การแสดงสถานะของ PLC
Display status of PLC

โดย

นาย ธวัชชัย ก้องกิตติตระกูล

นาย สุธรรม ไชยมุติ

นาย สุริยา แป้นเกิด

อาจารย์ที่ปรึกษา

อ.เชื้อ นกอยู่

เลขเรียกหนังสือ ปพ วศก ธรม

เลขทะเบียน 040572

วัน เดือน ปี 19 ตค มย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมการวัดคุมทางอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

ปริญญานิพนธ์ปีการศึกษา 2541

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

เรื่อง การแสดงสภาวะพีแอลซี

โดย

นายธวัชชัย ก้องกิตติคระกูล

นายสุธรรม ไชยมุติ

นายสุริยา แป้นเกิด

.....อาจารย์ที่ปรึกษา

(อ.เชื้อ นกอยู่)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบแสดงสภาวะพีแอลซี

DISPLAY STATUS OF PLC SYSTEM

โดย นายธวัชชัย ก้องกิตติตระกูล
นายสุธรรม ไชยมุติ
นายสุริยา แป้นเกิด

อาจารย์ที่ปรึกษา อ.เชื้อ นกอยู่

บทคัดย่อ

โครงงานนี้เป็นการนำเสนอเครื่องแสดงสภาวะการทำงานของเครื่องควบคุมแบบตรรกที่สามารถแสดงสภาวะการทำงานของเครื่องควบคุมแบบตรรกเป็นข้อความภาษาไทยซึ่งสามารถเปลี่ยนข้อความภาษาไทยที่จะแสดงได้ โดยมีไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงาน มีการรับอินพุทจากเครื่องควบคุมแบบตรรก มีการรับการโปรแกรมจากคีย์บอร์ด มีจอแสดงผลแบบกราฟฟิกเป็นตัวแสดงผล มีระบบไฟสำรองป้องกันการเสียหายของข้อมูล และระบบวอช์ทด็อกตรวจสอบการทำงานของไมโครคอนโทรลเลอร์ ในกรณีที่ตัวไมโครคอนโทรลเลอร์ทำงานผิดพลาด

Abstract

This project presents the display status of PLC which is able to work in Thai language which can be altered. The system is controlled by microcontroller and has received functions inputs from PLC, program changing made by the keyboard,there is a power backup system to prevent damages in data, there is a watch dog system to check the performance of microcontroller in case error.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หัวข้อ	หน้า
1.1 แนวความคิดในการทำปริญญานิพนธ์	1
1.2 เนื้อหาโดยสังเขปของปริญญานิพนธ์	1
2.1 หลักการโดยทั่วไปของการแสดงสภาวะพีแอลซี	2
2.2 ความต้องการของระบบแสดงสภาวะพีแอลซี	3
2.3 คุณสมบัติของอุปกรณ์ที่ใช้ในระบบแสดงสภาวะพีแอลซี	3
2.3.1 อุปกรณ์ไมโครคอนโทรลเลอร์	3
2.3.2 จอแสดงผลกราฟฟิกแอลซีดี	10
2.3.3 คีย์บอร์ด	16
2.3.4 พอร์ทสื่อสารอนุกรม	18
2.3.5 โปรโตคอล	22
2.3.6 พีแอลซี	42
3.1 หลักการออกแบบวงจร	46
3.2 การอินเตอร์เฟส	46
3.2.1 การอินเตอร์เฟสกับกราฟฟิกแอลซีดี	46
3.2.2 การอินเตอร์เฟสกับคีย์บอร์ด	46
3.2.3 การอินเตอร์เฟสกับพีแอลซี	47
3.3 วิธีการออกแบบตัวอักษรภาษาไทย	48
3.4 วิธีการใช้งานเครื่องแสดงสภาวะพีแอลซี	53
4.1 วิธีการทดลอง	55
4.2 ผลการทดลอง	55
5.1 ปัญหาที่เกิดขึ้นกับการทดลอง	61
5.2 แนวทางในการแก้ปัญหา	61
ภาคผนวก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 แนวความคิดในการทำปริญญานิพนธ์

โรงงานอุตสาหกรรมโดยส่วนใหญ่ในปัจจุบันนิยมใช้พีแอลซี(PLC)ในการควบคุมการทำงาน เมื่อระบบการควบคุมมีปัญหาตัวพีแอลซีจะแจ้งการทำงานผิดพลาดนั้นเป็นรหัสตัวเลขผู้ควบคุมต้องไปเปิดคู่มือการใช้งานที่เป็นภาษาอังกฤษเพื่อหาว่ารหัสที่แจ้งออกมามีความหมายว่าอย่างไร แล้วจึงทำการแก้ไขการทำงานที่ผิดพลาดนั้น ในกรณีที่ผู้ควบคุมอ่านคู่มือการใช้งานแล้วแปลความหมายของรหัสที่แจ้งการผิดปกติในคู่มือไม่ถูกต้องจะไม่ทราบถึงสาเหตุที่แท้จริง จึงทำให้เสียเวลาในการตรวจสอบหาสาเหตุที่เกิดขึ้น

สำหรับปริญญานิพนธ์นี้จะได้นำเสนอถึงการแจ้งการทำงานผิดพลาดเป็นภาษาไทย ซึ่งประกอบด้วยหน่วยย่อยๆ คือหน่วยของการประมวลผลส่วนกลาง(CPU),หน่วยแสดงผล (Graphic LCD),หน่วยอินพุท(Input) และหน่วยการรับข้อมูล(RS-232) ซึ่งเป็นหน่วยการทำงานหลักของระบบ โดยในเนื้อหาจะแจกแจงหน่วยการทำงานดังกล่าวโดยละเอียด ทฤษฎีและหลักการของระบบ,การทำงานของระบบ,การออกแบบและการสร้างระบบ,การทดลอง,ตลอดจนแนวทางในการพัฒนาและประยุกต์ใช้งานระบบ

1.2 เนื้อหาโดยสังเขปของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ เป็นการออกแบบระบบการแสดงสภาวะพีแอลซี ซึ่งสามารถนำไปประยุกต์ใช้งานกับกระบวนการที่ใช้ พีแอลซีควบคุมได้ทางคณะผู้จัดทำได้จัดแบ่งเนื้อหาของระบบแสดงสภาวะ PLC ไว้ในบทต่าง ๆ ดังต่อไปนี้

บทที่ 2 ทฤษฎีและหลักการ เป็นการกล่าวถึงทฤษฎีต่าง ๆ ที่เกี่ยวกับระบบโดยรวมทั้งหมดตั้งแต่แนวความคิดเริ่มต้นและการสร้างระบบไปจนถึงหน้าที่ คุณสมบัติของส่วนต่าง ๆ ที่เกี่ยวข้องกับระบบ

บทที่ 3 การออกแบบและการสร้างเป็นการกล่าวถึง วิธีการออกแบบตามที่ได้มีแนวความคิดไว้แล้วในทางทฤษฎีและการเลือกใช้อุปกรณ์ต่าง ๆ เพื่อนำมาใช้กับระบบ เพื่อให้ได้ประสิทธิภาพสูงสุด เท่าที่เป็นไปได้

บทที่ 4 การทดลองและผลการทดลอง ซึ่งจะกล่าวถึงการทดลอง โดยสรุปของปริญญานิพนธ์ โดยการเปรียบเทียบกับความต้องการของระบบ ที่คาดหมายเอาไว้ในทางทฤษฎี

บทที่ 5 บทสรุปปัญหา แนวทางแก้ไข และการพัฒนาระบบ เป็นการกล่าวถึง ปัญหาต่าง ๆ ที่เกิดขึ้นกับระบบที่ได้ทดลอง และปฏิบัติพร้อมทั้งแนวทางในการแก้ไขและพัฒนาให้ระบบมีประสิทธิภาพ และประโยชน์ที่ได้มีคุณค่าสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 หลักการโดยทั่วไปของการแสดงสภาวะพีแอลซี(PLC)

ในระบบการแสดงสภาวะพีแอลซีนั้นมีส่วนประกอบที่สำคัญอยู่ 3 ส่วนคือส่วนของหน่วยประมวลผลส่วนกลาง(Central Processing Unit), หน่วยของการรับอินพุท(Input Unit) และหน่วยแสดงผล(GLCD) ซึ่งทั้ง 3 ส่วนนี้จะต้องมีการทำงานที่สัมพันธ์กัน

2.1.1 หน่วยประมวลผลส่วนกลาง

จะทำหน้าที่ควบคุมการทำงานทั้งหมด เช่น การรับค่ารหัสจาก พีแอลซี ผ่านทางพอร์ทอนุกรมRS232มาเป็นตัวสั่งการให้จอแสดงผลกราฟฟิกแอลซีดีแสดงข้อความแจ้งความผิดพลาดตามที่ได้ตั้งไว้การรับ อินพุทจากคีย์บอร์ดเพื่อที่จะตั้งหน้าที่การทำงานหรือในกรณีที่ไฟฟ้าขัดข้องหน่วยประมวลผลนี้จะนำค่าที่ได้จากการกำหนดหน้าที่การทำงานเก็บไว้ในส่วนของ แมมโมรี่แบคอัพ(Memory Backup)หลังจากที่ระบบไฟฟ้าเป็นปกติค่าเหล่านี้จะถูกโหลด(Load)ออกมาใช้งานตามเดิมจะเห็นได้ว่าหน้าที่การทำงานของหน่วยประมวลผลส่วนกลางนี้จะต้องอาศัยอุปกรณ์ซึ่งสามารถที่จะทำการโปรแกรม(Program)ลงไปได้และอุปกรณ์ที่ว่านี้ก็จะเป็นอุปกรณ์ไมโครคอนโทรเลอร์(Microcontroller)ซึ่งจะได้กล่าวถึงอุปกรณ์ชนิดนี้ในส่วนถัดไป

2.1.2 หน่วยของการรับอินพุท

เป็นหน่วยการตั้งหน้าที่การทำงานของระบบการแสดงสภาวะพีแอลซี เช่นจะให้ระบบรอรับรหัสค่าผิดพลาดจากพีแอลซี การส่งคำสั่งไปให้พีแอลซีส่งข้อมูลกลับมาเพื่อแสดงผลทางหน้าจอกราฟฟิกแอลซีดี ซึ่งทั้งสองหน้าที่นี้จะทำการติดต่อกันระหว่างพีแอลซีกับหน่วยประมวลผลส่วนกลางได้โดยผ่านทางพอร์ทอนุกรมอาร์เอสสองสามสอง และการเปลี่ยนแปลงข้อความในการแจ้งความผิดพลาดในการทำงานของพีแอลซี ในที่นี้จะเลือกใช้คีย์บอร์ดซึ่งเป็นอุปกรณ์อีกชนิดหนึ่งที่สามารถกำหนดค่าให้ระบบทำงานตามหน้าที่การทำงานได้ตามที่ต้องการ

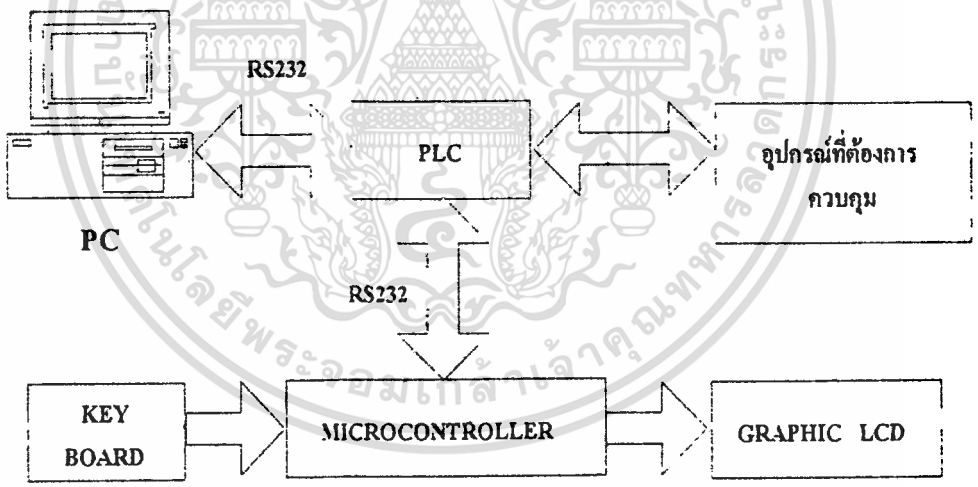
2.1.3 หน่วยแสดงผล

เป็นหน่วยซึ่งถูกควบคุมโดยหน่วยประมวลผลส่วนกลางซึ่งจะมีการทำงานที่สัมพันธ์กับอินพุทในการแสดงการตั้งค่า และการแสลงข้อความในการแจ้งความผิดพลาด อุปกรณ์ที่เลือกใช้คือกราฟฟิกแอลซีดี(Graphic LCD)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ความต้องการของระบบแสดงสภาวะพีแอลซี

ก่อนที่จะกำหนดเป้าหมายความต้องการของระบบจำเป็นอย่างยิ่งที่จะต้องรู้ถึงการทำงานของระบบก่อนระบบนี้จะมีการทำงานดังนี้คือเมื่ออุปกรณ์อินพุทหรืออุปกรณ์เอาท์พุทที่ถูกควบคุมการทำงานโดยพีแอลซีทำงานผิดพลาดพีแอลซีจะสามารถรับรู้ได้ว่าจุดใดทำงานผิดพลาดโดยการรับรู้นี้จะเกิดจากการเขียนแลดเดอร์(Ladder)ควบคุมการทำงานของผู้ควบคุม หลังจากนั้นพีแอลซีจะส่งระหัสค่าผิดพลาดออกไปทางพอร์ทอนุกรม RS232ไปให้ไมโครคอนโทรลเลอร์ (Microcontroller) เมื่อไมโครคอนโทรลเลอร์รับค่าระหัสความผิดพลาดมาแล้วจะทำการส่งข้อความภาษาไทยไปยังจอแสดงผลเพื่อให้จอแสดงผลทำการแสดงผลต่อไป ข้อความภาษาไทยที่แสดงตามรหัสความผิดพลาดยังสามารถเปลี่ยนแปลงได้โดยการกดเลือกตัวอักษรได้ทางคีย์บอร์ดที่ต่ออยู่ร่วมกับไมโครคอนโทรลเลอร์นอกจากนี้ยังต้องมีการตรวจสอบการทำงานของไมโครคอนโทรลเลอร์ด้วยโดยการใช้ระบบวอท์ชด็อก(Watch dog)ในการตรวจสอบในกรณีที่ไมโครคอนโทรลเลอร์ไม่ทำงานระบบวอท์ชด็อกจะทำการรีเซ็ท(Reset)ตัวไมโครคอนโทรลเลอร์เพื่อให้ตัวไมโครคอนโทรลเลอร์เริ่มทำงานใหม่อีกครั้งหนึ่ง



รูปที่ 2.1 แสดงบล็อคไดอะแกรมระบบสภาวะของ PLC

2.3 คุณสมบัติของอุปกรณ์ที่ใช้ในระบบแสดงสภาวะพีแอลซี

2.3.1 อุปกรณ์ไมโครคอนโทรลเลอร์

คุณสมบัติของอุปกรณ์ไมโครคอนโทรลเลอร์เบอร์ 8051

1. ใช้แหล่งจ่ายไฟขนาด 5 โวลท์เพียงแหล่งเดียว
2. เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิท
3. มีวงจรออสซิเลเตอร์และวงจรผลิตสัญญาณนาฬิกาภายในไอซี

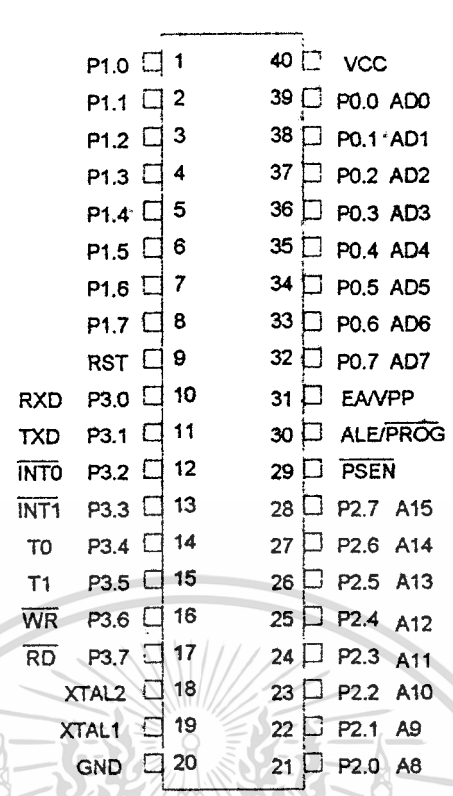
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. มีขาสัญญาณอินพุตเอาต์พุตจำนวน 32 บิท
5. สามารถเชื่อมต่อหน่วยความจำข้อมูลภายนอก (external data memory) โดยอ้างตำแหน่งแอดเดรสได้ถึง 64 กิโลไบท์
6. สามารถเชื่อมต่อหน่วยความจำโปรแกรมภายนอก (external program memory) โดยอ้างตำแหน่งแอดเดรสได้ถึง 64 กิโลไบท์
7. มีหน่วยความจำโปรแกรมภายในตัว(on- chip pro-gram memory) ขนาด 4 กิโลไบท์
8. มีหน่วยความจำข้อมูลภายในตัว (on-chip data memory) ขนาด 128 ไบต์
9. หน่วยความจำข้อมูลภายในบางส่วนสามารถเข้าถึงข้อมูลระดับบิตได้ด้วย ส่งผลให้การเขียนโปรแกรมทำได้ง่ายมากขึ้น
10. มีไทเมอร์/เคาน์เตอร์ (timer/counters) ขนาด 16 บิต จำนวน 2ตัว
11. การอินเตอร์รัปต์สามารถทำได้จาก 5 แหล่งกำเนิด โดยการอินเตอร์รัปต์ยังสามารถจัดระดับความสำคัญได้เป็น 2 ระดับ
12. มีพอร์ตสื่อสารอนุกรมภายในตัวเอง ซึ่งทำงานเป็นแบบฟูลดูเพล็กซ์ (full duplex)
13. มีคำสั่งในการคำนวณคณิตศาสตร์และทางตรรกศาสตร์
14. คำสั่งโดยส่วนใหญ่ใช้เวลาการทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 เมกะเฮิรตซ์

โครางสร้างภายนอกของไมโครคอนโทรลเลอร์เบอร์ 8051

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุกเบอร์จะมีตำแหน่งขาพื้นฐานที่เหมือนกัน ดังแสดงในรูปที่ 2.2 สำหรับหน้าที่การใช้งานของแต่ละขามีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดงการจัดตำแหน่งขาต่างๆ ของไมโครคอนโทรลเลอร์ตระกูล MCS-51

หน้าที่การใช้งานของแต่ละขามีดังนี้

- ขา Vcc เป็นขาป้อนแรงดันไฟเลี้ยง + 5 โวลต์
- ขา Vss เป็นขากราวด์
- ขาพอร์ต 0 (Port 0) มี 8 ขา ได้แก่ขา P0.0-P0.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดขาพอร์ตเหล่านั้นอยู่ในสถานะปล่อยลอย ซึ่งในสถานะนี้เองที่สามารถนำมาใช้เป็นพอร์ตอินพุตเอาต์พุตและมันยังถูกใช้งานในการติดต่อกับหน่วยความจำภายนอกด้วย โดยทำหน้าที่ในการกำหนดตำแหน่งแอดเดรสไบต์ต่ำ (A0-A7) ซึ่งจะใช้งานเป็นแบบมัลติเพล็กซ์กับการรับส่งข้อมูลขนาด 8 บิต (D0-D7)
- ขาพอร์ต 1 (Port 1) มี 8 ขา ได้แก่ขา P1.0-P1.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากนี้สำหรับเบอร์ 8032 และ8052 ขาพอร์ตP1.0 และ P1.1 จะถูกนำมาใช้งานเป็นขา T2 และ T2EX ตามลำดับด้วย
- ขาพอร์ต 2 (Port 2) มี 8 ขาได้แก่ขา P2.0-P2.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วมันยังถูกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานในการติดต่อกับหน่วยความจำภายนอกด้วย โดยทำหน้าที่ในกำหนดตำแหน่งแอดเดรสไบต์สูง (A8-A15)

- ขาพอร์ต 3 (Port 3) มี 8 ขาได้แก่ขาP3.0-P3.7 เป็นขาพอร์อินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับใช้งานทั่วไปโดยถ้าใช้งานเป็นอินพุตพอร์ตอินพุทต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในหน้าที่พิเศษต่าง ๆ ดังแสดงในตารางที่ 1

ขาพอร์ท	หน้าที่ พิเศษ
P3.0	RXD(serial input port)
P3.1	TXD(serial output port)
P3.2	INT0(external interrupt 0)
P3.3	INT1(external interrupt 1)
P3.4	T0(Timer 0 external input)
P3.5	T1(Timer 1 external input)
P3.6	WR(external data memory write strobe)
P3.7	RD(external data memory read strobe)

ตารางที่ 1 แสดงหน้าที่พิเศษของแต่ละขาของพอร์ท P3

- ขารีเซต(RST)ใช้สำหรับการรีเซตการทำงานของไมโครคอนโทรลเลอร์โดยการรีเซตต้องคงสถานะเป็น 1 อย่างน้อยนาน 2 แมชีนไซเกิลในขณะที่ออสซิลเลอร์ยังทำงานอยู่
- ขา ALE/PROG เป็นขาสัญญาณเพื่อทำหน้าที่ควบคุมการแลตช์ (latch) ค่าตำแหน่งแอดเรสไบต์ต่ำ (Address Latch Enable) เมื่อต้องการติดต่อกับหน่วยความจำภายนอก นอกจากนี้ขานี้ยังทำหน้าที่เป็นอินพุตรับพัลส์ในการโปรแกรม (program pulse input) ในส่วนของหน่วยความจำ EPROM สำหรับไมโครคอนโทรลเลอร์ในตระกูล MCS-51 ที่มีหน่วยความจำโปรแกรมภายในเป็น EPROM
- ขา PSEN (Program Store Enable) ทำหน้าที่เป็นสัญญาณสโตรบเพื่ออ่านคำสั่งจากหน่วยความจำโปรแกรมภายนอกเมื่อไมโครคอนโทรลเลอร์ประมวลผลคำสั่งจากหน่วยความจำภายนอก ขานี้จะส่งสัญญาณสโตรบจำนวน 2 ครั้งในแต่ละแมชีนไซเกิล แต่ในขณะที่ติดต่อกับหน่วยความจำข้อมูลภายนอกจะไม่มีการส่งสัญญาณสโตรบแต่อย่างใด
- ขาEA/VPP (External Access enable/Vpp) เป็นขาสำหรับการเลือกใช้หน่วยความจำโปรแกรมจากภายในหรือจากภายนอก โดยถ้ามีสถานะเป็น 0 จะหมายถึงให้ไมโครคอนโทรลเลอร์รับคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารนี้ทุกครั้งที่มีการนำไปใช้

จากหน่วยความจำภายนอกที่ตำแหน่งแอดเดรส 0-0FFFH (0-1FFFH ถ้าเป็นเบอร์ 8052) อย่างไรก็ตามถ้าบิตป้องกัน(security bit)ในหน่วยความจำEPROMถูกโปรแกรมไว้ไมโครคอนโทรลเลอร์จะไม่รับคำสั่งจากหน่วยความจำภายนอกเลย นอกจากนี้ขานี้ยังทำหน้าที่รับแรงดันไฟสำหรับการโปรแกรม (Vpp) ขนาด 21 โวลต์ เพื่อใช้ในระหว่างการโปรแกรม EPROM

- ขา XTAL1และขาXTAL2 เป็นขาอินพุตและเอาต์พุตของวงจรอินเวอร์ติ้งออสซิลเลเตอร์แอมปลิไฟเออร์ (inverting oscillator amplifier) สำหรับใช้ต่อร่วมกับคริสตอลภายนอก

การจัดหน่วยความจำ

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 แบ่งชนิด หรือหน้าที่ของหน่วยความจำออกเป็น 2 ส่วนคือหน่วยความจำโปรแกรม(program memory) และหน่วยความจำข้อมูล (data memory)

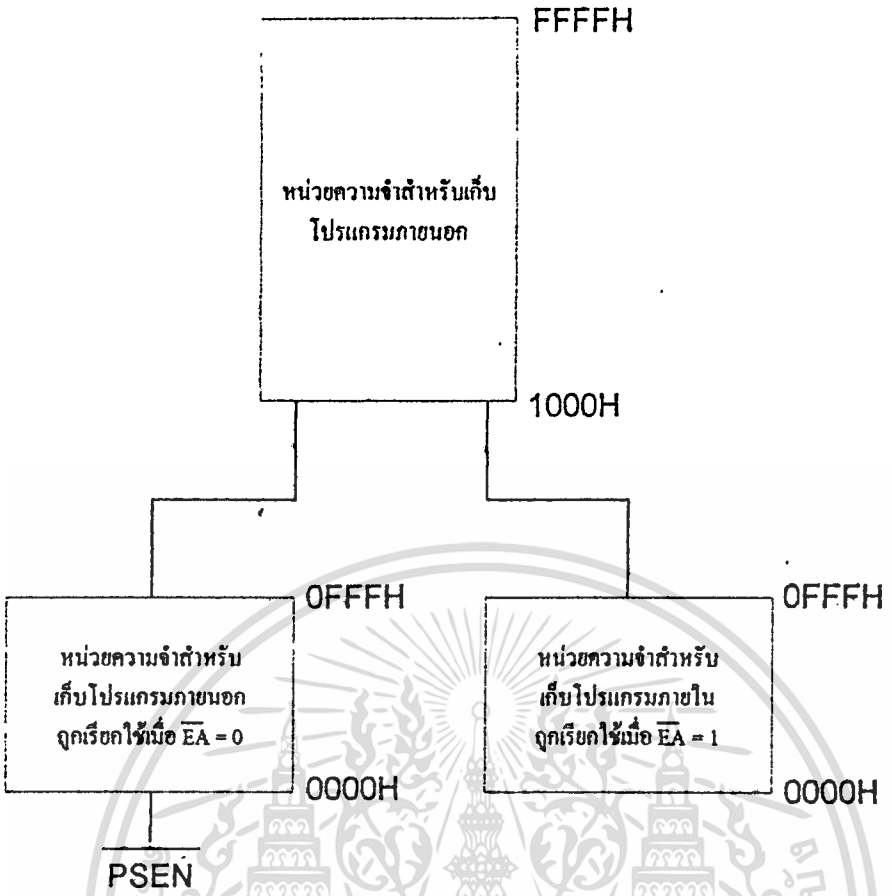
หน่วยความจำโปรแกรมจะใช้สำหรับเก็บโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์ ซึ่งจะเป็นหน่วยความจำแบบแฟลช(flash memory)

สำหรับหน่วยความจำข้อมูลจะใช้สำหรับเก็บข้อมูลหรือค่าตัวแปรต่าง ๆ จากการทำงานของโปรแกรม ซึ่งในเบอร์ 8051 จะมีหน่วยความจำในส่วนนี้เท่ากับ 128 ไบท์ ซึ่งจะเป็นหน่วยความจำแบบ RAM

หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมสามารถแบ่งออกได้เป็นส่วนคือ หน่วยความจำโปรแกรมภายในและหน่วยความจำโปแกรมภายนอก หน่วยความจำโปรแกรมภายในจะถูกเลือกใช้งานถ้าขาสัญญาณ EA มีค่าเป็น 1 โดยจะถูกใช้งานในช่วงแอดเดรส 0-0FFFH (หรือช่วงแอดเดรส 0-1FFFH ในเบอร์8052) นอกเหนือจากช่วงแอดเดรสนี้จะใช้หน่วยความจำโปรแกรมภายนอกทั้งหมด ในกรณีตรงกันข้ามถ้าขาสัญญาณ EA มีค่าเป็น 0 ในช่วงแอดเดรส 0-0FFFH (หรือช่วงแอดเดรส 0-1FFFH ในเบอร์8052) จะถูกใช้จากหน่วยความจำภายนอก หรือกล่าวได้ว่าถ้าขาสัญญาณ EA มีค่าเป็น 0 จะเป็นการเลือกใช้หน่วยความจำโปรแกรมภายนอกทั้งหมดตลอดช่วงแอดเดรส ดังแสดงในรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



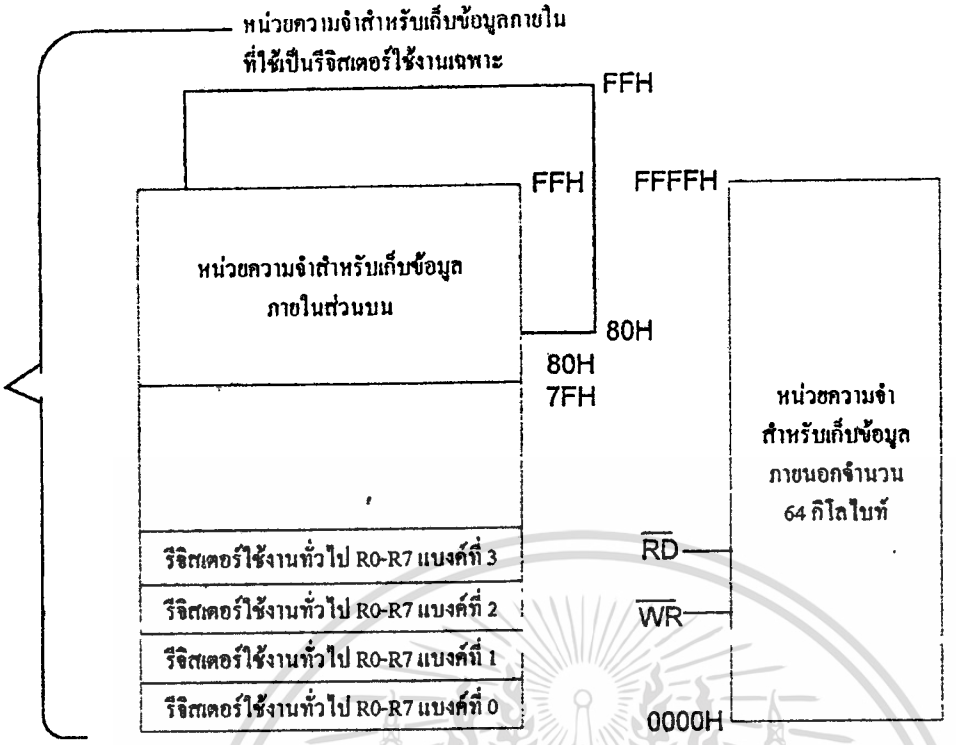
รูปที่ 2.3 หน่วยความจำโปรแกรม

หน่วยความจำข้อมูล

หน่วยความจำข้อมูลสามารถแบ่งออกได้เป็น 2 ส่วน คือ หน่วยความจำข้อมูลภายในและหน่วยความจำข้อมูลภายนอกสำหรับหน่วยความจำข้อมูลภายในยังแบ่งออกได้เป็น 2 ส่วนย่อย คือ ส่วนที่ใช้เก็บข้อมูทั่วไปและส่วนที่ใช้เป็นรีจิสเตอร์หน้าที่พิเศษหรือ SFR (Special Function Register) โดยส่วนที่ใช้เก็บข้อมูลทั่วไปจะถูกใช้สำหรับเก็บข้อมูลหรือค่าตัวแปรต่าง ๆ จากการทำงานของโปรแกรมส่วนรีจิสเตอร์หน้าที่พิเศษจะถูกใช้งานเป็นรีจิสเตอร์ควบคุมการทำงานและบอกสถานะการทำงานของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์เบอร์ 8051 จะมีหน่วยความจำข้อมูลภายในขนาด 128 ไบต์ ดังแสดงในรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 หน่วยความจำสำหรับเก็บข้อมูล

รีจิสเตอร์ใช้งานทั่วไป

รีจิสเตอร์ใช้งานทั่วไปมีไว้สำหรับให้ผู้เขียนโปรแกรมสามารถนำข้อมูลไปพักไว้ชั่วคราว หรือใช้งานทั่วไปได้ตามต้องการ ซึ่งรีจิสเตอร์ใช้งานทั่วไปนี้มีอยู่ด้วยกัน 8 ตัวคือรีจิสเตอร์ R0-R7 โดยรีจิสเตอร์ทั้ง 8 ตัวถูกจัดให้อยู่รวมกันและมีให้เลือกใช้ถึง 4 แบงค์(bank) นั่นคือมีรีจิสเตอร์ใช้งานทั่วไปถึง 32 ตัวให้ใช้งาน เพียงแต่การเลือกใช้รีจิสเตอร์ R0-R7 ในแบงค์ใดแบงค์หนึ่งจะถูกกำหนดจากบิท RSO,RS1 ในรีจิสเตอร์หน้าที่พิเศษ PSW ดังนั้นการเลือกใช้จึงเลือกได้เพียงแบงค์เดียวในขณะใดขณะหนึ่ง อย่างไรก็ตามค่าข้อมูลที่เก็บไว้ในรีจิสเตอร์แบงค์ใดก็ตามที่มีชื่อเดียวกันแต่อยู่คนละแบงค์จะไม่มีผลซึ่งกันและกันเลยทำให้ผู้เขียนโปรแกรมใช้งานรีจิสเตอร์ทั่วไปได้ทั้ง 32 ตัวอย่างเต็มที่และไม่ยุ่งยากในการเขียนโปรแกรม

รีจิสเตอร์หน้าที่พิเศษ (SFR)

รีจิสเตอร์หน้าที่พิเศษมีบทบาทอย่างมากในการควบคุมการงานของไมโครคอนโทรลเลอร์ และทำให้การเขียนโปรแกรมสามารถทำได้สะดวกมากขึ้น รีจิสเตอร์หน้าที่พิเศษทำหน้าที่สำคัญคือควบคุมการทำงานในส่วนต่าง ๆ ภายในไมโครคอนโทรลเลอร์และทำหน้าที่แสดงสถานะการทำงาน ซึ่งในรีจิสเตอร์หน้าที่พิเศษบางตัวยังสามารถเข้าถึงได้ในระดับบิต (bit addressable) ด้วย ดังแสดงในรูปที่ 2.9 การจัดหน่วยความจำและตำแหน่งของรีจิสเตอร์หน้าที่พิเศษต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งแอดเดรส	MSB							LSB	รีจิสเตอร์หน้าที่พิเศษ
	WDT	T32	SERR	IZC	P3HZ	P2HZ	P1HZ	ALF	
0F8H	FF	FE	FD	FC	FB	FA	F9	F8	IOCON
0F0H	F7	F6	F5	F4	F3	F2	F1	F0	B
0E0H	E7	E6	E5	E4	E3	E2	E1	E0	ACC
	CY	AC	F0	RS1	RS0	0V	F1	P	
0D0H	D7	D6	D5	D4	D3	D2	D1	D0	PSW
	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	
0C8H	CF	CE	CD	CC	CB	CA	C9	C8	T2CON
	PCT		PT2	PS	PT1	PX1	PT0	PX0	
0B8H	BF		BD	BC	BB	BA	B9	B8	IP
0B0H	B7	B6	B5	B4	B3	B2	B1	B0	P3
	EA		ET2	ES	ET1	EX1	ET0	EX0	
0A8H	AF		AD	AC	AB	AA	A9	A8	IE
0A0H	A7	A6	A5	A4	A3	A2	A1	A0	P2
	SM0	SM1	SM2	REN	TB8	RB8	T1	R1	
98H	9F	9E	9D	9C	9B	9A	99	98	SCON
90H	97	96	95	94	93	92	91	90	P1
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
88H	8F	8E	8D	8C	8B	8A	89	88	TCON
					DPH	DPL	SP		
80H	87	86	85	84	83	82	81	80	P0

ตารางที่ 2 แสดงการจัดหน่วยความจำและตำแหน่งของรีจิสเตอร์หน้าที่พิเศษ

2.3.2 การอินเตอร์เฟส(Interface)กับจอแสดงผลกราฟฟิกแอลซีดี(Graphic LCD)

ในระบบที่ใช้งานไมโครคอนโทรลเลอร์ทำงานแบบระบบโดดเดี่ยว(Stand Alone Microcontroller Applications) บ่อยครั้งที่จำเป็นต้องติดต่อกับอุปกรณ์แสดงผลด้วย เพื่อแสดงข้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้นำไปดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารนี้ทุกครั้งที่มีการนำไปใช้

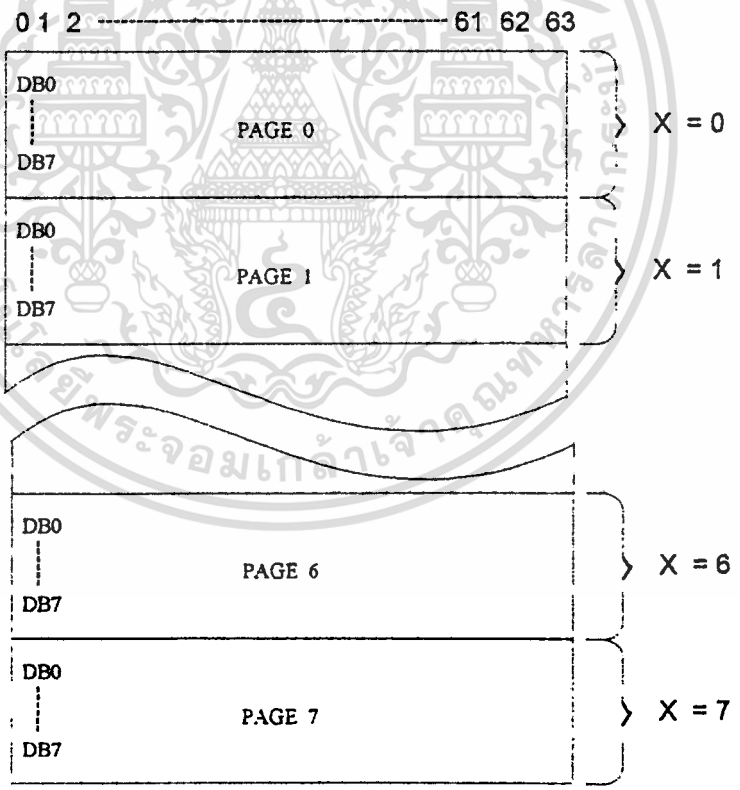
ความ ตัวเลข ผลการวัดหรือข้อมูลต่างๆ ในการสื่อสารกับผู้ใช้งานได้ และอุปกรณ์แบบหนึ่งที่สามารถนำมาใช้งานได้เป็นอย่างดีก็คือจอแสดงผล แบบกราฟฟิกแอลซีดี

DV – 12864 เป็น LCD Graphic ขนาด 128 * 64 Dot ซึ่งมี Controller ภายใน(HD61202, HD61203) โดยการทำงานของController จะมีลักษณะการแบ่งการควบคุมไว้ดังนี้

Iine คือการอ้างถึงบรรทัดของข้อมูลภายในจะแบ่งเป็น 64 แถว (com1-com64)

Page (X- address) เป็นการอ้างถึงหน้าต่างของการแสดงผล ภายในหนึ่ง page ประกอบไปด้วย 8 Line ซึ่งจะเป็นการอ้างถึงข้อมูลด้วย data-bus โดยตรง ภายในของ LCD จะประกอบด้วย 8 Page ซึ่งถูกชี้โดย X – register โดยเมื่อต้องการให้ LCD แสดงผลที่หน้าต่างใดของจอ เราจะต้องตั้งค่า X ให้กับ LCD ซึ่งเมื่อตั้งค่า X ให้กับ LCD แล้ว ค่า X นั้นจะไม่มีการเปลี่ยนแปลงจนกระทั่งจะมีการตั้งค่าใหม่ให้กับ LCD

Segment (y – address) เป็นค่าพอยท์เตอร์ในการชี้ที่อยู่ของข้อมูล ซึ่งภายใน LCD จะถูกควบคุมการชี้ของข้อมูล โดยHD61202 โดยในตัวHD61202 จะสามารถชี้ที่อยู่ของข้อมูลได้ 64 segment. ซึ่ง HD61202 ทั้งสองตัวก็จะสามารถทำการอ้าง segment ได้ถึง 128 segment



รูปที่ 2.5 แสดงการแบ่งการควบคุม

ข้อควรระวัง เมื่อค่า Y ถูกเพิ่มขึ้นมากกว่า 63 แล้ว ค่า Y จะยังไม่เป็นการอ้างถึงข้อมูล segment ที่ 64 (segment 0 ของ cs2) ดังนั้นตัวโปรแกรมจะต้องช่วยจัดการในส่วนนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งควบคุมของ LCD

1. Display ON/OFF

	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	1	1	1	1	1	D

ตารางที่ 3 แสดงคำสั่ง Display ON/OFF

เป็นคำสั่งควบคุมการแสดงผล โดยการแสดงผลจะขึ้นอยู่กับค่า D (DBO) เมื่อค่า D เป็น 1 LCD จะทำการแสดงผล และเมื่อค่า D เป็น 0 LCD จะไม่ทำการแสดงผล ข้อมูลภายใน LCD จะไม่มีการเปลี่ยนแปลงเนื่องจากคำสั่งนี้

2. Display start line

	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	1	1	A	A	A	A	A	A

ตารางที่ 4 แสดงคำสั่ง Display start line

ค่า A จะเป็นค่าหมายเลขบรรทัด ที่จะให้แอลซีดีแสดงผลเป็นบรรทัดแรกของจอภาพ

3. Set Page (X-Address)

	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	1	0	1	1	1	A	A	A

ตารางที่ 5 แสดงคำสั่ง Set Page (X-Address)

ค่า AAA ของคำสั่ง จะเป็นการตั้งค่า เอ็กแอดเดรส(X –Address) ซึ่งหลังจากทำคำสั่งนี้แล้ว ข้อมูลจาก DB0- DB7 จะเป็นการติดต่อกับ แรม(RAM) ที่ PAGE นี้ตลอด จนกว่าจะมีการตั้งค่าใหม่ให้กับแอลซีดี

4. Set Y-Address

	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	0	0	0	1	A	A	A	A	A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6 แสดงคำสั่ง Set Y-Address

ค่า A จะเป็นการตั้งค่าของ Y- Address (ค่า Y จะมีค่าอยู่ระหว่าง 0-63) และค่า Y จะเพิ่มขึ้นครั้งละ 1 เมื่อมีการอ่านหรือเขียนข้อมูลจาก CPU

5. Status Read

	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	1	0	busy	0	on/off	reset	0	0	0	0

ตารางที่ 7 แสดงคำสั่ง Status Read

เป็นการอ่านค่าสถานะของ LCD โดยถ้าค่า Busy เป็น 1 LCD จะทำงานในส่วนภายใน ซึ่งจะทำให้ไม่สามารถทำการควบคุม LCD ในขณะนี้ได้ เพราะฉนั้นเพื่อให้แน่ใจในการควบคุมครั้งต่อไป จะต้องตรวจค่า Busy ให้ได้ค่าเป็น 0 เสียก่อน

6. Write Display Data

	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	0	1	D	D	D	D	D	D	D	D

ตารางที่ 8 แสดงคำสั่ง Write Display Data

การเขียนข้อมูลเข้าไปใน LCD ซึ่งข้อมูล DDDDDDDD จะถูกเก็บใน LCD RAM และค่า Y จะถูกเพิ่มขึ้น 1

7.Read display Data

	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Code	1	1	D	D	D	D	D	D	D	D

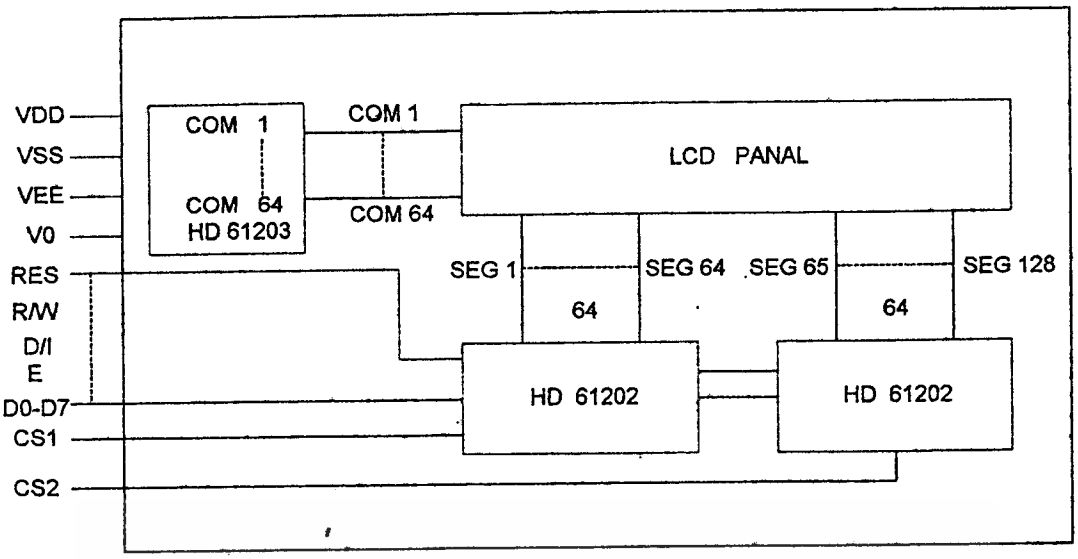
ตารางที่ 9 แสดงคำสั่ง Read Display Data

เป็นการอ่านข้อมูลที่แสดงผล โดย LCD จะให้ค่าของข้อมูลออกมาที่ Data bus ค่า Y จะถูกเพิ่มค่าขึ้น 1 เช่นเดียวกับการเขียนข้อมูล

โครงสร้างภายในของ LCD

จะประกอบด้วยส่วนของ Controller โดย HD61203 จะควบคุมการอ้างถึง Page ของข้อมูล และ HD61202 จะควบคุมในการอ้างของ Segment ซึ่งในการใช้งาน เราจะต้อง Control ส่วนเหล่านี้โดยการส่งระหัสควบคุมไปที่ขาของ LCD ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงโครงสร้างภายใน และขาควบคุม

ขา RST เป็นขาที่ใช้ Reset การทำงานของ LCD

ขา E เป็นขา Enable การรับส่งข้อมูล จะทำงานที่ Logic HIGH และขอบขาลง

ขา R/W เป็นขาที่ใช้กำหนด การอ่านหรือเขียนข้อมูล

ขา D/I ใช้บอกถึงข้อมูลใน Data-bus ว่าเป็นระหัสควบคุม หรือเป็นส่วนของข้อมูล

ขา CS1 Chip Select ของ HD61201 ตัวแรก

ขา CS2 Chip Select ของ HD61201 ตัวที่สอง

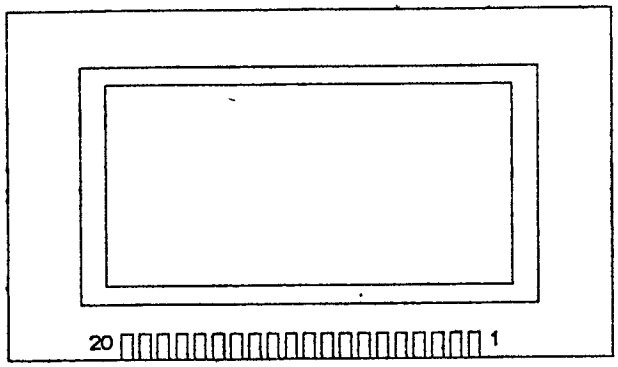
ขา Data-Bus เป็นขาใช้ส่งข้อมูล หรือระหัสควบคุม

หมายเหตุ

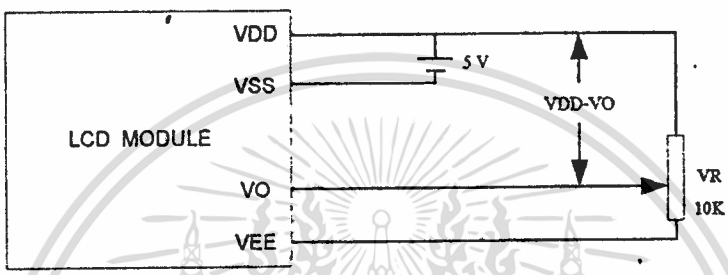
เมื่อCS1 เป็น High และCS2 เป็น Low จะเป็นการอ้างถึง segment ที่ 0-63 และเมื่อ CS1 เป็น Low CS2 เป็น High จะเป็นการอ้างถึง segment ที่ 64 – 127

โครงสร้างภายนอกของกราฟฟิกแอลซีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดงตำแหน่งขาของกราฟฟิกแอลซีดี



รูปที่ 2.8 แสดงขาแหล่งจ่ายไฟ และการต่อใช้งาน

แสดงหน้าที่การใช้งานของแต่ละขา

ขาที่ 1 VSS เป็นขากราวด์(Ground)

ขาที่ 2 VDD เป็นแรงดันไฟเลี้ยงวงจร Logic

ขาที่ 3 VO เป็นแรงดันไฟเลี้ยงแอลซีดี

ขาที่ 4 D/I ถ้ามีสถานะเป็น High จะเป็นส่วนของข้อมูล ถ้าเป็น Low จะเป็นระหัสควบคุม

ขาที่ 5 R/W ถ้ามีสถานะเป็น High จะเป็นการอ่านข้อมูล ถ้ามีสถานะเป็นLow จะเป็นการเขียนข้อมูล

ขาที่ 6 E เป็นขาอีนาเบิล(Enable)ใช้ควบคุมการทำงานของแอลซีดีในการติดต่อกับภายนอก เมื่อมีสถานะเป็น High จะเป็นการบอกให้รับทราบการติดต่อ และเมื่อเปลี่ยนสถานะจาก High เป็น Low จะเป็นการติดต่อกับข้อมูลทาง Data bus

ขาที่ 7-14 DB0-DB7 ใช้ส่งข้อมูลหรือระหัสควบคุม

ขาที่ 15 CS1 เป็นการอ้าง Segment ที่ 0-63 จะทำงานเมื่อมีสถานะเป็น High

ขาที่ 16 CS2 เป็นการอ้าง Segment ที่ 64-127 จะทำงานเมื่อมีสถานะเป็น High

ขาที่ 17 RST เป็นขารีเซ็ต

ขาที่ 18 VEE เป็นขาจ่ายแรงดันไฟลบ

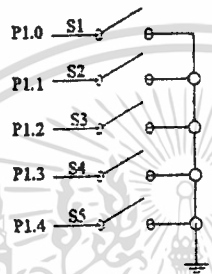
ขาที่ 19 และ20 ไม่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 การอินเตอร์เฟสกับคีย์บอร์ด

โครงงานที่ถูกสร้างขึ้นส่วนมากแล้วจำเป็นต้องติดต่อกับอุปกรณ์อินพุท เช่นตัวเซ็นเซอร์ต่างๆ,สัญญาณไฟฟ้า,คีย์บอร์ด เป็นต้น โดยเฉพาะคีย์บอร์ดที่ใช้ติดต่อกับผู้ใช้งานในการควบคุมเครื่อง ดังนั้นจึงจำเป็นต้องมีในเกือบทุกๆโครงงานที่สร้างขึ้นเพื่อการพัฒนาโครงงานของคุณได้อย่างรวดเร็ว ในที่นี้จะขอกล่าวถึงเทคนิคการอินเตอร์เฟสกับคีย์บอร์ด 4 รูปแบบที่ใช้งานกันอยู่ทั่วไปสำหรับตรวจจับสถานะการเปิดและปิดสวิตช์ในแต่ละรูปแบบเป็นหลักเพื่อให้สามารถมองเห็นแนวทางการเขียนโปรแกรมได้ ซึ่งแต่ละรูปแบบการเชื่อมต่อสวิตช์ก็จะมีข้อดีและข้อเสียแต่ต่างกันไป

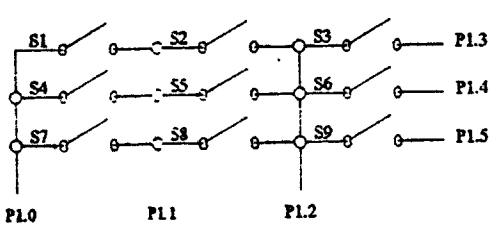
การเชื่อมต่อสวิตช์โดยตรงกับพอร์ท



รูปที่ 2.9 แสดงการเชื่อมต่อสวิตช์โดยตรงกับพอร์ท

รูปที่ 2.9 แสดงการเชื่อมต่อสวิตช์โดยตรงกับพอร์ทซึ่งเป็นแบบที่ง่ายที่สุด การกดสวิตช์แต่ละตัวจะทำให้ขาพอร์ทนั้นถูกต่อลงกราวด์โดยตรง จึงไม่จำเป็นต้องมีวงจรจำกัดกระแสก็ได้ เพราะว่าภายในตัวไมโครคอนโทรลเลอร์มีตัวต้านทานพูลอัพต่ออยู่แล้ว การเขียนโปรแกรมทำได้ง่ายมากโดยใช้คำสั่งตรวจสอบสถานะของแต่ละบิทในพอร์ท เช่นคำสั่ง JB หรือ JNB ข้อเสียของวงจรแบบนี้คือสิ้นเปลืองจำนวนขาพอร์ทจำนวนมาก ถ้าต้องการใช้สวิตช์มากเท่าใดก็ต้องขยายพอร์ทให้มากตามไปด้วย เช่นถ้าต้องการออกแบบเป็นคีย์บอร์ดสำหรับป้อนตัวอักษรขนาด 60 คีย์ ผู้ออกแบบต้องจัดสร้างขาพอร์ทให้ได้ถึง 60 ขาพอร์ท ซึ่งแทบจะเป็นไปได้ยากและสิ้นเปลืองอุปกรณ์มาก

การเชื่อมต่อสวิตช์แบบเมตริกซ์

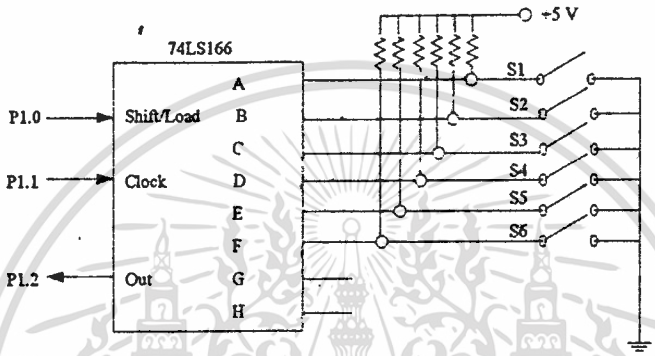


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.10 แสดงการเชื่อมต่อสวิตช์แบบเมตริกซ์

