สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การจำลองเครื่องพีแอลซี

PLC SIMULATOR

โดย

นายจีระวุธ ฤทธิ์ยากุล

นายเนติกุล จำปาทอง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน 45717
วัน, เดือน, ปี 3 ก.พ. 2546

.b.....
.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจำลองเครื่องพีแอลซี

PLC SIMULATOR

โดย

นายจีระวุธ ฤทธิ์ยากุล

นายเนติกุล จำปาทอง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชา วิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การจำลองเครื่องพีแอลซี (PLC Simulator)

ผู้จัดทำ

1. นาย จีระวุธ ฤทธิ์ยากุล รหัสนักศึกษา 41014078
2. นาย เนติกุล จำปาทอง รหัสนักศึกษา 41014231

อาจารย์ที่ปรึกษา

(อาจารย์ เทพจิตร์ เชยโภคา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจำลองเครื่อง พีแอลซี

นาย จีระวุธ ฤทธิ์ยากุล 41014078
นาย เนติกุล จำปาทอง 41014231
อาจารย์ เทพจิตร์ เชยโภคา อาจารย์ที่ปรึกษา
ปีการศึกษา 2544

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ เป็นการรวบรวมข้อมูลและผลงานที่ได้คิดค้นขึ้น ผลงานชิ้นนี้เป็นโปรแกรมใช้งานที่มีประโยชน์ต่อการนำไปใช้ในงานอุตสาหกรรม โดยมีรูปแบบเดียวกับเครื่องควบคุมพีแอลซี(PLC) คือสามารถใช้คำสั่งการใช้งาน โหลด(LOAD), แอน(AND),ออ(OR) เป็นต้น ผลที่ได้จะแสดงออกโดยความต่างศักย์ 5 โวลต์ ซึ่งชุดคำสั่งที่เป็นวงจรไฟฟ้าที่เรียกว่า แลดเดอร์ (Ladder) สามารถเก็บไว้และนำกลับมาใช้ใหม่ได้อีกครั้ง โดยที่ไม่จำเป็นจะต้องเขียนขึ้นมาใหม่ ส่วนภายในของโปรแกรมหรือการเขียนโปรแกรมนี้ ใช้โปรแกรมสร้างแอพพลิเคชั่น เดลไฟล(Delphi) ในการเขียนทั้งหมด ส่วนของโครงสร้างภายในของโปรแกรมส่วนหลักๆก็จะใช้ อิมเมจ(Image) ในการแสดงผล โครงงานชิ้นนี้สร้างขึ้นมาเพื่อเพิ่มความสะดวกแก่ผู้ใช้ มีประโยชน์โดยเพิ่มทางเลือกในการซื้อเครื่องควบคุมพีแอลซีได้อีกทางหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PLC Simulator

Mr.Jeerawut Rityakul No. 41014078

Mr.Netikul Jumpathong No.41014231

Mr.Theprachit Choeypoca Advisor

2001

Abstract

This thesis is PLC (Programmable Logic Control) Simulator which consists of the invention and the collecting of data. The project is the application that will be useful for the industrial sector. It works like the PLC that the user can use the functions like "LOAD","AND","OR", and etc. The output is 5 volts resistance. Moreover, the "Ladder" which is the circuit network, can be kept and reused, so it does not need to program it again. All program or the programming use a compiler called "Delphi". The main internal structure of the program will show the output through "Image". This project is invented to make the user more comfortable and to let them have another alternative apart from the PLC.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
สารบัญ	III
สารบัญรูปภาพ	VI
สารบัญตาราง	X
บทที่1 บทนำ	1
1.1 วัตถุประสงค์	2
1.2 ขอบเขตของงานที่ทำ	2
บทที่ 2 ทฤษฎี	3
2.1 โปรแกรมเมเบิลลอจิกคอนโทรลเลอร์	3
2.1.1 ประวัติความเป็นมา	3
2.1.2 โปรแกรมเมเบิลลอจิกคอนโทรลเลอร์	3
2.1.3 โครงสร้างของพีแอลซี	6
2.1.4 ส่วนประกอบของพีแอลซี	8
2.1.5 ความสามารถของพีแอลซี	20
2.1.6 ตัวอย่างการใช้งานพีแอลซีในอุตสาหกรรม	21
2.1.7 การทำงานของพีแอลซี	22
2.1.8 การทำงานของวงจรแลดเดอร์	24
2.1.9 หลักการทำงานของพีแอลซี	25
2.1.10 วงจรตรรก (ลอจิก)	25
2.1.11 กฎการเขียนโปรแกรมภาษาแลดเดอร์	31
2.1.12 หลักการเขียนโปรแกรมภาษาแลดเดอร์	31
2.1.13 การเขียนโปรแกรมภาษาแลดเดอร์	31
2.2 Delphi 5.0	35
2.2.1 ความสามารถของ Delphi 5.0	36
2.2.2 สร้างแอพพลิเคชันแบบ Event-Driven	37
2.2.3 รู้จักกับออบเจ็ค	38
2.2.4 ส่วนประกอบของการทำงานของ Delphi	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5 ขั้นตอนการสร้างแอพพลิเคชันเบื้องต้นกับ Delphi 5.0 42
2.2.6 การใช้งานคอมโพเนนต์ลาเบล 44
2.2.7 การใช้งานคอมโพเนนต์อีดิท 44
2.2.8 การใช้งานคอมโพเนนต์เมโม 44
2.2.9 การใช้งานคอมโพเนนต์อิมเมจ 45
2.3 การติดต่อระหว่างเครื่องคอมพิวเตอร์ และ อุปกรณ์อินพุตและเอาท์พุต 45
2.3.1 บล็อกไดอะแกรมของการติดต่อระหว่างเครื่องคอมพิวเตอร์ และ อุปกรณ์อินพุตและเอาท์พุต 45
2.3.2 ลักษณะเบื้องต้น 45
2.3.3 หน้าที่ของขาต่างๆ 47
2.3.4 การใช้งาน 8255 48
บทที่ 3 การออกแบบและรูปแบบการทำงาน 49
3.1 ส่วนของแลดเดอร์ 49
3.2 ส่วนของบูลลีน 51
3.3 ส่วนของคอมไพล์ 54
3.4 ส่วนของรัน 54
3.5 ส่วนของอุปกรณ์อินพุตและเอาท์พุต 54
บทที่ 4 การใช้งานและผลการทดลอง 55
4.1 การใช้งาน 55
4.2 ตัวอย่างการใช้จริงและผลที่ได้ 58
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ 60
5.1 ปัญหาที่พบ 60
5.2 การแก้ไข 60
5.3 สรุปผลการทดลอง 61
ภาคผนวก 62
ภาคผนวก ก กราฟการทำงานของโปรแกรม 62
ภาคผนวก ข คู่มือการใช้บอร์ด ET-PC8255 84
ภาคผนวก ค อุปกรณ์อินพุตและอุปกรณ์เอาท์พุต 95
ภาคผนวก ง รายละเอียด ชิพเบอร์ ULN2803A และ TLP521 100
กิตติกรรมประกาศ 110

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม 111

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 พีแอลซี ชนิดหนึ่งของ Omron 4
รูปที่ 2.2 ลักษณะการเดินสายไฟที่เรียกว่า ฮาร์ดวาย 5
รูปที่ 2.3 โครงสร้างของพีแอลซี 7
รูปที่ 2.4 ส่วนประกอบของซีพียู 8
รูปที่ 2.5 ตัวอย่างอุปกรณ์ที่ต่อเข้ากับซีพียูชนิดหนึ่ง 9
รูปที่ 2.6 ตัวอย่างการต่ออุปกรณ์จากพีแอลซี ชนิดหนึ่ง 10
รูปที่ 2.7 ตัวอย่างการต่ออุปกรณ์กับพีแอลซี 11
รูปที่ 2.8 I/O Scan และ โปรแกรม Scan 12
รูปที่ 2.9 การสแกนตามลำดับก่อนหลังชนิดหนึ่ง 12
รูปที่ 2.10 ตารางบิตสมมติของสัญญาณ I/O 13
รูปที่ 2.11 ตัวอย่างอุปกรณ์ที่ใช้เป็นสัญญาณอินพุต 14
รูปที่ 2.12 ตัวอย่างอุปกรณ์ที่เป็นส่วนของเอาท์พุต 15
รูปที่ 2.13 บล็อกไดอะแกรมของ AC อินเทอเฟส อินพุตโมดูล 15
รูปที่ 2.14 วงจรและการต่อสาย AC อินพุต โมดูล 16
รูปที่ 2.15 บล็อคไดอะแกรมของ AC อินเทอเฟสเอาท์พุตโมดูล 17
รูปที่ 2.16 วงจรของ AC เอาท์พุตโมดูล 17
รูปที่ 2.17 การต่อเทอโมคัปเปิ้ลเข้ากับอนาล็อคอินพุตโมดูล 18
รูปที่ 2.18 เครื่องป้อนโปรแกรมของ Omron 19
รูปที่ 2.19 เครื่องป้อนโปรแกรมของ Toshiba 19
รูปที่ 2.20 เครื่องป้อนโปรแกรมของ Koyo 20
รูปที่ 2.21 ตัวอย่างการกวนของเหลว 22
รูปที่ 2.22 วงจรการใช้รีเลย์ 23
รูปที่ 2.23 วงจรการใช้พีแอลซี (อินพุตโมดูล) 23
รูปที่ 2.24 วงจรการใช้พีแอลซี (เอาท์พุตโมดูล) 24
รูปที่ 2.25 วงจรแลดเดอร์ 24
รูปที่ 2.26 ไฟในห้องจะติดได้ก็ต่อเมื่อต่อสะพานและหลอดไฟอยู่ในกล่องเท่านั้น 26
รูปที่ 2.27 สัญญาณให้ตื่นนอน 26
รูปที่ 2.28 สัญลักษณ์ของ AND Gate ที่มีอินพุต 2 ตัว 27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.29 ถ้าอินพุตทั้งหมดเป็น “1” ได้เอาท์พุตเป็น “1” แต่ถ้าอินพุตตัวใดตัวหนึ่งเป็น “0” เอาพุตจะเป็น “0” 27
รูปที่ 2.30 หลอดไฟจะติดได้เมื่อสวิตช์ A และ B ติดเท่านั้น 28
รูปที่ 2.31 สัญญาณของ OR Gate ที่มีอินพุต 2 ตัว 28
รูปที่ 2.32 แสดงสภาวะของหลอดไฟ ณ สวิตช์ 29
รูปที่ 2.33 การทำงานของ OR Gate หลอดไฟจะติดเมื่อสวิตช์ A หรือ B ติด 29
รูปที่ 2.34 สัญลักษณ์ของ NOT Gate 29
รูปที่ 2.35 หลอดไฟจะติดถ้าสวิตช์ไม่ถูกกด 29
รูปที่ 2.36 ถ้ามีสัญญาณไฟฟ้าเป็น “1” และสวิตช์ความดันเปิดจะมีสัญญาณเตือนเกิดขึ้น 30
รูปที่ 2.37 ให้สัญญาณไฟฟ้าเป็น “1” และสวิตช์ความดันเป็น “1” จะไม่มีสัญญาณ 30
รูปที่ 2.38 สัญลักษณ์ของ NAND Gate ที่มีอินพุต 2 ตัว 30
รูปที่ 2.39 สัญลักษณ์ของ NOR Gate ที่มีอินพุต 2 ตัว 30
รูปที่ 2.40 คำสั่ง LD 31
รูปที่ 2.41 คำสั่ง OUT 31
รูปที่ 2.42 คำสั่ง AND 32
รูปที่ 2.43 คำสั่ง OR 32
รูปที่ 2.44 คำสั่ง LD NOT 32
รูปที่ 2.45 คำสั่ง AND NOT 33
รูปที่ 2.46 คำสั่ง OR NOT 33
รูปที่ 2.47 คำสั่ง AND LD 33
รูปที่ 2.48 คำสั่ง OR LD 34
รูปที่ 2.49 คำสั่ง TIMER 34
รูปที่ 2.50 คำสั่งหน่วงเวลา 34
รูปที่ 2.51 คำสั่ง COUNTER 35
รูปที่ 2.52 คำสั่งนับจำนวน 35
รูปที่ 2.53 แอพพลิเคชันตัวอย่างที่สร้างจาก Delphi 37
รูปที่ 2.54 แนวคิดของ Massage Loop ของ Windows 38
รูปที่ 2.55 องค์ประกอบของแอพพลิเคชัน 38
รูปที่ 2.56 ตัวอย่างการกำหนดค่าพรอพเพอร์ตี้ใน Object Inspector 39
รูปที่ 2.57 เรียกใช้เมธอดของออบเจ็กต์ในโปรแกรม 40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.58 คอมโพเนนต์ต่างๆ ใน Component Palette 40
รูปที่ 2.59 องค์ประกอบการทำงานของ Delphi 5.0 41
รูปที่ 2.60 Code Explorer และ Code Editor 42
รูปที่ 2.61 เลือกคอมโพเนนต์จาก Component Palatte 43
รูปที่ 2.62 บล็อกไดอะแกรมของการติดต่อระหว่างเครื่องคอมพิวเตอร์ 45
และอุปกรณ์อินพุตและเอาท์พุต
รูปที่ 2.63 ตำแหน่งขาต่างๆ ของ 8255 46
รูปที่ 2.64 แผนผังภายในของ 8255 46
รูปที่ 3.1 คำสั่ง Load 49
รูปที่ 3.2 คำสั่ง Load not 49
รูปที่ 3.3 คำสั่ง OUT 49
รูปที่ 3.4 คำสั่ง Timer 49
รูปที่ 3.5 คำสั่ง Counter 49
รูปที่ 3.6 รูปเส้นนอน 49
รูปที่ 3.7 รูปเส้นตั้ง 50
รูปที่ 3.8 รูปปุ่ม Delete 50
รูปที่ 3.9 คำสั่ง Load 51
รูปที่ 3.10 คำสั่ง Load not 51
รูปที่ 3.11 คำสั่ง And 51
รูปที่ 3.12 คำสั่ง Or 51
รูปที่ 3.13 คำสั่ง And Load 51
รูปที่ 3.14 คำสั่ง Or Load 51
รูปที่ 3.15 คำสั่ง Out 52
รูปที่ 3.16 คำสั่ง Timer 52
รูปที่ 3.17 คำสั่ง Counter 52
รูปที่ 3.18 หน้าจอบูลลีน 52
รูปที่ 3.19 รูปวงจรแลดเดอร์ 53
รูปที่ 4.1 หน้าจอหลัก 55
รูปที่ 4.2 หน้าจอแอดเดรส 56
รูปที่ 4.3 การคอมไพล์ 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.4 การเขียนวงจรที่ถูกต้องตามรูปแบบ 57
รูปที่ 4.5 การเขียนแลดเดอร์ที่ไม่ถูกต้องตามรูปแบบ 57
รูปที่ 4.6 แสดงหน้าจอบูลลีน 58
รูปที่ 4.7 การเซฟวงจรแลดเดอร์ 58
รูปที่ 4.8 วงจรตัวอย่าง 1 59
รูปที่ 4.9 วงจรตัวอย่าง 2 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 เปรียบเทียบระหว่างระบบซีเควนซ์กับระบบพีแอลซี 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในการควบคุมขบวนการผลิตของโรงงานอุตสาหกรรมเกือบทุกประเภท จะมีการควบคุมซีเคว้นซ์ (Sequence Control) อยู่เสมอ เช่น การควบคุมการสตาร์ทมอเตอร์ การควบคุมสายพานลำเลียง ระบบกำจัดน้ำเสีย การควบคุมแบบนี้มักจะใช้รีเลย์แม่เหล็กไฟฟ้า ไทม์เมอร์ เคาน์เตอร์ คอนแทกเตอร์ ลิมิตสวิตช์ปุ่มกด เป็นองค์ประกอบที่สำคัญในการนำมาประกอบเป็นวงจรในตู้ควบคุม และเป็นวิธีการแบบเดียวกันที่เราคุ้นเคยกันมาแต่ในอดีต

ปัจจุบัน วิทยาการทางด้านอิเล็กทรอนิกส์และไมโครคอมพิวเตอร์เจริญขึ้นมาก ทำให้เกิดอุปกรณ์ควบคุมชนิดใหม่ที่มีชื่อเรียกว่า Porogramable Logic Control หรือ PLC แนวความคิดของอุปกรณ์ควบคุมแบบใหม่นี้แตกต่างจากวงจรควบคุมโดยใช้รีเลย์แม่เหล็กไฟฟ้าอย่างสิ้นเชิง เป็นการปฏิวัติเทคนิคทางด้านการควบคุมอัตโนมัติจากฮาร์ดแวร์มาเป็นซอฟแวร์ ต่อไปการออกอแบบวงจรควบคุมซีเคว้นซ์จะเร็วขึ้นอีก 10 เท่า การประกอบวงจรจะง่ายมากเพียงแต่โปรแกรมเข้าไปใน PLC นี้เท่านั้นจะจดจำวงจรและทำงานตามวงจรได้โดยไม่ผิดพลาด

ในปัจจุบันจะพบว่าเครื่อง PLC ถูกนำมาใช้ในระบบควบคุมอัตโนมัติอย่างแพร่หลาย โดยเฉพาะในโรงงานอุตสาหกรรมต่างๆ เป็นที่แน่นอนว่า PLC ย่อมมีราคาที่สูงตามความสามารถของมันด้วย จึงได้มีการคิดจะนำเอาเครื่องคอมพิวเตอร์ หรือ PC มาใช้แทน PLC เนื่องจาก PC ในปัจจุบันมีความสามารถจะทำงานได้หลายอย่างอีกทั้งยังมีราคาที่ถูกกว่า PLC มาก ข้อดีอีกประการหนึ่งคือ การโปรแกรมผ่าน PC เราสามารถตรวจสอบข้อผิดพลาดในการเขียนโปรแกรมได้ด้วย

โครงงานนี้จะประกอบไปด้วย 2 ส่วนคือ ส่วนของซอฟแวร์ และ ส่วนของฮาร์ดแวร์ ซึ่งในส่วนของซอฟแวร์ จะเป็นการเขียนโปรแกรมเลียนแบบการทำงานของ PLC โดยใช้โปรแกรม Delphi 5 ในการเขียน ส่วนของฮาร์ดแวร์จะใช้การ์ด PC 8255 เป็น Output ซึ่งรวมเป็น PLC Simulator ซึ่งลักษณะการใช้งานคือใช้ภาษาแลดเดอร์ในการโปรแกรมแล้วส่งข้อมูลออกไปทาง Output ซึ่งสัญญาณที่ออกมาสามารถนำไปใช้งานควบคุมได้จริง

ส่วนของซอฟแวร์นั้นจะสามารถเลียนแบบการทำงานของ PLC ได้ไม่หมดทุกความสามารถของ PLC โดยจะมีฟังก์ชั่นพื้นฐาน LD AND OR LDNOT ANDNOT ORNOT และ OUT รวมไปถึง TIMER และ COUNTER อีกทั้งยังสามารถแสดงในส่วนของ Boolean ได้ด้วย ซึ่งจะเป็นพื้นฐานและจุดเริ่มต้นในการพัฒนาซอฟแวร์ต่อๆไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 วัตถุประสงค์

1.1.1 สามารถสร้าง พีแอลซี ได้เองทั้งซอฟแวร์ และ ฮาร์ดแวร์

1.1.2 สามารถสร้างฮาร์ดแวร์ของ พีแอลซี โดยใช้ตัวประมวลผลที่มีประสิทธิภาพ

1.1.3 สามารถสร้างซอฟแวร์ของ พีแอลซี ที่สามารถเขียนโปรแกรมภาษาแลดเดอร์ได้

1.1.4 การใช้งานได้จริงในระดับที่น่าพอใจ

1.1.5 สามารถสร้าง พีแอลซี ที่ใช้ต้นทุนต่ำ

1.2 ขอบเขตของงานที่ทำ

1.2.1 สามารถที่จะเขียนโปรแกรมได้ง่าย ใช้งานได้ง่าย และใช้งานได้จริง

1.2.2 การใช้งานเป็นการอินเทอเฟสโดยการเขียนโปรแกรมควบคุมบนเครื่องคอมพิวเตอร์ โดยมีการรับอินพุตเข้ามาประมวลผลที่เครื่องคอมพิวเตอร์และส่งข้อมูลออกไปที่อุปกรณ์เอาท์พุต

1.2.3 สามารถรับคำสั่งพื้นฐานของ พีแอลซี ได้ เช่น คำสั่ง LD, AND, OR, LD NOT, AND NOT, OR NOT, OUT, TIMER, COUNTER, AND LD, OR LD

1.2.4 สามารถแสดงในส่วนของแลดเดอร์ และ ในส่วนของบูลลีนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1 โปรแกรมเมเบิล ลอจิก คอนโทรลเลอร์(Programmable Logic Controller)

2.1.1 ประวัติความเป็นมา

เมื่อปี พ.ศ. 2511 ในฝ่าย ไฮโดรเมติก(Hydromatic) ของบริษัท เจนเนอรัล มอเตอร์ (General Motors) ประเทศสหรัฐอเมริกา ได้คิดค้นอุปกรณ์ควบคุมแบบใหม่เพื่อใช้ทดแทนวงจรไฟฟ้าแบบเดิมที่ใช้กันอยู่ในโรงงานอุตสาหกรรมของบริษัทและในปีพ.ศ. 2512 ได้ถูกผลิตขึ้นจำหน่ายในประเทศสหรัฐอเมริกาเป็นแห่งแรก ส่วนในประเทศญี่ปุ่น พีแอลซี(PLC) ได้ถูกพัฒนาขึ้นมาภายหลังจากที่บริษัท ออมรอน(OMRON) ประเทศญี่ปุ่นประสบความสำเร็จในการผลิตโซลิด สเตทรีเลย์(Solid State Relay) ในปี พ.ศ. 2503 หลังจากนั้น 5 ปี พีแอลซี ก็ถูกนำออกจำหน่ายสู่ท้องตลาดจนเป็นที่แพร่หลายในเวลาต่อมา

พีแอลซี ของแต่ละบริษัทจะมีชื่อเรียกแตกต่างกันในแต่ละประเทศคือ พีซี(PC) หรือ โปรแกรมเมเบิล คอนโทรลเลอร์(Programmable Controller) เรียกกันในประเทศอังกฤษ, พีแอลซี หรือ โปรแกมเมเบิล ลอจิก คอนโทรลเลอร์ เรียกกันในประเทศสหรัฐอเมริกา, พีบีเอส หรือ โปรแกรมเมเบิล ไบนารี ซีสเตม(Programmable Binary System) เรียกกันในกลุ่มประเทศสแกนดิเนเวีย

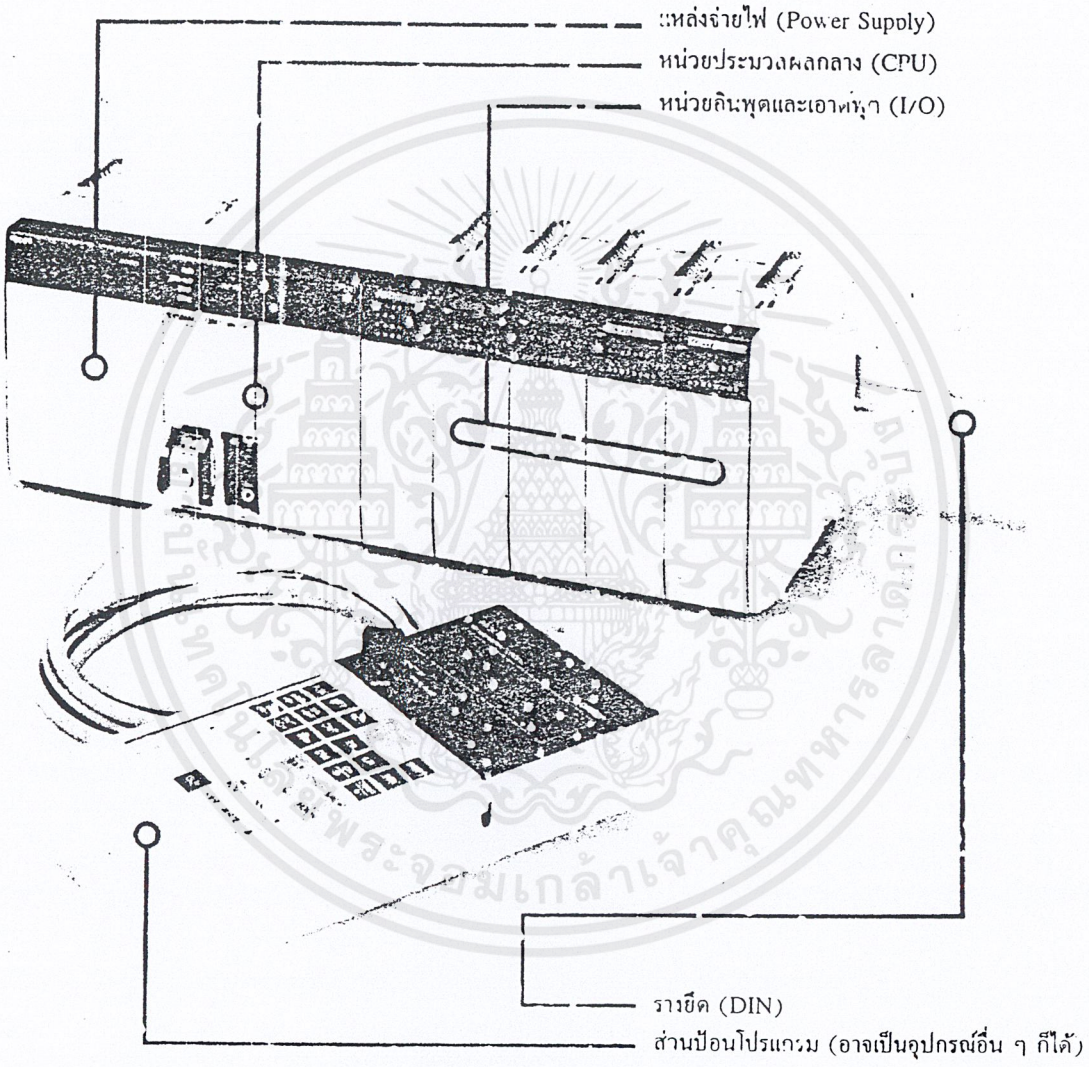
2.1.2 โปรแกรมเมเบิล ลอจิก คอนโทรลเลอร์

พีแ อลซี เป็นอุปกรณ์ชนิดโซลิด สเตท(Solid State) ที่ทำงานแบบลอจิก(Logic Function) การออกแบบการทำงานของ พีแอลซี จะคล้ายกับหลักการทำงานของคอมพิวเตอร์ จากหลักการพื้นฐานแล้ว พีแอลซี จะประกอบไปด้วยอุปกรณ์ที่เรียกว่า โซลิด สเตท ดิจิตอล ลอจิก อีลิเมนต์(Solid State Digital Elements) เพื่อให้ทำงานและตัดสินใจแบบลอจิก พีแอลซี ใช้สำหรับควบคุมกระบวนการทำงานของเครื่องจักรและอุปกรณ์ในโรงงานอุตสาหกรรม

การใช้ พีแอลซี สำหรับควบคุมเครื่องจักรหรืออุปกรณ์ต่างๆ ในโรงงานอุตสาหกรรมจะมีข้อได้เปรียบกว่าการใช้ระบบรีเลย์(Relay) ซึ่งจำเป็นจะต้องเดินสายไฟฟ้า หรือที่เรียกว่า ฮาร์ด วาย(Hard Wired) ฉะนั้นเมื่อมีความจำเป็นที่ต้องเปลี่ยนกระบวนการผลิต หรือลำดับการทำงานใหม่ ก็ต้องเดินสายไฟฟ้าใหม่ ซึ่งเสียเวลาและค่าใช้จ่ายสูง แต่เมื่อเปลี่ยนมาใช้ พีแอลซี แล้ว การเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานหม่นั้นทำได้โดยการเปลี่ยนโปรแกรมใหม่เท่านั้น

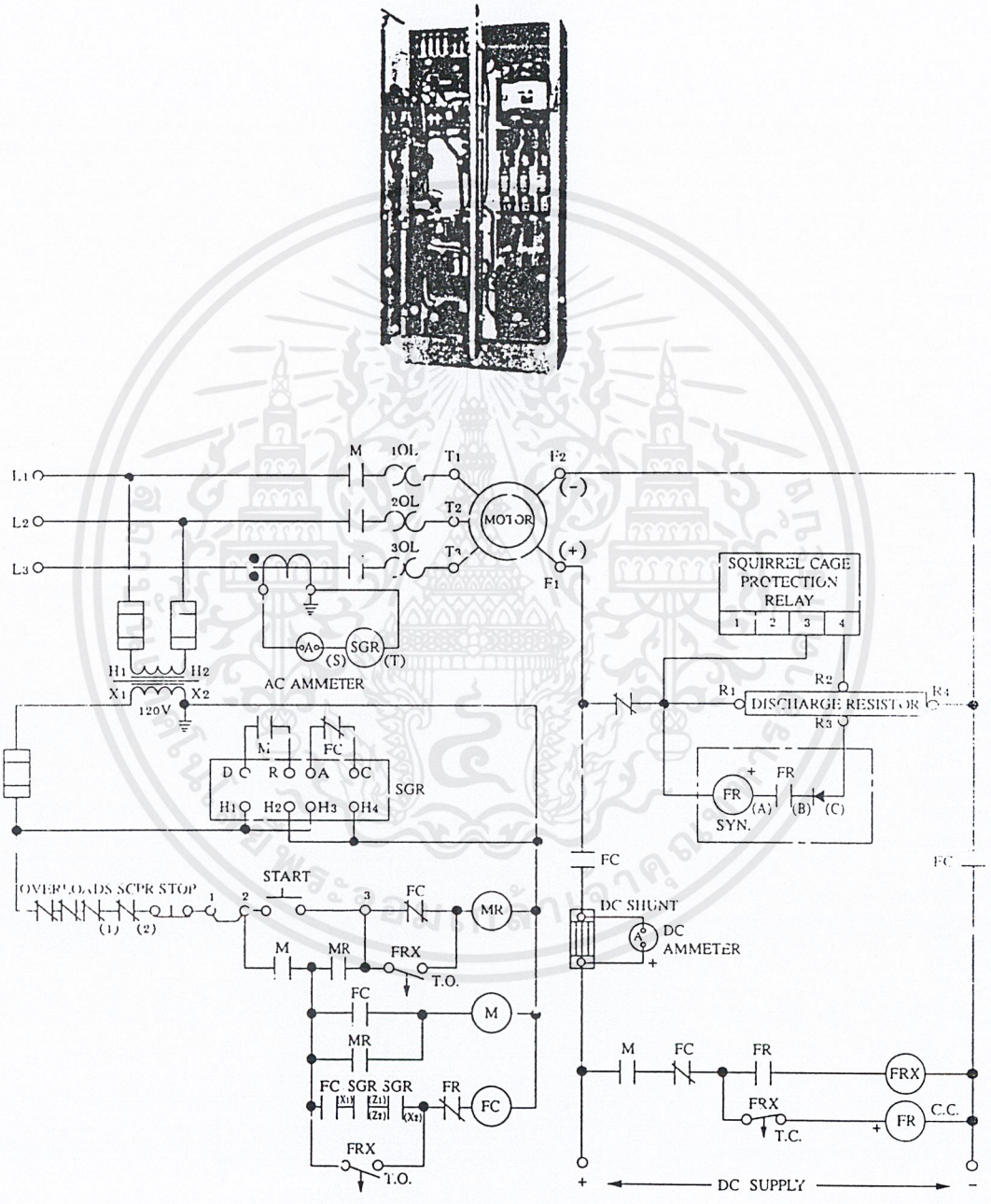
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้แล้ว พีแอลซี ยังใช้ระบบโซลิด สเตท ซึ่งน่าเชื่อถือกว่าระบบเดิม การกินกระแสไฟฟ้าน้อยกว่า และสะดวกกว่าเมื่อต้องการขยายขั้นตอนการทำงานของเครื่องจักร



รูปที่ 2.1 พีแอลซี ชนิดหนึ่งของ Omron

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 ลักษณะการเดินสายไฟที่เรียกว่า ฮาร์ดวาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	ระบบซีเคว้นซ์(Sequence) หรือใช้การเดินสายไฟ	ระบบโปรแกรมเมเบิลลอจิก คอนโทรลเลอร์
การควบคุมระบบ	ปรับเปลี่ยนแก้ไขเพิ่มเติมทำได้ยาก	สามารถปรับเปลี่ยนแก้ไขเพิ่มเติมได้ง่าย
การซ่อมหรือแก้ไข	ทำได้ยาก	ทำได้ง่าย
การติดต่อกับอุปกรณ์ภายนอก	ทำได้ยาก	ทำได้ง่าย
อายุการใช้งาน	น้อยกว่า เพราะมีส่วนของการเคลื่อนที่มาก	มากกว่า เพราะส่วนที่เคลื่อนที่มีน้อย
ติดต่อกับอุปกรณ์ไกลๆ	ทำได้ยุ่งยาก เพราะต้องเดินสายไฟยาวขึ้น	ทำได้ง่าย การเดินสายไฟน้อย
ความเร็วในการทำงาน	ช้า	เร็ว
ขนาด	ใหญ่	เล็ก
สัญญาณรบกวน	ดีมาก	ดี
การติดตั้ง	ใช้เวลามาก	ใช้เวลาน้อย
การทำงานที่ระบบซับซ้อน	ยาก ต้องใช้รีเลย์จำนวนมาก	ง่าย สะดวก

ตารางที่ 2.1 เปรียบเทียบระหว่างระบบซีเควนซ์กับระบบพีแอลซี

2.1.3 โครงสร้างของพีแอลซี

พีแอลซี เป็นอุปกรณ์คอมพิวเตอร์สำหรับใช้ในงานอุตสาหกรรม พีแอลซี ประกอบด้วย หน่วยประมวลผลกลาง หน่วยความจำ หน่วยรับข้อมูล หน่วยส่งข้อมูล และหน่วยป้อนโปรแกรม พีแอลซี ขนาดเล็กส่วนประกอบทั้งหมดของ พีแอลซี จะรวมกันเป็นเครื่องเดียว แต่ถ้าเป็นขนาดใหญ่ สามารถแยกออกเป็นส่วนแประกอบย่อยๆ ได้

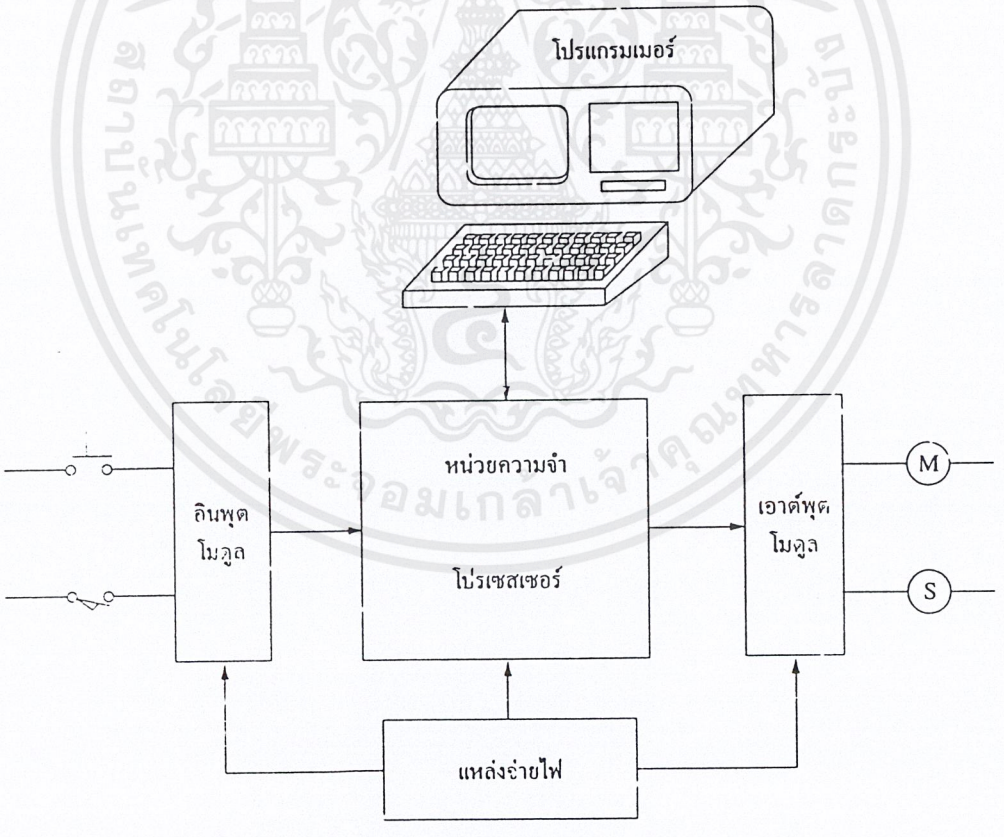
หน่วยความจำของ พีแอลซี ประกอบด้วย หน่วยความจำชนิด แรม(RAM) และ รอม(ROM) หน่วยความจำชนิด แรม ทำหน้าที่เก็บโปรแกรมของผู้ใช้และข้อมูลสำหรับใช้ในการปฏิบัติงานของ พีแอลซี ส่วน รอม ทำหน้าที่เก็บโปรแกรมสำหรับใช้ในการปฏิบัติงานของ พีแอลซี ตามโปรแกรมของผู้ใช้ รอม ย่อมาจาก รีด โอนรี เมมโมรี(Read Only Memory) สามารถโปรแกรมได้แต่ลบไม่ได้ ถ้าชำรุดแล้วซ่อมไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.1 แรม(Random Access Memory) หน่วยความจำประเภทนี้จะมีแบตเตอรี่เล็กๆ ต่อไว้เพื่อใช้เลี้ยงข้อมูลเมื่อเกิดไฟดับ การอ่านและเขียนโปรแกรมลงใน แรม ทำได้ง่ายมาก จึงเหมาะกับการใช้งานในระยะทดลองเครื่องที่มีการเปลี่ยนแปลงแก้ไขโปรแกรมบ่อยๆ

2.1.3.2 อีพรอม(Erasable Programmable Read Only Memory) หน่วยความจำชนิด อีพรอมนี้จะต้องใช้เครื่องมือพิเศษในการเขียนโปรแกรม การลบโปรแกรมทำได้โดยใช้แสงอัลตราไวโอเลต หรือตากแดดร้อนๆ นานๆ มีข้อดีตรงที่โปรแกรมจะไม่สูญหายแม้ไฟดับ จึงเหมาะกับการใช้งานที่ไม่ต้องการเปลี่ยนโปรแกรม

2.1.3.3 อีอีพรอม(Electrical Erasable Programmable Read Only Memory) หน่วยความจำชนิดนี้ไม่ต้องใช้เครื่องมือพิเศษในการเขียนและลบโปรแกรม โดยใช้วิธีการทางไฟฟ้าเหมือนกับ แรม นอกจากนั้นก็ไม่จำเป็นต้องมีแบตเตอรี่สำรองไฟเมื่อไฟดับ ราคาจะแพงกว่า แต่จะรวมคุณสมบัติที่ดีของทั้ง แรม และ อีพรอม เอาไว้ด้วยกัน



รูปที่ 2.3 โครงสร้างของพีแอลซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 ส่วนประกอบของ พีแอลซี

พีแอลซี แบ่งออกได้ 3 ส่วนด้วยกันคือ

2.1.4.1 ส่วนที่เป็นหน่วยประมวลผลกลาง(Control Processing Unit : CPU)

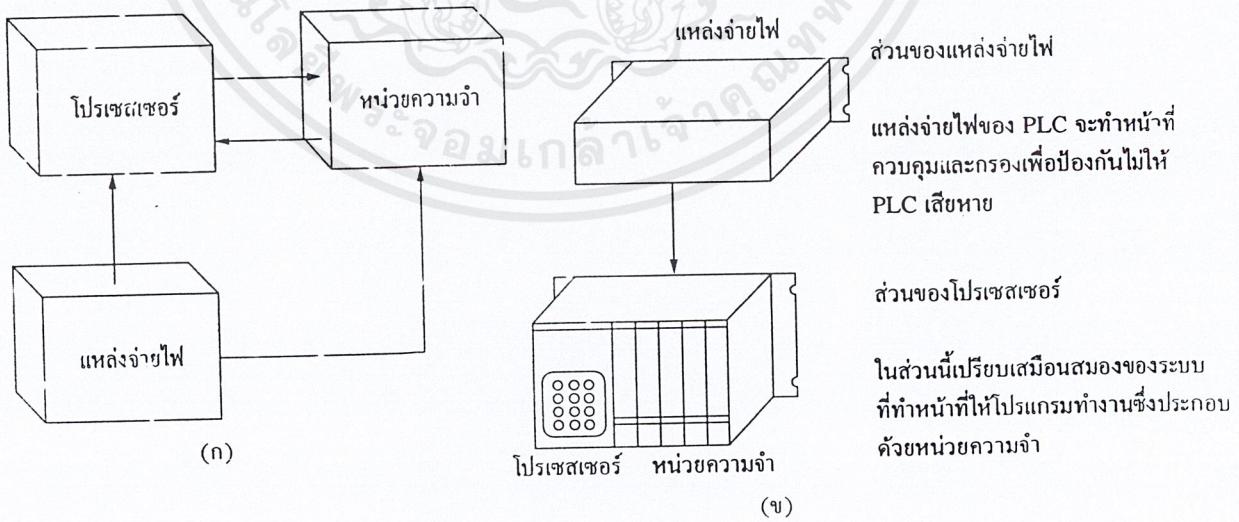
2.1.4.2 ส่วนที่เป็นอินพุต และ เอาท์พุต(Input Output : I/O)

2.1.4.3 ส่วนที่เป็นอุปกรณ์การโปรแกรม(Programming Device)

2.1.4.1 ซีพียู

ซีพียู เป็นส่วนมันสมองของระบบ ภายใน ซีพียู จะประกอบไปด้วยวงจรลอจิก เกต(Logic Gate) ชนิดต่างๆ หลายชนิด และมี ไมโครโปรเซสเซอร์ เบส (Microprocessor based) ใช้สำหรับแทนอุปกรณ์จำพวกรีเลย์(Relay) เคาน์เตอร์ (Counter) ไทเมอร์(Timer) และซีเควนเซอร์ (Sequencers) เพื่อให้ผู้ใช้ได้ออกแบบใช้วงจรรีเลย์ แลดเดอร์ ลอจิก(Rwlay Ladder Logic) เข้าไปได้

