



การออกแบบระบบควบคุมประตูอัตโนมัติ
A DESIGN OF GATE AUTOMATION SYSTEM



โดย
นาย เจนวิทย์ ศรีชุมจันทร์
นาย สมเกียรติ เดชสุชน

วัน เดือน ปี..... 22.ค.ค.2541
เลขทะเบียน..... 039097
เลขเรียกหนังสือ..... 10337 จ 698 ก.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ปร
039097
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบระบบควบคุมประตูอัตโนมัติ
A DESIGN OF GATE AUTOMATION SYSTEM

โดย

นาย เจนวิทย์ ศรีชุมจันทร์ 38013053

นาย สมเกียรติ เดชสุชน 38013077

อาจารย์ที่ปรึกษา

รศ.ดร. ถวิล พึ่งมา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2540

ภาควิชาวิศวกรรมโทรคมนาคม

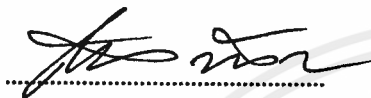
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การออกแบบระบบควบคุมประตูอัตโนมัติ

A DESIGN OF GATE AUTOMATION SYSYTEM

ผู้จัดทำ

1. นายเจนวิทย์ ศรีชุมจันทร์ 38013053
2. นายสมเกียรติ เชนสุชน 38013077



อาจารย์ที่ปรึกษา

(รศ.ดร. ถวิติ พึ่งมา)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมประตูอัตโนมัติ
GATE AUTOMATION SYSTEM
โดย นายเจนวิทย์ ศรีจุมจันทร์ 38013053
นายสมเกียรติ เศษุขน 38013077
อาจารย์ที่ปรึกษา รศ.ดร.ถวิล พึ่งมา

บทคัดย่อ

โครงการนี้เป็นระบบควบคุมการเปิดปิดประตูอัตโนมัติ (Gate Automation System) โดยมีซีพียู (CPU) เป็นตัวควบคุมการทำงานของระบบ ซึ่งมีการรับอินพุต(Input)จากรีโมทควบคุม(Remote Control) มีการรับการโปรแกรมจากคีย์บอร์ด (Keyboard) และมีจอแสดงผล(Display Monitor) เป็นตัวแสดงผล มีวงจรควบคุมการหมุนของมอเตอร์ มีกลอนไฟฟ้า(Solenoid Bolt) เป็นตัวล็อคประตู มีระบบไฟสำรอง (Power Backup) ป้องกันการเสียหายของข้อมูล และใช้อินฟราเรด(Infrared) ในการตรวจจับการทำงานในระบบในกรณีที่เกิดการทำงานผิดพลาดของระบบ

Abstract

This project is a system for controlling the automatic gate by using CPU to control operating system which receive the input signal from remote control from keyboard and display as the output on LCD. There are circuit for controlling to turn the motor, the function of solenoid bolt is used to lock the gate. The power backup is used to protect the damage of the data by using the infrared to sense the operating system in case of the fault system.

สารบัญ

หัวข้อ	หน้า
1.1 แนวความคิดในการทำปริญญาโท	1
1.2 เนื้อหาโดยสังเขปของปริญญาโท	1
2.1 หลักการโดยทั่วไปของระบบควบคุมประจําอัตโนมัติ	2
2.2 ความต้องการของระบบควบคุมประจําอัตโนมัติ	2
2.3 คุณสมบัติของอุปกรณ์ที่ใช้ในระบบควบคุมประจําอัตโนมัติ	5
2.3.1 อุปกรณ์ไมโครคอนโทรลเลอร์	5
2.3.2 การอินเตอร์เฟสกับจอแสดงผล LCD โมดูล	13
2.3.3 การอินเตอร์เฟสกับคีย์บอร์ด	18
2.3.4 คุณลักษณะของของอุปกรณ์อาร์ทีซี	22
2.3.5 คุณลักษณะของสวิทช์แม่เหล็ก	23
2.3.6 มอเตอร์เหนี่ยวนำเฟสเดียว	24
3.1 หลักการออกแบบวงจร	31
3.2 การอินเตอร์เฟส	31
3.2.1 การอินเตอร์เฟสกับแอลซีดี โมดูล	31
3.2.2 การอินเตอร์เฟสกับคีย์บอร์ด	33
3.2.3 การอินเตอร์เฟสกับอุปกรณ์อาร์ทีซี	37
3.2.4 วงจรขับมอเตอร์	37
3.2.5 วงจรขับกลอนและกระดิ่งไฟฟ้า	38
3.3 วงจรและการทำงานของระบบควบคุมประจําอัตโนมัติ	38
3.3.1 โฟลว์ชาร์ตการทำงานของระบบ	41
3.3.2 ฟังก์ชันการทำงานของระบบควบคุมประจําอัตโนมัติ	45
3.3.3 การใช้งานระบบควบคุมประจําอัตโนมัติ	45
3.4 การออกแบบแผ่นลายวงจร	45
4.1 วิธีการทดลอง	48
4.2 ผลการทดลอง	48
5.1 ปัญหาที่เกิดขึ้นกับการทดลอง	53
5.2 แนวทางการแก้ปัญหา	53
ภาคผนวก	

บทที่ 1

บทนำ

1.1 แนวความคิดในการทำปริญญาโท

การดำรงชีวิตของมนุษย์ในปัจจุบันล้วนยึดติดกับสิ่งอำนวยความสะดวกในรูปแบบของเครื่องใช้ไฟฟ้า อันทันสมัยมากมาย แต่ละแบบก็มีความสะดวกต่อการใช้งานแตกต่างกัน ระบบควบคุมการปิดเปิดประตูอัตโนมัติ ก็เป็นอีกรูปแบบหนึ่งซึ่งทำให้ผู้ใช้สามารถ ควบคุมการปิดหรือเปิดประตูได้อย่างสะดวก โดยการใช้รีโมตควบคุมระยะไกล ซึ่งลักษณะของประตูในที่นี้ก็จะประตูที่มีลักษณะ การปิดหรือเปิดในแนวขวางของทิศทางการเข้าออกซึ่งจะใช้มอเตอร์เป็นตัวขับเคลื่อน ให้ประตูปิดหรือเปิดยกตัวอย่าง เช่น ประตูรั้วบ้าน, ประตูห้างสรรพสินค้าที่ทันสมัยต่างๆ เป็นต้น

สำหรับปริญญาโทนี้จะได้นำเสนอถึงระบบควบคุมการปิดเปิดประตู ซึ่งประกอบด้วยหน่วยย่อยต่างๆคือ หน่วยของการประมวลผลส่วนกลาง(CPU) ,หน่วยของอินพุต(input) และหน่วยของเอาต์พุต(output) ซึ่งเป็นหน่วยการทำงานหลักของระบบ โดยในเนื้อหาจะแจกแจงหน่วยการทำงานดังกล่าวโดยละเอียด ทฤษฎี และหลักการของระบบ,การทำงานของระบบ,การออกแบบและการสร้างระบบ,การทดลอง,ทดลองจนแนวทางในการพัฒนาและประยุกต์ใช้งานระบบ

1.2 เนื้อหาโดยสังเขปของปริญญาโท

ปริญญาโทฉบับนี้ เป็นการออกแบบระบบควบคุมประตูอัตโนมัติ ซึ่งสามารถนำไปประยุกต์ใช้กับประตูในสถานที่ต่างๆ ตามความเหมาะสม และเหมาะสมกับสภาพสังคมในปัจจุบันที่ต้องการความสะดวกสบายให้สมกับยุคสมัยที่เราเรียกกันว่า “โลกาภิวัตน์” (Globalization) ทางคณะผู้จัดทำได้จัดแบ่งเนื้อหาของระบบควบคุมประตูอัตโนมัติไว้ในบทต่างๆดังต่อไปนี้

บทที่ 2 ทฤษฎีและหลักการเป็นการกล่าวถึงทฤษฎีต่างๆ ที่เกี่ยวกับระบบโดยรวมทั้งหมด ตั้งแต่แนวความคิดเริ่มต้นในการสร้างระบบไปจนถึงหน้าที่คุณสมบัติของส่วนต่างๆ ที่เกี่ยวข้องกับระบบ

บทที่ 3 การออกแบบและการสร้างเป็นการกล่าวถึง วิธีการออกแบบตามที่ได้มีแนวความคิดไว้แล้วในทางทฤษฎี และการเลือกใช้อุปกรณ์ต่างๆ เพื่อนำมาใช้กับระบบ เพื่อให้ได้ประสิทธิภาพสูงสุด เท่าที่เป็นไปได้ ในขณะที่เดียวกัน ก็คำนึงถึงต้นทุนในการสร้างที่ต่ำที่สุดด้วย

บทที่ 4 การทดลองและผลการทดลอง ซึ่งจะกล่าวถึงการทดลองโดยสรุปของปริญญาโท โดยการเปรียบเทียบกับ ความต้องการของระบบที่คาดหวังเอาไว้ในทางทฤษฎี

บทที่ 5 บทสรุปปัญหา แนวทางแก้ไข และการพัฒนาระบบเป็นการกล่าวถึง ปัญหาต่างๆที่เกิดขึ้นกับระบบที่ได้ทดลองและปฏิบัติ พร้อมทั้งแนวทางในการแก้ไขและพัฒนาให้ระบบมีประสิทธิภาพ และประโยชน์ที่ได้มีคุณค่าสูงสุด

บทที่ 2

ทฤษฎีและหลักการ

2.1 หลักการ โดยทั่วไปของระบบควบคุมประจําอัตโนมัติ

ในระบบควบคุมประจําอัตโนมัตินั้นมีส่วนประกอบที่สำคัญอยู่ 3 ส่วนคือส่วนของหน่วยประมวลผล ส่วนกลาง(Central Processing Unit),หน่วยของการรับอินพุต(Input unit) และหน่วยของเอาต์พุต (Output Unit) ซึ่งทั้ง 3 ส่วนนี้จะต้องมีการทำงานที่สัมพันธ์กัน

2.1.1 หน่วยประมวลผลส่วนกลาง

จะทำหน้าที่ควบคุมระบบการทำงานทั้งหมด เช่น การรับอินพุตจากเครื่องควบคุมระยะไกล (Remote Control) มาเป็นตัวสั่งการให้ประจําปิดหรือเปิด การรับข้อมูลจากคีย์บอร์ดเพื่อที่จะตั้งฟังก์ชันการทำงาน(Function Set)หรือในกรณีที่ระบบไฟฟ้าขัดข้องหน่วยประมวลผลนี้จะนำค่าที่ได้จากการตั้งฟังก์ชันการทำงานเก็บไว้ในส่วนของเมมโมรี่แบคอัพ(Memmory Backup) หลังจากทีระบบไฟฟ้าเป็นปกติค่าเหล่านี้ก็จะถูกโหลด(load) ออกมาใช้งานตามเดิม จะเห็นได้ว่าหน้าที่การทำงานของหน่วยประมวลผลส่วนกลางนี้จะต้องอาศัยอุปกรณ์ที่สามารถที่จะทำการ โปรแกรม(Program) ไว้ได้และอุปกรณ์ที่ว่านี้ก็จะ เป็น อุปกรณ์ไมโครคอนโทรลเลอร์(Microcontroller) ซึ่งจะได้กล่าวถึงอุปกรณ์ชนิดนี้ในส่วนถัดไป

2.1.2 หน่วยอินพุต

เป็นหน่วยรับข้อมูลและสั่งการให้ระบบควบคุมประจําอัตโนมัติทำงาน เช่น เครื่องควบคุมระยะไกลซึ่งทำหน้าที่สั่งการให้ประจําปิดหรือเปิด อุปกรณ์ตรวจจับอินฟราเรด(Infrared Sensor)ซึ่งคอยตรวจจับสิ่งกีดขวางทางเข้าออกประจําและหากมีสิ่งกีดขวางอยู่ระบบควบคุมประจําอัตโนมัติจะหยุดการขับเคลื่อนประจํา จนกว่าสิ่งกีดขวางนั้นจะออกไปจากทิศทางของการตรวจจับ ทั้งนี้เพื่อความปลอดภัย ยกตัวอย่างเช่น กรณีที่มีรถจอดขวางประจําอยู่ระบบขับเคลื่อนประจําจะต้องหยุดการทำงานมิฉะนั้นอาจทำให้เกิดความเสียหายกับทรัพย์สินได้ คีย์บอร์ดก็เป็นอุปกรณ์รับอินพุตอีกชนิดหนึ่งซึ่งจะรับข้อมูลการตั้งค่าการทำงานให้ระบบทำงานในลักษณะต่างๆตามความต้องการ

2.1.3 หน่วยเอาต์พุต

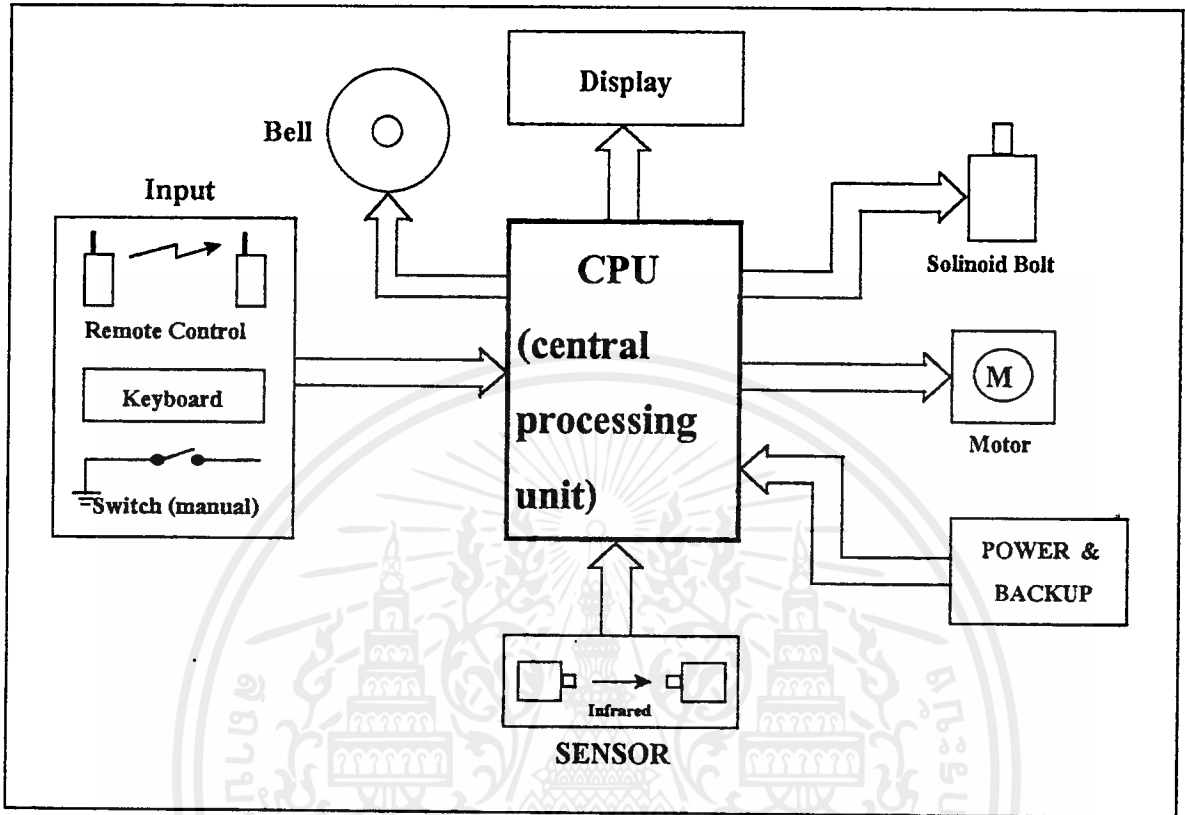
เป็นหน่วยซึ่งจะถูกควบคุมโดยหน่วยประมวลผลส่วนกลางซึ่งจะมีการทำงานที่สัมพันธ์กับอินพุต หน่วยของเอาต์พุตก็เช่นหน่วยขับเคลื่อนมอเตอร์ซึ่งจะรับคำสั่งจากหน่วยประมวลผลให้ทำการขับเคลื่อนประจํา ปิดหรือเปิด อุปกรณ์กลอนไฟฟ้า อุปกรณ์แสดงผลการตั้งค่าต่างๆของระบบซึ่งก็จะเป็นแอลซีดีโมดูล(LCD Module) ลำโพงและอุปกรณ์ที่แสดงสถานะการทำงานของประจํา เช่น หลอดแอลอีดี(LED) เป็นต้น

ลักษณะบล็อคลำโพงและมอเตอร์ของระบบควบคุมประจําอัตโนมัติแสดงดังรูปที่ 2.1 ซึ่งจากบล็อคลำโพงและมอเตอร์ ได้แสดงถึงส่วนประกอบต่างๆดังที่ได้กล่าวไว้แล้วในเบื้องต้น ต่อไปนี้จะได้กล่าวถึงลักษณะความต้องการของระบบ และอุปกรณ์ในส่วนย่อยต่างๆในทางทฤษฎี

2.2 ความต้องการของระบบควบคุมประจําอัตโนมัติ

ก่อนที่จะกำหนดเป้าหมายความต้องการของระบบจำเป็นต้องรู้ถึงโครงสร้างที่ชัดเจนของ ลักษณะประจําและการติดตั้งอุปกรณ์ของระบบกับประจํา ซึ่งประจําจะมีลักษณะการปิดเปิดในแนวขวางของทิศทาง การเข้าออก และในการนำระบบไปติดตั้งใช้งานจริงนั้น จะมีมอเตอร์เป็นตัวขับเคลื่อนซึ่งติดอยู่กับประจํา มีเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวรับสัญญาณควบคุมระยะไกล (Remote Detector) เพื่อคอยรับสัญญาณจากเครื่องควบคุมระยะไกลและส่งสัญญาณไปยังส่วนควบคุมระบบให้ทำการเปิดหรือปิดประตู, มีสวิตช์กดปุ่ม (Manual Switch) ไว้สำหรับ



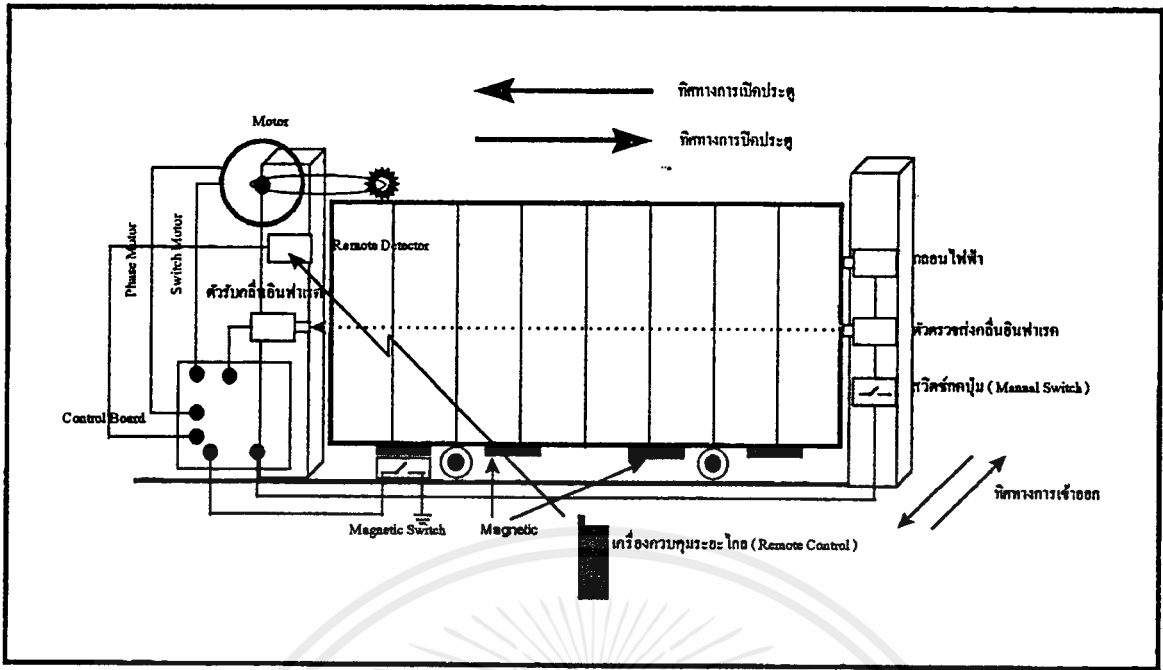
รูปที่ 2.1 แสดงลักษณะบล็อกไดอะแกรมของระบบควบคุมประตูอัตโนมัติ

ปิดเปิดประตู ในกรณีที่ไม่ได้ใช้เครื่องควบคุมระยะไกล, มีเครื่องตรวจจับอินฟราเรดคอยตรวจจับสิ่งกีดขวางทางเข้าออกของประตู, มีกลอนไฟฟ้าและมีบอร์ดควบคุม (Control Board) ซึ่งอุปกรณ์ทั้งหลายเหล่านี้จะถูกติดตั้งอยู่ตามจุดต่างๆ ของประตูดังแสดงในรูปที่ 2.2

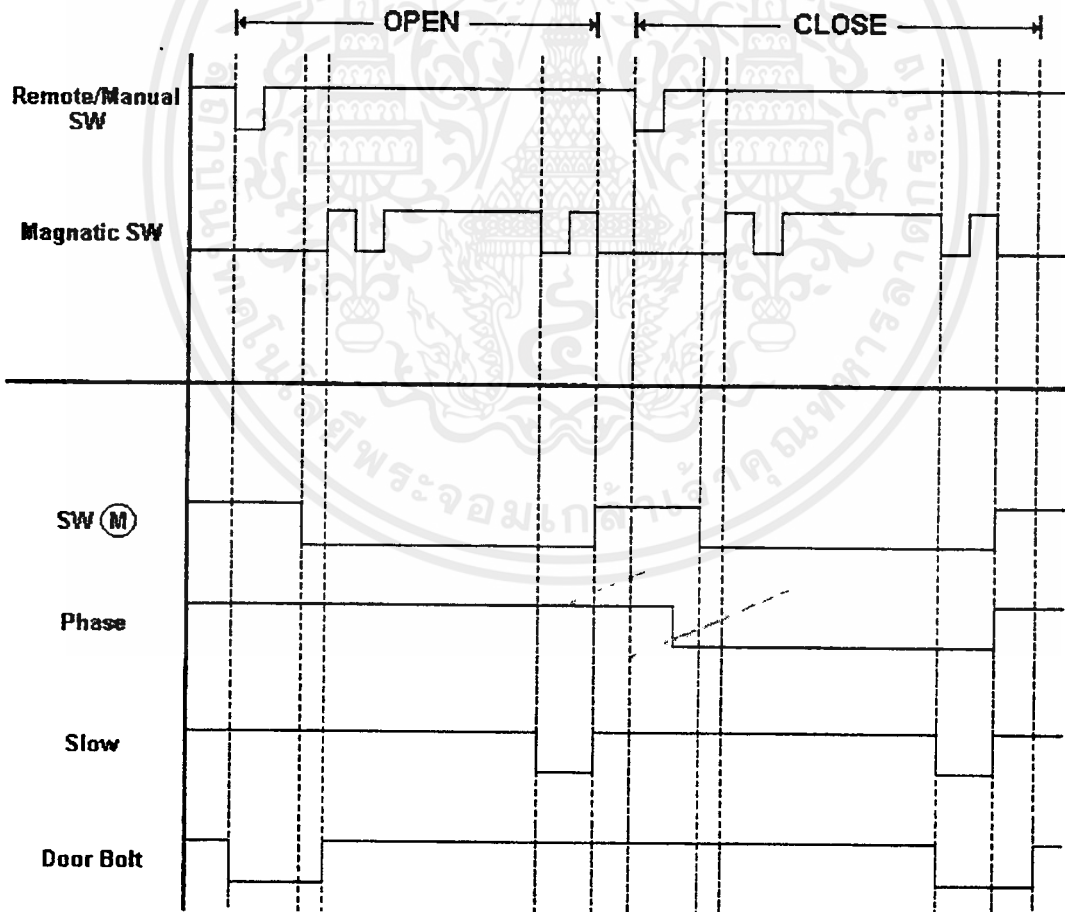
หลังจากที่ได้กำหนดลักษณะการติดตั้งของอุปกรณ์ต่างๆ ลงไปอย่างชัดเจนทำให้เราสามารถทราบว่าจะมีการทำงานในลักษณะอย่างไร ซึ่งเมื่อพิจารณาถึงอุปกรณ์ที่เกี่ยวข้องกับระบบเราพอที่จะสร้างความสัมพันธ์ของอุปกรณ์เหล่านี้ได้อย่างคร่าวๆ ตามไทมิ่งไดอะแกรม (Timing Diagram) ซึ่งไทมิ่งไดอะแกรมจะเป็นตัวบ่งชี้ให้ทราบว่าส่วนของอุปกรณ์ควบคุมของระบบนั้นจะต้องควบคุมระบบการทำงานเป็นไปในลักษณะใด รูปที่ 2.3 แสดงไทมิ่งไดอะแกรมของระบบควบคุมประตูอัตโนมัติ

จากไทมิ่งไดอะแกรมของรูปที่ 2.3 เมื่อมีสัญญาณจากรีโมตคอนโทรลหรือสวิตช์กดปุ่ม ซึ่งจะเป็นสัญญาณพัลส์เข้ามาจะมีการเปลี่ยนแปลงของระดับสัญญาณซึ่งใช้ในการควบคุมอุปกรณ์ต่างๆ ตั้งแต่สวิตช์แม่เหล็ก, สวิตช์ที่ป้อนจ่ายกระแสไฟฟ้าให้กับมอเตอร์, กลอนประตูไฟฟ้า และพัลส์ซึ่งใช้ในการควบคุมให้มอเตอร์หมุนช้าลงในช่วงที่ประตูกำลังจะปิดสุดและเปิดสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดงการติดตั้งอุปกรณ์ต่างๆของระบบ



รูปที่ 2.3 แสดงไทมิ่งไดอะแกรมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 คุณสมบัติของอุปกรณ์ที่ใช้ในระบบควบคุมประตูดัด โนมัติ

2.3.1 อุปกรณ์ไมโครคอนโทรลเลอร์

คุณสมบัติของไมโครคอนโทรลเลอร์เบอร์ AT89C52

1. เป็นไมโครคอนโทรลเลอร์ขนาด 8 bit
2. มีวงจรออสซิลเลเตอร์ (oscillator) และวงจรผลิตสัญญาณนาฬิกา (clock oscillator) ภายในไอซี
3. มีขาสัญญาณอินพุตเอาต์พุตจำนวน 32 bit
4. สามารถเชื่อมต่อหน่วยความจำข้อมูลภายนอก (external data memory) โดยอ้างแอดเดรสได้ ถึง 64 Kbyte
5. สามารถเชื่อมต่อหน่วยความจำโปรแกรมภายนอก(external program memory) โดยอ้างแอดเดรสได้ ถึง 64 Kbyte
6. มีหน่วยความจำโปรแกรมภายในตัว (flash memory) ขนาด 8 Kbyte
7. มีหน่วยความจำข้อมูลภายในตัวขนาด 256 byte
8. หน่วยความจำข้อมูลภายในบางส่วนสามารถเข้าถึงข้อมูลระดับบิตได้ด้วย ส่งผลให้การเขียนโปรแกรมทำได้ง่ายมากขึ้น
9. มีไทมเมอร์/เคาน์เตอร์ (timer/counters) ขนาด 16 bit จำนวน 3 ตัว
10. การอินเตอร์รัปต์ (interrupt) สามารถทำได้จาก 6 แหล่งกำเนิด โดยการอินเตอร์รัปต์ยังสามารถ จัดระดับความสำคัญได้เป็น 2 ระดับ
11. มีพอร์ตสื่อสารอนุกรมภายในตัวเอง ซึ่งทำงานเป็นแบบฟูลดูเพล็กซ์ (full duplex)
12. มีชุดคำสั่งในการคำนวณทางคณิตศาสตร์และทางตรรกศาสตร์ และยังใช้ชุดคำสั่งเดียวกันกับไมโครคอนโทรลเลอร์ตระกูล MCS-51
13. มีคำสั่งโดยส่วนใหญ่ใช้เวลาการทำงานเพียง 1 ไมโครวินาที (uSec) เมื่อใช้คริสตอล (crystal) ความถี่ 12 เมกะเฮิร์ต (MHz)
14. สามารถใช้เครื่องคอมพิวเตอร์ PC ทำการลบและเขียนโปรแกรมได้โดยตรง
15. สามารถโปรแกรมทับได้นับพันครั้งเหมือนลึขการ โปรแกรมในลักษณะเดียวกันกับ EEPROM โดยการโปรแกรมสามารถโปรแกรมได้ในลักษณะ “ In - System Programming “ บนการ์ดคอนโทรล ที่เราออกแบบขึ้นมาได้เสีย หรือจะโปรแกรมโดยใช้เครื่องโปรแกรมทั่วไปเช่น Pro -100 เป็นต้น
16. ในการแก้ไขปรับปรุงโปรแกรมเดิมหรือการ โปรแกรมใหม่ สามารถทำได้โดยการ โปรแกรมทับลงไป โดยไม่ต้องลบด้วยแสงอัลตราไวโอเลต (ultraviolet)
17. ต้องการแหล่งจ่ายไฟ 5 volt เพียงชุดเดียว

