคำสั่งของผู้ใช้เข้าไปในหน่วยความจำของ PLC/PCได้โดยตรงหรือใช้ในการรับ-ส่งโปรแกรมและส่งข้อมูลต่าง ๆ ระหว่าง PLC/PC และเทป

2.2.3.4. เครื่องป้อนโปรแกรมลงบนหน่วยความจำ (Memory Burner)

เครื่องป้อนโปรแกรมแบบนี้ทำหน้าที่ถ่ายโปรแกรมที่มีอยู่ลงเก็บไว้ในหน่วยความจำแบบ ROM เพื่อให้โปรแกรมต่างๆ คงอยู่ตลอดเวลา เมื่อโปรแกรมต่างๆ ได้ถูกตรวจสอบและแก้ไขจนเรียบร้อยสมบูรณ์แล้ว

2.2.3.5. คอมพิวเตอร์

เครื่องควบคุม PLC/PC บางรุ่นได้ออกแบบให้ สามารถติดต่อกับคอมพิวเตอร์ได้เพื่อให้การโปรแกรมทุกอย่างสามารถทำบนคอมพิวเตอร์ได้

2.3 ลักษณะการโปรแกรมให้กับเครื่องควบคุมแบบตรรกที่โปรแกรมได้

ภาษาที่ใช้ในการเขียนโปรแกรมคำสั่งสำหรับ PLC/PC นั้นสามารถจะจำแนกออกได้เป็น 4 ภาษาคือ

1. ภาษาแลดเดอร์ไดอะแกรม
2. ภาษาคำสั่งบูลีน
3. ภาษาคำสั่งในรูปบล็อก
4. ภาษาระดับสูง

ภาษาแลดเดอร์ไดอะแกรมหรือคำสั่งบูลีนเป็นภาษาที่นิยมใช้และเข้าใจ ได้ง่ายสำหรับการเขียนโปรแกรมคำสั่งของ PLC/PC ในการควบคุมแบบซีเควนซ์ที่เป็น ON-OFF การเขียนโปรแกรมคำสั่งในภาษาแลดเดอร์ ไดอะแกรมหรือคำสั่งบูลีนจึงใช้กันมากใน PLC หรือ PC ทุกขนาด ส่วนภาษาคำสั่งในรูปแบบบล็อก หรือ ภาษาระดับสูง เช่นภาษาซี ภาษาปาสคาล ภาษาฟอร์แทรน นั้นจะเหมาะสำหรับ PLC/PC ในการควบคุมแบบอนาล็อกการจัดการเกี่ยวกับข้อมูล การทำรายงานและอื่นๆ ซึ่งมักจะเป็น PLC ขนาดใหญ่ PLC แต่ละแบบอาจใช้คำสั่งเพียงภาษาเดียวหรือหลายภาษารวมกันในการเขียนโปรแกรมคำสั่งก็ได้ เช่น ภาษาแลดเดอร์ไดอะแกรมร่วมกับคำสั่งบูลีนเพียงภาษาเดียว หรือใช้ภาษาแลดเดอร์ไดอะแกรมหรือคำสั่งในรูปบล็อก ภาษาคำสั่งบูลีนร่วมกับคำสั่งในรูปบล็อก หรือภาษาคำสั่งบูลีนร่วมกับภาษาระดับสูง เป็นต้น อย่างก็ตามในหัวข้อนี้จะกล่าวถึงการเขียนโปรแกรมคำสั่งของ PLC/PC โดยใช้ภาษาแลดเดอร์ไดอะแกรม และคำสั่งบูลีนเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 โปรแกรมภาษาแลดเดอร์ไดอะแกรม

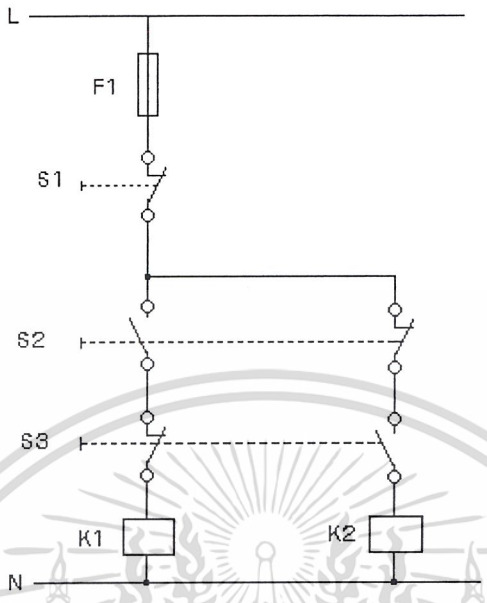
สัญลักษณ์พื้นฐานของภาษาแลดเดอร์ไดอะแกรม ดังที่ทราบกันแล้วว่า PLC/PC เป็นเครื่องควบคุมที่ใช้วิธีการโปรแกรม และใช้ซอฟท์แวร์แทนอุปกรณ์ต่างๆ เช่น รีเลย์ ตัวตั้งเวลา ตัวนับ ดังนั้นการเขียนโปรแกรมคำสั่งจึงใช้สัญลักษณ์พื้นฐานต่อไปนี้แทนอุปกรณ์ต่างๆ

สัญลักษณ์	ความหมาย
+--] [--+	จะใช้แทนอุปกรณ์ทางด้านอินพุทต่าง ๆ ได้แก่ สวิทซ์ ไฟฟ้าหรือหน้าสัมผัสของรีเลย์ ตัวตั้งเวลา ตัวนับ หรือรีเลย์ภายในเป็นต้น โดยที่อุปกรณ์นั้นปกติจะมีค่า สภาวะเป็นลอจิก “0” หน้าสัมผัสปกติเปิดหรือ N.O.
+--] / [--+	จะใช้แทนอุปกรณ์ทางด้านอินพุทต่างๆ ได้แก่ สวิทซ์ไฟฟ้าหรือหน้าสัมผัสของรีเลย์ ตัวตั้งเวลา ตัวนับ หรือรีเลย์ภายในเป็นต้น โดยที่อุปกรณ์นั้นปกติจะมีค่าสภาวะเป็นลอจิก “1”
+--( )--+	เอาท์พุทปกติไม่ทำงานจะใช้แทนอุปกรณ์ทางด้านเอาท์พุทต่างๆ ได้แก่ หลอดไฟฟ้ามอเตอร์ไฟฟ้า ขดลวดของรีเลย์ ขดลวดโซลีนอยด์ (Solenoid) ตัวตั้งเวลา ตัวนับ รีเลย์ภายใน เป็นต้น โดยที่ปกติแล้วอุปกรณ์ทางด้านเอาท์พุทจะมีค่าสภาวะเป็นลอจิก “0”
+--(/) --+	เอาท์พุทปกติทำงาน จะใช้แทนอุปกรณ์ทางด้านเอาท์พุทต่างๆ ได้แก่ หลอดไฟฟ้ามอเตอร์ไฟฟ้า ขดลวดของรีเลย์ ขดลวดโซลีนอยด์ (Solenoid) ตัวตั้งเวลา ตัวนับ รีเลย์ภายใน เป็นต้น โดยที่ปกติแล้วอุปกรณ์ทางด้านเอาท์พุทจะมีค่า สภาวะเป็น ลอจิก “1”

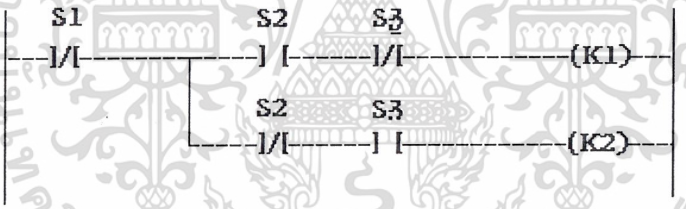
ตารางที่ 2-4 แสดงถึงตัวอย่างของการใช้ฟังก์ชั่นลอจิก และสัญลักษณ์พื้นฐานในการเขียนโปรแกรมแลดเดอร์ของ PLC/PC

ภาษาแลดเดอร์ไดอะแกรม เป็นการโปรแกรมที่เข้าใจได้ง่าย เนื่องจากมีลักษณะคล้ายกับรีเลย์ไดอะแกรมจึงทำได้ง่าย ดังตัวอย่างแสดงให้เห็นความสัมพันธ์ระหว่าง รีเลย์ไดอะแกรมซึ่งเป็นวงจรควบคุมการกลับทางหมุนของมอเตอร์แบบ JOGGING ในรปที่ 2-10 และเปลี่ยนมาใช้ PLC/PC ควบคุมโดยเขียนเป็นโปรแกรมแลดเดอร์ไดอะแกรม ในรปที่ 2-11 แลดเดอร์ไดอะแกรมจะแบ่งออกเป็นส่วน ๆ แต่ละส่วนคือ เอาท์พุท 1 จุดหรือเอาท์พุทมากกว่า 1 จุด แต่เอาท์พุทเหล่านี้จะมีลอจิกเหมือนกันแต่ละส่วนเหล่านี้เรียกว่า “รังค์ (RUNG)” ดังในรูปที่ 2-12 เป็นโปรแกรมที่ให้ผลการควบคุมเหมือนเดิม แต่แยกออกเป็น 2 รังค์

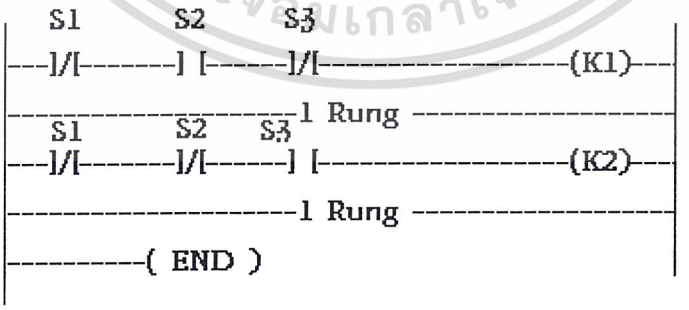
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2- 10 วงจรรีเลย์ควบคุมการหมุนกลับทางของมอเตอร์แบบ JOGGING



รูปที่ 2-11 แลดเดอร์ไดอะแกรมควบคุมการหมุนกลับทางของมอเตอร์แบบ JOGGING



รูปที่ 2-12 แสดงการแบ่งแลดเดอร์ไดอะแกรมออกเป็นรังค์

การโปรแกรมด้วยแลดเดอร์ไดอะแกรมจะสะดวกและมีประสิทธิภาพมาก แต่ข้อเสียของการโปรแกรมแบบนี้ คืออุปกรณ์ที่ใช้ในการเขียนโปรแกรมจะต้องมีหน่วยแสดงผลที่สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

แสดงอักษรหรืออุปกรณ์ที่ใช้ในการโปรแกรมแบบนี้มักจะใช้จอ CRT แต่ปัจจุบันจะมีซอฟท์แวร์สนับสนุนการเขียนแลดเดอร์บนคอมพิวเตอร์และเชื่อมโยงข้อมูลกับ PLC/PC ทางพอร์ทอนุกรม โดยคอมพิวเตอร์ทำหน้าที่ในการสร้าง และแก้ไขแลดเดอร์ไดอะแกรม แล้วทำการแปลเป็นชุดคำสั่งของเครื่องควบคุมนอกจากนั้นแล้วขณะที่เครื่องควบคุมทำงานอยู่ก็สามารถตรวจสอบการทำงานได้โดยการแสดงผลซึ่งจะแสดงเป็นแลดเดอร์ไดอะแกรมและเมื่ออินพุท หรือเอาท์พุทตัวใดมีการเปลี่ยนแปลงสภาวะก็จะมีการเปลี่ยนแปลงตามสภาวะของอินพุทหรือเอาท์พุทตัวนั้นๆ ด้วย

2.3.2 โปรแกรมภาษาคำสั่งแบบบูลีน

การโปรแกรมชนิดนี้มีลักษณะคล้ายกับสัญลักษณ์ของพีชคณิตบูลีนนอกจากจะเป็นสัญลักษณ์ ของพีชคณิตบูลีนแล้วก็ยังมีการใช้ฟังก์ชั่นพิเศษต่างๆ อีกมากเช่น ตัวนับ ตัวตัวเวลา เป็นต้น ภาษาคำสั่งบูลีน จะมีความสัมพันธ์กันกับ ภาษาแลดเดอร์ไดอะแกรม สามารถที่จะแปลความหมายถึงกันได้ดังตัวอย่างในตารางที่2-5 ซึ่งเป็นชุดคำสั่งบูลีนที่เขียนจากแลดเดอร์ไดอะแกรมในรูปที่ 2-12 การเขียนโปรแกรมเป็นแลดเดอร์ไดอะแกรมสามารถแสดงกระบวนการของโปรแกรมได้ง่ายต่อการเข้าใจ แต่ไม่สะดวกในการป้อนลงบนเครื่องควบคุม จึงมีการแปลงจากแลดเดอร์ไดอะแกรม เป็นภาษาคำสั่งบูลีน เพื่อป้อนลงชุดรับข้อมูลแบบพกพาของเครื่องควบคุม PLC/PC ซึ่งจะกระทำได้สะดวกมากขึ้น

แอดเดรส	คำสั่ง	อุปกรณ์อินพุท/เอาท์พุท
0000	LD NOT	S1
0001	AND	S2
0002	AND NOT	S3
0003	OUT	K1
0004	LD NOT	S1
0005	AND NOT	S2
0006	AND	S3
0007	OUT	K2
0008	END	

ตารางที่ 2-5 ตัวอย่างโปรแกรมชุดคำสั่งบูลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

30947

บทที่ 3
เครื่องควบคุมแบบตรรกที่โปรแกรมได้กับการติดต่อสื่อสาร

เครื่องควบคุม PLC /PC นอกจากจะใช้ในการควบคุมเครื่องจักรแล้ว ยังได้มีการพัฒนาขีดความสามารถในการสื่อสารข้อมูล การควบคุมระยะไกล การจัดการข้อมูล เพื่อเพิ่มประสิทธิภาพในการควบคุมให้กว้างมากยิ่งขึ้น ในบทนี้จะกล่าวถึงพื้นฐานการสื่อสารข้อมูลทั่วไป และการสื่อสารข้อมูลของเครื่องควบคุม PLC/PC

3.1 การติดต่อสื่อสารข้อมูลทั่วไป

ส่วนประกอบเบื้องต้นในการสื่อสารข้อมูลแบ่งได้ออกเป็น 3 ส่วน คือ

1. ฝ่ายกำเนิดข้อมูล (Transmitter)
2. ตัวกลางในการส่งผ่านข้อมูล (Medium)
3. ฝ่ายรับข้อมูล (Receiver)

ซึ่งแต่ละส่วนมีความสัมพันธ์กันดังรูปที่ 3-1

รูปที่ 3-1 แสดงความสัมพันธ์ของส่วนประกอบหลักในการสื่อสารข้อมูล

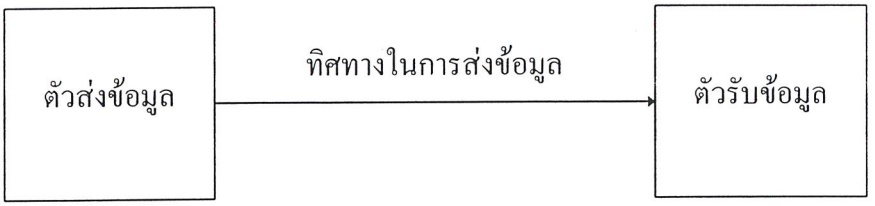
การจำแนกรูปแบบในการสื่อสารข้อมูลอาจจัดแบ่งได้หลายแบบ เช่น แบ่งตามชนิดของสัญญาณของข้อมูล คือ การสื่อสารข้อมูล อนาล็อก (Analog) และ การสื่อสารข้อมูลดิจิตอล (Digital) หรือแบ่งตามวิธีการส่งข้อมูล ในบทนี้จะกล่าวถึงการสื่อสารข้อมูลดิจิตอล ซึ่งเป็นข้อมูลชนิดที่เกี่ยวข้องกับเครื่องคอมพิวเตอร์และ เครื่องควบคุม PLC/PC

3.1.1 วิธีการส่งข้อมูล

แบ่งได้ออกเป็น 3 แบบคือ

1. การส่งแบบทิศทางเดียว (Simplex) การส่งแบบนี้ทิศทางการส่งจะคงที่โดยกำหนดในครั้งแรกว่าฝ่ายใดเป็นฝ่ายส่ง ฝ่ายใดเป็นฝ่ายรับ การส่งแบบนี้แม้ไม่คล่องตัวแต่เหมาะสมในการใช้งานบางประเภท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

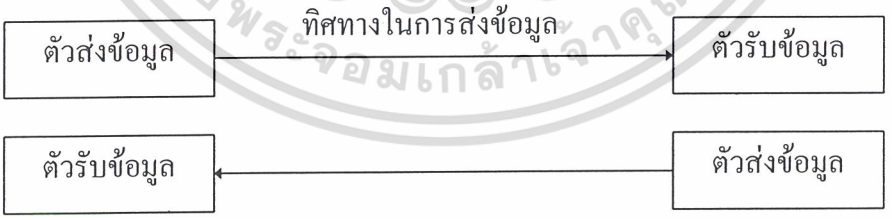


รูปที่ 3-2 การส่งแบบทิศทางเดียว (Simplex)

2. การส่งแบบฮาล์ฟดูเพล็กซ์ (Half Duplex) เป็นการส่งข้อมูลในทิศทางใดทิศทางหนึ่งในเวลาใดเวลาหนึ่ง คือ ทั้ง 2 สถานีสามารถผลัดกันส่งได้แต่จะส่งพร้อมๆกันไม่ได้

รูปที่ 3-3 การส่งแบบฮาล์ฟดูเพล็กซ์ (Half Duplex)

3. การส่งแบบสองทิศทาง(Full Duplex) เป็นการส่งข้อมูลในทิศทางใด ก็ได้ เวลาใดก็ได้ คือ สามารถส่งและรับข้อมูลในเวลาเดียวกันได้



รูปที่ 3-4 การส่งแบบสองทิศทาง (Full Duplex)

วิธีการสื่อสารข้อมูลดิจิตอลของคอมพิวเตอร์แบ่งเป็น 2 ชนิด

1. การสื่อสารข้อมูลแบบขนาน (Parallel)

เป็นการติดต่อสื่อสารโดยส่งข้อมูลออกเป็นครั้งละ 1 ไบต์ คือ ครั้งละ 8 บิท จากอุปกรณ์ส่งไปยังอุปกรณ์รับ ดังนั้นตัวกลางระหว่างเครื่องคอมพิวเตอร์ทั้งสองเครื่องจะต้องมีช่องทางให้ข้อมูล

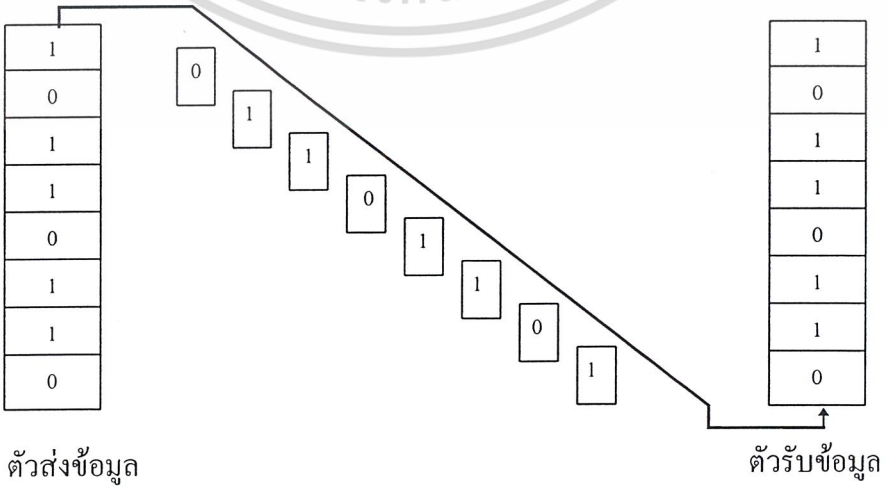
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดินทางอย่างน้อย 8 ช่องทาง โดยมากจะใช้เป็นสายขนานแต่เนื่องจากมีสัญญาณสูญหายไปกับอิมพิแดนซ์ของสาย ซึ่งมีความสัมพันธ์กับระยะทางของสาย ดังนั้นระยะทางระหว่างเครื่องสองเครื่อง ไม่ควรจะเกิน 100 ฟุต ทำให้การส่งแบบนี้ได้ระยะทางไม่ไกลนักแต่สามารถส่งข้อมูลได้รวดเร็วเพราะส่งออกมาพร้อมกันทีละ 8 บิท อุปกรณ์ที่ติดต่อแบบขนานกับคอมพิวเตอร์ เช่น เครื่องพิมพ์ เป็นต้น

รูปที่ 3-5 การสื่อสารข้อมูลแบบขนาน

2. การสื่อสารข้อมูลแบบอนุกรม (Serial)

ในการติดต่อแบบนี้ ข้อมูลจะถูกส่งออกมาทีละบิท ระหว่าง จุดส่งและจุดรับ ซึ่งการส่งข้อมูลแบบนี้จะช้ากว่าการส่งแบบขนาน แต่ใช้ตัวกลางในการสื่อสารเพียงช่องเดียวหรือสายเพียงคู่เดียว ทำให้ค่าใช้จ่ายสำหรับสื่อกลางถูกกว่าแบบขนาน การส่งแบบนี้ใช้สำหรับส่งในระยะทางไกลมากกว่า 100 ฟุต ข้อมูลจากจุดส่งจะต้องถูกเปลี่ยนให้เป็นอนุกรมก่อนแล้วค่อยทยอยส่งออกไปทีละบิท ไปยังจุดรับ ที่จุดรับจะต้องมีรูปแบบในการเปลี่ยนข้อมูลที่ถูกส่งมาให้เป็นแบบขนาน ในการแปลงต้องมีรูปแบบที่เหมาะสมเพื่อป้องกันการผิดพลาดของข้อมูล



รูปที่ 3-6 การสื่อสารข้อมูลแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ข้อตกลงในการติดต่อสื่อสาร (โปรโตคอล : Protocols)

ในการสื่อสารข้อมูลจะต้องมีกฎหรือข้อกำหนดในการสื่อสารข้อมูล หรือที่นิยมเรียกว่า โปรโตคอล (Protocols) ซึ่งเป็นส่วนที่จะกำหนดมาตรฐานในการควบคุม และ จัดการระบบการสื่อสารข้อมูล

สำหรับรายละเอียดที่กล่าวในหัวข้อนี้จะเกี่ยวข้องเฉพาะในส่วนของโปรโตคอลการควบคุมการเชื่อมโยงข้อมูล (Data Link Control Protocols หรือ DLCP) ซึ่งจัดการในส่วนของขั้นตอนและหลักการต่างๆคือโครงสร้างและรายละเอียดของข้อมูล วิธีในการสื่อสารข้อมูลการตรวจสอบแก้ได้ไขความผิดพลาดของข้อมูล และขบวนการในการควบคุมการติดต่อสื่อสาร โดย DLCP แบ่งได้ตามโครงสร้างของข้อมูล 2 แบบ คือ Byte-Oriented Protocols และ Bit-Oriented Protocols

3.1.2.1. ไบท์โอเรียนโปรโตคอล (Byte-oriented protocols)

โปรโตคอลแบบนี้เป็นโปรโตคอลที่การสื่อสารข้อมูล และการควบคุมการทำงานจะทำโดยใช้ลักษณะข้อมูลที่เป็นตัวอักษร (character) หรือไบท์ (byte) หรือ อาจเรียกว่า Character Oriented Protocols ซึ่งแบ่งออกเป็น

1. อะซิงโครนัสโปรโตคอล (Asynchronous protocols) โปรโตคอลในการสื่อสารข้อมูลนี้จะใช้การสื่อสารข้อมูลแบบ Half-Duplex ที่มีลักษณะการสื่อสารข้อมูลแบบอะซิงโครนัส ซึ่งเป็นการสื่อสารข้อมูลแบบพื้นฐานที่ใช้งานมาเป็นเวลานานแล้ว จึงมีรายละเอียดและขั้นตอนในการสื่อสารข้อมูลที่ทำให้มีโอกาสเกิดความผิดพลาดได้น้อย และยังมีข้อดีที่การสื่อสารข้อมูลแบบนี้มีในโครงสร้างการทำงานที่ง่าย อุปกรณ์ที่ใช้ในการสื่อสารข้อมูลก็ไม่สลับซับซ้อน และมีราคาถูก โปรโตคอลแบบนี้จึงเหมาะสำหรับใช้ในระบบขนาดเล็ก

2. ไบนารี่ซิงโครนัสโปรโตคอล (Binary synchronous protocols) โปรโตคอลแบบนี้จะมีลักษณะการทำงานที่ใช้งานข้อมูลเป็นลักษณะไบท์ และยังคงใช้การสื่อสารข้อมูลแบบซิงโครนัส มีรายละเอียดและขั้นตอนในการสื่อสารข้อมูลที่ให้ความน่าเชื่อถือมากกว่า อีกทั้งยังสามารถใช้อัตราเร็วในการสื่อสารข้อมูลที่สูงกว่าโดยตัวอย่าง ของการสื่อสารข้อมูลแบบนี้ที่ได้กำหนดเป็นมาตรฐานแล้วคือ การสื่อสารข้อมูลตามมาตรฐาน BSC (Binary Synchronous Communications) ซึ่งเป็นโปรโตคอลที่มีลักษณะของข้อมูลแบบไบท์ที่ได้รับความนิยมนำไปใช้งาน

3.1.2.2 บิทโอเรียนท์โปรโตคอล (Bit-oriented protocols)

โปรโตคอลแบบนี้เป็นโปรโตคอลที่การสื่อสารข้อมูล และการควบคุมการทำงานจะทำโดยใช้ลักษณะข้อมูลที่เป็นบิทโดยมีตัวอย่างของการสื่อสารข้อมูลในลักษณะนี้ที่มีการกำหนดขึ้นเป็นมาตรฐานแล้ว คือ HDLC (High-level Data Link Control) โดยมีโครงสร้างของข้อมูลแบบซิงโครนัส เช่นเดียวกับ BSC แต่ต่างกันที่มีลักษณะของข้อมูลเป็นแบบบิท ซึ่งโปรโตคอลแบบนี้มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีที่สามารถสื่อสารข้อมูลแบบสองทางได้ทำให้การสื่อสารข้อมูลได้รวดเร็วกว่าแต่โปรโตคอลแบบนี้ก็มีรายละเอียด และโครงสร้างในการสื่อสารข้อมูลที่สลับซับซ้อนมากทำให้การควบคุมการทำงานทำได้ยากและต้องใช้อุปกรณ์ที่มีราคาสูง จึงไม่เหมาะที่จะนำไปใช้งานกับระบบขนาดเล็ก

3.1.2.3 แพ็กเก็ท (Packet of Information)

รูปแบบของแพ็กเก็ทในระบบโครงข่ายจะประกอบด้วยส่วนต่าง ๆ ดังนี้

1. HEADER จะประกอบด้วย

-Peramble or start of packet indicator เป็นส่วนเริ่มแรกของแพ็กเก็ท และในบางระบบอาจใช้ในการซิงค์กับสัญญาณนาฬิกาของตัวส่งและตัวรับด้วย

-Control information ส่วนนี้เป็นข้อมูลที่บอกถึงวัตถุประสงค์ของแพ็กเก็ทนั้นว่าใช้ทำอะไรเช่นเพื่อการจัดการระบบเพื่อดู Status ของ mode หรืออื่นๆ

นอกจากส่วนต่าง ๆ เหล่านี้แล้ว ในส่วน Header อาจจะมีส่วนที่เป็น Sequencial Number เป็นส่วนที่บอกให้ทราบถึงลำดับของแพ็กเก็ทในกรณีที่ข้อมูลมีความยาวหลายแพ็กเก็ท

2. INFORMATION

-Data field เป็นส่วนของข้อมูลจริงที่ต้องการจะส่ง

3. TAILER

-Frame Check Sequence (FCS) เป็นส่วนที่ใช้ตรวจสอบความถูกต้องของข้อมูลซึ่งอาจเป็น parity bit, Check sum หรือ CRC เป็นต้น

-End of Packet Indicator เป็นส่วนที่บอกให้ทราบว่าสิ้นสุดของข้อมูลแล้ว

PREAMBLE	DA	SA	CONTROL	INFORMATION	FCS	STOP
HEADER			INFORMATION		TAILER	

รูปที่ 3-7 แสดงโครงสร้างของแพ็กเก็ท

3.1.3 การควบคุมความผิดพลาดในการส่งข้อมูล (Error Control)

การที่วงจรส่งข้อมูลขาดชั่วขณะ และผลกระทบต่อสัญญาณรบกวน (Noise) ทำให้ระดับแรงดันไฟฟ้าลดลงเป็นผลให้เกิดความผิดพลาดในการส่ง ดังนั้นต้องทำการค้นหาความผิดพลาดที่เกิดขึ้นระหว่างการส่งข้อมูลและแก้ไขความผิดพลาดให้ถูกต้อง

3.1.3.1 วิธีการเพิ่มบิตเข้าไปที่ข้อมูล(Parity Bit) เพื่อตรวจสอบความผิดพลาดมี 2 วิธี

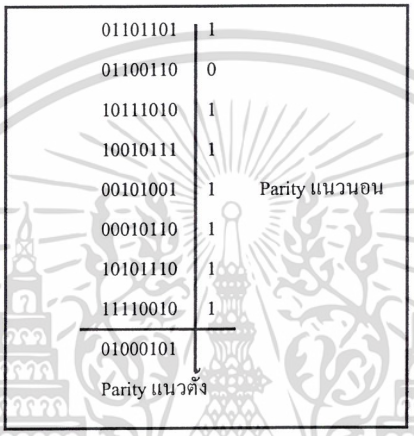
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 .วิธี Parity แนวนอน

จะเพิ่ม 1 บิต เข้าไปที่แต่ละตัวอักษรที่จะส่งโดยจะกำหนดว่าเป็นการตรวจสอบแบบ Odd หรือ Even Number แล้วผลรวมของข้อมูลจะเป็น Odd Number ที่ด้านรับจะทำการตรวจสอบว่าผลรวมของข้อมูลจะเป็น Odd Number หรือ Even Number

2 .วิธี Parity แนวตั้ง

จะตรวจสอบความผิดพลาดโดยการเพิ่ม 1 บิตเข้าไปที่ในแต่ละบล็อก



รูปที่ 3-8 แสดงรูปแบบการตรวจสอบด้วย Parity Bit

3.1.3.2. วิธี Patrol Diffuse Inspection หรือ การใช้ FCS : Frame Check Sequence

สำหรับข้อมูลอันหนึ่งจะสร้าง Error Inspection Sign (CRC Sign) ขนาด 2 Byte ที่คำนวณได้บนพื้นฐานของกฎที่กำหนดไว้ แล้วเพิ่มเครื่องหมายไปที่ข้อมูลอันนั้นที่ทางด้านรับจะตรวจสอบข้อมูลโดย Inspection Sign ได้กำหนดไว้โดยสูตรการคำนวณที่เหมือนกัน (Blast Error ความผิดพลาดของข้อมูลที่ต่อเนื่อง) ก็สามารถตรวจสอบได้ ทำให้มีความน่าเชื่อถือสูง

3.1.4 อัตราการส่งข้อมูล (Data Transmission Rate)

Transmission Rate นั้นแยกออกได้เป็น "Data Transmission Speed" และ "Modulation Speed"

3.1.4.1. Data Transmission Speed

Data Transmission Speed หรือ Bit rate นั้นจะแสดงในรูปของจำนวน Bit ที่สามารถส่งได้ภายใน 1 วินาที มีหน่วยเป็น บิต ต่อ วินาที โดยเขียนว่า Bit / Sec หรือ Bps จะแสดง Definition ของ Data Transmission Speed จะแสดงด้วยสมการข้างล่างนี้

$$S = (1/T) \log_2 n$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

S : Data Transmission Speed

T : Continue Time ของ 1 pulse ของ Transmission Pulse

n : จำนวนของค่าสภาวะใน 1 Pulse

ในกรณีที่ Pulse 1 ลูก มีค่า 2 ค่าคือ 0 และ 1 เป็น สมการจะเป็น S = 1/T โดยที่ T เป็นเวลาต่อเนื่อง(Continue Time) ของ Pulse 1 1/T จึงแสดงถึงจำนวนของ Pulse ใน 1 วินาที ซึ่งก็คือจำนวนบิทที่สามารถส่งไปได้ใน 1 วินาทีนั่นเอง

3.1.4.2 Modulation Speed

Modulation Speed หรือ Baud rate ก็คือจำนวนครั้งที่สามารถ Modulate ได้ใน 1 วินาที วิธีการModulation นั้นมี Amplitude Mod , Frequency Mod , Phase Mod และอื่นๆ ในกรณีที่ T แสดงถึงช่วงเวลาในการเปลี่ยนแปลงสภาวะนั้น จะสามารถกำหนดได้ดังสมการข้างล่างนี้โดยหน่วยจะเป็น Baud

B = 1/T

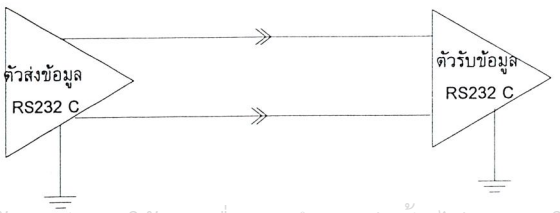
B : Modulation Speed (Baud)

T : เวลาในการเปลี่ยนจากสภาวะหนึ่งเป็นอีกสภาวะ

3.1.5 มาตรฐานในการสื่อสารข้อมูลแบบอนุกรม

3.1.5.1. การสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C

การสื่อสารข้อมูลแบบอนุกรมที่ใช้งานอยู่ในปัจจุบันนั้น ได้มีการกำหนดมาตรฐานการรับส่งข้อมูลไว้หลายแบบด้วยกัน แต่ที่ได้รับความนิยมนำมาใช้งานอย่างมาก คือ การสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C และที่มาตรฐานนี้เป็นที่นิยมเนื่องจากเป็น ระบบการสื่อสารข้อมูลที่ใช้ในเครื่องไมโครคอมพิวเตอร์ IBM PC ซึ่งเป็นคอมพิวเตอร์ที่มีใช้อย่างแพร่หลายมากจากอดีตจนถึงปัจจุบันมาตรฐานการสื่อสารนี้ในการออกแบบเบื้องต้นได้ออกแบบ สำหรับการเชื่อมต่อกับเครื่องโมเด็ม (MODEM) ซึ่งเป็นอุปกรณ์ที่ใช้การสื่อสารข้อมูลระหว่างคอมพิวเตอร์ผ่านทางสายโทรศัพท์ ซึ่งทำให้อัตราการรับส่งข้อมูลถูกจำกัดให้มีค่าที่ค่อนข้างต่ำ มาตรฐาน RS-232C นี้ได้ออกแบบให้มีโครงสร้างการสื่อสารเป็นแบบจุดต่อจุดเท่านั้น โดยมีลักษณะสมบัติทางไฟฟ้าและทางกายภาพ ดังแสดงในตารางที่ 5 และรูปที่ 3-9

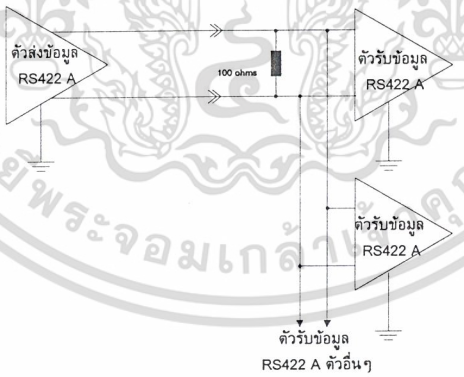


รูปที่ 3-9 แสดงโครงสร้างของการสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5.2. การสื่อสารข้อมูลแบบอนุกรมมาตรฐาน RS-422A

ในการออกแบบระบบสื่อสารข้อมูลจากที่ผ่านมาจนถึงปัจจุบัน ได้มีการพยายามที่จะออกแบบให้การสื่อสารข้อมูลได้รวดเร็วขึ้นและมีระยะในการสื่อสารที่มากขึ้นด้วย ซึ่งที่ผ่านมาการสื่อสารข้อมูลตามมาตรฐาน RS-232C ได้ออกแบบเพื่อใช้เชื่อมต่อกับโมเด็มเท่านั้น จึงไม่ได้คำนึงถึงความเร็วและระยะทางในการสื่อสารต่อมาได้มีมาตรฐานในการสื่อสารข้อมูลใหม่ที่ได้ออกแบบมาเพื่อรองรับความต้องการของผู้ใช้งานที่ต้องการให้การรับส่งข้อมูลได้ระยะทางไกลและรวดเร็ว มาตรฐานนี้คือ RS-422A ซึ่งการที่มาตรฐานสื่อสารนี้สามารถรับส่งข้อมูลได้ไกลและรวดเร็วขึ้นเนื่อง มาจากหลักการที่ใช้สัญญาณเป็นแบบดิฟเฟอเรนเชียลดังแสดงในรูปที่ 3-10 ซึ่งหลักการก็คือสัญญาณที่รับส่งจะเป็นการเปรียบเทียบระหว่างสัญญาณ 2 เส้นเทียบกับมาตรฐาน RS-232C ที่สัญญาณทุกสัญญาณจะเทียบกับกราวด์ ซึ่งในการสื่อสารในระยะทางไกลๆ แล้วสัญญาณจะถูกลดทอนไปและเมื่อสัญญาณถูกลดทอนถึงจุดๆ หนึ่งสัญญาณนั้นก็จะผิดพลาดไปจากความเป็นจริง ทำให้การรับส่งข้อมูลเกิดผิดพลาดขึ้น แต่สำหรับสัญญาณแบบดิฟเฟอเรนเชียล การลดทอนของสัญญาณก็จะไปลดทอนทั้งสองสายด้วยค่าที่เท่ากันหรือใกล้เคียงกันและความแตกต่างของสัญญาณทั้ง 2 เส้น จากตัวส่งไปยังตัวรับก็ยังมีค่าเท่าเดิมหรือเปลี่ยนแปลงน้อย จึงทำให้ผลของการลดทอนต่อสัญญาณที่ระยะการสื่อสารที่ไกลไม่มีผลต่อการสื่อสารข้อมูล พร้อมทั้งสามารถติดต่อกับตัวรับได้ถึง 10 ตัว ดังแสดงตารางในตารางที่ 3-1 การเปรียบเทียบมาตรฐานการสื่อสารข้อมูล



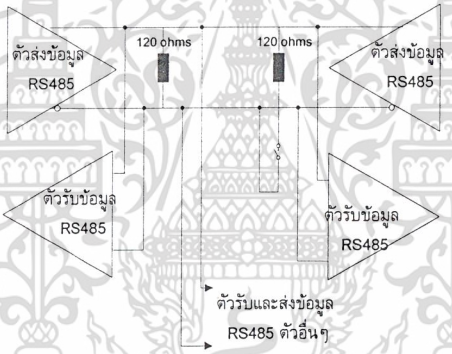
รูปที่ 3-10 แสดงโครงสร้างของการสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-422A

3.1.5.3. การสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-485

การสื่อสารข้อมูลตามมาตรฐานที่กล่าวมาข้างต้นคือ RS-232C นั้นเป็นมาตรฐานการสื่อสารข้อมูลในแบบที่ใช้สื่อสารระหว่างอุปกรณ์ต่ออุปกรณ์ หรือ จุดต่อจุด (Point-to-Point) ส่วน RS-422A นั้นเป็นมาตรฐานที่พัฒนามาจาก RS-232C ให้ได้ระยะทางไกลขึ้นและอัตราการสื่อสารเพิ่มขึ้น แต่ก็ยังเป็นการสื่อสารข้อมูลจากอุปกรณ์หนึ่งตัวไปยังอุปกรณ์อื่นๆ ได้สูงสุด 10 ตัวเท่านั้น ไม่สามารถส่งย้อนกลับจากอุปกรณ์ตัวรับมาตัวส่งได้ หรือกล่าวได้ว่าการสื่อสารข้อมูลตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐาน RS-422A นั้นเป็นการสื่อสารข้อมูลแบบ Simplex คือทิศทางของข้อมูลเป็นแบบทางเดียวตลอดเวลา ดังนั้นถ้าต้องการออกแบบระบบให้เป็นลักษณะโครงข่ายข้อมูลก็จะไม่สามารถทำได้ จึงมีการพัฒนามาตรฐานการสื่อสารข้อมูลขึ้นใหม่เพื่อรองรับความต้องการนี้ คือ มาตรฐาน RS-485 ซึ่งเป็นมาตรฐานที่อาศัยหลักการของสัญญาณแบบดิฟเฟอเรนเชียลเช่นเดียวกับมาตรฐาน RS-422A แต่สามารถสื่อสารข้อมูลได้ทั้ง 2 ทิศทางในสายสัญญาณเพียงคู่เดียว ซึ่งก็คือการสื่อสารข้อมูลแบบ Half-Duplex จากผลของการใช้สัญญาณในลักษณะดิฟเฟอเรนเชียลนี้ ทำให้ระยะทางและความเร็วในการสื่อสารข้อมูลมีค่าสูง เช่นเดียวกับมาตรฐานการสื่อสารข้อมูล RS-422A แต่มาตรฐาน RS-485 สามารถที่จะสื่อสารระหว่างอุปกรณ์ทั้งการส่งของอุปกรณ์ได้สูงสุด 32 ตัว หรืออาจกล่าวได้ว่าการสื่อสารตามมาตรฐาน RS-485 เป็นการสื่อสารแบบหลายจุด (Multipoint-Communication) ดังแสดงค่าเปรียบเทียบในตารางที่3-1 และแสดงโครงสร้างในรูปที่ 3-11



