

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ภาควิชาวิศวกรรมศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญาโท

ชื่อหัวข้อ เครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส

VARIABLE SPEED DRIVE FOR 3-PHASE INDUCTION MOTOR

ชื่อนักศึกษา	1. นายปฏิภาณ	ชนะชัย	รหัสประจำตัว	41031412
	2. นายภัทรฤทธิ์	ผาตินาวิน	รหัสประจำตัว	41031415
	3. นายภูสิต	เพชรพุทธากร	รหัสประจำตัว	41031416
	4. นายเอกปริญญา	หางแก้ว	รหัสประจำตัว	41031430

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชา อิเล็กทรอนิกส์และคอมพิวเตอร์
อาจารย์ที่ปรึกษา อาจารย์วรวิทย์ สมหา

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. อาจารย์วรวิทย์ สมหา	
2. อาจารย์กิติพงศ์ มะโน	
3. อาจารย์อำพล ทองระอา	
4. อาจารย์ไพบุลย์ พวงวงศ์ตระกูล	
5. อาจารย์สุระชัย พิมพ์สาลี	

วัน/เดือน/ปีที่สอบ วันศุกร์ที่ 12 พฤษภาคม พ.ศ. 2543 เวลา 12.00 น.

สถานที่สอบ ห้อง ค.310 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม

(ผศ.วิสุทธิ์ อธิพรธรรม)

หัวหน้าภาควิชาวิศวกรรมศาสตร์วิศวกรรม

วันที่ 11 เดือน ก.ค. พ.ศ. 2543



เลขหมู่.....
เลขทะเบียน... 37193
วัน, เดือน, ปี - 5 ก.ย. 2543

ปริญญานิพนธ์

เครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส

VARIABLE SPEED DRIVE FOR 3-PHASE INDUCTION MOTOR

นายปฏิภาณ	ชนะชัย
นายภัทรฤทธิ์	ผาดินาวิน
นายภูสิต	เพชรพุทธานุสร
นายเอกปริญญ์	หางแก้ว

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์

ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

ปริญญานิพนธ์

เรื่อง เครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส

Variable Speed Drive for 3-Phase Induction Motor

วัตถุประสงค์

1. เพื่อศึกษาหลักการทำงานของระบบควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535
2. เพื่อศึกษาหลักการทำงานของไมโครคอนโทรลเลอร์เบอร์ AT90S8535
3. เพื่อศึกษาหลักการเขียน โปรแกรมเพื่อควบคุมมอเตอร์ 3 เฟส
4. เพื่อออกแบบวงจรและขั้นตอนการทำงานของวงจรควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535
5. เพื่อสร้างเครื่องควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535
6. เพื่อนำเครื่องควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535 ไปใช้งานจริงได้

ประโยชน์ที่คาดว่าจะได้รับ

1. รู้หลักการทำงานของระบบควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535
2. รู้หลักการทำงานของไมโครคอนโทรลเลอร์เบอร์ AT90S8535
3. รู้หลักการเขียน โปรแกรมเพื่อควบคุมมอเตอร์ 3 เฟส
4. รู้หลักการออกแบบวงจรและขั้นตอนการทำงานของวงจรควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535
5. สามารถสร้างเครื่องควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535
6. สามารถนำเครื่องควบคุมความเร็วของมอเตอร์ 3 เฟส โดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535 ไปใช้งานจริงในประเภทอุตสาหกรรมได้

I

ชื่อหัวข้อ	เครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส
นักศึกษา	นายปฏิภาณ ชนะชัย นายภัทรฤทธิ์ ผาตินาวิน นายภูสิต เพชรพุกชางกูร นายเอกปริญญา หางแก้ว
อาจารย์ที่ปรึกษา	อาจารย์รวิทย์ สมหา
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชา	อิเล็กทรอนิกส์และคอมพิวเตอร์
ปีการศึกษา	2542

บทคัดย่อ

ปริิญาานิพนธ์ฉบับนี้้นำเสนอเครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟสโดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535 ซึ่งมีคุณสมบัติการทำงานที่สามารถใช้กับมอเตอร์เหนี่ยวนำกระแสสลับขนาด 1/4 แรงม้า ระบบนี้จะใช้หลักการมอดูเลตสัญญาณตามความกว้างของพัลส์ (Pulse width Modulation) ที่สร้างจากไมโครคอนโทรลเลอร์เบอร์ AT90S8535 เพื่อใช้ควบคุมความเร็วของมอเตอร์ 3 เฟส สามารถกำหนดความเร็ว และกำหนดทิศทางการหมุนของมอเตอร์ 3 เฟสได้ ซึ่งสามารถนำไปประยุกต์ใช้งานอุตสาหกรรมต่างๆได้

II

Thesis Title	Variable Speed Drive for 3-Phase Induction Motor
Students	Mr.Pathipan Chanachai Mr.Pattararit Phatinawin Mr.Pusit Petputthangkul Mr.Akeparin Hangkeaw
Advisor	Mr.Worawit Somha
Education Level	Bachelor of Science in Industrial Education
Program in	Electronics and Computer
Academic Year	1999

ABSTRACT

This thesis presents a project of variable speed drive for three phase induction motor by using Microcontroller AT90S8535 which the quality of Microcontroller AT90S8535 can be used with a 1/4 horse power induction motor. This system uses the pulse width modulation signals which is generated from Microcontroller AT90S8535 to control the speed of three phase motor, to set the speed and the direction of movement of three phase motor. The principle of this project can be applied for other industries.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงได้ด้วยดีนั้น ทางกลุ่มผู้จัดทำ ขอขอบคุณอาจารย์ที่ปรึกษาคืออาจารย์วรวิทย์ สมหา ซึ่งให้ความช่วยเหลือในทุกๆ ด้าน

ขอขอบคุณเพื่อนร่วมงานในกลุ่มทุกคนที่ร่วมแรงร่วมใจในการทำงานจนบรรลุเป้าหมาย และจากความสำเร็จในครั้งนี้ขอมอบให้กับบุคคลที่ได้ให้การศึกษาตั้งแต่อดีตจนปัจจุบัน คือ คุณพ่อ คุณแม่ ครู อาจารย์ และผู้มีพระคุณทุกท่าน

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	X
บทที่ 1 บทนำ	1
1.1 แนวความคิดในการทำปริญญานิพนธ์	1
1.2 จุดความสามารถของโครงการ	1
1.3 เนื้อหาโดยสังเขปของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 มอเตอร์ไฟฟ้ากระแสสลับ	3
2.2 มอเตอร์เหนี่ยวนำ	4
2.2.1 เมื่อให้ส่วนเคลื่อนที่มีลักษณะเป็นจานตัวนำกลมบาง	4
2.2.2 เมื่อให้ส่วนเคลื่อนที่เป็นทรงกระบอก	6
2.3 มอเตอร์เหนี่ยวนำสามเฟส	8
2.3.1 มอเตอร์เหนี่ยวนำสามเฟสแบบสไลควเรลเกจโรเตอร์	8
2.3.2 มอเตอร์เหนี่ยวนำสามเฟสแบบวาวด์โรเตอร์	12
2.4 วิธีสตาร์ทมอเตอร์เหนี่ยวนำ	15
2.4.1 วิธีสตาร์ทมอเตอร์เหนี่ยวนำแบบสไลควเรลเกจโรเตอร์	15
2.4.2 วิธีสตาร์ทมอเตอร์เหนี่ยวนำแบบวาวด์โรเตอร์	18
2.5 วิธีควบคุมความเร็วของมอเตอร์เหนี่ยวนำ	19
2.5.1 การควบคุมความเร็วทางด้านสเตเตอร์	19
2.5.2 การควบคุมความเร็วทางด้านโรเตอร์	20
2.6 วิธีการระบุตำแหน่งการหมุนให้กับมอเตอร์เหนี่ยวนำ	22

สารบัญ (ต่อ)

เรื่อง	หน้า
2.7 การมอดูเลตสัญญาณตามความกว้างของพัลส์	28
2.7.1 หลักการมอดูเลตสัญญาณตามความกว้างของพัลส์	28
2.7.2 เทคนิคการมอดูเลตสัญญาณตามความกว้างของพัลส์ แบบเทียบพื้นที่ใต้เส้นโค้งไซน์	29
2.7.3 ข้อกำหนดทางเทคนิคการมอดูเลตสัญญาณตามความกว้างของพัลส์ แบบเทียบพื้นที่ใต้เส้นโค้งไซน์	30
2.8 การเปลี่ยนจำนวนพัลส์ PWM ที่สัมพันธ์กับความถี่	31
2.8.1 เทคนิค Minimum Ripple Current	35
2.8.2 วิธีการ Adaptive Current Control PWM	36
2.9 ไมโครคอนโทรลเลอร์ AT90S8535	38
2.9.1 โครงสร้างภายใน	38
2.9.2 รายละเอียดของขาสัญญาณ	40
2.9.3 การใช้งาน CRYSTAL OSILATOR	40
2.9.4 สถาปัตยกรรมภายในและรีจิสเตอร์ใช้งานทั่วไป	41
2.9.5 การใช้งานในโหมด PWM	50
2.9.6 การแปลงสัญญาณอนาลอกเป็นดิจิตอลใน AT90S8535	52
2.9.7 พอร์ตอินพุต และพอร์ตเอาต์พุต	53
2.9.8 กลุ่มคำสั่ง	55
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	68
3.1 แผนผังการทำงานของโครงการ	68
3.2 วงจรภาคขับเคลื่อนมอเตอร์และสวิตซ์กำลัง	69
3.3 วงจรควบคุมและสร้างสัญญาณ PWM	71
3.4 วงจรป้องกัน	72
3.5 วงจรแหล่งจ่ายไฟ	72
3.5.1 แหล่งจ่ายไฟฟ้ากระแสตรง 5 โวลต์	73

สารบัญ (ต่อ)

เรื่อง	หน้า
3.5.2 แหล่งจ่ายไฟฟ้ากระแสตรง 12 โวลต์	73
3.5.3 แหล่งจ่ายไฟฟ้ากระแสตรง 310 โวลต์	73
3.6 หลักการเขียน โปรแกรม	74
3.7 การออกแบบโครงสร้างของโครงการ	76
บทที่ 4 การทดลอง และผลการทดลอง	77
4.1 การทดลองแหล่งจ่ายไฟฟ้ากระแสตรง	77
4.2 การทดลองวงจร PWM	77
4.3 การทดลอง Deadtime	79
4.4 การทดลองตำแหน่งและความเร็ว	80
บทที่ 5 บทสรุป ปัญหา แนวทางการแก้ปัญหา และการพัฒนา	81
5.1 บทสรุป	81
5.2 ปัญหาและแนวทางแก้ไข	81
5.3 แนวทางการพัฒนา	83
ภาคผนวก ก ภาพเครื่องต้นแบบในปริยญาณิพนธ์	84
ภาคผนวก ข การรวบรวมลายวงจรพิมพ์ที่ใช้ในโครงการ	88
ภาคผนวก ค การรวบรวมโปรแกรมการทำงาน	93
ภาคผนวก ง การสรุปรายการอุปกรณ์ที่ใช้ในโครงการทั้งหมด	104
ภาคผนวก จ คู่มือการใช้งานของโครงการ	121
บรรณานุกรม	125
ประวัติผู้แต่ง	126

สารบัญรูป

VII

สารบัญรูป

รูป	หน้า
รูปที่ 2.1 ส่วนประกอบเบื้องต้นของมอเตอร์ไฟสลัป	4
รูปที่ 2.2 แท่งแม่เหล็กรูปตัวยูวางคว่ำเข้าหาจานตัวนำ	5
รูปที่ 2.3 ลักษณะเส้นแรงแม่เหล็กรวมรอบตัวนำภายใต้ขั้วแม่เหล็ก N	6
รูปที่ 2.4 ตัวนำรูปทรงกระบอกเคลื่อนที่ได้เนื่องจากกระแสไฟฟ้าเหนี่ยวนำ	7
รูปที่ 2.5 การต่อขดลวดเฟส A	9
รูปที่ 2.6 การต่อขดลวดเฟส A และเฟส B	10
รูปที่ 2.7 การต่อขดลวดเฟส A เฟส B และเฟส C	10
รูปที่ 2.8 การต่อขดลวดมอเตอร์สามเฟสแบบสตาร์	10
รูปที่ 2.9 การต่อขดลวดมอเตอร์สามเฟสแบบเดลต้า	11
รูปที่ 2.10 วิธีต่อปลายสายขดลวดมอเตอร์สามเฟสเพื่อให้หมุนในทิศทางหนึ่ง	12
รูปที่ 2.11 วิธีต่อปลายสายขดลวดมอเตอร์สามเฟส เพื่อให้หมุนในอีกทิศทางหนึ่งซึ่งตรงข้ามกับรูปที่ 2.10	12
รูปที่ 2.12 การต่อมอเตอร์สามเฟสแบบวาวด์โรเตอร์	14
รูปที่ 2.13 การต่อมอเตอร์สามเฟสแบบวาวด์โรเตอร์ให้หมุนกลับทิศทาง	14
รูปที่ 2.14 วงจรสตาร์ทโดยตรง	15
รูปที่ 2.15 วงจรสตาร์ทลดแรงดันป้อน โดยต่อความต้านทานอันดับกับมอเตอร์	16
รูปที่ 2.16 วงจรสตาร์ทแบบต่อออโต-ทรานส์ฟอร์มเมอร์อันดับกับมอเตอร์	17
รูปที่ 2.17 วงจรสตาร์ทแบบสตาร์-เดลต้า	18
รูปที่ 2.18 วงจรสตาร์ทสลีปริงมอเตอร์แบบต่อตรง	19
รูปที่ 2.19 วิธีควบคุมความเร็วโดยใช้มอเตอร์ 2 ตัวทำงานร่วมกัน	21
รูปที่ 2.20 เทคนิคในการเริ่มต้นหมุนมอเตอร์เหนี่ยวนำที่มีสลีปริง	25
รูปที่ 2.21 การต่อขดลวดแบบสตาร์-เดลต้า	26
รูปที่ 2.22 หลักการสร้างสัญญาณ PWM ตามวิธีฮาร์โมนิกย่อย	29
รูปที่ 2.23 สัญญาณ PWM ตามหลักการเทียบพื้นที่ใต้โค้งไซน์	31
รูปที่ 2.24 ความสัมพันธ์ระหว่างความถี่มูลฐานและความถี่สวิทช์ของอุปกรณ์สวิทช์ซึ่งเทคนิคการกำจัดฮาร์โมนิกส์	32

สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 2.25 รูป PWM โดยวิธีกำจัดฮาร์โมนิกส์	33
รูปที่ 2.26 หลักการเกิดรูปคลื่น PWM โดยเทคนิค Adaptive Current Control	36
รูปที่ 2.27 วงจรควบคุมเทคนิค Adaptive Current Control	37
รูปที่ 2.28 โครงสร้างภายนอกและตำแหน่งขา	38
รูปที่ 2.29 แผนผังการทำงาน	39
รูปที่ 2.30 การใช้งาน OSC ภายใน MCU	41
รูปที่ 2.31 การใช้งาน OSC ภายนอก MCU	41
รูปที่ 2.32 สถาปัตยกรรมแบบ RISC ของ AT90S8535	42
รูปที่ 2.33 โครงสร้างของหน่วยความจำ	43
รูปที่ 2.34 โครงสร้างของรีจิสเตอร์ใช้งานทั่วไป	43
รูปที่ 2.35 การจัดการหน่วยความจำ SRAM การเข้าถึงข้อมูล	44
รูปที่ 2.36 Register Direct, Single Register Rd	45
รูปที่ 2.37 Register Direct, Two Register Rd and Rr	45
รูปที่ 2.38 การเข้าถึงข้อมูลแบบ Direct Register Addressing Rd and Rr	46
รูปที่ 2.39 การเข้าถึงข้อมูลแบบ Direct Register Addressing	46
รูปที่ 2.40 การเข้าถึงข้อมูลแบบ Data Indirect with Displacement	47
รูปที่ 2.41 การเข้าถึงข้อมูลแบบ Data Indirect Addressing	47
รูปที่ 2.42 การเข้าถึงข้อมูลแบบ Data Indirect with Pre-Displacement	48
รูปที่ 2.43 การเข้าถึงข้อมูลแบบ Data Indirect with Post-Displacement	48
รูปที่ 2.44 การเข้าถึงข้อมูลแบบ Constant Addressing Using the LPM Instruction	49
รูปที่ 2.45 การเข้าถึงข้อมูลแบบ Code Memory Constant Addressing	49
รูปที่ 2.46 การเข้าถึงข้อมูลแบบ Relative Program Addressing	49
รูปที่ 2.47 การ Fetch และ Executions แบบขนาน	50
รูปที่ 2.48 ผลของ Unsynchronized	51
รูปที่ 3.1 แผนผังการทำงานของโครงงาน	68
รูปที่ 3.2 วงจรสวิตชิงแบบฮาล์ฟบริดจ์	69

สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 3.3 วงจรชุดขับมอเตอร์และสวิตซ์กำลัง	70
รูปที่ 3.4 วงจรสร้างสัญญาณ PWM	71
รูปที่ 3.5 วงจรป้องกัน	72
รูปที่ 3.6 วงจรแหล่งจ่ายไฟกระแสตรง 5 โวลต์	73
รูปที่ 3.7 วงจรแหล่งจ่ายไฟกระแสตรง 12 โวลต์	73
รูปที่ 3.8 วงจรแหล่งจ่ายไฟกระแสตรง 310 โวลต์	74
รูปที่ 3.9 ผังการทำงานของโปรแกรม	75
รูปที่ 3.10 โครงสร้างตัวเครื่อง	76
รูปที่ 4.1 ความถี่สวิตซ์ที่มีประมาณ 48 พัลส์ ใน 1 ความถี่หลักมูล	78
รูปที่ 4.2 ลักษณะการตรงกันข้ามของแรงดันที่ไบอัสขาเกตของมอสเฟต ในเฟส 1/1 และ 1/2 ในกิ่งเดียวกัน	78
รูปที่ 4.3 ค่า Deadtime สูงสุด 5 ไมโครวินาทีที่กำหนดจากไมโครคอนโทรลเลอร์	79

สารบัญตาราง

รูป	หน้า
ตารางที่ 1 การกำหนดค่าสูงสุดของ PWM	50
ตารางที่ 2 การกำหนดสัญญาณที่ขา PD5(OC1A) และ PD6(OC1B) ในโหมด PWM	50
ตารางที่ 3 เอาต์พุตของ PWM (OCR1X หรือ TOP)	52
ตารางที่ 4 การใช้งานฟังก์ชันอื่นๆ ของพอร์ต B	54
ตารางที่ 5 ฟังก์ชันอื่นๆ ของพอร์ต D	55

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปริยญาณิพนธ์

ในปัจจุบันนี้ประเทศไทยเป็นประเทศที่กำลังพัฒนาจากประเทศเกษตรกรรมไปเป็นประเทศอุตสาหกรรม การศึกษาในด้านอุตสาหกรรมจึงมีความสำคัญเป็นอย่างยิ่ง และเทคโนโลยีทางด้านงานอุตสาหกรรมได้มีการพัฒนาไปอย่างรวดเร็ว โดยเฉพาะงานควบคุมที่ใช้มอเตอร์ ซึ่งในโรงงานอุตสาหกรรมส่วนมากในปัจจุบันมักจะใช้มอเตอร์เหนี่ยวนำไฟฟ้ากระแสสลับชนิด 3 เฟส เนื่องจากสามารถควบคุมความเร็วและทิศทางการหมุนของมอเตอร์ได้แน่นอนมีความแม่นยำสูง และในการควบคุมมอเตอร์เหนี่ยวนำชนิด 3 เฟสนี้ จำเป็นจะต้องมีระบบควบคุมที่มีประสิทธิภาพสูง และระบบควบคุมมอเตอร์เหนี่ยวนำ 3 เฟส ที่ใช้อยู่ในปัจจุบัน เป็นระบบควบคุมที่ต้องสั่งซื้อจากต่างประเทศซึ่งมีราคาสูง เป็นผลทำให้มีการลงทุนสูงในงานอุตสาหกรรมนั้นๆ

ปริยญาณิพนธ์ฉบับนี้จึงได้นำเสนอ เครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส ซึ่งมีคุณสมบัติที่ใช้ควบคุมความเร็วและทิศทางการหมุนของมอเตอร์ได้อย่างมีประสิทธิภาพ เป็นที่น่าพอใจ และคาดว่าจะจะเป็นประโยชน์ต่อการศึกษาและประยุกต์ใช้งานต่อไป

1.2 ขีดความสามารถของโครงการ

เครื่องควบคุมความเร็วมอเตอร์เหนี่ยวนำ 3 เฟส ได้ออกแบบโดยใช้ไมโครคอนโทรลเลอร์เบอร์ AT90S8535 โดยมีคุณสมบัติที่เหมาะสมสำหรับสร้างสัญญาณเพื่อควบคุมการทำงานของมอเตอร์เหนี่ยวนำ 3 เฟส ซึ่งเครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส มีขีดความสามารถดังนี้

1. สามารถควบคุมความเร็วและทิศทางการหมุนของมอเตอร์เหนี่ยวนำ 3 เฟสได้
2. ใช้กับมอเตอร์ไฟฟ้าเหนี่ยวนำ 3 เฟส ขนาดไม่เกิน 1/4 แรงม้า
3. สามารถนำไปประยุกต์ใช้งานด้านอุตสาหกรรม เช่น ควบคุมมอเตอร์เหนี่ยวนำ 3 เฟส โดยต่อร่วมกับเครื่องวัดระยะกันชนหลัง (Backgauge) ในเครื่องตัดเหล็กหรือพับเหล็กได้

1.3 เนื้อหาโดยสังเขปของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ประกอบด้วยเนื้อหาส่วนต่างๆ ดังนี้

บทที่ 2 ทฤษฎีและหลักการ จะกล่าวถึงหลักการทำงานของมอเตอร์ไฟฟ้ากระแสสลับที่เป็นมอเตอร์เหนี่ยวนำแบบ 3 เฟส รวมทั้งวิธีการควบคุมความเร็ว การกำหนดทิศทางการหมุนของมอเตอร์เหนี่ยวนำแบบ 3 เฟส และการใช้หลักการมอดูเลตสัญญาณตามความกว้างของพัลส์ หรือ Pulse Width Modulation (PWM) ตลอดจนคุณสมบัติและการใช้งานไมโครคอนโทรลเลอร์บอร์ด AT90S8535

บทที่ 3 การออกแบบ การสร้าง และการทำงาน จะกล่าวถึง แผนผังการทำงานของโครงการ, การออกแบบวงจร, หน้าที่ของวงจรต่างๆ และการใช้งานอุปกรณ์ต่างๆ ที่นำมาใช้ในโครงการเครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส

บทที่ 4 การทดลอง และผลการทดลอง จะกล่าวถึงผลการทดลองในการควบคุมความเร็วมอเตอร์ และการควบคุมทิศทางการหมุนของมอเตอร์ ตลอดจนรูปคลื่นสัญญาณตามจุดทดสอบต่างๆ ของแต่ละวงจรในโครงการ

บทที่ 5 บทสรุป ปัญหา แนวทางแก้ปัญหา และการพัฒนา จะกล่าวถึงข้อบกพร่องต่างๆ ของระบบที่ทำในโครงการ, การอธิบายสาเหตุของข้อบกพร่อง, แนวทางการปรับปรุงแก้ไขให้เป็นระบบที่ดีขึ้น และแนวทางในการที่จะสามารถพัฒนาโครงการให้มีประสิทธิภาพมากกว่าระบบเดิมที่สร้างขึ้นมา

ภาคผนวก ก เครื่องต้นแบบ

ภาคผนวก ข วงจรและแผ่นวงจรพิมพ์

ภาคผนวก ค แผนผังการทำงานและโปรแกรม

ภาคผนวก ง รายละเอียดและคุณสมบัติของอุปกรณ์

ภาคผนวก จ คู่มือการใช้งาน

บทที่ 2

ทฤษฎี และหลักการ

ในการใช้งานมอเตอร์เหนี่ยวนำ 3 เฟส จะต้องทราบถึงหลักการทำงานของมอเตอร์ไฟฟ้ากระแสสลับ ซึ่งมีทั้งมอเตอร์เหนี่ยวนำและมอเตอร์แบบซิงโครนัส โดยการใช้งานจะแตกต่างกันไปตามรูปแบบการใช้งาน ส่วนที่สำคัญอีกส่วนหนึ่งก็คือส่วนของการควบคุมการทำงานของมอเตอร์ โดยมอเตอร์เหนี่ยวนำ 3 เฟสจะใช้หลักการของการมอดูเลตสัญญาณตามความกว้างของพัลส์ (Pulse Width Modulation) ส่วนสุดท้ายคือคุณสมบัติและหลักการทำงานของไมโครคอนโทรลเลอร์ AT90S8535 ซึ่งการประยุกต์ใช้งานให้สามารถทำงานได้อย่างมีประสิทธิภาพจะต้องใช้ไมโครคอนโทรลเลอร์ในการควบคุม

2.1 มอเตอร์ไฟฟ้ากระแสสลับ

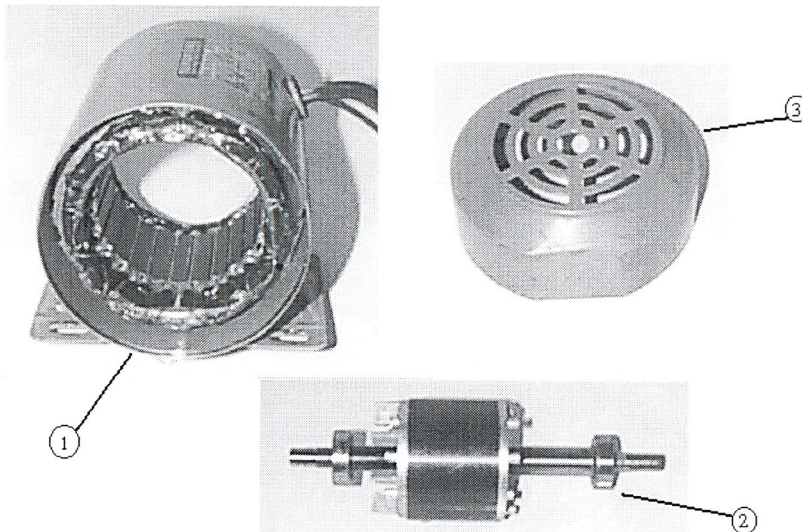
มอเตอร์ไฟฟ้ากระแสสลับมีทั้งมอเตอร์เหนี่ยวนำและมอเตอร์แบบซิงโครนัสซึ่งมีการใช้งานอย่างแพร่หลาย ตั้งแต่โรงงานอุตสาหกรรมขนาดเล็กไปจนถึงโรงงานอุตสาหกรรม ขนาดใหญ่ หรือแม้แต่บ้านพักอาศัย ซึ่งอาจจะอยู่ในลักษณะของเครื่องอำนวยความสะดวกทั้งที่เป็นสิ่งจำเป็นและฟุ่มเฟือยในรูปแบบต่างๆ กันมากมายตั้งแต่ขนาดเล็กๆ ไปจนถึงขนาดใหญ่ๆ มอเตอร์ไฟฟ้ากระแสสลับเป็นมอเตอร์ที่ทำงานด้วยความเร็วคงที่เป็นส่วนใหญ่ยกเว้นเพียงบางชนิดที่อาจจะออกแบบให้สามารถปรับความเร็วได้แต่ก็เป็นส่วนน้อย

มอเตอร์ไฟฟ้ากระแสสลับคือเครื่องกลไฟฟ้าที่สามารถเปลี่ยนพลังงานไฟฟ้าให้เป็นพลังงานกลในการเปลี่ยนพลังงานไฟฟ้าให้เป็นพลังงานกลนี้พลังงานไฟฟ้าไม่ได้นำเข้าสู่ส่วนโรเตอร์ของมอเตอร์โดยตรงแต่ได้จากการเหนี่ยวนำหรือที่เรียกว่าอินดักชัน (Induction) ดังนั้นจึงเรียกมอเตอร์ไฟฟ้ากระแสสลับว่ามอเตอร์เหนี่ยวนำ มอเตอร์ไฟฟ้ากระแสสลับนี้บางทีเรียกชื่อสั้นๆ ว่ามอเตอร์ไฟสลับ หรือมอเตอร์กระแสสลับ แบ่งออกได้เป็นสองแบบด้วยกัน คือ แบบที่มีโรเตอร์เป็นสquirrel cage (Squirrel Cage) หรือทรงกระบอก เรียกมอเตอร์แบบนี้ว่า squirrel cage motor (Squirrel Cage Motor) และแบบที่โรเตอร์พันด้วยเส้นลวดเล็กๆ ที่เรียกว่า วาวด์โรเตอร์ (Wound Rotor) เรียกมอเตอร์แบบนี้ว่า วาวด์โรเตอร์มอเตอร์ (Wound Rotor Motor) หรือเรียกว่า สลิปริงมอเตอร์ (Slip Ring Motor)

มอเตอร์ทั้งสองแบบนี้มีส่วนประกอบคล้ายๆ กันดังนี้คือส่วนอยู่กับที่ (Stator) เหมือนกัน ฝาครอบ (End Place) เหมือนกันจะแตกต่างกันก็เพียงเฉพาะส่วนเคลื่อนที่ (Rotor) เท่านั้น

และมอเตอร์ไฟสลัหรืออินคักซ์นอเตอร์นี้ ยังแบ่งออกได้เป็น 2 ชนิดด้วยกันคือ ชนิดหนึ่งเฟส (Single -Phase) และชนิด 3 เฟส (Three-Phase) แต่โดยทั่วไปในงาน อุตสาหกรรมขนาดใหญ่จะใช้ มอเตอร์ 3 เฟสเป็นจำนวนมากและเป็นที่ยอมรับใช้โดยเฉพาะงานด้านควบคุม ส่วนประกอบเบื้องต้นของมอเตอร์ไฟสลัทั้งชนิดหนึ่งเฟสและ 3 เฟสจะเหมือนกัน โดยประกอบไปด้วยสามส่วนต่อไปนี้ดังรูปที่ 2.1

1. ส่วนอยู่กับที่ หรือ สเตเตอร์
2. ส่วนเคลื่อนที่ หรือ โรเตอร์
3. ฝาครอบ



รูปที่ 2.1 ส่วนประกอบเบื้องต้นของมอเตอร์ไฟสลั

2.2 มอเตอร์เหนี่ยวนำ

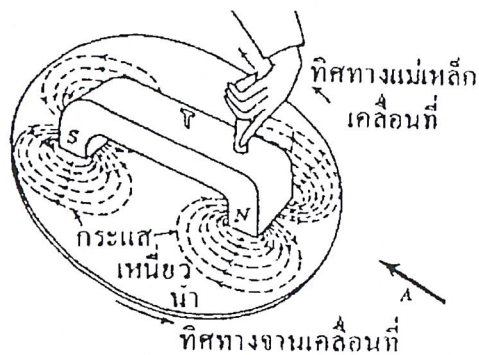
หลักการเหนี่ยวนำแม่เหล็กไฟฟ้า (Electromagnetic Induction) เบื้องต้นในมอเตอร์เหนี่ยวนำ ซึ่งเป็นมอเตอร์ไฟฟ้กระแสสลั มีหลักการดังนี้

2.2.1 เมื่อให้ส่วนเคลื่อนที่มีลักษณะเป็นจานตัวนำกลมบาง

ทำได้โดยนำเอาแท่งแม่เหล็กรูปตัวยู (U) วางคว่ำให้ขั้วแม่เหล็ก N และ S หันเข้าหาจานตัวนำที่ทำด้วยแผ่นเหล็ก, ทองแดงหรืออลูมิเนียม ดังรูปที่ 2.2 โดยให้จานตัวนำและแท่งแม่เหล็กมีศูนย์กลางร่วมกัน เส้นแรงแม่เหล็กจากขั้ว N จะพุ่งเข้าหาจานตัวนำและกลับเข้าหา ขั้ว S เมื่อแท่งแม่เหล็กและจานตัวนำอยู่นิ่งกับที่จะไม่เกิดการเหนี่ยวนำแม่เหล็กไฟฟ้าที่ส่วนจานตัวนำ

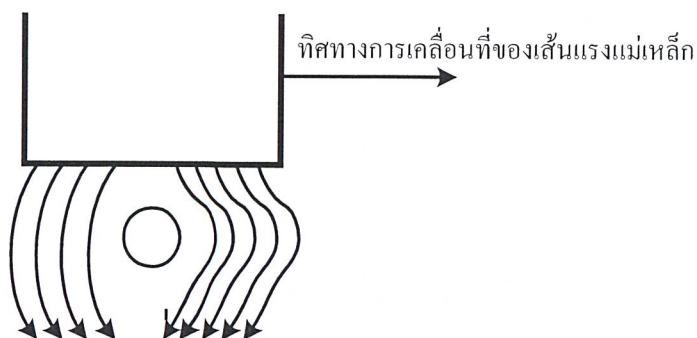
ถึงแม้ว่าจะมีเส้นแรงแม่เหล็กพุ่งผ่านจานตัวนำก็ตามเพราะเส้นแรงแม่เหล็กและจานตัวนำไม่มีการเคลื่อนที่

ถ้าให้จานตัวนำอยู่กับที่แล้วให้แท่งแม่เหล็กเคลื่อนที่ทวนเข็มนาฬิกาในกรณีนี้ก็เหมือนกับว่าให้แท่งแม่เหล็กอยู่กับที่แล้วให้จานตัวนำเคลื่อนที่ตามเข็มนาฬิกาจะเกิดการเหนี่ยวนำแม่เหล็กไฟฟ้าขึ้นที่จานตัวนำจะเกิดทั้งแรงเคลื่อนไฟฟ้าเหนี่ยวนำและกระแสไฟฟ้าเหนี่ยวนำทิศทางการเคลื่อนที่ของกระแสไฟฟ้าเหนี่ยวนำหาได้โดยใช้กฎมือขวา (Right Hand Rule) ของเฟลมมิ่งจะได้ว่าทิศทางการไหลของกระแสไฟฟ้าเหนี่ยวนำภายใต้ขั้วแม่เหล็ก N จะไหลออกจากใต้ขั้วแม่เหล็ก N ไปสู่ขอบจานตัวนำเมื่อมองจากจุด A เข้าหาจานตัวนำ และก็จะพบว่าทิศทางการไหลของกระแสไฟฟ้าเหนี่ยวนำภายใต้ขั้วแม่เหล็ก S จะไหลออกจากขอบจานตัวนำให้เข้าสู่ภายใต้ขั้วแม่เหล็ก S หากทิศทางการไหลของกระแสเหนี่ยวนำได้จากกฎมือขวาของเฟลมมิ่ง



รูปที่ 2.2 แท่งแม่เหล็กรูปตัวยูวางคว่ำเข้าหาจานตัวนำ

จากทิศทางการไหลของกระแสเหนี่ยวนำภายใต้ขั้วแม่เหล็กแต่ละขั้วจะพิจารณาต่อไปได้อีกว่าที่ขั้วแม่เหล็ก N เส้นแรงแม่เหล็กที่เกิดขึ้นจากกระแสเหนี่ยวนำด้านซ้ายมือจะเสริม กับเส้นแรงแม่เหล็กที่พุ่งออกมาจากขั้วแม่เหล็ก N และเส้นแรงแม่เหล็กที่เกิดจากกระแสการเหนี่ยวนำด้านขวามือจะหักล้างกับเส้นแรงแม่เหล็กที่พุ่งออกจากขั้วแม่เหล็ก N ที่ขั้วแม่เหล็ก S เส้นแรงแม่เหล็กที่เกิดจากกระแสเหนี่ยวนำด้านซ้ายมือจะหักล้างกับเส้นแรงแม่เหล็กที่พุ่งเข้าหาขั้วแม่เหล็ก S และเส้นแรงแม่เหล็กที่เกิดจากกระแสเหนี่ยวนำด้านขวามือจะเสริมกับเส้นแรงแม่เหล็กที่พุ่งเข้าหาขั้วแม่เหล็ก S ทิศทางการเคลื่อนที่ของเส้นแรงแม่เหล็กที่เกิดจากกระแสเหนี่ยวนำนี้ หาได้จากกฎมือขวา (Right Hand Rule) ดังนั้นลักษณะของเส้นแรงแม่เหล็กรวม รอบตัวนำภายใต้ขั้วแม่เหล็ก N จะมีลักษณะดังรูปที่ 2.3

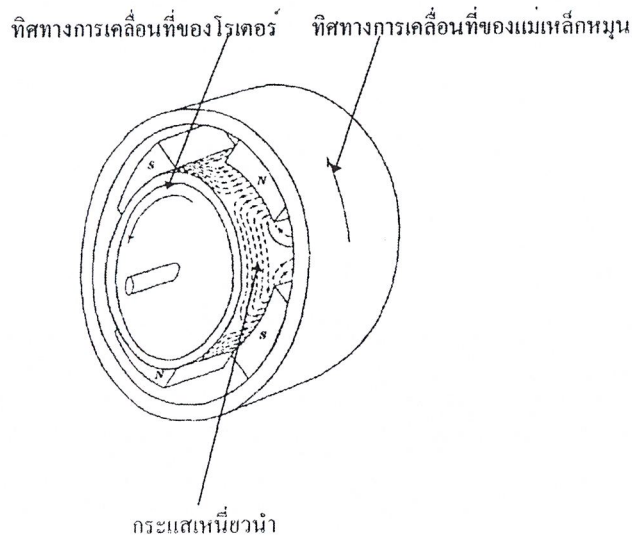


รูปที่ 2.3 ลักษณะเส้นแรงแม่เหล็กที่รวมรอบตัวนำภายใต้ขั้วแม่เหล็ก N

จากรูปที่ 2.3 ซึ่งเป็นเส้นแรงแม่เหล็กที่รวมรอบตัวนำที่งานตัวนำภายใต้ขั้วแม่เหล็ก N นี้ เกิดจากเส้นแรงแม่เหล็กที่เกิดจากกระแสเหนี่ยวนำและเส้นแรงแม่เหล็กจากขั้วแม่เหล็ก N เมื่อเป็นเช่นนี้จะทำให้เกิดแรงขึ้นที่ตัวนำภายใต้ขั้วแม่เหล็ก N และมีทิศทางการเคลื่อนที่ไปทางขวามือ ดังรูปที่ 2.3 หรืองานตัวนำจะมีทิศทางการเคลื่อนที่ทวนเข็มนาฬิกา นั่นก็คือส่วน งานตัวนำจะเคลื่อนที่ไปทางเดียวกับทิศทางการเคลื่อนที่ของแท่งแม่เหล็กที่กำหนดไว้ในครั้งแรกซึ่งทิศทางการเคลื่อนที่ของงานตัวนำนี้ก็ได้โดยใช้กฎมือซ้ายของเฟลมมิ่งและงานตัวนำซึ่งเป็น ส่วนเคลื่อนที่นี้จะเคลื่อนที่หรือหมุนด้วยความเร็วที่น้อยกว่าความเร็วของแท่งแม่เหล็กเคลื่อนที่ หรือแท่งแม่เหล็กหมุนเสมอ

2.2.2 เมื่อให้ส่วนเคลื่อนที่เป็นทรงกระบอก

จากรูปที่ 2.4 ส่วนเคลื่อนที่ที่เกนตัวนำรูปทรงกระบอกจะอยู่ภายในโครงเหล็กทรงกระบอกที่มีขั้วแม่เหล็ก N และ S ยึดติดอยู่ 2 คู่ หรือ 4 ขั้วแม่เหล็กสมมุติว่าถ้าทำให้ตัวนำทรงกระบอกอยู่กับที่แล้วให้ขั้วแม่เหล็กที่ยึดติดอยู่กับ โครงเหล็กเคลื่อนที่ทวนเข็มนาฬิกาที่เหมือนกับว่าถ้าให้ขั้วแม่เหล็กอยู่กับที่แล้วทำให้ตัวนำทรงกระบอกเคลื่อนที่ตามเข็มนาฬิกาจะเกิดการเหนี่ยวนำแม่เหล็กไฟฟ้าขึ้นที่ตัวนำทรงกระบอก โดยจะเกิดทั้งแรงเคลื่อนไฟฟ้าเหนี่ยวนำและกระแสไฟฟ้าเหนี่ยวนำ ทิศทางการเคลื่อนที่หรือการไหลของกระแสเหนี่ยวนำหาได้โดยใช้กฎมือขวาของเฟลมมิ่ง จะได้ว่ากระแสเหนี่ยวนำภายใต้ขั้วแม่เหล็ก N จะไหลจากขอบแกนทรงกระบอกเข้าสู่ภายใต้ขั้วแม่เหล็ก N และกระแสเหนี่ยวนำภายใต้ขั้วแม่เหล็ก S จะไหลออกจากแกนทรงกระบอกผ่านภายใต้ขั้วแม่เหล็ก S ไปยังขอบแกนซ้ายมือด้านนอก



รูปที่ 2.4 ตัวนำรูปทรงกระบอกเคลื่อนที่ได้เนื่องจากกระแสไฟฟ้าเหนี่ยวนำ

ดังนั้นทิศทางการเคลื่อนที่ของกระแสเหนี่ยวนำภายใต้ขั้วแม่เหล็ก N ทั้งสองขั้วนี้จะมีทิศทางเหมือนกัน คือไหลจากขอบด้านนอกของแกนตัวนำซ้ายมือผ่านภายใต้ขั้ว N ไปสู่ขอบของแกนตัวนำขวามือ และทิศทางการเคลื่อนที่ของกระแสเหนี่ยวนำภายใต้ขั้วแม่เหล็ก S ทั้งสองขั้วนี้ก็จะมี ลักษณะทิศทางเหมือนกัน คือไหลจากขอบแกนตัวนำด้านในขวามือ ผ่านภายใต้ขั้ว S ออกมาสู่ขอบแกนตัวนำด้านซ้ายมือ

เมื่อเกิดกระแสเหนี่ยวนำขึ้นภายใต้ขั้วแม่เหล็กแล้วจะทำให้แกนตัวนำทรงกระบอกเคลื่อนที่หมุนไปซึ่งทิศทางการเคลื่อนที่หมุนไปของแกนตัวนำทรงกระบอกนี้หาได้โดยใช้ กฎมือซ้ายของ คีอังกนีวหัวแม่มือนี้ชี้และนิ้วกลางให้ตั้งฉากแก่กันและกันเมื่อให้นิ้วชี้แทนทิศทางการเคลื่อนที่ของเส้นแรงแม่เหล็ก N หรือ S นิ้วกลางแทนทิศทางการไหล ของกระแสเหนี่ยวนำภายใต้ขั้วแม่เหล็ก N หรือ S แล้วนิ้วหัวแม่มือจะแทนทิศทางการเคลื่อนที่ของแกนตัวนำทรงกระบอกที่เคลื่อนที่ไปดังนั้นจะทำให้แกนตัวนำทรงกระบอกตามรูปที่ 2.4 หมุนทวนเข็มนาฬิกาตามทิศทางของขั้วแม่เหล็กที่กำหนดให้เคลื่อนที่ไว้ครั้งแรกแต่อย่างไรก็ตามเราไม่สามารถที่จะทำให้ขั้วแม่เหล็กที่ยึดติดอยู่กับ โครมมอเตอร์เคลื่อนที่ได้แต่เป็นข้อที่น่าสังเกตก็คือถ้าโรเตอร์ของมอเตอร์เคลื่อนที่หมุนไปในทิศทางใดขอให้เข้าใจว่าสนามแม่เหล็กจากขั้วแม่เหล็กที่ยึดติดกับ โครมจะต้องเคลื่อนที่ในทิศทางนั้นเสมอ

2.3 มอเตอร์เหนี่ยวนำสามเฟส

มอเตอร์ในโรงงานอุตสาหกรรมส่วนมากเป็นมอเตอร์เหนี่ยวนำสามเฟสเพราะมอเตอร์ชนิดนี้มีราคาไม่แพงมากนัก ต้องการการดูแลรักษาน้อยและมีความเร็วเกือบคงที่คือความเร็วจะลดลงจากสภาพไม่มีโหลดจนกระทั่งขับโหลดเต็มที่เพียงไม่กี่เปอร์เซ็นต์ แต่มีข้อเสียคือ

1. ควบคุมความเร็วได้ยาก
2. ขณะที่โหลดน้อยจะทำงานที่ Power Factor ต่ำและล่าช้า
3. กระแสไฟฟ้าเริ่มต้นหมุน มักจะเป็น 5 เท่า หรือ 7 เท่าของกระแสไฟฟ้าขณะขับโหลดเต็มที่ (หรือกระแสไฟฟ้าที่ปกติ)

โครงสร้างพื้นฐานของมอเตอร์เหนี่ยวนำสามเฟสมีส่วนประกอบที่สำคัญ คือ ส่วนอยู่กับที่ (Stator) และส่วนที่หมุน (Rotor) ส่วนอยู่กับที่จะมีหน้าที่ยึดขดลวดอาร์มเจอร์ที่บรรจุอยู่ในสลีต ส่วนที่หมุนจะมีหน้าที่ในการหมุนแกนซึ่งมอเตอร์เหนี่ยวนำสามเฟส มี 2 แบบประกอบไปด้วยแบบสไลควเรลเกจโรเตอร์และแบบวาวด์โรเตอร์