โครงสร้างภายนอกของไมโครคอนโทรลเลอร์เบอร์ AT89C52

ไมโครคอนโทรลเลอร์เบอร์ AT89C52 จะมีตำแหน่งขาพื้นฐานที่เหมือนกันกับไมโครคอน

โทรลเลอร์เบอร์ 80C52 และไมโครคอนโทรลเลอร์ทั่วไป ดังแสดงในรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(T2) P1.0	1	40	VCC
(T2 EX) P1.1	2	39	P0.0 (AD0)
P1.2	3	38	P0.1 (AD1)
P1.3	4	37	P0.2 (AD2)
(SS) P1.4	5	36	P0.3 (AD3)
(MOSI) P1.5	6	35	P0.4 (AD4)
(MISO) P1.6	7	34	P0.5 (AD5)
(SCK) P1.7	8	33	P0.6 (AD6)
RST	9	32	P0.7 (AD7)
(RXD) P3.0	10	31	EA/VPP
(TXD) P3.1	11	30	ALE/PROG
(INT0) P3.2	12	29	PSEN
(INT1) P3.3	13	28	P2.7 (A15)
(T0) P3.4	14	27	P2.6 (A14)
(T1) P3.5	15	26	P2.5 (A13)
(WR) P3.6	16	25	P2.4 (A12)
(RD) P3.7	17	24	P2.3 (A11)
XTAL2	18	23	P2.2 (A10)
XTAL1	19	22	P2.1 (A9)
GND	20	21	P2.0 (A8)

รูปที่ 2.4 แสดงการจัดตำแหน่งขาต่างๆของไมโครคอนโทรลเลอร์เบอร์ AT89C52

หน้าที่การใช้งานของแต่ละขามีดังนี้

- ขา Vcc เป็นขาป้อนแรงดันไฟเลี้ยง + 5 volt
- ขา Vss เป็นขากราวด์ (ground)
- ขาพอร์ต 0 (Port 0) มี 8 ขา ได้แก่ขา P0.0 - P0.7 เป็นขาพอร์ตอินพุตเอาต์พุต (input output port) แบบ 2 ทิศทางสำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ต (input port) ต้องทำการเขียนค่า 1 ไปยังแต่ละ bit ของพอร์ต เพื่อกำหนดให้ขาพอร์ตเหล่านั้นอยู่ในสถานะปล่อยลอย(1) ซึ่งในสถานะนี้เองที่สามารถนำมาใช้เป็นพอร์ตอินพุตอิมพีแดนซ์สูงได้ นอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในการติดต่อกับหน่วยความจำภายนอกด้วย โดยทำหน้าที่การกำหนดตำแหน่งแอดเดรสไบต์ต่ำ (A0 - A7) ซึ่งจะใช้งานเป็นแบบมัลติเพล็กซ์กับการรับส่งข้อมูลขนาด 8bit (D0 - D7)
- ขาพอร์ต 1 (Port 1) มี 8 ขา ได้แก่ขา P1.0 - P1.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต และ ขาพอร์ต P1.0 และ P1.1 สามารถใช้งานเป็นขา T2 และ T2EX ตามลำดับ
- ขาพอร์ต 2 (Port 2) มี 8 ขา ได้แก่ขา P2.0 - P2.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในการติดต่อกับหน่วยความจำภายนอกด้วย โดยทำหน้าที่การกำหนดตำแหน่งแอดเดรสไบต์สูง (A8 - A15)
- ขาพอร์ต 3 (Port 3) มี 8 ขา ได้แก่ขา P3.0 - P3.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในหน้าที่พิเศษต่างๆ ดังแสดงในตารางที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ตารางที่ 1 แสดงหน้าที่พิเศษของแต่ละขาของพอร์ต P3

- ขารีสต (RST) ใช้สำหรับการรีเซ็ตการทำงานของไมโครคอนโทรลเลอร์ โดยการรีเซ็ตต้องคงสถานะเป็น 1 อย่างน้อยนาน 2 แมกซีนไซเคิล ในขณะที่ออสซิลเลเตอร์ยังทำงานอยู่
- ขา ALE/ $\overline{\text{PROG}}$ เป็นขาสัญญาณเพื่อทำหน้าที่ควบคุมการแลตช์ (Latch) ค่าตำแหน่งแอดเดรสไบต์ค่า (Address Latch Enable) เมื่อต้องการติดต่อกับหน่วยความจำภายนอก นอกจากนี้ขานี้ยังทำหน้าที่เป็นอินพุตรับพัลส์ในการรับโปรแกรม (program pulse input) ในส่วนของหน่วยความจำแบบแฟลช (Flash memory) สำหรับไมโครคอนโทรลเลอร์เบอร์ AT89C52 ที่มีหน่วยความจำโปรแกรมภายในเป็นแบบแฟลช (Flash memory)
- ขา $\overline{\text{PSEN}}$ (Program Store Enable) ทำหน้าที่เป็นสัญญาณสโตรบเพื่ออ่านคำสั่งจากหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ประมวลผลคำสั่งจากหน่วยความจำภายนอก ขานี้จะส่งสัญญาณสโตรบจำนวน 2 ครั้งในแต่ละแมกซีนไซเคิล แต่ในขณะที่ติดต่อกับหน่วยความจำภายนอกจะไม่มีการส่งสัญญาณสโตรบแต่อย่างใด
- ขา $\overline{\text{EA/VPP}}$ (External Access enable/VPP) เป็นขาสำหรับการเลือกใช้หน่วยความจำโปรแกรมจากภายในหรือจากภายนอก โดยถ้ามีสถานะเป็น 0 จะหมายถึงให้ไมโครคอนโทรลเลอร์รับคำสั่งจากหน่วยความจำภายนอกที่ตำแหน่งแอดเดรส 0-1FFFH อย่างไรก็ตามถ้าบิตป้องกัน (security bit) ในหน่วยความจำแบบแฟลช (Flash memory) ถูกโปรแกรมไว้ ไมโครคอนโทรลเลอร์จะไม่รับคำสั่งจากหน่วยความจำภายนอกเลย นอกจากนี้ขานี้ยังทำหน้าที่รับแรงดันไฟสำหรับการโปรแกรม (Vpp) ขนาด 12 โวลต์ เพื่อใช้ในระหว่างการโปรแกรม
- ขา XTAL1 และขา XTAL2 เป็นขาอินพุตและเอาต์พุตของวงจรอินเวอร์ตติ้งออสซิลเลเตอร์แอมพลิไฟเออร์ (inverting oscillator amplifier) สำหรับใช้คู่ร่วมกับคริสตัลภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การโปรแกรม AT89C52

โดยปกติแล้ว ไมโครคอนโทรลเลอร์นี้จะอยู่ในสภาวะว่าง (contact = FFH) และพร้อมที่จะได้รับการโปรแกรม ซึ่งสามารถโปรแกรมได้ทั้งแบบใช้แรงดันค่าสูง (+12V) และในแบบแรงดันค่าต่ำ (ที่ระดับแรงดัน Vcc นั่นคือ +5V) กรณีโปรแกรมในโหมดแรงดันไฟฟ้าสูง 12 V สามารถทำได้โดยการใช้เครื่องโปรแกรมทั่วไป ที่สามารถจะโปรแกรมลงใน EEPROM ได้ โดยใช้ซอฟต์แวร์ที่สามารถโปรแกรม AT89C52 ส่วนการโปรแกรมแบบค่าแรงดันค่านั้นทำให้สามารถโปรแกรมได้ในแบบ

In - System Programming ได้

ขั้นตอนการโปรแกรม

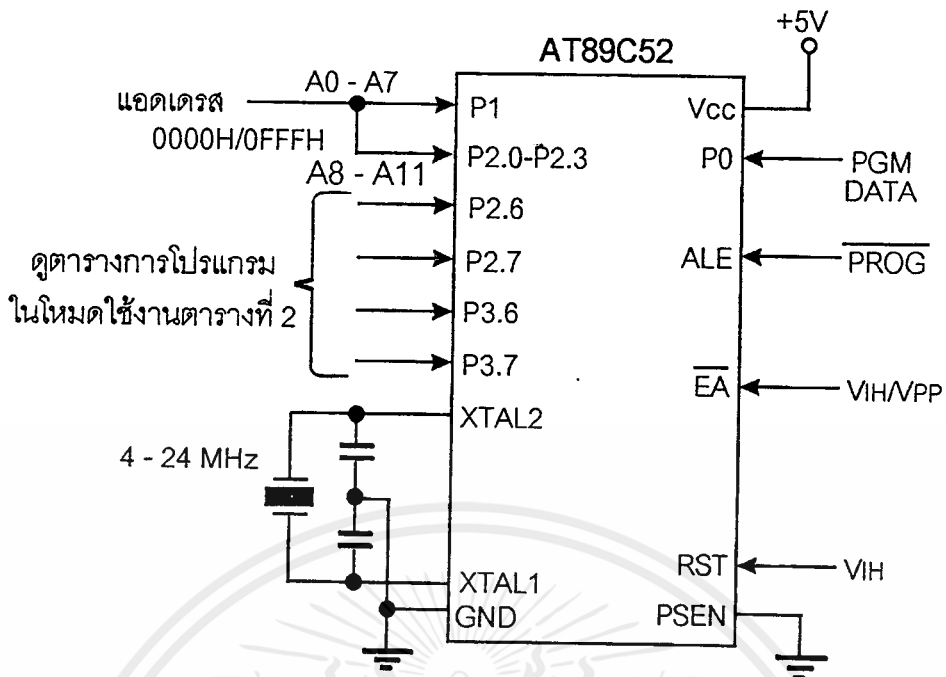
ก่อนโปรแกรม AT89C52 โลင်းแอดเดรสข้อมูลและสัญญาณควบคุม จะต้องกำหนดขึ้นตามตารางที่ 2 และจะต่อขาใช้งานและสัญญาณเข้าไปยัง AT89C52 ตามรูปที่ 2.5 และการตรวจสอบการโปรแกรมต่อตามรูปที่ 2.6

ขาค่อใช้งาน โหมดการทำงาน	RST	PSEN	ALE/ PROG	EA/ Vpp	P2.6	P2.7	P3.6	P3.7
เขียนรหัสข้อมูล	H	L		H/12V(1)	L	H	H	H
อ่านรหัสข้อมูล	H	L	H	H	L	L	H	H
การลบข้อมูลในชิป	H	L	(2)	H/12V	H	L	L	L
กำหนดไบต์การอ่าน	H	L		H	L	L	L	L

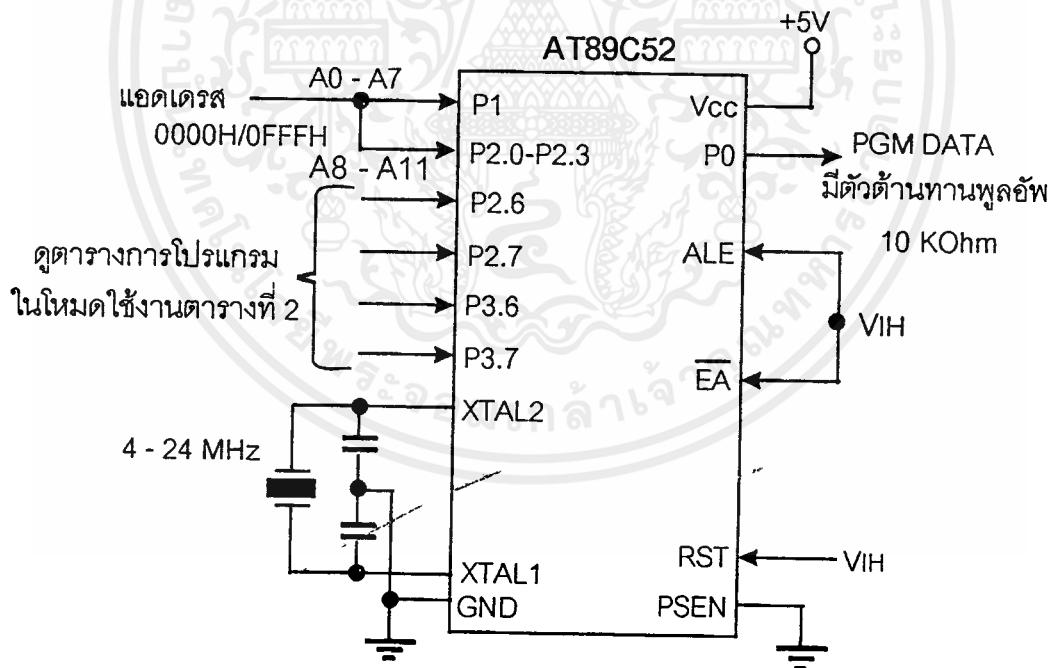
หมายเหตุ (1) ระดับแรงดันในการเขียนและลบข้อมูลที่ระดับแรงดันค่าสูง (+12V)

(2) คาบเวลาของพัลส์ในการลบข้อมูลในชิปเท่ากับ 10 มิลลิวินาที

ตารางที่ 2 แสดงการเซตค่าสัญญาณขณะทำการ โปรแกรม



รูปที่ 2.5 แสดงการต่อใช้งานขณะโปรแกรมเข้าไปใน AT89C52



รูปที่ 2.6 แสดงการทดสอบการโปรแกรมใน AT89C52

จากนั้นจะเริ่มกระทำตามขั้นตอนดังนี้

1. ป้อนค่าตำแหน่งแอดเดรสบนไลน์แอดเดรส
2. ป้อนไบต์ข้อมูลเข้าทางไลน์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ป้อนไบต์ข้อมูลเข้าทางไลน์ข้อมูล
3. กำหนดค่าสัญญาณควบคุมที่ถูกต้องเข้าทางขา P2.6 , P2.7 , P3.6 , P3.7
4. กำหนดค่าแรงดันป้อนให้ขา EA/Vpp ไว้ที่ + 12 V ในกรณีที่ค่าแรงดันสูง
5. ป้อนพัลส์ที่ขา ALE/PROG เมื่อจะโปรแกรมหนึ่งไบต์เข้าไปในหน่วยความจำแฟลช จากนั้นวงจรรอบการเขียนข้อมูลจะเกิดขึ้นเองตามมา โดยในระยะเวลาใน 1 รอบ จะไม่เกิน 1.5 มิลลิวินาทีเสร็จแล้วเริ่มทำขั้นตอนที่ 1 ถึง 5 โดยการเปลี่ยนแอดเดรสและข้อมูลชุดถัดไปจนครบทั้งหมดหรือได้รับ Object File ที่ต้องการแล้ว

หน่วยความจำภายในของ ไมโครคอนโทรลเลอร์เบอร์ AT89C52

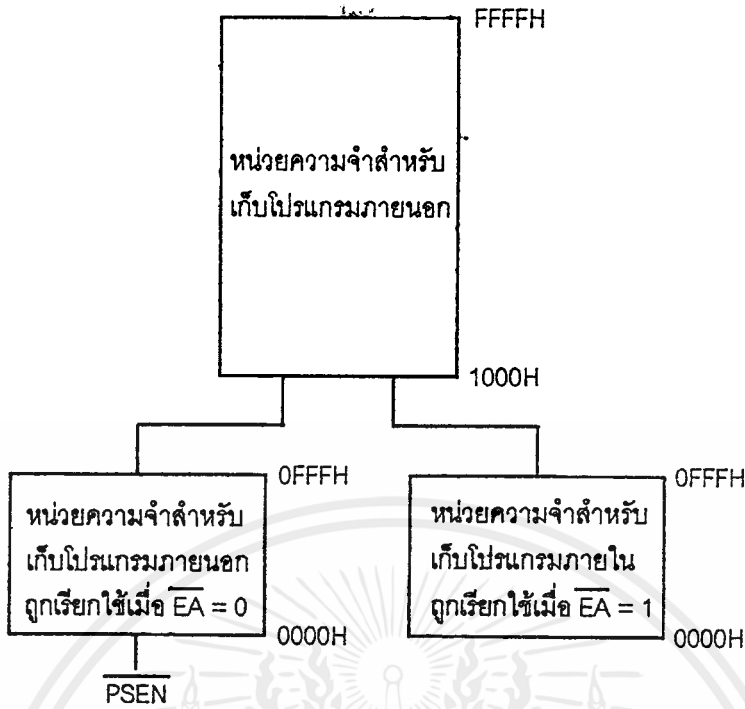
ในไมโครคอนโทรลเลอร์เบอร์ AT89C52 แบ่งชนิดหรือหน้าที่ของหน่วยความจำออกเป็น 2 ส่วนคือ หน่วยความจำโปรแกรม (program memory) และหน่วยความจำข้อมูล (data memory)

หน่วยความจำโปรแกรมจะใช้สำหรับเก็บโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์ ซึ่งเป็นหน่วยความจำแบบแฟลช (flash memory)

1 สำหรับหน่วยความจำข้อมูลจะใช้สำหรับเก็บข้อมูลหรือค่าตัวแปรต่างๆ จากการทำงานของโปรแกรม ซึ่งในเบอร์ AT89C52 จะมีหน่วยความจำในส่วนนี้เท่ากับ 256 ไบต์ ซึ่งจะเป็นหน่วยความจำแบบ RAM

หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมสามารถแบ่งออกได้เป็น 2 ส่วนคือ หน่วยความจำโปรแกรมภายในและหน่วยความจำโปรแกรมภายนอก หน่วยความจำโปรแกรมภายในถูกเลือกใช้งานถ้าขาสัญญาณ EA มีค่าเป็น 1 โดยจะถูกใช้งานในช่วงแอดเดรส 0-1FFFFH นอกเหนือจากช่วงแอดเดรสนี้จะใช้หน่วยความจำโปรแกรมภายนอกทั้งหมด ในกรณีตรงกันข้ามถ้าขาสัญญาณ EA มีค่าเป็น 0 ในช่วงแอดเดรส 0-1FFFFH จะถูกใช้จากหน่วยความจำภายนอก หรือกล่าวได้ว่าถ้าขาสัญญาณ EA มีค่าเป็น 0 จะเป็นการเลือกใช้หน่วยความจำโปรแกรมภายนอกทั้งหมดตลอดช่วงแอดเดรส ดังแสดงในรูปที่ 2.7

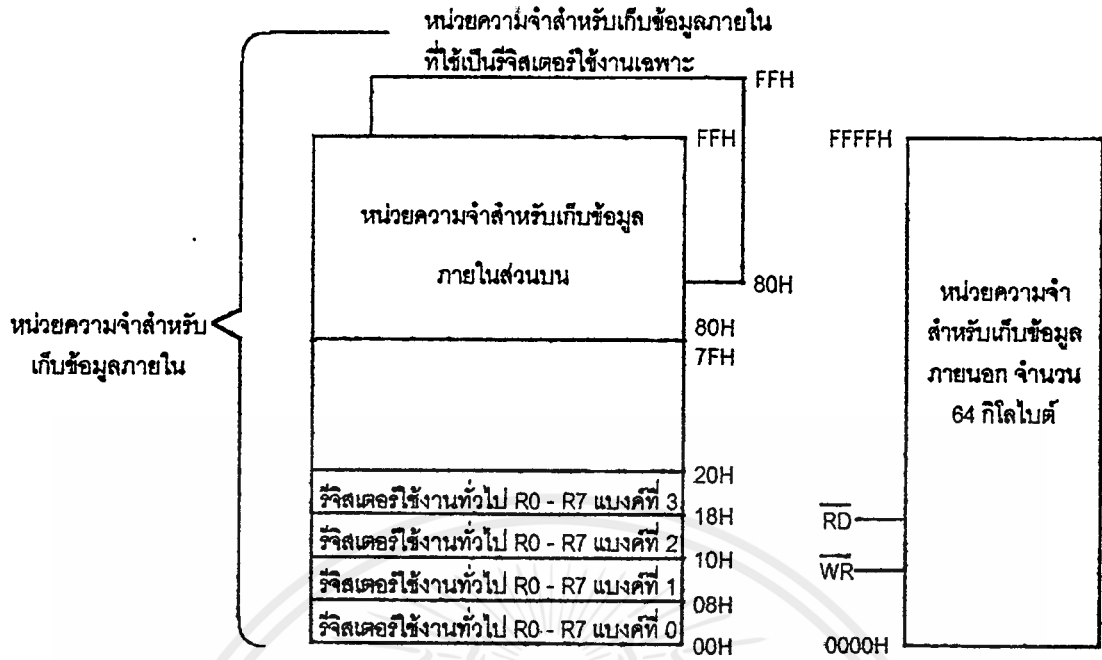


รูปที่ 2.7 หน่วยความจำสำหรับเก็บโปรแกรม

หน่วยความจำข้อมูล

หน่วยความจำข้อมูลสามารถแบ่งออกได้เป็น 2 ส่วนคือ หน่วยความจำข้อมูลภายในและหน่วยความจำข้อมูลภายนอก สำหรับหน่วยความจำข้อมูลภายในยังแบ่งออกได้เป็น 2 ส่วนย่อย คือ ส่วนที่ใช้เก็บข้อมูลทั่วไปและส่วนที่ใช้เป็นรีจิสเตอร์หน้าที่พิเศษหรือ SFR (Special Function Register) โดยส่วนที่ใช้เก็บข้อมูลทั่วไปจะถูกใช้สำหรับเก็บข้อมูลหรือค่าตัวแปรต่างๆจากการทำงานของโปรแกรม ส่วนรีจิสเตอร์ควบคุมการทำงานและบอกสถานะการทำงานของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์เบอร์ AT89C52 จะมีหน่วยความจำข้อมูลภายในขนาด 256 ไบต์ ดังแสดงในรูปที่ 2.8



รูปที่ 2.8 หน่วยความจำสำหรับเก็บข้อมูล

รีจิสเตอร์หน้าที่พิเศษ (SFR)

รีจิสเตอร์หน้าที่พิเศษมีบทบาทอย่างมากในการควบคุมการทำงานของไมโครคอนโทรลเลอร์และทำให้การเขียนโปรแกรมสามารถทำได้สะดวกมากขึ้น รีจิสเตอร์หน้าที่พิเศษทำหน้าที่สำคัญคือควบคุมการทำงานในส่วนต่างๆ ภายในไมโครคอนโทรลเลอร์และทำหน้าที่แสดงสถานะการทำงาน ซึ่งในรีจิสเตอร์หน้าที่พิเศษบางตัวยังสามารถเข้าถึงได้ในระดับบิต (bit addressable) คิวคังแสดงรูปที่ 2.9 การจัดหน่วยความจำและตำแหน่งของรีจิสเตอร์หน้าที่พิเศษต่างๆ

รีจิสเตอร์ใช้งานทั่วไป

รีจิสเตอร์ใช้งานทั่วไปมีไว้สำหรับให้ผู้เขียนโปรแกรมสามารถนำข้อมูลไปพักไว้ชั่วคราวหรือใช้งานทั่วไปได้ตามต้องการ ซึ่งรีจิสเตอร์ใช้งานทั่วไปนี้มีอยู่ด้วยกัน 8 ตัวถูกจัดให้อยู่รวมกันและมีให้เลือกใช้ถึง 4 แบงค์ (bank) นั่นคือมีรีจิสเตอร์ใช้งานทั่วไปถึง 32 ตัวให้ใช้งาน เพียงแค่การเลือกใช้รีจิสเตอร์ R0-R7 ในแบงค์ใดแบงค์หนึ่งจะถูกกำหนดจากบิต RS0,RS1 ในรีจิสเตอร์หน้าที่พิเศษ PSW ดังนั้นการเลือกใช้จึงเลือกได้เพียงแบงค์เดียวในขณะที่ขณะหนึ่ง อย่างไรก็ตามค่าข้อมูลที่เก็บไว้ในรีจิสเตอร์แบงค์ใดก็ตามที่มีชื่อเดียวกันแต่อยู่คนละแบงค์จะไม่มีผลซึ่งกันและกันเลขทำให้ผู้เขียนโปรแกรมใช้งานรีจิสเตอร์ทั่วไปนี้ได้ทั้ง 32 ตัวอย่างเต็มที่และไม่ยุ่งยากในการเขียนโปรแกรม

F8								FF
F0	B							F7
E8								EF
E0	ACC							E7
D8								DF
D0	PSW							D7
C8	T2CON		RCAPL	RCAPH	TL2	TH2		CF
C0								C7
B8	IP							BF
B0	P3							B7
A8	IE							AF
A0	P2							A7
98	SCON	SBUF						9F
90	P1							97
88	TCON	TMOD	TL0	TL1	TH0	TH1		8F
80	P0	SP	DPL	DPH			.PCON	87

↑
BIT ADDRESSABLE

รูปที่ 2.9 แสดงการจัดหน่วยความจำและตำแหน่งของรีจิสเตอร์หน้าที่พิเศษต่างๆ

2.3.2 การอินเตอร์เฟซ (Interface) กับจอแสดงผล LCD (Liquid Crystal Display) โมดูล

ในระบบที่ใช้งานไมโครคอนโทรลเลอร์ทำงานแบบระบบโดดเดี่ยว (Stand alone Microcontroller applications) บ่อยครั้งที่จำเป็นต้องติดต่อกับอุปกรณ์แสดงผลด้วย เพื่อแสดงข้อความ ตัวเลข ผลการวัดค่าหรือข้อมูลต่างๆ ในการสื่อสารกับผู้ใช้งานได้ และอุปกรณ์หนึ่งที่ถูกนำมาใช้งานกันอย่างกว้างขวาง ก็คือจอแสดงผล LCD โมดูล ตัวอย่างที่เห็นกันได้บ่อยก็คือ นำมาใช้ในการแสดงผลแบบเมนู เพื่อให้ผู้ใช้งานกดคีย์บอร์ดในการเลือกรูปแบบการทำงานต่อไป เป็นต้น

การส่งผ่านข้อมูลทั้งตัวอักษรและคำสั่งควบคุมถูกป้อนเข้ากับบัสข้อมูล 2 ทิศทางของระบบ จริงๆ แล้วการใช้งานจอแสดงผล LCD โมดูล สามารถทำงานได้ในทั้งโหมด 4 บิต ($D_0 - D_3$) หรือโหมด 8 บิต ($D_0 - D_7$) ก็ได้ แต่เนื่องจากในระบบที่เรากำลังศึกษาอยู่เป็นไมโครคอนโทรลเลอร์แบบ 8 บิต ดังนั้นการเชื่อมต่อ LCD โมดูล จึงต้องใช้งานแบบ 8 บิต กับบัสข้อมูลได้พอดี และสามารถเขียนโปรแกรมให้ทำงานในโหมดส่งผ่านข้อมูล 8 บิต กับจอ LCD ได้ทันที อย่างไรก็ตามในบางกรณีผู้อ่านอาจต้องศึกษาเพิ่มเติมการใช้งาน LCD โมดูลในโหมดส่งผ่านข้อมูล 4 บิตด้วย เช่น ถ้ามีพอร์ตเหลือใช้งานเพียงพอร์ตเดียว แต่จำเป็นต้องมีการต่อใช้งาน

