สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบคอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน

(COMPUTERIZED HOME CONTROL SYSTEM)

โดย

1.นาย ธวัชชัย รุ่งสว่าง รหัสนักศึกษา 40012090
2.นาย พิพัฒน์ โชคชัยภักดี รหัสนักศึกษา 40012096
3.นาย ธีรชัย เล้าเจริญวัฒนากุล รหัสนักศึกษา 40013409

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต
สาขาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2542

เลขหมู่.....
เลขทะเบียน 36815
วัน, เดือน, ปี 29 ส.ค. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2542

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบคอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน

(COMPUTERIZED HOME CONTORL SYSTEM)

ผู้จัดทำ

1.นาย ธวัชชัย รุ่งสว่าง รหัสนักศึกษา 40012090

2.นาย พิพัฒน์ โชคชัยภักดี รหัสนักศึกษา 40012096

3.นาย ธีรชัย เล้าเจริญวัฒนากุล รหัสนักศึกษา 40013409

..... อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์ วิริยะ กองรัตน)

..... อาจารย์ที่ปรึกษา

(อาจารย์ อาจินต์ น่วมสำราญ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบคอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน

นายธวัชชัย รุ่งส่วาง
นายพิพัฒน์ โชคชัยภักดี
นาย ธีรชัย เล้าเจริญวัฒนากุล
ผศ.วิริยะ กองรัตน อาจารย์ที่ปรึกษา
อ.อาจินต์ น่วมสำราญ อาจารย์ที่ปรึกษา
ปีการศึกษา 2542

บทคัดย่อ

โครงงานนี้มีวัตถุประสงค์ในการควบคุมอุปกรณ์ไฟฟ้าภายในอาคาร เพื่อให้สะดวกต่อการดูแลและควบคุมจากศูนย์กลางเดียวกัน โดยมีคอมพิวเตอร์ซึ่งได้ทำการพัฒนาโปรแกรมมาใช้แสดงการทำงานของอุปกรณ์ไฟฟ้าและประมวลผลร่วมกับไมโครคอลโทรลเลอร์ 8051 โดยการสื่อสารจะผ่าน RS-232 ซึ่งลักษณะของงานจะสามารถควบคุมได้ทั้งจากส่วนของคอมพิวเตอร์ และส่วนของสวิตช์ ณ ตำแหน่งต่างๆของอาคาร การควบคุมเช่นนี้จะอาศัยการทำงานในลักษณะของสวิตช์สามทาง แล้วตรวจสอบผลการทำงานของอุปกรณ์ไฟฟ้าโดยชุดตรวจสภาวะการทำงาน ข้อมูลที่ได้จะถูกส่งกลับไปยังคอมพิวเตอร์เพื่อประมวลผล และแสดงสถานะของอุปกรณ์ไฟฟ้าในขณะนั้นให้ผู้ใช้ทราบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTERIZED HOME CONTROL SYSTEM

THAWATCHAI RUNGSAWANG
PIPAT CHOKCHAIPAKDEE
THEERACHAI LAOCHAROENWATTANAKUL
Asst.prof. VIRIYA KONGRATANA Advisor
ARJIN NURMSAMRAN Advisor
Year 1999

Abstract

The purpose of this project is to be able to control and manage electric systems in homes or buildings centrally. This can be achieved by developing a computer program that shows the usage of all electrical appliances.

This computer program,worked in conjunction with microcontroller 8051, will communicate through RS-232-C.The control can be done either from a computer unit or from 3-way electric switches in the buildings.The usage of electrical appliances will then be monitored and the result will be sent back to the computer unit for processing. Finally, users will be informed of the current status of those appliances.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ I

สารบัญ II

สารบัญรูป V

สารบัญตาราง VIII

บทที่ 1 บทนำ 1

1.1 กล่าวนำ 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 4

2.1 สถาปัตยกรรมของไมโครคอลโทรลเลอร์แบบชิพเดี่ยวตระกูล 8051 4

2.2 โครงสร้างพื้นฐานของไมโครคอลโทรลเลอร์ 4

2.3 ไทเมอร์ / เคาน์เตอร์ 5

2.4 การสื่อสารแบบอนุกรม 8

2.5 การอินเตอร์รัปต์ 15

2.6 ชิพสนับสนุน 8255 17

2.7 วงจรดาร์ลิงตัน 19

2.8 เดลไฟล์ 20

บทที่ 3 หลักการออกแบบการทำงาน 26

3.1 ลักษณะโครงสร้างของระบบ 27

3.2 การออกแบบซอฟต์แวร์ 27

3.3 การออกแบบฮาร์ดแวร์ 34

3.4 คอมพิวเตอร์ส่วนบุคคล 35

3.5 พอร์ตสื่อสารข้อมูลอนุกรม 36

3.6 ส่วนประมวลผล 37

3.7 ส่วนขยายพอร์ตอินพุต / และเอาท์พุต 39

3.8 การจัดวางตำแหน่งของหน่วยความจำและส่วนขยายพอร์ต 39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินพุต / เอาท์พุต (8255)
3.9 ลักษณะการต่อพอร์ตอินพุต / และเอาท์พุต 40
บทที่ 4 การทดลองและผลการทดลอง 44
4.1 การทดลองในส่วนของโปรแกรม 44
4.2 โปรแกรมการทดสอบไมโครคอลโทรเลอร์ 44
4.3 โปรแกรมการทดลองของโปรแกรมควบคุม 53
4.4 การทดลองในส่วนของฮาร์ดแวร์ 57
4.5 การทดสอบส่วนขยายพอร์ตอินพุต / เอาท์พุต (8255) 57
4.6 การทดสอบชุดตรวจสถานะ 58
4.7 การทดสอบวงจรขับรีเลย์ 59
บทที่ 5 การติดตั้งและขั้นตอนการใช้โปรแกรม 61
5.1 การติดตั้งโปรแกรม 61
5.2 การใช้งานโปรแกรม 64
บทที่ 6 สรุปวิจารณ์การทดลองและการประยุกต์ใช้งาน 68
6.1 การติดต่อระหว่างคอมพิวเตอร์กับไมโครคอลโทรลเลอร์ 68
6.2 การทำงานของส่วนอุปกรณ์ภายนอก 68
6.3 การทำงานในการควบคุมระบบแสงสว่างในอาคาร 69
6.4 แนวทางการประยุกต์ใช้งาน 69
ภาคผนวก
กิตติกรรมประกาศ
เอกสารอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 ตำแหน่งในแต่ละบิตของรีจิสเตอร์ TCON 6
รูปที่ 2.2 ตำแหน่งในแต่ละบิตของรีจิสเตอร์ TMOD 7
รูปที่ 2.3 ลักษณะสัญญาณการทำงานโหมด 0 10
รูปที่ 2.4 ลักษณะสัญญาณการทำงานโหมด 1 10
รูปที่ 2.5 ลักษณะสัญญาณการทำงานโหมด 2 11
รูปที่ 2.6 ลักษณะสัญญาณการทำงานโหมด 3 11
รูปที่ 2.7 ไดอะแกรมของขาสัญญาณ DB - 25 ตัวผู้ 13
รูปที่ 2.8 ไดอะแกรมของขาสัญญาณ DB - 9 ตัวผู้ 14
รูปที่ 2.9 แสดงรายละเอียดของรีจิสเตอร์ IE 15
รูปที่ 2.10 แสดงรายละเอียดของรีจิสเตอร์ IP 16
รูปที่ 2.11 แสดงบิตต่างๆของคำสั่งควบคุม 18
รูปที่ 2.12 วงจรดาร์ลิงตัน 20
รูปที่ 2.13 หน้าจอของเคลไฟล์โดยรวม 21
รูปที่ 2.14 หน้าต่างหลัก 22
รูปที่ 2.15 สปีดบาร์หน้าต่างหลัก 23
รูปที่ 2.16 หน้าต่างฟอร์ม 23
รูปที่ 2.17 หน้าต่างยูนิต(unit) 24
รูปที่ 2.18 หน้าต่างออบเจคอินสเปคเตอร์ 25
รูปที่ 3.1 ลำดับขั้นตอนการทำงาน 26
รูปที่ 3.2 การเชื่อมต่อของระบบ RS – 232 -C 36
รูปที่ 3.3 บล็อกไดอะแกรมของ EPROM 38
รูปที่ 3.4 บล็อกไดอะแกรมของ RAM 38
รูปที่ 3.5 แสดงตำแหน่งของหน่วยความจำและส่วนขยายพอร์ตอินพุต / เอาท์พุต 40
รูปที่ 3.6 แสดงวงจรการตรวจสถานะการทำงาน 41
รูปที่ 3.7 แสดงวงจรดาร์ลิงตันควบคุมรีเลย์ 43
รูปที่ 3.8 แสดงการต่อวงจรรวมของชุดอินพุต เอาท์พุตและชุดตรวจสถานะ 44
รูปที่ 4.1 แสดงลักษณะของสัญญาณข้อมูล 61H ที่ออกจากขา TXD 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 4.2 แสดงการอ่านค่าสัญญาณข้อมูล 61H ที่ออกจากขา TXD 47
รูปที่ 4.3 แสดงลักษณะของสัญญาณข้อมูล 41H ที่ออกจากขา TXD 48
รูปที่ 4.4 แสดงลักษณะของสัญญาณข้อมูล 45H ที่ออกจากขา TXD 48
รูปที่ 4.5 แสดงลักษณะของสัญญาณข้อมูล 61 H ในมาตรฐาน RS-232-C 49
รูปที่ 4.6 แสดงอ่านค่าสัญญาณข้อมูล 61H ที่ออกจากขา TXD ในมาตราฐาน RS - 232 – C 50
รูปที่ 4.7 แสดงลักษณะของสัญญาณข้อมูล 41H ในมาตราฐาน RS-232-C 50
รูปที่ 4.8 แสดงอุปกรณ์ของชุดตรวจสถานะของหลอดไฟ 53
รูปที่ 4.9 โปรแกรมที่ใช้ในการส่งข้อมูล 54
รูปที่ 4.10 แสดงโปรแกรมที่รับข้อมูล 54
รูปที่ 4.11 แสดงการทำงานของหลอดขณะปิดของส่วนควบคุมหลอดไฟ 55
รูปที่ 4.12 แสดงการทำงานของหลอดขณะเปิดของส่วนควบคุมหลอดไฟ 55
รูปที่ 4.13 แสดงการทำงานของหลอดขณะปิดของส่วนควบคุมเต้ารับ 56
รูปที่ 4.14 แสดงการทำงานของหลอดขณะเปิดของส่วนควบคุมเต้ารับ 56
รูปที่ 4.15 แสดงตำแหน่งการวัดแรงดันของวงจรตรวจสอบสถานะ 58
รูปที่ 5.1 แสดงการเข้าเมนู Seting เพื่อเลือก Control Panel 61
รูปที่ 5.2 แสดงการเลือกไอคอน BDE Administrator ของหน้าต่าง Control Panel 61
รูปที่ 5.3 เมนูของ Object ในหน้าต่าง BDE Administrator 62
รูปที่ 5.4 แสดงการกำหนดชนิดของ Data base เป็นแบบ STANDARD 62
รูปที่ 5.5 แสดงการ PATH เพื่อกำหนด Data base ไปไว้ที่ C:\project2\PC HOME 63
รูปที่ 5.6 เลือกเมนู Apply หลังจากกำหนดค่าตำแหน่งที่ PATH 63
รูปที่ 5.7 แสดงการ SAVE Data base 64
รูปที่ 5.8 แสดงหน้าต่างการเข้าโปรแกรมควบคุม 64
รูปที่ 5.9 แสดงหน้าต่างให้ผู้ใช้เลือกเมนูและเลือกส่วนควบคุม 65
รูปที่ 5.10 แสดงหน้าต่างที่ 3 ซึ่งควบคุมการทำงานของชั้นที่ 1 65
รูปที่ 5.11 แสดงหน้าต่างการ เปิด –ปิด โดยการตั้งเวลา 66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 5.12 แสดงส่วนการตั้งเวลาของเต้ารับ 66
รูปที่ 5.13 แสดงส่วนการตั้งเวลาของหลอดไฟ 67
รูปที่ 5.14 แสดงข้อความเตือน 67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 หน้าที่การทำงานต่างๆของขาในพอร์ต 3 5
ตารางที่ 2.2 แสดงโหมดการทำงานของไทเมอร์ 7
ตารางที่ 2.3 คุณสมบัติของ RS – 232 - C 12
ตารางที่ 2.4 ไดอะแกรมของขาสัญญาณ DB - 25 ตัวผู้ 13
ตารางที่ 2.5 ไดอะแกรมของขาสัญญาณ DB - 9 ตัวผู้ 14
ตารางที่ 3.1 แสดงรหัสสของตัวอักษร 27
ตารางที่ 4.1 แสดงรห้สแอสกี้ของอักษร A-R และ a-r 45
ตารางที่ 4.2 ผลการเปรียบเทียบแรงดันตกคร่อมแต่ละส่วนของวงจรตรวจสอบ สถานะ 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 กล่าวนำ

การนำอุปกรณ์ที่ใช้ช่วยอำนวยความสะดวกในการดำเนินชีวิตประจำวันนั้นจะสามารถพบเห็นได้มากมาย ซึ่งส่วนมากจะเป็นการควบคุมอุปกรณ์ต่างๆในลักษณะกึ่งอัตโนมัติ อันได้แก่ การควบคุมการทำงานของระบบต่างๆของอาคารให้มีประสิทธิภาพ โดยนำเอาระบบคอมพิวเตอร์มาประยุกต์ใช้ร่วมกับระบบต่างๆของอาคาร เช่น ระบบไฟฟ้า ระบบปรับอากาศ ระบบแจ้งเตือนเพลิงไหม้ ระบบลิฟต์และบันไดเลื่อน ระบบแสงสว่างและระบบเสริมอื่นๆ แล้วทำการตรวจสอบสถานะหรือสภาวะการทำงานของอุปกรณ์ต่างๆในระบบนั้น ซึ่งติดตั้งไว้เพื่ออำนวยความสะดวกหรือให้บริการแก่ผู้ใช้ โดยใช้คอมพิวเตอร์ควบคุมหรือเปิด-ปิดอุปกรณ์ต่างๆ ให้ทำงาน

การทำงานของโครงงานนี้จะเป็นการควบคุมการทำงานของอุปกรณ์ไฟฟ้าโดยนำไมโครคอลโทรลเลอร์มาช่วยในการอำนวยความสะดวก ซึ่งถูกออกแบบมาเพื่อให้ผู้ใช้สามารถสั่งงานได้สองทิศทางคือ จากทางคอมพิวเตอร์และจากทางสวิตช์ โดยข้อมูลที่ได้จากการสั่งงานทั้งสองส่วนจะถูกนำมาประมวลผลโดยไมโครคอลโทรลเลอร์(MCS-51) เพื่อทำการควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า โดยการส่งข้อมูลในระบบจะเป็นการส่งข้อมูลสื่อสารผ่านทางพอร์ตอนุกรมแบบมาตราฐาน RS-232-C ส่งต่อไปยังไมโครคอลโทรลเลอร์(MCS-51) จากนั้นจึงตรวจสถานะการทำงานของอุปกรณ์ไฟฟ้า แล้วจึงนำผลที่ได้จากการควบคุมดังกล่าวกลับมาแสดงยังหน้าจอมอนิเตอร์ของคอมพิวเตอร์

1.2 วัตถุประสงค์ของโครงงาน

โครงงานนี้มีวัตถุประสงค์เพื่ออำนวยความสะดวกและเป็นการส่งเสริมการอนุรักษ์พลังงาน การใช้ทรัพยากรพลังงานไฟฟ้าให้คุ้มค่าอีกรูปแบบหนึ่ง ซึ่งสามารถดูแลและควบคุมจากศูนย์กลางเดียวกันโดยนำอุปกรณ์อำนวยความสะดวกต่างๆ และอุปกรณ์ที่ติดตั้งเพื่อป้องกันชีวิตและทรัพย์สินกระจายติดตั้งทั่วไปภายในอาคาร ตามลักษณะการใช้งานของอุปกรณ์มารวมกันไว้ที่ศูนย์กลางควบคุมเดียวกัน ทำให้สะดวกต่อการควบคุม สั่งการอุปกรณ์ การตรวจสอบสภาวะการทำงานของอุปกรณ์ การตรวจวัดและรายงานผลการทำงานที่ผิดปกติของอุปกรณ์การประสานงานและดูแลรักษาอุปกรณ์ภายในอาคารโดยลักษณะของโครงงานจะเป็นการติดต่อกันระหว่างคอมพิวเตอร์กับไมโครคอลโทรลเลอร์(MCS-51) เพื่อทำการควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า โดยการส่งข้อมูลในระบบจะเป็นการส่งข้อมูลสื่อสารผ่านทางพอร์ตอนุกรมแบบมาตราฐาน RS-232-C แล้วนำผลที่ได้จากการควบคุมดังกล่าวกลับมาแสดงยังหน้าจอมอนิเตอร์ของคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

จากลักษณะของโครงงานดังที่กล่าวมา จะประกอบไปด้วยอุปกรณ์แต่ละส่วนที่มีหน้าที่ต่างๆกัน เพื่อทำการควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้าดังนี้

1.ไมโครคอลโทรลเลอร์ตระกูล MCS-51

ในส่วนของไมโครคอลโทรลเลอร์จะใช้รับข้อมูลที่ได้จากการสั่งงานจากสองทิศทางคือ จากทางคอมพิวเตอร์และจากทางสวิตช์ โดยส่งข้อมูลสื่อสารผ่านทางพอร์ตอนุกรมแบบมาตราฐาน RS-232-C แล้วนำมาประมวลผลเพื่อควบคุมให้ไอซีเบอร์ 8255 ส่งผลไปควบคุมรีเลย์เพื่อทำการเปิด-ปิดอุปกรณ์ไฟฟ้า

2. ชิพสนับสนุนประเภทอินพุต/เอาท์พุต

ชิพสนับสนุนดังกล่าวคือ 8255 เป็นอุปกรณ์ที่ทำหน้าที่ขยายอินพุต-เอาท์พุตพอร์ต ซึ่งจะประกอบด้วยพอร์ต 3 พอร์ต แต่ละพอร์ตมี 8 บิต ในโครงงานนี้ ใช้ชิพ 8255 จำนวน 1 ตัว เพื่อให้เพียงพอกับอุปกรณ์ไฟฟ้าในพื้นที่ที่ต้องการออกแบบ โดย 8255 จะถูกแบ่งให้มีหน้าที่เป็นอินพุตจำนวน 8 บิต และเอาท์พุตอีกจำนวน 16 บิต ซึ่งการกำหนดอินพุตและเอาท์พุตทั้ง 32 บิต จะต้องกำหนดในริจิสเตอร์ Control ก่อนการใช้งาน 8255

3.การสื่อสารระหว่างคอมพิวเตอร์กับไมโครคอลโทรลเลอร์ (MCS-51)

การสื่อสารระหว่างคอมพิวเตอร์กับไมโครคอลโทรลเลอร์(MCS-51) จะเป็นแบบอนุกรมมาตรฐาน RS-232-C ผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์ ซึ่งเป็นสื่อกลางในการติดต่อระหว่างอุปกรณทั้งสอง

4.โปรแกรมที่ใช้แสดงผลบนจอคอมพิวเตอร์

เป็นโปรแกรมที่เขียนขึ้นมาจากโปรแกรมเดลไฟล์(DELPHI) ใช้ในการเขียนกราฟฟิกเพื่อสร้างรูปแบบในการสั่งการควบคุมผ่านทางหน้าจอคอมพิวเตอร์และทำการเปรียบเทียบแอดเดรสของตำแหน่งที่ควบคุมมาแสดงผล

5. ส่วนควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า

เป็นส่วนควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า ซึ่งจะถูกควบคุมโดยไมโครคอลโทรลเลอร์ การควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้านั้น จะใช้อุปกรณ์ประเภทรีเลย์ซึ่งมีความสามารถในการแยกวงจร ทางด้านไฟฟ้าระดับแรงดัน220โวลท์ที่จ่ายให้กับอุปกรณ์ไฟฟ้าที่ต้องการควบคุมกับอุปกรณ์ขยายอินพุต/เอาท์พุตประเภท 8255 ที่ควบคุมโดยไมโครคอลโทรลเลอร์(MCS-51) ที่ระดับแรงดันทีทีแอล(TTL , 5 volt)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ส่วนวงจรทำการตรวจสอบสถานะของอุปกรณ์ไฟฟ้า

วงจรตรวจสอบสถานะนี้ เป็นตัวบอกถึงสถานะที่แท้จริงของอุปกรณ์ส่องสว่างว่ามีการเปิด-ปิดจริงหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 สถาปัตยกรรมของไมโครคอลโทรลเลอร์แบบชิพเดี่ยวตระกูล 8051

ไมโครคอลโทรลเลอร์แบบชิพเดี่ยว (Single Chip Microcontroler) คือไมโครคอมพิวเตอร์ขนาดเล็กที่บรรจุไว้ในแผงวงจรรวม(Integrated Circuit)เพียงชิพเดี่ยวซึ่งเหมาะสำหรับงานควบคุมอุปกรณ์อื่นๆแบบอัตโนมัติ และมีความสะดวกในการใช้งาน

2.2 โครงสร้างของไมโครคอลโทรลเลอร์

2.2.1 หน่วยประมวลผลกลางขนาด 8 บิต

2.2.2 หน่วยประมวลผลสำหรับข้อมูลแบบบิต (Boolean Processor)

2.2.3 ความสามารถในการอ้างตำแหน่งของรอม ได้ 64 กิโลไบต์

2.2.4 ความสามารถในการอ้างตำแหน่งของแรม ได้ 64 กิโลไบต์

2.2.5 หน่วยความจำแบบแรม ภายในจำนวน 128 ไบต์ ประกอบด้วย

1. รีจิสเตอร์แบงก์ 4 แบงก์ประกอบด้วยรีจิสเตอร์ขนาด 8 จำนวน 8 รีจิสเตอร์ (RO - R7)
2. มีหน่วยความจำจำนวน 16 ไบต์ ที่สามารถอ้างแอดเอรสเพื่อควบคุมการทำงานในระดับบิตได้
3. มีหน่วยความจำสำหรับใช้งานทั่วไป 80 ไบต์

2.2.6 พอร์ตอินพุต/พอร์ตเอาท์พุต แบบขนานจำนวน 32 บิต แบ่งเป็น 4 พอร์ตดังนี้

1. พอร์ต 0 (ขา 32-39) เป็นพอร์ตที่ใช้งานสองหน้าที่ หน้าที่แรกใช้เป็นอินพุต/เอาท์พุตพอร์ต ส่วนอีกหน้าที่หนึ่งนั้นใช้ควบคุมหน่วยความจำภายนอก เมื่อต้องการขยายระบบให้ใหญ่ขึ้นโดยจะให้สัญญาณที่มัลติเพลกซ์ระหว่างบัสแอดเดรสกับบัสข้อมูลออกมา (AD7-AD0)
2. พอร์ต 1 (ขา 1-8) ใช้เป็นอินพุต/เอาท์พุตพอร์ตอย่างเดียว ใช้สัญลักษณ์เรียงกันเป็น P1.0, P1.1 จนถึง P1.7 พอร์ตนี้ใช้เชื่อต่อกับอุปกรณ์ภายนอกหน้าที่เดียวเท่านั้น ยกเว้น 8032/8052 ที่ใช้ P1.0 และ P1.1 เป็นอินพุต/เอาท์พุต หรืออินพุตภายนอกของไทเมอร์ชุดที่สาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. พอร์ต 2 (ขา 21-28) เป็นพอร์ตที่ใช้งานสองหน้าที่เช่นกันหน้าที่แรกเป็นอินพุต/เอาท์พุต หน้าที่สองใช้ควบคุมหน่วยความจำภายนอก โดยให้สัญญาณแอดเดรสไบต์สูง (A8-A15) ออกมา
4. พอร์ต 3 (ขา10-17) เป็นพอร์ตที่ใช้งานสองหน้าที่เช่นกัน หน้าที่แรกเป็นอินพุต/เอาท์พุตหน้าที่ที่สองแยกออกได้หลายฟังก์ชั่นดังนี้

บิต	ชื่อ	ใช้งานเป็น
P3.0	RXD	รับข้อมูลแบบอนุกรม(serial data input)
P3.1	TX	ส่งข้อมูลแบบอนุกรม(serial data input)
P3.2	INTO	อินเตอร์รัปต์ภายนอกหมายเลข 0
P3.3	INT1	อินเตอร์รัปต์ภายนอกหมายเลข 1
P3.4	T0	รับสัญญาณจากภายนอกของไทเมอร์หมายเลข 0
P3.5	T1	รับสัญญาณจากภายนอกของไทเมอร์หมายเลข 1
P3.6	WR	สัญญาณเขียนหน่วยความจำภายนอก(write)
P3.7	RD	สัญญาณอ่านหน่วยความจำภายนอก(read)

ตารางที่ 2.1 หน้าที่การทำงานต่างๆของขาในพอร์ต 3

2.2.7 วงจร Timer/Counter ขนาด 16 บิต จำนวนสองวงจร

2.2.8 วงจรสื่อสารแบบอนุกรมแบบฟูลดูเพลกซ์ (Full Duplex) เรียกว่า SBUF

2.2.9 วงจรควบคุมการอินเตอร์รัปต์จากแหล่งกำเนิดสัญญาณ 5 แหล่ง จากภายใน 3 แหล่ง จากภายนอก 2 แหล่ง

2.2.10 ส่วนของออสซิลเลเตอร์(Oscillator)และวงจรสร้างสัญญาณนาฬิกาอยู่ภายใน

2.3 ไทเมอร์/เคาน์เตอร์

8051 มีไทเมอร์/เคาน์เตอร์ขนาด 16 บิต 2 ตัวคือ T0 และ T1 ส่วนเคาน์เตอร์จะแบ่งเป็นรีจิสเตอร์ 8 บิต 2 ตัวคือ TH0 และTL0 และรีจิสเตอร์ T1 ซึ่งมีขนาด 16 บิตเช่นกันจะประกอบด้วยรีจิสเตอร์ขนาด 8 บิต 2 ตัวถึง TH1 และ TL1

การควบคุมการทำงานจะสั่งผ่านรีจิสเตอร์ TCON (Timer control special function register) และ TMOD (Timer mode control special function register) ซึ่งมีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-รีจิสเตอร์ TCON

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

รูปที่ 2.1 ตำแหน่งในแต่ละบิตของรีจิสเตอร์ TCON

TF1 หมายถึง โอเวอร์โฟลว์แฟล็กของไทเมอร์ 1 มีค่าเป็น 1 เมื่อเกิดโอเวอร์โฟลว์ขึ้น (หมายถึงนับจนครบค่าสูงสุดคือทุกบิตในไทเมอร์ 1 เป็น 1 ทั้งหมดแล้ววนรอบกลับมาเป็น 0 ทั้งหมดใหม่อีกครั้งหนึ่ง) TF1จะถูกเคลียร์ให้เป็น 0 เมื่อไมโครคอลโทรลเลอร์กลับจากการทำงานในโปรแกรมบริการอินเตอร์รัปต์(interupt service routine)

TR1 หมายถึง บิตที่ใช้บังคับให้ไทเมอร์ 1 ทำงานหรือหยุดทำงาน ถ้า TR1 เป็นลอจิก 1 ไทเมอร์จะทำงาน ถ้าเป็นลอจิก 0 ไทเมอร์ 1 หยุดทำงาน(หยุดทำงานหมายถึง หยุดนับพัลส์แต่มิใช่รีเซ็ตค่าในไทเมอร์ 1 ให้เป็น 0000 ใหม่)

TF0 หมายถึง โอเวอร์โฟลว์แฟล็กของไทเมอร์ 0(การทำงานต่างๆเป็นเช่นเดียวกับTF1มีค่าเป็น 1 เมื่อเกิดโอเวอร์โฟวจ์ขึ้นและจะถูกเคลียร์เป็น 0 เมื่อไมโครคอลโทรลเลอร์กลับจากการทำงานในส่วนของโปรแกรมบริการอินเตอร์รัปต์

TR0 หมายถึง บิตที่ใช้บังคับให้ไทเมอร์ 0 ทำงานหรือหยุดทำงาน ถ้าเป็น 1 จะเป็นการให้ไทเมอร์ทำงานและถ้าเป็น 0 จะเป็นการให้ไทเมอร์ 0 หยุดนับแต่ค่าภายในจะไม่เป็น 0

TE1 หมายถึง การยอมให้อินเตอร์รัปต์จากภายนอก โดยผ่านทางพอร์ต 3 ขา 3.3(INT1) บิตนี้จะถูกเซ็ตเป็น 1 เมื่อสัญญาณจากภายนอกเข้ามากระตุ้นที่ขา 3.3 โดยจะทำการอินเตอร์รัปต์เมื่อสัญญาณเปลี่ยนสภาวะจากลอจิก 1 มาเป็นลอจิก 0 ตำแหน่ง เริ่มต้นของโปรแกรมบริการอินเตอร์รัปต์อยู่ที่แอดเดรส์ 0013H เมื่อทำคำสั่ง RET1 (Return From interrupt) IE1 จะถูกเคลียร์โดยอัตโนมัติ จะเห็นว่าบิตนี้ไม่มีส่วนเกี่ยวข้องใดๆกับไทเมอร์/เคาน์เตอร์

IT1 หมายถึง การควบคุมสัญญาณที่เข้ามากระตุ้นที่ขา 3.3 ของพอร์ต 3ว่าเป็นการกระตุ้นแบบใดจึงจะเกิดการอินเตอร์รัปต์ขึ้น ถ้า IT1 = 0 เป็นการกระตุ้นโดยลอจิก 0 (สัญญาณเป็น 0) ข้อแตกต่างคือ ถ้าเกิดการอินเตอร์รัปต์ขึ้นแล้วไปกระทำโปรแกรมบริการอินเตอร์รัปต์จนเสร็จสิ้น สมมติว่า กลับมาที่เวลาซึ่งสัญญาณยังเป็นลอจิก 0 อยู่ ในลักษณะนี้จะไม่เกิดอะไรขึ้น แต่ถ้าให้ IT1=0 ซึ่งเป็นการกระตุ้นที่ลอจิก 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IE0 หมายถึง การยอมให้อินเตอร์รัปต์จากภายนอก โดยผ่านทางพอร์ต 3 ขา 3.2 (INT0) การอินเตอร์รัปต์ที่ขานี้ สัญญาณต้องเป็นการกระตุ้นแบบของขอบขาลงหรือสามารถโปรแกรมได้ เช่นเดียวกับ IE1 ตำแหน่งแอดเดรสเริ่มต้นของโปรแกรมบริการอินเตอร์รัปต์อยู่ที่แอดเดรส 0003H

IT0 หมายถึง การควบคุมสัญญาณที่เข้ามาอินเตอร์รัปต์ IE0 เช่นเดียวกับ IT1

-รีจิสเตอร์ TMOD

7	6	5	4	3	2	1	0
gate	C/T	M1	M0	gate	C/T	M1	M0

รูปที่ 2.2 ตำแหน่งในแต่ละบิตของรีจิสเตอร์ TMOD

gate เป็นการอีนาเบิ้ลออร์เกตเพื่อควบคุมให้ไทเมอร์ 1 หรือ 0 ทำงานหรือหยุดทำงาน ถ้าบิตนี้เป็นลอจิก 1 ไทเมอร์จะทำงาน แต่ถ้า TR1/0 ใน TCON รีจิสเตอร์ ถูกเซ็ตเป็น 1 และสัญญาณที่ต่ออยู่กับขา INT 1/0 เป็นลอจิก1

C/T ถ้าเป็นลอจิก 1 เป็นการเลือกให้ไทเมอร์ 1 หรือ 0 ทำงานในลักษณะของเคาน์เตอร์ทำการนับพัลส์จากภายนอกที่ป้อนเข้ามาโดยผ่านทางขา 3.5 (T1)หรือ 3.4 (T0) แต่ถ้าเป็นลอจิก 0 เป็นการเลือกให้ไทเมอร์ 1 หรือ 0 ทำงานในลักษณะของไทเมอร์ โดยการนับพัลส์ที่ได้จากภายในตัวของ 8051

M1 จะเป็นบิตที่ใช้ควบคุมให้ไทเมอร์/เคาน์เตอร์ทำงานในโหมดใดบิตที่ 1

M0 จะเป็นบิตที่ใช้ควบคุมให้ไทเมอร์/เคาน์เตอร์ทำงานในโหมดใดบิตที 0

โดยมีรายละเอียดดังตารางที่2.2

M1	M2	โหมด
0	0	0
0	1	1
1	0	2
1	1	3

ตารางที่ 2.2 แสดงโหมดการทำงานของไทเมอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดการทำงานของไทเมอร์/เคาน์เตอร์

โหมด 0

โหมด 0 ใช้ไทเมอร์ขนาด 13 บิต โดยใช้ไทเมอร์ไบต์สูง(THX) 8 บิต๖ฮอนุกรมกับไทเมอร์ไบต์ต่ำ(TLX)อีก 5 บิต (บิต 0-4) รวมทั้งสิ้น 13 บิตอีกสามบิตของไบต์ต่ำ(TLX)จะไม่ถูกนำมาใช้

โหมด 1

ไทเมอร์โหมด 1 ขนาด 16 บิ9ทำงานเหมือนกับโหมด 0 ต่างกันเพียงใช้เต็ม 16 บิตเท่านั้น สัญญาณนาฬิกาป้อนให้กับรีจิสเตอร์ไทเมอร์ที่เกิดขึ้นจาก TLX และ THX ประกอบกัน เมื่อได้รับสัญญาณนาฬิกาไทเมอร์จะเริ่มนับจาก 0000H ไปจนถึงค่า FFFFH และค่าของโอเวอร์โฟลว์จะเกิดขึ้นเมื่อเกิดการนับเปลี่ยนจาก FFFFH ไปเป็น 0000H โอเวอร์โฟลว์แฟล็กจะเซ็ตค่าที่ใส่ลงไปสามารถกระทำได้โดยซอฟแวร์โดยบิตที่มีความสำคัญสูงสุด (MSB) จะอยู่ในบิตที่ 7 ของ THX และบิตที่มีความสำคัญน้อยสุดจะอยู่ในบิต 0 ของ TLX

โหมด 2

โหมด 2 ใช้ไทเมอร์ไบต์ต่ำ(TLX) ทำงานเป็นไทเมอร์ขนาด 8 บิต ในขณะที่ไทเมอร์สูง (THX) ใช้เก็บค่าที่ต้องการให้โหลดใหม่เอาไว้เมื่อเกิดโอเวอร์โฟลว์ขึ้น(จากFFH-00H) นอกจากแฟล็กของไทเมอร์จะเซ็ตแล้วค่าใน(THX)จะถูกโหลดเข้าสู่(TLX)ด้วย

โหมด 3

โหมด 3 เป็นการแยกใช้ไทเมอร์อิสระจากกัน โดยไทเมอร์ 0 ในโหมด 3 จะถูกแยกออกเป็นไทเมอร์ขนาด 8 บิต 2 ชุดคือ TL0 และ TH0 ไทเมอร์ทั้งสองชุดจะแยกทำงานอิสระจากกันแฟล็ก TF0 จะเปิดเมื่อ TL0 เกิดการโอเวอร์โฟล้วและแฟล็ก TF1 จะใช้เมื่อ TH0 เกิดโอเวอร์โฟลว์

ไทเมอร์ 1 จะหยุดในโหมด 3 แต่สามารถทำงานได้โดยการสั่งให้ทำงานในโหมดอื่นแต่มีข้อกำหนดว่าแฟล็กโอเวอร์โฟลว์ของไทเมอร์ 1 จะไม่มีผลต่อตัวไทเมอร์ 1 เมื่อเกิดโอเวอร์โฟลว์เนื่องจากแฟล็กโอเวอร์โฟลว์ของไทเมอร์ 1 ต่ออยู่กับ TH0 (เมื่อเลือกใช้โหมด3)การกระทำในลักษณะนี้เหมือนกับว่ามีไทเมอร์ขนาด 8 บิตให้ใช้ถึง 3 ชุดโดยใช้ไทเมอร์ 0 ในโหมด 3 ทำงานที่ต้องการ ส่วนไทเมอร์ 1 อาจใช้เป็นตัวกำเนิดอัตราบอด(buad rate gernerator) ในการใช้านพอร์ตอนุกรมได้หรือใช้ในงานอื่นที่ไม่เกี่ยวข้องกับอินเตอร์รัปต์

2.4 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมจะสามารถแบ่งออกเป็น 3 ลักษณะตามการรับส่งข้อมูลดังนี้

2.4.1 การรับส่งทางเดียว (Simplex)การสื่อสารข้อมูลที่สามารถส่งข้อมูลไปได้ในทิศทางเดียวเท่านั้นไม่สามารถส่งสวนทิศทางไม่ว่าจะเป็นเวลาใดก็ตาม(ส่งสวนทางเดิมไม่ได้และส่งพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ได้ เช่น การส่งข้อมูลจากคอมพิวเตอร์ไปยังจอภาพส่งได้เฉพาะจากคอมพิวเตอร์ไปยังจอภาพ จอภาพไม่สามารถส่งข้อมูลมาให้คอมพิวเตอร์ได้)

2.4.2 การรับส่งแบบผลัดกันส่ง (Half Duplex)มีคุณสมบัติสามารถรับส่งข้อมูลได้ แต่ว่าจะต้องสลับกันส่ง ไม่สามารถที่จะส่งพร้อมกันได้อุปกรณ์ที่ใช้ในการสื่อสารแบบ Half Duplex ลักษณะการทำงาน เมื่ออีกฝ่ายหนึ่งส่งอีกฝ่ายหนึ่งจะทำหน้าที่รับ จนกระทั่งฝ่ายแรกจบการส่งฝ่ายหลังก็จะทำการส่งสลับกันเรื่อยไป ซึ่งไม่สามารถส่งพร้อมกันได้

2.4.3 การรับส่งแบบสวนทางกันได้พร้อมกัน (Full Duplex) การสื่อสารแบบนี้ผู้รับและผู้ส่งสามารถรับและส่งพร้อมๆ กันในเวลาเดียวกัน ไม่จำเป็นต้องรอให้ฝ่ายหนึ่งจบเสียก่อนอย่างในการสื่อสารแบบ Half Duplex เช่นการพูดโทรศัพท์ การสื่อสารข้อมูลระหว่างคอมพิวเตอร์สองเครื่องมีใช้ทั้งแบบ Half Duplex และ Full Duplex ซึ่งขึ้นอยู่กับลักษณะของงานและการเชื่อมต่อ ในโครงงานนี้จะแบ่งการสื่อสารออกเป็น 2 มาตราฐานคือ

1. การสื่อสารแบบอนุกรมของ 8051
2. มาตราฐาน RS-232-C

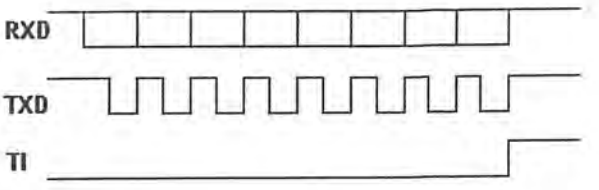
2.4.3.1 การสื่อสารแบบอนุกรมของ 8051

พอร์ตสื่อสารอนุกรมของไมโครคอลโทรลเลอร์ มีโครงสร้างการทำงานในแบบที่เรียกว่าฟูลดูเพล็กซ์ (Full Duplex) ในการรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกัน โดยทางด้านส่งใช้ขา TXD (พอร์ท 3.1) ทางด้านรับใช้ขา RXD (พอร์ต 3.0) SBUF ใช้เป็นบัฟเฟอร์สำหรับรับและส่งข้อมูลอนุกรม

พอร์ตสื่อสารอนุกรมของ 8051 สามารถโปรแกรมการทำงานได้หลายโหมดด้วยการโดยเลือกที่บิต SM1 และ SM0 ซึ่งอยู่ในรีจิสเตอร์ควบคุม SCON การทำงานทั้ง 4 โหมด ของพอร์ตสื่อสารอนุกรมมีดังนี้

โหมด 0 : พอร์ตสื่อสารอนุกรม 8 บิต โดยการส่งจะเลื่อนออกทีละบิตโดยส่งบิต DO ออกไปก่อนทางขา RXD และไม่มีการส่ง start bit แต่จะส่ง shift clock ทางขา TXD (ความเร็ว1/12 เท่าของ CPU Clock) การส่งข้อมูลแบบอนุกรมในโหมด 0 จะกำหนดข้อมูลใน SCON บิต SM0 และ SM1 มีค่าเป็น 0 ทั้งคู่ เป็นการกำหนดให้รีจิสเตอร์ รับหรือส่งข้อมูลขนาด 8 บิต โดยใช้ขา RXD เพียงขาเดียวเท่านั้นที่ต่อกับสายสัญญาณที่ใช้ส่งข้อมูลภายนอก ขา TXD จะถูกต่อกับสายกำเนิดสัญญาณนาฬิกาอยู่ภายในท่อไอซีที่ใช้ส่งพัลส์ออกมาภายนอก เพื่อใช้กำหนดความถี่อ้างอิงในการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ลักษณะสัญญาณการทำงานโหมด 0