2.3.1 มอเตอร์เหนี่ยวนำสามเฟสแบบสไลควเรลเกจโรเตอร์

มอเตอร์แบบนี้มีใช้งานกันแพร่หลายที่สุด โดยเฉพาะอย่างยิ่งในงานอุตสาหกรรมโดยปกติแล้วมอเตอร์สามเฟสแบบนี้จะทำงานด้วยความเร็วคงที่ตลอดเวลาส่วนประกอบที่สำคัญมีดังนี้ คือ สเตเตอร์ โรเตอร์ และฝาครอบห้วท้ายซึ่งฝาครอบนี้ อาจจัดเข้าเป็นสเตเตอร์ก็ได้ แกนโรเตอร์ (Rotorcore) จะทำด้วยแผ่นเหล็กบางลามิเนตลักษณะกลมเจาะรูด้านนอกโดยรอบเป็นสลีตนำมายึดติดกันให้เป็นแกนรูปทรงกระบอกรูสลีตโดยรอบฝังด้วยตัวนำไฟฟ้าซึ่งอาจจะเป็นทองแดงหรือโลหะผสมหรืออลูมิเนียมในกรณีที่สลีตฝังด้วยแท่งตัวนำปลายแต่ละด้านของแท่งตัวนำที่โผล่ออกไปที่ด้านทั้งสองของแกนโรเตอร์จะถูกต่อเชื่อมเข้าด้วยกันด้วยแหวนตัวนำโดยการเชื่อมให้หลอมละลายติดกันแต่ก่อนใช้วิธีนี้ซึ่งมีปัญหาอย่างมาก คือรอยต่ออาจหลอมละลาย หรือแยกจากกัน ด้วยแรงเหวี่ยงหนีศูนย์กลางในขณะที่มอเตอร์กำลังทำงาน

ในปัจจุบันนี้ตัวนำที่บรรจุอยู่ในสลีตของโรเตอร์จะใช้วิธีหล่อตัวนำไฟฟ้าซึ่งปกติจะเป็นโลหะผสมหรืออลูมิเนียมเข้าไปด้วยรวมทั้งหล่อแหวนตัวนำเชื่อมตัวนำในสลีตแต่ละด้าน ของโรเตอร์ด้วย

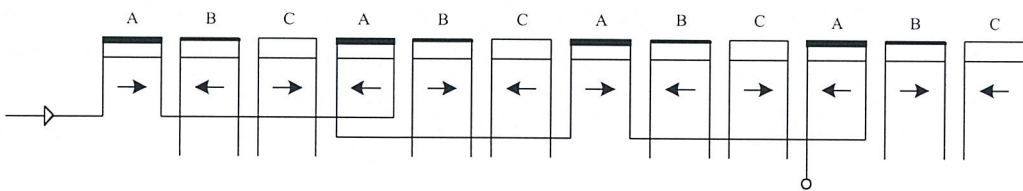
สำหรับสเตเตอร์จะประกอบด้วยโครงเหล็กแกนขดลวด และขดลวด โครงเหล็กนี้จะทำด้วยเหล็กหล่อแต่ถ้าแบบที่ผิวนอกเรียบอาจทำด้วยแผ่นเหล็กเหนียวมันให้เป็นรูปทรงกระบอกแล้วเชื่อมให้ติดกันก็ได้ส่วนด้านนอกจะมีกล่องต่อสายและขาตั้งซึ่งลักษณะของสเตเตอร์มอเตอร์สามเฟส, แกนเหล็กสเตเตอร์หรือแกนขดลวดจะทำด้วยแผ่นเหล็กลามิเนตเช่นเดียวกับแกนโรเตอร์

โดยตัดตรงกลางแผ่นลามิเนตออกแล้วเซาะด้านในให้เป็นร่องเกิดโดยรอบที่เรียกว่าสล๊อตแล้วนำเอาแผ่นลามิเนตเหล่านี้มาอัดบีบติดกัน แล้วทำการพันขดลวด สเตเตอร์

ฝาครอบหัวท้ายของมอเตอร์จะทำด้วยเหล็กหล่อมีรูตรงกลางสำหรับอัดแบร็ริงรองรับเพลลาให้ติดแน่นทั้งสองข้างที่ฝากรอบนี้จะเจาะให้มีช่องสำหรับการถ่ายเทอากาศด้วย คือเป็นช่องสำหรับให้ลมร้อนออกมาเป็นการช่วยระบายความร้อนของมอเตอร์ด้วยแบร็ริงที่อัดติดกับรูตรงกลางของฝากรอบอาจจะเป็นแบบปลอกหรือ สลึฟ (Sleeve Bearing) และแบบบอลล์ (Ball Bearing) แต่ส่วนมากจะใช้แบร็ริงแบบบอลล์ลักษณะของฝากรอบดังรูป 2. 1

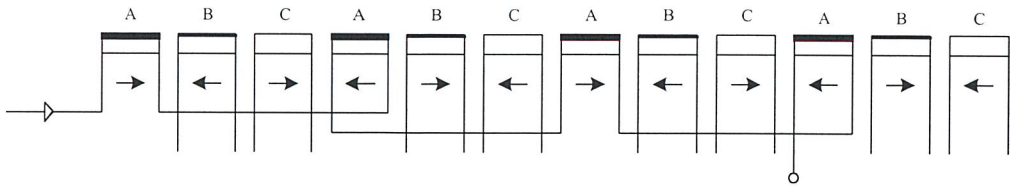
การต่อมอเตอร์สไลวเรสเกจสามเฟสใช้งานโดยมอเตอร์สามเฟสนั้น โพลแต่ละโพลจะประกอบด้วยขดลวด 3 ชุดหรือ 3 เฟส คือ เฟส 1 เฟส 2 และเฟส 3 หรือ เฟส A เฟส B และเฟส C การต่อขดลวดมี 2 แบบด้วยกันคือแบบสตาร์หรือวาย (Y) และแบบเดลต้า (Δ) ซึ่ง จะได้กล่าวถึงรายละเอียดต่อไปโดยจะเริ่มต้นภายหลังจากการพันขดลวดเสร็จเรียบร้อยแล้วขดลวดแต่ละเฟสที่ขั้วแม่เหล็กแต่ละโพลนั้นจะมีปลายสาย 2 เส้น คือ ต้นกับปลาย

การต่อขดลวดแต่ละเฟสกระทำได้ดังนี้ คือ เริ่มต้นจากขดลวดเฟส A ต่อสายไฟเข้าที่ ต้นของเฟส A ที่โพลที่ 1 แล้วต่อปลายของเฟส A ที่โพลที่ 1 เข้ากับปลายของเฟส A ที่โพลที่ 2 (ทิศทางของกระแสจะไหลสวนทางกัน) ต่อต้นของเฟส A ที่โพลที่ 2 เข้ากับต้นของ เฟส A ที่ โพลที่ 3 และต่อปลายของเฟส A ที่โพลที่ 3 เข้ากับปลายของเฟส A ที่โพลที่ 4 ในที่สุดจะเหลือต้นกับปลายของเฟส A เพียง 2 เส้นดังรูปที่ 2.5



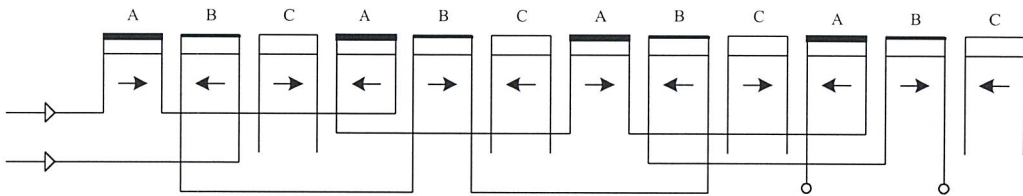
รูปที่ 2.5 การต่อขดลวดเฟส A

การต่อขดลวดเฟส B และเฟส C ก็กระทำในลักษณะเดียวกัน คือ เริ่มต้นที่เฟส B ของ โพลที่ 1 ไปเรื่อยจนถึงโพลที่ 4 จะเหลือปลายสาย 2 เส้นเช่นเดียวกับเฟส A ดังรูปที่ 2.6



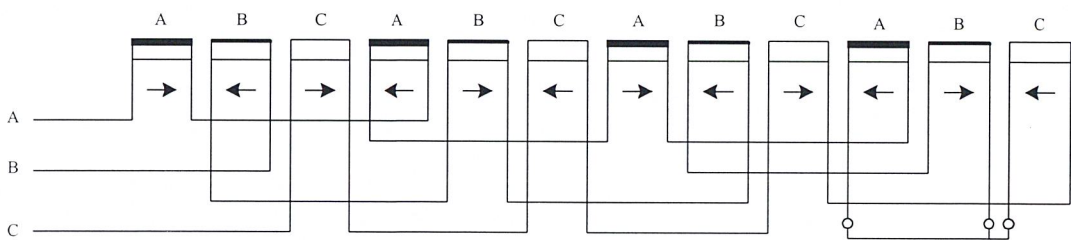
รูปที่ 2.6 การต่อขดลวดเฟส A และเฟส B

การต่อขดลวดเฟส C ก็เริ่มจากขดลวดเฟส C ที่โพลที่ 1 ไปเรื่อยจนถึงโพลที่ 4 จะเหลือปลายสายเพียง 2 เส้น ดังรูปที่ 2.7



รูปที่ 2.7 การต่อขดลวดเฟส A เฟส B และเฟส C

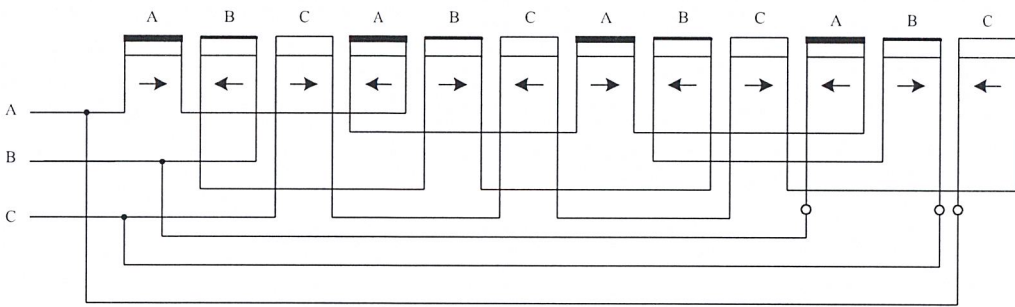
การต่อขดลวดมอเตอร์สามเฟสแบบสโควเรลเจจใช้งานมี 2 แบบด้วยกันคือ ต่อแบบสตาร์และต่อแบบเดลต้า สำหรับการต่อขดลวดให้เป็นแบบสตาร์นั้นให้นำเอาปลายของขดลวดเฟส A เฟส B และเฟส C มาต่อเข้าด้วยกัน ดังรูปที่ 2.8 จุดตอนนี้เรียกว่า Star Point



รูปที่ 2.8 การต่อขดลวดมอเตอร์สามเฟสแบบสตาร์

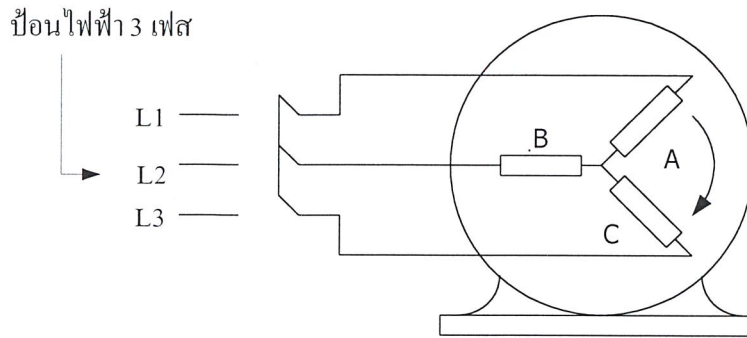
ส่วนการต่อขดลวดมอเตอร์สามเฟสแบบเดลตานั้นกระทำได้ดังนี้คือการต่อขดลวดเฟส A เฟส B และเฟส C ในรูปที่ 2.8 ให้เอาปลายสายของเฟส A ที่โพลที่ 4 ต่อเข้ากับต้นของเฟส B ที่

ส่วนการต่อขดลวดมอเตอร์สามเฟสแบบเดลต้านั้นกระทำได้ดังนี้คือจากการต่อขดลวดเฟส A เฟส B และเฟส C ในรูปที่ 2.8 ให้เอาปลายสายของเฟส A ที่โพลที่ 4 ต่อเข้ากับต้นของเฟส B ที่โพลที่ 1 และเอาปลายสายของเฟส B ที่โพลที่ 4 ต่อเข้ากับต้นของเฟส C ที่โพลที่ 1 จะได้ขดลวดแบบเดลต้านี้ดังรูปที่ 2.9 การต่อขดลวดมอเตอร์แบบสตาร์ก็เพื่อใช้กับไฟฟ้าแรงดันสูง เช่นแรงดัน 380 โวลต์ เป็นต้น และการต่อขดลวดมอเตอร์แบบเดลต้านี้เพื่อใช้กับไฟฟ้าแรงดันต่ำเช่น 220 โวลต์

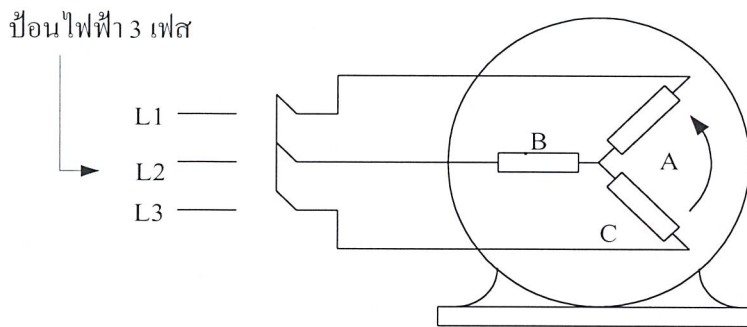


รูปที่ 2.9 การต่อขดลวดมอเตอร์สามเฟสแบบเดลต้านี้

ในการต่อสายมอเตอร์สามเฟสเพื่อให้มอเตอร์หมุนกลับทางนั้น กระทำได้ง่ายๆ ดังนี้ คือ ถ้าแต่เดิมต่อปลายสายของขดลวดเฟส A เฟส B และเฟส C เข้ากับแรงดันไฟฟ้า เฟส A เฟส B และเฟส C ตามลำดับ มอเตอร์สามเฟสตัวนั้นก็จะหมุนไปในทิศทางหนึ่ง การที่จะกลับ ทิศทางการหมุนของมอเตอร์ให้สลับปลายสายของขดลวดเฟส A กับเฟส B หรือ สลับปลาย สายของขดลวดเฟส B กับเฟส C หรือสลับปลายสายของขดลวดเฟส A กับเฟส C เพียงคู่หนึ่ง คู่ใดเพียงคู่เดียว มอเตอร์ก็จะหมุนกลับทาง ไม่ควรต่อสลับปลายสายพร้อมกันทั้งสามปลายในเวลาเดียวกัน เพราะจะทำให้มอเตอร์หมุนในทิศทางเดิมอีก การต่อปลายสายมอเตอร์สามเฟสสมมุติว่าให้หมุนตามเข็มนาฬิกา ต่อตามรูปที่ 2.10 และเมื่อต้องการต่อมอเตอร์สามเฟสให้หมุนทวนเข็มนาฬิกา ก็ต่อตามรูปที่ 2.11



รูปที่ 2.10 วิธีต่อปลายสายขดลวดมอเตอร์สามเฟสเพื่อให้หมุนในทิศทางหนึ่ง



รูปที่ 2.11 วิธีต่อปลายสายขดลวดมอเตอร์สามเฟส เพื่อให้หมุนในอีกทิศทางหนึ่งซึ่งตรงข้ามกับรูปที่ 2.10

2.3.2 มอเตอร์เหนี่ยวนำสามเฟสแบบวาวด์โรเตอร์

มอเตอร์แบบนี้จะมีใช้เฉพาะบางงานเท่านั้น โดยสามารถปรับความเร็วได้ง่าย และสะดวก ซึ่งต่างจากมอเตอร์แบบสไลด์เรลเกจที่ทำงานด้วยความเร็วคงที่เสมอ มอเตอร์แบบนี้มี ส่วนประกอบเหมือนกับมอเตอร์แบบสไลด์เรลเกจ คือประกอบด้วย สเตเตอร์ โรเตอร์และฝาครอบ สเตเตอร์และฝาครอบ โดยส่วนของจะเหมือนกับมอเตอร์แบบชนิดสไลด์เรลเกจทุก ประการ แม้แต่ การพันขดลวด การต่อขดลวด ดังนั้นจึงจะไม่กล่าวรายละเอียดเพิ่มเติมอีก ส่วนที่แตกต่างกันคือ โรเตอร์ โดยโรเตอร์แบบวาวด์โรเตอร์นี้จะใช้ทำด้วยแผ่นเหล็กลามิเนทเจาะรู เป็นสล๊อตและมีรู สำหรับการระบายความร้อนแล้วนำแผ่นลามิเนทแต่ละแผ่นมาอัดเข้าด้วยกันเป็นแกน โรเตอร์เช่นเดียวกับแกนโรเตอร์ของมอเตอร์สไลด์เรลเกจทุกประการที่แตกต่างกันจริงๆก็เฉพาะขดลวดที่พันลงในสล๊อต โรเตอร์จะพันด้วยเส้นลวดทองแดงกลมหรือแบนซึ่งแล้วแต่จะออกแบบโดยพันให้แต่ละเฟสห่างกัน 120 องศาไฟฟ้าและพันให้มีจำนวนโพลที่สเตเตอร์ทุกประการด้วยเมื่อพันขดลวด

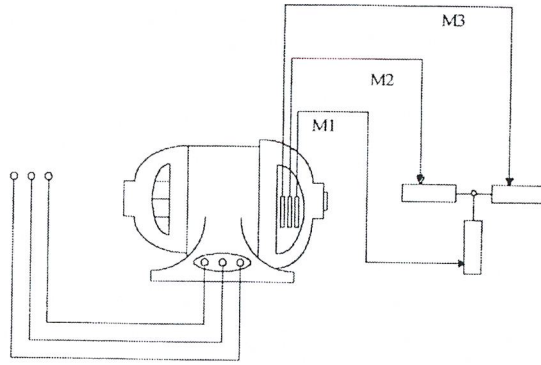
ลงในสล็อตโรเตอร์เสร็จแล้วจะต่อขดลวดเป็นแบบสตาร์เหลือปลายสายต่อเข้ากับสลีปริงเพียง 3 ปลายเท่านั้นสลีปริงที่ยึดติดอยู่กับเพลารอเตอร์นี้แต่ละตัวจะมีฉนวนรองรับเพื่อไม่ให้เกิดการติดต่อกันทางไฟฟ้าที่สลีปริงจะมีแปรงถ่าน (Carbon Brushes) วางสัมผัสอยู่ในช่องแปรงถ่านและมีสปริงกดให้สัมผัสแน่นตลอดเวลาด้วยจากแปรงถ่านนี้จะต่อสายออกไปภายนอกเพื่อต่อเข้ากับความต้านทานเพื่อ ควบคุมความเร็วต่อไป

มอเตอร์เหนี่ยวนำสามเฟสแบบวาวด์โรเตอร์มีหลักการทำงานเมื่อป้อนแรงดันไฟฟ้ากระแสสลับให้กับขดลวดสเตเตอร์จะเกิดสนามแม่เหล็กหมุนและมีความเร็วที่เรียกว่าความเร็วซิงโครนัสความเร็วซิงโครนัสนี้จะมากหรือน้อยก็ขึ้นอยู่กับจำนวน โพลของมอเตอร์สนามแม่เหล็กหมุนจะยุบตัวและขยายตัวตัดกับขดลวดที่ส่วนโรเตอร์ทำให้เกิดแรงเคลื่อนเหนี่ยวนำและกระแสเหนี่ยวนำด้วยกระแสเหนี่ยวนำนี้จะไหลผ่านสลีปริงผ่านความต้านทานภายนอกที่ต่อร่วมกับขดลวดโรเตอร์เพื่อช่วยในการสตาร์ท และปรับความเร็ว

ขณะเริ่มเดินความต้านทานภายนอกจะต่อแบบสตาร์ความต้านทานที่ต่อเข้าไปร่วมกับขดลวดโรเตอร์จะมีผลทำให้แรงบิดเริ่มต้นดีขึ้นและเป็นผลทำให้อิมพีแดนซ์ของส่วนโรเตอร์เพิ่มขึ้นกระแสโรเตอร์ก็จะลดลงแต่กระแสเก็บบจะมีเฟสเดียวกันกับแรงเคลื่อนเหนี่ยวนำและเส้นแรงแม่เหล็กจากขดลวดสเตเตอร์จึงเป็นผลทำให้แรงบิดตอนสตาร์ทสูงขึ้น

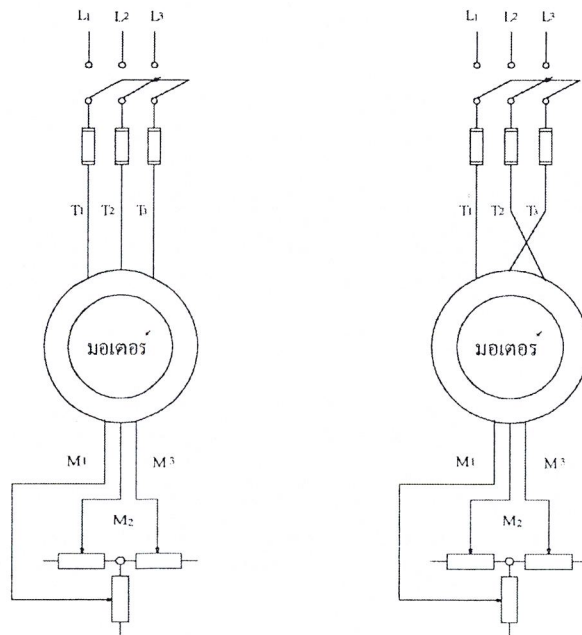
เมื่อความเร็วมอเตอร์สูงขึ้นแล้วจึงค่อยๆ ลดความต้านทานภายนอกออกจากวงจรของขดลวดโรเตอร์เมื่อลดความต้านทานภายนอกออกหมดสลีปริงจะต่อลัดวงจรที่จุดต่อสตาร์ของความต้านทานภายนอกขณะนี้มอเตอร์จะหมุนด้วยความเร็วเต็มพิกัด (Rated Speed) ถ้าต้องการปรับความเร็วของมอเตอร์แบบนี้ก็ให้เพิ่มความต้านทานหรือให้ลดความต้านทานภายนอกเข้ากับวงจรขดลวดโรเตอร์ปกติแล้วจะปรับความเร็วให้ต่ำกว่าความเร็วเต็มพิกัด

การต่อมอเตอร์สามเฟสแบบวาวด์โรเตอร์เพื่อใช้งานใช้วิธีการต่อขดลวดสเตเตอร์เริ่มต้นตั้งแต่การต่อกลุ่มคอยล์ (Coil Group) หรือต่อขดลวดแต่ละชุดหรือแต่ละเฟสภายในแต่ละโพลการต่อขดลวดแต่ละเฟสของมอเตอร์การต่อขดลวดเป็นแบบสตาร์หรือเดลต้าทำได้เช่นเดียวกับมอเตอร์สามเฟสแบบสไลควอเรลเกจ ในการต่อมอเตอร์สามเฟสแบบวาวด์โรเตอร์เข้ากับแหล่งจ่ายไฟฟ้า ต่อได้ตามรูปที่ 2.12



รูปที่ 2.12 การต่อมอเตอร์สามเฟสแบบวาวด์โรเตอร์

สำหรับการต่อมอเตอร์สามเฟสแบบวาวด์โรเตอร์ให้หมุนกลับทิศทางก็กระทำได้เช่นเดียวกับมอเตอร์สามเฟสแบบสไลด์เรลเกจกล่าวคือให้สลับปลายสายที่ต่อเข้ากับส่วนขดลวดสเตเตอร์เพียงคู่หนึ่งคู่ใดเท่านั้น ดังรูปที่ 2.13 ในรูปที่ 2.13 ก. เป็นการต่อมอเตอร์ให้หมุนใน ทิศทางหนึ่ง และรูปที่ 2.13 ข. เป็นการต่อมอเตอร์ให้หมุนในอีกทิศทางหนึ่งซึ่งตรงกันข้าม แต่ถ้าต่อปลายสายที่โรเตอร์ให้สลับกัน มอเตอร์จะไม่หมุนกลับทางแต่อย่างใด



ก.) มอเตอร์จะหมุนในทิศทางหนึ่ง ข.) มอเตอร์จะหมุนในอีกทิศทางหนึ่ง

รูปที่ 2.13 การต่อมอเตอร์สามเฟสแบบวาวด์โรเตอร์ให้หมุนกลับทิศทาง

2.4 วิธีสตาร์ทมอเตอร์เหนี่ยวนำ

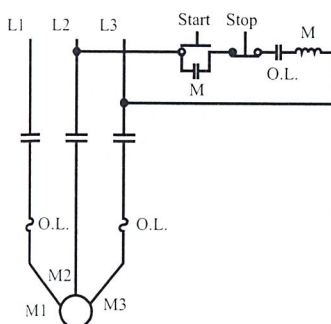
มอเตอร์ไฟฟ้ากระแสลับที่เรียกว่าอินดักชันมอเตอร์ก็เหมือนกับหม้อแปลงไฟฟ้านั่นก็คือโรเตอร์ทำหน้าที่เป็นขดลวดทุกขดขั้วจะต่างกันเพียงชนิดเดียวคือโรเตอร์ของมอเตอร์จะเชื่อมครบวงจรแบบวงจรตัดตลอดเวลา ดังนั้นเมื่อป้อนไฟสลับให้กับขดลวดสเตเตอร์แล้วกระแสอินพุตของขดลวดสเตเตอร์ (ขดลวดปฐมภูมิ) จะสูงมากโดยเฉพาะอย่างยิ่งในช่วงสตาร์ทเนื่องจากในช่วงสตาร์ทนี้จะยังไม่มีแรงเคลื่อนเหนี่ยวนำย้อนกลับเกิดต้านกับแรงดันป้อนกระแสในช่วงสตาร์ทของมอเตอร์เหนี่ยวนำจะมีค่าประมาณ 5-7 เท่าของกระแสเมื่อมอเตอร์ทำงานเต็มพิกัดแต่แรงบิดในช่วงนี้จะมีค่าประมาณ 1.5- 2.5 เท่าของแรงบิดที่โหลดเต็มพิกัดเท่านั้น กระแสไฟฟ้าที่มีค่าสูงในช่วงนี้จะส่งผลทำให้เกิดแรงดันไฟลดลง (Drop) ในสายอินจะมีผลต่อการทำงานของอุปกรณ์ ไฟฟ้าอื่นที่ต่อร่วมในวงจรเดียวกัน นั่นคือมอเตอร์ที่มี ขนาดตั้งแต่ 30 แรงม้าขึ้นไปจึงไม่ควรที่จะสตาร์ทโดยตรง (Direct Start)

แรงบิดสตาร์ทของมอเตอร์เหนี่ยวนำสามารถที่จะแก้ไขให้ดีขึ้นได้โดยการเพิ่มความต้านทานให้กับวงจรโรเตอร์ซึ่งกรณีนี้สามารถใช้กับมอเตอร์สามเฟสแบบสลีปรिंगได้เป็นอย่างดี แต่ไม่สามารถไปใช้กับมอเตอร์แบบสไลวเรลเกจได้ อย่างไรก็ตามวิธีการลดกระแสสตาร์ทก็สามารถทำได้โดยการลดแรงกันป้อนของสเตเตอร์ในช่วงสตาร์ท เมื่อมอเตอร์ทำงานแล้วจึงเพิ่มแรงดันให้หิมค่าตามพิกัดกำหนด

2.4.1 วิธีสตาร์ทมอเตอร์เหนี่ยวนำแบบสไลวเรลเกจโรเตอร์

1.) วิธีสตาร์ทโดยตรง (Direct Start หรือ On Line Start)

วิธีนี้มักจะใช้กับมอเตอร์ขนาดแรงม้าต่ำๆ คือไม่ควรเกิน 30 แรงม้า ดังได้กล่าวแล้วว่ากระแสสตาร์ทจะมีค่าที่สูงมาก ประมาณ 5-7 เท่าของกระแสเต็มพิกัด แต่แรงบิดต่ำมากคือเพียง 1.5-2.5 เท่าของแรงบิดที่โหลดเต็มพิกัดเท่านั้น วงจรสตาร์ทโดยตรงเป็นดังรูปที่ 2.14



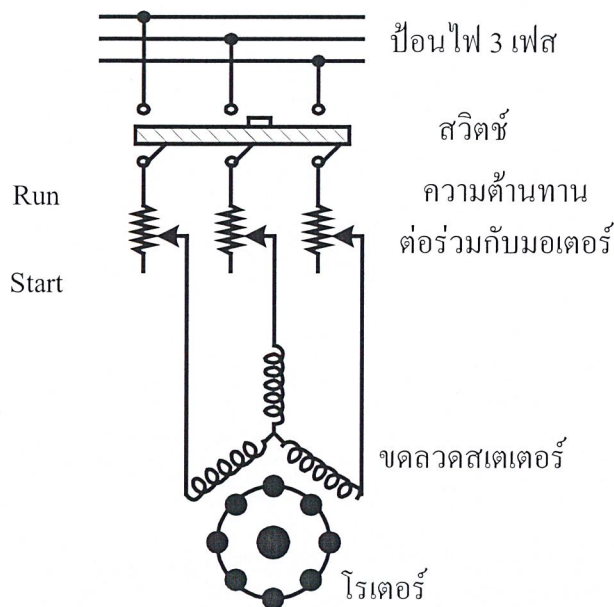
รูปที่ 2 .14 วงจรสตาร์ทโดยตรง

2.) วิธีสตาร์ทโดยลดแรงดันป้อน (Reduce Voltage Start)

วิธีสตาร์ทมอเตอร์เหนี่ยวนำแบบลดแรงดันป้อนนี้มีหลายวิธีด้วยกันดังนี้

ก. ต่อความต้านทานอันดับกับมอเตอร์ (Primary Resistor or Rheostat)

การต่อวิธีนี้เพื่อที่จะลดแรงดันป้อนในช่วงสตาร์ทให้ต่ำลง โดยที่กระแสไฟฟ้าในช่วงสตาร์ทก็จะลดลงด้วย เนื่องจากกระแสจะแปรผันโดยตรงกับแรงดัน และแรงบิดแปรผันกับแรงดันกำลังสอง สมมุติว่าถ้าแรงดันป้อนลดลง 500 % แต่แรงบิดสตาร์ทจะลดลงเพียง 25% ของแรงบิดเมื่อโหลดเต็มพิกัดเท่านั้น เมื่อมอเตอร์ทำงานแล้วมีความเร็วสูงขึ้นแล้วจึงค่อยๆลดความต้านทานออกจากวงจรขดลวดสเตเตอร์วงจรสตาร์ทแบบนี้ดังรูปที่ 2.15



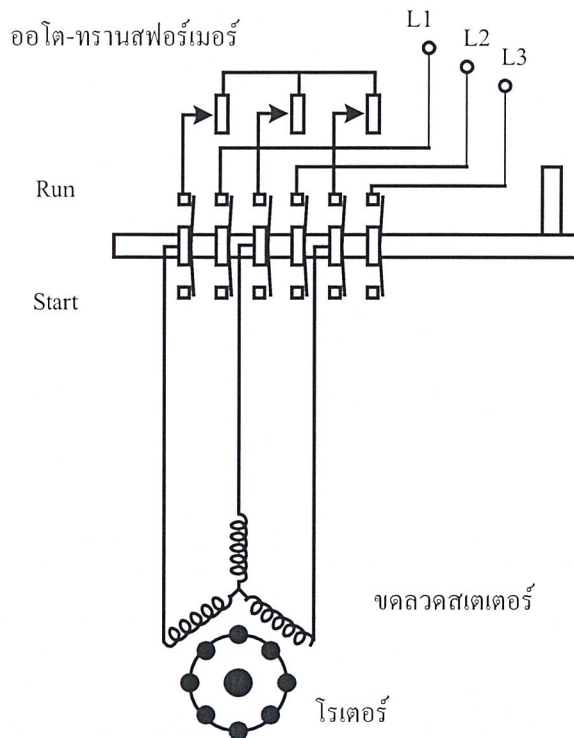
รูปที่ 2.15 วงจรสตาร์ทลดแรงดันป้อนโดยต่อความต้านทานอันดับกับมอเตอร์

ข. วิธีต่อออโต-ทรานส์ฟอร์มเมอร์ (Auto-Transformer) อันดับกับมอเตอร์

วิธีสตาร์ทมอเตอร์แบบนี้บางทีเรียกว่า ออโต-สตาร์ทเตอร์ (Auto-Starter) หรือเรียกว่าคอมเพนเซเตอร์ (Compensator) ซึ่งจะประกอบด้วย ออโต-ทรานส์ฟอร์มเมอร์ และเป็นสวิตช์อัตโนมัติตามรูปใช้ออโต-ทรานส์ฟอร์มเมอร์หนึ่งเฟส 3 ตัว หรือออโต-ทรานส์ฟอร์มเมอร์สามเฟส 1 ตัว เมื่อเริ่มสตาร์ทจะสับสวิตช์ไปยังตำแหน่งสตาร์ท จะสังเกตเห็นว่าแรงดันป้อน มอเตอร์จะมีค่าลดลงเหลือเพียง $X V$ เท่านั้น เมื่อมอเตอร์มีความเร็วสูงขึ้นประมาณ 80% ของความเร็วเต็มพิกัดแล้วจึงสับสวิตช์ไปยังตำแหน่งรัน (Run) ขณะนี้แรงดันป้อนมอเตอร์จะมีค่าตามพิกัดคือ V สวิตช์ที่ใช้ถ้า

ใช้กับมอเตอร์ขนาดแรงม้าต่ำจะใช้ชนิด Air Break แต่ถ้ามอเตอร์ขนาดแรงม้าสูงจะใช้สวิตซ์ชนิด Oil-Immersed ทั้งนี้เพื่อลดการสปาร์ก (Spark) ที่หน้าสัมผัส

วิธีสตาร์ทแบบนี้จะออกแบบให้มีรอยต่อ (Tap) ที่หม้อแปลงเป็น 3 ขนาดด้วยกัน คือ 80, 65 และ 50 เปอร์เซ็นต์ของแรงดันป้อนเต็มพิกัด เพื่อให้สามารถเลือกแรงดันสตาร์ทได้อย่างเหมาะสม วงจรสตาร์ทแบบนี้ดังรูปที่ 2.16

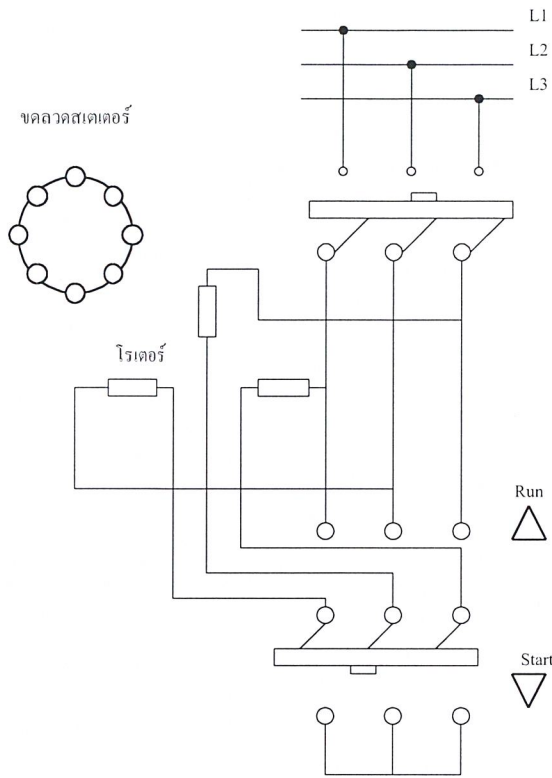


รูปที่ 2.16 วงจรสตาร์ทแบบต่อออโต-ทรานส์ฟอร์เมอร์อันดับกับมอเตอร์

ค. วิธีสตาร์ทแบบสตาร์เดลต้า (Star-Delta Starter)

วิธีนี้จะใช้กับมอเตอร์สามเฟสที่มีลักษณะที่ขดลวดสเตรเตอร์ทำงานแบบเดลต้าเท่านั้น วงจรสตาร์ทจะประกอบด้วยสวิตซ์ชนิด 2 ทาง แบบ TPDT โดยที่สวิตซ์นี้จะทำการต่อขดลวดสเตรเตอร์แบบสตาร์ในตอนสตาร์ท และต่อขดลวดแบบเดลต้าเมื่อมอเตอร์ทำงานปกติขณะสตาร์ท ขดลวดต่อแบบสตาร์นั้น แรงดันป้อนขดลวดแต่ละขุคจะมีค่าของแรงดันปกติ แต่แรงบิดสตาร์ทจะมีค่าเพียง 1/3 ของแรงบิดเมื่อโหลดเต็มพิกัด คือเมื่อมอเตอร์ต่อแบบเดลต้า และทำงานที่แรงดันเต็มพิกัด และกระแสสตาร์ทจะลดลงในสัดส่วนเดียวกันกับแรงดันป้อนขณะสตาร์ท คือลดลง 1/3 ของ

กระแสไหลลดเต็มพิกัด วิธีสตาร์ทมอเตอร์วิธีนี้มีราคาถูก และการควบคุมทำได้ง่ายและสะดวก วงจรสตาร์ทดังรูปที่ 2.17



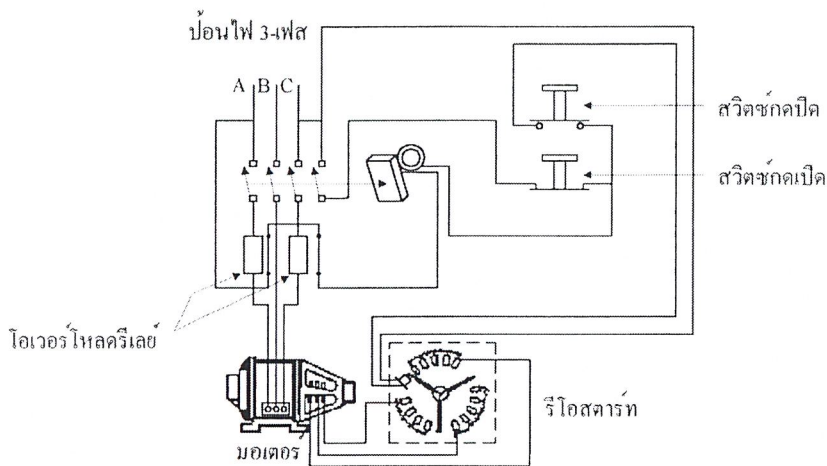
รูปที่ 2.17 วงจรสตาร์ทแบบสตาร์- เดต้า

2.4.2 วิธีสตาร์ทมอเตอร์เหนี่ยวนำแบบวาวด์โรเตอร์

มอเตอร์แบบวาวด์โรเตอร์หรือสลีปรिंगมอเตอร์นี้มักจะสตาร์ทด้วยแรงดันเต็มพิกัดคือป้อนแรงดันตามพิกัดให้กับขดลวดสเตเตอร์เลยกระแสไฟฟ้าตอนสตาร์ทจะถูกปรับให้ลดลงโดยการต่อความต้านทานที่ปรับค่าได้เข้ากับวงจรโรเตอร์ ความต้านทานที่ปรับค่าได้ หรือที่เรียกว่า รีโอสตาร์ทจะต่อร่วมกับโรเตอร์ด้วยการต่อแบบสตาร์ทความเร็วสูงขึ้นแล้วจึงค่อยๆ ลดความต้านทานลงจนกระทั่งตัดออกหมด และต่อลัดวงจรโรเตอร์ เหมือนกับรูปที่ 2.12 และ 2.13

การเพิ่มความต้านทานเข้าไปในโรเตอร์นั้นไม่เพียงแต่ทำให้กระแสสตาร์ท ลดลงเท่านั้น ยังทำให้แรงบิดสตาร์ทสูงขึ้นด้วย เนื่องจากเพาเวอร์แฟคเตอร์ดีขึ้น

รีโอสตาร์ทที่ต่อร่วมกับมอเตอร์อาจจะใช้แบบปรับความต้านทานได้เป็นขั้นๆ หรือแบบปรับค่าได้อย่างต่อเนื่องก็ได้และอาจจะเป็นแบบบังคับด้วยมือหรือแบบงานอัตโนมัติก็ได้ ซึ่งการสตาร์ทมอเตอร์เหนี่ยวนำแบบวาวด์โรเตอร์เป็นวงจรดังรูปที่ 2.18



รูปที่ 2.18 วงจรสตาร์ทสลลิปริงมอเตอร์แบบต่อตรง

2.5 วิธีควบคุมความเร็วของมอเตอร์เหนี่ยวนำ

มอเตอร์เหนี่ยวนำสามเฟสจะมีความเร็วค่อนข้างคงที่ความเร็วเปลี่ยนแปลงที่โหลดเต็ม พิกัด (Speed Regulation) จะน้อยกว่า 5% การควบคุมความเร็วทำได้ยาก และมีความยุ่งยากกว่าการควบคุมความเร็วของ ดี.ซี. ซึ้นมอเตอร์ วิธีควบคุมความเร็วของมอเตอร์เหนี่ยวนำสามารถแบ่งได้เป็นการควบคุมความเร็วทางด้านสเตเตอร์และการควบคุมความเร็วทางด้านโรเตอร์

2.5.1 การควบคุมความเร็วทางด้านสเตเตอร์

การควบคุมความเร็วทางด้านสเตเตอร์มีวิธีการทำดังนี้

1.) เปลี่ยนแปลงแรงดันที่ป้อน

เป็นวิธีที่ง่ายและราคาถูกแต่ไม่นิยมนำไปใช้เพราะความเร็วเปลี่ยนแปลงได้น้อยมาก และจะมีผลทำให้ความหนาแน่นเส้นแรงแม่เหล็กของมอเตอร์เปลี่ยนแปลงด้วย

2.) เปลี่ยนความถี่ที่ป้อน

วิธีนี้ก็มักไม่ค่อยนำไปใช้ ทั้งนี้จากกฎที่ว่า ความเร็วเชิงโรตอร์จะเปลี่ยนแปลงเมื่อความถี่เปลี่ยนไปวิธีนี้จะใช้กรณีที่มีมอเตอร์เหนี่ยวนำเป็นโหลดของเครื่องกำเนิดเท่านั้นความถี่จากเครื่อง

กำเนิดเปลี่ยนแปลงได้เมื่อปรับความเร็วของต้นกำลังที่ขับเคลื่อนเครื่องกำเนิดอย่างไรก็ตามการปรับความเร็วของต้นกำลังที่ขับเคลื่อนเครื่องกำเนิดก็ยังมีขีดจำกัดอยู่

3.) เปลี่ยนจำนวนโพลของสเตเตอร์

วิธีนี้สามารถนำไปใช้กับมอเตอร์สโควเรลเจกได้เพราะสเตเตอร์จะมีจำนวนโพลมากเท่าไรก็ตามโรเตอร์ก็จะมีจำนวนเท่ากันด้วยซึ่งเมื่อจำนวนโพลเปลี่ยนแปลงไปความเร็วเชิงโรตอร์สก็เปลี่ยนแปลงตามไปด้วยวิธีเปลี่ยนจำนวนโพลของสเตเตอร์กระทำได้โดยการพันขดลวดสามเฟสหลายชุดลงในสล็อต เช่น มอเตอร์ขนาด 36 สล็อต อาจจะใช้พันขดลวดสเตเตอร์ไว้ 2 ชุด ซึ่งอาจจัดเป็นชนิด 4 โพล และ 6 โพล ถ้าความถี่ป้อน 50 ไซเคิล ความเร็วจะมีค่า 1500 และ 1000 รอบ/นาที ถ้าพันขดลวดไว้เป็น 3 ชุด ซึ่งอาจจัดเป็น 4 โพล 6 โพล และ 8 โพล ความเร็วจะมีค่า 1500, 1000 และ 750 รอบ/นาที ขดลวดสเตเตอร์แต่ละชุดจะถูกใช้งานไม่พร้อมกัน ถ้าให้ขดลวดชุดหนึ่งทำงานขดลวดชุดอื่นจะถูกปล่อยไว้เฉยๆเมื่อจะให้มอเตอร์ทำงานที่ความเร็วอื่นก็จะปลดขดลวดชุดเก่าออกและต่อขดลวดชุดใหม่แทนซึ่งก็มีความยุ่งยากบ้างแต่ถ้าใช้สวิตช์อัตโนมัติก็จะกระทำได้รวดเร็วขึ้นวิธีนี้มักจะมีกับมอเตอร์ลิฟต์และมอเตอร์รถรางและรถไฟ และใช้กับมอเตอร์ขนาดเล็กที่ขับเคลื่อนมือกล

2.5.2 การควบคุมความเร็วทางด้านโรเตอร์

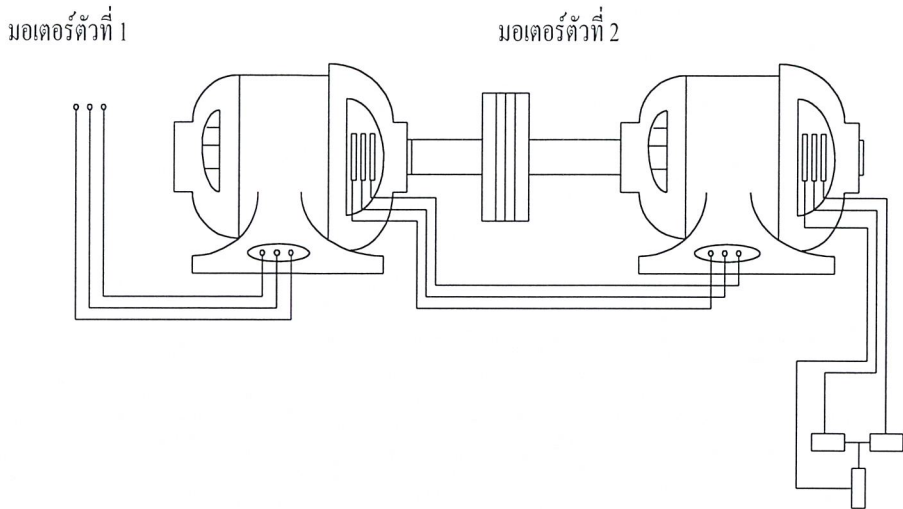
การควบคุมความเร็วทางด้านโรเตอร์มีวิธีการดังนี้

