

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ระบบชาวด์สตูดิโอโดยใช้บอร์ด TMS320C6211

SOUND STUDIO SYSTEM USING TMS320C6211 BOARD

โดย

นางสาวพัชรินทร์ ประดิษฐ์เวทิน 44010325

นางสาวสุมาลี เตชะสีบ 44010551

นายสุวิทย์ ภูมิฤทธิกุล 44010566

อาจารย์ที่ปรึกษา

รศ.ดร.ปราโมทย์ วาดเขียน

เลขหมู่.....

เลขทะเบียน..... 61389

วัน,เดือน,ปี..... 17 ก.ค. 2549

b..... 11590015
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบขาค์สตูดิโอโดยใช้อ์บอร์ด TMS320C6211
SOUND STUDIO SYSTEM USING TMS320C6211 BOARD



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

น.พ

ภาควิชา

วิศวกรรมโทรคมนาคม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูอาจารย์ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเผยแพร่ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม
น.พ

ปริญญาานิพนธ์ปีการศึกษา 2547

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบชาวด์สตูดิโอโดยใช้บอร์ด TMS320C6211

SOUND STUDIO SYSTEM USING TMS320C6211 BOARD

ผู้จัดทำ

1. นางสาวพัชรินทร์ ประดิษฐ์เวทิน 44010325

2. นางสาวสุมาลี เตชะสืบ 44010551

3. นายสุวิทย์ ภูมิฤทธิกุล 44010566

.....
.....

อาจารย์ที่ปรึกษา

(รศ.ดร.ปราโมทย์ วาดเขียน)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบชาวด์สตูดิโอโดยใช้บอร์ด TMS320C6211

Sound Studio System using TMS320C6211 Board

โดย นางสาวพัชรินทร์ ประดิษฐ์เวทิน 44010325

นางสาวสุมาลี เศษะสีบ 44010551

นายสุวิทย์ ภูมิฤทธิกุล 44010566

อาจารย์ที่ปรึกษา รศ.ดร.ปราโมทย์ วาดเขียน

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการสร้างระบบชาวด์สตูดิโอซึ่งใช้บอร์ด TMS320C6211 เป็นตัวประมวลผลผ่านการสั่งงานโดยคอมพิวเตอร์โดยมีฟังก์ชันหลักที่ใช้ในการปรับแต่งสัญญาณเสียง 2 ส่วนคือ (1) อีควอลไลเซอร์ ย่าน 0-22 กิโลเฮิร์ต ที่มีตัวกรองความถี่ผ่านแบบคิจิตอลทั้งหมด 10 ย่านความถี่และสามารถปรับค่าอัตราขยายได้ตั้งแต่ -12 เดซิเบลถึง 12 เดซิเบล และ (2) ดีเลย์เอฟเฟ็กต์ซึ่งมีฟังก์ชันของการทำเอ็คโค, รีเวิร์บ และ คอรัส

ABSTRACT

Sound Studio System implemented by a TMS320C6211 board which is controlled by a computer is presented. The two main functions using for adjusting audio signal are composed of (1) a 0-22 kHz equalizer which has 10 of -12 dB to 12dB adjustable gain digital band pass filters and (2) delay effect which has function for echo, reverb and chorus.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	4
2.1 แหล่งกำเนิดของการสั่นสะเทือน	4
2.2 ลักษณะสมบัติของคลื่นเสียง	6
2.3 เสียงเดินทางในอากาศได้อย่างไร	6
2.4 เสียงอย่างง่ายและเสียงเชิงซ้อน	7
2.5 ความถี่สเปกตรัมของเสียงซ้ำ	8
2.6 ความถี่สเปกตรัมของเสียงไม่ซ้ำ	9
2.7 เฟส	10
2.8 เสียงในรูปลักษณะทางไฟฟ้า	11
2.9 เดซิเบล	12
2.10 เสียงกับการได้ยิน	13
2.11 กลไกของการรับฟังเสียง	13
2.12 ความถี่กับการได้ยิน	15
2.13 ความดังของเสียงกับการได้ยิน	16
2.14 แบบคอร์ดเสียง	18
2.15 การปรับแต่งลักษณะของเสียง	19
2.15.1 อีควอไลเซอร์ (Audio Equalizer)	19
2.15.2 เสียงสะท้อน (Echo)	22
2.15.3 เสียงจำลองการสะท้อนของห้อง	24
2.15.4 เสียงคอรัส (Chorus)	25
2.16 การประมวลผลสัญญาณดิจิทัล (Digital Signal Processing)	26
2.16.1 การกรองดิจิทัล (Digital Filter)	26
2.16.2 การแปลงดิจิทัล (Digital Transform)	26
2.17 ข้อได้เปรียบของการประมวลผลสัญญาณดิจิทัล	27
2.18 ข้อเสียของการประมวลผลสัญญาณเชิงดิจิทัล	28
2.19 การแปลงแซด (Z-Transform)	28
2.19.1 บทนิยามของการแปลงแซด	29
2.19.2 การแปลงแซดผกผัน (Inverse Z-Transform)	30
2.20 สมการผลต่างสืบเนื่อง (Difference equation)	31
2.21 ตัวกรองดิจิทัล (Digital filter)	32
2.21.1 ความรู้เบื้องต้นของตัวกรองดิจิทัล (Introduction to Digital filter)	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.21.2 โครงสร้างของตัวกรองดิจิทัล	33
2.21.3 ชนิดของตัวกรองดิจิทัล (Type of digital filter)	34
2.21.4 คุณสมบัติของตัวกรองสัญญาณดิจิทัลแบบเอฟโฟออาร์	35
2.22 ตัวกรองผลตอบสนองอิมพัลส์จำกัด (Finite Impulse Response Filter)	36
2.22.1 การออกแบบตัวกรองแบบไม่ป้อนกลับโดยใช้อัลกอริทึมฟูรีเยอร์	37
2.22.2 การออกแบบตัวกรองแบบไม่ป้อนกลับโดยใช้วิธีหน้าต่าง	39
2.22.2.1 วินโดว์สี่เหลี่ยม (Rectangular window)	39
2.22.2.2 วินโดว์แบบฮานและแฮมมิง (Hann and Hamming window)	42
2.22.2.3 วินโดว์แบบแบล็คแมน (Blackman window)	43
2.23 ความรู้เบื้องต้นเกี่ยวกับบอร์ด DSP TMS320C6211	44
2.23.1 บทนำ	44
2.23.2 ภาพรวมของสถาปัตยกรรมของ TMS320C6211	45
2.23.3 การแม็บบเมโมรี่	45
2.23.4 ชุดคำสั่งของ TMS320C6x	46
2.24 โปรแกรม Code Composer Studio (CCS)	48
2.24.1 Hardware Setup	48
2.24.2 Code & Build	48
2.24.3 การโปรแกรม (Program)	50
2.24.4 Test	51
2.25 การเขียนโปรแกรมติดต่อกับบอร์ด	51
2.25.1 คำสั่งที่ใช้ในการติดต่อกับบอร์ดมีดังนี้	53
2.25.2 วิธีการส่งผ่านข้อมูล	53
2.25.2.1 Read Without Auto-Increment	54
2.25.2.2 Read With Auto-Increment	54
2.25.2.3 Write With and Without Auto-Increment	55
บทที่ 3 การออกแบบ และการสร้าง	57
3.1 ขั้นตอนการออกแบบวงจรกรองผลตอบสนองอิมพัลส์จำกัด	57
3.2 สรุปขั้นตอนการออกแบบวงจรกรองผลตอบสนองอิมพัลส์จำกัด	59
3.3 การออกแบบ FIR Low pass filter ที่ใช้ในการทดลอง	60
3.4 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	64
3.5 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	68
3.6 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	72
3.7 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
3.8 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	80
3.9 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	84
3.10 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	88
3.11 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง	92
3.12 การออกแบบ FIR Highpass filter ที่ใช้ในการทดลอง	96
3.13 การเขียนโปรแกรม Code Composer Studio เพื่อให้ทำงานตามสมการต่างๆ	100
3.13.1 linker command file (file.cmd)	100
3.13.2 Interrupt vector File	101
3.13.3 Initialization/Communication File (C6xdskinit.c)	101
3.13.4 โปรแกรมหลัก (Main Program)	103
3.13.4.1 เอ็กโค (Echo)	106
3.13.4.2 รีเวิร์บ (Reverb)	109
3.13.4.3 คอรัส (Chorus)	112
3.13.4.4 ตัวกรองความถี่แบบ FIR	115
บทที่ 4 การทดลองและผลการทดลอง	122
4.1 การทดลองที่ 1 การทดลองผ่านโปรแกรมเอ็กโค	122
4.2 การทดลองที่ 2 การทดลองผ่านโปรแกรมรีเวิร์บ	123
4.3 การทดลองที่ 3 การทดลองผ่านโปรแกรมคอรัส	125
4.4 การทดลองที่ 4 ผลตอบสนองแอมพลิจูดและเฟสของวงจรกรองความถี่คิจิตอล	129
4.5 การทดลองที่ 5 ลักษณะของสัญญาณไซน์ที่ได้หลังจากผ่านตัวกรองความถี่	146
4.6 การทดลองที่ 6 ลักษณะของสัญญาณรูปคลื่นต่างๆที่ได้หลังจากผ่านตัวกรองความถี่	156
4.7 การทดลองที่ 7 ลักษณะของสัญญาณไซน์เมื่อทำการปรับอิตวอไลเซอร์	157
บทที่ 5 สรุปและวิจารณ์	160
กิตติกรรมประกาศ	161
บรรณานุกรม	162

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 1.1 ระบบที่ใช้ไมโครโปรเซสเซอร์ ซึ่งจะสามารถทำงานเป็นอะไรก็ได้ ขึ้นกับโปรแกรมที่ใส่	1
รูปที่ 1.2 ส่วนประกอบของ DSP	2
รูปที่ 2.1 แหล่งกำเนิดเสียงที่เหมือนทรงกลมมีการสั่นสะเทือนแพร่กระจาย คลื่นออกเป็นทรงกลม	5
รูปที่ 2.2 คลื่นตามยาวที่เกิดขึ้นจากผลการอัด และ ขยายของอากาศ	5
รูปที่ 2.3 ลักษณะคลื่นตามขวาง การเคลื่อนที่ทุกๆจุดของตัวกลางจะเคลื่อนที่ ขึ้นลงในทิศทางที่ตั้งฉากกับทิศทางการเคลื่อนที่ของคลื่น	5
รูปที่ 2.4 รูปคลื่นเสียงมีลักษณะเป็นคลื่นไซน์ โดยคาบของคลื่นแทนด้วย t และความถี่แทนด้วย $1/t$	6
รูปที่ 2.5 เงามันสมมูลสำหรับการเลือกรูปคลื่นอย่างง่าย	7
รูปที่ 2.6 แสดงโหมดการสั่นสะเทือนของเชือก	9
รูปที่ 2.7 ค่าความถี่สเปกตรัมของรูปคลื่นที่ไม่ซ้ำ	10
รูปที่ 2.8 ค่าความถี่สเปกตรัมของรูปคลื่นที่ไม่ซ้ำ	11
รูปที่ 2.9 การรวมกันของคลื่นเสียง 2 สัญญาณ ที่มีเฟสตรงข้ามกัน	11
รูปที่ 2.10 ไมโครโฟนทำหน้าที่เปลี่ยนความกดดันของอากาศที่เกิดขึ้นจากเสียงต่างๆ ให้เป็นการเปลี่ยนแปลงในรูปของแรงดันไฟฟ้า ซึ่งการอัดตัวของอากาศจะได้ ค่าแรงดันไฟฟ้าบวก และการขยายตัวของอากาศ จะได้ค่าแรงดันไฟฟ้าลบ	12
รูปที่ 2.11(a) แผนภาพกลไกของหู	14
รูปที่ 2.11(b) แผนภาพกลไกของหู	14
รูปที่ 2.12 ตำแหน่งสั่นสะเทือนสูงสุดของเยื่อบาซิลาร์เมมเบรนเคลื่อนที่ไป ทางช่องรูปไข่โอวอลวินโดว์ เช่นเดียวกับการเพิ่มขึ้นความถี่	15
รูปที่ 2.13 กราฟความดังของเสียงในหน่วยที่เป็นฟอน	16
รูปที่ 2.14 กราฟความสัมพันธ์ความดังของเสียงในหน่วยฟอนและหน่วยโซน	17
รูปที่ 2.15 กราฟแสดงขนาด Masking tone	18
รูปที่ 2.16 ชนิดของอิกวอไลเซอร์ที่ค่า Q ขึ้นกับจำนวนองศาของการปรับเพิ่มหรือลด	20
รูปที่ 2.17 ค่า Q ของวงจรถึงที่	20
รูปที่ 2.18 ผลตอบสนองเชิงความถี่ของตัวกรองที่ใช้	21
รูปที่ 2.19 โครงสร้างอิกวอไลเซอร์เสียง 5 ช่อง	22
รูปที่ 2.20 การสร้างเสียงสะท้อน	23
รูปที่ 2.21 ผลตอบสนองต่ออิมพัลส์ของระบบ	23
รูปที่ 2.22 การสร้างเสียงสะท้อนโดยใช้วงจรป้อนกลับ	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 2.23 ผลตอบสนองต่ออิมพัลส์ของระบบ	23
รูปที่ 2.24 การสร้างเสียงจำลองการสะท้อนของห้อง	25
รูปที่ 2.25 โครงสร้างการทำเสียงคอรัส	26
รูปที่ 2.26(a) แผนภาพอุปกรณ์ในการประมวลผลสัญญาณดิจิทัล	31
รูปที่ 2.26(b) แผนภาพแทนสมการผลต่างสลับเนื่องอันดับ 1	31
รูปที่ 2.27 บล็อกไดอะแกรมของ Transversal filter โดยที่มี $h(k), (k = 0,1,2,\dots,n-1)$	33
รูปที่ 2.28 การวิเคราะห์สมการผลต่างสลับเนื่องในรูปแบบของบล็อกไดอะแกรม	33
รูปที่ 2.29 การวิเคราะห์สมการผลต่างสลับเนื่องในรูปแบบ Signal Flow Graph Diagram	34
รูปที่ 2.30 บล็อกไดอะแกรมหลักการของตัวกรองดิจิทัล	34
รูปที่ 2.31 โครงสร้างของตัวกรองสัญญาณดิจิทัลแบบเฟอไออาร์	36
รูปที่ 2.32 ลักษณะสัญญาณของผลตอบสนองอิมพัลส์จำกัด	36
รูปที่ 2.33 ผลตอบสนองอูคมคติ	38
รูปที่ 2.34 ผลตอบสนองแอมพลิจูดของวินโดว์สี่เหลี่ยม	40
รูปที่ 2.35(a) วิธีการทำวินโดว์ในโดเมนความถี่	41
รูปที่ 2.35(b) วิธีการทำวินโดว์ในโดเมนความถี่	41
รูปที่ 2.35(c) วิธีการทำวินโดว์ในโดเมนความถี่	42
รูปที่ 2.36 สเปกตรัมของวินโดว์แบบสี่เหลี่ยม วินโดว์แบบฮานและวินโดว์แบบแฮมมิง กรณี $N = 11$	42
รูปที่ 2.37 สถาปัตยกรรมของของ TMS320C6211	44
รูปที่ 2.38 การ Add file to project	49
รูปที่ 2.39 การ Rebuild all	50
รูปที่ 2.40 การ โปรแกรม	50
รูปที่ 2.41 บล็อกไดอะแกรมของ TMS320C6211	51
รูปที่ 2.42 บล็อกไดอะแกรมของHPI	52
รูปที่ 3.1 คุณสมบัติของวงจรกรองความถี่	57
รูปที่ 3.2 การตอบสนองความถี่	57
รูปที่ 3.3 จำกัดจำนวนของการตอบสนองความถี่	57
รูปที่ 3.4 เลื่อนการตอบสนองความถี่มาทางขวาทั้งหมด	58
รูปที่ 3.5 การตอบสนองความถี่ของหน้าต่าง	58
รูปที่ 3.6 การตอบสนองความถี่ของ $h(n)$	58
รูปที่ 3.7 ขั้นตอนการออกแบบวงจรกรองผลตอบสนองอิมพัลส์จำกัด	59
รูปที่ 3.8 แผนภาพการทำงานโปรแกรม liker command file	100

	หน้า
รูปที่ 3.9 แผนภาพการทำงาน โปรแกรม Initialization/Communication File	102
รูปที่ 3.10 แผนภาพการทำงาน โปรแกรมย่อยฟังก์ชันการติดต่อด้วยการอินเทอร์รัพ	103
รูปที่ 3.11 แผนภาพการทำงาน โปรแกรมหลัก	104
รูปที่ 3.12 แผนภาพการทำงานฟังก์ชัน comm_intr (ติดต่อกับ ISR) หรือ c_int11	105
รูปที่ 3.13 แผนภาพการเขียนโปรแกรมเอ็ดโค	107
รูปที่ 3.14 slide bar ที่ได้	109
รูปที่ 3.15 แผนภาพการเขียนโปรแกรมรีเวบ	110
รูปที่ 3.16 แผนภาพการเขียนโปรแกรมคอร์ธ	113
รูปที่ 3.17 แผนภาพการเขียนโปรแกรมฟิลเตอร์	116
รูปที่ 4.1 ผลเมื่อป้อนสัญญาณพัลซรูปไซน์ผ่านโปรแกรมเอ็ดโค	122
รูปที่ 4.2 ไฟล์เสียงก่อนและหลังป้อนเข้าโปรแกรมเอ็ดโคโดยพล็อต MATLAB	123
รูปที่ 4.3 ผลเมื่อป้อนสัญญาณพัลซรูปไซน์ผ่านโปรแกรมรีเวบ	124
รูปที่ 4.4 ไฟล์เสียงก่อนและหลังป้อนเข้าโปรแกรมรีเวบโดยพล็อต MATLAB	124
รูปที่ 4.5 ผลเมื่อป้อนสัญญาณผ่านบอร์ดที่โปรแกรมคอร์ธที่ปรับค่า 50	125
รูปที่ 4.6 ผลเมื่อป้อนสัญญาณผ่านบอร์ดที่โปรแกรมคอร์ธที่ปรับค่า 100	126
รูปที่ 4.7 สเปกตรัมของสัญญาณอินพุตก่อนผ่านโปรแกรมคอร์ธ	127
รูปที่ 4.8 สเปกตรัมของสัญญาณเอาต์พุตหลังผ่านโปรแกรมคอร์ธ	128
รูปที่ 4.9 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่ต่ำผ่านที่ได้จากการออกแบบ	130
รูปที่ 4.10 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่ต่ำผ่าน ที่ได้จากการทดลอง	130
รูปที่ 4.11 ผลตอบสนองเฟสของตัวกรองความถี่ต่ำผ่าน ที่ได้จากการทดลอง	131
รูปที่ 4.12 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 450Hz ถึง 845Hz ที่ได้จากการออกแบบ	131
รูปที่ 4.13 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 450Hz ถึง 845Hz ที่ได้จากการทดลอง	132
รูปที่ 4.14 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 450Hz ถึง 845Hz ที่ได้จากการทดลอง	132
รูปที่ 4.15 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 845Hz ถึง 1240Hz ที่ได้จากการออกแบบ	133
รูปที่ 4.16 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 845Hz ถึง 1240Hz ที่ได้จากการทดลอง	133
รูปที่ 4.17 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 845Hz ถึง 1240Hz ที่ได้จากการทดลอง	134

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 4.18 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 1240Hz ถึง 1635Hz ที่ได้จากการออกแบบ	134
รูปที่ 4.19 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 1240Hz ถึง 1635Hz ที่ได้จากการทดลอง	135
รูปที่ 4.20 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 1240Hz ถึง 1635Hz ที่ได้จากการทดลอง	135
รูปที่ 4.21 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 1635Hz ถึง 2030Hz ที่ได้จากการออกแบบ	138
รูปที่ 4.22 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 1635Hz ถึง 2030Hz ที่ได้จากการทดลอง	136
รูปที่ 4.23 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 1635Hz ถึง 2030Hz ที่ได้จากการทดลอง	136
รูปที่ 4.24 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2030Hz ถึง 2425Hz ที่ได้จากการออกแบบ	137
รูปที่ 4.25 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2030Hz ถึง 2425Hz ที่ได้จากการทดลอง	138
รูปที่ 4.26 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2030Hz ถึง 2425Hz ที่ได้จากการทดลอง	138
รูปที่ 4.27 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2425Hz ถึง 2820Hz ที่ได้จากการออกแบบ	139
รูปที่ 4.28 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2425Hz ถึง 2820Hz ที่ได้จากการทดลอง	139
รูปที่ 4.29 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2425Hz ถึง 2820Hz ที่ได้จากการทดลอง	140
รูปที่ 4.30 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2820Hz ถึง 3215Hz ที่ได้จากการออกแบบ	140
รูปที่ 4.31 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2820Hz ถึง 3215Hz ที่ได้จากการทดลอง	141
รูปที่ 4.32 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 2820Hz ถึง 3215Hz ที่ได้จากการทดลอง	141
รูปที่ 4.33 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 3215Hz ถึง 3610Hz ที่ได้จากการออกแบบ	142

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 4.34 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 3215Hz ถึง 3610Hz ที่ได้จากการทดลอง	142
รูปที่ 4.35 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน เท่ากับ 3215Hz ถึง 3610Hz ที่ได้จากการทดลอง	143
รูปที่ 4.36 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่สูงผ่าน ความถี่ผ่าน เท่ากับ 3610Hz ที่ได้จากการออกแบบ	143
รูปที่ 4.37 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่สูงผ่านความถี่ผ่าน เท่ากับ 3610Hz ที่ได้จากการทดลอง	144
รูปที่ 4.38 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่ทั้ง 10 ตัวรวมกัน	145
รูปที่ 4.39 Channel#1 คือสัญญาณ ไซน์ที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณหลังผ่านตัวกรองความถี่ต่ำผ่าน	146
รูปที่ 4.40 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 450Hz ถึง 845Hz	147
รูปที่ 4.41 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz	148
รูปที่ 4.42 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 1240Hz ถึง 1635Hz	149
รูปที่ 4.43 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 1635Hz ถึง 2030Hz	150
รูปที่ 4.44 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 2030Hz ถึง 2425Hz	151
รูปที่ 4.45 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 2425Hz ถึง 2820Hz	152
รูปที่ 4.46 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 2820Hz ถึง 3215Hz	153

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 4.47 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 3215Hz ถึง 3610Hz	154
รูปที่ 4.48 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวม สัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จาก ตัวกรองความถี่กลางผ่านในช่วงความถี่สูงผ่านเท่ากับ 3610Hz	155
รูปที่ 4.49 Channel#1 คือสัญญาณอินพุตเป็นสัญญาณคลื่นรูปสี่เหลี่ยม Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่ต่ำผ่าน	156
รูปที่ 4.50 Channel#1 คือสัญญาณอินพุตเป็นสัญญาณคลื่นรูปสามเหลี่ยม Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่ต่ำผ่าน	157
รูปที่ 4.51 ลักษณะของอควอไลเซอร์เมื่อปรับค่า Gain ในแต่ละย่านความถี่	158
รูปที่ 4.52 Ch#1 อินพุตเป็นสัญญาณไซน์ความถี่ต่างๆรวมกัน Ch#2 เอาท์พุทที่ได้หลังจากผ่านโปรแกรมอควอไลเซอร์	158
รูปที่ 4.53 ลักษณะของอควอไลเซอร์เมื่อปรับค่า Gain ในแต่ละย่านความถี่	159
รูปที่ 4.54 Ch#1 อินพุตเป็นสัญญาณไซน์ความถี่ต่างๆรวมกัน Ch#2 เอาท์พุทที่ได้หลังจากผ่านโปรแกรมอควอไลเซอร์	159

สารบัญตาราง

	หน้า
ตารางที่ 2.1 ระดับความดังของเสียงจากแหล่งกำเนิดเสียง	13
ตารางที่ 2.2 ผลการแปลงแซดและการแปลงแซดผกผัน	30
ตารางที่ 2.3 คุณสมบัติของวินโดว์แบบต่างๆ	43
ตารางที่ 2.4 คุณสมบัติของวินโดว์แบบต่างๆ เมื่อมีอันคืบต่างๆ	43
ตารางที่ 2.5 memory map	46
ตารางที่ 2.6 ชุดคำสั่งต่างๆ	47
ตารางที่ 2.7 ช่วงเวลาที่ <i>HPID</i> อยู่ในสถานะ high ของการอ่านแบบ Without Auto-Increment	54
ตารางที่ 2.8 ช่วงเวลาที่ <i>HRDY</i> _ อยู่ที่สถานะ high ขณะทีอ่านแบบ With Auto-Increment	55
ตารางที่ 2.9 ช่วงเวลาที่ <i>HRDY</i> _ อยู่ที่สถานะ high ระหว่างการเขียน	56
ตารางที่ 3.1 ค่า $w_H(n)$ ที่ได้จาก Hamming window	61
ตารางที่ 3.2 ค่า $h_{LP}(n)$ ที่ได้จากการคำนวณ	63
ตารางที่ 3.3 ค่า $w_H(n)$ ที่ได้จาก Hamming window	65
ตารางที่ 3.4 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	67
ตารางที่ 3.5 ค่า $w_H(n)$ ที่ได้จาก Hamming window	69
ตารางที่ 3.6 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	71
ตารางที่ 3.7 ค่า $w_H(n)$ ที่ได้จาก Hamming window	73
ตารางที่ 3.8 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	75
ตารางที่ 3.9 ค่า $w_H(n)$ ที่ได้จาก Hamming window	77
ตารางที่ 3.10 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	79
ตารางที่ 3.11 ค่า $w_H(n)$ ที่ได้จาก Hamming window	81
ตารางที่ 3.12 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	83
ตารางที่ 3.13 ค่า $w_H(n)$ ที่ได้จาก Hamming window	85
ตารางที่ 3.14 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	87
ตารางที่ 3.15 ค่า $w_H(n)$ ที่ได้จาก Hamming window	89
ตารางที่ 3.16 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	91
ตารางที่ 3.17 ค่า $w_H(n)$ ที่ได้จาก Hamming window	93
ตารางที่ 3.18 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	95
ตารางที่ 3.19 ค่า $w_H(n)$ ที่ได้จาก Hamming window	97
ตารางที่ 3.20 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ	99
ตารางที่ 3.21 แอสเครสของตัวอินเทอร์รัพต่างๆ	101

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

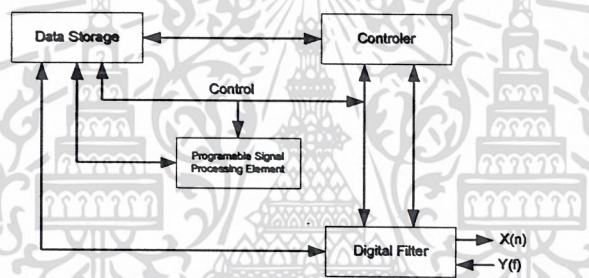
บทที่ 1

บทนำ

ดิจิตอลซิกแนลโพรเซสซิง (Digital Signal Processing: DSP) คือ การเปลี่ยนรูปแบบของสัญญาณ โดยใช้การคำนวณทางคณิตศาสตร์ด้วยดิจิตอลซิกแนลโพรเซสเซอร์ (Digital Signal Processor) ซึ่งจะเห็นได้ว่าเราสามารถนำสัญญาณนั้นไปส่งไปในสมการคณิตศาสตร์ แล้วได้ผลลัพธ์ออกมาโดยตรงทีเดียว

ในช่วง 10 กว่าปีที่ผ่านมา ความก้าวหน้าทางด้านดิจิตอลซิกแนลโพรเซสซิงได้เกิดขึ้นเป็นอย่างมาก มีการค้นพบอัลกอริทึม (Algorithm) ใหม่ ๆ เพิ่มขึ้นเรื่อยๆ และการพัฒนาเครื่องมือใหม่ๆ เพื่อช่วยในการออกแบบตามอัลกอริทึมนั้นอีกด้วย อุปกรณ์ระบบดิจิตอลซิกแนลโพรเซสซิงนั้นจะมีอยู่ 2 แบบใหญ่ๆ ตามไมโครโพรเซสเซอร์ที่ใช้ คือ

1. ระบบที่ใช้ไมโครโพรเซสเซอร์ ออกแบบมาเพื่อให้ทำงานอย่างใดอย่างหนึ่งโดยเฉพาะ
2. ระบบที่ใช้ไมโครโพรเซสเซอร์ ซึ่งจะสามารถทำงานเป็นอะไรก็ได้ขึ้นอยู่กับโปรแกรมที่ใส่



รูปที่ 1.1 ระบบที่ใช้ไมโครโพรเซสเซอร์ ซึ่งจะสามารถทำงานเป็นอะไรก็ได้ขึ้นอยู่กับโปรแกรมที่ใส่

ความก้าวหน้าทางอิเล็กทรอนิกส์ครั้งใหญ่เกิดขึ้นเมื่อมีการสร้างดิจิตอลซิกแนลโพรเซสเซอร์แบบความเร็วสูงขึ้น (High-Speed Signal Processor) ซึ่งปัจจุบันนี้ได้มีใช้กันอย่างแพร่หลายในราคาไม่แพงนัก

ดิจิตอลซิกแนลโพรเซสเซอร์จำเป็นจะต้องมีความเร็วสูงและถูกออกแบบมาโดยเฉพาะให้สามารถทำงานได้อย่างมีประสิทธิภาพ โดยใช้ข้อดีของการประมวลผลแบบขนานและการใช้ชุดคำสั่งที่สร้างขึ้นมาโดยเฉพาะ ทั้งนี้ก็เพื่อการประมวลผลแบบเวลาจริง (Real Time) นั่นเอง การประมวลผลแบบเวลาจริงคือเมื่อเราใส่สัญญาณ อินพุตเข้าไปก็จะได้สัญญาณเอาต์พุตออกมาเลยโดยจะมีเวลาหน่วงน้อยมาก (Delay เท่ากับ 20 ms) ซึ่งจะมีประโยชน์ในการใช้งานมากกว่าการประมวลผลแบบไม่ใช้เวลาจริง

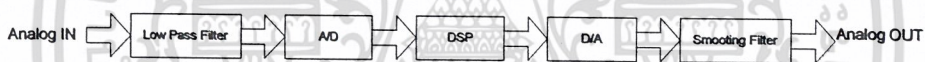
ด้วยความสามารถของดิจิตอลซิกแนลโพรเซสซิงจึงได้นำไปใช้ในรูปแบบต่างๆ ซึ่งให้ผลลัพธ์ที่ดีกว่าโซลูชันทางอนาล็อก เช่น ความน่าเชื่อถือ, ความยืดหยุ่น, การทดแทนได้ง่าย, ขนาดที่เล็กกว่าและเสถียรภาพในระยะยาวและความสามารถของดิจิตอลซิกแนลโพรเซสเซอร์ที่สามารถเปลี่ยนโปรแกรมการทำงานได้ง่ายทำให้การใช้งานดิจิตอลซิกแนลโพรเซสเซอร์นี้น่าสนใจ เช่น สามารถเปลี่ยนให้ดีขึ้นเมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัลกอริทึมใหม่ๆถูกค้นพบหรือความสามารถทำงานได้หลายอย่างโดยเพียงแค่เปลี่ยนโปรแกรมให้กับมันเท่านั้น

ในทางอิเล็กทรอนิกส์โดยส่วนใหญ่มักเกี่ยวข้องกับสัญญาณทางไฟฟ้า เช่น การกรองสิ่งที่ไม่ต้องการออกจากสัญญาณอินพุต การแยกเอาข้อมูลออกมาหรือไม่ก็เป็นการสร้างรูปคลื่นที่ต้องการขึ้นมา หรือการเปลี่ยนลักษณะทางแอมพลิจูดของสัญญาณซึ่งในอดีตการกระทำเหล่านี้จะทำโดยใช้วงจรรอนาลอกที่ออกแบบมาเพื่องานชนิดนั้นๆ นอกจากนั้นวงจรรอนาลอกยังไม่เหมาะที่จะสร้างเป็นวงจรรขนาดใหญ (VLSI) อีกด้วย คุณสมบัติของวงจรรอนาลอกนั้นจะขึ้นอยู่กับค่าของอุปกรณ์ที่ใช้ เช่น R, C ที่มีค่าไม่แน่นอนที่สำคัญจะมีกระแสรั่วไหลที่แปรตามเวลาและยังขึ้นอยู่กับระดับแรงดันและอุณหภูมิอีกด้วย การออกแบบวงจรรอนาลอกนั้นก็ยังน่าเบื่อเพราะวงจรรหนึ่งๆจะทำงานได้อย่างเดียวเท่านั้นแต่ถ้าต้องการให้ทำงานอื่นๆก็ต้องเปลี่ยนอุปกรณ์และยังต้องเดินสายกันใหม่ นอกจากนี้แล้วการทดสอบวงจรรทางรอนาลอกอาจเป็นปัญหาได้เนื่องจาก ค่าอิมพีแดนซ์ของเครื่องมือที่วัดต่อเข้าไปในวงจรร ปัญหาพื้นฐานของวงจรรอนาลอกเหล่านี้ สามารถแก้ไขได้โดยใช้ดิจิทัลซิกแนลโปรเซสซิ่ง ข้อได้เปรียบอีกประการของการใช้ดิจิทัลซิกแนลโปรเซสซิ่ง แทนวงจรรอนาลอกคือการเปลี่ยนคุณสมบัติของการประมวลผลทำได้โดยง่าย ซึ่งจะทำให้เราสามารถทดลองในทุกทางที่เป็นไปได้เพื่อเลือกเอาผลที่ดีที่สุดที่เราพอใจ

โดยทั่วไประบบดิจิทัลซิกแนลโปรเซสซิ่งจะประกอบด้วยส่วนต่างๆดังรูป



รูปที่ 1.2 ส่วนประกอบของ DSP

สัญญาณอนาลอกอินพุต จะถูกป้อนเข้าสู่ Low Pass Filter เพื่อที่จะจำกัดช่วงความถี่ของสัญญาณอินพุต จากนั้นก็จะส่งต่อไปยังอนาลอก/ดิจิทัลคอนเวอร์เตอร์ (Analog to Digital Converter: A/D) แล้วส่งให้ระบบประมวลสัญญาณดิจิทัลต่อไป ระบบประมวลผลสัญญาณดิจิทัลก็จะนำเอาอัลกอริทึมที่มีอยู่มากกระทำกับข้อมูล จากนั้นก็จะส่งสัญญาณข้อมูลที่ถูกแก้ไขแล้วไปให้ส่วนดิจิทัล/อนาลอกคอนเวอร์เตอร์ (Digital to analog Converter:D/A) เพื่อเปลี่ยนสัญญาณข้อมูลกลับเป็นสัญญาณอนาลอกตามต้องการ แต่จากเอาต์พุตของ D/A ซึ่งจะมีลักษณะเป็นขั้นๆแยกกัน ดังนั้นเราจำเป็นต้องใช้วงจรรกรองสัญญาณให้เรียบ (Low Pass Filter) เพื่อที่จะขจัดสัญญาณความถี่ที่ไม่ต้องการออกไป

หากเราจะมองข้อเสียของดิจิทัลซิกแนลโปรเซสซิ่งก็มี แต่ข้อเสียหรือปัญหาที่เกิดขึ้นแตกต่างจากที่พบในวงจรรอนาลอก เช่น การเกิดควอนไทซ์เซชัน (Quantization), การเกิดสัญญาณรบกวน (Noise), Idle Channel Noise, ความไม่ราบเรียบของสัญญาณ, ความละเอียดของสัญญาณ เป็นต้น ตัวแปรที่สำคัญอีกตัวแปรหนึ่งที่จะแยกระบบดิจิทัลซิกแนลโปรเซสซิ่งต่างๆออกตามการใช้งานคือ ค่าอัตราการสุ่มข้อมูล (Sampling Rate) ยิ่งอัตราสุ่มข้อมูลมากยิ่งต้องใช้ระบบประมวลผลที่มีความเร็วสูงยิ่งขึ้นด้วย อัตราการสุ่มนี้จะมีผลต่อการนำไปใช้งาน ถ้ามีอัตราการสุ่มข้อมูลที่สูงก็จะใช้งานในย่านความถี่ที่สูงได้ โดยคุณภาพของสัญญาณข้อมูลยังคงยอมรับได้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริษัทเทกซ์อินสตุเมนต์ ได้ผลิตไอซีที่เป็นตัวประมวลผลออกมาสำหรับการใช้งานทางด้านของดิจิทัลซิกแนลโปรเซสซึ่งโดยเฉพาะซึ่งเป็นตระกูลที่มีชื่อว่า TMS320 ไอซีตระกูลนี้จะเป็นลักษณะไมโครโปรเซสเซอร์แบบชิพเดี่ยว (Single Chip) สามารถทำงานได้กว้างขวางและมีความเร็วในการประมวลผลสูง

เราจึงนำเอา TMS320C6211 มาใช้ในงานวิเคราะห์สัญญาณแบบตัวกรองความถี่โดยเราจะออกแบบให้สามารถเปลี่ยนค่าความถี่ผ่านและอื่นๆ โดยคอมพิวเตอร์ทำให้สามารถเลือกใช้และเปลี่ยนแปลงค่าความถี่ผ่านได้โดยง่ายทางคอมพิวเตอร์ทั้งยังสามารถปรับปรุงและเปลี่ยนแปลงหรือพัฒนาโปรแกรมเพิ่มเติมได้อย่างง่ายดาย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

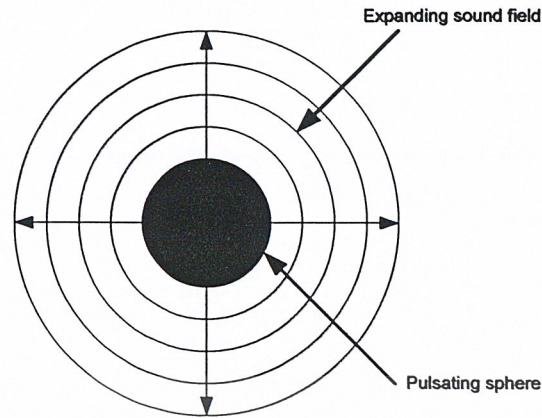
ทฤษฎีและหลักการ

ในบทนี้จะแบ่งเป็นสามส่วนใหญ่ ในส่วนแรกกล่าวถึงความรู้ทั่วไปเกี่ยวกับคุณลักษณะและคุณสมบัติต่างๆ ของเสียงที่ควรทราบ การปรับแต่งลักษณะเสียง ซึ่งประกอบไปด้วย อีควอไลเซอร์, เสียงสะท้อน, เสียงจำลองการสะท้อนของห้องและเสียงคอรัส ส่วนที่สองจะเป็นความรู้พื้นฐานของการประมวลผลสัญญาณดิจิทัล, บทนิยามและทฤษฎีการออกแบบตัวกรองสัญญาณดิจิทัลโดยเน้นไปที่การออกแบบตัวกรองสัญญาณแบบเอฟไออาร์โดยวิธีการใช้วินโดว์แบบต่างๆ และส่วนสุดท้ายเป็นเรื่องของความรู้เบื้องต้นและสถาปัตยกรรมของบอร์ด DSP TMS320C6211, ชุดคำสั่งต่างๆและการโปรแกรม Code Composer Studio เพื่อการใช้งาน

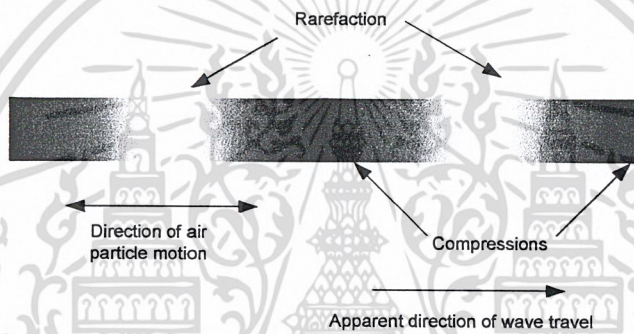
2.1 แหล่งกำเนิดของการสั่นสะเทือน

คลื่นเป็นลักษณะอย่างหนึ่งของการถ่ายทอดพลังงานจากตำแหน่งหนึ่งไปยังอีกตำแหน่งหนึ่ง พลังงานอาจอยู่ในรูปของพลังงานกล พลังงานความร้อน พลังงานแสง พลังงานแม่เหล็กไฟฟ้า หรือ พลังงานเสียง เป็นต้น

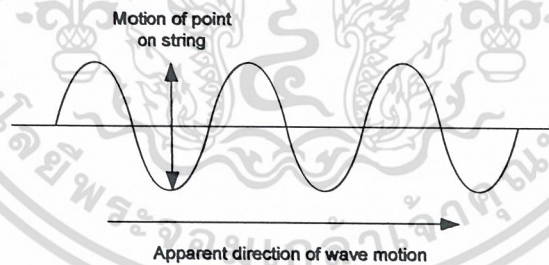
เสียงจะเกิดขึ้นเมื่อวัตถุสั่นสะเทือน เป็นเหตุทำให้อากาศโดยรอบวัตถุเคลื่อนที่ พิจารณารูปทรงกลมดังรูปที่ 2.1 จะเห็นการเคลื่อนที่เข้าออกรอบๆ ทรงกลม ในขณะที่เกิดการเปลี่ยนแปลงนั้นจะเกิดการอัดและขยายเป็นไปอย่างต่อเนื่องเรื่อยๆ จากทรงกลม มีลักษณะเช่นเดียวกับการเกิดคลื่น ทำให้เกิดคลื่นขึ้นนั่นเอง หากโยนก้อนหินลงไปในน้ำ คลื่นน้ำก็จะเกิดขึ้น โดยอาศัยน้ำเป็นตัวกลางทำให้คลื่นเคลื่อนที่ไปข้างๆและแผ่กระจายออกไป โดยที่น้ำกลางสระไม่ได้เคลื่อนที่ตามไปด้วยเพียงแต่กระเพื่อมขึ้นลงเท่านั้น เช่นเดียวกับคลื่นในรูปที่ 2.3 เป็นคลื่นที่เกิดขึ้นในเส้นเชือกที่มีการสั่นสะเทือนในลักษณะที่ทุกๆ อนุภาคของตัวกลางไม่วิ่งตามคลื่น แต่เคลื่อนที่ขึ้นลงในทิศทางที่ตั้งฉากกับทิศทางการเคลื่อนที่ของคลื่น เรียกคลื่นชนิดนี้ว่า คลื่นตามขวาง (Transverse wave) ส่วนคลื่นที่สลับกับคลื่นตามยาวได้แก่ คลื่นที่เกิดขึ้น ในลักษณะที่อนุภาคของตัวกลางเคลื่อนที่ไป-มาในทิศทางขนานกับทิศทางการเคลื่อนที่ของคลื่น เช่น คลื่นในหลอดสปริง เมื่อปลายถูกตรึงเอาไว้ทั้งสองข้าง และบางส่วนของสปริงถูกดึงตามยาวแล้วปล่อยทันที



รูปที่ 2.1 แหล่งกำเนิดเสียงที่เหมือนทรงกลมมีการดันสะเทือนแผ่กระจายคลื่นออกเป็นทรงกลม



รูปที่ 2.2 คลื่นตามยาวที่เกิดขึ้นจากผลการอัด และ ขยายของอากาศ

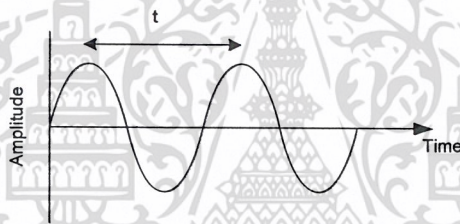


รูปที่ 2.3 ลักษณะคลื่นตามขวาง การเคลื่อนที่ทุกๆจุดของตัวกลางจะเคลื่อนที่ขึ้นลงในทิศทางที่ตั้งฉากกับทิศทางการเคลื่อนที่ของคลื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ลักษณะสมบัติของคลื่นเสียง

ลักษณะการแกว่งของแหล่งกำเนิดทำให้เกิดเป็นความถี่ของคลื่นเสียง มีหน่วยเป็นเฮิรตซ์ (Hz) หรือรอบต่อวินาที (cps) จำนวนของการอัดและขยายของอากาศที่มีผลมาจากการเคลื่อนที่ของทรงกลมคือ วนสูงหรือแอมพลิจูดของคลื่นเสียงที่มีส่วนเกี่ยวข้องกับความดัง (Loudness) ของเสียงที่บ่งบอกถึงความเข้มเสียงที่มาถึงหูมนุษย์ ส่วนระยะทางระหว่างยอดคลื่นที่อยู่ใกล้ชิดกันหรือระยะทางของหนึ่งลูกคลื่น เรียกว่า ความยาว(Wavelength) ของคลื่นเสียง เรียกเป็นภาษากรีกว่า แลมด้า อย่างไรก็ตาม ความยาวคลื่นจะขึ้นอยู่กับความเร็วของคลื่นเสียง เมื่อคลื่นเดินทางด้วยความเร็ว ระยะทางระหว่างยอดคลื่นจะมีค่ามากกว่าคลื่นที่เดินทางช้า โดยให้เวลาคงที่ระหว่างช่วงอัดของยอดคลื่น ดังรูปที่ 2.4 ลักษณะสมบัติของคลื่นเสียงสามารถแทนได้ด้วยกราฟ แกนทางแนวตั้งเป็นค่าแอมพลิจูด แกนทางแนวนอนเป็นเวลา โดยการแทนช่วงอัดของอากาศเป็นบวกและช่วงขยายเป็นลบ สำหรับการเคลื่อนที่เมื่อคลื่นผ่านไปในตัวกลางอนุภาคของตัวกลางจะสั่นสะเทือนด้วยอัตราเดียวกันกับการสั่นสะเทือนของวัตถุที่เป็นต้นกำเนิด การสั่นสะเทือนนั้นเป็นไปอย่างมีจังหวะ คือเป็นไปในรูปแบบซิมเปิลฮาร์โมนิก (Simple Harmonic Motion : S.H.M.) ซึ่งได้รูปคลื่นเสียงที่รู้จักกันโดยทั่วไปคือ รูปคลื่นไซน์ที่ปราศจากฮาร์โมนิกใดๆ



รูปที่ 2.4 รูปคลื่นเสียงมีลักษณะเป็นคลื่นไซน์ โดยคาบของคลื่นแทนด้วย t และความถี่แทนด้วย $1/t$

2.3 เสียงเดินทางในอากาศได้อย่างไร

คลื่นเสียงในอากาศเป็นชนิดของคลื่นตามยาวที่โมเลกุลของอากาศเคลื่อนที่รอบตำแหน่งสมดุลกับจุดกึ่งกลางในแนวที่ขนานกับทิศทางการเคลื่อนที่ของเสียง ดังนั้นเมื่อมีเสียงเดินทางไปในตัวกลางที่เป็นอากาศ จะเกิดการอัดและขยายตัวสลับกันไปดังที่กล่าวมาแล้ว ลักษณะของคลื่นที่มีการสั่นสะเทือนของตัวกลางนี้จะเป็นการสั่นสะเทือนของตัวกลางนี้จะเป็นการสั่นสะเทือนที่มีความยืดหยุ่น ซึ่งเป็นคุณสมบัติของตัวกลางที่เรียกว่า Elastic Property ทั้งนี้หากตัวกลางไม่มีความยืดหยุ่นแล้ว การสั่นสะเทือนของส่วนใดส่วนหนึ่งของตัวกลางย่อมไม่สามารถทำให้ส่วนที่อยู่ถัดไปของตัวกลางนั้นสั่นสะเทือนได้

ในกรณีของคลื่นเสียงในอากาศ ถ้าหากอากาศมีความหนาแน่นสม่ำเสมอ คลื่นเสียงจะมีอัตราเร็วคงที่ประมาณ 340 เมตรต่อวินาทีแต่ความเร็วของเสียงในอากาศที่จุดน้ำเป็นน้ำแข็งความเร็วจะลดลงเหลือประมาณ 330 เมตรต่อวินาที ในเหล็กซึ่งเป็นตัวกลางที่เป็นของแข็งความเร็วของเสียงจะมีค่าประมาณ 5100 เมตรต่อวินาที

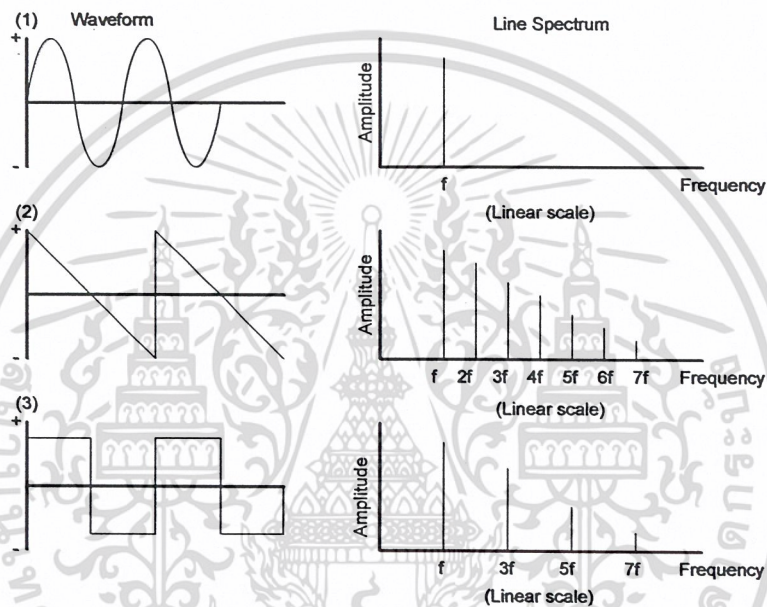
ค่าความถี่และค่าความยาวคลื่นของเสียงจะเกี่ยวข้องโดยตรงกับค่าความเร็วของเสียง (C) จะได้ว่า

$$C = f\lambda \quad (2.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 เสียงอย่างง่ายและเสียงเชิงซ้อน

เสียงจะมีรูปคลื่นเป็นรูปไซน์ ซึ่งเป็นผลมาจากระบบการสั่นสะเทือนอย่างง่ายของน้ำหนักที่ผูกติดไว้ที่ปลายสปริง คลื่นรูปไซน์ที่มีเสียงบริสุทธิ์มากที่สุด จะประกอบด้วยพลังงานที่มีความถี่เดียว (Fundamental) ปราศจากฮาร์โมนิกหรือเสียงทุ้มแหลมใดๆ จึงมักเรียกคลื่นรูปไซน์นี้ว่าคลื่นที่มีเสียงทุ้มแหลมที่บริสุทธิ์ (Pure tone) แต่ถ้าหากแหล่งกำเนิดเสียงถูกผสมผสานกับความถี่แปลกปลอม จะทำให้เกิดเป็นรูปคลื่นเชิงซ้อนหรือเสียงรบกวน (Noise) ก็จะมีเกิดขึ้น เช่นเสียงรบกวนที่เกิดขึ้นในเครื่องบันทึกเสียงดนตรีต่างๆหรือเสียงฮัม เป็นต้น



รูปที่ 2.5 เงามันสมมูลสำหรับการเลือกรูปคลื่นอย่างง่าย

- (1) คลื่นรูปไซน์ ประกอบด้วยส่วนประกอบเดียวที่ความถี่เดียวหรือความถี่มูลฐาน
- (2) คลื่นรูปฟันเลื่อย ประกอบด้วยความถี่มูลฐานและผลคูณเลขจำนวนเต็มกับค่าความถี่มูลฐาน โดยมีค่าของแอมพลิจูดลดลงอย่างสม่ำเสมอ
- (3) คลื่นรูปสี่เหลี่ยม ประกอบด้วยส่วนประกอบที่ผลคูณเลขคี่ของความถี่มูลฐาน

ลักษณะสำคัญของเสียงที่มีความทุ้มแหลมของเสียง (Pitch) ที่แน่นอนจะไม่ใช่เป็นรูปคลื่นเชิงซ้อน ซึ่งความทุ้มแหลมของเสียงนี้ก็คือความถี่นั่นเอง ความถี่ในการรับฟังที่ตอบสนองได้ไวที่สุดคือประมาณ 3000 Hz สำหรับรูปที่ 2.5 แสดงสเปกตรัมของแถบความถี่ตามแนวนอนและแทนค่าแอมพลิจูดในแนวตั้ง ซึ่งบ่งบอกความสัมพันธ์ต่อกันของค่าความถี่ที่แตกต่างกับความแตกต่างของค่าแอมพลิจูด ดังรูปที่ 2.5(2) จะเห็นว่าค่าความถี่ $2f, 3f$ ต่างเป็นค่าความถี่เชิงซ้อนทั้งสิ้น โดยเรียกความถี่ $2f$ ว่าฮาร์โมนิกที่ 2

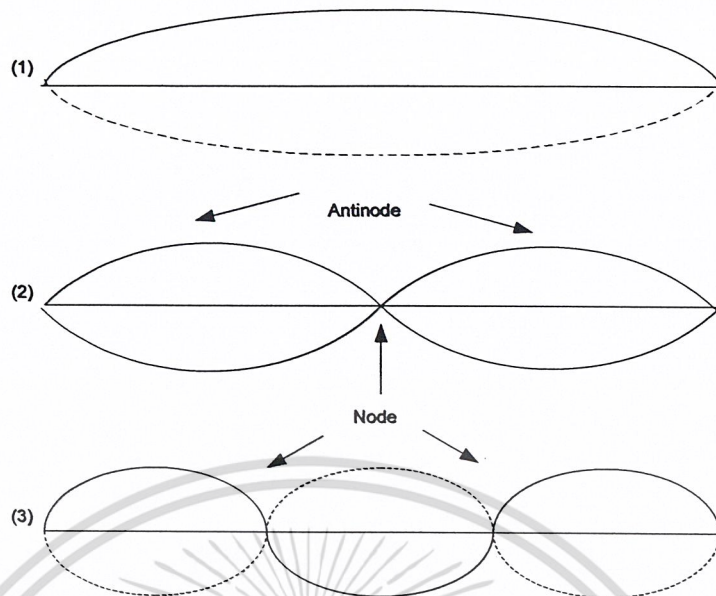
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือ First overtone และความถี่ $3f$ ว่าฮาร์โมนิกที่ 3 หรือ Second overtone หากวงจรขยายมีการขยาย สัญญาณผิดรูปคลื่นไปจากเดิมทำให้เกิดฮาร์โมนิกเพิ่มขึ้น เป็นผลทำให้เสียงที่ได้ผิดไปจากเดิม ซึ่งความถี่ฮาร์โมนิกจะเกิดขึ้นมากหรือน้อยขึ้นอยู่กับเปอร์เซ็นต์ความเพี้ยน

2.5 ความถี่สเปกตรัมของเสียงซำ

ลักษณะของคลื่นอย่างง่าย เส้นสเปกตรัมมีส่วนประกอบเดียวที่ความถี่ของรูปคลื่น ไซน์เป็ความถี่มูลฐานในการออสซิลเลต ส่วนรูปคลื่นที่เกิดขึ้นซ้ำ เช่น รูปคลื่นสี่เหลี่ยมจะมีความถี่มูลฐานเช่นเดียวกัน และประกอบด้วยความถี่ที่สูงกว่าความถี่มูลฐานค่าต่างๆกัน โดยเรียกความถี่ที่สูงกว่านี้ว่า ความถี่ฮาร์โมนิก

ฮาร์โมนิกประกอบด้วยเสียงที่มีความถี่เป็นทวีคูณของความถี่เดิมเป็นสองเท่า สามเท่าและสี่เท่า เป็นต้น เช่นเสียงที่มีความถี่เดิมเท่ากับ 100 Hz ประกอบด้วยฮาร์โมนิก 200 Hz 400 Hz และ 600 Hz จากรูปที่ 2.6 ความถี่ฮาร์โมนิกที่เกิดขึ้น เป็นการสั่นสะเทือนตามจำนวนของโน้ตที่สอดคล้องกับผลคูณเลขจำนวนเต็มของความถี่เดิมของการสั่นสะเทือนของเส้นเชือก ความถี่จะสอดคล้องกับโน้ตของเส้นเชือกที่เคลื่อนที่ขึ้นและลงทั้งหมด ด้วยเหตุที่ฮาร์โมนิกสอดคล้องกับโน้ตตามรูปแบบของการสั่นสะเทือน คือ แบ่งความถี่ฮาร์โมนิกตามค่าสูงสุดและต่ำสุดของการเคลื่อนที่ตามเส้นเชือก สามารถมองเห็นโน้ตที่ 2 ทำให้เกิดเป็นค่ายอดสองค่าของการสั่นสะเทือนและโน้ตที่ 3 เกิดเป็นค่ายอดสามค่าตามลำดับจากรูปที่ 2.6 ความถี่มูลฐานจะมีค่าความถี่สูงสุด แต่แอมพลิจูดของความถี่ฮาร์โมนิกจะลดลงเมื่อความถี่สูงขึ้น แต่ไม่เสมอไป เช่น เสียงจากเครื่องดนตรีชนิดต่างๆ คือ เปียโน, ฉิ่ง, กลอง เป็นลักษณะของเสียงทั่วไปที่มักจะเกิดจากคลื่นความถี่ต่างๆผสมกันอย่างซับซ้อน แต่สามารถแยกได้เป็นคลื่นมูลฐานหรือคลื่นฮาร์โมนิกต่างๆ แล้วประกอบเข้าด้วยกันทำให้เกิดเป็นเสียงดนตรีที่มีความถี่แตกต่างกันโดยเส้นสเปกตรัมมีความถี่มูลฐานเดียวกัน ทั้งนี้ความถี่ที่สูงขึ้นระหว่างเครื่องดนตรีบางชิ้น อาจมีค่าแอมพลิจูด ลดลงหรือสูงขึ้น สำหรับการได้ยินเสียงความถี่มูลฐานหรือความถี่เดิมจะได้ยินชัดที่สุด ดังนั้นเสียงที่เกิดจากคลื่นความถี่ต่างๆ จะผสมกันอย่างซับซ้อน แต่ยังสามารถแยกได้ว่าเป็นเสียงของเครื่องดนตรีชนิดใด เสียงแตกต่างกันอย่างไร มีชื่อเรียกว่าอะไรและมีคุณภาพของเสียงเป็นอย่างไร



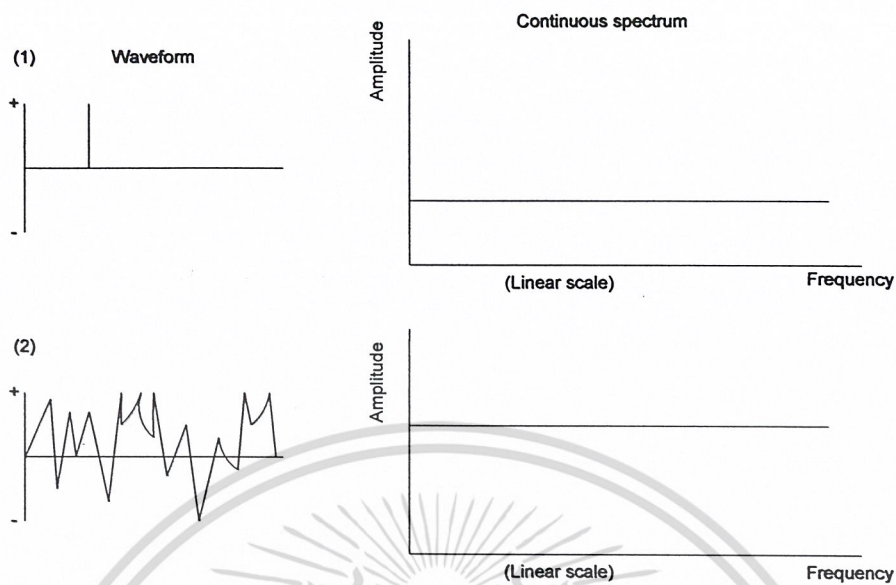
รูปที่ 2.6 โหมดการสั่นสะเทือนของเชือก

- (1) ความถี่มูลฐาน
- (2) ฮาร์โมนิกที่ 2 แอนติโนดมีแอมพลิจูดสูงสุด
- (3) ฮาร์โมนิกที่ 3 โหนดแอมพลิจูดจะเป็นศูนย์

2.6 ความถี่สเปกตรัมของเสียงไม่ซ้ำ

ลักษณะของรูปคลื่นที่ไม่เกิดซ้ำหรือคลื่นเดิม จะไม่มีความท่วมแหลมของเสียงเกิดขึ้นเป็นเหตุให้สัญญาณรบกวนของเสียงมักปะปนมากับความถี่เดิม ทำให้คุณภาพของเสียงตก ความผิดเพี้ยนเกิดขึ้น เพราะเสียงรบกวนจากแหล่งกำเนิดต่างๆที่ป้อนเข้าวงจรขยายเสียง รวมทั้งอุปกรณ์ประกอบภายในวงจรสัญญาณรบกวนทางอุณหภูมิที่เป็นคุณสมบัติประจำตัวของอุปกรณ์นั้นๆ ตลอดทั้งการเดินสายวงจรนั้น นอกจากนี้ยังมีความถี่ออสซิลเลตที่มีผลต่อวงจรขยาย หากเกิดออสซิลเลตนานๆ โดยไม่ปิดสวิตช์เครื่องลำโพงและอุปกรณ์ภายในเครื่องอาจเสียหายได้ จากรูปที่ 2.7 รูปคลื่นที่ปรากฏ จะมีเสียงแปลกปลอมเป็นสัญญาณรบกวนที่ประกอบด้วยความถี่หลายความถี่มีพลังงานคงที่ตลอดย่านความถี่สเปกตรัม เมื่อคิดค่าเฉลี่ยแถบของความถี่จะเป็นเส้นตรง จึงเรียกว่าสัญญาณรบกวนขาว (White noise) และหากความถี่หลายความถี่ที่กล่าวมาแล้วมีความแรงของสัญญาณที่ปรากฏเท่าๆกัน ความถี่สเปกตรัมจะเป็นเส้นตรงตลอดย่านความถี่ แต่ถ้าค่าของความถี่สูงขึ้น สัญญาณของพลังงานจะลดลง เรียกว่าสัญญาณรบกวนชมพู (Pink noise) มีค่าเท่ากับพลังงานต่อออกเตฟ (Octave)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ค่าความถี่สเปกตรัมของรูปคลื่นที่ไม่ซ้ำ