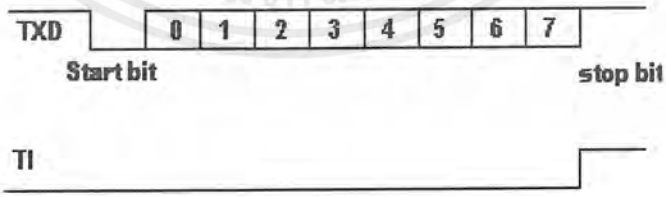
โหมด 1 : การส่งข้อมูลในโหมดนี้จะต้องมีการกำหนดอัตราบอดให้กับไมโครคอลโทรลเลอร์ โดยใช้ไทเมอร์ 1 เป็นตัวสร้างอัตราบอดซึ่งจะใช้ในโหมด 2 เพราะสามารถโหลดค่าเข้าใน TH1 ได้โดยอัตโนมัติ โดยค่าที่กำหนดสามารถคำนวณได้ดังนี้

$$f_{buad} = \frac{2^{smod}}{32_d} \times \frac{Oscillator\ frequency}{12_d \times [256_d - (TH1)]} \qquad (2.1)$$

เมื่อ fbuad = อัตราบอดที่ใช้ส่งข้อมูล (บิตต่อวินาที)

TH1 = ค่าที่กำหนดใน TH1 ของไทเมอร์ 1

ลักษณะของสัญญาณในโหมดที่ 1 จะประกอบไปด้วยบิตรวมทั้งหมด 10 บิต เป็นบิตข้อมูล (data bit) 8 บิต เป็นบิตเริ่มต้น(start bit) 1 บิต และ บิตหยุด(stop bit) 1 บิต และสามารถเปลี่ยนแปลงความเร็วในการส่งข้อมูลไป โดยขึ้นกับบิต SMOD ใน PCON และอัตราโอเวอร์โฟลว์ของ Timer 1 จากสูตรการคำนวณถ้า SMOD เป็น 0 จะเป็นความถี่ปกติ แต่ถ้าหากเป็น 1 ความถี่จะเพิ่มขึ้นเป็นสองเท่า



รูปที่ 2.4 ลักษณะสัญญาณการทำงานโหมด 1

โหมด 2 : ในโหมดนี้จะต้องมีการกำหนดอัตราบอดเช่นกัน โดยการสื่อสารในโหมดนี้มักจะใช้เป็นสื่อสารระหว่างไมโครคอลโทรลเลอร์ด้วยกันเอง การคำนวณหาอัตราบอดสามารถคำนวณได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

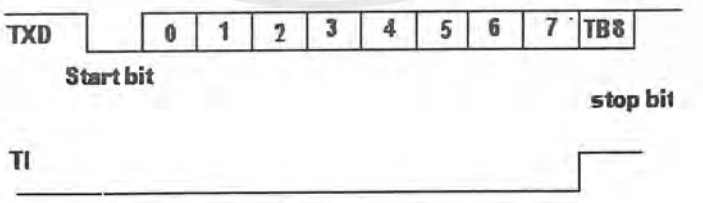
$$f_{buad} = \frac{2^{smod}}{64_d} \times Oscillator\ frequency \qquad (2.2)$$

เมื่อ fbuad = อัตราบอดที่ใช้ส่งข้อมูล (บิตต่อวินาที)

ลักษณะของสัญญาณของโหมดที่ 2 ประกอบด้วยบิตสื่อสารทั้งหมด 11 บิต ใช้เป็นบิตข้อมูล 9 บิต เป็นบิตเริ่มต้น 1 บิตและเป็นบิตหยุด 1 บิต ความเร็วในการรับส่งข้อมูลเท่ากับ 1/32 และ 1/64 ของCPU Clock โดยขึ้นกับบิต ใน SMOD ใน PCON ดังในสมการที่ 2.2

รูปที่2.5 ลักษณะสัญญาณการทำงานโหมด 2

โหมด 3 : การทำงานของโหมดนี้จะเหมือนกับโหมด 2 ทุกประการ แต่อัตราบอดจะหาได้แน่นอนกว่าในโหมด 2 โดยการคำนวณอัตราบอดจะเหมือนกับโหมด 1 ดังสมการที่ 2.1 และใช้ไทเมอร์ 1 มาทำการสร้างความถี่เช่นเดียวกับโหมด 1 ลักษณะของสัญญาณจะเหมือนกับโหมด 2 คือประกอบด้วยบิตสื่อสารทั้งหมด 11บิต UART โดยเป็นบิตข้อมูล 9 บิต เป็นบิตเริ่มต้น 1 บิตและเป็นบิตหยุด 1 บิตเหมือนโหมด 2 ยกเว้นอัตราความเร็วจะขึ้นกับบิต SMODในPCON และอัตราโอเวอร์โฟลว์ของ Timer 1 สำหรับ 8051 หรือขึ้นกับอัตราโอเวอร์โฟลว์ของ Timer 2 สำหรับ 80C154D



รูปที่ 2.6 ลักษณะสัญญาณการทำงานโหมด 3

จากรูปการส่งข้อมูลทั้ง 4 โหมดจะสังเกตเห็นว่า TI จะมีการเซ็ตเป็น 1 ก็ต่อเมื่อข้อมูลบิตที่ 7 ส่งออกไปยังพอร์ตอนุกรมเรียบร้อยแล้ว ซึ่งบิต TI จะเป็นบิตที่มีความสำคัญมากในการรับและส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลผ่านพอร์ตอนุกรม เนื่องจากการส่งข้อมูลทุกครั้งจะต้องตรวจสอบ TI เสมอ โดยการตรวจสอบจะต้องใช้ซอฟต์แวร์เป็นตัวตรวจสอบ

2.4.3.2 การสื่อสารแบบอนุกรมมาตรฐาน RS-232-C

การรับส่งข้อมูลแบบอนุกรมที่มีการใช้งานอยู่ในปัจจุบัน ได้มีการกำหนดมาตราฐานในการรับส่งข้อมูลไว้หลายแบบด้วยกันเพื่อที่จะทำให้อุปกรณ์ที่ผลิตจากผู้ผลิตที่แตกต่างกันสามารถทำงานร่วมกันได้ มาตราฐานการรับส่งข้อมูลที่ได้รับความนิยมเป็นอย่างมากคือ การรับส่งข้อมูลแบบอนุกรมตามมาตราฐาน RS-232-C ซึ่งมาตราฐานนี้เป็นมาตราฐานที่ใช้ในสำหรับการเชื่อมต่อกับโมเด็ม (MODEM,Modulator/Demodulator) ซึ่งเป็นอุปกรณ์ที่ใช้ในการสื่อสารระหว่างคอมพิวเตอร์ผ่านทางสายโทรศัพท์และอัตราการรับส่งข้อมูลถูกจำกัดให้มีค่าค่อนข้างต่ำ มาตราฐาน RS-232-C นี้ได้มีการออกแบบให้มีโครงสร้างในการรับส่งข้อมูลแบบจุดต่อจุดเท่านั้น คุณลักษณะของสัญญาณทางไฟฟ้าที่ถูกใช้งานมี 2 ลักษณะคือ SPACE แสดงถึงไบนารี 0 หรือแรงดันไฟฟ้าบวกและ MARK แสดงถึงไบนารี 1 หรือแรงดันไฟฟ้าลบ แรงดันไฟฟ้าบวก (สถานะ SPACE) อยู่ระหว่าง +5 ถึง +15 โวลต์ สำหรับเอาท์พุต และระหว่าง +3 ถึง +15 โวลต์ สำหรับอินพุต ซึ่งความแตกต่างนี้มีไว้เผื่อกรณีที่แรงดันไฟฟ้าสูญหายเนื่องจากความยาวของสายสัญญาณ ในทำนองเดียวกัน แรงดันไฟฟ้าลบ (สถานะ MARK) ถูกกำหนดไว้ระหว่าง -5 ถึง -15 โวลต์ สำหรับเอาท์พุต และ –3 ถึง –15 โวลต์ สำหรับอินพุต

สังเกตว่า ถ้าให้สายสัญญาณยาวเกินไป ระดับไฟฟ้าจะตกลงเกินขอบเขตที่ยอมได้ นอกจากนี้ ประจุไฟฟ้าที่เกิดขึ้นจะมีผลกับคุณภาพของสัญญาณ โดยการเปลี่ยนสถานะจากแรงดันไฟฟ้าบวกไปลบไม่ชัดเจน ซึ่ง RS-232-C ไม่ได้มุ่งหมายให้นำไปใช้กับระยะทางไกล และโดยทั่วไป 50 ฟุต เป็นระยะทางไกลที่สุดในการใช้สายปกติที่อัตราการส่งข้อมูลปกติ ถ้าหากอุปกรณ์อยู่ห่างกันมากอาจจะต้องใช้โมเด็ม หรือใช้มาตราฐานอื่นเข้ามาช่วย

คุณสมบัติของ RS-232-C

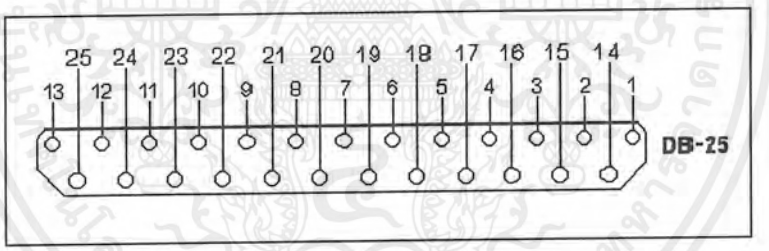
ตารางที่ 2.3 คุณสมบัติของ RS-232-C

พารามิเตอร์	RS-232-C
โหลดการทำงาน	Single-ended
จำนวนตัวรับและตัวส่ง	1 ตัวรับ 1 ตัวส่ง
ความยาวของคู่สายสัญญาณ (ฟุต)	50
อัตราการส่งข้อมูล(บิตต่อวินาที)	20K

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Maximum common Mode voltage	±25V
Driver output	±5V ต่ำสุด ±15V สูงสุด
Driver load Ω	3K ถึง 7K
Driver slew rate	30V/μs
กระแสลิมิตเมื่อเอาท์พุตลัดวงจร	500mA ลัดกับ VccกับGnd
ค่าความต้านทานเอาท์พุตของตัวส่ง Ω	NA (Power ON) 300 (Power Off)
ค่าความต้านทานอินพุตของตัวรับ Ω	3K ถึง 7K
ความไวของตัวรับ	±3V

รายละเอียดขาสัญญาณของหัวต่อ DB-25 ตัวผู้



รูปที่ 2.7 ไดอะแกรมของขาสัญญาณ DB-25 ตัวผู้

DB-25 แบบหัวต่อตัวผู้ (MALE)

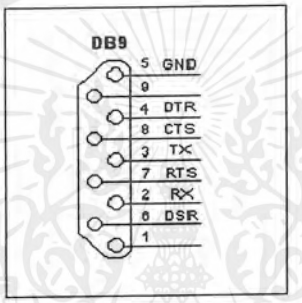
ตารางที่ 2.4 ไดอะแกรมของขาสัญญาณ DB-25 ตัวผู้

ขา	สัญญาณ
1	GF (Frame Ground)
2	TD (Transmitted Data)
3	RD (Receive Data)
4	RTS (Request To Send)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5	CTS (Clear To Send)
6	DSR (Data Set Ready)
7	SG (Signal Ground)
8	CD (Carrier Detect)
20	DTR (Data Terminal Ready)

รายละเอียดขาสัญญาณของหัวต่อ DB-9 ตัวผู้



รูปที่ 2.8 ไดอะแกรมของขาสัญญาณ DB-9 ตัวผู้

DB-9 แบบหัวต่อตัวผู้(MALE)

ตารางที่ 2.5 ไดอะแกรมของขาสัญญาณ DB-9 ตัวผู้

ขา	สัญญาณ
1	CD (Carrier Detect)
2	RD (Receive Data)
3	TD (Transmitted Data)
6	DSR (Data Set Ready)
ขา	สัญญาณ
7	RTS (Request To Send)
8	CTS (Clear To Send)
9	RI (Ring Indicator)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การอินเตอร์รัปต์

การเขียนที่มีการติดต่อกับฮาร์แวร์นี้จะมีวิธีการเขียนซอฟต์แวร์หลายวิธี ในทางปฏิบัติแต่จะมีวิธีหนึ่งที่ใช้สัญญาณทางฮาร์แวร์เข้ามากระตุ้นเพื่อกระโดดไปทำโปรแกรมที่ต้องการ ซึ่งก็คือการอินเตอร์รัปต์นั่นเอง

2.5.1 แหล่งกำเนิดของอินเตอร์รัปต์ของ 8051

8051 ยอมให้เกิดอินเตอร์รัปต์ได้ทั้งหมด 5 แหล่งด้วยกัน 3 แหล่งมาจากภายในตัว 8051 เอง ได้แก่ การอินเตอร์รัปต์ที่เกิดจากไทเมอร์แฟล็ก 0 ไทเมอร์แฟล็ก 1 และจากพอร์ตอนุกรม (RI และ TI) การอินเตอร์รัปต์ทั้งหมดจะถูกควบคุมโดยโปรแกรมผ่านทางอินเตอร์รัปต์อีนาเบิ้ลรีจิสเตอร์ (IE) อินเตอร์รัปต์ไพออริตี้รีจิสเตอร์ (IP) และรีจิสเตอร์ควบคุมไทเมอร์ (TCON)

รายละเอียดของรีจิสเตอร์ IE

7	6	5	4	3	2	1	0
EA	-	ET2	ES	ET1	EX1	ET0	EX0

รูปที่ 2.9 แสดงรายละเอียดของรีจีสเตอร์ IE

สัญลักษณ์	รายละเอียด
EA	บิตแสดงการยอมให้มีการอินเตอร์รัปต์ เมื่อเซ็ตให้เป็น 1 แสดงว่าต้องการให้มีการอินเตอร์รัปต์เกิดขึ้น และเคลียร์ให้เป็น 0 เมื่อไม่ต้องการให้มีการอินเตอร์รัปต์
-	ไม่ใช้งาน
ET2	สงวนไว้ใช้งานภายหน้า
ES	ยอมให้มีการอินเตอร์รัปต์เนื่องมาจากพอร์ตอนุกรม โดยการเซ็ต ES ให้เป็น 1 และเคลียร์เป็น 0 เมื่อไม่ต้องการให้มีการอินเตอร์รัปต์จากพอร์ตอนุกรม
ET1	ยอมให้มีการอินเตอร์รัปต์เนื่องมาจากไทเมอร์ 1 เกิดโอเวอร์โฟลว์เซ็ตให้เป็น 1เมื่อต้องการให้มีการอินเตอร์รัปต์จากไทเมอร์ 1 และเคลียร์เป็น 0 เมื่อไม่ต้องการให้เกิดการอินเตอร์รัปต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EX1 ยอมให้มีการอินเตอร์รัปต์มาจากภายนอกผ่านขา INT1 โดยเซ็ตเป็น 1 เมื่อต้องการให้มีการอินเตอร์รัปต์และเคลียร์เป็น 0 เมื่อไม่ต้องการให้เกิดอินเตอร์รัปต์

ET0 ยอมให้มีการอินเตอร์รัปต์เนื่องมาจากไทเมอร์ 0 เกิดโอเวอร์โฟลว์เซ็ตให้เป็น 1เมื่อต้องการให้มีการอินเตอร์รัปต์จากไทเมอร์ 1 และเคลียร์เป็น 0 เมื่อไม่ต้องการให้เกิดการอินเตอร์รัปต์

EX0 ยอมให้มีการอินเตอร์รัปต์มาจากภายนอกผ่านขา INT0 โดยเซ็ตเป็น 1 เมื่อต้องการให้มีการอินเตอร์รัปต์และเคลียร์เป็น 0 เมื่อไม่ต้องการให้เกิดอินเตอร์รัปต์

รายละเอียดของรีจิสเตอร์ IP

7	6	5	4	3	2	1	0
-	-	PT2	PS	PT1	PX1	PT0	PX0

รูปที่ 2.10 แสดงรายละเอียดของรีจีสเตอร์ IP

สัญลักษณ์	รายละเอียด
-	ไม่ใช้งาน
-	ไม่ใช้งาน
PT2	สงวนไว้ใช้ภายหน้า
PS	ลำดับความสำคัญของการอินเตอร์รัปต์จากพอร์ตอนุกรม
PT1	ลำดับความสำคัญของการอินเตอร์รัปต์จากไทเมอร์ 1
PX1	ลำดับความสำคัญของการอินเตอร์รัปต์จากภายนอก INT1
PT0	ลำดับความสำคัญของการอินเตอร์รัปต์จากไทเมอร์ 0
PX0	ลำดับความสำคัญของการอินเตอร์รัปต์จากภายนอก INT 0

2.5.2 ตำแหน่งของการอินเตอร์รัปต์

ตำแหน่งแอดเดรสที่โปรแกรมเรียก(Call)เมื่อเกิดการอินเตอร์รัปต์จะถูกกำหนดอย่างแน่นอนไว้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การอินเตอร์รัปต์ที่เกิดจาก	ตำแหน่งแอดเดรส
IE 0	0003H
TF 0	000BH
IE1	0013H
TF1	001BH
Serial	0023H

2.5.3 ลำดับความสำคัญของอินเตอร์รัปต์

ในกรณีที่เกิดการอินเตอร์รัปต์ขึ้นพร้อมกัน และมีการกำหนดความสำคัญไว้เป็น 1 เหมือนกัน(หมายถึงผู้ใช้กำหนดลำดับความสำคัญไว้เท่ากัน) 8051 จะเข้าจัดลำดับความสำคัญใหม่เรียงกันไว้ดังนี้

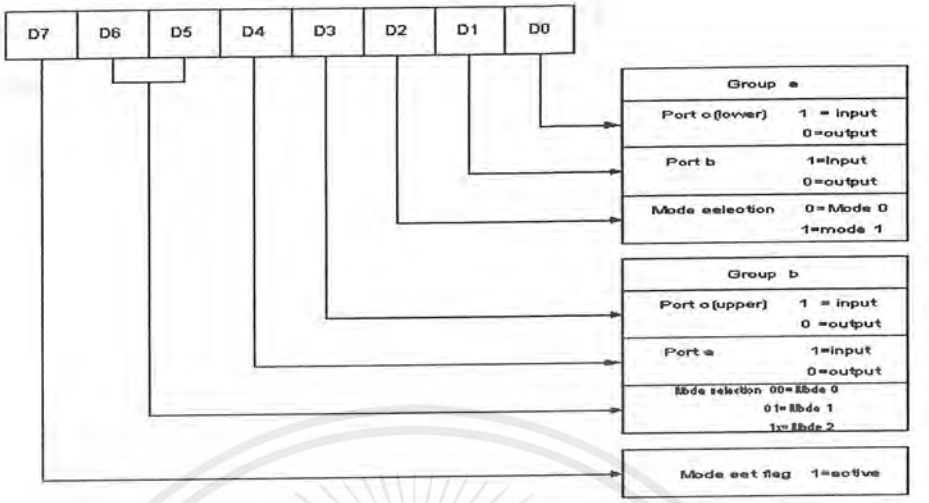
1. IE0 (สูงสุด)
2. TF0
3. IE1
4. TF1
5. Serial (ต่ำสุด)

2.6 ชิพสนับสนุน 8255 (Programable Peripheral Interface , PPI)

8255 เป็นอุปกรณ์ที่ได้รับการออกแบบมาเพื่อทำหน้าที่เป็นพอร์ต สำหรับการรับส่งข้อมูลแบบขนานระหว่างอุปกรณ์ภายนอกกับไมโครคอลโทรลเลอร์เราสามารถที่จะเปลี่ยนแปลงลักษณะการทำงานของพอร์ตให้เป็นเอาท์พุตหรืออินพุตได้เพียงการส่งข้อมูลควบคุมจาก ไมโครคอลโทรลเลอร์ก่อนที่จะเริ่มการใช้งาน

8255 ประกอบด้วยบล็อกการทำงานหลายส่วน ส่วนที่เชื่อมกับอุปกรณ์ภายนอกโดยตรงซึ่งถูกจำแนกออกเป็น 2 กลุ่ม คือ Group A Control และ Group B Control โดยจะสามารถแบ่งออกเป็น 3 พอร์ตได้แก่ พอร์ต A (PA0-PA7) พอร์ต B (PB0-PB7) และพอร์ต C (PC0-PC7) ส่วนที่กำหนดหน้าที่การทำงานของพอร์ตทั้ง 3 มีชื่อว่า รีจิสเตอร์ควบคุมซึ่งมีรายละเอียดภายในดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
36815



รูปที่ 2.11 แสดงบิตต่างๆ ของคำสั่งควบคุม

โหมดการทำงานของ 8255 มีอยู่ทั้งหมด 3 โหมด ได้แก่

- โหมด 0 : Basic I/O
- โหมด 1 : Strobe Input , Strobe Output
- โหมด 2 : Strobe Bidirection I/O

การทำงานของ 8255 โหมด 0

ในการกำหนดลักษณะให้ 8255 ทำงานในโหมด 0 นั้นจะต้องส่งคำสั่งควบคุมให้แก่ รีจิสเตอร์ควบคุมก่อน คำสั่งควบคุมนี้จะกำหนดลักษณะการทำงานแต่ละพอร์ตของ 8255

การทำงานของ 8255 โหมด 1

การทำงานของ 8255 โหมด 1 เป็นการทำงานในลักษณะที่มีการตรวจสอบสภาวะการรับส่งข้อมูล (Handshaking) สาเหตุที่ต้องเป็นเช่นนี้เพราะเมื่อนำ 8255 ไปต่อกับอุปกรณ์ภายนอกที่มีการทำงาน ช้ากว่าการทำงานของไมโครคอลโทรลเลอร์ เมื่อมีการส่งข้อมูลหลาย ๆ ไบต์อุปกรณ์ภายนอกจะรับข้อมูลไม่ทัน การผิดพลาดจึงเกิดขึ้นได้ ดังนั้นเมื่อใช้งาน 8255 ในโหมดนี้เป็นการกำหนดให้พอร์ต A และพอร์ต B เป็นพอร์ตสำหรับใช้รับส่งข้อมูลส่วนพอร์ต C ใช้เป็นสัญญาณตอบโต้กับอุปกรณ์ภายนอกโดย 4 บิตบนจะเป็นสัญญาณตอบโต้สำหรับพอร์ต A และ 4 บิตล่างใช้เป็นสัญญาณตอบโต้สำหรับพอร์ต B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ 8255 โหมด 2

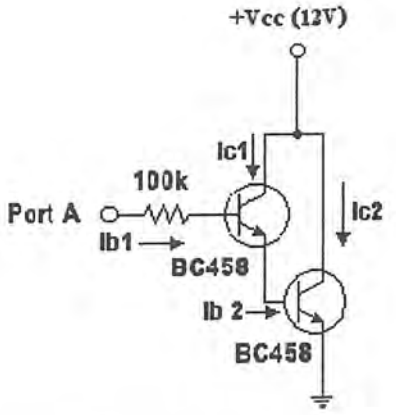
การทำงานในโหมดนี้ก็คือ การใช้พอร์ต A เป็นอินพุตและเอาท์พุตแลตซ์ (output latch) หมายถึงการเก็บเอาข้อมูลไว้เพื่อรออุปกรณ์ภายนอกรับเอาข้อมูลออกไป ส่วนอินพุตแลตซ์ (Input Latch) หมายถึงการเก็บข้อมูลที่อุปกรณ์ภายนอกส่งเข้ามา เพื่อให้ไมโครคอลโทรลเลอร์อ่านเข้าไปใช้งาน ในการส่งข้อมูลให้อุปกรณ์ภายนอกนั้น ขั้นตอนแรกที่เกิดขึ้นคือ ไมโครคอลโทรลเลอร์จะต้องส่งข้อมูลไปแลตซ์ไว้ในพอร์ต A ซึ่งจะทำให้ OUTPUT BUFFER FULL (OBF) ถูกเซ็ตให้เป็นจริง สัญญาณนี้จะบอกให้อุปกรณ์ภายนอกรู้ว่า ขณะนี้พอร์ต A มีข้อมูลอยู่และบอกให้ไมโครคอลโทรลเลอร์รู้ว่าข้อมูลของพอร์ต A ที่ส่งไปใช้ยังไม่ถูกอุปกรณ์ภายนอกอ่านออกไป อุปกรณ์ภายนอกจะต้องส่งสัญญาณ ACK (ACKNOWLEDGE) ให้กับ8255 สัญญาณนี้จะเป็นการอีนาเบิ้ลให้ข้อมูลที่อยู่ในพอร์ต A ส่งออกไปยัง PA0-PA7 และเป็นการรีเซ็ต OBF เพื่อเป็นการบ่งบอกให้ไมโครคอลโทรลเลอร์รู้ว่าข้อมูลที่อยู่ภายในพอร์ต A ถูกอุปกรณ์ภายนอกอ่านออกไปแล้ว และสามารถส่งข้อมูลใหม่ไปให้พอร์ต A ได้อีก

ในการรับข้อมูลจากอุปกรณ์ภายนอกนั้น อุปกรณ์ภายนอกต้องตรวจสอบสถานะของ INPUT BUFFER FULL (IBF) เสียก่อนถ้า IBF มีลอจิกเป็น 1 แสดงว่าขณะนี้พอร์ต A มีข้อมูลอยู่ โมโครคอลโทรลเลอร์ยังไม่ได้อ่านข้อมูลเข้าไป แต่ถ้ามีลอจิกเป็น 0 แสดงว่าไม่มีข้อมูลอยู่ในพอร์ต A อุปกรณ์ภายจอกจะส่งข้อมูลและสัญญาณ STROBE (STB) ให้ 8255 สัญญาณนี้จะทำให้มีการนำข้อมูลแลตซ์ไว้ในพอร์ต A และเซ็ตให้ IBF มีสถานะเป็นจริง ไมโครคอลโทรลเลอร์จะตรวจสอบสถานะของ IBF ได้โดยการอ่านข้อมูลจากพอร์ต C เมื่อ IBF มีสถานะเป็นจริงแสดงว่าข้อมูลที่พร้อมที่จะให้ไมโครคอลโทรลเลอร์อ่านออกไปเมื่อไมโครคอลโทรลเลอร์อ่านข้อมูลออกไปแล้ว IBF จะมีสถานะลอจิกเป็น 0 และอุปกรณ์ภายนอกสามารถส่งข้อมูลใหม่เข้ามาได้

2.7 วงจรดาร์ลิงตัน (Darlington)

วงจรดาร์ลิงตันดังในรูปที่ 2.12 หากพิจารณาให้ดีก็คือวงจรขยายอิมิตเตอร์นั่นเอง โดยจะใช้ทรานซิสเตอร์สองตัวต่อกัน โดยทรานซิสเตอร์ตัวหลังจะเป็นโหลดของทรานซิสเตอร์ตัวแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 วงจรดาร์ลิงตัน

จากรูปที่ 2.12 ถ้ามีสัญญาณป้อนเข้าวงจรระหว่างเบสของทรานซิสเตอร์ Q1 กับกราวนด์จะทำให้กระแสไหลในทิศทางดังรูป กระแส I_{C1} จะมีค่าเป็น β เท่าของกระแส I_{B1} ($I_{C1} = \beta.I_{B1}$) โดย β เป็นอัตราขยายกระแสของทรานซิสเตอร์ Q2 กระแส I_{B1} และ I_{C1} จะไหลออกทางอิมิตเตอร์ไปเข้าเบสของทรานซิสเตอร์ Q2 เป็นกระแส ดังนั้นอาจกล่าวได้ว่า

$$I_{B2} = I_{B1} + \beta_1 . I_{B1}$$
$$= (1 + \beta_1). I_{B1}$$

กระแส I_{B2} จะไหลเข้าทางเบสและเป็นกระแสที่ทำให้เกิด I_{C2} โดยอัตราขยายจะขึ้นอยู่กับ β_2 ดังนั้นกระแสรวมคือกระแสที่ไหลผ่านรีเลย์ (I_L)

$$I_L = I_{B2} + I_{C2}$$
$$= I_{B2} + \beta_2 . I_{B2}$$
$$= (1 + \beta_2). I_{B2}$$

แต่ $I_{B2} = (1 + \beta_1). I_{B1}$

ดังนั้น $I_L = (1 + \beta_2). (1 + \beta_1). I_{B1}$

$$\therefore \quad I_L \approx \beta_1.\beta_2.I_{B1}$$

จากสมการแสดงให้เห็นว่า วงจรดาร์ลิงตันจะช่วยในการขยายกระแสได้มากทีเดียว

2.8 เดลไฟล์ (Delphi)

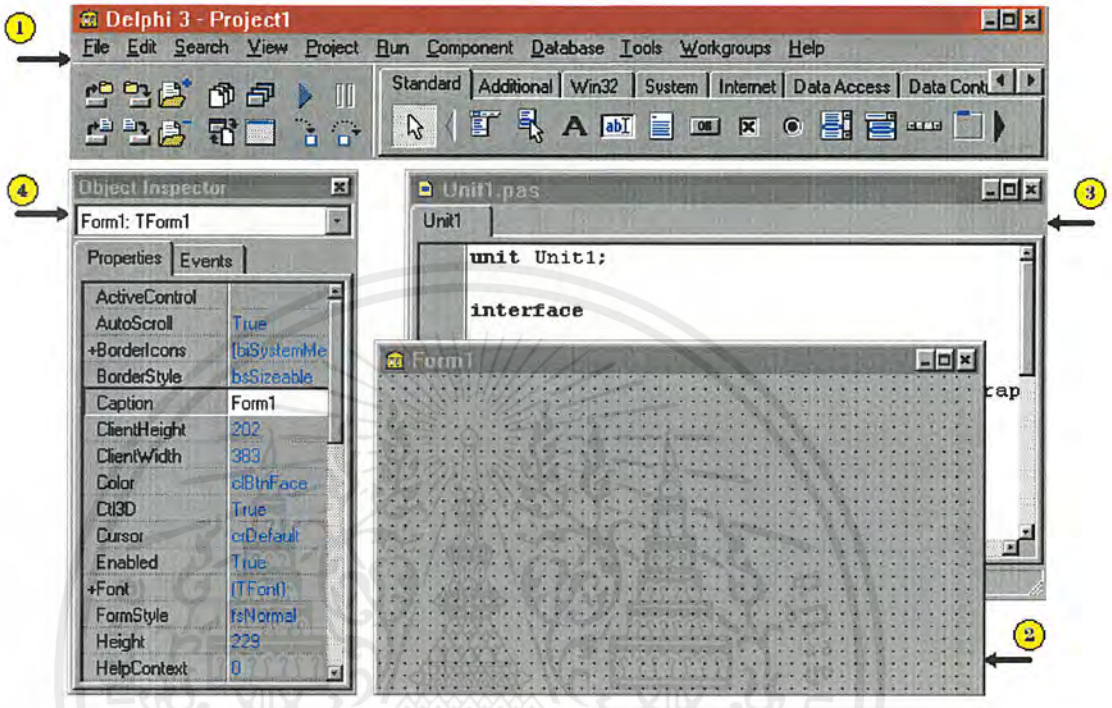
2.8.1 หน้าจอของเดลไฟล์

หน้าจอของเดลไฟล์แบ่งออกได้เป็น 4 ส่วนใหญ่ๆ คือ

1. หน้าต่างหลัก
2. หน้าต่างฟอร์ม (Form)
3. หน้าต่างยูนิต (Unit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.หน้าต่างออบเจ็กต์อินสเปคเตอร์ (Object Inspector)



รูปที่ 2.13 หน้าจอของเดลไฟล์โดยรวม

2.8.2 หน้าต่างหลัก

หน้าต่างหลักประกอบด้วย

1. ไตเติลบาร์ (Title bar) จะแสดงรายชื่อของชิ้นงาน (Application) แต่ละชิ้นงาน
2. เมนูบาร์ (Menu bar) จะแสดงรายการตั้งแต่ File จนถึง Help
3. กล่องคอมโปเนนต์ (Component) จะเป็นที่เก็บคอมโปเนนต์ต่างๆ ที่จะไปวางไว้บนฟอร์ม
4. สปีดบาร์ (Speed bar) ประกอบไปด้วย

   4.1 OPEN PROJECT คือ ปุ่มที่ใช้เปิดโปรเจกต์ใหม่ที่สร้างไว้แล้ว

   4.2 OPEN FILE คือ ปุ่มที่ใช้เปิดไฟล์ใหม่ที่สร้างไว้แล้ว

   4.3 SAVE ALL คือ ปุ่มที่ใช้เซฟโปรเจกต์ของงานที่กำลังทำอยู่

   4.4 SAVE FILE คือ ปุ่มที่ใช้เซฟเฉพาะบางไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ADD FILE TO PROJECT คือ ปุ่มที่ใช้ในการทำยูนิต (ไฟล์นามสกุล PAS) ของโปรเจกต์อื่นเข้ามาทำงานร่วมกับโปรเจกต์ที่เรากำลังทำงานอยู่

4.6 REMOVE FILE TO PROJECT คือ ปุ่มที่ใช้ในการทำยูนิตของโปรเจกต์ที่กำลังใช้งานอยู่ออกไป

4.7 SELECT UNIT FORM LIST คือ ปุ่มที่ใช้เลือกยูนิตที่มีอยู่แล้วในโปรเจกต์ขึ้นมาใช้งาน

4.8 TOGGLE FORM / UNIT คือ ปุ่มที่ใช้ในการสลับการทำงานระหว่างหน้าต่างฟอร์มกับหน้าต่างยูนิต

4.9 SELECT FORM FROM LIST คือ ปุ่มที่ใช้เลือกฟอร์มที่มีอยู่แล้วในโปรเจกต์ขึ้นมาใช้

4.10 NEW FORM คือ ปุ่มที่ใช้เรียกฟอร์มใหม่ขึ้นมาใช้ร่วมงานกับโปรเจกต์ที่กำลังใช้งานอยู่

4.11 RUN คือ ปุ่มที่ทำให้ตัว Project compiles เริ่มทำงาน

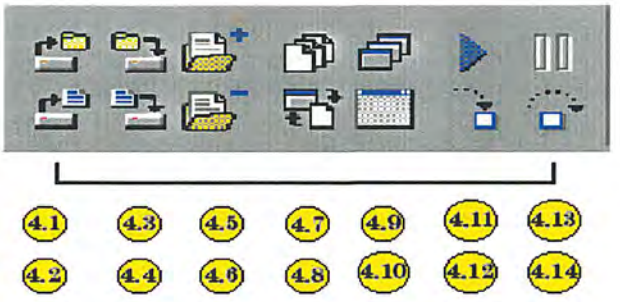
4.12 TRACE INTO คือ ปุ่มที่ใช้แสดงผลการทำงานของโปรเจกต์ทีละขั้นตอน

4.13 PAUSE คือ ปุ่มที่ทำให้การทำงานของโปรเจกต์หยุดชั่วคราว

4.14 STEP OVER คือ ปุ่มที่ใช้ในการข้ามการแสดงผลบางช่วงไป

รูปที่ 2.14 หน้าต่างหลัก

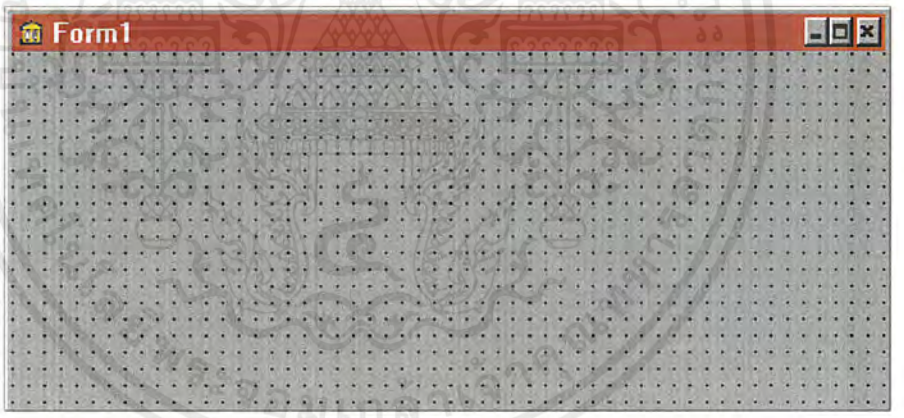
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 สปีดบาร์หน้าต่างหลัก

2.8.3 หน้าต่างฟอร์ม

หน้าต่างฟอร์มเป็นหน้าต่างที่มีไว้สำหรับแสดงผลในการทำงานของตัวโปรเจกต์ เวลาใช้ให้นำคอมโปเนนต์ที่ต้องการใช้ลงไปวาง แล้วกำหนด Properties กับ Events ในออบเจกต์อินสเปคเตอร์

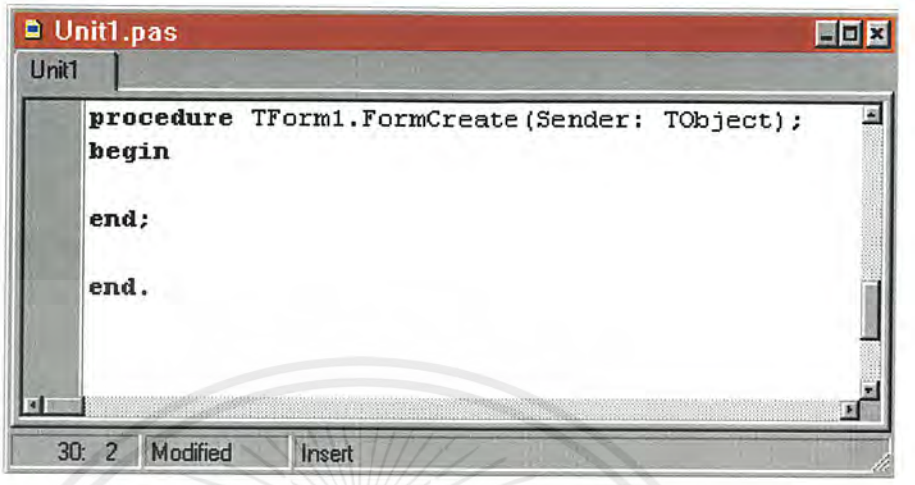


รูปที่ 2.16 หน้าต่างฟอร์ม (Form)

2.8.4 หน้าต่างยูนิต

หน้าต่างยูนิตเป็นหน้าต่างที่มีไว้สำหรับใส่เงื่อนไขในการทำงานของโปรแกรมโดยผ่านทาง Events ของในออบเจกต์อินสเปคเตอร์ หรือจะใส่เงื่อนไขโดยตรงก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



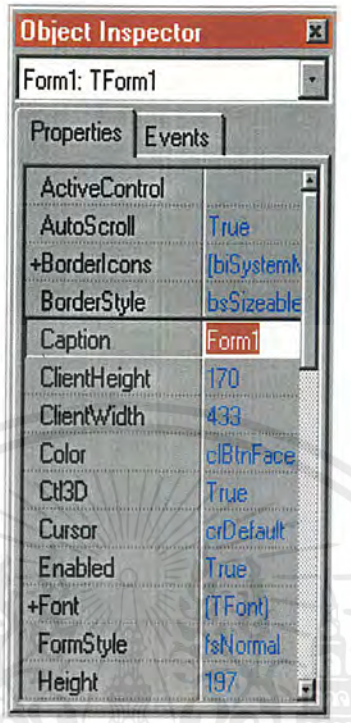
รูปที่ 2.17 หน้าต่างยูนิต (Unit)

2.8.5 หน้าต่างออบเจกต์อินสเปคเตอร์

หน้าต่างออบเจกต์อินสเปคเตอร์จะประกอบด้วย

1. Properties จะเป็นตัวกำหนดการทำงานของคอมโปรเนนต์ต่างๆ
2. Eventsจะเป็นตัวผ่านจากคอมโปเนนต์ไปกำหนดเงื่อนไขต่างๆในยูนิต เช่น
   - OnClick จะทำงานก็ต่อเมื่อเราเลือกด้วยเมาส์หรือดด้วยคีย์บอร์ด
   - OnMouseDown จะทำงานก็ต่อเมื่อเราคลิกเมาส์ลง
   - OnMouseUp จะทำงานก็ต่อเมื่อเราคลิกเมาส์ขึ้น
   - OnKeyDown จะทำงานโดยการกดคีย์ลง และจะไปทำงานที่Key Code ที่กำหนดไว้ในเงื่อนไขของ Events นั้นๆ ในโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 หน้าต่างออบเจกต์อินสเปคเตอร์ (Object Inspector)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการออกแบบ

ลักษณะการออกแบบโครงงาน จะเป็นการติดต่อกันระหว่าง คอมพิวเตอร์กับไมโครคอลโทรลเลอร์(MCS-51) โดยการส่งข้อมูลสื่อสารกันระหว่างอุปกรณ์ทั้งสองตัว จะทำงานสลับกันเป็นอุปกรณ์ต้นทางกับอุปกรณ์ปลายทาง โดยมีการรับข้อมูลอินพุท(Input)จากผู้ใช้ได้ 2 ทาง คือ การควบคุมจากคอมพิวเตอร์โดยจะส่งข้อมูลที่มีการเปลี่ยนแปลงในลักษณะแอดเดรสต่างๆกันให้กับไมโครคอลโทรลเลอร์(MCS-51) ซึ่งไมโครคอลโทรลเลอร์(MCS-51)จะเป็นตัวประมวลผลโดยการเปรียบเทียบแอดเดรสที่เข้ามาแล้วส่งออกไปแสดงผลให้กับส่วนเอาท์พุท(Output) และอีกส่วนหนึ่งสามารถควบคุมได้จากสวิตช์ไฟ ซึ่งการควบคุมจากทั้งสองส่วนจะส่งผลต่อการเปลี่ยนแปลงของส่วนเอาท์พุต(Output)ด้วย ดังนั้นจึงต้องมีชุดตรวจสอบสถานะของอุปกรณ์ เป็นตัวตรวจการเปลี่ยนแปลงของอุปกรณ์ไฟฟ้า แล้วส่งสัญญาณกลับไปให้ไมโครคอลโทรลเลอร์(MCS-51) เป็นตำแหน่งแอดเดรสของอุปกรณ์ไฟฟ้าที่เปลี่ยนแปลง จากนั้นจึงนำแอดเดรสที่ได้นี้กลับไปเปรียบเทียบแอดเดรสของข้อมูลในคอมพิวเตอร์เพื่อให้คอมพิวเตอร์แสดงผลสถานะจริงของอุปกรณ์ไฟฟ้าในแต่ละจุด