รูปที่ 2.10 แสดงการเชื่อมต่อสวิตช์แบบเมตริกซ์ สวิตช์แต่ละตัวจะถูกเชื่อมต่อกันแบบแถวและคอลัมน์ในรูปแบบเมตริกซ์ การตรวจสอบการกดคีย์ใดบนคีย์บอร์ดทำได้โดยการป้อนค่าตรวจสอบค่าหนึ่งไปยังด้านคอลัมน์และตรวจสอบการเปลี่ยนแปลงที่เกิดขึ้นทางด้านแถว หรือกล่าวได้ว่าตำแหน่งของการกดคีย์ได้จากการเขียนโปรแกรมสแกนคีย์ด้านแถว ข้อดีของการเชื่อมต่อสวิตช์แบบเมตริกซ์คือสามารถใช้สวิตช์ได้มากขึ้นในขณะที่สิ้นเปลืองขาพอร์ทจำนวนน้อยเช่น ใช้ขาพอร์ทเพียง 16 ขาพอร์ทสามารถต่อคีย์บอร์ดได้ถึง 64 คีย์ ข้อเสียของวงจรนี้คือไม่สารารถรับการกดคีย์พร้อมกันได้และต้องเขียนโปรแกรมในการตรวจสอบคีย์ที่ยุ่งยากซับซ้อน

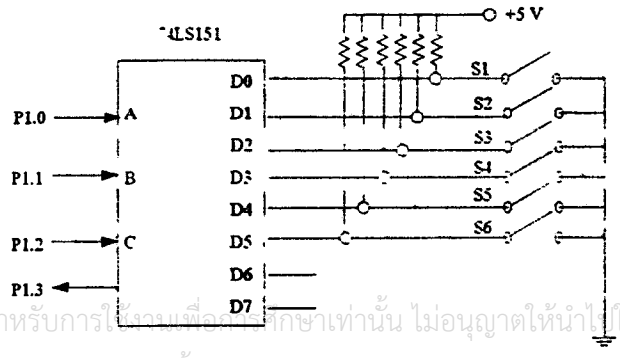
การเชื่อมต่อสวิตช์ผ่านชิฟต์รีจิสเตอร์



รูปที่ 2.11 แสดงการเชื่อมต่อสวิตช์ผ่านชิฟต์รีจิสเตอร์

รูปที่ 2.11 แสดงการเชื่อมต่อสวิตช์ผ่านชิฟต์รีจิสเตอร์จะเห็นได้ว่าวงจรนี้ต้องการขาพอร์ทน้อยมากเพียง 3 ขาพอร์ท โดยใช้ขาพอร์ทหนึ่งกำหนดโหมดที่ขา Shift/Load ให้อยู่ในสภาวะชิฟต์ข้อมูลและให้กำเนิดพัลส์จำนวน 8 พัลส์ เพื่อทำการชิฟต์ข้อมูลสถานะของสวิตช์ผ่านขาพอร์ท P1.2 ไปใช้งาน นั่นคือใช้ขาพอร์ทเพียง 3 ขาพอร์ทเท่านั้นก็ทำงานได้แล้ว รวมทั้งการเขียนโปรแกรมควบคุมก็ทำได้ง่าย ถ้าต้องการเพิ่มจำนวนคีย์มากขึ้นทำได้เพิ่มชิฟต์รีจิสเตอร์มากขึ้นให้เท่าเทียมกัน อย่างไรก็ตามข้อเสียที่เกิดขึ้นคือใช้เวลาในการอ่านสถานะของคีย์ทั้งหมดเป็นเวลานานจนกระทั่งชิฟต์ข้อมูลได้ครบตามจำนวนคีย์

การเชื่อมต่อสวิตช์แบบมัลติเพล็กซ์เซอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

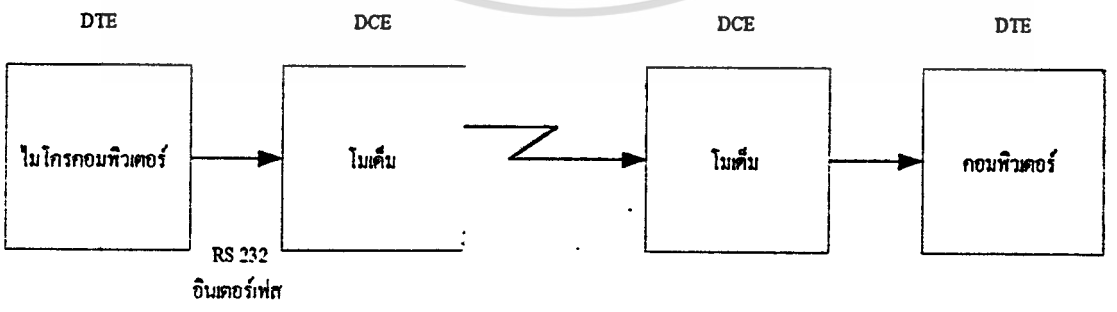
รูปที่ 2.12 แสดงการเชื่อมต่อแบบมัลติเพล็กเซอร์

รูปที่ 2.12 แสดงการเชื่อมต่อแบบมัลติเพล็กเซอร์ ซึ่งต้องการขาพอร์ทจำนวน 4 ขาพอร์ท พอร์ท P1.0, P1.1และ P1.2 ทำหน้าที่ควบคุมไอซีมัลติเพล็กเซอร์เพื่อเลือกสวิตช์ S1-S6 ที่ต้องการจะติดต่อด้วย สถานะของคีย์ที่ถูกเลือกจะถูกส่งกลับไปยังไมโครคอนโทรลเลอร์ผ่านขาพอร์ท P1.3 วงจรแบบนี้สามารถเพิ่มจำนวนคีย์ได้ง่ายและทำได้เป็นจำนวนมาก อีกทั้งการเขียนโปรแกรมก็ทำได้ง่ายเช่นเดียวกัน

2.3.4 พอร์ทสื่อสารอนุกรม

พอร์ทสื่อสารเป็นส่วนสำคัญส่วนหนึ่งที่มีอยู่บนไมโครคอมพิวเตอร์ 16 บิท พอร์ทสื่อสารนี้มีชื่อเรียกอีกอย่างว่า คอม-พอร์ท(COM PORT) ผู้ออกแบบพอร์ทสื่อสารต้องการให้เป็นไปตามมาตรฐานการเชื่อมต่อแบบอนุกรมที่เรียกว่า RS 232C พอร์ทสื่อสารนี้เป็นทางออกของข้อมูลที่ผู้ใช้สามารถส่งหรือรับกับระบบอื่นได้ พอร์ทสื่อสารจึงเป็นพอร์ทที่จำเป็นและผู้จัดซื้อไมโครคอมพิวเตอร์มักจะกำหนดอยู่ในสเปกด้วยเสมอ

มาตรฐาน RS 232C ได้จัดพิมพ์ขึ้นเมื่อ ปี ค.ศ. 1969 โดยสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิคส์แห่งสหรัฐอเมริกา RS ย่อมาจาก Recommened Standard ส่วน 232 เป็นหมายเลขบ่งบอกของมาตรฐานตัวนี้ C เป็นหมายเลขของฉบับท้ายสุดของมาตรฐานตัวนี้ จุดประสงค์ของมาตรฐานตัวนี้ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งข้อมูลปลายทาง (Data terminal Equipment DTE) กับอุปกรณ์สื่อสารข้อมูล (Data Communication Equipment DCE) สำหรับผู้ใช้ไมโครคอมพิวเตอร์ DTE ก็หมายถึงตัวไมโครคอมพิวเตอร์และ DCE ก็หมายถึงโมเด็ม อุปกรณ์ต่างๆ เช่น เครื่องพิมพ์ที่รับสัญญาณแบบอนุกรมอาจจะเป็นได้ทั้ง DTE,และ DCE ขึ้นอยู่กับผู้ผลิต ข้อแตกต่างของ DTE และ DCE จะเห็นได้จาก รูปที่ 2.13 จากรูปนี้เราจะเห็นได้ว่า RS 232C มีส่วนสำคัญอย่างใหญ่หลวงสำหรับการสื่อสารข้อมูลระหว่างไมโครคอมพิวเตอร์



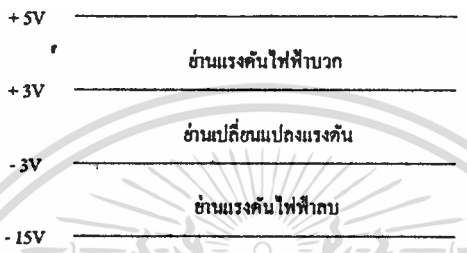
รูปที่ 2.13 การใช้ RS 232C เชื่อมต่ออุปกรณ์

ความเร็วและระยะทางของการเชื่อมต่อ RS 232C สามารถเชื่อมต่อการถ่ายโอนข้อมูลได้จาก 0 – 20,000 บิตต่อวินาที ซึ่งเพียงพอสำหรับไมโครคอมพิวเตอร์ที่มีขนาดอัตราบอด 110 ถึง

9600 บอด ความยาวของสายเชื่อมต่อโดยสัญญาณมาตรฐานของ RS 232C จำกัดอยู่แค่ 50 ฟุต ซึ่งเพียงพอสำหรับการสื่อสารไมโครคอมพิวเตอร์กับอุปกรณ์รอบนอก

ลักษณะของสัญญาณ RS 232C

เพื่อเป็นหลักประกันว่าข้อมูลถูกส่งออกไปอย่างถูกต้อง และอุปกรณ์ถูกควบคุมอย่างถูกต้อง จำเป็นต้องมีข้อตกลงกันในเรื่องของสัญญาณที่ใช้ มาตรฐาน RS 232C กำหนดย่านของแรงดันไฟฟ้าในสัญญาณเพื่อสนองจุดประสงค์ข้างบน ดังแสดงในรูปที่ 2.14



รูปที่ 2.14 ย่านของแรงดันไฟฟ้าที่ใช้ในสัญญาณ RS 232C

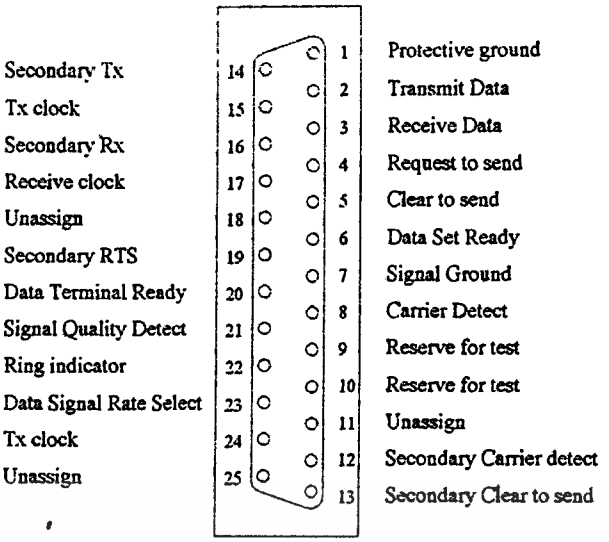
สำหรับไมโครคอมพิวเตอร์บางเครื่อง ใช้แต่สัญญาณลอจิกออกมาเป็นสัญญาณของ RS 232C เลย อย่างเช่น อะซิงโครนัสอะแดปเตอร์ของ IBM PC ในกรณีเช่นนี้ระยะทางของสายที่เชื่อมต่ออาจจะไปได้สั้นกว่า 50 ฟุต ดังที่กล่าวเอาไว้เนื่องจากระดับของกราวนด์เปลี่ยนแปลงไป อันเนื่องจากการสูญเสียไปในความต้านทานของสาย ผู้ที่เคยใช้ IBM PC อาจจะเคยประสบกับปัญหานี้มาแล้วว่า เอ๊ะทำไมต่อสัญญาณ RS 232C เกินกว่า 10 ฟุต แล้วใช้งานไม่ได้ แต่อย่างไรก็ตาม RS 232 ของ IBM PC ยังมีโอกาสให้เลือกใช้ 20 มิลลิแอมแปร์ กระแสวนกลับทนแรงดันไฟฟ้า

การกำหนดขาของขั้วต่อ RS 232C

ในทางฟิสิกส์แล้ว มาตรฐานของ RS 232C กำหนดข้อต่อแบบ DB-25 แต่ละขาของข้อต่อกำหนดไว้ดังในรูปที่ 2.15 อย่างไรก็ตามผู้ผลิตไมโครคอมพิวเตอร์อาจจะใช้ข้อต่อชนิดอื่นที่นอกเหนือจาก DB-25 ยกตัวอย่างเช่น Fujitsu F-8,IBM AT, IBM Jr เป็นต้น ตัวเมียของข้อต่อควรจะอยู่ที่ตัวโมเด็ม ขณะที่ตัวผู้ควรจะอยู่ที่ asynchronous communication adapter หรือที่ตัวไมโครคอมพิวเตอร์เอง อย่างไรก็ตามผู้ผลิตหลายรายไม่ได้ทำตามกฎเกณฑ์ที่ว่านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟ
39420
ว31
040572



รูปที่ 2.15 แสดงการกำหนดขาของขั้วต่อ RS232

หน้าที่การทำงานของขาสัญญาณแต่ละขา

Transmit Data (TD ขาที่ 2)

เป็นสัญญาณที่ส่งออกจาก DTE (หรือตัวไมโครคอมพิวเตอร์) ไปยังโมเด็มหรือต่อเข้าโดยตรงกับตัวไมโครคอมพิวเตอร์ตัวอื่น หรือเครื่องพิมพ์ เมื่อไม่มีสัญญาณส่งออกสถานภาพของลอจิกที่ขานี้จะมีค่าเท่ากับ "1" หรือเทียบเท่ากับสต๊อปบิต

Reccive Data (RD ขาที่ 3)

เป็นทางของสัญญาณเข้าไปยัง DTE หรือไมโครคอมพิวเตอร์ เมื่อไม่มีสัญญาณรับเข้ามาขานี้จะมีสถานภาพทางลอจิก เป็น "1"

Request To Send (RTS ขาที่ 4)

ใช้สำหรับส่งสัญญาณไปยังโมเด็มหรือเครื่องพิมพ์เป็นการเรียกร้องที่จะส่งสัญญาณทางขา 2 สัญญาณนี้ใช้คู่กับ CTS หรือ Clear to send อุปกรณ์รับหากได้รับสัญญาณ RTS จะตรวจสอบตัวเองว่าพร้อมจะตรวจตัวเองว่าพร้อมจะรับสัญญาณได้หรือยัง หากพร้อมที่จะรับก็ส่งสัญญาณออกไปที่สาย CTS

Clear t To Send (CTS ขาที่ 5)

ดังอธิบายไว้ใน RTS เมื่อสัญญาณนี้อยู่ในสถานออฟ (negative voltage หรือลอจิก "1") หมายความว่า อุปกรณ์รับกำลังบอกว่าพร้อมที่จะรับข้อมูลแล้ว

Data Set Ready (DSR ขาที่ 6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อสัญญาณสายนี้อยู่ในสถานออน (หรือลอจิก 0) เป็นการบอกไมโครคอมพิวเตอร์หรือฝ่ายส่งว่า โมเด็มต่อเข้ากับสายโทรศัพท์เรียบร้อยแล้วและพร้อมที่จะส่งได้แล้ว โมเด็มที่มีการหมุนหมายเลขอัตโนมัติจะส่งสัญญาณสายนี้ไปบอกให้คอมพิวเตอร์รู้ว่าต่อโทรศัพท์ได้สำเร็จแล้ว

Signal Ground (SG ขาที่ 7)

SG ทำหน้าที่เป็นระดับแรงดันอ้างอิงสำหรับทุกๆ สายของสัญญาณ จะมีแรงดันเป็น "0" เมื่อเทียบกับสัญญาณตัวอื่น

Carrier Detect (CD ขาที่ 8)

โมเด็มจะส่งสัญญาณที่อยู่ในสถานออน (ลอจิก "0") ไปบอกไมโครคอมพิวเตอร์ เมื่อได้รับสัญญาณจากโมเด็มของอีกฝ่ายหนึ่ง สัญญาณนี้จะนำไปจุด LED บอกว่าได้รับสัญญาณจากโมเด็มอีกฝ่ายหนึ่งแล้ว ไฟ LED จะอยู่บนหน้าปัดของโมเด็มเอง

Data Terminal Ready (DTR ขาที่ 20)

คอมพิวเตอร์เปิดสัญญาณสายนี้ให้ออน (ลอจิก "0") เมื่อพร้อมที่จะติดต่อกับโมเด็ม โมเด็มส่วนมากจะไม่รายงานสถานภาพของตัวเอง (CD,DSR และ CTS) ให้คอมพิวเตอร์รู้ หากคอมพิวเตอร์ไม่เปิดสัญญาณ DTS

Ring Indicator (RI ขาที่22)

สัญญาณนี้ใช้ในโมเด็มที่เป็นระบบตอบได้อัตโนมัติ (Auto-answer) สัญญาณนี้จะออนเมื่อมีสัญญาณกระดิ่งมา และออฟระหว่างเสียงดังของกระดิ่ง

จุดอ่อนของ EIA RS 232C พอสรุปได้ 3 ประการ

1. ใช้ระดับแรงดันไฟเลี้ยง –15 โวลต์ นอกเหนือ –5 โวลต์ ซึ่งใช้ในวงจรลอจิก

2.ค่าตัวเก็บประจุของอุปกรณ์รับสัญญาณ RS 232C รวมทั้งตัวเก็บประจุสเตย์(Stay capacitance) ในสายจะต้องไม่มากกว่า 2500 pf สายที่รวมๆ กันหลายสาย ส่วนมากจะมีตัวเก็บประจุสเตย์ประมาณ 40-50 pf ต่อ 1 ฟุต ดังนั้นสายนี้จะต่อได้ยาวสุด 50 ฟุต ก่อนที่ค่าตัวเก็บประจุสเตย์จะมากกว่า 2500 pf ถ้าหากตัวเก็บประจุสเตย์มากกว่าที่กำหนดนี้ ช่วงเวลาการเปลี่ยนแปลงระดับของสัญญาณจะมากกว่า 4 เปอร์เซ็นต์ ตามที่ยอมให้ได้ในมาตรฐาน RS 232C เมื่อเป็นเช่นนี้จะทำให้ฝ่ายรับตีความสัญญาณผิดไปจากความจริง มาร์กบิต (mark bit) จะยาวกว่าสเปซบิต (space bit) หรือ สเปซบิตยาวกว่ามาร์กบิต ขึ้นอยู่กับวงจรการตรวจสอบการผิดเพี้ยนแบบนี้เรียกว่า " Bias distrotion"

3.ปัญหาของสัญญาณไฟฟ้า ที่ EIA ไม่แก้เอาไว้ สำหรับวงจรที่ใช้ ไอซี ก็คือปัญหาเรื่องกราวนด์ที่แตกต่างกันตามมาตรฐาน EIA สัญญาณที่ส่งออกเทียบกับกราวนด์ของเครื่องส่งเท่านั้น ถ้าหากเครื่องรับกับเครื่องส่งมีระดับแรงดันกราวนด์แตกต่างกัน สมมติว่า 2 โวลต์ กระแสที่ไหลในเส้นที่เป็นกราวนด์ก็จะเกิดขึ้น สมมติความต้านทานของสายเป็น 0 ความต่างศักย์ที่เกิดจากกระแสกราวนด์ก็จะไม่มี ความต่างศักย์ของกราวนด์ระหว่างเครื่องรับและเครื่องส่งก็จะคงเท่าเดิม

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับของสัญญาณที่ฝ่ายส่งและฝ่ายรับมองเห็นก็จะแตกต่างกัน สมมติว่าระดับของแรงดันกราวนด์ต่างกัน 2 โวลต์ ฝ่ายส่งใส่แรงดันเข้าไป 5 โวลต์ ฝ่ายรับจะมองเห็นแค่ 3 โวลต์เท่านั้น ในทางกลับกันถ้าฝ่ายส่งใส่แรงดัน –5 โวลต์ ฝ่ายรับจะมองเห็นเป็น –7 โวลต์

ความต่างศักย์ของกราวนด์จะคงที่ 2 โวลต์ ไม่ว่าฝ่ายส่งจะใส่แรงดันเข้าไปเท่าไหร่ก็ตาม ผลของกราวด์ที่แตกต่างกันนี้อาจจะเกิดจากสถานีรับและสถานีส่งมีระบบไฟฟ้าที่กราวนด์แตกต่างกันก็ได้

2.3.5 โปรโตคอล(POTOCAL)

โปรโตคอล คือชุดของกฎที่มีไว้เพื่อจัดการกับอุปกรณ์ที่ร่วมกันใช้ช่องสัญญาณให้ทำตามกระบวนการสื่อสารอย่างมีลำดับ โปรโตคอลใช้กับการส่งถ่ายข้อมูลในอุปกรณ์ชนิดโหมดบล็อก ซึ่งปรกติจะมีอุปกรณ์ปลายทางต่ออยู่ในระบบหลายจุด(multipoint) มีอยู่ส่วนน้อยของโปรโตคอลที่ใช้กับอุปกรณ์อะซิงโครนัส(โหมดอักษร) ในระบบจุดต่อจุด เพราะมันไม่ต้องการการควบคุมที่ซับซ้อนแต่อย่างไร

หน้าที่หลักของโปรโตคอล

- สร้างกรอบข่าวสาร(framing) หน้าที่คือการแบ่งข่าวสารออกเป็นบล็อกๆซึ่งประกอบไปด้วย ส่วนเนื้อหา(text) และส่วนควบคุม มีการกำหนดขนาดของบล็อกที่ยาวที่สุด การแบ่งข่าวสารจะต้องไม่เกินข้อจำกัดเหล่านี้ ส่วนที่เพิ่มเติมมาคือ ข้อมูลควบคุม เช่น ฟิลด์ที่อยู่

- การควบคุมความผิดพลาด มีหน้าที่คือ การตรวจจับความผิดพลาด เช่น การใช้ CRC 16 บิท และการแก้ไขเมื่อมีการตรวจพบความผิดพลาด เช่น การร้องขอให้ส่งสัญญาณข่าวสารกลับมาใหม่ ซึ่งรายละเอียดจะเกี่ยวข้องกันอย่างมากกับประสิทธิภาพของการส่งข้อมูลนั้นๆ

- การควบคุมสาย คือ ชุดกฎเกณฑ์ ที่ใช้กำหนดว่าอุปกรณ์ตัวใดบนช่องการสื่อสารจะได้รับความยินยอมให้ทำการติดต่อสื่อสารได้ในเวลาที่จะกำหนดให้อย่างไร ประโยชน์ของการควบคุมสายจะดูไม่ค่อยมีความหมายในการส่งข้อมูลระบบจุดต่อจุด แต่จะมีคุณค่าเด่นชัดในระบบต่อหลายจุด

- การควบคุมการไหล เป็นการใช้คลื่นพาหะรองในระบบดูเพล็กซ์เป็นตัวช่วยทำการควบคุม ในระบบล็อกซิงโครนัสนั้นมีการควบคุมการไหลที่แตกต่างไปบ้าง แต่จัดให้มีขึ้นเพื่อให้บรรลุวัตถุประสงค์และหน้าที่อันเดียวกันกับที่กล่าวมาแล้ว

- การควบคุมลำดับ คล้ายกับการควบคุมความผิดพลาด แต่มันอยู่ในระดับที่สูงกว่า กฎการควบคุมลำดับจะรับรองไม่ให้เกิดการหาย หรือการซ้ำซ้อนของข้อมูลที่ไปถึงสถานีปลายทาง ทั้งยังไม่ให้เกิดการสลับลำดับของข้อมูลด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรโตคอลมีอยู่หลายชนิดด้วยกัน เช่น โปรโตคอล BISYNC ,HDLC ,HOST LINK และMODBUS แต่ในที่นี้จะของกล่าวถึง มอดบัสโปรโตคอล(MODBUS POTOCAL) และโฮสลิงค์โปรโตคอล(HOST LINK POTOCAL)

มอดบัสโปโตคอล

มอดบัสโปโตคอล จะมีคุณสมบัติเฉพาะตัวบางอย่าง เช่น รูปแบบกรอบข่าวสาร ลำดับกรอบข่าวสาร, communication errors , exception condition และ function ต่าง ๆ

คุณสมบัติบางอย่างสามารถเลือกได้ เช่น การเลือก transmission meduim , bord ratd , character parity , number of stop bits และ transmission mode (ASCII หรือ RTU) ซึ่งผู้ใช้เลือกพารามิเตอร์ โดยเซ็ตที่แต่ละ station ซึ่งพารามิเตอร์เหล่านี้ไม่สามารถเปลี่ยนได้ ขณะที่ระบบกำลังทำงาน

สำหรับฮาร์ดแวร์ที่ส่ง message บน data line Message ต้องอยู่ใน envelope ซึ่ง envelope จะออกจากฮาร์ดแวร์ผ่านพอร์ต และถูกส่งจาก line ถึง address device ในกรณีนี้ Modbus Protocol จัดให้ envelope อยู่ในรูปของ message frame

ข้อมูลใน message เป็นแอดเดรสของตัวรับ สิ่งที่ตัวรับต้องทำคือ data ที่จะใช้ และการเช็ค errors

เมื่อ Message เข้าถึง Modbus slave มันจะเข้าไปที่ address device ผ่านทาง port เดียวกัน address device จะเคลื่อนย้าย envelope อ่าน message ถ้าไม่มี errors เกิดขึ้น จะมีการตอบรับเกิดขึ้น มันจะวาง message ไว้ใน envelope และจะเปลี่ยนเป็นตัวส่ง ข้อมูลใน message ที่ตอบสนองเป็นแอดเดรสของ slave data ที่ได้มาจะเป็นผลของการกระทำและเป็นการเช็ค errors จะไม่มีการตอบสนองถ้า message เป็น broadcast ที่แสดงโดย address 0

อุปกรณ์ตัวรับจะเขียนและส่ง response ของตัวมันเอง ถ้า message ที่ผิดถูกรับเข้าไป โดยส่วนใหญ่ master สามารถส่ง message อื่น ๆ ถึงหลาย slave ได้เร็วเท่าที่มันรับตอบสนอง หรือหลังจากผู้ใช้เลือกช่วงเวลาถ้าไม่มีผลการตอบสนอง

message ทั้งหมดอาจจะส่งเป็น query , ให้ slave ตอบสนอง สำหรับ message ที่ไม่ต้องการตอบสนอง อาจจะส่งเป็น broadcast message , message เหล่านี้คือ modify coil status , modefy rigister conterts force multiple coil , preset multiple registers

Mode of Transmission

Modes of Transmission (โหมดของการส่งผ่าน) เป็นโครงสร้างของลักษณะเฉพาะของข้อมูลภายในMassage และ เป็นระบบตัวเลขที่ใช้ส่ง Data ซึ่งมีอยู่ 2 Mode คือ ASCII (American Standart Code for Interchange)และ RTU (Remote Terminal Unit) โหมดที่เลือกใช้ขึ้นอยู่กับ equiment ที่ใช้เป็น Modbus master ซึ่งทั้งสองโหมดนี้ ถ้าใช้ต้องใช้โหมดใดโหมดหนึ่ง จะใช้ร่วมกันไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Characteristic	ASCII (7-bit)	RTU (8-BIT)
Coding system	hexadecimal (uses ASCII Printable characters:0-9,A-F)	8-bit binary
Number of bits per character		
start bits	1	1
data bits (least significant first)parity (optional)	7 1 (1 bit sent for even or odd parity,no bit for no parity)	8 1 (1-bit sentfor even or odd parity,no bits no parity)
stop bits	1 or 2	1 or 2
Error checking	LRC (Longitudinal Redundancy Check)	CRC (Cyclical Redundancy Check)