1.) ต่อรีโอสตาร์ทเข้ากับมอเตอร์

วิธีนี้สามารถนำไปใช้กับมอเตอร์เหนี่ยวนำสลลิ่งเท่านั้นโดยจะทำให้ความเร็วของมอเตอร์ลดลงสตาร์ทเตอร์ที่ใช้กับโรเตอร์จะต้องเป็นแบบที่สามารถใช้งานได้ต่อเนื่องซึ่งอย่างไรก็ตามเมื่อต่อรีโอสตาร์ทเข้าไปในวงจรโรเตอร์จะทำให้เกิดความสูญเสียเนื่องจากมีกระแสเพิ่มขึ้นเป็นเหตุทำให้ประสิทธิภาพของมอเตอร์ลดลงนั่นก็คือความสูญเสียของกระแสจะเป็นสัดส่วนโดยตรงกับความเร็วที่ลดลงวิธีนี้จึงมักใช้ในช่วงเวลาสั้นๆเท่านั้น

2. วิธีการให้มอเตอร์สองตัวทำงานร่วมกัน

วิธีนี้จะใช้มอเตอร์ 2 ตัวทำงานร่วมกันโดยต่อร่วมเพลาดียวกันดังรูปที่ 2.19 ดังนั้นมอเตอร์ทั้ง 2 ตัวจะหมุนด้วยความเร็วเท่ากับ



รูปที่ 2.19 วิธีควบคุมความเร็วโดยใช้มอเตอร์ 2 ตัวทำงานร่วมกัน

ต่อขลวดสเตเตอร์ของมอเตอร์ตัวที่ 1 เข้ากับสายเมนของการไฟฟ้า ขณะเดียวกันก็ต่อเอาต์พุตจากมอเตอร์ตัวที่ 1 เข้ากับขลวดสเตเตอร์ของมอเตอร์ตัวที่ 2 นั่นก็คือมอเตอร์ตัวที่ 1 จะต้องเป็นมอเตอร์สลีปริ่งโดยที่ขลวดสเตเตอร์และขลวดโรเตอร์จะต้องมีสัดส่วน 1:1

หลักการทำงานก็คือ เมื่อป้อนแรงดันด้วยความถี่ f ให้กับมอเตอร์ตัวที่ 1 จะเกิดแรงเคลื่อนเหนี่ยวนำด้วยความถี่ f เกิดที่โรเตอร์ของมอเตอร์ตัวที่ 1 และแรงเคลื่อนเหนี่ยวนำนี้ก็จะป้อนให้กับมอเตอร์ตัวที่ 2 มอเตอร์ทั้งสองก็จะเกิดแรงบิดขึ้นมา เมื่อความเร็วสูงขึ้นความถี่ของโรเตอร์ของมอเตอร์ตัวที่ 1 ก็จะลดลงทำให้ความเร็วซิงโครนัสของมอเตอร์ตัวที่ 2 ลดลงด้วยเป็นเหตุให้ความเร็วของมอเตอร์ตัวที่ 1 ลดลงด้วย อย่างไรก็ตามวิธีนี้ก็ไม่ค่อยได้นำไปใช้

3.) วิธีการป้อนแรงเคลื่อนเหนี่ยวนำให้กับโรเตอร์

วิธีการนี้มีหลักการง่ายๆ คือป้อนแรงเคลื่อนหรือแรงดันไฟฟ้าให้กับโรเตอร์ ของมอเตอร์ตัวที่จะปรับความเร็วโดยใช้ความถี่ของแรงดันป้อน โรเตอร์เท่ากับความถี่เหนี่ยวนำของมอเตอร์ด้วย มอเตอร์ที่จะควบคุมความเร็วต้องเป็นมอเตอร์สลีปริ่งถ้าแรงดันป้อนมีเฟสเดียวกับแรงเคลื่อนเหนี่ยวนำของโรเตอร์จะทำให้กระแสในโรเตอร์มากขึ้นทำให้ดูเหมือนว่าความต้านทานลดลงและถ้าแรงดันป้อนมีเฟสต่างกัน 180 องศา กับแรงเคลื่อนเหนี่ยวนำของโรเตอร์จะทำให้กระแสโรเตอร์ลดลงทำให้ดูเหมือนว่าความต้านทานโรเตอร์เพิ่มขึ้น ลักษณะแบบนี้จึงทำให้ความเร็วของมอเตอร์เปลี่ยนแปลงไป

2.6 วิธีการระบุตำแหน่งการหมุนให้กับมอเตอร์เหนี่ยวนำ

การระบุตำแหน่งการหมุนให้กับมอเตอร์เหนี่ยวนำจะต้องศึกษาจากคุณลักษณะของมอเตอร์เหนี่ยวนำซึ่งมอเตอร์เหนี่ยวนำแบบวาล์วโรเตอร์ที่ปรับความเร็วได้ จะใช้ความต้านทานที่ปรับค่าจากภายนอกได้ต่ออนุกรมเข้ากับขดลวดโรเตอร์ เพื่อปรับความเร็วหรือเพิ่มแรงบิดเริ่มต้น หมุนมอเตอร์แบบนี้นำไปใช้กับโหลดที่ต้องเริ่มต้นหมุนบ่อยๆ เช่น ปั่นจั่น เป็นต้น ส่วนงานที่ต้องการกำลังกลก่อนข้างคงที่และเริ่มต้นหมุนไม่บ่อยนัก เช่นมอเตอร์ที่ใช้ขับปั๊มเครื่องเป่าอากาศ พัดลม เป็นต้น จะใช้มอเตอร์เหนี่ยวนำที่มีโรเตอร์แบบทรงกระบอก

คุณลักษณะแรงบิดและความเร็วในมอเตอร์เหนี่ยวนำ 3 เฟสทำให้ทราบข้อมูลการทำงานที่มีผลต่อการระบุตำแหน่งการหมุนของมอเตอร์ดังนี้

1. แรงบิดที่เกิดขึ้น (Developed Torque) เป็นศูนย์ที่ความเร็วซิงโครนัส
2. ความแตกต่างระหว่างแรงบิด และความเร็วเกือบเป็นเชิงเส้น ในช่วงระหว่างแรงบิดขณะไม่มีโหลด (No Load Torque) กับแรงบิดขณะจ่ายโหลดเต็มที่ (Full Load Torque) เราเรียกช่วงนี้ว่าย่านทำงานปกติ (Normal Operating Range) ของมอเตอร์
3. มีแรงบิดสูงสุด (Maximum Torque ; T_{max} ซึ่งบางครั้งเรียกว่า Pull-Out Torque) เป็น 2-3 เท่า ของแรงบิดที่พิกัดขนาดจ่ายโหลดเต็มที่
4. มีแรงบิดเริ่มต้นหมุน (Starting Torque) มากกว่าแรงบิดที่พิกัดเล็กน้อย ดังนั้นมอเตอร์จึงเริ่มต้นขับโหลดหรือต่อกับโหลดขณะเริ่มต้นหมุนได้ทันทีโดยแรงบิดที่โหลด นั้นจะต้องไม่มากกว่าแรงบิดที่พิกัด
5. การสลับสายคู่หนึ่งจะทำให้สนามแม่เหล็กเปลี่ยนทิศทางมอเตอร์จึงหมุนกลับทิศทางได้ แต่ก่อนที่จะกลับทิศทางนั้นจะมีอยู่จุดหนึ่งที่มอเตอร์หยุดอยู่กับที่อย่างรวดเร็วซึ่งเราเรียกว่า Plugging ซึ่งนำไปสู่หลักการควบคุมมอเตอร์แบบหนึ่ง
6. ถ้าโรเตอร์ของมอเตอร์เหนี่ยวนำถูกขับด้วยความเร็วที่สูงกว่า ความเร็วซิงโครนัสจะมีผลทำให้แรงบิดที่เกิดขึ้นในมอเตอร์กลับทิศทางและมอเตอร์ดังกล่าวจะทำงานเป็นเครื่องกำเนิดไฟฟ้า ซึ่งเปลี่ยนกำลังกลเป็นกำลังไฟฟ้า

การที่จะระบุตำแหน่งการหมุนของมอเตอร์เพื่อที่จะให้มอเตอร์สามารถหมุนไปได้ ตามทิศทางที่ต้องการจะต้องมีการคำนวณดังนี้

1.) การคำนวณแรงบิดเริ่มต้นหมุน

แรงบิดที่เริ่มต้นหมุนเกิดขึ้นขณะโรเตอร์ยังไม่เคลื่อนที่ทำให้ค่าสลิป (s) เท่ากับ 1 จึงเขียนสมการดังนี้

$$I_{R(st)} = \frac{V_1}{\sqrt{(R_S + R'_R) + (X_S + X'_R)^2}} \quad (2.1)$$

ถ้าต้องการทราบค่าแรงบิดเริ่มต้นหมุนจะต้องหาค่า Input Power ของโรเตอร์ขณะทำการเริ่มต้นหมุน ($P_{iR(st)}$) โดยใช้สมการ

$$P_{iR(st)} = 3I_{R(st)}^2 R'_R \quad (2.2)$$

ขณะเริ่มต้นหมุนฟลักแม่เหล็กที่สเตเตอร์ซึ่งหมุนด้วยความเร็วเชิงโคโรนัสจะทำให้เกิดการเหนี่ยวนำและสร้างแรงบิดที่เริ่มต้นหมุน (T_{st}) ที่โรเตอร์ดังนั้นความเร็วเชิงมุมทางกลที่ทำให้เกิด (T_{st}) จะเป็ความเร็วเชิงมุมเชิงโคโรนัส (ω_s) เขียนสมการได้ดังนี้

$$T_{st} = \frac{P_{iR(st)}}{\omega_s} \quad (2.3)$$

โดย

$$\omega_s = \frac{2\pi n_s}{60} \quad (2.4)$$

2.) การคำนวณแรงบิดสูงสุด (Maximum Torque)

โดยทั่วไปการเพิ่มขึ้นของ โหลดจะมีขนาดต่ำกว่าค่าพิคคของมอเตอร์เล็กน้อยแต่ถ้าโหลดเพิ่มขึ้นเรื่อยๆจะมีอยู่จุดหนึ่งที่มอเตอร์ไม่อาจสร้างแรงบิดเพิ่มขึ้นได้อีกเรียกว่ามอเตอร์มีแรงบิดสูงสุด ถ้าโหลดเพิ่มขึ้นเกินกว่านี้มอเตอร์จะหยุดหมุน

จากหลักการส่งผ่านกำลังไฟฟ้าสูงสุดของทฤษฎีวงจรไฟฟ้าทำให้ทราบว่า เมื่อวงจรมีอิมพีแดนซ์ของโหลด (RMS) มีค่าเหมาะสมกับอิมพีแดนซ์ของแหล่งจ่ายไฟฟ้าอิมพีแดนซ์ของโหลดจะได้รับกำลังไฟฟ้าสูงสุดซึ่งสอดคล้องกับการหาค่าสลิปขณะแรงบิดสูงสุด (S_{mt}) ที่หาได้จาก

$$S_{mt} = \frac{R'_R}{\sqrt{R_s^2 + (X_s + X'_R)^2}} \quad (2.5)$$

ตามปกติความต้านทานของสเตเตอร์ (RS) จะมีค่าน้อยเมื่อเทียบกับรีแอกแตนซ์เทียบเคียง ($X_e = X_s + X'_R$) จึงเขียนสมการ (2.5) ใหม่ ได้ดังนี้

$$S_{mt} = \frac{R'_R}{X_e} \quad (2.6)$$

เมื่อ $X_e =$ รีแอกแตนซ์เทียบเคียง

ถ้าใช้ค่า S_{mt} นี้หาค่ากระแสไฟฟ้าที่โรเตอร์ขณะแรงบิดสูงสุด ($I_{R(mt)}$) จะได้สมการ

$$I_{R(mt)} = \frac{V_1}{\sqrt{2X_e}} \quad (2.7)$$

และได้สมการ

$$P_{iR(mt)} = 3I_{R(mt)}^2 \frac{R'_R}{S_{mt}} = \frac{3V_1^2}{2X_e} \quad (2.8)$$

สามารถหาค่าแรงบิดสูงสุดของโรเตอร์ได้จาก

$$T_{mt} = \frac{P_o(mt)}{\omega_r(mt)} \quad (2.9)$$

จะได้สมการ

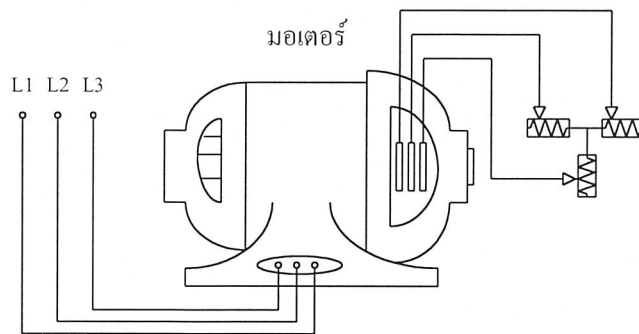
$$P_o(mt) = P_dR(mt) - P_{mech} \quad (2.10)$$

และสมการ

$$P_dR(mt) = P_iR(mi) (1 - S_{mt}) \quad (2.11)$$

เทคนิคในการเริ่มต้นหมุน (Induction Motor Starting Technique) ในการอธิบายเกี่ยวกับเทคนิคการเริ่มต้นหมุนจะกล่าวถึงมอเตอร์ที่มีสลิบริงและไม่มีสลิบริงแยกกันเพื่อให้เข้าใจได้ง่ายขึ้น

1. เทคนิคในการเริ่มต้นหมุนมอเตอร์เหนี่ยวนำที่มีสลิบริงทำได้โดยต่อความต้านทานที่ปรับค่าได้จากภายนอกอนุกรมกับขดลวดที่โรเตอร์ (โดยผ่านสลิบริง) ทั้ง 3 เฟส แล้วลัดวงจรที่จุดต่อรวม ดังรูปที่ 2.20



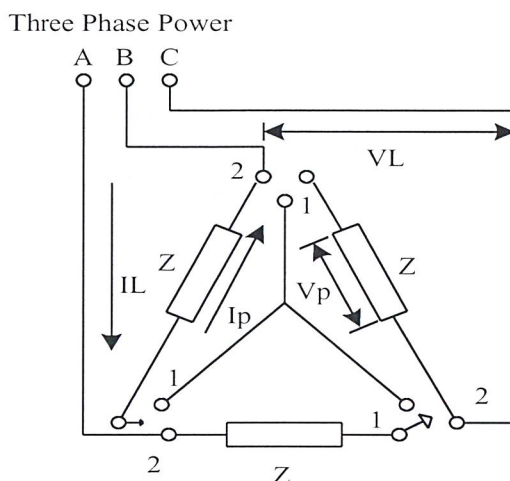
รูปที่ 2.20 เทคนิคในการเริ่มต้นหมุนมอเตอร์เหนี่ยวนำที่มีสลิบริง

เมื่อจ่ายแรงดันไฟฟ้าที่พิกัดให้ขดลวดที่สเตเตอร์กระแสไฟฟ้าเริ่มต้นหมุนจะถูกจำกัดให้มีค่าต่ำลงเนื่องจากความต้านทานดังกล่าวถ้าปรับค่าความต้านทานลดลงเรื่อยๆ ความเร็วของมอเตอร์จะค่อยๆ เพิ่มขึ้นและเมื่อปรับความต้านทานให้มีค่าเท่ากับศูนย์มอเตอร์จะหมุนด้วยความเร็วเต็มพิกัด

2. เทคนิคในการเริ่มต้นหมุนของมอเตอร์เหนี่ยวนำที่ไม่มีสลิบริงทำได้ 3 วิธี คือ วิธีการต่อมอเตอร์โดยตรงกับสายวิธีการควบคุมแรงดันไฟฟ้าขณะเริ่มต้นหมุน และวิธีการต่อขดลวดแบบสตาร์-เดลต้า

เทคนิคการต่อมอเตอร์โดยตรงกับสาย (Across The Line Starting) เทคนิคนี้จะมีปัญหาเกี่ยวกับค่ากระแสไฟฟ้าเริ่มต้นหมุนเพราะมอเตอร์ขณะเริ่มต้นหมุนจะใช้กระแสไฟฟ้า 5-10 เท่าของกระแสไฟฟ้าขณะขับโหลดเต็มที่ เช่น มอเตอร์มีขนาด 220 โวลต์ และกระแสไฟฟ้าขณะขับโหลดเต็มที่ 10 แอมแปร์ต้องใช้กระแสไฟฟ้าเริ่มต้นหมุน 250-500 แอมแปร์ทันทีลักษณะเช่นนี้จะทำให้เกิดปัญหาต่างๆเช่นการมีคสลับชั่วขณะของแสงสว่างและการตัดอัตโนมัติของอุปกรณ์ป้องกันด้วยเหตุนี้จึงใช้กับมอเตอร์ขนาดเล็กที่มีพิกัดไม่เกิน 5 แรงม้า หรือ 3.5 กิโลวัตต์เท่านั้น

วิธีการควบคุมแรงดันไฟฟ้าขณะเริ่มต้นหมุนจะเป็นวิธีที่ง่ายที่สุดซึ่งใช้ในห้องทดลอง คือการใช้ Variac 3 เฟส ปรับค่าแรงดันไฟฟ้าหรือการใช้ Auto-Transformer 3 เฟส ที่ปรับค่าได้ด้วยมือ



รูปที่ 2.21 การต่อขดลวดแบบ สตาร์-เดลต้า

วิธีการต่อขดลวดแบบ Y - Δ (Wye - Delta Starting) มีวิธีการคือขณะเริ่มต้นหมุนขดลวดที่สเตเตอร์จะต่อแบบ Y เมื่อความเร็วของมอเตอร์เป็น 75 % ของความเร็วซิงโครนัสจึงเปลี่ยนมาต่อแบบ Δ ด้วยสวิตช์อยู่ที่รีเลย์ (ดังรูปที่ 2.21 เมื่อสวิตช์อยู่ที่ตำแหน่ง 1 ขดลวดที่สเตเตอร์จะต่อแบบ Y และเมื่อสวิตช์อยู่ที่ตำแหน่ง 2 ขดลวดที่สเตเตอร์จะต่อแบบ Δ) วิธีนี้เป็นวิธีที่ใช้ในทางปฏิบัติจริง

ขณะเริ่มต้นหมุนเหตุผลที่ต่อขดลวดแบบ Y ก็เพราะทำให้กระแสไฟฟ้าเริ่มต้นหมุนมีค่าต่ำกว่าการต่อขดลวดแบบ Δ

การพิสูจน์ว่าเมื่อต่อขดลวดที่สเตเตอร์แบบ Y กระแสไฟฟ้าขณะเริ่มต้นหมุนต่ำกว่าการต่อแบบ Δ ทำได้โดยการพิจารณาเมื่อ

$$V_L = \text{แรงดันไฟฟ้าที่สาย}$$

$$V_P = \text{แรงดันไฟฟ้าที่เฟส}$$

$$Z = \text{อิมพีแดนซ์ต่อเฟส}$$

$$I_L = \text{กระแสไฟฟ้าที่สาย}$$

$$I_P = \text{กระแสไฟฟ้าที่เฟส}$$

การเริ่มต้นหมุนที่ต่อขดลวดแบบ Δ จะได้จากสมการ

$$I_P = \frac{V_L}{Z} \quad (2.12)$$

$$I_{L\Delta} = \sqrt{3}I_P = \frac{\sqrt{3}V_L}{Z} \quad (2.13)$$

การเริ่มต้นหมุนที่ต่อขดลวดแบบ Y จะได้สมการ

$$V_P = \frac{V_L}{\sqrt{3}} \quad (2.14)$$

$$I_P = \frac{V_P}{Z} = \frac{V_L}{\sqrt{3}Z} \quad (2.15)$$

$$I_{LY} = I_P = \frac{V_L}{\sqrt{3}Z} \quad (2.16)$$

หากกระแสไฟฟ้าเริ่มต้นหมุนขณะที่ต่อขดลวดแบบ Y ด้วยกระแสไฟฟ้าเริ่มต้นหมุนขณะ
ที่ต่อขดลวดแบบ Δ จะได้สมการ

$$\frac{I_{LY}}{I_{L\Delta}} = \frac{V_L/(\sqrt{3}Z)}{\sqrt{3}V_L/Z} \quad (2.17)$$

$$= \frac{1}{\sqrt{3}\sqrt{3}} = \frac{1}{3} \quad (2.18)$$

$$I_{LY} = \frac{1}{3} I_{L\Delta} \quad (2.19)$$

จะเห็นได้ว่ากระแสไฟฟ้าเริ่มต้นหมุนของการต่อขดลวดแบบ Y เป็น $1/3$ ของการต่อขด
ลวดแบบ Δ

แต่อย่างไรก็ตามการต่อขดลวดแบบ Y จะมีแรงบิดเริ่มต้นหมุนต่ำกว่าแรงบิดที่พิกัด 1/3 เท่า ทั้งนี้เพราะแรงบิดเริ่มต้นหมุนจะเป็นสัดส่วนกับแรงดันไฟฟ้าที่สายยกกำลังสองนอกจากนี้การต่อขดลวดแบบ Y นั้นมีแรงดันไฟฟ้าตกคร่อมขดลวดที่สเตเตอร์เท่ากับ $V_L / 3$

2.7 การมอดูเลตสัญญาณตามความกว้างของพัลส์

2.7.1 หลักการมอดูเลตสัญญาณตามความกว้างของพัลส์

เทคนิคในการสร้างสัญญาณ Pulse Width Modulation (PWM) เพื่อใช้ในการควบคุมอุปกรณ์สวิตช์กำลังในวงจรอินเวอร์เตอร์เพื่อให้ได้แรงดันและความถี่ตามต้องการที่นิยมใช้กันทั่วไปมี 2 วิธีคือ

1.) เทคนิค PWM แบบฮาร์โมนิกย่อย (Sub Harmonic PWM.Technique)

วิธีการนี้เรียกอีกชื่อหนึ่งว่า “หลักการ PWM แบบสัญญาณไซน์ (Sinusoidal PWM)” เป็นสัญญาณควบคุมที่ได้จากการสุ่มระหว่างสัญญาณอ้างอิงรูปไซน์กับสัญญาณรูปคลื่นสามเหลี่ยมดังรูปที่ 2.22

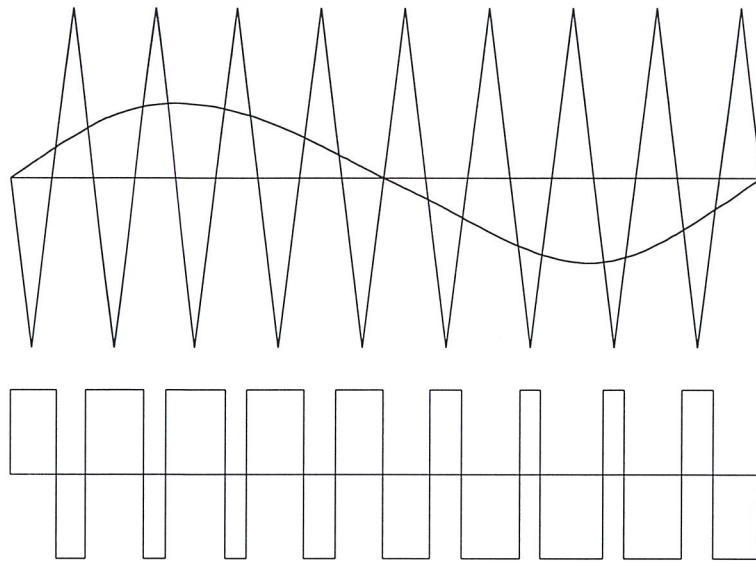
จากรูปที่ 2.22 ความถี่การสวิตช์สัญญาณ PWM จะเท่ากับความถี่ของสัญญาณคลื่นพาห์รูปคลื่นสามเหลี่ยมทั้งนี้เราสามารถคำนวณหาค่ามุมสวิตช์ (Switching Angle) ที่เกิดขึ้นใน 1/4 วัฏจักรของความถี่เอาต์พุตได้ตามสูตร

$$A \sin(\alpha_1) = (-1) \left[2i - \frac{2P\alpha}{\pi} \right] \quad (2.20)$$

โดย A = ตัวประกอบความสูงระหว่างค่าสูงสุดสัญญาณอ้างอิงกับสัญญาณคลื่นพาห์

P = อัตราส่วนระหว่างความถี่สัญญาณพาห์กับความถี่สัญญาณอ้างอิง

α_1 = มุมสวิตช์ใดๆ ในช่วง 1/4 วัฏจักรโดย $1 < 1 < (p-1)/2$



P=9 M=0.5

รูปที่ 2.22 หลักการสร้างสัญญาณ PWM ตามวิธีฮาร์โมนิกย่อย

อย่างไรก็ตาม ในกรณีที่ต้องการออกแบบวงจรควบคุมแบบประยุกต์ใช้งานไมโครโปรเซสเซอร์เราสามารถคำนวณค่ามุมสวิทช์สำหรับทุกความถี่เอาต์พุตเพื่อเก็บไว้ในหน่วยความจำของวงจรได้เช่นกัน

2.) เทคนิค PWM แบบกำจัดฮาร์โมนิก (Harmonic Elimination Technique)

วิธีการนี้ใช้สำหรับวงจรควบคุมแบบดิจิทัล หรือ ควบคุมด้วยไมโครโปรเซสเซอร์เพียงเท่านั้น โดยนำค่ามุมที่คำนวณได้ตามหลักการนี้สำหรับทุกความถี่ขาออกเพื่อเก็บไว้ในหน่วยความจำของวงจรหลักการคำนวณมาจากทฤษฎีฟูรีเออร์ตามรูปคลื่น PWM

2.7.2 เทคนิคการมอดูเลตสัญญาณตามความกว้างของพัลส์แบบเทียบพื้นที่ใต้เส้นโค้ง

ไซน์

จากหลักการสร้างสัญญาณ PWM ทั้งสองหลักการที่กล่าวมาแล้วนั้นจะเห็นว่าข้อดี ข้อเสียแตกต่างกัน เช่น แบบฮาร์โมนิกย่อยสามารถคำนวณหามุมได้ง่ายแต่วงจรควบคุมมักเป็นแบบอนาลอกที่ยุ่งยากในการเปลี่ยนแปลง หรือ เพิ่มฟังก์ชันในวงจรควบคุมนั้นๆ ส่วนวิธีกำจัดฮาร์โมนิกสามารถลดฮาร์โมนิกในระบบลงได้อย่างมากอีกทั้งวงจรควบคุมจะต้องทำงานเป็นวงจรดิจิทัลหรือไมโครโปรเซสเซอร์ซึ่งสะดวกต่อการเปลี่ยนแปลงเพิ่มเติมฟังก์ชันในวงจรควบคุมแต่ก็ยุ่งยากในการคำนวณมากไปและเช่นเดียวกันขณะที่ความถี่สูงขึ้นจำนวนฮาร์โมนิกที่สามารถกำจัดได้ก็จะน้อยลงเรื่อยๆ ไม่เป็นไปตามต้องการที่ทุกความถี่เอาต์พุต

สำหรับเทคนิค PWM แบบเทียบพื้นที่ได้เส้นโค้งนี้ยังคงใช้วิธีคำนวณมุมสวิทช์ เพื่อนำไปเก็บในหน่วยความจำของวงจรควบคุมซึ่งเป็นแบบดิจิทัลหรือไมโครโปรเซสเซอร์ เช่นกันทั้งนี้จะใช้หลักการแบ่งฐานเวลาของสัญญาณที่ความถี่หลักมูลขาออกใน 1 วัฏจักรออกเป็นส่วนย่อยๆ ช่วงละเท่าๆ กัน จำนวน N_{st} ส่วนจำนวนส่วนย่อยนี้จะต้องเป็นเลขคู่และเป็น ตัวคูณของ 3 (ในกรณีอินเวอร์เตอร์ 3 เฟส) ซึ่งจะมีค่าน้อยเท่าไรก็ได้ ซึ่งเมื่อแบ่งส่วนย่อยมากขึ้นเท่าใดกระแสของมอเตอร์ก็จะใกล้เคียงรูปคลื่นไซน์มากขึ้นเท่านั้น

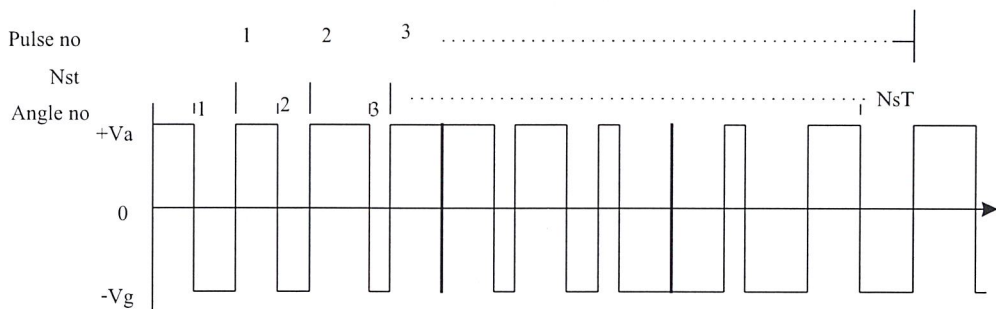
2.7.3 ข้อกำหนดทางเทคนิคการมอดูเลตสัญญาณตามความกว้างของพัลส์แบบเทียบพื้นที่ได้เส้นโค้งไซน์

การสร้างสัญญาณแบบ PWM เพื่อควบคุมการสวิทช์ของอุปกรณ์สวิทช์กำลังในวงจรอินเวอร์เตอร์สามารถทำได้ทั้งแบบแอนะล็อกและแบบดิจิทัลซึ่งแบบดิจิทัลมีข้อได้เปรียบกว่าแบบอนาล็อกอยู่หลายประการ เช่น ปรับแต่งค่ามุมสวิทช์ได้ตามต้องการปรับปรุงระบบควบคุมวงจรได้ง่ายปรับปรุงเสถียรภาพของการทำการควบคุมได้ เป็นต้น ในปัจจุบันใช้วงจรของการสร้างสัญญาณ PWM คือ ต้องสร้างสัญญาณ PWM ที่มีความถี่การสวิทช์ (Switching Frequency : f_s) คงที่ ความกว้างระหว่างมุมสวิทช์, ความสูงขององค์ประกอบหลักมูลจะแปรเปลี่ยนไปตามค่าความถี่หลักมูลที่ต้องการ ทั้งนี้อัตราโวลต์ต่อเอิร์ตซ์ จะต้องมีค่าคงที่ตามที่กำหนดจำนวนมุมสวิทช์ในครึ่งไซเคิลจะเท่ากับอัตราส่วน ความถี่มอดูเลต (Frequency Modulation Ratio : p) อีกทั้งฮาร์โมนิกลำดับต่างๆ ที่อาจเป็นผลเสียต่อแรงบิด และความเร็วของมอเตอร์ ควรมีค่าเป็นศูนย์ทั้งนี้ฮาร์โมนิกที่เกิดขึ้นมักมีความถี่ไม่ต่ำกว่าค่าความถี่การสวิทช์ที่เลือกใช้

อย่างไรก็ตามเพื่อให้สามารถรองความถี่ฮาร์โมนิกที่มีผลกระทบต่อมอเตอร์ และระบบไฟฟ้าออกโดยง่าย จึงมักใช้ความถี่การสวิทช์ค่อนข้างสูงสำหรับอินเวอร์เตอร์ขนาดกำลังต่ำ สำหรับอินเวอร์เตอร์ขนาดกำลังงานสูง ทั้งนี้เพราะความถี่การสวิทช์สูงสุดที่สามารถเลือกใช้ ขึ้นอยู่กับช่วงเวลา Turn On และ Turn Off ของอุปกรณ์สวิทช์กำลังในอินเวอร์เตอร์เช่นกัน

วงจร PWM ที่ได้ออกแบบสร้างใช้หลักการออกแบบเทียบเคียงพื้นที่ได้เส้นโค้งไซน์ดังได้กล่าวถึงหลักการ และผลการจำลองกระแสแล้ว ทั้งนี้โดยการคำนวณหาค่ามุมสวิทช์และตัวนับค่ามุม เมื่อได้ค่าตัวนับของมุมสวิทช์ทุกมุมของทุกความถี่ขาออกแล้ว จึงนำมาสร้างเป็น “แพทเทิร์นการสวิทช์” เก็บไว้ในส่วนหน่วยความจำอีพรอม (EPROM) ของวงจร เพื่อให้โปรแกรมหลักสามารถเรียกใช้ได้ขณะสร้างสัญญาณที่ความถี่หลักมูลใดๆ วงจร PWM นี้สามารถควบคุมความถี่หลักมูลได้ตั้งแต่ 0 ถึง 60 เฮิร์ตซ์ โดยปรับได้ขึ้นละ 1 เฮิร์ตซ์ โดยใช้ความถี่การสวิทช์ค่อนข้างคงที่ที่ 1,440 เฮิร์ตซ์โดยประมาณ ทำให้สามารถกำจัดความถี่ที่เป็นความถี่ฮาร์โมนิก ที่มีค่าต่ำกว่า 1.4 กิโลเฮิร์ตซ์ได้

สัญญาณ PWM โดยใช้หลักการเทียบเคียงพื้นที่ใต้โค้งไซน์นี้ จะมีการแบ่งฐานเวลาออกเป็นส่วนย่อยๆ ช่วงละเท่าๆ กันออกเป็น N_{st} ส่วน มุมสวิทช์ต่างๆ ที่จะต้องคำนวณหาเพื่อเก็บไว้เป็น“แพทเทิร์นการสวิทช์” เก็บไว้ในส่วนหน่วยความจำ คือมุมที่ทำให้วงรอบภาระ (Duty cycle) ของแต่ละส่วนย่อยๆ นั้น มีค่าเทียบเท่าขนาดของสัญญาณไซน์ ณ ค่ามุนั้นๆ ซึ่งจะเห็นที่ได้ว่ามุมสวิทช์ที่จะคำนวณหา มีจำนวน N_{st} มุมเช่นกัน ดังรูป 2.23

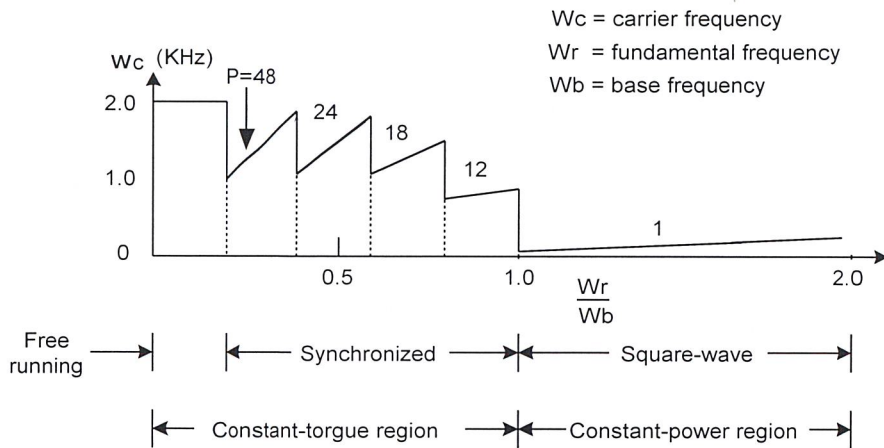


รูปที่ 2.23 สัญญาณ PWM ตามหลักการเทียบพื้นที่ใต้โค้งไซน์

2.8 การเปลี่ยนจำนวนพัลส์ PWM ที่สัมพันธ์กับความถี่

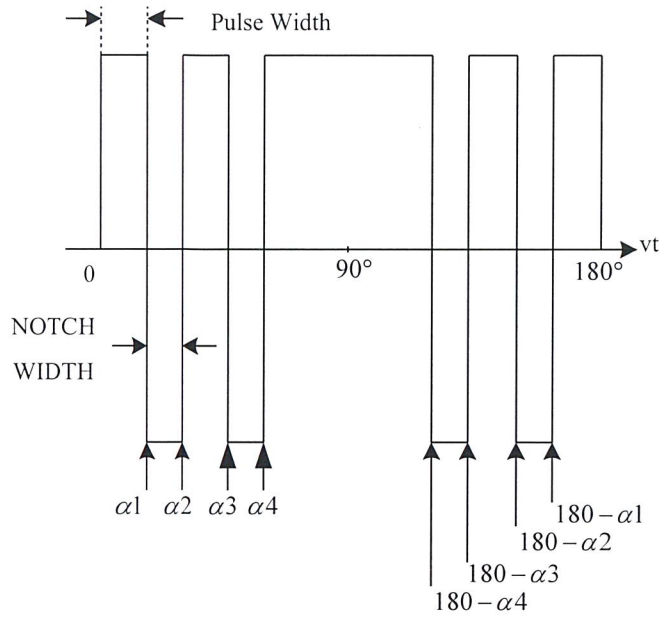
สำหรับในงานการขับเคลื่อนความเร็วรอบมอเตอร์นั้น แรงดันและความถี่จะต้องเปลี่ยนแปลงตามรูปที่ 2.24 โดยในย่านกำลังคงที่แรงดันจะต้องถูกควบคุมโดยหลักการของ PWM เพื่อให้พัลส์ใน Airgap มีค่าคงที่ เพื่อลดฮาร์โมนิกส์ในแรงดันและกระแสของเอาต์พุตอินเวอร์เตอร์และลดการสูญเสียของการสวิทช์ซึ่งในอุปกรณ์สวิทช์จะมีการควบคุมพัลส์ PWM โดยวิธีการเปลี่ยนเกียร์ (Gear-Changing) แบบซิงโครนัสทำให้จำนวนพัลส์ PWM ต่อไซเคิล เปลี่ยนอย่างทันทีทันใด ซึ่งในช่วงความถี่ของอินเวอร์เตอร์ต่ำจะกำหนดให้จำนวนพัลส์ PWM สูงในช่วงความถี่อินเวอร์เตอร์สูงจะทำให้จำนวนพัลส์ PWM ต่ำลง ก็จะเป็นการจำกัดความถี่ของการสวิทช์ซึ่งไม่ให้เกินขอบเขตที่กำหนดไว้ จากรูปที่ 2.24 ที่ความถี่มูลฐานต่ำ ความถี่ของคลื่นหาหะถูกรักษาให้มีค่าคงที่การทำงานของ อินเวอร์เตอร์จะอยู่ใน โหมด ฟรีรันนิ่ง หรือ อะซิงโครนัส ซึ่งจะทำให้ เฟสมีการเลื่อนไปมาอย่างต่อเนื่อง ทำให้เกิดปัญหา Subharmonic แต่อย่างไรก็ตามผลอันตรายเหล่านี้สามารถไม่คำนึงถึงเพราะอัตราส่วนของความถี่มีค่ามาก เมื่อความถี่มูลฐานมีค่าสูงขึ้น ค่า P จะเปลี่ยนค่าเป็นสเต็ปแบบซิงโครนัสมีค่าลดลงเพื่อที่จะใช้ความถี่ของคลื่นพาหะที่มีค่าสูงสุดและต่ำสุด ยังอยู่ในขอบเขตที่กำหนด จนกระทั่งใกล้ความถี่กำหนดจะมีการเปลี่ยนจากโหมดซิงโครนัส PWM ไปสู่โหมด Square Wave ซึ่งในขณะนี้ความถี่ของคลื่นพาหะจะมีค่าเท่ากับความถี่มูลฐาน ดังนั้นการควบคุม

คุมพัลส์จึงควรออกแบบให้เหมาะสม เพื่อไม่ให้เกิดการเปลี่ยนค่า P และนอกจากนี้ควรหลีกเลี่ยงปัญหาการเปลี่ยนกลับไปกลับมาของค่า P ที่อยู่ติดกันโดยการสร้างแถบ ฮิสเทอรีซิส (Hysteresis Band) ณ จุดวิกฤตของการเปลี่ยนค่า P



รูปที่ 2.24 ความสัมพันธ์ระหว่างความถี่มูลฐานและความถี่สวิทช์ของอุปกรณ์สวิทช์ซึ่งเทคนิคการกำจัดฮาร์โมนิกส์

เทคนิคนี้สามารถควบคุมแรงดันมูลฐานและกำจัดองค์ประกอบฮาร์โมนิกส์ที่ไม่ต้องการออกไป หลักการของวิธีการนี้ จะทำการคำนวณหามุมของการสวิทช์โดยใช้สมการการกระจายอนุกรมฟูเรียร์แบบทั่วไป ตำแหน่งมุมและสถานะสวิทช์ซึ่งจะถูกเก็บในรูปแบบ Look-up Table ไว้ในหน่วยความจำ โดยมีไมโครโปรเซสเซอร์และ Down Counter ทำงานร่วมกันเพื่อสร้างสัญญาณ PWM โดยดึงข้อมูลที่เก็บไว้ในหน่วยความจำออกมา



รูปที่ 2.25 สัญญาณ PWM โดยวิธีกำจัดฮาร์โมนิกส์

จากรูปที่ 2.25 ลักษณะคลื่นจะสมมาตร 1/4 ของคาบและสมการอนุกรมฟูเรียร์ของฟังก์ชันคือ

$$f(t) = \sum_{n=1}^{\infty} (a_n \cos n\theta + b_n \sin n\theta) \tag{2.21}$$

ซึ่ง

$$a_n = \frac{1}{\pi} \int_0^{2\pi} f(t) \cos n\theta d\theta \tag{2.22}$$

$$b_n = \frac{1}{\pi} \int_0^{2\pi} f(t) \sin n\theta d\theta \tag{2.23}$$

เนื่องจากลักษณะแรงดันรูปคลื่น PWM ตามรูป 2.25 ดังนั้นจะมีแต่ฮาร์โมนิกส์ที่เท่านั้นจะ
ได้ว่า

$$a_n = 0 \tag{2.24}$$

$$b_n = \frac{4}{\pi} \int_0^{\pi/2} f(t) \sin n\theta d\theta \quad (2.25)$$

สมมติให้แรงดันรูปคลื่น PWM มีแอมพลิจูด $f(t) = 1$
 ดังนั้น

$$b_n = \frac{4}{\pi} [1 + 2(-\cos n\alpha_1 + \cos n\alpha_2 - \dots + \cos n\alpha_n)] \quad (2.26)$$

$$= \frac{4}{n\pi} \left(1 + 2 \sum_{k=1}^k (-1)^k \cos n\alpha_k \right) \quad (2.27)$$

โดยที่ สมการ (1) จะประกอบด้วยตัวแปร k ตัว ซึ่งจะมีสมการ k สมการเพื่อที่จะแก้สมการ มุมจำนวน k ตัว ซึ่งแรงดันมูลฐานสามารถถูกควบคุมได้และจะมีฮาร์โมนิกส์ที่ถูกกำจัด $k-1$ ฮาร์โมนิกส์ อย่างไรก็ตาม ฮาร์โมนิกส์ อันดับสามและอันดับฮาร์โมนิกส์อันดับที่มีผลคูณของ 3 จะถูกจำกัดไปเมื่อใช้กับไปสามเฟส เช่น สมมติให้ $k=3$ ดังนั้นสามารถเขียนสมการได้ 3 สมการดังนี้คือ

องค์ประกอบมูลฐาน

$$b_1 = \frac{4}{\pi} (1 - 2 \cos \alpha_1 + 2 \cos \alpha_2 - 2 \cos \alpha_3) \quad (2.28)$$

องค์ประกอบฮาร์โมนิกส์อันดับ 5

$$b_5 = \frac{4}{5\pi} (1 - 2 \cos 5\alpha_1 + 2 \cos 5\alpha_2 - 2 \cos 5\alpha_3) = 0 \quad (2.29)$$

องค์ประกอบฮาร์โมนิกส์อันดับ 7

$$b_7 = \frac{4}{7\pi} (1 - 2 \cos 7\alpha_1 + 2 \cos 7\alpha_2 - 2 \cos 7\alpha_3) = 0 \quad (2.30)$$

ซึ่งสมการ 3 สมการข้างบนนี้ สามารถใช้วิธีการทางนิวเมอริคอลโดยใช้วิธีนิวตันราปสันหลายตัวแปรเพื่อแก้สมการ $\alpha_1, \alpha_2, \alpha_3$ สำหรับองค์ประกอบมูลฐานที่ต้องการ

ข้อเสียของวิธีการนี้ คือถ้าต้องการกำจัดฮาร์โมนิกส์หลายๆ อันดับ จะทำให้มีการเก็บและตำแหน่งมูม และสภาวะมูมสวิตซิ่งในหน่วยความจำเป็นจำนวนมากทำให้ไม่สะดวก ดังนั้นจึงมีการใช้เทคนิค ผสมของเทคนิค PWM (Hybrid PWM) ซึ่งเป็นเทคนิคอย่างหนึ่งที่น่าสนใจมากโดยเมื่ออินเวอร์เตอร์ทำงานในย่านความถี่และแรงดันต่ำจะใช้วิธี SPWM ในขณะที่ย่านความถี่แรงดันสูงจะใช้วิธีกำจัดฮาร์โมนิกส์ ทำให้ไม่มีปัญหาเกี่ยวกับการสูญเสียเนื่องจากฮาร์โมนิกส์และสวิตซิ่ง วิธีกำจัดฮาร์โมนิกส์นี้ยังสามารถใช้ในย่าน Constant Power ได้อีกด้วย

2.8.1 เทคนิค Minimum Ripple Current

ข้อเสียของเทคนิคการกำจัดฮาร์โมนิกส์อีกอย่างหนึ่งก็คือ จะทำให้ฮาร์โมนิกส์อันดับต่ำที่ไม่ต้องการกำจัดมีค่าสูงมาก เพราะการสูญเสียในแมชชีนเนื่องจากผลฮาร์โมนิกส์ถูกกำหนดโดยอาร์เอ็มเอส ของกระแสรีปเปิล (RMS Ripple Current) ดังนั้นกระแสรีปเปิลจึงเป็นพารามิเตอร์ที่ควรพิจารณาให้มีค่าน้อยที่สุดแทนที่จะไปกำจัดฮาร์โมนิกส์ที่ต้องการออกไปเทคนิคนี้จะสมมติให้ไม่คิดผลของ Skin Effect ของพารามิเตอร์แมชชีนซึ่งอาจจะเป็นสิ่งที่ไม่ถูกต้องมากนัก โดยเฉพาะอย่างยิ่งแล้วสำหรับขดโรเตอร์ สมการของ RMS Ripple Current ถูกกำหนดโดย

$$I_{\text{ripple}} = \sqrt{I_3^2 + I_5^2 + I_7^2 + \dots} \quad (2.31)$$

$$= \sqrt{\frac{I_{3m}^2}{2} + \frac{I_{5m}^2}{2} + \frac{I_{7m}^2}{2} + \frac{I_{9m}^2}{2} + \dots} \quad (2.32)$$

$$= \sqrt{\frac{1}{2} \sum_{n=3}^{\infty} \left[\frac{v_n}{n\omega_a L} \right]^2} \quad (2.33)$$