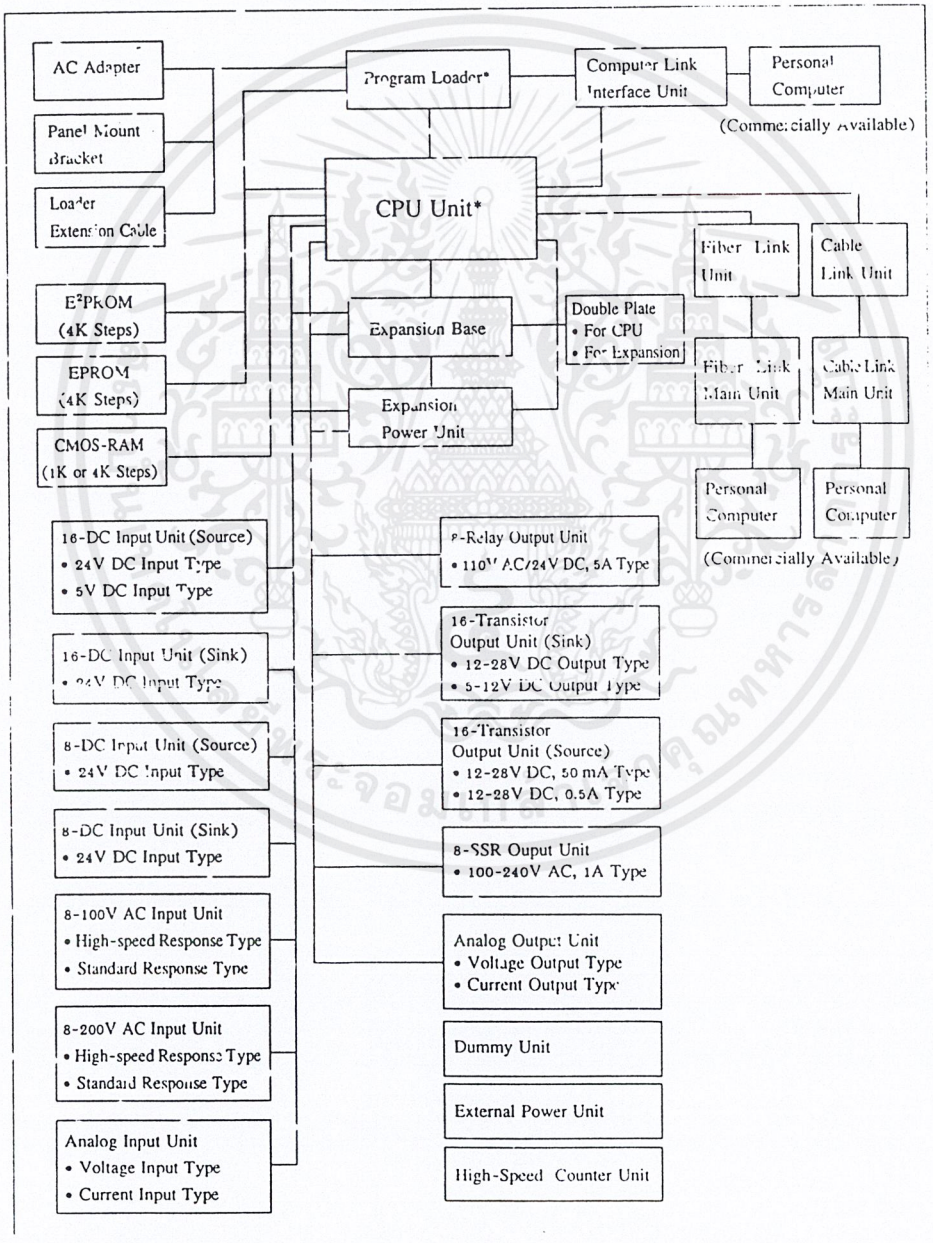
ซีพียู จะยอมรับ(Read) อินพุต เดต้า(Input Data) จากอุปกรณ์ให้สัญญาณ (Sensing Device) ต่างๆ จากนั้นจะปฏิบัติการและเก็บข้อมูลโดยใช้โปรแกรมจากหน่วยความจำ และส่งข้อมูลที่เหมาะสมถูกต้องไปยังอุปกรณ์ควบคุม(Control Device) แหล่งของกระแสไฟฟ้าตรง (DC Current) สำหรับใช้สร้างโวลต์ต่ำ(Low Level Voltage) ซึ่งใช้โดยโปรเซสเซอร์(Processor) และไอโอ โมดูล (I/O Modules) และแหล่งจ่ายไฟนี้จะเก็บไว้ที่ ซีพียู หรือแยกออกไปติดตั้งที่จุดอื่นก็ได้ขึ้นอยู่กับผู้ผลิตแต่ละราย



รูปที่ 2.4 ส่วนประกอบของซีพียู

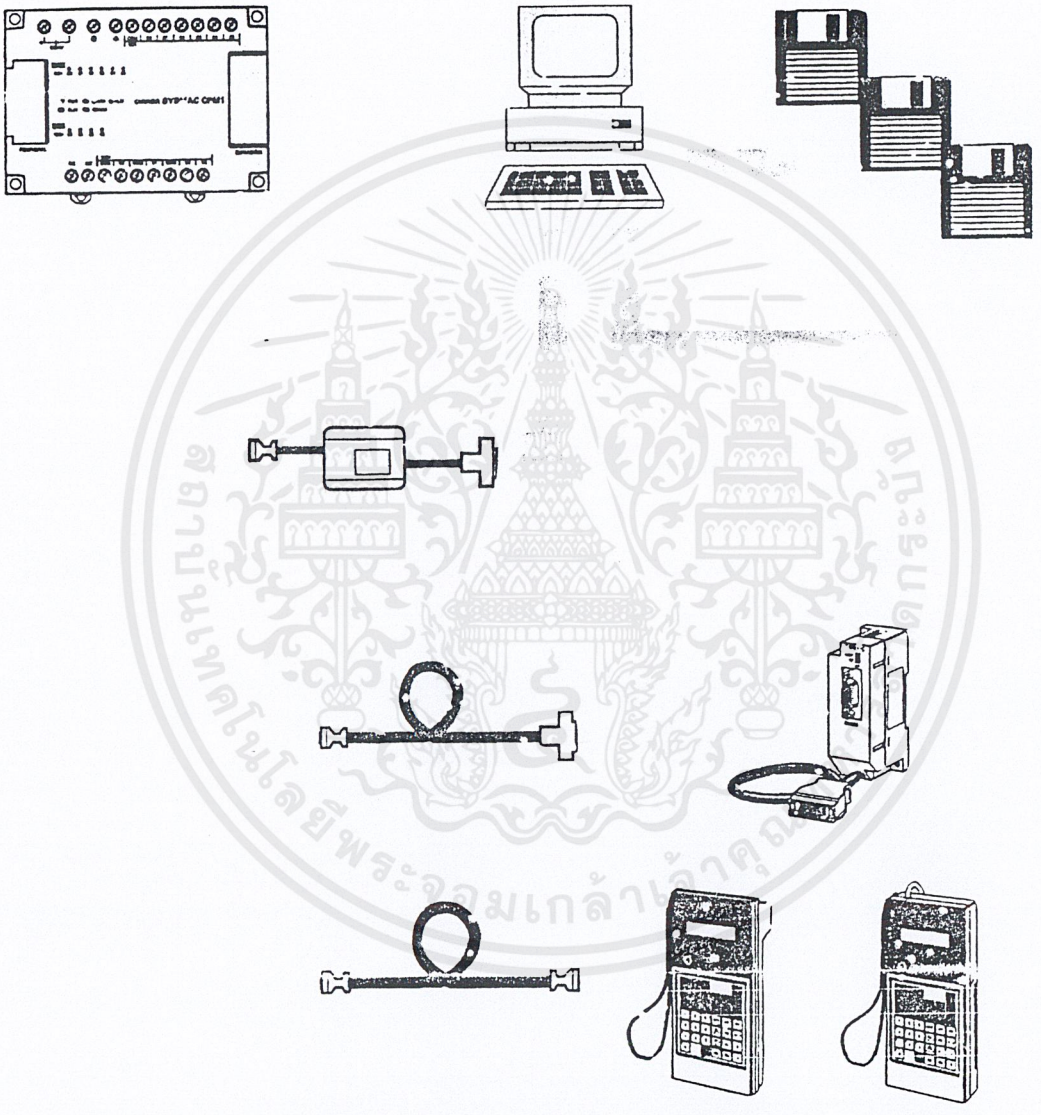
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.4 (ก) เป็น ซีพียู ที่รวมแหล่งจ่ายไฟเข้าด้วยกัน ส่วนรูป 2.4 (ข) จะแยกแหล่งจ่ายไฟออกมาต่างหาก ยังมีส่วนที่สำคัญที่อยู่ในชุดของ ซีพียู อีกชุดหนึ่ง คือ โปรเซสเซอร์ เมมโมรี โมดูล (Processor Memory Module) ซึ่งเป็นส่วนสำคัญชิ้นใหญ่ในตัวเรือน (Housing) ของ ซีพียู มันเป็นสมองที่ควบคุมโปรแกรม ภายในประกอบด้วยไมโครโปรเซสเซอร์ เมมโมรี ชิพ (Microprocessor Memory Chips) ที่ทำหน้าที่เก็บและเรียกข้อมูลจากหน่วยความจำ แล้วติดต่อกับวงจรที่ต้องการ



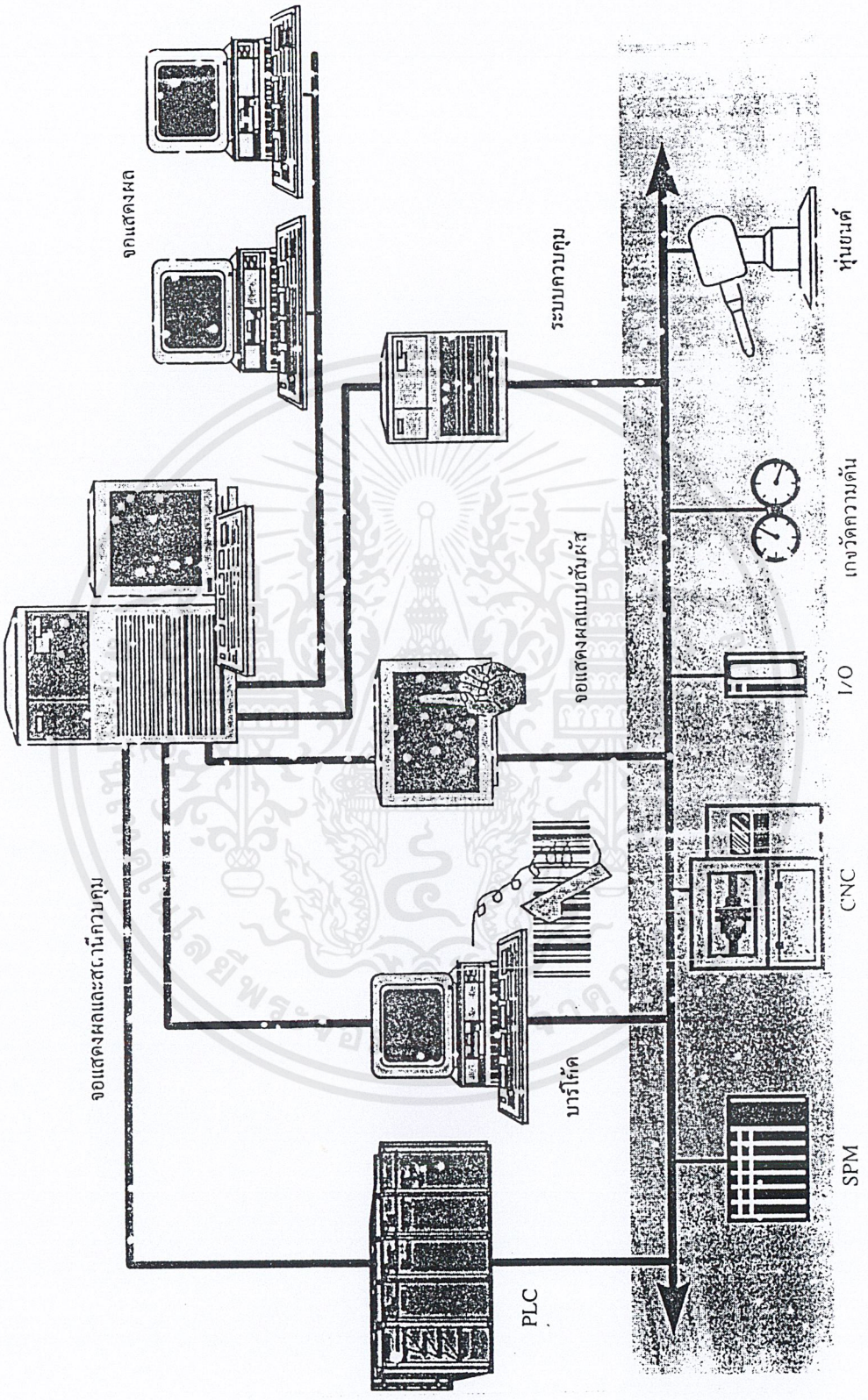
รูปที่ 2.5 ตัวอย่างอุปกรณ์ที่ต่อเข้ากับซีพียูชนิดหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ตัวอย่างการต่ออุปกรณ์จากพีแอลซี ชนิดหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



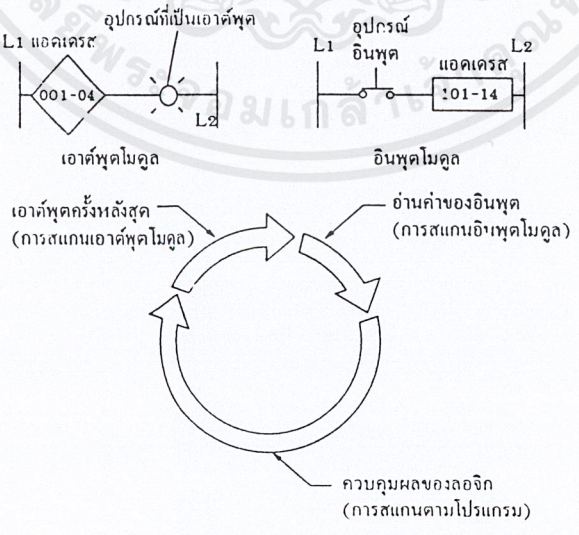
รูปที่ 2.7 ตัวอย่างการต่ออุปกรณ์กับพีแอลซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประมวลผลของ ซีพียู จากโปรแกรมทำได้โดยรับข้อมูลมาจากหน่วยอินพุตและเอาท์พุต และส่งข้อมูลมาจากหน่วยอินพุตและเอาท์พุต และส่งข้อมูลสุดท้ายที่ได้จากการประมวลผลไปยังหน่วยเอาท์พุต เรียกว่า การสแกน (Scan) ซึ่งใช้เวลาจำนวนหนึ่ง เรียกว่า เวลาสแกน (Scan Time) เวลาในการสแกนแต่ละรอบใช้เวลาประมาณ 1 ถึง 100 msec. (10 msec. = 100 ครั้งต่อวินาที) ทั้งนี้ขึ้นอยู่กับข้อมูลและความยาวของโปรแกรม หรือจำนวนอินพุต และ เอาท์พุต หรือจำนวนอุปกรณ์ที่ต่อจาก พีแอลซี เช่น เครื่องพิมพ์ จอภาพ เป็นต้น อุปกรณ์เหล่านี้จะทำให้เวลาในการสแกนยาวขึ้น การเริ่มต้นการสแกนเริ่มจากรับค่าของสภาวะของอุปกรณ์จากหน่วยอินพุตมาเก็บไว้ในหน่วยความจำ (Memory) เสร็จแล้วทำการปฏิบัติการตามโปรแกรมที่เขียนไว้ทีละคำสั่งจากหน่วยความจำนั้นจนสิ้นสุด แล้วส่งไปที่หน่วยเอาท์พุต

ตามรูปที่ 2.8 แสดงการสแกนของ พีแอลซี ที่ประกอบด้วย ไอ/โอ สแกน (I/O Scan) และ โปรแกรม สแกน (Program Scan) ซึ่งเป็นการสแกนเดี่ยว (Single PLC Scan)

รูปที่ 2.8 I/O Scan และ โปรแกรม Scan



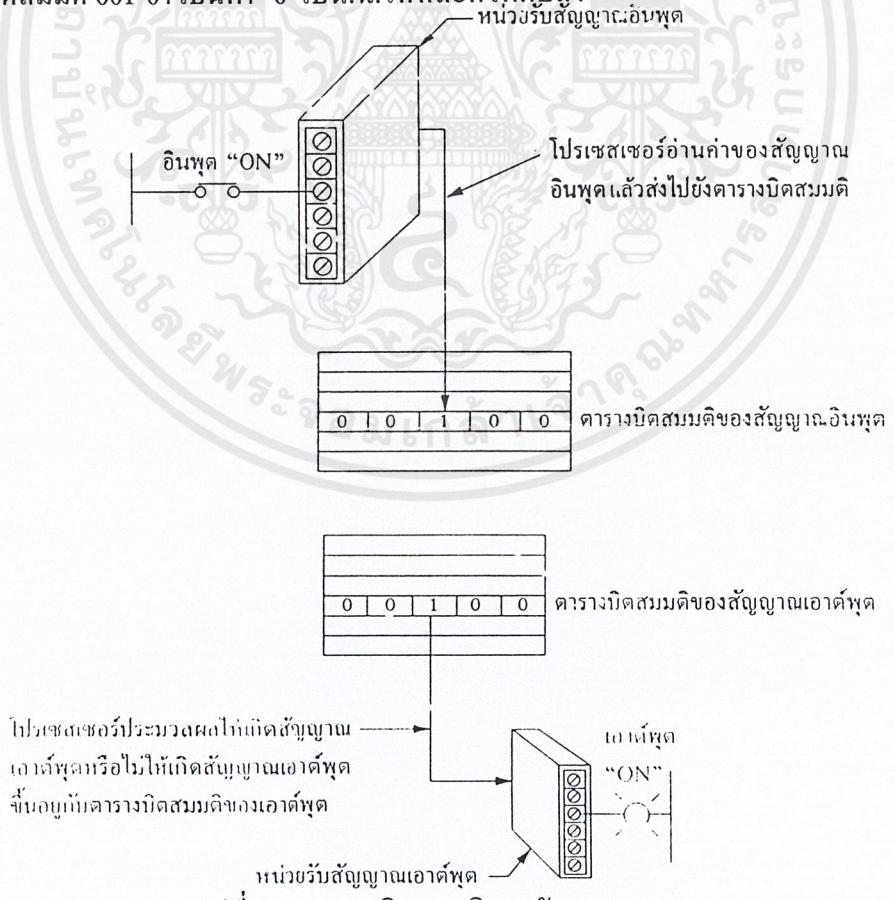
รูปที่ 2.9 การสแกนตามลำดับก่อนหลังชนิดหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูที่ 2.9 แสดงถึงการปฏิบัติการของโปรแกรม และมีการเปลี่ยนแปลงระหว่างเวลาที่กำลังสแกนซึ่ง พีแอลซี ไม่สามารถที่จะเปลี่ยนแปลงค่านั้นได้ แต่จะรับไว้ในหน่วยความจำ หลังจากนั้นเมื่อทำการสแกนตามโปรแกรมเรียบร้อยแล้ว จึงนำผลการเปลี่ยนแปลงครั้งหลังสุดส่งออกไปที่หน่วยของเอาท์พุต

เมื่ออุปกรณ์ที่เป็นอินพุตต่อกับแอดเดรส (Address) 101-14 อยู่ในตำแหน่งปิด (Closed) จะทำให้อินพุต โมดูล (Input Module) ส่งสัญญาณเป็นค่า "1" (ON) ไปยังตารางบิตสมมติ 101-14 เมื่อโปรแกรมถูกสแกนก็ทำให้โปรเซสเซอร์บิต 101-14 อยู่ในสภาวะ "1" (ON) ด้วย จึงสรุปว่าเมื่ออินพุต 101-14 เป็น "1" รัง (Rung) นี้จะมีสภาวะเป็นความจริง (True) จึงทำให้โปรเซสเซอร์ตั้งค่าตารางบิตสมมติของเอาท์พุต 001-04 เป็น "1" และโปรเซสเซอร์มีสภาวะเป็น "ON" ทำให้หลอดไฟสว่างขึ้น และ กระบวนการนี้จะกระทำต่อไปตราบใดที่โปรเซสเซอร์ยังมีสภาวะ "ON" อยู่หรืออยู่ใน "RUN" mode

ถ้าอุปกรณ์ที่เป็นอินพุตถูกทำให้เปิดออก (Open) จะทำให้ค่า "0" เข้าไปแทนที่ในตารางบิตสมมติทันที เป็นผลให้รัง อ่านค่าเป็น "False" ทำให้โปรเซสเซอร์ตั้งค่าเอาท์พุตในตารางบิตสมมติ 001-04 เป็นค่า "0" เป็นผลให้หลอดไฟดับลง



รูปที่ 2.10 ตารางบิตสมมติของสัญญาณ I/O

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.2 ส่วนของอินพุตและเอาท์พุต (I/O Unit)

ส่วนของอินพุตและเอาท์พุต จะต้องร่วมกับชุดควบคุมเพื่อรับสภาวะและสัญญาณต่างๆ เช่น หน่วยอินพุตรับสัญญาณหรือสภาวะแล้วส่งไปยัง ซีพียู เพื่อประมวบผล เมื่อซีพียู ประมวลผลแล้วจะส่งให้ส่วนของเอาท์พุต เพื่อให้อุปกรณ์ทำงานตามที่โปรแกรมเอาไว้

สัญญาณอินพุตจากภายนอกที่เป็นสวิตช์และตัวรวจจับชนิดต่างๆ จะถูกแปลงให้เป็นสัญญาณที่เหมาะสมถูกต้อง ไม่ว่าจะเป็น เอซี(AC) หรือ ดีซี(DC) เพื่อให้ส่งให้ ซีพียู ดังนั้น สัญญาณเหล่านี้จะต้องมีความถูกต้องไม่เช่นนั้นแล้ว ซีพียู จะเสียหายได้

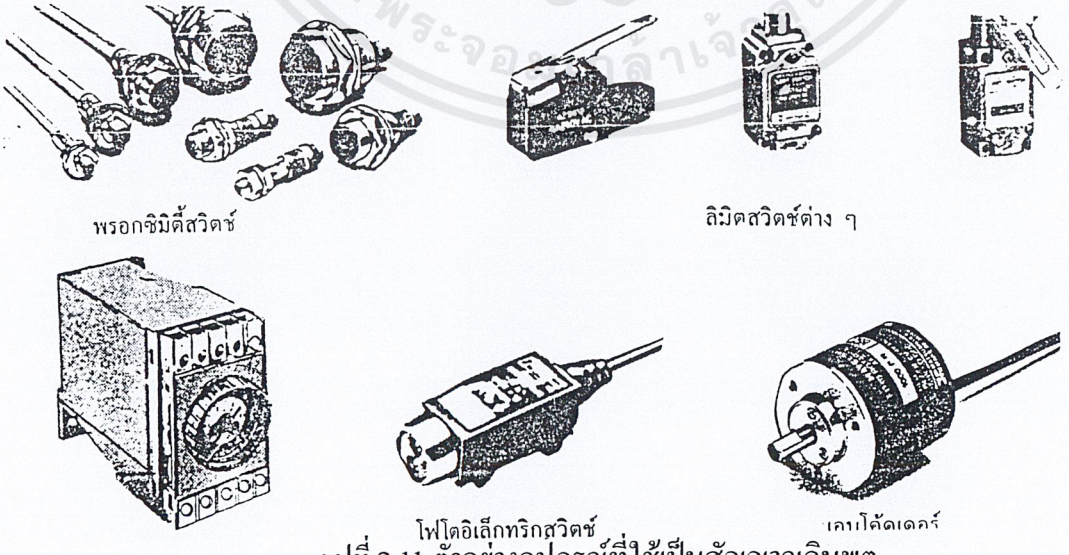
สัญญาณอินพุตที่ดีจะต้องมีคุณสมบัติและหน้าที่ดังนี้

2.1.4.2.1 ทำให้สัญญาณเข้า ได้ระดับที่เหมาะสมกับ พีแอลซี

2.1.4.2.2 การส่งสัญญาณระหว่าง อินพุตกับ ซีพียู จะติดต่อกันด้วยลำแสง ซึ่งอาศัยอุปกรณ์ประเภทโฟโตทรานซิสเตอร์ เพื่อต้องการแยกสัญญาณ (Isolate) ทางไฟฟ้าให้ออกจากกัน เป็นการป้องกันไม่ให้ ซีพียู เสียหายเมื่ออินพุตเกิดลัดวงจร

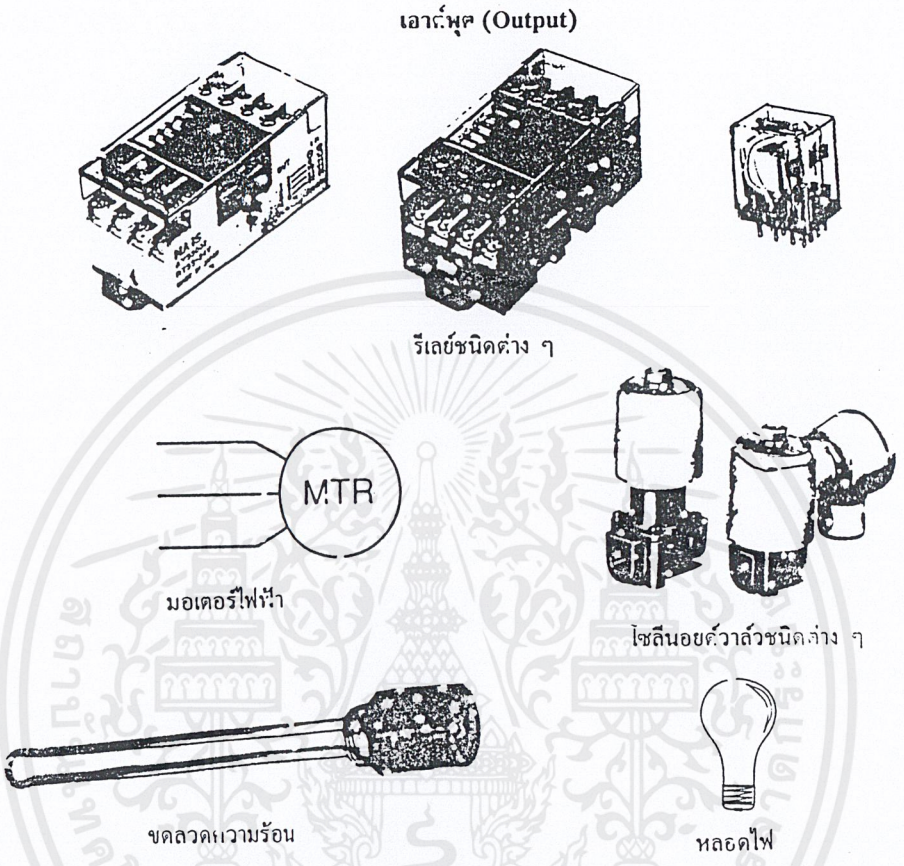
2.1.4.2.3 หน้าสัมผัสจะต้องไม่สั่นสะเทือน (Contact Chattering)

ในส่วนของเอาท์พุต จะทำหน้าที่รับค่าสภาวะที่ได้จากการประมวลผลของซีพียู แล้วนำค่าเหล่านี้ไปควบคุมอุปกรณ์ทำงาน เช่น รีเลย์ โซลีนอยด์ หรือหลอดไฟ เป็นต้น นอกจากนั้นแล้ว ยังทำหน้าที่แยกสัญญาณของหน่วยประมวลผลกลาง (ซีพียู) ออกจากอุปกรณ์เอาท์พุต โดยปกติเอาท์พุตนี้จะมีความสามารถขับโหลดด้วยกระแสไฟฟ้าประมาณ 1-2 แอมแปร์ แต่ถ้าโหลดต้องการกระแสไฟฟ้ามากกว่านี้ จะต้องต่อเข้ากับอุปกรณ์ขับอื่นเพื่อขยายให้รับกระแสไฟฟ้ามากขึ้น เช่น รีเลย์ หรือคอนแทคเตอร์ เป็นต้น



รูปที่ 2.11 ตัวอย่างอุปกรณ์ที่ใช้เป็นสัญญาณอินพุต

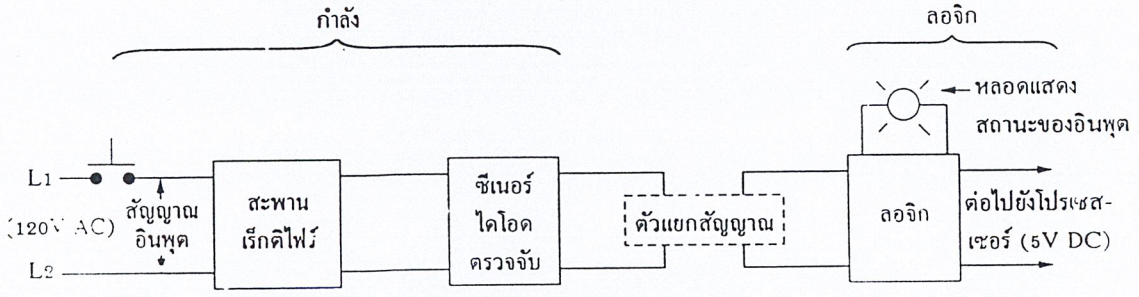
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 ตัวอย่างอุปกรณ์ที่เป็นส่วนของเอาท์พุต

ต่อไปนี้จะเป็นตัวอย่างโมดูลของอินพุตและเอาท์พุตชนิดต่างๆ

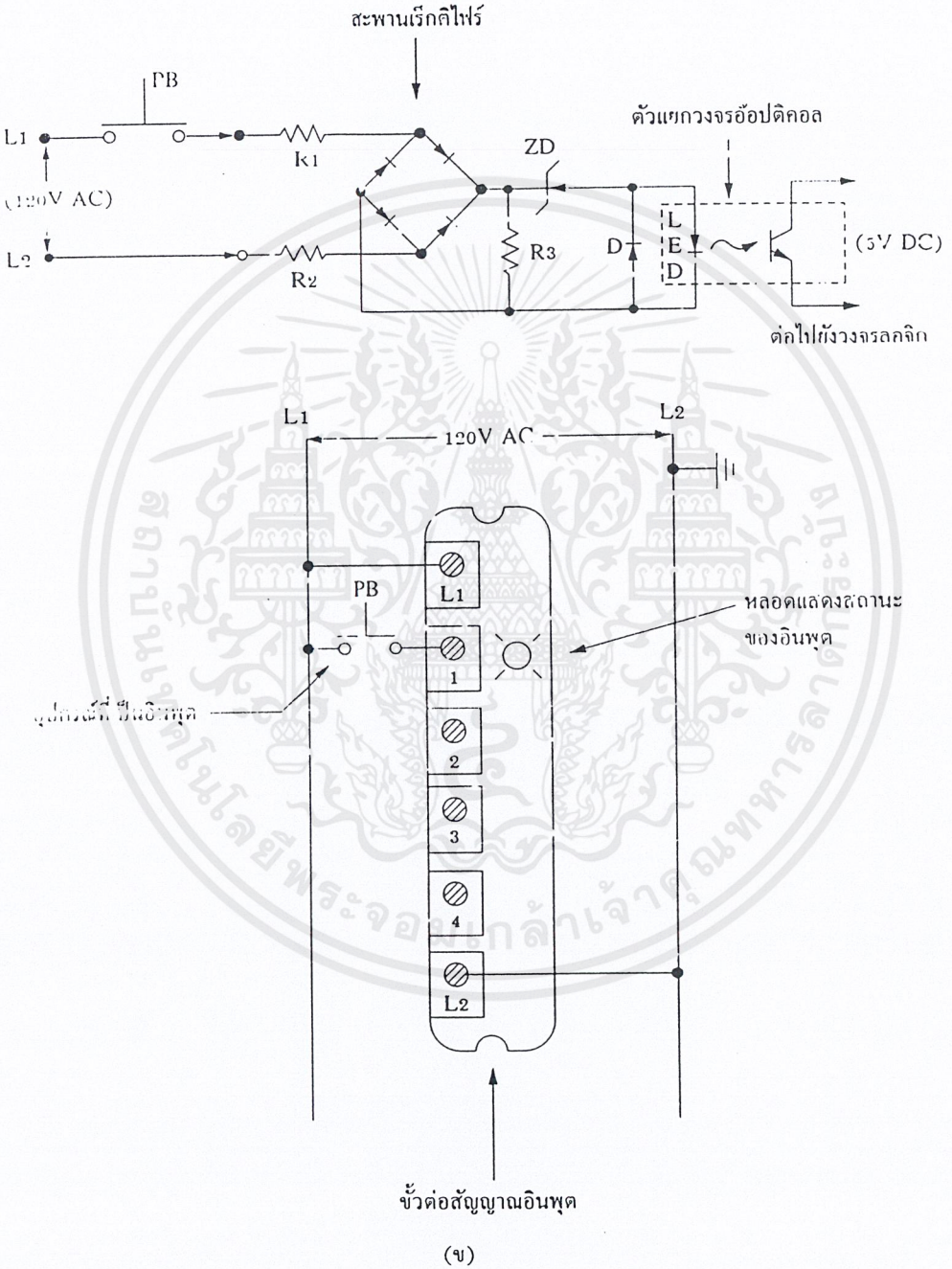
2.1.4.2.4 บล็อกไดอะแกรมของ AC Interface Input Module



รูปที่ 2.13 บล็อกไดอะแกรมของ AC อินเทอเฟส อินพุตโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.2.5 วงจรของ AC Input Module ดังรูปที่ 2.14 (ก) และการต่อสาย อินพุต โมดูล (Input Module) ชนิดหนึ่งดังรูปที่ 2.14 (ข)



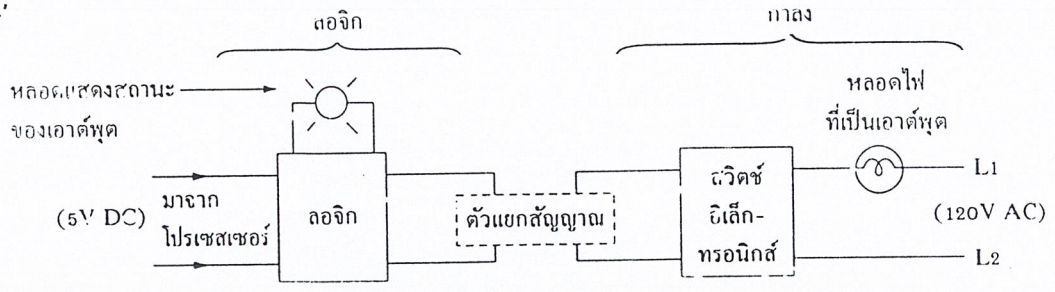
(ข)

รูปที่ 2.14 วงจรและการต่อสาย AC อินพุต โมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

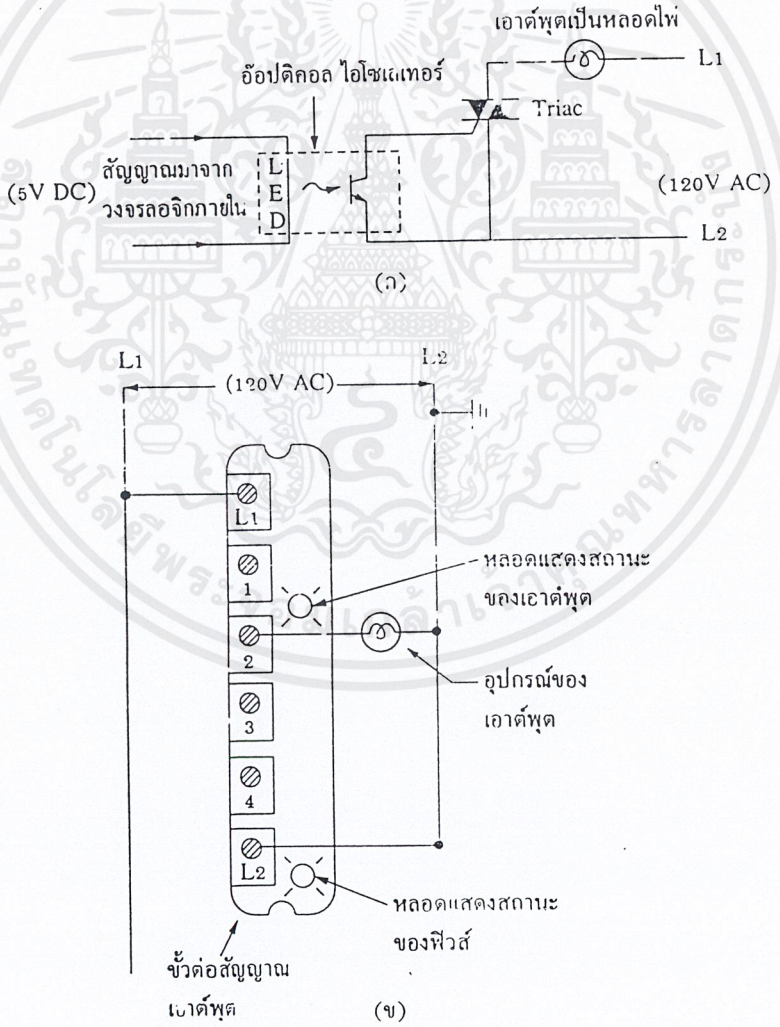
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.1.4.2.6 บล็อกไดอะแกรมของ AC Interface เอาท์พุต โมดูล (Output Module)



รูปที่ 2.15 บล็อคไดอะแกรมของ AC อินเทอเฟสเอาท์พุตโมดูล

2.1.4.2.7 วงจร AC Output Module ดังรูปที่ 2.16 (ก) และการต่อสายของเอาท์พุต โมดูล ดังรูปที่ 2.16 (ข)

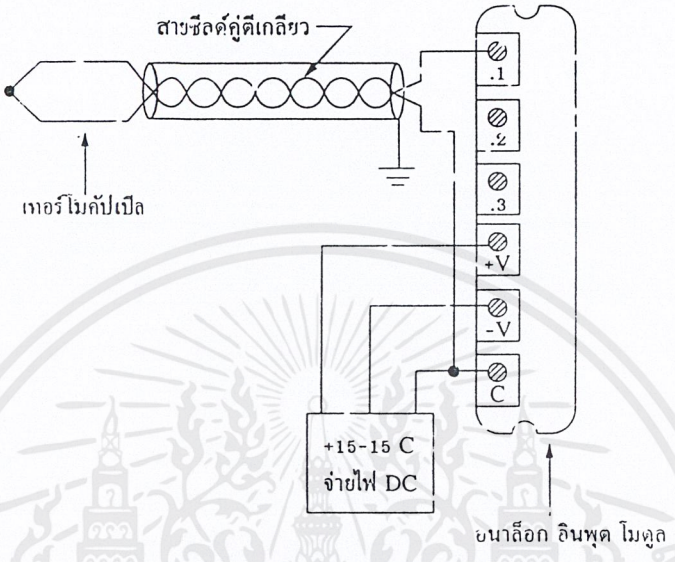


รูปที่ 2.16 วงจรของ AC เอาท์พุตโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

45717

2.1.4.2.8 การต่อเทอร์โมคัปเปิลเข้ากับอนาล็อก อินพุต โมดูล (Analog Input Module)



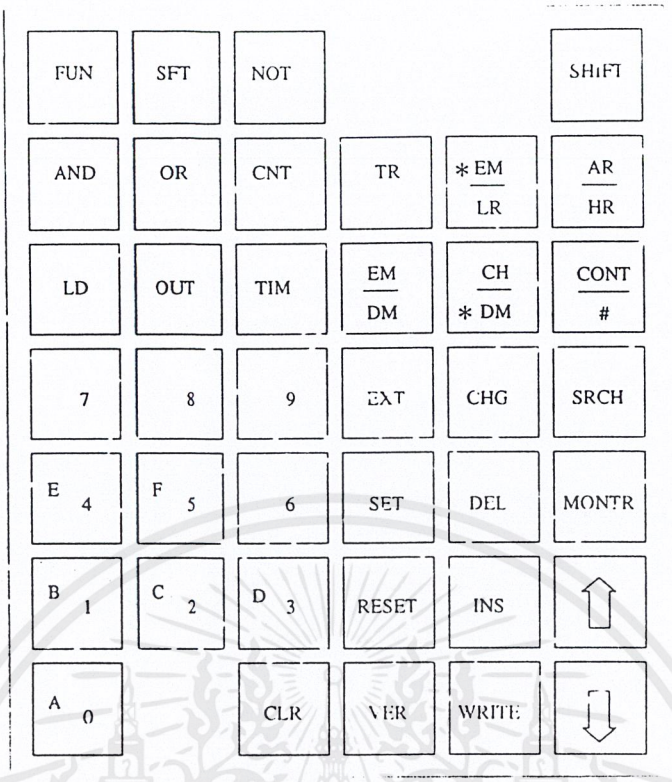
รูปที่ 2.17 การต่อเทอโมคัปเปิ้ลเข้ากับอนาล็อคอินพุตโมดูล

2.1.4.3 เครื่องป้อนโปรแกรม (Programming Device)