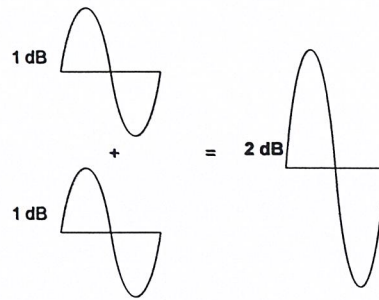
- (1) รูปพัลส์
- (2) สัญญาณรบกวน

2.7 เฟส

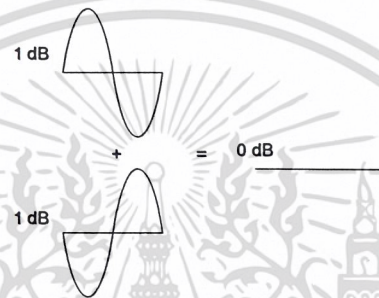
เฟสคือตำแหน่งมุมของคลื่นเสียงที่มากระทบเยื่อแก้วหูของมนุษย์ เมื่อคลื่นเสียงเคลื่อนที่ครบรอบ คลื่นเสียงจะมีมุมรวม 360 องศา ถ้าตำแหน่งของมุมตกกระทบแตกต่างกัน รูปร่างที่ได้ของคลื่นก็จะแตกต่างกัน หากคลื่นเสียงที่ตกกระทบมีมากกว่าหนึ่งสัญญาณ จะมีผลต่อความดังของเสียงที่ได้ยินแตกต่างกัน ผลความดังของเสียงมีค่าโดยตรงกับเฟสของคลื่นเสียงเหล่านั้นที่มาตกกระทบเพราะสัญญาณเสียงที่ส่งหลายสัญญาณเหล่านั้นสามารถรวมกันได้ ซึ่งการรวมกันบางครั้งอาจเสริมกันหรือบางครั้งอาจหักล้างกัน

จากรูปที่ 2.8 และ 2.9 แสดงการรวมกันของคลื่นเสียง 2 สัญญาณ ที่มีความดังของเสียงสัญญาณละ 1 dB ในรูป 2.8 คลื่นเสียงทั้งสองมีเฟสเหมือนกัน ผลของการรวมกันทำให้คลื่นเสียงที่ได้ออกมามีระดับความดังเพิ่มขึ้นเป็นเท่าตัว เป็น 2dB แต่ในรูปที่ 2.9 คลื่นเสียงทั้ง 2 มีเฟสตรงข้ามกัน มีผลทำให้การรวมกันเป็นแบบหักล้างกัน ทำให้คลื่นเสียงหักล้างกันหมด จะไม่ได้ยินคลื่นดังออกมาหรือระดับความดังลดลงมากๆ

กรณีที่คลื่นเสียงรวมกันมีเฟสต่างกัน ผลการรวมกันทำให้บางช่วงหรือบางตำแหน่งของเฟสสัญญาณเสริมกันและบางช่วงบางตำแหน่งสัญญาณจะหักล้างกัน คลื่นเสียงที่ได้ออกมาจะมีรูปร่างคลื่นและตำแหน่งเฟสที่เปลี่ยนไป เสียงที่ดังออกมาอาจผิดเพี้ยนไปจากเสียงที่ต้นกำเนิดที่ส่งมา การรวมกันของคลื่นเสียงมีเฟสต่างกัน



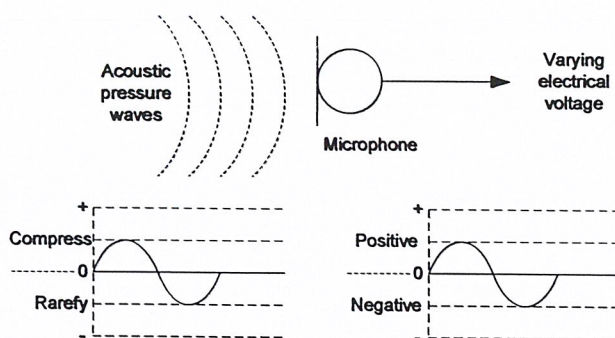
รูปที่ 2.8 การรวมกันของคลื่นเสียง 2 สัญญาณ ที่มีเฟสเหมือนกัน



รูปที่ 2.9 การรวมกันของคลื่นเสียง 2 สัญญาณ ที่มีเฟสตรงข้ามกัน

2.8 เสียงในรูปลักษณะทางไฟฟ้า

เสียงที่ได้ยินเกิดจากการสั่นสะเทือนของวัตถุ ขณะที่เสียงเดินทางไปในตัวกลาง อนุภาคของอากาศจะเกิดการอัดตัวและบีบตัว แรงกดดันกล่าวเมื่อกระทบกับไมโครโฟน ส่งผลให้ไมโครโฟนทำหน้าที่เปลี่ยนคลื่นเสียงให้เป็นพลังงานไฟฟ้า โดยนำเสียงที่ได้ไปขยายบันทึกและผสมกับสัญญาณอื่นๆ เป็นต้น สำหรับแอมพลิจูดของสัญญาณที่เกี่ยวกับเสียงในสัญญาณไฟฟ้าหมายถึงค่าแรงดันไฟฟ้านั่นเอง จากรูปที่ 2.10 คลื่นรูปไซน์ที่ได้เกิดจากการอัดตัวและขยายตัวของอากาศ สัญญาณครึ่งบวกเกิดจากการอัด และในช่วงจังหวะขยายตัวของอากาศ จะได้สัญญาณครึ่งทางลบสลับกันไปเรื่อยๆ จึงเป็นเหตุให้เกิดค่ากระแสสลับหรือ AC ขึ้น ลักษณะของการเปลี่ยนแปลงทางไฟฟ้า ที่เกิดจากสัญญาณเสียงจะนำไปใช้งานต่อไป



รูปที่ 2.10 ไมโครโฟนทำหน้าที่เปลี่ยนความกดดันของอากาศที่เกิดขึ้นจากเสียงต่างๆ ให้เป็นการเปลี่ยนแปลงในรูปของแรงดันไฟฟ้า ซึ่งการอัดตัวของอากาศจะได้ค่าแรงดันไฟฟ้าบวกและการขยายตัวของอากาศจะได้ค่าแรงดันไฟฟ้าลบ

ในระบบ AC ค่าความต้านทานจะถูกแทนด้วยค่าอิมพีแดนซ์ที่ประกอบด้วยค่าความต้านทานและค่ารีแอกแตนซ์ โครรีแอกแตนซ์จะเปลี่ยนแปลงตามค่าความถี่ของสัญญาณ ดังนั้นค่าอิมพีแดนซ์ของอุปกรณ์ทางไฟฟ้าก็จะเปลี่ยนแปลงตามค่าความถี่ของสัญญาณด้วย ตัวเก็บประจุเป็นอุปกรณ์ทางไฟฟ้าที่มีค่าอิมพีแดนซ์สูงที่สัญญาณความถี่ต่ำและค่าอิมพีแดนซ์ต่ำที่สัญญาณความถี่สูง ทั้งนี้จะไม่ให้กระแสไฟตรงไหลผ่าน สำหรับตัวเหนี่ยวนำก็เป็นอุปกรณ์ทางไฟฟ้าชนิดหนึ่งที่มีอิมพีแดนซ์มีค่าสูงกับสัญญาณความถี่สูงและอิมพีแดนซ์มีค่าต่ำกับสัญญาณความถี่ต่ำ ดังนั้นตัวเก็บประจุและตัวเหนี่ยวนำจะมีผลต่อค่าความถี่หรือ AC นั้นเอง

2.9 เดซิเบล

เดซิเบล (Decibel) หรือ dB เป็นหน่วยพื้นฐานที่ใช้แพร่หลายในวิศวกรรมระบบเสียงจะพบเห็นอยู่บ่อยๆ เช่นเดียวกับหน่วยของโวลต์หรือวัตต์และใช้ในการวัดระดับสัญญาณเสียงโดยมีความสัมพันธ์กับลอการิทึม (Logarithm) ซึ่งวิศวกรโทรศัพท์เรียกหน่วยนี้ว่า ลอการิทึมของอัตราส่วนระหว่างกำลังเสียงสองสัญญาณและตั้งหน่วยพื้นฐานของระบบเป็นเบล (Bel) เพื่อให้เกียรติแก่เอเล็กซานเดอร์ เกรแฮม เบล ผู้คิดประดิษฐ์โทรศัพท์ขึ้นเป็นคนแรก

$$\text{จะได้ค่าเบลดังนี้} \quad \text{Bel} = \log (P_1/P_2) \quad (2.2)$$

$$\text{dB} = 10 \log (P_1/P_2) \quad (2.3)$$

จากหน่วยวัด dB จะกำหนดเสียงที่หูมนุษย์ฟังได้ยินสูงสุดโดยไม่เป็นอันตรายต่อหู คืออยู่ในช่วง 120dB ลงมาส่วนเสียงมนุษย์ฟังได้ยินเบาที่สุดคือ 0dB ถ้าหากต่ำกว่า 0dB จะฟังไม่ได้ยิน นอกจากนี้ระดับความดังของเสียง SPL (Sound Pressure Levels) ใช้ค่าวัดเป็น dB อ้างอิงกับความดันที่ $2 \times 10^{-5} \text{ Nm}^{-2}$ (0.0002 ไมโครบาร์ หรือ 0.0002 dyne/cm^2) ที่ความถี่มาตรฐาน 1000 Hz

$$\text{SLP} = 20 \log (P_1/P_2) \quad (2.4)$$

เมื่อ $\text{SLP} =$ ระดับความดังของเสียง (dB)

$P_1 =$ ความดังของเสียงใดๆ (Nm^{-2})

$P_2 =$ ความดังของเสียงที่ระดับเปรียบเทียบ (Nm^{-2})

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แหล่งกำเนิดเสียง	ระดับความดัง (dB)
เสียงเบาที่สุดที่มนุษย์ได้ยิน	0
เสียงลมหายใจ	10
เสียงใบไม้กระทบกันเมื่อลนพัด	20
เสียงในสำนักงานเงียบๆ	30
เสียงนกร้องระยะทางไกลๆ	40
เสียงสำนักงานที่เปิดประตูทิ้งไว้	50
เสียงสนทนาห่างกัน 3 ฟุต	60
เสียงสนทนาห่างกัน 1 ฟุต	70
เสียงวงดนตรีออเครสตราในระดับเฉลี่ย	80
เสียงพัดลมระบายอากาศขนาดใหญ่	90
เสียงเลื่อยยนต์	100
เสียงวงดนตรีคอนเสิร์ตแนวร็อก	110
เสียงเครื่องบินไอพ่นบินขึ้นในระยะ 1500 ฟุต	120
หูมนุษย์เริ่มได้รับอันตรายจากการได้ยิน	130-140
จุดที่ทำให้หูหนวกเฉียบพลัน	150-160

ตารางที่ 2.1 ระดับความดังของเสียงจากแหล่งกำเนิดเสียง

2.10 เสียงกับการได้ยิน

กลไกการได้ยินของเสียงคือหูของมนุษย์นั่นเองหูจะทำหน้าที่รับความถี่ของคลื่นเสียงในระดับต่างๆ โดยระบบประสาทหูจะเปลี่ยนคลื่นเสียงเหล่านั้นให้เป็นสัญญาณผ่านเซลล์ไปยังสมอง ซึ่งสมองจะทำหน้าที่แปลสัญญาณให้เป็นการรับรู้ต่อไป ส่วนการรับรู้เกี่ยวกับเสียงหรือการได้ยินจะชัดเจนสมบูรณ์อย่างไรมันขึ้นอยู่กับแหล่งกำเนิดเสียงและการทำงานของหูมนุษย์

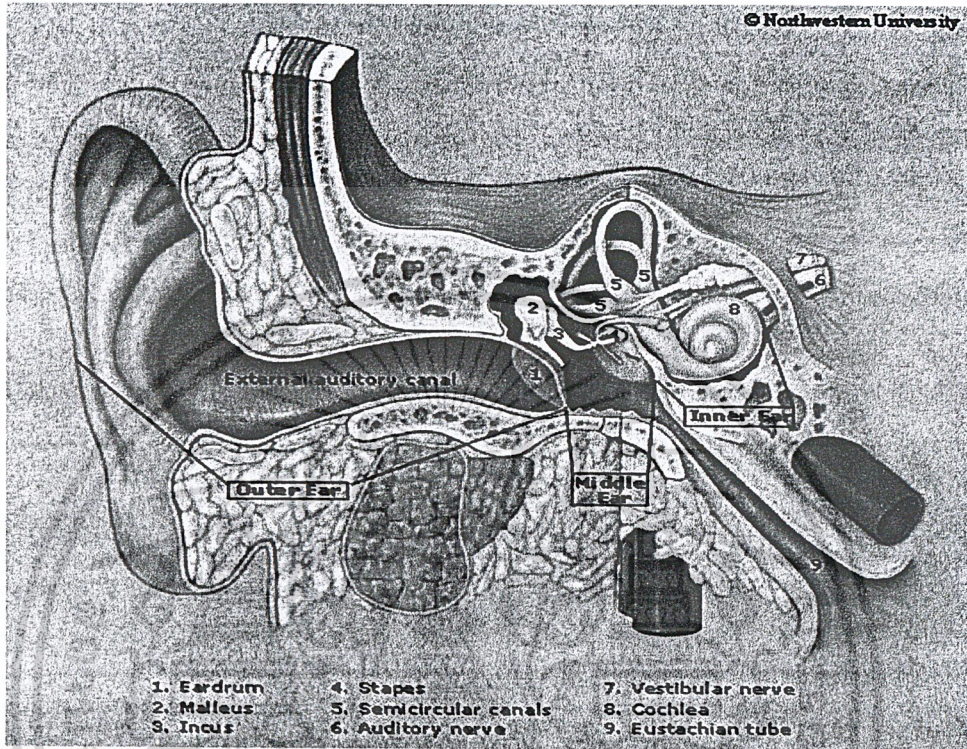
2.11 กลไกของการรับฟังเสียง

แม้ว่าจะไม่ใช่การศึกษาด้านสรีรศาสตร์โดยตรงแต่ก็มีความจำเป็นต้องเรียนรู้ลักษณะและส่วนประกอบของหู ดังรูปที่ 2.11 เป็นแผนผังโครงสร้างภายในหูที่ประกอบด้วยกลไกที่สำคัญ 3 ส่วนคือหูส่วนนอก(External ear) ประกอบด้วย ใบหู(Pinna), รูหู(External auditory canal)และเยื่อแก้วหู(Tympanum) ซึ่งเป็นเยื่อบางๆ ต่อจากเยื่อแก้วหูคือ หูส่วนกลาง(Middle ear) ประกอบด้วยกระดูกรูปก้อน(Malleus) ,กระดูกรูปทั่ง(Incus)และกระดูกรูปโกลน(Stapes) ซึ่งทำหน้าที่สำคัญคือ รับพลังงานต้นสะเทือนที่ส่งมาจากเยื่อแก้วหู แล้วขยายสัญญาณเสียงที่ต้นสะเทือนผ่านเข้าไปยังหูส่วนใน(Inner ear) ซึ่งเริ่มจากเนื้อเยื่อบางๆ เรียกว่า โอวอลวินโดว์ (Oval window) หรือช่องรูปไข่ซึ่งติดกับกระดูกรูปโกลน

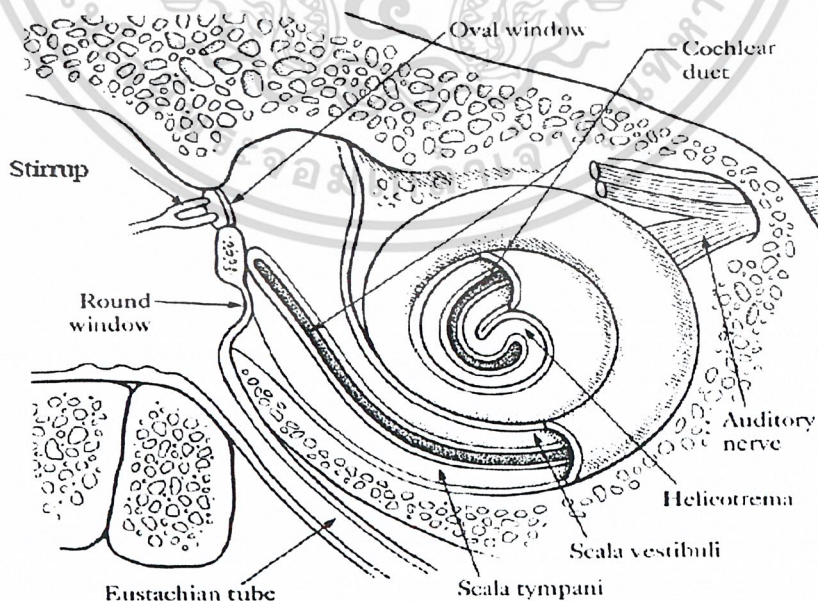
ด้านในติดกับท่อรูปคล้ายหอยโข่งมีของไหลอยู่ภายในเรียกว่า คอเคลีย (Cochlea) และแผ่นบาซิลาร์เมมเบรน (Basilar membrane) ที่กั้นภายในคอเคลียช่วงปลายสุดของเยื่อบางบาซิลาร์เมมเบรนห่างจากหู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนกลาง โดยมีช่องว่างเล็กๆเรียกว่าเฮลิโคทริมา (Helicotrema) ที่ปล่อยให้ของเหลวไหลผ่านจากช่องที่สูงกว่าไปยังช่องที่ต่ำกว่า ทั้งนี้ภายในคอเคลียมีเนื้อเยื่อประกอบไปด้วยประสาทรับเสียงและนำสัญญาณเสียงที่ได้ส่งต่อไปยังสมอง โดยสมองจะทำหน้าที่แปลงสัญญาณให้เป็นการรับรู้เกี่ยวกับการได้ยินของเสียงต่อไป



รูปที่ 2.11(a) แผนภาพกลไกของหู

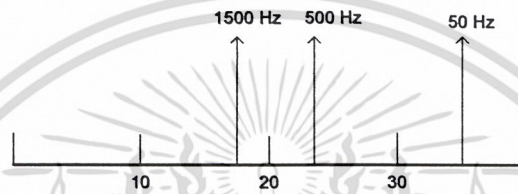


รูปที่ 2.11(b) แผนภาพกลไกของหู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 ความถี่กับการได้ยิน

การเคลื่อนที่ของบาซิลาร์เมมเบรน ขึ้นอยู่กับความถี่ของคลื่นเสียง นั่นคือส่วนของคลื่นเสียง จะเคลื่อนที่เข้าใกล้ช่องรูปไข่ ความดันของคลื่นเสียงจะถูกส่งผ่านต่อเข้าไปในของไหล Cochlea fluid ได้ ความดันเท่าเทียมกันทุกจุด แต่ถ้ากระดูกรูปโกลนสั้นและส่งผ่านความดันให้ช่องรูปไข่ ซึ่งความดันนี้จะ เป็นผลให้หน้าต่างกลม (Round window) ที่อยู่ล่างช่องรูปไข่ โป่งออกโดยที่ปริมาตรของไหลยังคงเท่าเดิม ทำให้ของไหลในคอเคลียเกิดสั่นปลายประสาทหรือเซลล์ขน (Hair cells) จากนั้นเซลล์ขนก็จะทำหน้าที่ เปลี่ยนคลื่นเสียงให้เป็นอิมพัลส์ไฟฟ้าส่งไปตามโศตประสาทถึงสมอง สมองก็จะแปลสัญญาณออกมา ทำให้เกิดการได้ยินเป็นคำพูด หรืออื่นๆ เป็นต้น



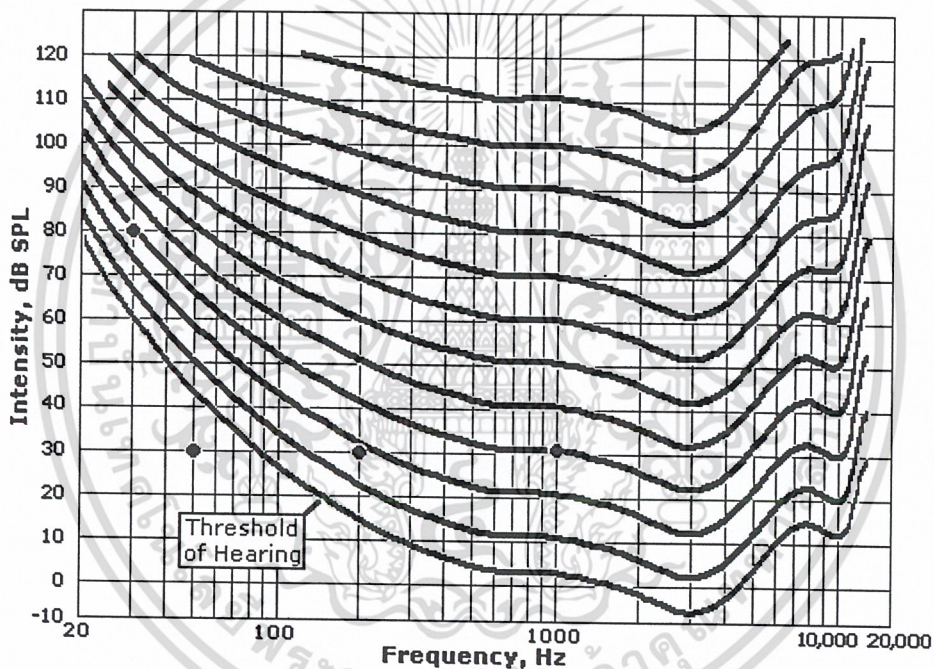
รูปที่ 2.12 ตำแหน่งสันสะท้อนสูงสุดของเยื่อบาซิลาร์เมมเบรนเคลื่อนที่ไปทางช่องรูปไข่โอวอลวินโดว์ เช่นเดียวกับการเพิ่มขึ้นความถี่

จากรูปที่ 2.12 เยื่อบาซิลาร์เมมเบรน โดยเฉลี่ยมีความยาวประมาณ 3.5 เซนติเมตร ส่วนเซลล์ขนมีประมาณ 30,000 เส้น เมื่อพิจารณาความถี่เสียงกับการได้ยินของมนุษย์แล้ว ไซว่ามนุษย์จะได้อินคลื่นเสียงได้ทุกชนิด เพราะที่ความถี่ต่ำมากๆ ประมาณ 30 Hz ลงไปมนุษย์จะไม่ได้ยิน แต่รับรู้ได้จากที่คลื่นเสียงมากระทบประสาทหู จึงจำเป็นต้องใช้วงจรขยายเสียงเข้าช่วย สำหรับความถี่กลางจะมีแถบความถี่คาบเกี่ยวกันประมาณ 200 Hz ขึ้นไปจนถึง 4000 Hz ส่วนความถี่สูงกว่านี้มักจะฟังไม่ได้ยิน โดยพิจารณาได้จากเสียงเครื่องดนตรีชนิดต่างๆ

การเลือกความถี่ของมนุษย์อาจกล่าวได้ว่า มนุษย์จัดเป็นตัวกรองความถี่ชนิดหนึ่งที่สามารถเลือกความถี่เสียงได้ ถ้าเสียง 2 เสียงมีความถี่ต่างกันเล็กน้อย เช่น สัญญาณความถี่ 200 Hz และ 201 Hz ดังขึ้นพร้อมกัน ความดังรวมที่ได้จะไม่สม่ำเสมอ กล่าวคือจะมีจังหวะที่เสียงดังและไม่ดังสลับกันไปปรากฏการณ์เป็นการสอดแทรกของคลื่นอย่างหนึ่ง ซึ่งทำให้เกิดการบีต (Beat) ขึ้น เนื่องจากเสียง 2 เสียงที่มาถึงเยื่อแก้วหูซึ่งเป็นเยื่อบางๆ จะเกิดการสั่นสะเทือนและส่งสัญญาณการสั่นสะเทือนเข้าไปยังหูส่วนกลางและหูส่วนใน ผลที่ได้จะเป็นการรวมของคลื่น จุดใดที่คลื่นมีเฟสเดียวกันเสียงก็จะดัง ตรงกันข้ามจุดใดที่คลื่นมีเฟสต่างกันเสียงก็จะค่อย ความถี่ของการบีตเรียกว่า Beat Frequency มีความถี่เท่ากับผลต่างคือ 201-200 Hz เท่ากับ 1 Hz หูจะได้ยินการบีต 1 ครั้งต่อวินาที แต่ถ้าหากมีความแตกต่างมากมนุษย์ก็จะรับฟังไม่รู้เรื่อง เช่น ความถี่ของการบีตเกิน 10 Hz เป็นต้น

2.13 ความดังของเสียงกับการได้ยิน

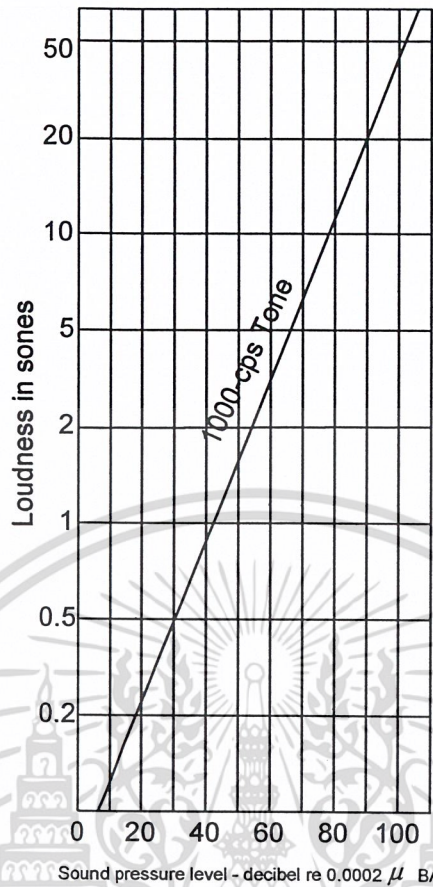
ความดังของเสียงเป็นการวัดเสียงจากความรู้สึกรับรู้ของการได้ยินและมีความสัมพันธ์กันระดับความเข้มของเสียงและความถี่ของเสียง โดยหน่วยวัดความดังของเสียงคือ ฟอน(Phon) ณ ความถี่ของเสียง 1000 Hz ถ้าระดับความเข้มเป็น 0 dB หมายถึงเสียงที่พอจะได้ยินเรียกว่า Threshold of Hearing ตรงกับความดัง 0 ฟอน หากต่ำกว่าระดับนี้หูจะฟังไม่ได้ยิน และถ้าระดับเสียงดังเป็น 140 ฟอน เรียกว่า Threshold of pain เสียงจะดังมากจนเจ็บหูและปวดศีรษะ ดังรูปที่ 2.13 เป็นกราฟแสดงความสัมพันธ์ระหว่างระดับความเข้มของเสียง (แกนตั้งสเกลธรรมดา) ใช้อิงความดันที่ $2 \times 10^{-5} \text{ Nm}^{-2}$ กับความถี่เสียงที่มนุษย์ฟังได้ (ใช้แกนอนโดยแบ่งสเกลเป็นลอการิทึม) เส้นกราฟแต่ละเส้นจะบอกให้เราทราบว่า บนเส้นกราฟนั้นๆเสียงมีความดังเท่ากัน จึงเรียกรูปที่ได้นี้ว่ากราฟแสดงความดังของเสียงในหน่วยที่เป็นฟอน ซึ่งตัวเลขที่แสดงความดังของเสียงเป็นฟอน จะเป็นตัวเลขเดียวกับที่บอกระดับความเข้มของเสียง เมื่อความถี่เป็น 1000 Hz



รูปที่ 2.13 กราฟความดังของเสียงในหน่วยที่เป็นฟอน

เนื่องจากฟอนเป็นหน่วยความดังของเสียงที่ไม่เป็นสัดส่วน กล่าวคือ เสียง 10 ฟอนไม่ดังเป็น 2 เท่าของเสียง 5 ฟอน จึงมีหน่วยความดังของเสียงอีกหน่วยเรียกว่า โซน (Sone) ซึ่งเป็นหน่วยความดังของเสียงที่เป็นสัดส่วนคือเสียง 2x โซน จะดังเป็น 2 เท่าของเสียง x โซน หน่วยความดังของเสียงฟอนและโซนมีความสัมพันธ์กัน โดยค่าความดังของเสียง 1 โซน ประมาณเท่ากับ 40 ฟอน หากความดังในหน่วยฟอนเพิ่มขึ้น 10 ฟอน คือเป็น 50 ฟอนความดังในหน่วยโซนจะเพิ่มขึ้นเป็น 2 เท่าเสมอไป คือเป็น 2 โซน ในทางตรงกันข้าม ถ้าความดังในหน่วยฟอนลดลง จากเดิม 10 ฟอน ความดังในหน่วยโซนจะลดลงครึ่งหนึ่งของค่าเดิม ดังรูปที่ 2.14 ความดังของเสียงบนเส้นกราฟจะมีลักษณะเป็นสเกลเชิงเส้น

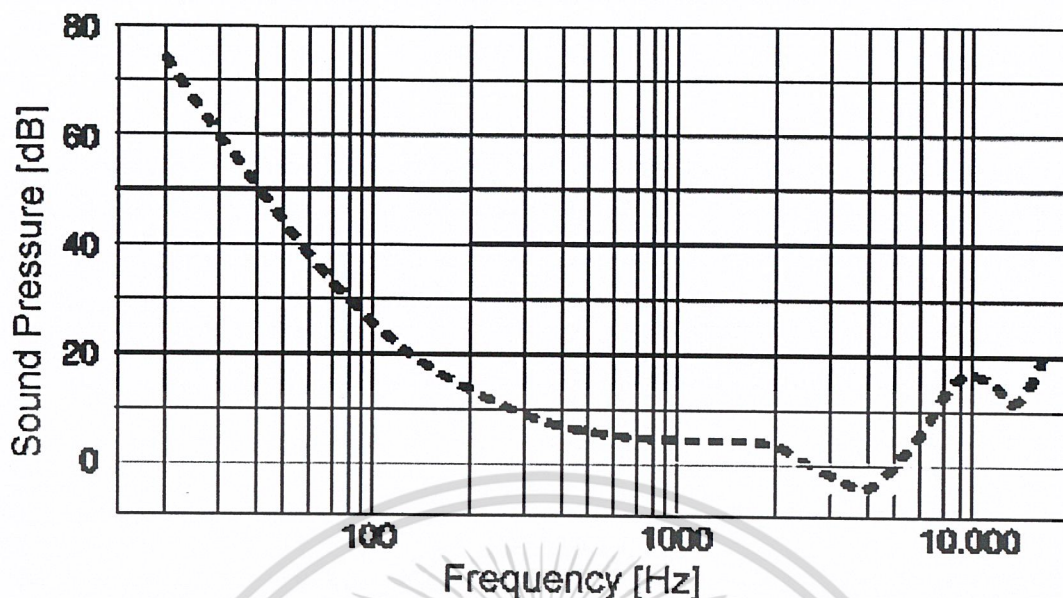
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 กราฟความสัมพันธ์ความดังของเสียงในหน่วยฟอนและหน่วยโซน

หูมนุษย์จะรับฟังเสียงในช่วงความดังที่น้อยที่สุด 0 ฟอนจนถึงความดังที่สุด 140 ฟอน เป็นย่านการเปลี่ยนแปลงความดัง (Dynamic range)

อีกตัวประกอบหนึ่งที่สำคัญ ณ จุดความดังของเสียงที่พอจะฟังได้ยิน (Threshold of hearing) ถูกยกขึ้นไปให้สูงที่ความถี่หนึ่ง โดยเฉพาะกับอีกเสียงหนึ่งที่มีความถี่คล้ายคลึงกัน โดยเสียงหนึ่งจะถูก Masking (ปิดบังหรือกลบ) ซึ่งอาจจะกล่าวได้ว่า ขณะที่รับฟังเสียงอยู่จะมีอีกเสียงหนึ่งค่อยๆ มีความดังมากขึ้น จึงทำให้ไม่ได้ยินเสียงแรก เพราะว่าเสียงที่สองมีความดังมากกว่าแล้วจะปิดบังหรือกลบเสียงแรก



รูปที่ 2.15 กราฟขนาด Masking tone

2.14 แบนด์วิดท์เสียง

แบนด์วิดท์เสียง หมายถึงแถบความกว้างของสัญญาณความถี่ อุปกรณ์ที่มีต่อระบบส่งผ่านสัญญาณ กล่าวโดยสรุปจะได้ว่า แบนด์วิดท์คือระยะห่างระหว่างความถี่สูงสุดกับความถี่ต่ำสุดของแบนด์วิดท์ที่ระบุไว้ ตัวอย่างเช่น มนุษย์จะมีแบนด์วิดท์ของการได้ยินอยู่ระหว่าง 20 Hz ถึง 20 kHz และมีความกว้างของย่านความถี่เป็น 19980 Hz ซึ่ง IEC (International Electrotechnical Commission) ได้กำหนดช่วงความถี่เสียงเป็น 22.4 Hz ถึง 22.4 kHz ส่วนการวิเคราะห์ย่านความถี่เสียงจะเป็นการวิเคราะห์การวัดเสียงรบกวนเพราะหากเสียงรบกวนเกิดขึ้นในย่านความถี่เสียงอย่างสม่ำเสมอ จะเป็นผลทำให้การวัดระดับเสียงเปลี่ยนไป ค่าแบนด์วิดท์ก็จะเปลี่ยนแปลงแตกต่างไปจากเดิม ทั้งนี้ขึ้นอยู่กับวงจรฟิลเตอร์โดยใช้ย่านความถี่ในแต่ละออกเตฟ เพราะในแต่ละออกเตฟจะมีความถี่เสียง 2 ความถี่ คือความถี่ที่สูงกว่าจุดตัด (Upper cut off) และความถี่ที่ต่ำกว่าจุดตัด (Lower cut off) เป็นอัตราส่วนต่อกันช่วงความถี่ของแบนด์วิดท์ที่เปลี่ยนแปลงขึ้นกับวงจรฟิลเตอร์ชนิดต่างๆ เช่น

Low pass	จะยอมให้ความถี่ที่ต่ำกว่าที่กำหนดผ่านได้
High pass	จะยอมให้ความถี่ที่สูงกว่าที่กำหนดผ่านได้
Band pass	จะยอมให้ความถี่ในช่วงที่กำหนดผ่านได้ ส่วนความถี่นอกเหนือจากที่กำหนด จะถูกลดทอนขนาดของสัญญาณลง
All pass	จะยอมให้ทุกความถี่ผ่านไปได้ โดยไม่ลดทอนขนาดของสัญญาณ ทั้งนี้อาจมีการเปลี่ยนแปลงทางเฟสของสัญญาณนี้อาจเกิดขึ้นบางความถี่เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

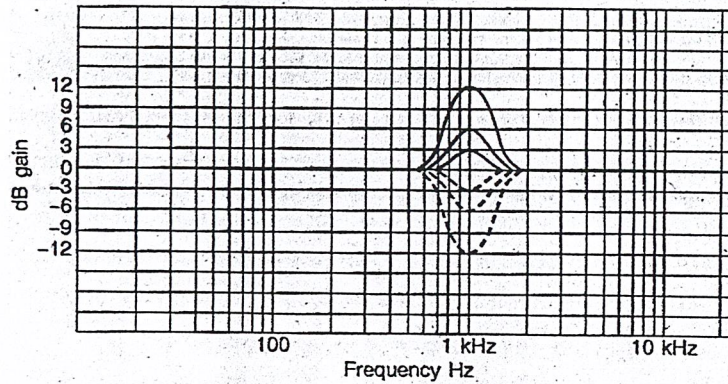
2.15 การปรับแต่งลักษณะของเสียง

เสียงเป็นสัญญาณที่เหมาะสมสำหรับนำมาประมวลผล เนื่องจากย่านความถี่ของเสียงจัดอยู่ในย่านความถี่ต่ำ คือ เสียงพูดสามารถสุ่มได้คุณภาพดีด้วยอัตรา 8 kHz และเสียงดนตรีสามารถสุ่มได้คุณภาพดีด้วยอัตรา 44 kHz และเสียงยังเป็นสิ่งที่ละเอียดอ่อนมาก เนื่องจากมนุษย์สามารถรับรู้ได้ด้วยการฟัง เช่น เสียงทุ้มหรือเสียงแหลม, ทิศทางของเสียงมาจากทางใด, ใครคือผู้พูด ซึ่งลักษณะเหล่านี้จะถูกนำไปประมวลผลด้วยสัญญาณดิจิทัล

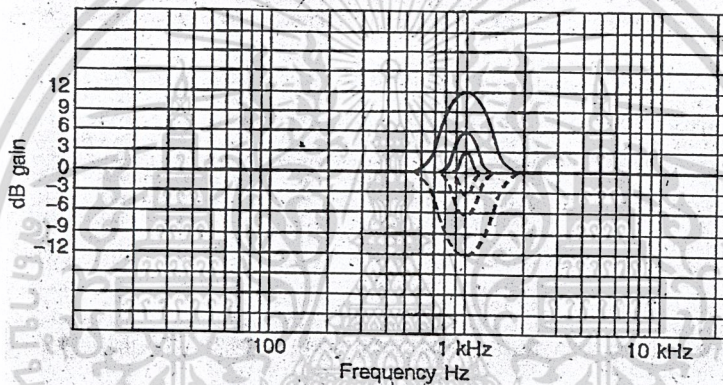
2.15.1 อีควอลไลเซอร์ (Equalizer)

ปัจจุบันบรรดาเครื่องเสียงเกือบทุกชนิดไม่ว่าจะเป็นชุดเครื่องเสียงหรือวิทยุ-เทปขนาดใดก็ตามต่างพยายามปรับแต่งเสียงให้ได้ไพเราะมากขึ้น รวมทั้งเป็นการจูงใจผู้ซื้อ ทางผู้ผลิตจึงเพิ่มภาคอีควอลไลเซอร์เข้าไปด้วย จำนวนช่องหรือย่านความถี่จะขึ้นกับขนาดและราคาของเครื่องเสียง อีควอลไลเซอร์เป็นส่วนสำคัญในการควบคุมปรับแต่งหรือปรับสมดุลระดับเสียงในการเพิ่มลดสัญญาณเสียงในแต่ละระดับให้ดังออกมาน้อยตามความต้องการของผู้ฟัง ทั้งนี้เพราะว่าวงจรควบคุมเสียงทุ้มแหลมแบบพาสซีฟหรือแอคทีฟ ไม่สามารถทำให้ระดับของเสียงออกมาได้ตามต้องการ เช่น ขณะเพิ่มเสียงทุ้ม เสียงแหลมก็จะถูกกดไว้ ในทำนองเดียวกันขณะเพิ่มเสียงแหลม เสียงทุ้มก็จะถูกกดไว้เช่นกัน ทำให้วงจรปรับเสียงทุ้มแหลมไม่สามารถปรับยกระดับเสียงทุ้มแหลมพร้อมกันได้ แต่วงจรอีควอลไลเซอร์สามารถทำได้โดยแบ่งระดับเสียงให้เป็นหลายย่าน ซึ่งอัตราขยายแต่ละย่านสามารถปรับให้เพิ่มหรือลดได้ถึง $\pm 12dB$ หรือมากกว่า

อีควอลไลเซอร์นิยมใช้งานอย่างแพร่หลาย เพื่อแก้ปัญหาสมดุลของระบบเสียงให้ถูกต้อง จากรูปที่ 2.16 เป็นรูปแบบที่เป็นไปได้ของวงจรฟิลเตอร์ โดยเลือกใช้ความถี่ 1 kHz มีการปรับยกและปรับลดระดับ 3 ระดับ การเลื่อนปรับเพิ่มและลดสูงสุดทั้งสองแบบของรูปกราฟในส่วนของค่า Q จะคล้ายๆกัน กล่าวคือ Q จะมีค่าสูงทั้งคู่ นั่นคือแบบแรก จะมีความลาดชันของเส้นโค้งน้อยกว่าเมื่อปรับเปลี่ยนการเพิ่มหรือลดระดับเสียงลง ค่าของ Q จะเปลี่ยนแปลงสอดคล้องกับจำนวนองศาที่หันเหออกจากตำแหน่งศูนย์กลางความถี่



รูปที่ 2.16 ชนิดของอิกวอไลเซอร์ที่ค่า Q ขึ้นกับจำนวนองศาของการปรับเพิ่มหรือลด



รูปที่ 2.17 ค่า Q ของวงจรคงที่

ดังนั้นขณะค่า Q คงที่ จะทำให้เกิดผลตอบสนองทางความถี่เป็นช่วงแคบและชันมาก ลักษณะของเส้นโค้งจากรูปที่ 2.17 Q ที่ได้จะมีค่าสูง ทั้งนี้เพราะว่าค่า Q ของฟิลเตอร์จะมีค่าเท่ากับความถี่ศูนย์กลางหารด้วยแบนด์วิดท์ อิกวอไลเซอร์บางแบบจะมีช่องสัญญาณเดี่ยวบ้างหรือแบบสเตอริโอบ้าง

การทำอิกวอไลเซอร์ด้วยการประมวลผลทางสัญญาณดิจิทัล จะใช้ตัวกรองแบบดิจิทัลแบ่งย่านความถี่เป็นหลายๆย่านและผู้ใช้สามารถปรับอัตราขยายได้ในแต่ละย่าน เช่น อิกวอไลเซอร์ ที่มี 5 ย่านความถี่ โดยใช้อัตราส่วนเท่ากับ 44.1 kHz และตัวกรองแต่ละตัวใช้ในแต่ละย่านความถี่และมีความถี่ตัดคือ

1. $H_1(z)$ ผ่านย่านความถี่ 0 - 3 kHz
2. $H_2(z)$ ผ่านย่านความถี่ 3 - 7 kHz
3. $H_3(z)$ ผ่านย่านความถี่ 7 - 11 kHz
4. $H_4(z)$ ผ่านย่านความถี่ 11 - 15 kHz
5. $H_5(z)$ ผ่านย่านความถี่ 15 - 22.05 kHz

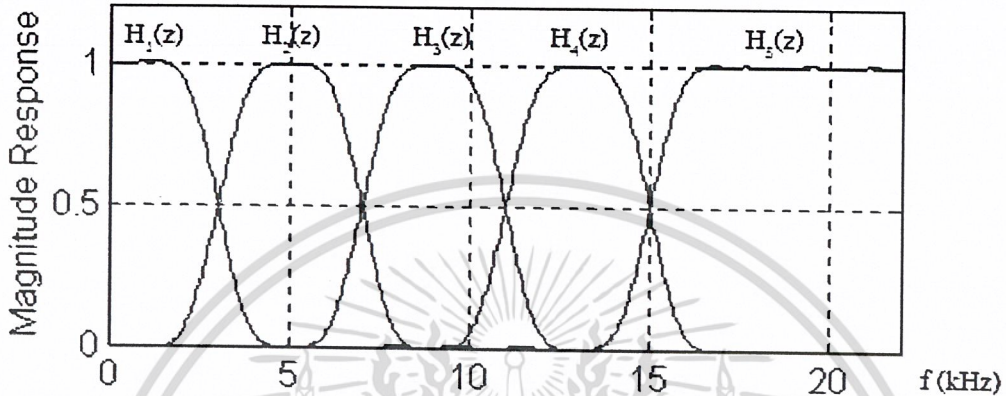
ตัวกรองทั้ง 5 สามารถสร้างโดยใช้ตัวกรอง FIR หรือ IIR ก็ได้ โดยต้องออกแบบให้ผลรวมของ

ฟังก์ชันถ่ายโอนทั้งหมดรวมแล้วเป็นตัวกรองแบบผ่านตลอด (all pass filter) ที่มีอัตราขยายคงที่ตลอดทุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติเห็นาไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่ การออกแบบนี้ทำได้ง่ายมากโดยใช้ตัวกรองแบบ FIR โดยให้ความถี่ตัดของตัวกรองมีค่าเท่ากับจุดความถี่ปลายของความถี่ ซึ่งจะได้ผลตอบสนองเชิงความถี่ของแต่ละย่านความถี่ดังแสดงในรูปที่ 2.18 สังเกตว่า แถบเปลี่ยนของย่านความถี่ที่อยู่ติดกันจะมีลักษณะที่สมมาตรกัน คือขณะที่ผลตอบสนองทางความถี่ของย่านหนึ่งลาดลง ผลตอบสนองในย่านที่ติดกันก็จะลาดขึ้นในลักษณะที่สมมาตรกัน ซึ่งก็จะทำให้ผลตอบสนองความถี่รวมเป็นหนึ่ง



รูปที่ 2.18 ผลตอบสนองเชิงความถี่ของตัวกรองที่ใช้

การออกแบบให้ ตัวกรองตัวที่หนึ่งเป็น LPF, ตัวที่สองถึงสี่เป็น BPF และตัวที่ห้าเป็น HPF ซึ่งมีผลตอบสนองต่ออิมพัลส์ คือ

$$\begin{aligned}
 h_1(n) &= w(n) \left[\frac{\sin(\omega'_1)(n-m)}{\pi(n-M)} \right] \\
 h_2(n) &= w(n) \left[\frac{\sin(\omega'_2)(n-m) - \sin(\omega'_1)(n-m)}{\pi(n-M)} \right] \\
 h_3(n) &= w(n) \left[\frac{\sin(\omega'_3)(n-m) - \sin(\omega'_2)(n-m)}{\pi(n-M)} \right] \\
 h_4(n) &= w(n) \left[\frac{\sin(\omega'_4)(n-m) - \sin(\omega'_3)(n-m)}{\pi(n-M)} \right] \\
 h_5(n) &= w(n) \left[\delta(n-M) - \frac{\sin(\omega'_4)(n-m)}{\pi(n-M)} \right]
 \end{aligned} \tag{2.5}$$

โดยที่ $M=25$ $\omega'_i = 2\pi f_i / f_s$ เมื่อ f_1, f_2, f_3 และ f_4 เท่ากับ 3, 7, 11, 15 kHz ตามลำดับ สังเกตได้ว่าผลบวกของ $h_i(n)$ ทุกตัว จะได้เป็นตัวกรองแบบผ่านตลอด ที่สัญญาณขาออกจะล่าหลังสัญญาณขาเข้า M จุด เขียนเป็นสมการได้ว่า

$$h_1(n) + h_2(n) + h_3(n) + h_4(n) + h_5(n) = \delta(n-M) \tag{2.6}$$

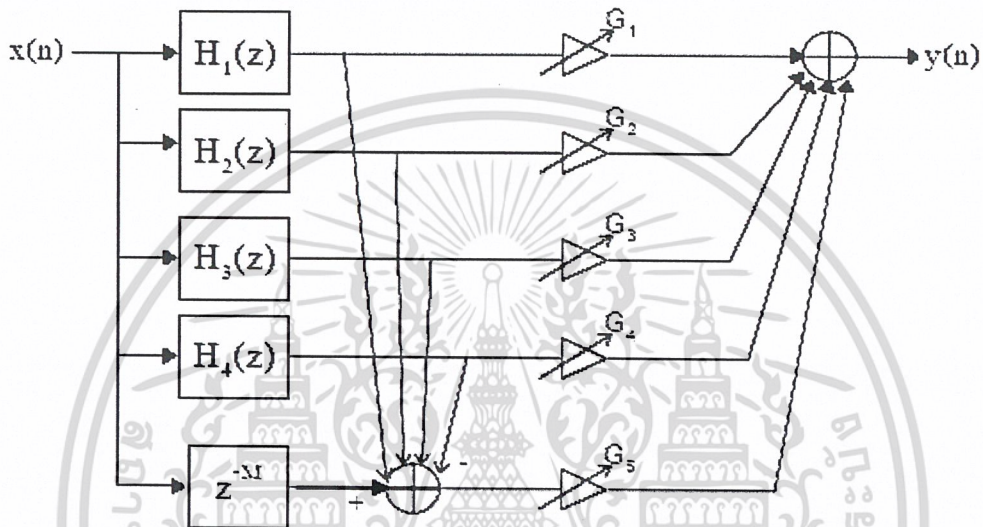
แปลง z ได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$H_1(z) + H_2(z) + H_3(z) + H_4(z) + H_5(z) = z^{-M} \quad (2.7)$$

ผลตอบสนองจากตัวกรองทั้งหมด จะนำมาผ่านตัวคูณที่ปรับค่าได้โดยผู้ใช้และนำผลคูณทั้งหมดมาบวกกันเป็นผลลัพธ์สุดท้าย จะได้ว่าฟังก์ชันถ่ายโอนรวมของระบบนี้ คือ

$$H_{total}(z) = G_1 H_1(z) + G_2 H_2(z) + G_3 H_3(z) + G_4 H_4(z) + G_5 H_5(z) \quad (2.8)$$



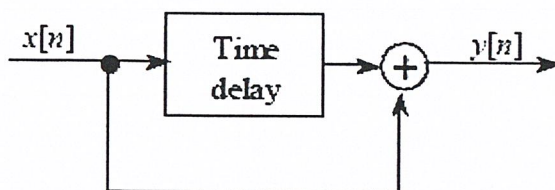
รูปที่ 2.19 โครงสร้างอิกวไลเซอร์เสียง 5 ช่อง

การใช้ตัวกรอง FIR มีข้อดี คือ ออกแบบสัมประสิทธิ์ของตัวกรองได้ง่ายและให้เฟสที่เป็นเชิงเส้นที่สมบูรณ์ในทุกย่านความถี่ มีข้อเสีย คือ ต้องการการประมวลผลที่ค่อนข้างมาก เพราะต้องคำนวณตัวกรอง FIR อันดับ 40 ถึง 4 ตัว ดังนั้น ถ้าหากมีข้อจำกัดในเรื่องทรัพยากรในการประมวลผลอาจต้องพยายามออกแบบให้เป็นตัวกรองแบบ IIR ซึ่งจะทำให้ ไปได้ยากกว่า เพราะผลตอบสนองเชิงความถี่ของตัวกรอง IIR ไม่สมมาตรกันในช่วงแถบเปลี่ยนเหมือนตัวกรอง FIR

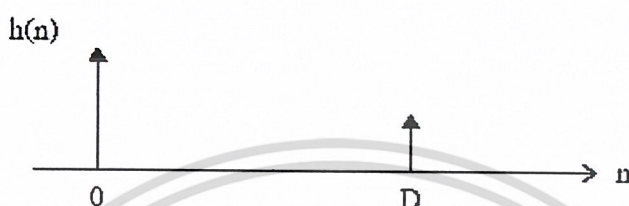
2.15.2 เสียงสะท้อน (Echo)

เสียงสะท้อนเป็นการทำเสียงปกติให้เป็นเสียงที่สะท้อนมาจากผนังถ้ำ จะสามารถสร้างได้ง่ายๆ โดยนำเสียงไปเข้าวงจร Time Delay แล้วนำมารวมกับเสียงอินพุต ดังรูปที่ 2.20 จะสามารถสร้างเสียงสะท้อนได้เสียงเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

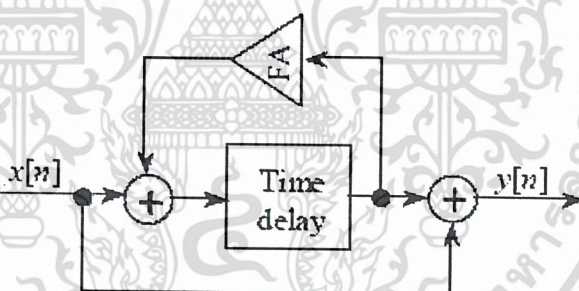


รูปที่ 2.20 การสร้างเสียงสะท้อน

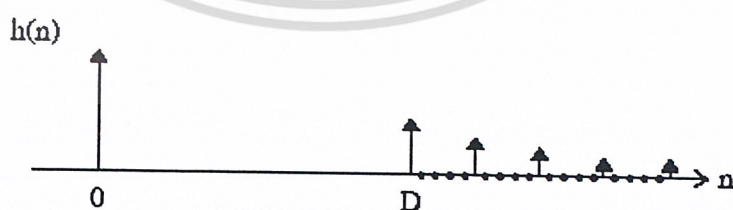


รูปที่ 2.21 ผลตอบสนองต่ออิมพัลส์ของระบบ

การสร้างเสียงสะท้อนไปสะท้อนมาหลายๆเสียงจนกระทั่งเสียงสะท้อนนั้นจางหายไป สามารถทำได้โดยเพิ่มวงจรป้อนกลับและทำการลดทอนเสียงไปพร้อมกัน



รูปที่ 2.22 การสร้างเสียงสะท้อน โดยใช้วงจรป้อนกลับ



รูปที่ 2.23 ผลตอบสนองต่ออิมพัลส์ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เสียงสะท้อนสามารถอธิบายโดยใช้ สมการอนุพันธ์อันดับที่ 1 (First order equation)

$$y[n] = x[n] + \gamma \cdot x[n-d] \quad (2.9)$$

โดย

$y[n-d]$	เป็นการสุ่มเสียงที่ได้จากอัตราขยายที่เท่ากัน
d	เป็นค่าเวลาที่ผู้ใช้ต้องการปรับแต่งเพื่อให้ได้ยินเสียงสะท้อน
$\gamma < 1$	เป็นอัตราขยายของวงจรป้อนกลับเป็นตัวควบคุมค่าเวลาขีดยึดของเสียงที่เราได้ยิน

จากหลักการของ DSP เสียงสะท้อนสามารถอธิบายได้โดยใช้หลักการ Circular Buffer สำหรับการประมวลผลของแต่ละช่องสัญญาณ ที่พักข้อมูลจะเก็บข้อมูลของเสียงสะท้อนในแต่ละช่วงเวลา ในแต่ละช่วงเวลากการสุ่มสัญญาณเสียงจะถูกรวมกับค่าที่ได้จากตัวพักข้อมูลและผลที่ได้จะถูกส่งออกเป็นผลลัพธ์และนำไปรวมกับตัวพักข้อมูลเดิม

พิจารณาที่พักข้อมูลของเสียงสะท้อนที่มีความยาวเท่ากับ N จะมีอัลกอริทึมง่าย ๆ สำหรับการสร้างเสียงสะท้อน โดยทำตามขั้นตอนดังนี้

```
Output = input + gain buffer[k - delay];
Buffer[k] = output;
k=(k+1) mod N;
k - delay = (k - delay) mod N;
```

โดย

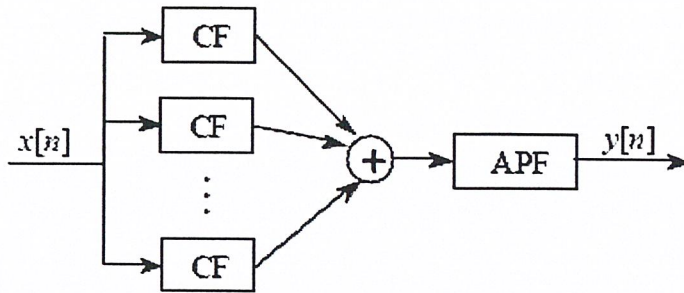
k	เป็นตัวชี้ที่อยู่ปัจจุบันของตัวเก็บข้อมูล
gain	เป็นส่วนประกอบของวงจรป้อนกลับที่มีการลดทอน ซึ่งผู้ใช้สามารถควบคุมเองได้

2.15.3 เสียงจำลองการสะท้อนของห้อง (reverb)

เสียงสะท้อนก้องจะมีลักษณะเหมือนได้ยินเสียงสะท้อนมาจากห้องขนาดใหญ่หรือผนังตึก เสียงสะท้อนในห้องคือเสียงที่สะท้อนมาจากกำแพง หลังคาและพื้นห้อง เสียงที่ได้ยินในแต่ละช่วงเวลาจะเป็นการรวมเสียงจากแหล่งกำเนิดกับเสียงที่มาจากการสะท้อน

เสียงกระตุ้น เช่น เสียงปรบมือจะลดทอนสลายแบบเอกซ์โปเนนเชียล ห้องแสดงคอนเสิร์ตจะถูกออกแบบเพื่อให้การยึดเยื้อของเสียงเหมาะสมสำหรับเสียงทุกเสียงที่ถูกสร้างขึ้นภายในห้อง

อุปกรณ์สร้างเสียงสะท้อนก้องแบบดิจิทัลสามารถอธิบายโดยการใช้ฟิลเตอร์หลาย ๆ ตัวต่อแบบขนานโดยมีเวลาหน่วงที่แตกต่างกันผลลัพธ์ที่ได้จากฟิลเตอร์จะถูกรวมกันและนำไปผ่านฟิลเตอร์แบบผ่านตลอดเพื่อที่จะสร้างเสียงสะท้อนก้องโดยดูได้จากรูปที่ 2.24



รูปที่ 2.24 การสร้างเสียงจำลองการสะท้อนของห้อง

การอธิบายเสียงสะท้อนห้องจะใช้หลักการของ DSP ในการสร้างฟิลเตอร์ เสียงสะท้อนห้องจะสามารถใช้ที่พิกซ์ข้อมูลตัวเดียวกันที่ใช้ในเสียงสะท้อนแต่จะต่างที่อัลกอริทึม ในเสียงสะท้อนห้องจะไม่เก็บผลลัพธ์การสุ่มไว้ในตัวพิกซ์ข้อมูล แต่จะเก็บเสียงที่ทำการสุ่มจากแหล่งกำเนิด

การอธิบายแบบทั่วไปของเสียงสะท้อนห้องแบบดิจิทัลสามารถดูได้จากออกการิทึมดังต่อไปนี้

$$y[n] = x[n] + \gamma \cdot x[n-d_1] + \gamma^2 \cdot x[n-d_2] + \gamma^3 \cdot x[n-d_3] + \gamma^4 \cdot x[n-d_4] + \gamma^5 \cdot x[n-d_5] + \gamma^6 \cdot x[n-d_6] + \gamma^7 \cdot x[n-d_7] + \gamma^8 \cdot x[n-d_8] \quad (2.10)$$

$y[n-d]$ เป็นการสุ่มเสียงที่ได้จากอัตราขยายที่เท่ากัน

d เป็นค่าเวลาที่ผู้ใช้ต้องการ ปรับแต่งเพื่อให้ได้ยินเสียงสะท้อนมีค่าไม่เท่ากัน

$\gamma < 1$ เป็นอัตราขยายของวงจรป้อนกลับเป็นตัวควบคุมค่าเวลาซัดเยื่อของเสียงที่เราได้ยิน

2.15.4 เสียงคอรัส (Chorus)

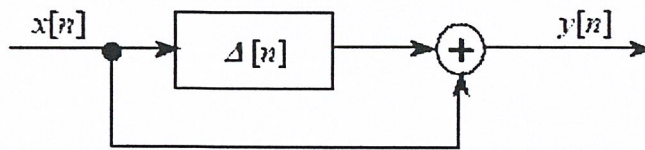
คอรัสเป็นการจำลองการประสานเสียง สมมติว่าเรามีเสียงเพลงซึ่งเล่นด้วยเครื่องดนตรีเพียงชนิดเดียว เราสามารถจำลองเสียงประสานให้เหมือนเป็นการเล่นเครื่องดนตรีชนิดเดียวกัน เพลงเดียวกันแต่มีผู้เล่นหลายๆคนพร้อมกันได้

สามารถอธิบายได้ด้วยอัลกอริทึมง่ายๆ

$$y[n] = x[n] + x[n - \Delta[n]] \quad (2.11)$$

โดย

$\Delta[n]: Z \rightarrow Z$ เป็นสัญญาณที่ไม่ต่อเนื่องและมีคาบ สามารถ อธิบายได้ดังรูป



รูปที่ 2.25 โครงสร้างการทำเสียงคอรัส

การดีเลย์จะเพิ่มขึ้นจนถึงค่า ค่าหนึ่งแล้วก็จะลดลงไปจนถึงค่าต่ำสุด ค่าสูงสุด อัตราการเพิ่มขึ้นและการลดลงของฟังก์ชัน $\Delta[n]$ สามารถจัดการกับตัวแปรในการทำเสียงคอรัส โดยปกติแล้วการทำเสียงคอรัสจะถ่วงเวลาให้ล่าหลังระหว่าง 40 msec ถึง 60 msec ที่อัตรา 0.25 Hz

พิจารณาในทาง DSP การสร้างเสียงคอรัสจะสามารถอธิบายได้โดย Circular Buffer สำหรับการประมวลผลของแต่ละช่องสัญญาณ ผลลัพธ์ที่จะเกิดจากการรวมสัญญาณเสียงปัจจุบันกับสัญญาณเสียงที่ผ่านมาช่วงเวลา

```
Buffer[i] = input[i];
Output[i] = buffer [i] + buffer[I - offset [i] ];
```

การห้วงเวลาจะสามารถกู้สัญญาณกลับมาจากตัวพักข้อมูลได้ คำนวณ โดยใช้ฟังก์ชัน $\Delta[n]$ ระหว่างค่าต่ำสุดและค่าสูงสุด ซึ่งค่านี้สามารถกำหนดได้โดยตรงจากผู้ใช้

2.16 การประมวลผลสัญญาณดิจิทัล (Digital Signal Processing)

สามารถแบ่งได้เป็น 2 หัวข้อใหญ่ๆ คือ

2.16.1 การกรองดิจิทัล (Digital Filter) การกรองดิจิทัลมีอยู่หลายประเภทด้วยกัน อย่างไรก็ตามสำหรับในหัวข้องานวิจัยนี้ เราจะสนใจเฉพาะการกรองดิจิทัลแบบเป็นระบบเชิงเส้น ซึ่งมีตัวกรองดิจิทัลอยู่ 2 ชนิดคือ ตัวกรองไม่ป้อนกลับดิจิทัล (Non-recursive Digital Filter) และตัวกรองป้อนกลับดิจิทัล (Recursive Digital Filter) ตัวกรองทั้ง 2 ชนิดเป็นวงจรพื้นฐานของวงจรกรองดิจิทัลแบบอื่นๆ รายละเอียดเกี่ยวกับคุณสมบัติ หลักการออกแบบหลักการสร้างจะกล่าวถึงในบทต่อไป