ซึ่ง I_3, I_5, \dots กระแส rms ของฮาร์โมนิกส์อันดับ 3, 5 และอื่นๆ

I_{3m}, I_{5m}, \dots etc = ค่าพิคพลาคของกระแสฮาร์โมนิกส์

N = อันดับฮาร์โมนิกส์

V_n = ค่าพิคขององค์ประกอบฮาร์โมนิกส์ อันดับที่ n

I = ลีคเกจ อินดักแตนซ์ ของแมชชีน ต่อ เฟส

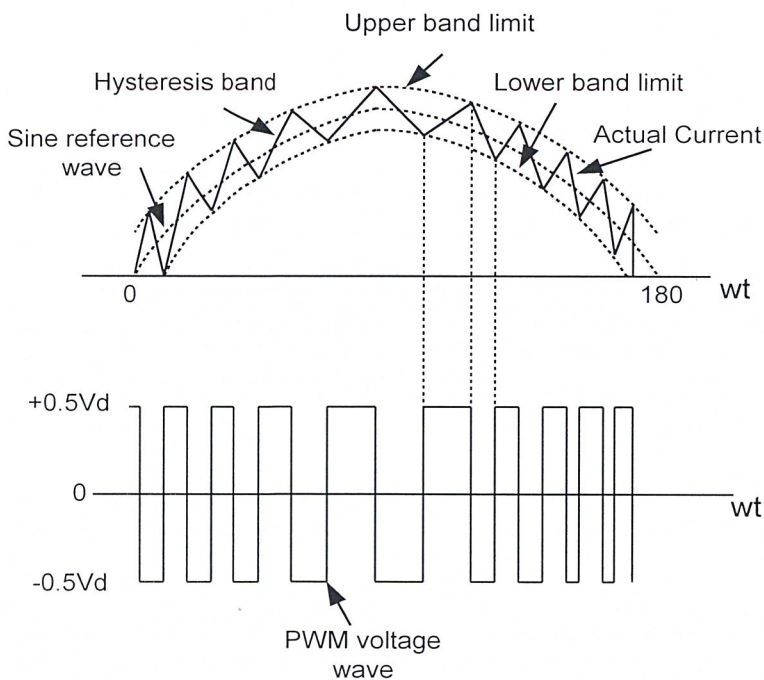
ω_a = ความถี่เชิงมุมมูลฐาน

ดังนั้น กำลังการสูญเสียของขดลวดเนื่องจากผลของฮาร์โมนิกส์ คือ $P_L = 3I_{ripple}^2 R$ ซึ่ง R คือ Effective Resistance ของแมชชีนต่อเฟส

จากสมการที่ 2.33 ซึ่งเป็นการหาดำแหน่งมุมแทนลงในสมการที่ 2.38 ก็จะพบว่า I_{ripple} เป็น ฟังก์ชันของตำแหน่งมุม โดยการใช้โปรแกรมทางคอมพิวเตอร์ก็สามารถแก้ α ตำแหน่งมุมที่ทำให้เกิด I_{ripple} น้อยที่สุดได้ ดังนั้นเทคนิคการพิจารณาให้ Harmonic Loss มีค่าน้อยที่สุดจะเป็นสิ่งที่ต้องการมากกว่าเทคนิคการกำจัดฮาร์โมนิกส์

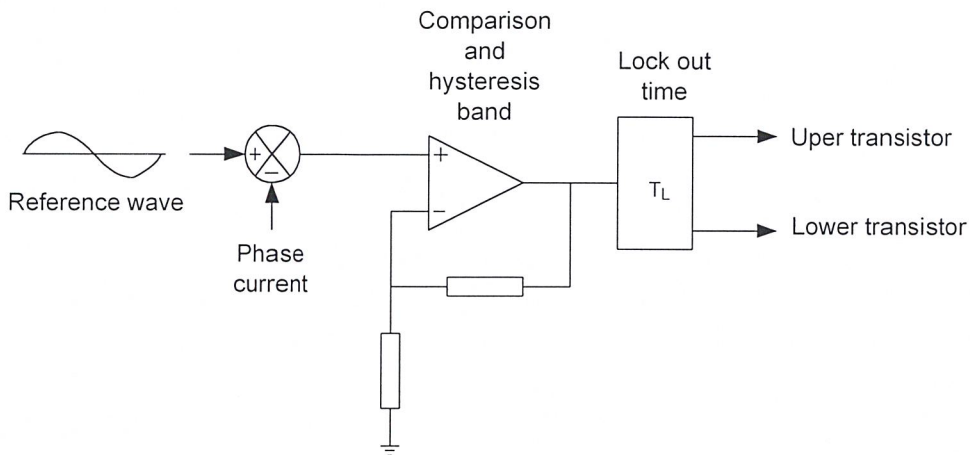
2.8.2 วิธีการ Adaptive Current Control PWM

จากการวิเคราะห์ลักษณะฮาร์โมนิกส์ของรูปคลื่นแรงดันเอาต์พุตของ VSI ที่ได้กล่าวแล้ว นั้นได้สมมติให้ แรงดัน ดีซี-ลิงก์ มีค่าในทางอุดมคติจะมีค่าคงที่โดยไม่เกิดการริบเบิล แต่ในทางปฏิบัติ ริบเบิลจะเกิดขึ้นได้เนื่องจากค่าของ LC หรือ C Filter มีค่าไม่มากพอ ดังนั้น เทคนิค PWM ที่สามารถควบคุมกระแสโดยขอบเขตของฮิสเตอร์รีซิส หรือมีการปรับตัวเองโดยอัตโนมัติก็จะแก้ปัญหานี้ได้เป็นอย่างดี เทคนิคนี้จะใช้การควบคุมกระแส ลักษณะการเกิดตามรูปที่ 2.26



รูปที่ 2.26 หลักการเกิดรูปคลื่น PWM โดยเทคนิค Adaptive Current Control

โดยวงจรควบคุมจะสร้างกระแสไซน์อ้างอิงมีขนาดและความถี่ตามต้องการซึ่งจะนำมาเปรียบเทียบกับกระแสเฟสจริงตามรูป 2.27

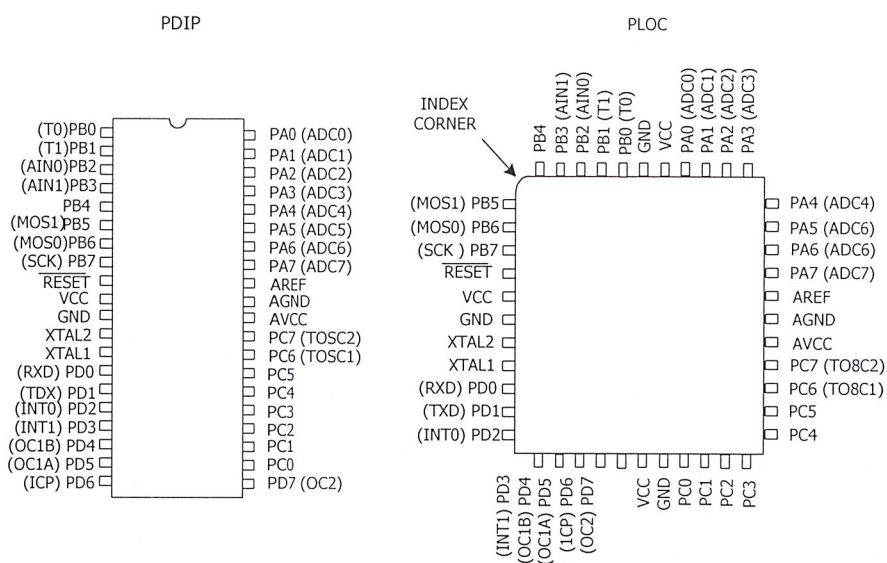


รูปที่ 2.27 วงจรควบคุมเทคนิค Adaptive Current Control

เมื่อกระแสเกินค่าขอบเขตของฮิสเตอร์รีซิส ที่กำหนดไว้ ทรานซิสเตอร์ตัวบนในกิ่งของวงจรบริดจ์อินเวอร์เตอร์จะหยุดนำกระแส (Turn-Off) ในขณะที่ทรานซิสเตอร์ตัวล่างจะนำกระแส (Turn On) เป็นผลให้แรงดันเอาท์พุทเปลี่ยนจาก $+0.5V_d$ เป็น $-0.5V_d$ และกระแสเริ่มลดลงจนกระทั่งกระแสลดลงมาถึงขอบล่างของฮิสเตอร์รีซิส จะมีการเปลี่ยนสวิตช์ โดยทรานซิสเตอร์ตัวบนจะนำกระแส และทรานซิสเตอร์ตัวล่างจะหยุดนำกระแส ซึ่งเป็นการครอบรอบของการสวิตช์ในรูปที่ 2.27 T_L จะเป็นเวลาเดดไทม์ของวงจรลอคเอาท์ เพื่อป้องกันไม่ให้เกิดการฟลลท์เนื่องจาก Shoot-Through ของทรานซิสเตอร์ตัวบนและตัวล่าง กระแสจริงจะถูกบังคับให้ติดตามสัญญาณไซน์อ้างอิงให้อยู่ในขอบเขตฮิสเตอร์รีซิสที่ต้องการ โดยการสวิตช์กลับไปกลับมาของทรานซิสเตอร์ตัวบนและตัวล่าง ดังนั้นกระแสริบเบิล พิก-ทู-พิก จึงถูกควบคุมให้ปรับตัวอยู่ในขอบเขตฮิสเตอร์รีซิสตามค่า V_d ที่มีการกระเพื่อม ดังนั้นกระแส Ripple RMS ซึ่งจะสัมพันธ์กับกระแส Ripple พิก-ทู-พิก ซึ่งจะถูกควบคุมทำให้ลดการ Heating ของแมชชีน การควบคุมกระแส พิกอย่างอินสแตนเตเนียส (Instantaneous) จะเป็นข้อดีสำหรับอุปกรณ์สวิตช์ซึ่งอย่างเช่นเพาเวอร์ทรานซิสเตอร์ซึ่งจะไวต่อกระแสพิกมาก โหมดการควบคุมกระแสสามารถเปลี่ยนจากโหมด PWM ไปสู่โหมดแรงดันสแควร์เวฟ ซึ่งอยู่ในย่านกำลังคงที่ ได้อย่างต่อเนื่อง

2.9 ไมโครคอนโทรลเลอร์เบอร์ AT90S8535

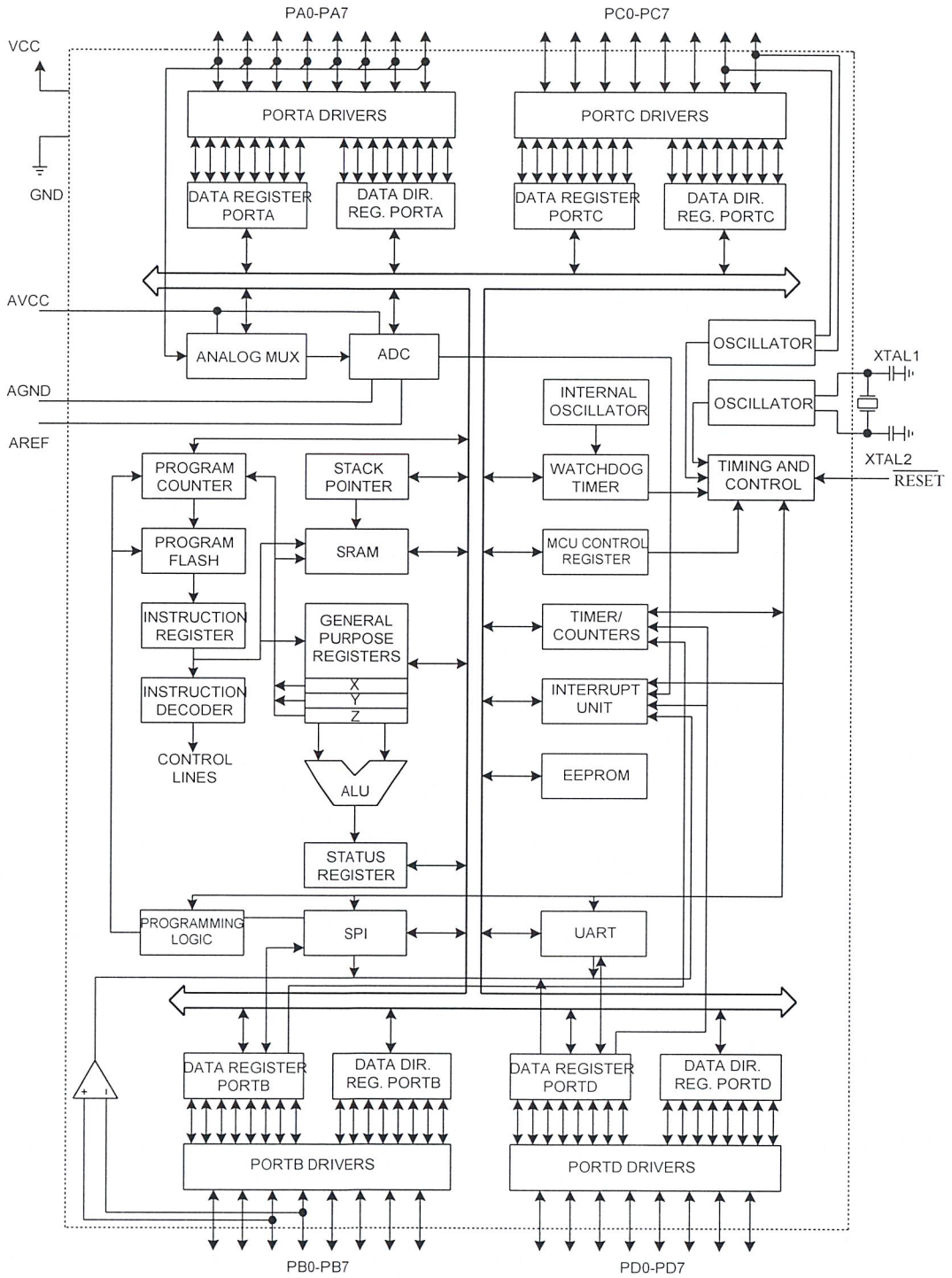
AT90S8535 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ที่มีสถาปัตยกรรมแบบ RISC (Reduce Instruction Set Computer) ซึ่งทำให้การประมวลผลมีความเร็ว 1 คำสั่ง/1 Clock หรือ CPU สามารถประมวลคำสั่งได้ 1 MIPS / MHz ภายในประกอบด้วยรีจิสเตอร์ใช้งานทั่วไปขนาด 8 บิต จำนวน 32 ตัว ซึ่งแต่ละตัวจะต่อเข้ากับ ALU โดยตรง ทำให้การประมวลต่อ 1 คำสั่งมีความเร็วกว่า CPU ที่มีสถาปัตยกรรมแบบ CISC ดังรูปที่ 2.28



รูปที่ 2.28 โครงสร้างภายนอกและตำแหน่งขา

2.9.1 โครงสร้างภายใน

AT90S8535 จะมีหน่วยความจำสำหรับ Program Memory แบบ Flash ขนาด 4 KByte หน่วยความจำสำหรับ Data Memory แบบ EEPROM ขนาด 512 Byte และหน่วยความจำแบบ RAM ขนาด 512 Byte มีพอร์ตที่สามารถทำงานได้ 2 ทิศทาง จำนวน 32 เส้นสัญญาณ และระบบ Timer/Counter จำนวน 3 ชุดที่โหมคการทำงานเสริมในส่วนของการสร้างสัญญาณ PWM และส่วนของการตรวจจับสัญญาณ Input Capture มีอุปกรณ์สื่อสารข้อมูลอนุกรม แบบ UART และ SPI และยังมีระบบการแปลงสัญญาณ Analog to Digital ขนาด 10 บิตจำนวน 8 ช่องสัญญาณ ที่มาพร้อมกับ MCU มี Watchdog Timer เพื่อตรวจสอบการทำงานของไมโครคอนโทรลเลอร์ และมีระบบการประหยัดพลังงานอีก 3 ระบบ



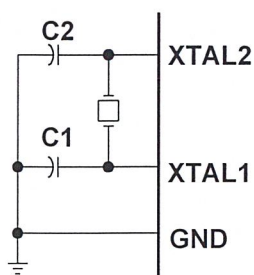
รูปที่ 2.29 แผนผังการทำงาน

2.9.2 รายละเอียดของขาสัญญาณ

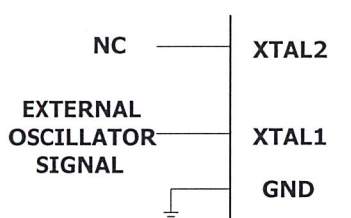
1. Vcc ขาจ่ายไฟให้กับ CPU
2. GND กราวด์
3. Port A (PA7..PA0) เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้แต่ละขาของพอร์ตสามารถ PULL UP ภายในอิสระแยกจากกัน ซึ่งสามารถรับกระแส SINK ได้ 20 mA โดยพอร์ต A ยังใช้เป็นขาอินพุตเพื่อรับสัญญาณอนาล็อกในส่วนของกาแปลงสัญญาณ ANALOG TO DIGITAL
4. Port B (PB7..PB0) เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้แต่ละขาของพอร์ตสามารถ PULL UP ภายในอิสระแยกจากกัน ซึ่งสามารถรับกระแส SINK ได้ 20 mA โดยในแต่ละขาสัญญาณจะถูกใช้งานในฟังก์ชันอื่นๆอีก
5. Port C (PC7..PC0) เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้แต่ละขาสัญญาณสามารถ PULL UP ภายในอิสระแยกจากกัน ซึ่งสามารถรับกระแส SINK ได้ 20 mA และยังสามารถนำไปใช้งานอื่นๆอีก
6. Port D (PD7..PD0) เป็นพอร์ต 2 ทิศทาง ขนาด 8 บิต โดยสามารถกำหนดให้แต่ละขาสัญญาณสามารถ PULL UP ภายในอิสระแยกจากกัน ซึ่งสามารถรับกระแส SINK ได้ 20 mA และยังสามารถนำไปใช้งานอื่นๆอีก
7. RESET ขารีเซ็ต
8. XTAL1 เป็นขาอินพุตของ OSC
9. XTAL2 เป็นขาอินพุตของ OSC
10. Avcc ใช้จ่ายไฟให้กับวงจร Analog to Digital ที่อยู่ภายใน MCU
11. AREF เป็นขาแรงดันอ้างอิงที่ใช้งานในส่วนของวงจร Analog to Digital
12. AGND ขากราวด์ของวงจร Analog to Digital

2.9.3 การใช้งาน CRYSTAL OSCILLATOR

โดย XTAL1 เป็นขา อินพุต และขา XTAL2 เป็นเอาต์พุต ซึ่งถ้าต้องการใช้ OSC ภายในจะต้องต่อ CRYSTAL คร่อมขา XTAL และ XTAL2 โดยมี CAPACITOR ต่อจากขาทั้ง 2 ลงกราวด์ ถ้าต้องการ OSC จากภายนอก ให้ปล่อยขา XTAL2 ลอย และป้อน CLOCK เข้าที่ขา XTAL1 ซึ่งแสดงไว้ในรูปที่ 2.30 และรูปที่ 2.31



รูปที่ 2.30 การใช้ OSC ภายใน MCU



รูปที่ 2.31 การใช้ OSC ภายนอก MCU

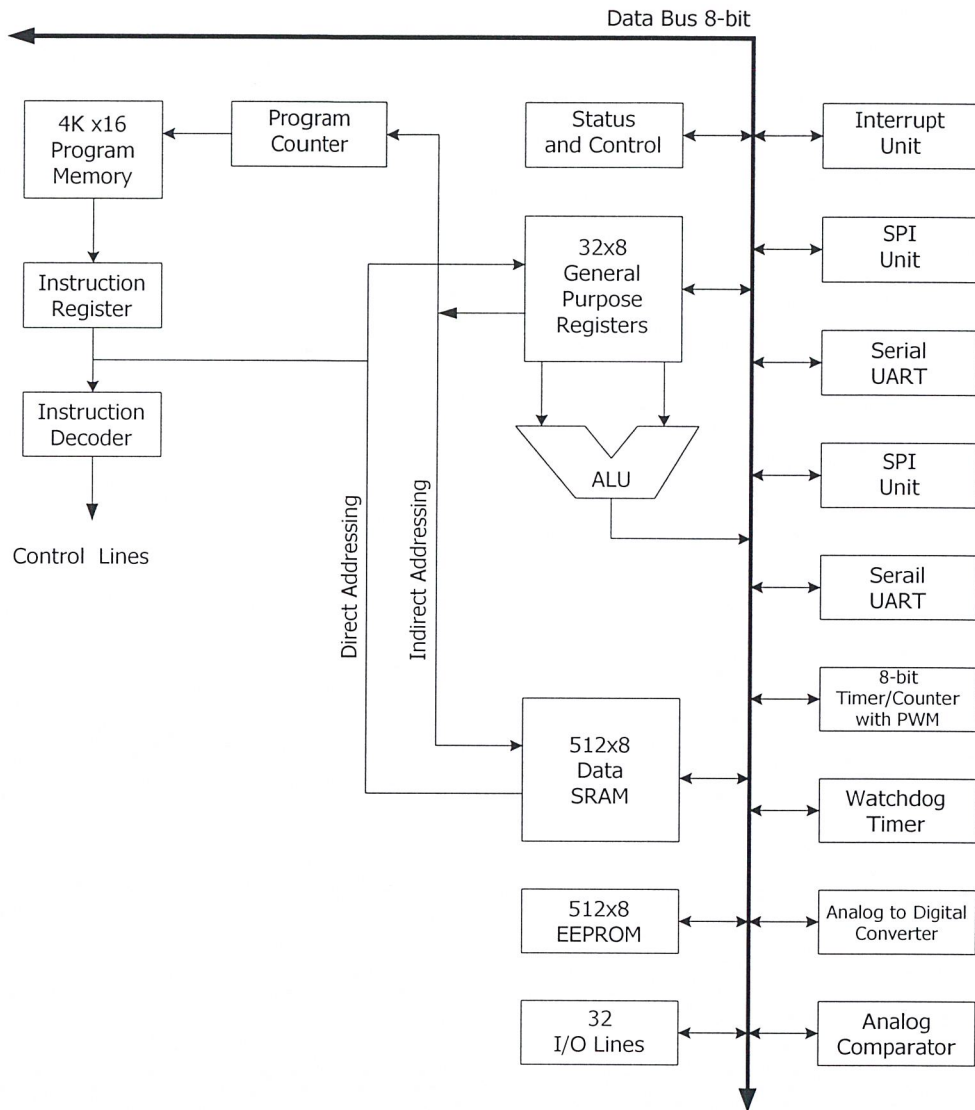
2.9.4 สถาปัตยกรรมภายในและรีจิสเตอร์ใช้งานทั่วไป

1.) สถาปัตยกรรมภายใน

โครงสร้างภายในจะประกอบด้วยรีจิสเตอร์ใช้งานทั่วไปขนาด 8 บิตจำนวน 32 ตัวที่สามารถเข้าถึงข้อมูลได้ใน 1 Clock ซึ่งหมายความว่า MCU สามารถจัดการข้อมูลภายในรีจิสเตอร์ใช้งานทั่วไปได้เสร็จภายใน 1 Clock ของสัญญาณนาฬิกา

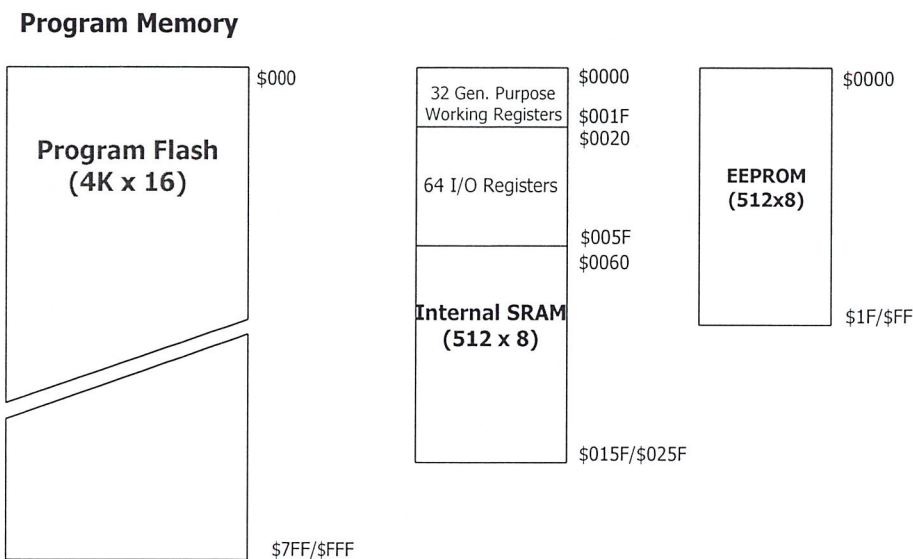
โดยรีจิสเตอร์ R26-R31 เป็นรีจิสเตอร์ขนาด 8 บิต จำนวน 6 ตัว สามารถจับคู่เพื่อใช้เป็นรีจิสเตอร์ขนาด 16 บิต 3 ตัว โดยใช้ชื่อว่ารีจิสเตอร์ X, Y และ Z

ALU จะสนับสนุนการกระทำทางคณิตศาสตร์และลอจิกระหว่างรีจิสเตอร์ กับรีจิสเตอร์ หรือระหว่างรีจิสเตอร์กับค่าคงที่ ซึ่งเรียกรวมการใช้รีจิสเตอร์ใช้งานทั่วไป สามารถกระทำได้โดยการอ้างหน่วยความจำภายใน ที่ตำแหน่ง \$00-\$1F จำนวน 32 ตำแหน่ง และใน MCU ได้จัดแบ่งให้มีรีจิสเตอร์ที่ใช้ควบคุมการทำงานของหน่วยอินพุตและเอาต์พุตต่างๆ อีก 64 ตำแหน่ง โดยสามารถเรียกรวมการใช้งานได้โดยการอ้างตำแหน่งหน่วยความจำที่ตำแหน่ง \$20-\$5F ดังรูปที่ 2.32 แสดงสถาปัตยกรรมแบบ RISC ของ AT90S8535



รูปที่ 2.32 สถาปัตยกรรมแบบ RISC ของ AT90S8535

ระบบการทำงานของไมโครคอนโทรลเลอร์ใช้หลักการออกแบบของ HAVARD ด้วยการแยกระบบบัสของ PROGRAM และ DATA ออกจากกัน โดยโปรแกรมจะมีการประมวลผลด้วย SINGLE LEVEL PIPELINING ซึ่งทำให้ CPU สามารถ Fetch และ Execute คำสั่งได้ภายใน 1 คาบเวลาด้วยคำสั่ง JUMP และ CALL แบบ RELATIVE ที่สามารถกระโดดข้ามการทำงานได้ไกลถึง 2K/4K ซึ่งใน 1 คำสั่งจะใช้รหัสการทำงาน 16 บิต หรือ 1 WORD โดยทุกครั้งที่มีการ อินเตอร์รัพท์ หรือการข้ามไปทำงานในโปรแกรมน้อยค่าของ PROGRAM COUNTER (PC) จะถูกเก็บลง STACK ซึ่งจะใช้พื้นที่หน่วยความจำใน SRAM บางส่วนเพื่อทำเป็นพื้นที่ของ STACK ดังรูปที่ 2.33 แสดงโครงสร้างของหน่วยความจำ



รูปที่ 2.33 โครงสร้างของหน่วยความจำ

รีจิสเตอร์ทั้งหมดสามารถใช้ชุดคำสั่ง เพื่อเข้าถึงได้โดยตรงและจะใช้ช่วงเวลาการเข้าถึงเพียง 1 Clock โดยคำสั่ง SBCI, SUBI, CPI, ANDI และ ORI ซึ่งกระทำระหว่างรีจิสเตอร์กับรีจิสเตอร์และคำสั่ง LDI ที่ใช้โหลดค่าคงที่เข้าในรีจิสเตอร์ จะต้องใช้งานกับรีจิสเตอร์ R16-R31 ส่วนคำสั่ง SBC, SUB, CP, AND และ OR และคำสั่งใช้งานอื่นๆสามารถใช้งานได้ในรีจิสเตอร์ทั่วไปดังรูปที่ 2.34

	7	0	Addr.	
General Purpose Working Registers	R0		\$00	
	R1		\$01	
	R2		\$02	
	...			
	R13		\$0D	
	R14		\$0E	
	R15		\$0F	
	R16		\$10	
	R17		\$11	
	...			
	R26		\$1A	X-register low byte
	R27		\$1B	X-register high byte
	R28		\$1C	Y-register low byte
	R29		\$1D	Y-register high byte
	R30		\$1E	Z-register low byte
	R31		\$1F	Z-register high byte

รูปที่ 2.34 โครงสร้างของรีจิสเตอร์ใช้งานทั่วไป

2.) หน่วยประมวลผลทางคณิตศาสตร์และลอจิก

ระบบการประมวลผลที่มีประสิทธิภาพของ AVR คือ ALU สามารถสื่อสารข้อมูลโดยตรงกับ รีจิสเตอร์ใช้งานทั่วไปได้ทั้ง 32 ตัว โดย ALU ได้จัดแบ่งระบบการจัดการข้อมูลไว้ 3 ส่วนคือ ส่วนของการจัดการทางคณิตศาสตร์ ส่วนของการกระทำทางลอจิก และในส่วนของการทำงานลอจิก และในส่วนของการทำงานกับบิต

3.) หน่วยความจำ SRAM

หน่วยความจำภายใน MCU จัดไว้ 608 ตำแหน่งโดยหน่วยความจำทั้งหมดถูกแบ่งออกเป็นพื้นที่ของรีจิสเตอร์ใช้งานทั่วไป รีจิสเตอร์ใช้งาน I/O และหน่วยความจำภายใน SRAM

โดย 96 ตำแหน่งแรกจะถูกออกแบ่งเป็นส่วนของรีจิสเตอร์ 512 ตำแหน่ง ถูกจัดไว้เป็นหน่วยความจำภายใน SRAM

การเข้าถึงข้อมูลถูกแบ่งออกเป็น 5 ส่วน คือ Direct, Indirect with Displacement, Indirect with PreDecrement และ Indirect with Post-Increment

Registers		Data Address Space
R0		\$0000
R1		\$0001
R2		\$0002
...		...
R29		\$001D
R30		\$001E
R31		\$001F

I/O Register		
\$00		\$0020
\$01		\$0021
\$02		\$0022
...		...
\$3D		\$005D
\$3E		\$005E
\$3F		\$005F

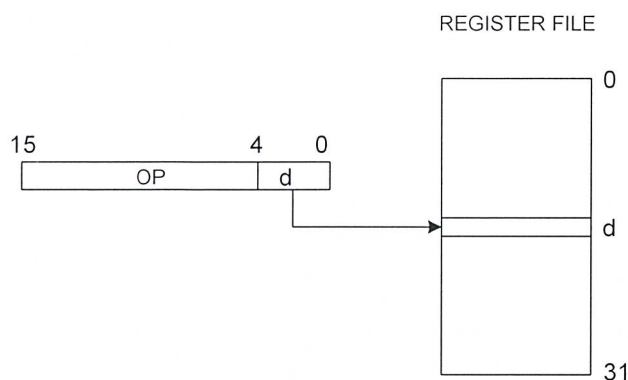
		Internal SRAM
		\$0060
		\$0061
		...
		\$015E/\$025E
		\$015F/\$025F

รูปที่ 2.35 การจัดการหน่วยความจำ SRAM การเข้าถึงข้อมูล

4.) การเข้าถึงข้อมูล

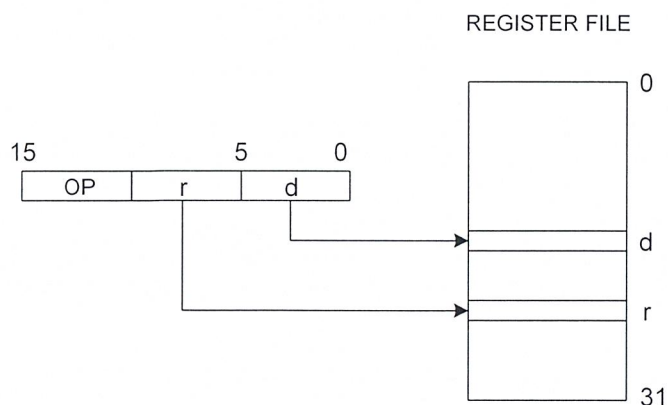
ไมโครคอนโทรลเลอร์ AT90S8535 เป็นไมโครคอนโทรลเลอร์ที่ถูกสร้างขึ้นมาจากสถาปัตยกรรมแบบ RISC โดย PROGRAM MEMORY เป็นแบบ FLASH ซึ่งโครงสร้างภายในถูกออกแบบให้มีการเข้าถึงข้อมูลได้หลายลักษณะดังนี้

4.1) Register Direct, Single Register Rd ค่าของตัวกระทำจะเก็บไว้ในรีจิสเตอร์ d(Rd) ดังรูปที่ 2.36



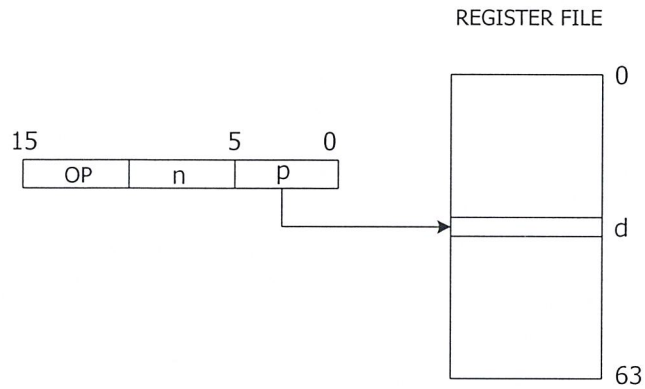
รูปที่ 2.36 Register Direct, Single Register Rd

4.2) Register Direct, Two Register Rd And Rr ค่าของตัวกระทำจะเก็บไว้ในรีจิสเตอร์ r(Rr) และ d(Rd) และผลของการกระทำจะเก็บไว้ในรีจิสเตอร์ d(Rd) ดังรูปที่ 2.37



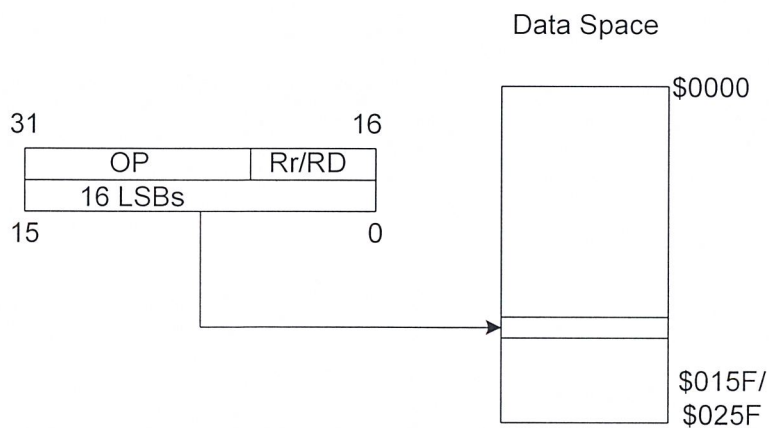
รูปที่ 2.37 Register Direct, Two Register Rd And Rr

4.3) **I/O Direct** ตำแหน่งของตัวกระทำจะถูกเก็บไว้ใน 6 บิตของรหัสคำสั่ง n คือ ตำแหน่งของรีจิสเตอร์ที่เป็น SOURCE หรือ DESTINATION ดังรูปที่ 2.38



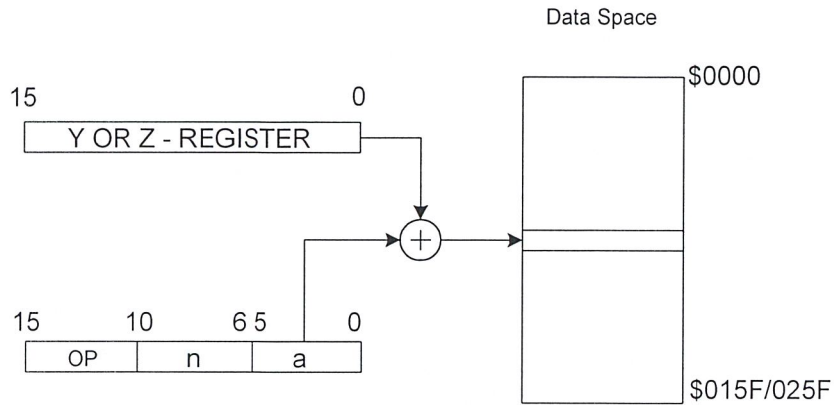
รูปที่ 2.38 การเข้าถึงข้อมูลแบบ Direct Register Addressing R_d and R_r

4.4) **DATA Direct** ตำแหน่งหน่วยความจำ 16 บิต จะเก็บไว้ใน 16 บิตล่างของรหัสคำสั่งที่มีขนาด 2 WORD โดย R_d , R_r จะใช้ในการกำหนดเป็นรีจิสเตอร์ SOURCE หรือ DESTINATION ดังรูปที่ 2.39



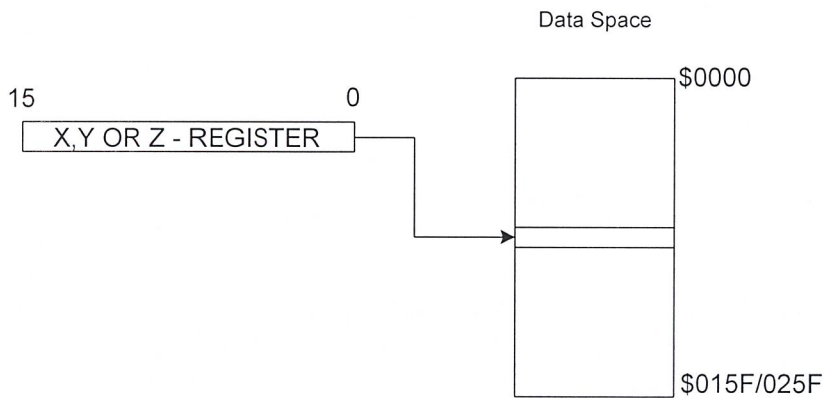
รูปที่ 2.39 การเข้าถึงข้อมูลแบบ Direct Data Addressing

4.5) **Data Indirect With Displacement** ตำแหน่งของตัวกระทำคือผลบวกของรีจิสเตอร์ Y หรือ Z กับตำแหน่งหน่วยความจำขนาด 6 บิตที่บรรจุรหัสคำสั่งดังรูปที่ 2.40



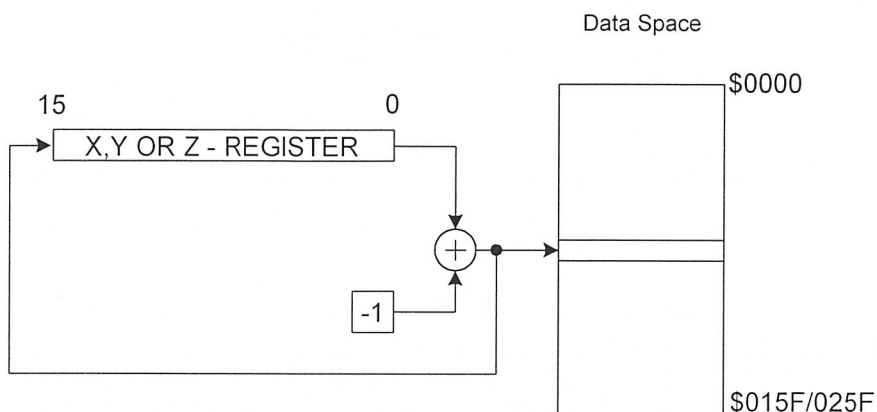
รูปที่ 2.40 การเข้าถึงข้อมูลแบบ Data Indirect with Displacement

4.6) Data Indirect Addressing ตำแหน่งของตัวกระทำคือค่าในของรีจิสเตอร์ X, Y หรือ Z ดังรูปที่ 2.41



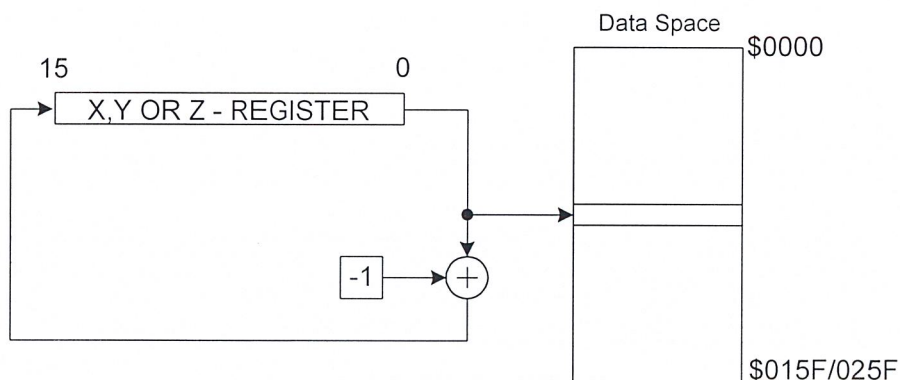
รูปที่ 2.41 การเข้าถึงข้อมูลแบบ Data Indirect Addressing

4.7) Data Indirect With Pre-Displacement รีจิสเตอร์ X, Y และ Z จะถูกลดค่าก่อนการทำคำสั่ง 1 คำ ซึ่งตำแหน่งของตัวกระทำคือค่าที่อยู่ในรีจิสเตอร์ X, Y หรือ Z ภายหลังจากที่ลดค่าแล้ว 1 คำ ดังรูปที่ 2.42



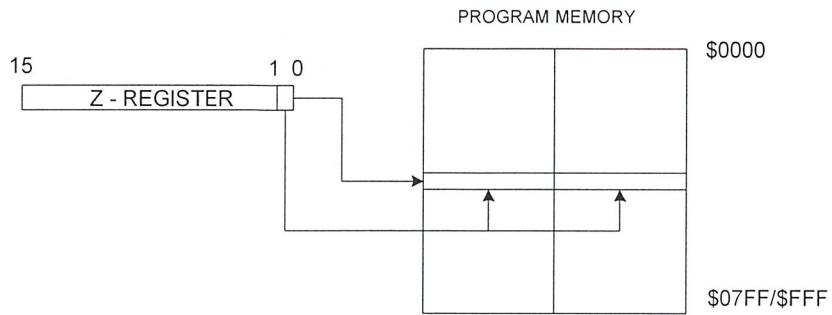
รูปที่ 2.42 การเข้าถึงข้อมูลแบบ Data Indirect with Pre-Displacement

4.8) **Data Indirect With Post- Increment** ตำแหน่งของตัวกระทำคือค่าที่บรรจุอยู่ในรีจิสเตอร์ X, Y หรือ Z โดยเมื่อการทำคำสั่งแล้วค่าในรีจิสเตอร์ X, Y หรือ Z จะถูกเพิ่มขึ้น 1 ค่า ดังรูปที่ 2.43



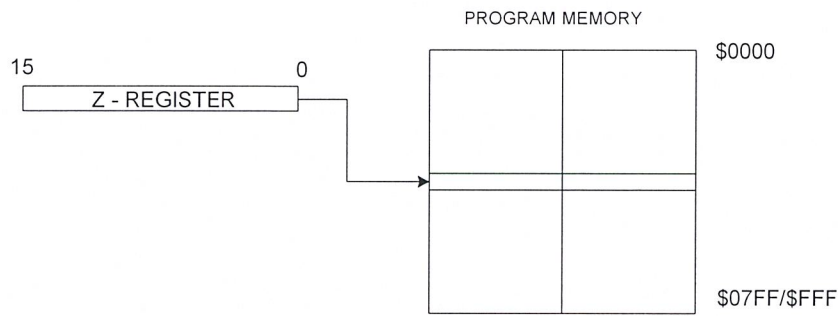
รูปที่ 2.43 การเข้าข้อมูลแบบ Data Indirect with Post- Increment

4.9) **Constant Addressing Using The LPM Instruction** ตำแหน่งหน่วยความจำที่ CPU จะกระทำคำสั่งจะถูกเก็บไว้ในรีจิสเตอร์ Z โดย 15 บิตบน ของรีจิสเตอร์ Z จะถูกใช้ในการกำหนดตำแหน่งของหน่วยความจำในช่วง 0-2K/4K ส่วน LSB บิต ในรีจิสเตอร์ Z จะเป็นบิตที่ใช้เลือก LOW BYTE หรือ HIGH BYTE (LSB = 0 จะเป็นการเลือก HIGH BYTE) ดังรูปที่ 2.44



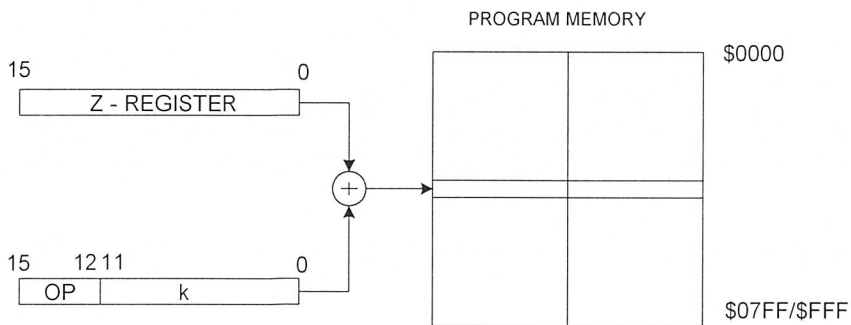
รูปที่ 2.44 การเข้าถึงข้อมูลแบบ Constant Addressing Using the LPM Instruction

4.10) Indirect Program Addressing, RJMP And RCALL CPU จะกระทำ คำสั่งที่ ตำแหน่งหน่วยความจำที่บรรจุในรีจิสเตอร์ ดังรูปที่ 2.45



รูปที่ 2.45 การเข้าถึงข้อมูลแบบ Code Memory Constant Addressing

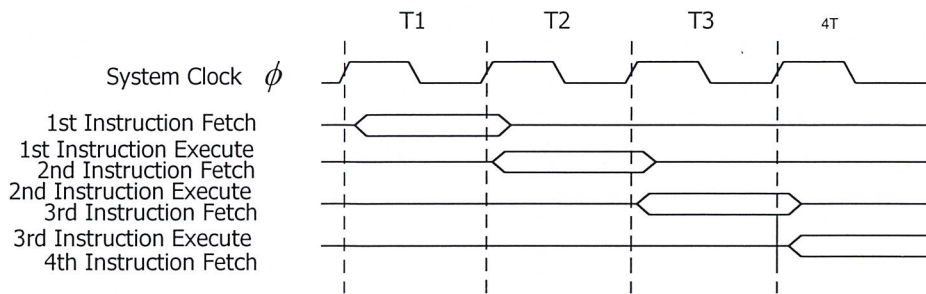
4.11) Relative Program Addressing, RJMP And RCALL CPU จะกระโดดไปทำงานที่ ตำแหน่ง PC+k+1 โดยค่าของ k จะอยู่ระหว่าง -2048 ถึง 2047 ดังรูปที่ 2.46



รูปที่ 2.46 การเข้าถึงข้อมูลแบบ Relative Program Addressing

5.) หน่วยความจำข้อมูลแบบ EEPROM

ไมโครคอนโทรลเลอร์ AT90S4434/AT90S8535 มีหน่วยความจำแบบ EEPROM ขนาด 256/512 BYTE ซึ่งสามารถอ่านและเขียนครั้งละ 1 byte โดยสามารถทำการลบและเขียนข้อมูลใหม่ได้ 100,000 ครั้ง ดังรูปที่ 2.47



รูปที่ 2.47 การ Fetch และ Executions แบบขนาน

2.9.5 การใช้งาน ในโหมด PWM

การทำงานในโหมด PWM ของไมโครคอนโทรลเลอร์ AT90S8535 จะอยู่ในส่วนของ Timer/Counter1 จะสามารถเลือกใช้งานได้ 8 ถึง 9 บิต โดยเอาท์พุทที่ได้จะออกที่ขา PD5(OC1A) และขา PD(OC1B)

ในการทำงาน Timer/Counter1 จะนับขึ้นและนับลง ซึ่งจะนับขึ้นจาก 000 ถึงค่าสูงสุด (ตามที่กำหนดในตารางที่ 1) และจะนับจากค่าสูงสุดลงมาที่ 0000 แล้วจึงนับขึ้นอีกครั้ง

เมื่อค่าใน Timer/Counter1 เท่ากับค่าในรีจิสเตอร์ OCR1A หรือ OCR1B จะทำให้ขา PD5 (OC1A)/PD1(OC1B) เปลี่ยนแปลงตามที่กำหนดในบิต COM1A1/COM1A0 หรือ COM1B0

ตารางที่ 1 การกำหนดค่าสูงสุดของ PWM

PWM Resolution	Timer TOP value	Frequency
8-bit	\$00FF(255)	$F_{TC}/510$
9-bit	\$01FF(511)	$F_{TC}/1022$
10-bit	\$03FF(1023)	$F_{TC}/2046$

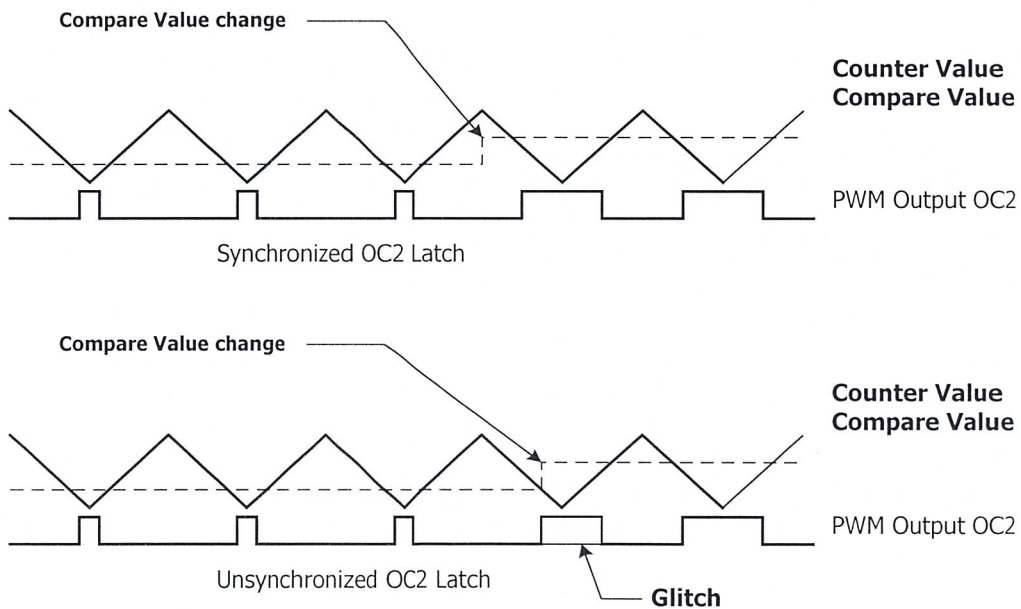
ตารางที่ 2 การกำหนดลักษณะสัญญาณที่ขา PD5(OC1A) และ PD6(OC1B) ในโหมด PWM

COM1X1	COM1X0	Effect on OCX1
0	0	Not connected
0	1	Not connected
1	0	Cleared on compare match, Uncounting. Set on compare match; Downcounting (non-inverted PWM).
1	1	Cleared on compare match, Downcounting. Set on compare match; upcounting (inverted PWM).