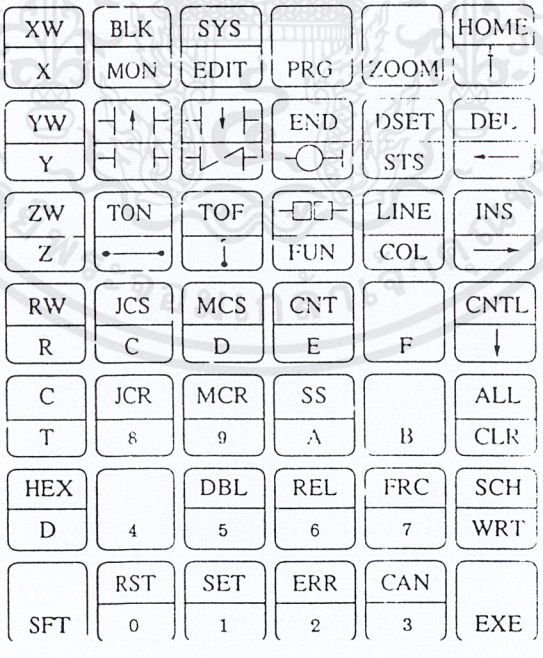
หน้าที่ของเครื่องป้อนโปรแกรมคือ ควบคุมโปรแกรมของผู้ใช้หน่วยความจำของ พีแอลซี นอกจากนั้นแล้วยังทำหน้าที่ติดต่อระหว่างผู้ใช้กับ พีแอลซี เพื่อให้ผู้ใช้สามารถตรวจการปฏิบัติงานของ พีแอลซี และผลการควบคุมเครื่องจักรและกระบวนการตามโปรแกรมควบคุมที่ผู้ใช้เขียนขึ้นได้อีกด้วย

เครื่องป้อนโปรแกรม (Hand Held) แต่ละยี่ห้อจะไม่เหมือนกันแต่มีจุดประสงค์อันเดียวกัน ต่อไปนี้จะเป็นตัวอย่างของเครื่องป้อนโปรแกรมยี่ห้อต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 เครื่องป้อนโปรแกรมของ Omron



รูปที่ 2.19 เครื่องป้อนโปรแกรมของ Toshiba

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AUX	MODE	CPU	STAT	SAVE
Z SG	S STR	SP STRN	GX OUT	GY CNT
U ISG	V AND	W ANDN	X SET	Y MLS
P CV	Q OR	R ORN	S RST	T MLR
K JMP	L ANDST	M ORST	N TMR	O INST#
H 7	I 8	J 9	←	FD REF FIND
E 4	F 5	G 6	→	OFF DEL
B 1	C 2	D 3	PREV	ON INS
A 0	CLR	SHFT	NEXT	ENT

รูปที่ 2.20 เครื่องป้อนโปรแกรมของ Koyo

2.1.5 ความสามารถของ พีแอลซี

พีแอลซี สามารถควบคุมงานได้ 3 ลักษณะ คือ

2.1.5.1 งานที่ทำตามลำดับก่อนหลัง (Sequence Control) ตัวอย่างเช่น

2.1.5.2 การทำงานของระบบรีเลย์

2.1.5.3 การทำงานของไทเมอร์ เคาน์เตอร์

2.1.5.4 การทำงานของ พี ซี บี การ์ด (P.C.B Card)

2.1.5.5 การทำงานในระบบกึ่งอัตโนมัติ ระบบอัตโนมัติ หรืองานที่เป็นกระบวนการทำงานของเครื่องจักรกลต่างๆ

2.1.5.2 งานควบคุมสมัยใหม่ (Sophisticated Control) ตัวอย่างเช่น

2.1.5.2.1 การทำงานทางคณิตศาสตร์ เช่น บวก ลบ คูณ หาร

2.1.5.2.2 การควบคุมแบบอนาล็อก (Analog Control) เช่น การควบคุมอุณหภูมิ (Temperature) การควบคุมความดัน (Pressure) เป็นต้น

2.1.5.2.3 การควบคุม พี ไอ ดี (P.I.D. (Proportional, Intergral, Derivation))

2.1.5.2.4 การควบคุม สเต็ป มอเตอร์ (Step Motor)

2.1.5.2.5 การควบคุมเซอร์โวมอเตอร์ (Servo motor Control)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5.2.6 อินฟอเมชั่น แฮนด์ลิ้ง (Information Handling)

2.1.5.3 การควบคุมเกี่ยวกับงานอำนวยการ (Supervisory Control) ตัวอย่างเช่น

2.1.5.3.1 งานสัญญาณเตือน (Alarm) และ โปรเซส มอนิเตอร์ริง (Process Monitoring)

2.1.5.3.2 Fault Diagnostic And Monitoring

2.1.5.3.3 งานต่อร่วมกับคอมพิวเตอร์ (RS-232C/RS422)

2.1.5.3.4 Printer / ASCII Interfacing

2.1.5.3.5 งานควบคุมอัตโนมัติในโรงงานอุตสาหกรรม (Factory Automation Networking)

2.1.5.3.6 แลน (LAN (Local Area Network))

2.1.5.3.7 แวน (WAN (Wide Area Network))

2.1.5.3.8 FA., FMS., CIM. เป็นต้น

2.1.6 ตัวอย่างการใช้ พีแอลซี ในอุตสาหกรรมต่างๆ

2.1.6.1 การผสมวัตถุดิบ

2.1.6.2 การขนถ่ายผลิตภัณฑ์

2.1.6.3 การกำจัดน้ำเสีย

2.1.6.4 การขุดเจาะน้ำมัน

2.1.6.5 การแปรรูปไม้

2.1.6.6 การทำเยื่อกระดาษ

2.1.6.7 การประกอบชิ้นส่วนรถยนต์

2.1.6.8 การพ่นสี

2.1.6.9 การย่อยเยื่อไม้

2.1.6.10 การตรวจสอบคุณภาพ

2.1.6.11 การประหยัดพลังงาน

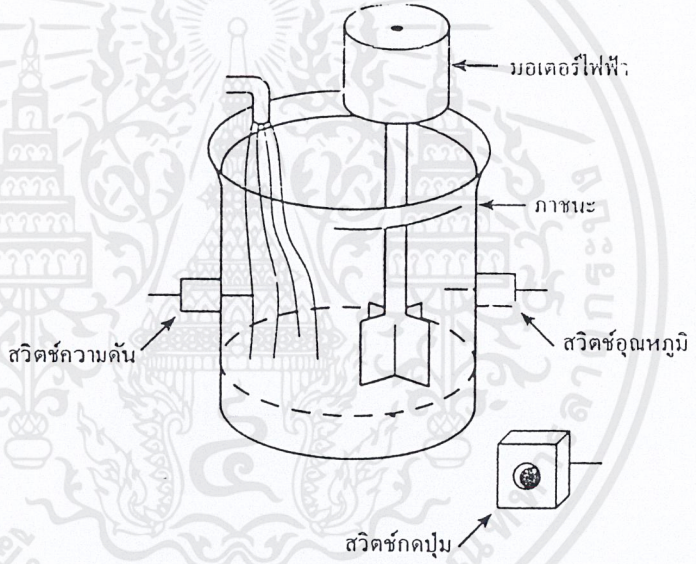
2.1.6.12 การแยกแร่

2.1.6.13 การกำจัดน้ำเสีย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 การทำงานของพีแอลซี

เครื่องจักรที่ควบคุมด้วย พีแอลซี จะมีความสามารถเขียนโปรแกรมการทำงานของเครื่องจักร และมีความยืดหยุ่นในการเขียนโปรแกรม เช่น การเปลี่ยนแปลงแก้ไขเพิ่มเติมก็สามารถกระทำได้ ซึ่งรวมถึงมีไทเมอร์ เคาน์เตอร์ หรือคำสั่งชนิดพิเศษต่างๆ เช่น มูฟ ดาต้า (MOV Data) และอื่นๆ อีกมากมายเพื่อใช้ควบคุมอุปกรณ์ภายนอก ไม่ว่าจะเป็นมอเตอร์ โซลินอยด์ (Solinid) หลอดไฟ เป็นต้น นอกจากนี้ ยังมีการติดต่อสื่อสารระหว่าง พีแอลซี กับคอมพิวเตอร์ เพื่อแลกเปลี่ยนข้อมูลระหว่างกันหรืออาจจะติดต่อกับจอชนิดสัมผัส (Touch Screen) เพื่ออำนวยความสะดวกต่อสัญญาณอินพุตและสัญญาณเอาท์พุต ยิ่งไปกว่านั้นการติดต่อกับคอมพิวเตอร์เพื่อให้คอมพิวเตอร์เป็นตัวควบคุม พีแอลซี อีกทีหนึ่ง ซึ่งจะทำให้ขีดความสามารถในการควบคุมสูงขึ้นอีก และต่อไปนี้จะได้กล่าวถึงการทำงานของ พีแอลซี โดยเริ่มจากตัวอย่างการควบคุมมอเตอร์กวนของเหลวในภาชนะโดยมีเงื่อนไขต่างๆ



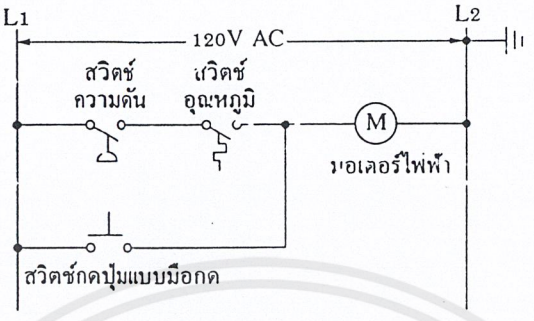
รูปที่ 2.21 ตัวอย่างการกวนของเหลว

ตามรูปที่ 2.21 ใช้มอเตอร์ไฟฟ้าสำหรับกวนของเหลวที่บรรจุอยู่ในภาชนะ เมื่อมีเงื่อนไขว่า มีอุณหภูมิ และความดันถึงค่าที่กำหนดเท่านั้น หรือสามารถให้มอเตอร์ไฟฟ้าทำงานด้วยการกดสวิตช์ก็ได้ โดยแยกสวิตช์กดออกมาต่างหาก

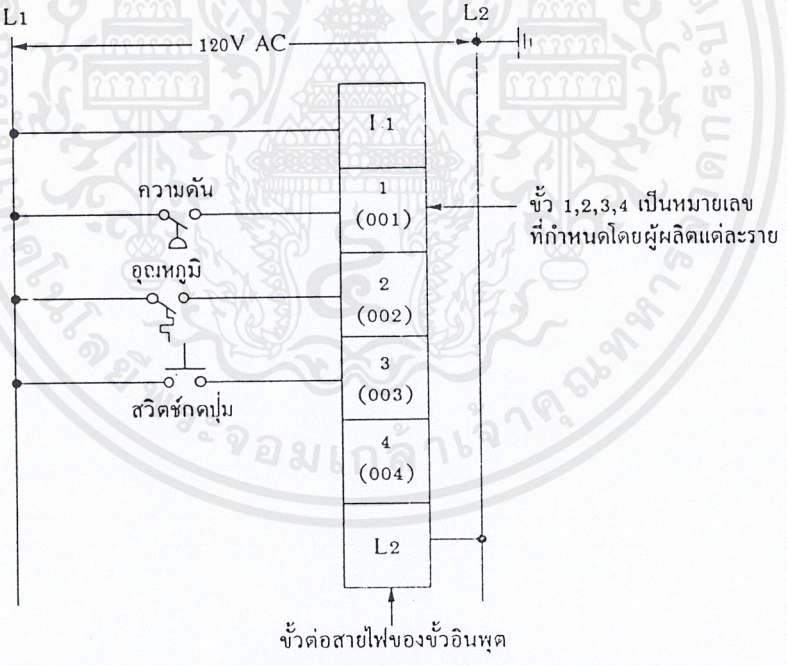
มอเตอร์ไฟฟ้าจะหมุนได้ก็ต่อเมื่อหน้าคอนแทคของเซนเซอร์อุณหภูมิ และเซนเซอร์ความดันต่อกันการต่อวงจรทำได้จากการใช้วงจรรีเลย์ ตามรูปที่.. และการใช้ พีแอลซี ตามรูปที่... ในกรณีของการใช้วงจรรีเลย์นั้น มอเตอร์ (M) ทำงานได้เมื่อสวิตช์ของอุณหภูมิและความดันต่อกันหรือสวิตช์มือกดถูกกดแต่ถ้าใช้ พีแอลซี ทำงาน ตามรูปที่... เป็นการต่ออุปกรณ์ด้านอินพุต ซึ่งมีคอนแทคของความดัน คอนแทคของอุณหภูมิ และคอนแทคของสวิตช์กดปุ่มแบบมือกด ซึ่งต่อเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปยังขั้วสัญญาณเข้า ส่วนรูปที่ 2.22 เป็นการต่ออุปกรณ์ทำงาน (มอเตอร์ไฟฟ้า) เข้ากับขั้วต่อสัญญาณออก

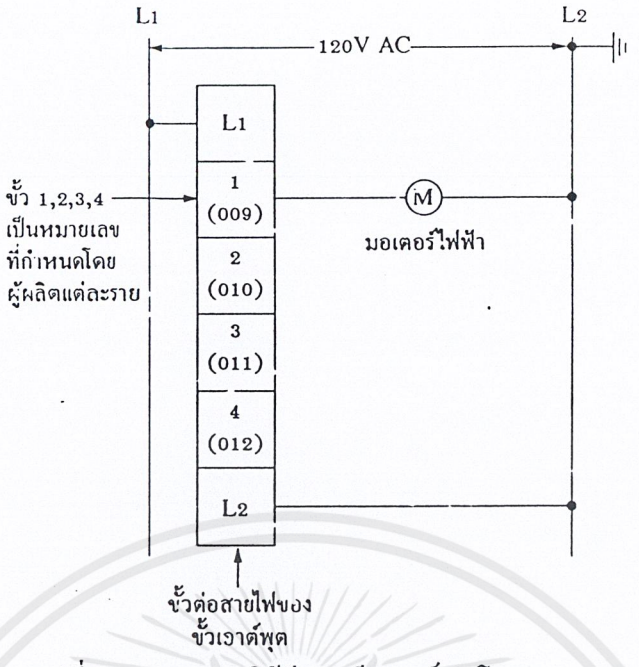


รูปที่ 2.22 วงจรการใช้รีเลย์



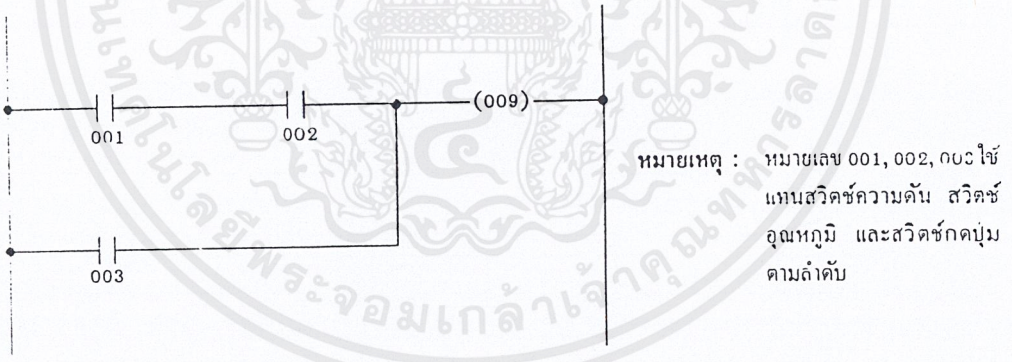
รูปที่ 2.23 วงจรการใช้พีแอลซี (อินพุตโมดูล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 วงจรการใช้พีแอลซี (เอาท์พุตโมดูล)

การเขียน พีแอลซี แลดเดอร์ (PLC Ladder Logic Diagram) เพื่อส่งโปรแกรมเข้าไปยังหน่วยความจำของ ซีพียู ดูได้จากรูปที่...



รูปที่ 2.25 วงจรแลดเดอร์

2.1.8 การทำงานของวงจรแลดเดอร์

เมื่อ พีแอลซี อยู่ในสภาวะพร้อมทำงาน (RUN Mode) แล้ว เมื่อมีโปรแกรมถูกป้อนเข้าไปยังหน่วยความจำของ ซีพียู ทำให้ ซีพียู ประมวลผลและได้ผลลัพธ์เป็นสัญญาณเอาท์พุต หน้าคอนแทคตามรูปที่... ซึ่งเป็นชนิดปกติเปิด (Normally Open) เพราะฉะนั้น ถ้าคอนแทค 001 และ 002

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อกัน ก็ทำให้เกิดเอาท์พุต 009 หรือหน้าคอนแทค 003 ต่อกัน ก็ทำให้เกิดเอาท์พุต 009 ได้เช่นกัน ลักษณะนี้เรียกว่า รัง คือมีสัญญาณอินพุตหนึ่งหรือมากกว่าที่ทำให้เกิดเอาท์พุตหนึ่งหรือมากกว่า

2.1.9 หลักการทำงานของ พีแอลซี

อธิบายตามเหตุการณ์ต่าง มีดังต่อไปนี้คือเมื่อ สัญญาณอินพุตเข้ามาจะถูกเก็บเป็นความจำไว้ในส่วนของความจำ (หน้าคอนแทคที่ต่อกัน เรียกว่า ลอจิก 1 ส่วนคอนแทคที่ไม่ต่อกัน เรียกว่า ลอจิก 0) หลังจากนั้นแลดเดอร์ไดอะแกรมก็จะสรุปผลรวมกับคอนแทคภายในว่าให้เป็นคอนแทคเปิด หรือ ปิด ขึ้นอยู่กับการบันทึกของหน่วยความจำ ถ้าหากต้องการสัญญาณเอาท์พุต ค่าของลอจิกต้องเป็นเลข 1 ซึ่งหมายถึงชุดหน้าคอนแทคของโมดูล อินเทอเฟส (Module Interface) ต่อกัน แต่ถ้าไม่มีกระแสไฟฟ้าไหลในวงจรทำให้ขดลวดความจำของคอยล์ (Coil Memory) มีค่าเป็นลอจิกเลข 0 และโมดูลอินเทอเฟส ไม่ต่อกัน

การทำงานของ พีแอลซี เมื่อครบ 1 รอบของลำดับดังกล่าวนี้เรียกว่า สแกน ส่วน สแกนไทม์ คือเวลาที่ต้องการสำหรับ 1 รอบการทำงาน ซึ่งเป็นตัววัดค่าความเร็วการทำงานของพีแอลซี 1 สแกนใช้เวลาประมาณ 1-100 ms. ขึ้นอยู่กับความยาวของโปรแกรมและชนิดของอินพุต เอาท์พุต

2.1.10 วงจรตรรก (ลอจิก)

เมื่อได้ทราบหลักการของเลขฐานชนิดต่างๆ แล้ว หลักการทำงานของ พีแอลซี ก็ยังใช้วงจรตรรก (ลอจิก) เพื่อให้เกิดสัญญาณเอาท์พุตที่มีเงื่อนไข (สัญญาณอินพุต) ชนิดต่างๆ หลักการของวงจรตรรกมีดังต่อไปนี้

วงจรตรรก หมายถึง วงจรไฟฟ้าที่ประกอบด้วยอุปกรณ์อิเล็กทรอนิกส์ หรือระบบรีเลย์ที่มีสัญญาณเพียง 2 ระดับ หรือ 2 สภาวะเท่านั้น พีแอลซี ใช้สัญญาณไฟฟ้า 2 ระดับแทน 2 เหตุการณ์ที่ต่างกัน เช่น การปิดเปิดวาลว์ การปิดเปิดสวิตช์ เป็นต้น วงจรตรรกมี 2 ชนิด คือแบบบวก (Positive Logic) แบบลบ (Negative Logic) ลอจิกแบบวงจรใช้สัญญาณไฟฟ้าระดับสูง แทนสภาวะลอจิก “1” และใช้สัญญาณไฟฟ้าระดับต่ำ แทนสภาวะลอจิก “0” ส่วนวงจรลอจิกแบบลบจะใช้สัญญาณไฟฟ้าระดับต่ำแทนสภาวะลอจิก “1” และใช้สัญญาณไฟฟ้าระดับสูง แทนสภาวะลอจิก “0”

สภาวะทางลอจิก คือ สภาวะ “1” หรือ “0” ใช้แทนการทำงานของอุปกรณ์ที่เปลี่ยนแปลง 2 สภาวะระบบควบคุมที่ใช้ระบบรีเลย์ และพีแอลซี จะนำสภาวะของอุปกรณ์เหล่านี้มาปฏิบั

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

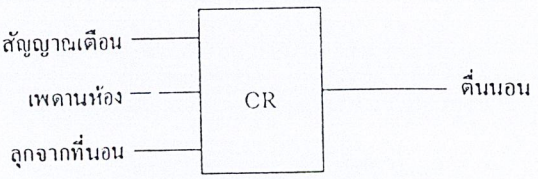
ติลอจิกด้วยกัน เพื่อให้เข้ากันกับเงื่อนไขการควบคุม ปฏิบัติการลอจิกประกอบด้วย AND OR และ NOT เพื่อทำให้สภาวะอินพุตต่างๆ เช่น A, B ทำให้เกิดเอาท์พุต Y เป็นต้น

พีชคณิตบูลลีนมีไว้สำหรับอธิบายความสัมพันธ์ทางลอจิก ทำให้เข้าใจได้ง่ายยิ่งขึ้น ตัวอย่างสมการบูลลีนของรูปที่... เขียนได้ว่า Y (แสงไฟฟ้า) = A (สวิตช์) . B (หลอดไฟ) วงจรลอจิกที่ใช้วิธีการเดินสายไฟเชื่อมต่ออุปกรณ์ต่างๆ เช่น รีเลย์ สวิตช์ ซึ่งมีความยุ่งยากและแก้ไขเพิ่มเติมได้ยาก ส่วน พีแอลซี ใช้โปรแกรมลอจิกกำหนดเงื่อนไขการควบคุม แทนการเดินสายไฟเชื่อมต่ออุปกรณ์ต่างๆ ดังกล่าวมาแล้ว จึงทำให้ง่ายขึ้น

พีแอลซี แทนวงจรรีเลย์ด้วยปฏิบัติการทางลอจิก AND OR และ NOT ซึ่งกำหนดตามเงื่อนไขที่ต้องการควบคุม โดยใช้คำสั่งหรือภาษา พีซี (PC Language) ภาษาพื้นฐานที่ พีแอลซีใช้ในการควบคุมแบบ “ON” หรือ “OFF” คือ ภาษาแลดเดอร์ และภาษาบูลลีน ภาษาแลดเดอร์ใช้สัญลักษณ์ของหน้าสัมผัสในการเขียนโปรแกรม การเปลี่ยนวงจรรีเลย์ให้เป็นโปรแกรม พีแอลซี ทำได้โดยใช้หน้าสัมผัสภาษาแลดเดอร์แทนสัญญาณรีเลย์

การทำงานของอุปกรณ์ดิจิตอล (Digital Equipment) จะอยู่บนหลักการพื้นฐานของลอจิกพื้นฐาน 3 ตัว คือ AND OR และ NOT แต่ละตัวจะมีหลักการของตัวเองและมีสัญลักษณ์ของตัวเอง ต่อไปนี้จะให้ Y เป็นเอาท์พุต และสัญญาณอินพุต ให้เป็นตัวอักษร ABC ส่วนเลข 1 หมายถึง มีสัญญาณ เลข 0 หมายถึง ไม่มีสัญญาณ

รูปที่ 2.26 ไฟในห้องจะติดได้ก็ต่อเมื่อต่อสะพานและหลอดไฟอยู่ในกล่องเท่านั้น



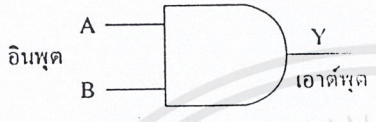
รูปที่ 2.27 สัญญาณให้ตื่นนอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีสัญญาณเตือน 3 อย่างเพื่อให้ตื่นนอน คือ มีสัญญาณดังขึ้น (Alarm) หรือเพดานห้องยุบลงมาหรือลุกออกจากที่นอน

2.1.10.1 หลักการของ AND Gate

AND Gate ทำให้เกิดสัญญาณเอาท์พุตได้ก็ต่อเมื่อ มีอินพุตทั้ง A และ B มีค่า "1"

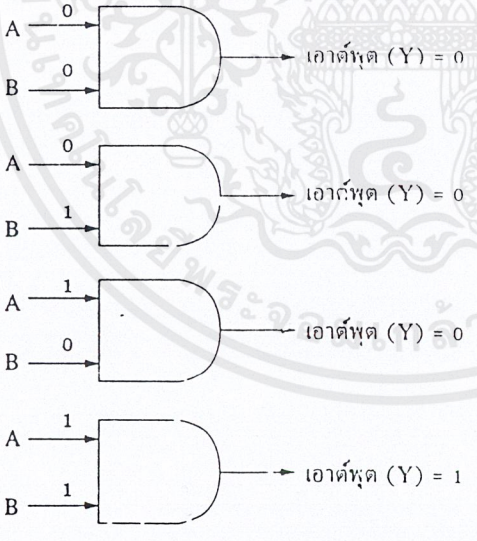


สัญลักษณ์ของ AND Gate

อินพุต		เอาต์พุต
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

ตารางตามความเป็นจริงของ AND

รูปที่ 2.28 สัญลักษณ์ของ AND Gate ที่มีอินพุต 2 ตัว

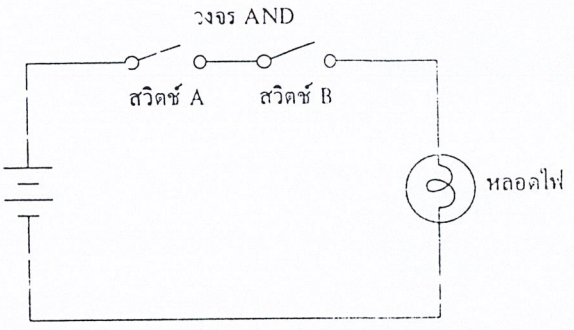


A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

ตารางตามความเป็นจริง

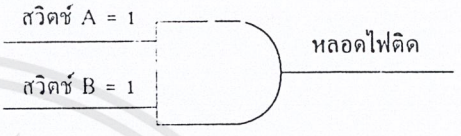
รูปที่ 2.29 ถ้าอินพุตทั้งหมดเป็น "1" ได้เอาท์พุตเป็น "1" แต่ถ้าอินพุตตัวใดตัวหนึ่งเป็น "0" เอาพุตจะเป็น "0"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



สวิตช์ A	สวิตช์ B	หลอดไฟ
ปิด (0)	เปิด (0)	ดับ (0)
เปิด (0)	ปิด (1)	ดับ (0)
ปิด (1)	เปิด (0)	ดับ (0)
ปิด (1)	ปิด (1)	ติด (1)

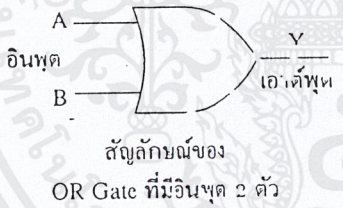
ตารางตามความเป็นจริง



รูปที่ 2.30 หลอดไฟจะติดได้เมื่อสวิตช์ A และ B ติดเท่านั้น

2.1.10.2 หลักการของ OR Gate

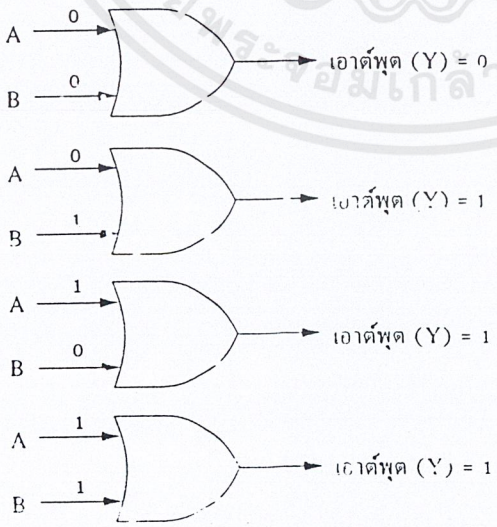
OR Gate สามารถมีอินพุตหลายๆ ตัวได้ แต่จะมีเอาท์พุตเพียงตัวเดียวเท่านั้น ถ้าเอาท์พุตเท่ากับ "1" แสดงว่ามีอินพุตตัวใดตัวหนึ่ง หรือหลายตัวเท่ากับ "1"



สัญลักษณ์ของ OR Gate ที่มีอินพุต 2 ตัว

อินพุต		เอาต์พุต
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

ตารางตามความเป็นจริง

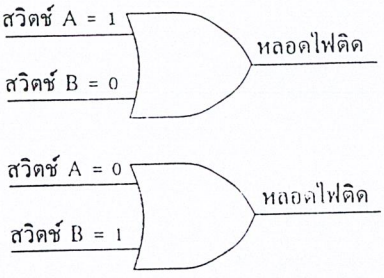


อินพุต		เอาต์พุต
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

ตารางตามความเป็นจริง

รูปที่ 2.31 สัญญาณของ OR Gate ที่มีอินพุต 2 ตัว

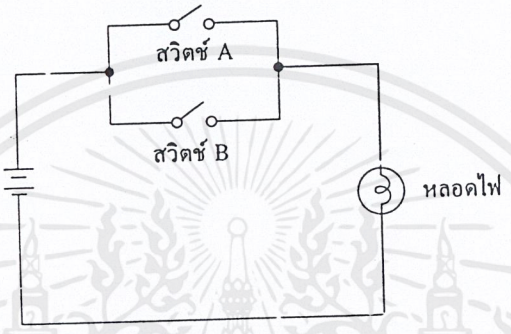
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



สวิตช์ A	สวิตช์ B	หลอดไฟ
เปิด (0)	เปิด (0)	ดับ (0)
เปิด (0)	ปิด (1)	ติด (1)
ปิด (1)	เปิด (0)	ติด (1)
ปิด (1)	ปิด (1)	ติด (1)

ตารางตามความเป็นจริง

รูปที่ 2.32 แสดงสภาวะของหลอดไฟ ณ สวิตช์



รูปที่ 2.33 การทำงานของ OR Gate หลอดไฟจะติดเมื่อสวิตช์ A หรือ B ติด

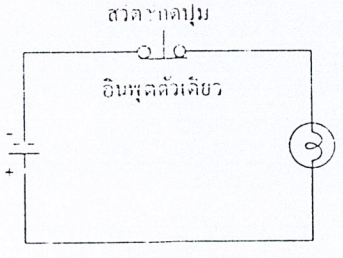
2.1.10.3 หลักการของ NOT Gate

NOT Gate จะไม่เหมือนกับ AND หรือ OR Gate คือ NOT Gate จะมีอินพุตเพียงตัวเดียวเท่านั้นถ้าเอาท์พุตเท่ากับ "1" แสดงว่าอินพุตเท่ากับ "0" ถ้าเอาท์พุตกับ "0" แสดงว่าอินพุตเท่ากับ "1"

A	NOT A
0	1
1	0

ตารางตามความเป็นจริงของ NOT

รูปที่ 2.34 สัญลักษณ์ของ NOT Gate

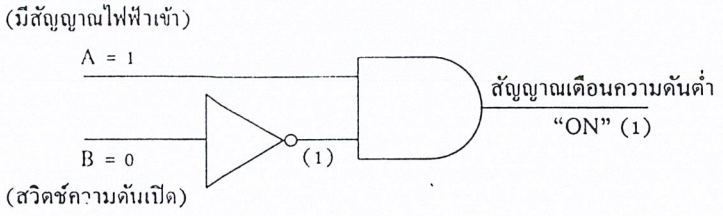


สวิตช์กดปุ่ม	หลอดไฟ
ไม่กดสวิตช์ (0)	ติด (1)
กดสวิตช์ (1)	ดับ (0)

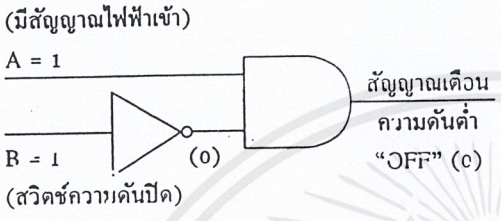
ตารางตามความเป็นจริง

รูปที่ 2.35 หลอดไฟจะติดถ้าสวิตช์ไม่ถูกกด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.36 ถ้ามีสัญญาณไฟฟ้าเป็น "1" และสวิตช์ความดันเปิดจะมีสัญญาณเตือนเกิดขึ้น



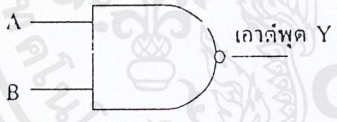
สวิตช์ความดัน	สัญญาณไฟฟ้า	สัญญาณเตือนของความดัน
0	1	1
1	1	0

ตารางตามความเป็นจริง

รูปที่ 2.37 ให้สัญญาณไฟฟ้าเป็น "1" และสวิตช์ความดันเป็น "1" จะไม่มีสัญญาณ

2.1.10.4 หลักการของ NAND Gate

NAND Gate จะทำงานตรงข้ามกับ AND Gate

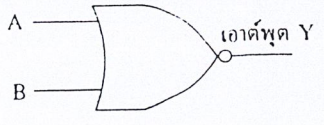


อินพุต		เอาต์พุต
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

รูปที่ 2.38 สัญลักษณ์ของ NAND Gate ที่มีอินพุต 2 ตัว

2.1.10.5 หลักการของ NOR Gate

NOR Gate จะทำงานตรงข้ามกับ OR Gate



อินพุต		เอาต์พุต
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

ตารางตามความเป็นจริงของ NOR

รูปที่ 2.39 สัญลักษณ์ของ NOR Gate ที่มีอินพุต 2 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.11 กฎการเขียนโปรแกรมภาษาแลดเดอร์

2.1.11.1 คำสั่งหน้าสัมผัสจำนวนไม่จำกัด

2.1.11.2 คำสั่งข้อมูลจำกัดจำนวนครั้งเดียว

2.1.11.3 คำสั่งส่งข้อมูลไม่มีหน้าสัมผัสกำกับไม่ได้

2.1.12 หลักการเขียนโปรแกรมภาษาแลดเดอร์

2.1.12.1โปรแกรมเข้าใจง่าย

2.1.12.2 แยกวงจรแลดเดอร์

2.1.12.3 เก็บสภาวะข้อมูลชั่วคราวในรีเลย์ช่วย

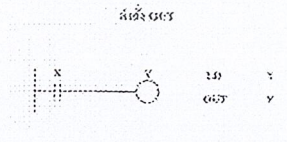
2.1.12.4 การใช้หน้าสัมผัสปกติเปิดและหน้าสัมผัสปกติปิด

2.1.13 การเขียนโปรแกรมภาษาแลดเดอร์

2.1.13.1 คำสั่ง LD

รูปที่ 2.40 คำสั่ง LD

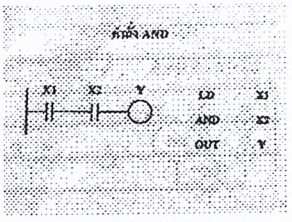
2.1.13.2 คำสั่ง OUT



รูปที่ 2.41 คำสั่ง OUT

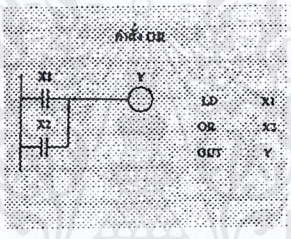
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.13.3 คำสั่ง AND



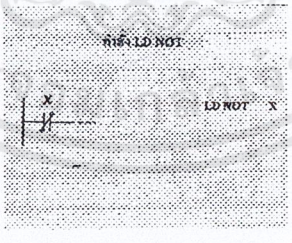
รูปที่ 2.42 คำสั่ง AND

2.1.13.4 คำสั่ง OR



รูปที่ 2.43 คำสั่ง OR

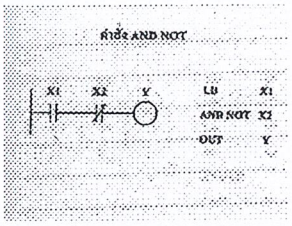
2.1.13.5 คำสั่ง LD NOT



รูปที่ 2.44 คำสั่ง LD NOT

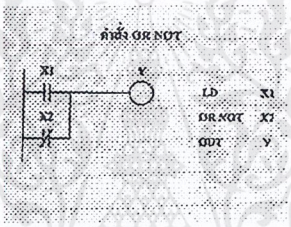
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.13.6 คำสั่ง AND NOT



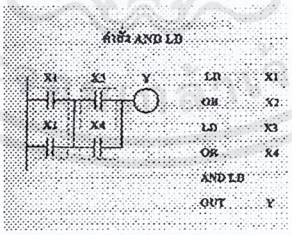
รูปที่ 2.45 คำสั่ง AND NOT

2.1.13.7 คำสั่ง OR NOT



รูปที่ 2.46 คำสั่ง OR NOT

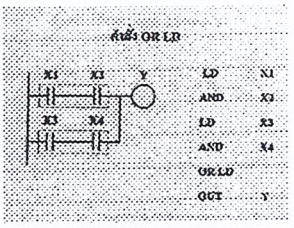
2.1.13.8 คำสั่ง AND LD



รูปที่ 2.47 คำสั่ง AND LD

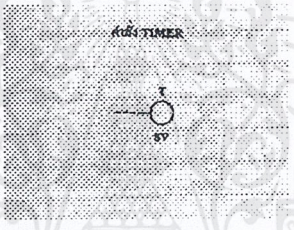
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.13.9 คำสั่ง OR LD



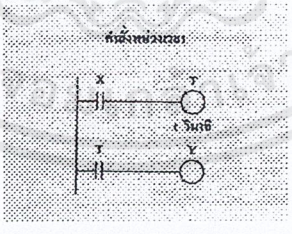
รูปที่ 2.48 คำสั่ง OR LD

2.1.13.10 คำสั่ง TIMER



รูปที่ 2.49 คำสั่ง TIMER

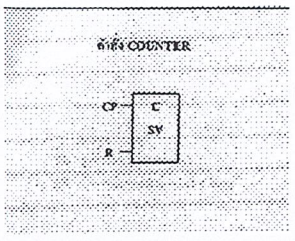
2.1.13.11 คำสั่งหน่วงเวลา



รูปที่ 2.50 คำสั่งหน่วงเวลา

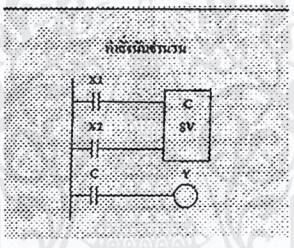
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.13.12 คำสั่ง COUNTER



รูปที่ 2.51 คำสั่ง COUNTER

2.1.13.13 คำสั่งนับจำนวน



รูปที่ 2.52 คำสั่งนับจำนวน

2.2 Delphi 5.0

Delphi เป็นเครื่องมือสำหรับสร้างแอพพลิเคชันสำหรับรันบน Windows 95/98/2000 ที่ผลิตโดยบริษัท Inprise (ชื่อเดิมก็คือ Borland) ซึ่งเป็นบริษัทที่แวดวงของนักพัฒนาแอพพลิเคชันรู้จักและยอมรับในตัวผลิตภัณฑ์เป็นอย่างดี

Delphi นั้นเป็นเครื่องมิพัฒนาแอพพลิเคชันแบบ Visual Programming (เหมือนกับ Visual Basic, Visual C++, Power Builder ฯลฯ) ซึ่งทำให้เราสามารถเห็นผลลัพธ์การทำงานไปพร้อมๆ กับการลงมือสร้างแอพพลิเคชัน

จุดเด่นที่สำคัญมากของความเป็น Visual Programming คือช่วยลดเวลาของการสร้างแอพพลิเคชันนั่นเพราะแทนที่เราจะไปทุ่มเวลาไปปรับแต่งส่วนติดต่อผู้ใช้ หรืองานที่ไม่จำเป็นหรืองานซ้ำๆ ซากๆ เราก็มอบภาระเหล่านี้ให้ Delphi เสีย สำหรับเราก็มุ่งเข้าไปแก้ปัญหาที่เป็นหัวใจการทำงานของแอพพลิเคชันดีกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ความสามารถของ Delphi 5.0

Delphi ได้พัฒนามาจนถึงเวอร์ชัน 5.0 ซึ่งประกอบไปด้วยเครื่องมือช่วยในการออกแบบ, สร้าง และทดสอบแอพพลิเคชันที่หลากหลาย ช่วยให้ผลงานออกมาได้อย่างรวดเร็ว, สะดวกสบายจนน่าประหลาดใจ สำหรับความสามารถของ Delphi 5.0 ที่เราน่าจะรู้จักได้แก่

2.2.1.1 สร้างแอพพลิเคชันจาก VCL

การสร้างแอพพลิเคชันแบบ Visual Programming นั้น เกิดจากการนำเอาออบเจ็คต่างๆ มาประกอบกันเป็นแอพพลิเคชัน ซึ่งอ็อปเจ็คเหล่านั้นส่วนใหญ่จะอยู่ในรูปของส่วนติดต่อผู้ใช้ (User Interface) ซึ่งทำให้เราสามารถสร้างแอพพลิเคชันได้อย่างรวดเร็ว ใช้งานได้ง่าย และสวยงาม

2.2.1.2 เครื่องมือสร้างแอพพลิเคชันที่เพียบพร้อม

Delphi ได้เข้ามาช่วยให้การพัฒนาแอพพิเคชันเป็นเรื่องที่ไม่ยุ่งยากอีกต่อไป เพราะมีการรวบรวมเอาเครื่องมือที่จำเป็นต่างๆ ไว้ให้ใช้งานกันอย่างครบถ้วน และเครื่องมือแต่ละตัวก็ยังมีความน่าใช้งานอีกด้วย

2.2.1.3 ความเร็วของแอพพลิเคชันที่สร้างจาก Delphi 5.0

Delphi นั้นได้ชื่อว่าเป็นคอมไพเลอร์โดยสมบูรณ์ในตัว ช่วยให้แอพพลิเคชันที่สร้างขึ้นเป็นไฟล์ .EXE ที่ใช้งานได้ทันที (ไม่ต้องเสียเวลาแปล เหมือนแอพพลิเคชันในลักษณะสคริปต์ เช่น Visual Basic ที่ต้องแปลทุกครั้ง) และใช้งานได้ในทุกเครื่อง โดยไม่จำเป็นต้องมีไฟล์พิเศษเพิ่มเติมเลย ทำให้แอพพลิเคชันที่สร้างจาก Delphi มีขนาดเล็ก กินทรัพยากรของระบบน้อย จึงทำงานได้รวดเร็วมาก

2.2.1.4 ความสามารถด้านการเขียนโปรแกรม

การเขียนโปรแกรมนั้นเป็นเรื่องของความมีระบบระเบียบโดยธรรมชาติอยู่แล้ว แต่เมื่อความซับซ้อนของโปรแกรมมีมากขึ้นทำให้การเขียนโปรแกรมเป็นเรื่องยุ่งยากอย่างหลีกเลี่ยงไม่ได้ ทำให้มีหลายๆ วิธีที่คิดขึ้นมาเพื่อลดความซับซ้อนลงซึ่ง Delphi ก็มีหลากหลายวิธีที่ช่วยให้เราเขียน และทดสอบโปรแกรมที่เขียนได้อย่างสะดวกสบาย