รูปที่ 3-11 แสดงโครงสร้างของการสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-485

ตารางที่ 3-1 แสดงค่าเปรียบเทียบคุณสมบัติทางไฟฟ้าของ RS-232 C , RS-422A , และ RS -485

พารามิเตอร์	RS-232-C	RS-422-A	RS-485
โหมดการทำงาน	Single-ended	Differential	Differential
จำนวนของตัวรับและส่งที่ยอมรับได้	1 ตัวส่ง 1 ตัวรับ	1 ตัวส่ง 10 ตัวรับ	32 ตัวส่ง 32 ตัวรับ
ความยาวของคู่สายสูงสุด	50 ฟุต	4000 ฟุต	4000 ฟุต
อัตราการส่งข้อมูลสูงสุด	20k bps	10 M bps	10 M bps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์	RS-232-C	RS-422-A	RS-485
maximum common mode voltage	2.5 V	6V - 2.5 V	12 V -7 V
Driver output	5V ต่ำสุด 15V สูงสุด	2V ต่ำสุด	1.5 V ต่ำสุด
Driver load (ohm)	3K ถึง 7K	100 ต่ำสุด	60 ต่ำสุด
Driver slew rate	30V/us สูงสุด	NA	NA
กระแสลิมิต เมื่อเอาท์พุทลัดวงจร	500mA ลัดวงจรกับ VCC หรือ GND	150 mA ลัดวงจรกับ GND	150 mA ลัดกับGND 250 mA ลัดกับ12V
ค่าความต้านทานเอาท์พุทของตัวส่ง (โอห์ม)	NA (power on) 300 (power off)	NA (power on) 60k (power off)	120k (power on) 120k (power off)
ค่าความต้านทานอินพุทตัวรับ	3k ถึง 7k ohm	4k ohm	12k ohm
ความไวของตัวรับ	3V	200 mV	200 mV

3.2 รูปแบบการสื่อสารข้อมูลของเครื่องควบคุมแบบตรรกที่โปรแกรมได้

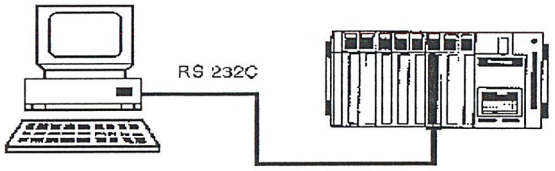
การต่อวงจรในการติดต่อสื่อสารข้อมูลของเครื่องควบคุม PLC/PC อาจแบ่งออกได้ 3 แบบด้วยกันคือ

1. การต่อวงจรแบบจุดต่อจุด (Point to Point)
2. การต่อวงจรแบบหลายจุด (Multi point)
3. การต่อวงจรแบบเครือข่ายแลน (LAN: Local Area Network)

3.2.1 การต่อวงจรแบบจุดต่อจุด

ในการติดต่อสื่อสารข้อมูลแบบจุดต่อจุด จะใช้มาตรฐานการสื่อสารแบบ RS 232 C ซึ่งเป็นการติดต่อในระยะทางที่ไม่ไกลมาก เช่น การติดต่อระหว่างเครื่องควบคุมPLC/PC กับเครื่องคอมพิวเตอร์ เพื่อทำการตรวจสอบสภาวะการทำงาน หรือ การควบคุมจากคอมพิวเตอร์ หรือสนับสนุนการทำงานอื่นๆ การติดต่อระหว่างเครื่องควบคุม PLC/PC กับอุปกรณ์อื่น เช่น โมเด็ม (MODEM) อุปกรณ์ควบคุมสัมผัสหน้าจอ (Touch Screen) คีย์บอร์ด (Key board) เครื่องอ่านรหัสแถบ (Bar-code Reader) เครื่องพิมพ์ เครื่องบันทึก (Recorder) เครื่องรายงานผล (Reporter) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

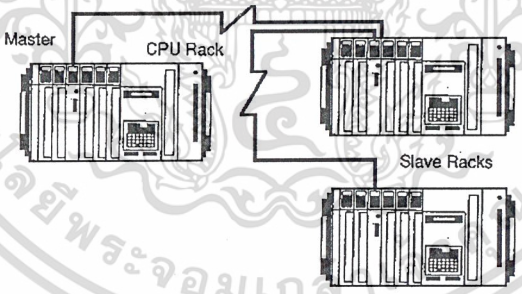


รูปที่ 3-12 ระบบการติดต่อแบบจุดต่อจุด

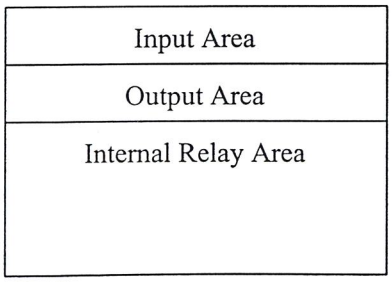
3.2.2 การต่อวงจรแบบหลายจุด (Multi point)

3.2.2.1. การติดต่อแบบอินพุทและเอาท์พุทระยะไกล (Remote I/O System)

เป็นการเชื่อมโยงสื่อสารข้อมูลในลักษณะ ที่ตัวประมวลผลของเครื่องควบคุมหลัก กับปลายทางอุปกรณ์ที่ถูกควบคุมอยู่ไกลกัน และอาจมีมากกว่า 1 จุด ดังนั้นจึงมีความจำเป็นที่จะมีการเชื่อมโยงในระยะไกล ในการติดต่อลักษณะนี้จะใช้มาตรฐานการสื่อสารแบบ RS 485 บนตัวนำสองเส้น โดยที่ตัวควบคุมหลัก(Master) จะทำการรับอินพุททั้งจากพื้นที่ของตัวควบคุม และหน่วยระยะไกล(Slave) จากนั้นจะทำการประมวลผล แล้วส่งเอาท์พุทไปยังส่วนต่างๆ โดยที่อินพุทและเอาท์พุทของหน่วยระยะไกลจะจองพื้นที่ของรีเลย์ภายใน (Internal Relay) ของตัวควบคุมหลักโดยการเซทระบบนั้นจะแตกต่างกันไปตามบริษัทผู้ผลิต



รูปที่ 3-13 การเชื่อมโยงแบบอินพุทและเอาท์พุทระยะไกล (Remote I/O System)



รูปที่ 3-14 แสดงการจำลองพื้นที่ของตัวควบคุมหลักขณะทำงานปกติ

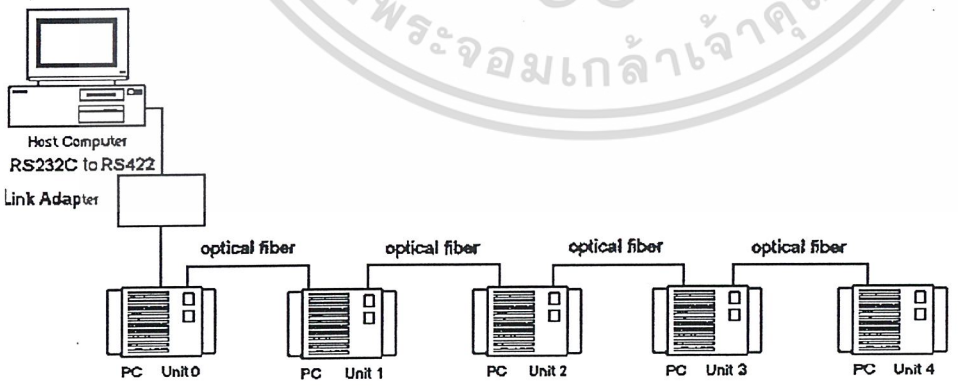
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Input Area
Output Area
Internal Relay Area
Input Remote Output Remote

รูปที่ 3-15 แสดงการจำลองพื้นที่ของตัวควบคุมหลักขณะเมื่อมีการทำงานร่วมกับหน่วยอินพุทเอาท์พุทระยะไกล

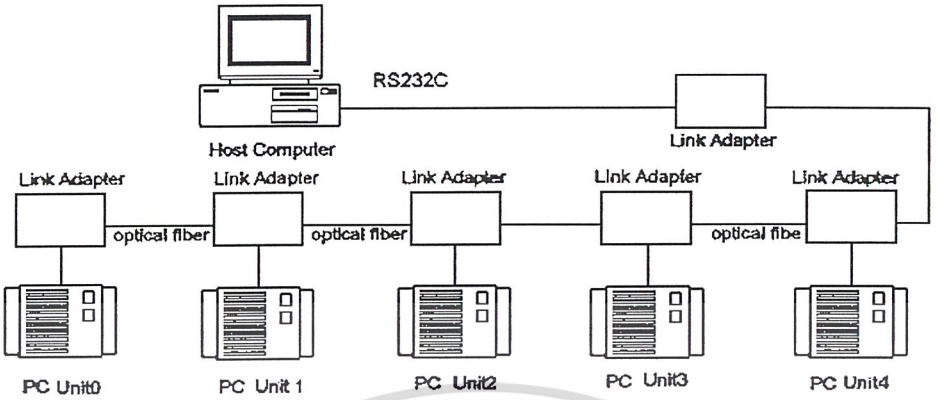
3.2.2.2 การติดต่อแบบพีซีลิ้งค์ (PC Link System)

เป็นการเชื่อมโยงสื่อสารข้อมูลในลักษณะของการแบ่งพื้นที่ในการอ่านเขียนของ PLC/PC แต่ละตัว เครื่องควบคุม PLC/PC แต่ละตัวสามารถรับรู้ข่าวสาร ซึ่งกันและกันได้ มาตรฐานในการสื่อสารข้อมูลชนิดนี้มักจะเป็นแบบ RS422A หรือเป็นรูปแบบการสื่อสารผ่าน สายใยแก้วนำแสง (Fiber optic) หรือผ่านสายโคแอคเชียล โดยจะมีตัว Link Adapter เป็นตัวแปลงรูปแบบข้อมูลให้เป็นมาตรฐานตรงกัน อาจจะมีการเชื่อมโยงกับเครื่องคอมพิวเตอร์ เข้ามาในระบบการลิ้งค์ เพื่อสนับสนุนการทำงานของเครื่องควบคุม PLC/PC ซึ่งจะเรียกว่าโฮสคอมพิวเตอร์ (Host Computer) มีลักษณะการต่อวงจรดังนี้



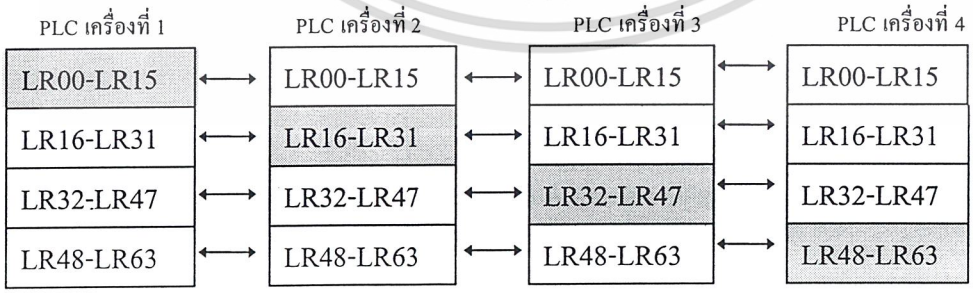
รูปที่ 3-16 การต่อแบบ Serial link

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-17 การต่อแบบ Parallel link

ตัวอย่างการจัดแบ่งพื้นที่ของเครื่องควบคุมPLC/PC กำหนดให้มีเครื่องควบคุมจำนวน 4 ชุด เชื่อมโยงในเครื่องควบคุมแต่ละเครื่องจะมีพื้นที่ของรีเลย์ที่ทำหน้าที่ในการเชื่อมโยงข้อมูล (Link -Relay) ถ้ามีเครื่องควบคุมที่เชื่อมต่อในระบบมากพื้นที่ก็จะถูกแบ่งออกไปตามจำนวน เช่น 4 ชุด พื้นที่จะถูกแบ่งออกเป็น 4 ส่วน เช่นเครื่องควบคุมมีพื้นที่ของ ลิ้งค์รีเลย์จำนวน 64 Word ก็จะถูกแบ่งเป็นส่วนละ 16 Word ส่วนแรกจะเป็นพื้นที่ของเครื่องควบคุมตัวที่ 1 ส่วนที่สองจะเป็นของเครื่องควบคุมตัวที่ 2 และสามตามลำดับ โดยพื้นที่ส่วนอื่นจะถูกกันไว้ไม่ให้สามารถเขียนข้อมูลลงไปได้ (Read Only) ยกเว้นพื้นที่ของตัวเอง ดังนั้นถ้า PLC/PC เครื่องที่ 1 ต้องการรับข้อมูลของPLC เครื่องที่ 3 ก็สามารถอ่านได้จากพื้นที่ของเครื่องควบคุมส่วนที่สาม และถ้าต้องการส่งข้อมูลให้ก็เขียนลงในพื้นที่ของส่วนแรก แล้วPLC ตัวที่ 3 จะทำการอ่านข้อมูลในส่วนแรก ก็จะสามารถทราบข้อมูลของเครื่องควบคุม PLC ตัวที่ 1

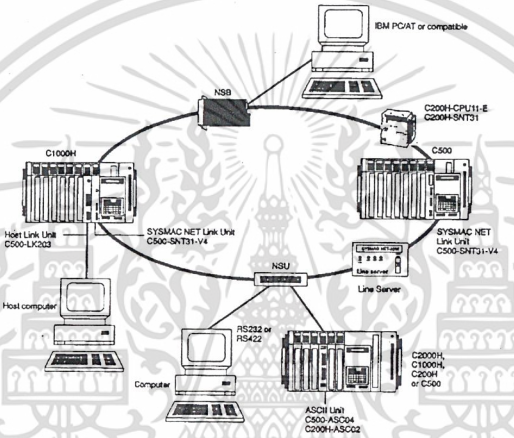


รูปที่ 3-18 การแบ่งพื้นที่สำหรับ PC Link System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

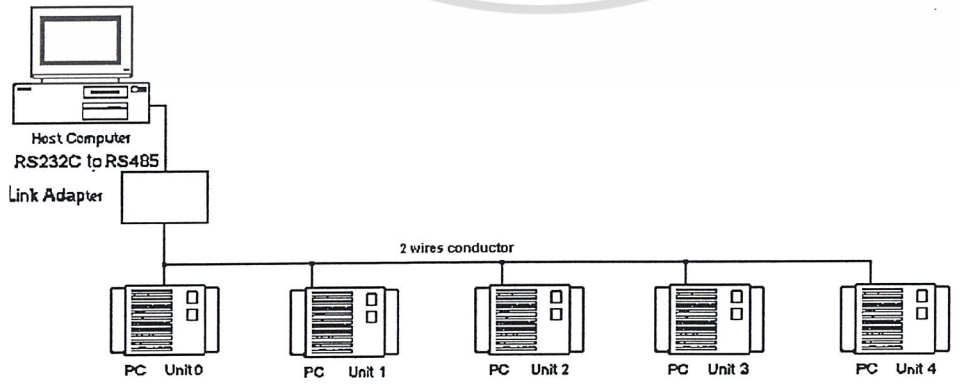
3.2.3 การต่อวงจรสื่อสารแบบเครือข่ายท้องถิ่น (Local area Network for PLC/PC)

การเชื่อมโยงในลักษณะของเครือข่ายท้องถิ่นนี้เป็นระบบขนาดใหญ่ประกอบด้วย อุปกรณ์หลายชนิดและหลายวงรอบด้วยกัน ซึ่งจะเป็นลักษณะเครือข่ายแบบวงแหวน (Ring Network) ในแต่ละวงรอบ สามารถที่จะเชื่อมโยงไปยังวงรอบอื่นๆ ได้ ซึ่งจะมีความซับซ้อนในระบบการติดต่อสื่อสารมากขึ้น รูปแบบสายนำสัญญาณจะมีทั้งสายใยแก้วนำแสง (Fiber optic) และสายตัวนำในแบบต่างๆ ในระบบโครงข่ายก็จะมีเครื่องคอมพิวเตอร์ส่วนบุคคลทำหน้าที่เป็นตัวจัดการข้อมูลเช่นเดียวกันกับเครื่องเซิฟเวอร์ (Computer Server) บนโครงข่ายของคอมพิวเตอร์



รูปที่ 3-19 ระบบการติดต่อสื่อสารแบบเครือข่ายท้องถิ่น

สำหรับระบบโครงข่ายที่ได้ออกแบบในวิทยานิพนธ์นี้เป็นการเชื่อมโยงแบบหลายจุดที่ใช้มาตรฐานการสื่อสารแบบ RS 485 โดยมีโฮสคอมพิวเตอร์เป็นตัวจัดการข้อมูลต่างๆให้กับเครื่องควบคุมที่เชื่อมต่อในระบบโครงข่าย ดังรูปต่อไปนี้ รายละเอียดจะได้กล่าวในบทที่ 5

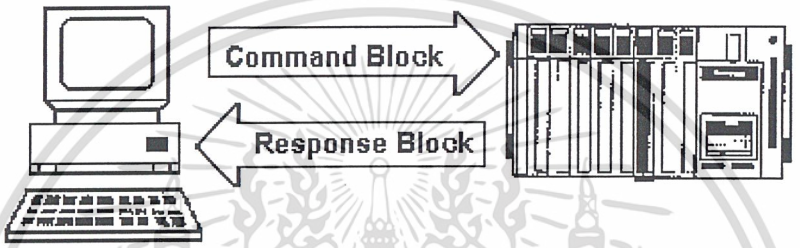


รูปที่3-20 แสดงระบบโครงข่ายแบบหลายจุดโดยใช้มาตรฐานการสื่อสาร RS 485

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ข้อตกลงในการสื่อสารของเครื่องควบคุม PLC/PC

รูปแบบของข้อตกลงในการสื่อสารของเครื่องควบคุม PLC/PC โดยทั่วไปจะเป็นลักษณะการถามตอบกันระหว่างเครื่องควบคุมกับอุปกรณ์ภายนอก ซึ่งอุปกรณ์ภายนอกมักจะเป็นฝ่ายถามก่อนโดยการส่งบล็อกคำสั่ง (Command Block) ออกไป จากนั้นเครื่องควบคุมจะทำการตรวจสอบแล้วส่งบล็อกตอบสนองกลับมา (Response Block) ตัวอย่างการสื่อสารข้อมูลกันระหว่างเครื่องคอมพิวเตอร์กับเครื่องควบคุมPLC/PC



รูปที่ 3-21 แสดงการติดต่อระหว่างคอมพิวเตอร์กับเครื่องควบคุม PLC/PC

3.3.1 รูปแบบของบล็อก (Block Format)

ลักษณะของบล็อกของเครื่องควบคุมPLC/PC แต่ละผู้ผลิตจะแตกต่างกันไปแต่จะมีพื้นฐานเดียวกันดังตัวอย่างของบล็อกคำสั่งต่อไปนี้

รูปที่ 3-22 แสดงรูปแบบของบล็อก

1. หมายเลขเครื่อง ในการเชื่อมต่อที่เป็นโครงข่ายแบบหลายจุดนั้นเครื่องควบที่เชื่อมต่ออยู่ในระบบจะมีมากกว่า 1 เครื่อง การกำหนดว่าต้องการส่งข้อมูลให้กับเครื่องควบคุมตัวใด เช่น @05 คือ เครื่องหมายเลขที่ 05

2. ส่วน HEADER เป็นส่วนของคำสั่งหลักที่จะกำหนดว่าต้องการกระทำกับข้อมูลส่วนใด เช่น ต้องการอ่านข้อมูลของ อินพุท ต้องการเขียนข้อมูลให้กับตัวนับ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ส่วน TEXT เป็นส่วนของข้อมูล เช่น ค่าที่อ่านได้จากอินพุท หรือ ค่าที่จะต้องเขียนลงในพื้นที่ต่าง

4. ส่วนของ FCS เป็นส่วนของการควบคุมความผิดพลาดของข้อมูลซึ่งได้จากการคำนวณ

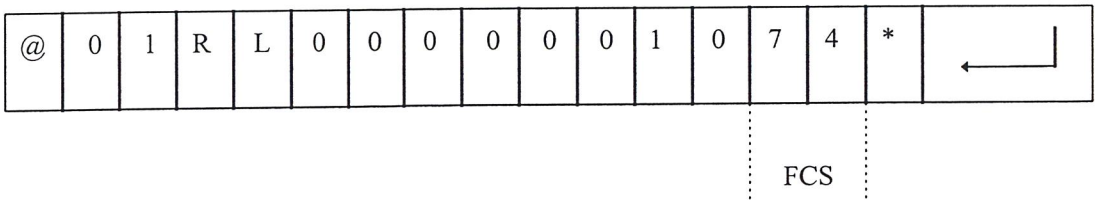
5. ส่วนของ TERMINAL เป็นส่วนที่ปิดท้ายบอกให้ทราบว่าจบบล็อก และมักจะติดตามด้วยรหัสของ Carrier Return (CR)

3.3.2 การคำนวณ FCS

FCS หรือ Frame Check Sequence เป็นสิ่งที่ใช้ในการตรวจสอบความผิดพลาดของการสื่อสาร เป็นการเปลี่ยนข้อมูล 8 บิต เป็น 2 ตัวอักษรข้อมูลASCII แล้วนำมาทำการ EXCLUSIVE - OR (XOR) โดยเริ่มจาก @ จนถึงตัวอักษรตัวสุดท้ายของ TEXT การคำนวณFCS สามารถจะใช้วิธีอื่นนอกจากจะใช้วิธี XOR ดังที่ได้กล่าวไว้แล้วในหัวข้อที่ 3.1.3 เรื่อง การควบคุมความผิดพลาด

อักขระ	รหัส(ASCII BINARY)	(HEX)	XOR (HEX)
@	0100 0000	[40]	
0	0011 0000	[30]	112 [70]
1	0011 0001	[31]	65 [41]
R	0101 0010	[52]	19 [13]
L	0100 1100	[4C]	95 [5F]
0	0011 0000	[30]	111 [6F]
0	0011 0000	[30]	95[5F]
0	0011 0000	[30]	111[6F]
0	0011 0000	[30]	95 [5F]
0	0011 0000	[30]	111 [6F]
0	0011 0000	[30]	95 [5F]
1	0011 0001	[31]	110 [6E]
0	0011 0000	[30]	94 [5E]
*	0010 1010	[2A]	116 [74]

จากการคำนวณที่ได้จะสร้างเป็นบล็อกได้ดังนี้



รูปที่ 3-23 แสดงรูปแบบของบล็อกและการคำนวณ FCS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบทนี้แสดงให้เห็นถึงรูปแบบในการสื่อสารข้อมูลของคอมพิวเตอร์ เปรียบเทียบกับการติดต่อสื่อสารของเครื่องควบคุม PLC/PC ซึ่งจะเป็นแนวทางในการพัฒนาการติดต่อสื่อสารข้อมูลของเครื่องควบคุม PLC/PC เพื่อทำให้เครื่องควบคุม PLC/PC สามารถควบคุมเครื่องจักรในระบบที่มีขนาดใหญ่ให้สามารถทำงานได้อย่างสัมพันธ์กัน ซึ่งจะทำให้การควบคุมมีประสิทธิภาพมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบและพัฒนาเครื่องควบคุมแบบตรรกที่โปรแกรมได้

บทนี้จะกล่าวถึงลักษณะของโครงสร้างแต่ละส่วนของเครื่องควบคุมแบบตรรกที่โปรแกรมได้ที่ได้ออกแบบขึ้นมา เครื่องควบคุมแบบตรรกที่โปรแกรมได้เป็นระบบคอมพิวเตอร์ขนาดเล็กชนิดหนึ่ง ที่ได้ออกแบบให้ใช้เพื่องานควบคุมโดยเฉพาะที่ ประกอบด้วยส่วนประมวลผล ส่วนหน่วยความจำ และทางเข้าออกของข้อมูล หรือ อินพุทเอาท์พุท ซึ่งอุปกรณ์ต่างๆ จะต้องมีความเชื่อถือได้สูง เนื่องจากส่วนประมวลผลทั่วไปของคอมพิวเตอร์ มีลักษณะการทำงานเป็นแบบตรรก สามารถที่จะเลื่อนข้อมูล หมุนวนข้อมูล ในลักษณะต่างๆ ได้ทุกรูปแบบตามแต่ผู้ใช้งานจะโปรแกรมให้มีการทำงานในลักษณะใด ทั้งนี้จะต้องขึ้นอยู่กับความสามารถของระบบปฏิบัติการที่ออกแบบด้วย ซึ่งจะได้กล่าวต่อไป และด้วยความสามารถประมวลผลดังกล่าวจึงเป็นที่มาในการออกแบบเครื่องควบคุม แบบตรรกที่โปรแกรมได้

4.1 คุณสมบัติของเครื่องควบคุมที่โปรแกรมได้

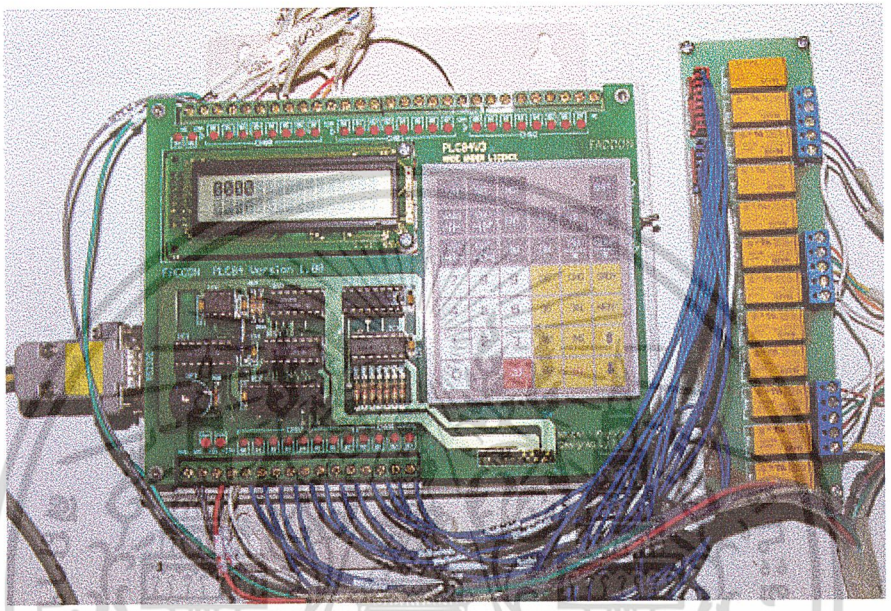
โดยจะขอกล่าวอย่างกว้างๆ ก่อนคือ เมื่อต้องออกแบบระบบควบคุมแบบตรรกที่โปรแกรมได้ สิ่งแรกจะต้องพิจารณาถึงคือ ความสามารถของ หน่วยประมวลผล และอุปกรณ์ประกอบร่วม เช่น พอร์ท หน่วยสื่อสารข้อมูล หน่วยแสดงผล และรับข้อมูลจากผู้ใช้ (HMI: Human Machine Interface) ว่าสถาปัตยกรรมของไมโครโปรเซสเซอร์ใดจะเหมาะสมที่สุด เมื่อพิจารณาสถาปัตยกรรมของไมโครโปรเซสเซอร์ Z84C11 (ZILOG) ผู้เขียนและผู้ทำวิจัยเห็นว่ามีความเหมาะสมมากที่สุดทั้งระบบฮาร์ทแวร์ และซอร์ฟแวร์ จึงได้นำมาทำการทดลองใช้เป็นหน่วยประมวลผลของเครื่องควบคุมแบบตรรกที่โปรแกรมได้ ซึ่งจะกล่าวถึงรายละเอียดดังนี้

4.1.1 ทางด้านฮาร์แวร์ เป็นไมโครโปรเซสเวอร์ที่ประมวลผลคราวละ 8 บิท ทำงานที่ความเร็วสูงโดยเฉลี่ย 1.5 ล้านคำสั่งต่อวินาที ที่ความถี่ 12 MHz ซึ่งเป็นความเร็วที่น่าพอใจสำหรับการควบคุมแบบเวลาจริง มีวงจรพอร์ทเข้าออกของข้อมูล 40 บิทมีวงจรสร้างฐานเวลาติดตั้งร่วมด้วยซึ่งเป็นชิพขนาด 100 ขา การติดตั้งส่วนต่างๆ ดังกล่าวไว้ภายในเป็นข้อดี คือผลจากสัญญาณรบกวนน้อยมาก การออกแบบระบบทั้งหมดนั้นจะไม่ยุ่งยาก และที่สำคัญคือจะมีขนาดเล็ก

4.1.2 ทางด้านซอร์ฟแวร์ ชุดคำสั่งของไมโครโปรเซสเซอร์ ไม่ว่าจะเป็นชุดคำสั่งแบบตรรก เช่น AND OR COMPLEMENT XOR ชุดคำสั่งคำนวณเชิงเลข เช่น ADD SUB MUL DIV ชุดคำสั่งเคลื่อนย้ายข้อมูลทั้งแบบ 8 บิท และ 16 บิท สามารถนำมาใช้งานในการทำโปรแกรมได้อย่างง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อชุดคำสั่งพร้อมและเอื้ออำนวยในการทำโปรแกรมแล้ว เป็นผลให้การจัดการกับหน่วยความจำ การสื่อสารข้อมูลกับอุปกรณ์ประกอบร่วม และ การออกแบบระบบปฏิบัติการเป็นไปได้โดยสะดวก



รูปที่ 4-1 เครื่องควบคุมแบบตรรกที่โปรแกรมได้ที่ออกแบบขึ้นมา

4.2 การออกแบบวงจร

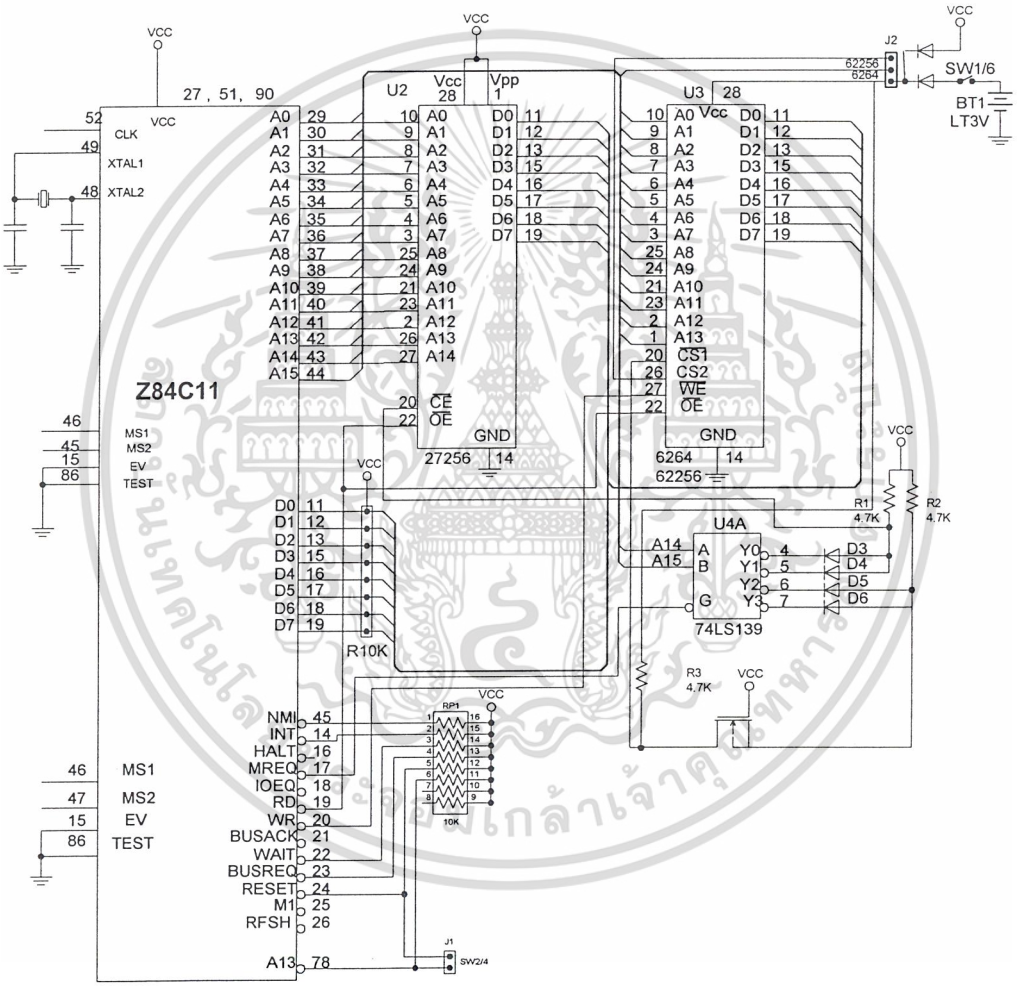
วงจรที่ออกแบบถูกแบ่งออกเป็นส่วน ๆ ได้หลายส่วน ได้แก่

4.2.1 วงจรหน่วยประมวลผลกลาง ใช้ชิพไมโครโปรเซสเซอร์ Z84C11 ที่มีสายสัญญาณกำหนดตำแหน่ง (Address Bus) โดยแต่ละสายสัญญาณจะมีชื่อเรียกเฉพาะคือ A0 - A15 จำนวน 16 บิทหรือสายสัญญาณ ซึ่งสามารถกำหนดตำแหน่งของหน่วยความจำได้ 65535 ตำแหน่ง และสายสัญญาณข้อมูล (Data Bus) จำนวน 8 บิท นอกจากนี้จะมี สายสัญญาณสำหรับควบคุมจังหวะการทำงาน ได้แก่ RD WR IORQ MERQ INT NMI BUSRQ BUSAK ซึ่งจะใช้กำกับจังหวะการอ่านเขียนในหน่วยความจำหรืออุปกรณ์ประกอบร่วมที่เป็นวงจรรวมภายนอก การเชื่อมต่อสัญญาณต่างๆ ในวงจรที่ได้ออกแบบได้ปฏิบัติตามไดอะแกรมเวลาของผู้ผลิตอย่างเคร่งครัด ทำให้การผลทำงานเป็นที่น่าพอใจ

4.2.2 หน่วยความจำ ถูกแบ่งออกเป็น 2 ส่วน ได้แก่ หน่วยความจำสำหรับระบบปฏิบัติการ หน่วยความจำสำหรับโปรแกรมผู้ใช้ อย่างละ 32 Kbytes รวมทั้งหมด 64 Kbytes ใช้การถอดรหัสจากสัญญาณตำแหน่ง (Address Bus) เพื่อเลือกโดย IC74LS139 ซึ่งเป็น IC ถอดรหัส 2 to 4

เอกสารนี้มีลิขสิทธิ์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Decoder โดย A14 และ A15 เป็นสัญญาณกำหนดตำแหน่งเพื่อถอดรหัสแบ่งหน่วยความจำออกเป็น 4 ส่วน ส่วนละ 16 Kbytes แต่หน่วยความจำที่ใช้นั้นมีขนาดความจุเป็น 32 Kbytes ต่อชิพ จึงจำเป็นต้องรวมสัญญาณที่ ถอดรหัสได้แล้วนั้นด้วย Diode ดังวงจรที่น่าสังเกต ในส่วนที่เป็นหน่วยความจำแบบอ่าน/เขียนได้จะมีแบตเตอรี่เพื่อสำรองกระแสไฟฟ้าไว้เพื่อรักษาสถานะของข้อมูลในกรณีปิดการทำงาน



รูปที่ 4-2 แสดงวงจรหน่วยประมวลผลประกอบกับหน่วยความจำ

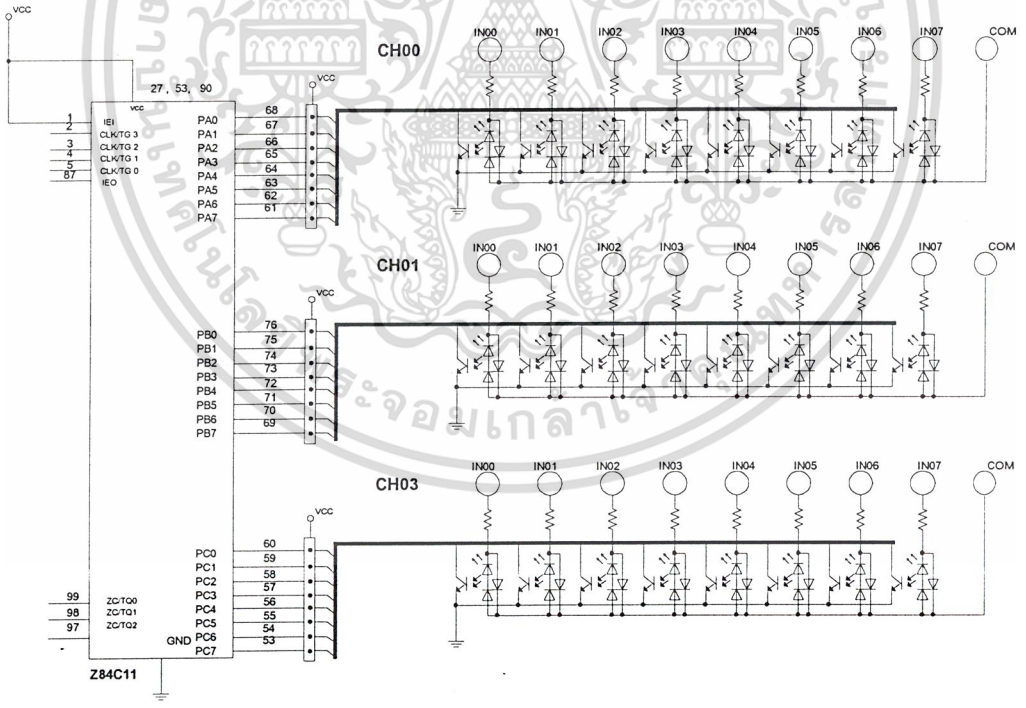
4.2.3 หน่วยอินพุทและเอาท์พุท

มีหน้าที่เชื่อมต่อกับอุปกรณ์ภายนอก ทั้งการรับค่าสภาวะจากภายนอก เช่นการเปิดปิดของสวิทช์ต่าง ๆ ระดับของเหลว อุณหภูมิ ความดัน หรือส่งค่าสภาวะออกไปยังอุปกรณ์ควบคุม เช่น รีเลย์ มอเตอร์ไฟฟ้า ปั๊ม วาล์ว เป็นต้น จากที่ได้กล่าวแล้วว่าภายในชิพไมโครโปรเซสเซอร์ได้รวม

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ทไว้สำหรับเป็นประตูทางเข้าออกข้อมูลสำหรับการประมวลผล ซึ่งมีถึง 40 บิท หรือ 5 พอร์ท ซึ่งจะมีชื่อเรียกในแต่ละพอร์ท คือ พอร์ท A B C D และ E สำหรับวงจรที่ได้ออกแบบนั้นจะใช้พอร์ท A B C เป็นอินพุทพอร์ท เป็นจำนวน 24 บิท ส่วนพอร์ท D และ E เป็นเอาท์พุทพอร์ท จำนวน 16 บิท อย่างไรก็ตามสำหรับเอาท์พุทพอร์ทนั้นจะถูกนำมาใช้สำหรับส่งข้อมูลออกภายนอกเพียง 12 บิท ส่วนอีก 4 บิทถูกนำมาใช้กับการแสดงสถานะในการทำงานอันได้แก่ สถานะของการปฏิบัติการหรือผิดพลาด (RUN/ERROR) สถานะของความพร้อมระบบ (READY) กำเนิดสัญญาณกดคีย์ของผู้ใช้งาน (KEY BEEP) วงจรสำหรับหน่วยอินพุทและเอาท์พุทจะได้แสดงดังรูปที่ 4-4

4.2.3.1 หน่วยอินพุท ทำหน้าที่รับค่าสภาวะจากอุปกรณ์ภายนอกที่มีการเปลี่ยนแปลงสภาวะเป็นแบบลอจิกออกแบบโดยใช้ Opto Isolator ซึ่งเป็นวงจรเชื่อมต่อทางแสง เพื่อหลีกเลี่ยงแรงดันไฟฟ้าที่สูงเกินของอุปกรณ์ตรวจจับจากภายนอกหรือผิดปกติ อันอาจจะเข้ามาทำอันตรายแก่วงจรภายใน หมายถึงระบบไมโครโปรเซสเซอร์

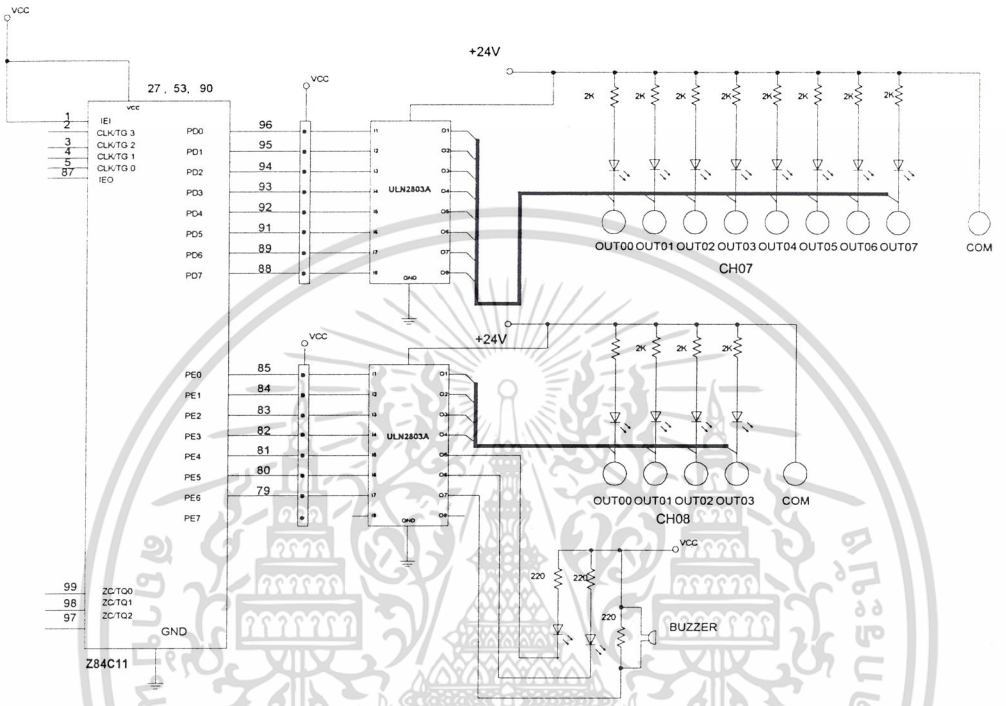


รูปที่ 4-3 แสดงวงจรหน่วยอินพุท

4.2.3.2 หน่วยเอาท์พุท ทำหน้าที่นำข้อมูลจากหน่วยประมวลผลกลางส่งออกไปยังอุปกรณ์ควบคุมซึ่งเชื่อมต่ออยู่ให้มีสภาวะเป็นไปตามโปรแกรมผู้ใช้ ออกแบบโดยวงจรทรานซิสเตอร์อาเรย์ ULN2803 สามารถ สวิทซ์แรงดัน 24 โวลท์ กระแสตรงที่ 100 มิลลิแอมป์ ใน

