สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายโทรศัพท์ควบคุมด้วยพีซี และอาร์ซียู

Electrical Equipments Controlled by PC and RCU via Telephone Line

T042285

โดย

นายสิทธิชัย ช่อสลิด

นายอรุณ สุขสำราญ

เลขหมู่.........
เลขทะเบียน 42285
วัน, เดือน, ปี 16 พ.ค. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายโทรศัพท์ควบคุมด้วยพีซี และอาร์ซียู

Electrical Equipments Controlled by PC and RCU via Telephone Line

โดย

นายสิทธิชัย ช่อสลิด

นายอรุณ สุขสำราญ

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์สุชาติ คูณทวีเทพ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา2543

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายโทรศัพท์ควบคุมด้วยพีซี และอาร์ซียู

(Electrical Equipments Controlled by PC and RCU via Telephone Line)

ผู้จัดทำ

1. นายสิทธิชัย ช่อสลิด รหัส 41013238

2. นายอรุณ สุขสำราญ รหัส 41013240

ปริญญานิพนธ์ฉบับนี้ได้ผ่านการตรวจสอบ โดยอาจารย์ที่ปรึกษาแล้ว

.... อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์สุชาติ คูณทวีเทพ)

...19.../...3.../...44...

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายโทรศัพท์ควบคุมด้วยพีซีและอาร์ซียู

นายสิทธิชัย ช่อสลิด
นายอรุณ สุขสำราญ
ผศ.สุชาติ คูณทวีเทพ (อาจารย์ที่ปรึกษา)
ปีการศึกษา2543

บทคัดย่อ

โครงงาน "ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายโทรศัพท์ควบคุมด้วยพีซี และอาร์ซียู" เป็นการควบคุมอุปกรณ์ไฟฟ้าทั่วๆ ไปด้วยคอมพิวเตอร์ในการควบคุมนั้นผู้ใช้สามารถสั่งให้อุปกรณ์ทำงานหรือไม่ทำงานได้โดยโปรแกรมบนคอมพิวเตอร์ พร้อมกับแสดงผลสถานะ การทำงานในรูปกราฟฟิกบนหน้าจอคอมพิวเตอร์ คำสั่งที่ถูกสั่งงานจากโปรแกรมนั้น จะส่งไปให้ไมโครคอนโทรลเลอร์ MCS-51 ตัดสินใจควบคุมอุปกรณ์ไฟฟ้าตามที่สั่งมา และตรวจสอบสภาวะการทำงานของอุปกรณ์ไฟฟ้าแล้วส่งไปแสดงผลที่คอมพิวเตอร์ การเชื่อมต่อระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์ใช้มาตรฐานของ RS-232

นอกจากจะสามารถควบคุมอุปกรณ์ไฟฟ้าจากคอมพิวเตอร์แล้วยังสามารถควบคุมจาก RCU (Remote Control Unit) ได้อีกด้วย โดยใช้หลักการความถี่คู่ของระบบโทรศัพท์(DTMF) โดยมีการเข้ารหัสและการถอดรหัสแล้วแปลงเป็นสัญญาณดิจิตอลส่งให้ไมโครคอนโทรลเลอร์

นอกจากนั้นยังตรวจสอบสถานะ การทำงานของอุปกรณ์ไฟฟ้า ทำได้ทุกครั้งที่มีการสั่งมาโดยใช้หลักการเหนี่ยวนำกระแสไฟฟ้า เมื่อมีกระแสไฟฟ้าไหลเข้าสู่อุปกรณ์ไฟฟ้าก็จะมีการเหนี่ยวนำกระแสไปยังชุดควบคุมสถานะการทำงานแล้วส่งกลับไปยังไมโครคอนโทรลเลอร์แล้วส่งต่อไปแสดงผลที่คอมพิวเตอร์หรือหลอด LED ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Equipment Controlled by PC and RCU via Telephone Line

Mr. Sittichai Chosalid

Mr. Arun Suksomran

Assist. Prof. Suchat Khuntaweetep

Educational year 2000

Abstract

The project is Electrical Equipment Controlled by PC and RCU via Telephone Line. It is general electrical equipment load controlled by computer. Load is controlled to turn on or turn off by program and load status is presented on computer's monitor graphic. The computer program is developed with Microsoft Visual Basic 6 language. A personal computer is connected to MCS-51 microcontroller with computer's series port RS-232 protocol. The microcontroller send controlling signal to load that selected from program, turn on or turn off. It can also be a RCU (Remote control unit) that using DTMF (Dual Tone Multi-Frequency) to be other controller unit. The microcontroller controls load via telephone line.

For this system we can check load status by using current inducement from current flow to load and can display the results by computer and LED.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ได้สำเร็จลุล่วงไปด้วยดีเป็นผลมาจากความช่วยเหลืออย่างดียิ่งของผู้ช่วยศาสตราจารย์สุชาติ คูณทวีเทพ อาจารย์ที่ปรึกษาปริญญานิพนธ์ และอาจารย์ประจำภาควิชาอิเล็กทรอนิกส์ทุกท่าน ที่ได้ให้คำแนะนำและข้อคิดเห็นต่าง ๆ ของปริญญานิพนธ์ คณะผู้จัดทำปริญญานิพนธ์นี้ต้องขอขอบพระคุณเป็นอย่างสูงมา ณ โอกาสนี้

นอกจากนี้ คณะผู้จัดทำขอกราบขอบพระคุณบิดามารดา และผู้มีพระคุณที่ให้การสนับสนุนทางด้านการเงิน และให้กำลังใจแก่ผู้จัดทำเสมอมาจนสำเร็จการศึกษา

สิทธิชัย ช่อสลิด ผู้จัดทำ
(นายสิทธิชัย ช่อสลิด)

อรุณ สุขสำราญ ผู้จัดทำ
(นายอรุณ สุขสำราญ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ........ ก

Abstract........ ข

กิตติกรรมประกาศ........ ค

สารบัญรูป........ จ

สารบัญตาราง........ ช

บทที่ 1 บทนำ........ 1

บทที่ 2 ทฤษฎีเบื้องต้น........ 3

2.1 ไมโครคอลโทรลเลอร์ 8951........ 3

2.2 มาตรฐานพอร์ตอนุกรมแบบ RS-232........ 23

2.3 การเหนี่ยวนำไฟฟ้า........ 27

บทที่ 3 แนวคิดและการออกแบบ........ 30

3.1 ชุดควบคุม........ 31

3.2 ชุดสั่งการควบคุมและการแสดงผล........ 31

3.3 ชุดตรวจสอบการทำงาน........ 33

บทที่ 4 วงจรและการทำงาน........ 34

4.1 วงจรชุดควบคุม........ 35

4.2 การควบคุมและแสดงผล........ 38

4.3 วงจรตรวจสอบสถานะการทำงาน........ 51

บทที่ 5 การทดลองและผลการทดลอง........ 53

5.1 การทดสอบโปรแกรม CON_SYS........ 53

5.2 ส่วนของชุดควบคุม........ 54

5.3 ส่วนของชุดตรวจสอบสถานะ........ 55

5.4 การทดสอบทั้งโครงงาน........ 57

บทที่ 6 บทสรุปและแนวทางพัฒนา........ 59

บรรณานุกรม

ภาคผนวก

ภาคผนวก ก รายละเอียดของวงจรรวมและโปรแกรม

ภาคผนวก ข คุณสมบัติของอุปกรณ์ที่เกี่ยวข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 (ก) แสดงตำแหน่งขาของไมโครคอนโทรลเลอร์ 8951 4

รูปที่ 2.1 (ข) แสดงวงจรสำหรับรีเซตไมโครคอนโทรลเลอร์ 8951 เมื่อจ่ายพลังงานโดยอัตโนมัติ (Power on reset) 5

รูปที่ 2.2 ตำแหน่งของหน่วยความจำทั้งแบบไบท์และแบบบิท 7

รูปที่ 2.3 โครงสร้างแต่ละบิทในพอร์ตอินพุท/เอาท์พุทของ 8951 16

รูปที่ 2.4 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 0 19

รูปที่ 2.5 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1 20

รูปที่ 2.6 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3 20

รูปที่ 2.7 ความหมายของขาคอนเน็กเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232 24

รูปที่ 2.8 แสดงลักษณะการเชื่อมต่อ DTE แบบนัลโมเด็ม 25

รูปที่ 2.9 สัญญาณที่รับและส่งผ่านพอร์ต RS 232 เทียบกับ TTL/CMOS 26

รูปที่ 2.10 ลักษณะของ RS 232 Level Converters 27

รูปที่ 2.11 สนามแม่เหล็กของขดลวดโซลีนนอยด์ 27

รูปที่ 2.12 วงจรแม่เหล็กที่ประกอบด้วยขดลวดพันบนแกนเหล็กเมื่อให้กระแสไฟฟ้าไหลผ่านในขดลวดจะให้กำเนิดเส้นแรงแม่เหล็กไหลในแกนหลัก 28

รูปที่ 3.1 แนวทางการออกแบบ 30

รูปที่ 4.1 ลำดับการทำงานโดยรวม 34

รูปที่ 4.2 วงจรของชุดควบคุม 35

รูปที่ 4.3 อุปกรณ์เลือกตำแหน่ง 36

รูปที่ 4.4 การใช้งาน 74LS373 พักข้อมูล 37

รูปที่ 4.5 ลักษณะการเชื่อมต่อคอมพิวเตอร์กับชุดควบคุม 39

รูปที่ 4.6 วงจร RS-232 Converter และการเชื่อมต่อ 39

รูปที่ 4.7 (ก) หน้าต่างแสดงรายละเอียด 40

รูปที่ 4.7 (ข) หน้าต่างย่อยแสดงคณะผู้จัดทำ 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 4.7 (ค)	หน้าต่างยกเลิกการทำงาน	41
รูปที่ 4.8	หน้าต่างป้อนรหัสผ่าน	41
รูปที่ 4.9	หน้าต่างหลักของโปรแกรม	42
รูปที่ 4.10	หน้าต่างกำหนดเอง	42
รูปที่ 4.11	หน้าต่างคุณสมบัติ	43
รูปที่ 4.12	หน้าต่างเปลี่ยนรหัสผ่าน	43
รูปที่ 4.13	หน้าต่างช่วยเหลือ	44
รูปที่ 4.14	แสดงวงจรจำลองสัญญาณ DTMF	49
รูปที่ 4.15	แสดงวงจรภาคถอดรหัสสัญญาณ DTMF	50
รูปที่ 4.16	แสดงวงจรตรวจสอบสถานะการทำงาน	51
รูปที่ 5.1	วงจรตรวจสอบสถานะและส่วนที่สัมพันธ์กับไฟ 220 Vac	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 การจำแนกแอดเดรสรีจิสเตอร์ 9

ตารางที่ 2.2 การกำหนดค่าของบิทที่อยู่ภายในรีจิสเตอร์PSW 9

ตารางที่ 2.3 ความสามารถในการอ้างถึงแบบบิทของรีจิสเตอร์ 11

ตารางที่ 2.4 ความหมายของรีจิสเตอร์โปรแกรมสเตตัสเวิร์ด(PSW) 13

ตารางที่ 2.5 ความหมายของรีจิสเตอร์โปรแกรมเพาเวอร์คอนโทรล(PCON) 14

ตารางที่ 2.6 อัตราการกำเนิด baud rate ค่ามาตรฐานต่าง ๆ 22

ตารางที่ 2.7 ความหมายของขาคอนเน็กเตอร์พอร์ตอนุกรมตามมาตรฐานRS-232 25

ตารางที่ 4.1 การจัดตำแหน่งของพื้นที่หน่วยความจำ 36

ตารางที่ 4.2 ความหมายของรหัส SR 47

ตารางที่ 5.1 แสดงการทดสอบการรับส่งข้อมูลของคอมพิวเตอร์ 53

ตารางที่ 5.2 ผลการทดสอบของชุดควบคุม 54

ตารางที่ 5.3 การรับส่งข้อมูลระหว่างคีย์ และ MCS-51 55

ตารางที่ 5.4 ผลการทดลองชุดตรวจสอบสถานะเพื่อหาขนาดรอบของN2ที่เหมาะสม 56

ตารางที่ 5.5 การทดสอบการเปิดปิดอุปกรณ์ไฟฟ้าในกรณีอุปกรณ์ไฟฟ้าปกติ 57

ตารางที่ 5.6 การทดสอบกับอุปกรณ์ไฟฟ้าที่ชำรุด 58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ความเจริญก้าวหน้าของเทคโนโลยี คอมพิวเตอร์ และไมโครคอนโทรลเลอร์ ในปัจจุบันก้าวไปอย่างรวดเร็ว โดยเฉพาะเทคโนโลยีด้านคอมพิวเตอร์มีการพัฒนาอย่างรวดเร็วมาก คอมพิวเตอร์มีราคาถูกลงในขณะที่คุณภาพดีขึ้นจนเกือบจะกลายมาเป็นเครื่องใช้ประจำบ้านของทุกบ้านในปัจจุบันแล้ว คาดว่าในอนาคตทุกบ้านคงต้องมีคอมพิวเตอร์ประจำบ้าน ซึ่งหาซื้อได้ในราคาถูก การใช้คอมพิวเตอร์ควบคุมอุปกรณ์ต่าง ๆ คงเป็นสิ่งที่ผู้ใช้ต้องการ เพื่อความสะดวกสบายของตนเอง แต่คอมพิวเตอร์ไม่สามารถทำทุกอย่างได้ด้วยตัวมันเองทั้งหมด จำเป็นต้องเชื่อมต่อกับอุปกรณ์อื่น คอมพิวเตอร์จะเป็นเพียงส่วนเชื่อมต่อกับผู้ใช้หรือส่วนประมวลผลเท่านั้น

โครงงาน "ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายโทรศัพท์ควบคุมด้วยพีซีและอาร์ซียู" ซึ่งก็มีความหมายในตัวมันเองคือ การควบคุมอุปกรณ์เครื่องใช้ไฟฟ้าต่าง ๆ ด้วยคอมพิวเตอร์ เราจะใช้คอมพิวเตอร์เป็นส่วนเชื่อมต่อกับผู้ใช้งาน หรือเป็นส่วนแสดงผล และสั่งการควบคุมอุปกรณ์ไฟฟ้าต่าง ๆ จะใช้คอมพิวเตอร์เชื่อมต่อกับไมโครคอนโทรลเลอร์ตระกูล MCS-51 (ซึ่งเป็นไมโครคอนโทรลเลอร์ที่นิยมใช้ตัวหนึ่ง เนื่องจากมีคุณสมบัติที่หน้าสนใจหลายอย่าง เช่น มีการควบคุมพอร์ทได้เป็นแบบบิท สามารถรับและส่งข้อมูลแบบอนุกรมได้ เป็นต้น ซึ่งจะกล่าวต่อไปในบทที่ 2) โดยผ่านทางพอร์ทอนุกรมของคอมพิวเตอร์ที่ใช้ตามมาตรฐาน RS – 232 ดังนั้นที่ไมโครคอนโทรลเลอร์จะต้องแปลงระดับแรงดันตามมาตรฐานของ RS – 232 ให้เป็นระดับแรงดันที่ใช้กับไมโครคอนโทรลเลอร์เองเราจะใช้ไอซีเบอร์ MAX 232 เป็นตัวแปลงระดับแรงดัน

โดยไมโครคอนโทรลเลอร์จะเป็นหัวใจหลัก เป็นตัวรับคำสั่งจากคอมพิวเตอร์แล้วนำคำสั่งที่ได้ไปตัดสินใจเลือกอุปกรณ์ที่จะควบคุม และคอยตรวจสอบสถานะการทำงานของโหลดที่ถูกส่งกลับมาจากชุดตรวจสอบสถานะ ซึ่งจะต้องใช้ความเข้าใจเป็นพิเศษในการเขียนโปรแกรมเพื่อควบคุมการทำงานของไมโครคอนโทรลเลอร์ เพื่อทำให้ไมโครคอนโทรลเลอร์มีความฉลาดรวดเร็วที่สุด และสัมพันธ์กับคอมพิวเตอร์ หรือสื่อสารกันได้อย่างมีประสิทธิภาพ ในการสื่อสารระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์ จำเป็นต้องมีรูปแบบหรือกติกาที่เป็นมาตรฐานในการรับและส่งข้อมูล เพื่อให้เกิดความเข้าใจตรงกัน ซึ่งได้มีการกำหนดมาตรฐานต่าง ๆ ไว้ใช้ในโครงงานนี้แล้ว ดังจะได้กล่าวต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการควบคุมอุปกรณ์ไฟฟ้าจำเป็นต้องมีความระมัดระวังเป็นพิเศษ เพราะอาจเกิดอันตรายได้ ในโครงงานนี้จึงได้มีการแยกระบบไฟของชุดควบคุมกับชุดขับโหลด ซึ่งมีระดับแรงดันไฟสูงออกจากกันโดยสิ้นเชิง โดยการแยกทางแสงและการเหนี่ยวนำสนามไฟฟ้า ทำให้เกิดความปลอดภัยสูงสุดกับผู้ใช้

ชุดตรวจสอบสถานะ การทำงานจะใช้หลักการเหนี่ยวนำกระแส โดยให้ต่อขดลวดพันบนแกนเฟอร์ไรท์ขนาด 1.5 เซนติเมตร (เส้นผ่านศูนย์กลางของแกนตามลักษณะของแกนเทอร์ลอยทรงกลม) โดยพันขดลวดประมาณ 3 – 9 รอบ ทางด้านขดปฐมภูมิ และพันลวดประมาณ 40 รอบ ในขดทุติยภูมิ แล้วนำมาต่ออนุกรมกับสายไฟอุปกรณ์ไฟฟ้า เมื่อมีการกระตุ้นให้อุปกรณ์ไฟฟ้าทำงานด้วยชุดขับโหลดทำให้มีกระแสไฟไหลผ่านขดลวดไปยังอุปกรณ์ไฟฟ้า อุปกรณ์ไฟฟ้าทำงานและกระแสที่ไหลผ่านขดปฐมภูมิจะเกิดการเหนี่ยวนำข้ามขดไปยังทุติยภูมิ ทำให้เกิดกระแสไหลในขดทุติยภูมิ แต่กระแสที่ไหลมีขนาดต่ำมากจึงจำเป็นต้องมีการขยายก่อนที่จะส่งกลับไปยังไมโครคอนโทรลเลอร์ เป็นที่น่าสังเกตุว่าการใช้หลักการนี้ในการตรวจสอบสถานะการทำงานของโหลดเราจะได้ทราบว่าโหลดทำงานจริงหรือไม่ เพราะถ้ามีกระแสไหลแสดงว่าโหลดทำงาน จึงเป็นข้อดีประการหนึ่งของโครงงานนี้

ในการใช้คอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้านั้นในบางครั้งคอมพิวเตอร์อาจมีปัญหา ไม่สามารถเปิดหรือเข้าโปรแกรมได้ (ซึ่งมีโอกาสเกิดขึ้นได้) จึงได้มีการสร้างวิธีการสั่งงานอีกรูปแบบหนึ่ง เพื่อสำรองไว้ใช้งานหรือยามที่ไม่ต้องการเปิดคอมพิวเตอร์ โดยการใช้คีย์ขนาดเมตริก 3 X 4 เป็นตัวสั่งงานแทน และเป็นRCUด้วย มีการแสดงผลการกดคีย์ด้วยเสียง และแสดงสถานะ การทำงานของโหลดด้วยหลอด LED จึงทำให้โครงงานนี้มีความคล่องตัวมาก

ชุดขับโหลดจะรวมอยู่กับชุดตรวจสอบสถานะ การทำงาน โดยแยกเป็นอิสระจากไมโครคอนโทรลเลอร์ และใช้สายโทรศัพท์เป็นตัวเชื่อมต่อระหว่างชุดทั้งสอง เหตุที่เลือกใช้สายโทรศัพท์เนื่องจากในสายโทรศัพท์หนึ่งเส้นจะประกอบไปด้วยสายที่อยู่ภายใน 4 เส้น ซึ่งเราจำเป็นต้องใช้มันในการส่งสัญญาณควบคุมและรับสัญญาณตรวจสอบสถานะ และสายโทรศัพท์ยังหาซื้อได้โดยทั่วไปตามท้องตลาด

เครื่องควบคุมอุปกร์ไฟฟ้าชุดนี้ได้ผ่านการทดสอบแล้วและยืนยันด้วยผลการทดลองที่บันทึกไว้ว่าสามารถใช้งานได้จริง ซึ่งจะก่อให้เกิดความสะดวกสะบายต่อผู้ใช้ในการนำไปควบคุมอุปกรณ์ไฟฟ้าต่าง ๆ ได้มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีเบื้องต้น

2.1 ไมโครคอนโทรลเลอร์ 8951

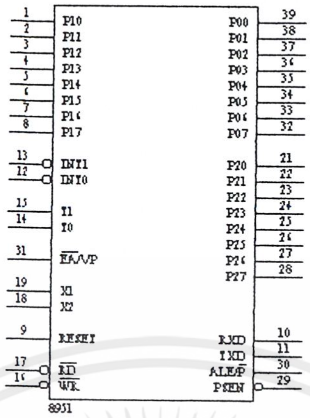
ชิพไมโครโปรเซสเซอร์ตระกูล MCS – 51 เป็นชิพที่ใช้ในการควบคุมโดยเฉพาะ จากจุดประสงค์หลักดังกล่าวนี้ จึงมีวงจรพื้นฐานที่จำเป็นรวมอยู่ในชิพแล้ว ทำให้สามารถทำงานได้โดยไม่จำเป็นต้องพึ่งอุปกรณ์ภายนอกอีกเท่าใดนัก ไมโครโปรเซสเซอร์ตระกูลนี้จึงมีชื่อเรียกอีกอย่างหนึ่งว่า ***ไมโครคอนโทรลเลอร์*** หรือ ***ซิงเกิลชิพ (Single chip)***

2.1.1 การจัดขาและลักษณะภายนอกของ 8951

- *ขา Vss (ขา120)* สำหรับต่อลงกราวด์
- *ขา Vcc (ขา 40)* สำหรับต่อแหล่งจ่ายแรงดันไฟฟ้ากระแสตรงขนาด 5 โวลท์
- *ขา พอร์ท 0 (ขา 32-39)* ใช้เป็นขาสำหรับพอร์ท 0 ขนาด 8 บิท (P0.0-P0.7) แบบโอเพนเดรนไบไดเรกชันแนล (Open Drain Bi-directional) พอร์ทนี้สามารถใช้งานเป็นอินพุท เอาท์พุททั่วไปได้ โดยหากใช้งานเป็นอินพุทพอร์ท ต้องโหลดค่า “1” ไปยังแต่ละบิทของพอร์ทนี้ เพื่อบังคับใช้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ High impedance) นอกจากใช้งานเป็นอินพุทเอาท์พุทแล้วทั่วไปแล้ พอร์ท 0 ยังใช้ในการติดต่อหน่วยความจำภายนอกชิพด้วย โดยส่งค่าแอดเดรสไบท์ต่ำ (A0 – A7) และมัลติเพล็กซ์กับการรับส่งข้อมูล (D0 – D7) จากหน่วยความจำภายนอกโดยมีวงจรพูลอัพภายใน
- *ขา พอร์ท 1 (ขา 1-8)* ใช้เป็นขาสำหรับพอร์ท 1 ขนาด 8 บิท (P1.0-P1.7) พร้อมด้วยการพูลอัพภายใน พอร์ทนี้สามารถใช้งานเป็นอินพุทเอาท์พุททั่วไปได้ เมื่อถูกกำหนดเป็นอินพุทต้องโหลดค่า “1” ไปยังแต่ละบิทของพอร์ทนี้ จะถูกยกระดับแรงดันขึ้นเป็น “1” ด้วยการพูลอัพภายใน แต่สามารถถูกดึงระดับสัญญาณให้ลงต่ำเป็น “0” ได้จากวงจรภายนอกจะเป็นการจ่ายกระแสออกเนื่องจากการพูลอัพภายใน นอกจากนี้ยังใช้งานในหน้าที่พิเศษ ตามรายการข้างล่างนี้

ขาพอร์ท	ขา	การทำงานตามฟังก์ชั่นพิเศษ
P1.0	1	ใช้เป็นอินพุทให้เคาน์เตอร์ของไทเมอร์ 2
P1.1	2	ใช้เป็นสัญญาณควบคุมการทำงานของไทเมอร์ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 (ก) แสดงตำแหน่งขาของไมโครคอนโทรลเลอร์ 8951

- *ขา พอร์ท 2 (ขา 21-28)* ใช้เป็นขาสำหรับพอร์ท 2-ขนาด 8 บิท (P2.0-P2.7) แบบโอเพนเดรนไบไดเรกชันแนล (Open Drain Bi-directional) พอร์ทนี้สามารถใช้งานเป็นอินพุทเอาท์พุททั่วไปได้ โดยหากใช้งานเป็นอินพุทพอร์ท ต้องโหลดค่า "1" ไปยังแต่ละบิทของพอร์ทนี้ เพื่อบังคับใช้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ High impedance) นอกจากใช้งานเป็นอินพุทเอาท์พุททั่วไปแล้วแล้ว พอร์ท 2 ยังใช้ในการติดต่อหน่วยความจำภายนอกชิพด้วย โดยใช้สำหรับส่งค่าแอดเดรสไบท์สูง (A8 – A15) และมีวงจรพูลอัพภายใน

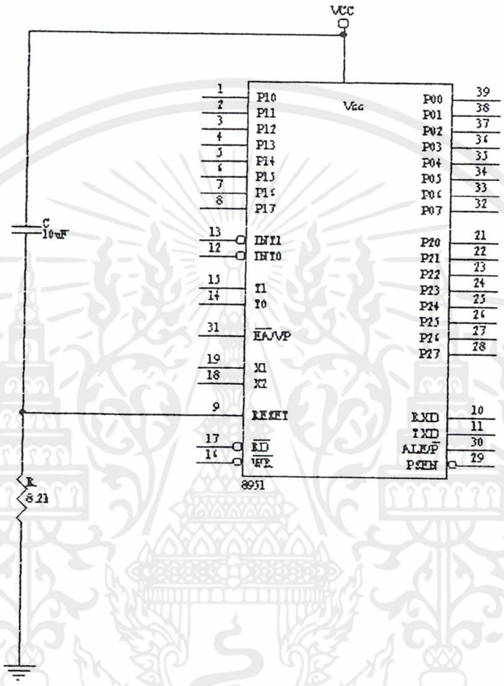
- *ขา พอร์ท 3 (ขา 10-17) ใช้เป็นขาสำหรับพอร์ท* 3 (P3.0-P3.7) สามารถใช้งานเป็นอินพุทเอาท์พุททั่วไปได้ หากใช้งานเป็นอินพุทพอร์ท ต้องโหลดค่า "1" ไปยังแต่ละบิทของพอร์ทนี้ เพื่อบังคับใช้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ High impedance) โดยใช้วงจรพูลอัพภายใน นอกจากนี้ยังใช้งานในหน้าที่พิเศษ ตามรายการข้างล่างนี้

ขาพอร์ท	ขา	การทำงานตามฟังก์ชั่นพิเศษ	
P3.0	10	RxD	พอร์ทอนุกรมอินพุท
P3.1	11	TxD	พอร์ทอนุกรมเอาท์พุท
P3.2	12	INT0	อินเตอร์รัปต์ภายนอกตัวที่ 0
P3.3	13	INT1	อินเตอร์รัปต์ภายนอกตัวที่ 1
P3.4	14	T0	สัญญาณกระตุ้นเข้าที่ตัวนับ 0
P3.5	15	T1	สัญญาณกระตุ้นเจ้าที่ตัวนับ 1
P3.6	16	WR	สัญญาณควบคุมการเขียน
P3.7	17	RD	สัญญาณควบคุมการอ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การที่จะทำงานตามฟังก์ชั่นข้างบน จะต้องเริ่มโปรแกรมด้วยการส่งค่า "1" ไป แลทช์ ไว้ก่อนที่จะให้ทำงานตามฟังก์ชั่นข้างบน

- *ขา RST(ขา 9)* ต้องคงสถานะค่าสูงเป็นเวลาประมาณอย่างน้อยสองวัฏจักร (2 Machine cycle) ระหว่างที่ ออสซิลเลเตอร์ ทำงานขณะที่ต้องการรีเซตทั้งระบบงาน



รูปที่ 2.1 (ข) แสดงวงจรสำหรับรีเซตไมโครคอนโทรลเลอร์ 8951 เมื่อจ่ายพลังงานโดยอัตโนมัติ (Power on reset)

- *ขา ALE/PROG (ขา 30)* เป็นขาแอดเดรสแลทซ์อีเนเบล (Address latch enable) ด้วยการส่งพัสส์ ออกไปใช้สำหรับแลทซ์ค่าแอสเดรสไบท์ต่ำจากพอร์ท 0 ในระหว่างการเข้าถึงข้อมูลจากหน่วยความจำภายใน ALE จะถูกส่งสัญญาณนาฬิกาออกมา ในอัตราเร็วคงที่ ที่ 1/8 ของความถี่ออสซิลเลเตอร์ ตลอดเวลา แม้ว่าจะไม่มีการเข้าถึงข้อมูลภายใน ดังนั้น จึงสามารถที่จะให้สัญญาณจากขานี้เป็นตัวส่งสัญญาณภายนอก หรือเป็นสัญญาณนาฬิกา แต่อย่างไรก็ตามความถี่สัญญาณนี้จะลดความถี่ช้าลงไปครึ่งหนึ่ง ระหว่างการติดต่อกับหน่วยความจำภายนอก และ ขานี้ยังใช้ควบคุมการเขียนโปรแกรมลงไปใน EPROM ภายในชิพอีกด้วย

- *ขา PSEN (ขา 29)* เป็นสโตรป (Strop) สำหรับอ่านข้อมูลจากโปรแกรมหน่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความจำภายนอก เมื่อชิพทำงานด้วยโปรแกรมจากหน่วยความจำภายนอก ขา PSEN จะสร้างสโตรปต่อสองครั้งภายใน 1 แมชีนไซเคิล (machine cycle)

- *ขา EA/Vpp (ขา 31)* เป็นขาสำหรับใช้เลือกให้ MCS-51 ทำงานจากโปรแกรมที่อยู่ภายในหรือภายนอกชิพ ถ้ามีสถานะต่ำตัว MCS-51 ทำงานตามหน่วยความจำภายนอก ซึ่งขยายโปรแกรมได้ยาวถึง 64 กิโลไบท์ และหากมีสถานสูง MCS-51 จะใช้โปรแกรมจากหน่วยความจำที่เก็บโปรแกรมอยู่ภายในชิพ

- *ขา XTAL 2(ขา 18)* ใช้เป็นตัวเอาท์พุทจากตัวออสซิลเลเตอร์
- *ขา XTAL 1(ขา 19)* ใช้เป็นตัวอินพุทเข้าสู่ตัวออสซิลเลเตอร์

2.1.2 การจัดหน่วยความจำและการเชื่อมต่อ

หน่วยความจำของไมโครคอนโทรลเลอร์ 8951 แบ่งเป็น 2 ประเภท คือ

1. *หน่วยความจำโปรแกรม (Program Memory)* จำหน่วยความจำประเภทนี้ คือ ROM ใช้เก็บโปรแกรมที่ใช้ในการควบคุมระบบ ซึ่งเป็นหน่วยความจำประเภท non - volatile

2. *หน่วยความจำข้อมูล (Data Memory)* หน่วยความจำประเภทนี้ได้แก่ RAM ใน MCS-51 เบอร์ 8951 จะมีหน่วยความจำโปรแกรมในชิพ และสามารถอ้างหน่วยความจำโปรแกรมภายนอกได้มากที่สุด และอ้างหน่วยความจำข้อมูลภายนอกได้มากที่สุด 64 กิโลไบท์ สำหรับหน่วยความ RAM จะประกอบด้วย พื้นที่ใช้งานทั่วไป, รีจีสเตอร์แบงก์ พื้นที่ใช้งานระดับบิท และรีจีสเตอร์ฟังก์ชั่นพิเศษใน8951 จะมีหน่วยความจำภายในตั้งแต่ตำแหน่ง 00H ถึง FFH และสามารถอ้างหน่วยความจำโปรแกรมภายนอกได้ ถ้าอ่านข้อมูลจากหน่วยความจำโปรแกรมขา PSEN จะแอกทีฟ นอกจากนี้ 8951 สามารถอ้างถึงหน่วยความจำข้อมูลภายนอกได้ 64 กิโลไบท์ โดยการติดต่อกับหน่วยความจำนี้ขา RD และ WR จะแอกทีฟ สำหรับหน่วยความจำข้อมูลภายในจะแบ่งออกได้ ดังนี้

1.) ชุดรีจีสเตอร์4 ชุดแต่ละชุดเรียกว่ารีจีสเตอร์แบ็งค์ ที่ตำแหน่ง00H – 1FH โดยแต่ละชุดประกอบด้วยรีจีสเตอร์R0 ถึง R7

2.) หน่วยความจำสามารถเข้าถึงข้อมูลระดับบิทได้ ตำแหน่ง 20H ถึง 2FH

3.) หน่วยความจำใช้งานทั่วไปตำแหน่ง 30H ถึง 7FH

4.) รีจีสเตอร์ฟังก์ชั่นพิเศษตำแหน่ง 80H ถึง FFH

แผนผังการจัดหน่วยความจำข้อมูลภายในแสดงได้ดังรูปซึ่งจะมีการอ้างถึงตำแหน่งหน่วยความจำข้อมูลภายในจะอ้างได้สองแบบ คือ การอ้างไปที่ตำแหน่งของไบท์ (เขียนหมายเลขตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Byte Address		Bit Address							
Bit - Address Location	7F	General Purpose RAM							
	30								
	2F	8F	8E	8D	8C	8B	8A	#	#
	2E	#	#	#	#	#	#	#	#
	2D	7F	7E	7D	7C	7B	7A	#	#
	2C	#	#	#	#	#	#	#	#
	2B	6F	6E	6D	6C	6B	6A	#	#
	30	#	#	#	#	#	#	#	#
	29	4F	4E	4D	4C	4B	4A	#	#
	28	#	#	#	#	#	#	#	#
	27	3F	3E	3D	3C	3B	3A	#	#
	26	#	#	#	#	#	#	#	#
	25	2F	2E	2D	2C	2B	2A	#	#
	24	#	#	#	#	#	#	#	#
	23	1F	1E	1D	1C	1B	1A	#	#
	22	#	#	#	#	#	#	#	#
	21	0F	0E	0D	0C	0B	0A	9	8
	20	7	6	5	4	3	2	1	0
	1F–18	BANK 3							
	17–10	BANK 2							
	0F–8	BANK 1							
	7–0	Default register Bank for R0 - R7							
		RAM							

Byte Address	Bit Address								
FF									
FE	F7	E6	F5	F4	F3	F2	F1	F0	B
E0	F7	E6	E5	E4	E3	E2	E1	E0	AAC
D0	D7	D6	D5	D4	D3	D0	-	D0	PSW
B8	-	-	-	B0	BB	BA	B9	B8	IP
B0	B7	B6	B5	B4	B3	B2	B1	B0	P3
A8	AF	AE	AD	AC	AB	AA	A9	A8	IE
A0	A7	A6	A5	A4	A3	A2	A1	A0	P2
99	Not bit address								SBUF
98	9F	9E	9D	9C	9B	9A	#	#	SCON
90	#	#	#	#	#	#	#	#	P1
8D	Not bit address								TH1
8C	Not bit address								TH0
8B	Not bit address								TL1
8A	Not bit address								TL0
89	Not bit address								TMOD
88	8F	8E	8D	8C	8B	8A	#	#	TCON
87	Not bit address								PCON
83	Not bit address								PCON
82	Not bit address								DPL
81	Not bit address								SP
80	#	#	#	#	#	#	#	#	PO
	Spatial		Function			Registers			

รูปที่ 2.2 ตำแหน่งของหน่วยความจำทั้งแบบไบท์และแบบบิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้านนอก) หรือการอ้างไปที่ตำแหน่งของบิท (เขียนหมายเลขตำแหน่งด้านใน) โดยตำแหน่งของหน่วยความจำที่อ้างเป็นแบบบิทได้มีตำแหน่ง บิทที่แน่นอน

ใน 8951 จะแบ่งหน่วยความจำประเภทนี้เป็นอีก 2 ประเภท หน่วยความจำโปรแกรมภายนอก (external memory) กล่าวคือ ROM ที่มาต่อภายนอกตัว 8951 ส่วนอีกประเภทคือ หน่วยความจำภายใน (Internal memory) ได้แก่ ROM ภายในตัวไมโครคอนโทรลเลอร์เอง การเลือกใช้หน่วยความจำโปรแกรมภายนอก หรือหน่วยความจำโปรแกรมภายในทำได้โดยการให้สัญญาณทางไฟฟ้าที่ขา EA โดย

- สัญญาณทางไฟฟ้าที่ขา EA/ เป็นลอจิก 0 หมายถึง หน่วยความจำโปรแกรมภายนอก

- สัญญาณทางไฟฟ้าที่ขา EA/ เป็นลอจิก 1 หมายถึง หน่วยความจำโปรแกรมภายใน

*<u>หมายเหตุ</u> หน่วยความจำโปรแกรมนั้น ไมโครคอนโทรลเลอร์ตระกูล 8951 สามารถอ้างข้อมูลได้ 64 กิโลไบท์

