สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมา

HEMATOCRIT CENTRIFUGE

T050364

โดย

นายวัชรศักดิ์ ทำบุญ

นายสิทธิชัย ดอนสี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....
เลขทะเบียน 50364
วัน,เดือน,ปี 13 พ.ค. 2547

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมา

HEMATOCRIT CENTRIFUGE

โดย

นายวัชรศักดิ์ ทำบุญ 43015227

นายสิทธิชัย ดอนสี 43015285

อาจารย์ที่ปรึกษา

ดร.กิติพล ชิตสกุล

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้าและอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2545

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมา

ผู้จัดทำ

1. นายวัชรศักดิ์ ทำบุญ 43015227
2. นายสิทธิชัย ดอนสี 43015285

..... อาจารย์ที่ปรึกษา

(ดร. กิตติพล ชิตสกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมา

HEMATOCRIT CENTRIFUGE

นายวัชรศักดิ์ ทำบุญ 43015227 3R/1

นายสิทธิชัย ดอนสี 43015285 3R/2

โครงงานนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....

(ดร. กิติพล ชิตสกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมา

นายวัชรศักดิ์ ทำบุญ
นายสิทธิชัย ดอนสี
ดร.กิติพล ชิตสกุล (อาจารย์ที่ปรึกษา)
ภาคการศึกษาที่ 2 ปีการศึกษา 2545

บทคัดย่อ

ปริญญานิพนธ์นี้นำเสนอการพัฒนาเครื่องปั่นแยกเม็ดเลือดแดงออกจากพลาสมา ซึ่งเป็นเครื่องมือทางการแพทย์ชนิดหนึ่ง มีส่วนประกอบที่สำคัญ ได้แก่ มอเตอร์ความเร็วสูงและชุดควบคุมความเร็วของมอเตอร์ที่สามารถเลือกความเร็วได้ตามต้องการคือ 3,000 รอบต่อนาทีใช้สำหรับปั่นให้เลือดหรือยูเรียตกตะกอน และ 12,000 รอบต่อนาทีใช้สำหรับปั่นแยกเม็ดเลือดแดงออกจากพลาสมา ในปริญญานิพนธ์นี้จะให้รายละเอียดเกี่ยวกับไมโครคอนโทรลเลอร์, วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอะนาล็อก, วงจร PWM และวงจรสับไฟ (Chopper circuit) ที่ใช้ในโครงงานนี้

การทำงานของเครื่องใช้หลักการของแรงหนีศูนย์กลางเพื่อการแยกเม็ดเลือดแดงออกจากพลาสมาโดยใช้มอเตอร์ความเร็วสูง ซึ่งใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89S8252 มาเป็นตัวควบคุมระบบของเครื่องปั่นฯ โดยจะส่งข้อมูลมาควบคุมชุดแปลงสัญญาณดิจิตอลเป็นสัญญาณอะนาล็อก ให้ส่งเอาท์พุทไปคอนโทรลชุดควบคุมความเร็วของมอเตอร์ตามค่าความเร็วที่ต้องการ และมีชุดป้อนกลับที่จะทำการป้อนกลับค่าความเร็วมาให้ไมโครคอนโทรลเลอร์ทำการเปรียบเทียบกับค่าอ้างอิง เพื่อที่จะควบคุมให้ความเร็วของมอเตอร์นั้นคงที่ มีชุดแสดงผลทางจอ แอลซีดีไว้แสดงรายการต่าง ๆ ของเครื่องปั่นแยกเม็ดเลือดแดงออกจากพลาสมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HEMATOCRIT CENTRIFUGE

Mr.Watcharasak Tamboon

Mr.Sittichai Donsri

Dr. Kitiphol Chitsakul (Advisor)

2nd Semester Academic year 2002

Abstract

This thesis presents the development of a Hematocrit Centrifuge, which is a medical apparatus. This apparatus consists of a universal motor and a control circuit for stirring the blood filled tubes at specific speed of 3,000 rpm for Centrifuge and 12,000 rpm for hematocrit. The micro controller, digital to analog circuit, PWM circuit and chopper circuit used in the project have been detailed.

Based on the principle of centrifugal forces, the red blood cells are separated from the plasma by using high-speed motor. We use a micro controller family MCS-51, an AT89S8252, to control the system such as digital-to-analog converter (DAC), a universal motor and the feedback part to stabilize the speed corresponding to a set reference value. In addition, a LCD display is also included for monitoring the menu function.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สามารถสำเร็จลุล่วงได้ด้วยดีนั้น ทางผู้จัดทำขอขอบคุณอาจารย์ที่ปรึกษา ดร.กิติพล ชิตสกุล ที่ให้คำปรึกษาเกี่ยวกับข้อมูลรายละเอียดต่าง ๆ ขั้นตอนในการทำงาน รวมทั้งอาจารย์ในภาควิชาและนอกภาควิชาทุกท่านที่ได้ให้คำแนะนำ ขอขอบคุณหัวหน้าแผนกเวชศาสตร์ชันสูตร โรงพยาบาลจุฬาลงกรณ์ และเจ้าหน้าที่ทุกท่านที่ให้ความช่วยเหลือเรื่องข้อมูลตลอดจนอุปกรณ์และเครื่องมือต่าง ๆ ที่ได้บริจาคเพื่อการศึกษา ขอขอบคุณรุ่นพี่และเพื่อน ๆ ทุกคนที่ให้คำปรึกษา ขอขอบพระคุณบิดา มารดา และผู้มีพระคุณทุกท่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของโครงงาน	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 โครงสร้างของโครงงาน	2
1.4 โครงสร้างของรายงาน	2
บทที่ 2 อุปกรณ์ขับเคลื่อน	3
2.1 มอเตอร์ยูนิเวอร์ซัล (Universal Motor)	3
2.1.1 ส่วนประกอบที่สำคัญของมอเตอร์ยูนิเวอร์ซัล	3
2.1.2 การต่อมอเตอร์ยูนิเวอร์ซัล	5
2.1.3 การทำงานของมอเตอร์ยูนิเวอร์ซัล	5
2.1.4 การกลับทางหมุนของมอเตอร์ยูนิเวอร์ซัล	5
2.1.5 การบังคับความเร็วของมอเตอร์ยูนิเวอร์ซัล	6
บทที่ 3 อุปกรณ์กำลัง	7
3.1 ทรานซิสเตอร์กำลัง	7
3.1.1 การเลือกใช้ทรานซิสเตอร์กำลัง	7
3.1.2 คุณสมบัติของทรานซิสเตอร์กำลัง	7
3.1.3 คำจำกัดความที่เกี่ยวข้องกับทรานซิสเตอร์	8
3.1.4 ความสัมพันธ์กับโหลดความเหนี่ยวนำ	9
3.1.5 วงจรป้องกันทรานซิสเตอร์อิ่มตัว	9
3.1.6 เทคนิคการขับเบสสำหรับทรานซิสเตอร์	11
3.1.7 ปรากฏการณ์ Secondary Breakdown ของทรานซิสเตอร์	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 วงจรควบคุมแรงดันไฟฟ้าดีซี 20
3.2.1 ชนิดของวงจรสับไฟฟ้า 22
3.3 โฟโต้คัปเปลอร์ 25
3.3.1 โฟโต้ไอโซเลเตอร์ 25
3.3.2 วงจรทำงานพื้นฐานสำหรับโฟโต้ไอโซเลเตอร์ 28
บทที่ 4 พัลส์วิดมอดดูเลชั่น 30
4.1 พัลส์วิดมอดดูเลชั่น 30
4.2 หลักการเบื้องต้นของพัลส์วิดมอดดูเลชั่น 31
4.3 การทำงานของไอซี 33
4.4 การใช้ไอซี TL 494 ร่วมกับทรานซิสเตอร์ภายนอก 35
บทที่ 5 การใช้งานไมโครคอนโทรลเลอร์ MCS-51 37
5.1 คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ MCS-51 38
5.2 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51 40
5.3 ไทเมอร์/เคาน์เตอร์ในไมโครคอนโทรลเลอร์ MCS-51 43
5.3.1 การทำงานเป็นไทเมอร์ 43
5.3.2 การทำงานเป็นเคาน์เตอร์ 43
5.3.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของไทเมอร์/เคาน์เตอร์ 44
5.3.4 โหมดการทำงานของไทเมอร์/เคาน์เตอร์ 46
5.4 อินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51 48
5.4.1 การจัดการอินเตอร์รัปต์ 48
5.4.2 การเขียนโปรแกรมย่อยบริการอินเตอร์รัปต์ 49
5.4.3 รีจิสเตอร์ที่เกี่ยวข้องกับการบริการอินเตอร์รัปต์ 49
5.4.4 แหล่งกำเนิดสัญญาณอินเตอร์รัปต์ 51
5.4.5 ลำดับความสำคัญของการอินเตอร์รัปต์ 52
5.5 การขับโมดูลแสดงผลแบบผลึกเหลว (LCD Module) 52
5.5.1 รายละเอียดเกี่ยวกับโมดูล LCD 52
5.5.2 โครงสร้างภายในของตัวควบคุมโมดูล 53
5.5.3 โมดูล LCD ขนาด 24 ตัวอักษร 2 บรรทัด 54
5.5.4 คำสั่งควบคุมโมดูล LCD 55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5.5 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD 58
5.5.6 จังหวะการทำงานของ LCD โมดูล 59
5.6 การขยายพอร์ตอินพุทเอาท์พุทด้วยอุปกรณ์ระบบบัส I^2C 59
5.6.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C 59
5.6.2 คุณสมบัติโดยทั่วไปของบัส I^2C 60
5.6.3 หลักการของบัน I^2C 61
5.6.4 สภาวะที่เกิดขึ้นบนบัส I^2C 61
5.6.5 การทำงานบนบัส I^2C 63
5.6.6 อุปกรณ์ที่ใช้การเชื่อมต่อบัส I^2C 64
5.6.7 การต่ออุปกรณ์ระบบบัส I^2C กับไมโครคอนโทรลเลอร์ 65
5.6.8 การเขียนโปรแกรมติดต่อบัส I^2C 66
5.6.9 การขยายจำนวนพอร์ตอินพุทเอาท์พุทด้วยไอซี PCF8574A 67
5.7 วงจรแปลงสัญญาณดิจิตอลเป็นอะนาลอกแบบ R-2R 70
5.7.1 ความรู้เบื้องต้นเกี่ยวกับวงจร DAC แบบ R-2R 70
5.7.2 การเชื่อมต่อวงจร DAC เข้ากับไมโครคอนโทรลเลอร์ 72
บทที่ 6 การออกแบบและขั้นตอนการทำงานของโปรแกรม 73
6.1 การออกแบบ 73
6.1.1 การออกแบบตัวนับรอบความเร็วมอเตอร์: Counter 73
6.1.2 การออกแบบตัวตั้งเวลา: Timer 74
6.1.3 การออกแบบโปรแกรมซอฟท์สตาร์ท 75
6.1.4 การออกแบบโปรแกรมชดเชยความเร็ว 75
6.2 ขั้นตอนการทำงานของโปรแกรม 76
6.2.1 ขั้นตอนการเริ่มต้นการทำงาน 77
6.2.2 ขั้นตอนการเลือกโหมดการทำงาน 79
6.2.3 ขั้นตอนการทำงานของโหมดที่เลือกไว้ 80
6.2.4 ขั้นตอนการจบการทำงาน 89
บทที่ 7 การทดลองและผลการทดลอง 91
7.1 การทดลองภาคควบคุมการขับมอเตอร์ 91
7.2 การทดลองภาคขับมอเตอร์ 92

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3 การทดลองชุดตรวจจับกระแสเกิน 94
7.4 การทดลองวัดกระแสที่ไหลผ่านมอเตอร์ เทียบกับความเร็วรอบที่ได้ 94
7.5 การทดลองวัดการเพิ่มความเร็วรอบมอเตอร์เทียบกับเวลา 95
7.6 การทดลองวัดความเร็วรอบด้วยโปรแกรมที่เขียนขึ้นเทียบ
กับเครื่องวัดความเร็วรอบของ YOKOGAWA 97
7.7 การทดลองปั่นเลือดจริงด้วยเครื่องที่สร้างขึ้นเทียบกับ
เครื่องที่โรงพยาบาลใช้อยู่ 98
บทที่ 8 บทสรุป 100
บรรณานุกรม 102
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของโครงงาน

เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมา เป็นเครื่องมือทางการแพทย์อย่างหนึ่ง ที่จำเป็นในห้องปฏิบัติการ แต่มีราคาค่อนข้างสูงเนื่องจากต้องนำเข้ามาจากต่างประเทศ ส่วนประกอบหลักที่สำคัญของเครื่องปั่นนี้ ประกอบด้วย มอเตอร์ความเร็วสูง ชุดขับมอเตอร์ และชุดตั้งเวลา ซึ่งในห้องปฏิบัติการหลายแห่งยังใช้เครื่องที่เป็นระบบกำหนดเองอยู่ และยังสามารถทำงานได้ที่ความเร็วเดียวเท่านั้น ดังนั้นถ้าหากเราสามารถประยุกต์ใช้วงจรอิเล็กทรอนิกส์และไมโครคอนโทรลเลอร์ไปควบคุมการทำงานของเครื่องปั่นแล้ว จะทำให้เครื่องปั่นทำงานในระบบ กึ่งอัตโนมัติได้ ทั้งยังสามารถปรับความเร็วของมอเตอร์ได้อีกด้วย ทำให้สามารถใช้เครื่องปั่นในงานที่แตกต่างกันได้ ทำให้เกิดความสะดวกในการใช้งาน และ เป็นการเพิ่มประสิทธิภาพการทำงานให้กับเครื่องปั่นได้เป็นอย่างดี

จากความคิดข้างต้นนี้ จึงนำไปสู่จุดเริ่มต้นของโครงงานพัฒนาเครื่องปั่นเม็ดเลือดออกจากพลาสมา โดยใช้ไมโครคอนโทรลเลอร์ในการควบคุมการทำงานของเครื่องปั่น โดยสามารถปรับความเร็วของเครื่องปั่นได้ 2 ระดับ หรือทำงานปั่นได้ 2 แบบ มีไทเมอร์จับเวลา มีจอแสดงผล LCD ให้ผู้ใช้งานติดต่อหรือใช้งานเครื่องปั่นได้สะดวก และมีชุดตรวจจับความเร็วเพื่อนำมาประมวลผลในการชดเชยความเร็วให้คงที่ตลอดเวลา

1.2 วัตถุประสงค์ของโครงงาน

ศึกษาวงจรต่างๆที่สำคัญ ที่เกี่ยวกับการควบคุมและการขับมอเตอร์ ให้มีเสถียรภาพทางด้านความเร็ว และสามารถนำมาใช้งานได้จริง เช่น

- วงจรพัลส์วิดธ์มอดูเลเตอร์
- วงจรดิจิตอลอะนาล็อกคอนเวอร์เตอร์
- วงจรขับมอเตอร์

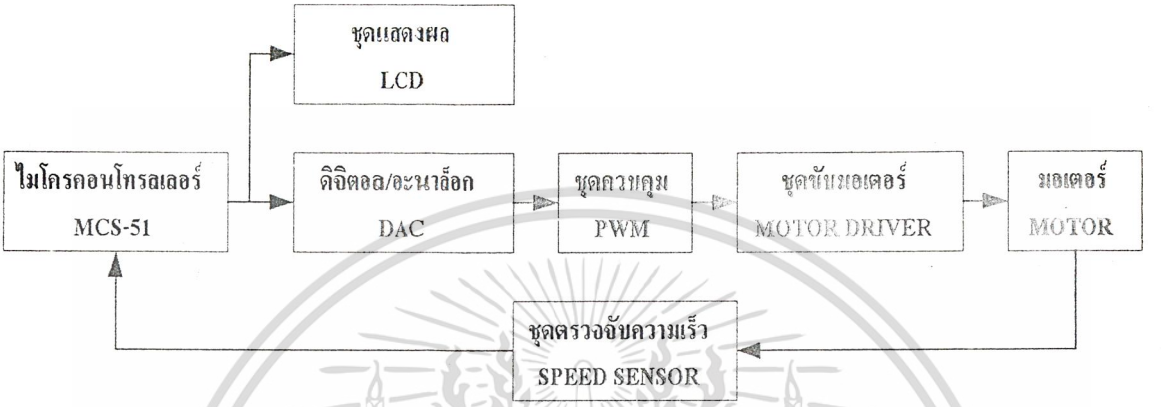
ศึกษาการเขียนโปรแกรมแอสเซมบลี เพื่อ

- ควบคุมการทำงานของไมโครคอนโทรลเลอร์
- การขับจอแสดงผล LCD
- การตั้งเวลาหรือไทเมอร์
- ตรวจจับความเร็วและชดเชยความเร็ว
- การส่งข้อมูลผ่าน I2C BUS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 โครงสร้างของโครงงาน

โครงสร้างของโครงงานจะประกอบด้วยชุดควบคุมซึ่งใช้วงจร พัลส์วิดมอดดูเลชั่น PWM และชุดขับมอเตอร์ ซึ่งใช้วงจรทรานซิสเตอร์ในการขับมอเตอร์ โครงสร้างของโครงงานแสดงดังรูปที่ 1.1



รูปที่ 1.1 โครงสร้างของโครงงาน

1.4 โครงสร้างของรายงาน

รายงานนี้ได้รวบรวมรายละเอียด แนวคิด การสร้างและทดสอบโครงงาน โดยได้มีการแบ่งรายงานออกเป็นบทต่างๆ ทั้งนี้เพื่อความสะดวกต่อการศึกษาและทำความเข้าใจ ในแต่ละบทประกอบด้วยเนื้อหาสำคัญดังนี้

บทที่ 1 ได้กล่าวถึงความเป็นมาของโครงงานนี้ และเนื้อหาของรายงานพอสังเขป

บทที่ 2 กล่าวถึงอุปกรณ์ที่ใช้ในการขับเคลื่อน

บทที่ 3 อุปกรณ์กำลัง

บทที่ 4 พัลส์วิดมอดดูเลชั่น

บทที่ 5 การใช้งานไมโครคอนโทรลเลอร์ จอแสดงผล LCD และการส่งข้อมูลผ่าน I2C BUS

บทที่ 6 การออกแบบและขั้นตอนการทำงานของโปรแกรม

บทที่ 7 การทดลองและผลการทดลอง

บทที่ 8 สรุปและนำเสนอปัญหาที่เกิดขึ้น ในการทำงาน แนวทางการแก้ไข และการพัฒนาต่อไปในอนาคต

ภาคผนวก กล่าวถึง การใช้งานหรือคู่มือการใช้งานเครื่องปั่นฯ

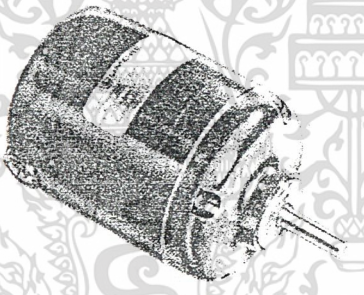
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
อุปกรณ์ขับเคลื่อน

2.1 มอเตอร์ยูนิเวอร์ซัล (UNIVERSAL MOTORS)

มอเตอร์ยูนิเวอร์ซัล หรือบางทีเรียกซีรี่มอเตอร์ เป็นมอเตอร์ที่ใช้ได้ทั้งไฟกระแสตรง (Direct Current) และกระแสสลับ(Alternating Current) มอเตอร์ชนิดนี้ที่มีใช้ส่วนมากเป็นมอเตอร์ขนาดเล็ก เช่น มอเตอร์เครื่องตัดผมบางแบบ มอเตอร์สว่านเจาะ มอเตอร์เครื่องดูดฝุ่น มอเตอร์เครื่องขัดพื้น และมอเตอร์เย็บผ้า ฯลฯ เป็นต้น

ยูนิเวอร์ซัลมอเตอร์นี้มีกำลังสตาร์ท มีความเร็วสูง ปกติจะหมุนตั้งแต่ 3,000 รอบ ถึง 8,000 รอบต่อนาที บางตัวหมุนถึง 10,000 หรือ 20,000 รอบต่อนาที ความเร็วเปลี่ยนแปลงตามงานที่ใช้ มอเตอร์ชนิดนี้เมื่อวิ่งตัวเปล่า (ไม่ใช้อะไร) ความเร็วจะสูงขึ้นอย่างรวดเร็ว อย่างน่ากลัว ดังนั้น มอเตอร์ชนิดนี้ เมื่อจะใช้หมุนอะไรก็มักจะติดตั้งอยู่กับสิ่งนั้นเสมอ เพื่อไม่ให้มีโอกาสหมุนตัวเปล่าเป็นเวลานาน มอเตอร์อาจจะเสียหายได้ ดูรูปที่ 2.1



รูปที่ 2.1 ลักษณะของยูนิเวอร์แซลมอเตอร์

2.1.1 ส่วนประกอบที่สำคัญของมอเตอร์ยูนิเวอร์ซัล

1. โครง **(Frame)** หรือเปลือกของมอเตอร์ เป็นที่ยึดแกนเหล็ก หรือ คอล์ยของมอเตอร์ เปลือกของมอเตอร์ นี้อาจจะทำด้วยเหล็กหล่อเหล็กเหนียว หรืออลูมิเนียมก็ได้

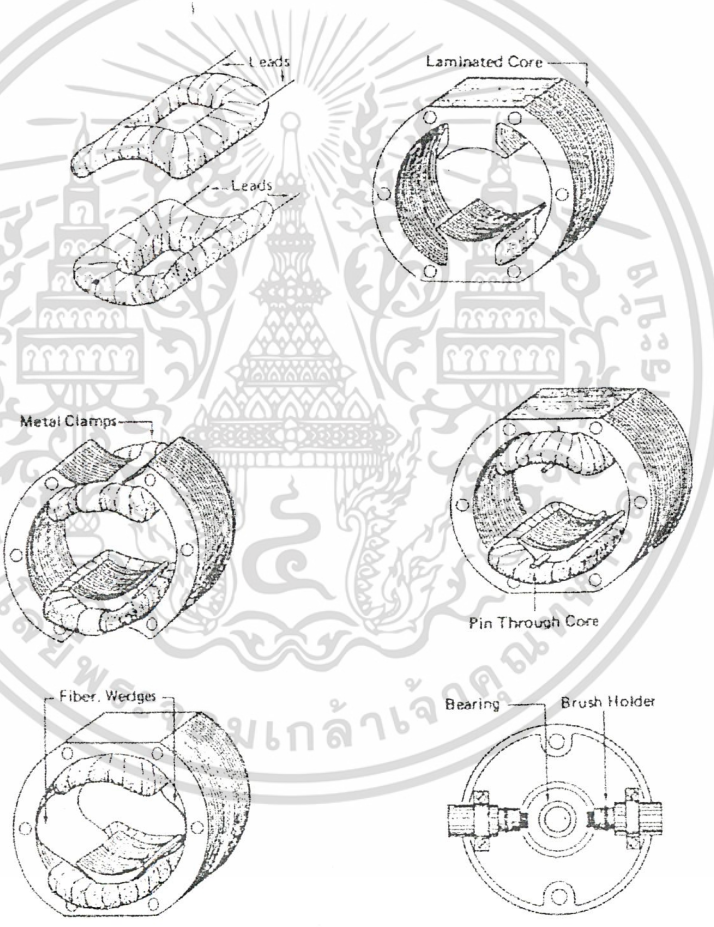
2. **สเตเตอร์ (Stator)** หรือ บางทีเรียกแกนเหล็ก บางทีก็เรียกคอร์ (Core) บางทีเรียกฟิลด์คอร์ (Field Core) ตามภาษาอังกฤษ หากจะแปลตามภาษาอังกฤษก็คือ แกนเหล็กที่ทำให้เกิดสนามแม่เหล็ก ส่วนมากฟิลด์คอร์ของมอเตอร์ยูนิเวอร์ซัลนี้ มักจะมี 2 โพล คือมีโพลยื่นออกมา 2 โพล ฟิลด์คอร์หรือสเตเตอร์ประกอบด้วยแผ่นเหล็กซิลิกอน (Silicon sheet Steel) อัดติดกันแน่นด้วยสกรูหรือน๊อต หรือหมุดย้ำเหมือนกับสเตเตอร์รีพัลชั่น คอล์ย 2 ชุดที่พันอยู่กับสเตเตอร์หรือคอร์เราเรียกว่าฟิลด์คอล์ย (Field Coil)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. **อเมเจอร์ (Armature)** คล้ายอเมเจอร์หรือโรเตอร์ของมอเตอร์รีพัลชั่น ทั้งในลักษณะการพันคอล์ยและแกนเหล็ก ผิดกันเฉพาะคอมมิวเตเตอร์ของมอเตอร์ยูนิเวอร์ซัล ไม่มีการช๊อร์ทกันไว้เหมือนมอเตอร์รีพัลชั่น

4. **ฝาปิดทั้งสองข้าง (End Plate)** เหมือนกับมอเตอร์อื่นๆ ฝาปิดทั้งสองข้างของมอเตอร์เป็นที่ติดตั้งลูกปืน เป็นที่รองรับของเพลา หรือแกนของมอเตอร์ ที่ฝาข้างหนึ่งของมอเตอร์ยูนิเวอร์ซัล จะมีแปรงถ่านติดตั้งอยู่ 2 ชุดเหมือนมอเตอร์รีพัลชั่น

5. **แปรงถ่าน (Brush)** แปรงถ่านต้องพอดีกับซองถ่าน อย่าให้หลวมหรือแน่นจนเกินไป แปรงถ่านมีหน้าที่เป็นตัวนำกระแสไฟ

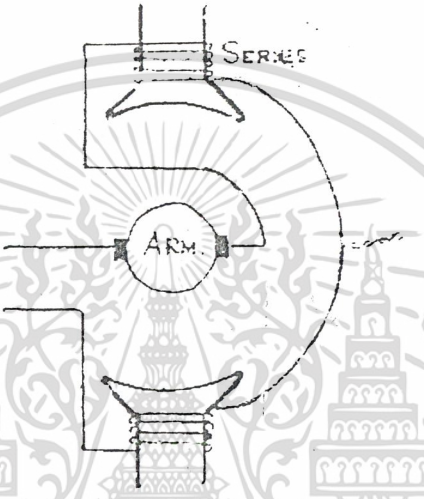


รูปที่ 2.2 แสดงโครงสร้างภายในของมอเตอร์ยูนิเวอร์ซัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 การต่อมอเตอร์ยูนิเวอร์ซัล

การต่อมอเตอร์ยูนิเวอร์ซัล คล้ายกับการต่อมอเตอร์รีพัลชั่น ผิดกันที่วงจรของฟิลด์คอล์ยกับวงจรของอเมเจอร์ของมอเตอร์ยูนิเวอร์ซัล ต่อซีรี่กันด้วยแปรงถ่าน คือสายเมนที่ต่อเข้ามอเตอร์ 2 เส้น ต่อตรงเข้ากับฟิลด์คอล์ย ทั้งสอง เส้นละคอล์ย ปลายอีกข้างหนึ่งของฟิลด์คอล์ย ต่อเข้ากับแปรงถ่าน แปรงถ่านทั้งสองจะเป็นสะพานต่อเชื่อมระหว่างอเมเจอร์ และฟิลด์คอล์ยให้ซีรี่กัน ดูรูปที่ 2.3



รูปที่ 2.3 การต่อซีรี่ส์ ของมอเตอร์ยูนิเวอร์ซัล

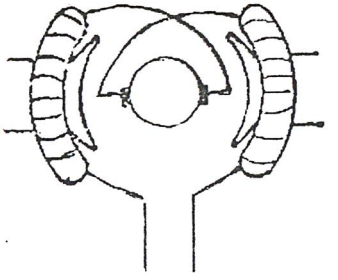
2.1.3 การทำงานของมอเตอร์ยูนิเวอร์ซัล

มอเตอร์แบบนี้อเมเจอร์และขดลวดสนามแม่เหล็ก (Field Coil) ต่ออันดับกัน ดังนั้นเมื่อเราสับสวิตช์ กระแสก็จะไหลผ่านทั้งอเมเจอร์ และขดลวดสนามแม่เหล็ก จึงทำให้เกิดสนามแม่เหล็กไฟฟ้าขึ้นทั้ง 2 ส่วน คือ ทางอเมเจอร์ และ ขดลวดสนามแม่เหล็ก มาผลักซึ่งกันและกัน อเมเจอร์จึงเคลื่อนตัวหมุนไปได้

2.1.4 การกลับทางหมุนของมอเตอร์ยูนิเวอร์ซัล

เมื่อมอเตอร์หมุนไปทางหนึ่ง ต้องการจะให้หมุนไปอีกทางหนึ่ง วิธีกลับทางหมุนได้ง่ายที่สุดก็คือ สลับสายที่ต่อเข้ากับแปรงถ่านนั้น คือสมมติว่า ฟิลด์คอล์ย ก ต่อเข้ากับแปรงถ่าน 1 และฟิลด์คอล์ย ข ต่อเข้ากับแปรงถ่าน 2 ให้สลับสายฟิลด์คอล์ย ก ไปต่อแปรงถ่าน 2 และเอาสายฟิลด์คอล์ย ข ไปต่อกับแปรงถ่าน 1 ดูรูปที่ 2.4 ประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การต่อมอเตอร์ยูนิเวอร์ซัลกลับทางหมุน

2.1.5 การบังคับความเร็วของมอเตอร์ยูนิเวอร์ซัล

มอเตอร์ยูนิเวอร์ซัล สามารถที่จะปรับหรือบังคับความเร็วของการหมุนได้หลายวิธี ดังนี้

1. ใช้ลวดความต้านทาน (Resistance Wire) โดยใช้ลวดความต้านทานต่อซีรี่กับมอเตอร์ อย่างเช่นมอเตอร์จักรเย็บผ้า ลวดความต้านทานจะอยู่ที่เราใช้เท้าเหยียบนั้น เมื่อมีลวดความต้านทานต่อซีรี่กับวงจรของมอเตอร์มาก มอเตอร์จะหมุนช้า เมื่อลดลวดความต้านทานออก มอเตอร์จะหมุนเร็วขึ้น ที่บังคับความเร็วของมอเตอร์จักรบางแบบใช้ถ่าน (Carbon Block) ทำเป็นแผ่นกลมๆหลายๆ แผ่นเรียงซ้อนกันอยู่ หากเราเหยียบให้แผ่นถ่านชิดกันแน่น มอเตอร์จะหมุนเร็ว เมื่อเราปล่อยให้แปรงถ่านติดกันหลวมๆ จะมีความต้านทานเกิดขึ้น มอเตอร์จะหมุนช้าลง

2. พันฟิลด์คอล์ยเป็นหลายชุด เมื่อต้องการให้มอเตอร์หมุนเร็ว ก็ใช้ฟิลด์คอล์ยรอบน้อยๆ การต้องการให้หมุนช้า ก็ใช้ฟิลด์คอล์ยรอบมาก ซึ่งมีลักษณะเหมือนโช๊ค (Choke) ของมอเตอร์โช๊ดโพล หรือมอเตอร์คาปาซิเตอร์

3. ใช้เซ็นตริฟิวกัลสวิตช์ ปกติมอเตอร์เช่น มอเตอร์ผสมของ มอเตอร์บดของ มอเตอร์กวนของ มักจะมีเซ็นตริฟิวกัลสวิตช์สำหรับคอนโทรลความเร็วของมอเตอร์ สามารถจะปรับให้มอเตอร์หมุนช้าเร็วเท่าใดก็ได้ การทำงานของเซ็นตริฟิวกัลสวิตช์นี้ คือเมื่อมอเตอร์หมุนไปด้วยแรงเหวี่ยงของการหมุน ส่วนหนึ่งของเซ็นตริฟิวกัลสวิตช์ จะดันหน้าทองขาวให้ตัดวงจรของมอเตอร์ออก คือไม่ให้ไฟผ่านมอเตอร์ มอเตอร์ก็จะหมุนช้าลง เมื่อช้าลงถึงจุดความเร็วที่ตั้งไว้ เซ็นตริฟิวกัลสวิตช์ ก็จะตัดวงจรอีก มอเตอร์ก็จะหมุนเร็วขึ้น เมื่อหมุนเร็วมาก เซ็นตริฟิวกัลสวิตช์ก็จะตัดวงจรอีก เป็นจังหวะสลับกันไปเช่นนี้ แต่จังหวะตัดและต่อวงจรจะถี่มาก จนแทบไม่รู้ว่ามีการตัดและต่อวงจรตลอดเวลา การตัดและต่อวงจรนี้ ทำให้หน้าทองขาวสึกเร็ว เขาจึงใช้คาปาซิเตอร์(Capacitor) และรีซิสเตอร์ (Resistor) ต่อขนานกันและคร่อมไว้ ระหว่างหน้าทองขาวทั้งสองนั้น เพื่อกันการสปาร์คซึ่งทำให้การสึกกร่อนของหน้าทองขาวน้อยลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
อุปกรณ์กำลัง

3.1 ทรานซิสเตอร์กำลัง

3.1.1 การเลือกใช้ทรานซิสเตอร์กำลัง

พารามิเตอร์พื้นฐานของทรานซิสเตอร์สำหรับการออกแบบ CONVERTER คือ

1) ความสามารถในการทนแรงดันขณะเปิดวงจร

2) ความสามารถในการทนกระแสขณะปิดวงจร

พารามิเตอร์ทั้งสองจะขึ้นอยู่กับชนิดของ CONVERTER ด้วยอุปกรณ์ที่นิยมนำมาใช้เป็นสวิตช์อิเล็กทรอนิกส์นั้นได้แก่ ทรานซิสเตอร์ และมอสเฟท ซึ่งอุปกรณ์สวิตช์ทั้งสองชนิดนั้นต่างมีข้อดีและข้อเสียแตกต่างกัน ในปัจจุบันทรานซิสเตอร์จะมีราคาที่ถูกกว่ามอสเฟท ในขณะที่มอสเฟทนั้นมีการใช้งานที่ง่ายกว่าทรานซิสเตอร์ ทั้งนี้เนื่องจากการออกแบบวงจรขับเกทไม่มีความซับซ้อนมากเกินไปเมื่อเทียบกับวงจรขับเบสของทรานซิสเตอร์ และอีกสิ่งหนึ่งที่สำคัญมากก็คือการทำงานของทรานซิสเตอร์นั้นถูกจำกัดด้วยความถี่คัทออฟ (Cut off) ของตัวมันเอง (ประมาณ 500 kHz) แต่มอสเฟทกลับสามารถทำงานได้ด้วยความถี่สูง ๆ ถึง 200 kHz ทีเดียว และแน่นอนว่าการทำงานที่ความถี่สูง ๆ ขนาดของอุปกรณ์ต่าง ๆ ย่อมมีขนาดเล็กลง ซึ่งเป็นข้อดีในการลดขนาดของแหล่งจ่ายไฟตรงสวิตชิ่งในตัว

3.1.2 คุณสมบัติของทรานซิสเตอร์กำลัง

ทรานซิสเตอร์เป็นอุปกรณ์ที่ต้องใช้กระแสในการขับเบสเพื่อควบคุมจำนวนการไหลของกระแสคอลเลคเตอร์โดยจำนวนการไหลของกระแสนี้จะขึ้นอยู่กับอัตราการขยายกระแสของทรานซิสเตอร์ตามความสัมพันธ์

$$\beta = \frac{I_C}{I_B} \qquad (3.1)$$

เมื่อ I_C คือ กระแสคอลเลคเตอร์

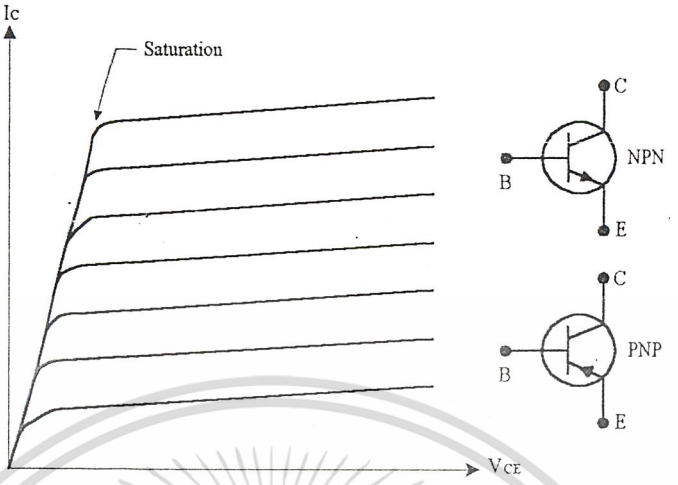
I_B คือ กระแสเบส

การทำงานของทรานซิสเตอร์จะแบ่งการทำงานเป็น 2 สภาวะ

1) สภาวะเชิงเส้น (Linear) มักใช้ในงานด้านวงจรขยาย

2) สภาวะอิ่มตัว (Saturation) มักใช้ในงานเป็นสวิตช์ปิด – เปิดวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงคุณสมบัติของทรานซิสเตอร์

ในรูปที่ 3.1 แสดงกราฟลักษณะสมบัติความสัมพันธ์ระหว่างแรงดัน และกระแสของทรานซิสเตอร์ จากกราฟแสดงถึงบริเวณอิ่มตัว ซึ่ง ณ ที่บริเวณนี้จะพบว่ากระแสเบสจำนวนหนึ่งสามารถทำให้ทรานซิสเตอร์ปิดวงจร (นำกระแสเต็มที่) เกิดกระแสคอลเลคเตอร์จำนวนมากไหลผ่านในขณะที่แรงดันตกคร่อมคอลเลคเตอร์กับอิมิตเตอร์มีค่าต่ำมาก

ในความเป็นจริงแล้วการนำทรานซิสเตอร์มาทำเป็นสวิตช์นั้นสามารถควบคุมได้ที่ขาเบสของทรานซิสเตอร์ โดยใช้กระแสขับที่ขาเบสเพื่อให้ทรานซิสเตอร์ปิดวงจร และถ้ากลับทิศทางของกระแสเบสก็จะทำให้ทรานซิสเตอร์เปิดวงจร

3.1.3 คำจำกัดความที่เกี่ยวข้องกับทรานซิสเตอร์ (โหลดชนิดความต้านทาน)

พิจารณารูป 3.2 เป็นรูปสัญญาณแรงดันตกคร่อมระหว่างเบสกับอิมิตเตอร์ (V_{BE}) และคอลเลคเตอร์กับอิมิตเตอร์ (V_{CE}) ของทรานซิสเตอร์ NPN โดยควบคุมที่ I_B

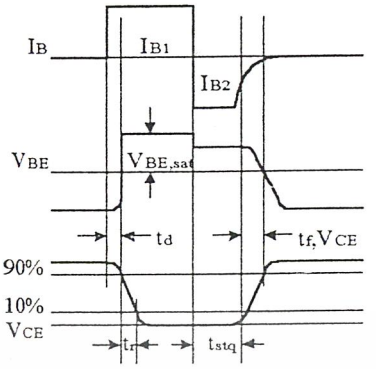
เวลาหน่วง (Delay Time, t_d) คือ ช่วงเวลาจากการจ่ายกระแสเบส I_{B1} ไปยังจุดที่ทำให้ V_{CE} ตกลงถึง 90% ของค่าแรงดันเริ่มต้น

เวลาขอบขาขึ้น (Rise Time, t_r) คือ ช่วงเวลาเมื่อ V_{CE} ตกลงถึง 10% ของค่าแรงดันเมื่อ 90%

เวลาสะสม (Storage Time, t_{stg}) คือ ช่วงเวลาเริ่มจากการกลับทิศทางของกระแส I_{B2} เป็นผลทำให้ขนาดของแรงดัน V_{CE} เพิ่มขึ้น 10% ของค่าแรงดันครั้งสุดท้ายที่ OFF

เวลาขอบขาลง (Fall Time, $t_{f,VCE}$) คือ ช่วงเวลาเมื่อแรงดัน V_{CE} ที่เพิ่มขึ้นตั้งแต่ 10% จนถึง 90%

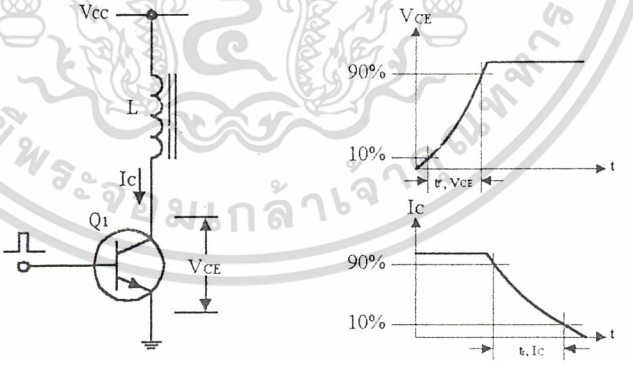
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงรูปคลื่นของทรานซิสเตอร์สวิตชิ่ง

3.1.4 ความสัมพันธ์กับโหลดความเหนี่ยวนำ (Inductive load)

จากที่กล่าวมาแล้วในเรื่องของคำจำกัดความต่าง ๆ ของช่วงเวลาในการสวิตชิ่งของทรานซิสเตอร์ในเทอมของแรงดัน V_{CE} เมื่อโหลดที่ใช้นั้นเป็นความต้านทาน อย่างไรก็ดี เมื่อนำทรานซิสเตอร์ไปขับโหลดที่เป็นความเหนี่ยวนำ รูปสัญญาณที่ได้นั้นย่อมแตกต่างกัน เนื่องจากคุณสมบัติของความเหนี่ยวนำที่ถึงแม้จะจ่ายแรงดันให้กับความเหนี่ยวนำแล้วก็ตาม กระแสก็ยังไม่สามารถไหลได้ในทันทีทันใด ดังนั้นในช่วงที่ทรานซิสเตอร์เปิดวงจร เราจะพบว่าแรงดัน V_{CE} ของทรานซิสเตอร์จะมีค่าเท่ากับแรงดันแหล่งจ่ายก่อนที่กระแสจะลดลงดังรูป 3.3



รูปที่ 3.3 แสดงรูปคลื่นแรงดันกับกระแสของทรานซิสเตอร์เมื่อขับโหลดความเหนี่ยวนำ

3.1.5 วงจรป้องกันทรานซิสเตอร์อิ่มตัว

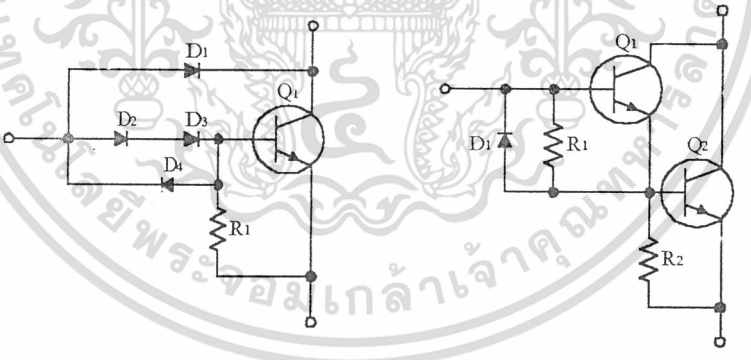
การอิ่มตัวและการเกิดช่วงเวลาสะสมจะมีผลทำให้ความเร็วในการสวิตช์ของทรานซิสเตอร์ลดลง ดังนั้นวิธีหลีกเลี่ยงเหตุการณ์ดังกล่าวจึงมีการคิดค้นเทคนิคต่าง ๆ เพื่อเพิ่มประสิทธิภาพของแหล่งจ่ายไฟสวิตชิ่งนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.4 (ก) แสดงการใช้ไดโอดป้องกันการอิ่มตัว (ANTISATURATION DIODES) หรือบางตำราเรียกว่าไดโอด BAKER CLAMP เมื่อทรานซิสเตอร์ปิดวงจรจะเห็นว่าที่ขาเบสนั้นมีไดโอด D_2, D_3 ต่ออนุกรมอยู่ สมมติทุกตัวมีแรงดันตกคร่อมเมื่อนำกระแสเท่ากับ 0.7 V ดังนั้นแรงดันที่ขาเบสจะต้องไม่ต่ำกว่าแรงดันอินพุท 1.4 V แต่เนื่องจากที่ขาคอลเลคเตอร์ของทรานซิสเตอร์นั้นยังมีไดโอด D_1 ต่ออยู่อีกหนึ่งตัวก็จะทำให้มีแรงดันตกคร่อมเมื่อนำกระแสอีก 0.7V ทำให้แรงดันคอลเลคเตอร์ต้องไม่ต่ำกว่าแรงดันอินพุท 0.7 V ด้วย เพราะฉะนั้นแรงดันที่คอลเลคเตอร์จะต้องมีค่ามากกว่าแรงดันที่ขาเบสเท่ากับ 1.4 – 0.7 = 0.7 V ทำให้ทรานซิสเตอร์ทำงานในสภาวะที่ไม่มีโอกาสอิ่มตัวได้เลย

ไดโอดป้องกันการอิ่มตัวนี้ จำเป็นจะต้องใช้ชนิดที่มีความเร็วสูง เช่น ชนิด Fast recovery เนื่องจากความถี่ที่ใช้ในการสวิตช์นั้นมีความเร็วสูงตั้งแต่ 20 kHz ขึ้นไป ไดโอด D_2 และ D_3 ควรมีแรงดัน Reverse blocking rating ต่ำ ๆ แต่สำหรับไดโอด D_1 จะต้องมีค่าแรงดันไม่ต่ำกว่า 2 V_{CE} โดยทั่วไปใช้ประมาณ 800 V PIV

ส่วนไดโอด D_4 เป็นไดโอดประเภท " Wrap – around " มีหน้าที่ดึงกระแสเบสของทรานซิสเตอร์เปิดวงจร โดยเป็นตัวคายประจุของค่าเก็บประจุระหว่างเบสกับอิมิตเตอร์ ทำให้ช่วงเวลาสะสมลดลง



(ก) (ข)

รูปที่ 3.4 วงจรป้องกันทรานซิสเตอร์อิ่มตัว

(ก) ไดโอดป้องกันการอิ่มตัวของ Q_1

(ข) การใช้วงจรดาร์ลิงตันป้องกัน Q_2 อิ่มตัว

สำหรับในรูป 3.4 (ข) แสดงการใช้เทคนิคของวงจรดาร์ลิงตัน (darlington) โดยมีหลักการทำงานคล้ายกับเทคนิคแรก คือ ทรานซิสเตอร์ Q_1 จะเป็นตัวป้องกันมิให้ทรานซิสเตอร์ Q_2 เกิดการอิ่มตัว แต่มีข้อแม้คือ ทรานซิสเตอร์ Q_1 จะต้องเปิดวงจรก่อนทรานซิสเตอร์ Q_2 ไดโอด D_1 มีหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ลดอิมพีแดนซ์ให้กับกระแสเบสย้อนกลับของ Q_2 เพื่อเปิดวงจร ตัวต้านทาน R_1 และ R_2 เป็นความต้านทานต่ำ

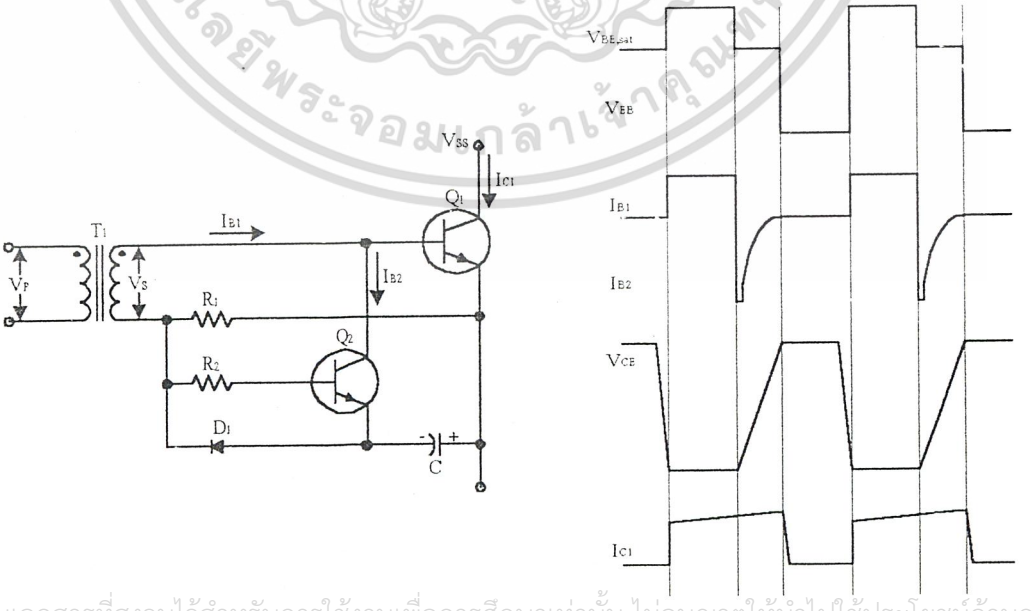
3.1.6 เทคนิคการขับเบสสำหรับทรานซิสเตอร์

วงจรขับกระแสคงที่ (Constant Drive Current Circuits)