รูปที่ 3 .1 ลำดับขั้นตอนการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1ลักษณะโครงสร้างของระบบ

โครงสร้างของระบบที่ใช้ควบคุมโดยรวมจะประกอบไปด้วยการควบคุมโดยส่วนของฮาร์ดแวร์และการควบคุมโดยส่วนของซอฟต์แวร์ ซึ่งทั้งสองส่วนจะมีความสัมพันธ์กันโดยตรง หากขาดส่วนใดส่วนหนึ่ง ก็จะไม่สามารถควบคุมให้เป็นไปตามวัตถุประสงค์ที่ต้องการได้การออกแบบจะเป็นการออกฮาร์ดแวร์และออกแบบซอฟต์แวร์

3.2 การออกแบบซอฟต์แวร์

3.2.1 การออกแบบของโปรแกรมบนคอมพิวเตอร์

การออกแบบในส่วนของโปรแกรมควบคุมจะออกแบบให้มีการควบคุมได้ 2 วิธี คือ การควบคุมโดยตรงและการตั้งเวลาเปิด-ปิดโดยอัตโนมัติ

การควบคุมโดยตรง การควบคุมแบบนี้ผู้ใช้สามารถที่จะควบคุมอุปกรณ์ไฟฟ้านั้นได้ โดยการใช้เมาส์ไปคลิกปุ่มที่ควบคุมอุปกรณไฟฟ้านั้นตามต้องการ เมื่อผู้ใช้คลิกที่ปุ่มโปรแกรมจะส่งข้อมูลที่เป็นรหัสของหลอดนั้นออกไปยังพอร์ต โดยอาศัยตัวแปร sendup (กำหนดเป็นตัวแปรในโปรแกรมควบคุม) เป็นส่งรหัสของหลอดแต่ละหลอดให้ไมโครคอลโทรลเลอร์ จากนั้นจึงทำการเรียกฟังก์ชั่นที่ใช้ส่งข้อมูล ซึ่งฟังก์ชั่นที่ใช้จะอยู่ในหน้าต่างที่ 2 โดยส่วนที่ 1 จะเป็นส่วนควบคุมหลอดไฟจะมีรหัสตัว a ถึงหลอดสุดท้ายคือ หลอดที่ 8ซึ่งจะมีรหัสตัว h และส่วนที่ 2 เป็นส่วนควบคุมเต้ารับโดยจะมีรหัสตัว i จนถึงเต้ารับตัวสุดท้ายคือ เต้ารับที่ 8 ซึ่งจะมีรหัสตัว p โดยรหัสของหลอดและเต้ารับทั้งหมดแสดงในตารางที่ 3.1 และคำสั่งที่ใช้ส่งข้อมูลดังนี้

```
sendup:= 'a';    กำหนดรหัสของหลอดที่ต้องการควบคุมของหลอดไฟดวงที่ 1
sendup:= 'i';    กำหนดรหัสของเต้ารับที่ต้องการควบคุมของเต้ารับจุดที่ 1
Form2.send;      เรียกฟังก์ชั่นที่ใช้ส่งข้อมูลในหน้าต่างที่ 2
```

ตารางที่ 3.1 แสดงรหัสของตัวอักษร

ตัวอักษร	รหัสของหลอด
a	หลอดที่ 1
b	หลอดที่ 2
c	หลอดที่ 3
d	หลอดที่ 4
e	หลอดที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร	รหัสของหลอด
f	หลอดที่ 6
g	หลอดที่ 7
h	หลอดที่ 8

ตัวอักษร	รหัสของเต้ารับ
i	เต้ารับที่ 1
j	เต้ารับที่ 2
k	เต้ารับที่ 3
l	เต้ารับที่ 4
m	เต้ารับที่ 5
n	เต้ารับที่ 6
o	เต้ารับที่ 7
p	เต้ารับที่ 8
q	ทำการรีเซ็ตข้อมูลหลอดไฟในไมโครคอลโทรลเลอร์
r	ทำการรีเซ็ตข้อมูลเต้ารับในไมโครคอลโทรลเลอร์

การควบคุมโดยอัตโนมัติ ได้มีการออกแบบให้สามารถเปิด-ปิด ได้เองโดยการให้ผู้ใช้ป้อนเวลาที่ต้องการเปิด-ปิด ลงในช่องควบคุมของแต่ละช่องของแต่ละหลอด เมื่อผู้ใช้ป้อนข้อมูลเสร็จแล้วจะต้องเก็บข้อมูลลงในโปรแกรม การเก็บข้อมูลดังกล่าวนั้นจะต้องอาศัย Database เข้ามาช่วย จากนั้นโปรแกรมจะนำข้อมูลดังกล่าวมาประมวลผล แต่การประมวลผลนั้นไม่สามารถที่จะกระทำบนดาต้าเบส (Database) ได้โดยตรง จึงต้องมีการดึงข้อมูลดังกล่าวมาพักไว้ใน StringGrid โดยใช้ฟังก์ชั่นดังนี้

```
procedure Tform5.copydata;
var row:integer;
begin
For row := 1 to 8 do
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
table1.recno:=row;
stringgrid1.Cells[0,row-1]:= inttostr(table1.FieldByName('lamp number').asinteger);
stringgrid1.Cells[1,row-1]:= timetostr(table1.FieldByName('timer on').asdatetime);
stringgrid1.Cells[2,row-1]:= timetostr(table1.FieldByName('timer off').asdatetime);
stringgrid1.Cells[3,row-1]:= datetostr(table1.FieldByName('chang in').asdatetime);
stringgrid1.Cells[4,row-1]=datetostr(table1.FieldByName('chang out').asdatetime);
stringgrid1.Cells[5,row-1]:=inttostr(table1.FieldByName('age of lamp').asinteger);
end;
End;
```

การควบคุมการ เปิด-ปิด ของหลอดแต่ละหลอดและของเต้ารับแต่ละจุด จะทำงานโดยการเปรียบเทียบเวลาของผู้ใช้ที่ป้อนไว้ในครั้งแรกและนำมาไว้ในสตริงกริด (StringGrid) กับเวลาของเครื่องโดยใช้คำสั่งcomparestr (s1,s2) เข้ามาช่วย เมื่อถึงเวลาที่กำหนดเวลาที่มีการสั่งให้เปิดจะทำการตรวจสอบว่าหลอดนั้นเปิดอยู่หรือไม่ หากปิดอยู่จะทำการส่งรหัสของหลอดนั้นออกไปยังพอร์ตอนุกรม ในส่วนของเต้ารับเมื่อถึงกำหนดเปิดก็จะทำการส่งรหัสของเต้ารับนั้นออกไปยังพอร์ตอนุกรม และในการสั่งให้มีการปิดก็เช่นกันจะทำการตรวจสอบว่าหลอดนั้นเปิดหรือไม่ หากปิดจะไม่ส่งข้อมูลออกทางพอร์ตอนุกรมแต่ถ้าเปิดจะส่งข้อมูลออกทางพอร์ตอนุกรมและในส่วนของเต้ารับเมื่อถึงกำหนดปิดก็จะทำการส่งรหัสของเต้ารับนั้นออกไปยังพอร์ตอนุกรม

การแสดงอายุการใช้งานของหลอดแต่ละหลอด เนื่องจากการใช้งานของหลอดแต่ละหลอดจะมีกำหนดระยะการใช้งานตามมาตราฐานของหลอดแต่ละชนิด การกำหนดอายุการใช้งานของหลอดผู้ใช้จะต้องป้อนอายุการใช้งานซึ่งมีหน่วยเป็นชั่วโมง ในช่องอายุการใช้งาน แต่เนื่องจากโปรแกรมจะไม่ทราบว่า ภายในหนึ่งวันการทำงานของหลอดเป็นเวลานานกี่ชั่วโมง ดังนั้นผู้ใช้จะต้องป้อนเวลาที่ต้องการเปลี่ยนหลอดโดยใช้การทำงานโดยเฉลี่ยต่อวันเป็นส่วนช่วยในการพิจารณา จากนั้นต้องป้อนวันที่ต้องเปลี่ยนหลอดให้กับโปรแกรม โปรแกรมจะทำเก็บข้อมูลในส่วนนี้ลงในดาต้าเบส(Database) และเมื่อผู้ใช้เรียกหน้าต่างการตั้งเวลาครั้งต่อไป โปรแกรมจะทำการตรวจสอบว่าหลอดใดถึงกำหนดเปลี่ยน หากพบว่าหลอดใดถึงกำหนดเปลี่ยนจะทำการแจ้งให้ผู้ใช้ทราบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การออกแบบของโปรแกรมของไมโครคอลโทรลเลอร์

โปรแกรมไมโครคอลโทรลเลอร์ที่ได้ออกแบบไว้นั้นสามารถแบ่งออกได้เป็น 4 ส่วนใหญ่ คือ โปรแกรมกำหนดหน้าที่การทำงานของฮาร์ดแวร์ โปรแกรมที่ใช้รับคำสั่งจากคอมพิวเตอร์ โปรแกรมที่ใช้รับข้อมูลจากชุดตรวจสถานะ และโปรแกรมที่ส่งข้อมูลกลับไปยังคอมพิวเตอร์

- **โปรแกรมกำหนดหน้าที่การทำงานของฮาร์ดแวร์**

การทำงานของอุปกรณ์ต่างๆจะต้องกำหนดหน้าที่ให้กับอุปกรณ์นั้น ซึ่งในโครงงานนี้มีเป็นการสื่อสารผ่านพอร์ตอนุกรมจึงต้องมีการกำหนดหน้าที่ของรีจิสเตอร์ที่เกี่ยวข้องดังนี้

รีจิสเตอร์ SCON

การสื่อสารผ่านพอร์ตอนุกรมในโครงงานจะใช้การสื่อสารในโหมด 1 เนื่องจากในโหมดนี้การสื่อสารจะเป็นแบบมาตราฐานสามารถติดต่อกับคอมพิวเตอร์ได้และในโหมด 2 และโหมด 3 จะใช้ในการติดต่อระหว่างไมโครคอลโทรลเลอร์กับไมโครคอลโทรลเลอร์ ดังนั้นจึงมีการกำหนดบิตต่างๆของดังนี้

SM1 = 1 เพื่อกำหนดโหมดการสื่อสารเป็นโหมด 1

REN = 1 เพื่อรับข้อมูลจากอนุกรม

บิต	7	6	5	4	3	2	1	0
รีจิสเตอร์	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
50H	0	1	0	1	0	0	0	0

รีจิสเตอร์ TMOD

เนื่องจากการสื่อสารในโหมด 1 จะใช้ไทเมอร์ 1 เป็นตัวสร้างความถี่จึงต้องกำหนดหน้าที่ให้กับไทเมอร์ด้วย โดยไทเมอร์จะทำงานในโหมด 2 สามารถโหลดค่าเริ่มต้นเข้าไปไว้ใน TH1 โดยอัตโนมัติหลังจากการเกิดโอเวอร์โฟลว์ โดยกำหนดรีจิสเตอร์ต่างๆดังนี้

M1 = 1 เพื่อเลือกโหมดการทำงานไทเมอร์เป็นโหมด 2

บิต	7	6	5	4	3	2	1	0
รีจิสเตอร์	Gate	C/T	M1	M0	Gate	C/T	M1	M0
20H	0	0	1	0	0	0	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดอัตราบอด

การเลือกใช้อัตราบอดของโครงงานนี้จะใช้อัตราบอดเป็น 9600 เฮิรตซ์ เนื่องจากอัตราบอดที่ 9600 เฮิรตซ์ นี้เป็นอัตราบอดมาตราฐานที่มีความถูกต้องมากที่สุดและไม่ทำให้การสื่อสารเกิดการล่าช้ามากเกินไปโดยการคำนวณจะใช้สมการที่ 2.1

จากสมการ 2.1

$$f_{buad} = \frac{2^{s\,\mathrm{mod}}}{32_d} \times \frac{Oscillator\ frequency}{12_d \times [256_d - (TH1)]}$$

กำหนดให้

f_{buad} = 9600 เฮิรตซ์

smod = 0

Oscillator frequency = 11.0592 เมกะเฮิรตซ์

TH1 = ค่าเริ่มต้นจากการคำนวณที่กำหนดให้ TH1 เพื่อใช้สร้างความถี่

แทนค่าในสมการ

$$TH1 = \frac{(2^0 \times 11.0592 \times 10^6)}{(32_d \times 12_d \times 9600_d)} \doteq 253_d = 0FD_h$$

รีจิสเตอร์ Control ของ 8255

การกำหนดรีจิสเตอร์ Control ของ 8255 จะกำหนดตามการทำงานของ 8255 ตัวนั้น ซึ่งขึ้นอยู่กับการออกแบบฮาร์ดแวร์ว่าให้พอร์ตใดทำหน้าที่เป็นอินพุตหรือเอาท์พุต โดยในโครงงานนี้ได้กำหนดให้ 8255 อยู่ที่ตำแหน่ง 0E0E0H – 0E0E3H ทำหน้าที่เป็นอินพุตทั้งสองพอร์ต และทำหน้าที่เป็นเอาท์พุตหนึ่งพอร์ต ซึ่งกำหนดในรีจิสเตอร์ Control ของ 8255 ทั้งสองตัวได้ดังนี้

กำหนดรีจิสเตอร์ Control

บิต	7	6	5	4	3	2	1	0
รีจิสเตอร์	D7	D6	D5	D4	D3	D2	D1	D0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9Bh	1	0	0	1	0	0	1	0

โปรแกรมในส่วนของการกำหนดหน้าที่การทำงานของอุปกรณ์ต่างๆมีดังนี้

```
mov   scon,#52h
mov   tmod,#20h
mov   th1,#0fdh
setb  tr1
mov   dptr,#pcontrol1
mov   a,#89h
movx  @dptr,a
```

- **โปรแกรมที่ใช้รับข้อมูลจากชุดตรวจสถานะ**

โปรแกรมในส่วนนี้จะทำการตรวจสอบการทำงานของหลอดอยู่ตลอดเวลา เมื่อมีการเปลี่ยนแปลงสถานะของหลอดเมื่อใด จะทำการประมวลผลและหาบิตที่เปลี่ยนแปลงและเก็บข้อมูลดังกล่าวไว้ในรีจิสเตอร์ R5 จากนั้นจะส่งผลที่ได้ให้กับโปรแกรมที่ส่งข้อมูลกลับไปยังคอมพิวเตอร์ ซึ่งโปรแกรมในส่วนนี้จะยังคงไม่ทราบว่า หลอดที่เปลี่ยนแปลงนั้นมีการเปลี่ยนแปลงจากสภาวะปิดเป็นเปิดหรือจากสภาวะเปิดเป็นปิด คำสั่งที่ใช้มีดังนี้

```
main_1: mov   dptr,#pc1
        lcall   delay
        movx  a,@dptr
        mov   r4,a
        cjne  a,03h,out11
```

จากโปรแกรมข้างต้น จะเป็นการรับข้อมูลจากชุดตรวจสถานะของส่วนควบคุมหลอด และทำการเปรียบเทียบว่ามีการเปลี่ยนแปลงหรือไม่ ถ้ามีจะไปทำในโปรแกรมย่อย out11 และจะเก็บหลอดที่เปลี่ยนแปลงไว้ในรีจิสเตอร์ R5 แล้วจึงไปทำในส่วนของโปรแกรมส่งข้อมูลกลับไปยังคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
out1:   mov   a,r4
        xrl   a,r3
        mov   r5,a
        mov   a,r4
        mov   r3,a
        mov   a,r5
```

- **โปรแกรมส่งข้อมูลกลับไปยังคอมพิวเตอร์**

โปรแกรมในส่วนนี้จะทำต่อจากโปรแกรมข้อมูลจากชุดตรวจสถานะ โดยนำข้อมูลที่อยู่ในรีจิสเตอร์ R5 มาประมวลผล เพื่อหาว่าหลอดใดมีการเปลี่ยนแปลงและมีการเปลี่ยนแปลงแบบใด โดยการตรวจเช็คว่ามีการเปลี่ยนแปลงแบบใดนั้นจะใช้ข้อมูลในรีจิสเตอร์ R4 เป็นตัวแสดงสภาวะในขณะนั้น

```
b0ch1:  anl   a,#01h
        cjne  a,#01h,b1ch1
        mov   a,r4
        anl   a,#01h
        cjne  a,#1h,b100
        mov   a,#41h
        lcall send
        sjmp  b1ch1
b100:   mov   a,#61h
        lcall send
        sjmp  b1ch1
```

```
send:   mov   sbuf,a
        jnb   ti,$
        clr   ti
        mov   a,#00
        mov   sbuf,a
        jnb   ti,$
        clr   ti
        lcall delay
        ret
```

จากโปรแกรมจะเป็นการตรวจสอบว่าหลอดที่ 1 มีการเปลี่ยนแปลงหรือไม่ ถ้าไม่มีจะไปทำการตรวจหาหลอดที่มีการเปลี่ยนแปลงต่อไปจนกระทั่งครบทั้ง 8 หลอด แต่ถ้ามีจะทำการตรวจสอบรีจิสเตอร์ R4 ว่าในขณะนี้หลอดที่ 1 เปิดหรือปิด ถ้าปิดอยู่จะส่ง 61H(a) ออกทางพอร์ตอนุกรมแต่ถ้าเปิดอยู่จะส่ง 41H(A) ออกทางพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **โปรแกรมที่ใช้รับคำสั่งจากคอมพิวเตอร์**

การสั่งงานจากคอมพิวเตอร์จะส่งข้อมูลทางพอร์ตอนุกรม โดยใช้บิตข้อมูลทั้งหมด 8 บิต เมื่อไมโครคอลโทรลเลอร์รับข้อมูลมาทั้ง 8 บิตแล้วจะมีรีจิสเตอร์ที่ชื่อว่า RI เซ็ต ดังนั้นการเขียนโปรแกรมในส่วนนี้จะทำตรวจเช็คจาก RI แล้วจึงไปอ่านข้อมูลจากรีจิสเตอร์ SBUF จากนั้นนำข้อมูลประมวลผลเพื่อหาว่าเป็นของหลอดใดแล้วจึงสั่งชุดเอาท์พุตตามที่ผู้ใช้ต้องการ

```
main_2:   jb     ri,inter
          sjmp   main_1
inter:    clr    ri
          mov    a,sbuf
          mov    r1,a
          mov    a,r1
l1:       cjne   a,#61h,l2
          mov    a,#01h
          xrl    a,r6
          mov    r6,a
          mov    dptr,#pa1
          movx   @dptr,a
          sjmp   main_1
```

จากโปรแกรมข้างต้นหากผู้ใช้ต้องการควบคุมหลอดที่ 1 ของส่วนควบคุมหลอด ข้อมูลจะส่งออกมาเป็น 61H โปรแกรมจะทำเปรียบเทียบว่าเป็น 61h หรือไม่ ถ้าไม่ใช่จะไปทำงานในโปรแกรมย่อย l2 แต่ถ้าเป็นของหลอดที่ 1 จะทำการเปลี่ยนแปลงหน้าสัมผัสเพื่อเปลี่ยนสถานะของหลอดที่ 1

3.3 การออกแบบงานในส่วนฮาร์ดแวร์

ฮาร์ดแวร์เป็นส่วนหนึ่งที่ประกอบไปด้วย อุปกรณ์ต่างๆที่มีหน้าที่แตกต่างกันออกไป นำมารวมกันให้มีขีดความสามารถเป็นไปตามวัตถุประสงค์ของงานซึ่งลักษณะของโครงงานจะเป็นการควบคุมจากคอมพิวเตอร์ที่เป็นจุดศูนย์กลางในการสั่งงานเพื่อไปควบคุมอุปกรณ์ไฟฟ้ายังส่วนต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ต้องการ และสามารถสั่งงานได้จากอุปกรณ์ที่ใช้ควบคุมการเปิด-ปิดอุปกรณ์ที่มีอยู่แล้วได้โดยตรงได้ นั่นคือการควบคุมในลักษณะสองทิศทางและสามารถที่จะตรวจสอบการทำงานของโหลด ที่ถูกควบคุมจากทั้งสองส่วนกลับมาแสดงผลยังคอมพิวเตอร์ได้

3.3.1 โครงสร้างของระบบฮาร์ดแวร์

จากลักษณะการทำงานดังที่กล่าวข้างต้น สามารถที่จะจำแนกโครงสร้างฮาร์ดแวร์ในระบบเพื่อให้มีการทำงานในส่วนต่างๆที่สัมพันธ์กัน ดังนี้

1. คอมพิวเตอร์ส่วนบุคคล(PC)
2. พอร์ตสื่อสารข้อมูลอนุกรม
3. ส่วนประมวลผล(MCS-51)
4. ส่วนขยายพอร์ตอินพุท/เอาท์พุท (8255)
5. ส่วนควบคุมอุปกรณ์ไฟฟ้า
6. วงจรตรวจสอบสถานะการทำงาน

3.4 คอมพิวเตอร์ส่วนบุคคล (Personal Computer, PC)

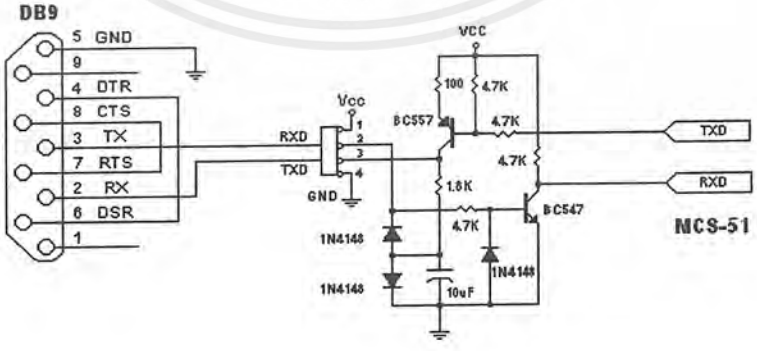
เป็นระบบที่ใช้งานแพร่หลายและคุ้นเคยที่สุด ระบบนี้ถูกออกแบบมาเพื่อใช้งานทั่วไป (General Purpose) ไม่มีการเจาะจงลักษณะการใช้งานดังนั้นการประยุกต์ใช้งานใดๆจะขึ้นกับโปรแกรมที่ใช้ โดยไม่ต้องเปลี่ยนโครงสร้างฮาร์ดแวร์ใหม่ แต่อาจจะมีการเพิ่มเติมอุปกรณ์บางส่วนเข้าไปในระบบได้ เช่น สามารถประยุกต์ใช้งานคอมพิวเตอร์ส่วนบุคคลเป็นเครื่องคำนวณหรืออาจจะใช้เป็นเครื่องส่งข้อมูลข่าวสาร หรือ เครื่องควบคุมการทำงานของเครื่องจักรได้ แต่ข้อเสียของระบบคือขนาดของระบบที่เหมาะสมนั้นควรจะเป็นระบบที่มีขนาดใหญ่พอสมควร แต่ถ้าในระบบที่มีขนาดเล็กจะไม่เหมาะสมเนื่องจากราคาของคอมพิวเตอร์ส่วนบุคคลค่อนค้างสูง จึงทำไม่เหมาะสมเท่าที่ควร จึงควรใช้ไมโครคอลโทรลเลอร์จะเหมาะสมกว่าลักษณะที่เกี่ยวข้องกับการในโครงงานนี้จะใช้คอมพิวเตอร์ส่วนบุคคลในการออกแบบและสร้างตัวโปรแกรมขึ้นมา เพื่อนำไปควบคุมอุปกรณ์ไฟฟ้าโดยตัวโปรแกรมที่ออกแบบควบคุมนี้ จะสามารถสร้างขึ้นมาจากโปรแกรมเดลไฟล์ (Delphi)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 พอร์ตสื่อสารข้อมูลอนุกรม

การสื่อสารข้อมูลกันระหว่างเครื่องคอมพิวเตอร์กับไมโครคอลโทรลเลอร์(MCS-51) จะกระทำได้โดยผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์ กับขา TXD และ RXD ของไมโครคอลโทรลเลอร์ (MCS-51) แต่เนื่องจากสัญญาณที่จากขาทั้งสองนี้ อยู่ในระดับ TTL แต่ลักษณะของการสื่อสารผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์ จะเป็นการสื่อสารแบบRS-232-C ซึ่งการส่งข้อมูลระหว่างระบบนี้ สัญญาณที่ได้จะตรงข้ามกับความเป็นจริง คือ ลอจิก "1" จะมีระดับแรงดัน -3 ถึง -25 โวลท์ ส่วนลอจิก "0" จะมีระดับแรงดัน +3 ถึง +25 โวลท์ ฉะนั้นจึงต้องมีวงจรอินเตอร์เฟสเพื่อจัดระดับสัญญาณที่เหมาะสม ซึ่งจะได้ระดับสัญญาณประมาณ $\pm$ 5 โวลท์

การเชื่อมสัญญาณระหว่างเครื่องคอมพิวเตอร์กับไมโครคอลโทรลเลอร์ (MCS-51) สามารถทำได้โดย ใช้สัญญาณ 3 เส้นคือ สาย TXD ไว้ใช้ในการส่งข้อมูลจากพอร์ต อนุกรม สาย RXD เอาไว้ใช้สำหรับรับข้อมูลจากพอร์ตอนุกรม และสายเส้นสุดท้ายคือ สาย GND เอาไว้เป็นกราว์ดของระบบการส่งข้อมูล โดยจะใช้หัวต่อชนิด DB9 หรือ DB25 เพื่อใช้ในการเป็นหัวต่อกับเครื่องคอมพิวเตอร์ทางพอร์ตอนุกรมของคอมพิวเตอร์ แต่ในการรับ-การส่งข้อมูลของเครื่องคอมพิวเตอร์นั้นโปรแกรมจะทำการแฮนด์เช็คทางฮาร์ดแวร์ด้วยแต่เนื่องจากชุดไมโครคอลโทรลเลอร์ (MCS-51) ที่ใช้ในการประมวลผลไม่สามารถทำการแฮนด์เช็คได้โดยตรง จึงต้องทำการป้อนกลับของสัญญาณนี้ให้กับตัวเองโดยต่อขา RTS เข้ากับขา CTS และต่อขา DSR เข้ากับ ขา DTR ดังรูปที่ 3.2



รูปที่ 3.2 การเชื่อมต่อของระบบ RS -232C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 ส่วนประมวลผล(MCS-51)

ส่วนควบคุมหลักเป็นส่วนที่สำคัญเป็นการควบคุมการทำงานทั้งหมดในโครงงานนี้ได้เลือกใช้ไมโครคอลโทรลเลอร์ 8051 ตระกูล(MCS-51) ทำการกำหนดค่าสัญญาณต่างๆเพื่อทำการควบคุมอุปกรณ์และทำการประมวลผล ซึ่งการติดต่อระหว่างอุปกรณ์ภายนอกจะใช้สัญญาณข้อมูล (Date Bus) และสัญญาณแอดเดรส (Address Bus) และการสื่อสารแบบอนุกรมจะส่งข้อมูลผ่านขา TX และ RX โดยสัญญาณที่เกิดขึ้นทั้งหมดจะเป็นสัญญาณในรูปแบบของสัญญาณทีทีแอล(TTL) คือที่ระดับสัญญาณลอจิก "1" จะมีระดับแรงดันเป็น 5 โวลท์ และ ที่ระดับสัญญาณลอจิก "0" จะมีระดับแรงดันเป็น 0 โวลท์

3.6.1 โครงสร้างอุปกรณ์ของบอร์ดควบคุมหลัก

โครงสร้างอุปกรณ์ของบอร์ดควบคุมหลัก มีส่วนประกอบที่สำคัญต่างๆดังนี้

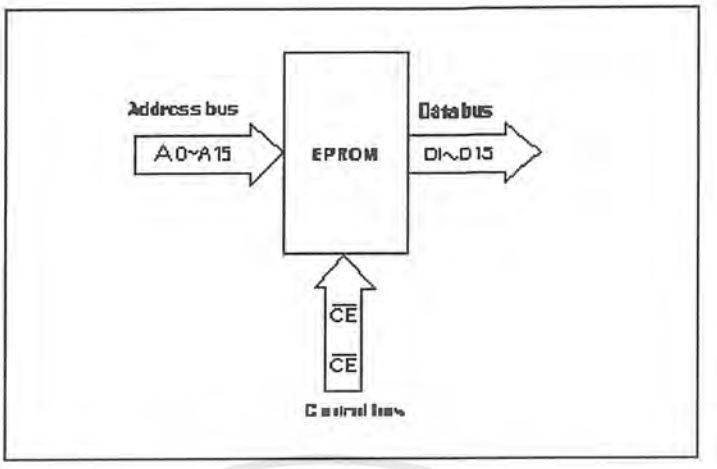
1. ไมโครคอลโทรลเลอร์ (8051)
2. อุปกรณ์แยกสัญญาณแอดเดรสกับสัญญาณข้อมูล (74LS373)
3. หน่วยความจำภายนอก (EPROM,RAM)
4. อุปกรณ์ถอดรหัสตำแหน่งสัญญาณ (74 LS138)
5. ส่วนขยายพอร์ตอินพุต/เอาท์พุต (8255)

3.6.2 การจัดวางตำแหน่งอุปกรณ์ในบอร์ดควบคุมหลัก

จากที่กล่าวมาในทางทฤษฎี ไมโครคอลโทรลเลอร์ (8031)จะประกอบด้วย พอร์ต P0 จะทำหน้าที่เป็นทั้งแอดเดรสและดาต้าโดยเมื่อซีพียูต้องการอ่านโปรแกรมจะส่งแอดเดรสไบต์ต่ำ (A0~A7)ออกทางพอร์ต P0 และส่งสัญญาณ ALE ให้กับ 74LS373 ทำการแลตช์ เพื่อเป็นแอดเดรสของหน่วยความจำ ส่วนแอดเดรสไบต์สูง(A8~A15) จะออกทางพอร์ต P2

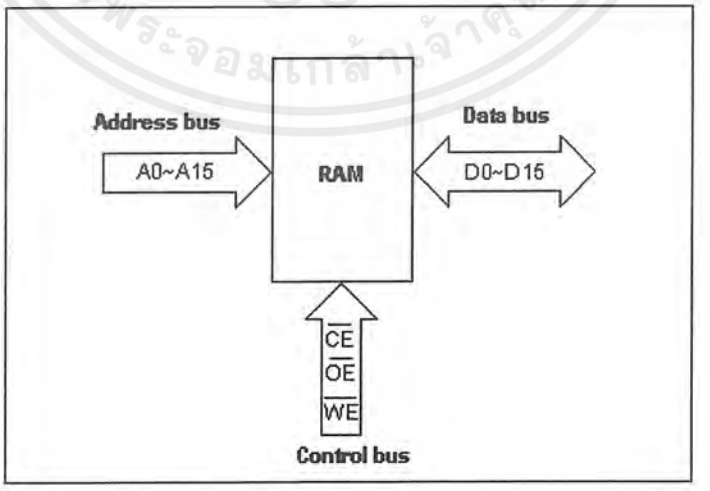
เมื่อถึงไซเกิลของเฟตช์คำสั่งที่ขา PSEN ของซีพียู จะส่งพัลส์ลอจิก "0" ออกยังบัสควบคุม (Control Bus) ส่งต่อไปให้กับเอาท์พุทอีนาเบิ้ล (Output Enable,OE) ส่วนชิปอีนาเบิ้ล(Chip Enable,CE) จะได้รับพัลส์ลอจิก "0" จากการถอดรหัสมาจาก 74LS138 จากนั้นก็จะสามารถ ป้อนตำแหน่งแอดเดรสเข้าที่สายสัญญาณแอดเดรสอินพุต(A0~A15) ของ EPROM ข้อมูลที่ถูกเก็บไว้ในตำแหน่งนั้นก็จะส่งออกมาผ่านทางบัสข้อมูล(Data Bus) เป็น (D0~D7) ผ่านไปยังไมโครคอลโทรล-เลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 บล็อกไดอะแกรมของ EPROM

ในส่วนของหน่วยความจำ RAM ก็จะมีลักษณะเช่นเดียวกับ EPROM ทั้งในส่วนของการต่อสัญญาณทางบัสแอดเดรส(Address Bus) หรือ บัสข้อมูล(Data Bus) ที่มีเพิ่มขึ้นมาคือ สายสัญญาณควบคุมการเขียนข้อมูล(WE) โดยบัสข้อมูล(Data Bus) จะเป็นทั้งอินพุต (เมื่อเขียนข้อมูลลงไป)และเอาท์พุต(เมื่ออ่านข้อมูลออกมา) ดังนั้นข้อมูลบัสจึงมีลักษณะเป็นแบบสองทิศทาง (Bi-Directional Bus) การติดต่อกันระหว่างส่วนประมวลผลซีพียู(CPU) กับหน่วยความจำ RAM สามารถกระทำด้วยสัญญาณ RD และ WR ซึ่งสัญญาณทั้งสองนี้จะออกมาก็ต่อเมื่อส่วนประมวลผลซีพียู (CPU) ทำคำสั่งเกี่ยวกับการอ่านหรือเขียนหน่วยความจำภายนอก



รูปที่ 3.4 บล็อกไดอะแกรมของ RAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ส่วนขยายพอร์ตอินพุต/เอาท์พุต (8255)

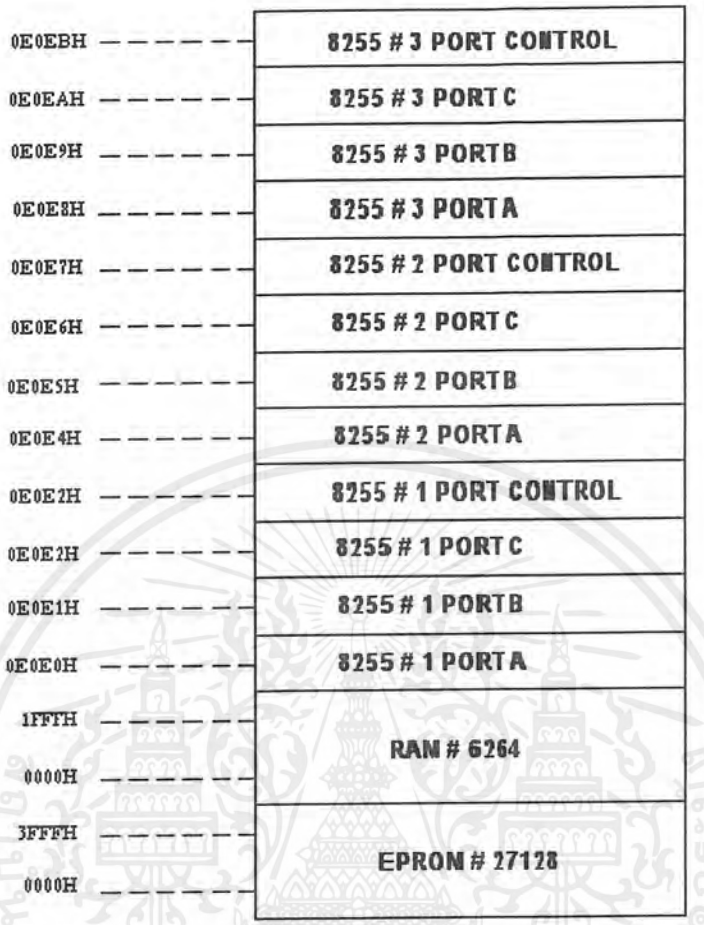
ไมโครคอลโทรลเลอร์ จะมีพอร์ต P1 ที่ทำหน้าที่เป็นอินพุตและเอาท์พุตพอร์ต ในการเชื่อมต่อกับอุปกรณ์ภายนอกได้ค่อนข้างจำกัด จึงต้องนำอุปกรณ์ 8255 ที่มีหน้าที่ขยายพอร์ตอินพุต/เอาท์พุตพอร์ตมาเชื่อมต่อกับอุปกรณ์ภายนอก เพื่อให้เพียงพอกับจำนวนอุปกรณ์แสงสว่างที่ในแต่ละพื้นที่ที่ต้องการควบคุม โดยลักษณะภายใน 8255 ประกอบด้วยพอร์ตจำนวน 3 พอร์ต คือ พอร์ต A (PA0-PA7) พอร์ต B (PB0-PB7) และพอร์ต C (PC0-PC7) ส่วนที่กำหนดการทำงานของทั้ง 3 พอร์ต คือ Group A Control โดยจะเชื่อมต่อผ่านทางบัสข้อมูล(Data Bus)ของ 8255 ส่วนที่ทำหน้าที่เชื่อมต่อระหว่างบัสข้อมูลของไมโครคอลโทรลเลอร์กับ 8255 คือ บัสข้อมูล(Data Bus) และ Read/Write Control Logic เพื่อทำการรับ-ส่งข้อมูลระหว่างกันตามระดับลอจิกของขาสัญญาณ RD และ WR ที่ได้จากการถอดรหัสแอดเดรสมาจาก 74LS138

พื้นที่ที่ต้องการควบคุมอุปกรณ์แสงสว่างในโครงงานนี้ ได้กำหนดลักษณะของพื้นที่ที่ต้องการควบคุมคือ อาคาร โดยเป็นลักษณะของอาคารที่มีจำนวน 1 ชั้น มีอุปกรณ์แสงสว่าง(ภายในอาคาร)ที่ต้องการควบคุมออกเป็น 8 ตำแหน่งและเต้ารับที่ต้องการควบคุมอีก 8 ตำแหน่ง รวมมีอุปกรณ์ไฟฟ้าที่ต้องการควบคุมเป็น 16 ตำแหน่ง ซึ่งในแต่ละส่วนจะมีลักษณะการทำงานดังนี้คือ การควบคุมการเปิด-ปิดอุปกรณ์แสงสว่าง และการตรวจสอบสถานะการทำงานของอุปกรณ์แสงสว่าง และการควบคุมการเปิด-ปิดเต้ารับ จากลักษณะการทำงานที่กล่าว นำมากำหนดหน้าที่ของการรับ-ส่งค่าข้อมูลให้กับ 8255 ในอาคารได้ดังนี้คือ กำหนดให้ พอร์ตA(PA0-PA7) จำนวน 8 บิต ส่งข้อส่งข้อมูลให้ชุดขับรีเลย์เป็นเอาท์พุตของหลอดไฟ กำหนดให้พอร์ต B (PB0-PB7) จำนวน 8 บิตส่งข้อส่งข้อมูลให้ชุดขับรีเลย์เป็นเอาท์พุตของเต้ารับและพอร์ตC(PC0-PC7) จำนวน 8 บิต เป็นการตรวจสอบสถานะของหลอดไฟ

3.8 การจัดวางตำแหน่งของหน่วยความจำและส่วนขยายพอร์ตอินพุต/เอาท์พุต (8255)

การจัดวางตำแหน่งของหน่วยความจำและส่วนขยายพอร์ตอินพุต/เอาท์พุต (8255) นั้นสามารถเลือกใช้ IC (Integrate Circuit) เบอร์ 74LS138 เป็นตัวถอดรหัสอุปกรณ์ดังกล่าว โดยได้มีการออกแบบตำแหน่งของอุปกรณ์แต่ละตัวดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 แสดงตำแหน่งของหน่วยความจำและส่วนขยายพอร์ตอินพุต/เอาท์พุต (8255)

3.9 ลักษณะการต่อพอร์ตอินพุตและพอร์ตเอาท์พุต

3.9.1 พอร์ตอินพุต

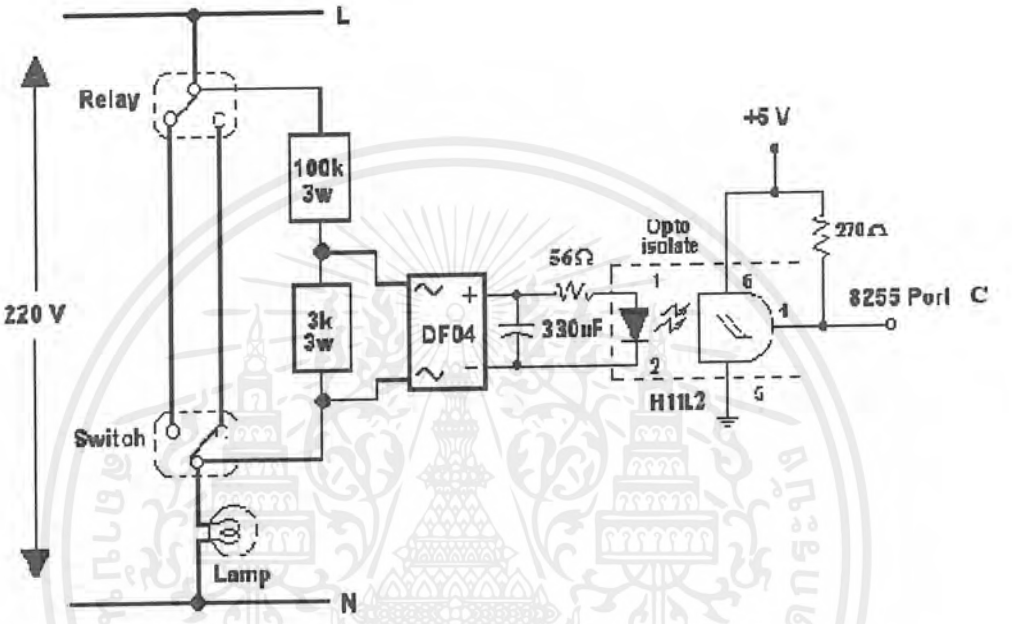
ประกอบด้วยความต้านทาน 100k โอห์ม ขนาด 3 วัตต์ และ 3k โอห์ม ขนาด 3 วัตต์ ต่ออนุกรมกัน ซึ่งทำเป็นวงจรแบ่งแรงดัน จากนั้นนำแรงดันตกคร่อมความต้านทาน 3k โอห์ม แปลงเป็นแรงดันกระแสตรงจ่ายกระแสให้ชุดออปโต้ลอจิก H11L2 โดยมีความต้านทาน 330 โอห์ม เป็นตัวลิมิตกระแสของออปโต้ลอจิก H11L2

3.9.2 การทำงานของพอร์ตอินพุต

การต่อวงจรจะใช้รีเลย์ชนิดที่มี NO และ NC แบบ Commond ร่วม และสวิทช์เป็นแบบสามทางโดยวงจรจะต่ออยู่ที่จุด Commond ของสวิทช์และ Commond ของรีเลย์ ในกรณีที่เปิดอุปกรณ์แสงสว่าง ทำให้งานแรงดันตกคร่อมชุดอินพุตเป็นศูนย์ แต่ถ้าปิดอุปกรณ์แสงสว่างจะมีกระแสไหล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผ่านชุดอินพุต ทำให้มีแรงดันตกคร่อมความต้านทาน 3k โอห์ม จึงมีกระแสจ่ายให้กับออปโต้ลอจิก H11L2 และออปโต้ลอจิก H11L2 จะทำงานแสดงสภาวะลอจิก ให้กับ Port C ของ 8255



รูปที่ 3.6 แสดงวงจรการตรวจสอบสถานะการทำงาน

3.9.3 การออกแบบชุดตรวจสถานะการทำงานของระบบแสงสว่าง

จากคุณสมบัติการทำงานของ Opto Isolate (H11L2) จะพบว่าทำงานที่พิกัด

$$V_{f(\max)} = 15V \text{ , } I_{f(on)} = 1.6mA$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ออกแบบทำงานที่พิกัด

$$V_{f(\max)} = 1V \ , I_{f(on)} = 1.5mA, V_{dc} = 1.5V$$

$$R_L = \left[\frac{\left(V_{dc} - V_f\right)}{I_f}\right] = \left[\frac{(1.5-1)}{1.5}\right] = 333\Omega$$

$$\therefore R_L = 330\Omega$$

$$V_p = \frac{V_{dc}}{0.636} = \frac{1.5}{0.636} = 2.538V$$

$$V_{\text{rms}} = 0.707V \times V_p = 0.707 \times 2.538 = 1.667V = V_{ab}$$

$$I_f = I_{dc} = 0.636 \times I_p = 0.636(0.707 \times I_{rms})$$

$$I_{rms} = \frac{I_{dc}}{(0.636 \times 0.707)} = \frac{1.5mA}{(0.636 \times 0.707)} = 1.667mA$$

- กำหนดกำลังไฟฟ้าทำงานที่ 0.5 W

$$V_{R1} = 220 - V_{ab} = 220 - 1.667 = 218.33V$$

$$I_t = \frac{P}{V_{R1}} = \frac{0.5}{218.333} = 2.29mA$$

$$R_1 = \frac{V_{R1}}{I_t} = \frac{218.333}{2.29mA} = 95k\Omega$$

$$\therefore R_1 \approx 100k\Omega$$

$$R_2 = \frac{V_{ab}}{(I_t - I_{rms})} = \frac{1.667}{(2.29mA - 1.667mA)} = 2.6k\Omega$$

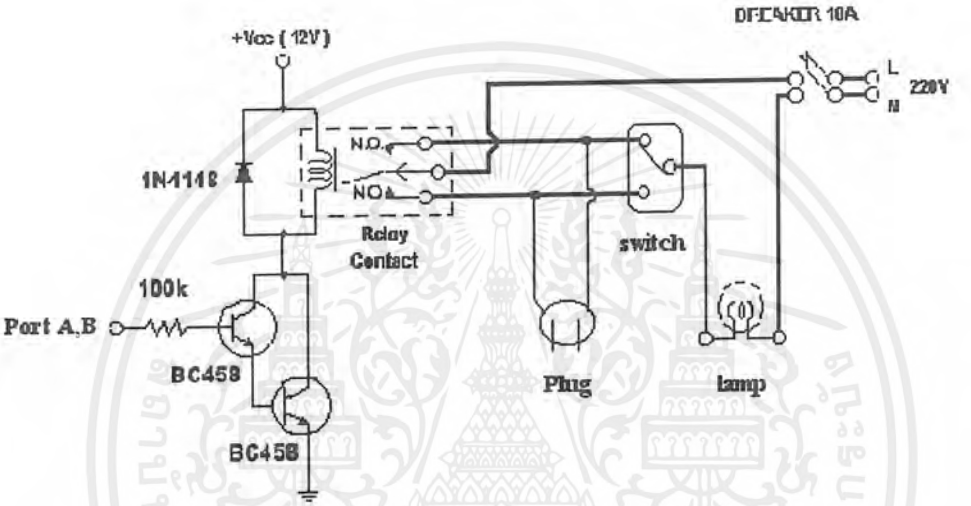
$$\therefore R_2 \approx 3k\Omega$$

จากการคำนวณใช้กำลังไฟฟ้า 0.5 W แต่การเลือกใช้งานจริง ได้เลือกกำลังไฟฟ้าที่มากกว่าค่าที่คำนวณ ซึ่งเลือกใช้กำลังไฟฟ้า 3 W

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9.4 พอร์ตเอาท์พุท

การออกแบบของพอร์ตเอาท์พุทจะให้ 8255 จ่ายกระแสเอาท์พุทน้อยที่สุด จึงได้ออกแบบให้มีทรานซิสเตอร์ 2 ตัว ต่อลักษณะเป็นวงจรดาร์ลิงตัน เพื่อช่วยจ่ายกระแสแทน 8255 ดังรูปที่ 3.7

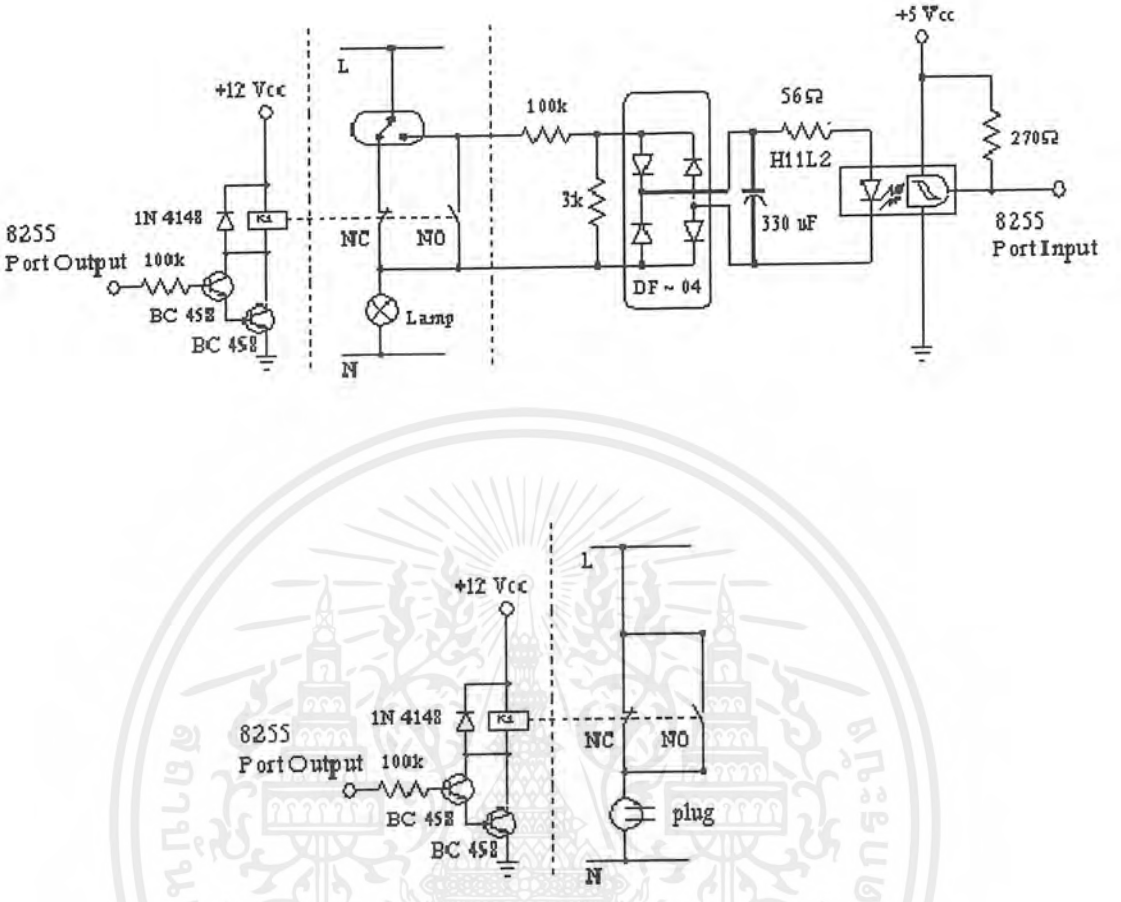


รูปที่ 3.7 แสดงวงจรดาร์ลิงตั้นควบคุมรีเลย์

3.9.5 การทำงานของพอร์ตเอาท์พุต

การทำงานของพอร์ตเอาท์พุตจะรับสภาวะลอจิกจาก Port A,Port B ของ 8255 เมื่อลอจิกมีสภาวะเป็น “1” จะมีกระแสเบสให้ทรานซิสเตอร์ตัวแรก และกระแสเบสนี้จะทำให้เกิดกระแสคอลเลตเตอร์และอิมิตเตอร์กระแสอิมิตเตอร์นี้จะไปเป็นกระแสเบสของทรานซิสเตอร์ของตัวที่สอง ทรานซิสเตอร์ตัวที่สองจะทำงาน รีเลย์จะมีกระแสไหลผ่านทำให้รีเลย์ทำงาน แต่ถ้าสภาวะมีลอจิกเป็น “0” จะไม่มีกระแสเบสจ่ายให้กับทรานซิสเตอร์ ทำให้ทรานซิสเตอร์ทั้งสองตัวไม่ทำงานและรีเลย์จะไม่ทำงานด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงการต่อวงจรรวมของชุดอินพุต ชุดเอาท์พุต และชุดตรวจสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

การทดลองนี้สามารถแบ่งออกเป็น 2 ส่วนใหญ่ๆคือ ส่วนของ Software และ Hardware

4.1 การทดลองในส่วนของโปรแกรม

โปรแกรมที่ใช้ทดสอบแบ่งเป็น 2 โปรแกรม คือ โปรแกรมทดสอบไมโครคอลโทรลเลอร์ (MCS-51) และโปรแกรมทดสอบเดลไฟล์(Delphi)

4.2 โปรแกรมทดสอบไมโครคอลโทรลเลอร์ (MCS-51)

จะแบ่งออกเป็นสองส่วนคือ โปรแกรม รับ - ส่ง ข้อมูลจากพอร์ตอนุกรม และ โปรแกรมรับข้อมูลจากชุดตรวจสถานะ

โปรแกรมรับข้อมูลจากพอร์ตอนุกรม พอร์ตอนุกรมของไมโครคอลโทรลเลอร์ (MCS-51) จะต่อกับพอร์ตอนุกรมของคอมพิวเตอร์ (Com1,Com2) ซึ่งการส่งข้อมูลจะใช้ตัวอักษรตั้งแต่ตัว a – p แต่ทางไมโครคอลโทรลเลอร์จะรับข้อมูลเป็นรหัสแอสกี้ โดยรหัสแอสกี้ของอักษรแต่ตัวแสดงในตารางที่ 4.1 เช่น ถ้าส่งข้อมูลจากคอมพิวเตอร์เป็น a ทางด้าน MCS-51 จะรับมาเป็น 61H สังเกตจากผลการทดลองโปรแกรม