Note: X=A or B

เมื่อ OCR1 มีค่าเท่ากับ 0000 หรือค่าสูงสุด จะทำให้เอาต์พุตขา OC1A/OCA1B มีลอจิกเป็น LOW หรือ HIGH ตามที่กำหนดในบิต COM1B1/COM1B0 ซึ่งแสดงในรูปที่ 53

เมื่อ Timer/Counter1 เกิด Overflow และค่าการนับเป็น 0000 จะทำให้บิตTOV1 เซ็ตเป็น 1



รูปที่ 2.48 ผลของ Unsynchronized OCR1 Latching

ตารางที่ 3 เอาต์พุตของ PWM (OCR1X หรือ TOP)

COM1X1	COM1X0	OCR1X	Output OC1X
1	0	\$0000	L
1	0	TOP	H
1	1	\$0000	H
1	1	TOP	L

Note: X = A or B

2.9.6 การแปลงสัญญาณอนาลอกเป็นดิจิทัล ใน AT90S8535

ใน AT90S8535 จัดให้มีวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลขนาด 10 บิต 8 Channel ซึ่งแต่ละ Channel จะรับสัญญาณเข้ามาทางแต่ละขาของพอร์ต A โดยในระบบจะมีวงจร SAMPLE AND HOLD เพื่อช่วยให้สัญญาณอนาลอกที่รับเข้ามาเพื่อแปลงสัญญาณดิจิทัลมีระดับสัญญาณคงที่ โดยวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัลจะมีแหล่งจ่ายไฟและกราวด์แยกกันต่างหากจากแหล่งจ่ายไฟของระบบ ซึ่งในการใช้งานจริงไม่ควรให้ความแตกต่างของแรงดันของวงจรอนาลอกและแรงดันไฟของระบบแตกต่างกันเกิน $\pm 0.3V$ ซึ่งในการใช้งานจะต้องจ่ายแรงดันไฟอ้างอิงและกราวด์ที่ขา AREF ในช่วงของระดับแรงดัน $V_{cc} - AGND$

ในส่วนของการแปลงสัญญาณอนาลอกเป็นดิจิทัล สามารถทำงานได้ 2 MODE คือ แบบ Single Conversion Mode และ แบบ Free Running Mode

การทำงาน Single Conversion Mode ผู้ใช้ต้องเป็นผู้กำหนดการใช้งานขึ้นเอง แต่ในส่วน ของ แบบ Free Running Mode วงจร Analog to Digital จะเป็นตัวจัดการอ่านข้อมูลและเก็บใน ADC Data Register ซึ่งบิต ADFR ใน Register ADCSR จะเป็นบิตที่ใช้เลือกโหมดการใช้งานของวงจร Analog to Digital

การกำหนดให้วงจร Analog to Digital ทำงาน สามารถทำได้โดยการ เช็ทบิตADEN ใน รีจิสเตอร์ ADCSR ให้เป็น 1 ซึ่งสัญญาณอนาลอกจะถูก Conversion เมื่อเช็ทบิต ADSC ให้เป็น 1 โดยบิตนี้จะ เป็น 1 ไปตลอดจนกระทั่ง Conversion สัญญาณจะเสร็จเรียบร้อย จึงทำให้บิตนี้เป็น 0 โดยอัตโนมัติ แต่ถ้ามีการเปลี่ยน Channel ของการแปลงสัญญาณขณะที่ Channel เดิมยัง Conversion อยู่ วงจร Analog to Digital จะ Conversion สัญญาณ Channel เดิมให้เสร็จก่อนแล้วจึง Conversion สัญญาณ Channel ถัดไป โดยข้อมูลที่ได้อาจจากการแปลงสัญญาณอนาลอกเป็นดิจิทัล จะ

เก็บไว้ในรีจิสเตอร์ ADCH และ ADCL ซึ่งการอ่านข้อมูลจะต้องอ่านข้อมูลจากรีจิสเตอร์ ADCL ก่อนแล้วจึงอ่านข้อมูลจาก ADCH

วงจร Analog to Digital ต้องการ Clock ในการ Sampling ในช่วง 50-200 KHz โดยใน Free Running Mode วงจรต้องการ 13 Clock ในการแปลงสัญญาณจาก Analog to Digital ซึ่งหมายความว่าช่วงเวลาของการแปลงในช่วง 65-260 us

ส่วนของการ Single Conversion จะใช้ 14 Clock ซึ่งการควบคุม Clock ที่จะจ่ายให้กับวงจร Analog to Digital สามารถควบคุมได้จากบิต ADPS0-ADPS2

2.9.7 พอร์ตอินพุต และพอร์ตเอาต์พุต

1. พอร์ต A

เป็นพอร์ต 2 ทิศทางขนาด 8 บิต โดยมีหน่วยควบคุมการทำงานต่างๆ ของพอร์ต คือ รีจิสเตอร์ PORTA (Data Register) อยู่ที่ตำแหน่ง \$1B(\$3B), รีจิสเตอร์ DDRA (Data Direction Register) อยู่ที่ตำแหน่ง \$1A(\$3A) และ PINA (Port A Input Pins) อยู่ที่ตำแหน่ง \$19(\$39) โดย PINA จะสามารถอ่านได้อย่างเดียว ไม่สามารถเขียนข้อมูลลงได้ ในขณะที่ PORTA และ DDRA สามารถอ่านและเขียนได้ โดยแต่ละขาสัญญาณของพอร์ต A สามารถกำหนดให้มีความต้านทาน PULL-UP ทำงานในสถานะ Active หรือไม่ Active ก็ได้ โดยแต่ละขาสัญญาณของพอร์ต A สามารถรับกระแส (Sink Current) ได้ 20 mA ซึ่งถ้าขาของพอร์ต A ใช้งานเป็นอินพุตและภายนอกมีความต้านทาน Pull Low ในขณะที่ความต้านทาน Pull Up ภายใน Active จะทำให้ MCU จ่ายกระแสออกมาภายนอก

พอร์ต A จะถูกนำไปใช้งานอีกอย่างหนึ่งคือ เป็นขาอินพุตของวงจร Analog to Digital โดยถ้าบางขาของพอร์ต A ถูกใช้งานเป็นขาเอาต์พุตของพอร์ต ในขณะที่บางขาของพอร์ต A ถูกใช้เป็นขาอินพุตของ Analog to Digital ผู้ใช้จะต้องไม่เปลี่ยนแปลงสถานะการทำงานของพอร์ต A ในขณะที่วงจร Analog to Digital กำลัง Conversion สัญญาณซึ่งอาจจะทำให้การทำการแปลงสัญญาณ ถูกขัดจังหวะ จะมีผลให้การแปลงสัญญาณผิดพลาด

2. พอร์ต B

พอร์ต B เป็นพอร์ตสองทิศทางขนาด 8 บิต โดยมีรีจิสเตอร์ที่ใช้ควบคุมพอร์ต B อยู่ 3 ตัว คือ รีจิสเตอร์ PORTB อยู่ที่ตำแหน่ง \$18(\$38), รีจิสเตอร์ DDRB อยู่ที่ตำแหน่ง \$17(\$37) และ PINB อยู่ที่ตำแหน่ง \$16(\$36) โดยพอร์ต B แต่ละขาสามารถแยกกำหนดให้มีความต้านทาน PULL UP ได้ตามต้องการ ซึ่งในแต่ละขาสามารถรับกระแส (SINK CURRENT) ได้ 20mA โดยขาของพอร์ต B สามารถใช้งานเป็นฟังก์ชันอื่น ๆ ได้ดังตารางที่ 4

ตารางที่ 4 การใช้งานฟังก์ชันอื่น ๆ ของพอร์ต B

Port Pin	Alternate Function
PB0	T0 (Timer/Counter 0 external counter input)
PB1	T1 (Timer/Counter 1 external counter input)
PB2	AIN0 (Analog comparater positive input)
PB3	AIN1 (Analog comparater positive input)
PB4	SS (SPI Slave Select $\overline{\text{input}}$)
PB5	MOSI (SPI Bus Master Output/Slave Input)
PB6	MISO (SPI Bus Master Output/Slave Input)
PB7	SCK (SPI Bus Clock)

3. พอร์ต C

พอร์ต C เป็นพอร์ตสองทิศทางขนาด 8 บิต โดยมีหน่วยควบคุมการทำงานของพอร์ต จำนวน 3 หน่วย คือ PORTC(Data Register - PORTC) อยู่ที่ตำแหน่งหน่วยความจำ \$15(\$35), DDRC (Data Direction Register – DDRC) อยู่ที่ตำแหน่งหน่วย \$14(\$34) และ PINC (Port C Input Pin – PINC) อยู่ที่ตำแหน่ง \$13(\$33) โดย PINC จะสามารถอ่านได้อย่างเดียว ในขณะที่ PORTC และ DDRC จะสามารถทั้งอ่านและเขียนได้ โดยแต่ละขาของพอร์ต C สามารถแยกการกำหนดความต้านทาน PULL UP ได้ ในขณะที่พอร์ต C แต่ละขาสามารถรับกระแส (SINK CURRENT) ได้ 20mA โดยถ้าภายในกำหนดให้มีความต้านทาน PULL UP และภายนอกมีความต้านทาน PULL LOW จะทำให้ MCU จ่ายกระแสออกภายนอก

4. พอร์ต D

พอร์ต D เป็นพอร์ตสองทิศทางขนาด 8 บิต ที่มีหน่วยควบคุมการทำงานของพอร์ต คือ PORTD(Data Register – PORTD) อยู่ที่ตำแหน่ง \$12(\$32), DDRD(Data Direction Register) อยู่ที่ตำแหน่ง \$11(\$31) และ PIND(Port D Input Pins) อยู่ที่ตำแหน่ง \$10(\$30) โดย PIND สามารถอ่านได้อย่างเดียว ในขณะที่ PORTD และ DDRD สามารถทั้งอ่านและเขียน โดยพอร์ต D สามารถรับกระแสได้ 20mA ซึ่งแต่ละขาของพอร์ต D สามารถเลือกฟังก์ชันการทำงานอื่น ๆ ได้อีกดังตารางที่ 5

ตารางที่ 5 ฟังก์ชันอื่น ๆ ของพอร์ต D

Port Pin	Alternate Function
PD0	RDX (UART Input line)
PD1	TDX (UART Output line)
PD2	INT0 (External interrupt 0 input)
PD3	INT1 (External interrupt 1 input)
PD4	OC1B (Timer/Counter 1 output compare B match output)
PD5	OC1A (Timer/Counter 1 output compare A match output)
PD6	ICP (Timer/Counter 1 input capture pin)
PD7	Os2 (Timer/Counter 2 output compare math output)

2.9.8 กลุ่มคำสั่ง

1. กลุ่มคำสั่งการกระทำทางคณิตศาสตร์และลอจิก

ADD Rd,Rr

คำสั่งบวกข้อมูลระหว่างรีจิสเตอร์ 2 ตัว ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ Rd

ADC Rd,Rr

คำสั่งบวกข้อมูลระหว่างรีจิสเตอร์ 2 ตัวและ Carry Flag ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ Rd

ADIW RdI,K

คำสั่งบวกข้อมูลแบบ WORD ระหว่างรีจิสเตอร์กับค่าคงที่ ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ RdI

SUB Rd,Rr

คำสั่งลบข้อมูลระหว่างรีจิสเตอร์ 2 ตัว ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ Rd

SUBI Rd,K

คำสั่งลบข้อมูลระหว่างรีจิสเตอร์กับค่าคงที่ ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ Rd

SBC Rd,Rr

คำสั่งลบข้อมูลระหว่างรีจิสเตอร์ 2 ตัว กับ Carry Flag ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ Rd

SBCI Rd,K

คำสั่งลบข้อมูลระหว่างรีจิสเตอร์กับค่าคงที่ และ Carry Flag ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ Rd

SBIW Rd,K

คำสั่งลบข้อมูลขนาด WORD

AND Rd,Rr

คำสั่ง AND ข้อมูลระหว่างรีจิสเตอร์ 2 ตัว

ANDI Rd,K

คำสั่ง AND ข้อมูลระหว่างรีจิสเตอร์กับค่าคงที่

OR Rd,Rr

คำสั่ง OR ข้อมูลระหว่างรีจิสเตอร์ 2 ตัว

ORI Rd,K

คำสั่ง OR ข้อมูลระหว่างรีจิสเตอร์ กับค่าคงที่ ผลลัพธ์ที่ได้เก็บไว้ในรีจิสเตอร์ Rd

EOR Rd,Rr

คำสั่ง Exclusive OR ข้อมูลระหว่างรีจิสเตอร์ 2 ตัว

COM Rd

คำสั่งทำ One Complement ข้อมูลในรีจิสเตอร์ Rd

NEG Rd

คำสั่งทำ Two Complement ข้อมูลในรีจิสเตอร์ Rd

SBR Rd,K

คำสั่งเซตบิตในรีจิสเตอร์ Rd

CBR Rd,K

คำสั่งเคลียร์บิตในรีจิสเตอร์ Rd

INC Rd

คำสั่งเพิ่มค่าของข้อมูลในรีจิสเตอร์ Rd ขึ้น 1 ค่า

DEC Rd

คำสั่งลดค่าของข้อมูลในรีจิสเตอร์ Rd ลง 1 ค่า

TST Rd

คำสั่งทดสอบข้อมูลในรีจิสเตอร์ Rd โดยจะมีผลกับ Flag Z,N,V

CLR Rd

คำสั่งเคลียร์ข้อมูลในรีจิสเตอร์ Rd ให้มีค่าเป็น 00

SER Rd

คำสั่งเซตข้อมูลในรีจิสเตอร์ Rd ให้มีค่าเป็น FF

2. กลุ่มคำสั่งการกระโดด

RJMP k

คำสั่งกระโดดแบบ Relative

IJMP

คำสั่งกระโดดไปยังตำแหน่งหน่วยความจำที่ถูกชี้โดยรีจิสเตอร์ Z

RCALL k

คำสั่งกระโดดไปยังโปรแกรมย่อยที่ k

ICALL

คำสั่งกระโดดไปทำงานโปรแกรมย่อยที่ชี้ตำแหน่งโดยรีจิสเตอร์ Z

RET

คำสั่งกลับจากโปรแกรมย่อย

RETI

คำสั่งกลับจากโปรแกรมบริการอินเตอร์รัพท์

CPSE Rd,Rr

คำสั่งเปรียบเทียบข้อมูลในรีจิสเตอร์ Rd กับข้อมูลในรีจิสเตอร์ Rr โดยถ้าข้อมูลทั้งสองมีค่าเท่ากัน CPU ข้ามการทำคำสั่งถัดไป 1 คำสั่ง

CP Rd,Rr

คำสั่งเปรียบเทียบข้อมูลในรีจิสเตอร์ Rd กับข้อมูลในรีจิสเตอร์ Rr โดยจะมีผลต่อ Flag Z, N, V, C, H

CPC Rd,Rr

คำสั่งเปรียบเทียบข้อมูลในรีจิสเตอร์ Rd กับข้อมูลในรีจิสเตอร์ Rr และ Carry Flag โดยมีการกระทำ Rd-Rr-C ซึ่งทำให้มีผลกับ Flag Z, N, V, C, H

CPI Rd,k

คำสั่งเปรียบเทียบข้อมูลในรีจิสเตอร์ Rd กับค่าคงที่ โดย CPU จะกระทำ Rd-k ซึ่งมีผลกับ Flag Z, N, V, C, H

SBRC Rr,b

เมื่อ CPU ทำคำสั่ง SBRC จะทำให้ CPU ข้ามการกระทำคำสั่งถัดไป 1 คำสั่ง ถ้าบิตที่ b ในรีจิสเตอร์ Rr มีค่าเป็น 0

SBRS Rr,b

เมื่อ CPU ทำคำสั่ง SBRS จะทำให้ CPU ข้ามการกระทำคำสั่งถัดไป 1 คำสั่ง ถ้าบิตที่ b ใน รีจิสเตอร์ Rr ถูกเซ็ต

SBIC P,b

เมื่อ CPU ทำคำสั่ง SBIC จะทำให้ CPU ข้ามการกระทำคำสั่งถัดไป 1 คำสั่ง ถ้าบิตที่ b ใน รีจิสเตอร์ I/O ถูกเคลียร์

SBIS P,b

เมื่อ CPU ทำคำสั่ง SBIS จะทำให้ CPU ข้ามการกระทำคำสั่งถัดไป 1 คำสั่ง ถ้าบิตที่ b ใน รีจิสเตอร์ I/O ถูกเซ็ต

BRBS s,k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ี่ ถ้า Flag Status ในรีจิสเตอร์ SREG ถูกเซ็ต [(SREG(s)) = 1] โดยค่าใน $PC = PC + k + 1$

BRBC s,k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ี่ ถ้า Flag Status ในรีจิสเตอร์ SREG ถูกเคลียร์ [(SREG(s)) = 0] โดยค่าใน $PC = PC + k + 1$

BREQ k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ี่ ถ้า Zero Flag เซ็ต ($Z = 1$) โดยค่าใน $PC = PC + k + 1$

BRNE k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ี่ ถ้า Zero Flag เซ็ต ($Z = 0$) โดยค่าใน $PC = PC + k + 1$

BRCS k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ี่ ถ้า Carry Flag เซ็ต ($C = 1$) โดยค่าใน $PC = PC + k + 1$

BRCC k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ี่ ถ้า Carry Flag เคลียร์ ($C = 0$) โดยค่าใน $PC = PC + k + 1$

BRSR k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ี่ ถ้า Carry Flag เคลียร์ ($C = 0$) โดยค่าใน $PC = PC + k + 1$

BRLO k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ชี้ ถ้า Carry Flag เซ็ต ($C = 1$) โดยค่าใน
 $PC = PC + k + 1$

BRMI k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ชี้ ถ้า Flag เซ็ต ($N = 1$) โดยค่าใน $PC =$
 $PC + k + 1$

BRPL k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ชี้ ถ้า Flag เคลียร์ ($N = 1$) โดยค่าใน PC
 $= PC + k + 1$

BRGE k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ชี้ ถ้า Flag เคลียร์ ($N \oplus V = 0$) โดยค่า
 ใน $PC = PC + k + 1$

BRLT k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ชี้ ถ้า Flag เคลียร์ ($N \oplus V = 1$) โดยค่า
 ใน $PC = PC + k + 1$

BRHS k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ชี้ ถ้า Half Carry Flag เซ็ต ($H = 1$) โดย
 ค่าใน $PC = PC + k + 1$

BRHC k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ PC ชี้ ถ้า Half Carry Flag เคลียร์ ($H = 0$)
 โดยค่าใน $PC = PC + k + 1$

BRTS k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ $PC = PC + k + 1$ ชี้ ถ้า T Flag เซ็ต

BRTC k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ $PC = PC + k + 1$ ชี้ ถ้า T Flag เคลียร์

BRVS k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ $PC = PC + k + 1$ ชี้ ถ้า Overflow Flag เซ็ต

BRVC k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ $PC = PC + k + 1$ ชี้ ถ้า Overflow Flag
 เคลียร์

BRIE k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ $PC = PC + k + 1$ ถ้า Interrupt Enable บิต

(I = 1)

BRID k

เป็นคำสั่งกระโดดข้ามไปทำคำสั่งในตำแหน่งที่ $PC = PC + k + 1$ ถ้า Interrupt Disable

บิต (I = 0)

3. กลุ่มคำสั่งในการเคลื่อนย้ายข้อมูล

MOV Rd,Rr

เป็นคำสั่งในการเคลื่อนย้ายข้อมูลจากรีจิสเตอร์ ไปยังรีจิสเตอร์

Rd คือ รีจิสเตอร์ R0 – R31

Rr คือ รีจิสเตอร์ R0 – R31

LDI Rd,k

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิต เข้าไปในรีจิสเตอร์

Rd คือ รีจิสเตอร์ R16 – R31

k คือ ค่าคงที่ขนาด 8 บิต 0 – 255

LD Rd,X

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ X
ชี้ เข้าไปในรีจิสเตอร์

Rd คือ รีจิสเตอร์ R0 – R31

X คือ รีจิสเตอร์ X

LD Rd,X+

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ X
ชี้ เข้าไปในรีจิสเตอร์ทั่วไป และเพิ่มค่ารีจิสเตอร์ X ขึ้น 1 หลังจากทีโหลดข้อมูลเสร็จแล้ว

Rd คือ รีจิสเตอร์ R0 – R31

X คือ รีจิสเตอร์ X

LD Rd,-X

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์
X-1 ชี้ เข้าไปในรีจิสเตอร์ทั่วไป และลดค่ารีจิสเตอร์ X ลง 1 ค่าก่อนที่จะโหลดข้อมูลลง

Rd คือ รีจิสเตอร์ R0 – R31

X คือ รีจิสเตอร์ X

LD Rd,Y

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y ซึ่เข้าไปในรีจิสเตอร์

Rd คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

LD Rd,Y+

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y ซึ่เข้าไปในรีจิสเตอร์ทั่วไป และเพิ่มค่ารีจิสเตอร์ Y ขึ้น 1 หลังจากทีโหลดข้อมูลเสร็จแล้ว

Rd คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

LD Rd,-Y

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y-1 ซึ่เข้าไปในรีจิสเตอร์ทั่วไป และลดค่ารีจิสเตอร์ Y ลง 1 ค่าก่อนที่จะโหลดข้อมูลลง

Rd คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

LD Rd,Y+q

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y+q ซึ่เข้าไปในรีจิสเตอร์

Rd คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

q คือ ค่าระยะห่าง มีค่า 0 – 63

LD Rd,Z

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z ซึ่เข้าไปในรีจิสเตอร์

Rd คือ รีจิสเตอร์ R0 – R31

Z คือ รีจิสเตอร์ Z

LD Rd,Z+

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z ซึ่เข้าไปในรีจิสเตอร์ทั่วไป และเพิ่มค่ารีจิสเตอร์ Z ขึ้น 1 หลังจากทีโหลดข้อมูลเสร็จแล้ว

Rd คือ รีจิสเตอร์ R0 – R31

Z คือ รีจิสเตอร์ Z

LD Rd,-Z

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z-1 ซึ่งเข้าไปในรีจิสเตอร์ทั่วไป และลดค่ารีจิสเตอร์ Z ลง 1 ค่าก่อนที่จะโหลดข้อมูลลง

Rd คือ รีจิสเตอร์ R0 – R31

Z คือ รีจิสเตอร์ Z

LD Rd,Z+q

เป็นคำสั่งที่ใช้ในการโหลดข้อมูลขนาด 8 บิตที่อยู่ในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z+q ซึ่งเข้าไปในรีจิสเตอร์

Rd คือ รีจิสเตอร์ R0 – R31

Z คือ รีจิสเตอร์ Z

q คือ ค่าระยะห่าง มีค่า 0 – 63

LDS Rd,k

เป็นคำสั่งโหลดข้อมูลขนาด 8 บิต ที่อยู่ในหน่วยความจำ SRAM ในตำแหน่งที่ k เข้าในรีจิสเตอร์ทั่วไป

Rd คือ รีจิสเตอร์ R0 – R31

k คือ ตำแหน่งหน่วยความจำใน SRAM

ST X,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ X ซึ่ง

Rr คือ รีจิสเตอร์ R0 – R31

X คือ รีจิสเตอร์ X

ST X+,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ X ซึ่ง และทำการเพิ่มค่ารีจิสเตอร์ X ขึ้น 1 ค่า

Rr คือ รีจิสเตอร์ R0 – R31

X คือ รีจิสเตอร์ X

ST -X,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ X-1 ซึ่ง

Rr คือ รีจิสเตอร์ R0 – R31

X คือ รีจิสเตอร์ X

ST Y,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y ซี่

Rr คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

ST Y+,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y ซี่ และทำการเพิ่มค่ารีจิสเตอร์ Y ขึ้น 1 ค่า

Rr คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

ST -Y,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y-1 ซี่

Rr คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

ST Y+q,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Y+q ซี่

Rr คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Y

q คือ ค่าระยะห่าง 0 – 63

ST Z,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z ซี่

Rr คือ รีจิสเตอร์ R0 – R31

Z คือ รีจิสเตอร์ Z

ST Z+,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z ซี่ และทำการเพิ่มค่ารีจิสเตอร์ Z ขึ้น 1 ค่า

Rr คือ รีจิสเตอร์ R0 – R31

Z คือ รีจิสเตอร์ Z

ST -Z,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z-1 ซี่

Rr คือ รีจิสเตอร์ R0 – R31

Y คือ รีจิสเตอร์ Z

ST Z+q,Rr

เป็นคำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ทั่วไปลงในหน่วยความจำตำแหน่งที่รีจิสเตอร์ Z+q ซี่

Rr คือ รีจิสเตอร์ R0 – R31

Z คือ รีจิสเตอร์ Z

q คือ ค่าระยะห่าง 0 – 63

STS k,Rr

คำสั่งบันทึกข้อมูลขนาด 8 บิต จากรีจิสเตอร์ใช้งานทั่วไปลงในหน่วยความจำตำแหน่งที่ k

Rr คือ รีจิสเตอร์ R0 – R31

k คือ ตำแหน่งในหน่วยความจำ 0 – 65535

LPM

โหลด Program Memory

IN Rd,P

คำสั่งอ่านข้อมูลจากรีจิสเตอร์ I/O มาเก็บไว้ในรีจิสเตอร์ทั่วไป

Rd คือ รีจิสเตอร์ R0 – R31

P คือ รีจิสเตอร์ อินพุต/เอาต์พุต

OUT P,Rd

คำสั่งเขียนข้อมูลจากรีจิสเตอร์ทั่วไป เข้าไปในรีจิสเตอร์อินพุต/เอาต์พุต

Rd คือ รีจิสเตอร์ R0 – R31

P คือ รีจิสเตอร์ อินพุต/เอาต์พุต

PUSH Rr

คำสั่งเก็บค่ารีจิสเตอร์ Rr ลงใน STACK

POP Rr

คำสั่งย้ายค่าจาก STACK มาเก็บในรีจิสเตอร์ทั่วไป

4. กลุ่มคำสั่งการกระทำทางบิตและการทดสอบบิต

SBI P,b

คำสั่งเซตบิต b ในรีจิสเตอร์ I/O ตำแหน่งที่ P

I/O (P,b) 1

SBI P,b

คำสั่งเคลียร์บิต b ในรีจิสเตอร์ I/O ตำแหน่งที่ P

I/O (P,b) 0

LSL Rd

เลื่อนข้อมูลในรีจิสเตอร์ Rd ไปทางซ้าย 1 บิต

LSR Rd

เลื่อนข้อมูลในรีจิสเตอร์ Rd ไปทางขวา 1 บิต

ROL Rd

หมุนข้อมูลไปทางซ้าย 1 บิต โดยผ่าน Carry Flag

ROR Rd

หมุนข้อมูลไปทางขวา 1 บิต โดยผ่าน Carry Flag

SWAP Rd

คำสั่งสลับค่า 4 บิต High และ 4 บิต Low ในรีจิสเตอร์ Rd

BSET s

เซต Flag

BCLR s

เคลียร์ Flag

BST Rr,b

คำสั่งโหลดบิต T ลงในบิต b ของรีจิสเตอร์ Rr

BLD Rr,b

คำสั่งโหลดบิต b ในรีจิสเตอร์ Rr เก็บไว้ในบิต T

SEC	คำสั่งเซ็ต Carry Flag
CLC	คำสั่งเคลียร์บิต Carry Flag
SEN	คำสั่งเซ็ต Negative Flag
CLN	คำสั่งเคลียร์ Negative Flag
SEZ	คำสั่งเซ็ต Zero Flag
CLZ	คำสั่งเคลียร์ Zero Flag
SEI	คำสั่ง Enable Global Interrupt
CLN	คำสั่ง Disable Global Interrupt
SES	คำสั่งเซ็ต Signed test Flag
CLS	คำสั่งเคลียร์ Signed test Flag
SEV	คำสั่งเซ็ต 2's complement Overflow
CLV	คำสั่งเคลียร์ 2's complement Overflow
SET	คำสั่งเซ็ตบิต T ในรีจิสเตอร์ SREG
CLT	คำสั่งเคลียร์ T ในรีจิสเตอร์ SREG
SEH	คำสั่งเซ็ต Half Carry Flag ในรีจิสเตอร์ SREG

CLH

คำสั่งเคลียร์ Half Carry Flag ในรีจิสเตอร์ SREG

NOP

คำสั่ง NO OPERATION

SLEEP

คำสั่งให้ CPU ทำงาน SLEEP MODE

WDR

คำสั่งรีเซ็ต Watchdog

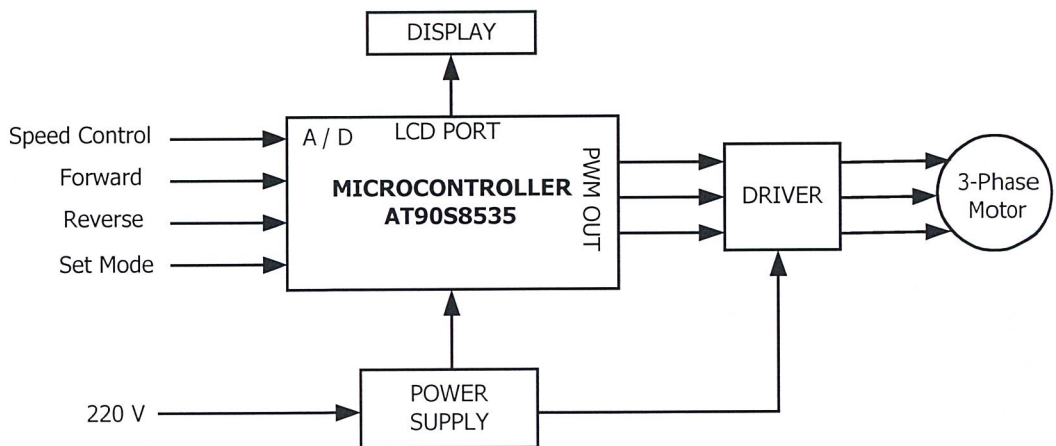
บทที่ 3

การออกแบบ การสร้าง และการทำงาน

เนื้อหาในบทนี้ กล่าวถึงการออกแบบและการสร้างส่วนต่างๆ ของโครงงานเครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส โดยนำหลักการจากทฤษฎีและหลักการที่ได้กล่าวมาแล้วในบทที่ 2 มาประกอบการออกแบบและสร้างวงจรต่างๆ ที่จะนำไปใช้ในโครงงาน

การออกแบบ การสร้าง และการทำงาน จะประกอบไปด้วยแผนผังการทำงานของโครงงาน, วงจรภาคขับเคลื่อนมอเตอร์และสวิทซ์กำลัง, วงจรสร้างสัญญาณ PWM, วงจรแสดงผลและรับคีย์, วงจรป้องกัน, วงจรแหล่งจ่ายไฟ, วิธีการเขียนโปรแกรม และโครงสร้างของโครงงานและการออกแบบ

3.1 แผนผังการทำงานของโครงงาน



รูปที่ 3.1 แผนผังการทำงานของโครงงาน

จากรูปที่ 3.1 เป็นแผนผังการทำงานของโครงงานทั้งหมด โดยไมโครคอนโทรลเลอร์ AT90S8535 จะเป็นตัวควบคุมการทำงานทั้งหมด ซึ่งจะรับข้อมูลจากสวิทซ์ประกอบด้วยสวิทซ์ Forward, Reverse และ Set Mode โดยสวิทซ์ Forward และ Reverse จะเป็นสวิทซ์ควบคุมทิศทาง การหมุนของมอเตอร์ ส่วนสวิทซ์ Set Mode จะเป็นสวิทซ์ตั้งค่าฟังก์ชันการทำงาน และสามารถควบคุมความเร็วของมอเตอร์ได้โดยการปรับค่าแรงดันไฟฟ้าจากตัวต้านทานปรับค่าได้ ไมโคร

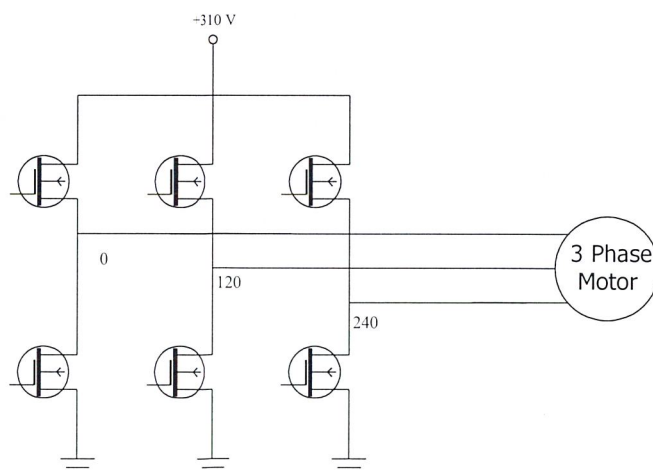
คอนโทรลเลอร์จะแปลงค่าแรงดันไฟฟ้าซึ่งเป็นสัญญาณอนาลอกเป็นสัญญาณดิจิทัล โดยใช้วงจร A/D ซึ่งอยู่ในตัวไมโครคอนโทรลเลอร์ เมื่อไมโครคอนโทรลเลอร์ได้รับสัญญาณอินพุทเหล่านี้ จะนำสัญญาณที่ได้ไปประมวลผลเพื่อสร้างสัญญาณ PWM ไปควบคุมวงจรมอเตอร์ ในส่วนของการแสดงผลจะใช้ LCD ในการแสดงผล โดยไมโครคอนโทรลเลอร์จะมีวงจรขับ LCD อยู่ในตัวไมโครคอนโทรลเลอร์ โดยจะแสดงผลการควบคุมความเร็วและทิศทางการหมุนของมอเตอร์ และแสดงการตั้งค่าฟังก์ชันการทำงาน

3.2 วงจรภาคขั้วมอเตอร์และสวิตซ์กำลัง

ส่วนที่สำคัญในการสร้างเครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส คือส่วนของภาคขั้วมอเตอร์และสวิตซ์กำลัง จึงต้องออกแบบและสร้างให้มีประสิทธิภาพสูงสุด

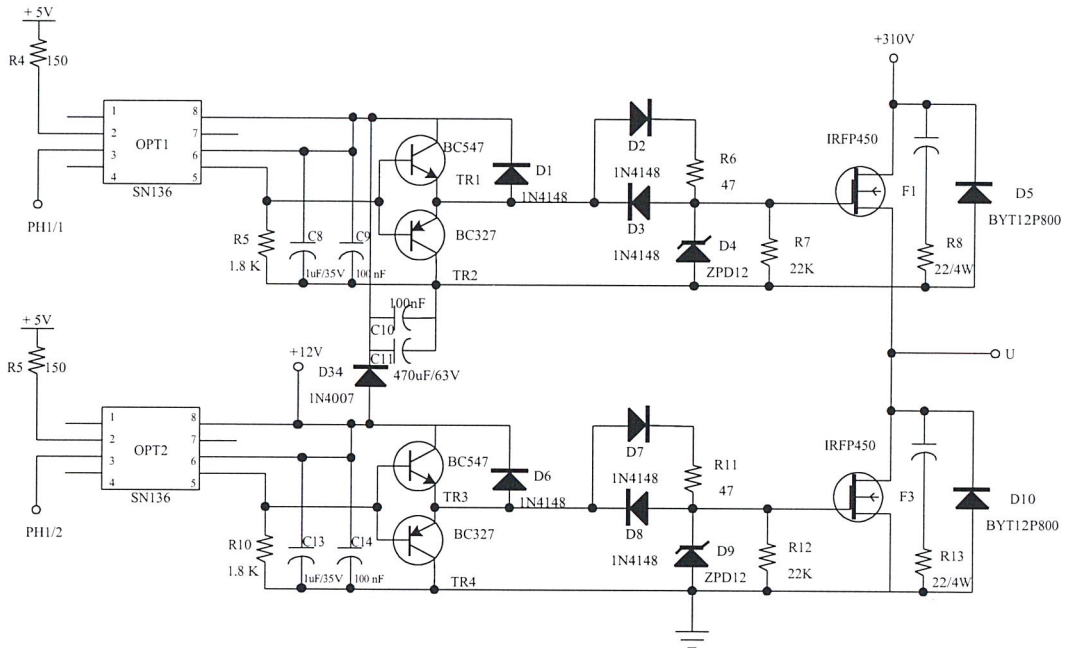
การออกแบบ เนื่องจากมอเตอร์ที่ใช้เป็นมอเตอร์ 3 เฟส มีขนาด 1/4 แรงม้า เมื่อมอเตอร์เป็นแบบ 3 เฟส จะต้องแบ่งกำลังให้แต่ละเฟสทำงานเท่ากัน ดังนั้นแต่ละส่วนจะต้องได้รับกำลังต่ำสุด จึงจะสามารถจ่ายกำลังงานได้ตามพิกัดซึ่งเป็นตัวที่เราจะต้องนำมาใช้ในการหาอุปกรณ์อิเล็กทรอนิกส์กำลังที่จะนำมาทำหน้าที่ในการสวิตซ์

การสร้างชุดสวิตซ์ซึ่งกำลังนี้จะใช้มอสเฟตทำหน้าที่เป็นอุปกรณ์สวิตซ์ เนื่องจากสามารถที่จะทำการไปอัสได้ง่าย ไม่ลำบากในการจัดวงจร มีความเร็วในการสวิตซ์สูง และมี Time Recovery ต่ำ เมื่อเทียบกับไบโพลาร์ทรานซิสเตอร์ โดยเฉพาะชุดที่ใช้ในการขับชุดสวิตซ์ ซึ่งจะเกี่ยวเนื่องไปถึงการขั้วมอเตอร์ให้สามารถทำงานได้ สามารถทำงานได้ง่ายไม่ยุ่งยาก ซึ่งวงจรสวิตซ์ที่ใช้ในโครงการนี้เป็นแบบฮาล์ฟบริดจ์ โดยใช้ทั้งหมด 3 ชุด ดังรูปที่ 3.2



รูปที่ 3.2 วงจรสวิตซ์แบบฮาล์ฟบริดจ์

ในโครงการจะใช้มอสเฟตเบอร์ IREP 450 เป็นตัวสวิตซึ่ง โดยสามารถแสดงรูปแบบวงจรสวิตซึ่งที่ ใช้ขั้วมอเตอร์ได้จริง ทั้งภาคหน้าและภาคหลัง ได้ดังรูปที่ 3.3 ซึ่งในที่นี้จะแสดงเพียงเฟสเดียว และ อีกสองเฟสจะมีลักษณะเหมือนกัน