หน่วยความจำข้อมูล

หน่วยความจำข้อมูล มีหน้าที่สำหรับเก็บข้อมูล หรือตัวแปรที่เกิดขึ้นในขณะที่กำลังประมวลผลโปรแกรมไว้เป็นการชั่วคราว โดยพื้นฐานแล้วหน่วยความจำข้อมูลจัดเป็นหน่วยความจำ RAM แบบสแตติก ดังนั้น เมื่อไม่มีการจ่ายไฟฟ้าให้กับระบบ ก็จะมีผลทำให้ข้อมูลที่จัดเก็บไว้ภายในหน่วยความจำนี้สูญหายไป

พื้นที่ของหน่วยความจำข้อมูลของ 8951 สามารถมีได้สูงสุดไม่เกิน 64 กิโลไบท์ และแยกประเภทออกเป็นสองลักษณะตามตำแหน่งที่ตั้งของหน่วยความจำนั้น

- *หน่วยความจำโปรแกรมภายใน(Internal Data Memory)* ซึ่งเป็นหน่วยความจำข้อมูลที่อยู่ภายในตัวของไอซีไมโครคอนโทรลเลอร์เอง

- *หน่วยความจำข้อมูลภายนอก (External Data Memory)* ซึ่งเป็นการใช้ไอซีหน่วยความจำข้อมูลมาเพิ่มเติมเข้าไปในวงจร ลักษณะเดียวกับการนำไอซี EPROM มาใช้งานเป็นหน่วยความจำโปรแกรมนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำข้อมูลภายใน

หน่วยความจำข้อมูลภายใน 8951 มีจำนวนทั้งหมด 256 ไบท์ โดยจำแนกออกเป็นสองลักษณะ คือ พื้นที่เฉพาะสำหรับตัวประมวลผลกลาง หรือเรียกว่า ริจิสเตอร์ R0 - R7 และพื้นที่ใช้งานทั่วไปสำหรับโปรแกรมใช้งานที่ผู้ใช้สร้างขึ้นมา

หน่วยความจำขนาด 128 ไบท์แรกบริเวณแอดเดรส 00H - 1FH จำนวน 32 ไบท์ จำแนกออกเป็นกลุ่ม (Blank) 8 ไบท์ จำนวน 4 กลุ่ม ซึ่งมีชื่อเรียกว่า รีจิสเตอร์R0 - R7 ดังตารางต่อไปนี้

ตารางที่ 2.1 การจำแนกแอดเดรสรีจิสเตอร์

แอดเดรส	รีจิสเตอร์แบงก์	ชื่อรีจิสเตอร์ที่ใช้งาน
00H – 07H	0	R0 - R7
08H - 0FH	1	R0 - R7
10H – 17H	2	R0 - R7
18H - 1FH	3	R0 - R7

จะเป็นได้ว่าชื่อของรีจิสเตอร์ไม่ว่าจะอยู่ในรีจิสเตอร์แบงก์ใด ก็จะมีชื่อR0 ถึง R7 เหมือนกันทั้งสิ้น ดังนั้น ในการใช้งานผู้ใช้จะต้องให้ความระมัดระวังว่าต้องการรีจิสเตอร์ นั้น ๆ จากแบงก์ในการสวิทช์ เลือกแต่ละกลุ่มของรีจิสเตอร์นี้ก็ทำได้ง่าย เพียงการกำหนดค่าของบิทที่อยู่ภายในรีจิสเตอร์ PSW เท่านั้นตามตาราง ต่อไปนี้

ตารางที่ 2.2 การกำหนดค่าของบิทที่อยู่ภายในรีจิสเตอร์ PSW

รีจิสเตอร์แบงก์	บิท RS0	บิท RS1	ตำแหน่งหน่วยความจำ
0	0	0	0000H
1	0	1	0008H
2	1	0	0010H
3	1	1	0018H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตามโดยทั่วไปก็มักจะมีการใช้งานรีจิสเตอร์ R0 - R7 เฉพาะในแบงก์ 0 เท่านั้น ดังนั้นพื้นที่ของแบงก์อื่น ๆ ที่เหลือก็สามารถนำมาใช้ในลักษณะของหน่วยความจำข้อมูลภายในปกติด้วยการอ้างถึงหมายเลขของ แอดเดรสนั้น ๆ โดยตรง

บริเวณแอดเดรส 20H -2FH จำนวน 16 ไบท์ บริเวณพื้นที่เป็นส่วนสำหรับผู้ใช้ซึ่งจะมีความพิเศษต่างไปหน่วยความจำส่วนอื่น ๆ เนื่องจาก ผู้ใช้อาจสามารถอ้างถึงหน่วยความจำบริเวณนี้ได้ทั้งในลักษณะของไบท์ข้อมูล เช่น ปกติ หรืออาจจะเป็น บิทข้อมูล ได้โดยตรง ดังนั้นหากว่ามองในลักษณะบิทข้อมูลแล้ว ก็จะมีพื้นที่ตัวแปรแบบบิทให้ใช้งานได้มากถึง 128 บิท โดยตำแหน่งแรกของบิทจะเป็นบิทซึ่งเริ่มต้นนับจากบิทนัยสำคัญต่ำสุด (LSB) ของแอดเดรส 20H เรื่อยไปจนกระทั่งถึงบิทที่ 127 ซึ่งเป็นบิทนัยสำคัญสูงสุด (MSB) ของแอดเดรส 2FH

ความสามารถในการใช้งานพื้นที่ส่วนนี้แบบบิทข้อมูลโดยตรงนี้นับว่าน่าสนใจมาก และถือเป็นการใช้งาน 8951 อย่างเต็มประสิทธิภาพทีเดียว เนื่องจากว่า 8951 ได้รับการออกแบบมาก็มักจะเป็นเพียงการอ่านค่าสถานะลอจิก ของสัญญาณ หรือกรณีการส่งออกข้อมูลก็จะเป็นการกำหนดสภาวะลอจิกให้กับวงจรภายนอกผ่านทางบิทใดบิทหนึ่งอยู่แล้ว ดังนั้นหากว่ามีการกำหนดบิทหรืออ่านค่าของบิทมาโดยตรงแทนที่จะต้องทำลอจิกขั้นต้นกับข้อมูลทั้งไบท์เพื่อต้องการทราบผลเพียงหนึ่งบิท เช่นที่กระทำกันในโปรเซสเซอร์โดยทั่วไป ก็จะเพิ่มความสะดวกและรวดเร็วในการเขียนโปรแกรมควบคุมมาก

บริเวณแอดเดรส 30H - 7FH เป็นบริเวณที่สามารถนำไปใช้งานได้อย่างอิสระ โดยสามารถอ้างถึงได้เฉพาะ ในลักษณะของไบท์ข้อมูลตามปกติเท่านั้น

หน่วยความจำขนาด 128 ไบท์ถัดไป พื้นที่ตั้งแต่บริเวณตั้งแต่แอดเดรส 80H - FFH เป็นบริเวณของหน่วยความจำที่มีการใช้งานเฉพาะจาก 8951 เท่านั้น โดยจะนำมาใช้เป็นตำแหน่งของรีจิสเตอร์หน้าพิเศษ (Special Function Register หรือ SFR) จำนวน 20 ตำแหน่ง ดังแสดงแผนการในรูป สำหรับไมโครคอนโทรลเลอร์เบอร์ 8951 จะมีหน่วยความจำข้อมูลภายในสำหรับการใช้งานเพิ่มมากขึ้นกว่าเบอร์อื่น ๆ เช่น 8951 หรือ 8751 อีก 128 ไบท์ โดยจะอยู่บริเวณช่วงแอดเดรส 80H ถึง FFH เช่นกัน ซึ่งแม้ว่า จะเป็นพื้นที่ที่มีหมายเลขแอดเดรสเดียวกับส่วนของ รีจิสเตอร์หน้าที่พิเศษ แต่ในความเป็นจริงแล้วจะเป็นพื้นที่หน่วยความจำอีกบริเวณหนึ่ง ซึ่งมีการซ้อนกัน(Overlap) กันให้อยู่ในบริเวณแอดเดรสส่วนนี้ ซึ่งหากว่าผู้ใช้งานต้องการจะเก็บข้อมูลในพื้นที่บริเวณนี้ก็จะต้องใช้การอ้างถึง หน่วยความจำแบบโดยอ้อม (Indirect Addressing) เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 รีจิสเตอร์หน้าที่พิเศษ

รีจิสเตอร์หน้าที่พิเศษ (SFR) เป็นรีจิสเตอร์สำหรับการความคุมหน้าที่และการทำงานของอุปกรณ์ หรือพอร์ทของ 8951 ทั้งหมด โดยมีตำแหน่งอยู่ในบริเวณแอดเดรส 80H - FFH การใช้งานรีจิสเตอร์หน้าที่พิเศษเหล่านี้ สามารถทำได้ทั้งการระบุถึงชื่อของรีจิสเตอร์ หรือตำแหน่งแอดเดรสที่เป็นของรีจิสเตอร์นั้นก็ได้

ในตารางแสดงเห็นลักษณะการจัดพื้นที่หน่วยความจำ สำหรับรีจิสเตอร์หน้าที่พิเศษเหล่านี้ โดยมีข้อสังเกตว่ารีจิสเตอร์ที่อยู่ตำแหน่งแอดเดรสที่เป็นจำนวนทวีคูณของค่า 8 จะสามารถอ้างถึงในระดับบิทได้ด้วย นั่นคือ แอดเดรส 80H, 88H, 90H, 98H, A0H, A8H, B0H, B8H, D0H, E0H และ F0H)

ตารางที่ 2.3 ความสามารถในการอ้างถึงแบบบิทของรีจิสเตอร์

ชื่อรีจิสเตอร์	การอ้างถึงแบบบิท	ชื่อรีจิสเตอร์	การอ้างถึงแบบบิท
ACC	ได้	IE	ได้
B	ได้	TMOD	ไม่ได้
PSW	ได้	TCON	ได้
SP	ไม่ได้	TH0	ไม่ได้
DPTR	ไม่ได้	TL0	ไม่ได้
P0	ได้	TH1	ไม่ได้
P1	ได้	TL1	ไม่ได้
P2	ได้	SCON	ได้
P3	ได้	SBUF	ไม่ได้
IP	ได้	PCON	ไม่ได้

- *แอกคิวมูเลเตอร์ (Accumulator) หรือ ACC*

เป็นรีจิสเตอร์ขนาด 8 บิท ทำหน้าที่ในการเก็บข้อมูลที่จะส่งให้กับหน่วยทำงานในซีพียูและเก็บผลลัพธ์ ที่ได้จากการทำงานนั้น การทำงานของรีจิสเตอร์นี้มีลักษณะเช่นเดียวกับตัวแอกคิวมูเลตอร์ของโปรเซสเซอร์ทั่วไป การใช้งานภายในโปรแกรมจะเรียกว่า รีจิสเตอร์A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รีจิสเตอร์ B

เป็นรีจิสเตอร์ที่ใช้สำหรับการทำคำสั่งการคูณและหารตัวเลข ในกรณีที่ไม่ใช้ในการคำนวณทางด้านคณิตศาสตร์ ก็สามารถนำไปใช้งานเช่นเดียวกับรีจิสเตอร์ทั่วไปได้

- โปรแกรมเคาน์เตอร์ (Program Counter)

เป็นรีจิสเตอร์ที่ใช้การชี้ตำแหน่งแอดเดรสของหน่วยความจำโปรแกรม ซึ่งจะต้องไปทำงานในลำดับถัดไป การใช้งานภายในโปรแกรมจะเรียกว่า รีจิสเตอร์PC

- สแต็กพอยน์เตอร์ (Stack Pointer)

เป็นรีจิสเตอร์ของ 8 บิท ทำหน้าที่เก็บตำแหน่งของตัวชี้หรือพอยน์เตอร์ (Pointer) ของบริเวณสแต็ก (Stack) สำหรับเก็บข้อมูลแอกคิวมูเลเตอร์ รีจิสเตอร์ต่าง ๆ รวมทั้งข้อมูลจากโปรแกรมโดยปกติแล้วเมื่อทำการเริ่มต้น ระบบใหม่ภายหลังจากการเริ่มจ่ายไฟฟ้า หรือมีการรีเซต (Reset) เกิดขึ้นภายในสแต็กพอย์นเตอร์จะมีค่า 07H ซึ่งเป็นตำแหน่งแอดเดรสภายในบริเวณเนื้อที่ 128ไบท์แรกของหน่วยความจำข้อมูลภายใน การใช้งานภายในโปรแกรมจะเรียกว่ารีจิสเตอร์SP

- ตัวชี้ข้อมูล หรือ ดาต้าพอยน์เตอร์ (Data Pointer)

เป็นรีจิสเตอร์ขนาด16 บิท ซึ่งเรียกว่าDPTR และสามารถใช้งานแยกเป็นรีจิสเตอร์ขนาด 8 บิทสองตัวคือ รีจิสเตอร์ DPH และ DPL เพื่อเก็บค่าของแอดเดรสของหน่วยความจำที่ต้องใช้งานภายในโปรแกรมหรืออาจจะเป็นแอดเดรสของอุปกรณ์ภายนอก ซึ่งกำหนดให้ติดต่อกันโดยใช้ตำแหน่งของหน่วยความจำภายในโปรแกรม

- รีจิสเตอร์ที่เกี่ยวข้องกับพอร์ท (Port Register)

รีจิสเตอร์เหล่านี้จะมีความเกี่ยวข้องกับการทำงานของอินพุทเอาท์พุทโดยตรงซึ่งแต่ละตัวจะเป็นรีจิสเตอร์ขนาด8 บิท สามารถใช้งานได้ทั้งในลักษณะของการอินพุท หรือการเอาท์พุทข้อมูลได้ การดำเนินการใด ๆ ที่เกี่ยวกับพอร์ททั้งสี่นี้จะมีผลทำให้ข้อมูลที่ตำแหน่งของพอร์ทนี้เปลี่ยนแปลงไปเช่นกัน นอกจากนี้พอร์ท P0 และ P2 ยังสามารถนำมาใช้ในการติดต่อกับหน่วยความจำโปรแกรมหรือหน่วยความจำข้อมูลภายนอกได้โดยพอร์ท P2 จะเป็นค่าของแอดเดรส 8 บิทบนของหน่วยความจำ ส่วนพอร์ท P0 นั้นในช่วงเริ่มแรกจะเป็นค่าของแอดเดรส 8 บิทล่างของหน่วยความจำช่วงเวลาต่อมาจึงจะนำพอร์ท P0 ไปใช้เป็นบัสสำหรับการรับ หรือ การส่งข้อมูลกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยอุปกรณ์ภายนอกสำหรับพอร์ท P3 นั้นนอกเหนือจากจะใช้ในฐานะของพอร์ท อินพุท/เอาท์พุท เช่นปกติแล้วยังนำมาใช้ในฐานะ บัสควบคุมเกี่ยวกับสัญญาณอินเตอร์รัปต์ได้อีกด้วย

- โปรแกรมสเตตัสเวิร์ด (PSW)

รีจิสเตอร์นี้ทำหน้าที่บอกถึงแฟล็กสภาวะ การทำวานต่าง ๆ รวมทั้งบิทสำหรับการกำหนดเลือกแบงก์ (Bank) ของรีจิสเตอร์ที่ใช้งานด้วย แสดงได้ดังนี้

* บิท: PSW ตำแหน่ง: D0H ค่าบิทเริ่มต้น: 0000 0111

ตารางที่ 2.4 ความหมายของรีจิสเตอร์ โปรแกรมสเตตัสเวิร์ด (PSW)

ชื่อบิท	ตำแหน่ง	ความหมาย
CY	PSW.7	Carry Flag
AC	PSW.6	Auxiliary Carry Flag
F0	PSW.5	Flag 0
RS1	PSW.4	Select Bank bit 1
RS0	PSW.3	Select Bank bit 0
OV	PSW.2	Overflow Flag
-	PSW.1	
P	PSW.0	Parity Flag

- รีจิสเตอร์ (SBUF)

เป็นบัฟเฟอร์ขนาด 8 บิท สำหรับการสื่อสารข้อมูลแบบอนุกรมทั้งการรับและการส่งข้อมูลซึ่งตามความเป็นจริงแล้วบัฟเฟอร์นี้มีอยู่ด้วยกานสองชุด และแยกจากกันอย่างชัดเจนสำหรับการส่งและการรับ โดยซีพียูจะทำการเลือกบัฟเฟอร์ที่เหมาะสมให้โดยอัตโนมัติ

- รีจิสเตอร์ (PCON)

เป็นรีจิสเตอร์ที่ใช้ในการควบคุมหน้าที่การทำงานในสามลักษณะ ซึ่งได้แก่ การควบคุมการทำงานของโปรเซสเซอร์ (บิท IDL และ PD) การกำหนดอัตราการทวีคูณของอัตราเร็วในการสื่อสารข้อมูลอนุกรม (บิท SMOD) และแฟล็กสภาวะสำหรับการใช้งานทั่วไป (บิท GRO และ GR1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิท PD (Power down) เป็นการกำหนดให้ลดกำลังไฟฟ้าที่จ่ายให้กับส่วนของโปรเซสเซอร์ ภายในลงโดยยังคงมีกำลังไฟฟ้าจ่ายให้กับส่วนหน่วยความจำข้อมูลภายใน ผ่านทางขาสัญญาณ RSR วิธีการนี้มักนำมาใช้ในกรณีที่มีการตรวจสอบการไม่มีกำลังไฟฟ้า (Power failure) โดยวงจรตรวจสอบภายนอกจะต้องมีการอินเตอร์รัปต์เข้ามา เพื่อทำการเก็บข้อมูลที่กำลังประมวลผลอยู่ก่อนและเมื่อมีกระแสไฟฟ้าจ่ายให้เป็นปกติแล้ว จึงค่อยนำข้อมูลมาประมวลผล ต่อไป

บิท IDL (Idle Mode) เป็นการกำหนดให้โปรเซสเซอร์หยุดการทำงานชั่วขณะ (Sleep) และจำกลับมาอยู่ในสภาพปกติอีกครั้งเมื่อ ทำการรีเซตทางฮาร์ดแวร์ หรือมีการอินเตอร์รัปต์อย่างใดอย่างหนึ่ง เกิดขึ้นการทำงานในลักษณะนี้สามารถเกิดขึ้นได้ ก็เนื่องจากว่าสภาวะการหยุดทำงานชั่วขณะนั้น เป็นเพียงการห้ามไม่ให้มีสัญญาณนาฬิกาจ่ายให้ส่วนของ โปรเซสเซอร์เท่านั้นส่วนของวงจรการอินเตอร์รัปต์พอร์ทอนุกรมและวงจรนับ/จับเวลา ยังคงมีสัญญาณนาฬิกาอยู่เป็นปกติ

* ชื่อบิท: PCON ตำแหน่ง: 97H ค่าเริ่มต้น: 0xxx0000

ตารางที่ 2.5 ความหมายของรีจิสเตอร์ โปรแกรมเพาเวอร์คอนโทรล (PCON)

ชื่อบิท	ตำแหน่ง	ความหมาย
SMOD	PCON.7	บิททวีคูณของอัตราบวกปกติ
-	PCON.6	-
-	PCON.5	-
-	PCON.4	-
GF1	PCON.3	แฟลกสำหรับให้ผู้ใช้งานทั่วไป Flag 0
GF0	PCON.2	แฟลกสำหรับให้ผู้ใช้งานทั่วไป Flag 1
PD	PCON.1	บิทสำหรับการกำหนด Power down
IDL	PCON.0	บิทสำหรับการกำหนด Ideal โหมด

- รีจิสเตอร์ IP, TMOD, SCON

เป็นกลุ่มของรีจิสเตอร์ที่ทำหน้าที่กำหนดการควบคุม และการทำงานของอินเตอร์รัปต์ต่าง ๆ ของ 8951

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 พอร์ทอินพุท/เอาท์พุท ของ 8951

พอร์ท มีความหมายถึง แอดเดรสหนึ่งที่ได้รับการกำหนดไว้ เพื่อการโอนย้ายข้อมูลระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก การกำหนดประเภทของการติดต่อขึ้นอยู่กับทิศทางการไหลของข้อมูลเมื่อพิจารณาจากไมโครคอนโทรลเลอร์เป็นหลัก ดังนั้น การนำเข้าข้อมูลจากวงจรภายนอกจึงเรียกว่า อินพุท และในกรณีตรงกันข้ามเพื่อ ส่งออกข้อมูลก็จะเรียกว่า เอาท์พุท

เมื่อพิจารณาถึงวิธีการส่งข้อมูลภายใน พอร์ทจะสามารถแยกประเภทของพอร์ทออกได้เป็น 2 ลักษณะ คือ พอร์ทแบบขนาน (Parallel port) ซึ่งทำการส่งจำนวนบิทข้อมูลทั้งหมดออกมาหรือนำเข้าไปพร้อมกันในคราวเดียว และพอร์ทแบบอนุกรม (Serial port) ซึ่งทำการโอนย้ายข้อมูลคราวละบิท ๆ จนครบจำนวน แต่สำหรับในบทนี้จะกล่าวถึงเฉพาะในส่วนของพอร์ทแบบขนานเท่านั้น สำหรับการทำงานของพอร์ทแบบอนุกรมจะได้กล่าวภายหลัง

1. พอร์ทแบบขนานของ 8951

8951 มีโครงสร้างของพอร์ทที่สามารถใช้งานแบบขนานได้จำนวนทั้งหมด 4 พอร์ท เรียกชื่อตามลำดับว่า พอร์ท 0,1,2 และ 3 และเป็นพอร์ทขนาด 8 บิท ทั้งหมด การใช้งานพอร์ทสามารถทำได้ทั้งในลักษณะของเส้นสัญญาลักษณะเดี่ยว ๆ หรือกลุ่มของสัญญาณได้ นอกจากนี้ พอร์ท 0,1,2 และ 3 ยังสมามรถนำไปใช้งานอื่น ๆ ที่ไม่ใช้เป็นพอร์ท อินพุท/เอาท์พุทได้ โดยพอร์ท 0 จะทำหน้าที่ มัลติเพล็กซ์ ระหว่างบัสแอดเดรสไบท์ต่ำและบัสข้อมูลสำหรับการติดต่อกับวงจรประกอบร่วมข้อมูลบัสแอดเดรสไบท์สูง ซึ่งจะส่งออกมาทางพอร์ท 2 สำหรับพอร์ท 3 นั้น นอกเหนือไปจากความสามารถ เช่น พอร์ทปกติแล้วสามารถนำไปเป็นขาสัญญาณของการอินเตอร์รัปต์ต่าง ๆ ซึ่งรวมทั้งการสร้างสัญญาณ ควบคุม RD และ WR เพื่อทำหน้าที่อ่านหรือเขียนหน่วยความจำข้อมูลภายนอกด้วย การใช้งานพอร์ทลักษณะงานแบบอื่น ๆ ที่ไม่ใช่เป็นพอร์ทเอาท์พุทนี้จะดำเนินการโดย 8951 เองโดยอัตโนมัติ

1.1 โครงสร้างการทำงานของพอร์ท 8951

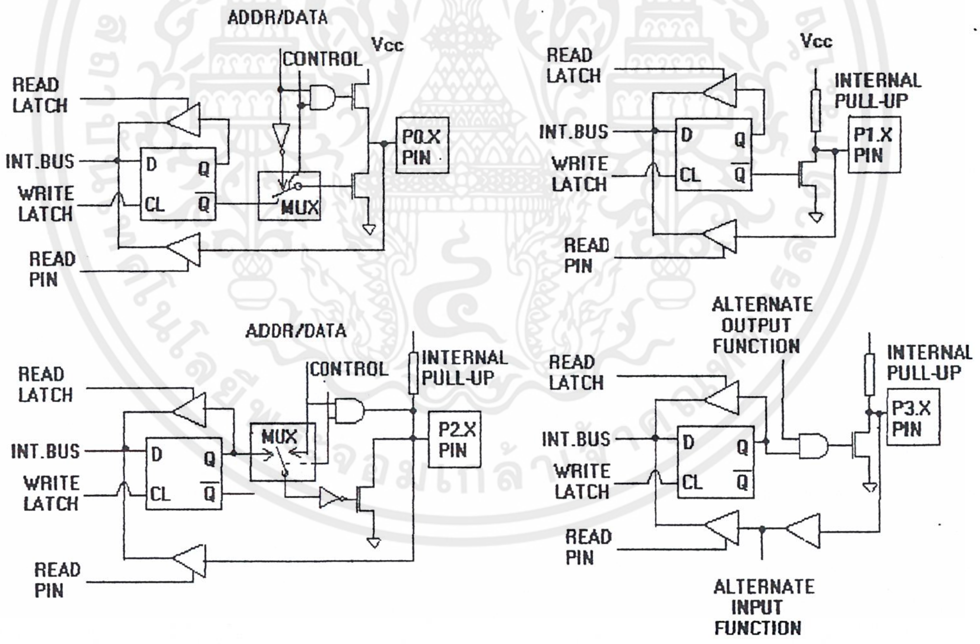
จากลักษณะโครงสร้างของแต่ละบิท ภายในพอร์ททั้งหมดของ 8951 ซึ่งได้แสดงไว้ในรูปนั้นจะเห็นว่ามีความคล้ายคลึงกันตามลักษณะโครงสร้างที่เรียกว่า “พอร์ทควอซิไบไดเร็กชันเนล” (Quasi - bidirectional port) ยกเว้นพอร์ท 0 ซึ่งเพียงแต่ไม่มีตัวต้านทานทำหน้าที่ พูลอัพสัญญาณไว้ภายในเท่านั้น ซึ่งมีผลให้บิทนั้น ๆ ของพอร์ท 1,2 และ 3 เป็นสภาวะของลอจิกสูง ตัวต้านทานนี้มีค่าประมาณ 50 kΩ ซึ่งเป็นค่าที่สูงมาก และทำให้อุปกรณ์ภายนอกสามารถขับสัญญาณของพอร์ทเหล่านี้เป็นลอจิกต่ำได้ง่าย สำหรับบิทของพอร์ท 0 นั้น แม้ว่าจะมีหลักการทำงานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คล้ายคลึงกันกับบิทของพอร์ทอื่น ๆ แต่เนื่องจากการที่ไม่มีตัวต้านทานทำหน้าที่ พูลอัพภายในไว้ทำให้เมื่อทรานซิสเตอร์ ที่ทำหน้าที่ขับสัญญาณเอาท์พุทนั้นหยุดการทำงาน ก็จะเป็นผลให้ขาสัญญาณนี้อยู่ในสภาวะอิมพีแดนซ์สูงแทน

1.2 การใช้งานพอร์ทเป็นการเอาท์พุท

เมื่อมีการส่งข้อมูลที่มีค่าเป็น 0 ให้กับแต่ละบิทของพอร์ททุกพอร์ท ข้อมูลนี้จะถูกส่งให้กับฟลิปฟลอปซึ่งจะค้างค่านี้ไว้ และมีผลทำให้ทรานซิสเตอร์ที่ทำหน้าที่ขับสัญญาณเอาท์ พุทนั้นทำงาน ดังนั้น ขาสัญญาณก็จะมีสภาวะ ลอจิก เป็นลอจิกต่ำส่วนการส่งข้อมูลที่มีค่าเป็น 1 ออกมานั้น ในกรณีที่เป็นการทำงานในแต่ละบิทของพอร์ท 1,2 หรือ 3 จะทำให้ทรานซิสเตอร์ที่ทำหน้าที่ขับสัญญาณเอาท์พุทนั้นหยุดการทำงาน มีผลทำให้ขาของสัญญาณเป็นลอจิกสูงด้วยตัวต้านทานที่พูลอัพอยู่ภายในนั้น แต่สำหรับการทำงานในแต่ละบิททางพอร์ท 0 นั้น จะมีผลที่แตกต่างออกไป



รูปที่ 2.3 โครงสร้างแต่ละบิทในพอร์ทอินพุท/เอาท์พุทของ 8951

โดยขาสัญญาณจะเป็นสภาวะอิมพีแดนซ์สูงแทนภายในเชื่อมต่ออยู่นั่นเอง ดังนั้น ในการใช้งานพอร์ท 0 เป็นการเอาท์พุทข้อมูล จึงจำเป็นต้องใช้ตัวต้านทานภายนอก พูลอัพสัญญาณไว้กับลอจิกสูงแทน ความสามารถอีกประการหนึ่งเกี่ยวกับพอร์ทอินพุท/เอาท์พุทของ 8951 เป็นวิธีการอ่านลอจิกจากพอร์ท ซึ่งมีได้สองวิธี คือ การอ่านค่าลอจิกที่ขาสัญญาณ(Port pin) และวิธีการอ่านลอจิกของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การแลทช์ที่พอร์ท (Port latch) ดังจะสังเกตได้จากวิธีการอ่านค่าจากพอร์ททั้งสองแบบนี้จะช่วยให้ระบบทำงานได้ด้วยความถูกต้องมากยิ่งขึ้น ยกตัวอย่าง เช่น หากว่าพอร์ทถูกนำไปต่อกับขาเบสของทรานซิสเตอร์แบบ NPN และขาอิมิตเตอร์ต่อกับกราวด์ ของระบบเมื่อมีการส่งค่า 1 ออกไปจะมีผลทำให้ทรานซิสเตอร์ทำงาน ในขณะนั้นถ้าซีพียูมีการอ่านค่าลอจิกจากขา สัญญาณของพอร์ทนี้ก็จะได้ค่าลอจิกต่ำเนื่องจากมองเห็นค่าศักย์ไฟฟ้าระหว่างขาเบส และขาอิมิตเตอร์ ซึ่งมีค่าประมาณ 0.6 โวลท์ แทนดังนั้นในกรณีเช่นนี้หากว่าเป็นการอ่านค่าจากลอจิกของการแลทช์ ก็จะได้รับค่าระดับลอจิกสูงซึ่งเป็นค่าที่ถูกต้องสภาพที่เป็นจริง

1.3 คุณสมบัติของพอร์ทอินพุท/เอาท์พุท

ดังได้กล่าวแล้วว่าพอร์ท 1 , 2 และ 3 ของ 8951 มีตัวต้านทาน (ซึ่งสร้างขึ้นจาก FET) ทำหน้าที่ พูลอัพขาสัญญาณไว้และมีค่าประมาณ 50 kΩ ซึ่งถือว่ามีค่าที่สูงมากเป็นผลให้การเปลี่ยนแปลงระดับสัญญาณลอจิก จากสูงไปต่ำทำได้อย่างลวดเร็ว แต่ในกรณีตรงข้ามจะให้เวลาการเปลี่ยนแปลงระดับสัญญาณนานกว่ามาก ทั้งนี้เนื่องจากว่า กระแสจะไหลผ่านตัวต้านทานนี้ได้น้อยมาก ดังนั้นในการแก้ปัญหาจึงได้มีการออกแบบตัวต้านทานเพิ่มขึ้นอีกหนึ่งตัวขนานไว้โดยมีค่าประมาณ 1 kΩ เรียกว่า ความต้านทาน สปีดอัพ (Speed - up resistor) ซึ่งยอมให้กระแสไหลผ่านได้มากขึ้นประมาณ 50 - 100 เท่า และจะมีการเชื่อมต่อตัวต้านทานที่เพิ่มขึ้นนี้เฉพาะเมื่อมีการเปลี่ยนระดับสัญญาณจากลอจิกต่ำไปเป็นลอจิกสูงเท่านั้น โดยใช้เวลาประมาณ 2 คล็อกไซเคิล 4.6 คำสั่ง การใช้งานพอร์ทอินพุท/เอาท์พุท เนื่องจาก 8951 ใช้หลักการที่เรียกว่า“กำหนดหน่วยความจำ (Memory mapped system)” กล่าวคือ การอ้างถึงพอร์ท รีจิสเตอร์ หรืออุปกรณ์ต่าง ๆ ภายในระบบจะเป็นการติดต่อกับหน่วยความจำตำแหน่งหนึ่งเท่านั้น ดังนั้น ในการดำเนินการเพื่อนำเข้าหรือส่งออกข้อมูลกับพอร์ท จึงใช้คำสั่งการอ่านค่าจากหน่วยความจำซึ่งถูกออกแบบให้เป็นตำแหน่งของพอร์ท หรือ คำสั่งการเขียนค่าข้อมูลไปยังตำแหน่งหน่วยความจำ ดังนั้น จะสังเกตเห็นได้ว่าในตารางชุดคำสั่งของ 8951 จะไม่มีคำสั่งที่เกี่ยวกับการทำงานพอร์ทแต่ประการใด เช่น คำสั่ง IN (นำเข้าข้อมูลจากพอร์ท) หรือ คำสั่ง OUT (ส่งข้อมูลออกจากพอร์ท) เป็นต้น นอกจากนี้ 8951 ยังมีชุดคำสั่งที่จัดการข้อมูลแบบบิทได้โดยตรง (Single-bit Operation) ดังนั้น เราสามารถที่จะใช้คำสั่ง เซตบิท (SETB) เพื่อกำหนดค่าเป็น 1 หรือคำสั่ง เคลียร์(CLR) เพื่อทำให้บิทมีค่าเป็น 0 คำสั่งเหล่านี้มีประโยชน์มากและทำให้ลดความซับซ้อนในการใช้คำสั่งภายในโปรแกรมลงได้มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42285

2. พอร์ทสื่อสารข้อมูลแบบอนุกรมใน MCS – 51

MCS– 51 มีพอร์ทสำหรับสื่อสารข้อมูลแบบอนุกรมที่สามารถรับและส่งข้อมูลแบบอนุกรมได้ โดยผู้ใช้ไม่จำเป็นต้องต่อชิพที่ทำหน้าที่รับส่งข้อมูลแบบอนุกรมโดยเฉพาะเพิ่มแต่อย่างใดเลย การนำ MCS – 51 ไปประยุกต์ใช้งานที่ต้องมีการติดต่อสื่อสารข้อมูลแบบอนุกรมกับวงจรภายนอกอื่น ๆ จึงทำได้สะดวกและมีความคล่องตัวสูงมาก

พอร์ทสื่อสารข้อมูลแบบอนุกรมที่มีใน MCS–51 สามารถทำงานได้ในแบบ full duplex หมายความว่า MCS–51 สามารถรับและส่งข้อมูลได้พร้อม ๆ กัน โดยในการรับข้อมูลจะมีการบัฟเฟอร์ข้อมูลให้ด้วย จึงทำให้ MCS–51 สามารถกำหนดการรับข้อมูลไบท์ที่สองซึ่งถูกส่งตามเข้ามาก่อนที่ไบท์แรกที่ได้รับเข้ามาจะถูกอ่านจากรีจิสเตอร์ใช้งานเฉพาะที่ใช้สำหรับข้อมูล(Receive register) เพื่อนำไปเก็บไว้ในหน่วยความจำต่อไป หากไบท์แรกยังไม่ถูกอ่านเมื่อได้รับไบท์ที่สองเรียบร้อยแล้วข้อมูลจะหายไปหนึ่งไบท์

พอร์ทสื่อสารข้อมูลแบบอนุกรมใน MCS–51 ประกอบด้วยรีจิสเตอร์ขนาด 8 บิทจำนวนสองตัวแต่ละตัวมีชื่อเรียกตามหน้าที่ดังนี้คือ

- รีจิสเตอร์สำหรับรับข้อมูลใช้รับข้อมูลที่ส่งเข้ามาจากภายนอก
- รีจิสเตอร์สำหรับส่งข้อมูล (Transmit register) ใช้ส่งข้อมูลจาก MCS–51 ออกไปภายนอก

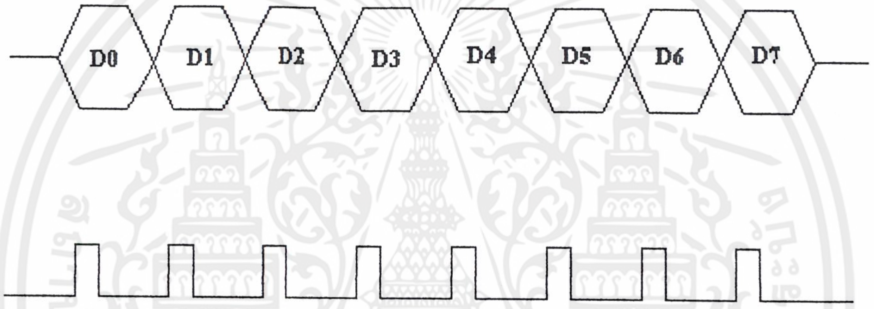
รีจิสเตอร์ทั้งสองมีตำแหน่งเดียวกันในรีจิสเตอร์ใช้งานเฉพาะ คือ ตรงกับตำแหน่งของรีจิสเตอร์ใช้งานเฉพาะ SBUF (ตำแหน่ง 99H) ในหน่วยความจำสำหรับเก็บข้อมูลภายในชิพที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะ การเข้าถึงข้อมูลในรีจิสเตอร์แต่ละตัว MCS–51 จะทราบเองว่าผู้ใช้ต้องการติดต่อกับรีจิสเตอร์ตัวในโดยตรวจสอบจากรหัสคำสั่ง ทั้งนี้เพราะการเขียนข้อมูลไปไว้ในรีจิสเตอร์ใช้งานเฉพาะ SBUF หมายถึงการโหลดข้อมูลไปที่รีจิสเตอร์สำหรับส่งข้อมูลเพื่อส่งข้อมูลออกไปภายนอก ส่วนการอ่านข้อมูลจากรีจิสเตอร์ใช้งาเฉพาะ SBUF จะหมายถึงนำค่าที่รับเข้ามาได้จากภายนอกที่เก็บไว้ในรีจิสเตอร์สำหรับรับข้อมูลมาใช้งาน