ตารางที่ 10 แสดงให้เห็นถึง characteristic ของ ASCII และ RTU

ตัวอักขระ ASCII ง่ายในการมองเห็นเมื่อต้องแก้ไข ซึ่งจะเหมาะกับ ภาษาคอมพิวเตอร์ระดับสูง เช่น Fortran

ในโหมด RTU, data ถูกส่งเป็นอักขระ (character) เลขฐานสอง 8 บิท ในโหมด ASCII แต่ละ RTU character ถูกแบ่งเป็น 4 บิท สองส่วน และจะเปลี่ยนเป็นเลขฐานสิบหก จะเห็นได้ว่า ASCII จะใช้ตัวอักขระเพียง 2 ตัว และในการถอดระหัส (decoding) และจัดการ (handling), ASCII จะง่ายกว่า นอกจากนี้ ใน RTU อักขระของ Message ต้องใช้การส่งผ่านต่อเนื่องกันไป ในแต่ละโหมด ASCII สามารถที่จะหยุดได้ถึง 1 วินาที ระหว่างตัวอักขระ

การตรวจสอบความผิดพลาด (ERROR DETECTION)

จะมี error 2 แบบที่อาจจะเกิดขึ้นได้ใน Communication System

- Transmission error
- Programming error or Operation error

ความผิดพลาดจากการสื่อสาร (Communications error) ทำให้บิทเปลี่ยนแปลงภายในMassage เช่น 0001,0100 กลายเป็น 0001,0110 สาเหตุที่เกิดขึ้นบ่อยเนื่องจาก สัญญาณรบกวนจากสัญญาณไฟฟ้าที่ไม่ต้องการในช่วงสื่อสาร สัญญาณไฟฟ้าจากเครื่องจักร ฯลฯ ซึ่งจะตรวจสอบโดย character framing, a parity check redundancy check เมื่อ characterframing, parity check หรือ

redundancy check ตรวจสอบการ communication error, process ของ message จะหยุด pc slave จะไม่การทำหรือตอบสนองต่อ message

เมื่อเกิด communication error ขึ้น pc slave จะไม่แน่ใจว่า message นี้ติดต่อกับมัน ดังนั้น CPU อาจจะเป็นการตอบรับ message ที่ไม่ใช่message ของมันดังที่ตอนเริ่มต้น มันเป็นสิ่งสำคัญต่อโปรแกรม mod bus mastet จะสมมติให้ communication error เกิดขึ้นถ้าไม่มีการตอบสนองในเวลาที่เหมาสม ช่วงความยาวของเวลานี้ขึ้นอยู่กับ baud rate, ชนิดของ message และ scan time ของ pc slave เวลาช่วงนี้จะถูกกำหนดครั้งหนึ่ง ที่ master อาจถูกโปรแกรมตอนเริ่มติดต่อกับ message โดยอัตโนมัติ

ในโหมด RTU จะมีข้อมูล 9 บิท (data 8 bit และparity bit) ส่วนในโหมด ASCII จะมีข้อมูล 8 bit(data 7 bit และ parity bit) ถ้าไม่ใช้ parity ก็ไม่ต้องส่ง Parity bit ออกไป parity มีรูปแบบให้หลายรูปแบบ สำหรับ Modbus system, Interface Units มีโครงสร้างแบบไม่มี parity ใช้ even parity หรือ odd parity ในเวลาที่มีการติดตั้ง โดยรูปแบบโครงสร้างทั้งหมดของ Unit ในระบบจะเป็นแบบเดียวกัน Modbus system จะกำหนดว่า parity bit ควรจะเป็น 1 หรือ 0

สำหรับ Even Parity

- ถ้ามีเลข 1 เป็นจำนวน คู่ เติม 0 ที่ Parity bit
- ถ้ามีเลข 1 เป็นจำนวน คี่ เติม 1 ที่ Parity bit เพื่อให้ 1 เป็นคู่

สำหรับ Odd Parity

- เพิ่ม 1 หรือ 0 ที่ Parity bit เพื่อให้จำนวนเลข 1 ใน data เป็นเลขคี่

Modbus system มีหลายระดับของการตรวจสอบความผิดพลาด เพื่อให้มั่นใจในคุณภาพของการส่ง data ในการตรวจสอบ Error หลาย ๆ บิท ที่ Parity ไม่เปลี่ยนแปลง เป็นระบบที่ใช้Redundancy checks คือ CRC (Cyclical Redundancy Check) และ LRC (Longitudinal Redundancy Check) โดยปกติ RTU นิยมใช้CRL และ ASCII นิยมใช้ LRC

CRC 16 (Cyclic Redundacy Check)

Error Check Sequence Message (เฉพาะ Data bits เท่านั้น ไม่เกี่ยวกับบิท start/stop และ parity บิท) ถูกพิจารณาเป็นเลขฐานสองที่ต่อเนื่องกัน MSB ของมันจะถูกส่งออกไปเป็นอันดับแรก message จะถูกคูณด้วย X^{16} (เลื่อนไป 16 บิท) จากนั้นหารด้วย $X^{16}+X^{15}+X^2+1$ แสดงเป็นเลขฐานสอง (11000000000000101) Digit ผลหารที่เป็นจำนวนเต็มเราจะไม่สนใจ และเศษที่เหลือ 16 บิท นำมาบวกเข้ากับ message (MSB fitst) เป็นสอง CRC check bytes ผลลัพธ์ message จะรวมกับ CRC เมื่อหารด้วย polynomial คือ ($X^{16}-X^{15}+X^2+1$) ที่ตัวรับ จะให้ตัวเศษเป็น 0 ถ้าไม่เกิด error ขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเขียนขั้นตอนรูปแบบ CRC-16 ได้ดังนี้

1. ทำ Register 16 bit เป็น 1 ทั้งหมด
2. Exclusive OR ที่ byte 8 biteแรกกับ byte ลำดับสูงของรีจีสเตอร์ 16 บิท เก็บผลในรีจีเตอร์ 16 บิท
3. เลื่อนรีจีสเตอร์ 16 บิท ไปทางขวา 1 บิท
4a. ถ้า bit ที่ถูกเลื่อนไปเป็น 1 Exclusive OR กับ polynomial 1010,0000,0000,0001 กับรีจีสเตอร์ 16 บิท
4b ถ้าบิทที่ถูกเลื่อนไปเป็น 0 กลับไปทำตาม stepที่ 3
5. ทำซ้ำ step 3 และ 4 จนกระทั่งได้เลื่อนไป 8 ครั้ง
6. Exclusive OR byte 8 bite ต่อไปกับรีจีสเตอร์ 16 bite
7. ทำซ้ำ step 3 ถึง step 6 จนกระทั่งทุกไบท์ของ Message ได้ exclusive OR กับ รีจีสเตอร์ 16 บิท และเลื่อนไป 8 ครั้ง
8. Register 16 bit จะมีการเช็ค error 2 byte และถูกเพิ่ม message MSB เป็นอันดับแรก

LRC (Longitudinal Redundancy Check)

Error Check สำหรับโหมด ASCII คือ LRC โดย Error Check เป็น 8 bit ในเลขฐาน สอง และถูกส่งแทนASCII 2 หลัก ในฐานสิบหก และบวกกับเลขฐานสองที่ไม่มี Wraparound Carry จากนั้นทำผลลัพธ์เป็น 2 Complement ที่ตัวรับ LRC จะถูกคำนวณใหม่ และเปรียบเทียบกับ LRC ที่ส่งออกมา ดังรูปตารางข้างล่างนี้

Address	0	2		0000	0010
Function	0	1		0000	0001
Start Add H.O	0	0		0000	0000
Start Add L.O	0	0		0000	0000
Quantity of Pts	0	0		0000	0000
	0	8		+0000	1000
				0000	1011
			1's Complement	1111	0100
			+1		1
			2's Complement	1111	0101
				F	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Error Check	F	5	F 5
เมื่อตัวรับ รับค่ามาแล้วจะนำค่าที่ได้มาบวกกัน			0000 0010
ถ้าผลลัพธ์ที่ได้เป็นศูนย์แสดงว่าข้อมูลที่ได้รับมา			0000 0001
ไม่มี Error โดยปกติผลลัพธ์ที่ได้จะมีค่าเกิน			0000 0000
8 Bit แต่ในที่นี้ จะสนใจเฉพาะค่า 8 bit ทาง			0000 0000
ด้าน Byte ต่ำเท่านั้น			0000 0000
		O.K	0000 1000
		Error Check	1111 0101
		Sum	0000 0000

ตารางที่ 11 แสดงการคำนวณ Error Check

ASCII Framing

Framing ในโหมด ASCII transmission จะมีการใช้ colon (:) เป็นตัวเริ่มของเฟรม และ carriage return (CR) , line feed (LF) เป็นปลายสุดของเฟรม

LF ใช้เป็น char ที่จะแสดงว่า trasmitting station พร้อมที่จะได้รับการตอบสนองทันที ตามตารางที่ 12

BEG OF FRAME	ADDRESS	FUNCTION	DATA	ERROR CHECK	EOF	READY TO RECRESP
:	2-CHAR 16-BITS	2-CHAR 16-BITS	N * 4-CHAR N * 16-BITS	2-CHAR 16-BITS	CR	LF

ตาราง 12 ASCII MESSAGE FRAME FORMAT

Remote Terminal Unit (RTU) Framing

เฟรม synchronization สามารถที่จะใช้ในโหมด RTU เท่านั้นโดยการจำลอง synchronors message อุปกรณ์ตัวรับจะติดตามเวลาที่ผ่านพ้นไประหว่างการได้รับ char ถึงเวลา 3.5 char ผ่านไปโดยไม่มี char ตัวใหม่ หรือการสิ้นสุดของเฟรม ดังนั้น device จะคิดว่า byte ต่อไปที่รับมาจะเป็น address ตามตารางที่ 13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T1 T2 T3	ADDRESS	FUNCTION	DATA	CHECK	T1 T2 T3
	8-BITS	8-BITS	N*8-BITS	16-BITS	

ตารางที่ 13 RTU MESSAGE FRAME FORMAT

Address Field

address field จะอยู่ถัดจากตัวเริ่มแรกของเฟรมและประกอบด้วย 8 บิท (RTU) หรือ 2 char (ASCII) บิทเหล่านี้จะแสดงให้ผู้ใช้กำหนด address ของ slave device ที่ต้องรับ message ที่ส่งโดย Master

แต่ละ Slave จะต้องกำหนด address slave เท่านั้น ที่จะสนองต่อ query ที่เก็บ address ของมัน เมื่อ slave ส่งการตอบสนองaddress ของ slave จะบอกให้ master รู้ว่า slave ไหน ที่กำลังติดต่อ ใน broadcast message แอสเดรส 0 จะถูกใช้ slave ทั้งหมดที่ได้อธิบายนี้ เป็นการอ่านและเป็นการกระทำบน massage แต่ไม่มี message ตอบสนอง

Function Field

รหัส Function จะบอก address slave ว่า ต้องปฏิบัติหน้าที่อะไร ระหัสของ Modbus Function ถูกออกแบบให้มีลักษณะเฉพาะ สำหรับทำงานร่วมกับ PC บนระบบการสื่อสารของ Mosbus ตามตารางที่ 1-2 จะแสดงระหัสฟังก์ชันต่าง ๆ ความหมาย และการกระทำเริ่มแรก

บิทลำดับสูงใน field นี้ ถูกเซ็ทโดย slave device แสดงที่อื่น ๆ สัมพันธ์กับการตอบสนองธรรมดาที่กำลังติดต่อกับ Master device บิทนี้ยังคงเป็น 0 ถ้า message เป็น query หรือ message ตอบสนองธรรมดา

CODE	MEANING	ACTION
01	Read Coil Status	ได้รับสถานะปัจจุบัน (on/off) ของกลุ่มของ logic coils
02	Read Input Status	ได้รับสถานะปัจจุบัน (on/off) ของกลุ่มของ discret input
03	Read Holding Registers	ได้รับค่า binary ปัจจุบันในหนึ่ง holding registers หรือ มากกว่านี้
04	Read Input Registers	ได้รับค่า binary ปัจจุบัน ในหนึ่ง input register หรือมากกว่านี้
05	Force Single	ให้ logic coil อยู่ในสภาวะ on หรือ off

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CODE	MEANING	ACTION
06	Preset Single Register	ใส่ค่า binary พิเศษ ใน holding register
07	Read Exception Status	ได้รับสถานะ (on/off) ของ coil ทั้ง8 ของaddress เป็นตัวควบคุมไม่อิสระผู้ใช้ logic สามารถโปรแกรม coil เหล่านี้ ให้แสดงสถานะของ slave message สั้นๆ จะมีความรวดเร็วในการอ่านสถานะ
08	Loopback Diagnostic Test	Message ทดสอบการวินิจฉัย ส่งไปที่ slave หาค่า TEST กระบวนการสื่อสาร
09	Program(เฉพาะ 484)	ให้ master เลียนแบบการกระทำของ programing panal และเปลี่ยนแปลงลอจิกของ pc slave
10	Poll Program Complete(เฉพาะ 484)	ยอมให้ master ติดต่อกับ slave ตัวอื่นได้ ถ้า slave ตัวนั้นกำลังทำงานกับโปรแกรมที่ยาว slave จะถูกตรวจสอบตามช่วงเวลา ถ้าการทำงานของมันเสร็จสมบูรณ์หลังจาก code 9 ได้ส่งออกไปแล้ว
11	Fetch Event Counter Communication	ยอมให้ master ส่ง single query และต่อมากำหนดว่าการทำงานจะเสร็จหรือไม่ โดยเฉพาะเมื่อการสื่อสารError บนคำสั่งหรือการตอบสนอง
12	Fetch Communications Event Log	ยอมให้ master ทำให้การสื่อสารกลับคืนมา ที่เก็บข้อมูลและติดต่อแต่ละข้อมูลกับ slave นั้น ถ้าการติดต่อไม่สมบูรณ์ log จะให้ข้อมูลการ error เกิดขึ้น
13	Program(เฉพาะ 184/384,484,584)	ถ้า master จำลองการกระทำของ programing panal และการเปลี่ยนแปลงลอจิกของ pc slave

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CODE	MEANING	ACTION
14	Poll Program Complete(เฉพาะ 184/384,484,584)	ยอมให้ master ติดต่อสื่อสารกับ slave อื่นๆ ถ้า slave ตัวนั้นกำลังทำงานบนโปรแกรมที่ยาว slave ถูกสำรวจตามกำหนดเวลาจะเห็นได้ถ้าโปรแกรมทำงานเสร็จสมบูรณ์ซึ่งจะส่งออกไปหลังจากได้เก็บรหัสฟังก์ชัน13ที่ได้ส่งมาแล้วเท่านั้น
15	Force Multiple Coils	ทำให้อนุกรมของ logic coilsที่ต่อเนื่องถูกกำหนดสถานะ on หรือ off
16	Preset Mulitple Registers	ใส่ค่า binary พิเศษ ในอนุกรมของ holding ที่ต่อเนื่องกัน
17	Report Slave I.D	ให้ master กำหนดชนิดของ address slave และสถานะ หลอดไฟการทำงานของslave
18	Program (เฉพาะ 884,Micro 84)	ให้ master จำลองการกระทำของ programming panal และเปลี่ยนลอจิกสภาวะ pc
19	Reset Communications Link	รีเซ็ท slave ให้รู้สถานะหลังจาก error ไม่สามารถกลับคืนสถานะเดิมได้ รีเซ็ท byte ตามลำดับ
20	Read General Rreference(เฉพาะ 584L)	การแสดงข้อมูลเก็บในไฟล์ memory ที่ขยายออกไป
21	Write General Reference (เฉพาะ 584L)	เข้าไปดูหรือเปลี่ยนข้อมูลที่เก็บในไฟล์ memory ที่ขยายออก
22-64	Reserved For Expanded Functions	
65-72	Reserved For User Functions	สงวนไว้เพื่อผู้ใช้สำหรับฟังก์ชันนี้ ที่จะไม่ถูกใช้โดย Gorld ในผลิตภัณฑ์ในอนาคต
73-119	Illegal Functions	
120-127	Reserved	สงวนไว้สำหรับใช้ภายใน
128-255	Reserved	สงวนไว้ exception responses

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 14 Modbus Function Codes

Data Field

data field จะเก็บข้อมูลที่ต้องการโดย slave ทำตัวเป็นฟังก์พิเศษ หรือมันจะเก็บ data สะสมไว้โดย slave ในการตอบสนองต่อ query ข้อมูลนี้อาจจะมี values, address reference หรือ limits เช่น ตัวอย่าง function code จะบอกว่า slave จะอ่านที่ holding register และ data field จำเป็นต้องแสดงว่า จะเริ่มที่ register ไหน และจะอ่านจำนวนเท่าไหร่ แอดเดรสและข้อมูลจะเปลี่ยนแปลงตามชนิดและความสามารถของ pc ที่สัมพันธ์กับ slave

Error checking

Field นี้ จะให้ master และ slave เช็ค message สำหรับ errors ในการ transmission บางครั้งจะมีสัญญาณรบกวนทางไฟฟ้าหรือสัญญาณรบกวนอื่นทำให้มั่นใจว่า master หรือ slave ไม่ได้ทำให้ message นั้นเปลี่ยนไประหว่าง transmission

ในการเพิ่มความปลอดภัยและประสิทธิภาพของ Modbus system error check field ใช้ LRC ในโหมด ASCII และใช้ CRC-16 ในโหมด RTU

รูปแบบของ MODBUS FUNCTIONS

จุดประสงค์ของเรื่องนี้ต้องการกำหนดรูปแบบโดยทั่วไปสำหรับคำสั่งเฉพาะซึ่งจะเป็นประโยชน์ต่อผู้เขียนโปรแกรมของระบบมอดบัส

รูปแบบแต่ละ query message เป็นตัวอย่างในโหมดการส่งผ่าน ASCII และอธิบายถึงหน้าที่ของ query message ที่จัดให้ รวมทั้ง response message โดยทั่วไป

Message ใน function code 1-6, 15 , และ 16 แสดงให้เห็นโดยเฉพาะที่ตำแหน่งใน programable controller ที่ต้องเข้าไป ฟังก์ชัน 1,5 และ 15 อ้างอิงถึง coil (oxxx(x)) ,code 2 อ้างถึง input (1xxx(x)) ,code 4 อ้างถึง input register (3xxx(x)) และ code 3 ,6 และ 16 อ้างถึง holding register (4xxx(x)) แอดเดรสทั้งหมดที่อ้างอิงใน modbus message มีจำนวนสัมพันธ์กับ 0 ตัวอย่าง เช่น holding register ตัวแรก ใน 584 จะเป็นรีจิสเตอร์ 40001 และจะอ้างอิงเป็น 0000 เช่นเดียวกัน ใน coil 00127 จะเป็น 0126 (decimal) = 007E (hex) ตัวเลขทั้งหมดในรูปแบบ modbus จะใช้เป็นเลขฐาน 16

ตัวอย่างที่ให้นี้ จะเป็น Read Output Register 4108 4110 สำหรับ Modbus Slave ทำงานที่ address 06 ซึ่งค่าตัวเลขในตารางจะอยู่ในรูปฐาน 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDR	FUNC	DATA START REG HO	DATA START REG LO	DATA # OF REGS HO	DATA # OF REGS LO	ERROR CHECK FIELD	
06	03	00	6E	00	03	89	LRC

ตารางที่ 15 แสดงตัวอย่างการอ่านค่าเอาท์พุทรีจิสเตอร์

Errors ในการโปรแกรมหรือ การทำงาน ทำให้ data ใน message ผิดเพี้ยนไปไม่ถูกตามรูปแบบ จะทำให้ไม่มีการตอบสนองจาก pc ให้มันอินเตอร์เฟสหรือยุ่งยากในการสื่อสารกับ slave Error เหล่านี้จะมีผลใน Exception Response จาก Solfware ของ Master (Modbus Communication Handler) หรือ PC Slave ขึ้นอยู่กับประเภทของ Error

ระหัสของ Exeption Response แสดงตามตารางที่ 16 เมื่อ PC Slave ตรวจพบ Error เหล่านี้มันจะส่ง message ที่ตอบสนองกลับไปที่ master ซึ่งประกอบด้วย slave address, function code , error code และ error check field จะแสดงที่ผลการตอบรับจะแจ้งให้ทราบถึงการ error, บิทลำดับสูงของ function code จะถูกเซ็ทเป็น 1

CODE	NAME	MEANING
01	Illegal Function	Message function ที่รับมาไม่ยอมกระทำกับ address slave ถ้าคำสั่งที่ส่งออกมาไม่มีฟังก์ชันที่จะทำงานกับมัน
02	Illegal Data Address	แอดเดรสที่อ้างใน data field ไม่ใช่แอดเดรสสำหรับตำแหน่งของ address slave
03	Illegal Data Value	ค่าที่อ้างอิงใน data field ไม่ได้
04	Failure In Associated Device	PC ของ slave ล้มเหลวในการตอบสนองต่อ messageหรือ error เกิดขึ้น (ดู note 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CODE	NAME	MEANING
05	Acknowledge	Slave PC ได้ยอมรับและกำลังโปรเซสร์ คำสั่งโปรแกรมยาว ๆ จะส่ง Poll Program Complete ออกมาเมื่อการโปรเซสเสร็จ Poll message จะส่งไป pc ก่อนที่มันจะเสร็จ จะเป็นผลให้การตอบสนอง message ถูกปฏิเสธ(ดู note 2)
06	Busy Rejected Message	message ถูกรับโดยไม่มี error แต่ pc กำลังโปรเซสคำสั่งโปรแกรมเวลานาน ต้องติดต่อใหม่หลังจากนั้นเมื่อ pc ว่าง (ดู note 2)
07	Nak-Negative Acknowledgment	ฟังก์ชันโปรแกรม ไม่สามารถทำงานได้ การส่ง poll ได้รับรายละเอียด Device ตามข้อมูลที่ error
08	Memory Parity Error	การอ่าน Modbus ของ memory ที่เพิ่มขึ้นและตรวจสอบ บิท memory ที่กำลังเข้าไป error อาจจะไม่เกิดขึ้น ถ้าการทดสอบล้มเหลว

ตารางที่ 16 Exception Response Codes

Note 1

สำหรับ Function Code 1-19 ,Excpt code 4 อาจแสดงเพียงส่วนของ qrery ที่ถูกโปรเซสก่อนที่ error ที่แก้ไม่ได้เกิดขึ้นที่สถานีตอบสนองการรับ Excpt code 4 ต้องการการแจ้งให้ทราบทันที

Note 2

Excpt code 5 และ 6 ใช้สำหรับรุ่น 884 เท่านั้น และ device ขึ้นอยู่กับการตอบสนองที่ error ต่อ function code 18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

QUERY MESSAGE

QUERY นี้ต้องการติดต่อกับอินพุท 1245 ใน slave ตัวที่ 10 ถ้าตัวคอนโทรลเลอร์นี้เป็น 1K-484 ค่าตัวเลขอินพุทจะไม่ถูกต้อง

SLAVE ADDR	FUNC	H.O. START ADDR	L.O. START ADDR	H.O. NO.CF COILS	L.O. NO.OF COILS	ERROR CHECK FIELD	
0A	01	04	A1	00	01	4F	LRC

ตารางที่ 17 แสดงการอ่านค่าสถานะของคอยล์(Coil)

Function code field จะถูก set เป็น 1 ที่บิทลำดับสูง และได้ EXCTP 1 CODE 2 ซึ่งจะเป็น illegal data address

SLAVE ADDR	FUNC	EXCEPTION CODE	ERROR CHECK	
0A	81	02	73	LCR

ตารางที่ 18 Read coil Status Error Response

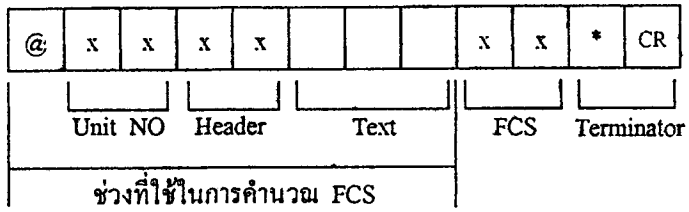
โฮสลิงค์โปรโตคอล (Host Link Potocal)

โฮสลิงค์โปรโตคอลที่ใช้อยู่ในโรงงานอัตโนมัติจะมีวัตถุประสงค์เพื่อทำการสื่อสารข้อมูลระยะไกล เฝ้ามองสถานะของระบบในสายการผลิต (Status Monitoring) หรือแม้แต่การบังคับสถานะให้เป็นไปตามต้องการ (Status Forceing) ทำการเชื่อมโยงสถานะกันในแต่ละส่วนของการผลิต (Status Link) ตรวจสองวินิจฉัยตลอดจนแก้ไขระบบแบบระยะไกล (Remoted Diagnostics) เพื่อให้เกิดประสิทธิภาพในการควบคุมแบบกระจาย และยังเป็นศูนย์รวมของข้อมูล เมื่อข้อมูลต่างๆ ถูกนำมาวิเคราะห์อย่างเหมาะสมถูกต้องแล้ว ผลที่ตามมาก็คือต้นทุนการผลิตจะลดลงอันเป็นวัตถุประสงค์ของระบบ

รูปแบบกรอบข่าวสาร

ในแต่ละบล็อคจะประกอบไปด้วยอักขระที่ใช้ในการสื่อสารแต่ละอักขระจะเป็น ASCII จัดเรียงไว้ดังที่แสดงในรูปที่ 2.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 แสดงรูปแบบกรอบข่าวสารของโฮสลิงค์โปรโตคอล

@ เป็นอักขระเริ่มต้นของบล็อค

Unit NO ใช้อ้างอิงกับโฮสลิงค์ตัวที่เท่าไร โดยมีค่าตั้งแต่ 00-31

Header เป็นรหัสคำสั่งให้ปฏิบัติ

Text เป็นข้อความหรือข้อมูลประกอบคำสั่งที่ตามมาจาก Header

FCS เป็นรหัสกำกับความถูกต้องของข้อมูลในกรอบข่าวสารนี้

*CR เป็นรหัสปิดท้ายในแต่ละกรอบข่าวสาร

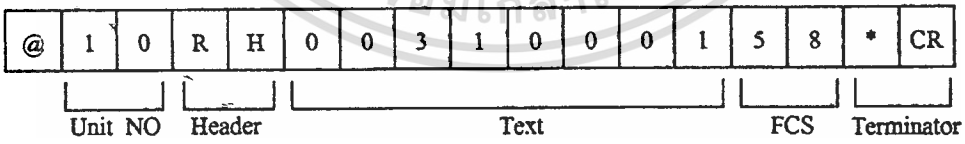
ถ้าความยาวของกรอบข่าวสารมีมากคือมากกว่า 131 อักขระอาจจะต้องมีมากกว่า 1 กรอบข่าวสาร

การคำนวณหาค่า FCS

FCS(Frame Checksum) เป็นข้อมูลขนาด 8 บิท และถูกแปลงเป็นแอสกีจำนวน 2 อักขระ ข้อมูลขนาด 8 บิทนี้ได้จากผลลัพธ์ของการ Exclusive OR ตามลำดับเรื่อยมาตั้งแต่อักขระเริ่มต้นไปจนถึงอักขระสุดท้ายของ Text ในแต่ละกรอบข่าวสารมีการคำนวณดังนี้