พอร์ตนี้อีกกับระบบส่วนอื่น ๆ ด้วย การเลือกใช้ LCD โมดูล ให้เป็นโหมดส่งผ่านข้อมูล 4 บิต นี้ จึงเป็นวิธีช่วยลดจำนวนสายสัญญาณของพอร์ตที่มีจำกัดให้เหลือใช้ได้มากขึ้น

คำสั่งควบคุมการทำงานของ LCD โมดูล

คำสั่งควบคุมการทำงานของ LCD โมดูลที่ใช้คอนโทรลเลอร์ของฮิตาชิ ตามค่าคำสั่งนั้นมีความยาวประมาณ 30 หน้า สำหรับคำอธิบายครบทุกคุณสมบัติของ LCD โมดูล ดังนั้นในที่นี้คงจะกล่าวถึงคำสั่งและการใช้งานทั้งหมดคงไม่ไหวแน่ การนำเสนอคงทำได้เฉพาะคำสั่งที่มีความสำคัญ และควรทราบครอบคลุมการใช้งานทั่วไปได้เพียงเท่านั้น ถ้าผู้อ่านท่านใดต้องการศึกษารายละเอียดของ LCD โมดูล รุ่นใดก็คงต้องหาค่าคำสั่ง (data book) จากผู้ผลิตหรือผู้แทนจำหน่ายมาศึกษา

ความสามารถหลักที่ LCD โมดูลนิยมถูกหยิบยกมาใช้งานก็เพราะมันสามารถแสดงผลตัวอักษรแอสกี (ASCII Character) ซึ่งมีอยู่ในคอนโทรลเลอร์ของ LCD โมดูลและใน LCD โมดูลแต่ละรุ่นก็มักใช้คอนโทรลเลอร์เบอร์เดียวกันเสมอ นั่นคือคำสั่งควบคุมต่าง ๆ จึงใช้เหมือนกันใน LCD โมดูลแต่ละรุ่น เช่น ตารางชุดคำสั่งทั่วไปดังแสดงในตารางที่ 8.1 ซึ่งใช้งานได้กับ LCD โมดูลทั่วไป เช่น รุ่น H2570, LM016L, LM1612A เป็นต้น

ก่อนที่จะเริ่มต้นศึกษารูปแบบคำสั่งควบคุมต่าง ๆ ผู้อ่านควรทราบถึงคำศัพท์บางคำเกี่ยวกับองค์ประกอบและการทำงานพื้นฐานของคอนโทรลเลอร์ใน LCD โมดูลกันก่อน การทำงานภายใน LCD โมดูลจะมีบัฟเฟอร์อยู่ภายใน ซึ่งมีความจุประมาณ 80 ตัวอักษร มีชื่อเรียกว่า DD-RAM (Display Data RAM) มีตำแหน่งแอดเดรสอยู่ระหว่าง 00H ถึง 04FH ยกตัวอย่างเช่นถ้าเป็น LCD โมดูล ขนาด 16 ตัวอักษร 1 บรรทัด DD-RAM ที่จัดเก็บข้อมูลตัวอักษรที่จะแสดงผลทั้ง 16 ตัวอักษรจะถูกเก็บอยู่ที่ตำแหน่งแอดเดรส 00H ถึง 0FH โดยเริ่มต้นจากด้านซ้ายของจอแสดงผล

การทำให้เกิดช่องว่าง (window) ใน DD-RAM สามารถทำได้โดยการใช้คำสั่งชิฟต์ตัวอักษรในการแสดงผล หรืออีกวิธีหนึ่งที่ย่างมากขึ้นก็คือในขณะที่มีข้อมูลตัวอักษรเก็บไว้ในแอดเดรสค่าหนึ่ง การแสดงผลตัวอักษรต่อไปหรือทำให้เกิดช่องว่างอาจทำได้โดยการกำหนดจุดเริ่มต้นของแอดเดรสใหม่แต่แตกต่างจากเดิมไป 1 แอดเดรสของ DD-RAM ซึ่งก็ทำให้ได้ผลลัพธ์เช่นเดียวกัน

ในกรณีที่เป็นการแสดงผล LCD โมดูลแบบ 2 บรรทัดเช่น ในรุ่น LM16255 เป็นต้น บรรทัดแรกจะใช้ตำแหน่งแอดเดรสเริ่มต้นที่ 000H สำหรับตัวอักษรเริ่มต้น และในบรรทัดที่ 2 จะเริ่มต้นที่ตำแหน่งแอดเดรส 040H การเขียนโปรแกรมสำหรับ LCD โมดูลแบบ 2 บรรทัดย่อมมีความซับซ้อนมากกว่า LCD โมดูลแบบ 1 บรรทัด แต่นั่นก็ไม่ใช่อุปสรรคใหญ่หากผู้อ่านเข้าใจการควบคุมการทำงานได้ดีพอแล้ว

ผู้เขียนโปรแกรมหรือผู้ใช้งานสามารถทราบถึงการแสดงผลด้วยเคอร์เซอร์ (Cursor) ซึ่งเป็นตัวชี้ตำแหน่งต่อไปของตัวอักษรที่จะแสดงผลใน DD-RAM และตำแหน่งนี้ถูกเรียกว่าแอดเดรสเคาน์เตอร์ (Address Counter) เคอร์เซอร์สามารถกำหนดให้มีการแสดงผลหรือไม่ก็ได้ขึ้นอยู่กับเขียนโปรแกรมและกำหนดให้มีการแสดงผลแบบกะพริบได้ด้วย

โดยทั่วไปการเขียนโปรแกรมเพื่อความคมการแสดงผลของ LCD เมื่อมีการส่งข้อมูลตัวอักษรไปยัง LCD คือให้ทำการเลื่อนตัวอักษรเดิมไปและเลื่อนเคอร์เซอร์ตามไปด้วยเพื่อชี้ตำแหน่งของตัวอักษรต่อไป การเขียนโปรแกรมให้แสดงผลแบบนี้ทำให้ง่ายต่อการอ่านและเข้าใจของผู้ใช้งานได้มากที่สุด

ใน LCD โมดูลมีหน่วยความจำส่วนหนึ่งที่สำคัญก็คือ CG-RAM ซึ่งทำหน้าที่เก็บข้อมูลรายละเอียดโครงสร้างของตัวอักษรที่สามารถนำมาแสดงผลได้ หน่วยความจำของตัวอักษรที่สามารถนำมาแสดงผลได้ หน่วยความจำในส่วนนี้ผู้ใช้งานสามารถที่จะสร้างตัวอักษรใด ๆ ได้ตามต้องการแล้วเก็บลงในหน่วยความจำ ส่วนนี้สำหรับเรียกมาใช้งานได้ด้วย ถ้าหากตัวอักษรที่มีอยู่แล้วนั้นไม่เพียงพอต่อการใช้งาน การใช้งานใน ส่วนของ CG-Ram นี้สามารถดูได้จากค่าชิตของ LCD โมดูลรุ่นนั้น ๆ ที่ซื้อมาใช้งานได้

การส่งคำสั่งควบคุมไปยัง LCD โมดูลสามารถกำหนดเป็นรหัสคำสั่งต่าง ๆ ซึ่งจะกล่าวต่อไปนี้ออกที่ ตำแหน่งแอดเดรส 0C001H โดยกำหนดให้ขาสัญญาณ RS มีค่าเป็น "0" และเซตโหมดให้เป็นการอ่านหรือ เขียนได้ด้วยขาสัญญาณ R/W ต่อไปจะกล่าวถึงคำสั่งพื้นฐานที่ใช้งานบ่อยครั้งในการควบคุมการทำงานของ LCD โมดูล สำหรับเครื่องหมายดอกจัน (*) แทนสถานะของบิตที่จะเป็นสถานะใดก็ได้ที่ไม่สนใจและไม่ผลกับ คำสั่งนั้น ๆ

คำสั่งเคลียร์จอแสดงผล (Clear Display)

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
0	0	0	0	0	0	0	1

เมื่อคอนโทรลเลอร์ประมวลผลคำสั่งนี้จะทำให้ข้อมูลใน DD-RAM ทั้งหมด ถูกแทนที่ด้วยค่า 20H=space ในรหัสแอสกี ทำให้จอแสดงผลไม่ปรากฏตัวอักษรใด ๆ บนจอภาพ เคอร์เซอร์จะถูกเซตให้อยู่ที่ ตำแหน่งเริ่มต้นใหม่อีกครั้งและยกเลิกผลการใช้คำสั่งเลื่อนข้อมูล (Display Shift) ที่ผ่านมาแล้ว

คำสั่งเลื่อนเคอร์เซอร์ไปยังตำแหน่งเริ่มต้น (Return Home)

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
0	0	0	0	0	0	1	*

คำสั่งนี้มีผลทำให้รีเซตเคอร์เซอร์ให้กับไปอยู่ที่ตำแหน่งเริ่มต้นใหม่ และรีเซตคำสั่งเลื่อนข้อมูลที่ผ่านมาแล้วโดยที่ข้อมูลตัวอักษรใน DD-RAM ไม่เกิดการเปลี่ยนแปลง นั่นคือตัวอักษรบนจอแสดงผลจะยังคง เหมือนเดิมไม่เปลี่ยนแปลง

คำสั่งกำหนดโหมดป้อนข้อมูล (Entry Mode Set)

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
0	0	0	0	0	1	I/D	S

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งนี้ถูกใช้ประโยชน์ในการกำหนดการทำงานของ LCD โมดูลหลังจากเกิดการส่งไบต์ข้อมูลไปยังจอแสดงผลบิต I/D (increase/decrease bit) ทำหน้าที่กำหนดการเพิ่ม (I/D = 1) หรือลด (I/D = 0) ค่าตำแหน่งแอดเดรสใน DD-RAM 1 แอดเดรสอัตโนมัติเมื่อเกิดการอ่านหรือเขียนตัวอักษร ค่าของตำแหน่งแอดเดรสจะถูกเก็บอยู่ในแอดเดรสเคาน์เตอร์ (AC)

บิต S (Shift bit) เป็นบิตที่ใช้กำหนดลักษณะการเลื่อนข้อมูลตัวอักษรอย่างอัตโนมัติ โดยถ้าบิต S=1 เมื่อมีการส่งไบต์ข้อมูลใหม่เกิดขึ้นตัวเคอร์เซอร์จะอยู่กับที่ แต่ตัวอักษรข้อมูลเดิมจะถูกดันไปทางซ้าย แต่ถ้าบิต S = 0 เมื่อเกิดข้อมูลใหม่ตัวเคอร์เซอร์จะเลื่อนไปทางขวา

คำสั่งควบคุมการแสดงผล (Display ON/Off Control)

D ₇				D ₀			
0	0	0	0	1	D	C	B

คำสั่งนี้ใช้สำหรับควบคุมการแสดงผลในการปิดหรือเปิดการแสดงผลหน้าจอและเคอร์เซอร์ โดยปราศจากการเปลี่ยนแปลงข้อมูลใน DD-RAM ผู้เขียนโปรแกรมสามารถกำหนดหน้าจอแสดงผลให้ปิดหรือเปิดได้ด้วยบิต D โดยกำหนดให้บิต D = 1 เป็นการเปิดจอแสดงผลและถ้า D = 0 เป็นการปิดจอแสดงผล เช่นเดียวกันสำหรับบิต C = 0 และบิต C = 1 เป็นการควบคุมให้เคอร์เซอร์เปิดหรือปิดตามลำดับ และบิต B ซึ่งเป็นบิตกำหนดว่าจะให้เคอร์เซอร์กะพริบหรือไม่

คำสั่งควบคุมการเลื่อนเคอร์เซอร์และตัวอักษร (Cursor or Display Shift)

D ₇				D ₀			
0	0	0	1	S/C	R/L	*	*

คำสั่งนี้ใช้สำหรับควบคุมการเลื่อนของเคอร์เซอร์และตัวอักษรที่แสดงผล ซึ่งมีความสำคัญและใช้งานบ่อยเนื่องจากผู้เขียนโปรแกรมต้องแสดงผลข้อความบนจอแสดงผลตามแนวนอน

คำสั่งเซตฟังก์ชัน (Function Set)

D ₇				D ₀			
0	0	1	DL	N	F	*	*

คำสั่งนี้ใช้สำหรับเซตโหมดการทำงานของ LCD โมดูลหลังจากรีเซตการทำงานหรือเริ่มต้นทำงานระบบทุกครั้ง ในที่นี้เราจะใช้ LCD โมดูลในโหมด 8 บิตและใช้งานเพียง 1 บรรทัดเท่านั้น ความหมายของบิตต่างๆมีดังนี้

บิต DL = 1 หมายถึงทำงานในโหมดอินเตอร์เฟซแบบ 8 บิต

บิต DL = 0 หมายถึงทำงานในโหมดอินเตอร์เฟซแบบ 4 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต N = 1 หมายถึงทำงานแบบ 1 บรรทัด

บิต N = 0 หมายถึงทำงานแบบ 2 บรรทัดหรือมากกว่า

บิต F = 0 หมายถึงทำงานในโหมดความละเอียด 5*7 จุด

บิต F = 1 หมายถึงทำงานในโหมดความละเอียด 5*10 จุด

คำสั่งเซตตำแหน่งแอดเดรสใน CG-RAM (Set CG-RAM Address)

D ₇							D ₀
0	1	a5	a4	a3	a2	a1	a0

คำสั่งนี้ใช้สำหรับกำหนดค่าตำแหน่งแอดเดรสใน CG-RAM ภายใน LCD โมดูลเพื่อทำการส่งผ่านข้อมูลโดยกำหนดค่าตำแหน่งแอดเดรสใน CG-RAM แนนอนค่าหนึ่งเพื่อถ่ายทอคไบต์ข้อมูลต่อไป บิต a0-a5 เป็นบิตกำหนดตำแหน่งใน CG-RAM ซึ่งจะถูกโหลดไปเก็บไว้ในแอดเดรสเคาน์เตอร์ AC

คำสั่งเซตตำแหน่งแอดเดรสใน DD-RAM (Set DD-RAM Address)

D ₇							D ₀
1	a6	a5	a4	a3	a2	a1	a0

คำสั่งนี้ใช้สำหรับกำหนดค่าตำแหน่งแอดเดรสใน DD-RAM ภายใน LCD โมดูลเพื่อทำการส่งผ่านข้อมูลโดยกำหนดค่าตำแหน่งแอดเดรสใน DD-RAM แนนอนค่าหนึ่งเพื่อถ่ายทอคไบต์ข้อมูลต่อไป บิต a0-a6 เป็นบิตกำหนดตำแหน่งแอดเดรสใน DD-RAM ซึ่งจะถูกโหลดไปเก็บไว้ในแอดเดรสเคาน์เตอร์ AC

คำสั่งอ่านแฟล็กบิวซี (Read Busy Flag)

D ₇							D ₀
BF	a6	a5	a4	a3	a2	a1	a0

เมื่อต้องการอ่านสถานะของแฟล็กบิวซีต้องกำหนดคำสั่งสัญญาณ R/W เป็น "1" เสมอ แฟล็กบิวซีเป็นตัวบอกสถานะการทำงานของ LCD โมดูลว่าอยู่ในสถานะพร้อมรับข้อมูลหรือไม่ โดยถ้า BF = 1 หมายถึงขณะนี้ LCD โมดูลยังไม่พร้อมรับข้อมูลต่อไป อันเนื่องมาจากกระบวนการประมวลผลจากคำสั่งหรือไบต์ข้อมูลที่ผ่านมายังไม่เสร็จสิ้นแต่ถ้า BF = 0 หมายถึงขณะนี้ LCD โมดูลพร้อมที่จะรับคำสั่งใหม่หรือข้อมูลใหม่ได้แล้ว นอกจากนี้คำสั่งนี้ทำให้ทราบสถานะของแฟล็กบิวซีแล้วในขณะเดียวกันที่บิต a0-a6 จะถูกอ่านด้วยซึ่งเป็นค่าที่เก็บอยู่ในแอดเดรสเคาน์เตอร์ AC

คำสั่งเขียนข้อมูล ไปยัง CG หรือ DD-RAM (Write Data to CG or DD-RAM)

D ₇				D ₀			
d7	d6	d5	d4	d3	d2	d1	d0

เมื่อต้องการใช้คำสั่งนี้ต้องกำหนดให้ขาสัญญาณ R/W = 0 และ RS = 1 เสมอ คำสั่งนี้ใช้สำหรับเขียนข้อมูลไปยัง CG หรือ DD-RAM ในตำแหน่งแอดเดรสที่ต้องการ ซึ่งขึ้นอยู่กับกำหนัดตำแหน่งแอดเดรสใน CG หรือ DD-RAM ก่อนหน้าที่จะส่งข้อมูลนี้แล้ว คำสั่งนี้จะมีผลเกี่ยวเนื่องถึงการกำหนดโหมดการทำงานก่อนหน้านี้ด้วยว่าจะให้ทำการเพิ่มหรือลดค่าใน AC หลังจากส่งผ่านไบต์ข้อมูลแล้ว ซึ่งเป็นผลมาจากคำสั่งกำหนดโหมดป้อนข้อมูล (Entry mode command)

คำสั่งอ่านข้อมูลจาก CG หรือ DD-RAM (Read Data from CG or DD-RAM)

D ₇				D ₀			
d7	d6	d5	d4	d3	d2	d1	d0

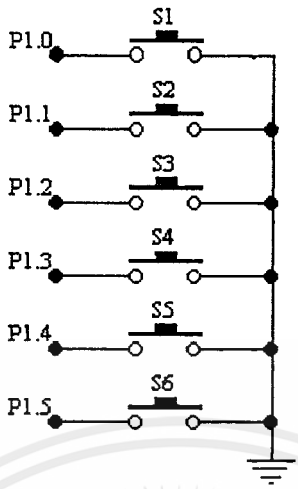
เมื่อต้องการใช้คำสั่งนี้ต้องกำหนดให้ขาสัญญาณ R/W = 1 และ RS = 1 เสมอ คำสั่งนี้ใช้สำหรับอ่านข้อมูลจาก CG หรือ DD-RAM ในตำแหน่งแอดเดรสที่ต้องการซึ่งขึ้นอยู่กับกำหนัดตำแหน่งแอดเดรสใน CG หรือ DD-RAM ก่อนหน้าที่จะใช้คำสั่งอ่านข้อมูลนี้แล้ว

2.3.3 การอินเตอร์เฟสกับคีย์บอร์ด

โครงการที่ถูกสร้างขึ้นส่วนมากแล้วจำเป็นต้องติดต่อกับอุปกรณ์อินพุต เช่น ตัวเซ็นเซอร์ต่าง ๆ , สัญญาณไฟฟ้า, คีย์บอร์ด เป็นต้น โดยเฉพาะคีย์บอร์ดที่ใช้ติดต่อกับผู้ใช้งานในการควบคุมเครื่อง ดังนั้นมันจึงจำเป็นต้องมีในเกือบทุก ๆ โครงการที่สร้างขึ้นเพื่อการพัฒนาโครงการของคุณได้อย่างรวดเร็ว ก่อนที่จะจบในบทสุดท้ายนี้จึงขอกว่าถึงเทคนิคการอินเตอร์เฟสกับคีย์บอร์ด 6 รูปแบบที่ใช้งานกันอยู่ทั่วไปสำหรับตรวจจับสถานะการเปิดและปิดการสวิตซ์ในแต่ละรูปแบบเป็นหลักเพื่อให้สามารถมองเห็นแนวทางการเขียนโปรแกรมได้ ซึ่งแต่ละรูปแบบการเชื่อมต่อสวิตซ์ก็จะมีข้อดีและข้อเสียแตกต่างกันไป



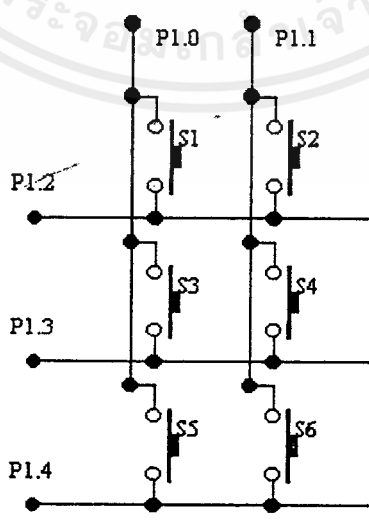
แบบเชื่อมต่อสวิตช์โดยตรงกับพอร์ต



รูปที่ 2.10 แสดงการเชื่อมต่อสวิตช์โดยตรงกับพอร์ต

รูปที่ 2.10 แสดงการเชื่อมต่อสวิตช์โดยตรงกับพอร์ตซึ่งเป็นแบบที่ง่ายที่สุด การกดสวิตช์แต่ละตัวจะทำให้ขาพอร์ตนั้นถูกต่อลงกราวด์โดยตรง ซึ่งไม่จำเป็นต้องมีวงจรจำกัดกระแสก็ได้เพราะว่าภายในตัวไมโครคอนโทรลเลอร์มีตัวต้านทานพูลอัพต่ออยู่แล้ว การเขียนโปรแกรมทำได้ง่ายมากโดยมากใช้คำสั่งตรวจสอบสถานะของแต่ละบิตในพอร์ต เช่นคำสั่ง JB หรือ JNB ข้อเสียของวงจรแบบนี้คือสิ้นเปลืองจำนวนขาพอร์ตนานานมาก ถ้าต้องการใช้สวิตช์มากเท่าใดก็ต้องขยายพอร์ตให้มากตามไปด้วย ซึ่งเป็นเรื่องยุ่งยาก เช่น ถ้าต้องการออกแบบเป็นคีย์บอร์ดสำหรับป้อนตัวอักษรขนาด 60 คีย์ ผู้ออกแบบต้องจัดสร้างขาพอร์ตนี่ให้ได้ถึง 60 ขาพอร์ตนี่ ซึ่งแทบจะเป็นไปไม่ได้ยากและสิ้นเปลืองอุปกรณ์มาก

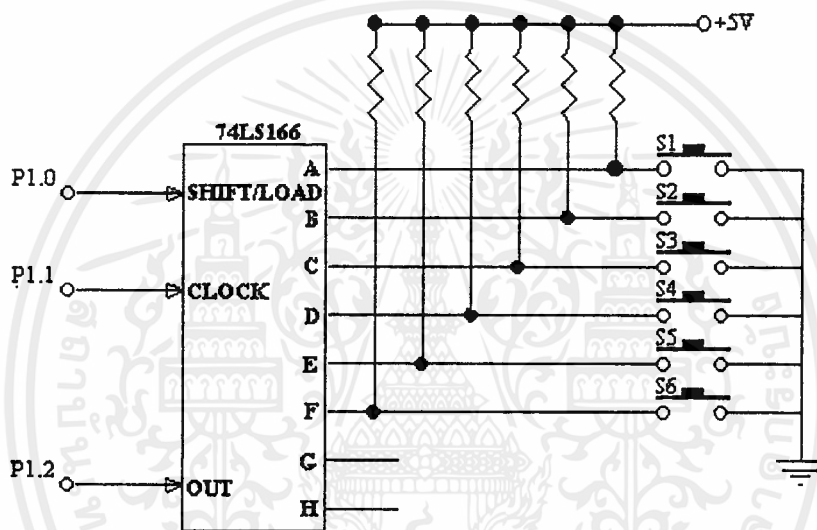
แบบเชื่อมต่อสวิตช์แบบเมตริกซ์



รูปที่ 2.11 แสดงการเชื่อมต่อสวิตช์แบบเมตริกซ์

รูปที่ 2.11 แสดงการเชื่อมต่อสวิทช์แบบเมตริกซ์ สวิทช์แต่ละตัวจะถูกเชื่อมต่อกันแบบแถวและคอลัมน์ในรูปแบบของเมตริกซ์ การตรวจสอบการกดคีย์โดยบนคีย์บอร์ดทำได้โดยการป้อนค่าตรวจสอบค่าหนึ่งไปยังด้านคอลัมน์และตรวจสอบการเปลี่ยนแปลงที่เกิดขึ้นทางด้านแถว หรือกล่าวได้ว่าตำแหน่งของการกดคีย์ได้จากการเขียนโปรแกรมสแกนคีย์ด้านแถว ข้อดีของการเชื่อมต่อสวิทช์แบบเมตริกซ์คือสามารถใช้สวิทช์ได้มากขึ้นในขณะที่สิ้นเปลืองขาพอร์ตจำนวนน้อยเช่น ใช้ขาพอร์ตเพียง 16 ขาพอร์ต สามารถต่อคีย์บอร์ดได้ถึง 64 คีย์ ข้อเสียของวงจรนี้คือไม่สามารถรับการกดคีย์พร้อมกันได้และต้องเขียนโปรแกรม ในการตรวจสอบคีย์ที่ยุ่งยากซับซ้อน

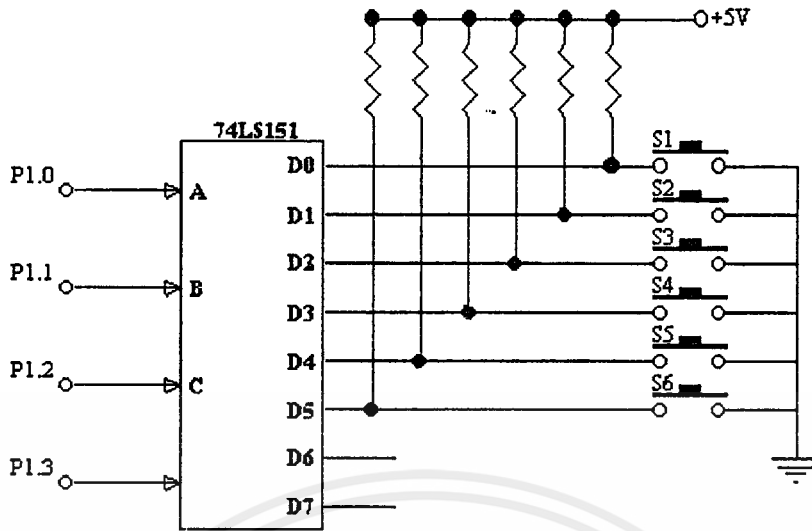
แบบเชื่อมต่อสวิทช์ผ่านชิพตรีจิสเตอร์



รูปที่ 2.12 แสดงการเชื่อมต่อสวิทช์ผ่านชิพตรีจิสเตอร์

รูปที่ 2.12 แสดงการเชื่อมต่อสวิทช์ผ่านชิพตรีจิสเตอร์จะเห็นได้ว่าวงจรนี้ต้องการขาพอร์ตน้อยมากเพียง 3 ขาพอร์ต โดยใช้ขาพอร์ตหนึ่งกำหนดโหมดที่ขา SHIFT/LOAD ให้อยู่ในสถานะชิพข้อมูลและให้กำเนิดพัลส์จำนวน 8 พัลส์ เพื่อทำการชิพข้อมูลสถานะของสวิทช์ผ่านขาพอร์ต P_{1,2} ไปใช้งาน นั่นคือใช้ขาพอร์ตเพียง 3 ขาพอร์ตเท่านั้นก็ทำงานได้แล้ว รวมทั้งการเขียนโปรแกรมควบคุมก็ทำได้ง่าย ถ้าต้องการเพิ่มจำนวนคีย์มากขึ้นทำได้โดยเพิ่มชิพตรีจิสเตอร์มากขึ้นให้เท่าเทียมกัน อย่างไรก็ตามข้อเสียที่เกิดขึ้นคือใช้เวลาในการอ่านสถานะของคีย์ทั้งหมดเป็นเวลานานจนกระทั่งชิพข้อมูลได้ครบตามจำนวนคีย์