2.1 โหมดการรับส่งข้อมูลอนุกรม

การใช้งานพอร์ทสื่อสารข้อมูลแบบอนุกรมใน MCS–51 มีความสะดวกและคล่องตัวสูง ทั้งนี้เนื่องจากผู้ใช้สามารถกำหนดการทำงาที่แตกต่างกันได้ถึง 4 ประเภท โดยสามารถกำหนดได้จากค่าของบิทในรีจิสเตอร์ใช้งานเฉพาะ SCON ดังแสดงในรูปที่ 2.4 การใช้งานที่แตกต่างกัน 4 ประเภทนี้มีจุดประสงค์เพื่อความคล่องตัวในการรับหรือส่งข้อมูลแบบอนุกรมแต่ละประเภท ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

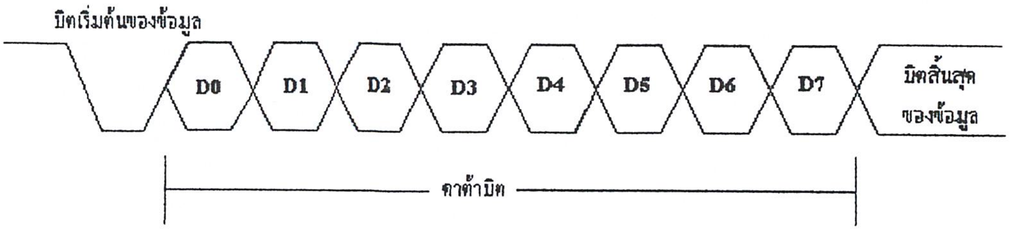
โหมด 0 การทำงานของพอร์ทสื่อสารข้อมูลแบบอนุกรมในโหมด 0 ขา RXD จะใช้สำหรับรับและส่งข้อมูล ส่วนขา TXD มีไว้เพื่อใช้สร้างสัญญาณ Shift clock เพื่อกำหนดจังหวะในการรับและส่งข้อมูล (ข้อมูลจะถูกรับหรือส่งตามจังหวะของสัญญาณ Shift clock) ในโหมดนี้การรับส่งข้อมูลจะเป็นแบบ 8 บิท (บิทข้อมูล 8 บิท) โดยเริ่มรับและส่งบิทต่ำที่สุดก่อน (LSB first) อัตราการรับส่งข้อมูลในการทำงานโหมด 0 ถูกกำหนดไว้ที่ 1/12 ของความถี่ออสซิลเลเตอร์ที่ใช้การทำงานของพอร์ทสื่อสารข้อมูลแบบอนุกรมในโหมด 0 จะไม่มีบิทเริ่มต้นของข้อมูล (Start bit) และบิทสิ้นสุดของข้อมูล (Stop bit) เพราะจังหวะการรับและส่งข้อมูลถูกกำหนดจากสัญญา Shift clock แล้ว



รูปที่ 2.4 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ทสื่อสารข้อมูลแบบอนุกรมโหมด 0

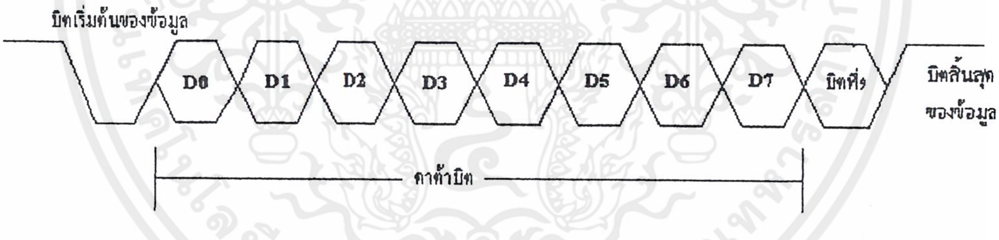
โหมด 1 การทำงานแบบที่สองหรือการทำงานโหมด 1 นี้มีการรับและส่งข้อมูลครั้งละ 10 บิท ข้อมูลจะถูกส่งออกไปภายนอกผ่านทางขา TXD และรับข้อมูลเข้ามาทางขา RXD ข้อมูลทั้ง 10 บิท ประกอบด้วยบิทเริ่มต้นของข้อมูล 1 บิท (มีค่าเป็น 0 เสมอ) บิทข้อมูล 8 บิท (รับและส่งบิทต่ำสุดก่อน) และบิทสิ้นสุดของข้อมูลอีก 1 บิท (มีค่าเป็น 1 เสมอ) ในขณะทำการรับข้อมูลค่าในบิทสิ้นสุดของข้อมูลที่รับได้จะอยู่ในบิท RB8 ของรีจิสเตอร์ใช้งานเฉพาะ SCON อัตราเร็วในการรับหรือส่งข้อมูลของพอร์ทสื่อสารข้อมูลแบบอนุกรม ในโหมดนี้สามารถเปลี่ยนแปลงได้ดังจะได้กล่าวในรายละเอียดต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ทสื่อสารข้อมูลแบบอนุกรมโหมด 1

โหมด 2 การทำงานแบบที่ 3 หรือการทำงานโหมด 2 จะมีการรับและส่งข้อมูลครั้งละ 11 บิท ข้อมูลจะถูกส่งออกภายนอกผ่านขา TXD และรับเข้ามาผ่านทางขา RXD ข้อมูลที่รับและส่งทั้ง 11 บิท ประกอบด้วยบิทเริ่มต้นของข้อมูล 1 บิท (มีค่าเป็น 0 เสมอ) บิทข้อมูล 8 บิท (รับหรือส่งบิทต่ำสุดก่อน) ตามด้วยบิทที่ 9 ซึ่งเป็นบิทที่สามารถกำหนดให้มีค่าเป็น 0 หรือ 1 ก็ได้ และบิทสุดท้ายคือบิทสิ้นสุดของข้อมูล (มีค่าเป็น 1 เสมอ) ดังนั้นจำนวนบิทที่รับส่งทั้งหมด 11 บิทจะประกอบด้วยบิทต่าง ๆ ดังนี้



รูปที่ 2.6 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ทสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3

ในขณะทำการส่งข้อมูลบิทที่ 9 จะได้จากค่าในบิท TB8 ของรีจิสเตอร์ใช้งานเฉพาะ SCON ดังแสดงในรูปที่ 2.5 บิทนี้สามารถถูกกำหนดให้มีค่าเป็น 0 หรือ 1 อย่างไรก็ได้ ส่วนใหญ่ในการใช้งานจริง มักจะใช้บิทนี้สำหรับตรวจสอบความถูกต้องของข้อมูลที่รับหรือส่ง (Parity bit) โดยจะนำบิท P (Parity) ในรีจิสเตอร์ PSW ไปไว้ในบิท TB8 ส่วนในขณะรับข้อมูลบิทที่ 9 จะไม่ปรากฏอยู่ในบิท RB8 ของรีจิสเตอร์ SCON โดยไม่สนใจบิทสิ้นสุดของข้อมูล ค่าอัตราเร็วรับและส่งข้อมูลโหมดนี้ถูกกำหนดไว้ที่ 1/32 หรือ 1/64 ของความถี่ของออสซิลเลเตอร์ที่ใช้

โหมด 3 การทำงานของพอร์ทสื่อสารข้อมูลแบบอนุกรมแบบสุดท้าย คือการทำงานในโหมด 3 ในการทำงานโหมดนี้ข้อมูลจำนวน 11 บิทถูกส่งผ่านขา TXD และถูกรับเข้ามาทางขา RXD ข้อมูลทั้ง 11 บิท ประกอบด้วย บิทเริ่มต้นของข้อมูล 1 บิท (เป็น 0 เสมอ) บิทข้อมูล 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิท (รับและส่งบิทต่ำสุดก่อน) ตามด้วยบิทที่ 9 ซึ่งเป็นบิทที่สามารถกำหนดค่าได้เหมือนในโหมด 2 และบิทสุดท้ายค่าบิทสิ้นสุดของข้อมูล (เป็น 1 เสมอ) อัตราเร็วในการรับหรือส่งข้อมูลสามารถเปลี่ยนแปลงได้ดังนั้นจะเห็นว่ารูปแบบในการรับส่งข้อมูลในโหมด 3 จะเหมือนกับโหมด 2 ทุกอย่างแต่ในโหมดนี้สามารถกำหนดค่าอัตรเร็วในการรับและส่งข้อมูลได้ตามความต้องการของผู้ใช้

2.2 อัตราเร็วในการรับและส่งข้อมูล

Baud rate หมายถึง อัตราเร็วรับหรือส่งข้อมูล โดยใน MCS – 51 ค่าอัตราเร็วในการรับและส่งข้อมูลจะมีค่าเท่าใดขึ้นอยู่กับการทำงานในแต่ละโหมดของพอร์ทสื่อสารข้อมูลแบบอนุกรมดังนี้

$$\textbf{Baud rate โหมด 0} = \frac{\textbf{ความถี่ออสซิลเลเตอร์ที่ใช้}}{\textbf{12}}$$

หากใช้คริสตอลความถี่ 12 เมกะเฮิรตซ์ ค่า Baud rate ของพอร์ทสื่อสารอนุกรมในโหมด 0 จะมีค่าสูงถึง 1 เมกะเฮิรตซ์

ในโหมด 2 ค่า Baud rate ขึ้นอยู่กับค่าของบิท SMOD ที่อยู่ในรีจิสเตอร์ที่ใช้งานเฉพาะ PCON โดย

- บิท SMOD = 0 ค่า baud rate จะเป็น 1/64 ของความถี่ออสซิลเลเตอร์ที่ใช้
- บิท SMOD = 1 ค่า baud rate จะเป็น 1/32 ของความถี่ออสซิลเลเตอร์ที่ใช้

หลังจากการรีเซตMCS–51 ค่าในบิท SMOD จะเป็น 0 เสมอ และเราสามารถเขียนสูตรสำหรับคำนวณค่า Baud rate ได้ดังสมการนี้

$$\textbf{Baud rate โหมด 2} = \frac{[2^{(SMOD)} \times (\textbf{ความถี่ออสซิลเลเตอร์ที่ใช้})]}{\textbf{64}}$$

หากใช้คริสตอลความถี่ 12 เมกะเฮิรตซ์ Baud rate สูงสุดในการทำงานในโหมดนี้คือ 375 k

Baud rate ในโหมด 1 และ 3 จะถูกกำหนดโดยอัตราการเกิด Overflow ของไทม์เมอร์ 1 แต่ถ้าเป็น 8052 ซึ่งมีรีจิสเตอร์สำหรับใช้เป็นไทม์เมอร์หรือเคาน์เตอร์เพิ่มให้อีก 1 ตัว จะสามารถใช้ไทม์เมอร์ 2 ที่มีเพิ่มมานี้เป็นตัวกำหนด baud rate ได้ ทำให้มีรีจิสเตอร์สำหรับการใช้งานเป็นไทม์เมอร์ หรือ เคาน์เตอร์ ที่สามารถนำมากำหนด baud rate รวมจำนวน 2 ตัว (ไทม์เมอร์ 1 และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทม์เมอร์ 2) โดยอาจใช้ตัวใดตัวหนึ่งในการกำหนด Baud rate สำหรับการรับข้อมูล ส่วนอีกตัวหนึ่งกำหนด Baud rate สำหรับการส่งข้อมูลทำให้การรับและส่งข้อมูลมีค่า Baud rate ที่ต่างกันได้

เมื่อใช้ไทม์เมอร์ 1 เป็นตัวกำหนด Baud rate สำหรับการทำงานของพอร์ทสื่อสารข้อมูลแบบอนุกรมในโหมด 1 และ 3 ค่าของ Baud rate ที่ได้ถูกกำหนดด้วยอัตราการเกิด Overflow ของไทม์เมอร์ 1 และขึ้นอยู่กับบิท SMOD ในรีจิสเตอร์ PCON ซึ่งเขียนเป็นสมการที่ใช้คำนวณหา Baud rate ได้ดังนี้

$$\text{Baud rate โหมด 1,3} = \frac{2^{(SMOD)} \times \text{อัตราการเกิด overflow ของไทม์เมอร์ 1}}{32}$$

เนื่องจากเมื่อเกิด Overflow ในไทม์เมอร์ตัวใดจะทำให้เกิดสัญญาณอินเตอร์รัปต์เพื่อบอกให้ CPU ทราบ ดังนั้น เมื่อเรานำไทม์เมอร์ 1 มาเป็นตัวกำหนด baud rate จึงควรห้ามการเกิดอินเตอร์รัปต์ ขึ้นในระหว่างการรับหรือส่งข้อมูล และเนื่องจากตัวไทม์เมอร์ 1 เองยังสามารถถูกกำหนดให้สามารถทำงานเป็นไทม์เมอร์หรือเคาน์เตอร์ อย่างใดอย่างหนึ่ง ซึ่งมีโหมดการทำงานย่อย

ตารางที่ 2.6 อัตราการกำเนิด baud rate ค่ามาตรฐานต่าง ๆ

Baud rate	**ความถี่ของคริสตอล**	**บิท SMOD**	ไทม์เมอร์ 1		
			C/T	**โหมด**	**ค่าที่ใช้โหลด**
Mode 0 Max: 1 MHz	12 MHz	X	X	X	X
Mode 2 Max: 375KHz	12 MHz	1	X	X	X
Mode 1,3: 62.5KHz	12 MHz	1	0	2	FFH
19.2KHz	11.059 MHz	1	0	2	FDH
9.6KHz	11.059 MHz	0	0	2	FDH
4.8KHz	11.059 MHz	0	0	2	FAH
2.4KHz	11.059 MHz	0	0	2	F4H
1.2KHz	11.059 MHz	0	0	2	E8H
137.5KHz	11.059 MHz	0	0	2	1DH
110KHz	6 MHz	0	0	2	72H
110KHz	12 MHz	0	0	1	FEEBH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลงไปอีก 4 โหมดดังกล่าวมาแล้ว ดังนั้น ในการใช้งานพอร์ทสื่อสารข้อมูลแบบอนุกรมจึงต้องทำความเข้าใจในเรื่องไทม์เมอร์หรือเคาน์เตอร์ให้ละเอียด

การใช้งานพอร์ทสื่อสารข้อมูลแบบอนุกรมที่พบบ่อยที่สุดนั้น ไทม์เมอร์ 1 จะถูกกำหนดให้ทำงานเป็นไทม์เมอร์ในโหมด 2 (Auto – Reload) ในกรณีนี้ Baud rate จะถูกกำหนดโดยสมการ ดังนี้

$$\textbf{Baud rate โหมด 1,3} = \frac{2^{SMOD} \times \text{ความถี่ออสซิลเลเตอร์ที่ใช้}}{32 \times 12 \times (265 - TH1)}$$

ดังนั้นค่าที่ต้องโหลดไปไว้ยังรีจิสเตอร์ TH1 จะสามารถคำนวณได้จากสมการข้างบนและค่า baud rate ที่ใช้กันมาก แสดงในตารางที่ 2.6

2.2 มาตรฐานพอร์ทอนุกรมแบบ RS–232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS–232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS–232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านทางสายโทรศัพท์ ไปยังคอมพิวเตอร์อีกชุดหนึ่ง ซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association: EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS–232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB–25 โดยกำหนดความยาวสูงสุดของสายสัญญาที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ –3 ถึง –12 V แสดงว่ามีข้อมูล (MARK) และ +3 ถึง +12 V แสดงว่าเป็นช่องว่าง (Space)

มาตรฐาน RS–232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment: DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating Equipment: DCE) ไว้ว่าอุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัว เช่น ไมโครคอนโทรลเลอร์ หรือ ไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างบิทข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้ง 2 จะกระทำผ่านมาตรฐาน RS–232

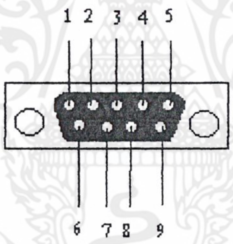
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่เห็นได้ชัดคือ คอนเน็กเตอร์ ของ DTE จะเป็นตัวผู้ ส่วนของ DCE จะเป็นตัวเมีย ซึ่งพอร์ทอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

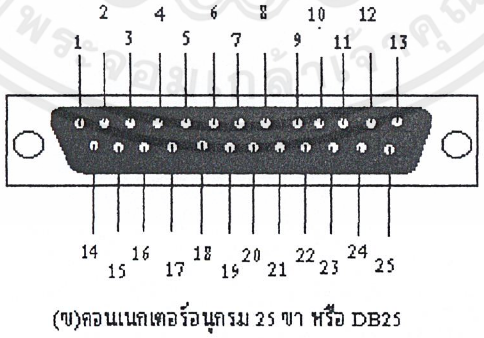
สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ทอนุกรม RS–232 มักถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์ โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20 เมตร

2.2.1 คอนเน็กเตอร์สำหรับพอร์ท RS – 232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS–232 จะใช้คอนเน็กเตอร์แบบ DB–25 ตัวผู้หรือ DB–9 ตัวผู้ซึ่งคอนเน็กเตอร์แบบ DB–25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็กเตอร์แบบ DB–9 เนื่องจากขาอื่นๆที่เคยใช้ในอดีตปัจจุบันไม่มีการใช้งานจึงถูกยกเลิกไปโดยแสดงรูปร่างแลตำแหน่งขาในรูปที่ 2.7



(ก)คอนเนคเตอร์อนุกรม 9 ขา หรือ DB9



(ข)คอนเนคเตอร์อนุกรม 25 ขา หรือ DB25

รูปที่ 2.7 แสดงลักษณะของคอนเน็กเตอร์ 9 ขา (ก) และ 25 ขา(ข)

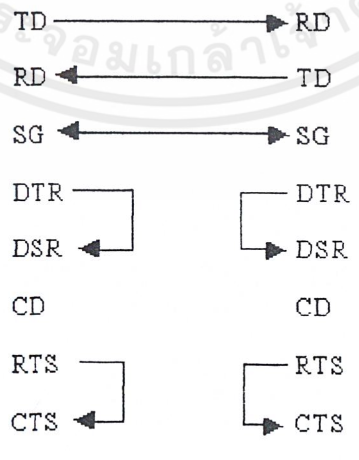
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.7 ความหมายของขา คอนเน็กเตอร์พอร์ทอนุกรม ตามมาตรฐาน RS – 232

คอนเน็กเตอร์ DB – 9	คอนเน็กเตอร์ DB –25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	Data Carrier Detect: DCD	อินพุท
2	3	Received Data: RxD	อินพุท
3	2	Transmitted Data: TxD	เอาท์พุท
4	20	Data Terminal Ready: DTR	เอาท์พุท
5	7	Signal Ground: GND	-
6	6	Data Set Ready: DSR	อินพุท
7	4	Request To Send: RTS	เอาท์พุท
8	5	Clear To Send: CTS	อินพุท
9	22	Ring Indicator: RI	อินพุท

2.2.2 นัลโมเด็ม (Null Modem)

นัลโมเด็ม ใช้เพื่อต่อเชื่อม DTE เข้าด้วยกัน มีลักษณะการต่อสายดังรูปที่ 2.8 ลักษณะการต่อสายของนัลโมเด็มนั้น ใช้เพียงสาย 3 เส้น (TD RD และ SG) โดยที่สาย TD ของคอมพิวเตอร์เครื่องหนึ่งจะต่อไปยังสาย RD ของคอมพิวเตอร์อีกเครื่องหนึ่งในทางกลับกันสาย RD



รูปที่ 2.8 แสดงลักษณะการเชื่อมต่อ DTE แบบ นัลโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของคอมพิวเตอร์เครื่องหนึ่ง จะต่อไปยังสาย TD ของคอมพิวเตอร์อีกเครื่องหนึ่ง ส่วนสาย SG ของคอมพิวเตอร์ทั้งสองเครื่องจะต่อเข้าด้วยกัน เนื่องจากเมื่อสัญญาณ DTR แอกทีฟจะทำให้ DSR และ CD แอกทีฟตามทันที ทำให้คอมพิวเตอร์เข้าใจว่าโมเด็มพร้อมทำงานแล้ว ส่วนการต่อสาย RTS ต่อกับ CTS ของคอมพิวเตอร์ทั้งสองเครื่องนั้นเพื่อต้องการให้เครื่องคอมพิวเตอร์นั้นติดต่อสื่อสารกันด้วยอัตราเร็วที่เท่ากัน

2.2.3 RS – 232 Level converters

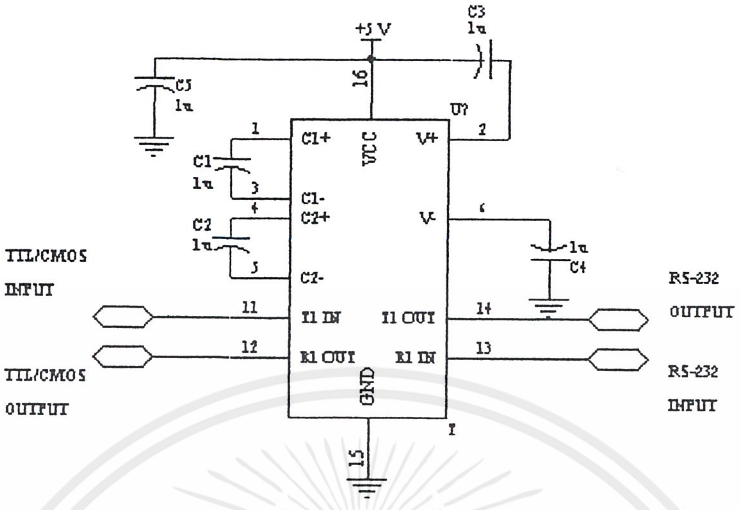
RS 232 Waveform การสื่อสารโดยใช้ RS 232 เป็นการสื่อสารแบบอะซิงโครนัส (Asynchronous) นั้นคือ ในการสื่อสารไม่จำเป็นต้องส่งสัญญาณนาฬิกาไปกับข้อมูลซึ่งแต่ละเวิร์ดของข้อมูลจะใช้บิทเริ่มต้น (Start bit) และบิทสิ้นสุด (Stop bit)

รูปที่ 2.9 สัญญาณที่รับและส่ง ผ่านพอร์ท RS 232 เทียบกับ TTL / CMOS

RS 232 Level Converters อุปกรณ์โดยส่วยใหญ่จะต้องการระดับแรงดัน TTL และ CMOS ดังนั้นขั้นตอนแรกในการต่ออุปกรณ์เหล่านี้เข้ากับ RS 232 จะต้องแปลงระดับแรงดันของ RS 232 เป็นระดับแรงดันที่สามารถที่ใช้กัน TTL ได้ (5 V) ซึ่งสามารถทำได้โดยใช้ RS 232 Level Converters

แสดงลักษณะขาและวงจรภายในของชิพ MAX 232 ซึ่งจะสร้างระดับแรงดัน +10 V และ –10 V จากแรงจ่าย 5 V ซึ่งในชิพนี้จะประกอบด้วยตัวรับและตัวส่ง 2 ชุด ดังแสดงในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

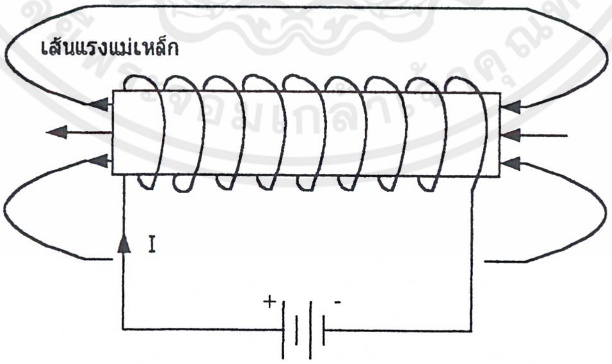


รูปที่ 2.10 ลักษณะของ RS 232 Level Converters

2.3 การเหนี่ยวนำไฟฟ้า

2.3.1 สนามแม่เหล็กของขดลวดโซลีนอยด์

เมื่อต่อปลายทั้งสองของขดลวดโซลีนอยด์ (Solenoid: ตัวนำเส้นตรงที่นำมาพันรอบแท่งวัสดุกลมหลาย ๆ รอบทำให้เป็นขด) เข้ากับแหล่งต้นกำเนิดไฟฟ้ากระแสตรงขณะที่ให้กระแสไฟนำ



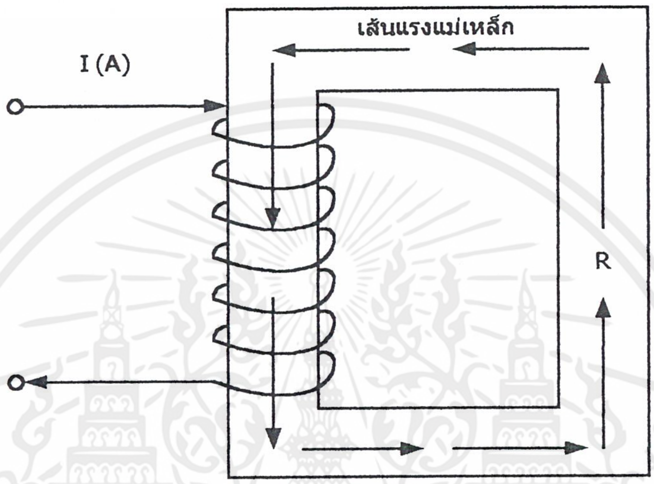
รูปที่ 2.11 สนามแม่เหล็กของขดลวดโซลีนอยด์

ไหลผ่านขดลวด จะให้กำเนิดเส้นแรงแม่เหล็กล้อมรอบขดลวดในทิศทางที่เป็นไปตามกฎตัวแม่มือขวา ดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 วงจรแม่เหล็ก

เมื่อต่อขดลวดที่พันอยู่บนแกนเข้ากับต้นกำเนิดพลังงานไฟฟ้า กระแสที่ไหลในขดลวดจะให้กำเนิดเส้นแรงแม่เหล็กในแกนตามทิศทางที่หาได้ตามกฎหัวแม่มือขวา และคำนวณได้ตามกฎของโอห์มเช่นเดียวกับวงจรไฟฟ้า



รูปที่ 2.12 วงจรแม่เหล็กที่ประกอบด้วยขดลวดพันบนแกนเมื่อให้กระแสไฟฟ้าไหลผ่านในขดลวดจะให้กำเนิดเส้นแรงแม่เหล็กไหลในแกนหลัก

2.3.3 การแบ่งชนิดของสาร

สารทุกชนิดในโลกนี้ ถ้าไม่เป็นสารแม่เหล็กก็เป็นสารที่ไม่เป็นแม่เหล็ก การแบ่งชนิดของสารนี้เราดูได้จากปรากฏการณ์ที่เห็นได้ด้วยตาว่า สารนั้นแสดงคุณสมบัติของแม่เหล็กออกมาหรือไม่ เมื่อวางสารนั้นไว้ในสนามแม่เหล็กที่แรง อย่างไรก็ดีในบางครั้งก็จำเป็นที่จะต้องพิจารณาคุณสมบัติของแม่เหล็กบางประการที่ไม่ปรากฏด้วยเหตุผลอันนี้เอง การแบ่งชนิดให้เฉพาะเจาะจงมากขึ้นจึงมีความจำเป็นอย่างยิ่ง ชนิดของสารแบ่งได้ดังนี้

1. สารไดอะแมกเนติค (Diamagnetic material) เป็นสารที่ถูกผลักด้วยแม่เหล็กด้วยแรงน้อย ๆ อะตอมของสารพวกนี้โมเลกุลแม่เหล็กจะไม่จัดเรียงตัวกัน แต่ถ้านำสารพวกนี้ไปวางในสนามแม่เหล็กที่แรงมาก ๆ อะตอมบางอะตอมจะจัดเรียงตัวกันกลายเป็นแม่เหล็กที่มีอำนาจแม่เหล็กอ่อน ๆ และขั้วแม่เหล็กที่เกิดขึ้นจะผลักกับแท่งแม่เหล็ก สารไดอะแมกเนติคที่พบเห็นกันทั่ว ๆ ไปได้แก่ อันติโมนี (antimony) ทองแดง ทอง ปรอท เงิน และสังกะสี

2. สารพาราแมกเนติค (Paramagnetic materials) สารชนิดนี้จะถูกดูดด้วยแม่เหล็กด้วยแรงอ่อน ๆ เนื่องจากอะตอมบางอะตอมของสารจะจัดเรียงตัวกันเป็นโมเลกุลแม่เหล็กตามธรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชาติ เมื่อนำสารนี้ไปวางไว้ในสนามแม่เหล็กที่แรง โมเลกุลแม่เหล็กก็จะจัดเรียงตัวกัน ขั้วแม่เหล็กที่เกิดขึ้นจะถูกดูดด้วยแม่เหล็ก สารพาราแมกเนติกที่พบเห็นกันทั่ว ๆ ไปได้แก่ อลูมิเนียม โครเมี่ยม (chromium) แมงกานีส (manganese) และพลาตินัม (platinum)

3. สารเฟอร์โรแมกเนติค (Ferromagnetic materials) สารชนิดนี้จะถูกแม่เหล็กดูดอย่างแรง และจัดได้ว่าเป็นสารแม่เหล็กที่มีความสำคัญและนำไปใช้ประโยชน์มากในวงการไฟฟ้าและอิเลคทรอนิคส์ แบ่งได้ดังนี้

- ธาตุเฟอร์โรแมกเนติค ได้แก่ธาตุโคบอลท์(Cobalt) เหล็ก (iron) และนิคเกิ้ล (Nickel) ธาตุเหล่านี้เมื่อวางไว้ในสนามแม่เหล็ก (จากภายนอก) โมเลกุลแม่เหล็กจะจัดเรียงตัวกันอย่างเป็นระเบียบ ภายในเนื้อสารจะมีแรงเนื่องจากไฟฟ้าสถิตย์ที่พยายามที่จะให้โมเลกุลแม่เหล็กยังคงเรียงตัวกันไปในทิศทางเดียวกันอยู่เสมอ หลังจากที่สารนั้นถูกทำให้เป็นแม่เหล็กแล้ว ถ้าหากไม่มีแรงเนื่องจากไฟฟ้าสถิตในเนื้อสารนี้แล้ว ปริมาณความร้อนแม้แต่เพียงขนาดที่อุณหภูมิห้อง (27 องศาเซลเซียส) ก็จะมากพอที่จะทำให้โมเลกุลของแม่เหล็กไม่เรียงตัวไปในทิศทางเดียวกันได้ (disalignment)

- เฟอร์โรแมกเนติคอัลลอย (Ferromagnetic Alloys) แม่เหล็กที่เป็นแม่เหล็กถาวรคุณภาพสูงซึ่งมักให้กันมากในงานเกี่ยวกับอุตสาหกรรม และประยุกต์ใช้ในงานพาณิชย์จะทำด้วยพวกเฟอร์โรแมกเนติค อัลลอยเกือบทั้งสิ้น ที่รู้จักกันดีก็ได้แก่ อัลนิโค(Alnico) ซึ่งเป็นอัลลอยของอลูมิเนียม นิเกิ้ล โคบอลท์ และเหล็ก แม่เหล็กถาวรที่ทำเป็นรูปร่างและขนาดต่าง ๆ กันที่ทำจากอัลนิโค นำไปใช้ในอุปกรณ์ชนิดต่าง ๆ เช่น ลำโพง มอเตอร์ แมกนิโต (magneto)

- เฟอร์ไรท์ (Ferrites) เฟอร์ไรท์เป็นสารแม่เหล็กที่โครงสร้างมีลักษณะคล้ายโครงสร้างของพวกเซรามิค (ceramic) สารพวกนี้ประกอบด้วยธาตุโลหะกับอ๊อกไซด์ของโลหะ เมื่อนำทั้งสองอย่างมาผสมกันแล้วอัดส่วนผสมลงไปในแบบที่มีรูปร่างตามที่ต้องการ แล้วอบด้วยความร้อนอุณหภูมิสูง ๆ

เฟอร์ไรท์แบ่งออกเป็น 2 ชนิด คือ ชนิดอ่อน (Soft) และแข็ง (hard) เฟอร์ไรท์ชนิดอ่อน (soft ferrites) ทำจากสารพวกนิเกิ้ล – สังกะสี หรือ มังกานีส – สังกะสี ซึ่งจะเติมพวกอ๊อกไซด์ของแมกนีเซียม และโคบอลท์ลงไปด้วย ชนิดนี้มักใช้ทำเป็นแกนของขดลวดชนิดต่าง ๆ

เฟอร์ไรท์ชนิดแข็ง (Hard ferrites) ส่วนมากประกอบด้วยสารประกอบของแบเรี่ยม (barium compound) เป็นพื้นแล้วเติมพวกอ๊อกไซด์ของตะกั่ว อ๊อกไซด์ของทองแดงและอ๊อกไซด์ของบิสมัทลงไปด้วย เฟอร์ไรท์พวกนี้มักใช้ทำเป็นแม่เหล็กถาวร เช่น ที่ใช้ในลำโพง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

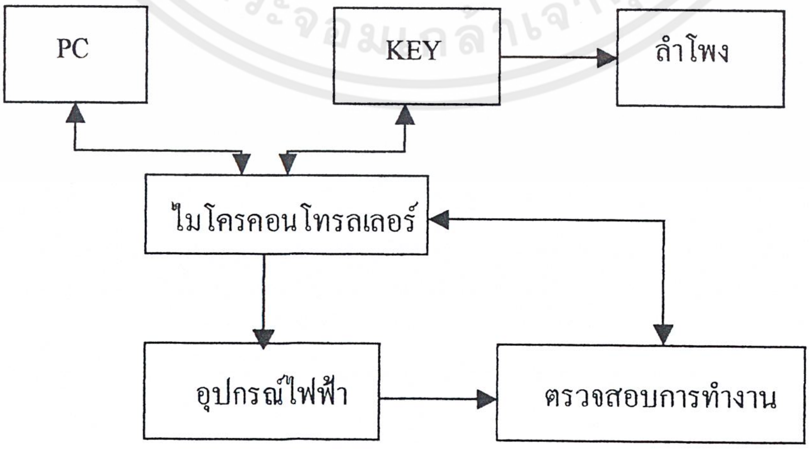
บทที่ 3
แนวคิดและการออกแบบ

เนื่องจากในปัจจุบันคอมพิวเตอร์เข้ามามีบทบาทกับชีวิตประจำวันมาก และเริ่มมีความจำเป็นกับการดำรงชีวิตมากขึ้น ในขณะที่ราคาของคอมพิวเตอร์ถูกลงแต่ได้คุณภาพที่สูงขึ้น คอมพิวเตอร์จึงน่าจะกลายมาเป็นเครื่องใช้ประจำบ้าน เพื่ออำนวยความสะดวกสบายในการดำเนินชีวิต จึงเป็นจุดเริ่มต้นของโครงการนี้

การควบคุมอุปกรณ์ไฟฟ้าด้วยคอมพิวเตอร์จะเพิ่มความสะดวกมากขึ้น โดยเราสามารถนั่งอยู่ที่คอมพิวเตอร์ หรือจุดใดจุดหนึ่งในบ้านควบคุมการปิดเปิดอุปกรณ์ไฟฟ้าทุกอย่างภายในบ้านได้ พร้อมทั้งสามารถที่จะตรวจสอบได้ว่าอุปกรณ์หรือเครื่องใช้ตัวใดทำงานหรือไม่ทำงาน และมีการแสดงผลที่จอคอมพิวเตอร์หรือเสียงจากลำโพงได้ ดังนั้นจึงเกิดแนวคิดในการทำโครงงานนี้ขึ้นและเพื่อความสะดวกในการออกแบบจะแบ่งการทำงานออกเป็น 3 ส่วนใหญ่ คือ

- ชุดควบคุมอุปกรณ์หลัก คือ ไมโครคอนโทรลเลอร์ 8951
- ชุดการควบคุมและการแสดงผล คือ คอมพิวเตอร์ และ รีโมท
- ชุดตรวจสอบการทำงานและรวมส่วนที่เกี่ยวข้องกับไฟสลับด้วย

โดยสามารถแสดงแนวทางการทำงานดังรูปที่ 3.1



รูปที่ 3.1 แนวทางการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 ชุดควบคุม

ชุดควบคุมหรือหน่วยประมวลผล มีหน้าที่ตัดสินใจ และควบคุมการทำงานของอุปกรณ์ หรือรับคำสั่งจากคอมพิวเตอร์และคีย์ โดยการทำงานของหน่วยประมวลผลจะมีส่วนเกี่ยวข้องกับการรับและส่งสัญญาณ เพื่อให้เกิดประสิทธิภาพการทำงาน ดังนั้นจึงมีรายละเอียดดังนี้

1. สามารถรับอินพุทจากแหล่งต่าง ๆ ได้ เช่น คอมพิวเตอร์ คีย์ หรือชุดควบคุม
2. สามารถเชื่อมต่อกับอุปกรณ์อื่นได้ง่ายและสะดวก
3. ความเร็วในการติดต่อ ต้องใช้เวลาที่น้อย
4. ต้องมีความสามารถเก็บข้อมูล เช่นโปรแกรมเพื่อลดต้นทุน เช่น ไม่ต้องต่อหน่วยความจำภายนอก
5. สามารถใช้งานอุปกรณ์ไมโครคอนโทรลเลอร์ได้ง่ายและสะดวก
6. สามารถเปลี่ยนแปลงและแก้ไขข้อมูลได้ง่าย
7. ความเร็วในการทำงานหรือประมวลผล มีความเร็วสูง