ตัวอย่างการคำนวณ FCS

กำหนดให้มีรูปแบบในการส่งดังต่อไปนี้



รูปที่ 2.17 แสดงตัวอย่างรูปแบบที่ต้องการส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	รหัสแอสกี	BCD	BCD	
@	40	0100	0000	
1	31	0011	0001	
		0111	0001	XOR
0	30	0011	0000	
		0100	0001	XOR
R	52	0101	0010	
		0001	0011	XOR
H	48	0100	1000	
		0101	1011	XOR
0	30	0011	0000	
		0110	1011	XOR
0	30	0011	0000	
		0101	1011	XOR
3	33	0011	0011	
		0110	1000	XOR
1	31	0011	0001	
		0101	1001	XOR
0	30	0011	0000	
		0110	1001	XOR
0	30	0011	0000	
		0101	1001	XOR
0	30	0011	0000	
		0110	1001	XOR
1	31	0011	0001	
		0101	1000	XOR
FCS		5	8	

ตารางที่ 19 แสดงการคำนวณค่า FCS

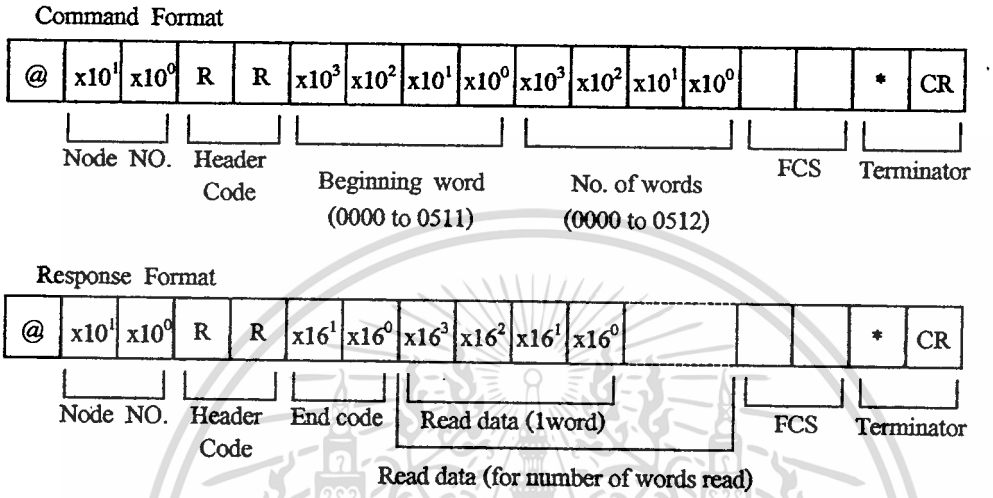
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า FCS ที่คำนวณได้ในตัวอย่างนี้มีค่าเท่ากับ 58 ต้องทำการแปลงเป็นรหัสแอสกีเสียก่อน ที่จะทำการส่งโดยเลข 5 จะมีรหัสแอสกีเท่ากับ 35 และเลข 8 จะมีรหัสแอสกีเท่ากับ 38

คำสั่งที่ใช้ในการติดต่อกับพีแอลซีของ omron

1. IR/SR AREA READ (RR)

อ่านค่าสภาวะของ IR(Internal Relay) และ SR(Special Relay) โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 **ถึง** 0511 **และมีจำนวนข้อมูลที่ต้องได้ถึง** 512 **ข้อมูลจากการกำหนดที่** NO. of words

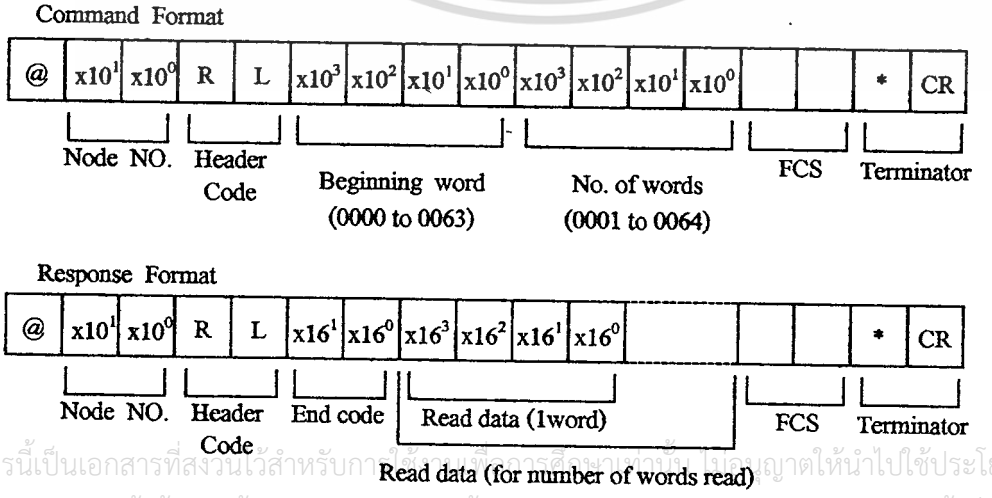


รูปที่ 2.18 แสดงคำสั่งอ่านค่าจาก Internal Relay หรือ Specail Relay

รูปแบบของการตอบสนองจากพีแอลซี(Response Format**) จะถูกส่งกลับมาในรูปแบบของเลขฐานสิบหก ข้อมูลจะถูกส่งกลับมาและข้อมูลตัวแรกจะเป็นไปตามที่กำหนดไว้ใน** Beginning word

2. LR AREA READ (RL)

อ่านค่าสภาวะของ RL(Link Relay) โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 **ถึง** 0063 **และมีจำนวนข้อมูลที่ต้องการ จาก** 0001 **ถึง** 0064 **ข้อมูลจากการกำหนดที่** NO. of words



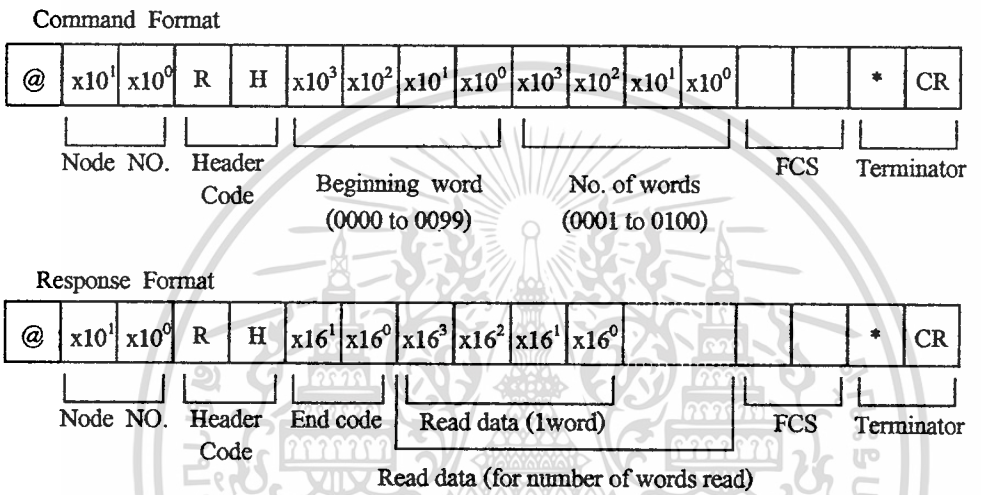
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.19 แสดงคำสั่งอ่านค่าจาก Link Relay

รูปแบบของการตอบสนองจากพีแอลซี(Response Format) จะถูกส่งกลับมาในรูปแบบของเลขฐานสิบหก ข้อมูลจะถูกส่งกลับมาและข้อมูลตัวแรกจะเป็นไปตามที่กำหนดไว้ใน Beginning word

3. HR AREA READ (RH)

อ่านค่าสภาวะของ RH(Holding Relay) โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 ถึง 0099 และมีจำนวนข้อมูลที่ต้องการจาก 0001 ถึง 0100 ข้อมูลจากการกำหนดที่ NO. of words



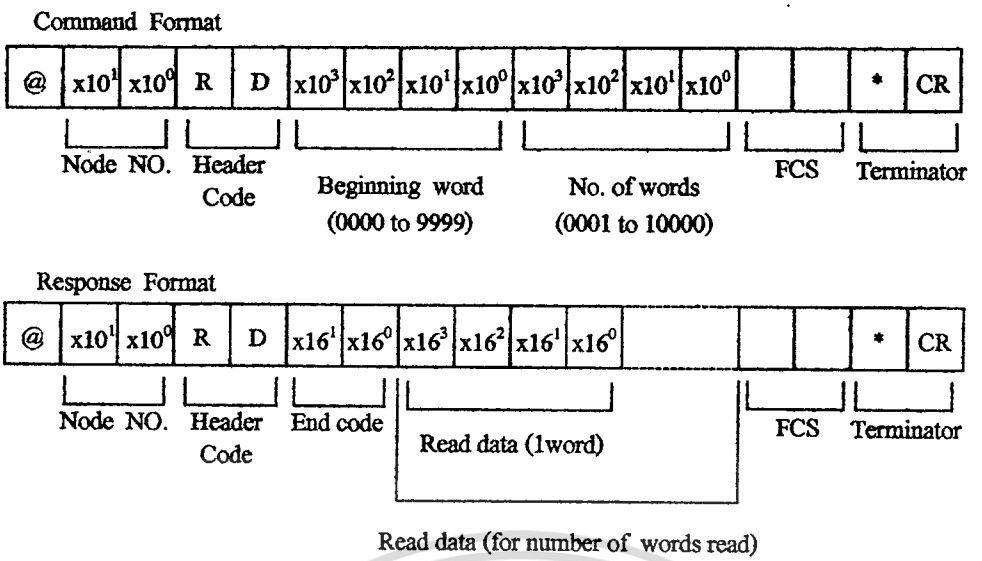
รูปที่ 2.20 แสดงคำสั่งอ่านค่าจาก Holding Relay

รูปแบบของการตอบสนองจากพีแอลซี(Response Format) จะถูกส่งกลับมาในรูปแบบของเลขฐานสิบหก ข้อมูลจะถูกส่งกลับมาและข้อมูลตัวแรกจะเป็นไปตามที่กำหนดไว้ใน Beginning word

4. DM AREA READ (RD)

อ่านค่าข้อมูลที่เก็บอยู่ใน Data Memory โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 ถึง 9999 และมีจำนวนข้อมูลที่ต้องการจาก 0001 ถึง 10000 ข้อมูลจากการกำหนดที่ NO. of words

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

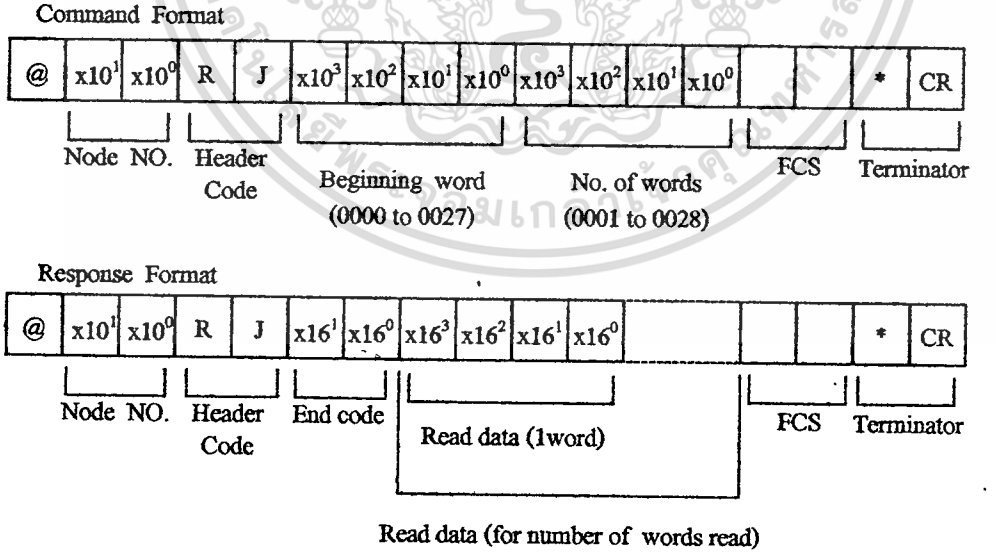


รูปที่ 2.21 แสดงคำสั่งอ่านค่าจาก Data Memory

รูปแบบของการตอบสนองจากพีแอลซี(Response Format) จะถูกส่งกลับมาในรูปแบบของเลขฐานสิบหก ข้อมูลจะถูกส่งกลับมาและข้อมูลตัวแรกจะเป็นไปตามที่กำหนดไว้ใน Beginning word

5. AR AREA READ (RJ)

อ่านค่าสภาวะของ Auxiliary Relay โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 ถึง 0027 และมีจำนวนข้อมูลที่ต้องได้จาก 0001 ถึง 0028 ข้อมูลจากการกำหนดที่ NO. of words



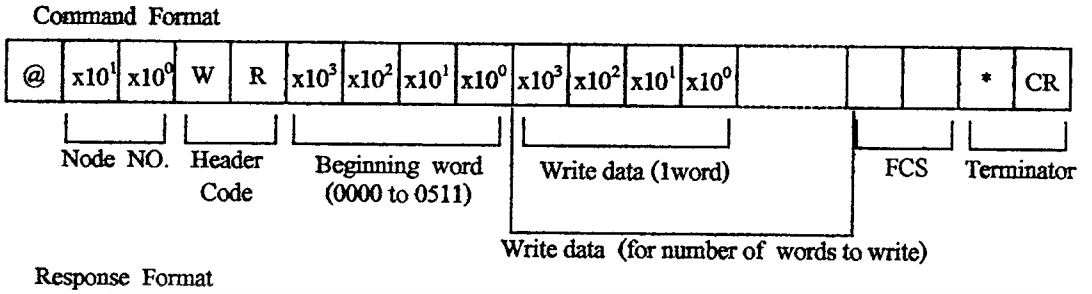
รูปที่ 2.22 แสดงคำสั่งอ่านค่าจาก Auxiliary Relay

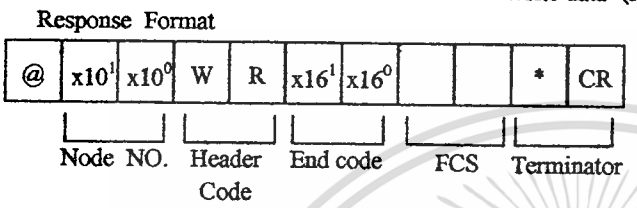
รูปแบบของการตอบสนองจากพีแอลซี(Response Format) จะถูกส่งกลับมาในรูปแบบของเลขฐานสิบหก ข้อมูลจะถูกส่งกลับมาและข้อมูลตัวแรกจะเป็นไปตามที่กำหนดไว้ใน Beginning word

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. IR/SR AREA WRITE (WR)

เขียนข้อมูลที่ Internal Relay หรือ Special Relay โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 ถึง 0063 ซึ่งจะเขียนข้อมูลเป็นแบบ Word



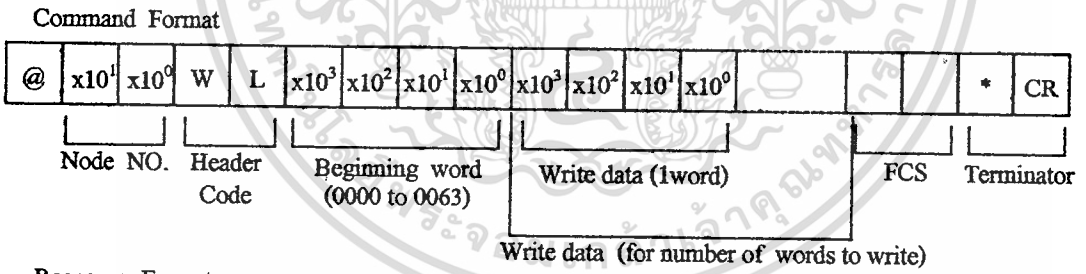


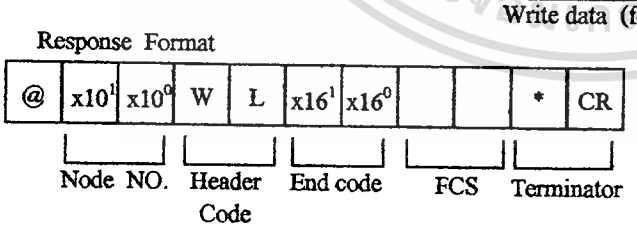
รูปที่ 2.23 แสดงการเขียนข้อมูลที่ Internal Relay หรือ Special Relay

รูปแบบการตอบสนองจากพีแอลซีที่ได้จากการเขียนข้อมูลที่ IR หรือ SR ตามจำนวนที่กำหนดและเริ่มต้นการเขียนเป็นเลขฐานสิบหกตามการกำหนดการเริ่มต้น

7. LR AREA WRITE (WL)

เขียนข้อมูลที่ Link Relay โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 ถึง 0063 ซึ่งจะเขียนข้อมูลเป็นแบบ Word





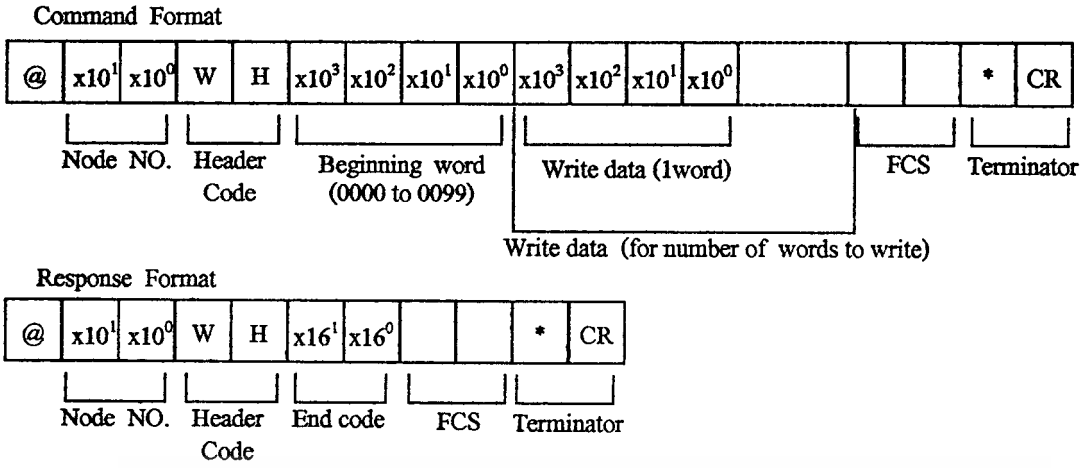
รูปที่ 2.24 แสดงการเขียนข้อมูลที่ Link Relay

รูปแบบการตอบสนองจากพีแอลซีเขียนข้อมูลที่ Link Relay ตามจำนวนที่กำหนดและเริ่มต้นการเขียนเป็นเลขฐานสิบหกตามการกำหนดการเริ่มต้น

8. HR AREA WRITE (WH)

เขียนข้อมูลที่ Holding Relay โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 ถึง 0099 ซึ่งจะเขียนข้อมูลเป็นแบบ Word

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

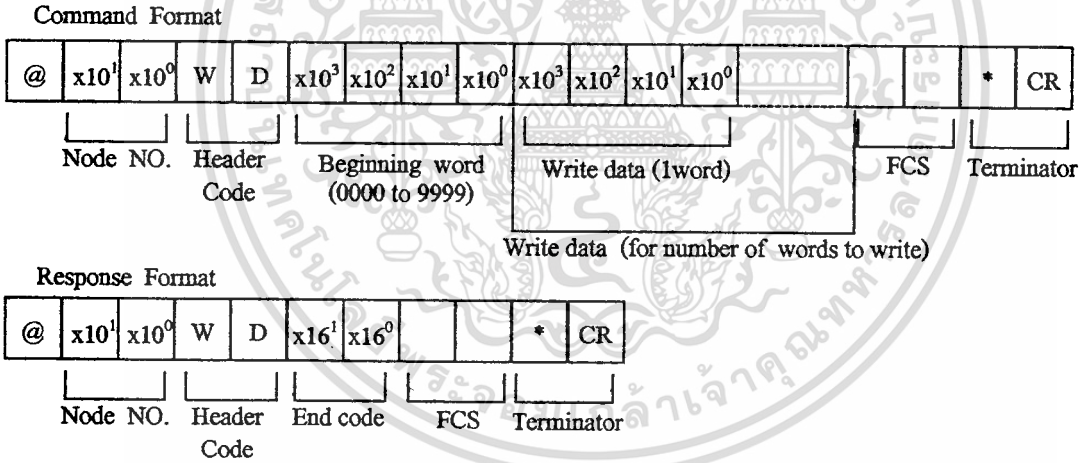


รูปที่ 2.25 แสดงการเขียนข้อมูลที่ Holding Relay

รูปแบบการตอบสนองจากพีแอลซีเขียนข้อมูลที่ Holding Relay ตามจำนวนที่กำหนดและเริ่มต้นการเขียนเป็นเลขฐานสิบหกตามการกำหนดการเริ่มต้น

9. DM AREA WRITE (WD)

รูปแบบการตอบสนองจากพีแอลซีเขียนข้อมูลที่ Data Memory โดยมีตำแหน่งเริ่มต้นซึ่งสามารถกำหนดได้ที่ Beginning word 0000 ถึง 9999 ซึ่งจะเขียนข้อมูลเป็นแบบ Word



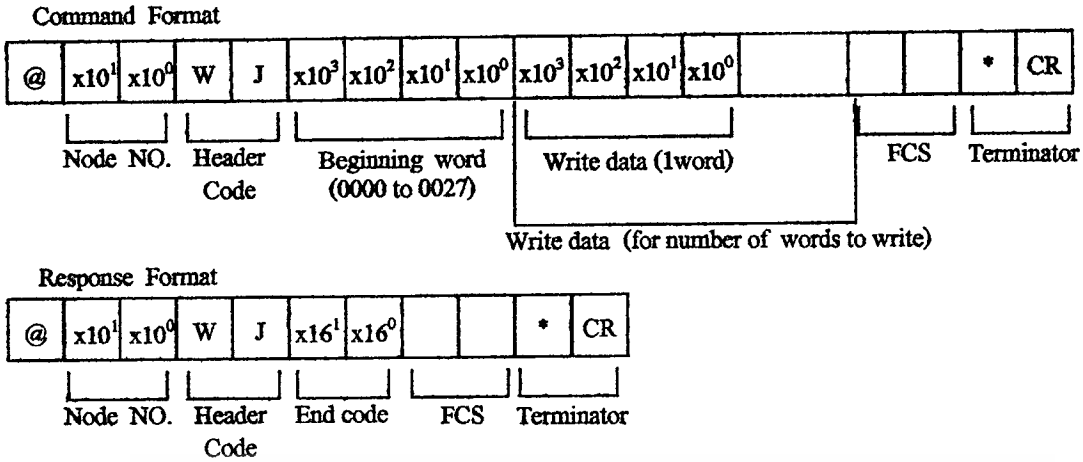
รูปที่ 2.26 แสดงการเขียนข้อมูลที่ Data memory

รูปแบบการตอบสนองจากพีแอลซีเขียนข้อมูลที่ Data Memory ตามจำนวนที่กำหนดและเริ่มต้นการเขียนเป็นเลขฐานสิบหกตามการกำหนดการเริ่มต้น

10. AR AREA WRITE (WJ)

เขียนข้อมูลที่ Auxiliary Relay โดยมีตำแหน่งเริ่มต้นโดยสามารถกำหนดได้ที่ Beginning word 0000 ถึง 0027 ซึ่งจะเขียนข้อมูลเป็นแบบ Word

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 แสดงการเขียนข้อมูลที่ Auxiliary Relay

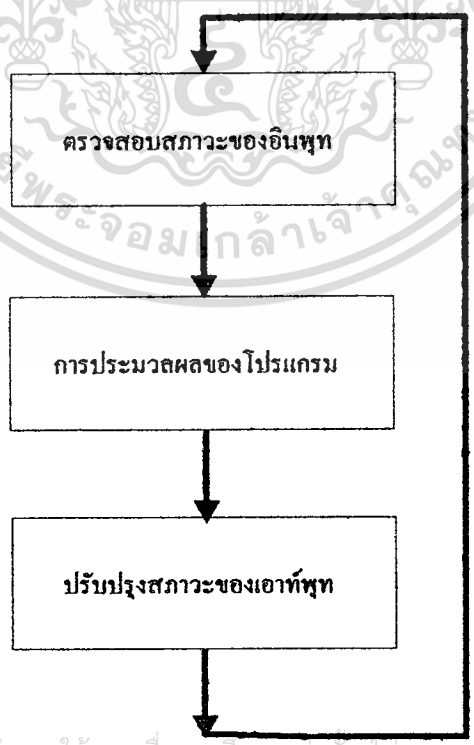
รูปแบบการตอบสนองจากพีแอลซีเขียนข้อมูลที่ Auxiliary Relay ตามจำนวนที่กำหนดและเริ่มต้นการเขียนเป็นเลขฐานสิบหกตามการกำหนดการเริ่มต้น

2.3.6 พีแอลซี(PLC)

พีแอลซี(PLC) ย่อมาจาก PROGRAMMABLE LOGIC CONTROLLER เป็นอุปกรณ์ชนิดหนึ่งที่นิยมใช้กันมากในงานของระบบควบคุมแบบซีเควนซ์(SEQUENCE CONTROL)

การทำงานของ PLC

PLC ทำงานโดยการสแกน (SCAN) โปรแกรมอย่างต่อเนื่องการ SCAN 1 รอบจะประกอบด้วยขั้นตอนที่สำคัญ 3 ขั้นตอน (อาจจะมีมากกว่า 3 ขั้นตอนก็ได้) ดังรูปที่ 2.28



รูปที่ 2.28 แสดงการวงรอบการทำงานของพีแอลซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 1 ตรวจสอบสภาวะของอินพุท

ขั้นแรก PLC จะตรวจสอบอินพุทแต่ละอินพุทว่ามีสภาวะ ON หรือ OFF โดยจะทำการตรวจสอบดูว่า sensor ที่ต่ออยู่กับอินพุทตัวแรกว่ามีสภาวะเป็นอย่างไรแล้วจึงตรวจสอบอินพุทตัวที่ 2 และ 3 ไปเรื่อย ๆ จากนั้นจะทำการบันทึกข้อมูลที่ได้ลงในหน่วยความจำ (memory) เพื่อจะใช้ในขั้นตอนต่อไป

ขั้นตอนที่ 2 การประมวลผลของโปรแกรม

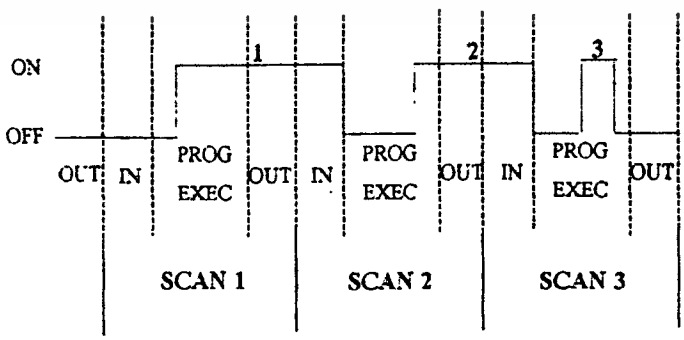
ขั้นตอนนี้ PLC จะทำการประมวลผลตามโปรแกรมที่ได้เขียนขึ้นโดยจะทำเพียงคำสั่งละครั้ง ตัวอย่างเช่น ตัวโปรแกรมอาจจะบอกว่า "ถ้าอินพุทแรกมีสภาวะเป็น ON จากนั้นจะทำให้เอาท์พุทแรกมีสภาวะ ON" เนื่องเรารู้มาแล้วว่าอินพุทที่มีสภาวะเป็น ON หรือ OFF จากขั้นตอนที่แล้ว ดังนั้นจึงทำให้ PLC สามารถตัดสินใจได้ว่าเอาท์พุทแรกควรจะมีสภาวะเป็นอย่างไร จากนั้นก็จะเก็บผลที่ได้ไว้ในหน่วยความจำเพื่อใช้ในขั้นตอนต่อไป