2.16.2 การแปลงดิจิทัล (Digital Transform) การแปลงดิจิทัลโดยเฉพาะการแปลงคิสริตฟูรีเยอร์ (Discrete Fourier Transform หรือ DFT) ที่ทำการประมวลผลโดยใช้ขั้นตอนวิธีที่เรียกว่า การแปลงฟาสฟูรีเยอร์ (Fast Fourier Transform หรือ FFT) มีส่วนทำให้การประมวลผลสัญญาณดิจิทัลได้รับความนิยมมากขึ้น

ตัวอย่างการนำไปประยุกต์ใช้งานของการประมวลผลสัญญาณดิจิทัลมีให้เห็นอยู่มากในปัจจุบัน เช่น เทคนิคการสื่อสารดิจิทัล ซึ่งอาจเป็นการมอดูเลต (modulation) หรือการเข้ารหัส (coding) สัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดิจิทัล นอกจากนี้ยังมีใช้ในระบบควบคุมดิจิทัล (digital control system) และอุปกรณ์เครื่องมือดิจิทัล เราอาจหมายถึง (ก) การประมวลผลดิจิทัลของสัญญาณใดๆ ที่ไม่ได้อยู่ในรูปแบบสัญญาณดิจิทัล หรือ

(ข) การประมวลผล (ด้วยวิธีใดก็ตาม) ของสัญญาณที่อยู่ในรูปแบบสัญญาณดิจิทัลแล้ว

ความหมายในข้อ (ก) นั้นหมายถึง การประมวลผลสัญญาณในธรรมชาติทั่วไป ซึ่งในสัญญาณธรรมชาตินั้นปกติอยู่ในรูปแบบสัญญาณอนาล็อก (analog signal) ก่อนการประมวลผลเราต้องทำการแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัลเสียก่อน ส่วนความหมายในข้อ (ข) เป็นการประมวลผลสำหรับสัญญาณที่มาจากอุปกรณ์ดิจิทัล เช่น คอมพิวเตอร์หรือระบบควบคุมดิจิทัล สัญญาณจากอุปกรณ์เหล่านี้เป็นสัญญาณดิจิทัลอยู่แล้ว จึงไม่จำเป็นต้องมีการแปลงรูปสัญญาณอีก

เหตุผลสำคัญที่ทำให้การประมวลผลสัญญาณดิจิทัลได้รับความสนใจมากขึ้นก็เนื่องจากเทคโนโลยีในการสร้างวงจรรวมหรือวงจรรวมไอซี (integrated circuit) สามารถสร้างวงจรมีผลผลิตบนสารกึ่งตัวนำขึ้นเดียวกันได้และสามารถผลิตครั้งละจำนวนมากขึ้นได้ จึงทำให้ราคาต่อวงจรถูกลงเป็นอย่างมาก โดยเฉพาะอย่างยิ่งวงจรรวมแบบแอลเอสไอ (Large Scale Integrated Circuit:LSI) ที่สำคัญๆ เช่น ไมโครโปรเซสเซอร์ (Microprocessor) ซึ่งมีราคาถูกมากและมีผู้สนใจนำมาทำเป็นตัวประมวลสัญญาณมากขึ้น นอกจากนี้การประมวลผลสัญญาณดิจิทัลยังให้ความน่าเชื่อถือและความแม่นยำในการคำนวณดีกว่าการประมวลผลสัญญาณอนาล็อกมาก

2.17 ข้อได้เปรียบของการประมวลผลสัญญาณดิจิทัล

ข้อได้เปรียบของการประมวลผลสัญญาณดิจิทัลกล่าวได้เป็นข้อๆ ดังนี้

2.17.1 เหมาะสำหรับอุปกรณ์ที่อยู่ในรูปแบบสัญญาณดิจิทัลอยู่แล้ว เช่น ผลลัพธ์จากคอมพิวเตอร์ ไมโครโปรเซสเซอร์หรือข้อมูลจากระบบควบคุมดิจิทัล เป็นต้น ทั้งนี้เนื่องจากสัญญาณหรือข้อมูลจากอุปกรณ์เหล่านี้ ถ้าหากต้องการนำไปประมวลผลสัญญาณอนาล็อก ก็จำเป็นต้องมีวงจรแปลงรูปแบบสัญญาณดิจิทัลเป็นอนาล็อก (Digital to Analog converter หรือ D/A) เพื่อทำการแปลงสัญญาณดิจิทัลให้เป็นสัญญาณอนาล็อกก่อน หลังจากการประมวลผลด้วยระบบประมวลผลอนาล็อกแล้วก็ต้องแปลงผลลัพธ์ที่เป็นสัญญาณอนาล็อกเป็นเป็นดิจิทัล (Analog to Digital converter หรือ A/D) เพื่อส่งกลับไปยังอุปกรณ์ดิจิทัลต่อไป ซึ่งจะเห็นได้ว่าระบบประมวลผลมีความซับซ้อนมากขึ้น นอกจากนี้ถ้าหากเป็นระบบที่ต้องทำการประมวลผลสัญญาณพร้อมกันหลายสัญญาณ (ซึ่งในคอมพิวเตอร์มักทำในลักษณะนี้) ก็จะทำให้ระบบยุ่งยากมากขึ้น

2.17.2 อุปกรณ์ทางด้านดิจิทัลมีราคาถูก ขนาดเล็ก ประสิทธิภาพสูง ความแม่นยำและความแม่นยำสูง นอกจากนี้การพัฒนาให้มีประสิทธิภาพสูงขึ้นก็เป็นไปอย่างรวดเร็ว ข้อนี้เป็นข้อได้เปรียบที่สำคัญ เพราะว่าการประมวลผลสัญญาณดิจิทัลสามารถคำนวณได้อย่างแม่นยำและต้องการให้มีความละเอียดเท่าใดก็ได้ แต่การประมวลผลสัญญาณอนาล็อกการคำนวณที่ให้ความละเอียดเกินกว่า หนึ่งในพันส่วนหรือ 0.001 นั้นทำได้ยากมาก นอกจากนี้อุปกรณ์ที่ใช้ในระบบของการประมวลผลสัญญาณอนาล็อก เช่น ตัวต้านทาน ตัวขยายสัญญาณ ยังมีคุณสมบัติแปรค่าได้ตามสภาพแวดล้อม เช่น อุณหภูมิ ความชื้น อายุการใช้งาน ให้ความแม่นยำและความน่าเชื่อถือของระบบการประมวลผลสัญญาณอนาล็อกมีต่ำ อย่างไรก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็ตามในปัจจุบันการประมวลผลสัญญาณดิจิทัลที่ให้ความแม่นยำสูงต้องใช้อุปกรณ์ราคาแพง ถ้ามีการพัฒนาให้มีราคาถูกลงก็จะเป็นผลดียิ่ง

2.17.3 การรับและส่งข้อมูลหรือสัญญาณดิจิทัล ทำได้แน่นอนกว่าทั้งนี้เนื่องจากสัญญาณดิจิทัลมีแค่ 2 ระดับ คือ 0 และ 1 เท่านั้น ถ้าหากรูปคลื่นสัญญาณผิดเพี้ยนไปก็สามารถแก้ไขและสร้างขึ้นใหม่ให้เหมือนเดิมได้โดยง่าย

2.17.4 การประมวลผลสัญญาณดิจิทัลสามารถทำเป็นแบบระบบแบ่งเวลาการใช้ (time-shared system) ได้ ดังนั้นจึงสามารถทำการประมวลผลพร้อมๆกันได้หลายช่องสัญญาณ นอกจากนี้ในระบบเดียวกันยังสามารถโปรแกรมให้ทำงานได้หลายรูปแบบด้วย

2.17.5 ระบบการประมวลผลสัญญาณดิจิทัลทำได้ง่าย เนื่องจากขั้นตอนวิธี (algorithm) การประมวลผลสัญญาณมักจะประกอบไปด้วย การบวก การลบ การคูณ การหาร และการเลื่อนตัวเลขเท่านั้น

2.17.6 ระบบประมวลผลสัญญาณดิจิทัลมีความคล่องตัวสูง เนื่องจากสามารถทำการมัลติเพล็กซ์กับข้อมูลหรือสัญญาณเสียงหรือสัญญาณภาพได้ การมัลติเพล็กซ์ยังสามารถทำเป็นแบบการมัลติเพล็กซ์แบบแบ่งเวลา (Time Division Multiplex หรือ TDM) การมัลติเพล็กซ์แบบแบ่งรหัส (Code Division Multiplex หรือ CDM) นอกจากนี้การส่งสัญญาณยังส่งได้ในอัตราบิต (Bit Rate) หรืออัตราบอด (Baud Rate) ต่างๆกันได้

2.18 ข้อเสียของการประมวลผลสัญญาณเชิงดิจิทัล

เนื่องจากการเปลี่ยนไปตามธรรมชาติของสัญญาณซึ่งปกติมีความต่อเนื่อง การประมวลผลสัญญาณเชิงดิจิทัลจึงมีข้อจำกัดบางอย่างเช่น

ตัวประมวลผลจะต้องมีไฟเลี้ยงตลอดเวลาในขณะที่วงจรประมวลผลสัญญาณอนาลอก (เช่น วงจรกรองความถี่ชนิดพาสซีฟ) ไม่จำเป็นต้องใช้ไฟเลี้ยงอย่างไรก็ตามปัจจุบันนี้วงจรประมวลผลสัญญาณอนาลอกที่ซับซ้อน ก็ขาดไฟเลี้ยงไม่ได้เช่นกัน

2.18.1 ตัวประมวลผลไม่สามารถที่จะทำงานที่ความถี่สูงมากๆได้

2.18.2 หากการประมวลผลกระทำในแวลล้อมของสัญญาณเชิงอนาลอก วงจรส่วนแปลงสัญญาณ ให้ความซับซ้อนของระบบโดยรวมเพิ่มขึ้น อีกทั้งวงจรงดงกล่าวจะมีข้อจำกัดคือขนาดของสัญญาณต้องพอเหมาะ ไม่ใหญ่หรือเล็กเกินไป

2.18.3 สำหรับข้อมูลข่าวสารชุดเดียวกันการประมวลผลสัญญาณเชิงดิจิทัลจะต้องการแถบความกว้างของความถี่สูงกว่า (เช่น ความถี่ของการสุ่มจะต้องมากกว่าความถี่ของสัญญาณที่ถูกสุ่มอย่างน้อย 2 เท่า)

2.18.4 ตัวประมวลผลมีความซับซ้อนสูง ในการออกแบบต้องอาศัยความรู้และทักษะมากกว่า

2.18.5 สัญญาณเดิมเมื่อเปลี่ยนเป็นสัญญาณดิจิทัล ความถูกต้องบางส่วนจะขาดหายไป (เนื่องจากการ) ตัดทิ้ง (truncation) และการปัดเศษ (round off) โดยที่ไม่สามารถจะทำได้เหมือนเดิมได้

2.19 การแปลงแซด (Z-Transform)

การแปลงแซดเป็นการแปลงอีกชนิดหนึ่งซึ่งนำออกแบบตัวกรองสัญญาณทั้งหลายนิยมใช้อย่างถึงวิธีการนี้ จะเป็นการแปลงลำดับซึ่งเป็นตัวเลขมีเขียนเรียงกันให้เป็นฟังก์ชันของตัวแปรเชิงซ้อน (Complex variable) หรือกล่าวให้ง่ายคือ การเอาลำดับชุดหนึ่งไปสร้างฟังก์ชันตามกฎเกณฑ์ทางคณิตศาสตร์ที่กำหนดไว้ ดังนั้นคุณสมบัติต่างๆ ที่มีอยู่ในลำดับจะถูกส่งเข้าไปอยู่ในฟังก์ชันที่สร้างขึ้นด้วยและจะใช้อักษร Z ในภาษาอังกฤษเป็นสัญลักษณ์ของตัวแปรในฟังก์ชันที่สร้างขึ้น

การแปลงแซดเป็นสิ่งสำคัญอย่างหนึ่งในระบบการประมวลผลสัญญาณดิจิทัล นั่นคือใช้ในการออกแบบเพื่อวิเคราะห์สัญญาณผิดพลาด (Error) ในตัวกรองดิจิทัล (digital filter) โดยเฉพาะในระบบตัวกรองดิจิทัลแบบไออาร์ (IIR Filter) ซึ่งมักจะถูกใช้ในการคำนวณหาค่าสัมประสิทธิ์ของตัวกรองดิจิทัลและวิเคราะห์ข้อผิดพลาดของการควอนไทซ์ (Quantization Error) คือใช้ในการแสดง โครงสร้างของตัวกรองดิจิทัล

2.19.1 บทนิยามของการแปลงแซด

ถ้าหากเรามีลำดับสัญญาณเข้าในโดเมนเวลา (Time Domain) หรือ $x(0), x(1), \dots, x(m)$ ซึ่งเราสามารถเขียนแทนลำดับนี้ด้วยสมการที่ (2.12)

$$x(n) = [x(0), x(1), \dots, x(m)]$$

$$x(n) = \sum_{m=-\infty}^{\infty} x(m)\delta(n-m) \quad (2.12)$$

และถ้าทำการแปลงแซด ซึ่งจะเขียนแทนด้วย ตัวดำเนินการ Z ของสมการจะได้

$$Z\{x(n)\} = Z\left\{\sum_{m=-\infty}^{\infty} x(m)\delta(n-m)\right\}$$

$$Z\{x(n)\} = \sum_{m=-\infty}^{\infty} x(m)Z\{\delta(n-m)\} \quad (2.13)$$

โดยที่ $Z\{x(n)\} = \exp\{-m\}$ ดังนั้นนิยามของการแปลงแซดคือ

$$X(Z) = Z\{x(n)\} = \sum_{m=-\infty}^{\infty} x(m)\exp(-m)$$

$$X(Z) = \sum_{m=-\infty}^{\infty} x(m)Z^{-m} \quad (2.14)$$

เมื่อ Z เป็นจำนวนเชิงซ้อนหรือ $Z = \exp(-m)$ เนื่องจากธรรมชาติ m ของการแปลงแซดในสมการที่ (2.14) แปรค่าจาก ∞ ถึง $-\infty$ ดังนั้น การแปลงแซดจึงมีชื่อเรียกอีกอย่างว่า ผลการแปลงแซดสองข้าง (two-side Z-Transform) แต่ในทางปฏิบัติเรามักจะมีลำดับ $x(m)$ โดยที่ m มีค่ามากกว่าศูนย์ ฉะนั้นจึงเรียกว่า ผลการแปลงแซดข้างเดียว (one-side Z-Transform) ซึ่งนิยามได้โดย

$$X(Z) = \sum_{m=0}^{\infty} x(m)Z^{-m} \quad (2.15)$$

เราจะเห็นได้ชัดจากสมการที่ (2.13) และ (2.14) ว่าแท้จริงแล้ว การแปลงแซดคือการประยุกต์ใช้

การแปลงลาปลาซกับสัญญาณที่เป็นลำดับนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.19.2 การแปลงแฉดผกผัน (Inverse Z-Transform)

การแปลงแฉดผกผันเป็นการพิจารณาการแปลงผกผันจากฟังก์ชัน $X(Z)$ ให้กลับเป็นลำดับสัญญาณ $x(n)$ ซึ่งเป็นเครื่องมือที่ทำให้เราสามารถดูถึงผลตอบสนองอิมพัลส์หรือผลตอบสนองต่อฟังก์ชันหนึ่งหน่วยได้ ถ้าหากเรารู้ตำแหน่งโพล (pole) และซีโร (zero) ในโดเมนแฉดของระบบคิติดอล

ถ้าให้ $f(n)$ เป็นการแปลงแฉดผกผันของ $F(Z)$ การแปลงผกผันนี้นิยามได้คือ

$$f(n) = (1/2\pi j) \oint_C F(z)Z^{n-1} dz = z^{-1}\{F(z)\} \quad (2.16)$$

โดยที่ c เป็นค่าคอนทัวร์ในทิศทางทวนเข็มนาฬิกาที่ล้อมรอบจุดฐาน (singular point) ของ $F(Z)Z^{-1}$

สำหรับฟังก์ชันที่ซับซ้อน การทำการแปลงแฉดผกผันโดยใช้สมการที่ (2.16) อาจทำได้ค่อนข้างยุ่งยาก โดยทั่วไปมักนิยมใช้วิธีการดังนี้

การหาผลการแปลงแฉดผกผันโดยใช้ทฤษฎีเศษค่าง: ถ้าเราให้

$$F(Z)Z^{-1} = F_0(Z) = N(Z) / \sum_{i=1}^k (Z - p_i)^{m_i} \quad (2.17)$$

โดยที่ k และ m_i เป็นจำนวนเต็มบวก

	$f(nT)$	$F(z)$
1	$\delta(nT)$	1
2	$u(nT)$	$Z/(Z-1)$
3	$u(nT-T)k$	$k/(Z-1)$
4	$u(nT)kw^n$	$kZ/(Z-w)$
5	$u(nT-T)kw^{n-1}$	$k/(Z-w)$
6	$u(nT)\exp\{-\alpha nT\}$	$Z/(Z-\exp(-\alpha T))$
7	$u(nT)\sin n\omega T$	$\frac{Z \sin \omega T}{Z^2 - 2Z \cos \omega T + 1}$
8	$u(nT)\cos n\omega T$	$\frac{Z(Z - \cos \omega T)}{Z^2 - 2Z \cos \omega T + 1}$
9	$u(nT)e^{-\alpha nT} \sin n\omega T$	$\frac{Ze^{\alpha T} \sin \omega T}{Z^2 - 2Ze^{-\alpha T} \cos \omega T + e^{-2\alpha T}}$
10	$u(nT)e^{-\alpha nT} \cos n\omega T$	$\frac{Z(Z - e^{\alpha T} \cos \omega T)}{Z^2 - 2Ze^{\alpha T} \cos \omega T + e^{-2\alpha T}}$

ตารางที่ 2.2 ผลการแปลงแฉดและการแปลงแฉดผกผัน

2.20 สมการผลต่างสืบเนื่อง (Difference equation)

ในทำนองเดียวกับการวิเคราะห์ระบบบนาลอก ที่อธิบายคุณสมบัติของระบบในโดเมนเวลาโดยใช้สมการเชิงอนุพันธ์ (Differential equation) ในระบบดิจิทัลก็มีสมการผลต่างสืบเนื่อง ไว้ในการอธิบายคุณสมบัติของระบบในโดเมนเวลา ซึ่งสมการผลต่างสืบเนื่องอันดับ n อาจเขียนเป็น

$$y(n) = \sum_{i=0}^m a_i x(n-i) - \sum_{i=0}^m b_i y(n-i) \quad (2.18)$$

โดยที่ $x(n)$ เป็นลำดับสัญญาณเข้า

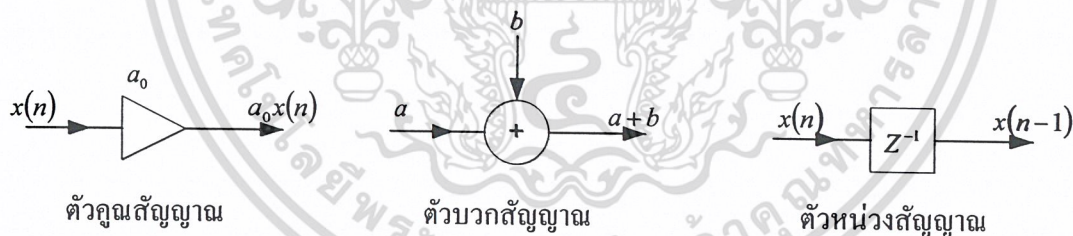
$y(n)$ เป็นลำดับสัญญาณออก

a_i, b_i เป็นค่าสัมประสิทธิ์

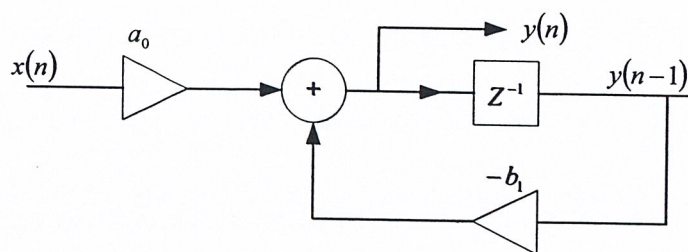
เพื่อเป็นประโยชน์ในการกล่าวถึงเบื้องต้นนี้จะพิจารณากรณีของ สมการผลต่างสืบเนื่องอันดับ 1 (1* order difference equation) ซึ่งอาจเขียนได้จากสมการที่ (2.19) คือ

$$y(n] = a_0 x(n) - b_1 y(n-1) \quad (2.19)$$

จากสมการที่ (2.19) ซึ่งเป็นตัวกรองดิจิทัลแบบง่ายที่สุดจะเห็นได้ว่า ถ้าต้องการสร้างตัวประมวลผลสัญญาณนี้ ต้องมีอุปกรณ์ 3 ชนิด คือ ตัวคูณเลข (multiplier), ตัวหน่วงลำดับสัญญาณ (delay) และตัวรวมสัญญาณ (summer) ซึ่งทั้ง 3 อุปกรณ์อาจเขียนสัญลักษณ์แทนได้ดังรูปที่ (2.26(a)) และเมื่อประยุกต์ใช้สัญลักษณ์นี้กับสมการที่ (2.19) ก็จะได้แผนภาพวงจรประมวลผลสัญญาณของสมการผลต่างสืบเนื่องอันดับที่ 1 ดังรูปที่ (2.26(b)) ถ้าสังเกตต่อไปก็จะเห็นว่าความจริงแล้วระบบการประมวลผลในสมการที่ (2.18) ก็ใช้อุปกรณ์ที่แสดงในรูปที่ (2.26(a)) เช่นกัน



รูปที่ 2.26(a) แผนภาพอุปกรณ์ในการประมวลผลสัญญาณดิจิทัล



รูปที่ 2.26(b) แผนภาพแทนสมการผลต่างสืบเนื่องอันดับ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.21 ตัวกรองดิจิทัล (Digital filter)

2.21.1 ความรู้เบื้องต้นของตัวกรองดิจิทัล (Introduction to Digital filter)

สัญญาณดิจิทัล (discrete-time signal) ที่ได้จากการวัดอาจจะมีสัญญาณอย่างอื่นที่ไม่ต้องการ เรียกว่า สัญญาณรบกวน (Noise) ปะปนผสมเข้ามาด้วย จึงจำเป็นต้องนำสัญญาณนี้ผ่านเข้าไปในอุปกรณ์อย่างหนึ่งซึ่งเรียกว่า ตัวกรองสัญญาณ (Filter) ซึ่งทำหน้าที่กรองสัญญาณรบกวนออก ที่ด้านเอาต์พุทของตัวกรองสัญญาณเราจะได้สัญญาณที่เราต้องการ ในการกรองสัญญาณนั้นก่อนอื่นเราจะต้องรู้จักคุณลักษณะเฉพาะตัวของสัญญาณที่ต้องการจะวัดและสัญญาณรบกวนที่เกิดขึ้น เพื่อให้สามารถสร้างตัวกรองสัญญาณที่มีคุณสมบัติยอมให้เฉพาะสัญญาณที่ต้องการผ่านออกมาได้เท่านั้น กระบวนการกรองสัญญาณดิจิทัลนี้ หากใช้กระบวนการทางดิจิทัล (เช่น ใช้คอมพิวเตอร์) จะเรียกว่า ตัวกรองสัญญาณดิจิทัล (Digital filter)

ตัวกรองดิจิทัลเป็นสิ่งที่สำคัญมากอย่างหนึ่งในการนำไปใช้ในระบบการประมวลผลดิจิทัล (Digital Signal Processing: DSP) เป็นการนำขั้นตอนวิธี (algorithm) ทางคณิตศาสตร์มาใช้ในระบบฮาร์ดแวร์หรือซอฟต์แวร์ กล่าวคือ สัญญาณอินพุทดิจิทัลจะทำการสร้างสัญญาณเอาต์พุทดิจิทัล โดยใช้นำขั้นตอนวิธีของการกรองสัญญาณ เพื่อให้ได้รับสัญญาณจากการกรองสัญญาณตามความต้องการเพื่อนำไปใช้ในตัวกรองดิจิทัล สำหรับชนิดของตัวกรองสัญญาณที่สำคัญจะถูกกำหนดโดย

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \quad (2.20)$$

โดยที่ $h(k)$ คือ สัมประสิทธิ์ของตัวกรองสัญญาณ

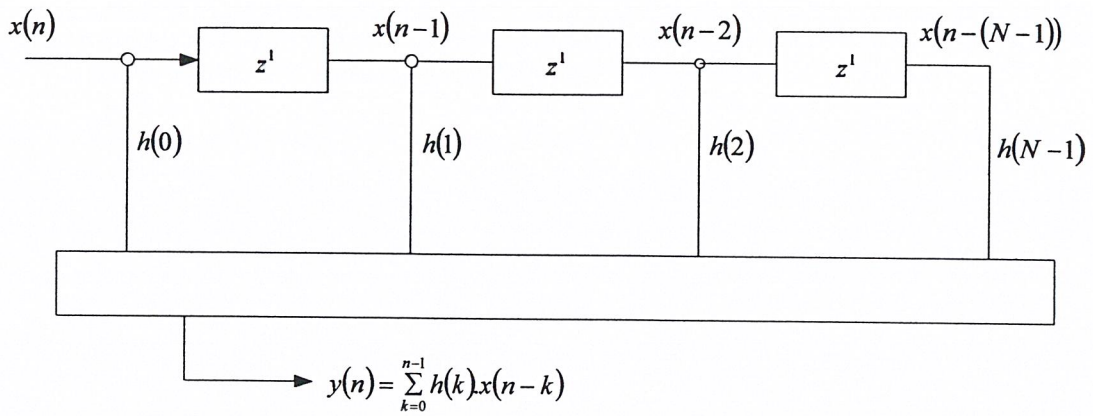
k คือ $0, 1, 2, \dots, n-1$

$x(n)$ คือ อินพุทของตัวกรองสัญญาณ

$y(n)$ คือ เอาต์พุทของตัวกรองสัญญาณ

ซึ่งสำหรับตัวกรองสัญญาณที่ให้มาค่าสัมประสิทธิ์จะกำหนดคุณสมบัติ (Characteristic) ของตัวกรองสัญญาณนี้จะขึ้นอยู่กับวิธีการคอนโวลูชัน (Convolution) ที่แท้จริงของสัญญาณและการตอบสนองต่ออิมพัลส์ของตัวกรองสัญญาณในโดเมนเวลา ซึ่งก็คือ $h(k)$

รูปดังกล่าวแสดงถึงบล็อกไดอะแกรม ซึ่งแสดงนิยามของตัวกรองดิจิทัลที่กำหนดไว้ในสมการที่ (2.20) โดยที่รูปแบบของตัวกรองสัญญาณชนิดนี้เป็นที่รู้จักกันอย่างแพร่หลาย ดังรูปที่ 2.27 Z^{-1} จะแสดงถึงการหน่วงเวลาในการสุ่มตัวอย่างหนึ่งครั้ง

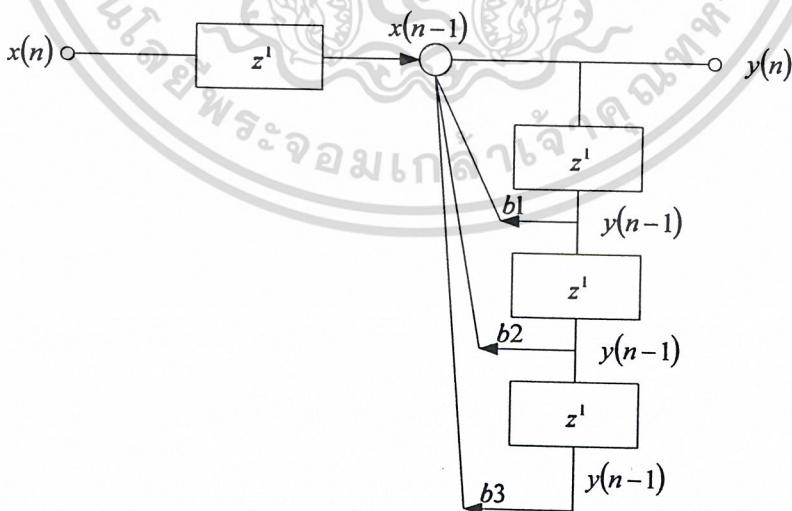


รูปที่ 2.27 บล็อกไดอะแกรมของตัวกรองสัญญาณ โดยมี $h(k), (k = 0, 1, 2, \dots, n-1)$

วัตถุประสงค์ของตัวกรองสัญญาณโดยรวมๆแล้ว เป็นการกำจัดหรือลดสัญญาณรบกวนจากสัญญาณที่ต้องการ โดยการดึงข้อมูลของกลุ่มที่สนใจช่วงหนึ่งออกมาซึ่งสัญญาณรบกวนในระบบอื่นๆ อาจจะปะปนมาด้วย ดังนั้นเราจึงทำการกรองสัญญาณที่แท้จริงออกมาเพื่อที่จะสามารถนำไปใช้ในการวิเคราะห์ให้ถูกต้อง

2.21.2 โครงสร้างของตัวกรองดิจิทัล (Realization structure for digital filter)

ตัวกรองสัญญาณดิจิทัล (discrete-time filter) จะถูกนำมาแสดงบ่อยๆในรูปแบบของบล็อกไดอะแกรมหรือ Signal Flow Graph ซึ่งบล็อกไดอะแกรมจะเป็นการสะดวกกว่าที่ใช้ในการแสดงสมการผลต่างสืบเนื่อง (Difference equation) ซึ่งสามารถพิจารณาดังตัวอย่าง ตัวกรองดิจิทัลด้วยสมการผลต่างสืบเนื่อง ดังรูปต่อไปนี้



รูปที่ 2.28 การวิเคราะห์สมการผลต่างสืบเนื่องในรูปแบบของบล็อกไดอะแกรม

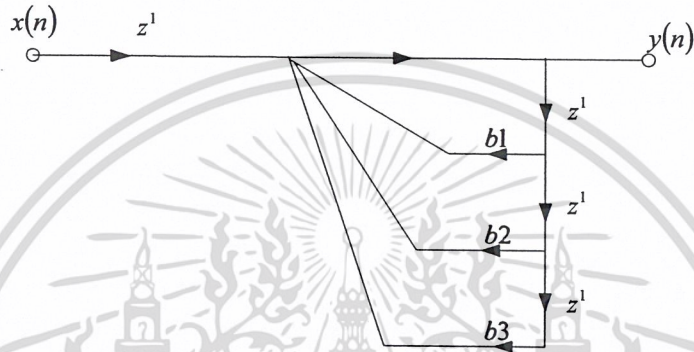
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เคชการสมมตสมการดั่งนี้

$$y(n) = x(n-1) - b_1y(n-1) + b_2(n-1) + b_3y(n-3) \tag{2.21}$$

บลลอกคโคะแกรมแสดงถงสมการ (2.21) จากรูปสลัญลคกษณ Z แสดงถงการหน่วงเวลาของช่วงเวลาหนึ่งหน่วย ซงอาจพิจารณาในโหนดตางๆ ถูกสรแสดงถงการคูณและค่าคงตัวที่ถัดมา ทำการคูณกับตัวประกอบ ความสัมพันธระหวางสมการผลตางสลับเนองและบลลอกคโคะแกรมจะค่อขยปรากฏซงและ Signal Flow Graph ก้ใช้แสดงกับสมการผลตางสลับเนองเช่นเดยวกันดงรูปที่ 2.29 โดยสมมตสมการดั่งนี้

$$y(n) = x(n-1) - b_1y(n-1) + b_2y(n-2) + b_3y(n-3) \tag{2.22}$$

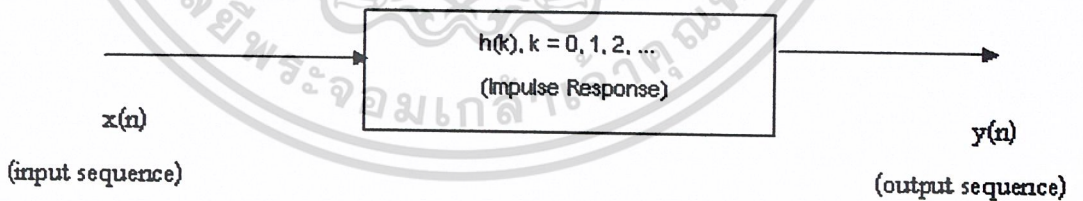


รูปที่ 2.29 การวิเคราะห์สมการผลตางสลับเนองในรูปแบบ Signal Flow Graph Diagram

2.21.3 ชนิดของตัวกรองดิจิตอล (Type of digital filter)

ตัวกรองดิจิตอลที่ใช้กันอย่างกว้างขวางแบ่งออกเป้น 2 ชนิด

1. Finite Impulse Response (FIR)
2. Infinite Impulse Response (IIR)



รูปที่ 2.30 บลลอกคโคะแกรมหลักการของตัวกรองดิจิตอล

ตัวกรองสัญญาณแต่ละชนิด จะมีรูปแบบพื้นฐานแต่ละตัวซึ่งสามารถแสดงลำดับของการตอบสนองต่ออิมพัลส์ $h(k); (k = 0, 1, 2, \dots)$ ดงรูปที่ 2.30 สัญญาณอินพุตและเอาท์พุทที่ตัวกรองสัญญาณจะถูกนำมาเกี่ยวพันโดยผลรวมของสมการ ซงสมการแต่ละชนิดแสดงได้ดงตอไปนี้

$$y(n) = \sum_{k=0}^{\alpha} h(k)x(n-k) \quad : \text{IIR digital filter} \tag{2.23}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(n) = \sum_{k=0}^{n-1} h(k)x(n-k) \quad : \text{ FIR digital filter} \quad (2.24)$$

จากสมการข้างต้นสำหรับตัวกรองดิจิทัลแบบไอโออาร์นั้น ผลการตอบสนองต่ออิมพัลส์จะอยู่ในช่วงที่ไม่จำกัด แต่สำหรับตัวกรองดิจิทัลแบบเอฟไออาร์จะมีผลตอบสนองอิมพัลส์อยู่ในช่วงที่กำหนด โดยเริ่มที่ $h(k)$ สำหรับตัวกรองดิจิทัลแบบเอฟไออาร์จะมีเพียง N ค่า ส่วนในการหาค่าเอาต์พุตของตัวกรองดิจิทัลแบบไอโออาร์ ไม่สามารถเป็นไปได้ เนื่องจากสมการมีความยาวของผลการตอบสนองอิมพัลส์นั้นยาวมาก (∞ : Infinite) ดังนั้นเราจึงแทนสมการของตัวกรองดิจิทัลแบบไอโออาร์ในรูปของรูปแบบการป้อนกลับ

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^{\infty} a_k x(n-k) - \sum_{k=0}^{\infty} b_k x(n-k) \quad (2.25)$$

โดยที่ a_k และ b_k เป็นสัมประสิทธิ์ของตัวกรองสัญญาณ ดังนั้นจากสมการข้างต้นคือสมการที่แตกต่างกันระหว่างตัวกรองดิจิทัลแบบเอฟไออาร์และตัวกรองดิจิทัลแบบไอโออาร์ จากสมการเหล่านั้นจะมีค่า $h(k)$ สำหรับตัวกรองดิจิทัลแบบเอฟไออาร์และ a_k และ b_k สำหรับตัวกรองดิจิทัลแบบไอโออาร์ ซึ่งทั้งสองสมการเป็นสิ่งสำคัญมากที่ใช้ในการออกแบบตัวกรองสัญญาณเป็นส่วนใหญ่ จากสมการกระแสตัวอย่างเอาต์พุต $y(n)$ เป็นฟังก์ชันของเอาต์พุตที่ผ่านมา ซึ่งจะทำให้เปรียบเทียบกับสมการของตัวกรองดิจิทัลแบบเอฟไออาร์ แล้วกระแสการสุ่มตัวอย่างเอาต์พุต $y(n)$ จะเป็นเพียงฟังก์ชันของค่าอินพุตของอดีตและปัจจุบันเท่านั้นเอง แต่อย่างไรก็ตาม เมื่อ b_k มีค่าเท่ากับ 0 สมการตัวกรองดิจิทัลแบบไอโออาร์ก็คือ สมการตัวกรองดิจิทัลแบบเอฟไออาร์นั่นเอง

และจากสมการข้างล่างนี้ก็คือ ฟังก์ชันถ่ายโอน (Transfer Function) ของตัวกรองดิจิทัลแบบเอฟไออาร์และตัวกรองดิจิทัลแบบไอโออาร์ ซึ่งค่าฟังก์ชันถ่ายโอนเหล่านี้จะมีประโยชน์มากในการคำนวณหาผลตอบสนองความถี่ (Frequency Response)

$$H(Z) = \frac{\sum_{k=0}^{\infty} a_k Z^{-k}}{1 + \sum_{k=1}^{\infty} b_k Z^{-k}} \quad : \text{ IIR digital filter} \quad (2.26)$$

$$H(Z) = \sum_{k=0}^{n-1} h(k)Z^{-k} \quad : \text{ FIR digital filter} \quad (2.27)$$

2.21.4 คุณสมบัติของตัวกรองสัญญาณดิจิทัลแบบเอฟไออาร์

- Characteristic ของตัวกรองสัญญาณดิจิทัลแบบเอฟไออาร์ จะแสดงดังสมการ

$$y(n) = \sum_{k=0}^{n-1} h(k)x(n-k) \quad (2.28)$$

$$H(Z) = \sum_{k=0}^{n-1} h(k)Z^{-k} \quad (2.29)$$

โดยที่ $h(k); k = 0, 1, 2, \dots, n-1$ เป็นสัมประสิทธิ์ในการตอบสนองของตัวกรองสัญญาณ

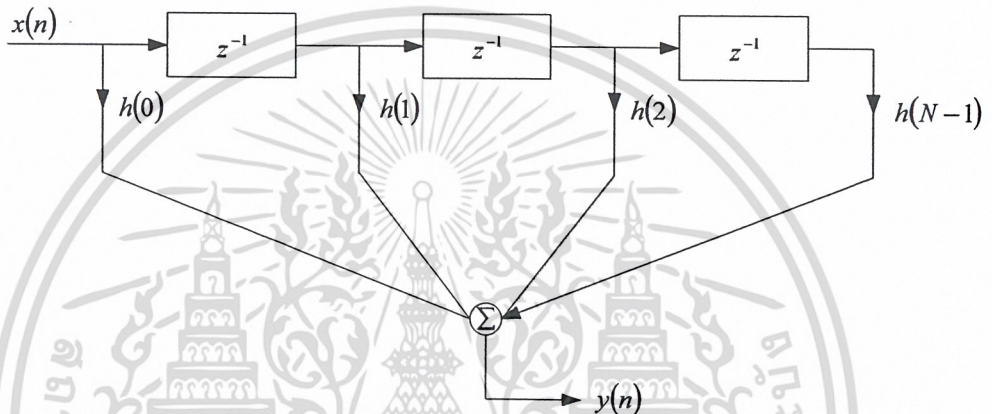
$H(Z)$ เป็นฟังก์ชันถ่ายโอนของตัวกรองสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั่นคือจำนวนสัมประสิทธิ์ของตัวกรองสัญญาณของสมการที่ (2.28) เป็น FIR DifferenceEquation ซึ่งมันจะเป็นสมการใน โดเมนเวลาและพิจารณาตัวกรองสัญญาณคิจิตอลแบบเอฟไออาร์ในรูปแบบของ Non-recursive form ซึ่งจะทำให้การสุ่มตัวอย่างเอาท์พุท $y(n)$ เป็นฟังก์ชันอินพุทในอดีตและปัจจุบันเท่านั้น $x(n)$ เมื่อตัวกรองสัญญาณถูกนำไปใช้ในรูปแบบนี้ นั่นก็จะเป็นการหาค่าโดยตรงจะได้ค่าคงที่ (Stable) และฟังก์ชันถ่ายโอนของตัวกรองสัญญาณ จะใช้ในการวิเคราะห์และหาค่าตัวกรองสัญญาณในทางด้านการตอบสนองทางความถี่

- ตัวกรองสัญญาณคิจิตอลแบบเอฟไออาร์มีการตอบสนองทางเฟสเป็นเชิงเส้น
- ตัวกรองสัญญาณคิจิตอลแบบเอฟไออาร์ง่ายที่จะนำไปใช้ในการประมวลผลสัญญาณ

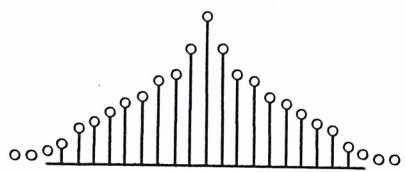
คิจิตอล



รูปที่ 2.31 โครงสร้างของตัวกรองสัญญาณคิจิตอลแบบเอฟไออาร์

2.22 ตัวกรองผลตอบสนองอิมพัลส์จำกัด (Finite Impulse Response Filter)

เป็นตัวกรองที่ผลตอบสนองอิมพัลส์ของมันมีค่าดับที่มีค่าแตกต่างกันไปจากศูนย์มีจำนวนพจน์คงที่ โดยทั่วไปมักจะเป็นตัวกรองแบบไม่มีการป้อนกลับของสัญญาณ ซึ่งตัวกรองที่ไม่มีการป้อนกลับสัญญาณนี้มีข้อคืออยู่ที่ สามารถออกแบบให้มีผลตอบสนองเฟสเป็นเชิงเส้นได้ง่ายและจะเสถียรเสมอ จึงเหมาะที่จะนำไปใช้ในงานที่มีผลตอบสนองทางเฟสมีความสำคัญมากๆ เช่น ในงานที่ต้องการให้ผลตอบสนองทางเฟสเป็นสัญญาณเชิงเส้น เพื่อไม่ให้ลักษณะรูปคลื่นของสัญญาณและเวลาในการเกิดสัญญาณผิดเพี้ยนไปได้



รูปที่ 2.32 ลักษณะสัญญาณของผลตอบสนองอิมพัลส์จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบตัวกรองชนิดนี้มีหลายวิธีด้วยกัน เช่น การออกแบบโดยใช้อนุกรมฟูรีเยอร์, การออกแบบโดยใช้วิธีวินโดว์แบบต่างๆ เช่น วินโดว์สี่เหลี่ยม, วินโดว์แบบฮาน แอมมิ่งและแบล็กแมนหรือใช้โคเซอร์วินโดว์ โดยจะกล่าวถึงวิธีออกแบบแต่ละวิธีดังนี้

2.22.1 การออกแบบตัวกรองแบบไม่ป้อนกลับโดยใช้อนุกรมฟูรีเยอร์

ตัวกรองแบบไม่ป้อนกลับนี้สามารถเขียนได้โดยสมการของอนุกรมฟูรีเยอร์ (Fourier series) ซึ่งในการคำนวณจะเห็นว่าความยาวของตัวกรองนั้นมีจำกัด ถ้าให้ $x(n)$ เป็นลำดับสัญญาณเข้าและ $y(n)$ เป็นลำดับสัญญาณออก เราสามารถเขียนสมการผลต่างสืบเนื่องได้ดังนี้

$$\begin{aligned} y(n) &= \sum_{k=-M}^m C_k x(n-k) \\ &= C_0 x(n) + \sum_{k=1}^{-M} C_k x(n-k) + \sum_{k=1}^M C_k x(n-k) \\ &= C_0 x(n) + \sum_{k=1}^M \{C_{-k} x(n-k) + C_k x(n-k)\} \end{aligned} \quad (2.30)$$

สมมติให้ $T = 1$ แล้วทำการแปลงแซค เพื่อเขียนฟังก์ชันถ่ายโอนได้เป็น

$$H(Z) = C_0 + \sum_{k=1}^M \{C_{-k} Z^k + C_k Z^{-k}\} = \sum_{k=-M}^M C_k Z^{-k}$$

หรือ

$$H(\omega) = C_0 + \sum_{k=1}^M \{C_{-k} e^{j\omega k} + C_k e^{-j\omega k}\} = \sum_{k=-M}^M C_k e^{-j\omega k} \quad (2.31)$$

เมื่อ ω คือ ค่าความถี่

ถ้าพิจารณากรณีตัวกรองสมมาตรคู่ หรือ $C_k = C_{-k}$

$$H(\omega) = C_0 + \sum_{k=1}^M C_k \{e^{j\omega k} + e^{-j\omega k}\} \quad (2.32a)$$

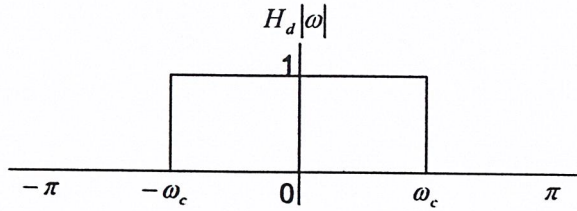
$$H(\omega) = C_0 + \sum_{k=1}^M 2C_k \{e^{j\omega k} + e^{-j\omega k}\} / 2 \quad (2.32b)$$

$$H(\omega) = a_0 + \sum a_k \cos \omega k$$

โดยกำหนดให้ $a_0 = C_0$ และ $a_k = C_k$ และ $k > 0$

ผลที่ได้จากสมการจะเห็นได้ว่า ตัวกรองไม่ป้อนกลับดิจิทัลชนิดสมมาตรคู่สามารถเขียนอธิบายโดยใช้อนุกรมฟูรีเยอร์ที่ประกอบด้วยค่าสัมประสิทธิ์เฉพาะโคไซน์เท่านั้นและค่าสัมประสิทธิ์ของอนุกรมฟูรีเยอร์ a_k เป็นค่าสัมประสิทธิ์ของตัวกรองด้วย ในทำนองเดียวกัน เราอาจแสดงได้ว่าสำหรับตัวกรองไม่ป้อนกลับดิจิทัลชนิดสมมาตรคู่สามารถเขียนแทนด้วยอนุกรมฟูรีเยอร์ที่มีเฉพาะพจน์ไซน์เท่านั้นและค่าสัมประสิทธิ์จะเป็นจำนวนจินตภาพเสมอ

โดยปกติแล้วการออกแบบตัวกรองสัญญาณนั้น กระทำโดยการทำการประมาณค่าให้ตัวกรองที่ ออกแบบมีผลตอบสนองความถี่ให้ใกล้เคียงกับผลตอบสนองอุดมคติ ดังรูปที่ 2.33 ให้มากที่สุดเท่าที่จะทำ ได้



รูปที่ 2.33 ผลตอบสนองอุดมคติ

จากรูป ω_c เป็นค่าความถี่ตัด (Cut off frequency) จากผลตอบสนองอุดมคติ $H_d(\omega)$ สามารถ กระจายให้อยู่ในพจน์ของโดเมนเวลา โดยการประยุกต์การตัดแปลงฟูรีเยอร์ได้ดังนี้

$$H_d(\omega) = \sum_{-\infty}^{\infty} h_d(k) e^{-jk\omega T} \quad (2.33)$$

โดยในกรณีนี้ $h_d(k)$ หาได้จาก

$$h_d(k) = (1/2\pi) \int_{-\omega_c/2}^{\omega_c/2} H_d(\omega) e^{jk\omega T} d\omega \quad (2.34)$$

จากสมการด้านบนจึงได้ว่า ผลตอบสนองความถี่ของตัวกรองไม่ป้อนกลับดิจิทัลมีค่าเป็น

$$H_d(\omega) = \sum_{k=-M}^M C_k e^{-jk\omega T} \quad (2.35)$$

เมื่อเปรียบเทียบสมการกันแล้วจะเห็นว่า ผลตอบสนองความถี่ของตัวกรองใกล้เคียงกับ ผลตอบสนองอุดมคติก็ต่อเมื่อ M มีค่าเป็นอนันต์ แต่ในทางปฏิบัติเป็นไปได้ยาก จึงมีการตัดปลายอนุกรม ฟูรีเยอร์ให้เหลือเท่าที่จำเป็นและจัดให้

$$C_k = h_d(k) \quad (2.36)$$

โดยที่ $k = -M, \dots, 0, \dots, M$ ส่วนความยาวของตัวกรอง $N = (2M + 1) = (n - 1) / 2$

จากผลการตอบสนองแอมพลิจูด พบว่าถ้า M หรือ N มีค่ามากมีผลทำให้

- ผลตอบสนองแอมพลิจูดใกล้เคียงผลตอบสนองอุดมคติมากขึ้น
- จำนวนลูกคลื่น (ripple) มากขึ้น โดยที่คาบของลูกคลื่นลดลง
- ผลตอบสนองแอมพลิจูดมีความคมมากขึ้นหรือบริเวณความถี่ที่ผลตอบสนองเปลี่ยนจากค่าหนึ่ง

เป็นค่าใกล้ศูนย์หรือที่เรียกว่าแถบเปลี่ยนสถานะ (Transition band) แคบลง

โดยทั่วไปในการออกแบบตัวกรองไม่ป้อนกลับดิจิทัล เราต้องการลดจำนวนลูกคลื่นให้น้อย ที่สุดและให้มีแถบเปลี่ยนสถานะกว้างขึ้น ดังนั้นในการออกแบบจึงจำเป็นต้องเลือกอย่างใดอย่างหนึ่ง สำหรับการออกแบบตัวกรองไม่ป้อนกลับดิจิทัลโดยวิธีเขียนฟูรีเยอร์ สรุปขั้นตอนการทำได้ดังนี้

- กำหนดผลตอบสนองทางอุดมคติที่ต้องการ โดยการกำหนดความถี่คัทออฟ

- ทำการอินทิเกรตผลตอบสนองอุดมคติ เมื่อหาค่าคัทออฟผลตอบสนองอิมพัลส์ $h_d(k)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลือกให้สัมประสิทธิ์ $C_k = h_d(k)$

2.22.2 การออกแบบตัวกรองแบบไม่ป้อนกลับโดยใช้วิธีหน้าต่าง (Window)

ในวิธีนี้เริ่มพิจารณาจากสมการพื้นฐานก่อน โดยเริ่มจาก $H_d(\omega)$ แทนผลตอบสนองความถี่อุดมคติและผลตอบสนองนี้สามารถเขียนแทนด้วยอนุกรมฟูริเยร์ยาวอนันต์พจน์ซึ่งมีสมการคือ

$$H_d(\omega) = \sum_{n=-\infty}^{\infty} h_d(n) e^{-jk\omega T} \quad (2.37)$$

โดยที่พัลส์ $h_d(n)$ หาได้จาก

$$h_d(n) = \pi \int_{-\pi}^{\pi} H_d \omega e^{jn\omega T} d\omega \quad (2.38)$$

โดยที่ค่า ω_s นี้เป็นค่าความถี่ในการสุ่มตัวอย่าง ในการออกแบบตัวกรองไม่ป้อนกลับดิจิทัลที่ผ่านมาระบุว่าเกิดปัญหาคือ 2 ประการ คือ

ประการแรก เราต้องทำการตัดปลายของอนุกรมให้มีจำนวนพจน์จำกัดหรือ

$$H_d(\omega) = \sum_{n=-(N-1)/2}^{(N-1)/2} h_d(n) e^{-jn\omega T} \quad (2.39)$$

ซึ่งผลลัพธ์ก็คือ ทำให้เกิดลูกคลื่นบนผลตอบสนองแอมพลิจูด

ประการที่สอง เมื่อตัดปลายอนุกรมแล้วยังไม่สามารถนำไปใช้งานได้ เพราะผลตอบสนองอิมพัลส์เริ่มจากลำดับที่ $n = -(N-1)/2$ ซึ่งในระบบเวลาจริง ไม่สามารถสร้างลำดับที่เวลาเป็นลบได้ จึงทำการแก้ไขโดยการหวนผลตอบสนองอิมพัลส์ออกไป โดยเริ่มจากค่า $n = 0$ หรือ

$$H_d(\omega) = \sum_{n=0}^{N-1} h_d(n) e^{-jn\omega T} \quad (2.40)$$

จะเห็นได้ว่าการตัดปลายอนุกรมฟูริเยร์นั้น ก็คือการที่เราเลือกให้

$$h(n) = h_d(n) \quad (2.41)$$

ซึ่งสมการนี้ ถ้าหากเขียนให้อยู่ในรูปของสมการทั่วไปก็คือ การนำเอาลำดับมาคูณกับค่าลำดับจำกัด หรือ

$$h(n) = h_d(n) w(n) \quad (2.42)$$

โดยที่ $w(n)$ แทนลำดับของวินโดว์แบบต่างๆ สำหรับสมการที่ผ่านมานั้นเป็นวินโดว์สี่เหลี่ยม (Rectangular window) ที่ลำดับ มีคุณสมบัตินิยามได้เป็น

$$\begin{aligned} w_R(n) &= 1 && ; 0 \leq n \leq N-1 \\ &= 0 && ; \text{otherwise} \end{aligned} \quad (2.43)$$

2.22.2.1 วินโดว์สี่เหลี่ยม (Rectangular window)

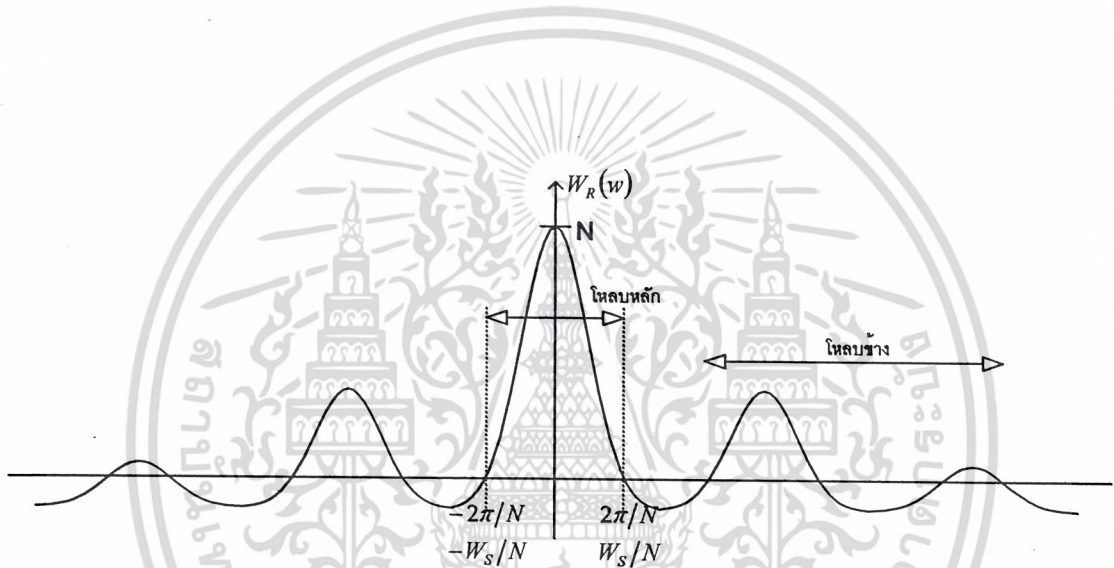
อาณานิยามโดยทั่วไปได้ดังนี้

$$\begin{aligned} w_R(n) &= 1 && ; |n| \leq (N-1)/2 \\ &= 0 && ; \text{otherwise} \end{aligned} \quad (2.44)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลตอบสนองความถี่ของวินโดว์แบบนี้หาได้โดยการประยุกต์ใช้ผลการแปลงแซดเข้า กับสมการแล้วทำการแทนค่าให้ $Z = e^{j\omega T}$ เมื่อจัดรูปสมการใหม่จะได้

$$\begin{aligned} W_R(\omega) &= \sum_{n=-(N-1)/2}^{(N-1)/2} e^{-jn\omega T} \\ &= \frac{\{e^{j\omega(N-1)T/2} - e^{-j\omega(N-1)T/2}\}}{\{1 - e^{-j\omega T}\}} \\ &= \frac{\{e^{j\omega NT/2} - e^{-j\omega NT/2}\}}{\{e^{j\omega T/2} - e^{-j\omega T/2}\}} \\ &= \frac{\sin(\omega NT/2)}{\sin(\omega T/2)} \end{aligned} \quad (2.45)$$



รูปที่ 2.34 ผลตอบสนองแอมพลิจูดของวินโดว์สี่เหลี่ยม

จากรูปที่ 2.34 จากผลตอบสนองแอมพลิจูดของวินโดว์สี่เหลี่ยม ซึ่งพบว่ามีความสมบัติ ดังนี้

- ความกว้างของ Main lobe มีค่า $4\pi/N$ และมีค่าขนาดเป็น N
- เมื่อ N มีค่ามากขึ้น ความกว้างของ Main lobe ลดลง
- Side lobe มีการแกว่งโดยผ่านจุดศูนย์ทุกๆค่าความถี่และมีขนาดเล็กลงไปเรื่อยๆ

และ $W_R(\omega) = 0$ ณ ค่าความถี่ $\omega = m\omega_s/N$ เมื่อ m เป็นค่าบวกและค่าลบของ $1, 2, \dots$ ผลนี้เป็นที่น่าพอใจ ถ้า N มีค่ามาก โหลบข้างจะเข้ามารวมกันใกล้ความถี่ $\omega = 0$ มากขึ้น

- ถ้านิยามให้พจน์อัตราถูกคลื่น (Ripple Ratio) หรือ RR เป็น

$$RR = \frac{\text{ค่าขนาดสูงสุดของโหลบข้าง} * 100}{\text{ค่าขนาดสูงสุดของโหลบหลัก}} \quad (2.46)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การที่นำเอาลำดับผลตอบสนองอิมพัลส์มาคูณกับลำดับของวินโดว์ ซึ่งเป็นการทำในโดเมนเวลา เมื่อพิจารณาในโดเมนความถี่ ก็เท่ากับเป็นการนำเอาผลตอบสนองความถี่อูคมคติมาทำการคูณประสานกับผลตอบสนองความถี่ของวินโดว์ หรือ

$$H(\omega) = (1/2\pi) \int_{-\pi}^{\pi} H_d e^{j\theta} W_R e^{j(\omega-\theta)} d\theta \quad (2.47)$$

ผลประสานนี้อธิบายได้ดังรูปที่ 2.35 ซึ่งจะเห็นได้ว่า

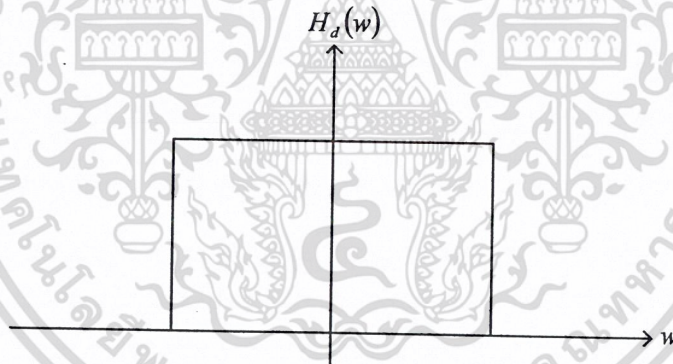
- ค่าความชันของโพลหลักของสเปกตรัมของวินโดว์ ทำให้เกิดแถบเปลี่ยนสถานะของผลตอบสนองความถี่ ดังนั้น ถ้า N มากขึ้นก็จะทำให้แถบเปลี่ยนสถานะแคบลง

- ลูกคลื่นบนผลตอบสนองความถี่เป็นผลมาจากโพลข้างของสเปกตรัมวินโดว์ ดังนั้น ถ้าเลือกค่า N ค่ามาก โพลข้างเลื่อนเข้าใกล้ความถี่ $\omega = 0$ มากขึ้น เป็นผลให้เกิดลูกคลื่นบนยอดของผลตอบสนองความถี่มากขึ้น

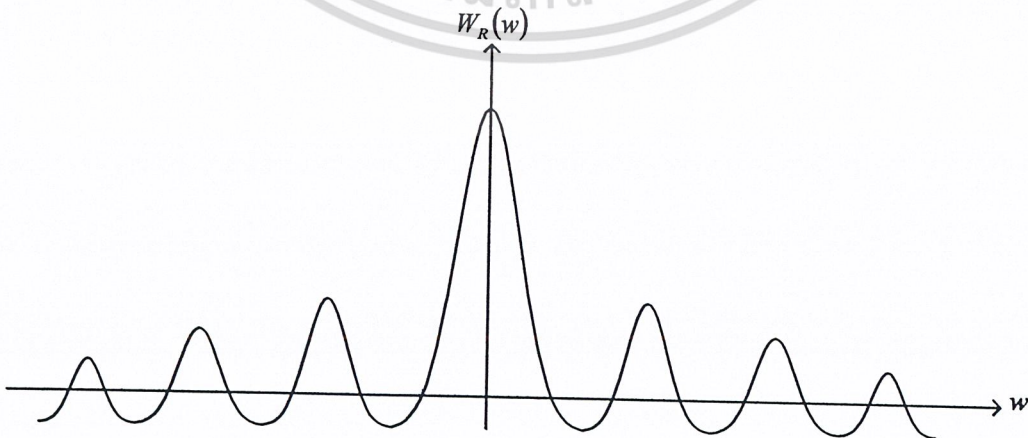
ด้วยผลทั้งสองประการนี้ เราจึงต้องหาวินโดว์หรือตัดแปลงวินโดว์ที่เหลี่ยมให้มีคุณสมบัติ คือ

- ให้โพลหลักของผลตอบสนองความถี่แคบ แต่ภายในโพลหลักสามารถครอบคลุมสัญญาณทั้งหมดไว้ให้ได้มากที่สุด

- ให้ผลตอบสนองความถี่ของวินโดว์มีขนาดของโพลข้างน้อยและมีค่าขนาดลดลงอย่างรวดเร็วเมื่อความถี่สูงขึ้น

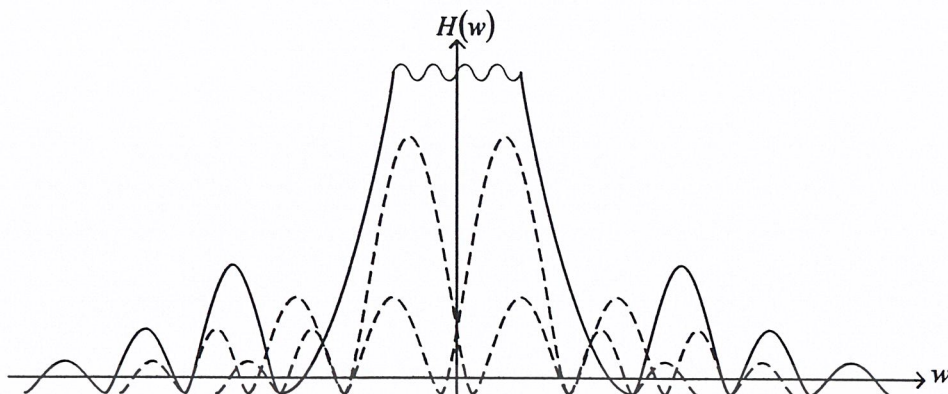


รูปที่ 2.35(a) วิธีการทำวินโดว์ในโดเมนความถี่



รูปที่ 2.35(b) วิธีการทำวินโดว์ในโดเมนความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.35(c) วิธีการทำวินโดว์ในโดเมนความถี่

