

การเปรียบเทียบรูปแบบสัญญาณสวิตช์ที่ใช้สำหรับ
เครื่องทำความสะอาดด้วยอัลตราโซนิค

COMPARISON OF SWITCHING STRATEGIES FOR
AN ULTRASONIC CLEANER



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของงานวิจัยของคณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตศึกษา

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2551

KMITL-2008-EN-M-020-642

การเปรียบเทียบรูปแบบสัญญาณการสวิตช์ที่ใช้สำหรับ
เครื่องทำความสะอาดด้วยอัลตราโซนิค

COMPARISON OF SWITCHING STRATEGIES FOR
AN ULTRASONIC CLEANER



ชาญกิต บัวศรี
CHANKIT BUASRI

เลขหมู่.....
เลขทะเบียน..... 82659
วัน,เดือน,ปี..... 21 ก.ค. 2551

b. 11ค 50 250
i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่หอสมุดเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
บัณฑิตวิทยาลัย
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2551

**COMPARISON OF SWITCHING STRATEGIES FOR
AN ULTRASONIC CLEANER**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING**

SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2008

KMITL-2008-EN-M-020-342



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

COPYRIGHT 2008

ไม่ว่าการพิมพ์ การฟอกสี หรือการดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การเปรียบเทียบรูปแบบสัญญาณการสวิตซ์ที่ใช้สำหรับเครื่องทำความสะอาดด้วยอัลตราโซนิค

Comparison of Switching Strategies for an Ultrasonic Cleaner

นักศึกษา นายชาญกิต บัวศรี

รหัสประจำตัว 46060322

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมไฟฟ้า

อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร.อนุวัฒน์ จางวนิชเลิศ

คณะกรรมการสอบวิทยานิพนธ์	ลายมือชื่อ
รศ.ดร.วิจิตร	กนิเรศ
ดร.ภพ	จันทร์เจริญสุข
ดร.สุรพงศ์	สุวรรณกวิน
ดร.สมยศ	เกียรติวนิชวิไล
ผศ.ดร.อนุวัฒน์	จางวนิชเลิศ

วัน/เดือน/ปี ที่สอบ 20 พฤษภาคม 2551 เวลา 09.00-11.00 น.

สถานที่สอบ ณ ห้องประชุม 2 ชั้น 3 อาคาร A

บัณฑิตวิทยาลัยรับรองแล้ว

(รศ.ดร.รวีวรรณ ชินะตระกูล)

คณบดีบัณฑิตวิทยาลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานวันที่ 30 เมษายน 2551
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การเปรียบเทียบรูปแบบสัญญาณการสวิตช์ที่ใช้สำหรับเครื่อง ทำความสะอาดด้วยอัลตราโซนิค
นักศึกษา	นายชาญกิต บัวศรี
รหัสประจำตัว	46060322
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2551
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.อนุวัฒน์ จางวนิชเลิศ

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนอการเปรียบเทียบสัญญาณ ของเครื่องทำความสะอาดด้วยคลื่น อัลตราโซนิคโดยใช้ไมโครโปรเซสเซอร์สร้างสัญญาณเปรียบเทียบระหว่าง PWM, PWM แบบ เลื่อนเฟส และ PDM ให้วงจรอินเวอร์เตอร์แบบฟลูบริคขนาด 700 W เพื่อขับหัวอัลตราโซนิค ทรานสดิวเซอร์ โดยทดสอบที่ความถี่เรโซแนนท์ 30 kHz เพื่อหารูปแบบ ที่ให้ประสิทธิภาพออกมา ดีที่สุดในการทำทำความสะอาดอุปกรณ์ที่ผิวเรียบ และผิวไม่เรียบ โดยจะมุ่งเน้นในการนำไปใช้ในการ ทำความสะอาดอุปกรณ์ประเภทไฟฟ้า และอิเล็กทรอนิกส์ โดยใช้ตัวกลางของเหลวที่ใช้ทำความสะอาด เป็นน้ำยาเอนกประสงค์ (WD-40) หัวอัลตราโซนิคทรานสดิวเซอร์ จะทำหน้าที่กำเนิดคลื่น อัลตราโซนิคโดยการเปลี่ยนพลังงานไฟฟ้าให้เป็นพลังงานกล คลื่นอัลตราโซนิคเมื่อผ่านของเหลว จะทำให้เกิดกระบวนการคาวิเตชัน และอิมโพลชัน ซึ่งเป็นขบวนการในการทำมาสะอาด สัญญาณแบบ PWM คือป้อนสัญญาณแบบต่อเนื่องทำให้เกิดฟองคาวิเตชันจนมีขนาดใหญ่ และแตก ตัวลงจนเกิดอิมโพลชัน จึงได้นำมาป้อนสัญญาณแบบไม่ต่อเนื่องโดยใช้ PS-PWM และ PDM ทำให้ ฟองคาวิเตชันมีขนาดไม่ใหญ่จนแตกตัว จึงมีคาวิเตชันมากกว่าป้อนสัญญาณแบบ PWM เพื่อนำมาใช้กับอุปกรณ์ที่ผิวไม่เรียบ โดยการทดลองได้ชั่งน้ำหนักก่อน และหลังชำระล้างด้วยสัญญาณ PWM, PS-PWM และ PDM น้ำหนักได้หายไป 5.86, 6.51 และ 8.06 กรัมตามลำดับ ซึ่งสัญญาณ แบบไม่ต่อเนื่องเมื่อนำไปใช้กับอุปกรณ์ที่ผิวไม่เรียบจะมีประสิทธิภาพมากกว่าเนื่องจากขนาดฟองคาวิ เตชันที่มีขนาดเล็กนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	COMPARISON OF SWITCHING STRATEGIES FOR AN ULTRASONIC CLEANER
Student	Mr. Chankit Buasri
Student ID.	46060322
Degree	Master of Engineering
Program	Electrical Engineering
Year	2008
Thesis Advisor	Asst. Prof. Dr. Anuwat Jangwanitlert

ABSTRACT

This thesis presents the comparison of Control Signal Patterns ultrasonic cleaner by using the microprocessor for generating PWM, PWM phase shift control and PDM patterns, these patterns are supplied in the 700 W full-bridge inverter to drive the ultrasonic transducers. Resonant frequency (30 kHz) is tested in order to find the efficiency of ultrasonic cleaner which one should be the best in flat and not flat equipment. The cleaning is focus on electrical and electronic equipment by used WD-40 is liquid medium fill in the tank. Ultrasonic transducer can generate ultrasonic wave to transfer electrical energy to mechanical energy. Cavitation and implosion are occurred in liquid when ultrasonic wave travels through liquid medium in the process of cleaning. The PWM pattern is fed in continuous signal to generate maximum cavitation bubbles and implosion occurred when the bubbles is exploded. Then, uncontinuous signals : PS-PWM and PDM patterns is used to control the size of cavitation bubbles in order not to get maximum size. The cavitation of uncontinuous signal is occurred more than that of continuous signal and used in a rough equipment board. The test is measured in weight from before and after cleaning by feeding PWM, PS-PWM and PDM signal patterns. The weight is gone approximately 5.86, 6.51 and 8.06 g, respectively. When used uncontinuous signal for rough equipment board will have efficacy because of small sizes of cavitation babbles.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.ดร. อนุวัฒน์ งามวิมลเลิศ ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ข้าพเจ้ารู้สึกทราบบ้างในความอนุเคราะห์จากท่าน อาจารย์ และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยี พระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณ บัญชา หิรัญสิงห์ และเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมไฟฟ้า สถาบัน เทคโนโลยี พระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจ เสมอมา

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะวิศวกรรมศาสตร์ที่ให้ความช่วยเหลือ ในเรื่องต่างๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

ชาญกิต บัวศรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมติฐานของการศึกษา.....	2
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย.....	2
1.5 การเปรียบเทียบระหว่างวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน.....	3
1.6 ขอบเขตการวิจัย.....	3
1.7 ขั้นตอนการศึกษา.....	4
บทที่ 2 ทฤษฎีการทำความสะอาดด้วยคลื่นอัลตราโซนิค.....	5
2.1 ทฤษฎีคลื่นอัลตราโซนิค.....	5
2.2 การใช้คลื่นอัลตราโซนิคทำความสะอาด.....	7
2.2.1 Cavitation and Implosion.....	7
2.2.2 ประโยชน์ของคลื่นอัลตราโซนิคในกระบวนการทำความสะอาด และการชำระล้าง.....	8
2.3 อัลตราโซนิคทรานสดิวเซอร์.....	13
2.4 เร็กติไฟเยอร์และอินเวอร์เตอร์.....	18
2.4.1 เร็กติไฟเยอร์.....	18
2.4.2 อินเวอร์เตอร์.....	19
2.5 พีดีบีดับลิวเอ็มฟลูอิดริคจ์คอนเวอร์เตอร์แบบไม่ควบคุมให้เข้าไปได้ประโยชน์ด้วยแรงดัน ไม่ต่ำกว่าค่า.....	21
2.6 พีดีบีดับลิวเอ็มแบบเลื่อนเฟสฟลูอิดริคจ์คอนเวอร์เตอร์เอกสารทุกครั้งที่มีการนำไปใช้.....	24
2.7 พีดีเอ็มฟลูอิดริคจ์คอนเวอร์เตอร์.....	30
2.8 ฮาร์โมนิก.....	33

สารบัญ (ต่อ)

	หน้า
2.9 สรุปและข้อเสนอแนะ.....	37
บทที่ 3 วงจรควบคุมและสร้างสัญญาณ.....	38
3.1 บทนำ.....	38
3.2 การสร้างสัญญาณ PWM.....	42
3.3 การสร้างสัญญาณ PS-PWM.....	45
3.4 การสร้างสัญญาณ PDM.....	48
3.5 สรุป.....	50
บทที่ 4 ขั้นตอนการทดลองและผลการทดลอง.....	51
4.1 ขั้นตอนการทดลอง.....	51
4.2 ผลการทดลอง.....	56
4.3 สรุป.....	64
บทที่ 5 สรุปผลการวิจัยและการวิเคราะห์.....	65
5.1 บทสรุป.....	65
5.2 ข้อเสนอแนะ.....	66
บรรณานุกรม.....	67
ภาคผนวก.....	68
ภาคผนวก ก. Datasheet.....	69
ภาคผนวก ข. ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่.....	110
ประวัติผู้เขียน.....	116

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
4.1	
ค่าที่วัดความสกปรกก่อนและหลังชำระล้างเปรียบเทียบกัน ทั้งสามรูปแบบสัญญาณ.....	60



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 คลื่นเสียงที่เกิดจากการเคลื่อนที่ของการสั่นสะเทือน.....	6
2.2 คลื่นเสียงที่เกิดจากปรากฏการณ์ฟ้าผ่า.....	6
2.3 ธรรมชาติของคลื่นเสียงที่มีช่วงอัดและขยาย.....	7
2.4 กระบวนการการเกิดควิเตชันและ อิม โฟชั่น.....	8
2.5 หน้าสัมผัสของชิ้นงานที่มีคราบสกปรกติดอยู่โดยชิ้นงานแช่อยู่ในน้ำยาทำความสะอาด.....	9
2.6 น้ำยาที่ติดกับคราบสกปรกเกิดการอึดตัว และแสดงให้เห็นว่าน้ำยาที่ยังไม่อึดตัว เข้าไม่ถึง.....	9
2.7 กระบวนการควิเตชัน และอิม โฟชั่น ช่วยให้เกิดการหมุนเวียนของน้ำยาทำความสะอาด.....	10
2.8 อนุภาคที่ไม่ละลายในสารละลายซึ่งเกาะติดชิ้นงานอยู่.....	11
2.9 การทำความสะอาด โดยใช้แปรง.....	12
2.10 ผลของอุณหภูมิอันเนื่องมาจากปัจจัยต่างๆ.....	13
2.11 เมคานิคเคิลทรานสดิวเซอร์.....	14
2.12 เปียโซอิเล็กทริก.....	16
2.13 อัลตราโซนิกทรานสดิวเซอร์แบบ เปียโซอิเล็กทริก.....	17
2.14 วงจรฟลูเวฟเร็กติไฟเออร์.....	19
2.15 อินเวอร์เตอร์ (a) แบบฮาล์ฟบริดจ์ (b) แบบฟูลบริดจ์.....	20
2.16 พีคัปบลิวเอ็มฟูลบริดจ์คอนเวอร์เตอร์.....	21
2.17 สัญญาณการทำงานของพีคัปบลิวเอ็มฟูลบริดจ์คอนเวอร์เตอร์.....	22
2.18 การทำงานของ PWM ที่สภาวะ 1 ถึง 2.....	23
2.19 การทำงานของสัญญาณ PWM แบบเล็อนเฟส.....	25
2.20 การทำงานของ PWM แบบเล็อนเฟสที่สภาวะ 1 ถึง 8.....	29
2.21 การทำงานของสัญญาณ PDM.....	31
2.22 การทำงานของสัญญาณ PDM ที่สภาวะที่ 1 ถึง 3.....	32
2.23 แสดงถึงฮาร์มอนิกลำดับที่ 3, 5 และ 7 ที่มุมต่างๆทำให้สัญญาณ ไซน์มีรูปร่างผิดเพี้ยน.....	34
3.1 หลักการทำงานของเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิก.....	38
3.2 บีหัวอัลตราโซนิกทรานสดิวเซอร์ที่อยู่ใต้ห้องแทงค์ (tank) ที่บรรจุของเหลวที่ใช้ทำ ไม่ว่ากรณี ความสะอาดทั้งหมดมิให้ลดแปลงเป็นแก๊สและทั้งหมดจึงเก็บรวมรวมกัน ที่ผิวกรรมไปให้.....	39
3.3 คลื่นเสียงอัลตราโซนิกเมื่อผ่านตัวกลางของเหลว และเกิดกระบวนการทำความสะอาด.....	39

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.4 รูปวงจรไมโครโปรเซสเซอร์ dsPIC 30F2010.....	40
3.5 วงจรขับเกทของเพาเวอร์มอสเฟต.....	41
3.6 วงจรภาคกำลัง.....	41
3.7 รูปคลื่นแรงดันเอาต์พุตของสัญญาณควบคุมทั้ง 3 แบบคือ PDM, PWM, PS-PWM ที่ใช้ขับโหลด(หัวอัลตราโซนิค).....	42
3.8 Flowchart การสร้างรูปสัญญาณ PWM.....	44
3.9 คาบเวลาที่ใช้เขียน โปรแกรมรูปสัญญาณแบบ PWM.....	44
3.10 ช่วงสัญญาณที่แบ่งเป็น 8 ช่วงเพื่อใช้ในการเขียนโปรแกรม.....	45
3.11 Flowchart การสร้างรูปสัญญาณแบบ PS-PWM.....	47
3.12 Flowchart การสร้างรูปสัญญาณแบบ PDM.....	49
4.1 วงจรสมมูล อัลตราโซนิคทรานสดิวเซอร์ที่ใช้ทั่วไป.....	51
4.2 วัดค่าอิมพีแดนซ์ที่หัวอัลตราโซนิคทรานสดิวเซอร์.....	51
4.3 รูปแรงดันและกระแส ที่หัวอัลตราโซนิคทรานสดิวเซอร์ โดยการซิมูเลตด้วย โปรแกรม MATLAB ด้วยการป้อนรูปสัญญาณ PWM.....	52
4.4 รูปแรงดันและกระแส ที่หัวอัลตราโซนิคทรานสดิวเซอร์ โดยการซิมูเลตด้วย โปรแกรม MATLAB ด้วยการป้อนรูปสัญญาณ PS-PWM.....	53
4.5 กำลังไฟฟ้าที่ความถี่รอบๆความถี่เรโซแนนท์.....	53
4.6 รูปแรงดันและกระแสเอาต์พุตเมื่อป้อนสัญญาณแบบ (a) PWM (b) PS-PWM (c) PDM.....	55
4.7 รูปวงจรที่วัดค่า.....	56
4.8 ค่ากำลังงานไฟฟ้าที่แรงดันต่างๆ.....	56
4.9 ค่าประสิทธิภาพกับกำลังงานไฟฟ้าต่างๆ.....	57
4.10 วัดค่าฮาร์มอนิกของแรงดันเอชีอินพุต.....	58
4.11 วัดค่าฮาร์มอนิกของแรงกระแสเอชีอินพุต.....	58
4.12 น้ำยาที่ใช้ทำความสะอาดอุปกรณ์ไฟฟ้าและอิเล็กทรอนิกส์ที่นำมาใช้ในการทดลองนี้.....	59
4.13 บอร์ดอิเล็กทรอนิกส์ที่นำมาใช้ชำระล้างในการทดลอง.....	59
4.14 รูปชั่งน้ำหนักก่อนชำระล้าง และน้ำหนักหลังชำระล้าง (a) PS-PWM (b) PDM (c) PWM.....	63
4.15 เครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิค.....	64

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันการทำความสะอาดด้วยคลื่นอัลตราโซนิกเป็นที่นิยมและใช้กันอย่างแพร่หลาย เนื่องจากเป็นวิธีทำความสะอาดที่มีประสิทธิภาพ รวดเร็ว และไม่ทำลายสิ่งแวดล้อม ความสะอาดกลายเป็นสิ่งมุ่งเน้นที่สำคัญในหลาย ๆ โรงงาน เช่น โรงงาน อิเล็กทรอนิกส์ ซึ่งให้ความสำคัญกับความสะอาดอยู่แล้วนั้น ก็ได้ให้ความสำคัญมากขึ้น ทำให้เกิดความก้าวหน้าทางเทคโนโลยีการทำความสะอาด

การทำความสะอาดด้วยคลื่นอัลตราโซนิก เป็นกระบวนการทำความสะอาดประเภทจุ่มและล้างในของเหลว (immersion cleaning) ที่มีประสิทธิภาพ สามารถกำจัดสิ่งสกปรกได้หลายชนิด เช่น เศษผงของเครื่องจักร น้ำมัน อนุภาคสิ่งสกปรกเล็กๆ ได้เป็นอย่างดี

ระบบการทำความสะอาดด้วยคลื่นอัลตราโซนิก ประกอบด้วย ทรานสดิวเซอร์ (transducer) วงจรขับ แทงค์ (tank) และตัวกลางของเหลว ตัวทรานสดิวเซอร์ จะแปลงพลังงานที่ได้จากเครื่องกำเนิดให้เป็นการสั่นสะเทือนจากพลังงานเสียง ซึ่งการสั่นสะเทือนนี้จะถูกส่งผ่านแพร่กระจายไปตามแทงค์ และจะผลิตฟองอากาศควิเดชั่น (cavitations) ขึ้นมาในตัวกลางของเหลวภายในแทงค์ บรรจุฟองอากาศเหล่านี้จะจัดวางตัว (formation) และยุบตัวลง (collapse) เพื่อให้เกิดปฏิกิริยาการขัดถูขึ้น (scrubbing) ซึ่งจะใช้ในการกำจัดสิ่งสกปรกได้เป็นอย่างดี

เครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิกจากแต่เดิมได้ใช้สัญญาณ Pulse - Width Modulation (PWM) [1] ในการทำให้หัวอัลตราโซนิกทรานสดิวเซอร์ทำงานเพื่อสร้างคลื่นอัลตราโซนิกผ่านตัวกลางของเหลวในแทงค์ ที่ใช้ทำความสะอาดอุปกรณ์ วิทยานิพนธ์นี้ได้ทดลองใช้รูปสัญญาณ Pulse Density Modulation (PDM) [2] และ Phase-Shifted Pulse - Width Modulation (PS-PWM) [3] ในการทำให้เกิดคลื่นอัลตราโซนิก เพื่อเปรียบเทียบปฏิกิริยาของการทำความสะอาด และประสิทธิภาพของแต่ละรูปสัญญาณ เครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิกส่วนใหญ่ใช้ในการทำความสะอาดเครื่องมือแพทย์ ซึ่งอุปกรณ์ที่วิทยานิพนธ์นี้มุ่งเน้นนำมาใช้ทำความสะอาด คือ อุปกรณ์ประเภทอิเล็กทรอนิกส์และไฟฟ้าเป็นหลัก เช่นแผงบอร์ดต่างๆ ซึ่งมักจะมีอุปกรณ์ตัวเล็กๆ จำนวนมากวางติดๆกัน ทำให้แผงวงจรพวกนี้มีชอกเล็กชอกน้อยจำนวนมาก เมื่อใช้ไปนานๆจะมีพวกฝุ่นผง หรือคราบสกปรกที่เกาะติด จากแต่เดิมใช้น้ำยาเอนกประสงค์แบบสเปรย์ ในการฉีดทำความสะอาดตามชอกเล็กๆ การทดลองนี้จึงได้นำน้ำยาเอนกประสงค์แบบของเหลวที่ไม่ทำความเสียหายให้กับอุปกรณ์อิเล็กทรอนิกส์ มาใช้เป็นตัวกลางของเหลวในแทงค์ เพื่อใช้กระบวนการทำความสะอาดด้วยคลื่นอัลตราโซนิก ทำให้น้ำยาแทรกซึมเข้าไปในชอกเล็กๆ

แล้วนำคราบสกปรกนั้นออกมาภายนอก จึงเป็นอีกทางเลือกในการทำความสะอาดอุปกรณ์ไฟฟ้า และ อิเล็กทรอนิกส์

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์ฉบับนี้มุ่งหวังเพื่อศึกษาการทำความสะอาดอุปกรณ์ไฟฟ้า และอิเล็กทรอนิกส์ ด้วยคลื่นอัลตราโซนิค ด้วยนำวิธีการเดิมที่มีอยู่นำมาปรับใช้ให้เข้ากับอุปกรณ์ที่จะชำระล้าง โดยวิธีการเดิมที่มีอยู่ ส่วนใหญ่นำไปใช้กับอุปกรณ์ ที่มีลักษณะราบเรียบ ไม่ขรุขระ จึงนิยมใช้ในเครื่องมือที่ต้องการความสะอาด เช่นเครื่องมือทางการแพทย์ เมื่อนำมาใช้กับอุปกรณ์ที่มีลักษณะเป็นชิ้นเล็กๆ จำนวนมาก เช่น อุปกรณ์แผงอิเล็กทรอนิกส์ในปัจจุบัน กระบวนการทำความสะอาดของเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิค จะเกิดฟองอากาศขึ้นมา เพื่อจะทำความสะอาดอุปกรณ์ ซึ่งกระบวนการทำความสะอาดจะกล่าวโดยละเอียดในบทต่อไป เมื่ออุปกรณ์ที่จะทำความสะอาดมีซอกเล็กซอกน้อยจำนวนมาก จึงจำเป็นที่จะต้องใช้ฟองอากาศที่มีขนาดเล็กตามไปด้วย จึงได้นำรูปแบบสัญญาณที่ไม่ต่อเนื่องมาป้อนให้กับหัวอัลตราโซนิคทรานสดิวเซอร์ เพื่อศึกษากระบวนการทำความสะอาดด้วยคลื่นอัลตราโซนิค เปรียบเทียบกับรูปแบบสัญญาณที่ต่อเนื่องเพื่อดูประสิทธิภาพการทำมาสะอาด และรูปแบบไหนเหมาะสมกับอุปกรณ์ที่นำมาชำระล้าง ชนิดไหน

1.3 สมมติฐานของการศึกษา

จากหลักการเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิค กระบวนการสำคัญในการทำมาสะอาดด้วยวิธีนี้ก็คือ กระบวนการคาวิเทชัน และ อิมโพชั่น ซึ่งกระบวนการคาวิเทชัน เกิดจากการที่คลื่นขยายตัวจนเกิดฟองอากาศ และ โตขึ้นจนในที่สุดก็เกิดการยุบตัวอย่างรุนแรงของคาวิเทชัน เป็นผลให้เกิดการ อิมโพชั่น ซึ่งจะทำให้คลื่น ช็อกเวฟ แผ่กระจายออกไปจากบริเวณที่ยุบตัว ซึ่งเราได้้นำการสร้างคลื่นแบบไม่ต่อเนื่องเพื่อว่า เมื่อเกิดการ คาวิเทชัน แล้วก็หยุดการสร้างคลื่นเพื่อให้คาวิเทชัน ยุบตัวลงในที่สุดเป็นผลให้เกิดการอิมโพชั่น เนื่องจากยังเกิดกระบวนการคาวิเทชัน และ อิมโพชั่น มากเท่าไร ก็จะดีสำหรับอุปกรณ์ที่นำมาทำความสะอาด ที่มีรูปร่างที่ไม่ราบเรียบ

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

เมื่อหัวอัลตราโซนิคทรานสดิวเซอร์ได้สร้างคลื่นเสียงย่านอัลตราโซนิค และได้ถูกส่งผ่านตัวกลางที่ไม่ยืดหยุ่นเช่น น้ำหรือของเหลวทั่วไป จะมีการเปลี่ยนแปลงอย่างต่อเนื่องในช่วงเวลาที่แอมพลิจูด หรือความดัง (Loudness) ของเสียงนั้นต่ำ

ขณะที่แอมพลิจูดของเสียงเพิ่มขึ้นนั้น ขนาดของความดันที่เป็นลบ (Negative Pressure) ในบริเวณที่คลื่นขยายตัว (Rarefaction) จะเป็นสาเหตุที่เพียงพอจะทำให้ของเหลวแตกตัวออกในที่สุดซึ่งเป็นปรากฏการณ์ที่เรียกว่า คาวิเทชัน (Cavitation) จะแกว่ง (Oscillate) ภายใต้อิทธิพลของความดันที่เป็นบวก และจะโตขึ้นจนมีขนาดที่ไม่มีเสถียรภาพ ในที่สุดจะเกิดการยุบตัวลงอย่างรุนแรงของคาวิเทชัน เป็นผลให้เกิดการอิมโพลชัน (Implosion) ซึ่งจะทำให้คลื่น Shock Wave แผ่กระจายออกไปจากบริเวณที่ยุบตัว (Collapse) การยุบตัว (Collapse) และการอิมโพลชันของคาวิเทชันที่เกิดขึ้นจำนวนมากในของเหลวนั้นเป็นผลที่เกี่ยวข้องเนื่องมาจากการกระตุ้นของคลื่นอัลตราโซนิคนั่นเอง

กระบวนการคาวิเทชัน และ อิมโพลชัน ของคลื่นอัลตราโซนิค นั้นจะกำจัดชั้นที่อึดตัวออกไปได้อย่างดี ทำให้สารทำลายใหม่ที่จะเข้ามาทำลายสามารถเข้ามาขจัดสิ่งสกปรกได้ต่อไปซึ่งจะเป็นประโยชน์ โดยเฉพาะอย่างยิ่งเมื่อจะทำความสะอาดบริเวณพื้นผิวที่ไม่เรียบ

1.5 การเปรียบเทียบระหว่างวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน

การทำความสะอาดด้วยคลื่นอัลตราโซนิค อุปกรณ์ที่ใช้สร้างคลื่นอัลตราโซนิคเพื่อให้เกิดกระบวนการทำความสะอาดเกิดขึ้นคือ หัวอัลตราโซนิคทรานสดิวเซอร์ ซึ่งอัลตราโซนิคทรานสดิวเซอร์ทำงานก็ต่อเมื่อ ป้อนสัญญาณที่ความถี่เรโซแนนท์ของหัว โดยแต่เดิมได้ป้อนสัญญาณแบบ PWM โดยความถี่เรโซแนนท์ได้มีหลายความถี่เนื่องจาก แต่ละความถี่ก็จะเหมาะสมกับอุปกรณ์ที่แตกต่างกัน และกระบวนการทำความสะอาดก็แตกต่างกันด้วย ซึ่งจะได้กล่าวต่อไปในเรื่องของกระบวนการทำความสะอาด ต่อมาได้มีการพัฒนาเครื่องทำความสะอาด โดยได้ติดตั้งหัวอัลตราโซนิคทรานสดิวเซอร์หลายๆความถี่เรโซแนนท์ได้แยกกัน เพื่อให้เครื่องสามารถครอบคลุมในการทำความสะอาดอุปกรณ์ได้หลายชนิด [4] แต่ต้องสร้างตัวขับหัวอัลตราโซนิคทรานสดิวเซอร์ขึ้นมาหลายความถี่ด้วย ทำให้แพงขึ้นเพราะจำนวนอุปกรณ์เพิ่มขึ้นมา วิทยานิพนธ์ฉบับนี้ได้นำเสนอโดยการป้อนสัญญาณแบบ PS-PWM และ PDM เพื่อศึกษากระบวนการทำความสะอาด และประสิทธิภาพของแต่ละรูปแบบสัญญาณเพื่อเปรียบเทียบกัน

1.6 ขอบเขตการวิจัย

ในวิทยานิพนธ์ฉบับนี้ได้นำเสนอวิธีการป้อนรูปแบบสัญญาณ PS-PWM และ PDM เปรียบเทียบกับรูปแบบสัญญาณแบบเดิมที่ใช้ PWM แล้วทำการวัดรูปของสัญญาณ แรงดัน กระแส วัดค่าฮาร์มอนิกและประสิทธิภาพของแต่ละรูปแบบสัญญาณที่ใช้ในหัวอัลตราโซนิคทรานสดิวเซอร์

1.7 ขั้นตอนของการศึกษา

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

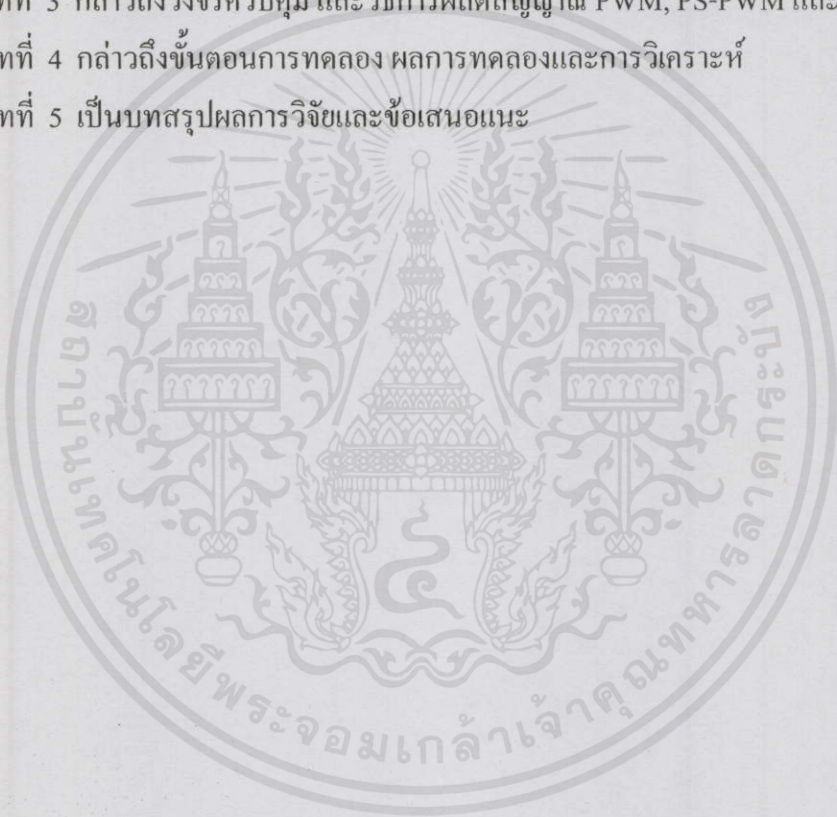
บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการทำความสะอาดด้วยคลื่นอัลตราโซนิคโดยจะกล่าวถึง คลื่นเสียง อุปกรณ์ที่ใช้ในการสร้างคลื่นเสียงย่านอัลตราโซนิคและกระบวนการทำความสะอาดที่เกิดขึ้น ไดโอด ซีเนอร์ไดโอด เพาเวอร์มอสเฟต พีดีบีลิวเอ็มฟลูบริคจ์คอนเวอร์เตอร์ พีดีบีลิวเอ็มแบบเลื่อนเฟสฟลูบริคจ์คอนเวอร์เตอร์ และ พีดีเอ็มฟลูบริคจ์คอนเวอร์เตอร์

บทที่ 3 กล่าวถึงวงจรควบคุม และวิธีการผลิตสัญญาณ PWM, PS-PWM และ PDM

บทที่ 4 กล่าวถึงขั้นตอนการทดลอง ผลการทดลองและการวิเคราะห์

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีการทำความสะอาดด้วยคลื่นอัลตราโซนิค

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องในการวิจัย และพื้นฐานของระบบ การทำความสะอาดด้วยคลื่นอัลตราโซนิค อุปกรณ์ในเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิค วงจรและ การสร้างรูปแบบสัญญาณให้หัวอัลตราโซนิคทรานสดิวเซอร์ที่ใช้ทำการทดลองใน วิทยานิพนธ์นี้

2.1 ทฤษฎีคลื่นอัลตราโซนิค

คลื่นอัลตราโซนิค (Ultrasonic) คือคลื่นเสียงที่อยู่เหนือกว่าขีดจำกัดที่มนุษย์ สามารถได้ยินหรือรับฟังได้ ความถี่ของคลื่นเสียงนั้นเป็นสิ่งที่แสดงถึงระดับ (Pitch) หรือความสูงต่ำ (Tone) ของมันเอง โดยที่ความถี่ต่ำจะให้ระดับเสียงที่ต่ำ (Bass Tone) ความถี่สูงจะให้ระดับเสียงที่สูง (Treble Tone)

คลื่นเสียงอัลตราโซนิคนั้นก็คือคลื่นเสียงที่มีระดับเสียง และความถี่สูงเกินกว่าความสามารถของมนุษย์จะรับฟังได้ด้วยหู ความถี่ที่สูงกว่า 18 kHz โดยทั่วไปจะพิจารณาให้เป็นคลื่น อัลตราโซนิค สำหรับความถี่คลื่นอัลตราโซนิคที่ใช้สำหรับทำความสะอาดนั้นจะอยู่ในช่วง 20,000 รอบต่อวินาทีหรือ 20 kHz ถึงเกินกว่า 100,000 kHz

ในการใช้งานทำความสะอาดทั่วไปส่วนใหญ่ในอุตสาหกรรมนั้น จะใช้ความถี่ในช่วง 20 kHz ถึง 50 kHz ความถี่ที่สูงกว่า 50 kHz จะใช้โดยทั่วไปในงานการทำความสะอาดขนาดเล็กที่ ละเอียด ตั้งโต๊ะได้ เช่น ในร้านเพชรพลอย และร้านหม้อพื้น

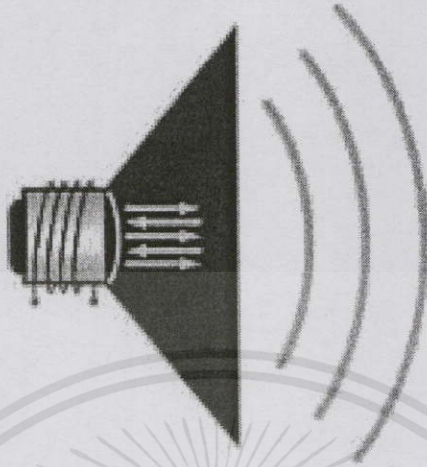
เพื่อที่จะเข้าใจถึงกลไกการทำงานของคลื่นอัลตราโซนิคได้นั้น จำเป็นต้องมีพื้นฐานความ เข้าใจในเรื่องคลื่นเสียงว่ามันกำเนิดขึ้นได้อย่างไร และมันสามารถเดินทางผ่านตัวนำหรือตัวกลาง ได้อย่างไรด้วย

ในพจนานุกรมนั้นให้คำจำกัดความคลื่นเสียงว่าเป็นการเคลื่อนที่ของการสั่นสะเทือน ส่งผ่านตัวกลางที่สามารถยืดหยุ่นได้ เช่น ของแข็ง ของเหลว หรืออากาศ

คลื่นเสียงจะกำเนิดขึ้นได้นั้น ก็ต่อเมื่อมีการเคลื่อนที่ของตัวกลางนำคลื่นเสียงที่เกิดซ้ำ หลาย ๆ ครั้ง หรือแค่ครั้งเดียว เช่น การเกิดฟ้าผ่า หรือการเคลื่อนที่ของการสั่นสะเทือน

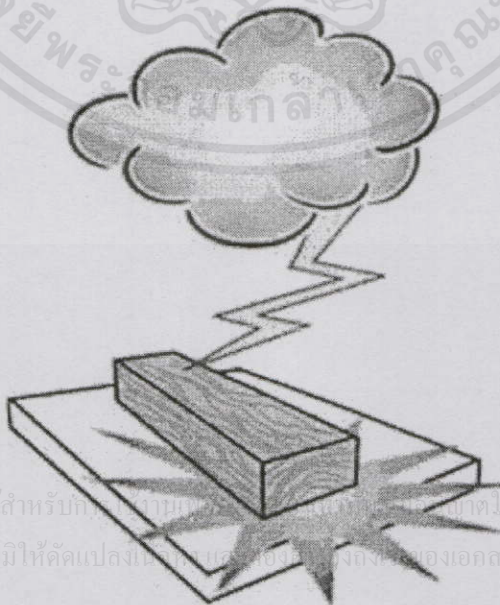
การเคลื่อนที่ของอากาศที่บริเวณ Cone ของลำโพง นั้นเป็นตัวอย่างที่ดีสำหรับอธิบายการ สั่นสะเทือนที่เกิดจากการเคลื่อนไหวทางกล ขณะที่ Cone ของลำโพงเคลื่อนที่ถอยหลังและ เดินหน้านั้น อากาศทางด้านหน้าของ Cone จะถูกอัดตัวและขยายตัวเพื่อทำให้คลื่นเสียงออกมา ซึ่ง

จะเดินทางตามอากาศมาเรื่อย ๆ จนกระทั่งจะหายไปในที่สุด โดยส่วนมากเรามักจะรู้จักคลื่นเสียงว่าเป็นแบบชนิดนี้ซึ่งกำเนิดจากการเคลื่อนไหวทางกล ดังรูปที่ 2.1



รูปที่ 2.1 คลื่นเสียงที่เกิดจากการเคลื่อนที่ของการสั่นสะเทือน

คลื่นเสียงอีกแบบที่กำเนิดจากปรากฏการณ์ฟ้าผ่า หรือที่เรียกว่า Shock นั้นเกิดจากการเปลี่ยนแปลงของปริมาณความจุของอากาศอย่างทันทีทันใดซึ่งเป็นผลมาจากการคายประจุไฟฟ้าในอากาศ ตัวอย่างอื่นของการ Shock นี้ จะเป็นแหล่งกำเนิดของคลื่น Single Compression ที่แพร่กระจายออกไปจากแหล่งกำเนิด ดังรูปที่ 2.2



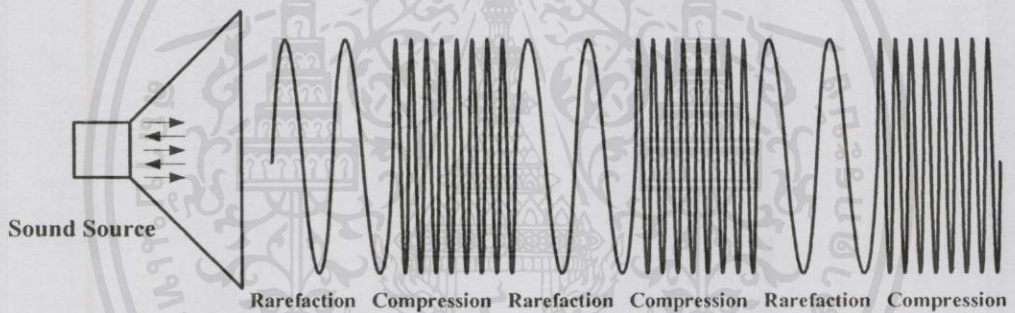
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับศึกษาเท่านั้น ไม่สามารถนำออกไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาเอกสารไปลงนิตยสารหรือสิ่งพิมพ์ของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.2 คลื่นเสียงที่เกิดจากปรากฏการณ์ฟ้าผ่า

ตามรูปที่ 2.3 จะใช้ขดลวดสปริงเป็นสิ่งที่ใช้แทน เพื่อแสดงถึงแต่ละโมเลกุลของตัวกลางนำคลื่นเสียง โดยแต่ละโมเลกุลในตัวกลางนั้นจะได้รับอิทธิพลจากโมเลกุลเคียงข้างที่ติดๆ กัน เช่นเดียวกันกับขดลวดของสปริงก็จะส่งผลกระทบต่อขดลวดที่อยู่ข้างเคียงด้วยเหมือนกัน

แหล่งกำเนิดคลื่นเสียงตามแบบจำลองนั้นจะอยู่ทางด้านซ้าย การอัด (Compression) ที่ถูกสร้างจากแหล่งกำเนิดคลื่นเสียงจะเป็นเช่นเดียวกับการผลัดกันของขดลวดที่อยู่ติดกัน สิ่งสำคัญอย่างหนึ่งที่ต้องทราบคือ ถึงแม้ว่าคลื่นจะเดินทางจากปลายข้างหนึ่งของสปริงไปยังอีกข้างหนึ่งก็ตามแต่ละขดลวดก็จะยังคงรักษาตำแหน่งเดิมของมัน ด้วยเหตุผลนี้แต่ละขดลวดของสปริงในส่วนแรกจะเกิดการอัดตัว (Compression) จากการผลัดกันของขดลวดที่ติดกัน และต่อจากนั้นจะเป็นส่วนของการขยายตัว (Rarefaction) ซึ่งขดลวดที่ติดกันจะถอยออกห่างจากกัน

ณ ตำแหน่งใดๆ ก็ตามในตัวกลางนำคลื่นเสียงจะถูกทำให้มีการอัดตัว และจากนั้นจะขยายตัวสลับกันไป บริเวณที่มีการอัดตัวนั้นความดันในตัวกลางจะเป็นบวก และบริเวณที่มีการขยายตัวความดันของตัวกลางจะเป็นลบ



รูปที่ 2.3 ธรรมชาติของคลื่นเสียงที่มีช่วงอัดและขยาย

2.2 การใช้คลื่นอัลตราโซนิคทำความสะอาด

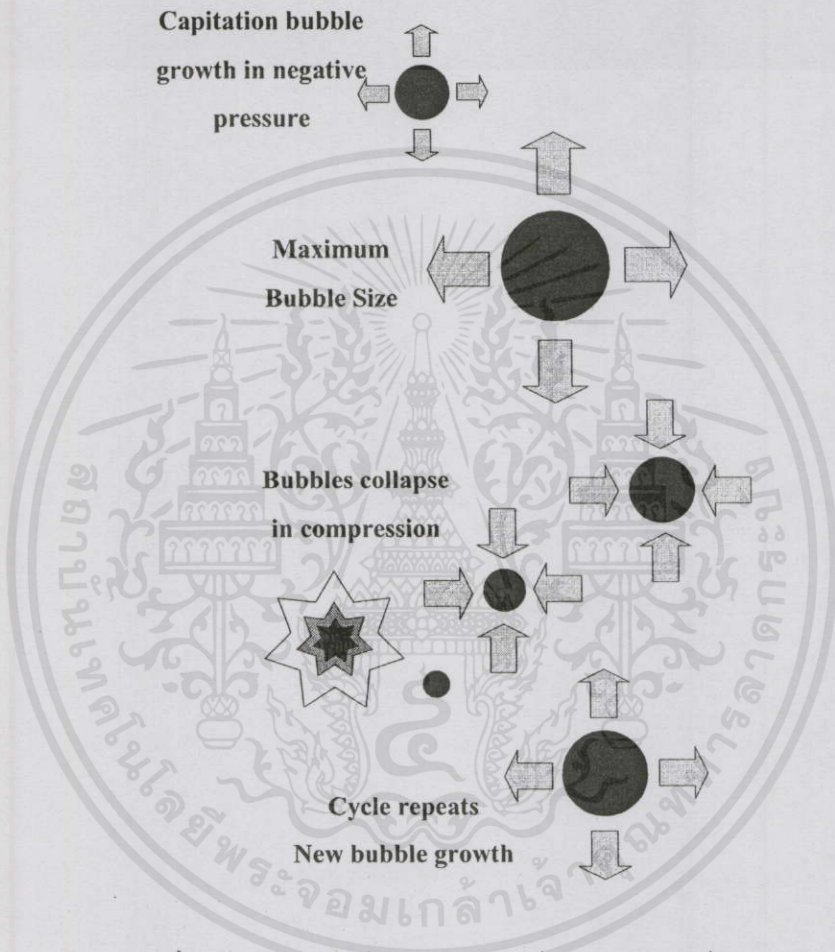
2.2.1 CAVITATION AND IMPLOSION

ในสื่อตัวกลางที่ยืดหยุ่น เช่น อากาศ และของแข็งเกือบทั้งหมดนั้น จะมีการเปลี่ยนแปลงอย่างต่อเนื่องเกิดขึ้นในขณะที่คลื่นเสียงถูกส่งผ่าน ส่วนในตัวกลางที่ไม่ยืดหยุ่น เช่น น้ำหรือของเหลวทั่วไปจะมีการเปลี่ยนแปลงอย่างต่อเนื่องในช่วงเวลาที่แอมพลิจูดหรือความดัง (Loudness) ของเสียงนั้นต่ำ

จากรูปที่ 2.4 ขณะที่แอมพลิจูดของเสียงเพิ่มขึ้นนั้น ขนาดของความดันที่เป็นลบในบริเวณที่คลื่นขยายตัว จะเป็นสาเหตุที่เพียงพอจะทำให้ของเหลวแตกตัวออกในที่สุด ซึ่งเป็นปรากฏการณ์ที่เรียกว่า คาวิเตชัน (Cavitation) จะแกว่ง (Oscillate) ภายใต้อิทธิพลของความดันที่เป็นบวก และจะโตขึ้นจนมีขนาดที่ไม่มีเสถียรภาพ ในที่สุดจะเกิดการยุบตัวลงอย่างรุนแรงของคา

วิเตชัน เป็นผลให้เกิดการ อิมโพชั่น (Implosion) ซึ่งจะทำให้คลื่น Shock Wave แผ่กระจายออกไป จากบริเวณที่ยุบตัว (Collapse) การยุบตัว และการอิมโพชั่นของควิเตชัน ที่เกิดขึ้นจำนวนมากใน ของเหลว นั้นเป็นผลที่เกี่ยวข้องเนื่องมาจากการกระตุ้นของคลื่นอัลตราโซนิคนั่นเอง

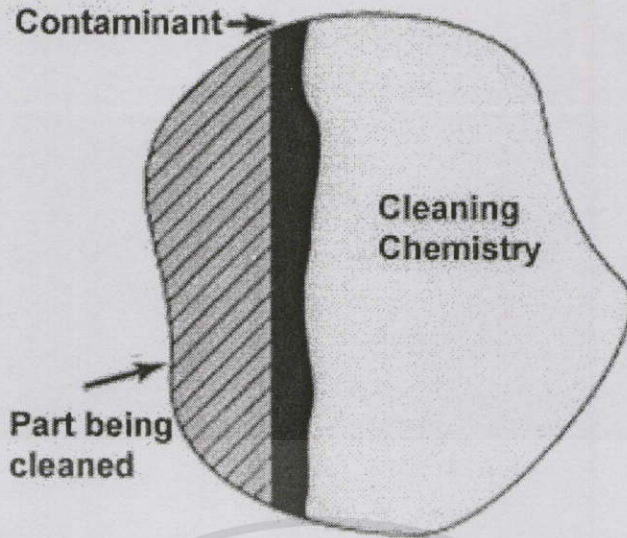
ได้มีการคำนวณว่า บริเวณที่เกิดอิมโพชั่น ของฟองอากาศควิเตชันนั้น จะมีอุณหภูมิที่สูงกว่า 10,000 องศาฟาเรนไฮด์ และมีความดันที่สูงกว่า 10,000 PSI



รูปที่ 2.4 กระบวนการการเกิดควิเตชัน และ อิมโพชั่น

2.2.2 ประโยชน์ของคลื่นอัลตราโซนิค ในกระบวนการทำความสะอาดและการชำระล้าง

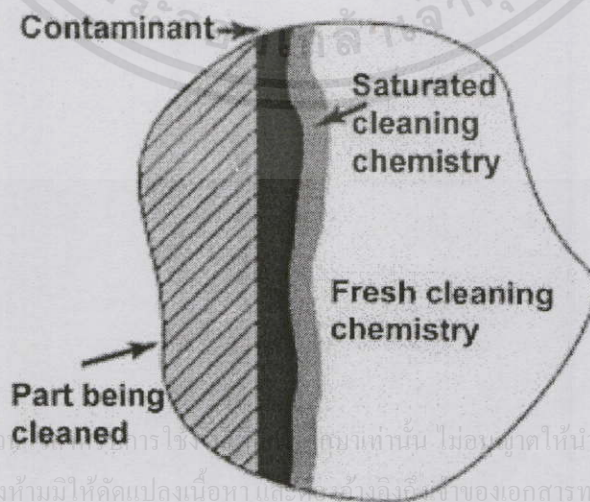
ในการทำความสะอาดนั้น สิ่งสกปรกจะถูกกำจัดโดยการทำละลายรวมตัว หรือโดยการเคลื่อนย้ายขจัดออกทิ้ง หรือทั้งสองอย่างเกิดขึ้นพร้อมกัน ขึ้นอยู่กับว่าสารละลายที่ใช้ นั้นสามารถละลายสิ่งสกปรกเหล่านั้นได้หรือไม่ ผลที่เกิดทางกลเนื่องจากพลังงานของคลื่นอัลตราโซนิค นั้นสามารถช่วยเพิ่มความเร็วของการรวมตัวหรือการขจัดออกทิ้งของสิ่งสกปรก ซึ่งจะเป็นประโยชน์ ในการทำความสะอาดรวมทั้งกระบวนการชำระล้างด้วย สารเคมีที่หลุดตกค้างจากการทำความสะอาด จะถูกกำจัดออกอย่างรวดเร็วโดยสิ้นเชิงด้วยกระบวนการชำระล้างของอัลตราโซนิค