2.2.1.5 ความสามารถด้านการตรวจสอบโปรแกรม

เมื่อความซับซ้อนของโปรแกรมมีมากขึ้น ความผิดพลาดย่อมเพิ่มมากขึ้นตามตัว ทำให้เราต้องมีวิธีการที่ช่วยให้การเขียนโปรแกรมกลั่นกรองเอาข้อผิดพลาดออกไปให้มากที่สุด เพื่อการทำงานที่ถูกต้องของแอพพลิเคชัน ซึ่ง Delphi 5.0 มีความสามารถนี้อย่างเต็มเปี่ยม

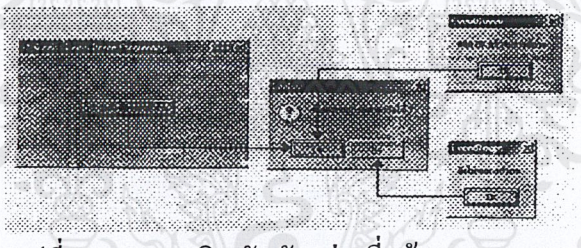
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อแอพพลิเคชัน กับผู้ที่เขียนโปรแกรมนั้นอยู่กันคนละที่ เช่นในการทำงานแบบเครือข่าย หรือในอินเตอร์เน็ตนั้น การตรวจสอบและแก้ไขข้อผิดพลาดต่างๆ จึงเป็นเรื่องที่ยุ่งยากไม่ใช้น้อย แต่นับว่ายังโชคดีที่ Delphi มีความสามารถในการตรวจสอบ และแก้ไขความผิดพลาดจากระยะไกลให้เราใช้งานด้วย

2.2.2 สร้างแอพลิเคชันแบบ Event-Driven

แอพพลิเคชันที่สร้างจาก Delphi นั้นมีวิธีการสร้างที่แตกต่างจากการเขียนโปรแกรมแบบเดิมๆ ที่เราเคยเรียนรู้มาก่อน หากเราอาจเคยเขียนโปรแกรมจากภาษา BASIC, C หรือ Pascal ก็จะเห็นว่ามีแนวคิดที่ค่อนข้างต่างกัน ซึ่งเราจะเรียกวิธีการที่สร้างแอพพลิเคชันด้วย Delphi ว่า Event-Driven (แปลว่า เหตุการณ์พาไป)

Event-Driven ที่จริงก็คือการเขียนโปรแกรมในลักษณะที่ว่า "ถ้ามีเหตุการณ์เกิดขึ้น เราจะจัดการกับมันอย่างไร" เช่นถ้าผู้ใช้คลิกที่ปุ่ม Exit เราจะทำอย่างไร เราอาจจะถามผู้ใช้ว่า แน่ใจแล้วนะว่าจะจบการทำงานของโปรแกรม ถ้าผู้ใช้ยืนยันก็จบไป แต่ถ้าไม่ก็ให้แอพพลิเคชันทำงานต่อไป เป็นต้น

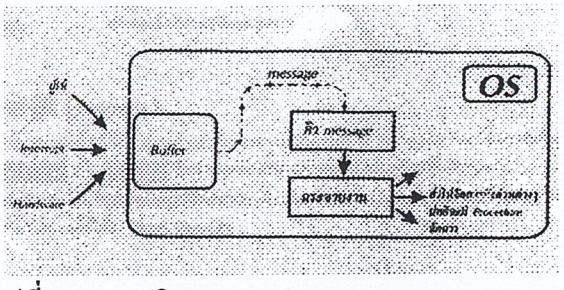


รูปที่ 2.53 แอพพลิเคชันตัวอย่างที่สร้างจาก Delphi

2.2.2.1 ระบบ Masage Loop ของ Windows

Event-Driven ถือได้ว่าเป็นเหมือนหลักการทำงานของ Windows ก็ว่าได้ ที่จริงแล้วเบื้องหลังการทำงานของ Windows ก็คือจะมีการรอรับการทำงานจากผู้ใช้, จากฮาร์ดแวร์ต่างๆ ส่งเหตุการณ์ต่างๆ เข้ามาสู่ Message Loop ซึ่งจะพิจารณาว่า มีเหตุการณ์อะไรเกิดขึ้น ซึ่งแต่ละเหตุการณ์ Windows ได้เตรียมวิธีการจัดการแต่ละเหตุการณ์เอาไว้แล้ว

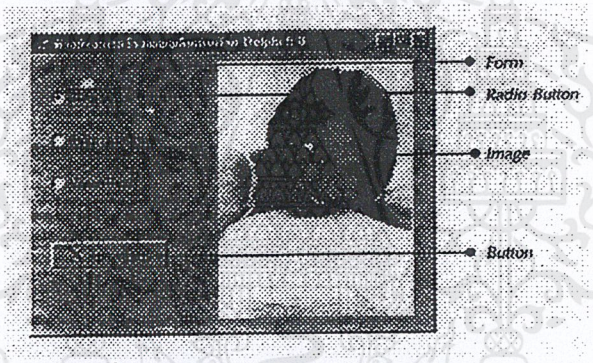
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.54 แนวคิดของ Massage Loop ของ Windows

2.2.3 รู้จักกับออบเจ็กต์

จากแอพพลิเคชันที่ผ่านมาเราจะเห็นว่า แอพพลิเคชันประกอบไปด้วย วินโดว์, ปุ่มกด หรือ ตัวเลือก ซึ่งเราจะเรียกสิ่งต่างๆ ที่นำมาสร้างเป็นแอพพลิเคชันนี้ว่า ออบเจ็กต์ (Oject)



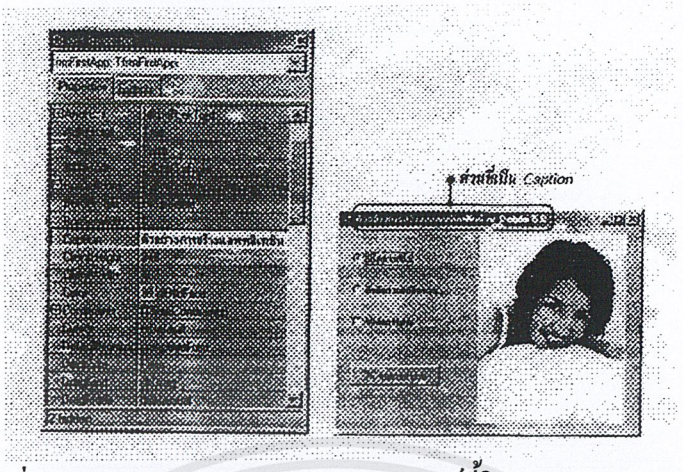
รูปที่ 2.55 องค์ประกอบของแอพพลิเคชัน

2.2.3.1 พรอพเพอร์ตี้ (Property)

จะเห็นได้ว่าแต่ละออบเจ็กต์มีลักษณะเฉพาะตัวที่แสดงความเป็นออบเจ็กต์ เช่น ปุ่มกด ก็จะมีข้อความที่อยู่บนตัวมัน (Caption) เช่น ปุ่มจบการทำงาน หรือ ปุ่มบอกเวลา ซึ่งเราเรียกลักษณะเฉพาะตัวนี้ว่าพรอพเพอร์ตี้ (Property)

ในออเจ็กต์ชนิดเดียวกันก็จะมีพรอพเพอตี้อย่างเดียวกัน แต่อาจจะมีค่าแต่ละพรอพเพอร์ตี้ที่แตกต่างกัน โดยเราสามารถกำหนดค่าพรอพเพอร์ตี้ได้ใน Oject Inspector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.56 ตัวอย่างการกำหนดค่าพรอพเพอร์ตี้ใน Object Inspector

ลักษณะการกำหนดพรอพเพอร์ตี้ใน Delphi นั้นมีอยู่ 3 รูปแบบ ได้แก่

1. การกำหนดค่าอย่างอิสระ เช่น พรอพเพอร์ตี้ Name, พรอพเพอร์ตี้ Caption เป็นต้น

2. การกำหนดค่าจากตัวเลือกที่มีให้ใน Object Inspector เช่น พรอพเพอร์ตี้ Cursor, พรอพเพอร์ตี้ BackColor เป็นต้น

3. การกำหนดค่าจากไดอะล็อกบ็อกซ์ เช่น พรอพเพอตี้ Font เป็นต้น

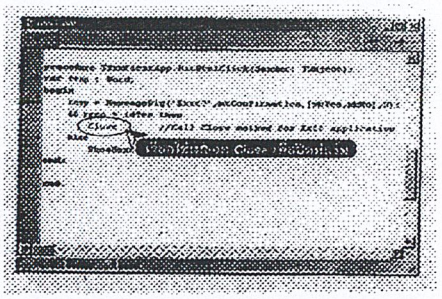
นอกจากการระบุค่าพรอพเพอร์ตี้จาก Object Inspector ซึ่งถือเป็นการกำหนดค่าก่อนการทำงานของแอพพลิเคชัน (Design-Time) แล้ว เรายังสามารถกำหนดค่า หรือเปลี่ยนค่าพรอพเพอร์ตี้ได้โดยการเขียนลงไปในโปรแกรมของเรา ซึ่งก็จะทำให้มีการกำหนดค่า หรือเปลียนแปลงค่าในขณะที่แอพพลิเคชันทำงาน (Run-Time) ได้เช่นกัน

2.2.3.2 เมธอด (Method)

นอกจากจะใช้พรอพเพอรตี้ในการบอกความแตกต่างของออบเจ็กต์แต่ละตัวแล้ว ยังมีอีกสิ่งหนึ่งที่ออบเจ็กต์แต่ละตัวมักจะมีนั่นก็คือเมธอด (Method)

เมธอดก็คือ ความสามารถในการทำงานอย่างใดอย่างหนึ่งของออพเจ็กต์ เช่น ออบเจ็กต์ Form มีความสามารถในการวาดวงกลมได้ เป็นต้น ในการเรียกใช้เมธอดเราจะเรียกโดยการเขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.57 เรียกใช้เมธอดของออบเจ็กต์ในโปรแกรม

2.2.3.3 อีเวนต์ (Event)

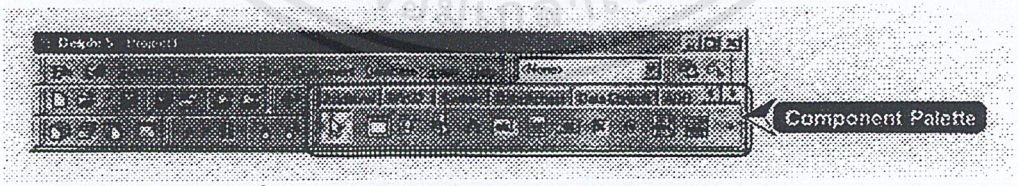
อีเวนต์ก็คือ เหตุการณ์ที่เกิดขึ้นกับแต่ละออบเจ็กต์ เช่น ปุ่ม ก็จะมีเหตุการณ์ที่ผู้ใช้คลิกเมาส์ (อีเวนต์ OnClick) หรือกรอบข้อความ จะมีเหตุการณ์เมื่อผู้ใช้กรอกข้อความ (อีเวนต์ OnKeyPress)

เราสามารถเขียนคำสั่งต่างๆ เพื่อจัดการกับเหตุการณ์ที่เกิดขึ้น โดยการเลือกเหตุการณ์ที่ต้องการจากแท็บ Event ใน Object Inspector

2.2.3.4 ออบเจ็กต์ กับ คอมโพเนนต์ของ Delphi

ข้อดีของการเขียนโปรแกรมเพื่อสร้างแอพพลิเคชันใน Delphi ก็คือ เมื่อเราสมมติสิ่งต่างๆ เป็นออบเจ็กต์ที่มีทั้งพรอพเพอร์ตี้, เมธอดแล้ว เราก็สามารถนำมาใช้งานได้เรื่อยๆ (Resuable Object) ทำให้ไม่ต้องเสียเวลามาสร้างกันใหม่ทุกครั้งที่สร้างแอพพลิเ คชัน

Delphi จะเรียกออบเจ็กต์ต่างๆ ที่สามารถนำกลับมาใช้ในการสร้างแอพพลิเคชันว่า คอมโพเนนต์ (Component) ซึ่งจะถูกเก็บไว้ใน Component Palette พร้อมให้เรานำมาใช้งาน

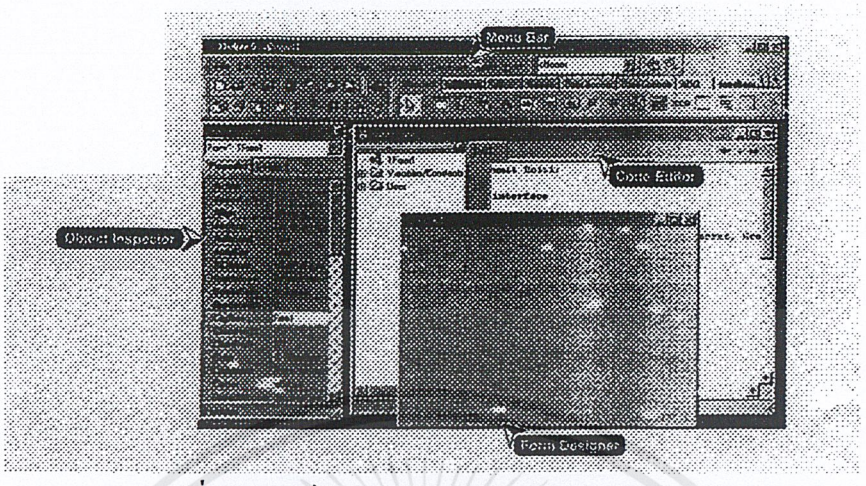


รูปที่ 2.58 คอมโพเนนต์ต่างๆ ใน Component Palette

2.2.4ส่วนประกอบการทำงานของ Delphi

การทำงานเพื่อสร้างแอพพลิเคชันกับ Delphi 5.0 นั้นเราจะอยู่ภายใต้สภาวะแวดล้อมการพัฒนาแบบครบครัน นั่นคือ ประกอบด้วยเครื่องมือชนิดต่างๆ ที่นำมาสร้าง และทดสอบแอพพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.59 องค์ประกอบการทำงานของ Delphi 5.0

2.2.4.1 Menu Bar

Menu Bar เป็นส่วนที่เก็บคำสั่ง เพื่อเรียกใช้งาน Delphi 5.0 ในรูปเมนู ซึ่งอยู่บนสุดของการแสดงผล

2.2.4.2 From Designer

From Designer เป็นส่วนที่เราออกแบบหน้าตาของฟอร์ม ซึ่งถือเป็นพื้นที่แสดงผลหลักที่เราสามารถนำเอาออบเจ็กต์ต่างๆ มาวางไว้ภายในให้เป็นแอพพลิเคชันตามที่เราต้องการได้

2.2.4.3 Object Inspector

เราสามารถกำหนดค่าพรอพเพอร์ตี้ของออบเจ็กต์ต่างๆ ผ่านทาง Object Inspector นอกจากนี้เรายังเริ่มต้นจัดการกับอีเวนต์ต่างๆ ของออบเจ็กต์ผ่านทาง Oject Inspector เช่นเดียวกัน

2.2.4.4 Component Palette

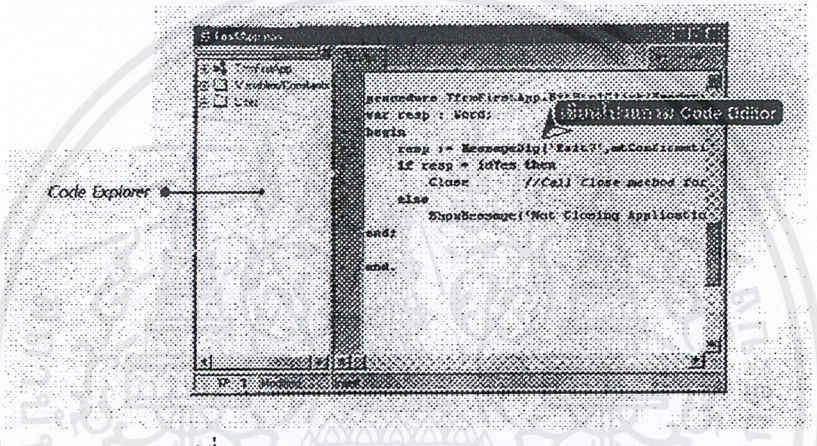
Component Palette คือส่วนที่เก็บคอมโพเนนต์ต่างๆ แยกเป็นหมวดหมู่เอาไว้ พร้อมให้เรานำไปประกอบในแอพพลิเคชันที่จะสร้างขึ้น และไม่ใช่จะมีเพียงแค่คอมโพเนนต์ที่ Delphi 5.0 เตรียมไว้ให้เท่านั้น เรายังสามารถสร้างคอมโพเนนต์มาใช้งานเอง แล้วบรรจุไว้ใน Component Palette ได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4.5 Code Editor

Code Editor เป็นส่วนที่เราใช้ในการเขียนคำสั่งต่างๆ เข้าไปเพื่อกำกับการทำงานของแอพพลิเคชันให้สอดคล้องกันซึ่งคำสั่งต่างๆ ในภาษาปาสคาลที่อยู่ใน Code Editor จะประกอบไปด้วยส่วนที่ Delphi เตรียมไว้ให้ และส่วนที่เราต้องเขียนเพิ่มเติมเข้าไป

ใน Code Editor เรามักจะเห็น Code Exploer ควบคู่ไปด้วย ซึ่ง Code Exploer จะช่วยให้เราเห็นตัวแปร, โปรแกรมย่อย, ออบเจ็กต์, พรอพเพอร์ตี้ เมธอดของออบเจ็กต์ต่างๆ ที่ประกอบกันเป็นแอพพลิเคชัน



รูปที่ 2.60 Code Explorer และ Code Editor

2.2.5 ขั้นตอนการสร้างแอพพลิเคชันเบื้องต้นกับ Delphi 5.0

เพื่อความเข้าใจในวิธีการสร้างแอพพลิเคชัน ตอนนี้เราจะมาทดลองสร้างแอพพลิเคชันด้วยตัวของเราทีละขั้นตอนดังต่อไปนี้

(หน้าตาของแอพพลิเคชัน)

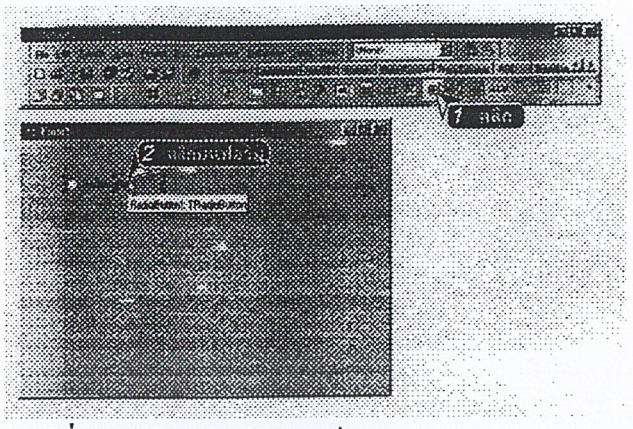
2.2.5.1 ขั้นแรก : ออกแบบการทำงาน และหน้าตาของแอพพลิเคชัน

ถือเป็นขั้นตอนแรกที่ต้องทำ ต้องนึกให้ดีว่าอยากจะให้หน้าของแอพพลิเคชันเป็นอย่างไรถ้า นึกไม่ออก หรือมีความซับซ้อน ก็วาดลงในกระดาษก่อน

2.2.5.2 ขั้นสอง : วางคอพโพเนนต์ที่ต้องการลงในฟอร์ม

ขั้นต่อมา เราก็นำเอาคอมโพเนนต์ต่างๆ จาก Component Palatte มาวางบนฟอร์มใน Form Designer โดยให้เราคลิกเลือกที่คอมโพเนนต์ที่ต้องการ จากนั้นมาคลิกที่ฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.61 เลือกคอมโพเนนต์จาก Component Palatte

จัดวางคอมโพเนนต์ต่างๆ ให้อยู่ในตำแหน่งที่ต้องการอย่างสวยงาม สังเกตว่าคอมโพเนนต์ต่างๆ ไม่ห่างหรือไม่เบียดเสียดกันเกินไป

2.2.5.3 ขั้นที่สาม : กำหนดพรอพเพอร์ตี้ให้แต่ละคอมโพเนนต์

เมื่อได้หน้าตาของแอพพลิเคชันแล้ว เราก็มากำหนดพรอพเพอร์ตี้ให้แต่ละคอมโพเนนต์ โดยใช้ Object Inspector กำหนดค่าที่ต้องการ หลังจากการกำหนดพรอพเพอร์ตี้ให้กับคอมโพเนนต์ต่างๆ แล้วจะได้หน้าตาแอพพลิเคชันดังนี้

2.2.5.4 ขั้นที่สี่ : เขียนคำสั่งกำกับการทำงานของแต่ละคอมโพเนนต์

เพื่อให้การทำงานร่วมกันของออบเจ็กต์ หรือคอมโพเนนต์ต่างๆ เป็นไปอย่างราบรื่น เราจำเป็นจะต้องเขียนคำสั่งขึ้นมากำกับการทำงาน โดยเราจะเขียนไว้ใน Code Editor

เราเรียก Code Editor มาใช้งานโดยเลือกเมนู View > Code Editor หรือใช้การดับเบิ้ลคลิกที่คอมโพเนนต์ใดๆ ในฟอร์ม

จากคำสั่งที่เราเขียนขึ้น เราสั่งให้มีการโหลดภาพตาราที่เราชื่นชอบ เมื่อคลิกที่ RadioButton แต่ละตัวและที่ปุ่ม จบการทำงาน ให้มีการสอบถามก่อนว่า ต้องการจบการทำงานหรือไม่ ถ้าคลิกปุ่ม YES ก็ให้สิ้นสุดการทำงาน แต่ถ้าคลิกปุ่ม NO ก็ให้ยังคงทำงานต่อไป

2.2.5.5 ขั้นที่ห้า : ทำการคอมไพล์

เพื่อให้คอมพิวเตอร์เข้าใจคำสั่งต่างๆ ที่เราเขียนขึ้น เราจะต้องทำการคอมไพล์ (Compile) คำสั่งที่เขียนขึ้น ซึ่งก็คือการแปลคำสั่งต่างๆ ที่เขียนขึ้น (Source Code) จากภาษาปาสคาล ซึ่งมนุษย์อย่างเราเข้าใจ และอ่านได้รู้เรื่อง ให้เป็นภาษาที่เครื่องคอมพิวเตอร์เข้าใจ (Object Code) และทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคอมไพล์ทำได้โดยเลือกเมนู Project > Compile ชื่อโปรเจ็กต์ ในการคอมไพล์นั้น Delphi จะตรวจสอบความถูกต้องของคำสั่งที่ใช้ให้ด้วย ถ้าผิดพลาดจะแจ้งให้ทราบ

2.2.5.6 ขั้นที่หก : ทดสอบการทำงานของแอพพลิเคชัน

เมื่อคอมไพล์เสร็จ ขั้นตอนถัดไปคือ ทดสอบการทำงาน ว่ามีความถูกต้องและตรงกับความต้องการของเราหรือไม่โดยเลือกเมนู Run > Run ซึ่งจะเป็นการสั่งให้แอพพลิเคชันที่สร้างขึ้นเริ่มทำงาน

เราสามารถลัดขั้นตอนในการคอมไพล์ให้เริ่มทำงานได้เลย แต่ที่จริงแล้ว Delphi จะคอมไพล์ก่อนจากนั้นจึงสั่งให้แอพพลิเคชันเริ่มทำงานตามที่เราต้องการ

2.2.5.7 ขั้นที่เจ็ด : บันทึกแอพพลิเคชันที่สร้างขึ้นเก็บไว้

เมื่อสร้างแอพพลิชันจนแล้วเสร็จ เราสามารถบันทึกไฟล์ต่างๆ เก็บไว้ เพื่อการเรียกใช้งาน หรือแก้ไขในครั้งต่อๆ ไปได้ ในการบันทึกจะเริ่มจากบันทึกรายละเอียดของฟอร์มและโปรแกรมที่เราเขียนขึ้นมา โดยคลิกปุ่ม SAVE หรือเรียกเมนู File > Save ซึ่งไฟล์ที่บันทึกจะมีนามสกุลเป็น .PAS

จากนั้นเราจะบันทึกโปรเจ็กต์ ซึ่งได้ชื่อว่าเป็นตัวแทนของการเรียกใช้แอพพลิเคชัน (เราจะได้เรียนรู้รายละเอียดเกี่ยวกับโปรเจ็กต์ในบทถัดไป) โดยเรียกเมนู File > Save Project As... ซึ่งโปรเจ็กต์ที่บันทึกจะมีนามสกุลเป็น .DPR

จากขั้นตอนทั้ง 7 ขั้นตอน จะเห็นว่าการสร้างแอพพลิเคชันให้น่าใช้งานไม่ใช่เรื่องยาก เพียงแต่ต้องฝึกฝนกันต่อไป

2.2.6 การใช้งานคอมโพเนนต์ลาเบล (Label)

คอมโพเนนต์ ลาเบล จะทำหน้าที่แสดงข้อความตามที่เราต้องการ ซึ่งข้อความดังกล่าวมักจะเป็นข้อความที่ไม่มีการเปลี่ยนแปลงแล้ว หรือไม่มีการรับค่าจากผู้ใช้งาน บางคนจึงมักเรียก ลาเบล ว่า "ป้ายข้อความ"

2.2.7 การใช้งานคอมโพเนนต์ อีดิท (Edit)

คอมโพเนนต์ อีดิท ทำหน้าที่รับข้อความจากผู้ใช้งานที่รับเข้ามา นอกจานี้ยังสามารถแสดงข้อความได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.8 การใช้งานคอมโพเนนต์ เมโม (Memo)

คอมโพเนนต์ เมโม นั้นมีความสามารถเช่นเดียวกับคอมโพเนนต์ อีดิท แต่ต่างกันที่เมโมสามารถแสดงข้อความได้มากกว่า 1 บรรทัด

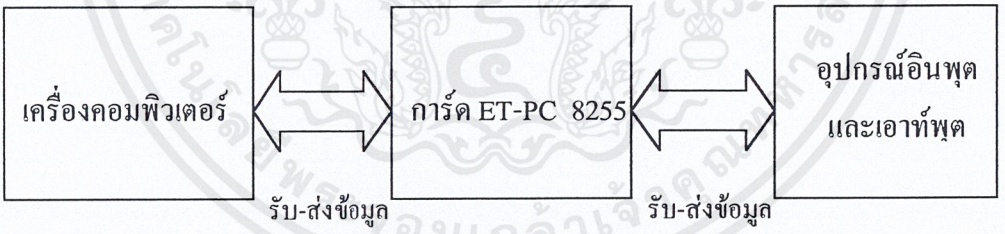
2.2.9 การใช้งานคอมโพเนนต์อิมเมจ (Image)

คอมโพเนนต์ อิมเมจ จำทำหน้าที่แสดงภาพที่ต้องแสดง

2.3 การติดต่อระหว่างเครื่องคอมพิวเตอร์ และ อุปกรณ์อินพุตและเอาท์พุต

ในการติดต่อระหว่างอุปกรณ์อินพุตและเอาท์พุตกับเครื่องคอมพิวเตอร์นั้นใช้การ์ด ET-PC 8255 เข้ามาช่วยในการติดต่อ โดยทำหน้าที่เป็นพอร์ตคอยช่วยในการส่งและรับข้อมูลจากตัวอุปกรณ์อินพุตและเอาท์พุต ตัวอินพุตจะใช้พอร์ตที่ 2 และ เอาท์พุตจะใช้พอร์ตที่ 1 รายละเอียดของการ์ด ET-PC 8255 จะอยู่ในภาคผนวก ข

2.3.1 บล็อกไดอะแกรมของการติดต่อระหว่างเครื่องคอมพิวเตอร์ และ อุปกรณ์อินพุตและเอาท์พุต



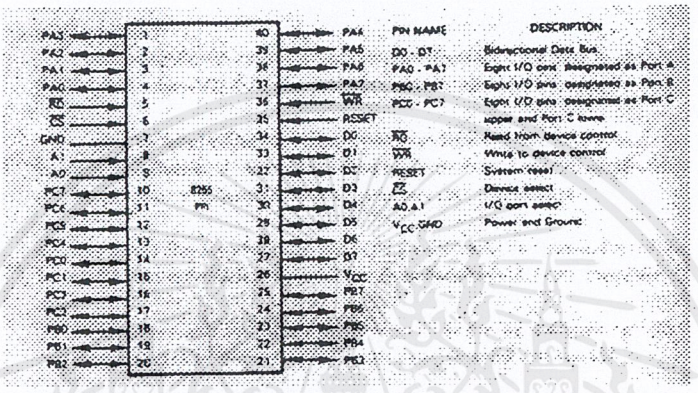
รูปที่ 2.62 บล็อกไดอะแกรมของการติดต่อระหว่างเครื่องคอมพิวเตอร์
และอุปกรณ์อินพุตและเอาท์พุต

2.3.2 ลักษณะเบื้องต้นของ ชิพเบอร์ 8255

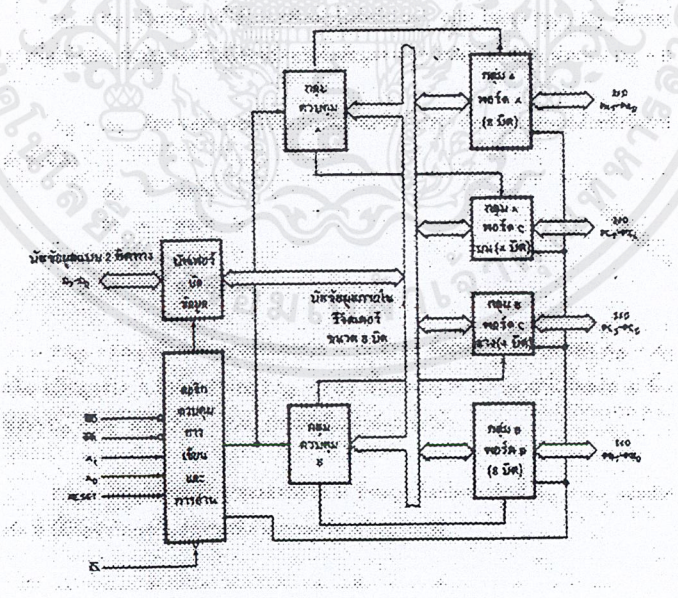
8255 นั้นเป็นไอซี LSI ขนาด 40 ขา จากรูปที่ 9 แสดงตำแหน่งของขาต่างๆ ทั้ง 40 ขา ส่วนรูป XXX แสดงแผนผังภายในของ 8255 ซึ่ง 8255 นี้มีพอร์ทสำหรับรับส่งข้อมูลอยู่ด้วยกัน 3 พอร์ท มีชื่อดังนี้คือ A, B และ C โดยพอร์ท C นี้จะแบ่งออกเป็น 2 ส่วน คือ พอร์ท C ล่าง(CLO) กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C บน (CHI) นอกจากนี้แล้วยังมีพอร์ทอีกพอร์ทหนึ่งซึ่งทำหน้าที่ควบคุมการทำงานของพอร์ท A, B และ C โดยการรับคำสั่งมาจาก CPU พอร์ทนี้เราเรียกว่าพอร์ทควบคุม (Control port) พอร์ทนี้จะใช้งานก็ต่อเมื่อ CPU ต้องการกำหนดลักษณะการทำงานของพอร์ท A, B และ C หรือต้องการเปลี่ยนแปลงหลังจากที่กำหนดไว้เดิม CPU จะส่งรหัสควบคุมมาทางดาต้าบัส (Data Bus) ให้แก่พอร์ทควบคุมนี้



รูปที่ 2.63 ตำแหน่งขาต่างๆ ของ 8255



รูปที่ 2.64 แผนผังภายในของ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดรหัสที่ใช้ในการควบคุมพอร์ทต่างๆ นี้จะกล่าวในตอนต่อไป ในทางปฏิบัติผู้ออกแบบระบบต้องนำรหัสควบคุมที่ได้มาตามข้อกำหนดของ 8255 นี้ไปใส่ในโปรแกรมเพื่อให้ CPU ทำการส่งรหัสควบคุมนี้มายังพอร์ทควบคุมเมื่อระบบนั้นเริ่มต้นทำงาน

2.3.3 หน้าที่ของขาต่างๆ

Chip Select ขานี้ใช้สำหรับรับสัญญาณจากภายนอกเพื่อใช้ในการเลือกว่าจะได้ 8255 ตัวนี้ทำงานหรือไม่ โดยแต่ถ้าขานี้ได้รับลอจิก 0 จะทำให้ 8255 เชื่อมต่อเข้ากับระบบบัสต่างๆ ของ CPU และพร้อมที่จะติดต่อกับ CPU ได้แต่เป็นลอจิก 1 มันก็จะปลดตัวเองออกจากระบบบัสของ CPU (โดยการเป็น Hi-Z)

Read Enable เป็นขาอินพุทที่จะรับสัญญาณจาก CPU ถ้าขานี้ได้รับลอจิก 0 และขณะนั้นขา Chip Select ต้องเป็นลอจิก 0 ด้วย 8255 จะทำการส่งข้อมูลจากพอร์ทที่ CPU ต้องการติดต่อด้วยนั้นได้แก่ CPU ทางดาต้าบัส

Write Enable มีหน้าที่การทำงานตรงข้ามกับขา Read Enable คือ ถ้าขา Write Enable นี้ได้รับลอจิก 0 (Chip Select ต้องเป็น 0 ด้วยเช่นกัน) 8255 จะรับข้อมูลจากดาต้าบัสของ CPU ส่งออกไปยังพอร์ทที่ CPU ส่งออกไปยังพอร์ทที่ CPU กำหนดไว้

RESET คือ ขาที่ทำหน้าที่ Reset 8255 เมื่อใดที่ 8255 ได้รับสัญญาณ Reset มันจะกลับเข้าสู่โหมดอินพุทคือทุกๆ พอร์ทจะเป็นอินพุทพอร์ท ขา RESET นี้ใช้เมื่อต้องการเคลียร์สถานะต่างๆ ของ 8255

D_0-D_7 คือขาข้อมูลที่ใช้ในการติดต่อรับส่งข้อมูลกับ CPU โดยขา D_0-D_7 นี้จะต่อเข้ากับดาต้าบัสของ CPU เพื่อให้ CPU ส่งข้อมูลออกไปยังพอร์ทหรือรับข้อมูลจากพอร์ทส่งให้แก่ CPU ผ่านทาง D_0-D_7 นี้

A_0-A_1 คือขาแอดเอรสที่ใช้ในการเลือกพอร์ทที่ CPU ต้องการจะติดต่อด้วยซึ่งมีความเป็นไปได้ทั้งหมด 4 ค่า ดังนี้คือ

00 = พอร์ท A

01 = พอร์ท B

10 = พอร์ท C

11 = พอร์ทควบคุม

PA_0-PA_7 เป็นขาสัญญาณของพอร์ท A ใน 8255 ซึ่งจะถูกเลือกโดยค่าของ A_0-A_1 และเมื่อพอร์ทนี้ถูกเลือกใช้ข้อมูลต่างๆ ก็จะถูกส่งผ่าน PA_0-PA_7 นี้ไปยัง D_0-D_7 (กรณีที่ให้พอร์ท A นี้เป็นอินพุทพอร์ท) หรือจาก D_0-D_7 มายัง PA_0-PA_7 (กรณีที่เป็นเอาท์พุทพอร์ท)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PB_0-PB_7 เป็นขาสัญญาณของพอร์ท B ซึ่งจะถูกเลือกโดยลอจิกที่ A_0-A_1 เช่นกันกับพอร์ท A และพอร์ท B นี้จะมีข้อจำกัดการรับส่งข้อมูลที่ต่างจากพอร์ท A ในบางกรณี

PC_0-PC_7 เป็นสายสัญญาณของพอร์ท C ซึ่งแบ่งออกเป็น 2 กลุ่มคือ PC_0-PC_3 และ PC_4-PC_7 โดยแต่ละกลุ่มสามารถแยกกันทำงานได้โดยอิสระกลุ่มหนึ่งอาจเป็นอินพุทพอร์ทในขณะที่อีกกลุ่มหนึ่งเป็นเอาท์พุทพอร์ทได้แต่จะทำงานพร้อมๆ กันโดยการเลือกด้วยลอจิกที่ A_0-A_1

2.3.4 การใช้งาน 8255

8255 นั้นแบ่งลักษณะการทำงานออกเป็น 3 โหมด (Mode) ด้วยกันคือ

โหมด 0 เป็นโหมดอินพุทหรือเอาพุทพอร์ทอย่างใดอย่างหนึ่ง ซึ่งทั้ง 3 พอร์ทคือ A, B และ C สามารถทำงานในโหมดนี้ได้

โหมด 1 เป็นโหมดอินพุทหรือเอาท์พุทพอร์ทอย่างใดอย่างหนึ่งเช่นกันแต่จะมีลักษณะการทำงานเป็นลักษณะของ Handshaking ในโหมดนี้ทำงานได้เฉพาะพอร์ท A และ B

โหมด 2 เป็นโหมด Bi-Direction คือเป็นได้ทั้งอินพุทและเอาท์พุทพอร์ทในเวลาเดียวกันและทำงานแบบ Handshaking เช่นเดียวกับโหมด 1 ในโหมดนี้ใช้ได้เฉพาะพอร์ท A เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและรูปแบบการทำงาน

3.1 ส่วนของแลดเดอร์ (Ladder)

ในส่วนนี้จะมีการสร้างปุ่มSpeedbutton ไว้ให้เลือกว่าจะสร้างวงจรLadderในรูปใดโดยที่

เป็น LD รับคำสั่ง Load

รูปที่ 3.1 คำสั่ง Load

เป็น LDNOTรับคำสั่ง Load not

รูปที่ 3.2 คำสั่ง Load not

เป็น OUT รับคำสั่ง Out

รูปที่ 3.3 คำสั่ง OUT

เป็น TIM รับคำสั่ง Timer

รูปที่ 3.4 คำสั่ง Timer

เป็น CNT รับคำสั่ง Counter

รูปที่ 3.5 คำสั่ง Counter

เป็น Line1 รับคำสั่ง Line1

รูปที่ 3.6 รูปเส้นนอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น Line2 รับคำสั่ง Line2

รูปที่ 3.7 รูปเส้นตั้ง

เป็น DEL รับคำสั่ง Delete

รูปที่ 3.8 รูปปุ่ม Delete

เมื่อเลื่อนเม้าส์ (Mouse)ไปกดที่ปุ่มดังรูปข้างต้น ปุ่มนั้นก็จะยุบลงไปเพื่อแสดงให้เห็นว่าตอนนี้อยู่ในสถานะใด หลังจากนั้นเลื่อนเม้าส์ไปคลิ๊ก (Click)ลงบนตารางอิมเมจ (Image)ที่อยู่ทางขวามือ จะมีหน้าต่างขึ้นมาอีกหน้าต่างหนึ่งเพื่อถามถึงค่าแอดเดรส(Address) หรือค่าของตัวแปรโดยจะขึ้นอยู่กับอิมเมจนั้นๆ ว่าเป็นแบบใด หลังจากนั้นก็จะ ปรากฏรูปบนอิมเมจดังกล่าว และค่าที่ได้ใส่ในตอนแรกก็จะแสดงไว้บนรูปนั้น

ในส่วนของคำสั่งและรูปแบบของคำสั่งที่เกี่ยวข้องกับการเขียนโปรแกรมเดลไฟล(Delphi) คือ
อิมเมจ(Image)การที่จะเปลี่ยนแปลงรูปในแต่ละรูป หลังจากกดปุ่มSpeedbutton ก็จะมีการเก็บค่าเป็นรหัสไว้สำหรับไปเรียกรูปที่มีรหัสตรงกับรหัสที่เก็บไว้

คำสั่งที่ใช้งานของอิมเมจส่วนใหญ่จะใช้ Image.Picture เป็นคำสั่งที่ใช้ในการเปลี่ยนรูปในอิมเมจแต่ละอิมเมจ โดยจะอยู่ในรูปแบบดังนี้

Image1.Picture := Ladder01.Picture;
อิมเมจที่หนึ่งก็จะเปลี่ยนรูปไปเป็นรูปที่ชื่อLadder01

ส่วนของหน้าต่างที่ถามถึงค่าแอดเดรส หรือค่าของตัวแปรก็จะใช้อิดิท(Edit)และลาเบล(Label)ในการรับค่าและแสดงค่า คำสั่งที่ใช้งานของอิดิทก็จะใช้Edit.Text เป็นคำสั่งที่อ่านค่าที่เขียนบนอิดิท และในส่วนของคำสั่งการใช้งานของลาเบลจะใช้ Label.Caption เป็นคำสั่งที่ใช้แสดงค่าที่ต้องการจะแสดง โดยทั้ง2 อย่างจะใช้รวมกันอยู่ในรูปดังนี้

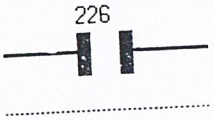
Label01.Caption := Edit1.Text;
ที่Label01 จะแสดงค่าที่ปรากฏอยู่บนEdit1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนของบูลีน(Boolean)

เป็นส่วนที่รับค่ามาจากอิมเมจที่แสดงอยู่ทางขวามือของหน้าจอหลัก ในส่วนนี้จะแสดงให้เห็นถึงการวางหรือการเขียนวงจรบนหน้าจอ สามารถที่จะแปลงเป็นภาษาของเครื่องพีแอลซี (PLC) ได้ เพื่อสะดวกแก่การอ่าน โดยที่