3.2 ชุดสั่งการควบคุมและการแสดงผล

ชุดการควบคุมและการแสดงผล มีหน้าที่เชื่อมต่อหรือติดต่อกับผู้ใช้โดยตรงในการออกแบบต้องให้ส่วนนี้สามารถใช้งานได้ง่ายไม่ยุ่งยาก ผู้ใช้ไม่ต้องมีความชำนาญพิเศษก็สามารถใช้ได้อย่างมีประสิทธิภาพ แบ่งออกเป็น 2 ส่วน

3.2.1 การติดต่อกับผู้ใช้ผ่านคอมพิวเตอร์

การติดต่อกับผู้ใช้ผ่านคอมพิวเตอร์จะใช้คอมพิวเตอร์เป็นตัวกลางในการสั่งการควบคุมและแสดงผล คอมพิวเตอร์จะเชื่อมต่อกับชุดควบคุมโดยใช้พอร์ทสื่อสารอนุกรมของคอมพิวเตอร์ จึงจำเป็นต้องมีการพัฒนาโปรแกรมขึ้นมารายละเอียดของโปรแกรมต้องมีความสามารถดังต่อไปนี้ เพื่ออำนวยความสะดวกต่อผู้ใช้

1. ต้องมีการรักษาความปลอดภัยในการควบคุม โดยใส่รหัสผ่านทุกครั้งที่ใช้โปรแกรม เพื่อป้องกันมิให้บุคคลที่ไม่เกี่ยวข้องมาใช้งาน ในกรณีที่นำไปประยุกต์ควบคุมอุปกรณ์ที่สำคัญ
2. รหัสผ่านต้องสามารถแก้ไขได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ต้องพัฒนาโปรแกรมบนระบบปฏิบัติการที่เป็นที่นิยมใช้ เช่น ระบบปฏิบัติการ Windows เพราะจะสะดวกต่อการใช้งาน

4. ต้องสามารถเปลี่ยนแบบ หรือสามารถเคลื่อนย้ายตำแหน่งอุปกรณ์ไฟฟ้าได้ เช่น โปรแกรมแสดงเป็นภาพโครงสร้างของบ้านและภาพอุปกรณ์ไฟฟ้า โดยทั่วไปบ้านของผู้ใช้แต่คนจะไม่เหมือนกัน การควบคุมจะนำไปควบคุมอุปกรณ์ไฟฟ้าที่ต่างกัน และตำแหน่งวางอุปกรณ์ที่ต่างกัน เพื่อความสะดวกตรงส่วนนี้ต้องยืดหยุ่นหรือปรับเปลี่ยนได้

5. ต้องมีสัญลักษณ์แทนอุปกรณ์ไฟฟ้าหลายชนิดที่ครอบคลุมอุปกรณ์ไฟฟ้าที่มีทั้งหมด หรือสัญลักษณ์ต้องสามารถเปลี่ยนแปลงได้ เช่น เมื่อนำไปควบคุมหลอดไฟฟ้าต้องมีสัญลักษณ์ที่มองแล้วทราบว่าเป็นหลอดไฟฟ้า และตัวจะเปลี่ยนไปควบคุมเครื่องปรับอากาศก็ต้องมีสัญลักษณ์ออกเครื่องปรับอากาศ เป็นต้น

6. สัญลักษณ์ที่ใช้แทนอุปกรณ์ไฟฟ้าต้องสามารถแสดงสภาวะ การทำงานงานอุปกรณ์ไฟฟ้านั้นได้ เช่น ควบคุมหลอดไฟฟ้า เมื่อเปิดไฟหลอดไฟเป็นสีเขียว และเมื่อปิดเป็นสีแดง เป็นต้น

7. ต้องมีส่วนที่มีไว้สำหรับกำหนดคุณสมบัติโดยรอบของโปรแกรม

8. การควบคุม หรือตรวจสอบสถานะต้องทำงานได้ง่าย เช่น ใช้คลิ๊กขวาแล้วมี Popup menu ขึ้นมา หรืออาจใช้เป็น Tool box ก็ได้

9. ต้องมีรายละเอียดการใช้โปรแกรม (Help)

10. การปิดเปิดโปรแกรมต้องทำได้ง่าย

11. การเพิ่ม หรือเอาโปรแกรมออกจากคอมพิวเตอร์ต้องทำได้สะดวก

12. ต้องมีส่วนแสดงสถานะ การทำงานของโปรแกรม

13. ไม่มีข้อจำกัดของโปรแกรมต่อคุณสมบัติของคอมพิวเตอร์

14. โปรแกรมต้องมีขนาดเล็กที่สุด

15. โปรแกรมต้องใช้ทรัพยากรของคอมพิวเตอร์ขณะทำงานน้อยที่สุด

และอาจเพิ่มเติมส่วนประกอบอื่น ๆ เข้าไปอีก เพื่อความสะดวกของผู้ใช้มากที่สุด

3.2.2 การติดต่อกับผู้ใช้ผ่าน Key

ในการใช้คอมพิวเตอร์แต่ละครั้งจะกินพลังงานค้อนข้างมาก หรือในกรณีที่คอมพิวเตอร์ชำรุด หรือในการควบคุมที่ต้องการความรวดเร็ว และต้องเป็น RCU ด้วย จึงต้องมีการควบคุมอีกส่วนหนึ่งที่เข้ามารองรับสิ่งเหล่านี้ โดยการใช้ Key เมตริก 3 X 4 ในการสั่งงานและใช้ลำโพง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นตัวแสดงผล 2 สิ่งนี้จะช่วยได้มากและต้องมีข้อกำหนดต่าง ๆ เพื่อให้เกิดความสะดวกมากที่สุด มีรายละเอียดดังต่อไปนี้

1. Key ต้องมีขนาดเหมะสม
2. เสียงต้องได้ยินชัดเจน
3. ต้องมีหลอดไฟแสดงสถานะของอุปกรณ์ไฟฟ้า
4. ต้องมีความแน่นอน
5. ต้องมีความเร็วสูงกว่าคอมพิวเตอร์
6. ต้องกินไฟน้อยที่สุด
7. ต้องมีความปลอดภัยสูง
8. ต้องมีความสะดวกหรือสามารถใช้งานได้ง่าย

3.3 ชุดตรวจสอบการทำงาน

การควบคุมที่กล่าวมาทั้งหมดจะไร้ความหมายถ้าไม่ทราบสภาวะการทำงานของอุปกรณ์ไฟฟ้าที่เราควบคุมอยู่ จึงจำเป็นต้องมีชุดตรวจสอบสภาวะการทำงาน โดยจะต้องมีเงื่อนไขโปรแกรมตรวจสอบดังนี้

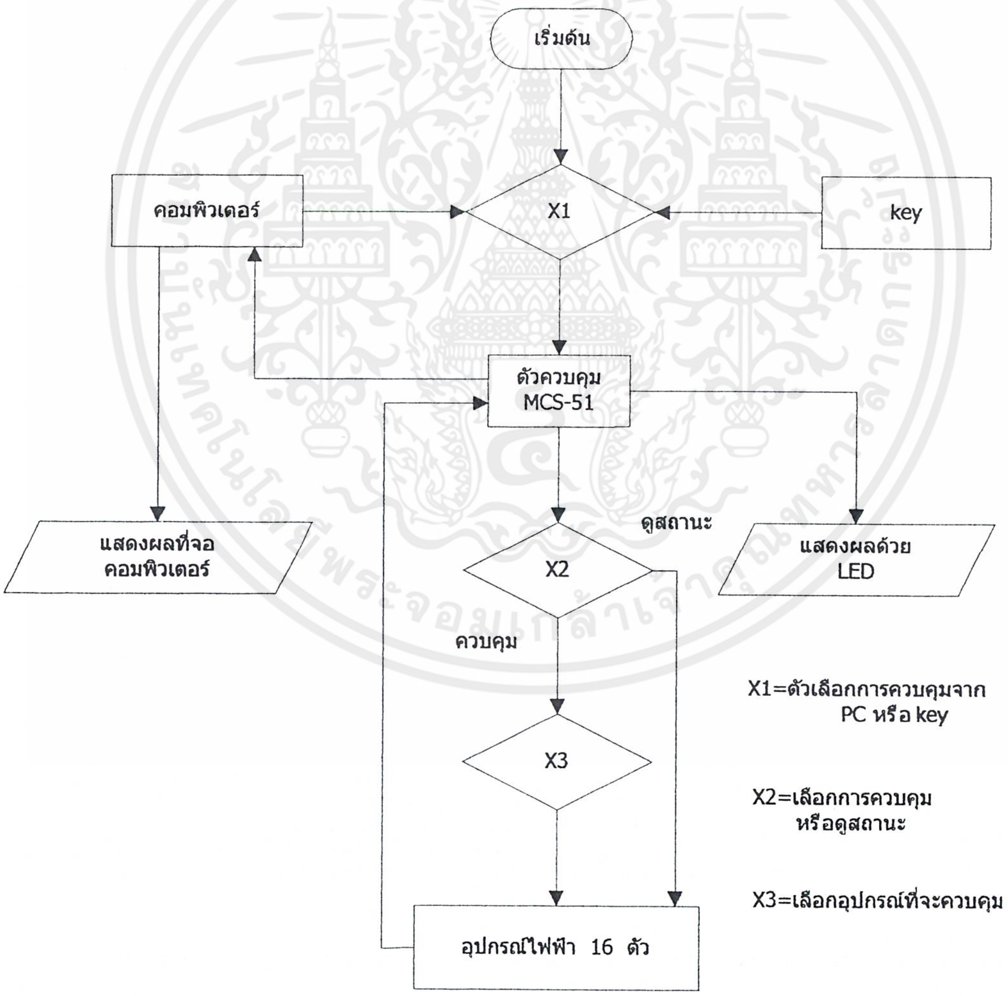
1. ต้องมีการเชื่อมต่อกับชุดควบคุม
2. ต้องแสดงสภาวะได้ตลอดเวลา
3. สภาวะที่ตรวจสอบต้องเป็นการตรวจสอบที่อุปกรณ์ไฟฟ้ามิใช่การตรวจสอบที่การสั่งงาน เช่น ตรวจสอบจากกระแสที่ไหลเข้าอุปกรณ์ไฟฟ้า
4. ตัวตรวจสอบต้องไม่เป็นข้อจำกัดของอุปกรณ์ไฟฟ้า
5. ต้องตรวจสอบต้องตรวจอุปกรณ์ได้ทุกชนิดโดยไม่มีเงื่อนไข
6. ต้องมีความปลอดภัยสูง เพราะอุปกรณ์ไฟฟ้าใช้พลังงานสูง เช่น ใช้การแยกกราวด์ หรือ ใช้การแยกทางแสง

เงื่อนไขที่กล่าวมาทั้งหมดทั้ง 3 หัวข้อเป็นเงื่อนไขที่ดีซึ่งจะทำให้เกิดความสะดวกสบายกับผู้ใช้ได้มากที่สุด แต่ในการออกแบบจริงบางส่วนอาจทำไม่ได้ บางส่วนอาจทำได้ดีกว่า จึงต้องมีการปรับเปลี่ยนโดยยึดความสะดวก และความปลอดภัยในการใช้งานเป็นสิ่งสำคัญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
วงจรและการทำงาน

จากแนวคิดการทำงานในบทที่ 3 เราสามารถนำมาออกแบบวงจรโดยให้มีคุณสมบัติที่กล่าวมา โดยทำการแบ่งวงจรออกเป็น 3 ส่วน เช่นเดียวกับแนวคิดเพื่อความสะดวกโดยสามารถสรุปรายละเอียดเป็นลำดับขั้นการทำงานได้ดังรูปที่ 4.1



รูปที่ 4.1 ลำดับการทำงานโดยรวม

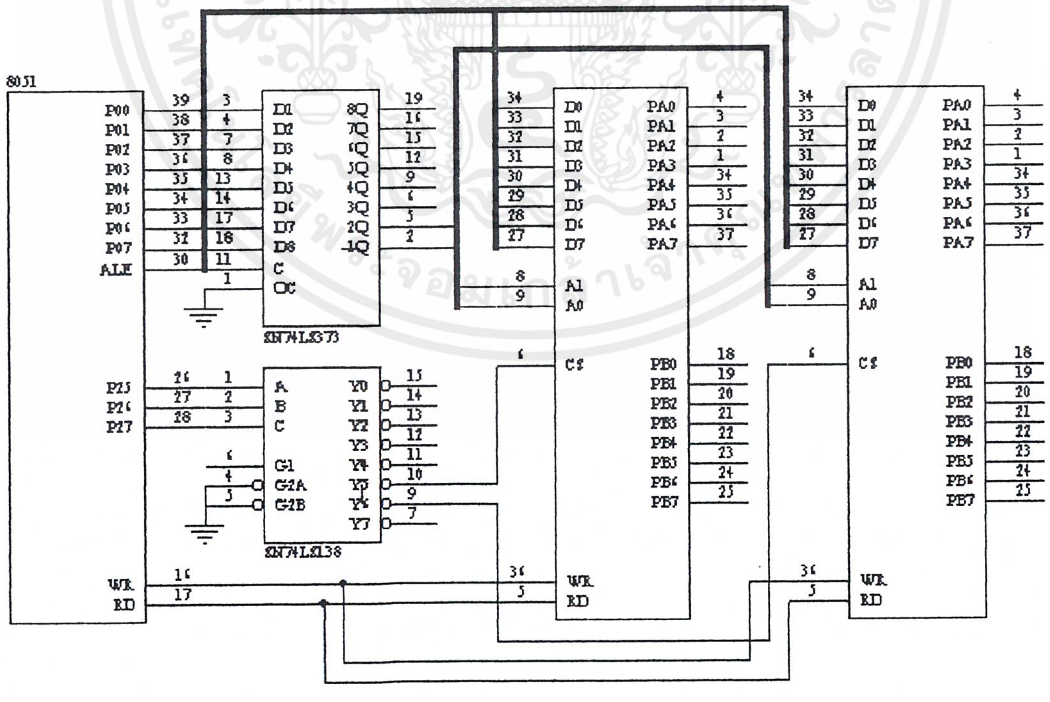
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถเลือกการควบคุมและการแสดงผลได้จาก 2 ส่วนคือ ส่วนแรกควบคุมและแสดงผลที่คอมพิวเตอร์ ส่วนที่สองคือควบคุมที่ key แสดงผลที่หลอด LED โดยทั้งสองต้องติดต่อกับไมโครคอนโทรลเลอร์ MCS-51 ซึ่งจะเป็นตัวนำคำสั่งที่ได้รับจากสองส่วนบนไปปฏิบัติ และคอยตรวจสอบสถานะ การทำงานของอุปกรณ์แล้วส่งกลับไปแสดงผลโดยสามารถแยกอธิบายได้เป็น 3 วงจรดังนี้

- วงจรชุดควบคุม
- วงจรชุดการควบคุมและแสดงผล
- วงจรชุดตรวจสอบสถานะ การทำงาน

4.1 วงจรชุดควบคุม

วงจรชุดควบคุมเป็นส่วนที่ประกอบด้วยอุปกรณ์ไมโครคอนโทรลเลอร์ซึ่งมีลักษณะการเชื่อต่อดังรูปที่ 4.2



รูปที่ 4.2 วงจรของชุดควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1 รายละเอียดและส่วนประกอบของวงจร

วงจรชุดควบคุม ประกอบด้วยอุปกรณ์ด้านไมโครที่นำมาต่อร่วมกับ MCS-51 แต่ละชิ้นมีหน้าที่ดังนี้

1. อุปกรณ์เลือกตำแหน่ง

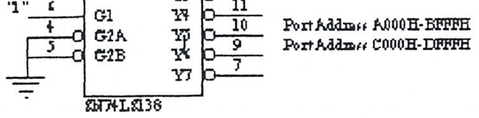
เนื่องจากมีความจำเป็นต้องใช้พอร์ทในการติดต่อมากกว่า 4 พอร์ท ที่มีใช้งาน MCS-51 ดังนั้นจึงจำเป็นต้องมีการขยายพอร์ท โดยพอร์ทที่เพิ่มขึ้นมาจะใช้พื้นที่ของหน่วยความจำภายนอกบางส่วน (ตำแหน่ง 0000H – FFFFH) สัญญาณที่ใช้อ้างตำแหน่งคือ A0 – A15 สัญญาณที่รับส่งข้อมูลได้แก่ D0 – D7 และสัญญาณควบคุมการติดต่อได้แก่ RD และ WR

การจัดตำแหน่งของพื้นที่หน่วยความจำ มีลักษณะการแบ่งออกเป็นส่วน ๆ ดังนี้

ตารางที่ 4.1 การจัดตำแหน่งของพื้นที่หน่วยความจำ

8K RAM	E000H - FFFFH
8K RAM	C000H – DFFFH
8K RAM	A000H – BFFFH
8K RAM	8000H – AFFFH
8K RAM	6000H – 7FFFH
8K RAM	4000H – 5FFFH
8K RAM	2000H – 3FFFH
8K RAM	0000H – 1FFFH

อุปกรณ์ที่นำมาเลือกตำแหน่งในที่นี้ คือ ไอซีเลือกตำแหน่ง 74LS138 ซึ่งมีลักษณะการต่อใช้งานดังรูป สัญญาณที่ควบคุมการเลือกตำแหน่งได้แก่ A15 A14 A13



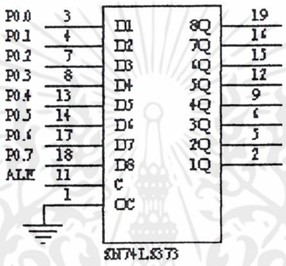
รูปที่ 4.3 อุปกรณ์เลือกตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. อุปกรณ์เก็บพักข้อมูล

เนื่องจากมีความจำเป็นต้องมีการเก็บพักข้อมูลชั่วคราวเพื่อให้เกิดการทำงานที่สอดคล้องกับการทำงานของ MCS-51 จึงจำเป็นต้องนำอุปกรณ์ชิ้นนี้มาต่อรวม ซึ่งได้แก่ ไอซี 74LS373

สัญญาณข้อมูลที่นำมาจาก MCS-51 จะนำมาจากขา D0 – D7 และสัญญาณควบคุมได้แก่ ALE ซึ่งควบคุมการพักเก็บข้อมูลของไอซี 74LS373 ลักษณะการจัดตำแหน่งขา และการใช้งานมีลักษณะดังนี้



รูปที่ 4.4 การใช้งาน 74LS373 พักข้อมูล

3. อุปกรณ์ Peripheral Interface (PPI)

อุปกรณ์ที่ทำหน้าที่เชื่อมต่อนี้ได้แก่ ไอซี 8255 PPI ซึ่งทำหน้าที่เชื่อมต่อระหว่างไมโครโปรเซสเซอร์ กับอุปกรณ์ภายนอกโดยในตัวของ 8255 จะมีพอร์ทสำหรับเชื่อมต่อกับอุปกรณ์ภายนอกแบบขนานอยู่ 3 พอร์ท ๆ ละ 8 บิท โดยสามารถกำหนดให้เป็นพอร์ทอินพุท หรือเอาท์พุทได้ด้วยโปรแกรม ลักษณะการต่อใช้งานของ 8255 จะมีลักษณะดังรูปที่ 4.2

4.1.2 การทำงานของวงจร

การทำงานของวงจรเริ่มจากการรับสัญญาณอินพุทของตัว MCS-51 ดังที่กล่าวมาแล้วว่าสัญญาณอินพุทที่นำมาเป็นสัญญาณควบคุมนำมาจากแผงสวิทช์คีย์ และคอมพิวเตอร์ ดังนั้นเมื่อได้รับสัญญาณอินพุทจากแหล่งกำเนิดแล้วตัวไมโครคอนโทรลเลอร์ จะเกิดจากอินเตอร์รัพให้เกิดการทำงานตรวจสอบแหล่งสัญญาณควบคุม

การเกิดอินเตอร์รัพของ 8051 สามารถเกิดได้ 2 กรณี ได้แก่ การเกิดขึ้นเนื่องจากการอินเตอร์รัพที่ขา PB2 ซึ่งเป็นอินเตอร์รัพจากภายนอกนำสัญญาณมาจากภาคถอดรหัสสัญญาณโทรศัพท์ ส่วนที่สอง สัญญาณอินเตอร์รัพนำมาจากพอร์ทสื่อสารอนุกรม ซึ่งมีความเร็วในการสื่อสาร 9600 บิทต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากการเกิดอินเตอร์รัพภายใน MCS-51 แล้ว MCS-51 จะมีการกำหนดให้อุปกรณ์รับส่ง 8255 มีโหมดการทำงานดังนี้ คือ พอร์ท A เป็นพอร์ทเอาท์พุท พอร์ท B เป็นพอร์ทอินพุท รับข้อมูลจากภาคตรวจสอบสถานะ

การเลือกตำแหน่งของ 8255 สามารถทำโดยส่งแอดเดรสออกไปเลือกตำแหน่งโดยมีตำแหน่งดังนี้ A000H, A001H, A002H, A008H ซึ่งเป็นตำแหน่งของพอร์ท A พอร์ท B พอร์ท C และคอนโทรพอร์ทตามลำดับ สำหรับ 8255 ตัวที่ 1

ส่วน 8255 ตัวที่ 2 มีตำแหน่งแอดแครส ดังนี้ C000H, C001H, C0002H, and C0003H ดังนั้นการควบคุมอุปกรณ์สามารถทำโดยส่งข้อมูลออกไปยังพอร์ท A ของ 8255 ทั้ง 2 ตัว และการรับสัญญาณอินพุทส่งให้คอมพิวเตอร์เพื่อแสดงผลด้วยพอร์ท B ทั้ง 2 ตัว สำหรับลักษณะการต่อของ 8255 พอร์ท B เป็นอินพุทนั้นจำเป็นต้องมีการต่อความต้านทาน pull down เพื่อความเสถียรของสัญญาณอินพุทที่จะรับไปแสดงผลต่อไป

4.2 การควบคุมและแสดงผล

วงจรชุดควบคุมและแสดงผลก็แบ่งออกเป็น 2 ส่วนเช่นเดียวกับแนวการออกแบบ คือมีวงจรที่เชื่อมต่อระหว่างคอมพิวเตอร์กับชุดควบคุมและวงจรการเชื่อมต่อKey เข้ากับชุดควบคุม ทำให้เกิดความคล่องตัวในการออกแบบ และสะดวกต่อการอธิบาย

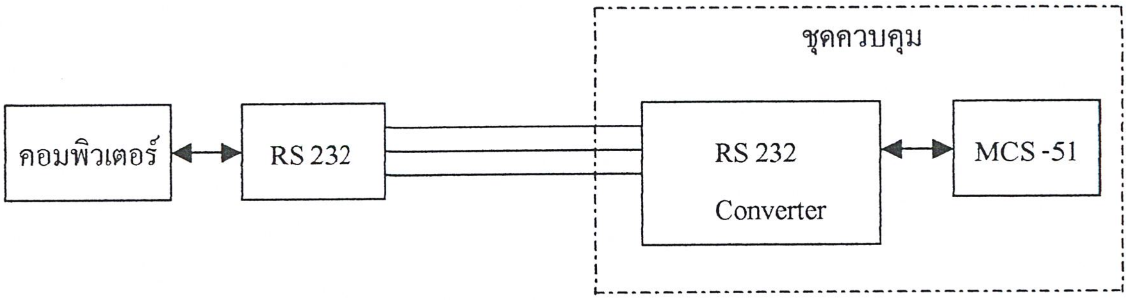
4.2.1 วงจรเชื่อมต่อคอมพิวเตอร์กับชุดควบคุม

ในหัวข้อนี้จะเป็นการออกแบบส่วนที่เชื่อมต่อระหว่างคอมพิวเตอร์กับชุดควบคุม ซึ่งประกอบไปด้วยส่วนของฮาร์ดแวร์ คือ ส่วนเชื่อมต่อกับคอมพิวเตอร์ ส่วนแปลงระดับแรงดันของชุดควบคุม ส่วนของซอฟแวร์ คือ โปรแกรมที่ใช้เชื่อมต่อ หรือติดต่อ หรือแสดงผลให้ผู้ใช้งานทราบ และยังรวมไปถึงกติกาที่ใช้เพื่อให้สามารถส่งข้อมูลและเข้าใจตรงกันทั้งสองฝ่าย ตามรายละเอียดดังต่อไปนี้

1. ส่วนที่เชื่อมต่อและส่วนที่แปลงระดับแรงดันระหว่าง PC กับชุดควบคุม

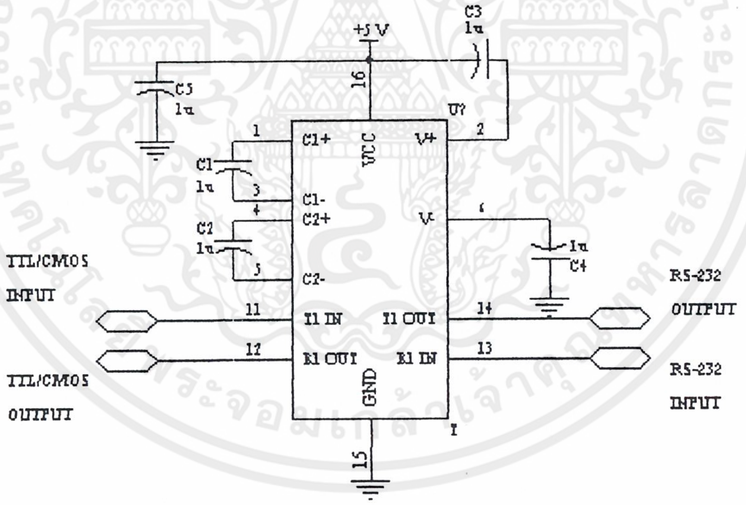
ในการเชื่อมต่อคอมพิวเตอร์เข้ากับชุดควบคุมเราจะใช้พอร์ทอนุกรมของคอมพิวเตอร์ตามมาตรฐานของ RS – 232 โดยใช้การต่อแบบนิลโมเด็ม (ดังที่กล่าวมาแล้วในบทที่ 2) ซึ่งมีลักษณะการต่อดังรูปที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 ลักษณะการเชื่อมต่อคอมพิวเตอร์ กับ ชุดควบคุม

RS–232 Converter เป็นส่วนที่แปลงระดับแรงดันที่ใช้ใน RS–232 ของคอมพิวเตอร์ (ซึ่งมีขนาด 10 V) ให้เป็นขนาดแรงดันที่ MCS–51 หรือชุดควบคุม (ซึ่งอยู่ที่ 5 VDC) ต้องการและในทางกลับกันก็มีหน้าที่ปรับระดับแรงดังของชุดควบคุมให้เพียงพอ หรือเหมาะสำหรับส่งไปยังพอร์ทอนุกรมตามมาตรฐานของ RS–232 ลักษณะของวงจรที่ใช้ในการแปลงหรือ RS –232 Converter แสดงดังรูปที่ 4.6



รูปที่ 4.6 วงจร RS – 232 Converter และการเชื่อมต่อ

รูปที่ 4.6 แสดงลักษณะขาและวงจรที่ใช้ต่อร่วมกับ MAX 232 ซึ่งจะสร้างระดับแรงดัน 10 V จากระดับแรงดันของ TTL เมื่อชุดควบคุมจะส่งข้อมูลไปหาคอมพิวเตอร์ และเมื่อคอมพิวเตอร์จะส่งข้อมูลมาหาชุดควบคุม MAC 232 จะแปลงระดับแรงดันตามมาตรฐานของ RS –232 ไปเป็นระดับแรงดัน TTL ที่ชุดควบคุมต้องการ โดย MAX232 จะสร้างแรงดัน 10 V จากแหล่งจ่ายของตัวมันเพียง 5 V เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

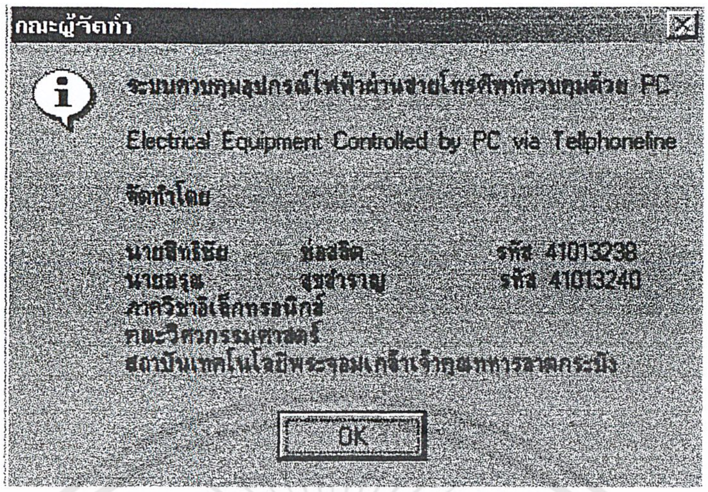
2. ส่วนของโปรแกรมบนคอมพิวเตอร์

ในการพัฒนาโปรแกรมบนคอมพิวเตอร์เลือกใช้โปรแกรม MS Visual Basic 6 ในการพัฒนาและโปรแกรมที่พัฒนาซึ่งเรียกว่า CON_SYS ซึ่งย่อมาจาก Controlling System โดย CON_SYS จะมีหน้าตาคล้ายกับโปรแกรมใน Windows OS ทั่วไปมีส่วนหลัก คือ ส่วนของ ฟอร์มที่เป็นรูปแบบบ้านที่ใช้สำหรับควบคุม และมีอุปกรณ์ไฟฟ้าเป็นไอคอนวางอยู่บนแบบบ้าน อุปกรณ์ไฟฟ้าสามารถเคลื่อนย้ายตำแหน่ง หรือปรับเปลี่ยนรูปได้ ตามแนวทางการออกแบบที่กล่าวมาแล้วในบทที่ 4 และยังมีส่วนประกอบต่าง ๆ อีกมาก เพื่ออำนวยความสะดวกกับผู้ใช้ ดังมีรายละเอียดดังต่อไปนี้

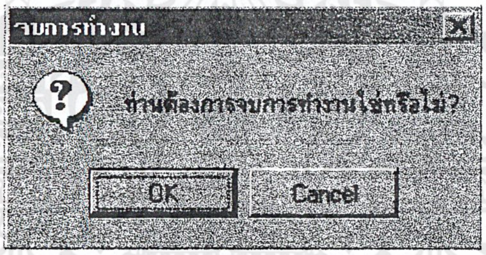
2.1 หน้าต่างแสดงรายละเอียดจะเป็นส่วนที่แสดงรายละเอียดว่าโปรแกรมนี้ชื่ออะไร ใครเป็นผู้สร้างซึ่งเป็นรายละเอียดเพียงคราว ๆ รายละเอียดเต็ม ๆ จะอยู่ในตัวโปรแกรมซึ่งมีหน้าตาดังรูปที่ 4.7

รูปที่ 4.7 ก หน้าต่างแสดงรายละเอียด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

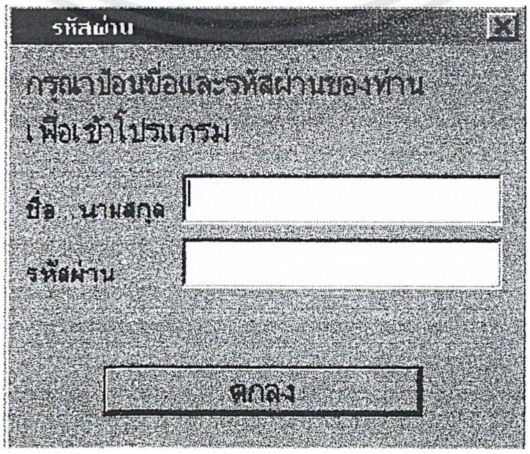


รูปที่ 4.7 ข หน้าต่างย่อยแสดงคณะผู้จัดทำ



รูปที่ 4.7 ค หน้าต่างยกเลิกการทำงาน

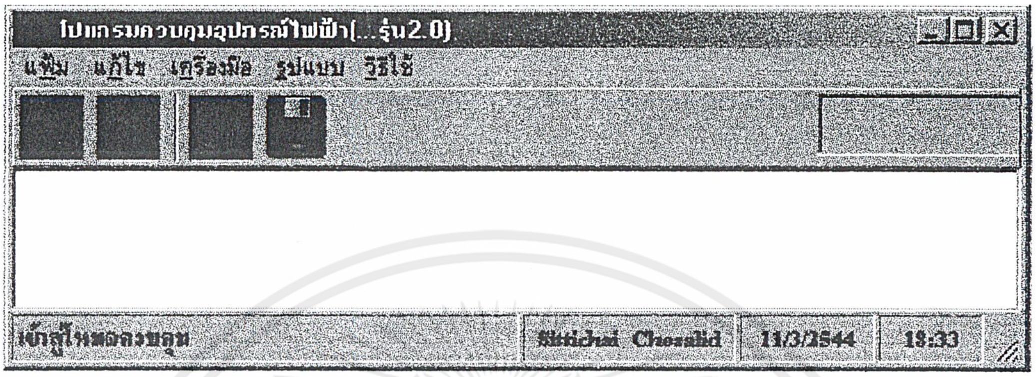
2.2 หน้าต่างป้อนชื่อและรหัสผ่าน หน้าต่างนี้จะแสดงขึ้นหลังจากหน้าต่างแรกแสดงประมาณ 3 วินาที เพื่อให้ผู้ใช้ป้อนชื่อและรหัสผ่านมีลักษณะดังรูปที่ 4.8



รูปที่ 4.8 หน้าต่างป้อนรหัสผ่าน

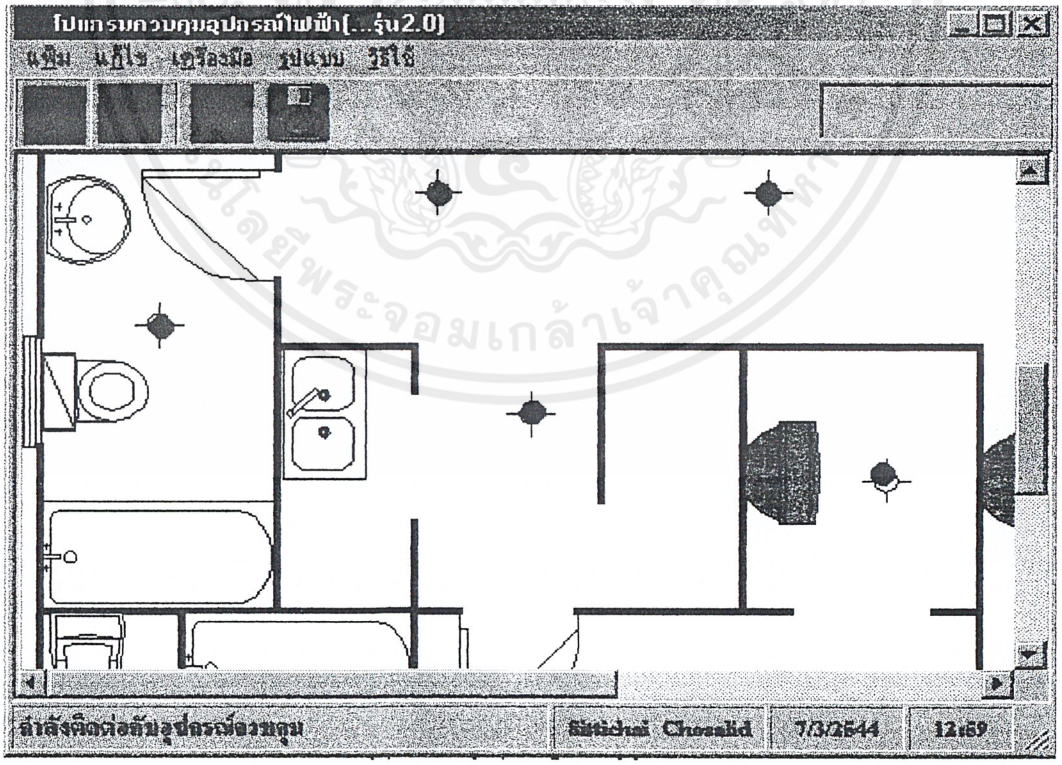
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 หน้าต่างหลักของโปรแกรม จะเป็นหน้าต่างหลักของโปรแกรม รายละเอียดและการทำงานทุกอย่างจะกระทำงานหน้าต่างนี้ ดังรูปที่ 4.9