รูปที่ 2.5 หน้าสัมผัสของชิ้นงานที่มีคราบสกปรกติดอยู่โดยชิ้นงานแช่อยู่ในน้ำยาทำความสะอาด [4]

การขัดสิ่งสกปรกออกด้วยการทำลายรวมตัวนั้น สารเคมีที่เป็นตัวทำลายต้องเข้าไปถึงสิ่งสกปรกและทำลายสิ่งสกปรกออก กระบวนการทำความสะอาดจะเกิดบริเวณพื้นผิวสัมผัสระหว่างสารเคมีที่ทำความสะอาดกับสิ่งสกปรกเท่านั้น ดังรูปที่ 2.5

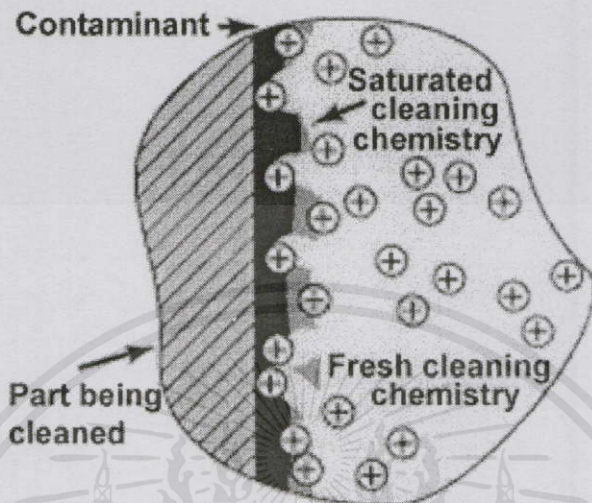
ขณะที่ตัวทำลายที่ใช้ทำความสะอาดกำลังรวมตัวทำลายสิ่งสกปรกอยู่นั้น จะเกิดขึ้นของสารละลายหยุดลง ตัวทำลายไม่สามารถจับตัวกับสิ่งสกปรกได้อีก เนื่องจากเกิดการอิมตัวของสารละลายที่ทำความสะอาด และยังกั้นไม่ให้ตัวทำลายตัวใหม่ที่จะเข้ามาทำลายสามารถเข้ามาได้ ดังรูปที่ 2.6



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้ใช้เฉพาะเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและเผยแพร่ข้อมูลของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.6 น้ำยาที่ติดกับคราบสกปรกเกิดการอิมตัว และแสดงให้เห็นว่าน้ำยาที่ยังไม่อิมตัวเข้าไม่ถึง [4]

กระบวนการควิเตชันและอิมโพชั่นของคลื่นอัลตราโซนิคนั้นจะกำจัดชั้นที่อึดตัวออกไปได้อย่างดี ทำให้สารทำลายใหม่ที่จะเข้ามาทำลายสามารถเข้ามาจับสิ่งสกปรกได้ต่อไป ซึ่งจะเป็ประโยชน์ โดยเฉพาะอย่างยิ่งเมื่อจะทำความสะอาดบริเวณพื้นผิวที่ไม่เรียบ [4] ดังรูปที่ 2.7

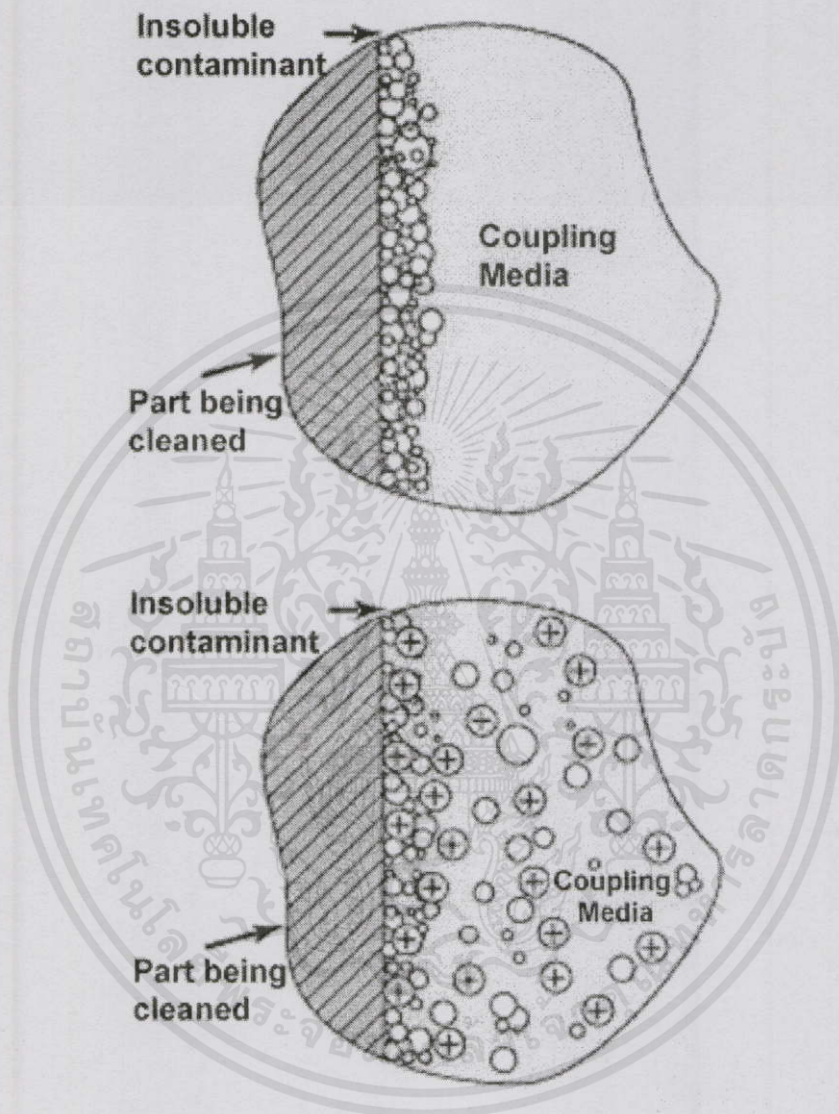


รูปที่ 2.7 กระบวนการควิเตชัน และอิมโพชั่น ช่วยให้เกิดการหมุนเวียนของน้ำยาทำความสะอาด [4]

ในระหว่างที่คลื่นเสียงมีความกดดันที่เป็นลบนั้น (Negative Pressure) ของเหลวจะแตกตัวออก และฟองอากาศควิเตชันก็จะเริ่มมีการจัดวางตัว ขณะที่ Negative Pressure ภายในฟองอากาศเพิ่มขึ้นนั้น ก๊าซที่ถูกทำลาย (Dissolved Gas) ในของเหลวที่มีควิเตชันจะเริ่มแพร่กระจายเข้ามาเข้าไปในฟองอากาศ ขณะที่ Negative Pressure ถูกลดลงในช่วงการขยายตัวของคลื่นเสียงและมีความดันสูงขึ้นจนถึงความดันบรรยากาศแล้วนั้น ฟองอากาศจะเริ่มยุบตัวลงเนื่องจากแรงดึงผิวของมันเองที่ตกลงมา

ในระหว่างช่วงการอัดตัวของคลื่นเสียง ก๊าซที่ฟุ้งกระจายอยู่ภายในฟองอากาศจะถูกอัดตัวและในที่สุดก็จะเริ่มแพร่กระจายเข้ามาในฟองอากาศอีกครั้ง อย่างไรก็ตามกระบวนการนี้ จะไม่สามารถสำเร็จได้จนกว่าฟองอากาศจะบรรจุเต็มไปด้วยก๊าซ ตั้งแต่ที่ก๊าซยังไม่เริ่มแพร่เข้ามาในฟองอากาศ กระทั่งฟองอากาศถูกอัดตัว และเมื่อฟองอากาศถูกอัดตัวแล้ว พื้นผิวหน้ารอบ ๆ ฟองอากาศที่ก๊าซสามารถแพร่เข้ามาได้จะถูกลด ด้วยเหตุผลนี้ฟองอากาศควิเตชันที่บรรจุก๊าซอยู่ซึ่งถูกจัดวางตัวอยู่ในของเหลว จะไม่ยุบตัวลงให้เกิดอิมโพชั่น แต่จะอยู่ในรูปก๊าซที่ถูกอัดตัวให้มีขนาดเล็กๆ ในของเหลวมากกว่า ปรากฏการณ์นี้จะเป็นประโยชน์อย่างยิ่งในการลดขนาดของก๊าซในของเหลวลง ฟองอากาศขนาดเล็ก ๆ ที่มีก๊าซอยู่ภายในจะรวมกลุ่มเข้าด้วยกันจนกว่าจะเพียงพอให้มันลอยตัวขึ้นเหนือผิวหน้าของของเหลวในที่สุด

รูปที่ 2.8 สิ่งสกปรกบางชนิดจะประกอบด้วย อนุภาคของสารที่ไม่สามารถทำละลายได้ ซึ่งจะเกาะติดกันอย่างหลวมๆ ด้วยแรงจากพันธะ Ionic และ Cohesive Force อนุภาคเหล่านี้สามารถกำจัดออกมาได้โดยเพียงแค่แยกแรงที่ดึงดูดมันเอาไว้ออกจากกัน



รูปที่ 2.8 อนุภาคที่ไม่ละลายในสารละลายซึ่งเกาะติดชิ้นงานอยู่ [4]

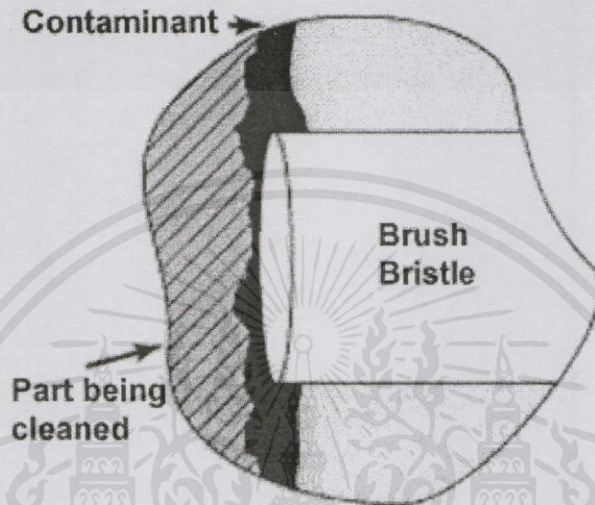
กระบวนการควิเทชั่น และอิมโพชั่น ซึ่งเป็นผลที่ได้จากการใช้คลื่นอัลตราโซนิคจะกำจัดและเคลื่อนย้ายสิ่งสกปรกที่เกาะติดอย่างหลวมๆ เช่น ฝุ่นละอองที่ติดอยู่บริเวณพื้นผิว เพื่อที่จะให้ได้ประสิทธิภาพที่ดีขึ้นนั้น ตัวกลาง Coupling Medium จำเป็นต้องสามารถทำให้อนุภาคเปื่อยเพื่อที่จะกำจัดออกไปได้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิใช้คลื่นเสียงเหนือหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

สิ่งสกปรกต่าง ๆ นั้น ตามธรรมชาติแล้วจะมีความซับซ้อน ซึ่งประกอบด้วยการรวมกันของส่วนที่สามารถทำละลายได้และส่วนที่ไม่ทำละลาย ผลที่ได้จากคลื่นอัลตราโซนิค นั้นมีอย่าง

มากมาย เช่นในกรณีนี้คือ มันจะไปกระตุ้นกลไกการทำงาน ช่วยให้การทำละลายของสิ่งสกปรกที่ละลายได้ และการขจัดอนุภาคที่ไม่ทำละลายออกไป เกิดได้เร็วขึ้นกว่าเดิม

การทำงานของคลื่นอัลตราโซนิค ยังแสดงให้เห็นว่า ช่วยเพิ่มความเร็วหรือช่วยส่งเสริมในการเกิดปฏิกิริยาเคมีหลายๆ อย่างอีกด้วยและนี่เองอาจจะเป็นสาเหตุที่ทำให้ในบริเวณที่เกิดอิมโพซันจะมีระดับพลังงานที่สูง ซึ่งจะทำให้เกิดอุณหภูมิและความดันที่สูงมาก



รูปที่ 2.9 การทำความสะอาดโดยใช้แปรง [4]

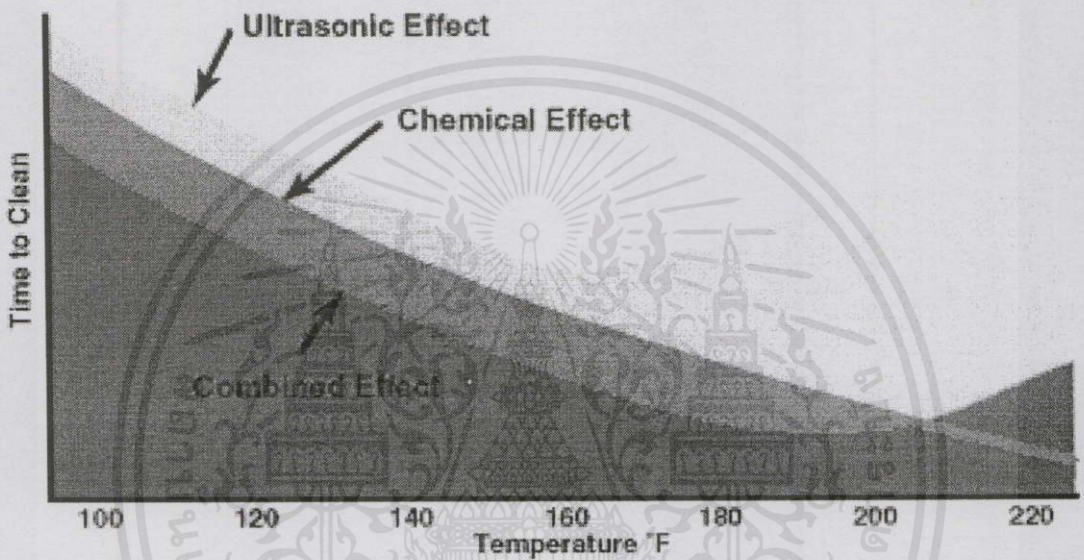
ตามที่ได้กล่าวและแสดงมาแล้วทั้งหมดนั้น พื้นผิวของส่วนที่จะถูกทำความสะอาด ได้ถูกกำหนดให้เป็นพื้นราบ ซึ่งในความเป็นจริงเป็นไปได้ยากมากที่จะแบนราบ แต่จะประกอบไปด้วยเนิน หลุม ซึ่งขรุขระ ตามรูปจะแสดงให้เห็นว่า ทำไมพลังงานจากคลื่นอัลตราโซนิค จึงมีประสิทธิภาพดีกว่า และทำไมจึงช่วยเสริมในการทำความสะอาดมากกว่าทางเลือกอื่น ๆ เช่น การฉีดสเปรย์ทำความสะอาด การใช้แปรงขัด (รูปที่ 2.9) การใช้ลมเป่า หรือแม้แต่ Electro Cleaning ความช่วยเหลือของคลื่นอัลตราโซนิค ในการทำความสะอาดพื้นผิวของส่วนที่เป็นเชิงซ้อน (Complex Part) นั้นดีกว่าและเป็นเรื่องที่ควรจะสนใจเป็นอย่างยิ่งอีกด้วย

น้ำยาทำความสะอาดที่เราเลือกนำมาใช้นับได้ว่าเป็นส่วนสำคัญ ซึ่งสัมพันธ์กับประสิทธิภาพ ในกระบวนการทำความสะอาดด้วยคลื่นอัลตราโซนิค น้ำยาที่เลือกใช้นั้น ควรที่จะสอดคล้องและเหมาะสมกับวัสดุที่เราจะทำความสะอาดและไม่ทำอันตรายต่อตัวชิ้นงาน ตลอดจนส่วนประกอบต่างๆ ที่อยู่บนชิ้นงานจึงจะมีประสิทธิภาพสูงในการทำความสะอาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้า ไม่อนุญาตให้นำไปใช้ประโยชน์ใดๆ โดยปกติ น้ำยาทำความสะอาดแทบทุกชนิด มักจะสามารถนำมาใช้ได้ แต่อย่างไรก็ตาม ไม่ว่าจะชนิดใดก็ตาม อีกทั้งยังมีให้คิดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ ควรจะหลีกเลี่ยงสารที่มีคุณสมบัติใกล้เคียงกับสารที่ใช้ในสเปรย์

สารละลายแต่ละชนิดจะเกิดกระบวนการควิเตชันที่รุนแรงที่สุด ได้ที่อุณหภูมิที่แตกต่างกันไป เช่นน้ำบริสุทธิ์จะเกิดกระบวนการควิเตชันที่รุนแรงที่สุดที่อุณหภูมิประมาณ 160 องศาฟาเรนไฮด์

แต่ถ้าในการพิจารณาว่าเราจะใช้อุณหภูมิที่ค่าเท่าไรนั้น ต้องพิจารณาจากเงื่อนไขอื่นด้วย จากรูปที่ 2.10 เช่น น้ำสะอาดมีอุณหภูมิที่ทำให้เกิดกระบวนการควิเตชันที่รุนแรงที่สุดที่ 160 องศาฟาเรนไฮด์ แต่อุณหภูมิที่เหมาะสมจะใช้งานจะอยู่ที่ 180-190 องศาฟาเรนไฮด์ เพราะผลของปฏิกิริยาทางเคมีและกระบวนการควิเตชัน จะมีผลทำให้อุณหภูมิสูงขึ้น



รูปที่ 2.10 ผลของอุณหภูมิต่างกันเนื่องจากปัจจัยต่างๆ [4]

ความหนืด (viscosity) ของสารที่ใช้ ควรจะมีค่าให้น้อยที่สุดเพื่อที่จะให้ได้มาซึ่งกระบวนการควิเตชันที่รุนแรงที่สุด เนื่องจากถ้าสารละลายมีความหนืดมาก มันจะไม่ตอบสนองต่อควิเตชันได้เร็วพอ และการสั่นก็จะเกิดน้อยด้วย ซึ่งความหนืดของสารนี้พบว่ามันจะลดลงเมื่ออุณหภูมิเพิ่มขึ้น

2.3 อัลตราโซนิกทรานสดิวเซอร์

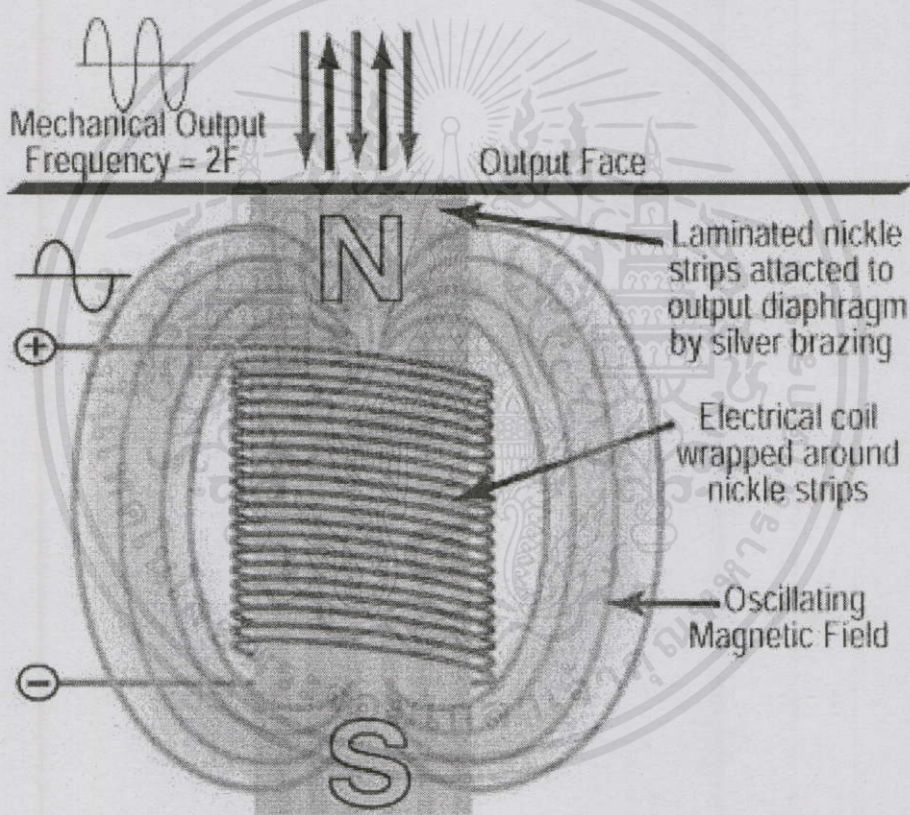
อัลตราโซนิกทรานสดิวเซอร์ คือ อุปกรณ์ซึ่งสามารถแปลงพลังงานกลโดยการสั่นไปมา ซึ่งทำให้เกิดคลื่นอัลตราโซนิกกระจายไปในวัตถุตัวกลาง หรือจากพลังงานของคลื่นอัลตราโซนิก แปลงไปเป็นพลังงานกลซึ่งสามารถทำให้เกิดแรงดันไฟฟ้าขึ้นได้จากตัวทรานสดิวเซอร์นั้น

ทรานสดิวเซอร์ที่ทำให้เกิดคลื่นอัลตราโซนิคมี 2 ชนิดใหญ่ ๆ คือ

1. เมคคานิกเกิลทรานสดิวเซอร์ (Mechanical Transducers) รูปที่ 2.11
2. อิเล็กโตรเมคคานิกเกิลทรานสดิวเซอร์ (Electromechanical Transducers)

Electromechanical Transducers จะเปลี่ยนพลังงานไฟฟ้าเป็นพลังงานเสียง ที่ใช้กันอยู่มี 4 ชนิด

1. แบบมูฟวิงคอยล์ (Moving Coil)
2. แบบอิเล็กโตรสแตติก (Electrostatic)
3. แบบเพียโซอิเล็กทริก (Piezoelectric)
4. แบบเฟอร์โรอิเล็กทริก (Ferroelectric)



รูปที่ 2.11 เมคคานิกเกิลทรานสดิวเซอร์ (Mechanical Transducers) [4]

แบบ Moving Coil นั้นคล้าย ๆ กับ Moving Coil ของลำโพง มีประโยชน์มากในการเป็นตัวสั่น (Vibrator) ชนิดริบบอน (Ribbon) ทำด้วยอะลูมิเนียมบาง ๆ วางไว้ระหว่างขั้วทั้งสองของแม่เหล็ก มันทำตัวเสมือนเป็นขดลวดของมูฟวิงคอยล์ ใช้ทำให้เกิดคลื่นอัลตราโซนิค

แบบ Electrostatic ใช้หลักการที่ว่า ในระหว่าง Parallel Plate Capacitor เมื่อเราให้ความต่างศักย์ระหว่างแผ่นทั้งสองจะทำให้เกิดแรงขึ้น หลักใหญ่ของมันจะใช้ทำเป็นลำโพงเสียงที่มีคุณภาพสูง แต่ก็ใช้ทำให้เกิดการสั่นแบบอัลตราโซนิคในของแข็ง

แบบ Piezoelectric และ Ferroelectric ในผลึกที่ประกอบด้วยโพล่าโมเลกุล (Polar Molecule) ซึ่งขาดจุดศูนย์กลางของความสมมาตร (Center of Symmetry) ถ้าตัดเอา Plate อันเนื่องมาจากผลึกนี้ แล้วทำให้เกิดการบิดงอ (Deform) ด้วยเครื่องมือกล จะทำให้เกิดความต่างศักย์ขึ้นระหว่างผิวหน้าทั้งสองของ Plate ปรากฏการณ์นี้เราเรียกว่า ไคเร็กเปียโซอิเล็กทริก เอฟเฟ็ค (Direct Piezoelectric Effect) ถ้าศักย์ระหว่างผิวหน้าทั้งสองของ Plate นี้เกิดบิดเบี้ยวไปได้เองเราเรียกปรากฏการณ์นี้ว่า อินเวอร์สเปียโซอิเล็กทริกเอฟเฟ็ค (Inverse Piezoelectric Effect)

สำหรับแบบอินเวอร์สเปียโซอิเล็กทริกเอฟเฟ็ค จะเกิดโดยให้สนามไฟฟ้าระหว่าง Plate ทั้งสองทำให้จุดศูนย์กลางของประจุทั้งสองเคลื่อนออกจากกันในทิศทางตรงกันข้าม ทำให้เซลล์บิดเบี้ยวไป ถ้ามีแผ่นควอทซ์ซึ่งตัดมาจากผลึกควอทซ์ ในทิศทางที่จะทำให้เกิดสั้นสะเทือนขนานกับสนามไฟฟ้าที่ใช้ เมื่อเราให้สนามไฟฟ้ากระแสสลับ Plate นี้จะขยายและหดตัวด้วยความถี่เดียวกันกับสนามไฟฟ้า ซึ่งจะเหมือนกับลูกสูบ ดังนั้นมันจะเป็นตัวกำเนิดคลื่นเสียงได้

ปัจจุบันมีสารไม่กี่ชนิดที่จะให้ปรากฏการณ์เปียโซอิเล็กทริก สารไคอิเล็กทริกทุกชนิดเป็นสารอิเล็กโทรสแตติก แต่ปกติแล้วผลนี้จะน้อยมาก แต่สารที่ให้ผลอย่างมากได้แก่ สารไททานาเท (Titanates) ซีคอนาเท (Zirconates) และในกรณีเช่นนี้เรียกว่า สารเฟอร์โรอิเล็กทริก ปกติสารเหล่านี้จะอยู่ในรูปของส่วนผสมเซรามิก ซึ่งประกอบไปด้วยโดเมน (Domain) ที่มีขั้วอิสระรวมตัวกันมากๆ ถ้านำสารแบบนี้มาสักหนึ่งชนิด เอาไปเผาให้อุณหภูมิสูงกว่าอุณหภูมิเคอริก (Curie Temperature) ของมัน แล้วทำให้เย็นภายใต้สนามไฟฟ้าสูงๆ ทิศทางของขั้วของโดเมนต่างๆ จะวางตัวไปแนวเดียวกับสนามและจะยังคงอยู่ต่อไปเช่นนั้น แม้ว่าเอาสนามไฟฟ้าออกไปนั้นแสดงว่ามันแสดงคุณสมบัติ เช่น Permanent Electric บางทีมันจะเป็นสารเปียโซอิเล็กทริก ถ้าหากว่ามันประพฤติตัวไปตามสมการของเปียโซอิเล็กทริก

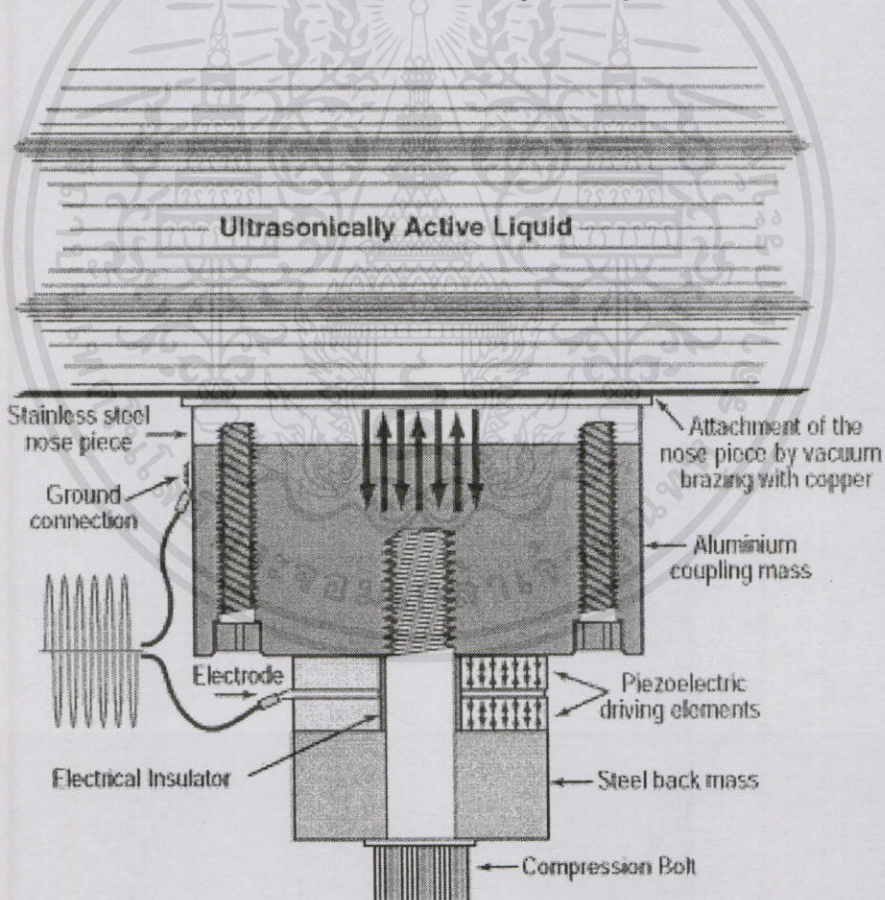
ทรานสดีวเซอร์ที่ใช้ในทางอัลตราโซนิคจะมีคุณสมบัติพิเศษคือ มีเปียโซอิเล็กทริกเอฟเฟ็ค และที่ใช้กันมากคือ ผลึกแควอทซ์ (Quartz) , แบเรียมติทานาเท (Barium titanate) และลีดเซอโคเนท (Lead zirconate titanate) สารเหล่านี้เมื่อนำมาทำทรานสดีวเซอร์ ยังแสดงคุณสมบัติเป็นตัวเก็บประจุทางไฟฟ้าด้วย ค่าตัวเก็บประจุเป็นไมโครฟารัดจะได้สัดส่วนกับค่าคงที่ไคอิเล็กทริก (dielectric constant , K) ของวัสดุคูณด้วยพื้นที่ และหารด้วยความหนาโครงสร้างของทรานสดีวเซอร์มีความคล้ายตัวเก็บประจุที่ว่า อิเล็กโตรดที่อยู่สองข้างถูกแยกโดยช่องที่มีสารเซรามิก คุณสมบัติบางประการของสารเซรามิกที่ได้รับความร้อน จะทำให้ไดโพล (dipole) เคลื่อนไหวได้โดยเสรีและที่จุดนี้เมื่อได้รับศักย์ไฟฟ้าเข้าไป ก็จะมีการจัดเรียงตัวและมีคุณสมบัติเปียโซอิเล็กทริก ถ้าภายหลังเซรามิกที่มีคุณสมบัตินี้ถูกเผาให้ร้อนถึงอุณหภูมิดังกล่าวอีกคุณสมบัติเปียโซอิเล็กทริกจะหมดไป ดังนั้นความร้อนจะสามารถทำลาย ทรานสดีวเซอร์ได้ โดยเฉพาะถ้าอุณหภูมิสูงเกินเคอริก

ค่าคงที่ไดอิเล็กตริกเป็นค่าที่แสดงว่าทรานสดีวเซอร์ต้องการพลังงานไฟฟ้าเท่าใด จึงจะเปลี่ยนเป็นพลังงานกลได้ ทั้งแบเรียมติทานาเท และ ลีดเซอโคเนท มีประสิทธิภาพมากกว่าผลึกแคว

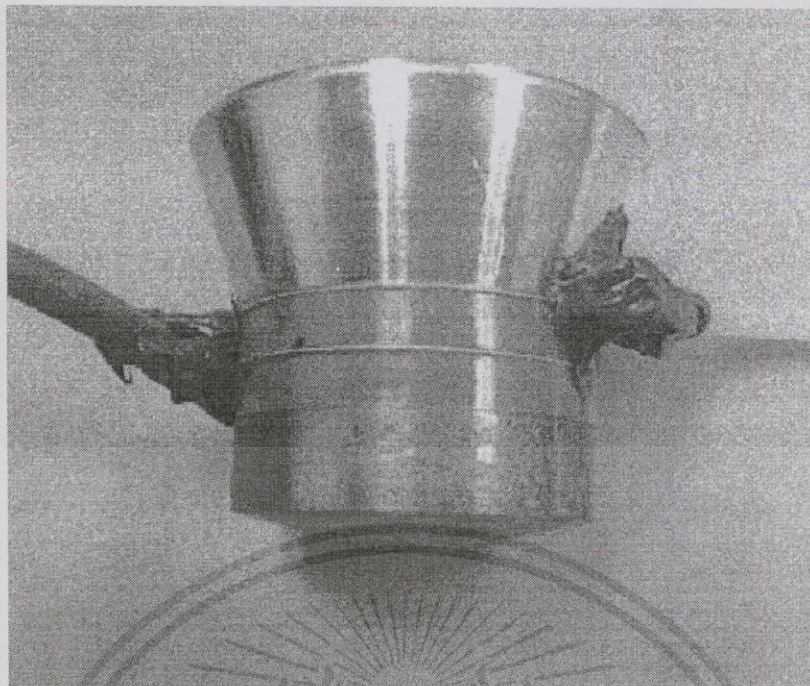
ควอทซ์ 5-6 เท่า เนื่องจากค่าคงที่ไดอิเล็กตริกต่ำมากและมีค่าคงที่ (Coupling constant) ต่ำมากจึงไม่ค่อยได้นำควอทซ์มาใช้ในเครื่องอัลตราโซนิกในปัจจุบัน

สำหรับวิทยานิพนธ์นี้ได้ใช้อัลตราโซนิกทรานสดิวเซอร์แบบ เปียโซอิเล็กทริกโครงสร้างภายในจะประกอบด้วย ชิ้นสารเปียโซอิเล็กทริก (Piezoelectric element) ซึ่งสามารถทำได้จากสารหลายชนิด เช่น ผลึกแร่ควอทซ์ (Quartz), แบเรียมทิตาเนท (Barium titanate) ,เกดลือร็อกเซล (Rochelle salt) ฯลฯ ดังรูปที่ 2.12 แต่ที่นิยมใช้กันอย่างแพร่หลายคือชิ้นสารเปียโซอิเล็กทริกที่ทำมาจากแบเรียมทิตาเนท ซึ่งได้จากการผสมสารติตาเนียมกับแบเรียมคาบอเนท ที่อุณหภูมิประมาณ $1300 - 1400^{\circ}\text{C}$ และผลิดออกมาในรูปของชิ้นสารเซรามิกซึ่งจะเป็นรูปสี่เหลี่ยมโดยส่วนใหญ่

จากชิ้นเซรามิกนี้จะถูกประกบด้วยแผ่นเพลท 2 แผ่น ซึ่งส่วนใหญ่ทำมาจากแผ่นเงิน เพื่อต่อเป็นขาของอิเล็กทรอนิกส์นั่นเอง ด้านข้างหนึ่งของแผ่นเพลทจะถูกยึดเพื่อไม่ให้เกิดการสั่น ส่วนอีกด้านจะยึดกับแผ่นไดอะแฟรมซึ่งสามารถเคลื่อนไหวไปมาได้ โดยแผ่นไดอะแฟรมนี้จะถูกสร้างให้มีลักษณะคล้ายเลนส์เพื่อรวมคลื่นให้มีความเร็วสูงขึ้น ดังรูปที่ 2.13



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิ
รูปที่ 2.12 เปียโซอิเล็กทริก (Piezoelectric) [4] ทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 อัลตราโซนิคทรานสดิวเซอร์แบบ เปียโซอิเล็กทริก (Piezoelectric)

ปรากฏการณ์แบบเปียโซอิเล็กทริก สามารถอธิบายได้คือ เมื่อสภาวะปกติยังไม่มี ความดันหรือคลื่นมาตกกระทบกับชิ้นสารเซรามิก ประจุไฟฟ้าซึ่งอยู่ภายในชิ้นสารจะเกิดการสมดุล ทำให้แผ่นเพลทเงินที่ประกบอยู่มีศักย์ไฟฟ้าที่เท่า ๆ กัน ซึ่งผลรวมของประจุไฟฟ้าบนแผ่นเพลทมีค่าเท่ากับศูนย์ แต่เมื่อมีความดันหรือคลื่นเสียงมาตกกระทบชิ้นสารเซรามิก จะทำให้เกิดการเปลี่ยนแปลงทางกายภาพของชิ้นสารคือ เกิดมีประจุตรงข้ามขึ้นบนแผ่นเพลทที่ประกบชิ้นสารเซรามิกอยู่ซึ่งผลของประจุไฟฟ้าตรงข้ามนี้ทำให้เกิดเป็นแรงดันตกคร่อมแผ่นเพลททั้งสอง ขนาดของแรงดันตกคร่อมนี้ จะขึ้นอยู่กับขนาดของความดันหรือคลื่นที่มาตกกระทบชิ้นสารเซรามิกที่ โกงงอไปมา

ทรานสดิวเซอร์แบบเปียโซอิเล็กทริก สามารถที่จะเปลี่ยนแปลงไปมาระหว่างพลังงานกล กับพลังงานไฟฟ้า ในกรณีการเปลี่ยนแปลงจากพลังงานไฟฟ้าเป็นพลังงานกล คือ ตัวส่งคลื่นอัลตราโซนิคซึ่งการทำงานคือ เมื่อป้อนแรงดันไฟฟ้ากระแสสลับ ซึ่งอาจมีความถี่ในช่วง 20 - 100 kHz แรงดันที่ป้อนนี้จะทำให้ชิ้นสารเซรามิกเกิดการเปลี่ยนแปลงรูปร่าง โดยที่ด้านหนึ่งของชิ้นสารเซรามิกจะถูกกดกับแผ่นไดอะแฟรม เมื่อชิ้นสารเซรามิกเกิดการบิดตัวไปมาจะทำให้แผ่นไดอะแฟรมนี้อัดกับอากาศโดยรอบเกิดเป็นคลื่นความถี่เดียวกับสัญญาณที่ป้อนเข้า ซึ่งคลื่นที่ผลิตได้จะมีกำลังเอาต์พุตสูงสุดเมื่อความถี่ที่ป้อนเข้ามาตรงกับความถี่เรโซแนนท์ ซึ่งเป็นความถี่ทางกลตามธรรมชาติของชิ้นสารเซรามิกชิ้นนั้น ส่วนความถี่อื่น ๆ นอกเหนือจากนี้จะทำให้กำลังเอาต์พุตมีค่าลดลงกว่าความถี่เรโซแนนท์มาก

ในทางกลับกันการเปลี่ยนจากพลังงานกลเป็นพลังงานไฟฟ้า คือ ตัวรับคลื่นอัลตราโซนิก โดยจะมองได้ว่าเป็นแหล่งจ่ายหนึ่งๆ เพราะมันไม่ต้องการแหล่งจ่ายอื่นๆ มาป้อนก็จะสามารถสร้างแรงดันออกมาได้ เพียงแต่มีขนาดแรงดันไม่สูงมากนัก โดยการทำงานของตัวรับคลื่นเกิดจากเมื่อมีความดันมาตกกระทบแผ่นไดอะแฟรม ซึ่งค่าความถี่ของคลื่นทำให้ชิ้นสารเซรามิกถูกบีบไปมา ซึ่งจะทำให้เกิดปรากฏการณ์เปียโซอิเล็กทริกบนชิ้นสาร ทำให้สามารถสร้างแรงดันไฟฟ้ากระแสสลับที่มีความถี่เดียวกับคลื่นที่มาตกกระทบได้

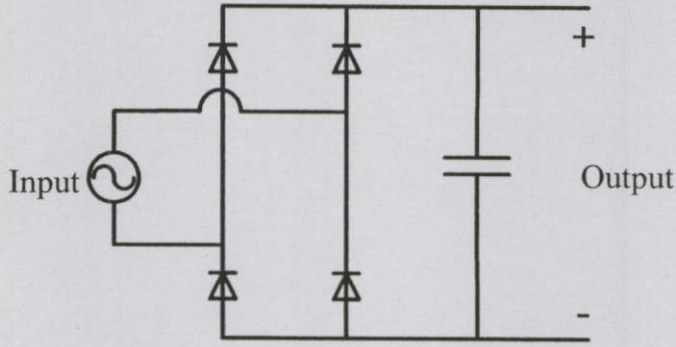
ระบบการทำความสะอาดด้วยคลื่นอัลตราโซนิก ประกอบด้วยทรานสดิวเซอร์(transducer) วงจรขับ แทงค์ (tank) และตัวกลางของเหลว ตัวทรานสดิวเซอร์ จะแปลงพลังงานที่ได้จากเครื่องกำเนิดให้เป็นการสั่นสะเทือนจากพลังงานเสียง ซึ่งการสั่นสะเทือนนี้จะถูกส่งผ่านแพร่กระจายไปตามแทงค์ และจะผลิตฟองอากาศวิเศษ (cavitations) ขึ้นมาในตัวกลางของเหลวภายในแทงค์ บรรจุ ฟองอากาศเหล่านี้จะจับตัว (formation) และยุบตัวลง (collapse) เพื่อให้เกิดปฏิกิริยาการขัดถู (scrubbing) ซึ่งจะใช้ในการกำจัดสิ่งสกปรกได้เป็นอย่างดี

ที่ได้กล่าวมาแล้วข้างต้นเป็นทฤษฎีเกี่ยวกับการเกิดเสียง พื้นฐานทางคลื่นเสียงย่านอัลตราโซนิก กระบวนการทำความสะอาดด้วยคลื่นอัลตราโซนิก และอัลตราโซนิกทรานสดิวเซอร์ แต่ระบบการทำความสะอาดด้วยคลื่นอัลตราโซนิก อุปกรณ์ที่ทำให้เกิดคลื่นเสียงย่านอัลตราโซนิกนั้น นอกจากอัลตราโซนิกทรานสดิวเซอร์แล้ว ยังต้องมีวงจรขับหัวอัลตราโซนิกทรานสดิวเซอร์ที่เป็นส่วนประกอบของวงจรขับดังจะได้อีกต่อไป

2.4 เรกติไฟาย และอินเวอร์เตอร์

2.4.1 เรกติไฟาย (Rectifier)

ภาคแหล่งจ่ายไฟฟ้ากระแสตรงจะประกอบไปด้วยวงจรเรกติไฟายที่ได้รับแรงดันไฟฟ้ากระแสสลับความถี่ 50 Hz จากแหล่งจ่ายการไฟฟ้า เมื่อผ่านวงจรเรกติไฟายจะทำการแปลงแรงดันไฟฟ้ากระแสสลับให้เป็นแรงดันไฟฟ้ากระแสตรง โดยวงจรเรกติไฟายนี้จะมีคุณสมบัติคือยอมให้กระแสไฟฟ้าไหลผ่านไปยังไหลกลับเพียงทิศทางเดียว โดยสามารถแบ่งวงจรออกเป็น 2 รูปแบบหลักๆคือ วงจรฮาล์ฟเวฟเรกติไฟายเออร์ (Half wave rectifier) และวงจรฟูลเวฟเรกติไฟายเออร์ (Full wave rectifier) ในวิทยานิพนธ์ได้นำวงจรฟูลเวฟเรกติไฟายเออร์ดังรูปที่ 2.14 มาใช้เนื่องจากให้แรงดันไฟตรงที่สูงกว่าและเรียบกว่ามาก เนื่องจาก วงจรแบบนี้จะมีกระแสไฟฟ้าเฉลี่ยด้านเข้าหรือด้านแหล่งจ่ายไฟฟ้ากระแสสลับมีค่าเป็นศูนย์ ซึ่งจะเป็ผลคือต่อหม้อแปลงไฟฟ้า และข้อดีอีกข้อคือจะมีค่าตัวประกอบระลอกคลื่น (Ripple factor) ต่ำกว่าเมื่อนำมาเปรียบเทียบกับวงจรฮาล์ฟเวฟเรกติไฟายเออร์

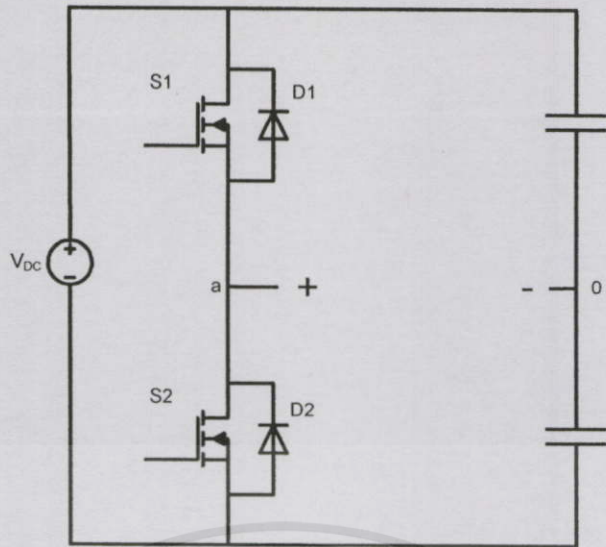


รูปที่ 2.14 วงจรฟลูเวฟเรกติไฟเออร์ (Full wave rectifier)

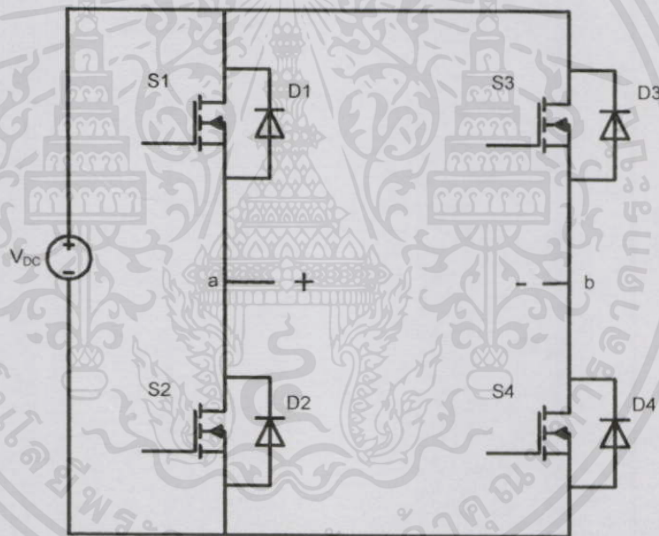
2.4.2 อินเวอร์เตอร์ (Inverter)

อินเวอร์เตอร์จะแบ่งออกเป็นสองชนิด [5] คือ แบบฮาล์ฟบริดจ์ (Half-bridge) และแบบฟลูบริดจ์ (Full-bridge) ในแบบฮาล์ฟบริดจ์ จะมีตัวเก็บประจุสองตัวต่อลำดับกันอยู่ระหว่างแหล่งจ่ายแรงดันไฟฟ้ากระแสตรงและหากกำหนดให้ตัวเก็บประจุสองตัวมีค่าเท่ากัน จะทำให้แรงดันไฟฟ้าตกคร่อมตัวเก็บประจุแต่ละตัวจะมีค่าเท่ากันคือ $V_d/2$ จุดกึ่งกลางแรงดันไฟฟ้า (จุด 0 ในรูปที่ 2.15 (a)) จะมีค่าคงที่เมื่อเทียบกับบัสลบ (N) ดังแสดงในรูปที่ 2.15 (a) ส่วนวงจรอินเวอร์เตอร์แบบฟลูบริดจ์ จะประกอบไปด้วยสองกึ่ง ดังรูปที่ 2.15 (b) โดยแบบฟลูบริดจ์จะมีกำลังไฟฟ้าสูงกว่าแบบฮาล์ฟบริดจ์สองเท่าจึงเหมาะที่จะเลือกใช้เมื่อต้องการจ่ายกำลังไฟฟ้าโหลดสูงขึ้น ซึ่งวิทยานิพนธ์นี้ได้เลือกใช้ อินเวอร์เตอร์แบบฟลูบริดจ์

เงื่อนไขสำคัญที่อินเวอร์เตอร์ทั้งสองแบบ คือการทำงานของสวิตช์ S1, S4 กับ S2, S3 จะต้องไม่ทำงานพร้อมกันในทุกช่วงเวลา มิฉะนั้นแล้วจะเกิดการลัดวงจรระหว่างบัสบวกกับบัสลบ ในอุดมคติเวลาการสวิตช์ของ S1, S4 กับ S2, S3 จะตรงข้ามกัน แต่ในทางปฏิบัติจะต้องการช่วงเวลาที่ไม่สวิตช์ทั้งคู่ไม่นำกระแส ซึ่งจะเรียกว่าเดดไทม์ (Dead time) โดยเดดไทม์จะอยู่ในช่วงเวลาก่อนจะเปลี่ยนสถานะการณีสวิตช์ จากนำกระแสเป็นไม่นำกระแสหรือจากไม่นำกระแสเป็นนำกระแส



(a)



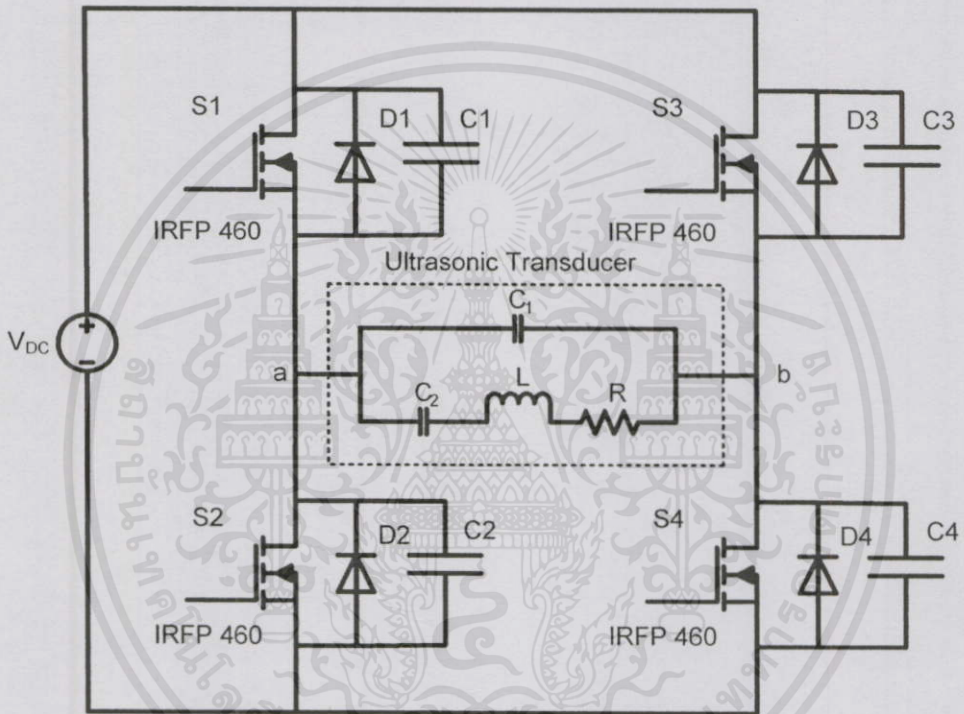
(b)