เอกสารนี้ถูกจัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ผู้ใช้นำไปใช้ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ต้องการนำไปขับรีเลย์ขนาดใหญ่จะต้องผ่านรีเลย์แบบกระแสตรงที่แรงดัน 24 โวลท์ช่วงหนึ่งก่อน

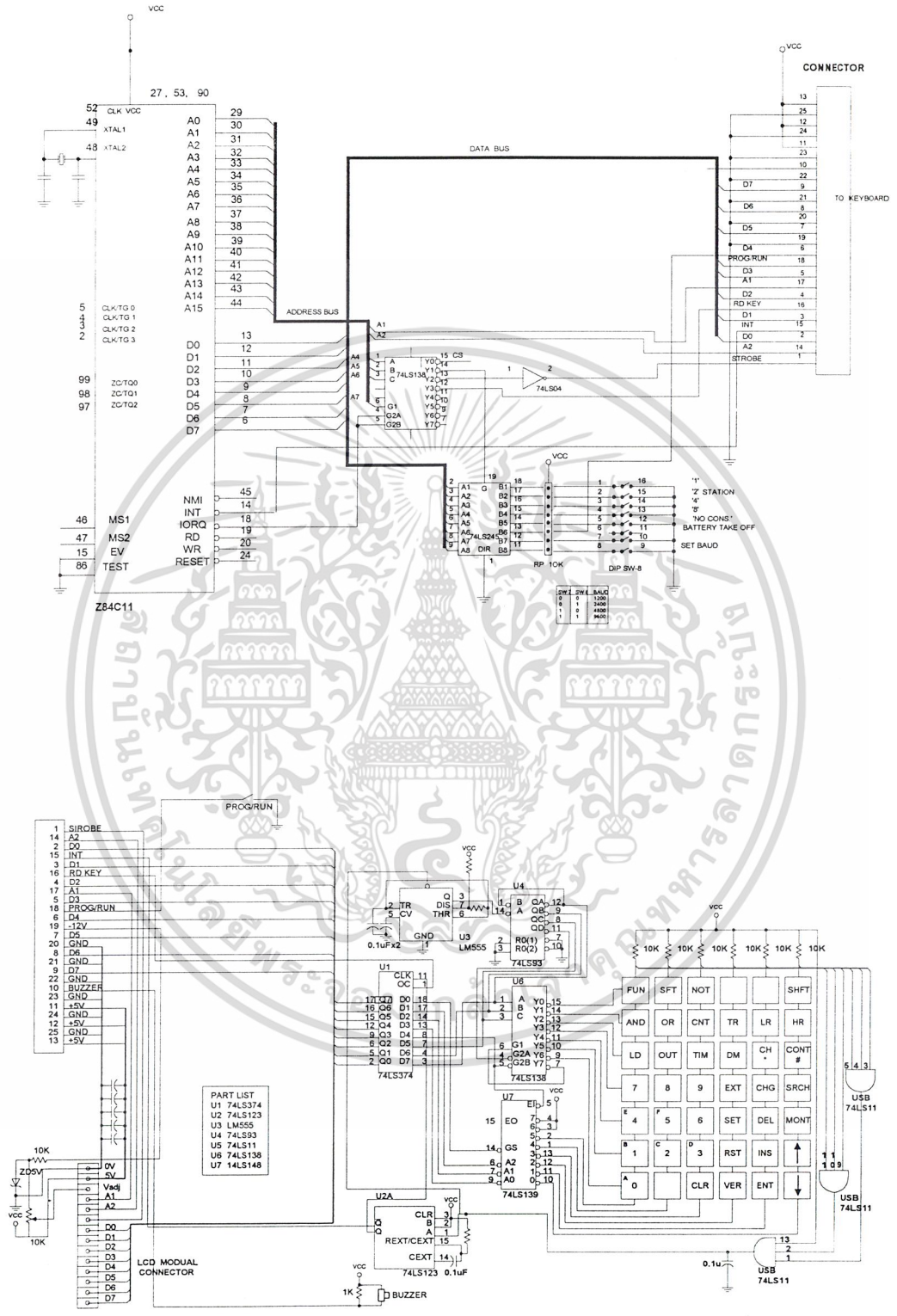


รูปที่ 4-4 แสดงวงจรหน่วยเอาท์พุท

4.2.4 หน่วยแสดงผลและรับข้อมูลกับผู้ใช้

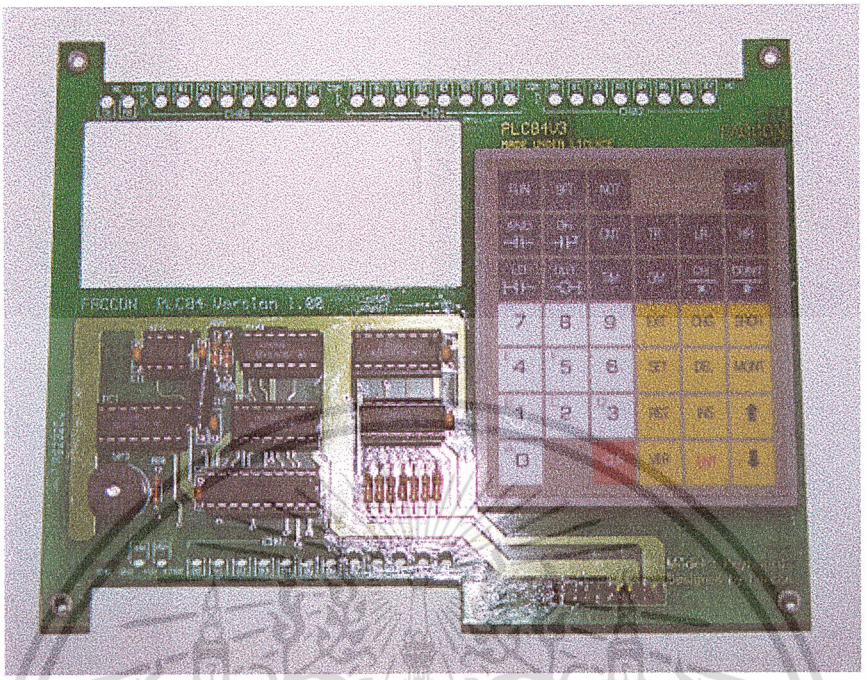
หน่วยแสดงผลและรับข้อมูลกับผู้ใช้ หรือ HMI : Human Machine Interface ทำหน้าที่แสดงผลและป้อนคำสั่งโปรแกรมผู้ใช้ แบบโต้ตอบผู้ใช้กับเครื่องควบคุม โดยส่วนแสดงผลจะใช้โมดูล LCD : Liquid Crystal Display ขนาด 16 อักขระ 2 บรรทัด และวงจรคีย์แบบเมตริกขนาด 42 คีย์ สำหรับวงจรโมดูล LCD เป็นวงจรรวมขนาดใหญ่มาก โดยนำมาต่อร่วมกับหน่วยประมวลผลกลาง ส่งผ่านข้อมูลแบบ ASCII กับรหัสควบคุมอีกเล็กน้อยก็สามารถควบคุมการแสดงผลได้ตามต้องการและ ส่วนวงจรคีย์ 42 คีย์สามารถอธิบายการทำงานโดยเริ่มจาก IC 555 เป็นวงจรกำเนิดสัญญาณนาฬิกาเพื่อแจกแจงให้วงจรนับ IC 74LS93 ทำการสร้างสัญญาณรหัสฐานสองให้กับ IC 74LS138 เพื่อถอดรหัสเป็นสัญญาณสแกนคีย์ที่เมตริกทาง Column ทั้ง 6 ส่วนวงจรตรวจจับการกดคีย์จะใช้ IC 74LS11 รหัสของคีย์ถูกเข้ารหัสโดย IC74LS148 ส่งให้กับ IC 74LS374 พร้อมกับสัญญาณสแกน เมื่อตรวจจับการกดคีย์ได้แล้วก็ จะส่งสัญญาณให้กับ IC 74LS123 เพื่อสร้างสัญญาณบอกแก่หน่วยประมวลผลกลางว่ามีการกดคีย์ เพื่อจะได้กลับมารับรหัสของคีย์ที่ถูกกด และจากนั้นนำไปคำนวณหา รหัสคีย์ในการใช้งานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

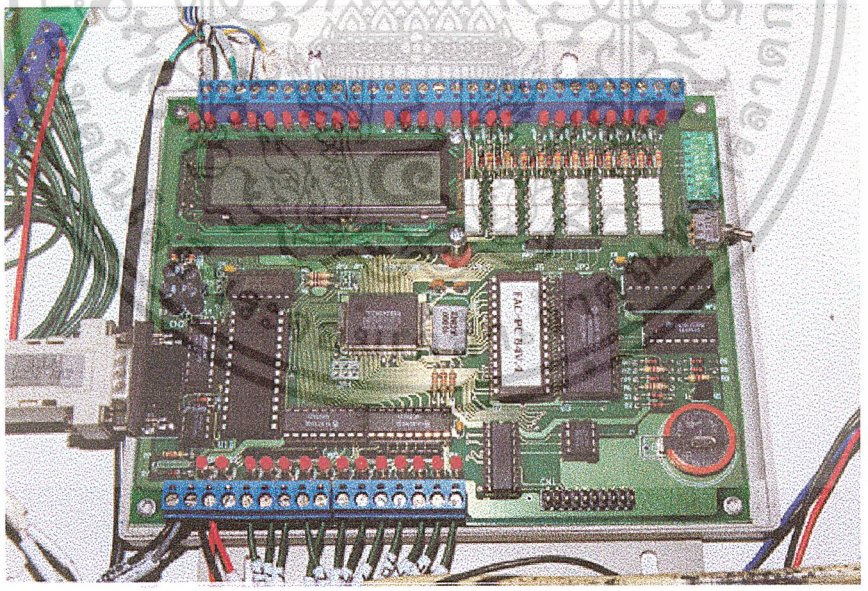


รูปที่ 4-5 แสดงวงจรหน่วยแสดงผลและรับข้อมูลของผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-6 แสดงชุดคีย์ป้อนโปรแกรมสำหรับผู้ใช้



รูปที่4-7 แสดงให้ภายในของเครื่องควบคุมPLC/PC

4.2.5 หน่วยสร้างฐานเวลา

หน่วยสร้างฐานเวลาทำหน้าที่สร้างฐานเวลาให้กับเครื่องควบคุมเนื่องจากฟังก์ชั่นการควบคุมบางฟังก์ชั่นจะนำคาบเวลามาทำการประมวลผลร่วมด้วย การสร้างฐานเวลากระทำโดยวงจร CTC ที่อยู่ภายในชิพไมโครโปรเซสเซอร์โดยภายในจะมีวงจรนับที่โปรแกรมได้ขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8 บิท จำนวน 4 ชุด สามารถนับความถี่ได้สูงถึง 4 MHz เมื่อนำความถี่มาหารตามค่าที่โปรแกรมไว้ จะได้คาบเวลาที่เหมาะสมเพื่อนำมาใช้เป็นฐานเวลา ฐานเวลาสำหรับเครื่องควบคุมมีหลายคาบเวลาได้แก่ 10 ,100 , 200 มิลลิวินาที

4.3 ระบบปฏิบัติการ

ระบบปฏิบัติการสำหรับเครื่องควบคุมหมายถึงระบบที่ใช้ควบคุมการทำงานทั้งหมด มีหน้าที่รับคำสั่งโดยการคีย์จากผู้ใช้ในลักษณะโต้ตอบผ่านทาง HMI ทำการติดต่อสื่อสารข้อมูลกับ Host Computer ทำการแปลคำสั่งบูลีนของผู้ใช้เป็นรหัสในการปฏิบัติ และทำงานตามฟังก์ชั่นควบคุม โดยจะแยกอธิบายเป็นหัวข้อได้ดังนี้

4.3.1 การแบ่งพื้นที่หน่วยความจำ

4.3.2 โปรแกรมผู้ใช้และคำสั่งสำหรับเครื่องควบคุม

4.3.3 การจัดข้อมลของโปรแกรมผู้ใช้

4.3.4 การแปลชุดคำสั่งจากโปรแกรมผู้ใช้เป็นภาษาเครื่อง

4.3.5 การทำงานของฟังก์ชั่นควบคุม

4.3.1 การแบ่งพื้นที่หน่วยความจำ

หน่วยความจำของระบบปฏิบัติการมีจำนวน 64 Kbytes นั้นถูกแบ่งออกเป็นส่วนๆ ดังนี้

32 Kbytes R only	โปรแกรมบริหารงานระบบ
	โปรแกรมควบคุม
24 Kbytes R/W	โปรแกรมผู้ใช้
8 Kbytes R/W	ตารางข้อมูล และ รีจีสเตอร์

ตารางที่ 4-1 แสดงการแบ่งพื้นที่ใช้งานของหน่วยความจำ

4.3.1.1 โปรแกรมบริหารงานระบบ ถูกใช้เป็นโปรแกรมจัดวางระบบข้อมูลโปรแกรมผู้ใช้ ทำการแปลโปรแกรมผู้ใช้เป็นรหัสปฏิบัติ ติดต่อสื่อสารกับ Host Computer และสื่อสารกับหน่วย HMI พื้นที่โปรแกรมส่วนนี้ผู้ใช้ไม่สามารถเปลี่ยนแปลงแก้ไขข้อมูลในหน่วยความจำได้

4.3.1.2 โปรแกรมควบคุม เป็นพื้นที่หน่วยความจำของโปรแกรมในการประมวลผลฟังก์ชั่นควบคุมที่ผู้ใช้ทำการโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.3 ตารางข้อมูล เป็นพื้นที่หน่วยความจำที่ผู้ใช้สามารถเปลี่ยนแปลงแก้ไขได้ ภายในตารางข้อมูล จะถูกเก็บข้อมูลที่มีลักษณะ “1” หรือ “0” ซึ่งใช้แทนสภาวะ “ON” หรือ “OFF” ทางไฟฟ้า ของข้อมูลอินพุท ข้อมูลเอาท์พุท ข้อมูลของรีเลย์ภายใน ข้อมูลรีเลย์ค้างสภาวะ (Holding Relay) ข้อมูลแต่ละพื้นที่ที่ได้ออกแบบไว้มีดังนี้

ตารางอินพุท-เอาท์พุทและตารางรีเลย์ภายในถูกกำหนดขึ้นเป็นจำนวน 30 ตำแหน่งโดยเริ่มจาก 00-29 แบ่งออกดังนี้

ตำแหน่ง	หน้าที่
00 - 06	สำหรับเก็บข้อมูลอินพุทจากภายนอกตำแหน่งละ 56 บิท
07 - 09	สำหรับใช้เป็นข้อมูลของเอาท์พุทที่จะต้องถูกส่งออกไปสู่ภายนอก ตำแหน่งละ 8 บิท รวม 24 บิท
10 - 27	สำหรับใช้เป็นรีเลย์ภายในหรือรีจีสเตอร์สำหรับช่วยในการทำงาน ตำแหน่งละ 12 บิท
28 - 29	สำหรับเป็นรีเลย์พิเศษซึ่งในแต่ละบิทมีคุณลักษณะดังนี้ ตำแหน่งละ 8 บิท

2800	เป็นบิทกำเนิดความถี่คาบเวลา 0.01 วินาที
2801	เป็นบิทกำเนิดความถี่คาบเวลา 0.1 วินาที
2802	เป็นบิทความถี่คาบเวลา 0.2 วินาที
2803	เป็นบิทกำเนิดความถี่คาบเวลา 0.5 วินาที
2804	“ON”
2805	“ON”
2806	“OFF”
2807	“OFF”
2900	เป็นบิทสำหรับควบคุมเอาท์พุททั้งหมด “ON” = Disable output “OFF” = Enable output
2901	สถานะ “ON” One scantime
HR00-HR99	รีเลย์ คงค่าสถานะจำนวน 100 ตำแหน่ง รวม 800 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-2 แสดงตารางอินพุทเอาท์พุทและ รีเลย์ภายใน

4.3.2 โปรแกรมผู้ใช้และคำสั่งสำหรับเครื่องควบคุม

คำสั่งที่ใช้กันโดยทั่วไปของเครื่องควบคุมเป็นคำสั่งบูลีนที่มีการสื่อความหมายทางตรรกที่ตรงไปตรงมากับผู้ใช้ได้เป็นอย่างดีเช่นสมการบูลีนต่อไปนี้ สามารถแสดงด้วยคำสั่งบูลีนได้คือ

$$X = \left[(A \bullet B) + \overline{(C \bullet D)} \right] \bullet F$$

ลำดับที่	คำสั่ง	อินพุท/เอาท์พุท	ความหมาย
0000	LD	A	นำค่าสภาวะ A
0001	AND	B	กระทำการ AND กับ B
0002	LD	NOTC	นำค่าสภาวะตรงข้าม C
0003	AND	D	กระทำการ AND กับ D
0004	ORLD		กระทำการ OR ทั้งสองวงเล็บ
0005	AND	F	กระทำการ AND กับ F
0006	OUT	X	ส่งผลลัพธ์ออกที่ X
0007	END		จบสิ้นคำสั่ง (กลับไปทำใหม่)

ตารางที่ 4-3 แสดงความสัมพันธ์ของสมการบูลีนกับชุดคำสั่งบูลีน

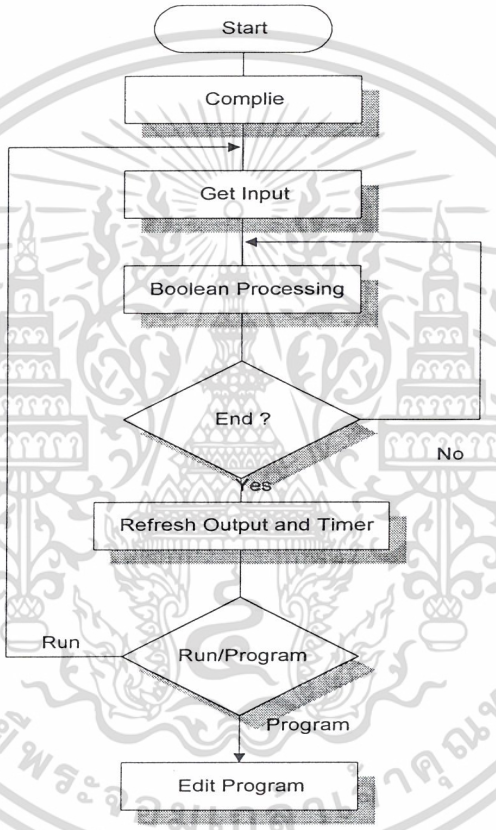
คำสั่ง END เป็นสิ่งที่บ่งบอกว่า จะต้องกระโดดไปกระทำลำดับที่เริ่มต้นใหม่และกระทำเรื่อยไปจนกว่าจะมีคำสั่งให้หยุดทำงาน ลักษณะการทำงานที่วนรอบสามารถที่จะรับค่าสภาวะที่มีการเปลี่ยนแปลงในตัวแปรอินพุทได้เกือบจะต่อเนื่องตราบเท่าที่เวลาในการวนรอบ (scan time) จะเอื้ออำนวย เพื่อที่จะนำมาประมวลผลและให้ผลลัพธ์ออกไปควบคุม เวลาในการวนรอบดังกล่าวขึ้นอยู่กับประสิทธิภาพของหน่วยประมวลผลและการทำงานของระบบจัดการซึ่งการทำงานในแต่ละวงรอบจะแสดงได้ดังผังต่อไปนี้

สาระสำคัญของวงรอบการทำงานพอสรุปได้คือ

1. รับค่าสภาวะจากภายนอกทางอินพุท
2. วางลงในหน่วยความจำ
3. ปฏิบัติการในคำสั่งบูลีน
3. ส่งผลลัพธ์ในหน่วยความจำ
4. ส่งผลลัพธ์ออกทางเอาท์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำส่วนนี้เก็บโปรแกรมควบคุมที่ผู้ใช้เขียนขึ้นโดยใช้หน่วยป้อนโปรแกรมซึ่งถูกเปลี่ยนให้เป็นรหัสของชุดคำสั่งนั้น ๆ ข้อมูลในหน่วยความจำของโปรแกรมผู้ใช้จะเป็นรหัสของชุดคำสั่งซึ่งเป็นคำสั่งทางลอจิก คำสั่งการเคลื่อนย้ายข้อมูล ควบคู่กับตำแหน่งอินพุท/เอาท์พุท จากนั้นโปรแกรมบริหารระบบจะทำการแปลความหมายรหัสเหล่านี้ ให้เป็นชุดคำสั่งของไมโครโปรเซสเซอร์ที่ใช้เป็นหน่วยประมาลผลกลาง



รูปที่ 4-8 แสดงผังภาพการทำงานในวงรอบ

4.3.3 การจัดข้อมูลของโปรแกรมผู้ใช้

ข้อมูลของโปรแกรมผู้ใช้จะมีลักษณะเป็นชุดคำสั่งแต่ละชุดคำสั่งถูกจัดเป็นเรคคอร์ดซึ่งภายในประกอบด้วยฟิลด์ 2 ฟิลด์แต่ละเรคคอร์ดมี ขนาด 3 ไบท์โดยมีโครงสร้างข้อมูลดังนี้

Record

Code:Blte;

Data : Word; 2 bytes

End.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Record NO	Byte	word
RECORD 1	CODE	DATA
RECORD 2	CODE	DATA
RECORD END	CODE	DATA

ตารางที่ 4 -4 แสดงการเก็บข้อมูลของโปรแกรมผู้ใช้ในหน่วยความจำ

เรคคอร์ดของชุดคำสั่งจะต้องทำการเก็บรหัสชุดคำสั่งและ ข้อมูลตำแหน่งตารางข้อมูล ส่วนฟิลด์CODE ใช้กำหนดรหัสของคำสั่งที่ผู้ใช้ทำการโปรแกรมขึ้นเพื่อให้เครื่องควบคุมทำงานตามฟังก์ชั่นการควบคุม คำสั่งที่ได้สร้างขึ้นมามีทั้งหมด 40 คำสั่งดังรายละเอียดต่อไปนี้

รหัส	คำสั่ง	ความหมาย
80	LD	นำค่าสภาวะจากตารางข้อมูลมาเป็นผลลัพธ์
81	LD NOT	นำค่าสภาวะตรงข้างจากตารางข้อมูลมาเป็นผลลัพธ์
82	LD HR	นำค่าสภาวะตรงข้ามจากตารางข้อมูล HR มาเป็นผลลัพธ์
83	LD NOT HR	นำค่าสภาวะตรงข้ามจากตารางข้อมูล HR มาเป็นผลลัพธ์
84	AND	กระทำลอจิก AND กับสภาวะที่นำจากตารางข้อมูล
85	AND NOT	กระทำลอจิก AND กับสภาวะตรงข้ามที่นำจากตารางข้อมูล
86	AND HR	กระทำลอจิก AND กับสภาวะที่นำจากตารางข้อมูล HR
87	AND NOT HR	กระทำลอจิก AND กับสภาวะตรงข้ามที่นำจากตารางข้อมูล HR
88	OR	กระทำลอจิก OR กับสภาวะที่นำจากตารางข้อมูล
89	OR NOT	กระทำลอจิก OR กับสภาวะตรงข้ามที่นำจากตารางข้อมูล
8A	OR HR	กระทำลอจิก OR กับสภาวะที่นำจากตารางข้อมูล HR
8B	OR NOT HR	กระทำลอจิกORกับสภาวะตรงข้ามที่นำจากตารางข้อมูล HR
8C	LATCH	คงค่าสภาวะเมื่อมีการ SET และค้างจนกว่าจะ RESET
8D	OUT	นำผลลัพธ์ทางลอจิกและเก็บไว้ในตารางข้อมูล
8E	OUT NOT	นำผลลัพธ์ทางลอจิกและทำตรงข้ามเก็บไว้ในตารางข้อมูล
91	TRANSUP	คงค่าสภาวะหนึ่งรอบการทำงานเมื่อรับสัญญาณขึ้น
92	TRANSDW	คงค่าสภาวะหนึ่งรอบการทำงานเมื่อรับสัญญาณขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัส	คำสั่ง	ความหมาย
95	AND TIM	กระทำลอจิก AND กับสภาวะที่นำจากตัวตั้งเวลา
96	AND CNT	กระทำลอจิก AND กับสภาวะที่นำจากตัวนับ
97	AND NOT TIM	กระทำลอจิก AND กับสภาวะตรงข้ามที่นำจากตัวตั้งเวลา
98	ANDNOT CNT	กระทำลอจิก AND กับสภาวะตรงข้ามที่นำจากตัวนับ
99	OR TIM	กระทำลอจิก OR กับสภาวะที่นำจากตัวตั้งเวลา
9A	OR CNT	กระทำลอจิก OR กับสภาวะที่นำจากตัวนับ
9B	OR NOT CNT	กระทำลอจิก OR กับสภาวะตรงข้ามที่นำจากตัวตั้งเวลา
9C	OR NOT CNT	กระทำลอจิก OR กับสภาวะตรงข้ามที่นับจากตัวนับ
9D	LD TIM	นำค่าสภาวะจากตัวตั้งเวลามาเป็นผลลัพธ์
9E	LD CNT	นำค่าสภาวะจากตัวนับมาเป็นผลลัพธ์
9F	LD NOT TIM	นำค่าสภาวะตรงข้ามจากตัวตั้งเวลามาเป็นผลลัพธ์
A0	LD NOT CNT	นำค่าสภาวะตรงข้ามจากตัวนับมาเป็นผลลัพธ์
A1	TIM	เรียกใช้ตัวตั้งเวลา
A2	CNT	เรียกใช้ตัวนับ
A3	SFT	ทำการเลื่อนข้อมูลในตารางข้อมูลทางขวาหนึ่งบิท
B0	ANDLD	กระทำลอจิก AND กับสภาวะทำนำจาก STACK
B1	ORLD	กระทำลอจิก OR กับสภาวะทำนำจาก STACK
B2	END	สิ้นสุดคำสั่งให้ทำรอบใหม่
B3	NOP	ไม่มีการทำงาน
B4	MCS	เริ่มให้มีการควบคุมหลัก
B5	MCR	สิ้นสุดการควบคุมหลัก
B6	JMP	กระโดดข้ามไปที่คำสั่ง JME
B7	JME	เป็นคำสั่งที่สิ้นสุดการกระโดด

ตารางที่ 4-5 แสดงตารางคำสั่งที่ได้ออกแบบไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้โปรแกรมคำสั่งบูลลีนจะได้รหัสที่นำเก็บไว้ในหน่วยความจำในพื้นที่โปรแกรมผู้ใช้ดังนี้

Address	Instruction	Operand	CODE	WORD	BIT
0000	LD NOT HR	0000	90	00	00
0001	AND	0105	85	01	05
0002	OR NOT	2007	99	20	07
0003	OUT	0700	9E	07	00
0004	END		01	00	00

ตารางที่ 4-6 การจัดเก็บรหัสจากโปรแกรมคำสั่งบูลีน

4.3.4 การแปลชุดคำสั่งจากโปรแกรมผู้ใช้เป็นภาษาเครื่อง

หน่วยความจำตรงพื้นที่โปรแกรมผู้ใช้มีลักษณะของข้อมูลเป็นแบบรหัสชุดคำสั่ง ซึ่งหน่วยประมวลผลไม่สามารถที่จะนำไปปฏิบัติโดยตรง จำเป็นจะต้องแปลเพื่อให้หน่วยประมวลผลสามารถจะรับรู้ถึงคำสั่งที่ผู้ใช้ได้ทำการโปรแกรมไว้ โดยหลักการแปลจะใช้หลักการอ่านข้อมูลที่ละเรคคอร์ดเปรียบเทียบรหัสในฟิลด์ CODE เมื่อเปรียบค่าเท่ากันแล้วทำให้ได้รหัสใหม่ ซึ่งรหัสที่ได้ใหม่นี้จะเป็นตัวชี้ตำแหน่งของโปรแกรมย่อยของคำสั่งนั้นๆ ส่วนตำแหน่งตารางข้อมูลที่อยู่ในเรคคอร์ดเดียวกันจะใช้เป็นพารามิเตอร์ในการส่งผ่านให้กับโปรแกรมย่อยดังกล่าว สำหรับจำนวนพารามิเตอร์ที่จะต้องทำการผ่านค่าไปให้นั้นขึ้นอยู่กับคำสั่งที่ทำการเปรียบเทียบได้ ส่วนการเปรียบเทียบจะสิ้นสุดเมื่อเป็นคำสั่ง ENDภายหลังการแปลเสร็จสิ้น จะได้รหัสใหม่ ซึ่งเป็นชุดคำสั่งของหน่วยประมวลผล ซึ่งรหัสที่ได้ใหม่นี้ ถูกวางอยู่ในตำแหน่งพื้นที่ของโปรแกรมผู้ใช้อีกส่วนหนึ่งดังนี้

ใช้ HL ชี้ตำแหน่ง อินพุท / เอาท์พุท
เรียกโปรแกรมย่อยคำสั่งนั้น CALL SUBn
ใช้ HL ชี้ตำแหน่ง อินพุท / เอาท์พุท
เรียกโปรแกรมย่อยคำสั่งนั้น CALL SUBn

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่ได้กล่าวมาแล้วว่าหน่วยประมวลผลกลางใช้ Z-84C11 เป็นไมโครโปรเซสเซอร์ ดังนั้นการผ่านค่าพารามิเตอร์ให้กับโปรแกรมย่อยจะใช้การผ่านค่าไปกับ รีจีสเตอร์ HL ซึ่งเป็นรีจีสเตอร์ของหน่วยประมวลผลที่ใช้ในการทดลอง ตัวอย่างการสร้างรหัสเพื่อการผ่านค่าพารามิเตอร์ไปกับรีจีสเตอร์และรหัสเพื่อเรียกโปรแกรมย่อยของคำสั่ง ซึ่งรหัสเหล่านั้นได้ถูกแสดงเป็นรหัสภาษาเครื่อง (Assembly) โดยพิจารณาคำสั่งดังต่อไปนี้

คำสั่งบูลีน	แปลแล้วจะได้รหัส Z84C11
LD 0700	คำสั่ง LD ตำแหน่งอินพุท / เอาท์พุท เท่ากับ 0700 รหัสที่ได้ LD HL,0700 และ CALL LD_SUB
AND 0001	คำสั่ง AND ตำแหน่งอินพุท / เอาท์พุท เท่ากับ 0001 รหัสที่ได้ LD HL,0001 และ CALL AND_SUB
OUT 0700	คำสั่ง OUT ตำแหน่งอินพุท / เอาท์พุท เท่ากับ 0700 รหัสที่ได้ LD HL,0700 และ CALL OUT_SUB

ตารางที่ 4- 7 แสดงการแปลชุดคำสั่งเป็นภาษาเครื่อง

4.3.5 การทำงานของฟังก์ชั่นควบคุม

ฟังก์ชั่นควบคุมสำหรับเครื่องควบคุมที่โปรแกรมได้ที่ใช้ในการควบคุมกระบวนการทางอุตสาหกรรมนั้น จะมีลักษณะของการทำงานคล้ายกับการทำงานของหน่วยประมวลผลทั่วไปที่มีคำส่งการควบคุมแบบลอจิกเช่น AND OR ด้วยเหตุนี้เองไมโครโปรเซสเซอร์จึงถูกเลือกเพื่อใช้เป็นหน่วยประมวลผลของเครื่องควบคุมที่โปรแกรมได้ ส่วนโปรแกรมที่สามารถทำให้ไมโครโปรเซสเซอร์มีการทำงานเป็นอุปกรณ์ควบคุม หรือเครื่องควบคุมที่โปรแกรมได้นั้นเป็นเรื่องที่ค่อนข้างที่จะซับซ้อนเนื่องจากการทำงานมีลักษณะการซ้อนกันของโปรแกรมผู้ใช้กับโปรแกรมที่จะให้มีการปฏิบัติการทำงานโดยตรง อีกทั้งฟังก์ชั่นการควบคุมกับคำสั่งที่ใช้ควบคุมการทำงานของไมโครโปรเซสเซอร์ก็มีความคล้ายคลึงกัน ซึ่งต่อไปจะเป็นการอธิบายการทำงานของฟังก์ชั่นควบคุมของเครื่องควบคุมที่โปรแกรมได้ในแต่ละฟังก์ชั่นที่เป็นโครงสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางซอร์ฟแวร์ ใช้ควบคุมไมโครโปรเซสเซอร์แทนผังการทำงาน เพื่อให้ไมโครโปรเซสเซอร์ ทำงานในคำสั่งต่างๆ โปรแกรมที่เขียน สำหรับฟังก์ชั่นควบคุมจะเป็นภาษาเครื่องของ Z-84C11

คำสั่ง LD

```
PROCEDURE LD_SUB (IOASSIGN)
BEGIN
        RESULT := DATATABLE [IOASSIGN];
        IF RESULT > 0 THEN RESULT := SFF ELSE RESULT := S00;
END;
```

คำสั่ง LD NOT

```
PROCEDURE LDNOT_SUB (IOASSIGN)
BEGIN
        LD_SUB (IOASSIGN);
        POP RESULT FROM STACK;
        COMPLEMENT RESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง AND

```
PROCEDURE AND_SUB (IOASSIGN);
BEGIN
        POP TEMPRESULT FROM STACK;
        LD_SUB (IOASSIGN);
        POP RESULT FROM STACK;
        RUSULT := RESULT AND TEMPRESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง AND NOT

```
PROCEDURE ANDNOT_SUB (IOASSIGN);
BEGIN
        POP TEMPRESULT FROM STACK;
        LDNOT_SUB (IOASSIBN);
        POP RESULT FROM STACK;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RESULT := RESULT AND TEMPRESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง OR

```
PROCEDURE OR_SUB (IOASSIGN);
BEGIN
        POP TEMPRESULT FROM STACK;
        LD_SUB (IOASSIGN);
        POP RESULT FROM STACK;
        RESULT := RESULT OR TEMPRESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง OR NOT

```
PROCEDURE ORNOT_SUB (IOASSIGN);
BEGIN
        POP TEMPRESULT FROM STACK;
        LDNOT_SUB (IOASSIGN);
        POP RESULT FROM STACK;
        RESULT := RESULT OR TEMPRESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง ANDLD

```
PROCEDURE ANDLD_SUB (IOASSIGN);
BEGIN
        POP TEMPRESULT FROM STACK;
        POP RESULT FROM STACK;
        RESULT := RESULT AND TEMPRESULT;
        PUSH RESULT TO STACK;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง ORLD

```
PROCEDURE ORLD_SUB (IOASSIGN);
BEGIN
        POP TEMPRESULT FROM STACK;
        POP RESULT FROM STACK;
        RESULT := RESULT OR TEMPRESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง OUT

```
PROCEDURE OUT_SUB (IOASSIGN);
BEGIN
        POP RESULT FROM STACK;
        DATATABLE [IOASSIGN] :=RESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง OUT NOT

```
PROCEDURE OUTNOT_SUB (IOASSIGN);
BEGIN
        POP RESULT FROM STACK;
        DATATABLE [IOASSIGN] := COMPLEMENT RESULT;
        PUSH RESULT TO STACK;
END;
```

คำสั่ง TIM

```
PROCEDURE TIM_SUB (TIMCH);
BEGIN
        POP RESULT FROM STACK;
        IF RESULT = SFF  THEN
        BEGIN
                CHECK  BIT14 OF REGISTERTABLE  [TIMCH];
                IF BIT14 OF REGISTERTABLE [TIMCH] = SET THEN
                        DECHEMENT REGISTERTABLE [TIMCH];
        IF BITO TO BIT 13 OF REGISTERTABLE  [TIMCH]  = 0 THEN
                        SET  BIT15  OF REGISTERTABLE [TIMCH];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            PUSH RESULT TO STACK
        END
        ELSE
        BEGIN
            RESET BIT14 , BIT15;
            REGISTERTABLE [TIMCH] := TIMEREGISTER;
            PUSH RESULT TO STACK;
        END;
END;
```