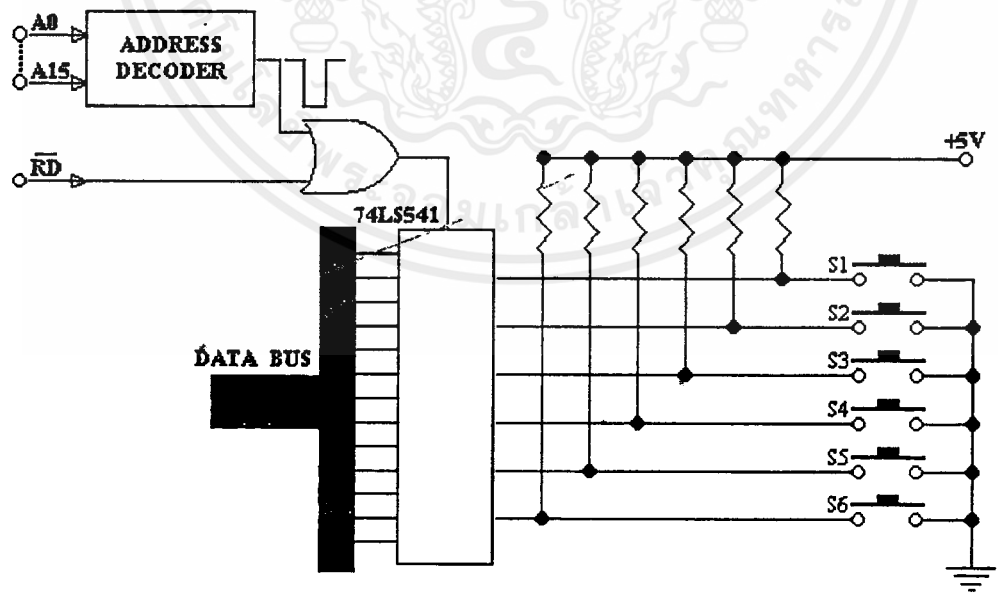
แบบเชื่อมต่อสวิทช์แบบมัลติเพล็กซ์



รูปที่ 2.13 แสดงการเชื่อมต่อแบบมัลติเพล็กซ์เซอร์

วงจรการเชื่อมต่อแบบนี้แสดงในรูปที่ 2.13 ซึ่งต้องการขาพอร์ตจำนวน 4 ขาพอร์ต พอร์ต P_{1.0}, P_{1.1} และ P_{1.2} ทำหน้าที่ควบคุมไอซีมัลติเพล็กซ์เซอร์เพื่อเลือกสวิทช์ s₁-s₆ ที่ต้องการจะติดต่อกับ สถานะของคีย์ที่ถูกเลือกจะถูกส่งกลับไปยังไมโครคอนโทรลเลอร์ผ่านขาพอร์ต P_{1.3} วงจรแบบนี้สามารถเพิ่มจำนวนคีย์ได้ง่ายและทำได้เป็นจำนวนมาก อีกทั้งการเขียนโปรแกรมก็ทำได้ง่ายเช่นเดียวกัน

แบบเชื่อมต่อสวิทช์เข้ากับระบบบัสข้อมูล

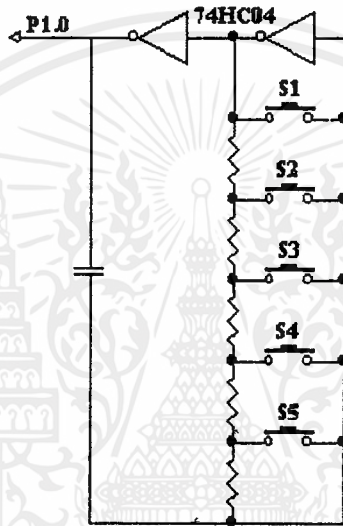


รูปที่ 2.14 แสดงการเชื่อมต่อในระบบบัสข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรการเชื่อมต่อแบบนี้แสดงในรูปที่ 2.14 ซึ่งทำการเชื่อมต่อสวิตช์เข้ากับระบบบัสข้อมูลเพื่อทำการอ่านสถานะของสวิตช์เข้าสู่ระบบบัสข้อมูลของไมโครคอนโทรลเลอร์ การเชื่อมต่อแบบนี้มักถูกนำมาใช้เมื่อไม่มีขาพอร์ตเหลือไว้ใช้งานเลย การติดต่อกับคีย์บอร์ดทำได้โดยการอ้างตำแหน่งแอดเดรสไปยังวงจรแอดเดรสดีโคเดอร์ และทำการแอกตีฟสัญญาณ RD เพื่อส่งสัญญาณอินาเบิลไปยัง ไอซีบัพเฟอร์ส่งผ่านสถานะของคีย์ต่าง ๆ เข้าสู่ระบบบัสข้อมูล ถึงแม้ว่าถ้าพิจารณาดูที่การเขียนโปรแกรมจะเห็นได้ว่าไม่ยากเลยแต่ว่าในทางปฏิบัติการต่อวงจรแบบนี้จะทำให้เกิดความยุ่งยากในทางฮาร์ดแวร์มาก เพราะจะต้องเชื่อมต่อวงจรเข้ากับทั้งระบบบัสแอดเดรสและบัสข้อมูล

แบบเชื่อมต่อสวิตช์เข้ากับวงจรมัลติไวเบรเตอร์



รูปที่ 2.15 แสดงการเชื่อมต่อแบบมัลติไวเบรเตอร์

วงจรการเชื่อมต่อสวิตช์รูปแบบนี้ ก็คือแบบเชื่อมต่อสวิตช์เข้ากับวงจรมัลติไวเบรเตอร์ (multivibrator) ดังแสดงในรูปที่ 2.15 การใช้งานต้องการขาพอร์ตเพียง 1 ขาพอร์ตเท่านั้น การจับวงจรอยู่ในรูปของวงจรถอสซิลเลเตอร์โดยใช้อินเวอร์เตอร์เกิดจำนวน 2 ตัว ผลิตความถี่ออกมาค่าหนึ่งซึ่งความถี่นี้จะเกิดการเปลี่ยนค่าไปเมื่อมีการกดปุ่มใดปุ่มหนึ่ง การตรวจจับว่าเกิดการกดสวิตช์ตัวใดทำได้โดยการวิเคราะห์จากโปรแกรมที่เขียนขึ้น ซึ่ง จะทำการวัดค่าความถี่ที่ปรากฏที่ขาพอร์ต P_{1,0} และนำมาเปรียบเทียบกับค่าความถี่ที่ใช้แทนสวิตช์แต่ละตัวที่กำหนดไว้ใน โปรแกรมแล้ว

ถึงแม้ว่าวงจรมีต้องการขาพอร์ตน้อยที่สุดจากวงจรต่าง ๆ ที่กล่าวมาทั้งหมด(ยกเว้นแบบเชื่อมต่อกับระบบบัสข้อมูล) แต่วงจรมีข้อเสีย 2 ประการ คือ จำเป็นต้องเขียน โปรแกรมควบคุมการทำงานที่ยุ่งยากซับซ้อน และการเพิ่มจำนวนคีย์ทำได้ไม่มากนัก เพราะข้อจำกัดในการแบ่งความถี่ที่แตกต่างกันสำหรับคีย์แต่ละตัว

2.3.4 คุณลักษณะของของอุปกรณ์อาร์ทีซี(Real Time Clock)

DS1202 เป็นตัวเก็บข้อมูลเวลาซึ่งจะมีฝั่งเวลา/เดือน และหน่วยความจำสแตติกแรมจำนวน 24 ไบต์ อยู่ในตัวซึ่งสามารถที่จะติดต่อกับไมโครคอนโทรลเลอร์โดยการติดต่อแบบอนุกรม ฝั่งเวลา/เดือนของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1202 จะเป็นตัวผลิตข้อมูลวินาที,นาฬิกา,ชั่วโมง,วัน,วันที่,เดือนและปี ข้อมูลเกี่ยวกับเวลาภายใน DS1202 สามารถเลือกรูปแบบได้โดยมีแบบ 24 ชั่วโมงกับ 12 ชั่วโมง

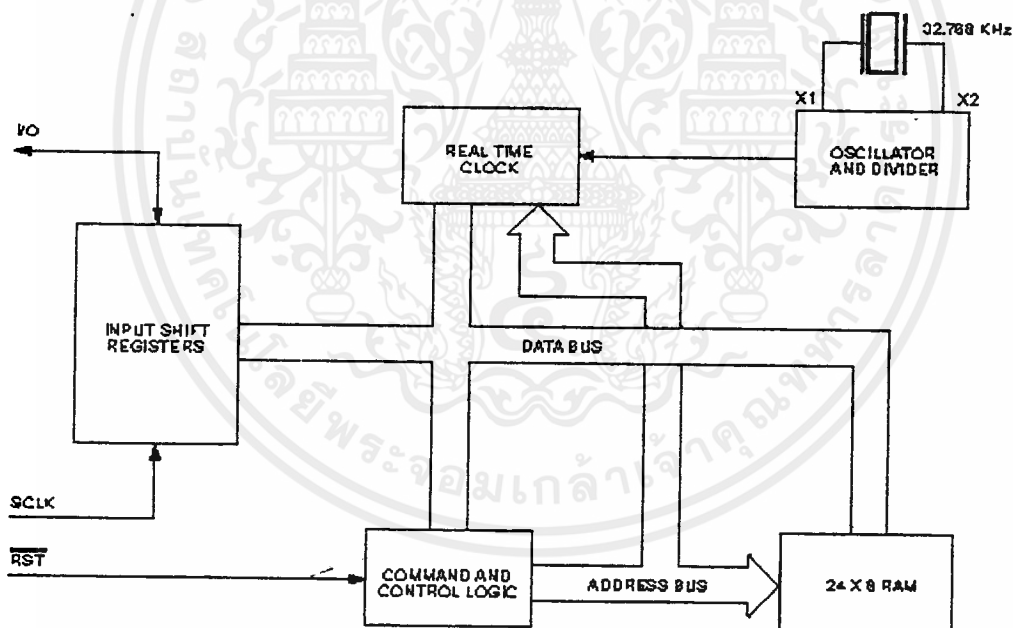
การแลกเปลี่ยนข้อมูลของ DS1202 นั้นจะใช้การแลกเปลี่ยนข้อมูลแบบซิงโครไนซ์ซีเรียลคอมมูนิเคชัน (Synchronized Serial Communication) ซึ่งจะใช้สายเพียงสามเส้น ในการแลกเปลี่ยนข้อมูล โดยการแลกเปลี่ยนข้อมูลนั้นสามารถแลกเปลี่ยนได้ในแบบ 1 ไบต์ของ 1 ข้อมูล หรือจะแลกเปลี่ยนแบบขบวนข้อมูล (Burst) โดยการแลกเปลี่ยนครั้งละ 24 ไบต์

การทำงาน

ส่วนประกอบหลักๆของ DS1202 ประกอบไปด้วยชิพรีจิสเตอร์,คอนโทรลลอจิก,ออสซิลเลเตอร์,Real Time Clock และหน่วยความจำข้อมูลแรม

การเริ่มต้นการทำงานจะต้องทำการป้อนข้อมูลให้กับขา RST เป็น i เสียก่อนแล้วจึงทำการป้อนข้อมูลจำนวน 8 บิตเข้าไปในชิพรีจิสเตอร์โดยข้อมูลจะประกอบไปด้วยแอดเดรสและคำสั่งควบคุม

หลังจากที่ป้อนข้อมูลเข้าไปแล้ว DS1202 ก็จะทำการผลิตข้อมูลเกี่ยวกับเวลาจำนวน 8 บิตซึ่งข้อมูลที่ผลิตขึ้นมาจะถูกส่งไปยังชิพรีจิสเตอร์และข้อมูลเวลาจะถูกส่งออกไปเมื่อต้องการอ่านข้อมูลและสามารถที่จะเปลี่ยนเวลาได้โดยการเขียนข้อมูลเข้าไป



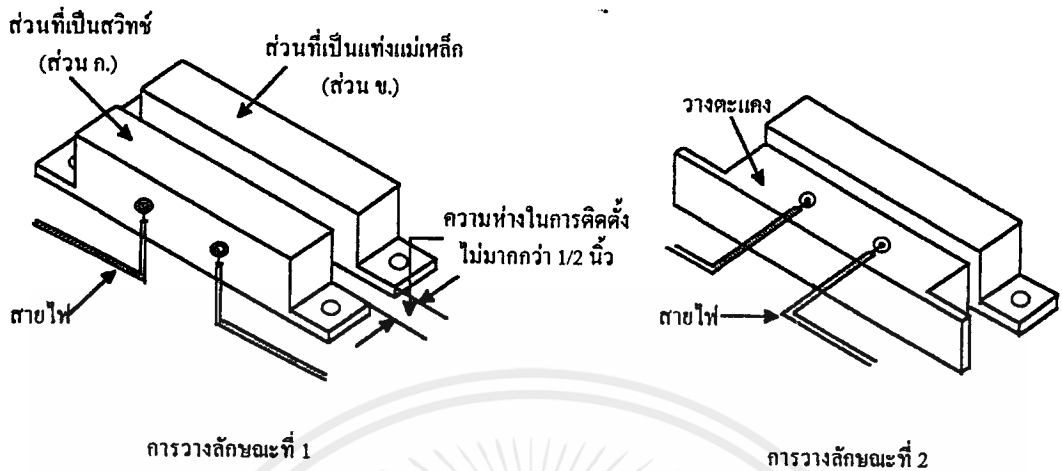
รูปที่ 2.16 แสดงบล็อกไดอะแกรมของอุปกรณ์อาร์ทีซี

2.3.5 คุณสมบัติของสวิทช์แม่เหล็ก (Magnetic switch)

สวิทช์แม่เหล็กนี้เป็นสวิทช์ที่ติดตั้งง่าย มีความแน่นอนสูง มีทั้งชนิด NO และชนิด NC การติดตั้งจะติดตั้งจะติดตั้งส่วนที่จะซ่อนแม่เหล็กไว้ในส่วนที่เคลื่อนไหวได้ ส่วนตัวสวิทช์จะติดอยู่กับส่วนคงที่และต่อสายเป็นวงจรรอกมา หากใช้ในระบบกันขโมยอาจต้องซ่อนสวิทช์เหล่านี้ไว้ ขณะที่ทั้งสองส่วนนี้ประกอบอยู่ใกล้กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กัน สวิตช์จะถูกอำนาจแม่เหล็กกระทำอยู่ แต่ถ้าเมื่อไรชิ้นส่วนแม่เหล็กเคลื่อนออกสวิตช์ก็จะเปลี่ยนตำแหน่งสั่งงานไปที่แผงควบคุมทันที ตัวอย่างสวิตช์ชนิดนี้และวิธีการติดตั้งแสดงไว้ดังรูปที่ 2.17



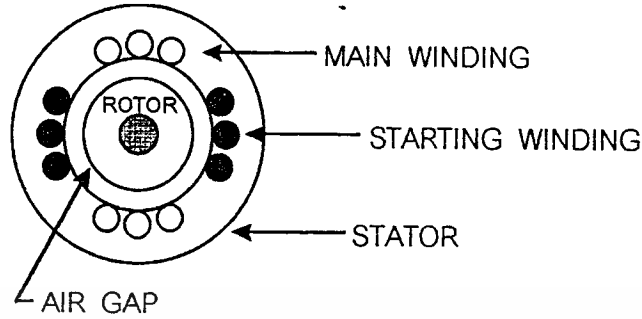
รูปที่ 2.17 แสดงตัวอย่างสวิตช์แม่เหล็ก

2.3.6 มอเตอร์เหนี่ยวนำเฟสเดียว (Single - Phase Induction Motors)

มอเตอร์เหนี่ยวนำเฟสเดียว เป็นมอเตอร์ที่มีโครงสร้างและหลักการทำงานในการที่ทำให้มันหมุนได้คล้ายกับมอเตอร์เหนี่ยวนำหลายเฟส จะแตกต่างกันตรงที่ตัวอยู่กับที่มีการพันขดลวดเป็นแบบเฟสเดียวและได้รับไฟจากไฟเฟสเดียวด้วย ดังนั้นจึงทำให้เกิดสนามแม่เหล็กสลับไปมาขึ้นที่ตัวอยู่กับที่ซึ่งไม่ใช่สนามแม่เหล็กหมุนที่ความเร็ว synchronous เหมือนอย่างกับพวกที่พันแบบสองเฟสหรือแบบสามเฟส จากเหตุผลนี้จึงทำให้เส้นแรงแม่เหล็กที่สลับไปมานี้ไม่สามารถที่จะทำให้เกิดแรงบิดหมุนขึ้นได้ในขณะที่ตัวหมุนยังอยู่ นี่คือสาเหตุว่ามอเตอร์เฟสเดียวจึงไม่มีแรงบิดเริ่มแรกในตัวของมันเอง

ตัวหมุน (rotor) ของมอเตอร์แบบนี้ส่วนใหญ่เป็นแบบกรงกระรอก (squirrel cage) ส่วนทางตัวอยู่กับที่นั้นก็มีการพันขดลวดตัวนำคิดแยกแตกต่างกันไปบ้างแล้วแต่ละชนิดของมอเตอร์ การที่กล่าวว่ามีมอเตอร์เฟสเดียวมีโครงสร้างและหลักการทำงานในการที่ทำให้มันหมุนได้คล้ายกับมอเตอร์เหนี่ยวนำหลายเฟสนั้นหมายความว่ามอเตอร์แบบนี้มีทั้งตัวอยู่กับที่และตัวหมุน โดยตัวอยู่กับที่นั้นจะมีการพันขดลวดเพื่อทำให้เกิดสนามแม่เหล็กหมุนในช่วงขณะเริ่มแรกหรือมีอยู่ตลอดไปก็ได้โดยอาศัยอุปกรณ์ช่วยอื่นๆ เช่น ขดเริ่มทำให้มันหมุน (starting winding) หรือตัวขั้วประจุ (capacitor) ต่อร่วมกับตัวทำให้มันหมุน หรืออาจทำให้เกิดสนามแม่เหล็กเลื่อนหรือเคลื่อนที่ไปรอบหรือเกิดสนามแม่เหล็ก โดยอาศัยคุณสมบัติของตัวหมุนและตัวอยู่กับที่ร่วมกันก็ได้ ส่วนตัวหมุนนั้นก็อาจเป็นแบบกรงกระรอกหรืออาจไม่เป็นก็ได้แต่ก็ต้องเป็นตัวที่จะทำให้เกิดมีการเหนี่ยวนำขึ้นในตัวมันเพื่อทำให้เกิดแรงบิดขึ้นได้เมื่อมีสนามแม่เหล็กหมุนหรือเลื่อนได้ผ่านตัวมัน ซึ่งจะกล่าวเป็นเรื่องๆไปตามชนิดของมอเตอร์แบบต่างๆดังนี้

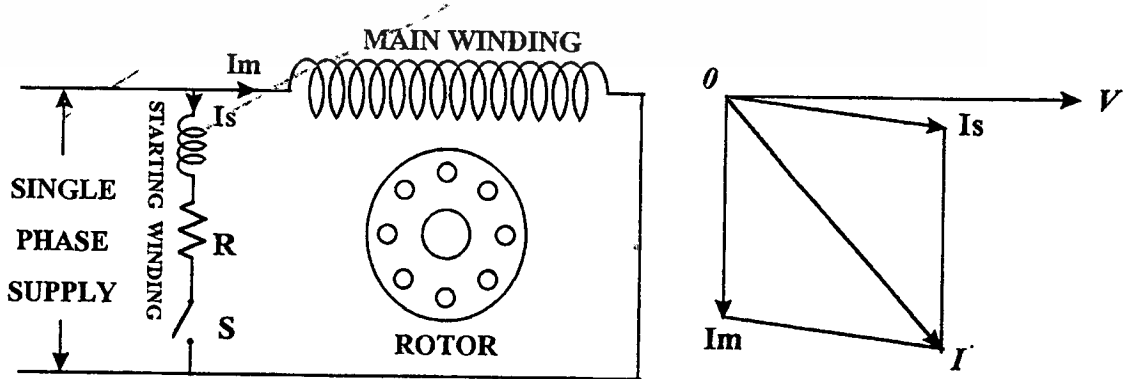
(ก) มอเตอร์ที่แยกเฟสออกจากกัน (Split - Phase Motor) โครงสร้างของมอเตอร์ประกอบด้วย



รูปที่ 2.18 แสดงรูปตัดของมอเตอร์

1. ตัวอยู่กับที่ (stator) มีขดลวดอยู่ 2 ชุด ชุดหนึ่งมีไว้สำหรับใช้งานตามปกติเรียกว่าชุดสำหรับวิ่ง (running winding หรือ main winding) ชุดขดลวดชุดนี้มีค่าความต้านทานต่ำแต่ reactance สูงกว่าชุดสำหรับทำให้มอเตอร์เริ่มหมุน โดยที่มีขนาดเส้นลวดใหญ่กว่าชุดหมุน ชุดสำหรับวิ่งหรือ ใช้งานนี้เป็นชุดหลักที่มีไฟผ่านอยู่ตลอดเวลา ไม่ว่าจะเป็นขณะเริ่มหมุนที่แรกหรือขณะที่ทำงานปกติ ชุดที่สองเป็นชุดสำหรับเริ่มทำให้หมุน (starting winding) ขดลวดชุดนี้พันไว้ที่ตัวอยู่กับที่เช่นกัน แต่จะให้ลงในช่อง (slot) เป็นมุมห่างไปจากชุดหลัก 90° ไฟฟ้า ทั้งนี้เพื่อทำให้เกิดสนามแม่เหล็กหมุนขึ้นในขณะเริ่มแรก ชุดนี้มีความต้านทานของขดลวดตัวนำสูงแต่ reactance ต่ำและเส้นลวดมีขนาดเล็ก ขดลวดชุดนี้ต่ออนุกรมกับสวิตช์หนีศูนย์กลาง (centrifugal switch) แล้วจึงนำไปต่อขนานกับชุดสำหรับวิ่ง ถ้าไม่ใช้สวิตช์หนีศูนย์กลางก็อาจใช้ relay เป็นตัวตัดชุดเริ่มทำให้หมุนหรือชุดช่วยทำให้หมุน (starting winding or auxiliary winding) ก็ได้ ดังเช่นที่ใช้กับมอเตอร์อัดน้ำยา (compressor) ของตู้เย็นหรือเครื่องปรับอากาศก็ได้

2. ตัวหมุน (rotor) เป็นแบบกรงกระรอกธรรมดาซึ่งอาจจะมี 2 กรงซ้อนกันหรือกรงเดียวก็ได้แล้วแต่จุดประสงค์การใช้งาน ตัวนำที่ฝังอยู่ในตัวหมุนเป็นรูปกรงกระรอกนี้อาจจะฝังเฉียงๆ (shew) ก็ได้ ที่แกนของตัวหมุนมีสวิตช์หนีศูนย์กลางติดอยู่ด้วย ทั้งนี้เพื่อตัดชุดเริ่มทำให้หมุนออกไปหลังจากที่มอเตอร์มีความเร็วประมาณ 75% ของ synchronous



รูปที่ 2.19 แสดงวงจรของและเฟสไดอะแกรมของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

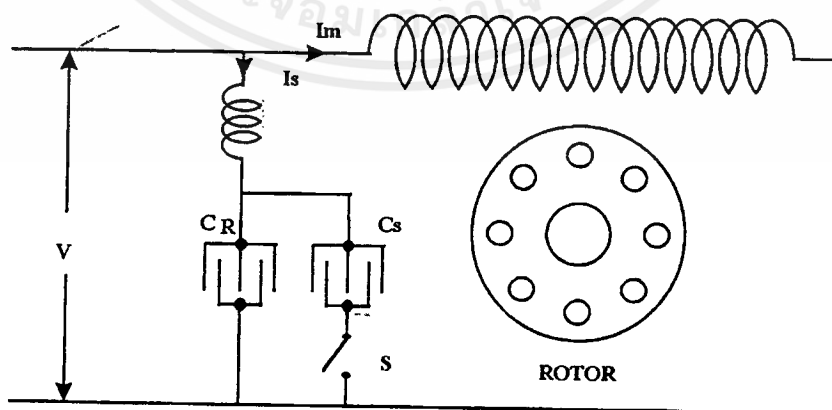
จากรูปที่ 2.19 จะพบว่ามีความต้านทานที่มีค่าสูงอีก 1 ตัวต่ออนุกรมเข้ากับขดลวดที่ทำให้หมุน (starting winding) ความต้านทานตัวนี้อาจไม่ต้องใช้ก็ได้ ถ้าหากว่าความต้านทานของขดลวดที่เริ่มทำให้หมุนมีค่าสูงพอ

จากคุณสมบัติของขดลวดทั้งสองที่กล่าวมาแล้ว จึงทำให้กระแส I_s และ I_m มีมุมแตกต่างกันเกือบ 90° ดังรูปที่ 2.19 กระแส I_s มาทีหลัง (lag) แรงเคลื่อนไฟ V ไปเป็นมุมเล็กน้อย ทั้งนี้เพราะค่าความต้านทานสูงกว่าค่า reactance มาก ฉะนั้นจึงทำให้กระแส I_s เกือบทับกับ (inphase) V แต่ว่ากระแส I_m อันเนื่องมาจากขดหลัก (Main winding) นั้น มีความต้านทานต่ำแต่ reactance สูง ฉะนั้นจึงทำให้ I_m มาทีหลัง V เกือบ 90° เมื่อเป็นเช่นนี้จึงทำให้เกิดสนามแม่เหล็กหมุนขึ้น เพื่อใช้ในการหมุนของมอเตอร์ส่วนสวิตช์หนีศูนย์กลาง S นี้ จะทำหน้าที่ตัดไฟของขดเริ่มหมุน (starting winding) ออกไป หลังจากที่มีมอเตอร์วิ่งได้เกือบใกล้ความเร็วพิกัดหรือประมาณ 75% ของความเร็วที่ synchronous

มอเตอร์แบบนี้มีแรงบิดเริ่มแรกประมาณ 1 ถึง 2 เท่าของแรงบิดที่พิกัด โดยใช้กระแสเริ่มแรกตั้งแต่ 5 ถึง 7 เท่าของกระแสที่พิกัดจึงทำให้เหมาะสมในการใช้งานกับเครื่องกลึงชนิดเล็ก ตัวแยกโดยอาศัยแรงหนีศูนย์กลาง (centrifugal separators) และตัวเป่า (blowers) เหล่านี้เป็นต้น

ทิศทางการหมุนของมอเตอร์เหล่านี้ก็กลับทิศทางการหมุนได้โดยการสลับขั้วต่อไฟของขดใดขดหนึ่งจากทั้ง 2 ขด (แต่ไม่ใช่สลับทั้ง 2 ขด) หลังจากนั้นจึงจ่ายไฟให้กับขั้วทั้งสอง ก็จะทำให้มอเตอร์หมุนกลับทิศ มอเตอร์แบบนี้บางทีก็เรียกว่า "มอเตอร์ที่เริ่มหมุนโดยใช้ความต้านทาน"

(ข) มอเตอร์ชนิดเริ่มหมุนด้วยตัวประจุและวิ่งด้วยการเหนี่ยวนำ (Capacitor Start Induction - Run Motor) มอเตอร์แบบนี้ก็มีโครงสร้างคล้ายกับแบบ (ก) ต่างกันที่ว่า มอเตอร์แบบนี้ทำให้กระแส I_s และ I_m มีเฟสแยกกันโดยใช้ตัวประจุหรือตัวควบแน่น (capacitor or condenser) ต่ออนุกรมกับขดเริ่มทำให้หมุน (starting winding) ดังรูปที่ 2.1



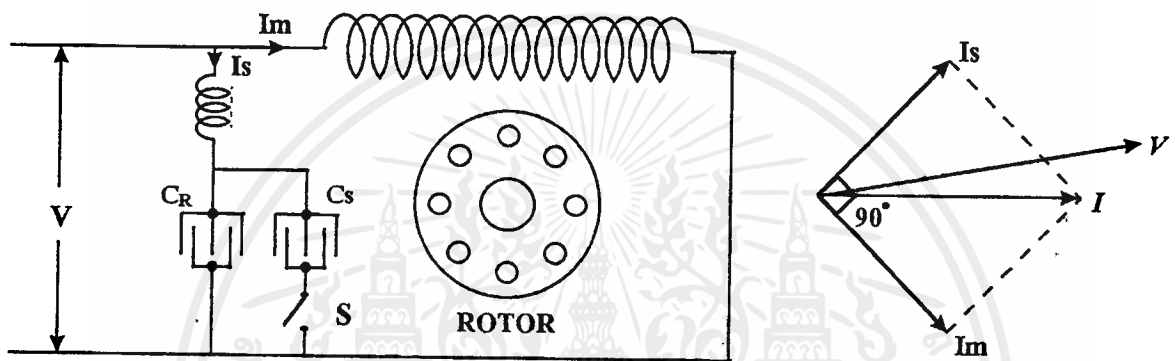
รูปที่ 2.20 แสดงวงจรของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.20 จะพบว่าตัวประจุไฟฟ้าที่ใช้เป็นชนิดสารละลาย (electrolytic) ตัวประจุนี้จะถูกตัดออกไปโดยสวิตช์หนีศูนย์กลางหลังจากที่มันมีความเร็วประมาณ 75% ของความเร็วที่ synchronous