ขั้นตอนที่ 3 ปรับปรุงสภาวะของเอาท์พุท

ขั้นตอนสุดท้ายคือ PLC จะทำการปรับปรุงสภาวะของเอาท์พุท โดยในการปรับปรุงค่าสภาวะของเอาท์พุทนี้จะขึ้นอยู่กับสภาวะของอิทพุทที่ได้มาจากขั้นตอนแรก และผลจากการประมวลผลตามโปรแกรมในช่วงขั้นตอนที่สอง

หลังจากเสร็จขั้นตอนที่สามแล้ว PLC จะกลับไปขั้นตอนที่ 1 และทำซ้ำไปเรื่อย ๆ อย่างต่อเนื่อง

ดังนั้น 1 scan time จะหมายถึงเวลาที่ใช้ในการทำงานตามขั้นตอนทั้ง 3 ที่ได้กล่าวมา

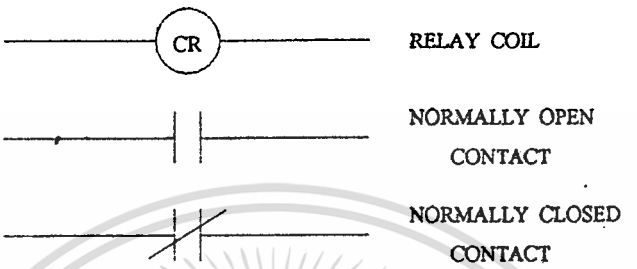


รูปที่ 2.29 แสดงเวลาในการทำงานทั้ง 3 ขั้นตอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RELAY LOGIC DIAGRAM

Relay คือ สวิทช์แม่เหล็กไฟฟ้าที่ทำงานโดย การจ่ายแรงดันให้กับขดลวด (coil) ทำให้เกิดสนามแม่เหล็กขึ้น สนามแม่เหล็กนี้จะดูด หน้าสัมผัส (contacts) ของรีเลย์เพื่อทำการเปิดหรือปิดวงจำไฟฟ้า



รูปที่ 2.30 แสดงสัญลักษณ์ของรีเลย์

Relay coil จะแทนด้วยวงกลม โดยจะมีอักษรย่อ กำกับคือ CR (Control Relay) ถ้าในวงจรมีรีเลย์มากกว่า 1 ตัว ก็จะใช้ตัวเลขกำกับ เช่น CR1 และ CR2 เป็นต้น โดยที่อักษรกำกับนี้จะใช้กับหน้าสัมผัสต่าง ๆ โดย

หน้าสัมผัสแบบปกติเปิด จะแทนด้วยเส้นตรงสองเส้นขนานกัน

ส่วนหน้าสัมผัสแบบปกติปิด จะแทนด้วยเส้นตรงสองเส้น ขนานกัน และมีเส้นตรงขีดคร่อม

ปกติแล้วหน้าสัมผัสจะ open อยู่ในขณะที่ coil ยังไม่ทำงาน เมื่อ coil ทำงานหน้าสัมผัสจะ close ทันที และเมื่อการทำงานของ coil หยุดลงจะทำให้ เกิดการหน่วงเวลาก่อนที่หน้าสัมผัสจะ open

หลักการโปรแกรมพีแอลซีโดยใช้ภาษาแลดเดอร์

ภาษาแลดเดอร์ ประกอบด้วยสัญลักษณ์หน้าสัมผัส มีลักษณะคล้ายวงจรรีเลย์ การเขียนโปรแกรมภาษาแลดเดอรจากวงจรรีเลย์จึงทำได้ง่าย พีแอลซีที่ใช้ภาษาแลดเดอร์คือพีแอลซีขนาดเล็กและขนาดกลาง ปัจจุบันคำสั่งภาษาแลดเดอร์มีครบทั้ง 6 กลุ่มคำสั่งคือ วงจรรีเลย์และปฏิบัติตรรก,การหน่วงเวลาและนับจำนวน,การคำนวณทางคณิตศาสตร์,การจัดการข้อมูล,การเคลื่อนย้ายข้อมูล และคำสั่งควบคุมโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของการเขียน LADDER DIAGRAM

1.สัญลักษณ์ของอุปกรณ์ที่แทนโหลดความต้านทานหรือความนำ จะถูกวางไว้ด้านขวาของแผนภาพ

2. สัญลักษณ์ของอุปกรณ์ที่แทนหน้าสัมผัส จะถูกวางไว้ด้านซ้ายของแผนภาพ

3. อุปกรณ์ที่ถูกต่อขนานกับอุปกรณ์ตัวอื่นเรียกว่า "Branch" ชุดภาพที่เป็นเส้นเชื่อมต่ออุปกรณ์ จากซ้ายมือไปจนถึงขวามือของเรียกว่า "Rung"

4. สัญลักษณ์ของอุปกรณ์ในแผนภาพจะถูกแสดงไว้สถานะปกติ โดยจะมีการเปลี่ยนสถานะเมื่อตัวอุปกรณ์เกิดการทำงาน (energizec)

5. อุปกรณ์ที่ทำหน้าที่หยุดการทำงาน (Stop) โดยปกติแล้วจะถูกต่อแบบอนุกรม อุปกรณ์ที่ทำหน้าที่เริ่มการทำงาน (Start) จะถูกต่อแบบขนาน

ข้อแนะนำในการเขียน LADDER DIAGRAM

1. กำหนดกระบวนการที่ต้องการจะควบคุม

2. วาดแบบร่างของการบวนการทำงานอย่างคร่าว ๆ โดยต้องแน่ใจว่าอุปกรณ์ทั้งหมดได้ถูกกำหนดไว้ในแผนภาพนั้น

3. กำหนดลำดับการทำงาน เขียนรายละเอียดของลำดับการทำงานแต่ละขั้นตอนให้มากที่สุดเท่าที่จะเป็นไปได้ โดยให้เขียนในรูปของ ประโยคคำพูด หรือเขียนลงในตาราง

4. ทำการเขียน LADDER DIAGRAM จากลำดับการทำงานที่ได้นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

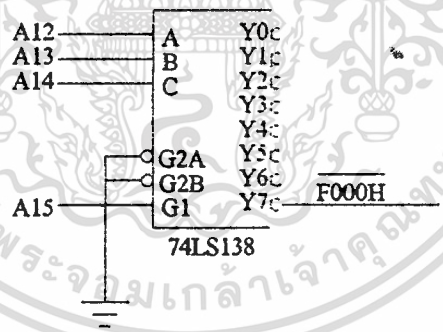
3.1 หลักการออกแบบวงจร

เนื่องจากการแสดงสะภาวะของพีแอลซีมีส่วนสำคัญที่ถือเป็นหัวใจหลักของระบบ คือ หน่วยประมวลผลส่วนกลางซึ่งใช้ไมโครคอนโทรลเลอร์ในตระกูล MCS-51 เป็นตัวควบคุมระบบการทำงานทั้งหมด และทางผู้จัดทำได้เลือกใช้ไมโครคอนโทรลเลอร์เบอร์ 8051ใช้งานร่วมกับหน่วนความจำภายนอกด้วย เพราะโปรแกรมในการควบคุมมีขนาดใหญ่มากจึงจำเป็นที่จะต้องใช้รอม(ROM)จากภายนอกที่มีขนาดพื้นที่หน่วยความจำสามสิบสองกิโลไบท์ และเพื่ออำนวยความสะดวกในการพัฒนาโปรแกรมต่อไปด้วย

3.2 การอินเตอร์เฟส(Interface)

3.21 การอินเตอร์เฟสกับกราฟฟิกแอลซีดี

สำหรับการอินเตอร์เฟสกับกราฟฟิกแอลซีดีนั้นจะใช้การอินเตอร์เฟสแบบเมมโมรี่แมบ (MEMORY MAP) โดยใช้ไอซีเบอร์ 74LS138 เป็นตัวถอดรหัสหน่วยความจำซึ่งในโครงงานนี้จะอยู่ที่ตำแหน่ง F000H เหตุผลที่เลือกการอินเตอร์เฟสแบบเมมเมอรี่แมบ เพราะว่าเป็นการประหยัดพอร์ทของซีพียู(CPU)



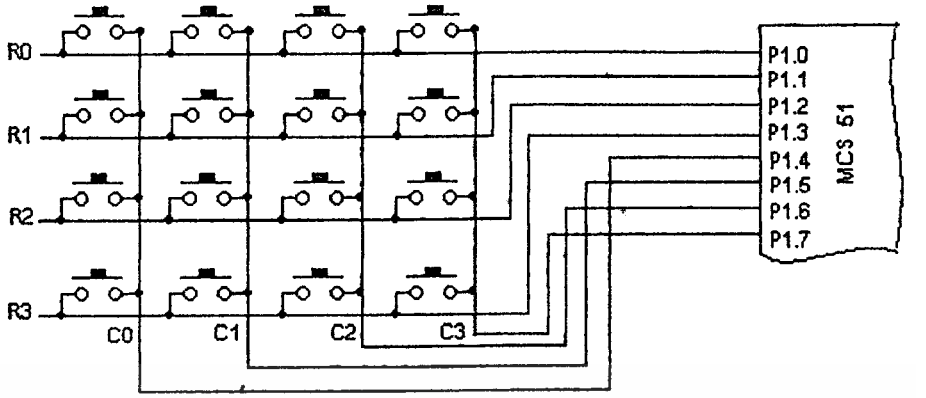
รูปที่ 3.1 แสดงวงจรถอดรหัสหน่วยความจำ

3.2.2 การอินเตอร์เฟสกับคีย์บอร์ด

คีย์บอร์ดที่ใช้มีทั้งหมด 16 คีย์ด้วยกันโดยแบ่งเป็นตัวเลข 0-9 เป็นจำนวน 10 คีย์ คีย์ยกเลิกการเซตค่า(CANCEL) 1 คีย์ คีย์ตกลง(ENTER) 1 คีย์ คีย์เลื่อนไปทางซ้าย 1 คีย์ คีย์เลื่อนไปทางขวา 1 คีย์ คีย์เพิ่มค่า 1 คีย์ และคีย์ลดค่า 1 คีย์ สำหรับการต่อคีย์จะเป็นแบบเมตตริก ทำให้มีสายต่อใช้งานกับพอร์ทหนึ่งของไมโครคอนโทรลเลอร์เพียง 8 เส้นเท่านั้นจากทั้งหมด 16 คีย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงเป็นการประหยัดพอร์ทที่จะใช้ในการอินเตอร์เฟสกับคีย์บอร์ด ลักษณะการอินเตอร์เฟสคีย์บอร์ดกับไมโครคอนโทรลเลอร์แสดงดังรูปที่ 3.2

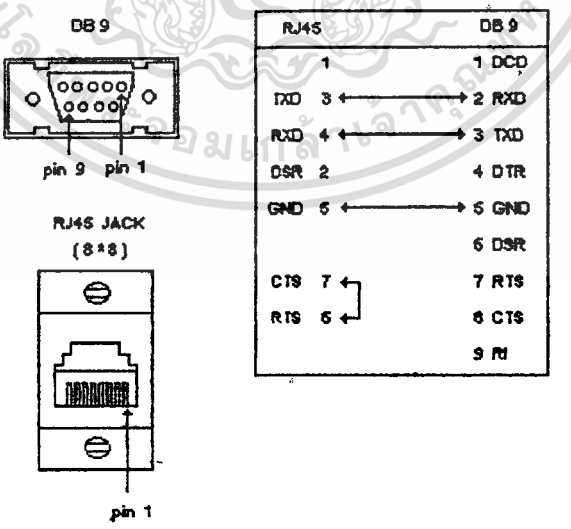


รูปที่ 3.2 แสดงการอินเตอร์เฟสกับคีย์บอร์ด

3.2.3 การอินเตอร์เฟสกับพีแอลซี(PLC)

เนื่องจากใช้พีแอลซีของโมดิคอน (MODICON) ในการทดลองซึ่งมีมาตรฐานในการติดต่อเป็นแบบ RS-232

การส่งข้อมูลจากพีแอลซีจะส่งผ่านทางขั้วต่อแบบ RJ-45 ส่วนทางด้านรับข้อมูลของไมโครคอนโทรลเลอร์จะใช้ DB 9 ลักษณะการอินเตอร์เฟสจะแสดงดังรูปที่ 3.3



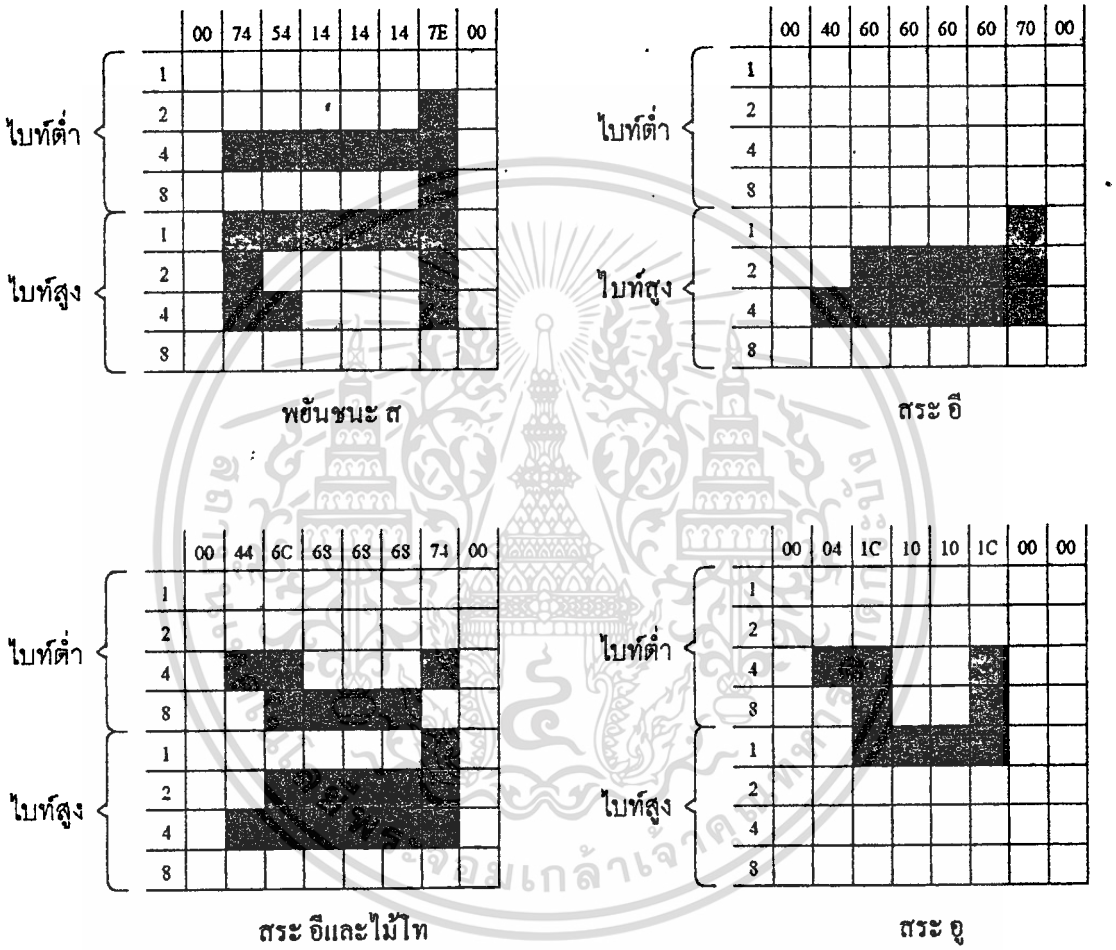
รูปที่ 3.3 แสดงการอินเตอร์เฟสระหว่างพีแอลซีกับไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 วิธีการออกแบบตัวอักษรภาษาไทย

หน้าจอของกราฟิกแอลซีดีในหนึ่งแถวจะมีจุดในแนวตั้งทั้งหมดแปดจุดและทั้งหน้าจอมีทั้งหมดแปดแถว จึงกำหนดให้ตัวอักษรหนึ่งตัวมีขนาด 8 * 8 จุดดังแสดงในรูปที่ 3.4

ในการออกแบบสระจะสังเกตุเห็นจากรูปที่ 3.4 ได้ว่าจะมีการออกแบบให้สระอีรวมอยู่กับวรรณยุกต์ไม้โทเพื่อให้เกิดความสะดวกในการเขียนโปรแกรม



รูปที่ 3.4 แสดงวิธีการกำหนดตัวอักษร

ตัวอย่างการออกแบบตัวอักษร “ส”

- ทำตารางตามรูปที่ 1มีขนาดแปดคูณแปด
- กำหนดจุดให้มีตัวอักษรเป็นรูป “ส”
- หลักที่หนึ่งไบท์สูงไม่มีจุดที่กำหนดจึงให้มีค่าเป็น 0
- หลักที่หนึ่งไบท์ต่ำไม่มีจุดที่กำหนดจึงให้มีค่าเป็น 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อรวมไบท์สูงและไบท์ต่ำจึงมีค่าเท่ากับ 00H
- หลักที่สองไบท์สูงมีจุดอยู่ในตำแหน่งที่ 4, 2 และ1 ทำการคำนวณโดยการบวกตำแหน่งทั้งสาม โดยบวกเป็นเลขฐานสิบหกจะได้ 4+2+1 = 7
- หลักที่สองไบท์ต่ำมีจุดอยู่ในตำแหน่งที่ 4 เพียงจุดเดียวจึงมีค่าเป็น 4
- กำหนดให้ผลรวมไบท์สูงเป็นไบท์แรกและผลรวมไบท์ต่ำเป็นไบท์ที่สองจะได้ค่าเท่ากับ 74H
- หลักที่สามไบท์สูงมีจุดอยู่ในตำแหน่งที่ 4 และ1 ทำการคำนวณโดยการบวกตำแหน่งทั้งสอง โดยบวกเป็นเลขฐานสิบหกจะได้ 4+1 = 5
- หลักที่สามไบท์ต่ำมีจุดอยู่ในตำแหน่งที่ 4 เพียงจุดเดียวจึงมีค่าเป็น 4
- กำหนดให้ผลรวมไบท์สูงเป็นไบท์แรกและผลรวมไบท์ต่ำเป็นไบท์ที่สองจะได้ค่าเท่ากับ 54H
- หลักที่สี่ไบท์สูงมีจุดอยู่ในตำแหน่งที่ 1 เพียงจุดเดียวจึงมีค่าเป็น 1
- หลักที่สี่ไบท์ต่ำมีจุดอยู่ในตำแหน่งที่ 4 เพียงจุดเดียวจึงมีค่าเป็น 4
- กำหนดให้ผลรวมไบท์สูงเป็นไบท์แรกและผลรวมไบท์ต่ำเป็นไบท์ที่สองจะได้ค่าเท่ากับ 14H
- หลักที่ห้าไบท์สูงมีจุดอยู่ในตำแหน่งที่ 1 เพียงจุดเดียวจึงมีค่าเป็น 1
- หลักที่ห้าไบท์ต่ำมีจุดอยู่ในตำแหน่งที่ 4 เพียงจุดเดียวจึงมีค่าเป็น 4
- กำหนดให้ผลรวมไบท์สูงเป็นไบท์แรกและผลรวมไบท์ต่ำเป็นไบท์ที่สองจะได้ค่าเท่ากับ 14H
- หลักที่หกไบท์สูงมีจุดอยู่ในตำแหน่งที่ 1 จึงให้มีค่าเป็น 1
- หลักที่หกไบท์ต่ำมีจุดอยู่ในตำแหน่งที่ 4 เพียงจุดเดียวจึงมีค่าเป็น 4
- กำหนดให้ผลรวมไบท์สูงเป็นไบท์แรกและผลรวมไบท์ต่ำเป็นไบท์ที่สองจะได้ค่าเท่ากับ 14H
- หลักที่เจ็ดไบท์สูงมีจุดอยู่ในตำแหน่งที่ 4, 2 และ1 ทำการคำนวณโดยการบวกตำแหน่งทั้งสาม โดยบวกเป็นเลขฐานสิบหกจะได้ 4+2+1 = 7
- หลักที่เจ็ดไบท์ต่ำมีจุดอยู่ในตำแหน่งที่ 8,4 และ2 ทำการคำนวณโดยการบวกตำแหน่งทั้งสามโดยบวกเป็นเลขฐานสิบหกจะได้ E
- กำหนดให้ผลรวมไบท์สูงเป็นไบท์แรกและผลรวมไบท์ต่ำเป็นไบท์ที่สองจะได้ค่าเท่ากับ 7EH
- หลักที่แปดไบท์สูงไม่มีจุดที่กำหนดจึงให้มีค่าเป็น 0
- หลักที่แปดไบท์ต่ำไม่มีจุดที่กำหนดจึงให้มีค่าเป็น 0
- เมื่อรวมไบท์สูงและไบท์ต่ำจึงมีค่าเท่ากับ 00H
- เมื่อรวมค่าที่ได้ทั้งหมดตัวอักษร"ส"จะมีข้อมูลเป็น00H,74H,54H,14H,14H,14H,7EH,00H

ข้อมูลตัวอักษรที่ได้ออกแบบไว้

ก(KOR_KAI) ข้อมูลคือ 00H,74H,0CH,04H,04H,04H,7CH,00H

ข(KHOR_KHAI) ข้อมูลคือ 00H.04H,3CH,40H,40H,3CH,00H,00H

ข(KOR_KUA) ข้อมูลคือ 00H,04H,08H,3CH,40H,40H,3CH,00H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค(KOR_KWAI)	ข้อมูลคือ	00H,7CH,64H,14H,14H,04H,7CH,00H
ฆ(KOR_RAKANG)	ข้อมูลคือ	00H,04H,28H,7CH,20H,20H,20H,7CH
ง(NGOR_NGOO)	ข้อมูลคือ	00H,10H,20H,40H,44H,7CH,00H,00H
จ(JOR_JAN)	ข้อมูลคือ	00H,00H,04H,14H,14H,64H,7CH,00H
ฉ(CHOR_CHING)	ข้อมูลคือ	00H,14H,74H,44H,44H,24H,7CH,20H
ช(CHOR_CHANG)	ข้อมูลคือ	00H,00H,04H,3CH,40H,44H,3AH,02H
ซ(SOR_SO)	ข้อมูลคือ	00H,04H,08H,3CH,40H,44H,3AH,02H
ฌ(CHOR_KACHER)	ข้อมูลคือ	00H,74H,4CH,04H,24H,7CH,20H,7CH
ญ(YOR_YING)	ข้อมูลคือ	00H,74H,4CH,84H,84H,FCH,40H,7CH
ฎ(CHOR_CHADA)	ข้อมูลคือ	00H,40H,74H,8CH,84H,84H,FCH,00H
ฏ(TOR_PATAK)	ข้อมูลคือ	00H,40H,74H,8CH,C4H,84H,FCH,00H
ฐ(TOR_TAN)	ข้อมูลคือ	00H,00H,88H,CCH,ACH,ECH,7CH,00H
ฑ(TOR_MONTO)	ข้อมูลคือ	00H,04H,08H,7CH,10H,08H,04H,7CH
ฒ(TOR_POOTOA)	ข้อมูลคือ	00H,7CH,54H,78H,24H,7CH,20H,7CH
ณ(NOR_NAND)	ข้อมูลคือ	00H,74H,4CH,04H,7CH,20H,7CH,20H
ด(DOR_DEK)	ข้อมูลคือ	00H,7CH,44H,54H,34H,04H,7CH,00H
ต(TOR_TOA)	ข้อมูลคือ	00H,7CH,44H,54H,38H,04H,7CH,00H
ถ(TOR_TUNG)	ข้อมูลคือ	00H,74H,4CH,04H,04H,04H,7CH,00H
ท(TOR_TAHARN)	ข้อมูลคือ	00H,04H,7CH,10H,08H,04H,7CH,00H
ธ(TOR_TONG)	ข้อมูลคือ	00H,00H,68H,54H,54H,54H,74H,00H
น(NOR_NOO)	ข้อมูลคือ	00H,04H,7CH,20H,20H,20H,7CH,20H
บ(BOR_BAIMAI)	ข้อมูลคือ	00H,04H,7CH,40H,40H,40H,7CH,00H
ป(POR_PLA)	ข้อมูลคือ	00H,04H,7CH,40H,40H,40H,7EH,00H
ผ(POR_PAUNG)	ข้อมูลคือ	00H,7CH,44H,40H,70H,40H,7CH,00H
ฝ(FOR_FAR)	ข้อมูลคือ	00H,7CH,44H,40H,70H,40H,7EH,00H
พ(POR_PAN)	ข้อมูลคือ	00H,04H,7CH,40H,70H,40H,7CH,00H
ฟ(FOR_FUN)	ข้อมูลคือ	00H,04H,7CH,40H,70H,40H,7EH,00H
ภ(POR_SOMPOA)	ข้อมูลคือ	00H.40H,74H,0CH,04H,04H,7CH,00H
ม(MOR_MA)	ข้อมูลคือ	00H.24H,7CH,20H,20H,20H,7CH,00H
ย(YOR_YAK)	ข้อมูลคือ	00H.7CH,54H,50H,40H,40H,7CH,00H
ร(ROR_RU)	ข้อมูลคือ	00H,00H,08H,14H,14H,54H,74H,00H
ล(LOR_LING)	ข้อมูลคือ	00H,74H,54H,14H,14H,14H,7CH,00H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารทุกครั้งที่มีการนำไปใช้