รูปที่ 2.15 อินเวอร์เตอร์ (Inverter) (a) แบบฮาล์ฟบริดจ์ (b) แบบฟูลบริดจ์

จากวงจรที่ได้กล่าวมาแล้วนำมาสร้างชุดขับหัวอัลตราโซนิคทรานสดิวเซอร์ โดย
 วิทยานิพนธ์นี้ได้ ทดลองใช้วงจรขับสัญญาณป้อนให้หัวอัลตราโซนิคทรานสดิวเซอร์ 3 รูปแบบ
 สัญญาณ คือ PWM PS-PWM และ PDM เพื่อเปรียบเทียบวงจร และรูปสัญญาณที่ใช้ในการ
 ทดลอง จะกล่าวดังต่อไปนี้

2.5 พีดีบีบลิวเอ็มฟลูบริดจ์คอนเวอร์เตอร์

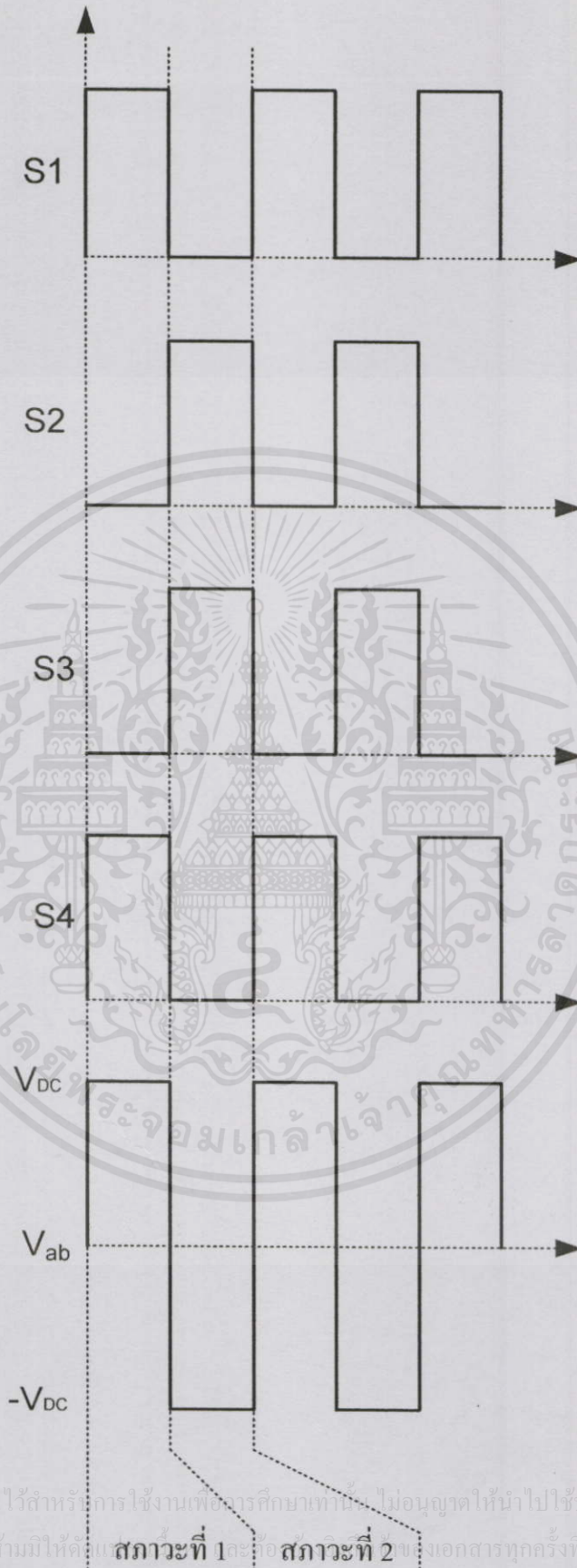
พีดีบีบลิวเอ็มฟลูบริดจ์คอนเวอร์เตอร์ (PWM Full - Bridge Converter) เป็นรูปแบบหนึ่งของแหล่งจ่ายกำลังงานแบบสวิตช์โหมคคอนเวอร์เตอร์ มีหลักการทำงานโดยอาศัยการสวิตช์ของอุปกรณ์สวิตช์ ส่งผ่านกำลังงานไฟฟ้าจากแหล่งจ่ายไฟฟ้ากระแสตรงไปสู่โหลด ซึ่งใช้อุปกรณ์สวิตช์ทั้งหมด 4 ตัวด้วยกัน ดังแสดงที่รูป 2.16 การทำงานของอุปกรณ์สวิตช์จะทำงานพร้อมกันเป็นคู่ๆ คือ S1 คู่กับ S4 และ S2 คู่กับ S3 ทำงานสลับกันคู่ละครั้งคาบเวลาการสวิตช์



รูปที่ 2.16 พีดีบีบลิวเอ็มฟลูบริดจ์คอนเวอร์เตอร์

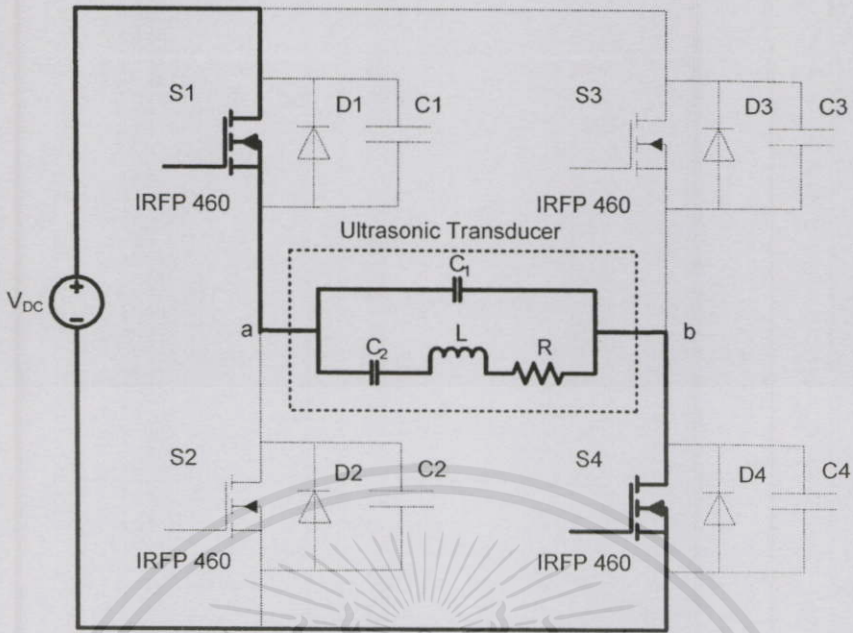
อุปกรณ์สวิตช์ที่ใช้คือ มอสเฟตทั้ง 4 ตัวจะทำงานแยกอิสระจากกัน โดยรับสัญญาณ PWM เพื่อจะทำงานเป็นคู่ๆ ให้ S1 และ S4 นำกระแสโดยที่ S2 และ S3 ยังคงหยุดนำกระแสอยู่ สลับกับการให้ S1 และ S4 หยุดนำกระแส แล้วก็ค่อย ให้ S2 และ S3 นำกระแส ทำอย่างนี้ไปเรื่อยๆ ดังรูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

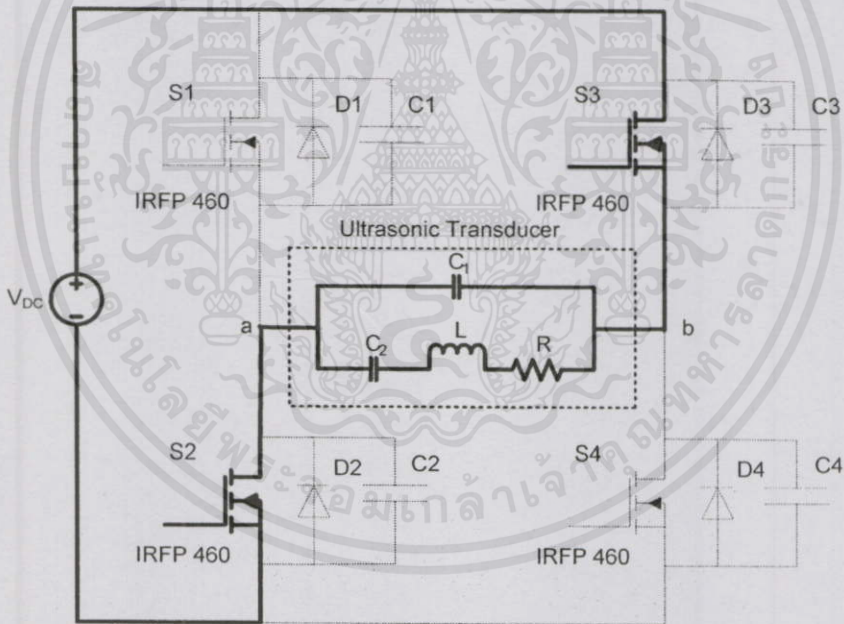


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกและเผยแพร่เอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.17 สัญญาณการทำงานของฟีดแบ็คลิฟต์เอ็มฟลูอิดจ็คนเวอร์เตอร์



สภาวะที่ 1



สภาวะที่ 2

รูปที่ 2.18 การทำงานของ PWM ที่สภาวะ 1 ถึง 2

เอกสารนี้เป็นการทำงานของ PWM แบ่งได้เป็น 2 สภาวะโดยการกำหนดการทำงานจากสัญญาณขับ
 ในเกทของอุปกรณ์สวิตช์ ดังแสดงในรูปที่ 2.17 รูปสัญญาณแรงดันเอาต์พุตของคอนเวอร์เตอร์ที่จุด
 ab มีลักษณะคล้ายสัญญาณสี่เหลี่ยม วงจรแบบเลื่อนเฟส ที่สภาวะการทำงานทั้ง 2 ดังรูปที่ 2.18

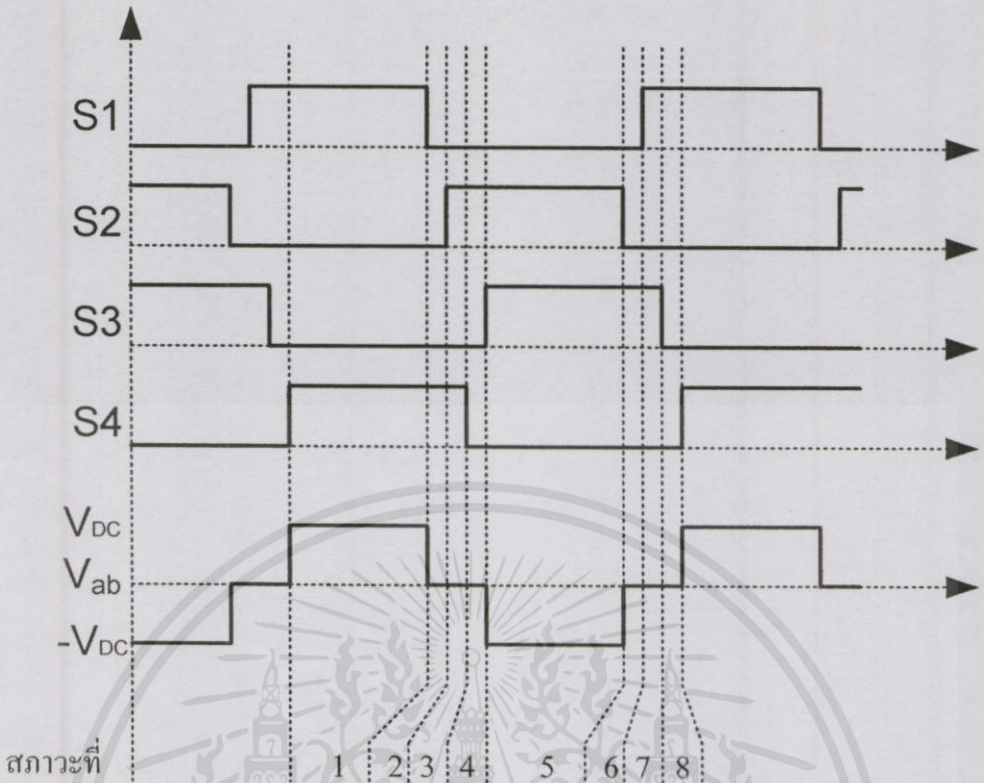
สภาวะที่ 1 สวิตช์ S1 และ S4 turn on กระแสจากแหล่งจ่ายกระแสตรงไหลผ่าน S1 ไปสู่โหลด และไหลผ่าน S4 กลับสู่แหล่งจ่ายครบวงจร แรงดันตกคร่อมโหลดเท่ากับแรงดันที่แหล่งจ่ายกระแสตรง

สภาวะที่ 2 S2 และ S3 turn on กระแสจากแหล่งจ่ายกระแสตรงไหลผ่าน S3 ไปสู่โหลด เมื่อกระแสไหลผ่านโหลด จะเกิดการสะสมพลังงาน จากนั้นไหลผ่าน S2 กลับสู่แหล่งจ่ายครบวงจร แรงดันตกคร่อมโหลดเท่ากับแรงดันที่แหล่งจ่ายแต่มีค่าติดลบเพราะกลับขั้วกับสภาวะที่ 1

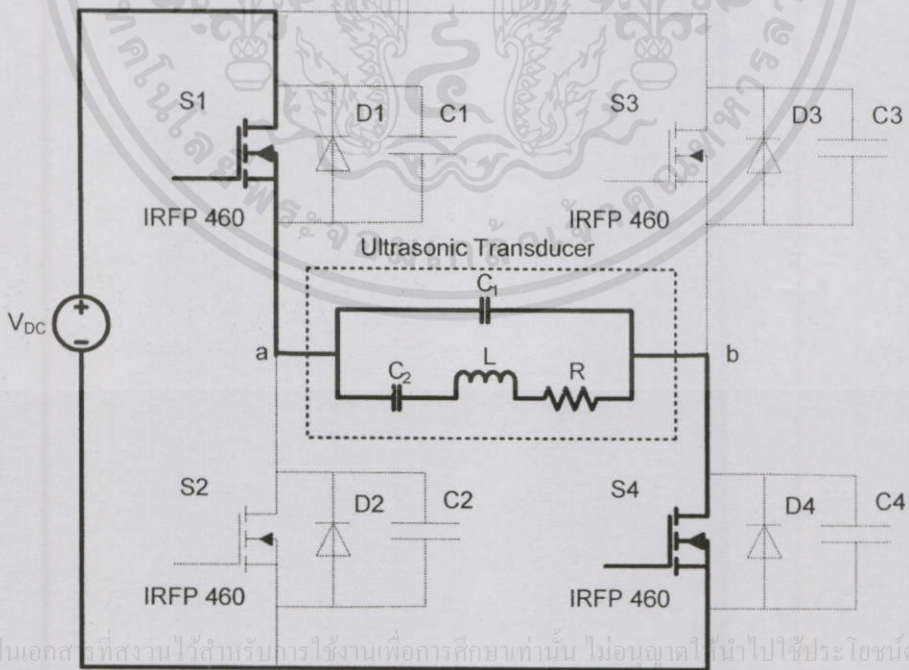
2.6 พีดับบลิวเอ็มแบบเลื่อนเฟสฟลูบริดจ์คอนเวอร์เตอร์

พีดับบลิวเอ็มแบบเลื่อนเฟสฟลูบริดจ์คอนเวอร์เตอร์ (Phase - Shift PWM Full - Bridge Converter) ดังในรูปที่ 2.19 มีโครงสร้างเหมือนกับ พีดับบลิวเอ็มฟลูบริดจ์คอนเวอร์เตอร์ ดังรูปที่ 2.17 แต่มีลักษณะขั้วขาเกของอุปกรณ์สวิตช์แตกต่างกัน โดยปกติสัญญาณ PWM จะทำงานเป็นคู่ๆ ให้ S1 และ S4 นำกระแส โดยที่ S2 และ S3 ยังคงหยุดนำกระแสอยู่ สลับกับการให้ S1 และ S4 หยุดนำกระแส แล้วก็ค่อย ให้ S2 และ S3 นำกระแส ทำอย่างนี้ไปเรื่อยๆ แต่ PWM แบบเลื่อนเฟส ช่วงเวลา S1 และ S4 นำกระแส S1 จะนำกระแส ก่อน S4 เป็นเวลา Φ และช่วงเวลาที่หยุดนำกระแส ก็จะช้ากว่า S1 เพราะช่วงนำกระแส ช้ากว่านั่นเอง ส่วน S2 และ S3 ก็เหมือนกัน ช่วงนำกระแส S3 จะช้ากว่า S2 เป็นเวลา Φ และช่วงหยุดนำกระแส ก็จะช้ากว่า S2 ด้วย จากรูปสัญญาณ V_{G1} แสดงถึงคาบเวลาการทำงานของ S1, V_{G2} แสดงถึงคาบเวลาการทำงานของ S2, V_{G3} แสดงถึงคาบเวลาการทำงานของ S3, V_{G4} แสดงถึงคาบเวลาการทำงานของ S4, V_{ab} คือแรงดันที่โหลด ที่จะนำไปใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

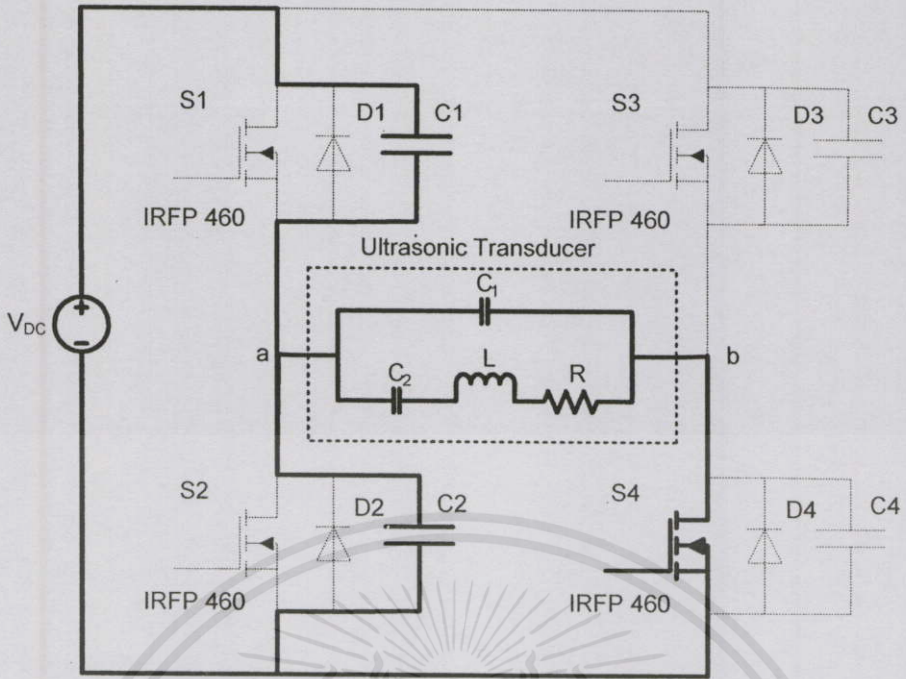


รูปที่ 2.19 การทำงานของสัญญาณ PWM แบบเลื่อนเฟส

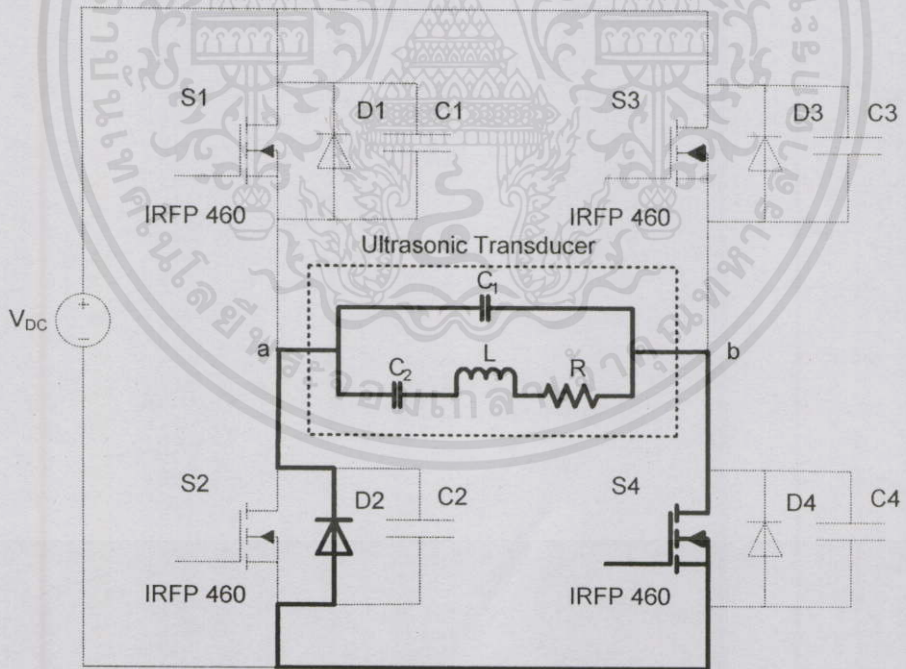


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาของเอกสารอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถานะที่ 1

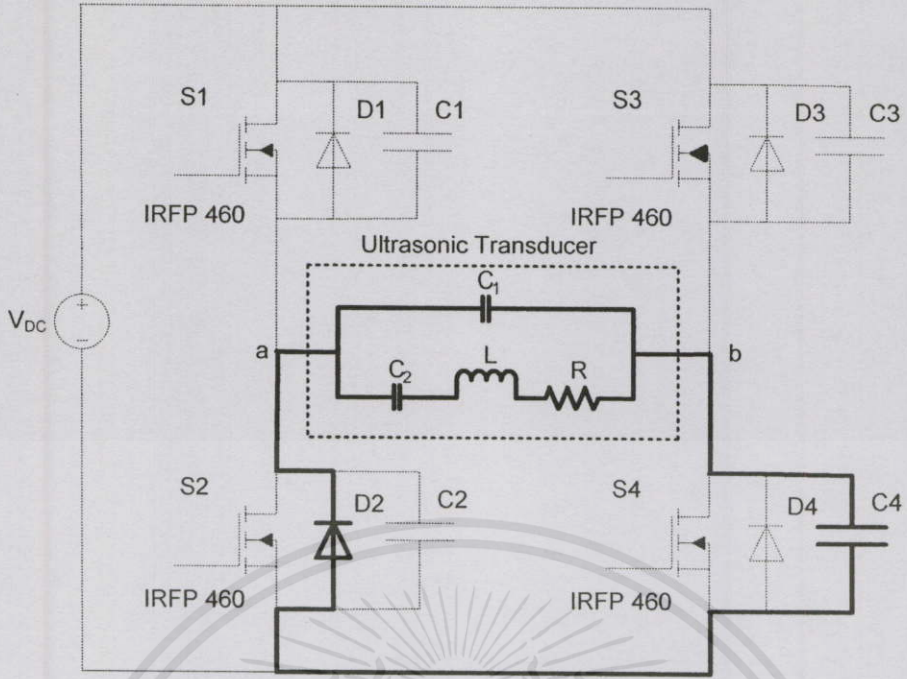


สภาวะที่ 2

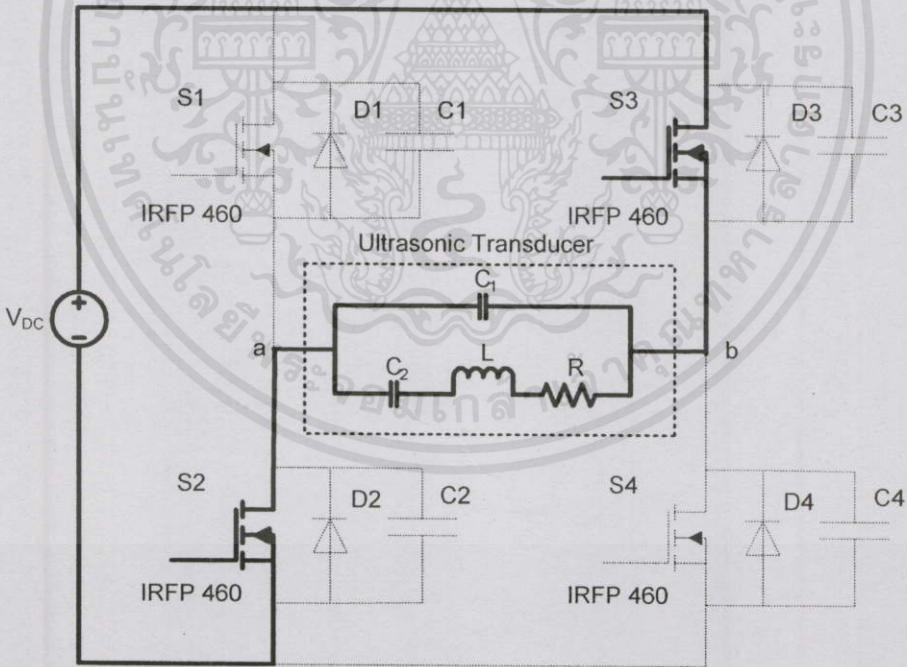


สภาวะที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

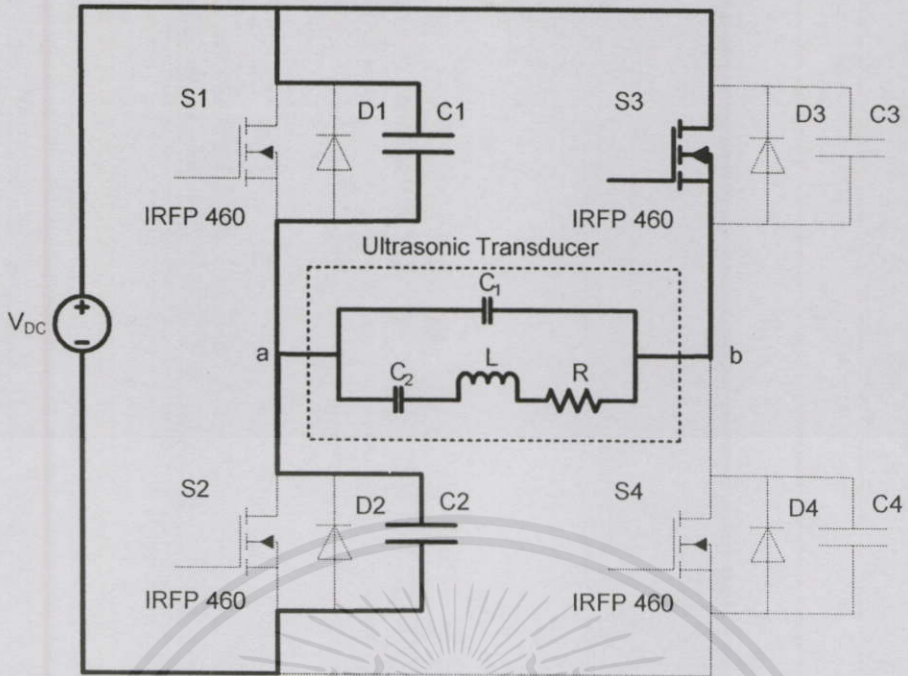


สถานะที่ 4

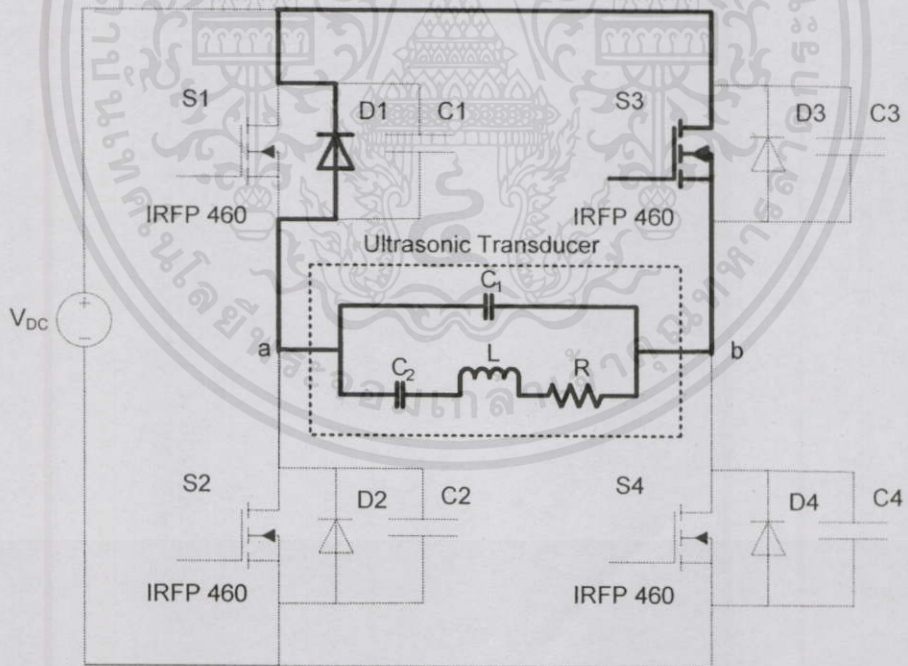


สถานะที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

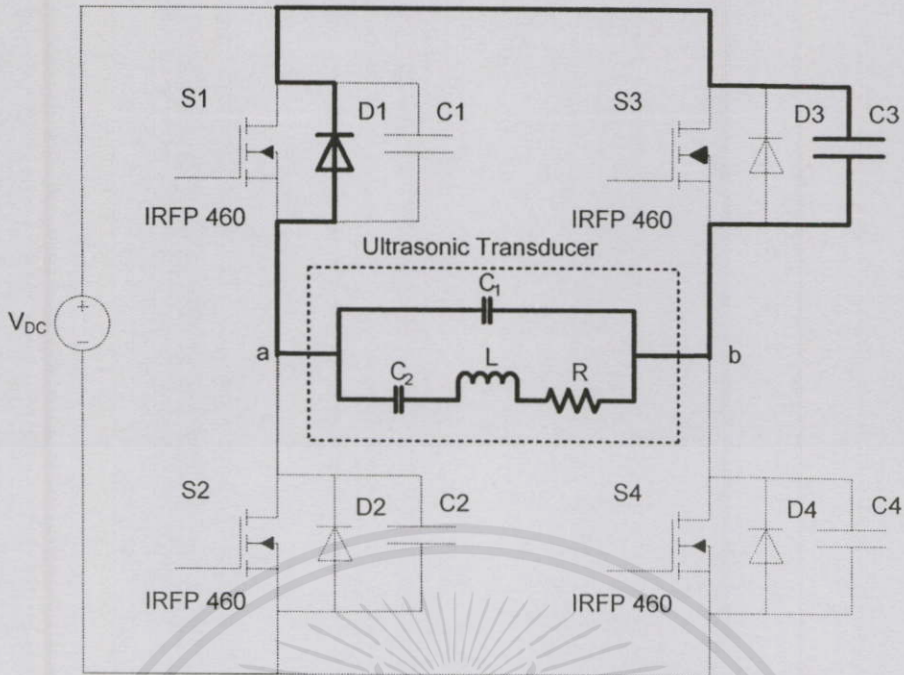


สถานะที่ 6



สถานะที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



สภาวะที่ 8

รูปที่ 2.20 การทำงานของ PWM แบบเลื่อนเฟสที่สภาวะ 1 ถึง 8

การทำงานของ PWM แบบเลื่อนเฟสแบ่งได้เป็น 8 สภาวะโดยการกำหนดการทำงานจากสัญญาณขั้วเกทของอุปกรณ์สวิตช์ ดังแสดงในรูปที่ 2.19 รูปสัญญาณแรงดันเอาต์พุตของคอนเวอร์เตอร์ที่จุด ab มีลักษณะคล้ายสัญญาณสี่เหลี่ยม โดยที่จุดที่แรงดันเป็นศูนย์ เกิดจากการเลื่อนเฟสขั้วสัญญาณเกท ของคู่อุปกรณ์สวิตช์ วงจรของแหล่งจ่ายกำลังไม่ครบรอบ ทำให้แรงดันไม่สามารถปรากฏที่เอาต์พุต สัญญาณแรงดันที่ได้จึงเป็นศูนย์ วงจรแบบเลื่อนเฟสที่สภาวะการทำงานทั้ง 8 ดังรูปที่ 2.20

สภาวะที่ 1 สวิตช์ S_1 และ S_4 turn on กระแสจากแหล่งจ่ายกระแสตรงไหลผ่าน S_1 ไปสู่โหลด และไหลผ่าน S_4 กลับสู่แหล่งจ่ายครบวงจร แรงดันตกคร่อมโหลดเท่ากับแรงดันที่แหล่งจ่ายกระแสตรง

สภาวะที่ 2 สวิตช์ S_1 turn off ในขณะที่ S_4 ยังคง turn on ตามรูปแบบของสัญญาณขั้วเกทที่จ่ายให้ S_4 กระแสไหลต่อเนื่องที่ผ่าน S_1 จะเปลี่ยนมาไหลผ่าน D_2 แทน

สภาวะที่ 3 ช่วงเวลาฟรีวีล (Free Wheeling) ในสภาวะนี้แหล่งจ่ายไม่สามารถจ่ายกระแสให้ โหลดได้เพราะ S_1 เปิดวงจรช่วงเวลานี้พลังงานที่สะสมในโหลด ไหลผ่าน S_4 ไปสู่ D_2 และไหลกลับสู่แหล่งจ่าย ครบวงจร ช่วงเวลาฟรีวีลนี้ไดโอด D_2 ได้รับฟอร์เวิร์ดไบอัสทำให้ครบวงจรคืนพลังงาน เพื่อลดการเสียหายของอุปกรณ์สวิตช์จากการได้รับรีเวิร์สไบอัสที่สูงเกินขีดจำกัดได้ และแรงดันตกคร่อมโหลดมีค่าเท่ากับศูนย์ เนื่องจากสวิตช์ S_2 ได้ปิดวงจร D_2 ได้รับฟอร์เวิร์ดไบอัส

สภาวะที่ 4 กระแสที่ไหลผ่าน S4 เปลี่ยนมาไหลผ่าน D4 กลับสู่โหลดจากการที่ S4 เปิด วงจรพลังงานที่สะสมในโหลด ลดลงเกือบหมด และ S2 turn on โดยที่ S4 turn off

สภาวะที่ 5 S2 และ S3 turn on กระแสจากแหล่งจ่ายกระแสตรงไหลผ่าน S3 ไปสู่โหลด เมื่อกระแสไหลผ่านโหลด จะเกิดการสะสมพลังงาน จากนั้นไหลผ่าน S2 กลับสู่แหล่งจ่ายครบ วงจร แรงดันตกคร่อมโหลด เท่ากับแรงดันที่แหล่งจ่ายแต่มีค่าติดลบเพราะกลับขั้ว กับสภาวะที่ 1

สภาวะที่ 6 S2 turn off ในขณะที่ S3 ยังได้รับสัญญาณแรงดันขั้วเกทและ turn on อยู่ใน กรณีที่เป็น โหลดสะสมพลังงานกระแสะจะไม่หยุดไหลอย่างทันทีทันใด แต่จะค่อยๆลดลง ดังนั้น กระแสไหลต่อเนื่องที่ผ่าน S2 จะเปลี่ยนมาไหลผ่าน C2 แทน S2 ที่เปิดวงจร กระแสไหลผ่าน S3 ไปสู่โหลด ครบวงจร

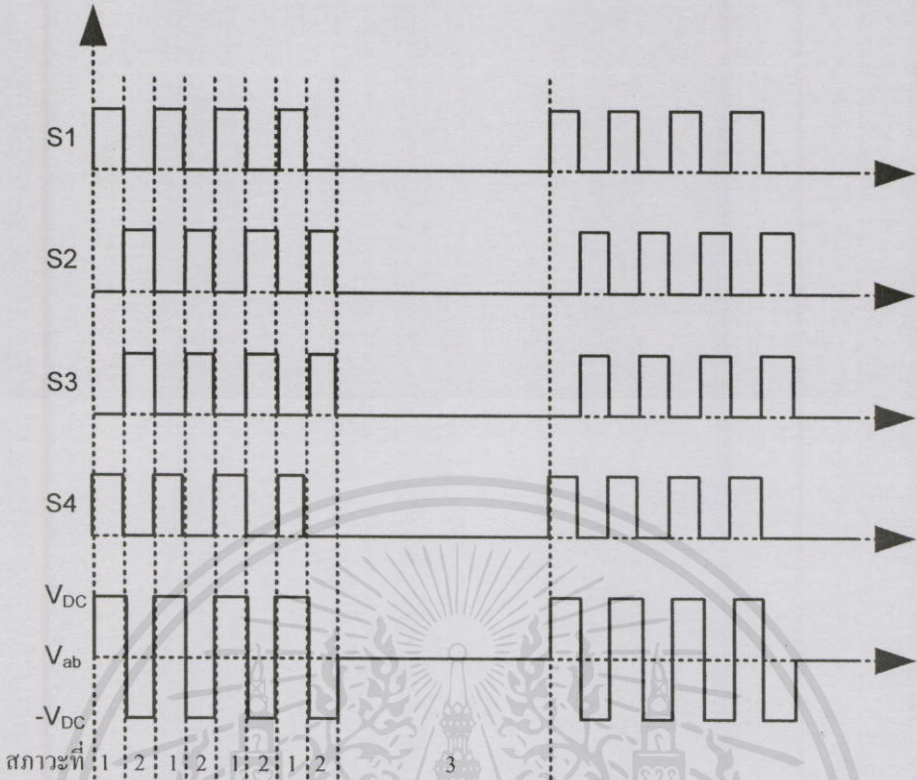
สภาวะที่ 7 ช่วงฟรีวิล เนื่องจาก S2 เปิดวงจร ตัดทางเดินกระแสจากแหล่งจ่ายไฟฟ้า กระแสตรงออก กระแสที่ไหลผ่านโหลด ทำให้พลังงานที่สะสมอยู่ถูกจ่ายออกมาเป็น กระแสไฟฟ้าไหลผ่าน D1 ไปสู่ S3 ครบวงจร ช่วงเวลาฟรีวิลนี้ D1 นำกระแส เนื่องจากศักย์ไฟฟ้า ที่ขาซอส เริ่มสูงกว่าขาเดรน ในกรณีนี้แรงดันตกคร่อมสวิตช์จะถูกกำจัดให้เท่าแรงดันเริ่มต้นที่ตก คร่อมไดโอดลักษณะที่เกิดขึ้นแบบนี้ ช่วยลดการเสียหายของอุปกรณ์สวิตช์จากการได้รับรีเวิร์ส ไบอัสที่สูงเกินขีดจำกัด และเนื่องจาก S3 ปิดวงจรและ D1 ได้รับฟอร์เวิร์ดไบอัส ทำให้แรงดันตก คร่อมโหลดมีค่าเท่ากับศูนย์

สภาวะที่ 8 S1 turn on และ S3 turn off กระแสไหลผ่าน S3 เปลี่ยนมาไหลผ่าน D3 กลับสู่ โหลด และ S1 หลังจากสภาวะนี้แล้ว คอนเวอร์เตอร์จะกลับไปเริ่มการทำงานที่สภาวะที่ 1 อีกครั้ง เป็นวงรอบการทำงานต่อไป

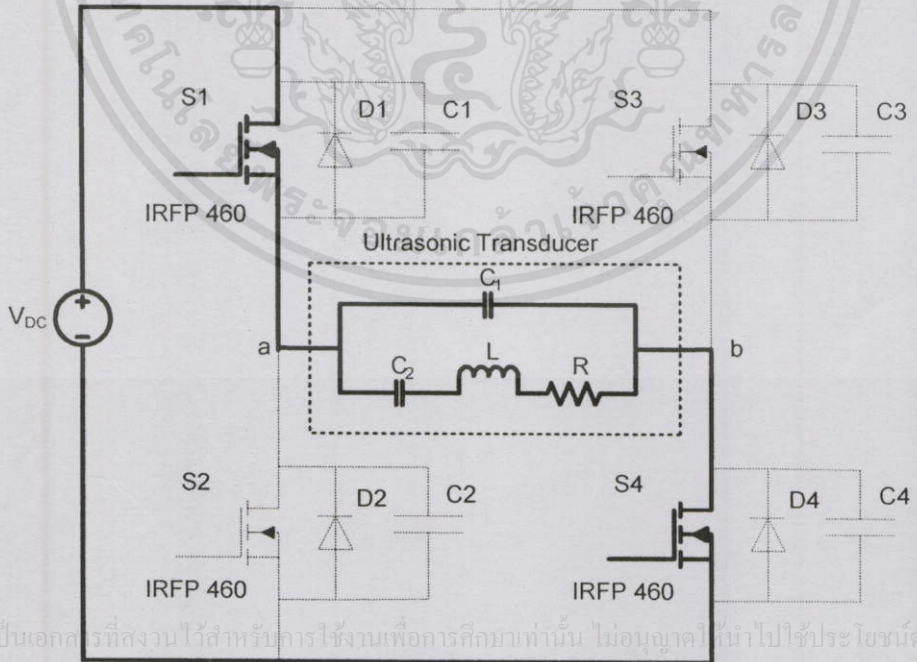
2.7 พีดีเอ็มฟลูบริดจ์คอนเวอร์เตอร์

พีดีเอ็มฟลูบริดจ์คอนเวอร์เตอร์ (Pulse Density Modulation Full Bridge Converter) มี โครงสร้างเหมือนกับ พีดีบีทีเอ็มฟลูบริดจ์คอนเวอร์เตอร์ ดังรูปที่ 2.17 แต่มีลักษณะขั้วขาเกท ของอุปกรณ์สวิตช์แตกต่างกัน

จากรูปที่ 2.21 แสดงถึงการทำงานของสัญญาณ PDM และรูปที่ 2.22 แสดงถึงแรงดันและ กระแสที่ออกมาจากวงจรภาคกำลังที่จ่ายให้โหลด โดยที่มอสเฟทจะทำงานเป็นคู่ๆ S1 กับ S4 turn on สลับกับ S2 กับ S3 ทำอย่างนี้ไปเรื่อยๆ 4 คาบเวลา และหยุดจ่ายแรงดัน 4 คาบเวลาเช่นกัน โดย ช่วงหยุด S1, S2, S3 และ S4 turn off เพื่อที่จะทำให้แรงดันมีค่าเป็นศูนย์ โดยจะจ่ายแรงดันเป็น สัญญาณและหยุด เป็นอย่างนี้ไปเรื่อยๆ

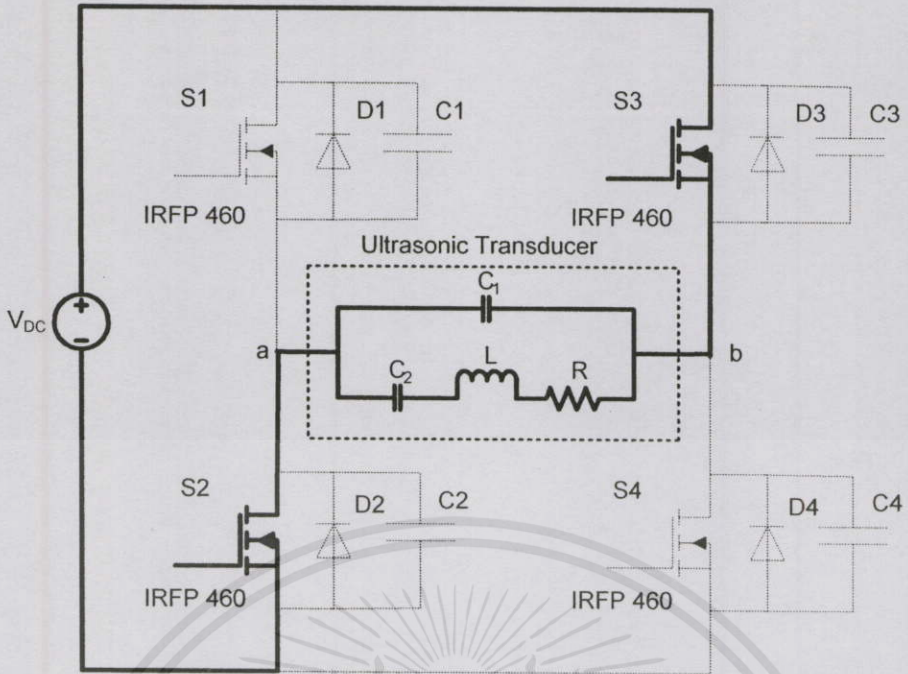


รูปที่ 2.21 การทำงานของสัญญาณ PDM

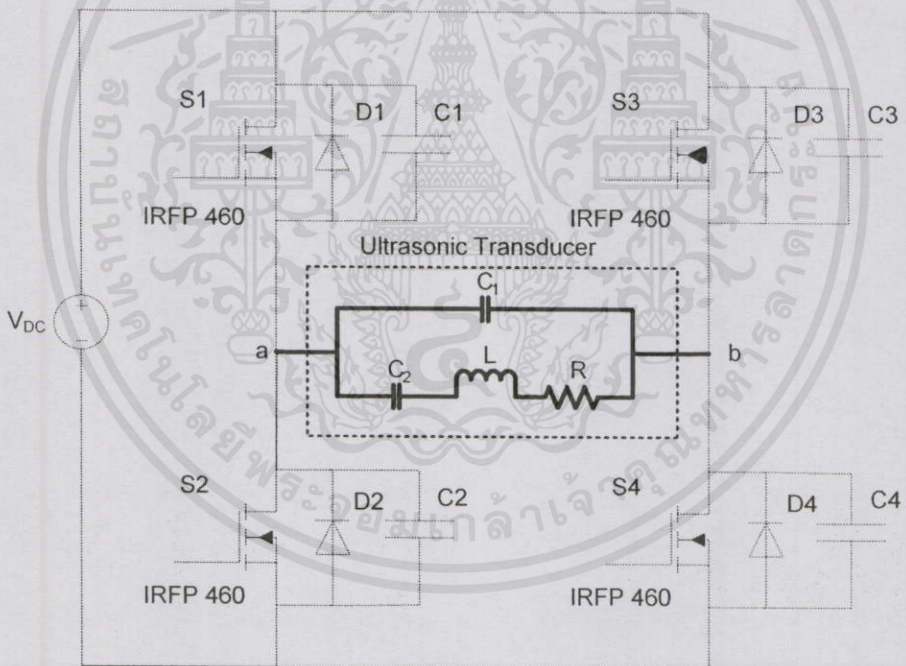


สถานะที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



สภาวะที่ 2



สภาวะที่ 3

รูปที่ 2.22 การทำงานของสัญญาณ PDM ที่สภาวะที่ 1 ถึง 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 สภาวะที่ 1 สวิตช์ S1 และ S4 turn on กระแสจากแหล่งจ่ายกระแสตรงไหลผ่าน S1 ไปสู่
 โหลด และไหลผ่าน S4 กลับสู่แหล่งจ่ายกระแสตรง แรงดันตกคร่อมโหลดเท่ากับแรงดันที่
 แหล่งจ่ายกระแสตรง

สภาวะที่ 2 S2 และ S3 turn on กระแสจากแหล่งจ่ายกระแสตรงไหลผ่าน S3 ไปสู่โหลด เมื่อกระแสไหลผ่านโหลด จะเกิดการสะสมพลังงาน จากนั้นไหลผ่าน S2 กลับสู่แหล่งจ่ายครบวงจร แรงดันตกคร่อมโหลด เท่ากับแรงดันที่แหล่งจ่ายแต่มีค่าติดลบเพราะกลับขั้ว กับสภาวะที่ 1

สภาวะที่ 3 คือช่วงเวลาที่รูปสัญญาณที่โหลดเป็นศูนย์ S1, S2, S3 และ S4 turn off ทำให้ไม่มีกระแสจากแหล่งจ่ายไหลไปสู่อุปกรณ์

2.8 ฮาร์โมนิก (Harmonic)