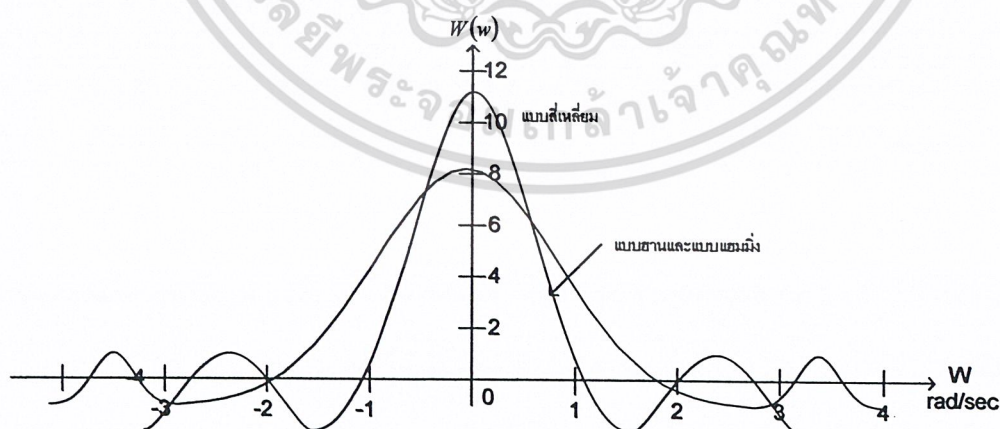
2.22.2.2 วินโดว์แบบฮานและแฮมมิง (Hann and Hamming window)

วินโดว์แบบฮาน (Hann window) และวินโดว์แบบแฮมมิง (Hamming window) สามารถเขียนเป็นสมการของลำดับได้ โดยให้ $T = 1$ วินาที คือ

$$w_H(n) = \alpha + \{1 - \alpha\} \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2 \quad = 0$$

$$; \text{otherwise} \quad (2.48)$$

โดยที่วินโดว์แบบฮานและวินโดว์แบบแฮมมิง ดังในรูปที่ 2.36 ซึ่งแสดงการเปรียบเทียบกันระหว่างความถี่ของวินโดว์สามแบบด้วยกันคือ วินโดว์แบบสี่เหลี่ยม, วินโดว์แบบฮานและวินโดว์แบบแฮมมิง ซึ่งจะเห็นได้ว่าวินโดว์แบบสี่เหลี่ยมมีพลังงานที่อยู่นอกโหลบหลักอยู่มาก ส่วนวินโดว์แบบฮานและวินโดว์แบบแฮมมิง พลังงานได้ถ่ายเข้าไปในโหลบหลักมากถึงเกือบ 99 %



รูปที่ 2.36 สเปกตรัมของวินโดว์แบบสี่เหลี่ยม วินโดว์แบบฮานและวินโดว์แบบแฮมมิง กรณี $N = 11$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.22.2.3 วินโดว์แบบแบล็กแมน (Blackman window)

$$w_m(n) = 0.42 + 0.5 \cos\left(\frac{2n\pi}{N-1}\right) + 0.08 \cos\left(\frac{4n\pi}{N-1}\right) \quad ; |n| \leq (N-1)/2$$

$$= 0 \quad ; \text{otherwise} \quad (2.49)$$

วินโดว์แบบนี้ให้คุณสมบัติดีกว่าวินโดว์แบบฮานและวินโดว์แบบแฮมมิง คือสามารถถ่ายทอดพลังงานเข้าไปในโหลบหลักได้มากกว่า ดังแสดงตารางที่ 2.3 และ 2.4

ชนิดของวินโดว์	Main lobe width	Ripple Ratio % N = 31	Peak-amplitude of Side lobe	Main-stop band Attenuation(dB)
Rectangular	$2\omega_s / N$	21.80	-13	-21
Hann	$4\omega_s / N$	2.67	-31	-44
Hamming	$4\omega_s / N$	0.82	-41	-53
Blackman	$6\omega_s / N$	0.12	-57	-74

ตารางที่ 2.3 คุณสมบัตินของวินโดว์แบบต่างๆ

ชนิดของวินโดว์	Ripple Ratio		
	N = 11	N = 21	N = 31
Rectangular	22.34	21.89	21.8
Hann	2.62	2.67	2.67
Hamming	1.47	0.93	0.82
Blackman	0.08	0.12	0.12

ตารางที่ 2.4 คุณสมบัตินของวินโดว์แบบต่างๆ เมื่อมีอินดิสต่างๆ

2.23 ความรู้เบื้องต้นเกี่ยวกับบอร์ด DSP TMS320C6211

2.23.1 บทนำ

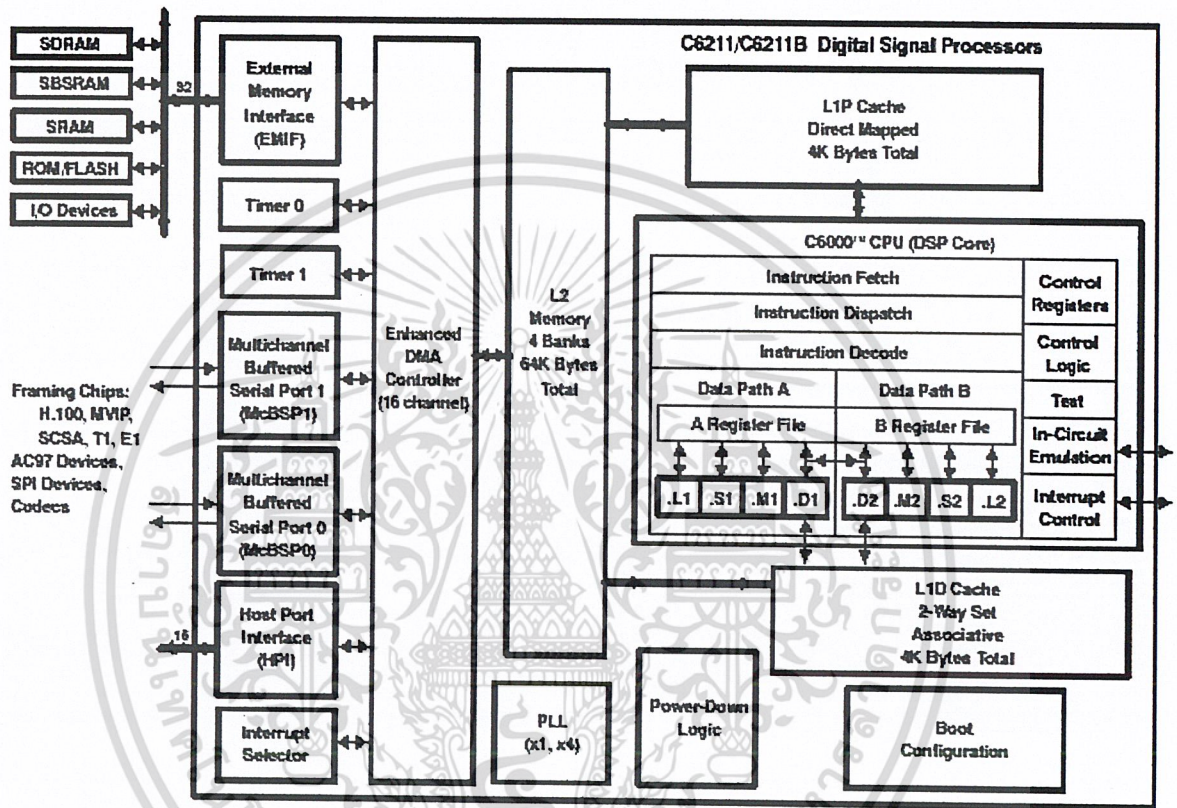
ในชิ้นงานนี้จะนำเสนอโดยผ่านบอร์ด TMS320C6211 ของ TEXAS INSTRUMENT โดยบอร์ดชนิดนี้มีโปรเซสเซอร์ตระกูล TMS320 ซึ่งถูกสร้างและออกแบบเพื่อตอบสนองความต้องการทางด้านการประมวลผลสัญญาณทางดิจิทัลซึ่งมีการนำมาใช้ในทางสื่อสารมากมายเพราะเป็นโปรเซสเซอร์ที่ประมวลผลบนพื้นฐานของคณิตศาสตร์ขั้นสูงและการแก้ปัญหาทาง Hardware และ Software

โดยโปรเซสเซอร์ TMS320C6211 นั้นสร้างมาจากเทคโนโลยีโครงสร้างแบบ VLIW (Very-Long-Instruction-Word) ซึ่งจะทำให้สนับสนุนในการพัฒนามาถึงการใช้อุปกรณ์ระดับสูงซึ่งบอร์ดนี้สามารถมีคอมไพเลอร์ที่ใช้ภาษา C/C++ ได้โดยใช้โปรแกรม CODE COMPOSER เป็นตัวคอมไพล์, โปรแกรมและโปรเซสเซอร์เป็นแบบ 32 บิต แบบ Fixed-point

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประสิทธิภาพอันสูงของ TMS320C6x เกิดจากความแน่นอนและกว้างขวางทางไดนามิกของส่วนคำนวณทางทศนิยมมีหน่วยความจำขนาดใหญ่และมี DMA Controller เพื่อให้ทำงานได้อย่างมีประสิทธิภาพได้ดีขึ้นซึ่งทำให้นิยมนำมาประยุกต์ใช้ในระบบสื่อสาร การประมวลผลเสียงและภาพแบบดิจิทัล

functional block and CPU (DSP core) diagram



รูปที่ 2.37 สถาปัตยกรรมของของ TMS320C6211

2.23.2 ภาพรวมของสถาปัตยกรรมของ TMS320C6211

TMS320C62X ประกอบไปด้วยหน่วยความจำภายในและตัวติดต่อภายนอก (serial port และตัวเชื่อมต่อหน่วยความจำภายนอก เป็นต้น) และส่วนที่สำคัญที่สุดคือตัวรีจิสเตอร์และหน่วยฟังก์ชันที่ทำหน้าที่ประมวลผลคำสั่งซึ่งจะเห็นจากรูปที่ 2.37 เป็นโครงสร้างภายในของ CPU และตัวเชื่อมต่อภายนอกต่างๆ

2.23.2.1 ส่วนกลางของการปฏิบัติการ (Core DSP Operation)

ส่วนนี้จะเป็นส่วนประมวลผลกลางโดยจะทำการแบบ Sum of Product หรือแบบ MAC (Multiply-Accumulate) และตัว Multiplier 2 ตัว แต่ละตัวซึ่งแต่ละตัวปฏิบัติการแบบ 16 บิต แต่ละหนึ่งรอบของสัญญาณนาฬิกา

2.23.2.2 รีจิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลที่เป็นตัวเลขที่เข้ามาใน CPU นั้น เมื่อผ่านการปฏิบัติการแล้วก็จะถูกจัดเก็บไว้ในรีจิสเตอร์เพราะรีจิสเตอร์มีการติดต่อโดยตรงกับบัสของ CPU จึงทำให้การเก็บค่าลงในรีจิสเตอร์นั้นเร็วกว่าการติดต่อกับหน่วยความจำภายนอก

CPU C62x จะมีรีจิสเตอร์สองตัวคือ A และ B โดยแต่ละตัวมีรีจิสเตอร์แบบ 32 บิต จำนวน 16 ตัว (A0-A15, B0-B15) ซึ่งสามารถนำไปใช้เก็บข้อมูลหรือตัวชี้แอสเซสเซอร์ข้อมูล

รีจิสเตอร์นั้นสามารถเก็บข้อมูลได้ระหว่าง 16 บิต ถึง 40 บิต ถ้าข้อมูลมีค่าเกิน 32 บิต เช่น 40 บิต จะมีการเก็บข้อมูลเป็นรีจิสเตอร์คู่คือจะเก็บ 32 บิตแรก (LSBs) ไว้ก่อนแล้วนำอีก 8 บิตหลังมาเก็บไว้อีกในรีจิสเตอร์ต่อไป

2.23.2.3 หน่วยฟังก์ชัน

CPU C62x นั้น มีตัวฟังก์ชันที่เป็นตัวบอกการโอเปอร์เรตหรือการบอกการปฏิบัติการและในรีจิสเตอร์มีอยู่ 4 หน่วยฟังก์ชัน คือ .M, .L, .S และ .D ที่เห็นในรูปที่ 2.37 หน่วยฟังก์ชันทั้ง 4 ก็มีการติดต่อกับรีจิสเตอร์ A และ B โดย A มีการติดต่อกับ .M1, .L1, .S1 และ .D1 และ B ติดต่อกับ .M2, .L2, .S2 และ .D2

โดยแต่ละหน่วยฟังก์ชันมีหน้าที่

.M	multiply operation
.L	logical and arithmetic
.S	branch, bit manipulation (โยกย้าย) and arithmetic operations
.D	loading/storing and arithmetic operations
.S, .L	ใช้ใน arithmetic, logical and branch instructions ถ้าในการส่งข้อมูลก็จะใช้ .D

2.23.3 การแม็บบิเมโมรี

การแม็บบิเมโมรีเป็นการจำเป็นก่อนที่จะมีการเขียนโปรแกรมในส่วนต่างๆเราจึงต้องรู้ส่วนของเมโมรีต่างๆดังตารางที่ 2.5 ดังนี้

MEMORY BLOCK DESCRIPTION	BLOCK SIZE (BYTES)	HEX ADDRESS RANGE
Internal RAM (L2)	64K	0000 0000 – 0000 FFFF
Reserved	24M – 64K	0001 0000 – 017F FFFF
External Memory Interface (EMIF) Registers	256K	0180 0000 – 0183 FFFF
L2 Registers	256K	0184 0000 – 0187 FFFF
HPI Registers	256K	0188 0000 – 018B FFFF
McBSP 0 Registers	256K	018C 0000 – 018F FFFF
McBSP 1 Registers	256K	0190 0000 – 0193 FFFF
Timer 0 Registers	256K	0194 0000 – 0197 FFFF
Timer 1 Registers	256K	0198 0000 – 019B FFFF
Interrupt Selector Registers	256K	019C 0000 – 019F FFFF
EDMA RAM and EDMA Registers	256K	01A0 0000 – 01A3 FFFF
Reserved	6M – 256K	01A4 0000 – 01FF FFFF
QDMA Registers	52	0200 0000 – 0200 0033
Reserved	736M – 52	0200 0034 – 2FFF FFFF
McBSP Q/I Data	256M	3000 0000 – 3FFF FFFF
Reserved	1G	4000 0000 – 7FFF FFFF
EMIF CE0 [†]	256M	8000 0000 – 8FFF FFFF
EMIF CE1 [†]	256M	9000 0000 – 9FFF FFFF
EMIF CE2 [†]	256M	A000 0000 – AFFF FFFF
EMIF CE3 [†]	256M	B000 0000 – BFFF FFFF
Reserved	1G	C000 0000 – FFFF FFFF

[†] The number of EMIF address pins (EA[21:2]) limits the maximum addressable memory (SDRAM) to 128MB per CE space. To get 256MB of addressable memory, additional general-purpose output pin or external logic is required.

ตารางที่ 2.5 memory map

2.23.4 ชุดคำสั่งของ TMS320C6x

2.23.4.1 รูปแบบของแอสเซมบลีโค้ด

Label I [] Instruction Unit Operands; comment

2.23.4.2 ประเภทของคำสั่งต่างๆ

2.23.4.2.1 Add/Subtract/Multiply

- a) ADD .L1 A3, A7, A7 ; A3+A7 แล้วนำไปเก็บไว้ใน A7
 b) SUB .S1 A1, 1, A1 ; A1-1 เก็บไว้ใน A1
 c) MPY .M2 A7, B7, B6 ; คูณ 16 LSBs ของ A7, B7 เก็บไว้ใน B6
 II MPYH .M2 A7, B7, A6 ; คูณ 16 MSBs ของ A7, B7 เก็บไว้ใน A6

2.23.4.3 Load/Save

- a) LDH .D2 *B2++,B7 ; Load (B2) ไปใน B7 แล้วเพิ่ม B2
 LDH .D1 *A2++,A7 ; Load (A2) ไปใน A7 แล้วเพิ่ม A2
 b) STW .D2 A1, *+A4[20] ; จัดเก็บ A1 ไปใน A4 offset ด้วย 20

2.24.4.4 Branch/Move

- Loop MVK .S1 x, A4 ; ย้าย 16 LSBs ของแอสเซมบลีคำสั่ง x ไปยัง A4
 LoopMVKH .S1 x, A4 ; ย้าย 16 MSBs ของแอสเซมบลีคำสั่ง x ไปยัง A4
 SUB .S1 A1, 1, A1 ; ลดค่า A1
 [A1] B .S1 Loop ; Branch ไปยัง Label Loop ถ้า A1 ไม่เท่ากับ 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOP 5 ; 5 no operation
 STW .D1 A3, *A7 ; จัดเก็บ A3 ลง A7

.L Unit	.M Unit	.S Unit	.D Unit
ABS	MPY	ADD	ADD
ADD	MPYH	ADDK	ADDAB
ADDU	MPYHL	ADD2	ADDAH
AND	MPYHLU	AND	ADDAW
CMPEQ	MPYHSLU	B disp	LDB
CMPGT	MPYHSU	B IRP ^a	LDBU
CMPGTU	MPYHU	B NRP ^a	LDH
CMPLT	MPYHULS	B reg	LDHU
CMPLTU	MPYHUS	CLR	LDW
LMBD	MPYLH	EXT	LDB (15-Bit offset) ^b
MV	MPYLHU	EXTU	LDBU(15-Bit offset) ^b
NEG	MPYLSHU	MV	LDBU(15-Bit offset) ^b
NORM	MPYLUHS	MVC ^a	LDH(15-Bit offset) ^b
NOT	MPYSU	MVK	LDBU(15-Bit offset) ^b
OR	MPYU	MVKH	MV
SADD	MPYUS	MVKLH	STB
SAT	SMPY	NEG	STH
SSUB	SMPYH	NOT	STW
SUB	SMPYHL	OR	STB(15-Bit offset) ^b
SUBU	SMPYLH	SET	STH(15-Bit offset) ^b
SUBC		SHL	STW(15-Bit offset) ^b
XOR		SHR	SUB
ZERO		SHRU	SUBAB
		SSHL	SUBAH
		SUB	SUBAW
		SUBU	ZERO
		SUB2	
		XOR	
		ZERO	

^a S2 Only ^b D2 Only

ตารางที่ 2.6 ชุดคำสั่งต่างๆ

2.24 โปรแกรม Code Composer Studio (CCS)

โปรแกรมนี้เป็นการโปรแกรมที่พัฒนามาใช้งานบนวินโดวส์ซึ่งเป็นตัวคอมไพล์เรอร์ภาษาซี แล้วใช้การเชื่อมโยงกับภาษาแอสเซมบลี โดยโปรแกรมนี้สามารถเขียนได้ทั้งภาษาซีและแอสเซมบลีและโปรแกรมนี้ยังสามารถพล็อตกราฟเป็นแบบกราฟฟิคได้

โปรแกรมนี้ทำการติดต่อกับบอร์ดผ่านพอร์ต LPT1 คือ โปรแกรมจะทำการโปรแกรมข้อมูลที่เราเขียนทั้งหมดลงบอร์ดและยังสามารถควบคุมผ่านพอร์ตได้อีก

โดยการใช้บอร์ดกับโปรแกรมวิธีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Hardware Setup → Code & Build → Program → Test

2.24.1 Hardware Setup

2.24.1.1 ลงโปรแกรม CCS

2.24.1.2 ทำการ Setup CCS โดยแน่ใจว่าบอร์ดนั้นเสียบกับพอร์ต LPT 1 อยู่แล้วให้เช็คที่ Bios ว่าเซตพอร์เป็นแบบ ECP หรือ EPP หรือไม่ และต่อมาทำการ Run โปรแกรม Setup CCS ทำการเลือกพอร์ตให้ตรงกับ Bios

2.24.1.3 ทำการ reset บอร์ดด้วยการกดปุ่ม reset ที่บอร์ดหรือเสียบหรือไฟ Stand by ต้องเป็น โบนารี 1-7

2.24.1.4 ทำการ Run CCS

2.24.2 Code & Build

2.24.2.1 ไฟล์แต่ละชนิด

- file.mak เป็นไฟล์โปรเจกที่เราสร้างขึ้น
- file.c เป็นไฟล์ C code ที่เราเขียน
- file.asm เป็นไฟล์ assembly code ที่เราเขียน หรือเกิดจากคอมไพล์จาก C code
- file.h เป็นไฟล์ header ที่ไว้ #include ใน C
- file.lib เป็นไฟล์ ที่เราต้องแอดเพราะเป็น library ที่ไว้สนับสนุน ในบอร์ดรุ่นนี้ต้องโหลดไฟล์ rts6201.lib
- file.cmd เป็น link ไฟล์ ที่เราต้องเขียนเพื่อทำการ map memory
- file.obj เป็น object ไฟล์ ที่สร้างโดย Assembly code
- file.out เป็นไฟล์สุดท้ายเมื่อทำการ build ไฟล์ทั้งหมดเป็นไฟล์ที่ไว้โปรแกรม
- file.gel General Extension Language เขียนคล้ายกับภาษาซีเขียน เพื่อเป็นการปรับเปลี่ยนค่าแอมพลิจูดต่างๆด้วยไฮลด์บาร์

2.24.2.2 การสร้างโปรแกรม การคอมไพล์ และการ Build

2.24.2.2.1 ทำการสร้างโปรเจกโดยควรสร้าง Subfolder รอไว้ก่อนแล้วไปที่ Project → New ตั้งชื่อ save ลง Subfolder ที่ตั้งไว้

2.24.2.2.2 ทำการ add Library โดยไปที่ Add file to Project ไปที่ Folder

C:\ti\c6000\cgtools\lib\rts6201.lib

2.24.2.2.3 ทำการ add file header โดยไปที่ Add file to Project ไปที่ Folder

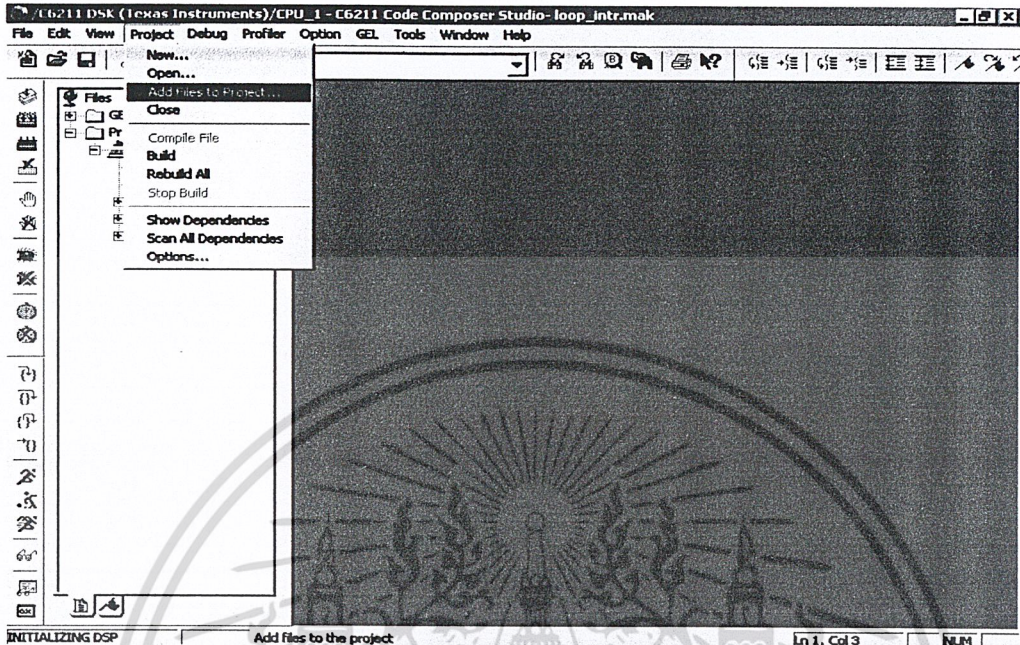
C:\ti\c6000\cgtools\include\c6x.h

2.24.2.2.4 ทำการเขียน link file file.cmd เป็นไฟล์ที่ทำการแม็บเมโมรี่โดยการ


เขียนเป็นภาษาแอสเซมบลี แล้วทำการ add file เข้าโปรเจก

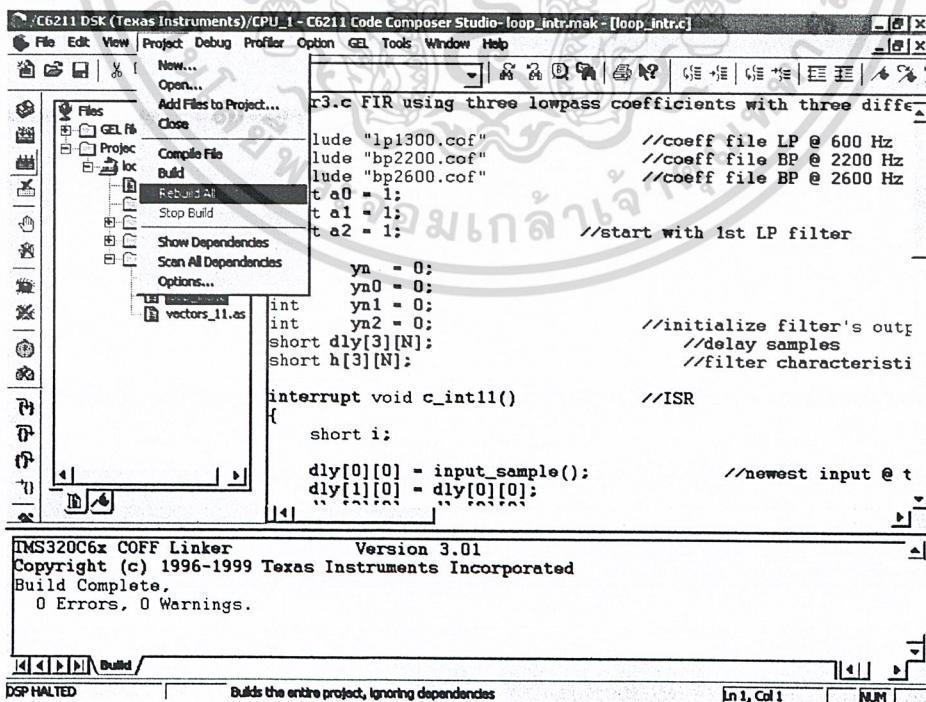
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.24.2.2.5 ต่อไปก็ทำการเขียน File assembly และ C เพื่อให้ทำงานตามที่เรากออกแบบ แล้วทำการ add file เข้าโปรเจก



รูปที่ 2.38 การ Add file to project

2.24.2.2.6 ทำการ Build all เพื่อสร้างไฟล์.out โดยไปที่ Project → Rebuild all เพื่อ Build ทุกไฟล์หรือ click 



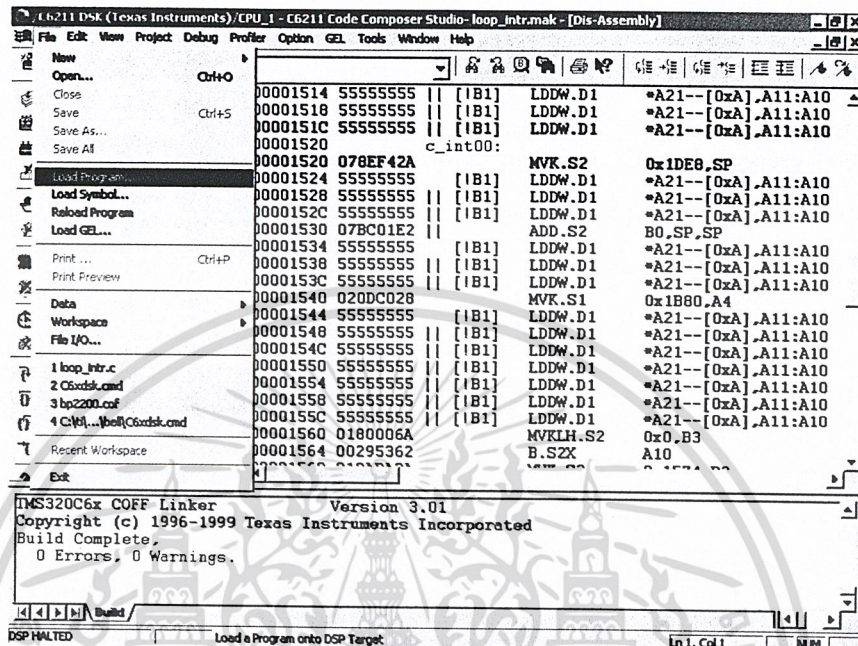
รูปที่ 2.39 การ Rebuild all

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.24.3 การโปรแกรม (Program)

เป็นการนำไฟล์ที่ได้เป็นไฟล์สุดท้ายนั้น โปรแกรมลงบอร์ดดังนี้

นำ file.out ไปทำการโปรแกรม โดยไปที่ file → Load Program



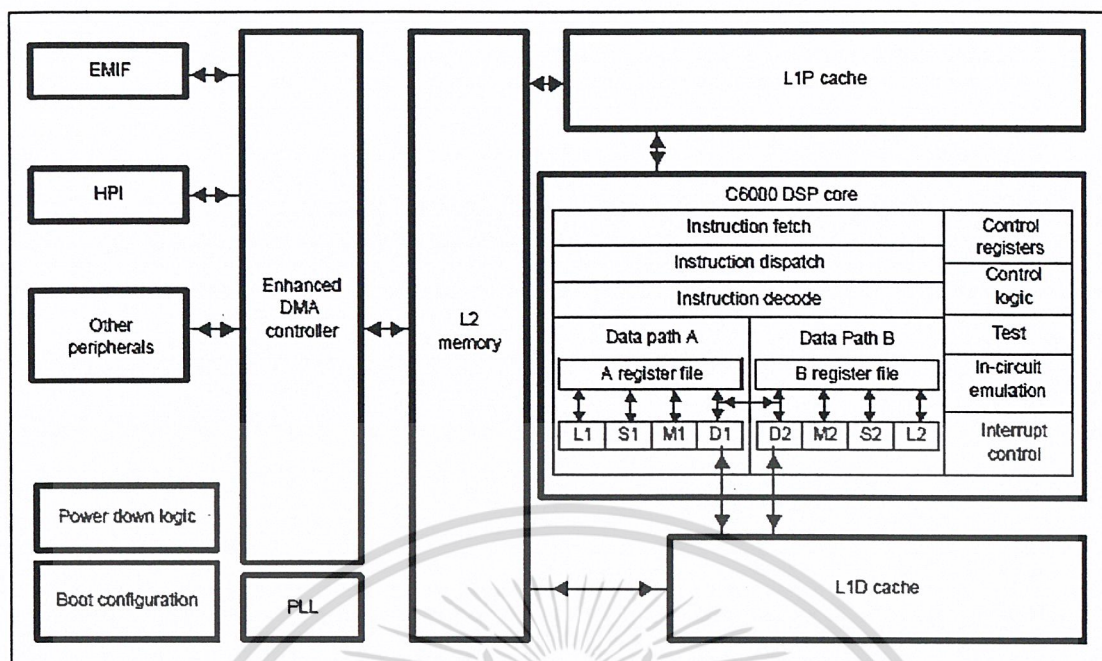
รูปที่ 2.40 การโปรแกรม

2.24.4 Test

ทำการทดสอบโดยการป้อนสัญญาณ ที่ Input Mic ของบอร์ด (JP7) และวัดสัญญาณ output speaker (JP6)

2.25 การเขียนโปรแกรมติดต่อกับบอร์ด

Host Port Interface (HPI) คือพอร์ทขนาน 16 บิตที่ต่อเชื่อมกับเครื่องคอมพิวเตอร์และต่อเข้าถึงหน่วยความจำโดยตรง ในอุปกรณ์ของเครื่องโฮสจะมีฟังก์ชันหลักในการเชื่อมต่อซึ่งจะทำให้การติดต่อเป็นไปได้ง่ายขึ้น เครื่องโฮสและหน่วยความจำของซีพียูสามารถแลกเปลี่ยนข้อมูลโดยทางหน่วยความจำภายในและหน่วยความจำภายนอก เครื่องโฮสยังสามารถเข้าถึงหน่วยความจำของอุปกรณ์ภายนอกได้ การเชื่อมต่อหน่วยความจำของซีพียูเตรียมไว้สำหรับตัวควบคุม DMA การแบ่งแอดเดรสรีจิสเตอร์และค่ารีจิสเตอร์ทำให้ไม่สามารถเข้าไปติดต่อกับซีพียูได้จึงต้องใช้ช่อง DMA ที่สำรองเอาไว้ ซึ่งต่อกับตัวควบคุมของ HPI เรียกว่า *HPIC* ,โฮสที่สามารถเข้าถึงที่อยู่ของ HPI เรียกว่า *HPIA* และโฮสที่สามารถเข้าถึงข้อมูลของ HPI เรียกว่า *HPID* *HPIC* จะใช้ข้อมูลจากภายนอกเพื่อควบคุมการเชื่อมต่อของสัญญาณ

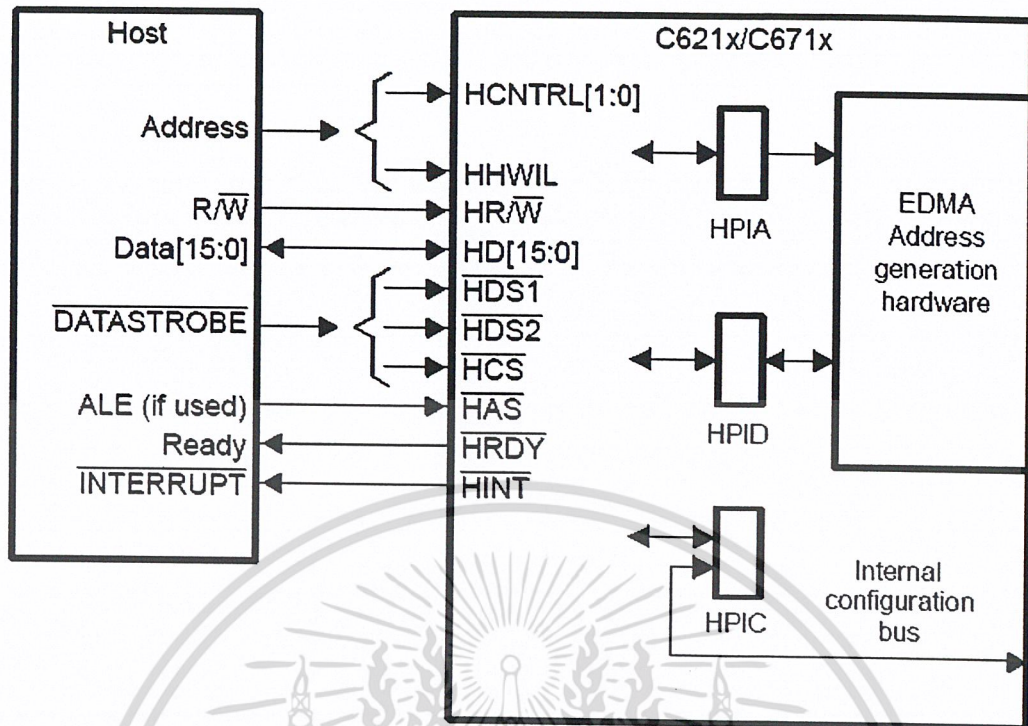


รูปที่ 2.41 บล็อกไดอะแกรมของ TMS320C6211

HPI ของ C6211 อนุญาตให้โฮสภายนอกเข้ามาอ่านและเขียนข้อมูลจาก C6211 ในช่องทางที่วางอยู่ HPI ของ C6211 เชื่อมโยงโดยตรงกับอุปกรณ์ที่อยู่ภายใน โดยเฉพาะช่อง EDMA ที่ถูกใช้ในการดำเนินการเข้าถึง HPI ของ C6211 อุปกรณ์ภายในจะเป็นตัวรับผิดชอบคำสั่งในการอ่านและเขียนด้วย

HPI มี 32 บิตที่เป็นบิตข้อมูลต่อกับซีพียูจะมี 16 บิตสำหรับการเชื่อมต่อภายนอกโดยอัตโนมัติเป็นการรวมกันกับ 16 บิตที่ใช้สำหรับการส่งผ่านข้อมูล เมื่ออุปกรณ์ของโฮสถ่ายโอนข้อมูลผ่านทาง HPID ช่อง DMA ที่สำรองจะเข้าถึงตำแหน่งที่อยู่ที่อยู่ของซีพียู

บัสข้อมูล 16 บิต $HD[15:0]$ แลกเปลี่ยนข้อมูลข่าวสารกับเครื่องโฮสเนื่องมาจากโครงสร้าง 32 บิตของสถาปัตยกรรมของชิพ ในการถ่ายโอนข้อมูลกับโฮสทุกครั้งจะส่งข้อมูลที่ละ 16 บิตที่ต่อเนื่องกันสองครั้ง ใน HPID การเขียนข้อมูล $HBE[1:0]$ จะเลือกไบต์ที่ต้องการเขียน สำหรับ HPIA, HPIC และ HPID กระบวนการอ่านไบต์ที่อื่นาเบ็ดไไว้จะไม่ถูกใช้ หน้าพิเศษของขา HHWIL จะเป็นตัวชี้ว่าบิตที่หนึ่ง หรือบิตที่สองที่จะถูกส่งออกไป การควบคุมจากตัวควบคุมภายในรีจิสเตอร์บิตจะเป็นตัวกำหนดว่าบิตที่หนึ่ง หรือบิตที่สองที่จะถูกแทนที่ลงไป ในบิตที่สำคัญที่สุดของบิตนั้นๆ สำหรับการส่งแบบ 32 บิตทีเดียวในหนึ่งครั้งเครื่องโฮสจะไม่ทำการเบรกลำดับที่ส่งมาของ HPI



รูปที่ 2.42 บล็อกไดอะแกรมของ HPI

สโตรบข้อมูล 2 ตัว $\overline{HDS1}$ และ $\overline{HDS2}$ เป็นสโตรบที่ใช้การเลือกอ่านหรือการเขียน HR/\overline{W} และสโตรบที่อยู่ \overline{HAS} HPI สามารถที่ทำการเชื่อมต่อกับโฮสต์ได้อย่างง่ายโดยใช้การมัลติเพลกเซอร์หรือแบ่งหน้าที่ให้แอดเดรสบัส หรือคาต้าบัสเลย โดยใช้สโตรบข้อมูลและสโตรบที่ใช้อ่านและเขียน หรือแยกออกมาเป็น 2 สโตรบที่ใช้สำหรับอ่านและเขียน

$HCNTL[1:0]$ เป็นตัวควบคุมอินพุตที่เป็นตัวชี้ว่าจะเข้าไปติดต่อกับ HPI ตัวไหน ใช้อินพุตเหล่านี้โฮสต์จะสามารถระบุตัวที่ต้องการติดต่อกับ $HPIA$, $HPIC$ หรือ $HPID$ ได้ อินพุตเหล่านี้จะส่งมาเป็นสายพร้อมกับ $HHWIL$ โดยทั่วไปอินพุตถูกขับโดยบิตของบัสแอดเดรส หรือฟังก์ชันของอินพุตบิต เครื่องโฮสต์สามารถรบกวนการทำงานของซีพียูโดย $HPIC$ ซีพียูสามารถกระตุ้นเอาท์พุทของ \overline{HINT} เพื่อนำไปแทรกแซงการทำงานของเครื่องโฮสต์

2.25.1 คำสั่งที่ใช้ในการติดต่อกับบอร์ดมีดังนี้

<code>dsk6x_open()</code>	เริ่มการเชื่อมต่อกับบอร์ด
<code>dsk6x_close()</code>	ยุติการเชื่อมต่อกับบอร์ด
<code>dsk6x_reset_board()</code>	รีเซตค่าบนบอร์ด
<code>dsk6x_reset_dsp()</code>	รีเซตเฉพาะบอร์ด
<code>dsk6x_coff_load()</code>	โหลด COFF ลงไปที่หน่วยความจำ
<code>dsk6x_hpi_open()</code>	เปิด HPI สำหรับการติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

dsk6x_hpi_close()	ปิด HPI สำหรับการติดต่อ
dsk6x_hpi_read()	อ่านข้อมูลจากหน่วยความจำโดยใช้ HPI
dsk6x_hpi_write()	เขียนข้อมูลลงบนหน่วยความจำโดยใช้ HPI
dsk6x_generate_int()	สร้างตัวขัดจังหวะการติดต่อ

2.25.2 วิธีการส่งผ่านข้อมูล

การที่เราจะคำนวณจำนวนรอบการทำงานของ CPU ในการแลกเปลี่ยนข้อมูลระหว่างเครื่องคอมพิวเตอร์กับบอร์ด TMS320C6211 เราจะใช้ HPI

เครื่องคอมพิวเตอร์จะมีการต่อกับหน่วยความจำของตัวประมวลผลโดยผ่านทาง HPI 16 บิต จำนวนของรอบการทำงานของ CPU ในการแลกเปลี่ยนข้อมูลจะขึ้นอยู่กับวิธีการส่งผ่านข้อมูลสามารถแบ่งได้เป็น 4 วิธีคือ

1. Read With Auto-Increment
2. Read Without Auto-Increment
3. Write With Auto-Increment
4. Write Without Auto-Increment

HPI จะมี 32 บิตข้อมูล และ 16 บิตที่เป็นการเชื่อมต่อแบบอะซิงโครนัส การเชื่อมต่อจะเป็นไปอย่างอัตโนมัติโดยการส่งที่ละ 16 บิต จำนวน 2 ครั้ง ขา *HRDY*__ อนุญาตให้มีการแทรกแซงของเครื่องโฮสต์ ในขณะที่อยู่ในสถานะการรอ ดังนั้น *HRDY*__ จะส่งผลกระทบต่อความเร็วของการส่งผ่านข้อมูล เมื่อ *HRDY*__ อยู่ในสถานะ Low จะเป็นตัวแสดงว่า HPI พร้อมสำหรับการส่งผ่านข้อมูล เมื่อเปลี่ยนเป็นสถานะ High จะมีการดีเลย์ระหว่างการส่งผ่านข้อมูล *HRDY*__ จะมีลักษณะพิเศษของความเร็วในการส่งผ่านข้อมูลไปยัง HPI ข้อยกเว้นของ HPI ในการส่งผ่านข้อมูลสามารถอธิบายได้ว่าการส่งผ่านข้อมูลไปยังหรือส่งจาก HPI Address Register (*HPIA*) หรือ HPI Control Register (*HPIC*) การส่งผ่านข้อมูลของ *HPIA* และ *HPIC* จะไม่รวมสัญญาณ *HRDY*__ ข้อมูลจะถูกจัดเก็บจากรีจิสเตอร์เหล่านี้ทันทีโดยปราศจากการรวมช่อง *DMA* สำรอง

2.25.2.1 Read Without Auto-Increment

เมื่อโฮสต์ตัดสินใจว่าจะเลือกการอ่านข้อมูลวิธีนี้จาก *HPID* HPI จะส่งการร้องขอการอ่านไปยังช่อง *DMA* สำรอง และ *HRDY*__ จะมีสถานะ High สถานะการณีนี้อาจเกิดขึ้นที่ขอบข้างครั้งแรกของ *HSTROBE*__ ที่ขา *HRDY*__ ยังคงเป็นสถานะ High จนกระทั่งช่อง *DMA* สำรองโหลดข้อมูลไปยัง *HPID* ตอนเริ่มต้นของการอ่านครั้งที่ 2 การเข้าถึงข้อมูลจะถูกแสดงเรียบร้อยแล้วใน *HPID*

Read Without Auto-Increment	Duration of $HRDY_high$
Internal Data Memory	15-17
Internal Program Memory	15-17
SDRAM	25-27
SDRAM(new page accessed)	37-39
SBSRAM(1/2 x)	23-25
SBSRAM(1/ x)	21-23
Internal Peripheral Bus	13-15

ตารางที่ 2.7 ช่วงเวลาที่ $HPID$ อยู่ในสถานะ High ของการอ่านแบบ Without Auto-Increment จากตารางจะแสดงช่วงที่ $HRDY_high$ อยู่ในสถานะ High เมื่อซีพียูของเครื่องโฮสทำการอ่านแบบ Without Auto-Increment จากหน่วยความจำของบอร์ดโดยใช้ HPI

$$t_{CYC}(READ) = t_w(HSTBH - HW1) + t_w(HRDYH) + t_D(HRDYL - HSTBH) + t_w(HSTBH - HW2) + t_w(HSTBL - HW2) \quad (2.50)$$

โดยที่

$t_{CYC}(READ)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ต้องการสำหรับการอ่านแบบ Without Auto-Increment

$t_w(HSTBH - HW1)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ $HSTROBE_high$ อยู่ที่สถานะ High ก่อนที่จะมีการขอ 16 บิตแรก

$t_w(HRDYH)$ คือจำนวนรอบของสัญญาณนาฬิกาของ HPI แสดงสถานะว่ากำลังทำงาน ($HRDY_high$ อยู่ที่สถานะ High)

$t_D(HRDYL - HSTBH)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ $HRDY_high$ อยู่ที่สถานะ Low และเวลาที่ $HSTROBE_high$ อยู่ที่สถานะ High โฮสภายนอกจะเป็นตัวตัดสินใจค่านี้

$t_w(HSTBH - HW2)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ $HSTROBE_high$ อยู่ที่สถานะ High ก่อนที่จะมีการขอ 16 บิตหลัง

$t_w(HSTBL - HW2)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ $HSTROBE_low$ อยู่ที่สถานะ Low ระหว่างการส่งผ่านข้อมูลช่วงหลัง

2.25.2.2 Read With Auto-Increment

ลักษณะของ Auto-Increment จะมีผลต่อการเข้าถึงโฮส สำหรับ $HPID$ ตัวอ่าน/เขียนลักษณะนี้ จะกำจัดความต้องการของโฮสที่จะโหลด $HPIA$ สำหรับขั้นตอนการอ่านข้อมูลจะถูกชี้โดยแอดเดรส เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถัดไปคั้งนั้นหลังจาก 16 บิตหลังได้ถูกอ่านแล้ว $HRDY_$ จะอยู่ที่สถานะ High จากนั้น HPI จะ busy ก่อนที่จะได้รับข้อมูล

คั้งนั้นครั้งแรกของการส่งข้อมูล $HSTROBE_$ จะอยู่ในสถานะ High และ $HRDY_$ จะบอก ว่า HPI อยู่ในช่วงการทำงานในส่วนของภายนอกของ HPI เราจะสามารถสรุปได้ว่า $HCS_$ อยู่สถานะ Low

Read With Auto-Increment	Duration of $HRDY_ High$
Internal Data Memory	17-19
Internal Program Memory	17-19
SDRAM	27-29
SBSRAM(1/2 x)	25-27
SBSRAM(1/ x)	23-25

ตารางที่ 2.8 ช่วงเวลาที่ $HRDY_$ อยู่ที่สถานะ High ขณะที่อ่านแบบ With Auto-Increment

สมการด้านล่างใช้คำนวณเวลาที่คั้งใช้ในการอ่านแบบ With Auto-Increment

$$t_{CYC}(AC_READ) = \frac{t_w(HRDYH) + t_D(HRDYL - HSTBH) + t_w(HSTBH - HW2) + t_w(HSTBL - HW2)}{(2.51)}$$

โดยที่

$t_{CYC}(AC_READ)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ต้องการสำหรับการอ่านแบบ With Auto-Increment

$t_w(HRDYH)$ คือจำนวนรอบของสัญญาณนาฬิกาของ HPI แสดงสถานะว่ากำลังทำงาน ($HRDY_$ อยู่ที่สถานะ High)

$t_D(HRDYL - HSTBH)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ระหว่างที่ $HRDY_$ อยู่ที่สถานะ Low และเวลาที่ $HSTROBE_$ อยู่ที่สถานะ High โสสภายนอกจะเป็นตัวตัดสินคั้งนี้

$t_w(HSTBH - HW2)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ $HSTROBE_$ อยู่ที่สถานะ High ก่อนที่จะมีการขอ 16 บิตหลัง

$t_w(HSTBL - HW2)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ $HSTROBE_$ อยู่ที่สถานะ Low ระหว่างการส่งผ่านข้อมูลช่วงหลัง

2.25.2.3 Write With and Without Auto-Increment

ขณะที่ $HPID$ อยู่ในขั้นตอนเขียนทีละส่วนของ $HPID$ ถูกส่งผ่านไปไปยังเครื่องโฮส ขณะที่การเขียนกำลังจะสิ้นสุด $HRDY_$ จะอยู่ที่สถานะ High ที่ขอบขาขึ้นครั้งที่ 2 ของ $HSTROBE_$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ *HPID* ถูกส่งบิตออกไป 32 บิตไปยังแอดเดรสที่ถูกกำหนดโดย *HPIA* การเขียนแบบ Write With Auto-Increment และแบบ Write Without Auto-Increment ต้องการจำนวนรอบของสัญญาณนาฬิกาเท่ากัน

Write	Duration of <i>HRDY_High</i>
Internal Data Memory	10-12
Internal Program Memory	10-12
SDRAM	15-17
SDRAM(new page accessed)	27-29
SBSRAM(1/2 x)	14-16
SBSRAM(1/ x)	12-14
Internal Peripheral Bus	10-12

ตารางที่ 2.9 ช่วงเวลาที่ *HRDY_High* อยู่ที่สถานะ High ระหว่างการเขียน
สมการด้านล่างใช้ในการคำนวณเวลาที่ต้องการในการเขียน

$$t_{CYC}(WRITE) = t_w(HRDYH) + t_D(HRDYL - HSTBH) + t_w(HSTBH - HW2) + t_w(HSTBL - HW2) \quad (2.52)$$

โดยที่

$t_{CYC}(WRITE)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ใช้ในการเขียน

$t_w(HRDYH)$ คือจำนวนรอบของสัญญาณนาฬิกาของ HPI แสดงสถานะว่ากำลังทำงาน (*HRDY_High* อยู่ที่สถานะ High)

$t_D(HRDYL - HSTBH)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ระหว่างที่ *HRDY_High* อยู่ที่สถานะ Low และเวลาที่ *HSTROBE_High* อยู่ที่สถานะ high โสศภายนอกจะเป็นตัวตัดสินค่านี้

$t_w(HSTBH - HW2)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ *HSTROBE_High* อยู่ที่สถานะ High ก่อนที่จะมีการขอ 16 บิตหลัง

$t_w(HSTBL - HW2)$ คือจำนวนรอบของสัญญาณนาฬิกาที่ *HSTROBE_Low* ระหว่างการส่งผ่านข้อมูลช่วงหลัง

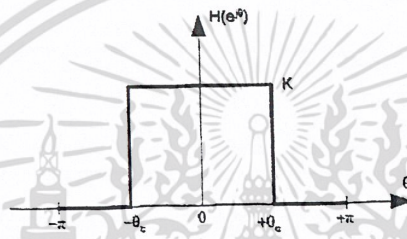
บทที่ 3

การออกแบบและการสร้าง

บทนี้อธิบายและแสดงขั้นตอนการออกแบบวงจรกรองผลตอบสนองอิมพัลส์จำกัดโดยใช้วิธีวินโดว์แบบแฮมมิง จากนั้นก็จะเป็นการอธิบายและแสดงการเขียนโปรแกรม Code Composer Studio เพื่อให้บอร์ด TMS320C6211 ทำงานตามคำสั่งต่างๆที่กำหนด ประกอบด้วยเอ็คโค, รีเวิร์บ, คอรัสและตัวกรองความถี่แบบ FIR

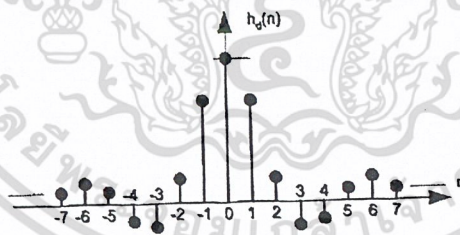
3.1 ขั้นตอนการออกแบบวงจรกรองผลตอบสนองอิมพัลส์จำกัด

3.1.1 กำหนดคุณสมบัติของวงจรกรองความถี่ที่ต้องการ (f_c , f_s และค่า Order ที่ต้องการ)



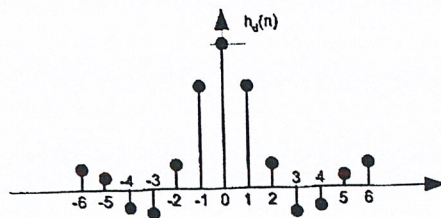
รูปที่ 3.1 คุณสมบัติของวงจรกรองความถี่

3.1.2 หากค่าสัมประสิทธิ์หรือ $h(n)$ จะทำให้การตอบสนองความถี่ตามข้อ (3.1.1) (ค่า $h(n)$ ที่ได้จะสมมาตร (Linear phase) และ Non-causal (มี n ค่าที่เป็นลบ))



รูปที่ 3.2 การตอบสนองความถี่

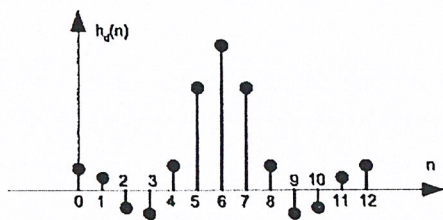
3.1.3 จำกัดจำนวนของการตอบสนองความถี่ (กำหนดความยาว N ของวงจรกรองความถี่)



รูปที่ 3.3 จำกัดจำนวนของการตอบสนองความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 เลื่อนการตอบสนองความถี่มาทางขวาทั้งหมด



รูปที่ 3.4 เลื่อนการตอบสนองความถี่มาทางขวาทั้งหมด

3.1.5 เลือกหน้าต่างที่เหมาะสมกับความถี่ ($w(n)$)



รูปที่ 3.5 การตอบสนองความถี่ของหน้าต่าง

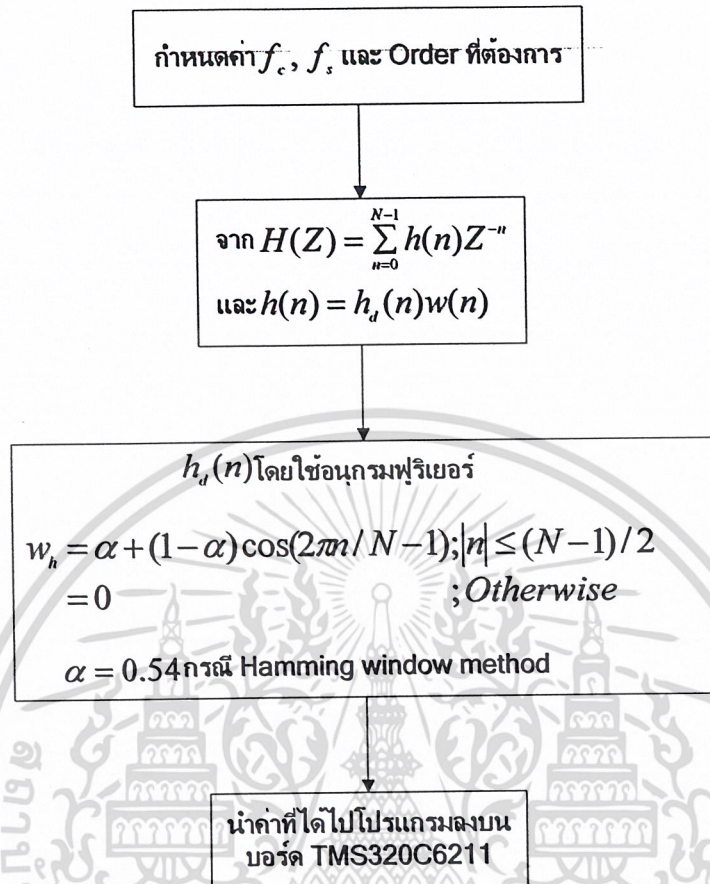
3.1.6 ปรับค่าของสัมประสิทธิ์ $h(n) = h_d(n)w(n)$



รูปที่ 3.6 การตอบสนองความถี่ของ $h(n)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 สรุปขั้นตอนการออกแบบวงจรกรองผลตอบสนองอิมพัลส์จำกัด



รูปที่ 3.7 ขั้นตอนการออกแบบวงจรกรองผลตอบสนองอิมพัลส์จำกัด

3.3 การออกแบบ FIR Lowpass filter ที่ใช้ในการทดลอง

Lowpass filter 81st – order

Directform structure

Hamming window Method

$$f_c = 450\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81 - 1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h_{LP}(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.1

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.1 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก
$$h_{LP}(n) = h_d(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่
$$\theta_c = \omega T = \frac{2\pi \times 450}{8 \times 10^3} = 0.1125\pi \text{ rad}$$

เมื่อวงจรกรองความถี่ต่ำผ่านสามารถเขียนแทนได้โดย

$$\begin{aligned} H(e^{j\theta}) &= 1 && ; |\theta| < \theta_c \\ &= 0 && ; \theta_c < |\theta| < \pi \end{aligned}$$

โดยที่
$$\alpha = \frac{N-1}{2} = 40$$

และหาค่า $h_{d_{LP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{LP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{LP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$\begin{aligned} h_{d_{LP}}(n) &= \frac{1}{2\pi} \int_{-\theta_c}^{\theta_c} H_{d_{LP}}(e^{j\theta}) e^{-jn\theta} d\theta \\ &= \frac{1}{2\pi} \int_{-\theta_c}^{\theta_c} e^{jn\theta} d\theta \\ &= \frac{e^{jn\theta}}{2\pi jn} \Big|_{-\theta_c}^{\theta_c} \\ &= \frac{1}{n\pi} \left(\frac{e^{jn\theta_c} - e^{-jn\theta_c}}{2j} \right) \\ &= \frac{\sin(n\theta_c)}{\pi(n\theta_c)} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40 \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{LP}(n) = h_{d_{LP}}(n)w_H(n)$$

จะได้ $h_{LP}(n)$ ดังตารางที่ 3.2

n	$h_{d_{LP}}(n)$	$h_{LP}(n)$	n	$h_{d_{LP}}(n)$	$h_{LP}(n)$	n	$h_{d_{LP}}(n)$	$h_{LP}(n)$
40	0.022538	0.021654	13	-0.06885	-0.01197	-14	-0.06265	-0.02965
39	0.021631	0.015371	12	-0.06687	-0.00559	-15	-0.05006	-0.04059
38	0.017935	0.006509	11	-0.05551	-0.0141	-16	-0.03325	-0.03304
37	0.011765	0.00137	10	-0.03435	-0.02019	-17	-0.01455	-0.01335
36	0.003758	0.00042	9	-0.00377	-0.00337	-18	0.003769	0.002358
35	-0.00518	-0.00182	8	0.034978	0.034922	-19	0.019719	0.005618
34	-0.01399	-0.00978	7	0.079861	0.067144	-20	0.03176	0.002856
33	-0.02155	-0.0206	6	0.1282	0.065551	-21	0.038919	0.005915
32	-0.02684	-0.02617	5	0.176876	0.035069	-22	0.04084	0.017793
31	-0.02905	-0.02166	4	0.222586	0.017895	-23	0.03777	0.029421
30	-0.0277	-0.01105	3	0.26212	0.058986	-24	0.030485	0.030051
29	-0.02272	-0.00302	2	0.292631	0.16082	-25	0.020169	0.018917
28	-0.01447	-0.00144	1	0.311889	0.270917	-26	0.008252	0.00547
27	-0.00376	-0.0012	0	0	0	-27	-0.00376	-0.0012
26	0.008252	0.00547	-1	0.311889	0.270917	-28	-0.01447	-0.00144
25	0.020169	0.018917	-2	0.292631	0.16082	-29	-0.02272	-0.00302
24	0.030485	0.030051	-3	0.26212	0.058986	-30	-0.0277	-0.01105
23	0.03777	0.029421	-4	0.222586	0.017895	-31	-0.02905	-0.02166
22	0.04084	0.017793	-5	0.176876	0.035069	-32	-0.02684	-0.02617
21	0.038919	0.005915	-6	0.1282	0.065551	-33	-0.02155	-0.0206
20	0.03176	0.002856	-7	0.079861	0.067144	-34	-0.01399	-0.00978
19	0.019719	0.005618	-8	0.034978	0.034922	-35	-0.00518	-0.00182
18	0.003769	0.002358	-9	-0.00377	-0.00337	-36	0.003758	0.00042
17	-0.01455	-0.01335	-10	-0.03435	-0.02019	-37	0.011765	0.00137
16	-0.03325	-0.03304	-11	-0.05551	-0.0141	-38	0.017935	0.006509
15	-0.05006	-0.04059	-12	-0.06687	-0.00559	-39	0.021631	0.015371
14	-0.06265	-0.02965	-13	-0.06885	-0.01197	-40	0.022538	0.021654

ตารางที่ 3.2 ค่า $h_{LP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 450\text{Hz}, f_U = 845\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81-1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.3