รูปที่ 4.9 หน้าต่างหลักของโปรแกรม

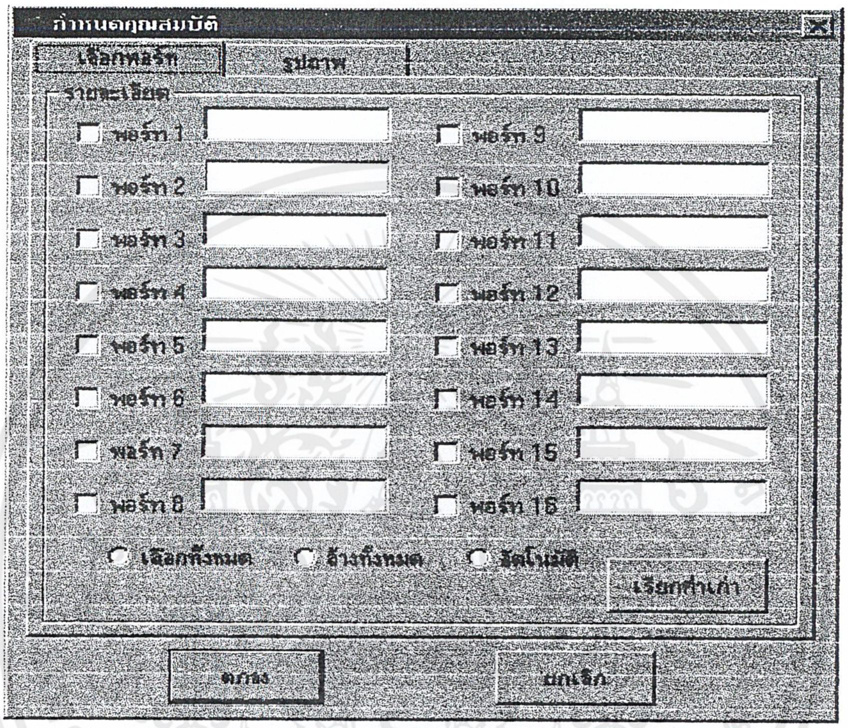
2.4 หน้าต่างกำหนดเอง เป็นหน้าต่างย่อยจะเป็นรูปแบบบ้าน และมีอุปกรณ์ที่จะควบคุมวางอยู่ ณ ตำแหน่งต่าง ๆ ดังรูปที่ 4.10



รูปที่ 4.10 หน้าต่างกำหนดเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 หน้าต่างคุณสมบัติ จะเป็นหน้าต่างที่กำหนดคุณสมบัติโดยรวมของโปรแกรม ซึ่งจะเป็นตัวเลือกรูปไอคอน หรือสัญลักษณ์แทนอุปกรณ์ที่จะควบคุม และรายละเอียดในรูปตัวอักษร ดังรูปที่ 4.11



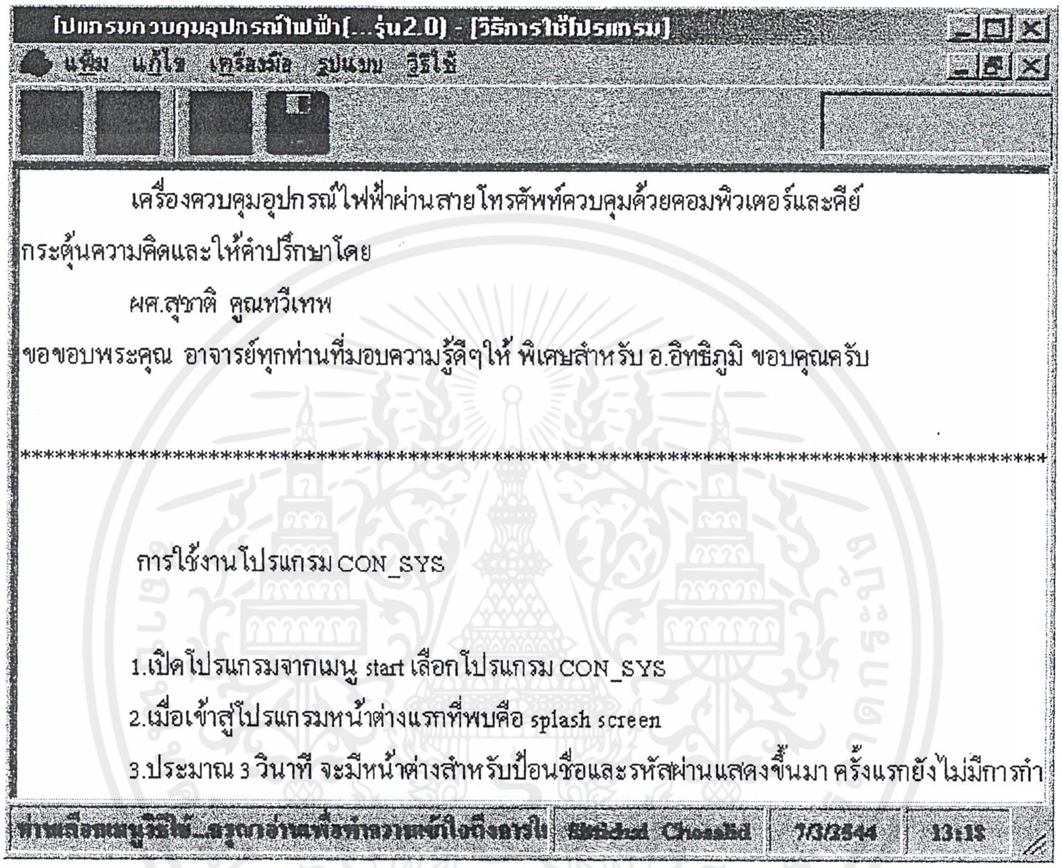
รูปที่ 4.11 หน้าต่างคุณสมบัติ

2.6 หน้าต่างเปลี่ยนรหัสผ่าน เป็นหน้าต่างสำหรับเปลี่ยนรหัสผ่านและชื่อผู้ใช้

รูปที่ 4.12 หน้าต่างเปลี่ยนรหัสผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 หน้าต่างวิธีใช้ เป็นหน้าต่างที่ให้ความช่วยเหลือในการใช้งานโปรแกรมของผู้ใช้มีลักษณะดังรูปที่ 1.13



รูปที่ 4.13 หน้าต่างช่วยเหลือ

3. การใช้โปรแกรม

1. เปิดโปรแกรมโดยการเลือกที่เมนู Start เลือก Program เลือก CON_ SYS
2. เมื่อเข้าสู่โปรแกรมหน้าต่างแรกที่พบคือ Splash screen
3. ประมาณ 3 วินาที จะมีหน้าต่างสำหรับป้อนชื่อและรหัสผ่านแสดงขึ้นมา ครั้งแรกชื่อและรหัสผ่านยังไม่ถูกกำหนด ให้กดปุ่ม “ตกลง” ได้เลย
4. จะเข้าสู่หน้าต่างหลักของโปรแกรม ซึ่งประกอบด้วย เมนู , Toolbox, Progress bar และ Status bar

4.1 เมนูจะประกอบด้วย

เมนูแฟ้ม มีเมนูย่อยดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เริ่มใหม่ ใช้กรณีที่ต้องการเริ่มต้นใหม่โดยไม่ออกจากโปรแกรม
- บันทึกตำแหน่ง ใช้บันทึกตำแหน่งหลังจากมีการเปลี่ยนแปลงตำแหน่ง
- จบการทำงาน

เมนูแก้ไข มีเมนูย่อยดังนี้

- เปลี่ยนตำแหน่งอุปกรณ์ เลือกเมื่อต้องการเปลี่ยนตำแหน่งอุปกรณ์
- เปลี่ยนไอคอน เลือกเมื่อต้องการเปลี่ยนสัญลักษณ์แทนอุปกรณ์ไฟฟ้า

เมนูเครื่องมือ มีเมนูย่อยดังนี้

- ควบคุม เลือกเมื่อต้องการควบคุมอุปกรณ์ไฟฟ้าตามที่เลือก
- ขอดูสถานะ เลือกเมื่อต้องการขอดูสถานะ การทำงานของอุปกรณ์ไฟฟ้า
- เลือกการควบคุมอัตโนมัติ มี 4 รูปแบบคือ เลือกทั้งหมด , เลือก 8 ชุดแรก , เลือก 8 ชุดหลัง และไม่เลือกทั้งหมด
- กำหนดพอร์ท เลือกใช้ได้ 2 พอร์ท คือ พอร์ทอนุกรม 1 และพอร์ทอนุกรม 2
- เปลี่ยนรหัสผ่าน ใช้กำหนดชื่อและรหัสผ่านในครั้งแรก และใช้เปลี่ยนในครั้งต่อไป

เมนูรูปแบบ มีเมนูย่อยดังนี้

- กำหนดเอง เป็นการเลือกฟอร์มแบบบ้านขึ้นมาแสดง
- หน้าต่างคุณสมบัติ ใช้กำหนดคุณสมบัติ

เมนูวิธีการใช้ บอกรายละเอียดและการใช้งานโปรแกรม

4.2 Tool box ประกอบด้วย

- สัญลักษณ์รูปค้อน ใช้กดเพื่อควบคุมอุปกรณ์ไฟฟ้า
- สัญลักษณ์รูปเลนขยาย ใช้กดเพื่อขอดูสถานะ การทำงาน
- สัญลักษณ์รูปวัฏจักร ใช้กดเพื่อเปลี่ยนตำแหน่งสัญลักษณ์ที่ใช้แทนอุปกรณ์
- สัญลักษณ์รูปแผ่นดิสก์ ใช้กดเพื่อบันทึกตำแหน่งที่มีการเปลี่ยนแปลง

4.3 Progress bar ใช้แสดงความก้าวหน้าการควบคุมและการขอดูสถานะการทำงานของอุปกรณ์ไฟฟ้า

4.4 Status bar แสดงเหตุการณ์ต่าง ๆ ที่ผู้ใช้กำลังทำงานอยู่

5. เริ่มต้นการใช้งานด้วยการกำหนดชื่อและรหัสผ่านของตนเอง โดยเลือกเมนูเครื่องมือ และเปลี่ยนรหัสผ่านตามลำดับแล้วกำหนดดังนี้

- ใส่ชื่อในช่อง “ชื่อเจ้าของรหัส”
- ช่องรหัสเดิมไม่ต้องกำหนด เพราะรหัสเดิมไม่มีการกำหนดมาให้จะใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อมีการเปลี่ยนรหัสผ่านในครั้งต่อไป

- ใส่รหัสผ่านในช่อง "รหัสผ่านใหม่"
- ยืนยันรหัสผ่านในช่อง "ยืนยันรหัสใหม่"
- กดปุ่ม "ตกลง" เป็นการสิ้นสุดการกำหนดรหัสผ่าน
- จะมีกล่องข้อความแสดงความสำเร็จในการกำหนดรหัสผ่านถ้าไม่สำเร็จให้ทำใหม่ซ้ำเหมือนเดิม

6. กำหนดพอร์ทที่ใช้เชื่อมต่อกับชุดควบคุมผู้ใช้จะทราบว่าตนต่ออยู่กับพอร์ทใด แล้วทำการกำหนดให้ตรงกัน โดยปกติกำหนดไว้ที่พอร์ทอนุกรม 1

7. เรียกฟอร์มแบบบ้านขึ้นมา โดยเลือกเมนูรูปแบบและกำหนดเองตามลำดับ ถ้าต้องการเปลี่ยนแบบบ้านให้นำแบบบ้านที่ต้องการเปลี่ยน (โดยต้องเป็นBitmap file) ตั้งชื่อว่า myplan.bmp แล้วนำมาเก็บไว้ใน Program files, CON_SYS ตามลำดับ และต้องลบ myplan.bmp ของเดิมทิ้ง (ผู้ใช้ต้องทำตามที่กำหนดไว้ซึ่งอยู่นอกเหนือโปรแกรม CON_SYS)

8. เลือกเมนูแฟ้มและเริ่มต้นใหม่ ทุกครั้งที่มีการเปลี่ยนแบบบ้าน จะมีหน้าต่างกำหนดคุณสมบัติแสดงขึ้นมา ให้กำหนดค่าต่าง ๆ ตามต้องการจากนั้นกดปุ่ม "ตกลง" จะปรากฏฟอร์มรูปแบบบ้านขึ้นมา แต่สัญลักษณ์อุปกณ์ไฟฟ้าจะวางซ้อนกันอยู่ให้เลือกเมนูเปลี่ยนตำแหน่ง (หรือTool box รูปวัฏจักร) เพื่อย้ายไปยังตำแหน่งที่ต้องการเสร็จแล้วให้เลือกเมนูบันทึกตำแหน่ง หรือ Tool box รูปแผ่นดิสก์

9. เมื่อต้องการให้อุปกรณ์ใดทำงานให้ดับเบิ้ลคลิ๊กที่อุปกรณ์นั้น แล้วเลือกเมนูควบคุมและสังเกต การควบคุมจาก Progress bar

10. เมื่อต้องการขอดูสถานะ การทำงานให้เลือกเมนูขอดูสถานะ

<u>หมายเหตุ</u>

การใช้งานสามารถเลือกหรือสั่งงานได้ 3 วิธี วิธีแรกคือ เลือกที่เมนู วิธีที่2 เลือกที่ Tool box และวิธีที่ 3 ใช้คลิ๊กขวา สถานะ การทำงานต่าง ๆ จะถูกแสดงที่ Status bar

4. กติกาที่ใช้ในการเชื่อมต่อ

ในการสื่อสารระหว่าง PC กับ MCS – 51 ใช้การติดต่อแบบ Half Duplex: HDX ซึ่งเป็นการสื่อสารแบบ 2 ทิศทาง แต่สลับเวลากันส่งกล่าวคือ ในขณะที่ PC เป็นตัวส่ง MCS – 51 จะเป็นตัวรับ และถ้า MCS – 51 เป็นตัวส่ง PC จะเป็นตัวรับ

และในการสื่อสารกันระหว่างอุปกรณ์ 2 ตัวจำเป็นต้องมีมาตรฐานในการรับและส่งข้อมูล เพื่อที่การสื่อสารจะได้บรรลุเป้าหมาย มาตรฐานที่ใช้กันโดยทั่วไปมีมากมาย เช่น มาตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐานของ Intel Hex File ซึ่งใช้กันมากในการสื่อสารกับ MCS – 51 แต่เนื่องจากในงานนี้เป็นการสื่อสารกันแบบง่ายมีการรับและส่งข้อมูลไม่มากจึงมีโอกาสผิดพลาดน้อย เราจึงเลือกการกำหนดมาตรฐานขึ้นใช้งานเอง ทำให้เกิดความคล่องตัวในการสื่อสารมากขึ้น เนื่องจากไม่มีการตรวจสอบ

ตารางที่ 4.2 ความหมายของรหัส SR

รหัส (ฐานสิบหก)	ชนิด (ตำแหน่ง/ข้อมูล)	หน้าที่การทำงาน
00	ตำแหน่ง	แสดงตำแหน่ง (Address) ของคอมพิวเตอร์
01	ตำแหน่ง	แสดงตำแหน่งของเครื่องควบคุมชุดที่ 1
02	ตำแหน่ง	แสดงตำแหน่งของเครื่องควบคุมชุดที่ 2
03	ตำแหน่ง	แสดงตำแหน่งของเครื่องควบคุมชุดที่ 3
11	ข้อมูล	แสดงอุปกรณ์ตัวที่ 1 ทำงาน
12	ข้อมูล	แสดงอุปกรณ์ตัวที่ 2 ทำงาน
13	ข้อมูล	แสดงอุปกรณ์ตัวที่ 3 ทำงาน
14	ข้อมูล	แสดงอุปกรณ์ตัวที่ 4 ทำงาน
15	ข้อมูล	แสดงอุปกรณ์ตัวที่ 5 ทำงาน
16	ข้อมูล	แสดงอุปกรณ์ตัวที่ 6 ทำงาน
17	ข้อมูล	แสดงอุปกรณ์ตัวที่ 7 ทำงาน
18	ข้อมูล	แสดงอุปกรณ์ตัวที่ 8 ทำงาน
19	ข้อมูล	แสดงอุปกรณ์ตัวที่ 9 ทำงาน
1A	ข้อมูล	แสดงอุปกรณ์ตัวที่ 10 ทำงาน
1B	ข้อมูล	แสดงอุปกรณ์ตัวที่ 11 ทำงาน
1C	ข้อมูล	แสดงอุปกรณ์ตัวที่ 12 ทำงาน
1E	ข้อมูล	แสดงอุปกรณ์ตัวที่ 13 ทำงาน
1F	ข้อมูล	แสดงอุปกรณ์ตัวที่ 14 ทำงาน
20	ข้อมูล	แสดงอุปกรณ์ตัวที่ 15 ทำงาน
21	ข้อมูล	แสดงอุปกรณ์ตัวที่ 16 ทำงาน
AA	ข้อมูล	รหัสแสดงการควบคุม
CC	ข้อมูล	รหัสแสดงการขอดูสถานะ การทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความผิดพลาดซึ่งอาจเป็นจุดด้อยของการสื่อสารนี้ แต่เพื่อเลี่ยงการผิดพลาดซึ่งมีโอกาสผิดน้อยนี้ด้วยการตรวจสอบวิธีอื่นแทน

มาตรฐานที่กำหนดขึ้นใช้งานเองนี้ขอเรียกว่า มาตรฐาน SR โดยรายละเอียดของ SR แสดงตามตารางที่ 4.2

ในการสื่อสารตามที่กล่าวมาแล้วในโครงงานนี้จะมีการสื่อสารได้ 3 ลักษณะคือ

1. PC ส่งรหัสไปควบคุมอุปกรณ์
2. PC ส่งรหัสไปขอดูสถานะ การทำงาน
3. MCS – 51 ส่งสถานะ การทำงานมาให้ PC

3. 1 PC ส่งรหัสไปควบคุมอุปกรณ์

PC จะเป็นตัวส่งส่วน MCS – 51 จะเป็นตัวรับตามมาตรฐานของ SR ที่เรากำหนดขึ้นจะต้องมีลำดับการส่ง ดังต่อไปนี้

[AA NN CC N1 N2]

AA	คือ	ตำแหน่งของเครื่องควบคุมที่ PC จะติดต่อด้วย
NN	คือ	จำนวนอุปกรณ์ย่อยที่มีการสั่งให้ทำงาน
CC	คือ	รหัสที่บอกว่าเป็นการควบคุม (มีค่าเป็น AAH)
N1, N2…	คือ	อุปกรณ์ย่อยที่มีการสั่งให้ทำงาน

ตัวอย่างเช่น PC ส่งรหัสดังต่อไปนี้ไปให้ MCS – 51

[01 04 AA 11 13 17 18]

มีความหมายว่า PC ต้องการติดต่อกับเครื่องควบคุม เครื่องที่ 1 (รหัส 01H) มีการสั่งให้อุปกรณ์ย่อยในเครื่องที่ 1 ทำงาน (AAH) จำนวน 4 ตัว (04H) ประกอบด้วยตัวที่ 1 (11H) ตัวที่ 3 (13H) ตัวที่ 7 (17H) ตัวที่ 8 (18H)

3.2 PC ส่งรหัสไปขอดูสถานะ การทำงาน

PC จะเป็นตัวส่ง และMCS – 51 จะเป็นตัวรับ โดยมีลำดับการส่งตามมาตรฐานของ SR ดังนี้

[AA XX CC]

AA	คือ	ตำแหน่งของเครื่องควบคุมที่ PC จะขอดูสถานะ
XX	คือ	เลขใด ๆ ก็ได้ในเลขฐานสิบหก (xxH)
CC	คือ	รหัสขอดูสถานะ (CCH)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. 3 MCS – 51 ส่งสถานะไปให้ PC

จะตรงกันข้ามกับสองข้อแรก คือ MCS – 51 จะเป็นตัวส่ง และ PC จะเป็นตัวรับ โดยมีรายละเอียดตามมาตรฐาน SR ดังต่อไปนี้

[AA NN   N1   N2]

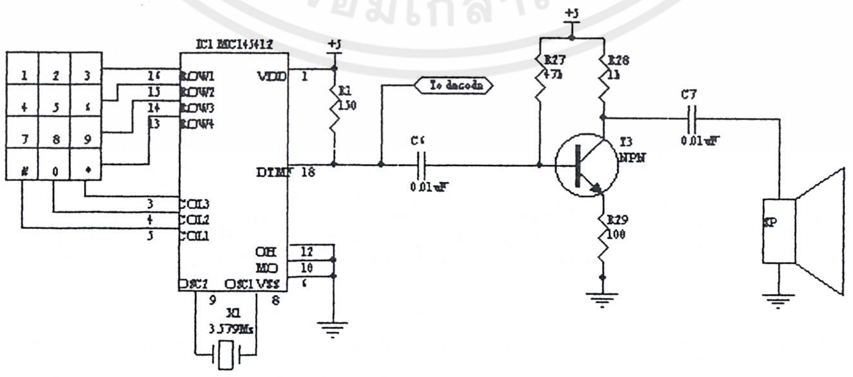
AA	คือ	ตำแหน่งของ PC (00H)
NN	คือ	จำนวนอุปกรณ์ย่อยที่มีการทำงานอยู่ในขณะที่ขอดูสถานะ
N1, N2 …	คือ	อุปกรณ์ย่อยต่าง ๆ ที่มีการทำงานอยู่

<u>หมายเหตุ</u>

ในการอธิบายเราจะแสดงด้วยเลขฐานสิบหกเสมอ เนื่องจากใน MCS – 51 ใช้เลขฐานสิบหกในการเขียนโปรแกรมควบคุม เพื่อจะได้เป็นการง่ายในการดูรายละเอียดของโปรแกรมที่แนบมาตอนท้ายของเล่ม แต่ในการสื่อสารผ่าน RS 232 PC PORT นั้นจะใช้เลขฐานสองในการสื่อสารเท่านั้น

4.2.2 วงจรการเชื่อมต่อคีย์เข้ากับชุดควบคุม

ส่วนของคีย์จะปีนส่วนที่มีการสำรองไว้ในกรณีที่ควมพิวเตอร์เกิดปัญหาหรือกรณีที่ต้องการควบคุมอุปกรณ์ไฟฟ้าโดยไม่เปิดคอมพิวเตอร์ โดยจะใช้หลักการเดียวกับระบบโทรศัพท์แบบความถี่คู่ (DTMF : Dual Tone Multi-Frequency) มีการนำสองความถี่มารวมกันในการกดแต่ละครั้ง และมีการแสดงผลด้วยเสียงทำให้เวลากดทราบว่ากดคีย์อะไร โดยจะมีเสียงเหมือนกับการกด

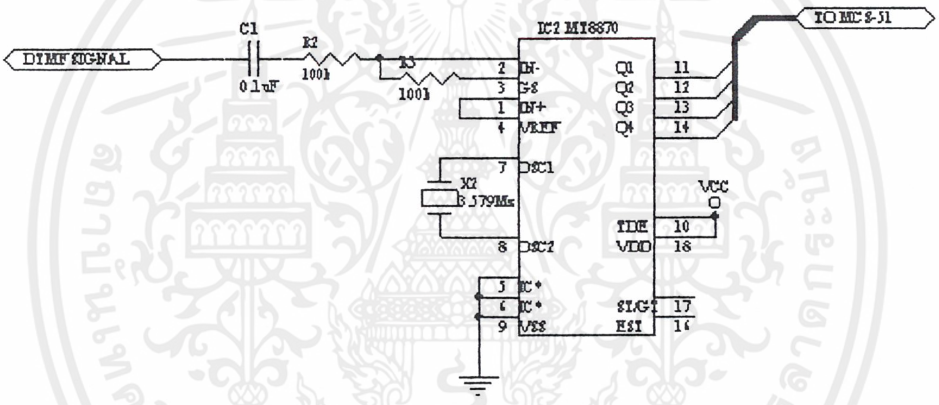


รูปที่ 4.14 แสดงวงจรจำลองสัญญาณ DTMF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์โทรศัพท์ทั่วไป แล้วนำความถี่ของสองความถี่ที่รวมกันมาแยกออกแล้วเข้ารหัสเป็นเลขฐานสอง ส่งต่อไปให้ไมโครคอนโทรลเลอร์ทำการประมวลผลต่อไปโดยมีรายละเอียดวงจรดังรูปที่4.14 และ รูปที่1.15

สัญญาณจำลอง DTMF เป็นสัญญาณที่สร้างจาก ไอซีเบอร์ MC 145412 โดยการต่อร่วมกับอุปกรณ์รอบข้าง ซึ่งได้แก่ คริสตอล 3.579 MHz ใช้ในการกำเนิดสัญญาณนาฬิกาใช้กับวงจร การเลือกสัญญาณความถี่เอาท์พุท DTMF สามารถทำได้โดยการต่อคีย์สวิทช์ในการเลือก ซึ่งสวิทช์ที่ให้ในการต่อนี้เป็นสวิทช์เมตริกขนาด 3 X 4 สัญญาณเอาท์พุท DTMF ที่ได้จะเป็นสัญญาณป้อนใช้กับภาคอินพุท ของวงจรถอดรหัสสัญญาณ DTMF (DTMF decoder circuit)



รูปที่ 4.15 แสดงวงจรภาคถอดรหัสสัญญาณ DTMF

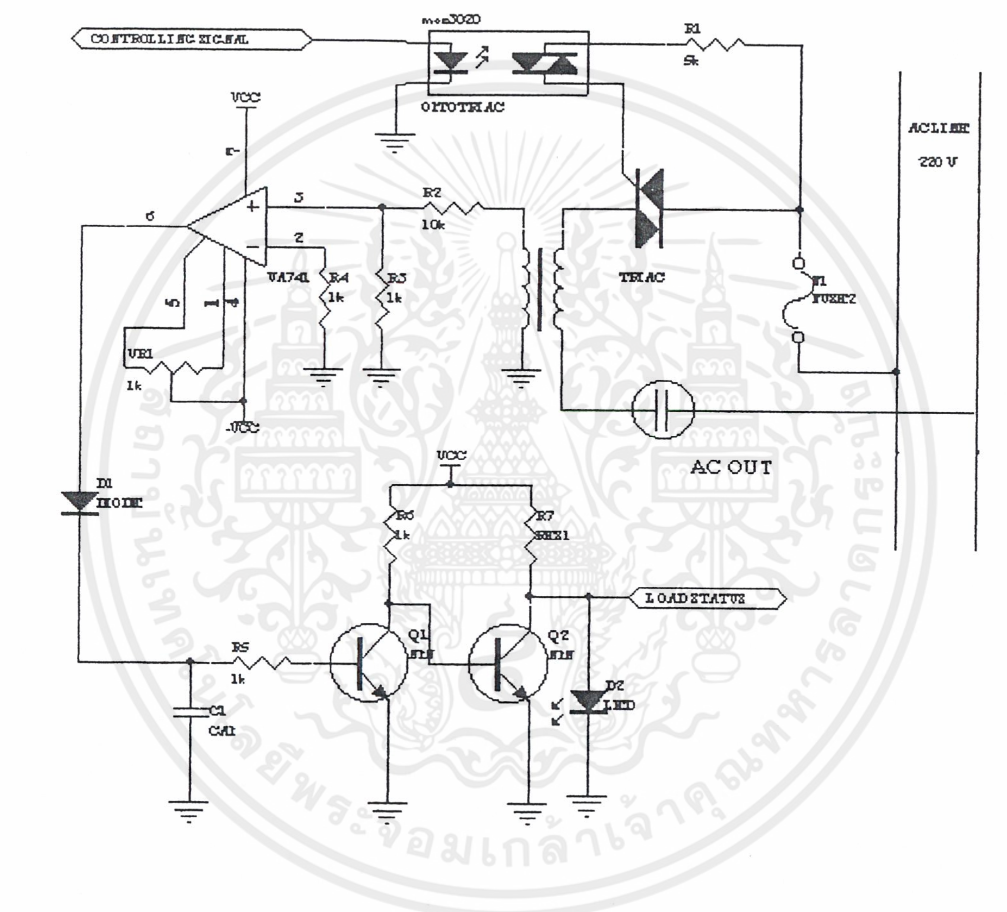
ในรูป 4.15 เป็นวงจรภาคถอดรหัสสัญญาณ DTMF โดยที่จะทำหน้าที่รับเอาสัญญาณ DTMF ที่เข้ามาแล้วเปลี่ยนความถี่ DTMF ให้เป็นสัญญาณดิจิตอลฐานสองสี่บิทออกทางด้านเอาท์พุท การทำงานของวงจร DTMF decoder หรือการทำงานของไอซีเบอร์ MT8870 เป็น DTMF decoder ซึ่งรับสัญญาณ DTMF เข้ามาทางอินพุท ผ่านความต้านทาน R2 = R3 = 100 k และคาปาซิเตอร์ C1 = 0.01 uF ซึ่งประกอบกันเพื่อเปรียบเทียบกับสัญญาณที่เข้ามาทางอินพุท จากนั้นทำการ decoder สัญญาณที่เข้ามาทางอินพุทให้เป็นสัญญาณดิจิตอลฐานสอง (BCD) โดยสัญญาณที่ decoder แล้วจะส่งออกที่ขาQ1–Q4 (ขา11–ขา14)ของMT8870 ส่งไปให้ไมรโครคอนโทรลเลอร์ ที่ขาพอร์ท (P1.0 – P1.3)

ส่วนขา OSC1 และ OSC2 (ขา 7 และขา 8) จะต่อกับคริสตอล ความถี่ 3.579 MHz เพื่อใช้ในการกำเนิดฐานเวลาให้กับการทำงานของ MT8870

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 วงจรตรวจสอบสถานะ การทำงาน

ดังที่ได้กล่าวมาแล้วว่าการตรวจสอบสถานะ การทำงานของอุปกรณ์ก็เป็นส่วนหนึ่ง และเป็นส่วนที่ทำงานร่วมกับแรงดันไฟ 220 V จึงต้องใช้ความระวังเป็นพิเศษ ตัวแสดงรายละเอียดของวงจรดังรูป 4.16 โดยแสดงชุดควบคุมที่ใช้ควบคุมอุปกรณ์ไฟฟ้าได้ 1 ชิ้น



รูปที่ 4.16 แสดงวงจรตรวจสอบสถานะ การทำงาน

จากรูปจะเป็นได้ว่าเราจะแยกส่วนระหว่างชุดควบคุมกับไฟ 220 V ด้วยการแยกทางแสง และการเหนี่ยวนำกระแสผ่านขดลวด ทำให้เกิดความปลอดภัยสูง เพราะชุดควบคุมจะไม่มีกราวด์ร่วมกัน หรือแนวกันเป็นอิสระนั่นเอง

การทำงานชุดควบคุมจะส่งแรงดันไปควบคุมไตรแอค ผ่านสายโทรศัพท์ผ่านการคลัปปลิ้ง ทางแสงด้วย ไอซี MOC 3020 เมื่อ MOC 3020 ได้รับแรงดันไฟตรงที่ I/P เพียงพอที่จะทำให้ LED ภายในตัวมันปล่อยแสงออกมา จะทำให้ไตรแอคทำงาน มีความต้านทานประมาณ 0 โอห์ม เกิดกระแสไหล (กระแสสลับ) ผ่าน R1 ผ่านไดแอคที่อยู่ในตัว MOC 3020 ไปกระตุ้นไตรแอคขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกทของ MAC 210 A ในการออกแบบจะให้กระแสส่วนนี้มีค่ามากพอที่จะทำให้ไตรแอคทำงานได้ และเราจะใช้ Line ของไฟ 220Vac 1 Line มาผ่านไตรแอค เมื่อไตรแอคทำงานก็จะครบวงจรทำให้อุปกรณ์ไฟฟ้าทำงานได้ และเมื่อเราต้องการจะปิดอุปกรณ์ไฟฟ้าก็เพียงแต่หยุดกระตุ้นที่ขาเกทของไตรแอค หรือหยุดจ่ายไฟให้ MOC 3020 นั่นเอง

ที่กล่าวมาจะเป็นส่วนควบคุมที่อยู่ในชุดควบคุมในการตรวจสอบสถานะเราจะใช้ขดลวดเป็นในลักษณะเดียวกับหม้อแปรง โดยให้ขดปฐมภูมิมีจำนวนรอบที่น้อยที่สุด จากการทดลองพบว่าใช้ประมาณ 4 รอบ และขดทุติยภูมใช้ประมาณ 30 รอบขึ้นไป โดยใช้ลวดเบอร์เดียวกัน ประมาณเบอร์ 20 รายละเอียดจะกล่าวถึงในบทการทดลองต่อไป

การตรวจสอบทำได้โดยนำขดปฐมภูมิไปอนุกรมกับไตรแอค หรือก็คือทางเดินกระแส Line หนึ่งที่ไปนำอุปกรณ์ไฟฟ้าเมื่อมีกระแสไหลไปยังอุปกรณ์ไฟฟ้า กระแสก็จะไหลผ่านขดปฐมภูมิด้วย เมื่อมีกระแสไหลผ่านขดปฐมภูมิ กระแสก็จะถูกเหนี่ยวนำข้ามไปยังขดทุติยภูมิ เนื่องจากกระแสที่เหนี่ยวนำข้ามไปมีขนาดน้อยมาก ขดทุติยภูมิจึงต้องมีจำนวนรอบที่มากเพื่อที่จะได้รับกระแสมากเพียงพอที่จะนำไปตรวจสอบต่อไป

แต่เนื่องจากในตอนเริ่มทำงานของอุปกรณ์ไฟฟ้าจะเกิดกระแสชั่วขณะที่มีค่าสูงมากเราจึงนำ R2 ไปต่ออนุกรม เพื่อลดผลจากกระแสส่วนนี้ จากนั้นเปลี่ยนกระแสที่ได้เป็นแรงดันด้วย R3 แล้วนำไปเข้า Op–Amp จะเป็นคลื่นสี่เหลี่ยม เนื่องจากกระแสที่ขดทุติยภูมิเป็นคลื่นไซน์ นำคลื่นที่ได้ไปสร้างเป็นไฟตรงด้วยไดโอด และตัวเก็บประจุแล้วไปป้อนให้ Q1 และ Q2 เพื่อสร้างเป็นสัญญาณป้อนกลับไปยังชุดควบคุม เหตุที่ต้องใช้ Q1 และ Q2 เพื่อต้องการให้ได้ไฟบวกในขณะอุปกรณ์ทำงาน และได้ 0 V ขณะอุปกรณ์ไม่ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การทดลองและผลการทดลอง

ในการทดลองหรือทดสอบโครงงานนี้เราจะแบ่งการทดสอบออกเป็น 3 ส่วน เพื่อความสะดวกและแยกต่อกันเป็นอิสระ โดยการป้อนอินพุทและเอาท์พุท ที่แต่ละชุดต้องการที่เหมือนจริง เนื่องจากในการทดสอบระบบใหญ่ ๆ จะทำให้หาจุดบกพร่องยาก โดยการทดสอบแยกเป็นส่วน ๆ ดังนี้

1. ส่วนของโปรแกรม CON_SYS
2. ส่วนของชุดควบคุม ซึ่งไม่รวมชุดที่เกี่ยวข้องกับไฟ 220V
3. ส่วนของชุดตรวจสอบสถานะ การทำงานและส่วนที่เกี่ยวข้องกับไฟ 220V

เมื่อทดสอบผ่านแล้วเราจะทดสอบทุกชุดร่วมกัน โดยมีเงื่อนไขการควบคุมและการขอดูสถานะของอุปกรณ์ไฟฟ้าที่เปลี่ยนไปเรื่อย ๆ ตามสถานะการณ์ต่าง ๆ

5.1 การทดสอบโปรแกรม CON_SYS

โดยการส่งและการรับข้อมูลกับบอร์ด JAZZ ของบริษัทศิลา และมีการทดสอบตามตารางที่ 5.1

ตารางที่ 5.1 แสดงการทดสอบการรับส่งข้อมูลของคอมพิวเตอร์

ตัวส่ง	ตัวรับ	ข้อมูลที่ส่ง (ฐาน 16)	ข้อมูลที่รับ (ฐาน 16)	หมายเหตุ
คอมพิวเตอร์	บอร์ด JAZZ	01AA00	01AA00	OK
คอมพิวเตอร์	บอร์ด JAZZ	01AA0111	01AA0111	OK
คอมพิวเตอร์	บอร์ด JAZZ	01AA021112	01AA021112	OK
คอมพิวเตอร์	บอร์ด JAZZ	01AA0411121618	01AA0411121618	OK
คอมพิวเตอร์	บอร์ด JAZZ	01AA08111216181A1B1C1D	01AA08111216181A1B1C1D	OK
บอร์ด JAZZ	คอมพิวเตอร์	01AA00	01AA00	OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.1(ต่อ) แสดงการทดสอบการรับส่งข้อมูลของคอมพิวเตอร์

ตัวส่ง	ตัวรับ	ข้อมูลที่ส่ง (ฐาน 16)	ข้อมูลที่รับ (ฐาน 16)	หมายเหตุ
บอร์ด JAZZ	คอมพิวเตอร์	01AA0111	01AA0111	OK
บอร์ด JAZZ	คอมพิวเตอร์	01AA021112	01AA021112	OK
บอร์ด JAZZ	คอมพิวเตอร์	01AA0411121618	01AA0411121618	OK
บอร์ด JAZZ	คอมพิวเตอร์	01AA08111216181A1B1C1D	01AA08111216181A1B1C1D	OK

ผลจากการทดลองที่ได้สรุปได้ว่าการรับส่งข้อมูลผ่านพอร์ทอนุกรมของคอมพิวเตอร์ประสบความสำเร็จ

*<u>หมายเหตุ</u> ที่คอมพิวเตอร์จะแสดงผลในรูปของกราฟฟิก ส่วนที่บอร์ด JAZZ จะแสดงผลเป็นตัวเลขฐาน 16 ดังที่แสดงในตาราง