จากที่ได้กล่าวมาแล้วในข้างต้น เกี่ยวกับข้อจำกัดต่าง ๆ ของการใช้ทรานซิสเตอร์เป็นสวิตช์ ทำให้พอที่จะทราบถึงวิธีการที่จะหลีกเลี่ยงมิให้เกิดการอิ่มตัว โดยการควบคุมให้กระแส I_{B1} มีขนาดที่พอเพียงและลดช่วงเวลาสะสมให้น้อยที่สุด เพื่อเป็นการเพิ่มประสิทธิภาพการสวิตชิ่ง และควบคุมกระแสย้อนกลับ I_{B2} ให้มีขนาดที่เพียงพอเช่นเดียวกัน

สังเกตพบว่าเมื่อกระแส I_{B2} มีค่าเพิ่มขึ้นจะส่งผลให้ช่วงเวลาสะสมและช่วงเวลาขาลงนั้นลดลง ทำให้แรงดัน V_{EB} เพิ่มขึ้นด้วย ปรากฏการณ์ลักษณะนี้จะทำให้พลังงาน reverse-bias secondary break down หรือ E_{SB} นั้นลดลงด้วย และสำหรับการออกแบบวงจรขับจะต้องคำนึงถึงเรื่องกระแสย้อนกลับเป็นพิเศษมิฉะนั้นจะเป็นอันตรายต่อทรานซิสเตอร์สวิตชิ่งได้ เมื่อทำงานในโหมด Secondary breakdown ในหัวข้อ 3.1.7 จะกล่าวถึงรายละเอียดของ Reverese-bias Secondary breakdown อีกครั้ง

ในทางปฏิบัตินั้นวงจรขับกระแสย้อนกลับควรมีความต้านทานแหล่งจ่าย (Source Impedance) ต่ำ เพื่อที่จะเพิ่ม I_{B2} และลด V_{EB} ในคาต้าชีท (Data Sheet) นั้นทางผู้ผลิตจะบอกค่าจำกัดของแรงดันไบแอสย้อนกลับระหว่างอิมิตเตอร์กับเบสได้ โดยในทางปฏิบัติจะใช้ V_{EB} ระหว่าง –2V ถึง –5V



รูปที่ 3.5 วงจรขับเบสโดยใช้หม้อแปลงขับ Q_1 และรูปสัญญาณแรงดันและกระแสที่น่าสนใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรขับเบสที่นิยมใช้แสดงในรูป 3.5 โดยที่วงจรมีหลักการทำงานดังนี้ เมื่อช่วงบวกของแรงดันพัลส์ V_S ปรากฏที่หม้อแปลงด้านทุติยภูมิทำให้เกิดกระแสเบสตรง I_{B1} ไหลเข้าที่ขาเบสของทรานซิสเตอร์ Q_1 ทำให้ทรานซิสเตอร์ Q_1 ปิดวงจร ตัวต้านทาน R_1 เป็นตัวจำกัดกระแส ค่ากระแสนี้จะเป็นตัวกำหนดอัตราการขยาย ในทางปฏิบัติแล้วอัตราการขยายจะอยู่ระหว่าง 8 และ 10

ซึ่งการคำนวณหาค่ากระแสคอลเลคเตอร์หาได้จากกำลังเอาท์พุท และชนิดของ CONVERTER ที่ใช้และกระแสเบสหาได้จากสมการ 1

แรงดันเก็บประจุตกคร่อมตัวเก็บประจุเท่ากับ

$$V_C = V_S - V_{BE} - V_D \qquad (3.2)$$

เมื่อ V_S คือ แอมปลิจูดของแรงดันหม้อแปลงขดทุติยภูมิ

V_{BE} คือ แรงดันอิ่มตัวระหว่างเบสกับอิมิตเตอร์ของ Q_1

V_D คือ แรงดันไบแอสตรงของไดโอด D

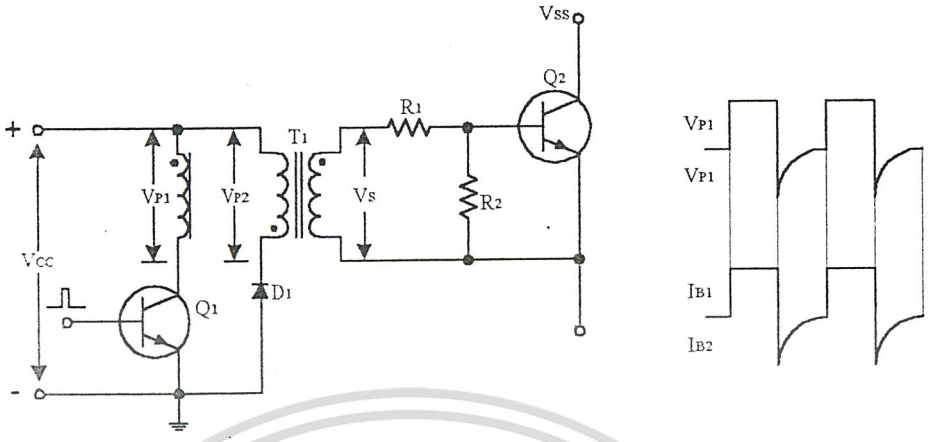
ถ้าสมมติให้ $V_{BE} = V_D = 0.7$ ดังนั้นจากสมการ 4.2 จะได้

$$V_C = V_S - 0.7 \qquad (3.3)$$

เมื่อแรงดันที่ขดปฐมภูมิเท่ากับศูนย์ ดังนั้นแรงดันที่ตกคร่อมขดทุติยภูมิ ก็ย่อมมีค่าเท่ากับศูนย์ด้วยเช่นกัน ซึ่งขณะนี้ตัวเก็บประจุ C นั้นมีการเก็บประจุจนเต็มที่ทำหน้าที่เป็นตัวไบแอสตรงกับเบสของทรานซิสเตอร์ Q_2 เป็นผลทำให้ทรานซิสเตอร์ Q_2 ปิดวงจร สังเกตว่าขณะนี้ที่ขาเบสของทรานซิสเตอร์ Q_1 จะมีศักย์แรงดันเป็นลบเกิดกระแสเบสย้อนกลับ I_{B2} (กระแสนี้เกิดจากขั้วของตัวเก็บประจุที่ต่อคร่อมระหว่างรอยต่อเบสกับอิมิตเตอร์) ทำให้ทรานซิสเตอร์ Q_1 เปิดวงจรในที่สุด โดยขนาดของกระแส I_{B2} นั้นขึ้นอยู่กับขนาดของตัวเก็บประจุ ความต้านทานของวงจร และคุณลักษณะสมบัติของทรานซิสเตอร์ Q_1 และ Q_2

อีกเทคนิคหนึ่งที่จะช่วยเพิ่มประสิทธิภาพของแหล่งจ่ายไฟตรงสวิตชิ่งแสดงดังรูป 3.6 ซึ่งจุดเด่นของวงจรนี้คือ ใช้อุปกรณ์น้อยชิ้น การทำงานของวงจรมีดังนี้ เมื่อทรานซิสเตอร์ Q_1 ปิดวงจร หม้อแปลงขับด้านปฐมภูมิจะต่อกับแรงดันของแหล่งจ่าย V_{CC} และสะสมพลังงานเกิดเป็นแรงดันพัลส์ปฐมภูมิ V_{P1} แรงดันพัลส์นี้จะถูกเหนี่ยวนำไปยังขดทุติยภูมิ (สังเกตหม้อแปลงมีขั้วของการพันขดปฐมภูมิกับขดทุติยภูมิที่เหมือนกัน) เกิดแรงดันพัลส์บวกที่ขดทุติยภูมิ V_S ขับให้ทรานซิสเตอร์ Q_2 นำกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 วงจรขับโดยใช้หม้อแปลงเป็นตัวเชื่อมต่อ โดยสร้างการขับเบสย้อนกลับจากขดลวดตติยภูมิ

ตัวต้านทาน R_1 เป็นตัวต้านทานจำกัดกระแสเบส ถูกออกแบบให้มีจุดประสงค์ เพื่อขับให้ I_{B1} มีขนาดพอที่จะทำให้ทรานซิสเตอร์ Q_2 ปิดวงจรโดยปราศจากการโอเวอร์ไดรว์ (Overdrive) และการอิ่มตัวจนเกินไป

สำหรับตัวต้านทาน R_2 เป็นตัวกำหนดกระแสรั่วไหลสำหรับรอยต่อระหว่างเบสกับอิมิตเตอร์ของทรานซิสเตอร์ Q_2 โดยในทางปฏิบัติจะเป็นตัวต้านทานค่าต่ำ ๆ ประมาณ 50 Ω ถึง 100 Ω

เมื่อทรานซิสเตอร์ Q_1 เปิดวงจร พลังงานที่เก็บสะสมในหม้อแปลงจะกลับมาเป็น V_{CC} โดยเกิดจากขดตติยภูมิ (Tertiary) และไดโอด D_1 (สังเกตว่าขั้วการพันของขดลวดตติยภูมินี้จะตรงกันข้ามกับขดปฐมภูมิ) ทำให้แรงดันพัลส์ V_{P2} ที่ได้เกิดการเปลี่ยนแปลงขั้ว และเหนี่ยวนำเป็นพัลส์ลบปรากฏยังขดทุติยภูมิเป็นผลทำให้เกิดกระแสย้อนกลับ I_{B2} ตามต้องการเพื่อให้ทรานซิสเตอร์ Q_2 เปิดวงจรหยุดนำกระแสนั่นเอง

หัวใจของวงจรนี้ขึ้นอยู่กับการออกแบบหม้อแปลง สัดส่วนการพันระหว่างขดปฐมภูมิกับขดทุติยภูมิจะต้องเลือกไม่ให้แรงดันที่ได้มากเกินกว่าแรงดันของ V_{BE} และ V_{EB} โดยปกติแล้วขดปฐมภูมิและขดตติยภูมิจะมีจำนวนรอบที่เท่ากัน

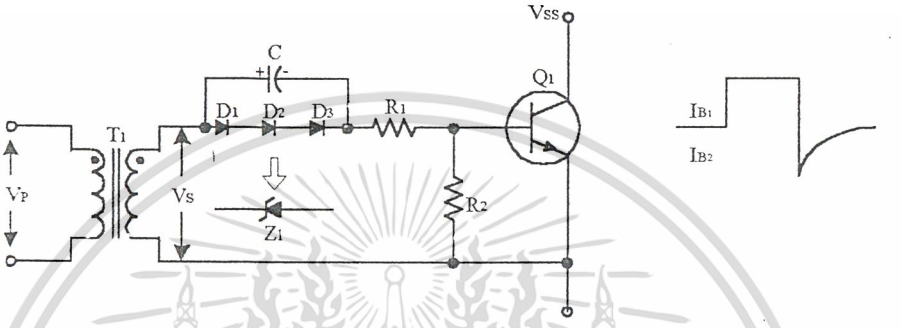
ในทางปฏิบัติแล้วในการพันขดลวดระหว่างขดปฐมภูมิกับขดตติยภูมิจะต้องพันให้แน่นเพื่อหลีกเลี่ยงแรงดันสไปรค์ (Spike) ที่เกิดจากค่าความเหนี่ยวนำรั่วไหล ทรานซิสเตอร์ Q_1 จะต้องให้แรงดันคอลเลคเตอร์ขณะเปิดวงจรมีค่าอย่างน้อยที่สุด 2 V_{CC}

สำหรับวงจรในรูป 3.7 เป็นการพัฒนาจากวงจรที่ผ่านมามีการทำงานดังนี้คือ เมื่อแรงดันพัลส์ V_P เป็นบวกก็จะเหนี่ยวนำให้เกิดเป็นพัลส์ V_S ขับให้ทรานซิสเตอร์ Q_1 ปิดวงจรเกิดการเก็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประจุที่ C โดยมีขั้วแรงดันดังรูป 3.7 แรงดันที่ตัวเก็บประจุขณะนี้จะมีค่าประมาณ 2.1 V. โดยเกิดจากแรงดันตกคร่อมไดโอด D_1, D_2และ D_3 ซึ่งไดโอดทั้งสามตัวนี้ทำหน้าที่เหมือนซีเนอร์ไดโอด

เมื่อแรงดันขดปฐมภูมิมีค่าเป็นศูนย์ แรงดันที่ขดทุติยภูมิจึงมีค่าเท่ากับศูนย์ด้วย ขณะนี้แรงดันประจุที่สะสมที่ตัวเก็บประจุ C ที่ต่อคร่อมระหว่างรอยต่อเบสกับอิมิตเตอร์มีค่ามากพอที่จะทำให้เกิดกระแสเบสย้อนกลับ I_{B2} ทำให้ทรานซิสเตอร์ Q_1 เปิดวงจร เพื่อลดเวลาละสมที่เกิดขึ้น



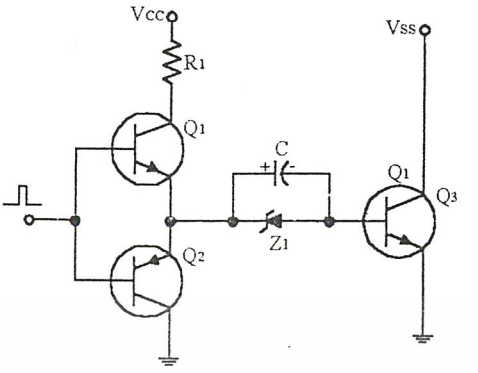
รูปที่ 3.7 แสดงวงจรขับเบสโดยใช้หม้อแปลงเป็นตัวแยกระบบทางไฟฟ้า และสร้างพัลส์ขับให้ทรานซิสเตอร์ปิดวงจร โดยใช้ตัวเก็บประจุสร้างแรงดันลบทำให้ทรานซิสเตอร์เปิดวงจร

จากวงจรในรูป 3.7 ทำให้พบว่าสามารถนำไปขับเบสของทรานซิสเตอร์กำลังได้โดยตรงดังรูป 3.8 ซึ่งทรานซิสเตอร์ Q_1 กับ Q_2 นั้นจะทำการสวิตช์ตรงกันข้ามเสลับกัน สร้างแรงดัน V_{CC} กับกราวนด์ให้กับเบสของทรานซิสเตอร์ Q_3 เป็นผลทำให้ทรานซิสเตอร์ Q_3 เปิดวงจร

เมื่อทรานซิสเตอร์ Q_1 ปิดวงจรทำให้ Q_3 ปิดวงจรด้วย ขณะเดียวกันนั้นตัวเก็บประจุ C ก็จะเก็บประจุโดยมีซีเนอร์ไดโอด Z_1 เป็นตัวจำกัดแรงดันประจุของ C ซึ่งในทางปฏิบัติแล้วจะใช้แรงดัน V_Z เท่ากับ 4.3 V. และตัวต้านทาน R_1 จะเป็นตัวกำหนดกระแสเบสตรง I_{B1}

จนกระทั่งทรานซิสเตอร์ Q_1 เปิดวงจร และ Q_2 ปิดวงจร แรงดันประจุของตัวเก็บประจุ C ที่ต่อคร่อมระหว่างเบสกับอิมิตเตอร์ของทรานซิสเตอร์ Q_3 นั้นมีค่าเพียงพอที่จะทำให้เกิดกระแสเบสย้อนกลับ I_{B2} โดยที่ขนาดของกระแส I_{B2} นี้จะขึ้นอยู่กับอัตราการขยาย (gain) ของทรานซิสเตอร์ Q_2, ประจุของตัวเก็บประจุ C และความต้านทานของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

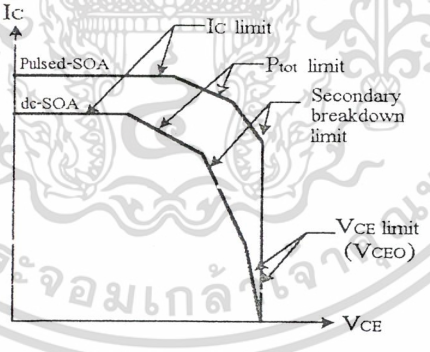


รูปที่ 3.8 วงจรขับเบสโดยใช้ตัวเก็บประจุเป็นตัวเชื่อมต่อโดยตรง

3.1.7 ปรากฏการณ์ Secondary Breakdown ของทรานซิสเตอร์

(1.) Secondary Breakdown เมื่อป้อนไบแอสตรง

โดยปกติแล้วทางโรงงานหรือผู้ผลิตนั้นจะกำหนดกราฟในลักษณะของพื้นที่ทำงานที่ปลอดภัย (Safe Operating Area, SOA) ดังแสดงในกราฟความสัมพันธ์ระหว่างกระแส I_C กับแรงดัน V_{CE} ดังรูป 3.9



รูปที่ 3.9 แสดง SOA ของทรานซิสเตอร์

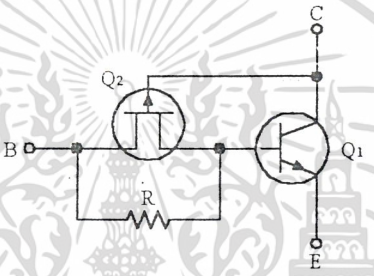
ซึ่งพื้นที่ในกราฟนั้นจะบอกให้ผู้ออกแบบทราบว่าขอบเขตในการทำงานที่ปลอดภัยของทรานซิสเตอร์มีข้อจำกัดสูงสุดในย่านใดบ้าง กราฟนี้จะใช้พิจารณาเมื่อเส้น load line ขณะได้รับไบแอสตรงอยู่ในพื้นที่ SOA จึงจะปลอดภัย

ปรากฏการณ์ Secondary Breakdown เมื่อป้อนไบแอสตรง จะเกิดเมื่อผู้ออกแบบกำหนดจุดการทำงานของทรานซิสเตอร์เกินนอกเหนือจากขอบเขตพื้นที่ SOA ทำให้ความร้อนที่ตัวทรานซิสเตอร์กำลังสูงขึ้น สัมประสิทธิ์อุณหภูมิ (Temperature coefficient) ของรอยต่อระหว่างเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับอิมิตเตอร์เป็นลบ ความร้อนจะเพิ่มขึ้น เมื่อมีกระแสไหลเพิ่มขึ้นก่อให้เกิดกำลังสูญเสียสูงขึ้นตามไปด้วยจนกระทั่งค่าสัมประสิทธิ์อุณหภูมิของแรงดันเบรคดาวน์ระหว่างคอลเลคเตอร์กับอิมิตเตอร์เป็นลบด้วย ถ้าปรากฏการณ์ดังกล่าวนี้ยังไม่มีการแก้ไขจะทำให้รอยต่อระหว่างคอลเลคเตอร์กับอิมิตเตอร์พังทลายส่งผลให้ทรานซิสเตอร์เสียหายได้ เนื่องจากความร้อนที่เพิ่มขึ้นอย่างไม่มีขีดจำกัด

จากผลเสียที่ได้ของปรากฏการณ์ Secondary breakdown เมื่อป้อนไบแอสตรงดังที่กล่าวมาแล้ว จึงทำให้มีผู้คิดค้นวิธีจะหลีกเลี่ยงมิให้เกิดขึ้น เช่น โรงงาน National Semiconductor ได้ปรับปรุงขั้นตอนการผลิตทรานซิสเตอร์ให้สามารถทำงานที่ระดับกำลังงานและแรงดันคอลเลคเตอร์สูงสุดโดยไม่เกิดความเสียหาย ในรูปที่ 3.10 แสดงวงจรภายในของทรานซิสเตอร์ที่ถูกพัฒนาขึ้น



รูปที่ 3.10 การป้องกัน Secondary breakdown โดยการต่อ JFET อนุกรมกับเบสของทรานซิสเตอร์

เทคนิคในการแก้ไขนี้จะเพิ่ม JFET อนุกรมกับเบสของทรานซิสเตอร์กำลัง โดยที่ JFET จะทำหน้าที่เหมือนกับ ballast resister ซึ่งค่าความต้านทานนี้จะเปลี่ยนแปลงตามแรงดันระหว่างคอลเลคเตอร์กับเบส ทำให้กำลังสูญเสียที่เกิดขึ้นคงที่ อุณหภูมิที่ตัวทรานซิสเตอร์จึงไม่สูงขึ้น

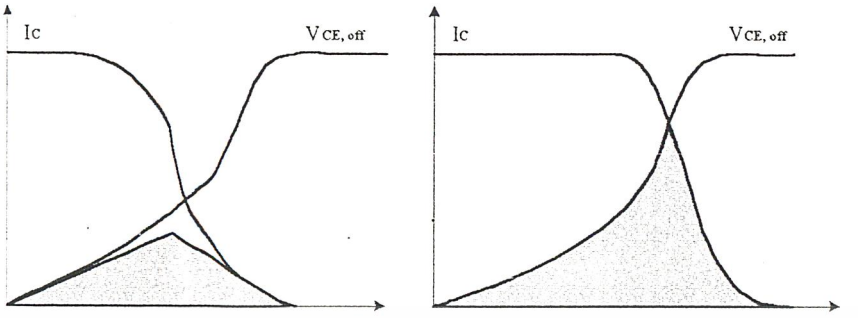
(2.) Secondary breakdown เมื่อป้อนไบแอสกลับ

จากที่ได้กล่าวมาแล้ว ในเรื่องช่วงเวลาสะสม และค่าสูญเสียเมื่อนำทรานซิสเตอร์กำลังไปใช้ในงานสวิตชิ่งนั้น ถ้าหากช่วงเวลาสะสมมีค่ามากเกินไปจะทำให้หม้อแปลงเกิดการอิ่มตัว เป็นผลให้ย่านการเรกกูเลตถูกจำกัดให้แคบลง

สำหรับค่าสูญเสียในการสวิตชิ่งย่อมจะส่งผลโดยตรงกับประสิทธิภาพของแหล่งจ่าย ทำให้ผู้ออกแบบที่ต้องการแหล่งจ่ายประสิทธิภาพสูง ต้องคำนึงถึงค่าพารามิเตอร์ทั้งสองด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



(ก) (ข)

รูปที่ 3.11 แสดงกราฟความสัมพันธ์ของ I_C และ V_{CE} ของทรานซิสเตอร์
ก) เมื่อโหลดแบบความต้านทาน
ข) เมื่อโหลดแบบความเหนี่ยวนำ

ในรูปที่ 3.11 แสดงลักษณะสมบัติของทรานซิสเตอร์กำลังที่แรงดันสูงขณะเปิดวงจรเมื่อใช้กับโหลดความต้านทาน และโหลดความเหนี่ยวนำ สังเกตพบว่าเมื่อโหลดที่ใช้เป็นโหลดความเหนี่ยวนำจะทำให้เกิดพลังงาน (Energy) สูงกว่า เมื่อเทียบกับโหลดความต้านทาน (พิจารณาพื้นที่แรเงา แสดงถึงพลังงานที่เกิดขึ้นนั่นเอง) ซึ่งพลังงานที่เกิดขึ้นนี้จะเป็นสาเหตุทำให้เกิดปรากฏการณ์ Secondary breakdown ได้

กำลังงานขณะเกิด Secondary breakdvn เมื่อป้อนไบแอสกลับ (E_{SB}) สามารถคำนวณได้จากสมการ

$$E_{SB} = \frac{1}{2} L_{eff} I_c^2 \qquad (3.4)$$

เมื่อ
$$L_{eff} = \frac{V_{CEX}}{V_{CEX} - V_{CC}} \qquad (3.5)$$

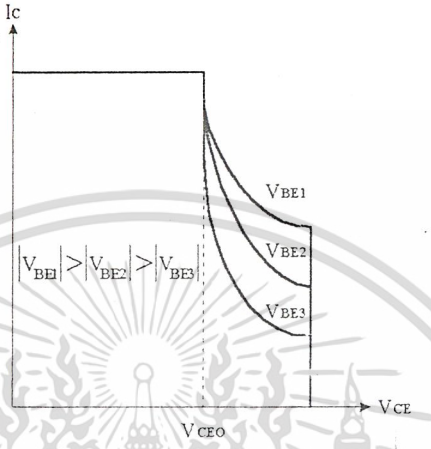
โดยที่ E_{SB} มีหน่วยเป็นจูล (Joule) ซึ่งในการทดสอบทรานซิสเตอร์จะปลดขาเบสออกหรือเพิ่มความต้านทานขาเบสให้สูง ๆ และกำหนดให้ทรานซิสเตอร์ทำงานใกล้กับแรงดัน break-down, V_{CEX} เพื่อหาความสัมพันธ์ของพารามิเตอร์ E_{SB} (E_{SB} จะมีย่านกว้างมากระหว่างมิลลิจูลถึงจูล)

พื้นที่ทำงานที่ปลอดภัยเมื่อป้อนไบแอสกลับ (Reverse-Bias Safe Operating Area RBSOA) แสดงในรูป 3.12 คล้ายกับที่ได้กล่าวมาแล้วในกรณีของกราฟ SOA สำหรับกราฟของ RBSOA แสดงเมื่อแรงดัน V_{CE} ต่ำกว่าแรงดัน V_{CEO} เป็นพื้นที่ที่ปลอดภัยจากแรงดันไบแอสกลับ V_{EB} แต่ก็ยังถูกจำกัดด้วยอัตราทนกระแสคอลเลคเตอร์ I_C ของตัวทรานซิสเตอร์เอง เมื่อใดที่แรงดัน V_{CE} มีค่าสูงเกินกว่าแรงดัน V_{CEO} พบว่ากระแสคอลเลคเตอร์จะลดลงอย่างสัมพันธ์กับแรงดันไบแอสกลับที่ป้อนเข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

50364

ดังนั้นจึงเป็นเหตุผลว่าแรงดันไบแอสกลับ V_{EB} ที่มีค่าสูง ๆ จะเป็นสาเหตุให้เกิดการพังทลายของรอยต่อระหว่างเบสกับอิมิตเตอร์ขณะเปิดวงจร ผู้ออกแบบมักจะใช้ไดโอด หรือวงจรสนับเบอร์ (snubber) เป็นตัวป้องกันปรากฏการณ์ดังกล่าว

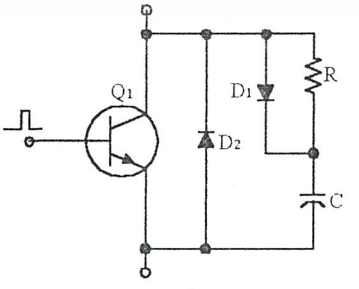


รูปที่ 3.12 พื้นที่ทำงานที่ปลอดภัยเมื่อป้อนไบแอสกลับ RBSOA

ย้อนกลับมาพิจารณาวงจรขับเบสอีกครั้ง โดยปกติแล้วกระแสเบสย้อนกลับ I_{B2} นั้นถ้ายิ่งมีค่ามากเท่าใด ก็จะทำให้ช่วงเวลาสะสมน้อยลงเท่านั้น แต่การที่ทำให้เกิดกระแสย้อนกลับมีค่ามาก ๆ นั้น ย่อมส่งผลอันตรายโดยตรงกับทรานซิสเตอร์ โดยเฉพาะการพังทลายของรอยต่อระหว่างเบสกับอิมิตเตอร์ได้ ในทางปฏิบัติแล้วมีวิธีป้องกันมิให้เกิดปรากฏการณ์เช่นนี้ได้โดย

1) ใช้ค่าแรงดัน V_{CE} ต่ำ ๆ ขณะที่ทรานซิสเตอร์เปิดวงจร

2) ลดกระแสคอลเลคเตอร์ ขณะที่แรงดันคอลเลคเตอร์เพิ่มขึ้น โดยทั่วไปแล้วมักนิยมใช้วิธีที่ 2 เป็นวิธีป้องกันมากกว่าเพราะสามารถออกแบบวงจรป้องกันได้โดยไม่ยุ่งยากนัก ซึ่งวงจรในการแก้ปัญหานั้นก็คือวงจรสนับเบอร์ ดังแสดงในรูป 3.13



รูปที่ 3.13 วงจรสนับเบอร์ต่อคร่อมทรานซิสเตอร์สวิตชิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.13 แสดงการใช้วงจร RC สนับเบอร์เพื่อป้องกันทรานซิสเตอร์สวิตชิ่งเสียหาย เมื่อเปิดวงจร การทำงานมีดังนี้ เมื่อทรานซิสเตอร์ Q_1 เปิดวงจรตัวเก็บประจุ C จะเก็บประจุผ่านไดโอด D_1 ค่าแรงดันที่ตัวเก็บประจุ C เท่ากับ $V_{CC} - V_D$

เมื่อทรานซิสเตอร์ Q_1 ปิดวงจรตัวเก็บประจุ C จะคายประจุผ่านตัวต้านทาน R ทำให้พลังงานสูญเสียในรูปของความร้อนจึงเกิดขึ้นที่วงจรสนับเบอร์แทนทรานซิสเตอร์สวิตชิ่ง ทำให้ทรานซิสเตอร์ปลอดภัยจากความเสียหายได้

ในการออกแบบวงจรสนับเบอร์ทางปฏิบัตินั้นจะใช้รูป 3.11 (ข) เป็นรูปอ้างอิงสมการของพลังงานที่เกิดขึ้นระหว่างที่ทรานซิสเตอร์เปิดวงจรสามารถเขียนได้ดังนี้

$$E = \frac{CV_{CE}^2}{2} = \frac{I_C V_{CE}(t_r + t_f)}{V_{CE}} \qquad (3.6)$$

เมื่อ I_C คือ กระแสคอลเลคเตอร์ I_C สูงสุด, A

V_{CE} คือ แรงดันระหว่างคอลเลคเตอร์กับอิมิตเตอร์สูงสุด, V

t_r คือ เวลาขอบขาขึ้น, μS

t_f คือ เวลาขอบขาลง, μS

แก้สมการ 4.7 เพื่อหาค่าตัวเก็บประจุ C จะได้

$$C = \frac{I_C(t_r + t_f)}{V_{CE}} \qquad (3.7)$$

เนื่องจากตัวเก็บประจุ C จะเก็บประจุ ขณะที่ทรานซิสเตอร์เปิดวงจร และคายประจุผ่านตัวต้านทาน R ขณะช่วงเวลาทรานซิสเตอร์ปิดวงจร (t_{on}) แรงดันตกคร่อมตัวเก็บประจุเขียนได้ดังนี้

$$V = V_{CE}.\exp(-\frac{t_{on}}{RC}) \qquad (3.8)$$

ในทฤษฎีจะถือว่าเมื่อตัวเก็บประจุได้เก็บประจุจนเต็มที่นั้นจะมีแรงดันใกล้เคียงกับแรงดัน V_{CE} ดังนั้นจึงต้องเลือกค่า $-(\frac{t_{on}}{RC})$ มีค่าเข้าใกล้ 1 ซึ่งมักจะเลือกค่า RC โดยให้ตัวเก็บประจุสามารถคายประจุ ณ ช่วงเวลาสิ้นสุดของการปิดวงจร (t_{on})

จากในการแก้ปัญหาวงจรไฟฟ้าเบื้องต้น เราทราบว่าสามารถแทนช่วงเวลาคงที่ 5τ สำหรับตัวเก็บประจุใช้ในการคายประจุจนหมดผ่านตัวต้านทานโดย $\tau = RC$ สมมติว่าในกรณีนี้ตัวเก็บประจุจะเริ่มคายประจุ เมื่อสิ้นสุด 3τ

ดังนั้นค่าความต้านทานสูงสุดที่ใช้ในการคายประจุหาได้จาก

$$R = \frac{t_{on}}{3C} \qquad (3.9)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าความต้านทาน R ที่คำนวณได้จากสมการ 9 สามารถนำมาหากระแสขณะคายประจุผ่านทรานซิสเตอร์ขณะปิดวงจร โดยจำกัดค่ามิให้เกิน 25 เปอร์เซ็นต์ของกระแสคอลเลคเตอร์ 0.25 I_{cc} โดยหาจากสมการ

$$I_{dis} = \frac{V_{CE}}{R} \qquad (3.10)$$

ถ้าค่าความต้านทานที่คำนวณได้มีค่าต่ำเกินไป ทำให้กระแส I_{dis} มีค่าสูงกว่า 0.25 I_C ดังนั้นค่าความต้านทาน R จะต้องเลือกอีกครั้ง เพื่อให้เหมาะสมตรงกับเงื่อนไข

ขั้นตอนต่อไปเป็นการคำนวณหากำลังไฟฟ้าของตัวต้านทาน

$$P_R = \frac{1}{2} C V_{CE}^2 f \qquad (3.11)$$

เมื่อ f คือ ความถี่ที่ใช้งานของ CONVERTER, kHz

3.2 วงจรควบคุมแรงดันไฟฟ้าดีซี

เนื่องจากสวิตช์ที่ใช้ทรานซิสเตอร์ สามารถทำการเปิดกระแสและทำการปิดกระแสได้ทั้งสองอย่าง จึงสามารถนำไปใช้ควบคุมแรงดันไฟฟ้าดีซีได้ด้วยการปิดเปิดสวิตช์ซ้ำ ๆ กันเป็นจังหวะที่เหมาะสม แรงเคลื่อนไฟฟ้าดีซีที่ส่งไปยังภาระจะขาดออกเป็นห้วง ๆ ดังแสดงในรูปที่ 3.14 ซึ่งเราสามารถควบคุมค่าเฉลี่ยของแรงดันเลี้ยงวงจรให้สูงหรือต่ำลงด้วยการควบคุมเวลาเปิด (= t_{on}), เวลาเปิด (= t_{off}) หรือคาบของเวลาปิดเปิด ($T = t_{on} + t_{off}$) ให้แปรค่าต่าง ๆ เป็นจังหวะซึ่งสามารถเขียนเป็นสมการได้ดังนี้

$$V(t) = \begin{cases} E & \text{เมื่อ } t = t_{on} \\ 0 & \text{เมื่อ } t = t_{off} \end{cases} \qquad (3.12)$$

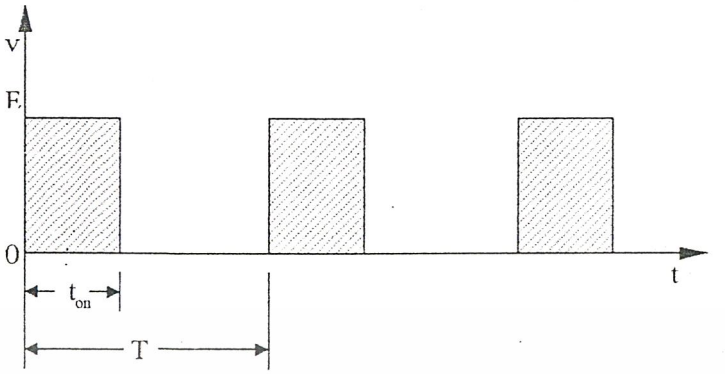
เมื่อกระจายออกเป็นอนุกรม Fourier เราจะได้ค่าสัมประสิทธิ์

$$a_o = \frac{1}{T/2}\int_0^T E\,dt = 2\frac{t_{on}}{T}E \qquad (3.13)$$

$$a_n = \frac{1}{T/2}\int_0^T E\cos\left(\frac{n\pi t}{T/2}\right)dt = \frac{2E}{T}\left[\int_0^{t_{on}} \cos\frac{2n\pi t}{T}\,dt + \int_{on}^{T}(0).\cos\frac{2n\pi t}{T}\,dt\right]$$

$$= \frac{E}{n\pi}\sin\frac{2n\pi t_{on}}{T}, \quad n = 1, 2, 3, \ldots \qquad (3.14)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 รูปคลื่นแรงดันที่ได้จากการปิดเปิดไฟฟ้าดีซีเป็นจังหวะหรือของวงจรสับไฟฟ้า

และ

$$b_n = \frac{1}{T/2}\int_0^T E\sin\left(\frac{n\pi t}{T/2}\right)dt = \frac{2E}{T}\left[\int_0^{t_{on}} \sin\frac{2n\pi t}{T}\,dt + \int_{on}^{T}(0).\sin\frac{2n\pi t}{T}\,dt\right]$$

$$= \frac{E}{n\pi}\left(1-\cos\frac{2n\pi t_{on}}{T}\right),\ n = 1, 2, 3,\ldots \qquad (3.15)$$

ทำให้เราได้อนุกรม Fourier ดังนี้

$$V(t) = E\left[\frac{t_{on}}{T} + \frac{1}{\pi}\sum_{n=1}^{\infty}\frac{1}{n}\left\{\left(\sin\frac{2n\pi t_{on}}{T}\right)\left(\cos\frac{2n\pi t}{T}\right) + \left(1-\cos\frac{2n\pi t_{on}}{T}\right)\left(\sin\frac{2n\pi t}{T}\right)\right\}\right] \qquad (3.16)$$

ซึ่งเราสามารถคำนวณหาค่าแรงดันเฉลี่ยได้ดังนี้

$$V_{VAG} = \frac{t_{on}}{T}E = \frac{t_{on}}{t_{on}+t_{off}}E \qquad (3.17)$$

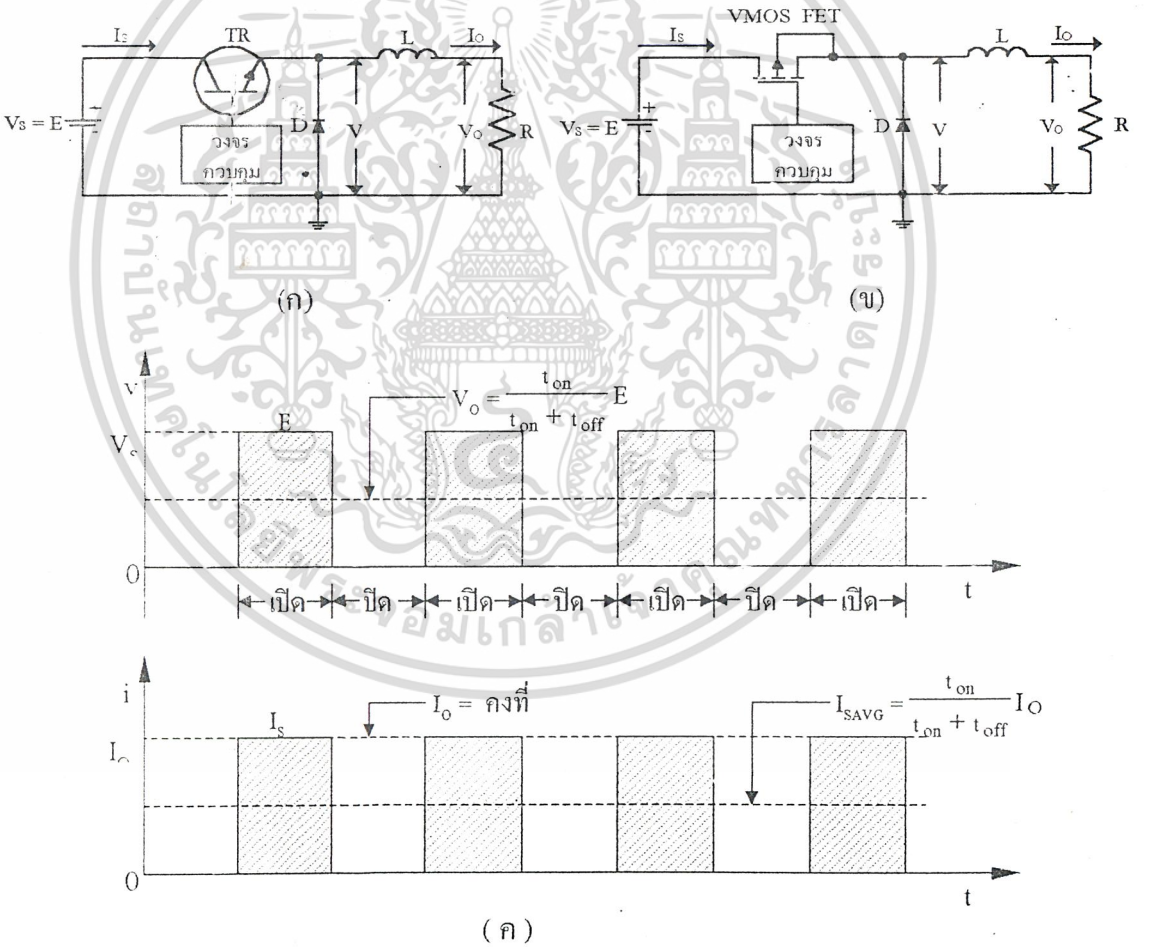
วงจรที่จะสามารถให้แรงดันขาดตอนเป็นห้วง ๆ ที่สามารถเปลี่ยนค่าแรงดันเฉลี่ยได้ดังรูปคลื่นที่แสดงในรูปที่ 3.15 นี้ เป็นวงจรที่เรียกว่า " วงจรสับไฟฟ้า " (chopper) ซึ่งทำขึ้นได้หลายแบบ วงจรแบบที่นิยมใช้ในปัจจุบันเป็นวงจรที่ใช้ไธริสเตอร์เป็นส่วนประกอบสำคัญ แต่ในหัวข้อนี้จะกล่าวถึงวงจรแบบที่ใช้ทรานซิสเตอร์กำลังแบบไบโพลาร์และเฟทกำลังชนิดวีมอสเป็นส่วนประกอบสำคัญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 ชนิดของวงจรสับไฟฟ้า แบ่งเป็น 2 ชนิด คือ

1. วงจรสับไฟฟ้าลดค่าลง (step-down chopper)

โดยการนำเอาทรานซิสเตอร์กำลังแบบไบโพลาร์หรือเฟทกำลังชนิดวีมอสเข้าต่อในวงจรเป็นอนุกรมคั่นกลางระหว่างแหล่งจ่ายไฟฟ้ากับภาระดังกล่าวในรูปที่ 3.15 (ก) และ (ข) ตามลำดับ ซึ่งเมื่อทรานซิสเตอร์กำลังหรือเฟทกำลังทำงานปิดเปิดกระแสจะทำให้ได้แรงดันและกระแสมีรูปคลื่นดังแสดงในรูปที่ 3.15 (ค) การปรับเปลี่ยนแรงดันเฉลี่ยทำได้ด้วยการควบคุมวงจรเบสของทรานซิสเตอร์กำลังหรือวงจรจุดชนวนเกตของเฟทกำลังให้ทำงานปิดเปิดวงจรเป็นจังหวะโดยการแปรเวลาปิด เวลาเปิด หรือทั้งคู่ แรงดันเฉลี่ยที่ได้รับจะมีค่าต่ำกว่าค่าแรงดันเลี้ยงวงจรเสมอ จึงเรียกวงจรนี้ว่า "วงจรสับไฟฟ้าลดค่าลง"



รูปที่ 3.15 (ก) วงจรสับไฟฟ้าลดค่าลงโดยใช้ทรานซิสเตอร์กำลังแบบไบโพลาร์

(ข) วงจรสับไฟฟ้าลดค่าลงโดยใช้เฟทกำลังชนิดวีมอส

(ค) รูปคลื่นแรงดันและกระแสของ (ก) และ (ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 3.15 (ก) และ (ข) ตัวเหนี่ยวนำ L ทำหน้าที่จัดรูปคลื่นแรงเอาท์พุทให้ราบเรียบลง ฟรีวีลลิงไดโอดD ทำหน้าที่เป็นเส้นทางให้กระแสภาระไหลในระหว่างที่ทรานซิสเตอร์กำลังแบบไบโพลาร์หรือเฟทกำลังชนิดวีมอสปิดกระแส ถึงแม้ว่า วงจรนี้จะทำให้แรงดันเฉลี่ยลดค่าลง แต่แหล่งจ่ายไฟฟ้าจะจ่ายกระแส I_s ที่มียอดสูงสุดที่ค่าเท่ากับกระแสภาระ I_o ดังนั้นกระแสเฉลี่ยจากแหล่งจ่ายไฟฟ้า I_{SAVG} จะมีค่าน้อยกว่ากระแสภาระซึ่งมีความสัมพันธ์ดังสมการที่ (3.18)

$$I_{SAVG} = \frac{t_{on}}{t_{on} + t_{off}} I_o \qquad (3.18)$$

โดยที่กระแสภาระ I_o มีค่าคงที่ เมื่อเราแปรเวลาปิด เวลาเปิด หรือทั้งคู่ จะทำให้เราสามารถปรับค่ากระแสเฉลี่ยให้สูงขึ้นขณะที่แรงดันเฉลี่ยลดค่าลง

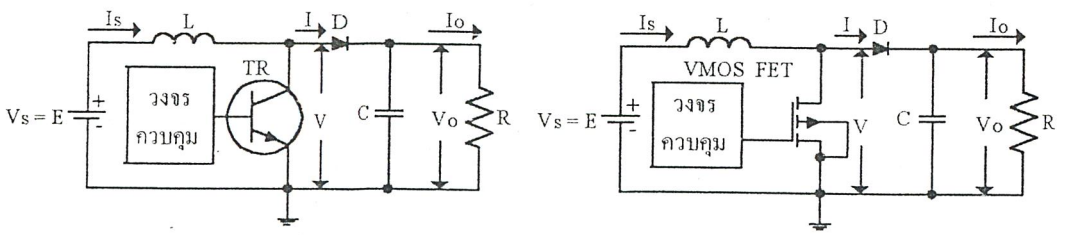
2. วงจรสับไฟฟ้ายกค่าขึ้น (step-up chopper)

โดยการเปลี่ยนแปลงวงจรในรูปที่ 3.15 (ก) และ (ข) ด้วยการนำเอาตัวเหนี่ยวนำ L มาต่อติดกับแหล่งจ่ายไฟฟ้าแบบอนุกรม และนำเอาทรานซิสเตอร์กำลังแบบไบโพลาร์หรือเฟทกำลังชนิดวีมอสซึ่งทำหน้าที่เป็นสวิตช์มาต่อลงในวงจรแบบขนานดังแสดงในรูปที่ 3.16 (ก) และ (ข) ตามลำดับเมื่อทรานซิสเตอร์กำลังแบบไบโพลาร์หรือเฟทกำลังชนิดวีมอสนี้ทำหน้าที่ปิดเปิดกระแสเป็นจังหวะจะให้รูปคลื่นกระแสและแรงดันดังแสดงในรูปที่ 3.16 (ค) โดยขณะที่สวิตช์เปิดกระแส ตัวเหนี่ยวนำ L จะกักเก็บพลังงานไว้ในรูปสนามแม่เหล็ก ครั้นเมื่อสวิตช์ปิดกระแส ตัวเหนี่ยวนำ L จะปล่อยพลังงานที่กักเก็บไว้ออกเลี้ยงทำให้เกิดมีแรงดันขึ้นบวกเพิ่มเข้ากับแรงดันเลี้ยงวงจรป้อนเข้าสู่ภาระผ่านไดโอด โดยที่กระแส I_s จากแหล่งจ่ายไฟฟ้ามีค่าคงที่ และพลังงานมีค่าคงที่ขณะที่สวิตช์ปิดกระแส ดังนั้นเราสามารถเขียนสมการได้ดังข้างล่าง

$$V_o t_{off} = E(t_{on} + t_{off}) \qquad (3.19)$$

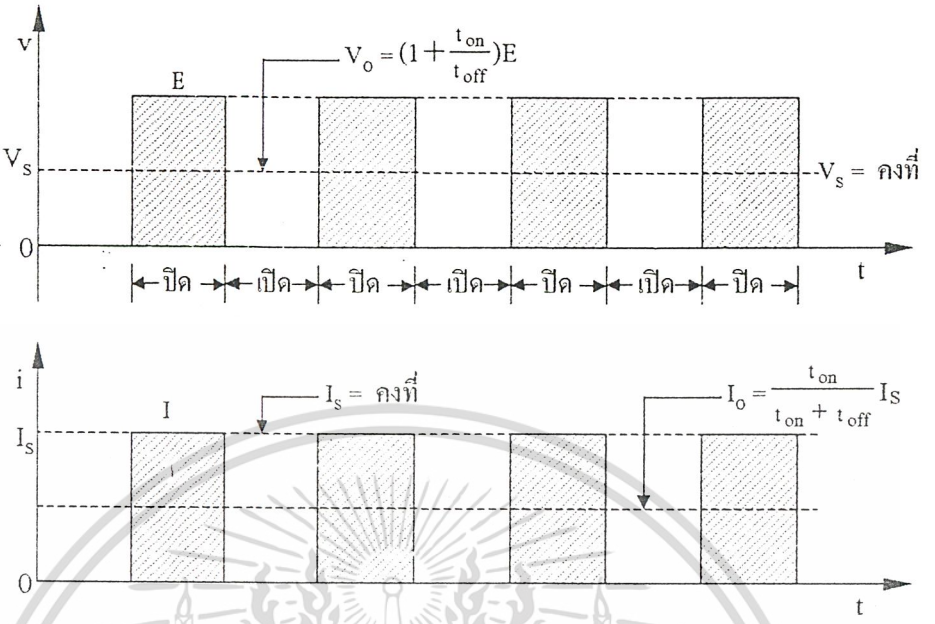
เพราะฉะนั้นทำให้เราได้ความสัมพันธ์

$$V_o = (1 + \frac{t_{on}}{t_{off}}) E \qquad (3.20)$$



(ก) (ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ค)