รูปที่ 3.3 วงจรชุดขั้วมอเตอร์และสวิตซึ่งกำลัง

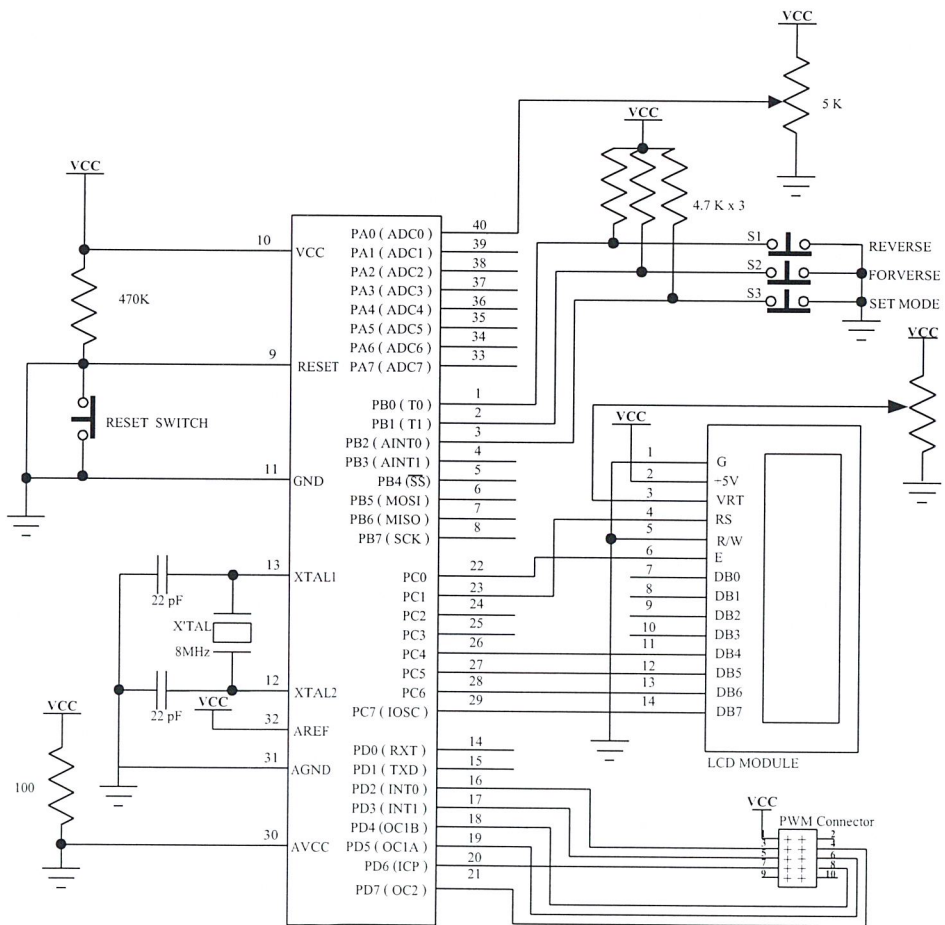
การสร้างชุดสวิตซึ่งจะออกแบบให้ชุดขั้วมอเตอร์แยกกันทางไฟฟ้ากับชุดสร้างสัญญาณ PWM โดยใช้อุปกรณ์ ออปโต (Opto) เป็นตัวแยก ซึ่งข้อมูลจาก PWM จะมีขนาดแรงดันเปลี่ยนแปลงอยู่ระหว่าง 0 ถึง 5 โวลต์ ส่วนภายในชุดสวิตซึ่งที่ขาเกตของเพาเวอร์มอสเฟต จะเปลี่ยนแปลงแรงดันระหว่าง 0 ถึง 12 โวลต์ ตามสัญญาณจาก PWM

จากรูปที่ 3.3 เนื่องจากความต้านทานระหว่างขาเกตและขาซอร์ส มีความต้านทานสูงมาก จึงสามารถเทียบแรงดันไบอัสระหว่างมอสเฟตในกิ่งเดียวกันของวงจรฮัลไฟบริดจ์กับกราวนด์ได้

การป้องกันมอสเฟตไม่ให้เกิดความเสียหาย เนื่องจากแรงดันป้อนกลับ จะใช้ไดโอดความเร็วสูงต่อคร่อมกับตัวมอสเฟต ซึ่งคุณสมบัติของไดโอดจะต้องทนกระแสได้สูง และ Time Recovery ต่ำเช่นกัน

3.3 วงจรควบคุมและสร้างสัญญาณ PWM

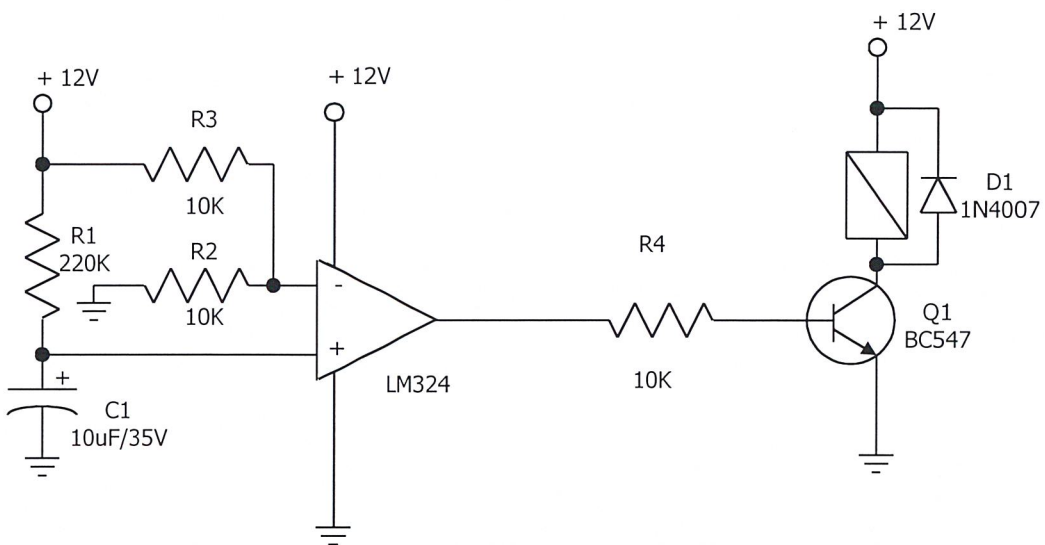
จากวงจรรูปที่ 3.4 จะใช้ไมโครคอนโทรลเลอร์ AT90S8535 จะเป็นตัวควบคุมการทำงานทั้งหมด ซึ่งจะรับข้อมูลจากสวิตช์ประกอบด้วยสวิตช์ Forward, Reverse และ Set Mode โดยสวิตช์ Forward และ Reverse จะเป็นสวิตช์ควบคุมทิศทางการหมุนของมอเตอร์ ส่วนสวิตช์ Set Mode จะเป็นสวิตช์ตั้งค่าฟังก์ชันการทำงาน และสามารถควบคุมความเร็วของมอเตอร์ได้โดยการปรับค่าแรงดันไฟฟ้าจากตัวต้านทานปรับค่าได้ ไมโครคอนโทรลเลอร์จะแปลงค่าแรงดันไฟฟ้าซึ่งเป็นสัญญาณอนาลอกเป็นสัญญาณดิจิทัล โดยใช้วงจร A/D ซึ่งอยู่ในตัวไมโครคอนโทรลเลอร์ เมื่อไมโครคอนโทรลเลอร์ได้รับสัญญาณอินพุทเหล่านี้จะนำสัญญาณที่ได้ไปประมวลผลเพื่อสร้างสัญญาณ PWM ไปควบคุมวงจรขับมอเตอร์ ในส่วนของการแสดงผลจะใช้ LCD ในการแสดงผล โดยไมโครคอนโทรลเลอร์จะมีวงจรขับ LCD อยู่ในตัวไมโครคอนโทรลเลอร์ โดยจะแสดงผลการควบคุมความเร็วและทิศทางการหมุนของมอเตอร์ และแสดงการตั้งค่าฟังก์ชันการทำงาน



รูปที่ 3.4 วงจรสร้างสัญญาณ PWM

3.4 วงจรป้องกันแรงดันกระชาก

ในส่วนของวงจรเรกติไฟเออร์ จะป้องกันในกรณีของการจ่ายไฟในวงจรแหล่งจ่ายไฟฟ้ากระแสตรง 310 โวลต์ให้แก่วงจร ในตอนแรกกรณีตัวเก็บประจุค่า 10 ไมโครฟารัด 35 โวลต์ ทำการชาร์จประจุถึงกระแสลมมาก เราจะทำการหน่วงเวลาไว้ก่อน โดยให้กระแสผ่านรีซิสเตอร์ค่า 220 โอห์ม ก่อนแล้วค่อยให้รีเลย์ตัดไฟเข้าวงจร 100 % ซึ่งสามารถป้องกันทั้งแรงดันที่กระชากเข้าวงจรมอเตอร์เร็วเกินไปด้วย ซึ่งพิจารณาออกแบบได้ดังรูปที่ 3.5



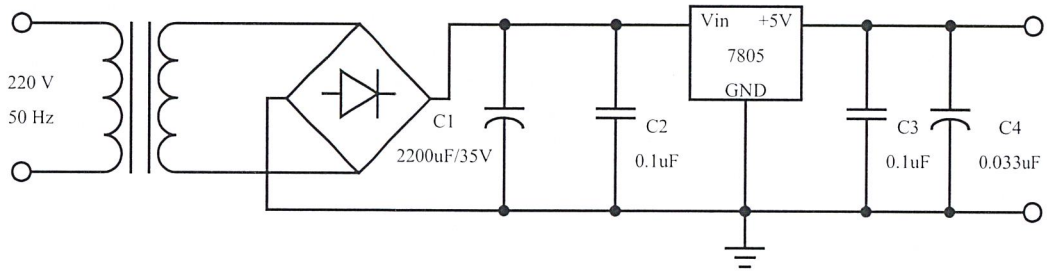
รูปที่ 3.5 วงจรป้องกันแรงดันกระชากที่จะเกิดกับมอเตอร์

3.5 วงจรแหล่งจ่ายไฟ

การออกแบบวงจรแหล่งจ่ายไฟจะแบ่งออกเป็น 3 ชุดหลักๆ ดังนี้
 แหล่งจ่ายไฟฟ้ากระแสตรง 5 โวลต์ ใช้จ่ายไฟให้ชุดไมโครคอนโทรลเลอร์ และชุดสร้างสัญญาณ PWM
 แหล่งจ่ายไฟฟ้ากระแสตรง 12 โวลต์ ใช้จ่ายไฟให้ชุดขับเพาเวอร์มอเตอร์
 แหล่งจ่ายไฟฟ้ากระแสตรง 310 โวลต์ ใช้จ่ายไฟให้เพาเวอร์มอเตอร์ เพื่อส่งถ่ายกำลังงานไปยังมอเตอร์ โดยการสวิตชิงอินเวอร์เตอร์

3.5.1 วงจรแหล่งจ่ายไฟกระแสตรง 5 โวลต์

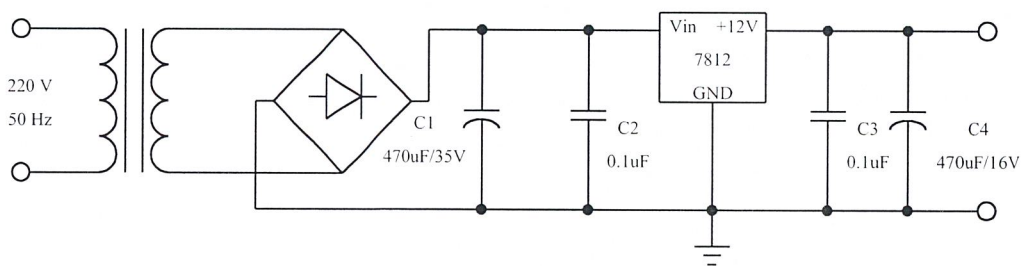
แหล่งจ่ายไฟฟ้ากระแสตรง 5 โวลต์ จะเป็นแหล่งจ่ายไฟที่ใช้จ่ายไฟให้ชุดไมโครคอนโทรลเลอร์ โดยการสร้างแหล่งจ่ายไฟนี้ จะจ่ายกระแสได้สูงสุดประมาณ 1 แอมแปร์ และใช้ไอซีเบอร์ 7805 เป็นตัวจำกัดแรงดันให้คงที่ โดยแหล่งจ่ายไฟฟ้ากระแสตรงจะออกแบบ วงจรได้ดังรูปที่ 3.6



รูปที่ 3.6 วงจรแหล่งจ่ายไฟฟ้ากระแสตรง 5 โวลต์

3.5.2 วงจรแหล่งจ่ายไฟกระแสตรง 12 โวลต์

การออกแบบวงจรแหล่งจ่ายไฟฟ้ากระแสตรง 12 โวลต์ จะใช้เป็นแหล่งจ่ายแรงดันให้กับชุดเพาเวอร์มอสเฟตและชุดวงจรป้องกัน ซึ่งวงจรชุดนี้ อุปกรณ์ที่ใช้ในวงจรไม่มากจึงกินกระแสต่ำ ซึ่งใช้ไอซีเร็กกูเลเตอร์ เบอร์ 7812 เพียงตัวเดียวในการจำกัดแรงดัน ซึ่งสามารถออกแบบวงจรได้ดังรูปที่ 3.7

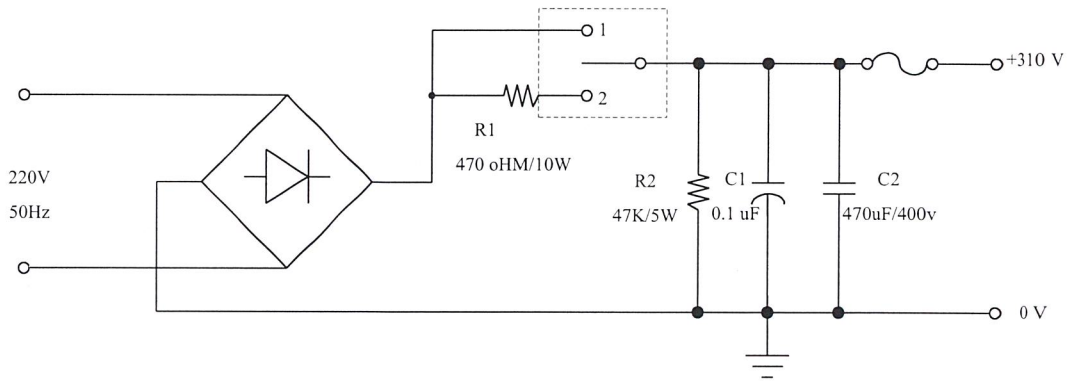


รูปที่ 3.7 วงจรแหล่งจ่ายไฟฟ้ากระแสตรง 12 โวลต์

3.5.3 วงจรแหล่งจ่ายไฟกระแสตรง 310 โวลต์

เป็นแหล่งจ่ายไฟที่ต้องระมัดระวังในการต่อใช้งานเป็นพิเศษ เพราะมีแรงดันสูงถึง 310 โวลต์ ซึ่งใช้จ่ายแรงดันให้กับเพาเวอร์มอสเฟต โดยในวงจรจะใช้บริดจ์ซึ่งทนแรงดันได้สูงสุดถึง

400 โวลต์ และทนกระแสได้ถึง 20 แอมแปร์ และเนื่องจากเป็นไฟที่ไม่ต้องการความเร็วของแรงดันมากนัก ดังนั้นจึงใช้ตัวเก็บประจุฟิลเตอร์ที่มีค่าต่ำๆ เพียง 470 ไมโครฟารัด 400 โวลต์ โดยก่อนต่อเข้าวงจรเพาเวอร์มอสเฟต จะต่อฟิวส์ 10 แอมแปร์ไว้ก่อน เพื่อป้องกันการเกิดอันตรายที่จะเกิดขึ้นกรณีมอสเฟตเกิดการลัดวงจร ซึ่งสามารถทำการออกแบบวงจรได้ดังรูปที่ 3.8

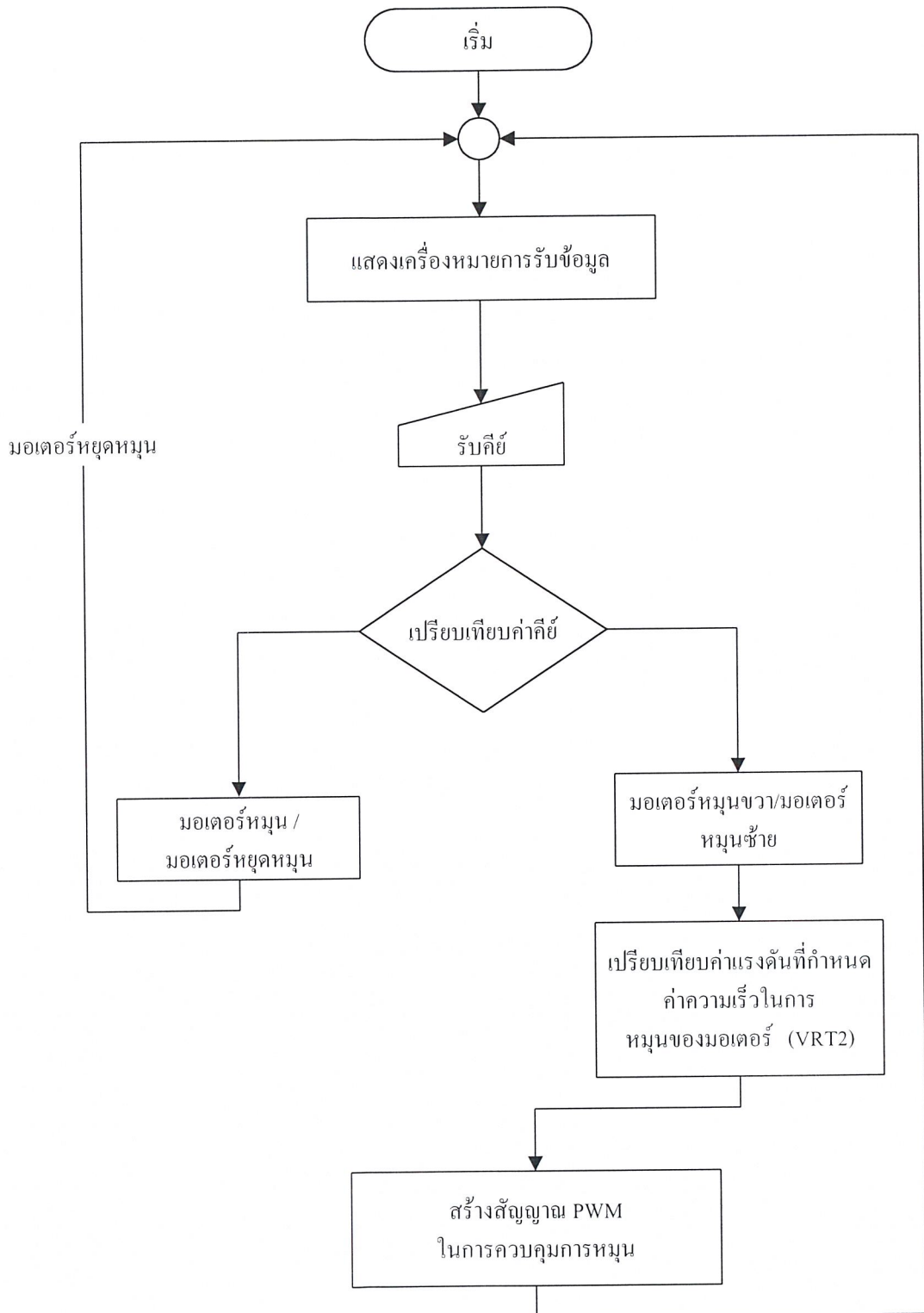


รูปที่ 3.8 วงจรแหล่งจ่ายไฟกระแสตรง 310 โวลต์

3.6 หลักการเขียนโปรแกรม

ในการเขียนโปรแกรมควบคุมเครื่องนั้น จำเป็นจะต้องรู้เรื่องฮาร์ดแวร์ และเข้าใจจุดประสงค์ที่จะทำให้เครื่องทำงาน รวมทั้งเรื่องการเขียนโปรแกรมคอมพิวเตอร์เป็นอย่างดี

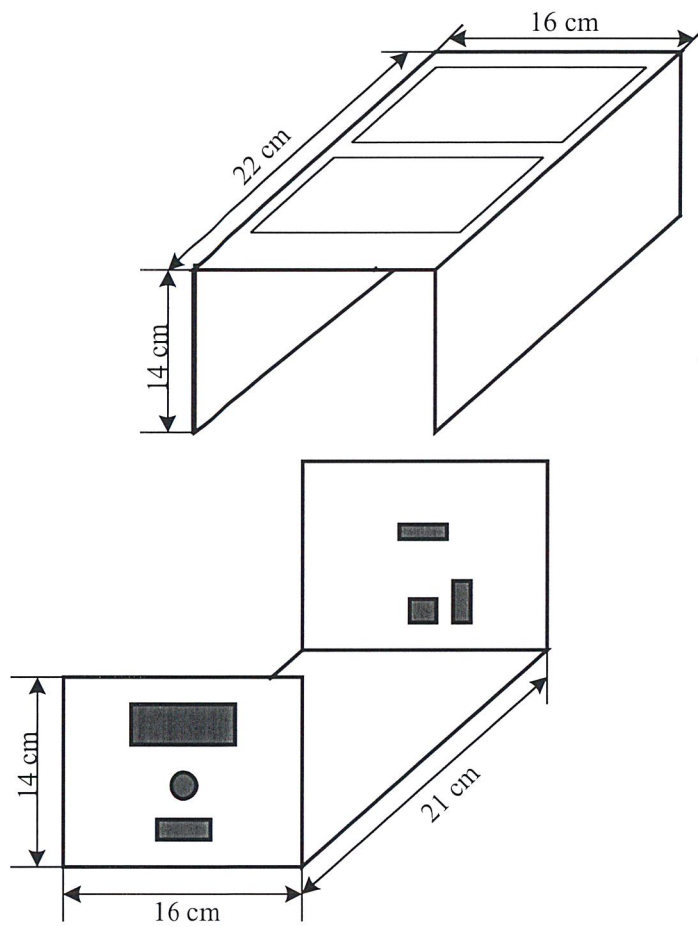
หลักในการเขียนโปรแกรมควบคุมเครื่อง คือ จะต้องสามารถควบคุมได้ทั้งความเร็วและทิศทางในการหมุนของมอเตอร์ โดยในการควบคุมนั้น โดยจะรับค่ามาจากสวิทช์และ VRT2 ที่ใช้ในการกำหนดทิศทางและความเร็วในการหมุนของมอเตอร์จากนั้นจึงมาสร้างสัญญาณ PWM ในการควบคุมการหมุนของมอเตอร์ โดยแสดงดังรูปที่ 3.9 ซึ่งเป็นผังแสดงการทำงานของโปรแกรม



รูปที่ 3.9 ผังการทำงานของโปรแกรม

3.7 การออกแบบโครงสร้างของโครงการ

ส่วนประกอบที่สำคัญอีกส่วนหนึ่งคือ โครงเครื่อง ซึ่งมีลักษณะเป็นกล่องสี่เหลี่ยมทำจากพลาสติกชนิดหนาเพื่อให้ตัวงานมีน้ำหนักเบากว่ากล่องโลหะ โดยมีขนาด กว้าง x ยาว x สูง เท่ากับ 16 x 21 x 14 เซนติเมตร โดยภายในจะบรรจุวงจรทั้งหมดของโครงการ ดังรูปที่ 3.10



รูปที่ 3.10 โครงสร้างกล่องของตัวเครื่อง

บทที่ 4

การทดลอง และผลการทดลอง

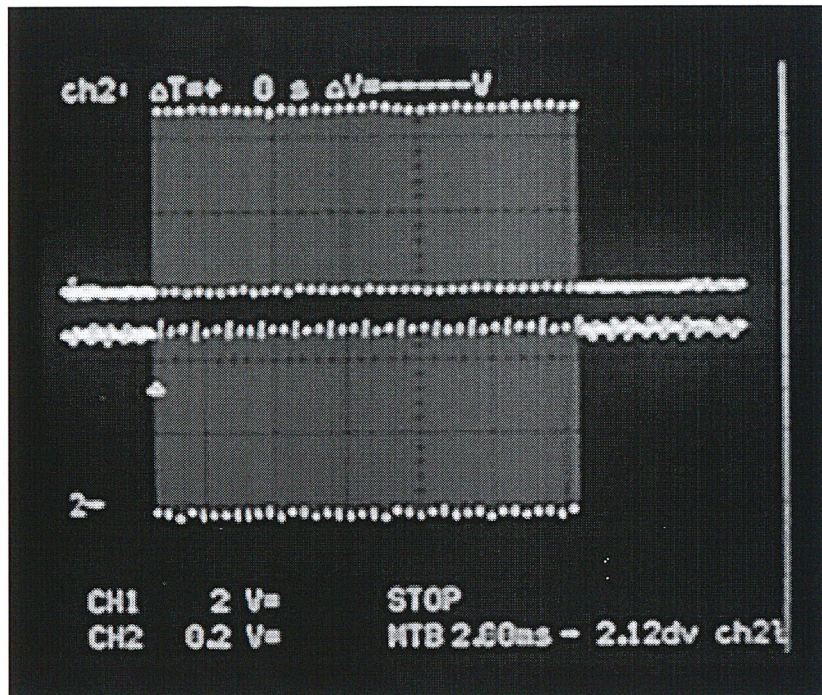
เนื้อหาในบทนี้ จะกล่าวถึงการทดลองและผลของการทดลองของส่วนประกอบในวงจรของโครงการที่สำคัญ และการทดลองทดสอบเครื่อง ซึ่งการทดลองในส่วนของวงจร Power จะต้องอาศัยความระมัดระวังและรอบคอบในการทดลอง โดยเฉพาะอย่างยิ่งวงจรในส่วนของวงจรสวิตซ์กำลัง เพราะว่าแรงดันที่ใช้มีค่าสูงมาก โดยการทดลองจะมีรูปคลื่นสัญญาณตามจุดต่างๆ ที่ทดสอบบันทึกเป็นผลการทดลอง เพื่อเปรียบเทียบกับสัญญาณที่ได้ โดยจะกำหนดความถี่สวิตซ์ซึ่งสามารถปรับได้จากความต้านทานปรับค่าได้ ในย่านประมาณ 288 เฮิร์ตซ์ ถึง 2.5 กิโลเฮิร์ตซ์ โดยรูปแบบของความถี่หลักมูลจะเปลี่ยนแปลงไปตามความถี่สวิตซ์โดยอัตโนมัติ โดยใน 1 ความถี่หลักมูลจะมีช่วง On ของความถี่สวิตซ์ 48 ครั้งเสมอ ผลคือทำให้ความถี่หลักมูลเปลี่ยนแปลงจากประมาณ 10 เฮิร์ตซ์ถึง 50 เฮิร์ตซ์ อุปกรณ์และเครื่องมือที่ใช้ในการทดลองคือ ออสซิลโลสโคป 100 เมกะเฮิร์ตซ์ 1 เครื่อง, มัลติมิเตอร์ดิจิตอล 1 เครื่อง, มัลติมิเตอร์แอนะล็อก 1 เครื่อง และวงจรที่ได้ออกแบบไว้

4.1 การทดลองวงจรแหล่งจ่ายไฟฟ้ากระแสตรง

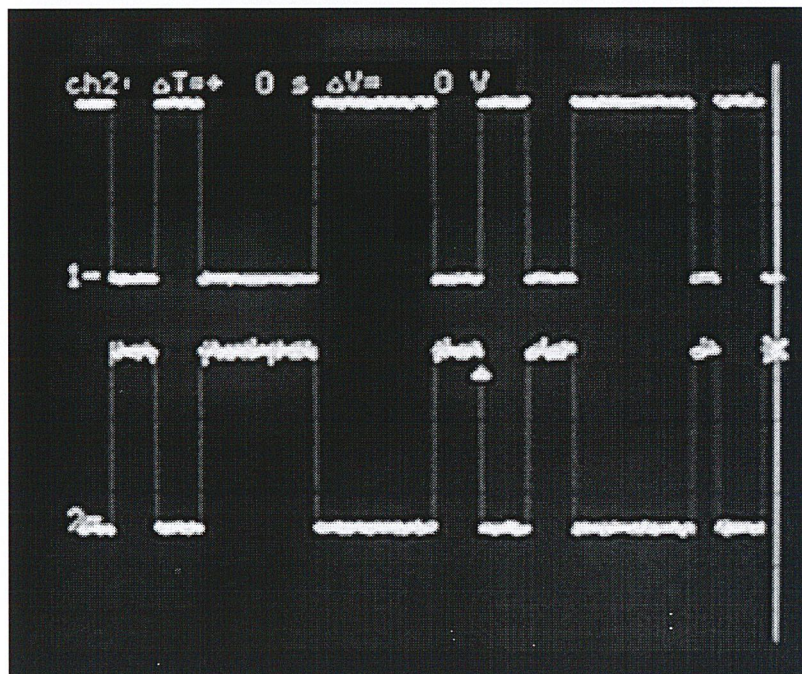
ผลการทดลองแหล่งจ่ายทั้ง 3 แหล่งจ่าย สามารถที่จะให้เอาต์พุต ได้ตรงตามที่ออกแบบไว้ คือ 5 โวลต์, ± 12 โวลต์ และ 310 โวลต์

4.2 การทดลองวงจร PWM

การกำเนิดสัญญาณ PWM ซึ่งจะผลิตสัญญาณ PWM จากไมโครคอนโทรลเลอร์ โดยให้เอาต์พุตออกมาเป็น 3 เฟส แตกต่างกัน 120 องศา ซึ่งแต่ละเฟสจะแบ่งออกเป็น $phx/1$ และ $phx/2$ โดยวิธีการกำเนิดสัญญาณ จะใช้ไมโครคอนโทรลเลอร์ เป็นตัวควบคุมสัญญาณที่ออกมา ในการทดลองนี้จะแสดงผลการทดลองของ $ph1/1$, $ph2/2$ และ $ph1/1$ และ $ph3/1$ เพื่อให้เห็นความต่างเฟสและความถี่สวิตซ์ที่มีผลที่รูปคลื่น PWM รวมทั้ง $ph1/1$, $ph2/2$ เพื่อให้เห็นลักษณะการตรงกันข้ามของแรงดันในการไปอัสมอสเฟต



รูปที่ 4.1 ความถี่สวิตซ์ซึ่งที่มีประมาณ 48 พัลส์ใน 1 ความถี่หลักมูล

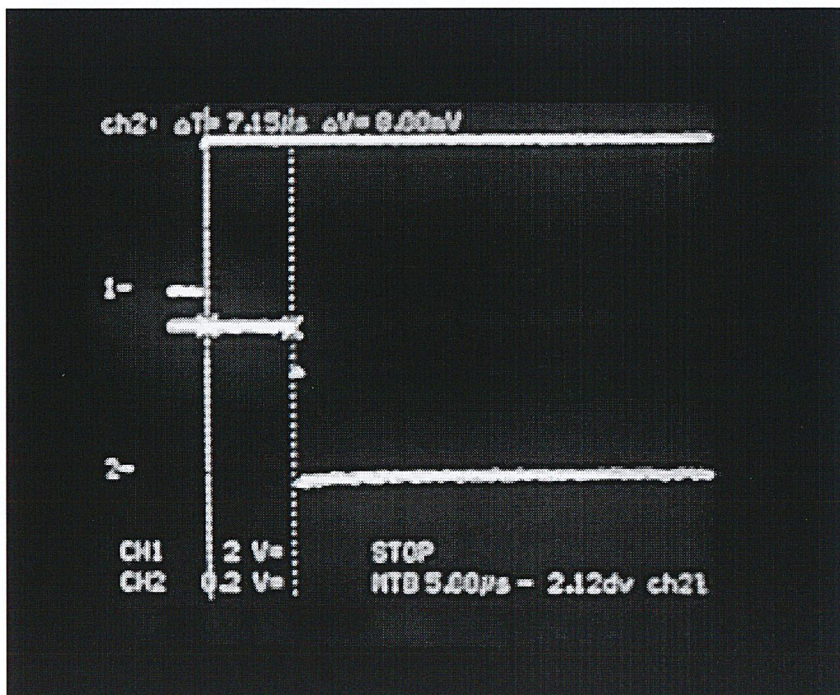


รูปที่ 4.2 ลักษณะการตรงกันข้ามของแรงดันที่ไบอัสขาเกตของมอสเฟตในเฟส 1/1 และ 1/2 ใน
กึ่งเดียวกัน

4.3 การทดลอง Deadtime

เนื่องจากสวิตซ์ที่ใช้เป็นแบบ ฮาล์ฟบริดจ์อินเวอร์เตอร์ จะใช้มอสเฟต จำนวน 2 ตัวต่ออนุกรมกันในกิ่งเดียวกัน โดยเอาต์พุตจะต่อออกที่จุดกึ่งกลางระหว่างมอสเฟตทั้ง 2 ตัวและต่อแรงดันขนาด 310 โวลท์ ที่ด้านหนึ่ง และอีกด้านหนึ่งต่ออยู่กับกราวด์ ดังนั้นจะต้องไม่ให้มอสเฟตแต่ละตัว Turn On พร้อมกัน เพราะอาจทำให้เพาเวอร์มอสเฟตพังได้เนื่องจากการเกิดการช็อตทอล์ก

ดังนั้นจึงได้มีการกำหนด Deadtime ซึ่งเป็นค่าที่กำหนดช่วงห่างของการ Turn On ของมอสเฟตแต่ละตัว ซึ่งใช้แรงดันบวกทั้งคู่การกำหนด Turn On นั้น สามารถกำหนดค่าได้จาก 0-15 ที่ไมโครคอนโทรลเลอร์ โดยที่ค่า "0" หมายถึงไม่มีค่าเวลา DeadTime และ "15" จะมีค่า DeadTime สูงสุด 5 ไมโครวินาที ในการทดลองจะวัดที่เฟสใดเฟสหนึ่ง ใน 3 เฟส ของรูปคลื่นที่ไบอัสเพาเวอร์มอสเฟต โดยจะวัดรูปคลื่นที่เฟส 1 โดยวัดที่ ph1/1 และ ph1/2 โดยใช้ค่า Deadtime ที่ค่า 15 โดยจะวัดที่ขาเกตของเพาเวอร์มอสเฟต



รูปที่ 4.3 ค่า Deadtime สูงสุด 5 ไมโครวินาทีที่กำหนดจากไมโครคอนโทรลเลอร์

จากรูปที่ 4.3 จะเห็นว่าช่วงเวลาที่ขอบล่างและขอบข้างขึ้นของ ph1/1 และ ph1/2 ที่ขาเกตของเพาเวอร์มอสเฟตแต่ละตัว จะมีความชันที่มีลักษณะลาดขึ้นและลาดลง จะไม่มีความชันที่ได้

ตามทฤษฎี เป็นเพราะที่ขาเกิดของเพาเวอร์มอสเฟต จะมีค่าประจุสะสม ทำให้เกิดลักษณะของ Time Recovery

4.5 การทดลองความเร็วและทิศทาง

ในการทดลองความเร็วและทิศทางจะเป็นลักษณะของการทำการทดสอบเครื่อง โดยสามารถทดสอบได้เมื่อประกอบเครื่องเสร็จเรียบร้อยแล้ว

จากการทดลองสามารถปรับความเร็วต่ำสุดที่ความถี่หลักมูลประมาณ 10 เฮิร์ตซ์ และสูงสุดประมาณ 50 เฮิร์ตซ์ ส่วนในด้านการควบคุมทิศทางการหมุนก็สามารถควบคุมได้ดี

สรุปผลการทดลองบทที่ 4

จากผลการทดลองในแต่ละวงจร ให้ผลเป็นที่น่าพอใจ เมื่อนำวงจรแต่วงจรมาประกอบเข้าด้วยกัน ก็จะพบปัญหาบ้าง เช่น การชำรุดของอุปกรณ์ เกิดจากการลัดวงจร แต่ก็สามารถแก้ไขปัญหาได้ และสามารถทำงานได้ ในส่วนของการควบคุมวงจรด้วยไมโครคอนโทรลเลอร์ สามารถที่จะควบคุมวงจรได้เป็นอย่างดี โดยไม่มีปัญหามากนัก ซึ่งสามารถควบคุมการทำงานได้ตามต้องการ

การทดลองทดสอบเครื่องพบว่าสามารถที่จะควบคุมความเร็วและทิศทางได้เป็นอย่างดี เป็นไปตามเงื่อนไขของความเร็วและทิศทางที่กำหนด โดยผ่านทางโปรแกรม

บทที่ 5

บทสรุป ปัญหา แนวทางแก้ปัญหา และการพัฒนา

5.1 บทสรุป

ในการศึกษาและทดลองสร้างเครื่องควบคุมความเร็วสำหรับมอเตอร์เหนี่ยวนำ 3 เฟส เป็นแนวคิดที่จะศึกษาระบบที่มีส่วนในการใช้งานอุตสาหกรรม เพื่อพัฒนาระบบควบคุมเครื่องจักรที่ใช้ในงานอุตสาหกรรม ให้สามารถใช้งานได้มีประสิทธิภาพ การจัดสร้างโครงการนี้ต้องมีการศึกษารายละเอียดต่างๆ เป็นอย่างมาก ศึกษาถึงความเป็นไปได้ของโครงการ และการศึกษาจากการสอบถามผู้ที่มีความชำนาญในเรื่องที่เกี่ยวกับโครงการที่ทางกลุ่มจัดทำ รวมถึงการศึกษาคูงานในโรงงานอุตสาหกรรมจริง เพื่อศึกษาการทำงานของระบบต่างๆ ที่ต้องการนำมาประยุกต์ใช้ในโครงการ

การสร้างโครงการนี้ใช้แนวคิดในการออกแบบวงจรแต่ละส่วนประกอบการทำงานของเครื่องจักรที่ใช้จริงในโรงงาน เป็นการดัดแปลง ออกแบบและพัฒนาวงจร เพื่อให้สามารถนำมาใช้งานได้จริง และมีราคาถูกลงกว่าเครื่องต้นแบบ สามารถนำโครงการที่สำเร็จไปประยุกต์ใช้งานได้ต่อไป

ส่วนการศึกษากการเขียนโปรแกรม จะศึกษาในเรื่องไมโครคอนโทรลเลอร์ AT90S8535 โดยละเอียด เพื่อสามารถนำมาควบคุมการทำงานของวงจรทางด้านฮาร์ดแวร์ได้ และส่วนสำคัญของการควบคุมมอเตอร์ คือการสร้างการมอดูเลตความกว้างของพัลส์ โดยที่ไมโครคอนโทรลเลอร์ AT90S8535 สามารถสร้างสัญญาณ PWM ได้โดยไม่ต้องใช้ไอซีสร้างสัญญาณ

จากการศึกษาและจัดสร้างโครงการนี้สำเร็จ ทำให้สามารถเข้าใจหลักการการทำงานของมอเตอร์เหนี่ยวนำกระแสสลับ 3 เฟส และเข้าใจการเขียนโปรแกรมเพื่อควบคุมมอเตอร์ให้ทำงานตามต้องการ การจัดสร้างโครงการนี้ได้เครื่องต้นแบบสมบูรณ์ตามวัตถุประสงค์ ความสามารถของโครงการ คือ สามารถควบคุมความเร็วของมอเตอร์ 3 เฟส ให้หมุนช้าหรือเร็วได้ตามกำหนด, สามารถควบคุมทิศทางการหมุนของมอเตอร์ 3 เฟสได้ โดยสามารถนำโครงการที่ศึกษานี้ไปประยุกต์ใช้งานอุตสาหกรรมได้ต่อไป เช่น ควบคุมมอเตอร์เหนี่ยวนำ 3 เฟสโดยต่อร่วมกับเครื่องวัดระยะกันชนหลัง (Backgauge) ในเครื่องตัดเหล็กหรือพับเหล็กได้

5.2 ปัญหาและแนวทางแก้ไข

1.) ปัญหาที่พบทางด้านฮาร์ดแวร์

1.1) มีความยากลำบากในการออกแบบตัวโครงสร้างของเครื่อง เกิดจากความไม่แน่นอนว่าจะนำไปใช้งานจริงกับเครื่องจักรชนิดใด เพราะสามารถนำไปใช้กับเครื่องจักรได้หลายชนิด จึงไม่สามารถที่จะสร้างโครงสร้างของเครื่องให้สามารถรองรับเครื่องจักรหลายๆ ประเภทได้

แนวทางการแก้ไข

ได้ปรึกษากับผู้มีความเชี่ยวชาญ ในด้านเครื่องจักรอุตสาหกรรม ได้แนะนำว่าในการออกแบบจะต้องเน้นเฉพาะเจาะจงที่เครื่องจักรเครื่องใดเครื่องหนึ่งเท่านั้น จึงจะทำให้การออกแบบทำได้ง่ายขึ้น และสามารถรองรับการทำงานของเครื่องจักรนั้นๆ ได้ประสิทธิภาพสูงสุด

1.2) อุปกรณ์บางตัวหาซื้อได้ยาก เพราะงานที่ทำเป็นงานเฉพาะด้าน

แนวทางการแก้ไข

ใช้อุปกรณ์ตัวเก่าที่มีมาใช้ทดแทนกันได้ถ้าคุณสมบัติในการทำงานมีความคล้ายคลึงกัน และสภาพการทำงานอยู่ในสภาพสมบูรณ์ หรือใช้อุปกรณ์ตัวใหม่ที่สามารถดัดแปลงให้ใช้งานได้ตามคุณสมบัติที่ต้องการ

1.3) การศึกษาแนวทางในการทำโครงงานนี้มีความยากลำบาก ในการเข้าไปศึกษาในโรงงานจริง เพราะโรงงานอุตสาหกรรมส่วนมากจะไม่เปิดเผยเรื่องเทคโนโลยี

แนวทางการแก้ไข

อาจติดต่อผ่านทางกลุ่มคนที่เรารู้จัก ส่วนอีกวิธีคือ ขอความช่วยเหลือจากอาจารย์ที่ปรึกษาปริญญาโท ให้อาจารย์นำไปขอความร่วมมือบริษัทที่รู้จัก ให้ทางกลุ่มได้เข้าไปศึกษาดูงาน

2.ปัญหาที่พบด้าน Software

2.1) การเขียนโปรแกรมเพื่อควบคุมบอร์ดไมโครคอนโทรลเลอร์ชุดสำเร็จ ที่ซื้อมาจะทำด้วยความยากลำบาก บางทีก็เกิดการทำงานที่ไม่แน่นอนของบอร์ด

แนวทางการแก้ไข

ก่อนที่จะเขียนโปรแกรมควรสำรวจความเรียบร้อยของบอร์ดให้ดี ทำการสร้างส่วนของฮาร์ดแวร์ให้สามารถรองรับการต่อกับบอร์ดไมโครคอนโทรลเลอร์ได้ และสามารถนำบอร์ดนี้ไปขอรับคำแนะนำจากผู้เชี่ยวชาญและรับคำปรึกษามาเป็นแนวทางในการเขียนโปรแกรม

2.2) ในการเขียนโปรแกรมนั้นพบปัญหาในเรื่องของคำสั่งต่างๆ ที่ใช้ในการเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์เบอร์ AT90S8535 เนื่องจากเป็นไมโครคอนโทรลเลอร์เบอร์ใหม่ ทำให้ไม่สามารถเขียนโปรแกรมได้ตามที่ต้องการ

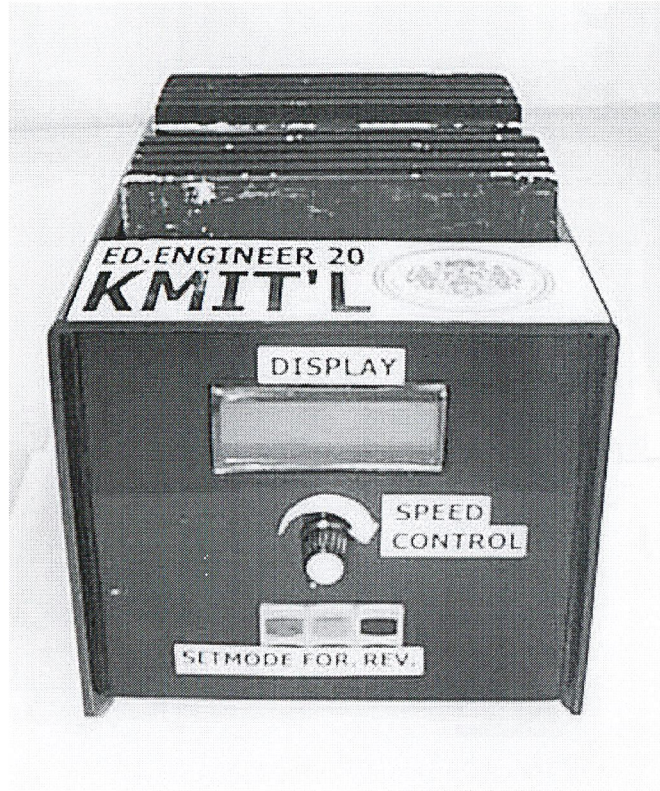
แนวทางการแก้ไข

ได้นำปัญหาไปปรึกษากับอาจารย์ที่ปรึกษา เพื่อขอคำแนะนำ และได้ติดต่อกับทางบริษัทผู้จำหน่ายเพื่อแก้ไขปัญหา

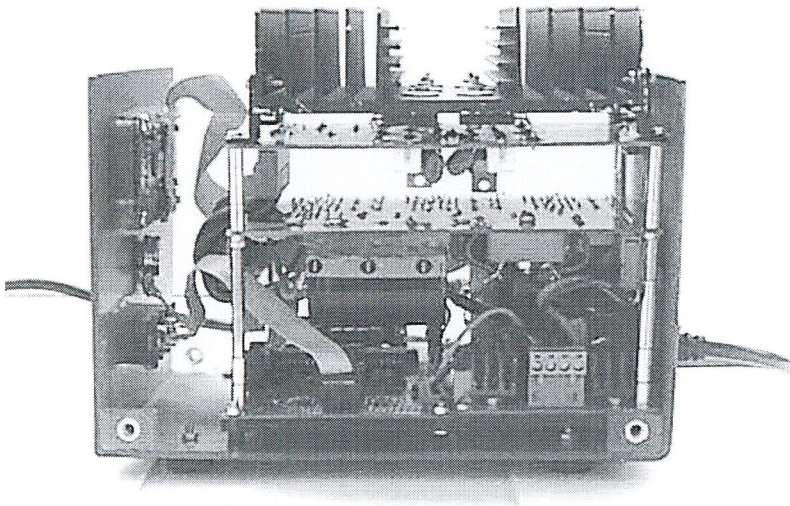
5.3 แนวทางการพัฒนา

1. พัฒนาให้สามารถใช้งานกับเครื่องจักรได้หลายชนิดขึ้น
2. พัฒนาจัดสร้างชุดกลไกให้เหมาะสมกับตัวเครื่องจักร