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.3 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 450}{8 \times 10^3} = 0.1125\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 845}{8 \times 10^3} = 0.21125\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$H(e^{j\theta}) = 1 \quad ; \theta_L \leq |\theta| \leq \theta_U$$

$$= 0 \quad ; |\theta| < \theta_L, |\theta| > \theta_U$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$h_{d_{BP}}(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta}) e^{j\theta n} d\theta$$

$$= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{j\theta n} d\theta + \int_{\theta_L}^{\theta_U} e^{j\theta n} d\theta \right\}$$

$$= \frac{1}{2\pi} \left\{ \frac{e^{j\theta n}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{j\theta n}}{nj} \Big|_{\theta_L}^{\theta_U} \right\}$$

$$= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\}$$

$$= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\}$$

$$= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.4

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00012	-0.00011	13	0.041921	0.007289	-14	0.025267	0.011958
39	-0.00215	-0.00153	12	0.050011	0.00418	-15	0.007042	0.005709
38	-0.00572	-0.00208	11	0.04418	0.011221	-16	-0.00672	-0.00668
37	-0.00894	-0.00104	10	0.023057	0.013552	-17	-0.01286	-0.0118
36	-0.00973	-0.00109	9	-0.00957	-0.00856	-18	-0.01171	-0.00732
35	-0.00673	-0.00237	8	-0.04534	-0.04527	-19	-0.00635	-0.00181
34	-4.8E-05	-3.3E-05	7	-0.0736	-0.06188	-20	-0.00096	-8.6E-05
33	0.008591	0.008211	6	-0.08475	-0.04334	-21	0.001092	0.000166
32	0.016373	0.01596	5	-0.07357	-0.01459	-22	-0.00143	-0.00062
31	0.020432	0.015231	4	-0.04128	-0.00332	-23	-0.0073	-0.00569
30	0.018995	0.007576	3	0.004373	0.000984	-24	-0.01356	-0.01337
29	0.012169	0.001616	2	0.051141	0.028106	-25	-0.0169	-0.01585
28	0.002008	0.0002	1	0.085921	0.074634	-26	-0.01516	-0.01005
27	-0.00819	-0.0026	0	0	0	-27	-0.00819	-0.0026
26	-0.01516	-0.01005	-1	0.085921	0.074634	-28	0.002008	0.0002
25	-0.0169	-0.01585	-2	0.051141	0.028106	-29	0.012169	0.001616
24	-0.01356	-0.01337	-3	0.004373	0.000984	-30	0.018995	0.007576
23	-0.0073	-0.00569	-4	-0.04128	-0.00332	-31	0.020432	0.015231
22	-0.00143	-0.00062	-5	-0.07357	-0.01459	-32	0.016373	0.01596
21	0.001092	0.000166	-6	-0.08475	-0.04334	-33	0.008591	0.008211
20	-0.00096	-8.6E-05	-7	-0.0736	-0.06188	-34	-4.8E-05	-3.3E-05
19	-0.00635	-0.00181	-8	-0.04534	-0.04527	-35	-0.00673	-0.00237
18	-0.01171	-0.00732	-9	-0.00957	-0.00856	-36	-0.00973	-0.00109
17	-0.01286	-0.0118	-10	0.023057	0.013552	-37	-0.00894	-0.00104
16	-0.00672	-0.00668	-11	0.04418	0.011221	-38	-0.00572	-0.00208
15	0.007042	0.005709	-12	0.050011	0.00418	-39	-0.00215	-0.00153
14	0.025267	0.011958	-13	0.041921	0.007289	-40	-0.00012	-0.00011

ตารางที่ 3.4 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 845\text{Hz}, f_U = 1240\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81-1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54)\cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46\cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.5

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.5 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 845}{8 \times 10^3} = 0.21125\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 1240}{8 \times 10^3} = 0.31\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$H(e^{j\theta}) = 1 \quad ; \theta_L \leq |\theta| \leq \theta_U$$

$$= 0 \quad ; |\theta| < \theta_L, |\theta| > \theta_U$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$h_{d_{BP}}(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta})e^{jn\theta} d\theta$$

$$= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{jn\theta} d\theta + \int_{\theta_L}^{\theta_U} e^{jn\theta} d\theta \right\}$$

$$= \frac{1}{2\pi} \left\{ \frac{e^{jn\theta}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{jn\theta}}{nj} \Big|_{\theta_L}^{\theta_U} \right\}$$

$$= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\}$$

$$= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\}$$

$$= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก
$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.6

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00032	-0.00031	13	-0.01545	-0.00269	-14	0.016724	0.007915
39	-0.00336	-0.00239	12	-0.04694	-0.00392	-15	0.029631	0.024025
38	-0.00608	-0.00221	11	-0.05233	-0.01329	-16	0.021118	0.020983
37	-0.00377	-0.00044	10	-0.02061	-0.01212	-17	0.004051	0.003717
36	0.00428	0.000479	9	0.032717	0.029257	-18	-0.0068	-0.00426
35	0.012832	0.004514	8	0.072705	0.07259	-19	-0.00644	-0.00184
34	0.014302	0.010001	7	0.068411	0.057517	-20	-0.00103	-9.2E-05
33	0.005286	0.005052	6	0.016724	0.008552	-21	0.000321	4.89E-05
32	-0.00952	-0.00928	5	-0.05184	-0.01028	-22	-0.00511	-0.00223
31	-0.01989	-0.01483	4	-0.09174	-0.00738	-23	-0.0114	-0.00888
30	-0.0177	-0.00706	3	-0.07366	-0.01658	-24	-0.01021	-0.01007
29	-0.00364	-0.00048	2	-0.0064	-0.00352	-25	0.000661	0.00062
28	0.012791	0.001271	1	0.067217	0.058387	-26	0.014383	0.009534
27	0.020286	0.006442	0	0	0	-27	0.020286	0.006442
26	0.014383	0.009534	-1	0.067217	0.058387	-28	0.012791	0.001271
25	0.000661	0.00062	-2	-0.0064	-0.00352	-29	-0.00364	-0.00048
24	-0.01021	-0.01007	-3	-0.07366	-0.01658	-30	-0.0177	-0.00706
23	-0.0114	-0.00888	-4	-0.09174	-0.00738	-31	-0.01989	-0.01483
22	-0.00511	-0.00223	-5	-0.05184	-0.01028	-32	-0.00952	-0.00928
21	0.000321	4.89E-05	-6	0.016724	0.008552	-33	0.005286	0.005052
20	-0.00103	-9.2E-05	-7	0.068411	0.057517	-34	0.014302	0.010001
19	-0.00644	-0.00184	-8	0.072705	0.07259	-35	0.012832	0.004514
18	-0.0068	-0.00426	-9	0.032717	0.029257	-36	0.00428	0.000479
17	0.004051	0.003717	-10	-0.02061	-0.01212	-37	-0.00377	-0.00044
16	0.021118	0.020983	-11	-0.05233	-0.01329	-38	-0.00608	-0.00221
15	0.029631	0.024025	-12	-0.04694	-0.00392	-39	-0.00336	-0.00239
14	0.016724	0.007915	-13	-0.01545	-0.00269	-40	-0.00032	-0.00031

ตารางที่ 3.6 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 1240\text{Hz}, f_U = 1635\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81-1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.7

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.7 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 1240}{8 \times 10^3} = 0.31\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 1635}{8 \times 10^3} = 0.40875\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$\begin{aligned} H(e^{j\theta}) &= 1 & ; \theta_L \leq |\theta| \leq \theta_U \\ &= 0 & ; |\theta| < \theta_L, |\theta| > \theta_U \end{aligned}$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$\begin{aligned} h_{d_{BP}}(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta}) e^{jn\theta} d\theta \\ &= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{jn\theta} d\theta + \int_{\theta_L}^{\theta_U} e^{jn\theta} d\theta \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{e^{jn\theta}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{jn\theta}}{nj} \Big|_{\theta_L}^{\theta_U} \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\} \\ &= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\} \\ &= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40 \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก
$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.8

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00052	-0.0005	13	-0.02246	-0.0039	-14	-0.0374	-0.0177
39	-0.00383	-0.00272	12	0.02855	0.002386	-15	-0.01067	-0.00865
38	-0.00289	-0.00105	11	0.056705	0.014402	-16	0.017122	0.017013
37	0.005483	0.000638	10	0.018137	0.01066	-17	0.017155	0.015738
36	0.011155	0.001248	9	-0.05187	-0.04639	-18	0.001317	0.000824
35	0.003074	0.001082	8	-0.06944	-0.06933	-19	-0.00556	-0.00158
34	-0.01247	-0.00872	7	-0.00363	-0.00305	-20	-0.00109	-9.8E-05
33	-0.0159	-0.01519	6	0.075199	0.038451	-21	-0.00047	-7.1E-05
32	0.000353	0.000344	5	0.071456	0.014167	-22	-0.00734	-0.0032
31	0.018645	0.013899	4	-0.01826	-0.00147	-23	-0.00777	-0.00605
30	0.016148	0.00644	3	-0.09241	-0.0208	-24	0.005375	0.005298
29	-0.00556	-0.00074	2	-0.06156	-0.03383	-25	0.017038	0.015981
28	-0.02085	-0.00207	1	0.042103	0.036572	-26	0.009218	0.00611
27	-0.0119	-0.00378	0	0	0	-27	-0.0119	-0.00378
26	0.009218	0.00611	-1	0.042103	0.036572	-28	-0.02085	-0.00207
25	0.017038	0.015981	-2	-0.06156	-0.03383	-29	-0.00556	-0.00074
24	0.005375	0.005298	-3	-0.09241	-0.0208	-30	0.016148	0.00644
23	-0.00777	-0.00605	-4	-0.01826	-0.00147	-31	0.018645	0.013899
22	-0.00734	-0.0032	-5	0.071456	0.014167	-32	0.000353	0.000344
21	-0.00047	-7.1E-05	-6	0.075199	0.038451	-33	-0.0159	-0.01519
20	-0.00109	-9.8E-05	-7	-0.00363	-0.00305	-34	-0.01247	-0.00872
19	-0.00556	-0.00158	-8	-0.06944	-0.06933	-35	0.003074	0.001082
18	0.001317	0.000824	-9	-0.05187	-0.04639	-36	0.011155	0.001248
17	0.017155	0.015738	-10	0.018137	0.01066	-37	0.005483	0.000638
16	0.017122	0.017013	-11	0.056705	0.014402	-38	-0.00289	-0.00105
15	-0.01067	-0.00865	-12	0.02855	0.002386	-39	-0.00383	-0.00272
14	-0.0374	-0.0177	-13	-0.02246	-0.0039	-40	-0.00052	-0.0005

ตารางที่ 3.8 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 1635\text{Hz}, f_U = 2030\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81-1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N-1)) \quad ; |n| \leq (N-1)/2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80/2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.9

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.9 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 1635}{8 \times 10^3} = 0.40875\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 2030}{8 \times 10^3} = 0.5075\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$\begin{aligned} H(e^{j\theta}) &= 1 && ; \theta_L \leq |\theta| \leq \theta_U \\ &= 0 && ; |\theta| < \theta_L, |\theta| > \theta_U \end{aligned}$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$\begin{aligned} h_{d_{BP}}(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta}) e^{jn\theta} d\theta \\ &= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{jn\theta} d\theta + \int_{\theta_L}^{\theta_U} e^{jn\theta} d\theta \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{e^{jn\theta}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{jn\theta}}{nj} \Big|_{\theta_L}^{\theta_U} \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\} \\ &= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\} \\ &= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40 \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.10

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00071	-0.00068	13	0.043741	0.007605	-14	0.010421	0.004932
39	-0.00345	-0.00245	12	-0.00084	-7.1E-05	-15	-0.02832	-0.02297
38	0.001979	0.000718	11	-0.057	-0.01448	-16	-0.01269	-0.0126
37	0.008803	0.001025	10	-0.01563	-0.00919	-17	0.014142	0.012974
36	-0.00057	-6.4E-05	9	0.064708	0.057864	-18	0.008813	0.005513
35	-0.01371	-0.00482	8	0.03693	0.036871	-19	-0.00383	-0.00109
34	-0.00338	-0.00237	7	-0.06432	-0.05407	-20	-0.00114	-0.0001
33	0.016691	0.015952	6	-0.05968	-0.03052	-21	-0.00123	-0.00019
32	0.008897	0.008672	5	0.054761	0.010857	-22	-0.0075	-0.00327
31	-0.01675	-0.01249	4	0.079885	0.006423	-23	0.001127	0.000878
30	-0.01435	-0.00572	3	-0.03681	-0.00828	-24	0.014521	0.014314
29	0.013731	0.001824	2	-0.09379	-0.05155	-25	0.002813	0.002638
28	0.017933	0.001782	1	0.012973	0.011269	-26	-0.01819	-0.01206
27	-0.0085	-0.0027	0	0	0	-27	-0.0085	-0.0027
26	-0.01819	-0.01206	-1	0.012973	0.011269	-28	0.017933	0.001782
25	0.002813	0.002638	-2	-0.09379	-0.05155	-29	0.013731	0.001824
24	0.014521	0.014314	-3	-0.03681	-0.00828	-30	-0.01435	-0.00572
23	0.001127	0.000878	-4	0.079885	0.006423	-31	-0.01675	-0.01249
22	-0.0075	-0.00327	-5	0.054761	0.010857	-32	0.008897	0.008672
21	-0.00123	-0.00019	-6	-0.05968	-0.03052	-33	0.016691	0.015952
20	-0.00114	-0.0001	-7	-0.06432	-0.05407	-34	-0.00338	-0.00237
19	-0.00383	-0.00109	-8	0.03693	0.036871	-35	-0.01371	-0.00482
18	0.008813	0.005513	-9	0.064708	0.057864	-36	-0.00057	-6.4E-05
17	0.014142	0.012974	-10	-0.01563	-0.00919	-37	0.008803	0.001025
16	-0.01269	-0.0126	-11	-0.057	-0.01448	-38	0.001979	0.000718
15	-0.02832	-0.02297	-12	-0.00084	-7.1E-05	-39	-0.00345	-0.00245
14	0.010421	0.004932	-13	0.043741	0.007605	-40	-0.00071	-0.00068

ตารางที่ 3.10 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 2030\text{Hz}, f_U = 2425\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81 - 1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54)\cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46\cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.11

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.11 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 2030}{8 \times 10^3} = 0.5075\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 2425}{8 \times 10^3} = 0.60625\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$\begin{aligned} H(e^{j\theta}) &= 1 & ; \theta_L \leq |\theta| \leq \theta_U \\ &= 0 & ; |\theta| < \theta_L, |\theta| > \theta_U \end{aligned}$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$\begin{aligned} h_{d_{BP}}(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta}) e^{j\theta n} d\theta \\ &= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{j\theta n} d\theta + \int_{\theta_L}^{\theta_U} e^{j\theta n} d\theta \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{e^{j\theta n}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{j\theta n}}{nj} \Big|_{\theta_L}^{\theta_U} \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\} \\ &= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\} \\ &= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40 \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.12

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00088	-0.00084	13	-0.03265	-0.00568	-14	0.029841	0.014123
39	-0.00232	-0.00165	12	-0.02714	-0.00227	-15	0.014139	0.011463
38	0.005698	0.002068	11	0.053182	0.013507	-16	-0.02337	-0.02322
37	0.002603	0.000303	10	0.013095	0.007697	-17	-0.00216	-0.00198
36	-0.01134	-0.00127	9	-0.06965	-0.06229	-18	0.012143	0.007597
35	0.000832	0.000293	8	0.01114	0.011122	-19	-0.00153	-0.00044
34	0.015434	0.010793	7	0.07623	0.064091	-20	-0.00119	-0.00011
33	-0.00717	-0.00686	6	-0.04111	-0.02102	-21	-0.00193	-0.00029
32	-0.01599	-0.01559	5	-0.06922	-0.01372	-22	-0.00553	-0.00241
31	0.014265	0.010634	4	0.070109	0.005637	-23	0.009257	0.007211
30	0.012336	0.00492	3	0.048418	0.010896	-24	0.006259	0.00617
29	-0.01938	-0.00257	2	-0.0911	-0.05006	-25	-0.01646	-0.01544
28	-0.00557	-0.00055	1	-0.01739	-0.01511	-26	-0.00171	-0.00113
27	0.020318	0.006453	0	0	0	-27	0.020318	0.006453
26	-0.00171	-0.00113	-1	-0.01739	-0.01511	-28	-0.00557	-0.00055
25	-0.01646	-0.01544	-2	-0.0911	-0.05006	-29	-0.01938	-0.00257
24	0.006259	0.00617	-3	0.048418	0.010896	-30	0.012336	0.00492
23	0.009257	0.007211	-4	0.070109	0.005637	-31	0.014265	0.010634
22	-0.00553	-0.00241	-5	-0.06922	-0.01372	-32	-0.01599	-0.01559
21	-0.00193	-0.00029	-6	-0.04111	-0.02102	-33	-0.00717	-0.00686
20	-0.00119	-0.00011	-7	0.07623	0.064091	-34	0.015434	0.010793
19	-0.00153	-0.00044	-8	0.01114	0.011122	-35	0.000832	0.000293
18	0.012143	0.007597	-9	-0.06965	-0.06229	-36	-0.01134	-0.00127
17	-0.00216	-0.00198	-10	0.013095	0.007697	-37	0.002603	0.000303
16	-0.02337	-0.02322	-11	0.053182	0.013507	-38	0.005698	0.002068
15	0.014139	0.011463	-12	-0.02714	-0.00227	-39	-0.00232	-0.00165
14	0.029841	0.014123	-13	-0.03265	-0.00568	-40	-0.00088	-0.00084

ตารางที่ 3.12 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 2425\text{Hz}, f_U = 2820\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81 - 1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.13

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.13 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 2425}{8 \times 10^3} = 0.60625\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 2820}{8 \times 10^3} = 0.705\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$H(e^{j\theta}) = 1 \quad ; \theta_L \leq |\theta| \leq \theta_U$$

$$= 0 \quad ; |\theta| < \theta_L, |\theta| > \theta_U$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$h_{d_{BP}}(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta}) e^{jn\theta} d\theta$$

$$= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{jn\theta} d\theta + \int_{\theta_L}^{\theta_U} e^{jn\theta} d\theta \right\}$$

$$= \frac{1}{2\pi} \left\{ \frac{e^{jn\theta}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{jn\theta}}{nj} \Big|_{\theta_L}^{\theta_U} \right\}$$

$$= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\}$$

$$= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\}$$

$$= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.14

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00102	-0.00098	13	-0.00261	-0.00045	-14	-0.03208	-0.01518
39	-0.00068	-0.00048	12	0.046262	0.003866	-15	0.026594	0.021562
38	0.006094	0.002212	11	-0.04554	-0.01156	-16	0.001176	0.001168
37	-0.00641	-0.00075	10	-0.01054	-0.00619	-17	-0.01643	-0.01507
36	-0.00321	-0.00036	9	0.066112	0.05912	-18	0.009734	0.006089
35	0.013471	0.004739	8	-0.05452	-0.05443	-19	0.001009	0.000288
34	-0.01013	-0.00708	7	-0.02173	-0.01827	-20	-0.00123	-0.00011
33	-0.00677	-0.00647	6	0.083163	0.042523	-21	-0.00253	-0.00038
32	0.019202	0.018718	5	-0.05759	-0.01142	-22	-0.002	-0.00087
31	-0.01128	-0.00841	4	-0.03438	-0.00276	-23	0.011112	0.008656
30	-0.01014	-0.00404	3	0.094681	0.021307	-24	-0.00951	-0.00937
29	0.02145	0.002849	2	-0.05447	-0.02994	-25	-0.00617	-0.00579
28	-0.00973	-0.00097	1	-0.0461	-0.04005	-26	0.018894	0.012524
27	-0.01162	-0.00369	0	0	0	-27	-0.01162	-0.00369
26	0.018894	0.012524	-1	-0.0461	-0.04005	-28	-0.00973	-0.00097
25	-0.00617	-0.00579	-2	-0.05447	-0.02994	-29	0.02145	0.002849
24	-0.00951	-0.00937	-3	0.094681	0.021307	-30	-0.01014	-0.00404
23	0.011112	0.008656	-4	-0.03438	-0.00276	-31	-0.01128	-0.00841
22	-0.002	-0.00087	-5	-0.05759	-0.01142	-32	0.019202	0.018718
21	-0.00253	-0.00038	-6	0.083163	0.042523	-33	-0.00677	-0.00647
20	-0.00123	-0.00011	-7	-0.02173	-0.01827	-34	-0.01013	-0.00708
19	0.001009	0.000288	-8	-0.05452	-0.05443	-35	0.013471	0.004739
18	0.009734	0.006089	-9	0.066112	0.05912	-36	-0.00321	-0.00036
17	-0.01643	-0.01507	-10	-0.01054	-0.00619	-37	-0.00641	-0.00075
16	0.001176	0.001168	-11	-0.04554	-0.01156	-38	0.006094	0.002212
15	0.026594	0.021562	-12	0.046262	0.003866	-39	-0.00068	-0.00048
14	-0.03208	-0.01518	-13	-0.00261	-0.00045	-40	-0.00102	-0.00098

ตารางที่ 3.14 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 2820\text{Hz}, f_U = 3215\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81 - 1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.15

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.15 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 2820}{8 \times 10^3} = 0.705\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 3215}{8 \times 10^3} = 0.80375\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$\begin{aligned} H(e^{j\theta}) &= 1 & ; \theta_L \leq |\theta| \leq \theta_U \\ &= 0 & ; |\theta| < \theta_L, |\theta| > \theta_U \end{aligned}$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$\begin{aligned} h_{d_{BP}}(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta}) e^{jn\theta} d\theta \\ &= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{jn\theta} d\theta + \int_{\theta_L}^{\theta_U} e^{jn\theta} d\theta \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{e^{jn\theta}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{jn\theta}}{nj} \Big|_{\theta_L}^{\theta_U} \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\} \\ &= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\} \\ &= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40 \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.16

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00114	-0.00109	13	0.035938	0.006248	-14	-0.00656	-0.00311
39	0.001118	0.000794	12	-0.05029	-0.0042	-15	-0.01739	-0.0141
38	0.002936	0.001066	11	0.034608	0.00879	-16	0.023948	0.023796
37	-0.00849	-0.00099	10	0.007967	0.004682	-17	-0.01527	-0.01401
36	0.010278	0.00115	9	-0.05451	-0.04875	-18	0.002723	0.001704
35	-0.00467	-0.00164	8	0.07493	0.074811	-19	0.003394	0.000967
34	-0.00657	-0.00459	7	-0.0517	-0.04346	-20	-0.00126	-0.00011
33	0.016538	0.015806	6	-0.0064	-0.00327	-21	-0.003	-0.00046
32	-0.01776	-0.01731	5	0.066868	0.013258	-22	0.002098	0.000914
31	0.007901	0.00589	4	-0.09242	-0.00743	-23	0.005435	0.004233
30	0.007787	0.003106	3	0.064755	0.014572	-24	-0.01388	-0.01368
29	-0.01957	-0.0026	2	0.002437	0.00134	-25	0.015207	0.014264
28	0.019904	0.001978	1	-0.07041	-0.06116	-26	-0.0061	-0.00404
27	-0.00881	-0.0028	0	0	0	-27	-0.00881	-0.0028
26	-0.0061	-0.00404	-1	-0.07041	-0.06116	-28	0.019904	0.001978
25	0.015207	0.014264	-2	0.002437	0.00134	-29	-0.01957	-0.0026
24	-0.01388	-0.01368	-3	0.064755	0.014572	-30	0.007787	0.003106
23	0.005435	0.004233	-4	-0.09242	-0.00743	-31	0.007901	0.00589
22	0.002098	0.000914	-5	0.066868	0.013258	-32	-0.01776	-0.01731
21	-0.003	-0.00046	-6	-0.0064	-0.00327	-33	0.016538	0.015806
20	-0.00126	-0.00011	-7	-0.0517	-0.04346	-34	-0.00657	-0.00459
19	0.003394	0.000967	-8	0.07493	0.074811	-35	-0.00467	-0.00164
18	0.002723	0.001704	-9	-0.05451	-0.04875	-36	0.010278	0.00115
17	-0.01527	-0.01401	-10	0.007967	0.004682	-37	-0.00849	-0.00099
16	0.023948	0.023796	-11	0.034608	0.00879	-38	0.002936	0.001066
15	-0.01739	-0.0141	-12	-0.05029	-0.0042	-39	0.001118	0.000794
14	-0.00656	-0.00311	-13	0.035938	0.006248	-40	-0.00114	-0.00109

ตารางที่ 3.16 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11 การออกแบบ FIR Bandpass filter ที่ใช้ในการทดลอง

Bandpass filter 81st – order

Directform structure

Hamming window Method

$$f_L = 3215\text{Hz}, f_U = 3610\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81 - 1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.17

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.17 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_L = \omega T = \frac{2\pi \times 3215}{8 \times 10^3} = 0.80375\pi \text{ rad}$$

$$\theta_U = \omega T = \frac{2\pi \times 3610}{8 \times 10^3} = 0.9025\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$\begin{aligned} H(e^{j\theta}) &= 1 & ; \theta_L \leq |\theta| \leq \theta_U \\ &= 0 & ; |\theta| < \theta_L, |\theta| > \theta_U \end{aligned}$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{BP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{BP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{BP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$\begin{aligned} h_{d_{BP}}(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{BP}}(e^{j\theta}) e^{j\theta n} d\theta \\ &= \frac{1}{2\pi} \left\{ \int_{-\theta_U}^{-\theta_L} e^{j\theta n} d\theta + \int_{\theta_L}^{\theta_U} e^{j\theta n} d\theta \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{e^{j\theta n}}{nj} \Big|_{-\theta_U}^{-\theta_L} + \frac{e^{j\theta n}}{nj} \Big|_{\theta_L}^{\theta_U} \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-j\theta_L n} - e^{-j\theta_U n}) + \frac{1}{nj} (e^{j\theta_U n} - e^{j\theta_L n}) \right\} \\ &= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{j\theta_U n} - e^{-j\theta_U n}}{2j} \right) - \frac{1}{n} \left(\frac{e^{j\theta_L n} - e^{-j\theta_L n}}{2j} \right) \right\} \\ &= \frac{1}{n\pi} \{ \sin(\theta_U n) - \sin(\theta_L n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40 \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{BP}(n) = h_{d_{BP}}(n)w_H(n)$$

จะได้ $h_{BP}(n)$ ดังตารางที่ 3.18

n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$	n	$h_{d_{BP}}(n)$	$h_{BP}(n)$
40	-0.00123	-0.00118	13	-0.04266	-0.00742	-14	0.036839	0.017435
39	0.002665	0.001894	12	0.037911	0.003168	-15	-0.02446	-0.01983
38	-0.00193	-0.0007	11	-0.02119	-0.00538	-16	0.010619	0.010552
37	-0.00139	-0.00016	10	-0.00538	-0.00316	-17	0.000241	0.000221
36	0.006625	0.000741	9	0.036267	0.032431	-18	-0.00557	-0.00349
35	-0.01214	-0.00427	8	-0.06378	-0.06368	-19	0.005265	0.0015
34	0.015878	0.011103	7	0.08009	0.067336	-20	-0.00128	-0.00012
33	-0.01609	-0.01537	6	-0.07951	-0.04065	-21	-0.00331	-0.0005
32	0.012007	0.011704	5	0.060319	0.011959	-22	0.005601	0.00244
31	-0.00424	-0.00316	4	-0.0256	-0.00206	-23	-0.00393	-0.00306
30	-0.00532	-0.00212	3	-0.01728	-0.00389	-24	-0.00161	-0.00159
29	0.014089	0.001871	2	0.05844	0.032117	-25	0.009269	0.008694
28	-0.01959	-0.00195	1	-0.08801	-0.07645	-26	-0.01638	-0.01086
27	0.020345	0.006461	0	0	0	-27	0.020345	0.006461
26	-0.01638	-0.01086	-1	-0.08801	-0.07645	-28	-0.01959	-0.00195
25	0.009269	0.008694	-2	0.05844	0.032117	-29	0.014089	0.001871
24	-0.00161	-0.00159	-3	-0.01728	-0.00389	-30	-0.00532	-0.00212
23	-0.00393	-0.00306	-4	-0.0256	-0.00206	-31	-0.00424	-0.00316
22	0.005601	0.00244	-5	0.060319	0.011959	-32	0.012007	0.011704
21	-0.00331	-0.0005	-6	-0.07951	-0.04065	-33	-0.01609	-0.01537
20	-0.00128	-0.00012	-7	0.08009	0.067336	-34	0.015878	0.011103
19	0.005265	0.0015	-8	-0.06378	-0.06368	-35	-0.01214	-0.00427
18	-0.00557	-0.00349	-9	0.036267	0.032431	-36	0.006625	0.000741
17	0.000241	0.000221	-10	-0.00538	-0.00316	-37	-0.00139	-0.00016
16	0.010619	0.010552	-11	-0.02119	-0.00538	-38	-0.00193	-0.0007
15	-0.02446	-0.01983	-12	0.037911	0.003168	-39	0.002665	0.001894
14	0.036839	0.017435	-13	-0.04266	-0.00742	-40	-0.00123	-0.00118

ตารางที่ 3.18 ค่า $h_{BP}(n)$ ที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.12 การออกแบบ FIR Highpass filter ที่ใช้ในการทดลอง

Highpass filter 81st – order

Directform structure

Hamming window Method

$$f_c = 3610\text{Hz}, \text{ sampling frequency} = 8\text{kHz}$$

จาก Transfer function ของ FIR filter

$$H(Z) = \sum_{n=0}^{N-1} h(n)Z^{-n}$$

$$\frac{Y(Z)}{X(Z)} = h_0 + h_1Z^{-1} + h_2Z^{-2} + \dots + h_{(N-1)}Z^{-(N-1)}$$

จะได้

$$Y(Z) = h_0X(Z) + h_1Z^{-1}X(Z) + h_2Z^{-2}X(Z) + \dots + h_{(N-1)}Z^{-(N-1)}X(Z)$$

ทำการหาค่า $h(n)$ โดยที่ $n = 0, 1, 2, \dots, (81 - 1 = 80)$

ในที่นี้เราจะใช้การ Hamming window Method ซึ่งจะได้

$$h(n) = h_d(n)w_H(n)$$

โดยที่

$$w_H(n) = 0.54 + (1 - 0.54) \cos(2\pi n / (N - 1)) \quad ; |n| \leq (N - 1) / 2$$

$$= 0 \quad ; \text{otherwise}$$

$$w_H(n) = 0.54 + 0.46 \cos(2\pi n / 80) \quad ; |n| \leq (80 / 2)$$

$$= 0 \quad ; \text{otherwise}$$

จะได้ค่า $w_H(n)$ ดังตารางที่ 3.19

n	$w_H(n)$	n	$w_H(n)$	n	$w_H(n)$
40	0.960781	13	0.173863	-14	0.473272
39	0.710568	12	0.083576	-15	0.810793
38	0.362933	11	0.253979	-16	0.993648
37	0.116431	10	0.587745	-17	0.917399
36	0.111854	9	0.894241	-18	0.625595
35	0.351816	8	0.998409	-19	0.284903
34	0.699261	7	0.840753	-20	0.089911
33	0.955742	6	0.51132	-21	0.151992
32	0.974769	5	0.198267	-22	0.435686
31	0.745474	4	0.080398	-23	0.77896
30	0.398821	3	0.225034	-24	0.98575
29	0.132803	2	0.549566	-25	0.937947
28	0.099359	1	0.868634	-26	0.662853
27	0.31759	0	1	-27	0.31759
26	0.662853	-1	0.868634	-28	0.099359
25	0.937947	-2	0.549566	-29	0.132803
24	0.98575	-3	0.225034	-30	0.398821
23	0.77896	-4	0.080398	-31	0.745474
22	0.435686	-5	0.198267	-32	0.974769
21	0.151992	-6	0.51132	-33	0.955742
20	0.089911	-7	0.840753	-34	0.699261
19	0.284903	-8	0.998409	-35	0.351816
18	0.625595	-9	0.894241	-36	0.111854
17	0.917399	-10	0.587745	-37	0.116431
16	0.993648	-11	0.253979	-38	0.362933
15	0.810793	-12	0.083576	-39	0.710568
14	0.473272	-13	0.173863	-40	0.960781

ตารางที่ 3.19 ค่า $w_H(n)$ ที่ได้จาก Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก

$$h_{HP}(n) = h_{d_{HP}}(n)w_H(n) \quad ; n = 0, \pm 1, \pm 2, \dots, \pm \left(\frac{N-1}{2}\right)$$

โดยที่

$$\theta_c = \omega T = \frac{2\pi \times 3610}{8 \times 10^3} = 0.45125\pi \text{ rad}$$

เมื่อวงจรกรองช่วงความถี่ผ่านสามารถเขียนแทนได้โดย

$$\begin{aligned} H(e^{j\theta}) &= 1 & ; \theta_L \leq |\theta| \leq \theta_U \\ &= 0 & ; |\theta| < \theta_L, |\theta| > \theta_U \end{aligned}$$

โดยที่ $\alpha = \frac{N-1}{2} = 40$

และหาค่า $h_{d_{HP}}(n)$ โดย Fourier Series ได้ดังต่อไปนี้

$$H_{d_{HP}}(\theta_c) = h_0 + \sum_{n=-40}^{40} h_{d_{HP}}(n)e^{jn\theta_c}$$

$$h_0 = \theta_c$$

ทำการแปลงกลับ Fourier จะได้

$$\begin{aligned} h_{d_{HP}}(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{d_{HP}}(e^{j\theta}) e^{jn\theta} d\theta \\ &= \frac{1}{2\pi} \left\{ \int_{-\pi}^{-\theta_c} e^{jn\theta} d\theta + \int_{\theta_c}^{\pi} e^{jn\theta} d\theta \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{e^{jn\theta}}{nj} \Big|_{-\pi}^{-\theta_c} + \frac{e^{jn\theta}}{nj} \Big|_{\theta_c}^{\pi} \right\} \\ &= \frac{1}{2\pi} \left\{ \frac{1}{nj} (e^{-jn\theta_c} - e^{-jn\pi}) + \frac{1}{nj} (e^{jn\pi} - e^{jn\theta_c}) \right\} \\ &= \frac{1}{\pi} \left\{ \frac{1}{n} \left(\frac{e^{jn\pi} - e^{-jn\pi}}{2j} \right) - \frac{1}{n} \left(\frac{e^{jn\theta_c} - e^{-jn\theta_c}}{2j} \right) \right\} \\ &= \frac{1}{n\pi} \{ \sin(\pi n) - \sin(\theta_c n) \} \\ &= \frac{1}{n\pi} \{ -\sin(\theta_c n) \} \quad \text{เมื่อ } n = 0, \pm 1, \pm 2, \dots, \pm 40 \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก

$$h_{HP}(n) = h_{d_{HP}}(n)w_H(n)$$

จะได้ $h_{HP}(n)$ ดังตารางที่ 3.20

n	$h_{d_{HP}}(n)$	$h_{HP}(n)$	n	$h_{d_{HP}}(n)$	$h_{HP}(n)$	n	$h_{d_{HP}}(n)$	$h_{HP}(n)$
40	0.005709	0.005485	13	0.009921	0.001725	-14	-0.00078	-0.00037
39	0.007751	0.005508	12	0.020254	0.001693	-15	-0.00991	-0.00803
38	0.008309	0.003016	11	0.028042	0.007122	-16	-0.01604	-0.01593
37	0.007188	0.000837	10	0.031213	0.018345	-17	-0.01842	-0.0169
36	0.004526	0.000506	9	0.028146	0.025169	-18	-0.01706	-0.01067
35	0.00078	0.000274	8	0.017972	0.017944	-19	-0.0126	-0.00359
34	-0.00335	-0.00234	7	0.000781	0.000656	-20	-0.0062	-0.00056
33	-0.00703	-0.00672	6	-0.02232	-0.01141	-21	0.00078	0.000119
32	-0.0095	-0.00926	5	-0.04931	-0.00978	-22	0.006975	0.003039
31	-0.01016	-0.00757	4	-0.07744	-0.00623	-23	0.011295	0.008798
30	-0.00876	-0.0035	3	-0.10367	-0.02333	-24	0.013088	0.012902
29	-0.00546	-0.00072	2	-0.12498	-0.06869	-25	0.012223	0.011464
28	-0.00078	-7.7E-05	1	-0.13888	-0.12064	-26	0.009071	0.006013
27	0.004404	0.001399	0	0	0	-27	0.004404	0.001399
26	0.009071	0.006013	-1	-0.13888	-0.12064	-28	-0.00078	-7.7E-05
25	0.012223	0.011464	-2	-0.12498	-0.06869	-29	-0.00546	-0.00072
24	0.013088	0.012902	-3	-0.10367	-0.02333	-30	-0.00876	-0.0035
23	0.011295	0.008798	-4	-0.07744	-0.00623	-31	-0.01016	-0.00757
22	0.006975	0.003039	-5	-0.04931	-0.00978	-32	-0.0095	-0.00926
21	0.00078	0.000119	-6	-0.02232	-0.01141	-33	-0.00703	-0.00672
20	-0.0062	-0.00056	-7	0.000781	0.000656	-34	-0.00335	-0.00234
19	-0.0126	-0.00359	-8	0.017972	0.017944	-35	0.00078	0.000274
18	-0.01706	-0.01067	-9	0.028146	0.025169	-36	0.004526	0.000506
17	-0.01842	-0.0169	-10	0.031213	0.018345	-37	0.007188	0.000837
16	-0.01604	-0.01593	-11	0.028042	0.007122	-38	0.008309	0.003016
15	-0.00991	-0.00803	-12	0.020254	0.001693	-39	0.007751	0.005508
14	-0.00078	-0.00037	-13	0.009921	0.001725	-40	0.005709	0.005485

ตารางที่ 3.20 ค่า $h_{HP}(n)$ ที่ได้จากการคำนวณ

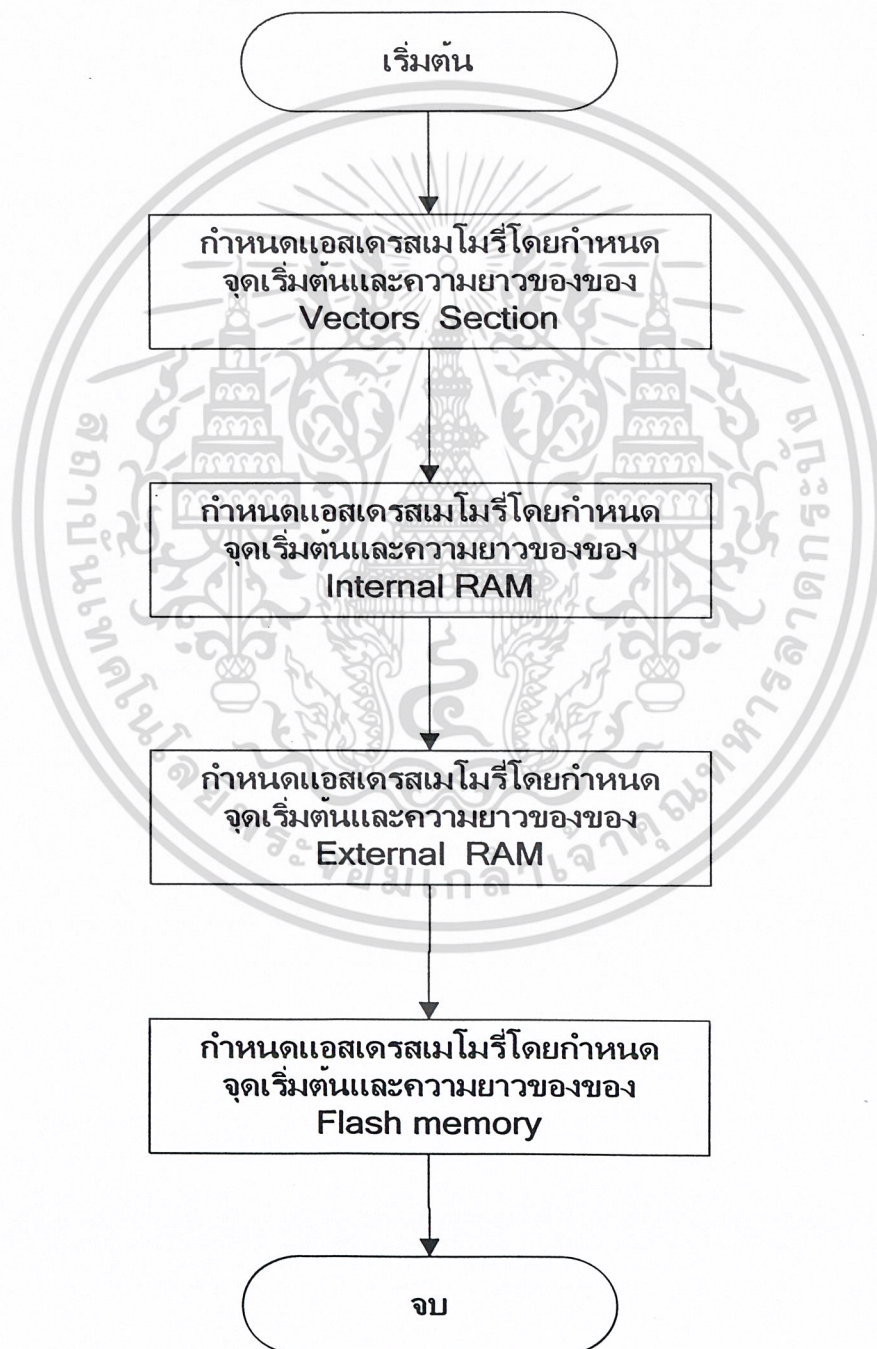
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.13 การเขียนโปรแกรม Code Composer Studio เพื่อให้ทำงานตามสมการต่างๆ

ในที่นี้การเขียน โปรแกรมในส่วนต่างๆทุกสมการต้องมีไฟล์พื้นฐานในการอินเทอร์เฟซเมมโมรี การจัดรีจิสเตอร์ ฟังก์ชันหน้าที่ต่างๆที่เหมือนกันเราจึงจำเป็นต้องสร้างไฟล์พื้นฐานไฟล์ที่สร้างขึ้นมีดังต่อไปนี้

3.13.1 Linker command file (file.cmd)

เป็นไฟล์ที่ทำการเมมโมรี มีแผนภาพการทำงานดังนี้



รูปที่ 3.8 แผนภาพการทำงานโปรแกรม Liker command file

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.13.2 Interrupt vector File

การอินเทอร์รัพเป็นชื่อเรียกกระบวนการที่เข้ามาขัดจังหวะการทำงานโดยปกติของ CPU เมื่อมีการอินเทอร์รัพเกิดขึ้นและมีการ enable การตอบสนองอินเทอร์รัพ ไว้ จะทำการกระโดดไปยัง address ในหน่วยความจำที่กำหนดไว้ เรียกตำแหน่งนั้นว่า interrupt vector address ดังนั้นจะต้องมีการเขียนโปรแกรมย่อย interrupt service routine (ISR) ไว้ที่ interrupt vector

ดังนั้นไฟล์ interrupt vector file นี้เป็นไฟล์ที่เลือกตัวอินเทอร์รัพเพื่อเชื่อมกับตัว ISR (Interrupt Service Routine)

โดยอินเทอร์รัพต่าง ๆ นั้นจะมีตาราง address ดังต่อไปนี้

RESET	0x0000 0000
NMI	0x0000 0020
reserved	0x0000 0040
reserved	0x0000 0060
INT4	0x0000 0080
INT5	0x0000 00a0
INT6	0x0000 00c0
INT7	0x0000 00e0
INT8	0x0000 0100
INT9	0x0000 0120
INT10	0x0000 0140
INT11	0x0000 0180
INT12	0x0000 01a0
INT13	0x0000 01c0
INT14	0x0000 01e0
INT15	0x0000 0200

ตารางที่ 3.21 แอสแตรสของตัวอินเทอร์รัพต่างๆ

3.13.3 Initialization/Communication File (C6xdskinit.c)

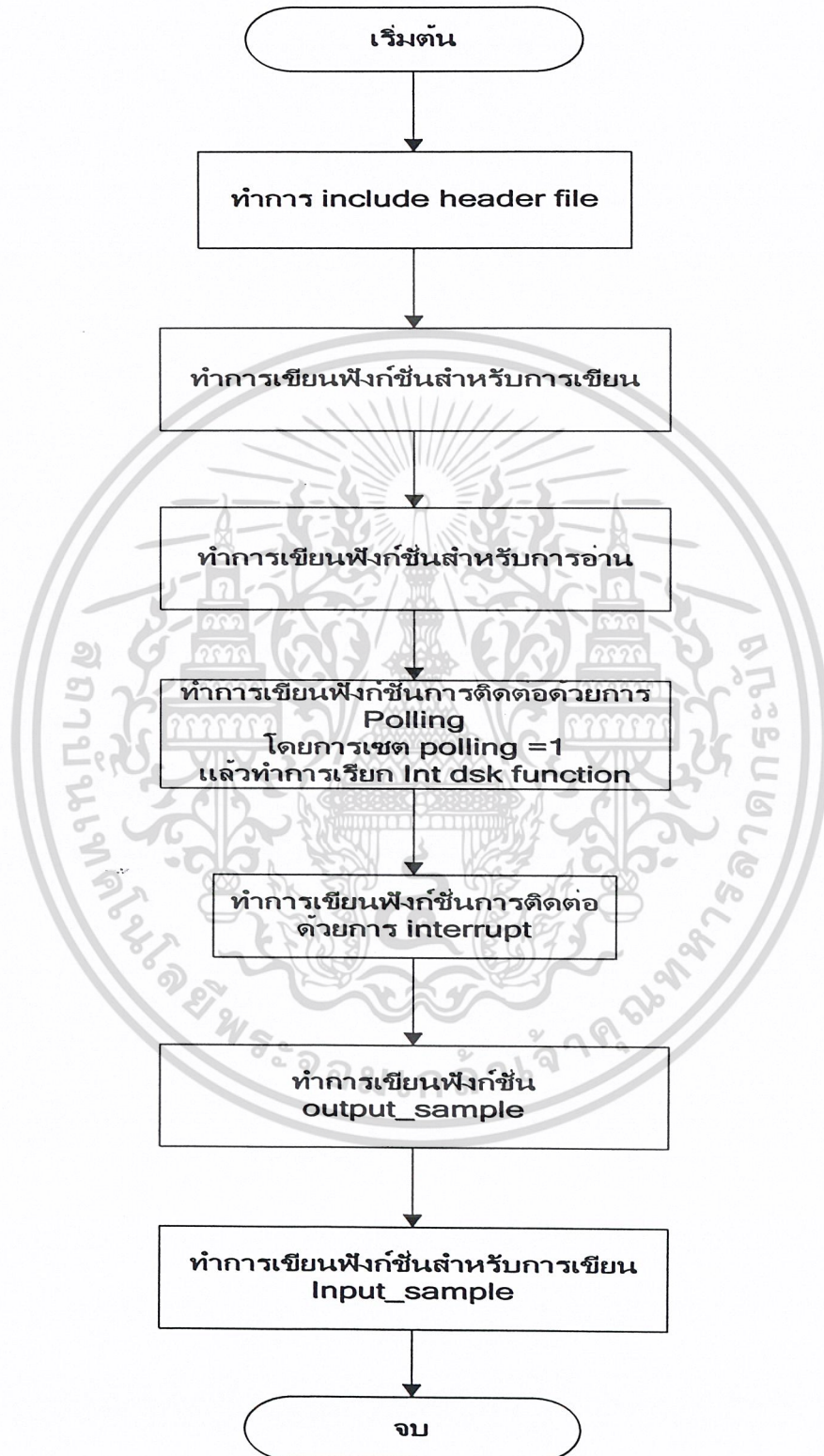
ไฟล์นี้จะ เป็นไฟล์คอมไพล์ร่วมกับฟังก์ชันต่างๆ ไว้ ฟังก์ชันอินเทอร์รัพ (comm.intr) การโพล (comm.poll) และฟังก์ชันการรับแซมเปิ้ลและส่งแซมเปิ้ลออก (Input sample, Output sample) และฟังก์ชัน Mcbsp0_read, Mcbsp0_write ฟังก์ชันเหล่านี้จะทำการติดต่อกับ DSK บอร์ดและยังมีการ include header file ทั้งหมดคือไฟล์ ดังนี้

- C6x.h เป็นไฟล์ที่โปรแกรม CCS มีไว้ให้เราสามารถนำมาเรียกใช้ได้เลยซึ่งไฟล์นี้เป็นการเชื่อมโยงภาษาซีให้เข้ากับภาษาแอสเซมบลี โดยมีแผนผังการทำงานดังรูปที่ 3.9
- C6xdsk.h เป็นไฟล์ที่ทำการกำหนดค่าต่างๆ ให้ภาษาซีนั้นรู้จักกับรีจิสเตอร์ต่างๆ
- C6xdskinit.h เป็นไฟล์ที่กำหนดชนิดของฟังก์ชันที่เราเขียนในโปรแกรมหลักว่าให้เป็นชนิดตัวแปรชนิดใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- C6xinterrupts.h เป็นไฟล์กำหนดค่าเริ่มต้นของตัวอินเทอร์รัพของบอร์ดที่เราทำการเลือก

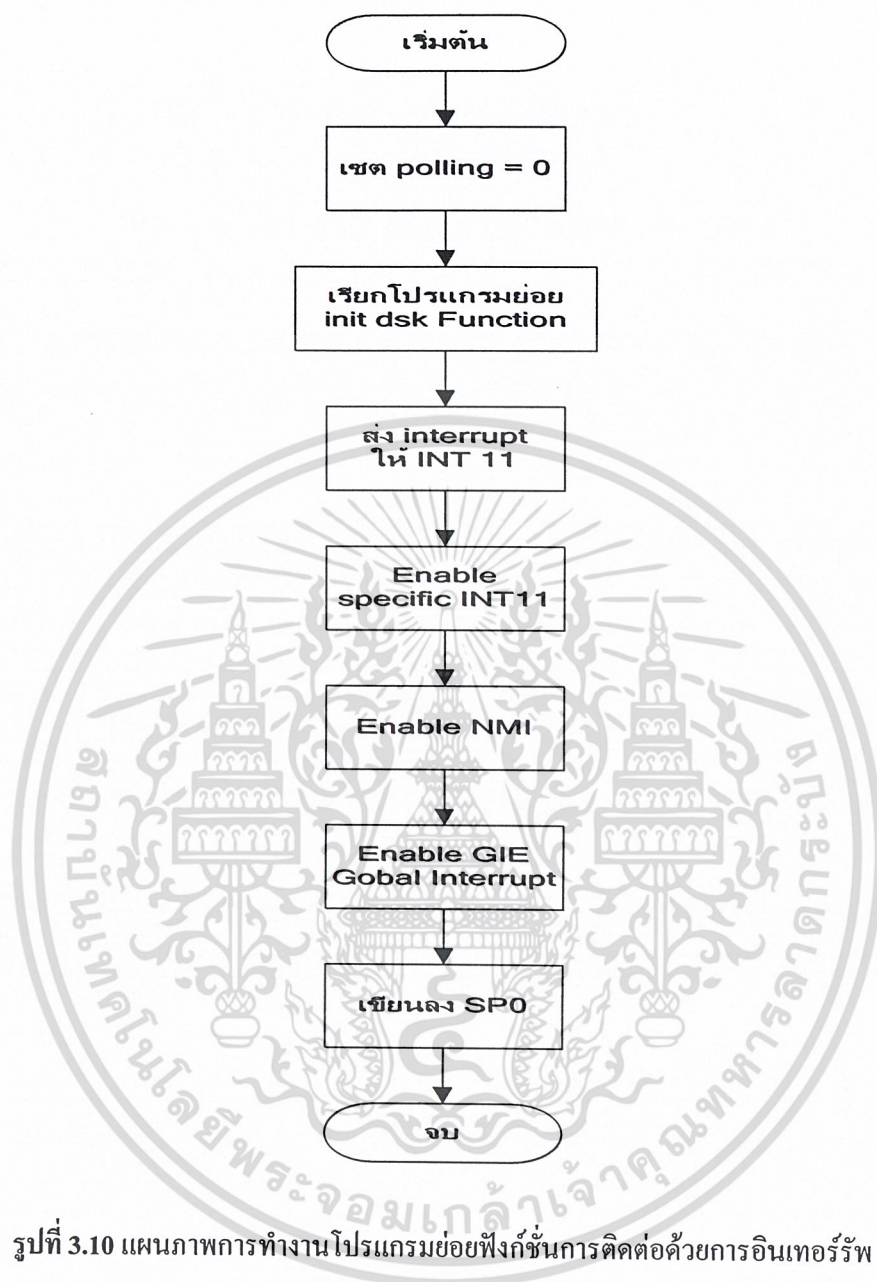
โดยโปรแกรมหลักมีแผนผังการเขียนโปรแกรมดังนี้



รูปที่ 3.9 แผนภาพการทำงานโปรแกรม Initialization/Communication File

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมย่อยฟังก์ชันการติดต่อดูด้วยการอินเทอร์รัพ



รูปที่ 3.10 แผนภาพการทำงานโปรแกรมย่อยฟังก์ชันการติดต่อดูด้วยการอินเทอร์รัพ

3.13.4 โปรแกรมหลัก (Main Program)

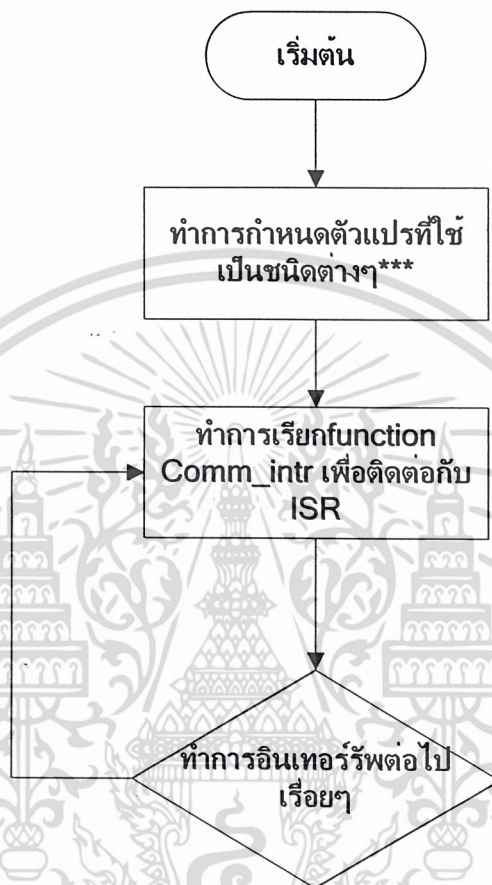
โปรแกรมนี้อาจจะเป็นโปรแกรมหลักคือ เราจะทำการเขียนสมการต่างๆที่ตัวโปรแกรมนี้อ และการเขียนโปรแกรมรับค่าอินพุตนั้นจะสามารถทำได้สองวิธีคือ

- แบบอินเทอร์รัพ
- แบบโพล

ในชิ้นงานนี้ขอเสนอแบบอินเทอร์รัพเพียงแบบเดียวคือแบบอินเทอร์รัพเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเขียนแบบอินเทอร์รัพชั่นเมื่อเกิดการอินเทอร์รัพแล้วระบบจะทำการติดต่อ ISR (interrupt Service Routine) โดยอัตโนมัติเพราะฉะนั้นการอินเทอร์รัพหนึ่งครั้ง รับหนึ่งแชนเปลี่จะนำเข้าไป และก็จะจับเข้าทำการประมวลผล (Process) ตามสมการต่างๆ โปรแกรมหลักก็มีการทำงานดังนี้

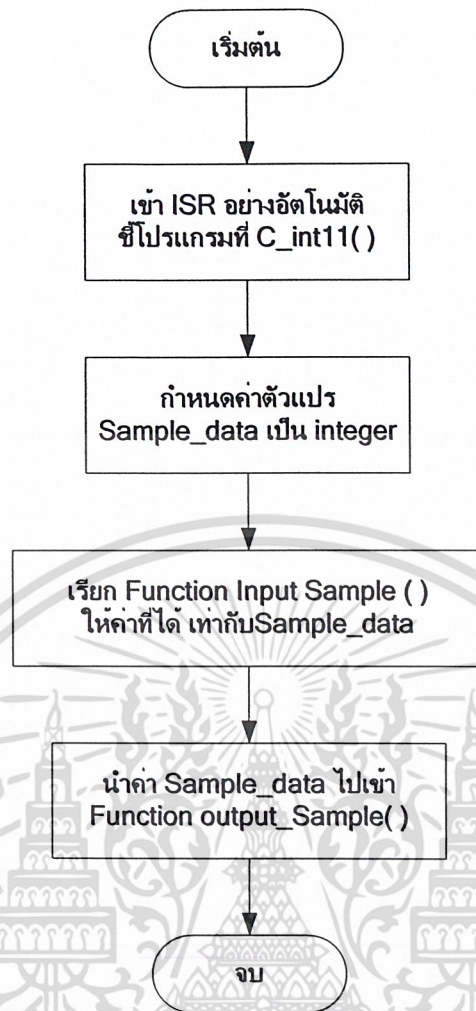


รูปที่ 3.11 แผนภาพการทำงาน โปรแกรมหลัก

***หมายเหตุ เป็นการกำหนดตัวแปรที่ใช้ในแต่ละโปรแกรม

ต่อไปเราจะกล่าวถึงเมื่อเข้าฟังก์ชัน comm_intr โปรแกรมก็จะถูกเรียกเข้า ISR ที่ชี้ค่าเป็น c_int11 แล้วทำการรับ input sample เข้ามาทำการประมวลผลต่างๆ เพราะฉะนั้นทุกโปรแกรมที่เขียนต่อไปก็ทำการเปลี่ยนแปลงโปรแกรมแค่ส่วน c_int11 เท่านั้น

เพราะฉะนั้นเริ่มแรกเราจะทำการเขียนโปรแกรมให้รับค่าเข้ามาแล้วเก็บค่าไว้ที่ตัวแปรหนึ่งแล้วนำค่าออกไปเลยเพื่อให้ง่ายต่อการโปรแกรม โปรแกรมในสมการต่อไปเราเรียกโปรแกรมนี้อันว่า loop_intr โปรแกรมนี้จะเป็นโปรแกรมที่ทำงานรับอินพุตแล้วทำส่งออกเลยทำให้อินพุตจะเท่ากับเอาที่พุทมีการทำงานดังนี้



รูปที่ 3.12 แผนภาพการทำงานฟังก์ชัน comm_intr (ติดต่อ ISR) หรือ c_int11

เพราะฉะนั้นต่อไปเราก็จะทำการเขียนโปรแกรมต่อไปเราจะแก้ไขในส่วนที่เป็น comm_intr หรือ ส่วนที่เป็น c_int11 เท่านั้น

ในส่วนต่อไปเราก็จะทำการเขียนโปรแกรมในส่วนต่างๆดังนี้

- เอ็กโค
- รีเวิร์บ
- คอรัส
- ตัวกรองความถี่แบบ FIR

โดยจะเปลี่ยนแปลงในส่วนโปรแกรมหลักในส่วน c_int11 เท่านั้น

```

ตัวอย่างโปรแกรม Loop_intr

interrupt void c_int(11)           // interrupt service routine
{
    int sample_data;

    sample_data = input_sample(); //input
    output_sample(sample_data);   //output
    return;
}

void main()
{
    comm_intr();
    while(1)
}

```

เปลี่ยนเฉพาะส่วนนี้

3.13.4.1 เอ็กโค (Echo)

จากบทที่ 2 สมการเอ็กโค คือ

$$y[n] = x[n] + \gamma \cdot x[n-d] \quad (2.9)$$

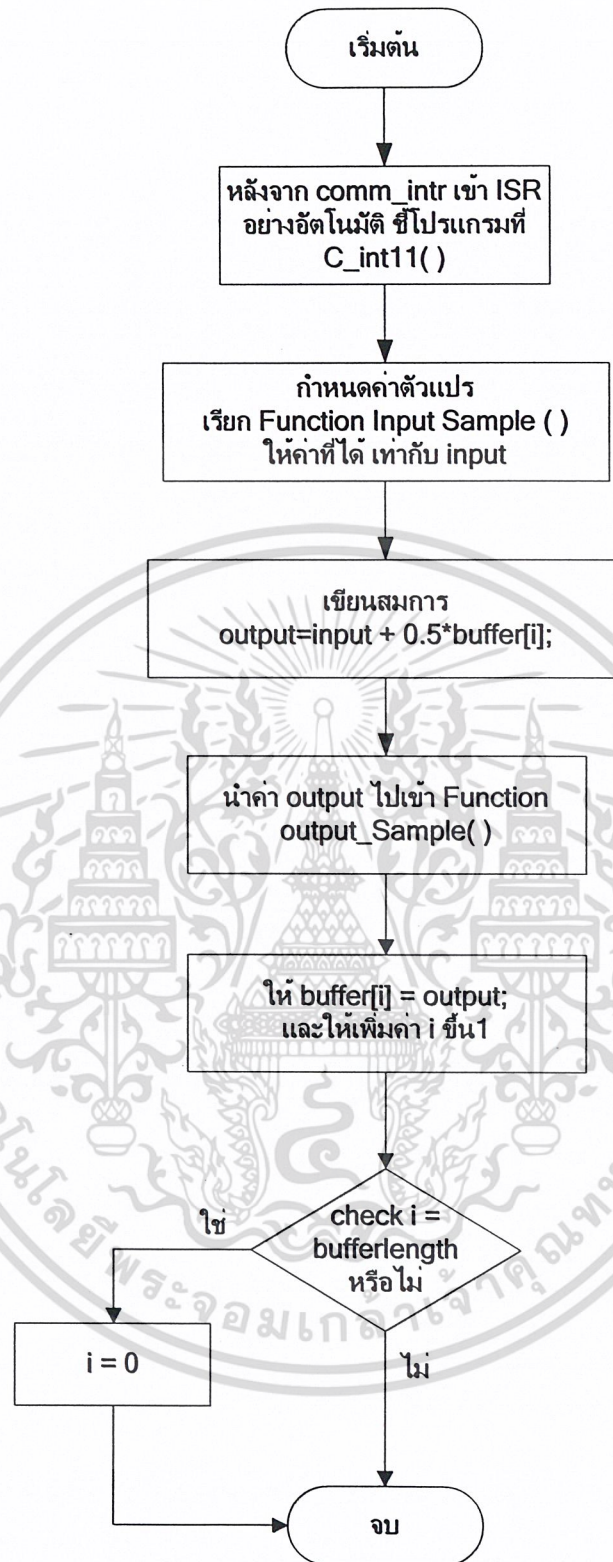
จากสมการ เราเห็นว่าค่าเอาต์พุตจะมีอินพุตที่มีการดีเลย์ด้วยเพราะฉะนั้นจะมีการสร้างตัวแปรที่เป็นบัฟเฟอร์เพื่อเก็บค่าไปเรื่อยๆเพื่อให้มีการเรียกค่าย้อนกลับได้