รูปที่ 3.16 (ก) วงจรสับไฟฟ้ายกค่าขึ้นโดยใช้ทรานซิสเตอร์กำลังแบบไบโพลาร์

(ข) วงจรสับไฟฟ้ายกค่าขึ้นโดยใช้เฟทกำลังชนิดวีมอส

(ค) รูปคลื่นกระแสและแรงดันของ (ก) และ (ข)

ซึ่งแสดงว่า เราอาจปรับค่าแรงดันเอาท์พุทเฉลี่ยให้สูงขึ้นได้ แต่กระแสภาระเฉลี่ยมีค่าลดลง ตามสมการข้างล่างนี้

$$I_O = \frac{t_{on}}{t_{on} + t_{off}} I_S \qquad (3.21)$$

สังเกตว่า L ทำหน้าที่เฉลี่ยค่าแรงดัน ขณะที่ C ทำหน้าที่เฉลี่ยกระแส ถ้าหากว่า L และ C มีค่าใหญ่ทั้งคู่จะทำให้รูปคลื่นที่ได้รับมีลักษณะราบเรียบอย่างสมบูรณ์ เราอาจลดขนาดของ L และ C ให้เล็กลงได้โดยการเพิ่มความถี่ในการปฏิบัติงานของวงจรสับไฟฟ้าให้มีค่าสูงขึ้น แต่วิธีการนี้จะทำให้เกิดมีกำลังสูญเพิ่มขึ้นในการปิดเปิดของสวิตช์เพราะฉะนั้นเราต้องออกแบบวงจรให้ใช้ความถี่สูงพอประมาณที่ไม่ให้กำลังสูญเกิดขึ้นมากเกินไป

โดยคำอธิบายข้างบนนี้ วงจรสับไฟฟ้าจะทำหน้าที่คล้าย ๆ กับเป็นหม้อแปลงไฟฟ้ากระแสตรงที่สามารถลดแรงดันลงหรือยกแรงดันให้สูงขึ้น ด้วยการปรับเปลี่ยน เวลาปิด เวลาเปิด หรือทั้งคู่ ซึ่งเป็นการปรับเปลี่ยนที่ง่ายกว่าการเปลี่ยนจำนวนรอบ (turn ratio) ของหม้อแปลงไฟฟ้าทั่ว ๆ ไป เรานิยมใช้วงจรสับไฟฟ้านี้ในการควบคุมอัตราเร็วของมอเตอร์ดีซีที่ใช้เป็นต้นกำลังในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขับเคลื่อนของยานพาหนะขนาดเล็ก เช่น รถยนต์ไฟฟ้า รถไฟฟ้าที่ใช้ในการเล่นกอล์ฟ รถยกของแบบส้อม เป็นต้น

3.3 โฟโตคัปเปลอร์

โฟโตคัปเปลอร์ (photocoupler) คือสิ่งประดิษฐ์ออปโตอิเล็กทรอนิกส์ชนิดหนึ่งซึ่งประกอบด้วยภาคเปล่งแสง (light emitting source) และภาครับแสง (light detector) อยู่ในชิปเดียวกัน แสงที่เปล่งออกมาจากภาคเปล่งแสงจะวิ่งเข้าสู่ภาครับแสง ระหว่างภาคเปล่งแสงและภาครับแสงจะเปิดช่องออกเพื่อให้วัตถุจากภายนอกสามารถวิ่งตัดแสงหรือไม่ก็ได้ ถ้าเราแบ่งโฟโตคัปเปลอร์ออกตามวิธีการตัดแสง จะสามารถแบ่งออกเป็นได้ 2 ชนิด คือ โฟโตไอโซเลเตอร์ (photoisolator) และโฟโตอินเทอร์รัปเตอร์ (photointerupter) ในที่นี้จะกล่าวถึงเฉพาะส่วนที่เกี่ยวข้องกับภาคขับมอเตอร์เท่านั้นคือ โฟโตไอโซเลเตอร์

3.3.1 โฟโตไอโซเลเตอร์ (Photoisolator)

โฟโตไอโซเลเตอร์เป็นโฟโตคัปเปลอร์ชนิดหนึ่ง ลักษณะโครงสร้างมีภาคเปล่งแสงอยู่ในภาชนะที่ปิดมิดชิด โดยไม่มีการอนุญาตให้แสงจากภายนอกทะลุเข้าหรือแสงจากภายในทะลุออกนอกภาชนะได้เลย ดังนั้นจึงไม่ต้องเป็นห่วงเรื่องแสงรบกวนจากภายนอก โดยความนิยมแล้ว โฟโต้คัปเปลอร์มักจะหมายถึงชนิดโฟโตไอโซเลเตอร์นี้

ขาสำหรับป้อนไฟฟ้าเข้าภาคเปล่งแสง จะถูกออกแบบให้แยกออกจากกันจากก้านขาของด้านเอาท์พุตด้วยฉนวนไฟฟ้าอย่างดี การขับภาคเปล่งแสงสามารถทำได้ทั้งวิธีกระแสไฟฟ้าตรงและกระแสไฟฟ้าพัลส์ความถี่สูงๆ

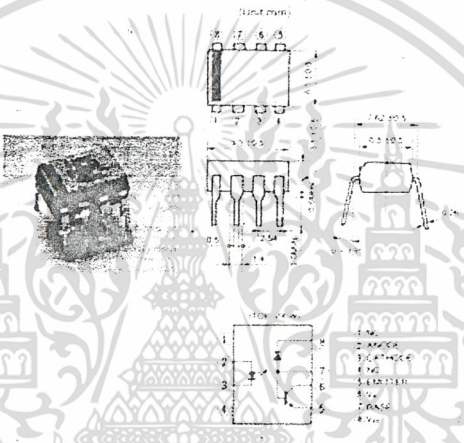
ปัจจุบันมีการใช้งานโฟโต้ไอโซเลเตอร์กันอย่างกว้างขวางมาก เช่นใช้เป็นอินเทอร์เฟซ (interface) ภายในอุปกรณ์การถ่ายโอนสัญญาณในเครื่องอุปกรณ์ O/A (office automation) ใน F/A (factory automation) ต่างๆได้โดยไม่มีการรบกวนจากสนามแม่เหล็กไฟฟ้าภายนอก รูปที่ 3.17 และ 3.18 แสดงตัวอย่างโฟโต้ไอโซเลเตอร์ที่บรรจุลงในภาชนะแบบขาตะขาบและแบบทรงกระบอกตามลำดับ

สิ่งที่ต้องคำนึงถึงที่สำคัญในการใช้งานโฟโตไอโซเลเตอร์มีหลายข้อ เช่น ค่า CRT ค่าความเร็วในการตอบสนอง และการทนทานต่อการพังทลายเมื่อมีสนามไฟฟ้าแรงสูงป้อนเข้าที่ขาด้านอินพุตและด้านเอาท์พุต โดยค่า CRT หมายถึง อัตราส่วนระหว่างกระแสไฟฟ้าเอาท์พุตและกระแสไฟฟ้าอินพุตและแสดงค่าเป็นเปอร์เซ็นต์ ในโฟโตไอโซเลเตอร์เพื่อให้ได้ CRT ที่มีค่าสูงสุดจึงต้องออกแบบให้ภาคเปล่งแสงอยู่ชิดกับภาครับแสงมากที่สุด แต่ในขณะเดียวกันก็ต้องระวังไม่ให้เกิดการพังทลายด้วยสนามไฟฟ้า

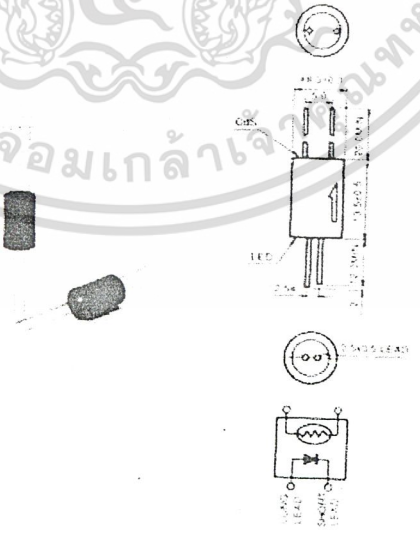
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.19 แสดงตัวอย่างความสัมพันธ์ระหว่างกระแสไฟฟ้าเอาท์พุตและกระแสไฟฟ้าอินพุต รูปที่ 3.20 แสดงความสัมพันธ์ระหว่างค่า CRT และกระแสไฟฟ้าอินพุต

ในการตรวจดูลักษณะความรวดเร็วในการตอบสนองต่อสัญญาณอินพุต อาจใช้สัญญาณกระแสไฟฟ้าอินพุตเป็นรูปคลื่นสี่เหลี่ยม และสังเกตแรงดันเอาท์พุต ดังแสดงในรูปที่ 3.21 เราเรียกระยะเวลาที่ V_{CE} ลดลง 90% ว่าช่วงเวลาเปิด (turn on time : t_{on}) และเรียกระยะเวลาที่ V_{CE} เพิ่มขึ้นเป็น 90% โดยนับจากหลังตัดกระแสไฟฟ้าอินพุตเป็นศูนย์ว่าช่วงเวลาเก็บสัญญาณ (storage time : t_s) และเรียกระยะเวลาที่ V_{CE} เพิ่มขึ้นเป็น 10% ว่าช่วงเวลาปิด (turn off time : t_{off})



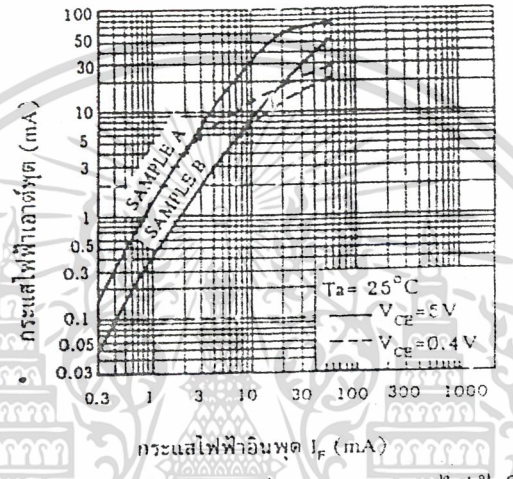
รูปที่ 3.17 ตัวอย่างโฟโต้ไอโซเลเตอร์ชนิดตะขาบ



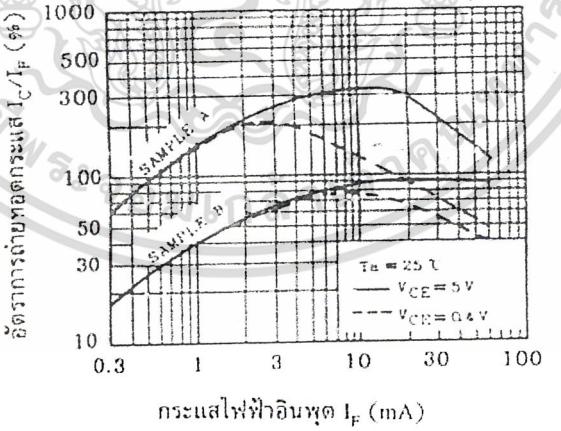
รูปที่ 3.18 ตัวอย่างโฟโต้ไอโซเลเตอร์ชนิดทรงกระบอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเร็วในการตอบสนองโดยทั่วไปจะขึ้นกับคุณสมบัติของภาครับแสง เช่น กรณีของโฟโต้เซลล์ชนิด CdS ใช้เวลาระดับมิลลิวินาที กรณีของโฟโต้ทรานซิสเตอร์ใช้เวลาระดับไมโครวินาที และกรณีของโฟโต้ดาร์ลิงตันใช้เวลาระดับ 10 ไมโครวินาที ในกรณีที่ต้องการความเร็วสูงมากเป็นพิเศษควรใช้ LED ชนิดที่ทำงานที่ความเร็วสูงและใช้พินโฟโต้ไดโอดรับแสง นอกจากนี้ความเร็วในการตอบสนองจะขึ้นอยู่กับค่าของกระแสอินพุต I_F และความต้านทานของโหลด R_L ด้วย ดังแสดงในรูปที่ 3.22 (กรณีนี้ใช้โฟโต้ทรานซิสเตอร์รับแสง)



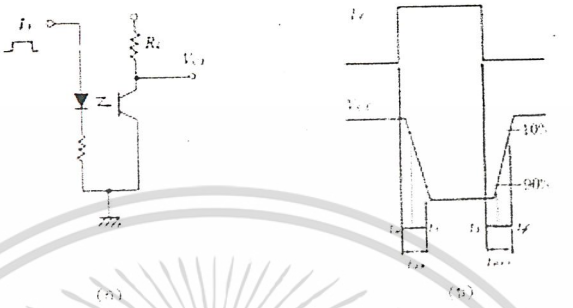
รูปที่ 3.19 ความสัมพันธ์ระหว่างกระแสไฟฟ้าเอาท์พุตและกระแสไฟฟ้าอินพุต I_F ของโฟโต้ไอโซเลเตอร์



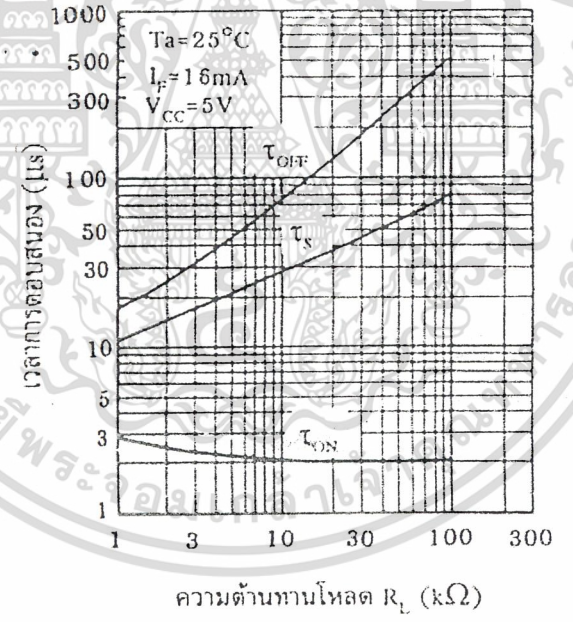
รูปที่ 3.20 ความสัมพันธ์ระหว่าง CRT (I_C/ I_F) และกระแสไฟฟ้าอินพุตในโฟโต้ไอโซเลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทนต่อแรงดันพังทลาย (V_{iso}) ก็เป็นพารามิเตอร์ที่สำคัญในโฟโต้ไอโซเลเตอร์ เพราะบางครั้งมีการใช้วงจรแรงดันสูงในอุปกรณ์ไฟฟ้าด้วย โดยทั่วไป V_{iso} จะมีค่าสูงประมาณกิโลโวลต์ (RMS) และทนได้ถึง 1 นาที



รูปที่ 3.21 (ก) วงจรวัดผลตอบสนองต่อแสงอินพุตแบบพัลส์ และ
(ข) รูปคลื่นของสัญญาณอินพุต I_F สัญญาณเอาท์พุต V_{CE}



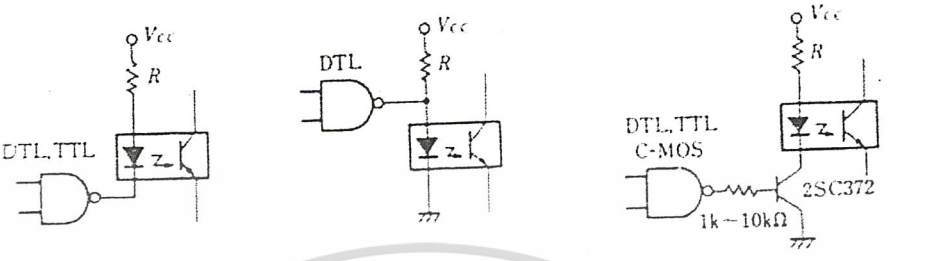
รูปที่ 3.22 ความสัมพันธ์ระหว่างเวลาตอบสนองและความต้านทานโหลด R_L ของโฟโต้ทรานซิสเตอร์

3.3.2 วงจรทำงานพื้นฐานสำหรับโฟโต้ไอโซเลเตอร์

รูปที่ 3.23 แสดงตัวอย่างวงจรขับโฟโต้ไอโซเลเตอร์ด้วยลอจิกไอซี ถ้าใช้กระแสไฟฟ้าอินพุตน้อย เช่น I_F = 10 ~ 15 mA จะใช้ DTL หรือ TTL เป็นตัวขับได้ แต่จะใช้ MOS หรือ C - MOS เป็นตัวขับไม่ได้ ถ้าจะใช้ MOS หรือ C – MOS เป็นตัวขับต้องใช้บัฟเฟอร์ทรานซิสเตอร์ช่วยดัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงในรูปที่ 3.23 (ค) ในกรณีรูป (ก) ถ้าเอาท์พุตของลอจิกไอซีเป็น " 0 " จะมี I_F ไหล แต่ในกรณีรูป (ข) และ (ค) ถ้าเอาท์พุตของลองจิกเป็น " 1 " จึงจะมี I_F ไหล รูปที่ 3.23 แสดงตัวอย่างวงจรด้านเอาท์พุตของโฟโต้ไอโซเลเตอร์



$V_{CC} = 5\ V$

$I_F = \dfrac{V_{CC} - (V_F - V_{OL})}{R}$

(ก)

$V_{CC} = 5\ V$

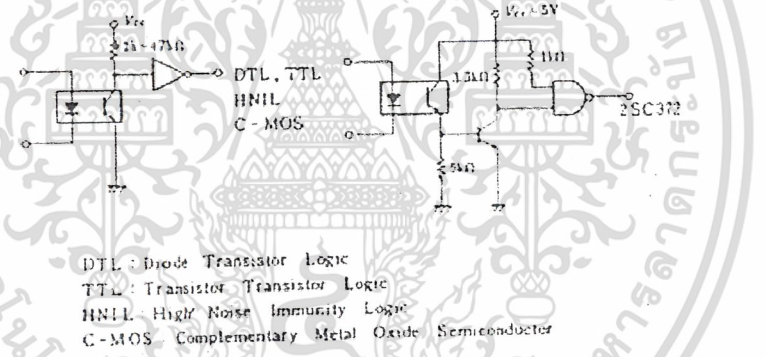
$I_F = \dfrac{V_{CC} - V_F}{R}$

(ข)

$I_F = \dfrac{V_{CC} - (V_F + V_{CE}(sat))}{R}$

(ค)

รูปที่ 3.23 ตัวอย่างวงจรขับโฟโต้ไอโซเลเตอร์ด้วยลอจิกไอซี



รูปที่ 3.24 ตัวอย่างวงจรด้านเอาท์พุตของโฟโต้ไอโซเลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
พัลส์วิดมอดดูเลชั่น

4.1 พัลส์วิดมอดดูเลชั่น (Pulse Width Modulation: PWM)

ในการควบคุมความเร็วมอเตอร์ให้คงที่หรือวงจรสวิตชิ่งซัพพลายที่ควบคุมแรงดันคงที่ นิยมใช้เทคนิคการควบคุมความกว้างของพัลส์ (PWM) ซึ่งเป็นการควบคุมโดยเปลี่ยนแปลงช่วงเวลาที่อุปกรณ์สวิตช์อิเล็กทรอนิกส์ เช่น ทรานซิสเตอร์, เพาเวอร์มอสเฟท, ไอจีบีที (IGBT) หรืออื่น ๆ นั้นนำกระแส

ลักษณะของ **PWM** โดยทั่วไปมีอยู่ 3 ลักษณะคือ

1. เปลี่ยนแปลงทั้งความถี่และความกว้างของพัลส์ (Variation of both frequency and pulse width)

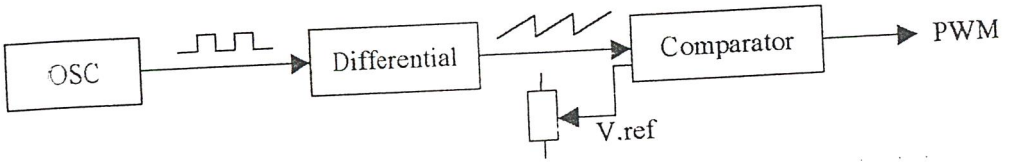
2. เปลี่ยนความถี่ โดยความกว้างของพัลส์คงที่ (Constant pulse width with variable frequency)

3. เปลี่ยนแปลงความกว้างของพัลส์ โดยความถี่คงที่ (Constant frequency with variable pulse width)

การทำงานทั้ง 3 ลักษณะมีข้อดีข้อเสียต่างกันไป การทำงานในลักษณะที่ 2 และลักษณะที่ 3 จะดีกว่าลักษณะที่ 1 เนื่องจากสามารถกำหนดความถี่หรือช่วงเวลา t_{on} ที่ทำให้สวิตชิ่งหรืออุปกรณ์ควบคุมทำงานอย่างมีประสิทธิภาพ แต่แบบที่ 1 นั้นความถี่และความกว้างของพัลส์ กำหนดขึ้นจากวงจรจึงอาจไม่ใช่ความถี่ที่เหมาะสมที่สุดสำหรับการทำงานก็ได้ ดังนั้นวงจร PWM ส่วนใหญ่จึงใช้การทำงานในลักษณะที่ 2 หรือลักษณะที่ 3 เสียเป็นส่วนใหญ่ ในปัจจุบันได้มีผู้ผลิตไอซีในงาน PWM ออกมาจำหน่ายโดยได้พัฒนาให้เป็นวงจรสำเร็จรูปใน IC ตัวเดียว เหมาะสำหรับใช้งานในด้านแหล่งจ่ายไฟตรงโดยเฉพาะ ซึ่งการใช้งานของไอซีเหล่านี้สามารถใช้งานได้ง่าย โดยต่ออุปกรณ์เพิ่มเติมภายนอกเพียงเล็กน้อย เพื่อใช้เป็นตัวกำหนดจุดการทำงานของไอซี PWM นั้น ๆ เพื่อให้การทำงานของวงจรมีความสมบูรณ์ขึ้น อาจมีการเพิ่มเติมวงจรสำหรับระบบต่าง ๆ เช่น วงจรเปรียบเทียบสำหรับป้องกันกระแส หรือแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 หลักการเบื้องต้นของพัลส์วิดมอดดูเลชั่น

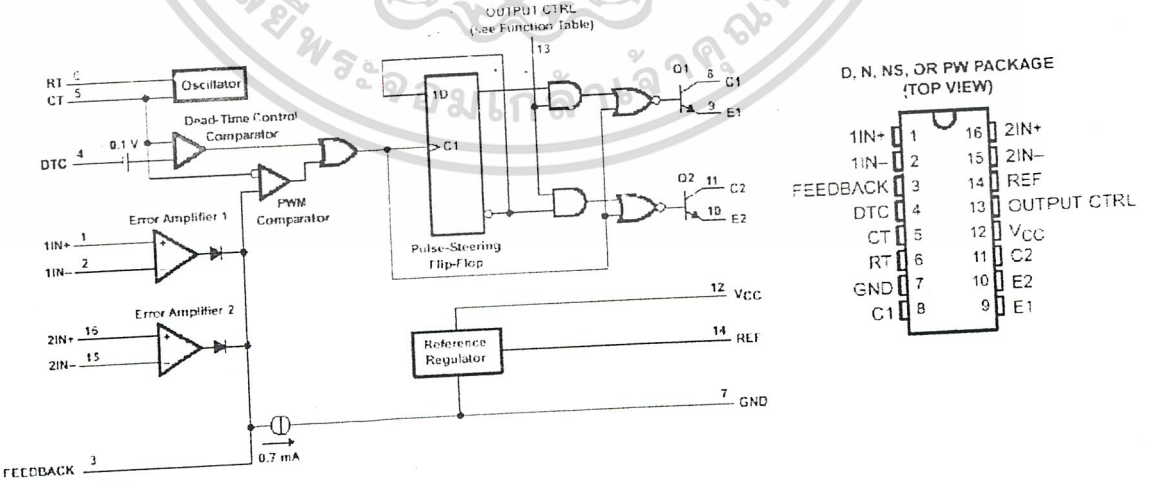


รูปที่ 4.1 แสดงรูปคลื่นสัญญาณเอาท์พุทของพัลส์วิดมอดดูเลชั่น

วงจรสร้างสัญญาณ Pulse Width Modulation (PWM) เป็นตัวกำเนิดสัญญาณความถี่และควบคุมความกว้างของสัญญาณพัลส์ตามที่ค่าคำสั่งจ่ายให้กับวงจร จากรูปกำหนดค่าคำสั่งเท่ากับ V_{in} ซึ่งเป็นระดับแรงดันไฟฟ้ากระแสตรงเพื่อนำไปเปรียบเทียบกับสัญญาณรูปฟันเลื่อยที่ถูกสร้างขึ้นภายในวงจร PWM หน้าที่ของวงจร PWM คือ สร้างสัญญาณรูปฟันเลื่อย และเปรียบเทียบสัญญาณรูปฟันเลื่อยกับระดับแรงดัน V_{in} ซึ่งจะได้เอาท์พุทเป็นสัญญาณพัลส์ที่เปลี่ยนแปลงค่า DUTY CYCLE ตามผลการเปรียบเทียบสัญญาณเอาท์พุทนี้จะนำไปผ่านวงจรขับมอเตอร์เพื่อควบคุมความเร็วของมอเตอร์ต่อไป ส่วนสัญญาณ Dead time control ที่เห็นในรูปเป็นสัญญาณอินพุทอีกขาหนึ่ง IC=TL494 มีไว้ในกรณีที่ต้องการควบคุม dead time โดยมีคุณสมบัติเดียวกันกับขา V_{in}

โครงสร้างภายในวงจรกำเนิดสัญญาณ PWM

ไอซีเบอร์ TL494 ที่ออกแบบมาเพื่อใช้ควบคุมการทำงานของคอนเวอเตอร์ โดยทำงานด้วยโหมดควบคุมจากแรงดัน มีโครงสร้างดังรูป



รูปที่ 4.2 แสดงการจัดโครงสร้างภายในและการจัดขา ไอซี TL494

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดคาบเวลาการทำงาน

วงจร TL494 เป็นวงจร PWM ที่มีความถี่คงที่ คาบเวลาการทำงานของพัลส์กำหนดโดยค่าของ RT และ CT จากภายนอกที่ขา 6 และขา 5 ของไอซี คาบเวลาการทำงานจะกำหนดได้จาก

Single-ended $$f_{osc} = \frac{1.1}{R_T C_T} \quad (4.1)$$

Push-pull $$f_{osc} = \frac{1.1}{2R_T C_T} \quad (4.2)$$

กำหนดให้ f_{osc} เท่ากับ ความถี่ของสัญญาณพัลส์

R_T, C_T เท่ากับ ค่าพารามิเตอร์ของวงจรกำเนิดสัญญาณพัลส์

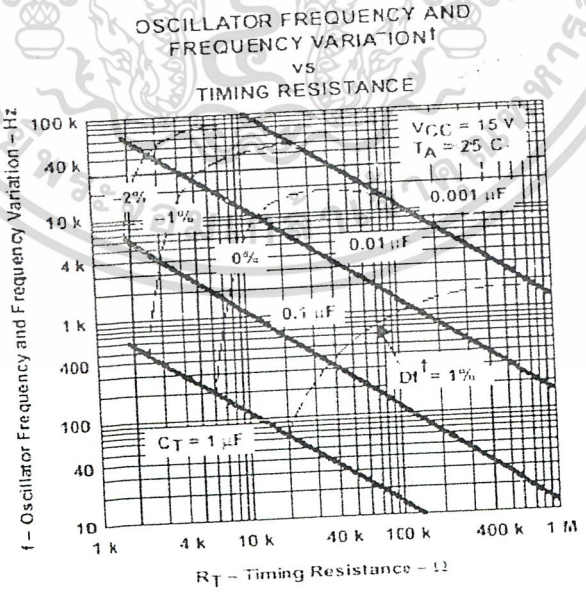
การหาค่า R_T, C_T หาได้ดังนี้

ความถี่ที่ใช้ในการขับมอเตอร์ จะเลือกใช้ที่ 10 kHz เหตุผลในการเลือกความถี่ในย่านนี้เพื่อต้องการให้กระแสของมอเตอร์มีริปเปิ้ลน้อย เราสามารถเลือกความถี่ให้มีค่ามากกว่านี้ แต่ผลตามมาจะมีสัญญาณรบกวนมาก จากคู่มือของ IC TL494 จะหาค่า R_T และค่า C_T จากสูตรดังนี้

$$R_T = \frac{1.1}{f_{osc} C_T} \quad (4.3)$$

พิจารณาเลือก C = 1 nF และความถี่ที่ 10 kHz จะได้ R_T มีค่า

$$R_T = 110\ k\Omega$$



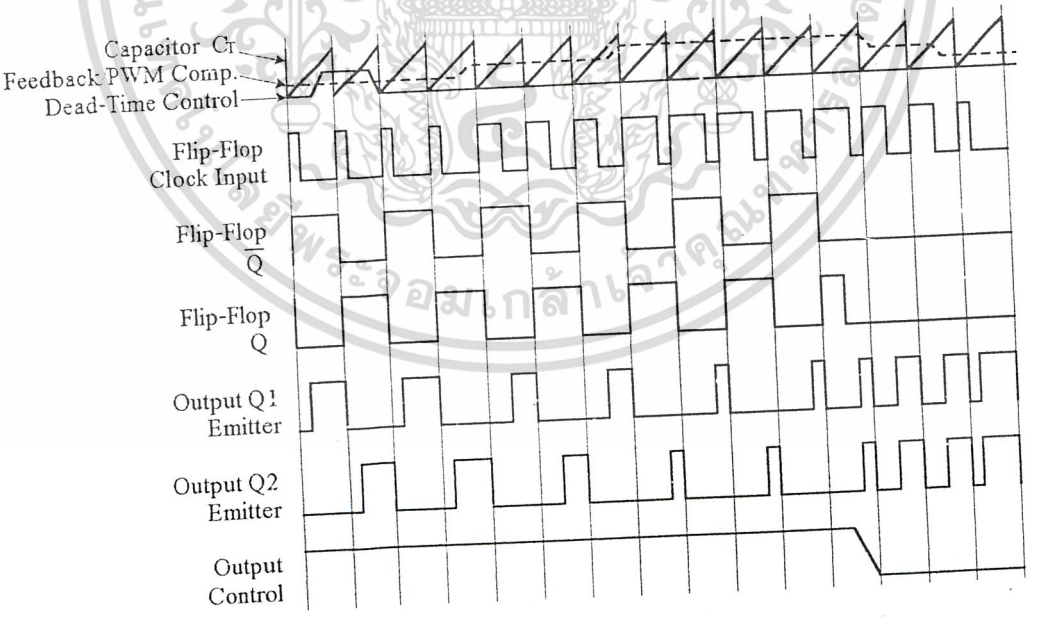
รูปที่ 4.3 แสดงความสัมพันธ์ของค่า R_T, C_T ในการกำหนดความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทำงานของไอซี

ความกว้างเอาท์พุทพัลส์ของไอซีจะได้จากการเปรียบเทียบสัญญาณฟันเลื่อยที่ขา 5 กับแรงดันที่ได้จากวงจรขยายความแตกต่าง (Error Amp) ทั้ง 2 ตัวที่ PWM Comparator ส่วน NOR เกทที่ควบคุมทรานซิสเตอร์เอาท์พุท Q_1 และ Q_2 จะทำงานก็ต่อเมื่อขา C_k ของ Flip-Flop อยู่ในสถานะ Low เท่านั้น ซึ่ง C_k จะเป็น Low ได้ก็ต่อเมื่อแรงดันของสัญญาณฟันเลื่อยมีค่ามากกว่าแรงดันที่มาจาก Error Amp ทั้ง 2 ตัวนั้น คือ แรงดันป้อนกลับจากเอาท์พุทของคอนเวอร์เตอร์หากมีค่าสูง ความกว้างเอาท์พุทพัลส์ของไอซีจะลดลง ในทางกลับกัน แรงดันป้อนกลับหากมีค่าลดลงความกว้างเอาท์พุทพัลส์ของไอซีจะเพิ่มขึ้น

ความกว้างเอาท์พุทพัลส์ของไอซีนี้สามารถกำหนดให้มีค่ามากที่สุดหรือมีค่าเท่ากับศูนย์ได้ด้วยการเปลี่ยนแปลงแรงดันที่ขา 3 จาก 0.5 โวลท์ จนถึง 3.5 โวลท์ ส่วน Error Amp ทั้งสองตัวจะมีช่วงของอินพุท คอมมอนโหมด ตั้งแต่ -0.3 ถึง $(V_{cc} - 2)$ โวลท์ และสามารถใช้ตรวจจับแรงดันหรือกระแสที่เอาท์พุทของคอนเวอร์เตอร์ได้ Error Amp ทั้ง 2 ตัวจะให้เอาท์พุทในสถานะ High (Active High) โดยต่อกันอยู่ในลักษณะ OR ที่ขา non-inverting ของ PWM Comparator การต่อกันในลักษณะนี้ Error Amp ตัวที่ทำให้เกิดความกว้างเอาท์พุทพัลส์ต่ำสุด จะเป็นตัวควบคุมความกว้างของเอาท์พุทพัลส์ของไอซี



รูปที่ 4.4 รูปคลื่นลักษณะการทำงาน IC#TL494

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

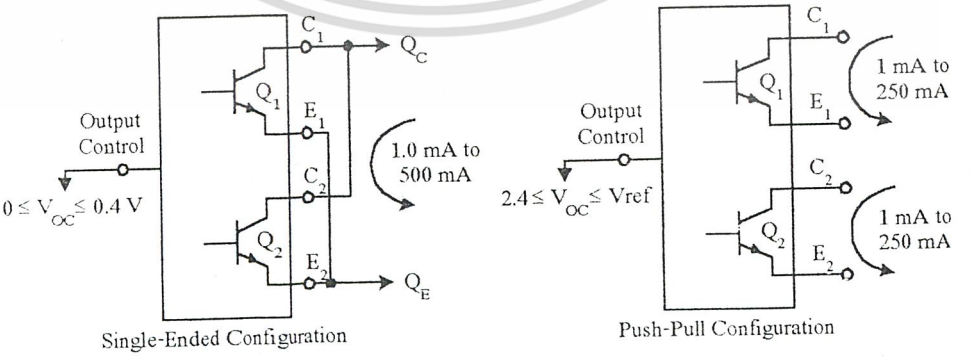
การกำหนดค่าเวลาเผื่อ t_d (dead time)

TL494 สามารถให้ผู้ใช้กำหนดค่าเวลาเผื่อ t_d ของวงจรได้เอง ด้วยการต่อแรงดันระหว่าง 0 ถึง 3.3 โวลท์ที่ขา 4 ของไอซี อย่างไรก็ตาม หากแรงดันที่ขา 4 มีค่าเท่ากับ 0 โวลท์ ค่าเวลาเผื่อต่ำสุดของไอซีจะไม่ต่ำกว่า 4 เปอร์เซ็นต์ของค่าคาบเวลาการทำงานเนื่องจากมีแรงดันออฟเซต 120 มิลลิโวลท์อยู่ภายใน ดังนั้นช่วงเวลา t_{on} สูงสุดของคอนเวอร์เตอร์ที่ได้จากไอซีจะเท่ากับ 48% ของค่าคาบเวลาเมื่อต่อขา 13 (Output control) เข้ากับขา 14 (+5 V_{ref}) และมีค่าเท่ากับ 96% ของค่าคาบเวลาเมื่อต่อขา 13 ลงกราวด์

การเลือกใช้ Q_1 และ Q_2 ที่เอาท์พุทของไอซี

เอาท์พุท Q_1 และ Q_2 ของไอซีสามารถทำงานได้ 2 โหมด คือ ทำงานพร้อมกัน หรือ สลับกันทำงาน ซึ่งสามารถเลือกการทำงานได้ที่ขา 13 (Output Control) โดยขณะที่ C_1 ดิสชาร์จเอาท์พุทของ Dead-Time Comparator จะให้พัลส์ออกมา C_k จะมีสถานะเป็น High และหยุดการทำงานของ Q_1 และ Q_2 ถ้ากำหนดให้ขา 13 มีสถานะเป็น High โดยการต่อเข้ากับขา 4 (+5V_{ref}) Q1 และ Q_2 จะสลับการทำงานของ Flip-Flop เพื่อให้ขับคอนเวอร์เตอร์แบบ พุช-พูล ในกรณีนี้คาบเวลาการทำงานจะเป็น 2 เท่าของค่าคาบเวลาสัญญาณฟันเลื่อยของไอซี แต่ถ้ากำหนดให้ขา 13 มีสถานะเป็น Low โดยการต่อลงกราวด์ (ยกเลิก Flip-Flop) Q_1 และ Q_2 เข้าด้วยกันได้ ถ้าต้องการให้นำกระแสได้มากขึ้น ในกรณีนี้คาบเวลาการทำงานจะมีค่าเท่ากับค่าคาบเวลาของสัญญาณฟันเลื่อยของไอซี(ช่วงเวลานำกระแสสูงสุดเท่ากับ 96% ของค่าคาบเวลา)

TL494 ต้องการไฟเลี้ยงในช่วง $7 \leq V_{CC} \leq 40$ โวลท์ มีแรงดันอ้างอิงภายใน V_{ref} = 5 โวลท์ และสามารถจ่ายกระแสได้ถึง 10 มิลลิแอมป์เพื่อใช้กับวงจรภายนอกได้ โดยมีค่าความถูกต้อง $\pm$1.5% ความคลาดเคลื่อนทางอุณหภูมิมีค่าน้อยกว่า 50 มิลลิโวลท์ เมื่อทำงานในช่วง 0° ถึง 70°

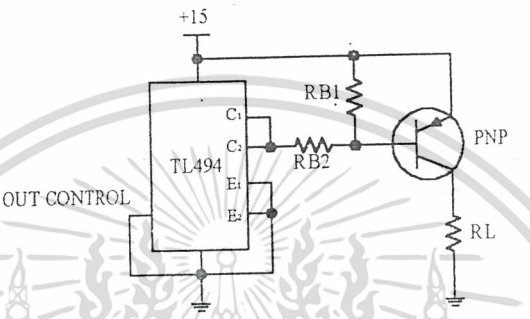


รูปที่ 4.5 แสดงลักษณะการเลือกใช้เอาท์พุทของ TL494

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การใช้ IC TL494 ร่วมกับทรานซิสเตอร์ภายนอก

การใช้ IC TL494 ในงานสวิตชิ่งที่ต้องการกระแสสูง ๆ โดยทำงานร่วมกับทรานซิสเตอร์ภายนอก สังเกตุว่าทรานซิสเตอร์ภายนอกที่ใช้ในวงจรจะใช้ทรานซิสเตอร์ชนิด PNP, NPN ถ้าเป็นชนิด PNP โดยมีตัวต้านทาน R_{B1}, R_{B2} เป็นส่วนไบแอส ซึ่งทรานซิสเตอร์ที่อยู่ภายใน IC TL494 จะทำหน้าที่เป็นตัวสวิตช์ต่อระหว่างขาเบสของทรานซิสเตอร์กับกราวด์และเปิดวงจร



รูปที่ 4.6 การใช้ TL494 กับทรานซิสเตอร์ภายนอก

ถ้าทรานซิสเตอร์ภายในเปิดวงจรก็ไม่มีกระแสไหลเข้าไปยังตัวต้านทานไบแอสทำให้ทรานซิสเตอร์ภายนอกเปิดวงจรด้วย แต่ถ้าทรานซิสเตอร์ภายในปิดวงจรทำให้ตัวต้านทาน R_{B2} ถูกต่อลงกราวด์ และเกิดกระแสไหลผ่านตัวต้านทานทั้งสอง (R_{B1}, R_{B2}) เกิดเป็นวงจรแบบแรงดันและทำให้ทรานซิสเตอร์นำกระแสในที่สุด

ในการออกแบบค่าตัวต้านทาน R_{B1}, R_{B2} นั้นจะสมมุติให้กระแส $I_{B\ leader}$ ไหลผ่าน R_{B1} และรวมกับกระแสเบสของทรานซิสเตอร์สวิตชิ่งผ่าน R_{B2} (สามารถเลือกกระแส $I_{B\ leader}$ ได้ประมาณ 1 ใน 10 ของกระแสเบสทรานซิสเตอร์สวิตชิ่ง)

สำหรับการเลือกกระแสเบสของทรานซิสเตอร์จะใช้กระแสค่าสูงสุดของคอลเลคเตอร์ I_{Cmax} และค่าอัตราขยาย β น้อยที่สุดของทรานซิสเตอร์ หาได้จาก DATASHEET, β_{min} สมการในการหากระแสเบส

$$I_{B\,min} = \frac{I_{C\,max}}{\beta_{min}} \qquad (4.4)$$

แรงดันตกคร่อม R_{B1} V_{BE} = 0.7 V เพื่อให้แน่ใจว่าทรานซิสเตอร์นำกระแส กระแสที่ไหลผ่าน R_{D1} คือกระแส $I_{D\ leader}$ หา R_{B1} ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_{B1} = \frac{V_{BE}}{I_{B\,leader}} \tag{4.5}$$

ในทางปฏิบัติจะเลือกใช้ค่า R_{B1} ที่มากกว่าจากการคำนวณเพื่อให้แน่ใจว่าแรงดันตกคร่อมระหว่างขาเบสกับอิมิตเตอร์อย่างน้อยที่ 0.7 V สมการคำนวณหาค่า R_{B2}

$$R_{B2} = \frac{V_{DC\,supply} - V_{BE} - V_{sat}}{I_{B\,leader} + I_{B\,min}} \tag{4.6}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การใช้งานไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ที่ใช้ในการทำโครงงานนี้จะอ้างอิงไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งมีหน่วยความจำภายในแบบแฟลช (flash memory) ของ Atmel Corporation มีเบอร์ขึ้นต้นด้วย AT89 เหตุผลที่ใช้ไมโครคอนโทรลเลอร์แบบนี้ในการทำโครงงานเพื่อใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีด้วยกันหลายประการดังนี้

1. หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช ทำให้สามารถลบและเขียนใหม่ได้นับพันครั้ง จึงสามารถใช้งานในรูปแบบของไมโครคอนโทรลเลอร์ชิปเดี่ยวไม่ต้องใช้หน่วยความจำภายนอกส่งผลให้สามารถใช้งานพอร์ตอินพุทเอาท์พุทของไมโครคอนโทรลเลอร์ได้อย่างเต็มประสิทธิภาย

2. ต้นทุนและเวลาในการพัฒนาระบบไมโครคอนโทรลเลอร์ลดลงอย่างมากเนื่องจากไม่ต้องใช้เครื่องมือพัฒนาจำพวกอีมูเลเตอร์และเครื่องโปรแกรมอีพรอม (EPROM)

3. บริษัทผู้ผลิตได้ทำการผลิตไมโครคอนโทรลเลอร์ตระกูลนี้ออกมาหลายเบอร์ และมีความสามารถแตกต่างกันไป ทำให้มีทางเลือกในการใช้งานสูง

4. ด้วยการใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ทำให้สามารถป้องกันการคัดลอกข้อมูลของหน่วยความจำโปรแกรมได้เป็นอย่างดี

5. ในบางเบอร์ของไมโครคอนโทรลเลอร์ที่ผลิตโดย Atmel สามารถทำการโปรแกรมข้อมูลในหน่วยความจำได้โดยที่ไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ออกมาทำการโปรแกรมใหม่ หรือเรียกว่า **การโปรแกรมในวงจร หรือ ในระบบ (In-system programming)** โดยใช้ลักษณะการติดต่อแบบ **SPI (Serial Peripheral Interface)** ทำให้การพัฒนาหรือการซ่อมบำรุง ตลอดจนการปรับปรุงหรืออัปเกรดข้อมูลในหน่วยความจำโปรแกรมทำได้อย่างสะดวก ภายใต้งบประมาณที่ไม่สูงมากนัก

6. ชุดคำสั่งและสถาปัตยกรรมพื้นฐานเหมือนกับไมโครคอนโทรลเลอร์ MCS-51 ของผู้ผลิตอื่นไม่ว่าจะเป็นอินเทล, ซีเมนส์ หรือดัลลัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1 คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ MCS-51

คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ตระกูล MCS-51 อนุกรม AT89xx มีดังนี้

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม ในบางเบอร์จะมีหน่วยความจำแบบอีอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานเป็นได้ทั้งอินพุทและเอาท์พุท
- มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์
- ไมเมอร์/เคาร์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัปต์ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ในภายชิป
- มีวงจรสื่อสารอนุกรมแบบ SPI สำหรับในอนุกรม AT89Sxx
- มีวอตช์ด็อกไทเมอร์ในตัว สำหรับในอนุกรม AT89Sxx

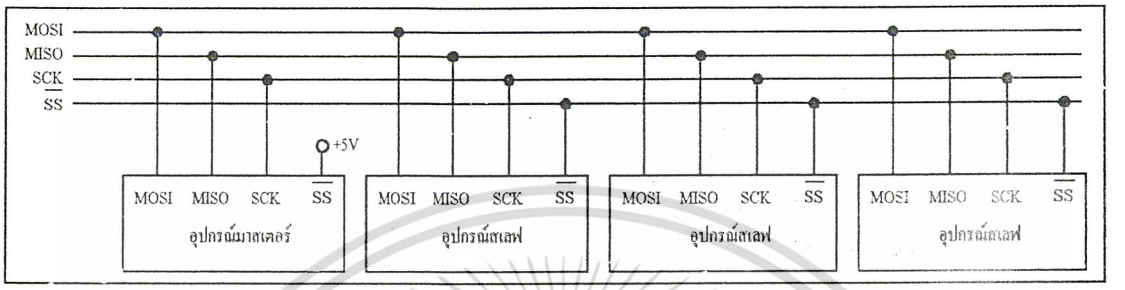
ในโครงงานนี้ใช้ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช เบอร์ AT89S8252 ซึ่งมีหน่วยความจำโปรแกรมขนาด 8 กิโลไบต์ หน่วยความจำข้อมูล แรม 256 ไบต์ อีอีพรอม 2 กิโลไบต์และจำนวนไทเมอร์/เคาร์เตอร์ 16 บิต 3 ตัว

คุณสมบัติเด่นอีกประการหนึ่งที่มีอยู่ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช ในอนุกรม AT89Sxx ของ Atmel คือ ส่วนเชื่อมต่ออุปกรณ์อนุกรม หรือ SPI (Serial Peripheral Interface) ซึ่งเป็นอีกหนึ่งรูปแบบของการขยายขีดความสามารถของไมโครคอนโทรลเลอร์สมัยใหม่ ซึ่งในอดีตมีเพียงไมโครคอนโทรลเลอร์ตระกูล 68HC11 ของโมโตโรล่าเท่านั้นที่มีความสามารถในส่วนนี้ แต่ในปัจจุบันและอนาคต SPI จะกลายเป็นคุณสมบัติมาตรฐานที่ต้องมีในไมโครคอนโทรลเลอร์ ดังจะเห็นได้จากในไมโครคอนโทรลเลอร์ PIC รุ่นใหม่ รวมถึงไมโครคอนโทรลเลอร์ AVR ซึ่งเป็นไมโครคอนโทรลเลอร์อีกตระกูลหนึ่งของ Atmel ต่างก็บรรจุส่วน SPI ไว้ด้วย