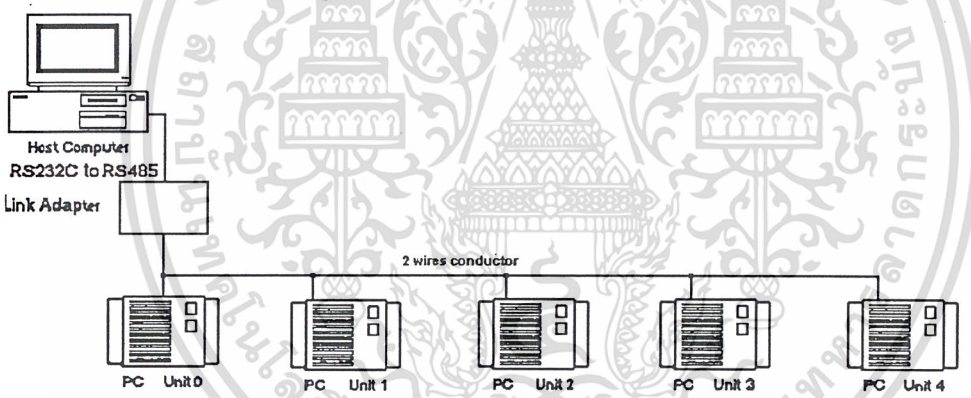
คำสั่ง CNT

```
PROCEDURE CNT_SUB (CNTCH);
BEGIN
        POP RESULT FROM STACK;
        IF RESULT = SFF THEN
        BEGIN
            CHECK BIT14 OF REGISTERTABLE [TIMCH];
            IF BIT14 OF REGISTERTABLE [TIMCH] = SET THEN
                DECHEMENT REGISTERTABLE [TIMCH];
        IF BIT0 TO BIT 13 OF REGISTERTABLE [TIMCH] = 0 THEN
                SET BIT15 OF REGISTERTABLE [TIMCH];
            PUSH RESULT TO STACK
        END
        ELSE
        BEGIN
            RESET BIT14 , BIT15;
            REGISTERTABLE [TIMCH] := TIMEREGISTER;
            PUSH RESULT TO STACK;
        END;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การออกแบบระบบโครงข่ายสำหรับ เครื่องควบคุมแบบตรรกที่โปรแกรมได้

ลักษณะโครงข่ายที่ออกแบบนี้เป็นแบบแยกกระจายโดยออกแบบให้สามารถเชื่อมต่อกันได้ 8 ชุด และขยายได้ถึง 32 ชุดตามข้อจำกัดของมาตรฐาน RS 485 ในระยะทางที่ไกลนับร้อยเมตร โดยแต่ละเครื่องจะมีตำแหน่งระบุประจำเครื่อง การกำหนดกระทำโดยปรับที่ DIP SWITCH ได้ตั้งแต่หมายเลข 00 ถึงหมายเลข 07 ในการติดต่อสื่อสารกับโฮสคอมพิวเตอร์จะกระทำได้โดยโฮสคอมพิวเตอร์จะเป็นตัวกำหนดว่าต้องการติดต่อสื่อสารกับเครื่องควบคุมเครื่องใด ข้อมูลจะถูกส่งกระจายไปยังโครงข่าย ถ้าข้อมูลตรงกับหมายเลขประจำเครื่องของเครื่องควบคุมเครื่องใด เครื่องควบคุมเครื่องนั้นก็จะส่งข้อมูลตอบสนองกับมายังโฮสคอมพิวเตอร์ตามข้อตกลงในการติดต่อสื่อสาร



รูปที่ 5-1 แสดงระบบโครงข่ายผ่านโฮสคอมพิวเตอร์ที่ทำการออกแบบ

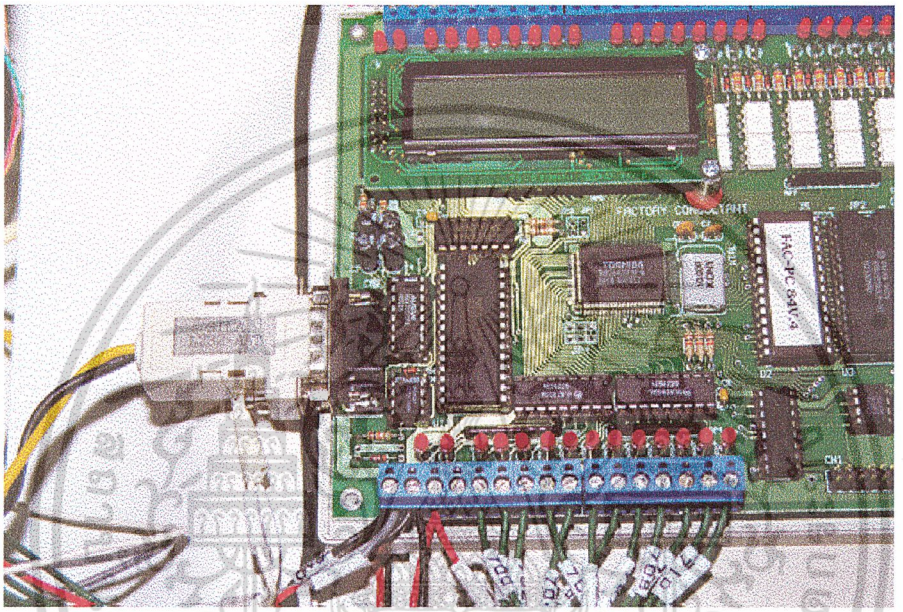
การออกแบบในแต่ละส่วนของระบบโครงข่ายแบ่งออกได้ดังนี้

5.1 การออกแบบทางด้านฮาร์ดแวร์

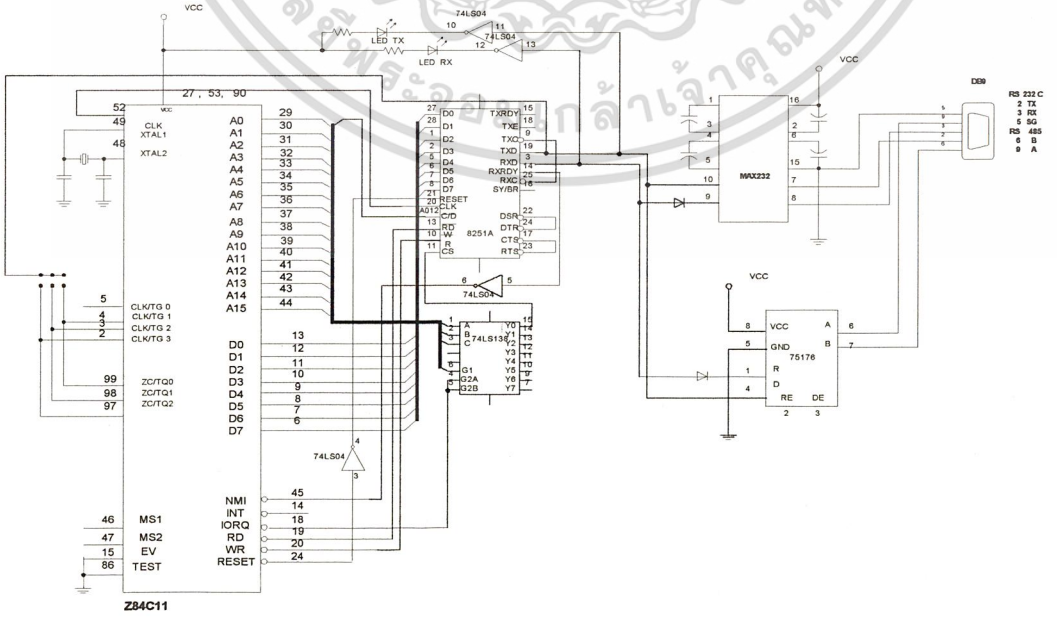
ในระบบฮาร์ดแวร์ของเครื่องควบคุม PLC / PC ถูกออกแบบให้สามารถติดต่อสื่อสารได้ทั้งแบบจุดต่อจุดตามมาตรฐาน RS 232 C และแบบหลายจุดตามมาตรฐาน RS 485 โดยใช้ชิพ INTEL 8251 ซึ่งเป็นวงจรรวมที่ใช้งานเพื่อสื่อสารข้อมูลแบบอนุกรม ได้เชื่อมต่อสายสัญญาณข้อมูลเข้ากับหน่วยประมวลผลกลางและสัญญาณควบคุม โดยหน่วยประมวลผลกลางได้มองเห็นวงจรสื่อสารข้อมูลเป็นพอร์ททางเข้าออกของข้อมูลในรูปแบบอินพุทเอาท์พุท และใช้วงจรขับสัญญาณข้อมูลอนุกรม (Serial Bus) เป็นวงจรรวมเบอร์ MAX232 ให้เป็นไปตามมาตรฐาน RS232C และวงจรรวมเบอร์ 75176 เป็นวงจรขับสัญญาณข้อมูลแบบอนุกรมให้เป็นไปตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐาน RS485 เพื่อใช้ในการสื่อสารข้อมูลในลักษณะจุดต่อจุด หรือ แบบ เครือข่าย ดังแสดงในรูปที่ 5-3 โดยใช้คอนเนคเตอร์ DB9 กำหนดให้ ขา 2 ขา 3 และ ขา 5 เป็นขาของสัญญาณ TX,RX และ SIGNAL GND ของมาตรฐาน RS232C และในกรณีที่ต้องการต่อใช้งานเป็นแบบ RS485 ขา9 และ ขา6 เป็นขาของสัญญาณ A และ B ตามลำดับดังแสดงในรูปที่ 5-3



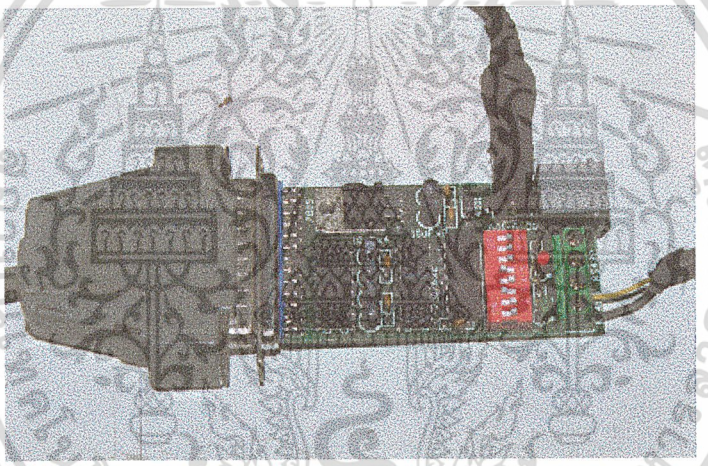
รูปที่ 5-2 แสดงส่วนของการติดต่อสื่อสารอนุกรม



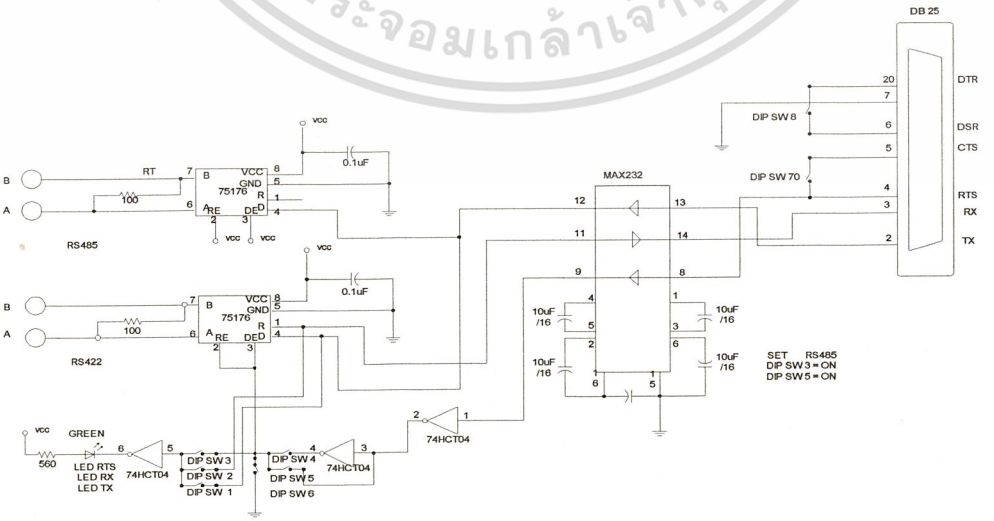
รูปที่ 5-3 วงจรสื่อสารข้อมูลอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากโฮสคอมพิวเตอร์มีการติดต่อสื่อสารอนุกรมแบบ RS232C ดังนั้นจึงจำเป็นจะต้องมีอุปกรณ์แปลงสัญญาณจากระบบมาตรฐาน RS232C เป็นแบบ RS485 ในกรณีที่ทำการเชื่อมต่อเป็นระบบโครงข่ายแบบหลายจุดโดยวงจรแปลงสัญญาณที่เลือกใช้นี้ประกอบด้วยวงจรรวมเบอร์ MAX232 รับสัญญาณ RS232C จากคอมพิวเตอร์และขับสัญญาณอนุกรมให้กับวงจรรวมเบอร์ 75176 ซึ่งเป็นวงจรขับสัญญาณอนุกรมให้เป็นไปตามมาตรฐาน RS485 โดยมีการควบคุมการส่งและการรับข้อมูลจากสัญญาณ RTS (REQUEST TO SEND) โดยในสภาวะปกติแล้วโฮสคอมพิวเตอร์จะอยู่ในสภาวะรอรับข้อมูลเมื่อต้องการส่งข้อมูลก็จะส่งสัญญาณให้ RTS เป็น ENABLE บอกให้อุปกรณ์แปลงสัญญาณเตรียมพร้อมส่งข้อมูลและเมื่อส่งข้อมูลจนครบก็ให้ RTS กลับสู่สภาวะเดิม เพื่อรอรับสัญญาณตอบสนอง



รูปที่ 5-4 อุปกรณ์แปลงสัญญาณ จาก RS232C ให้เป็น RS485



รูปที่ 5-5 วงจรแปลงสัญญาณ จาก RS232C ให้เป็น RS485

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การออกแบบทางด้านซอร์ฟแวร์

การออกแบบทางด้านซอร์ฟแวร์หรือการออกแบบโปรแกรมระบบจัดการข้อตกลงในการติดต่อสื่อสารข้อมูล (PROTOCOLS) ข้อมูลที่ได้รับเข้ามาทาง พอร์ทอนุกรมจะประกอบด้วยบล็อกคำสั่งแต่ละคำสั่ง เช่น การอ่านหรือการเขียนข้อมูลเป็นต้น ข้อมูลจะถูกเปลี่ยนจากอนุกรมให้เป็นแบบขนานโดย ชิฟ INTEL 8251 มาเก็บไว้ยังพื้นที่หน่วยความจำที่เตรียมไว้ (BUFFER) สำหรับบล็อกคำสั่งมีขนาดไม่เกิน256ไบท์ทันทีที่มีการส่งบล็อกคำสั่งเข้ามา CPUจะถูกอินเตอร์รัพให้มาตรวจสอบข้อมูลทีละ 1 ไบท์ โดย 3 ไบท์แรกจะเป็นหมายเลขประจำเครื่อง ถ้าตรงกันก็จะรับข้อมูลเข้ามาเก็บในบัฟเฟอร์ต่อไปแต่ถ้าไม่ตรงกัน บัฟเฟอร์จะถูกลบออก และออกจากโปรแกรมเพื่อรอรับข้อมูลต่อไป ซึ่งถ้าข้อมูลตรงกันตามข้อตกลงในการติดต่อสื่อสารโดยมีการตรวจสอบแล้ว CPUทำการถอดรหัสคำสั่งและประมวลผลตามบล็อกคำสั่ง จากนั้นจะเริ่มส่งบล็อกตอบสนองกลับไป เมื่อได้รับสัญญาณ CARRIER RETURN (CR) เป็นการบอกให้ทราบว่าจบบล็อกคำสั่ง

5.3 ข้อตกลงในการสื่อสารข้อมูลที่ทำการออกแบบ

ชุดของข้อมูลในการสื่อสารจะถูกเรียกว่าบล็อก บล็อกของข้อมูลจะถูกส่งจากเครื่องคอมพิวเตอร์ ไปในระบบการเชื่อมต่อ ซึ่งจะเรียกว่า บล็อกคำสั่ง (COMMAND BLOCK) และบล็อกของข้อมูลที่ถูกส่งจากระบบการเชื่อมต่อไปสู่ HOST จะเรียกว่า บล็อกตอบสนอง(RESPONSE BLOCK) ในระบบการเชื่อมต่อสื่อสารแบบหลายจุด แต่ละ บล็อก ไม่ว่าจะเป็นบล็อกคำสั่งหรือ บล็อกตอบสนองก็ตาม จะเริ่มต้นด้วย อักขระ “@” ตามด้วยตำแหน่งเฉพาะ (UNIT NUMBER) ตามด้วยคำสั่ง (HEADER) ข้อมูล (DATA) และ อักขระ “*” สิ้นสุดด้วยรหัสกำกับบล็อก (FRAME CHECK SEQUENCE CODE : FCS) และรหัสปิดท้ายบล็อกที่เป็นอักขระ [CR]

@	X	X	X	X	DATA	*	X	X	CR
	ตำแหน่ง		คำสั่ง				FCS		ปิดท้าย

รูปที่ 5-6 แสดงรูปแบบของบล็อกตอบสนอง

จำนวนอักขระในแต่ละบล็อกทั้งหมดจะต้องไม่เกิน 128 ตัว และช่วงของการคำนวณเพื่อหารหัสกำกับบล็อกจะอยู่ระหว่างอักขระเริ่มต้น (@)ไปจนถึงสิ้นสุด DATA ที่ เป็น “*”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งและคำตอบสนอง

ในขณะที่ทำการเชื่อมต่อเพื่อสื่อสารข้อมูล HOST สามารถที่จะทำการเฝ้ามองการดำเนินการและสามารถที่จะทำการควบคุมการทำงานของเครื่องควบคุม ถ้าอยู่ในสถานะการเฝ้ามองโฮสคอมพิวเตอร์จะต้องส่งคำสั่งไปถามตามชนิดของข้อมูลที่ต้องการ แต่ละเครื่องควบคุม หรือถ้าในสถานะการควบคุมก็สามารถที่จะส่งคำสั่งไปทำการเปลี่ยนแปลงค่าของข้อมูลที่อยู่ในหน่วยความจำโดยตรง เช่น ข้อมูลของ อินพุท/เอาท์พุท เป็นต้น เวลาของการตอบสนองจะแปรไปขึ้นอยู่กับความเร็วในการส่งผ่านข้อมูล จำนวนของข้อมูล และเวลาในการสแกนของเครื่องควบคุม และถ้าเวลาในการสื่อสารมากขึ้นก็เป็นผลให้เวลาในการสแกนมากขึ้นตามไปด้วย ต่อไปจะเป็นคำสั่งต่าง ๆ ที่เป็นข้อกำหนด ที่ใช้ในการสื่อสาร

1. บล็อกคำสั่งประกาศ

บล็อกคำสั่งประกาศถูกส่งโดยโฮสไปเพื่อตรวจสอบว่ามีเครื่องควบคุมอยู่ในการเชื่อม ต่อหรือในเครือข่ายหรือไม่

รูปแบบคำสั่ง

@	[Unit Number]	AT	*	[FCS]	[CR]

เช่น บล็อกคำสั่ง เป็น “@01AT*7E[CR]” หมายถึง โฮสส่งออกไปเพื่อ ประกาศถามถึงเครื่องควบคุมที่อยู่ในการเชื่อมต่อมีตำแหน่งเฉพาะที่ 01 หรือไม่ ถ้าเครื่อง ควบคุมมีจริงและได้รับคำประกาศ ดังกล่าวก็จะส่งคำตอบสนองออกไป ดังนี้

รูปแบบคำตอบสนอง

@	[Unit Number]	PC84	*	[FCS]	[CR]

เช่น บล็อกตอบสนองเป็น “@01PC84*74[CR]” หมายถึงเครื่องควบคุมตำแหน่ง เฉพาะที่ 01เท่านั้นที่ตอบออกไป ซึ่งจะแสดงให้ HOST ทราบว่าเครื่องควบคุมที่ตำแหน่ง 01 ในระบบการเชื่อมต่อยังคงทำงานปกติ

2. บล็อกคำสั่งอ่านพื้นที่ Input/Output/Internal Relay

รูปแบบคำสั่ง

@	[Unit Number]	RI	[ตำแหน่งเริ่มต้น]	[จำนวนข้อมูล]	*	[FCS]	[CR]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่น ถ้าโฮสต้องการอ่านพื้นที่ของ I/0 หรือ Internal Relay ของเครื่องควบคุมที่วางอยู่ ในระบบการเชื่อมต่อตำแหน่งเฉพาะที่ 02 ตำแหน่ง I/0 ที่ 0010 มาจำนวน 5 ตำแหน่งก็สามารถจัดบล็อกคำสั่ง ได้ดังนี้ "@02RI00100005"XX[CR]" (FCS XX : ASCII 2 digit)

รูปแบบคำตอบสนอง

@	[Unit Number]	RI	[ตำแหน่งเริ่มต้น]	[ข้อมูลXX(1)]	[ข้อมูลXX(..)]	[ข้อมูลXX(n)]	*	[FCS]	[CR]

เช่น ถ้าคำตอบสนองของเครื่องควบคุม PLC/PC ในตำแหน่งเฉพาะ ที่02ตอบมาเป็น "@02RI0010234455566AB*XX[CR]" ก็หมายความว่า ตำแหน่งหน่วยความจำของ I/O หรือ Internal Relay ที่ 0010 มีข้อมูล 23,44,55,66,AB [ฐาน16] ตามลำดับ

3. บล็อกคำสั่งเขียนพื้นที่ Input/Output/Internal Relay

รูปแบบคำสั่ง

@	[Unit Number]	WI	[ตำแหน่งเริ่มต้น]	[ข้อมูลXX(1)]	[ข้อมูลXX(..)]	[ข้อมูลXX(n)]	*	[FCS]	[CR]

เช่นต้องการเขียนข้อมูลลงในตำแหน่ง I/0 หรือ Internal Relay ของเครื่องควบคุมที่ตำแหน่งเฉพาะ 03 ในตำแหน่งที่ 0010 ด้วยข้อมูล 12,34,56,78,9A จำนวน 5 ตำแหน่ง ก็สามารถจัด บล็อกคำสั่ง ได้ดังนี้ "@03WI0010123456789A*XX[CR]"

รูปแบบคำตอบสนอง

@	[Unit Number]	WI	[รหัสตอบสนอง]	*	[FCS]	[CR]

(XX :Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@03WI00*XX[CR]"

4. บล็อกคำสั่งอ่านค่าเป้าหมาย Timer

เพื่อสามารถให้โฮสรู้ค่าเป้าหมาย (Set Value) ที่ได้ตั้งไว้ที่ตัวเวลา

รูปแบบคำสั่ง

@	[Unit Number]	RT	[ตำแหน่งเริ่มต้น]	[จำนวน]	*	[FCS]	[CR]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวตั้งเวลาที่ 12 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด บล็อกคำสั่ง ได้ดังนี้ “@04RT00120001*XX[CR]”

รูปแบบคำตอบสนอง

@	[Unit Number]	RT	[ข้อมูลXX(..)]	[ข้อมูลXX(n)]	*	[FCS]	[CR]

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้ค่าตอบสนองออกมาสมมุติว่าค่าเป้าหมายของตัวตั้งเวลาตำแหน่งที่ 12 มีค่า #0150 และ Block Response เป็น “@04RT00120150*XX[CR] “

5. บล็อกคำสั่งอ่านค่าเป้าหมาย Counter

เพื่อสามารถให้ โฮสรู้ค่าเป้าหมาย (Set Value) ที่ได้ตั้งไว้ที่ตัวตั้งนับ

รูปแบบคำสั่ง

@	[Unit Number]	RC	[ตำแหน่งเริ่มต้น]	[จำนวน]	*	[FCS]	[CR]

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวนับที่10 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด บล็อกคำสั่ง ได้ดังนี้ “@04RT00100001*XX[CR]”

รูปแบบคำตอบสนอง

@	[Unit Number]	RC	[ตำแหน่งเริ่มต้น]	[ข้อมูลXX(..)]	[ข้อมูลXX(n)]	*	[FCS]	[CR]

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวนับตำแหน่งที่ 10 มีค่า #8000 และ บล็อกตอบสนอง เป็น “@04RT0010800*XX[CR]”

6. บล็อกคำสั่งเขียนค่าเป้าหมาย Timer

เพื่อที่โฮสสามารถตั้ง แก้ไขค่าเป้าหมายใหม่ของตัวตั้งเวลาแก่เครื่องควบคุมที่อยู่ในระบบเชื่อมต่อได้ตามต้องการ

รูปแบบคำสั่ง

@	[Unit Number]	WT	[ตำแหน่งเริ่มต้น]	[ข้อมูลXX(..)]	[ข้อมูลXX(n)]	*	[FCS]	[CR]

เช่น ถ้าต้องการตั้งค่าเป้าหมายแก่ตัวตั้งเวลาของเครื่องควบคุมที่วางในระบบเชื่อมต่อที่มีตำแหน่งเฉพาะที่ 05 และตัวตั้งเวลาตำแหน่งที่ 00 ด้วยค่าเวลา #0200 สามารถจัด บล็อกคำสั่งได้ดังนี้ “@05WT0000200*XX[CR]”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบคำตอบสนอง

@	[Unit Number]	WT	[รหัสตอบสนองXX]	*	[FCS]	[CR]

(XX : Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@05WT00*XX[CR]"

7. บล็อกคำสั่งเขียนค่าเป้าหมาย Counter

ลักษณะของข้อกำหนดจะเหมือนกับเขียนค่าเป้าหมายให้กับตัวตั้งเวลา ต่างกันที่ Header

รูปแบบคำสั่ง

@	[Unit Number]	WC	[ตำแหน่งเริ่มต้น]	[ข้อมูลXX(..)]	[ข้อมูลXX(n)]	*	[FCS]	[CR]

เช่นถ้าต้องการตั้งค่าเป้าหมายแก่ตัวนับของเครื่องควบคุมที่วางในระบบเชื่อมต่อที่มีตำแหน่งเฉพาะที่ 05 และตัวตั้งเวลาตำแหน่งที่ 00 ด้วยค่านับ #0200 สามารถจัดบล็อกคำสั่งได้ดังนี้ "@05WC00000200*XX[CR]"

รูปแบบคำตอบสนอง

@	[Unit number]	WC	[รหัสตอบสนองXX]	*	[FCS]	[CR]

(XX : Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@05WC00*XX[CR]"

8. บล็อกคำสั่งอ่านค่าปัจจุบัน Timer

เพื่อที่โฮสสามารถที่จะทำการเฝ้ามองความเป็นไปของค่าเวลาที่กำลังทำงานของตัวตั้งเวลาในเครื่องควบคุมตำแหน่งเฉพาะใด ๆ ที่วางในระบบเชื่อมต่อได้

รูปแบบคำสั่ง

@	[Unit Number]	PT	[ตำแหน่งเริ่มต้น]	[จำนวน]	*	[FCS]	[CR]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวตั้งเวลาที่ 12 และ 13 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด บล็อกคำสั่ง ได้ดังนี้ "@04PT00120002*XX[CR]"

รูปแบบคำตอบสนอง

@	[Unit Number]	PT	[ตำแหน่งเริ่มต้น]	[ข้อมูลXX(..)]	[ข้อมูลXX(n)]	*	[FCS]	[CR]

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวตั้งเวลาตำแหน่งที่ 12 มีค่า #0150 และตัวตั้งเวลาตำแหน่งที่ 13 มีค่า #0200 บล็อกตอบสนอง เป็น"@04RT001201500200*XX[CR]"

9. อ่านค่าปัจจุบัน Counter

รูปแบบคำสั่ง

@	[Unit Number]	PC	[ตำแหน่งเริ่มต้น]	[จำนวนXX]	*	[FCS]	[CR]

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวนับตำแหน่งที่ 10 มีค่า #8000 และ บล็อกตอบสนองเป็น "@04PT00108000*XX[CR]"

10. อ่านจากพื้นที่โปรแกรม Up Load Program

เพื่อให้โฮสสามารถที่จะนำข้อมูลส่วนโปรแกรมบูลีนที่ผู้ใช้ได้ทำการโปรแกรมไว้ในหน่วยความจำนั้นขึ้นมาที่ โฮส เพื่อทำการเก็บรักษาไว้

รูปแบบคำสั่ง

@	[Unit Number]	UL	*	[FCS]	[CR]

เช่นถ้าโฮสต้องการนำโปรแกรมส่วนบูลีนจาก เครื่องควบคุมที่มีตำแหน่งเฉพาะเป็น 01 มาจากจุดเริ่มโปรแกรมถึงคำสั่งสุดท้าย(END) มาสามารถจัด บล็อกคำสั่ง ได้เป็น "@01UL*72[CR]" และเครื่องควบคุมที่ตำแหน่งเฉพาะ 01 จะให้คำตอบสนองออกมาเป็นชุดข้อมูลที่ละ 20 ไบท์ เริ่มที่ตำแหน่งในหน่วยความจำที่ 8000 และชุดต่อไปของข้อมูลจะมีตำแหน่งเริ่มต้นที่ สอดคล้องกับตำแหน่งที่ถูกถ่ายเทขึ้นมาด้วย หรือจนกว่าจะพบคำสั่ง END ดังรูปแบบดังต่อไปนี้ ข้อสังเกตรูปแบบคำตอบสนองนั้นจะมีลักษณะเหมือนกับคำสั่งในการเขียนลงพื้นที่โปรแกรม ทั้งนี้ก็เพื่อที่จะให้สามารถนำชุดข้อมูลดังกล่าวนั้นเขียนลงไปใหม่ที่ส่วนโปรแกรมได้เลยโดยไม่ต้องเปลี่ยนแปลงรูปแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบคำตอบสนอง

@	[Unit Number]	UL	[ตำแหน่งเริ่ม8000]	[ข้อมูลXX(..)]	[ข้อมูลXX(20)]	*	[FCS]	[CR]

11. เขียนลงพื้นที่โปรแกรม Down Load Program

เพื่อให้โฮสสามารถทำการส่งผ่านข้อมูลส่วนโปรแกรมบูลีนเข้าไปไว้ในพื้นที่โปรแกรมของ เครื่องควบคุมที่มีตำแหน่งเฉพาะนั้น ๆ ได้

รูปแบบคำสั่ง

@	[Unit Number]	DL	[ตำแหน่งเริ่ม8000]	[ข้อมูลXX(..)]	[ข้อมูล] (20)]	*	[FCS]	[CR]

ข้อมูลของโปรแกรมบูลีนที่ต้องการเขียนลงในหน่วยความจำส่วนโปรแกรมนั้นก็จะต้องไม่มากกว่า 20 ไบท์ เช่นเดียวกัน แลตำแหน่งที่ต้องการวางลงในหน่วยความจำก็สามารถกำหนดได้ (ตำแหน่งในหน่วยความจำส่วนโปรแกรมบูลีนเริ่มที่ 8000)

รูปแบบคำตอบสนอง

@	[Unit Number]	DL	[รหัสตอบสนองXX]	*	[FCS]	[CR]

(XX : Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น “@01DL00*XX[CR]”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การออกแบบโปรแกรมสนับสนุนการทำงานของเครื่องควบคุม PLC/PC

โปรแกรมสนับสนุนการทำงานของเครื่องควบคุม PLC/PC ที่ออกแบบขึ้นมานี้ ทำหน้าที่จัดการข้อมูลของระบบโครงข่ายบนโฮสคอมพิวเตอร์ โดยอาศัยรูปแบบข้อตกลงในการสื่อสาร (Protocols) ที่ได้ออกแบบขึ้นมา ทำงานภายใต้ระบบจัดการของ Microsoft windows 95 ในบทนี้จะกล่าวถึงการนำข้อตกลงในการสื่อสาร มาประยุกต์ใช้งานร่วมกับโปรแกรมที่ทำงานบนเครื่องคอมพิวเตอร์ ตลอดจนโครงสร้างและวิธีการในการออกแบบโปรแกรมในแต่ละส่วน ดังต่อไปนี้

6.1 คุณสมบัติทั่วไปของโปรแกรม สนับสนุนการทำงาน

1. เป็นโปรแกรมซึ่งใช้ในการเขียนหรือแก้ไขชุดคำสั่งบูลีน (Boolean editor) ตามรูปแบบของเครื่องควบคุม PLC/PC ที่ออกแบบขึ้นมา

2. สามารถนำโปรแกรมชุดคำสั่งบูลีน ที่ทำการเขียนหรือนำมาแก้ไข จัดเก็บลงในแฟ้มข้อมูล (Data file) หรือพิมพ์ออกยังเครื่องพิมพ์

3. สามารถนำโปรแกรมชุดคำสั่งบูลีน ที่ทำการเขียนหรือแก้ไขแล้ว หรือจากเปิดแฟ้มข้อมูล เพื่อ ถ่ายเท (Upload) จากเครื่องคอมพิวเตอร์ ไปยังเครื่องควบคุม PLC/PC ที่ต่ออยู่ในระบบโครงข่ายได้

4. สามารถนำโปรแกรมชุดคำสั่งบูลีน ที่เก็บอยู่ในหน่วยความจำของเครื่องควบคุม PLC/PC ถ่ายเท(Download) มายังเครื่องคอมพิวเตอร์เพื่อทำการแก้ไขหรือจัดเก็บต่อไป

5. เป็นโปรแกรมที่ใช้ในการตรวจสอบสถานะ (Monitoring) การทำงานของพื้นที่หน่วยความจำส่วนต่างๆ เช่น หน่วย อินพุท/เอาท์พุท ตัวตั้งเวลา และ ตัวนับ เป็นต้น นอกจากนี้ยังสามารถสั่งการควบคุม โดยการกำหนดค่า (Force)ให้กับพื้นหน่วยความจำดังกล่าวของเครื่องควบคุม PLC/PC ได้

6. เป็นโปรแกรมที่ใช้ทำการจัดการเชื่อมโยงข้อมูลให้กับเครื่องควบคุม PLC/PC ที่เชื่อมต่ออยู่ในระบบเพื่อให้เครื่องควบคุมที่ต่ออยู่ในระบบแต่ละเครื่อง สามารถสื่อสารข้อมูลถึงกันได้

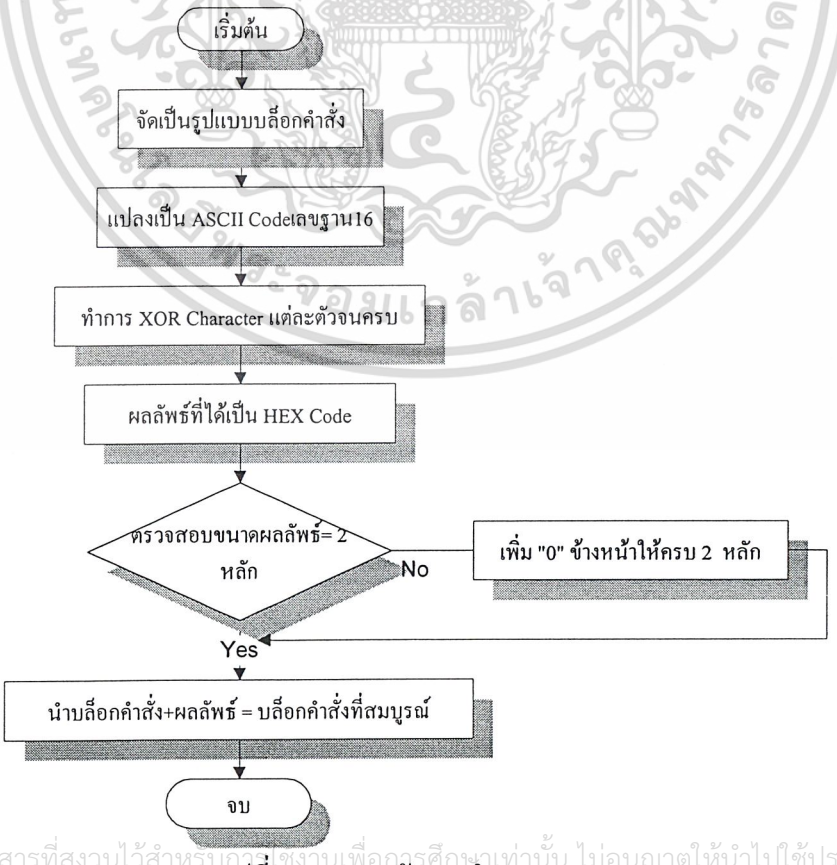
6.2 การออกแบบโปรแกรมเพื่อการตรวจสอบสภาวะการทำงานของเครื่องควบคุม PLC /PC

ในการตรวจสอบสภาวะการทำงาน ของเครื่องควบคุม PLC/ PC ได้แบ่งพื้นที่ออกเป็น 4 ส่วน

1. พื้นที่ของอินพุท/เอาท์พุท (Input / Output)
2. พื้นที่ของรีเลย์ภายใน (Internal Relay)
3. พื้นที่ของตัวตั้งเวลา (Timer)
4. พื้นที่ของตัวนับ (Counter)

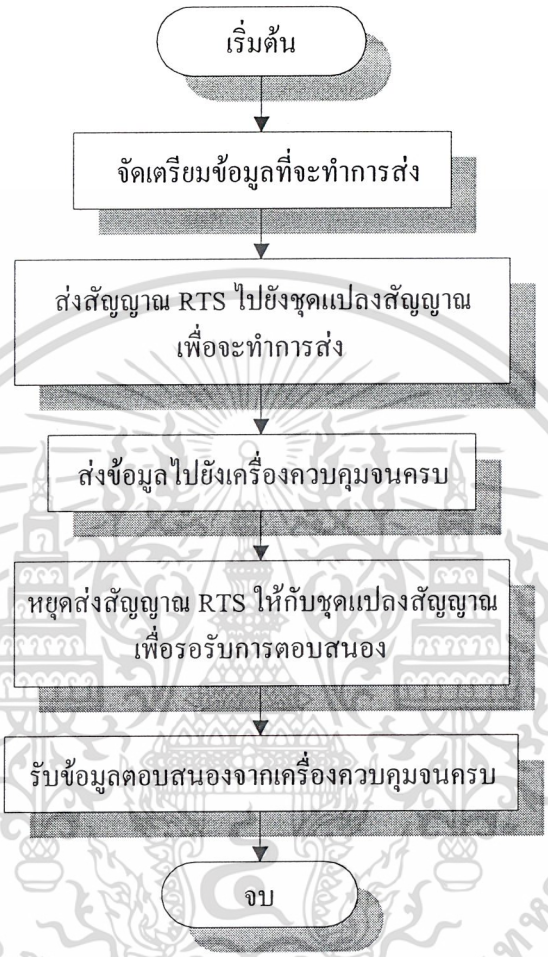
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเขียนโปรแกรมเพื่อที่จะทำการติดต่อสื่อสารกับเครื่องควบคุม PLC/PC จะต้องมีความเข้าใจเกี่ยวกับข้อตกลงในการติดต่อสื่อสารหรือโปรโตคอลของเครื่องควบคุมนั้นๆ เป็นอย่างดี ซึ่งรายละเอียดของโปรโตคอล ได้กล่าวไว้แล้วในบทที่ 5 สำหรับโปรแกรมใน ส่วนที่จะต้องมีการติดต่อสื่อสารนั้น แต่ละโปรแกรมที่ทำการออกแบบต้องมีการควบคุมความผิดพลาดเหมือนกันก็คือ โปรแกรมของการคำนวณ FCS (Frame Check Sequence) และการควบคุมการทำงานของชุดแปลงสัญญาณ (Link Adapter) แบบ RS 232 C เป็น RS 485 โครงสร้างและแนวทางในการออกแบบจะได้แสดงให้เห็นในแผนผังการทำงานรูปที่ 6-1 ในการเขียนโปรแกรมเพื่อติดต่อระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์ภายนอกที่เป็นมาตรฐาน RS 232 C สามารถที่จะทำการรับและส่งข้อมูลผ่านทางพอร์ทได้โดยตรง แต่สำหรับการส่งหรือรับข้อมูลแบบ RS 485 นั้นจะต้องกระทำผ่านชุด Link Adapter เพื่อกำหนดสภาวะว่าจะทำการส่งหรือรับ เนื่องจากมาตรฐาน RS 485 เป็นแบบ ฮาร์ฟดูเพล็ก จึงต้องมีการเขียนโปรแกรมเพื่อควบคุมการทำงานของ Link Adapter แบบ RS 232 C เป็น RS 485 ในการส่งหรือรับข้อมูล สำหรับชุด Link Adapter ที่ใช้นี้จะกำหนดให้ทำการส่งข้อมูลได้โดยให้สัญญาณลอจิกเป็น “1” ที่ ขา RTS (Request to send) ของ พอร์ท RS232C และกำหนดให้ทำการรับโดยการให้สัญญาณลอจิกเป็น “0” สรุปเป็นแผนผังดังรูปที่ 6-2



รูปที่ 6-1 แสดงผังการคำนวณ FCS

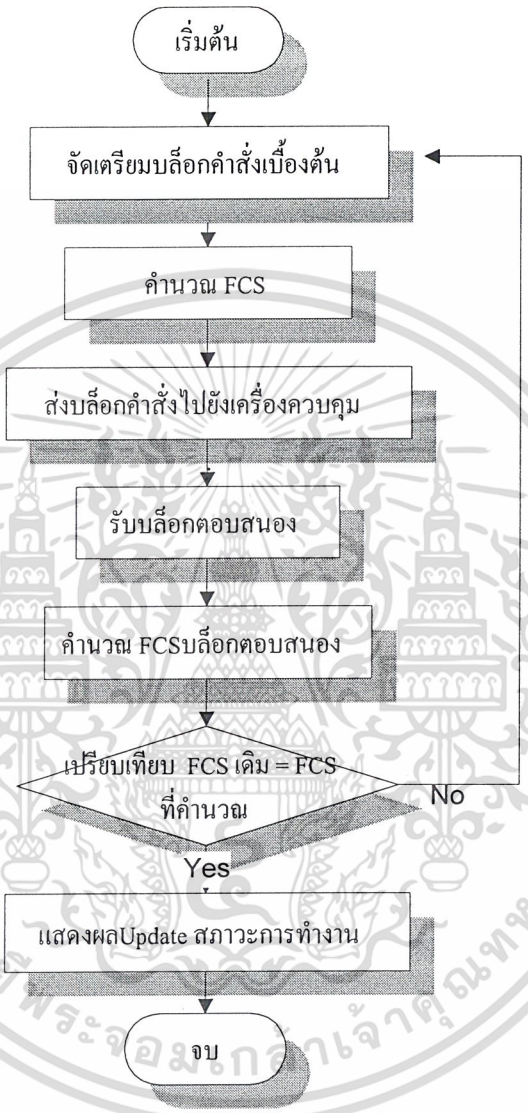
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-2 แสดงแผนผังการควบคุมการทำงานของ Link Adapter แบบ RS 232 C เป็น RS 485

จากรูปแบบข้อตกลงในการติดต่อสื่อสารที่ออกแบบขึ้นมาเพื่อใช้อ่าน ข้อมูลของอินพุท เอาท์พุท หน่วยรีเลย์ภายใน ตัวตั้งเวลา และ ตัวนับ จะได้บล็อกตอบสนองและนำบล็อกตอบสนองมาแสดงผลของสภาวะการทำงานในรูปแบบที่เข้าใจได้ง่ายดังแผนผังการทำงานในรูปที่ 6-3

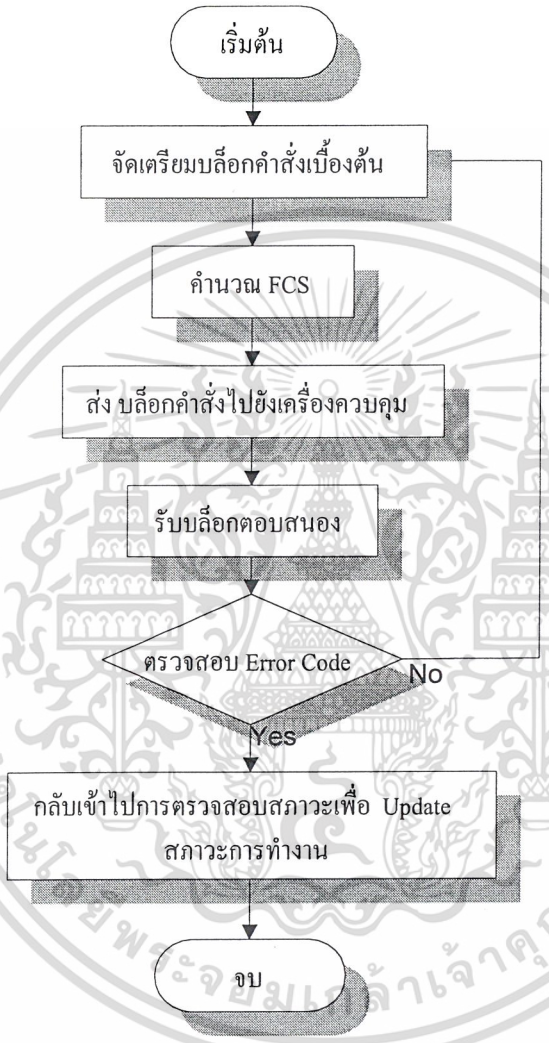
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-3 แสดงแผนผังการตรวจสอบสภาวะการทำงาน

การออกแบบโปรแกรมสำหรับกำหนดค่า (Force) ให้กับพื้นที่เอาท์พุทของเครื่องควบคุม ดังแผนผังการทำงานต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-4 แสดงแผนผังการกำหนดค่า(Force)

ในการแสดงผลของสภาวะการทำงานของหน่วยอินพุทเอาท์พุท และหน่วยรีเลย์ภายใน จะแสดงผลในรูปของ Bit แยกตามแต่ละ Word ในการสั่งการควบคุมโดยการกำหนดค่า (Force) นั้น จะกระทำเป็น Word โดยการระบุ Bit ในแต่ละ word นั้นให้มีสภาวะลอจิกเป็น “1” หรือเป็น “0” ส่วนการแสดงผลของ ตัวตั้งเวลา และตัวนับจะแสดงเป็นตัวเลข โดยมีฐานเวลาอยู่ที่ 0.1 วินาที สำหรับตัวตั้งเวลา แบ่งเป็น 2 แบบคือ Set Value คือค่าที่ตั้งจากการเขียนโปรแกรม และ Present Value คือค่าที่เปลี่ยนแปลงตามการทำงานเมื่อ ตัวตั้งเวลา หรือ ตัวนับนั้นๆ ทำงานอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 การออกแบบโปรแกรมเพื่อการถ่ายเทข้อมูลระหว่างเครื่องควบคุมPLC/PCกับคอมพิวเตอร์

ในการออกแบบโปรแกรมสำหรับการถ่ายเทข้อมูลระหว่างเครื่องควบคุม PLC/PC แบ่งออก เป็น 2 แบบ ด้วยกันคือ

6.3.1 การออกแบบโปรแกรมสำหรับการถ่ายเทข้อมูลระหว่างเครื่องควบคุม PLC/PC ไปยัง เครื่องคอมพิวเตอร์

ในการเขียนโปรแกรมส่วนนี้จะสนใจที่บล็อกตอบสนอง ซึ่งในบล็อกตอบสนองจะประกอบด้วยส่วนสำคัญที่เป็น Code และ Codeนี้จะบอกให้ทราบถึงโปรแกรมชุดคำสั่งบูลีนที่อยู่ภายในหน่วยความจำของเครื่องควบคุม Code จะประกอบด้วย 2 ส่วนด้วยกันคือ Instruction Code และ Data Code ในบทที่ 4 ได้กล่าวถึงInstruction Code ไว้อย่างละเอียดแล้ว ซึ่งหมายถึงรหัสของชุดคำสั่งต่างๆ เช่น LD AND OR เป็นต้น ส่วน Data Code หมายถึง พื้นที่อ้างอิงถึง เช่น อินพุท เอาท์พุท เป็นต้น ซึ่งรายละเอียดในการแปลความหมายสามารถ อธิบายได้จากตัวอย่างต่อไปนี้

ตัวอย่าง การแปลความหมายของบล็อกตอบสนอง

"@00DL800080000088070085000 18D0700B2*48"

เครื่องควบคุมจะทำการส่งบล็อกตอบสนองจนกว่าจะ ถึงรหัสของคำสั่ง "END" นั่นคือรหัส "B2" บล็อกตอบสนองจะทำการตัดตอนเมื่อ เกิน 52 ตัวอักษร โดยเริ่มต้นบรรทัดใหม่ เมื่อเครื่องคอมพิวเตอร์ได้รับบล็อกตอบสนองแล้วจะทำการตัดส่วนที่ไม่เกี่ยวข้องออกจนเหลือเพียง Instruction Code และ Data Code เท่านั้น โดยจัดแบ่งได้ดังนี้

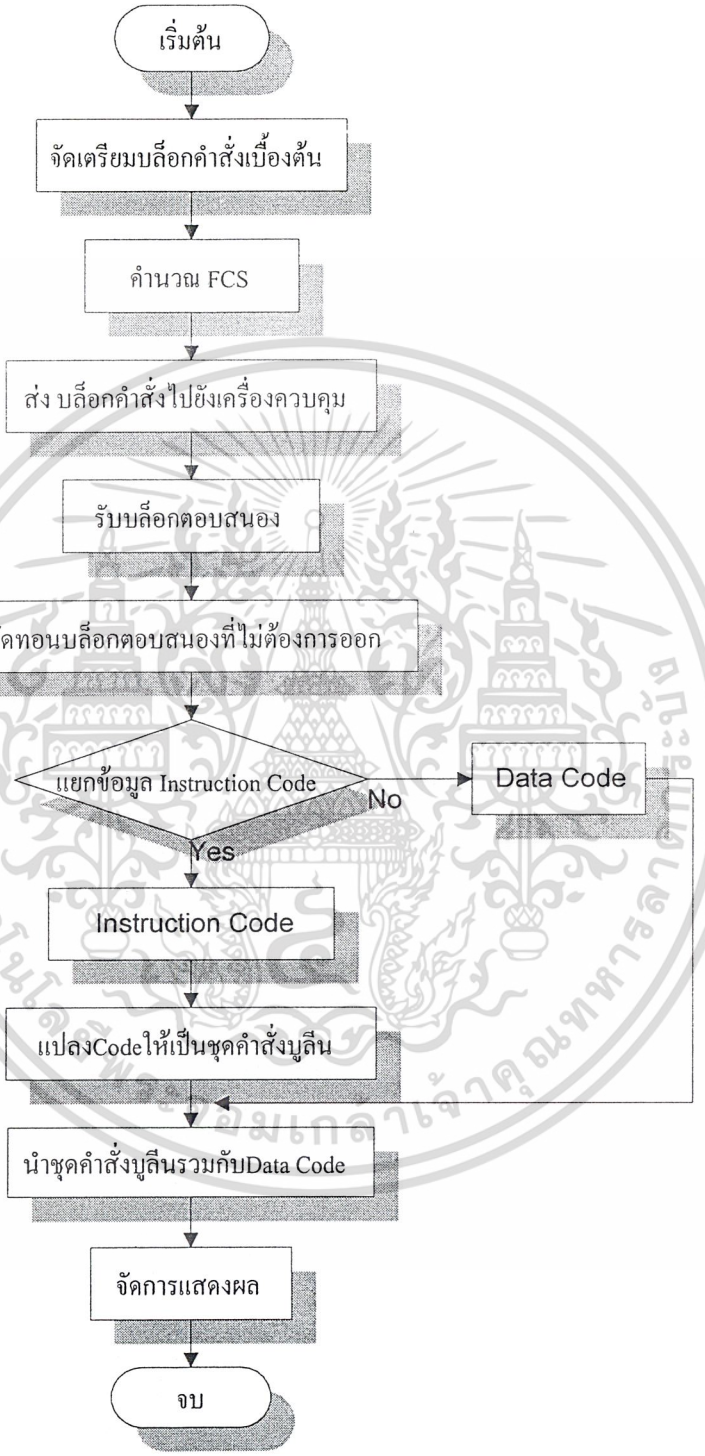
800000
880700
850001
8D0700
B2

Instruction Code จะมีขนาด 2 ตัวอักษร และ Data Code จะมีขนาด 4 ตัวอักษร จากนั้นเข้าสู่การทำงานของโปรแกรมเพื่อแปลความหมาย และผลที่ได้ คือ