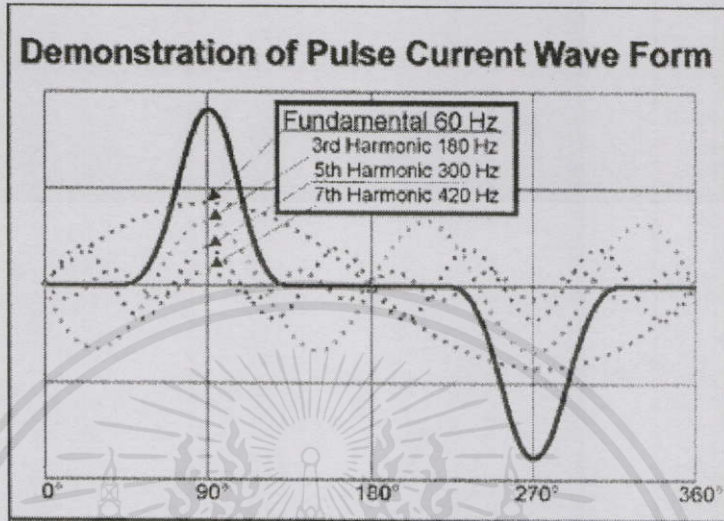
ในปัจจุบันการไฟฟ้าหรือผู้ใช้ไฟฟ้าได้ให้ความสำคัญกับคุณภาพไฟฟ้ามากขึ้น เนื่องจากในระบบไฟฟ้าและโดยเฉพาะในกระบวนการผลิตของอุตสาหกรรมนั้น ได้มีการใช้อุปกรณ์ที่มีเทคโนโลยีสูงกว่าเดิมในอดีต ซึ่งคุณลักษณะการทำงานของอุปกรณ์ดังกล่าวจะไวต่อการเปลี่ยนแปลงต่อกระแสและแรงดัน คือถ้ามีขนาดและรูปร่างผิดเพี้ยนไปจากสภาพการจ่ายไฟปกติ อาจจะทำให้อุปกรณ์มีการทำงานผิดพลาดหรือเกิดการชำรุดเสียหายขึ้นได้ ซึ่งเป็นปัญหาคุณภาพไฟฟ้าที่ต้องมีการป้องกันและแก้ไข โดยสาเหตุหลักที่ทำให้กระแสและแรงดันในระบบไฟฟ้ามีขนาดและรูปร่างผิดเพี้ยนไปจากสภาพการจ่ายไฟปกติ มีสาเหตุเกิดจากฮาร์โมนิกที่มีอยู่ในระบบไฟฟ้า ซึ่งเนื่องจากปัจจุบันโรงงานอุตสาหกรรมและอาคารพาณิชย์ มีการใช้อุปกรณ์สมัยใหม่เทคโนโลยีสูงที่ทำจากอุปกรณ์ทางด้าน โซลิติสเตด เพื่อเพิ่มประสิทธิภาพในการควบคุมกระบวนการผลิตให้มีคุณภาพและได้ปริมาณตามที่ต้องการและในอนาคตจะมีแนวโน้มการใช้มากขึ้นเรื่อยๆ โดยส่วนใหญ่เป็นอุปกรณ์ที่มีการทำงานแบบไม่เป็นเชิงเส้น (Non-linear load) ซึ่งเป็นแหล่งกำเนิดฮาร์โมนิก เช่นคอนเวอร์เตอร์ (Convertor) ตัวเรียงกระแสกำลัง (Power Rectifier) และชุดขับเคลื่อนปรับความเร็ว (Adjustable - Speed Drive) เป็นต้น

ด้วยผลของการใช้อุปกรณ์ที่มีการทำงานแบบไม่เป็นเชิงเส้น อุปกรณ์ดังกล่าวจะจ่ายกระแสฮาร์โมนิกเข้าสู่ระบบไฟฟ้าภายในของผู้ใช้ไฟเอง หรือถ้าเป็นอุปกรณ์ที่มีพิกัดขนาดใหญ่ กระแสฮาร์โมนิกนั้นอาจไหลเข้าสู่ระบบไฟฟ้าอื่นในบริเวณข้างเคียง จากผลกระทบของฮาร์โมนิกส์ทำให้กระแสและแรงดันในระบบมีขนาดและรูปร่างเพี้ยน (Distortion) ไปจากสภาพการจ่ายไฟปกติ ซึ่งเป็นผลทำให้อุปกรณ์มีการทำงานผิดพลาดหรือเกิดการชำรุดเสียหายได้

ฮาร์โมนิก [6] (Harmonic) คือส่วนประกอบในรูปสัญญาณคลื่นไซน์ (Sine wave) ของสัญญาณหรือปริมาณเป็นคาบใดๆซึ่งมีความถี่เป็นจำนวนเต็มเท่าของความถี่หลักมูล

(Fundamental Frequency ในระบบไฟฟ้าบ้านเรามีค่าเท่ากับ 50 Hz) เช่นฮาร์โมนิกลำดับที่ 3 มีค่าความถี่เป็น 150Hz และฮาร์โมนิกลำดับที่ 5 มีค่าความถี่เป็น 250Hz เป็นต้น

และผลของฮาร์โมนิกเมื่อรวมกันกับสัญญาณความถี่หลักมูลด้วยทางขนาด (Amplitude) และมุมเฟส (Phase Angle) ทำให้สัญญาณที่เกิดขึ้นมีขนาดเปลี่ยนไปและมีรูปสัญญาณเพี้ยนไปจากสัญญาณคลื่นไซน์ดังรูปที่ 2.23



รูปที่ 2.23 แสดงถึงฮาร์โมนิกส์ลำดับที่ 3, 5 และ 7 ที่มีมุมต่างๆ ทำให้สัญญาณไซน์มีรูปร่างผิดเพี้ยน [6]

ในทางคณิตศาสตร์สามารถใช้ออนุกรมฟูเรียร์อธิบายคุณลักษณะของฮาร์โมนิกส์ได้ โดยสัญญาณหรือฟังก์ชัน ที่เป็นคาบใดๆ สามารถกระจายให้อยู่ในรูปผลรวมของฟังก์ชันตรีโกณมิติที่มีความถี่ต่างๆเป็นฟังก์ชันคาบที่เขียนแทนด้วย $f(t)$ ดังสมการที่ 2.1

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega_0 t) + \sum_{n=1}^{\infty} b_n \sin(n\omega_0 t) \quad (2.1)$$

$$a_0 = \frac{1}{T} \int f(t) dt$$

$$a_n = \frac{2}{T} \int f(t) \cos n\omega_0 t dt$$

$$b_n = \frac{2}{T} \int f(t) \sin n\omega_0 t dt$$

เมื่อ T คือ คาบของสัญญาณและ n คือเลขจำนวนเต็มบวก ในกรณีที่ $n = 0$ จะเป็นการเป็นเอกสารเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น เมื่อผู้ดูเห็นว่ามีอะไรที่ผิดปกติในกราฟความถี่มูลฐาน (Fundamental Frequency) หรือกรณีที่ n มีค่ามากกว่าศูนย์เราเรียกความถี่นี้ว่าฮาร์โมนิกส์ลำดับที่ n ซึ่งเป็นได้ทั้งลำดับคู่และคี่

ค่าความเพี้ยนฮาร์โมนิกกรรมมาตรฐาน IEC และ IEEE ใช้ค่าความเพี้ยนฮาร์โมนิกส์ %THD (Total Harmonic Distortion) เป็นค่าบอกระดับความเพี้ยนฮาร์โมนิก โดยเทียบจาก

อัตราส่วนระหว่างค่ารากที่สองของผลบวกกำลังสองของส่วนประกอบฮาร์มอนิก กับค่าของส่วนประกอบความถี่หลักมูลเทียบเป็นร้อยละ ซึ่งจะแยกออกเป็น ค่าความเพี้ยนกระแสฮาร์มอนิกรวม และค่าความเพี้ยนแรงดันฮาร์มอนิกรวม

ค่าความเพี้ยนกระแสฮาร์มอนิกรวม (Total Harmonic Current Distortion : THD_I)

$$\%THD_I = \frac{\sqrt{\sum_{h=2}^{\infty} I_h^2 (rms)}}{I_1 (rms)} \times 100\% \quad (2.2)$$

ค่าความเพี้ยนแรงดันฮาร์มอนิกรวม (Total Harmonic Voltage Distortion : THD_V)

$$\%THD_V = \frac{\sqrt{\sum_{h=2}^{\infty} V_h^2 (rms)}}{V_1 (rms)} \times 100\% \quad (2.3)$$

$V_h (rms)$: ค่า rms ของแรงดันฮาร์มอนิกลำดับที่ h

$I_h (rms)$: ค่า rms ของกระแสฮาร์มอนิกลำดับที่ h

$V_1 (rms)$: ค่า rms ของแรงดันที่ความถี่หลักมูล

$I_1 (rms)$: ค่า rms ของกระแสที่ความถี่หลักมูล

ผลกระทบของฮาร์มอนิกที่มีผลต่ออุปกรณ์ในระบบไฟฟ้า

ปัญหาฮาร์มอนิกที่ทำให้เกิดผลกระทบต่ออุปกรณ์ในระบบไฟฟ้าออกเป็น 2 กรณีคือ

- ทำให้อุปกรณ์ในระบบมีการทำงานผิดพลาดด้วยผลของค่าแรงดัน และกระแสฮาร์มอนิกที่มีขนาดและรูปคลื่นสัญญาณไซน์ผิดเพี้ยนไป

- ทำให้อุปกรณ์ในระบบมีอายุการใช้งานน้อยลงหรือเกิดการชำรุดเสียหาย เนื่องจากมีค่า rms ของแรงดันหรือกระแส สูงขึ้นที่เกิดจากค่าฮาร์มอนิก หรือมีการขยายของแรงดันและกระแสฮาร์มอนิก ที่เกิดจากฮาร์มอนิกรีโซแนนซ์

ปัญหาฮาร์มอนิกที่ส่งผลกระทบต่ออุปกรณ์ในระบบไฟฟ้านี้คือ

ผลของฮาร์มอนิกรีโซแนนซ์เกิดขึ้นในกรณีที่ความถี่เรโซแนนซ์ของระบบไปตรงกับความถี่ฮาร์มอนิกทำให้เกิดการขยายขนาดของแรงดันและกระแสฮาร์มอนิก เป็นผลทำให้อุปกรณ์ได้รับความเสียหายเนื่องจากได้รับกระแส และแรงดันเกินพิกัด

ผลของกระแสฮาร์มอนิกที่ไหลอยู่ในระบบจำหน่ายและสายส่ง ทำให้เกิดค่ากำลังสูญเสียในสายมากขึ้น ทำให้ประสิทธิภาพ การส่งจ่ายลดลง เนื่องจากกระแสฮาร์มอนิกทำให้ค่า rms ของกระแสและความต้านทานของสายสูงขึ้น

ผลของกระแสฮาร์มอนิก Triplen (ลำดับที่3,6,9..) จัดอยู่ในกลุ่มที่มีลำดับเป็นศูนย์ (Zero Sequence) ในระบบ 3 เฟส 4 สาย ฮาร์มอนิกกลุ่มนี้จะรวมกันกันไหลอยู่ในสายนิวตรอล อาจทำให้สายนิวตรอนหรือหม้อแปลงเสียหายได้หากไม่มีการออกแบบรองรับไว้

ผลของกระแสฮาร์มอนิกทำให้กำลังสูญเสียขณะมีโหลดและกำลังสูญเสียสเตรย์ฟลักซ์ (Stray Flux Loss) ของหม้อแปลงมีค่าเพิ่มขึ้น และทำให้ประสิทธิภาพการในรับ โหลดของหม้อแปลงลดลงไป (derating) ผลของแรงดันฮาร์มอนิกทำให้เกิดกำลังสูญเสียกระแสไหลวน (Eddy Current Loss) และกำลังสูญเสียฮิสเทอรีซิส (Hysteresis - Loss) เพิ่มขึ้น

ผลของกระแสฮาร์มอนิกทำให้เกิดความร้อนและความเครียดไดอิเล็กตริก (Dielectric Stress) กับตัวคาปาซิเตอร์ และอาจทำให้พีวส์ของตัวคาปาซิเตอร์ขาดง่ายกว่าการใช้งานปกติ ผลของแรงดันฮาร์มอนิกทำให้เกิดค่ากำลังสูญเสียในคาปาซิเตอร์ และผลจากภาวะเรโซแนนซ์ที่ตัวคาปาซิเตอร์ทำให้เกิดขยายกระแสและแรงดันฮาร์มอนิกขนาดใหญ่ ดังนั้นเพื่อความปลอดภัยในการใช้งานของคาปาซิเตอร์สามารถทนต่อค่ากระแสและแรงดันฮาร์มอนิก คาปาซิเตอร์ที่ออกแบบสร้างจากผู้ผลิตได้กำหนดตามมาตรฐาน IEEE Std. 18-1992

ผลของกระแสฮาร์มอนิกทำให้เกิดความร้อนในตัวพีวส์เพิ่มขึ้น ทำให้ลักษณะเวลากระแส (Time-Current Characteristic) ของพีวส์เปลี่ยนไปกรณีที่มีฟอลต์ระดับต่ำเกิดขึ้นพีวส์จะขาดก่อนในเวลาที่กำหนด หรือในกรณีที่พีวส์ขาดโดยไม่ทราบสาเหตุจะเป็นเหตุมาจากฮาร์มอนิกในกรณีที่เกิดภาวะเรโซแนนซ์ได้เช่นกัน

ผลของฮาร์มอนิกทำให้การทำงานของรีเลย์ผิดพลาดซึ่งขึ้นอยู่กับคุณสมบัติการทำงาน ของชนิดรีเลย์ การทำงานของรีเลย์ชนิด Electromagnetic ขึ้นอยู่กับค่ากระแสและแรงดัน rms ส่วนการทำงานของรีเลย์ชนิด Digital ขึ้นอยู่กับค่าแรงดันยอดคลื่น (Crest Voltage) จากการ Sampling และตรวจค่า Zero Crossing ค่ากระแสหรือแรงดันที่ศูนย์ โดยลักษณะที่ทำให้รีเลย์ทำงานผิดพลาดดังนี้

- ทำให้รีเลย์มีการทำงานช้าลง หรือทำงานด้วยค่า (Pickup Values) ที่สูง โดยปกติรีเลย์จะทำงานอย่างรวดเร็วและทำงานด้วยค่าเริ่มต่ำๆ

- กรณีที่มีกระแสฮาร์มอนิก Triplen มากพออาจทำให้กราวด์รีเลย์ทำงานผิดพลาด (False Trip)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

- ทำให้รีเลย์ระยะทาง (Distance Relay) ทำงานผิดพลาด ด้วยผลของกระแสฮาร์มอนิกที่ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีผลต่อเบสแรงเหน้า และต่อองอิงอิงของเอกสารทุกครั้งที่มีการนำไปใช้ ทำให้อิมพีแดนซ์เพิ่มขึ้นต่างจากค่าอิมพีแดนซ์ที่ทำการเซตตั้งที่ความถี่หลักมูล

- ทำให้รีเลย์สถิตแบบความถี่ต่ำ (Static Underfrequency Relay) มีความไวกว่าปกติ อาจทำให้เกิดการทริปผิดพลาด

- ทำให้รีเลย์กระแสและแรงดันเกิน (Overcurrent and Overvoltage Relay) ทำงานผิดพลาดตามคุณสมบัติที่ตั้งไว้

- ทำให้ความเร็วในการทำงานของรีเลย์ชนิดผลต่าง (Differential Relay) ทำงานช้าลง ผลของกระแสฮาร์มอนิกมีผลกระทบต่อความสามารถในการตัดกระแส (Current Interruption Capacity) ของอุปกรณ์สวิตช์เกียร์ คือทำให้ขนาดของอัตราค่ากระแสเทียบกับเวลา di / dt มีค่าสูงในขณะที่กระแสมีค่าเป็นศูนย์ เป็นผลทำให้เซอร์กิตเบรกเกอร์ไม่สามารถตัดกระแสได้เมื่อมีฮาร์มอนิก ซึ่งปัญหานี้จะเกิดกับอุปกรณ์อื่นๆที่ใช้ตัดกระแสได้เช่นกัน

ผลของฮาร์มอนิกทำให้มิเตอร์วัดค่าไฟฟ้า (Watt - Hour Meter) ซึ่งเป็นมิเตอร์ประเภทงานเหนี่ยวนำ (Induction Disk) ทำการวัดค่าผิดพลาดได้ ซึ่งโดยปกติการปรับแต่งมิเตอร์นั้นจะทำการปรับแต่งที่ความถี่หลักมูล

ผลของฮาร์มอนิกต่อเครื่องจักรไฟฟ้า ทำให้กำลังสูญเสียเพิ่มขึ้น เป็นผลทำให้เครื่องจักรร้อนกว่าปกติ ทำให้มอเตอร์เหนี่ยวนำ สามเฟสเกิดปรากฏการณ์ค็อกกิ้ง (Cogging) คือไม่สามารถสตาร์ทมอเตอร์ได้ จากการที่ความเร็วมอเตอร์ต่ำกว่าความเร็วซิงโครนัส และทำให้เกิดการออสซิลเลตทางกลของเครื่องจักรไฟฟ้า ซึ่งมีผลต่อประสิทธิภาพและแรงบิดของเครื่องจักร

ผลของฮาร์มอนิกทำให้เกิดสัญญาณรบกวน (Noise) ในระบบสื่อสารเช่นในระบบโทรศัพท์

ฮาร์มอนิกที่อยู่ในระบบไฟฟ้าเป็นปัญหาคุณภาพไฟฟ้าสำคัญเรื่องหนึ่ง เพราะปัจจุบันการใช้โหลดประเภทที่ไม่เป็นเชิงเส้นที่เป็นแหล่งจ่ายฮาร์มอนิก และโรงงานอุตสาหกรรมและในอาคารพาณิชย์นับวันมีการใช้โหลดดังกล่าวเพิ่มมากขึ้น ผลทำให้รูปคลื่นของแรงดันและกระแสเพี้ยนไปจากรูปไซน์ ซึ่งจะส่งผลกระทบต่ออุปกรณ์มีการทำงานผิดพลาด หรืออาจเกิดการเสียหายได้

2.9 สรุปและข้อแนะนำ

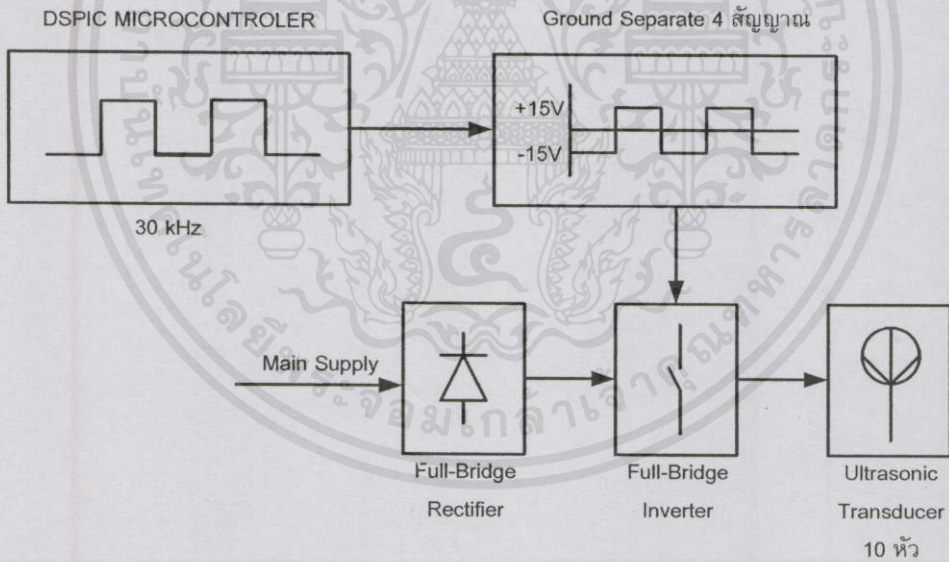
จากที่ได้กล่าวมาข้างต้น เป็นทฤษฎีของการทำความสะอาดด้วยคลื่นอัลตราโซนิค อุปกรณ์ที่สำคัญที่ทำให้เกิดคลื่นอัลตราโซนิคเพื่อนำมาใช้ในการทำความสะอาด วงจรและรูปแบบการสร้างสัญญาณที่นำมาใช้ในการทดลองในวิทยานิพนธ์นี้ โดยในการสร้างสัญญาณทั้ง 3 รูปแบบนั้นโครงสร้างของวงจรมันเหมือนกัน เพียงแต่สร้างสัญญาณที่มาขั้วเกท เพื่อให้ได้รูปสัญญาณที่ไหลตามต้องการเท่านั้น โดยสัญญาณที่สร้างมาจาก ไมโครโปรเซสเซอร์ dsPIC ซึ่งจะกล่าวถึงในบทต่อไป โดยแต่ละรูปสัญญาณเพียงแค่เปลี่ยนตัวโปรแกรมเท่านั้น ที่การทดลองนี้เลือกใช้ไมโครโปรเซสเซอร์ เพราะว่าง่ายต่อการปรับเปลี่ยนรูปสัญญาณทั้ง 3 รูปทำให้ไม่ว่าจะเลือกใช้วงจรไหนในการทำความสะอาดก็ไม่เกี่ยวข้องกับการเพิ่มค่าใช้จ่าย

บทที่ 3

วงจรควบคุมและสร้างสัญญาณ

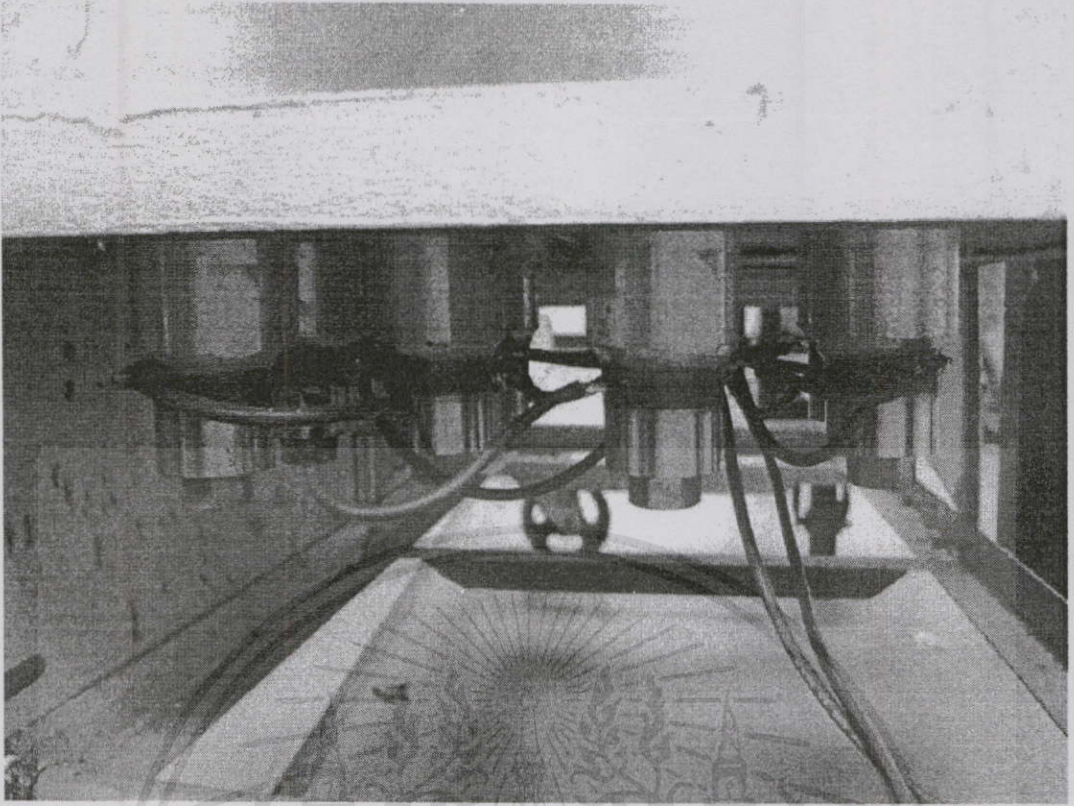
3.1 บทนำ

ในบทนี้จะกล่าวถึงหลักการการทำงานของเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิก การสร้างสัญญาณ PWM, PS-PWM และ PDM โดยไมโครโปรเซสเซอร์ dsPIC 30F2010 ดังรูปที่ 3.1 โดยการสร้างสัญญาณที่ความถี่เรโซแนนซ์ของหัวอัลตราโซนิกทรานสดิวเซอร์ที่ 30 kHz เพื่อส่งต่อให้วงจรอินเวอร์เตอร์แบบฟลูบริดจ์เพื่อสร้างสัญญาณให้หัวอัลตราโซนิกทรานสดิวเซอร์ต่อไปเพื่อทำให้เกิดกระบวนการทำความสะอาดที่ภายในแท่งคั้บรรจุของเหลวที่ใช้ทำความสะอาด ซึ่งได้กล่าวมาแล้วในบทที่ 2 ถึงกระบวนการทำความสะอาดที่เกิดขึ้น และการสร้างคลื่นความถี่ย่านอัลตราโซนิกโดยหัวอัลตราโซนิกทรานสดิวเซอร์



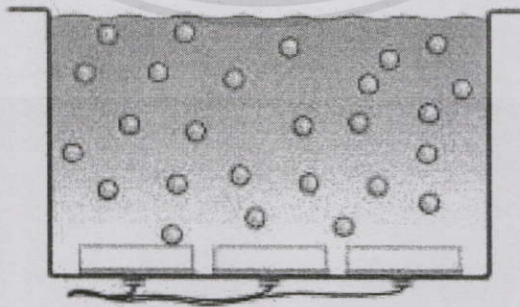
รูปที่ 3.1 หลักการทำงานของเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิก

หัวอัลตราโซนิกที่ใช้ในการทดลองมีรายละเอียดดังนี้ ความถี่เรโซแนนซ์ที่ใช้คือ 30 kHz 70W จำนวนที่ใช้ 10 หัว ต่อขานกันกระจายอยู่ทั่วพื้นที่ใต้ห้องแท้งค์ (tank) ที่ใส่ของเหลว ดังรูปที่ 3.2 ไม่ว่าจะฉีดน้ำทั้งต้น อีกทั้งห้ามมิให้คลื่นเปล่งเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 หัวอัลตราโซนิกทรานสดิวเซอร์ที่อยู่ใต้ท้องแทงค์ (tank) ที่บรรจุของเหลวที่ใช้ทำความสะอาด

รูปที่ 3.3 แสดงถึงรูปของกระบวนการทำความสะอาดด้วยคลื่นอัลตราโซนิก เมื่อเกิดกระบวนการควิเตชัน และอิมโพชัน ที่ตัวกลางของเหลวในแทงค์ จะเกิดฟองอากาศควิเตชันเพื่อทำความสะอาดอุปกรณ์ ตามที่ได้กล่าวมาแล้วเรื่อง ทฤษฎีการทำความสะอาดด้วยคลื่นอัลตราโซนิก

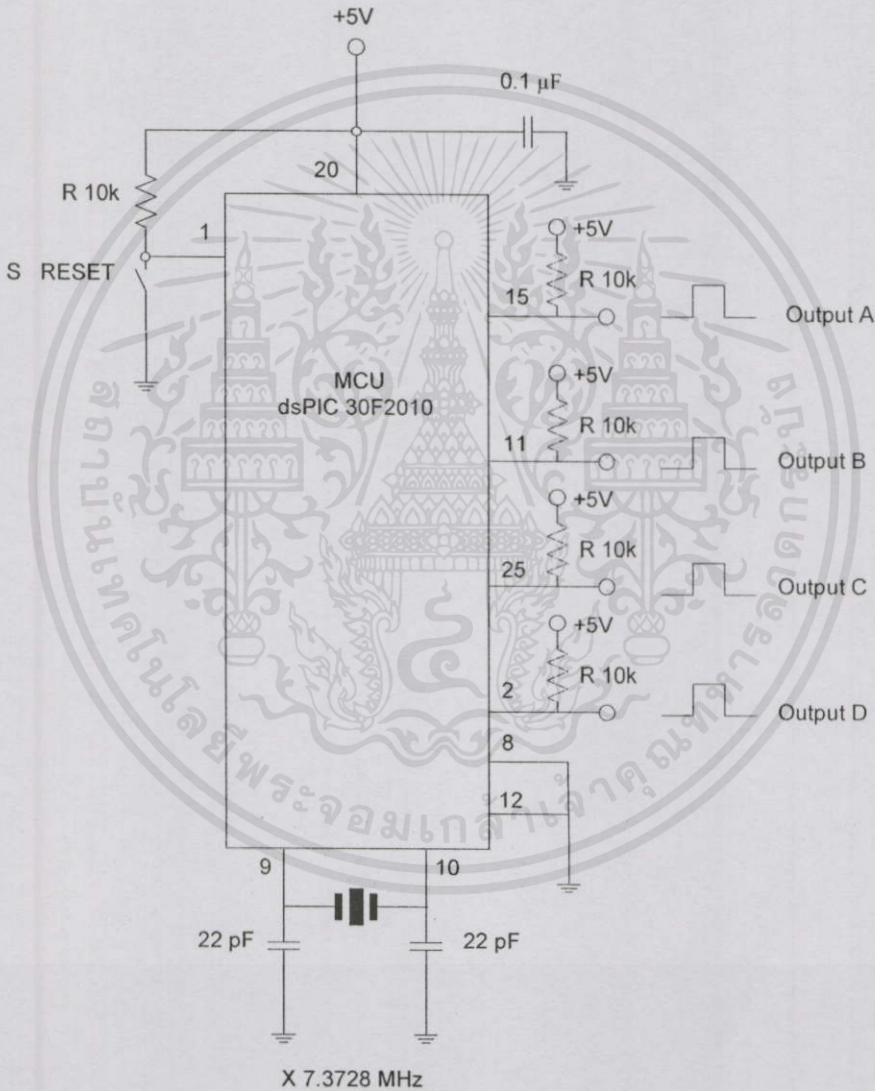


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น

รูปที่ 3.3 คลื่นเสียงอัลตราโซนิกเมื่อผ่านตัวกลางของเหลว และเกิดการนำไปใช้

กระบวนการทำความสะอาด

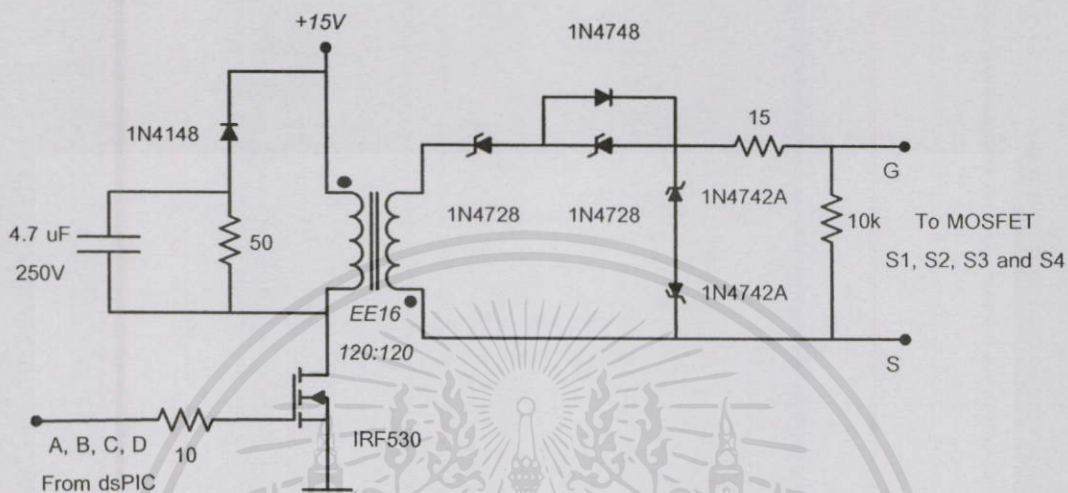
วงจรควบคุมที่ใช้สร้างสัญญาณ PWM, PS-PWM และ PDM นั้นได้มาจากไมโครโปรเซสเซอร์ dsPIC 30F2010 [7] ดังรูปที่ 3.4 โดยการเขียนโปรแกรมควบคุมด้วยภาษา C ผ่านโปรแกรม MPLAB C30 ซึ่งได้ถูกเขียนให้มีเอาต์พุต ทั้งหมด 4 เอาต์พุต A, B, C และ D ออกมาจากขาที่ 15, 11, 25 และ 2 ตามลำดับโดยสร้างสัญญาณที่มีความถี่เรโซแนนท์ของ อัลตราโซนิคทรานสดิวเซอร์ ที่ 30 kHz เพื่อส่งต่อไปให้วงจรขับเคลื่อนของเพาเวอร์มอสเฟต ทั้ง 4 ชุด โดยวิทยานิพนธ์นี้เลือกใช้ dsPIC ในการสร้างสัญญาณเพราะว่าสะดวกแก่การเปลี่ยนรูปสัญญาณให้กับชุดขับโหลด (หัวอัลตราโซนิค) เพราะไม่ต้องสร้างชุดสัญญาณ ทั้ง 3 แบบ



รูปที่ 3.4 รูปวงจรไมโครโปรเซสเซอร์ dsPIC 30F2010

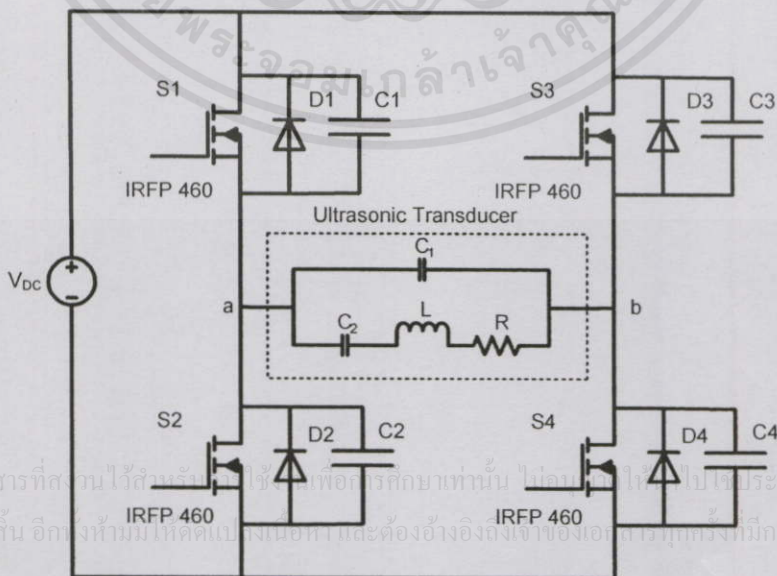
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ในส่วนของวงจรขับเคลื่อนของเพาเวอร์มอสเฟตนั้นจะใช้วงจรขับด้วยหม้อแปลงความถี่สูง โดยจะต่อมอสเฟตให้ทำหน้าที่เป็นสวิตช์ เพื่อช่วยในการขยายสัญญาณให้มีค่ามากขึ้น โดยผ่านหม้อแปลงที่มีอัตราส่วน 1:1 จากนั้นจะต่อขานานด้วยซีเนอร์ไดโอดเพื่อรักษาระดับแรงดัน

เอาท์พุทให้ลงที่ โดยค่าความต้านทานที่ต่อขนานด้านทุติยภูมิไว้เพื่อแบ่งกระแสที่ไหลผ่านซีเนอร์ไดโอด ดังรูปที่ 3.5 เพื่อส่งต่อไปยังวงจรภาคกำลัง โดยมีทั้งหมด 4 ชุด โดยรับสัญญาณจาก dsPIC จากรูปที่ 3.4 Output A, B, C และ D ส่งให้วงจรภาคกำลัง รูปที่ 3.6 โดยส่งให้ MOSFET S1, S2, S3 และ S4 ตามลำดับ



รูปที่ 3.5 วงจรขับเคลื่อนของเพาเวอร์มอสเฟต

วงจรภาคกำลังนี้เป็นวงจรที่ใช้อินเวอร์เตอร์แบบฟลูบริดจ์ ซึ่งใช้มอสเฟตทั้งหมด 4 ตัว ดังรูปที่ 3.6 โดยจะนำกระแส และหยุดการนำกระแส ตามจังหวะสัญญาณที่ถูกส่งมาจาก วงจรขับเคลื่อน ในรูปที่ 3.5

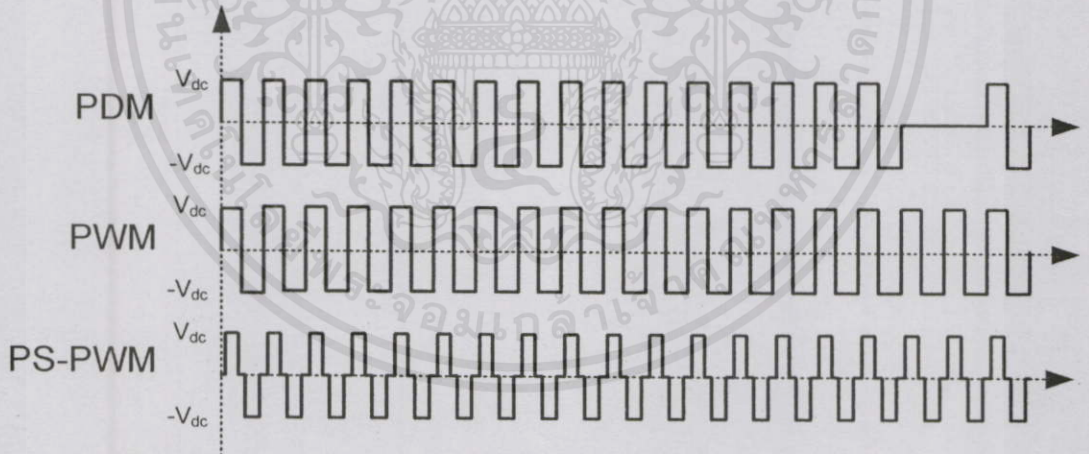


รูปที่ 3.6 วงจรภาคกำลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้ส่วนหนึ่ง ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มอสเฟตทั้ง 4 ตัวจะทำงานแยกอิสระจากกัน โดยรับสัญญาณ PWM จากวงจรขับเคลื่อนด้วยความถี่ 30 kHz การทำงานของสัญญาณที่ออกมาจากวงจรขับเคลื่อนโดยปกติสัญญาณ PWM จะทำงานเป็นคู่ๆ ให้ S1 และ S4 นำกระแสโดยที่ S2 และ S3 ยังคงหยุดนำกระแสอยู่ สลับกับการให้ S1 และ S4 หยุดนำกระแส แล้วก็ค่อย ให้ S2 และ S3 นำกระแส ทำอย่างนี้ไปเรื่อยๆ แต่ PS-PWM ช่วงเวลา S1 และ S4 นำกระแส S1 จะนำกระแส ก่อน S4 เป็นเวลา Φ และช่วงเวลาที่หยุดนำกระแส ก็จะช้ากว่า S1 เพราะช่วงนำกระแส ช้ากว่านั่นเอง ส่วน S2 และ S3 ก็เหมือนกัน ช่วงนำกระแส S3 จะช้ากว่า S2 เป็นเวลา Φ และช่วงหยุดนำกระแส ก็จะช้ากว่า S2 ด้วย จากรูปสัญญาณ V_{G1} แสดงถึงคาบเวลาการทำงานของ S1, V_{G2} แสดงถึงคาบเวลาการทำงานของ S2, V_{G3} แสดงถึงคาบเวลาการทำงานของ S3, V_{G4} แสดงถึงคาบเวลาการทำงานของ S4, V_{ab} คือแรงดันที่โหลด ที่จะนำไปใช้งาน

การทำงานของสัญญาณ PDM โดยที่มอสเฟตจะทำงานเป็นคู่ๆ S1 กับ S4 นำกระแส สลับกับ S2 กับ S3 ทำอย่างนี้ไปเรื่อยๆ โดยการทดลองนี้ ได้ใช้สัญญาณคาบเวลาทำงาน 16 คาบเวลา และหยุดจ่ายแรงดัน 2 คาบเวลาโดยช่วงหยุด S2 และ S4 นำกระแส เพื่อให้โหลดทั้ง 2 ด้านลงกราวด์เพื่อที่จะทำให้แรงดันมีค่าเป็นศูนย์ โดยจะจ่ายแรงดันเป็นสัญญาณและหยุด เป็นอย่างนี้ไปเรื่อยๆ

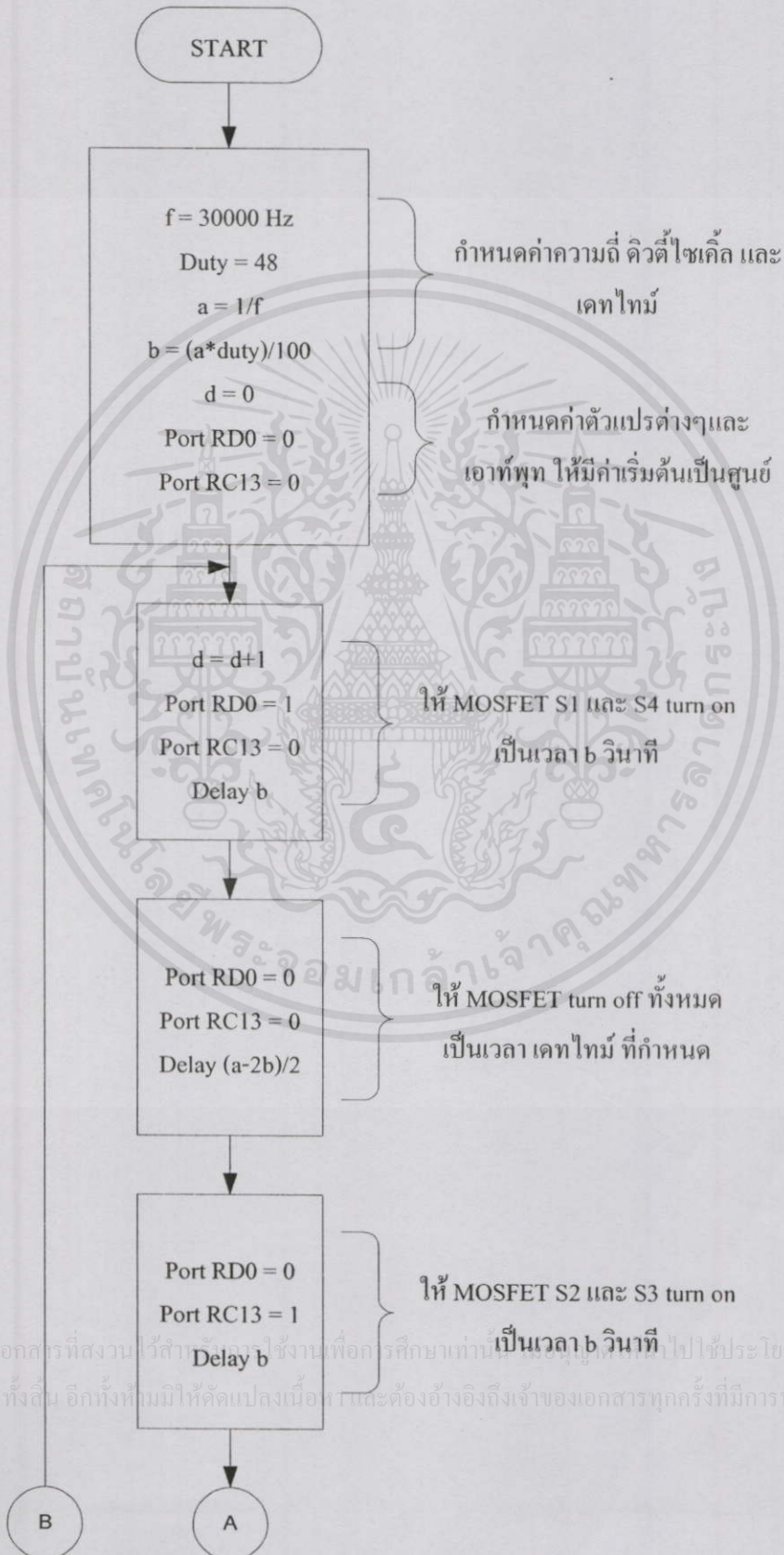


รูปที่ 3.7 รูปคลื่นแรงดันเอาต์พุตของสัญญาณควบคุมทั้ง 3 แบบคือ PDM, PWM และ PS-PWM ที่ใช้ในการขับโหลด (หัวอัลตราโซนิค)

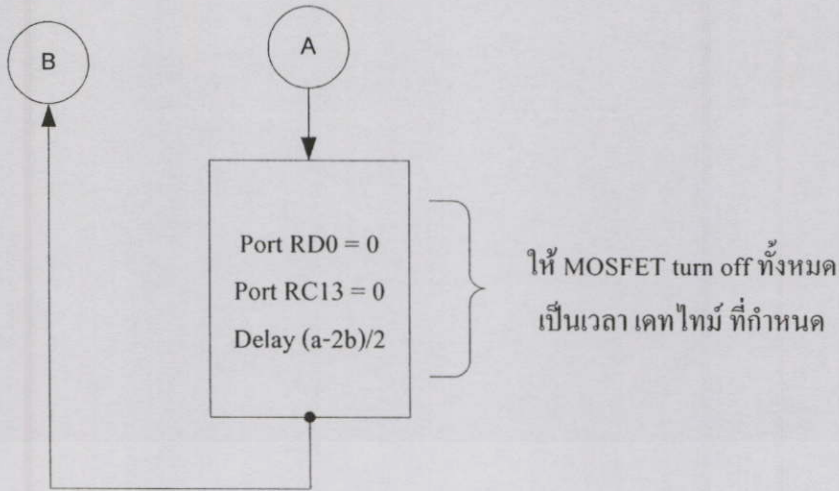
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
3.2 การสร้างสัญญาณ PWM
 ไม่ว่าจะถี่ใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อนี้จะกล่าวถึงการสร้างสัญญาณ PWM โดยได้กล่าวถึงรูปแบบสัญญาณ และสภาวะการทำงานไว้แล้วในบทที่ 2 โดยการสร้างสัญญาณนั้นได้มาจาก การเขียนโปรแกรมให้แก่ ตัว

dsPIC โดยหัวข้อนี้จะกล่าวถึงแต่ Flowchart และตัวโปรแกรม ดังรูปที่ 3.8 เนื่องจากตัวที่สร้างสัญญาณจริงๆคือ dsPIC เท่านั้นแต่ต่างกันตรงที่ตัวโปรแกรม ไม่ได้มีการปรับแต่งตัว dsPIC เพิ่มเติมเมื่อเปลี่ยนรูปสัญญาณ

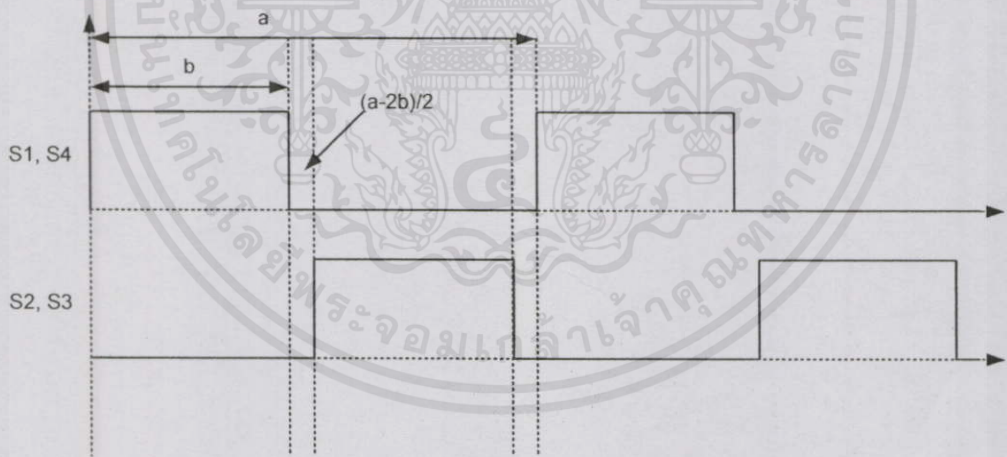


เอกสารนี้เป็นเอกสารที่สงวนไว้ว่าสามารถใช้งานเพื่อการศึกษาเท่านั้น ขออภัยหากนำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 Flowchart การสร้างรูปสัญญาณแบบ PWM

รูปที่ 3.8 โปรแกรมในการสร้างรูปสัญญาณ PWM เริ่มต้นจากการกำหนดค่าความถี่ที่ต้องการคือ 30 kHz และกำหนดค่าดีวตีไซเคิลไว้ที่ 48% แล้วจึงคำนวณค่าออกมาเป็นค่าคาบเวลาที่จะต้องทำงาน และหุคของรูปสัญญาณแต่ละมอสเฟตดังรูปที่ 3.9

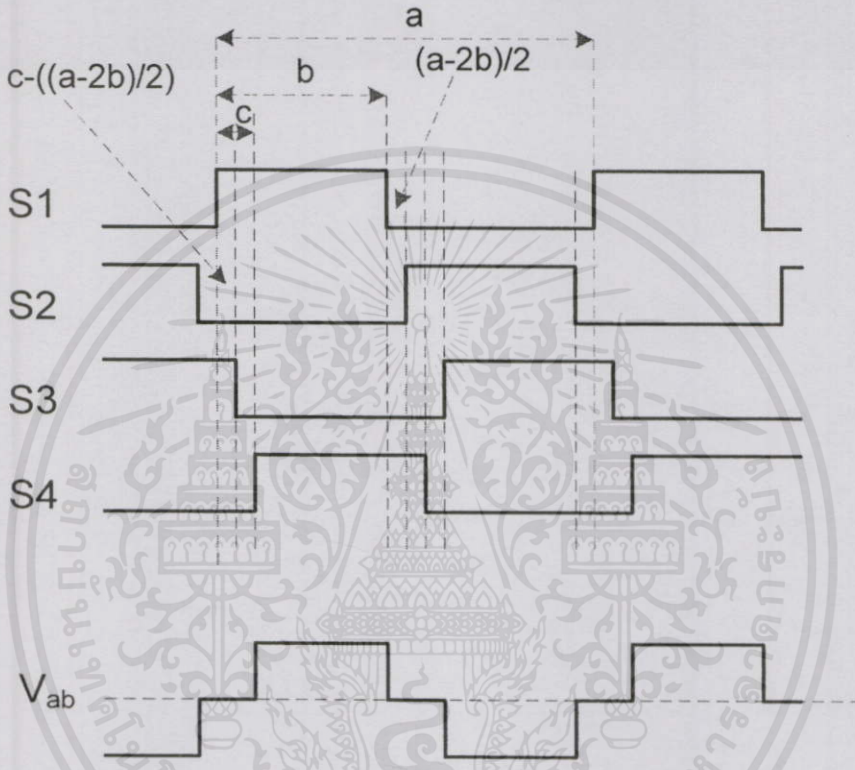


รูปที่ 3.9 คาบเวลาที่ใช้เขียนโปรแกรมรูปสัญญาณแบบ PWM

คำนวณได้ค่า $a = 33.33 \mu\text{s}$, $b = 15.99 \mu\text{s}$, $(a-2b)/2$ (คือช่วงเวลาเดทโทรม์) $= 0.675 \mu\text{s}$, แล้วกำหนดค่าเริ่มต้นต่างๆ ให้เป็นศูนย์ และมอสเฟตทั้งหมดค่าเริ่มต้นให้ turn off จากนั้น ให้ S1 และ S4 turn on เป็นเวลา $15.99 \mu\text{s}$ ช่วงเวลาเดทโทรม์ turn off มอสเฟตทั้งหมดเป็นเวลา $0.675 \mu\text{s}$ แล้ว S2 และ S3 ก็ turn on เป็นเวลา $15.99 \mu\text{s}$ ช่วงเวลาเดทโทรม์ turn off มอสเฟตทั้งหมดเป็นเวลา $0.675 \mu\text{s}$ ทำอย่างนี้ไปเรื่อยๆ