เราจะเห็นค่าที่เป็นสัมประสิทธิ์นั้นจะเป็นค่าคงที่ γ ที่เราคูณเข้าไปให้เสียงสะท้อนมีค่าลดทอนลงและค่าดีเลย์นั้นเป็นการตั้งว่าให้ค่าการสะท้อนยาวหรือสั้นตามค่ามากน้อย

ก่อนที่จะเขียน โปรแกรมเราจะทำการเพิ่มกำหนดตัวแปรต่างๆที่เราใช้ใน โปรแกรมหลักมีดังต่อไปนี้

- bufferlength เป็นค่าคงที่กำหนดการสะท้อนให้มากหรือน้อย
- กำหนดตัวแปรที่เก็บ buffer เป็น array ที่มีความยาวเท่ากับ Bufferlength
- ตัวแปรที่ใช้รันค่า array เป็น i
- ตัวแปรเก็บค่า input,output

โดยมีการเขียนโปรแกรมดังนี้



รูปที่ 3.13 แผนภาพการเขียนโปรแกรมเอ็กโค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Echo

```

short input, output;
short bufferlength = 3000;           //ขนาดบัฟเฟอร์
short buffer[3000];                 //สร้าง buffer เป็น array
short i = 0;
interrupt void c_int11()            //ISR
{
    input = input_sample();          //รับอินพุตแชนเนล
    output=input + 0.5*buffer[i];    //แชนเนลเก่าที่ถูกลดทอน+แชนเนลใหม่
    output_sample(output);          //ทำการนำเอาที่พุทออก

    buffer[i] = output;              //นำค่าที่ได้เก็บไว้ในบัฟเฟอร์เพื่อเป็นเสียงสะท้อน
    i++;                             //เพิ่ม i ทีละ หนึ่ง
    if (i >= bufferlength) i = 0;    //ถ้าบัฟเฟอร์เท่ากับที่กำหนด ให้ i=0
}

main()
{
    comm_intr();
    while(1);
}

```

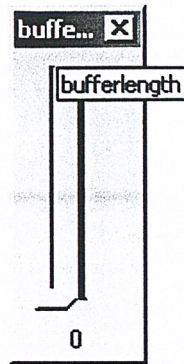
ต่อมาเมื่อทำงานได้เป็นเอ็คโคแล้วเราสามารถทำการเขียนโปรแกรมเสริมคือ file.gel (General Extension Language) คือการเขียนให้เป็น slide bar สามารถปรับค่าตัวแปรได้ในที่นี้เราจะปรับตัวแปร bufferlength ดังนี้

```

/*echo.gel Create slider*/
menuitem "echo delay" /*menu name */
slider bufferlength(0,3000,1,200,bufferlengthparameter)
{
    bufferlength = bufferlengthparameter;
}

```

แล้วทำการโหลด GEL โดยไปที่ File → Load GEL ทำการโหลด แล้วไปที่ GEL เลือกชื่อ menu ที่เราตั้งไว้ออกมาจะได้หน้าต่างดังนี้



รูปที่ 3.14 slide bar ที่ได้

3.13.4.2 รีเวิร์บ (Reverb)

จากบทที่ 2 สมการรีเวิร์บคือ

$$y[n] = x[n] + \gamma \cdot x[n-d_1] + \gamma^2 \cdot x[n-d_2] + \gamma^3 \cdot x[n-d_3] + \gamma^4 \cdot x[n-d_4] + \gamma^5 \cdot x[n-d_5] + \gamma^6 \cdot x[n-d_6] + \gamma^7 \cdot x[n-d_7] + \gamma^8 \cdot x[n-d_8] \quad (2.10)$$

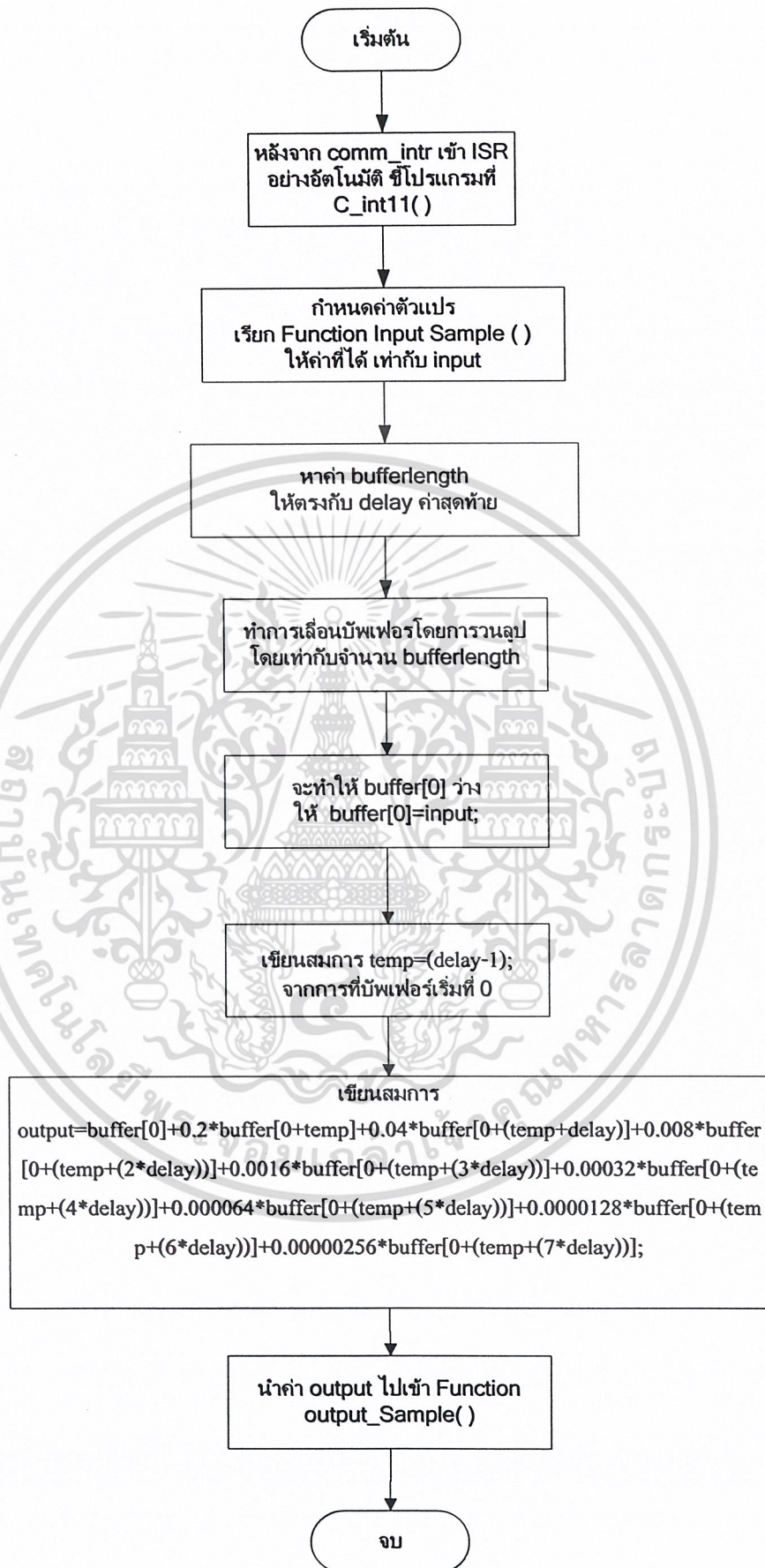
จากสมการเราก็ใช้บัฟเฟอร์เก็บค่าเหมือนกับการทำเอ็คโคและให้ง่ายต่อการสร้างเราเลยทำให้การคิดเลขในแต่ละค่าห่างกันเป็นค่าเดียวไปเรื่อยๆ โดยค่าแรกให้เป็น 0 เพราะฉะนั้นจะเริ่มจาก 0 ถ้าให้ค่าห่าง = 500 ค่า $d_1 = 499$, $d_2 = 999$, $d_3 = 1499$, $d_4 = 2999$, $d_5 = 3499$, $d_6 = 3999$

แต่ครั้งนี้ในการเก็บบัฟเฟอร์เราจะอิงตำแหน่งเป็นสำคัญเพราะฉะนั้นบัฟเฟอร์เราจะเป็นบัฟเฟอร์ที่เลื่อนได้โดยค่าซ้ายไปเก็บค่าขวามือทุกครั้งที่มีการอินเทอร์พ็พบัฟเฟอร์จะทำการเลื่อนขวาทุกตัว

ต่อมาก็ทำการสร้างตัวแปรเพื่อเอาไว้ใช้ตัวโปรแกรมหลัก

- bufferlength เป็นตัวบอกค่าสูงสุดของarray
- กำหนดตัวแปรที่เก็บ buffer เป็น array ที่มีความยาวมากที่สุดที่เราต้องการ
- ตัวแปร delay เป็นตัวกำหนดค่าความห่างของ delay
- ตัวแปรที่ใช้รับค่า array เป็น i
- ตัวแปรเก็บค่า input,output
- ตัวแปร temp ไว้สำหรับการคำนวณค่าแล้วพักไว้

โดยมีการเขียนโปรแกรมดังนี้



รูปที่ 3.15 แผนภาพการเขียนโปรแกรมรีเวิบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Reverb

```

short input, output;
short bufferlength;           //ตัวแปรเพื่อหา bufferlength
short delay=500;
short buffer[8000];          //สร้างบัฟเฟอร์จำนวนมากไว้ก่อน
short i;
short temp;

interrupt void c_int11()      //ISR
{
  input = input_sample();
  bufferlength=(8*delay)-1;
  for (i=bufferlength; i>0; i--)
  {
    buffer[i]=buffer[i-1];    //ทำการเลื่อนบัฟเฟอร์ไปทางขวา
  }
  buffer[0] = input;          //เก็บอินพุตใหม่เข้าบัฟเฟอร์ซ้ายสุดที่ว่าง
  temp=(delay-1);
  output=buffer[0]+0.2*buffer[0+temp]+0.04*buffer[0+(temp+delay)]+0.008
*buffer[0+(temp+(2*delay))]+0.0016*buffer[0+(temp+(3*delay))]+0.00032
*buffer[0+(temp+(4*delay))]+0.000064*buffer[0+(temp+(5*delay))]+0.000
0128*buffer[0+(temp+(6*delay))]+0.00000256*buffer[0+(temp+(7*delay))];
  output_sample(output);
}

main()
{
  comm_intr();
  while(1);
}

```

ต่อมาทำการเขียน File GEL โดยทำเหมือนเดิม โดยครั้งนี้จะปรับเปลี่ยนพารามิเตอร์ชื่อ delay ได้โปรแกรมดังนี้

```

/*reverb.gel Create slider*/

menuitem "reverb delay"

slider delay(0,1000,1,200,delayparameter)
{
  delay = delayparameter;    //ให้เปลี่ยนที่ตัวแปร delay ตัวแปร
}

```

3.13.4.3 คอรัส (Chorus)

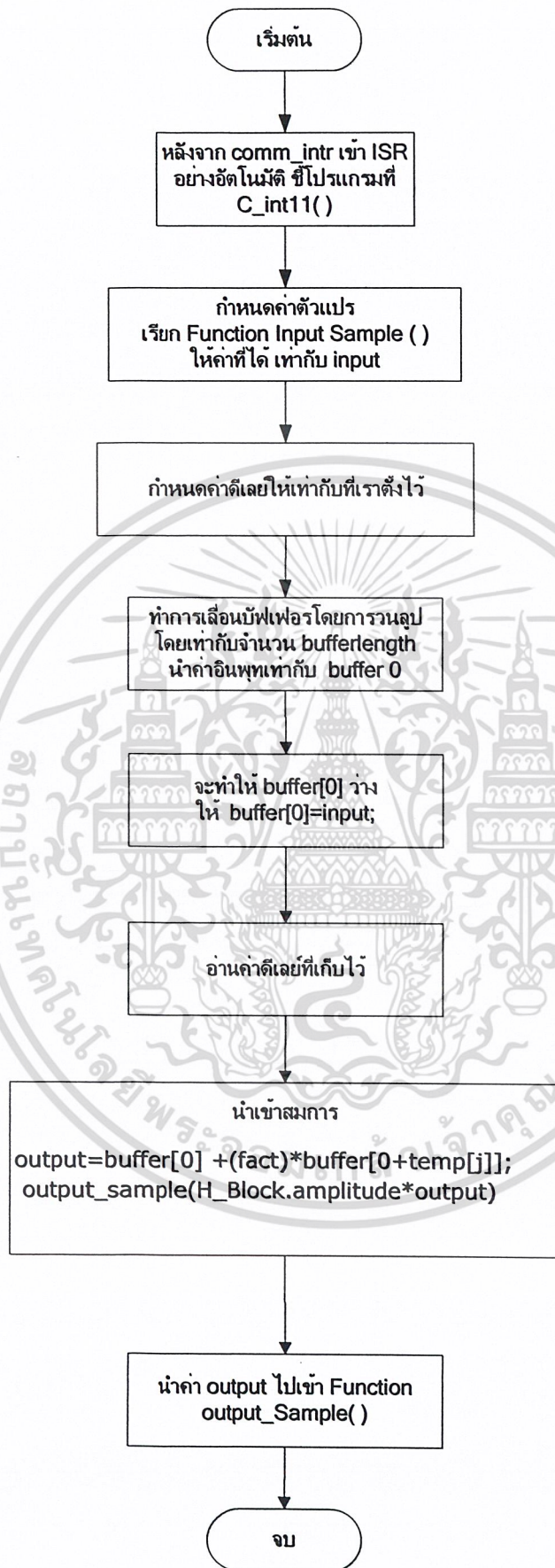
จากบทที่ 2 สมการคอรัสคือ

$$y[n] = x[n] + x[n - \Delta[n]] \quad (2.11)$$

จากสมการจะเห็นว่าค่าเฉลี่ยของสมการคอรัสจะเพิ่มขึ้นจนถึงค่า ค่าหนึ่งแล้วก็จะลดลง ไปจนถึงค่าต่ำสุด ค่าสูงสุด, อัตราการเพิ่มขึ้น และการลดลงของฟังก์ชัน $\Delta[n]$ สามารถจัดการกับตัวแปรในการทำเสียงคอรัส โดยปกติแล้วการทำเสียงคอรัสจะถ่วงเวลาให้ล่าหลังระหว่าง 40msec ถึง 60msec ที่อัตรา 0.25 Hz

เราจะสร้างค่าเฉลี่ยต่างๆแล้วเก็บในอาเรย์เพื่อให้ได้ค่าเฉลี่ยตามทฤษฎี
แผนภาพการเขียนโปรแกรมคอรัสมีดังนี้





รูปที่ 3.16 แผนภาพการเขียนโปรแกรมคอร์รัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Chorus

```

#define HOST_INTFC_C
#include "HPI.h"
#pragma DATA_SECTION(H_Block, "HostBlk");
HostInterfaceBlock H_Block = {
4,0
};
#define N 320
short temp[N]=
{320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332,
333, 334, 335, 336, 337, 338, 339,
340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352,
353, 354, 355, 356, 357, 358, 359,
360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372,
373, 374, 375, 376, 377, 378, 379,
380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392,
393, 394, 395, 396, 397, 398, 399,
400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412,
413, 414, 415, 412, 416, 418, 419,
420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432,
433, 434, 435, 436, 437, 438, 439,
440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452,
453, 454, 455, 456, 457, 458, 459,
460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472,
473, 474, 475, 476, 477, 478, 479,
480, 479, 478, 477, 476, 475, 474, 473, 472, 471, 470, 469, 468,
467, 466, 465, 464, 463, 462, 461,
460, 459, 458, 457, 456, 455, 454, 453, 452, 451, 450, 449, 448,
447, 446, 445, 444, 443, 442, 441,
440, 439, 438, 437, 436, 435, 434, 433, 432, 431, 430, 429, 428,
427, 426, 425, 424, 423, 422, 421,
420, 419, 418, 417, 416, 415, 414, 413, 412, 411, 410, 409, 408,
407, 406, 405, 404, 403, 402, 401,
400, 399, 398, 397, 396, 395, 394, 393, 392, 391, 390, 389, 388,
387, 386, 385, 384, 383, 382, 381,
380, 379, 378, 377, 376, 375, 374, 373, 372, 371, 370, 369, 368,
367, 366, 365, 364, 363, 362, 361,
360, 359, 358, 357, 356, 355, 354, 353, 352, 351, 350, 349, 348,
347, 346, 345, 344, 343, 342, 341,
340, 339, 338, 337, 336, 335, 334, 333, 332, 331, 330, 329, 328,
327, 326, 325, 324, 223, 322, 321};
short input, output;
short buffer[480]; //create buffer
short i,j,count;
float fact;

interrupt void c_iny11() //ISR
{
input = input_sample(); //newest input sample data
for (i=479; i>0; i--)
{
buffer[i]=buffer[i-1];
}
Buffer[0] = input; //store newest input sample

for(count=0;count<100;count++)
{
fact=H_Block.bufferlength/100;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

output=buffer[0]+(fact)*buffer[0+temp[j]];
output_sample(H_Block.amplitude*output);
}
    if(j>318)
        j=0;
    else
        j=j+1;
}

main()
{
    j=0;
    count=0;
    comm_intr();                //init DSK, codec, McBSP
    while(1);                   //infinite loop
}

```

3.13.4.4 ตัวกรองความถี่แบบ FIR

ขั้นตอนแรกเราต้องหาสัมประสิทธิ์จากฟิลเตอร์ที่เราต้องการแล้วนำค่านั้นมาใส่ลงในไฟล์ file.cof โดยจากที่คำนวณข้างต้นและทำการนอร์มอไรส์และราวน์ค่าได้ซึ่งมีรูปแบบดังนี้

```

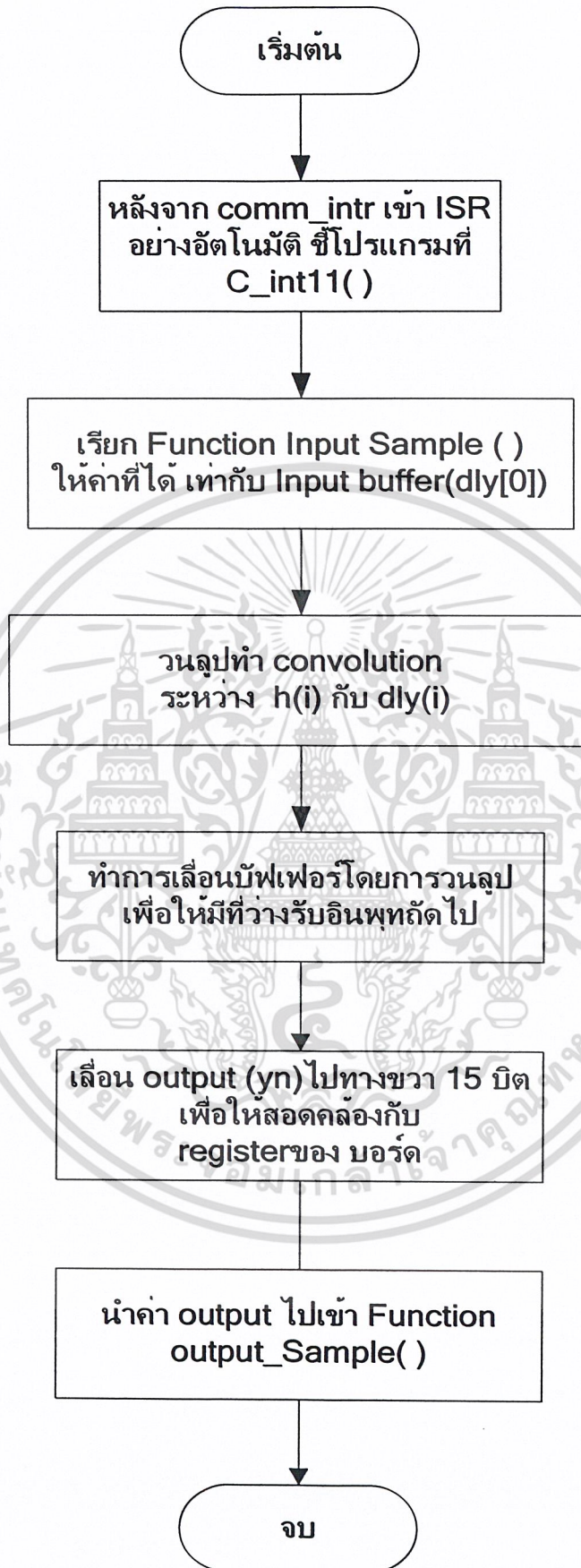
#define N 81                //order ของ ฟิลเตอร์
short hbp2200[N] = { -152, -163, -141, 52, -51, 122, 273, -182, -267,
66,-56, 104, 437, -155, -455, 57, -38, 51, 650, -47, -729,
-9, 19, -48, 937, 213, -1154, -206, 146, -225, 1407, 839,
-1994, -812, 478, -661, 2818, 3495, -6156, -6397, 7593,
7593, -6397, -6156, 3495, 2818, -661, 478, -812, -1994,
839, 1407, -225, 146, -206, -1154, 213, 937, -48, 19, -9,
-729, -47, 650, 51, -38, 57, -455, -155, 437, 104, -56, 66,
-267, -182, 273, 122, -51, 52, -141, -163, 152
};

```

ต่อมาก็ทำการinclude file.cof และ กำหนดตัวแปรที่ใช้ต่างๆลงในโปรแกรมหลักดังนี้

- yn เซตค่าให้เป็น 0 แล้วเป็นตัวเก็บค่าเอาท์พุท
- dly เป็นarray ที่เป็นเหมือนบัฟเฟอร์เก็บค่าอินพุทไว้
- ตัวแปรที่ใช้รับค่า array เป็น i

แล้วก็มีเขียนโปรแกรมตามผังดังนี้



รูปที่ 3.17 แผนภาพการเขียนโปรแกรมฟิลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมฟิลเตอร์แบบ FIR

```
#include "bs2700.cof" //include ค่าสัมประสิทธิ์ที่ออกแบบมา
int yn = 0;
short dly[N];

interrupt void c_int11() //ISR
{
    short i;

    dly[0] = input_sample(); //นำค่า input ใหม่เข้ามา
    yn = 0; //reset ค่า
    for (i = 0; i < N; i++)
        yn += (h[i] * dly[i]); //y(n) = h(i)* x(n-i) + y(n)
    for (i = N-1; i > 0; i--)
        dly[i] = dly[i-1]; //เลื่อนค่าไปทางขวาเพื่อรอรับค่าใหม่

    output_sample(yn >> 15); //เลื่อน 15 บิต เพื่อให้สอดคล้องกับรีจิสเตอร์
    return;
}

void main()
{
    comm_intr();
    while(1);
}
```

เมื่อเราได้ฟิลเตอร์หนึ่งตัวเรานำโปรแกรมมาเขียนใหม่เพื่อให้สามารถรันฟิลเตอร์ได้ถึงสิบตัวโดยพัฒนาเป็นโปรแกรมอิกวอลไลเซอร์เราทำการแบ่งย่านความถี่ออกเป็นสิบย่านเพื่อลองทำการออกแบบฟิลเตอร์สิบย่านพร้อมกันและสามารถปรับแอมพลิจูดฟิลเตอร์ทั้งสิบตัวได้โดยมีโปรแกรมดังนี้

โปรแกรมอิกวอลไลเซอร์โดยใช้ฟิลเตอร์แบบ FIR สิบย่าน

```
#include "LPF450.cof"
#include "BPF1_647.cof"
#include "BPF2_1042.cof"
#include "BPF3_1437.cof"
#include "BPF4_1832.cof"
#include "BPF5_2227.cof"
#include "BPF6_2622.cof"
#include "BPF7_3017.cof"
#include "BPF8_3412.cof"
#include "BPF9_3800.cof"

int yn = 0;
int yn0=0, yn1=0, yn2=0, yn3=0, yn4=0, yn5=0, yn6=0, yn7=0, yn8=0, yn9=0;
short dly[10][N];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

short h[10][N];
interrupt void c_int11()                                //ISR
{
    short i;
    dly[0][0] = input_sample();
    dly[1][0] = dly[0][0];
    dly[2][0] = dly[0][0];
    dly[3][0] = dly[0][0];
    dly[4][0] = dly[0][0];
    dly[5][0] = dly[0][0];
    dly[6][0] = dly[0][0];
    dly[7][0] = dly[0][0];
    dly[8][0] = dly[0][0];
    dly[9][0] = dly[0][0];
    yn = 0;
    yn0=0;
    yn1=0;
    yn2=0;
    yn3=0;
    yn4=0;
    yn5=0;
    yn6=0;
    yn7=0;
    yn8=0;
    yn9=0;
    for (i = 0; i < N; i++)
    {
        yn0 +=(h[0][i]*dly[0][i]);
        yn1 +=(h[1][i]*dly[1][i]);
        yn2 +=(h[2][i]*dly[2][i]);
        yn3 +=(h[3][i]*dly[3][i]);
        yn4 +=(h[4][i]*dly[4][i]);
        yn5 +=(h[5][i]*dly[5][i]);
        yn6 +=(h[6][i]*dly[6][i]);
        yn7 +=(h[7][i]*dly[7][i]);
        yn8 +=(h[8][i]*dly[8][i]);
        yn9 +=(h[9][i]*dly[9][i]);
    }
    for (i = N-1; i > 0; i--)
    {
        dly[0][i] = dly[0][i-1];
        dly[1][i] = dly[1][i-1];
        dly[2][i] = dly[2][i-1];
        dly[3][i] = dly[3][i-1];
        dly[4][i] = dly[4][i-1];
        dly[5][i] = dly[5][i-1];
        dly[6][i] = dly[6][i-1];
        dly[7][i] = dly[7][i-1];
        dly[8][i] = dly[8][i-1];
        dly[9][i] = dly[9][i-1];
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

yn=((H_Block.a0/10)*yn0)+((H_Block.a1/10)*yn1)+((H_Block.a2/10)*yn2)
+((H_Block.a3/10)*yn3)+((H_Block.a4/10)*yn4)+((H_Block.a5/10)*yn5)
+((H_Block.a6/10)*yn6)+((H_Block.a7/10)*yn7)+((H_Block.a8/10)*yn8)
+((H_Block.a9/10)*yn9); //mix filter
yn=H_Block.amplitude*yn;
output_sample(yn >> 15);
return;
}
void main()
{
    short i;
    for (i=0; i<N; i++)
    {
        dly[0][i] = 0;
        dly[1][i] = 0;
        dly[2][i] = 0;
        dly[3][i] = 0;
        dly[4][i] = 0;
        dly[5][i] = 0;
        dly[6][i] = 0;
        dly[7][i] = 0;
        dly[8][i] = 0;
        dly[9][i] = 0;

        h[0][i] = h0[i];
        h[1][i] = h1[i];
        h[2][i] = h2[i];
        h[3][i] = h3[i];
        h[4][i] = h4[i];
        h[5][i] = h5[i];
        h[6][i] = h6[i];
        h[7][i] = h7[i];
        h[8][i] = h8[i];
        h[9][i] = h9[i];
    }
    comm_intr();
    while(1);
}

```

และเมื่อเราออกแบบ โปรแกรมทั้งหมดเรียบร้อยแล้วเราจะเห็นได้ว่าค่าที่เราจะนำมาปรับแก้ โปรแกรมนั้นมีหลายค่าเช่น ค่า Bufferlength ของเอ็ดโค, ค่าคีย์ของรีวิบและค่าแกนของฟิลเตอร์เป็นต้น โดยข้างต้นนั้นเราจะใช้ Gel File ช่วยในการปรับแก้แต่เนื่องจากมีความยุ่งยากในการใช้และ โปรแกรมแต่ โปรแกรมนั้นก็ติดต่อกับผู้ใช้ได้ยากเพราะต้องใช้โปรแกรม Code Composer Studio ตลอดเวลา

เพราะฉะนั้นเราจึงต้องสร้าง GUI (Graphic User Interface) ขึ้นมาให้ติดต่อกับผู้ใช้ได้ง่ายโดยใช้ ภาษา Visual C++ แล้วนำโปรแกรมทุกโปรแกรมที่คอมไพล์เป็นสกุล OUT มาใช้งานเลยโดยติดต่อกับ โปรแกรมลงบอร์ดและปรับค่าตัวแปรให้เป็นรีลไทม์โดยติดต่อ ผ่าน HPI (Host User Interfece) ที่มีคำสั่ง ตามบทที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะต้องเขียนในส่วนของ Linker command file (file.cmd) เป็นดังนี้ เพื่อติดต่อผ่านเมโมรี HPI(Host Block) โดยจะเริ่มที่เมโมรี 80000000H และมีความกว้าง 4H ดังนี้

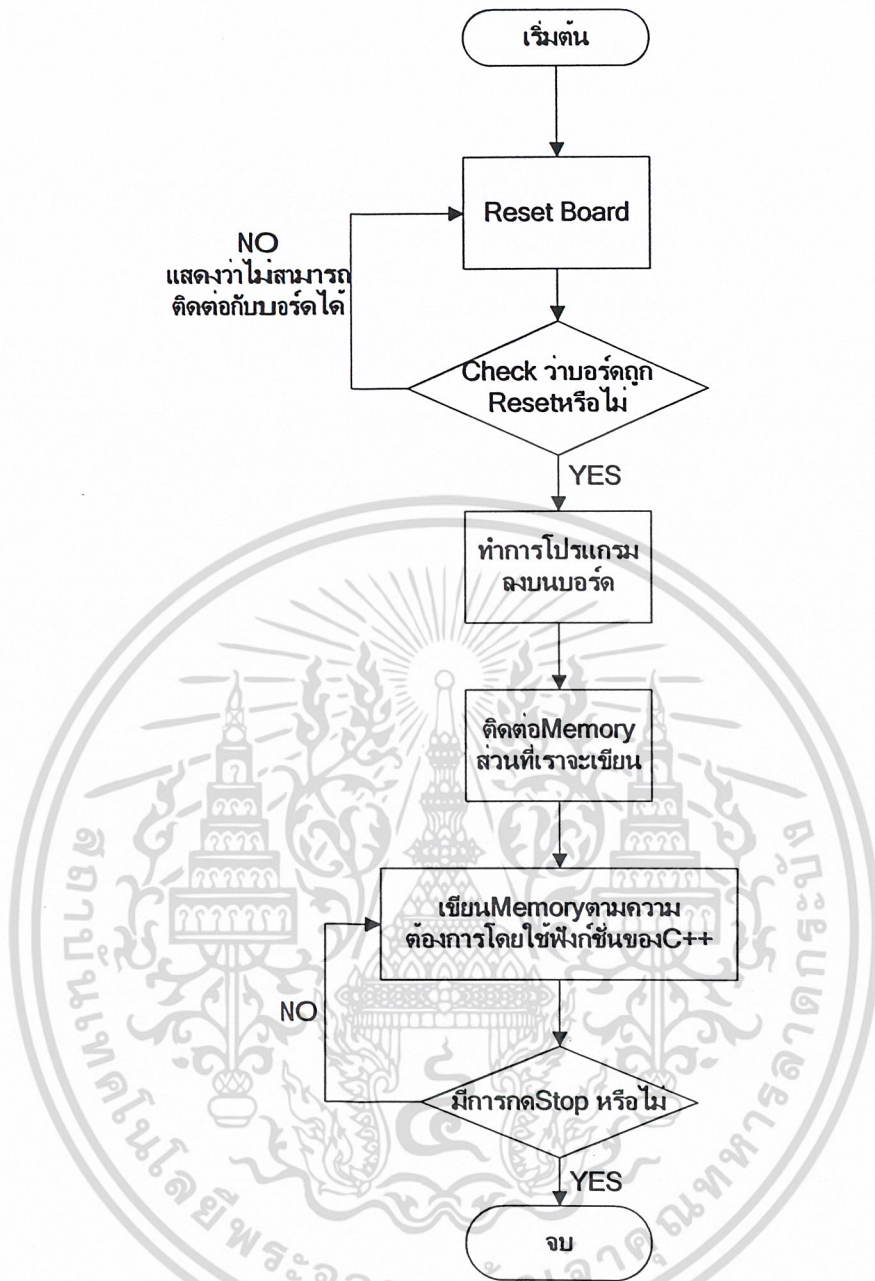
```
/*C6xdsk.cmd Generic Linker command file*/
```

```
MEMORY
{
  VECS:      org =      0h,   len = 0x220
  IRAM:      org = 0x0000220, len = 0x000FDC0 /*internal memory*/
  HOST_CTRL: org = 0x8000000, len = 0x0000040 /* HPI MEMORY*/
  SDRAM:     org = 0x8000050, len = 0x0100000
  FLASH:    org = 0x9000000, len = 0x0002000 /*flash memory*/
}

SECTIONS
{
  HostBlk  > HOST_CTRL
  vectors  > VECS
  .text    > IRAM
  .bss     > IRAM
  .cinit   > IRAM
  .stack   > IRAM
  .system  > SDRAM
  .const   > IRAM
  .switch  > IRAM
  .far     > SDRAM
  .cio     > SDRAM
}
```

และทำการเปลี่ยนตัวแปรที่ใช้ในการปรับค่าในโปรแกรมให้อิงกับกับ HPI ใหม่เลยโดยเราจะไม่ใช้เจลไฟล์โดยเหมือนกับฟิลเตอร์สปีดข่านข้างบนโดยจะแนบโค้ดทั้งหมดที่เปลี่ยนแปลงแล้วและมีความถูกต้องไว้ในภาคผนวก (CD-ROM) รวมทั้งโค้ดของ Visual C++ ด้วย

โดยส่วนของ Visual C++ ในส่วนการโปรแกรมการรีเซตแล้วการเปลี่ยนค่ามีดังนี้



รูปที่ 3.18 แผนภาพการเขียนโปรแกรม GUI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกำรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

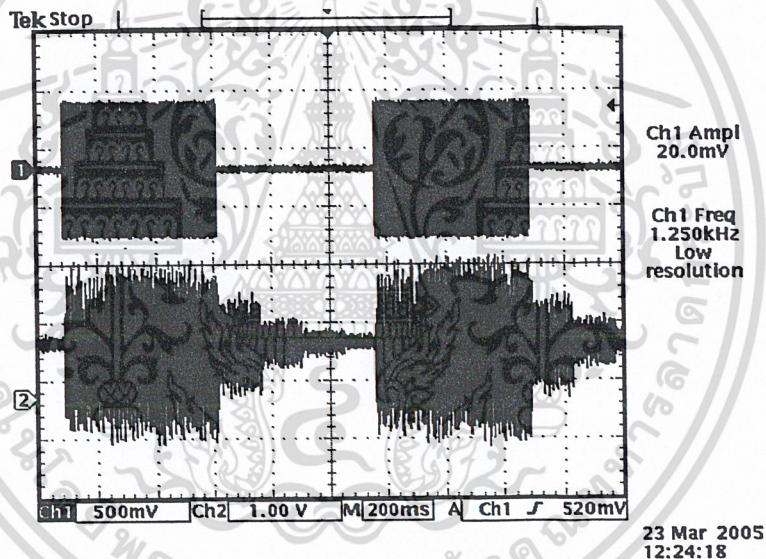
บทที่ 4

การทดลองและผลการทดลอง

บทนี้จะกล่าวถึงการทดลองและแสดงผลที่ได้จากการทดลองของโครงการนี้ ซึ่งมีการทดลองแอมป์, การทดลองรีเฟรช, การทดลองคอร์รัสและการทดลองในส่วนของวงจรองความถี่คิตติคอล (ที่ได้จากการออกแบบจากบทที่ 3) ซึ่งจะแสดงค่าเฟสและแอมพลิจูดของสัญญาณที่ผ่านวงจรองความถี่ผ่านเชิงเลข, การทดลองแอมป์, การทดลองรีเฟรชและการทดลองคอร์รัส

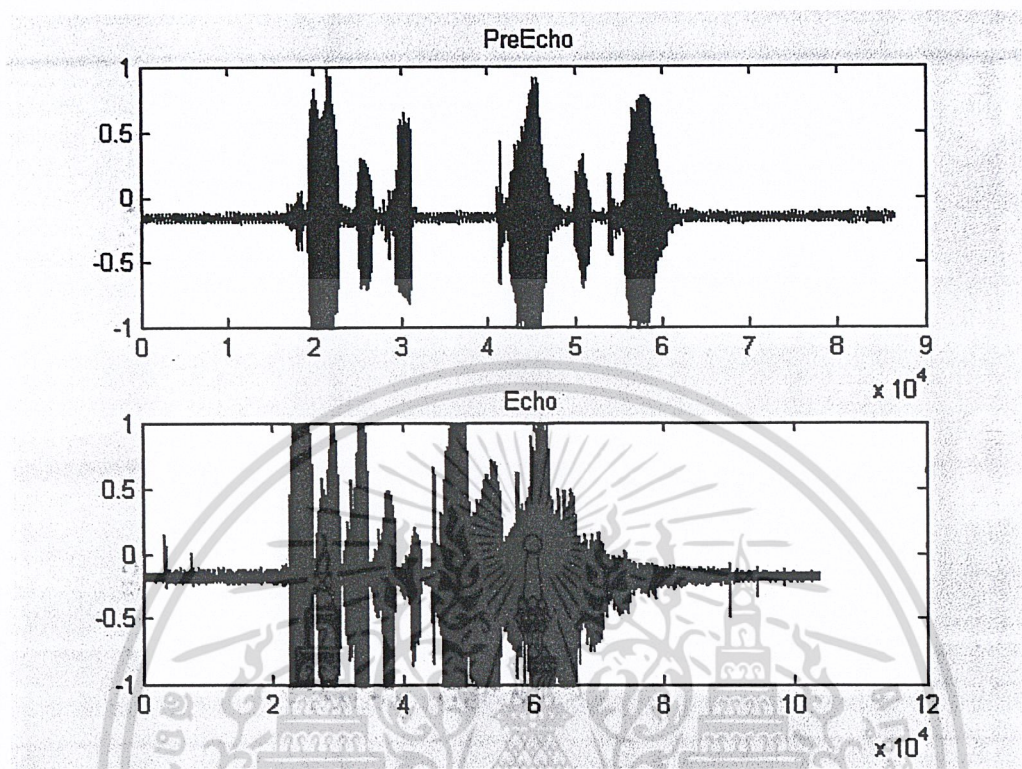
4.1 การทดลองที่ 1 การทดลองผ่านโปรแกรมเอ็ดโค

4.1.1 ทำการทดลองป้อนสัญญาณผ่านบอร์ดที่โปรแกรมเอ็ดโค โดยอินพุทเราจะใช้สัญญาณไซน์ที่ป้อนผ่านอนาล็อกสวิตช์โดยควบคุมโดยคล็อก 1Hz โดยจะได้สัญญาณออกมาเป็นพัลส์รูปไซน์แล้วนำมาเป็นอินพุทป้อนผ่านโปรแกรมเอ็ดโคจะได้เอาท์พุทดังรูปที่ 4.1 โดยจะเห็นค่าดีเลย์ได้ชัด



รูปที่ 4.1 ผลเมื่อป้อนสัญญาณพัลส์รูปไซน์ผ่านโปรแกรมเอ็ดโค

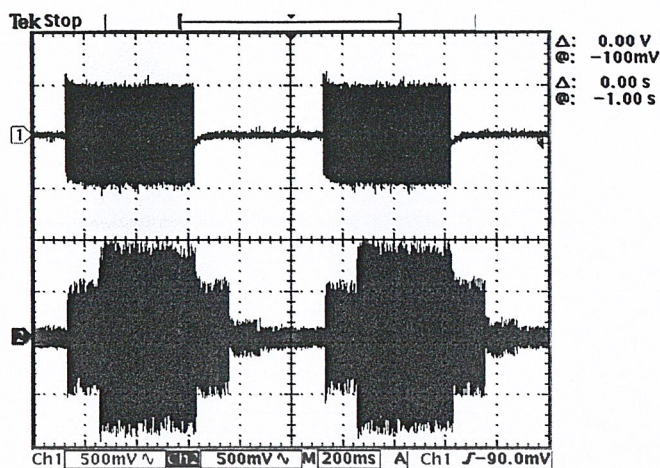
4.1.2 ทำการอัดเสียงเป็นไฟล์นำไฟล์ที่ได้นั้นไปเล่นผ่านโปรแกรมเอ็ด โคแล้วนำกลับมาอัดอีกครั้งหนึ่งแล้วนำเข้ามาแมตแลปเพื่อเทียบกันได้ดังนี้



รูปที่ 4.2 ไฟล์เสียงก่อนและหลังป้อนเข้าโปรแกรมเอ็ด โคโดยพล็อต MATLAB

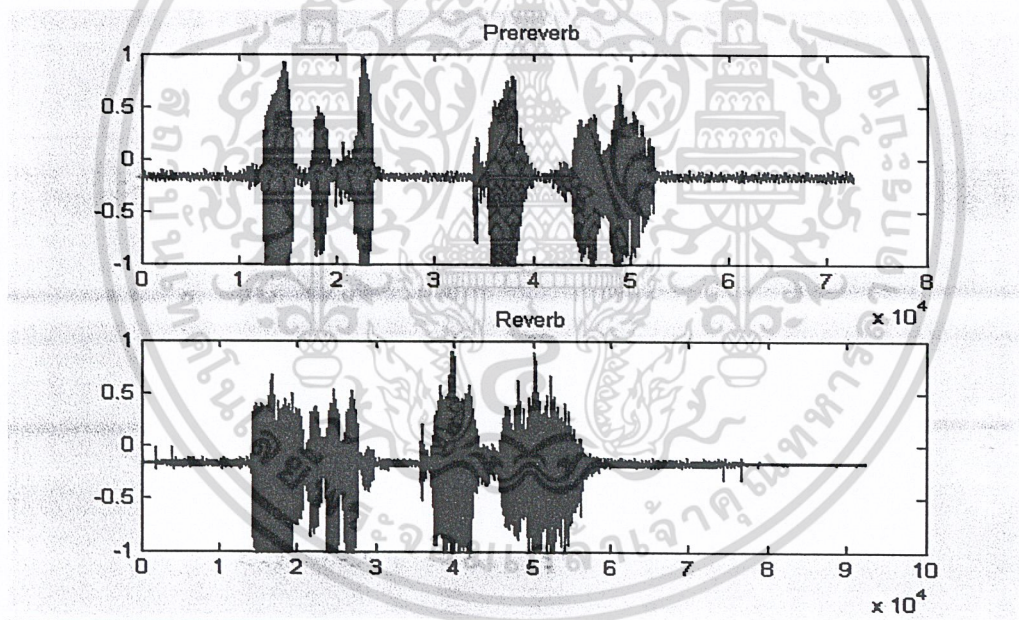
4.2 การทดลองที่ 2 การทดลองผ่านโปรแกรมรีเว็บบ

4.2.1 ทำการทดลองป้อนสัญญาณผ่านบอร์ดที่โปรแกรมเอ็ด โค โดยอินพุตเราจะใช้สัญญาณไซน์ที่ป้อนผ่านอนาล็อกสวิตช์โดยควบคุมโดยคล็อก 1Hz เหมือนการทดลองเอ็ด โค โดยจะได้สัญญาณออกมาเป็นพัลส์รูปไซน์แล้วนำมาเป็นอินพุตป้อนผ่านโปรแกรมรีเว็บบจะได้เอาท์พุตดังรูปที่ 4.2 โดยจะเห็นค่าคิเลี่ยมมาทับกับสัญญาณเดิมทำให้สัญญาณมีขนาดเพิ่มขึ้น



รูปที่ 4.3 ผลเมื่อป้อนสัญญาณพัลส์รูปไซน์ผ่านโปรแกรมรีเวิร์บ

4.2.2 ทำการอัดเสียงเป็นไฟล์นำไฟล์ที่ได้นั้นไปเล่นผ่านโปรแกรมรีเวิร์บแล้วนำกลับมาอัดอีกครั้งหนึ่งแล้วนำเข้าแมตแลปเพื่อตัดเทียบกันได้ดังนี้

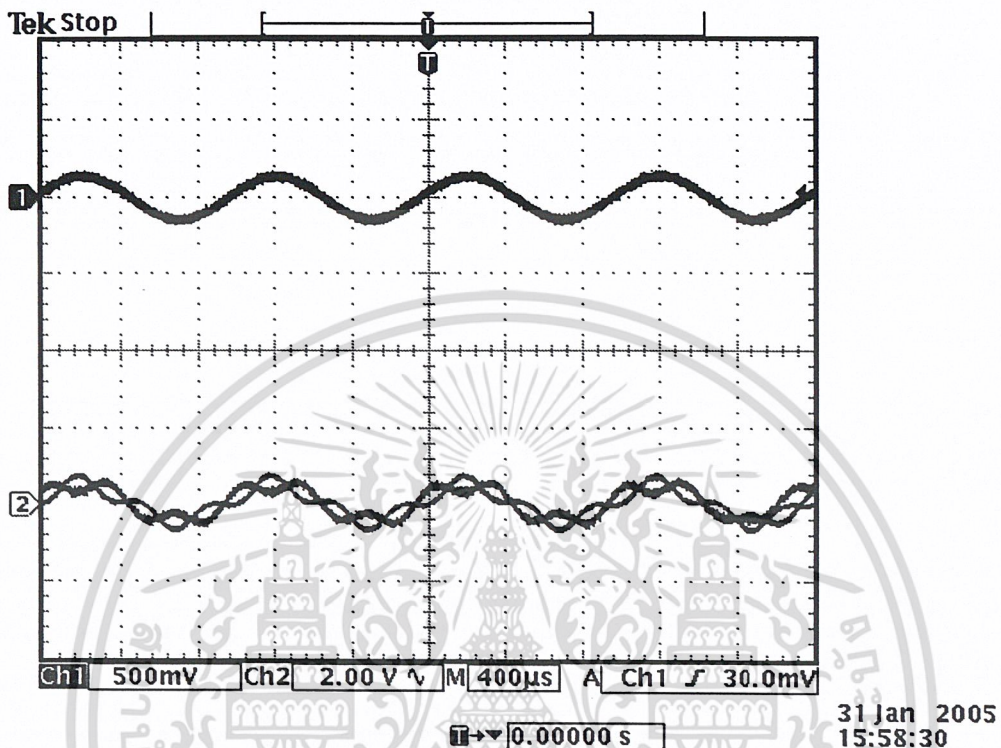


รูปที่ 4.4 ไฟล์เสียงก่อนและหลังป้อนเข้าโปรแกรมรีเวิร์บโดยพล็อต MATLAB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

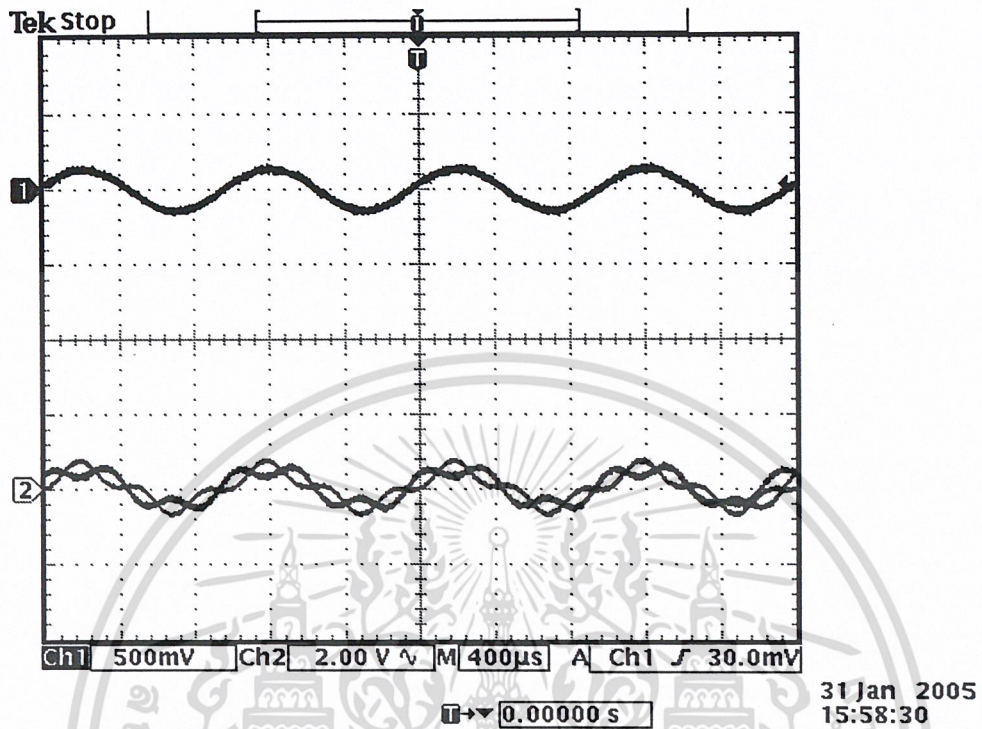
4.3 การทดลองที่ 3 การทดลองผ่านโปรแกรมคอร์ธ

4.3.1 ทำการทดลองป้อนสัญญาณผ่านบอร์ดที่โปรแกรมคอร์ธ โดยอินพุตที่ป้อนเราจะใช้สัญญาณไซน์ 1kHz ป้อนผ่านโปรแกรมคอร์ธที่ปรับค่า 50 จะได้เอาต์พุตดังรูปที่ 4.5



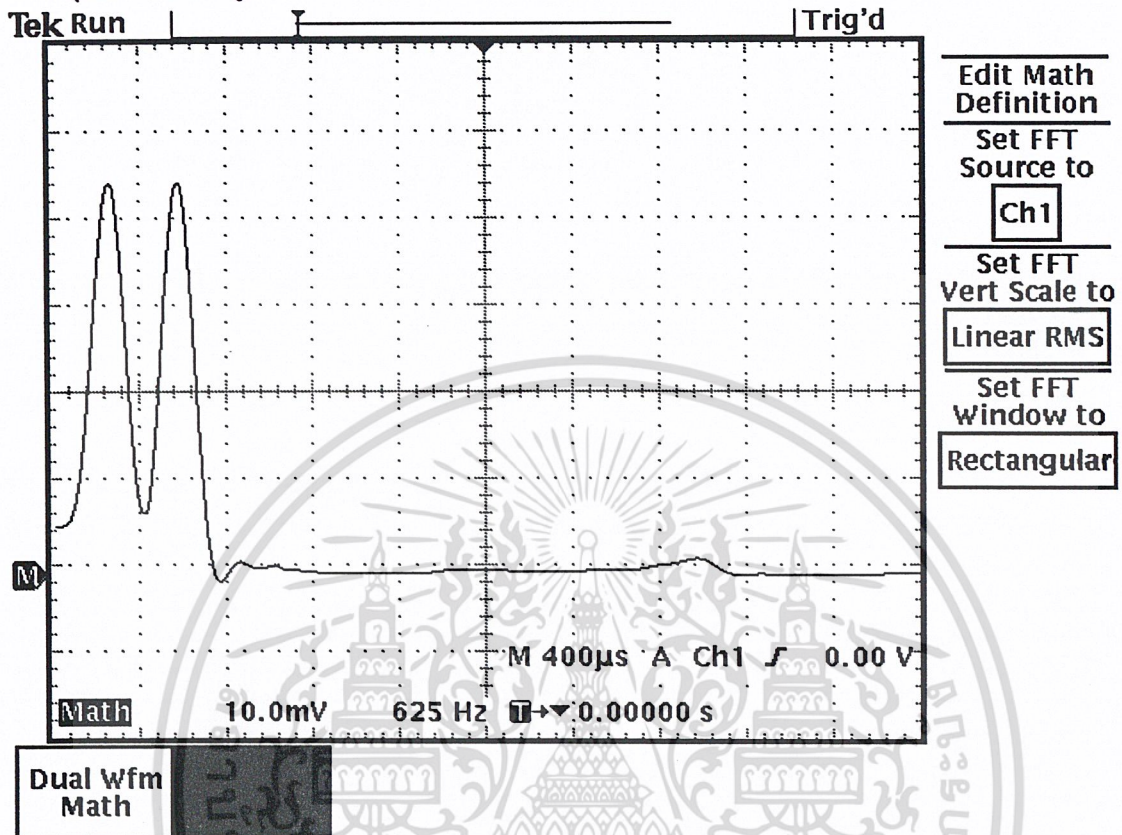
รูปที่ 4.5 ผลเมื่อป้อนสัญญาณผ่านบอร์ดที่โปรแกรมคอร์ธที่ปรับค่า 50

4.3.2 ทำการทดลองป้อนสัญญาณผ่านบอร์ดที่โปรแกรมคอร์ต โดยอินพุตที่ป้อนเราจะใช้สัญญาณไซน์ 1kHz ป้อนผ่านโปรแกรมคอร์ตที่ปรับค่า 100 จะได้เอาต์พุตดังรูปที่ 4.6

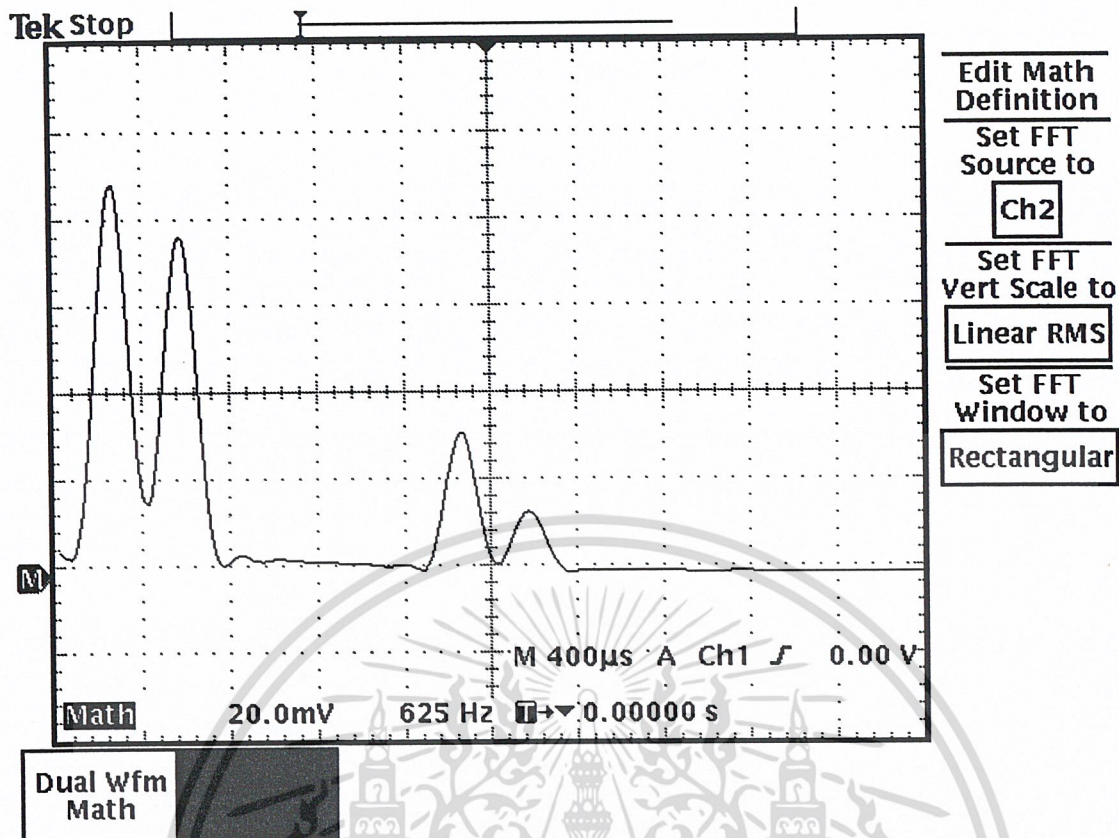


รูปที่ 4.6 ผลเมื่อป้อนสัญญาณผ่านบอร์ดที่โปรแกรมคอร์ตที่ปรับค่า 100

4.3.3 ทำการทดลองป้อนสัญญาณไซน์ผ่านโปรแกรมคอร์ส โดยอินพุตเป็นสัญญาณไซน์ ความถี่ 1kHz เก็บผลการทดลองในแกนความถี่โดยสเปกตรัมของอินพุตมีลักษณะดังรูปที่ 4.7 และ เอาท์พุทมีลักษณะดังรูปที่ 4.8



รูปที่ 4.7 สเปกตรัมของสัญญาณอินพุตก่อนผ่าน โปรแกรมคอร์ส



รูปที่ 4.8 สเปกตรัมของสัญญาณเอาต์พุตหลังจากโปรแกรมคอร์รัส

จากผลการทดลองที่ได้จะเห็นได้ว่าสัญญาณเอาต์พุตที่ได้หลังจากโปรแกรมคอร์รัสมีส่วนของสเปกตรัมที่เพิ่มขึ้นมาจากสเปกตรัมของอินพุต

4.4 การทดลองที่ 4 ผลตอบสนองแอมพลิจูดและเฟสของวงจรกรองความถี่ดิจิทัล

ทำการทดลองวงจรกรองความถี่ผ่านเชิงเลข ในการออกแบบค่าสัมประสิทธิ์โดยใช้โปรแกรม MATLAB กำหนดค่า ในการทดลองนี้ประกอบด้วยตัวกรองความถี่ 10 ตัว ดังนี้

4.4.1 วงจรกรองความถี่ดิจิทัลแบบความถี่ต่ำผ่าน (Digital Lowpass Filter) ที่ความถี่ผ่านเท่ากับ 450Hz

4.4.2 วงจรกรองความถี่ดิจิทัลแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 450Hz ถึง 845Hz

4.4.3 วงจรกรองความถี่ดิจิทัลแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz

4.4.4 วงจรกรองความถี่ดิจิทัลแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 1240Hz ถึง 1635Hz

4.4.5 วงจรกรองความถี่เชิงเลขแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 1635Hz ถึง 2030Hz

4.4.6 วงจรกรองความถี่ดิจิทัลแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 2030Hz ถึง 2425Hz

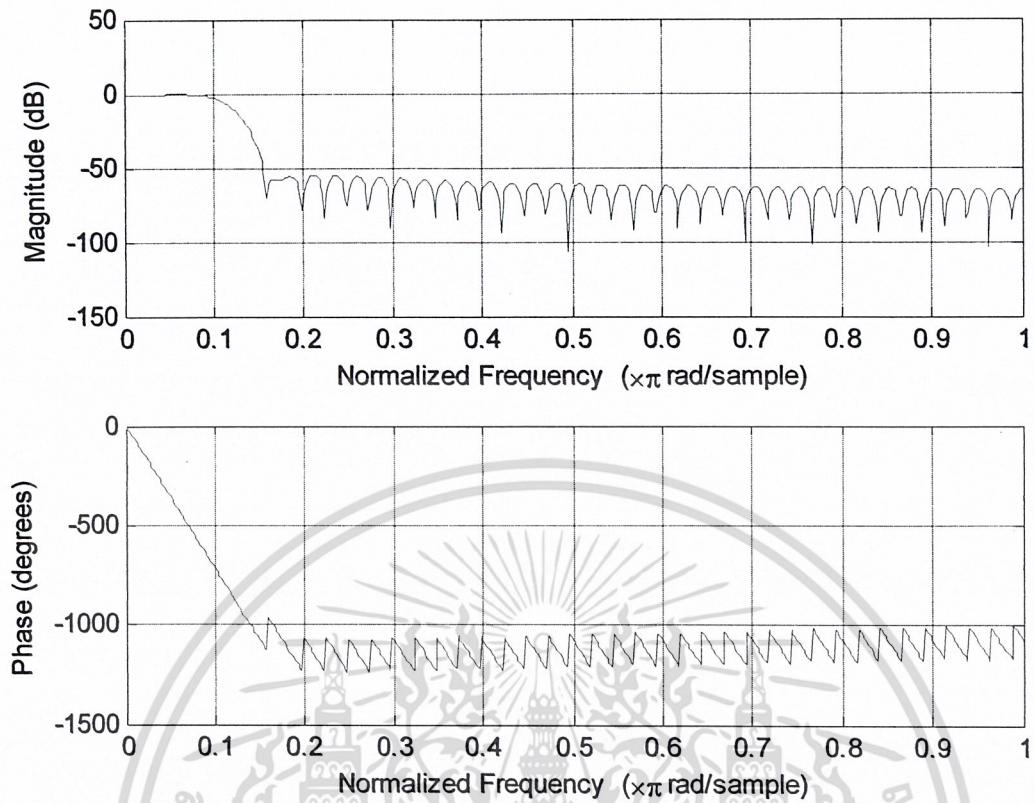
4.4.7 วงจรกรองความถี่ดิจิทัลแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 2425Hz ถึง 2820Hz

4.4.8 วงจรกรองความถี่ดิจิทัลแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 2820Hz ถึง 3215Hz

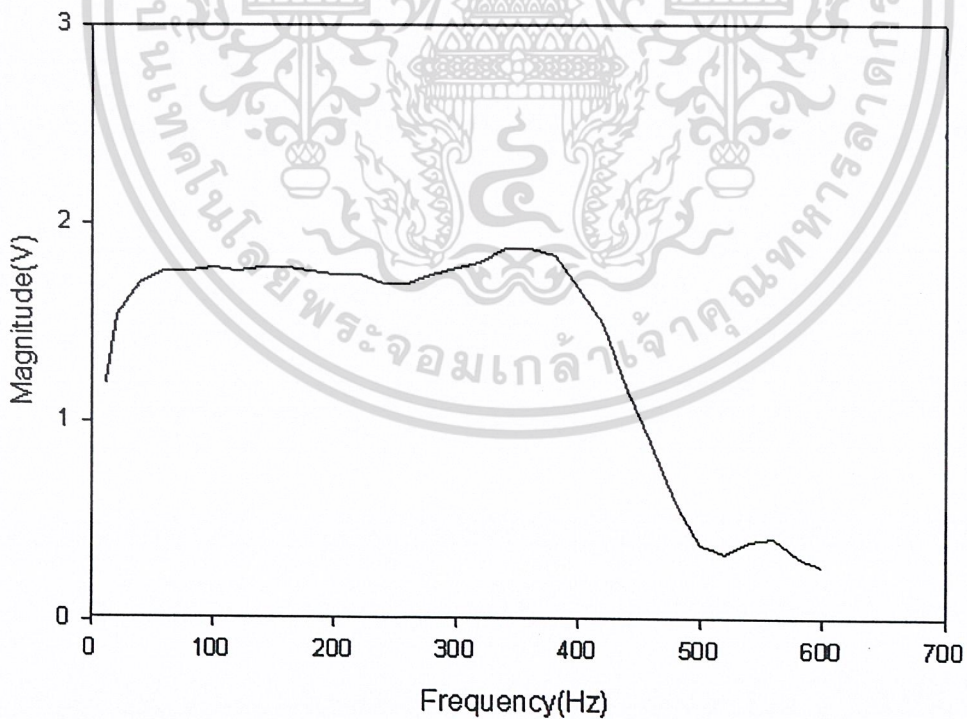
4.4.9 วงจรกรองความถี่ดิจิทัลแบบความถี่กลางผ่าน (Digital Bandpass Filter) ที่ช่วงความถี่ผ่านเท่ากับ 3215Hz ถึง 3610Hz