```
LD      0000
OR      0700
ANDNOT  0001
OUT     0700
END
```

ขั้นตอนทั้งหมดอธิบายได้จากแผนผังรูปที่6-5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

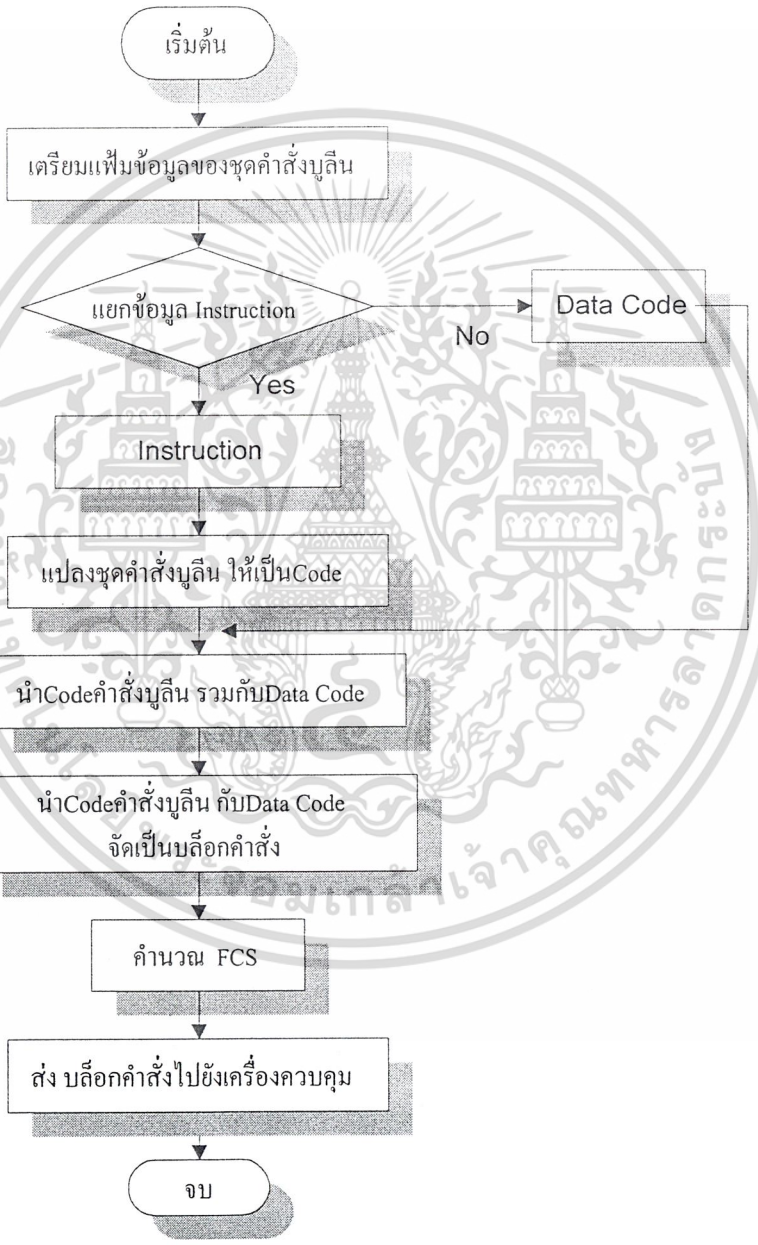


รูปที่6-5 แสดงแผนผังของการถ่ายเทข้อมูลจากเครื่องควบคุม PLC/ PC ไปยังเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.2 การออกแบบโปรแกรม สำหรับการถ่ายเทข้อมูลระหว่างเครื่องคอมพิวเตอร์ไปยังเครื่องควบคุม

ในการออกแบบโปรแกรมส่วนนี้ก็จะนำชุดคำสั่งมาแปลงเป็น Code แล้วจัดเป็นบล็อกคำสั่ง ซึ่งพอจะสรุปเป็นผังการทำงานได้ดังนี้



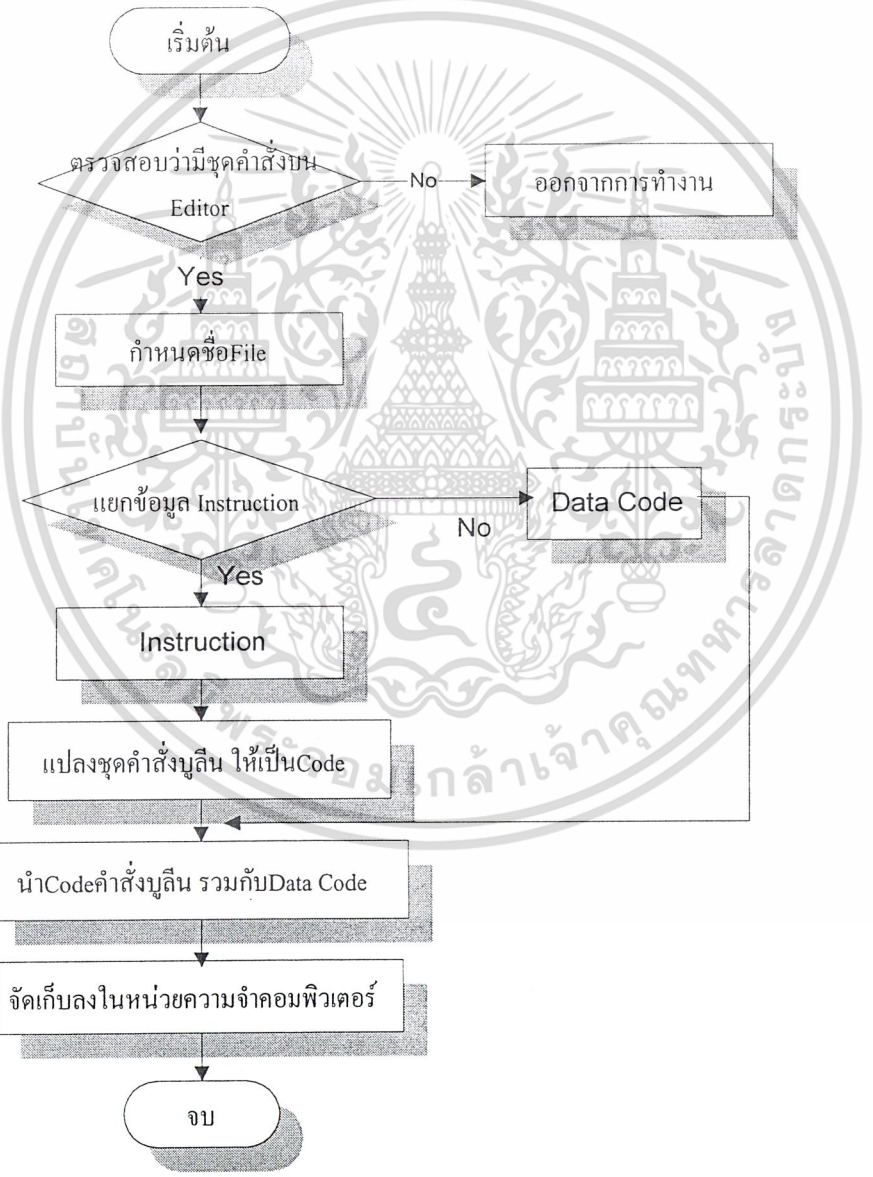
รูปที่6-6 แสดงแผนผังของการถ่ายเทข้อมูลจากเครื่องคอมพิวเตอร์ไปยัง เครื่องควบคุม PLC/PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 การออกแบบโปรแกรมเพื่อการจัดการแฟ้มข้อมูล

ในการออกแบบโปรแกรมส่วนของการจัดเก็บชุดคำสั่งบูลีนลงในแฟ้มข้อมูล จะจัดเก็บเป็นรหัส คำสั่ง(Instruction Code และ Data Code) เพื่อเป็นการประหยัดหน่วยความจำในการจัดเก็บ แบ่งเป็น 3 ส่วนคือ

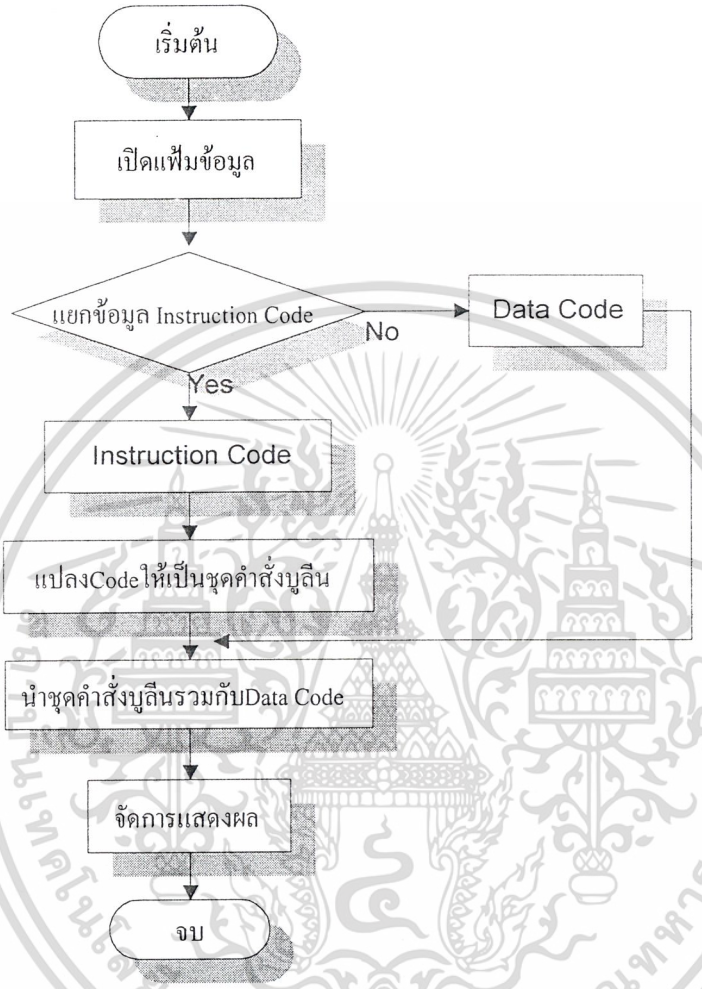
6.4.1 การบันทึกข้อมูล (Save File) สรุปเป็นผังการทำงานได้ดังนี้



รูปที่ 6-7 แสดงแผนผังของการบันทึกข้อมูลลงแฟ้มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

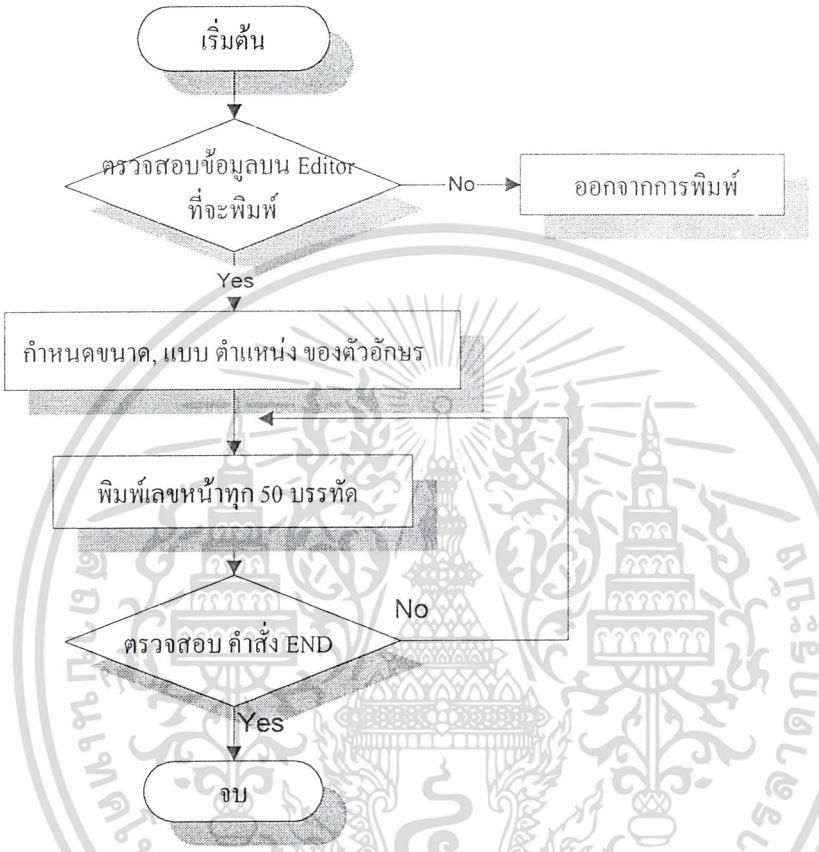
6.4.2 การเปิดแฟ้มข้อมูล (Open File) สรุปเป็นผังการทำงานได้ดังนี้



รูปที่6-8 แสดงแผนผังของการเปิดแฟ้มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4.3 การพิมพ์ข้อมูล (Print File) สรุปเป็นผังการทำงานได้ดังนี้



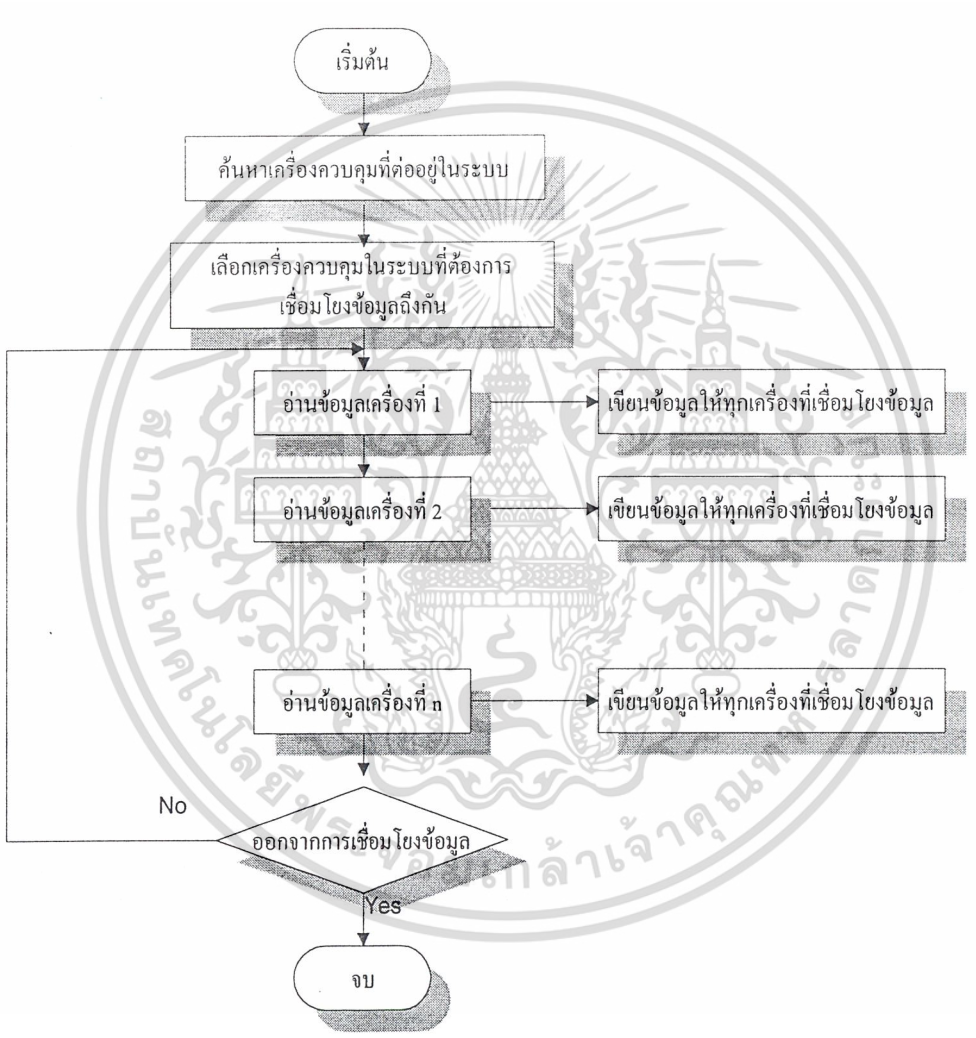
รูปที่ 6-9 แสดงแผนผังของการพิมพ์แฟ้มข้อมูล

6.5 การออกแบบโปรแกรม เพื่อการเชื่อมโยงโครงข่ายผ่านโฮสคอมพิวเตอร์

โดยทั่วไปเครื่องควบคุมPLC/PC แต่ละเครื่องจะที่เชื่อมต่ออยู่ในระบบ จะทำงานแยกอิสระต่อกัน สำหรับโปรแกรมส่วนนี้ จะช่วยทำให้เครื่องควบคุม PLC/PC แต่ละเครื่องที่ต่ออยู่ในระบบสื่อสารกันได้ เพื่อให้การทำงานของเครื่องควบคุม PLC/PC แต่ละเครื่องสัมพันธ์กันตามลักษณะของงานที่ทำการควบคุม โดยมีคอมพิวเตอร์เป็นตัวจัดการในลักษณะที่คล้ายกับพีซีลิ้งค์ โดยมีการแบ่งพื้นที่ในการสื่อสารกัน แตกต่างกันก็คือระบบพีซีลิ้งค์ใช้ตัวฮาร์ดแวร์ของเครื่องควบคุม PLC/PC เป็นตัวส่งข้อมูลถึง เครื่องควบคุมเครื่องอื่นโดยตรง แต่ระบบที่ออกแบบนี้จะใช้โฮสคอมพิวเตอร์เป็นตัวกลางในการกระจายข่าวสาร ให้กับเครื่องควบคุม PLC/PC ที่อยู่ภายในระบบ มีข้อดีคือไม่ต้องทำการปรับปรุงใดๆกับฮาร์ดแวร์ของเครื่องควบคุม และสามารถยกเลิกการเชื่อมโยงจากโฮส คอมพิวเตอร์ได้ทันที เป็นการทำงานในลักษณะการโพล (Polling) ในเบื้องต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ทำการจัดแบ่งพื้นให้กับเครื่องควบคุม PLC/PC เครื่องละ 1 Word โดยใช้พื้นที่ของรีเลย์ค้างสภาวะ HR00 ถึง HR07 โดยแบ่งตามการตั้งหมายเลขประจำเครื่อง (Unit Number) เช่น Unit 00 ก็จะใช้ HR 00 ในการเขียนข้อมูลให้เครื่องอื่นได้รับทราบข้อมูลของตน อย่างไรก็ตามในการทำงานในลักษณะนี้ไม่สามารถที่จะจัดแบ่งพื้นที่ได้มาก ๆ เนื่องมีปัญหาในเรื่องของความเร็วในการส่งข้อมูล ซึ่งสรุปการออกแบบได้ดังผังการทำงานดังต่อไปนี้



รูปที่ 6-10 แสดงแผนผังของการเชื่อมโยงโครงข่ายผ่านโฮสคอมพิวเตอร์

ในบทนี้ได้กล่าวถึงโครงสร้างหลักในการออกแบบโปรแกรม สำหรับตัวโปรแกรม วิธีการใช้งาน ตลอดจนผลการทดลอง จะแสดงรายละเอียดในบทต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

การประยุกต์การใช้งานและผลการทดสอบการทำงาน

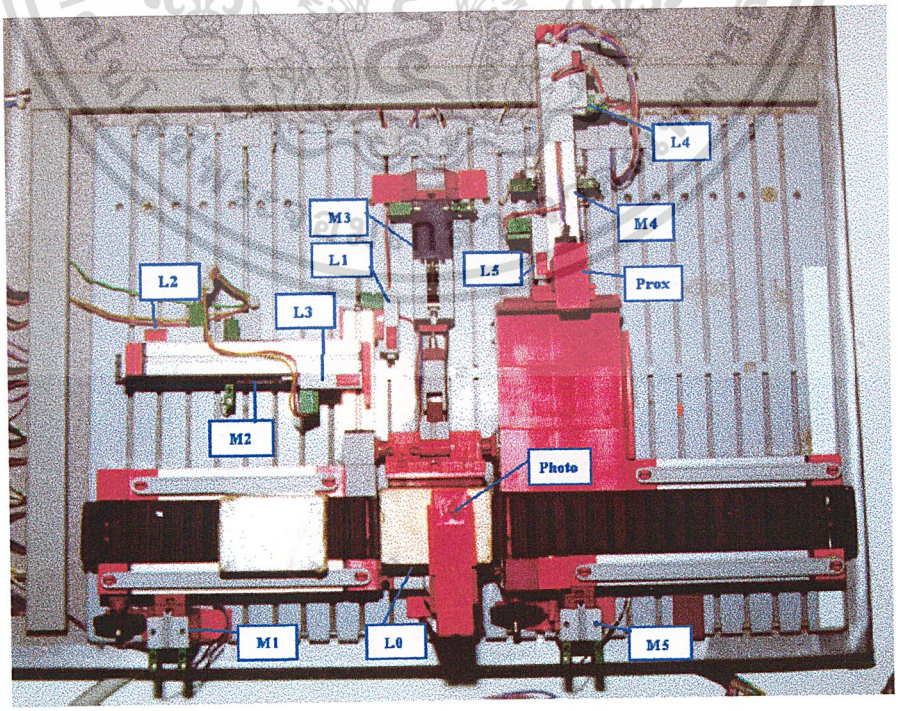
ในบทนี้นำเสนอผลการทดสอบ และ วิธีการการทำงานของเครื่องควบคุม PLC/PC โดยมีชุดจำลองสายพานลำเลียงเป็นตัวอย่างแสดงการควบคุมตลอดจน การทดสอบระบบโครงข่ายผ่านโฮส คอมพิวเตอร์ โดยแสดงให้เห็นการทำงาน แต่ละขั้นตอนของโปรแกรมสนับสนุนการทำงาน เช่นการตรวจ สอบสภาวะการทำงาน การควบคุมผ่านโฮสคอมพิวเตอร์ การจัดการแฟ้มข้อมูล รวมไปจนถึงการเชื่อมต่อทางฮาร์ดแวร์ โดยแบ่งการทดสอบออกเป็นส่วนดังนี้

7.1 การประยุกต์ใช้งานของเครื่องควบคุม PLC/PC

เครื่องควบคุม PLC/PC ที่ออกแบบนี้ ได้นำชุดจำลองสายพานลำเลียง จำนวน 2 ชุด เป็นตัวอย่างในการทดสอบ ซึ่งชุดสายพานจะประกอบด้วย อินพุทที่เป็นสวิทซ์ ลิมิตสวิทซ์ อุปกรณ์ตรวจจับ (Sensor) และเอาท์พุทที่เป็นมอเตอร์กระแสตรงขนาด 24โวลท์ โดยแบ่งการทดสอบเป็น 2 ตัวอย่างด้วยกัน

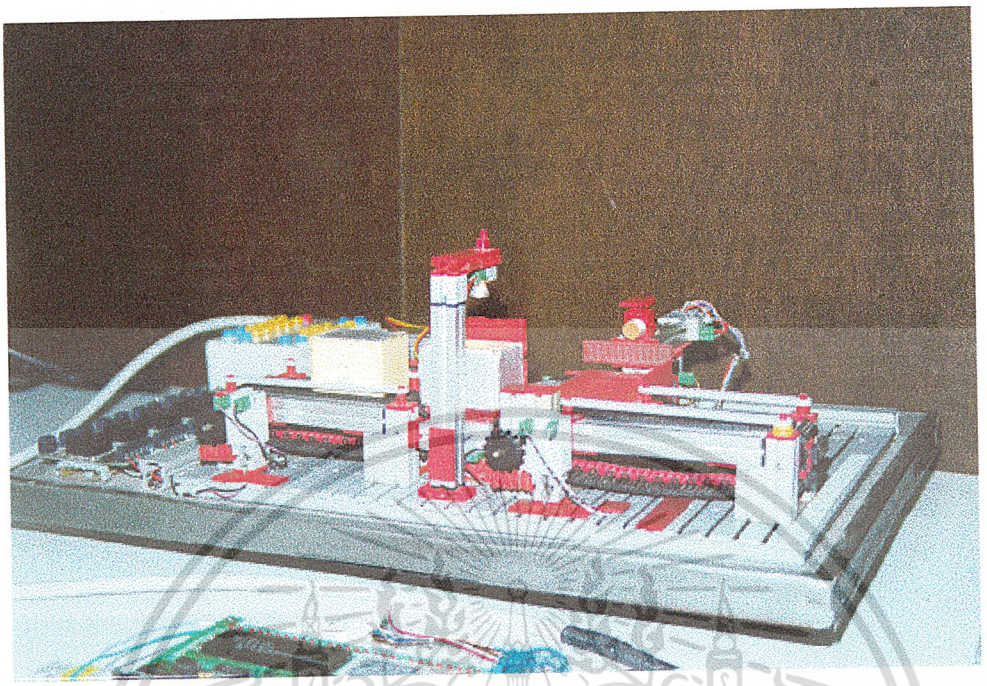
7.1.1 ตัวอย่างการควบคุมที่1 ชุดสายพานลำเลียงคัดเลือกกล่อง

7.1.1.1 ส่วนประกอบของชุดสายพานลำเลียงคัดเลือกกล่อง



รูปที่7-1 แสดงอุปกรณ์ของชุดสายพานลำเลียงคัดเลือกกล่อง(ด้านบน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่7-2 แสดงอุปกรณ์ของชุดสายพานลำเลียงคัดเลือกกล่อง(ด้านข้าง)

7.1.1.2 ขั้นตอนในการทำงานของชุดสายพานลำเลียงคัดเลือกกล่อง

มอเตอร์ M_1 เริ่มทำงาน สายพานเลื่อนสิ่งของซึ่งเป็นกล่องสี่เหลี่ยม (มีด้านด้านหนึ่งเป็นโลหะ) ไปจนถึงถาดพลิก Photo sensor ทำงานสั่งให้มอเตอร์ M_3 ทำงาน กล่องถูกพลิกมาอีกตำแหน่ง มอเตอร์ M_2 ทำงานผลักให้กล่องเลื่อนมาหาชุดตรวจจับโลหะ จากนั้นกลับเข้าที่เดิม มอเตอร์ M_4 ทำงานพร้อมกับตรวจสอบว่าด้านโลหะสัมผัสกับ Proximity Sensor หรือไม่ จากนั้นกลับเข้าที่เดิม มอเตอร์ M_5 ทำงานโดยที่ ถ้าหากการตรวจจับพบว่า Proximity สัมผัสกับด้านโลหะก็จะส่งกล่องสี่เหลี่ยม ออกไปเป็นการจบการทำงาน แต่ถ้าไม่ใช่จะส่งกลับมาเพื่อพลิกด้านถัดไปแล้วตรวจสอบอีกครั้ง

7.1.1.3 การกำหนดอินพุทเอาท์พุทของชุดสายพานลำเลียงคัดเลือกกล่อง

ในกำหนด อินพุทเอาท์พุทเพื่อควบคุมการทำงานโดยใช้เครื่องควบคุม PLC/PC โดยที่รีเลย์ แต่ละตัวจะมีตัวเลข 4 หลักด้วยกัน สองหลักแรกเป็นหมายเลขประจำ Word แต่ละWord มีหน้าที่ต่างกันไป เช่น อินพุท เอาท์พุท หรือ รีเลย์ภายใน และสองหลักสุดท้ายเป็นหมายเลขประจำ Bit โดยเริ่มต้นจาก 00 ตัวอย่างเช่น 0703 คือ Word ที่07 มีหน้าที่เป็น เอาท์พุท บิทที่03

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

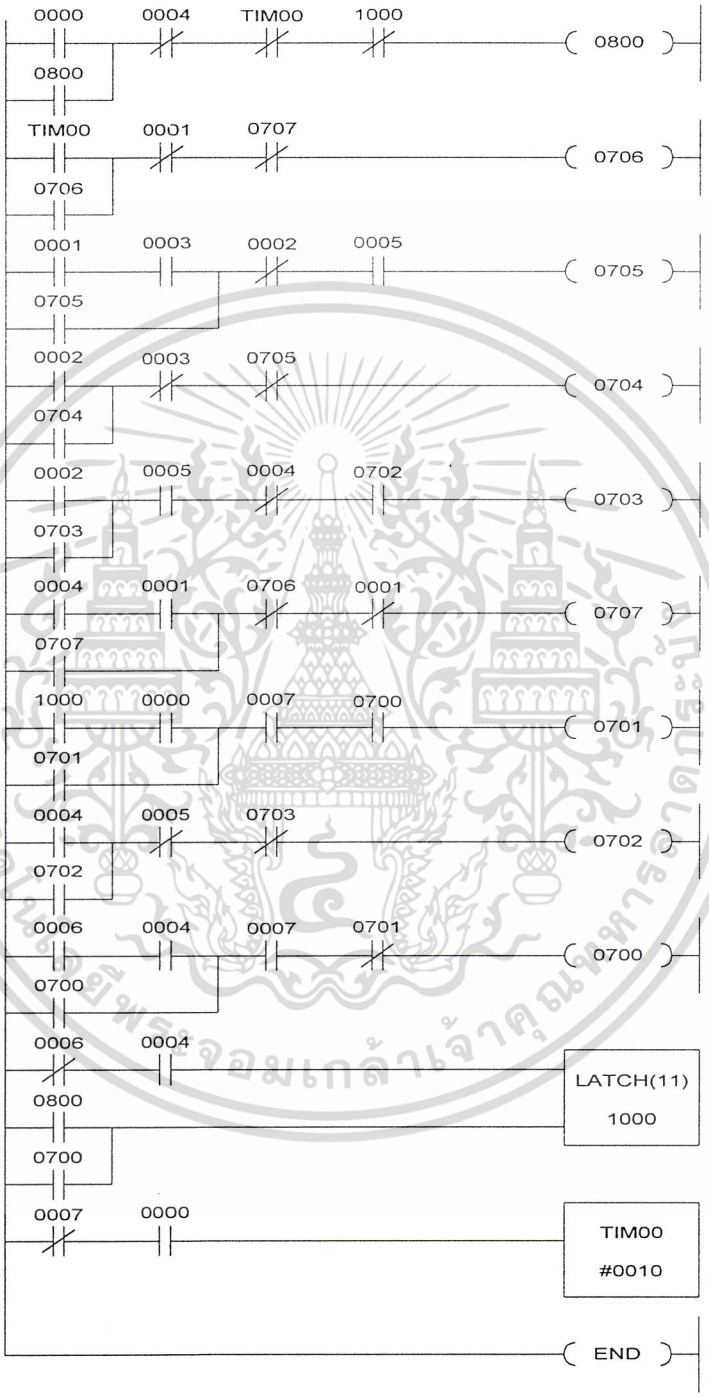
อินพุทรีเลย์	อุปกรณ์	หมายเหตุ
0000	ลิมิตสวิทซ์ L_0	ปกติเปิด
0001	ลิมิตสวิทซ์ L_1	ปกติเปิด
0002	ลิมิตสวิทซ์ L_2	ปกติเปิด
0003	ลิมิตสวิทซ์ L_3	ปกติเปิด
0004	ลิมิตสวิทซ์ L_4	ปกติเปิด
0005	ลิมิตสวิทซ์ L_5	ปกติเปิด
0006	Proximity Sensor Prox	ปกติปิด
0007	Photo sensor Photo	ปกติปิด

เอาท์พุทรีเลย์	อุปกรณ์	หมายเหตุ
0700	มอเตอร์ M_5 FF	ไปข้างหน้า
0701	มอเตอร์ M_5 RV	กลับหลัง
0702	มอเตอร์ M_4 IN	เลื่อนเข้า
0703	มอเตอร์ M_4 OUT	เลื่อนออก
0704	มอเตอร์ M_2 IN	เลื่อนเข้า
0705	มอเตอร์ M_2 OUT	เลื่อนออก
0706	มอเตอร์ M_3 OUT	เลื่อนออก
0707	มอเตอร์ M_3 IN	เลื่อนเข้า
0800	มอเตอร์ M_1	

ตารางที่7-1 แสดงการกำหนดอินพุทเอาท์พุทของชุดสายพานลำเลียงคัดเลือกกล่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

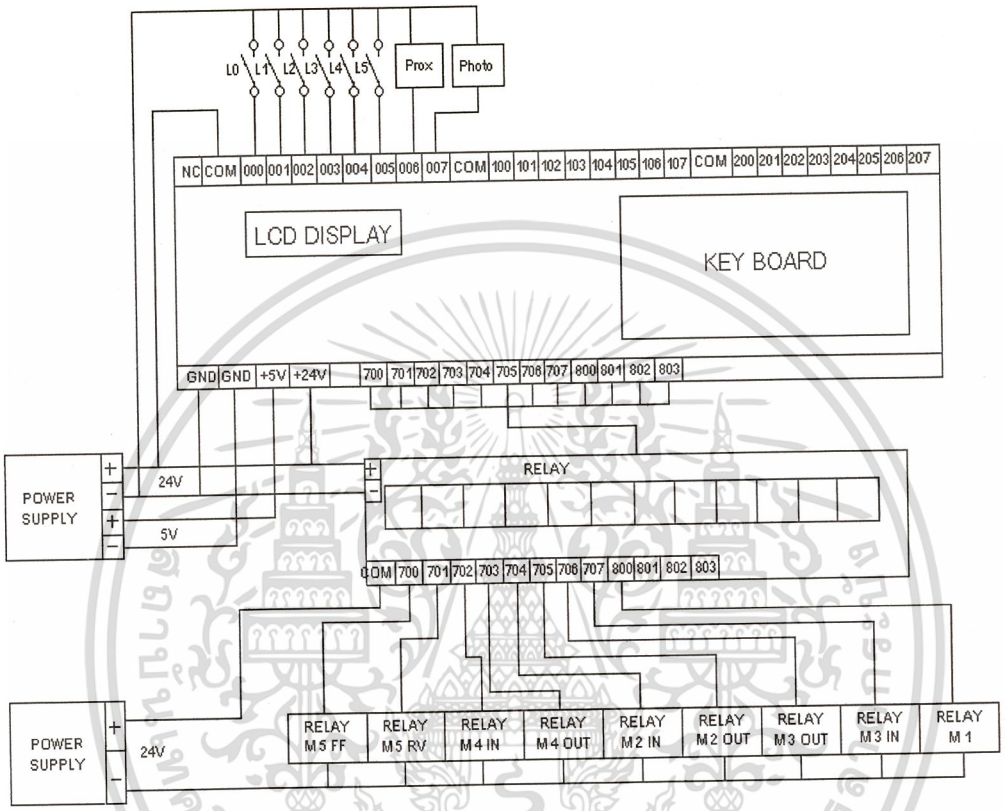
7.1.1.3 โปรแกรมแลดเดอร์ไดอะแกรมของชุดสายพานลำเลียงคัดเลือกกล่อง



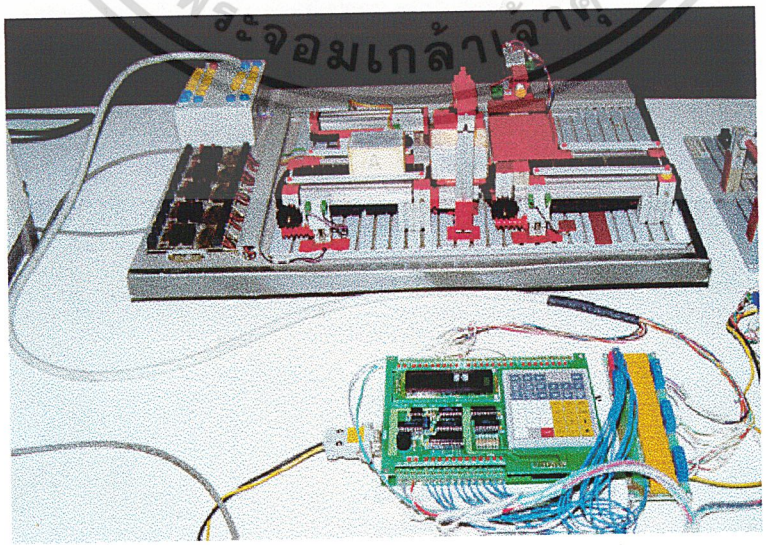
รูปที่7-3 แสดงโปรแกรมแลดเดอร์ไดอะแกรมของชุดสายพานลำเลียงคัดเลือกกล่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.1.4 วงจรการเดินสายของเครื่องควบคุมPLC/PCกับอุปกรณ์ภายนอก



รูปที่ 7-4 แสดงการต่อวงจรเดินสายไฟของชุดสายพานลำเลียงคัดเลือกกล่อง

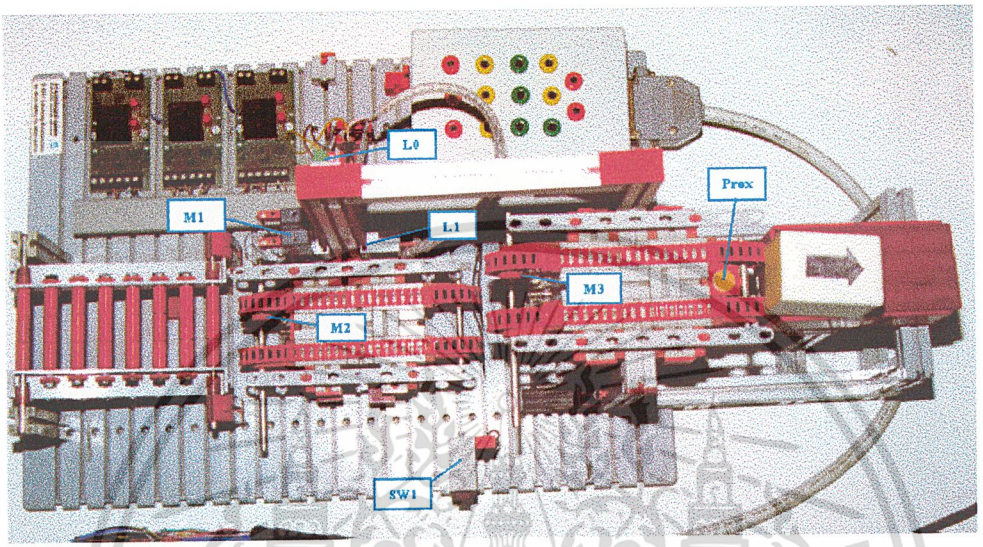


รูปที่7-5 แสดงการเดินสายไฟระหว่างเครื่องควบคุมกับชุดสายพานลำเลียงคัดเลือกกล่อง

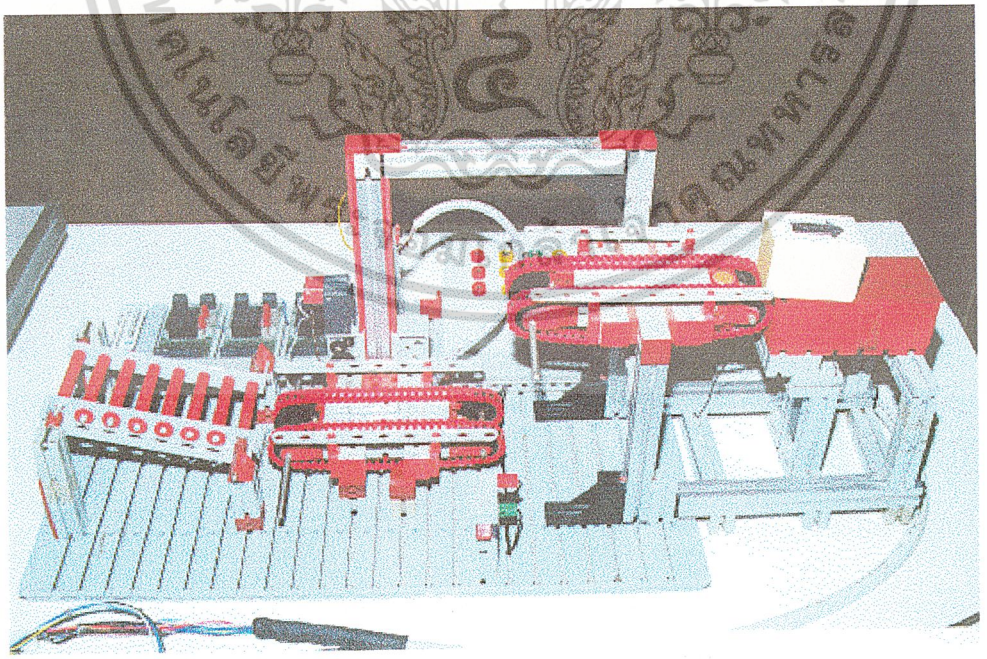
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.2 ตัวอย่างการควบคุมที่ 2 ชุดสายพานยกกล่องอัตโนมัติ

7.1.2.1 ส่วนประกอบของชุดสายพานยกกล่องอัตโนมัติ



รูปที่7-6 แสดงอุปกรณ์ของชุดสายพานยกกล่องอัตโนมัติ(ด้านบน)



รูปที่7-7 แสดงอุปกรณ์ของชุดสายพานยกกล่องอัตโนมัติ(ด้านข้าง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.2.2 ขั้นตอนในการทำงานของชุดสายพานยกกล่องอัตโนมัติ

เมื่อมีของวางที่สายพาน 1 ผู้ปฏิบัติงานกด SW1 มอเตอร์ M1 จะทำการเคลื่อนสายพานเลื่อนกล่องขึ้นบนสายพาน โดยตั้งเวลาไว้ที่ 2 วินาที จากนั้นมอเตอร์ M2 จะทำการยกของขึ้นชั้นบน และส่งต่อให้กับสายพาน 2 โดยตั้งเวลาไว้ที่ 3 วินาที จากนั้นกล่องจะเลื่อนไปตามสายพานจนผ่าน Proximity Sensor ตรวจจับด้านโลหะจากนั้นมอเตอร์ M2 จะยกชุดสายพาน 1 ลงสู่ตำแหน่งปกติ ในเวลาเดียวกัน สายพาน 2 ก็จะหยุดโดยหน่วงเวลา 1.5 วินาที ในกรณีที่ไม่พบด้านที่ไม่เป็นโลหะ ระบบจะรีเซทตัวเองภายใน 5 วินาที

7.1.2.3 การกำหนดอินพุทเอาท์พุทของชุดสายพานยกกล่องอัตโนมัติ

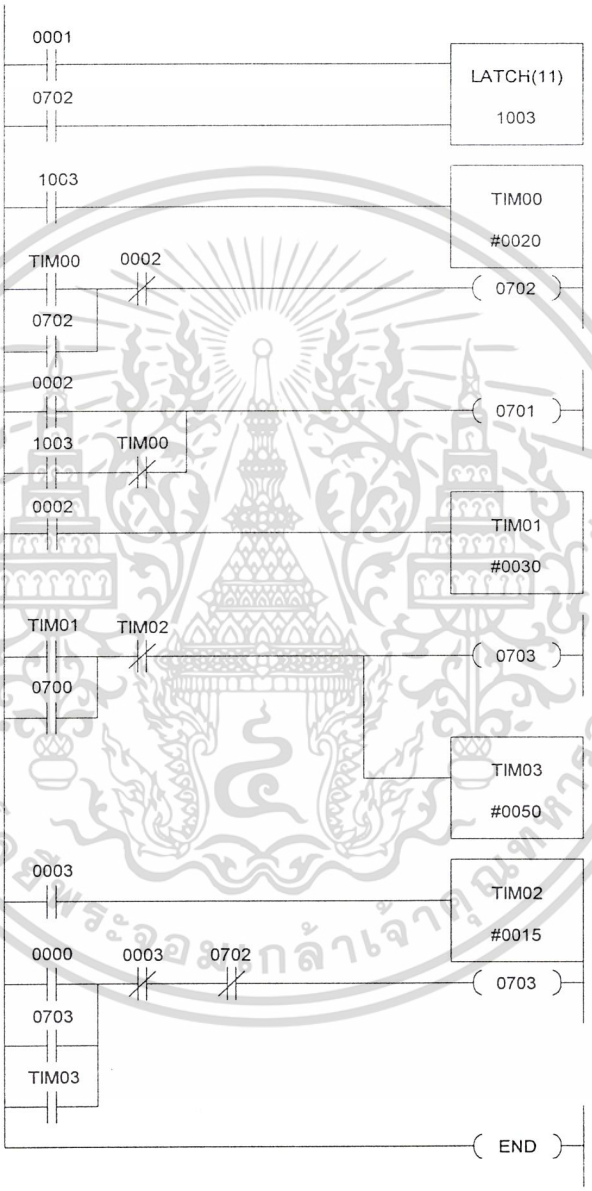
อินพุทรีเลย์	อุปกรณ์	หมายเหตุ
0000	Proximity Sensor (Prox)	ปกติเปิด
0001	Start Switch (SW_1)	ปกติเปิด
0002	ลิมิตสวิทซ์ L_0	ปกติเปิด
0003	ลิมิตสวิทซ์ L_1	ปกติปิด

เอาท์พุทรีเลย์	อุปกรณ์	หมายเหตุ
0700	มอเตอร์ M_3	สายพาน 2
0701	มอเตอร์ M_2	สายพาน 1
0702	มอเตอร์ M_{1UP}	ขึ้นข้างบน
0703	มอเตอร์ M_{1DOWN}	ลงข้างล่าง

ตารางที่7-2 แสดงการกำหนดอินพุทเอาท์พุทของชุดสายพานยกกล่องอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

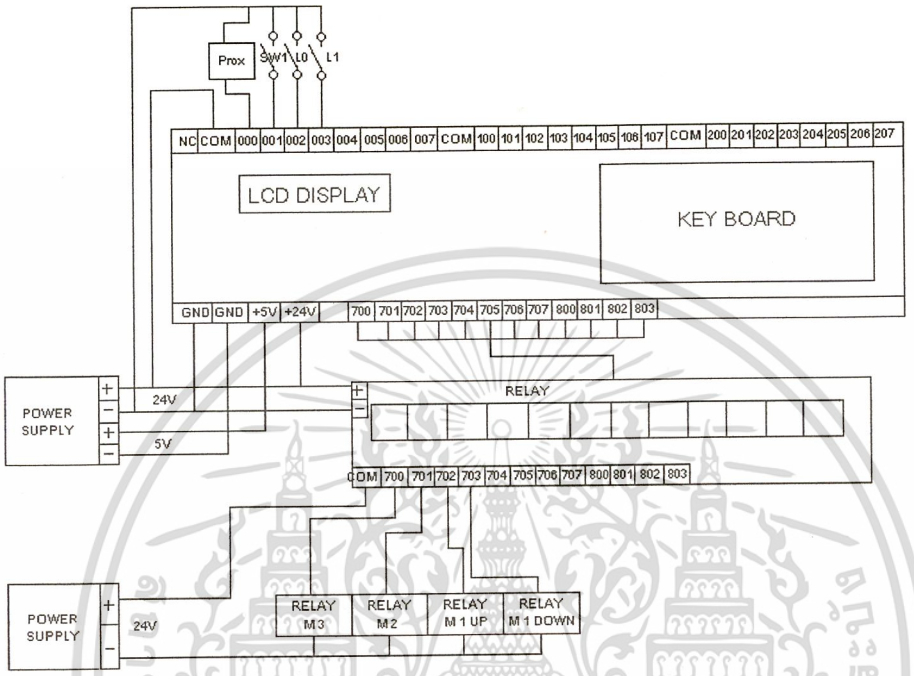
7.1.2.4 โปรแกรมแลดเดอร์ไดอะแกรมของชุดสายพานยกกล่องอัตโนมัติ



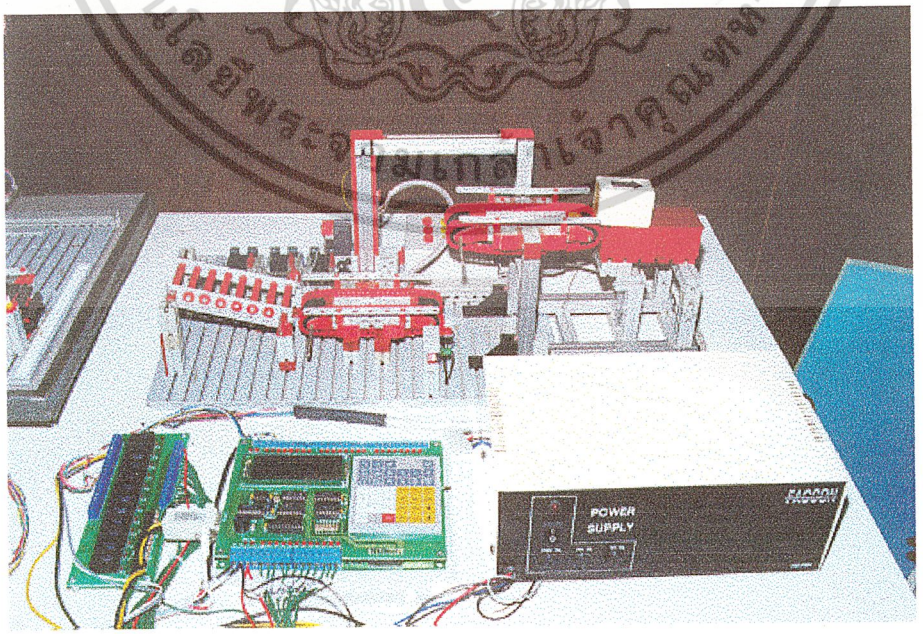
รูปที่7-8 แสดงโปรแกรมแลดเดอร์ไดอะแกรมของชุดสายพานยกกล่องอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.2.5 วงจรการเดินสายไฟของเครื่องควบคุม PLC/PC กับอุปกรณ์ภายนอก



รูปที่7-9 แสดงการต่อวงจรเดินสายไฟของชุดสายพานยกกล่องอัตโนมัติ

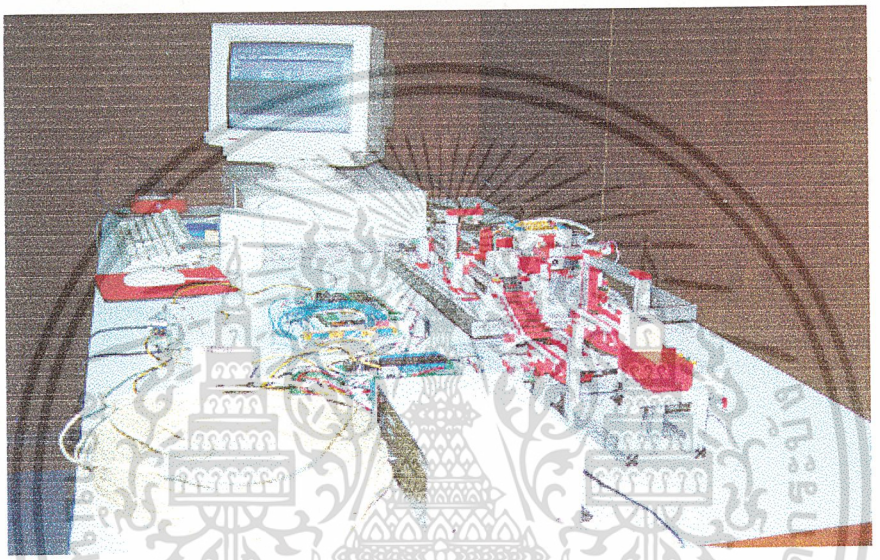


รูปที่7-10 แสดงการเดินสายไฟระหว่างเครื่องควบคุมกับชุดสายพานยกกล่องอัตโนมัติ

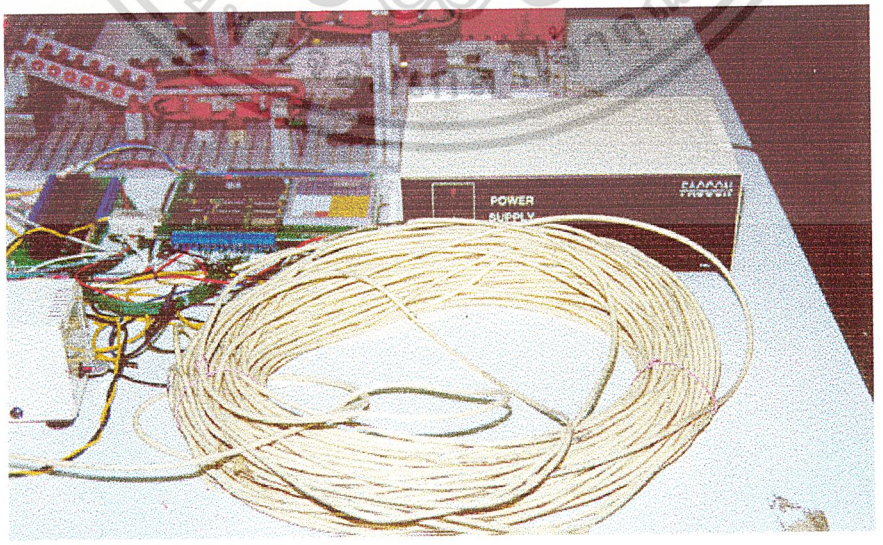
เอกสารนี้เป็นลิขสิทธิ์ของ [illegible] ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2 การประยุกต์ใช้งานของเครื่องควบคุม PLC/PC ร่วมกับโปรแกรมสนับสนุนการทำงาน

การทดสอบการทำงานของโปรแกรมสนับสนุนการทำงานโดยต่อเครื่องควบคุม PLC/PC จำนวน 2 เครื่องเข้าในระบบ โดยใช้สายตัวนำความยาว 100 เมตรเชื่อมต่อระหว่างโฮสคอมพิวเตอร์ กับเครื่องควบคุม PLC/PC และได้แสดงผลการทดสอบของโปรแกรมรวมกับชุดจำลองสายพาน ชุดที่ 2 ดังต่อไปนี้

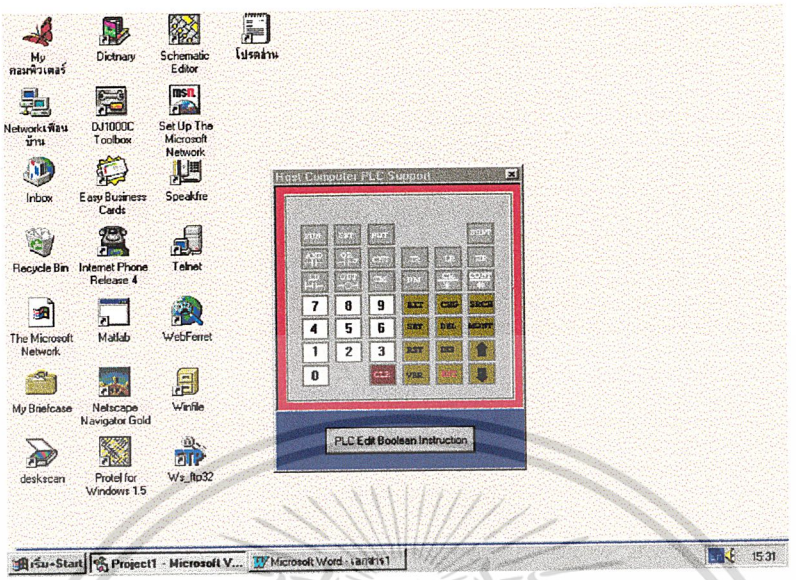


รูปที่7-11 แสดงการทดสอบระบบโครงข่ายผ่านโฮสคอมพิวเตอร์

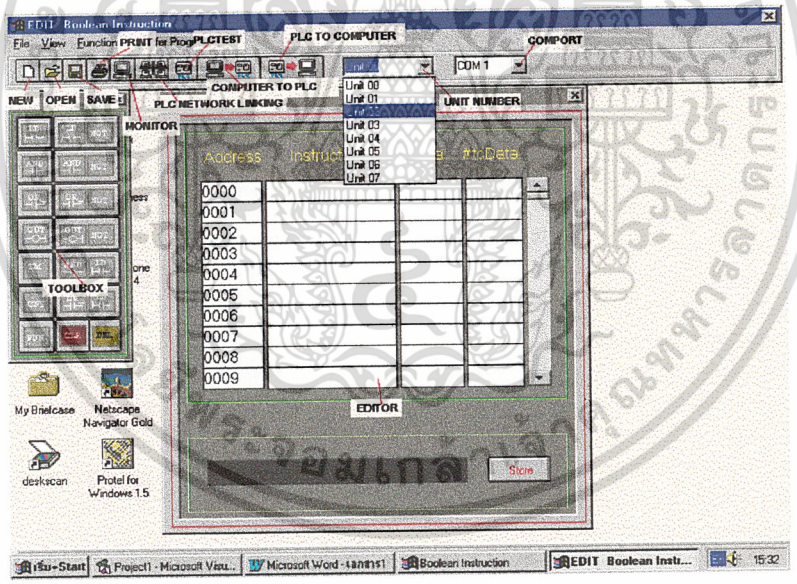


รูปที่7-12 แสดงสายนำสัญญาณความยาว 100 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่7-13 แสดงการเริ่มต้นเข้าสู่โปรแกรม



รูปที่ 7-14 แสดงหน้าที่ของปุ่มคำสั่งในโปรแกรม

7.2.1 หน้าที่ของปุ่มคำสั่งในโปรแกรมสนับสนุนการทำงาน