3.3 การสร้างสัญญาณ PS-PWM

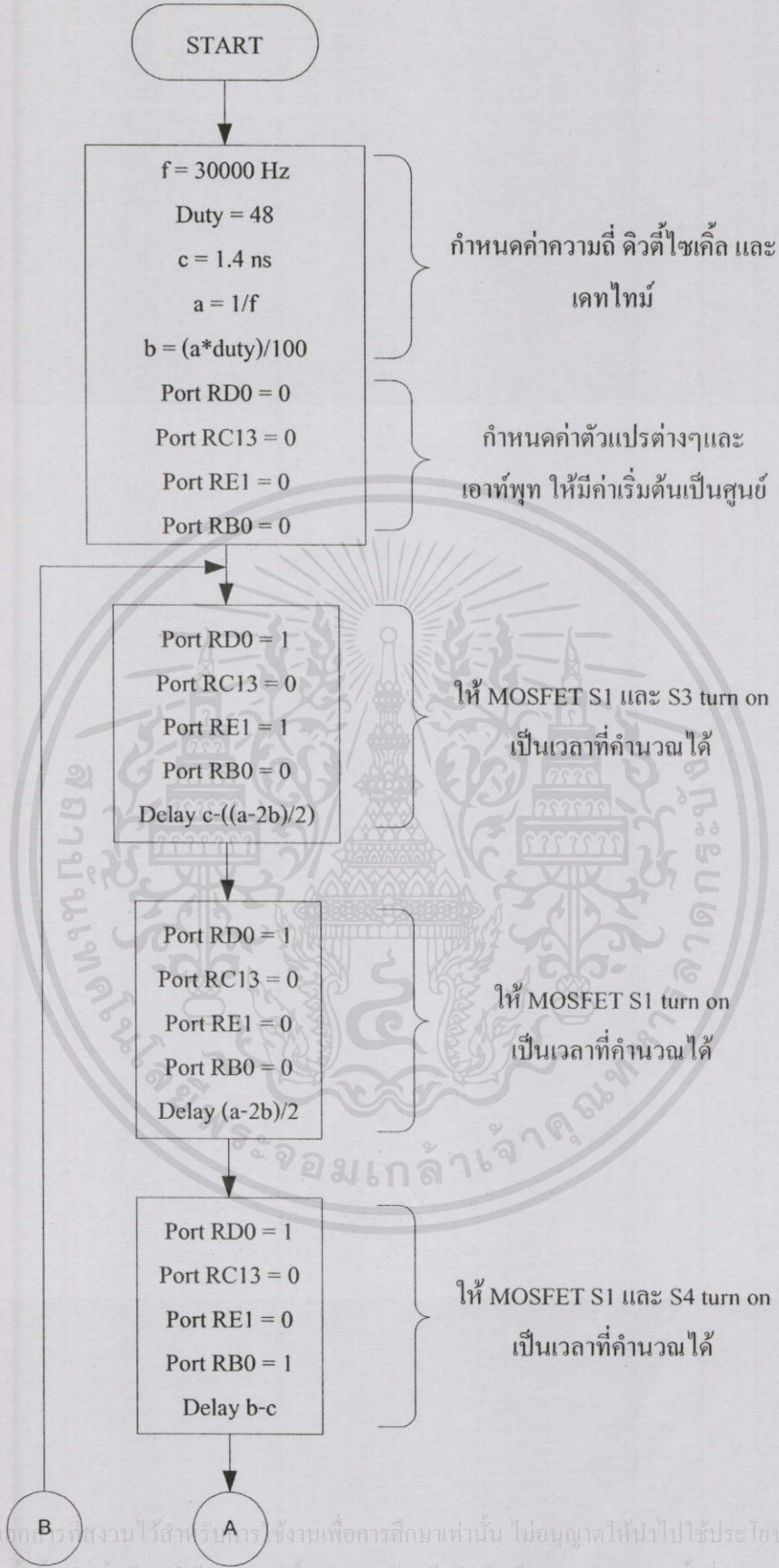
ในการสร้างสัญญาณ PS-PWM ซึ่งได้กล่าวถึงรูปแบบสัญญาณ และสภาวะการทำงานทั้งหมดไว้แล้วในบทที่ 2 ซึ่งได้แบ่งช่วงสัญญาณเป็น 8 ช่วง ดังในรูปที่ 3.10 เพื่อเขียนเป็นโปรแกรมในการสร้างสัญญาณดังรูปที่ 3.11



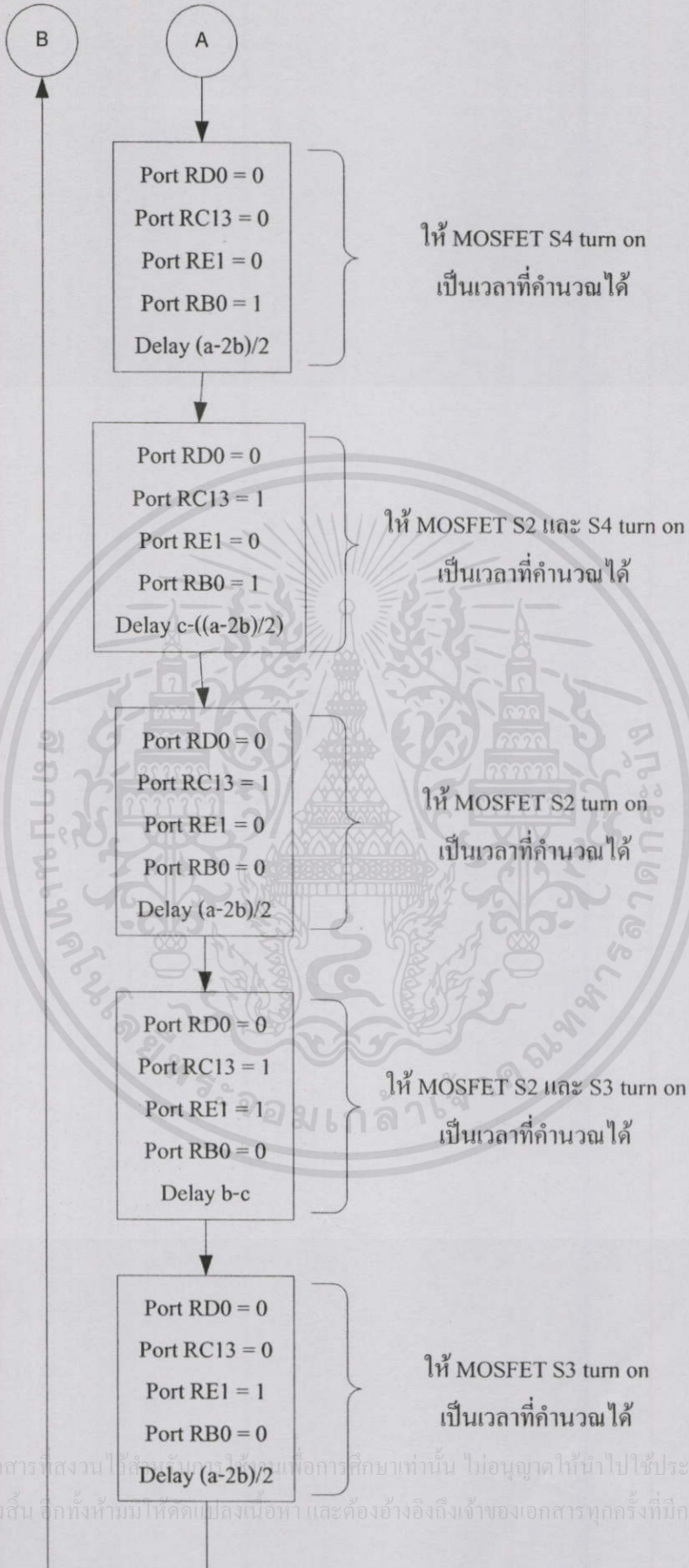
รูปที่ 3.10 ช่วงสัญญาณที่แบ่งเป็น 8 ช่วงเพื่อใช้ในการเขียนโปรแกรม

รูปที่ 3.11 โปรแกรมในการสร้างรูปสัญญาณ PWM เริ่มต้นจากการกำหนดค่าความถี่ที่ต้องการคือ 30 kHz และกำหนดค่าความถี่ไซเคิลไว้ที่ 48% แล้วจึงคำนวณค่าออกมาเป็นค่าคาบเวลาที่จะต้องทำงาน และหุคของรูปสัญญาณแต่ละมอสเฟตดังรูปที่ 3.10

คำนวณได้ค่า $a = 33.33 \mu\text{s}$, $b = 15.99 \mu\text{s}$, c (ค่าช่วงเวลาที่เลื่อนเฟส) $= 1.4 \text{ ns}$, $(a-2b)/2$ (คือช่วงเวลาเดทไทม์) $= 0.675 \mu\text{s}$, $c - ((a-2b)/2) = 0.673 \mu\text{s}$ แล้วกำหนดค่าเริ่มต้นต่างๆ ให้เป็นศูนย์ และมอสเฟตทั้งหมดค่าเริ่มต้นให้ turn off จากนั้น ให้ S1 และ S3 turn on เป็นเวลา $0.673 \mu\text{s}$ ช่วงเวลาเดทไทม์ S1 turn on เป็นเวลา $0.675 \mu\text{s}$ แล้ว S1 และ S4 ก็ turn on เป็นเวลา $15.98 \mu\text{s}$ ช่วงเวลาเดทไทม์ S4 turn on เป็นเวลา $0.675 \mu\text{s}$ แล้ว S2 และ S4 turn on เป็นเวลา $0.673 \mu\text{s}$ ช่วงเวลาเดทไทม์ S2 turn on เป็นเวลา $0.675 \mu\text{s}$ แล้ว S2 และ S3 ก็ turn on เป็นเวลา $15.98 \mu\text{s}$ ช่วงเวลาเดทไทม์ S3 turn on เป็นเวลา $0.675 \mu\text{s}$ ทำอย่างนี้ไปเรื่อยๆ



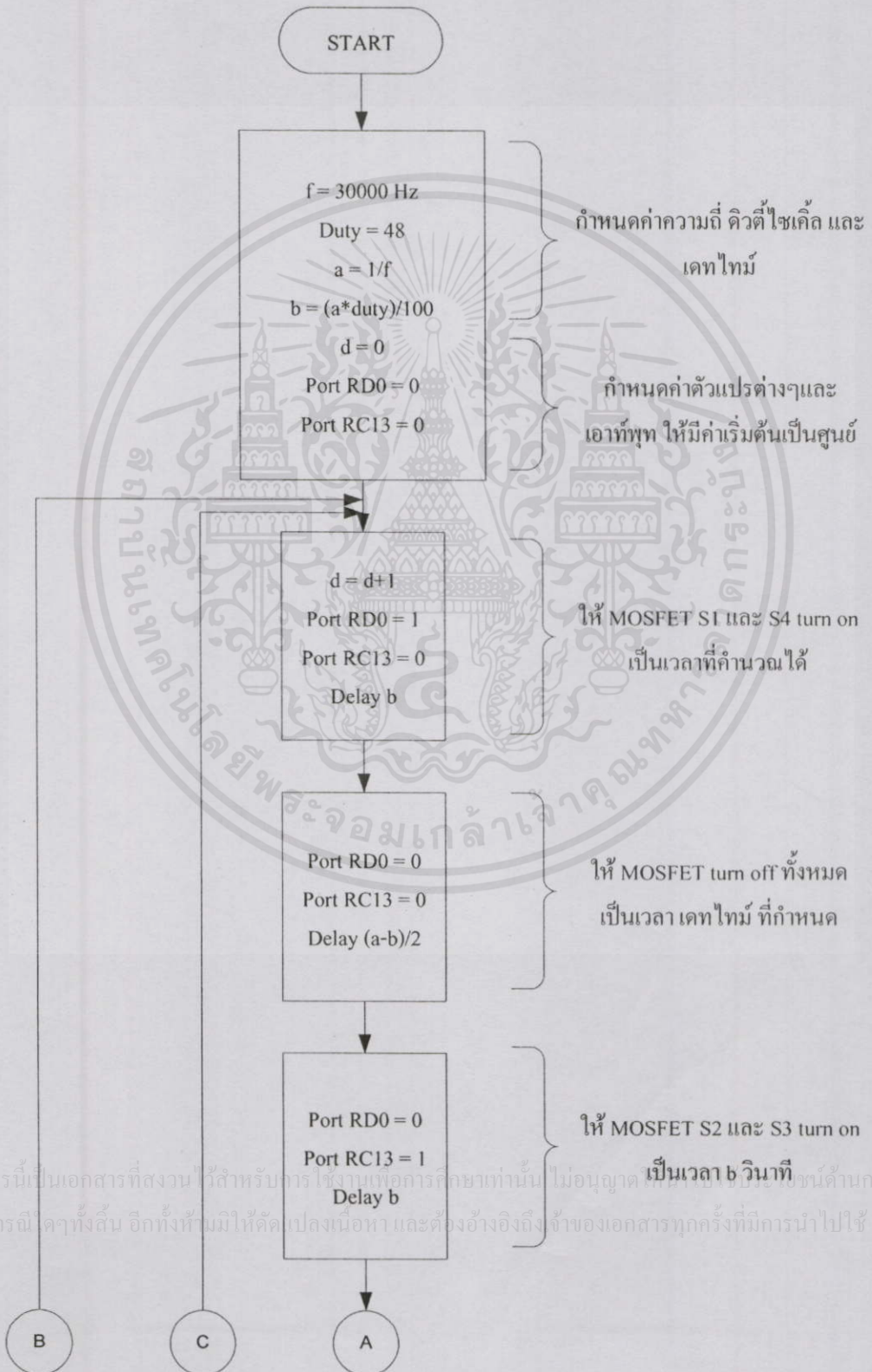
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



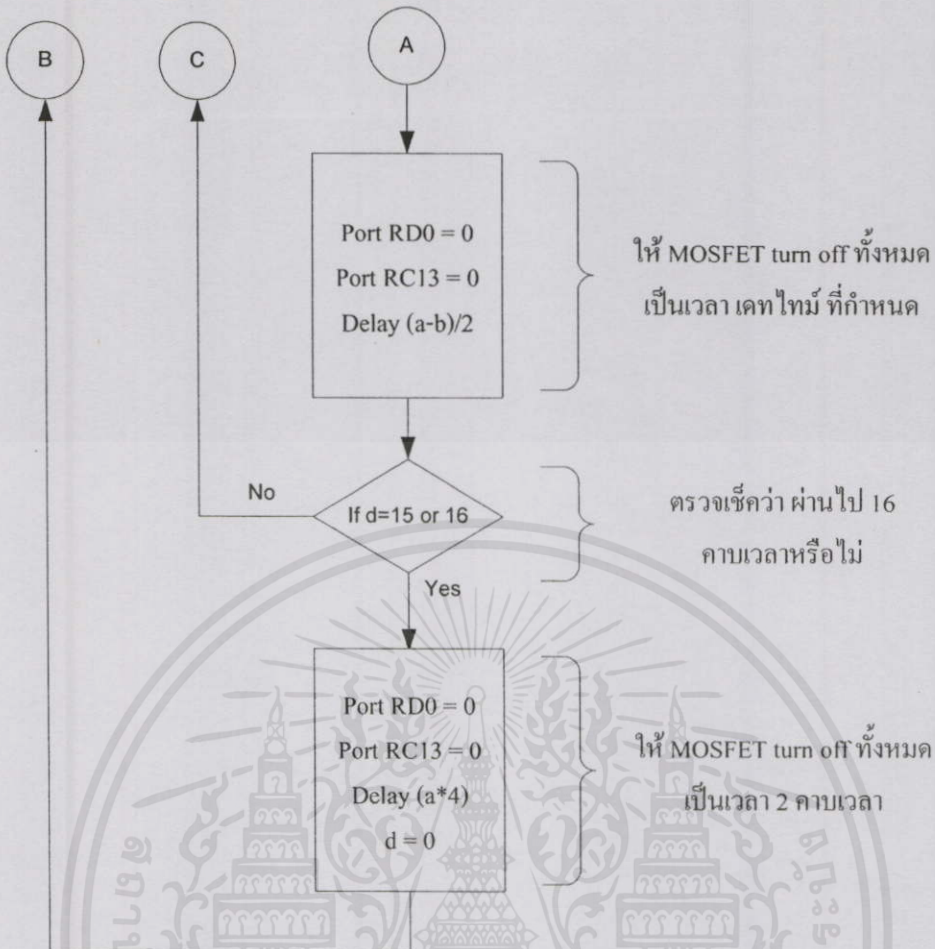
รูปที่ 3.11 Flowchart การสร้างรูปสัญญาณแบบ PS-PWM

3.4 การสร้างสัญญาณ PDM

ในการสร้างสัญญาณ PDM ซึ่งได้กล่าวถึงรูปแบบสัญญาณ และสภาวะการทำงานทั้งหมดไว้แล้วในบทที่ 2 รูปที่ 3.12 เพื่อเขียนเป็นโปรแกรมในการสร้างสัญญาณ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ขออนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งยังมีให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 Flowchart การสร้างรูปสัญญาณแบบ PDM

รูปที่ 3.12 โปรแกรมในการสร้างรูปสัญญาณ PDM เริ่มต้นจากการกำหนดค่าความถี่ที่ต้องการคือ 30 kHz และกำหนดค่าควิซีไซเคิลไว้ที่ 48% แล้วจึงคำนวณค่าออกมาเป็นค่าคาบเวลาที่จะต้องทำงาน และหยุดของรูปสัญญาณแต่ละมอสเฟต

คำนวณได้ค่า $a = 33.33 \mu\text{s}$, $b = 15.99 \mu\text{s}$, $(a-2b)/2$ (คือช่วงเวลาเดทไทม์) = $0.675 \mu\text{s}$, แล้วกำหนดค่าเริ่มต้นต่างๆ ให้เป็นศูนย์ และมอสเฟตทั้งหมดค่าเริ่มต้นให้ turn off จากนั้น ให้ S1 และ S4 turn on เป็นเวลา $15.99 \mu\text{s}$ ช่วงเวลาเดทไทม์ turn off มอสเฟตทั้งหมดเป็นเวลา $0.675 \mu\text{s}$ แล้ว S2 และ S3 ก็ turn on เป็นเวลา $15.99 \mu\text{s}$ ช่วงเวลาเดทไทม์ turn off มอสเฟตทั้งหมดเป็นเวลา $0.675 \mu\text{s}$ ทำอย่างนี้ไปเรื่อยๆ และตรวจสอบว่าเป็นรอบที่ 15 และ 16 หรือเปล่า ถ้าใช่ก็ให้มอสเฟตทั้งหมด turn off

เอกสารนี้จัดทำขึ้นไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ จากโปรแกรมที่เขียน ทั้ง 3 รูปแบบสัญญาณที่ได้กล่าวมาแล้ว เมื่อนำไปโปรแกรมที่ ไมโครโปรเซสเซอร์เพื่อนำไปขับสัญญาณที่ขาเกต ของมอสเฟตเพื่อให้ได้รูปสัญญาณที่โหลด (หัวอัลตราโซนิกรานสควิเซอร์) ตามต้องการ เพื่อให้เกิดคลื่นอัลตราโซนิก ผ่านตัวกลาง

ของเหลวที่บรรจุภายในแท่งค์ และเกิดกระบวนการทำความสะอาดตามที่ได้กล่าวมาแล้ว เรื่อง
ทฤษฎีการทำความสะอาดด้วยคลื่นอัลตราโซนิค

3.5 สรุป

วงจรที่ใช้ขับหัวอัลตราโซนิคทรานสดิวเซอร์นับว่าเป็นส่วนที่สำคัญของเครื่องทำความ
สะอาดด้วยคลื่นอัลตราโซนิค ในบทที่ 3 นี้ได้กล่าวถึงรายละเอียดของวงจรที่ใช้ในการทดลอง
วงจรควบคุม และการสร้างสัญญาณทั้งสามรูปแบบนี้ได้ใช้วงจรแบบเดียวกันทั้งหมดรวมทั้งภาค
กำลัง สัญญาณควบคุมที่ได้แตกต่างกันคือการปรับเปลี่ยนโปรแกรมของไมโครโปรเซสเซอร์เพียง
เท่านั้น ทำให้ง่ายต่อการเปลี่ยนรูปสัญญาณเพื่อใช้ในการทดลองในวิทยานิพนธ์นี้ วิธีการเขียน
โปรแกรมของแต่ละรูปสัญญาณได้กล่าวโดยละเอียดไว้แล้วในบทนี้ เครื่องทำความสะอาดด้วย
คลื่นอัลตราโซนิคที่ใช้ทั่วไป ใช้การป้อนรูปสัญญาณ PWM แบบต่อเนื่อง ในวิทยานิพนธ์นี้ได้
ทดลองการป้อนสัญญาณแบบไม่ต่อเนื่อง โดยได้เลือกรูปแบบ PS-PWM และ PDM มาใช้ในการ
สร้างคลื่นอัลตราโซนิคด้วยเพื่อเปรียบเทียบกันโดยการทดลอง ผลการทดลองดังจะได้อีกกล่าวในบท
ต่อไป

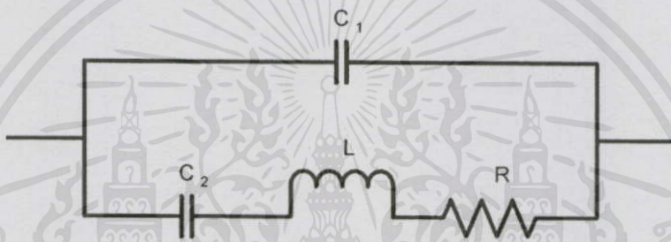
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ขั้นตอนการทดลอง และผลการทดลอง

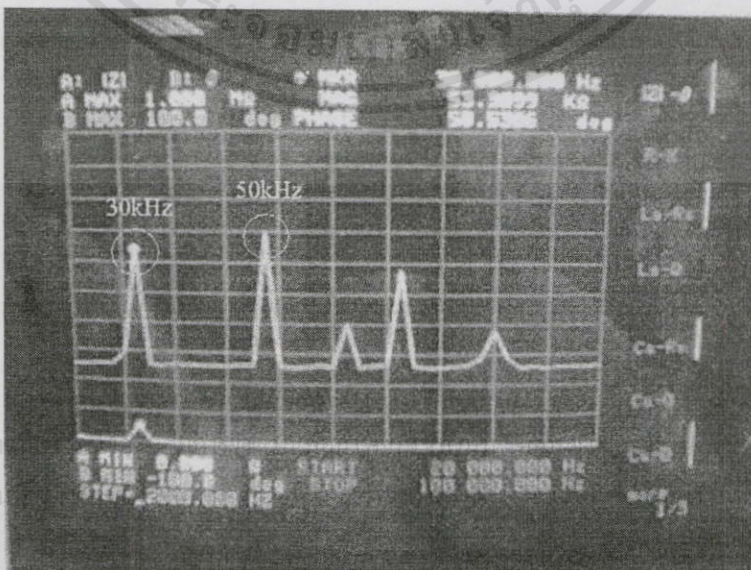
4.1 ขั้นตอนการทดลอง

ในบทนี้จะกล่าวถึงขั้นตอนในการทดลอง และผลการทดลอง ซึ่งอุปกรณ์ที่สำคัญในการสร้างคลื่นอัลตราโซนิก เพื่อให้เกิดกระบวนการทำความสะอาดที่ได้กล่าวมาแล้ว คือ หัวอัลตราโซนิกทรานสดิวเซอร์ ชนิดที่ใช้ยูนีเป็นชนิด เพียโซอิเล็กทริก (Piezoelectric) ซึ่งมีวงจรสมมูลทั่วไปดังรูปที่ 4.1



รูปที่ 4.1 วงจรสมมูล อัลตราโซนิกทรานสดิวเซอร์ที่ใช้ทั่วไป

เมื่อนำหัวอัลตราโซนิกทรานสดิวเซอร์ไปวัดที่เครื่อง RLC Analysis (HP 4194A) จึงได้ค่าออกมาดังนี้ $C_1 = 37.625 \text{ nF}$, $C_2 = 24.1 \text{ nF}$, $L = 460 \text{ } \mu\text{H}$, $R = 26.81 \Omega$ โดยวัดที่ความถี่ 30 kHz ดังรูปที่ 4.2 วัดค่าอิมพีแดนซ์ในช่วงความถี่ 20 kHz ถึง 100 kHz



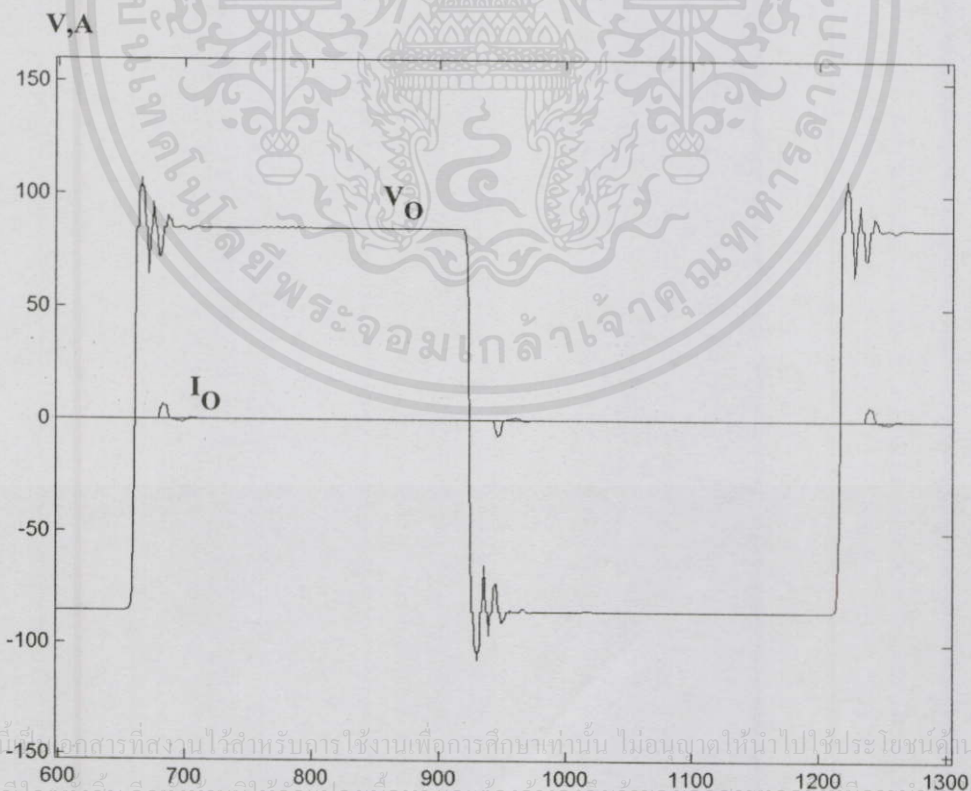
รูปที่ 4.2 วัดค่าอิมพีแดนซ์ที่หัวอัลตราโซนิกทรานสดิวเซอร์

โดยที่ค่าอิมพีแดนซ์ที่วัดได้ถ้าวงจรสมมูลเป็นวงจรอนุกรมจตุเรโซแนนซ์จะมีค่าต่ำสุด และถ้าวงจรสมมูลเป็นวงจรขนานจตุเรโซแนนซ์จะมีค่าขดสูงสุดดังรูปที่ 4.2 ค่าขดสูงสุดจะมีค่าเป็น ช่วงๆตามความถี่ ซึ่งเราเลือกใช้ช่วงที่มีค่าขดสูงสุด เพราะว่าวงจรสมมูลเป็นแบบขนาน และจะได้ค่ากำลังงานออกมาสูงสุดดังสมการที่ (4.1)

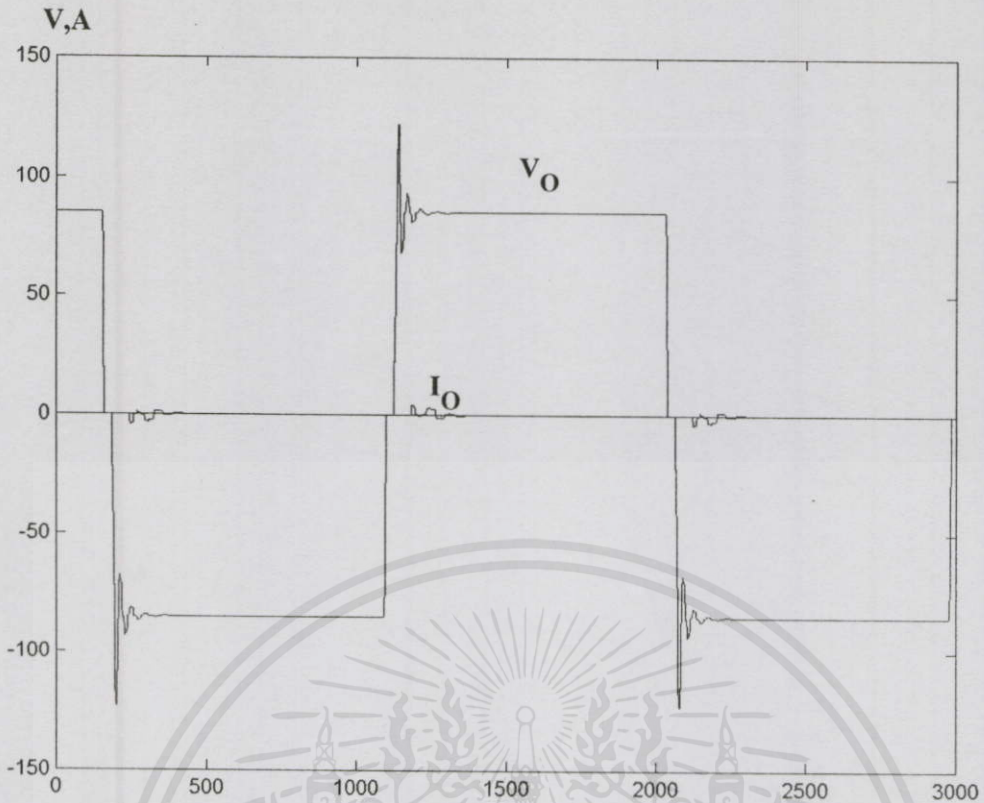
$$P = I^2 R \quad (4.1)$$

เนื่องจากสภาวะเรโซแนนซ์ $X_C = X_L$ จะเหลือแต่เฉพาะ R เท่านั้น ผลที่ได้จะเห็นว่าที่ความถี่ 30 kHz และ 50 kHz จะให้ค่ากำลังงานออกมาสูงสุดในย่าน 20-50 kHz ซึ่งเป็นย่านความถี่ที่ใช้ในการทำความสะอาดด้วยคลื่นอัลตราโซนิค ย่านความถี่ที่สูงขึ้นไปนั้น จะเหมาะสำหรับใช้คลื่นอัลตราโซนิคในงาน วักระยะทาง สแกนโครงสร้างในงานโยธา และ อัลตราซาวด์ในงานทางการแพทย์ เป็นต้น แต่จะเห็นได้ว่าที่ความถี่ 50 kHz มีค่ากำลังสูงสุด เพื่อลดความเค้น (Stress) ของอุปกรณ์จึงเลือกค่าความถี่ที่ 30 kHz ในการทดลอง ซึ่งสอดคล้องกับความถี่ที่การแนะนำที่ได้จาก Data Sheet [8]

จากค่าอิมพีแดนซ์ที่วัดได้นำไปซิมมูลेटโดยใช้โปรแกรม MATLAB เพื่อดูรูปแรงดันและกระแส ที่หัวอัลตราโซนิคทรานสดิวเซอร์ได้ดังรูปที่ 4.3 และ 4.4

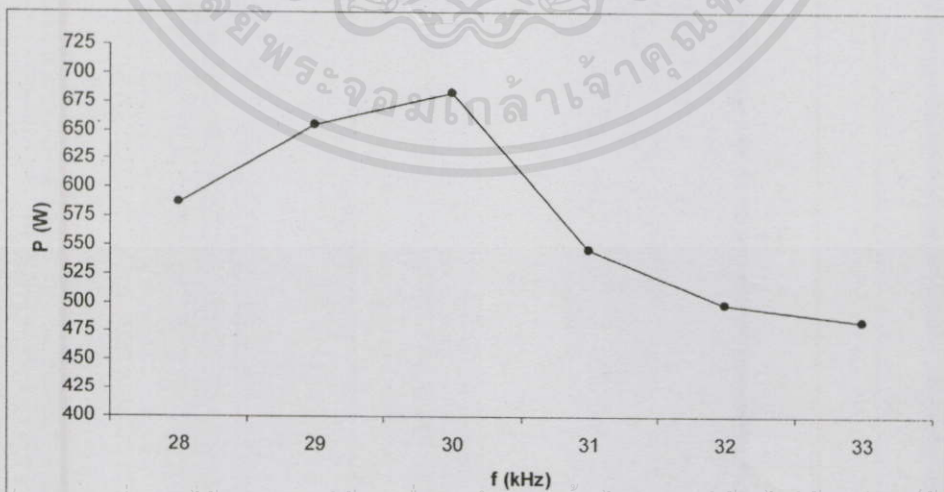


รูปที่ 4.3 รูปแรงดัน และกระแสที่หัวอัลตราโซนิคทรานสดิวเซอร์โดยการซิมมูลेटด้วยโปรแกรม MATLAB ด้วยการป้อนรูปสัญญาณ PWM



รูปที่ 4.4 รูปแรงดัน และกระแสที่หัวอัลตราโซนิคทรานสดิวเซอร์ โดยการซิมูเลตด้วยโปรแกรม MATLAB ด้วยการป้อนรูปสัญญาณ PS-PWM

จากนั้นจึงนำสัญญาณ PWM ป้อนที่หัวอัลตราโซนิคทรานสดิวเซอร์ ที่ความถี่รอบๆ ความถี่เรโซแนนท์ เพื่อความถี่ 30 kHz ได้ค่ากำลังงานออกมาสูงสุด 682 W ดังรูปที่ 4.5



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานานาชาติ ไม่อนุญาตให้ทำซ้ำโดยไม่ได้รับอนุญาต

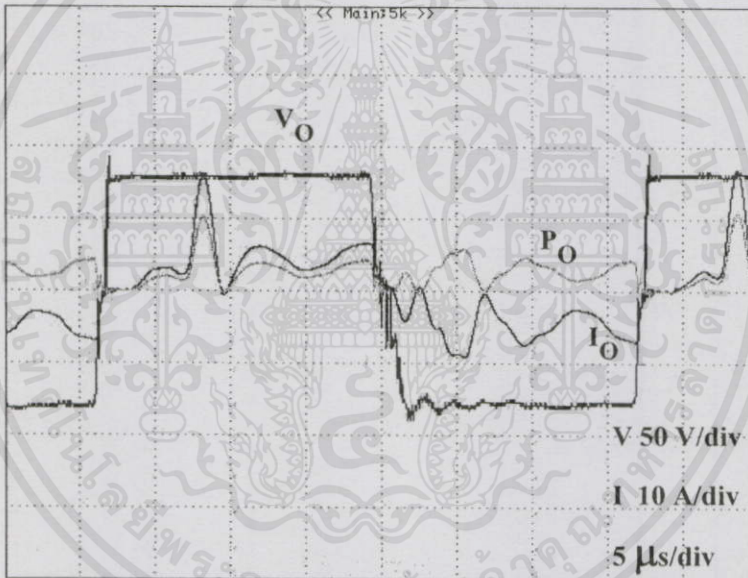
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.5 กำลังไฟฟ้าที่ความถี่รอบๆ ความถี่เรโซแนนท์

จากการทดลองที่ได้ป้อนรูปสัญญาณการขับมอเตอร์ทั้ง 3 แบบเมื่อวัดสัญญาณ ของแรงดันและกระแสออกมาได้ดังรูปที่ 4.6 จะเห็นได้ว่า รูปสัญญาณของกระแสไม่ได้อยู่ในรูปคลื่นไซน์ (Sinusoidal Waveform) ทำให้การคำนวณหาค่ากำลังเอาต์พุต ต้องหาโดยวิธี จากการหาค่าเฉลี่ยกำลังงานเอาต์พุต (P_o) ดังสมการที่ (4.2)

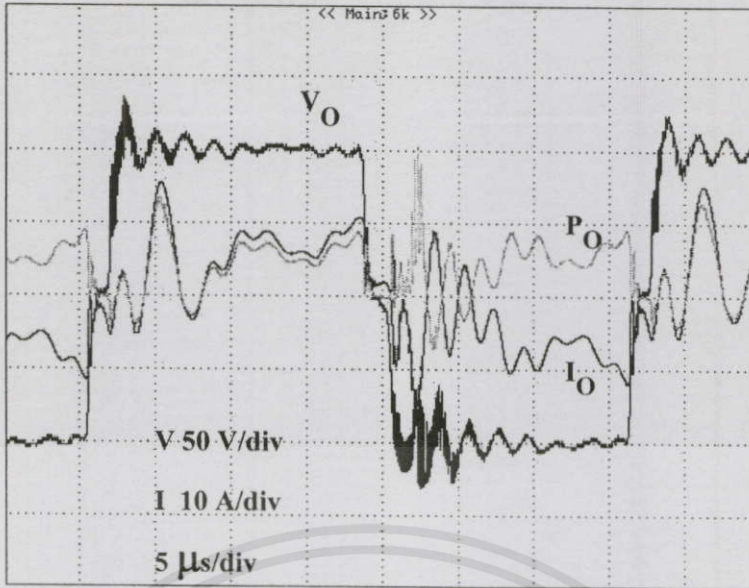
$$P_o = \frac{1}{T} \int_0^T V_o I_o dt \quad (4.2)$$

โดยที่ V_o คือแรงดันสถานะชั่วครู่ (instantaneous) และ I_o คือ กระแสสถานะชั่วครู่ การวัดค่าแรงดันและกระแส ได้จากคิิจิตอลออสซิลโลสโคป (YOKOGAWA DL1620) แล้วใช้ฟังก์ชันคณิตศาสตร์ ในการหาค่ากำลังงานไฟฟ้าเอาต์พุต

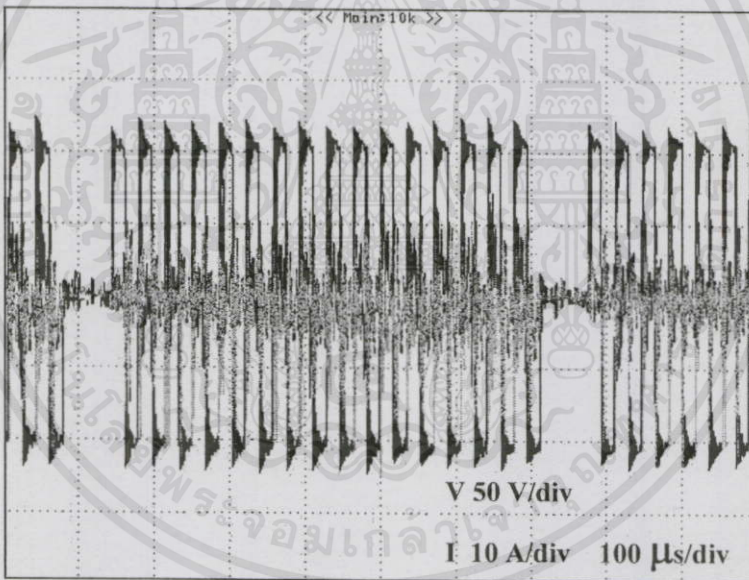


(a)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะมิใช่ก็ตาม อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(b)



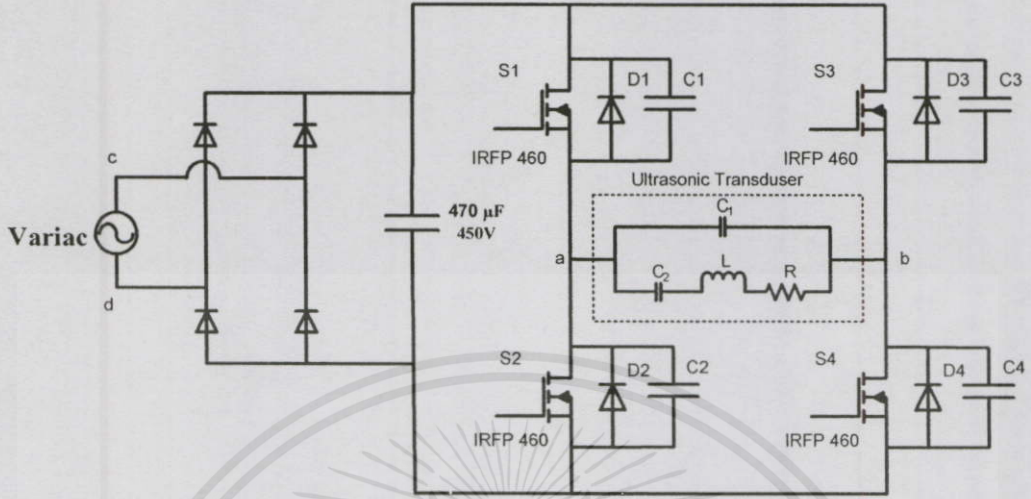
(c)

รูปที่ 4.6 รูปแรงดันและกระแสเอาต์พุตเมื่อป้อนสัญญาณแบบ

(a) PWM (b) PS-PWM (c) PDM

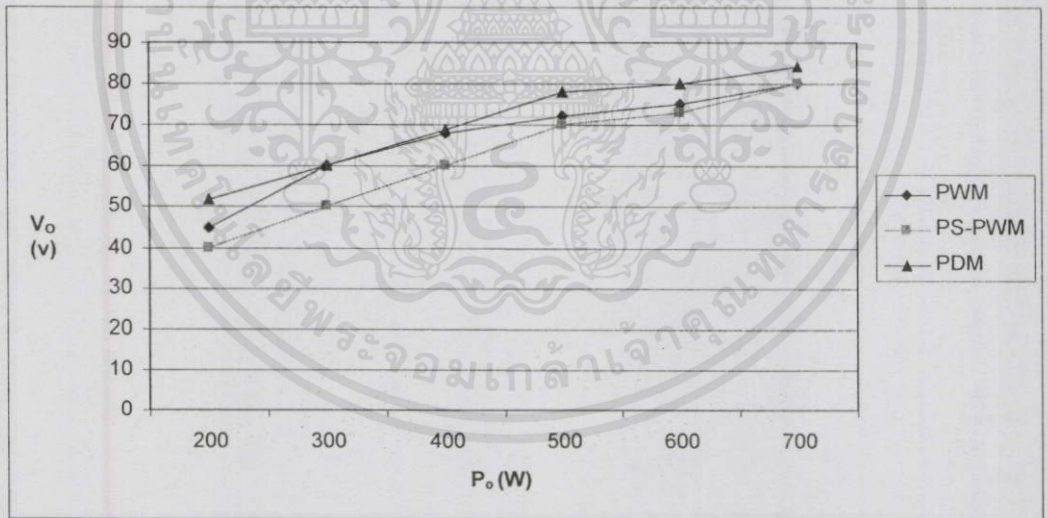
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการทำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้แก้ไขหรือใช้ประโยชน์ด้านการค้า
 จากนั้นทำการวัด โดยการเพิ่ม ค่าแรงดันเพื่อดูค่ากำลังงาน ไฟฟ้าที่ออกมาของแต่ละ
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกขั้นหนึ่งก็ได้ทดลองเนื้อหา และต้องอ้างอิงถึงเนื้อหาของเอกสารที่กล่าวไปใช้
 รูปคลื่นโดยแสดงได้ดังรูปที่ 4.7 โดยที่ V_{in} , I_{in} คือแรงดันและกระแสขาเข้าของวงจรรีกติไฟเออร์
 จุด cd จากรูปที่ 4.7 และ V_o , I_o คือแรงดันและกระแสของเอาต์พุต (โหลด) ของวงจรรีกติไฟเออร์

ของจุด ab จากรูปที่ 4.6 และก็ทำการวัดเปรียบเทียบค่าประสิทธิภาพของวงจรที่ได้จากการควบคุมรูปแบบคลื่นทั้งสามแบบ ที่กำลังไฟฟ้าต่าง ๆ กันที่ได้จากรูปที่ 4.6 จะได้ผลดังแสดงในรูปที่ 4.9



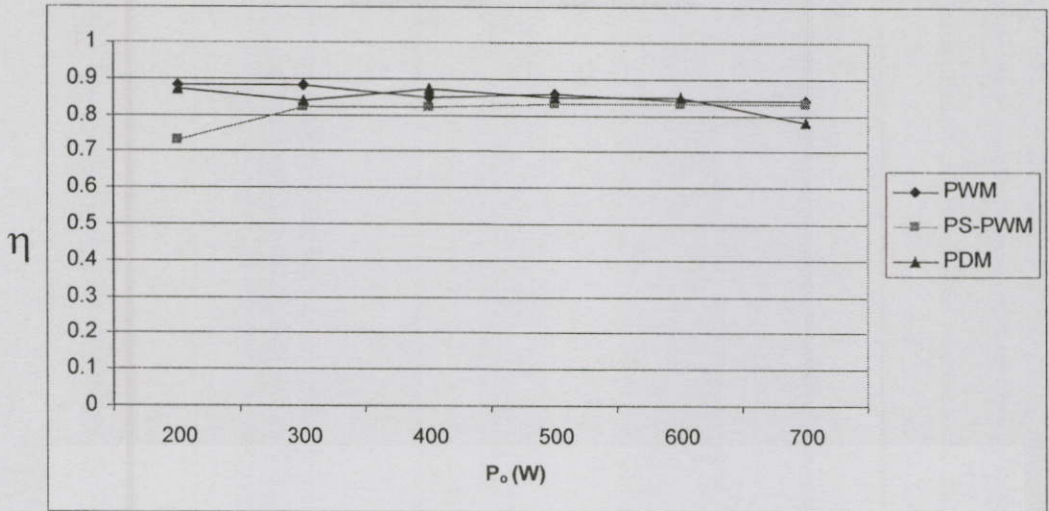
รูปที่ 4.7 รูปวงจรที่วัดค่า

4.2 ผลการทดลอง



รูปที่ 4.8 ค่ากำลังงานไฟฟ้าที่แรงดันต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะมิใช่ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

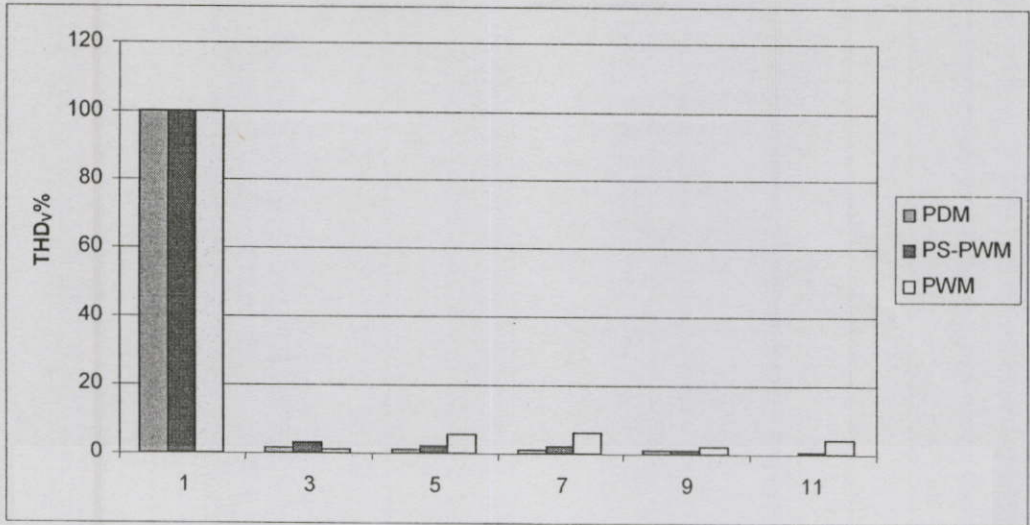


รูปที่ 4.9 ค่าประสิทธิภาพกับกำลังงานไฟฟ้าต่างๆ

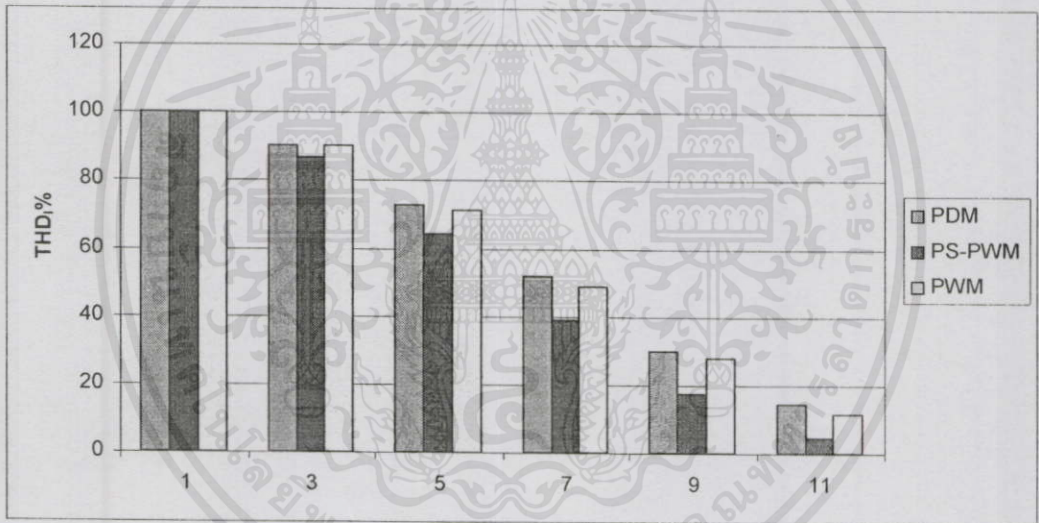
จากรูปที่ 4.8 จะเห็นได้ว่า ที่ค่ากำลังงานไฟฟ้าที่เท่ากันของรูปสัญญาณทั้ง 3 แบบ รูปแบบ PDM ให้แรงดันเอาต์พุตมากกว่าทุกรูปคลื่น เนื่องจากค่าเฉลี่ยกระแสเอาต์พุตสูงกว่าจากการมีช่วงหยุดนำกระแส 2 คาบเวลาใน 18 คาบเวลา ในรูปที่ 4.9 ในค่ากำลังงานไฟฟ้าตั้งแต่ 300 วัตต์ขึ้นไป รูปสัญญาณทั้ง 3 แบบ มีค่าประสิทธิภาพแตกต่างกัน 5-10% โดยประมาณ เนื่องจากผลจากฮาร์มอนิกของแรงดัน และกระแสที่เกิดขึ้นจากทางโหลด และไหลย้อนกลับไปทางแหล่งจ่ายเป็นผลทำให้รูปคลื่นแรงดัน และกระแสมีค่าสูงขึ้นและผิดเพี้ยนไปจากเดิมทำให้ประสิทธิภาพที่ออกมาจึงมีค่ามากกว่าความเป็นจริง และไม่แตกต่างกัน ดังที่ได้กล่าวไว้แล้วในบทที่ 2 เรื่องฮาร์มอนิกและผลกระทบที่เกิดจากฮาร์มอนิก