คำสั่ง LD มาจาก รูป

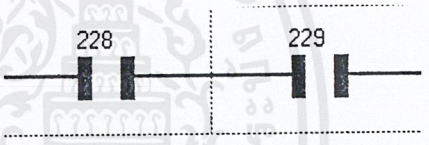


รูปที่ 3.9 คำสั่ง Load

คำสั่ง LDNOT มาจาก รูป

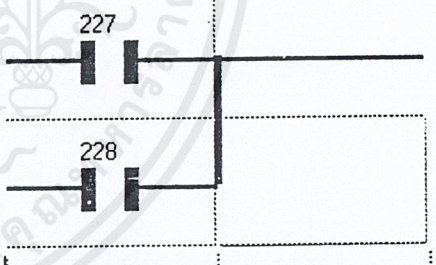
รูปที่ 3.10 คำสั่ง Load not

คำสั่ง AND มาจาก รูป



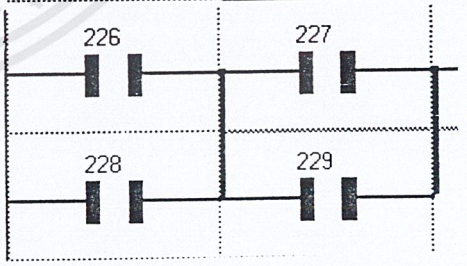
รูปที่ 3.11 คำสั่ง And

คำสั่ง OR มาจาก รูป



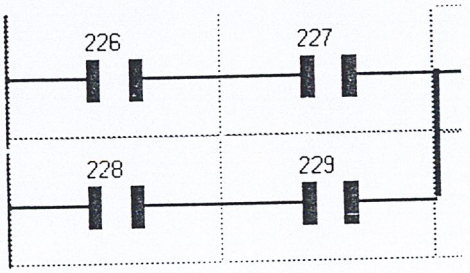
รูปที่ 3.12 คำสั่ง Or

คำสั่ง AND LD มาจาก รูป



รูปที่ 3.13 คำสั่ง And Load

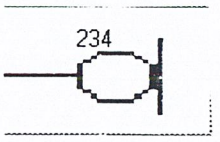
คำสั่ง OR LD มาจาก รูป



รูปที่ 3.14 คำสั่ง Or Load

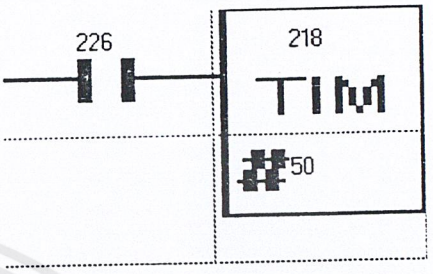
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง OUT มาจาก รูป



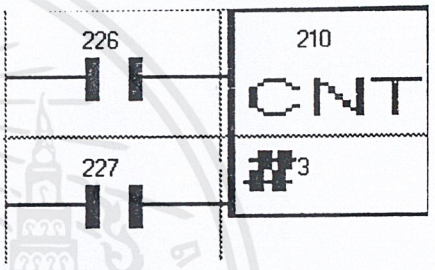
รูปที่ 3.15 คำสั่ง Out

คำสั่ง TIM มาจาก รูป



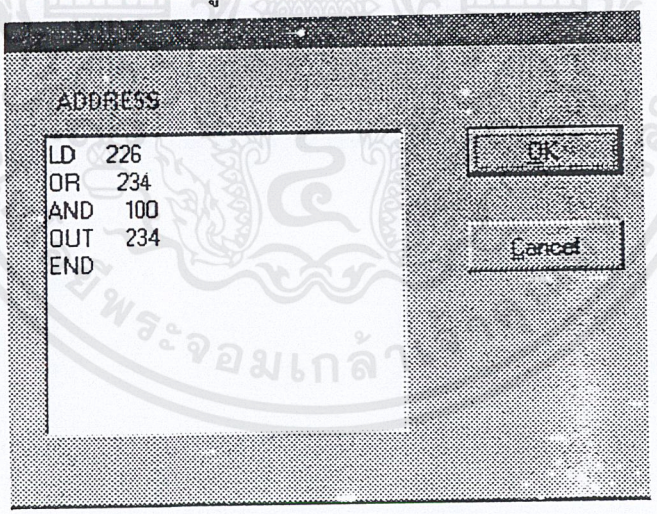
รูปที่ 3.16 คำสั่ง Timer

คำสั่ง CNT มาจาก รูป



รูปที่ 3.17 คำสั่ง Counter

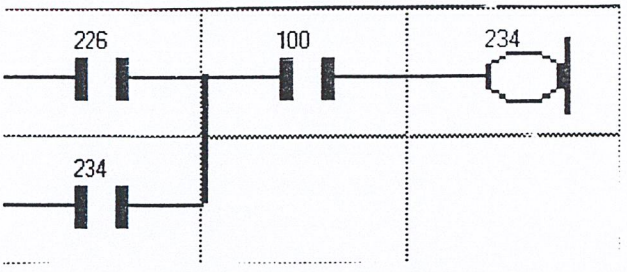
โดยจะแสดงไว้ในหน้าจอของบูลีนดังนี้



รูปที่ 3.18 หน้าจอบูลลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมาจากวงจรดังนี้



รูปที่ 3.19 รูปวงจรแลดเดอร์

การทำงานภายในก็จะมีการอ่านค่าของรหัสหรือจะเรียกว่าอินเด็กส์ (Index) อ่านค่าที่เก็บไว้ภายในการโชว์ (Show)ในแต่ละอิมเมจ เช่น อิมเมจที่โชว์รูปที่เป็น LD ก็จะมีการเก็บค่าอินเด็กส์ไว้คือ 01

หลังจากการกดดูBoolean โปรแกรมก็จะทำการอ่านค่าที่เก็บไว้ แล้วก็จะทำการเขียนลงบนเมโม(Memo) โดยโปรแกรมก็จะทำการอ่านและเปรียบเทียบกับรูปแบบว่า การเขียนวงจรLadder แบบนี้อยู่ในรูปแบบใด หลังจากนั้นก็จะเขียนลงบนเมโมว่า เป็น LD,AND,OR ,OUT เป็นต้น และตามด้วยค่าแอดเดรสหรือค่าของตัวแปร

โดยคำสั่งในการใช้งานของเมโม ที่ใช้ในการเขียน คือ

Memo1.Lines.Append(ค่าที่ต้องการใส่)

Memo1 ก็จะมีการเพิ่มทีละบรรทัด และจะเพิ่มได้ดีเมื่อใช้คำสั่งนี้ร่วมกับ คำสั่ง For โดยที่รูปแบบของคำสั่ง For เป็นดังนี้

```
For I:=0 to 10 do
(รูปแบบในการเขียนที่ต้องการกระทำ)
```

และนำคำสั่งทั้งสองมารวมกันจะเป็นดังนี้

```
For I:=0 to 10 do
Memo1.Lines.Append(ค่าที่ต้องการเขียน);
```

คำสั่งนี้ก็จะทำการเขียนในเมโม 10 ครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนของคอมไพล์(Compile)

ในส่วนนี้จะเป็นการตรวจสอบการเขียนวงจรLadder ว่าวงจรที่เขียนวางอยู่ในรูปแบบที่ถูกต้องหรือไม่ โดยจะมีการกำหนดรูปแบบที่ถูกต้องไว้ ซึ่งข้อมูลนี้ได้อ้างอิงจากหนังสือที่ใช้ในการเขียนLadder และได้มีการกำหนดไว้เผื่อหลายๆกรณีที่จะทำให้การเขียนวงจรLadder ผิดพลาด

หลังจากกดปุ่มคอมไพล์แล้ว ถ้าในกรณีที่วงจรที่เขียนอยู่ในรูปแบบที่ถูกต้องแล้ว ก็จะขึ้นหน้าต่างอีกอันหนึ่ง บอกว่าถูกต้องแล้ว แต่ในกรณีที่วงจรที่เขียนอยู่ในรูปแบบที่ไม่ถูกต้อง ก็จะขึ้นหน้าต่าง บอกว่าวางรูปแบบผิดและบอกว่าผิดตรงที่ใด

3.4 ส่วนของรัน(Run)

หลังจากที่ผู้ใช้เขียนโปรแกรมขึ้นมาและทำการคอมไพล์ เรียบร้อยแล้ว และ ดูคำสั่งภายในบูลลีน ก็ต้องมากดปุ่ม รัน เพื่อส่งและรับสัญญาณ จากอุปกรณ์อินพุต และ เอาท์พุต

หลักการของปุ่มตัวนี้จะทำหน้าที่ไปเลือกให้ตัวไทม์เมอร์ ซึ่งเวลาของมันจะเป็นเวลาที่เครื่องทำการสแกนชุดคำสั่งนั้นๆ หรือ ที่เรียกว่า สแกนไทม์ (Scan Time) โดยคำสั่งในตัวไทม์เมอร์ก็จะวนเช็คคำสั่งแต่ละบรรทัดว่าเป็นคำสั่งชนิดใด และ มีแอดเดรสเท่าใด เพื่อ ที่จะส่งข้อมูลออกไปติดต่อกับ อุปกรณ์อินพุตและเอาท์พ

3.5 ส่วนของอุปกรณ์อินพุตและเอาท์พุต

ตัวอุปกรณ์อินพุตจะต้องมีสวิตช์กดติด-กดดับจำนวนทั้งหมด 8 ช่องสัญญาณ เพื่อส่งสัญญาณอินพุต 5 โวลต์เข้ามาใน 8255 เพื่อให้คอมพิวเตอร์สามารถที่จะหาสัญญาณนั้นได้ โดยเมื่อไม่กดสวิตช์ มันจะส่งสัญญาณ 5 โวลต์ออกมา แต่เมื่อกด สัญญาณจะติกลงเป็น ศูนย์ ฉะนั้นในการเขียนโปรแกรมต้องทำการกลับค่าของมันก่อนจะนำไปใช้ในการคำนวณ

ตัวอุปกรณ์เอาท์พุตจะมีจำนวนทั้งหมด 8 ช่องสัญญาณ โดยสัญญาณเอาท์พุตจะส่งออกมาทาง 8255 เป็นสัญญาณ 5 โวลต์ โดยจะแสดงผลที่ หลอดไฟ LED และยังสามารถนำสัญญาณนี้ไปควบคุมการทำงานของรีเลย์ 12 โวลต์ เพื่อไปเป็นสวิตช์ควบคุมอุปกรณ์อื่นอีกทีหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

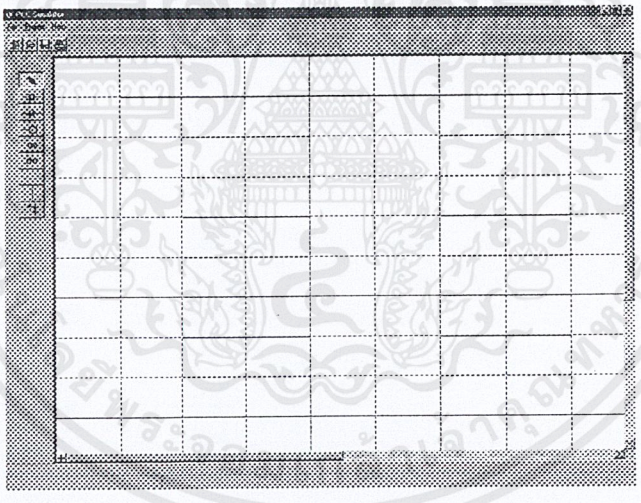
บทที่ 4
การใช้งานและผลการทดลอง

4.1 การใช้งาน

ในการใช้งานนั้นผู้ใช้จะสามารถใช้แอดเดรสได้ตั้งแต่ 1-241 โดยที่

226-233	จะเป็นแอดเดรสของอินพุต
234-241	จะเป็นแอดเดรสของเอาท์พุต
218-225	จะเป็นแอดเดรสของไทม์เมอร์
210-217	จะเป็นแอดเดรสของเคาน์เตอร์
1-209	จะเป็นแอดเดรสที่ใช้งานทั่วไป

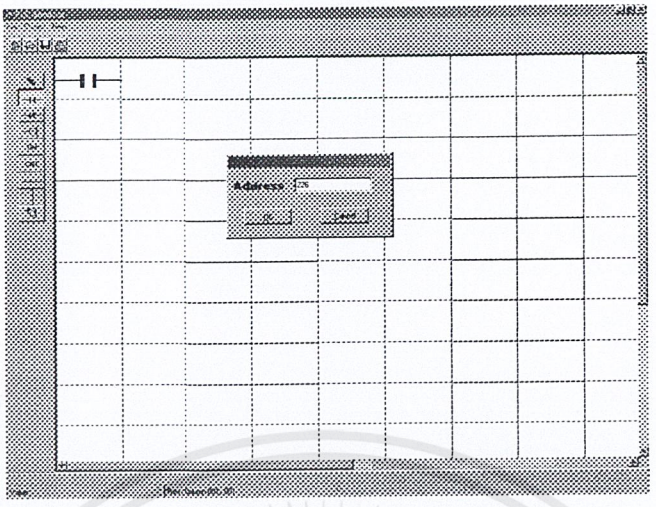
4.1.1 เมื่อทำการเปิดโปรแกรมขึ้นมาก็จะมีหน้าจอหลักดังนี้



รูปที่ 4.1 หน้าจอหลัก

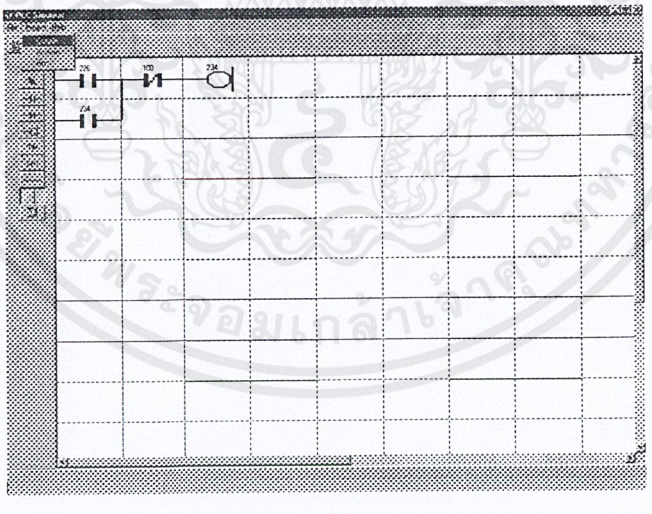
4.1.2 เราสามารถเลือกกดปุ่มคอมโพเนนต์ทางซ้ายมือได้แล้วมากดที่ อิมเมจ ทางขวามือก็จะปรากฏหน้าจอเพื่อที่จะถามถึง แอดเดรส ที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



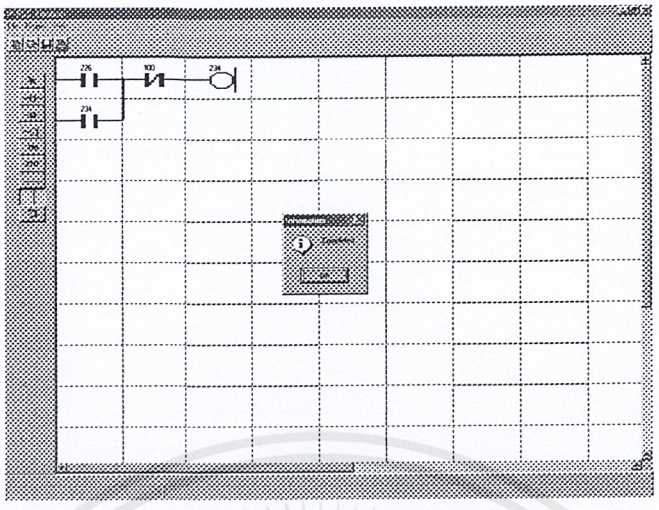
รูปที่ 4.2 หน้าจอแอดเดรส

4.1.3 หลังจากเขียนวงจรเสร็จแล้วสามารถตรวจสอบวงจรว่ามีข้อผิดพลาดหรือไม่คือวงจรตรงตามรูปแบบที่ถูกต้องหรือไม่ ถ้าไม่ถูกต้อง ก็จะบอกว่ามีข้อผิดพลาดอย่างไร ที่ใด ถ้าถูกต้อง ก็จะสามารถเข้าไปดูหน้าจอบูลลีนได้

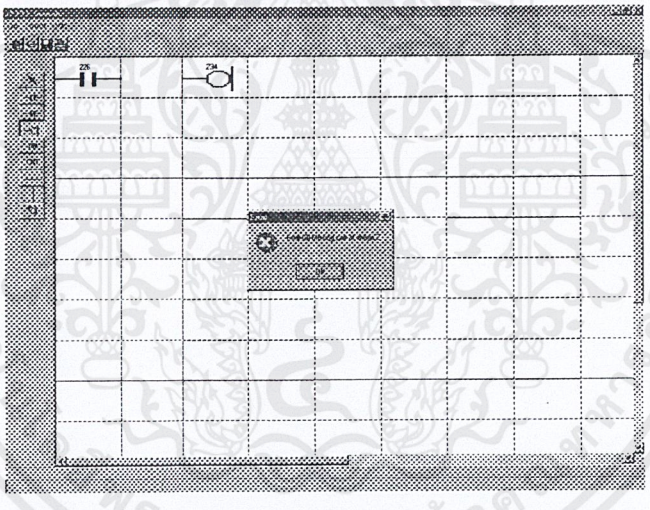


รูปที่ 4.3 การคอมไพล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



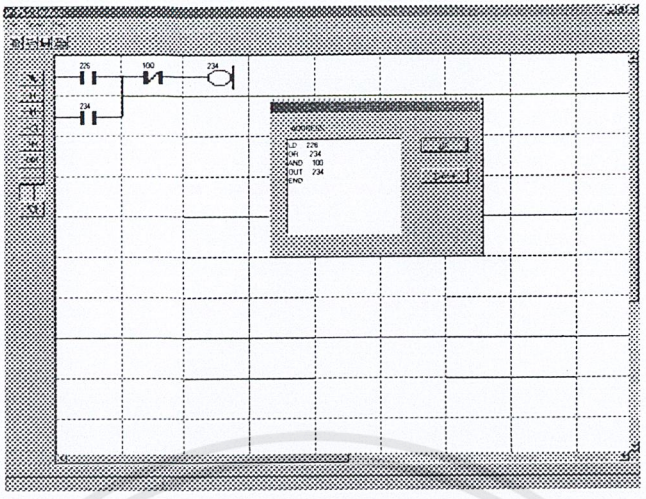
รูปที่ 4.4 การเขียนวงจรที่ถูกต้องตามรูปแบบ



รูปที่ 4.5 การเขียนแลดเดอร์ที่ไม่ถูกต้องตามรูปแบบ

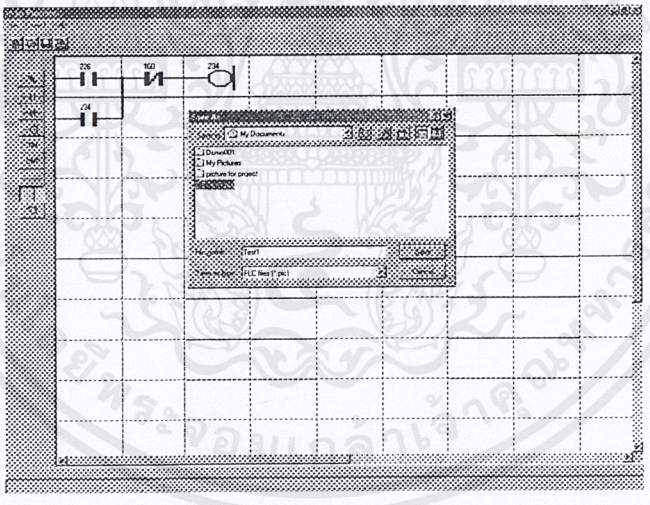
4.1.4 เมื่อทำการคอมไพล์ผ่านเรียบร้อยแล้วก็จะสามารถเข้าไปดูในส่วนของบูลลีนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงหน้าจอบูลลีน

4.1.5 การเซฟ และ การโอเพน สามารถทำได้โดยเลือกจากเมนูบาร์ข้างบน

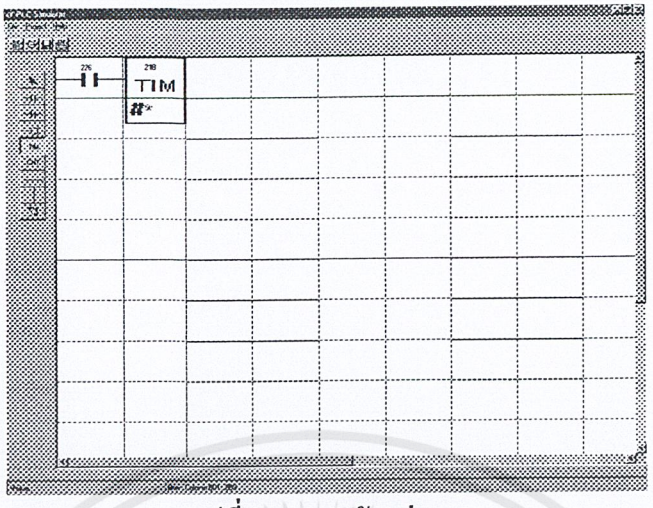


รูปที่ 4.7 การเซฟวงจรแลดเดอร์

4.2 ตัวอย่างการใช้จริง และผลที่ได้

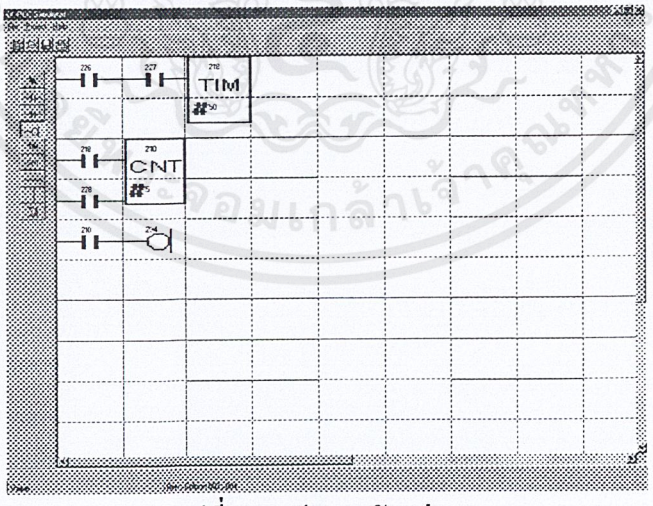
4.2.1 เมื่อป้อนคำสั่งแลดเดอร์ ดังรูปที่ 4.8 และทำการคอมไพล์เรียบร้อยและดูคำสั่งบูลลีนแล้ว ต่อมาเราก็ทำการรัน เพื่อนำเอาสัญญาณที่ได้จากการโปรแกรมออกมาทางอุปกรณ์ อินพุต และเอาท์พุต โดยที่ เมื่อกดที่ อินพุตที่ 226 แล้ว เอาท์พุต ที่ 234 ก็จะทำงาน (หลอดไฟดวงแรกสว่าง) แล้วเมื่อ ไม่มีสัญญาณอินพุตที่ 226 เข้ามา เอาท์พุต ก็ยังคงทำงานอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 วงจรตัวอย่าง 1

4.2.3 เมื่อป้อนคำสั่งแลดเดอร์ดังรูปที่ 4.9 ทำการคอมไพล์และดูคำสั่งบูลลีนเป็นที่เรียบร้อยแล้วก็ทำการส่งสัญญาณออกมาติดต่อกับอุปกรณ์ อินพุต และเอาท์พุต โดยเมื่อมีสัญญาณ มาจากแอดเดรสที่ 226 และ 227 แล้ว ไทม์เมอร์ที่ 218 จะทำการนับเวลา เป็นเวลา 5 วินาที หลังจากนั้น จะมีสัญญาณที่ แอดเดรส 210 แล้วไปทำให้ ตัวเคาน์เตอร์ 210 ทำการนับลงจาก 5 จนถึง 0 (เมื่อสัญญาณที่ แอดเดรส 210 ติดดับ เป็นจำนวน 5 ครั้ง) เมื่อมันนับจนถึงศูนย์แล้ว มีสัญญาณที่แอดเดรส 210 ไป ทำให้เอาท์พุตที่ 234 มีสัญญาณ นั่นคือมีไฟติดที่ ดวงที่ 2



รูปที่ 4.9 รูปวงจรตัวอย่าง 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลองและข้อเสนอแนะ

5.1 ปัญหาที่พบ

5.1.1 การทำคำสั่ง บูลลีน และ คอมไพล์ มักจะพบข้อผิดพลาดอยู่บ่อยๆ เนื่องมาจากต้องเช็คโปรแกรมที่ผู้ใช้ป้อนเข้ามาโดยโปรแกรมที่ป้อนเข้าจะมีหลากหลายรูปแบบมาก

5.1.2 โปรแกรมไม่เป็นเอกภาพ เนื่องจากมีความซับซ้อนสูงมาก

5.1.3 ขอบเขตของผลที่จะแสดงกว้างมาก อีกทั้งมีความหลากหลายในการนำเสนอ

5.1.4 การเขียนโปรแกรมมักพบปัญหาผิดพลาดแบบ Logical Error ซึ่งทำให้เสียเวลาในการหาจุดผิดพลาด เพราะไม่รู้จุดผิดที่แน่นอน

5.1.5 รูปแบบความน่าจะเป็นไม่มีรูปแบบตายตัว

5.1.6 ในการ รัน เพื่อที่จะทดสอบกับอุปกรณ์ อินพุต และ เอาท์พุต มักจะพบข้อผิดพลาดคือผลไม่ออกมาเป็นตามต้องการ โดยจะมาจากการที่อุปกรณ์ อินพุต และ เอาท์พุต มันจะมีการดีเลย์ข้อมูล ในการรับส่งกับเครื่องคอมพิวเตอร์

5.1.7 การใส่คอมโพเนนต์เข้ามาในตัวโปรแกรมเยอะๆ จะมีผลต่อการรันตัวโปรแกรมจะทำให้โปรแกรมประมวลผลช้า

5.1.8 การส่งค่าตัวแปรผ่านไปในฟอร์มต่างๆ ทำได้ยาก

5.2 การแก้ไข

5.2.1 ทำให้เกิดการวนลูปทุกครั้งที่มีการป้อนอินพุต

5.2.2 ลดขนาดของโปรแกรมให้เล็กลงที่สุดเท่าที่จะทำได้

5.2.3 กำหนดวิธีใช้งานที่ตายตัว

5.2.4 พยายามใช้ตัวแปรที่มีชื่อที่แน่นอน เข้าใจง่าย

5.2.5 มีการเช็คทดลองอยู่เรื่อยๆ ยิ่งมากเท่าไรยิ่งดี

5.2.6 ในการรันออกเอาพุตใช้ ไทม์เมอร์ในการแสกนไทม์เพียงตัวเดียว เพื่อลดการดีเลย์ของการรับส่งข้อมูลลง

5.2.7 พยายามใช้คอมโพเนนต์ให้น้อยที่สุดเท่าที่จะเป็นไปได้เพื่อให้โปรแกรมทำงานได้เร็วขึ้น

5.2.8 ในการส่งค่าตัวแปรไปยังฟอร์มต่างๆ ทำโดยใช้ เมโม มารับค่าแทนตัวแปรนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 สรุปผลการทดลอง

การจำลองเครื่องพีแอลซี เป็นการเลียนแบบการทำงานของพีแอลซีได้จริงในระดับที่น่าพอใจ โดยสามารถทำงานคำสั่งพื้นฐานของพีแอลซีได้ ในส่วนของโปรแกรมจะมีส่วนของ แลดเดอร์ กับ ส่วนของ บูลลีน ซึ่งจะเป็นผลดีกับผู้ที่เริ่มศึกษา พีแอลซี เพราะสามารถเห็นภาพได้อย่างชัดเจน ส่วนของการ คอมไพล์ก็จะเป็นตัวช่วยเช็คข้อผิดพลาดของการโปรแกรมของผู้ใช้ ทำให้การใช้งานมีความถูกต้องมากยิ่งขึ้น และในส่วนของการ รับส่งข้อมูล กับอุปกรณ์ อินพุต และ เอาท์พุต จะทำให้การจำลองเครื่องพีแอลซี นั้นมีความสมจริงมากขึ้น โดยสามารถนำสัญญาณออกมาควบคุมอุปกรณ์แบบ ซีเควนส์ได้จริง พื้นฐานของตัวโปรแกรมเองนั้น ก็จะมีความสามารถที่จะเซ็ฟ และ โอเพน เพื่อเก็บข้อมูลการเขียนโปรแกรมนั้นๆ ไว้ โดยความสามารถเหล่านี้จะเป็นแนวทางในการพัฒนา การจำลองเครื่องพีแอลซี ให้มีความสมจริงได้มากขึ้น นั้นคือมีความสามารถในการทำคำสั่งได้ใกล้เคียงกับพีแอลซี จริงๆ หรือ มีความสามารถพิเศษอื่นๆ ได้ในอนาคตต่อไป

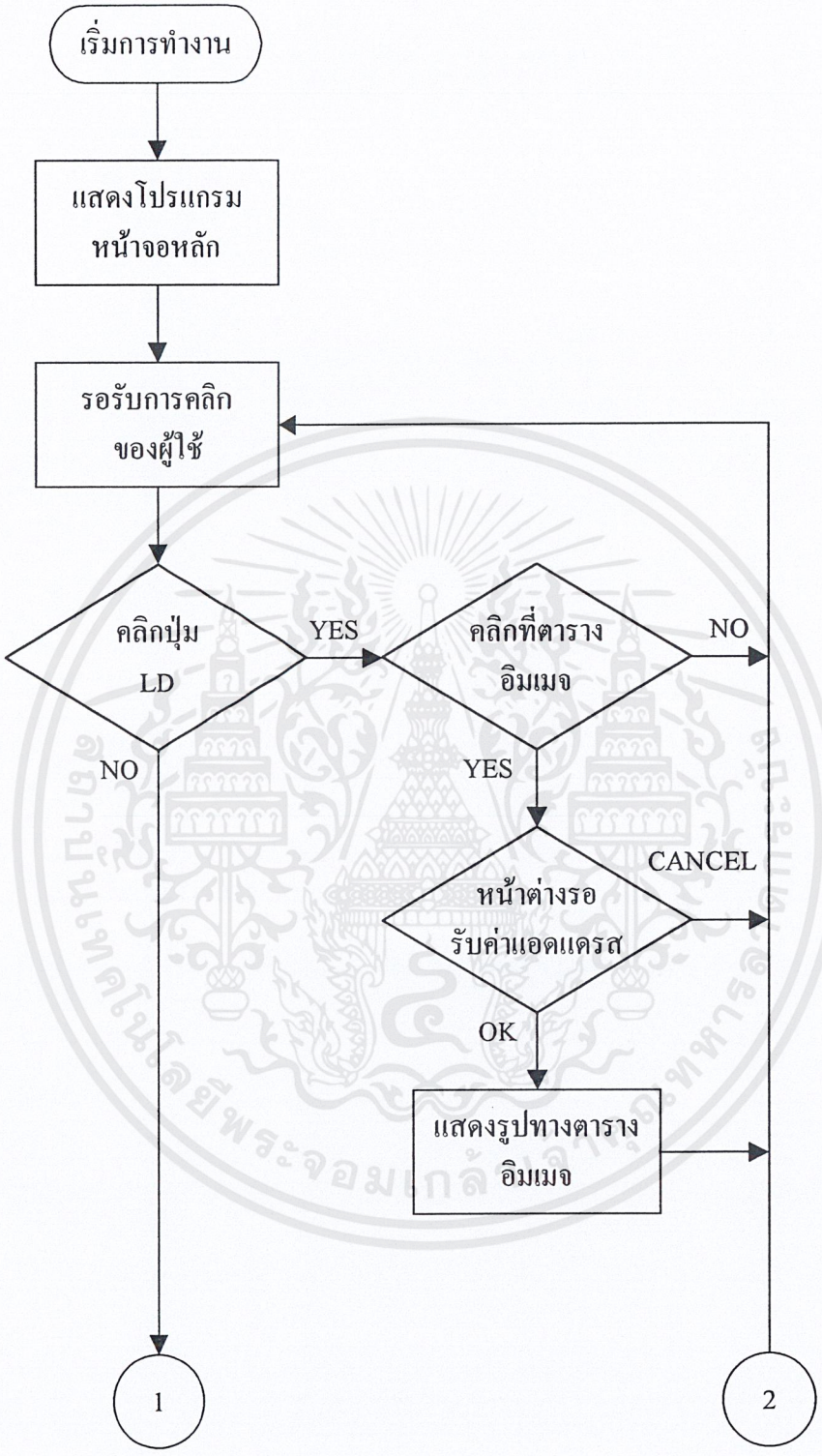
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

ภาคผนวก ก

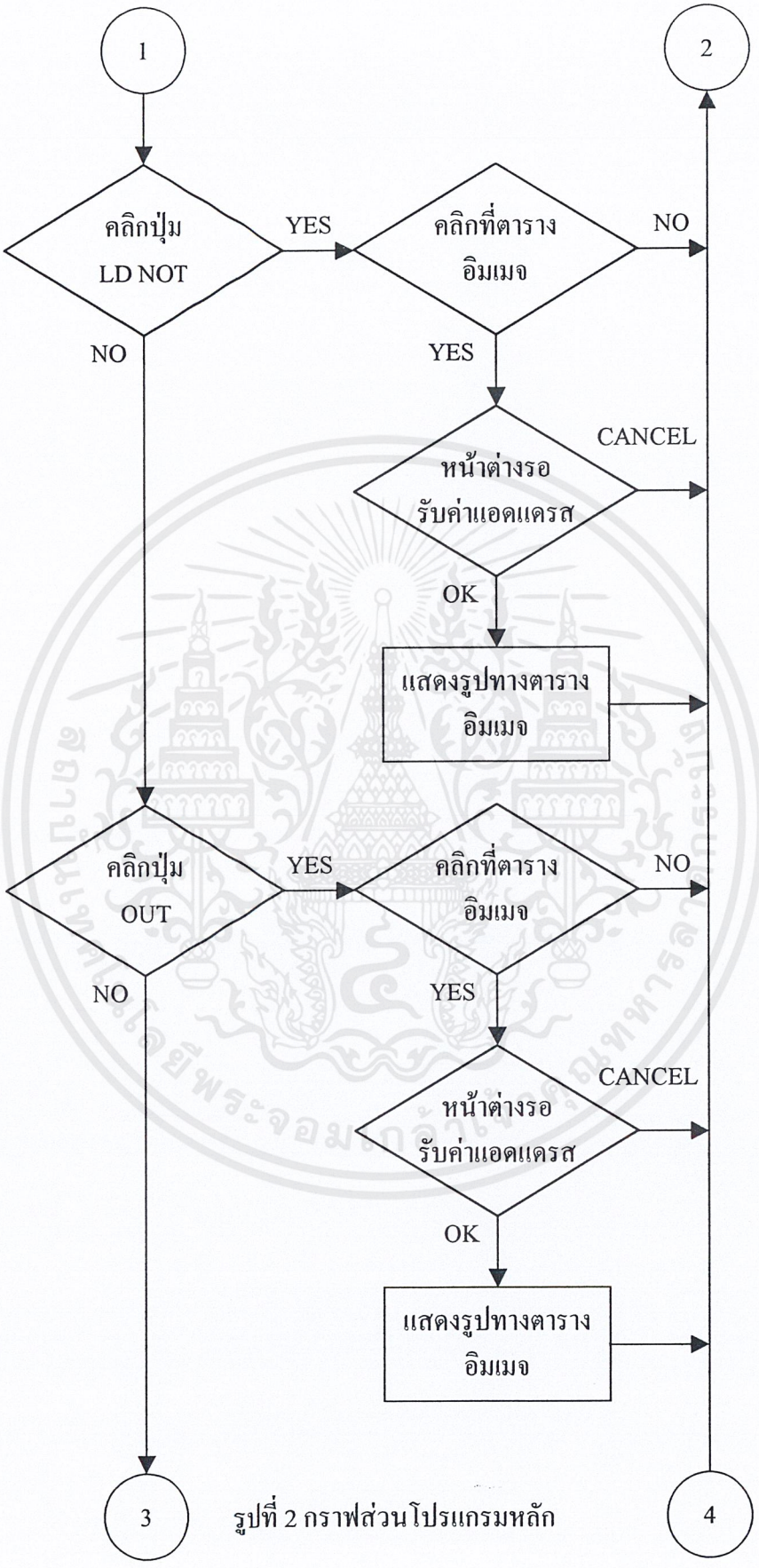
กราฟการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



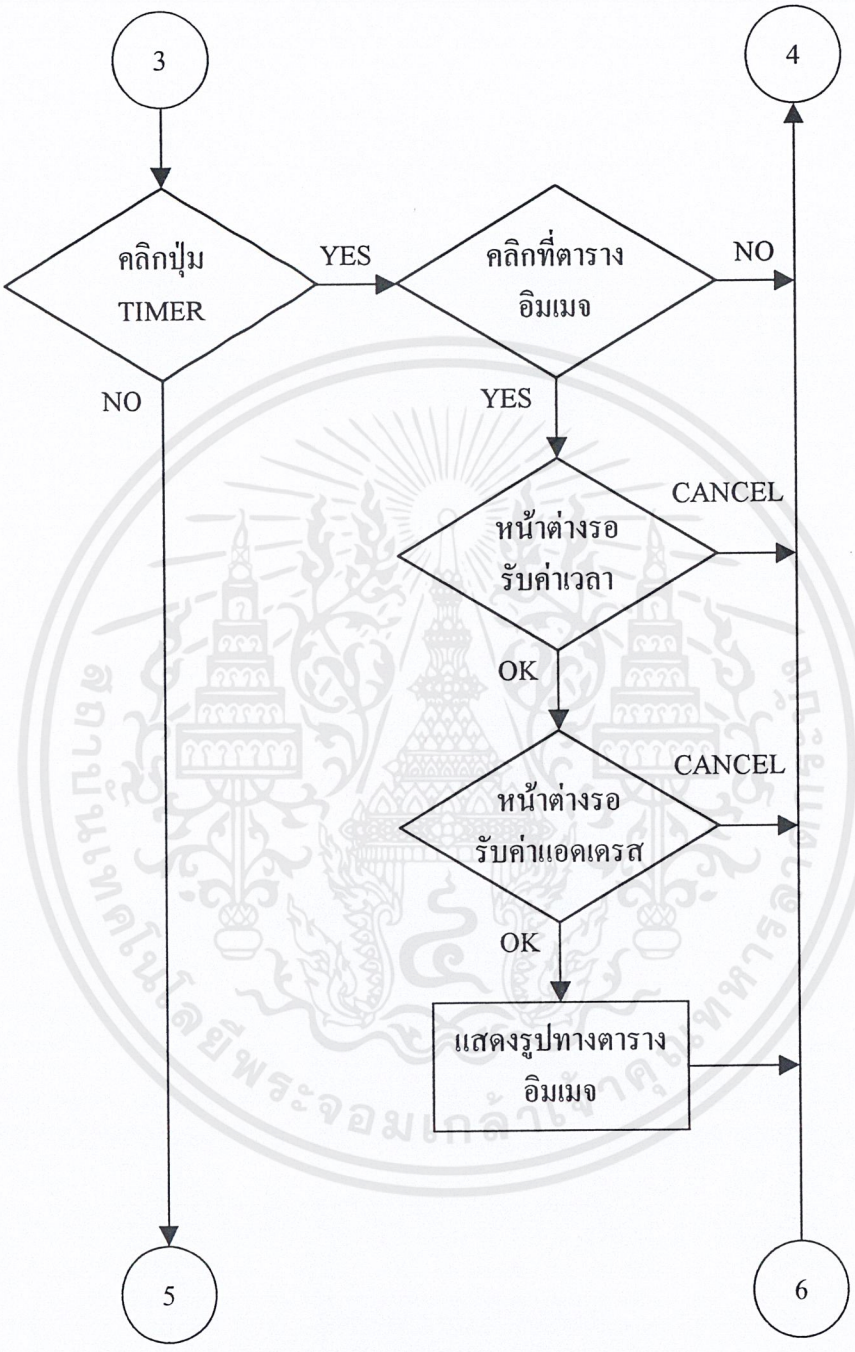
รูปที่ 1 กราฟส่วนโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



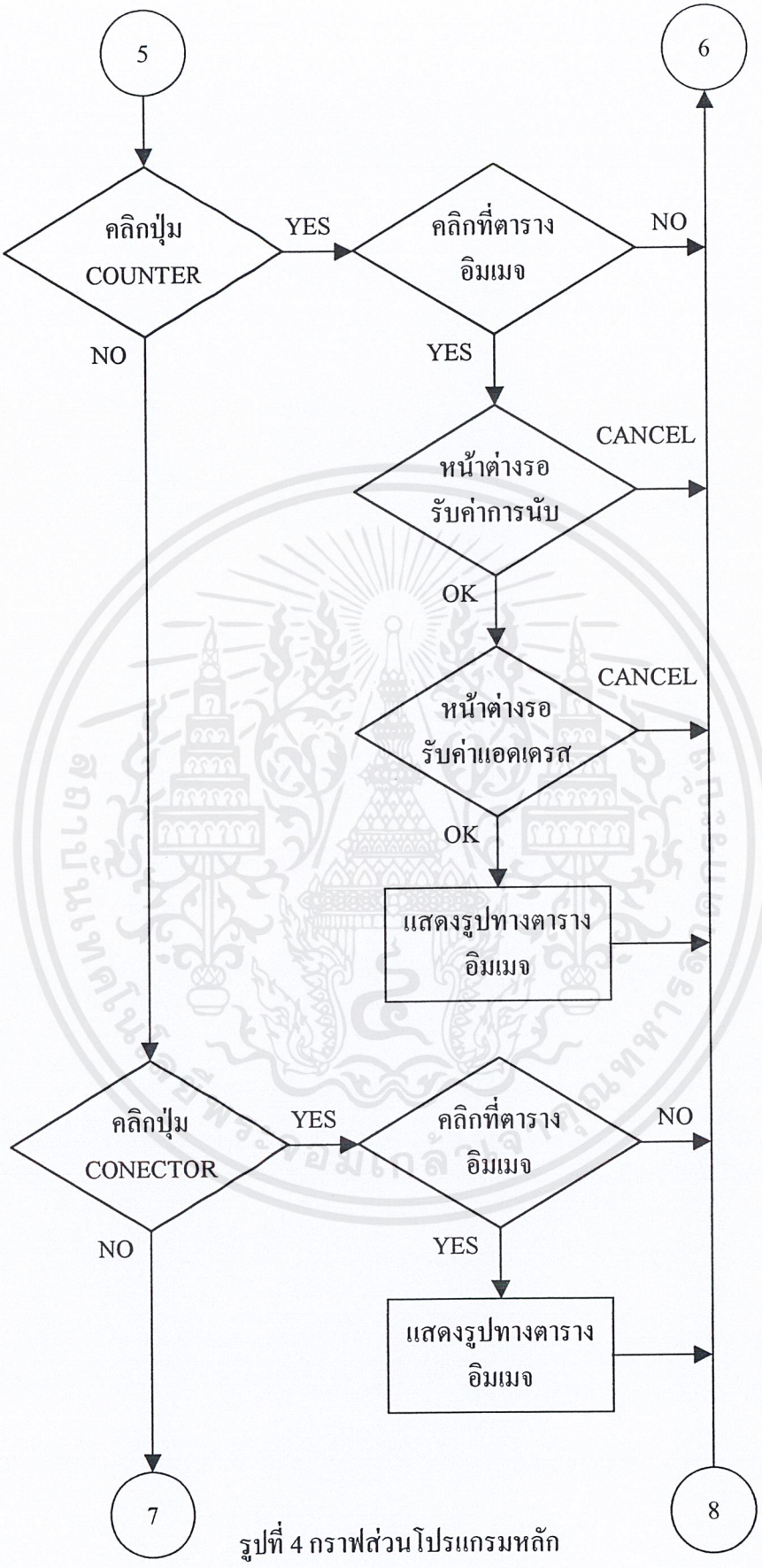
รูปที่ 2 กราฟส่วนโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