ว(VOR_WAN)	ข้อมูลคือ	00H,00H,04H,04H,04H,44H,7CH,00H
ศ(SOR_SALAR)	ข้อมูลคือ	00H,7CH,64H,14H,14H,04H,7EH,00H
ษ(SOR_RUSI)	ข้อมูลคือ	00H,04H,7CH,40H,58H,50H,7CH,10H
ส(SOR_SUA)	ข้อมูลคือ	00H,74H,54H,14H,14H,14H,7EH,00H
ห(HOR_HEEP)	ข้อมูลคือ	00H,04H,7CH,10H,08H,0CH,74H,00H
ฬ(LOR_JURA)	ข้อมูลคือ	00H,04H,7CH,40H,70H,44H,7EH,04H
อ(OR_ANG)	ข้อมูลคือ	00H,74H,54H,44H,44H,44H,7CH,00H
ฮ(HOR_NOKHOOK)	ข้อมูลคือ	00H,74H,54H,44H,44H,44H,7EH,00H
ะ(AA)	ข้อมูลคือ	00H,00H,6CH,6CH,48H,48H,24H,00H
า(AR)	ข้อมูลคือ	00H,00H,00H,04H,04H,04H,7CH,00H
อิ(I)	ข้อมูลคือ	00H,40H,60H,60H,60H,60H,40H,00H
อิ่(IAKE)	ข้อมูลคือ	00H,40H,60H,60H,60H,6CH,40H,00H
อิ้(ITO)	ข้อมูลคือ	00H,44H,6CH,68H,68H,68H,44H,00H
อี๊(ITEE)	ข้อมูลคือ	00H,4EH,6AH,64H,62H,6EH,48H,0CH
อิ๋(IJATAWA)	ข้อมูลคือ	00H,40H,64H,64H,6EH,64H,44H,00H
อิ์(IKARUN)	ข้อมูลคือ	00H,40H,60H,6CH,6AH,62H,42H,00H
อี(E)	ข้อมูลคือ	00H,40H,60H,60H,60H,60H,70H,00H
อี่(EAKE)	ข้อมูลคือ	00H,40H,60H,60H,60H,6CH,70H,00H
อี้(ETO)	ข้อมูลคือ	00H,44H,6CH,68H,68H,68H,74H,00H
อี๊(ETEE)	ข้อมูลคือ	00H,4EH,6AH,64H,62H,6EH,78H,0CH
อี๋(EJATAWA)	ข้อมูลคือ	00H,40H,64H,64H,6EH,64H,74H,00H
อื(AU)	ข้อมูลคือ	00H,40H,60H,60H,60H,50H,60H,00H
อื่(AUAKE)	ข้อมูลคือ	00H,40H,60H,60H,60H,5CH,60H,00H
อื้(AUTO)	ข้อมูลคือ	00H,44H,6CH,68H,68H,58H,64H,00H
อื๊(AUTEE)	ข้อมูลคือ	00H,4EH,6AH,64H,62H,5EH,68H,0CH
อื๋(AUJATAWA)	ข้อมูลคือ	00H,40H,64H,64H,6EH,54H,64H,00H
อือ(AUE)	ข้อมูลคือ	00H,40H,60H,60H,70H,40H,70H,00H
อื่อ(AUEAKE)	ข้อมูลคือ	00H,40H,60H,60H,70H,4CH,70H,00H
อื้อ(AUETO)	ข้อมูลคือ	00H,44H,6CH,68H,78H,48H,74H,00H
อื๊อ(AUETEE)	ข้อมูลคือ	00H,4EH,6AH,64H,72H,4EH,78H,0CH
อื๋อ(AUEJATAWA)	ข้อมูลคือ	00H,40H,64H,64H,7EH,44H,74H,00H
อุ(U)	ข้อมูลคือ	00H,00H,00H,04H,04H,1CH,00H,00H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

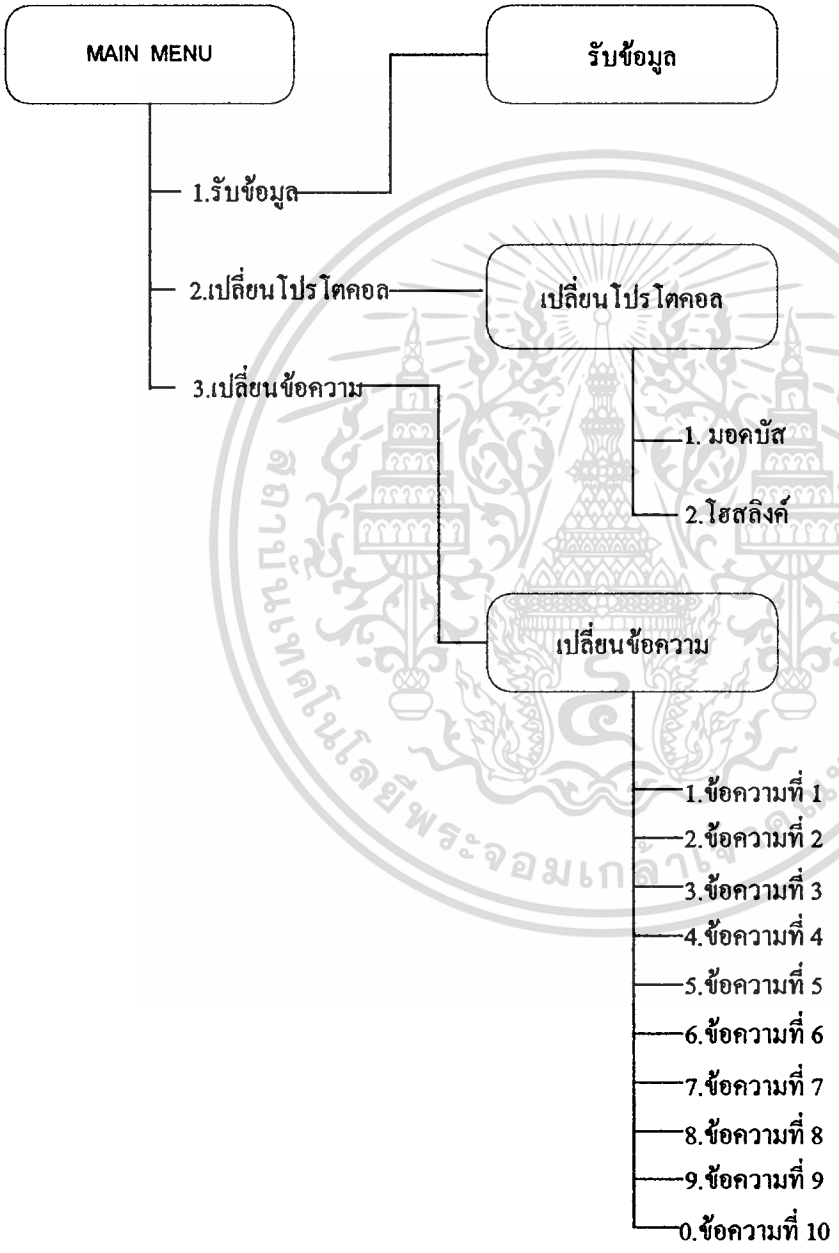
อู(UU)	ข้อมูลคือ	00H,04H,1CH,10H,10H,1CH,00H,00H
เอ(AO)	ข้อมูลคือ	00H,00H,7CH,40H,00H,00H,00H,00H
แอ(AIR)	ข้อมูลคือ	00H,7CH,40H,00H,7CH,40H,00H,00H
โอ(O)	ข้อมูลคือ	00H,02H,05H,05H,05H,7DH,41H,01H
อำ(UM)	ข้อมูลคือ	00H,00H,20H,50H,50H,20H,00H,00H
อ่ำ(UMAKE)	ข้อมูลคือ	00H,00H,20H,50H,5CH,20H,00H,00H
อ้ำ(UMTO)	ข้อมูลคือ	00H,04H,2CH,58H,58H,28H,04H,00H
อ๊ำ(UMTEE)	ข้อมูลคือ	00H,0EH,2AH,54H,52H,2EH,08H,0CH
อ๋ำ(UMJATAWA)	ข้อมูลคือ	00H,00H,24H,54H,5EH,24H,04H,00H
ใอ(MAI_MOON)	ข้อมูลคือ	00H,02H,05H,05H,03H,7FH,40H,00H
ไอ(MAI_MALAI)	ข้อมูลคือ	00H,01H,02H,02H,01H,7FH,40H,00H
อ่(MAI_AKE)	ข้อมูลคือ	00H,00H,00H,00H,60H,00H,00H,00H
อ้(MAI_TO)	ข้อมูลคือ	00H,20H,60H,40H,40H,40H,20H,00H
อ๊(MAI_TEE)	ข้อมูลคือ	00H,70H,50H,20H,10H,70H,40H,60H
อ๋((MAI_JATAWA)	ข้อมูลคือ	00H,20H,20H,70H,20H,20H,00H,00H
อั(MAI_HUNAKAD)	ข้อมูลคือ	00H,60H,60H,40H,40H,40H,20H,00H
อั่(MAI_HUNAKADAKE)	ข้อมูลคือ	00H,60H,60H,40H,4CH,40H,20H,00H
อั้(MAI_HUNAKADTO)	ข้อมูลคือ	00H,64H,6CH,48H,48H,48H,24H,00H
อั๊((MAI_HUNAKADTEE)	ข้อมูลคือ	00H,6EH,6AH,44H,42H,4EH,28H,0CH
อั๋(MAI_HUNAKADJATAWA)	ข้อมูลคือ	00H,60H,64H,44H,4EH,44H,24H,00H
ๆ(MAI_YAMOG)	ข้อมูลคือ	00H,00H,1CH,14H,48H,24H,1CH,00H
อี๋(MAI_TIKOO)	ข้อมูลคือ	00H,78H,48H,68H,48H,68H,68H,04H
อ์(KARUN)	ข้อมูลคือ	00H,00H,00H,60H,50H,10H,10H,00H
ฤ(LAU)	ข้อมูลคือ	0H,74H,4CH,04H,04H,04H,FCH,00H
ฯ(PAIYANG)	ข้อมูลคือ	00H,0CH,0CH,08H,08H,08H,7CH,00H
ZERO	ข้อมูลคือ	00H,00H,7CH,44H,44H,44H,7CH,00H
ONE	ข้อมูลคือ	00H,00H,08H,44H,7CH,40H,00H,00H
TWO	ข้อมูลคือ	00H,00H,74H,54H,54H,54H,5CH,00H
THREE	ข้อมูลคือ	00H,00H,44H,44H,54H,54H,7CH,00H
FOUR	ข้อมูลคือ	00H,00H,3CH,20H,20H,7CH,20H,00H
FIVE	ข้อมูลคือ	00H,00H,5CH,54H,54H,54H,74H,00H
SIX	ข้อมูลคือ	00H,00H,7CH,54H,54H,54H,74H,00H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SEVEN	ข้อมูลคือ	00H,00H,0CH,04H,04H,64H,1CH,00H
EIGHT	ข้อมูลคือ	00H,00H,7CH,54H,54H,54H,7CH,00H
NINE	ข้อมูลคือ	00H,00H,5CH,54H,54H,54H,7CH,00H

3.4 วิธีการใช้งานเครื่องแสดงสภาวะพีแอลซี

กดคีย์ตามตัวเลขที่แสดงอยู่ในคำสั่ง



รูปที่ 3.5 แสดงเมนูการเลือกหน้าที่การทำงานของเครื่องแสดงสภาวะพีแอลซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของการเปลี่ยนข้อความนั้นจะมีวิธีการดังนี้

- คีย์ลูกศรขึ้น ใช้หมุนตัวอักษรในตำแหน่งต่อไป เช่นจากตัวอักษร ก ไปอักษร ข สระ อิ ไป สระอี หรือสระ อุ ไป สระ อู เป็นต้น
- คีย์ลูกศรลง ใช้หมุนตัวอักษรในตำแหน่งก่อนหน้านี้ เช่นจากตัวอักษร ข ไปอักษร ก เป็นต้น
- คีย์ลูกศรซ้าย ใช้เลื่อนตำแหน่งที่จะทำการเปลี่ยนตัวอักษรไปทางซ้าย
- คีย์ลูกศรขวา ใช้เลื่อนตำแหน่งที่จะทำการเปลี่ยนตัวอักษรไปทางขวา
- คีย์ ESC ใช้ในการบันทึกข้อความที่ทำการเขียนเสร็จเรียบร้อยแล้ว
- คีย ENT กดครั้งที่ 1 เลื่อนไปทำการแก้สระด้านบนของตำแหน่งที่ชี้อยู่ กดครั้งที่ 2 เลื่อนไปทำการแก้สระด้านล่างของตำแหน่งที่ชี้อยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 วิธีการทดลอง

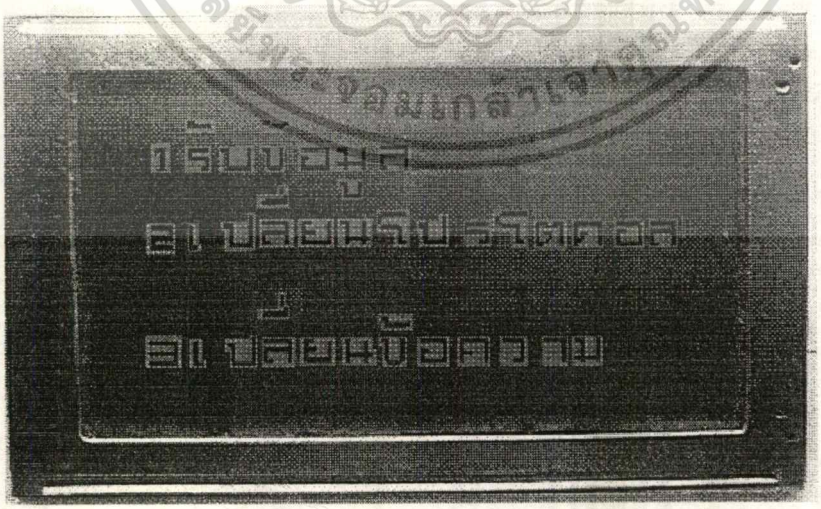
4.1.1 ทดลองการติดต่อกับพีแอลซีของโมดิคอน โดยการเขียนแลดเดอร์ให้พีแอลซีส่งค่ารหัสออกมาทางคอมสอง(COMM2)ของพีแอลซี และเชื่อมต่อกับคอมพิวเตอร์โดยการใช้โปรแกรมเทอร์มินอล(Terminal)ซึ่งอยู่ในแอสแซสเซอรี่ของโปรแกรมวินโดว์เป็นตัวรับโดยตั้งค่าการติดต่อเป็นDATA=7BIT,PARITY=EVEN,1 STOPBIT และBOUDRATE=9600 ตรวจสอบค่าที่ได้รับ

4.1.2 ทดลองทางด้านฮาร์ดแวร์(Hardware) ประกอบอุปกรณ์ลงบนแผ่นลายวงจรที่ได้ออกแบบไว้ ตรวจสอบความถูกต้องของแผ่นวงจรและป้อนไฟเข้าระบบ จากนั้นทำการโปรแกรมและปรับปรุงจนเป็นโปรแกรมที่ถูกต้องตามเงื่อนไขที่ต้องการ

4.1.3 เชื่อมต่ออุปกรณ์ทางฮาร์ดแวร์ทั้งหมดของระบบเข้าด้วยกัน,ตรวจสอบความถูกต้องในการเชื่อมต่อระบบจากนั้นจึงป้อนไฟเข้าระบบ,ทดลองรันโปรแกรมและแก้ไขข้อผิดพลาดทั้งในเรื่องอุปกรณ์ทางฮาร์ดแวร์ และซอฟแวร์จนได้ผลเป็นที่พึงพอใจ

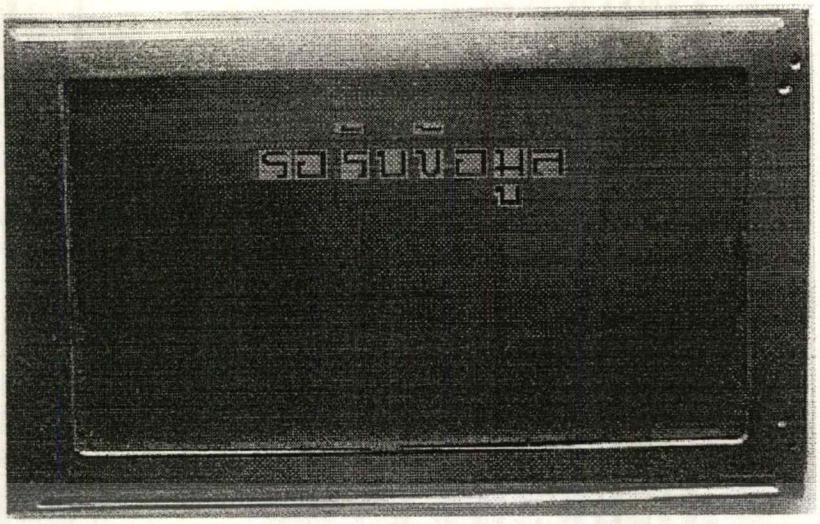
4.2 ผลการทดลอง

จากการแก้ไขและทดสอบโปรแกรมก็ปรากฏว่าสามารถที่จะโปรแกรมให้บอร์ดไมโครคอนโทรลเลอร์รับค่ารหัสที่พีแอลซีส่งมาและสามารถแสดงข้อความตามที่ได้กำหนดไว้ได้ซึ่งใช้กราฟฟิกแอลซีดีในการแสดงผลและได้ผลการทดลองดังรูปต่อไปนี้

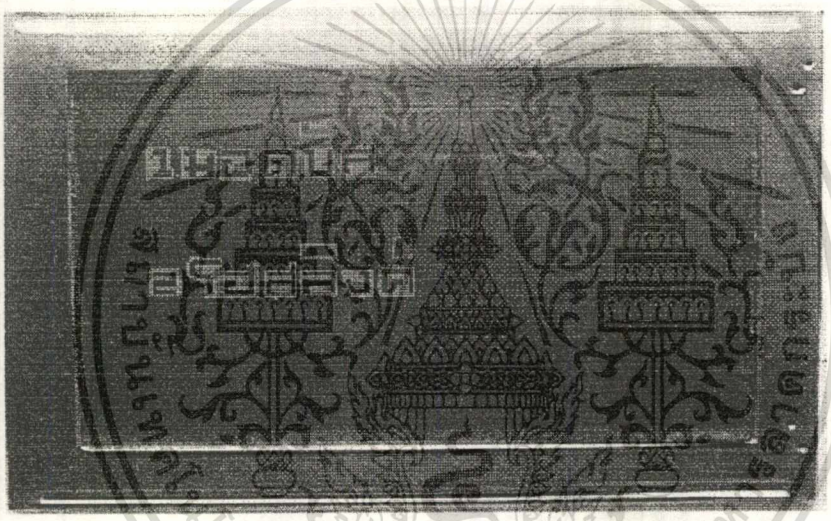


รูปที่ 4.1 แสดงหน้าจอเมนู

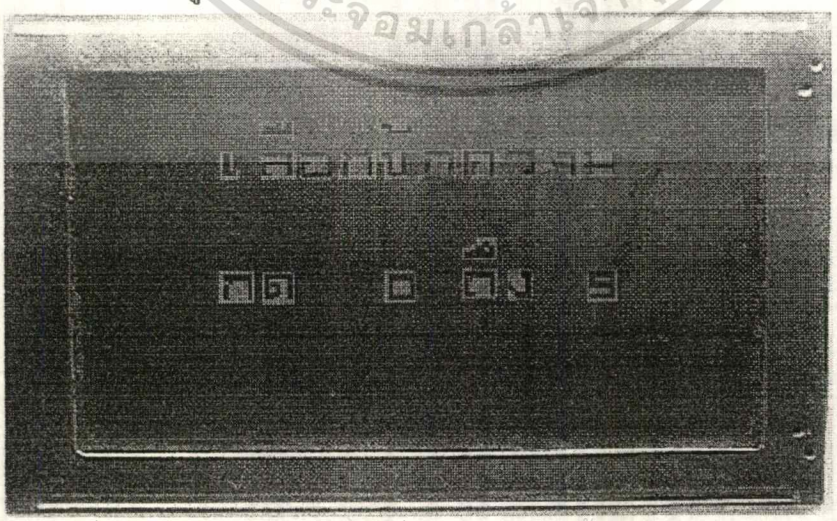
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงหน้าจอรอรับข้อมูล

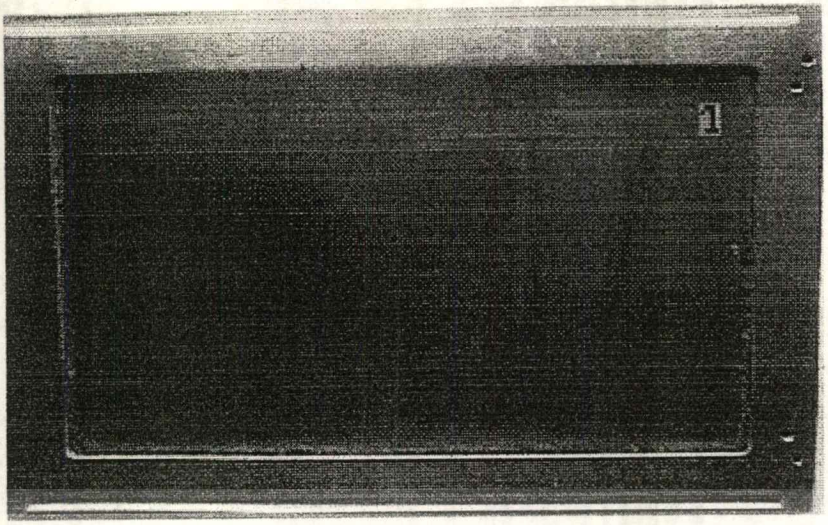


รูปที่ 4.3 แสดงหน้าจอเปลี่ยนโปรโตคอล

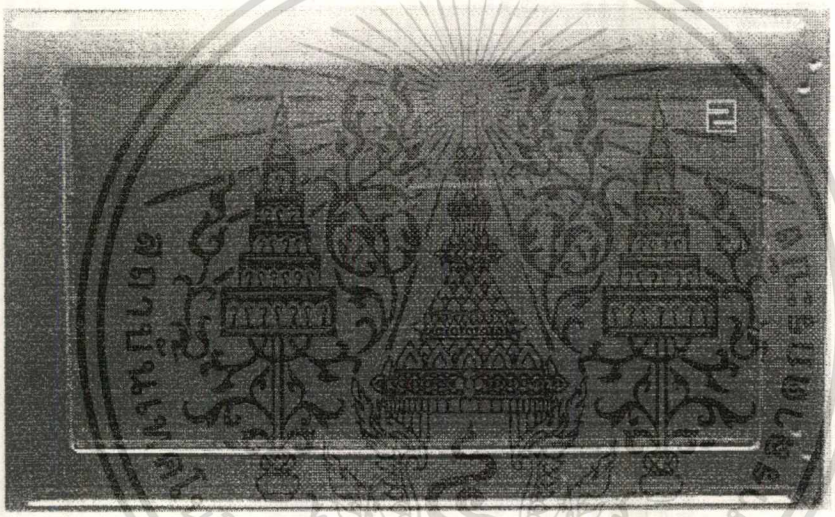


รูปที่ 4.4 แสดงหน้าจอเลือกข้อความ

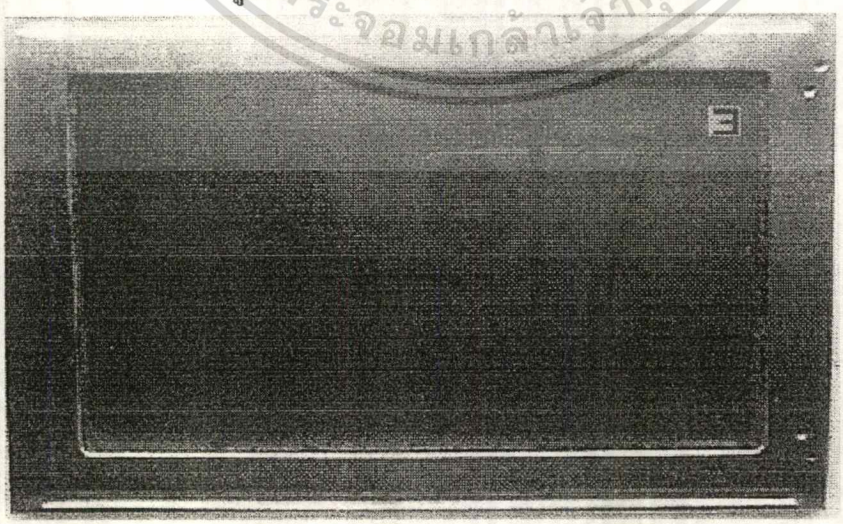
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงหน้าจอข้อความที่1