จากนั้นทำการวัดค่าฮาร์มอนิก (Harmonic) โดยใช้ เครื่องมือวัด Fluke 43B วัดทางด้านเอซีไลน์ออกมาได้ดังรูปที่ 4.10 และ 4.11 การวัดค่าฮาร์มอนิกจะเห็นได้ว่า ในการวัดที่แรงดันรูปคลื่น PDM จะมีค่าแอมพลิจูดน้อยกว่าทุกรูป และในการวัดที่กระแส รูปคลื่น PS-PWM มีค่าแอมพลิจูดน้อยกว่าทุกรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะมิใช่ใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



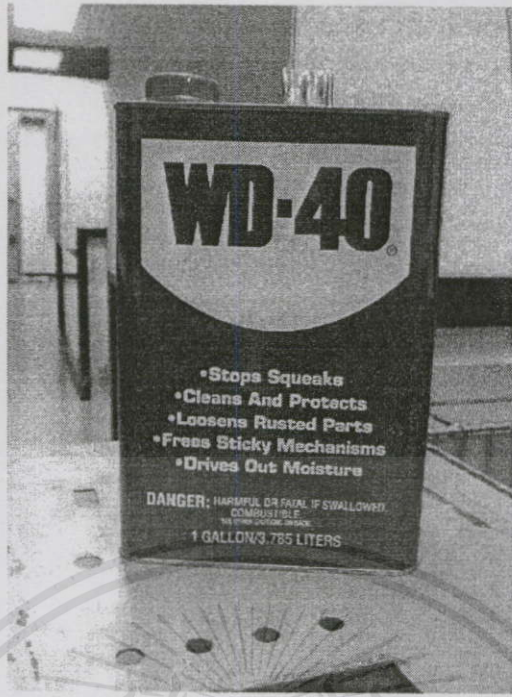
รูปที่ 4.10 วิศวค่าฮาร์มอนิกของแรงดันเอซีอินพุท



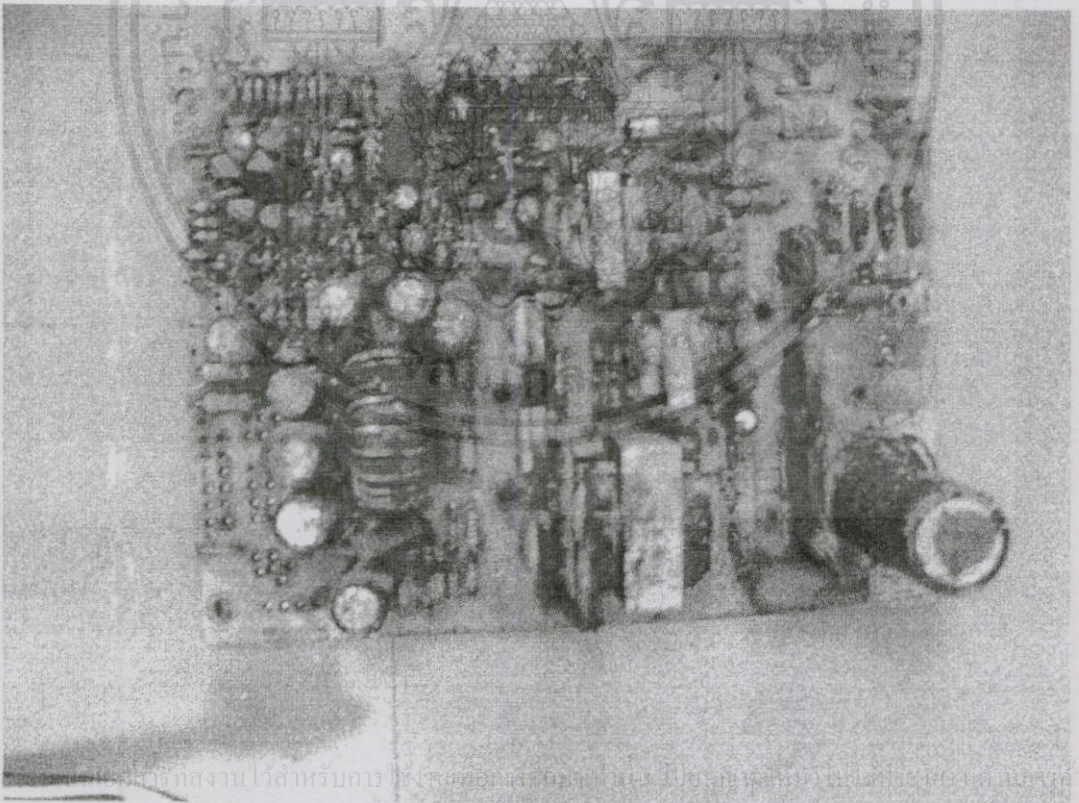
รูปที่ 4.11 วิศวค่าฮาร์มอนิกของกระแสเอซีอินพุท

โดยที่ $THD_v\%$ ของแรงดันอินพุทเอซีไลน์สำหรับหัวอัลตราโซนิคเมื่อถูกกระตุ้นด้วย สัญญาณ PWM = 100, 0.8, 5.3, 6.2, 1.9, 4 PS-PWM = 100, 2.8, 2.2, 1.8, 1.2, 0.4 และ PDM = 100, 1.4, 1.1, 1.1, 0.9, 0.2 $THD_v\%$ ของกระแสอินพุทเอซีไลน์สำหรับหัวอัลตราโซนิคเมื่อถูกกระตุ้นด้วย สัญญาณ PWM = 100, 90.1, 71.2, 48.6, 28.1, 11.6 PS-PWM = 100, 86.8, 64, 38.9, 17.5, 4.6 และ PDM = 100, 89.9, 72.6, 51.7, 29.7, 14.5 ที่ $n = 1, 3, 5, 7, 9, 11$ ตามลำดับ

จากนั้นได้ทำการชำระล้างบอร์คิเล็กทรอนิกส์ขนาด 15x18 เซนติเมตร 3 แผง ซึ่งมีอุปกรณ์บนบอร์คิแสดงในรูปที่ 4.13 และมีน้ำยาที่ใช้ในการทดลองดังรูปที่ 4.12 ซึ่งใช้น้ำยา 3.785 ลิตร [9] ในถังขนาด 40 ลิตร



รูปที่ 4.12 น้ำยาที่ใช้ทำความสะอาดอุปกรณ์ไฟฟ้าและอิเล็กทรอนิกส์ที่นำมาใช้ในการทดลอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานของนักศึกษาเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.13 ขอรด์อิเล็กทรอนิกส์ที่นำมาใช้ชำระล้างในการทดลอง

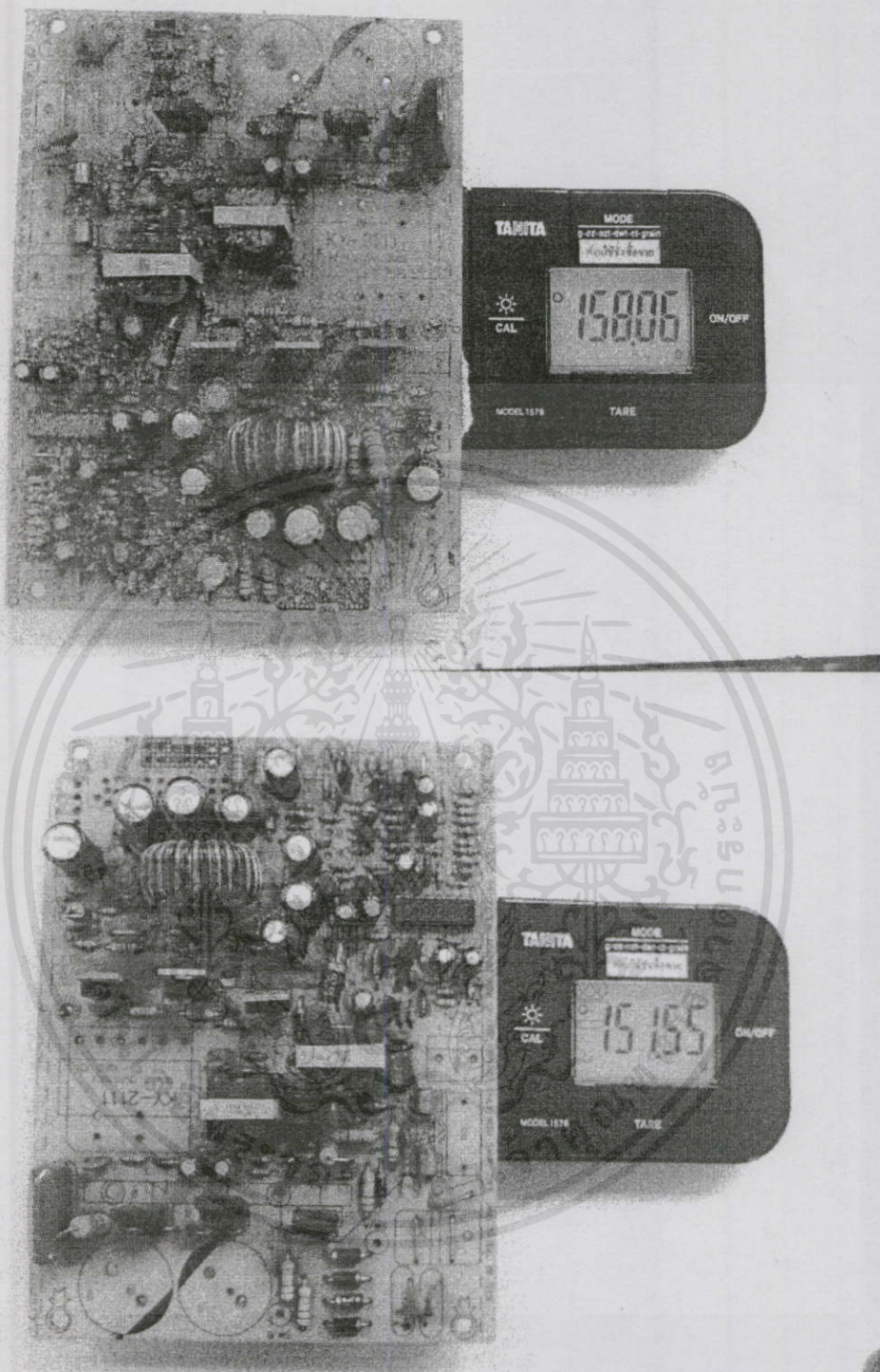
จากนั้นทำการวัดความสกปรกโดยได้นำบอร์ดอิเล็กทรอนิกส์ขนาด 15x18 เซนติเมตร นำฝุ่นมาใส่แล้วชั่งน้ำหนัก นำไปชำระล้าง เป็นเวลา 15 นาทีแล้วเป่าให้แห้ง และจึงนำมาชั่งน้ำหนัก ดังรูปที่ 4.14 รูปข้างบนคือน้ำหนักก่อนไปชำระล้าง รูปด้านล่างคือน้ำหนักชำระล้างแล้ว โดยครั้งแรกชั่งน้ำหนักที่มีฝุ่นได้ 158.06 กรัม แล้วจึงนำไปชำระล้างแล้วเป่าให้แห้ง นำมาชั่งน้ำหนักหลังชำระล้างด้วยรูปสัญญาณ PS-PWM ได้น้ำหนัก 151.55 กรัม ครั้งที่สองชั่งน้ำหนักที่มีฝุ่นได้ 159.36 กรัม แล้วจึงนำไปชำระล้างแล้วเป่าให้แห้ง นำมาชั่งน้ำหนักหลังชำระล้างด้วยรูปสัญญาณ PDM ได้น้ำหนัก 151.30 กรัม ครั้งที่สามชั่งน้ำหนักที่มีฝุ่นได้ 158.05 กรัม แล้วจึงนำไปชำระล้างแล้วเป่าให้แห้ง นำมาชั่งน้ำหนักหลังชำระล้างด้วยรูปสัญญาณ PWM ได้น้ำหนัก 152.19 กรัม

จากการทดลองน้ำหนักที่หายไป หลังจากชำระล้าง โดยรูปสัญญาณ PS-PWM, PDM และ PWM คือ 6.51, 8.06, และ 5.86 กรัมตามลำดับและบันทึกผลลงในตารางที่ 4.1 จากผลการทดลองที่ได้ PDM จะได้ดีที่สุด รองลงมาคือ PS-PWM และ PWM ตามลำดับซึ่งเหมาะสำหรับอุปกรณ์ที่นำมาชำระล้างชนิดผิวไม่เรียบ ซึ่งจากการใช้สัญญาณแบบ PS-PWM ก็ทำให้ผลการชำระล้างดีกว่าแบบเดิมคือ PWM แต่เนื่องจากเมื่อปรับเลื่อนเฟสออกไปเยอะ กำลังงานไฟฟ้าที่โหลดก็จะตกลงต้องมีการปรับแรงดันเพิ่ม เมื่อจำเป็นต้องปรับเยอะเพื่อต้องการกระบวนการทำความสะอาดที่เหมาะสมกับอุปกรณ์ที่นำมาชำระล้างนั้นก็ควรใช้แบบ PDM จะเหมาะกว่าแต่จะไม่มีการหมุนเวียนของสารตัวกลางที่ใช้ทำความสะอาด เนื่องจากแบบ PDM จะเกิดอิมโพลชันน้อยกว่าซึ่งได้กล่าวไว้ในบทที่ 2 เรื่องกระบวนการทำความสะอาด

ตารางที่ 4.1 ค่าที่วัดความสกปรกก่อนและหลังชำระล้างเปรียบเทียบกันทั้งสามรูปแบบสัญญาณ

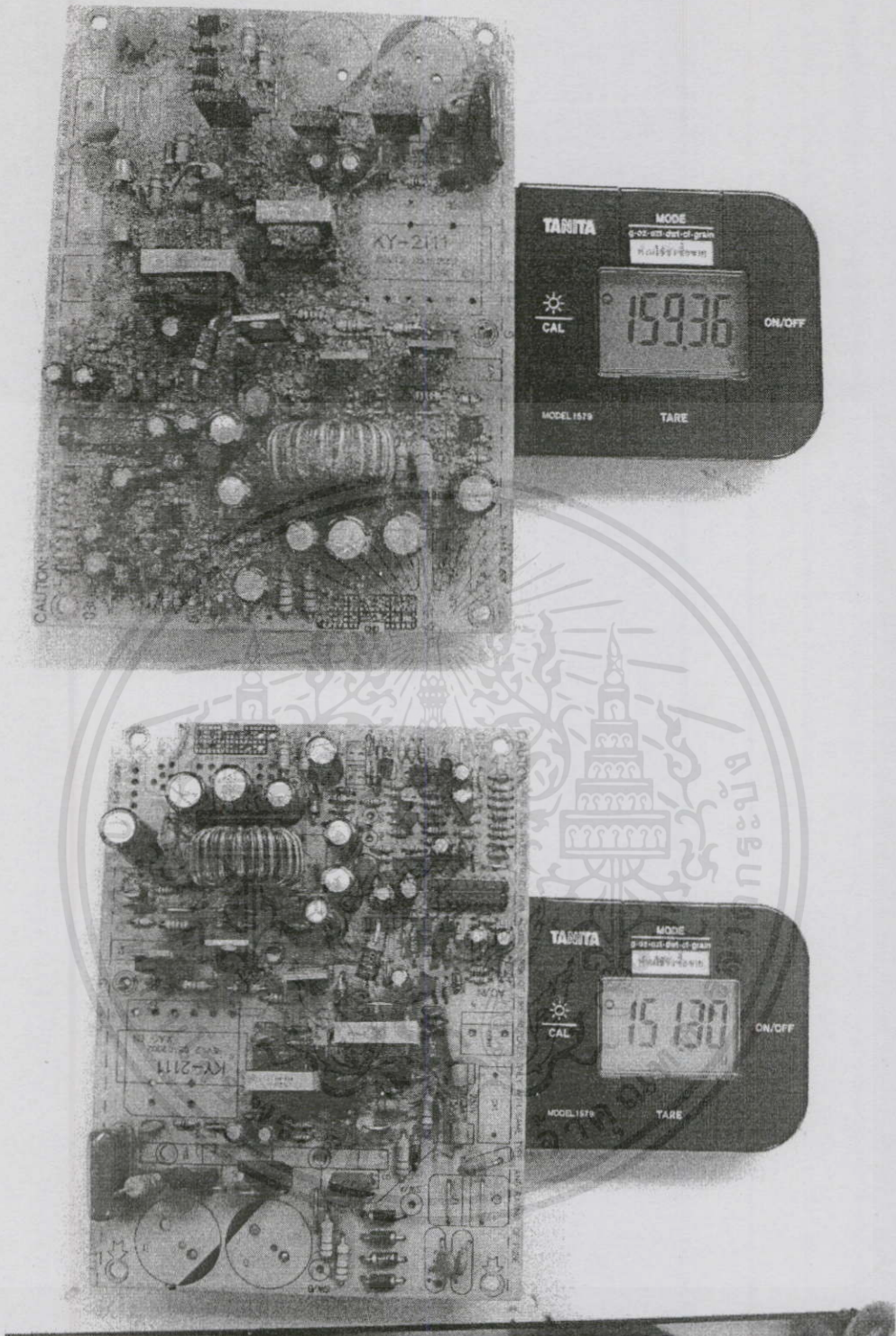
	น้ำหนัก (กรัม)				อุปกรณ์เปล่าๆ	%
	ก่อนชำระล้าง	หลังชำระล้าง	ฝุ่นที่หายไป			
PS-PWM	158.06	151.55	6.51	149.89	79.68	
PWM	158.05	152.19	5.86	149.89	71.81	
PDM	159.36	151.3	8.06	149.89	85.11	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



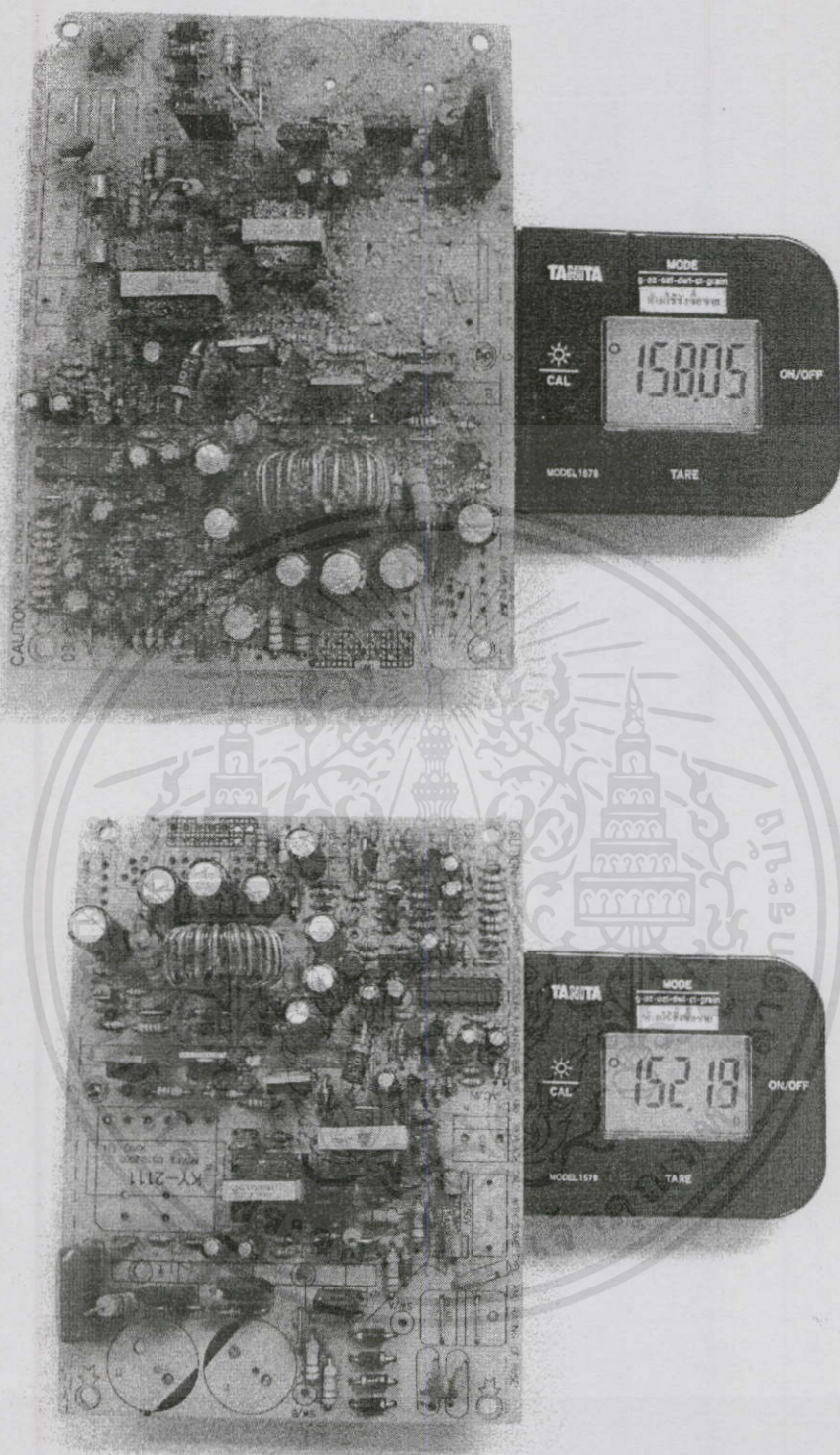
(a)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะฉีกใดทั้งคืน อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(b)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

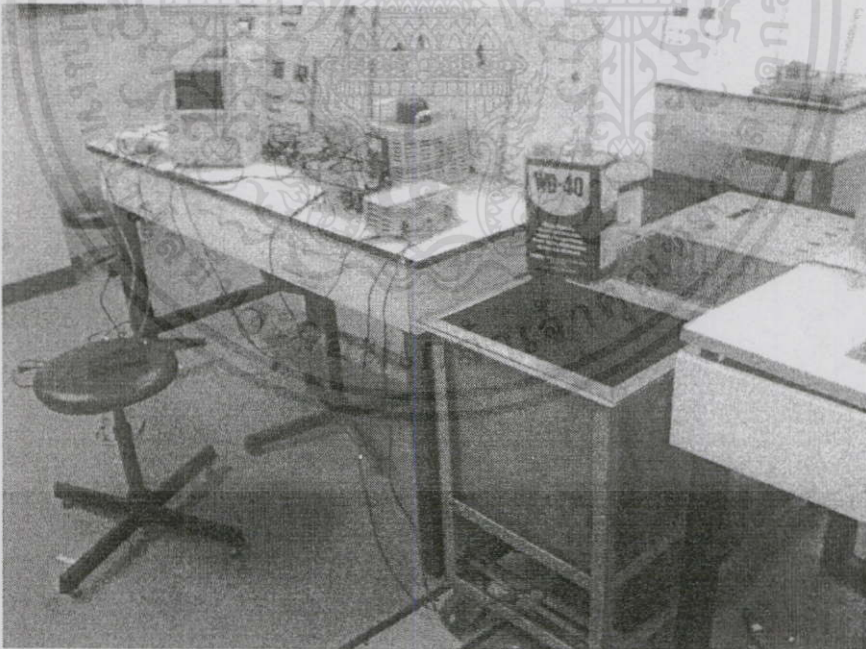


(c)

เอกสารนี้เป็นเอกสารรูปที่ 4.14 รูปชั่งน้ำหนักก่อนชำระล้าง และน้ำหนักหลังชำระล้าง ระเบียบด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้นำข้อมูลอันเป็นความลับของเอกสารทุกครั้งที่มีการนำไปใช้
 (a) PS-PWM (b) PDM (c) PWM

4.3 สรุป

รูปที่ 4.15 คือเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิคที่ใช้ในทดลองนี้ จากการทดลองโดยป้อนรูปสัญญาณเพื่อสร้างคลื่นอัลตราโซนิค แบบต่อเนื่อง (PWM) และแบบไม่ต่อเนื่อง (PWM แบบเลื่อนเฟส และ PDM) นั้น จากปฏิกิริยากระบวนการทำความสะอาดจะเห็นได้ว่า เมื่อป้อนสัญญาณแบบ PWM และ PWM แบบเลื่อนเฟส จะเห็นว่าของเหลวเกิดการแกว่งตัว และมีฟองอากาศ ต่างกับ PDM ซึ่งของเหลวจะนิ่งแต่มีฟองอากาศมากกว่า เกิดฟองอากาศเร็วกว่า ซึ่งคลื่น 2 แบบแรก ต้องใช้เวลาเกือบครึ่งชั่วโมงถึงจะเกิดฟองอากาศ แต่ PDM ใช้เวลาแค่ 5 นาทีก็เกิดฟองอากาศที่มีขนาดเล็กกว่า และมีจำนวนมากกว่า ซึ่งแบบ PDM เหมาะแก่อุปกรณ์ที่ผิวไม่ราบเรียบ แต่จะไม่มี การหมุนเวียนของน้ำยาทำความสะอาด แบบ PWM และ PWM แบบเลื่อนเฟส เหมาะแก่การทำ ความสะอาดอุปกรณ์ที่ผิวราบเรียบ และต้องการหมุนเวียนน้ำยาทำความสะอาด ซึ่งการใช้สัญญาณแบบไม่ต่อเนื่องของรูปสัญญาณ PWM แบบเลื่อนเฟส กระบวนการเกิดฟองอากาศไม่เหมือนกับ PDM เพราะว่าการหยุดการสร้างคลื่นของ PWM แบบเลื่อนเฟสหยุดแล้วสร้างอย่างนี้ไปเรื่อยๆ แต่ PDM สร้างไปเรื่อยๆแล้วหยุดนานกว่า ทำให้ความเร็วชั้นขุบตัวลงทำให้เกิดการอิมโพลชั่นน้อยกว่า ทำให้ไม่มี การหมุนเวียนของสารตัวกลางที่ใช้ทำความสะอาด



รูปที่ 4.15 เครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิค

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามทำซ้ำโดยไม่ขออนุญาตจากเจ้าของลิขสิทธิ์ ผู้ที่นำเอกสารนี้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการวิจัย และข้อเสนอแนะ

5.1 บทสรุป

เครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิก อุปกรณ์ที่สำคัญคืออัลตราโซนิกทรานสดิวเซอร์ เพื่อทำให้เกิดคลื่นความถี่ย่านอัลตราโซนิก เพื่อส่งผ่านตัวกลางของเหลว ทำให้เกิดกระบวนการทำความสะอาดเกิดขึ้น ดังที่ได้กล่าวมาแล้วในบทที่ 2 ความถี่ย่านที่ใช้ทำความสะอาดคือ ช่วงความถี่ 20 kHz – 50 kHz ซึ่งแต่ละความถี่ก็เหมาะกับงานทำความสะอาดอุปกรณ์แต่ละประเภท อัลตราโซนิกทรานสดิวเซอร์จะทำงานก็ต่อเมื่อ วงจรขับหัวอัลตราโซนิกทรานสดิวเซอร์สร้างรูปสัญญาณไซน์, รูปสัญญาณสี่เหลี่ยม หรือ รูปสัญญาณพัลส์ ให้มีความถี่เท่ากับความถี่เรโซแนนท์ทางกลของหัวนั้น

การสร้างรูปสัญญาณที่นำมาใช้ในการขับหัวอัลตราโซนิก ได้มาจากตัวไมโครโปรเซสเซอร์ dsPIC โดยการเขียนโปรแกรมในการสร้างรูปสัญญาณทั้งสามแบบ แล้วส่งให้วงจรขับเกตมอสเฟตของวงจรภาคกำลัง เพื่อขับโหลด (หัวอัลตราโซนิกทรานสดิวเซอร์) จำนวน 10 หัว โดยการทดลองนี้ เลือกใช้ไมโครโปรเซสเซอร์เพื่อง่ายต่อการสร้างสัญญาณทั้งสามแบบ และไม่มีผลกับการเลือกใช้สัญญาณไหนในการทำความสะอาด

ในวิทยานิพนธ์ฉบับนี้ได้นำเสนอ วิธีการป้อนคลื่นสัญญาณอีกสองชนิดให้กับหัวอัลตราโซนิกทรานสดิวเซอร์เพื่อเป็นอีกทางเลือกหนึ่ง ในการนำไปปรับใช้กับอุปกรณ์ที่จะนำมาชำระล้างให้เหมาะสม จากแต่เดิมที่ใช้คลื่นแบบ PWM เพียงอย่างเดียว ถ้าจะให้ฟองอากาศมีขนาดเล็กเพื่อให้เหมาะสมกับ อุปกรณ์ที่มีซอกเล็กซอกน้อยก็จะต้องเปลี่ยนหัวอัลตราโซนิกทรานสดิวเซอร์ ให้มีความถี่สูงขึ้น และต้องสร้างสัญญาณที่มีความถี่สูงเท่ากับความถี่เรโซแนนท์ของหัวอัลตราโซนิกทรานสดิวเซอร์ด้วยซึ่งนำสัญญาณแบบ PDM ไปใช้ก็จะใช้หัวอัลตราโซนิกทรานสดิวเซอร์เพียงความถี่เรโซแนนท์เดียวแต่ปรับใช้ได้กับหลายอุปกรณ์ที่จะนำมาชำระล้าง

วิธีการที่นำเสนอในวิทยานิพนธ์เป็นเทคนิคหนึ่งเท่านั้นที่ช่วยในการปรับปรุงสมรรถนะของเครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิกให้สามารถนำไปใช้ ชำระล้างอุปกรณ์ได้หลากหลายชนิดขึ้นโดยไม่จำเป็นต้องมีหลายเครื่อง หลายความถี่เรโซแนนท์ของหัวอัลตราโซนิกทรานสดิวเซอร์ โดยนำเครื่องเก่าที่มีอยู่เดิมนำมาปรับปรุง ตัวสร้างสัญญาณให้เป็นแบบ PDM หรือ PS-PWM

ในการทดลองได้ป้อนสัญญาณทั้งสามรูปแบบเพื่อเปรียบเทียบกัน โดยได้วัดค่ากำลังงานเฉลี่ยเอาต์พุต รูปสัญญาณ ค่าฮาร์มอนิก และผลการทดลอง ดังที่ได้แสดงไว้ในบทผลการทดลอง

จากการทดลอง โดยป้อนรูปสัญญาณเพื่อสร้างคลื่นอัลตราโซนิก แบบต่อเนื่อง (PWM) และแบบไม่ต่อเนื่อง (PS-PWM และ PDM) นั้น จากปฏิกิริยากระบวนการทำความสะอาดจะเห็นได้ว่า เมื่อป้อนสัญญาณแบบ PWM และ PS-PWM จะเห็นว่าของเหลวเกิดการแกว่งตัว และมีฟองอากาศ ต่างกับ PDM ซึ่งของเหลวจะนิ่งแต่มีฟองอากาศมากกว่า เกิดฟองอากาศเร็วกว่า ซึ่งคลื่น 2 แบบแรก ต้องใช้เวลาเกือบครึ่งชั่วโมงถึงจะเกิดฟองอากาศ แต่ PDM ใช้เวลาแค่ 5 นาทีก็เกิดฟองอากาศที่มีขนาดเล็กกว่า และมีจำนวนมากกว่า ซึ่งการใช้สัญญาณแบบไม่ต่อเนื่องของรูปสัญญาณ PWM แบบเลื่อนเฟส กระบวนการเกิดฟองอากาศ ไม่เหมือนกับ PDM เพราะว่าการหยุดการสร้างคลื่นของ PWM แบบเลื่อนเฟสหยุดแล้วสร้างอย่างนี้ไปเรื่อยๆ แต่ PDM สร้างไปเรื่อยๆ แล้วหยุดนานกว่า ทำให้ควิเดชั่นยุบตัวลงทำให้เกิดการอิมโพลชันน้อยกว่าตามที่ได้กล่าวไว้แล้ว ในบทที่ 2 ในเรื่องของกระบวนการในการทำความสะอาด ในการทดลองทำความสะอาด PDM ดีที่สุดในการใช้ทำความสะอาดอุปกรณ์ที่ผิวไม่เรียบ รองลงมาคือ PS-PWM และ PWM ตามลำดับ

5.2 ข้อเสนอแนะ

เครื่องทำความสะอาดด้วยคลื่นอัลตราโซนิก เป็นเครื่องทำความสะอาดที่มีเสียงดังรบกวน เนื่องจากหัวอัลตราโซนิกทรานสดิวเซอร์ติดกับตัวถังอะลูมิเนียม และมีการสั่นทำให้ตัวหัวอัลตราโซนิกทรานสดิวเซอร์ขัดถูกับตัวถังจนเกิดมีเสียงดัง ดังนั้นควรจะมีการปรับปรุงเรื่องการเก็บเสียงให้ดังน้อยลง โดยการทำให้ตัวถังเป็นระบบปิด แล้วใช้วัสดุเก็บเสียงปิดโดยรอบตัวเครื่อง ตัวถังที่ใช้เป็นอะลูมิเนียมขนาดความจุได้ 40 ลิตร ตัวโครงสร้างเครื่องเป็นหลักทำให้ตัวเครื่องทั้งหมดมีน้ำหนักมากถ้ามีการเปลี่ยนชนิดของตัวถังหรือโครงสร้างที่มีน้ำหนักเบา ก็จะทำให้สะดวกในการเคลื่อนย้าย และถ้ามีการเพิ่มหัว อัลตราโซนิกทรานสดิวเซอร์ให้มีจำนวนมากขึ้นก็จะทำให้เวลาในการทำความสะอาดน้อยลง และควรมีการทดลองเรื่องผลจากค่าเคคไทม์ของรูปแบบ PWM, PS-PWM และ PDM ซึ่งมีผลต่อกระบวนการทำความสะอาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] A. Jangwanitlert, P. Paisuwana and T. Vijaktakul, "Ultrasonic cleaner", Proc. in EECON 22, Kasetsart University, Vol 22, 2542.
- [2] H. Fujita and H. Akagi, "Control and Performance of a Pulse-Density-Modulated Series-Resonant Inverter for Corona Discharge Processes", IEEE Trans. on Industry Application, 1999, Vol. 35, pp. 621-627.
- [3] H. Kifune, Y. Hatanaka and M. Nakaoka, "Cost effective phase shifted pulse modulation soft switching high frequency inverter for induction heating applications", Proc. IEE Electrical Power Appl., 2004, Vol. 151, pp. 19-25.
- [4] <http://www.caeblackstone.com/technical.html>, CAE Blackstone co., Ltd., Jamestown NY, 1998.
- [5] วีระเชษฐ ชันเงิน และ วุฒิปท ธาราธิรเศรษฐ์, "อิเล็กทรอนิกส์กำลัง", ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2547
- [6] Wagner, V.E.; Balda, J.C.; Griffith, D.C.; McEachern, A.; Barnes, T.M.; Hartmann, D.P.; Phileggi, D.J.; Emanuel, A.E.; Horton, W.F.; Reid, W.E.; Ferraro, R.J.; Jewell, W.T.; "Effects of harmonics on equipment", IEEE Transaction on Power Delivery, April 1993, Vol.8, No.2, pp.672-680
- [7] N. Pakdeechart, N. Wongsuntornchai and C. Reimpornjitwirai, "Basic dsPIC Microcontroller with C Programming (MPLAB C30)", Innovative experiment co., Ltd., 2004
- [8] Belincourt, don. "Ultrasonic transducer materials." New-York, Plenum press, 1971.
- [9] <http://www.wd40.com>, WD-40 company co., Ltd., California USA, 2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก.

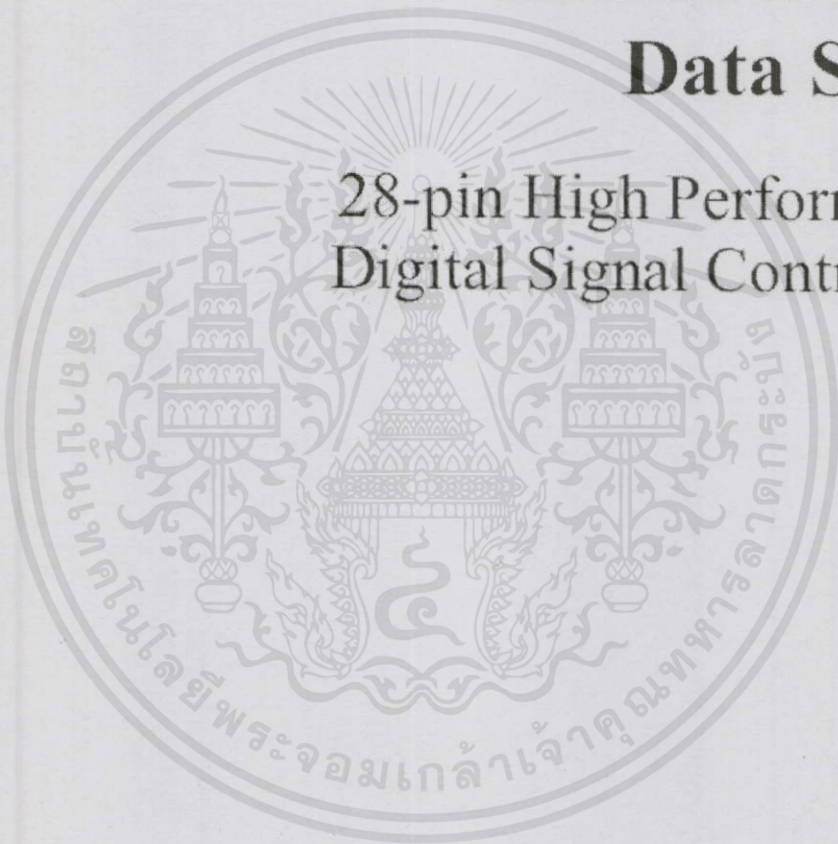
DATASHEET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



dsPIC30F2010 Data Sheet

28-pin High Performance
Digital Signal Controllers





dsPIC30F2010

28-pin dsPIC30F2010 Enhanced Flash 16-bit Digital Signal Controller

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the *dsPIC30F Family Reference Manual* (DS70046). For more information on the device instruction set and programming, refer to the *dsPIC30F Programmer's Reference Manual* (DS70030).

High Performance Modified RISC CPU:

- Modified Harvard architecture
- C compiler optimized instruction set architecture
- 84 base instructions with flexible addressing modes
- 24-bit wide instructions, 16-bit wide data path
- 12 Kbytes on-chip Flash program space
- 512 bytes on-chip data RAM
- 1 Kbyte non-volatile data EEPROM
- 16 x 16-bit working register array
- Up to 30 MIPS operation:
 - DC to 40 MHz external clock input
 - 4 MHz-10 MHz oscillator input with PLL active (4x, 8x, 16x)
- Up to 42 interrupt sources
 - 8 user selectable priority levels
 - 54 interrupt vectors
 - 8 processor exceptions and software traps

DSP Engine Features:

- Modulo and Bit-Reversed modes
- Two, 40-bit wide accumulators with optional saturation logic
- 17-bit x 17-bit single cycle hardware fractional/integer multiplier
- Single cycle Multiply-Accumulate (MAC) operation
- 40-stage Barrel Shifter
- Dual data fetch

Peripheral Features:

- High current sink/source I/O pins: 25 mA/25 mA
- Three 16-bit timers/counters; optionally pair up 16-bit timers into 32-bit timer modules
- Four 16-bit Capture input functions
- Two 16-bit Compare/PWM output functions
 - Dual Compare mode available
- 3-wire SPI™ modules (supports 4 Frame modes)
- I²C™ module supports Multi-Master/Slave mode and 7-bit/10-bit addressing
- Addressable UART modules with FIFO buffers

Motor Control PWM Module Features:

- 6 PWM output channels
 - Complementary or Independent Output modes
 - Edge and Center Aligned modes
- 4 duty cycle generators
- Dedicated time base with 4 modes
- Programmable output polarity
- Dead time control for Complementary mode
- Manual output control
- Trigger for synchronized A/D conversions

Quadrature Encoder Interface Module Features:

- Phase A, Phase B and Index Pulse input
- 16-bit up/down position counter
- Count direction status
- Position Measurement (x2 and x4) mode
- Programmable digital noise filters on inputs
- Alternate 16-bit Timer/Counter mode
- Interrupt on position counter rollover/underflow

Analog Features:

- 10-bit Analog-to-Digital Converter (A/D) with:
 - 500 Ksps (for 10-bit A/D) conversion rate
 - Six input channels
 - Conversion available during Sleep and Idle
- Programmable Brown-out Detection and Reset generation

dsPIC30F2010

Special Microcontroller Features:

- Enhanced Flash program memory:
 - 10,000 erase/write cycle (min.) for industrial temperature range, 100K (typical)
- Data EEPROM memory:
 - 100,000 erase/write cycle (min.) for industrial temperature range, 1M (typical)
- Self-reprogrammable under software control
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Flexible Watchdog Timer (WDT) with on-chip low power RC oscillator for reliable operation
- Fail-Safe clock monitor operation
- Detects clock failure and switches to on-chip low power RC oscillator
- Programmable code protection
- In-Circuit Serial Programming™ (ICSP™)
- Selectable Power Management modes
 - Sleep, Idle and Alternate Clock modes

CMOS Technology:

- Low power, high speed Flash technology
- Wide operating voltage range (2.5V to 5.5V)
- Industrial and Extended temperature ranges
- Low power consumption

dsPIC30F Motor Control and Power Conversion Family*

Device	Pins	Program Mem. Bytes/Instructions	SRAM Bytes	EEPROM Bytes	Timer 16-bit	Input Cap	Output Comp/Std PWM	Motor Control PWM	A/D 10-bit 500 Ksps	Quad Enc	UART	SPI™	I ² C™	CAN
dsPIC30F2010	28	12K/4K	512	1024	3	4	2	6 ch	6 ch	Yes	1	1	1	-
dsPIC30F3010	28	24K/8K	1024	1024	5	4	2	6 ch	6 ch	Yes	1	1	1	-
dsPIC30F4012	28	48K/16K	2048	1024	5	4	2	6 ch	6 ch	Yes	1	1	1	1
dsPIC30F3011	40/44	24K/8K	1024	1024	5	4	4	6 ch	9 ch	Yes	2	1	1	-
dsPIC30F4011	40/44	48K/16K	2048	1024	5	4	4	6 ch	9 ch	Yes	2	1	1	1
dsPIC30F5015	64	66K/22K	2048	1024	5	4	4	8 ch	16 ch	Yes	1	2	1	1
dsPIC30F6010	80	144K/48K	8192	4096	5	8	8	8 ch	16 ch	Yes	2	2	1	2

* This table provides a summary of the dsPIC30F2010 peripheral features. Other available devices in the dsPIC30F Motor Control and Power Conversion Family are shown for feature comparison.

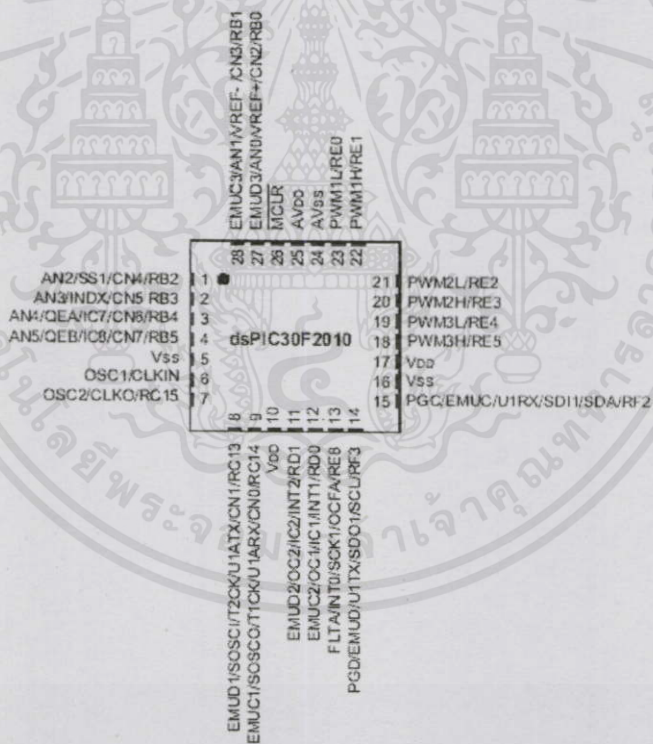
dsPIC30F2010

Pin Diagram

28-Pin SDIP and SOIC

MCLR	1	28	AVDD
EMUD3/AN1/VREF+/CN2/RB0	2	27	AVSS
EMUC3/AN1/VREF-/CN3/RB1	3	26	PWM1L/RE0
AN2/SS1/CN4/RB2	4	25	PWM1H/RE1
AN3/INDX/CN5/RB3	5	24	PWM2L/RE2
AN4/QEA1C7/CN6/RB4	6	23	PWM2H/RE3
AN5/QEB1C8/CN7/RB5	7	22	PWM3L/RE4
VSS	8	21	PWM3H/RE5
OSC1/CLKI	9	20	VDD
OSC2/CLKO/RC15	10	19	VSS
EMUD1/SOSCI/T2CK/U1ATX/CN1/RC13	11	18	PGC/EMUC/U1RX/SD11/SDA/RF2
EMUC1/SOSCO/T1CK/U1ARX/CN0/RC14	12	17	PGD/EMUD/U1TX/SD01/SCL/RF3
VDD	13	16	FLTA/INT0/SCK1/OCFA/RE8
EMUD2/OC2/IC2/INT2/RD1	14	15	EMUC2/OC1/IC1/INT1/RD0

28-Pin QFN



dsPIC30F2010

Table of Contents

1.0	Device Overview	5
2.0	CPU Architecture Overview.....	9
3.0	Memory Organization.....	19
4.0	Address Generator Units.....	31
5.0	Interrupts.....	37
6.0	Flash Program Memory.....	43
7.0	Data EEPROM Memory	49
8.0	I/O Ports	53
9.0	Timer1 Module.....	57
10.0	Timer2/3 Module.....	61
11.0	Input Capture Module.....	67
12.0	Output Compare Module.....	71
13.0	Quadrature Encoder Interface (QE1) Module	75
14.0	Motor Control PWM Module.....	81
15.0	SPI™ Module.....	91
16.0	I2C Module.....	95
17.0	Universal Asynchronous Receiver Transmitter (UART) Module	103
18.0	10-bit High Speed Analog-to-Digital Converter (A/D) Module.....	111
19.0	System Integration	119
20.0	Instruction Set Summary	133
21.0	Development Support.....	141
22.0	Electrical Characteristics.....	147
23.0	Packaging Information.....	187
	On-Line Support.....	197
	Systems Information and Upgrade Hot Line.....	197
	Reader Response	198
	Product Identification System.....	199

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at docerrors@mail.microchip.com or fax the Reader Response Form in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:
<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 792-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at www.microchip.com/cn to receive the most current information on all of our products.

dsPIC30F2010

1.0 DEVICE OVERVIEW

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the *dsPIC30F Family Reference Manual* (DS70046). For more information on the device instruction set and programming, refer to the *dsPIC30F Programmer's Reference Manual* (DS70030).

This document contains device specific information for the dsPIC30F2010 device. The dsPIC30F devices contain extensive Digital Signal Processor (DSP) functionality within a high performance 16-bit microcontroller (MCU) architecture. Figure 1-1 shows a device block diagram for the dsPIC30F2010 device.



dsPIC30F2010

Table 1-1 provides a brief description of device I/O pinouts and the functions that may be multiplexed to a port pin. Multiple functions may exist on one port pin.

When multiplexing occurs, the peripheral module's functional requirements may force an override of the data direction of the port pin.