5.2 ส่วนของชุดควบคุม

การรับและการส่งข้อมูลระหว่างคอมพิวเตอร์และ MCS-51 มีผลการทดสอบดังนี้

ตารางที่ 5.2 ผลการทดสอบของชุดควบคุม

ตัวส่ง	ตัวรับ	ข้อมูลที่รับส่ง	การแสดงผลของข้อมูล	หมายเหตุ
COMPUTER	MCS-51	01AA00	LED ติดสว่างที่พอร์ท A 8255	OK
COMPUTER	MCS-51	01AA0111	LED ติดสว่างที่พอร์ท A 8255	OK
COMPUTER	MCS-51	01AA021112	LED ติดสว่างที่พอร์ท A 8255	OK
COMPUTER	MCS-51	01AA03111213	LED ติดสว่างที่พอร์ท A 8255	OK
COMPUTER	MCS-51	01AA0411121718	LED ติดสว่างที่พอร์ท A 8255	OK
MCS-51	COMPUTER	01AA0111	Computer แสดงผลที่หน้าจอ	OK
MCS-51	COMPUTER	01AA021112	Computer แสดงผลที่หน้าจอ	OK
MCS-51	COMPUTER	01AA03111213	Computer แสดงผลที่หน้าจอ	OK
MCS-51	COMPUTER	01AA0411121718	Computer แสดงผลที่หน้าจอ	OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลจากการทดลองสรุปได้ว่า การรับส่งข้อมูลระหว่างคอมพิวเตอร์กับชุดควบคุมประสบความสำเร็จ

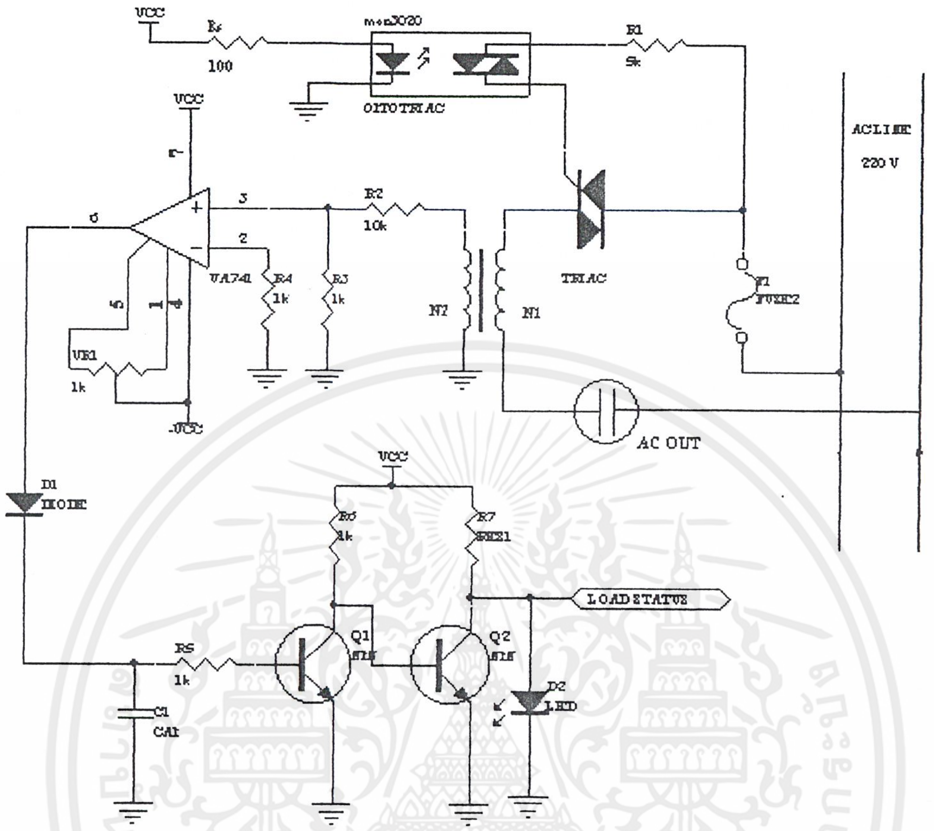
ตารางที่ 5.3 การรับส่งข้อมูลระหว่างคีย์ และ MCS-51

ตัวส่ง	ตัวรับ	ข้อมูล (ได้จากการกดคีย์)	การแสดงผล (ออกที่พอร์ทของ 2855)	หมายเหตุ
KEY	MCS-51	01	LED สว่าง	OK
KEY	MCS-51	02	LED สว่าง	OK
KEY	MCS-51	03	LED สว่าง	OK
KEY	MCS-51	04	LED สว่าง	OK
KEY	MCS-51	05	LED สว่าง	OK
KEY	MCS-51	06	LED สว่าง	OK
KEY	MCS-51	07	LED สว่าง	OK
KEY	MCS-51	08	LED สว่าง	OK
KEY	MCS-51	09	LED สว่าง	OK
KEY	MCS-51	10	LED สว่าง	OK
KEY	MCS-51	11	LED สว่าง	OK
KEY	MCS-51	12	LED สว่าง	OK
KEY	MCS-51	13	LED สว่าง	OK
KEY	MCS-51	14	LED สว่าง	OK
KEY	MCS-51	15	LED สว่าง	OK
KEY	MCS-51	16	LED สว่าง	OK

5.3 ส่วนของชุดตรวจสอบสถานะและส่วนที่สัมพันธ์กับไฟ 220 Vac

เป็นการทดลองเพื่อหาจำนวนรอบที่เหมาะสมของขดลวดด้านขดปฐมภูมิ ทดลองโดยทดสอบจากกระแสโหลดที่ไหลผ่านขดปฐมภูมิว่าสามารถเหนี่ยวนำข้ามไปยังขดทุติยภูมิได้โดยใช้ขดปฐมภูมิจำนวนกี่รอบ โดยจะมีรายละเอียดของวงจรที่ใช้ดังรูปที่ 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.1 วงจรตรวจสอบสถานะและส่วนที่สัมพันธ์กับไฟ 220 Vac สำหรับ 1 ช่อง

โดยการทดสอบเพื่อหาจำนวนรอบของขดปฐมภูมิ และทุติยภูมิ โดยมีรายละเอียดดังตารางที่ 5.4

ตารางที่ 5.4 ผลการทดลองชุดตรวจสอบสถานะเพื่อหาขนาดรอบของ N2ที่เหมาะสม

ขดปฐมภูมิ N1 (รอบ)	ขดทุติยภูมิ N2 (รอบ)	ชนิดของอุปกรณ์ไฟฟ้า					LED
		A	B	C	D	E	
0	30	√	√	√	√	√	ดับ
1	30	√	√	√	√	√	ดับ
2	30	√	√	√	√		ดับ
2	30					√	ติด
3	30	√					ดับ
3	30		√	√	√	√	ติด
4	30	√	√	√	√	√	ติด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>หมายเหตุ</u>

อุปกรณ์ไฟฟ้า	A	คือ	บอร์ด JAZZ V3.0 SLIA
อุปกรณ์ไฟฟ้า	B	คือ	พัดลม 35 W
อุปกรณ์ไฟฟ้า	C	คือ	พัดลม 45 W
อุปกรณ์ไฟฟ้า	D	คือ	ใช้TV ขนาด 7 นิ้ว
อุปกรณ์ไฟฟ้า	E	คือ	ใช้กาต้มน้ำ1670 W

5.4 การทดสอบทั้งโครงงาน

โดยการนำทุกชุดที่ทดสอบผ่านแล้วมาต่อร่วมกันแล้วทดลองพร้อมกัน หรือเป็นการทดลองการใช้งานจริงของเครื่องควบคุม โดยมีเงื่อนไขตามตารางต่อไปนี้

ตารางที่ 5.5 การทดสอบการเปิดปิดอุปกรณ์ไฟฟ้าในกรณีอุปกรณ์ไฟฟ้าปกติ

จำนวนอุปกรณ์ไฟฟ้าที่ถูกควบคุม																การสั่งงาน	ผลการควบคุม	สถานะที่ได้
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			
√	√															ON	OK	OK
√	√	√	√													ON	OK	OK
√	√	√	√	√	√											ON	OK	OK
√	√	√	√	√	√	√	√									ON	OK	OK
√	√	√	√	√	√											OFF	OK	OK
√	√	√	√													OFF	OK	OK
√	√															OFF	OK	OK
														√	√	ON	OK	OK
												√	√	√	√	ON	OK	OK
										√	√	√	√	√	√	ON	OK	OK
								√	√	√	√	√	√	√	√	ON	OK	OK
										√	√	√	√	√	√	OFF	OK	OK
												√	√	√	√	OFF	OK	OK
														√	√	OFF	OK	OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองกับอุปกรณ์ไฟฟ้าที่ชำรุด ทดลองโดยใช้อุปกรณ์ไฟฟ้า 8 ตัว ทำงานพร้อมกัน และมีตัวชำรุดตามตารางควบคุมให้อุปกรณ์ไฟฟ้าทำงาน และสังเกตผล โดยอุปกรณ์ทั้งหมดเป็นหลอดไฟ 3 หลอด กระติกต้มน้ำ 1 ตัว และพัดลม 4 ตัว

ตารางที่ 5.6 การทดสอบกับอุปกรณ์ไฟฟ้าบางตัวที่ชำรุด

ทดลองครั้งที่	อุปกรณ์ไฟฟ้าที่ปกติ				อุปกรณ์ไฟฟ้าที่ชำรุด				การทำงานของอุปกรณ์ไฟฟ้า	การแสดงผล
	1	2	3	4	5	6	7	8		
1	√								OK	OK
2	√	√							OK	OK
3	√	√	√						OK	OK
4	√	√	√	√					OK	OK
5					√				NO	NO
6					√	√			NO	NO
7					√	√	√		NO	NO
8					√	√	√	√	NO	NO
9	√				√				OK , NO	OK , NO

<u>หมายเหตุ</u> การแสดงผลมีสองส่วน คือ แสดงผลที่ LED และจอคอมพิวเตอร์ ซึ่งได้ผลที่เหมือนกันและอุปกรณ์ทั้งหมดเป็นหลอดไฟ 3 หลอด กระติกต้มน้ำ 1 ตัว และพัดลม 4 ตัว

จากผลการทดลองสรุปได้ว่า เครื่องควบคุมอุปกรณ์ไฟฟ้าสามารถทำงานได้อย่างถูกต้อง ทั้งการสั่งให้อุปกรณ์ไฟฟ้าทำงานและหยุดทำงาน รวมทั้งการแสดงผลที่จอคอมพิวเตอร์หรือ LED ถูกต้องตรงกับที่สั่งงาน และที่อุปกรณ์ไฟฟ้าทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
บทสรุปและแนวทางพัฒนา

สรุป

โครงงานนี้ประสบความสำเร็จเนื่องจากสามารถควบคุมการทำงานและตรวจสอบสถานะการทำงานของอุปกรณ์ไฟฟ้าอย่างถูกต้อง ก่อให้เกิดความสะดวกสบายต่อผู้ใช้งานในการนำไปควบคุมอุปกรณ์ไฟฟ้ามาก โดยมีการเขียนโปรแกรมสำหรับควบคุมและแสดงผลอุปกรณ์ที่คอมพิวเตอร์ด้วยภาษา MS Visual Basic 6 ทำให้โปรแกรมง่ายต่อการใช้งาน มีความสะดวกมาก และโปรแกรมการทำงานของ MCS–51 ยังมีความฉลาดทำให้การตัดสินใจควบคุมอุปกรณ์ไฟฟ้าแต่ละครั้งสะดวกและรวดเร็ว การตรวจสอบสถานะจึงเกิดความคล่องตัวสูง และได้สถานะการทำงานของอุปกรณ์ไฟฟ้าที่แท้จริงจากชุดตรวจสอบสถานะการทำงานด้วย แต่ปัญหาที่เกิดขึ้นคือ

1. ตัวตรวจสอบกระแสที่อนุกรมอยู่กับอุปกรณ์ไฟฟ้าจะมีผลกับอุปกรณ์ไฟฟ้าบางตัว เนื่องเกิดจากค่า L จากตัวตรวจจับกระแส สามารถแก้ไขได้ด้วยการลดจำนวนรอบของขดปฐมภูมิ(N1)ของชุดตรวจสอบกระแสลงมา แต่ในการลดจำนวนรอบนั้นต้องคำนึงถึงกระแสของอุปกรณ์ไฟฟ้าด้วย ถ้าอุปกรณ์ไฟฟ้าใช้กระแสมากจะสามารถลดจำนวนรอบได้มาก ถ้าอุปกรณ์ไฟฟ้าใช้กระแสน้อยจะสามารถลดจำนวนรอบได้น้อยเนื่องจากถ้าจำนวนรอบไม่เหมาะสมกับปริมาณกระแสของอุปกรณ์ไฟฟ้า ชุดตรวจสอบสถานะจะไม่ทำงาน
2. การเริ่มต้นการทำงานในแต่ละครั้งจะเกิดกะแสกระชาก ทำให้อุปกรณ์ไฟฟ้าทำงานชั่วขณะหนึ่งซึ่งอาจก่อให้เกิดอันตรายได้

แนวทางพัฒนา

อย่างไรก็ตามโครงงานนี้ยังสามารถพัฒนาให้เกิดความคล่องตัวได้สูงกว่านี้อีก คณะผู้จัดทำได้เล็งเห็นถึงประโยชน์ที่จะเกิดกับผู้ใช้งาน หากมีการนำไปพัฒนาต่อไป คณะผู้จัดทำก็เห็นด้วย และผู้จัดทำขอแนะนำแนวทางในการพัฒนาต่อไปตามหัวข้อดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การเชื่อมต่อคอมพิวเตอร์กับชุดควบคุม สามารถพัฒนาให้มีการตรวจสอบความถูกต้องของข้อมูลได้

2. การควบคุมสามารถพัฒนาให้ควบคุมผ่านเครือข่ายได้

3. การเชื่อมต่อระหว่างชุดควบคุมกับชุดตรวจสอบสถานะสามารถพัฒนาให้เป็นการเชื่อมต่อแบบไร้สายได้ ซึ่งจะก่อให้เกิดความคล่องตัวสูง

4. สามารถเพิ่มหน่วยความจำเพื่อเก็บสถานะ การทำงานของอุปกรณ์ไฟฟ้าไว้ใช้กรณีที่ไฟฟ้าเกิดขัดคล่องชั่วคราว เมื่อไฟปรกติจะสามารถทำงานตามคำสั่งสุดท้ายที่สั่งมาได้

5. กรณีที่ต้องการนำไปใช้กับอุปกรณ์ไฟฟ้าที่มีกำลังสูงอาจเปลี่ยนจากไตรแอคไปเป็นรีเลย์ เพื่อให้สามารถทนกำลังไฟฟ้าที่สูงขึ้นได้

6. สามารถเพิ่มวงจร Zero Crossing เพื่อลดปัญหาการกระชากของกระแสตอนเริ่มต้น

7. สามารถเปลี่ยนวิธีการ Drive ของ CPU ใหม่เป็น Active Low เพื่อลดปัญหาการเริ่มต้นการทำงานของ CPU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1) กิตติ ภักดีวัฒนะกุล. 2542. Visual Basic 6 ฉบับโปรแกรมเมอร์. กรุงเทพ : บริษัท เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด.

2) ชูชัย ธนสารตั้งเจริญ. 2533. การสื่อสารข้อมูล. กรุงเทพ : สำนักพิมพ์ฟิสิกส์เซ็นเตอร์.

3) นรินท์ เนาวประทีป. ออป แอมป์. กรุงเทพ : สำนักพิมพ์ฟิสิกส์เซ็นเตอร์.

4) ปรเมษฐ์ ประนยานันทน์. 2536. ไมโครคอลโทรลเลอร์ MCS-51. กรุงเทพ : บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน)

5) พีรศักดิ์ ศรีกังวาล. 2536. เบสิกขั้นสูง. กรุงเทพ : บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน)

6) ส่ง สุขดานนท์. ไฟฟ้าเบื้องต้น. กรุงเทพ : บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน)

7) ศุภชัย สุรินทร์วงศ์. วงจรแม่เหล็กและหม้อแปลงไฟฟ้าทั่วไป. กรุงเทพ : สมาคมส่งเสริมเทคโนโลยี(ไทย-ญี่ปุ่น).

8) John D. Kraus. 1981. ELECTROMAGNETICS Fourth Edition. Singapore : McGraw-Hill, Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

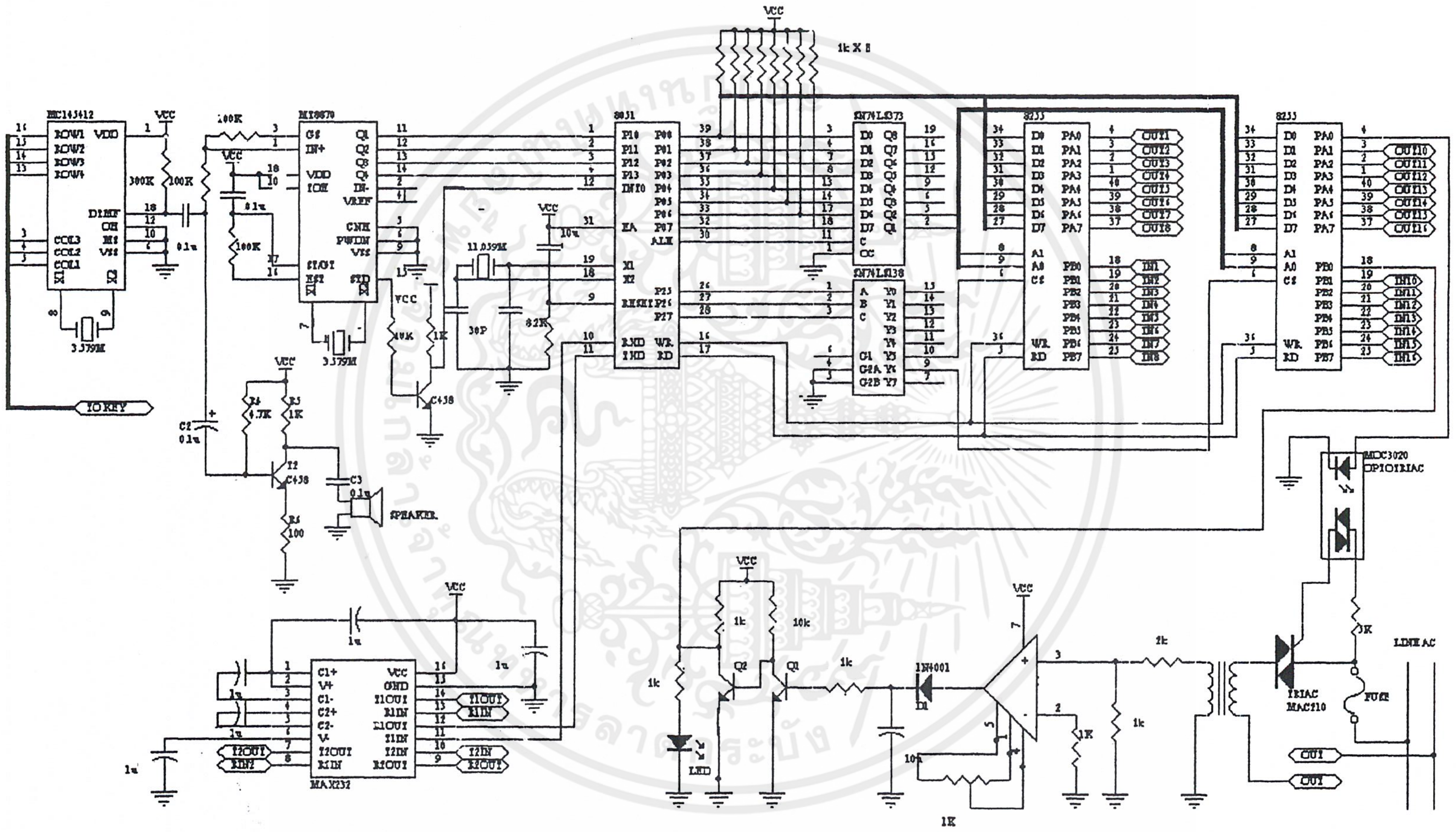
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

รายละเอียดของวงจรรวมและโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงรายละเอียดของวงจรรวมโดยมีชุดตรวจสอบสถานะและชุดขับโหลดเพียงชุดเดียวจาก 16 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของโปรแกรมบนคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

frmCON_SYS

```
Private Sub cmdAbout_Click()
   MsgBox "ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายโทรศัพท์ควบคุมด้วย PC  " + Chr(10) + Chr(10) + _
          "Electrical Equipment Controlled by PC via Tellphoneline" + Chr(10) + Chr(10) + _
          "จัดทำโดย" + Chr(13) + Chr(10) _
               + Chr(10) + "นายสิทธิชัย        ช่อสลิด            รหัส  41013238" _
               + Chr(10) + "นายอรุณ           สุขสำราญ           รหัส  41013240" _
               + Chr(10) + "ภาควิชาอิเล็กทรอนิกส์" _
               + Chr(10) + "คณะวิศวกรรมศาสตร์" _
               + Chr(10) + "สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง" _
                         , vbOKOnly + vbInformation, "คณะผู้จัดทำ"
   frmรหัสผ่าน.Show
End Sub
Private Sub cmdExit_Click()
      จบการทำงาน
      frmรหัสผ่าน.Show
End Sub
Private Sub Timer1_Timer()
     If Timer1.Interval = 1000 Then
          frmรหัสผ่าน.Show
          Timer1.Enabled = False
     End If
End Sub
```

frmHelp

```
Dim mdifrm As MDIForm1
Dim StatusText As String
Private Sub Form_Load()
      Dim iFileNum As Integer
      Dim DataText
      Set mdifrm = MDIForm1
      StatusText = mdifrm.StatusBar1.Panels(1)
      mdifrm.StatusBar1.Panels(1) = "ท่านเลือกเมนูวิธีใช้...กรุณาอ่านเพื่อทำความเข้าใจถึงการใช้งานโปรแกรม"
      iFileNum = FreeFile
      Open "c:\program files\con_sys\help.txt" For Input As #iFileNum
      txthelp.Text = Input(4150, #iFileNum)
      Close #iFileNum
End Sub
Private Sub Form_Unload(Cancel As Integer)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Set mdifrm = MDIForm1
        mdifrm.StatusBar1.Panels(1) = StatusText
End Sub
```

frm กำหนดเอง 1

```
Option Explicit
Dim mdifrm As MDIForm1
Dim NumName As Integer
Dim Address(1 To 16) As PrivateAddress
Sub ControllingStatus()
        Dim n As Integer
        Dim comp As Integer
        For n = 1 To 16
                If Image1(n).Picture = Image2(n).Picture Then
                        comp = 1
                        KeepDataForSend n, comp
                ElseIf Image1(n).Picture = Image3(n).Picture Then
                        comp = 0
                        KeepDataForSend n, comp
                End If
        Next n
End Sub
Private Sub Form_Activate()
        Number = 1
        frmกำหนดเอง1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่เลือก " & i & " อุปกรณ์."
End Sub
Private Sub Form_DragDrop(Source As Control, X As Single, Y As Single)
        Set mdifrm = MDIForm1
        Source.Top = Y - 240
        Source.Left = X - 240
        frmกำหนดเอง1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่เลือก " & i & " อุปกรณ์."
End Sub
Private Sub Form_DragOver(Source As Control, X As Single, Y As Single, State As Integer)
        frmกำหนดเอง1.Caption = "(" & X & "," & Y & ")"
End Sub
Private Sub Form_Load()
        Dim iFileNum As Integer
        Dim n As Integer, k As Integer
        Dim LoadPosition(1 To 16) As Position
        Dim xxx As Position
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim ToolTipTxt(1 To 16) As ToolTipDetail
Unload frmHelp
Unload frmเปลี่ยนรหัสผ่านใหม่
Unload frmเปลี่ยนไอคอน
Unload frmคุณสมบัติ
Set mdifrm = MDIForm1
mdifrm.Toolbar1.Buttons(5).Image = 5
mdifrm.Toolbar1.Buttons(5).Enabled = True
i = 0
Number = 1
mdifrm.StatusBar1.Panels(1) = "ขินดีตอนรับสู่โปรแกรมควบคุมอุปกรณ์ไฟฟ้า..เลือกเมนูวิธีใช้ถ้าต้องการ"
MyPlan.Picture = LoadPicture("c:\Program Files\Con_Sys\MyPlan.bmp")
frmกำหนดเอง1.Picture = MyPlan.Picture
frmกำหนดเอง1.Height = MyPlan.Height + 400
frmกำหนดเอง1.Width = MyPlan.Width + 100
iFileNum = FreeFile
Open "c:\Program Files\Con_sys\IconAddress.sit" For Random As #iFileNum Len = 162
For n = 1 To 16
        Get #iFileNum, n, Address(n)
        Image2(n).Picture = LoadPicture(Address(n).AdOn)
        Image3(n).Picture = LoadPicture(Address(n).AdOff)
Next n
Close #iFileNum
iFileNum = FreeFile
Open "c:\Program Files\Con_sys\ToolTip.sit" For Random As #iFileNum Len = 45
For k = 1 To 16
        Get #iFileNum, k, ToolTipTxt(k)
        Image1(k).ToolTipText = k & "  " & Trim(ToolTipTxt(k).PortDetail)
Next k
Close #iFileNum
Open "c:\Program Files\Con_sys\position.ini" For Random As #iFileNum Len = 17
For n = 1 To 16
        Get #iFileNum, n, LoadPosition(n)
        Image1(n).Left = CSng(LoadPosition(n).PortXaxis)
        Image1(n).Top = CSng(LoadPosition(n).PortYaxis)
        OnOff(n) = LoadPosition(n).PortOnOff
        If ToolTipTxt(n).PortUsed = "1" Then
                If OnOff(n) = 1 Then
                        i = i + 1
                        Image1(n).Visible = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    Image1(n).Picture = Image2(n).Picture
                ElseIf OnOff(n) = "0" Then
                    Image1(n).Visible = True
                    Image1(n).Picture = Image3(n).Picture
                End If
            End If
            Image1(n).DragIcon = Image1(n).Picture
        Next n
        Close #iFileNum
End Sub
```

Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)

```
    Set mdifrm = MDIForm1
    If Button = 2 Then
        PopupMenu mdifrm.popup
    End If
```

End Sub

Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)

```
    Set mdifrm = MDIForm1
    frmกำหนดเอง1.Caption = X & "," & Y
    mdifrm.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)"
```

End Sub

Private Sub Image1_DblClick(Index As Integer)

```
    Dim m As Integer
    Set mdifrm = MDIForm1
    If IconMove = False Then
        If OnOff(Index) = 0 Then
            Image1(Index).Picture = Image2(Index).Picture
            i = i + 1
            OnOff(Index) = 1
        Else
            Image1(Index).Picture = Image3(Index).Picture
            i = i - 1
            OnOff(Index) = 0
        End If
    End If
    frmกำหนดเอง1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่เลือก " & i & " อุปกรณ์."
    mdifrm.StatusBar1.Panels(1) = "เข้าสู่โหมดควบคุม"
    mdifrm.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)......" & "จำนวนอุปกรณ์ที่เลือก " & i & " อุปกรณ์."
```

End Sub

Private Sub Image1_DragDrop(Index As Integer, Source As Control, X As Single, Y As Single)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Set mdifrm = MDIForm1
 Image1(Index).Top = Image1(Index).Top + Y - 240
 Image1(Index).Left = Image1(Index).Left + X - 240
 frmกำหนดเอง1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่เลือก  " & i & "  อุปกรณ์."
End Sub
Private Sub Image1_DragOver(Index As Integer, Source As Control, X As Single, Y As Single, State As Integer)
 frmกำหนดเอง1.Caption = "(" & X & "," & Y & ")"
End Sub
Private Sub Image1_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X As Single, Y As Single)
    Set mdifrm = MDIForm1
    mdifrm.mnu1เปลี่ยไอคอน.Visible = True
    If Button = 2 Then
        PopupMenu mdifrm.popup
    End If
    mdifrm.mnu1เปลี่ยไอคอน.Visible = False
End Sub
```

Frmคุณสมบัติ

```
Dim n As Integer
Dim jj As Integer
Dim F As Integer
Dim G As Integer
Dim AddressNameOfPicOn As String
Dim AddressNameOfPicOff As String
Dim TapStrip1Change As Boolean
Dim frmกำหนด1 As frmกำหนดเอง1
Sub TapStrip1Selected()
 Frame1.Caption = "รายละเอียด"
 For n = 1 To 16
     chkport(n).Visible = True
     txtรายละเอียด(n).Visible = True
 Next n
 optเลือกทั้งหมด.Visible = True
 optล้างทั้งหมด.Visible = True
 optอัตโนมัติ.Visible = True
 cmdเรียกค่าเก่า.Visible = True
 PictureOn.Visible = False
 PictureOff.Visible = False
 cmdอิสระ.Visible = False
 lblคำนำ.Visible = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lblอุปกรณ์ทำงาน.Visible = False
        lblอุปกรณ์ไม่ทำงาน.Visible = False
        chkกำหนดรูป.Visible = False
        chkรูปอิสระ.Visible = False
        cmdForword.Visible = False
End Sub
Sub TapStrip2Selected()
        Dim n As Integer
        Frame1.Caption = "เปลี่ยนรูปภาพ"
        cmdForword.Visible = True
        cmdอิสระ.Visible = True
        PictureOn.Visible = True
        PictureOff.Visible = True
        lblคำนำ.Visible = True
        lblอุปกรณ์ทำงาน.Visible = True
        lblอุปกรณ์ไม่ทำงาน.Visible = True
        chkกำหนดรูป.Visible = True
        chkรูปอิสระ.Visible = True
        For n = 1 To 16
                chkport(n).Visible = False
                txtรายละเอียด(n).Visible = False
        Next n
        optเลือกทั้งหมด.Visible = False
        optล้างทั้งหมด.Visible = False
        optอัตโนมัติ.Visible = False
        cmdเรียกค่าเก่า.Visible = False
End Sub
Private Sub chkport_Click(Index As Integer)
        If chkport(Index).Value = 1 Then
                txtรายละเอียด(Index).Enabled = True
                txtรายละเอียด(Index).SetFocus
        Else
                txtรายละเอียด(Index).Enabled = False
        End If
End Sub
Private Sub chkกำหนดรูป_Click()
        If chkกำหนดรูป.Value = 1 Then
                chkรูปอิสระ.Enabled = False
                cmdForword.Enabled = True
                AddressNameOfPicOn = "c:\Program Files\Con_sys\pic\pic14.ico"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            AddressNameOfPicOff = "c:\Program Files\Con_sys\pic\pic15.ico"
            PictureOn.Picture = LoadPicture(AddressNameOfPicOn)
            PictureOff.Picture = LoadPicture(AddressNameOfPicOff)
            jj = 1
            F = 0
            G = 1
        Else
            chkรูปอิสระ.Enabled = True
            cmdForword.Enabled = False
        End If
End Sub
Private Sub chkรูปอิสระ_Click()
        If chkรูปอิสระ.Value = 1 Then
            chkกำหนดรูป.Enabled = False
            cmdอิสระ.Enabled = True
        Else
            chkกำหนดรูป.Enabled = True
            cmdอิสระ.Enabled = False
        End If
End Sub
Private Sub cmdForword_Click()
        Dim picon As String
        Dim picoff As String
        AddressNameOfPicOn = "c:\Program Files\Con_sys\pic\pic" & Format(F, "00") & ".ico"
        AddressNameOfPicOff = "c:\Program Files\Con_sys\pic\pic" & Format(G, "00") & ".ico"
        PictureOn.Picture = LoadPicture(AddressNameOfPicOn)
        PictureOff.Picture = LoadPicture(AddressNameOfPicOff)
        If F = 14 Then
            F = 0
            G = 1
        Else
            F = F + 2
            G = G + 2
        End If
End Sub
Private Sub cmdตกลง_Click()
        Dim ToolTipText(1 To 16) As ToolTipDetail
        Dim iFileNum As Integer
        Dim Address  As PrivateAddress
        Dim k As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Dim n As Integer
 Set frmกำหนด1 = frmกำหนดเอง1
iFileNum1 = FreeFile
If chkกำหนดรูป.Value = 1 Then
        Open "c:\Program Files\Con_sys\IconAddress.sit" For Random As #iFileNum1 Len = 162
        For n = 1 To 16
              Address.AdOn = AddressNameOfPicOn
              Address.AdOff = AddressNameOfPicOff
              Address.EndLine = Chr(13) & Chr(10)
              Put #iFileNum1, n, Address
        Next n
        Close #iFileNum1
End If
iFileNum1 = FreeFile
For k = 1 To 16
        If chkport(k).Value <> 0 Then
              TapStrip1Change = True
              Exit For
        Else
              TapStrip1Change = False
        End If
  Next k
If TapStrip1Change = True Then
        If MsgBox("กรุณายืนยันว่าท่านจะใช้รายละเอียดที่ท่านเปลี่ยนนี้ ท่านจะไม่สามารถกลับไปใช้รายละเอียดเดิมได้",
vbYesNoCancel + vbCritical, "กรุณายืนยันการเปลี่ยนแปลง") = vbYes Then
              Open "c:\Program Files\Con_sys\ToolTip.sit" For Random As #iFileNum1 Len = 45
              For n = 1 To 16
                    ToolTipText(n).PortNumber = Format(n, "00")
                    ToolTipText(n).PortDetail = txtรายละเอียด(n).Text
                    ToolTipText(n).PortUsed = chkport(n).Value
                    ToolTipText(n).EndLine = Chr(13) & Chr(10)
                    Put #iFileNum1, n, ToolTipText(n)
              Next n
              Close #iFileNum1
        End If
End If
If NewNew = False Then
        Unload Me
Else
        frmกำหนดเอง1.Show
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            For n = 1 To 16
                frmกำหนด1.Image1(n).Top = 20
                frmกำหนด1.Image1(n).Left = 20
            Next n
            NewNew = False
            Unload Me
        End If
End Sub
Private Sub cmdยกเลิก_Click()
        Unload Me
End Sub
Private Sub cmdเรียกค่าเก่า_Click()
        Dim iFileNum As Integer
        Dim ReadTooltipText(1 To 16) As ToolTipDetail
        Dim l As ToolTipDetail
        Dim k As Integer
        iFileNum = FreeFile
        Open "c:\Program Files\Con_sys\ToolTip.sit" For Random As #iFileNum Len = 45
        For n = 1 To 16
            Get #iFileNum, n, ReadTooltipText(n)
            If ReadTooltipText(n).PortUsed = 1 And ReadTooltipText(n).PortDetail <> "" Then
                chkport(n).Value = 1
                txtรายละเอียด(n).Text = ReadTooltipText(n).PortDetail
            Else
                chkport(n).Value = 0
                txtรายละเอียด(n).Text = ""
            End If
        Next n
        Close #iFileNum
End Sub
Private Sub cmdอิสระ_Click()
        frmเปลี่ยนไอคอน.Show
End Sub
Private Sub Form_Load()
        TapStrip1Selected
        lblคำนำ.Caption = "เลือกรายละเอียดเกี่ยวกับรูปภาพที่ต้องการใช้เป็นสัญลักษณ์" & Chr(10) + Chr(13) _
                        & "แทนการทำงานของอุปกรณ์ไฟฟ้า"
        If chkรูปอิสระ.Value = 1 Then
            chkกำหนดรูป.Enabled = False
        Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            chkกำหนดครูป.Enabled = True
        End If
End Sub
Private Sub optล้างทั้งหมด_Click()
        Dim n As Integer
        For n = 1 To 16
                chkport(n).Value = 0
                txtรายละเอียด(n).Text = ""
        Next n
End Sub
Private Sub optเลือกทั้งหมด_Click()
        Dim n As Integer
        For n = 1 To 16
                chkport(n).Value = 1
                txtรายละเอียด(n).Text = ""
        Next n
End Sub
Private Sub optอัตโนมัติ_Click()
        Dim n As Integer
        For n = 1 To 16
                chkport(n).Value = 1
                txtรายละเอียด(n).Text = "เอาท์พุทพอร์ท  " & Format(n, "00")
        Next n
End Sub
Private Sub TabStrip1_Click()
        TapStrip1Change = True
        If TabStrip1.Tabs(1).Selected Then
                TapStrip1Selected
        ElseIf TabStrip1.Tabs(2).Selected Then
                TapStrip2Selected
        End If
End Sub
```

frmเปลี่ยรหัสผ่าน

```
Private Sub cmdok_Click()
      Dim iFileNum As Integer
      Dim n As Integer
      n = Len(txtรหัสใหม่.Text)
      If txtรหัสเก่า.Text = password And txtรหัสใหม่.Text = txtยืนยัน.Text Then
            iFileNum = FreeFile
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Open "c:\Program Files\Con_sys\Password.ini" For Output As #iFileNum
            Print #iFileNum, n & txtรหัสใหม่.Text
            Print #iFileNum, txtNewName.Text
            MsgBox "เปลี่ยนรหัสสำเร็จ....รหัสจะใช้ได้ในคราวต่อไป"
            Close #iFileNum
    Else
            MsgBox "ไม่สามารถเปลี่ยนได้เนื่องจากรหัสเดิมไม่ถูกต้องหรือรหัสใหม่ไม่ตรงกัน"
    End If
    Unload Me
End Sub
```

frmเปลี่ยนไอคอน