กระแส I_m นี้มาที่หลัง V เป็นมุมใหญ่มุมหนึ่ง ในขณะที่เดียวกัน I_s มาก่อน V เป็นมุมหนึ่งเช่นกัน ดังนั้นจึงทำให้ได้กระแสลัพธ์ I ซึ่งเกือบจะทับกับ V ดังนั้นจึงทำให้แรงบิดเริ่มแรกสูง ซึ่งทำให้เหมาะสมที่จะนำไปขับตัวอัด (compressor) เครื่องผสมเหนียว (dough mixers) และอื่นๆ

(ค) มอเตอร์ชนิดที่ใช้ตัวประจุทั้งในขณะหมุนเริ่มแรกและในขณะวิ่งหรือทำงานตามปกติ (Capacitor - Start Capacitor - Run Motor) มอเตอร์แบบนี้มีตัวประจุไฟ 2 ตัวคือ C_s และ C_r



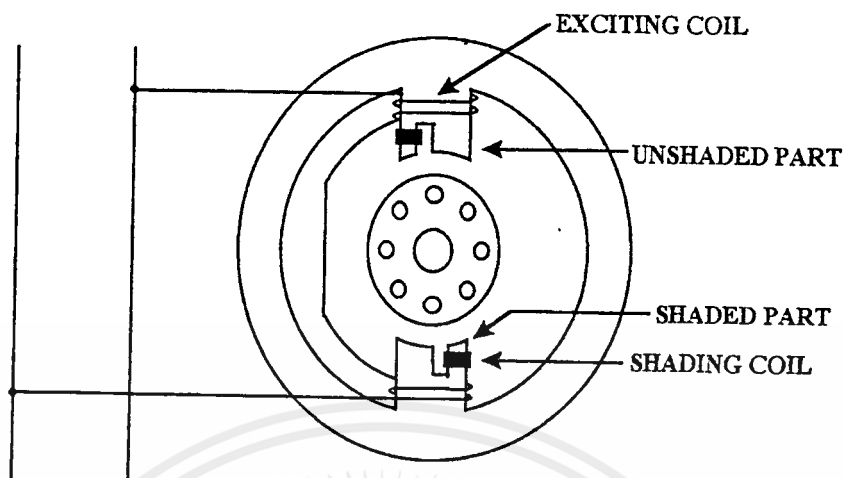
รูปที่ 2.21 แสดงวงจรและเฟสไดอะแกรมของมอเตอร์

จากรูปที่ 2.21 ตัว C_s ซึ่งใช้เป็นตัวทำให้มอเตอร์เริ่มหมุนนี้ มีความจุเกือบ 3 เท่าของ C_r ซึ่งใช้สำหรับวิ่งและเป็นตัวประจุแบบชนิด paper (กระดาษ) ตัว C_s นี้จะโดนตัดออกไปโดยอัตโนมัติ หลังจากที่มีมอเตอร์หมุนได้หลักสี่ถึงความเร็วที่กัก ในการตัด C_s นั้นนอกจากใช้สวิตช์หนีศูนย์กลางเป็นตัวตัดแล้ว ก็อาจใช้ current coil relay เป็นตัวตัดก็ได้ ส่วนใหญ่นิยมใช้ current coil relay เป็นตัวตัด โดยการใช้ C_r ที่เหมาะสมจึงทำให้กระแส I_s และ I_m แยกต่างกันอย่างเกือบ 90 ซึ่งทำให้เหมือนกับว่ามอเตอร์นั้นได้ไฟจากไฟ 2 เฟส

ข้อดีของการต่อตัวประจุนานเล็กไว้ในวงจรของมอเตอร์ก็คือ

1. ทำให้สามารถทน over - load ได้ดีขึ้น
2. มีตัวประกอบกำลัง (p.f.) สูงขึ้นกว่าเดิม
3. ทำให้วิ่งได้เสียบกว่า

(ง) Shaded - Pole Single - Phase Motors มอเตอร์แยกเฟสได้โดยการเหนี่ยวนำ มอเตอร์นี้เป็นแบบ บังขั้ว (salient pole) กล่าวคือ จะมีขั้วส่วนหนึ่งถูกบังไป ตัวที่ไปบังขั้วนี้ก็คือขดทองแดงหรือขดค้วนำที่ครบวงจรในตัวของมันเองอยู่แล้ว และนำไปบังขั้วดังกล่าวเสีย 1/3 ของขั้วทั้งหมด และส่วนที่บังขั้วของขั้วบนและขั้วล่างนั้นจะอยู่เอียงกัน ไปดังรูปที่ 2.22

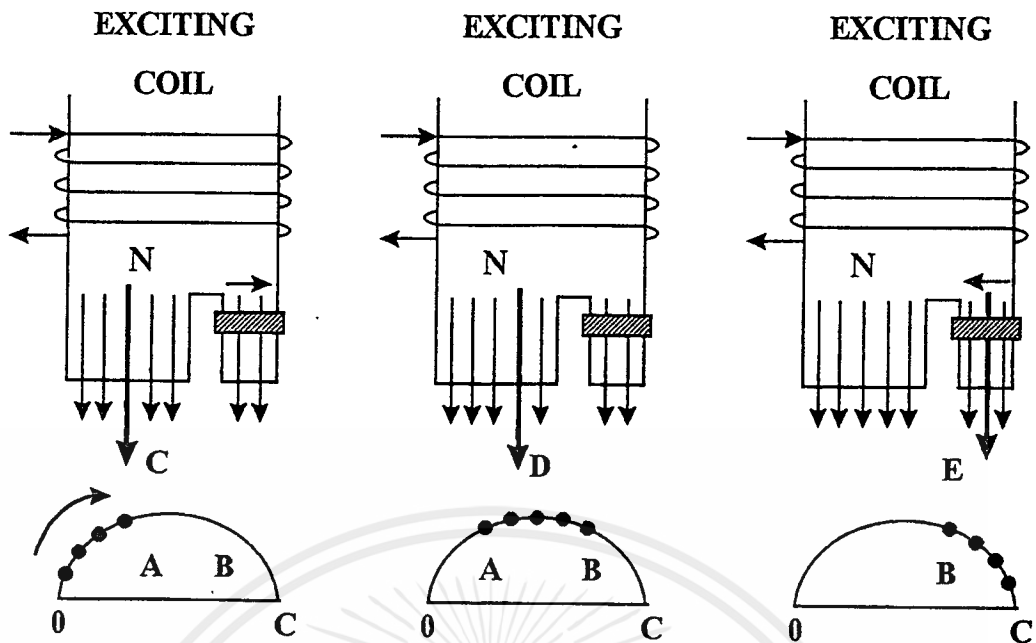


รูปที่ 2.22 แสดงการวางขั้วแม่เหล็กของมอเตอร์

ขั้วแม่เหล็กซึ่งเป็นเหล็กอบน้ำยาอัดแน่นติดกันนี้ ทำเป็นช่องประมาณ $1/3$ ของความยาวขั้วทางปลายด้านหนึ่ง นั่นคือให้หนึ่งขั้วแม่เหล็กโคนแบ่งออกเป็น 2 ส่วน ส่วนหนึ่งยาวประมาณ $1/3$ ของทั้งหมด อีกส่วนหนึ่งยาว $2/3$ ส่วนที่ยาว $1/3$ นี้ใส่ขดลวดทองแดงที่มีขนาดใหญ่และลัดวงจร ไปล้อมรอบที่ส่วนนี้ดังรูปที่ 4.1 ขดลวดทองแดงลัดวงจรที่ใส่ไปนี้เรียกว่า “shading coil” ส่วนนี้จึงเรียกว่าส่วนที่เป็น shaded (บังไว้) ส่วนอีกส่วนหนึ่งนั่นก็เป็น unshaded (ไม่บัง)

เมื่อมีกระแสไหลผ่านขดกระตุ้น (exciting coil) ซึ่งพันอยู่รอบๆ ขั้วแม่เหล็กทั้งขั้วนั้นย่อมทำให้แกนของแม่เหล็กเลื่อนจากส่วนที่ unshaded ไปยังส่วนที่ shaded จากการที่แกนของขั้วแม่เหล็กเคลื่อนที่ไปนี้จึงเป็นผลทำให้คล้ายกับว่ามีขั้วแม่เหล็กเคลื่อนที่ไปจริง ดังนั้นจึงทำให้ตัวหมุนเริ่มต้นหมุนไปในทิศทางของการเคลื่อนที่นี้ นั่นคือเคลื่อนที่จากส่วนที่ยังไม่ได้บังขั้ว (unshaded) ไปยังส่วนที่บังขั้วแล้ว (shaded)

การเคลื่อนที่หรือเลื่อนของแกนขั้วแม่เหล็กนี้เกิดจากหลักการที่ว่า ขด shading นั้นมี inductive สูงมาก ดังนั้นเมื่อมีไฟสลับวิ่งผ่านขดกระตุ้นไปในทางที่สูงขึ้นย่อมทำให้เกิดการเหนี่ยวนำในขด shading ซึ่งมีทิศทางสวนกันกับการเพิ่มจากขดกระตุ้นนั่นเองซึ่งยี่นไปตามหลักการของหม้อแปลง เมื่อเป็นเช่นนั้นย่อมทำให้ความหนาแน่นของเส้นแรงในส่วนที่เขียน shaded ลดลงไปเมื่อกระแสกระตุ้นเพิ่มขึ้น ในทำนองกลับกันเมื่อกระแสในขดกระตุ้นลดลงก็ย่อมทำให้ความหนาแน่นของสนามแม่เหล็กในส่วน shaded เพิ่มขึ้น (โดยที่สมมุติกระแสกระตุ้นนั้นเป็นลูกคลื่น sine)



รูปที่ 2.23 แสดงการเคลื่อนที่ของแกนขั้วแม่เหล็ก

จากรูปที่ 2.23(1) กระแสกระตุ้นเพิ่มขึ้นอย่างรวดเร็วตาม OA (ส่วนที่เป็นจุดๆ) ซึ่งทำให้เกิดการเหนี่ยวนำแรงเคลื่อนไฟในขด shading แรงแเคลื่อนนี้ทำให้เกิดกระแสจำนวนมากไหลในขด shading ที่มีค่าความต้านทานต่ำไปในทิศทางเดียวกันกับแรงเคลื่อนที่เกิดขึ้น กระแสนี้ทำให้เกิดสนามแม่เหล็กสวนกับสนามแม่เหล็กที่เกิดจากขดกระตุ้น ฉะนั้นจึงทำให้สนามแม่เหล็กส่วนใหญ่เกิดขึ้นที่ส่วน unshaded นั่นคือ แนวแกนของสนามแม่เหล็กก็อยู่ที่ส่วนกลางของส่วนนี้นั่นคือ ตามแนว NC

เมื่อพิจารณาต่อไปก็เห็นได้ว่ากระแสในขดกระตุ้นเกือบถึงค่าสูงสุดอยู่แล้วนั่นคือจาก A ไป B ดังรูปที่ 2.23(2) ที่จุดนี้การเปลี่ยนแปลงของเส้นแรงแม่เหล็กช้ามาก นั่นคือไม่มีการเหนี่ยวนำแรงเคลื่อนไฟที่ขด shading ฉะนั้นจึงไม่มีกระแสไหลในขด shading ย่อมทำให้เส้นแรงแม่เหล็กมีค่าสูงสุด (ไม่มีเส้นแรงส่วนจากขด shading) และกระจายไปทั่วขั้วที่ผิว เพราะฉะนั้นจึงทำให้แกนขั้วแม่เหล็กเคลื่อนมาอยู่ที่ศูนย์กลางของทั้งขั้วนั่นคือ ND

จากรูปที่ 2.23(3) พบว่ากระแสในขดกระตุ้นลดลงอย่างรวดเร็วจาก B ไป C ฉะนั้นจึงทำให้กระแสเหนี่ยวนำขึ้นในขด shading ซึ่งมีทิศทางตรงกันข้ามกับการลดของกระแสจึงทำให้มีเส้นแรงแม่เหล็กเพิ่มขึ้นในส่วนที่เป็น shading ฉะนั้นจึงทำให้แกนเส้นแรงแม่เหล็กเลื่อนไปอยู่ที่ศูนย์กลางของขด shading นั่นคือตามแนว NE

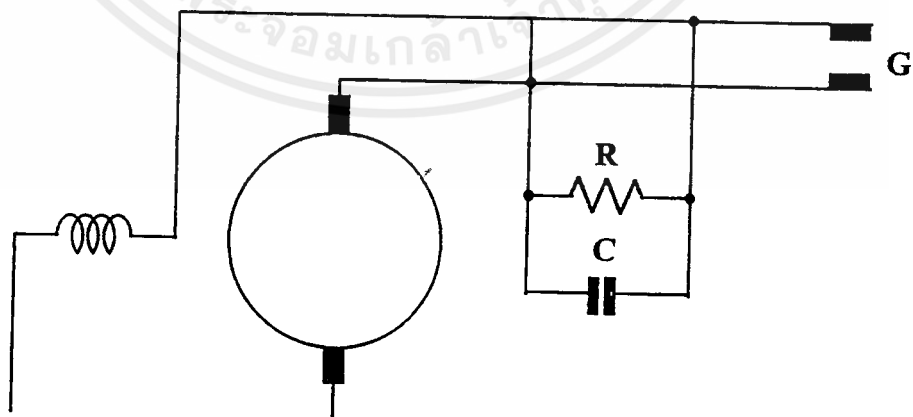
ดังนั้นจึงพบว่าในช่วงเวลาครึ่งคลื่นของกระแสไฟที่เป็นบวกของกระแสกระตุ้นนั้นทำให้ขั้วเหนือแม่เหล็กขั้วหนึ่งเลื่อนจากส่วนที่เป็น unshaded ไปยังส่วนที่เป็น shaded ของขั้วแม่เหล็กนั้นๆ สำหรับในช่วงครึ่งหลังที่เป็นลบนั้นก็ทำให้เกิดขั้วได้ขึ้นในทิศทางที่ทำให้สนามแม่เหล็กวิ่งจากส่วน unshaded ไปยังส่วนที่เป็น shaded เท่านั้น โดยที่มีเส้นแรงแม่เหล็กกลับทิศทางจากขั้ว N ผลที่เกิดขึ้นคล้ายกับมีขั้วแม่เหล็กถาวรอยู่ในอากาศโดยที่เคลื่อนที่จากซ้ายไปขวา ฉะนั้นจึงทำให้ทิศทางการเคลื่อนที่ของตัวหมุนวิ่งจากส่วนที่เป็น unshaded ไปยังส่วนที่เป็น shaded หรือว่าวิ่งจากส่วนที่ไม่มีวงแหวนล้อมรอบไปยังส่วนที่มีวงแหวนล้อมรอบที่ฝังอยู่ใน 1/3 ของขั้วแม่เหล็กนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่กล่าวมานั้นทำให้เห็นได้ว่า เมื่อต้องการทำให้มอเตอร์เฟสเดียวหมุนนั้นต้องหาวิธีการที่จะทำให้เกิดแรงบิดขึ้น โดยที่แรงบิดลัพธ์ที่ได้ไม่เท่ากับศูนย์โดยพยายามทำให้เกิดการเคลื่อนที่ของสนามแม่เหล็กลัพธ์ขึ้นในรูปแบบหรือกรรมวิธีต่างๆ

(ง) The Universal Motor มอเตอร์แบบนี้มีใช้กันนับไม่ถ้วนในบ้าน, ที่ทำงาน, โดร์ และ โรงงานอุตสาหกรรมทั้งหลายที่ต้องการใช้มอเตอร์ตัวเล็ก (ไม่ถึง 1 แรงม้า) เป็นตัวขับ มอเตอร์แบบนี้ใช้ได้ทั้งไฟสลับและไฟตรง ส่วนใหญ่เป็น series มอเตอร์ ทั้งนี้เพราะมีแรงบิดเริ่มแรกสูง โดยมีค่าประมาณ 3-4 เท่าของแรงบิดที่พิกัด มอเตอร์นี้ออกแบบมาเพื่อให้ใช้ได้กับไฟที่มีระดับแรงเคลื่อนอยู่ในช่วง 32 ถึง 250 โวลท์ และความถี่ 0 ถึง 60 คลื่น/วินาที โดยมีพิกัดต่ำกว่า $\frac{3}{4}$ แรงม้า ความเร็วเฉลี่ยสูงคืออยู่ในราว 7000 รอบ/นาที ที่ภาระปกติ ฉะนั้นจึงทำให้อัตราส่วนของแรงม้าต่อน้ำหนักมีค่าสูงกว่ามอเตอร์แบบธรรมดา มอเตอร์นี้ออกแบบมาให้มีสนามแม่เหล็กค้ำทั้งนี้เพื่อลดประกายไฟที่ commu และแกนเหล็กที่ตัวอยู่กับที่ก็ต้องเป็นแบบที่มีน้ำยาอบก่อน (laminated) ที่จะนำมาอัดทำเป็นแกนเหล็ก แปร่งถ่านที่ใช้ก็จะต้องมีความต้านทานสูงมาก เพื่อลดกระแสหมุนวนที่เกิดจากขดลวดที่โคจรลัดวงจรจากแปร่งถ่านนั้น ๆ บางทีก็ต้องใช้ช่อชดเชย (compensating) เพื่อลดประกายไฟนี้แต่ทั่ว ๆ ไป แล้วมอเตอร์ขนาดเล็กก็ไม่จำเป็นต้องใช้

ปกติ series มอเตอร์จะมีความเร็วสูงเมื่อไม่มีภาระ แต่ว่า univrsal มอเตอร์นี้ไม่เป็นเช่นนั้นเพราะว่ามีแรงเสียดทานและแรงลมมากพอที่จะทำให้ความเร็วตอนไม่มีภาระอยู่ในค่าที่ปลอดภัยโดยที่อยู่ในช่วง 15,000-20,000 รอบ/นาที ความเร็วปกติของมอเตอร์นี้ประมาณ 7,000 รอบ/นาที ที่ภาระตามปกติ ฉะนั้นจึงสูงไปสำหรับงานบางอย่าง แต่แก้ได้โดยใช้เพื่อทดแทนเพื่อลดความเร็วซึ่งมีอยู่ในตัวมอเตอร์ การเปลี่ยนความเร็วก็ทำได้โดยใช้ตัวควบคุม (governor) ให้อยู่ในช่วงที่ใช้งาน ได้พอสมควรสมผล รูป 4.6-1 ประกอบด้วยหน้าสัมผัส G หนึ่งคู่ผูกติดอยู่บนสปริงที่หมุนไปกับแกนของมอเตอร์ ในการปรับความเร็วที่ต้องการนั้นทำได้โดยทำให้หน้าสัมผัส G เบียดออก ส่วน R ก็จะเป็นตัวจำกัดกระแสไม่ให้สูงไปกว่าค่าที่ตั้งไว้ ส่วน C นั้นเป็นตัวจำกัดประกายไฟที่จะเกิดขึ้นที่หน้าสัมผัส G รูป 2.24 แสดงคุณลักษณะของ universal มอเตอร์ขนาด 1/16 แรงม้าซึ่งเป็นของ W.H Fromm, (Why Use Universal Motors) Elec.Mfg., July 1936.



รูป 2.24 ยูนิเวอร์ซอลมอเตอร์ใช้ตัวควบคุมความเร็วชนิดแรงหนีศูนย์กลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

3.1 หลักการออกแบบวงจร

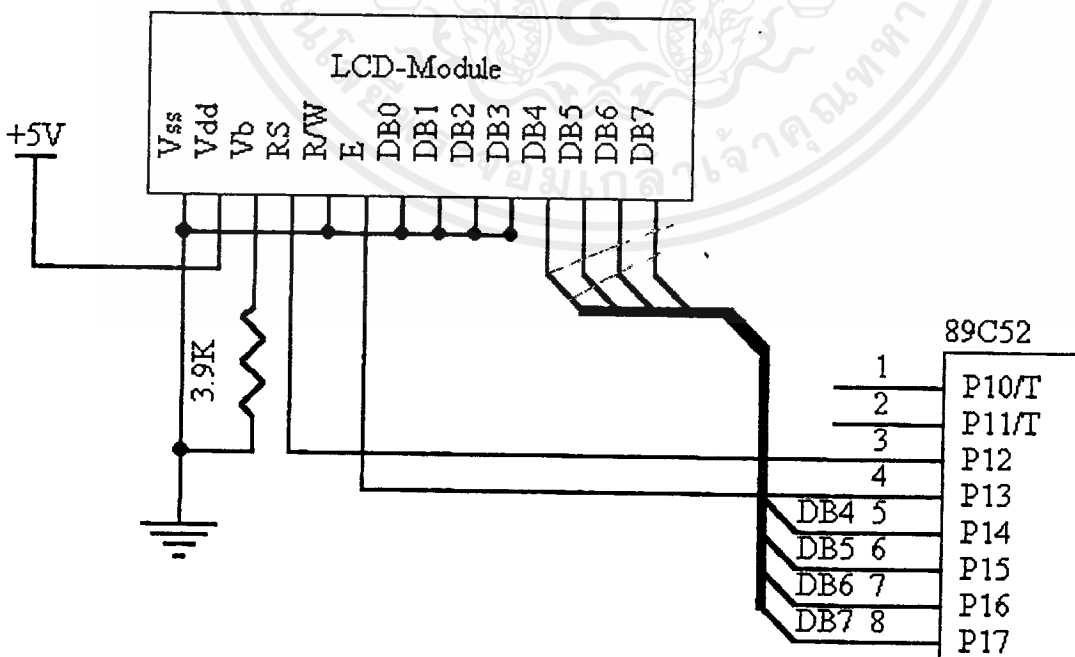
เนื่องจากระบบควบคุมอัตโนมัติมีส่วนการทำงานที่ถือว่าเป็นหัวใจหลักของระบบ คือหน่วยประมวลผลส่วนกลางซึ่งใช้ไมโครคอนโทรลเลอร์ในตระกูล MCS-51 มาเป็นตัวควบคุมระบบการทำงานทั้งหมด และทางผู้จัดทำก็ได้เลือกใช้ไมโครคอนโทรลเลอร์เบอร์ AT89C52 ซึ่งมีข้อได้เปรียบกว่าการใช้เบอร์ 80C32 ตรงที่ไม่ต้องใส่รอม (ROM) ภายนอก โดยสามารถทำการโปรแกรมรอมภายในตัวของไมโครคอนโทรลเลอร์เบอร์ AT89C52 ได้เลยทำให้ประหยัดอุปกรณ์ที่นำมาใช้กับระบบและอีกประการหนึ่งคือเมื่อทำการโปรแกรมเข้าไปในรอมภายในของชิพ (Chip) เบอร์นี้แล้วจะทำให้เราสามารถใช้อินพุท/เอาต์พุท พอร์ต (I/O Port) เพิ่มขึ้นมาอีก 2 พอร์ต คือ พอร์ต 0 และ พอร์ต 2 ซึ่งเดิมจะใช้ในการอ้างอิงแอดเดรสเพื่อที่จะเฟตช์ (Fetch) ออฟโค้ดจากรอมภายนอก ดังนั้นจะเห็นได้ว่าหากเราเลือกใช้ไมโครคอนโทรลเลอร์เบอร์นี้แล้วจะทำให้เราสามารถใช้อินพุท/เอาต์พุท พอร์ตทั้ง 4 พอร์ตของไมโครคอนโทรลเลอร์ตระกูลนี้ได้อย่างครบถ้วน ซึ่งก็เพียงพอกับระบบควบคุมที่ไม่ใหญ่เกินไปนัก

สำหรับระบบควบคุมประจําอัตโนมัตินี้ผู้จัดทำได้ทำการแยกส่วนต่างๆ ในการอินเตอร์เฟสกับไมโครคอนโทรลเลอร์และวงจรที่ใช้งานเกี่ยวกับระบบออกเป็นส่วนๆ เพื่อที่จะพิจารณาและทำความเข้าใจได้ง่าย และหลังจากนั้นจะนำส่วนต่างๆ ที่ได้แสดงไว้มารวมกับเป็นระบบควบคุมประจําอัตโนมัติ

3.2 การอินเตอร์เฟส (Interface)

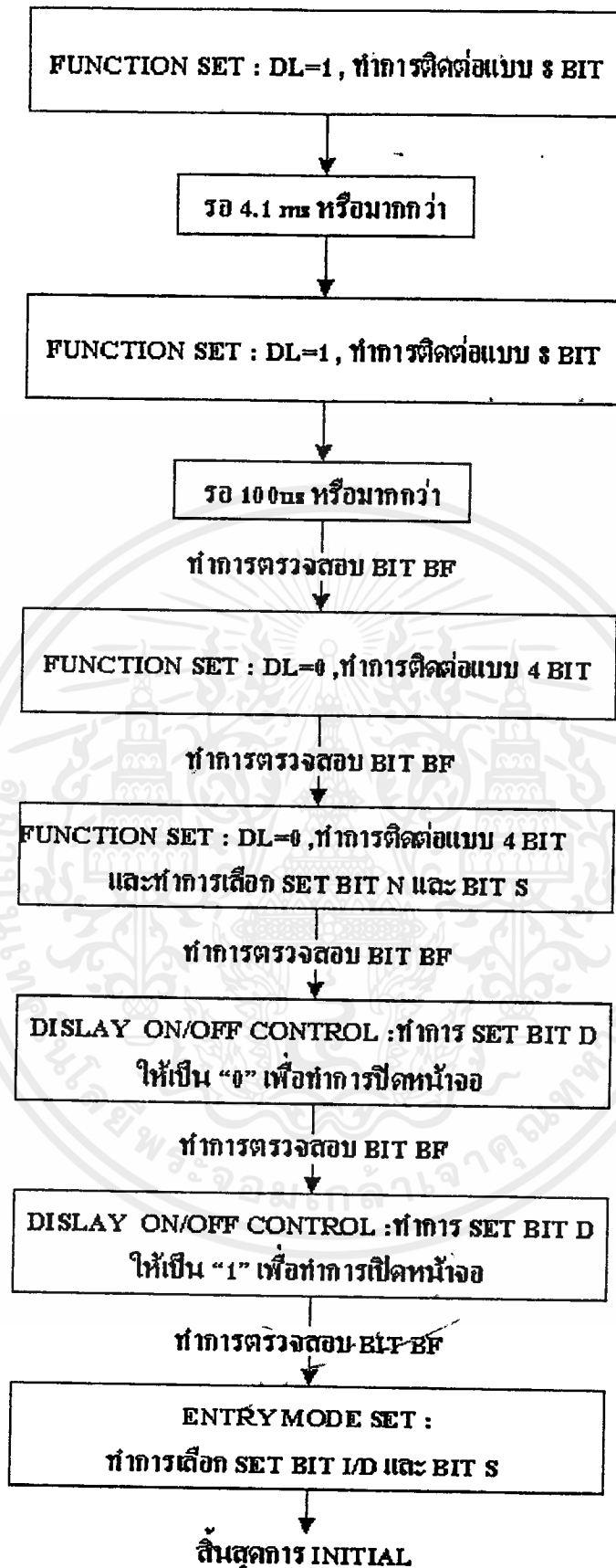
3.2.1 การอินเตอร์เฟสกับแอลซีดีโมดูล

สำหรับการอินเตอร์เฟสกับแอลซีดีโมดูลนั้นจะใช้สายสัญญาณเพียง 6 เส้นเท่านั้นคือ P1.2-P1.7 ซึ่งจะต่อกับขา RS ขา E และขา DB4-DB7 ของแอลซีดีโมดูลดังแสดงในรูปที่ 3.1