ส่วนเชื่อมต่ออุปกรณ์อนุกรม หรือ SPI นี้เป็นการเชื่อมต่อไมโครคอนโทรลเลอร์เข้ากับอุปกรณ์ภายนอกที่มีลักษณะการถ่ายทอดข้อมูลในลักษณะอนุกรมแบบซิงโครนัส (Synchronous) ความเร็วสูง ในการสื่อสารข้อมูลแบบซิงโครนัสจะต้องใช้สัญญาณนาฬิกาเพื่อกำหนดจังหวะในการถ่ายทอดข้อมูลร่วมกัน และต้องกำหนดสถานภาพของอุปกรณ์ในระบบอย่างชัดเจน เนื่องจากใน

เอกสารนี้มีไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ทำซ้ำ ดัดแปลง เผยแพร่ หรือแสวงหาประโยชน์ทางการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อแบบ SPI สามารถต่อพ่วงอุปกรณ์ได้เป็นจำนวนมาก โดยต้องมีการกำหนดตัวควบคุมหลักเป็นอุปกรณ์ตัวแม่หรือมาสเตอร์ (master) และอุปกรณ์ที่นำมาต่อพ่วงในระบบเป็นอุปกรณ์ลูกหรือสเลฟ (slave) ดังแสดงรูปแบบ SPI อย่างง่ายในรูปที่ 5.2



รูปที่ 5.1 รูปแบบของการเชื่อมต่อของ SPI อย่างง่าย

อุปกรณ์ตัวแม่หรือมาสเตอร์โดยส่วนใหญ่จะเป็นไมโครคอนโทรลเลอร์ และอุปกรณ์ตัวลูกหรือสเลฟ มักเป็นไอซีที่ทำหน้าที่เฉพาะพิเศษ เช่น หน่วยความจำ ไอซีขยายพอร์ต หรือ UART เป็นต้น แต่ด้วยการใช้ความสามารถของ SPI อาจสร้างโครงข่ายระบบควบคุมแบบมัลติโปรเซสเซอร์ได้ โดยทำการเชื่อมต่อไมโครคอนโทรลเลอร์หลายตัวเข้าด้วยกันโดยผ่านทางขาเชื่อมต่อของระบบ SPI นั่นหมายความว่า อุปกรณ์ตัวลูกหรือสเลฟนั้นก็คือไมโครคอนโทรลเลอร์หรือไมโครโปรเซสเซอร์อีกชุดหนึ่งหรืออีกหลาย ๆ ชุดในกรณีต่อพ่วงกันมากกว่า 2 ตัว

นอกเหนือไปจากการใช้ SPI ในการติดต่อสื่อสารข้อมูลแบบซิงโครนัสแล้ว ในไมโครคอนโทรลเลอร์อนุกรม AT89Sxx ยังใช้ความสามารถของส่วน SPI ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมภายในตัวชิปด้วย จึงทำให้กระบวนการที่เรียกว่า การโปรแกรมหน่วยความจำโปรแกรมในระบบหรือในวงจร (In-System Programming: ISP) ทำให้สามารถแก้ไขหน่วยความจำโปรแกรมได้โดยไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ออกจากระบบ เพียงต่อสายเข้าที่พอร์ต SPI คือขา SS, MOSI, MISO และ SCK (ซึ่งก็คือขา P1.4-P1.7) แล้วทำการรันซอฟต์แวร์ที่เขียนขึ้นเพื่อการโปรแกรมในระบบ ก็จะสามารถแก้ไขหน่วยความจำโปรแกรมได้แล้ว ซอฟต์แวร์ที่ใช้ในการเขียนข้อมูลลงหน่วยความจำโปรแกรมภายในตัวชิปนี้ ได้แก่

- SXA51 เป็นตัว Compile ไฟล์นามสกุล ASM ที่เขียนด้วยโปรแกรม Editor ต่าง ๆ ให้เป็นไฟล์นามสกุล HEX
- PonyProg - Serial Device Programmer Version 1.17h เป็นโปรแกรมที่ใช้ในการทำให้ไฟล์นามสกุล HEX ที่ได้จากการ Compile จากโปรแกรม SXA51 เป็นไฟล์นาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สกุล HEX ในรูปแบบบน Windows สามารถดาวน์โหลดได้ที่ http://www.LancOS.com

- Atmel AVR ISP Version 2.65 เป็นโปรแกรมที่ใช้ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมภายในตัวชิปไมโครคอนโทรลเลอร์ โดยใช้ไฟล์นามสกุล HEX จากโปรแกรม PonyProg สามารถดาวน์โหลดโปรแกรมได้ที่ http://www.atmel.com/

วิธีการโปรแกรมแบบนี้จะช่วยลดต้นทุนในเรื่องของค่าเครื่องโปรแกรมเวลาในการปรับปรุง และแก้ไขหน่วยความจำโปรแกรมภายในไมโครคอนโทรลเลอร์ เพราะเพียงแต่ต่อสายเข้าไปเท่านั้น ไม่ต้องถอดฝาเครื่องหรือส่วนห่อหุ้มตัวชิปออก เพื่อนำตัวชิปออกมาโปรแกรมภายนอก ทำให้ลดค่าใช้จ่ายในการนำเครื่องหรือชิ้นงานกลับมายังแผนกซ่อมบำรุง เพียงเพื่อแก้ไขข้อมูลภายในหน่วยความจำโปรแกรมของไมโครกอนโทรลเลอร์

5.2 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 ทุกเบอร์จะมีขาใช้งานพื้นฐานเหมือนกันดังแสดงในรูปที่ 5.1 โดยมีรายละเอียดขั้นต้น ดังนี้

ขา Vcc ใช้สำหรับต่อไฟเลี้ยง +5 V

ขา GND เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ

ขาพอร์ต 0 (P0.0-P0.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนั้นขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยใช้กระบวนการมัลติเพล็กซ์เข้าช่วย เพื่อสลับการทำงานให้เป็นได้ทั้งขาติดต่อแอดเดรสและขาข้อมูล

ขาพอร์ต 1 (P1.0-P1.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 1 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย นอกจากนั้นในอนุกรม AT89Sxx จะใช้ขา P1.0 เป็นขาอินพุทสำหรับนับค่าของไทเมอร์ 2 และ P1.1 เป็นขาอินพุททริกเกอร์ของไทเมอร์ 2 ในขณะที่ขา P1.4 ถึง P1.7 เป็นขาสำหรับเชื่อมต่อแบบ SPI เพื่อทำการโปรแกรมข้อมูลในระบบ

เอกสารนี้มีลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P1.4 ขา $\overline{SS}$ (Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์สเลฟในระบบการติดต่อแบบ SPI

P1.5 ขา MOSI (Master data Output, Slave data Input) ในการติดต่อกับพอร์ต SPI

P1.6 ขา MISO (Master data Input, Slave data Output) ในการติดต่อกับพอร์ต SPI

P1.7 ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI

ขาพอร์ต 2 (P2.0-P2.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 2 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนั้นขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความภายนอก (A8-A15)

ขาพอร์ต 3 (P3.0-P3.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 3 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนั้นขาพอร์ต 3 ยังเป็นขาที่มีหน้าที่การใช้งานพิเศษ ดังมีรายละเอียดขั้นต้นต่อไปนี้

P3.0 ใช้เป็นขาอินพุทสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา RxD

P3.1 ใช้เป็นขาอินพุทสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา TxD

P3.2 ใช้เป็นขาอินพุทสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 0 หรือขา $\overline{INT0}$

P3.3 ใช้เป็นขาอินพุทสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 1 หรือขา $\overline{INT1}$

P3.4 ใช้เป็นขาอินพุทสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 0 หรือขา T0

P3.5 ใช้เป็นขาอินพุทสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 1 หรือขา T1

P3.6 ใช้เป็นขาสัญญาณ $\overline{WR}$ ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

P3.7 ใช้เป็นขาสัญญาณ $\overline{RD}$ ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

ขารีเซต (Reset) ใช้ในการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยในการป้อนสัญญาณเพื่อรีเซต สถานะที่ขานี้ต้องอยู่ในระดับรีเซตอย่างน้อย 2 แมชีนไซเกิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างเป็นปกติ

ขา ALE/$\overline{PROG}$ (Address Latch Enable/Program pulse input) เป็นขาที่ใช้ในการควบคุมการแลตช์ของขาพอร์ต 0 เมื่อมีการใช้งานหน่วยความจำภายนอก นอกจากนั้นขานี้ยังใช้เป็น

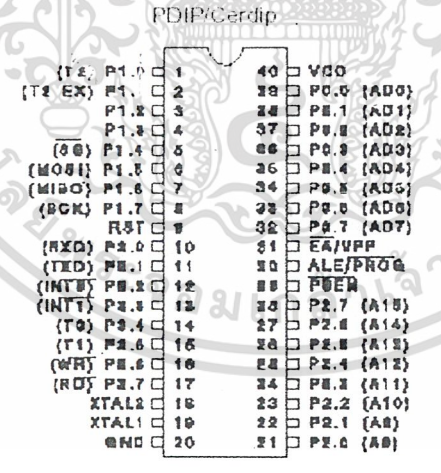
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาสำหรับรับพัลส์ของการโปรแกรมสำหรับโปรแกรมข้อมูลลงในไมโครคอนโทรลเลอร์ MCS-51 ในรุ่นที่มีหน่วยความจำโปรแกรมแบบอีอีพรอม

ขา $\overline{\text{PSEN}}$ **(Program Store Enable)** ขานี้ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก ตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณออกมาที่ขานี้ 2 ครั้งในแต่ละแมชีนไซเกิล แต่ถ้าหากติดต่อกับหน่วยความจำข้อมูลภายนอก ขานี้จะไม่มีการส่งสัญญาณใด ๆ ออกมา

ขา $\overline{\text{EA}}$ / Vpp **(External Access enable/Programming voltage input)** ใช้สำหรับเลือกการติดต่อหน่วยความจำโปรแกรมจากภายนอกหรือภายในตัวไมโครคอนโทรลเลอร์ ถ้าหากขานี้เป็น "0" เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายนอกแต่ถ้าหากขานี้เป็น "1" เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ นอกจากนี้ ที่ขานี้ยังใช้เป็นขาอินพุทสำหรับรับแรงดันไฟสูงสำหรับการโปรแกรมหน่วยความจำภายในไมโครคอนโทรลเลอร์ สำหรับไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชต้องการแรงดันสำหรับการโปรแกรม +12 V

ขา **XTAL1** และ **XTAL2** เป็นขาสำหรับต่อคริสตอลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์



รูปที่ 5.2 การจัดขามาตราฐานของไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ไทเมอร์/เคาน์เตอร์ในไมโครคอนโทรลเลอร์ MCS-51

ไทเมอร์/เคาน์เตอร์ **(Timer/Counter)** เป็นอีกหนึ่งส่วนประกอบที่สำคัญของไมโครคอนโทรลเลอร์ เนื่องจากในการทำงานของไมโครคอนโทรลเลอร์จะต้องมีการเก็บและตรวจสอบค่าของเวลาและจำนวนของสัญญาณนาฬิกาอยู่ตลอดเวลา ทั้งนี้เพื่อประโยชน์ในการสร้างฐานเวลา สร้างสัญญาณพัลส์ เปรียบเทียบค่าเวลา หรือเปรียบเทียบค่าของการนับ รวมไปถึงการกำหนดอัตราเร็วในการสื่อสารข้อมูลของพอร์ตอนุกรมด้วย

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C51 มีวงจรไทเมอร์/เคาน์เตอร์ขนาด 16 บิต 2 ตัว โดยค่าของวงจรไทเมอร์/เคาน์เตอร์นี้จะเก็บไว้ในรีจิสเตอร์ขนาด 16 บิตที่ชื่อ ไทเมอร์ 0 (Timer0) และไทเมอร์ 1 (Timer1) เรียกสั้น ๆ ว่า T0 และ T1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และอนุกรม AT89Sxx จะมีไทเมอร์/เคาน์เตอร์ถึง 3 ตัว คือ ไทเมอร์ 2 (Timer2: T2) เพิ่มเติมมา โดยรีจิสเตอร์ไทเมอร์ /เคาน์เตอร์ทั้งสามตัวสามารถกำหนดให้ทำงานเป็นตัวตั้งเวลาหรือไทเมอร์และตัวนับหรือเคาน์เตอร์ได้อย่างอิสระต่อกัน การนับค่าของไทเมอร์/เคาน์เตอร์ในไมโครคอนโทรลเลอร์ MCS-51 จะเป็นการนับขึ้นหรือเพิ่มค่าเพียงทางเดียว

5.3.1 การทำงานเป็นไทเมอร์

เมื่อกำหนดให้ทำงานเป็นตัวตั้งเวลาหรือไทเมอร์ ค่าของรีจิสเตอร์จะเพิ่มขึ้นในทุก ๆ แมชชีนไซเกิล ดังนั้นเมื่อทำงานเป็นไทเมอร์ รีจิสเตอร์จะทำการนับค่าของแมชชีนไซเกิลนั่นเอง และเนื่องจากแมชชีนไซเกิลประกอบด้วยคาบเวลาของวงจรกำเนิดสัญญาณนาฬิกา 12 คาบเวลา ดังนั้นอัตราในการนับของรีจิสเตอร์จึงเท่ากับ 1/12 ของความถี่สัญญาณนาฬิกา

5.3.2 การทำงานเป็นเคาน์เตอร์

เมื่อทำงานเป็นตัวนับหรือเคาน์เตอร์ ค่าของรีจิสเตอร์จะเพิ่มขึ้นก็ต่อเมื่อมีการเปลี่ยนแปลงของระดับลอจิกจาก "1" เป็น "0" เกิดขึ้นที่ขาอินพุททางฮาร์ดแวร์ของวงจรไทเมอร์/เคาน์เตอร์ ซึ่งก็คือ ขาT0 (P3.4), ขาT1 (P3.5) และขาT2 (P1.0) ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และอนุกรม AT89Sxx โดยจะมีการสุ่มรับสัญญาณจากขาอินพุทในทุก ๆ คาบเวลาที่ 2 ของสเตตที่ 5 (S5P2) ในแต่ละแมชชีนไซเกิล

เมื่อสัญญาณอินพุทเปลี่ยนแปลงจาก "1" เป็น "0" เป็นเวลาหนึ่งไซเกิล ในไซเกิลต่อมาค่าของการนับจะเพิ่มขึ้นหนึ่งค่า และจะไปปรากฏในรีจิสเตอร์ภายในคาบเวลาที่ 1 ของสเตตที่ 3 (S3P1) ของแมชชีนไซเกิลต่อไปหลังจากที่ตรวจจับพบการเปลี่ยนแปลงที่ขาไทเมอร์อินพุทแล้ว เมื่อเป็นเช่นนี้ในกระบวนการตรวจจับการเปลี่ยนแปลงของสัญญาณอินพุทที่ขาไทเมอร์จะต้องใช้ 2 แมชชีนไซเกิล อัตราการนับของเคาน์เตอร์จึงเท่ากับ 1/24 ของความถี่สัญญาณนาฬิกา ดังนั้น ความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สูงสุดของสัญญาณอินพุทที่ไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช สามารถตรวจจับได้จึงเท่ากับความถี่ของสัญญาณนาฬิกาหารด้วย 24 ยกตัวอย่าง ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89S8252 สามารถใช้สัญญาณนาฬิกาได้สูงสุด 24 MHz ดังนั้นความถี่สูงสุดของสัญญาณอินพุทที่ไทเมอร์/เคาน์เตอร์สามารถตรวจจับได้คือ 1 MHz

5.3.3 รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

ในการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีรีจิสเตอร์ที่ต้องเกี่ยวข้องเป็นพื้นฐานอยู่ 6 ตัว ดังมีรายละเอียดต่อไปนี้

- รีจิสเตอร์ไทเมอร์

มีด้วยกัน 4 ตัวคือ TL0 มีแอดเดรสอยู่ที่ 8AH, TH0 มีแอดเดรสอยู่ที่ 8CH, TL1 มีแอดเดรสอยู่ที่ 8BH และ TH1 มีแอดเดรสอยู่ที่ 8DH รีจิสเตอร์ทั้ง 4 ตัวจะอยู่ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษ หรือ SFR รีจิสเตอร์แต่ละตัวมีขนาด 8 บิต แต่ในการใช้งานโดยทั่วไปมักใช้รวมกันโดยจัดเป็นคู่คือ TL0 กับ TH0 รวมกันเป็นรีจิสเตอร์ Timer 0 ขนาด 16 บิต และTL1 กับ TH1 รวมกันเป็นรีจิสเตอร์ Timer 1 ขนาด 16 บิต โดยใน TL0 และ TL1 จะเก็บข้อมูล 8 บิตล่าง ส่วน TH0 และ TH1 เก็บข้อมูล 8 บิตบน รีจิสเตอร์ไทเมอร์ทั้ง 2 คู่เมื่อนำมาใช้งานร่วมกันจะสามารถเก็บค่าของการนับได้สูงสุด 65,536 ค่าหรือ FFFFH เมื่อนับถึงค่านี้แล้วก็จะวนไปเริ่มนับ 0000H ใหม่ และเมื่อเกิดการนับรอบใหม่ จะมีการเซตบิต TF0 หรือ TF1 ในรีจิสเตอร์ TCON ที่ใช้ควบคุมการทำงานของไทเมอร์ เพื่อแจ้งให้ทราบว่าเกิดการนับเกินค่าสูงสุดแล้ว การเซตบิต TF0 หรือ TF1 ขึ้นอยู่กับว่าเลือกใช้งานรีจิสเตอร์ไทเมอร์ตัวใดอยู่

- รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ หรือ

TCON (Timer/Counter Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 88H ในพื้นที่ของรีจิสเตอร์ SFR สามารถเข้าถึงได้ในระดับบิต มีรายละเอียดการทำงานดังนี้

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

TF1 (Timer 1 overflow flag): เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ Timer 1 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกัน โดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TR1 (Timer 1 run control bit): ใช้ในการเปิดปิดการทำงานของไทเมอร์ 1 (อีนาเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ถ้าต้องการให้ไทเมอร์ 1 ทำงานต้องเซตบิตนี้ให้เป็น "1"

TF0 (Timer 0 overflow flag): เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ Timer 0 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกัน โดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

TR0 (Timer 0 run control bit): ใช้ในการเปิดปิดการทำงานของไทเมอร์ 0 (อีนาเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ถ้าต้องการให้ไทเมอร์ 0 ทำงานต้องเซตบิตนี้ให้เป็น "1"

IE1 (External Interrupt 1 edge flag): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุทอินเตอร์รัปต์ 1 (INT1) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT1 (Interrupt 1 type control bit): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ โดยใช้ในการเลือกลักษณะของสัญญาณอินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุทอินเตอร์รัปต์ 1 (INT1) การเซตและเคลียร์ทำได้ด้วยกระบวนการซอฟท์แวร์

"0" เลือกขอบขาลงของสัญญาณ (Falling edge)

"1" เลือกระดับลอจิกต่ำ (low level triggered)

IE0 (External Interrupt 0 edge flag): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุทอินเตอร์รัปต์ 0 (INT0) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT0 (Interrupt 0 type control bit): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ โดยใช้ในการเลือกลักษณะของสัญญาณอินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุทอินเตอร์รัปต์ 0 (INT0) การเซตและเคลียร์ทำได้ด้วยกระบวนการซอฟท์แวร์

"0" เลือกขอบขาลงของสัญญาณ (Falling edge)

"1" เลือกระดับลอจิกต่ำ (low level triggered)

- **รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์หรือ**

TMOD (Timer/Counter Mode Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 89H ในพื้นที่ของรีจิสเตอร์ SFR ไม่สามารถเข้าถึงได้ในระดับบิต แบ่งการทำงานออกเป็น 2 ส่วนคือ 4 บิตล่างใช้ในการเลือกโหมดการทำงานของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์ 0 และ 4 บิตบนใช้ในการเลือกโหมดการทำงานของไทเมอร์ 1 ดังนั้นในการอธิบายการทำงานจะขออธิบายเพียงส่วนเดียวดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
GATE	$C/\overline{T}$	M1	M0	GATE	$C/\overline{T}$	M0	M0
ไทเมอร์ 1				ไทเมอร์ 0			

GATE: ใช้เลือกลักษณะการควบคุมการทำงานของไทเมอร์/เคาน์เตอร์

"0" ไทเมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRx ในรีจิสเตอร์ TCON เป็น "1" เรียกการควบคุมแบบนี้ว่า การควบคุมทางซอฟท์แวร์

"1" ไทเมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRx ในรีจิสเตอร์ TCON เป็น "1" และสถานะลอจิกที่ขาอินพุทอินเตอร์รัปท์ INT0 และ INT1 เป็น "1" เรียกการควบคุมแบบนี้ว่า การควบคุมทางฮาร์ดแวร์

$C/\overline{T}$ (Timer or Counter selector) : ใช้เลือกลักษณะการทำงานของไทเมอร์/เคาน์เตอร์

"0" เลือกให้ทำงานเป็นไทเมอร์ โดยใช้สัญญาณอินพุทจากสัญญาณนาฬิกาภายในไมโครคอนโทรลเลอร์

"1" เลือกให้ทำงานเป็นเคาน์เตอร์ โดยรับสัญญาณอินพุทจากภายนอกที่เข้ามาทางขา T0 และ T1

M1, M0 (Mode selector bit): ใช้เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์

"00" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ 13 บิต

"01" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ 16 บิต

"10" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ขนาด 8 บิตแบบตั้งค่าอัตโนมัติ

"11" สำหรับไทเมอร์ 0 เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์แยกส่วน โดยแยกออกเป็นไทเมอร์/เคาน์เตอร์ 8 บิต 2 ตัว รีจิสเตอร์ TL0 จะได้รับการควบคุมการเปิดปิดจากบิต TR0 ในรีจิสเตอร์ TCON และรีจิสเตอร์ TH0 ซึ่งเป็นไทเมอร์/เคาน์เตอร์ 8 บิตอีกตัวหนึ่ง จะได้รับการควบคุมจากบิต TR1 ในรีจิสเตอร์ TCON

สำหรับไทเมอร์ 1 เป็นการสั่งให้ไทเมอร์/เคาน์เตอร์ 1 หยุดทำงาน(ดิสเอเบิล)

5.3.4 โหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

ไทเมอร์ 0 และ ไทเมอร์ 1 สามารถเลือกโหมดการทำงานได้ 4 โหมดคือ โหมด 0 : ไทเมอร์/เคาน์เตอร์ 13 บิต (13 bit timer/counter), โหมด 1 : ไทเมอร์/เคาน์เตอร์ 16 บิต (16 bit timer/counter), โหมด 2: ตั้งค่าอัตโนมัติขนาด 8 บิต (8 bit auto-reload timer/counter) และโหมด 3:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์/เคาน์เตอร์แยกส่วน (split timer/counter) หรืออาจเรียกว่าโหมดไทเมอร์/เคาน์เตอร์ 8 บิตก็ได้ ในขณะที่ไทเมอร์ 2 มีโหมดการทำงาน 3 โหมดคือ โหมดแคปเจอร์หรือตรวจจับสัญญาณ (capture), โหมดตั้งค่าอัตโนมัติ (auto-reload) และโหมดกำเนิดอัตราเร็วในการสื่อสารข้อมูลอนุกรมหรืออัตราบอด (baud rate generator)

เนื่องจากในภาคการศึกษานี้ จะใช้เพียง ไทเมอร์ 0 เป็นเคาน์เตอร์ และไทเมอร์ 1 เป็นไทเมอร์ จะไม่ได้ใช้ไทเมอร์ 2 ดังนั้นจะอธิบายการทำงานในแต่ละโหมดของไทเมอร์ 0 และไทเมอร์ 1 เท่านั้น จะไม่อธิบายโหมดการทำงานของไทเมอร์ 2

การเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 สามารถกระทำได้ที่รีจิสเตอร์ TCON และ TMOD ร่วมกัน โดยรีจิสเตอร์ TCON ใช้ในการอีนาเบิลหรือดิสเอเบิลไทเมอร์/เคาน์เตอร์ ส่วน TMOD ใช้ในการเลือกโหมดและลักษณะการทำงาน และในการอธิบายโหมดการทำงานนี้จะใช้ไทเมอร์/เคาน์เตอร์ 1 ในการอธิบาย ซึ่งโหมดการทำงานจะมีลักษณะเหมือนกับไทเมอร์/เคาน์เตอร์ 0 ทุกประการ เพียงแต่เปลี่ยนรีจิสเตอร์และขาสัญญาณที่เกี่ยวข้องให้เป็นของไทเมอร์/เคาน์เตอร์ 0

1) การทำงานในโหมด 0 : ไทเมอร์/เคาน์เตอร์ 13 บิต

การทำงานในโหมดนี้จะเป็นการกำหนดให้ใช้งานรีจิสเตอร์ TL1 เพียง 5 บิต และ TH1 8 บิต โดย TL1 จะทำหน้าที่คล้ายเป็นปรีสเกลเลอร์หาร 32 สัญญาณอินพุทสำหรับการนับจะเลือกจากสัญญาณนาฬิกาภายในหรือภายนอกผ่านทางขา T1 ขึ้นอยู่กับการควบคุมของบิต C/T และ GATE ในรีจิสเตอร์ TMOD, บิต TR1 ในรีจิสเตอร์ TCON และสถานะของลอจิกที่ขาอินพุท INT1

เมื่อ TL1 นับครบ 32 คือจาก 0-31 ก็จะส่งสัญญาณไปยัง TH1 เพื่อทำการเพิ่มค่า ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 13 บิต เมื่อทำการนับครบรอบ ก็จะทำการเซตบิต TF1 ในรีจิสเตอร์ TCON

2) การทำงานในโหมด 1: ไทเมอร์/เคาน์เตอร์ 16 บิต

การทำงานในโหมดนี้จะคล้ายกับโหมด 0 แต่จะใช้งานรีจิสเตอร์ TL1 ครบ 8 บิต และ TH1 ครบ 8 บิต ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 16 บิต คือ 0000H-FFFFH เมื่อทำการนับครบรอบ ค่าของการนับจะเปลี่ยนจาก FFFFH เป็น 0000H ก็จะทำการเซตบิต TF1 ในรีจิสเตอร์ TCON

3) การทำงานในโหมด 2: ไทเมอร์/เคาน์เตอร์ 8 บิตแบบตั้งค่าอัตโนมัติ

การทำงานในโหมดนี้จะแยกรีจิสเตอร์ไทเมอร์ออกเป็น 2 ตัว ตัวละ 8 บิต โดยรีจิสเตอร์ TL1ทำหน้าที่เป็นตัวนับค่า ส่วน TH1 ใช้ในการเก็บค่าเริ่มต้นของการนับ เมื่อเริ่มต้นการทำงาน ค่าของรีจิสเตอร์ TH1 จะถูกส่งไปยังรีจิสเตอร์ TL1 ทำให้เมื่อเริ่มต้นการทำงานค่าของรีจิสเตอร์ TL1

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ TH1 จะเหมือนกัน เมื่อ TL1 นับถึง FFH และจะเริ่มต้นการนับรอบใหม่ จะทำการเซตบิต TF1 พร้อม ๆ กับทำการรับค่าการนับเริ่มต้นจาก TH1 ใหม่โดยอัตโนมัติ หรือเรียกกระบวนการนี้ว่า รีโหลด (reload) แม้ว่าจะมีการส่งค่าเริ่มต้นไปยัง TL1 แล้วก็ตาม ค่าของข้อมูลในรีจิสเตอร์ TH1 ก็ยังคงเป็นค่าเดิม ไม่มีการเปลี่ยนแปลง จนกว่าจะมีการกำหนดค่าใหม่ด้วยกระบวนการทางซอฟท์แวร์

4) การทำงานในโหมด 3: ไทเมอร์/เคาน์เตอร์แยกส่วนหรือไทเมอร์/เคาน์เตอร์ 8 บิต

ในโหมดนี้เป็นโหมดเดียวที่การทำงานของไทเมอร์ 0 และไทเมอร์ 1 ไม่เหมือนกัน ขออธิบายในส่วนของไทเมอร์ 1 ก่อน เมื่อเข้าสู่โหมดนี้ จะเป็นการสั่งให้ไทเมอร์/เคาน์เตอร์หยุดนับ ค่าของการนับก่อนหน้านี้จะถูกเก็บไว้ในรีจิสเตอร์ไทเมอร์ 1 มีลักษณะการทำงานเหมือนกับการดิสเอเบิลไทเมอร์/เคาน์เตอร์ 1 ด้วยการเคลียร์บิต TF1 ในรีจิสเตอร์ TCON

ส่วนการทำงานของไทเมอร์ 0 ในโหมดนี้จะแยกรีจิสเตอร์ไทเมอร์ 0 ออกเป็น 2 ตัว ตัวละ 8 บิต คือรีจิสเตอร์ TL0 และ TH0 โดยแยกการทำงานออกจากกัน รีจิสเตอร์ TL0 สามารถเลือกการทำงานได้เหมือนกับไทเมอร์/เคาน์เตอร์ตามปกติ ส่วนรีจิสเตอร์ TH0 สามารถทำงานในโหมดไทเมอร์ได้เพียงอย่างเดียว กล่าวคือ สามารถรับสัญญาณอินพุทจากสัญญาณนาฬิกาภายในเพียงทางเดียวเท่านั้น แต่การแจ้งการนับเกินยังคงเหมือนเดิม หากแต่ TL0 แจ้งผ่านบิต TF0 ในขณะที่ TH0 จะแจ้งผ่านทางบิต TF1

5.4 อินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51

การอินเตอร์รัปต์ (Interrupt) เป็นชื่อเรียกกระบวนการที่เข้ามาขัดจังหวะการทำงานโดยปกติของไมโครคอนโทรลเลอร์ ซึ่งในไมโครคอนโทรลเลอร์ MCS-51 เบอร์ AT89S8252 สามารถตอบสนองการอินเตอร์รัปต์ได้จาก 6 แหล่งกำเนิด ประกอบด้วย การรับสัญญาณอินเตอร์รัปต์จากภายนอกผ่านทางขา $\overline{\text{INT0}}$ และ $\overline{\text{INT1}}$, สัญญาณอินเตอร์รัปต์จากไทเมอร์/เคาน์เตอร์ T0, T1 และ T2 และสัญญาณอินเตอร์รัปต์จากพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ในขณะที่ไมโครคอนโทรลเลอร์

5.4.1 การจัดการอินเตอร์รัปต์

เมื่อมีการอินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51 เกิดขึ้น และมีการอีนาเบิลการตอบสนองการอินเตอร์รัปต์ไว้ กระบวนการหลังจากนั้นซีพียูจะทำการกระโดดไปยังแอดเดรสในหน่วยความจำที่กำหนดไว้ เรียกตำแหน่งแอดเดรสนี้ว่า แอดเดรสอินเตอร์รัปต์เวกเตอร์ (Interrupt vector address) ดังนั้นจะต้องมีการเขียนโปรแกรมย่อยการบริการอินเตอร์รัปต์ไว้ที่แอดเดรสอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เตอร์รัปต์เวกเตอร์นี้ โดยค่าของแอดเดรสอินเตอร์รัปต์เวกเตอร์จะแตกต่างกันไปในการอินเตอร์รัปต์แบบต่าง ๆ ดังมีรายละเอียดดังนี้

การอินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT0}}$ มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0003H

การอินเตอร์รัปต์จากไทเมอร์ 0 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 000BH

การอินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT1}}$ มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0013H

การอินเตอร์รัปต์จากไทเมอร์ 1 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 001BH

การอินเตอร์รัปต์จากพอร์ตอนุกรม มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0023H

การอินเตอร์รัปต์จากไทเมอร์ 2 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 002BH

5.4.2 การเขียนโปรแกรมย่อยบริการอินเตอร์รัปต์

มีหลักการโดยทั่ว ๆ ไป ดังนี้

1. ต้องเริ่มต้นด้วยแอดเดรสอินเตอร์รัปต์เวกเตอร์เสมอ เพื่อให้การตรวจสอบการทำงานทำได้ง่าย และแยกส่วนของโปรแกรมย่อยนี้ออกจากโปรแกรมหลักหรือโปรแกรมย่อยอื่น ๆ อย่างชัดเจนด้วยคำสั่ง ORG xxxxH (ค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์)

2. เมื่อเข้าสู่โปรแกรมย่อย ควรเก็บค่าของรีจิสเตอร์หรือแฟลกที่ใช้แสดงสถานะต่าง ๆ ซึ่งต้องมีการใช้งานในโปรแกรมย่อยบริการอินเตอร์รัปต์นี้ไว้ในสแต็กเสียก่อน เพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้นต่อการทำงานของทั้งโปรแกรมบริการอินเตอร์รัปต์นี้และโปรแกรมหลัก ด้วยคำสั่ง PUSH

3. เมื่อเขียนโปรแกรมบริการอินเตอร์รัปต์เรียบร้อยแล้ว ให้ทำการคืนค่าของรีจิสเตอร์ที่นำมาใช้ในโปรแกรมบริการอินเตอร์รัปต์ด้วยคำสั่ง POP ยกเว้นรีจิสเตอร์ที่ต้องการนำผลการกระทำในโปรแกรมบริการอินเตอร์รัปต์นี้ไปใช้งาน ซึ่งในทางปฏิบัติจริง ไม่พบมากนัก และไม่แนะนำให้เขียนโปรแกรมในลักษณะนี้

4. ปิดท้ายโปรแกรมย่อยบริการอินเตอร์รัปต์ด้วยคำสั่ง RETI เสมอ

5.4.3 รีจิสเตอร์ที่เกี่ยวข้องกับการบริการอินเตอร์รัปต์

การอินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช มีรีจิสเตอร์ที่ต้องเกี่ยวข้องอยู่ 2 ตัว ดังมีรายละเอียดดังต่อไปนี้

1) รีจิสเตอร์อีนาเบิลการอินเตอร์รัปต์ หรือ IE (Interrupt Enable register)

มีแอดเดรสอยู่ที่ A8H ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษ หรือ SFR มีขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต โดยสามารถเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ใช้ในการอีนาเบิลการตอบสนองการอินเตอร์รัปต์ในแบบต่าง ๆ มีรายละเอียดการทำงานดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

EA (Global enable/disable interrupt bit): ใช้ในการอีนาเบิลและดิสเอเบิลการตอบสนองการอินเตอร์รัปต์ทั้งหมด

"0" ดิสเอเบิลการอินเตอร์รัปต์ นั่นคือ กำหนดให้ไมโครคอนโทรลเลอร์ไม่ตอบสนองการอินเตอร์รัปต์ ไม่ว่าจะมาจากแหล่งกำเนิดใดก็ตาม

"1" อีนาเบิลการอินเตอร์รัปต์ นั่นคือ กำหนดให้ไมโครคอนโทรลเลอร์สามารถตอบสนองการอินเตอร์รัปต์จากแหล่งกำเนิดต่าง ๆ

นั่นคือ ถ้าต้องการให้ไมโครคอนโทรลเลอร์ตอบสนองการอินเตอร์รัปต์ไม่ว่าจะแหล่งกำเนิดใด จะต้องเซตบิตนี้ก่อนเสมอ

ET2 (Timer 2 interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวหรือการแคปเจอร์ ในไทเมอร์/เคาน์เตอร์ 2 จะมีเฉพาะในเบอร์ AT89C52 และอนุกรม AT89Sxx เท่านั้น

ES (Serial port interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการรับหรือส่งข้อมูลทางพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

ET1 (Timer 1 interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 1

EX1 (External interrupt 1 enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{INT1}$

ET0 (Timer 0 interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 0

EX0 (External interrupt 0 enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{INT0}$

สำหรับบิต 6 ของรีจิสเตอร์ IE ไม่มีการใช้งาน ต้องกำหนดให้เป็น "0" เสมอ

2) รีจิสเตอร์จัดลำดับความสำคัญการตอบสนองการอินเตอร์รัปต์ หรือ IP (Interrupt Priority register)

มีแอดเดรสอยู่ที่ B8H ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษหรือ SFR มีขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต โดยสามารถเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ใช้ในการเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ไม่ว่าในรูปแบบใดๆ ทั้งสิ้น ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับความสำคัญของการตอบสนองการอินเตอร์รัปต์ว่า ต้องการให้ตอบสนองสัญญาณอินเตอร์รัปต์จากแหล่งกำเนิดใดเป็นลำดับก่อนหลัง ถ้าต้องการให้การอินเตอร์รัปต์จากแหล่งกำเนิดใดมีความสำคัญสูงสุด ให้กำหนดที่บิตนั้นเป็น "1" มีรายละเอียดของรีจิสเตอร์ IP ดังนี้

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

PT2 (Timer 2 interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวหรือการแคปเจอร์ในไทเมอร์/เคาน์เตอร์ 2 , จะมีเฉพาะในเบอร์ AT89C52 และอนุกรม AT89Sxx เท่านั้น

PS (Serial port interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการรับหรือส่งข้อมูลทางพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

PT1 (Timer 1 interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 1

PX1 (External interrupt 1 priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{INT1}$

PT0 (Timer 0 interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 0

PX0 (External interrupt 0 priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{INT0}$

สำหรับบิต 6 และ 7 ของรีจิสเตอร์ IP ไม่มีการใช้งาน ต้องกำหนดให้เป็น "0" เสมอ

5.4.4 แหล่งกำเนิดสัญญาณอินเตอร์รัปต์

ในที่นี้จะกล่าวถึงสัญญาณอินเตอร์รัปต์จากไทเมอร์/เคาน์เตอร์ T0 และ T1 ตามที่ได้ใช้งานเท่านั้น

สัญญาณอินเตอร์รัปต์จกาไทเมอร์/เคาน์เตอร์ T0 และ T1

แหล่งกำเนิดสัญญาณอินเตอร์รัปต์นี้จัดเป็นแหล่งกำเนิดสัญญาณอินเตอร์รัปต์ภายในแบบหนึ่ง โดยใช้การเกิดโอเวอร์โฟลวจากการนับค่าในไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 เมื่อไทเมอร์ 0 เกิดการโอเวอร์โฟลวก็จะทำการเซตบิต TF0 ในรีจิสเตอร์ TCON และถ้าไทเมอร์ 1 เกิดโอเวอร์โฟลวบิต TF1 ในรีจิสเตอร์ TCON จะได้รับการเซตเช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตามการอินเตอร์รัปต์แบบนี้ จะเกิดขึ้นหรือมีการตอบสนองก็ต่อเมื่อมีการอีนาเบิลการอินเตอร์รัปต์ โดยการเซตบิต EA, ET0 และ ET1 ในรีจิสเตอร์ IE

5.4.5 ลำดับความสำคัญของการอินเตอร์รัปต์

การกำเนิดสัญญาณอินเตอร์รัปต์ในในไมโครคอนโทรลเลอร์ MCS-51 เบอร์ AT89S8252 มีได้จาก 6 แหล่ง ดังนั้นจึงต้องมีการจัดลำดับความสำคัญ ในกรณีที่เกิดการอินเตอร์รัปต์ขึ้นพร้อม ๆ กันจากหลายแหล่งกำเนิด โดยสามารถกำหนดลำดับความสำคัญได้ที่รีจิสเตอร์ IP

อย่างไรก็ตาม ลำดับความสำคัญของการอินเตอร์รัปต์โดยปกติหรือในกรณีกำหนดข้อมูลในรีจิสเตอร์ IP ให้เป็น "1" ทุกบิต (ยกเว้นบิต 6 และ 7) จะเรียงลำดับจากความสำคัญสูงสุดไปจนถึงต่ำสุดดังนี้

1. อินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT0}}$ หรือการเซตของบิต IE0
2. อินเตอร์รัปต์จากไทเมอร์ 0 หรือการเซตของบิต TF0
3. อินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT1}}$ หรือการเซตของบิต IE1
4. อินเตอร์รัปต์จากไทเมอร์ 1 หรือการเซตของบิต TF1
5. อินเตอร์รัปต์จากพอร์ตอนุกรม หรือการเซตของบิต RI หรือ TI
6. อินเตอร์รัปต์จากไทเมอร์ 2 หรือการเซตของบิต TF2 หรือ EXF2

5.5 การขับโมดูลแสดงผลแบบผลึกเหลว (LCD Module)

5.5.1 รายละเอียดเกี่ยวกับโมดูล LCD

ในโมดูล LCD จะมีส่วนประกอบหลัก ๆ 3 ส่วน ดังนี้

ตัวแสดงผล (Display) ภายในเป็นผลึกเหลวที่สามารถแสดงผลให้เห็นโดยอาศัยแสงจากภายนอก ดังนั้นจึงต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอ LCD

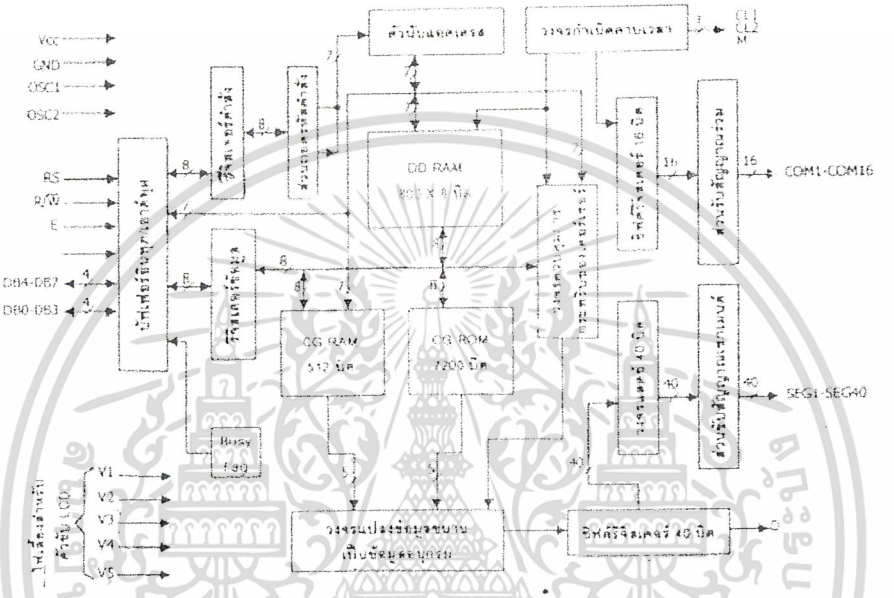
ตัวควบคุม (Controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของโมดูล LCD เช่น ลบจอภาพ แสดงตัวอักษร หรือเลื่อนเคอร์เซอร์ เป็นต้น ตัวควบคุมนี้ใช้ชิปควบคุมโดยเฉพาะ ชิปที่นิยมใช้คือ *เบอร์ HD44780* และ *HD61830* โดย HD44780 จะใช้ควบคุม LCD แบบอักขระ ส่วน HD61830 ใช้ควบคุม LCD แบบกราฟิก

ตัวขับ (Driver) เป็นตัวรับสัญญาณจากตัวควบคุมมาขับให้ตัวแสดงผลแสดงข้อมูลตามที่กำหนด ชิปที่ใช้ทำหน้าที่เป็นตัวขับนี้ได้แก่ *เบอร์ HD44100H* และ *MSM5259* เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5.2 โครงสร้างภายในของตัวควบคุมโมดูล LCD

ในการใช้งานโมดูล LCD จำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้าง และคำสั่งที่ใช้ในการควบคุมให้ดีเสียก่อน ในรายงานนี้ขอยกตัวอย่างโมดูล LCD แบบอักขระ เพราะสามารถเข้าใจได้ง่าย ให้รูปที่ 1 เป็นบล็อกไดอะแกรมภายในของชิปควบคุม LCD เบอร์ HD44780 ซึ่งใช้ในโมดูล LCD แบบอักขระ ประกอบด้วย



รูปที่ 5.3 ไดอะแกรมการทำงานของโมดูล LCD แบบอักขระ

บัฟเฟอร์อินพุทเอาท์พุท เป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก เพื่อที่จะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

รีจิสเตอร์คำสั่ง (Instruction Register: IR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอก เพื่อนำไปควบคุมการแสดงผล

รีจิสเตอร์ข้อมูล (Data Register: DR) เป็นรีจิสเตอร์ใช้รับข้อมูลจากอุปกรณ์ภายนอก เพื่อถ่ายทอดต่อไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษรเพิ่มเติมในแรมเก็บตัวอักษร

แรมเก็บข้อมูลแสดงผล (Display Data RAM: DDRAM) เป็นหน่วยความจำแรมทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตาราง (Look up-lable) ของตัวอักษรที่เก็บไว้ในหน่วยความจำรอม และแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รอมเก็บตัวอักษร (Character Generator ROM: CGROM) เป็นหน่วยความจำรอมที่ใช้เก็บข้อมูลตัวอักษรหรือสัญลักษณ์ที่สามารถอ่านออกไปแสดงที่ตัวแสดงผลได้ มีขนาด 7,200 บิต โดยจะถูกอ่านด้วยค่าของข้อมูลใน DDRAM

แรมเก็บตัวอักษร (Character Generator RAM: CGRAM) เป็นหน่วยความจำแรมที่ใช้เก็บอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512 บิต การเขียนและอ่านค่าไปใช้นั้นทำได้เช่นเดียวกับ CGROM คือ เขียนข้อมูลลงใน DDRAM แล้วตัวควบคุมจะมาอ่านค่าจาก CGRAM เอง

แฟลก BUSY เป็นส่วนที่ทำหน้าที่แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูลหรือคำสั่งหรือไม่ ดังนั้นก่อนการส่งข้อมูลหรือคำสั่งมายังตัวควบคุมต้องตรวจสอบสถานะของแฟลก BUSY นี้เสียก่อน

5.5.3 โมดูล LCD ขนาด 24 ตัวอักษร 2 บรรทัด (LCD 24x2)

สำหรับโมดูล LCD ที่นำมาใช้ในการทำโครงงานนี้ เป็นขนาด 24 ตัวอักษร 2 บรรทัด เนื่องจากมีขนาดที่เหมาะสมกับโครงงานนี้

โมดูล LCD ขนาด 24x2 มีขาต่อใช้งานทั้งสิ้น 14 ขา สำหรับรายละเอียดการทำงานของแต่ละขามีดังนี้

- V_{SS} (ขา 1) : ต่อกราวด์
- V_{DD} (ขา 2) : ต่อไฟเลี้ยง +5 โวลท์
- V_O (ขา 3) : เป็นขาอินพุทรับแรงดันเพื่อปรับความเข้มของการแสดงผล
- RS (ขา 4) : เป็นขาอินพุทใช้ในการแยกชนิดของข้อมูลที่ทำการประมวลผลในขณะนั้นว่าเป็นคำสั่งสำหรับรีจิสเตอร์ IR หรือเป็นข้อมูลสำหรับรีจิสเตอร์ DR โดยถ้าขานี้เป็น "0" ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขานี้เป็น "1" ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับการแสดงผล
- $R/\overline{W}$ (ขา 5) : เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับ LCD ถ้าเป็น "0" เป็นการกำหนดให้เขียนข้อมูล แต่ถ้าเป็น "1" จะเป็นการอ่านข้อมูล
- E (ขา 6) : เป็นขาอีนาเบิล LCD ให้ทำงาน
- D0-D7 (ขา7-14) : เป็นขาที่ใช้เป็นทางผ่านของข้อมูลระหว่าง LCD กับอุปกรณ์ภายนอกขนาด 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5.4 คำสั่งควบคุมโมดูล LCD

ในการเขียนคำสั่งลงในตัวควบคุม แน่นอนว่าต้องกำหนดให้ขา RS และ R/$\overline{W}$ เป็น "0" แล้วเขียนคำสั่งตามไป คำสั่งควบคุมโมดูล LCD ของชิปควบคุม HD44780 ที่สำคัญมี 10 คำสั่งดังนี้

1) คำสั่งเคลียร์ตัวแสดงผล (Clear display)

มีข้อมูลคำสั่งเป็น 01H เป็นคำสั่งที่ใช้เขียนข้อมูลช่องว่าง หรือ Space เข้าไปใน DDRAM ทั้งหมด เมื่อตัวควบคุมแอ็กซิคิวต์คำสั่งนี้ จะทำการกำหนดแอดเดรสของ DDRAM เป็น 0 เคอร์เซอร์จะกลับไปอยู่ที่ตำแหน่งซ้ายมือสุดของจอแสดงผล แล้วเซตบิต I/D (ซึ่งจะกล่าวถึงภายหลัง) ให้เป็น "1"