TABLE 1-1: PINOUT I/O DESCRIPTIONS

Pin Name	Pin Type	Buffer Type	Description
AN0-AN5	I	Analog	Analog input channels.
AVDD	P	P	Positive supply for analog module.
AVSS	P	P	Ground reference for analog module.
CLKI	I	ST/CMOS	External clock source input. Always associated with OSC1 pin function.
CLKO	O	—	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes. Always associated with OSC2 pin function.
CN0-CN7	I	ST	Input change notification inputs. Can be software programmed for internal weak pull-ups on all inputs.
EMUD	I/O	ST	ICD Primary Communication Channel data input/output pin.
EMUC	I/O	ST	ICD Primary Communication Channel clock input/output pin.
EMUD1	I/O	ST	ICD Secondary Communication Channel data input/output pin.
EMUC1	I/O	ST	ICD Secondary Communication Channel clock input/output pin.
EMUD2	I/O	ST	ICD Tertiary Communication Channel data input/output pin.
EMUC2	I/O	ST	ICD Tertiary Communication Channel clock input/output pin.
EMUD3	I/O	ST	ICD Quaternary Communication Channel data input/output pin.
EMUC3	I/O	ST	ICD Quaternary Communication Channel clock input/output pin.
IC1, IC2, IC7, IC8	I	ST	Capture inputs. The dsPIC30F2010 has 4 capture inputs. The inputs are numbered for consistency with the inputs on larger device variants.
INDX	I	ST	Quadrature Encoder Index Pulse input.
QEA	I	ST	Quadrature Encoder Phase A input in QE1 mode.
QEB	I	ST	Quadrature Encoder Phase A input in QE1 mode. Auxiliary Timer External Clock/Gate input in Timer mode.
INT0	I	ST	External interrupt 0
INT1	I	ST	External interrupt 1
INT2	I	ST	External interrupt 2
FLTA	I	ST	PWM Fault A input
PWM1L	O	—	PWM 1 Low output
PWM1H	O	—	PWM 1 High output
PWM2L	O	—	PWM 2 Low output
PWM2H	O	—	PWM 2 High output
PWM3L	O	—	PWM 3 Low output
PWM3H	O	—	PWM 3 High output
MCLR	I/P	ST	Master Clear (Reset) input or programming voltage input. This pin is an active low Reset to the device.
OCFA	I	ST	Compare Fault A input (for Compare channels 1, 2, 3 and 4).
OC1-OC2	O	—	Compare outputs.
OSC1	I	ST/CMOS	Oscillator crystal input. ST buffer when configured in RC mode; CMOS otherwise.
OSC2	I/O	—	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes.
PGD	I/O	ST	In-Circuit Serial Programming data input/output pin.
PGC	I	ST	In-Circuit Serial Programming clock input pin.

Legend: CMOS = CMOS compatible input or output Analog = Analog input
 ST = Schmitt Trigger input with CMOS levels O = Output
 I = Input P = Power

dsPIC30F2010

TABLE 1-1: PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Type	Buffer Type	Description
RB0-RB5	I/O	ST	PORTB is a bidirectional I/O port.
RC13-RC14	I/O	ST	PORTC is a bidirectional I/O port.
RD0-RD1	I/O	ST	PORTD is a bidirectional I/O port.
RE0-RE5, RE8	I/O	ST	PORTE is a bidirectional I/O port.
RF2, RF3	I/O	ST	PORTF is a bidirectional I/O port.
SCK1	I/O	ST	Synchronous serial clock input/output for SPI1.
SDI1	I	ST	SPI1 Data In.
SDO1	O	—	SPI1 Data Out.
SS1	I	ST	SPI1 Slave Synchronization.
SCL	I/O	ST	Synchronous serial clock input/output for I ² C.
SDA	I/O	ST	Synchronous serial data input/output for I ² C.
SOSCO	O	—	32 kHz low power oscillator crystal output.
SOSCI	I	ST/CMOS	32 kHz low power oscillator crystal input. ST buffer when configured in RC mode; CMOS otherwise.
T1CK	I	ST	Timer1 external clock input.
T2CK	I	ST	Timer2 external clock input.
U1RX	I	ST	UART1 Receive.
U1TX	O	—	UART1 Transmit.
U1ARX	I	ST	UART1 Alternate Receive.
U1ATX	O	—	UART1 Alternate Transmit.
VDD	P	—	Positive supply for logic and I/O pins.
VSS	P	—	Ground reference for logic and I/O pins.
VREF+	I	Analog	Analog Voltage Reference (High) input.
VREF-	I	Analog	Analog Voltage Reference (Low) input.

Legend: CMOS = CMOS compatible input or output Analog = Analog input
 ST = Schmitt Trigger input with CMOS levels O = Output
 I = Input P = Power

2.0 CPU ARCHITECTURE OVERVIEW

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the *dsPIC30F Family Reference Manual* (DS70046). For more information on the device instruction set and programming, refer to the *dsPIC30F Programmer's Reference Manual* (DS70030).

This document provides a summary of the dsPIC30F2010 CPU and peripheral function. For a complete description of this functionality, please refer to the *dsPIC30F Family Reference Manual* (DS70046).

2.1 Core Overview

The core has a 24-bit instruction word. The Program Counter (PC) is 23 bits wide with the Least Significant (LS) bit always clear (see Section 3.1), and the Most Significant (MS) bit is ignored during normal program execution, except for certain specialized instructions. Thus, the PC can address up to 4M instruction words of user program space. An instruction pre-fetch mechanism is used to help maintain throughput. Program loop constructs, free from loop count management overhead, are supported using the *DO* and *REPEAT* instructions, both of which are interruptible at any point.

The working register array consists of 16x16-bit registers, each of which can act as data, address or offset registers. One working register (W15) operates as a software stack pointer for interrupts and calls.

The data space is 64 Kbytes (32K words) and is split into two blocks, referred to as X and Y data memory. Each block has its own independent Address Generation Unit (AGU). Most instructions operate solely through the X memory AGU, which provides the appearance of a single unified data space. The Multiply-Accumulate (MAC) class of dual source DSP instructions operate through both the X and Y AGUs, splitting the data address space into two parts (see Section 3.2). The X and Y data space boundary is device specific and cannot be altered by the user. Each data word consists of 2 bytes, and most instructions can address data either as words or bytes.

There are two methods of accessing data stored in program memory:

- The upper 32 Kbytes of data space memory can be mapped into the lower half (user space) of program space at any 16K program word boundary, defined by the 8-bit Program Space Visibility Page (PSVPAG) register. This lets any instruction access program space as if it were data space, with a limitation that the access requires an additional cycle. Moreover, only the lower 16 bits of each instruction word can be accessed using this method.

- Linear indirect access of 32K word pages within program space is also possible using any working register, via table read and write instructions. Table read and write instructions can be used to access all 24 bits of an instruction word.

Overhead-free circular buffers (modulo addressing) are supported in both X and Y address spaces. This is primarily intended to remove the loop overhead for DSP algorithms.

The X AGU also supports bit-reversed addressing on destination effective addresses, to greatly simplify input or output data reordering for radix-2 FFT algorithms. Refer to Section 4.0 for details on modulo and bit-reversed addressing.

The core supports Inherent (no operand), Relative, Literal, Memory Direct, Register Direct, Register Indirect, Register Offset and Literal Offset Addressing modes. Instructions are associated with predefined Addressing modes, depending upon their functional requirements.

For most instructions, the core is capable of executing a data (or program data) memory read, a working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle. As a result, 3-operand instructions are supported, allowing $C = A+B$ operations to be executed in a single cycle.

A DSP engine has been included to significantly enhance the core arithmetic capability and throughput. It features a high speed 17-bit by 17-bit multiplier, a 40-bit ALU, two 40-bit saturating accumulators and a 40-bit bidirectional barrel shifter. Data in the accumulator or any working register can be shifted up to 15 bits right or 16 bits left in a single cycle. The DSP instructions operate seamlessly with all other instructions and have been designed for optimal real-time performance. The MAC class of instructions can concurrently fetch two data operands from memory, while multiplying two W registers. To enable this concurrent fetching of data operands, the data space has been split for these instructions and linear for all others. This has been achieved in a transparent and flexible manner, by dedicating certain working registers to each address space for the MAC class of instructions.

The core does not support a multi-stage instruction pipeline. However, a single stage instruction pre-fetch mechanism is used, which accesses and partially decodes instructions a cycle ahead of execution, in order to maximize available execution time. Most instructions execute in a single cycle, with certain exceptions.

The core features a vectored exception processing structure for traps and interrupts, with 62 independent vectors. The exceptions consist of up to 8 traps (of which 4 are reserved) and 54 interrupts. Each interrupt is prioritized based on a user assigned priority between 1 and 7 (1 being the lowest priority and 7 being the highest) in conjunction with a predetermined 'natural order'. Traps have fixed priorities, ranging from 8 to 15.

dsPIC30F2010

2.2 Programmer's Model

The programmer's model is shown in Figure 2-1 and consists of 16x16-bit working registers (W0 through W15), 2x40-bit accumulators (AccA and AccB), STATUS register (SR), Data Table Page register (TBLPAG), Program Space Visibility Page register (PSVPAG), DO and REPEAT registers (DOSTART, DOEND, DCOUNT and RCOUNT), and Program Counter (PC). The working registers can act as data, address or offset registers. All registers are memory mapped. W0 acts as the W register for file register addressing.

Some of these registers have a shadow register associated with each of them, as shown in Figure 2-1. The shadow register is used as a temporary holding register and can transfer its contents to or from its host register upon the occurrence of an event. None of the shadow registers are accessible directly. The following rules apply for transfer of registers into and out of shadows.

- PUSH.S and POP.S
W0, W1, W2, W3, SR (DC, N, OV, Z and C bits only) are transferred.
- DO instruction
DOSTART, DOEND, DCOUNT shadows are pushed on loop start, and popped on loop end.

When a byte operation is performed on a working register, only the Least Significant Byte of the target register is affected. However, a benefit of memory mapped working registers is that both the Least and Most Significant Bytes can be manipulated through byte wide data memory space accesses.

2.2.1 SOFTWARE STACK POINTER/ FRAME POINTER

The dsPIC[®] devices contain a software stack. W15 is the dedicated software stack pointer (SP), and will be automatically modified by exception processing and subroutine calls and returns. However, W15 can be referenced by any instruction in the same manner as all other W registers. This simplifies the reading, writing and manipulation of the stack pointer (e.g., creating stack frames).

Note: In order to protect against misaligned stack accesses, W15<0> is always clear.

W15 is initialized to 0x0800 during a Reset. The user may reprogram the SP during initialization to any location within data space.

W14 has been dedicated as a stack frame pointer as defined by the LNK and ULNK instructions. However, W14 can be referenced by any instruction in the same manner as all other W registers.

2.2.2 STATUS REGISTER

The dsPIC core has a 16-bit status register (SR), the LS Byte of which is referred to as the SR Low Byte (SRL) and the MS Byte as the SR High Byte (SRH). See Figure 2-1 for SR layout.

SRL contains all the MCU ALU operation status flags (including the Z bit), as well as the CPU Interrupt Priority Level status bits, IPL<2:0>, and the REPEAT active status bit, RA. During exception processing, SRL is concatenated with the MS Byte of the PC to form a complete word value which is then stacked.

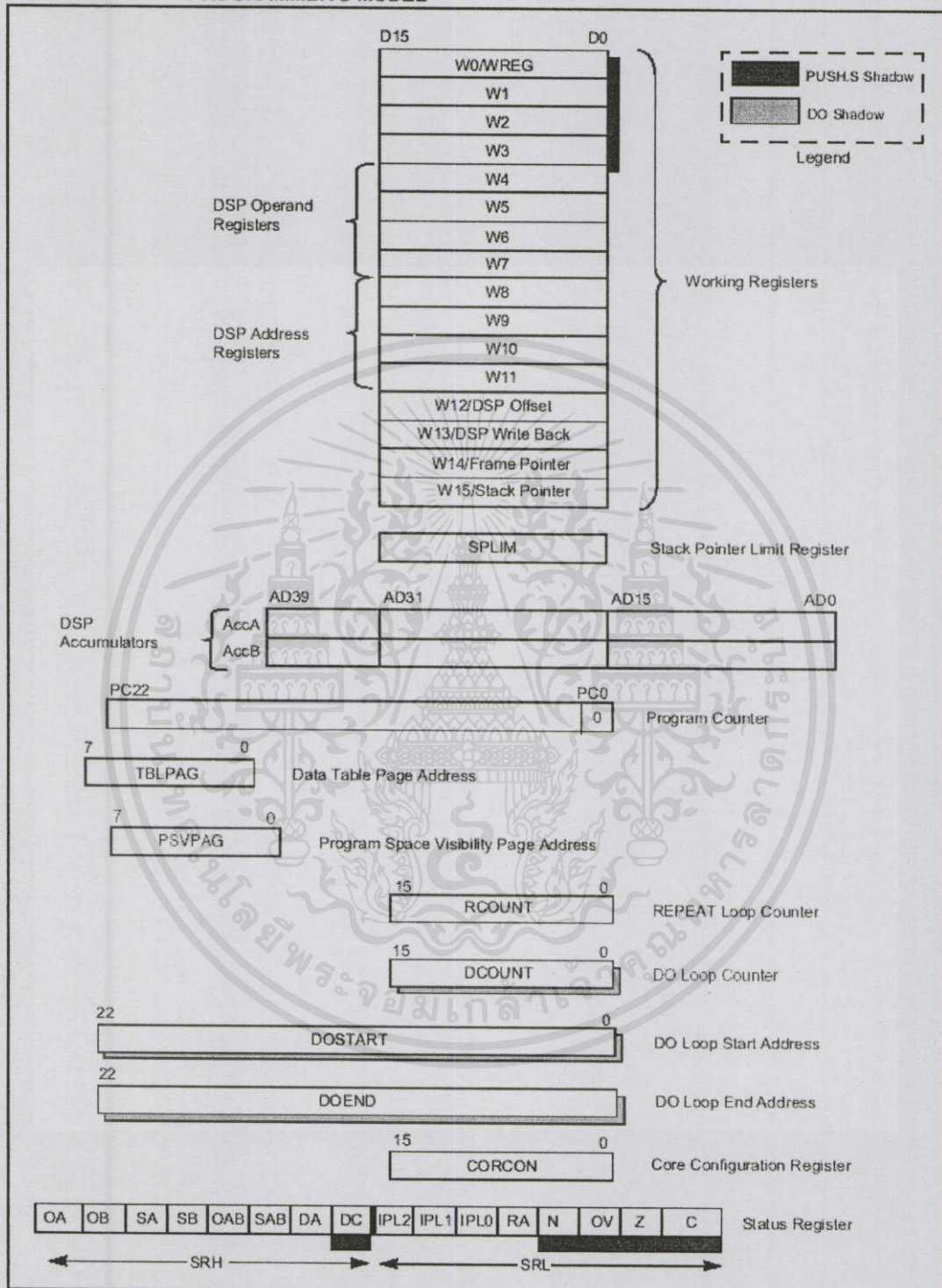
The upper byte of the STATUS register contains the DSP Adder/Subtractor status bits, the DO Loop Active bit (DA) and the Digit Carry (DC) status bit.

2.2.3 PROGRAM COUNTER

The Program Counter is 23 bits wide. Bit 0 is always clear. Therefore, the PC can address up to 4M instruction words.

dsPIC30F2010

FIGURE 2-1: PROGRAMMER'S MODEL



dsPIC30F2010

2.3 Divide Support

The dsPIC devices feature a 16/16-bit signed fractional divide operation, as well as 32/16-bit and 16/16-bit signed and unsigned integer divide operations, in the form of single instruction iterative divides. The following instructions and data sizes are supported:

1. DIVF – 16/16 signed fractional divide
2. DIV.sd – 32/16 signed divide
3. DIV.ud – 32/16 unsigned divide
4. DIV.sw – 16/16 signed divide
5. DIV.uw – 16/16 unsigned divide

The 16/16 divides are similar to the 32/16 (same number of iterations), but the dividend is either zero-extended or sign-extended during the first iteration.

The divide instructions must be executed within a REPEAT loop. Any other form of execution (e.g. a series of discrete divide instructions) will not function correctly because the instruction flow depends on RCOUNT. The divide instruction does not automatically set up the RCOUNT value, and it must, therefore, be explicitly and correctly specified in the REPEAT instruction, as shown in Table 2-1 (REPEAT will execute the target instruction {operand value+1} times). The REPEAT loop count must be set up for 18 iterations of the DIV/DIVF instruction. Thus, a complete divide operation requires 19 cycles.

Note: The Divide flow is interruptible. However, the user needs to save the context as appropriate.

TABLE 2-1: DIVIDE INSTRUCTIONS

Instruction	Function
DIVF	Signed fractional divide: $Wm/Wn \rightarrow W0$; Rem $\rightarrow W1$
DIV.sd	Signed divide: $(Wm+1:Wm)/Wn \rightarrow W0$; Rem $\rightarrow W1$
DIV.sw (or DIV.s)	Signed divide: $Wm/Wn \rightarrow W0$; Rem $\rightarrow W1$
DIV.ud	Unsigned divide: $(Wm+1:Wm)/Wn \rightarrow W0$; Rem $\rightarrow W1$
DIV.uw (or DIV.u)	Unsigned divide: $Wm/Wn \rightarrow W0$; Rem $\rightarrow W1$

dsPIC30F2010

2.4 DSP Engine

The DSP engine consists of a high speed 17-bit x 17-bit multiplier, a barrel shifter, and a 40-bit adder/subtractor (with two target accumulators, round and saturation logic).

The DSP engine also has the capability to perform inherent accumulator-to-accumulator operations, which require no additional data. These instructions are ADD, SUB and NEG.

The DSP engine has various options selected through various bits in the CPU Core Configuration Register (CORCON), as listed below:

1. Fractional or integer DSP multiply (IF).
2. Signed or unsigned DSP multiply (US).
3. Conventional or convergent rounding (RND).
4. Automatic saturation on/off for AccA (SATA).
5. Automatic saturation on/off for AccB (SATB).
6. Automatic saturation on/off for writes to data memory (SATDW).
7. Accumulator Saturation mode selection (ACCSAT).

Note: For CORCON layout, see Table 4-2.

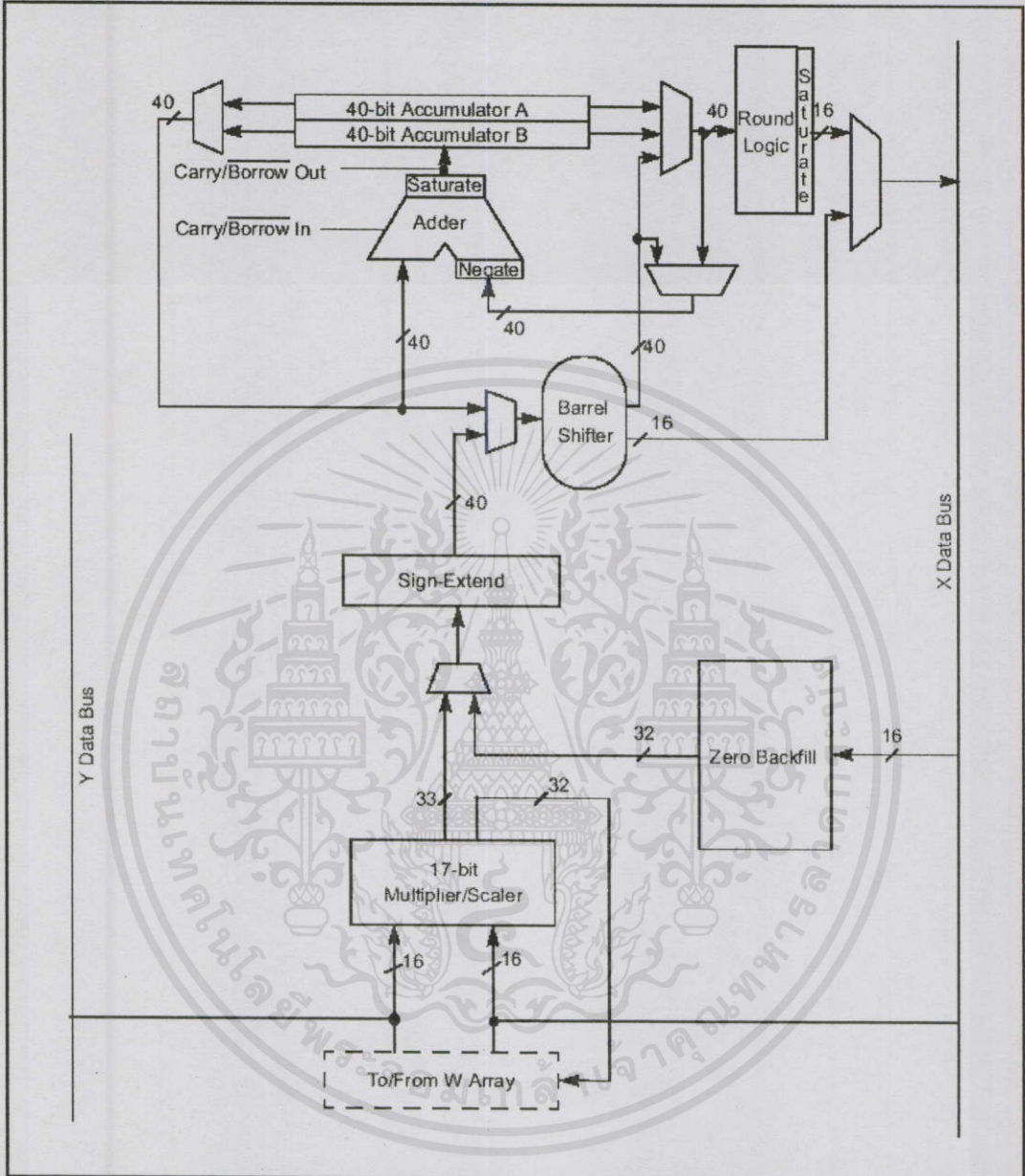
A block diagram of the DSP engine is shown in Figure 2-2.

TABLE 2-2: DSP INSTRUCTION SUMMARY

Instruction	Algebraic Operation	ACC WB?
CLR	$A = 0$	Yes
ED	$A = (x - y)^2$	No
EDAC	$A = A + (x - y)^2$	No
MAC	$A = A + (x * y)$	Yes
MAC	$A = A + x^2$	No
MOVSAC	No change in A	Yes
MPY	$A = x * y$	No
MPY.N	$A = -x * y$	No
MSC	$A = A - x * y$	Yes

dsPIC30F2010

FIGURE 2-2: DSP ENGINE BLOCK DIAGRAM



dsPIC30F2010

2.4.1 MULTIPLIER

The 17x17-bit multiplier is capable of signed or unsigned operation and can multiplex its output using a scaler to support either 1.31 fractional (Q31) or 32-bit integer results. Unsigned operands are zero-extended into the 17th bit of the multiplier input value. Signed operands are sign-extended into the 17th bit of the multiplier input value. The output of the 17x17-bit multiplier/scaler is a 33-bit value, which is sign-extended to 40 bits. Integer data is inherently represented as a signed two's complement value, where the MSB is defined as a sign bit. Generally speaking, the range of an N-bit two's complement integer is -2^{N-1} to $2^{N-1} - 1$. For a 16-bit integer, the data range is -32768 (0x8000) to 32767 (0x7FFF), including 0. For a 32-bit integer, the data range is -2,147,483,648 (0x8000 0000) to 2,147,483,645 (0x7FFF FFFF).

When the multiplier is configured for fractional multiplication, the data is represented as a two's complement fraction, where the MSB is defined as a sign bit and the radix point is implied to lie just after the sign bit (QX format). The range of an N-bit two's complement fraction with this implied radix point is -1.0 to $(1-2^{-N})$. For a 16-bit fraction, the Q15 data range is -1.0 (0x8000) to 0.999969482 (0x7FFF), including 0 and has a precision of 3.01518×10^{-5} . In Fractional mode, a 16x16 multiply operation generates a 1.31 product, which has a precision of 4.65661×10^{-10} .

The same multiplier is used to support the MCU multiply instructions, which include integer 16-bit signed, unsigned and mixed sign multiplies.

The MUL instruction may be directed to use byte or word sized operands. Byte operands will direct a 16-bit result, and word operands will direct a 32-bit result to the specified register(s) in the W array.

2.4.2 DATA ACCUMULATORS AND ADDER/SUBTRACTOR

The data accumulator consists of a 40-bit adder/subtractor with automatic sign extension logic. It can select one of two accumulators (A or B) as its pre-accumulation source and post-accumulation destination. For the ADD and LAC instructions, the data to be accumulated or loaded can be optionally scaled via the barrel shifter, prior to accumulation.

2.4.2.1 Adder/Subtractor, Overflow and Saturation

The adder/subtractor is a 40-bit adder with an optional zero input into one side and either true or complement data into the other input. In the case of addition, the carry/borrow input is active high and the other input is true data (not complemented), whereas in the case of subtraction, the carry/borrow input is active low and the other input is complemented. The adder/subtractor generates overflow status bits SA/SB and OA/OB, which are latched and reflected in the status register.

- Overflow from bit 39: this is a catastrophic overflow in which the sign of the accumulator is destroyed.
- Overflow into guard bits 32 through 39: this is a recoverable overflow. This bit is set whenever all the guard bits are not identical to each other.

The adder has an additional saturation block which controls accumulator data saturation, if selected. It uses the result of the adder, the overflow status bits described above, and the SAT/VB (CORCON<7:6>) and ACCSAT (CORCON<4>) mode control bits to determine when and to what value to saturate.

Six status register bits have been provided to support saturation and overflow; they are:

1. OA:
AccA overflowed into guard bits
2. OB:
AccB overflowed into guard bits
3. SA:
AccA saturated (bit 31 overflow and saturation)
or
AccA overflowed into guard bits and saturated (bit 39 overflow and saturation)
4. SB:
AccB saturated (bit 31 overflow and saturation)
or
AccB overflowed into guard bits and saturated (bit 39 overflow and saturation)
5. OAB:
Logical OR of OA and OB
6. SAB:
Logical OR of SA and SB

The OA and OB bits are modified each time data passes through the adder/subtractor. When set, they indicate that the most recent operation has overflowed into the accumulator guard bits (bits 32 through 39). The OA and OB bits can also optionally generate an arithmetic warning trap when set and the corresponding overflow trap flag enable bit (OVATEN, OVBTEN) in the INTCON1 register (refer to Section 5.0) is set. This allows the user to take immediate action, for example, to correct system gain.

dsPIC30F2010

The SA and SB bits are modified each time data passes through the adder/subtractor, but can only be cleared by the user. When set, they indicate that the accumulator has overflowed its maximum range (bit 31 for 32-bit saturation, or bit 39 for 40-bit saturation) and will be saturated (if saturation is enabled). When saturation is not enabled, SA and SB default to bit 39 overflow and thus indicate that a catastrophic overflow has occurred. If the COVTE bit in the INTCON1 register is set, SA and SB bits will generate an arithmetic warning trap when saturation is disabled.

The overflow and saturation status bits can optionally be viewed in the Status Register (SR) as the logical OR of OA and OB (in bit OAB) and the logical OR of SA and SB (in bit SAB). This allows programmers to check one bit in the Status Register to determine if either accumulator has overflowed, or one bit to determine if either accumulator has saturated. This would be useful for complex number arithmetic which typically uses both the accumulators.

The device supports three Saturation and Overflow modes.

1. **Bit 39 Overflow and Saturation:**
When bit 39 overflow and saturation occurs, the saturation logic loads the maximally positive 9.31 (0x7FFFFFFF) or maximally negative 9.31 value (0x80000000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user. This is referred to as 'super saturation' and provides protection against erroneous data or unexpected algorithm problems (e.g., gain calculations).
2. **Bit 31 Overflow and Saturation:**
When bit 31 overflow and saturation occurs, the saturation logic then loads the maximally positive 1.31 value (0x007FFFFFFF) or maximally negative 1.31 value (0x00800000) into the target accumulator. The SA or SB bit is set and remains set until cleared by the user. When this Saturation mode is in effect, the guard bits are not used (so the OA, OB or OAB bits are never set).
3. **Bit 39 Catastrophic Overflow**
The bit 39 overflow status bit from the adder is used to set the SA or SB bit, which remain set until cleared by the user. No saturation operation is performed and the accumulator is allowed to overflow (destroying its sign). If the COVTE bit in the INTCON1 register is set, a catastrophic overflow can initiate a trap exception.

2.4.2.2 Accumulator 'Write Back'

The MAC class of instructions (with the exception of MPY, MPY.N, ED and EDAC) can optionally write a rounded version of the high word (bits 31 through 16) of the accumulator that is not targeted by the instruction into data space memory. The write is performed across the X bus into combined X and Y address space. The following addressing modes are supported:

1. **W13, Register Direct**
The rounded contents of the non-target accumulator are written into W13 as a 1.15 fraction.
2. **[W13]+2, Register Indirect with Post-Increment:**
The rounded contents of the non-target accumulator are written into the address pointed to by W13 as a 1.15 fraction. W13 is then incremented by 2 (for a word write).

2.4.2.3 Round Logic

The round logic is a combinational block, which performs a conventional (biased) or convergent (unbiased) round function during an accumulator write (store). The Round mode is determined by the state of the RND bit in the CORCON register. It generates a 16-bit, 1.15 data value which is passed to the data space write saturation logic. If rounding is not indicated by the instruction, a truncated 1.15 data value is stored and the LS Word is simply discarded.

Conventional rounding takes bit 15 of the accumulator, zero-extends it and adds it to the ACCxH word (bits 16 through 31 of the accumulator). If the ACCxL word (bits 0 through 15 of the accumulator) is between 0x8000 and 0xFFFF (0x8000 included), ACCxH is incremented. If ACCxL is between 0x0000 and 0x7FFF, ACCxH is left unchanged. A consequence of this algorithm is that over a succession of random rounding operations, the value will tend to be biased slightly positive.

Convergent (or unbiased) rounding operates in the same manner as conventional rounding, except when ACCxL equals 0x8000. If this is the case, the LS bit (bit 16 of the accumulator) of ACCxH is examined. If it is '1', ACCxH is incremented. If it is '0', ACCxH is not modified. Assuming that bit 16 is effectively random in nature, this scheme will remove any rounding bias that may accumulate.

The SAC and SAC.R instructions store either a truncated (SAC) or rounded (SAC.R) version of the contents of the target accumulator to data memory, via the X bus (subject to data saturation, see Section 2.4.2.4). Note that for the MAC class of instructions, the accumulator write back operation will function in the same manner, addressing combined MCU (X and Y) data space through the X bus. For this class of instructions, the data is always subject to rounding.

dsPIC30F2010

2.4.2.4 Data Space Write Saturation

In addition to adder/subtractor saturation, writes to data space may also be saturated, but without affecting the contents of the source accumulator. The data space write saturation logic block accepts a 16-bit, 1.15 fractional value from the round logic block as its input, together with overflow status from the original source (accumulator) and the 16-bit round adder. These are combined and used to select the appropriate 1.15 fractional value as output to write to data space memory.

If the SATDW bit in the CORCON register is set, data (after rounding or truncation) is tested for overflow and adjusted accordingly. For input data greater than 0x007FFF, data written to memory is forced to the maximum positive 1.15 value, 0x7FFF. For input data less than 0xFF8000, data written to memory is forced to the maximum negative 1.15 value, 0x8000. The MS bit of the source (bit 39) is used to determine the sign of the operand being tested.

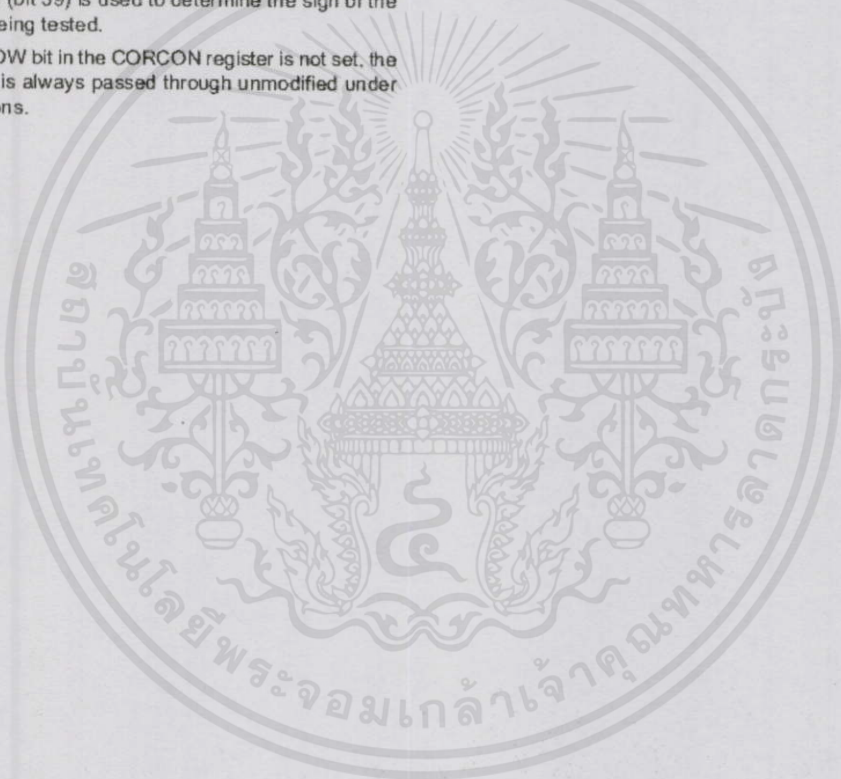
If the SATDW bit in the CORCON register is not set, the input data is always passed through unmodified under all conditions.

2.4.3 BARREL SHIFTER

The barrel shifter is capable of performing up to 15-bit arithmetic or logic right shifts, or up to 16-bit left shifts in a single cycle. The source can be either of the two DSP accumulators or the X bus (to support multi-bit shifts of register or memory data).

The shifter requires a signed binary value to determine both the magnitude (number of bits) and direction of the shift operation. A positive value will shift the operand right. A negative value will shift the operand left. A value of 0 will not modify the operand.

The barrel shifter is 40 bits wide, thereby obtaining a 40-bit result for DSP shift operations and a 16-bit result for MCU shift operations. Data from the X bus is presented to the barrel shifter between bit positions 16 to 31 for right shifts, and bit positions 0 to 15 for left shifts.





MPLAB[®] IDE Quick Start Guide



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, Accuron, dsPIC, KeeLoq, microID, MPLAB, PIC, PICmicro, PICSTART, PRO MATE, PowerSmart, rfPIC and SmartShunt are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.


AmpLab, FillerLab, Migratable Memory, MXDEV, MXLAB, SEEVAL, SmartSensor and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, dsPICDEM, dsPICDEM.net, dsPICworks, ECAN, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, Linear Active Thermistor, Mindi, MIWI, MPASM, MPLIB, MPLINK, PICkit, PICDEM, PICDEM.net, PICLAB, PICtail, PowerCal, PowerInfo, PowerMate, PowerTool, REAL ICE, rLAB, rfPICDEM, Select Mode, Smart Serial, SmartTel, Total Endurance, UNI/O, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2006, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

**QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==**

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona, Gresham, Oregon and Mountain View, California. The Company's quality system processes and procedures are for its PICmicro® 8-bit MCUs, KeeLoq® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Table of Contents

Chapter 1. What is MPLAB® IDE?	
1.1 An Overview of Embedded Systems	1
1.2 The Development Cycle	6
1.3 Project Manager	7
1.4 Language Tools	8
1.5 Target Debugging	9
1.6 Device Programming	10
1.7 Components of MPLAB IDE	10
1.8 MPLAB IDE Documentation	11
1.9 MPLAB IDE On-line Help	11
1.10 MPLAB IDE Updates and Version Numbering	14
Chapter 2. A Basic Tutorial for MPLAB IDE	
2.1 Introduction	15
2.2 MPLAB IDE Features and Installation	16
2.3 Tutorial Overview	18
2.4 Selecting the Device	19
2.5 Creating the Project	20
2.6 Setting Up Language Tools	21
2.7 Naming the Project	22
2.8 Adding Files to the Project	23
2.9 Building the Project	26
2.10 Creating Code	27
2.11 Building the Project Again	29
2.12 Testing Code with the Simulator	30
2.13 Tutorial Summary	37
Worldwide Sales and Service	38

MPLAB® IDE Quick Start Guide

NOTES:





Chapter 1. What is MPLAB® IDE?

1.1 AN OVERVIEW OF EMBEDDED SYSTEMS

MPLAB IDE is a software program that runs on a PC to develop applications for Microchip microcontrollers. It is called an Integrated Development Environment, or IDE, because it provides a single integrated "environment" to develop code for embedded microcontrollers. Experienced embedded systems designers may want to skip ahead to **Section 1.7 "Components of MPLAB IDE"**. It is also recommended that **Section 1.9 "MPLAB IDE On-line Help"** and **Section 1.10 "MPLAB IDE Updates and Version Numbering"** be reviewed. The rest of this chapter briefly explains embedded systems development and how MPLAB IDE is used.

1.1.1 Description of an "Embedded System"

An embedded system is typically a design making use of the power of a small microcontroller, like the Microchip PICmicro® MCU or dsPIC® Digital Signal Controller (DSCs). These microcontrollers combine a microprocessor unit (like the CPU in a desktop PC) with some additional circuits called "peripherals", plus some additional circuits on the same chip to make a small control module requiring few other external devices. This single device can then be embedded into other electronic and mechanical devices for low-cost digital control.

1.1.2 Differences Between an Embedded Controller and a PC

The main difference between an embedded controller and a PC is that the embedded controller is dedicated to one specific task or set of tasks. A PC is designed to run many different types of programs and to connect to many different external devices. An embedded controller has a single program and, as a result, can be made cheaply to include just enough computing power and hardware to perform that dedicated task. A PC has a relatively expensive generalized central processing unit (CPU) at its heart with many other external devices (memory, disk drives, video controllers, network interface circuits, etc.). An embedded system has a low-cost microcontroller unit (MCU) for its intelligence, with many peripheral circuits on the same chip, and with relatively few external devices. Often, an embedded system is an invisible part, or sub-module of another product, such as a cordless drill, refrigerator or garage door opener. The controller in these products does a tiny portion of the function of the whole device. The controller adds low-cost intelligence to some of the critical sub-systems in these devices.

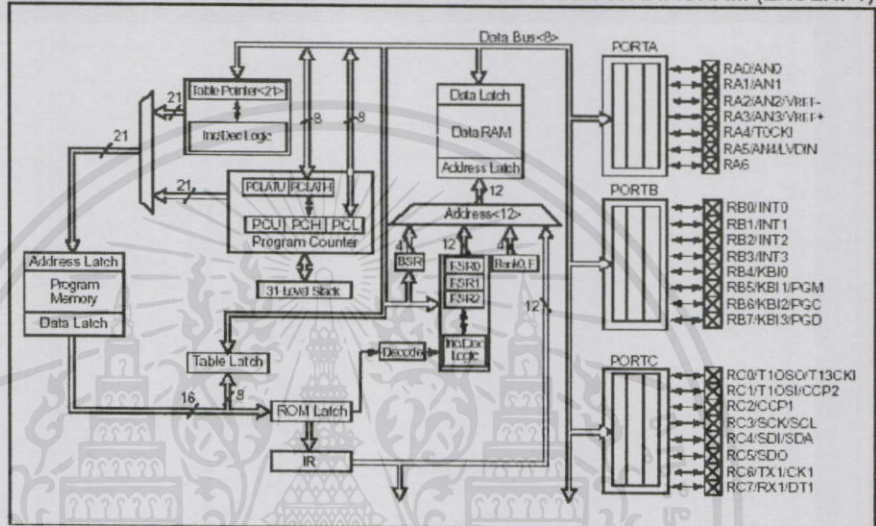
An example of an embedded system is a smoke detector. Its function is to evaluate signals from a sensor and sound an alarm if the signals indicate the presence of smoke. A small program in the smoke detector either runs in an infinite loop, sampling the signal from the smoke sensor, or lies dormant in a low-power "sleep" mode, being awakened by a signal from the sensor. The program then sounds the alarm. The program would possibly have a few other functions, such as a user test function, and a low battery alert. While a PC with a sensor and audio output could be programmed to do the same function, it would not be a cost-effective solution (nor would it run on a nine-volt battery, unattended for years!). Embedded designs use inexpensive microcontrollers to put intelligence into the everyday things in our environment, such as smoke detectors, cameras, cell phones, appliances, automobiles, smart cards and security systems.

MPLAB® IDE Quick Start Guide

1.1.3 Components of a Microcontroller

The PICmicro MCU has program memory for the firmware, or coded instructions, to run a program. It also has "file register" memory for storage of variables that the program will need for computation or temporary storage. It also has a number of peripheral device circuits on the same chip. Some peripheral devices are called I/O ports. I/O ports are pins on the microcontroller that can be driven high or low to send signals, blink lights, drive speakers – just about anything that can be sent through a wire. Often these pins are bidirectional and can also be configured as inputs allowing the program to respond to an external switch, sensor or to communicate with some external device.

FIGURE 1-1: PICmicro® MCU DATA SHEET – BLOCK DIAGRAM (EXCERPT)

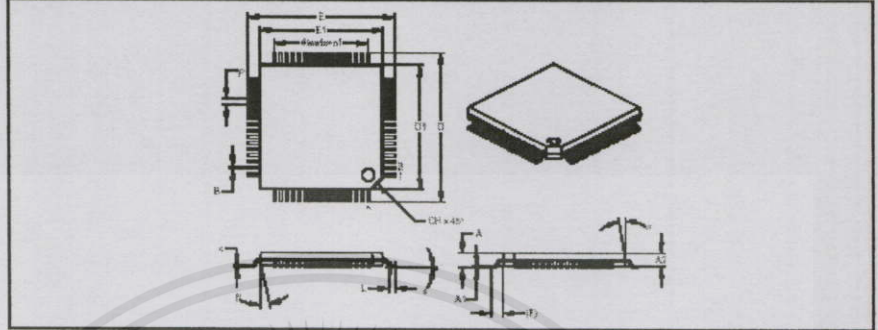


In order to design such a system, it must be decided which peripherals are needed for an application. Analog-to-Digital Converters (ADCs) allow microcontrollers to connect to sensors and receive changing voltage levels. Serial communication peripherals allow you to stream communications over a few wires to another microcontroller, to a local network or to the internet. Peripherals on the PICmicro MCU called "timers" accurately measure signal events and generate and capture communications signals, produce precise waveforms, even automatically reset the microcontroller if it gets "hung" or lost due to a power glitch or hardware malfunction. Other peripherals detect if the external power is dipping below dangerous levels so the microcontroller can store critical information and safely shut down before power is completely lost.

What is MPLAB® IDE?

The peripherals and the amount of memory an application needs to run a program largely determines which PICmicro MCU to use. Other factors might include the power consumed by the microcontroller and its "form factor," i.e., the size and characteristics of the physical package that must reside on the target design.

FIGURE 1-2: Example PICmicro® MCU DEVICE PACKAGE



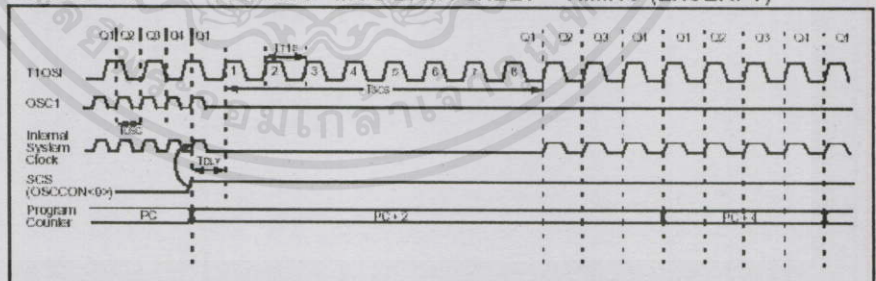
1.1.4 Implementing an Embedded System Design with MPLAB IDE

A development system for embedded controllers is a system of programs running on a desktop PC to help write, edit, debug and program code – the intelligence of embedded systems applications – into a microcontroller. MPLAB IDE runs on a PC and contains all the components needed to design and deploy embedded systems applications.

The typical tasks for developing an embedded controller application are:

1. Create the high level design. From the features and performance desired, decide which PICmicro MCU or dsPIC DSC device is best suited to the application, then design the associated hardware circuitry. After determining which peripherals and pins control the hardware, write the firmware – the software that will control the hardware aspects of the embedded application. A language tool such as an assembler, which is directly translatable into machine code, or a compiler that allows a more natural language for creating programs, should be used to write and edit code. Assemblers and compilers help make the code understandable, allowing function labels to identify code routines with variables that have names associated with their use, and with constructs that help organize the code in a maintainable structure.

FIGURE 1-3: PICmicro® MCU DATA SHEET – TIMING (EXCERPT)



MPLAB® IDE Quick Start Guide

FIGURE 1-4: PICmicro® MCU DATA SHEET – INSTRUCTIONS (EXCERPT)

RRNCF Rotate Right (no carry)									
Syntax:	[label] RRNCF #d [d]								
Operands:	0 <#< 255 0 <#< 8 0 <#< 8								
Description:	Rotate #d bits to the right by #d bits. 0 <#< 255 0 <#< 8								
Status Affected:	Z, C								
Encoding:	1111 1000 1111 1111								
Decompile:	The rotate right of register #d is performed one bit at a time. If #d is 1, the result is placed in W. If #d is 2, the result is placed in W. If #d is 3, the result is placed back in register #d (provided #d is 0, the address bank will be selected containing the bit). If #d is 1, then the bank will be selected for the DCH value available.								
Words:	1								
Cycles:	1								
C-Cycle Activity:									
	<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> <th>C3</th> <th>C4</th> </tr> </thead> <tbody> <tr> <td>Decode</td> <td>Read register #d</td> <td>Rotate</td> <td>Write to destination</td> </tr> </tbody> </table>	C1	C2	C3	C4	Decode	Read register #d	Rotate	Write to destination
C1	C2	C3	C4						
Decode	Read register #d	Rotate	Write to destination						
Registers:	WREG, #d, 1, 1								
Relative Instructions:	REG = 1144 1111								
After Instructions:	REG = 1111 1111								
Registers:	WREG, #d, 1, 1								
Relative Instructions:	REG = 1144 1111								
After Instructions:	REG = 1111 1111								

2. Compile, assemble and link the software using the assembler and/or compiler and linker to convert your code into "ones and zeroes" – machine code for the PICmicro MCUs. This machine code will eventually become the firmware (the code programmed into the microcontroller).
3. Test your code. Usually a complex program does not work exactly the way imagined, and "bugs" need to be removed from the design to get proper results. The debugger allows you to see the "ones and zeroes" execute, related to the source code you wrote, with the symbols and function names from your program. Debugging allows you to experiment with your code to see the value of variables at various points in the program, and to do "what if" checks, changing variable values and stepping through routines.
4. "Burn" the code into a microcontroller and verify that it executes correctly in the finished application.

Of course, each of these steps can be quite complex. The important thing is to concentrate on the details of your own design, while relying upon MPLAB IDE and its components to get through each step without continuously encountering new learning curves.

What is MPLAB® IDE?

Step 1 is driven by the designer, although MPLAB IDE can help in modeling circuits and code so that crucial design decisions can be made.

MPLAB IDE really helps with steps 2 through 4. Its Programmer's Editor helps write correct code with the language tools of choice. The editor is aware of the assembler and compiler programming constructs and automatically "color-keys" the source code to help ensure it is syntactically correct. The Project Manager enables you to organize the various files used in your application: source files, processor description header files and library files. When the code is built, you can control how rigorously code will be optimized for size or speed by the compiler and where individual variables and program data will be programmed into the device. You can also specify a "memory model" in order to make the best use of the microcontroller's memory for your application. If the language tools run into errors when building the application, the offending line is shown and can be "double clicked" to go to the corresponding source file for immediate editing. After editing, press the "build" button to try again. Often this write-compile-fix loop is done many times for complex code as the sub-sections are written and tested. MPLAB IDE goes through this loop with maximum speed, allowing you to get on to the next step.

Once the code builds with no errors, it needs to be tested. MPLAB IDE has components called "debuggers" and free software simulators for all PICmicro MCU and dsPIC DSC devices to help test the code. Even if the hardware is not yet finished, you can begin testing the code with the simulator, a software program that simulates the execution of the microcontroller. The simulator can accept a simulated input (stimulus), in order to model how the firmware responds to external signals. The simulator can measure code execution time, single step through code to watch variables and peripherals, and trace the code to generate a detailed record of how the program ran.

Once the hardware is in a prototype stage, a hardware debugger, such as MPLAB ICE or MPLAB ICD 2 can be used. These debuggers run the code in real time on your actual application. The MPLAB ICE physically replaces the microcontroller in the target using a high-speed probe to give you full control over the hardware in your design. The MPLAB ICD 2 uses special circuitry built into many Microchip MCUs with Flash program memory and can "see into" the target microcontrollers program and data memory. The MPLAB ICD 2 can stop and start program execution, allowing you to test the code with the microcontroller in place on the application.

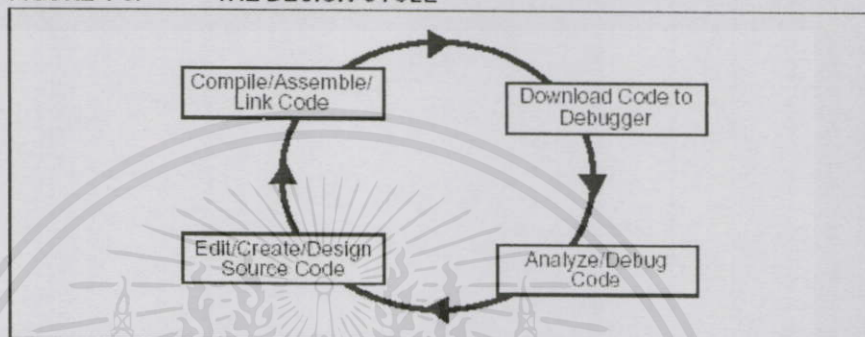
After the application is running correctly, you can program a microcontroller with one of Microchip's device programmers, such as PICSTART® Plus or MPLAB PM3. These programmers verify that the finished code will run as designed. MPLAB IDE supports most PICmicro MCUs and every dsPIC Digital Signal Controller.