```
pa1    equ 0e0e0h
pb1    equ 0e0e1h
pc1    equ 0e0e2h
pcon:  equ 0e0e3h
org 0000h
mov  dptr,#pcon
mov  a,#82h
movx @dptr,a
mov  dptr,#pa1
mov  a,r1
movx @dptr,a
mov  scon,#50h
mov  tmod,#20h
mov  th1,#0fdh
setb tr1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov  dptr,#pa1
        mov  a,#0ffh
        movx @dptr,a
inchar: jnb  ri,inchar
        clr  ri
        mov  a,sbuf
        mov  r1,a
        jnb  ri,$
        clr  ri
        mov  a,sbuf
        mov  a,r1
        mov  dptr,#pa1
        movx @dptr,a
        sjmp inchar
        end
```

ตารางที่ 4.1 แสดงรหัสแอสกี้ของอักษร A-R และ a-r

ตัวอักษร	รหัสแอสกี้	ตัวอักษร	รหัสแอสกี้
A	41H	a	61H
B	42H	b	62H
C	43H	c	63H
D	44H	d	64H
E	45H	e	65H
F	46H	f	66H
G	47H	g	67H
H	48H	h	68H
I	49H	i	69H
J	4AH	j	6AH
K	4BH	k	6BH
L	4CH	l	6CH
M	4DH	m	6DH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

N	4EH	n	6EH
O	4FH	o	6FH
P	50H	p	70H
Q	51H	q	71H
R	52H	r	72H

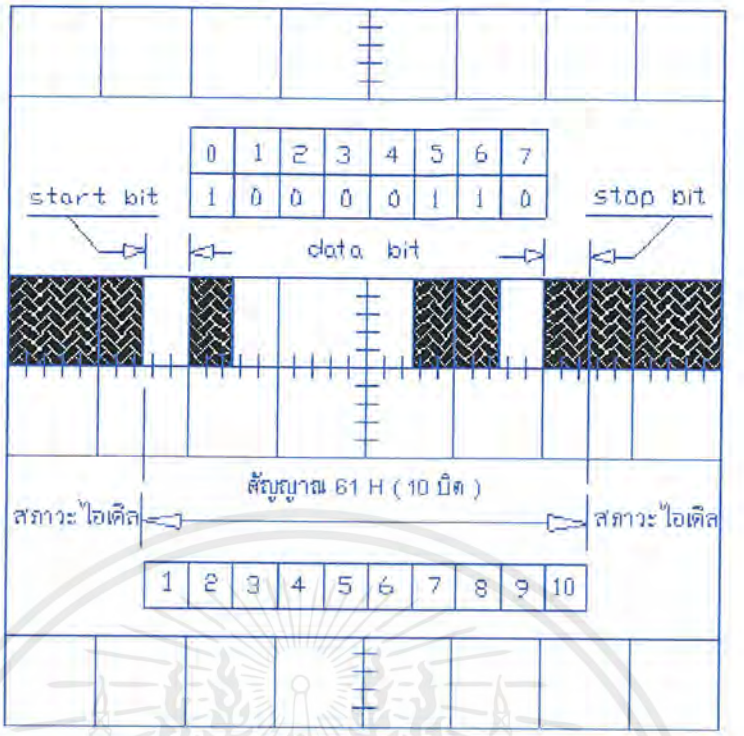
ซึ่งลักษณะของสัญญาณของ 61H ที่ส่งออกมาจากขา TXD แสดงดังรูปที่ 4.1 ซึ่งระดับของสัญญาณจะเป็นระดับแรงดันทีทีแอล คืออยู่ที่ +5V และ 0V

ลักษณะการสื่อสารแบบอนุกรมในโครงงานให้ไมโครคอลโทรลเลอร์ทำงานในโหมด1 ซึ่งลักษณะของสัญญาณนั้นจะมีการส่งบิตเริ่มต้นออกไปก่อน โดยสังเกตจากจุดที่ไม่มีการส่งข้อมูลใดๆ ออกมาจะมีสถาวะแรงดันจะเป็น +5V เมื่อส่งบิตเริ่มต้นออกมาระดับแรงดันจะเป็น 0V จากนั้นจึงส่งข้อมูลออกมา โดยจะเริ่มบิตที่ 1 ไปจนถึงบิตที่ 8 ซึ่งใช้ Volt/Div = 5V, Sec/Div = 1msและใช้การขยายในส่วนของ Sec/Div เป็น 5 เท่าดังนั้นจากรูป หากอ่านเวลาทั้งหมดได้เท่าไรจะต้องหารด้วย 5 จึงเป็นเวลาของคาบเวลาที่แท้จริง ซึ่งในโครงงานใช้อัตราบอด 9600 บิตต่อวินาที ดังนั้นการส่งข้อมูลหนึ่งบิตจะใช้เวลา 104μSec

จากรูปที่ 4.1 เป็นวัดสัญญาณในสายขณะที่มีการส่งข้อมูล 61H ซึ่งเมื่อเปลี่ยนเป็นฐาน 2 แล้วมีค่าดังนี้ 01100001b

รูปที่ 4.1 แสดงลักษณะของสัญญาณข้อมูล 61H ที่ออกจากขา TXD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงการอ่านค่าสัญญาณ 61H ที่ออกจากขา TXD

จากรูปที่ 4.2 แสดงลักษณะของสัญญาณหลังจากทำการปิดหลอดไฟ ในตำแหน่งของการควบคุมหลอดไฟหลอดที่ 1 มีผลทำให้ชุดตรวจสถานะจะส่งข้อมูลกลับไปสู่ไมโครคอลโทรเลอร์ ประมวลผลแล้วส่งข้อมูลค่า 61H ออกจากขา TXD ผ่านทางพอร์ตอนุกรมไปสู่โปรแกรมควบคุม ข้อมูล 61H ที่รับมีค่าตรงกับรหัสแอสกี้คือ 'a' โปรแกรมควบคุมที่ซึ่งจะทำการประมวลผลโดยใช้รหัส แอสกี้ ในที่นี้คือ 'a' จะทำการประมวลผล แล้วทำการเปลี่ยนค่าสีของหลอดไฟในตำแหน่งของการควบคุมหลอดไฟหลอดที่ 1 จาก "สีแดง" เป็น "สีน้ำเงิน"

สัญญาณที่ทำการส่งข้อมูลนี้จะเป็นการส่งข้อมูลกันระหว่าง ไมโครคอลโทรเลอร์เลอร์กับคอมพิวเตอร์ ดังนั้นจึงต้องทำการส่งข้อมูลเป็นแบบโหมด 1 ซึ่งมีลักษณะสัญญาณแบบ 10บิต ประกอบไปด้วย 1 บิตสตาร์ท , 8 บิตข้อมูล , 1 บิตหยุด และกำหนดอัตราบอดที่ 9600 บิต/วินาที

จากการคำนวน ณ อัตราบอดที่ 9600 บิต/วินาที จะพบว่าข้อมูล 1 บิต จะใช้เวลาดังนี้

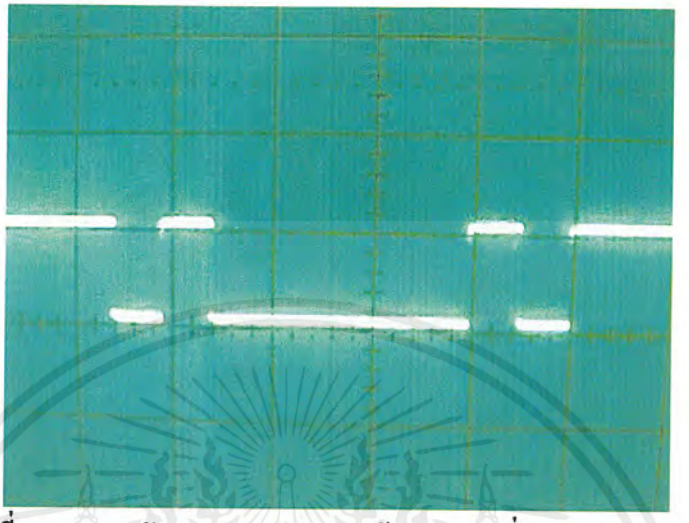
$$Data\ 1\ bit = \frac{\left(1\ bit \times 1\ \sec\right)}{9600\ bit/\sec} = 104\,\mu\sec$$

และเมื่ออ่านค่าจากหน้าจอออสซิโลสโคปตามที่ตั้งค่าสัญญาณไว้ที่ 5 Volts/Div และที่ 1ms (x5) Sec/Div จะพบว่าข้อมูลใน 1 บิต (0.5ช่อง) จะใช้เวลาดังนี้

$$Data\ 1\ bit = \frac{0.5\ m\sec}{5} = 100\,\mu\sec$$

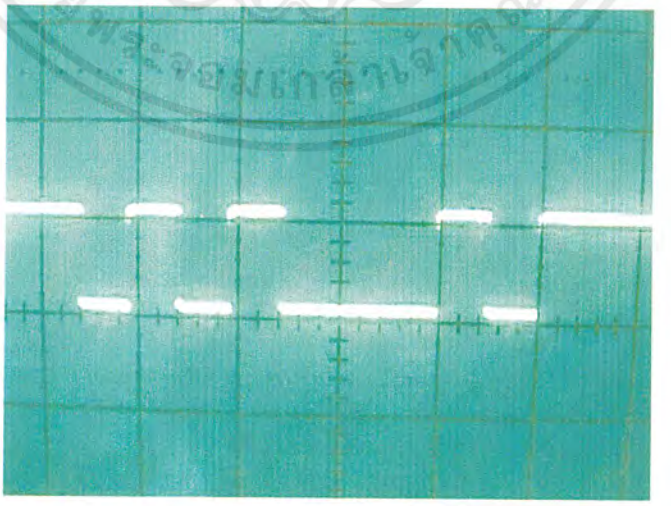
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นจะพบว่าค่าข้อมูล 1 บิต ที่อ่านค่าได้จากออสซิโลสโคป มีค่าเวลาใกล้เคียงกับค่าอัตราบอดที่กำหนดที่ 9600 บิต/วินาที



รูปที่ 4.3 แสดงลักษณะของสัญญาณข้อมูล 41H ที่ออกจากขา TXD

จากรูปที่ 4.3 แสดงลักษณะของสัญญาณหลังจากทำการเปิดหลอดไฟ ในตำแหน่งของการควบคุมหลอดไฟหลอดที่ 1 มีผลทำให้ชุดตรวจสถานะจะส่งข้อมูลกลับไปสู่ไมโครคอลโทรเลอร์ประมวลผลแล้วส่งข้อมูลค่า 41H ออกจากขา TXD ผ่านทางพอร์ตอนุกรมไปสู่โปรแกรมควบคุม ข้อมูล 41H ที่รับมีค่าตรงกับรหัสแอสกี้คือ 'A' โปรแกรมควบคุมที่ซึ่งจะทำการประมวลผลโดยใช้รหัสแอสกี้ ในที่นี้คือ 'A' จะทำการประมวลผลแล้วทำการเปลี่ยนค่าสีของหลอดไฟในตำแหน่งของหลอดไฟหลอดที่ 1 จาก "สีน้ำเงิน" เป็น "สีแดง"



รูปที่ 4.4 แสดงลักษณะของสัญญาณข้อมูล 45H ที่ออกจากขา TXD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

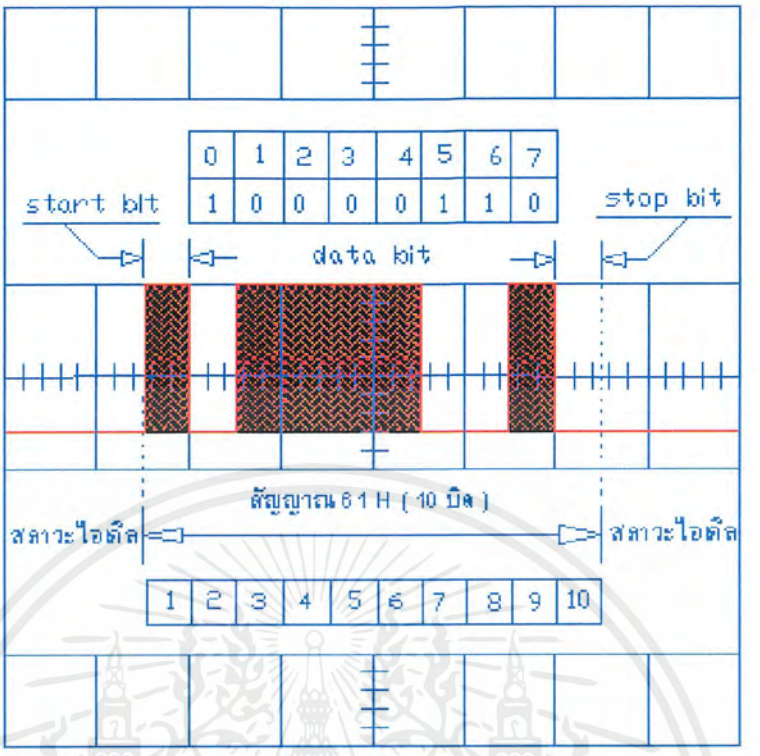
จากรูปที่ 4.4 แสดงลักษณะของสัญญาณหลังจากทำการเปิดหลอดไฟ ในตำแหน่งของการควบคุมหลอดไฟหลอดที่ 5 มีผลทำให้ชุดตรวจสถานะจะส่งข้อมูลกลับไปสู่ไมโครคอลโทรเลอร์ประมวลผลแล้วส่งข้อมูลค่า 45H ออกจากขา TXD ผ่านทางพอร์ตอนุกรมไปสู่โปรแกรมควบคุม ข้อมูล 45H ที่รับมีค่าตรงกับรหัสแอสกี้คือ 'E' โปรแกรมควบคุมที่ซึ่งจะทำการประมวลผลโดยใช้รหัส แอสกี้ ในที่นี้คือ 'E' จะทำการประมวลผลแล้วทำการเปลี่ยนค่าสีของหลอดไฟในตำแหน่งชั้นที่ 1 หลอดที่ 5 จาก "สีน้ำเงิน" เป็น "สีแดง"

เมื่อส่งข้อมูลออกจากขา TXD ของ MCS-51 ในระดับแรงดันทีทีแอล แล้วจะต้องมีการแปลงเป็นมาตราฐาน RS-232-C ส่งข้อมูลกลับมายังพอร์ตอนุกรมของคอมพิวเตอร์ เพื่อให้โปรแกรม ควบคุม แสดงผลโดยลักษณะของสัญญาณจะมี 2 ลักษณะคือ SPACE แสดงถึงไบนารี 0 หรือแรงดันไฟบวก และ MARK แสดงถึงไบนารี 1 หรือแรงดันไฟลบ

รูปที่ 4.5 แสดงลักษณะของสัญญาณ 61H ในมาตราฐาน RS-232-C

จากรูปที่ 4.6 จะสังเกตเห็นว่าบิตเริ่มต้นจะมีแรงดันเป็น +5V ซึ่งแสดงถึงลอจิกเป็น 0 และบิตที่1 จะมีสถาวะเป็น –3V ซึ่งแสดงถึงลอจิกเป็น 1 ดังนั้นเมื่อเปรียบเทียบกันระหว่างการสื่อสารแบบอนุกรมของไมโครคอลโทลเลอร์(MCS-51) กับมาตราฐาน RS-232-C แล้วจะมีลักษณะที่คล้ายคลึงกัน จะมีข้อแตกต่างกันกันอยู่บ้างที่ การสื่อสารของ MCS-51 จะมีระดับแรงดันที่ +5V กับ 0V และแรงดันไฟบวกจะแสดงถึงลอจิกที่เป็น 1 แรงดันไฟเป็นศูนย์จะแสดงถึงลอจิกที่เป็น 0 ในขณะที่มาตราฐาน RS-232-C จะใช้ระดับแรงดันที่มีค่าทั้งบวกและลบ โดยระดับแรงดันไฟที่เป็นบวกจะแสดงถึงลอจิกที่เป็น 0และระดับไฟที่เป็นลบจะแสดงถึงลอจิกที่เป็น 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงการอ่านค่าสัญญาณ 61H ที่ออกจากขา TXD ในมาตราฐาน RS-232-C

รูปที่ 4.7 แสดงลักษณะของสัญญาณ 41H ในมาตราฐาน RS-232-C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

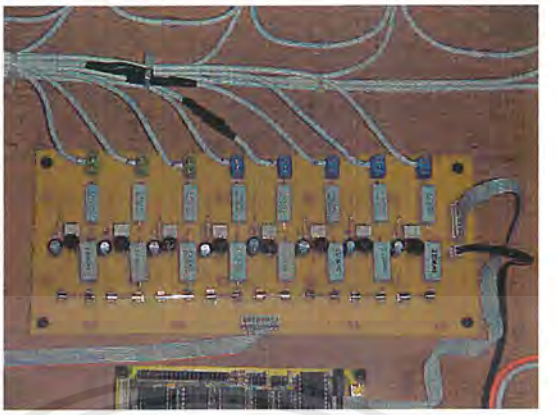
โปรแกรมรับข้อมูลจากชุดตรวจสถานะ การรับข้อมูลจากพอร์ตอินพุตจะรับข้อมูลในสภาวะลอจิก จากชุดอินพุทเพื่อให้ทราบว่าอุปกรณ์แสงสว่างที่ตำแหน่งใดเปิดหรือปิด จากนั้นจะส่งให้คอมพิวเตอร์แสดงผลบนหน้าจอโดยใช้โปรแกรมควบคุม

```
        pa1    equ 0e0e0h
        pb1    equ 0e0e1h
        pc1    equ 0e0e2h
        pcon:  equ 0e0e3h
        org    0000h
        mov    scon,#50h
        mov    tmod,#20h
        mov    th1,#0fdh
        setb   tr1
        mov    dptr,#pcon
        mov    a,#82h
        movx @dptr,a
        mov  dptr,#pa1
        mov  a,#0ffh
        mov  r6,a
        movx @dptr,a
main:   mov  dptr,#pb1
        movx a,@dptr
        mov  r3,a
main_1:mov  dptr,#pb1
        movx a,@dptr
        mov  r4,a
        cjne a,03h,out
        sjmp main_1
;out_8255
    out: mov  a,r4
        mov  r6,a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov  a,r4
        mov  r3,a
        xrl  a,r3
        mov  r5,a
        xrl   a,r6
        mov  dptr,#pa1
        mov  a,r6
        movx @dptr,a
        lcall  loop
        ljmp   main_1
loop:   mov   r0,#00
        mov   a,r5
loop_1: rrc    a
        inc    r0
        jnb    c,loop_1
        mov    a,r0
        mov    dptr,#table
        movc   a,@a+dptr
        mov    sbuf,a
        jnb    ti,$
        clr    ti
        mov    a,#00
        mov    sbuf,a
        jnb    ti,$
        clr    ti
        ret
table:  dfb  00h,61h,62h,63h,64h,65h,66h,67h,68h
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แสดงอุปกรณ์ของชุดตรวจสถานะของส่วนหลอดไฟ

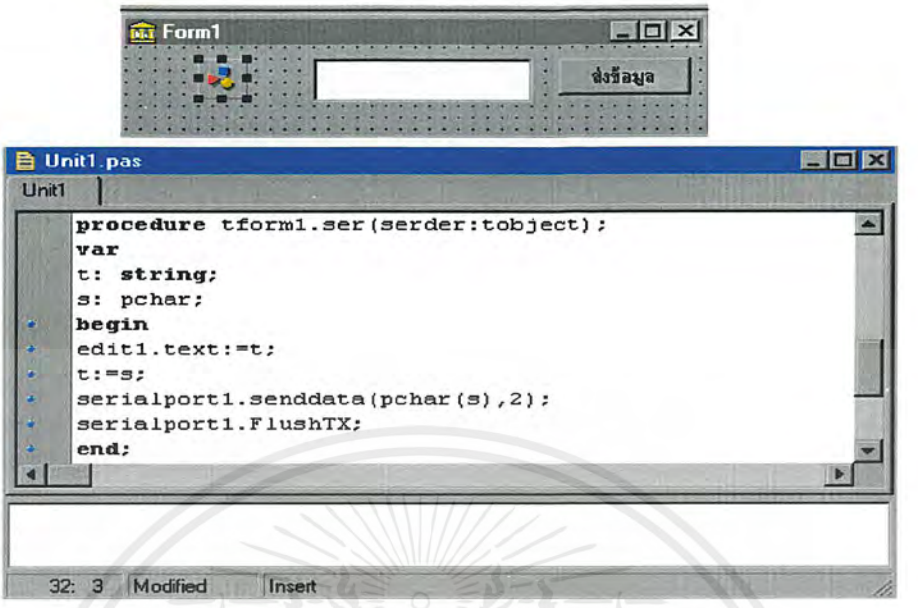
4.3 โปรแกรมการทดลองของโปรแกรมควบคุม

โปรแกรมจะแบ่งออกเป็น 3 ส่วนด้วยเช่นกัน คือ

1. ส่วนส่งข้อมูลไปยังไมโครคอลโทรลเลอร์(MCS-51)
2. ส่วนที่รับข้อมูลจากไมโครคอลโทรลเลอร์(MCS-51)
3. ส่วนแสดงผลบนหน้าจอคอมพิวเตอร์ส่วนบุคคล(PC)

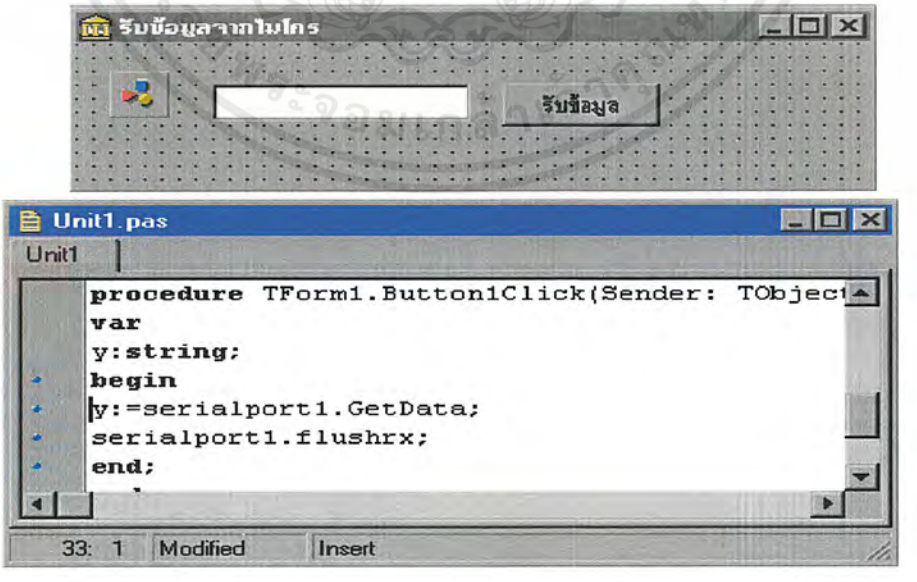
ส่วนส่งข้อมูลไปยังไมโครคอลโทรลเลอร์(MCS-51) ในส่วนนี้เมื่อผู้ใช้กดสวิทช์โปรแกรมจะตรวจสอบว่าสวิทช์ตัวใดถูกกดแล้วจะเรียกคำสั่งในการส่งข้อมูล ซึ่งข้อมูลที่ใช้ส่งนี้จะเป็นตัวอักษร a , b , c จนถึง p ซึ่งมีทั้งหมด 16 ตัวอักษรจึงเท่ากับจำนวนของโหลดทั้ง 16 จุด โดยอุปกรณ์แสงสว่างที่จุดแรกจะใช้ตัว a และเต้ารับที่จุดแรกจะใช้ตัว i เป็นรหัสในการติดต่อกับไมโครคอลโทรลเลอร์(MCS-51) และอุปกรณ์แสงสว่างตัวที่ 8 จะใช้ตัว h และเต้ารับตัวที่ 8 จะใช้ตัว p เป็นรหัสติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 โปรแกรมที่ใช้ในการส่งข้อมูล

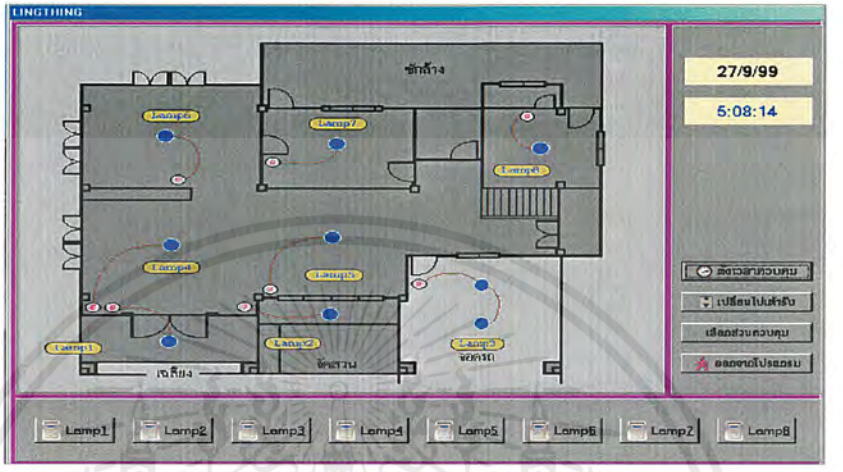
ส่วนรับข้อมูลจากไมโครคอลโทรลเลอร์(MCS-51) ตรวจข้อมูลจากชุดตรวจสถานะแล้วตรวจพบว่ามีข้อมูลใหม่เข้ามาไมโครคอลโทรลเลอร์จะทำการประมวลผล จากนั้นจะส่งผลที่ได้ไปให้เดลไฟล์ เดลไฟล์จะทำการตรวจสอบข้อมูลว่าเป็นของอุปกรณ์แสงสว่างดวงใด แล้วจึงทำการเปลี่ยนสีของอุปกรณ์แสงสว่างนั้น



รูปที่ 4.10 แสดงโปรแกรมที่ใช้รับข้อมูล

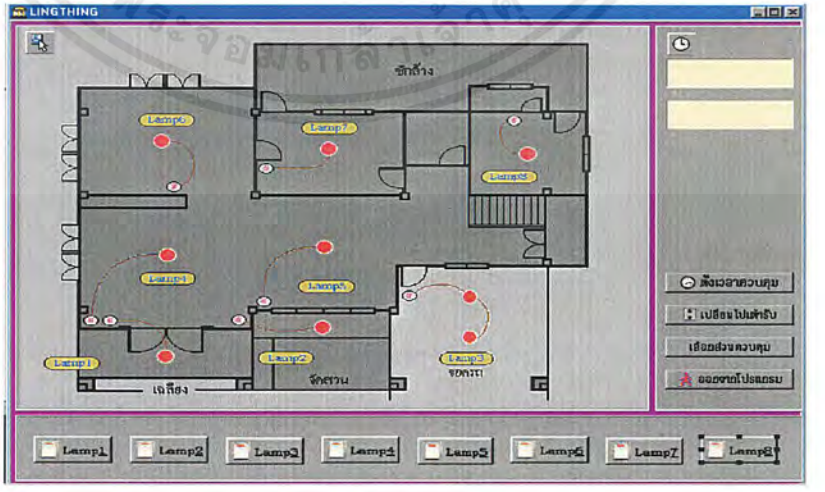
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนแสดงผลบนหน้าจอคอมพิวเตอร์ เมื่อโปรแกรมรับข้อมูลจากไมโครคอลโทรลเลอร์ (MCS-51) โปรแกรมควบคุมจะแสดงผลให้ผู้ใช้ทราบว่าอุปกรณ์แสงสว่างที่จุดใด เปิดหรือปิด โดยตำแหน่งของหลอดไฟจะเป็นการวางที่ตำแหน่งจริงซึ่งได้มีการจำลองรูปอาคาร ดังรูปที่ 4.11



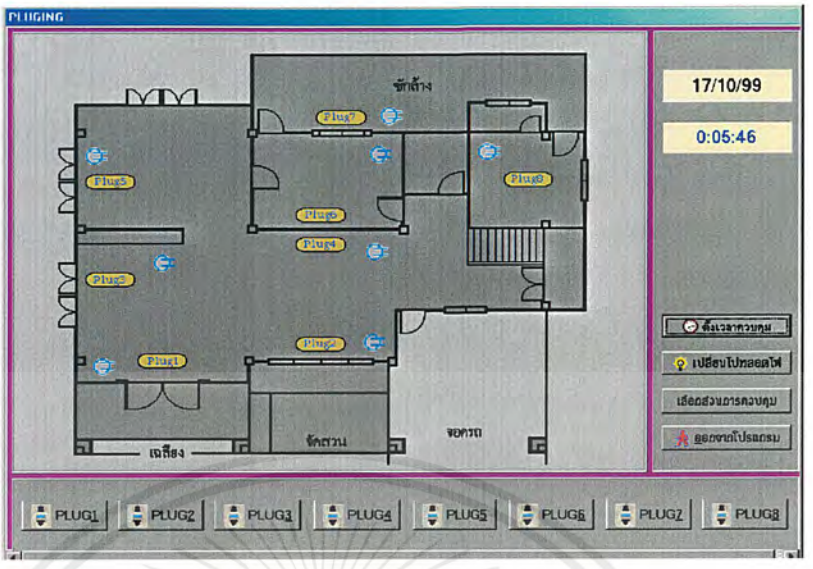
รูปที่ 4.11 แสดงการทำงานของหลอดขณะปิดของส่วนควบคุมหลอดไฟ