1. New การสร้างไฟล์ชุดคำสั่งบูลีนใหม่
2. Open การเปิดไฟล์จากหน่วยความจำ
3. Save การบันทึกไฟล์ลงหน่วยความจำ
4. Print การพิมพ์โปรแกรมชุดคำสั่งบูลีนออกทางเครื่องพิมพ์

เอกสารนี้เป็นการที่ได้รับอนุญาตให้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Monitor การเข้าสู่การตรวจสอบสภาวะ

6. PLC Network linking การเข้าสู่การเชื่อมโยงข้อมูลของ PLC ในระบบผ่านโฮสคอมพิวเตอร์

7. PLC Test การทดสอบความพร้อมของเครื่องควบคุมที่เชื่อมต่ออยู่ในระบบโครงข่าย

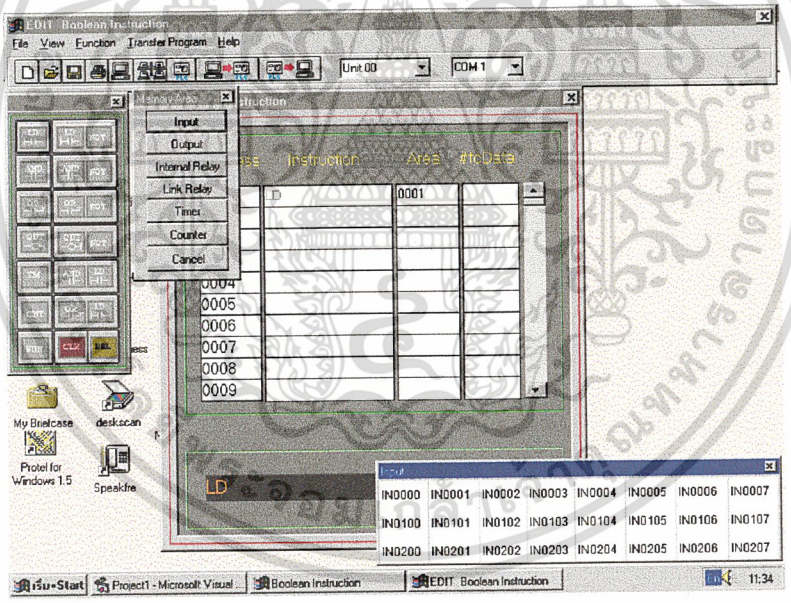
8. Computer to PLC การถ่ายเทโปรแกรมชุดคำสั่งบูลีนจากคอมพิวเตอร์ไปยังเครื่องควบคุมPLC

9. PLC to Computer การถ่ายเทโปรแกรมชุดคำสั่งบูลีนจาก เครื่องควบคุมPLCไปยังคอมพิวเตอร์

10. Unit Number การเลือกเครื่องควบคุมที่จะทำการติดต่อ

11. Comport การเลือกพอร์ทอนุกรมของคอมพิวเตอร์ที่จะทำการติดต่อ

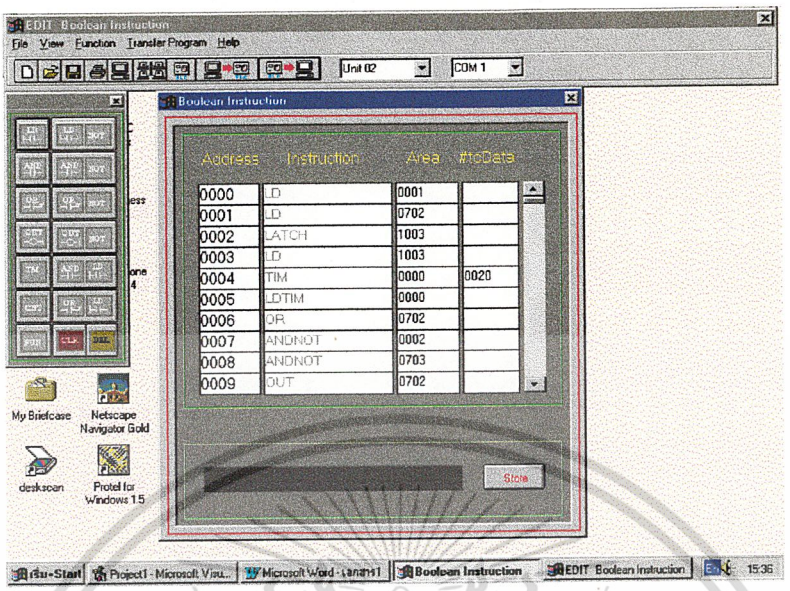
12. Toolbox ปุ่มคำสั่งในการเขียนชุดคำสั่งบูลีนลงบน EDITOR



รูปที่7-15 การเขียนโปรแกรมจาก Toolbox

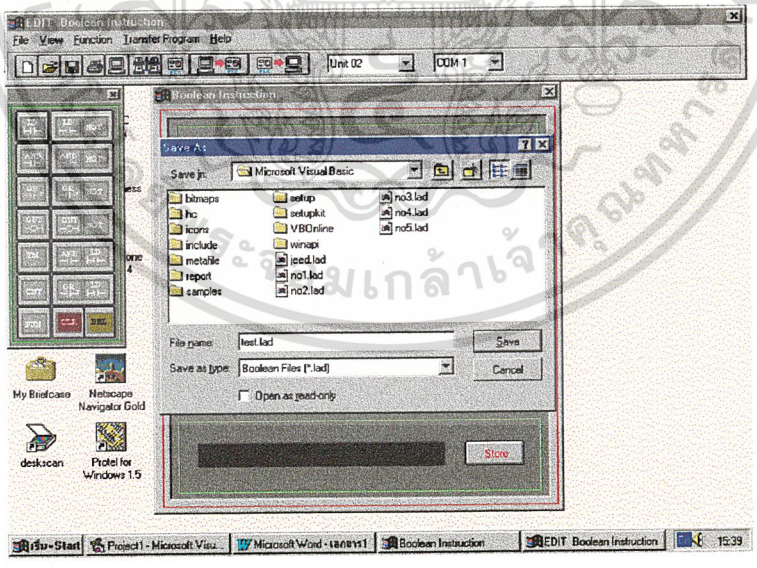
ในการเขียนโปรแกรมจะเริ่มโปรแกรมตั้งแต่ Address ที่ 0000 จนกระทั่งจบโปรแกรมแล้วตามด้วยคำสั่ง END โดยคลิกเมาส์ที่ปุ่มของ Tool Box เพื่อเลือกคำสั่ง จากนั้นจะปรากฏหน้าต่างของพื้นที่หน่วยความจำ เช่น อินพุท เอาท์พุท รีเลย์ภายใน หรือ ตัวตั้งเวลา ตัวนับ ให้เลือกและนำเมาส์คลิกที่บรรทัด ชุดคำสั่งก็จะปรากฏที่บรรทัดนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่7-16 แสดงชุดคำสั่งบูลีนบน Editor

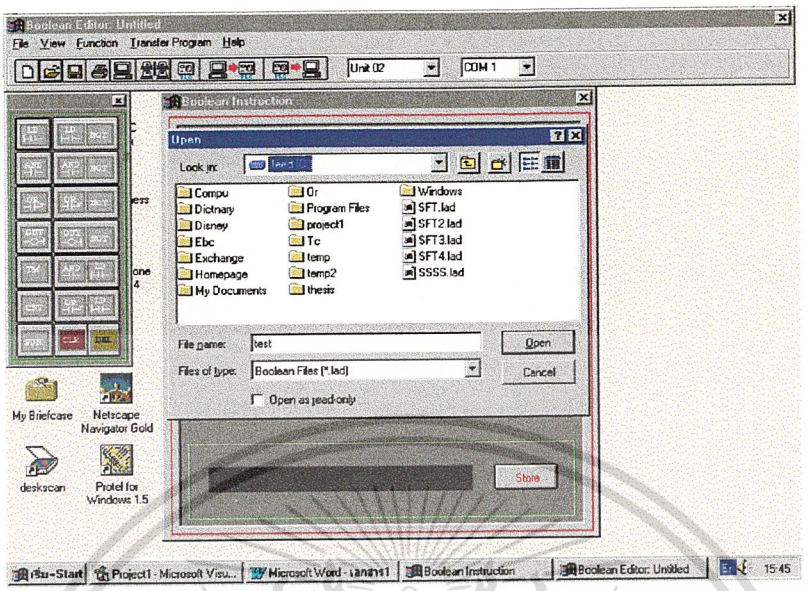
โปรแกรมชุดคำสั่งบูลีนเมื่อเสร็จแล้วต้องตามด้วยคำสั่ง END ได้จากการเลือกที่ ฟังก์ชัน เพื่อแสดงจุดสิ้นสุดของโปรแกรม ในขั้นตอนต่อไปจะทำการบันทึกข้อมูลลงบนไฟล์และทำการถ่ายเทข้อมูลจากคอมพิวเตอร์ไปยังเครื่องควบคุม PLC/PC



รูปที่7-17 การบันทึกโปรแกรมชุดคำสั่งบูลีนลงหน่วยความจำ

โปรแกรมที่เขียนแล้วสามารถบันทึกลงในแฟ้มข้อมูล โดยตั้งชื่อของไฟล์ และไดเร็กตอรีที่จะทำการจัดเก็บ

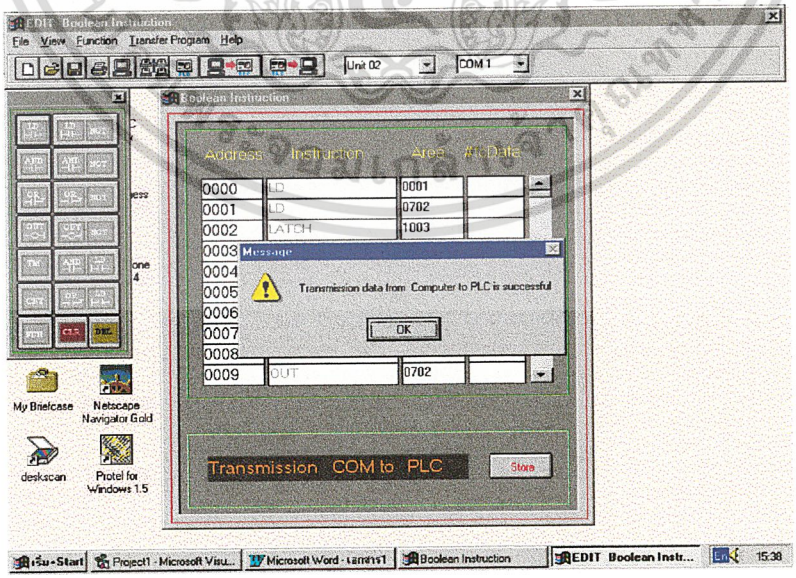
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่7-18 แสดงการเปิดไฟล์ข้อมูลจากหน่วยความจำ

การเปิดไฟล์ โดยการเรียกไฟล์ที่จัดเก็บขึ้นมาเพื่อใช้ งาน เช่น แก้ไขหรือ ถ่ายเทโปรแกรม โดยกำหนดให้ไฟล์มีนามสกุลว่า “ LAD”

การพิมพ์โปรแกรมจะกระทำได้โดยการคลิกที่ปุ่มคำสั่ง “ Print “ ในการพิมพ์กำหนดให้มี 50 บรรทัด ต่อ หน้า และ แสดงหน้าของโปรแกรมที่มุมบนด้านขวาดังตัวอย่างในภาคผนวกเป็นโปรแกรมที่ใช้ในการทดลอง

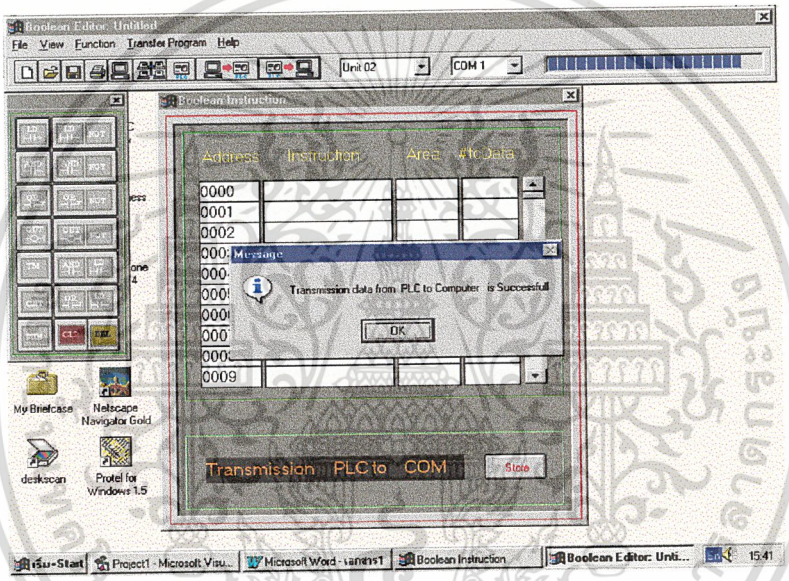


รูปที่7-19 แสดงการถ่ายเทโปรแกรมจากคอมพิวเตอร์ไปเครื่องควบคุม PLC/PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก่อนที่จะทำการถ่ายเทโปรแกรมจากคอมพิวเตอร์ไปยังเครื่องควบคุม PLC/PC ให้คลิกที่ปุ่ม "Store" เพื่อเตรียมข้อมูลไว้ในตัวแปรอาร์เรย์ ที่จัดทำเป็นเรคคอร์ด จากนั้นคลิกที่ปุ่ม Computer to PLC ชุดคำสั่งจะถูกแปลงให้เป็น Code ส่งให้กับเครื่องควบคุม PLC/PC จนกระทั่งถึงคำสั่ง "END" และมีข้อความแสดงให้ทราบดังรูปข้างต้น

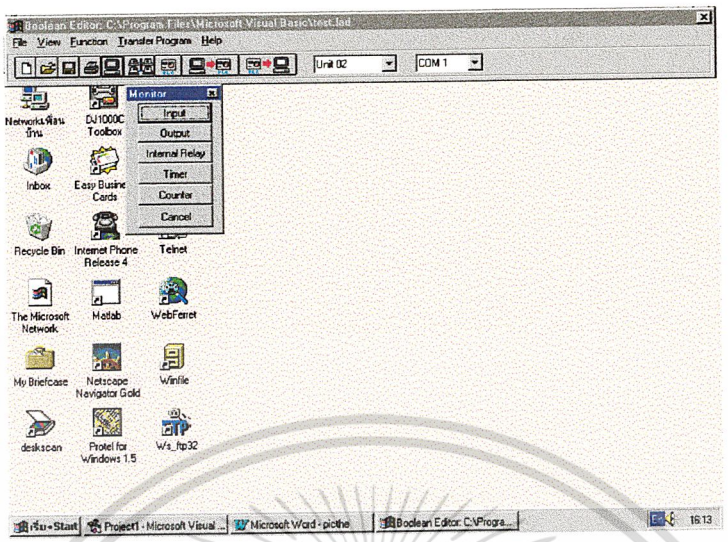
ในการสื่อสารถ้าหากมีข้อผิดพลาด เช่น สายสัญญาณ หรือ การตั้งหมายเลขประจำเครื่องไม่ถูกต้อง ก็จะมีข้อความแจ้งให้ทราบ



รูปที่7-20 การถ่ายเทโปรแกรมจากเครื่องควบคุม PLC/PC ไปยังคอมพิวเตอร์

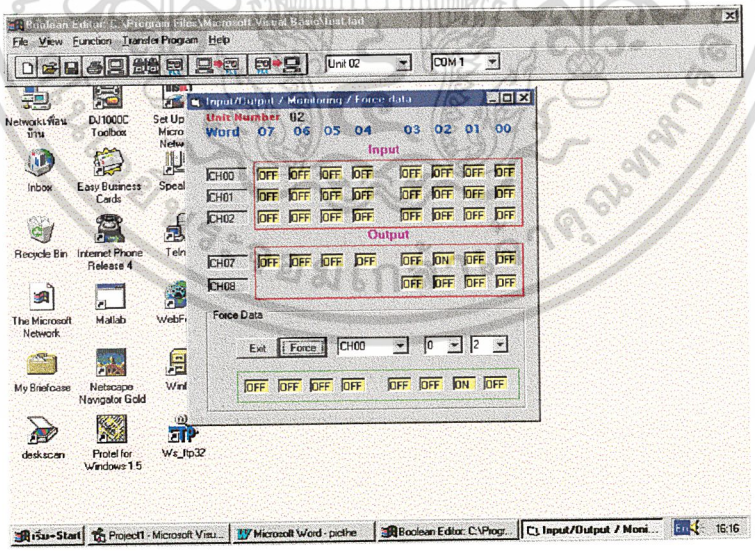
ในการถ่ายเทโปรแกรมจากเครื่องควบคุม PLC/PC มายังคอมพิวเตอร์ เพื่อที่จะทำการบันทึกหรือแก้ไขใหม่ โดยการ คลิกที่ปุ่ม PLC to Computer เครื่องควบคุมจะส่ง Code ของชุดคำสั่งทั้งหมดจนกระทั่งพบ Code ของคำสั่ง "END" และแสดงผลเป็นชุดคำสั่งบูลีนบน คอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่7-21 แสดงการเลือกตรวจสอบสภาวะ

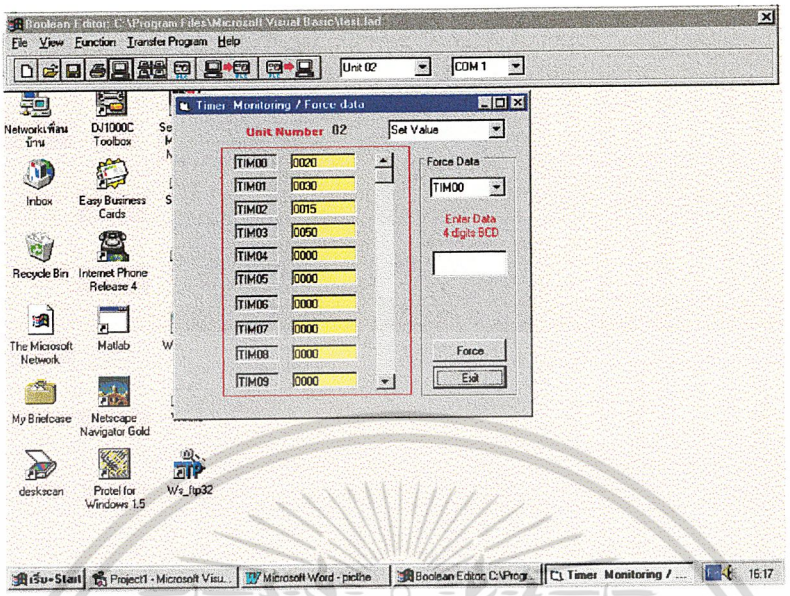
ในการตรวจสอบสภาวะสามารถที่จะเลือกดูสภาวะของ อินพุท เอาท์พุท รีเลย์ภายใน ตัวตั้งเวลา และ ตัวนับของเครื่องควบคุมตัวใดตัวหนึ่งที่เชื่อมต่อในโครงข่าย โดยเลือกจาก Unit Number



รูปที่7-22 การตรวจสอบสภาวะของ อินพุทและ เอาท์พุท

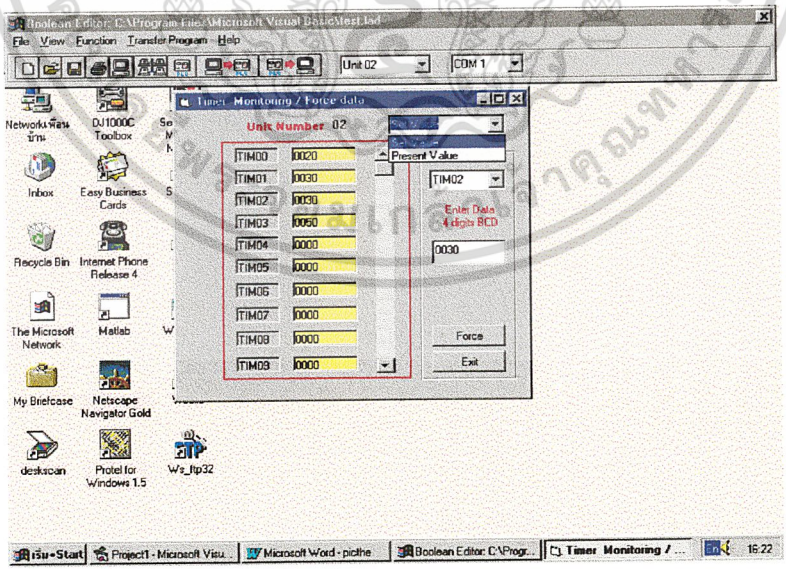
ในการแสดงผลของ อินพุท เอาท์พุท จะแสดงสภาวะ “ON” หรือ “OFF” ของแต่ละ Bit ในแต่ละ Word นั้น ส่วนในการกำหนดค่า (Force) จะกระทำได้เป็น Word โดยกำหนดเป็นเลขฐาน 16 และมีการแสดงสภาวะเป็นการ “ON” หรือ “OFF” ของแต่ละ Bit ในส่วนล่างของหน้าต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7-23 การตรวจสอบสภาวะของตัวตั้งเวลา

ในการแสดงผลของตัวตั้งเวลาหรือตัวนับ จะคล้ายกัน โดยแสดงผลเป็นตัวเลข 4 หลักในตัวตั้งเวลาจะตั้งช่วงได้ตั้งแต่ 0.1 ถึง 999.9 วินาที สำหรับตัวนับจะตั้งค่าการนับได้ตั้งแต่ 0 ถึง 9999 ตัวตั้งเวลาหรือตัวนับมีทั้งสิ้น 48 ตัว โดยทำการเลื่อนที่แถบ Scroll Bar เพื่อเลือกดูข้อมูลของตัวตั้งเวลา ตัวนับตัวอื่น



รูปที่7-24 การเปลี่ยนแปลงค่าของตัวตั้งเวลา

ในการกำหนดค่าเพื่อเปลี่ยนแปลงข้อมูลให้กับตัวตั้งเวลา ตัวนับ จะต้องทำการป้อนตัวเลขจำนวน 4 หลัก และ เลือกตัวตั้งเวลาหรือตัวนับที่ต้องการเปลี่ยนแปลง และคลิกที่ปุ่ม “Force”

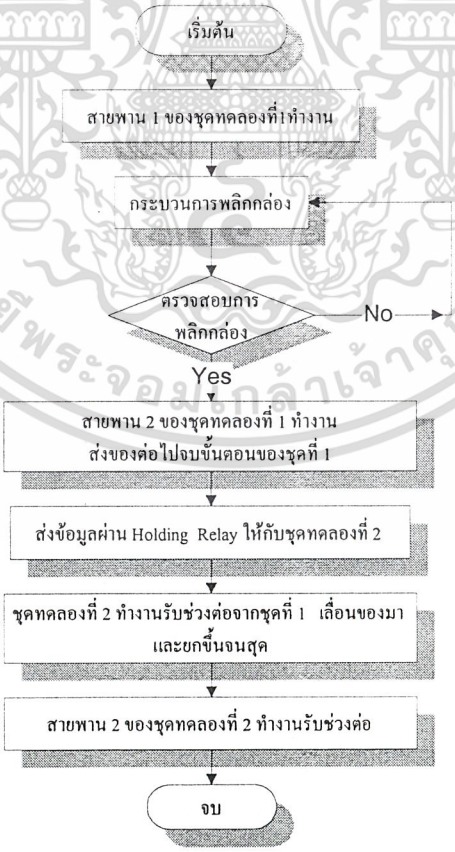
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจะเปลี่ยนแปลงและแสดงผล โดยกระทำได้ครั้งละ 1 ตัว จากรูปแสดงให้เห็นการเปลี่ยนแปลงค่าของ TIM02 จากการกำหนดค่าในการตรวจสอบสภาวะของตัวตั้งเวลาและตัวนับจะมีข้อมูล 2 ชนิดด้วยกันคือ

- Set Value คือค่าที่ตั้งการทำงานโดยทำการป้อนค่าขณะเขียนโปรแกรม
- Present Value คือค่าที่เปลี่ยนแปลงตามการทำงานของตัวตั้งเวลาหรือตัวนับตัวนั้น

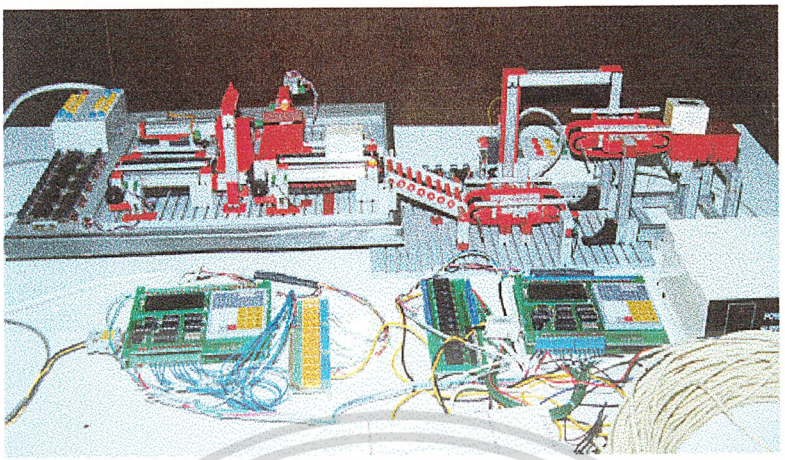
7.2.2 การทดสอบ การเชื่อมโยงข้อมูลโครงข่ายผ่านโฮสคอมพิวเตอร์

การทดสอบนี้จะทำการทดสอบการเชื่อมโยงข้อมูลของ เครื่องควบคุมทั้งสองเครื่องโดยมีโฮส คอมพิวเตอร์เป็นตัวจัดการข้อมูล และ มีการแบ่ง HR00 เป็นพื้นที่ของเครื่องควบคุม PLC/PC หมายเลข 00 และ HR01 ของเครื่องควบคุม หมายเลข 01 ตามลำดับ โดยกำหนดให้มีพื้นที่ในการเชื่อมโยงเครื่องละ 1 word หรือ 8 bit มีการเปลี่ยนแปลงโปรแกรม แลดเดอร์ ไดอะแกรมที่เครื่องควบคุม โดยกำหนดให้การเชื่อมโยงข้อมูลระหว่างสายพานทั้ง 2 ชุด ทำงานสอดคล้องกัน คือ สายพานชุดที่ 2 จะรับกล่องจากการพลิกกล่องของสายชุดสายพานชุดที่ 1 ถ้าหากสายพานชุดที่1 ไม่ส่งกล่องมา สายพานชุดที่ 2 ก็จะไม่ทำงาน ดังแผนผังการทำงานต่อไปนี้



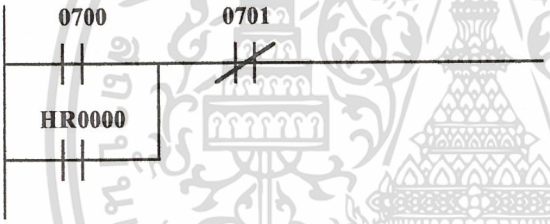
รูปที่7-25 แสดงผังการทำงานร่วมกันของชุดสายพาน 2 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้นไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่7-26 แสดงการเชื่อมต่อทางฮาร์ดแวร์ของชุดจำลองทั้ง 2 ชุด

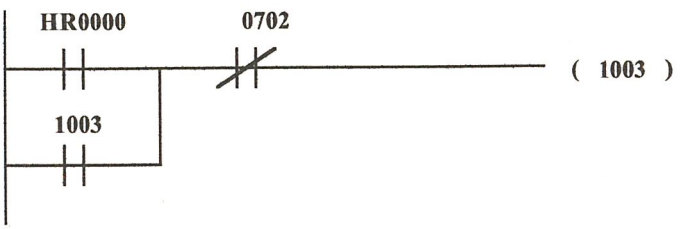
7.2.2.1โปรแกรมแลดเดอร์ที่เพิ่มให้กับสายพานชุดที่1



รูปที่7-27 โปรแกรมแลดเดอร์ไดอะแกรมที่แทรกในเครื่องควบคุมPLC/PCชุดที่1

เครื่องควบคุมPLC/PCชุดที่1 ถูกกำหนดหมายเลขประจำเครื่องเป็น UNIT 00 พื้นที่ HR ที่ทำการเขียนข้อมูลคือ HR00 โดยนำข้อมูลจาก 0700 ซึ่งเป็น Output Relay ของสายพานชุดที่1 ในการลำเลียงสายพานออกและใช้ Output Relay 0701 ซึ่งเป็นการลำเลียงสายพานกลับเข้าไปผ่านกระบวนการใหม่ ดังนั้น ถ้าหากสายพานชุดที่ 1 ทำงานครบตามเงื่อนไข HR0000 จะมีค่าสภาวะลอจิกเป็น 1

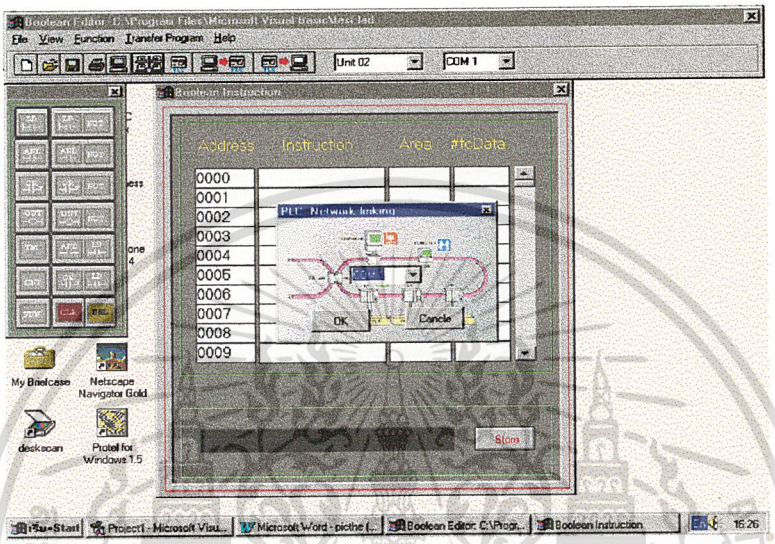
7.2.2.2 โปรแกรมแลดเดอร์ที่เพิ่มให้กับสายพานชุดที่ 2



รูปที่7-28 โปรแกรมแลดเดอร์ไดอะแกรมที่แทรกในเครื่องควบคุมPLC/PCชุดที่2

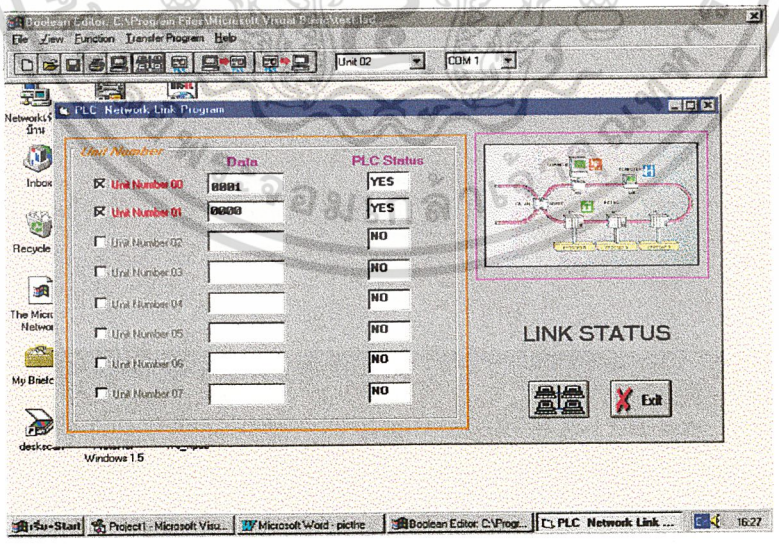
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่ทำการแทรกแทนการกด SW1 ซึ่งเป็นสวิทซ์สตาร์ทโดย รับข้อมูลจาก HR0000ซึ่งเป็นข้อมูลจากสายพานชุดที่1 ให้ชุดสายพานชุดที่ 2 ทำงานโดยอัตโนมัติ ทันทีที่มีกล่องเคลื่อนมาจากสายพานชุดที่1 ซึ่งการเชื่อมต่อทางฮาร์ดแวร์ยังคงเหมือนเดิม



รูปที่7-29 แสดงโปรแกรมเชื่อมโยงข้อมูลด้วยโฮสคอมพิวเตอร์

เริ่มต้น โปรแกรมเชื่อมโยงข้อมูลด้วยโฮสคอมพิวเตอร์ โดยการตั้งพอร์ทสื่อสารอนุกรมของคอมพิวเตอร์



รูปที่7-30 แสดงโปรแกรมเชื่อมโยงข้อมูลด้วยโฮสคอมพิวเตอร์

การยกเลิกการเชื่อมโยงข้อมูลด้วยโฮสคอมพิวเตอร์ ก็สามารถกระทำได้ทางซอฟแวร์ไม่ต้องทำการเปลี่ยนแปลงใดๆ ทางฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8
บทสรุป

การพัฒนาระบบการทำงานของเครื่องควบคุมแบบตรรกที่โปรแกรมได้หรือเครื่องควบคุม PLC/PC ส่วนมากจะมุ่งเน้นในส่วนการประยุกต์ใช้งานเพื่อการควบคุมในรูปแบบต่างๆ ลักษณะการควบคุมเป็นแบบจุดต่อจุด คือ เครื่องจักร 1 เครื่องต่อ เครื่องควบคุม 1 เครื่อง ในวิทยานิพนธ์นี้ได้นำเสนอการออกแบบและการสร้างเครื่องควบคุมPLC/PC ที่สามารถเชื่อมต่อกันเป็นระบบโครงข่ายแบบหลายจุด(Multipoint)พร้อมกับโปรแกรมสนับสนุนการทำงานซึ่งติดตั้งอยู่บนคอมพิวเตอร์ในที่นี้เรียกว่า โฮสคอมพิวเตอร์ทำหน้าที่ในการจัดการข้อมูล การเชื่อมต่อตามมาตรฐาน RS 485 ซึ่งใช้สายสัญญาณเพียง 2 เส้น และสามารถเชื่อมต่อกันได้ในระยะทางที่ไกลนับร้อยเมตร ทำให้การตรวจสอบสภาวะ การจัดการข้อมูล เช่น การจัดเก็บเป็นแฟ้มข้อมูล การแก้ไข และการถ่ายเทโปรแกรมของชุดคำสั่งบูลีน ของเครื่องควบคุมPLC/PC แต่ละเครื่องกับโฮสคอมพิวเตอร์ ที่ถูกจัดวางในสถานที่ที่ต่างกัน ทำได้สะดวกมากขึ้น ขณะเดียวกันก็สามารถเชื่อมต่อกันในลักษณะจุดต่อจุด(Point to point) ตามมาตรฐาน RS 232 C ได้เช่นกัน

นอกจากนี้โปรแกรมบนโฮสคอมพิวเตอร์ถูกออกแบบให้สามารถเชื่อมโยงข้อมูลของเครื่องควบคุมPLC/PC ที่เชื่อมต่ออยู่ในระบบได้ ทำให้เครื่องควบคุมและเครื่องจักรที่ทำการเชื่อมโยงข้อมูล ทำงานได้อย่างสัมพันธ์กัน

เครื่องควบคุมPLC/PC ที่ได้ทำการออกแบบนี้สามารถประยุกต์ใช้ในการควบคุมได้เป็นที่น่าพอใจ รวมทั้งการประยุกต์ใช้งานร่วมกับโปรแกรมสนับสนุนการทำงานได้เป็นอย่างดี แต่อย่างไรก็ตามการทำงานของไมโครโปรเซสเซอร์ ที่สำคัญประการแรกคือแหล่งจ่ายกำลังต้องสะอาด ไม่มีสัญญาณรบกวนคอยขัดจังหวะการทำงาน และปัญหาอีกประการที่พบคือตัวประมวลผลของเครื่องควบคุมPLC/PC ที่ออกแบบต้องทำงานหลายหน้าที่ เช่น ระบบปฏิบัติการ การแสดงผลการประมวลผลของโปรแกรมชุดคำสั่งบูลีน การรับข้อมูลจากแป้นคีย์ และ การติดต่อสื่อสาร ดังนั้นกรณีที่ต้องทำงานพร้อมกันหลายหน้าที่จะทำให้การทำงานช้าลง และเกิดการผิดพลาดได้ในบางครั้ง ตัวอย่างเช่นขณะที่ทำการสื่อสารข้อมูลพร้อมกับการกดแป้นคีย์ บางครั้งเครื่องจะไม่รับการกดแป้นคีย์หรือทำงานช้าลง

แนวทางในการพัฒนาเพื่อเพิ่มประสิทธิภาพการทำงาน คือออกแบบให้มีตัวประมวลผลอีกชุดเพื่อแยกกันทำหน้าที่ในการติดต่อสื่อสารซึ่งจะทำให้การสื่อสารข้อมูลเร็วขึ้น และการพัฒนาโปรแกรมสนับสนุนการทำงานให้สามารถตรวจสอบความถูกต้องของโปรแกรมชุดคำสั่งบูลีนก่อนที่จะนำไปใช้งานจริง การแสดงสภาวะการทำงานด้วยกราฟฟิก เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] Preter Hamann, Steave Willings " MINI-H-TYPE PROGRAMMABLE LOGIC CONTROLLERS OPERATION MANUAL" , 1990