MPLAB® IDE Quick Start Guide

1.2 THE DEVELOPMENT CYCLE

The process for writing an application is often described as a development cycle, since it is rare that all the steps from design to implementation can be done flawlessly the first time. More often code is written, tested and then modified in order to produce an application that performs correctly. The Integrated Development Environment allows the embedded systems design engineer to progress through this cycle without the distraction of switching among an array of tools. By using MPLAB IDE, all the functions are integrated, allowing the engineer to concentrate on completing the application without the interruption of separate tools and different modes of operation.

FIGURE 1-5: THE DESIGN CYCLE



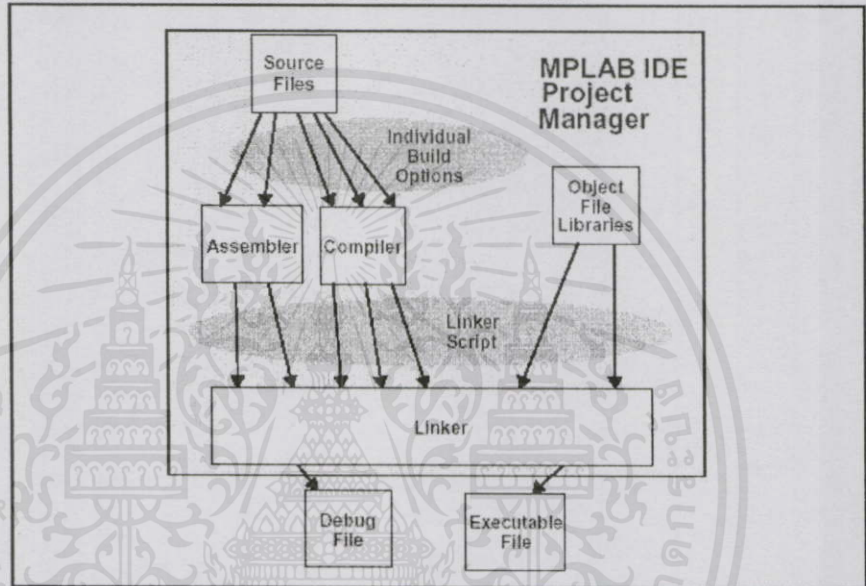
MPLAB IDE is a “wrapper” that coordinates all the tools from a single graphical user interface, usually automatically. For instance, once code is written, it can be converted to executable instructions and downloaded into a microcontroller to see how it works. In this process multiple tools are needed: an editor to write the code, a project manager to organize files and settings, a compiler or assembler to convert the source code to machine code and some sort of hardware or software that either connects to a target microcontroller or simulates the operation of a microcontroller.

What is MPLAB® IDE?

1.3 PROJECT MANAGER

The project manager organizes the files to be edited and other associated files so they can be sent to the language tools for assembly or compilation, and ultimately to a linker. The linker has the task of placing the object code fragments from the assembler, compiler and libraries into the proper memory areas of the embedded controller, and ensure that the modules function with each other (or are "linked"). This entire operation from assembly and compilation through the link process is called a project "build". From the MPLAB IDE project manager, properties of the language tools can be invoked differently for each file, if desired, and a build process integrates all of the language tools operations.

FIGURE 1-6: MPLAB® IDE PROJECT MANAGER



The source files are text files that are written conforming to the rules of the assembler or compiler. The assembler and compiler convert them into intermediate modules of machine code and placeholders for references to functions and data storage. The linker resolves these placeholders and combines all the modules into a file of executable machine code. The linker also produces a debug file which allows MPLAB IDE to relate the executing machine codes back to the source files.

A text editor is used to write the code. It is not a normal text editor, but an editor specifically designed for writing code for Microchip MCUs. It recognizes the constructs in the text and uses color coding to identify various elements, such as instruction mnemonics, C language constructs and comments. The editor supports operations commonly used in writing source code, such as finding matching braces in C, commenting and un-commenting out blocks of code, finding text in multiple files and adding special bookmarks. After the code is written, the editor works with the other tools to display code execution in the debugger. Breakpoints can be set in the editor, and the values of variables can be inspected by hovering the mouse pointer over the variable name. Names of variables can be dragged from source text windows and then dropped into a Watch window.

MPLAB® IDE Quick Start Guide

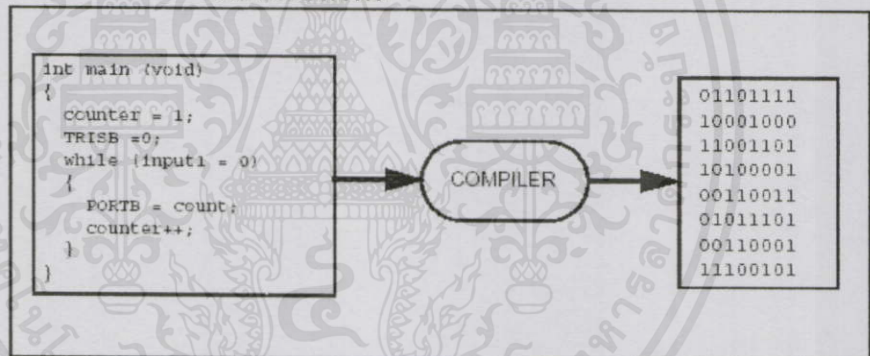
1.4 LANGUAGE TOOLS

Language tools are programs such as cross-assemblers and cross-compilers. Most people are familiar with language tools that run on a PC such as Visual Basic or C compilers. When using language tools for embedded systems, a "cross-assembler" or "cross-compiler" is used. These tools differ from typical compilers in that they run on a PC but produce code to run on another microprocessor, hence they "cross-compile" code for a microcontroller that uses an entirely different set of instructions from the PC.

The language tools also produce a debug file that MPLAB IDE uses to correlate the machine instructions and memory locations with the source code. This bit of integration allows the MPLAB IDE editor to set breakpoints, allows Watch windows to view variable contents, and lets you single step through the source code, watching the application execute.

Embedded system language tools also differ somewhat for compilers that run and execute on a PC because they must be very space conscious. The smaller the code produced, the better, because that allows the smallest possible memory for the target, which reduces cost. This means that techniques to optimize and enhance the code using machine specific knowledge are desirable. The size of programs for PCs typically extends into the megabytes for moderately complex programs. The size of simple embedded systems programs may be as small as a thousand bytes or less. A medium size embedded system might need 32K or 64K of code for relatively complex functions. Some embedded systems use megabytes of storage for large tables, user text messages or data logging.

FIGURE 1-7: A COMPILER CONVERTS SOURCE CODE INTO MACHINE INSTRUCTIONS



What is MPLAB® IDE?

1.5 TARGET DEBUGGING

In a development environment, the execution of the code is tested on a debugger. The debugger can be a software program that simulates the operation of the microcontroller for testing, or it can be special instrumentation to analyze the program as it executes in the application.

Simulators are built into MPLAB IDE so a program can be tested without any additional hardware. A simulator is a software debugger, and the debugger functions for the simulator are almost identical to the hardware debuggers, allowing a new tool to be learned with ease. Usually a simulator runs somewhat slower than an actual microcontroller, since the CPU in the PC is being used to simulate the operations of the microcontroller. In the case of MPLAB IDE, there are many simulators for each of the PICmicro MCU and the dsPIC DSC processors.

There are two types of hardware that can be used with MPLAB IDE: programmers and hardware debuggers. A programmer simply transfers the machine code from the PC into the internal memory of the target microcontroller. The microcontroller can then be plugged into the application and, hopefully, it will run as designed.

Usually, however, the code does not function exactly as anticipated, and the engineer is tasked with reviewing the code and its operation in the application to determine how to modify the original source code to make it execute as desired. This process is called debugging. As noted previously, the simulator can be used to test how the code will operate, but once a microcontroller is programmed with the firmware, many things outside the scope of the simulator come into play. Using just a programmer, the code could be changed, reprogrammed into the microcontroller and plugged into the target for retest, but this could be a long, laborious cycle if the code is complex, and it is difficult to understand exactly what is going wrong in the hardware.

This is where a hardware debugger is useful. Hardware debuggers can be in-circuit emulators, which use specialized hardware in place of the actual target microcontroller, or they can be in-circuit debuggers, which use microcontrollers that have special built-in debugging features. A hardware debugger, like a simulator, allows the engineer to inspect variables at various points in the code, and single step to follow instructions as the hardware interacts with its specialized circuitry.

Debugging usually becomes urgent near the end of the projected design cycle. As deadlines loom, getting the application to function as originally designed is the last step before going into deployment of the product, and often has the most influence on producing delays in getting a product out. That's where an integrated development environment is most important. Doing fine "tweaks" to the code, recompiling, downloading and testing all require time. Using all tools within a single environment will reduce the time around the "cycle." These last steps, where critical bugs are worked out, are a test for the embedded systems designer. The right tool can save time. With MPLAB IDE many tools can be selected, but they all will have a similar interface, and the learning curve from simulator to low-cost in-circuit debugger to powerful in-circuit emulator is small.

MPLAB® IDE Quick Start Guide

1.6 DEVICE PROGRAMMING

After the application has been debugged and is running in the development environment, it needs to be tested on its own. A device can be programmed with the in-circuit debugger or a device programmer. MPLAB IDE can be set to the programmer function, and the part can be "burned". The target application can now be observed in its nearly final state. Engineering prototype programmers allow quick prototypes to be made and evaluated. Some applications can be programmed after the device is soldered on the target PC board. Using In-Circuit Serial Programming™ (ICSP™) programming capability, the firmware can be programmed into the application at the time of manufacture, allowing updated revisions to be programmed into an embedded application later in its life cycle. Devices that support in-circuit debugging can even be plugged back into the MPLAB ICD 2 after manufacturing for quality tests and development of next generation firmware.

1.7 COMPONENTS OF MPLAB IDE

The MPLAB IDE has both built-in components and plug-in modules to configure the system for a variety of software and hardware tools.

1.7.1 MPLAB IDE Built-In Components

The built-in components consist of:

- **Project Manager**

The project manager provides integration and communication between the IDE and the language tools.

- **Editor**

The editor is a full-featured programmer's text editor that also serves as a window into the debugger.

- **Assembler/Linker and Language Tools**

The assembler can be used stand-alone to assemble a single file, or can be used with the linker to build a project from separate source files, libraries and recompiled objects. The linker is responsible for positioning the compiled code into memory areas of the target microcontroller.

- **Debugger**

The Microchip debugger allows breakpoints, single stepping, Watch windows and all the features of a modern debugger for the MPLAB IDE. It works in conjunction with the editor to reference information from the target being debugged back to the source code.

- **Execution Engines**

There are software simulators in MPLAB IDE for all PICmicro MCU and dsPIC DSC devices. These simulators use the PC to simulate the instructions and some peripheral functions of the PICmicro MCU and dsPIC DSC devices. Optional in-circuit emulators and in-circuit debuggers are also available to test code as it runs in the applications hardware.

What is MPLAB® IDE?

1.7.2 Additional Optional Components for MPLAB IDE

Optional components can be purchased and added to the MPLAB IDE:

- **Compiler Language Tools**

MPLAB C18 and MPLAB C30 C compilers from Microchip provide fully integrated, optimized code. Along with compilers from HI-TECH, IAR, microEngineering Labs, CCS and Byte Craft, they are invoked by the MPLAB IDE project manager to compile code that is automatically loaded into the target debugger for instant testing and verification.

- **Programmers**

PICSTART Plus, PICKIT 1 and 2, PRO MATE II, MPLAB PM3 as well as MPLAB ICD 2 can program code into target devices. MPLAB IDE offers full control over programming both code and data, as well as the Configuration bits to set the various operating modes of the target microcontrollers or digital signal controllers.

- **In-Circuit Emulators**

MPLAB ICE 2000 and MPLAB ICE 4000 are full-featured emulators for the PICmicro MCU and dsPIC DSC devices. They connect to the PC via I/O ports and allow full control over the operation of microcontroller in the target applications.

- **In-Circuit Debugger**

MPLAB ICD 2 provides an economic alternative to an emulator. By using some of the on-chip resources, MPLAB ICD 2 can download code into a target microcontroller inserted in the application, set breakpoints, single step and monitor registers and variables.

1.8 MPLAB IDE DOCUMENTATION

The following documents are available to help you use MPLAB IDE:

- “MPLAB® IDE Quick Chart” (DS51410)
- “MPLAB® IDE Quick Start Guide” (DS51281) – Chapters 1 and 2 of user’s guide.
- “MPLAB® IDE User’s Guide” (DS51519)

Other documents exist for various Microchip software and hardware tools that work with MPLAB IDE. Check the Microchip web site for downloadable PDF versions of all these documents.

1.9 MPLAB IDE ON-LINE HELP

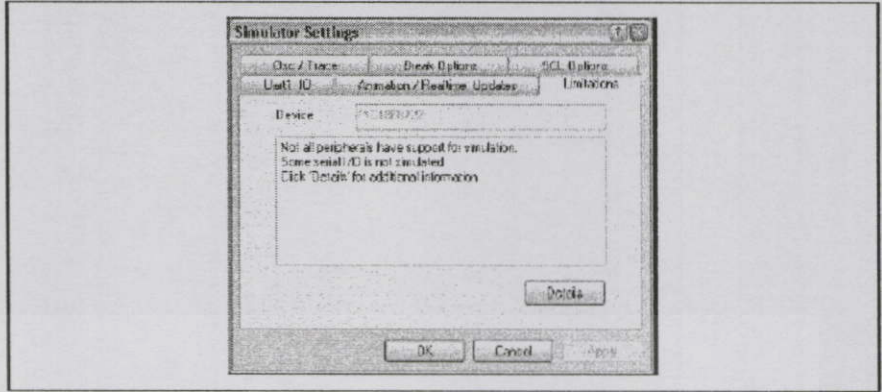
Since MPLAB IDE is under a constant state of change (see **Section 1.10 “MPLAB IDE Updates and Version Numbering”**) some details in this documentation may change. Dialogs might not appear exactly as they do in this manual, menu lists may be in different order, or may have new items. For this reason, the on-line help is the best reference to the version of MPLAB IDE being used.

MPLAB IDE comes with extensive on-line help, which is constantly being updated. If questions arise while using MPLAB IDE, be sure to check the on-line help for answers. Most importantly, the on-line help lists any restrictions that might exist for a particular tool in support of a particular device. Always try to review this section before working with a new device/tool combination.

The Limitations tab in the *Debugger>Settings* dialog displays any restrictions the simulator, emulator or in-circuit debugger might have, compared to the actual device being simulated. General limitations are shown in the text area.

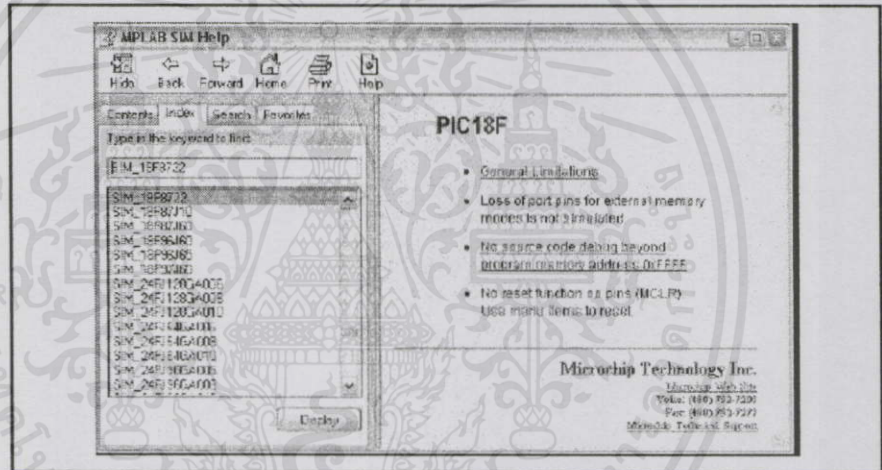
MPLAB® IDE Quick Start Guide

FIGURE 1-8: **DEBUGGER>SETTINGS: LIMITATIONS TAB**



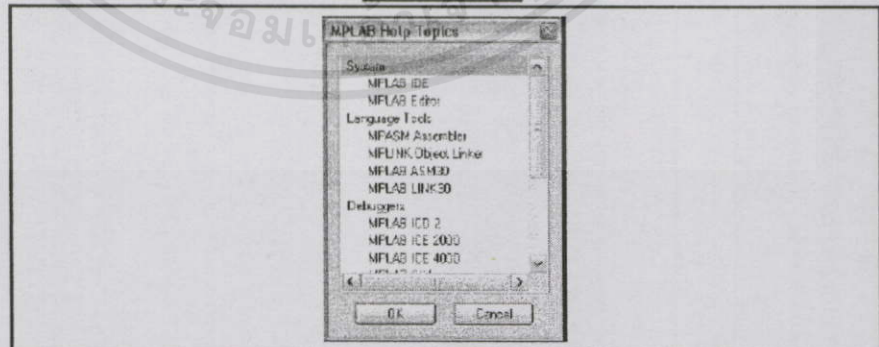
Press the **Details** button to show specific limitations of the device being debugged. From this display, help on general limitations related to the debugger can also be accessed.

FIGURE 1-9: **SIMULATOR LIMITATIONS DETAIL**



From the main MPLAB IDE Help menu, select **Help>Topics** to get a list of help on MPLAB IDE and all of its components.

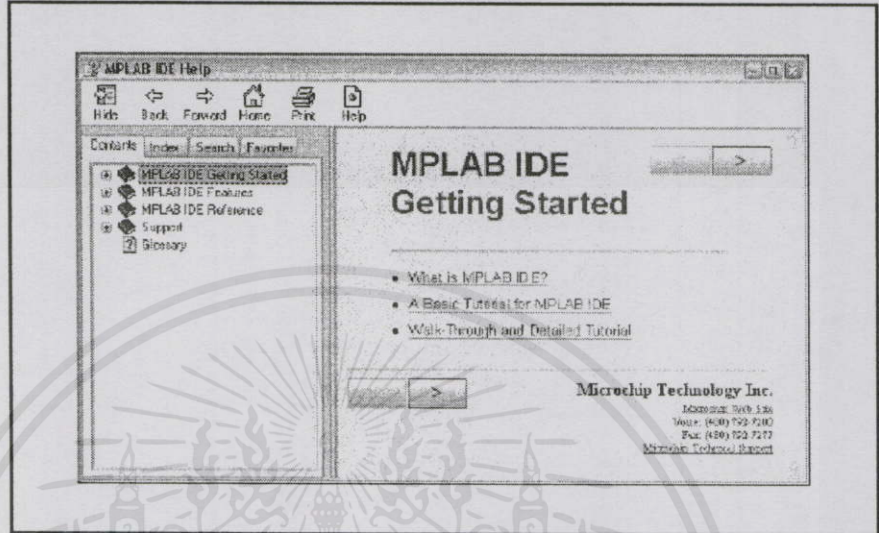
FIGURE 1-10: **MPLAB® IDE HELP>TOPICS MENU**



What is MPLAB® IDE?

MPLAB IDE Help covers all aspects of MPLAB IDE and all of the Microchip tools. It can be viewed in an outline, as an index and with a search utility for help on any MPLAB IDE topic. It also directs users to other types of assistance, such as the Microchip Update Notification system.

FIGURE 1-11: MPLAB® IDE HELP DIALOG



MPLAB® IDE Quick Start Guide

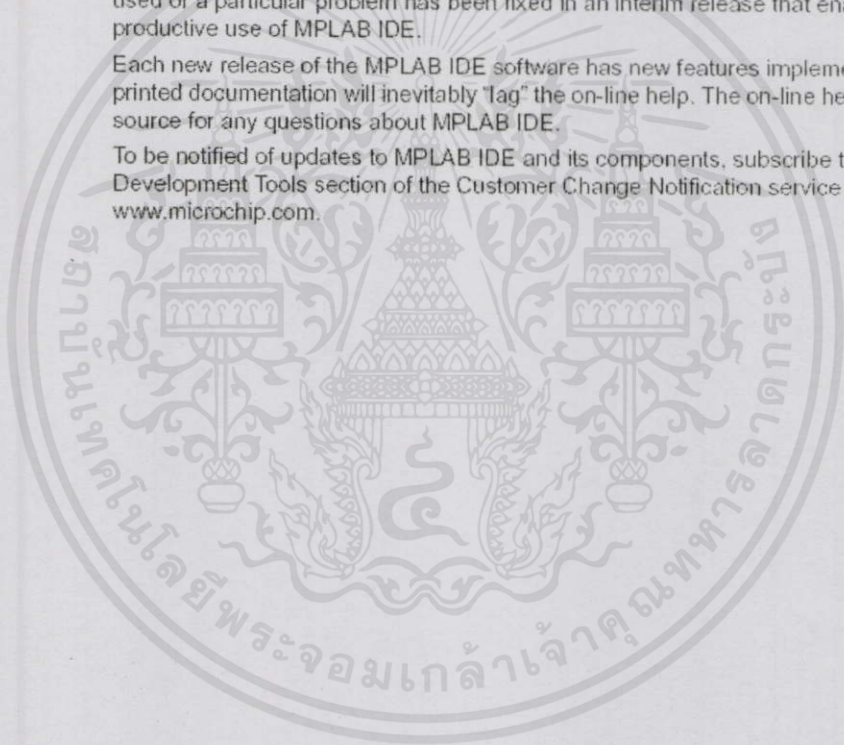
1.10 MPLAB IDE UPDATES AND VERSION NUMBERING

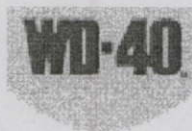
MPLAB IDE is an evolving program with thousands of users. Microchip is continually designing new microcontrollers with new features. Many new MPLAB IDE features come from customer requests and from internal usage. Continued new designs and the release of new microcontrollers ensure that MPLAB IDE will continue to evolve.

MPLAB IDE is scheduled for a version update approximately every four months to add new device support and new features. Additional "interim" releases are produced in between these major releases. The version numbering scheme for MPLAB IDE reflects whether it is a major production release or an interim release. If the version number ends in a zero, e.g., MPLAB IDE v7.00, v7.10 or v7.20, this identifies a major production release. If the version number ends with a digit other than zero, e.g., v7.11, v7.22 or v7.55, this identifies an interim release. Interim releases are typically provided for early adopters of new devices or components, for quick critical fixes or for previews of new features. These interim releases, although based on production releases that are fully tested, may have components that are not as fully tested, and therefore these releases are not recommended for rigorous design use. It is advised that production releases be used for development work, unless a new device or component is being used or a particular problem has been fixed in an interim release that enables more productive use of MPLAB IDE.

Each new release of the MPLAB IDE software has new features implemented, so the printed documentation will inevitably "lag" the on-line help. The on-line help is the best source for any questions about MPLAB IDE.

To be notified of updates to MPLAB IDE and its components, subscribe to the Development Tools section of the Customer Change Notification service on www.microchip.com.




WD-40 Company


Material Safety Data Sheet

1 - Chemical Product and Company Identification

Manufacturer: WD-40 Company	Chemical Name: Organic Mixture
Address: 1061 Cudahy Place (92110) P.O. Box 80607 San Diego, California, USA 92138 -0607	Trade Name: WD-40 Bulk Liquid
Telephone:	Product Use: Cleaner, Lubricant
Emergency only: 1-888-324-7596 (PROZAR)	MSDS Date Of Preparation: August 2007
Information: 1-888-324-7596	

2 - Hazards Identification

Emergency Overview:

DANGER! Harmful or fatal if swallowed. Combustible Liquid. Avoid eye contact. Use with adequate ventilation. Keep away from heat, sparks and all other sources of ignition.

Symptoms of Overexposure:

Inhalation: High concentrations may cause nasal and respiratory irritation and central nervous system effects such as headache, dizziness and nausea. Intentional abuse may be harmful or fatal.

Skin Contact: Prolonged and/or repeated contact may produce mild irritation and defatting with possible dermatitis.

Eye Contact: Contact may be mildly irritating to eyes. May cause redness and tearing.

Ingestion: This product has low oral toxicity. Swallowing may cause gastrointestinal irritation, nausea, vomiting and diarrhea. This product is an aspiration hazard. If swallowed, can enter the lungs and may cause chemical pneumonitis.

Chronic Effects: None expected.

Medical Conditions Aggravated by Exposure: Preexisting eye, skin and respiratory conditions may be aggravated by exposure.

Suspected Cancer Agent:

Yes No

3 - Composition/Information on Ingredients

Ingredient	CAS #	Weight Percent
Aliphatic Petroleum Distillates	64742-47-8	70%
	64742-48-9	
	64742-88-7	
Petroleum Base Oil	64742-65-0	>20%
Non-Hazardous Ingredients	Mixture	<10

See Section 8 for Exposure Limits

4 - First Aid Measures

Ingestion (Swallowed): Aspiration Hazard. DO NOT induce vomiting. Call physician, poison control centre or the WD-40 Safety Hotline at 1-888-324-7596 immediately.

Eye Contact: Flush thoroughly with water. Get medical attention if irritation persists.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Skin Contact: Wash with soap and water. If irritation develops and persists, get medical attention.

Inhalation (Breathing): If irritation is experienced, move to fresh air. Get medical attention if irritation or other symptoms develop and persist.

5 – Fire Fighting Measures

Extinguishing Media: Use water fog, dry chemical, carbon dioxide or foam. Do not use water jet or flooding amounts of water. Burning product will float on the surface and spread fire.

Special Fire Fighting Procedures: Firefighters should always wear positive pressure self-contained breathing apparatus and full protective clothing. Cool fire-exposed containers with water.

Unusual Fire and Explosion Hazards: Combustible liquid and vapour. Vapours are heavier than air and may travel along surfaces to remote ignition sources and flash back.

6 – Accidental Release Measures

Wear appropriate protective clothing (see Section 8). Eliminate all sources of ignition and ventilate area. Contain and collect liquid with an inert absorbent and place in a container for disposal. Clean spill area thoroughly. Report spills to authorities as required.

7 – Handling and Storage

Handling: Avoid contact with eyes. Avoid prolonged contact with skin. Avoid breathing vapors or aerosols. Use with adequate ventilation. Keep away from heat, sparks, hot surfaces and open flames. Wash thoroughly with soap and water after handling. Keep containers closed when not in use. Keep out of the reach of children.

Storage: Store in a cool, well-ventilated area, away from incompatible materials. NFPA 30 Class II Liquid.

8 – Exposure Controls/Personal Protection

Chemical	Occupational Exposure Limits
Aliphatic Petroleum Distillates	100ppm PEL
Petroleum Base Oil	5 mg/m ³ TWA (OSHA/ACGIH)
Non-Hazardous Ingredients	None Established

The Following Controls are Recommended for Normal Consumer Use of this Product

Engineering Controls: Use in a well-ventilated area.

Personal Protection:

Eye Protection: Avoid eye contact. Safety glasses or goggles recommended.

Skin Protection: Avoid prolonged skin contact. Chemical resistant gloves recommended for operations where skin contact is likely.

Respiratory Protection: None needed for normal use with adequate ventilation.

For Bulk Processing or Workplace Use the Following Controls are Recommended

Engineering Controls: Use adequate general and local exhaust ventilation to maintain exposure levels below that occupational exposure limits.

Personal Protection:

Eye Protection: Safety goggles recommended where eye contact is possible.

Skin Protection: Wear chemical resistant gloves.

Respiratory Protection: None required if ventilation is adequate. If the occupational exposure limits are exceeded, wear a NIOSH approved respirator. Respirator selection and use should be based on contaminant type, form and concentration. Follow OSHA 1910.134, ANSI Z88.2 and good Industrial Hygiene practice.

Work/Hygiene Practices: Wash with soap and water after handling.

9 – Physical and Chemical Properties

Boiling Point:	300°F (minimum)	Specific Gravity:	0.816 @ 70°F
----------------	-----------------	-------------------	--------------

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และส่งข้อมูลไปถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Solubility in Water:	Insoluble	pH:	Not Applicable
Vapour Pressure:	Not determined	Vapour Density:	Greater than 1
Percent Volatile:	70%	VOC:	568 grams/liter 70%
Coefficient of Water/Oil Distribution:	Not Determined	Appearance/Odour	Cloudy light amber / characteristic odour
Flash Point:	110°F (minimum) Tag Closed Cup	Flammable Limits: (Solvent Portion)	LEL: 1.0% UE:: 6.0%

10 – Stability and Reactivity

Stability: Stable

Hazardous Polymerization: Will not occur.

Conditions to Avoid: Avoid heat, sparks, flames and other sources of ignition.

Incompatibilities: Strong oxidizing agents.

Hazardous Decomposition Products: Carbon monoxide and carbon dioxide.

11 – Toxicological Information

The oral toxicity of this product is estimated to be greater than 5,000 mg/kg based on an assessment of the ingredients. This product is not classified as toxic by established criteria. It is an aspiration hazard.

None of the components of this product is listed as a carcinogen or suspected carcinogen or is considered a reproductive hazard.

12 – Ecological Information

No data is currently available.

13 - Disposal Considerations

If this product becomes a waste, it would be expected to meet the criteria of a RCRA ignitable hazardous waste (D001). However, it is the responsibility of the generator to determine at the time of disposal the proper classification and method of disposal. Dispose in accordance with federal, state, and local regulations.

14 – Transportation Information

DOT Surface Shipping Description: Excepted from Hazmat (49CFR 173.150 (F)) in non-bulk packagings. Bulk Packagings: Combustible Liquid, n.o.s. (contains Petroleum Distillates), NA1993, PG III

IMDG Shipping Description: UN1268, Petroleum Distillates, n.o.s. 3, PG III

15 – Regulatory Information

U.S. Federal Regulations:

CERCLA 103 Reportable Quantity: This product is not subject to CERCLA reporting requirements, however, oil spills are reportable to the National Response Center under the Clean Water Act and many states have more stringent release reporting requirements. Report spills required under federal, state and local regulations.

SARA TITLE III:

Hazard Category For Section 311/312: Acute Health, Fire Hazard

Section 313 Toxic Chemicals: This product contains the following chemicals subject to SARA Title III Section 313 Reporting requirements: None

Section 302 Extremely Hazardous Substances (TPQ): None

EPA Toxic Substances Control Act (TSCA) Status: All of the components of this product are listed on the TSCA inventory

Canadian Environmental Protection Act: All of the ingredients are listed on the Canadian Domestic Substances List or exempt from notification

Canadian WHMIS Classification: Class B-3 (Combustible Liquid)

This MSDS has been prepared according to the criteria of the Controlled Products Regulation (CPR) and the MSDS contains all of the information required by the CPR.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และดัดแปลงอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16 – Other Information:

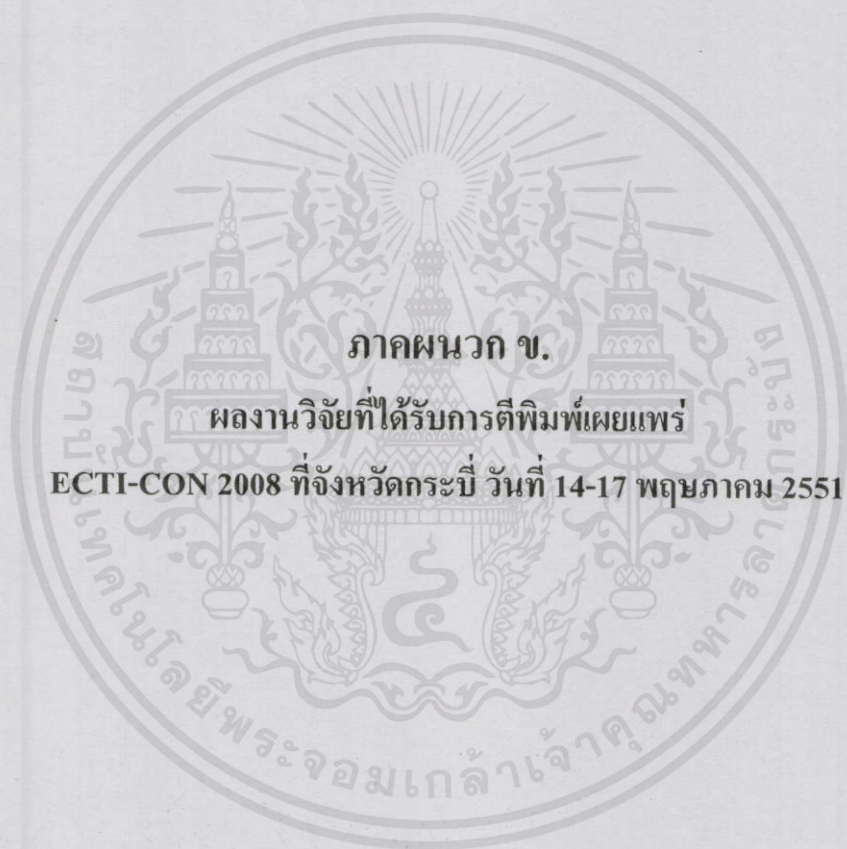
HMIS Hazard Rating:

Health – 1 (slight hazard), Fire Hazard – 2 (moderate hazard), Reactivity – 0 (minimal hazard)

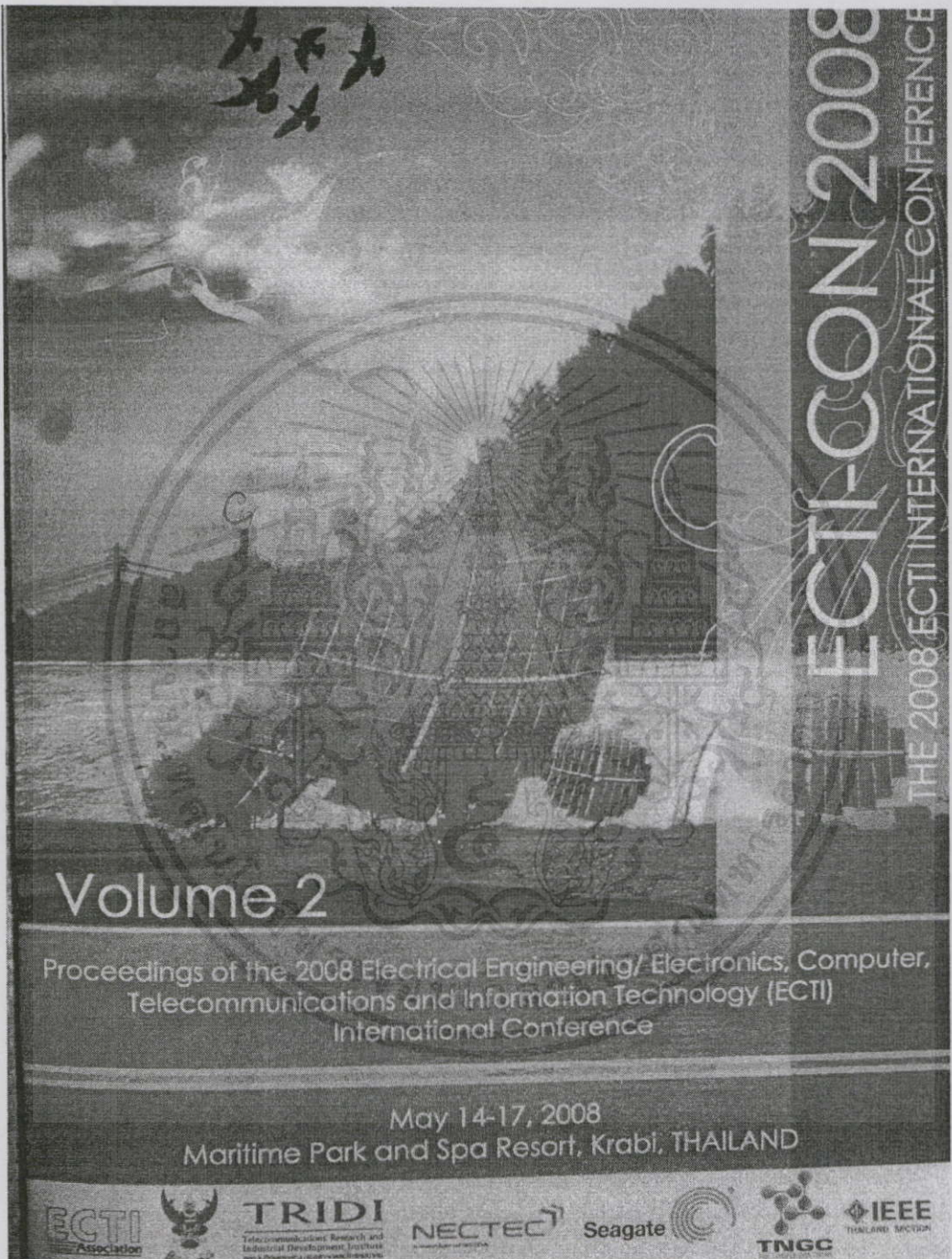
SIGNATURE: *Peter P. Ly* TITLE: Director of Global Quality Assurance
 REVISION DATE: August 2007 SUPERSEDES: March 2001



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ Page 4 of 4 เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Comparison of Switching Strategies for an Ultrasonic Cleaner

Chankit Buasri and Anuwat Jangwanitlert

Department of Electrical Engineering, King Mongkut's Institute of Technology Ladkrabang

Abstract - This paper presents the comparison of Control Signal Patterns in ultrasonic cleaner by using the microcontroller for generating Pulse-Width Modulation (PWM), Phase-Shift PWM control (PS-PWM) and Pulse Density Modulation (PDM) patterns. These control patterns are fed in the full-bridge inverter to drive the ultrasonic transducers. The resonant frequency of 30 kHz is tested in order to find the efficiency of ultrasonic cleaner which one should be the best. Ultrasonic transducer can generate ultrasonic wave to transfer electrical energy to mechanical energy. Cavitation and implosion are occurred in liquid when ultrasonic wave travels through liquid medium in the process of cleaning.

I. INTRODUCTION

Ultrasonic cleaners have favorite in present because cleaner methods have efficacy, fast and environment friendly. Ultrasonic cleaner is a method of immersion cleaner which is able to remove dirty and small particle purely. The transducer is essential equipment to generate vibration from ultrasonic wave. The vibration is sent to all under the tank. The cavitation and implosion are occurred in liquid when ultrasonic wave travels through liquid medium in the process of cleaning. Scrubbing is occurred from cavitation and if cavitations are many and fine, cleaning is the best [1]. Switching strategies use in the current ultrasonic cleaner that is PWM to generate continue ultrasonic wave travel through liquid medium in the process of cleaning. This paper is tested by supplying discontinuous ultrasonic wave (PS-PWM and PDM) [2, 3] and comparison 3 switching strategies which patterns are best with flat equipment or rough equipment under cleaning conditions. Previous research focuses on the clean medical equipment or lab equipment which is flat equipment. This paper concentrates on the electronic circuit board which is rough equipment. Liquid medium used is not break electronic circuit equipment (WD40) [4].

II. ULTRASONIC TRANSDUCER

The cleaner system by the ultrasonic wave includes transducer, signal drive, tank, and liquid medium. Transducer generates vibrate from ultrasonic wave. The vibration is sent to all under the tank. The cavitation and implosion are occurred in liquid when ultrasonic wave travels through liquid medium in the process of cleaning. Some contaminants comprise insoluble particles loosely attached and held in place by ionic or cohesive forces shown in Figure 1A. These particles need only be displaced sufficiently to break the attractive forces to be

removed. The cavitation and implosion as a result of ultrasonic activity displace and remove loosely held contaminants such as dust from surfaces depicted in Figure 1B. For this to be effective, it is necessary that the coupling medium be capable of wetting the particles to be removed [5].

The configuration of ultrasonic cleaner is shown in Figure 2. The switching signal pattern generates from dsPIC microcontroller at resonant frequency of 30 kHz for ultrasonic transducer. It is fed in the full-bridge inverter to drive the ultrasonic transducer. Main equipment to generate ultrasonic wave is ultrasonic transducer. Ultrasonic transducer transfer electrical energy to mechanical energy.

The piezoelectric transducer is used at ultrasonic cleaner. Transducers convert alternating electrical energy directly to mechanical energy by the use of piezoelectric effect. While certain materials change dimension, an electrical charge is applied to them. Electrical energy at the ultrasonic frequency is

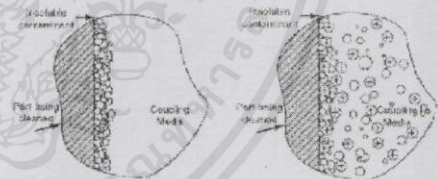


Figure 1A

Figure 1B

Figure 1 Ultrasonic cleaning by dissolution.

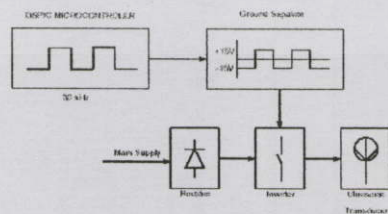


Figure 2 Configuration of ultrasonic cleaner.

from output sends to gate drive circuit shown in Figure 7 and feed to Full-bridge inverter shown in Figure 8 in order to drive ultrasonic transducers. The process of cleaning is occurred.

This test uses switching strategies of 3 patterns at PDM, PWM, PS-PWM shown in Figure 9 to drive ultrasonic transducer and comparison in process of cleaning.

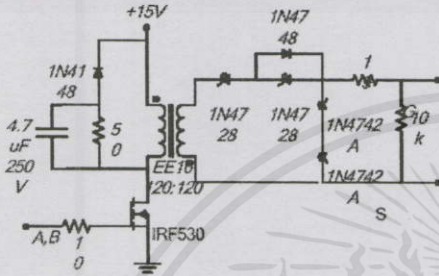


Figure 7 Gate Drive Circuit.

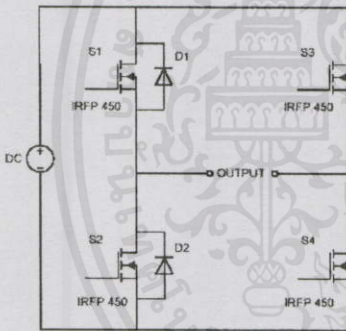


Figure 8 Full-Bridge Inverter.

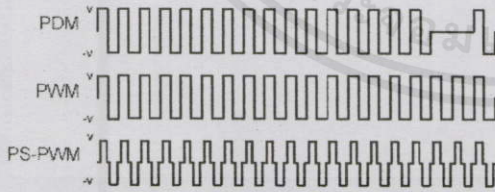


Figure 9 Switching output to drive ultrasonic transducers by using 3 patterns

IV. EXPERIMENTATION RESULTS

The load voltage and current at Ultrasonic Transducer are measured by Digital Storage Oscilloscope (YOKOGAWA DL1620) shown in Figure 10.

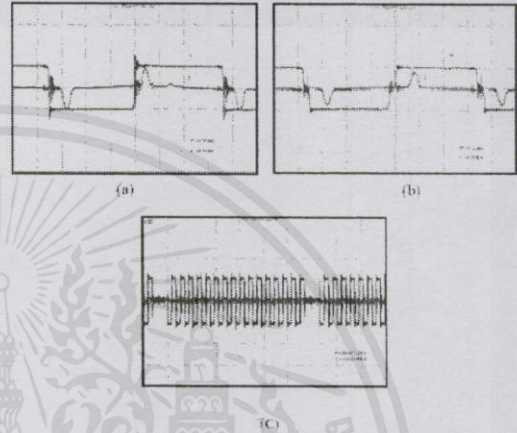


Figure 10 Load Voltage and Current waveforms due to switching patterns (a) PWM (b) PS-PWM (c) PDM.

From Figure 10, the load current is not sinusoidal waveform. The output power can be calculated from math function in oscilloscope follows as (2).

$$P_o = \frac{1}{T} \int_0^T v_o i_o dt \tag{2}$$

Where v_o is instantaneous output voltage and i_o is instantaneous output current. Then, measurement of the output power and input power is used for calculating the efficiency of ultrasonic cleaner as shown in Figure 11.

In addition, voltage and current harmonics are measured by Fluke 43B as shown in Figures 12 and 13.

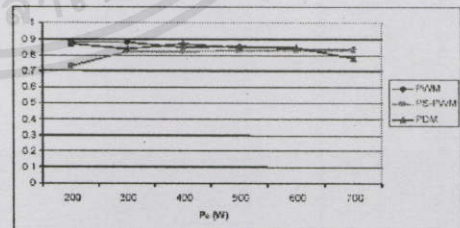


Figure 11 Efficiency of ultrasonic cleaner due to switching patterns.



Figure 12 Voltage harmonics

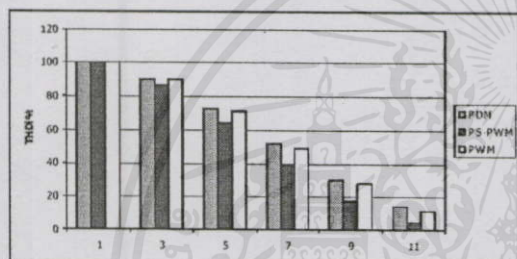


Figure 13 Current harmonics

From the results, % Total Harmonic Voltage Distortion (THD_v) of PWM pattern is equal to 100, 0.8, 5.3, 6.2, 1.9, 4, and %THD_v of PS-PWM pattern is equal to 100, 2.8, 2.2, 1.8, 1.2, 0.4, and that %THD_v of PDM is equal to 100, 1.4, 1.1, 1.1, 0.9, 0.2 at harmonic number $n = 1, 3, 5, 7, 9,$ and $11,$ respectively. Moreover, %THD_i of input current of PWM is equal to 100, 90.1, 71.2, 48.6, 28.1, 11.6, and %THD_i of PS-PWM is equal to 100, 86.8, 64, 38.9, 17.5, 4.6, and that %THD_i of PDM is equal to 100, 89.9, 72.6, 51.7, 29.7, 14.5 at the harmonic number $n = 1, 3, 5, 7, 9, 11,$ respectively.

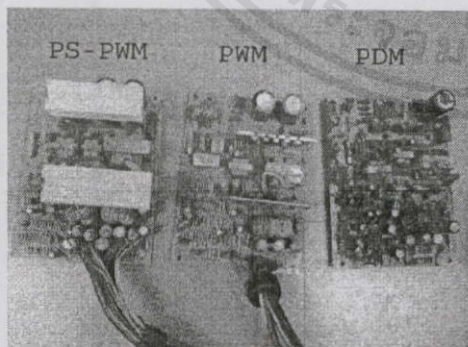


Figure 14 Equipment before using ultrasonic cleaner.

TABLE I
CLEANNESS MEASUREMENT

	Equipment Area	Dirty Area	Dirty Area	Cleanness
		Before Cleaning	After Cleaning	
(cm ²)				
Rough Side				
PS-PWM	159.5	159.5	33	79.3
PWM	152.25	152.25	73.12	52
PDM	159.5	159.5	6	96.23
Flat Side				
PS-PWM	159.5	52	3	94.23
PWM	152.25	67	3	95.52
PDM	159.5	146	3	97.94

From figure 14 and table 1 in test time 10 minutes, when we observe the use of PWM pattern, the deep-dirty is still in equipment. Also, for the use of PS-PWM and PDM patterns, a little dirty is left. It can be recorded in Table I.

V. CONCLUSION

From the test by led signal to generate continuous ultrasonic wave (PWM) and discontinuous (PS-PWM and PDM), the process of cleaning, in case of PWM and PS-PWM, for liquid medium is oscillated, but in case of PDM it is not oscillated. The cavitation bubble in case of PWM and PS-PWM is occurred about 30 minutes but in case of PDM it is occurred approximately 5 minutes and has much fine bubble. Then, referring to the introduction and process of cleaning by using ultrasonic wave, PDM switching strategy is the best for cleaning rough equipment. However, PWM and PS-PWM strategies are the best for cleaning flat equipment.

REFERENCES

- [1] A. Jangwittler, P. Parakwan and T. Vijaktakul, "Ultrasonic cleaner", Proc. in IECON 22, Kasetsart University, Vol. 22, 2542.
- [2] H. Kihme, Y. Hamada and M. Nakaoka, "Cost effective phase shifted pulse modulation soft switching high frequency inverter for induction heating applications", Proc. IEE Electrical Power Appl., 2004, Vol. 151, pp. 19-25.
- [3] H. Fujita and H. Akagi, "Control and Performance of a Pulse-Density-Modulated Series-Resonant Inverter for Corona Discharge Processes", IEEE Trans. on Industry Application, 1999, Vol. 35, pp. 621-627.
- [4] <http://www.wd40.com>, WD-40 company co., Ltd., California USA, 2007
- [5] <http://www.caeblackstone.com/technical.html>, CAE Blackstone co., Ltd., Jamestown NY, 1998.
- [6] Belmcourt, don. "Ultrasonic transducer materials" New-York, Plenum press, 1971.
- [7] N. Pakdeechart, N. Wongsuntornchai and C. Reimpornjitwirai, "Basic dsPIC Microcontroller with C Programming (MPLAB C30)", Innovative experiment co., Ltd., 2004.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และทำซ้ำงานอ้างอิงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายชาญกิต บัวศรี
วัน เดือน ปีเกิด	21 เมษายน 2521 ที่จังหวัดกรุงเทพมหานคร
ที่อยู่	44/213 หมู่บ้านปริณัติกษณ์ เพชรเกษม 69 ซอยเพชรเกษม 69 ถนน เพชรเกษม แขวงหนองแขม เขตหนองแขม จังหวัด กรุงเทพมหานคร 10160
ประวัติการศึกษา	2543 สำเร็จการศึกษาวិชากรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเอเชียอาคเนย์ 2546 เข้าศึกษาต่อในระดับปริญญาโท หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ประวัติการทำงาน	ปัจจุบันทำงานในบริษัท Professional System Engineering จำกัด ในตำแหน่ง วิศวกรไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้