```
Option Explicit
Dim AddressOn(1 To 16) As String
Dim AddressOff(1 To 16) As String
Private Sub cmdPortNum_Click(Index As Integer)
        With CommonDialog1
        .DialogTitle = "เลือกไอคอนที่หมายถึงการทำงาน"
        .Filter = "รูปภาพ(*.ico)|*.ico"
        .FileName = ""
        .ShowOpen
        AddressOn(Index) = .FileName
        txtAddressOn(Index).Text = .FileName
        .DialogTitle = "เลือกไอคอนที่หมายถึงการ ไม่ทำงาน"
        .FileName = ""
        .ShowOpen
        AddressOff(Index) = .FileName
        txtAddressOff(Index).Text = .FileName
        End With
End Sub
Private Sub cmdตกลง_Click()
        Dim n As Integer
        Dim iFileNum As Integer
        Dim Address As PrivateAddress
        iFileNum = FreeFile
        Open "c:\Program Files\Con_sys\IconAddress.sit" For Random As #iFileNum Len = 162
        For n = 1 To 16
                If AddressOff(n) <> "" And AddressOn(n) <> "" Then
                        Address.AdOn = AddressOn(n)
                        Address.AdOff = AddressOff(n)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                Address.EndLine = Chr(13) & Chr(10)
            End If
            Put #iFileNum, n, Address
        Next n
        Close #iFileNum
        Unload Me
End Sub
Private Sub cmdยกเลิก_Click()
    Unload Me
End Sub
Private Sub Form_Load()
    Dim n As Integer
    Dim Address(1 To 16) As PrivateAddress
    Dim iFileNum As Integer
    iFileNum = FreeFile
    Open "c:\Program Files\Con_sys\IconAddress.sit" For Random As #iFileNum Len = 162
    For n = 1 To 16
        Get #iFileNum, n, Address(n)
        AddressOn(n) = Address(n).AdOn
        txtAddressOn(n).Text = AddressOn(n)
        AddressOff(n) = Address(n).AdOff
        txtAddressOff(n).Text = AddressOff(n)
    Next n
End Sub
```

frmรหัสผ่าน

```
Private Sub cmdตกลง_Click()
    If txtPass.Text = password And txtName = YourName Then
        MDIForm1.Show
        MDIForm1.Enabled = True
        Unload Me
        Unload frmCON_SYS
    Else
        txtPass = ""
        txtPass.SetFocus
    End If
End Sub
Private Sub Form_Load()
    Dim pass As String
    Dim Passname As String
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Dim iFileNum As Integer
    Dim n As Integer
    iFileNum = FreeFile
    Open "c:\Program Files\Con_sys\Password.ini" For Input As #iFileNum
     Line Input #iFileNum, pass
    n = Mid(pass, 1, 1)
    password = Mid(pass, 2, n)
    Line Input #iFileNum, Passname
    YourName = Passname
    Close #iFileNum
End Sub
```

MDIForm1

```
Dim frmกำหนด1 As frmกำหนดเอง1
Sub บันทึกตำแหน่ง()
     Dim iFileNum As Integer
     Dim xxx(1 To 16) As Position
     Dim n As Integer
      Set frmกำหนด1 = frmกำหนดเอง1
      If mnu1เปลี่ยนตำแหน่งอุปกรณ์.Checked = True Or mnuเปลี่ยนตำแหน่งอุปกรณ์.Checked = True Then
             mnu1เปลี่ยนตำแหน่งอุปกรณ์.Checked = False
             mnuเปลี่ยนตำแหน่งอุปกรณ์.Checked = False
     Else
             mnu1เปลี่ยนตำแหน่งอุปกรณ์.Checked = True
             mnuเปลี่ยนตำแหน่งอุปกรณ์.Checked = True
     End If
      StatusBar1.Panels(1) = "กำลังบันทึกตำแหน่ง"
     iFileNum = FreeFile
     Open "c:\Program Files\Con_sys\position.ini" For Random As #iFileNum Len = 17
     For n = 1 To 16
             xxx(n).PortNumber = Format(n, "00")
             xxx(n).PortOnOff = OnOff(n)
             xxx(n).PortXaxis = Format(frmกำหนด1.Image1(n).Left, "000000")
             xxx(n).PortYaxis = Format(frmกำหนด1.Image1(n).Top, "000000")
             xxx(n).EndLine = Chr(13) & Chr(10)
             Put #iFileNum, n, xxx(n)
      Next n
     Close #iFileNum
    IconMove = False
               For n = 1 To 16
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
frmกำหนด1.Image1(n).DragMode = 0
        Next n
    mnuบันทึกตำแหน่งอุปกรณ์.Enabled = False
    mnu1บันทึกตำแหน่งอุปกรณ์.Enabled = False
    Toolbar1.Buttons(5).Image = 3
    StatusBar1.Panels(1) = "เข้สู่โหมดควบคุม"
End Sub
Sub เปลี่ยนตำแหน่ง()
    Dim n As Integer
    mnuบันทึกตำแหน่งอุปกรณ์.Enabled = True
    mnu1บันทึกตำแหน่งอุปกรณ์.Enabled = True
    Toolbar1.Buttons(5).Image = 5
    If mnu1เปลี่ยนตำแหน่งอุปกรณ์.Checked = True Or mnuเปลี่ยนตำแหน่งอุปกรณ์.Checked = True Then
        mnu1เปลี่ยนตำแหน่งอุปกรณ์.Checked = False
        mnuเปลี่ยนตำแหน่งอุปกรณ์.Checked = False
    Else
        mnu1เปลี่ยนตำแหน่งอุปกรณ์.Checked = True
        mnuเปลี่ยนตำแหน่งอุปกรณ์.Checked = True
    End If
    Set frmกำหนด1 = frmกำหนดเอง1
    If mnu1เปลี่ยนตำแหน่งอุปกรณ์.Checked = True Or mnuเปลี่ยนตำแหน่งอุปกรณ์.Checked = True Or
Toolbar1.Buttons(4).Enabled = True Then
            StatusBar1.Panels(1) = "ท่านกำลังอยู่ในโหมดเปลี่ยนตำแหน่ง..ท่านจะไม่สามารถทำการควบคุมได้จนกว่าจะทำ
การบันทึกตำแหน่ง"
            IconMove = True
            For n = 1 To 16
                frmกำหนด1.Image1(n).DragMode = 1
            Next n
    Else
            StatusBar1.Panels(1) = "เข้าสู่โหมดควบคุม"
            IconMove = False
            For n = 1 To 16
                frmกำหนด1.Image1(n).DragMode = 0
            Next n
    End If
End Sub
Private Sub MDIForm_Click()
    ProgressBar1.Value = 0
End Sub
Private Sub MDIForm_Load()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Unload frmCON_SYS
        Unload frmรหัสผ่าน
        Dim n As Integer
        MDIForm1.Top = 0
        MDIForm1.Left = 0
        If YourName <> "" Then
            StatusBar1.Panels(2) = YourName
            StatusBar1.Panels(2).ToolTipText = "เจ้าของรหัสผ่าน"
        Else
            StatusBar1.Panels(2) = "Sittichai  Chosalid"
            StatusBar1.Panels(2).ToolTipText = "เจ้าของลิขสิทธิ์"
        End If
        frmกำหนดเอง1.Show
End Sub
Private Sub mnu1ขอดูสถานะของอุปกรณ์ขณะนี้_Click()
        Dim n As Integer
        Set frmกำหนด1 = frmกำหนดเอง1
        DataStatus
        For n = 1 To 16
                frmกำหนด1.Image1(n) = frmกำหนด1.Image3(n)
                OnOff(n) = 0
        Next n
        RealDataStatus
        frmกำหนด1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่ทำงาน  " & i & "  อุปกรณ์."
        MDIForm1.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)..." & "จำนวนอุปกรณ์ที่ทำงาน  " & i & "  อุปกรณ์."
End Sub
Private Sub mnu1ตกลงที่จะควบคุมอุปกรณ์_Click()
        Set frmกำหนด1 = frmกำหนดเอง1
        frmกำหนด1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่เลือก  " & i & "  อุปกรณ์."
        MDIForm1.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)..." & "จำนวนอุปกรณ์ที่เลือก  " & i & "  อุปกรณ์."
        StatusBar1.Panels(1) = "ท่านกำลังส่งข้อมูลผ่านพอร์ตอนุกรมเพื่อควบคุม"
        frmกำหนด1.ControllingStatus
        SendData i
        StatusBar1.Panels(1) = ""
End Sub
Private Sub mnu1บันทึกตำแหน่งอุปกรณ์_Click()
        บันทึกตำแหน่ง
End Sub
Private Sub mnu1เปลี่ยนตำแหน่งอุปกรณ์_Click()
        เปลี่ยนตำแหน่ง
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub
Private Sub mnu1เปลี่ยไอคอน_Click()
        frmเปลี่ยนไอคอน.Show
End Sub
Private Sub mnuกำหนดเอง_Click()
        frmกำหนดเอง1.Show
End Sub
Private Sub mnuควบคุม_Click()
        Set frmกำหนด1 = frmกำหนดเอง1
        frmกำหนด1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่เลือก " & i & " อุปกรณ์."
        MDIForm1.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)..." & "จำนวนอุปกรณ์ที่เลือก " & i & " อุปกรณ์."
        StatusBar1.Panels(1) = "ท่านกำลังส่งข้อมูลผ่านพอร์ตอนุกรมเพื่อควบคุม"
        frmกำหนด1.ControllingStatus
        SendData i
End Sub
Private Sub mnuคุณสมบัติ_Click()
        frmคุณสมบัติ.Show
End Sub
Private Sub mnuจบการทำงาน_Click()
        StatusBar1.Panels(1) = "ท่านกำลังออกจากโปรแกรม กรุณายืนยัน"
        จบการทำงาน
        StatusBar1.Panels(1) = ""
End Sub
Private Sub mnuดูสถานะ_Click()
        For n = 1 To 16
                frmกำหนด1.Image1(n).Picture = frmกำหนด1.Image3(n).Picture
                OnOff(n) = 0
        Next n
        DataStatus
        RealDataStatus
        frmกำหนด1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่ทำงาน " & i & " อุปกรณ์."
        MDIForm1.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)....." & "จำนวนอุปกรณ์ที่ทำงาน " & i & " อุปกรณ์."
End Sub
Private Sub mnuบันทึกตำแหน่งอุปกรณ์_Click()
        บันทึกตำแหน่ง
End Sub
Private Sub mnuเปลี่ยนตำแหน่งอุปกรณ์_Click()
        เปลี่ยนตำแหน่ง
End Sub
Private Sub mnuเปลี่ยนรหัสผ่าน_Click()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        frmเปลี่ยนรหัสผ่านใหม่.Show
End Sub
Private Sub mnuเปลี่ยนไอคอน_Click()
        frmเปลี่ยนไอคอน.Show
End Sub
Private Sub mnuพอร์ตอนุกรม1_Click()
    MSComm1.CommPort = 1
    mnuพอร์ตอนุกรม1.Checked = True
    mnuพอร์ตอนุกรม2.Checked = False
End Sub
Private Sub mnuพอร์ตอนุกรม2_Click()
    MSComm1.CommPort = 2
    mnuพอร์ตอนุกรม2.Checked = True
    mnuพอร์ตอนุกรม1.Checked = False
End Sub
Private Sub mnuไม่เลือกทั้งหมด_Click()
    Dim n As Integer
    Set frmกำหนด1 = frmกำหนดเอง1
    For n = 1 To 16
        frmกำหนด1.Image1(n) = frmกำหนด1.Image3(n)
        OnOff(n) = 0
    Next n
    i = 0
End Sub
Private Sub mnuเริ่มใหม่_Click()
    Set frmกำหนด1 = frmกำหนดเอง1
    NewNew = True
    Unload frmกำหนดเอง1
    frmคุณสมบัติ.Show
End Sub
Private Sub mnuเลือก8ชุดแรก_Click()
    Dim n As Integer
    Set frmกำหนด1 = frmกำหนดเอง1
    For n = 1 To 16
        If n <= 8 Then
            frmกำหนด1.Image1(n) = frmกำหนด1.Image2(n)
            OnOff(n) = 1
        Else
            frmกำหนด1.Image1(n) = frmกำหนด1.Image3(n)
            OnOff(n) = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            End If
        Next n
        i = 8
End Sub
Private Sub mnuเลือก8ชุดหลัง_Click()
        Dim n As Integer
        Set frmกำหนด1 = frmกำหนดเอง1
        For n = 1 To 16
            If n > 8 Then
                frmกำหนด1.Image1(n) = frmกำหนด1.Image2(n)
                OnOff(n) = 1
            Else
                frmกำหนด1.Image1(n) = frmกำหนด1.Image3(n)
                OnOff(n) = 0
            End If
        Next n
        i = 8
End Sub
Private Sub mnuเลือกทั้งหมด_Click()
        Dim n As Integer
        Set frmกำหนด1 = frmกำหนดเอง1
        For n = 1 To 16
            frmกำหนด1.Image1(n) = frmกำหนด1.Image2(n)
            OnOff(n) = 1
        Next n
        i = 16
End Sub
Private Sub mnuวิธีใช้_Click()
    frmHelp.Show
End Sub
Private Sub Toolbar1_ButtonClick(ByVal Button As MSComctlLib.Button)
        Set frmกำหนด1 = frmกำหนดเอง1
        Dim n As Integer
        Select Case Button.Index
            Case 1
                    MDIForm1.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)....." & "จำนวนอุปกรณ์ที่เลือก  " & i & "
อุปกรณ์."
                    frmกำหนด1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่เลือก  " & i & "  อุปกรณ์."
                    StatusBar1.Panels(1) = "ท่านกำลังส่งข้อมูลผ่านพอร์ตอนุกรมเพื่อควบคุม"
                    frmกำหนด1.ControllingStatus
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SendData i
            Case 2
                For n = 1 To 16
                    frmกำหนด1.Image1(n).Picture = frmกำหนด1.Image3(n).Picture
                    OnOff(n) = 0
                Next n
                DataStatus
                RealDataStatus
                frmกำหนด1.Caption = "แผนผังแสดงการควบคุม....." & "จำนวนอุปกรณ์ที่ทำงาน " & i & " อุปกรณ์."
                MDIForm1.Caption = " โปแกรมควบคุมอุปกรณ์ไฟฟ้า(...รุ่น2.0)......" & "จำนวนอุปกรณ์ที่ทำงาน " & i & "
อุปกรณ์."
            Case 4
                เปลี่ยนตำแหน่ง
            Case 5
                บันทึกตำแหน่ง
        End Select
End Sub
```

Modules1