[2] Ahuja Vijay, "Design and Analysis of Computer Communication Networks" ,McGraw-Hill ,Inc.,1985

[3] Mark S. Burgess , "ADVANCED VISUAL BASIC" ,Addison-Wesley Publishing Co. ltd,1994

[4] Joe Campbell , "The RS 232 C Solution " ,Second Edition, TECH Publications,1989

[5] Lance A, Levenhal, Winthrop Sarille Z80 Assembly Language Subroutine

[6] Micro System Component Handbook Volume II, INTEL

[7] ZILOG INTELLIGENT PERIPHERAL CONTROLLER DATA BOOK

[8] เอกสารประกอบการอบรมเชิงปฏิบัติการ PROGRAMMABLE LOGIC CONTROLLER, ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

[9] วิริยะ กองรัตน์ , "เครื่องควบคุมแบบตรรกที่โปรแกรมได้" ,บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, พศ. 2531

[10] สุพรรณ กุลพาณิชย์ , "เทคนิคและการประยุกต์ใช้งาน Programmable Controller" , ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

[11] สุเธียร เกียรติสุนทร, "หลักการทำงานและเทคนิคการประยุกต์ใช้งาน PC/PLC" , ภาควิชาระบบควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

[12] ชูชัย ธนสารตั้งเจริญ , ทินกร ดุ๊ก, "การสื่อสารข้อมูล", ฟิสิกซ์ เซ็นเตอร์ กรุงเทพมหานคร

[13] สุทธิศักดิ์ พงศ์ธนาพาณิชย์, "Visual Basic 4.0 Professional" , ซีเอ็ดยูเคชั่น

[14] ทวีพล ซื่อสัตย์, วิริยะ กองรัตน์, ผ ศ. วิทยา ทิพสุวรรณพร , ผ ศ. สุพรรณ กุลพาณิชย์, " การออกแบบโปรแกรมโฮสคอมพิวเตอร์กับเครื่องควบคุมแบบตรรกจำนวนหลายจุด", การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 19 มหาวิทยาลัยขอนแก่น , 1996 , หน้า cp7- cp14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ผลงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์และได้รับการตีพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

100

การประชุมวิชาการ
ทางวิศวกรรมไฟฟ้า ครั้งที่ 19

คำแนะนำการจัดประชุมโดย

ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์
มหาวิทยาลัยขอนแก่น

7-8 พฤศจิกายน 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การออกแบบทรานสพิวเตอร์โมดูลสำหรับดาวเทียม TMSAT CM-160
Designing the Transputer Module for Thai Micro Satellite (TMSAT)
สมภพ ภูริวิกรัยพงศ์ Mare Fouquet สุเจตน์ จันทรังษ์

4. การออกแบบอุปกรณ์สื่อสารปลายทางในระบบ ISDN มาตรฐานแบบแถบความถี่แคบ CM-165
สำหรับการขอใช้บริการส่งสัญญาณเสียง
Design of Terminal Equipment in Narrow Band Standard ISDN for Speech Communication Service
ธีรพงษ์ สิทธิกุลธร ประสิทธิ์ ทีฆพุฒิ

5. ออปติคอลคับเปลอร์เส้นใยแก้วชนิด FBT CM-78
FBT Optical Fiber Coupler
อธิคม ฤกษ์บุตร

6. สายอากาศทรงกลมปล่อยคลื่นไมโครเวฟสำหรับให้ความร้อน CM-133
A Spherical Applicator for Microwave Hyperthermal
ชาญชัย เลาหะเพ็ญแสง โชคชัย แสงดาว ระวี พรหมหลวงศรี สมผล โกศัลวิตร์ โมไนย ไกรฤกษ์

สาขาวิศวกรรมคอมพิวเตอร์

กลุ่มที่ 1 ประธานกลุ่ม ผศ.ดร.กิติ ลิขิตอนุรักษ์
วันที่ 7 พฤศจิกายน 2539 เวลา 13.00-15.00 น
ห้อง บรรยาย C

1. การสร้างข่ายงานเวฟเล็ตด้วยซอฟต์แวร์ CP-1
Software Implementation of Wavelet Networks
จิระศักดิ์ จันทร์รัตนา วัชรพงษ์ โชวิฑูรกิจ

2. การออกแบบโปรแกรมโฮสคอมพิวเตอร์ กับเครื่องควบคุมแบบตรรกจำนวนหลายจุด CP-7
Support Software for Multiunit Programmable Logic Controllers
ทวีพล ซื่อสัตย์ วิริยะ กองรัตน์ วิทยา ทิพสุวรรณพร สุพรรณ กุลพาณิชย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP-7

การออกแบบโปรแกรมโฮสคอมพิวเตอร์ กับเครื่องควบคุมแบบตรรกจำนวนหลายชุด

(SUPPORT SOFTWARE FOR MULTIUNIT PROGRAMMABLE LOGIC CONTROLLERS)

ทวีหล ซื่อสัตย์ * วิริยะ กองรัตน์ ** ผศ.วิทยา ทิพย์สุวรรณพร** ผศ.สุพรรณ กุลพาณิชย์**

บทคัดย่อ

บทความนี้กล่าวถึงโปรแกรมการทำงานบนเครื่องคอมพิวเตอร์ ใช้เป็นระบบจัดการใน รูปแบบของโฮสคอมพิวเตอร์เพื่อต่อร่วมกับเครื่องควบคุมแบบตรรกที่โปรแกรมได้ จำนวนหลายชุดซึ่งฮาร์ดแวร์ได้ถูกพัฒนาขึ้นมาและเผยแพร่ในเอกสารอ้างอิงหมายเลข (1) โดยผ่านระบบสื่อสารข้อมูลตามมาตราฐาน RS 485 ที่ต่อได้ทั้งแบบจุดต่อจุด และแบบแยกกระจาย จุดประสงค์เพื่อทำการตรวจสอบสภาวะการทำงานของหน่วยความจำต่าง ๆ เพื่อช่วยในการติดตามผล หรือการอ่านเขียนข้อมูล เพื่อจัดเก็บลงในแฟ้มข้อมูล โปรแกรมที่ได้ออกแบบนี้ถูกพัฒนาโดย Visual Basic ที่ทำงานบน Microsoft Windows และมีฟังก์ชั่น อื่น ๆ ที่จำเป็น เพื่ออำนวยความสะดวกในการใช้งาน ที่กล่าวมานี้ จะสามารถกระทำบนเครื่องควบคุมที่ต่อร่วมอยู่ภายใต้ ระบบที่ได้ออกแบบไว้

Abstract

This paper present , soft ware for manage system in host computer . The computer have connect to the programmable logic control , through RS 485 serial communication bus , there are two methord for connecting system are point to point and multidrop

This software use for monitor , write data or upload and download boolean program to save in files which develope by Microsoft Visual Basic for Windows. Purpose for easeful operation with PLC

---

* นักศึกษา ปริญญาโท วิศวกรรมไฟฟ้า สจล.

** อาจารย์ประจำคณะวิศวกรรมศาสตร์

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม สจล.

1.บทนำ

ปัจจุบัน เทคโนโลยีทางด้านอุตสาหกรรม มีการพัฒนาและเปลี่ยนแปลงไปในทางที่สูง มากขึ้น ดังนั้นระบบควบคุมที่มีประสิทธิภาพโดยเฉพาะอย่างยิ่ง ระบบควบคุมสำหรับโรงงานอัตโนมัติ จึงนับได้ว่าเป็นสิ่งที่จำเป็น

เครื่องควบคุมแบบตรรก เป็นเครื่องควบคุมที่ถูกนำมาประยุกต์ใช้ในระบบโรงงานอัตโนมัติ ทั้งนี้เพราะเครื่องควบคุมแต่ละชุด จะควบคุมเครื่องจักร เครื่องกล ที่ถูกจัดวางในที่ต่าง ๆ กัน จึงไม่เป็นการสะดวกที่จะนำคอมพิวเตอร์ มาทำการตรวจสอบสภาวะหรือส่งค่าตัวแปรให้กับเครื่องควบคุม ชุดต่อชุด เป็นการสิ้นเปลืองค่าใช้จ่าย และได้ประสิทธิภาพการทำงานที่ต่ำแต่ถ้านำเอาคอมพิวเตอร์ เพียงเครื่องเดียวแล้วสามารถใช้ส่วนต่อเชื่อมกับเครื่องควบคุม PLC จำนวน หลาย ๆ เครื่องได้ทั้งหมด อำนวยความสะดวกต่อการทำงานได้เป็นอย่างมาก เนื่องจากผู้ควบคุมสามารถตรวจสอบการทำงานของเครื่องจักรกลได้ทุกเครื่องที่มีเครื่องควบคุมการทำงานต่อในระบบนี้อยู่

จึงเป็นเหตุผลให้มีการพัฒนา โปรแกรมบนคอมพิวเตอร์เพื่อให้เป็นโฮสคอมพิวเตอร์ กับเครื่องควบคุม จำนวนหลายชุดในระบบ ขณะเดียวกัน เครื่องควบคุมแต่ละเครื่องก็ยังคงทำงานเป็นอิสระต่อกัน

สำหรับงานวิจัยนี้ เป็นงานวิจัยต่อเนื่องจากบทความเรื่องเครื่องควบคุมแบบตรรกที่โปรแกรมได้ ในการประชุมทางวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 18 ดังนั้น ฮาร์ดแวร์ ของเครื่องควบคุมที่ถูกพัฒนา จะไม่นำมากล่าวทั้งหมด แต่จะกล่าวในส่วนที่เกี่ยวข้องและพัฒนาเพิ่มเติมขึ้นมา คือส่วนของการสื่อสารข้อมูล ที่มีการต่อ ควบคุมแบบแยกกระจาย (MULTIDROP) และการออกแบบโปรแกรมการทำงานบนโฮสคอมพิวเตอร์

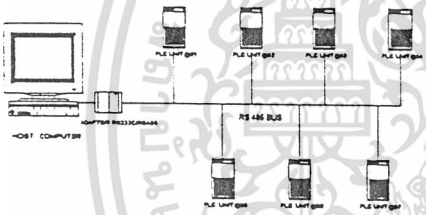
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP-8

2.โครงสร้างของระบบและข้อตกลงในการสื่อสารข้อมูล (PROTOCOL)

2.1ระบบทางด้านฮาร์ดแวร์ของระบบการต่อควบคุมแบบแยกกระจาย (MULTIDROP)

โดยต่อเป็นลักษณะ เครือข่ายกันระหว่างเครื่องควบคุมที่โปรแกรมได้ การสื่อสารข้อมูลกันภายใน จะถูกกระทำผ่านทาง พอร์ท อนุกรม เป็นแบบ RS485 ที่เชื่อมต่อแบบแยกกระจาย โดยเครื่องควบคุมแบบตรรก ถูกวางอยู่ในเครือข่ายได้ทั้งหมด 8 เครื่อง โดยแต่ละเครื่องจะมีตำแหน่งเฉพาะระบุประจำเครื่อง การกำหนดตำแหน่งกระทำได้โดย ปรับสวิทช์ (DIPSWITCHS) ได้ตั้งแต่ หมายเลข 0 ถึงหมายเลข 7 เพื่อวัตถุประสงค์ก็คือ การเรียกตรวจสอบสภาวะจากเครื่องควบคุมทั้งหมดได้ จากโฮสคอมพิวเตอร์ เครื่องเดียว ดังรูป 2-1

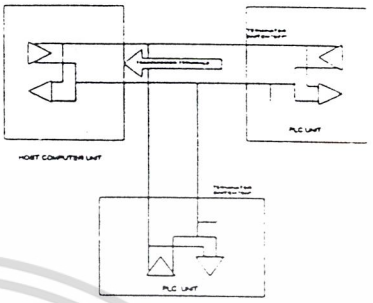


รูปที่2-1แสดงการเชื่อมต่อแบบแยกกระจาย(MULTIDROP)

2.2 ระบบทางด้านฮาร์ดแวร์ เพื่อใช้ในการสื่อสารข้อมูล

หรือติดต่อระหว่าง โฮสคอมพิวเตอร์และเครื่องควบคุมจำนวนหลายจุด

ในระบบฮาร์ดแวร์เดิมมีการสื่อสารข้อมูล แบบอนุกรมตามมาตรฐาน RS232 C ในรูปแบบของสัญญาณแรงดัน จะมีข้อจำกัดอยู่มากคือจะต่อกับคอมพิวเตอร์เพื่อตรวจสอบสภาวะต่าง ๆ ได้ เพียงจุดต่อจุด เท่านั้น และระยะทางไม่ไกลนัก จึงได้ทำการปรับปรุงให้เป็นแบบ RS485 ที่ส่งสัญญาณในรูปแบบของกระแส ทำให้ลดปัญหาเรื่องระยะทางและจำนวนจุดต่อ เพราะการต่อเป็นแบบกระจาย (MULTIDROP)



รูปที่ 2-2 แสดงลักษณะการสื่อสารข้อมูลแบบอนุกรม RS485

2.3 ข้อตกลงในการสื่อสารข้อมูล (PROTOCOL) กับอุปกรณ์ภายนอก

ชุดของข้อมูลในการสื่อสารจะถูกเรียกว่าบล็อค บล็อคของข้อมูลจะถูกส่งจากเครื่องคอมพิวเตอร์ (เครื่องคอมพิวเตอร์ต่อไปจะเรียกว่า HOST) ไปในระบบการเชื่อมต่อ ซึ่งจะเรียกว่าบล็อคคำสั่ง (COMMAND BLOCK) และบล็อคของข้อมูลที่ถูกส่งจากระบบการเชื่อมต่อไปสู่ HOST จะเรียกว่า บล็อคตอบสนอง(RESPONSE BLOCK) ในระบบการเชื่อมต่อสื่อสารแบบหลายจุด แต่ละบล็อค ไม่ว่าจะเป็นบล็อคคำสั่งหรือบล็อคตอบสนองก็ตาม จะเริ่มต้นด้วย อักขระ "@" ตามด้วยตำแหน่งเฉพาะ (UNIT NUMBER) ตามด้วยคำสั่ง (HEADER) ข้อมูล (DATA) และ อักขระ "*" สิ้นสุดด้วยรหัสกำกับบล็อค (FRAME CHECK SCQUENCE CODE : FCS) และรหัสปิดท้ายบล็อคที่เป็นอักขระ [CR]

รูปแบบของบล็อค (BLOCK FORMAT)

@	X	X	X	X	DATA	*	X	X	CR
(ตำแหน่ง)			(คำสั่ง)			FCS		ปิด	

ช่วงคำนวณรหัสกำกับบล็อก

จำนวนอักขระในแต่ละบล็อคทั้งหมดจะต้องไม่เกิน 128 ตัว และช่วงของการคำนวณเพื่อหา รหัสกำกับบล็อคจะอยู่ระหว่างอักขระเริ่มต้น (@)ไปจนถึงสิ้นสุด DATA ที่เป็น "*"

การคำนวณหารหัสกำกับบล็อค (FCS CALCULATION)

FCS เป็นข้อมูลขนาด 8 บิท ที่ถูกแปลงไปเป็นอักขระ ASCII และ 8 บิทดังกล่าวนี้เป็นผลลัพธ์จากการ EXCLUSIVE OR [XOR] กันเป็นลำดับของ แต่ละอักขระ ในบล็อค เริ่ม จากอักขระแรกจนถึงอักขระสุดท้ายของDATAเช่น {@01RL00000 010*74 [CR] }

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP-9

ความหมายคือ อ่านพื้นที่ LINK RELAY เริ่มที่ตำแหน่ง 0000 อ่านมาจำนวน 10 ตำแหน่ง

อักขระ	รหัส(ASCII BINARY) (HEX)	XOR (HEX)
@	0100 0000 [40]	
0	0011 0000 [30]	112 [70]
1	0011 0001 [31]	65 [41]
R	0101 0010 [52]	19 [13]
L	0100 1100 [4C]	95 [5F]
0	0011 0000 [30]	111 [6F]
0	0011 0000 [30]	95[5F]
0	0011 0000 [30]	111[6F]
0	0011 0000 [30]	95 [5F]
0	0011 0000 [30]	111 [6F]
0	0011 0000 [30]	95 [5F]
1	0011 0001 [31]	110 [6E]
0	0011 0000 [30]	94 [5E]
*	0010 1010 [2A]	116 [74]

ตัวอย่างโปรแกรมในการคำนวณรหัสกำกับบล็อก