จากรูปที่ 4.11 เป็นการแสดงสถานะของหลอดทั้งหมดในการควบคุมหลอดไฟ ซึ่งในขณะนี้หลอดยังคงปิดอยู่ โดยสังเกตจากสีของหลอดจะเป็นสีน้ำเงิน และเมื่อมีการควบคุมจากคอมพิวเตอร์หรือจากการกดสวิทช์ของหลอดที่ควบคุมหลอดแต่ละหลอด ซึ่งการควบคุมหลอดไฟยังคงทำงานได้ตามปรกติ โดยสังเกตจากหลอดเป็นสีแดง



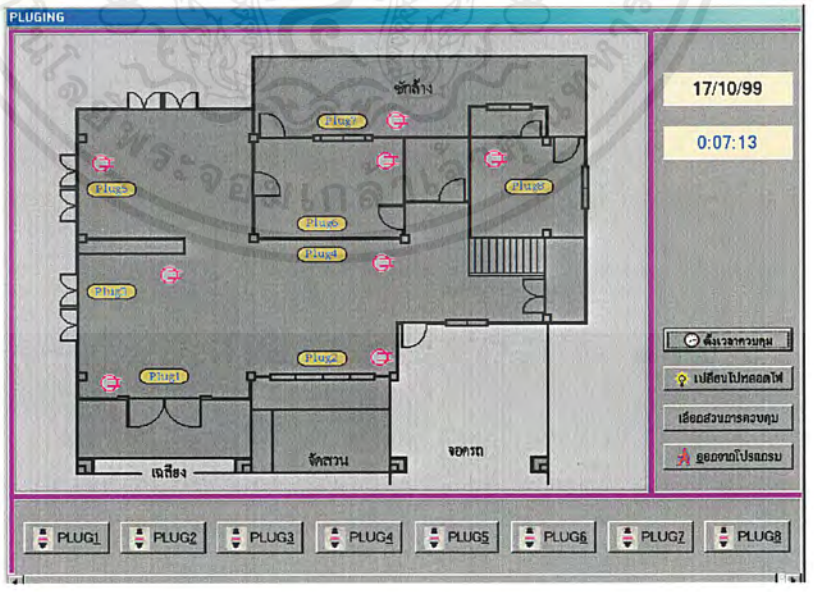
รูปที่ 4.12 แสดงการทำงานของหลอดขณะเปิดของส่วนควบคุมหลอดไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แสดงการทำงานของเต้ารับขณะปิดของส่วนควบคุมเต้ารับ

จากรูปที่ 4.13 เป็นการแสดงการทำงานของเต้ารับในส่วนที่ควบคุมเต้ารับ ซึ่งตำแหน่งการวางเต้ารับจะเป็นตำแหน่งจริงที่ติดตั้งในอาคาร โดยในรูปเป็นการแสดงขณะเต้ารับปิดอยู่ เมื่อมีการควบคุมจากทั้งคอมพิวเตอร์และจากสวิตช์ ยังคงทำงานได้ปรกติ ดังแสดงในรูปที่ 4.14



รูปที่ 4.14 แสดงการทำงานของเต้ารับขณะเปิดของส่วนควบคุมเต้ารับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองในส่วนของฮาร์ดแวร์

การทดสอบฮาร์ดแวร์สามารถแบ่ง ออกเป็น 3 ส่วนคือ ทดสอบ 8255 , ชุดตรวจสถานะและวงจรขับรีเลย์

4.5 การทดสอบ 8255

การทดสอบในส่วนนี้จะเป็นการตรวจสอบการวางตำแหน่งแอดเดรสของ 8255 ว่าสามารถทำงานได้ตรงกับแอดเดรสที่วางไว้จริงหรือไม่ ซึ่งจะวางไว้ที่ตำแหน่งแอดเดรสของ 8255 ไว้ที่ 0e0e0h - 0e0e2h โดยกำหนดพอร์ต A,B และ C เป็นพอร์ตเอาท์พุต ทั้งสามพอร์ต โปรแกรมที่ใช้ทดสอบมีดังนี้

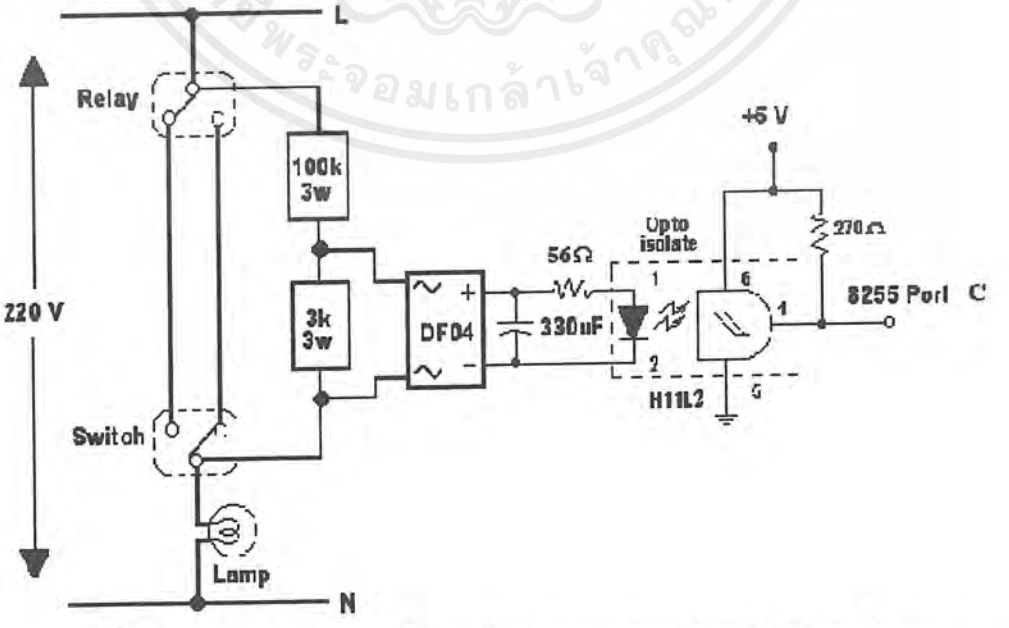
```
pa1         equ 0e0e0h
pb1         equ 0e0e1h
pc1         equ 0e0e2h
pcontrol1   equ 0e0e3h
org 0000h
main:
    mov     dptr,#pcontrol1
    mov     a,#92h
    movx    @dptr,a
    mov     dptr,#table
    mov     r0,#00
loop:
    mov     dptr,#table
    mov     a,r0
    movc    a,@a+dptr
    mov     dptr,#pc1
    movx    @dptr,a
    mov     dptr,#pa1
    movx    @dptr,a
    lcall   delay
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   delay
        lcall   delay
        inc     r0
        cjne    r0,#9,loop
        mov     r0,#00
        sjmp    loop
delay:
        mov     r1,#00
        mov     r2,#00
delay1:
        djnz    r1,$
        djnz    r2,delay1
        ret
table: dfb  00h,01h,02h,04h,08h,10h,20h,40h,80h
        end
```

4.6 การทดสอบชุดตรวจสถานะ

จากการออกแบบจะกำหนดปริมาณที่ไหลผ่านแต่ละจุด และได้นำวงจรที่ออกแบบไว้มาต่อและวัดค่าต่างดังนี้



รูปที่ 4.15 แสดงตำแหน่งการวัดแรงดันของวงจรตรวจสอบสถานะการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4..2 ผลการเปรียบเทียบแรงดันตกคร่อมแต่ละส่วนของวงจรตรวจสอบสถานะ

	ค่าจากคำนวณ (V)	ค่าจากวัด (V)
Vin	220	220
V100kΩ	218.33	216
V3kΩ	1.667	2.2
Vdc	1.5	1.4

4.7 การทดสอบวงจรขับรีเลย์

ในส่วนนี้เมื่อไมโครคอลโทรลเลอร์มีการประมวลผลและตรวจพบว่ามีการสั่งเปิด - ปิด หลอดไฟและเต้ารับ ไมโครคอลโทรลเลอร์จะส่งให้วงจรขับรีเลย์ของหลอดไฟและเต้ารับที่ตำแหน่งนั้นทำการ เปิด - ปิดหลอดไฟตามต้องการ โดยการทดสอบนี้จะใช้การรับข้อมูลจากสวิตช์มาบังคับการทำงานของรีเลย์ ซึ่งใช้โปแกรมทดสอบดังนี้

```
pa1         equ   0e0e0h
pb2         equ   0e0e1h
pc3         equ   0e0e2h
pcontrol1   equ   0e0e3h
org 0000h
mov  scon,#50h
mov  tmod,#20h
mov  th1,#0fdh
setb tr1
mov  dptr,#pcontrol1
mov  a,#92h
movx @dptr,a
main: mov  dptr,#pa1
movx a,@dptr
mov  r4,a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      cjne a,03,out
      sjmp main
out: mov  a,r4
      xrl  a,r3
      xrl  a,r5
      mov  dptr,#p1
      movx @dptr,a
      mov  a,r4
      mov  r3,a
      sjmp main
      end
```

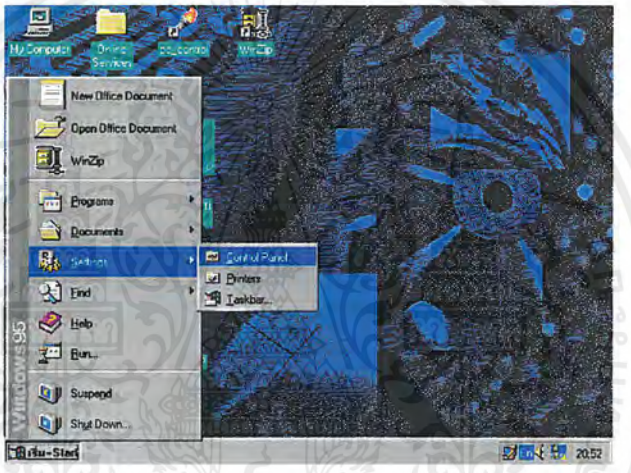
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การติดตั้งและขั้นตอนการใช้โปรแกรม

5.1 การติดตั้งโปรแกรม

การใช้โปรแกรมระบบคอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน ได้นั้นจะต้องมีการติดตั้งโปรแกรมเสียก่อน โดยจะต้องcopyโปรแกรมระบบคอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้าภายในบ้านไปไว้ที่C:\โดยสร้างไดเรกทรอรีชื่อ project 2 และจะต้องเพิ่มดาต้าเบส(Datadase) ของโปรแกรมให้กับโปรแกรมวินโดว์(Windows) เพื่อให้โปรแกรมสามารถทำงานได้ ซึ่งมีขั้นตอนดังนี้

1. เลือกเมนู Start >> Setting >> Control Panel ดังรูปที่ 5.1



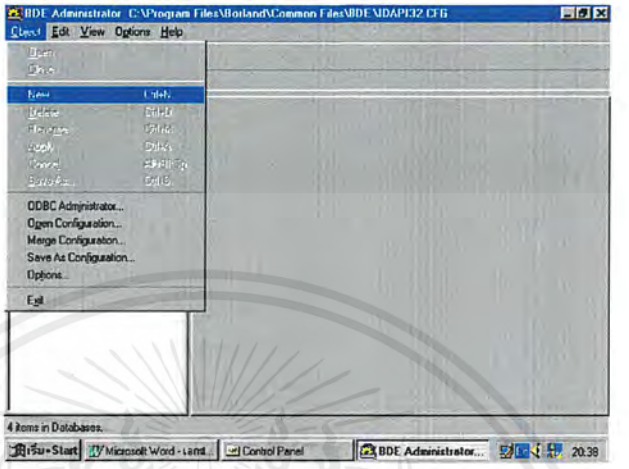
รูปที่ 5.1 แสดงการเข้าเมนู Setting เพื่อเลือก Control Panel

2. ในหน้าต่างของ Control Panel เลือก BDE Administrator ดังรูปที่ 5.2

รูปที่ 5.2 แสดงการเลือกไอคอน BDE Administrator ของหน้าต่าง Control Panel

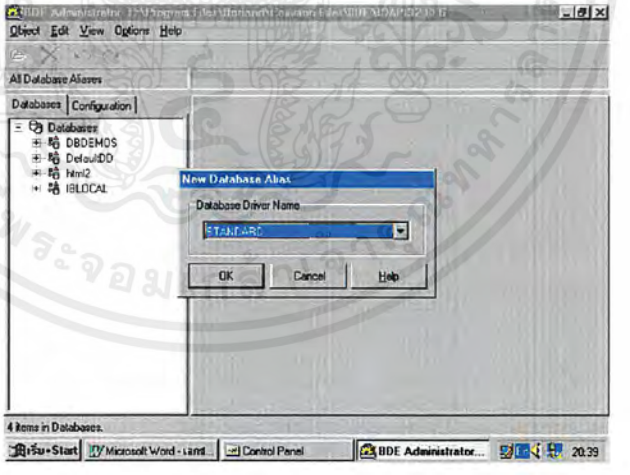
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อเลือก BDE Administrator จะปรากฏหน้าต่างที่แสดงรายละเอียดของ Database ที่มีอยู่ในเครื่องทั้งหมด ให้ทำการเลือกเมนู Object >> New ดังรูปที่ 5.3



รูปที่ 5.3 เมนูของเมนู Object ในหน้าต่าง BDE Administrator

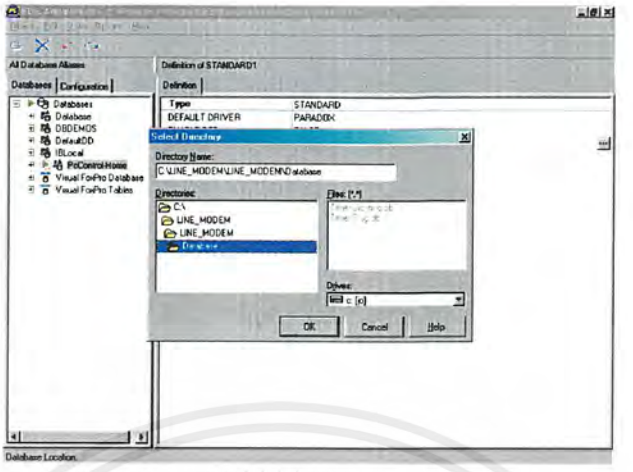
4. หลังจากเลือกNewแล้วจะมีหน้าต่างถามว่าเป็นDatabase ชนิดใดให้กำหนดเป็น STANDARD และจึงกดปุ่ม OK ดังรูปที่ 5.4



รูปที่ 5.4 แสดงการกำหนดชนิดของ Data Base เป็นแบบ STANDARD

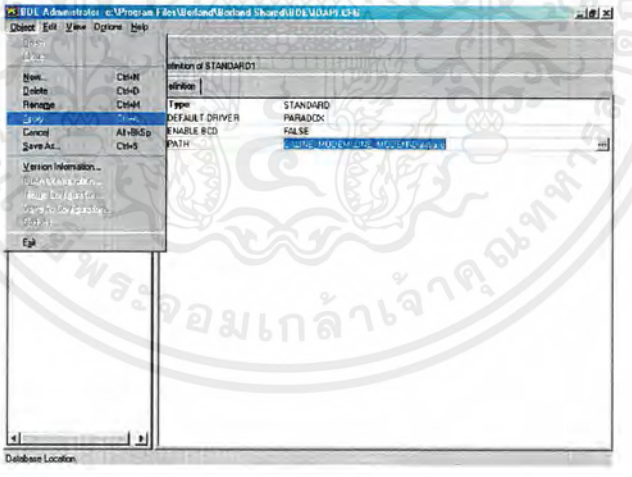
5. ให้เปลี่ยนชื่อของ STANDARD1 ที่มีการกระพริบอยู่เป็น Pc Home ส่วน PATH จะมีปุ่มทางด้านขวามือให้คลิกเพื่อกำหนดที่อยู่ของDatabaseของโปรแกรมให้เลือกไปที่ C:\project2\Database (ตำแหน่งที่ได้ copy โปรแกรมไปไว้ในตอนต้น) แล้วจึง OK ดังรูปที่ 5.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่5.5แสดงการPATHเพื่อกำหนดDatabaseไปไว้ที่C:\project 2\Database

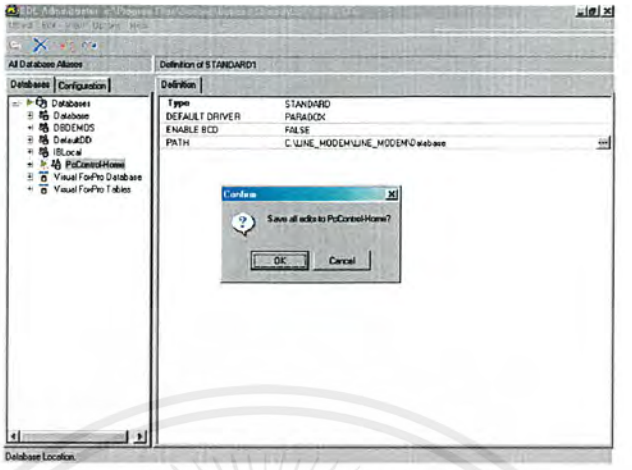
6.สังเกตที่ PATH จะต้องเป็น C:\project 2 \Database ถ้าถูกต้องให้ทำการ Apply โดยเลือกจากเมนู Object>>Apply ดังรูปที่ 5.6



รูปที่5.6 เลือกเมนู Apply หลังจากกำหนดตำแหน่งที่ PATH

7. ทำการ Save DataBase ทั้งหมดโดยเลือก OK จึงเป็นการเสร็จสมบูรณ์ดังรูปที่ 5.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.7 แสดงการ Save Database

5.2 การใช้งานโปรแกรม

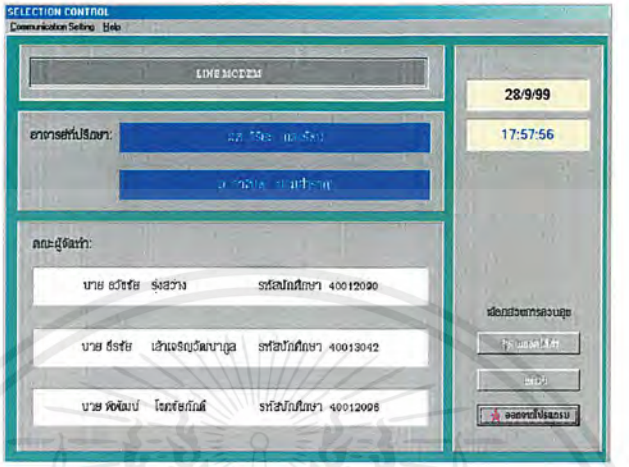
การใช้งานในส่วนนี้เป็นการใช้โปรแกรมควบคุมซึ่งมีรูปแบบ การใช้งานดังนี้

1.เมื่อเรียกโปรแกรมโปรแกรมระบบคอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน จะปรากฏหน้าต่างในการเริ่มเข้าสู่ตัวโปรแกรมควบคุมดังแสดงในรูปที่ 5.8 ดังนี้

รูปที่ 5.8 แสดงหน้าต่างการเข้าโปรแกรมควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

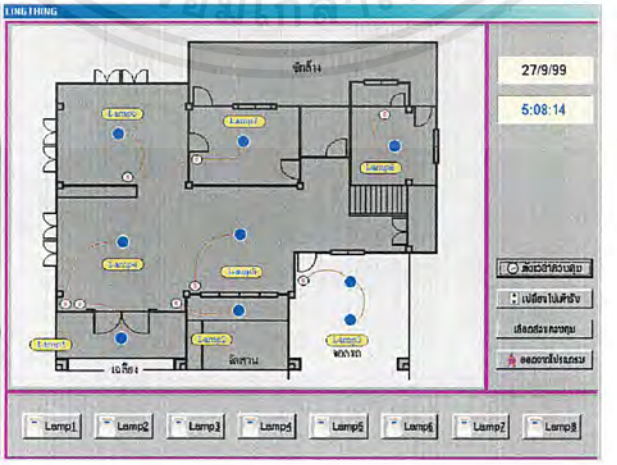
โดยในส่วนนี้จะเป็นส่วนที่นำผู้ใช้เข้าสู่โปรแกรมโดยทำการคลิกที่ปุ่มเข้าสู่โปรแกรมควบคุมเพื่อเข้าสู่หน้าต่างที่ 2 ดังแสดงในรูปที่ 5.9



รูปที่ 5.9 แสดงหน้าต่างให้ผู้ใช้เลือกเมนูและเลือกส่วนควบคุม

ในหน้าต่างที่ 2 มีเมนูประกอบไปด้วย เมนู Communication Setting และHelp ซึ่งหลังจากที่เข้าโปรแกรมแล้วจะต้องเลือกเมนู Communication Setting เพื่อเป็นการกำหนดพอร์ตที่ใช้ในการส่งข้อมูลไปยังชุดไมโครคอลโทรเลอร์ ซึ่งจะเป็นการสื่อสารแบบอนุกรมโดยจะประกอบไปด้วยพอร์ต com1,com 2,com 3 และ com4 จากนั้นจึงกดปุ่มส่วนควงบคุมหลอดไฟฟ้าหรือส่วนควบคุมเต้ารับเพื่อเลือกส่วนควบคุมที่ต้องการ

2. หลังจากกดปุ่มเลือกส่วนที่ต้องการควบคุมแล้วจะเข้าหน้าต่างถัดไป ซึ่งถ้าหากผู้ใช้เลือกส่วนของหลอดไฟก็จะแสดงหน้าต่างในการควบคุมหลอดไฟดังแสดงในรูปที่ 5.10



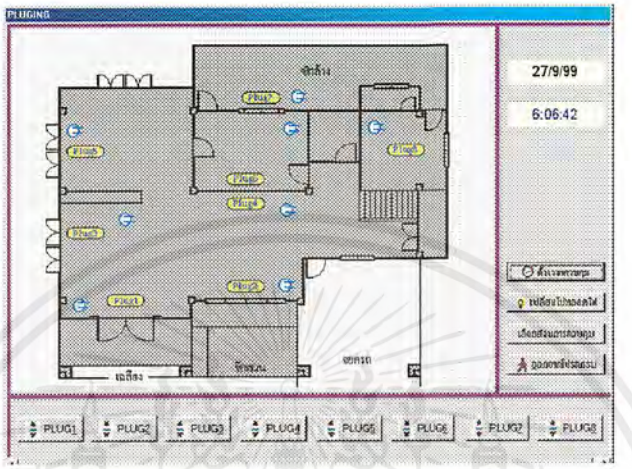
รูปที่5.10 แสดงการส่วนควบคุมหลอดไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในหน้าต่างนี้เราสามารถควบคุมโดยการสั่งที่ปุ่ม

ซึ่งเราได้จำลองการควบคุมหลอดทั้งหมดจำนวน 8 หลอด และในหน้าต่างนี้สามารถเปลี่ยนการควบคุมไปที่หน้าต่างของปลั๊กได้ที่ปุ่ม

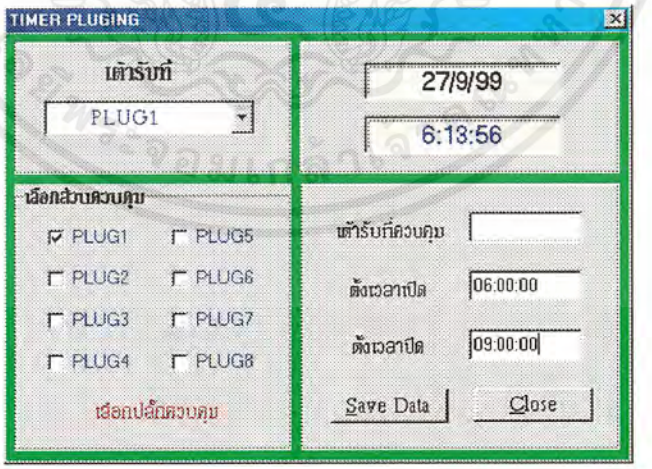
จะแสดงดังรูปที่ 5.11



รูปที่ 5.11 แสดงส่วนการควบคุมเต้ารับ

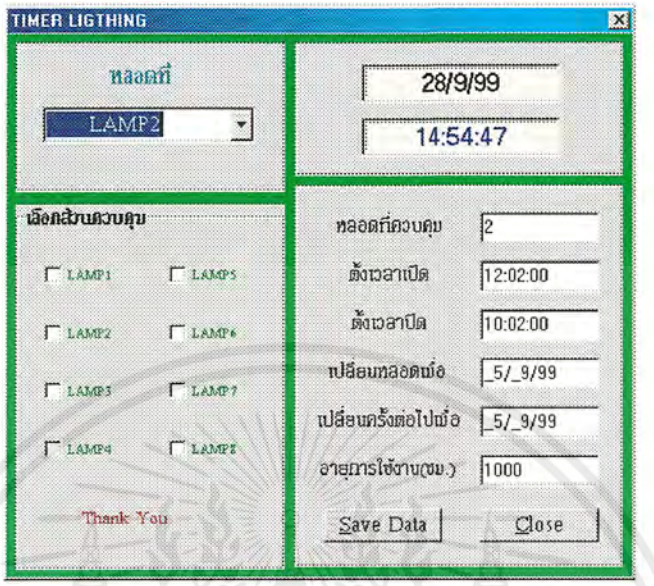
ในหน้าต่างนี้เราสามารถควบคุมการทำงานได้เหมือนกับหน้าต่างควบคุมหลอดไฟ และนอกจากนั้น แล้วเรายังกำหนดเวลา ปิด–เปิด ของเต้ารับแต่ละตัวได้ที่ปุ่ม

ซึ่งจะมีหน้าต่างการตั้งค่าเวลาดังรูปที่ 5.12



รูปที่ 5.12 แสดงส่วนการตั้งเวลาของเต้ารับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.13 แสดงส่วนการตั้งเวลาของหลอดไฟ

และจากรูปถ้าหลอดหมดอายุจะมีข้อความเตือนว่า "ถึงกำหนดเปลี่ยนหลอดใหม่" เพื่อประหยัดพลังงาน

รูปที่ 5.14 แสดงข้อความเตือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปวิจารณ์การทดลองและการประยุกต์ใช้งาน

6.1 การติดต่อระหว่างคอมพิวเตอร์กับไมโครคอลโทรลเลอร์

จากการทดลองส่งค่าจากโปรแกรมควบคุมผ่านทาง RS-232-C ให้ไมโครคอลโทรลเลอร์ (MCS-51) รับค่าและส่งค่าไปควบคุมชุดรีเลย์ โดยการจะส่งค่ารหัสที่ต้องการออกจากโปรแกรมควบคุม จากนั้นทำการตรวจสอบสภาวะการทำงานของรีเลย์รีเลย์ทำงานตรงกับค่าที่ได้รับเข้ามา การทดสอบได้ทำการทดสอบเปิด-ปิดรีเลย์อย่างต่อเนื่อง รีเลย์ยังคงทำงานได้อย่างถูกต้อง

จากนั้นให้ ไมโครคอลโทรลเลอร์ (MCS-51) รับค่าจากสวิตช์ ทำการส่งค่าผ่าน RS-232-C ให้คอมพิวเตอร์รับค่าออกมาแสดงผลทางหน้าของโปรแกรมควบคุม จากการทดลองโปรแกรมควบคุมสามารถรับค่าจากไมโครคอลโทรลเลอร์ (MCS-51) ได้อย่างถูกต้อง จากการทดลองการรับส่งค่าของคอมพิวเตอร์กับไมโครคอลโทรลเลอร์ (MCS-51) ทำงานได้อย่างถูกต้องในทุกๆ ส่วน

6.2 การทำงานของของส่วนอุปกรณ์ภายนอก

6.2.1 ชุดรีเลย์

การทำงานของชุดรีเลย์จะทำงานตามการทำงานของ 8255 โดยผ่านชุดที่ใช้ขับรีเลย์ โดยชุดที่ขับรีเลย์จะใช้ทรานซิสเตอร์ต่อคาร์ลิงตั้น จากการวัดค่าแรงเคลื่อนที่ตกคร่อมรีเลย์ขณะที่สั่งให้รีเลย์ทำงาน จะได้แรงเคลื่อนที่ตกคร่อมมีค่าประมาณ 12 V จากการทดลองการทำงานของชุดรีเลย์ชุดรีเลย์จะทำงานตามการทำงานของ 8255 ได้อย่างถูกต้อง

โดยการต่อวงจรจะต้องมีการต่อกราว์ดร่วมระหว่างชุดควบคุมจากระดับแรงดัน 5 โวลท์กับระดับแรงดัน 12 โวลท์ เข้าด้วยกัน ถ้าหากต้องการแยกระดับแรงดันทั้ง 2 ออกจากกันจะต้องมีการใช้ออปโต้(Opto Isolate)เข้าช่วย

6.2.2 ชุดตรวจสอบสถานะ

การทำงานของชุดตรวจสถานะ จะใช้วงจรแบ่งแรงดันโดยใช้แรงดันตกคร่อมความต้านทานและแปลงสัญญาณที่ได้เพื่อส่งข้อมูลให้กับพอร์ตอินพุตของ 8255 โดยการทำงานของออปโต้ที่ส่งออกมาให้กับพอร์ตอินพุตจะมีสถานะเดียวกับการทำงานของหลอด คือ หากในขณะที่หลอดเปิดออปโต้(Opto Isolate)จะให้ระดับ 5 โวลท์ แต่ถ้าหากขณะนั้นหลอดปิดอยู่ออปโต้(Opto Isolate)จะมีแรงดับ 0 โวลท์ จากการทดสอบการทำงานของชุดสถานะสามารถทำงานได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 การทำงานในการควบคุมระบบแสงสว่างและเต้ารับในอาคาร

6.3.1ในส่วนของแสงสว่างการควบคุมการทำงานใช้หลักการของสวิตช์ 3 ทาง เนื่องจากการต่อแบบนี้หากระบบควบคุมจากคอมพิวเตอร์มีปัญหาเกิดขึ้นระบบแสงสว่างยังสามารถทำงานควบคุมจากสวิตช์ของระบบได้ได้ทำการเช็คการทำงานของระบบ ระบบสามารถทำงานควบคุมระบบแสงสว่างได้อย่างถูกต้องตามที่ต้องการ เนื่องจากระบบเป็นสวิตช์ 3 ทางทำให้ไม่สามารถมองหาตำแหน่งการเปิดหรือปิดของสวิตช์จะต้องใช้ชุดตรวจสถานะเป็นตรวจเช็คและส่งผลที่ได้ไปแสดงที่หน้าจอคอมพิวเตอร์ ซึ่งระบบยังคงทำงานได้ตามปรกติ

6.3.2ในส่วนของเต้ารับจะใช้การเปิด-ปิดปกติโดยการสั่งการทางคอมพิวเตอร์ซึ่งไม่สามารถทำการตรวจสอบสภาวะการทำงานที่แท้จริงได้ ดั้งนั้นเพื่อป้องกันคอมพิวเตอร์เสียหายอันเนื่องมาจากการใช้งานใดๆจึงควรเดินสายวงจรเต้ารับแบบปกติซึ่งไม่ได้ผ่านการควบคุมจากคอมพิวเตอร์ไว้อีกชุดหนึ่งที่แต่ละจุดที่ติดตั้งเต้ารับ

6.4 แนวทางการประยุกต์ใช้งาน

การประยุกต์โครงงานนี้แบ่งออกเป็น 3 ส่วนคือ

1. การขยายโครงงานทาง Hardware

จากโครงงานนี้สามารถที่จะนำไปประยุกต์ใช้กับอาคารต่างๆได้ และสามารถที่จะขยายขีดความสามารถที่จะควบคุมสูงสุดได้ทั้งหมด 256 จุด โดยใช้ไมโครคอลโทรลเลอร์เพียงตัวเดียว ซึ่งจะมีการต่อ 8255 เข้ากับไมโครคอลโทรลเลอร์เพื่อทำหน้าที่เป็นพอร์ตขยาย โดย 8255 จำนวน 2 พอร์ต ต่อการควบคุม 8 จุดในกรณีต่อหลอดไฟและใช้ 1 พอร์ตต่อการควบคุมเต้ารับ 8 จุด ดังนั้นการขยายจุดควบคุมหลอดไฟ 128 จุดและเต้ารับ 128 จุด จะต้องใช้ 8255 ทั้งหมด 16 ตัว ต่อ ไมโครคอลโทรลเลอร์หนึ่งตัว

2. การขยายงานทางด้าน Software

เนื่องจากการสื่อสารในพอร์ตอนุกรมจะใช้บิตข้อมูลสูงสุดได้ทั้งหมด 8 บิต จึงเป็นข้อจำกัดในการเขียนโปรแกรม โดยจะต้องใช้บิตที่ 8 เป็นบิตที่แสดงถึงสถานะปัจจุบันของหลอดและจะใช้บิตที่ 1 ถึง บิตที่ 7 เป็นการแสดงถึงตำแหน่งหลอดว่าเป็นหลอดที่เท่าไร ซึ่งบิตที่ 1 ถึงบิตที่ 7 จะบอกได้สูงสุด 128 จุด การประยุกต์ทางด้านโปรแกรมแบ่งเป็นสองส่วนคือ

2.1 การพัฒนาโปรแกรมไมโครคอลโทรลเลอร์

การรับข้อมูลจากชุดตรวจสถานะ เป็นการตรวจสอบสถานะปัจจุบันของหลอดที่ตำแหน่งต่างๆ เพื่อความถูกต้องของระบบ โดยเขียนโปรแกรมอ่านข้อมูลของพอร์ต 8255 ที่ทำหน้าเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินพุต จากนั้นนำข้อมูลใหม่และข้อมูลเก่ามาเปรียบเทียบ หากมีการเปลี่ยนแปลงจะไปประมวลผลเพื่อหาสถานะปัจจุบันของหลอดที่เปลี่ยนแปลงและส่งข้อมูลที่ได้ไปยังคอมพิวเตอร์

การรับข้อมูลจากพอร์ตอนุกรม เมื่อรับข้อมูลมาแล้วจะทำการหารด้วย 8 เพื่อหาตำแหน่งพอร์ตเอาท์พุต และทำการลบด้วยค่าประจำพอร์ตนั้นเพื่อหาว่าข้อมูลเป็นของหลอดใด แล้วจึงทำการสั่งเอาท์พุตที่ต่ออยู่กับรีเลย์

2.2 การพัฒนาโปรแกรมควบคุม โปรแกรมควบคุมนั้นการส่งข้อมูลเพื่อทำการเปิด-ปิดไม่ว่าจะเป็นการเปิด-ปิดเองหรือเปิด-ปิดโดยการตั้งเวลา จะไม่ต้องคำนึงถึง บิตที่ 8 โดยจะกำหนดเป็น 1 หรือ 0 ก็ได้แต่จะต้องสอดคล้องกับโปรแกรมของไมโครคอลโทรลเลอร์ เนื่องจากโปรแกรมควบคุมจะไม่ทราบสถานะปัจจุบันของหลอดต่างๆ แต่จะอาศัยส่งข้อมูลไปยังไมโครคอลโทรลเลอร์และให้ไมโครคอลโทรลเลอร์ส่งข้อมูลในขณะนั้นมาให้โปรแกรมควบคุมแสดงผล

3 การควบคุมระยะไกล

ในโครงการนี้จะเป็นการใช้การสื่อสารระหว่างไมโครคอลโทรลเลอร์กับคอมพิวเตอร์โดยผ่าน RS-232-C ทำให้การควบคุมยังไม่ไกลเท่าที่ควร ซึ่งการพัฒนาในส่วนนี้สามารถทำให้มีประสิทธิภาพมากขึ้นได้โดยใช้การสื่อสารประเภทอื่นเข้ามาช่วยแต่การทำงานในส่วนอื่นนั้นคงทำงานตามปรกติไม่มีการเปลี่ยนแปลงแต่อย่างใดหลายวิธี แต่ขอยกตัวอย่างโดยการมาตราฐาน RS-485 ดังนี้

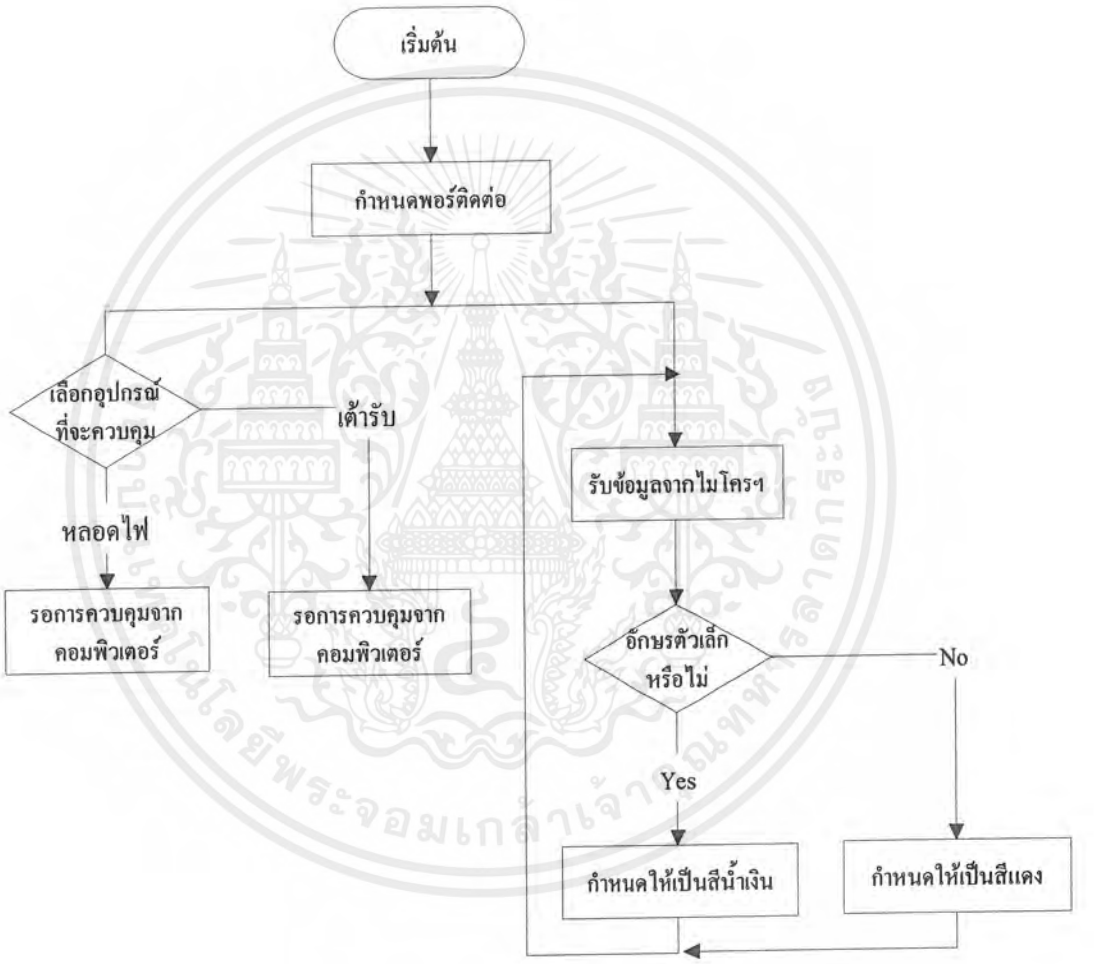
การใช้การสื่อสารโดยมาตราฐาน RS-485 เนื่องจากในมาตราฐาน RS-232-C นี้จะมีแรงดันเอาท์พุตและแรงดันอินพุตประมาณ 12-15 V และใช้สายสัญญาณทั้งหมด 3 เส้น การอ่านสัญญาณจะใช้การเทียบสัญญาณระหว่างขา TXD หรือ RXD กับสายกราวนด์ ทำให้เมื่อใช้ส่งข้อมูลในระยะทางที่ไกลขึ้นจะมีการสิ้นเปลืองและแรงดันที่ส่งมาจะลดลงตามระยะทางที่ไกลขึ้น แต่สำหรับมาตราฐาน RS-485 นั้นจะใช้แรงดันเอาท์พุตและแรงดันอินพุตประมาณ 5 V และใช้สายสัญญาณในการสื่อสาร 2 เส้น การอ่านข้อมูลในสายนั้นจะอ่านจากการเปลี่ยนขั้วของสัญญาณในสายทั้ง 2 เส้น ซึ่งใช้อ้างอิงซึ่งกันและกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