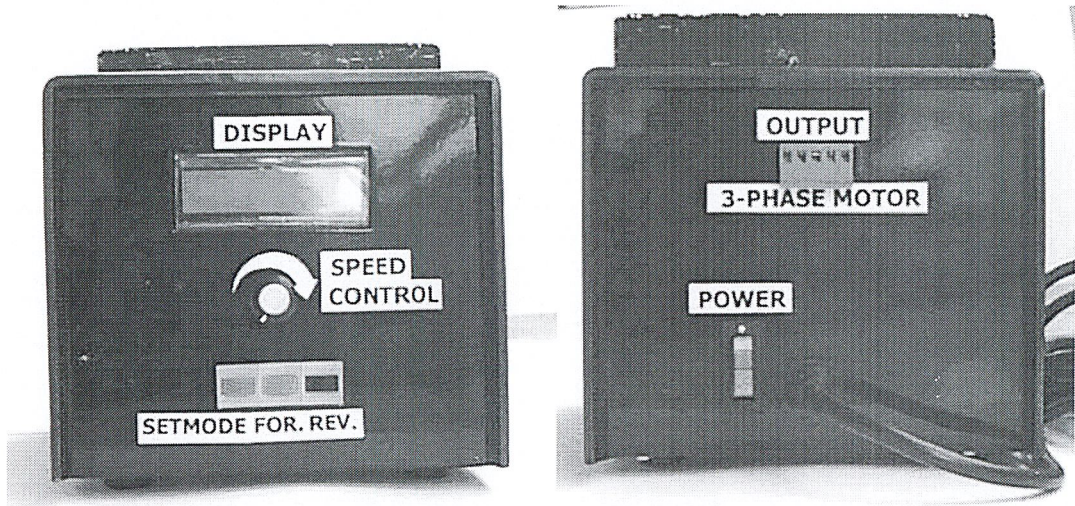
ภาคผนวก ก
เครื่องต้นแบบ



รูปที่ ก.1 เครื่องประกอบเสร็จ



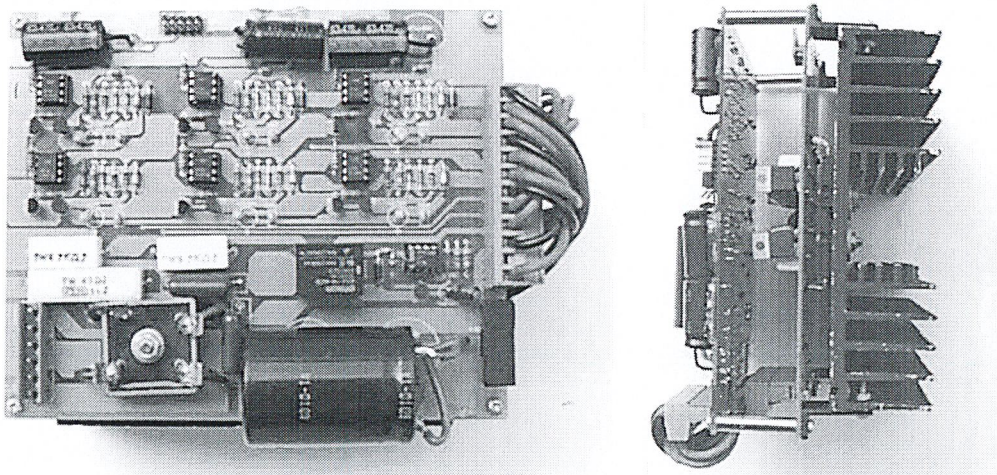
รูปที่ ก.2 การวางอุปกรณ์ภายในกล่อง



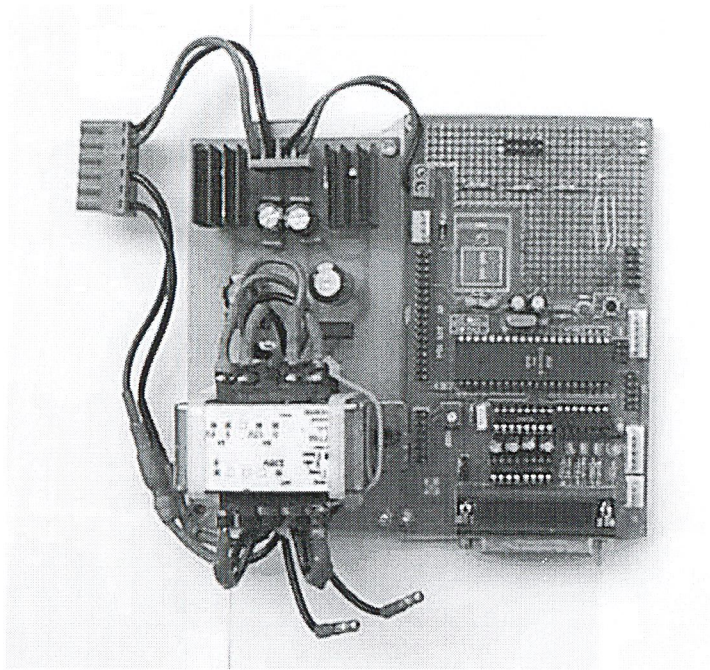
ก.) ภาพด้านหน้าของเครื่อง

ข.) ภาพด้านหลังของเครื่อง

รูปที่ ก.3 ภาพด้านหน้าและด้านหลังของเครื่อง

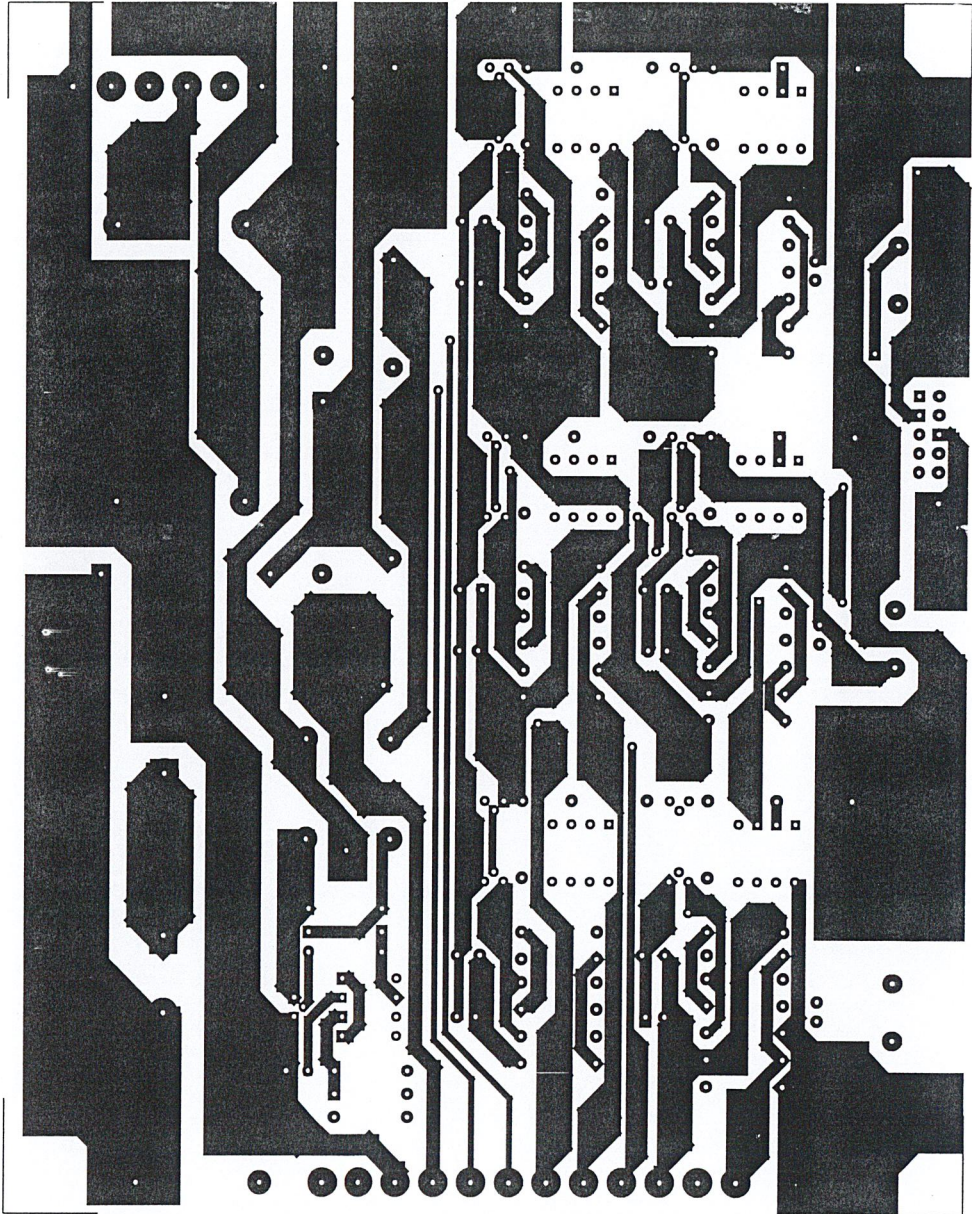


รูปที่ ก.4 ชุดขับมอเตอร์และสวิทซ์กำลัง

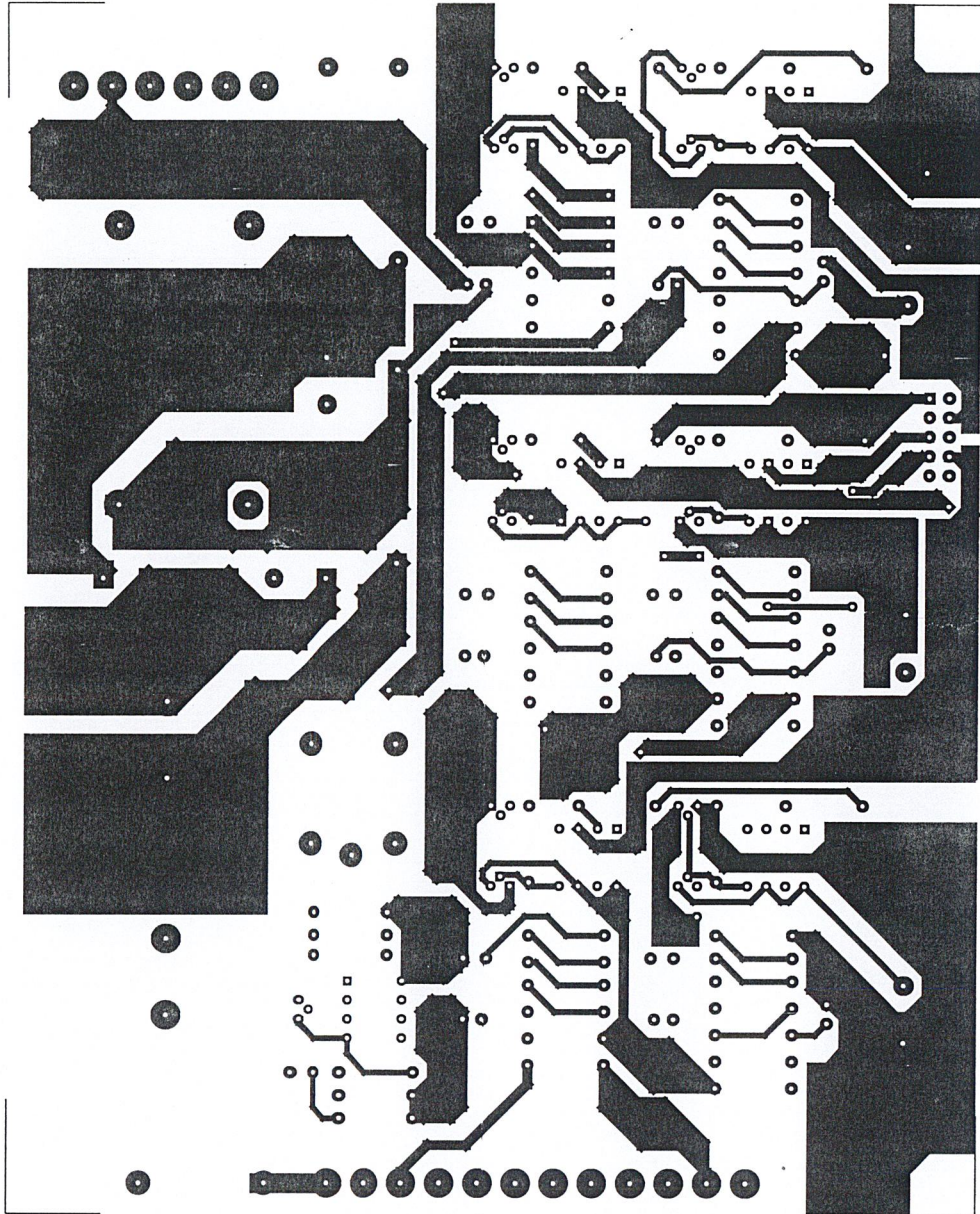


รูปที่ ก.5 ชุดควบคุมโดยใช้ไมโครคอนโทรลเลอร์และชุดเพาเวอร์ซัพพลาย

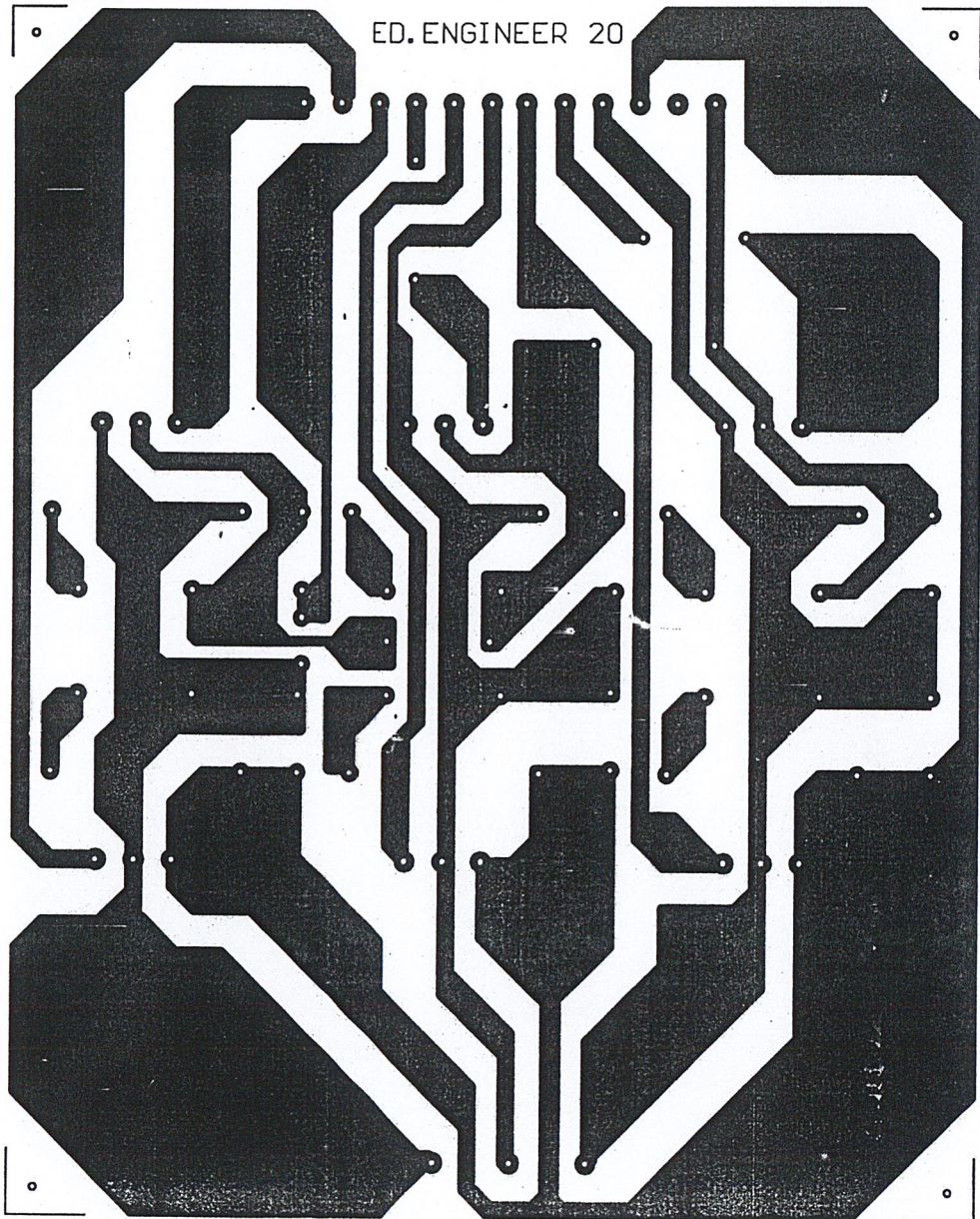
ภาคผนวก ข
วงจรและแผ่นวงจรพิมพ์



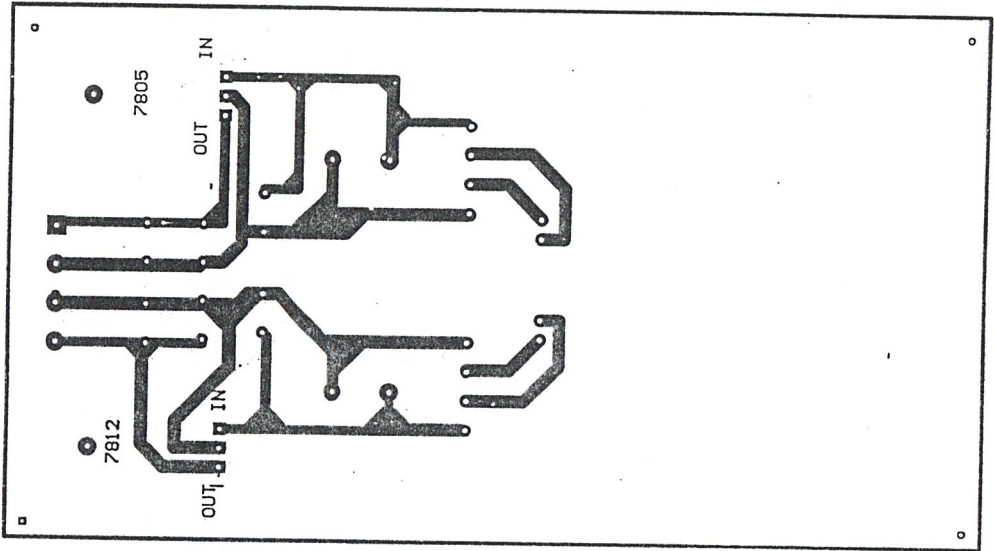
รูปที่ ข.1 ถายวงจรมพิมพ์ชุดขับมอเตอร์และสวิตซ์กำลัง (แผ่นที่ 1 ด้านบน)



รูปที่ ข.2 ลายวงจรพิมพ์ชุดขับมอเตอร์และสวิทซ์กำลัง (แผ่นที่ 1 ด้านล่าง)

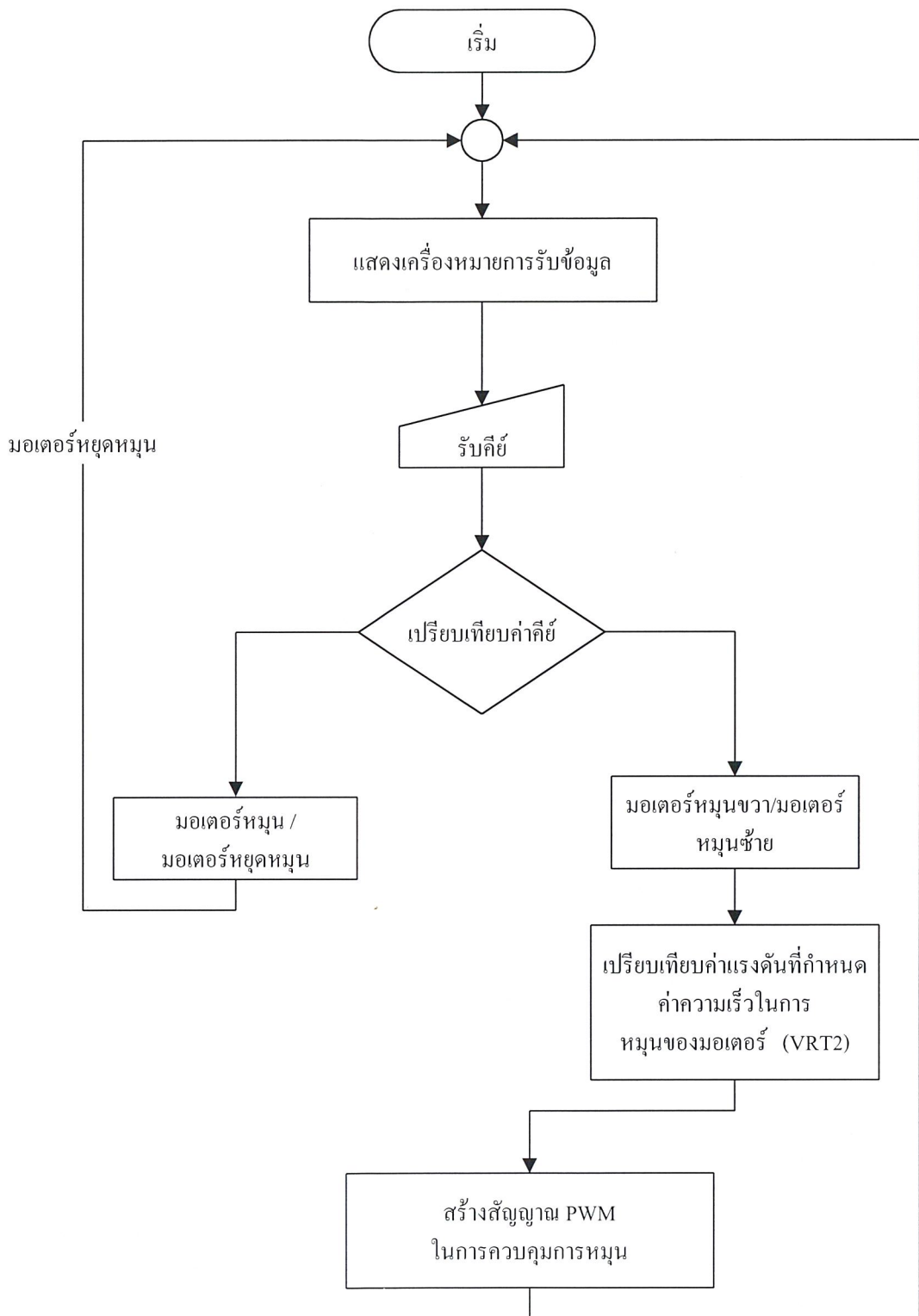


รูปที่ ข.3 ลายวงจรพิมพ์ชุดขับมอเตอร์และสวิตซ์กำลัง (แผ่นที่ 2)



รูปที่ ข.4 ตายวงจรพิมพ์ชุดภาคจ่ายไฟ

ภาคผนวก ค
ผังการทำงานและโปรแกรม



รูปที่ ข.1 ฟังก์การทำงานของโปรแกรม

```
;* File Name      : "inv.asm"  
;* Title         :  
;* Date          : 2/3/42  
;* Version       : 1.0  
;* Target MCU    : AT90S8535/AT90LS8535
```

```
.include "8535def.inc"
```

```
.org    $000  
rjmp    RESET    ;Reset Handle
```

```
.org    $009  
rjmp    TIM
```

```
.org    $00E    ;interrupt vector for ADC  
rjmp    ATOD
```

```
.def    tmp1 = r16  
.def    tmp2 = r17  
.def    hiad = r18  
.def    load = r19
```

```
; Start program
```

```
RESET:    ldi    tmp1,low(RAMEND)  
          out    SPL,tmp1    ;* init Stack Pointer Low  
          ldi    tmp1,high(RAMEND)  
          out    SPL+1,tmp1    ;* init Stack Pointer High  
          rcall  delay    ;* delay before initial display  
          rcall  delay
```

```

ldi r29,0xff      ;* Config port C for LCD
out  DDRC,r29

ldi r29,0xff
out  PORTC,r29

rcall initlcd     ;* init LCD

ldi r21,0x00      ;* Home position
rcall gotopos

ldi ZH,high(str1*2) ;* Show : 3 Phase Inverter
ldi ZL,low(str1*2)
rcall text

ldi r21,0x40      ;* Home position
rcall gotopos

ldi ZH,high(str2*2) ;* Show : Frequency :
ldi ZL,low(str2*2)
rcall text

sei              ;* clear global interrupt

ldi tmp1,0x00     ;* set porta as Hi impedance
out  DDRA,tmp1
out  PORTA,tmp1

ldi tmp1,0x00     ;* set chanl 1 ADC
out  ADMUX,tmp1

ldi tmp1,0b00011000 ;* set sampling rate
out  ADCSR,tmp1

```

```

sbi  ADCSR,ADEN      ;* enable ADC

ldi  tmp1,0xff      ;* set portd as output
out  DDRD,tmp1

ldi  tmp1,0xb0
out  PORTD,tmp1

ldi  tmp1,0x04      ;* Set timmer 0
out  TCCR0,tmp1

di   tmp1,0x01      ;* Enable timmer 0
out  TIMSK,tmp1

adcloop1:  bi  ADCSR,ADSC      ;* start conversion

ldi  r21,0x4a      ;* set position
rcall gotopos

andi  hiad,0x03      ;detect 3 bit low of ADCH
ldi  ZH,high(HEXTBL*2) ;get high of pointer
ldi  ZL,low(HEXTBL*2) ;get low of pointer
add  r30,hiad
ldi  hiad,0
adc  r31,hiad
lpm          ;get data from program memory
mov  r21,r0      ;put data to rs232
rcall wrlcd

push  load

```

```

swap load
andi load,0x0F ;detect 4 bit high of ADCL
ldi ZH,high(HEXTBL*2) ;get high of pointer
ldi ZL,low(HEXTBL*2) ;get low of pointer
add r30,load
ldi load,0
adc r31,load
lpm ;get data from program memory
mov r21,r0 ;put data to rs232
rcall wrlcd

pop load

andi load,0x0F ;detect 4 bit low of ADCL
ldi ZH,high(HEXTBL*2) ;get high of pointer
ldi ZL,low(HEXTBL*2) ;get low of pointer
add r30,load
ldi load,0
adc r31,load
lpm ;get data from program memory
mov r21,r0 ;put data to rs232
rcall wrlcd

rjmp adcloop1

ss: rjmp ss

;* Display text string to LCD
;* input : Z point to start of string

```

```

text:      lpm
           mov  r21,r0
           cpi  r21,0
           breq texto
           rcall wrlcd
           ld   r21,Z+
           rjmp text
texto:     ret

```

```

;* Set cursor position

```

```

;* input : r21

```

```

gotopos:  sbr  r21,0x80
           rcall wrins
           ret

```

```

;* Initial LCD

```

```

initlcd:  cbi  PORTC,1
           ldi  r21,0x33      ;set DL = 1 3-time
           rcall wrins
           ldi  r21,0x32      ;clear DL = 0;1-time
           rcall wrins
           ldi  r21,0x28      ;Function set
           rcall wrins      ;DL = 0 4 bit,N = 1 2 line
           ldi  r21,0x0c
           rcall wrins
           ldi  r21,0x06
           rcall wrins
           ldi  r21,0x01

```

```
rcall wrins
```

```
ret
```

```
;* Write command to LCD
```

```
;* input : r21
```

```
wrins:    cbi   PORTC,1      ;clear RS_LCD
          mov   r22,r21
          andi  r21,0xf0
          in   r16,PORTC
          andi  r16,0x0f
          or   r16,r21
          out  PORTC,r16    ;send high byte
          rcall enlcd
          swap r22
          andi  r22,0xf0
          in   r16,PORTC
          andi  r16,0x0f
          or   r16,r22
          out  PORTC,r16    ;send low byte
          rcall enlcd
          ret
```

```
;* Write charecter to LCD
```

```
;* input : r21
```

```
wrlcd:   sbi   PORTC,1
          mov   r22,r21
          in   r16,PORTC
          andi  r16,0x0f
```

```
andi r21,0xf0
or r16,r21
out portc,r16
rcall enlcd
swap r22
in r16,PORTC
andi r16,0x0f
andi r22,0xf0
or r16,r22
out PORTC,r16
rcall enlcd
ret
```

;* Sub function of LCD

```
enlcd: cbi PORTC,0
rcall busy
sbi PORTC,0
ret
```

```
busy: ldi r25,0x10
busy2: ldi r26,0xff
busy1: dec r26
brne busy1
dec r25
brne busy2
ret
```

;* Delay function

```

delay:      ldi r28,0xff
delay2:     ldi r29,0xff
delay1:     dec r29
            brne delay1
            dec r28
            brne delay2
            ret

```

```

de:         push r29
            ldi r29,0x10
de1:        dec r29
            brne de1
            pop r29
            ret

```

```

;* Interrupt function for ADC

```

```

ATOD:      in  load,ADCL      ;read low byte
            in  hiad,ADCH     ;read high byte
            reti

```

```

;* Timmer Interrupt

```

```

;* Gennerate Pulse Width Modulate (PWM)

```

```

TIM:       in  tmp1,PORTD
            ldi tmp2,0xb0
            eor tmp1, tmp2
            out PORTD, tmp1

            rcall de

```

```

;          in    tmp1,PORTD
          ldi    tmp2,0x4f
          eor    tmp1, tmp2
          out    PORTD, tmp1

          out    TCNT0,load
          reti

HEXTBL:   .db    0x30,0X31      ;table for convert hex to ascii
          .db    0X32,0X33
          .db    0X34,0X35
          .db    0X36,0X37
          .db    0X38,0X39
          .db    0X41,0X42
          .db    0X43,0X44
          .db    0X45,0X46

pwmdata:  .db    63, 71, 79, 87, 94, 101, 108, 113, 118, 121, 124, 125
          .db    126, 125, 124, 121, 118, 113, 108, 101, 94, 87, 79
          .db    71, 63, 55, 47, 39, 32, 25, 18, 13, 08, 05, 02, 01, 00
          .db    01, 02, 05, 08, 13, 18, 25, 32, 39, 47, 55

;*        3 Phase Inverter
str1:     .db    0x33, 0x20, 0x50, 0x68, 0x61, 0x73, 0x65, 0x20
          .db    0x49, 0x6e, 0x76, 0x65, 0x72, 0x74, 0x65, 0x72, 0

;*        Frequency -
str2:     .db    0x46, 0x72, 0x65, 0x71 ,0x75, 0x65, 0x6e, 0x63, 0x79
          .db    0x20, 0x2d, 0x20, 0

```

ภาคผนวก ง

รายละเอียดและคุณสมบัติของอุปกรณ์

รายการอุปกรณ์

รายการอุปกรณ์ภาคขับเคลื่อนมอเตอร์และสวิทซ์กำลัง

ตัวต้านทาน	47 กิโลโอห์ม	6 ตัว
ตัวต้านทาน	150 กิโลโอห์ม	6 ตัว
ตัวต้านทาน	22 กิโลโอห์ม	6 ตัว
ตัวต้านทาน	22 โอห์ม / 4 วัตต์	6 ตัว
ตัวเก็บประจุ	100 นาโนฟารัด	9 ตัว
ตัวเก็บประจุ	100 นาโนฟารัด/630 โวลต์	6 ตัว
ตัวเก็บประจุ	1 ไมโครฟารัด / 35 โวลต์	6 ตัว
ตัวเก็บประจุ	470 ไมโครฟารัด / 63 โวลต์	3 ตัว
ไดโอด	1N4007	3 ตัว
ไดโอด	BYT12P800	6 ตัว
ไดโอด	1N4148	18 ตัว
ซีเนอร์ไดโอด	ZPD12	6 ตัว
ทรานซิสเตอร์	BC547	6 ตัว
ทรานซิสเตอร์	BC327	6 ตัว
มอสเฟต	IRFP450	6 ตัว
ออปโตคัปเปิล	SN136	6 ตัว

รายการอุปกรณ์วงจรควบคุมและสร้างสัญญาณ PWM

ตัวต้านทาน	4.7 กิโลโอห์ม	3 ตัว
ตัวต้านทาน	470 กิโลโอห์ม	1 ตัว
ตัวต้านทาน	100 โอห์ม	1 ตัว
ตัวต้านทาน	5 กิโลโอห์ม	1 ตัว
ตัวต้านทานปรับค่าได้	5 กิโลโอห์ม	1 ตัว
ตัวต้านทานปรับค่าได้	10 กิโลโอห์ม	1 ตัว
ตัวเก็บประจุ	22 พิโคฟารัด	2 ตัว
คริสตอล	8 เมกะเฮิร์ต	1 ตัว
ไมโครสวิทช์		1 ตัว
สวิทช์กดดับปล่อยคิด		3 ตัว
ไอซี	AT90S/LS8535	1 ตัว
ชุดแสดงผล LCD		1 ตัว

รายการอุปกรณ์แหล่งจ่ายไฟฟ้ากระแสตรง 5 โวลต์

ตัวต้านทาน	470 โอห์ม / 5 วัตต์	1 ตัว
ตัวเก็บประจุ	2200 ไมโครฟารัด / 35 โวลต์	1 ตัว
ตัวเก็บประจุ	0.1 ไมโครฟารัด	2 ตัว
ตัวเก็บประจุ	0.033 ไมโครฟารัด	1 ตัว
ไดโอดบริดจ์เรกติไฟร์	100 โวลต์ / 1 แอมแปร์	1 ตัว
ไอซี	7805	1 ตัว
หม้อแปลง 5 โวลต์ 1 แอมแปร์		1 ตัว

รายการอุปกรณ์แหล่งจ่ายไฟฟ้ากระแสตรง 12 โวลต์

ตัวเก็บประจุ	470 ไมโครฟารัด / 35 โวลต์	1 ตัว
ตัวเก็บประจุ	0.1 ไมโครฟารัด	2 ตัว
ตัวเก็บประจุ	470 ไมโครฟารัด / 16 โวลต์	1 ตัว
ไดโอดบริดจ์เรกติไฟร์	100 โวลต์ / 1 แอมแปร์	1 ตัว
ไอซี	7812	1 ตัว
หม้อแปลง และ 12 โวลต์ 1 แอมแปร์		1 ตัว

Features

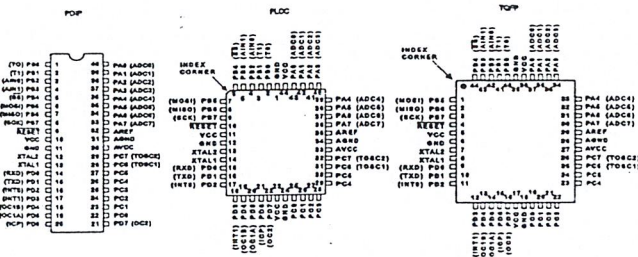
- Utilizes the AVR[®] Enhanced RISC Architecture
- AVR - High Performance and Low Power RISC Architecture
- 118 Powerful Instructions - Most Single Clock Cycle Execution
- 8K bytes of In-System Programmable Flash AT90S/LS8535
4K bytes of In-System Programmable Flash AT90S/LS4434
 - SPI Serial Interface for In-System Programming
 - Endurance: 1,000 Write/Erase Cycles
- 512 bytes EEPROM AT90S/LS8535
256 bytes EEPROM AT90S/LS4434
 - Endurance: 100,000 Write/Erase Cycles
- 512 bytes Internal SRAM AT90S/LS8535
256 bytes Internal SRAM AT90S/LS4434
- 8-Channel, 10-Bit ADC
- 32 x 8 General Purpose Working Registers
- 32 Programmable I/O Lines
- Programmable Serial UART
- V_{CC}: 4.0 - 6.0V AT90S4434/AT90S8535
V_{CC}: 2.7 - 6.0V AT90LS4434/AT90LS8535
- Speed Grades:
 - 0 - 8 MHz AT90S4434/AT90S8535
 - 0 - 4 MHz (AT90LS4434/AT90LS8535)
- Power-On Reset Circuit
- Up to 8 MIPS Throughput at 8 MHz
- RTC with Separate Oscillator and Counter Mode
- Two 8-Bit Timer/Counters with Separate Prescaler and Compare Mode
- One 16-Bit Timer/Counter with Separate Prescaler and Compare and Capture Modes
- 3 PWM channels
- External and Internal Interrupt Sources
- Programmable Watchdog Timer with On-Chip Oscillator
- On-Chip Analog Comparator
- Three Sleep Modes: Idle, Power Save, and Power Down
- Programming Lock for Software Security

Description

The AT90S4434/8535 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the AT90S4434/8535 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

(continued)

Pin Configurations



Rev. 1041A-05/98



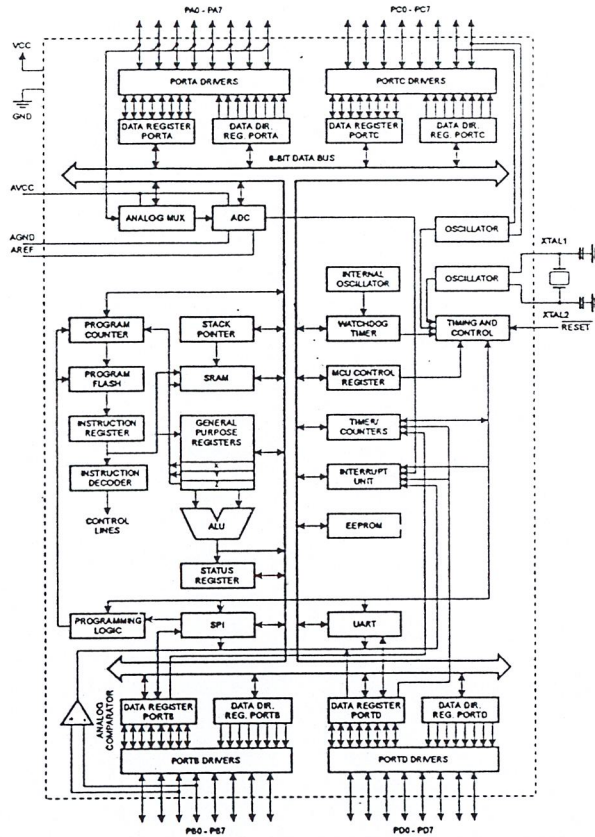
8-Bit AVR[®]
Microcontroller
with 4K/8K
Bytes In-System
Programmable
Flash

AT90S4434
AT90LS4434
AT90S8535
AT90LS8535
Advance
Information





Block Diagram



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC micro-controllers.

The AT90S4434/8535 provides the following features: 4K/8K bytes of In-System Programmable Flash, 256/512 bytes EEPROM, 256/512 bytes SRAM, 32 general purpose I/O lines, 32 general purpose working registers, RTC, three flexible timer/counters with compare modes, internal and

external interrupts, a programmable serial UART, 8-channel, 10-bit ADC, programmable Watchdog Timer with internal oscillator, an SPI serial port and three software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, timer/counters, SPI port and interrupt system to continue functioning. The Power Down mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power Save mode, the timer oscillator continues to run, allowing the user to maintain timer base while the rest of the device is sleeping.

The device is manufactured using Atmel's high density non-volatile memory technology. The on-chip ISP Flash

AT90S/LS4434 and AT90S/LS8535

allows the program memory to be reprogrammed in-system through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining an 8-bit RISC CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT90S4434/8535 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The AT90S4434/8535 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Comparison between AT90S4434 and AT90S8535

The AT90S4434 has 4K bytes of In-System Programmable Flash, 256 bytes of EEPROM, and 256 bytes of internal SRAM.

The AT90S8535 has 8K bytes of In-System Programmable Flash, 512 bytes of EEPROM, and 512 bytes of internal SRAM.

Table 1 summarizes the different memory sizes for the two devices.

Table 1. Memory Size Summary

Part	Flash	EEPROM	SRAM
AT90S4434	4K bytes	256 bytes	256 bytes
AT90S8535	8K bytes	512 bytes	512 bytes

Pin Descriptions

V_{CC}

Digital supply voltage

GND

Digital ground

Port A (PA7..PA0)

Port A is an 8-bit bi-directional I/O port. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers can sink 20mA and can drive LED displays directly. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated.

Port A also serves as the analog inputs to the A/D Converter.

Port B (PB7..PB0)

Port B is an 8-bit bi-directional I/O pins with internal pull-up resistors. The Port B output buffers can sink 20 mA. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated.

Port B also serves the functions of various special features of the AT90S4434/8535 as listed on page 52.

Port C (PC7..PC0)

Port C is an 8-bit bi-directional I/O port with internal pullup resistors. The Port C output buffers can sink 20 mA. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. Two Port C pins can alternatively be used as oscillator for Timer/Counter2.

Port D (PD7..PD0)

Port D is an 8-bit bidirectional I/O port with internal pull-up resistors. The Port D output buffers can sink 20 mA. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated.

Port D also serves the functions of various special features of the AT90S4434/8535 as listed on page 59.

RESET

Reset input. A low on this pin for two machine cycles while the oscillator is running resets the device.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier

AV_{CC}

This is the supply voltage pin for the A/D Converter. It should be externally connected to V_{CC} via a low-pass filter. See page 47 for details on operation of the ADC.

AREF

This is the analog reference input for the A/D Converter. For ADC operations, a voltage in the range AGND to AV_{CC} must be applied to this pin.

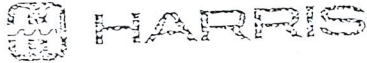
AGND

Analog ground. If the board has a separate analog ground plane, this pin should be connected to this ground plane. Otherwise, connect to GND.

Crystal Oscillators

XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or a ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. For the Timer Oscillator pins, PC6(OSC1) and PC7(OSC2), the crystal is connected directly between the pins. No external capacitors are needed. The oscillator is optimized for use with a 32,768 Hz watch crystal. An external clock signal applied to this pin goes through the same amplifier having a bandwidth of 256 kHz. The external clock signal should therefore be in the interval 0 Hz - 256 kHz.





IRFP450/451/452/453
IRFP450R/451R/452R/453R

N-Channel Power MOSFETs
 Avalanche Energy Rated*

August 1991

Features

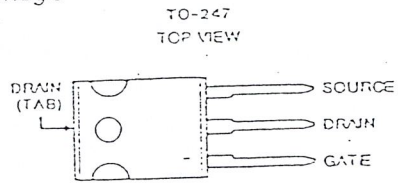
- 12A and 14A, 450V - 500V
- $r_{DS(on)} = 0.4\Omega$ and 0.5Ω
- Single Pulse Avalanche Energy Rated*
- SOA is Power-Dissipation Limited
- Nanosecond Switching Speeds
- Linear Transfer Characteristics
- High Input Impedance

Description

The IRFP450, IRFP451, IRFP452, and IRFP453 are n-channel enhancement-mode silicon-gate power field-effect transistors. IRFP450R, IRFP451R, IRFP452R and IRFP453R types are advanced power MOSFETs designed, tested, and guaranteed to withstand a specified level of energy in the breakdown avalanche mode of operation. All of these power MOSFETs are designed for applications such as switching regulators, switching converters, motor drivers, relay drivers, and drivers for high-power bipolar switching transistors requiring high speed and low gate-drive power. These types can be operated directly from integrated circuits.

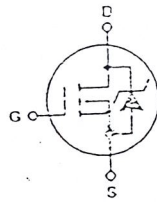
The IRFP types are supplied in the JEDEC TO-247 plastic package.

Package



Terminal Diagram

N-CHANNEL ENHANCEMENT MODE



Absolute Maximum Ratings ($T_C = +25^\circ\text{C}$). Unless Otherwise Specified

	IRFP450 IRFP450R	IRFP451 IRFP451R	IRFP452 IRFP452R	IRFP453 IRFP453R	UNITS
Drain-Source Voltage (1)	500	450	500	450	V
Drain-Gate Voltage ($R_{GS} = 20k\Omega$) (1)	500	450	500	450	V
Continuous Drain Current	14	14	12	12	A
$T_C = +25^\circ\text{C}$	14	14	12	12	A
$T_C = +100^\circ\text{C}$	8.8	8.8	7.9	7.8	A
Pulsed Drain Current (3)	56	56	48	48	A
Gate-Source Voltage	± 20	± 20	± 20	± 20	V
Maximum Power Dissipation	180	180	180	180	W
$T_C = +25^\circ\text{C}$	1.44	1.44	1.44	1.44	W/ $^\circ\text{C}$
Linear Derating Factor	52	52	45	49	A
Inductive Current, Clamped					A
(See Figure 14, $L = 100\mu\text{H}$)					A
Single Pulse Avalanche Energy Rating (4)	860	860	860	860	mJ
Operating and Storage Junction	-55 to +150	-55 to +150	-55 to +150	-55 to +150	$^\circ\text{C}$
Temperature Range					$^\circ\text{C}$
Maximum Lead Temperature for Soldering	300	300	300	300	$^\circ\text{C}$
(0.063" (1.6mm) from case for 10s)					

NOTES:

1. $T_J = +25^\circ\text{C}$ to $+150^\circ\text{C}$.
2. Pulse Test: Pulse width $\leq 300\mu\text{s}$, Duty Cycle $\leq 2\%$.
3. Repetitive rating: Pulse width limited by maximum junction temperature. See Transient Thermal Impedance Curve (Figure 5).
- *R Suffix Types Only

4. $V_{DS} = 50\text{V}$, starting $T_J = +25^\circ\text{C}$, $L = 7\text{mH}$, $R_{GS} = 25\Omega$, $I_{PEAK} = 14\text{A}$. See Figure 15.