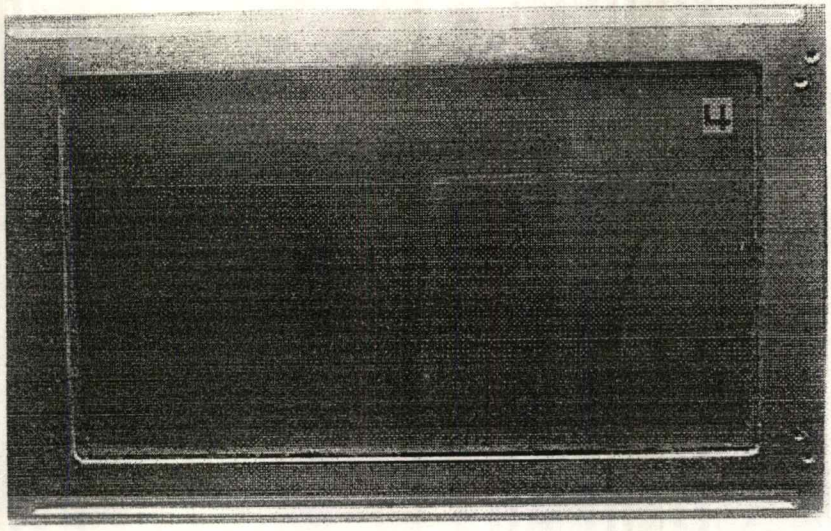


รูปที่ 4.6 แสดงหน้าจอข้อความที่2

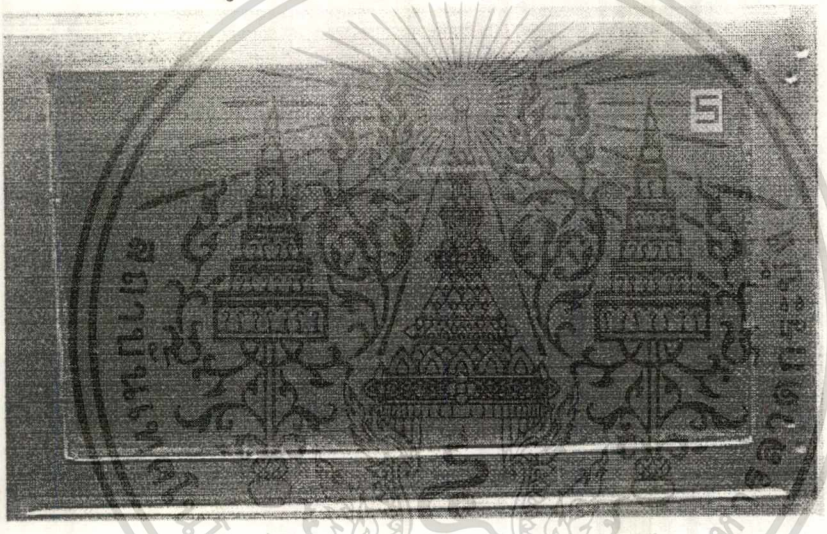


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

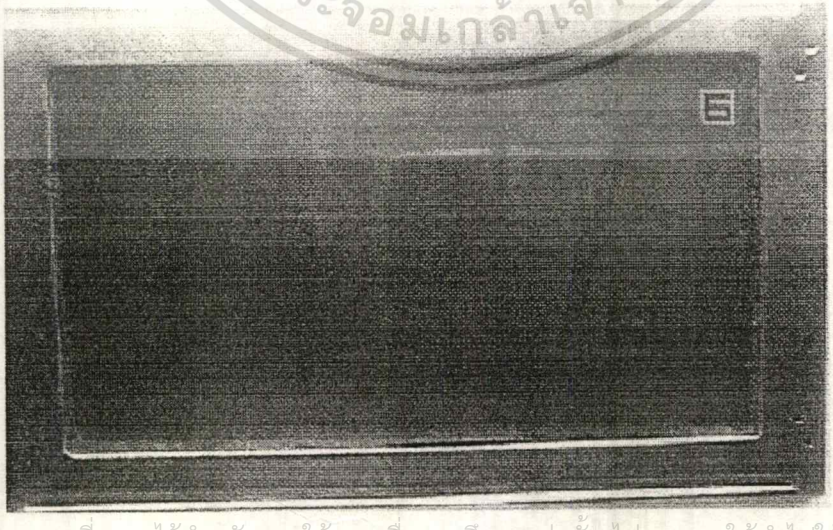
รูปที่ 4.7 แสดงหน้าจอข้อความที่3



รูปที่ 4.8 แสดงหน้าจอข้อความที่4

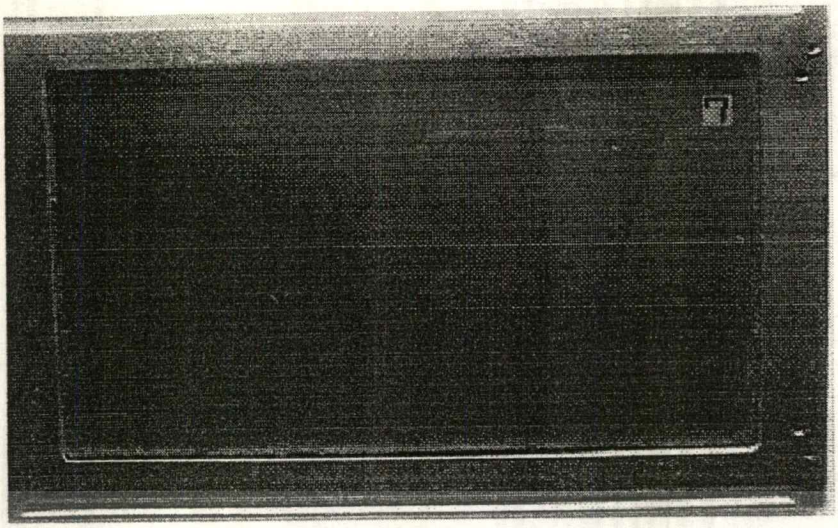


รูปที่ 4.9 แสดงหน้าจอข้อความที่5

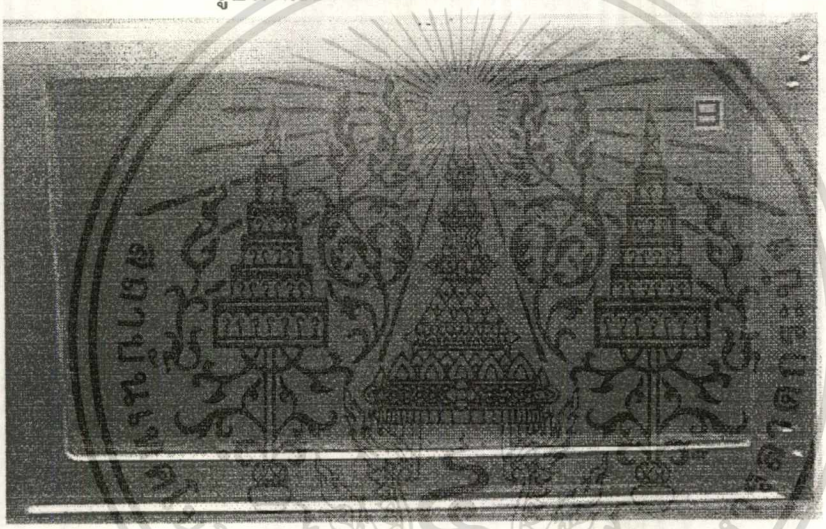


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

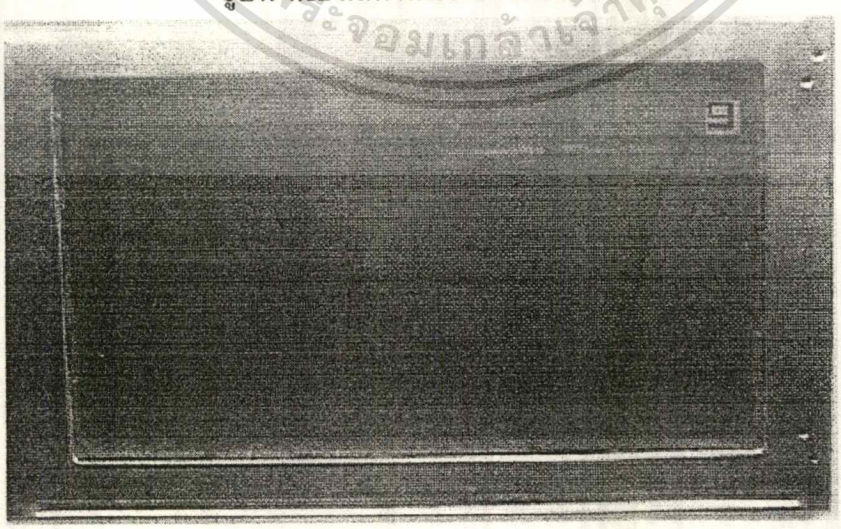
รูปที่ 4.10 แสดงหน้าจอข้อความที่6



รูปที่ 4.11 แสดงหน้าจอข้อความที่7

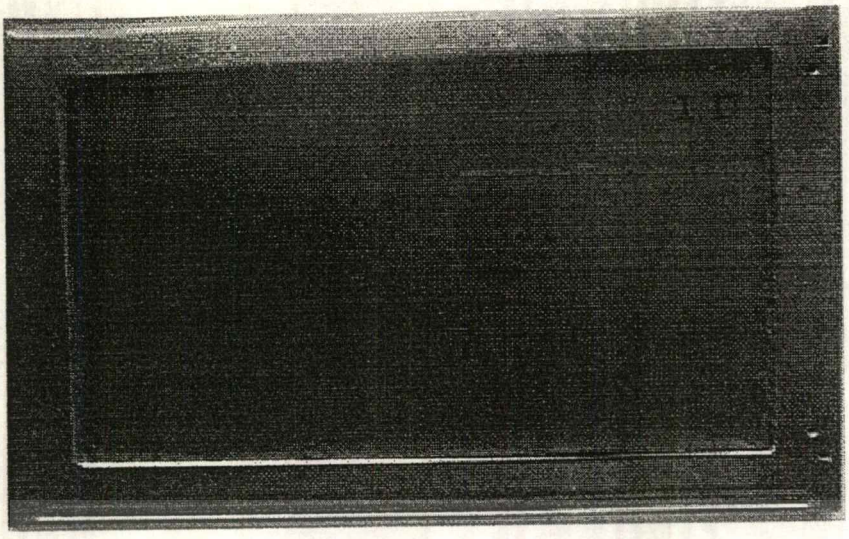


รูปที่ 4.12 แสดงหน้าจอข้อความที่8



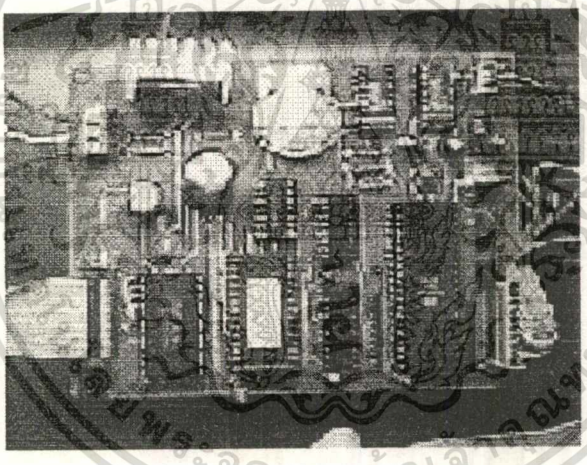
รูปที่ 4.13 แสดงหน้าจอข้อความที่9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

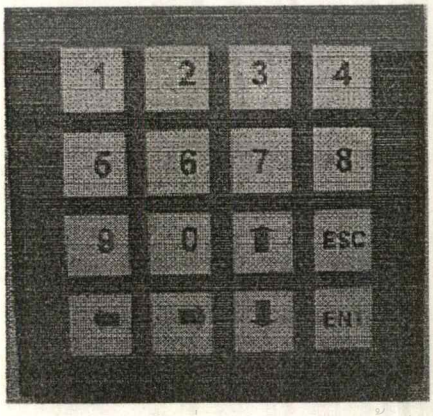


รูปที่ 4.14 แสดงหน้าจอข้อความที่10

สำหรับวงจรที่ใช้งานและอุปกรณ์ประกอบการใช้งานของระบบแสดงดังรูปต่อไปนี้



รูปที่ 4.15 แสดงบอร์ดควบคุมของเครื่องแสดงสภาวะพีแอลซี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.16 แสดงตำแหน่งของปุ่มคีย์บอร์ด

บทที่ 5

บทวิจารณ์และสรุป

จากการทดลองที่ผ่านมาผู้ปฏิบัติได้ประสบกับอุปสรรคและข้อบกพร่องที่เกิดขึ้นในการทดลองซึ่งเป็นสิ่งจำเป็นจะต้องทำการแก้ไขให้ดียิ่งขึ้น

5.1 ปัญหาที่เกิดขึ้นกับการทดลอง

5.1.1. แผ่นลายวงจรที่ได้ออกแบบไว้ลายทองแดงด้านบนและด้านล่างไม่ต่อถึงกัน

5.1.2. พีแอลซีไม่ส่งค่ารหัสออกมาหลักจากที่สั่งให้ส่งค่ารหัส

5.1.3. คอมพิวเตอร์แสดงค่ารหัสไม่ตรงกับที่พีแอลซีส่งมา

5.2 แนวทางในการแก้ปัญหา

5.2.1. ตรวจสอบลายวงจรแล้วพบว่าจุดต่อระหว่างลายวงจรด้านบน และด้านล่างไม่เชื่อมต่อกันแก้ไขโดยการเชื่อมต่อเข้าด้วยกัน

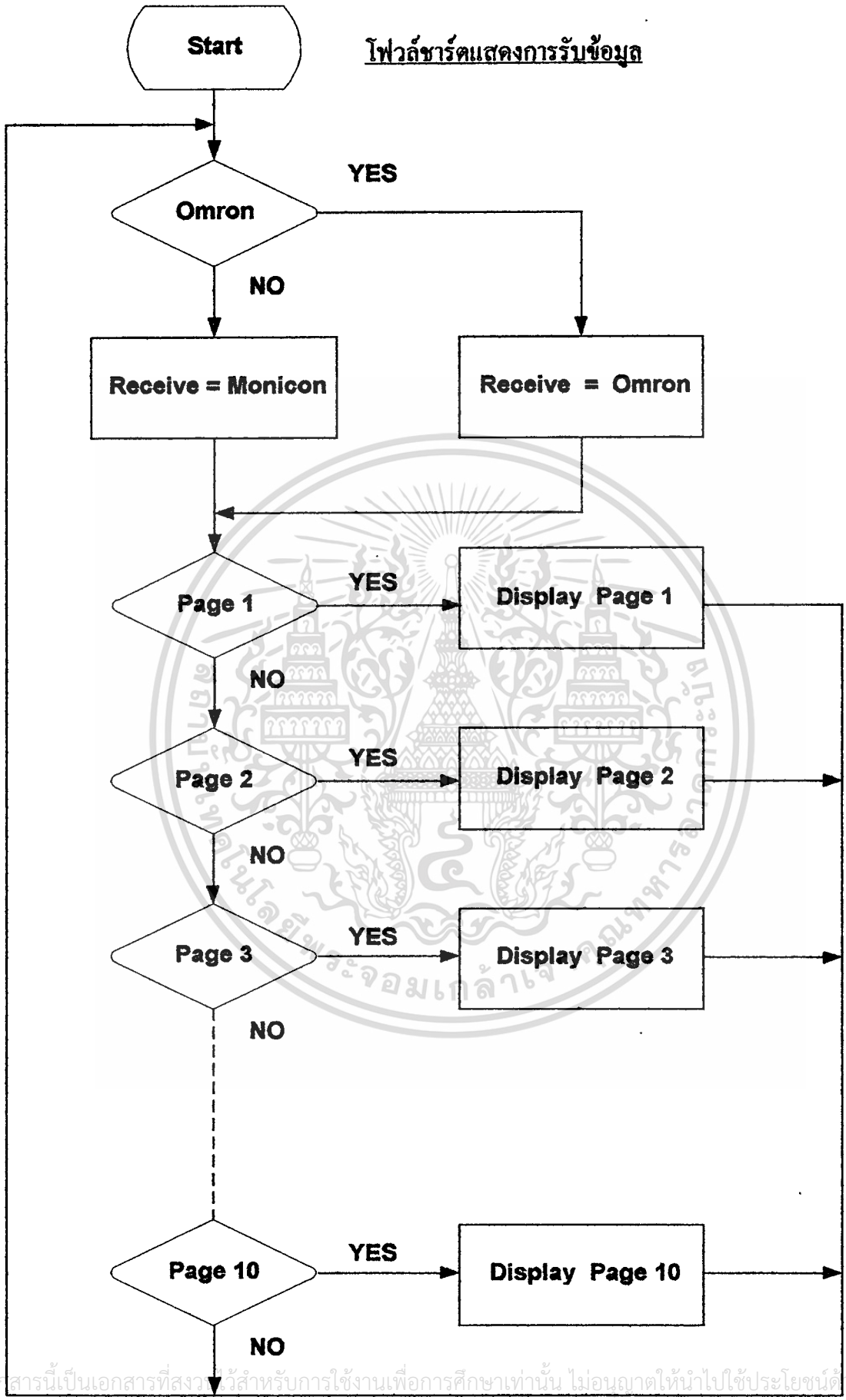
5.2.2. ตรวจสอบสายที่ทำการเชื่อมต่อระหว่างพีแอลซีและคอมพิวเตอร์พบว่าไม่ถูกต้องทำการแก้ไขให้ถูกต้องตามรูปแบบที่ได้กล่าวมาแล้วในส่วนของทฤษฎี

5.2.3. ตรวจสอบการกำหนดค่าการติดต่อที่คอมพิวเตอร์ และพีแอลซีพบว่าไม่ตรงกันทำการกำหนดให้ตรงกัน

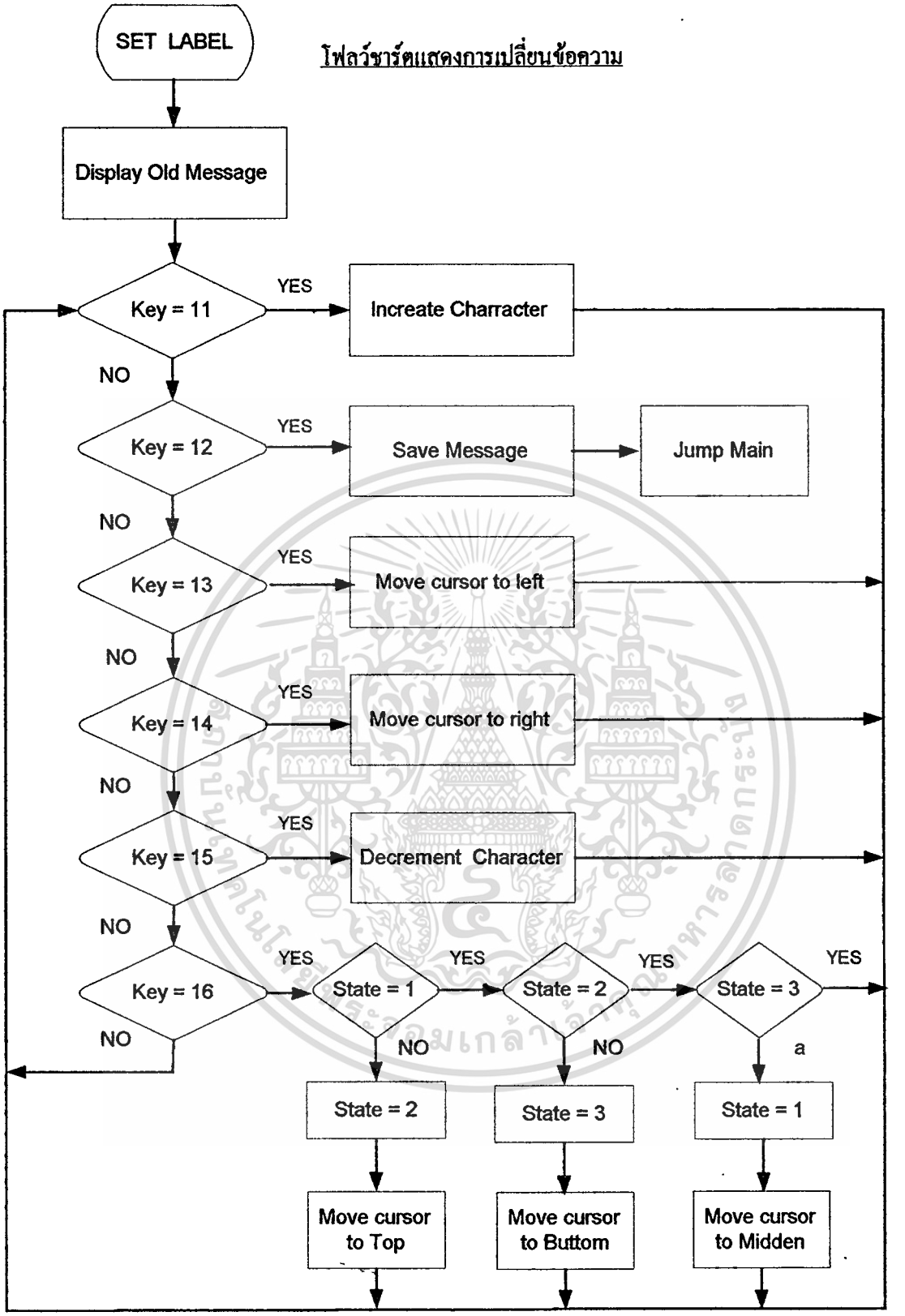
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

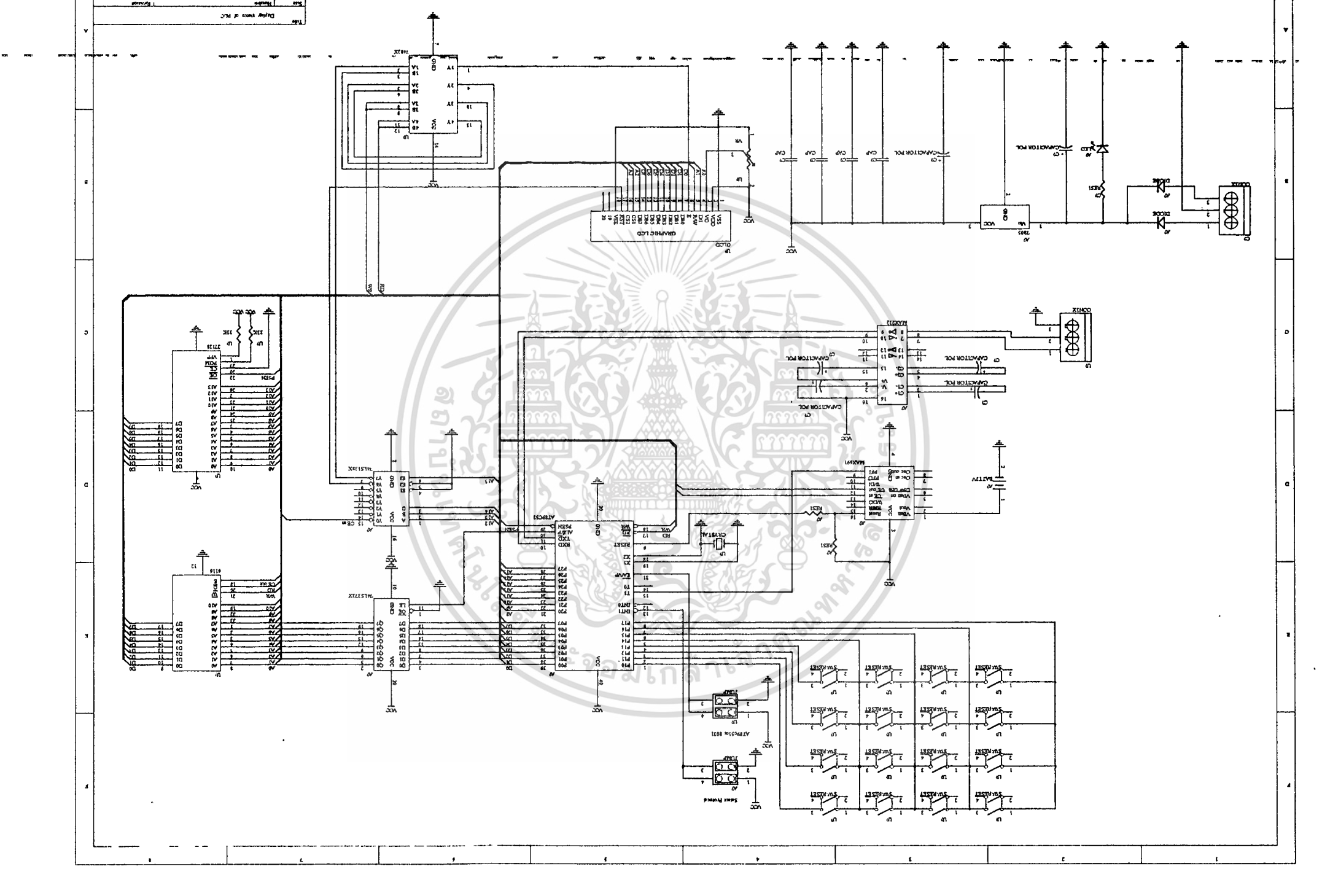
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

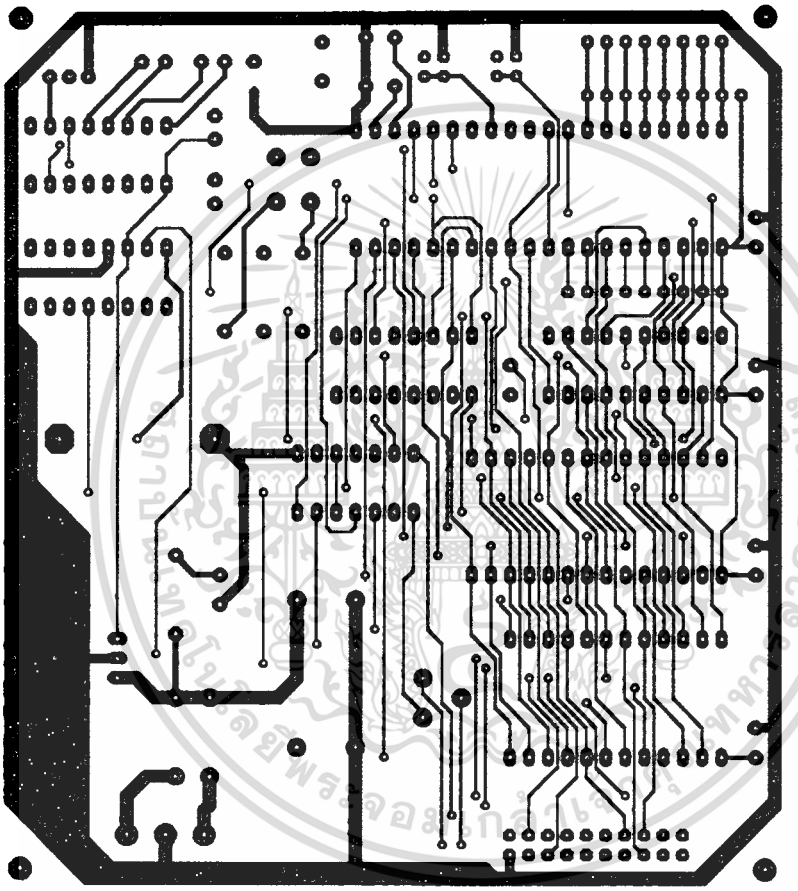


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

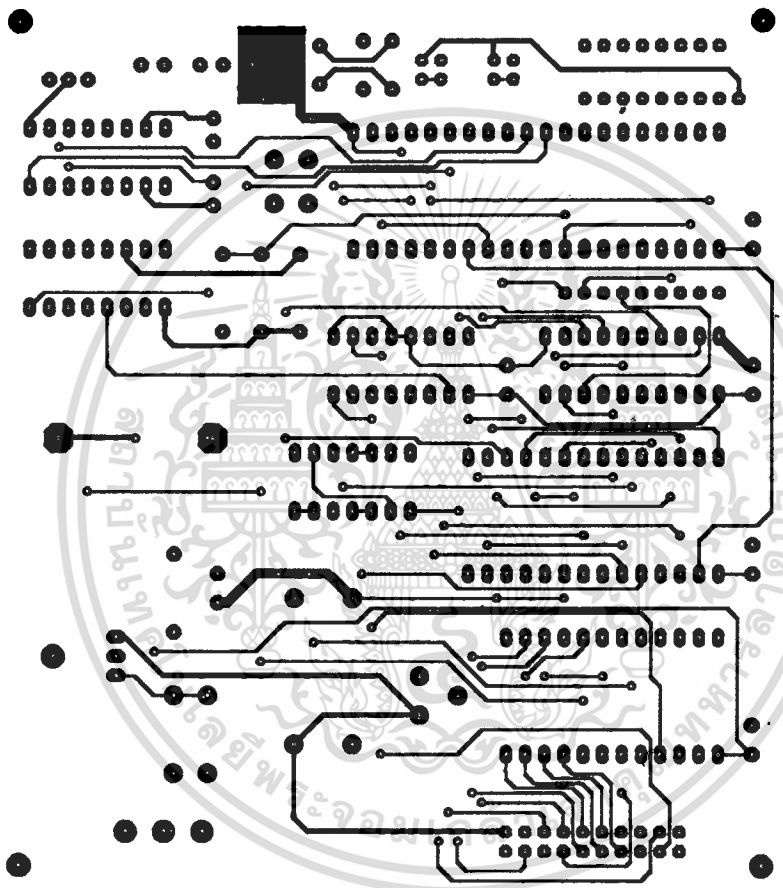


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ชิ้นนี้สามารถที่จะสำเร็จลงได้ก็ด้วยได้รับการช่วยเหลือของผู้มีอุปการะคุณหลายฝ่าย โดยอย่างยิ่ง พระคุณของบิดา มารดาผู้ซึ่งให้โอกาส และทุกสิ่งทุกอย่างแก่พวกเราคณะผู้จัดทำ ตลอดจนพระคุณของท่างอาจารย์ที่ปรึกษา ท่างอาจารย์เชื้อ นกอยู่ ที่คอยให้คำปรึกษา คำแนะนำและเครื่องมือรวมทั้งอุปกรณ์ที่ใช้ในการดำเนินการทดลอง และสุดท้ายก็คือกลุ่มเพื่อนๆทั้งหลายที่คอยให้คำแนะนำ แนวความคิด และกำลังใจในการทำงานแก่พวกเรา ทางคณะผู้จัดทำจึงขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้ด้วย

คณะผู้จัดทำ
ชวัชชัย ก้องกิตติตระกูล
สุธรรม ไชยมุติ
สุริยา แป้นเกิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

ไกรวุฒิ โรจน์ประเสริฐ. ไมโครโปรเซสเซอร์ 2.กรุงเทพมหานคร:เม็ดทรายพริ้นติ้ง,2539

สุพรรณ กุลพาณิชย์.PROGRAMMABLE CONTROLLER.กรุงเทพมหานคร:โรงพิมพ์ทิพย วิสุทธิ์,2533

วิวัฒน์ กิรานนท์. การสื่อสารข้อมูลดิจิตอล.สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง,2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้