2) คำสั่ง return home

ต้องกำหนดให้บิต 1 ของข้อมูลเป็น "1" เป็นคำสั่งให้เคอร์เซอร์เคลื่อนที่กลับไปยังตำแหน่งซ้ายสุดของจอแสดงผล แต่ข้อมูลบนจอแสดงผลไม่เปลี่ยนแปลง นั่นคือ ข้อมูลคำสั่งของคำสั่งนี้จะเป็น 02H หรือ 03H ก็ได้

3) คำสั่งเลือกโหมดการป้อนข้อมูล (Entry mode Set)

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

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

บิต S เป็นบิตที่ใช้ในการกำหนดลักษณะของการแสดงผล เมื่อมีการป้อนข้อมูล ถ้าหากบิต S เป็น "1" เมื่อเกิดข้อมูลใหม่บนจอแสดงผล ตัวเคอร์เซอร์จะอยู่กับที่ แต่ตัวอักษรข้อมูลเดิมจะถูกดันไปทางซ้าย แต่ถ้าหากบิตนี้เป็น "0" เมื่อเกิดข้อมูลใหม่ตัวเคอร์เซอร์จะเลื่อนไปทางขวามือ

บิต I/D เป็นบิตที่ใช้ในการกำหนดว่า เมื่อเขียนหรืออ่านข้อมูลแล้ว ทำให้แอดเดรสของ DDRAM เพิ่มขึ้นหรือลดลงหนึ่งแอดเดรส โดยถ้าบิตนี้เป็น "1" แอดเดรสของ DDRAM จะเพิ่มขึ้น แต่ถ้าเป็น "0" แอดเดรสจะลดลง

ดังนั้น ข้อมูลคำสั่งที่เกิดขึ้นสำหรับคำสั่งนี้ได้แก่ 04H-07H(4 ข้อมูลคำสั่ง) และที่ใช้บ่อยคือ 06H หมายถึง *กำหนดให้เมื่อเกิดข้อมูลใหม่ เคอร์เซอร์จะเลื่อนไปทางขวามือ และแอดเดรสของ DDRAM เพิ่มขึ้น*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) คำสั่งควบคุมการแสดงผล

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	0	1	D	C	B

บิต D ใช้ควบคุมการเปิดปิดจอแสดงผล ถ้าบิตนี้เป็น "1" จะเป็นการเปิดจอแสดงผล ถ้าเป็น "0" จะเป็นการปิดจอแสดงผล

บิต C ใช้ควบคุมการแสดงตัวเคอร์เซอร์บนจอแสดงผล ถ้าต้องการให้มีเคอร์เซอร์แสดงผลบนจอแสดงผล ต้องกำหนดให้บิตนี้เป็น "1" ถ้ากำหนดให้เป็น "0" จะเป็นการปิดเคอร์เซอร์ หรือไม่แสดงเคอร์เซอร์

บิต B ใช้ควบคุมการกะพริบของเคอร์เซอร์ ถ้าบิตนี้เป็น "1" เคอร์เซอร์จะกะพริบ

ดังนั้นจะมีข้อมูลคำสั่งได้ตั้งแต่ 08H-0FH (8 รูปแบบคำสั่ง) ที่ใช้บ่อยคือ 0CH *เป็นการสั่งให้เปิดจอแสดงผล แต่ไม่แสดงเคอร์เซอร์ และ* 0FH *เป็นการสั่งให้เปิดจอแสดงผล แสดงเคอร์เซอร์ และ สั่งให้เคอร์เซอร์กะพริบ*

5) คำสั่งควบคุมการเลื่อนเคอร์เซอร์และข้อมูลตัวอักษร

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	1	S/C	R/L	*	*

การควบคุมการเลื่อนเคอร์เซอร์และตัวอักษรบนจอแสดงผลขึ้นอยู่กับการกำหนดบิต S/C และ R/L ซึ่งสามารถสรุปได้ดังนี้

S/C	R/L	ลักษณะการเลื่อน	ข้อมูลคำสั่ง
0	0	เลื่อนเคอร์เซอร์ไปทางซ้าย	10H-13H
0	1	เลื่อนเคอร์เซอร์ไปทางขวา	14H-17H
1	0	เลื่อนตัวอักษรใหม่ไปทางซ้าย	18H-1BH
1	1	เลื่อนตัวอักษรใหม่ไปทางขวา	1CH-1FH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) คำสั่งกำหนดฟังก์ชั่นการทำงาน

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	1	DL	N	F	*	*

บิต DL ใช้กำหนดจำนวนบิตที่ใช้ติดต่อส่งผ่านข้อมูล ถ้าบิตนี้เป็น "0" จะเป็นการติดต่อแบบ 4 บิต แต่ถ้าเป็น "1" จะเป็นแบบ 8 บิต

บิต N ใช้กำหนดจำนวนบรรทัดของการแสดงผล ถ้าเป็น "0" จะแสดงผล 1 บรรทัด ถ้าเป็น "1" จะแสดงผล 2 บรรทัด ในกรณีที่จอแสดงผลสามารถแสดงได้มากกว่า 2 บรรทัด และต้องการให้แสดงผลมากกว่า 2 บรรทัด ก็กำหนดบิต N นี้ให้เป็น "1"

บิต F ใช้เลือกความละเอียดของตัวอักษรให้การแสดงผล ถ้าบิตนี้เป็น "0" จะเป็นการแสดงผลแบบ 5x7 จุด และถ้าเป็น "1" จะแสดงผลเป็นแบบ 5x10 จุด

ข้อมูลคำสั่งที่ใช้บ่อยคือ 38H *เป็นการกำหนดให้โมดูล LCD ทำงานในแบบ 8 บิต แสดงผล 2 บรรทัด และเลือกความละเอียดเป็น 5x7 จุด*

จุดที่น่าสังเกตคือ โมดูล LCD แบบ 16 ตัวอักษร 1 บรรทัด แม้จะมีบรรทัดการแสดงผลเพียง 1 บรรทัด แต่จะต้องกำหนด N ให้เป็น "1" เนื่องจากแอดเดรสของ DDRAM แบ่งเป็น 2 ช่อง คือ 00H และ 40H

7) คำสั่งเลือกแอดเดรสของ CGRAM

เมื่อต้องการกำหนดแอดเดรสของ CGRAM ต้องกำหนดให้บิต 7 เป็น "0" บิต 6 เป็น "1" ส่วนอีก 6 บิตที่เหลือจะแทนด้วยค่าแอดเดรสของ CGRAM จะต้องทำการกำหนดแอดเดรสด้วยคำสั่งนี้ ก่อนที่จะอ่านหรือเขียนข้อมูลให้ CGRAM โดยแอดเดรสของ CGRAM อยู่ระหว่าง 00H-3FH

8) คำสั่งเลือกแอดเดรสของ DDRAM

ใช้ในการเลือกแอดเดรสของ DDRAM ก่อนที่จะทำการอ่านหรือเขียนข้อมูล โดยบิต 7 ต้องเป็น "1" และข้อมูลอีก 7 บิตที่เหลือจะเป็นค่าแอดเดรสของ DDRAM ซึ่งแอดเดรสของ DDRAM จะอยู่ระหว่าง 00H-67H ทั้งนี้จำนวนแอดเดรสยังขึ้นกับการกำหนดสถานะที่บิต N ด้วย หากบิต N เป็น "0" แอดเดรสของ DDRAM จะอยู่ระหว่าง 00H-4FH และถ้าบิต N เป็น "1" แอดเดรสของ DDRAM จะมี 2 ช่วงคือ 00H-27H และ 40H-67H ซึ่งในโครงงานนี้ได้ใช้ LCD โมดูลแบบ 24 ตัวอักษร 2 บรรทัด ดังนั้นเมื่อให้บิต N เป็น "1" จะได้แอดเดรสของ DDRAM ในบรรทัดแรกเป็น 00H-17H และในบรรทัดที่สองเป็น 40H-57H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9) คำสั่งอ่านแฟลก BUSY และแอดเดรส

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
BF	A	A	A	A	A	A	A
	แอดเดรสไบต์สูง			แอดเดรสไบต์ต่ำ			

เป็นคำสั่งที่ใช้อ่านแฟลก BUSY (BF) โดยแฟลกนี้จะเป็นตัวบอกสถานะของตัวควบคุม LCD ว่าพร้อมจะรับข้อมูลอยู่หรือไม่ ถ้าหากบิต BF เป็น "0" แสดงว่าตัวควบคุม LCD พร้อมรับข้อมูลหรือคำสั่ง แต่ถ้าเป็น "1" แสดงว่า ขณะนี้ตัวควบคุม LCD ยังอยู่ในกระบวนการทำงานภายในหรือกำลังประมวลผลข้อมูลอยู่ ยังไม่พร้อมรับข้อมูลหรือคำสั่ง

เมื่อต้องการอ่านแฟลกต้องกำหนดให้ขา R/$\overline{W}$ เป็น "1" ด้วย แต่สัญญาณที่ RS ยังต้องเป็น "0" อยู่เพราะข้อมูลนี้เป็นข้อมูลคำสั่ง

นอกจากนี้ ยังใช้เป็นคำสั่งอ่านข้อมูลแอดเดรสของ CGRAM และ DDRAM ด้วย โดยบิต 0 ถึง บิต 6 เป็นค่าข้อมูลของแอดเดรสที่ต้องการอ่าน

นอกจากนี้ยังใช้เป็นคำสั่งอ่านข้อมูลแอดเดรสของ CGRAM และ DDRAM ด้วย โดยบิต 0 ถึง บิต 6 เป็นค่าข้อมูลของแอดเดรสที่ต้องการอ่าน

5.5.5 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD

ในการเขียนข้อมูลเพื่อควบคุมให้โมดูล LCD แสดงผลตามที่ผู้ใช้งานต้องการ ต้องส่งคำสั่ง (instruction) แล้วกำหนดโหมดการทำงานให้แก่โมดูล LCD ก่อน จากนั้นจึงค่อยส่งข้อมูล (data) ที่ต้องการแสดงผล เนื่องจากบัสข้อมูลของโมดูล LCD มี 8 เส้นคือ D0-D7 และใช้เป็นทางผ่านของทั้งคำสั่งและข้อมูล ดังนั้นในการส่งคำสั่งและข้อมูลจึงต้องอาศัยการกำหนดสัญญาณลอจิกที่ขา RS ถ้าหากที่ขา RS ได้ลอจิก "0" หมายความว่า ข้อมูลที่ป้อนให้แก่โมดูล LCD ขณะนั้นเป็นคำสั่ง ในทางตรงข้าม หากขา RS ได้รับลอจิก "1" ข้อมูลที่ป้อนให้ขณะนั้นเป็นข้อมูลที่ใช้ในการแสดงผล

เมื่อต้องการเขียนหรืออ่านข้อมูลใน CGRAM และ DDRAM เริ่มต้นต้องกำหนดแอดเดรสที่ต้องการอ่านหรือเขียนก่อน โดยใช้คำสั่งเลือกแอดเดรส จากนั้นกำหนดให้ขา RS เป็น "1" เพื่อแจ้งให้ตัวควบคุมภายในโมดูล LCD ทราบว่าข้อมูลที่ปรากฏต่อไปนี้เป็นข้อมูลปกติไม่ใช่คำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่ต้องการอ่านข้อมูลต้องกำหนดให้ขา $R/\overline{W}$ เป็น "1" ข้อมูลขนาด 8 บิต (หรือ 4 บิต) ก็จะปรากฏบนบัสข้อมูล โดยข้อมูลที่อ่านออกมาได้จะเป็นข้อมูลจากแอดเดรสของ CGRAM หรือ DDRAM ตามที่ต้องการ

ในกรณีที่ต้องการเขียนข้อมูล เมื่อกำหนดแอดเดรสและป้อนลอจิก "1" ให้ขา RS แล้ว ต้องกำหนดให้ขา $R/\overline{W}$ เป็น "0" ข้อมูลที่อยู่บนบัสข้อมูลจะถูกเขียนลงในรีจิสเตอร์ DR จากนั้นจึงถ่ายทอดลงใน DDRAM ต่อไป

5.5.6 จังหวะการทำงานของ LCD โมดูล

ในการติดต่อกับโมดูล LCD จะต้องมีการหน่วงเวลาหลังจากที่ทำการส่งรหัสคำสั่งหรือข้อมูล เนื่องจากต้องรอให้คอนโทรลเลอร์ภายใน LCD โมดูล แปลความหมายของรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นจึงจะรับข้อมูลหรือดำเนินการต่อไป

ดังนั้น ในการใช้งานโมดูล LCD ผู้เขียนโปรแกรมต้องมีโปรแกรมเพื่อหน่วงเวลารอให้โมดุล LCD พร้อมทำงานด้วย โดยเมื่อเริ่มจ่ายไฟให้แก่โมดูล LCD ต้องรอประมาณ 10 มิลลิวินาที เพื่อให้โมดูล LCD ทำการเตรียมความพร้อมหรืออินนิเชียล (initial) หลังจากนั้นก็จะกำหนดลอจิกให้แก่ขา RS ของโมดูล LCD แล้วต้องหน่วงเวลาอีกประมาณ 2 มิลลิวินาที เพื่อให้คอนโทรลเลอร์ใน LCD โมดูล แปลความหมายของลอจิกที่ RS ว่า ข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่งหรือเป็นข้อมูลที่ต้องการแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0-D7 (กรณีทำงานในโหมด 8 บิต) ขั้นตอนต่อไปจะเป็นการส่งสัญญาณพัลส์ไปที่ขา E เพื่ออีนาเบิลโมดูล LCD ให้รับข้อมูลจากบัสข้อมูลเข้าไป โดยพัลส์ที่ป้อนเข้าที่ขา E ของโมดูล LCD ต้องเป็นพัลส์ขอบขาขึ้น จากนั้นทำการหน่วงเวลา 2 มิลลิวินาที

ทั้งหมดที่กล่าวมา คือ ขั้นตอนและจังหวะในการทำงาน 1 รอบของโมดูล LCD จะเห็นได้ว่ามีโปรแกรมย่อยที่สำคัญอยู่ 3 โปรแกรมย่อยคือ โปรแกรมอินนิเชียล LCD, โปรแกรมหน่วงเวลา และโปรแกรมย่อยการส่งพัลส์เพื่ออีนาเบิลโมดูล LCD

5.6 การขยายพอร์ตอินพุทเอาท์พุทด้วยอุปกรณ์ระบบบัส I^2C

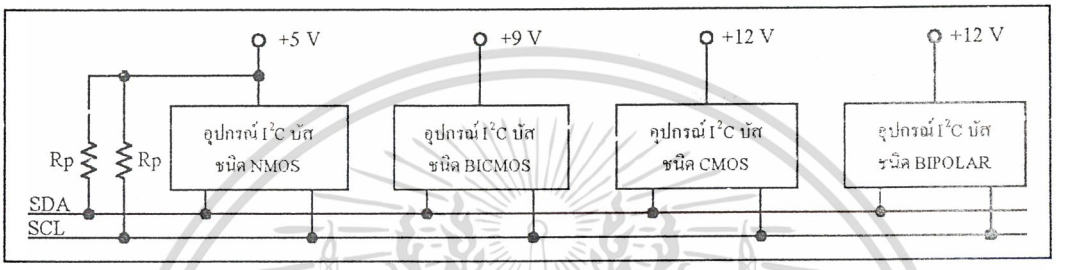
5.6.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C

I^2C ย่อมาจาก Inter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซีโดยบัส I^2C ซึ่งได้รับการพัฒนาขึ้นโดยฟิลิปส์ (Phillips) ด้วยจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ สั่งงาน และควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่ง คือ สายข้อมูล อีกเส้นหนึ่ง คือ สายสัญญาณนาฬิกาที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัส I^2C ทำได้ง่ายมาก เพียงต่อสายข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว

สายข้อมูลบนบัส I^2C มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรมหรือ SDA (Serial Data line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรมหรือ SCL (Serial Clock line) ในการอธิบายต่อไปนี้จะเรียกสายสัญญาณทั้งสองว่า สาย SDA และ SCL



รูปที่ 5.4 ผังแสดงการเชื่อมต่อของอุปกรณ์ต่าง ๆ บนระบบบัส I^2C

ในรูปที่ 5.4 แสดงผังของการเชื่อมต่ออุปกรณ์ต่าง ๆ บนบัส I^2C จะเห็นได้ว่า อุปกรณ์ที่ทำการเชื่อมต่อบนบัส I^2C มีหลากหลาย ไม่ว่าจะเป็นไอซีขยายพอร์ตอินพุทเอาท์พุท (I/O Expander), ไอซีแปลงสัญญาณอะนาลอกเป็นดิจิตอล (ADC) และแปลงสัญญาณดิจิตอลเป็นอะนาลอก (DAC), ไอซีรีลไทม์คล็อก (RTC) , ไอซีขับโมดูล LCD, หน่วยความจำอีอีพรอม และไมโครคอมพิวเตอร์

5.6.2 คุณสมบัติโดยทั่วไปของบัส I^2C

สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (Bi-directional line) ต้องมีการต่อตัวต้านทานพูลอัปกับแรงดัน +5V ไว้ตลอดเวลา เพื่อให้สายมีสถานะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งาน ทั้งยังช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสอง วงจรเอาท์พุทของอุปกรณ์ที่ต่ออยู่บนบัส I^2C ต้องมีลักษณะเป็นวงจรเดรนเปิด(Open-drain) หรือคอลเล็คเตอร์เปิด (Open-collector)

อัตราการถ่ายทอดข้อมูลบนบัส I^2C สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ (Standard mode) และสูงถึง 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง (Fast mode) อุปกรณ์ที่ต่อร่วมอยู่บนบัส I^2C จะต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 pF การเข้าถึงอุปกรณ์บนบัส I^2C ใช้ข้อมูลสำหรับการเข้าถึง 2 ค่า คือ 7 บิต (7-bits addressing) หรือ 10 บิต (10-bits addressing)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเด่นอีกประการหนึ่งของบัส I^2C คือ สามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้โดยอุปกรณ์บนบัส I^2C ตัวหนึ่งอาจใช้ไฟเลี้ยง +5V ในขณะที่อีกตัวหนึ่งใช้ไฟเลี้ยง +12V การต่อร่วมกันบนบัส I^2C สามารถกระทำได้ในลักษณะเดียวกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงเท่ากัน กล่าวคือ ให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่อตัวต้านทานพูลอัป (R_p) เข้ากับแรงดัน +5V ไว้ด้วยเสมอ ดังแสดงในรูปที่ 5.4

5.6.3 หลักการของบัส I^2C

บัส I^2C ประกอบด้วยสายสัญญาณ 2 เส้น ดังที่ได้กล่าวมาแล้ว คือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส หรือเรียกว่า โปรโตคอล (Protocol) เพื่อให้ผู้ใช้งานทราบว่า ขณะนี้อุปกรณ์ใดติดต่อกันอยู่ และอุปกรณ์ตัวใดเป็นตัวรับหรือตัวส่ง ต่อไปนี้จะขออธิบายลักษณะ หน้าที่ และนิยามของอุปกรณ์ที่ต่ออยู่บนบัส I^2C เพื่อเป็นข้อตกลงพื้นฐานก่อนที่จะอธิบายการทำงานของบัส I^2C ต่อไป

อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง (Transmitter)

อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ (Receiver)

ในอุปกรณ์บนบัส I^2C สามารถเป็นได้ทั้งตัวรับและตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับเพียงอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I^2C ที่ทำหน้าที่เป็นตัวส่งเพียงอย่างเดียว

อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I^2C เรียกว่า มาสเตอร์ (Master)

อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I^2C เรียกว่า สเลฟ (Slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I^2C คือ

(1) การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

(2) ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

5.6.4 สภาวะที่เกิดขึ้นบนบัส I^2C

มีด้วยกัน 5 สภาวะ ดังนี้

(1) **บัสว่าง** (Bus not busy) สภาวะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL เป็นลอจิกทั้งคู่ นั่นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

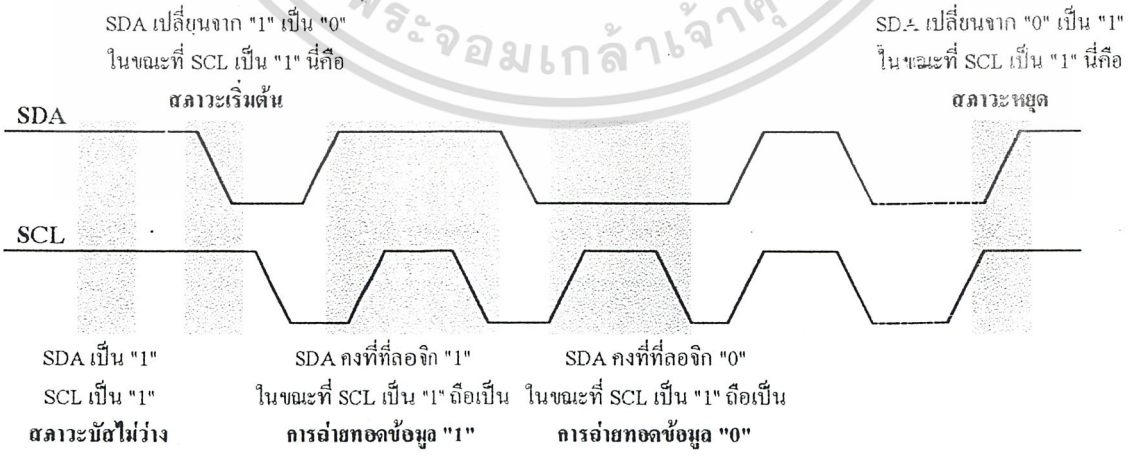
(2) **เริ่มต้นการถ่ายทอดข้อมูล** (Start data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า **สภาวะเริ่มต้น (START)**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) **หยุดการถ่ายทอดข้อมูล** (Stop data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า **สภาวะหยุด (STOP)**

(4) **ข้อมูลดำรงอยู่บนบัส** (Data valid) สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อสาย SCL เป็นลอจิกสูง สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับรู้ข้อมูลในจังหวะนั้นว่า เป็น "0" หรือ "1" ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ SDA ต้องคงที่ตลอดช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายทอดนั้นเกิดความผิดพลาดขึ้น

(5) รับรู้ข้อมูล (Acknowledge) เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิต เรียกว่า บิตรับรู้ (Acknowledge bit) มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกา เพื่อตอบสนองบิตรับรู้ที่ส่งมาจากตัวส่ง ทางด้านตัวรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้เพื่อตอบสนองให้ทราบว่าได้รับข้อมูลในแต่ละไบต์เรียบร้อยแล้ว



รูปที่ 5.5 ไดอะแกรมเวลาแสดงสถานะต่าง ๆ ในบัส I^2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 5.5 เป็นไดอะแกรมเวลาที่แสดงถึงการเกิดสภาวะต่าง ๆ บนบัส I^2C ไม่ว่าจะเป็นสภาวะบัสว่าง, เริ่มต้น, ถ่ายทอดข้อมูล, รับรู้, และหยุดการถ่ายทอดข้อมูล

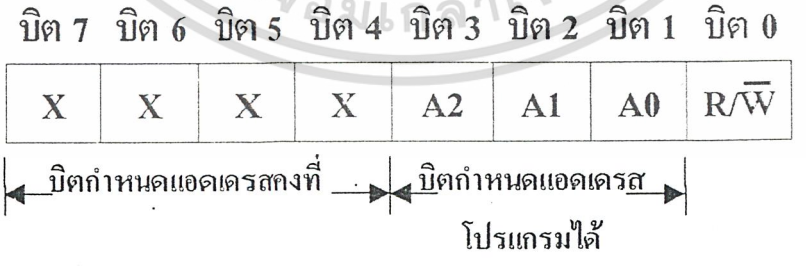
5.6.5 การทำงานบนบัส I^2C

ก่อนที่จะเริ่มต้นการถ่ายทอดข้อมูลระหว่างอุปกรณ์ต่าง ๆ ที่ต่ออยู่บนบัส ต้องมีการอ้างถึงเสียก่อนโดยการอ้างถึงอุปกรณ์บนบัส I^2C นั้นจะใช้การอ้างถึงแบบ 7 บิตหรือ 10 บิต ในกรณีที่มีอุปกรณ์ต่ออยู่บนบัสไม่มาก ใช้การอ้างถึงแบบ 7 บิตก็เพียงพอ แต่ถ้ามีอุปกรณ์ต่ออยู่บนบัสมากกว่า 127 แอดเดรส จำเป็นต้องใช้การอ้างถึงแบบ 10 บิต หลังจากที่ติดต่ออุปกรณ์แต่ละตัวได้เรียบร้อยแล้ว ก็จะเริ่มต้นการถ่ายทอดข้อมูลกันต่อไป

ดังนั้นหัวใจสำคัญในอันดับแรกของการทำงานบนบัส I^2C คือการอ้างถึงอุปกรณ์แต่ละตัว ซึ่งในที่นี้จะอธิบายรายละเอียดของการอ้างถึงทั้ง 2 รูปแบบ

การอ้างถึงแบบ 7 บิต (7 bits addressing)

ข้อมูลไบต์แรกที่เกิดขึ้นหลังจากสภาวะเริ่มต้นคือ ข้อมูลที่ใช้ในการอ้างถึงอุปกรณ์ที่ต้องการติดต่อ หรือ ข้อมูลกำหนดแอดเดรส โดยมีรูปแบบแสดงในรูปที่ 6 ใน 7 บิตบนรวมทั้งบิต MSB ด้วยจะเป็นข้อมูลแอดเดรสของอุปกรณ์สเลฟที่ต้องการติดต่อ โดยแบ่งเป็น *บิตแอดเดรสคงที่ (Fixed address bit)* จำนวน 4 บิต ซึ่งข้อมูลนี้อุปกรณ์แต่ละตัวจะถูกกำหนดมาจากผู้ผลิต ไม่สามารถเปลี่ยนแปลงแก้ไขได้ ถัดมาอีก 3 บิตเป็น *บิตกำหนดแอดเดรสที่สามารถโปรแกรมได้ (Programmable address bit)* โดยผู้ใช้งานต้องกำหนดสถานะลอจิกให้แก่ขา A0-A2 ของอุปกรณ์ที่มีการเชื่อมต่อแบบบัส I^2C ส่วนในบิต LSB เป็นบิตที่ใช้กำหนดการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวนั้น ๆ หากบิต LSB เป็น "0" หมายถึงต้องการเขียนข้อมูลไปยังอุปกรณ์นั้น ถ้าเป็น "1" จะเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ



รูปที่ 5.6 รูปแบบของข้อมูลกำหนดแอดเดรสที่ใช้ในการอ้างถึงแบบ 7 บิต

ข้อมูลในไบต์ต่อมาคือ ข้อมูลควบคุม (Control byte) ในอุปกรณ์แต่ละตัวมีการกำหนดข้อมูลควบคุมที่แตกต่างกันไป ยกตัวอย่าง ไอซีขยายพอร์ตมีข้อมูลควบคุมที่ใช้กำหนดว่า บิตใดเป็นอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พุท บิตใดเป็นเอาท์พุท ในขณะที่ไอซี ADC/DAC ต้องการข้อมูลควบคุมเพื่อกำหนดให้ทำงานเป็นวงจร ADC หรือ DAC เป็นต้น

ข้อมูลในไบต์ต่อมาคือ ข้อมูลที่ทำการถ่ายทอดจริง (Data)

หลังจากที่มีการถ่ายทอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟที่ได้รับการติดต่อต้องส่งสัญญาณรับรู้ตอบกลับมาด้วยทุกครั้ง เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้

การอ้างถึงแบบ 10 บิต

ในการอ้างถึงแบบนี้ยังคงใช้รูปแบบข้อมูลอนุกรมที่เหมือนกับแบบ 7 บิต หากแต่จะมีข้อมูลเพิ่มเติมขึ้นมาเล็กน้อย โดยในข้อมูลไบต์แรกหลังจากเกิดสภาวะเริ่มต้น ต้องกำหนดให้ 5 บิตบนมีข้อมูลเป็น 11110 ส่วนอีก 2 บิตถัดมาเป็นบิตแอดเดรสของอุปกรณ์ที่ต้องการติดต่อ ในบิต LSB ของข้อมูลไบต์แรกยังคงเป็นการกำหนดว่า ต้องการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวที่ต้องการติดต่อด้วย ข้อมูลไบต์ต่อมาเป็นข้อมูลแอดเดรสในไบต์ที่ 2 ของอุปกรณ์ที่ต้องการติดต่อด้วย ข้อมูลไบต์ถัดไปจึงเป็นข้อมูลควบคุม ข้อมูลหลังจากนั้นก็จะเป็นข้อมูลจริงที่ใช้ในการติดต่อเช่นเดียวกับการอ้างถึงแบบ 7 บิต หลังจากถ่ายทอดข้อมูลครบทุกไบต์ ต้องมีสภาวะรับรู้เกิดขึ้น เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้

5.6.6 อุปกรณ์ที่ใช้การเชื่อมต่อแบบบัส I^2C

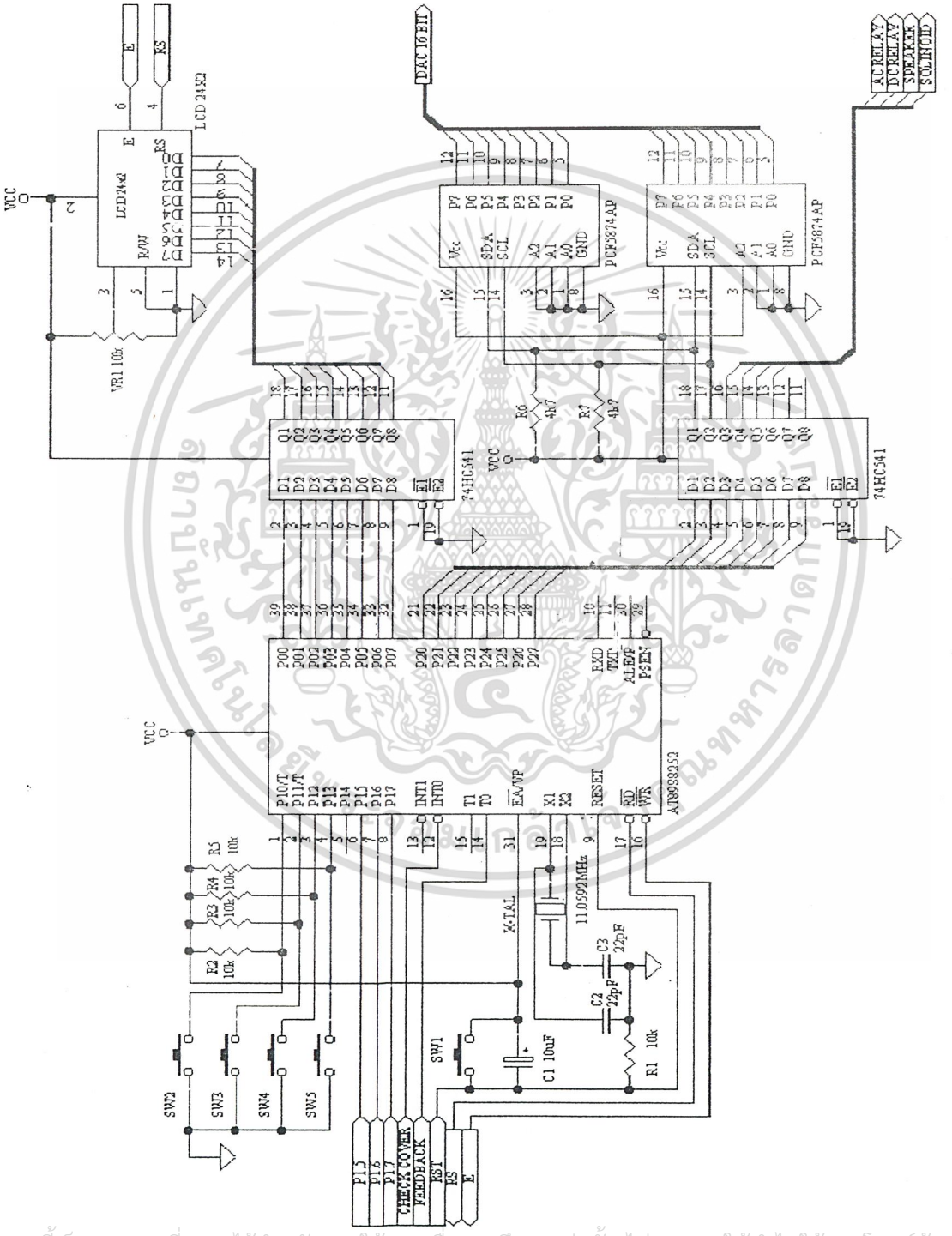
ในปัจจุบัน บัส I^2C ได้รับความนิยมเพิ่มมากขึ้นเรื่อย ๆ ด้วยข้อดีที่ชัดเจนคือ ใช้สายสัญญาณเพียง 2 เส้นเท่านั้น และการขยายระบบไมโครคอนโทรลเลอร์ที่มีจำนวนอินพุทเอาท์พุทและหน่วยความจำจำกัด สามารถทำได้ง่ายขึ้นด้วยระบบบัส I^2C เมื่อเป็นเช่นนี้จึงมีอุปกรณ์ เพอริเพอรัลที่ใช้การเชื่อมต่อแบบบัส I^2C มากมายจากหลายผู้ผลิตออกมาให้ได้ใช้งานกัน ดังมีตัวอย่างต่อไปนี้

- ไอซีขยายพอร์ตอินพุทเอาท์พุท (I/O expander) เบอร์ PCF8574, PCF8582, PCF8584
- ไอซีหน่วยความจำอีอีพรอมอนุกรม (Serial EEPROM) เบอร์ 24Cxx, PCF8570,PCF72/73
- ไอซี ADC/DAC เบอร์ PCF8591
- ไอซีรีลไทม์คล็อก (Real-time clock) เบอร์ PCF8583, PCF8593, PCF8598, 41T56C
- ไอซีขับ LCD โมดูล (LCD dirver) เบอร์ PCF8466, PCF8576, PCF8577/78, PCF8579, SAA1064
- ไอซีกำเนิดสัญญาณ DTMF (DTMF generator) เบอร์ PCD3311/12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.7 การต่ออุปกรณ์ระบบบัส I^2C กับไมโครคอนโทรลเลอร์ MCS-51

สามารถทำได้ง่ายมาก เพียงใช้ขาพอร์ต 2 ขา โดยกำหนดให้ขาหนึ่งเป็น SDA อีกขาหนึ่งเป็น SCL และต่อตัวต้านทานค่าประมาณ 4.7 kΩ พูลอัปที่ขาพอร์ตทั้งสองขา เพียงเท่านี้ก็สามารถติดต่อกับอุปกรณ์ระบบบัส I^2C ได้แล้ว



รูปที่ 5.7 วงจรตัวอย่างการต่อไมโครคอนโทรลเลอร์ MCS-51 เข้ากับระบบบัส I^2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 5.7 เป็นวงจรตัวอย่างการต่อไมโครคอนโทรลเลอร์ MCS-51 เข้ากับระบบบัส I^2C จากวงจรจะใช้ขาพอร์ต P1.0 เป็นขา SDA และ P1.1 เป็นขา SCL อุปกรณ์ที่ทำการติดต่อด้วยคือไอซีขยายพอร์ตอินพุทเอาท์พุทเบอร์ PCF8574

5.6.8 การเขียนโปรแกรมติดต่อบัส I^2C

เริ่มต้นด้วยการสร้างสภาวะมาตรฐานของบัส I^2C อันประกอบด้วย สภาวะเริ่มต้น, สภาวะสิ้นสุดการส่งข้อมูล, สภาวะหยุด และ สัญญาณนาฬิกาบนขา SCL

การสร้างสภาวะเริ่มต้น

1. เมื่อต้องการติดต่อกับบัส I^2C สิ่งแรกที่ต้องทำสำหรับไมโครคอนโทรลเลอร์ซึ่งถือว่าเป็นอุปกรณ์มาสเตอร์คือ การทำให้บัสว่างด้วยการกำหนดให้ขา SCL และ SDA มีลอจิกเป็น "1" ทั้งคู่
2. จากนั้นทำให้ขา SDA มีลอจิก "0" โดยที่ขา SCL ยังคงเป็นลอจิก "1" อยู่
3. กำหนดให้ขา SCL มีลอจิกเป็น "0" ถึงตอนนี้ทั้ง SCL และ SDA จะมีลอจิกเป็น "0" ทั้งคู่ พร้อมที่จะติดต่อได้แล้ว

การสร้างสภาวะหยุด

1. เมื่อต้องการหยุดส่งข้อมูลจะต้องส่งสภาวะหยุดออกไป โดยในตอนแรกต้องกำหนดให้ขา SCL และ SDA เป็นลอจิก "0" ทั้งคู่ก่อน
2. กำหนดให้ขา SCL มีลอจิกเป็น "1" โดย SDA ยังคงมีลอจิกเป็น "0"
3. จากนั้นทำให้ขา SDA มีลอจิกเป็น "1" ซึ่งจะทำให้ระบบบัสกลับเข้าบัสว่างอีกครั้ง พร้อมที่จะรับหรือส่งข้อมูลต่อไป

การส่งข้อมูลลอจิก "0" และลอจิก "1"

หลังจากที่ทำการส่งบิตเริ่มต้นแล้ว ลำดับต่อไปคือ จะต้องส่งข้อมูลควบคุมซึ่งจะเป็นขบวนของลอจิก "0" และลอจิก "1" สำหรับการส่งข้อมูลลอจิก "0" ต้องดำเนินการตามขั้นตอนดังนี้

1. ทำให้ขา SDA เป็น "0" สำหรับการส่งข้อมูลลอจิก "0"
2. ทำให้ขา SCL เป็น "1" สำหรับการป้อนสัญญาณนาฬิกา ในขณะที่ขา SDA ยังคงเป็น "0" อยู่
3. จากนั้นทำให้ขา SCL กลับมามีสถานะเป็นลอจิก "0" เหมือนเดิม

ในขณะที่การส่งข้อมูลลอจิก "1" มีขั้นตอนดังนี้

1. ทำให้ขา SDA มีลอจิกเป็น "1" สำหรับการส่งข้อมูลลอจิก "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำให้ขา SCL เป็น "1" สำหรับการส่งสัญญาณนาฬิกา ในขณะที่ขา SDA ยังคงเป็น "1" อยู่
3. จากนั้นทำให้ขา SCL กลับมามีสถานะเป็นลอจิก "0" เหมือนเดิม

5.6.9 การขยายจำนวนพอร์ตอินพุทเอาท์พุทด้วยไอซี PCF8574A

ข้อมูลเบื้องต้นของ PCF8574A

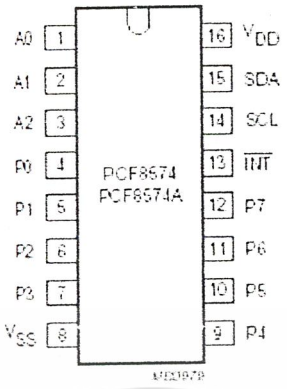
ไอซี PCF8574A มีคุณสมบัติดังนี้

- ทำงานที่ระดับแรงดันตั้งแต่ 2.5V ถึง 6V
- กินกระแสในสภาวะสแตนด์บายต่ำเพียง 10 μA
- มีเอาท์พุทอินเตอร์รัปต์แบบเดรนเปิด
- เอาท์พุทสามารถขับกระแสได้สูง โดยสามารถนำไปขับ LED ได้โดยตรง และสามารถแลตช์ค่าได้
- สามารถกำหนดตำแหน่งแอดเดรสของไอซีแต่ละตัวได้ทางฮาร์ดแวร์ ด้วยขา A0-A2 ทำให้สามารถต่อพ่วงกันได้ถึง 8 ตัว

การจัดขาของไอซีPCF8574A แสดงในรูปที่ 5.8 การทำงานของแต่ละขาแสดงในตารางที่ 1 ขาพอร์ตทั้ง 8 ขาของ PCF8574A สามารถกำหนดให้เป็นอินพุทหรือเอาท์พุทได้โดยอิสระโดยไม่จำเป็นต้องใช้คำสั่งควบคุมเพื่อเลือกให้เป็นขาเอาท์พุทหรือขาอินพุท ลักษณะวงจรภายในของพอร์ตอินพุทเอาท์พุทแสดงในรูปที่ 5.9 เมื่อจ่ายไฟให้กับ PCF8574A ครั้งแรก ขาพอร์ตทั้ง 8 ขาจะมีลอจิกเป็น "1" ซึ่งจะเป็นการจ่ายกระแสมาจากแหล่งจ่ายกระแสคงที่ภายในตัวไอซี ทำให้มีกระแสในขณะลอจิก "1" นี้เพียง 100 μA เท่านั้น ในกรณีที่ต้องการให้มีการจ่ายกระแสสูง ๆ จำเป็นต้องต่อตัวต้านทานพูลอัปเอาไว้ที่ขาพอร์ตเหล่านี้ด้วย

เมื่อต้องการให้ขาพอร์ตเหล่านี้ทำหน้าที่เป็นอินพุทจะต้องส่งสัญญาณให้ขาเหล่านี้มีลอจิก "1" เสียก่อน เมื่อขาอินพุทได้รับสัญญาณจากภายนอกป้อนเข้ามา ไอซี PCF8574A จะสร้างสัญญาณอินเตอร์รัปต์ (INT) ป้อนให้ไมโครคอนโทรลเลอร์หรือคอมพิวเตอร์รับรู้ แทนการต้องคอยตรวจสอบขาอินพุทอยู่ตลอดเวลา สัญญาณอินเตอร์รัปต์นี้จะถูกรีเซต เมื่อมีการอ่านค่าข้อมูลหรือมีการเปลี่ยนค่าของอินพุทไปสู่ค่าเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

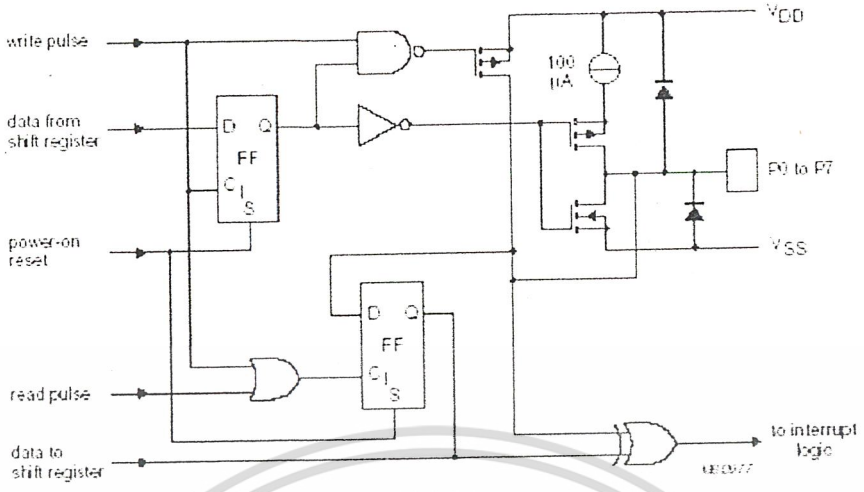


รูปที่ 5.8 การจัดขาของไอซีขยายพอร์ตอินพุทเอาท์พุท PCF8574/PCF8574A

ชื่อ	ตำแหน่งขา	หน้าที่
A0	1	อินพุทแอดเดรสตัวที่ 1
A1	2	อินพุทแอดเดรสตัวที่ 2
A2	3	อินพุทแอดเดรสตัวที่ 3
P0	4	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 0
P1	5	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 1
P2	6	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 2
P3	7	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 3
V_{SS}	8	กราวด์
P4	9	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 4
P5	10	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 5
P6	11	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 6
P7	12	พอร์ตอินพุทเอาท์พุท 2 ทิศทางบิต 7
INT	13	ขาเอาท์พุทอินเตอร์รัปต์(ทำงานที่ลอจิก 0)
SCL	14	ขาสัญญาณนาฬิกาสำหรับ I^2C บัส
SDA	15	ขาข้อมูลสำหรับ I^2C บัส
V_{DD}	16	ไฟเลี้ยง

ตารางที่ 1 รายละเอียดหน้าที่การทำงานของแต่ละขาของไอซี PCF8574/PCF8574A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.9 วงจรภายในขาพอร์ตของไอซี PCF8574/PCF8574A

การเขียนโปรแกรมควบคุม PCF8574A

เนื่องจาก PCF8574A มีการเชื่อมต่อเป็นแบบบัส I²C ดังนั้นการติดต่อจึงสามารถใช้โปรแกรมย่อยการติดต่อกับอุปกรณ์สเลฟ, การสร้างสภาวะเริ่มต้น, สภาวะหยุด, การอ่าน และเขียนข้อมูลที่แสดงไว้ในรูปที่ 10 ได้ ส่วนที่ต้องเปลี่ยนแปลงสำหรับ PCF8574A คือ ข้อมูลกำหนดแอดเดรส โดยข้อมูลของ PCF8574A มีรูปแบบดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	1	1	1	A2	A1	A0	R/$\overline{W}$

บิต A0, A1, A2 ใช้ในการระบุ PCF8574A ที่ใช้บนบอร์ดในกรณีที่มีการต่อ PCF8574 มากกว่า 1 ตัว โดยค่าของ A0-A2 จะมีความแตกต่างกันไปในแต่ละตัว สามารถกำหนดได้ทางฮาร์ดแวร์ โดยการต่อขา A0-A2 เข้ากับไฟเลี้ยง +5V เพื่อกำหนดเป็นลอจิก "1" หรือกราวด์เพื่อกำหนดเป็นลอจิก "0" ดังนั้นค่าแอดเดรสของ PCF8574A 1 ตัวจึงต้องนำสถานะที่กำหนดทางฮาร์ดแวร์ที่ขา A0-A2 มารวมด้วยจึงจะสมบูรณ์ ส่วนบิต R/W ใช้กำหนดว่าต้องการอ่านหรือเขียนข้อมูลกับไอซี PCF8574A ยกตัวอย่าง ถ้าหากกำหนดขา A0-A2 ลงกราวด์ทั้งหมด และต้องการอ่านข้อมูลจาก PCF8574A ข้อมูลกำหนดแอดเดรสที่ต้องส่งให้แก่ PCF8574A คือ 01110001B เป็นต้น

ไอซี PCF8574A ยังมีอีกเบอร์หนึ่งในอนุกรมเดียวกัน นั่นคือ PCF8574 ซึ่งก็มีข้อมูลกำหนดแอดเดรสที่แตกต่างกับ PCF8574A แต่ฟังก์ชั่นการทำงานเหมือนกันทุกประการ โดยข้อมูลกำหนดแอดเดรสของ PCF8574 มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	1	0	0	A2	A1	A0	$R/\overline{W}$

ดังนั้น จึงสามารถต่อพ่วงไอซีในอนุกรม PCF8574x ได้สูงถึง 16 ตัว

การส่งหรือเขียนข้อมูลไปยัง PCF8574A โดยมีลำดับขั้นตอนดังนี้

1. เตรียมข้อมูลกำหนดแอดเดรสของ PCF8574A
2. เรียกโปรแกรมย่อยการติดต่อกับอุปกรณ์สเลฟ
3. รอรับการตอบกลับจาก PCF8574A
4. ส่งข้อมูลไปยัง PCF8574A
5. เรียกโปรแกรมย่อยสภาวะหยุด

สำหรับการอ่านข้อมูลจาก PCF8574A มีลักษณะใกล้เคียงกันคือมีรูปแบบดังนี้

1. เตรียมข้อมูลกำหนดแอดเดรสของ PCF8574A
2. เรียกโปรแกรมย่อยการติดต่อกับอุปกรณ์สเลฟ
3. รอรับการตอบกลับจาก PCF8574A
4. อ่านข้อมูลจาก PCF8574A โดยใช้โปรแกรมย่อยการอ่านข้อมูลจากอุปกรณ์สเลฟ
5. เรียกโปรแกรมย่อยสภาวะหยุด