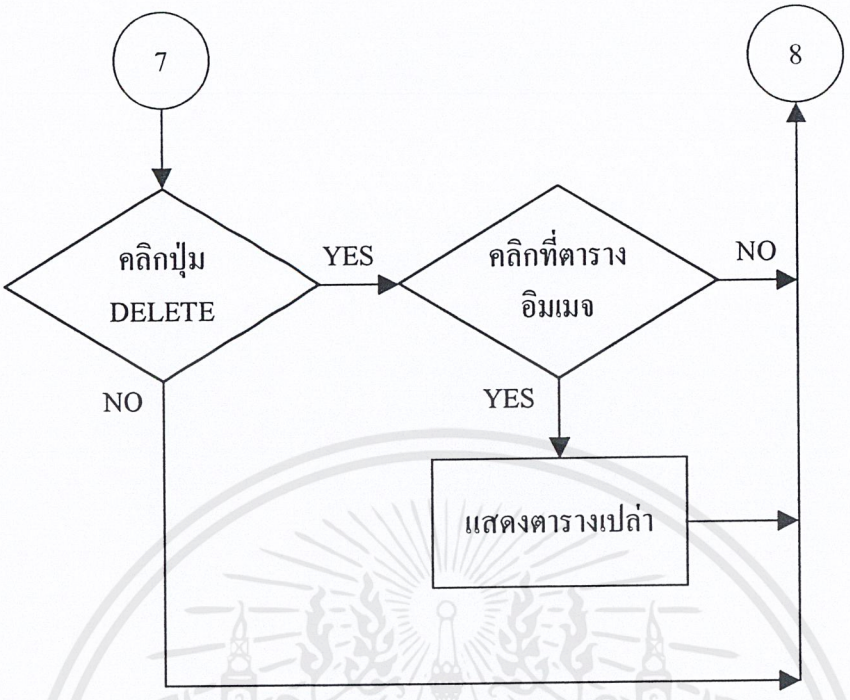
รูปที่ 3 กราฟส่วนโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



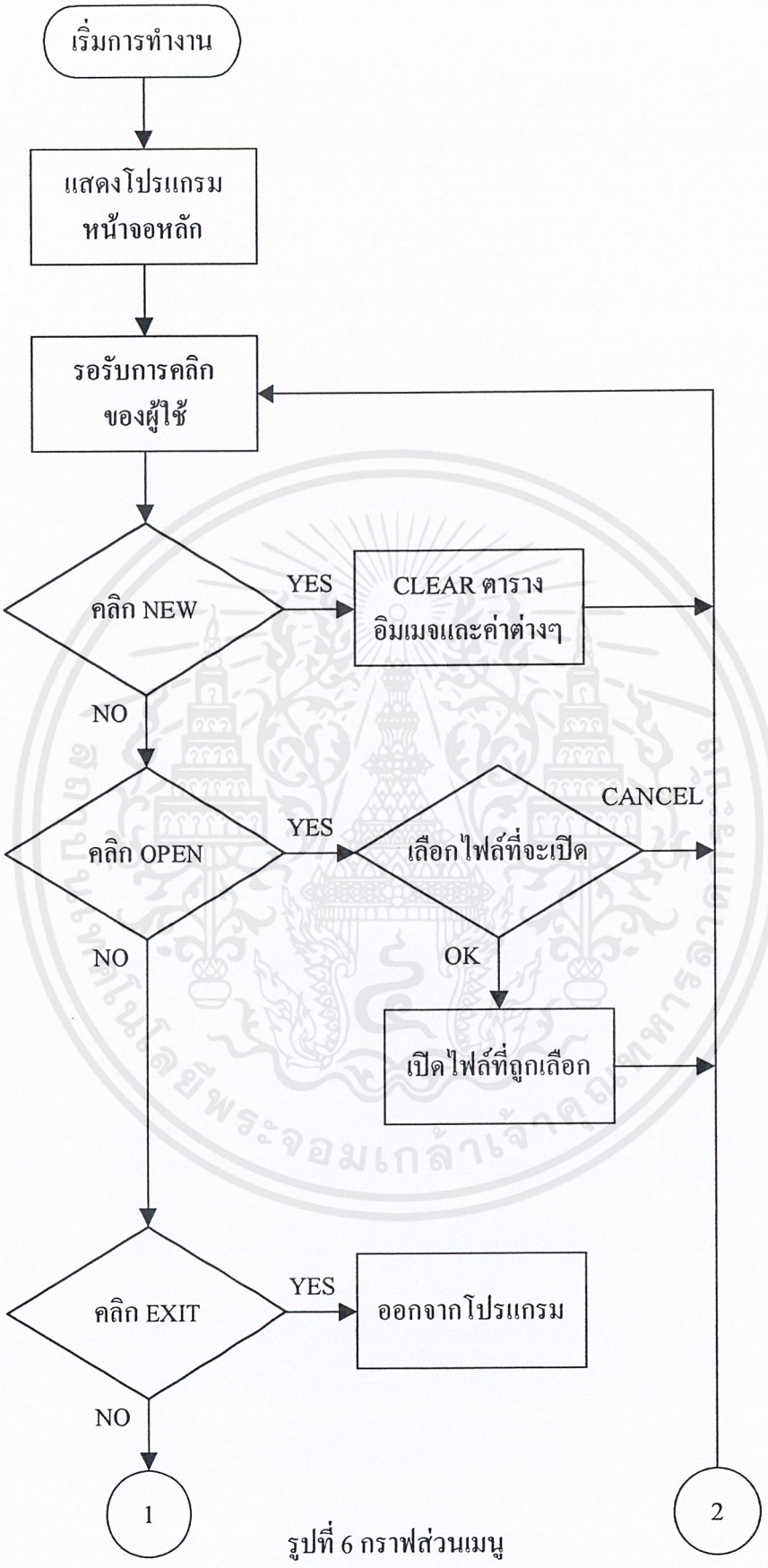
รูปที่ 4 กราฟส่วนโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



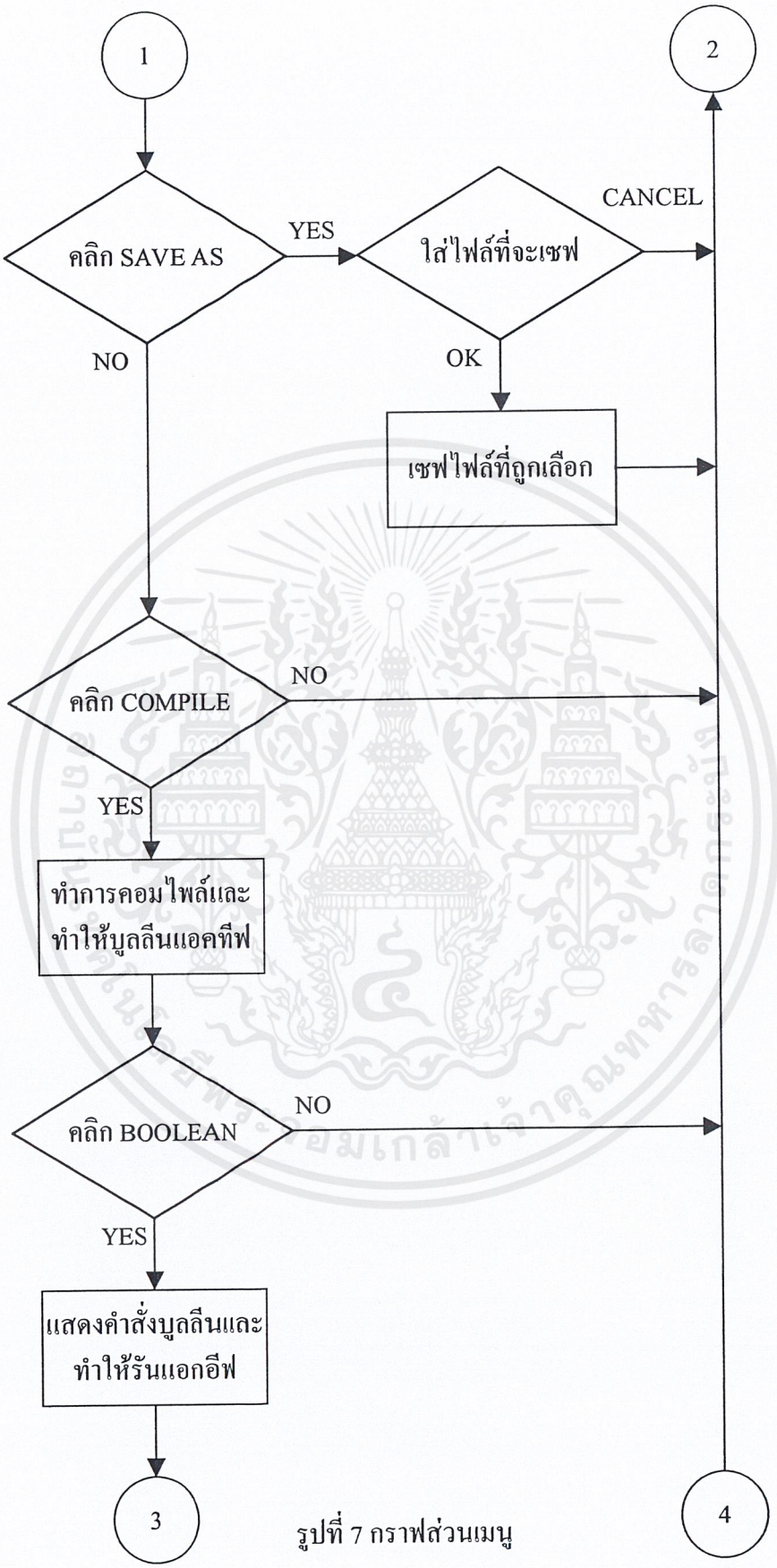
รูปที่ 5 กราฟส่วนโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



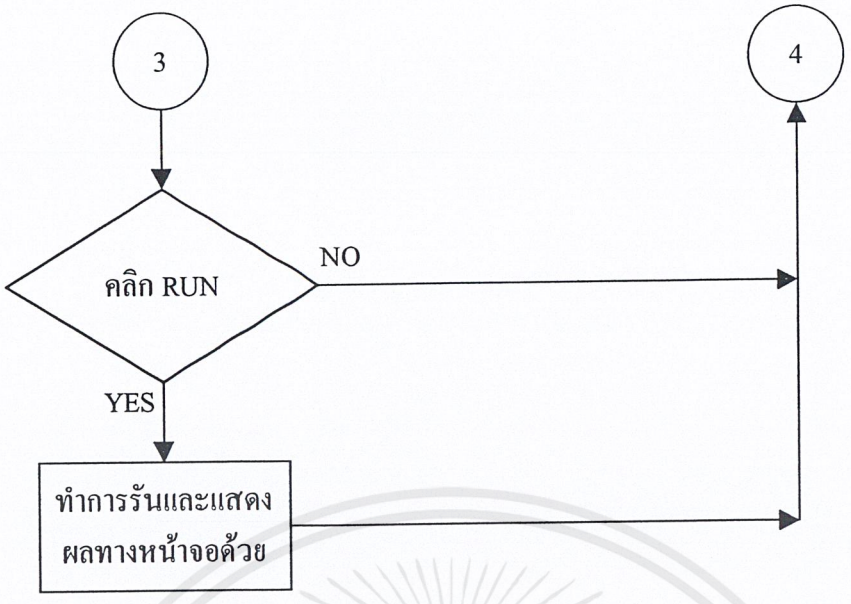
รูปที่ 6 กราฟส่วนเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



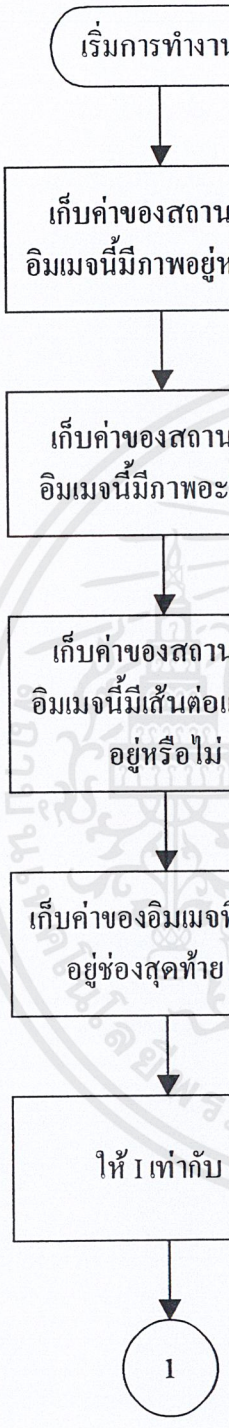
รูปที่ 7 กราฟส่วนเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



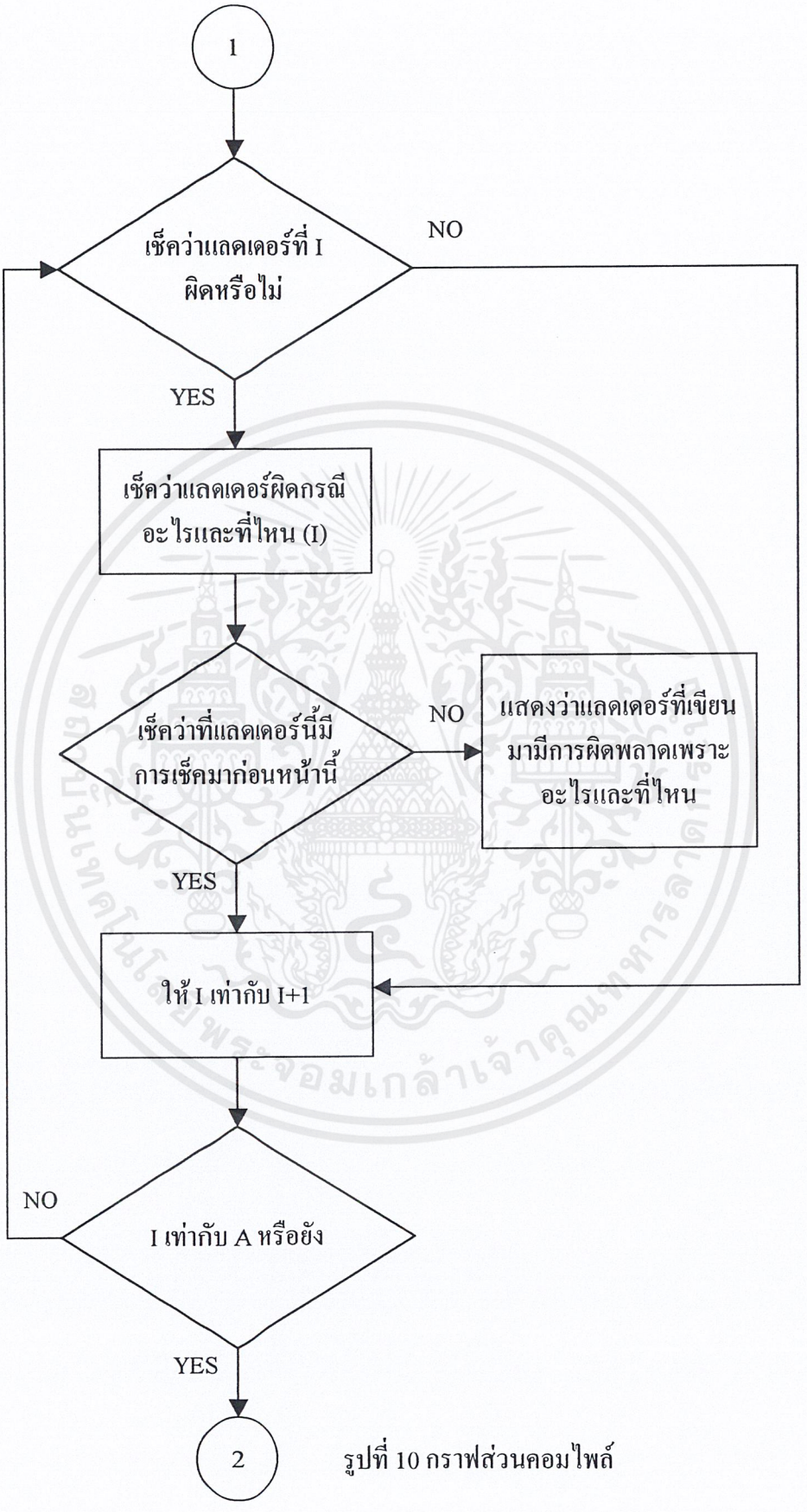
รูปที่ 8 กราฟส่วนเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



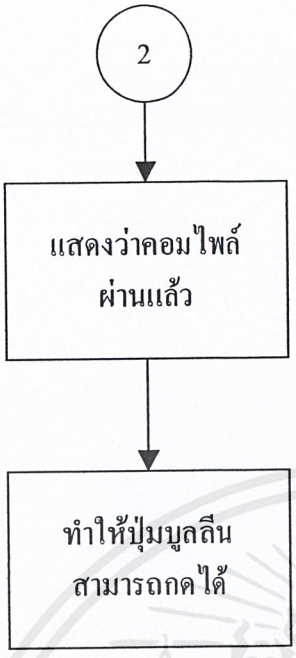
รูปที่ 9 กราฟส่วนคอมไพล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



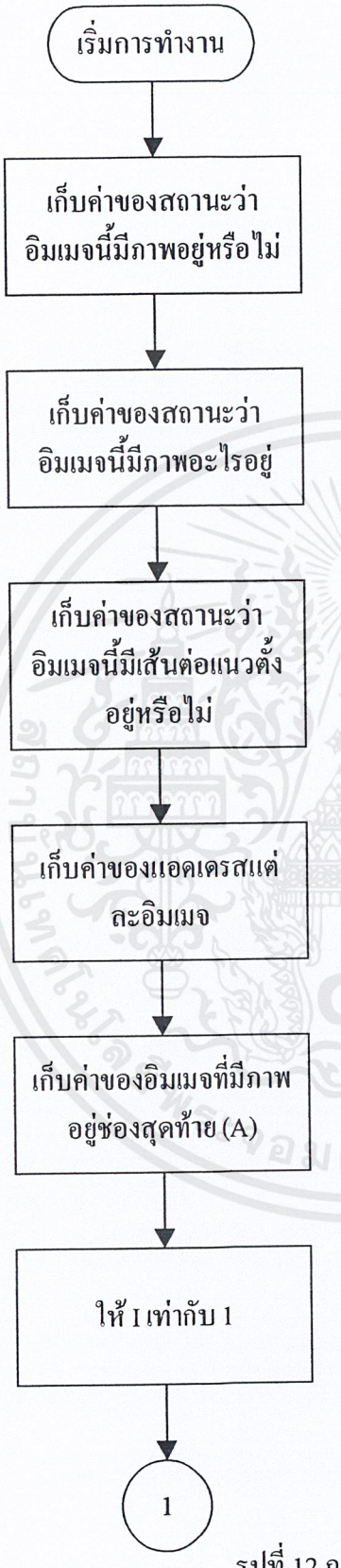
รูปที่ 10 กราฟส่วนคอมไพล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



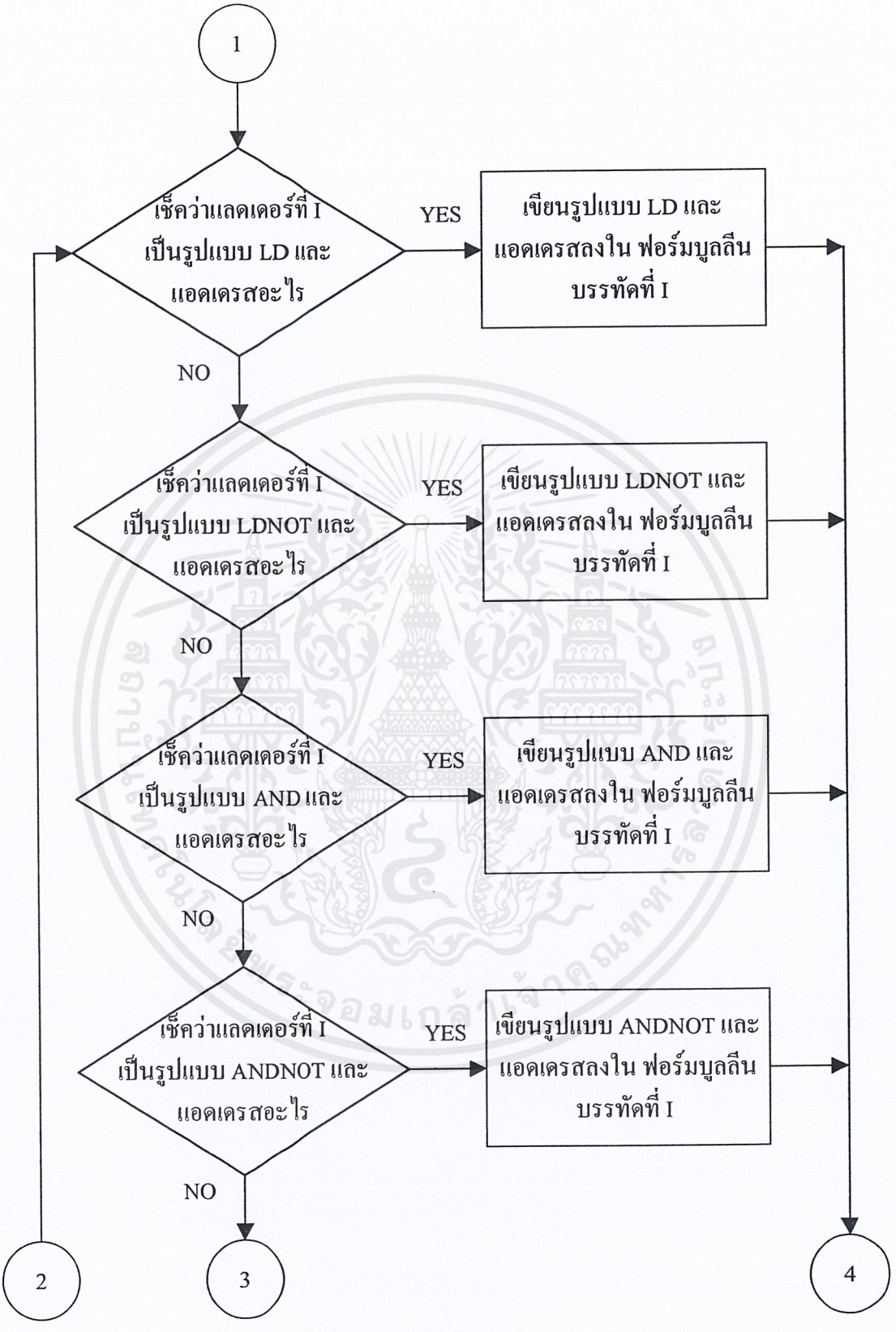
รูปที่ 11 กราฟส่วนคอมไพล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



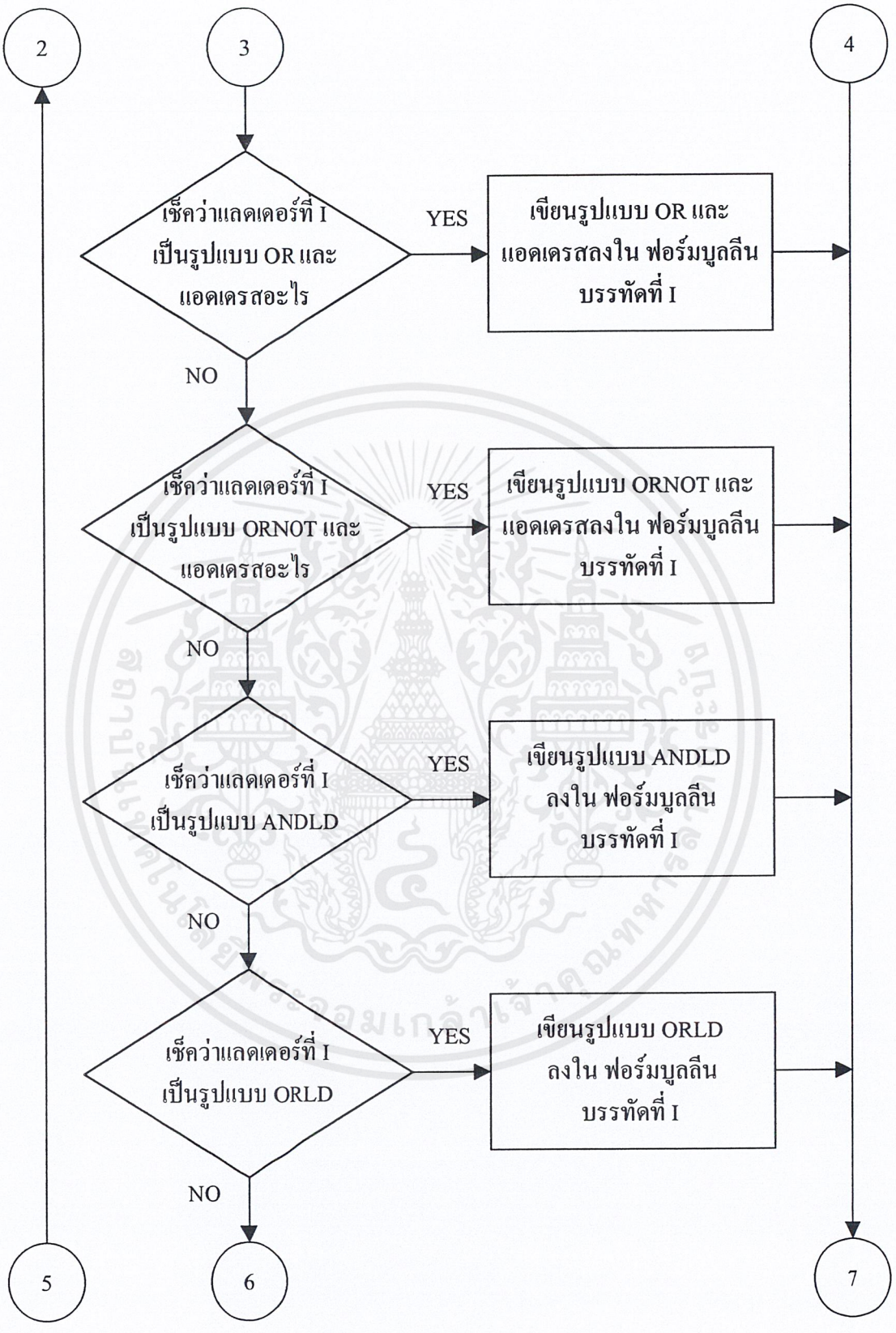
รูปที่ 12 กราฟส่วนบูลลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



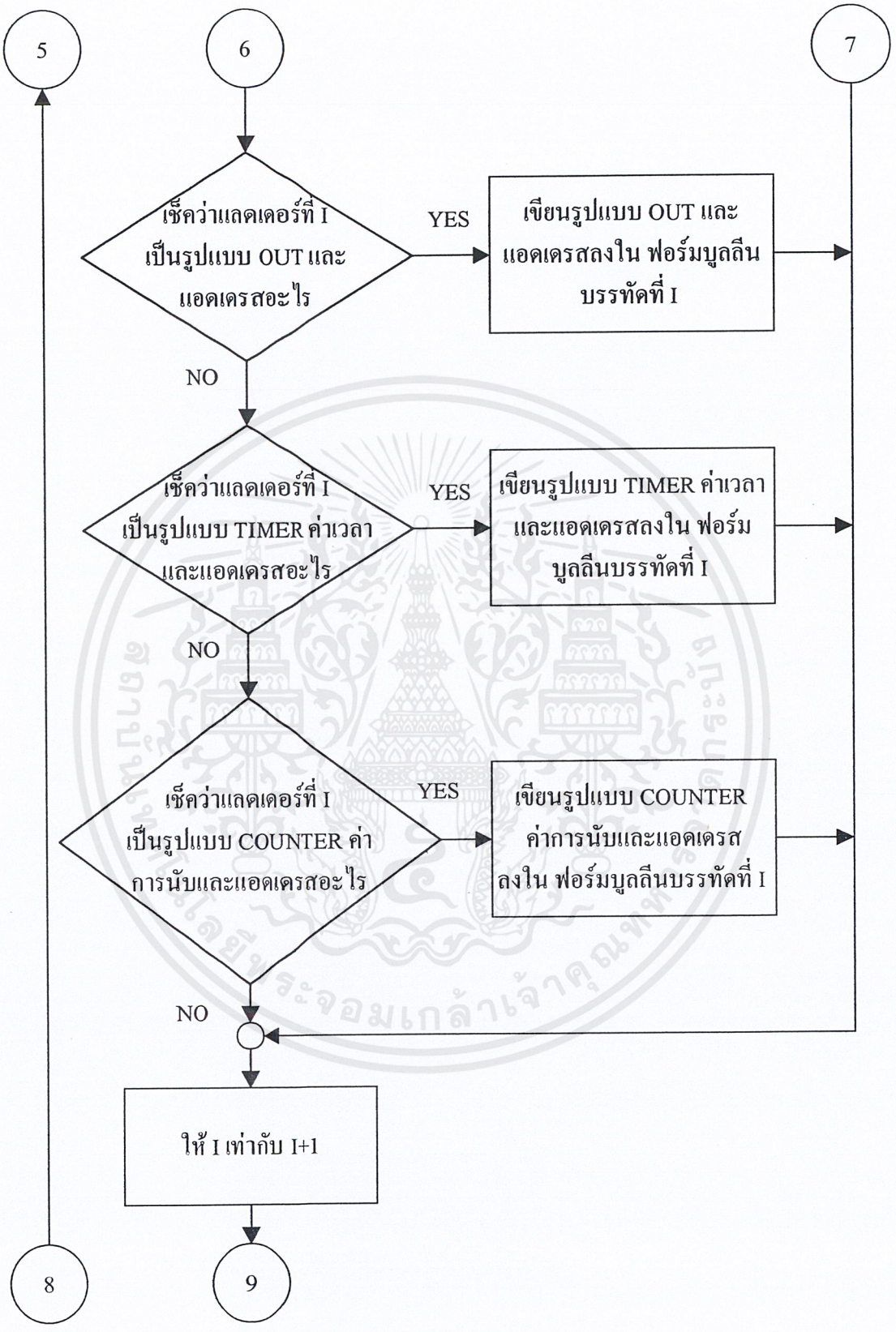
รูปที่ 13 กราฟส่วนบูลลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



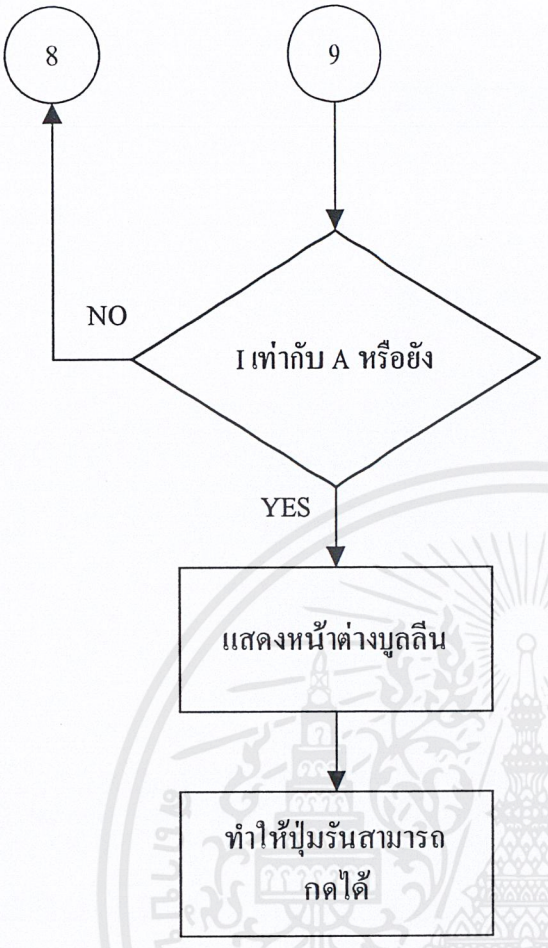
รูปที่ 14 กราฟส่วนบูลลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



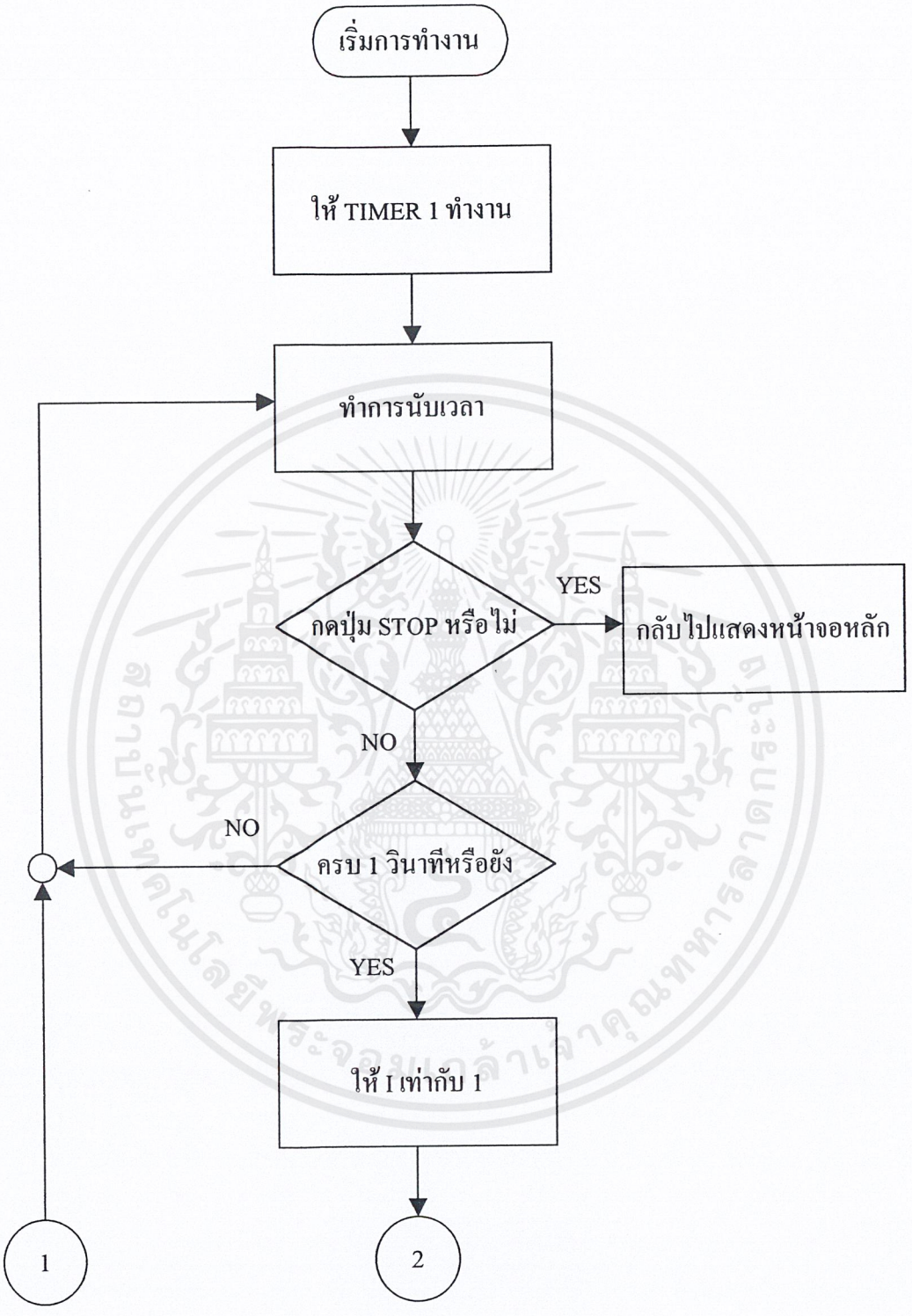
รูปที่ 15 กราฟส่วนบูลลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



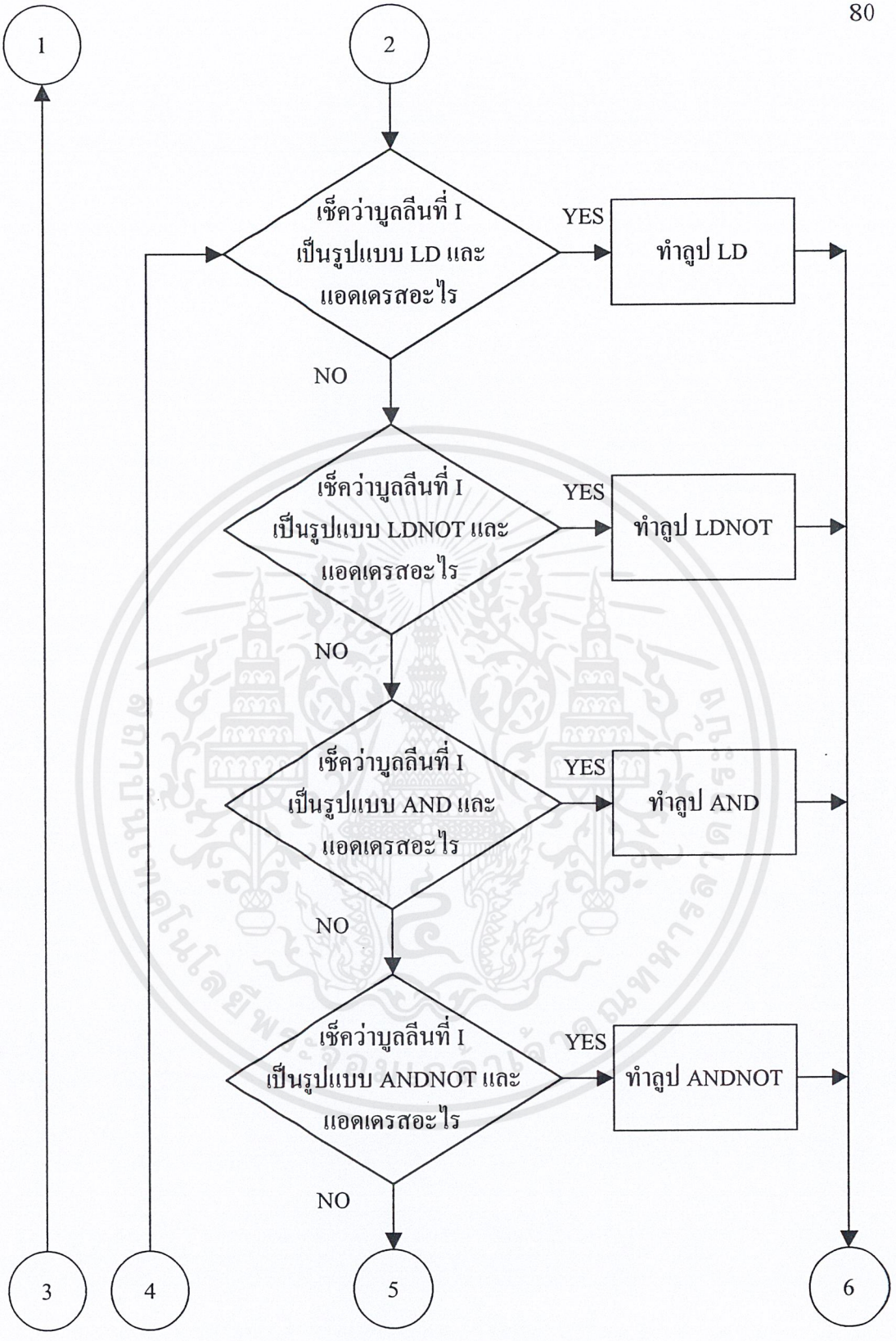
รูปที่ 16 กราฟส่วนบูลลีน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