4.4.10 วงจรกรองความถี่ดิจิทัลแบบความถี่สูงผ่าน (Digital Highpass Filter) ที่ความถี่ผ่านเท่ากับ 3610Hz

ทุกตัวกรองความถี่ถูกสุ่มด้วยค่าสัมประสิทธิ์ที่สุ่มขึ้นและมีความถี่คutoff เท่ากับ 8kHz และมี Order เท่ากับ 81

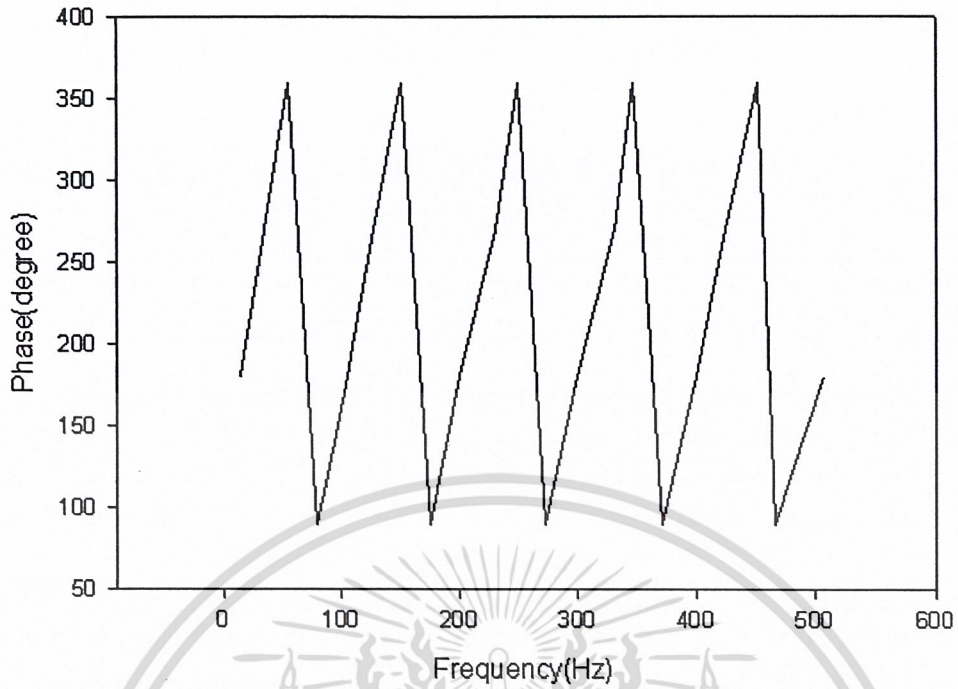


รูปที่ 4.9 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่ต่ำผ่าน ที่ได้จากการออกแบบ

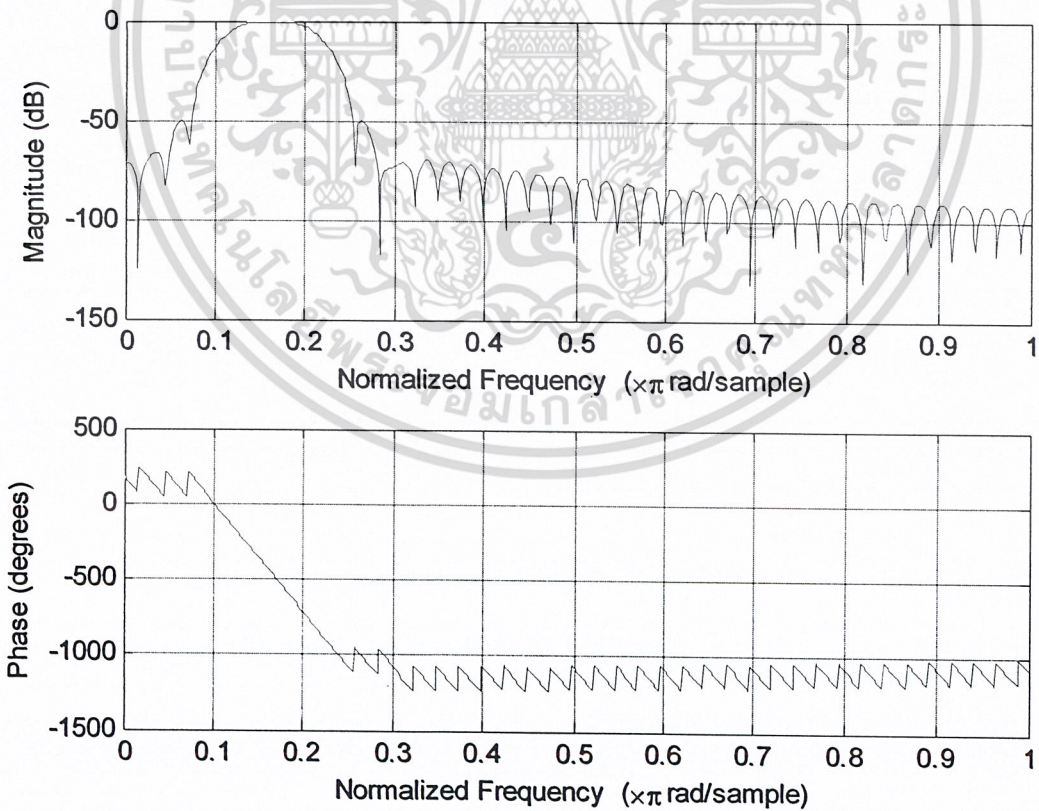


รูปที่ 4.10 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่ต่ำผ่าน ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

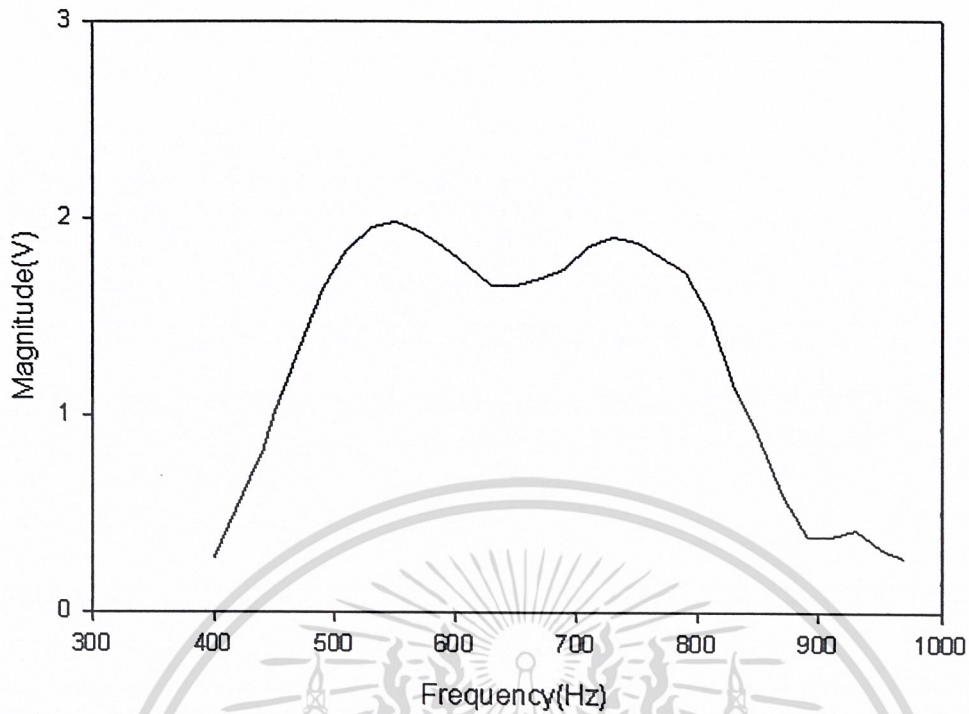


รูปที่ 4.11 ผลตอบสนองเฟสของตัวกรองความถี่ต่ำผ่าน ที่ได้จากการทดลอง

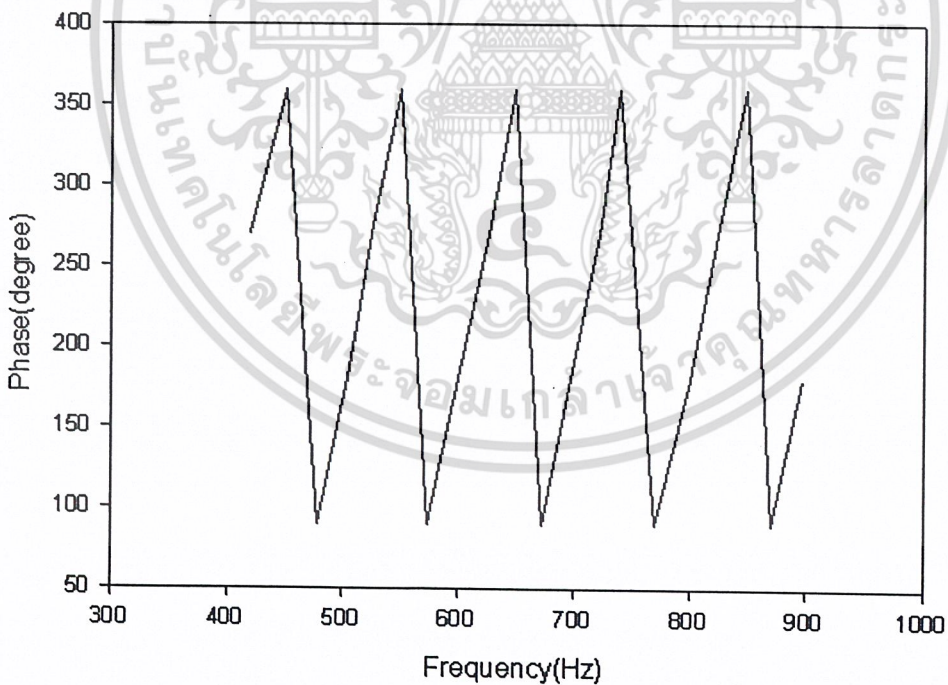


รูปที่ 4.12 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 450Hz ถึง 845Hz ที่ได้จากการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

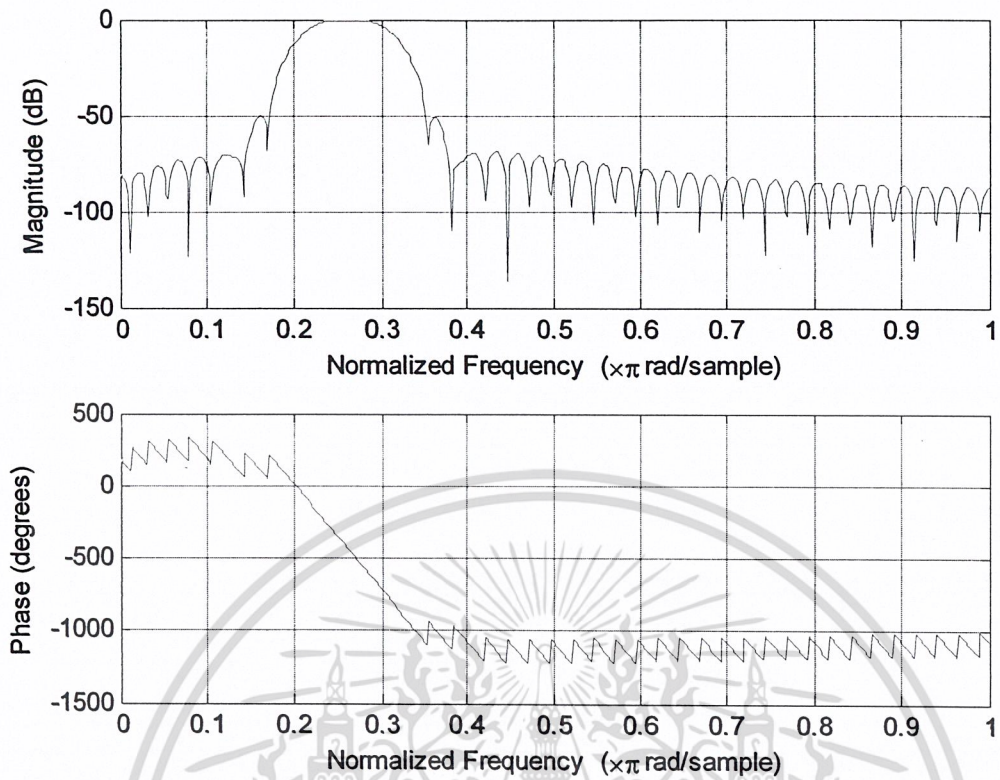


รูปที่ 4.13 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 450 Hz ถึง 845 Hz ที่ได้จากการทดลอง

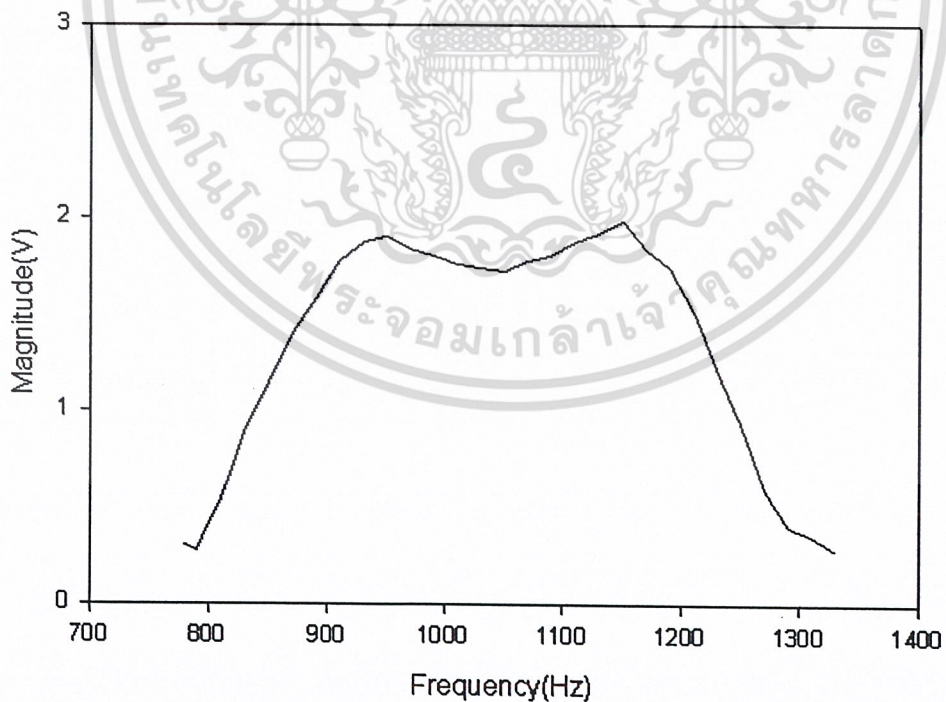


รูปที่ 4.14 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 450 Hz ถึง 845 Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

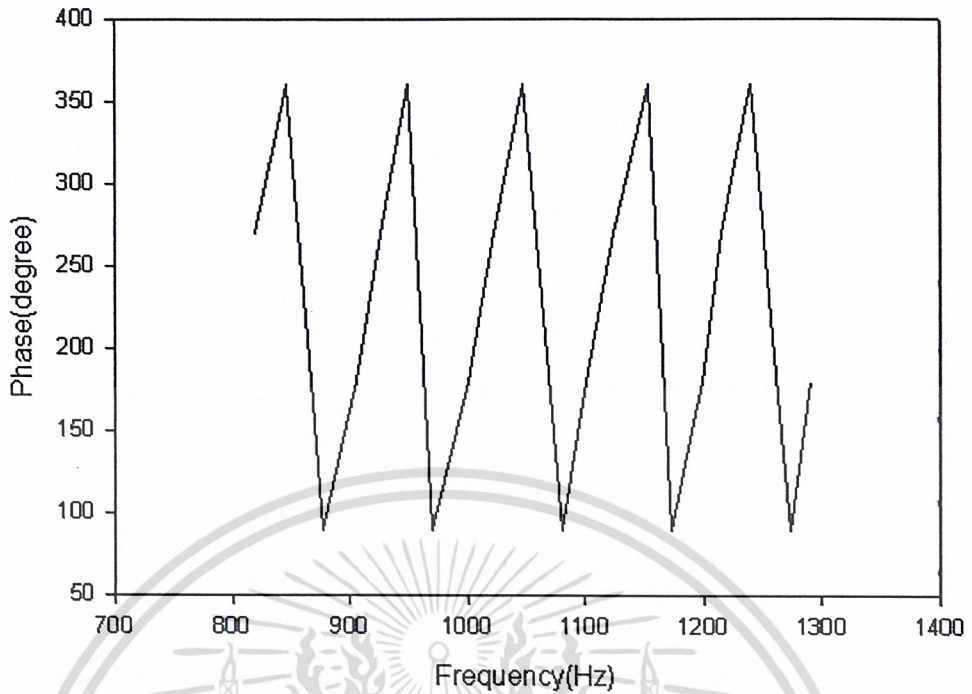


รูปที่ 4.15 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz ที่ได้จากการออกแบบ

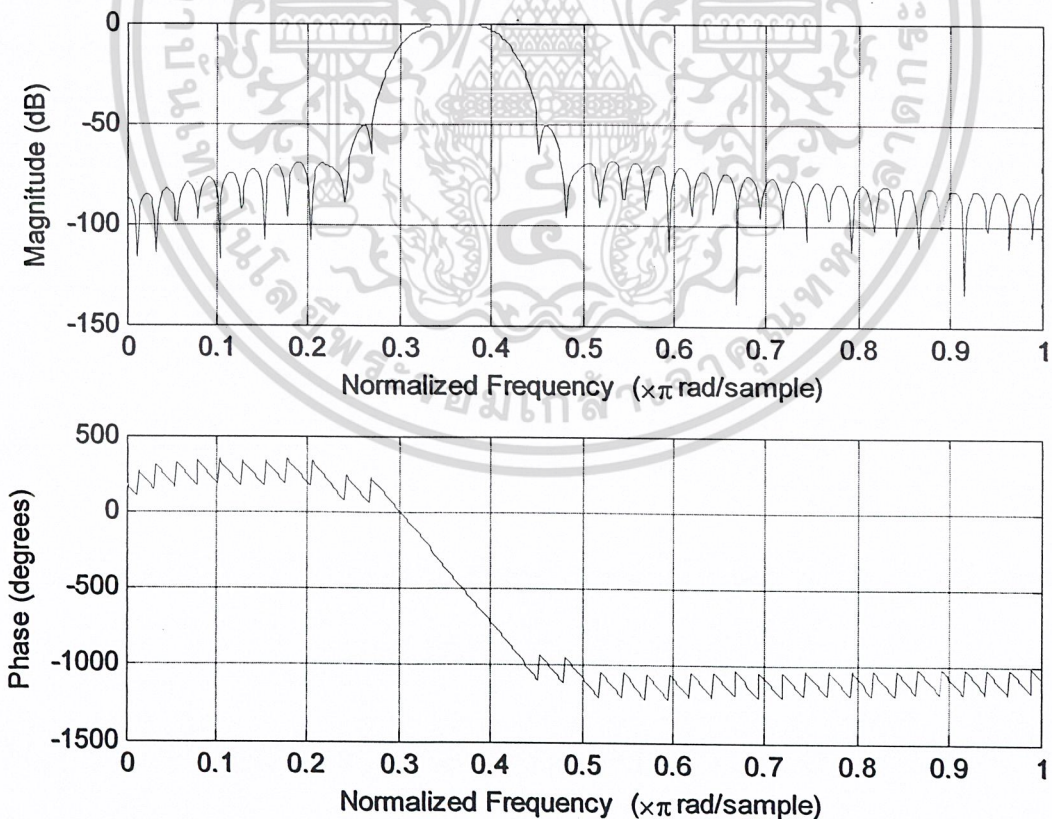


รูปที่ 4.16 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

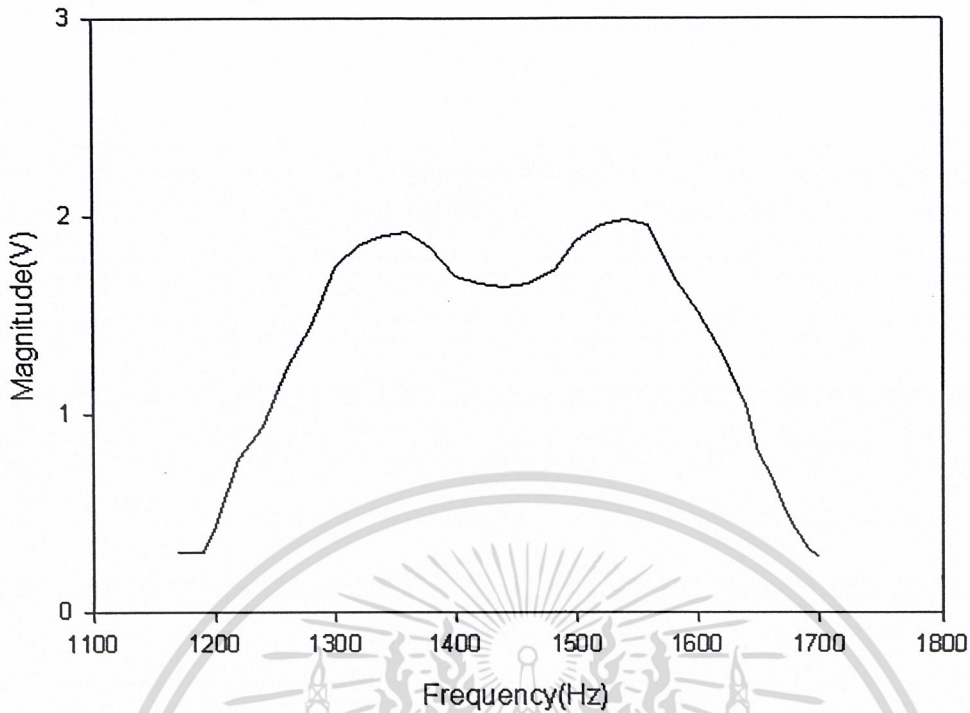


รูปที่ 4.17 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 845Hz ถึง 1240Hz ที่ได้จากการทดลอง

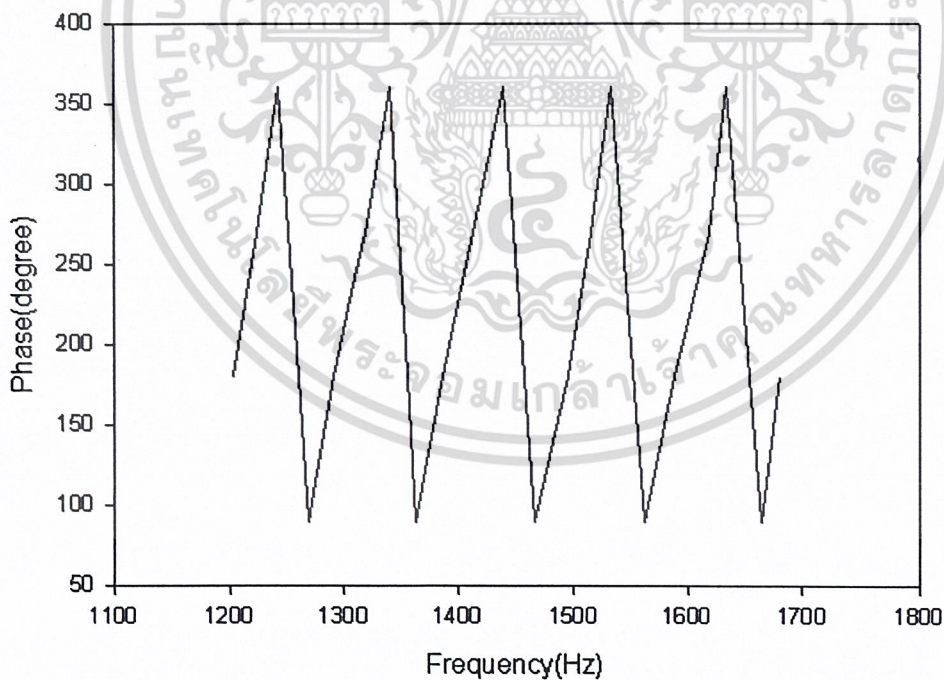


รูปที่ 4.18 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 1240Hz ถึง 1635Hz ที่ได้จากการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

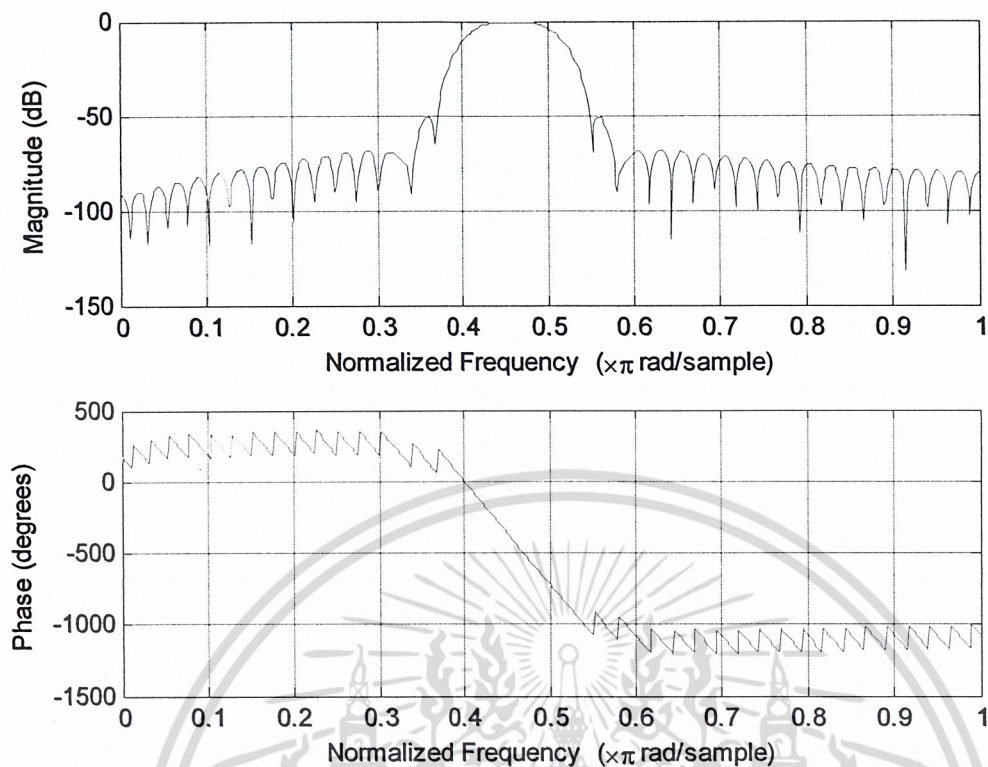


รูปที่ 4.19 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 1240 Hz ถึง 1635 Hz ที่ได้จากการทดลอง

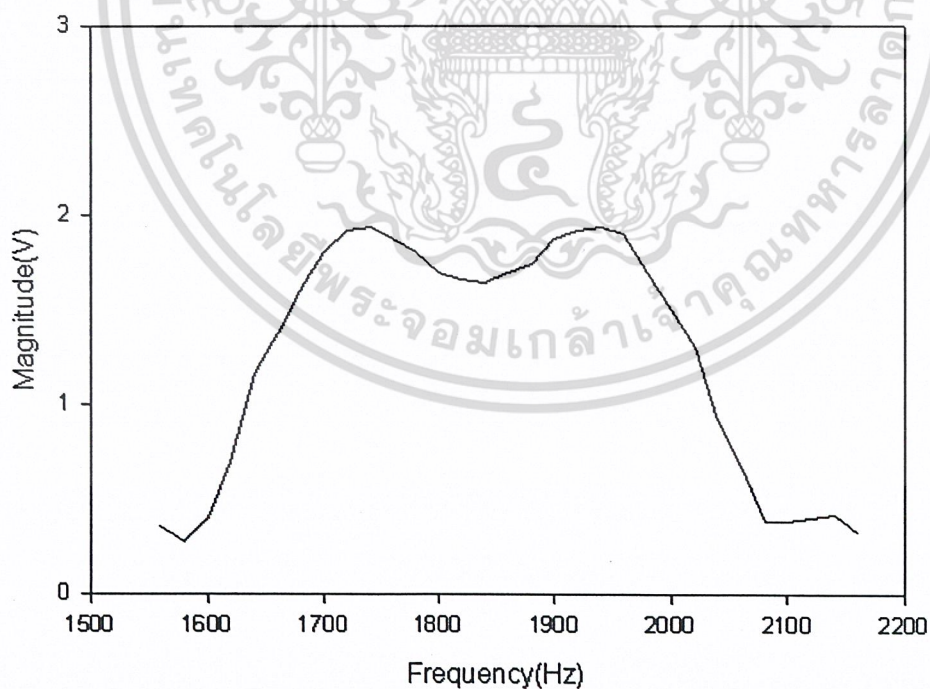


รูปที่ 4.20 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 1240 Hz ถึง 1635 Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

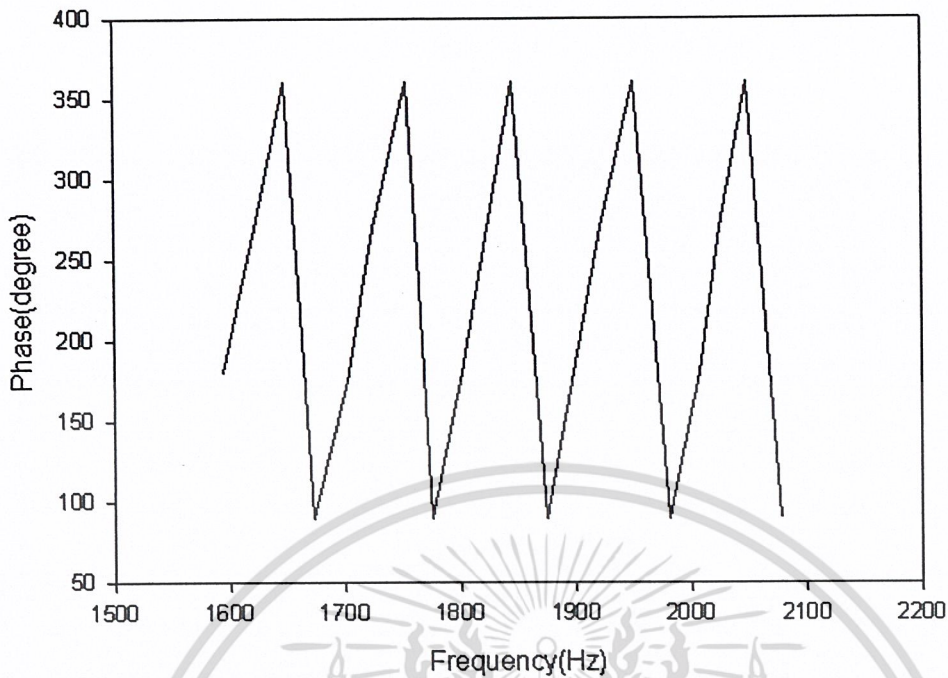


รูปที่ 4.21 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 1635Hz ถึง 2030Hz ที่ได้จากการออกแบบ

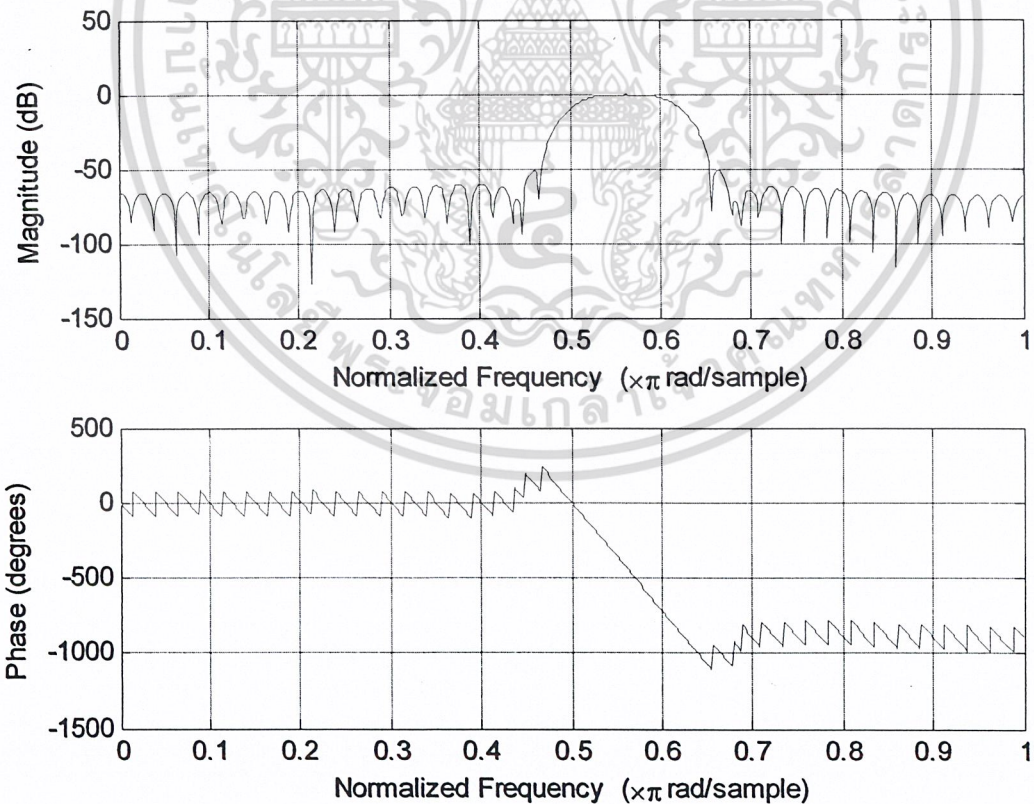


รูปที่ 4.22 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 1635Hz ถึง 2030Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

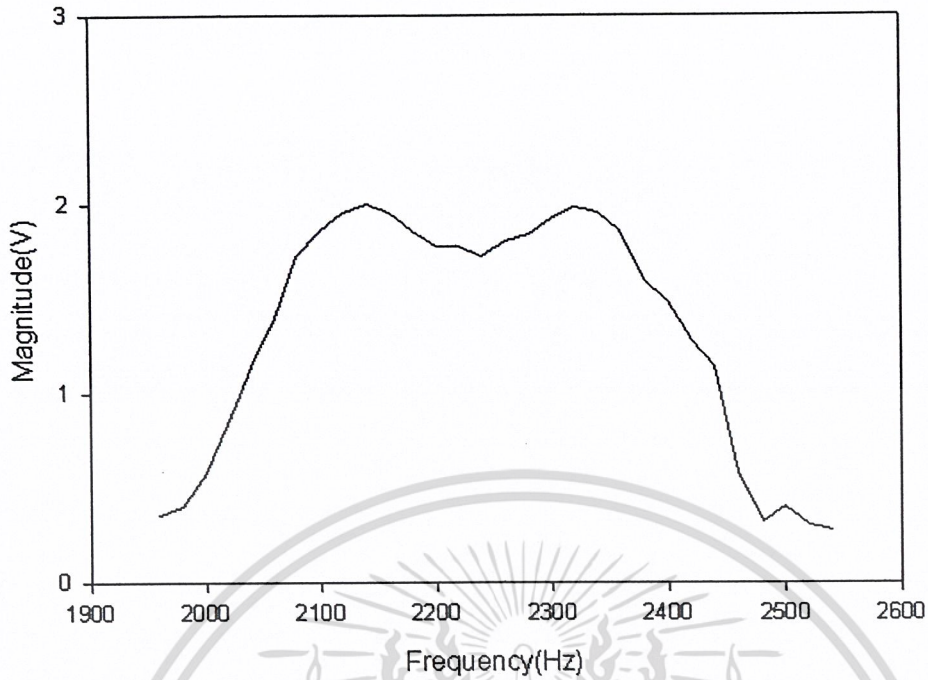


รูปที่ 4.23 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 1635 Hz ถึง 2030 Hz ที่ได้จากการทดลอง

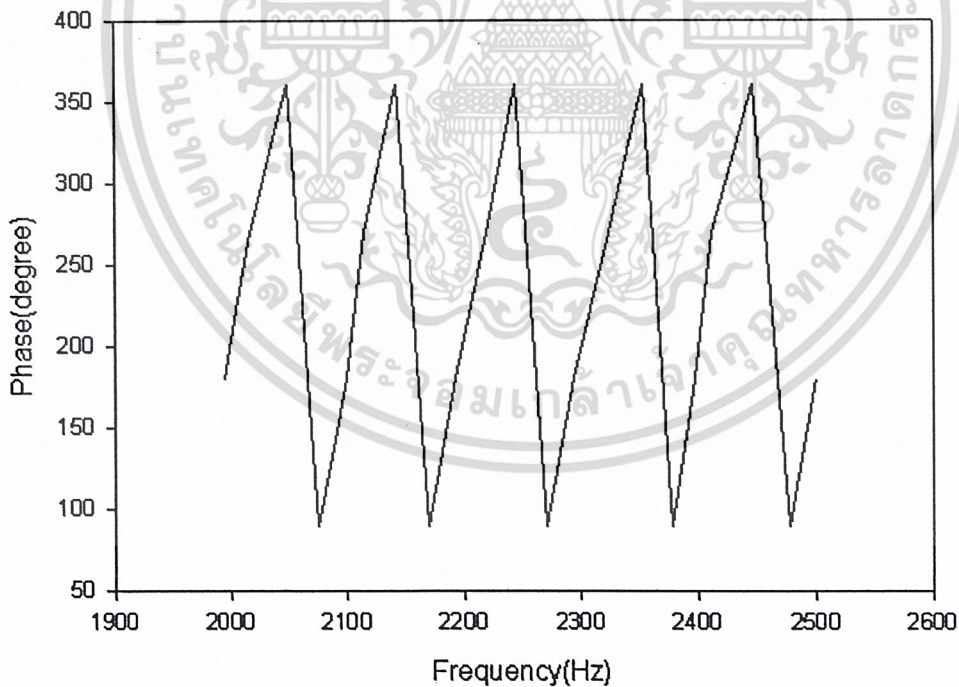


รูปที่ 4.24 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2030 Hz ถึง 2425 Hz ที่ได้จากการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

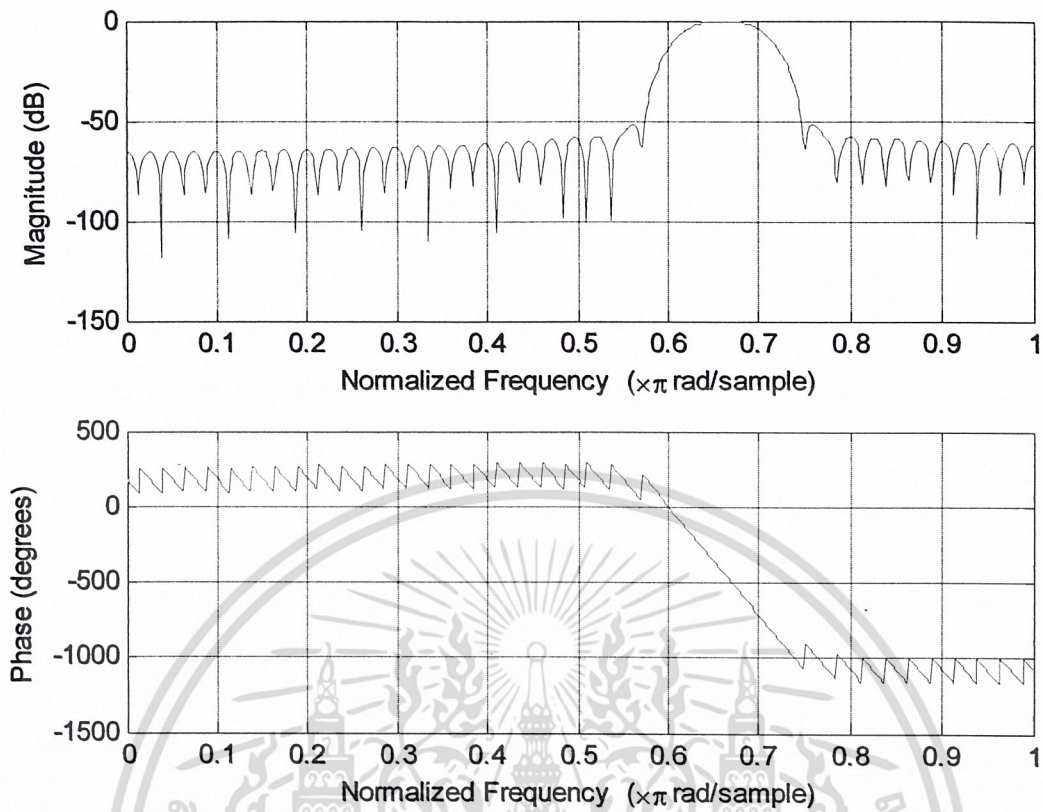


รูปที่ 4.25 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2030Hz ถึง 2425Hz ที่ได้จากการทดลอง

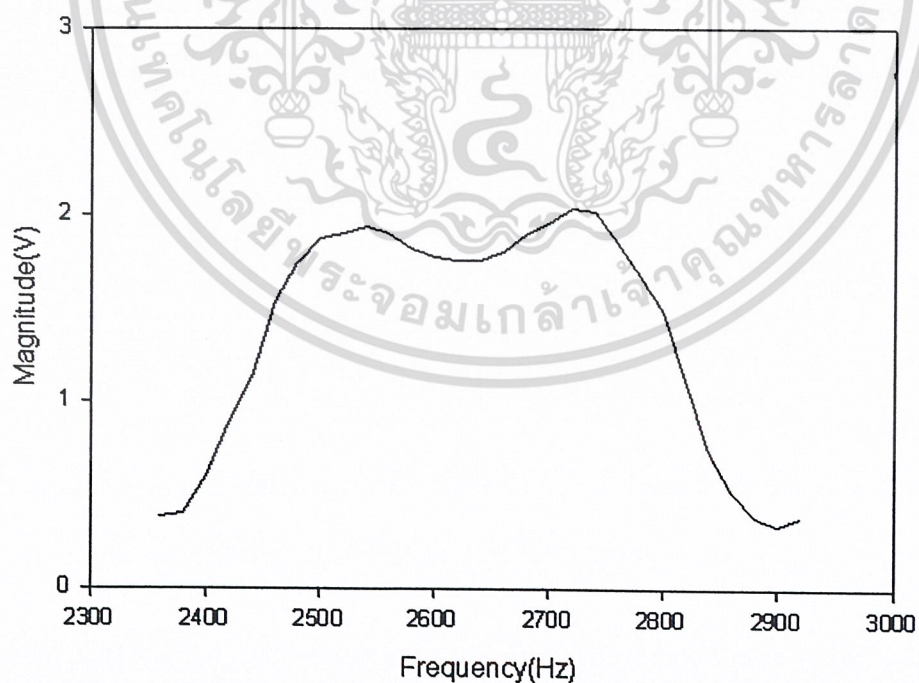


รูปที่ 4.26 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2030Hz ถึง 2425Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

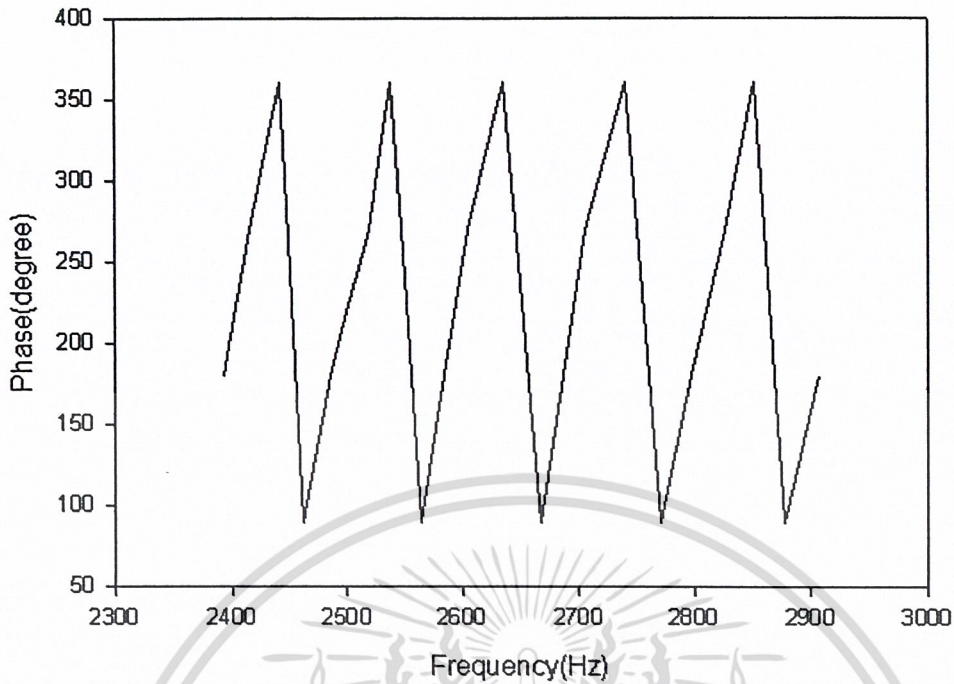


รูปที่ 4.27 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2425Hz ถึง 2820Hz ที่ได้จากการออกแบบ

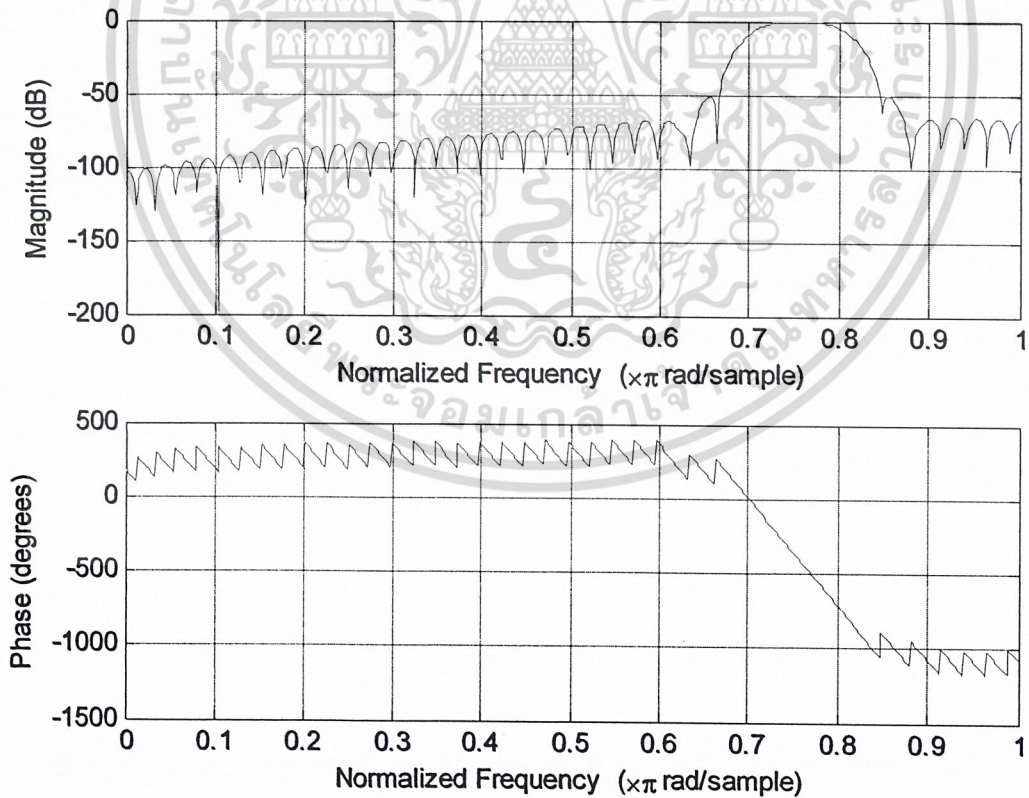


รูปที่ 4.28 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2425Hz ถึง 2820Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

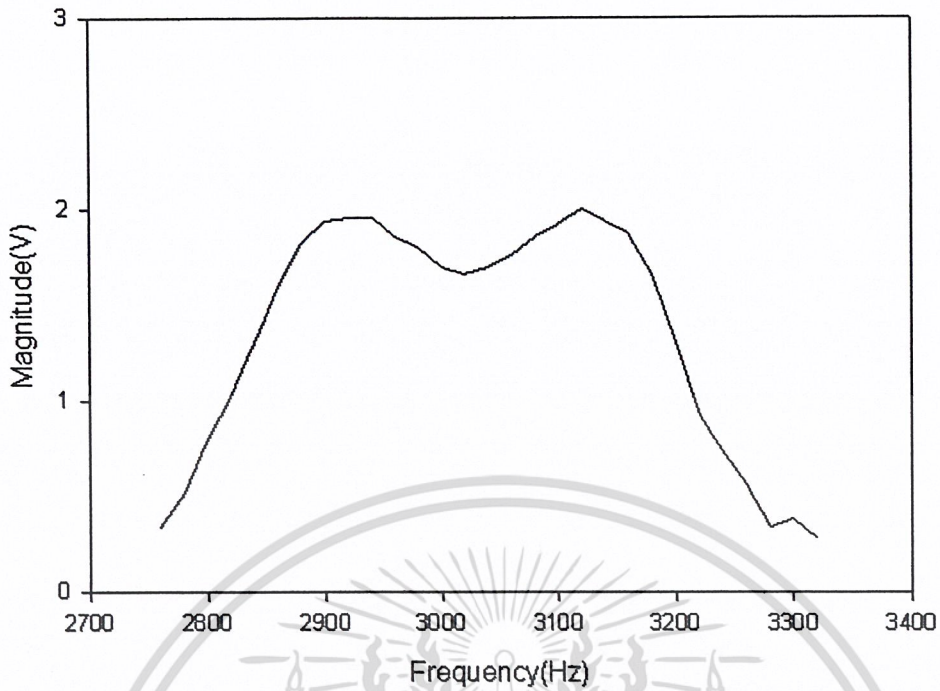


รูปที่ 4.29 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2425Hz ถึง 2820Hz ที่ได้จากการทดลอง

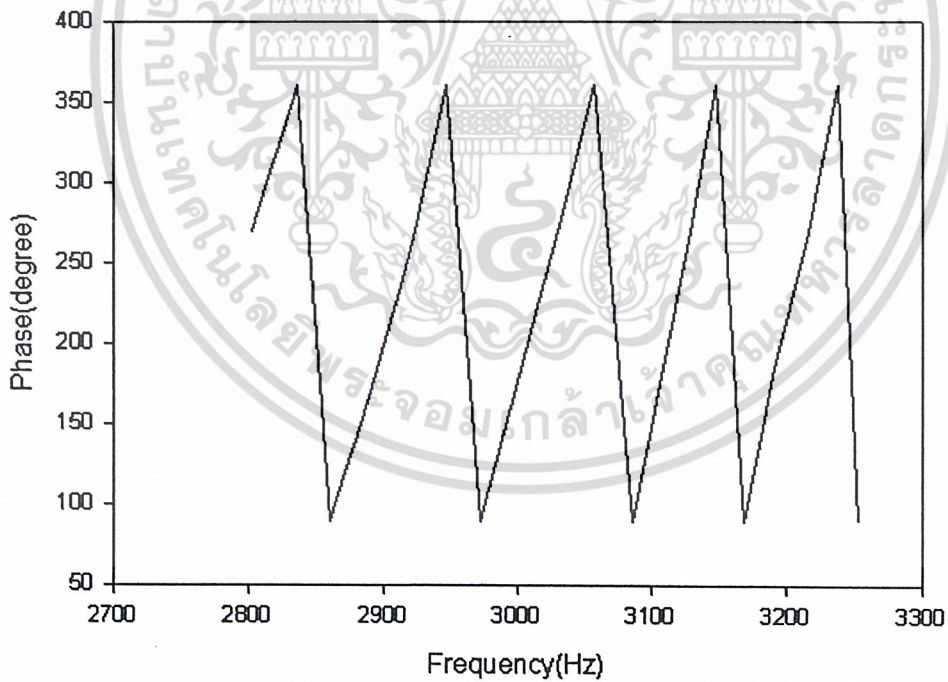


รูปที่ 4.30 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2820Hz ถึง 3215Hz ที่ได้จากการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

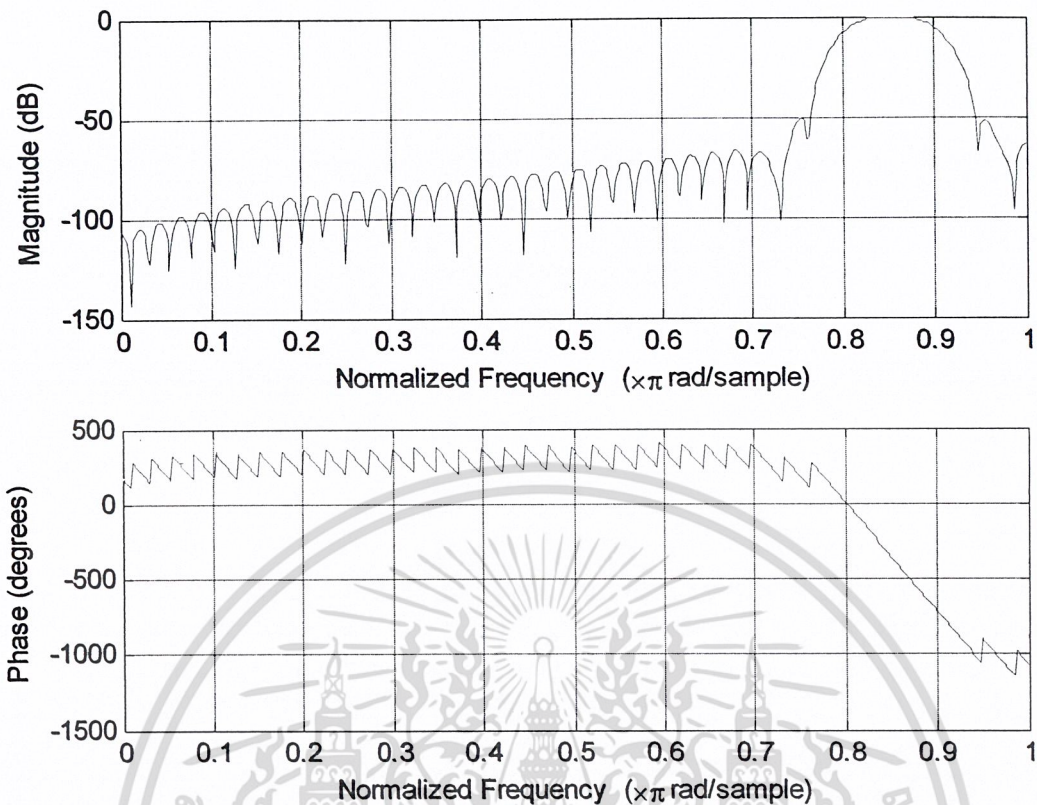


รูปที่ 4.31 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2820Hz ถึง 3215Hz ที่ได้จากการทดลอง

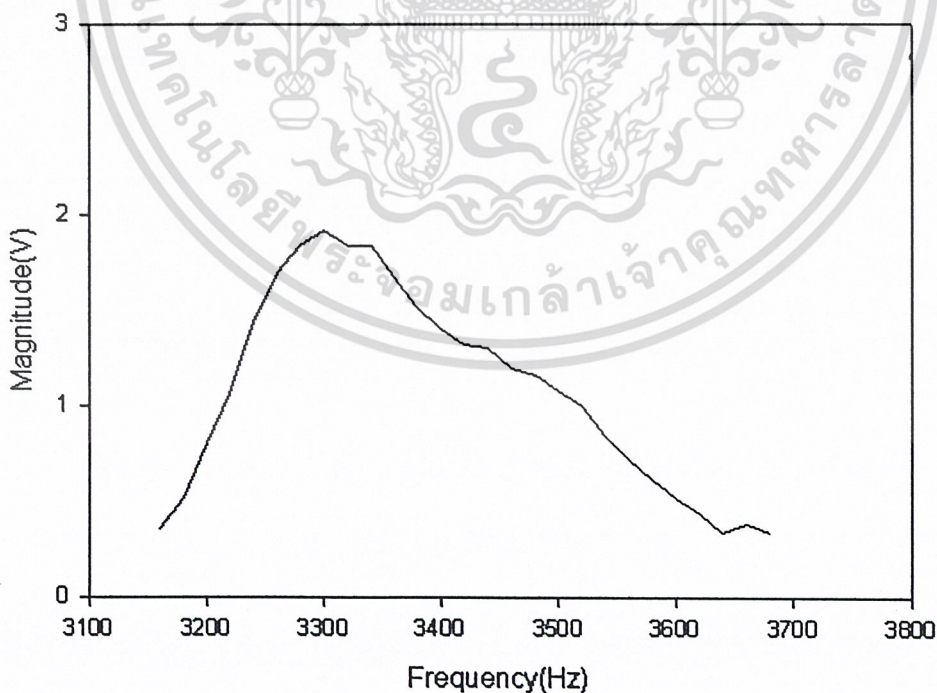


รูปที่ 4.32 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 2820Hz ถึง 3215Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

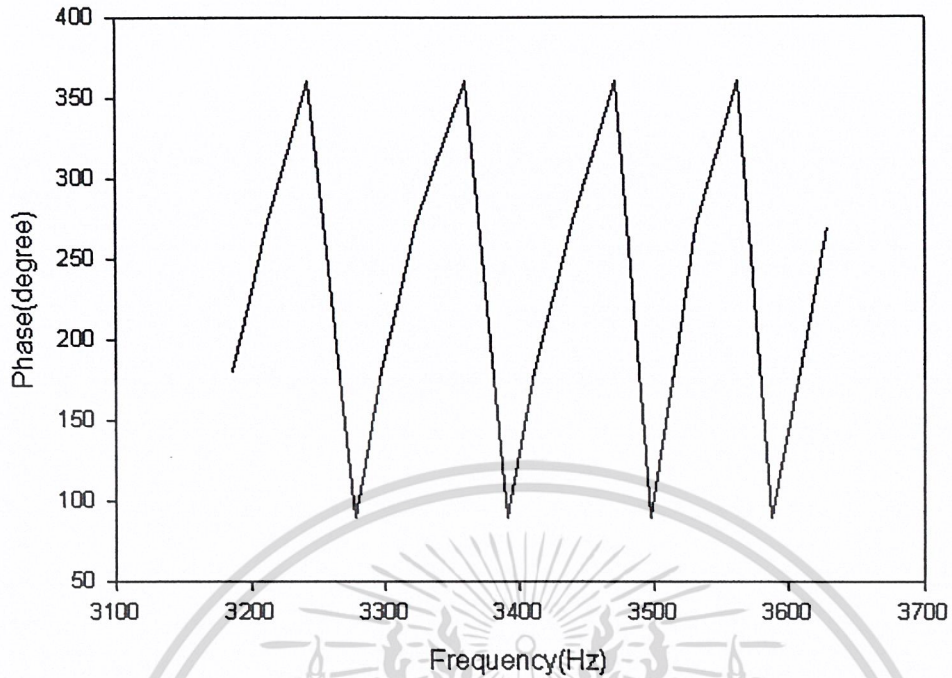


รูปที่ 4.33 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่านเท่ากับ 3215Hz ถึง 3610Hz ที่ได้จากการออกแบบ

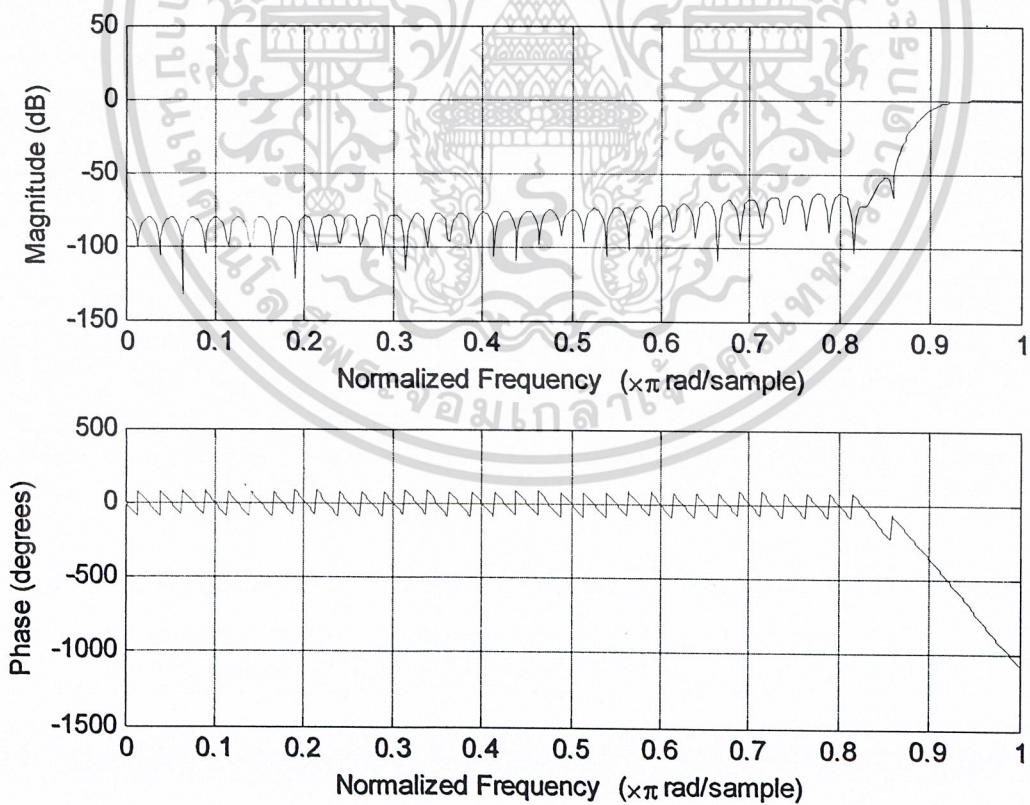


รูปที่ 4.34 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่านเท่ากับ 3215Hz ถึง 3610Hz ที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