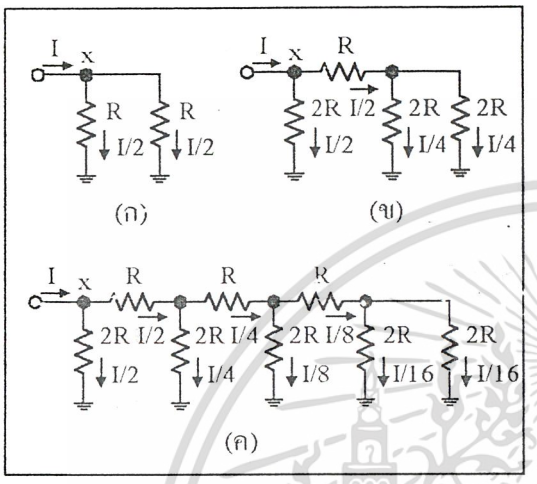
5.7 วงจรแปลงสัญญาณดิจิตอลเป็นอะนาลอกแบบ R-2R

5.7.1 ความรู้เบื้องต้นเกี่ยวกับวงจร DAC แบบ R-2R

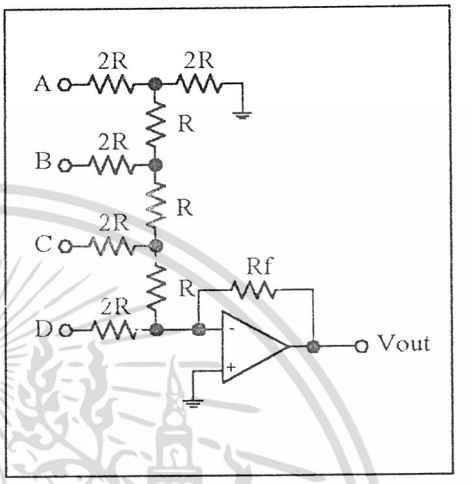
วงจร DAC แบบ R-2R แลดเดอร์เป็นวงจรอีกแบบหนึ่งที่ได้รับความนิยม ทั้งนี้เนื่องจากสามารถหาอุปกรณ์ในการสร้างวงจรง่าย กระบวนการสร้างวงจร DAC จะเริ่มต้นจากวงจรตัวต้านทางต่อขนานกัน 2 ตัว ดังในรูปที่ 5.9 (ก) หากตัวต้านทานสองตัวมีค่าเท่ากัน กระแสที่ไหลผ่านตัวต้านทางทั้งสองตัวจะมีค่าเท่ากันคือ เท่ากับ I/2 จากนั้นทำการเปลี่ยนค่าของตัวต้านทานเพื่อขยายสาขาของวงจรดังในรูปที่ 5.9 (ข) ค่าความต้านทานที่จุด X จะเท่ากัน ซึ่งก็คือ 2R ดังนั้นค่าความต้านทานทั้งทางด้านซ้ายและขวามือของจุด X จะมีค่าเท่ากันคือ 2R ทำให้ปริมาณกระแสที่ไหลผ่านเท่ากันคือ I/2 จากจุด X จะมีกระแสไหลเข้า I/2 จะถูกแยกเป็นสองทางด้วยตัวต้านทาน 2R สองตัวที่ต่อขนานกัน กระแสที่ไหลผ่านวงจรในสาขานี้จึงเป็น I/4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น หากต่อพ่วงวงจรตัวต้านทานเข้าไปอีกหลายสาขา ก็จะทำให้กระแสถูกเฉลี่ยมากขึ้น เหมือนกับการสร้างบันไดของกระแสทีละขั้น ด้วยอัตราส่วนของตัวต้านทานที่คงที่ คือ R และ 2R ดังในวงจรรูปที่ 5.9 (ค) จากนั้นนำวงจร R-2R แลดเดอร์ นี้ต่อเข้ากับวงจรแปลงค่ากระแสเป็นแรงดันโดยใช้ออปแอมป์ ดังในรูปที่ 5.11



รูปที่ 5.10 พื้นฐานของวงจรแปลงสัญญาณดิจิตอลเป็นอะนาลอกแบบ R-2R



รูปที่ 5.11 วงจร DAC แบบ R-2R ที่มีการใช้ออป แอมป์ในการแปลงค่ากระแสเป็นแรงดัน

แรงดันเอาท์พุท Vout จะมีค่าเท่ากับ โดยที่ Iin จะเป็นค่าผลรวมของกระแสที่ได้จากวงจร R-2R แลดเดอร์ ยกตัวอย่าง ป้อนข้อมูลดิจิตอล 1011 เข้าที่อินพุท DCBA ทำให้เกิดกระแสไหลผ่าน R2, R4 และ R8 เข้ารวมกันที่อินพุทของออปแอมป์กลายเป็น Iin ซึ่งมีค่า

$$Iin = \frac{I}{2} + \frac{I}{8} + \frac{I}{16}$$

ดังนั้น

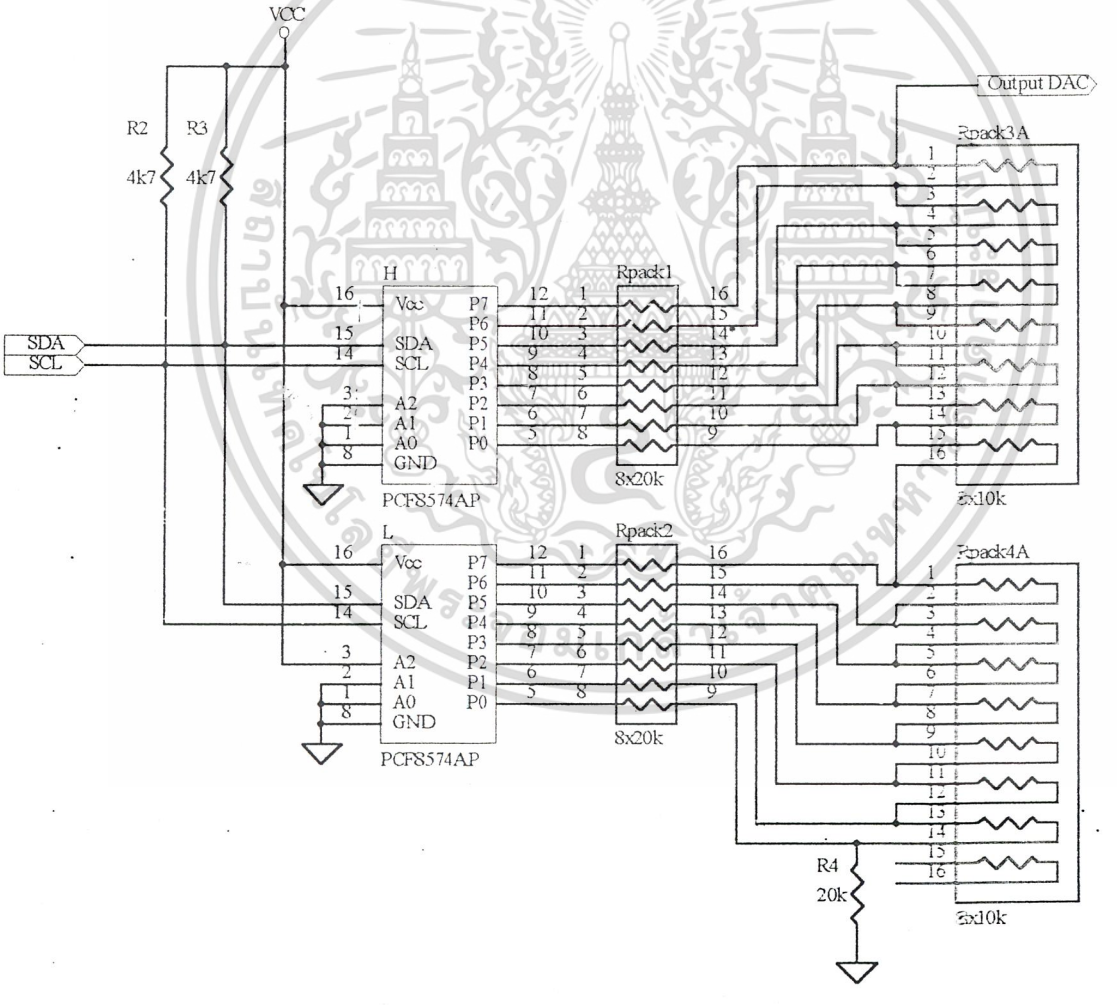
$$Vout = \left(\frac{I}{2} + \frac{I}{8} + \frac{I}{16}\right) \times Rf$$

จะเห็นได้ว่า การใช้วงจร DAC แบบ R-2R จะมีข้อดีคือ หาตัวต้านทานได้ง่ายกว่าแบบกำหนดน้ำหนักตัวต้านทานคือ ใช้ตัวต้านทานเพียง 2 ค่า ในขณะที่แบบกำหนดน้ำหนักตัวต้านทานต้องใช้ตัวต้านทานหลายค่า และต้องมีค่าทวีคูณกันไปเรื่อย ๆ ซึ่งอาจทำให้หาตัวต้านทานได้ยากขึ้นตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.7.2 การเชื่อมต่อวงจร DAC แบบ R-2R เข้ากับไมโครคอนโทรลเลอร์ MCS-51

ในการเชื่อมต่อวงจร DAC แบบ R-2R เข้ากับไมโครคอนโทรลเลอร์ MCS-51 สามารถต่อเข้ากับพอร์ตใดก็ได้โดยผ่านวงจรแลตช์ ทั้งนี้เพื่อให้สามารถข้อมูลดิจิตอลที่ส่งออกไปสามารถคงค่าอยู่ได้จนกว่าจะมีการส่งข้อมูลดิจิตอลออกไปใหม่ การเขียนโปรแกรมเพื่อติดต่อทำได้โดยพิจารณาวงจร DAC แบบ R-2R เป็นเหมือนกับอุปกรณ์เอาท์พุทแบบขนานตัวหนึ่ง เมื่อส่งข้อมูลดิจิตอลออกไป วงจร DAC นี้ก็จะแปลงเป็นสัญญาณอะนาลอกให้เอง ซึ่งในโครงงานนี้ทำการต่อวงจร DAC แบบ R-2R แลดเดอร์ เข้ากับไมโครคอนโทรลเลอร์โดยผ่าน I^2C ที่ทำหน้าที่ขยายพอร์ต เพราะในโครงงานนี้ใช้ความละเอียดสูง (16 Bit) จึงต้องมีการขยายพอร์ตโดยใช้ IC I^2C 2 ตัว แล้วต่อเข้ากับวงจร DAC แบบ R-2R แลดเดอร์ ดังแสดงในรูปที่ 5.12



รูปที่ 5.12 การต่อวงจร DAC แบบ R-2R แลดเดอร์ โดยใช้ I^2C ขยายพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
การออกแบบและขั้นตอนการทำงานของโปรแกรม

6.1 การออกแบบ

6.1.1 การออกแบบตัวนับรอบความเร็วของมอเตอร์ : Counter

จะใช้ Timer/Counter T0 เป็นตัวนับพัลส์ที่ป้อนกลับมาจากใบพัดตัดแสงของมอเตอร์ โดยจะตั้งการทำงานเป็นโหมด 1 คือ เป็นการทำงานแบบขนาด 16 บิต โดยก่อนเริ่มต้นการนับ จะทำการกำหนดค่าเริ่มต้นเป็น 0 ก่อนทุกครั้ง

ในการใช้งานนั้น ค่าความเร็วที่ต้องการหน่วยเป็นรอบต่อนาที (RPM: Rounds per minute) แต่ในการตรวจสอบนับรอบความเร็วนั้น หน่วยจะเป็นรอบต่อวินาที (RPS: Rounds per second) ดังนั้นจะได้ว่า

จาก $1\ \text{RPM} = \frac{1\ \text{round}}{1\ \text{min}}$

$= \frac{1\ \text{round}}{60\ \text{sec}}$

และใน 1 รอบ จะมีพัลส์เกิดขึ้นมา 6 พัลส์ เพราะใบพัดมีอยู่ 6 ชิ้ ดังนั้นจะได้ว่า

ค่าอัตราการนับใน 1 RPM $= \frac{1\ \text{round}}{60\ \text{sec}} \times 6 = \frac{1\ \text{round}}{10\ \text{sec}}$

เพราะฉะนั้นในโหมด HEMATOCRIT จะได้

ค่าอัตราการนับในความเร็ว 12,000 RPM $= \frac{12,000\ \text{round}}{10\ \text{sec}} = \frac{1,200\ \text{round}}{\text{sec}} = 1,200\ \text{RPS}$

และในโหมด CENTRIFUGE จะได้

ค่าอัตราการนับในความเร็ว 3,000 RPM $= \frac{3,000\ \text{round}}{10\ \text{sec}} = \frac{300\ \text{round}}{\text{sec}} = 300\ \text{RPS}$

ดังนั้นค่าความเร็วอ้างอิงที่ใช้ในการเปรียบเทียบในโปรแกรมก็คือ

โหมด HEMATOCRIT: 1,200 RPS = 04B0H

โหมด CENTRIFUGE: 300 RPS = 012CH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.2 การออกแบบตัวตั้งเวลา : Timer

จะใช้ Timer/Counter T1 เป็นตัวตั้งเวลา หรือ Timer โดยจะตั้งการทำงานในโหมด 1 คือ เป็นการทำงานแบบนับค่าขนาด 16 บิต ซึ่งจะนับได้สูงสุด 65,535 ครั้ง และอัตราในการนับของ Timer/Counter จะเท่ากับ $\frac{1}{12}$ ของความถี่สัญญาณนาฬิกา ซึ่งในการใช้งานจะใช้ความถี่สัญญาณนาฬิกาเท่ากับ 12 MHz

ดังนั้นจะได้อัตราในการนับเท่ากับ $\frac{12\ \text{MHz}}{12}$ ซึ่งจะได้เวลาในการนับหนึ่งครั้งเท่ากับ $\frac{12}{12\ \text{MHz}}$ มีค่าประมาณ 1 ไมโครวินาที

เริ่มต้นกำหนดค่า Timer ให้ทำการนับ 50 มิลลิวินาที โดยคำนวณหาค่าเริ่มต้นได้จาก

50 มิลลิวินาที = 50000 ไมโครวินาที

อัตราการนับสูงสุดของ Timer โหมดนี้ สามารถนับได้ 65536 ครั้ง หรือ 65536 ไมโครวินาที จึงจะเกิดอินเตอร์รัปต์ 1 ครั้ง ดังนั้นที่ 50000 ไมโครวินาที จะต้องกำหนดค่าเริ่มต้นเท่ากับ

65536 – 50000 = 15536

ทำการแปลงค่าเป็นเลขฐาน 16 จะได้ค่าเริ่มต้นการนับของ Timer

15536 = 3CB0H

เมื่อทำการนับครบ 50 มิลลิวินาทีแล้ว จะเกิดอินเตอร์รัปต์ขึ้น แล้วไปทำการเพิ่มค่าตัวทดขึ้น 1 หลังจากนั้นจะทำการกำหนดค่าเริ่มต้นให้กับ Timer ใหม่

ถ้าหากต้องการให้นับ 1 วินาที จะต้องให้ตัวทดมีค่าเท่ากับ

$$\frac{1\ \text{วินาที}}{50\ \text{มิลลิวินาที}} = 20$$

จะได้ค่าตัวทดเท่ากับ 20 เมื่อ Timer ทำการนับ 20 รอบจะได้เวลา 1 วินาที แล้วไปทำการเพิ่มค่าตัวทดที่เก็บค่าวินาที เมื่อตัวทดวินาที มีค่าเท่ากับ 60 จะได้เวลา 1 นาที แล้วไปทำการเพิ่มค่าตัวทดนาที จนนับได้ค่าที่ต้องการ

จากการออกแบบจะได้ค่าเริ่มต้นดังนี้

TL1 = 0B0H

TH1 = 3CH

โดยค่าเริ่มต้นตัวทดทุกตัวเท่ากับ 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.3 การออกแบบโปรแกรมซอฟท์สตาร์ท

โปรแกรมซอฟท์สตาร์ท เป็นส่วนที่ทำหน้าที่ในการขับมอเตอร์ให้เริ่มหมุนอย่างช้าๆ แล้วค่อยๆเพิ่มความเร็วขึ้น ทั้งนี้เพื่อเป็นการป้องกันการกระชากของมอเตอร์ และเป็นการป้องกันการเสียหายของวงจรขับมอเตอร์ การออกแบบโปรแกรมซอฟท์สตาร์ท มีดังนี้

1) หาค่าแรงดัน DTC ที่มอเตอร์เริ่มหมุน ได้แรงดันประมาณ 2.5 V ได้ข้อมูลที่ต้องส่ง I2C เท่ากับ $\frac{65535x2.5}{5}$ = 32767.5 แปลงเป็นฐานสิบหกได้ 7FFFH
2) หาค่าแรงดัน DTC ที่ทำให้มอเตอร์หมุน 3000 rpm ได้แรงดันประมาณ 2.462 V กำหนดเวลาในการสตาร์ทประมาณ 10 วินาที ดังนั้นแรงดันต้องลดลงวินาทีละ $\frac{2.5-2.462}{10}$ = 3.8 mV จะได้ขนาด STEP เท่ากับ $\frac{3.8mVx65535}{5V}$ = 50 หรือ 32H
3) หาค่าแรงดัน DTC ที่ทำให้มอเตอร์หมุน 12000 rpm ได้แรงดันประมาณ 0.5 V กำหนดเวลาในการสตาร์ทประมาณ 30 วินาที ดังนั้นแรงดันต้องลดลงวินาทีละ $\frac{2.5-0.5}{30}$ = 66.67 mV จะได้ขนาด STEP เท่ากับ $\frac{66.67mVx65535}{5V}$ = 873 หรือ 0369H
4) ทำการรอเมื่อทำส่วนสตาร์เสร็จตามเวลาที่ตั้งไว้ เนื่องจากแรงเฉื่อยของมอเตอร์ประมาณ 30 วินาที หลังจากนั้นจึงค่อยทำการชดเชยความเร็ว

6.1.4 การออกแบบโปรแกรมชดเชยความเร็ว

การออกแบบโปรแกรมชดเชยความเร็ว มีดังนี้

1) หาความแตกต่างของความเร็วที่ได้จาก Counter กับ ความเร็วอ้างอิงที่ได้จากข้อ 6.1.1
2) ทำการเลือกขนาด STEP ที่ใช้ในการเพิ่มหรือลดความเร็วดังนี้
   - ความเร็วแตกต่างกัน มากกว่า 2550 rpm
     ใช้ขนาด STEP = 256 หรือ 0100H
   - ความเร็วแตกต่างกัน 1000 - 2550 rpm
     ใช้ขนาด STEP = 80 หรือ 0050H
   - ความเร็วแตกต่างกัน 300 - 990 rpm
     ใช้ขนาด STEP = 10 หรือ 000AH
   - ความเร็วแตกต่างกัน 100 - 290 rpm
     ใช้ขนาด STEP = 3 หรือ 0003H
   - ความเร็วแตกต่างกัน 10 - 90 rpm ต้องพิจารณาว่าความเร็วขณะนี้มากกว่าความเร็วอ้างอิงหรือไม่ หากมากกว่าจะไม่มีการชดเชย แต่ถ้าน้อยกว่าจะมีการชดเชย ดังนี้
     ใช้ขนาด STEP = 256 หรือ 0100H

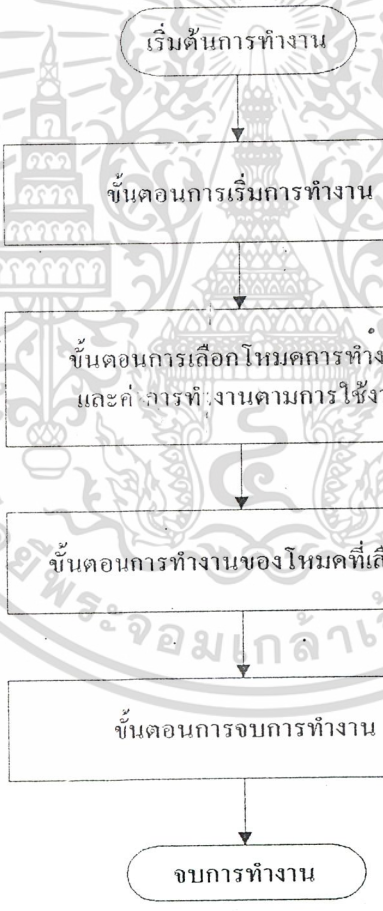
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 ขั้นตอนการทำงานของโปรแกรม

ขั้นตอนการทำงานของโปรแกรมสามารถแบ่งได้ออกเป็น 4 ขั้นตอนหลัก ๆ ดังนี้

1. ขั้นตอนการเริ่มต้นการทำงาน
2. ขั้นตอนการเลือกโหมดการทำงาน
   - โหมด HEMATOCRIT ใช้ความเร็วรอบ 12,000 รอบต่อนาที
   - โหมด CENTRIFUGE ใช้ความเร็วรอบ 3,000 รอบต่อนาที
3. ขั้นตอนการทำงานของโหมดที่ได้เลือกไว้
4. ขั้นตอนการจบการทำงาน

โดยมีโฟลวชาร์ตแสดงดังรูปที่ 6.1



รูปที่ 6.1 โฟลวชาร์ตแสดงขั้นตอนการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในแต่ละขั้นตอน จะสามารถแบ่งเป็นขั้นตอนย่อย ๆ ได้ดังต่อไปนี้

6.2.1 ขั้นตอนการเริ่มต้นการทำงาน โดยมีโฟลวชาร์ตดังรูปที่ 6.2 และขั้นตอนดังนี้

1) กำหนดแอดเดรสของตัวแปรต่าง ๆ และบิตของตัวแปรต่าง ๆ ในการใช้งาน
2) กำหนดแอดเดรสเริ่มต้นของโปรแกรม และแอดเดรสของอินเตอร์รัปต์ไทเมอร์0 และไทเมอร์ 1
3) กำหนดค่าเริ่มต้นของพอร์ตและเตรียม LCD โมดูลให้พร้อมในการทำงาน
4) แสดงข้อความบนหน้าจอ LCD ดังนี้

   ' KMITL '

   ' P R E S E N T '
5) ตรวจสอบว่าฝาครอบเปิดอยู่หรือไม่?

   5.1) ถ้าปิดอยู่ จะทำงานดังนี้

   - จะแสดงข้อความบนหน้าจอ LCD ดังนี้

     ' Please press OPEN key '

     ' For opening the cover. '

     แล้วจะรอการกดปุ่ม OPEN เพื่อให้ขดลวดโซลีนอยด์ทำงานปลดล็อคฝาครอบ
   - จากนั้นจะแสดงข้อความบนหน้าจอ LCD ดังนี้

     ' Please open the cover '

     ' Put or pick the tubes. '

     เพื่อรอการเปิดฝาครอบ ในการหยิบหรือวางหลอดเลือด และเมื่อฝาครอบถูกเปิดแล้ว ก็จะเข้าสู่ขั้นตอนลำดับที่ 6 ต่อไป

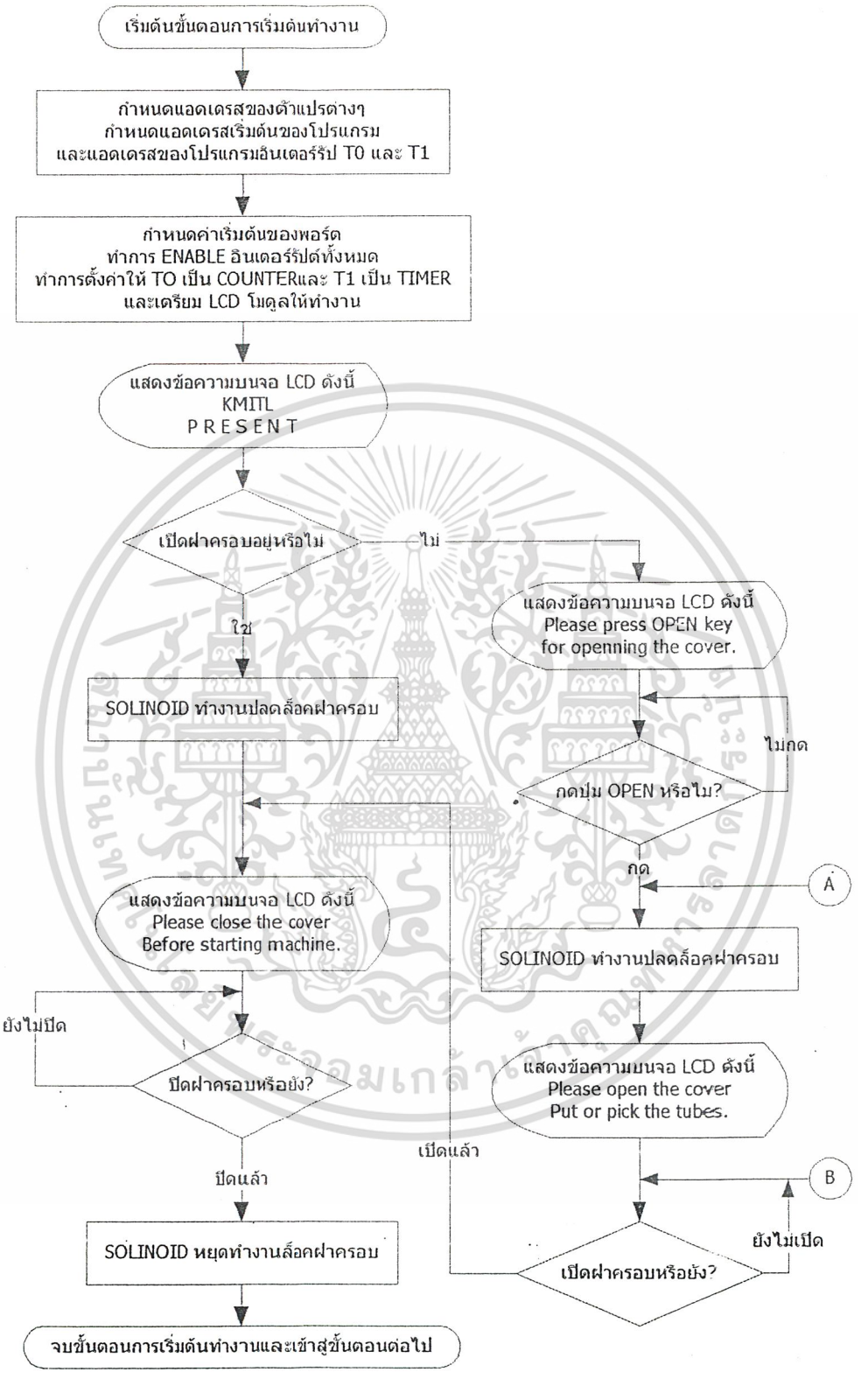
   5.2) ถ้าเปิดอยู่ SOLINOID จะทำงาน และจะทำขั้นตอนลำดับที่ 6 ต่อไป
6) แสดงข้อความบนหน้าจอ LCD ดังนี้

   ' Please close the cover '

   ' Before starting machine '

   และทำงานวนลูปรอการปิดฝาครอบ
7) เมื่อทำการปิดฝาครอบแล้ว ก็จะเข้าสู่ขั้นตอนการเลือกโหมดการทำงาน

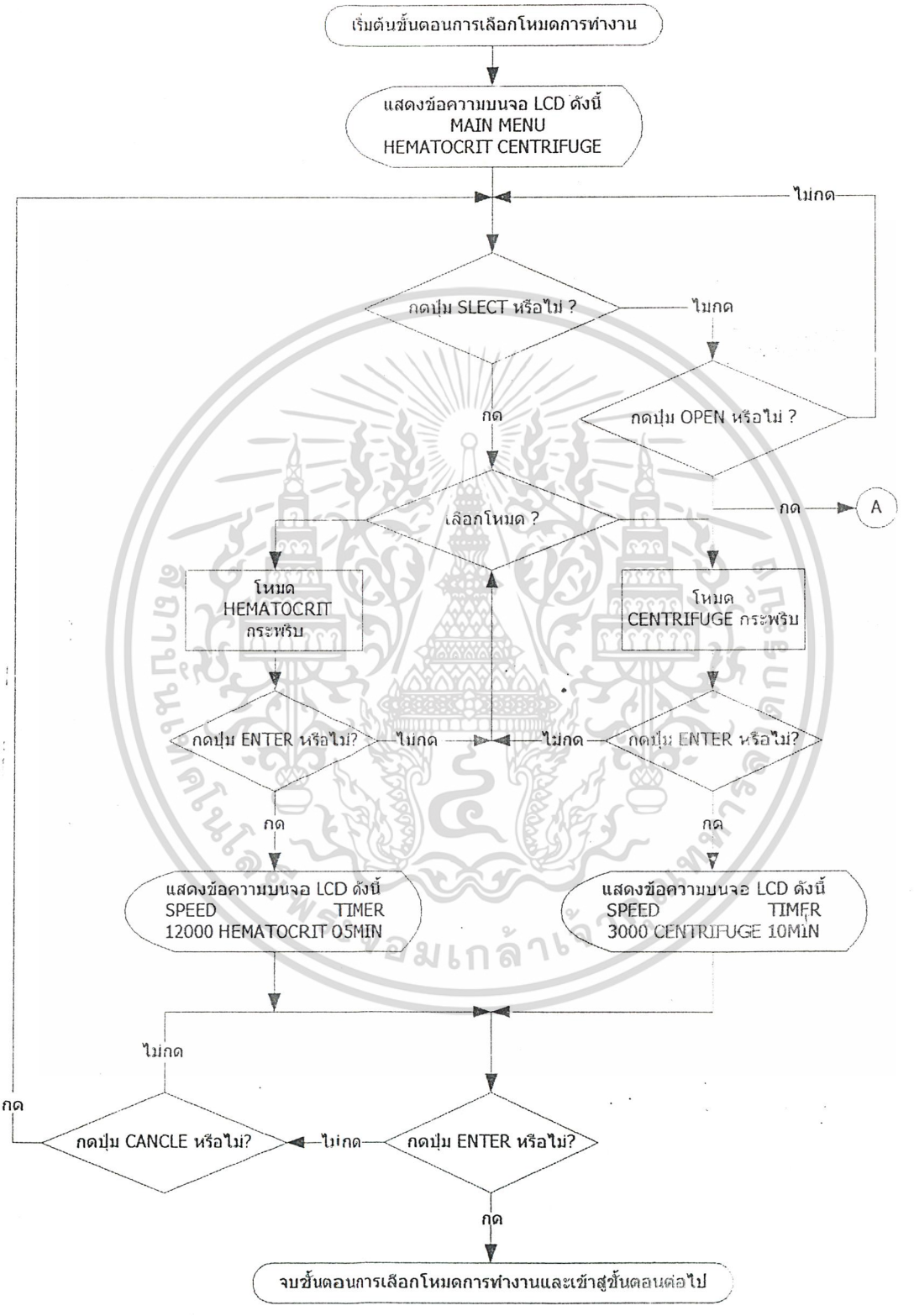
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.2 โฟลวชาร์ตแสดงขั้นตอนการเริ่มต้นการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.2 ขั้นตอนการเลือกโหมดการทำงาน โดยมีโฟลวชาร์ตดังรูปที่ 6.3 และขั้นตอนดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.3 แสดงโฟลวชาร์ตของขั้นตอนการเลือกโหมดการทำงาน

1) ต่อจากขั้นตอนเริ่มต้นโปรแกรม หลังจากปิดฝาครอบในขั้นตอนลำดับที่ 7 แล้ว จะปรากฏข้อความบนหน้าจอ LCD ดังนี้

' MAIN MENU '

'HEMATOCRIT CENTRIFUGE'

2) จากนั้นจะทำการวนลูปรอการกดปุ่ม OPEN และ SELECT โดยแต่ละปุ่มจะทำงานดังนี้

- ถ้ากดปุ่ม OPEN จะทำให้ขดลวดโซลีนอยด์ทำงาน เพื่อปลดล็อกฝาครอบ โดยจะกระโดดกลับไปทำงานต่อที่ขั้นตอนเริ่มต้นโปรแกรม ในลำดับที่ 5
- ถ้ากดปุ่ม SELECT จะทำการเลือกโหมดการทำงานระหว่างโหมด HEMATOCRIT และ CENTRIFUGE โดยแต่ละครั้งที่กดปุ่ม SELECT ชื่อโหมดจะกระพริบสลับกัน

3) หลังจากกดปุ่ม SELECT เพื่อเลือกโหมดแล้ว จะทำการรอการกดปุ่ม ENTER เพื่อยืนยันการเลือกโหมด และปุ่ม SELECT เพื่อทำการเปลี่ยนโหมด

4) เมื่อทำการกดปุ่ม ENTER แล้ว จะมีการแสดงข้อความเพื่อยืนยันโหมดดังนี้

- หากเลือกโหมด HEMATOCRIT จะแสดงข้อความดังนี้

' SPEED TIMER '

' 12000 HEMATOCRIT 05MIN '

- หากเลือกโหมด CENTRIFUGE จะแสดงข้อความดังนี้

' SPEED TIMER '

' 3000 CENTRIFUGE 10MIN '

5) หลังจากนั้น จะทำการวนลูปเพื่อรอการกดปุ่ม ENTER และปุ่ม CANCLE โดยแต่ละปุ่มจะทำงานดังนี้

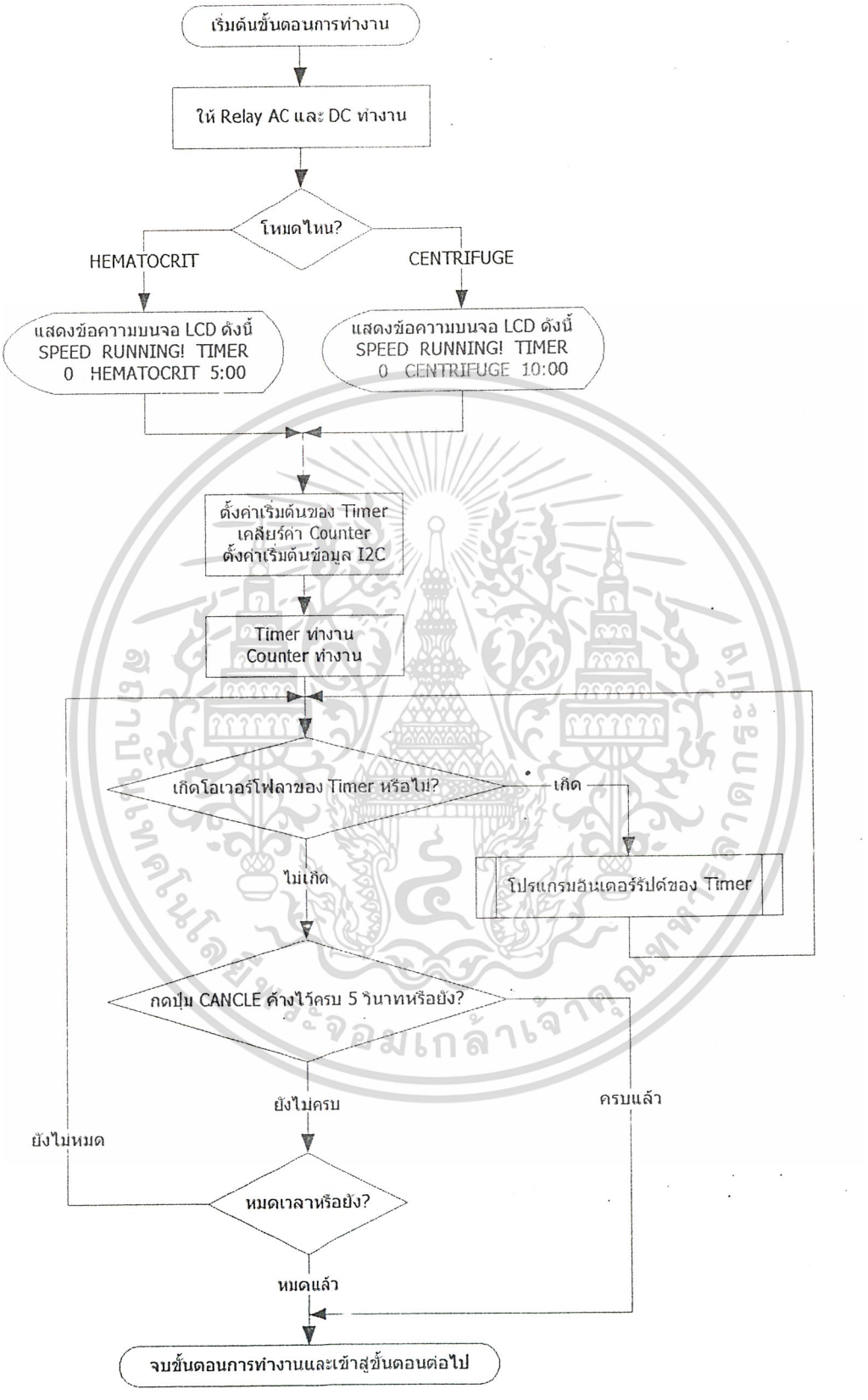
- ถ้ากดปุ่ม ENTER จะเข้าสู่ขั้นตอนต่อไป คือขั้นตอนการทำงานตามโหมดที่เลือกไว้
- ถ้ากดปุ่ม CANCLE จะกลับไปจุดเริ่มต้นของขั้นตอนนี้ คือ ข้อ 1

6.2.3 ขั้นตอนการทำงานของโหมดที่เลือกไว้ โดยมีโฟลวชาร์ตดังรูปที่ 6.4, 6.5, 6.6 , 6.7, 6.8, 6.9 , 6.10, 6.11 และขั้นตอนดังนี้

1) ส่งข้อมูลลอจิก "0" ไปยังพอร์ต P2.3 เพื่อให้ AC Relay ทำงาน

2) ส่งข้อมูลลอจิก "0" ไปยังพอร์ต P2.4 เพื่อให้ DC Relay ทำงาน ซึ่งหลังจากนี้มอเตอร์จะเริ่มทำงานแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 แสดงโฟลวชาร์ตขั้นตอนการทำงานตามโหมดที่เลือกไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ทำการแสดงข้อความบนจอ LCD ดังนี้

- หากเลือกโหมด HEMATOCRIT

  ‘ SPEED RUNNING! TIMER ‘

  ‘ 0 HEMATOCRIT 5:00 ‘

- หากเลือกโหมด CENTRIFUGE

  ‘ SPEED RUNNING! TIMER ‘

  ‘ 0 CENTRIFUGE 10:00 ‘

4) ทำการตั้งค่าเริ่มต้นของ Timer และค่าเริ่มต้นข้อมูลที่ส่งให้ I2C ให้มีค่าแรงดัน DTC เท่ากับ 2.5 V (จุดเริ่มทำงานของมอเตอร์) และเคลียร์ค่าของ Counter

5) เปิดการทำงานของ Timer และ Counter (เมื่อเกิดโอเวอร์โฟล์ว จะเข้าสู่โปรแกรมอินเตอร์รัปต์ โปรแกรมอินเตอร์รัปต์แสดงดังรูปที่ 6.5)

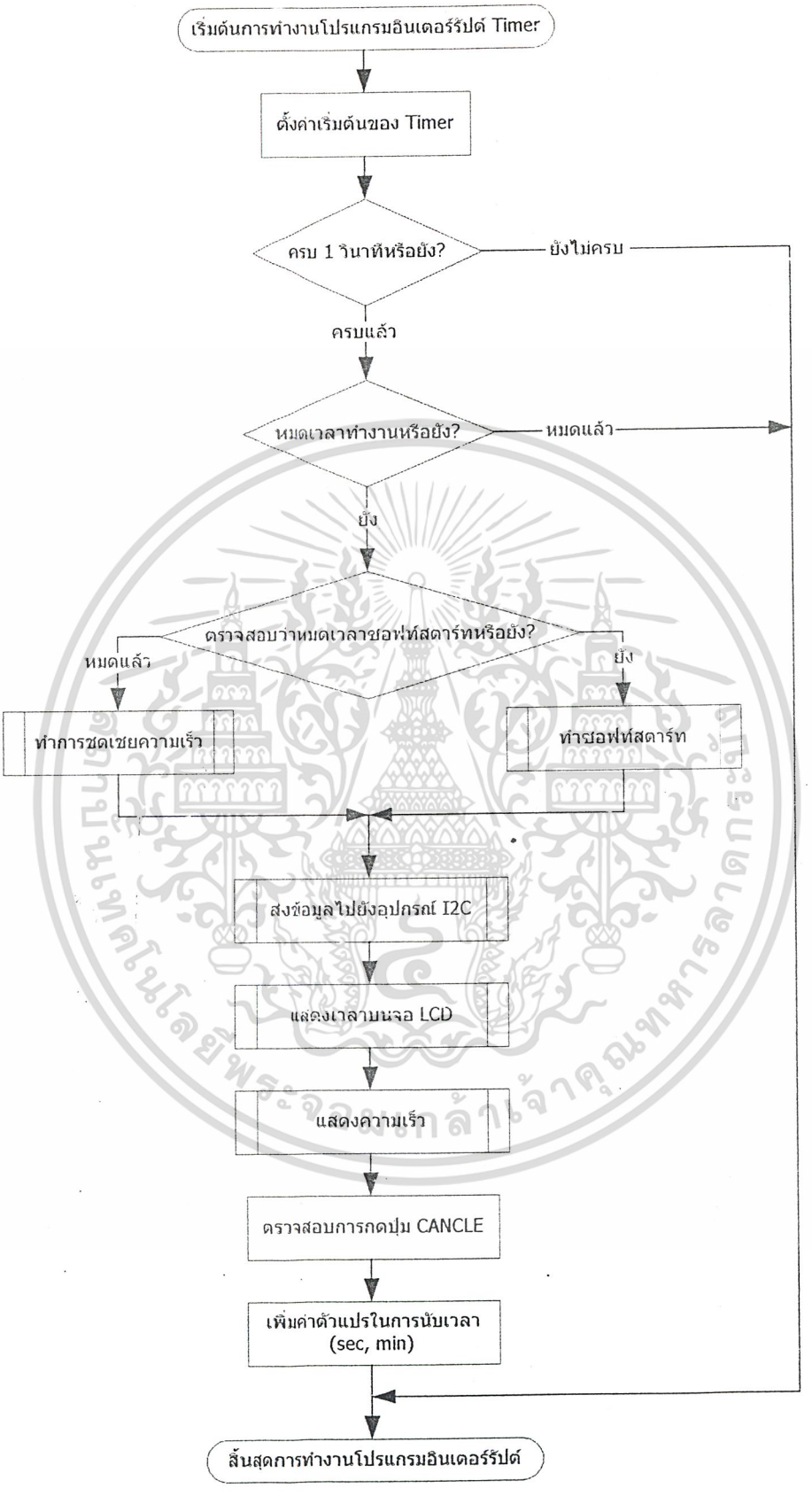
6) ทำการตรวจสอบว่ามีการกดปุ่ม CANCLE ครบ 5 วินาทีหรือยัง หากยัง ข้ามไปข้อ 8)

7) เข้าสู่ขั้นตอนต่อไป คือ ขั้นตอนจบการทำงาน

8) ตรวจสอบว่านับเวลาครบหรือยัง (เวลาหมด) หากยัง กลับไปข้อ 6)

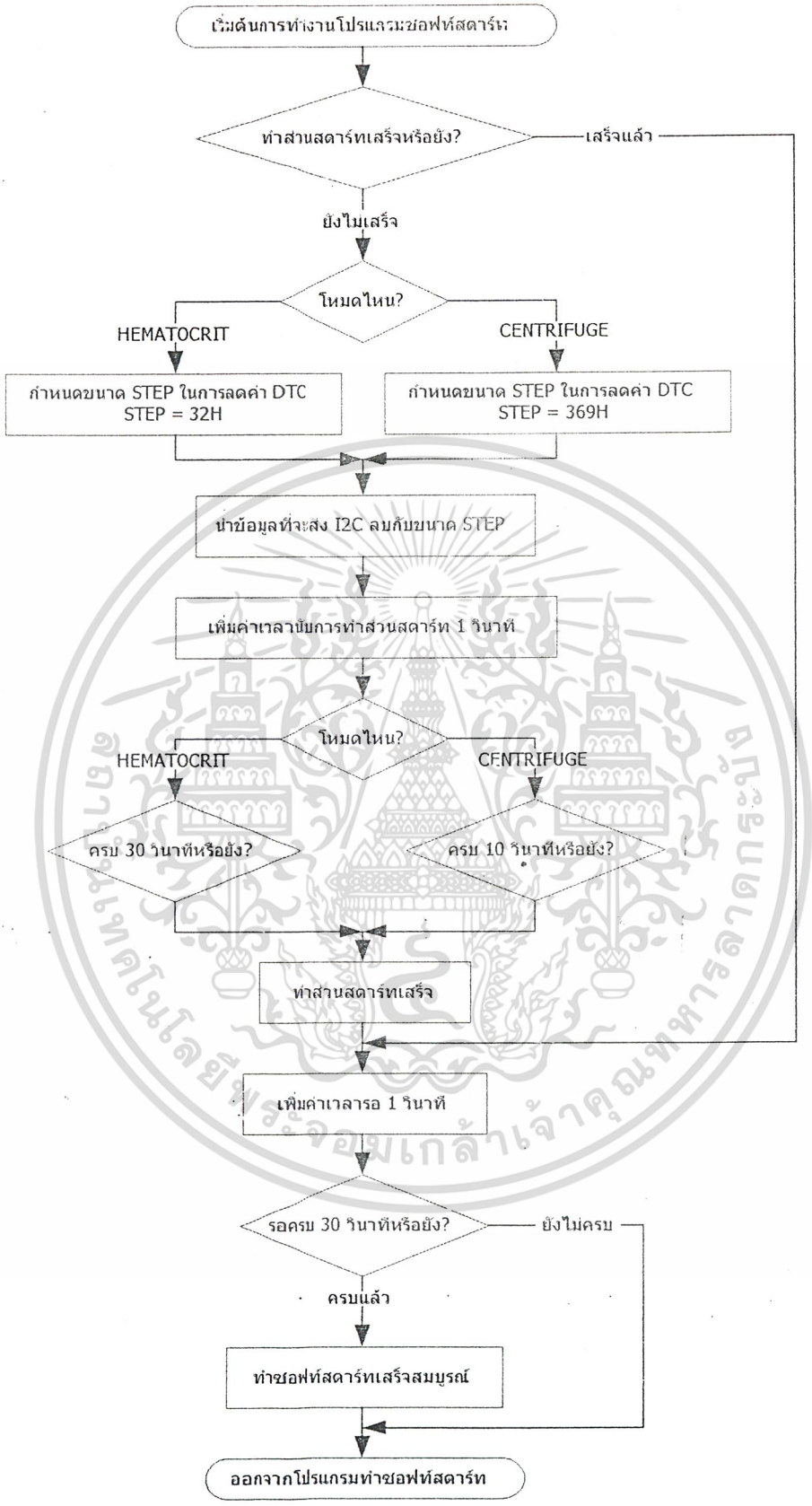
9) เข้าสู่ขั้นตอนต่อไป คือ ขั้นตอนจบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