CAUTION: These devices are sensitive to electrostatic discharge. Proper IC handling procedures should be followed.
 Copyright © Harris Corporation 1991

File Number 2331.1

IRFP450, IRFP451, IRFP452, IRFP453 IRFP450R, IRFP451R, IRFP452R, IRFP453R

Electrical Characteristics $T_C = 25^\circ\text{C}$, Unless Otherwise Specified

CHARACTERISTIC	SYMBOL	TEST CONDITIONS	LIMITS			UNITS
			MIN	TYP	MAX	
Drain-Source Breakdown Voltage IRFP450/452, IRFP450R/452R IRFP451/453, IRFP451R/453R	V_{DS}	$V_{GS} = 0V, I_D = 250\mu A$	500	-	-	V
			450	-	-	V
Gate Threshold Voltage	$V_{GS(TH)}$	$V_{DS} = V_{GS}, I_D = 250\mu A$	2.0	-	4.0	V
Gate-Source Leakage Forward	I_{GSS}	$V_{GS} = 20V$	-	-	500	nA
Gate-Source Leakage Reverse	I_{GSS}	$V_{GS} = -20V$	-	-	-500	nA
Zero Gate Voltage Drain Current	I_{DSS}	$V_{DS} = \text{Max Rating}, V_{GS} = 0V$	-	-	250	μA
		$V_{GS} = \text{Max Rating} \times 0.8, V_{GS} = 0V, T_J = +125^\circ\text{C}$	-	-	1000	μA
On-State Drain Current (Note 2) IRFP450/451, IRFP450R/451R IRFP452/453, IRFP452R/453R	$I_{D(ON)}$	$V_{DS} > I_{D(ON)} \times r_{DS(ON)} \text{ Max}, V_{GS} = 10V$	14	-	-	A
			12	-	-	A
Static Drain-Source On-State Resistance (Note 2) IRFP450/451, IRFP450R/451R IRFP452/453, IRFP452R/453R	$r_{DS(ON)}$	$V_{GS} = 10V, I_D = 7.5A$	-	0.3	0.4	Ω
			-	0.4	0.5	Ω
Forward Transconductance (Note 2)	g_{fs}	$V_{DS} \geq 50V, I_D = 7.5A$	9.3	13.8	-	S(O)
Input Capacitance	C_{ISS}	$V_{GS} = 0V, V_{DS} = 25V, f = 1.0MHz$ See Figure 10	-	2000	-	pF
Output Capacitance	C_{OSS}		-	400	-	pF
Reverse Transfer Capacitance	C_{RSS}		-	100	-	pF
Turn-On Delay Time	$t_{d(ON)}$	$V_{DD} = 250V, I_D = 14A, R_G = 6.1\Omega$	-	16	27	ns
Rise Time	t_r	See Figure 16. (MOSFET switching times are essentially independent of operating temperature.)	-	45	65	ns
Turn-Off Delay Time	$t_{d(OFF)}$		-	68	100	ns
Fall Time	t_f		-	41	60	ns
Total Gate Charge (Gate-Source + Gate-Drain)	Q_g	$V_{GS} = 10V, I_D = 14A, V_{DS} = 0.6V \text{ Max}$ Rating. See Figure 17 for test circuit. (Gate charge is essentially independent of operating temperature.)	-	82	130	nC
Gate-Source Charge	Q_{GS}		-	12	-	nC
Gate-Drain ("Miller") Charge	Q_{GD}		-	42	-	nC
Internal Drain Inductance	L_D	Measured between the contact screw on header that is closer to source end gate pins and center of center of die.				
Internal Source Inductance	L_S	Measured from the source lead, 6mm (0.25") from header and source bonding pad.				
Junction-to-Case	$R_{\theta JC}$		-	-	0.70	$^\circ\text{C/W}$
Case-to-Sink	$R_{\theta CS}$	Mounting surface flat, smooth and greased	-	0.10	-	$^\circ\text{C/W}$
Junction-to-Ambient	$R_{\theta JA}$	Free air operation	-	-	30	$^\circ\text{C/W}$

Source Drain Diode Ratings and Characteristics

Continuous Source Current (Body Diode)	I_S	Modified MOSFET symbol showing the integral reverse P-N junction rectifier.	-	-	14	A
Pulse Source Current (Body Diode) (Note 3)	I_{SM}		-	-	56	A
Diode Forward Voltage (Note 2)	V_{SD}	$T_J = +25^\circ\text{C}, I_S = 14A, V_{GS} = 0V$	-	-	1.4	V
Reverse Recovery Time	t_{rr}	$T_J = +150^\circ\text{C}, I_F = 13A, dI_F/dt = 100A/\mu s$	-	1300	-	ns
Reverse Recovered Charge	Q_{RR}	$T_J = +150^\circ\text{C}, I_F = 13A, dI_F/dt = 100A/\mu s$	-	7.4	-	μC
Forward Turn-on Time	t_{ON}	Intrinsic turn-on time is negligible. Turn-on speed is substantially controlled by $L_S + L_D$	-	-	-	-

NOTES: 1. $T_J = +25^\circ\text{C}$ to $+150^\circ\text{C}$
 2. Pulse Test: Pulse width $\leq 300\mu s$, Duty Cycle $\leq 2\%$
 3. Repetitive Rating: Pulse width limited by max. junction temperature. See Transient Thermal Impedance Curve (Figure 5)
 4. $V_{DD} = 50V$, Start $T_J = +25^\circ\text{C}$, $L = 7.6mH$, $R_{GS} = 25\Omega$, $I_{PEAK} = 14A$ (See Figure 15)

IRFP450, IRFP451, IRFP452, IRFP453 IRFP450R, IRFP451R, IRFP452R, IRFP453R

Performance Curves

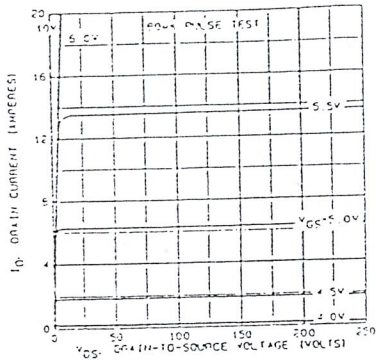


FIGURE 1. TYPICAL OUTPUT CHARACTERISTICS

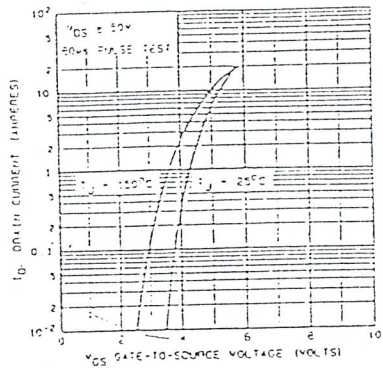


FIGURE 2. TYPICAL TRANSFER CHARACTERISTICS

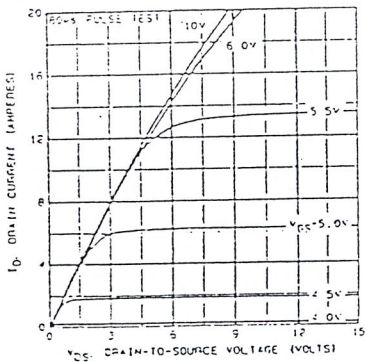


FIGURE 3. TYPICAL SATURATION CHARACTERISTICS

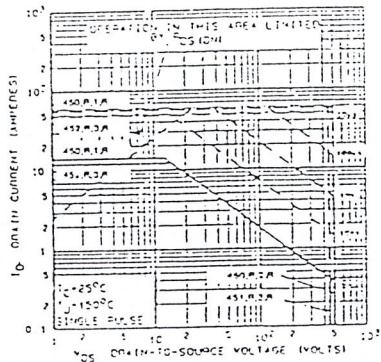


FIGURE 4. MAXIMUM SAFE OPERATING AREA

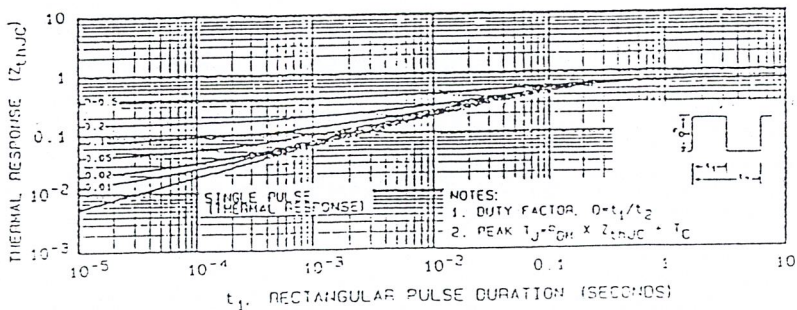


FIGURE 5. MAXIMUM EFFECTIVE TRANSIENT THERMAL IMPEDANCE, JUNCTION-TO-CASE vs. PULSE DURATION

IRFP450, IRFP451, IRFP452, IRFP453 IRFP450R, IRFP451R, IRFP452R, IRFP453R

Performance Curves (Continued)

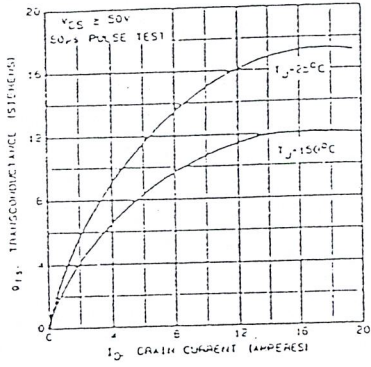


FIGURE 6. TYPICAL TRANSCONDUCTANCE vs DRAIN CURRENT

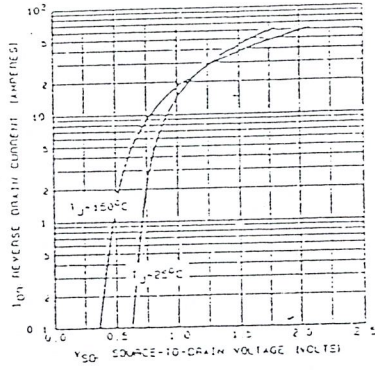


FIGURE 7. TYPICAL SOURCE-DRAIN DIODE FORWARD VOLTAGE

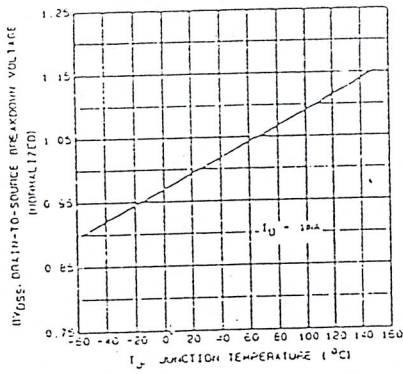


FIGURE 8. BREAKDOWN VOLTAGE vs TEMPERATURE

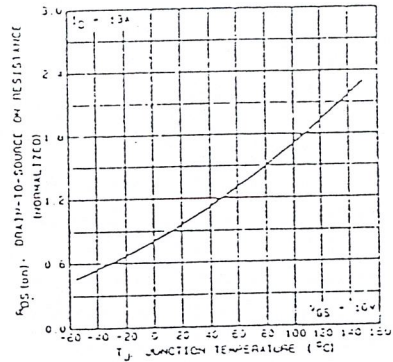


FIGURE 8. NORMALIZED ON-RESISTANCE vs TEMPERATURE

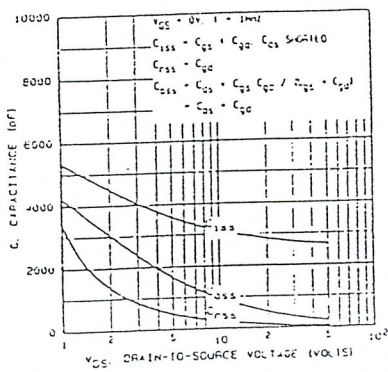


FIGURE 10. TYPICAL CAPACITANCE vs DRAIN-TO-SOURCE VOLTAGE

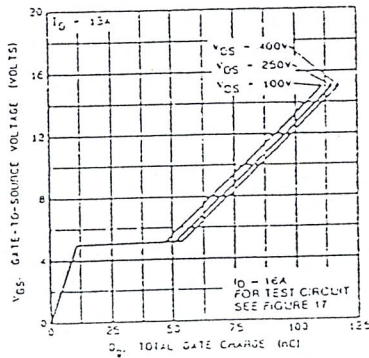


FIGURE 11. TYPICAL GATE CHARGE vs GATE-TO-SOURCE VOLTAGE

IRF450, IRF451, IRF452, IRF453 IRF450R, IRF451R, IRF452R, IRF453R

Performance Curves (Continued)

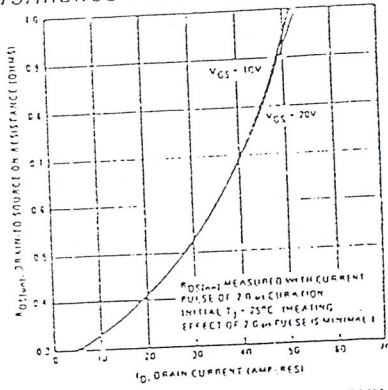


FIGURE 12. TYPICAL ON-RESISTANCE VS. DRAIN CURRENT

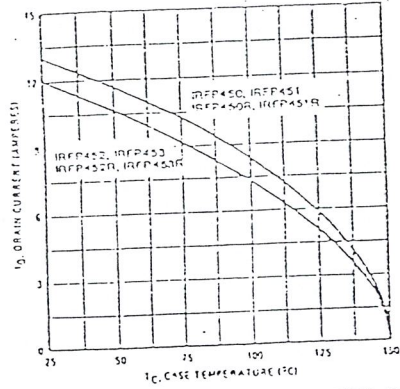


FIGURE 13. MAXIMUM DRAIN CURRENT VS. CASE TEMPERATURE

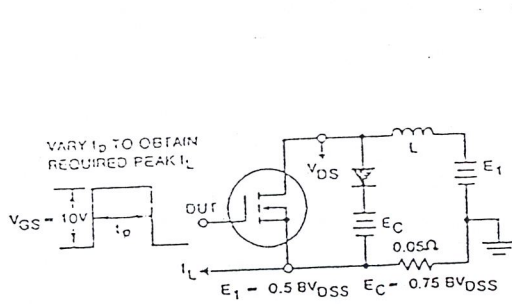


FIGURE 14a. CLAMPED INDUCTIVE TEST CIRCUIT

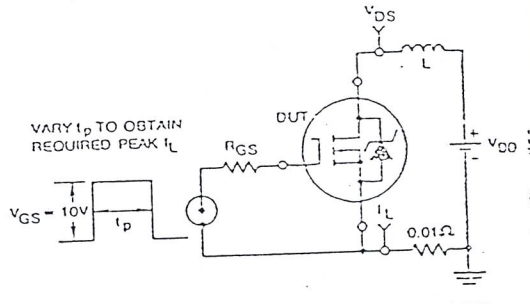


FIGURE 15a. UNCLAMPED ENERGY TEST CIRCUIT

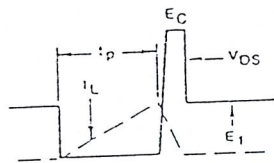


FIGURE 14b. CLAMPED INDUCTIVE WAVEFORMS

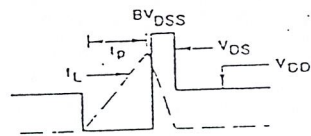


FIGURE 15b. UNCLAMPED ENERGY WAVEFORMS

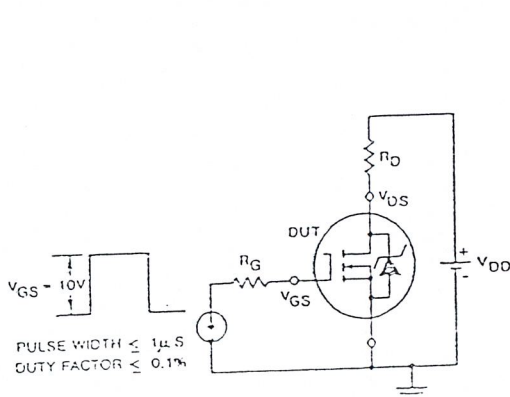


FIGURE 16. SWITCHING TIME TEST CIRCUIT

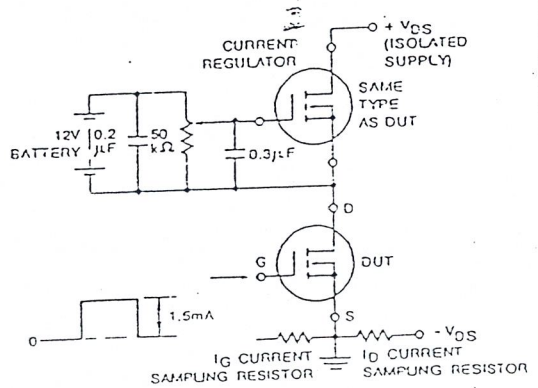
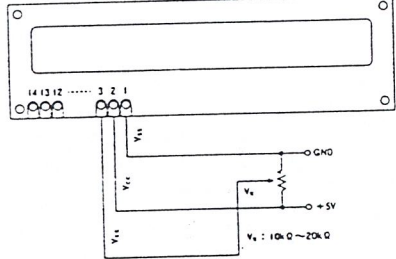


FIGURE 17. GATE CHARGE TEST CIRCUIT

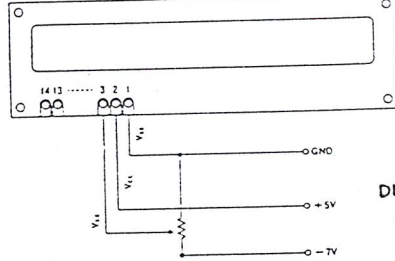
- เราสามารถต่อ VR ปรับค่าได้ 2 แบบ
- 1 คือ GND ในกรณี LCD แบบธรรมดา
- 2 คือ โฟลต ในกรณี LCD บางรุ่นที่ต้องใช้โฟลต เช่น LM2017
- *เราอาจประยุกต์ใช้โฟลตได้จาก ขา 6 IC MAX232 ได้

EXAMPLE OF POWER SUPPLY

Normal Temperature Type



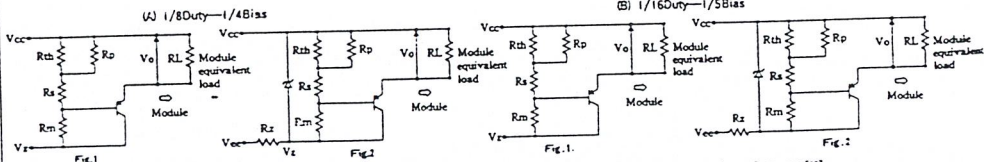
Extended Temperature Type



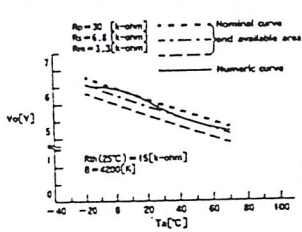
DMC20171

NOTE: If V_{cc} vary from recommended value, you cannot get proper contrast or viewing angle.

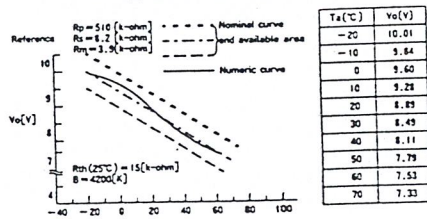
Examples of Temperature Compensation Circuits for Extended Temp Type. (Only for reference)



Thermistor: $R_{th}(25^{\circ}\text{C})=15(\text{k-ohm})$, $B=4200(\text{K})$
 Resistor: $R_p=30(\text{k-ohm})$, $R_a=6.8(\text{k-ohm})$, $R_m=3.3(\text{k-ohm})$
 Transistor: PNP Type
 $V_{cc} = +5\text{V}$, $V_{cc} = 0\text{V}$ (Logic Supply)
 $V_i = -1(\text{V})$ (-1.8 to -8.2[V])
 $V_{cc} < V_i(\text{V})$, $R_z = (V_i - V_{cc})/5(\text{k-ohm})$



Thermistor: $R_{th}(25^{\circ}\text{C})=15(\text{k-ohm})$, $B=4200(\text{K})$
 Resistor: $R_p=510(\text{k-ohm})$, $R_a=8.2(\text{k-ohm})$, $R_m=3.9(\text{k-ohm})$
 Transistor: PNP Type
 $V_{cc} = +5\text{V}$, $V_{cc} = 0\text{V}$ (Logic Supply)
 $V_i = -1(\text{V})$ (-10.725 to -11.775[V])
 $V_{cc} < V_i(\text{V})$, $R_z = (V_i - V_{cc})/5(\text{k-ohm})$



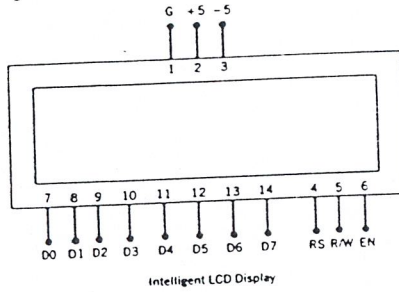
Specifications are subject to change without notice.

Intelligent LCD Display

In this section, we examine an intelligent LCD display of two lines, 20 characters per line, that is interfaced to the 8051. The protocol (handshaking) for the display is shown in Figure 8.8, and the interface to the 8051 in Figure 8.9.

The display contains two internal byte-wide registers, one for commands (RS = 0) and the second for characters to be displayed (RS = 1). It also contains a user-programmed RAM area (the character RAM) that can be programmed to generate any desired character that can be formed using a dot matrix. To distinguish between these two data areas, the hex command byte 80 will be used to signify that the display RAM address 00h is chosen.

FIGURE 8.8 Intelligent LCD Display



BIT	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	Function	
0	0	0	0	0	0	0	0	0	0	1	Clear LCD and memory, home cursor	
0	0	0	0	0	0	0	0	0	1	0	Clear and home cursor only	
0	0	0	0	0	0	0	0	1	I/O	S	Screen action as display character written S = 1/0: Shift screen/cursor	
0	0	0	0	0	0	1	D	<	D		I/O = 1/0: Cursor R/L screen L/R D = 1/0: Screen on/off C = 1/0: Cursor on/off B = 1/0: Cursor Blink/NoBlink	
0	0	0	0	0	1	S/C	R/L	0	0		S/C = 1/0: Screen/Cursor R/L = 1/0: Shift one space R/L	
0	0	0	0	1	DL	N	F	0	0		DL = 1/0: 8/4 Bits per character N = 1/0: 2/1 Rows of characters F = 1/0: 5X10/5X7 Dots/Character	
0	0	0	1	Character address								Write to character RAM Address after this
0	0	1	Display data address								Write to display RAM Address after this	
0	1	BF	Current address								BF = 1/0: Busy/Notbusy	
1	0	Character byte								Write byte to last RAM chosen		
1	1	Character byte								Read byte from last RAM chosen		

Port 1 is used to furnish the command or data byte, and ports 3.2 to 3.4 furnish register select and read/write levels.

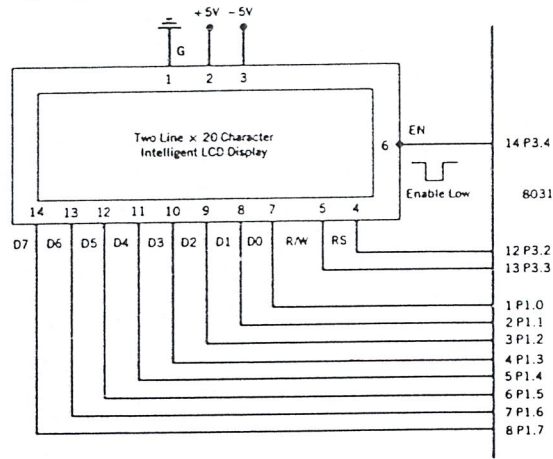
The display takes varying amounts of time to accomplish the functions listed in Figure 8.8. LCD bit 7 is monitored for a logic high (busy) to ensure the display is not overwritten. A slightly more complicated LCD display (4 lines x 40 characters) is currently being used in medical diagnostic systems to run a very similar program.

Lcdisp

The program "lcdisp" sends the message "hello" to an intelligent LCD display shown in Figure 8.8. Port 1 supplies the data byte. Port 3.2 selects the command (0) or data (1) registers. Port 3.3 enables a read (0) or write (1) level, and port 3.4 generates an active low-enable strobe.

DOT MATRIX LCD MODULE ETT

Intelligent LCD Circuit for "Lcdisp" Program



ADDRESS	MNEMONIC	COMMENT
	org 0000h	
lcdisp:	clr p3.2	;select the command register
	clr p3.3	;select write level
	mov a,#3fh	;command 8 bits/char. 2 rows, 5 x 10
	acall strobe	;strobe command to display
	mov a,#0eh	;command screen and cursor on, no blink
	acall strobe	
	mov a,#06h	;command cursor right as data displayed
	acall strobe	
	mov a,#01h	;clear all and home cursor
	acall strobe	
	setb p3.2	;select display data RAM register
	mov a,#'h'	;say "hello"
	acall strobe	
	mov a,#'e'	
	acall strobe	
	mov a,#'l'	
	acall strobe	
	acall strobe	
	mov a,#'o'	
	acall strobe	
here:	sjmp here	;message sent
:		
:		;the subroutine "strobe" is used to check for a display busy
:		;condition, and pulse P3.3 high-low-high to enable the display
:		;write or read
:		
strobe:	mov p1.#0ffh	;configure port 1 as an input
	setb p3.3	;set read level
wait:	setb p3.4	;generate read strobe
	clr p3.4	;enable the display
	jb p1.7,wait	;check for busy when BF = 1
	setb p3.4	;end of read strobe
	clr p3.3	;write character to display
	setb p3.2	;choose data RAM
	mov p1.a	;character to port 1
	clr p3.4	;generate write strobe
	setb p3.4	
	clr p3.2	;return with display as before call
	ret	
	end	

INTERFACE

V_{cc} is tied to ground while V_{ee} is tied to the +5 volt supply. Unless your application is heavily "I/O bound," it is easier to use one of the Z8's 8 bit ports for data, and use two CS

lines for address. Figure 2 shows a typical Z8 interface.

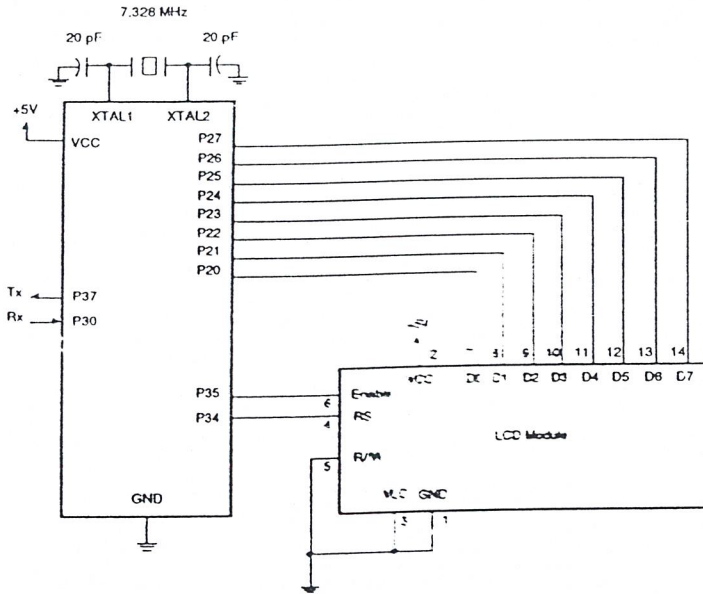


Figure 2. Typical Z8 LCD interface

INITIALIZATION

After power up, initialize the LCD before sending data. Note that the LCD module is very slow. Therefore, it is necessary to write a delay loop program in between instructions. Again, the LCD can be configured for either 8-bit or 4-bit data transfer. When operating in 4-bit mode, the upper nibble gets transferred first, followed by the lower nibble.

Table 7 shows complete instruction codes. In order to write the instruction codes to the LCD module, the RS line must be Low. Figure 3 shows an initialization sequence for an 8-bit transfer operation. The starting address for the DD RAM is 80H for the first eight characters. For the next eight characters, the starting address is C0H. If the DD RAM is programmed for auto-increment, then the DD RAM address is automatically incremented after each character write.

ALPHANUMERIC DOT MATRIX MODULES

DISPLAY CHARACTER POSITION AND DD RAM ADDRESS (CONTINUE)

2x16 DMM, 1/16 MUX N=1 : 2-LINE DISPLAY F=0 : 5X7 DOTS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	— DISPLAY POSITION
FIRST LINE	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	— DD RAM ADDRESS
SECOND LINE	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	

WHEN THE DISPLAY SHIFT OPERATION IS PERFORMED, THE DD RAM ADDRESS MOVED AS FOLLOW :

AFTER THE LEFT SHIFT INSTRUCTION

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	— DISPLAY POSITION
	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	— DD RAM ADDRESS
	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	

AFTER THE RIGHT SHIFT INSTRUCTION

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	— DISPLAY POSITION
	27	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	— DD RAM ADDRESS
	67	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	

2x20 DMM, 1/16 MUX N=1 : 2-LINE DISPLAY F=0 : 5X7 DOTS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	— DISPLAY POSITION
FIRST LINE	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	— DD RAM ADDRESS
SECOND LINE	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	

2x24 DMM, 1/16 MUX N=1 : 2-LINE DISPLAY F=0 : 5X7 DOTS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	— DISPLAY POSITION
FIRST LINE	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	— DD RAM ADDRESS
SECOND LINE	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	— DD RAM ADDRESS

2x40 DMM, 1/16 MUX N=1 : 2-LINE DISPLAY F=0 : 5X7 DOTS

	1	2	3	4	5	6	7	8	9	10	11	12	13	33	34	35	36	37	38	39	40	— DISPLAY POSITION
FIRST LINE	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	20	21	22	23	24	25	26	27	— DD RAM ADDRESS
SECOND LINE	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	60	61	62	63	64	65	66	67	— DD RAM ADDRESS

4x16 DMM, 1/16 MUX N=1 : 2-LINE DISPLAY F=0 : 5X7 DOTS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	— DISPLAY POSITION
FIRST LINE	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	— DD RAM ADDRESS
SECOND LINE	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	
THIRD LINE	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	
FOURTH LINE	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	

4x20 DMM, 1/16 MUX N=1 : 2-LINE DISPLAY F=0 : 5X7 DOTS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	— DISPLAY POSITION
FIRST LINE	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	— DD RAM ADDRESS
SECOND LINE	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	
THIRD LINE	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27	
FOURTH LINE	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67	

DMC162

● Display Format(16character × 2line) ● Display Fonts(5×8dots) ● Driving method(1/4D)

ABSOLUTE MAXIMUM RATINGS

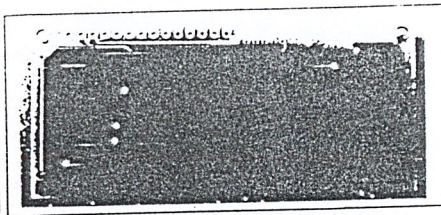
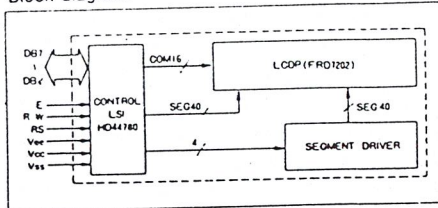
Item	Symbol	Test Condition	Standard Value			Unit
			min.	typ.	max.	
Power Supply Voltage for Logic	Vcc-Vss		0		7	V
Power Supply Voltage for LCD Drive	Vcc-Vee		0		13.5	V
Input "High" Voltage	Vi		Vss		Vcc	V
Operating Temperature	Ta		0		+50	°C
Storage Temperature	Tstg		-20		+70	°C

ELECTRICAL CHARACTERISTICS

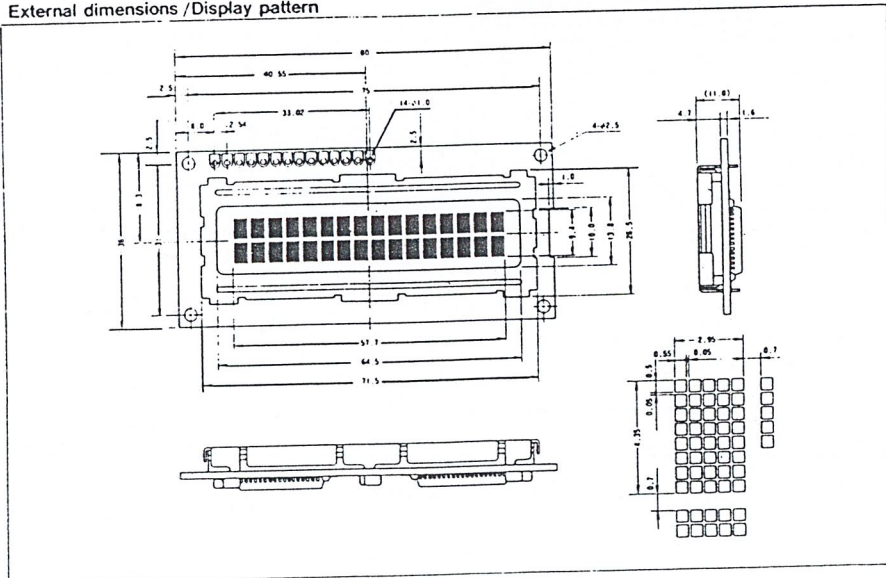
Item	Symbol	Test Condition	Standard Value			Unit
			min.	typ.	max.	
Input Voltage	ViH		2.2		Vcc	V
Input "Low" Voltage	ViL		-0.3		0.6	V
Output "High" Voltage	VoH	I _{OH} =0.25mA	2.4			V
Output "Low" Voltage	VoL	I _{OL} =1.2mA			0.4	V
Power Supply Current	Icc	Vcc=5.0V	0.5	2.0		mA

● Vcc=5.0V ± 5%, Ta=25°C

Block diagram



External dimensions / Display pattern



ภาคผนวก จ
คู่มือการใช้งาน

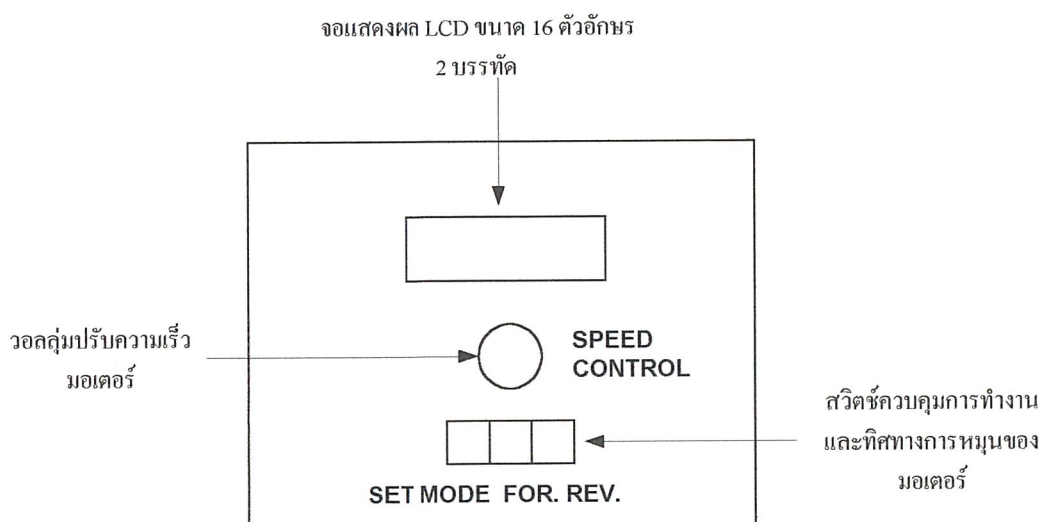
คู่มือการใช้เครื่องควบคุมความเร็วมอเตอร์เหนี่ยวนำ 3 เฟส

คุณสมบัติ

- สามารถขับมอเตอร์เหนี่ยวนำแบบ 3 เฟส ขนาดไม่เกิน 1 แรงม้า (764 วัตต์)
- สามารถปรับความถี่ได้ตั้งแต่ 3 – 50 Hz
- ความถี่ในการสวิตช์ (Switching) สามารถเลือกได้ 2 ค่า คือ 2400 Hz และ 4800 Hz
- แสดงผลด้วยจอแสดงผลแบบ LCD
- ควบคุมความเร็วด้วยโวลุ่มแบบปรับค่าได้
- ใช้ไฟเลี้ยง 220 โวลต์ 1 เฟส

ส่วนประกอบ

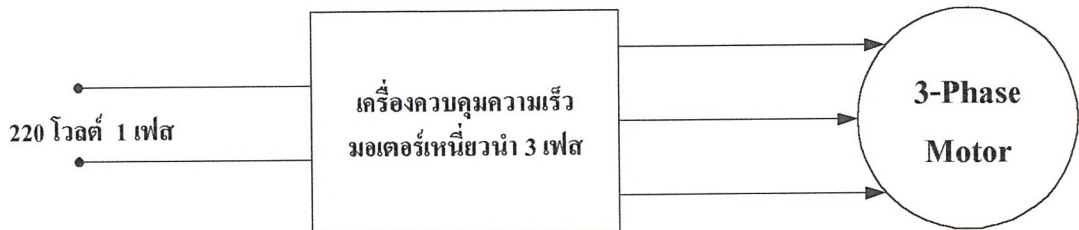
- สวิตช์ POWER ซึ่งอยู่ด้านหลังเครื่องทำหน้าที่เปิด-ปิด ไฟฟ้าเข้าเครื่อง
- จอแสดงผล LCD ทำหน้าที่แสดงการทำงานของมอเตอร์
- โวลุ่มปรับความเร็วมอเตอร์ ทำหน้าที่ปรับความเร็วในการหมุนของมอเตอร์
- สวิตช์ควบคุมฟังก์ชันการทำงาน ทำหน้าที่กำหนดโหมดการทำงานของมอเตอร์
- สวิตช์ FORVERS ทำหน้าที่ควบคุมให้มอเตอร์หมุนตามเข็มนาฬิกา
- สวิตช์ REVERSE ทำหน้าที่ควบคุมให้มอเตอร์หมุนทวนเข็มนาฬิกา



รูปที่ จ.1 ด้านหน้าของเครื่องควบคุมความเร็วมอเตอร์เหนี่ยวนำ 3 เฟส

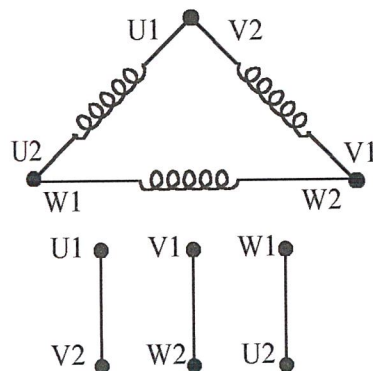
การต่อใช้งาน

1. การต่อใช้งานนั้นแสดงการต่อตาม รูปที่ จ.2 ซึ่งแสดงบล็อกไดอะแกรมของการต่อใช้งานของเครื่องควบคุมความเร็วของมอเตอร์เหนี่ยวนำแบบ 3 เฟส



รูปที่ จ.2 บล็อก ไดอะแกรมของการต่อใช้งาน

2. มอเตอร์ที่นำมาต่อใช้งานกับเครื่องควบคุมนั้นจะต้องขดลวดเป็นแบบเดลต้า ดังรูปที่ รูปที่ จ.3



รูปที่ จ.3 การต่อขดลวดแบบเดลต้าของมอเตอร์เหนี่ยวนำ 3 เฟส

3. การตั้งค่าพารามิเตอร์ โดยการกดปุ่ม SET. และเลือกหมายเลขของค่าพารามิเตอร์ โดยการใช้ปุ่ม FOR. และ REV. ซึ่งรายละเอียดของค่าพารามิเตอร์มีดังนี้
 - เปลี่ยนความถี่ในการ Switching
0 = 2400 Hz และ 1 = 4800 Hz
 - ระยะเวลาเริ่มต้นในการหมุนของมอเตอร์สามารถตั้งค่าได้ตั้งแต่ 0 วินาที ถึง 5 นาที
 - ระยะเวลาการหยุดหมุนของมอเตอร์สามารถตั้งค่าได้ตั้งแต่ 0 วินาที ถึง 5 นาที

การใช้งาน

1. การกำหนดให้มอเตอร์หมุนไปทางด้านขวาทำได้โดยกดปุ่ม FOR. ค้างไว้และสามารถเปลี่ยนค่าความเร็วได้โดยปรับที่โวลต์ปรับความเร็วมอเตอร์
2. การกำหนดให้มอเตอร์หมุนไปทางด้านซ้ายทำได้โดยกดปุ่ม REV. ค้างไว้และสามารถเปลี่ยนค่าความเร็วได้โดยปรับที่โวลต์ปรับความเร็วมอเตอร์

การนำไปประยุกต์ใช้งาน

เครื่องควบคุมความเร็วมอเตอร์เหนี่ยวนำ 3 เฟส สามารถนำไปประยุกต์ใช้กับเครื่องจักรที่มีการวัดระยะอัตโนมัติโดยต่อเครื่องควบคุมเข้ากับมอเตอร์ที่ต่อเข้ากับเพลาขับ โดยที่เครื่องควบคุมนี้ควบคุมความด้วยไมโครคอนโทรลเลอร์จึงทำให้สามารถควบคุมความเร็วมอเตอร์และตำแหน่งที่ต้องการได้

ข้อควรระวัง

1. การต่อสายจากเครื่องควบคุมเข้ากับมอเตอร์ ควรกดปุ่ม POWER ด้านหลังเครื่องให้ OFF ก่อนเพื่อปิดเครื่องและป้องกันการลัดวงจร
2. การต่อสายตามจุดต่างๆ ต้องให้แน่นหนาเพื่อป้องกันสายหลุดขณะมอเตอร์หมุนซึ่งอาจทำให้เกิดการลัดวงจรได้
3. มอเตอร์เหนี่ยวนำ 3 เฟสที่นำมาต่อร่วมกับเครื่องควบคุม ขนาดไม่เกิน 1 แรงม้า
4. การตรวจซ่อมเครื่องควรให้ช่างผู้ชำนาญเป็นผู้ตรวจซ่อม

บรรณานุกรม

- ณรงค์ ขอนตะวัน, มอเตอร์กระแสสลับ, เราวิณการพิมพ์, กรุงเทพฯ
- ธีรวัฒน์ ประกอบผล, การประยุกต์ใช้งานไมโครคอนโทรลเลอร์, สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), กรุงเทพฯ พิมพ์ครั้งที่ 1, 2540
- ประเสริฐ ฟูรัตนพาณิชย์, วันชัย เค่นพ่าย, ปริญญาธิพนธ์เครื่องควบคุมความเร็วมอเตอร์ 3 เฟส, คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, กรุงเทพฯ, 2537
- มงคล ทองสงคราม, เครื่องกลไฟฟ้ากระแสสลับ, รามการพิมพ์, กรุงเทพฯ 2535

ประวัติผู้แต่ง



ชื่อผู้ทำวิทยานิพนธ์

นายปฏิภาณ ชนะชัย

วันเดือนปีเกิด

30 พฤศจิกายน พ.ศ.2521

สถานที่เกิด

จังหวัดอุบลราชธานี

ภูมิลำเนาเดิม

จังหวัดอุบลราชธานี

ที่อยู่ปัจจุบัน

57/3 หมู่ 4 ถนนอ่อนนุช เขตลาดกระบัง
กรุงเทพ 10520

โทรศัพท์

01-8767856

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนมูลนิธิวัดศรีอุบลรัตนาราม

มัธยมศึกษาตอนต้น

โรงเรียนเบ็ญจะมะมหาราช

ประกาศนียบัตรวิชาชีพ (ปวช.)

วิทยาลัยเทคนิคอุบลราชธานี

ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)

วิทยาลัยเทคนิคมินบุรี

ปริญญาตรี

สาขาอิเล็กทรอนิกส์และคอมพิวเตอร์

ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ผลงานที่ได้รับ

-

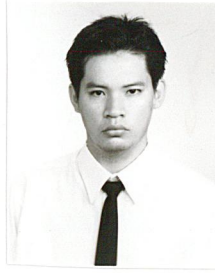
ทุนการศึกษา

-

คติพจน์

ความแน่นอนคือความไม่แน่นอน

ประวัติผู้แต่ง



ชื่อผู้ทำวิทยานิพนธ์	นายภัทรฤทธิ์ ผาดินาวิน
วันเดือนปีเกิด	2 พฤษภาคม พ.ศ.2520
สถานที่เกิด	จังหวัดกรุงเทพมหานคร
ภูมิลำเนาเดิม	จังหวัดอุดรธานี
ที่อยู่ปัจจุบัน	17/42 ถนนสีหบุรานุกิจ แขวงมีนบุรี เขตมีนบุรี กรุงเทพมหานคร 10510
โทรศัพท์	-
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนบ้านหนองวัวซอ
มัธยมศึกษาตอนต้น	โรงเรียนอุดรพิทยานุกูล
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคอุดรธานี
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคอุดรธานี
ปริญญาตรี	สาขาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ผลงานที่ได้รับ	-
ทุนการศึกษา	-
คติพจน์	เพียงความอดทนเท่านั้น จึงจะถึงฝั่ง

ประวัติผู้แต่ง



ชื่อผู้ทำวิทยานิพนธ์	นายภูสิต เพชรพุทธานุภ
วันเดือนปีเกิด	26 ตุลาคม พ.ศ. 2520
สถานที่เกิด	จังหวัดนครศรีธรรมราช
ภูมิลำเนาเดิม	จังหวัดนครศรีธรรมราช
ที่อยู่ปัจจุบัน	111/39 ถนนอ่อนนุช-ลาดกระบัง แขวงลาดกระบัง เขต ลาดกระบัง กรุงเทพมหานคร 10520
โทรศัพท์	01-6972875
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนอนุบาลจังหวัดนครศรีธรรมราช
มัธยมศึกษาตอนต้น	โรงเรียนศรีธรรมราชศึกษา
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคนครศรีธรรมราช
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคนครศรีธรรมราช
ปริญญาตรี	สาขาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ผลงานที่ได้รับ	-
ทุนการศึกษา	-
คติพจน์	ความพยายามอยู่ที่ไหน ความสำเร็จอยู่ที่นั่น

ประวัติผู้แต่ง



ชื่อผู้ทำวิทยานิพนธ์

นายเอกปริญญ์ ทางแก้ว

วันเดือนปีเกิด

24 มกราคม พ.ศ. 2521

สถานที่เกิด

จังหวัดกรุงเทพมหานคร

ภูมิลำเนาเดิม

จังหวัดนนทบุรี

ที่อยู่ปัจจุบัน

55/10 หมู่ 2 ถนนรามอินทรา เขตบางเขน
กรุงเทพ 10220

โทรศัพท์

02 - 9718332

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนชุมชนวัดราษฎร์นิยม

มัธยมศึกษาตอนต้น

โรงเรียนลาดปลาเค้าพิทยาคม

ประกาศนียบัตรวิชาชีพ (ปวช.)

วิทยาลัยเทคนิคมีนบุรี

ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)

วิทยาลัยเทคนิคมีนบุรี

ปริญญาตรี

สาขาอิเล็กทรอนิกส์และคอมพิวเตอร์

ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ผลงานที่ได้รับ

ทุนการศึกษา

กองทุนกู้ยืมเพื่อการศึกษาของรัฐบาล

คดีพจน์

ทุกปัญหา ล้วนยอมมีทางออกที่ดีเสมอ