```
Sub fcs()
        Dim a,b,c
        a = "@ 01 RL 00000001*"
        b = Len(a)
        c = 0
        for b= 0 to b - 1
           c = c Xor Asc (Mid (a,b+1,1))
        Next b
        fcs = HeX (c)
end Sub
```

คำสั่งและคำตอบสนอง

ในขณะที่ทำการเชื่อมต่อเพื่อสื่อสารข้อมูล HOST สามารถที่จะทำการเฝ้ามองการดำเนินการและสามารถที่จะทำการควบคุมเครื่องที่โปรแกรมได้ ถ้าอยู่ในสถานะการเฝ้ามอง HOST จะต้องส่งคำสั่งไปถามตามชนิดของข้อมูลที่ต้องการ แต่เครื่องควบคุม หรือถ้าในสถานะการควบคุมก็สมารถที่จะส่งคำสั่งไปทำการเปลี่ยนแปลงค่าของข้อมูลที่อยู่ในหน่วยความจำโดยตรง เช่น ข้อมูลของ อินพุท/เอาท์พุท เป็นต้น

เวลาของการตอบสนองจะแปรไปขึ้นอยู่กับความเร็วในการส่งผ่านข้อมูล จำนวนของข้อมูล และเวลาในการสแกนของเครื่องควบคุม และถ้าเวลาในการสื่อสารมากขึ้นก็เป็นผลให้เวลาในการสแกนมากขึ้นตามไปด้วย ต่อไปจะเป็นคำสั่งต่าง ๆ ที่เป็นข้อกำหนด (Protocol) ที่ใช้ในการสื่อสาร

บล็อกคำสั่งประกาศ

บล็อคคำสั่งประกาศถูกส่งโดย HOST ไปเพื่อตรวจสอบว่ามีเครื่องควบคุมอยู่ในการเชื่อม ต่อหรือในเครือข่ายหรือไม่

รูปแบบคำสั่ง " @[Unit Number]AT*[FCS][CR] "

เช่น BLOCK COMMAND เป็น "@01AT*7E[CR]" หมายถึง HOST ส่งออกไปเพื่อ ประกาศ ถามถึงเครื่องควบคุมที่อยู่ในการเชื่อมต่อมีตำแหน่งเฉพาะที่ 01 หรือไม่ ถ้าเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมมีจริงและได้รับคำประกาศ ดังกล่าวก็จะส่งคำตอบสนองออกไป ดังนี้

รูปแบบคำตอบสนอง "@[Unit Number]PC84*[FCS][CR] "

เช่น BlockResponseเป็น "@01PC84*74[CR]" หมายถึงเครื่องควบคุมตำแหน่ง เฉพาะที่ 01เท่านั้นที่ตอบออกไป ซึ่งจะแสดงให้ HOST ทราบว่าเครื่องควบคุมที่ตำแหน่ง 01 ในระบบการเชื่อมต่อยังคงทำงานปกติ

บล็อกคำสั่งอ่านพื้นที่ Input/Output/Internal Relay

รูปแบบคำสั่ง " @[Unit Number]RI[ตำแหน่งเริ่มต้น][จำนวนข้อมูล]*[FCS][CR] "

เช่นถ้า HOST ต้องการอ่านพื้นที่ของ I/O หรือ Internal Relay ของเครื่องควบคุมที่วงอยู่ ใน ระบบการเชื่อมต่อ ตำแหน่งเฉพาะที่ 02 ตำแหน่ง I/O ที่ 0010 มาจำนวน 5 ตำแหน่ง ก็สามารถจัด BlockCommand ได้ดังนี้ "@02RI00100005"XX[CR]" (FCS XX : ASCII 2 digit)

รูปแบบคำตอบสนอง " @[Unit Number]RI[ตำแหน่งเริ่มต้น][ข้อมูลXX(1)][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

เช่น ถ้าคำตอบสนอง ของเครื่องควบคุม ในตำแหน่งเฉพาะ ที่02ตอบมาเป็น "@02RI0010234455566AB*XX[CR]" ก็หมายความว่า ตำแหน่งหน่วย ความจำของ I/O หรือ Internal Relay ที่ 0010 มีข้อมูล 23,44,55,66,AB [ฐาน16] ตามลำดับ

บล็อกคำสั่งเขียนพื้นที่ Input/Output/Internal Relay

รูปแบบคำสั่ง "@[Unit Number]WI[ตำแหน่งเริ่มต้น][ข้อมูลXX(1)][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

เช่นต้องการเขียนข้อมูลลงในตำแหน่ง I/O หรือ Internal Relay ของเครื่องควบคุมที่ตำแหน่งเฉพาะ 03 ในตำแหน่งที่ 0010 ด้วยข้อมูล 12,34,56,78,9A จำนวน 5 ตำแหน่ง ก็สามารถจัด BlockCommand ได้ดังนี้ "@03WI00101234567 89A*XX[CR]"

รูปแบบคำตอบสนอง" @[Unit Number]WI[รหัสตอบสนอง]*[FCS][CR]"

(XX :Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@03WI00*XX[CR]"

บล็อกคำสั่งอ่านค่าเป้าหมาย Timer

เพื่อสามารถให้ Host รู้ค่าเป้าหมาย (Set Value) ที่ได้ตั้งไว้ที่ตัวเวลา

รูปแบบคำสั่ง "@[Unit Number]RT[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR] "

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวตั้งเวลาที่ 12 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด BlockCommand ได้ดังนี้ "@04RT00120001*XX[CR]"

รูปแบบคำตอบสนอง "@[Unit Number]RT[ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้ค่า ตอบสนองออกมาสมมุติว่าค่าเป้าหมายของตัวตั้งเวลาตำแหน่งที่ 12 มีค่า #0150 และ Block Response เป็น "@04RT00120150*XX[CR] "

บล็อกคำสั่งอ่านค่าเป้าหมาย Counter

เพื่อสามารถให้ Host รู้ค่าเป้าหมาย (Set Value) ที่ได้ตั้งไว้ที่ตัวตั้งนับ

รูปแบบคำสั่ง "@[Unit Number]RC[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR]

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวนับที่ 10 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด BlockCommand ได้ดังนี้ "@04RT00100001*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit Number]RC[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR] "

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวนับตำแหน่งที่ 10 มีค่า #8000 และ BlockResponse เป็น "@04RT0010800*XX[CR]"

บล็อกคำสั่งเขียนค่าเป้าหมาย Timer

เพื่อที่ Host สามารถตั้ง แก้ไขค่าเป้าหมายใหม่ของตัวตั้งเวลาแก่เครื่องควบคุมที่อยู่ในระบบเชื่อมต่อได้ตามต้องการ

รูปแบบคำสั่ง"@[Unit Number]WT[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นถ้าต้องการตั้งค่าเป้าหมายแก่ตัวตั้งเวลาของเครื่องควบคุมที่วางในระบบเชื่อมต่อที่มีตำแหน่งเฉพาะที่ 05 และตัวตั้งเวลาตำแหน่งที่ 00 ด้วยค่าเวลา #0200 สามารถจัด BlockCommand ได้ดังนี้ "@05WT0000200*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit Number]WT[รหัสตอบสนองXX]*[FCS][CR]"

(XX : Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@05WT00*XX[CR]"

บล็อกคำสั่งเขียนค่าเป้าหมาย Counter

ลักษณะของข้อกำหนดจะเหมือนกับเขียนค่าเป้าหมายให้กับตัวตั้งเวลา ต่างกันที่ Header

รูปแบบคำสั่ง"@[Unit Number]WC[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

เช่นถ้าต้องการตั้งค่าเป้าหมายแก่ตัวนับของเครื่องควบคุมที่วางในระบบเชื่อมต่อที่มีตำแหน่งเฉพาะที่ 05 และตัวตั้งเวลาตำแหน่งที่ 00 ด้วยคำนับ #0200 สามารถจัด BlockCommand ได้ดังนี้ "@05WC00000200*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit number]WC[รหัสตอบสนองXX]*[FCS][CR]"

(XX : Response Code)

00 = Data Complete

08 = Data Error

และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@05WC00*XX[CR]"

บล็อกคำสั่งอ่านค่าปัจจุบัน Timer

เพื่อที่ Host สามารถที่จะทำการเฝ้ามองความเป็นไปของค่าเวลาที่กำลังทำงานของตัวตั้งเวลาในเครื่องควบคุมตำแหน่งเฉพาะใด ๆ ที่วางในระบบเชื่อมต่อได้

รูปแบบคำสั่ง"@[Unit Number]PT[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR]

เช่นถ้าต้องการอ่านค่าเป้าหมายของตัวตั้งเวลาที่ 12 และ 13 ของเครื่องควบคุมที่อยู่ในตำแหน่งเฉพาะที่ 04 สามารถจัด BlockCommand ได้ดังนี้ "@04PT00120002*XX[CR]"

รูปแบบคำตอบสนอง"@[Unit Number]PT[ตำแหน่งเริ่มต้น][ข้อมูลXX(..)][ข้อมูลXX(n)]*[FCS][CR]"

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวตั้งเวลาตำแหน่งที่ 12 มีค่า #0150 และตัวตั้งเวลาตำแหน่งที่ 13 มีค่า #0200 BlockResponse เป็น "@04rt001201500200*XX[CR]"

อ่านค่าปัจจุบัน Counter

รูปแบบคำสั่ง"@[Unit Number]PC[ตำแหน่งเริ่มต้น][จำนวน]*[FCS][CR]"

และถ้าเครื่องควบคุมในตำแหน่งเฉพาะที่ 04 รับคำสั่งได้จากระบบเชื่อมต่อ ก็จะให้คำตอบสนองออกมา สมมุติว่าค่าเป้าหมายของตัวนับตำแหน่งที่ 10 มีค่า #8000 และ BlockResponse เป็น "@04pt00108000*XX[CR]"

อ่านจากพื้นที่โปรแกรม UpLoad Program

เพื่อให้ Host สามารถที่จะนำข้อมูลส่วนโปรแกรมบูลลีนที่ผู้ใช้ได้ทำการโปรแกรมไว้ในหน่วยความจำนั้นขึ้นมาที่ Host เพื่อทำการเก็บรักษาไว้

รูปแบบคำสั่ง"@[Unit Number]UL*[FCS][CR]"

เช่นถ้า Host ต้องการนำโปรแกรมส่วนบูลลีนจากเครื่องควบคุมที่มีตำแหน่งเฉพาะเป็น 01 มาจากจุดเริ่มโปรแกรมถึงคำสั่งสุดท้าย (END) มาสามารถจัด BlockCommand ได้เป็น "@01UL*72[CR]" และเครื่องควบคุมที่ตำแหน่งเฉพาะ 01 จะให้คำตอบสนองออกมาเป็นชุดข้อมูลที่ละ 20 Bytes เริ่มที่ตำแหน่งในหน่วยความจำที่ 8000 และชุดต่อไปของข้อมูลจะมีตำแหน่งเริ่มต้นที่ สอดคล้องกับตำแหน่งที่ถูกถ่ายเทขึ้นมาด้วย หรือจนกว่าจะพบคำสั่ง END ดังรูปแบบดังต่อไปนี้ ข้อสังเกต **รูปแบบคำตอบสนองนั้นจะมีลักษณะเหมือนกับคำสั่งในการเขียนลงพื้นที่โปรแกรม ทั้งนี้ก็เพื่อที่จะ** ให้สามารถนำชุดข้อมูลดังกล่าวนั้นเขียนลงไปใหม่ที่ส่วนโปรแกรมได้เลยโดยไม่ต้องเปลี่ยนแปลงรูปแบบ

รูปแบบคำตอบสนอง"@[Unit Number]UL[ตำแหน่งเริ่ม8000][ข้อมูลXX(..)][ข้อมูลXX(20)]*[FCS][CR]"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP-12

เขียนลงพื้นที่โปรแกรม DownLoad Program เพื่อให้ Host สามารถทำการส่งผ่านข้อมูลส่วนโปรแกรมบูลลีนเข้าไปไว้ในพื้นที่โปรแกรมของเครื่องควบคุมที่มีตำแหน่งเฉพาะนั้น ๆ ได้

รูปแบบคำสั่ง "@[Unit Number]DL[ตำแหน่งเริ่ม8000][ข้อมูลXX(..)][ข้อมูลXX(20)]*[FCS][CR]"

ข้อมูลของโปรแกรมบูลลีนที่ต้องการเขียนลงในหน่วยความจำส่วนโปรแกรมนั้นก็จะต้องไม่มากกว่า 20 Bytes เช่นเดียวกัน และตำแหน่งที่ต้องการวางลงในหน่วยความจำก็สามารถกำหนดได้ (ตำแหน่งในหน่วยความจำส่วนโปรแกรมบูลลีนเริ่มที่ 8000)

รูปแบบคำตอบสนอง"@[Unit Number]DL[รหัสตอบสนองXX]*[FCS][CR]"

(XX : Response Code)

00 = Data Complete

08 = Data Error

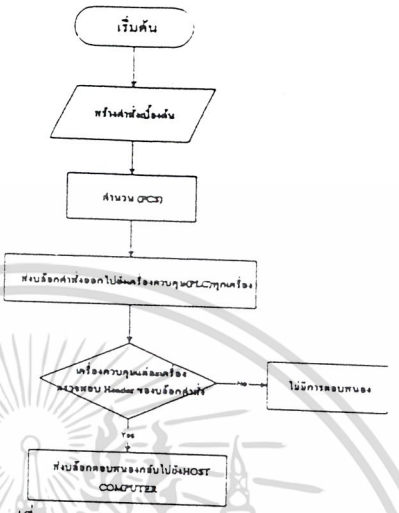
และถ้าเรียบร้อยก็จะตอบออกมาเป็น "@01DL00*XX[CR]"

3)แนวคิดและการออกแบบโปรแกรม

ในการเขียนโปรแกรมคำสั่งบูลลีนลงในเครื่องควบคุม PLC จะประสบปัญหาในเรื่องของการจัดเก็บข้อมูลลงในหน่วยความจำสำรอง และการตรวจสอบสภาวะการทำงาน คือไม่สามารถตรวจตรวจสอบสภาวะของหน่วยความจำต่าง ๆ ได้ครั้งละหลาย ๆ ตำแหน่งเพื่อแก้ไขปัญหาจึงได้ออกแบบโปรแกรมเพื่อรองรับความต้องการ โดยจัดแบ่งการติดต่อสื่อสารระหว่างเครื่องควบคุมกับโฮสคอมพิวเตอร์ ซึ่งการออกแบบจะได้แสดงดังต่อไปนี้

3.1 การส่งบล็อคคำสั่งจากโฮสคอมพิวเตอร์ไปยังเครื่องควบคุม PLC

โฮสคอมพิวเตอร์จะส่งบล็อคคำสั่งออกไปทางพอร์ทอนุกรมของคอมพิวเตอร์ ผ่านเครื่องแปลงระบบ RS232 เป็น RS 485 ไปยังเครื่องควบคุม PLCแต่ละเครื่อง เครื่องควบคุมจนตรวจสอบหัวของคำสั่ง หรือ หมายเลขประจำเครื่อง ถ้าตรงกันก็จะส่งบล็อคตอบสนองกลับไปยังโฮสคอมพิวเตอร์ซึ่งหัวของคำสั่ง (Header) นี้กำหนดได้จากโปรแกรมบนคอมพิวเตอร์



รูปที่3-1แสดงผังการทำงานการส่งบล็อคคำสั่งโฮสคอมพิวเตอร์

3.2 การรับบล็อคตอบสนองจากเครื่องควบคุม (PLC) ของโฮสคอมพิวเตอร์ เพื่อนำมาแสดงผล

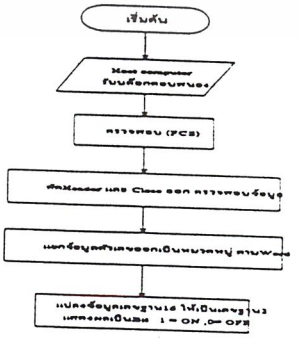
หลังจากที่ส่งบล็อคคำสั่ง เครื่องควบคุม (PLC) ที่มีหมายเลขระบุตำแหน่ง ตรงกันก็จะส่ง บล็อคตอบสนอง กลับมา ยกตัวอย่างการตรวจสอบสภาวะของ OUTPUT

@	X	X	X	X	DATA	*	X	X	CR
(ตำแหน่ง)		(คำสั่ง)				FCS		ปิด	

โปรแกรมจะตรวจสอบจนได้ข้อมูลที่ถูกต้องแล้ว ทำการ ตัดส่วนหัว [@,UNIT NO, COMMAND] และส่วนท้าย [*,FCS) ออกเหลือเพียงส่วนที่เป็นข้อมูล (DATA)

ส่วนข้อมูล (DATA) จะประกอบด้วย WORD ที่เริ่มต้น และตามด้วยข้อมูลที่เป็นเลขฐาน 16 WORD 2 หลัก (8 bit) จากนั้นก็นำข้อมูลมาแปลงเป็นเลขฐาน 2 แสดงผลออกเป็น bit ในแต่ละ word ซึ่งพอจะสรุปเป็นผังการทำงานดังนี้

รูปที่ 3 - 2 แสดงผังการทำงานของการแสดงผลข้อมูล



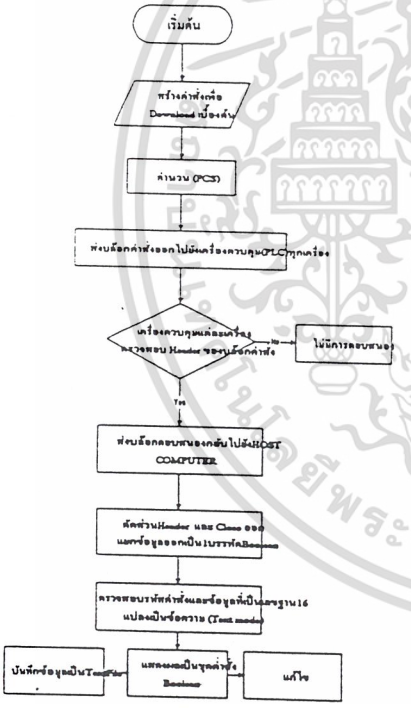
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP-13

3.3 การนำคำสั่งโปรแกรมบูลลีนจากเครื่องควบคุม (PLC) มาแก้ไขหรือบันทึกลงบนโฮสคอมพิวเตอร์

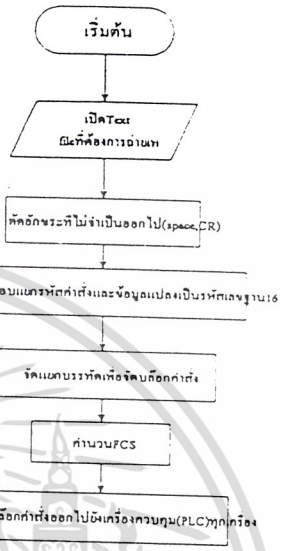
คือการส่งบล็อคคำสั่งเพื่อให้ PLC ส่งข้อมูลที่เป็นรหัสเลขฐาน 16 มายังโฮสคอมพิวเตอร์ จากนั้นโฮสคอมพิวเตอร์มีหน้าที่แปลงรหัสต่าง ๆ ให้เป็นข้อความ (Text mode) เพื่อแสดงผลบนหน้าจอ และสามารถที่จะแก้ไขหรือบันทึกข้อมูลต่าง ๆ เป็น แบบ Text File

ตัวอย่างการแปลงเลขรหัสเลขฐาน 16 ให้เป็นข้อความเช่น "80 0001"จะแยกออกเป็น 2 ส่วนคือ 2 ไบท์ แรก(80) เป็นคำสั่ง 4 ไบท์ ที่เหลือเป็นข้อมูล เมื่อ 80 = LD จะได้ว่า LD 0001 สำหรับรหัสต่าง ๆ ของคำสั่งบูลลีนนั้นอ้างอิงตามเอกสารอ้างอิงหมายเลข[1]



รูปที่ 3 - 3 แสดงผังการแปลงข้อมูลจาก PLC เป็นข้อความ

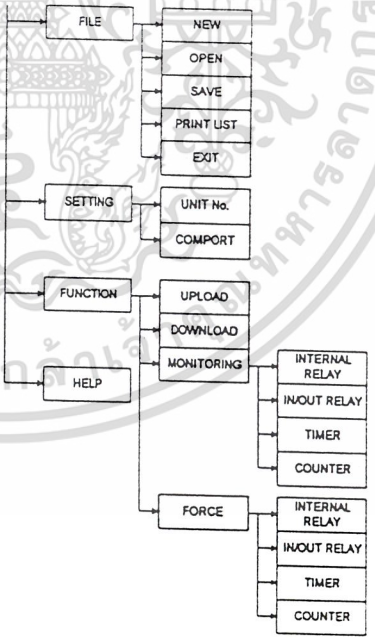
3.4 การนำโปรแกรมบูลลีนที่บันทึกไว้บนคอมพิวเตอร์ เพื่อถ่ายเทให้กับเครื่องควบคุม (PLC)



รูปที่ 3 - 4 แสดงผังการถ่ายเทโปรแกรมลงเครื่องควบคุม PLC

4 ผลการทดลอง

4.1 แผนผังของโปรแกรมที่ออกแบบ

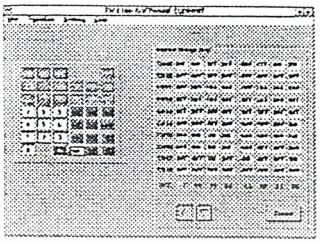


รูปที่ 4-1 แสดงแผนผังของโปรแกรม

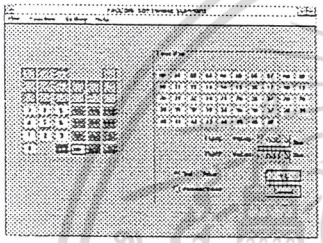
4.2 การตรวจสอบสภาวะของ รีเลย์ ตัวตั้งเวลา และตัวนับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

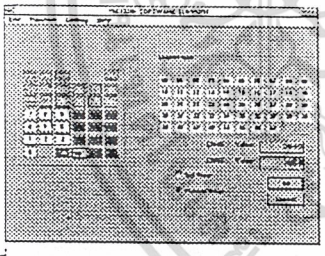
CP-14



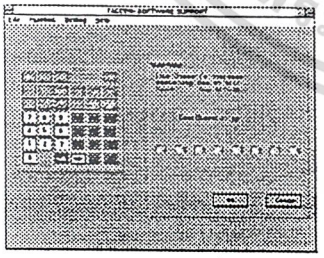
รูปที่ 4-2 สดงการตรวจสอบสภาวะของรีเลย์ภายใน



รูปที่ 4-3 แสดงการตรวจสอบ สภาวะของตัวตั้งเวลา

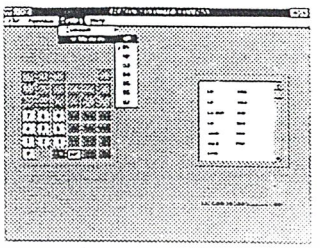


รูปที่ 4-4 แสดงการตรวจสอบสภาวะของตัวนับ



รูปที่ 4-5 แสดงการเขียนค่าลงในรีเลย์ INPUT และ OUTPUT

4.3 การแสดงโปรแกรมคำสั่งบูลลีนที่รับมาจากเครื่องคุม



รูปที่4-6 แสดงคำสั่งบูลลีนบนเครื่องคอมพิวเตอร์

5. บทสรุป

จากการศึกษาและทดลองใช้งาน ให้ผลการทำงาน
เป็นที่น่าพอใจในระดับหนึ่ง และมีแนวโน้มที่จะพัฒนา
ประสิทธิภาพของโปรแกรมให้สูงขึ้นต่อไปเพื่อความสะดวกใน
การเขียนโปรแกรมลงบนเครื่องควบคุม PLC เช่นเขียน
โปรแกรมเป็นบล๊อกการทำงานแล้ว แปลงเป็นชุดคำสั่งบูลลีน
เป็นต้น

โปรแกรมนี้ใช้ได้เฉพาะกับฮาร์ดแวร์ที่สร้างขึ้นเท่านั้น
แต่สามารถนำหลักการนี้ไป ใช้กับฮาร์ดแวร์อื่นได้ไม่ยาก

เอกสารอ้างอิง

[1] วิริยะ กองรัตน์, ผ.ศ. วิทยา ทิพย์สาวรรณพร, ผ.ศ. สุพรรณ กุลพาณิชย์, " เครื่องควบคุมแบบตรรกที่โปรแกรมได้ " ,การประชุมวิชาการทางวิศวกรรมทางไฟฟ้า ครั้งที่ 18 , มหาวิทยาลัยเทคโนโลยีมหานคร , 1995

[2] Perter Hamann, Steve Willings " MINI - H-TYPE PROGRAMMABLE LOGIC CONTROLLERS OPERATION MANUAL " , July 1990

[3] Joe Campbell , C Programmer ' s Guide to Serial Communications Second Edition , Sams Publishing, 1994

[4] Paul Bonner , PC MAGAZINE VISUAL BASIC UTILITYS , Ziff Daus press emeru ville, California , 1993

[5] Mark S. Burgess , ADVANCED VISUAL BASIC , Addison - Wesley Publishing Co. ltd., 1994

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

รายละเอียดข้อมูลไอซีเบอร์ Z84C11 และ เบอร์ 75176

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

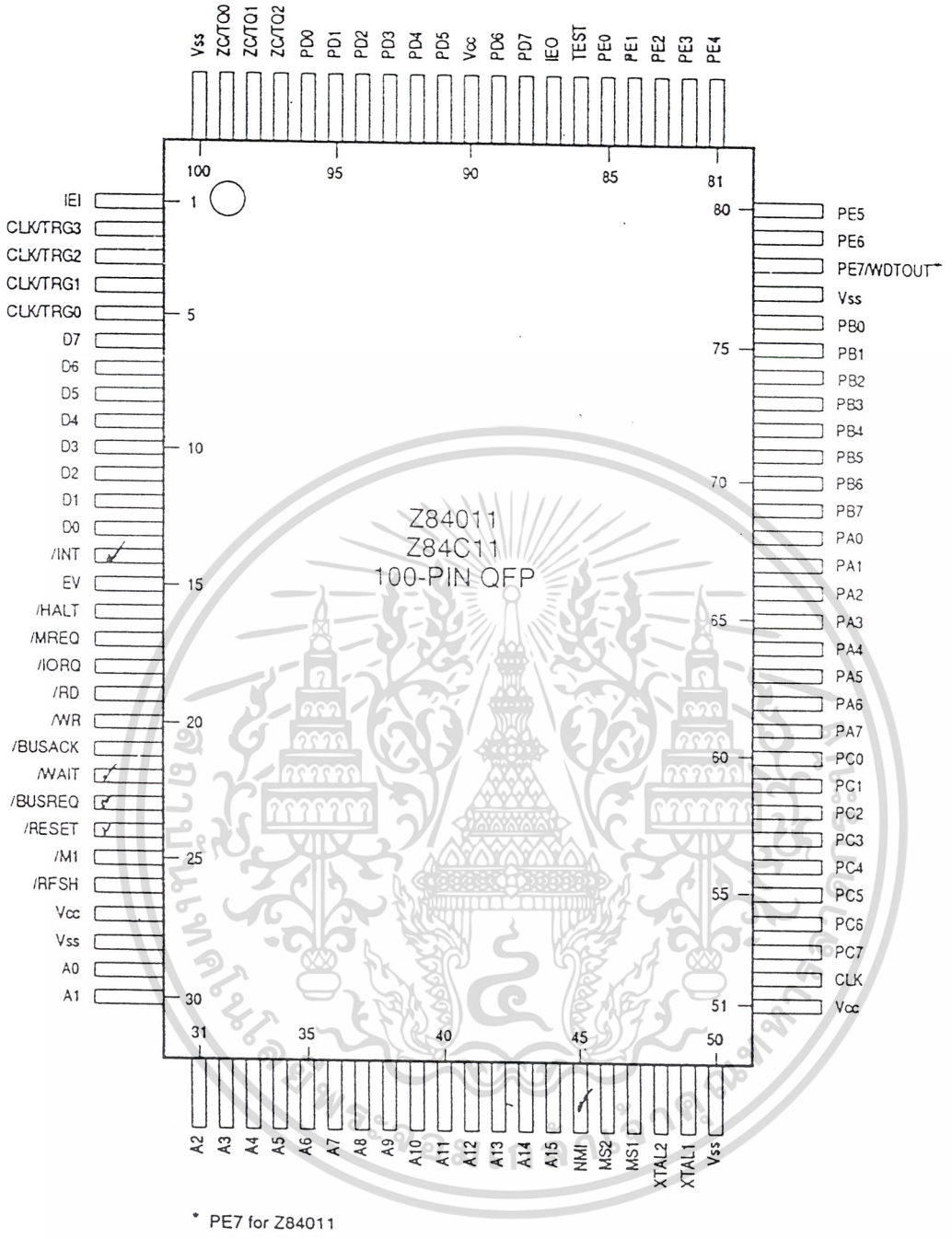
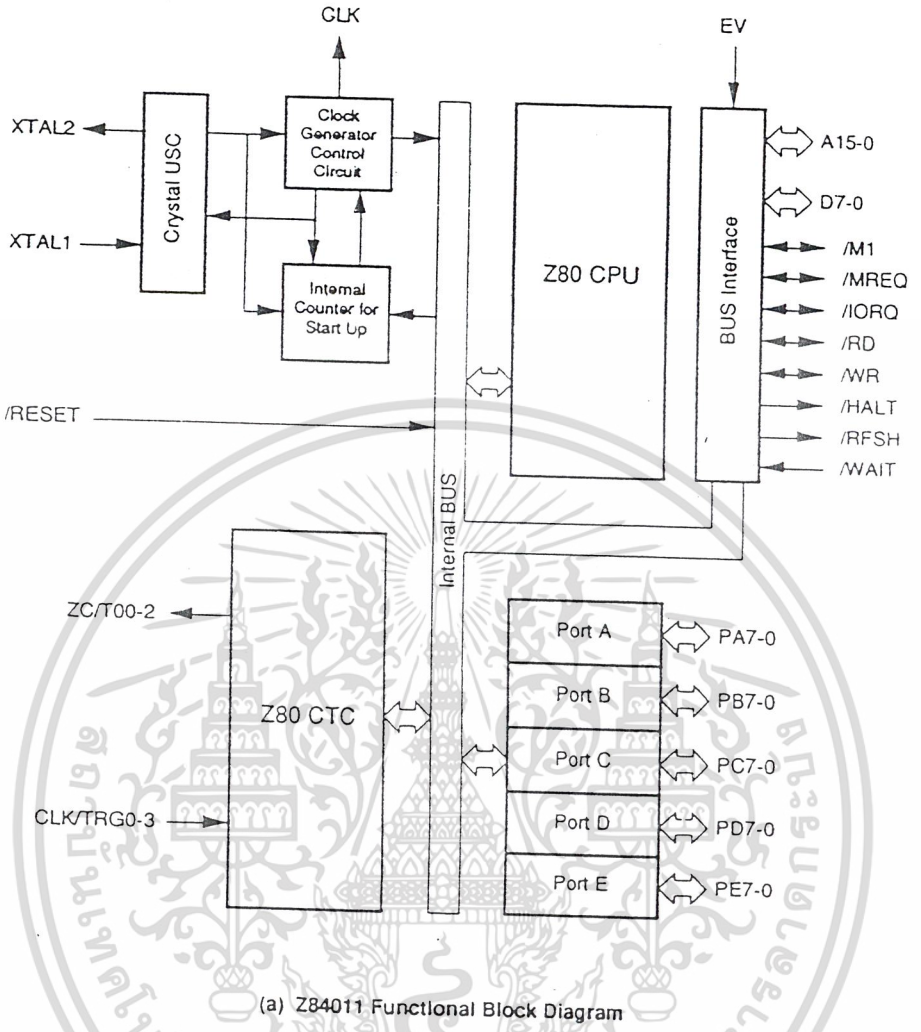


Figure 2. PIC Pin Assignment

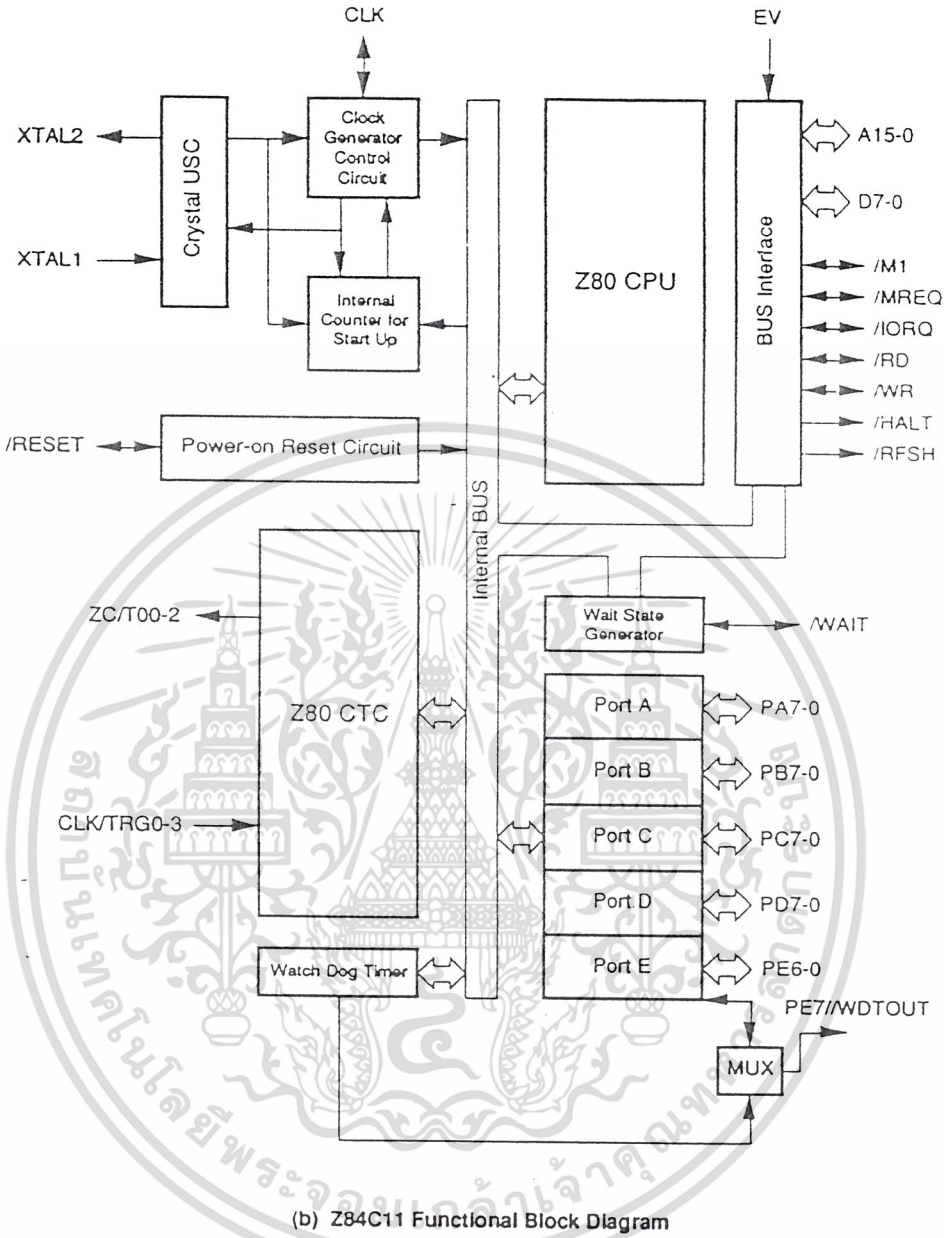
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(a) Z84011 Functional Block Diagram

Figure 1. PIC Functional Block Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(b) Z84C11 Functional Block Diagram

Figure 1. PIC Functional Block Diagram (Continued)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

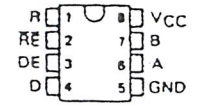
ADVANCE INFORMATION

SN75176B DIFFERENTIAL BUS TRANSCEIVER

D2618, JULY 1985

- Bidirectional Transceiver
- Meets EIA Standards RS-422A and CCITT Recommendations V.11 and X.27
- Designed for Multipoint Transmission on Long Bus Lines in Noisy Environments
- 3-State Driver and Receiver Outputs
- Individual Driver and Receiver Enables
- Wide Positive and Negative Input/Output Bus Voltage Ranges
- Driver Output Capability . . . ±60 mA Max
- Thermal Shutdown Protection
- Driver Positive and Negative Current Limiting
- Receiver Input Impedance . . . 12 kΩ Min
- Receiver Input Sensitivity . . . ±200 mV
- Receiver Input Hysteresis . . . 50 mV Typ
- Operates from Single 5 Volt Supply
- Low Power Requirements

D, JG, OR P
DUAL-IN-LINE PACKAGE
(TOP VIEW)



FUNCTION TABLE (DRIVER)

INPUT	ENABLE	OUTPUTS	
D	DE	A	B
H	H	H	L
L	H	L	H
X	L	Z	Z

FUNCTION TABLE (RECEIVER)

DIFFERENTIAL INPUTS	ENABLE	OUTPUT
A − B	RE	R
$V_{ID} \geq 0.2$ V	L	H
$0.2\text{ V} < V_{ID} < 0.2$ V	L	?
$V_{ID} \leq -0.2$ V	L	L
X	H	Z

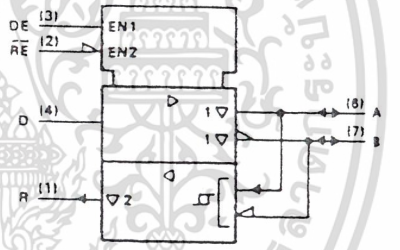
H = high level, L = low level, ? = indeterminate,
X = irrelevant, Z = high impedance (off)

description

The SN75176B differential bus transceiver is a monolithic integrated circuit designed for bidirectional data communication on multipoint bus transmission lines. It is designed for balanced transmission lines and meets EIA Standard RS-422A and CCITT Recommendations V.11 and X.27.

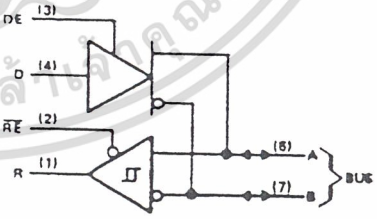
The SN75176B combines a three-state differential line driver and a differential input line receiver both of which operate from a single 5-volt power supply. The driver and receiver have active-high and active-low enables, respectively, that can be externally connected together to function as direction control. The driver differential outputs and the receiver differential inputs are connected internally to form differential input/output (I/O) bus ports that are designed to offer minimum loading to the bus whenever the driver is disabled or $V_{CC} = 0$ volts. These ports feature wide positive and negative common-mode voltage ranges making the device suitable for party-line applications.

logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

logic diagram (positive logic)



ADVANCE INFORMATION documents contain information on new products in the sampling or preproduction phase of development. Characteristic data and other specifications are subject to change without notice.

Copyright © 1985, Texas Instruments Incorporated

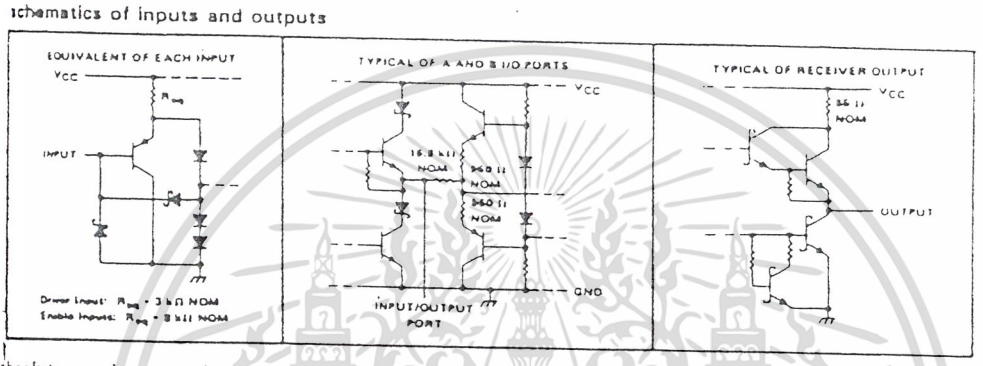
TEXAS INSTRUMENTS
INCORPORATED
POST OFFICE BOX 5012 • DALLAS, TEXAS 75222

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The driver is designed to handle loads up to 60 milliamperes of sink or source current. The driver features positive- and negative-current limiting and thermal shutdown for protection from line fault conditions. Thermal shutdown is designed to occur at a junction temperature of approximately 150°C. The receiver features a minimum input impedance of 12 kΩ, an input sensitivity of ±200 millivolts, and a typical input hysteresis of 50 millivolts.

The SN751768 can be used in transmission line applications employing the SN75172 and SN75174 quadruple differential line drivers and SN75173 and SN75175 quadruple differential line receivers.

schematics of inputs and outputs



absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1)	7 V
Voltage at any bus terminal	10 V to 15 V
Enable input voltage	5.5 V
Continuous total dissipation at (or below) 25°C free-air temperature (see Note 2):	
D Package	725 mW
JG Package	825 mW
P Package	1000 mW
Operating free-air temperature range	0°C to 70°C

NOTES: 1. All voltage values, except differential input/output bus voltage, are with respect to network ground terminal.
2. For operation above 25°C free-air temperature, derate the D package to 464 mW at 70°C at the rate of 5.8 mW/°C, derate the JG package to 528 mW at 70°C at the rate of 6.6 mW/°C, and derate the P package to 640 mW at 70°C at the rate of 8.0 mW/°C. In the JG package, SN75176B chips are glass mounted.

recommended operating conditions

		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.75	5	5.25	V
Voltage at any bus terminal (separately or common mode), V_I or V_{IC}		−7[†]		12	V
High-level input voltage, V_{IH}	D, DE, and $\overline{RE}$	2			V
Low-level input voltage, V_{IL}	D, DE, and $\overline{RE}$			0.8	V
Differential input voltage, V_{ID} (see Note 3)				±12	V
High-level output current, I_{OH}	Driver			−60	mA
	Receiver			−400	μA
Low-level output current, I_{OL}	Driver			60	mA
	Receiver			8	
Operating free-air temperature, T_A		0		70	°C

[†] The algebraic convention, where the less positive (more negative) limit is designated minimum, is used in this data sheet for common-mode input voltage and threshold voltage levels only.

NOTE 3: Differential input/output bus voltage is measured at the noninverting terminal A with respect to the inverting terminal B.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL APPLICATION

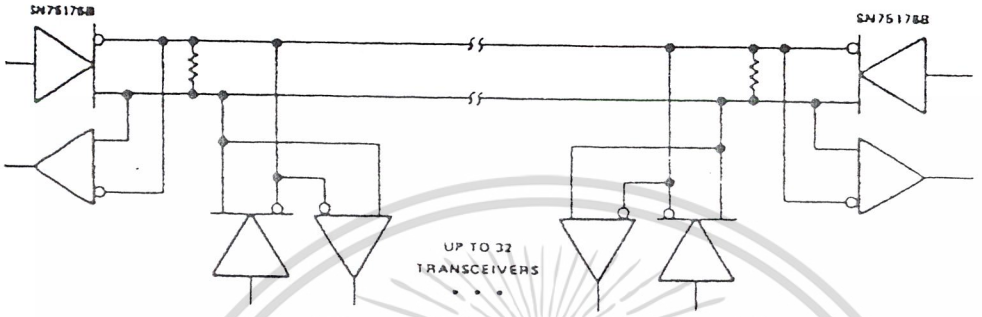


FIGURE 15. TYPICAL APPLICATION CIRCUIT

NOTE 7: The line should be terminated at both ends in its characteristic impedance. Stub lengths off the main line should be kept as short as possible.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ZiLOG

PRODUCT SPECIFICATION

Z84011/C11

PARALLEL I/O CONTROLLER

FEATURES

- Z84C00 Z80 CPU with CGC, Z84C30 Z80 CTC, five 8-bit parallel ports.
- High speed operation (6/10 MHz).
- Low power consumption in four operation modes:
  - 45 mA Typ. (Run mode).
  - 6 mA Typ. (Idle1 mode).
  - 9 mA Typ. (Idle2 mode; Not applicable to Z84011).
  - 1 µA Typ. (Stop mode).

  Wide operational voltage range (5V ±10%).
- TTL/CMOS compatible.
- Z84011 features:
  - Z84C00 Z80 CPU.
  - On-chip four channel Counter Timer Controller (Z80 CTC).
  - Built-in Clock Generator Controller (CGC).
  - Five 8-bit parallel ports.
  - 100-pin QFP Package.
  - Noise filter to CLK/TRG inputs of the Z80 CTC.
- Z84C11 features:
  - All Z84011 features.
  - Support of Idle 2 Mode.
  - Built-in Watch Dog Timer (WDT).
  - Power-on Reset and Reset Extension.
  - Wait State Generator.
  - Simplified EV Mode Selection.
  - Crystal Divide-by-One Option.
  - External Clock Input Option.

GENERAL DESCRIPTION

The Z84011 and Z84C11 Parallel I/O Controllers (PIC) are CMOS 8-bit microprocessors. They are integrated with the CTC and five 8-bit parallel ports into a single 100-pin QFP (Quad Flat Pack) package. The Z84C11 is an upward compatible version of the Z84011. Figure 1(a) shows the block diagram of Z84011, and Figure 1(b) shows the block diagram of the Z84C11. Figure 2 has pin assignments for both versions. These high end superintegrated Parallel I/O Controllers are targeted for a broad range of applications ranging from embedded controller to enhancement/cost reductions of existing hardware using Z80 based discrete peripherals.

Hereinafter, the word PIC on the description covering both versions (Z84C11 and Z84011) is used. Use Z84C11 on the description which applies only to the Z84C11, and use Z84011 which applies only to the Z84011.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

ตัวอย่างการพิมพ์ชุดคำสั่งบูลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Instruction	Area	Timer/Counter Data
0000	LD	0001	
0001	LD	0702	
0002	LATCH	1003	
0003	LD	1003	
0004	TIM	0000	0020
0005	LDTIM	0000	
0006	OR	0000	
0007	ANDNOT	0002	
0008	OUT	0702	
0009	LD	0002	
0010	LD	1003	
0011	ANDNOTTIM	0000	
0012	ORLD		
0013	OUT	0701	
0014	LD	0002	
0015	TIM	0001	0030
0016	LDTIM	0001	
0017	OR	0700	
0018	ANDNOTTIM	0002	
0019	OUT	0703	
0020	TIM	0003	0050
0021	LD	0003	
0022	TIM	0002	0015
0023	LD	0000	
0024	OR	0703	
0025	ORTIM	0003	
0026	ANDNOT	0003	
0027	ANDNOT	0702	
0028	OUT	0703	
0029	END		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Instruction	Area	Timer/Counter Data
0000	LD	0000	
0001	OR	0800	
0002	ANDNOT	0004	
0003	ANDNOTTIM	0000	
0004	ANDNOT	1000	
0005	OUT	0800	
0006	LDTIM	0000	
0007	OR	0706	
0008	ANDNOT	0001	
0009	ANDNOT	0707	
0010	OUT	0706	
0011	LD	0001	
0012	AND	0003	
0013	OR	0705	
0014	ANDNOT	0002	
0015	AND	0005	
0016	OUT	0705	
0017	LD	0002	
0018	OR	0704	
0019	ANDNOT	0003	
0020	ANDNOT	0705	
0021	OUT	0704	
0022	LD	0002	
0023	AND	0005	
0024	OR	0703	
0025	ANDNOT	0004	
0026	AND	0702	
0027	OUT	0703	
0028	LD	0004	
0029	AND	0001	
0030	OR	0707	
0031	ANDNOT	0706	
0032	ANDNOT	0000	
0033	OUT	0707	
0034	LD	1000	
0035	AND	0000	
0036	OR	0701	
0037	AND	0007	
0038	ANDNOT	0700	
0039	OUT	0701	
0040	LD	0004	
0041	OR	0702	
0042	ANDNOT	0005	
0043	ANDNOT	0703	
0044	OUT	0702	
0045	LD	0006	
0046	AND	0004	
0047	OR	0700	
0048	AND	0007	
0049	ANDNOT	0701	
0050	OUT	0700	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Instruction	Area	Timer/Counter Data	121
0051	LDNOT	0006		
0052	AND	0004		
0053	LD	0800		
0054	OR	0700		
0055	LATCH	1000		
0056	LDNOT	0007		
0057	AND	0000		
0058	TIM	0000	0010	
0059	END			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ นาย ทวีพล ซื่อสัตย์

วันเดือนปีเกิด 18 เมษายน 2515 อายุ 26 ปี

สถานที่ทำงาน ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ถนนฉลองกรุง
ลาดกระบัง
กรุงเทพฯ 10520

โทรศัพท์ที่สะดวกในการติดต่อ 326-7346-7

ประวัติการศึกษา

ปริญญาตรี	วศ. บ (การวัดคุมฯ)	สถาบันเทคโนโลยี พระจอมเกล้าเจ้าคุณทหารลาดกระบัง	
อนุปริญญา	ปวส. อีเลคทรอนิกส์	วิทยาลัยเทคนิคมีนบุรี	กรุงเทพฯ
	ปวช. อีเลคทรอนิกส์	วิทยาลัยเทคนิคมีนบุรี	กรุงเทพฯ
มัธยมศึกษาตอนต้น	ม.3	โรงเรียนบางกะปี	กรุงเทพฯ
ประถมศึกษา	ป.6	โรงเรียนคลองกะจะ	กรุงเทพฯ

ประวัติการทำงาน สอนวิชา Programmable logic controls system และวิชา Operations Research

ประสบการณ์หรือความชำนาญพิเศษ

เคยเป็นวิทยากรในโครงการร่วมมือระหว่างบริษัท Omron Electronics (Thailand) กับ สจล. เรื่อง Advance Factory automation

เคยร่วมประชุมวิชาการทางด้านวิศวกรรมไฟฟ้า 10 สถาบัน ครั้งที่ 19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้