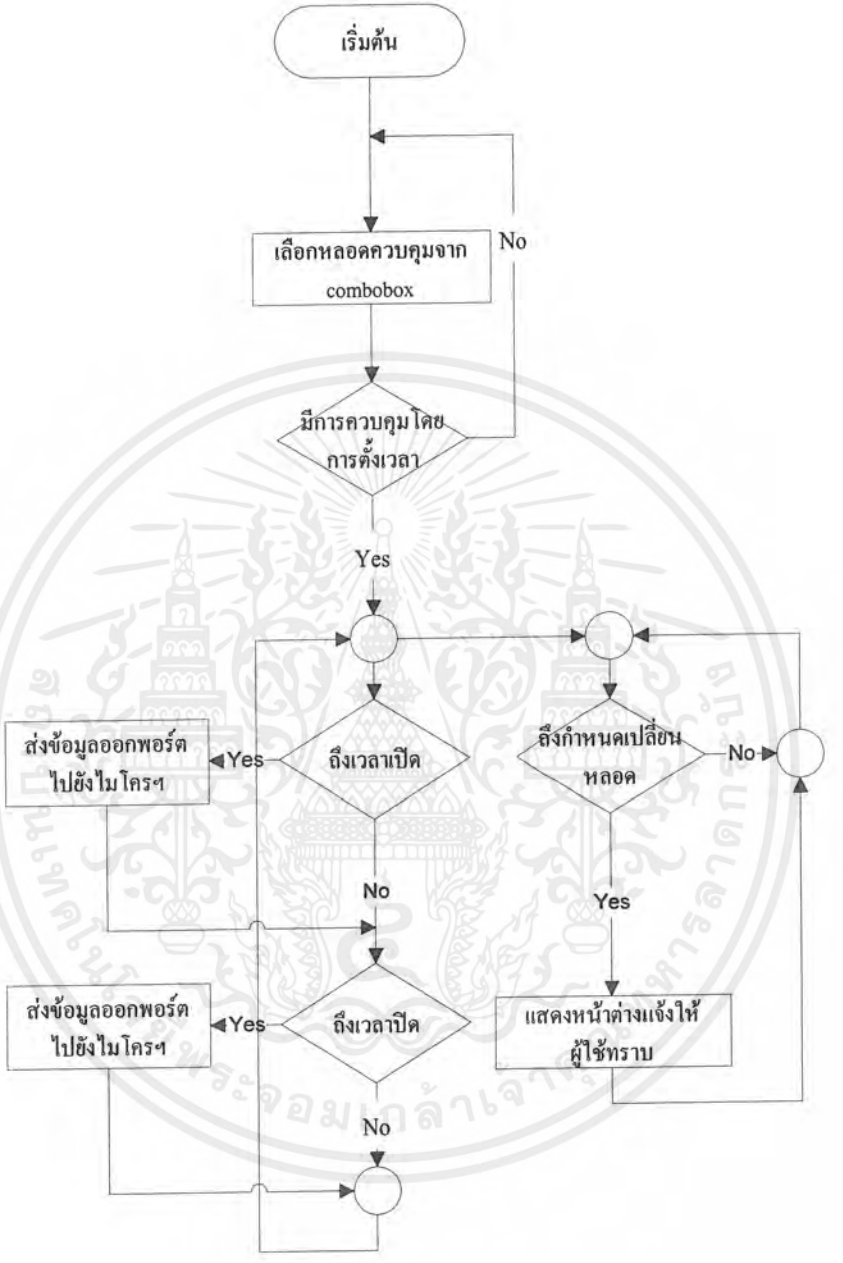
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับขั้นตอนการทำงานของโปรแกรมทั้งหมด



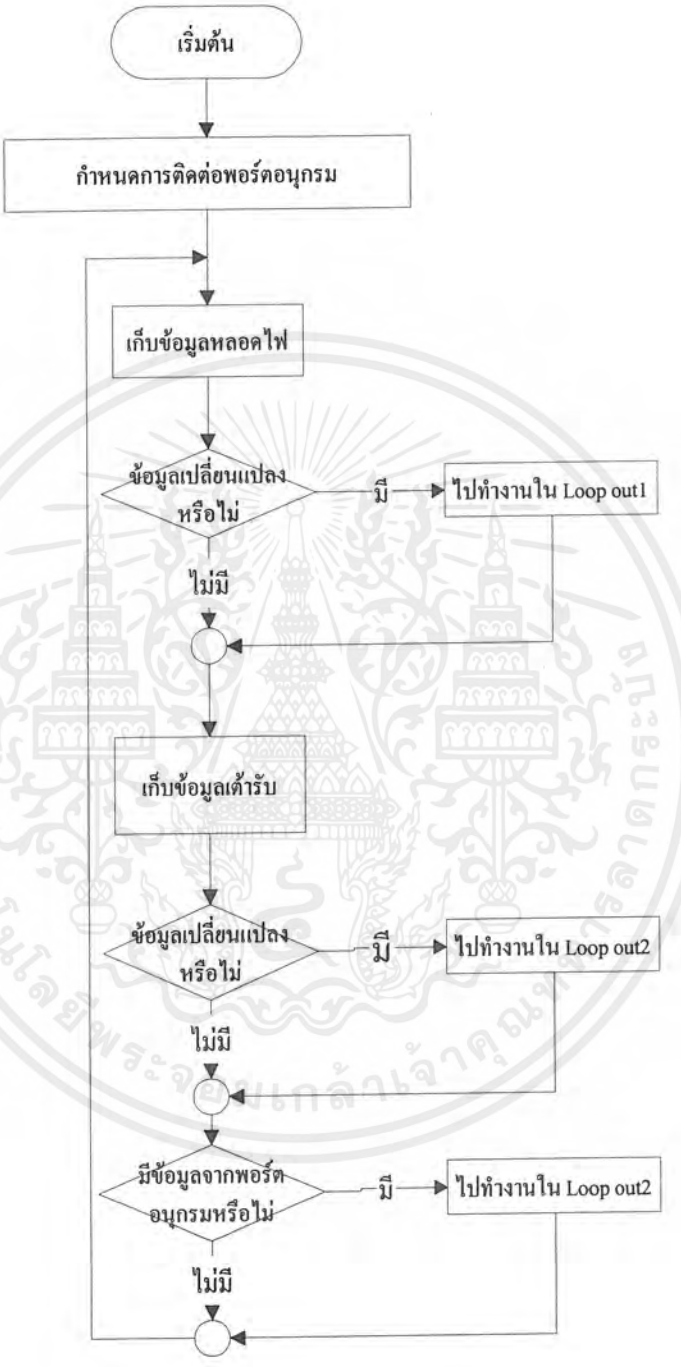
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับการทำงานของโปรแกรมเดลไฟล์ในส่วนการตั้งเวลา



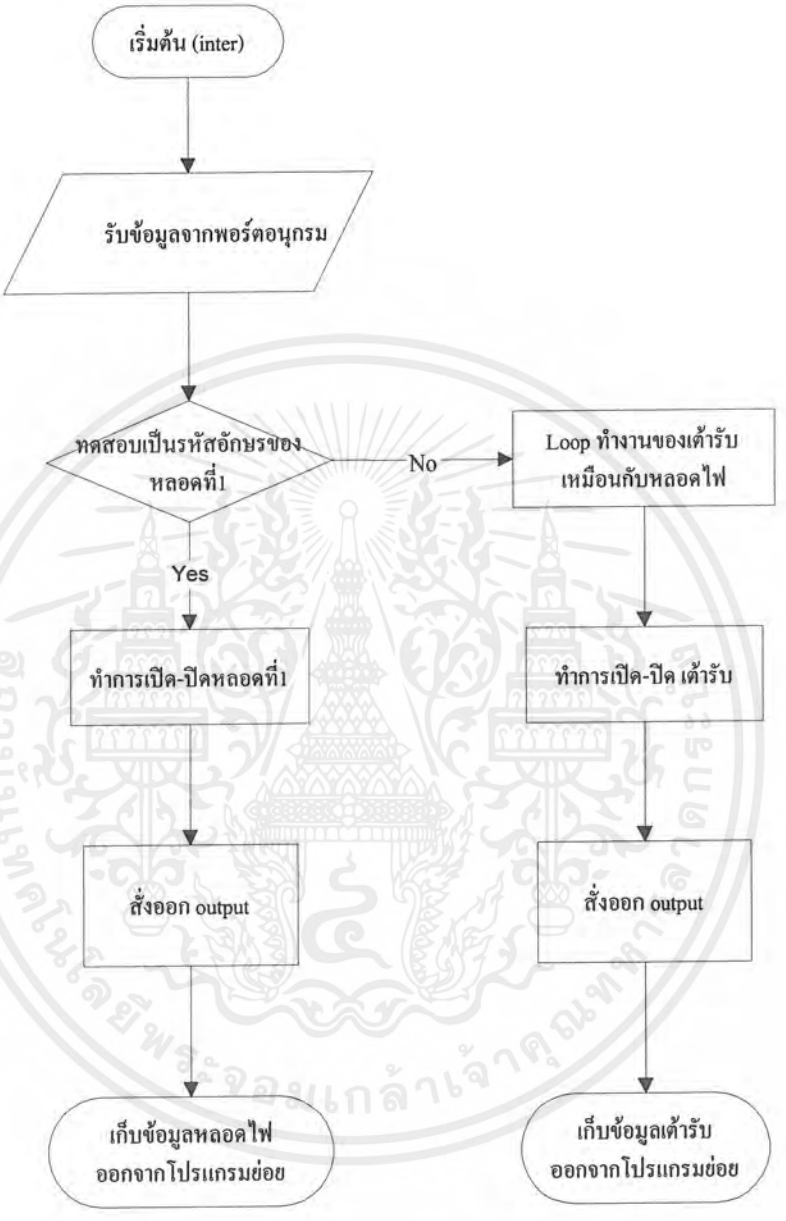
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับการทำงานของโปรแกรมหลักไมโครคอลโทรลเลอร์



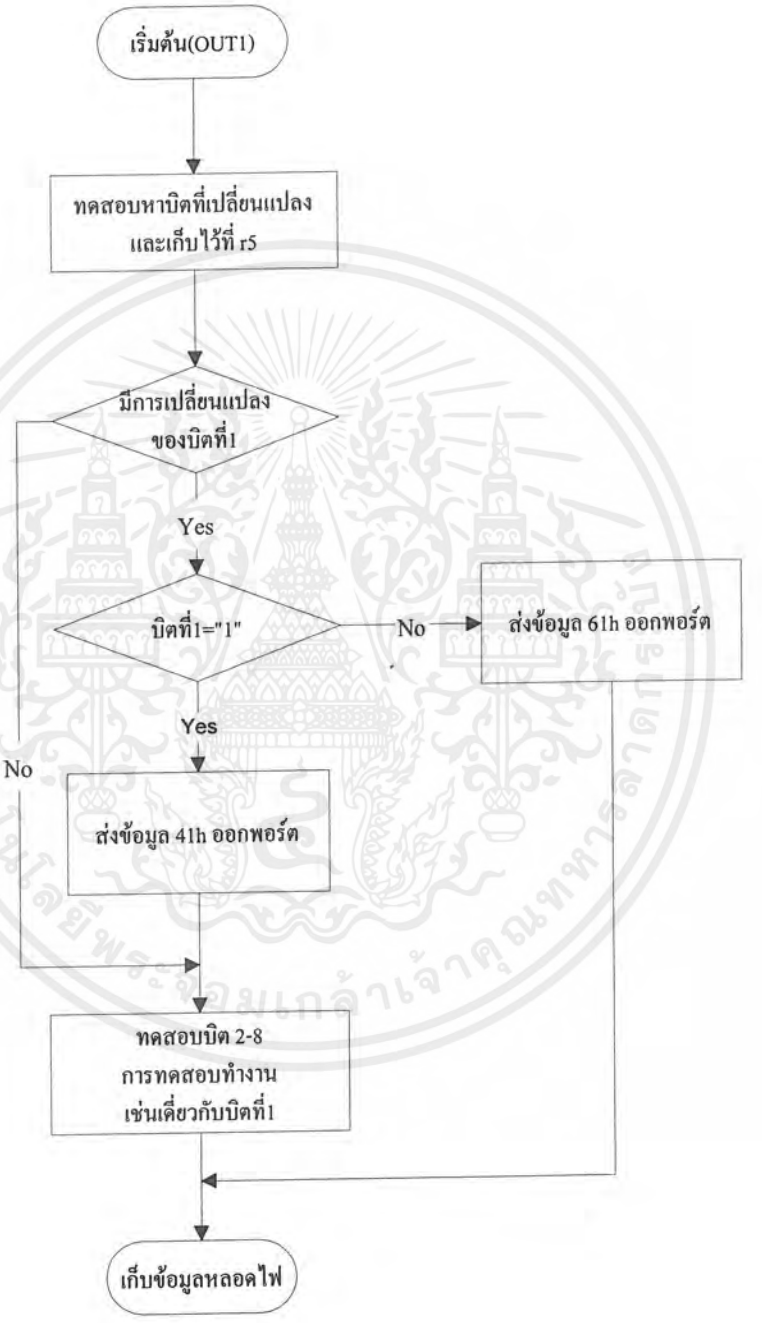
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับการทำงานของโปรแกรมย่อย inter



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับการทำงานของโปรแกรมย่อย OUT 1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมของไมโครคอลโทลเลอร์

```
        pa1     equ   0e0e0h        "Output  lamp"
        pb1     equ   0e0e1h        "Output  plug"
        pc1     equ   0e0e2h        "Input1"
        pcontrol1 equ   0e0e3h
        rb10  equ   08h
        rb11  equ   09h             " Keep stack of 1st CH1"
        rb12  equ   0ah             " Keep stack of 1st "
        rb13  equ   0bh             " Keep stack of 2 nd"
        rb14  equ   0ch
        rb15  equ   0dh
        rb16  equ   0eh
        rb17  equ   0fh
        org     0000h
main:   mov   sp,#30h
        mov    scon,#50h
        mov    tmod,#20h
        mov    th1,#0fdh
        setb   tr1
        mov    dptr,#pcontrol1
        mov    a,#89h
        movx   @dptr,a
        mov    dptr,#pa1                "Output lamp"
        mov    a,#00h
        movx   @dptr,a
        mov    dptr,#pb1                "Output plug"
        movx   @dptr,a
        mov    r6,a
        mov    rb11,a
        mov    r5,a
        mov    dptr,#pc1                " input lamp"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            movx   @dptr,a
            mov    r3,a
main_1:     lcall  delay
            lcall  delay
            lcall  delay
            mov    dptr,#pc1
            movx   a,@dptr
            mov    r4,a
            cjne   a,03h,out11
main_2:     jb     ri,inter
            sjmp   main_1
out11:      ljmp   out1
inter:      clr    ri
            mov    a,sbuf
            mov    r1,a
            mov    a,sbuf
            mov    a,r1
l1:         cjne   a,#61h,l2
            mov    a,#01h
            xrl    a,r6
            mov    r6,a
            mov    dptr,#pa1
            movx   @dptr,a
            sjmp   main_1
l2:         cjne   a,#62h,l3
            mov    a,#02h
            xrl    a,r6
            mov    r6,a
            mov    dptr,#pa1
            movx   @dptr,a
            sjmp   main_1
l3:         cjne   a,#63h,l4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov   a,#04h
        xrl   a,r6
        mov   r6,a
        mov   dptr,#pa1
        movx  @dptr,a
        sjmp  main_1
l4:     cjne  a,#64h,l5
        mov   a,#08h
        xrl   a,r6
        mov   r6,a
        mov   dptr,#pa1
        movx  @dptr,a
        sjmp  main_1
l5:     cjne  a,#65h,l6
        mov   a,#10h
        xrl   a,r6
        mov   r6,a
        mov   dptr,#pa1
        movx  @dptr,a
        sjmp  main_1
l6:     cjne  a,#66h,l7
        mov   a,#20h
        xrl   a,r6
        mov   r6,a
        mov   dptr,#pa1
        movx  @dptr,a
        sjmp  main_1
l7:     cjne  a,#67h,l8
        mov   a,#40h
        xrl   a,r6
        mov   r6,a
        mov   dptr,#pa1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        movx    @dptr,a
        ljmp    main_1
l8:     cjne    a,#68h,h21
        mov     a,#80h
        xrl     a,r6
        mov     r6,a
        mov     dptr,#pa1
        movx    @dptr,a
        ljmp    main_1
h21:    cjne    a,#69h,h22
        mov     a,#01h
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
        ljmp    main_1
h22:    cjne    a,#6ah,h23
        mov     a,#02h
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
        ljmp    main_1
h23:    cjne    a,#6bh,h24
        mov     a,#04h
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
        ljmp    main_1
h24:    cjne    a,#6ch,h25
        mov     a,#08h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
        ljmp    main_1
h25:    cjne    a,#6dh,h26
        mov     a,#10h
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
        ljmp    main_1
h26:    cjne    a,#6e4h,h27
        mov     a,#20h
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
        ljmp    main_1
h27:    cjne    a,#6fh,h28
        mov     a,#40h
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
        ljmp    main_1
h28:    cjne    a,#68h,reset1
        mov     a,#80h
        xrl     a,rb11
        mov     rb11,a
        mov     dptr,#pb1
        movx    @dptr,a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          ljmp    main_1
reset1:   cjne    a,#69h,return
          mov     r3,#00h
          ljmp    main_1
return:   ljmp    main_1
out1:     mov     a,r4
          xrl     a,r3
          mov     r5,a
          mov     a,r4
          mov     r3,a
          mov     a,r5
b0ch1:    anl     a,#01h
          cjne    a,#01h,b1ch1
          mov     a,r4
          anl     a,#01h
          cjne    a,#01h,b100
          mov     a,#41h
          lcall   send
          sjmp    b1ch1
b100:     mov     a,#61h
          lcall   send
          sjmp    b1ch1
b1ch1:    mov     a,r5
          anl     a,#02h
          cjne    a,#02h,b2ch1
          mov     a,r4
          anl     a,#02h
          cjne    a,#02h,b110
          mov     a,#42h
          lcall   send
          sjmp    b2ch1
b110:     mov     a,#62h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   send
        sjmp    b2ch1
b2ch1:  mov     a,r5
        anl     a,#04h
        cjne    a,#04h,b3ch1
        mov     a,r4
        anl     a,#04h
        cjne    a,#04h,b120
        mov     a,#43h
        lcall   send
        sjmp    b3ch1
b120:   mov     a,#63h
        lcall   send
        sjmp    b3ch1
b3ch1:  mov     a,r5
        anl     a,#08h
        cjne    a,#08h,b4ch1
        mov     a,r4
        anl     a,#08h
        cjne    a,#08h,b130
        mov     a,#44h
        lcall   send
        sjmp    b4ch1
b130:   mov     a,#64h
        lcall   send
        sjmp    b4ch1
b4ch1:  mov     a,r5
        anl     a,#10h
        cjne    a,#10h,b5ch1
        mov     a,r4
        anl     a,#10h
        cjne    a,#10h,b140
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov   a,#45h
        lcall send
        sjmp  b5ch1
b140:   mov   a,#65h
        lcall send
        sjmp  b5ch1
b5ch1:  mov   a,r5
        anl   a,#20h
        cjne  a,#20h,b6ch1
        mov   a,r4
        anl   a,#20h
        cjne  a,#20h,b150
        mov   a,#46h
        lcall send
        sjmp  b6ch1
b150:   mov   a,#66h
        lcall send
        sjmp  b6ch1
b6ch1:  mov   a,r5
        anl   a,#40h
        cjne  a,#40h,b7ch1
        mov   a,r4
        anl   a,#40h
        cjne  a,#40h,b160
        mov   a,#47h
        lcall send
        sjmp  b7ch1
b160:   mov   a,#67h
        lcall send
        sjmp  b7ch1
b7ch1:  mov   a,r5
        anl   a,#80h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        cjne    a,#80h,gomain_2
        mov     a,r4
        anl     a,#80h
        cjne    a,#80h,b170
        mov     a,#48h
        lcall   send
        sjmp    gomain_2
b170:   mov     a,#68h
        lcall   send
        sjmp    gomain_2
gomain_2: ljmp  main_2
send:   mov     sbuf,a      ;tranfer data to computer
        jnb     ti,$
        clr     ti
        mov     a,#00h
        mov     sbuf,a
        jnb     ti,$
        clr     ti
        lcall   delay
        ret
delay:  mov     rb16,#00h
        mov     rb17,#00h
delay1: djnz    rb16,$
        djnz    rb17,delay1
        ret
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมเดลไฟล์

```
unit PC_HOME1;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 StdCtrls, Buttons, ExtCtrls, jpeg;
type
 TForm1 = class(TForm)
  Image1: TImage;
  Panel1: TPanel;
  Panel2: TPanel;
  Panel3: TPanel;
  Panel4: TPanel;
  Panel5: TPanel;
  BitBtn1: TBitBtn;
  BitBtn2: TBitBtn;
  procedure BitBtn1Click(Sender: TObject);
  procedure BitBtn2Click(Sender: TObject);
 private
  { Private declarations }
 public
  { Public declarations }
 end;
var
 Form1: TForm1;
implementation
Uses PC_HOME2;
{$R *.DFM}
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
 Form2.Show;
 Form1.Visible:=False;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 end;
 procedure TForm1.BitBtn2Click(Sender: TObject);
begin
 if MessageDlg('คุณต้องการออกจากโปรแกรมหรือ?',mtConfirmation,
   [mbYes,MbNo],0)=mrYes Then
   Application.Terminate;
 end;
end.
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit PC_HOME2;
interface
uses
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
ExtCtrls, Menus, StdCtrls, Buttons, Serial;
type
TForm2 = class(TForm)
MainMenu1: TMainMenu;
CommunicationSetting1: TMenuItem;
PortCom11: TMenuItem;
N1: TMenuItem;
PortCom21: TMenuItem;
N2: TMenuItem;
PortCom31: TMenuItem;
N3: TMenuItem;
Portcom41: TMenuItem;
Panel1: TPanel;
Panel2: TPanel;
Panel3: TPanel;
Panel4: TPanel;
Panel5: TPanel;
Panel6: TPanel;
Panel7: TPanel;
Panel8: TPanel;
Panel9: TPanel;
Panel10: TPanel;
Panel11: TPanel;
Panel12: TPanel;
Panel13: TPanel;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Timer1: TTimer;
  Timer2: TTimer;
  BitBtn1: TBitBtn;
  BitBtn2: TBitBtn;
  BitBtn3: TBitBtn;
  SerialPort1: TSerialPort;
  procedure Timer1Timer(Sender: TObject);
  procedure Timer2Timer(Sender: TObject);
  procedure Send;
  procedure Receive;
  procedure PortCom11Click(Sender: TObject);
  procedure PortCom21Click(Sender: TObject);
  procedure PortCom31Click(Sender: TObject);
  procedure Portcom41Click(Sender: TObject);
  procedure BitBtn1Click(Sender: TObject);
  procedure BitBtn2Click(Sender: TObject);
  procedure BitBtn3Click(Sender: TObject);
  private
  { Private declarations }
  public
  { Public declarations }
  end;
  var
  Form2: TForm2;
  Sendup,Bring  :string ;
  s   :pchar  ;
implementation
Uses PC_HOME3,PC_HOME4;
{$R *.DFM}

procedure TForm2.Timer1Timer(Sender: TObject);
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Panel12.Caption:=Datetostr(Date);
 Panel13.Caption:=Timetostr(Time);
end;

procedure TForm2.Send;
begin
 s:=Pchar(Sendup);
 SerialPort1.SendData(Pchar(s),2);
 SerialPort1.FlushTX;
end;

procedure TForm2.Receive;
begin
 Bring:=SerialPort1.Getdata;
 SerialPort1.FlushRX;
end;

procedure TForm2.PortCom11Click(Sender: TObject);
begin
 begin
 PortCom11.Checked:=false;
 PortCom21.Checked:=false;
 PortCom31.Checked:=false;
 PortCom41.Checked:=false;
  with sender as tmenuitem do checked:=true;
  SerialPort1.CommPort:=cpCom1;
  SerialPort1.OpenPort(cpCom1);
  BitBtn1.Enabled:=true;
  BitBtn2.Enabled:=true;
  Panel11.ShowHint:=false;
 end;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm2.PortCom21Click(Sender: TObject);
begin
 begin
 PortCom11.Checked:=false;
 PortCom21.Checked:=false;
 PortCom31.Checked:=false;
 PortCom41.Checked:=false;
  with sender as tmenuitem do checked:=true;
  SerialPort1.CommPort:=cpCom2;
  SerialPort1.OpenPort(cpCom2);
  BitBtn1.Enabled:=true;
  BitBtn2.Enabled:=true;
Panel11.ShowHint:=false;
 end;
end;

procedure TForm2.PortCom31Click(Sender: TObject);
begin
 begin
 PortCom11.Checked:=false;
 PortCom21.Checked:=false;
 PortCom31.Checked:=false;
 PortCom41.Checked:=false;
  with sender as tmenuitem do checked:=true;
  SerialPort1.CommPort:=cpCom3;
  SerialPort1.OpenPort(cpCom3);
  BitBtn1.Enabled:=true;
  BitBtn2.Enabled:=true;
  Panel11.ShowHint:=false;
 end;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm2.Portcom41Click(Sender: TObject);
begin
 begin
 PortCom11.Checked:=False;
 PortCom21.Checked:=False;
 PortCom31.Checked:=False;
 PortCom41.Checked:=False;
  with Sender as TmenuItem do Checked:=True;
  SerialPort1.CommPort:=cpCom4;
  SerialPort1.OpenPort(cpCom4);
  BitBtn1.Enabled:=True;
  BitBtn2.Enabled:=True;
  Panel11.ShowHint:=False;
 end;
end;

procedure TForm2.BitBtn1Click(Sender: TObject);
begin
Sendup:='q';
Form2.Send;
Form3.Show;
Form2.Visible:=False;
end;

procedure TForm2.BitBtn2Click(Sender: TObject);
begin
Sendup:='r';
Form2.Send;
Form4.Show;
Form2.Visible:=False;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm2.BitBtn3Click(Sender: TObject);
begin
 if MessageDlg('คุณต้องการออกจากโปรแกรมหรือ?',mtconfirmation,
   [mbYes,mbNo],0)=mrYes Then
   begin
   SerialPort1.ClosePort;
   Application.Terminate;
   end;
end;

procedure TForm2.Timer2Timer(Sender: TObject);
begin
Form2.Receive;
 if Bring='' Then
   exit;
 if Bring='a' Then
   begin
    Form3.Shape1.Brush.Color:=clNavy;
Form3.BitBtn4.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
   end
 else
  if Bring='A' Then
    begin
     Form3.Shape1.Brush.Color:=clRed;
Form3.BitBtn4.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
    end
  else
   if Bring='b' Then
     begin
      Form3.Shape2.Brush.Color:=clNavy;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Form3.BitBtn5.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
      end
    else
     if Bring='B' Then
       begin
        Form3.Shape2.Brush.Color:=clRed;
Form3.BitBtn5.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
       end
      else
       if Bring='c' Then
         begin
          Form3.Shape8.Brush.Color:=clNavy;
          Form3.Shape9.Brush.Color:=clNavy;
Form3.BitBtn6.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
          end
        else
         if Bring='C' Then
          begin
           Form3.shape8.Brush.Color:=clRed;
           Form3.Shape9.Brush.Color:=clRed;
Form3.BitBtn6.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
           end
         else
          if Bring='d' Then
            begin
             Form3.Shape3.Brush.Color:=clNavy;
Form3.BitBtn7.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
            end
          else
           if Bring='D' Then
            begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Form3.Shape3.Brush.Color:=clRed;
Form3.BitBtn7.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
            end
          else
           if Bring='e' Then
            begin
             Form3.Shape4.Brush.Color:=clNavy;
Form3.BitBtn8.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
            end
          else
           if Bring='E' Then
            begin
             Form3.Shape4.Brush.Color:=clRed;
Form3.BitBtn8.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
            end
          else
           if Bring='f' Then
            begin
             Form3.Shape5.Brush.Color:=clNavy;
Form3.BitBtn9.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
            end
          else
           if Bring='F' Then
            begin
             Form3.Shape5.Brush.Color:=clRed;
Form3.BitBtn9.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
            end
          else
           if Bring='g' Then
            begin
             Form3.Shape6.Brush.Color:=clNavy;
Form3.BitBtn10.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                end
              else
               if Bring='G' Then
                 begin
                  Form3.Shape6.Brush.Color:=clRed;
Form3.BitBtn10.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
                 end
               else
                if Bring='h' Then
                  begin
                   Form3.Shape7.Brush.Color:=clNavy;
Form3.BitBtn11.Glyph.LoadFromFile('c:\project\picturepro\SWITCHCLO1.1.1.BMP');
                  end
                 else
                  if Bring='H' Then
                   begin
                    Form3.Shape7.Brush.Color:=clRed;
Form3.BitBtn11.Glyph.LoadFromFile('c:\project\picturepro\SWITCHOPE1.1.1.BMP');
                   end
 end;
 end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit PC_HOME3;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, StdCtrls, Buttons, Menus;
 type
 TForm3 = class(TForm)
  Panel1: TPanel;
  Panel2: TPanel;
  Panel3: TPanel;
  Panel4: TPanel;
  Panel5: TPanel;
  Image1: TImage;
  Timer1: TTimer;
  BitBtn1: TBitBtn;
  BitBtn2: TBitBtn;
  BitBtn3: TBitBtn;
  BitBtn4: TBitBtn;
  BitBtn5: TBitBtn;
  BitBtn6: TBitBtn;
  BitBtn7: TBitBtn;
  BitBtn8: TBitBtn;
  BitBtn9: TBitBtn;
  BitBtn10: TBitBtn;
  BitBtn11: TBitBtn;
  Shape1: TShape;
  Shape2: TShape;
  Shape3: TShape;
  Shape4: TShape;
  Shape5: TShape;
  Shape6: TShape;
  Shape7: TShape;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Shape8: TShape;
Shape9: TShape;
PopupMenu1: TPopupMenu;
BitBtn12: TBitBtn;
N1: TMenuItem;
procedure Timer1Timer(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure BitBtn10Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure Shape1MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape2MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape3MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape4MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape5MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape6MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape7MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape8MouseDown(Sender: TObject; Button: TMouseButton;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Shift: TShiftState; X, Y: Integer);
   procedure N1Click(Sender: TObject);
   procedure BitBtn12Click(Sender: TObject);
   private
   { Private declarations }
  public
   { Public declarations }
  end;
  var
  Form3: TForm3;
 implementation
Uses PC_HOME2,PC_HOME4,PC_HOME5;
{$R *.DFM}

procedure TForm3.Timer1Timer(Sender: TObject);
begin
Panel4.Caption:=DateToStr(Date);
Panel5.Caption:=TimeToStr(Time);
end;

procedure TForm3.BitBtn1Click(Sender: TObject);
begin
Sendup:='r';
Form2.Send;
Form4.Show;
Form3.Visible:=False;
end;

procedure TForm3.BitBtn2Click(Sender: TObject);
begin
Form2.Show;
Form3.Visible:=False;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;

procedure TForm3.BitBtn3Click(Sender: TObject);
begin
 if MessageDlg('คุณต้องการออกจากโปรแกรมหรือ?',mtconfirmation,
   [mbYes,mbNo],0)=mrYes Then
   Application.terminate;
end;

procedure TForm3.BitBtn4Click(Sender: TObject);
begin
if bitbtn4.Tag=0 then
  begin
  bitbtn4.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn4.Tag:=1;
  Sendup:='a';
  Form2.Send;
  end
else
  begin
  bitbtn4.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn4.Tag:=0;
  Sendup:='a';
  Form2.Send;
  end
end;

procedure TForm3.BitBtn5Click(Sender: TObject);
begin
if bitbtn5.Tag=0 then
  begin
  bitbtn5.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bitbtn5.Tag:=1;
    Sendup:='b';
    Form2.Send;
    end
  else
    begin
    bitbtn5.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
    bitbtn5.Tag:=0;
    Sendup:='b';
    Form2.Send;
    end
  end;

  procedure TForm3.BitBtn6Click(Sender: TObject);
  begin
  if bitbtn6.Tag=0 then
    begin
    bitbtn6.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
    bitbtn6.Tag:=1;
    Sendup:='c';
    Form2.Send;
    end
  else
    begin
    bitbtn6.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
    bitbtn6.Tag:=0;
    Sendup:='c';
    Form2.Send;
    end
  end;

  procedure TForm3.BitBtn7Click(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
if bitbtn7.Tag=0 then
  begin
  bitbtn7.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn7.Tag:=1;
  Sendup:='d';
  Form2.Send;
  end
else
  begin
  bitbtn7.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn7.Tag:=0;
  Sendup:='d';
  Form2.Send;
  end
end;

procedure TForm3.BitBtn8Click(Sender: TObject);
begin
if bitbtn8.Tag=0 then
  begin
  bitbtn8.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn8.Tag:=1;
  Sendup:='e';
  Form2.Send;
  end
else
  begin
  bitbtn8.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn8.Tag:=0;
  Sendup:='e';
  Form2.Send;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    end
  end;

  procedure TForm3.BitBtn9Click(Sender: TObject);
  begin
  if bitbtn9.Tag=0 then
    begin
    bitbtn9.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
    bitbtn9.Tag:=1;
    Sendup:='f';
    Form2.Send;
    end
  else
    begin
    bitbtn9.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
    bitbtn9.Tag:=0;
    Sendup:='f';
    Form2.Send;
    end
  end;

  procedure TForm3.BitBtn10Click(Sender: TObject);
  begin
  if bitbtn10.Tag=0 then
    begin
    bitbtn10.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
    bitbtn10.Tag:=1;
    Sendup:='g';
    Form2.Send;
    end
  else
    begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bitbtn10.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
    bitbtn10.Tag:=0;
    Sendup:='g';
    Form2.Send;
    end
  end;

  procedure TForm3.BitBtn11Click(Sender: TObject);
  begin
  if bitbtn11.Tag=0 then
    begin
    bitbtn11.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
    bitbtn11.Tag:=1;
    Sendup:='h';
    Form2.Send;
    end
  else
    begin
    bitbtn11.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
    bitbtn11.Tag:=0;
    Sendup:='h';
    Form2.Send;
    end
  end;

  procedure TForm3.Shape1MouseDown(Sender: TObject; Button: TMouseButton;
   Shift: TShiftState; X, Y: Integer);
  begin
  if bitbtn4.Tag=0 then
    begin
    bitbtn4.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
    bitbtn4.Tag:=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Sendup:='a';
    Form2.Send;
    end
  else
    begin
    bitbtn4.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
    bitbtn4.Tag:=0;
    Sendup:='a';
    Form2.Send;
    end
  end;

  procedure TForm3.Shape2MouseDown(Sender: TObject; Button: TMouseButton;
   Shift: TShiftState; X, Y: Integer);
  begin
  if bitbtn5.Tag=0 then
    begin
    bitbtn5.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
    bitbtn5.Tag:=1;
    Sendup:='b';
    Form2.Send;
    end
  else
    begin
    bitbtn5.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
    bitbtn5.Tag:=0;
    Sendup:='b';
    Form2.Send;
    end
  end;

  procedure TForm3.Shape8MouseDown(Sender: TObject; Button: TMouseButton;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Shift: TShiftState; X, Y: Integer);
 begin
 if bitbtn6.Tag=0 then
   begin
   bitbtn6.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
   bitbtn6.Tag:=1;
   Sendup:='c';
   Form2.Send;
   end
 else
   begin
   bitbtn6.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
   bitbtn6.Tag:=0;
   Sendup:='c';
   Form2.Send;
   end
 end;

procedure TForm3.Shape3MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
if bitbtn7.Tag=0 then
  begin
  bitbtn7.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn7.Tag:=1;
  Sendup:='d';
  Form2.Send;
  end
else
  begin
  bitbtn7.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn7.Tag:=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Sendup:='d';
  Form2.Send;
  end
end;

procedure TForm3.Shape4MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
if bitbtn8.Tag=0 then
  begin
  bitbtn8.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn8.Tag:=1;
  Sendup:='e';
  Form2.Send;
  end
else
  begin
  bitbtn8.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn8.Tag:=0;
  Sendup:='e';
  Form2.Send;
  end
end;

procedure TForm3.Shape5MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
if bitbtn9.Tag=0 then
  begin
  bitbtn9.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn9.Tag:=1;
  Sendup:='f';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Form2.Send;
  end
else
  begin
  bitbtn9.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn9.Tag:=0;
  Sendup:='f';
  Form2.Send;
  end
end;

procedure TForm3.Shape6MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
if bitbtn10.Tag=0 then
  begin
  bitbtn10.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn10.Tag:=1;
  Sendup:='g';
  Form2.Send;
  end
else
  begin
  bitbtn10.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn10.Tag:=0;
  Sendup:='g';
  Form2.Send;
  end
end;

procedure TForm3.Shape7MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
if bitbtn11.Tag=0 then
  begin
  bitbtn11.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHOPE1.1.1.bmp');
  bitbtn11.Tag:=1;
  Sendup:='h';
  Form2.Send;
  end
else
  begin
  bitbtn11.Glyph.LoadFromFile('c:\project2\picturepro\SWITCHCLO1.1.1.bmp');
  bitbtn11.Tag:=0;
  Sendup:='h';
  Form2.Send;
  end
end;

procedure TForm3.N1Click(Sender: TObject);
begin
Form5.Show;
end;

procedure TForm3.BitBtn12Click(Sender: TObject);
begin
Form5.Show;
end;
end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit PC_HOME4;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 StdCtrls, Buttons, ExtCtrls, Menus;
 type
 TForm4 = class(TForm)
  Panel1: TPanel;
  Panel2: TPanel;
  Panel3: TPanel;
  Panel4: TPanel;
  Image1: TImage;
  Panel5: TPanel;
  Timer1: TTimer;
  Panel6: TPanel;
  BitBtn1: TBitBtn;
  BitBtn2: TBitBtn;
  Image2: TImage;
  Image3: TImage;
  Image4: TImage;
  Image5: TImage;
  Image6: TImage;
  Image7: TImage;
  Image8: TImage;
  Image9: TImage;
  BitBtn4: TBitBtn;
  BitBtn5: TBitBtn;
  BitBtn6: TBitBtn;
  BitBtn7: TBitBtn;
  BitBtn8: TBitBtn;
  BitBtn9: TBitBtn;
  BitBtn10: TBitBtn;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BitBtn11: TBitBtn;
BitBtn3: TBitBtn;
BitBtn12: TBitBtn;
PopupMenu1: TPopupMenu;
N1: TMenuItem;
procedure Timer1Timer(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn4Click(Sender: TObject);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure BitBtn7Click(Sender: TObject);
procedure BitBtn8Click(Sender: TObject);
procedure BitBtn9Click(Sender: TObject);
procedure BitBtn10Click(Sender: TObject);
procedure BitBtn11Click(Sender: TObject);
procedure Image2MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image3MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image4MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image5MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image6MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image7MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image8MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image9MouseDown(Sender: TObject; Button: TMouseButton;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Shift: TShiftState; X, Y: Integer);
   procedure BitBtn12Click(Sender: TObject);
   procedure N1Click(Sender: TObject);
  private
   { Private declarations }
  public
   { Public declarations }
  end;
  var
  Form4: TForm4;
 implementation
Uses PC_HOME2,PC_HOME3,PC_HOME6;
{$R *.DFM}

procedure TForm4.Timer1Timer(Sender: TObject);
begin
Panel5.Caption:=DateToStr(date);
Panel6.caption:=TimeToStr(time);
end;

procedure TForm4.BitBtn1Click(Sender: TObject);
begin
Sendup:='q';
Form2.Send;
Form3.Show;
Form4.Visible:=False;
end;

procedure TForm4.BitBtn2Click(Sender: TObject);
begin
Form2.Show;
Form4.Visible:=False;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;

procedure TForm4.BitBtn3Click(Sender: TObject);
begin
  if MessageDlg('คุณต้องการออกจากโปรแกรมหรือ?',mtconfirmation,
    [mbYes,mbNo],0)=mrYes Then
    Application.Terminate;
end;

procedure TForm4.BitBtn4Click(Sender: TObject);
begin
if BitBtn4.Tag=0 then
  begin
  BitBtn4.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image2.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn4.Tag:=1;
  Sendup:='i';
  Form2.Send;
  end
else
  begin
  BitBtn4.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image2.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn4.Tag:=0;
  Sendup:='i';
  Form2.Send;
  end
end;

procedure TForm4.BitBtn5Click(Sender: TObject);
begin
if BitBtn5.Tag=0 then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  begin
  BitBtn5.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image3.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn5.Tag:=1;
  Sendup:='j';
  Form2.Send;
  end
else
  begin
  BitBtn5.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image3.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn5.Tag:=0;
  Sendup:='j';
  Form2.Send;
  end
end;

procedure TForm4.BitBtn6Click(Sender: TObject);
begin
if BitBtn6.Tag=0 then
  begin
  BitBtn6.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image4.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn6.Tag:=1;
  Sendup:='k';
  Form2.Send;
  end
else
  begin
  BitBtn6.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image4.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn6.Tag:=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Sendup:='k';
  Form2.Send;
  end
end;

procedure TForm4.BitBtn7Click(Sender: TObject);
begin
if BitBtn7.Tag=0 then
  begin
  BitBtn7.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image5.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn7.Tag:=1;
  Sendup:='l';
  Form2.Send;
  end
else
  begin
  BitBtn7.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image5.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn7.Tag:=0;
  Sendup:='l';
  Form2.Send;
  end
end;

procedure TForm4.BitBtn8Click(Sender: TObject);
begin
if BitBtn8.Tag=0 then
  begin
  BitBtn8.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image6.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn8.Tag:=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Sendup:='m';
  Form2.Send;
  end
else
  begin
  BitBtn8.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image6.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn8.Tag:=0;
  Sendup:='m';
  Form2.Send;
  end
end;

procedure TForm4.BitBtn9Click(Sender: TObject);
begin
if BitBtn9.Tag=0 then
  begin
  BitBtn9.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image7.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn9.Tag:=1;
  Sendup:='n';
  Form2.Send;
  end
else
  begin
  BitBtn9.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image7.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn9.Tag:=0;
  Sendup:='n';
  Form2.Send;
  end
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm4.BitBtn10Click(Sender: TObject);
begin
if BitBtn10.Tag=0 then
  begin
  BitBtn10.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image8.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn10.Tag:=1;
  Sendup:='o';
  Form2.Send;
  end
else
  begin
  BitBtn10.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image8.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn10.Tag:=0;
  Sendup:='o';
  Form2.Send;
  end
end;

procedure TForm4.BitBtn11Click(Sender: TObject);
begin
if BitBtn11.Tag=0 then
  begin
  BitBtn11.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image9.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn11.Tag:=1;
  Sendup:='p';
  Form2.Send;
  end
else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  begin
  BitBtn11.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image9.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn11.Tag:=0;
  Sendup:='p';
  Form2.Send;
  end
end;