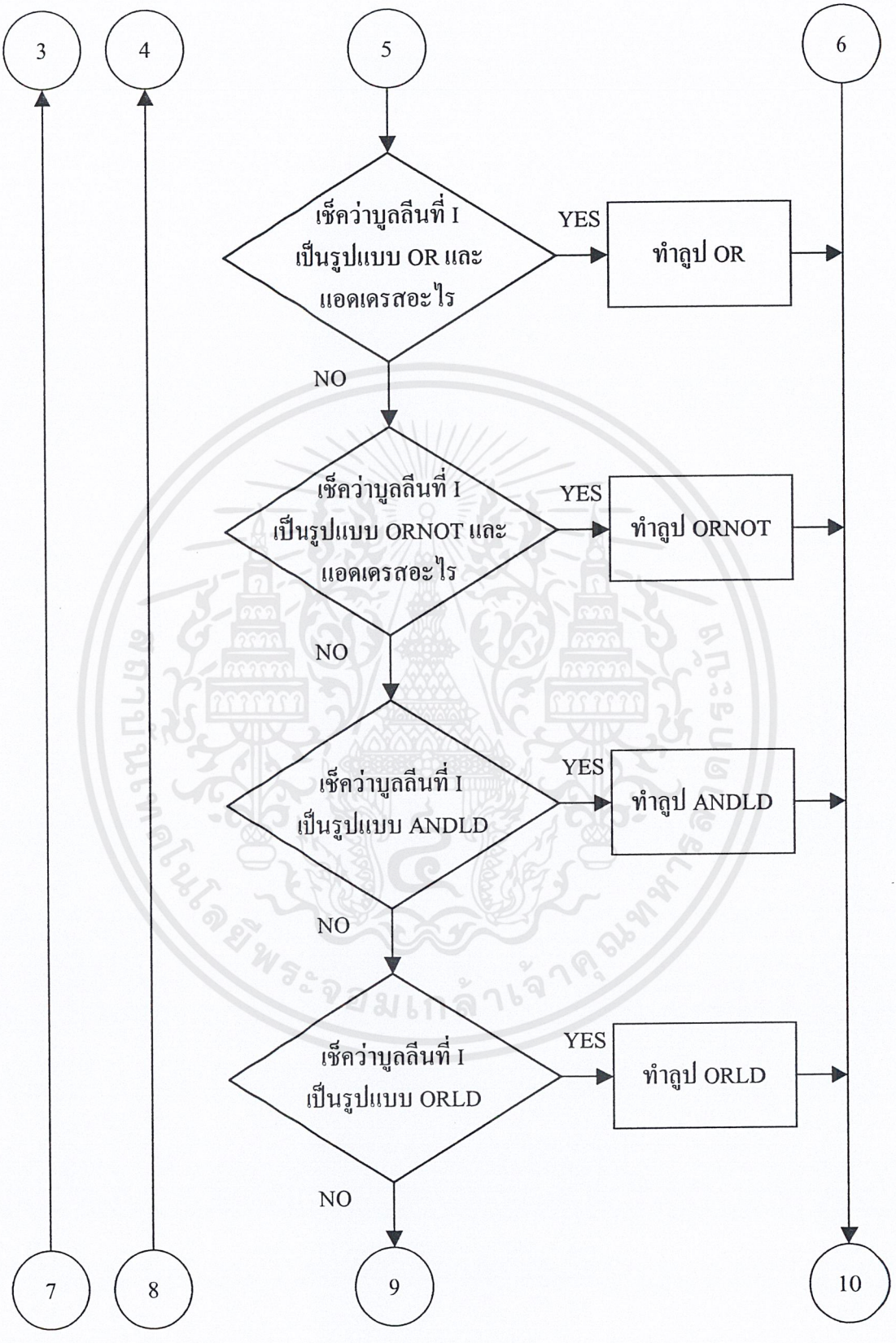
รูปที่ 17 กราฟส่วนรัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



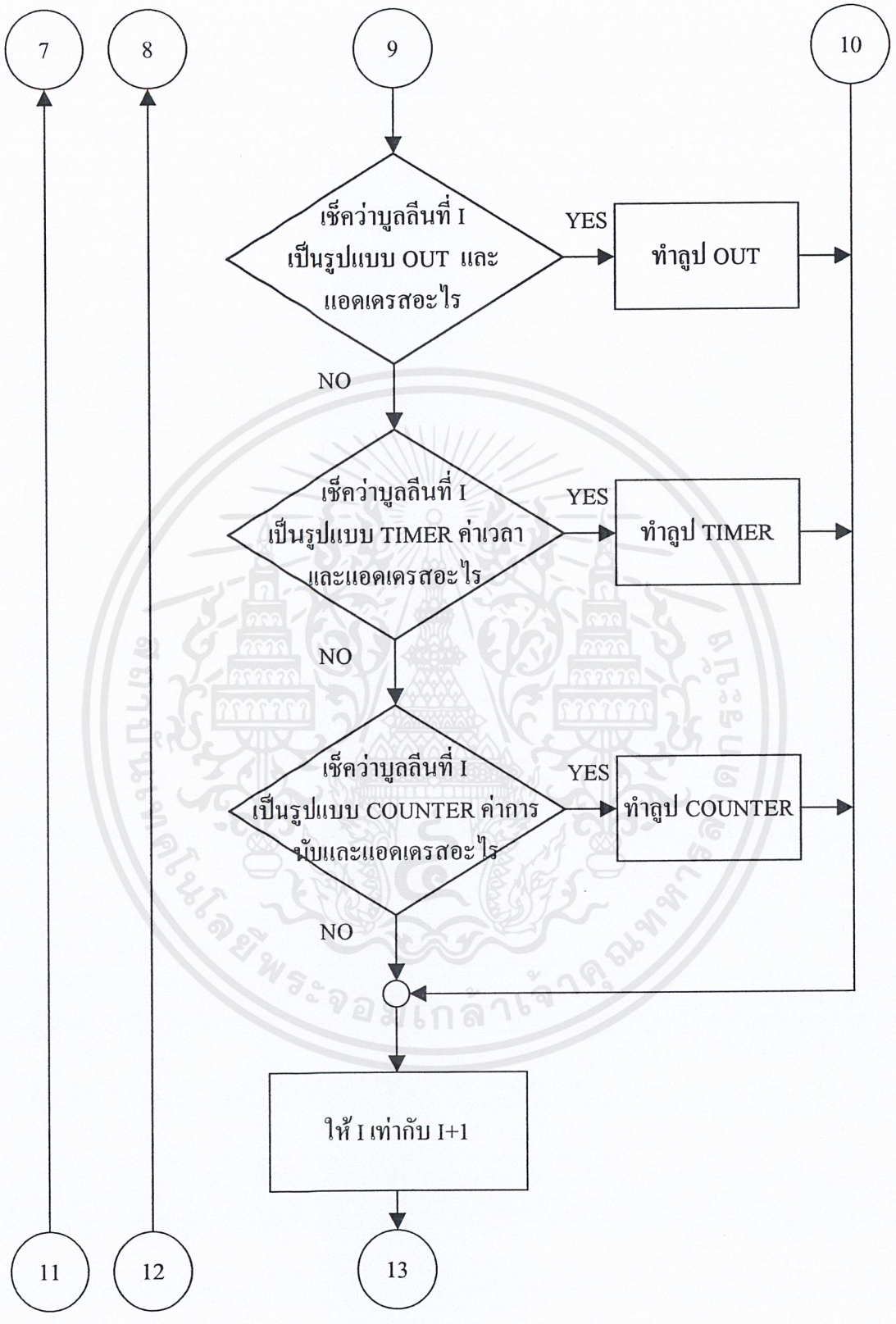
รูปที่ 18 กราฟส่วนรัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



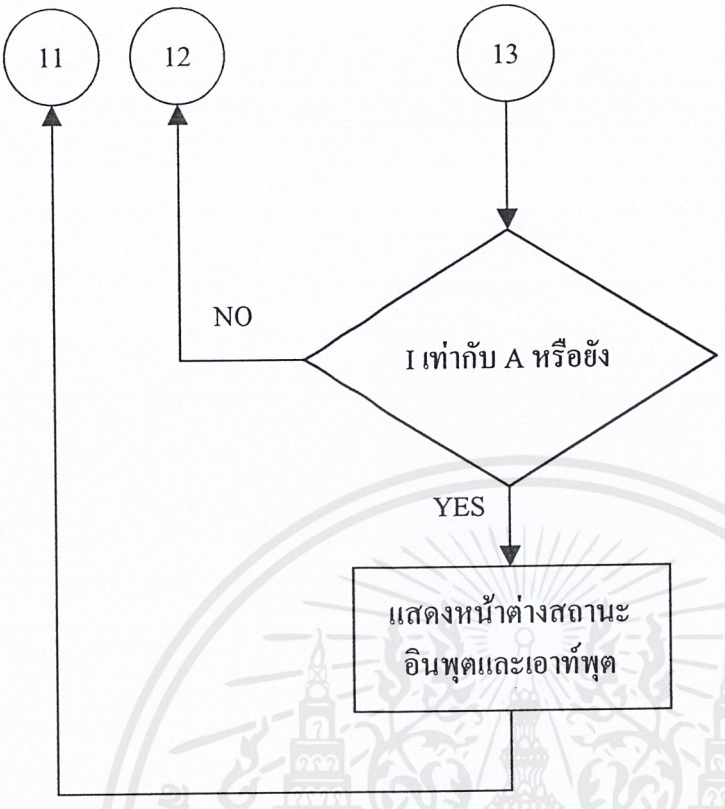
รูปที่ 19 กราฟส่วนรัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 20 กราฟส่วนรัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 21 กราฟส่วนรัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

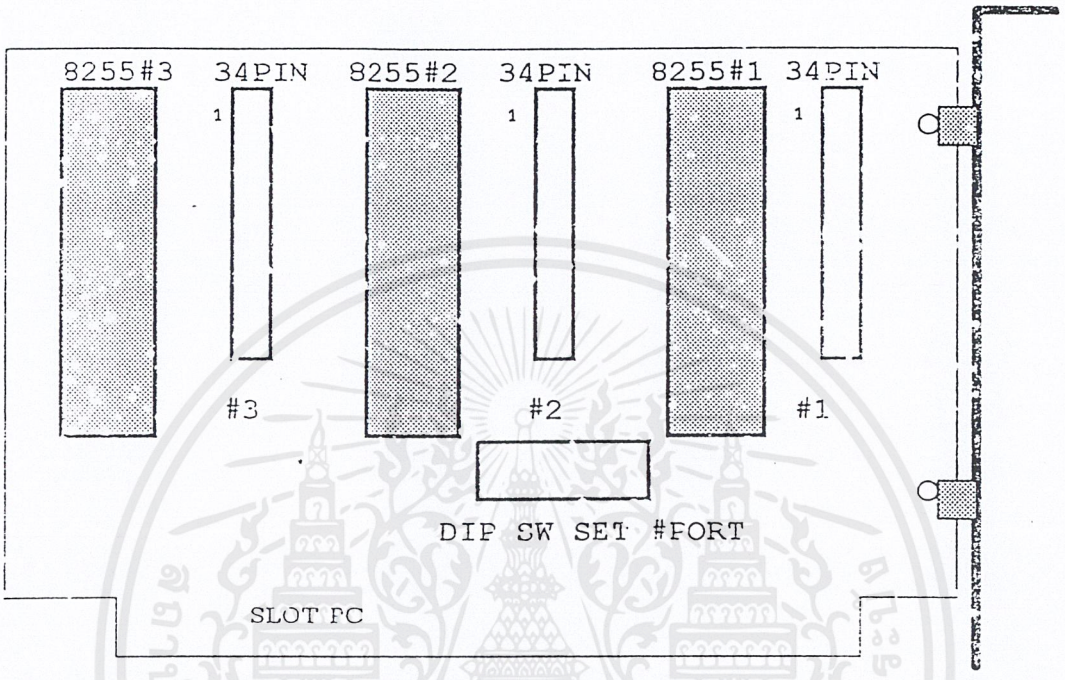
คู่มือการใช้งานบอร์ด ET-PC 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ET-PC CARD SERIAL

ET-PC 8255

ลักษณะของ ET-PC 8255



ET-PC 8255 จะเป็น CARD ต่อขยายระบบเครื่อง PC ให้มีส่วนของ INPUT , OUTPUT PORT ใช้งานมากขึ้น โดยจะมี PORT ใช้งานเป็น INPUT หรือ OUTPUT จำนวน 3 PORT หรือ 72 BIT I/O (TTL 5V)

การติดตั้ง ET-PC 8255 กับเครื่อง PC

1. ปิด SW. POWER ของเครื่องคอมพิวเตอร์ PC นั้นก่อน
2. เปิดฝาเครื่องคอมพิวเตอร์
3. SET DIP SW. ตำแหน่ง PORT ของ CARD ET-PC 8255 ไม่ให้ตรงกับตำแหน่ง PORT ของ CARD อื่นๆ (ดูได้ในเรื่อง SET DIP SW. ตำแหน่ง PORT)
4. นำ CARD ET-PC 8255 ใส่เข้าไปยังเครื่องคอมพิวเตอร์ PC ทาง SLOT PC 62 PIN ดูการใส่ CARD ให้เรียบร้อยสนิทดีด้วยกับ SLOT PC
5. ถ้ามีการต่อสายแพร์จาก 34 PIN ไปใช้งานก็ให้ต่อก่อนให้เรียบร้อยแล้วเสร็จก่อน
6. เปิด SW. POWER เครื่องคอมพิวเตอร์ PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

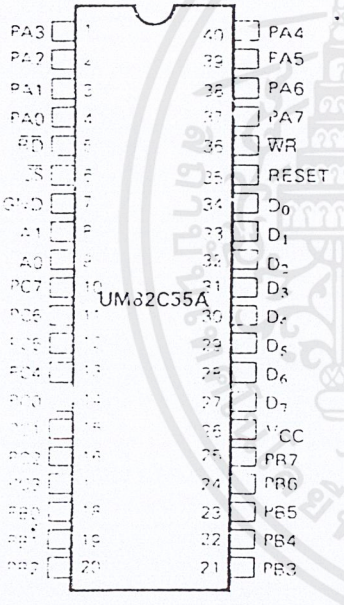
การทำงานของ ET-PC 8255

CARD ET-PC 8255 จะประกอบไปด้วย 2 ส่วนใหญ่ๆก็คือ ส่วน IC 8255 ซึ่งเป็น IC ที่ทำหน้าที่เป็น INPUT , OUTPUT PORT และส่วนของวงจร IC DECODE (เลือกตำแหน่งของ PORT 8255) คือ IC 74LS688 , 74LS139 และ DIP SW.

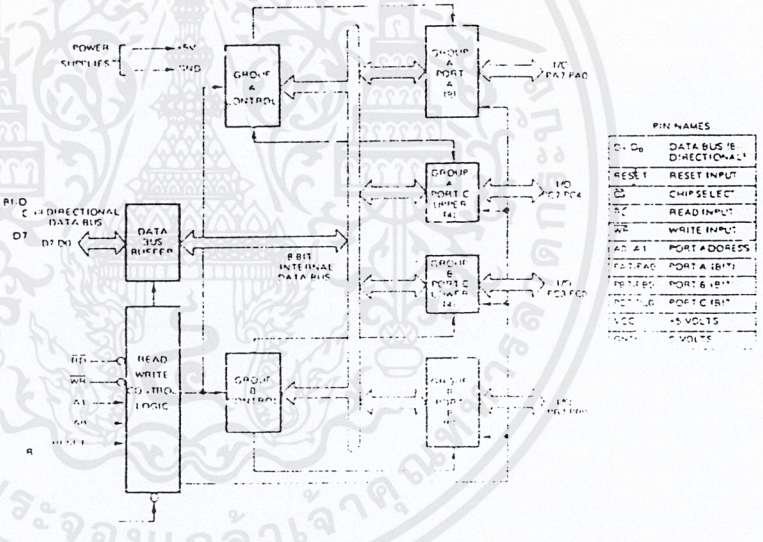
การใช้งาน IC 8255

IC 8255 นี้จะเป็น IC ซึ่งประกอบด้วย PORT ใช้งาน 3 PORT และอีก 1 PORT ควบคุมก่อนที่เราจะใช้งาน 8255 เราจะต้องส่งข้อมูลไปให้ยัง PORT ควบคุมก่อนว่าจะให้ PORT ทั้ง 3 PORT ของ 8255 ที่เหลือนั้นทำหน้าที่อะไร เป็น INPUT หรือ OUTPUT PORT เราจะต้องเป็นผู้กำหนด CONTROL CODE PORT ควบคุมดังรูป :-

Pin Configuration



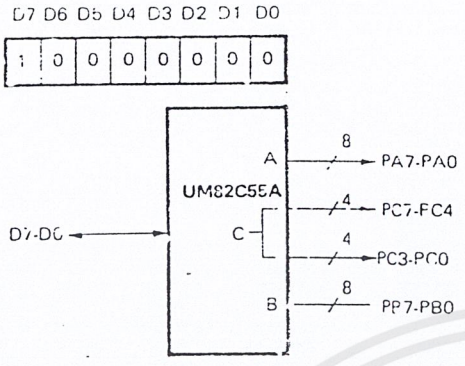
Block Diagram



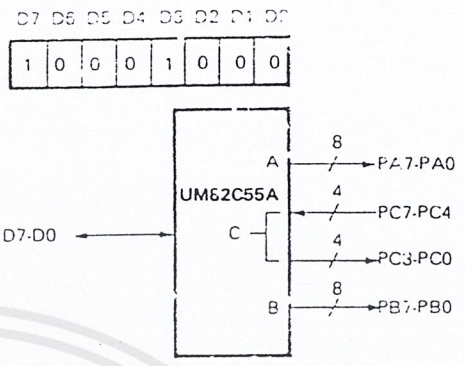
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mode 0 Configurations

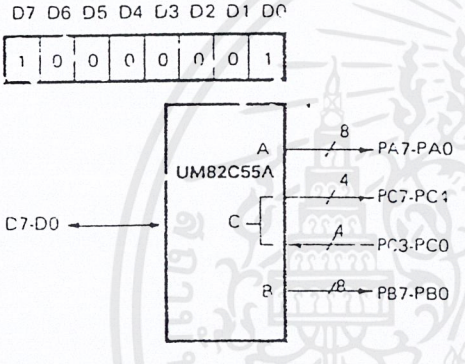
CONTROL WORD #0



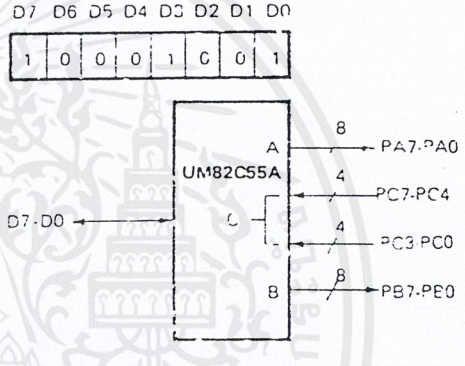
CONTROL WORD #4



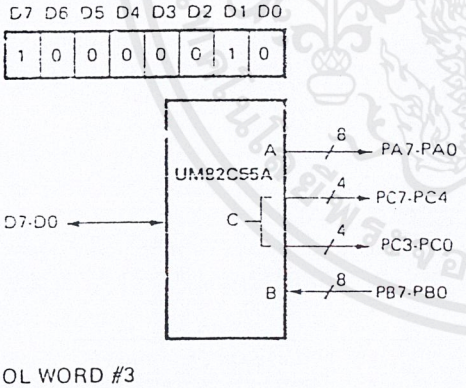
CONTROL WORD #1



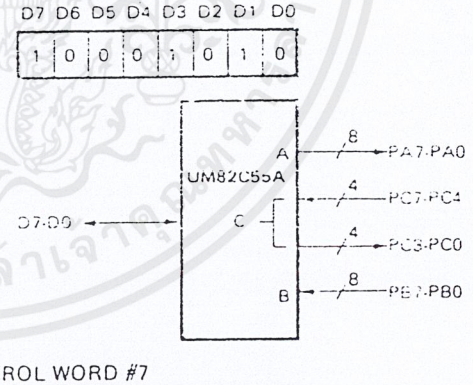
CONTROL WORD #5



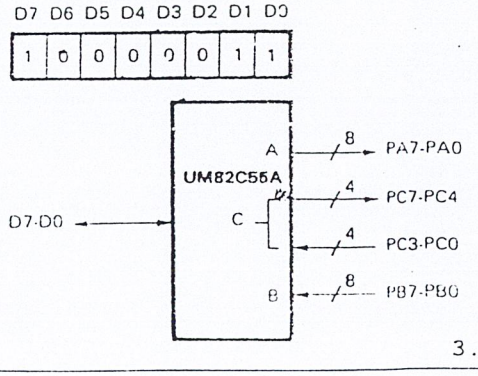
CONTROL WORD #2



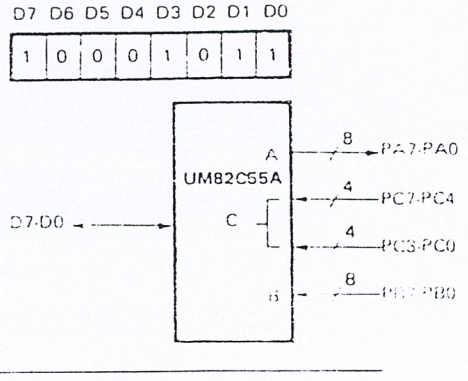
CONTROL WORD #6



CONTROL WORD #3

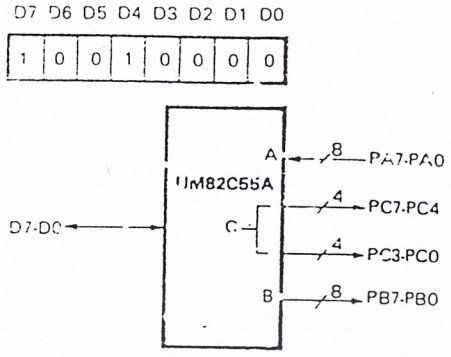


CONTROL WORD #7

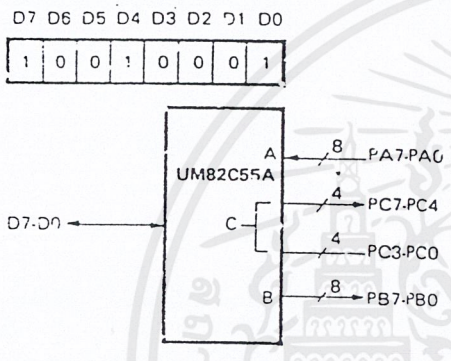


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

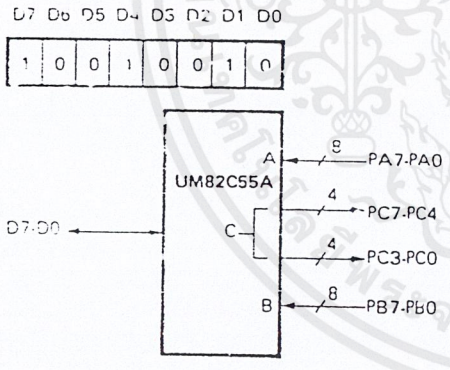
CONTROL WORD #8



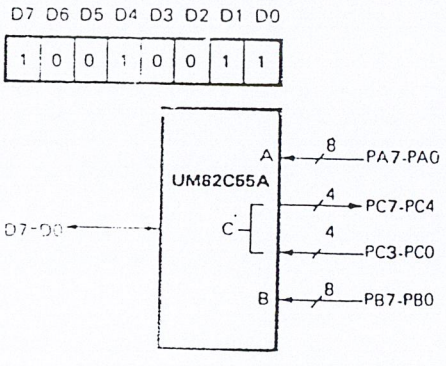
CONTROL WORD #9



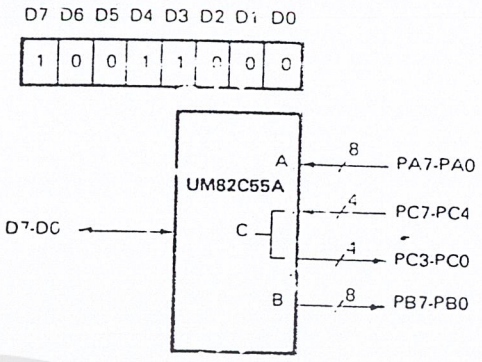
CONTROL WORD #10



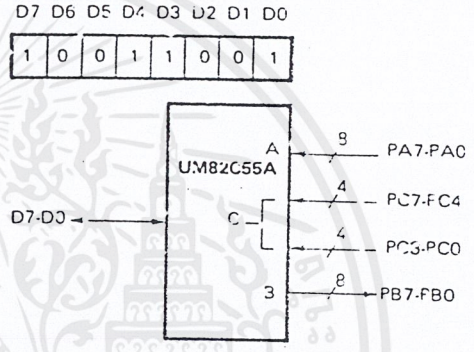
CONTROL WORD #11



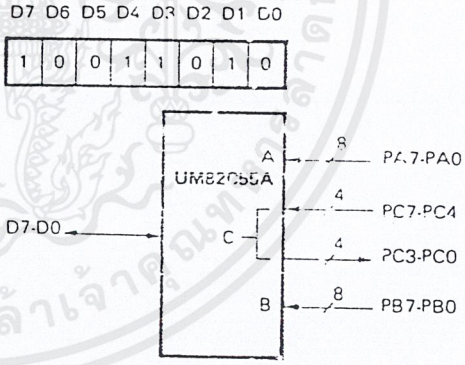
CONTROL WORD #12



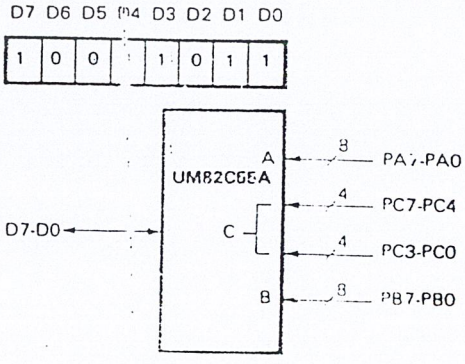
CONTROL WORD #13



CONTROL WORD #14



CONTROL WORD #15



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PC HARDWARE I/O MAP

8088 Class Systems

Address	Function
ØØØ–ØØF	DMA Controller (8237A)
Ø2Ø–Ø21	Interrupt controller (8259A)
Ø4Ø–Ø43	Timer (8253)
Ø6Ø–Ø63	PPI (8255A)
Ø8Ø–Ø83	DMA page register (74LS612)
ØAØ–ØAF	NMI – Non Maskable Interrupt
2ØØ–2ØF	Game Port Joystick controller
21Ø–217	Expansion Unit
2E8–2EF	COM4: Serial Port
2F8–2FF	COM2: Serial Port
3ØØ–31F	Prototype Card
32Ø–32F	Hard Disk
378–37F	Parallel Printer Port 1
38Ø–38F	SDLC
3BØ–3BF	MDA – Monochrome Adapter and printer
3DØ–3D7	CGA – Color Graphics Adapter
3E8–3EF	COM3: Serial Port
3FØ–3F7	Floppy Diskette Controller
3F8–3FF	COM1: Serial Port

80286 /386/486 Class Systems

Address	Function
ØØØ–Ø1F	DMA Controller #1 (8237A–5)
Ø2Ø–Ø3F	Interrupt controller #1 (8259A)
Ø4Ø–Ø5F	Timer (8254)
Ø6Ø–Ø6F	Keyboard (8Ø42)
Ø7Ø–Ø7F	NMI – Non Maskable Interrupt & CMOS RAM
Ø8Ø–Ø9F	DMA page register (74LS612)
ØAØ–ØBF	Interrupt controller #2 (8259A)
ØCØ–ØDF	DMA Controller #2 (8237A)
ØFØ–ØFF	8Ø287 Math Coprocessor
1FØ–1F8	Hard Disk
2ØØ–2ØF	Game Port Joystick controller
258–25F	Intel Above Board
278–27F	Parallel Printer Port 2
2E8–2EF	COM4: Serial Port
2F8–2FF	COM2: Serial Port
3ØØ–31F	Prototype Card
378–37F	Parallel Printer Port 1
38Ø–38F	SDLC or Bisynchronous Comm Port 2
3AØ–3AF	Bisynchronous Comm Port 1
3BØ–3BF	MDA – Monochrome Adapter
3BC–3BE	Parallel Printer on Monochrome Adapter
3CØ–3CF	EGA – Reserved
3DØ–3D7	CGA – Color Graphics Adapter
3E8–3EF	COM3: Serial Port
3FØ–3F7	Floppy Diskette Controller
3F8–3FF	COM1: Serial Port

PC Hardware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

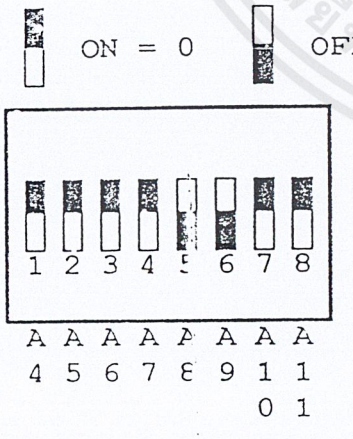
การ DECODE PORT

DECODE PORT 8255 บน CARD เราจะใช้ IC TTL 74LS688 , IC TTL 74LS139 และ DIP SW. 8 PIN เป็นวงจร DECODE เพื่อให้สามารถปรับ SET DIP SW. ตั้งตำแหน่งเบอร์ PORT ของ CARD ได้ โดยในการปรับ DIP SW. นั้นจะต้องไปตรงกับตำแหน่ง PORT ของเครื่องคอมพิวเตอร์ PC ด้วย ดังรูป :-

โดย CARD ET-PC 8255 จะใช้ตำแหน่ง PORT 12 PORT ต่อ CARD DECODE PORT

XX0H	PORT A (#1)
XX1H	PORT B (#1)
XX2H	PORT C (#1)
XX3H	CONTROL PORT (#1)
XX4H	PORT A (#2)
XX5H	PORT B (#2)
XX6H	PORT C (#2)
XX7H	CONTROL PORT (#2)
XX8H	PORT A (#3)
XX9H	PORT B (#3)
XXAH	PORT C (#3)
XXBH	CONTROL PORT (#3)

เราตั้งเบอร์ DECODE PORT ได้โดยการปรับ DIP SW. ซึ่งก็มีค่าเท่ากับค่า ADDRESS นั้นๆ เช่น เราตั้งตำแหน่ง 300H จะ SET DIP SW. ดังนี้ :-

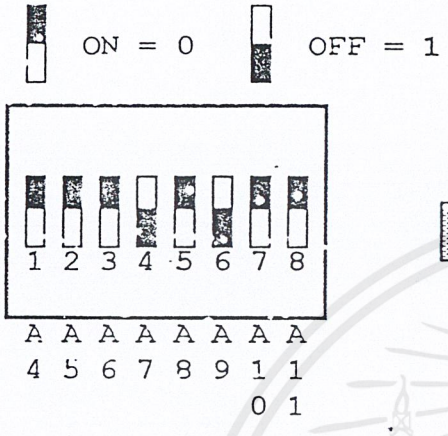


PORT 300H

A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
1	1	9	8	7	6	5	4	3	2	1	0				
1	0														
0	0	1	1	0	0	0	0	X	X	X	X				
								3			0			0	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

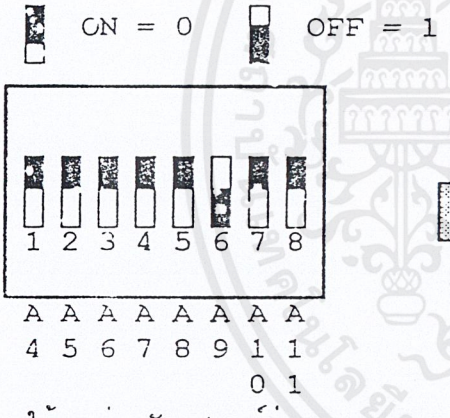
ถ้าต้องการตั้งตำแหน่ง 280H จะ SET DIP SW. ดังนี้ :-



PORT 280H

A	A	A	A	A	A	A	A	A	A	A	A
1	1	9	8	7	6	5	4	3	2	1	0
1	C										
0	0	1	0	1	0	0	0	X	X	X	X

2 8 0



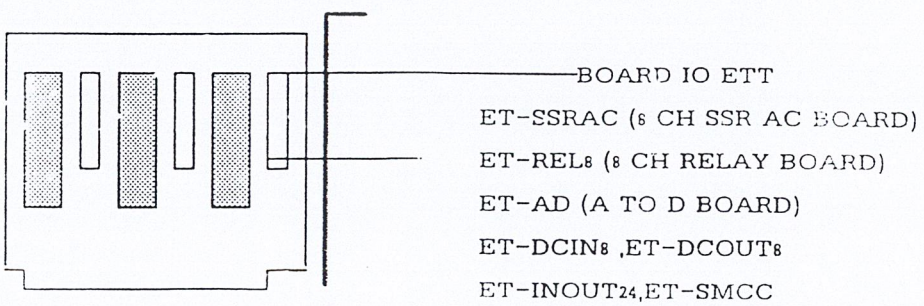
PORT 200H

A	A	A	A	A	A	A	A	A	A	A	A
1	1	9	8	7	6	5	4	3	2	1	0
1	0										
0	0	1	0	0	0	0	0	X	X	X	X

2 0 0

การใช้งานร่วมกับอุปกรณ์ต่างๆของ ETT

เราสามารถต่อ CONNECTOR 34 PIN ของ CARD ET-PC 8255 ไปยัง BOARD ต่างๆ ของอีทีทีได้ เช่น ต่อใช้งานกับ ET-SSRAC ซึ่งจะเป็น BOARD ควบคุมไฟ AC 220V ได้ 8 OUTPUT ดังรูป หรือจะต่อรับกับ ET-AD ซึ่งเป็น BOARD วัด VOLT , OHM , หรืออุณหภูมิได้ด้วยดังรูป :-



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPECIFICATIONS

LOGIC INPUT AND OUTPUT	: MIN	MAX
INPUT LOGIT LOW	: -0.5	0.8 VOLTS
INPUT LOGIT HIGH	: 2.0	5 VOLTS

OUTPUT LOW VOLTAGE PORTS : - 0.45 VOLTS
(I-SINK = 1.7 MA)

OUTPUT HIGH VOLTAGE PORTS : 2.4 - VOLTS
(I-SOURCE = 200 UA)

POWER CONSUMPTION	: 300 MA +5V
SIZE	: HALF SLOT (13 X 11 CM)
PORT	: 3 (8255 I/O PORT 3 X 8 BIT)
CONNECTOR	: 3 (34 PIN HEADER-STRIP ETT IO BUS)
DECODE PORT	: 8 POSITION DIP SW.

คำแนะนำในการประกอบ ET-PC 8255

1. การใส่อุปกรณ์ตามลำดับ คือ ตัวต้านทาน , ตัวเก็บประจุ , SOCKET
2. ตัวเก็บประจุให้ใส่ให้ถูกขั้วตามที่พิมพ์ด้วย
3. LED ขายาวคือ A ขาสั้นคือ K ใส่ตามรูปบน PCB
4. การบัดกรีให้กระทำเฉพาะด้านล่างเพียงด้านเดียว ด้านบนไม่ต้องบัดกรี
5. เมื่อเรียบร้อยแล้วใส่ IC ลงใน SOCKET ระวังไม่ให้ผิดตำแหน่งหรือกลับหัว
6. ประกอบแผ่นเหล็กยึด PRINT

รายการอุปกรณ์ ET-PC 8255

- IC 8255	3 ตัว	- CONNECTOR 34 PIN	3 ตัว
- 74LS139	1 ตัว	- DIP SW. 8 จุด	1 ตัว
- 74F245	1 ตัว	- R-PACK 10K 9 PIN	11 ตัว
- 74LS245	1 ตัว	- R 330	1 ตัว
- 74LS688	1 ตัว	- R 4.7K	1 ตัว
- C 0.1 UF	7 ตัว	- C 33 UF 16 V	2 ตัว
- แผ่นเหล็กยึด PRINT		- PRINT PCB	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

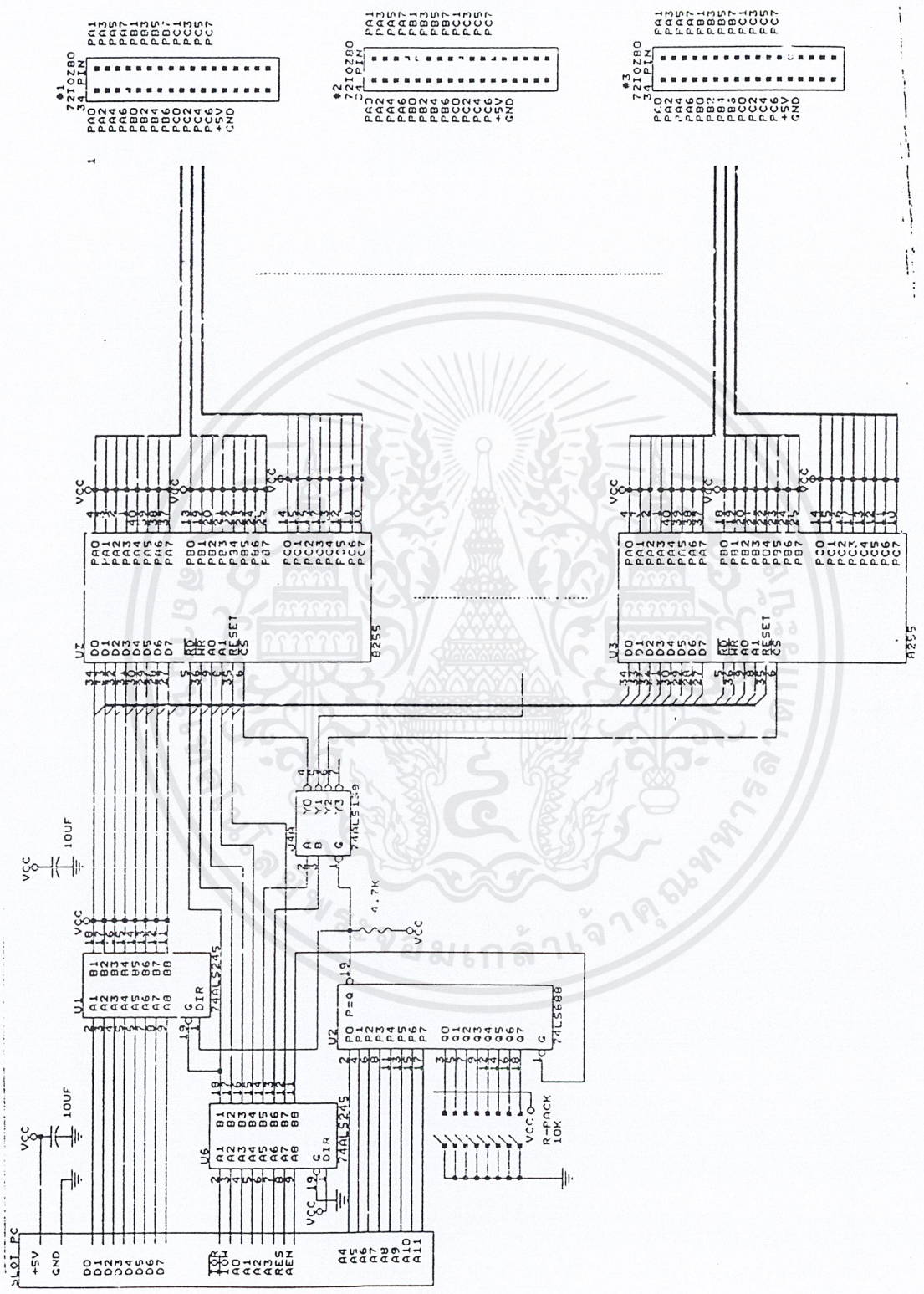
34 PIN I/O BUS

PA0	0	0	PA1
PA2	0	0	PA3
PA4	0	0	PA5
PA6	0	0	PA7
PB0	0	0	PB1
PB2	0	0	PB3
PB4	0	0	PB5
PB6	0	0	PB7
PC0	0	0	PC1
PC2	0	0	PC3
PC4	0	0	PC5
PC6	0	0	PC7
VCC	0	0	
GND	0	0	
	0	0	
	0	0	
	0	0	

B1	GND	IO CHCK	A1
B2	RST_DRV	SD7	A2
B3	+5VDC	SD6	A3
B4	IRQ9	SD5	A4
B5	-5VDC	SD4	A5
B6	DRQ2	SD3	A6
B7	-12VDC	SD2	A7
B8	0WS	SD1	A8
B9	+12VDC	SD0	A9
B10	GND	IO CHRDY	A10
B11	SMEMW	AEN	A11
B12	SMEMR	SA19	A12
B13	IOW	SA18	A13
B14	IOR	SA17	A14
B15	DACK3	SA16	A15
B16	DRQ3	SA15	A16
B17	DACK1	SA14	A17
B18	DRQ1	SA13	A18
B19	REFRESH	SA12	A19
B20	CLK	SA11	A20
B21	IRQ7	SA10	A21
B22	IRQ6	SA9	A22
B23	IRQ5	SA8	A23
B24	IRQ4	SA7	A24
B25	IRQ3	SA6	A25
B26	DACK2	SA5	A26
B27	T/C	SA4	A27
B28	BALE	SA3	A28
B29	+5VDC	SA2	A29
B30	OSC	SA1	A30
B31	GND	SA0	A31

D1	MEMCS16	SBHE	C1
D2	I/OCS16	LA23	C2
D3	IRQ10	LA22	C3
D4	IRQ11	LA21	C4
D5	IRQ12	LA20	C5
D6	IRQ15	LA19	C6
D7	IRQ14	LA18	C7
D8	DACK0	LA17	C8
D9	DRQ0	MEMR	C9
D10	DACK5	MEMW	C10
D11	DRQ5	SD8	C11
D12	DACK6	SD9	C12
D13	DRQ6	SD10	C13
D14	DACK7	SD11	C14
D15	DRQ7	SD12	C15
D16	+5VDC	SD13	C16
D17	MASTER	SD14	C17
D18	GND	SD15	C18

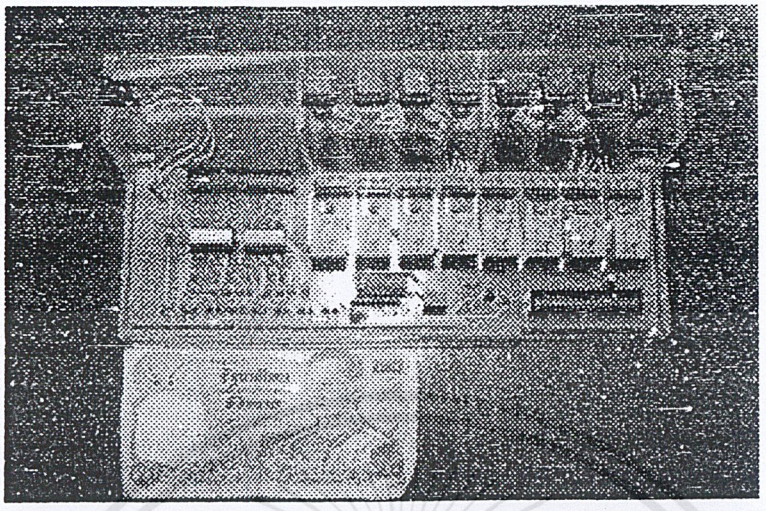
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