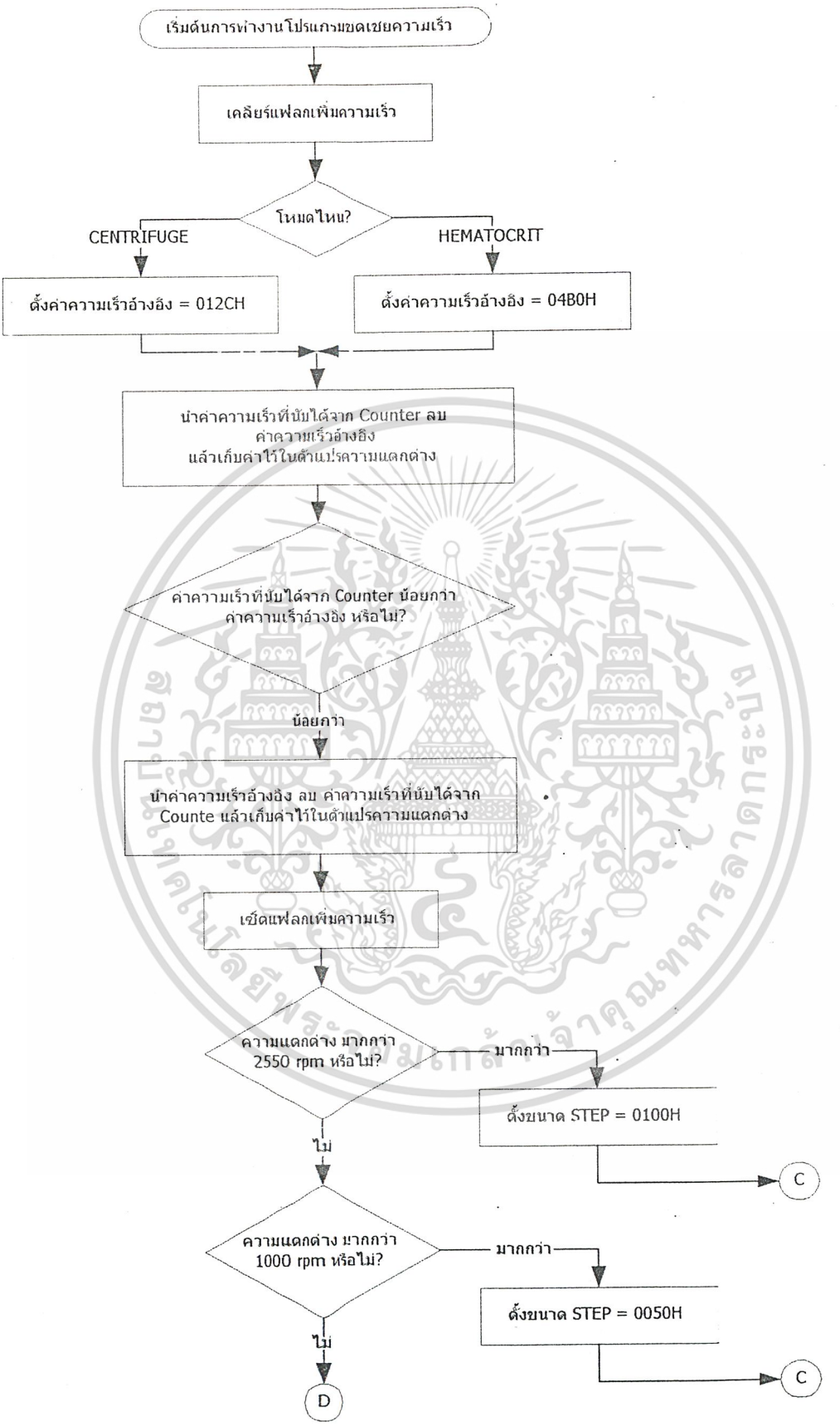
รูปที่ 6.5 แสดงโฟลวชาร์ตโปรแกรมอินเตอร์รัปต์ของตัวตั้งเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



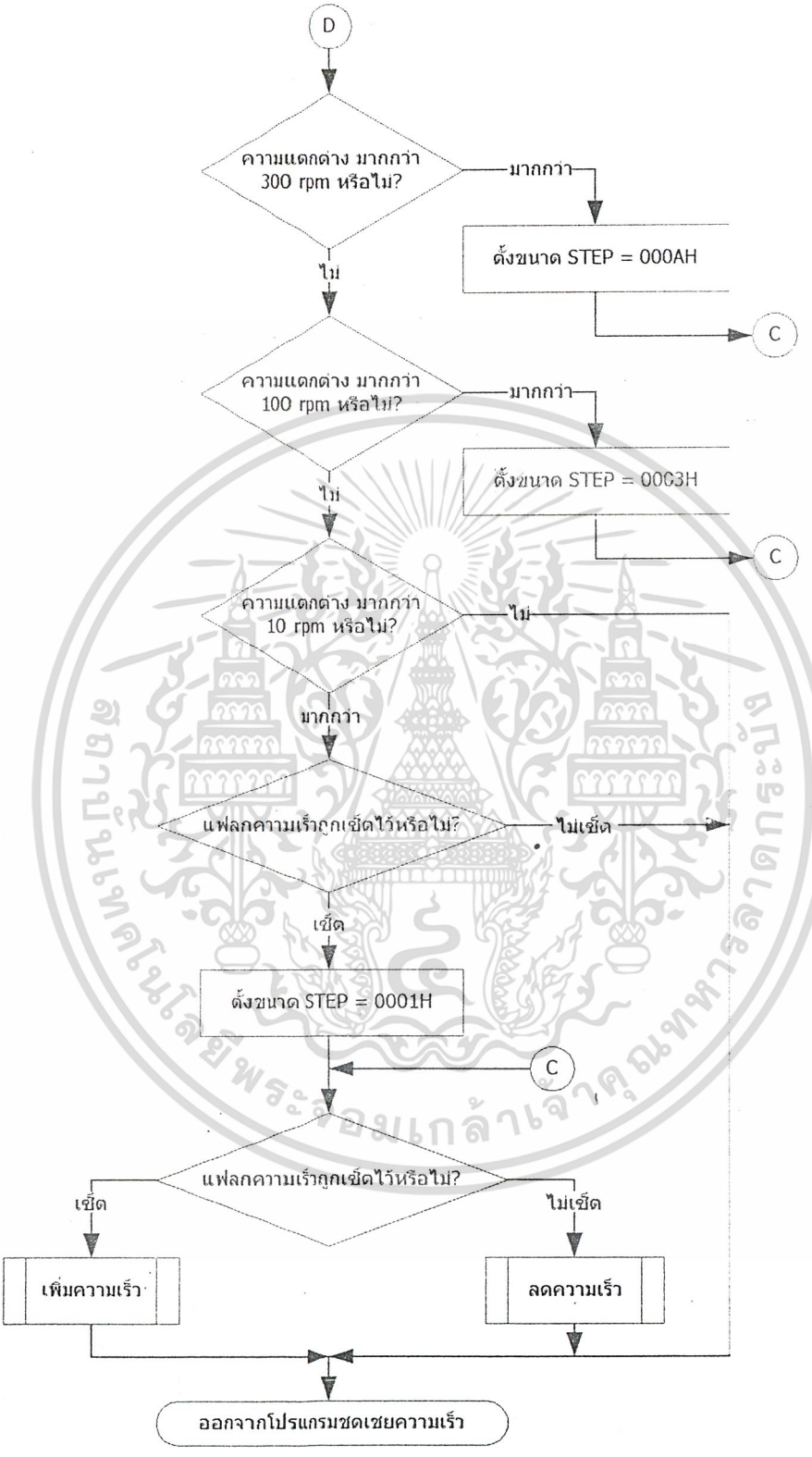
รูปที่ 6.6 โฟลวชาร์ตของโปรแกรมย่อยซอฟท์สตาร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



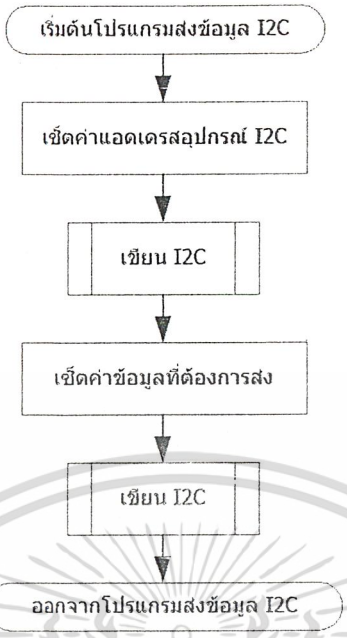
รูปที่ 6.7 โฟลวชาร์ตโปรแกรมย่อยส่วนชดเชยความเร็วมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

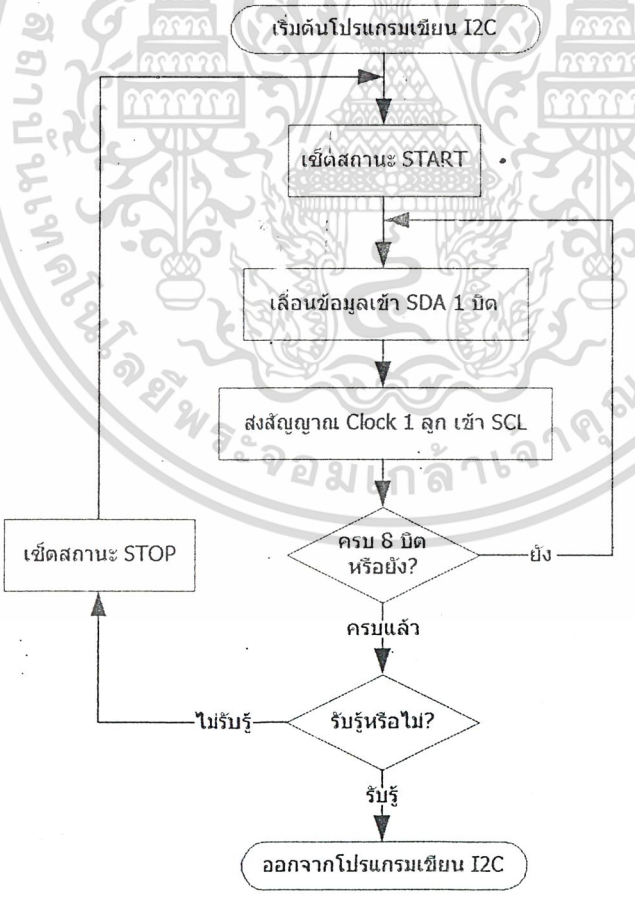


รูปที่ 6.7 โฟลวชาร์ตโปรแกรมย่อยส่วนชดเชยความเร็วมอเตอร์ (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

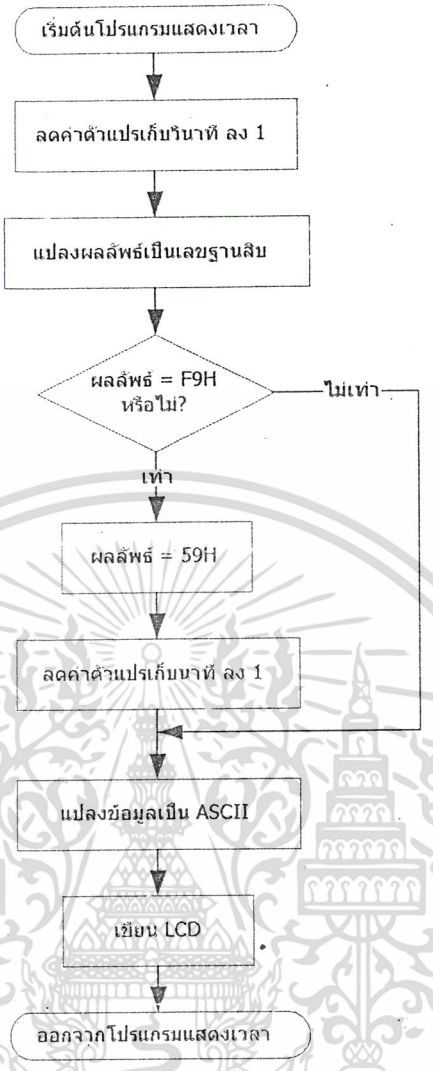


รูปที่ 6.8 โฟลวชาร์ตโปรแกรมย่อยส่งข้อมูล I2C

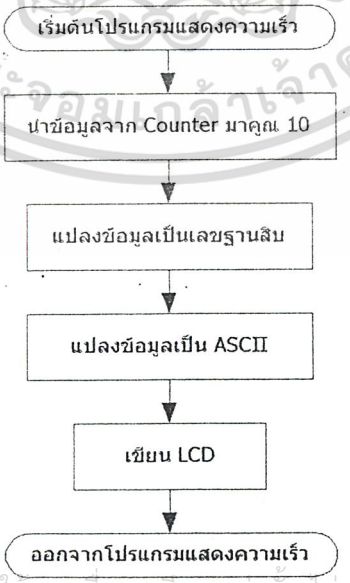


รูปที่ 6.9 โฟลวชาร์ตโปรแกรมย่อยเขียน I2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.10 โฟลวชาร์ตโปรแกรย่อยแสดงเวลา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.11 โฟลวชาร์ตโปรแกรมย่อยแสดงความเร็ว

6.2.4 ขั้นตอนการจบการทำงาน โดยมีโฟลวชาร์ตดังรูปที่ 6.12 และขั้นตอนการทำงานดังนี้

1) หยุดการทำงานของ DC Relay โดยการส่งข้อมูลลอจิก "1" ไปยังพอร์ต P2.4 และหยุดการทำงานของ AC Relay โดยการส่งข้อมูลลอจิก "1" ไปยังพอร์ต P2.3

2) แสดงข้อความบนจอ LCD ดังนี้

- หากจบการทำงานด้วยการกดปุ่ม CANCEL ค้าง 5 วินาที

  ‘ CANCEL! ‘

  ‘ Wait for motor stopped ‘

- หากจบการทำงาน เมื่อเวลาหมด

  ‘ FINISHED! ‘

  ‘ Wait for motor stopped ‘

3) รอการหยุดหมุนของมอเตอร์ โดยการตรวจสอบรอบความเร็วของมอเตอร์ด้วย Timer 0 ที่เป็นตัว Counter ไปเรื่อย ๆ จนกว่ามอเตอร์จะหยุดหมุน

4) เมื่อมอเตอร์หยุดหมุนให้ส่งข้อมูลลอจิก "0" ไปยังพอร์ต P2.5 เพื่อให้ BUZZER ทำงาน ส่งเสียงให้ผู้ใช้รับรู้ว่า การทำงานเสร็จแล้ว และแสดงข้อความบนจอ LCD ดังนี้

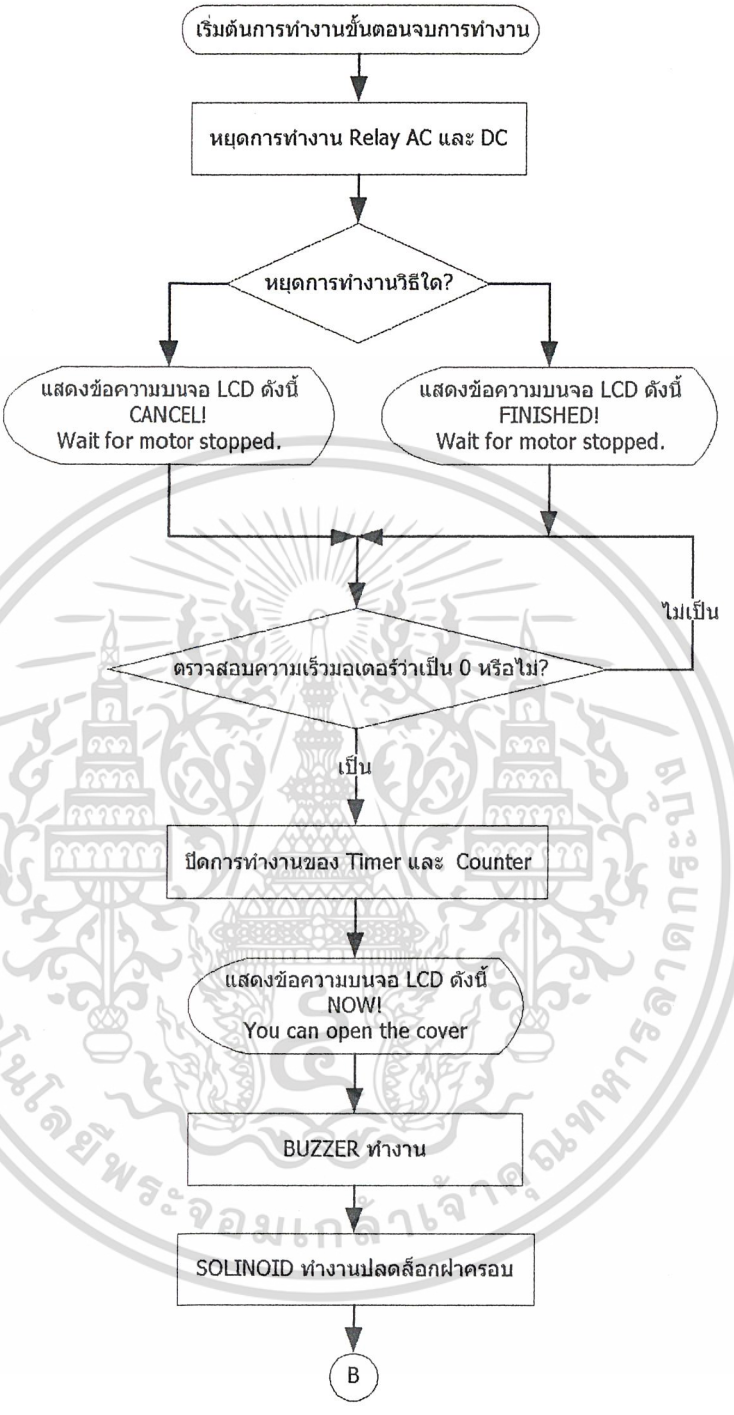
  ' N O W ! '

  ' You can open the cover '

5) SOLINOID ทำงาน

6) จากนั้นโปรแกรมจะกลับไปยังขั้นตอนการเริ่มต้นการทำงานในลำดับที่ 5 คือทำการรอการเปิดฝาครอบ และจากนั้นก็จะทำงานตามปกติเหมือนเดิมต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.12 โฟลวชาร์ตขั้นตอนการจบการทำงาน

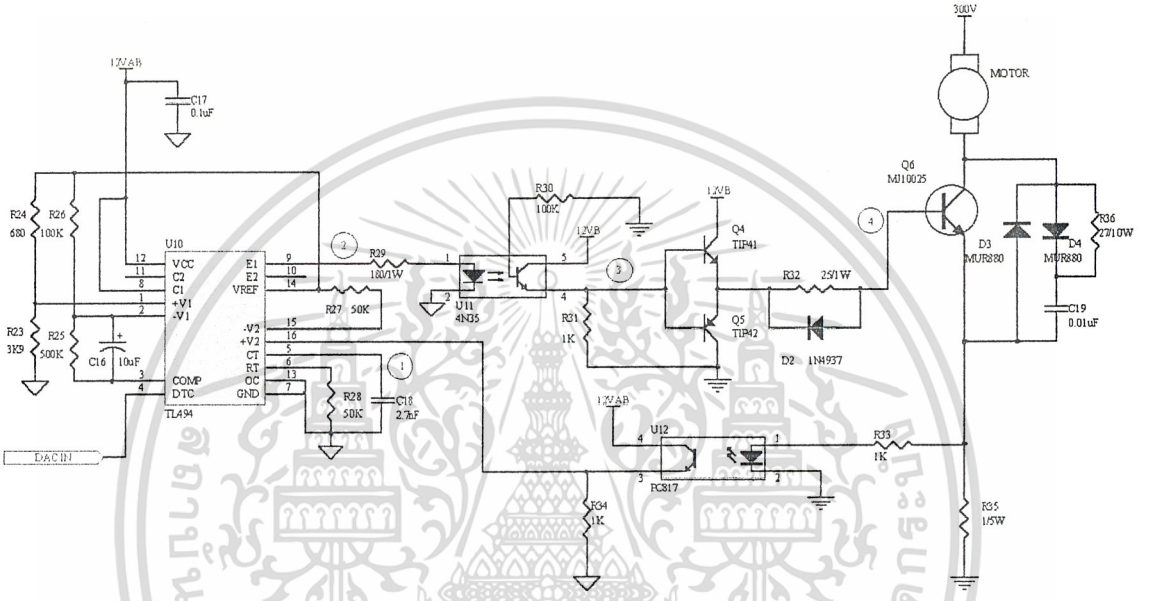
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

การทดลองและผลการทดลอง

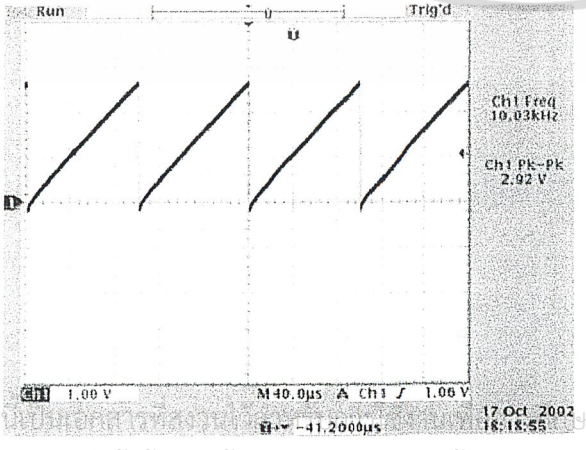
7.1 การทดลองภาคควบคุมการขับมอเตอร์

7.1.1 ต่อวงจรกำเนิดสัญญาณ Pulse Width Modulation (PWM) ดังรูปที่ 7.1



รูปที่ 7.1 วงจรควบคุมและขับมอเตอร์

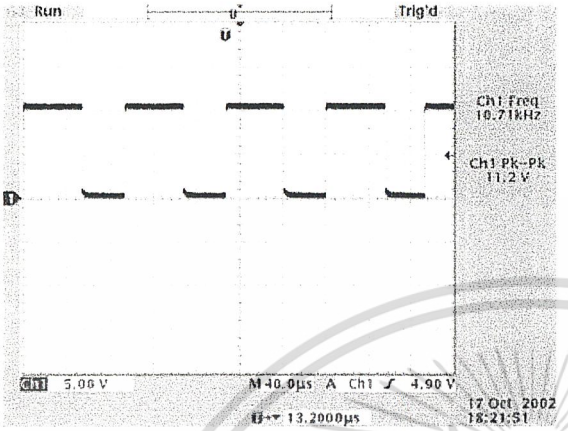
7.1.2 วัดสัญญาณที่ขา 5 ของ TL494 (จุดที่ 1 ของรูปที่ 7.1) จะได้เป็นรูป Saw Tooth ดังรูปที่ 7.2 ซึ่งมีความถี่เท่ากับ 10 kHz ตามที่คำนวณ ค่าความถี่ที่ได้นี้สามารถปรับเปลี่ยนได้โดยการเปลี่ยนค่าของ R_T หรือ C_T หรือสามารถดูเทียบได้จากกราฟในรูปที่ 4.3



รูปที่ 7.2 รูปสัญญาณที่ขา 5 ของ TL494

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.3 วัดสัญญาณที่เอาท์พุทของ TL494 ขา 9 (จุดที่ 2 ของรูปที่ 7.1) จะได้รูปสัญญาณดังรูปที่ 7.3



รูปที่ 7.3 สัญญาณที่ Output ขา 9 ของ TL494

จากการทดลองเราสามารถปรับความกว้างของพัลส์ได้โดยการปรับระดับแรงดันที่ DAC IN

7.1.4 ทดลองวงจรส่วน Soft Start โดยใช้หลักการของวงจรอินทิเกรเตอร์ โดยใช้ Error Amp ในตัว TL494 ร่วมกับ RC ภายนอกดังรูปที่ 7.1 จะพบว่า เมื่อเริ่มเปิดเครื่องจะไม่มีพัลส์ออกที่เอาท์พุท (ขา 9) ของ TL494 เลยจนกระทั่งเวลาผ่านไปเล็กน้อย ก็จะเริ่มมีพัลส์เกิดขึ้น แล้วความกว้างของพัลส์ก็จะค่อย ๆ ขยายกว้างขึ้นจนถึงค่า ๆ หนึ่งที่เราตั้งค่าอ้างอิงไว้ ซึ่งค่าอ้างอิงนี้เป็นค่าแรงดันอ้างอิงที่ได้จากวงจร DAC เช่นกัน เพราะว่าถ้าที่เอาท์พุทของ Error Amp มีค่าแรงดันปรากฏอยู่ จะทำให้ TL494 ไม่ทำงาน ซึ่งถ้าดูจากรูปวงจรจะพบว่า ช่วงแรกที่วงจรทำงาน จะมีแรงดัน V_o ประมาณ 4.3 V ซึ่งเกิดจาก R Divider ที่ขา Non-inverting แต่เมื่อเวลาผ่านไป C จะทำการเก็บประจุจนมีค่ามากกว่าแรงดันที่ขา Non-inverting ก็จะทำให้ V_o ค่อย ๆ ลดลงตามการเก็บประจุของ C ทำให้ TL494 ค่อย ๆ ปล่อยพัลส์ออกที่เอาท์พุท (ขา 9) โดยพัลส์จะค่อย ๆ ขยายกว้างขึ้น ซึ่งการเกิดพัลส์ และความกว้างของพัลส์จะขึ้นอยู่กับค่า C ด้วย เพราะว่าถ้า C มีค่ามาก การเกิดพัลส์จะช้าและพัลส์ก็จะค่อย ๆ ขยายกว้างขึ้นด้วย

7.2 การทดลองภาคขับมอเตอร์

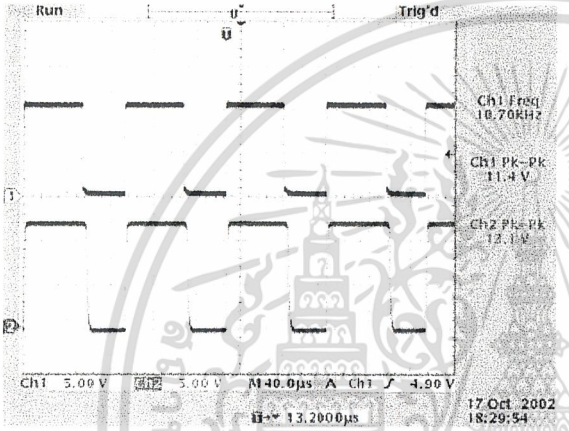
7.2.1 ต่อวงจรภาคควบคุมและภาคขับมอเตอร์ดังรูปที่ 7.1

7.2.2 ในการทดลองครั้งแรก ๆ นั้น ยังไม่ได้ใส่ชุด soft start เมื่อทำการเปิดเครื่อง จึงทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

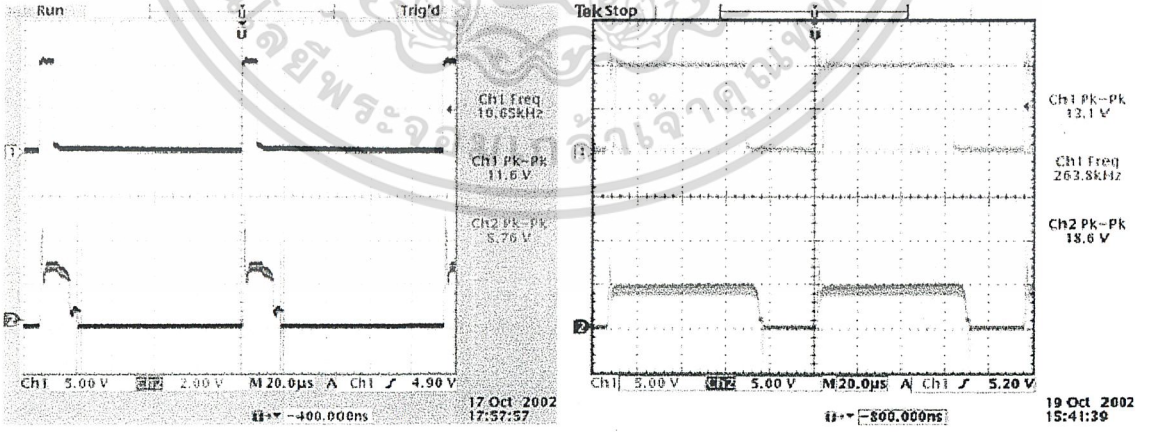
ให้เกิดการ ON อย่างทันทีทันใด ทำให้มีกระแสสูงไหลผ่านทรานซิสเตอร์ ทำให้ทรานซิสเตอร์ลัดวงจร แต่เมื่อใส่แล้วสามารถเปิดเครื่องได้โดยทรานซิสเตอร์ไม่ลัดวงจร แต่ในการเปิดเครื่องนั้นจะต้องป้อนไฟจ่ายไปรอที่มอเตอร์ก่อนแล้วจึงทำการ ON ชุดควบคุม เพราะถ้าไม่ทำแบบนี้ ในช่วงที่จ่ายไฟเข้าพร้อมกันจะทำให้ชุด soft start ทำงานไปก่อนแล้วจะทำให้มีพัลส์ไปรอที่ขาเบสของทรานซิสเตอร์เลย ทำให้เกิดกระแสกระชาก ส่งผลให้ทรานซิสเตอร์ลัดวงจรได้อีก

7.2.3 ทำการวัดสัญญาณที่ขาเอาท์พุทของออปโต้ ขา 4 (จุดที่ 3 ของรูปที่ 7.1) เทียบกับขาเอาท์พุทของ PWM ขา 9 (จุดที่ 2 ของรูปที่ 7.1) ของทรานซิสเตอร์จะได้สัญญาณดังนี้



รูปที่ 7.4 รูปสัญญาณเอาท์พุทของออปโต้เทียบกับเอาท์พุทของ PWM
Ch1 รูปสัญญาณเอาท์พุทของ PWM
Ch2 รูปสัญญาณเอาท์พุทของออปโต้

7.2.4 ทำการวัดสัญญาณที่ขาเบสของทรานซิสเตอร์ (จุดที่ 4 ของรูปที่ 7.1) เทียบกับสัญญาณเอาท์พุทของ PWM (จุดที่ 1 ของรูปที่ 7.1) ได้สัญญาณดังรูปที่ 7.5



(ก) (ข)

รูปที่ 7.5 รูปสัญญาณที่ขาเบสของทรานซิสเตอร์เทียบกับสัญญาณเอาท์พุทของ PWM
(ก) รูปสัญญาณที่ขาเบสที่ความเร็วต่ำ
(ข) รูปสัญญาณที่ขาเบสที่ความเร็วสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3 การทดลองชุดตรวจจับกระแสเกิน

7.3.1 ต่อวงจรโดยใช้ Error Amp อีกตัวหนึ่งของ TL494 มาทำเป็นวงจร Comparator ดังรูปที่ 7.1

7.3.2 จากวงจรจะพบว่า ที่ขา inverting จะมีแรงดันอยู่ประมาณ 5 V ซึ่งได้จากการต่อความต้านทาน 50k จากขา Vref มาเข้าที่ขา inverting ของ error amp ภายในตัวของ TL494

7.3.3 โดยปกติแล้ว ถ้าแรงดันที่ขา inverting มากกว่าขา Non-inverting จะทำให้ได้เอาท์พุทเป็น Low จึงส่งผลให้ TL494 ยังคงทำงานอยู่ แต่เมื่อใดที่มีกระแสไหลผ่าน R_{35} จนมีค่ามากกว่า 2 A (จากการคำนวณ) จะส่งผลให้ Op-to couple ทำงานเกิดมีแรงดันตกคร่อมความต้านทานที่ขา Non-inverting (ประมาณ 5 V) ซึ่งจะมีแรงดันมากกว่าขา inverting ส่งผลให้ TL494 จะหยุดจ่ายพัลส์ ซึ่งเป็นการตัดวงจรการทำงานของทรานซิสเตอร์ ทำให้ไม่เกิดความเสียหายแก่วงจร หรือถ้าเกิดก็น้อยลง

7.4 การทดลองวัดกระแสที่ไหลผ่านมอเตอร์ เทียบกับความเร็วรอบที่ได้

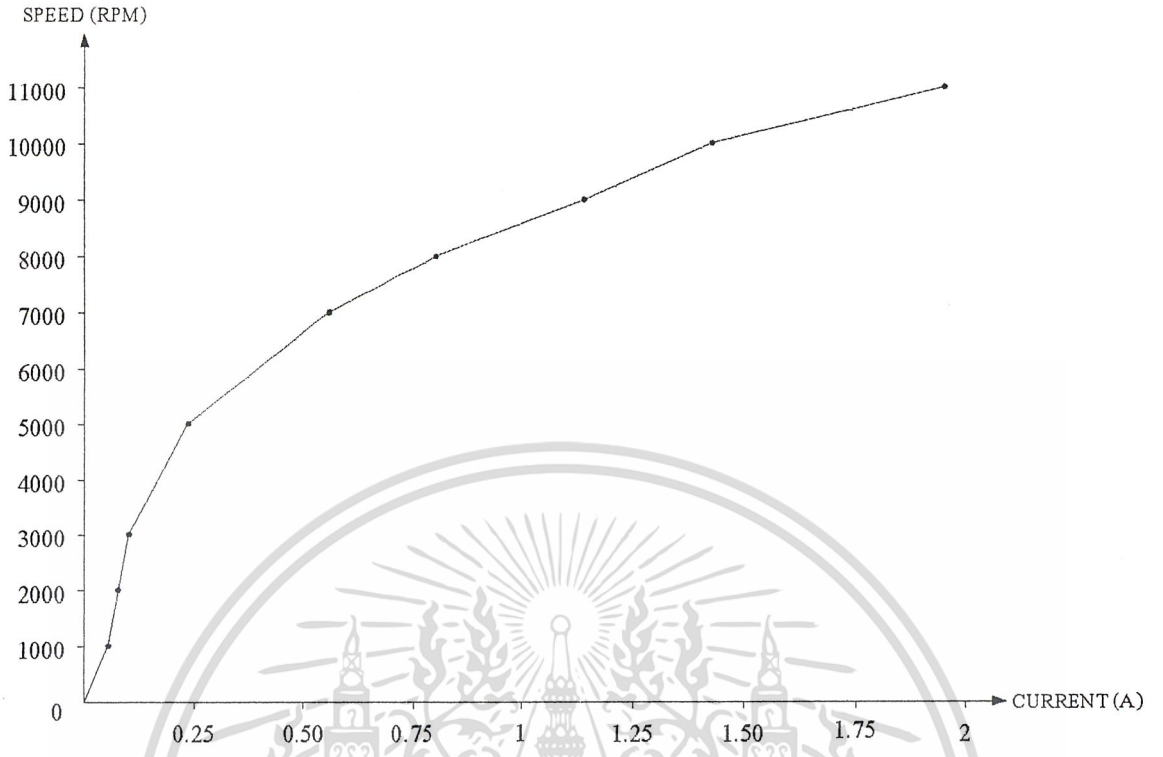
7.4.1 ต่อวงจรแบบเดียวกับการทดลองที่ 7.1

7.4.2 ใช้ VOM วัดคร่อม R_{35} ทำการปรับค่าระดับแรงดัน DAC ที่ป้อนเข้าที่ขา Dead Time Control ของ TL494 แล้วใช้ Tachometer วัดรอบมอเตอร์ ซึ่งจะได้ความสัมพันธ์ดังตารางต่อไปนี้

Speed (rpm.)	แรงดันที่ตกคร่อม R_{35} (V)	กระแสที่ไหลผ่านมอเตอร์ (A)
11,000	1.950	1.950
10,000	1.430	1.430
9,000	1.070	1.070
8,000	0.810	0.810
7,000	0.560	0.560
5,000	0.240	0.240
3,000	0.102	0.102
2,000	0.073	0.073
1,000	0.054	0.054

ตารางที่ 7.1 แสดงความเร็วและแรงดันตกคร่อมมอเตอร์ค่าต่าง ๆ โดยใช้ Digital Tachometer วัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.6 กราฟแสดงความสัมพันธ์ระหว่างความเร็วกับแรงดันของมอเตอร์

7.5 การทดลองวัดการเพิ่มความเร็วรอบมอเตอร์เทียบกับเวลา

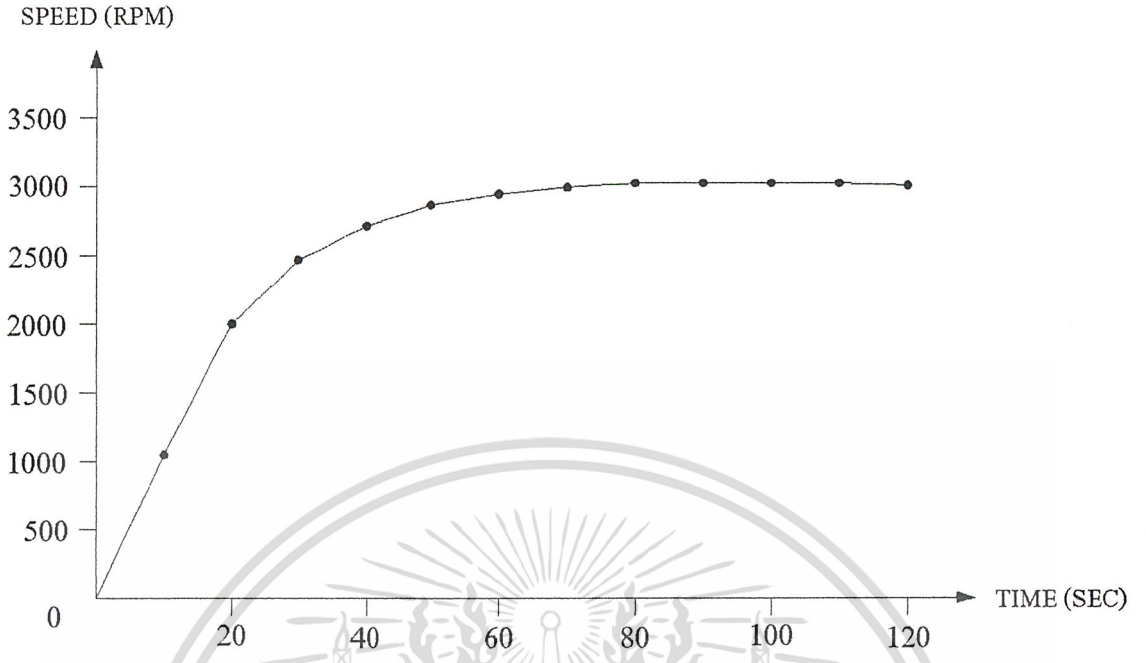
7.5.1 ทำการทดสอบโปรแกรมที่เขียนขึ้น โดยทดสอบการทำงานของเครื่องปั่นทั้ง 2 โหมด และทำการบันทึกค่าความเร็วที่ค่าเวลาต่างๆ ลงในตาราง

7.5.2 ผลการทดลองแสดงดังตารางที่ 7.2 และ 7.3

Time (Sec.)	Speed (rpm)	Time (Sec.)	Speed (rpm)
0	0	60	2960
10	1050	70	3010
20	1990	80	3040
30	2470	90	3050
40	2730	100	3030
50	2870	110	3030

ตารางที่ 7.2 แสดงความสัมพันธ์ระหว่างความเร็วมอเตอร์กับเวลา ในโหมด CENTRIFUGE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ดัดแปลง เผยแพร่ หรือแสวงหาประโยชน์ทางการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

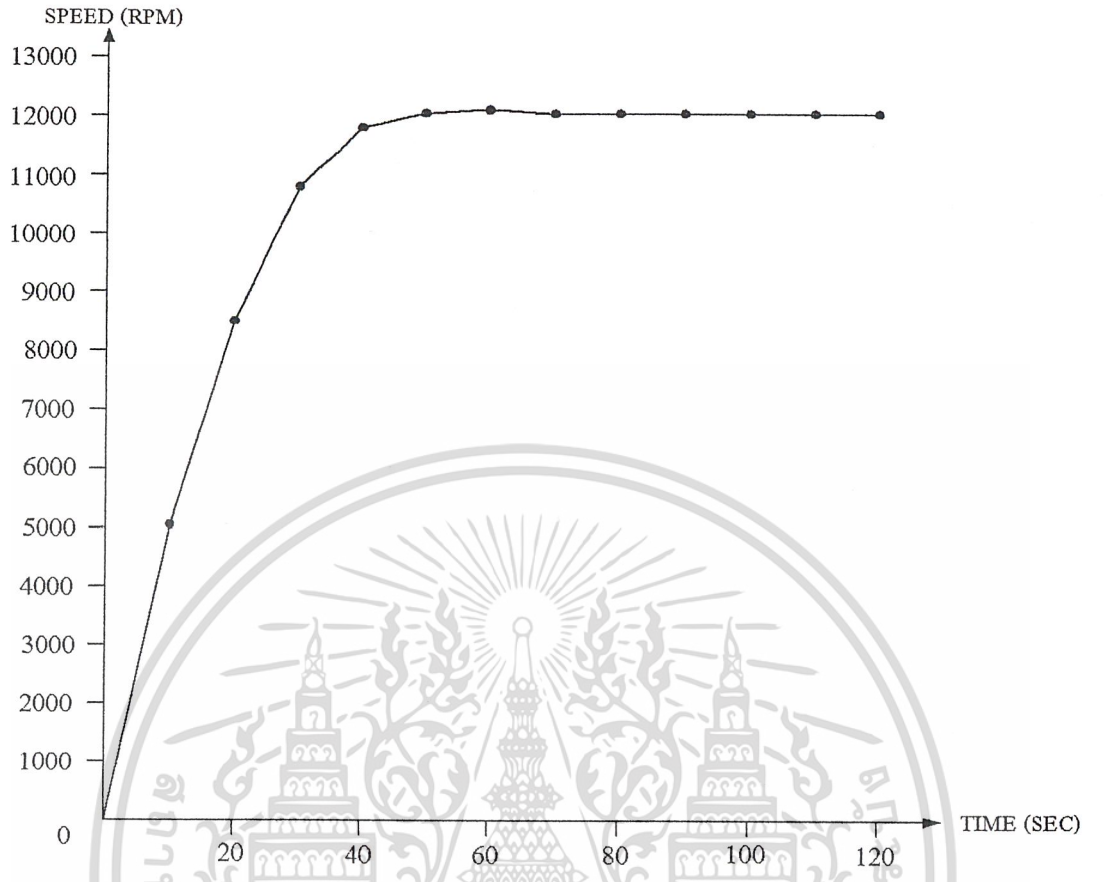


รูปที่ 7.7 กราฟแสดงความสัมพันธ์การเพิ่มความเร็วของมอเตอร์กับเวลาในโหมด CENTRIFUGE

Time (Sec.)	Speed (rpm)	Time (Sec.)	Speed (rpm)
0	0	60	12120
10	5100	70	12100
20	8580	80	12090
30	10840	90	12090
40	11880	100	12080
50	12090	110	12080

ตารางที่ 7.2 แสดงความสัมพันธ์ระหว่างความเร็วมอเตอร์กับเวลา ในโหมด HEMATOCRIT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.7 กราฟแสดงความสัมพันธ์การเพิ่มความเร็วของมอเตอร์กับเวลาในโหมด HEMATOCRIT

7.6 การทดลองวัดความเร็วรอบด้วยโปรแกรมที่เขียนขึ้นเทียบกับเครื่องวัดความเร็วรอบของ YOKOGAWA

7.6.1 ที่โหมด Centrifuge ความเร็วประมาณ 3000 rpm

โปรแกรมที่เขียนขึ้นอ่านความเร็วได้ 3000 rpm

เครื่องวัด YOKOGAWA อ่านความเร็วได้ 2990 rpm

ค่าความผิดพลาด $\frac{3000 - 2990}{2990} \times 100 =$ 0.33%

7.6.2 ที่โหมด Hematocrit ความเร็วประมาณ 12000 rpm

โปรแกรมที่เขียนขึ้นอ่านความเร็วได้ 12090 rpm

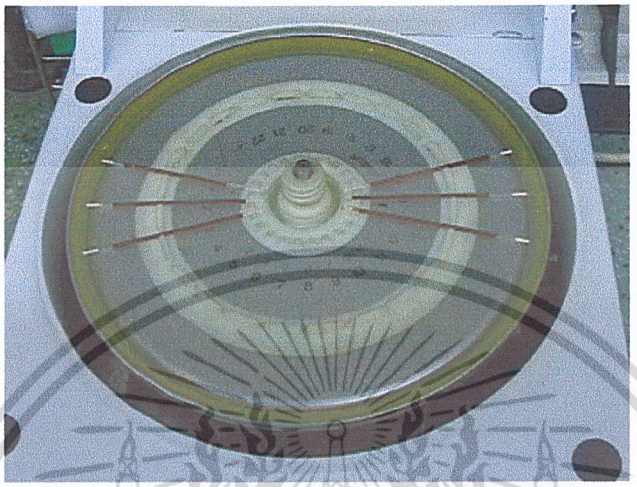
เครื่องวัด YOKOGAWA อ่านความเร็วได้ 12060 rpm

ค่าความผิดพลาด $\frac{12090 - 12060}{12060} \times 100 = 0.25\%$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.7 การทดลองปั่นเลือดจริงด้วยเครื่องที่สร้างขึ้นเทียบกับเครื่องที่โรงพยาบาลใช้อยู่

7.7.1 โหมด Hematocrit



รูปที่ 7.8 แสดงหลอดเลือดก่อนทำการปั่นโดยใช้โหมด Hematocrit

(ก) (ข)

รูปที่ 7.9 แสดงวัดค่าอัตราส่วนของหลอดเลือดหลังทำการปั่น

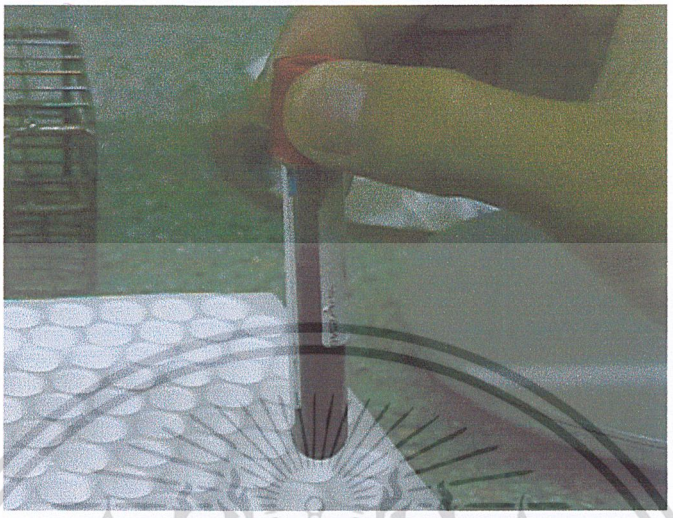
(ก) ทำการปั่นโดยใช้เครื่องของทางโรงพยาบาลเอง

(ข) ทำการปั่นโดยใช้เครื่องที่สร้างขึ้น

จากผลการทดลองดังรูปที่ 7.9 จะเห็นว่าผลของการวัดค่าอัตราส่วนของหลอดเลือดที่ทำการปั่นโดยเครื่องของทางโรงพยาบาล และเครื่องที่สร้างขึ้นได้ค่าอัตราส่วนที่เท่ากัน คือ 40% แสดงว่าเครื่องปั่นฯที่สร้างขึ้นสามารถใช้งานในการปั่นในโหมด Hematocrit ได้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.7.2 โหมด Centrifuge



รูปที่ 7.10 แสดงเลือดก่อนทำการปั่น

รูปที่ 7.11 แสดงเลือดหลังทำการปั่น

จากการทดลองปั่นเลือดโดยใช้โหมด Centrifuge จะทำให้เลือดตกตะกอน ซึ่งเครื่องที่สร้างขึ้นเมื่อใช้ปั่นแล้วสามารถทำให้เลือดตกตะกอนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

บทสรุป

เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมาเป็นเครื่องมือทางการแพทย์ชนิดหนึ่ง โดยอาศัยหลักการของแรงหนีศูนย์กลาง นำมาใช้ในการแยกเม็ดเลือดออกจากพลาสมา ซึ่งโดยทั่วไปแล้วจะต้องมีการนำเครื่องมือดังกล่าวเข้ามาจากต่างประเทศจึงทำให้มีราคาค่อนข้างสูง นอกจากนี้การทำงานของเครื่องจะมีลักษณะการทำงานเป็นแบบกำหนดเอง ซึ่งอาจทำให้เกิดความผิดพลาดในระหว่างการทำงานได้ จากข้อเสียที่กล่าวมานี้ จึงทำให้เกิดแนวความคิด ในการสร้างเครื่องปั่นแยกเม็ดเลือด ที่มีระบบการทำงานแบบกึ่งอัตโนมัติ โดยการประยุกต์ใช้วงจรอิเล็กทรอนิกส์ และไมโครคอนโทรลเลอร์มาควบคุมการทำงานของเครื่อง เพื่อให้สามารถปรับความเร็วของมอเตอร์ได้ตามต้องการ และยังสามารถทำงานในโหมดที่มีความเร็วรอบแตกต่างกัน ทำให้สะดวกในการใช้งาน

ในโครงงานนี้จะใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89S8252 มาเป็นตัวควบคุมการทำงานของเครื่องทั้งหมด โดยทำงานร่วมกับจอแสดงผล LCD เพื่อแสดงผลการทำงานของเครื่อง รวมถึงการแสดงค่าพารามิเตอร์ต่างๆ และยังมีการทำงานร่วมกับไอซีขยายพอร์ตแบบ I^2C ซึ่งต่อร่วมกับวงจร R-2R Ladder เพื่อสร้างเป็นวงจร DAC ขนาด 16 bit ซึ่งมีความละเอียดสูง เพื่อใช้ในการควบคุมความกว้างของพัลส์ที่นำไปควบคุมชุดขับมอเตอร์ จากการทดลองพบว่า ที่ความเร็วรอบต่ำๆ จะต้องการแรงดัน DAC ที่มีความละเอียดสูงมากๆ ส่วนความเร็วรอบสูง การเปลี่ยนแปลงของแรงดันดังกล่าวเพียงเล็กน้อยไม่ค่อยมีผลกระทบต่อความเร็วรอบมอเตอร์เท่าใดนัก