รูปที่ 3.1 แสดงการอินเตอร์เฟสกับแอลซีดีโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงไพล์วาร์ดของการตั้งค่าตั้งในการเซตค่าเริ่มต้นของแอลซีดีไมโคร

ในส่วนของขั้นตอนการ โปรแกรมให้แอลซีดีโมดูลแสดงผลนั้นในตอนเริ่มแรกเมื่อจ่ายไฟให้แอลซีดี โมดูลจะทำการรีเซต (Reset) ตัวมันเองโดยจะใช้เวลาประมาณ 10 ไมโครวินาทีและหลังจากนั้นเราจะต้องส่ง คำสั่งควบคุมให้มันทำงานตามไฟล์ชาร์ตคังแสดงในรูปที่ 3.2

หลังจากที่ได้เช็คสถานะการทำงานในช่วงเริ่มต้นของแอลซีดีโมดูลเรียบร้อยแล้วในขณะนี้ แอลซีดีโมดูลก็พร้อมที่จะแสดงผลในรูปของอักขระ โดยการเขียนรหัสตัวอักษรเข้าไปเก็บไว้ใน DD-RAM ของตัวแอลซีดี โมดูลก็จะปรากฏตัวอักษรขึ้นทันที ซึ่งโดยทั่วไปแล้วรหัสของตัวอักษรเหล่านี้ทางผู้ผลิตก็จะทำไว้ตรงกับ รหัสแอสกี (ASCII CODE) สำหรับตำแหน่งต่างๆของการแสดงผลนั้นเราสามารถที่จะกำหนดได้โดยการเซต ค่าของ DD-RAM ADDRESS ซึ่งค่าของแอดเดรสแสดงไว้ดังรูปที่ 3.3

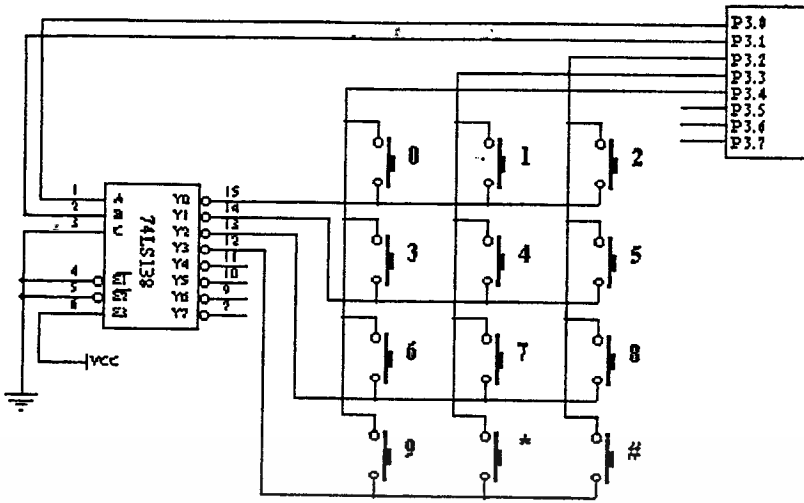
80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF
90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF

รูปที่ 3.3 แสดง DD-RAM ADDRESS สำหรับแอลซีดีโมดูล

3.2.2 การอินเตอร์เฟสกับคีย์บอร์ด

คีย์บอร์ดที่ใช้มีทั้งหมด 12 คีย์ด้วยกัน โดยแบ่งเป็นตัวเลข 0-9 เป็นจำนวน 10 ปุ่ม ปุ่มยกเลิกการเซตค่า (CANCEL) 1 ปุ่ม และปุ่มคกลง(ENTER) อีก 1 ปุ่มซึ่งลักษณะของการต่อวงจรเพื่ออินเตอร์เฟสนั้นจะอาศัยการ ถอดรหัสผ่าน ไอซีเบอร์ 74LS138 ดังนั้นขาที่ใช้เชื่อมต่อกับไมโครคอนโทรลเลอร์จะใช้เพียง 5 ขาเท่านั้น และ ในการเชื่อมต่อจะต่อเข้ากับขาอินเตอร์รัฟด้วย ดังนั้นในส่วนของ โปรแกรมการทำงานก็ไม่จำเป็นต้องมา คอยตรวจสอบการกดคีย์บอร์ดบ่อยๆ ซึ่งทำให้โปรแกรมสามารถทำงานได้อย่างอิสระ ต่อเมื่อมีการอินเตอร์รัฟเท่านั้นถึงจะมีการตรวจสอบคีย์บอร์ด

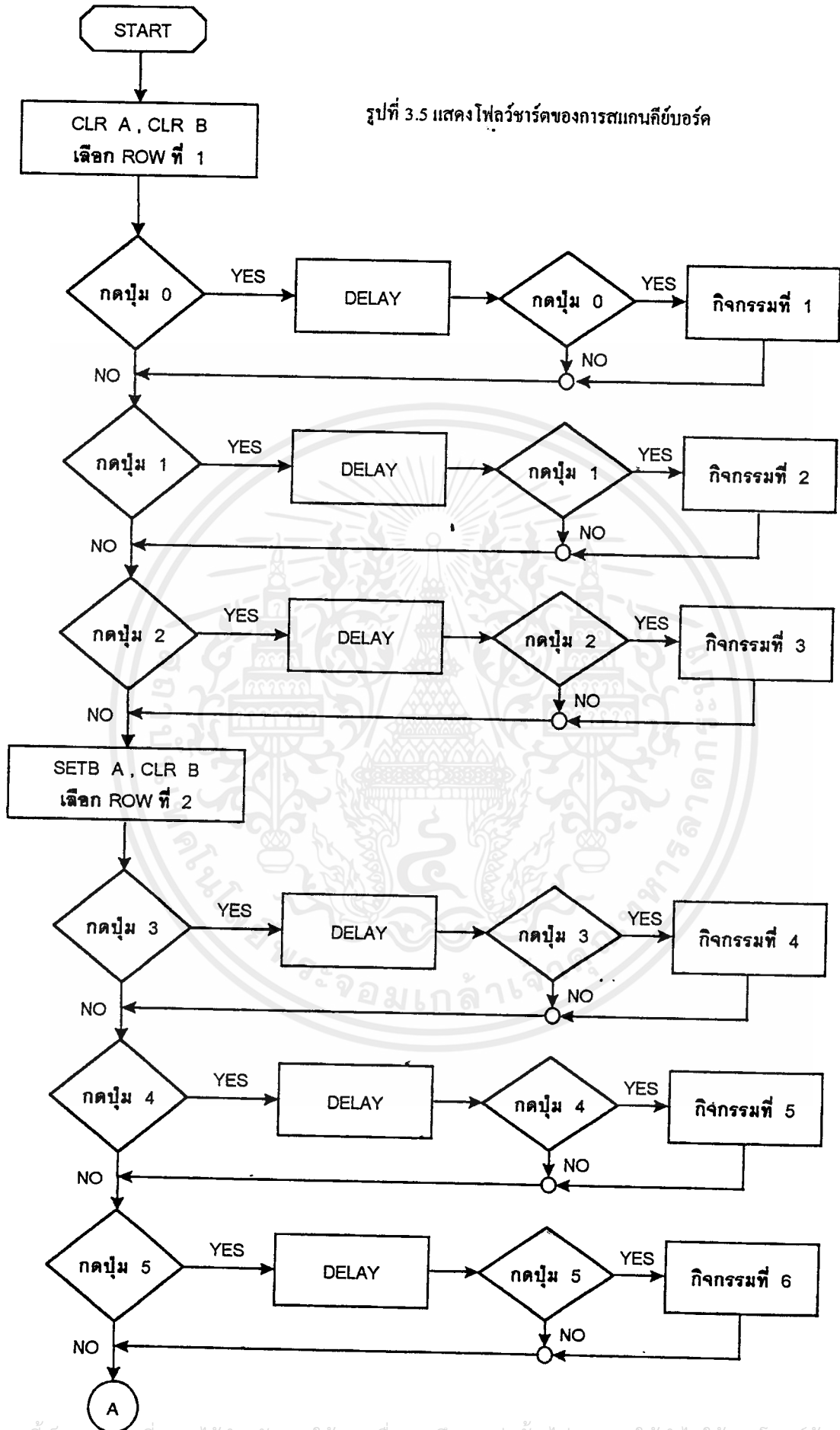
อินเตอร์รัฟที่ใช้กับการกดคีย์บอร์ดจะใช้อินเตอร์รัฟ 0 (INT0) ดังนั้นจากวงจรการเชื่อมต่อจะเห็นได้ว่าเมื่อเรากดปุ่ม ENTER เท่านั้นถึงจะมีการตรวจสอบคีย์บอร์ด และจะเป็นการเข้าสู่การเซตค่าต่างๆในการใช้งานของระบบ ลักษณะการอินเตอร์เฟสคีย์บอร์ดกับไมโครคอนโทรลเลอร์แสดงดังรูปที่ 3.4



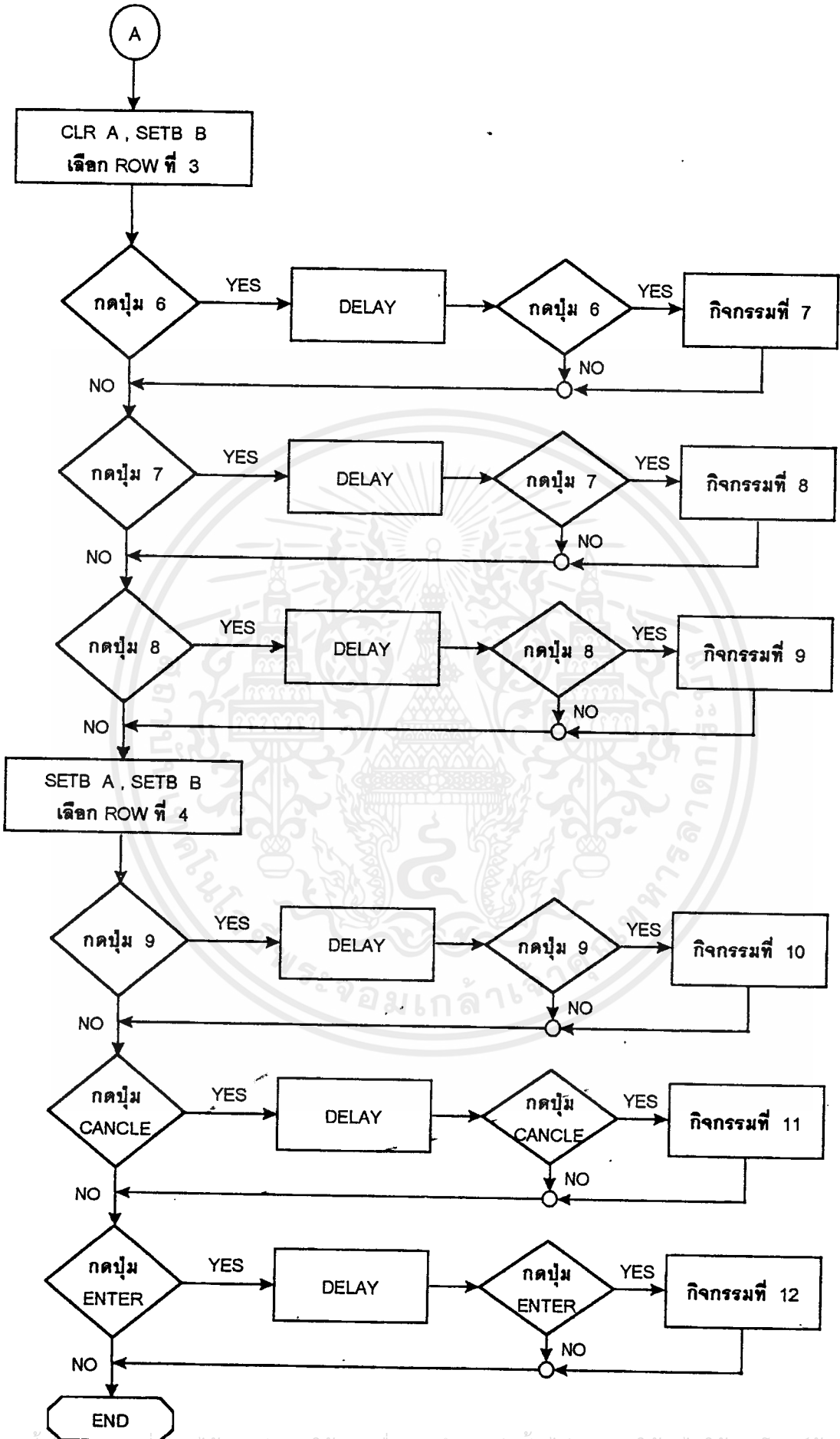
รูปที่ 3.4 แสดงการอินเทอร์เฟสกับคีย์บอร์ด
สำหรับการ โปรแกรมการสแกนคีย์บอร์ดสามารถโปรแกรมได้ตามโฟลว์ชาร์ตดังแสดงในรูปที่ 3.5



รูปที่ 3.5 แสดงโฟลว์ชาร์ตของการสแกนกีบ์บอร์ด



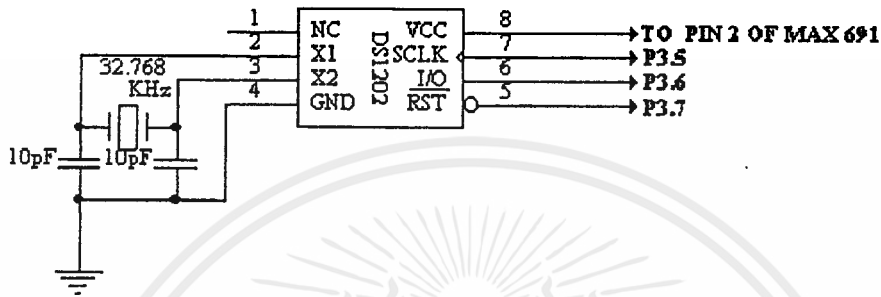
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



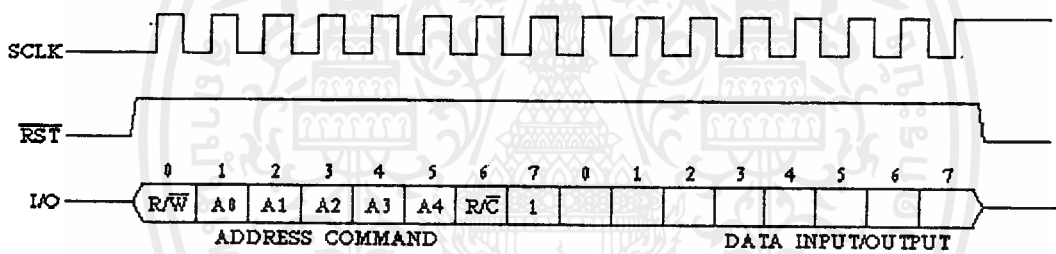
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 การอินเทอร์เฟสกับอุปกรณ์อาร์ทีซี (Real Time Clock)

อาร์ทีซีเป็นอุปกรณ์แสดงเวลา,วัน,เดือน,ปี และสามารถที่จะอ่านเขียนได้แบบอนุกรม การอินเทอร์เฟสจะใช้สายสัญญาณในการเชื่อมต่อเพียง 3 เส้นเท่านั้นดังที่ได้แสดงไว้ในรูปที่ 3.6 สำหรับการอ่านหรือเขียนข้อมูลกับอุปกรณ์อาร์ทีซีเราจะต้องส่งข้อมูลคำสั่ง (Command Byte) ไปยังอุปกรณ์ 8 บิต แล้วหลังจากนั้นก็เป็นการรับข้อมูลเข้ามาอีก 8 บิต ในกรณีของการอ่าน และหากเป็นกรณีของการเขียนข้อมูลลงไปก็จะเป็นการเขียนข้อมูลตามหลังข้อมูลคำสั่งอีก 8 บิต เช่นกัน ซึ่งลักษณะของสัญญาณที่ส่งไปจะเป็นสัญญาณแบบอนุกรมดังแสดงในรูปที่ 3.7



รูปที่ 3.6 แสดงการอินเทอร์เฟสกับอุปกรณ์อาร์ทีซี



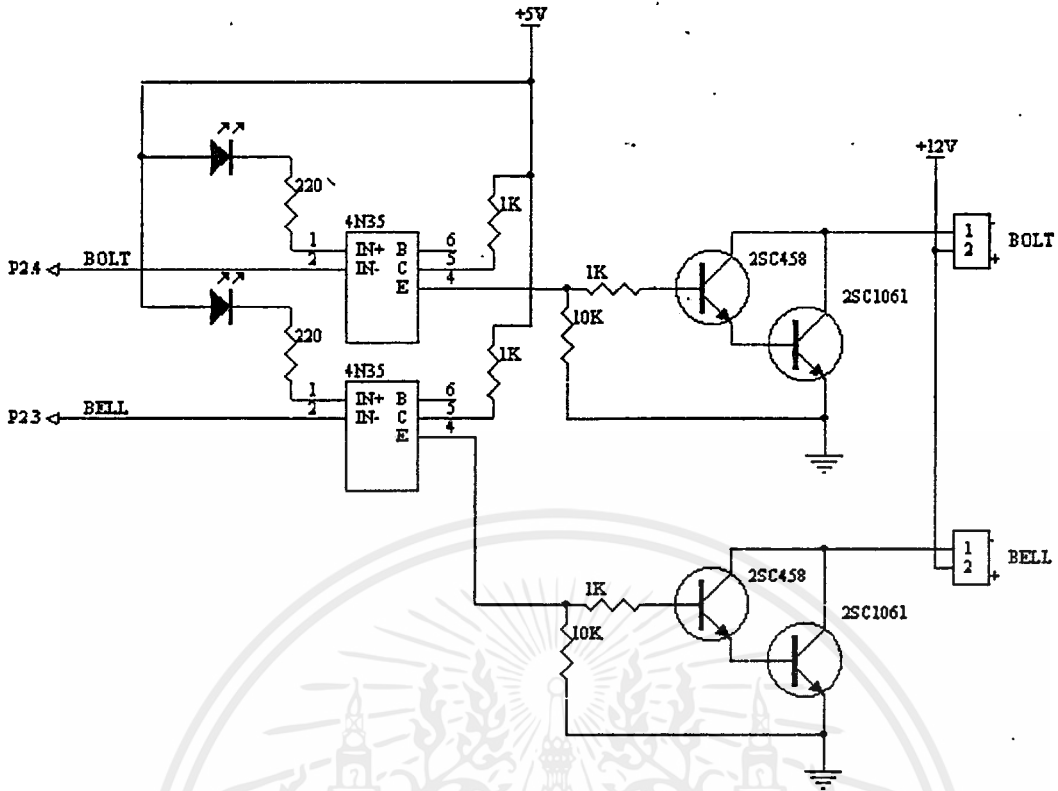
รูปที่ 3.7 รูปแบบของการเขียน/อ่านอุปกรณ์อาร์ทีซี

3.2.4 วงจรขับมอเตอร์

วงจรขับมอเตอร์ ดังแสดงดังรูปที่ 3.8 ซึ่งใช้ออปโตไดโอดแอก (Opto-Diac) เป็นตัวจุดชนวน (Trig) ให้ ไทรแอก (Triac) ทำงานทำให้มีกระแสไหลผ่านมอเตอร์ ชนิดของ ไทรแอกที่ใช้ขึ้นกับขนาดและชนิดของมอเตอร์ แต่อย่างน้อยควรทนแรงดันได้ไม่น้อยกว่า 3 เท่าของแรงดัน ไฟบ้าน อัตราการทนกระแสขึ้นกับกระแสสูงสุดของมอเตอร์หรือหาได้จากกำลังสูงสุดของมอเตอร์หารด้วยแรงดัน ไฟฟ้า แล้วคูณผลลัพธ์ด้วย 2 เช่น มอเตอร์ขนาด 500 W จะต้องการกระแส (500 W/220 V) * 2 \cong 5 A

สำหรับการควบคุมความเร็วเพื่อทำให้มอเตอร์หมุนช้าลง ก็ทำได้โดยการเปลี่ยนค่าความต้านทานที่ต่ออนุกรมอยู่กับ ตัวเก็บประจุตรงขาเกตของ ไทรแอก ซึ่งจากวงจรใช้ออปโต ไดโอด แอก ต่อขนานกับตัวความต้านทานซึ่งเมื่อออปโต ไดโอดทำงานความต้านทานรวมจะต่ำทำให้มีมุมนำกระแสเป็นศูนย์มีกระแสไหลผ่าน ไทรแอกเต็มทีส่งผลให้มอเตอร์หมุนเต็มกำลัง และ ในทางกลับกันเมื่อออปโต ไดโอดไม่ทำงานความต้านทานรวมจะมีค่าสูงและทำให้ ไทรแอกมีมุมนำกระแสเปลี่ยนไปซึ่งจากวงจรนั้นมีมุมนำกระแสประมาณ 90 องศา และในการเปลี่ยนทิศทางการหมุนของมอเตอร์ทำได้โดยการ ใช้รีเลย์ (Relay) เป็นตัวตัดต่อเฟสของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แสดงวงจรขับเคลื่อนและกระดิ่งไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจากวงจรได้ทำการแบ่งพอร์ตที่ใช้งานคือ พอร์ต 0 จะใช้เป็นพอร์ตอินพุตสำหรับรับอินพุตจากภายนอกและ พอร์ต 2 จะใช้เป็นเอาต์พุตที่ส่งสัญญาณไปควบคุมอุปกรณ์ภายนอกต่างๆ และทั้งสองพอร์ตมีการต่อออปโตไอโซเลเตอร์ไว้สำหรับการคัปเปิลระหว่างภาคควบคุมกับภาคขับเคลื่อนและเอาต์พุตต่างๆซึ่งแต่ละพอร์ตมีรายละเอียดดังต่อไปนี้

พอร์ตที่ใช้ในการรับอินพุตจากภายนอกระบบควบคุมคือ

P0.0	ใช้กับ	Magnetic switch
P0.1	ใช้กับ	Manual switch
P0.2	ใช้กับ	Remote control
P0.3	ใช้กับ	Infrared
P0.4	ใช้กับ	Mode selector
P0.5	ใช้กับ	Detect 12 volt fail
P0.6	ใช้กับ	Detect power fail
P0.7	ใช้กับ	Detect 3 volt fail

พอร์ตที่ใช้ในการส่งสัญญาณไปควบคุมอุปกรณ์ภายนอก

P2.0	ใช้กับ	Motor run
P2.1	ใช้กับ	Slow Motor
P2.2	ใช้กับ	Phase Motor
P2.3	ใช้กับ	Soloid bolt
P2.4	ใช้กับ	Electric bell
P2.5	ใช้กับ	LED open
P2.6	ใช้กับ	LED close
P2.7	ใช้กับ	Speaker

ในการเลือกโหมดการทำงานนั้นสามารถเลือกได้ 2 รูปแบบ คือ ทอกเกิล โหมด (Toggle Mode) และ ออโต้โหมด (Auto Mode)

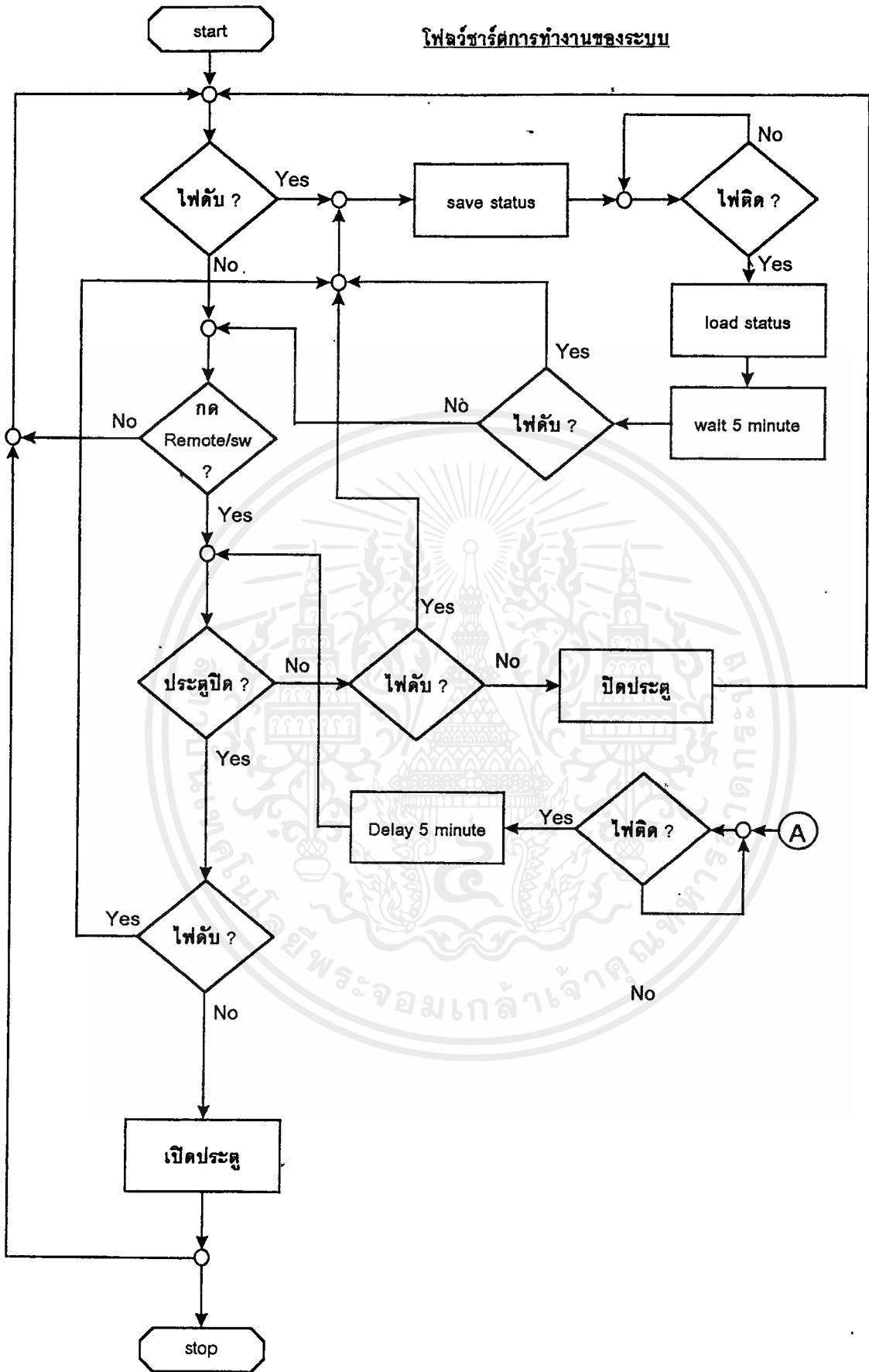
กรณีของออโต้โหมด หมายถึง เมื่อมีการกดรีโมทหรือสวิตช์กดปุ่มซึ่งจากเดิมประตูปิดอยู่ก็จะเปิดออก และไม่จำเป็นที่จะต้องกดรีโมทหรือสวิตช์กดปุ่มซ้ำประตูก็จะปิดเองโดยอัตโนมัติตามเวลาที่ได้ตั้งเอาไว้

กรณีของทอกเกิลโหมด หมายถึง เมื่อมีการกดรีโมทหรือสวิตช์กดปุ่มซึ่งจากเดิมประตูปิดอยู่ก็จะเปิดออก และประตูจะไม่ปิดจนกว่าจะมีการกดรีโมทหรือสวิตช์กดปุ่มซ้ำอีกครั้งหนึ่ง