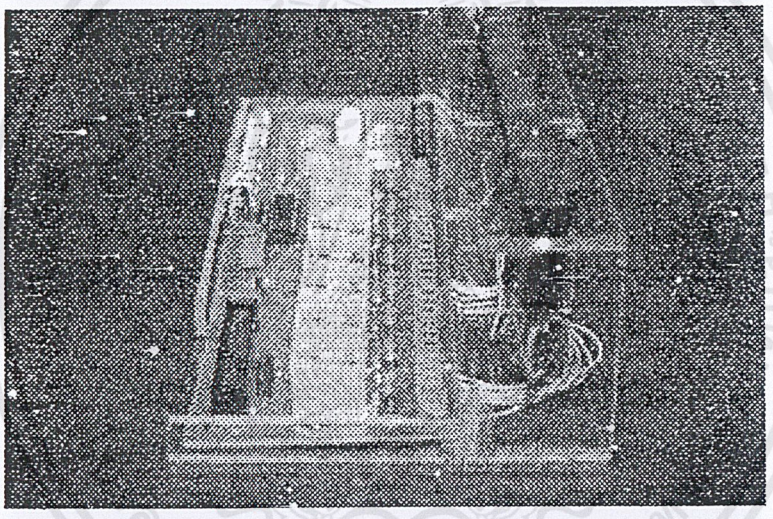


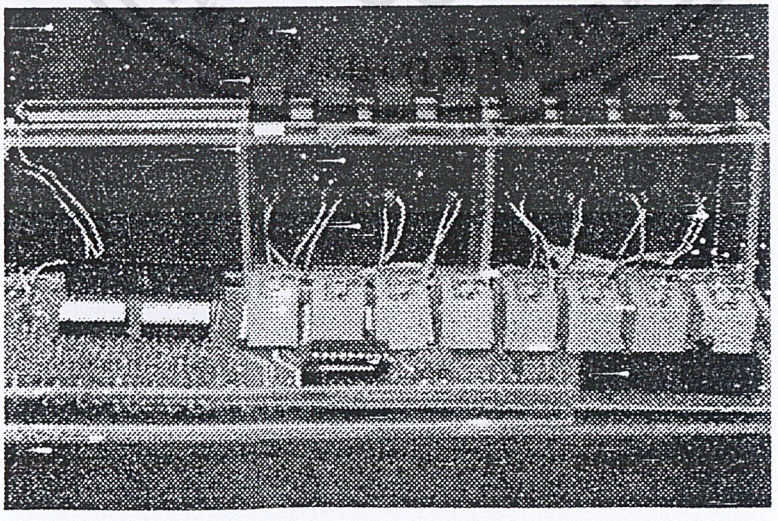
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
อุปกรณ์อินพุตและอุปกรณ์เอาท์พุต

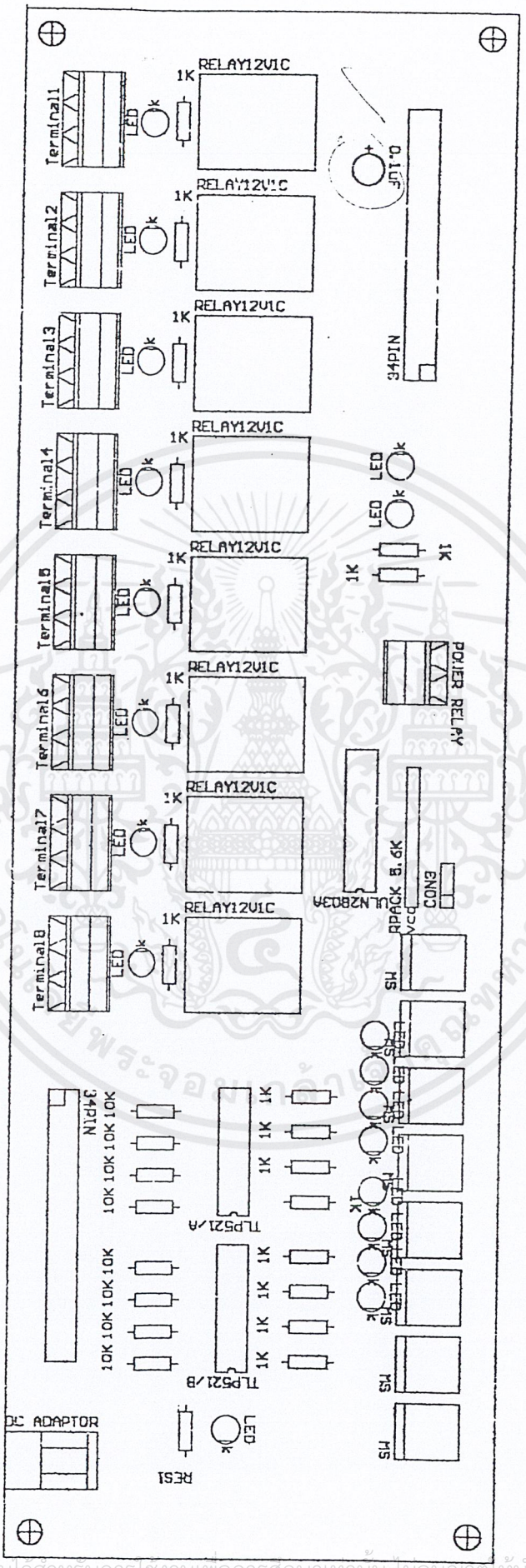
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



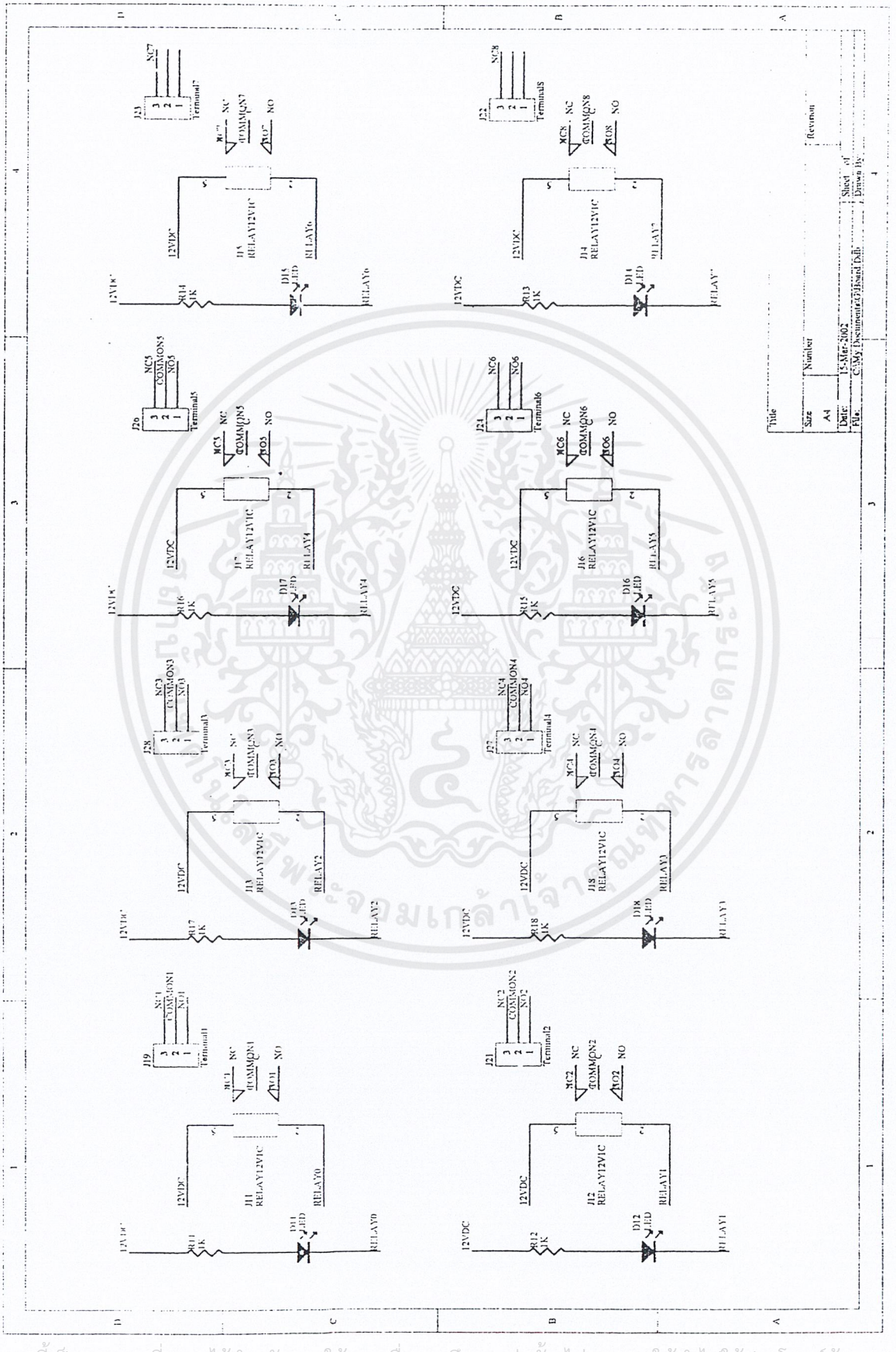




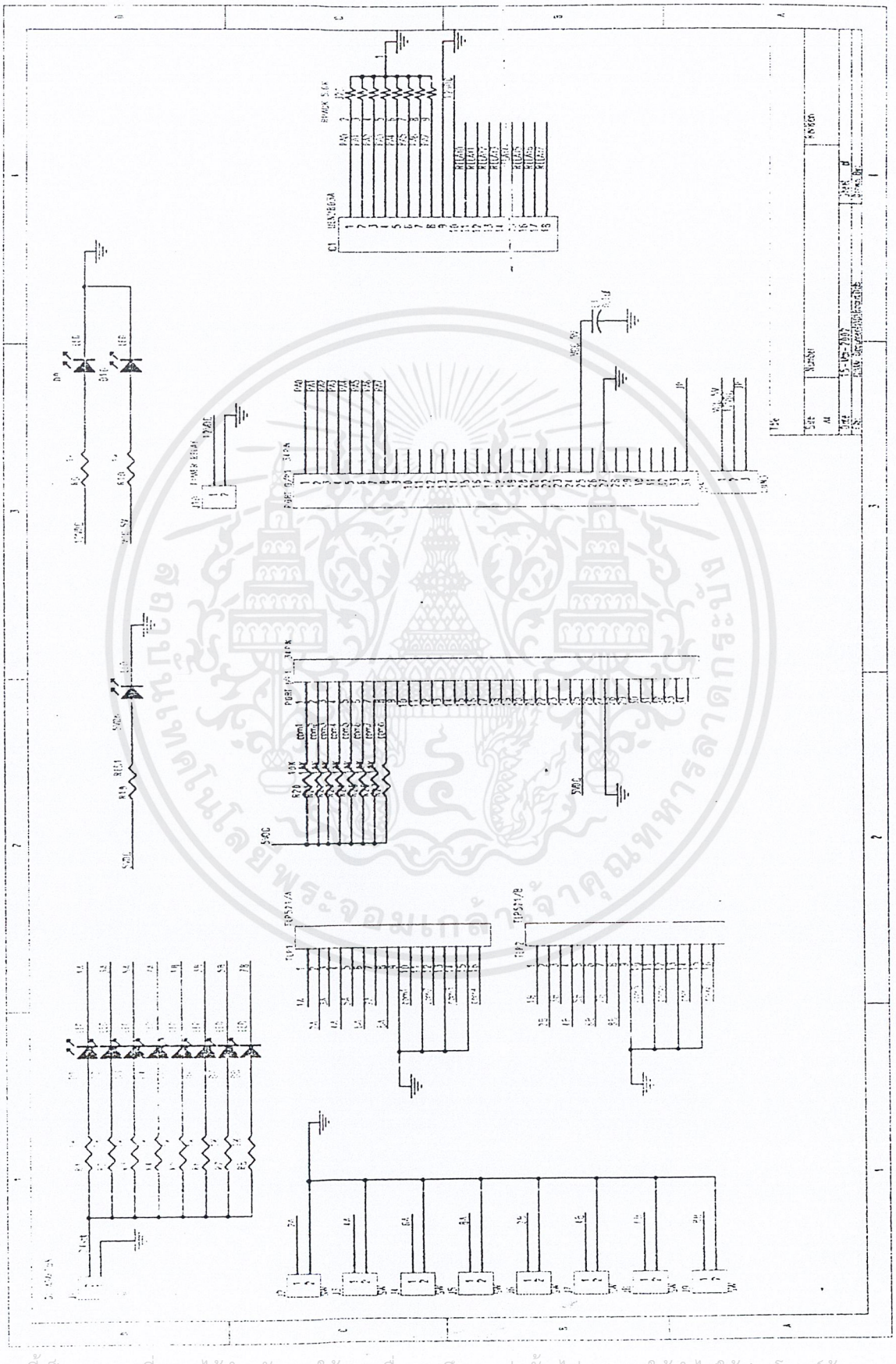
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

รายละเอียดชิพเบอร์ ULN2803A และ TLP521

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Order this document by ULN2803/D

MOTOROLA

ULN2803
ULN2804

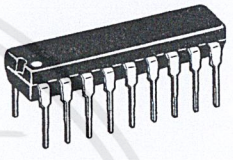
Octal High Voltage, High Current Darlington Transistor Arrays

OCTAL PERIPHERAL DRIVER ARRAYS

SEMICONDUCTOR TECHNICAL DATA

The eight NPN Darlington connected transistors in this family of arrays are ideally suited for interfacing between low logic level digital circuitry (such as TTL, CMOS or PMOS/NMOS) and the higher current/voltage requirements of lamps, relays, printer hammers or other similar loads for a broad range of computer, industrial, and consumer applications. All devices feature open–collector outputs and free wheeling clamp diodes for transient suppression.

The ULN2803 is designed to be compatible with standard TTL families while the ULN2804 is optimized for 6 to 15 volt high level CMOS or PMOS.



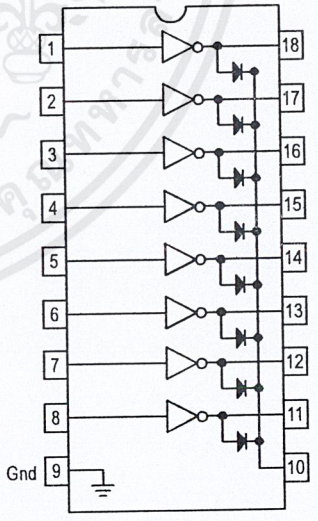
A SUFFIX
PLASTIC PACKAGE
CASE 707

MAXIMUM RATINGS (T_A = 25°C and rating apply to any one device in the package, unless otherwise noted.)

Rating	Symbol	Value	Unit
Output Voltage	V_O	50	V
Input Voltage (Except ULN2801)	V_I	30	V
Collector Current – Continuous	I_C	500	mA
Base Current – Continuous	I_B	25	mA
Operating Ambient Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	–55 to +150	°C
Junction Temperature	T_J	125	°C

$R_{\theta JA}$ = 55°C/W
Do not exceed maximum current limit per driver.

PIN CONNECTIONS



ORDERING INFORMATION

Device	Characteristics		
	Input Compatibility	$V_{CE(Max)}/I_{C(Max)}$	Operating Temperature Range
ULN2803A ULN2804A	TTL, 5.0 V CMOS 6 to 15 V CMOS, PMOS	50 V/500 mA	T_A = 0 to + 70°C

© Motorola, Inc. 1996

Rev 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ULN2803 ULN2804

ELECTRICAL CHARACTERISTICS (T_A = 25°C, unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Output Leakage Current (Figure 1)		I_{CEX}				µA
(V_O = 50 V, T_A = +70°C)	All Types		–	–	100	
(V_O = 50 V, T_A = +25°C)	All Types		–	–	50	
(V_O = 50 V, T_A = +70°C, V_I = 6.0 V)	ULN2802		–	–	500	
(V_O = 50 V, T_A = +70°C, V_I = 1.0 V)	ULN2804		–	–	500	
Collector–Emitter Saturation Voltage (Figure 2)		$V_{CE(sat)}$				V
(I_C = 350 mA, I_B = 500 µA)	All Types		–	1.1	1.6	
(I_C = 200 mA, I_B = 350 µA)	All Types		–	0.95	1.3	
(I_C = 100 mA, I_B = 250 µA)	All Types		–	0.85	1.1	
Input Current – On Condition (Figure 4)		$I_{I(on)}$				mA
(V_I = 17 V)	ULN2802		–	0.82	1.25	
(V_I = 3.85 V)	ULN2803		–	0.93	1.35	
(V_I = 5.0 V)	ULN2804		–	0.35	0.5	
(V_I = 12 V)	ULN2804		–	1.0	1.45	
Input Voltage – On Condition (Figure 5)		$V_{I(on)}$				V
(V_{CE} = 2.0 V, I_C = 300 mA)	ULN2802		–	–	13	
(V_{CE} = 2.0 V, I_C = 200 mA)	ULN2803		–	–	2.4	
(V_{CE} = 2.0 V, I_C = 250 mA)	ULN2803		–	–	2.7	
(V_{CE} = 2.0 V, I_C = 300 mA)	ULN2803		–	–	3.0	
(V_{CE} = 2.0 V, I_C = 125 mA)	ULN2804		–	–	5.0	
(V_{CE} = 2.0 V, I_C = 200 mA)	ULN2804		–	–	6.0	
(V_{CE} = 2.0 V, I_C = 275 mA)	ULN2804		–	–	7.0	
(V_{CE} = 2.0 V, I_C = 350 mA)	ULN2804		–	–	8.0	
Input Current – Off Condition (Figure 3) (I_C = 500 µA, T_A = +70°C)	All Types	$I_{I(off)}$	50	100	–	µA
DC Current Gain (Figure 2) (V_{CE} = 2.0 V, I_C = 350 mA)	ULN2801	h_{FE}	1000	–	–	–
Input Capacitance		C_I	–	15	25	pF
Turn–On Delay Time (50% E_I to 50% E_O)		t_{on}	–	0.25	1.0	µs
Turn–Off Delay Time (50% E_I to 50% E_O)		t_{off}	–	0.25	1.0	µs
Clamp Diode Leakage Current (Figure 6) (V_R = 50 V)	T_A = +25°C T_A = +70°C	I_R	–	–	50 100	µA
Clamp Diode Forward Voltage (Figure 7) (I_F = 350 mA)		V_F	–	1.5	2.0	V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ULN2803 ULN2804

TEST FIGURES

(See Figure Numbers in Electrical Characteristics Table)

Figure 1.

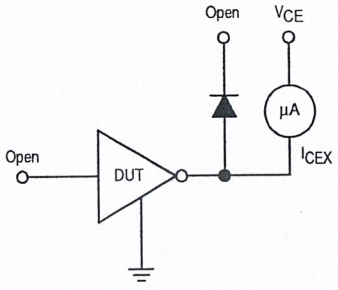


Figure 2.

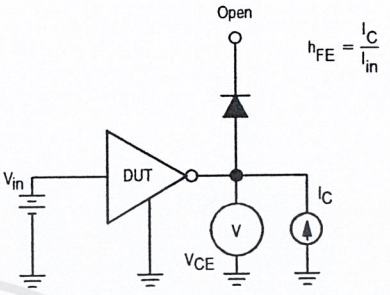


Figure 3.

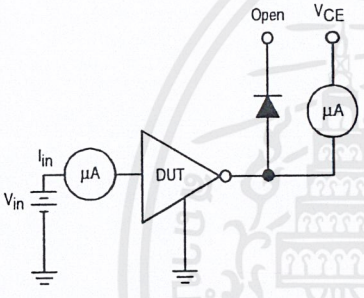


Figure 4.

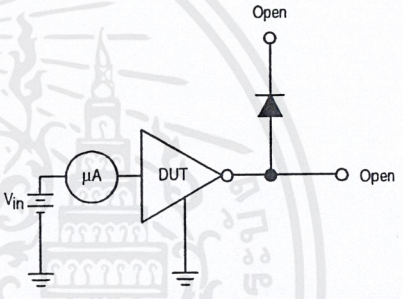


Figure 5.

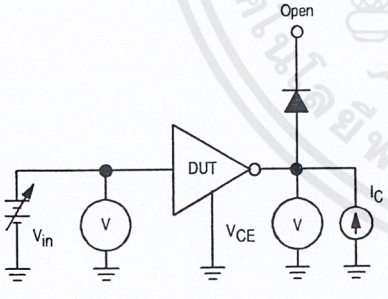


Figure 6.

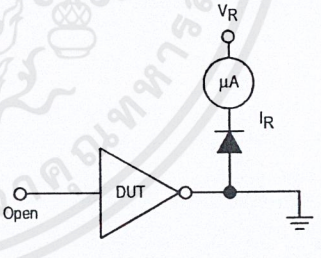
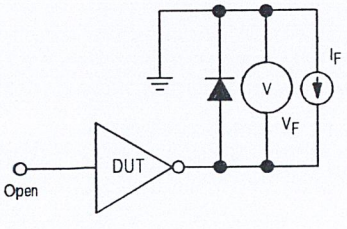


Figure 7.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL CHARACTERISTIC CURVES – T_A = 25°C, unless otherwise noted

Output Characteristics

Figure 8. Output Current versus Saturation Voltage

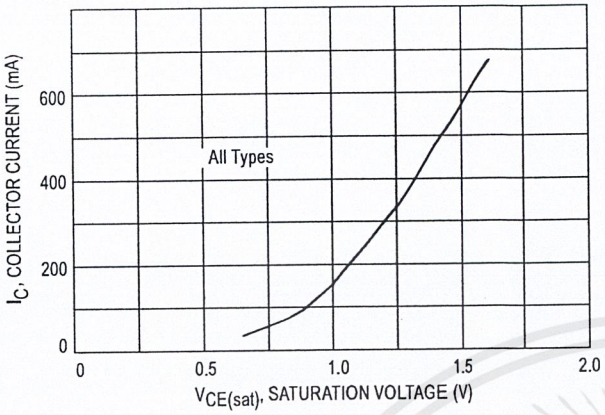
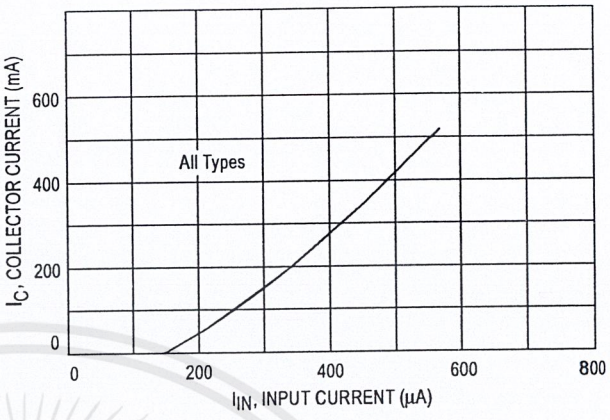


Figure 9. Output Current versus Input Current



Input Characteristics

Figure 10. ULN2803 Input Current versus Input Voltage

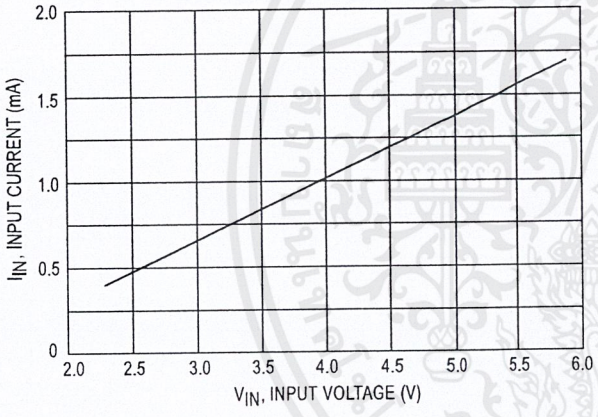


Figure 11. ULN2804 Input Current versus Input Voltage

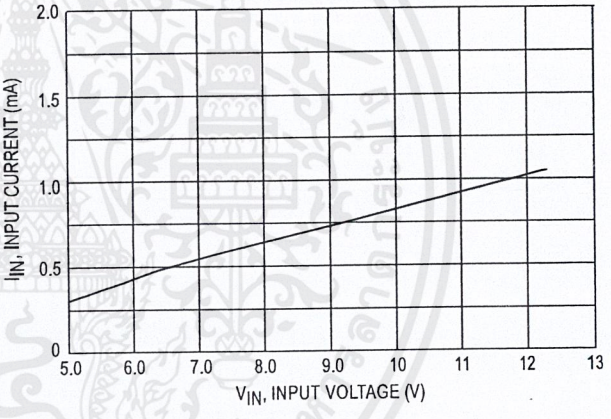
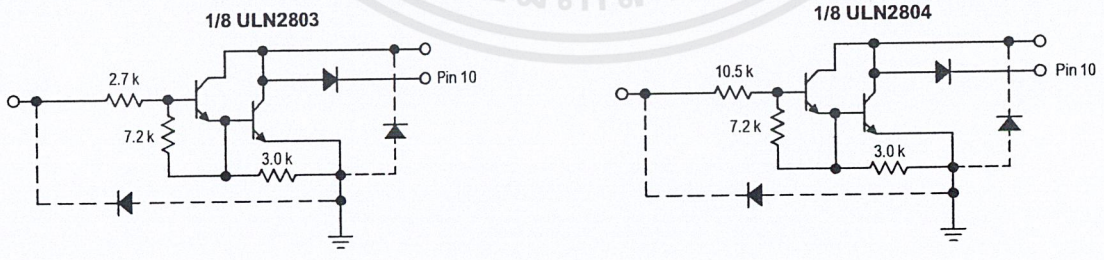


Figure 12. Representative Schematic Diagrams

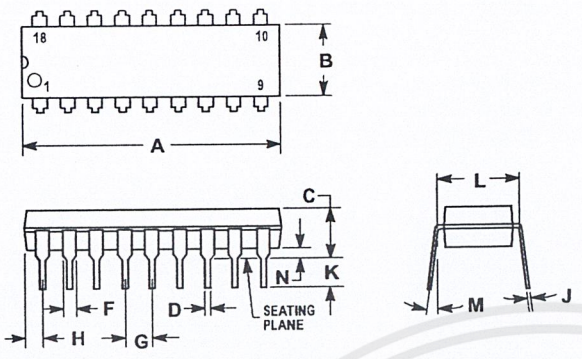


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ULN2803 ULN2804

OUTLINE DIMENSIONS

A SUFFIX
PLASTIC PACKAGE
CASE 707–02
ISSUE C



NOTES:
1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25 (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	22.22	23.24	0.875	0.915
B	6.10	6.60	0.240	0.260
C	3.56	4.57	0.140	0.180
D	0.36	0.56	0.014	0.022
F	1.27	1.78	0.050	0.070
G	2.54 BSC		0.100 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

How to reach us:
USA/EUROPE/Locations Not Listed: Motorola Literature Distribution;
P.O. Box 20912; Phoenix, Arizona 85036. 1–800–441–2447 or 602–303–5454

MFAX: RMFAX0@email.sps.mot.com – TOUCHTONE 602–244–6609
INTERNET: http://Design–NET.com

JAPAN: Nippon Motorola Ltd.; Tatsumi–SPD–JLDC, 6F Seibu–Butsuryu–Center,
3–14–2 Tatsumi Koto–Ku, Tokyo 135, Japan. 03–81–3521–8315

ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park,
51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852–26629298

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TLP521GB, TLP521-2GB, TLP521-4GB
TLP521, TLP521-2, TLP521-4

HIGH DENSITY MOUNTING PHOTOTRANSISTOR OPTICALLY COUPLED ISOLATORS

APPROVALS

- UL recognised, File No. E91231

DESCRIPTION

The TLP521, TLP521-2, TLP521-4 series of optically coupled isolators consist of infrared light emitting diodes and NPN silicon photo transistors in space efficient dual in line plastic packages.

FEATURES

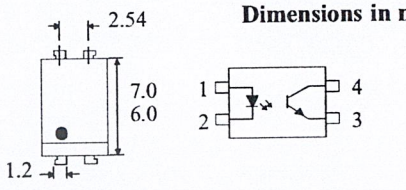
- Options :-
  10mm lead spread - add G after part no.
  Surface mount - add SM after part no.
  Tape&reel - add SMT&R after part no.
- High Current Transfer Ratio (50% min)
- High Isolation Voltage ($5.3kV_{RMS}$,$7.5kV_{PK}$)
- High BV_{CEO} (55Vmin)
- All electrical parameters 100% tested
- Custom electrical selections available

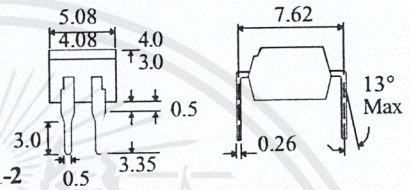
APPLICATIONS

- Computer terminals
- Industrial systems controllers
- Measuring instruments
- Signal transmission between systems of different potentials and impedances

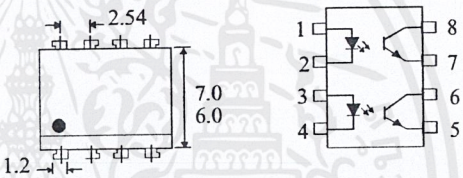
Dimensions in mm

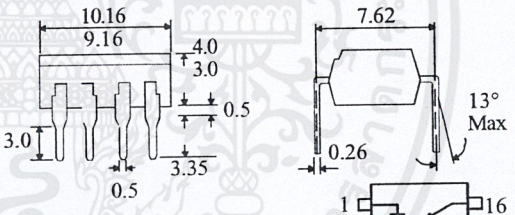
TLP521



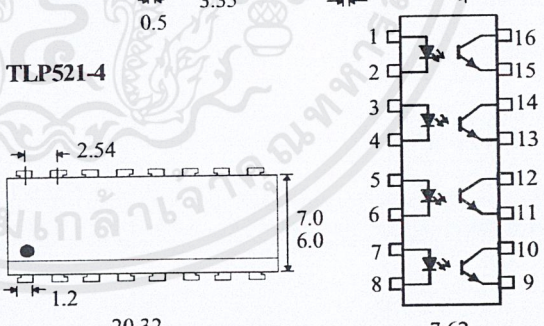


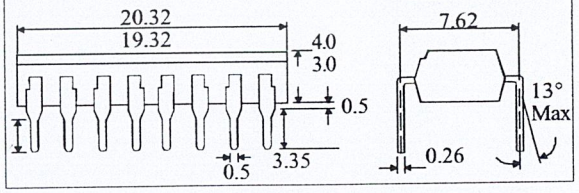
TLP521-2



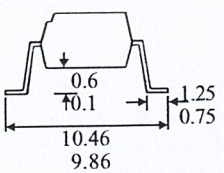


TLP521-4

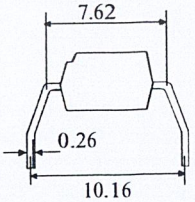




OPTION SM
SURFACE MOUNT



OPTION G



ISOCOM COMPONENTS LTD
Unit 25B, Park View Road West,
Park View Industrial Estate, Brenda Road
Hartlepool, Cleveland, TS25 1YD
Tel: (01429) 863609 Fax :(01429) 863581

ISOCOM INC
1024 S. Greenville Ave, Suite 240,
Allen, TX 75002 USA
Tel: (214) 495-0755 Fax: (214) 495-0901
e-mail info@isocom.com
http://www.isocom.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS
(25°C unless otherwise specified)

Storage Temperature	-55°C to + 125°C
Operating Temperature	-55°C to + 100°C
Lead Soldering Temperature (1/16 inch (1.6mm) from case for 10 secs)	260°C

INPUT DIODE

Forward Current	50mA
Reverse Voltage	5V
Power Dissipation	70mW

OUTPUT TRANSISTOR

Collector-emitter Voltage BV_{CEO}	55V
Emitter-collector Voltage BV_{ECO}	6V
Power Dissipation	150mW

POWER DISSIPATION

Total Power Dissipation (derate linearly 2.67mW/°C above 25°C)	200mW

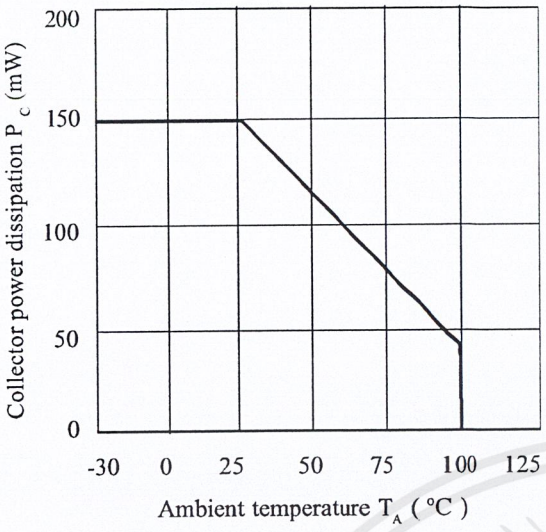
ELECTRICAL CHARACTERISTICS (T_A = 25°C Unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNITS	TEST CONDITION
Input	Forward Voltage (V_F)	1.0	1.15	1.3	V	I_F = 10mA
	Reverse Voltage (V_R)	5			V	I_R = 10μA
	Reverse Current (I_R)			10	μA	V_R = 5V
Output	Collector-emitter Breakdown (BV_{CEO}) (Note 2)	55			V	I_C = 0.5mA
	Emitter-collector Breakdown (BV_{ECO})	6			V	I_E = 100μA
	Collector-emitter Dark Current (I_{CEO})			100	nA	V_{CE} = 24V
Coupled	Current Transfer Ratio (CTR) (Note 2)					
	TLP521, TLP521-2, TLP521-4	50		600	%	5mA I_F, 5V V_{CE}
	CTR selection available BL	200		600	%	
	GB	100		600	%	
	GB	30			%	1mA I_F, 0.4V V_{CE}
	Collector-emitter Saturation Voltage $V_{CE\ (SAT)}$			0.4	V	8mA I_F, 2.4mA I_C
	-GB			0.4	V	1mA I_F, 0.2mA I_C
	Input to Output Isolation Voltage V_{ISO}	5300			V_{RMS}	See note 1
		7500			V_{PK}	See note 1
	Input-output Isolation Resistance R_{ISO}	$5x10^{10}$			Ω	V_{IO} = 500V (note 1)
	Rise Time tr		2		μs	V_{CC} = 10V ,
	Fall Time tf		3		μs	I_C = 2mA, R_L = 100Ω
	Turn-on Time ton		3		μs	
	Turn-off Time toff		3		μs	

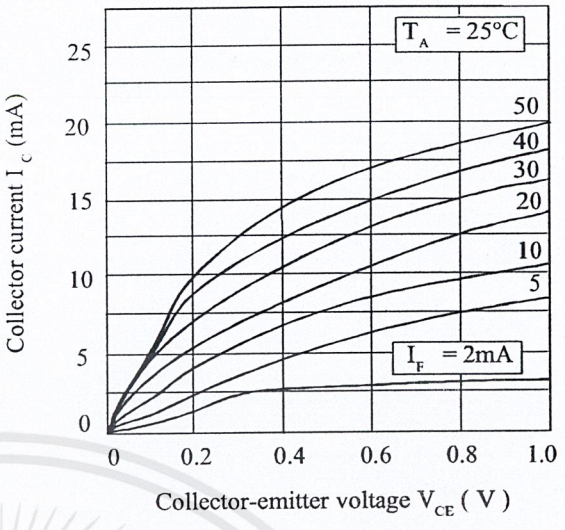
Note 1 Measured with input leads shorted together and output leads shorted together.
Note 2 Special Selections are available on request. Please consult the factory.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

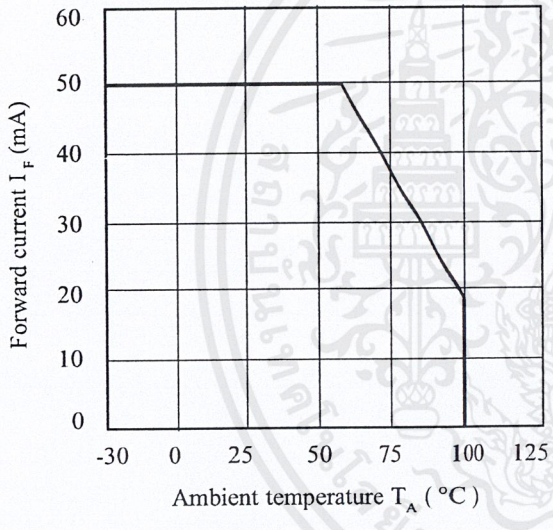
Collector Power Dissipation vs. Ambient Temperature



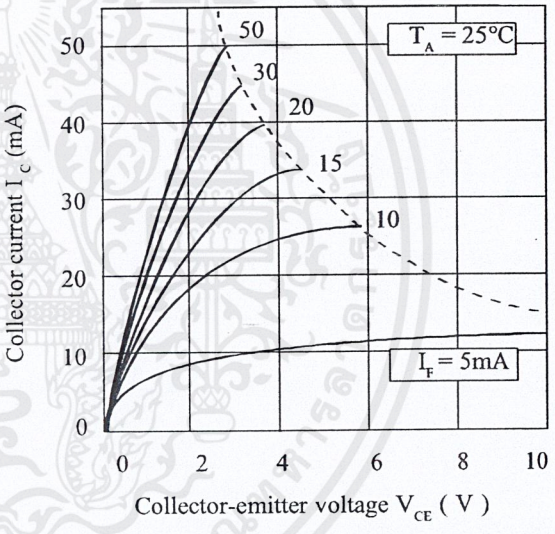
Collector Current vs. Low Collector-emitter Voltage



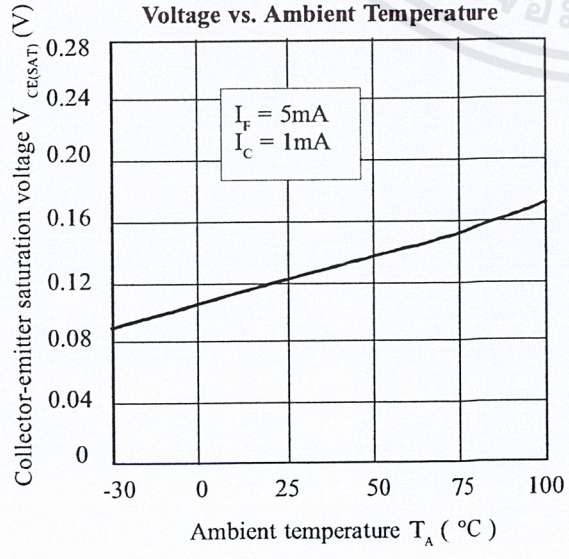
Forward Current vs. Ambient Temperature



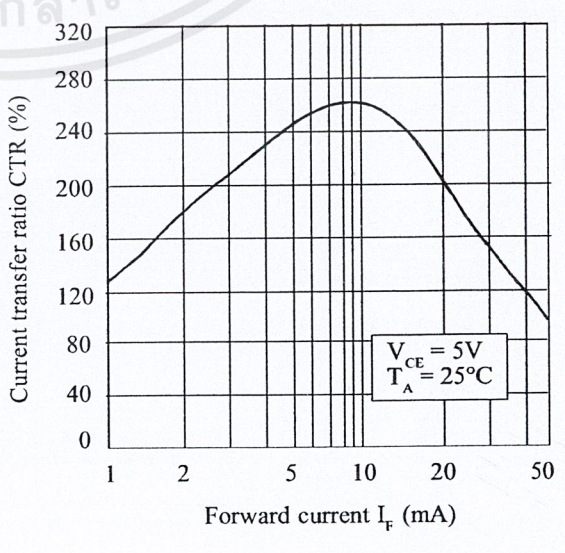
Collector Current vs. Collector-emitter Voltage



Collector-emitter Saturation Voltage vs. Ambient Temperature



Current Transfer Ratio vs. Forward Current



DB92546m-AAS/A1

7/12/00 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้คงจะไม่สามารถสำเร็จลุล่วงๆไปได้ด้วยดี ถ้าปราศจากบุคคลหลายๆ ท่านที่มีพระคุณอย่างสูงและที่ให้กำลังใจอยู่ข้างๆ แม้นยามที่ท้อแท้ก็ช่วยให้แก้ไขปัญหาได้อีกครั้ง ไม่ว่าจะเป็นเรื่องเกี่ยวกับโปรแกรม ชิ้นงาน หรือกระทั่งปริญญานิพนธ์ฉบับนี้ ทางผู้จัดทำจึงขอขอบพระคุณไว้ ณ ที่นี้

ขอขอบพระคุณ อาจารย์ เทพจิตร เชยโภคา อาจารย์ที่ปรึกษาที่คอยให้คำปรึกษา ชี้แนะแนวทางที่เป็นประโยชน์ในการทำโครงงานครั้งนี้

ขอขอบพระคุณ อาจารย์ทุกท่านในภาควิชาวิศวกรรมระบบควบคุม ที่ทุกท่านสั่งสอนความรู้และคำปรึกษาอันมีค่ายิ่งแก่ผู้จัดทำ

ขอขอบคุณ คุณ ชาญชัย จอประเสริฐกุล ที่ให้คำปรึกษาทางด้านโปรแกรม และทางด้านแนวคิดในการเขียนโปรแกรม

ขอขอบคุณ คุณ ณัฐวุฒิ ทองอารีย์ ที่คอยช่วยเหลือในด้านการทำอุปกรณ์ฮาร์ดแวร์

ขอขอบคุณ เพื่อนๆ พี่ๆ น้องๆ ที่ทุกคนให้กำลังใจ ให้คำปรึกษาทั้งในเรื่องที่เกี่ยวข้องกับโครงงาน และที่ไม่เกี่ยวข้องกับโครงงาน

อีกทั้งยังขอขอบพระคุณ คุณพ่อ คุณแม่ ที่ให้กำเนิด และคอยเป็นห่วงเป็นใยมาจนถึงทุกวันนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. บริษัท เอฟ.เอ.เทค จำกัด, "คู่มือการเขียนโปรแกรมสำหรับ MITSUBISHI FX series ฉบับปรับปรุงแก้ไขและเพิ่มเติมโปรแกรม"

2. ณรงค์ ตันชีวะวงศ์, "ระบบพีแอลซี" , สมาคมส่งเสริมเทคโนโลยีไทยญี่ปุ่น, 2544

3. สัจจะ จรัสรุ่งรวีวร และ จักรพงษ์ สุขประเสริฐ, "คู่มือสร้างแอพพลิเคชั่น ด้วย Delphi 5.0 ฉบับสมบูรณ์", อินโฟเพรส, 2543

4. Tom Swan, "Delphi4 Bible"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้