procedure TForm4.Image2MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
if BitBtn4.Tag=0 then
  begin
  BitBtn4.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image2.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn4.Tag:=1;
  Sendup:='i';
  Form2.Send;
  end
else
  begin
  BitBtn4.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image2.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn4.Tag:=0;
  Sendup:='i';
  Form2.Send;
  end
end;

procedure TForm4.Image3MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 begin
 if BitBtn5.Tag=0 then
   begin
   BitBtn5.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
   Image3.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
   BitBtn5.Tag:=1;
   Sendup:='j';
   Form2.Send;
   end
 else
   begin
   BitBtn5.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
   Image3.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
   BitBtn5.Tag:=0;
   Sendup:='j';
   Form2.Send;
   end
 end;

 procedure TForm4.Image4MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
 begin
 if BitBtn6.Tag=0 then
   begin
   BitBtn6.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
   Image4.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
   BitBtn6.Tag:=1;
   Sendup:='k';
   Form2.Send;
   end
 else
   begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  BitBtn6.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image4.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn6.Tag:=0;
  Sendup:='k';
  Form2.Send;
  end
end;

procedure TForm4.Image5MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
if BitBtn7.Tag=0 then
  begin
  BitBtn7.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image5.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn7.Tag:=1;
  Sendup:='l';
  Form2.Send;
  end
else
  begin
  BitBtn7.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image5.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn7.Tag:=0;
  Sendup:='l';
  Form2.Send;
  end
end;

procedure TForm4.Image6MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if BitBtn8.Tag=0 then
  begin
  BitBtn8.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image6.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn8.Tag:=1;
  Sendup:='m';
  Form2.Send;
  end
else
  begin
  BitBtn8.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image6.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn8.Tag:=0;
  Sendup:='m';
  Form2.Send;
  end
end;

procedure TForm4.Image7MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
if BitBtn9.Tag=0 then
  begin
  BitBtn9.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image7.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn9.Tag:=1;
  Sendup:='n';
  Form2.Send;
  end
else
  begin
  BitBtn9.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Image7.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn9.Tag:=0;
  Sendup:='n';
  Form2.Send;
  end
end;

procedure TForm4.Image8MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
if BitBtn10.Tag=0 then
  begin
  BitBtn10.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image8.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn10.Tag:=1;
  Sendup:='o';
  Form2.Send;
  end
else
  begin
  BitBtn10.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image8.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn10.Tag:=0;
  Sendup:='o';
  Form2.Send;
  end
end;

procedure TForm4.Image9MouseDown(Sender: TObject; Button: TMouseButton;
 Shift: TShiftState; X, Y: Integer);
begin
if BitBtn11.Tag=0 then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  begin
  BitBtn11.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
  Image9.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
  BitBtn11.Tag:=1;
  Sendup:='p';
  Form2.Send;
  end
else
  begin
  BitBtn11.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
  Image9.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
  BitBtn11.Tag:=0;
  Sendup:='p';
  Form2.Send;
  end
end;

procedure TForm4.BitBtn12Click(Sender: TObject);
begin
Form6.Show;
end;

procedure TForm4.N1Click(Sender: TObject);
begin
Form6.Show;
end;
end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit PC_HOME5;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, StdCtrls, Mask, Grids, Db, DBTables;
 type
TForm5 = class(TForm)
  Panel1: TPanel;
  Label1: TLabel;
  ComboBox1: TComboBox;
  GroupBox1: TGroupBox;
  Label2: TLabel;
  CheckBox1: TCheckBox;
  CheckBox2: TCheckBox;
  CheckBox3: TCheckBox;
  CheckBox4: TCheckBox;
  CheckBox5: TCheckBox;
  CheckBox6: TCheckBox;
  CheckBox7: TCheckBox;
  CheckBox8: TCheckBox;
  Panel2: TPanel;
  Panel3: TPanel;
  Panel4: TPanel;
  Panel5: TPanel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  Label8: TLabel;
  Edit1: TEdit;
  MaskEdit1: TMaskEdit;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    MaskEdit2: TMaskEdit;
    MaskEdit3: TMaskEdit;
    MaskEdit4: TMaskEdit;
    MaskEdit5: TMaskEdit;
    Button1: TButton;
    Button2: TButton;
    Timer1: TTimer;
    StringGrid1: TStringGrid;
    Table1: TTable;
    DataSource1: TDataSource;
    procedure Button2Click(Sender: TObject);
    procedure CopyData;
    procedure EditHide;
    procedure EditShow;
    procedure TestCheckBox;
    procedure ComboBox1Change(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure CheckDate2;
    procedure Checkdate1;
    private
    { Private declarations }
  public
    { Public declarations }
  end;
  var
  Form5: TForm5;
  rowtable,timeday3:Integer;
 implementation
Uses PC_HOME2,PC_HOME3;
{$R *.DFM}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm5.Button2Click(Sender: TObject);
begin
Close;
end;

procedure TForm5.CopyData;
var
row:Integer;
begin
for row:= 1 to 8 do
 begin
   Table1.RecNo:=row;
   StringGrid1.Cells[0,row-1]:=
    IntToStr(Table1.FieldByName('Lamp Number').AsInteger);
   StringGrid1.Cells[1,row-1]:=
    TimeToStr(Table1.FieldByName('Timer On').AsDateTime);
   StringGrid1.Cells[2,row-1]:=
    TimeToStr(Table1.FieldByName('Timer Off').AsDateTime);
   StringGrid1.Cells[3,row-1]:=
    DateToStr(Table1.FieldByName('Change In').AsDateTime);
   StringGrid1.Cells[4,row-1]:=
    DateToStr(Table1.FieldByName('Change Out').AsDateTime);
   StringGrid1.Cells[5,row-1]:=
    IntToStr(Table1.FieldByName('Age Of Lamp').AsInteger);
 end;
Table1.RecNo:=ComboBox1.ItemIndex+1;
end;

procedure TForm5.EditHide;
begin
Edit1.Enabled:=False;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MaskEdit1.Enabled:=False;
MaskEdit2.Enabled:=False;
MaskEdit3.Enabled:=False;
MaskEdit4.Enabled:=False;
MaskEdit5.Enabled:=False;
end;

procedure TForm5.EditShow;
begin
Edit1.Enabled:=True;
MaskEdit1.Enabled:=True;
MaskEdit2.Enabled:=True;
MaskEdit3.Enabled:=True;
MaskEdit4.Enabled:=True;
MaskEdit5.Enabled:=True;
end;

procedure TForm5.TestCheckBox;
begin
 case rowtable of
  1:begin
   if CheckBox1.Checked=True then
     begin
     EditHide;
     Label2.Caption:='Check Out Lamp 1';
     end
   else
     begin
     EditShow;
     Label2.Caption:='Thank You';
     end
   end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
2:begin
 if CheckBox2.Checked=True then
   begin
   EditHide;
   Label2.Caption:='Check Out Lamp 2';
   end
 else
   begin
   EditShow;
   Label2.Caption:='Thank You';
   end
 end;
3:begin
 if CheckBox3.Checked=True then
   begin
   EditHide;
   Label2.Caption:='Check Out Lamp 3';
   end
 else
   begin
   EditShow;
   Label2.Caption:='Thank You';
   end
 end;
4:begin
 if CheckBox4.Checked=True then
   begin
   EditHide;
   Label2.Caption:='Check Out Lamp 4';
   end
 else
   begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    EditShow;
    Label2.Caption:='Thank You';
    end
  end;
5:begin
  if CheckBox5.Checked=True then
    begin
    EditHide;
    Label2.Caption:='Checked Out Lamp 5';
    end
  else
    begin
    EditShow;
    Label2.Caption:='Thank You';
    end
  end;
6:begin
  if CheckBox6.Checked=True then
    begin
    EditHide;
    Label2.Caption:='Check Out Lamp 6';
    end
  else
    begin
    EditShow;
    Label2.Caption:='Thank You';
    end
  end;
7:begin
  if CheckBox7.Checked=True then
    begin
    EditHide;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Label2.Caption:='Check Out Lamp 7';
      end
    else
      begin
      EditShow;
      Label2.Caption:='Thank You';
      end
    end;
  8:begin
    if CheckBox8.Checked=True then
      begin
      EditHide;
      Label2.Caption:='Check Out Lamp 8';
      end
    else
      begin
      EditShow;
      Label2.Caption:='Thank You';
      end
    end;
end;
end;

procedure TForm5.ComboBox1Change(Sender: TObject);
begin
Table1.RecNo:=ComboBox1.ItemIndex+1;
rowtable:=Table1.RecNo;
TestCheckBox;
GroupBox1.Enabled:=True;
Edit1.Text:=IntToStr(Table1.FieldByName('Lamp Number').AsInteger);
MaskEdit1.Text:=TimeToStr(Table1.FieldByName('Timer On').AsDateTime);
MaskEdit2.Text:=TimeToStr(Table1.FieldByName('Timer Off').AsDateTime);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MaskEdit3.Text:=DateToStr(Table1.FieldByName('Change In').AsDateTime);
MaskEdit4.Text:=DateToStr(Table1.FieldByName('Change Out').AsDateTime);
MaskEdit5.Text:=IntToStr(Table1.FieldByName('Age Of Lamp').AsInteger);
end;

procedure TForm5.Timer1Timer(Sender: TObject);
var
comparetimer:Integer;
timernow,timeron,timeroff:String;
begin
timernow:=TimeToStr(Now);
Panel3.Caption:=DateToStr(Date);
Panel4.Caption:=TimeToStr(Time);
TestCheckBox;
if CheckBox1.Checked=True Then
 begin
 timeron:=StringGrid1.Cells[1,0];
 comparetimer:=CompareStr(timernow,timeron);
 if comparetimer=0 Then
   begin
   if Form3.Shape1.Brush.Color=clNavy Then
     begin
     Sendup:='a';
     Form2.Send;
     end
   end
 end;
if CheckBox2.Checked=True Then
 begin
 timeron:=StringGrid1.Cells[1,1];
 comparetimer:=CompareStr(timernow,timeron);
 if comparetimer=0 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      begin
      if Form3.Shape2.Brush.Color=clNavy Then
        begin
        Sendup:='b';
        Form2.Send;
        end
      end
    end;
  if CheckBox3.Checked=True Then
    begin
    timeron:=StringGrid1.Cells[1,2];
    comparetimer:=CompareStr(timernow,timeron);
    if comparetimer=0 Then
      begin
      if Form3.Shape8.Brush.Color=clNavy Then
        begin
        Sendup:='c';
        Form2.Send;
        end
      end
    end;
  if CheckBox4.Checked=True Then
    begin
    timerOn:=StringGrid1.Cells[1,3];
    comparetimer:=CompareStr(timernow,timeron);
    if comparetimer=0 Then
      begin
      if Form3.Shape3.Brush.Color=clNavy Then
        begin
        Sendup:='d';
        Form2.Send;
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      end
    end;
  if CheckBox5.Checked=true Then
    begin
    timeron:=StringGrid1.Cells[1,4];
    comparetimer:=CompareStr(timernow,timeron);
    if comparetimer=0 Then
      begin
      if Form3.Shape4.Brush.Color=clNavy Then
        begin
        Sendup:='e';
        Form2.Send;
        end
      end
    end;
  if CheckBox6.Checked=True Then
    begin
    timeron:=StringGrid1.Cells[1,5];
    comparetimer:=CompareStr(timernow,timeron);
    if comparetimer=0 Then
      begin
      if Form3.Shape5.Brush.Color=clNavy Then
        begin
        Sendup:='f';
        Form2.Send;
        end
      end
    end;
  if CheckBox7.Checked=True Then
    begin
    timeron:=StringGrid1.Cells[1,6];
    comparetimer:=CompareStr(timernow,timeron);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if comparetimer=0 Then
    begin
    if Form3.Shape6.Brush.Color=clNavy Then
      begin
      Sendup:='g';
      Form2.Send;
      end
    end
  end;
if CheckBox8.Checked=True Then
  begin
  timeron:=StringGrid1.Cells[1,7];
  comparetimer:=CompareStr(timernow,timeron);
  if comparetimer=0 Then
    begin
    if Form3.Shape7.Brush.Color=clNavy Then
      begin
      Sendup:='h';
      Form2.Send;
      end
    end
  end;
if CheckBox1.Checked=True Then
  begin
  timeroff:=StringGrid1.Cells[2,0];
  comparetimer:=CompareStr(timernow,timeroff);
  if comparetimer=0 Then
    begin
    if Form3.Shape1.Brush.Color=clRed Then
      begin
      Sendup:='a';
      Form2.Send;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      end
    end
  end;
if CheckBox2.Checked=True Then
  begin
  timeroff:=StringGrid1.Cells[2,1];
  comparetimer:=CompareStr(timernow,timeroff);
  if comparetimer=0 Then
    begin
    if Form3.Shape2.Brush.Color=clRed Then
      begin
      Sendup:='b';
      Form2.Send;
      end
    end
  end;
if CheckBox3.Checked=True Then
  begin
  timeroff:=StringGrid1.Cells[2,2];
  comparetimer:=CompareStr(timernow,timeroff);
  if comparetimer=0 Then
    begin
    if Form3.Shape8.Brush.Color=clRed Then
      begin
      Sendup:='c';
      Form2.Send;
      end
    end
  end;
if CheckBox4.Checked=True Then
  begin
  timeroff:=StringGrid1.Cells[2,3];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
comparetimer:=CompareStr(timernow,timeroff);
if CompareTimer=0 Then
  begin
  if Form3.Shape3.Brush.Color=clRed Then
    begin
    Sendup:='d';
    Form2.Send;
    end
  end
end;
if CheckBox5.Checked=True Then
  begin
  timeroff:=StringGrid1.Cells[2,4];
  comparetimer:=CompareStr(timernow,timeroff);
  if comparetimer=0 Then
    begin
    if Form3.Shape4.Brush.Color=clRed Then
      begin
      Sendup:='e';
      Form2.Send;
      end
    end
  end;
if CheckBox6.Checked=True Then
  begin
  timeroff:=StringGrid1.Cells[2,5];
  comparetimer:=CompareStr(timernow,timeroff);
  if comparetimer=0 Then
    begin
    if Form3.Shape5.Brush.Color=clRed Then
      begin
      Sendup:='f';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Form2.Send;
        end
      end
    end;
  if CheckBox7.Checked=True Then
    begin
    timeroff:=StringGrid1.Cells[2,6];
    comparetimer:=CompareStr(timernow,timeroff);
    if comparetimer=0 Then
      begin
      if Form3.Shape6.Brush.Color=clRed Then
        begin
        Sendup:='g';
        Form2.Send;
        end
      end
    end;
  if CheckBox8.Checked=True Then
    begin
    timeroff:=StringGrid1.Cells[2,7];
    comparetimer:=CompareStr(timernow,timeroff);
    if comparetimer=0 Then
      begin
      if Form3.Shape7.Brush.Color=clRed Then
        begin
        Sendup:='h';
        Form2.Send;
        end
      end
    end;
  end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm5.Button1Click(Sender: TObject);
begin
Try
StrToTime(MaskEdit1.Text);
Except
 begin
  ShowMessage('โปรดตั้งเวลาเปิดใหม่');
  exit;
 end;
end;
Try
StrToTime(MaskEdit2.Text);
Except
 begin
  ShowMessage('โปรดตั้งเวลาปิดใหม่');
  exit;
 end;
end;
Try
StrToDate(MaskEdit3.Text);
Except
 begin
  ShowMessage('โปรดใส่วันที่เปลี่ยนหลอดใหม่');
  exit;
 end;
end;
Try
StrToDate(MaskEdit4.Text);
Except
 begin
 ShowMessage('โปรดกำหนดวันเปลี่ยนหลอดใหม่');
 exit;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 end;
end;
Try
StrToInt(MaskEdit5.Text);
Except
 begin
  ShowMessage('โปรดใส่อายุการใช้งานของหลอดใหม่');
  exit;
 end;
end;
CheckDate2;
if timeday3 > 0 Then
begin
 ShowMessage('กำหนดวันเปลี่ยนหลอดใหม่อีกครั้ง');
 exit;
end;
Table1.Edit;
Table1.FieldByName('Timer On').AsDateTime:=StrToTime(MaskEdit1.Text);
Table1.FieldByName('Timer Off').AsDateTime:=StrToTime(MaskEdit2.Text);
Table1.FieldByName('Change In').AsDateTime:=StrToDate(MaskEdit3.Text);
Table1.FieldByName('Change Out').AsDateTime:=StrToDate(MaskEdit4.Text);
Table1.FieldByName('Age Of Lamp').AsInteger:=StrToInt(MaskEdit5.Text);
Table1.Post;
EditHide;
CopyData;
end;

procedure TForm5.FormActivate(Sender: TObject);
begin
CopyData;
Timer1Timer(Sender);
GroupBox1.Enabled:=False;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
EditHide;
CheckDate1;
end;

procedure TForm5.CheckDate1;
var
model1,model2,model3,row,row1:Integer;
S1,S2,S3,S4,S5:String;
begin
S1:=DateToStr(Now);
if S1[2]='/' Then
  begin
   if S1[4]='/' Then
    model1:=1
   else
    model1:=2
  end
else
  begin
   if S1[3]='/' Then
    begin
    if S1[5]='/' Then
      model1:=3
    else
      model1:=4
    end
  end;
case model1 of
 1:S2:='0'+S1[1]+S1[2]+'0'+S1[3]+S1[4]+S1[5]+S1[6]+S1[7]+S1[8];
 2:S2:='0'+S1[1]+S1[2]+S1[3]+S1[4]+S1[5]+S1[6]+S1[7]+S1[8]+S1[9];
 3:S2:=S1[1]+S1[2]+S1[3]+'0'+S1[4]+S1[5]+S1[6]+S1[7]+S1[8]+S1[9];
 4:S2:=S1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
S2:=S2[7]+S2[8]+S2[9]+S2[10]+S2[6]+S2[4]+S2[5]+S2[3]+S2[1]+S2[2];
for row:=0 to 7 do
begin
S3:=StringGrid1.Cells[4,row];
if S3[2]='/' Then
  begin
  if S3[4]='/' Then
    model2:=1
  else
    model2:=2
  end
else
  begin
  if S3[3]='/' Then
    begin
    if S3[5]='/' Then
      model2:=3
    else
      model2:=4
  end
end;
case model2 of
  1:S4:='0'+S3[1]+S3[2]+'0'+S3[3]+S3[4]+S3[5]+S3[6]+S3[7]+S3[8];
  2:S4:='0'+S3[1]+S3[2]+S3[3]+S3[4]+S3[5]+S3[6]+S3[7]+S3[8]+S3[9];
  3:S4:=S3[1]+S3[2]+S3[3]+'0'+S3[4]+S3[5]+S3[6]+S3[7]+S3[8]+S3[9];
  4:S4:=S3;
end;
S4:=S4[7]+S4[8]+S4[9]+S4[10]+S4[6]+S4[4]+S4[5]+S4[3]+S4[1]+S4[2];
model3:=CompareStr(S2,S4);
if model3>=0 Then
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
row1:=row+1;
S5:=IntToStr(row1);
ShowMessage('ถึงกำหนดเปลี่ยนหลอดที่'+S5);
end;
end;
end;
procedure TForm5.CheckDate2;
var
S1,S2,S3,S4:String;
timeday1,timeday2:Integer;
begin
S1:=DateToStr(StrToDate(MaskEdit3.Text));
S3:=DateToStr(StrToDate(MaskEdit4.Text));
if S1[2]='/' Then
  begin
  if S1[4]='/' Then
    timeday1:=1
  else
    timeday1:=2
  end
else
  begin
  if S1[3]='/' Then
    begin
    if S1[5]='/'then
    timeday1:=3
  else
    timeday1:=4
    end
 end;
case timeday1 of
 1:S2:='0'+S1[1]+S1[2]+'0'+S1[3]+S1[4]+S1[5]+S1[6]+S1[7]+S1[8];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  2:S2:='0'+S1[1]+S1[2]+S1[3]+S1[4]+S1[5]+S1[6]+S1[7]+S1[8]+S1[9];
  3:S2:=S1[1]+S1[2]+S1[3]+'0'+S1[4]+S1[5]+S1[6]+S1[7]+S1[8]+S1[9];
  4:S2:=S1;
 end;
S2:=S2[7]+S2[8]+S2[9]+S2[10]+S2[6]+S2[4]+S2[5]+S2[3]+S2[1]+S2[2];
if S3[2]='/' Then
   begin
   if S3[4]='/' Then
     timeday2:=1
   else
     timeday2:=2
   end
else
   begin
   if S3[3]='/' Then
     begin
     if S3[5]='/' Then
     timeday2:=3
   else
     timeday2:=4
     end
   end;
case timeday2 of
  1:S4:='0'+S3[1]+S3[2]+'0'+S3[3]+S3[4]+S3[5]+S3[6]+S3[7]+S3[8];
  2:S4:='0'+S3[1]+S3[2]+S3[3]+S3[4]+S3[5]+S3[6]+S3[7]+S3[8]+S3[9];
  3:S4:=S3[1]+S3[2]+S3[3]+'0'+S3[4]+S3[5]+S3[6]+S3[7]+S3[8]+S3[9];
  4:S4:=S3;
end;
S4:=S4[7]+S4[8]+S4[9]+S4[10]+S4[6]+S4[4]+S4[5]+S4[3]+S4[1]+S4[2];
timeday3:=CompareStr(S2,S4);
end;
end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unit PC_HOME6;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, StdCtrls, Mask, Db, DBTables, Grids;
 type
 TForm6 = class(TForm)
  Panel1: TPanel;
  Label1: TLabel;
  ComboBox1: TComboBox;
  GroupBox1: TGroupBox;
  Label2: TLabel;
  CheckBox1: TCheckBox;
  CheckBox2: TCheckBox;
  CheckBox3: TCheckBox;
  CheckBox4: TCheckBox;
  CheckBox5: TCheckBox;
  CheckBox6: TCheckBox;
  CheckBox7: TCheckBox;
  CheckBox8: TCheckBox;
  Panel2: TPanel;
  Panel3: TPanel;
  Panel4: TPanel;
  Panel5: TPanel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Edit1: TEdit;
  MaskEdit1: TMaskEdit;
  MaskEdit2: TMaskEdit;
  Button1: TButton;
  Button2: TButton;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Timer1: TTimer;
    StringGrid1: TStringGrid;
    Table1: TTable;
    DataSource1: TDataSource;
    procedure Timer1Timer(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure CopyData;
    procedure EditHide;
    procedure EditShow;
    procedure TestCheckBox;
    procedure ComboBox1Change(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
  var
  Form6: TForm6;
  taw:Integer;
implementation
uses PC_HOME2,PC_HOME4;
{$R *.DFM}
procedure TForm6.Button2Click(Sender: TObject);
begin
close;
end;

procedure TForm6.CopyData;
var
row:Integer;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
for row:=1 to 8 do
 begin
 Table1.RecNo:=row;
   StringGrid1.Cells[0,row-1]:=
    IntToStr(Table1.FieldByName('Plug Number').AsInteger);
   StringGrid1.Cells[1,row-1]:=
    TimeToStr(Table1.FieldByName('Timer On').AsDateTime);
   StringGrid1.Cells[2,row-1]:=
    TimeToStr(Table1.FieldByName('Timer Off').AsDateTime);
 end;
Table1.RecNo:=ComboBox1.ItemIndex+1;
end;

procedure TForm6.EditHide;
begin
Edit1.Enabled:=False;
MaskEdit1.Enabled:=False;
MaskEdit2.Enabled:=False;
end;

procedure TForm6.EditShow;
begin
Edit1.Enabled:=True;
MaskEdit1.Enabled:=True;
MaskEdit2.Enabled:=True;
end;

procedure TForm6.TestCheckBox;
begin
 case taw of
 1:begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 if CheckBox1.Checked=True then
   begin
   EditHide;
   Label2.Caption:='Check Out Plug 1';
   end
  else
   begin
   EditShow;
   Label2.Caption:='Thank You';
   end
 end;
2:begin
 if CheckBox2.Checked=True then
   begin
   EditHide;
   Label2.Caption:='Check Out Plug 2';
   end
  else
   begin
   EditShow;
   Label2.Caption:='Thank You';
   end
 end;
3:begin
 if CheckBox3.Checked=True then
   begin
   EditHide;
   Label2.Caption:='Check Out Plug 3';
   end
  else
   begin
   EditShow;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Label2.Caption:='Thank You';
    end
  end;
4:begin
  if CheckBox4.Checked=True then
    begin
    EditHide;
    Label2.Caption:='Check Out Plug 4';
    end
  else
    begin
    EditShow;
    Label2.Caption:='Thank You';
    end
  end;
5:begin
  if CheckBox5.Checked=True then
    begin
    EditHide;
    Label2.Caption:='Checked Out Plug 5';
    end
  else
    begin
    EditShow;
    Label2.Caption:='Thank You';
    end
  end;
6:begin
  if CheckBox6.Checked=True then
    begin
    EditHide;
    Label2.Caption:='Check Out Plug 6';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      end
    else
      begin
      EditShow;
      Label2.Caption:='Thank You';
      end
    end;
  7:begin
    if CheckBox7.Checked=True then
      begin
      EditHide;
      Label2.Caption:='Check Out Plug 7';
      end
    else
      begin
      EditShow;
      Label2.Caption:='Thank You';
      end
    end;
  8:begin
    if CheckBox8.Checked=True then
      begin
      EditHide;
      Label2.Caption:='Check Out Plug 8';
      end
    else
      begin
      EditShow;
      Label2.Caption:='Thank You';
      end
    end;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;

procedure TForm6.ComboBox1Change(Sender: TObject);
begin
Table1.RecNo:=ComboBox1.ItemIndex+1;
taw:=Table1.RecNo;
TestCheckBox;
GroupBox1.Enabled:=True;
Edit1.Text:=IntToStr(Table1.FieldByName('Plug Number').AsInteger);
MaskEdit1.Text:=TimeToStr(Table1.FieldByName('Timer On').AsDateTime);
MaskEdit2.Text:=TimeToStr(Table1.FieldByName('Timer Off').AsDateTime);
end;

procedure TForm6.Timer1Timer(Sender: TObject);
var
comparetimer:Integer;
timernow,timeron,timeroff:String;
begin
timernow:=TimeToStr(Now);
Panel3.Caption:=DateToStr(Date);
Panel4.Caption:=TimeToStr(Time);
TestCheckBox;
if CheckBox1.Checked=True Then
  begin
  timeron:=StringGrid1.Cells[1,0];
  comparetimer:=CompareStr(timernow,timeron);
  if comparetimer=0 Then
    begin
    Form4.BitBtn4.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
    Form4.Image2.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
    Sendup:='i';
    Form2.Send;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      end
    end;
  if CheckBox2.Checked=True Then
    begin
    timeron:=StringGrid1.Cells[1,1];
    comparetimer:=CompareStr(timernow,timeron);
    if comparetimer=0 Then
      begin
       Form4.BitBtn5.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
       Form4.Image3.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
       Sendup:='j';
       Form2.Send;
      end
    end;
  if CheckBox3.Checked=True Then
    begin
    timeron:=StringGrid1.Cells[1,2];
    comparetimer:=CompareStr(timernow,timeron);
    if comparetimer=0 Then
      begin
       Form4.BitBtn6.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
       Form4.Image4.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
       Sendup:='k';
       Form2.Send;
      end
    end;
  if CheckBox4.Checked=True Then
    begin
    timeron:=StringGrid1.Cells[1,3];
    comparetimer:=CompareStr(timernow,timeron);
    if comparetimer=0 Then
      begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Form4.BitBtn7.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
        Form4.Image5.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
        Sendup:='l';
        Form2.Send;
       end
     end;
   if CheckBox5.Checked=true Then
     begin
     timeron:=StringGrid1.Cells[1,4];
     comparetimer:=CompareStr(timernow,timeron);
     if comparetimer=0 Then
       begin
        Form4.BitBtn8.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
        Form4.Image6.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
        Sendup:='m';
        Form2.Send;
       end
     end;
  if CheckBox6.Checked=True Then
     begin
     timeron:=StringGrid1.Cells[1,5];
     comparetimer:=CompareStr(timernow,timeron);
     if comparetimer=0 Then
       begin
        Form4.BitBtn9.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
        Form4.Image7.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
        Sendup:='n';
        Form2.Send;
       end
     end;
  if CheckBox7.Checked=True Then
     begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
timeron:=StringGrid1.Cells[1,6];
comparetimer:=CompareStr(timernow,timeron);
if comparetimer=0 Then
  begin
   Form4.BitBtn10.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
   Form4.Image8.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
   Sendup:='o';
   Form2.Send;
  end
end;
if CheckBox8.Checked=True Then
 begin
 timeron:=StringGrid1.Cells[1,7];
 comparetimer:=CompareStr(timernow,timeron);
 if comparetimer=0 Then
   begin
   Form4.BitBtn11.Glyph.LoadFromFile('c:\project2\picturepro\plugon1.1.bmp');
   Form4.Image9.Picture.LoadFromFile('c:\project2\picturepro\plugsi1op.bmp');
   Sendup:='p';
   Form2.Send;
  end
end;
if CheckBox1.Checked=True Then
 begin
 timeroff:=StringGrid1.Cells[2,0];
 comparetimer:=CompareStr(timernow,timeroff);
 if comparetimer=0 Then
   begin
   Form4.BitBtn4.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
   Form4.Image2.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
   Sendup:='i';
   Form2.Send;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      end
    end;
  if CheckBox2.Checked=True Then
    begin
    timeroff:=StringGrid1.Cells[2,1];
    comparetimer:=CompareStr(timernow,timeroff);
    if comparetimer=0 Then
      begin
       Form4.BitBtn5.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
       Form4.Image3.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
       Sendup:='j';
       Form2.Send;
      end
    end;
  if CheckBox3.Checked=True Then
    begin
    timeroff:=StringGrid1.Cells[2,2];
    comparetimer:=CompareStr(timernow,timeroff);
    if comparetimer=0 Then
      begin
       Form4.BitBtn6.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
       Form4.Image4.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
       Sendup:='k';
       Form2.Send;
      end
    end;
  if CheckBox4.Checked=True Then
    begin
    timeroff:=StringGrid1.Cells[2,3];
    comparetimer:=CompareStr(timernow,timeroff);
    if comparetimer=0 Then
      begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Form4.BitBtn7.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
      Form4.Image5.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
      Sendup:='l';
      Form2.Send;
     end
    end;
  if CheckBox5.Checked=True Then
    begin
    timeroff:=StringGrid1.Cells[2,4];
    comparetimer:=CompareStr(timernow,timeroff);
    if comparetimer=0 Then
      begin
      Form4.BitBtn8.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
      Form4.Image6.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
      Sendup:='m';
      Form2.Send;
      end
    end;
  if CheckBox6.Checked=True Then
    begin
    timeroff:=StringGrid1.Cells[2,5];
    comparetimer:=CompareStr(timernow,timeroff);
    if comparetimer=0 Then
      begin
      Form4.BitBtn9.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
      Form4.Image7.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
      Sendup:='n';
      Form2.Send;
     end
    end;
  if CheckBox7.Checked=True Then
    begin
```

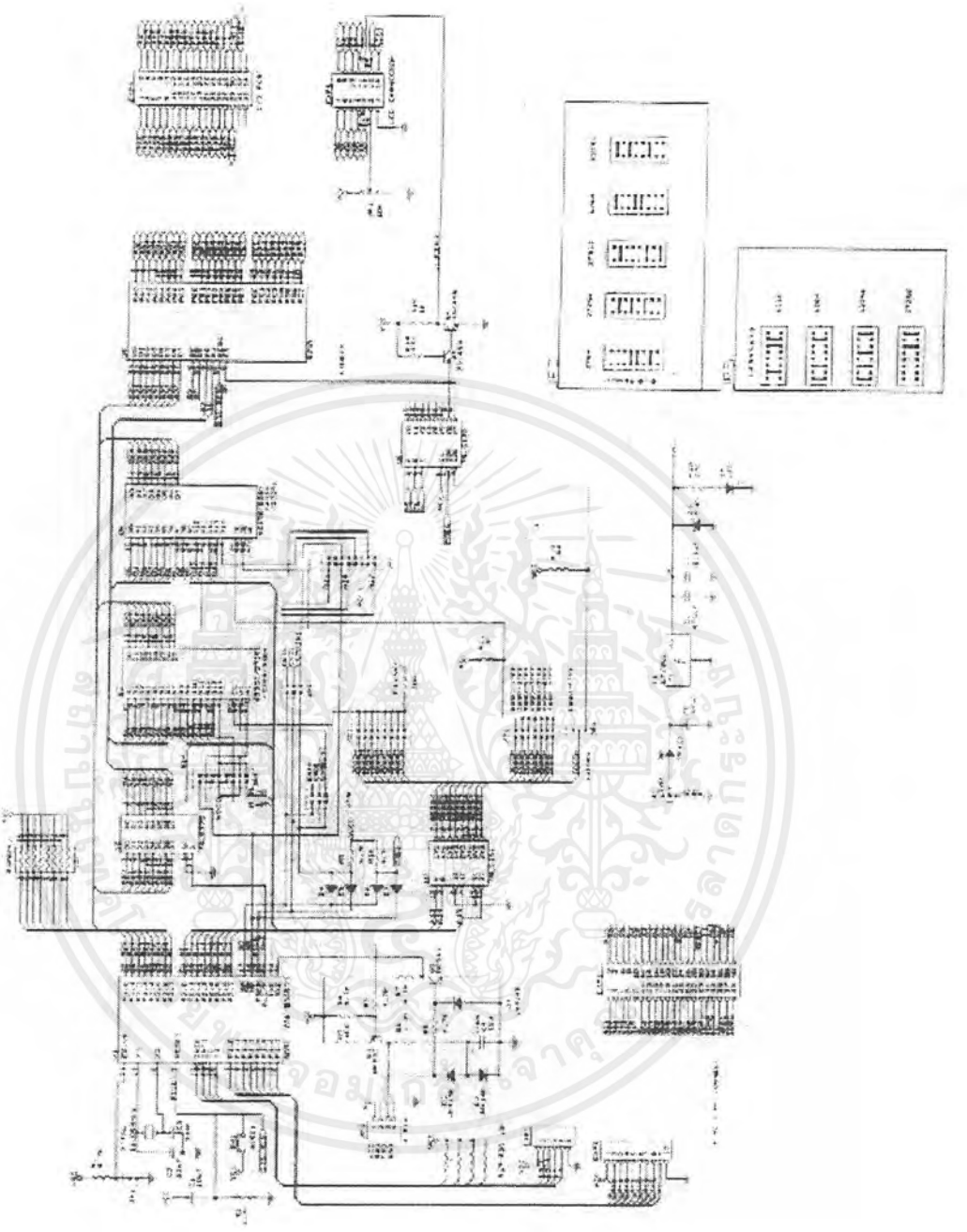
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  timeroff:=StringGrid1.Cells[2,6];
  comparetimer:=CompareStr(timernow,timeroff);
  if comparetimer=0 Then
    begin
     Form4.BitBtn10.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
     Form4.Image8.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
      Sendup:='o';
      Form2.Send;
    end
  end;
if CheckBox8.Checked=True Then
  begin
  timeroff:=StringGrid1.Cells[2,7];
  comparetimer:=CompareStr(timernow,timeroff);
  if comparetimer=0 Then
    begin
     Form4.BitBtn11.Glyph.LoadFromFile('c:\project2\picturepro\plugclo1.bmp');
     Form4.Image9.Picture.LoadFromFile('c:\project2\picturepro\plugsi1.bmp');
      Sendup:='p';
      Form2.Send;
    end
  end;
end;

procedure TForm6.Button1Click(Sender: TObject);
begin
Try
StrToTime(MaskEdit1.Text);
Except
 begin
  ShowMessage('โปรดตั้งเวลาเปิดใหม่');
  exit;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

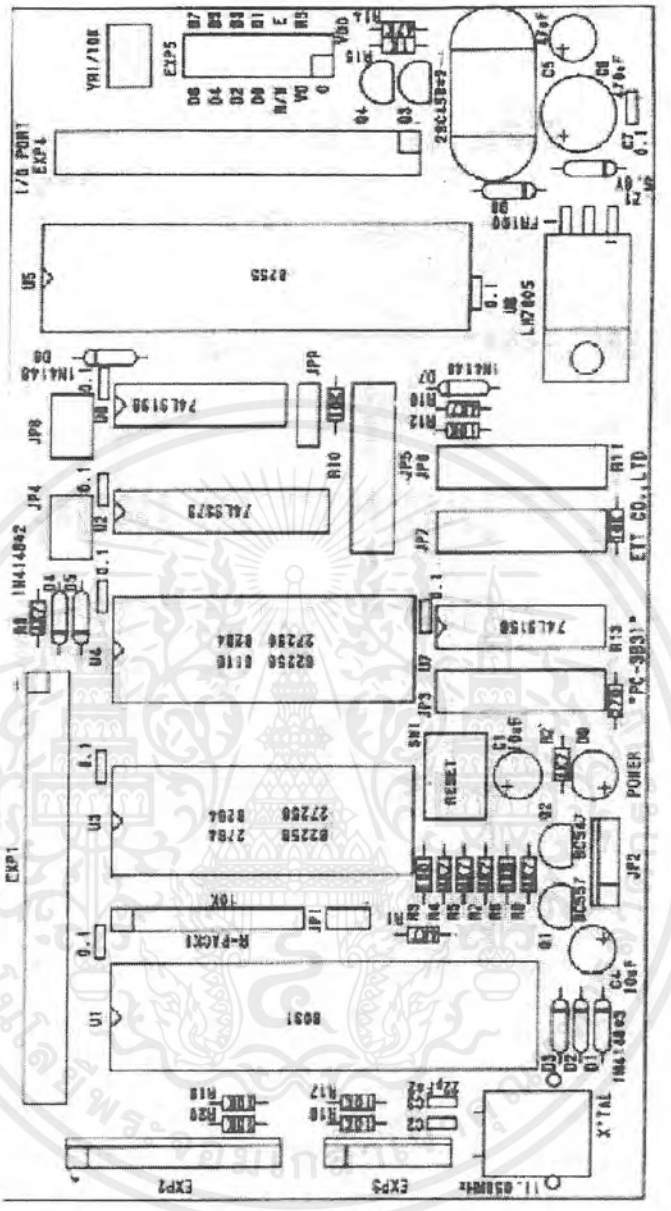
```
 end;
end;
Try
StrToTime(MaskEdit2.Text);
Except
 begin
  ShowMessage('โปรดตั้งเวลาปิดใหม่');
  exit;
 end;
end;
Table1.Edit;
Table1.FieldByName('Timer On').AsDateTime:=StrToTime(MaskEdit1.Text);
Table1.FieldByName('Timer Off').AsDateTime:=StrToTime(MaskEdit2.Text);
Table1.Post;
EditHide;
CopyData;
end;

procedure TForm6.FormActivate(Sender: TObject);
begin
CopyData;
Timer1Timer(Sender);
GroupBox1.Enabled:=False;
EditHide;
end;
end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



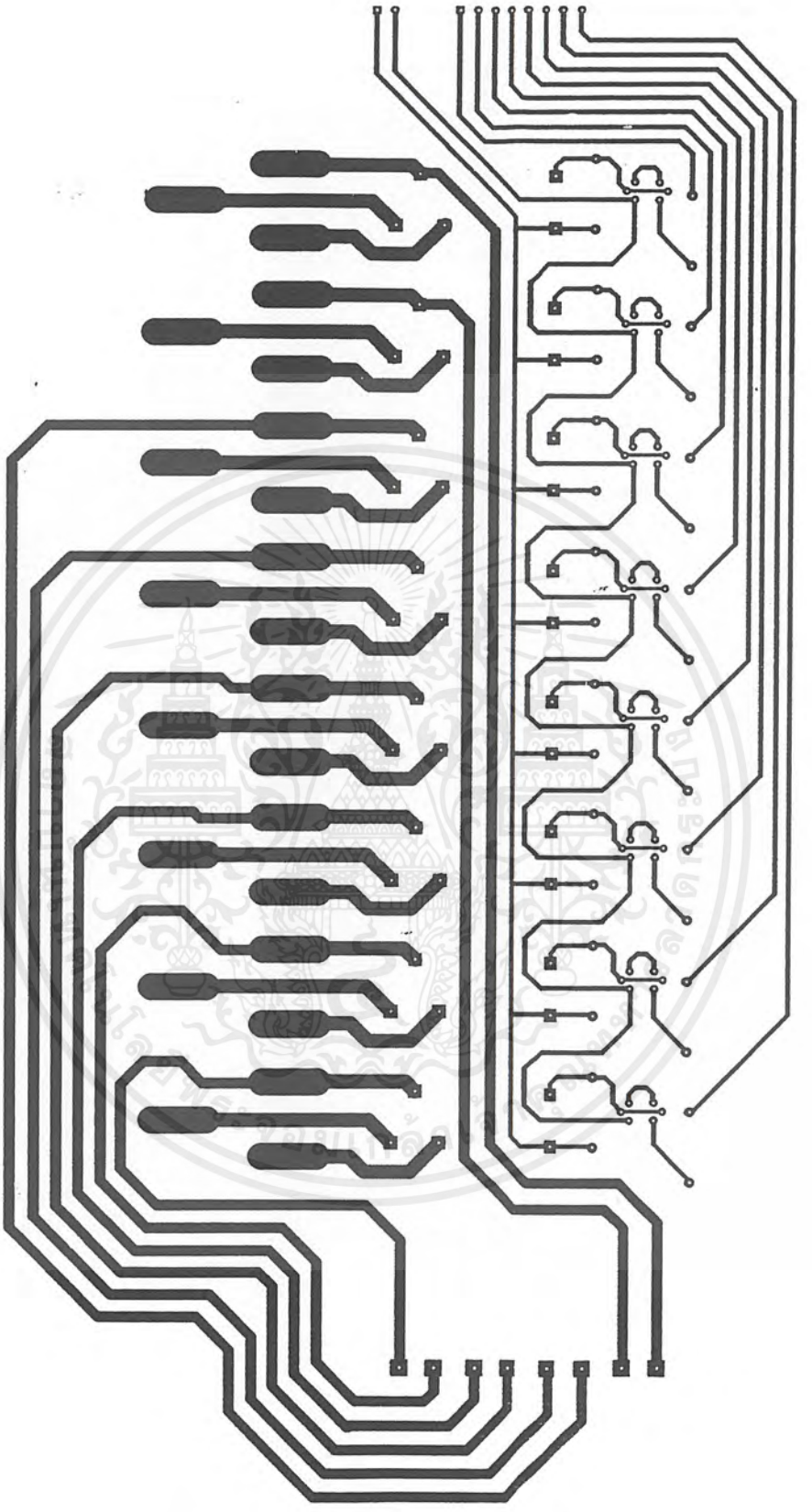
รูปแสดงวงจรของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