3.3.1 โพลีชาร์ตการทำงานของระบบ

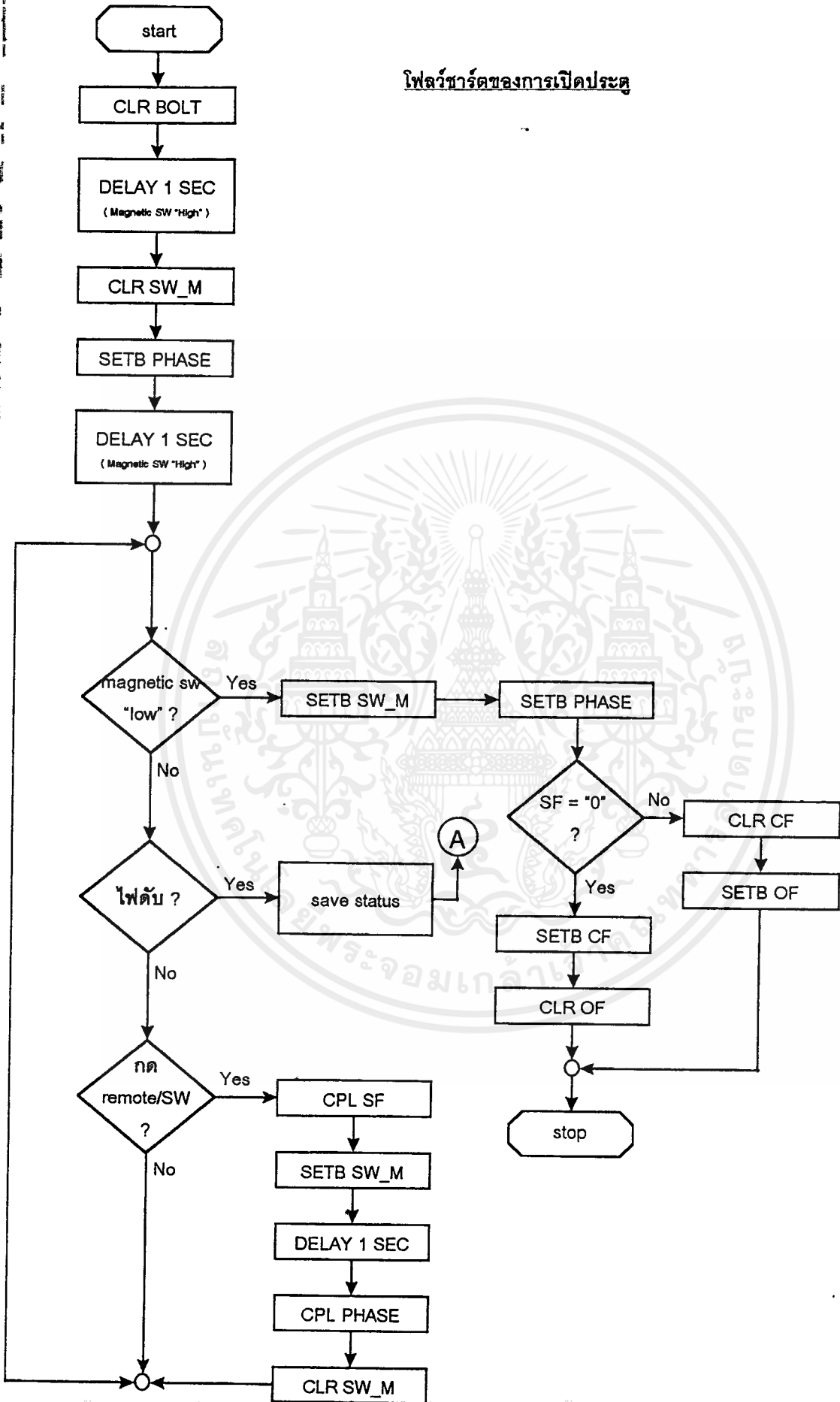
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลว์ชาร์ตการทำงานของระบบ



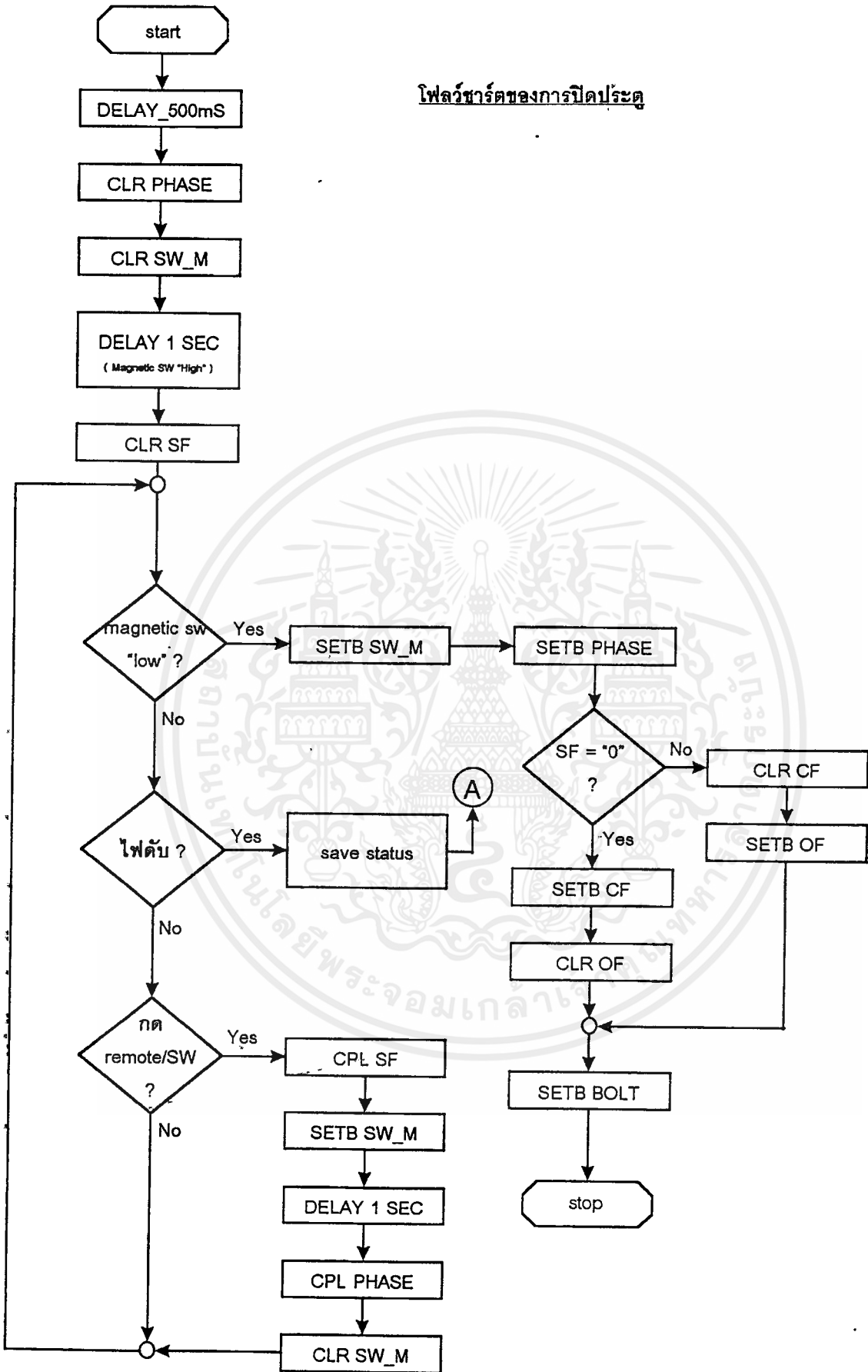
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลว์ชาร์ตของการเปิดประตู



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลว์ชาร์ตของการเปิดประตู



การทำงานของระบบแสดงโดยโฟลว์ชาร์ตดังรูปที่ 3.11

3.3.2 ฟังก์ชันการทำงานของระบบควบคุมประจุอัตโนมัติ

ในระบบได้กำหนดฟังก์ชันการเซตค่าการทำงานไว้ดังนี้

1. การตั้งเวลา
2. การตั้ง วัน,เดือน,ปี
3. การตั้งระยะเวลาของการปิดประจุในกรณีทำงานในโหมดคอดได้
4. การตั้งชื่อของเครื่อง
5. การตั้งระยะเวลาการหยุดมอเตอร์อัตโนมัติในกรณีที่เกิดความผิดพลาด

3.3.3 การใช้งานระบบควบคุมประจุอัตโนมัติ

ระบบควบคุมประจุอัตโนมัติเป็นระบบที่สามารถนำไปประยุกต์ใช้กับประจุนิตต่างๆได้เพราะมีฟังก์ชันไว้ให้เซตรูปแบบของการทำงานตามความเหมาะสมกับลักษณะของประจุ เช่นว่าในกรณีของประจุที่มีความกว้างต่างๆกัน เราก็สามารถที่จะตั้งค่าการหน่วงเวลาของการหยุดของมอเตอร์ได้ตามความเหมาะสม

สำหรับการเซตค่าของการทำงานในระบบทำได้โดยการกดปุ่ม ENTER หนึ่งครั้ง เมื่อกดแล้วจอแอลซีดีจะปรากฏหมายเลขของฟังก์ชันต่างๆตามที่ได้กล่าวไว้แล้ว และเมื่อต้องการเซตฟังก์ชันใดก็ได้โดยการกดตามหมายเลขที่ปรากฏอยู่ในขณะนั้น ซึ่งหมายเลขของฟังก์ชันมีดังต่อไปนี้

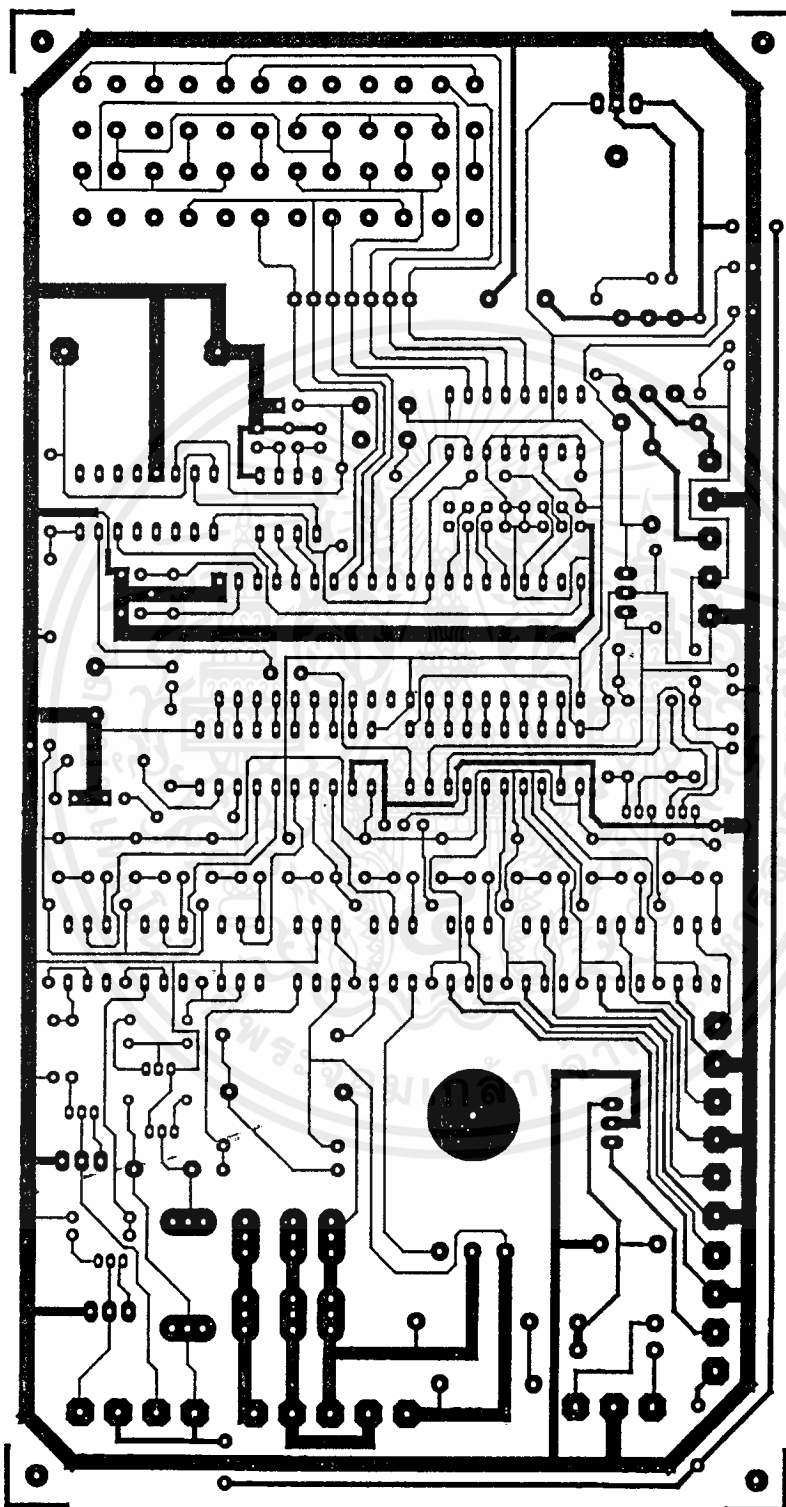
- | | |
|---------|--|
| หมายเลข | 1. Time Set - เป็นการตั้งเวลา |
| หมายเลข | 2. Date Set - เป็นการตั้งวัน,เดือน,ปี |
| หมายเลข | 3. Delay time - เป็นการตั้งเวลาในการหน่วงการปิดประจุ(กรณีโหมดคอดได้) |
| หมายเลข | 4. Set Label - เป็นการตั้งชื่อของเครื่องบนจอแอลซีดี |
| หมายเลข | 5. Motor Stop- เป็นการตั้งระยะเวลาในการรันของมอเตอร์ |

และในสภาวะปรกติจอแอลซีดีจะแสดงเวลา,วัน,เดือน,ปี ในปัจจุบันพร้อมทั้งมีสภาวะของประจุที่กำลังปิดหรือเปิดอยู่

3.4 การออกแบบแผ่นลายวงจร (Print circuit board)

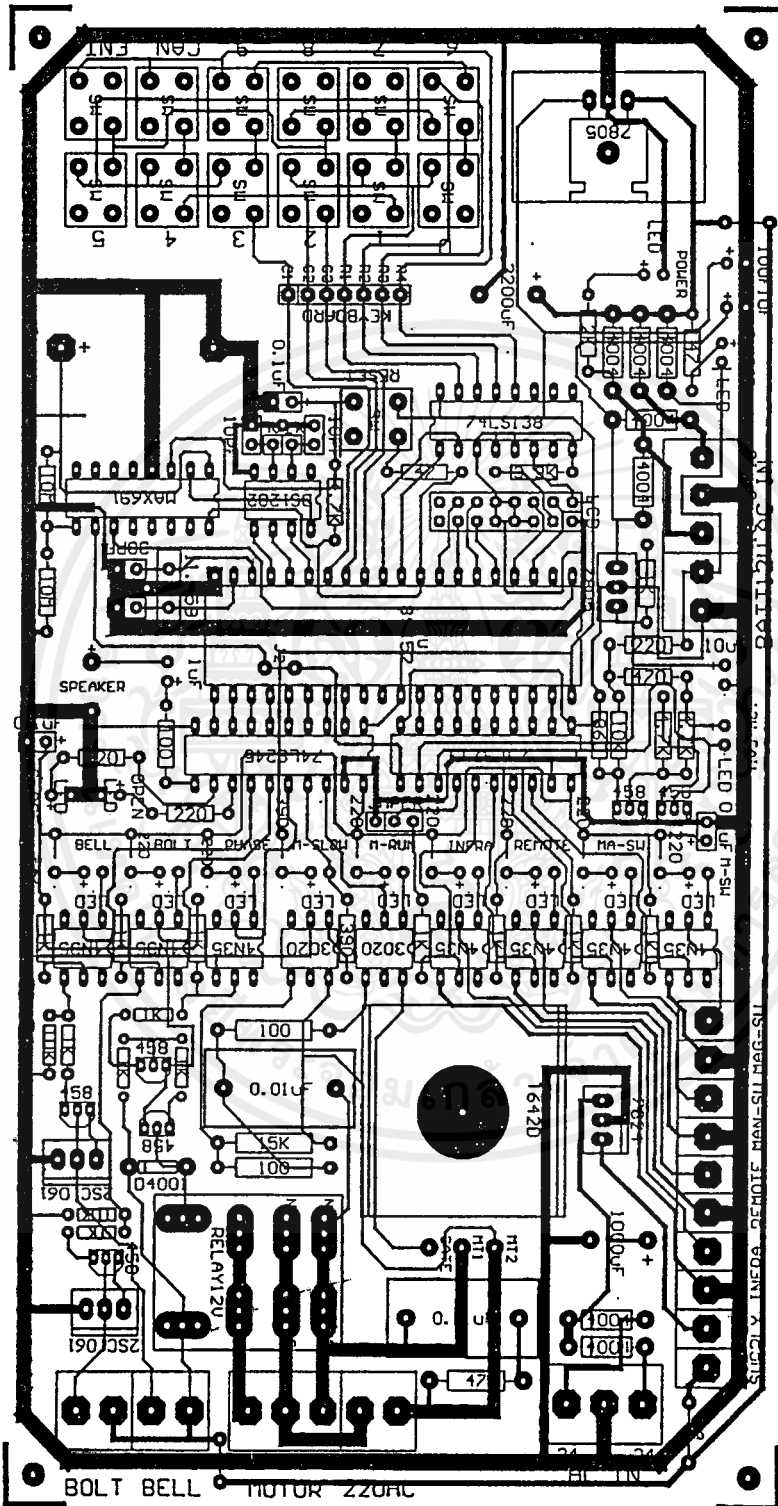
ในส่วนของการออกแบบแผ่นลายวงจรพิมพ์นั้นออกแบบด้วยการใช้โปรแกรมสำเร็จรูป Protel for windows ซึ่งเป็นขั้นตอนของการจัดวางอุปกรณ์ลงบนแผ่นลายทองแดงและการหาเส้นทางในการเดินลายวงจรอย่างเหมาะสมและประหยัดทรัพยากร ซึ่งจากขั้นตอนการออกแบบทำให้ได้แผ่นลายวงจรดังรูปที่ 3.12 และ

3.13



รูปที่ 3.12 แสดงลายวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 แสดงลายวงจรพร้อมทั้งการจัดวางคอมโพเนนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

ในส่วนของ การทดลอง ได้ทำการทดลอง ในด้านการเขียน โปรแกรมควบคุม ไปพร้อมๆกับการทดลอง ทางด้านฮาร์ดแวร์ (hardware) ซึ่งก็คือวงจรที่ได้แสดงไว้แล้วในบทที่ 3

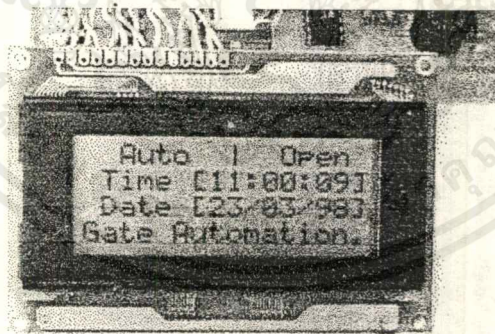
4.1 วิธีการทดลอง

4.1.1 ทดลองต่อฮาร์ดแวร์ซึ่งเป็นการต่อแบบจำลองลักษณะของสัญญาณที่ใช้ในการควบคุมอุปกรณ์ที่เกี่ยวข้องในระบบเช่น มอเตอร์, กลอนไฟฟ้า, กระดิ่งไฟฟ้า, สวิตช์แม่เหล็ก และอุปกรณ์อื่นๆที่เกี่ยวข้อง จากนั้นทำการโปรแกรมและปรับปรุงจนเป็น โปรแกรมที่ถูกต้องตามเงื่อนไขที่ต้องการ

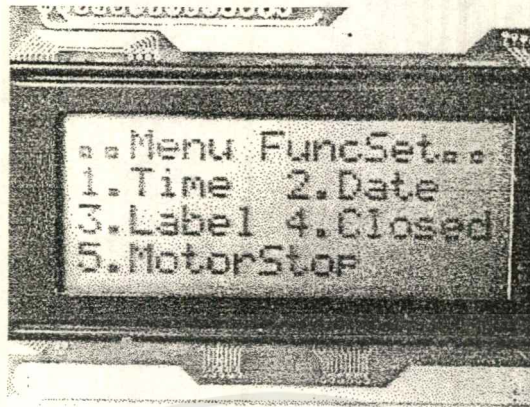
4.1.2 เชื่อมต่ออุปกรณ์ทางฮาร์ดแวร์ทั้งหมดของระบบลงบนแผ่นลายวงจรที่ได้ออกแบบไว้, ตรวจสอบความถูกต้องของแผ่นวงจรจากนั้นจึงป้อนไฟเข้าระบบ, ทดลองรัน โปรแกรม และแก้ไขข้อผิดพลาดทั้งในเรื่องของอุปกรณ์ทางฮาร์ดแวร์ และซอฟต์แวร์จนได้ผลที่พึงพอใจ

4.2 ผลการทดลอง

จากการทดลองและแก้ไขโปรแกรมก็ปรากฏว่าสามารถที่จะ โปรแกรมให้หลอดแอลอีดีติดสว่างตาม ไทม์มิ่งโคะแกรมได้และก็ได้ทดสอบฟังก์ชันการใช้งานต่างๆของระบบตามที่ได้กำหนดไว้ซึ่งใช้แอลซีดีในการแสดงผลและได้ผลการทดลองดังรูปต่อไปนี้



รูปที่ 4.1 แอลซีดีแสดงสถานะปกติ



รูปที่ 4.2 แอลซีดีขณะที่แสดงเมนูของการเซตฟังก์ชันใช้งาน

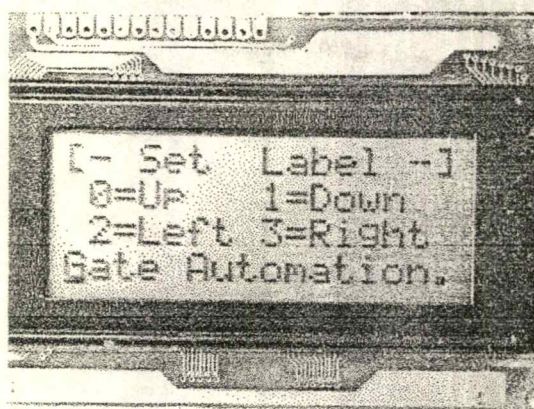


รูปที่ 4.3 แอลซีดีแสดงการเซตฟังก์ชันเวลา

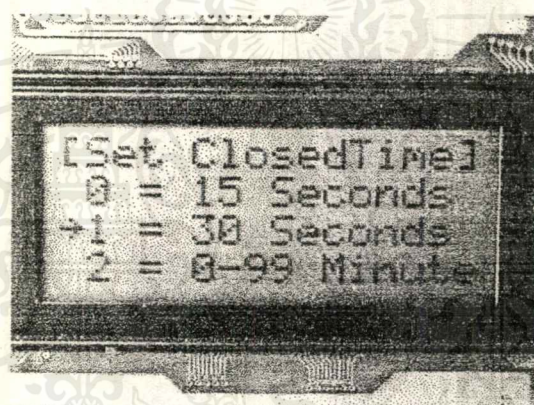


รูปที่ 4.4 แอลซีดีแสดงการเซต วัน,เดือน,ปี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แอลซีดีแสดงการเซตการตั้งชื่อลาเบล



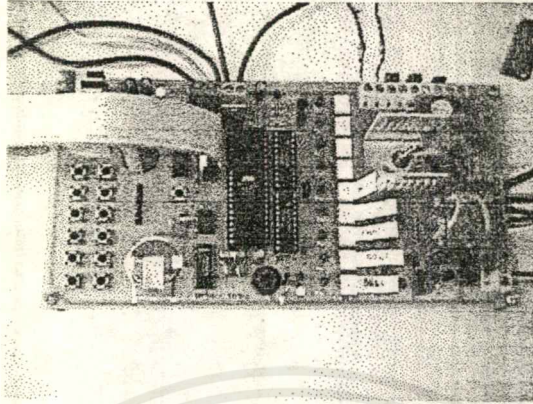
รูปที่ 4.6 แอลซีดีแสดงการเซตเวลาการปิดของประตู



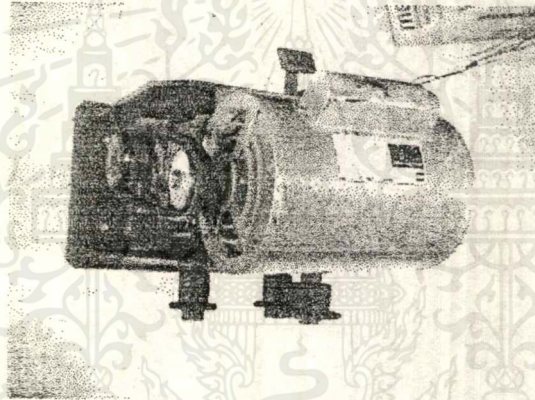
รูปที่ 4.7 แอลซีดีแสดงการเซตเวลาในการหยุดรันของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

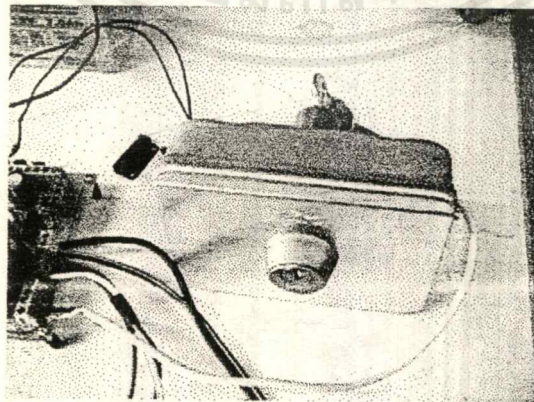
สำหรับวงจรที่ใช้งานและอุปกรณ์ประกอบการใช้งานของระบบแสดงคังรูปต่อไปนี้



รูปที่ 4.8 แสดงบอร์ดควบคุมของระบบควบคุมควบคุมประตูดิโนมัต

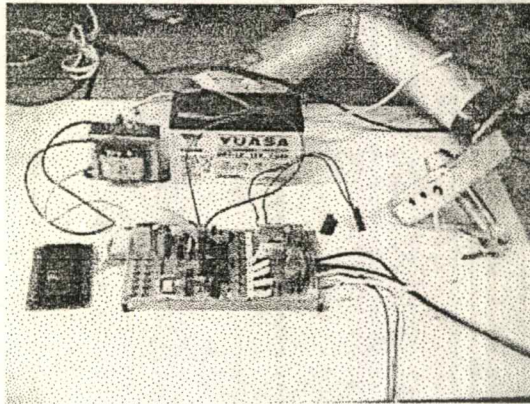


รูปที่ 4.9 แสดงมอเตอร์กระแสสลับที่จับประตู

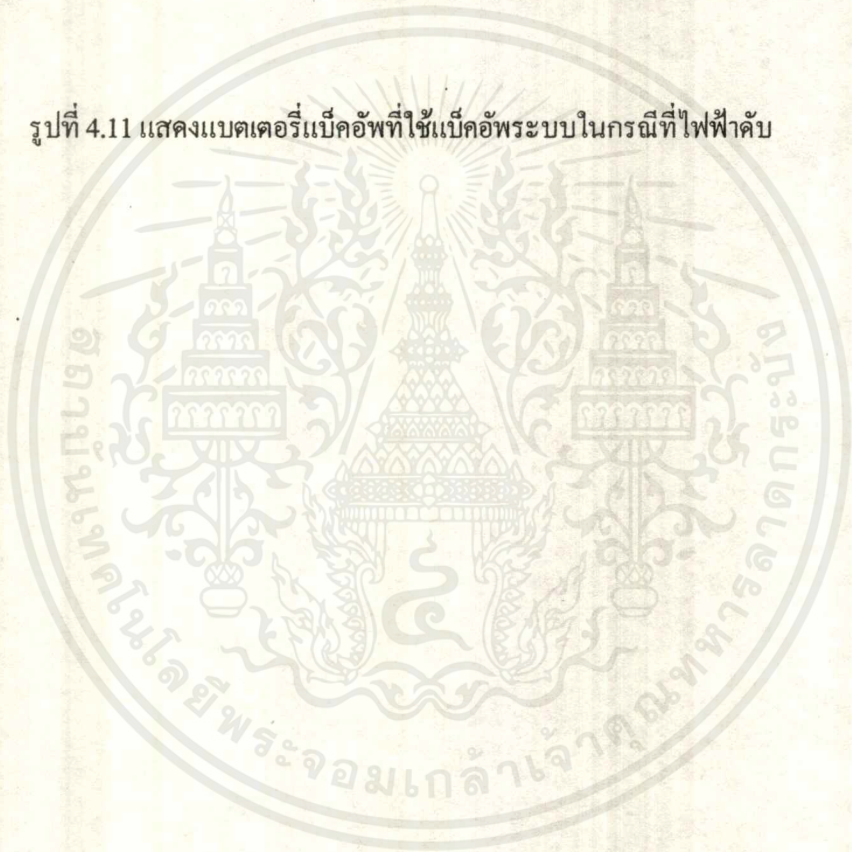


รูปที่ 4.10 แสดงกลอนไฟฟ้าที่ใช้ถือประตู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แสดงเบตเตอร์เบ็คอัพที่ใช้เบ็คอัพระบบในกรณีไฟฟ้าดับ