```
Global YourName As String
Global KeepMyData(1 To 16) As Integer
Global Data(1 To 16) As Integer
Global i As Integer
Global Number As Integer
Global Quantum As Integer
Global StatusLoad(1 To 16) As String
Global QuantityLoad As Integer
Global password As String
Global xPosition(1 To 8)
Global yPosition(1 To 8)
Global IconMove As Boolean
Global NewNew As Boolean
Global OnOff(1 To 16) As Integer
Type Position
    PortNumber As String * 2
    PortOnOff As String * 1
    PortXaxis As String * 6
    PortYaxis As String * 6
    EndLine As String * 2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Type
Type ToolTipDetail
        PortNumber As String * 2
        PortUsed As String * 1
        PortDetail As String * 40
        EndLine As String * 2
End Type
Type PrivateAddress
        AdOn As String * 80
        AdOff As String * 80
        EndLine As String * 2
End Type
Sub KeepDataForSend(Keep As Integer, comp As Integer)
        Dim k As Integer
        If comp = 1 Then
                For k = 1 To 16
                        If Keep = k Then
                                KeepMyData(k) = 16 + k
                                Exit For
                        End If
                Next k
        ElseIf comp = 0 Then
                For k = 1 To 16
                        If k = Keep Then
                        KeepMyData(k) = nul                    'ลบข้อมูล
                        Exit For
                        End If
                Next k
        End If
End Sub
Sub SendData(Quantity As Integer)
        Dim buf
        Dim m As Integer
        Dim mdifrm1 As MDIForm1
        Set mdifrm1 = MDIForm1
        mdifrm1.ProgressBar1.Value = 2
        mdifrm1.StatusBar1.Panels(1).Text = "กำลังติดต่อกับอุปกรณ์ควบคุม"
        With mdifrm1.MSComm1
        mdifrm1.ProgressBar1.Value = 3
        If .PortOpen <> True Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            .PortOpen = True            'เปิดport com1
            mdifrm1.ProgressBar1.Value = 4
        End If
        mdifrm1.ProgressBar1.Value = 5
        buf = Chr(Number)                               'Number
        .Output = buf
        mdifrm1.ProgressBar1.Value = 6
        buf = Chr(Quantity)             'จำนวนที่ on
        .Output = buf
        mdifrm1.ProgressBar1.Value = 7
        buf = Chr(170)                          'AAH
        .Output = buf
        mdifrm1.ProgressBar1.Value = 8
        For m = 1 To 16
            If KeepMyData(m) <> nul Then
                buf = Chr(KeepMyData(m))
                .Output = buf
                mdifrm1.ProgressBar1.Value = 9
            End If
        Next m
        mdifrm1.ProgressBar1.Value = 10
        If .PortOpen <> False Then
            .PortOpen = False
        End If
        mdifrm1.ProgressBar1.Value = 0
        .OutBufferCount = 0
        End With
End Sub
Sub DataStatus()
        Dim a, m As Integer, n As Integer, q As Integer
        Dim dOut(1 To 3)
        Dim dCode(1 To 3)
        Dim mdifrm1 As MDIForm1
        Set mdifrm1 = MDIForm1
line:
        mdifrm1.StatusBar1.Panels(1).Text = "กำลังติดต่อกับอุปกรณ์ควบคุม"
        mdifrm1.ProgressBar1.Value = 2
        With mdifrm1.MSComm1
        If .PortOpen <> True Then                       'เปิดพอร์ทอนุกรม
            .PortOpen = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If
            'กำหนดและส่งรหัสเพื่อขอคูข้อมูล
dOut(1) = Chr(Number)                          '01H
dOut(2) = Chr(1)
dOut(3) = Chr(204)                        'CCH
mdifrm1.ProgressBar1.Value = 4
For m = 1 To 3
    .Output = dOut(m)
Next m
'*************************************************
.InBufferCount = 0
            'กำหนดคุณสมบัติการรับข้อมูล
.InputMode = comInputModeText
Do Until .InBufferCount <> 0: Loop
mdifrm1.ProgressBar1.Value = 6
.InputLen = 1          'รับครั้งละ 8 บิท
For p = 1 To 3
            dCode(p) = .Input
Next p
mdifrm1.ProgressBar1.Value = 8
If Val(Asc(dCode(1))) <> 0 And Val(Asc(dCode(2))) <> Number Then
            InBufferCount = 0
            MsgBox "เกิดการผิดพลาด กด Yes เพื่อทำซ้ำ กด No เพื่อยกเลิก", vbYesNo + vbCritical, "Error"
            GoTo line:                    'ถ้าไม่ใช่ dCode ที่ต้องการกลับไปรอรับใหม่
End If
QuantityLoad = Val(Asc(dCode(3)))
i = QuantityLoad
If Val(Asc(dCode(3))) <> 255 Then
            Do Until .InBufferCount = Val(Asc(dCode(3))): Loop
            mdifrm1.ProgressBar1.Value = 9
            For n = 1 To Val(Asc(dCode(3)))
                    StatusLoad(n) = Str(Val(Asc(.Input)))
            Next n
Else
            For n = 1 To 16
                    StatusLoad(n) = 0
            Next n
End If
mdifrm1.ProgressBar1.Value = 10
.InBufferCount = 0                    'เคลียร์อินพุทบัฟเฟอร์
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        If .PortOpen <> False Then                          'เปิดพอร์ทอนุกรม
                .PortOpen = False
        End If
        End With
        mdifrm1.ProgressBar1.Value = 0
Line2:
End Sub
Sub RealDataStatus()
        Dim q As Integer, n As Integer
        Dim กำหนดเอง1 As frmกำหนดเอง1
        Set กำหนดเอง1 = frmกำหนดเอง1
        i = 0
        For n = 1 To 16
                Data(n) = 16 + n
        Next n
        If QuantityLoad <> 255 Then
          For n = 1 To QuantityLoad
                For q = 1 To 16
                        If StatusLoad(n) = Data(q) Then
                                กำหนดเอง1.Image1(q) = กำหนดเอง1.Image2(q)
                                OnOff(q) = 1
                                i = i + 1
                        End If
                Next q
          Next n
        Else
          For q = 1 To 16
                กำหนดเอง1.Image1(q) = กำหนดเอง1.Image3(q)
                OnOff(q) = 0
                If i <> 0 Then
                      i = i - 1
                Else
                      i = 0
                End If
          Next q
        End If
End Sub
Sub จบการทำงาน()
        Dim a
        a = MsgBox("ท่านต้องการจบการทำงานใช่หรือไม่?", vbOKCancel + vbQuestion, "จบการทำงาน")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        If a = vbOK Then
            End
        End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของโปรแกรม MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11.asm

```
0000                    ORG 0000H

0000 020050             LJMP MAIN


0050                    ORG 0050H


0050 758700      MAIN:  MOV PCON,#00H        ;CLEAR SMOD
0053 759850             MOV SCON,#50H        ;SERIAL MODE 1
0056 758920             MOV TMOD,#20H        ;TIMER1 MODE 2
0059 758DFD             MOV TH1,#0FDH        ;BAUD RATE 9600 BIT/SEC
005C 75900F             MOV P1,#0FH
005F 75A000             MOV P2,#00H


0062 D2B2               SETB P3.2
0064 D28E               SETB TR1
0066 D288               SETB IT0             ;ACTIVE HIGHT TO LOW
0068 D2BC               SETB PS              ;SET PARITY SERIAL  PORT
006A D2AF               SETB EA              ;ENABLE INTERRUPT
006C D2A8               SETB EX0             ;ENABLE INTERRUPT
006E D2AC               SETB ES              ;ENABLE INTERRUPT
0070 C2AB               CLR ET1


0072 1204BA             CALL DELY

0075 90A003             MOV DPTR,#0A003H
0078 7482               MOV A,#82H           ;8255 ONE A OUT,B IN,C OUT
007A F0                 MOVX @DPTR,A

007B 90C003             MOV DPTR,#0C003H
007E 748B               MOV A,#8BH           ;8255 TWO A OUT,B IN,C IN
0080 F0                 MOVX @DPTR,A

0081 020081             LJMP $



                 ;RECIVE INTERRUPT INT0 ADDRESS 0003H

0003                    ORG 0003H

0003 C2AF               CLR EA

0005 020300             LJMP KEY



                 ;RECIVE INTERRUPT INT0 ADDRESS 0003H

0300                    ORG 0300H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0300 7850            KEY:  MOV R0,#50H ;RECIVE VALUE FORM DMT TELLPHONE
0302 7951                  MOV R1,#51H

0304 A690                  MOV @R0,P1
0306 1204C7                LCALL W0

0309 B60A03          ARUN1:CJNE @R0,#0AH,L2

030C 020312          L1:   LJMP NUM1
030F 0203FD          L2:   LJMP ARUN2

0312 20B2FD          NUM1: JB P3.2,$
0315 A790                  MOV @R1,P1

0317 7400                  MOV A,#00H
0319 90A001                MOV DPTR,#0A001H
031C E0                    MOVX A,@DPTR

031D B70115                CJNE @R1,#01H,KEY2

0320 30E009                JNB 0E0H,Y1
0323 C2E0                  CLR 0E0H
0325 90A000                MOV DPTR,#0A000H
0328 F0                    MOVX @DPTR,A

0329 0203F6                LJMP ERROR1

032C D2E0            Y1:   SETB 0E0H
032E 90A000                MOV DPTR,#0A000H
0331 F0                    MOVX @DPTR,A

0332 0203F6                LJMP ERROR1

0335 B70215          KEY2: CJNE @R1,#02H,KEY3

0338 30E109                JNB 0E1H,Y2
033B C2E1                  CLR 0E1H
033D 90A000                MOV DPTR,#0A000H
0340 F0                    MOVX @DPTR,A

0341 0203F6                LJMP ERROR1

0344 D2E1            Y2:   SETB 0E1H
0346 90A000                MOV DPTR,#0A000H
0349 F0                    MOVX @DPTR,A

034A 0203F6                LJMP ERROR1

034D B70315          KEY3: CJNE @R1,#03H,KEY4

0350 30E209                JNB 0E2H,Y3
0353 C2E2                  CLR 0E2H
0355 90A000                MOV DPTR,#0A000H
0358 F0                    MOVX @DPTR,A

0359 0203F6                LJMP ERROR1

035C D2E2            Y3:   SETB 0E2H
035E 90A000                MOV DPTR,#0A000H
0361 F0                    MOVX @DPTR,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0362 0203F6              LJMP ERROR1
0365 B70415      KEY4:   CJNE @R1,#04H,KEY5

0368 30E309              JNB 0E3H,Y4
036B C2E3                CLR 0E3H
036D 90A000              MOV DPTR,#0A000H
0370 F0                  MOVX @DPTR,A

0371 0203F6              LJMP ERROR1

0374 D2E3        Y4:     SETB 0E3H
0376 90A000              MOV DPTR,#0A000H
0379 F0                  MOVX @DPTR,A

037A 0203F6              LJMP ERROR1

037D B70515      KEY5:   CJNE @R1,#05H,KEY6

0380 30E409              JNB 0E4H,Y5
0383 C2E4                CLR 0E4H
0385 90A000              MOV DPTR,#0A000H
0388 F0                  MOVX @DPTR,A

0389 0203F6              LJMP ERROR1

038C D2E4        Y5:     SETB 0E4H
038E 90A000              MOV DPTR,#0A000H
0391 F0                  MOVX @DPTR,A

0392 0203F6              LJMP ERROR1

0395 B70615      KEY6:   CJNE @R1,#06H,KEY7

0398 30E509              JNB 0E5H,Y6
039B C2E5                CLR 0E5H
039D 90A000              MOV DPTR,#0A000H
03A0 F0                  MOVX @DPTR,A

03A1 0203F6              LJMP ERROR1

03A4 D2E5        Y6:     SETB 0E5H
03A6 90A000              MOV DPTR,#0A000H
03A9 F0                  MOVX @DPTR,A

03AA 0203F6              LJMP ERROR1

03AD B70715      KEY7:   CJNE @R1,#07H,KEY8

03B0 30E609              JNB 0E6H,Y7
03B3 C2E6                CLR 0E6H
03B5 90A000              MOV DPTR,#0A000H
03B8 F0                  MOVX @DPTR,A

03B9 0203F6              LJMP ERROR1

03BC D2E6        Y7:     SETB 0E6H
03BE 90A000              MOV DPTR,#0A000H
03C1 F0                  MOVX @DPTR,A

03C2 0203F6              LJMP ERROR1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
03C5 B70815      KEY8: CJNE @R1,#08H,KEY9
03C8 30E709            JNB 0E7H,Y8
03CB C2E7              CLR 0E7H
03CD 90A000            MOV DPTR,#0A000H
03D0 F0                MOVX @DPTR,A

03D1 0203F6            LJMP ERROR1

03D4 D2E7        Y8:   SETB 0E7H
03D6 90A000            MOV DPTR,#0A000H
03D9 F0                MOVX @DPTR,A

03DA 0203F6            LJMP ERROR1

03DD B70916      KEY9: CJNE @R1,#09H,ERROR1

03E0 90C001            MOV DPTR,#0C001H
03E3 E0                MOVX A,@DPTR

03E4 30E009            JNB 0E0H,Y9
03E7 C2E0              CLR 0E0H
03E9 90C000            MOV DPTR,#0C000H
03EC F0                MOVX @DPTR,A

03ED 0203F6            LJMP ERROR1

03F0 D2E0        Y9:   SETB 0E0H
03F2 90C000            MOV DPTR,#0C000H
03F5 F0                MOVX @DPTR,A

03F6 76FF        ERROR1:MOV @R0,#0FFH
03F8 77FF              MOV @R1,#0FFH

03FA D2AF              SETB EA
03FC 32                RETI


03FD B601F6      ARUN2:CJNE @R0,#01H,ERROR1

0400 20B2FD            JB P3.2,$
0403 A790              MOV @R1,P1

0405 90C001            MOV DPTR,#0C001H
0408 E0                MOVX A,@DPTR

0409 B70A15      KEY10:CJNE @R1,#0AH,KEY11

040C 30E109            JNB 0E1H,Y10
040F C2E1              CLR 0E1H
0411 90C000            MOV DPTR,#0C000H
0414 F0                MOVX @DPTR,A

0415 0203F6            LJMP ERROR1

0418 D2E1        Y10:  SETB 0E1H
041A 90C000            MOV DPTR,#0C000H
041D F0                MOVX @DPTR,A

041E 0203F6            LJMP ERROR1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0421 B70115        KEY11:CJNE @R1,#01H,KEY12

0424 30E209              JNB 0E2H,Y11
0427 C2E2                CLR 0E2H
0429 90C000              MOV DPTR,#0C000H
042C F0                  MOVX @DPTR,A

042D 0203F6              LJMP ERROR1

0430 D2E2          Y11:  SETB 0E2H
0432 90C000              MOV DPTR,#0C000H
0435 F0                  MOVX @DPTR,A

0436 0203F6              LJMP ERROR1

0439 B70215        KEY12:CJNE @R1,#02H,KEY13

043C 30E309              JNB 0E3H,Y12
043F C2E3                CLR 0E3H
0441 90C000              MOV DPTR,#0C000H
0444 F0                  MOVX @DPTR,A

0445 0203F6              LJMP ERROR1

0448 D2E3          Y12:  SETB 0E3H
044A 90C000              MOV DPTR,#0C000H
044D F0                  MOVX @DPTR,A

044E 0203F6              LJMP ERROR1


0451 B70315        KEY13:CJNE @R1,#03H,KEY14

0454 30E409              JNB 0E4H,Y13
0457 C2E4                CLR 0E4H
0459 90C000              MOV DPTR,#0C000H
045C F0                  MOVX @DPTR,A

045D 0203F6              LJMP ERROR1

0460 D2E4          Y13:  SETB 0E4H
0462 90C000              MOV DPTR,#0C000H
0465 F0                  MOVX @DPTR,A

0466 0203F6              LJMP ERROR1

0469 B70415        KEY14:CJNE @R1,#04H,KEY15

046C 30E509              JNB 0E5H,Y14
046F C2E5                CLR 0E5H
0471 90C000              MOV DPTR,#0C000H
0474 F0                  MOVX @DPTR,A

0475 0203F6              LJMP ERROR1

0478 D2E5          Y14:  SETB 0E5H
047A 90C000              MOV DPTR,#0C000H
047D F0                  MOVX @DPTR,A

047E 0203F6              LJMP ERROR1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        
0481 B70515        KEY15:CJNE @R1,#05H,KEY16

0484 30E609              JNB 0E6H,Y15
0487 C2E6                CLR 0E6H
0489 90C000              MOV DPTR,#0C000H
048C F0                  MOVX @DPTR,A

048D 0203F6              LJMP ERROR1

0490 D2E6          Y15:  SETB 0E6H
0492 90C000              MOV DPTR,#0C000H
0495 F0                  MOVX @DPTR,A

0496 0203F6              LJMP ERROR1

0499 B70612        KEY16:CJNE @R1,#06H,ERROR2

049C 30E709              JNB 0E7H,Y16
049F C2E7                CLR 0E7H
04A1 90C000              MOV DPTR,#0C000H
04A4 F0                  MOVX @DPTR,A

04A5 0203F6              LJMP ERROR1

04A8 D2E7          Y16:  SETB 0E7H
04AA 90C000              MOV DPTR,#0C000H
04AD F0                  MOVX @DPTR,A

04AE 0203F6        ERROR2:LJMP ERROR1



                   ;END OF INTERRUP EXTERNAL INT0


04B1 7800          DELAY1:MOV R0,#0
04B3 7900          DY1:  MOV R1,#0
04B5 D9FE                DJNZ R1,$

04B7 D8FA                DJNZ R0,DY1
04B9 22                  RET

04BA 7D0A          DELY: MOV R5,#10
04BC 7E00          DELY1:MOV R6,#00H
04BE 7FFF          DELY2:MOV R7,#0FFH
04C0 DFFE          DELY3:DJNZ R7,DELY3
04C2 DEFA                DJNZ R6,DELY2
04C4 DDF6                DJNZ R5,DELY
04C6 22                  RET
04C7 7D1F          W0:   MOV R5,#01FH
04C9 7C1F          W1:   MOV R4,#01FH
04CB 7BFF          W2:   MOV R3,#0FFH
04CD DBFE                DJNZ R3,$
04CF DCFA                DJNZ R4,W2
04D1 DDF6                DJNZ R5,W1
04D3 22                  RET

0000=                              END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

คุณสมบัติของอุปกรณ์ที่เกี่ยวข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Compatible with MCS-51™ Products
- 4 Kbytes of In-System Reprogrammable Flash Memory
  Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-Level Program Memory Lock
- 128 x 8-Bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-Bit Timer/Counters
- Six Interrupt Sources
- Programmable Serial Channel
- Low Power Idle and Power Down Modes

8-Bit Microcontroller with 4 Kbytes Flash

Description

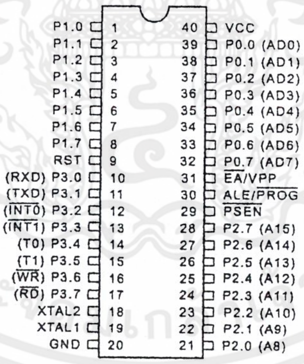
The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4 Kbytes of Flash Programmable and Erasable Read Only Memory (PEROM). The device is manufactured using Atmel's high density nonvolatile memory technology and is compatible with the industry standard MCS-51™ instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcomputer which provides a highly flexible and cost effective solution to many embedded control applications.

The AT89C51 provides the following standard features: 4 Kbytes of Flash, 128 bytes of RAM, 32 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator and clock circuitry. In addition, the AT89C51 is

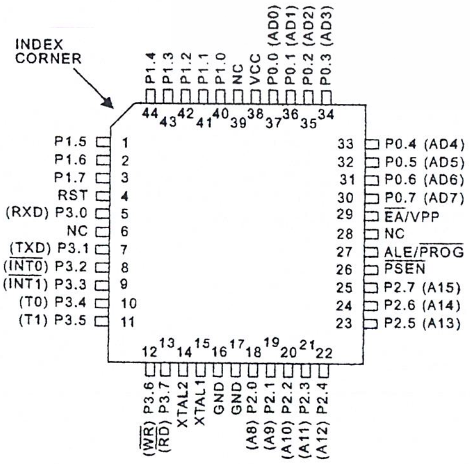
(continued)

Pin Configurations

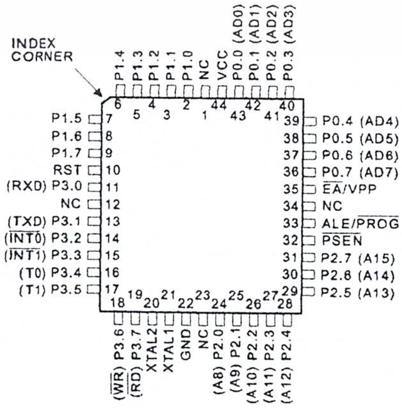
PDIP/Cerdip



PQFP/TQFP



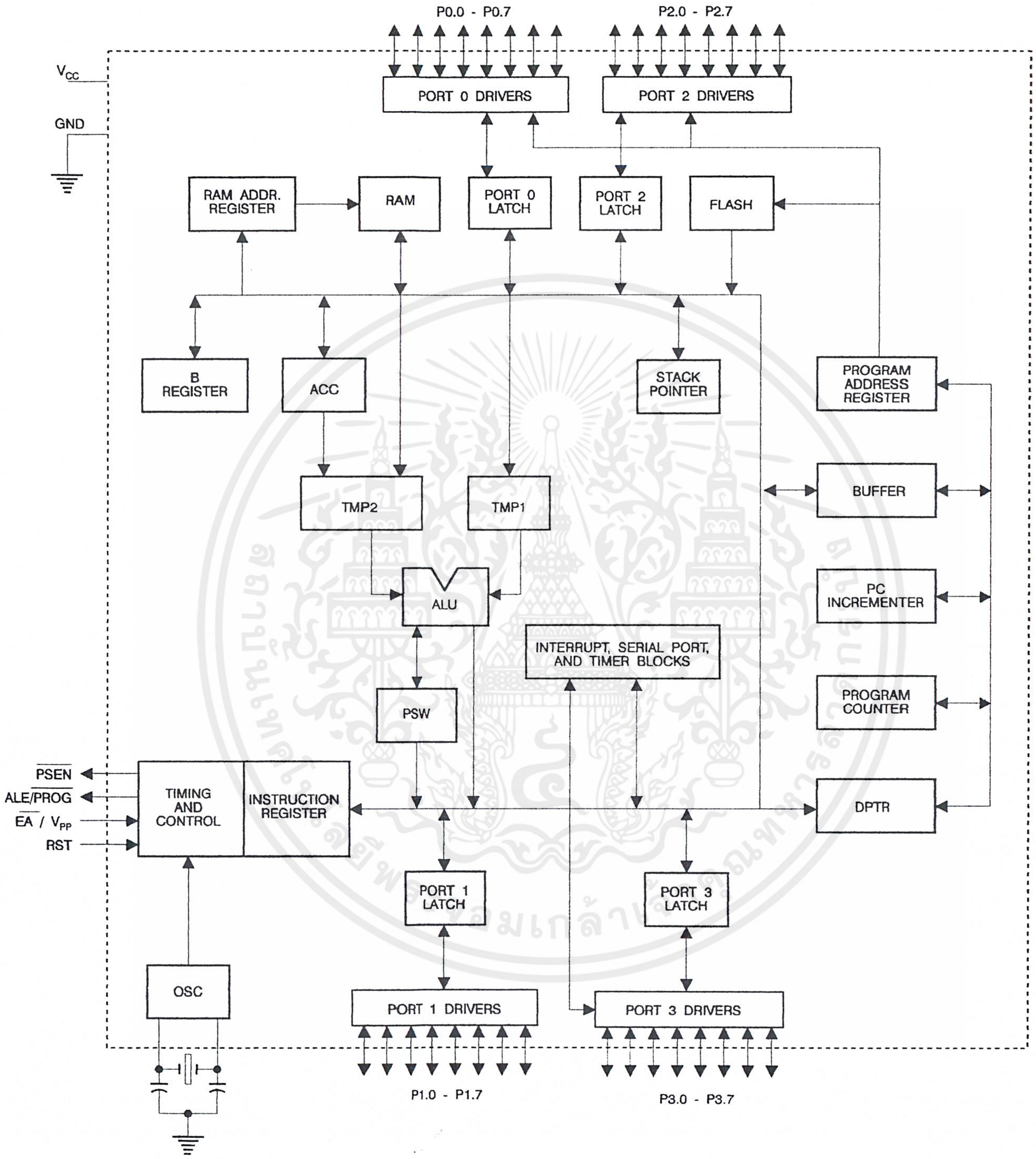
PLCC/LCC



0265E

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0 V to +7.0 V
Maximum Operating Voltage	6.6 V
DC Output Current	15.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. Characteristics

T_A = -40°C to 85°C, V_{CC} = 5.0 V ± 20% (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low Voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
V_{IL1}	Input Low Voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
V_{IH}	Input High Voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
V_{IH1}	Input High Voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
V_{OL}	Output Low Voltage(1) (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low Voltage(1) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High Voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5 V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High Voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5 V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45 V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2 V		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pulldown Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		20	mA
		Idle Mode, 12 MHz		5	mA
	Power Down Mode(2)	V_{CC} = 6 V		100	µA
		V_{CC} = 3 V		40	µA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
Maximum I_{OL} per port pin: 10 mA
Maximum I_{OL} per 8-bit port:
Port 0: 26 mA
Ports 1,2, 3: 15 mA
Maximum total IOL for all output pins: 71 mA
If IOL exceeds the test condition, VOL may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum VCC for Power Down is 2 V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. Characteristics

(Under Operating Conditions; Load Capacitance for Port 0, ALE/$\overline{\text{PROG}}$, and $\overline{\text{PSEN}}$ = 100 pF; Load Capacitance for all other outputs = 80 pF)

External Program and Data Memory Characteristics

Symbol	Parameter	12 MHz Oscillator		16 to 24 MHz Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	28		tCLCL-13		ns
t_{LLAX}	Address Hold After ALE Low	48		tCLCL-20		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		4tCLCL-65	ns
t_{LLPL}	ALE Low to $\overline{\text{PSEN}}$ Low	43		t_{CLCL}-13		ns
t_{PLPH}	$\overline{\text{PSEN}}$ Pulse Width	205		$3t_{CLCL}$-20		ns
t_{PLIV}	$\overline{\text{PSEN}}$ Low to Valid Instruction In		145		$3t_{CLCL}$-45	ns
t_{PXIX}	Input Instruction Hold After $\overline{\text{PSEN}}$	0		0		ns
t_{PXIZ}	Input Instruction Float After $\overline{\text{PSEN}}$		59		t_{CLCL}-10	ns
t_{PXAV}	$\overline{\text{PSEN}}$ to Address Valid	75		t_{CLCL}-8		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}$-55	ns
t_{PLAZ}	$\overline{\text{PSEN}}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{\text{RD}}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{WLWH}	$\overline{\text{WR}}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{RLDV}	$\overline{\text{RD}}$ Low to Valid Data In		252		$5t_{CLCL}$-90	ns
t_{RHDX}	Data Hold After $\overline{\text{RD}}$	0		0		ns
t_{RHDZ}	Data Float After $\overline{\text{RD}}$		97		$2t_{CLCL}$-28	ns
t_{LLDV}	ALE Low to Valid Data In		517		8tCLCL-150	ns
t_{AVDV}	Address to Valid Data In		585		9tCLCL-165	ns
t_{LLWL}	ALE Low to $\overline{\text{RD}}$ or $\overline{\text{WR}}$ Low	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{\text{RD}}$ or $\overline{\text{WR}}$ Low	203		$4t_{CLCL}$-75		ns
t_{QVWX}	Data Valid to $\overline{\text{WR}}$ Transition	23		t_{CLCL}-20		ns
t_{QVWH}	Data Valid to $\overline{\text{WR}}$ High	433		$7t_{CLCL}$-120		ns
t_{WHQX}	Data Hold After $\overline{\text{WR}}$	33		t_{CLCL}-20		ns
t_{RLAZ}	$\overline{\text{RD}}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{\text{RD}}$ or $\overline{\text{WR}}$ High to ALE High	43	123	t_{CLCL}-20	t_{CLCL}+25	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

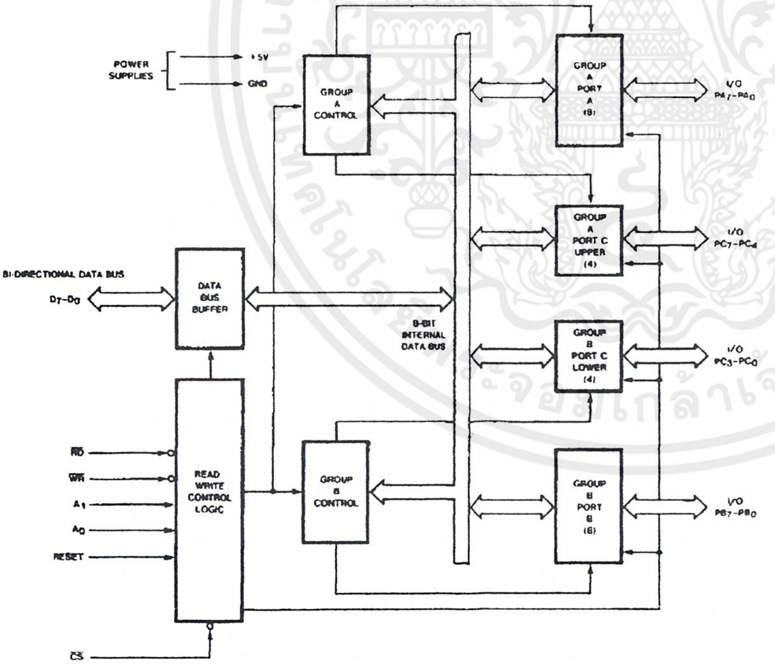
82C55A
CHMOS PROGRAMMABLE PERIPHERAL INTERFACE

- **Compatible with all Intel and Most Other Microprocessors**
- **High Speed, "Zero Wait State" Operation with 8 MHz 8086/88 and 80186/188**
- **24 Programmable I/O Pins**
- **Low Power CHMOS**
- **Completely TTL Compatible**
- **Control Word Read-Back Capability**
- **Direct Bit Set/Reset Capability**
- **2.5 mA DC Drive Capability on all I/O Port Outputs**
- **Available in 40-Pin DIP and 44-Pin PLCC**
- **Available in EXPRESS**
  - **—Standard Temperature Range**
  - **—Extended Temperature Range**

The Intel 82C55A is a high-performance, CHMOS version of the industry standard 8255A general purpose programmable I/O device which is designed for use with all Intel and most other microprocessors. It provides 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. The 82C55A is pin compatible with the NMOS 8255A and 8255A-5.

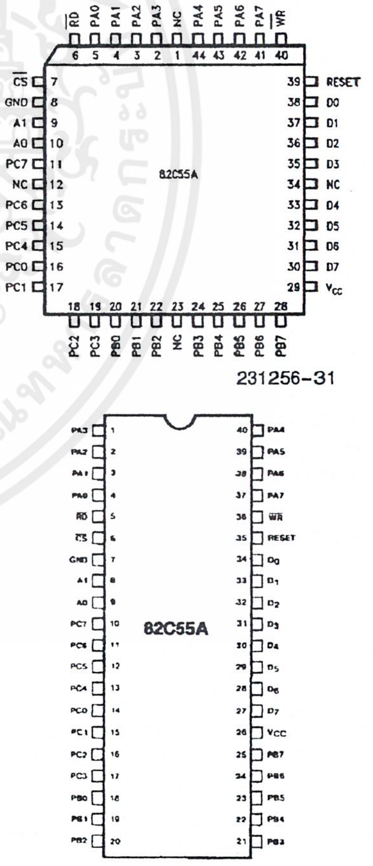
In MODE 0, each group of 12 I/O pins may be programmed in sets of 4 and 8 to be inputs or outputs. In MODE 1, each group may be programmed to have 8 lines of input or output. 3 of the remaining 4 pins are used for handshaking and interrupt control signals. MODE 2 is a strobed bi-directional bus configuration.

The 82C55A is fabricated on Intel's advanced CHMOS III technology which provides low power consumption with performance equal to or greater than the equivalent NMOS product. The 82C55A is available in 40-pin DIP and 44-pin plastic leaded chip carrier (PLCC) packages.



231256–1

Figure 1. 82C55A Block Diagram



231256–2

Figure 2. 82C55A Pinout

Diagrams are for pin reference only. Package sizes are not to scale.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1. Pin Description

<table>
<tr><th rowspan="2">Symbol</th><th colspan="2">Pin Number</th><th rowspan="2">Type</th><th rowspan="2">Name and Function</th></tr>
<tr><th>Dip</th><th>PLCC</th></tr>
<tr><td>PA_{3-0}</td><td>1–4</td><td>2–5</td><td>I/O</td><td>PORT A, PINS 0–3: Lower nibble of an 8-bit data output latch/buffer and an 8-bit data input latch.</td></tr>
<tr><td>$\overline{RD}$</td><td>5</td><td>6</td><td>I</td><td>READ CONTROL: This input is low during CPU read operations.</td></tr>
<tr><td>$\overline{CS}$</td><td>6</td><td>7</td><td>I</td><td>CHIP SELECT: A low on this input enables the 82C55A to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and WR are ignored otherwise.</td></tr>
<tr><td>GND</td><td>7</td><td>8</td><td></td><td>System Ground</td></tr>
<tr><td>A_{1-0}</td><td>8–9</td><td>9–10</td><td>I</td><td>ADDRESS: These input signals, in conjunction $\overline{RD}$ and $\overline{WR}$, control the selection of one of the three ports or the control word registers.
<table>
<tr><th>A_1</th><th>A_0</th><th>$\overline{RD}$</th><th>$\overline{WR}$</th><th>$\overline{CS}$</th><th>Input Operation (Read)</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>Port A - Data Bus</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Port B - Data Bus</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>Port C - Data Bus</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Control Word - Data Bus</td></tr>
<tr><td colspan="5"></td><td>Output Operation (Write)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Port A</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Port B</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Port C</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Control</td></tr>
<tr><td colspan="5"></td><td>Disable Function</td></tr>
<tr><td>X</td><td>X</td><td>X</td><td>X</td><td>1</td><td>Data Bus - 3 - State</td></tr>
<tr><td>X</td><td>X</td><td>1</td><td>1</td><td>0</td><td>Data Bus - 3 - State</td></tr>
</table></td></tr>
<tr><td>PC_{7-4}</td><td>10–13</td><td>11,13–15</td><td>I/O</td><td>PORT C, PINS 4–7: Upper nibble of an 8-bit data output latch/buffer and an 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.</td></tr>
<tr><td>PC_{0-3}</td><td>14–17</td><td>16–19</td><td>I/O</td><td>PORT C, PINS 0–3: Lower nibble of Port C.</td></tr>
<tr><td>PB_{0-7}</td><td>18–25</td><td>20–22, 24–28</td><td>I/O</td><td>PORT B, PINS 0–7: An 8-bit data output latch/buffer and an 8-bit data input buffer.</td></tr>
<tr><td>V_{CC}</td><td>26</td><td>29</td><td></td><td>SYSTEM POWER: + 5V Power Supply.</td></tr>
<tr><td>D_{7-0}</td><td>27–34</td><td>30–33, 35–38</td><td>I/O</td><td>DATA BUS: Bi-directional, tri-state data bus lines, connected to system data bus.</td></tr>
<tr><td>RESET</td><td>35</td><td>39</td><td>I</td><td>RESET: A high on this input clears the control register and all ports are set to the input mode.</td></tr>
<tr><td>$\overline{WR}$</td><td>36</td><td>40</td><td>I</td><td>WRITE CONTROL: This input is low during CPU write operations.</td></tr>
<tr><td>PA_{7-4}</td><td>37–40</td><td>41–44</td><td>I/O</td><td>PORT A, PINS 4–7: Upper nibble of an 8-bit data output latch/buffer and an 8-bit data input latch.</td></tr>
<tr><td>NC</td><td></td><td>1, 12, 23, 34</td><td></td><td>No Connect</td></tr>
</table>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

82C55A FUNCTIONAL DESCRIPTION

General

The 82C55A is a programmable peripheral interface 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 82C55A 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 82C55A 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.

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 82C55A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 82C55A.

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 be both written and read as shown in the address decode table in the pin descriptions. Figure 6 shows the control word format for both Read and Write operations. When the control word is read, bit D7 will always be a logic "1", as this implies control word mode information.

Ports A, B, and C

The 82C55A 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 82C55A.

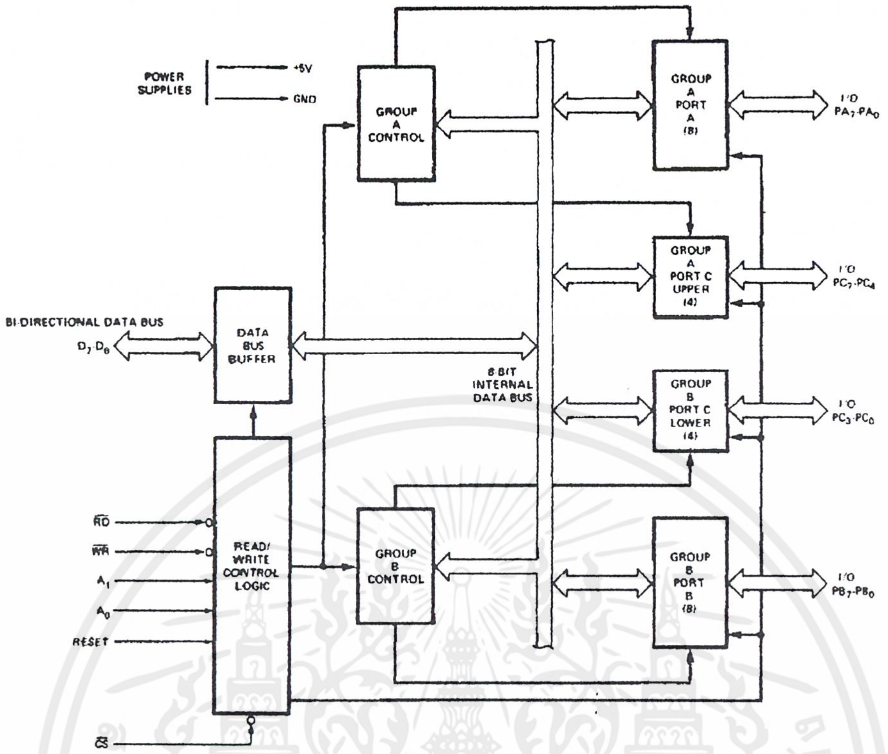
Port A. One 8-bit data output latch/buffer and one 8-bit input latch buffer. Both "pull-up" and "pull-down" bus hold devices are present on Port A.

Port B. One 8-bit data input/output latch/buffer. Only "pull-up" bus hold devices are present on Port B.

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. Only "pull-up" bus hold devices are present on Port C.

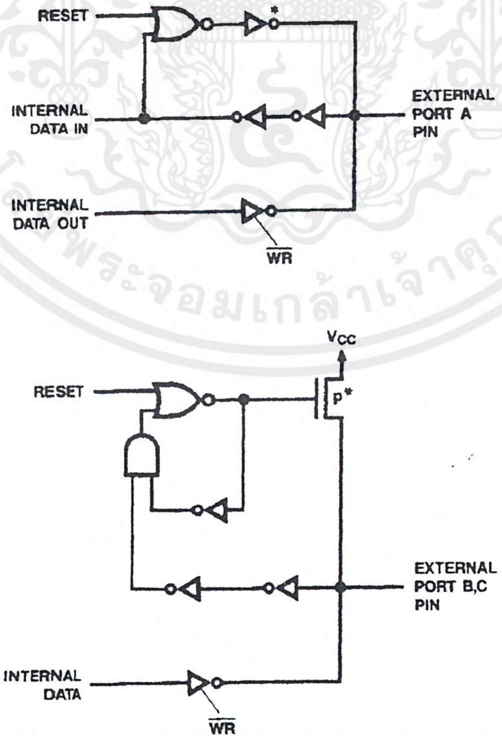
See Figure 4 for the bus-hold circuit configuration for Port A, B, and C.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



231256-3

Figure 3. 82C55A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions



231256-4

*NOTE:
Port pins loaded with more than 20 pF capacitance may not have their logic level guaranteed following a hardware reset.

Figure 4. Port A, B, C, Bus-hold Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

82C55A OPERATIONAL DESCRIPTION

Mode Selection

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

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 with all 24 port lines held at a logic "one" level by the internal bus hold devices (see Figure 4 Note). After the reset is removed the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need for pullup or pulldown devices in "all CMOS" designs. During the execution of the system program, any of the other modes may be selected by using a single output instruction. This allows a single 82C55A 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.

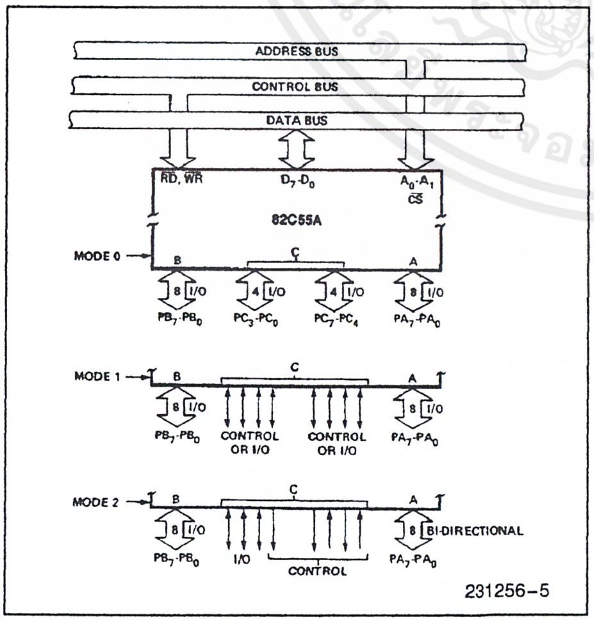


Figure 5. Basic Mode Definitions and Bus Interface

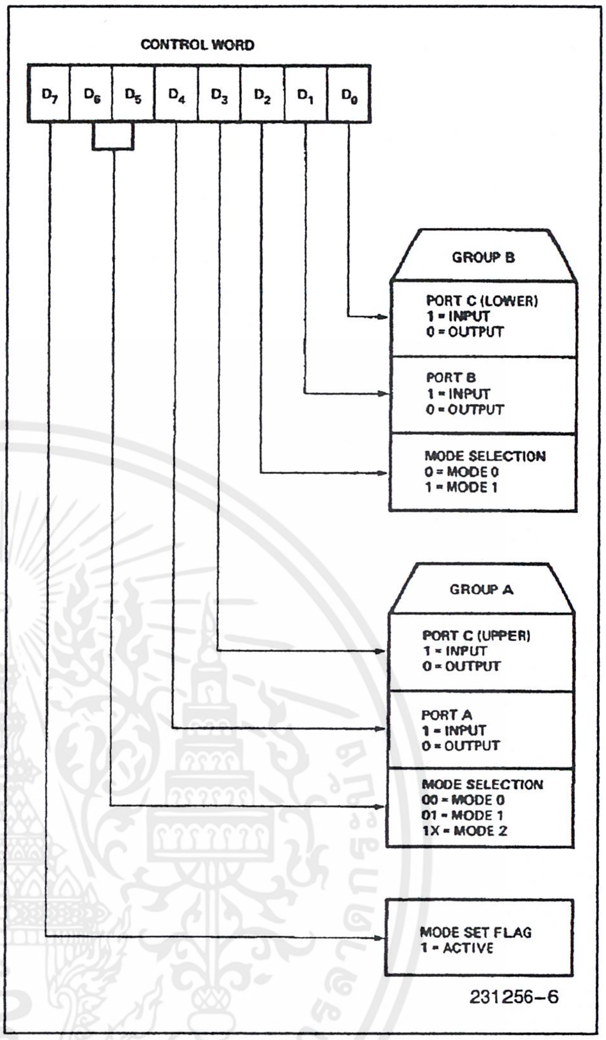


Figure 6. Mode Definition Format

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 82C55A 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.

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

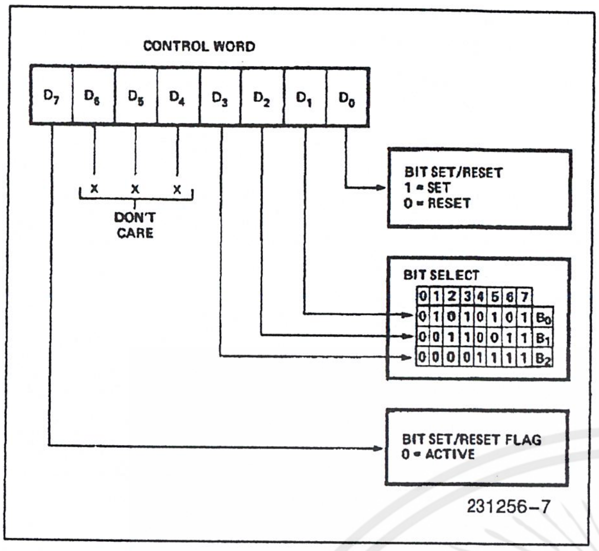


Figure 7. Bit Set/Reset Format

Interrupt Control Functions

When the 82C55A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the Programmer to disallow or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip-flop definition:

(BIT-SET)—INTE is SET—Interrupt enable
(BIT-RESET)—INTE is RESET—Interrupt disable

Note:
All Mask flip-flops are automatically reset during mode selection and device Reset.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias....0°C to + 70°C
Storage Temperature ....− 65°C to + 150°C
Supply Voltage ....− 0.5 to + 8.0V
Operating Voltage ....+ 4V to + 7V
Voltage on any Input..........GND − 2V to + 6.5V
Voltage on any Output ..GND − 0.5V to V_{CC} + 0.5V
Power Dissipation ....1 Watt

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

D.C. CHARACTERISTICS

T_A = 0°C to 70°C, V_{CC} = +5V ±10%, GND = 0V (T_A = −40°C to +85°C for Extended Temperture)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	−0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.5 mA
V_{OH}	Output High Voltage	3.0 V_{CC} − 0.4		V V	I_{OH} = −2.5 mA I_{OH} = −100 μA
I_{IL}	Input Leakage Current		±1	μA	V_{IN} = V_{CC} to 0V (Note 1)
I_{OFL}	Output Float Leakage Current		±10	μA	V_{IN} = V_{CC} to 0V (Note 2)
I_{DAR}	Darlington Drive Current	±2.5	(Note 4)	mA	Ports A, B, C R_{ext} = 500Ω V_{ext} = 1.7V
I_{PHL}	Port Hold Low Leakage Current	+50	+300	μA	V_{OUT} = 1.0V Port A only
I_{PHH}	Port Hold High Leakage Current	−50	−300	μA	V_{OUT} = 3.0V Ports A, B, C
I_{PHLO}	Port Hold Low Overdrive Current	−350		μA	V_{OUT} = 0.8V
I_{PHHO}	Port Hold High Overdrive Current	+350		μA	V_{OUT} = 3.0V
I_{CC}	V_{CC} Supply Current		10	mA	(Note 3)
I_{CCSB}	V_{CC} Supply Current-Standby		10	μA	V_{CC} = 5.5V V_{IN} = V_{CC} or GND Port Conditions If I/P = Open/High O/P = Open Only With Data Bus = High/Low $\overline{CS}$ = High Reset = Low Pure Inputs = Low/High

NOTES:
1. Pins A_1, A_0, $\overline{CS}$, $\overline{WR}$, $\overline{RD}$, Reset.
2. Data Bus; Ports B, C.
3. Outputs open.
4. Limit output current to 4.0 mA.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

19-4323; Rev 6; 3/96

MAXIM

+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

MAX220–MAX249

General Description

The MAX220-MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 communications interfaces, and in particular, for those applications where ±12V is not available.

These parts are particularly useful in battery-powered systems, since their low-power shutdown mode reduces power dissipation to less than 5μW. The MAX225, MAX233, MAX235, and MAX245-MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

Applications

Portable Computers

Low-Power Modems

Interface Translation

Battery-Powered RS-232 Systems

Multi-Drop RS-232 Networks

Features

Superior to Bipolar

- ♦ **Operate from Single +5V Power Supply (+5V and +12V—MAX231/MAX239)**
- ♦ **Low-Power Receive Mode in Shutdown (MAX223/MAX242)**
- ♦ **Meet All EIA/TIA-232E and V.28 Specifications**
- ♦ **Multiple Drivers and Receivers**
- ♦ **3-State Driver and Receiver Outputs**
- ♦ **Open-Line Detection (MAX243)**

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX220CPE	0°C to +70°C	16 Plastic DIP
MAX220CSE	0°C to +70°C	16 Narrow SO
MAX220CWE	0°C to +70°C	16 Wide SO
MAX220C/D	0°C to +70°C	Dice*
MAX220EPE	-40°C to +85°C	16 Plastic DIP
MAX220ESE	-40°C to +85°C	16 Narrow SO
MAX220EWE	-40°C to +85°C	16 Wide SO
MAX220EJE	-40°C to +85°C	16 CERDIP
MAX220MJE	-55°C to +125°C	16 CERDIP

Ordering Information continued at end of data sheet.

**Contact factory for dice specifications.*

Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value (μF)	SHDN & Three-State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	4.7/10	No		120	Ultra-low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes		200	Low-power shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✔	120	MAX241 + receivers active in shutdown
MAX225	+5	5/5	0	–	Yes	✔	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes		120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No		120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No		120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No		200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	–	No		120	No external caps
MAX233A	+5	2/2	0	–	No		200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No		120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	–	Yes		120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes		120	Shutdown, three state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No		120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No		120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No		120	Standard +5/+12V or battery supplies; single-package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes		120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes		120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✔	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No		200	Open-line detection simplifies cabling
MAX244	+5	8/10	4	1.0	No		120	High slew rate
MAX245	+5	8/10	0	–	Yes	✔	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	8/10	0	–	Yes	✔	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	8/9	0	–	Yes	✔	120	High slew rate, int. caps, nine operating modes
MAX248	+5	8/8	4	1.0	Yes	✔	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✔	120	Available in quad flatpack package

For free samples & the latest literature: http://www.maxim-ic.com, or phone 1-800-998-8800

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

ABSOLUTE MAXIMUM RATINGS—MAX220/222/232A/233A/242/243

Supply Voltage (V_{CC}) ....-0.3V to +6V
Input Voltages
T_{IN} ....-0.3V to (V_{CC} - 0.3V)
R_{IN} ....±30V
T_{OUT} (Note 1) ....±15V
Output Voltages
T_{OUT} ....±15V
R_{OUT} ....-0.3V to (V_{CC} + 0.3V)
Driver/Receiver Output Short Circuited to GND.........Continuous
Continuous Power Dissipation (T_A = +70°C)
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C)....842mW
18-Pin Plastic DIP (derate 11.11mW/°C above +70°C)....889mW
20-Pin Plastic DIP (derate 8.00mW/°C above +70°C) 440mW
16-Pin Narrow SO (derate 8.70mW/°C above +70°C)...696mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C)......762mW
18-Pin Wide SO (derate 9.52mW/°C above +70°C)......762mW
20-Pin Wide SO (derate 10.00mW/°C above +70°C)....800mW
20-Pin SSOP (derate 8.00mW/°C above +70°C) 640mW
16-Pin CERDIP (derate 10.00mW/°C above +70°C).....800mW
18-Pin CERDIP (derate 10.53mW/°C above +70°C).....842mW
Operating Temperature Ranges
MAX2_ _AC_ _, MAX2_ _C_ _ ....0°C to +70°C
MAX2_ _AE_ _, MAX2_ _E_ _ ....-40°C to +85°C
MAX2_ _AM_ _, MAX2_ _M_ _ ....-55°C to +125°C
Storage Temperature Range ....-65°C to +160°C
Lead Temperature (soldering, 10sec) ....+300°C

Note 1: Input voltage measured with T_{OUT} in high-impedance state, $\overline{SHDN}$ or V_{CC} = 0V.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243

(V_{CC} = +5V ±10%, C1-C4 = 0.1µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 TRANSMITTERS						
Output Voltage Swing	All transmitter outputs loaded with 3kΩ to GND		±5	±8		V
Input Logic Threshold Low				1.4	0.8	V
Input Logic Threshold High			2	1.4		V
Logic Pull-Up/Input Current	Normal operation			5	40	µA
	$\overline{SHDN}$ = 0V, MAX222/242, shutdown			±0.01	±1	
Output Leakage Current	V_{CC} = 5.5V, $\overline{SHDN}$ = 0V, V_{OUT} = ±15V, MAX222/242			±0.01	±10	µA
	V_{CC} = $\overline{SHDN}$ = 0V, V_{OUT} = ±15V			±0.01	±10	
Data Rate	Except MAX220, normal operation			200	116	kbits/sec
	MAX220			22	20	
Transmitter Output Resistance	V_{CC} = V+ = V- = 0V, V_{OUT} = ±2V		300	10M		Ω
Output Short-Circuit Current	V_{OUT} = 0V		±7	±22		mA
RS-232 RECEIVERS						
RS-232 Input Voltage Operating Range					±30	V
RS-232 Input Threshold Low	V_{CC} = 5V	Except MAX243 $R2_{IN}$	0.8	1.3		V
		MAX243 $R2_{IN}$ (Note 2)	-3			
RS-232 Input Threshold High	V_{CC} = 5V	Except MAX243 $R2_{IN}$		1.8	2.4	V
		MAX243 $R2_{IN}$ (Note 2)		-0.5	-0.1	
RS-232 Input Hysteresis	Except MAX243, V_{CC} = 5V, no hyst. in shdn.		0.2	0.5	1	V
	MAX243			1		
RS-232 Input Resistance			3	5	7	kΩ
TTL/CMOS Output Voltage Low	I_{OUT} = 3.2mA			0.2	0.4	V
TTL/CMOS Output Voltage High	I_{OUT} = -1.0mA		3.5	V_{CC} - 0.2		V
TTL/CMOS Output Short-Circuit Current	Sourcing V_{OUT} = GND		-2	-10		mA
	Shrinking V_{OUT} = V_{CC}		10	30		
TTL/CMOS Output Leakage Current	$\overline{SHDN}$ = V_{CC} or $\overline{EN}$ = V_{CC} ($\overline{SHDN}$ = 0V for MAX222), 0V ≤ V_{OUT} ≤ V_{CC}			±0.05	±10	µA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

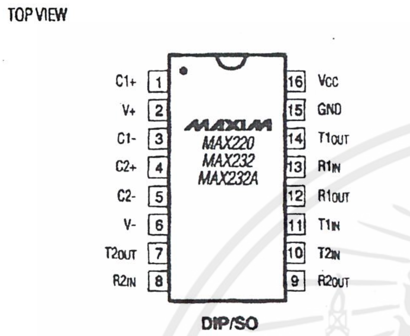
ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)

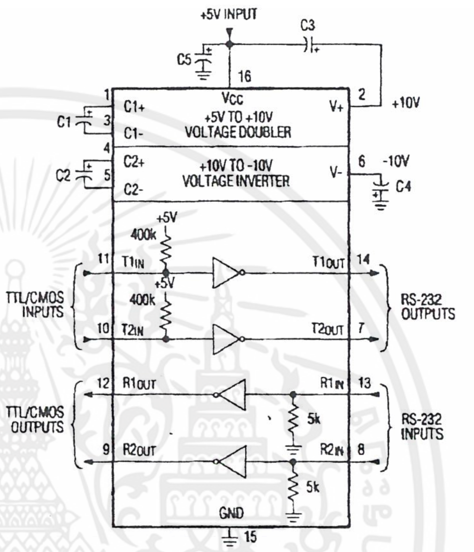
(V_{CC} = +5V ±10%, C1-C4 = 0.1µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
$\overline{EN}$ Input Threshold Low	MAX242			1.4	0.8	V
$\overline{EN}$ Input Threshold High	MAX242		2.0	1.4		V
POWER SUPPLY						
Operating Supply Voltage			4.5		5.5	V
V_{CC} Supply Current ($\overline{SHDN}$ = V_{CC}), Figures 5, 6, 9, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3kΩ load both inputs	MAX220		12		
		MAX222/232A/233A/242/243		15		
Shutdown Supply Current	MAX222/242	T_A = +25°C		0.1	10	µA
		T_A = 0° to +70°C		2	50	
		T_A = -40° to +85°C		2	50	
		T_A = -55° to +125°C		35	100	
$\overline{SHDN}$ Input Leakage Current	MAX222/242				±1	µA
$\overline{SHDN}$ Threshold Low	MAX222/242			1.4	0.8	V
$\overline{SHDN}$ Threshold High	MAX222/242		2.0	1.4		V
AC CHARACTERISTICS						
Transition Slew Rate	C_L = 50pF to 2500pF, R_L = 3kΩ to 7kΩ, V_{CC} = 5V, T_A = +25°C, measured from +3V to -3V or -3V to +3V	MAX222/232A/233A/242/243	6	12	30	V/µs
		MAX220	1.5	3	30	
Transmitter Propagation Delay TLL to RS-232 (normal operation), Figure 1	t_{PHLT}	MAX222/232A/233A/242/243		1.3	3.5	µs
		MAX220		4	10	
	t_{PLHT}	MAX222/232A/233A/242/243		1.5	3.5	
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (normal operation), Figure 2	t_{PHLR}	MAX222/232A/233A/242/243		0.5	1	µs
		MAX220		0.6	3	
	t_{PLHR}	MAX222/232A/233A/242/243		0.6	1	
		MAX220		0.8	3	
Receiver Propagation Delay RS-232 to TLL (shutdown), Figure 2	t_{PHLS}	MAX242		0.5	10	µs
	t_{PLHS}	MAX242		2.5	10	
Receiver-Output Enable Time, Figure 3	t_{ER}	MAX242		125	500	ns
Receiver-Output Disable Time, Figure 3	t_{DR}	MAX242		160	500	ns
Transmitter-Output Enable Time ($\overline{SHDN}$ goes high), Figure 4	t_{ET}	MAX222/242, 0.1µF caps (includes charge-pump start-up)		250		µs
Transmitter-Output Disable Time ($\overline{SHDN}$ goes low), Figure 4	t_{DT}	MAX222/242, 0.1µF caps		600		ns
Transmitter + to - Propagation Delay Difference (normal operation)	t_{PHLT} - t_{PLHT}	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to - Propagation Delay Difference (normal operation)	t_{PHLR} - t_{PLHR}	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

Note 2: MAX243 $R2_{OUT}$ is guaranteed to be low when $R2_{IN}$ is ≥ 0V or is floating.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





CAPACITANCE (µF)					
DEVICE	C1	C2	C3	C4	C5
MAX220	4.7	4.7	10	10	4.7
MAX232	1.0	1.0	1.0	1.0	1.0
MAX232A	0.1	0.1	0.1	0.1	0.1

Figure 5. MAX220/232/232A Pin Configuration and Typical Operating Circuit

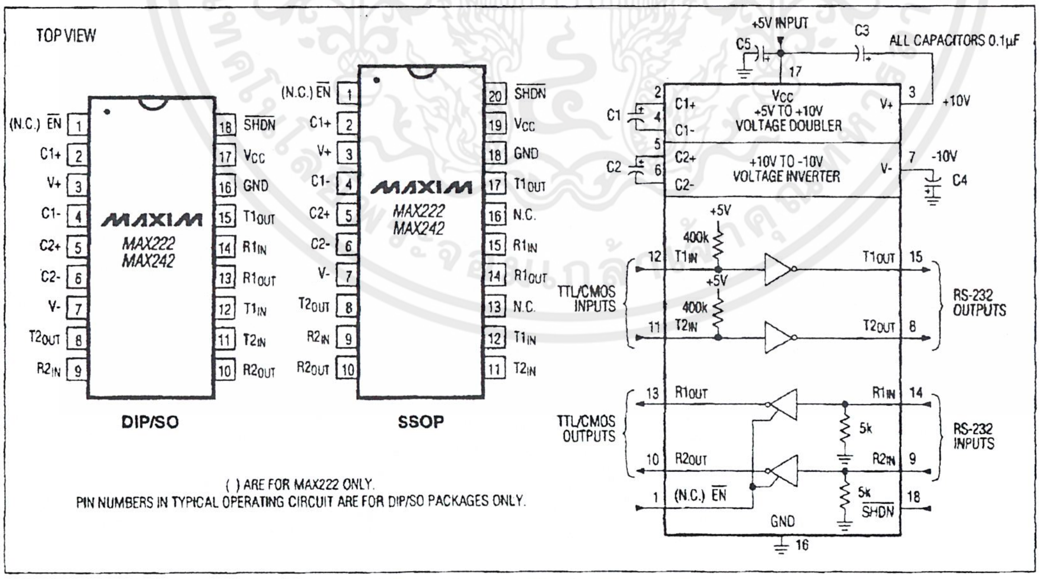


Figure 6. MAX222/MAX242 Pin Configuration and Typical Operating Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้