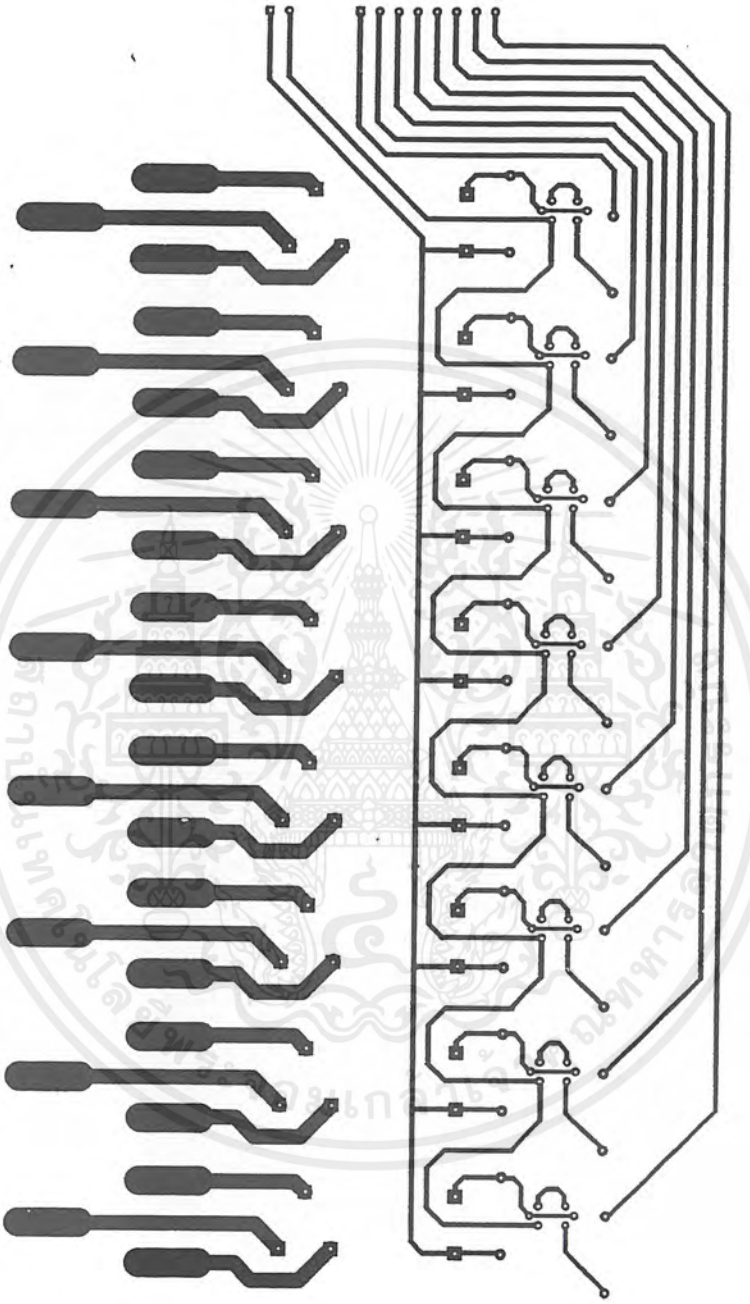
แสดงตำแหน่งอุปกรณ์ต่างๆ บนบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



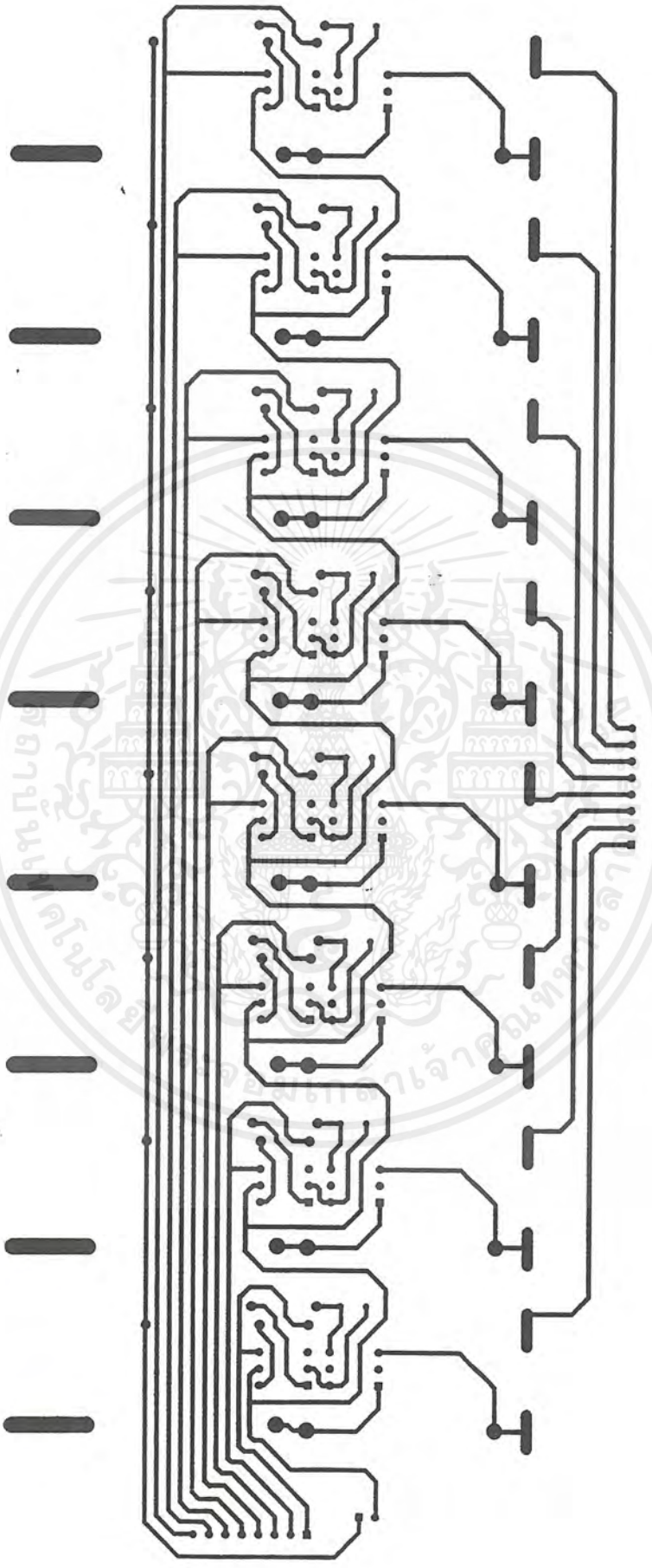
รูปแสดงลายวงจรพิมพ์ของบอดร์รีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงลายวงจรพิมพ์ของบอร์ดรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA

6-Pin DIP Optoisolators
Logic Output

H11L1
H11L2

6-PIN DIP OPTOISOLATORS LOGIC OUTPUT

CASE 730A-03
PLASTIC

. . . gallium arsenide IRED optically coupled to a high-speed integrated detector with Schmitt trigger output. Designed for applications requiring electrical isolation, fast response time, noise immunity and digital logic compatibility such as interfacing computer terminals to peripheral equipment, digital control of power supplies, motors and other servo machine applications.

- High Isolation Voltage — V_{ISO} = 7500 Vac pk Min
- Guaranteed Switching Times — t_{on}, t_{off} < 4 µs
- Built-In ON/OFF Threshold Hysteresis
- Economical, Standard Dual-In-Line Plastic Package
- UL Recognized, File No. E54915

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
INPUT LED			
Reverse Voltage	V_R	6	Volts
Forward Current — Continuous — Peak Pulse Width = 300 µs, 2% Duty Cycle	I_F	60 1.2	mA Amp
LED Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	120 1.41	mW mW/°C
OUTPUT DETECTOR			
Output Voltage Range	V_O	0-16	Volts
Supply Voltage Range	V_{CC}	3-16	Volts
Output Current	I_O	50	mA
Detector Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	150 1.76	mW mW/°C
TOTAL DEVICE			
Total Device Dissipation @ T_A = 25°C Derate above 25°C	P_D	250 2.94	mW mW/°C
Maximum Operating Temperature	T_A	−40 to +85	°C
Storage Temperature Range	T_{stg}	−55 to +150	°C
Soldering Temperature (10 s)		260	°C
Isolation Surge Voltage (Peak ac Voltage, 60 Hz, 1 Second Duration) (1)	V_{ISO}	7500	Volts

(1) Isolation surge voltage is an internal device dielectric breakdown rating.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

H11L1, H11L2

ELECTRICAL CHARACTERISTICS (T_A = 0 to 70°C)

Characteristic		Symbol	Min	Typ	Max	Unit
INPUT LED						
Reverse Leakage Current (V_R = 3 V, R_L = 1 MΩ)		I_R	—	0.05	10	μA
Forward Voltage (I_F = 10 mA) (I_F = 0.3 mA)		V_F	— 0.75	1.2 0.95	1.5 —	Volts
Capacitance (V_R = 0 V, f = 1 MHz)		C	—	18	—	pF
OUTPUT DETECTOR						
Operating Voltage		V_{CC}	3	—	15	Volts
Supply Current (I_F = 0, V_{CC} = 5 V)		$I_{CC(off)}$	—	1	5	mA
Output Current, High (I_F = 0, V_{CC} = V_o = 15 V)		I_{OH}	—	—	100	μA
COUPLED						
Supply Current (I_F = $I_{F(on)}$, V_{CC} = 5 V)		$I_{CC(on)}$	—	1.6	5	mA
Output Voltage, Low (R_L = 270 Ω, V_{CC} = 5 V, I_F = $I_{F(on)}$)		V_{OL}	—	0.2	0.4	Volts
Threshold Current, ON (R_L = 270 Ω, V_{CC} = 5 V)	H11L1 H11L2	$I_{F(on)}$	— —	1 —	1.6 10	mA
Threshold Current, OFF (R_L = 270 Ω, V_{CC} = 5 V)	H11L1 H11L2	$I_{H(off)}$	0.3 0.3	0.75 —	— —	mA
Hysteresis Ratio (R_L = 270 Ω, V_{CC} = 5 V)		$\frac{I_{F(off)}}{I_{F(on)}}$	0.5	0.75	0.9	
Isolation Voltage (1) 60 Hz, AC Peak, 1 second, T_A = 25°C		V_{ISO}	7500	—	—	Vac(pk)
Turn On Time	R_L = 270 Ω V_{CC} = 5 V, I_F = $I_{F(on)}$ T_A = 25°C	t_{on}	—	1.2	4	μs
Fall Time		t_f	—	0.1	—	
Turn-Off Time		t_{off}	—	1.2	4	
Rise Time		t_r	—	0.1	—	

(1) For this test IRED Pins 1 and 2 are common and Output Gate Pins 4, 5, 6 are common.

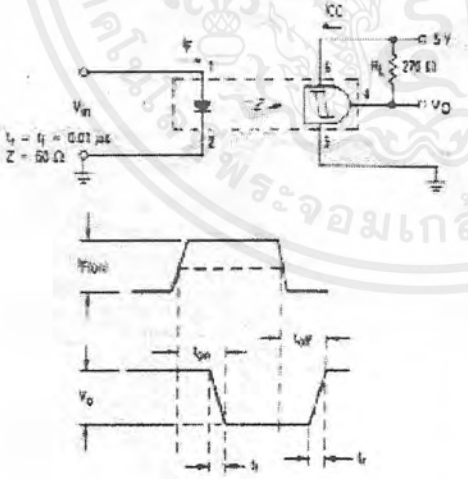
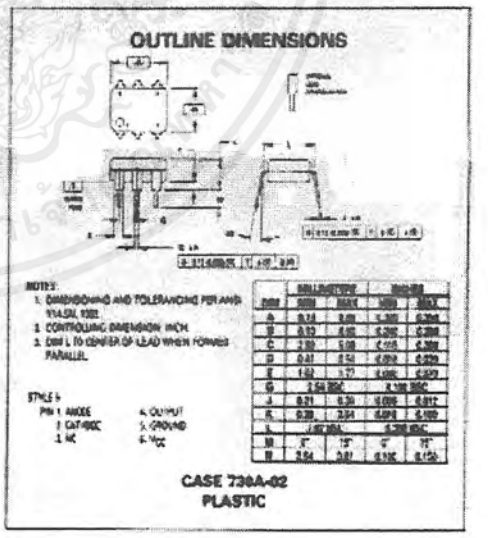


Figure 1. Switching Test Circuit



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

H11L1, H11L2

TYPICAL CHARACTERISTICS

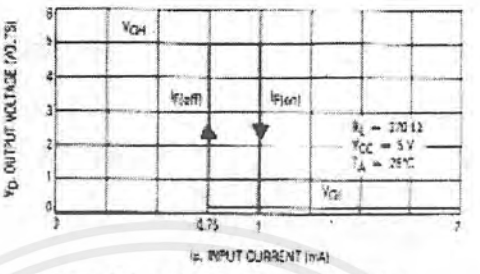


Figure 2. Transfer Characteristics for H11L1

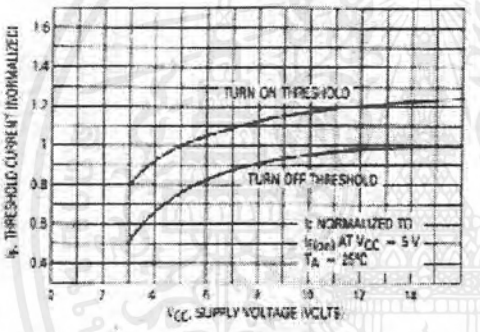


Figure 3. Threshold Current versus Supply Voltage

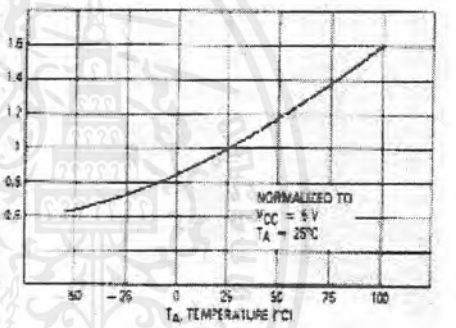


Figure 4. Threshold Current versus Temperature

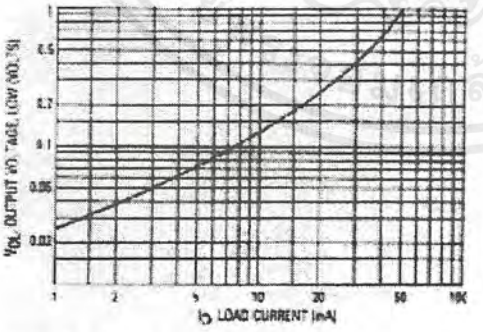


Figure 5. Output Voltage, Low versus Load Current

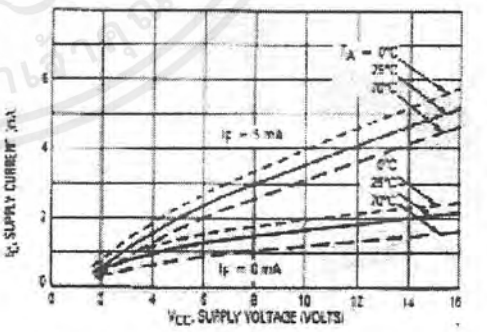


Figure 6. Supply Current versus Supply Voltage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CMOS single-chip 8-bit microcontroller

80C31/80C51/87C51

DESCRIPTION

The Philips 80C31/80C51/87C51 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The CMOS 8XC51 is functionally compatible with the NMOS 8031/8051 microcontrollers. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Philips epitaxial substrate minimizes latch-up sensitivity.

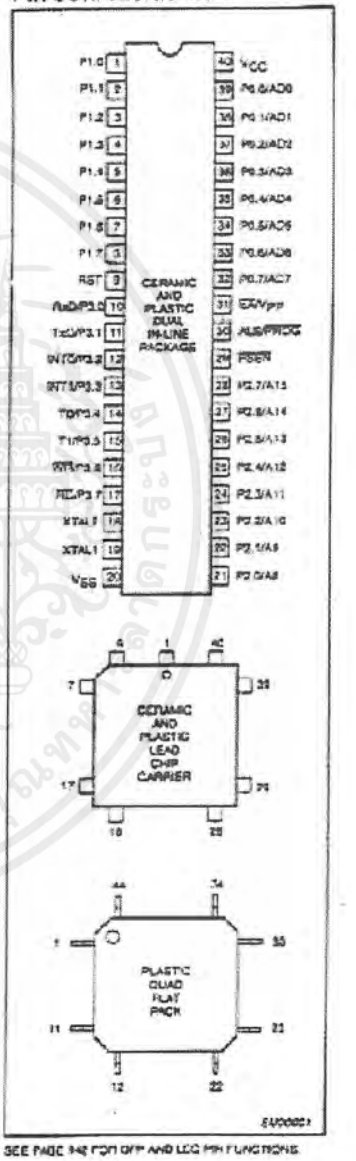
The 8XC51 contains a 4k x 8 ROM (80C51) EPROM (87C51), a 128 x 8 RAM, 32 I/O lines, two 16-bit counter/timers, a five-source, two-priority level nested interrupt structure, a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART, and on-chip oscillator and clock circuits.

In addition, the device has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

FEATURES

- 8031/8051 compatible
  - 4k x 8 ROM (80C51)
  - 4k x 8 EPROM (87C51)
  - ROMless (80C31)
  - 128 x 8 RAM
  - Two 16 bit counter/timers
  - Full duplex serial channel
  - Boolean processor
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- Five speed ranges at V_{CC} = 5V
  - 12MHz
  - 16MHz
  - 24MHz
  - 30MHz
  - 33MHz
- Five package styles
- Extended temperature ranges
- OTP package available

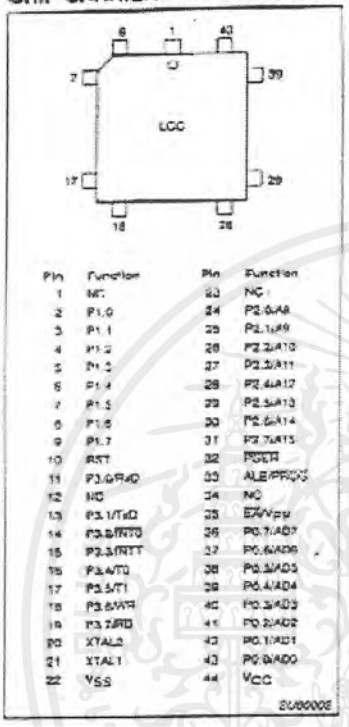
PIN CONFIGURATIONS



SEE PAGE 142 FOR OTP AND LCC PIN FUNCTIONS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

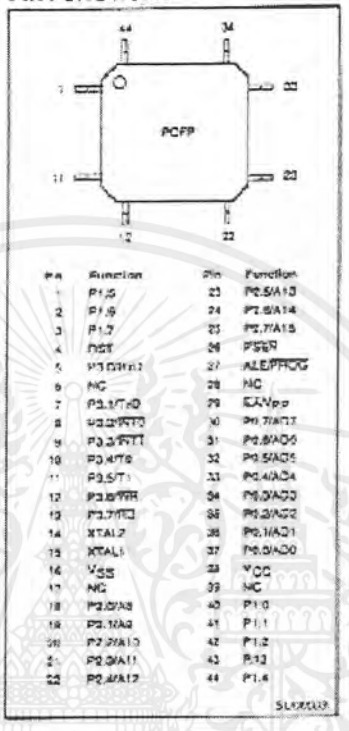
CERAMIC AND PLASTIC LEADED CHIP CARRIER PIN FUNCTIONS



Pin	Function	Pin	Function
1	NC	23	NC
2	P1.0	24	P2.6/A14
3	P1.1	25	P2.7/A15
4	P1.2	26	PSEN
5	P1.3	27	ALE/PROG
6	P1.4	28	NC
7	P1.5	29	P2.5/A13
8	P1.6	30	P2.6/A14
9	P1.7	31	P2.7/A15
10	RST	32	PSEN
11	P3.0/RxD	33	ALE/PROG
12	NC	34	NC
13	P3.1/TxD	35	EA/Vpp
14	P3.2/INT0	36	P0.7/AD7
15	P3.3/INT1	37	P0.6/AD6
16	P3.4/T0	38	P0.5/AD5
17	P3.5/T1	39	P0.4/AD4
18	P3.6/WR	40	P0.3/AD3
19	P3.7/RD	41	P0.2/AD2
20	XTAL2	42	P0.1/AD1
21	XTAL1	43	P0.0/AD0
22	V_{SS}	44	V_{CC}

SU00008

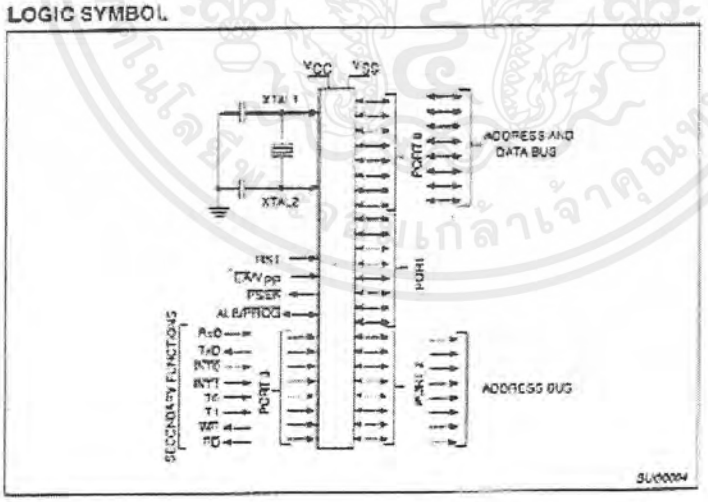
PLASTIC QUAD FLAT PACK PIN FUNCTIONS



Pin	Function	Pin	Function
1	P1.5	23	P2.5/A13
2	P1.6	24	P2.6/A14
3	P1.7	25	P2.7/A15
4	RST	26	PSEN
5	P3.0/RxD	27	ALE/PROG
6	NC	28	NC
7	P3.1/TxD	29	EA/Vpp
8	P3.2/INT0	30	P0.7/AD7
9	P3.3/INT1	31	P0.6/AD6
10	P3.4/T0	32	P0.5/AD5
11	P3.5/T1	33	P0.4/AD4
12	P3.6/WR	34	P0.3/AD3
13	P3.7/RD	35	P0.2/AD2
14	XTAL2	36	P0.1/AD1
15	XTAL1	37	P0.0/AD0
16	V_{SS}	38	V_{CC}
17	NC	39	NC
18	P2.0/A8	40	P1.0
19	P2.1/A9	41	P1.1
20	P2.2/A10	42	P1.2
21	P2.3/A11	43	P1.3
22	P2.4/A12	44	P1.4

SU00009

LOGIC SYMBOL



SU00004

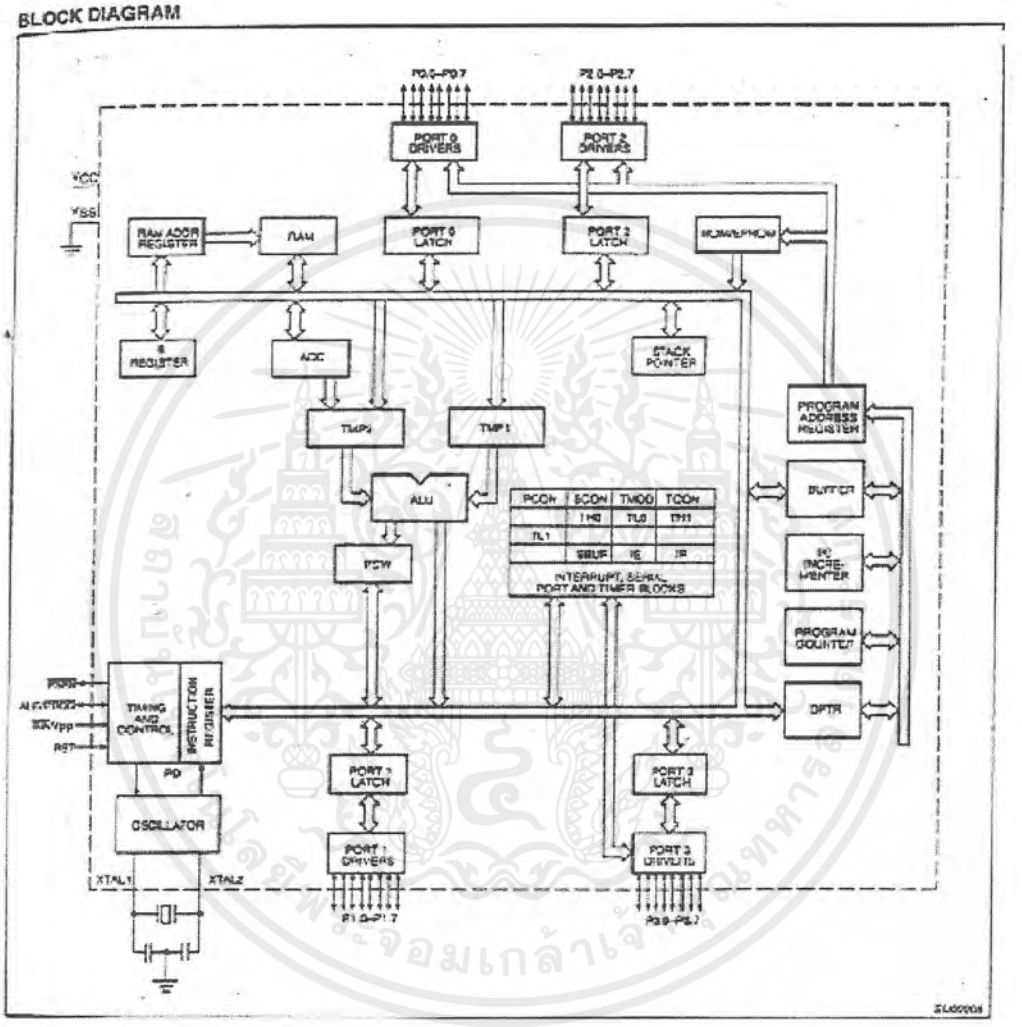
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIN DESCRIPTION

MNEMONIC	PIN NO. DIP	LCC	QFP	TYPE	NAME AND FUNCTION
V_{SS}	20	22	16	I	Ground: 0V reference.
V_{CC}	40	44	38	I	Power Supply: This is the power supply voltage for normal, idle, and power-down operation.
P0.0–0.7	39–32	43–36	37–30	I/O	Port 0: Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the 87C51. External pull-ups are required during program verification.
P1.0–P1.7	1–8	2–9	40–44, 1–3	I/O	Port 1: Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 1 also receives the low-order address byte during program memory verification.
P2.0–P2.7	21–28	24–31	18–25	I/O	Port 2: Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register.
P3.0–P3.7	10–17	11, 13–19	5, 7–13	I/O	Port 3: Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 3 also serves the special features of the 80C51 family, as listed below:
	10	11	5	I	RxD (P3.0): Serial input port
	11	13	7	O	TxD (P3.1): Serial output port
	12	14	8	I	INT0 (P3.2): External interrupt
	13	15	9	I	INT1 (P3.3): External interrupt
	14	16	10	I	T0 (P3.4): Timer 0 external input
	15	17	11	I	T1 (P3.5): Timer 1 external input
	16	18	12	O	WR (P3.6): External data memory write strobe
	17	19	13	O	RD (P3.7): External data memory read strobe
RST	9	10	4	I	Reset: A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to V_{SS} permits a power-on reset using only an external capacitor to V_{CC}.
ALE/PROG	30	33	27	I/O	Address Latch Enable/Program Pulse: Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input (PROG) during EPROM programming.
PSEN	29	32	26	O	Program Store Enable: The read strobe to external program memory. When the device is executing code from the external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory. PSEN is not activated during fetches from internal program memory.
EA/V_{PP}	31	35	29	I	External Access Enable/Programming Supply Voltage: EA must be externally held low to enable the device to fetch code from external program memory locations 0000H to 0FFFFH. If EA is held high, the device executes from internal program memory unless the program counter contains an address greater than 0FFFFH. This pin also receives the 12.75V programming supply voltage (V_{PP}) during EPROM programming.
XTAL1	19	21	15	I	Crystal 1: Input to the inverting oscillator amplifier and input to the internal clock generator circuits.
XTAL2	18	20	14	O	Crystal 2: Output from the inverting oscillator amplifier.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BLOCK DIAGRAM



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

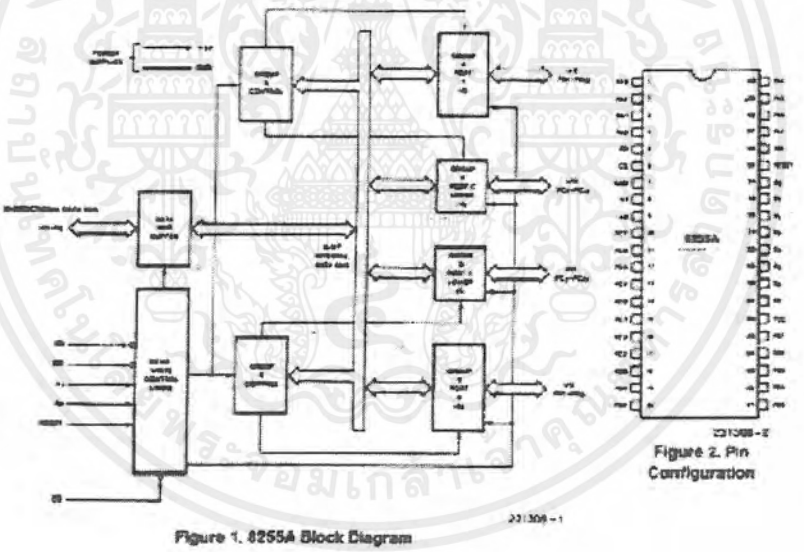
intel®

8255A/8255A-5
PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- Reduces System Package Count
- Improved DC Driving Capability
- Available in EXPRESS
  - Standard Temperature Range
  - Extended Temperature Range
- 40 Pin DIP Package or 44 Lead PLCC

(See Intel Packaging Outline Number 231369)

The Intel 8255A is a general purpose programmable I/O device designed for use with Intel microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.



231308-1

Figure 1. 8255A Block Diagram

231308-2

Figure 2. Pin Configuration

September 1987
Order Number: 231305-002

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8255A FUNCTIONAL DESCRIPTION

General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

($\overline{CS}$)

Chip Select. A "low" on this input pin enables the communication between the 8255A and the CPU.

($\overline{RD}$)

Read. A "low" on this input pin enables the 8255A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

($\overline{WR}$)

Write. A "low" on this input pin enables the CPU to write data or control words into the 8255A.

(A_0 and A_1)

Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus (A_0 and A_1).

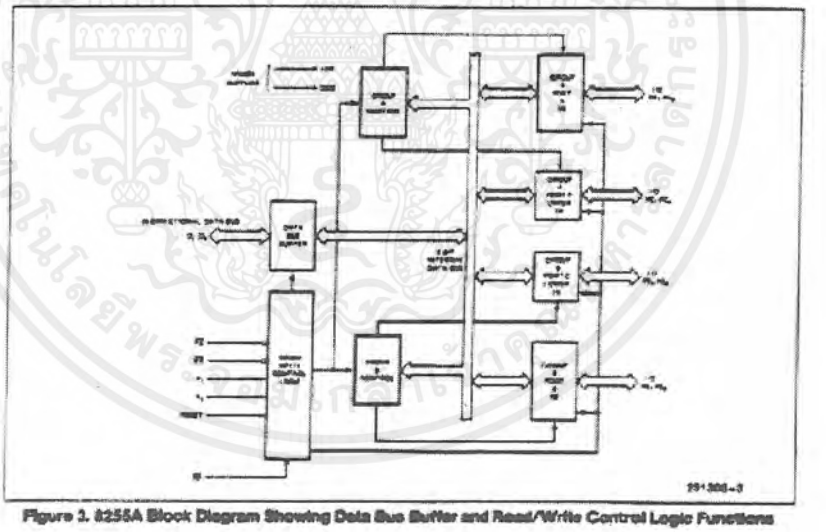


Figure 2. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8255A BASIC OPERATION

A_1	A_0	RD	WR	CS	Input Operation (READ)
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
					Output Operation (WRITE)
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	0	1	0	0	Data Bus → Port C
1	1	1	0	0	Data Bus → Control
					Disable Function
X	X	X	X	1	Data Bus → 3-State
1	1	0	1	0	Illegal Condition
X	X	1	1	0	Data Bus → 3-State

(RESET)

Reset. A "high" on this input clears the control register and all ports (A, B, C) are set to the input mode.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 8255A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 8255A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A—Port A and Port C upper ($C7-C4$)

Control Group B—Port B and Port C lower ($C3-C0$)

The Control Word Register can only be written into. No Read operation of the Control Word Register is allowed.

Ports A, B, and C

The 8255A contains three 8-bit ports (A, B, and C). All can be configured in a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 8255A.

Port A. One 8-bit data output latch/buffer and one 8-bit data input latch.

Port B. One 8-bit data input/output latch/buffer and one 8-bit data input buffer.

Port C. One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

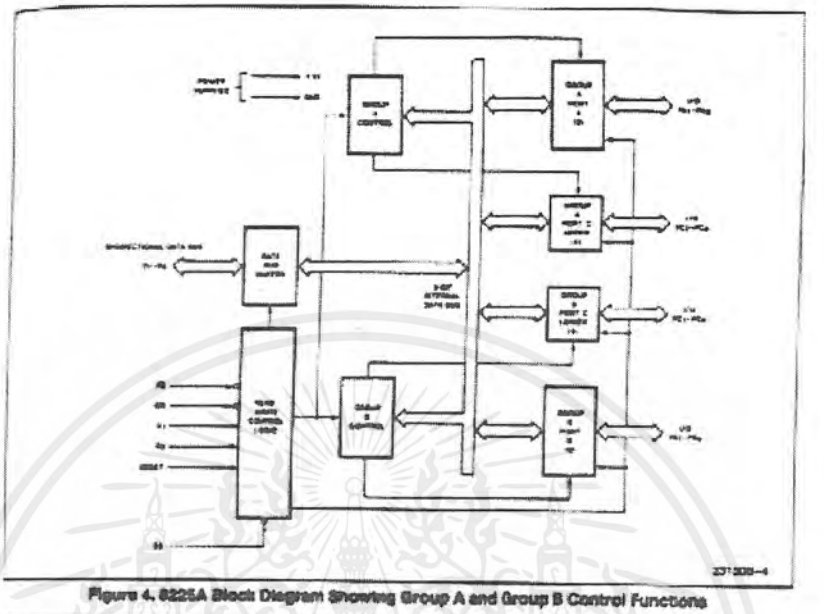
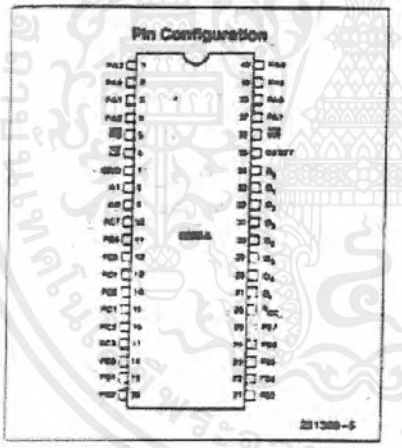


Figure 4. 8225A Block Diagram Showing Group A and Group B Control Functions



Pin Names	
D_7–D_0	Data Bus (Bi-Directional)
RESET	Reset Input
$\overline{CS}$	Chip Select
$\overline{RD}$	Read Input
$\overline{WR}$	Write Input
A0, A1	Port Address
PA7–PA0	Port A (BIT)
PB7–PB0	Port B (BIT)
PC7–PC0	Port C (BIT)
V_{CC}	+5 Volts
GND	0 Volts

8255A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mode 0—Basic Input/Output

Mode 1—Strobed Input/Output

Mode 2—Bi-Directional Bus

When the reset input goes "high" all ports will be set to the input mode (i.e., all 24 lines will be in the high impedance state). After the reset is removed the 8255A can remain in the input mode with no additional initialization required. During the execution of the system program any of the other modes may be selected using a single output instruction. This allows a single 8255A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

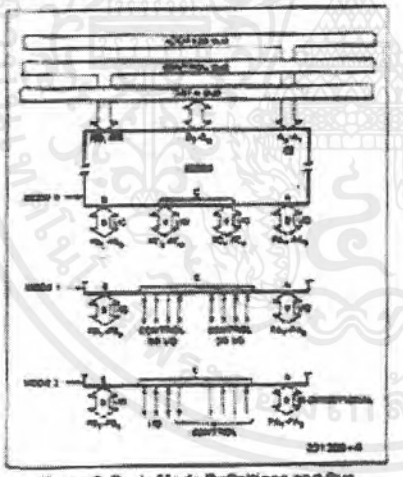


Figure 5. Basic Mode Definitions and Bus Interface

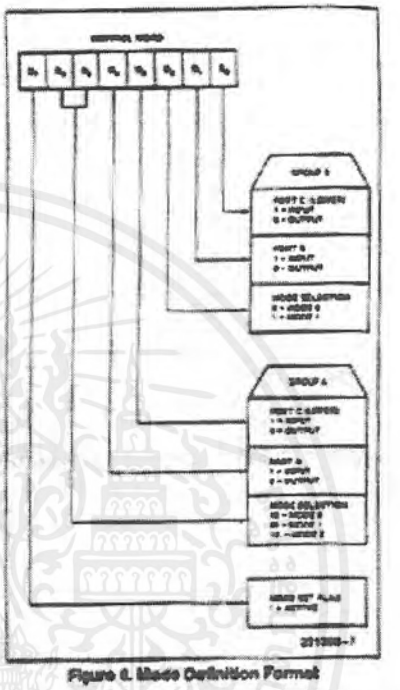


Figure 6. Mode Definition Format

The mode definitions and possible mode combinations may seem confusing at first but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 8255A has taken into account things such as efficient PC board layout, control signal definition vs PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set or Reset using a single OUTput instruction. This feature reduces software requirements in Control-based applications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณ ผศ.วิริยะ กองรัตน์ และอาจารย์ อาจินต์ น่วมสำราญ ซึ่งเป็นคณะอาจารย์ที่ปรึกษาเป็นอย่างสูง ที่ให้ความรู้ ความคิดริเริ่มและชี้แนะแนวทางต่างๆ ตลอดจนให้คำปรึกษาแนวทางการค้นคว้า การจัดทำปริญญานิพนธ์ฉบับนี้จนสำเร็จลุลวงไปได้ด้วยดี

รวมทั้งเพื่อนๆนักศึกษาทุกๆคนที่ให้คำแนะนำที่เป็นประโยชน์จนเป็นผลให้โครงงานนี้สำเร็จไปได้ด้วยดี

ขอขอบพระคุณมา ณ ที่นี้ด้วย

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1 สมยศ จุณณะปิยะ, "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ตระกูลMCS-51 ", สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,กรุงเทพฯ,154หน้า,2541

2 สุเจตน์ จันทรังษี, "ไมโครคอลโทรลเลอร์ชิพเดี่ยว 8051", โครงการตำราวิชาการวิทยาลัยมหานคร,178หน้า,2535

3 จักรพงษ์ สุขประเสริฐ กนกพร ภาวศุทธิกุลและสัจจะ จรัสรุ่งรวีวร, "คู่มือการเขียนโปรแกรมด้วยเดลไฟล์4.0" ,สำนักพิมพ์อินโฟเพรส,334หน้า, 2542

4 ทวีชัย ภูรีทิพย์, "ไขปัญหา RS-232-C", บริษัท ซีเอ็ด ยูเคชั่น จำกัด,432หน้า,2538

5 ปิยะ อำนวยพร, "เรียนลัดจากตัวอย่างเดลไฟล์ 3", บริษัท เฟิสท์ แปซิฟิก มีเดีย (ไทยแลนด์) จำกัด,285หน้า, 2541

6 กนก กุสุมาลย์นุกูล และ ไกรวุฒิ มั่นเสถียรสิน, " DELPHI 4 " ,บริษัท ซัคเซส มีเดีย จำกัด,324หน้า,2541

7 กณพ แก้วพิชัย และ ธนัท ชัยยุทธ, " การเขียนโปรแกรมภาษาปาสคาลเบื้องต้น", บริษัท ซีเอ็ดยูเคชั่น จำกัด,477หน้า,2538

8 Scoff L. warner and Paul Goldsman, "Delphi 2.0 By Example", Secon Edition, Que, 1996

9 Nathan and Ori Gurewich, "Teach yourself Database Programming With Delphi", Sams Publsing, 1995

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้