บทที่ 5

บทวิจารณ์และบทสรุป

จากการทดลองที่ผ่านมาผู้ปฏิบัติได้ประสบกับอุปสรรคและข้อบกพร่องที่เกิดขึ้นในการทดลองซึ่งเป็นสิ่งจำเป็นที่จะต้องทำการแก้ไขให้ดียิ่งขึ้น

5.1 ปัญหาที่เกิดขึ้นกับการทดลอง

ปัญหาส่วนใหญ่ที่เกิดขึ้นจะเกิดจากการอินเตอร์เฟซเชื่อมต่ออุปกรณ์ซึ่งหากทำการเชื่อมต่ออุปกรณ์ที่ไม่ถูกต้องหรือสายที่ใช้ในการเชื่อมต่อหลุดไปจะทำให้ลำบากในการค้นหาจุดบกพร่อง

ในส่วนของการขั้วมอเตอร์นั้นเมื่อมอเตอร์รันก็จะส่งผลให้ซีพียูเกิดการทำงานผิดพลาดหรือซีพียูถูกรีเซต และในกรณีที่ควบคุมมอเตอร์ให้หมุนช้าลงก็ยังมีปัญหาอยู่บ้างคือ มอเตอร์ไม่ได้ค่ออยู่กับโหลดหรือประตูที่ใช้งานจริงซึ่งหากมีการนำไปติดตั้งใช้งานจริงแล้วอาจจะต้องมีการปรับค่าของอุปกรณ์ในภาคขั้วมอเตอร์นี้อีกครั้งหนึ่ง และนอกจากนี้ในส่วนของเพาเวอร์ซัพพลาย, เพาเวอร์เบ็คอัพ ตลอดจนวงจรตรวจเช็คแบตเตอรี่เบ็คอัพ ก็มีปัญหาในส่วนของการทำงานไปเชื่อมต่อใช้งานจริงในวงจรเนื่องจากในขั้นตอนการทดลองนั้นได้แยกส่วนในการทดลอง

5.2 แนวทางในการแก้ปัญหา

การลดปัญหาที่เกิดขึ้นดังกล่าวมาแล้วนั้น ทำได้โดยการทดลองและทดสอบแต่ละส่วนของระบบจนแน่ใจว่าแต่ละส่วนสามารถทำงานได้อย่างถูกต้องหรือมีข้อผิดพลาดที่น้อยที่สุด จากนั้นในขั้นตอนการเชื่อมต่อส่วนต่างๆของระบบเข้าด้วยกันอย่างระมัดระวัง และทำการทดสอบวงจรทั้งหมดอีกครั้งจากนั้นทำการปรับเปลี่ยนค่าของอุปกรณ์ต่างๆ ที่เชื่อมต่อกันให้เหมาะสม และปัญหาที่ซีพียูกรีเซตเมื่อทำการรันมอเตอร์แก้ไขโดยการใส่ตัวเก็บประจุบายพาสตรงแหล่งจ่ายไฟตรงที่นำมาเลี้ยงซีพียู สำหรับส่วนของเพาเวอร์เบ็คอัพและวงจรตรวจสอบสภาพแบตเตอรี่ ทำได้โดยการปรับเปลี่ยนค่าความต้านทานในวงจร โดยการคำนวณในเรื่องของกระแสและแรงดันที่ใช้ให้เหมาะสม หรืออาจใช้ตัวความต้านทานที่สามารถปรับค่าได้ใส่แทนอุปกรณ์ตัวนั้นๆ

ในการพัฒนาเพื่อให้ได้ระบบที่สมบูรณ์แบบนั้น เป็นสิ่งจำเป็นอย่างขึ่งที่จะต้องนำระบบควบคุมประจุทั้งหมดไปทำการติดตั้ง และทดลองใช้งานในสภาพการณ์จริงๆ เพื่อประโยชน์ในการพัฒนาระบบในส่วนต่างๆ ซึ่งอาจจะเกิดปัญหาขึ้นได้ในภายหลัง ตลอดจนการปรับปรุงเปลี่ยนแปลงคุณภาพของโปรแกรมให้เหมาะสมและสมบูรณ์แบบที่สุดตามความคาดหมาย หรือจุดประสงค์ที่ต้องการ

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

; Gate Automation System Program
;INPUT

MAGNET	EQU	P0.0
MANUAL	EQU	P0.1
REMOTE	EQU	P0.2
INFRA	EQU	P0.3
MODE	EQU	P0.4
POWF12V	EQU	P0.5
POWF3V	EQU	P0.6
POWF3V	EQU	P0.7

;OUTPUT

SW_M	EQU	P2.7
SLOW	EQU	P2.6
PHASE	EQU	P2.5
BOLT	EQU	P2.4
BELL	EQU	P2.3
LED_O	EQU	P2.2
LED_C	EQU	P2.1
S	EQU	P2.0

;ORTHER

WDI	EQU	P1.0	; Watchdog Input	(691)
EX_IDEL	EQU	P3.3		
RS	EQU	P1.2	; Register Select	(LCD)
E	EQU	P1.3	; Enable	(LCD)
PH	EQU	P1	; Use to LCD 7 Bit	(1-7)
SCLK	EQU	P3.5	; Serial Clock	(RTC)
I_O	EQU	P3.6	; Input/Output Data	(RTC)
RST	EQU	P3.7	; Reset	(RTC)
A0	EQU	P3.0	; Pin A of #74LS138	
B0	EQU	P3.1	; Pin B of #74LS138	
C0	EQU	P3.4	; Column 0 of Keyboard	
C1	EQU	P1.1	; Column 1 of Keyboard	
C2	EQU	P3.2	; Column 2 of Keyboard	
K1	EQU	70H		
K2	EQU	71H		
K3	EQU	72H		
K4	EQU	73H		
F15	EQU	74H		
F30	EQU	75H		
FM	EQU	76H		
DF	EQU	77H	; Display Flag	
SDF	EQU	78H	; Single/Double Flag	
AF	EQU	79H	; Alarm Flag	
GF	EQU	7AH	; Get Time Flag	
IF1	EQU	7BH	; Interupt Flag LV1	
IF2	EQU	7CH	; Interupt Flag LV2	
IF3	EQU	7DH	; Interupt Flag LV2	
KF	EQU	7EH	; Key ESC Flag	
BUFDAT	EQU	30H	; Data Buffer	(LCD)
BUFADD	EQU	31H	; Address Buffer	(LCD)
SEC	EQU	81H	; Code Read SECCOUND	(RTC)
MIN	EQU	83H	; Code Read MINUTE	(RTC)
HR	EQU	85H	; Code Read HOUR	(RTC)
DATE	EQU	87H	; Code Read DATE	(RTC)
MONTH	EQU	89H	; Code Read MONTH	(RTC)
DAY	EQU	8BH	; Code Read DAY	(RTC)
YEAR	EQU	8DH	; Code Read YEAR	(RTC)
CONT	EQU	8FH	; Code Read CONTROL	(RTC)
BURST	EQU	0BFH	; Code Read BURST	(RTC)
WD	EQU	78H	; Address for WDI Counter	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SEC_L EQU 40H ; Buffer Low Byte SEC
SEC_H EQU 41H ; Buffer High Byte SEC
MIN_L EQU 42H ; Buffer Low Byte MIN
MIN_H EQU 43H ; Buffer High Byte MIN
HR_L EQU 44H ; Buffer Low Byte HR
HR_H EQU 45H ; Buffer High Byte HR
DATE_L EQU 46H ; Buffer Low Byte DATE
DATE_H EQU 47H ; Buffer High Byte DATE
MONT_L EQU 48H ; Buffer Low Byte MONTH
MONT_H EQU 49H ; Buffer High Byte MONTH
YEAR_L EQU 4AH ; Buffer Low Byte YEAR
YEAR_H EQU 4BH ; Buffer High Byte YEAR
DAY_L EQU 4CH ; Buffer Low Byte DAY
CLOSE_MIN EQU 4FH ; Value of close the door
; in Single Mode (BCD)

IFSEC EQU 35H
COMSEC EQU 36H
ASEC_L EQU 50H ; Buffer Low Byte ASEC
ASEC_H EQU 51H ; Buffer High Byte ASEC
AMIN_L EQU 52H ; Buffer Low Byte AMIN
AMIN_H EQU 53H ; Buffer High Byte AMIN
AHR_L EQU 54H ; Buffer Low Byte AHR
AHR_H EQU 55H ; Buffer High Byte AHR

```

ORG 0000H

JMP INITIAL_SYSTEM

ORG 0003H

JMP ISR_KEY

ORG 000BH

CPL WDI ;Trig. to WDI of MAX691
RETI

;Main Program
;Gate Automation System
INITIAL_SYSTEM:

```

CALL DELAY_500ms
CALL DELAY_500ms
CALL DELAY_500ms
SETB EA ;Interupt Enable All
SETB EX0 ;External Interupt 0
SETB PTO
SETB ET0
MOV TMOD,#02H ;Timer0 Mode2 Auto

```

Reload

```

MOV TH0,#00H
MOV TLO,#00H
SETB TR0 ;Start Timer0 for

```

Watchdog

```

CLR IF1
SETB IF2 ;Default Closed Time 30

```

seconds

```

CLR IF3
CALL INI_LCD
CLR IF1
CLR IF2
CLR IF3
CALL DEFAULT_TIME
CALL DEFAULT ;Default Label
CALL BACKUP_LINE4
MOV R1,#RAM17
CALL RESTORE
MOV 4FH,A
CALL KEY_SOUND ;Sound System Ready
SETB LED_C
SETB LED_O

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CLR K1
CLR K2
CLR K3
CLR K4
MOV 3AH,#01H
```

MAIN_PROGRAM:

```
JB IF1,_RE_INI_LCD
CALL STATUS
CALL LINE_4
JNB POWF12V,_FAIL_12V
JNB POWF3V,_FAIL_3V
CALL READ_RTC
CALL OUT_RTC
JMP _NOT_FAIL
```

FAIL_12V:

```
CALL WARNING
CALL DISPLAY_12V_FAIL
DJNZ 38H,_NOT_FAIL
CALL SOUND_FAIL
```

FAIL_3V:

```
CALL WARNING
CALL DISPLAY_3V_FAIL
DJNZ 39H,_NOT_FAIL
CALL SOUND_FAIL
```

NOT_FAIL:

```
JNB POWST,ELECTRIC_OFF
```

CHECK_REMOTE:

```
JB REMOTE,_CHECK_MANUAL
CALL DELAY_250mS
JB REMOTE,_CHECK_MANUAL
JMP _CHECK_DOOR_STATUS
```

CHECK_MANUAL:

```
JB MANUAL,MAIN_PROGRAM
CALL DELAY_250mS
JB MANUAL,MAIN_PROGRAM
JMP _CHECK_DOOR_STATUS
```

RE_INI_LCD:

```
CLR IF1
CALL RESTORE_LINE4
CALL INI_LCD
JMP MAIN_PROGRAM
```

ELECTRIC_OFF:

BACKUP_DATA:

```
CALL BACKUP_LINE4 ;Keep Label
MOV R1,#RAM17 ;Keep value of closed
MOV R2,4FH
CALL BACKUP
MOV R1,#RAM19 ;Keep Ststus
MOV R2,2EH
CALL BACKUP
MOV R1,#RAM20 ;Keep flag
MOV R2,2FH
CALL BACKUP
```

E_OFF:

```
CLR EX0
CALL WAIT_ELECTRIC_ON
JNB POWST,$ ;Stop System due to
```

Electric off

```
CALL DELAY_500mS
JNB POWST,_E_OFF
```

WAIT_5MIN:

```
CALL WAIT_5MINUTE
call sound
CALL DELAY_250MS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
call sound
CALL DELAY_250MS
call sound
CALL DELAY_250MS
call sound
CALL DELAY_250MS
call sound
JNB POWST,_E_OFF
```

RESTORE_DATA:

```
CALL RESTORE_LINE4
MOV R1,#RAM17
CALL RESTORE
MOV 4FH,A
MOV R1,#RAM19
CALL RESTORE
MOV 2EH,A
MOV R1,#RAM20
CALL RESTORE
MOV 2FH,A
CALL READ_RTC
CALL OUT_RTC
CALL STATUS
CALL LINE_4
SETB EX0
JB K1,_CLOSE_BY_ELECTRIC_ON:
JB K2,_CLOSE_BY_ELECTRIC_ON:
JB K3,_CLOSE_BY_ELECTRIC_ON:
JB K4,_CLOSE_BY_ELECTRIC_ON:
```

NOT_CLOSE:

```
JMP MAIN_PROGRAM
```

CLOSE_BY_ELECTRIC_ON:

```
JMP CLOSE_DOOR
```

CHECK_DOOR_STATUS:

```
CLR EX0
JNB POWST,_NOT_CLOSE
JNB K1,OPEN_DOOR
JNB K2,OPEN_DOOR
JNB POWST,_NOT_CLOSE
JMP CLOSE_DOOR
```

OPEN_DOOR:

```
CALL BIN_TO_HEX
CALL BASE_SECTIME
MOV IFSEC,COMSEC
CALL DELAY_1SEC
CALL DELAY_250ms
JB K1,IREGULAR_OPEN
JB K2,IREGULAR_OPEN
JB K3,IREGULAR_OPEN
JB K4,IREGULAR_OPEN
JMP REGULAR_OPEN
```

IREGULAR_OPEN:

```
SETB LED_O
CLR LED_C
CALL STATUS
CLR BELL
JB K2,_IRE_K3
SETB PHASE
CALL DELAY_1SEC
CLR SW_M
JMP OPEN_K2
```

_IRE_K3:

```
JB K3,_IRE_K4
SETB PHASE
CALL DELAY_1SEC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        CLR SW_M
        JMP OPEN_K3
_IRE_K4:
        CLR SLOW
        SETB PHASE
        CALL DELAY_1SEC
        CLR SW_M
        JMP OPEN_K4
REGULAR_OPEN:
OPEN_K1:
        SETB PHASE
        CLR BOLT
        CALL DELAY_1SEC
        SETB LED_O
        CLR LED_C
        CALL STATUS
        CLR BELL
        CLR SW_M
_OPEN_UNTIL_K1:
        JNB POWST,_BACKUP_OPEN_K1
        CALL TIMER
        JNB MAGNET,_OPEN_UNTIL_K1
        SETB K1
        CALL BIN_TO_HEX
        CALL BASE_SECTIME
        MOV IFSEC,COMSEC
        CALL DELAY_1SEC
        CALL DELAY_250ms
        JMP OPEN_K2
_BACKUP_OPEN_K1:
        JMP BACKUP_OPEN
OPEN_K2:
        JB MAGNET,_INFRA_OPEN_K2
        SETB BOLT
_OPEN_UNTIL_K2:
        JNB MAGNET,$
        SETB K2
        CALL BIN_TO_HEX
        CALL BASE_SECTIME
        MOV IFSEC,COMSEC
        CALL DELAY_1SEC
        CALL DELAY_250ms
        JMP OPEN_K3
_INFRA_OPEN_K2:
        JB INFRA,_ELECTRIC_OPK2
        SETB SW_M
        JNB INFRA,$
        CLR SW_M
_ELECTRIC_OPK2:
        JNB POWST,_BACKUP_OPEN_K2
        JNB REMOTE,_TOGGLE_OPEN_K2
        JNB MANUAL,_TOGGLE_OPEN_K2
        CALL READ_RTC
        CALL OUT_RTC
        CALL TIMER
        JMP OPEN_K2
_BACKUP_OPEN_K2:
        JMP BACKUP_OPEN
_TOGGLE_OPEN_K2:
        CALL DELAY_250ms
        JNB MANUAL,_MANUAL_OPK2
        JB REMOTE,_ELECTRIC_OPK2
_MANUAL_OPK2:
        SETB SW_M

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CALL DELAY_500ms
JB MAGNET, _TOGGLE_TCK2
SETB K2
CLR SLOW
CLR SW_M
JNB MAGNET, $
SETB SW_M
SETB SLOW

_TOGGLE_TCK2:
CALL DELAY_1SEC
JMP CLOSE_DOOR

OPEN_K3:
JB MAGNET, _INFRA_OPEN_K3
CLR SLOW

_OPEN_UNTIL_K3:
JNB MAGNET, $
SETB K3
CALL BIN_TO_HEX
CALL BASE_SECTIME
MOV IFSEC, COMSEC
CALL DELAY_1SEC
CALL DELAY_250ms
JMP OPEN_K4

_INFRA_OPEN_K3:
JB INFRA, _ELECTRIC_OPK3
SETB SW_M
JNB INFRA, $
CLR SW_M

_ELECTRIC_OPK3:
JNB POWST, _BACKUP_OPEN_K3
JNB REMOTE, _TOGGLE_OPEN_K3
JNB MANUAL, _TOGGLE_OPEN_K3
CALL READ_RTC
CALL OUT_RTC
CALL TIMER
JMP OPEN_K3

_BACKUP_OPEN_K3:
JMP BACKUP_OPEN

_TOGGLE_OPEN_K3:
CALL DELAY_250ms
JNB MANUAL, _MANUAL_OPK3
JB REMOTE, _ELECTRIC_OPK3

_MANUAL_OPK3:
SETB SW_M
CALL DELAY_500ms
JB MAGNET, _TOGGLE_TCK3
SETB K3
CLR SLOW
CLR SW_M
JNB MAGNET, $
SETB SW_M
SETB SLOW

_TOGGLE_TCK3:
CALL DELAY_1SEC
JMP CLOSE_DOOR

OPEN_K4:
JB MAGNET, _INFRA_OPEN_K4
SETB SW_M
SETB SLOW
SETB BELL
SETB K4
JMP IS_DOUBLE?

_INFRA_OPEN_K4:
JB INFRA, _ELECTRIC_OPK4

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในโครงการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SETB SW M
JNB INFRA,$
CLR SW M
_ELECTRIC_OPK4:
JNB POWST,_BACKUP_OPEN_K4
JNB REMOTE,_TOGGLE_OPEN_K4
JNB MANUAL,_TOGGLE_OPEN_K4
CALL READ_RTC
CALL OUT_RTC
CALL TIMER
JMP OPEN_K4
_BACKUP_OPEN_K4:
JMP BACKUP_OPEN
_TOGGLE_OPEN_K4:
CALL DELAY_250mS
JNB MANUAL,_MANUAL_OPK4
JB REMOTE,_ELECTRIC_OPK4
_MANUAL_OPK4:
SETB SW M
CALL DELAY_500mS
JB MAGNET,_TOGGLE_TCK4
SETB K4
_TOGGLE_TCK4:
SETB SLOW
CALL DELAY_1SEC
JMP CLOSE_DOOR
BACKUP_OPEN:
SETB SW M
SETB BELL
SETB SLOW
SETB BOLT
SETB EX0
JMP MAIN_PROGRAM
IS_DOUBLE?:
JNB MODE,DOUBLE
SINGLE:
CALL DELAY_1SEC
SETB EX0
JMP MAIN_PROGRAM
DOUBLE:
CALL STATUS
JNB F15,_SET_F30
CALL DELAY_15SEC
JMP CLOSE_DOOR
_SET_F30:
JNB F30,_SET_FM
CALL DELAY_30SEC
JMP CLOSE_DOOR
_SET_FM:
CALL CHECK_1_99
JMP CLOSE_DOOR
CLOSE_DOOR:
CALL BIN_TO_HEX
CALL BASE_SECTIME
MOV IFSEC,COMSEC
CALL DELAY_1SEC
CALL DELAY_250mS
JNB K1,IREGULAR_CLOSE
JNB K2,IREGULAR_CLOSE
JNB K3,IREGULAR_CLOSE
JNB K4,IREGULAR_CLOSE
JMP REGULAR_CLOSE
IREGULAR_CLOSE:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SETB LED_C
CLR LED_O
CALL STATUS
JNB K3, _IREC_K2
CLR PHASE
CALL DELAY_1SEC
CLR SW_M
JMP CLOSE_K3

_IREC_K2:
JNB K2, _IREC_K1
CLR PHASE
CALL DELAY_1SEC
CLR SW_M
JMP CLOSE_K2

_IREC_K1:
CLR SLOW
CLR PHASE
CLR BOLT
CALL DELAY_1SEC
CLR SW_M
JMP CLOSE_K1

REGULAR_CLOSE:
CLOSE_K4:
CLR PHASE
CALL DELAY_1SEC
SETB LED_C
CLR LED_O
CALL STATUS
CLR BELL
CLR SW_M

_CLOSE_UNTIL_K4:
JNB POWST, _BACKUP_CLOSE_K4
CALL TIMER
JNB MAGNET, _CLOSE_UNTIL_K4
CLR K4
CALL BIN_TO_HEX
CALL BASE_SECTIME
MOV IFSEC, COMSEC
CALL DELAY_1SEC
CALL DELAY_250ms
JMP CLOSE_K3

_BACKUP_CLOSE_K4:
JMP BACKUP_CLOSE

CLOSE_K3:
JB MAGNET, _INFRA_CLOSE_K3

_CLOSE_UNTIL_K3:
JNB MAGNET, $
CLR K3
CALL BIN_TO_HEX
CALL BASE_SECTIME
MOV IFSEC, COMSEC
CALL DELAY_1SEC
CALL DELAY_250ms
JMP CLOSE_K2

_INFRA_CLOSE_K3:
JB INFRA, _ELECTRIC_CLK3
SETB SW_M
JNB INFRA, $
CLR SW_M

_ELECTRIC_CLK3:
JNB POWST, _BACKUP_CLOSE_K3
JNB REMOTE, _TOGGLE_CLOSE_K3
JNB MANUAL, _TOGGLE_CLOSE_K3
CALL READ_RTC

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CALL OUT_RTC
CALL TIMER
JMP CLOSE_K3
_BACKUP_CLOSE_K3:
  JMP BACKUP_CLOSE
_TOGGLE_CLOSE_K3:
  CALL DELAY_250ms
  JNB MANUAL, _MANUAL_CLK3
  JB REMOTE, _ELECTRIC_CLK3
_MANUAL_CLK3:
  SETB SW_M
  CALL DELAY_500ms
  JB MAGNET, _TOGGLE_TOK3
  CLR K3
  CLR SLOW
  CLR SW_M
  JNB MAGNET, $
  SETB SW_M
  SETB SLOW
_TOGGLE_TOK3:
  CALL DELAY_1SEC
  JMP OPEN_DOOR
CLOSE_K2:
  JB MAGNET, _INFRA_CLOSE_K2
  CLR SLOW
  CLR BOLT
_CLOSE_UNTIL_K2:
  JNB MAGNET, $
  CLR K2
  CALL BIN_TO_HEX
  CALL BASE_SECTIME
  MOV IFSEC, COMSEC
  CALL DELAY_1SEC
  CALL DELAY_250ms
  JMP CLOSE_K1
_INFRA_CLOSE_K2:
  JB INFRA, _ELECTRIC_CLK2
  SETB SW_M
  JNB INFRA, $
  CLR SW_M
_ELECTRIC_CLK2:
  JNB POWST, _BACKUP_CLOSE_K2
  JNB REMOTE, _TOGGLE_CLOSE_K2
  JNB MANUAL, _TOGGLE_CLOSE_K2
  CALL READ_RTC
  CALL OUT_RTC
  CALL TIMER
  JMP CLOSE_K2
_BACKUP_CLOSE_K2:
  JMP BACKUP_CLOSE
_TOGGLE_CLOSE_K2:
  CALL DELAY_250ms
  JNB MANUAL, _MANUAL_CLK2
  JB REMOTE, _ELECTRIC_CLK2
_MANUAL_CLK2:
  SETB SW_M
  CALL DELAY_500ms
  JB MAGNET, _TOGGLE_TOK2
  CLR K2
  CLR SLOW
  CLR SW_M
  JNB MAGNET, $
  SETB SW_M
  SETB SLOW

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

__TOGGLE_TOK2:
    CALL DELAY_1SEC
    JMP OPEN_DOOR
CLOSE_K1:
    JB MAGNET, _INFRA_CLOSE_K1
    SETB SW_M
    SETB PHASE
    SETB SLOW
    SETB BELL
    CLR K1
    CALL DELAY_1SEC
    SETB BOLT
    SETB EX0
    JMP MAIN_PROGRAM
__INFRA_CLOSE_K1:
    JB INFRA, _ELECTRIC_CLK1
    SETB SW_M
    JNB INFRA, $
    CLR SW_M
__ELECTRIC_CLK1:
    JNB POWST, _BACKUP_CLOSE_K1
    JNB REMOTE, _TOGGLE_CLOSE_K1
    JNB MANUAL, _TOGGLE_CLOSE_K1
    CALL READ_RTC
    CALL OUT_RTC
    CALL TIMER
    JMP CLOSE_K1
__BACKUP_CLOSE_K1:
    JMP BACKUP_CLOSE
__TOGGLE_CLOSE_K1:
    CALL DELAY_250ms
    JNB MANUAL, _MANUAL_CLK1
    JB REMOTE, _ELECTRIC_CLK1
__MANUAL_CLK1:
    SETB SW_M
    CALL DELAY_500ms
    JB MAGNET, _TOGGLE_TOK1
    CLR K1
__TOGGLE_TOK1:
    SETB SLOW
    SETB BOLT
    CALL DELAY_1SEC
    JMP OPEN_DOOR
BACKUP_CLOSE:
    SETB SW_M
    SETB BELL
    SETB SLOW
    SETB BOLT
    SETB EX0
    JMP MAIN_PROGRAM

```

END

กิตติกรรมประกาศ

ปริญญานิพนธ์ชิ้นนี้สามารถที่จะสำเร็จลงได้ด้วยความช่วยเหลือของผู้มีอุปการะคุณหลายฝ่าย โดยเฉพาะอย่างยิ่ง พระคุณของบิดา มารดาผู้ซึ่งให้โอกาส และทุกสิ่งทุกอย่างแก่พวกเราคณะผู้จัดทำ ตลอดจนพระคุณท่านอาจารย์ที่ปรึกษา ท่านอาจารย์ดลวิฬ ฟังมา ที่คอยให้คำปรึกษา คำแนะนำและเครื่องมือเครื่องใช้ที่ใช้ในการดำเนินการทดลอง และสุดท้ายก็คงจะเป็นกลุ่มเพื่อนๆที่คอยให้กำลังใจและแนวความคิดบางสิ่งบางอย่างแก่เรา ทางคณะผู้จัดทำจึงขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้ด้วย

คณะผู้จัดทำ

เจนวิทย์ ศรีหุมนจันทร์

สมเกียรติ เดชสุชน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

ไกรวุฒิ โรจน์ประเสริฐสุด. ไมโครโปรเซสเซอร์ 2. กรุงเทพมหานคร: เม็คทราฟพริ้นติ้ง, 2539
สุเจตน์ จันทรัมย์. ไมโครคอนโทรเลอร์ซีพเคียว 8051. กรุงเทพมหานคร: วิทยาลัยมหานคร, 2535
ยี่น ภูววรรณ. อิเล็กทรอนิกส์อุตสาหกรรม. กรุงเทพมหานคร: บริษัท ซีเอ็ดยูเคชั่น จำกัด, 2534
สัมพันธ์ หาญชล. เครื่องกลไฟฟ้า 2. กรุงเทพมหานคร: สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้