เมื่อเริ่มต้นการทำงานของเครื่อง จะมีการส่งข้อมูลไปยังไอซีขยายพอร์ต เพื่อแปลงเป็นแรงดันควบคุมความกว้างของพัลส์ พบว่าจุดที่มอเตอร์เริ่มทำงานมีค่าแรงดัน DTC ประมาณ 2.5 V จากนั้นจะใช้โปรแกรมย่อย soft start เพื่อทำให้มอเตอร์ค่อยๆเพิ่มความเร็วขึ้นจนไปถึงระดับความเร็วที่ต้องการเพื่อป้องกันความเสียหายของภาคขับมอเตอร์ จากนั้นจะรอสักครู่เพื่อรอให้ความเร็วรอบมอเตอร์เพิ่มขึ้นจนเกือบคงที่ ทั้งนี้เนื่องจากแรงเฉื่อยของมอเตอร์ หลังจากนั้นจะเป็นการชดเชยความเร็วของมอเตอร์เพื่อให้อยู่ในระดับที่ต้องการ

การทดสอบการวัดการเพิ่มความเร็วของมอเตอร์ทั้ง 2 โหมด พบว่า การทำงานของทั้ง 2 โหมดใช้เวลาในการเพิ่มความเร็วไปจนถึงจุดอ้างอิงไม่ถึง 1 นาที และจากการทดสอบวัดความเร็วรอบมอเตอร์โดยใช้โปรแกรมที่เขียนขึ้นเทียบกับเครื่องมือวัดความเร็วรอบที่ได้รับการรับรองมาตรฐานและมีขายทั่วไป ในโครงงานนี้ใช้เครื่องวัดของ YOKOGAWA พบว่าความเร็วที่อ่านได้จากโปรแกรมที่เขียนขึ้นมีความแตกต่างจากเครื่องมือวัดของ YOKOGAWA เพียง 0.2-0.4 % เท่านั้น

การทดสอบประสิทธิภาพของเครื่องที่สร้างขึ้นเทียบกับเครื่องที่ใช้งานจริงของโรงพยาบาลจุฬาฯ ในโหมด Centrifuge พบว่าเครื่องที่สร้างขึ้นสามารถปั่นให้เลือดตกตะกอนได้เช่นเดียวกับเครื่องของทางโรงพยาบาล ส่วนในโหมด Hematocrit พบว่า อัตราส่วนของเม็ดเลือดเทียบกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พลาสมาที่อ่านได้หลังจากทำการปั่นด้วยเครื่องที่สร้างขึ้น มีความถูกต้องและใกล้เคียงกับเครื่องที่โรงพยาบาลใช้ รวมไปถึงเครื่อง ออโต้เมดเตด ซึ่งเป็นเครื่องที่ใช้สำหรับวิเคราะห์เม็ดเลือดอัตโนมัติ

ปัญหาอุปสรรคที่พบในการทำโครงงานนี้ พบว่า ประสิทธิภาพของมอเตอร์ลดลง โดยมอเตอร์ใช้กระแสในขับสูงขึ้นมากกว่าคุณสมบัติเดิมที่ความเร็วรอบเท่ากัน ทั้งยังมีกลิ่นเหม็นในขณะความเร็วรอบสูง จึงแก้ไขโดยการนำมอเตอร์ไปพันใหม่ ผลที่ได้คือทำให้มอเตอร์มีประสิทธิภาพดีขึ้น กลิ่นเหม็นหายไป หลังจากนั้นเป็นการทดสอบการทำงานของเครื่อง พบว่า ที่ความเร็วรอบสูงการนับความเร็วรอบมอเตอร์เกิดความผิดพลาดขึ้น ทั้งนี้เนื่องจากออปโต้ที่ใช้ในวงจรตรวจจับความเร็วรอบมอเตอร์มีกระแสไบอัสสูงเกินไป จึงทำการแก้ไข โดยการลดกระแสไบอัสลง ผลที่ได้คือ สามารถนับความเร็วได้สูงขึ้น

จากการทดสอบทั้งหมด สามารถกล่าวได้ว่า เครื่องปั่นเม็ดเลือดที่สร้างขึ้น สามารถนำไปใช้งานได้จริง การใช้งานก็ไม่ยุ่งยาก และสามารถพัฒนาให้มีประสิทธิภาพหรือความสะดวกในการใช้งานได้อีกด้วย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ทวาย สำเร็จ , " คู่มือซ่อมมอเตอร์ไฟฟ้า " พิมพ์ครั้งที่ 3 . สุนีย์การพิมพ์ : กรุงเทพฯ , 2520
2. ไชยชาญ หินเกิด , " เครื่องกลไฟฟ้ากระแสตรง " พิมพ์ครั้งที่ 7 . สมาคมส่งเสริมเทคโนโลยี (ไทย – ญี่ปุ่น) : กรุงเทพฯ , 2542
3. วิชัย ศังขจันทรานนท์ , " ควบคุมเครื่องกลไฟฟ้าด้วยอิเล็กทรอนิกส์กำลัง 1 " สมาคมส่งเสริมเทคโนโลยี (ไทย – ญี่ปุ่น) : กรุงเทพฯ , 2532
4. วัฒนชัย เพ็ญอัมพรกุล และ วัลลพ สกุลอิ่มทรัพย์ , " ปริญญานิพนธ์เครื่องปั่นแยกเม็ดเลือดออกจากพลาสมา " สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2544
5. Muhammad Harunur Rashid , " Power Electronics " Prentice Hall , 1988
6. อุดม จีนประดับ , "ไมโครคอนโทรลเลอร์ MCS-51" พิมพ์ครั้งที่ 1 . ศูนย์ผลิตตำราสถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ : กรุงเทพฯ , 2541
7. วรพจน์ กรแก้ววัฒนกุล และ ชัยวัฒน์ ลิ้มพรจิตรวิไล (เรียบเรียง) , " เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช " บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด : กรุงเทพฯ , 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งานเครื่องปั่นแยกเม็ดเลือดแดงออกจากพลาสมา

เครื่องปั่นแยกเม็ดเลือดแดงออกจากพลาสมานี้ สามารถที่จะทำงานได้ 2 โหมดดังนี้คือ

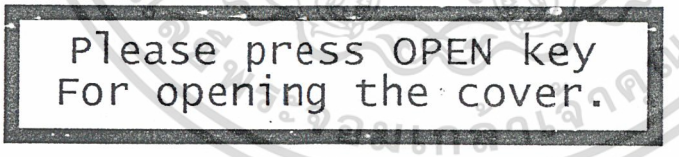
1. โหมดการปั่นแยกเม็ดเลือดแดงออกจากพลาสมา (HEMATOCRIT MODE) จะใช้ถาดที่ใส่หลอดเลือดขนาดเล็กในการปั่น
2. โหมดการปั่นเลือดตกตะกอน (CENTRIFUGE MODE) จะใช้ถาดที่ใส่หลอดเลือดขนาดใหญ่ในการปั่น

ดังนั้นในการใช้งาน จะต้องระวังเรื่องถาดที่ใส่หลอดเลือดด้วย โดยต้องใช้ให้ตรงกับโหมดการทำงานนั้น ๆ มิฉะนั้นจะทำให้เครื่องเสียหายได้

ขั้นตอนการใช้เครื่อง

1. เปิดสวิตช์ **POWER** ที่เครื่อง เพื่อเปิดใช้งานเครื่อง เมื่อเปิดแล้ว ที่หน้าจอ LCD จะแสดงผลดังนี้

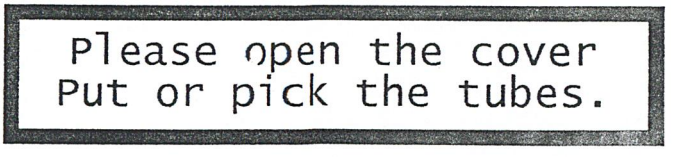
2. จากนั้น 2 วินาที ที่หน้าจอ LCD จะแสดงผลดังนี้



ทำการกดปุ่ม **OPEN** เพื่อปลดล็อคฝาครอบออก

ถ้าหากมีการเปิดฝาครอบค้างไว้ก่อนเปิดเครื่อง หลังจากขั้นตอนที่ 1 แล้ว จะข้ามไปยังขั้นตอนที่ 4

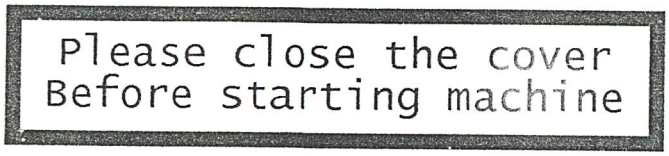
3. เมื่อกดปุ่ม **OPEN** แล้ว ที่หน้าจอ LCD จะแสดงผลดังนี้



ทำการเปิดฝาครอบออก เพื่อใส่หรือหยิบหลอดเลือด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. จากนั้น ที่หน้าจอ LCD จะแสดงดังนี้



ทำการปิดฝาครอบอีกครั้งหนึ่ง เพื่อเข้าสู่ขั้นตอนการเลือกโหมดการทำงาน

5. เมื่อทำการปิดฝาครอบแล้ว ที่หน้าจอ LCD จะแสดงดังนี้

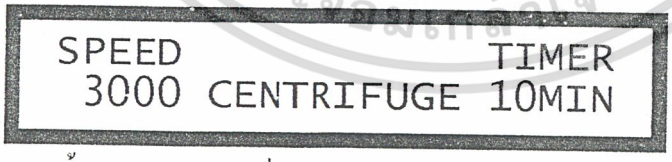
ทำการเลือกโหมดการทำงานด้วยการกดปุ่ม SEL โหมดการทำงานที่ถูกเลือกจะกระพริบ และทำการเข้าสู่โหมดการทำงานที่เลือกไว้ ด้วยการกดปุ่ม ENT

ก่อนกดปุ่ม ENT ถ้าหากต้องการเปลี่ยนถาดใหม่ ในขณะนี้สามารถทำได้ด้วยการกดปุ่ม OPEN อีกครั้ง การทำงานจะกลับไปขั้นตอนที่ 3

6. เมื่อทำการเลือกโหมดการทำงาน และเข้าสู่โหมดการทำงานที่เลือกนั้นแล้ว ที่หน้าจอ LCD จะแสดงดังนี้

- ถ้าเลือก และเข้าสู่โหมด HEMATOCRIT จะแสดงดังนี้

- ถ้าเลือก และเข้าสู่โหมด CENTRIFUGE จะแสดงดังนี้



จากนั้น - ถ้าต้องการสั่งให้ทำงาน กดปุ่ม ENT ก็จะเข้าสู่การทำงานในขั้นตอนข้อที่ 7

- ถ้าต้องการยกเลิกการเลือกใช้โหมดนี้ กดปุ่ม CANCEL และเมื่อกดปุ่มนี้แล้ว จะกลับทำงานในขั้นตอนข้อที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อกดปุ่ม ENT ค้างไว้ประมาณ 1 วินาที หลังจากนั้นเครื่องจะเริ่มทำงาน ที่หน้าจอ LCD จะแสดงดังนี้

   - ถ้าเลือก และเข้าสู่โหมด HEMATOCRIT จะแสดงดังนี้

   แสดงว่าเข้าสู่การทำงานแล้ว

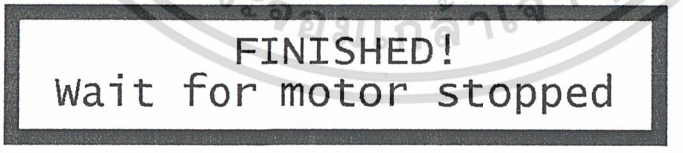
   - ถ้าเลือก และเข้าสู่โหมด CENTRIFUGE จะแสดงดังนี้

   แสดงว่าเข้าสู่การทำงานแล้ว และเวลาจะลดลงไปเรื่อยๆ

8. ในระหว่างการทำงาน สามารถหยุดการทำงานได้เลย โดยการกดปุ่ม CANCEL ค้างไว้ประมาณ5 วินาที เครื่องจะหยุดทำงาน หลังจากนั้นจะแสดงข้อความ

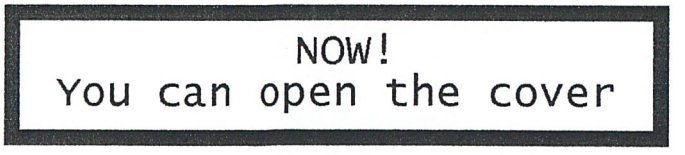
   เพื่อรอให้มอเตอร์หยุดหมุน และเข้าสู่ขั้นตอนที่ 10

9. เมื่อเครื่องทำงานไปเรื่อย ๆ จนกระทั่งหมดเวลาตามที่ตั้งไว้ ในโหมดนั้น ๆ เมื่อหมดเวลาแล้ว ที่หน้าจอ LCD จะแสดงดังนี้



   เป็นการรอการหยุดหมุนของมอเตอร์ ซึ่งตอนนี้จะยังไม่สามารถเปิดฝาครอบได้

10. เมื่อมอเตอร์หยุดหมุนแล้ว จะมีเสียงเตือน (ปี๊บ 2 วินาที) และที่หน้าจอ LCD จะแสดงดังนี้



   เมื่อเสียงเตือนดังเสร็จแล้ว สามารถเปิดฝาครอบเพื่อหยิบถาดออกได้ทันที หลังจากนั้นจะเข้าสู่ขั้นตอนที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่ในการใช้งานของแต่ละปุ่ม

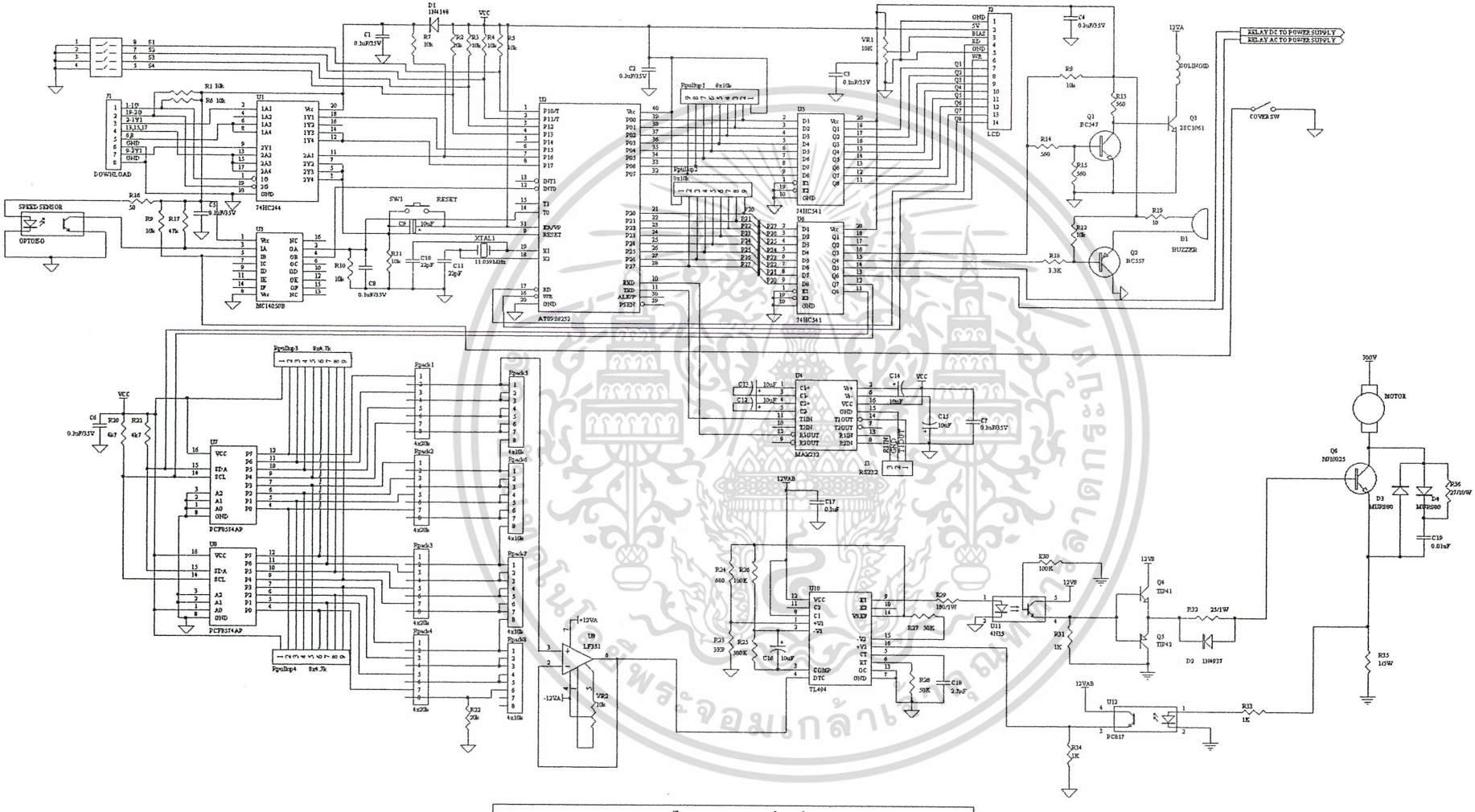
- **ปุ่ม OPEN** ทำหน้าที่สำหรับปลดล็อคฝาครอบ ให้สามารถเปิดได้ โดยปุ่มนี้จะไม่สามารถใช้งานได้ เมื่อเข้าสู่โหมดการทำงานใด โหมดการทำงานหนึ่งแล้ว และขณะเครื่องกำลังทำการปั่น
- **ปุ่ม CANCEL** ทำหน้าที่สำหรับยกเลิกโหมดการทำงานที่เลือกเข้าไป(ขณะเครื่องยังไม่ให้ทำการปั่น) และหยุดการทำงานของเครื่องแบบฉุกเฉิน
- **ปุ่ม SEL** (Select) ทำหน้าที่สำหรับเลือกโหมดการทำงาน
- **ปุ่ม ENT** (Enter) ทำหน้าที่สำหรับเข้าสู่โหมดการทำงานที่เลือกไว้ และเริ่มการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

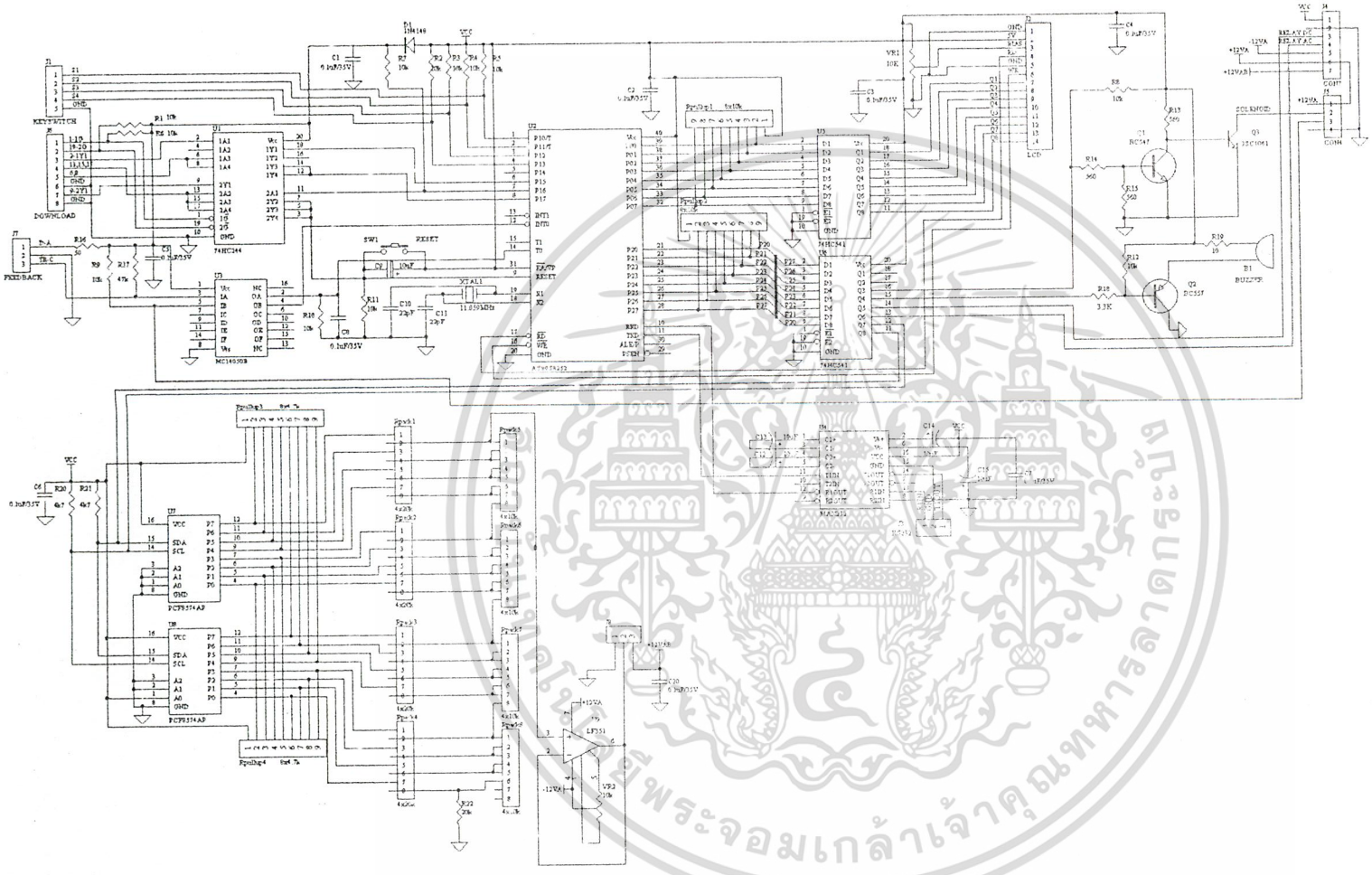
รูป ผ.1 แสดงภายในตัวเครื่องปั่นฯ

รูป ผ.2 แสดงภายนอกตัวเครื่องปั่นฯ

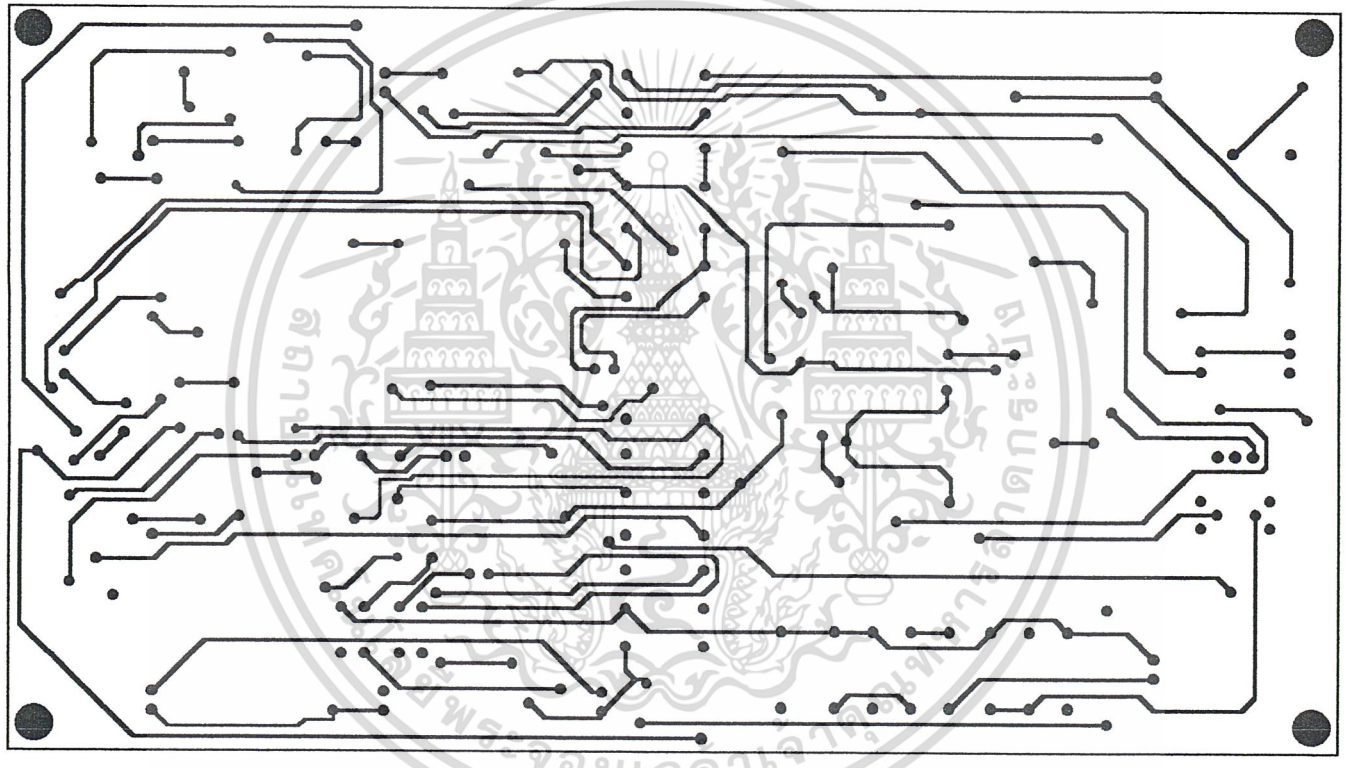
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



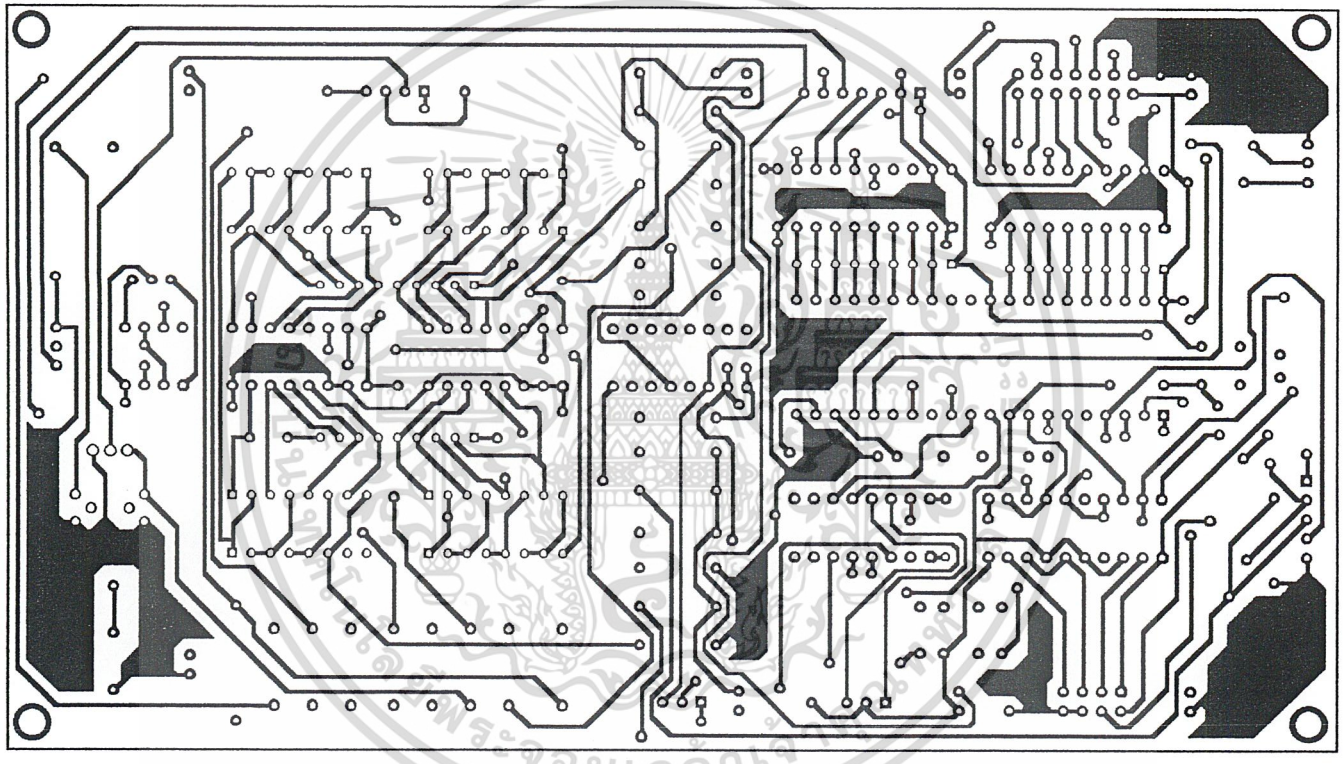
รูป ผ.3 แสดงวงจรรวมทั้งหมดของเครื่องปั่น (ยกเว้นภาคจ่ายไฟ)



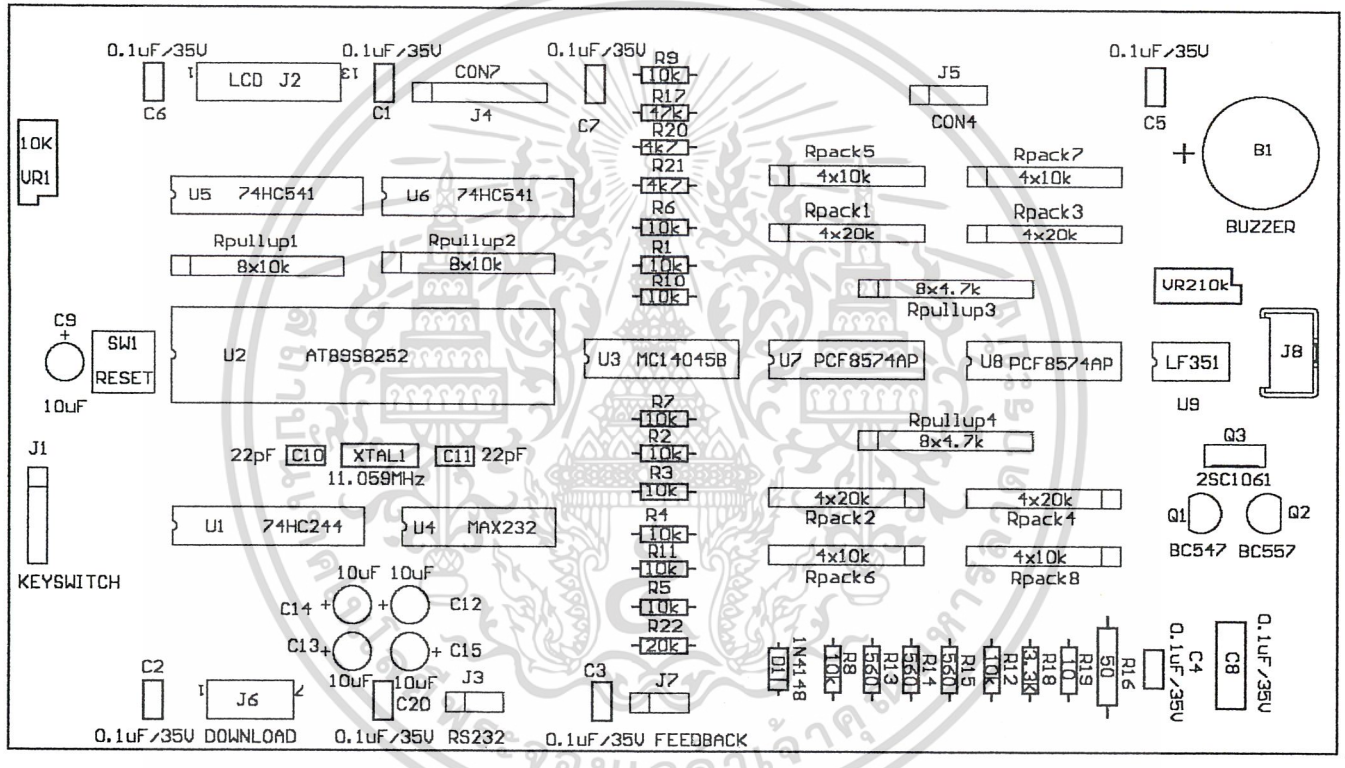
รูป ผ.4 แสดงวงจรไมโครคอนโทรลเลอร์



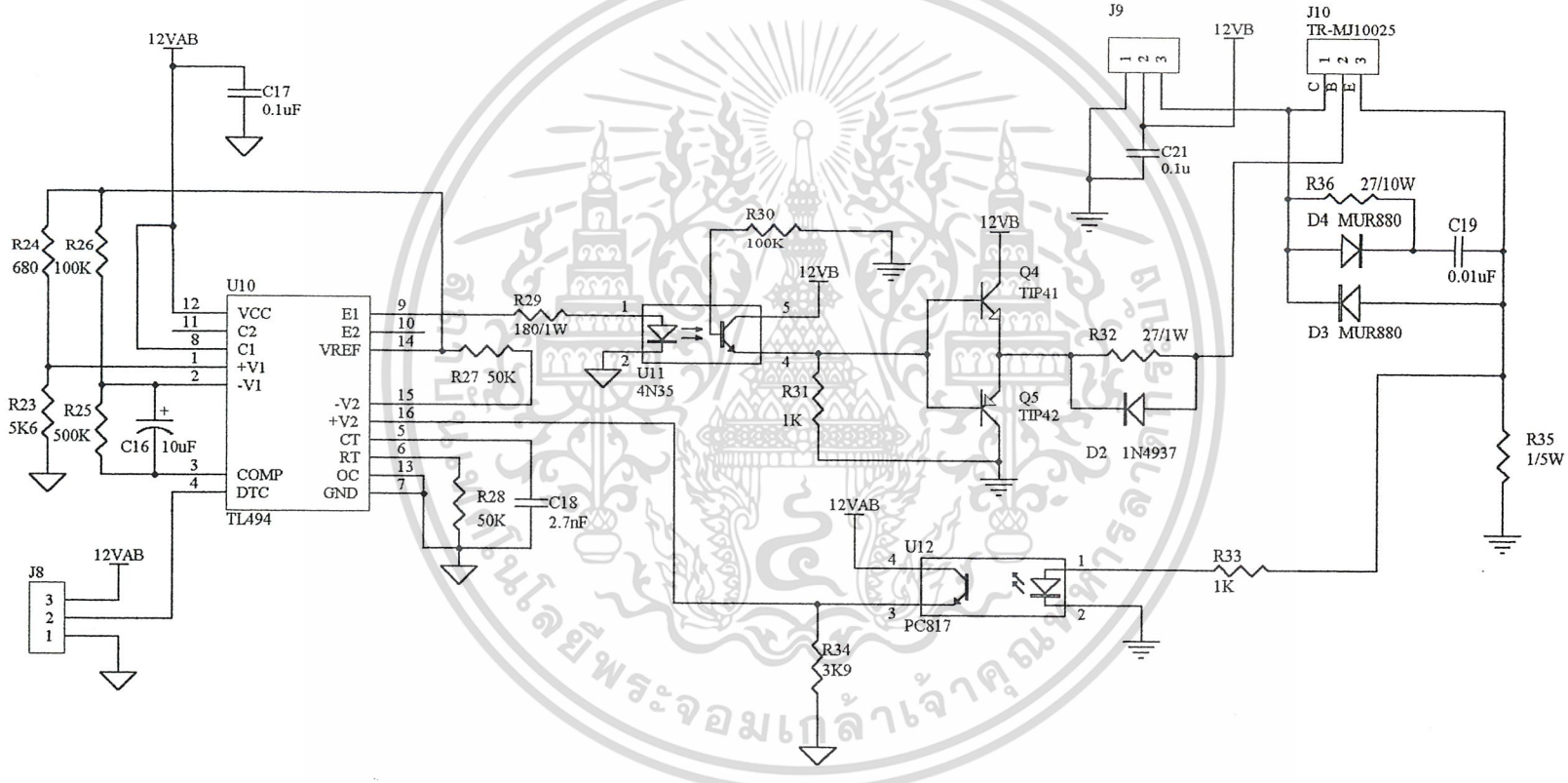
รูป ผ.5 แสดงลายวงจรด้านบนของชุดไมโครคอนโทรลเลอร์



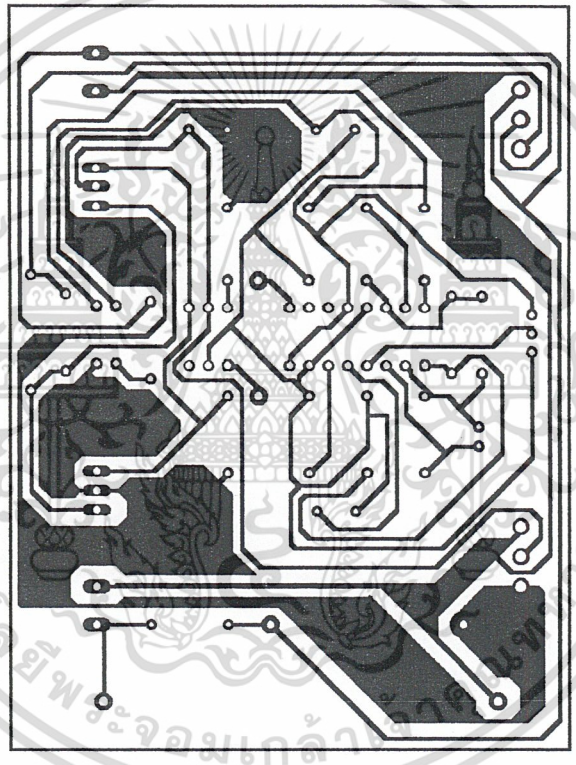
รูป ผ.6 แสดงลายวงจรด้านล่างของชุดไมโครคอนโทรลเลอร์



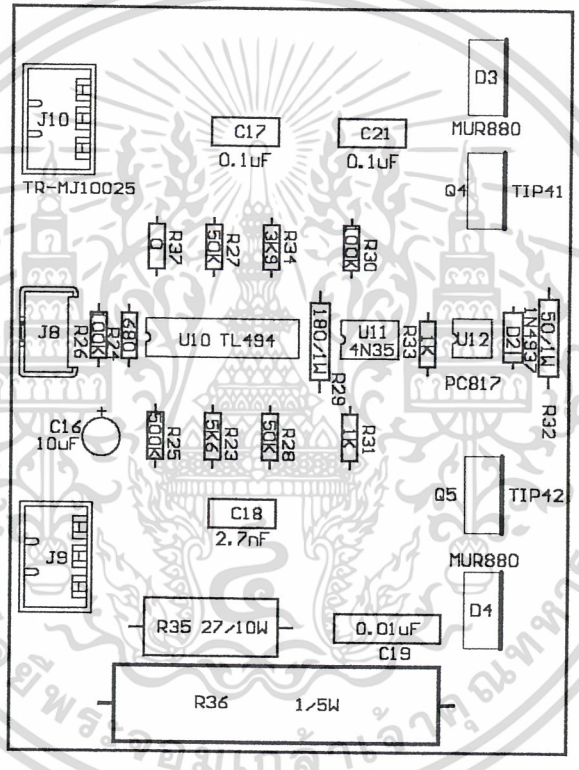
รูป ผ.7 แสดงการวางอุปกรณ์ของชุดไมโครคอนโทรลเลอร์



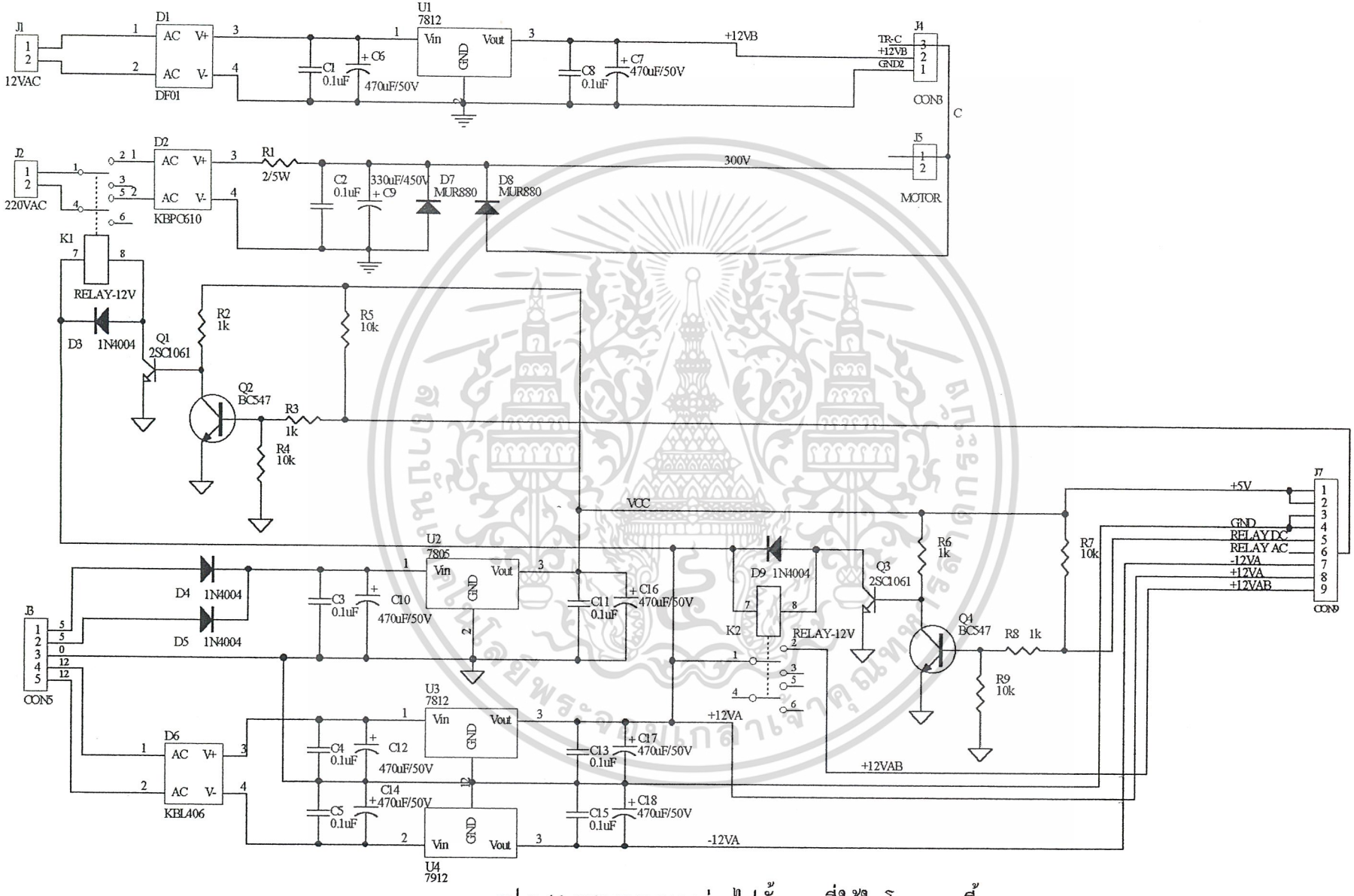
รูปผ.8 แสดงวงจรขับมอเตอร์



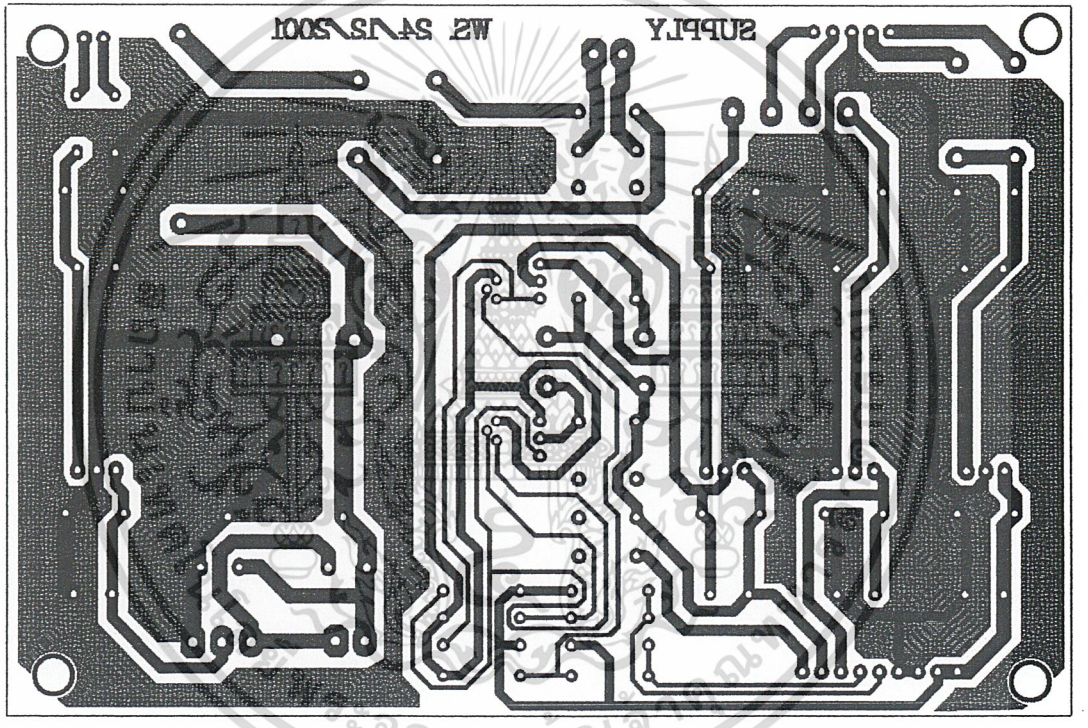
รูป ผ.9 แสดงลายวงจรด้านล่างของชุดขับมอเตอร์



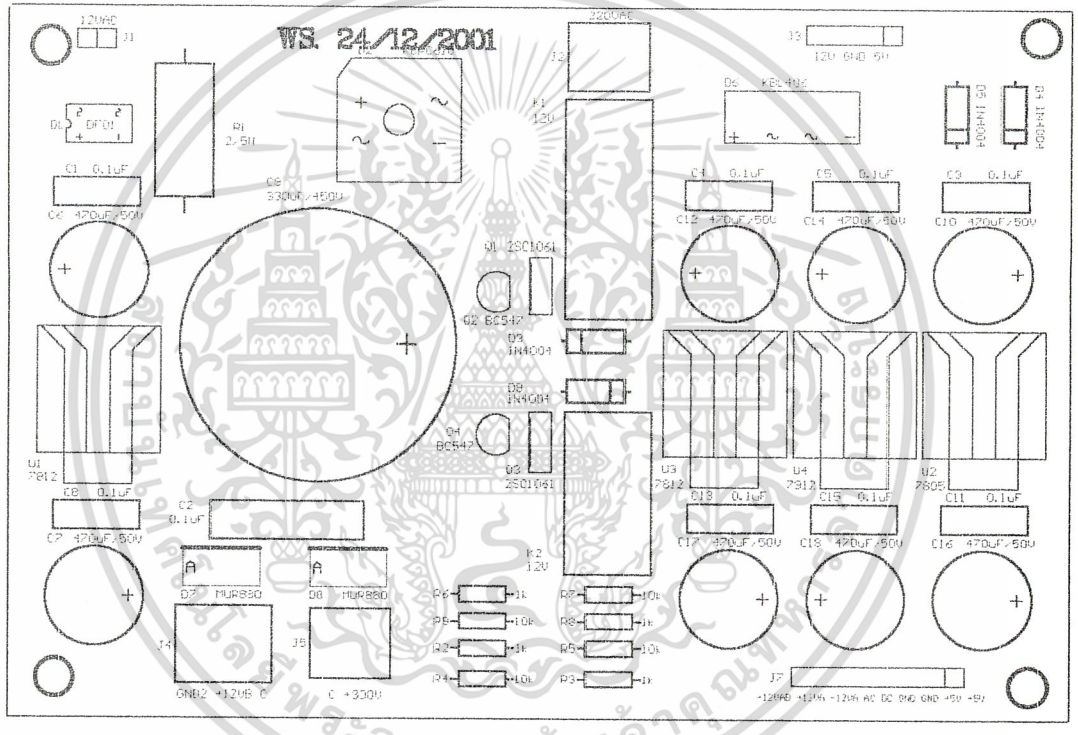
รูป ผ.10 แสดงการวางอุปกรณ์ของชุดขับมอเตอร์



รูป ผ.11 แสดงวงจรภาคจ่ายไฟทั้งหมดที่ใช้ในโครงงานนี้



รูป ผ.12 แสดงลายวงจรของภาคจ่ายไฟ



รูป ผ.13 แสดงการวางอุปกรณ์ของภาคจ่ายไฟ