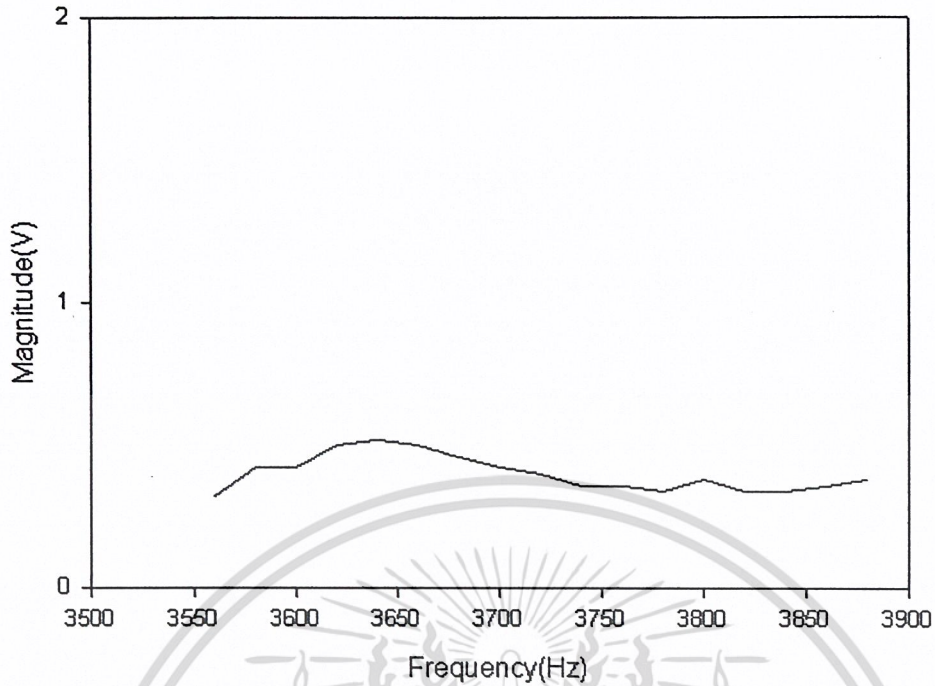


รูปที่ 4.35 ผลตอบสนองเฟสของตัวกรองความถี่กลางผ่านช่วงความถี่ผ่าน
เท่ากับ 3215 Hz ถึง 3610 Hz ที่ได้จากการทดลอง

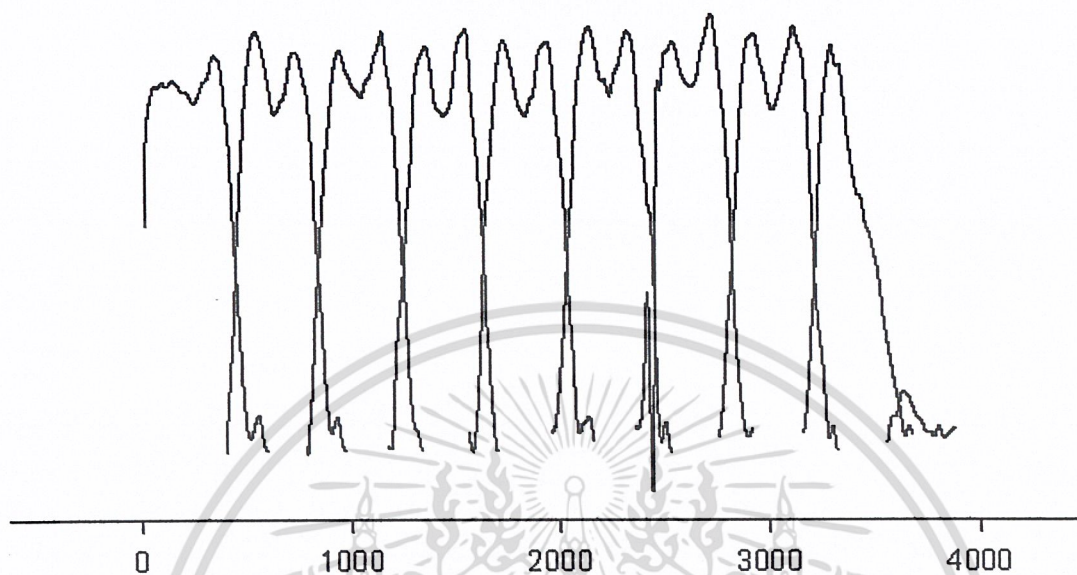


รูปที่ 4.36 ผลตอบสนองแอมพลิจูดและเฟสของตัวกรองความถี่สูงผ่านความถี่ผ่าน
เท่ากับ 3610 Hz ที่ได้จากการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.37 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่สูงผ่านความถี่ผ่านเท่ากับ 3610Hz ที่ได้จากการทดลอง



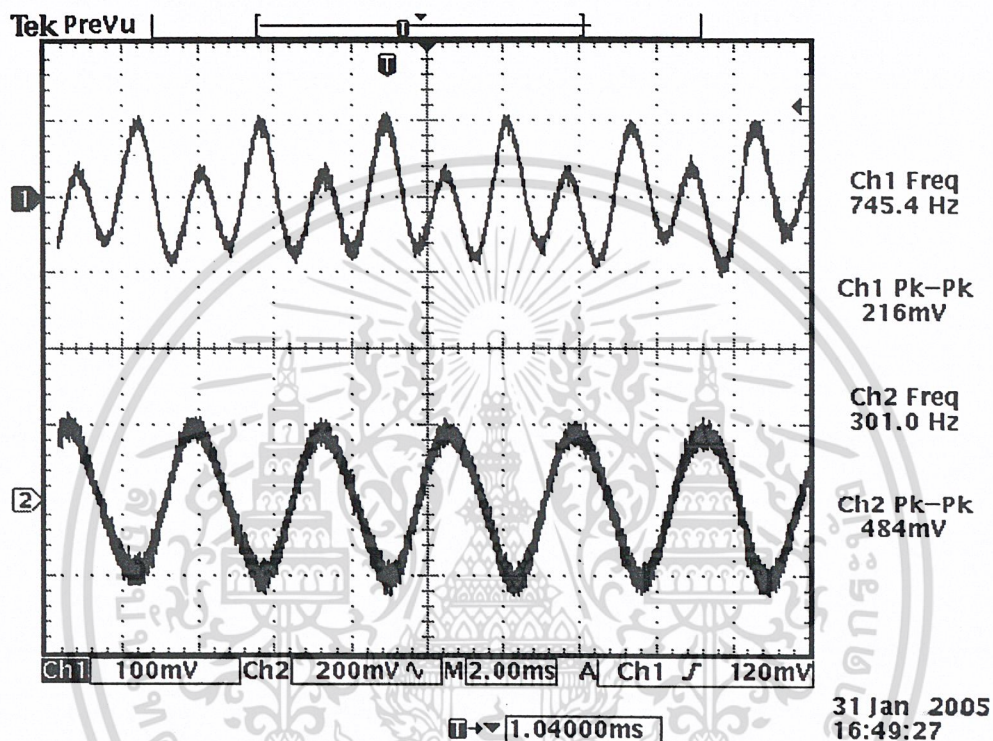
รูปที่ 4.38 ผลตอบสนองแอมพลิจูดของตัวกรองความถี่ต่ำ 10 ตัวรวมกัน

จากการทดลองนี้จะเห็นว่า ผลการตอบสนองของเฟสจะมีลักษณะเป็นเชิงเส้น (Linear) ซึ่งตรงตามการออกแบบ

4.5 การทดลองที่ 5 ลักษณะของสัญญาณไซน์ที่ได้หลังจากผ่านตัวกรองความถี่

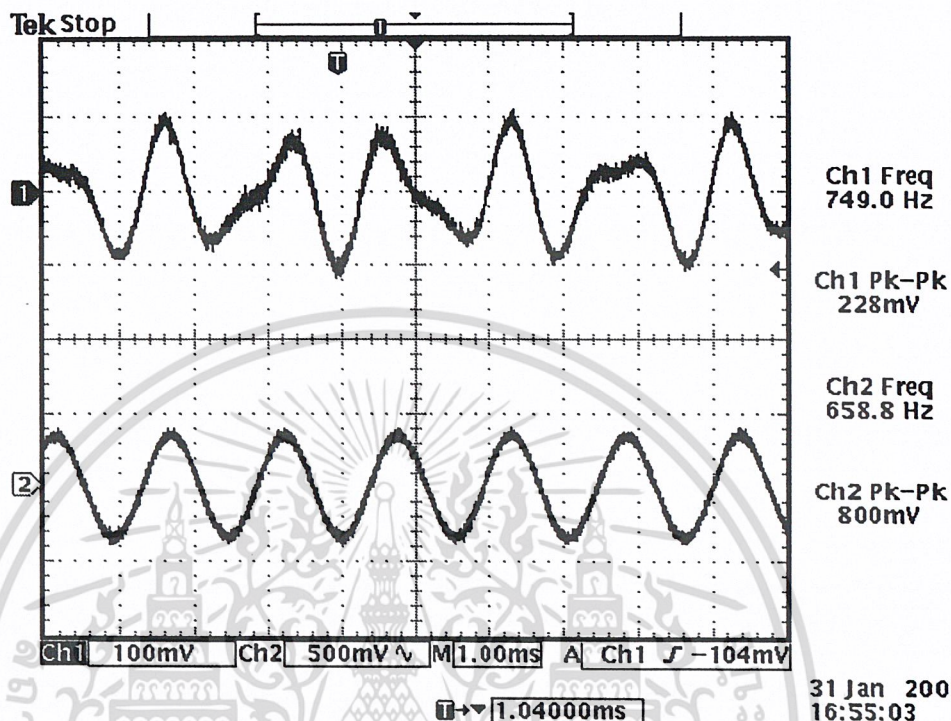
พิจารณาลักษณะของสัญญาณที่ออกมาจากตัวกรองความถี่ ว่ามีลักษณะของสัญญาณถูกต้องตรงกับที่ป้อนเข้าไปหรือไม่ โดยการทดลองนี้แบ่งออก ดังนี้

4.5.1 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่ต่ำผ่านที่ความถี่ผ่านเท่ากับ 450Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 300Hz และ 650Hz รวมกัน



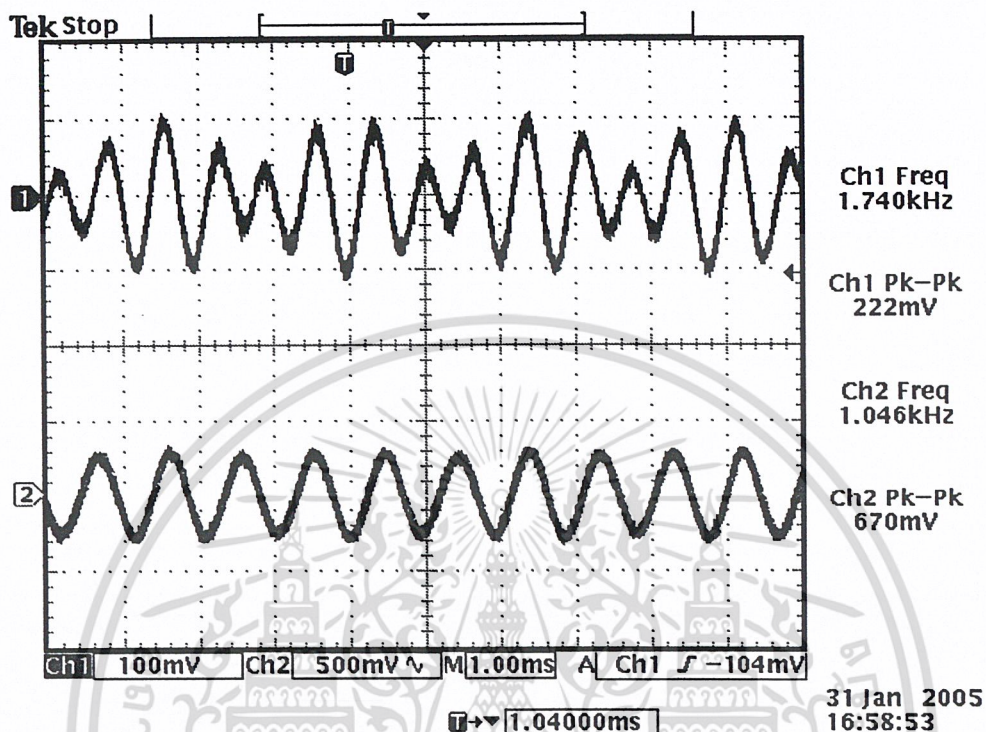
รูปที่ 4.39 Channel#1 คือสัญญาณ ไซน์ที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณหลังผ่านตัวกรองความถี่ต่ำผ่าน

4.5.2 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 450Hz ถึง 845Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 650Hz และ 1050Hz รวมกัน



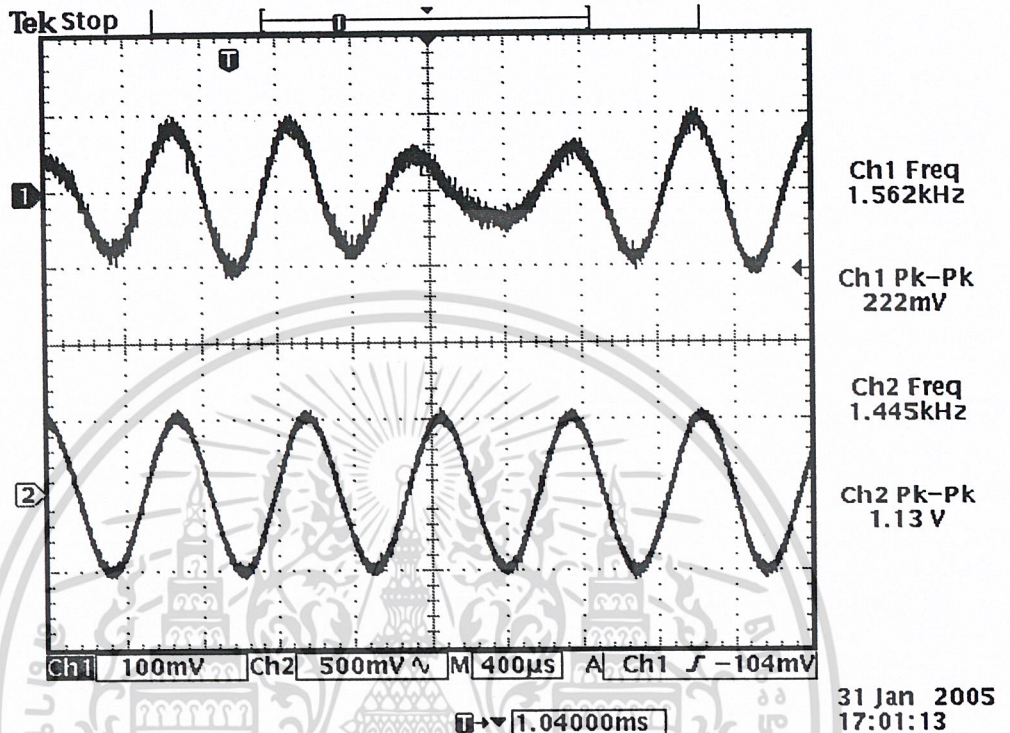
รูปที่ 4.40 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 450Hz ถึง 845Hz

4.5.3 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 1050Hz และ 1450Hz รวมกัน



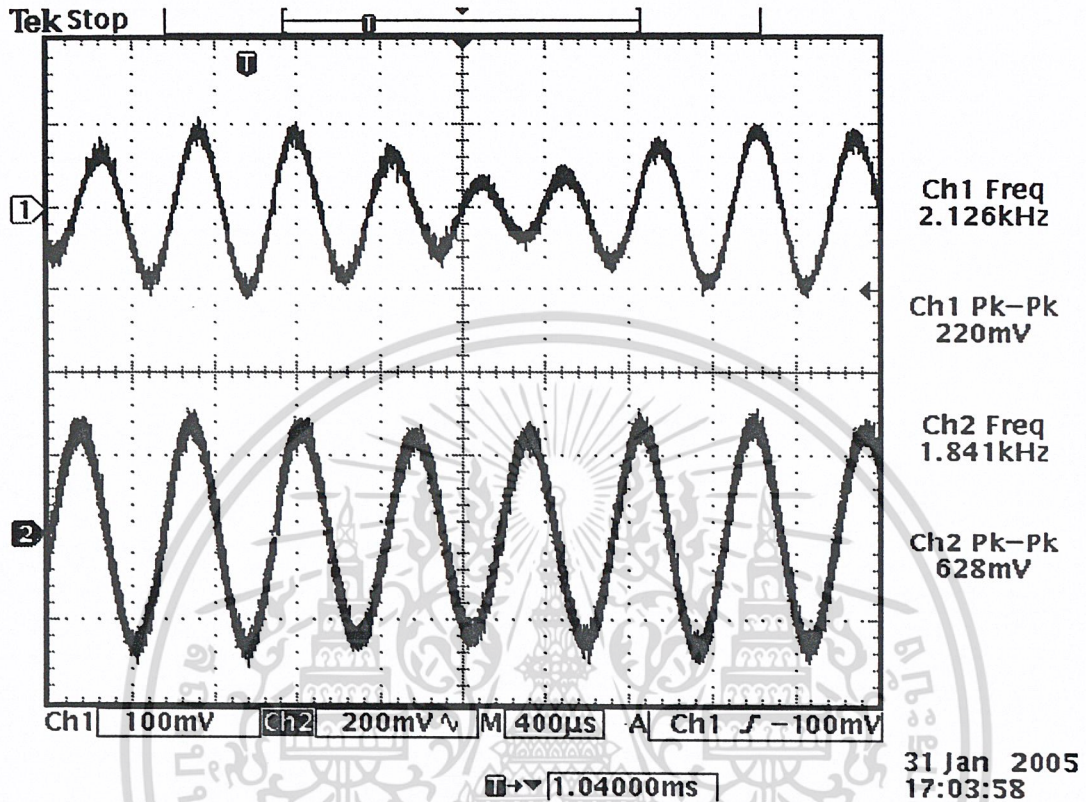
รูปที่ 4.41 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางที่ช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz

4.5.4 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 1240Hz ถึง 1635Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 1450Hz และ 1835Hz รวมกัน



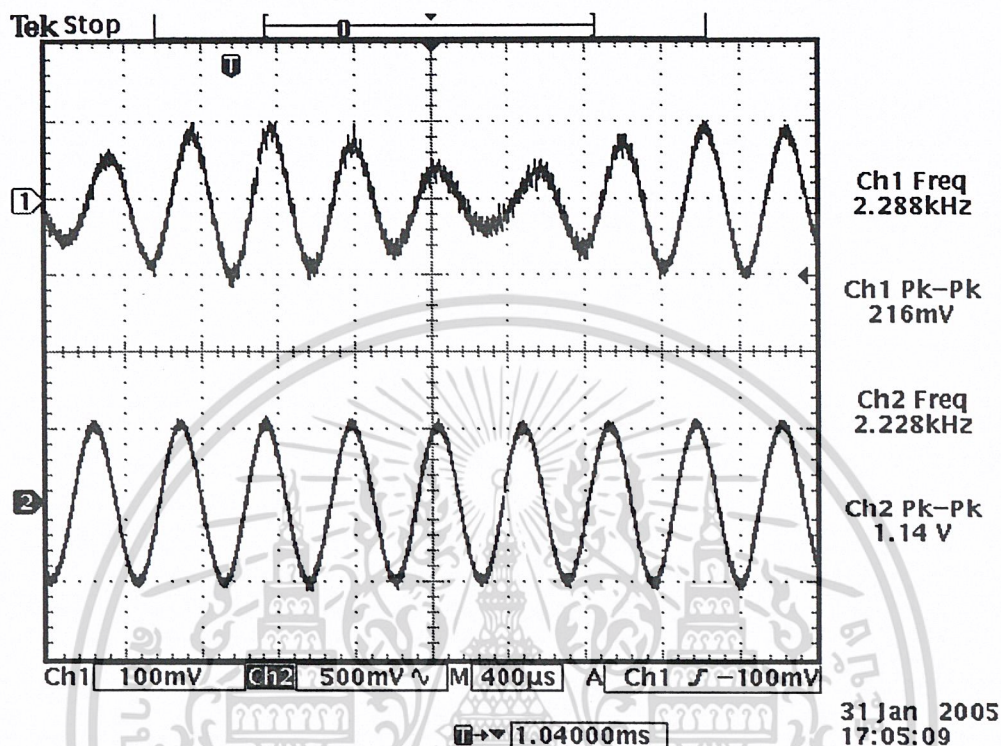
รูปที่ 4.42 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 1240Hz ถึง 1635Hz

4.5.5 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 1635 Hz ถึง 2030 Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ คือ สัญญาณไซน์ความถี่ 1835 Hz และ 2230 Hz รวมกัน



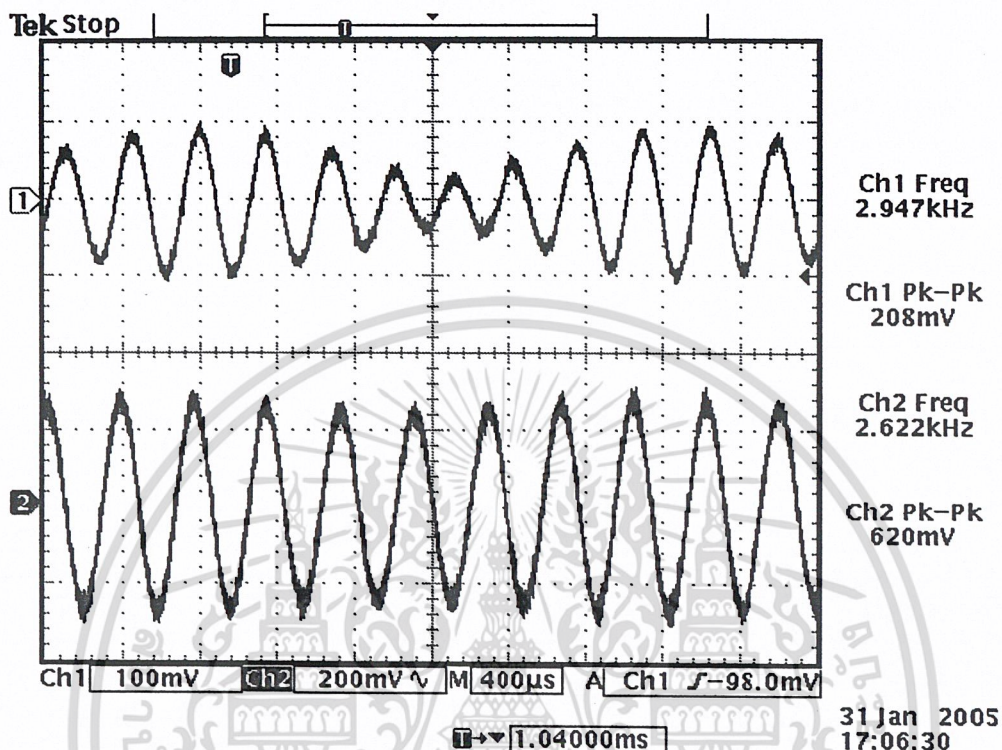
รูปที่ 4.43 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 1635 Hz ถึง 2030 Hz

4.5.6 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 2030Hz ถึง 2425Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 2230Hz และ 2630Hz รวมกัน



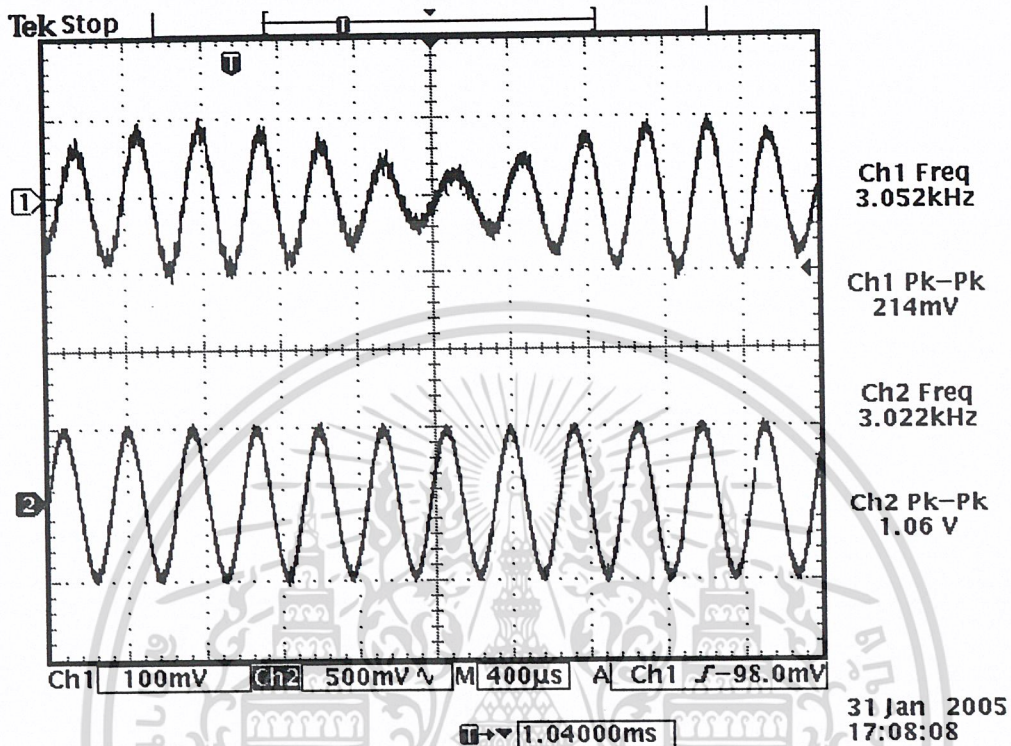
รูปที่ 4.44 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 2030Hz ถึง 2425Hz

4.5.7 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 450Hz ถึง 2425Hz ถึง 2820Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 2630Hz และ 3020Hz รวมกัน



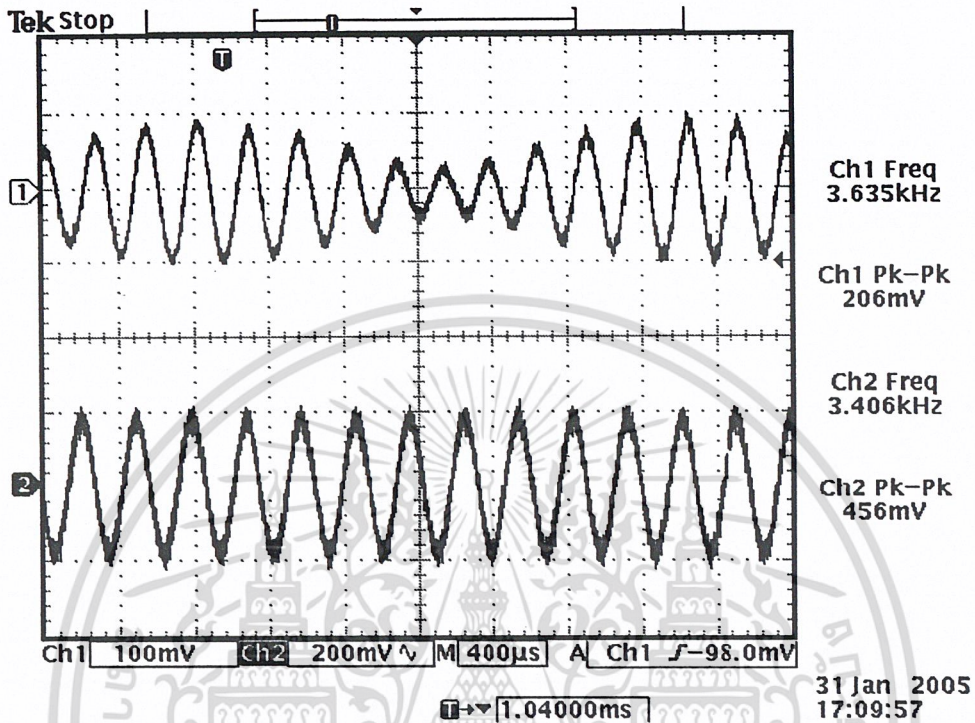
รูปที่ 4.45 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 2425Hz ถึง 2820Hz

4.5.8 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 2820Hz ถึง 3215Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 3020Hz และ 3420Hz รวมกัน



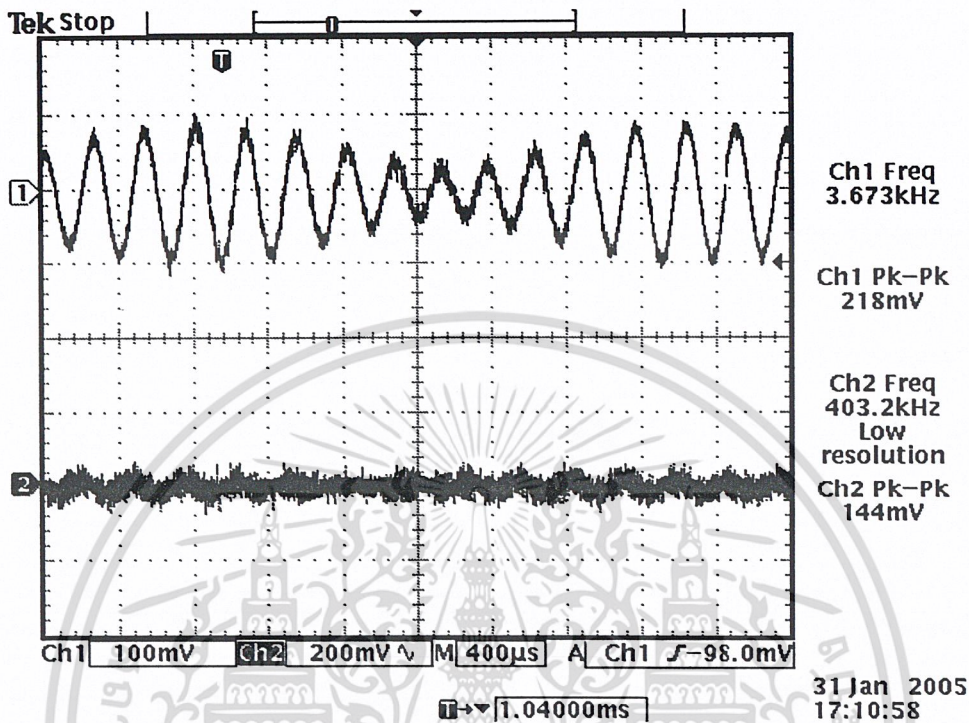
รูปที่ 4.46 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 2820Hz ถึง 3215Hz

4.5.9 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 3215Hz ถึง 3610Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 3420Hz และ 3800Hz รวมกัน



รูปที่ 4.47 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณ ไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 3215Hz ถึง 3610Hz

4.5.10 ทำการป้อนสัญญาณรูปไซน์เข้าสู่ตัวกรองความถี่สูงผ่านที่ความถี่ผ่านเท่ากับ 3610Hz โดยสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆคือ สัญญาณไซน์ความถี่ 3420Hz และ 3800Hz รวมกัน



รูปที่ 4.48 Channel#1 คือสัญญาณที่ป้อนเป็นสัญญาณที่ได้มาจากการรวมสัญญาณไซน์ความถี่ต่างๆ Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่สูงผ่านที่ความถี่ผ่านเท่ากับ 3610Hz

จากการทดลองหัวข้อ 4.5.1 พบว่าสัญญาณที่ได้หลังจากผ่านตัวกรองความถี่ต่ำผ่าน 450Hz เป็นสัญญาณไซน์ที่ความถี่ 300Hz และเช่นเดียวกันในจากการทดลองหัวข้อ 4.5.2 – 4.2.9 พบว่า

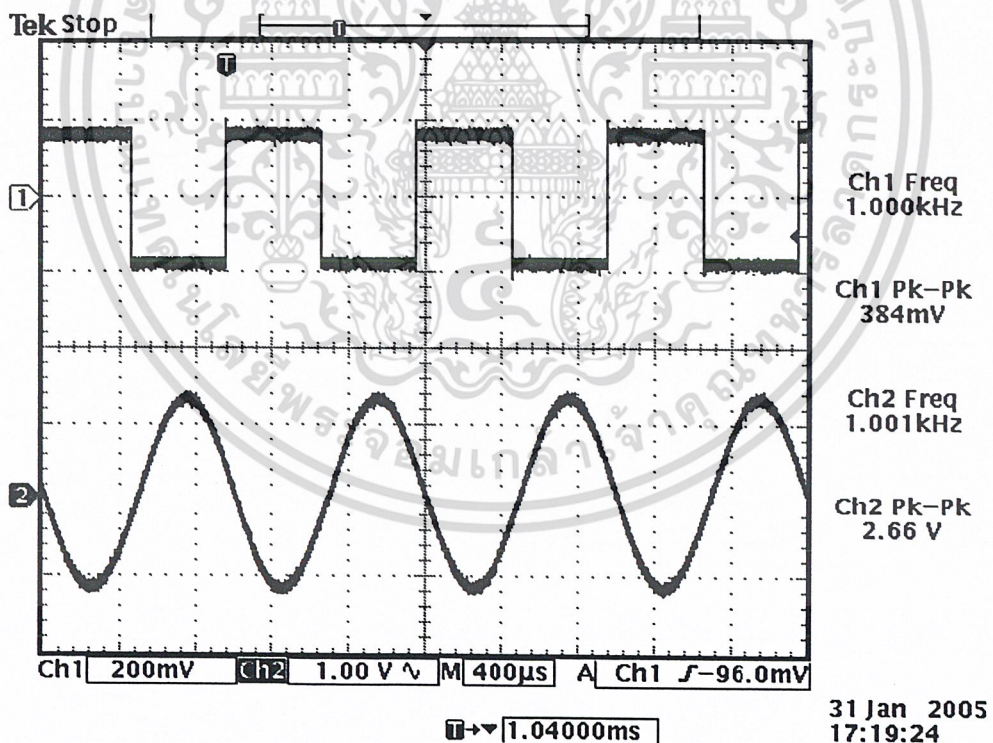
- สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 450Hz ถึง 845Hz เป็นสัญญาณไซน์ที่ความถี่ 650Hz
- สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz เป็นสัญญาณไซน์ที่ความถี่ 1050Hz
- สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 1240Hz ถึง 1635Hz เป็นสัญญาณไซน์ที่ความถี่ 1450Hz
- สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 1635Hz ถึง 2030Hz เป็นสัญญาณไซน์ที่ความถี่ 1835Hz
- สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านในช่วงความถี่ผ่านเท่ากับ 2030Hz ถึง 2425Hz เป็นสัญญาณไซน์ที่ความถี่ 2230Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

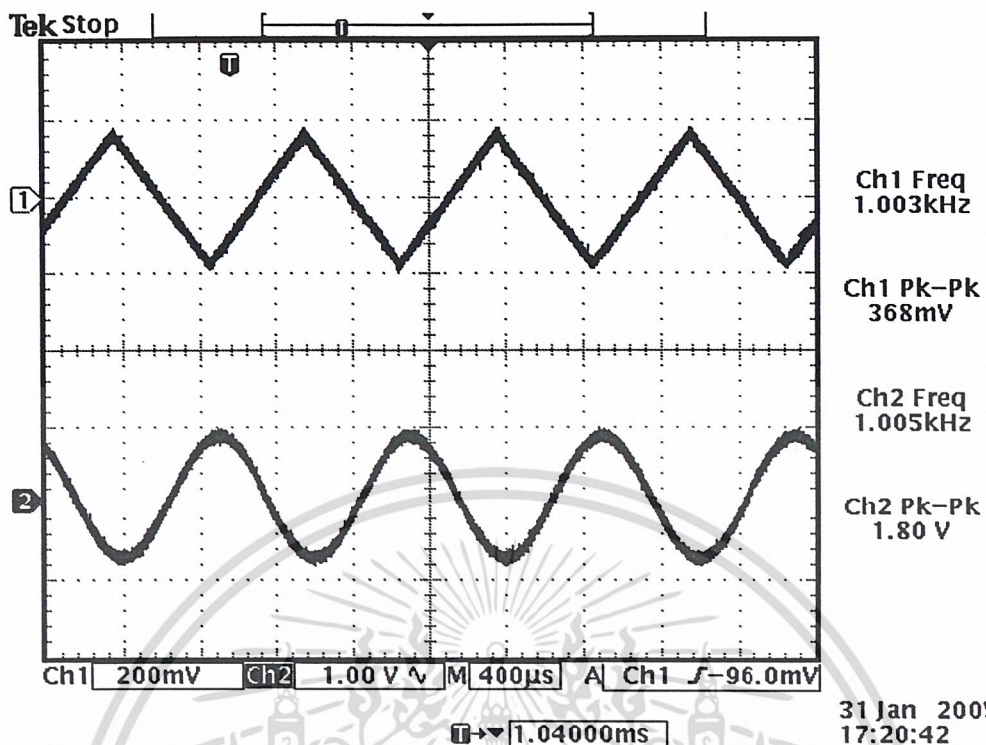
- สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 2425Hz ถึง 2820Hz เป็นสัญญาณไซน์ที่ความถี่ 2630Hz
 - สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 2820Hz ถึง 3215Hz เป็นสัญญาณไซน์ที่ความถี่ 3020Hz
 - สัญญาณที่ได้หลังจากผ่านตัวกรองความถี่กลางผ่านที่ช่วงความถี่ผ่านเท่ากับ 3215Hz ถึง 3610Hz เป็นสัญญาณไซน์ที่ความถี่ 3420Hz
- แสดงให้เห็นว่าตัวกรองความถี่ทุกตัวทำหน้าที่ตัดความถี่ที่ไม่ผ่านออกได้อย่างถูกต้องยกเว้นตัวกรองความถี่สูงผ่านที่ความถี่ผ่านเท่ากับ 3610Hz ที่ไม่มีสัญญาณผ่านเนื่องมาจากตัวโคเคคของบอร์ดจำกัดความถี่ไว้

4.6 การทดลองที่ 6 ลักษณะของสัญญาณรูปคลื่นต่างๆที่ได้หลังจากผ่านตัวกรองความถี่

พิจารณาลักษณะของสัญญาณที่ออกมาจากตัวกรองความถี่ ว่ามีลักษณะของสัญญาณถูกต้องตรงกับที่ป้อนเข้าไปหรือไม่ โดยการทดลองเราจะทำการป้อนสัญญาณรูปคลื่นต่างๆเข้าสู่ตัวกรองความถี่กลางผ่านช่วงความถี่ผ่านเท่ากับ 845Hz ถึง 1240Hz สัญญาณที่ป้อนจะประกอบไปด้วย สัญญาณคลื่นรูปสี่เหลี่ยมและสัญญาณคลื่นรูปสามเหลี่ยม



รูปที่ 4.49 Channel#1 คือสัญญาณอินพุตเป็นสัญญาณคลื่นรูปสี่เหลี่ยม
Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่าน

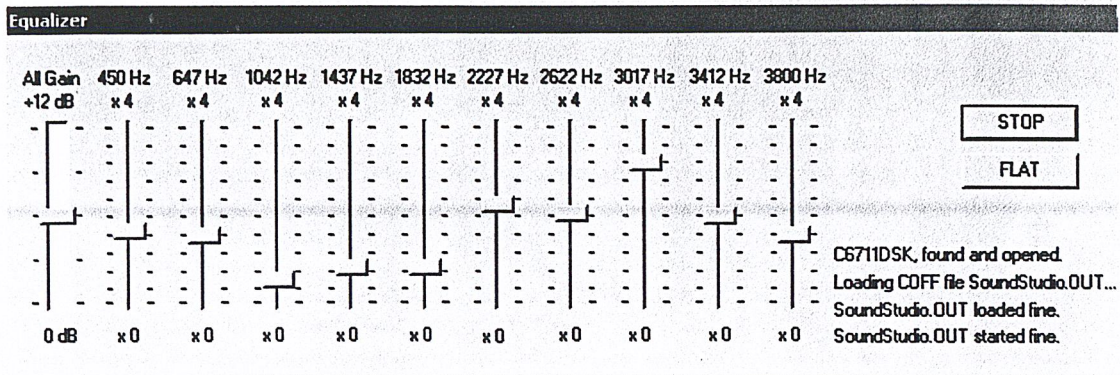


รูปที่ 4.50 Channel#1 คือสัญญาณอินพุตเป็นสัญญาณคลื่นรูปสามเหลี่ยม
Channel#2 คือสัญญาณที่ได้จากตัวกรองความถี่กลางผ่าน

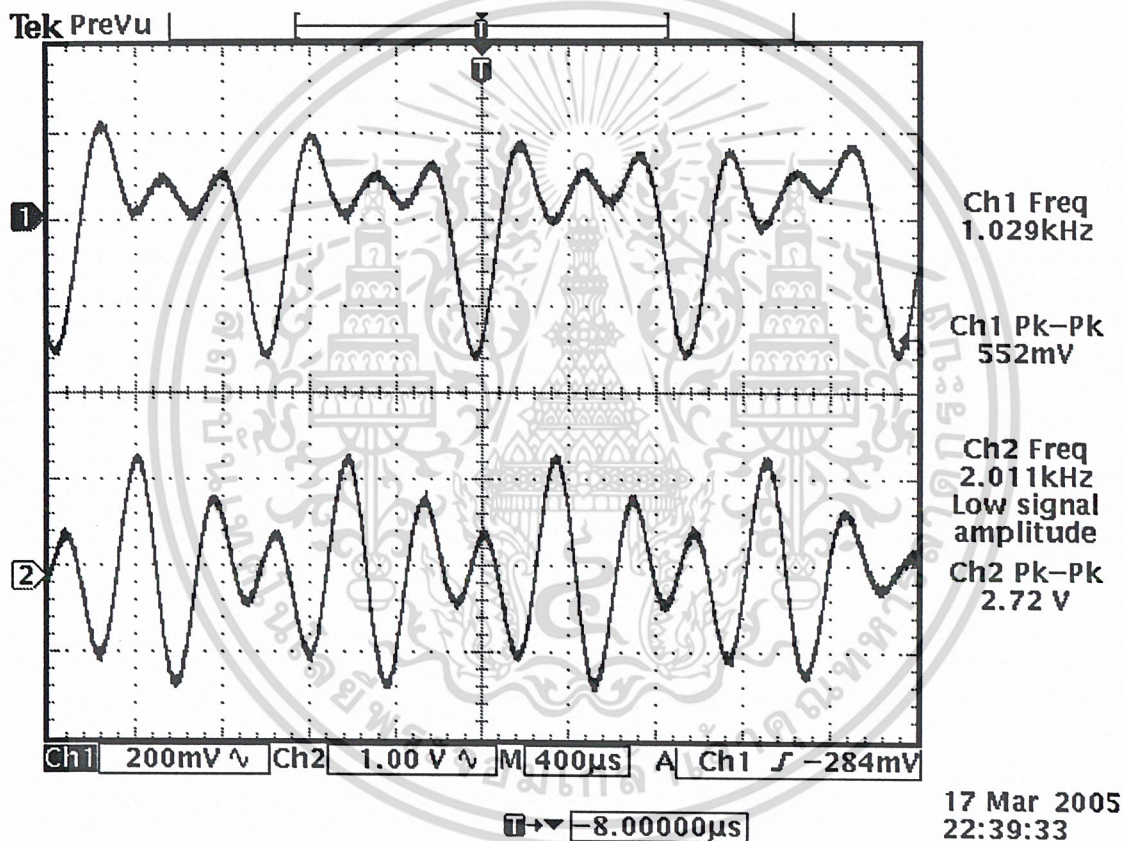
จากการทดลองที่ 6 จะเห็นได้ว่าลักษณะของสัญญาณที่ได้จากตัวกรองความถี่นั้น เมื่อเราป้อนสัญญาณสี่เหลี่ยมและสัญญาณสามเหลี่ยม รูปสัญญาณที่ได้จะมีค่าไม่ตรงกับสัญญาณที่ออกมา เกิดจากการผ่านตัวกรองความถี่กลางผ่านซึ่งจะกรองความถี่อื่นๆที่ไม่ได้อยู่ในช่วงความถี่ผ่านทิ้ง ซึ่งตรงตามทฤษฎีเนื่องจากองค์ประกอบของสัญญาณขาดหายไป

4.7 การทดลองที่ 7 ลักษณะของสัญญาณไซน์เมื่อทำการปรับอควอไลเซอร์

4.7.1 เริ่มต้นโดยทำการรันโปรแกรมอควอไลเซอร์ ทำการทดลองโดยการป้อนสัญญาณอินพุตผ่านที่เป็นสัญญาณไซน์ความถี่ต่างๆรวมกันคือ สัญญาณไซน์ความถี่ 1kHz , 2kHz และ 3kHz จากนั้นทำการปรับค่า Gain ในแต่ละย่านความถี่ซึ่งมีลักษณะดังรูปที่ 4.49 ได้สัญญาณเอาต์พุตดังรูปที่ 4.50



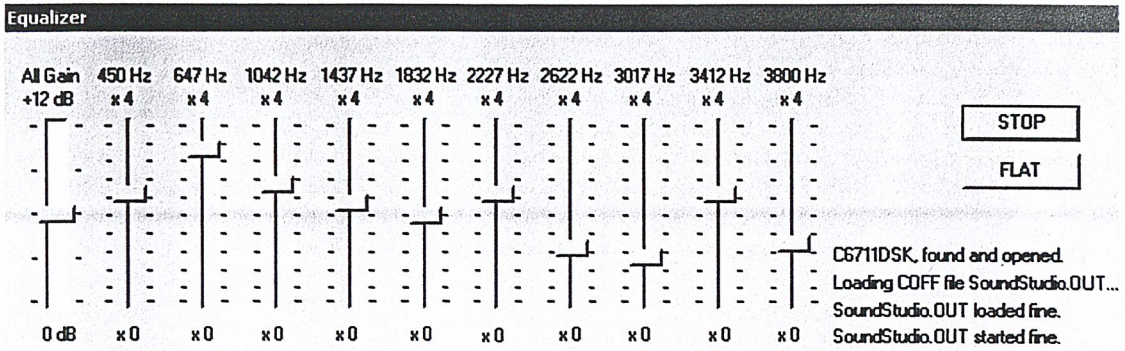
รูปที่ 4.51 ลักษณะของอิกวไลเซอร์เมื่อปรับค่า Gain ในแต่ละย่านความถี่



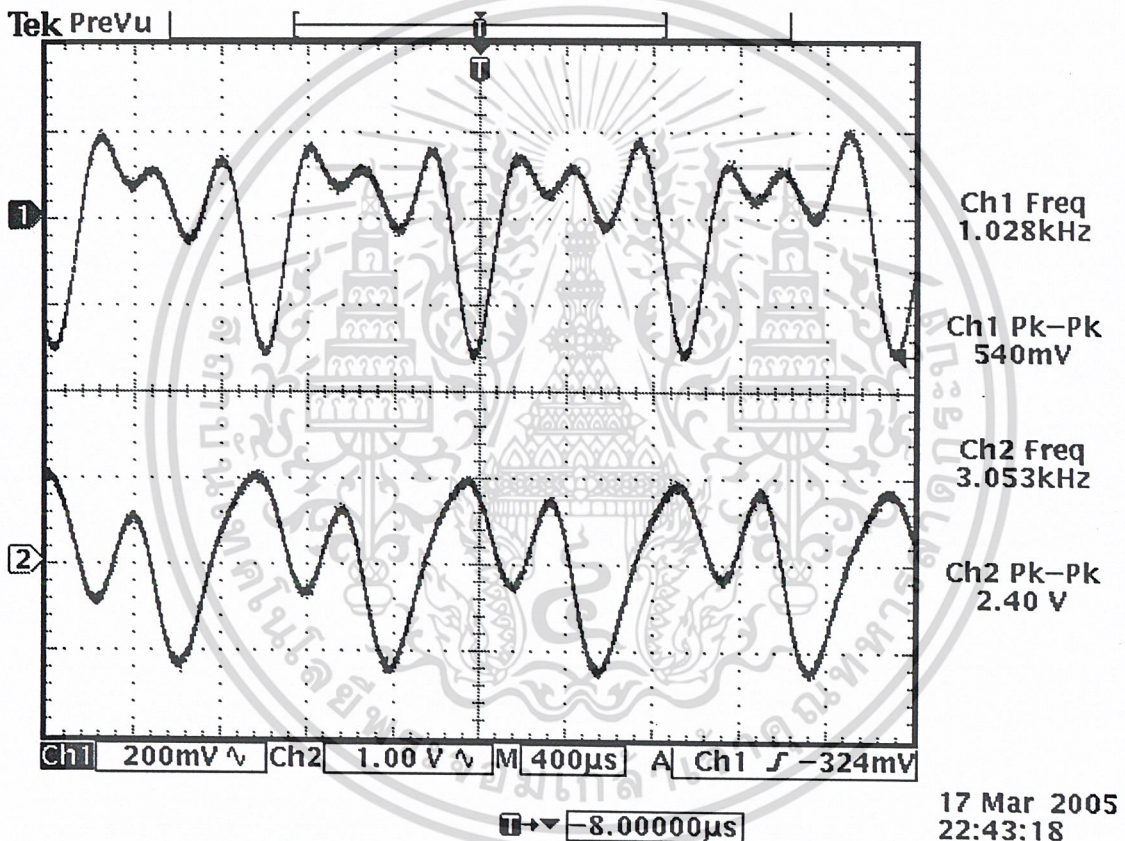
รูปที่ 4.52 Ch#1 อินพุตเป็นสัญญาณไซน์ความถี่ต่างๆรวมกัน
Ch#2 เอาท์พุทที่ได้หลังจากผ่านโปรแกรมอิกวไลเซอร์

4.7.2 เริ่มต้นโดยทำการรันโปรแกรมอิกวไลเซอร์ ทำการทดลองโดยการป้อนสัญญาณอินพุตผ่านที่เป็นสัญญาณไซน์ความถี่ต่างๆรวมกันคือ สัญญาณไซน์ความถี่ 1kHz , 2kHz และ 3kHz จากนั้นทำการปรับค่า Gain ในแต่ละย่านความถี่ซึ่งมีลักษณะดังรูปที่ 4.51 ได้สัญญาณเอาท์พุทดังรูปที่ 4.52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.53 ลักษณะของอิกวไลเซอร์เมื่อปรับค่า Gain ในแต่ละย่านความถี่



รูปที่ 4.54 Ch#1 อินพุตเป็นสัญญาณ ไซน์ความถี่ต่างๆรวมกัน
Ch#2 เอาท์พุทที่ได้หลังจากผ่าน โปรแกรมอิกวไลเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และบทสรุป

สรุปโปรแกรมที่ได้สร้างขึ้นมาบนบอร์ด TMS320C6211 นั้นเพื่อประกอบเป็นระบบขบวนการสตูดิโอที่สำเร็จมีดังนี้

- โปรแกรมเสียงเอ็คโค
- โปรแกรมเสียงรีเวิร์บ
- โปรแกรมเสียงคอร์ด
- โปรแกรมอิกวอลไลเซอร์ฟิลเตอร์แบบ FIR สิบย่านความถี่

ปัญหาที่เกิดขึ้นในการสร้างโปรแกรมนั้นปัญหาแรกเนื่องจากบอร์ดนี้มีความถี่แซมปลิงเท่ากับ 8000 Hz เท่านั้นเนื่องจาก Duagther Card ที่เราซื้อมาเพิ่มนั้นเกิดมีปัญหาคือมันไม่ Compatible จึงไม่สามารถทำในย่านเสียงพูดเท่านั้น

โดยโปรแกรมทั้งหมดจะมีส่วนของ GUI (Graphical User Interface) โดยใช้ Visual C++ คู่มือการใช้อยู่ภาคผนวกทางด้านหลัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการและรายงานฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีเนื่องจาก ได้ความรู้และประสิทธิภาพ วิชาจากบูรพาจารย์โดยเฉพาะอย่างยิ่ง อาจารย์ ปราโมทย์ วาดเขียน ที่ได้ให้คำปรึกษา พรีำสอน เคี้ยวเจ็ญ และให้การช่วยเหลือสนับสนุนทุกสิ่งทุกอย่างเป็นอย่างดีมาโดยตลอด

ขอขอบคุณ ทางภาควิชาโทรคมนาคม ที่ได้ให้โอกาสและให้การสนับสนุนเอื้อเฟื้ออุปการะ ทดลองต่างๆในการทำโครงการนี้

การทำโครงการนี้ไม่อาจสำเร็จลุล่วงไปได้ด้วยดีหากไม่ได้รับความรู้และคำแนะนำจากนาย เกรียงศักดิ์ พรหมภักดี ในการให้คำปรึกษาและแนะนำการใช้งานต่างๆที่เกี่ยวข้องกับการเขียน โปรแกรม เชื่อมต่อกับบอร์ดDSP (TMS320C6211) เป็นอย่างดีตลอดมา

รายงานฉบับนี้รวมถึงการเตรียมเสนองาน ไม่อาจเสร็จสิ้นได้หากไม่ได้รับความช่วยเหลือจาก นาย ปานวิทย์ ชูระนุติ ที่ให้การอนุเคราะห์เครื่องคอมพิวเตอร์ เครื่องปริ้นเตอร์ และเครื่องเขียนแผ่นซีดี รวมทั้งได้เป็นที่ปรึกษาให้กับคณะผู้จัดทำตลอดมา

และทั้งนี้เหนือสิ่งอื่นใด โครงการนี้ไม่อาจเสร็จสมบูรณ์ได้หากไม่ได้รับการช่วยเหลือและการ สนับสนุน กำลังใจทั้งหลายทั้งปวงจากคุณพ่อ คุณแม่ ผู้ปกครองและผู้ที่มีส่วนเกี่ยวข้องในการให้ คำปรึกษาที่ดีเสมอมาตลอดระยะเวลาของการทำโครงการนี้ ทางผู้จัดทำจึงขอขอบคุณทุกท่านอีกครั้งทั้งที่ เอ่ยนามและไม่ได้เอ่ยนามมาไว้ ณ ที่นี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

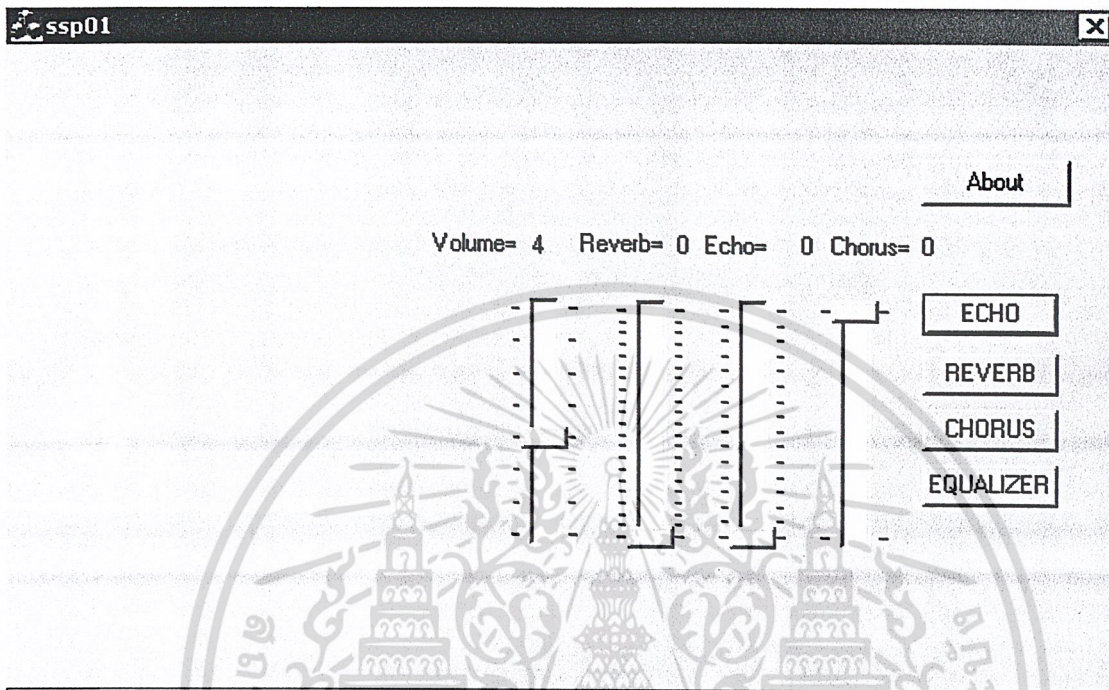
- [1] Rulph Chassaing, “DSP Application Using C and the TMS320C6x DSK”, A-Wiley – Interscience Publication
- [2] John G. Proakis & Dimitris G. Manolakis, “Digital Signal Processing”, Prentice-Hall International, INC.
- [3] Vinary K. Ingle & John G. Proakis, “Digital Sinal Processing USING MATLAB V.4”, PWS Publishing Company
- [4] Mihai V. Micea, Mircea Stratulat, Dan Ardelean, Daniel Aioanei, "Implementing Professional Audio Effects with DSPs", Transactions on Automatic Control and Computer Science, Vol. 46 (60), Periodica Politehnica, Timisoara, 2001, pp. (55-60), ISSN 1224-600X.
- [5] TMS320C6000 User's Guide
- [6] พรชัย ภววงษ์ศักดิ์, “การประมวลผลสัญญาณดิจิทัลเบื้องต้น”, มหาวิทยาลัยเทคโนโลยีมหานคร



ภาคผนวก

คู่มือการใช้โปรแกรม

หลังจากที่RUNโปรแกรมขึ้นมาจะปรากฏหน้าต่างดังรูป



รูป หน้าต่างควบคุมการใช้งาน

- การควบคุมความดังเบาของเสียง ใช้ Slider Bar อันซ้ายสุดในการควบคุม

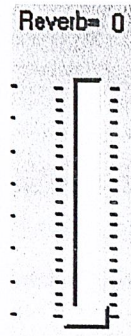
Volume= 4



รูป Slider Bar ที่ใช้ในการควบคุมความดังของเสียง

- การใช้งานฟังก์ชันรีเวิร์บขั้นแรกต้องไปคลิกที่ **REVERB** จากนั้นเราก็จะสามารถเลือกได้ว่าเราต้องการเสียงที่มีลักษณะรีเวิร์บมากน้อยเพียงใดโดยใช้ Slider Bar โดยด้านบนจะมีตัวเลขแสดงค่าในขณะที่ใช้งานฟังก์ชันรีเวิร์บยังสามารถปรับความดัง-เบาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป Slider Bar ที่ใช้ในการควบคุมการทำงานของฟังก์ชันรีเวิร์บ

- การใช้งานฟังก์ชันเอ็คโคขั้นแรกต้องไปคลิกที่ **ECHO** จากนั้นเราก็จะสามารถเลือกได้ว่าเราต้องการเสียงที่มีลักษณะเอ็คโคมากน้อยเพียงใดโดยใช้



รูป Slider Bar ที่ใช้ในการควบคุมการทำงานของฟังก์ชันเอ็คโค

- การใช้งานฟังก์ชันคอรัสขั้นแรกต้องไปคลิกที่ **CHORUS** จากนั้นเราก็จะสามารถเลือกได้ว่าเราต้องการเสียงที่มีลักษณะคอรัสมากน้อยเพียงใดโดยใช้

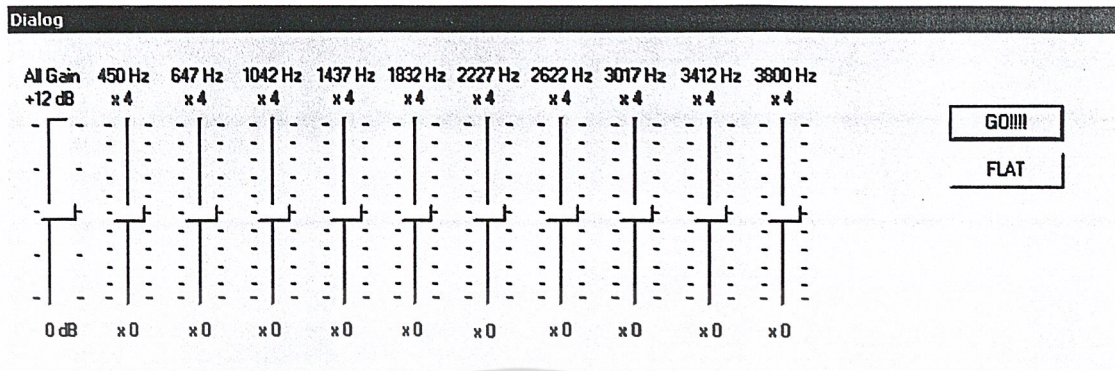


รูป Slider Bar ที่ใช้ในการควบคุมการทำงานของฟังก์ชันคอรัส

- การใช้งานอีควอไลเซอร์เมื่อเราไปคลิกที่ **EQUALIZER** จะปรากฏหน้าต่างดังรูป โดยจะประกอบด้วย 10 Slider bar แทนฟิลเตอร์ 10 ตัว สามารถ ตั้งเสียงสูง ต่ำได้ตามต้องการ โดยเลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Slider bar ส่วน Slider bar อันแรกเป็นตัวปรับเกนของระบบ ปุ่ม **FLAT** จะเป็นการ
เคลียร์ค่าต่างๆที่เซตไว้ กลับมาเป็นค่าเริ่มต้น



รูป หน้าต่างอีควอไลเซอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ซื้โปรแกรมและ Source Code



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้