

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

อีควอไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิก

DIGITAL AUDIO GRAPHIC EQUALIZER



โดย  
นาย วีระวัฒน์ วีระประเสริฐศักดิ์  
นางสาว ศิริวรรณ สุวะนาคะ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

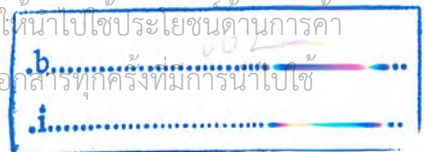
คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

หมู่.....  
ทะเบียน..... 55735  
เดือน,ปี..... 25 พ.ค. 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



DIGITAL AUDIO GRAPHIC EQUALIZER



A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
2003

หัวข้อวิทยานิพนธ์	อิกวอไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิก
TITLE	Digital Audio Graphic Equalizer
โดย	นาย วีระวัฒน์ วีระประเสริฐศักดิ์ รหัสนักศึกษา 43010410 นางสาว ศิริวรรณ สุวะนาคะ รหัสนักศึกษา 43010434
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร. กนก เจนจิระพงศ์เวช รศ. ชวลิต เบญจางคประเสริฐ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2546

วิทยานิพนธ์ฉบับนี้ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาด  
กระบัง

.....  
(รศ.ดร. กนก เจนจิระพงศ์เวช)  
อาจารย์ผู้ควบคุมวิทยานิพนธ์

.....  
(รศ. ชวลิต เบญจางคประเสริฐ)  
อาจารย์ผู้ควบคุมวิทยานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	อิควอไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิก
ชื่อนักศึกษา	นาย วีระวัฒน์ วีระประเสริฐศักดิ์ รหัสประจำตัว 43010410. นางสาว ศิริวรรณ สุวะนาคะ รหัสประจำตัว 43010434
อาจารย์ที่ปรึกษา	รศ.ดร. กนก เจนจิระพงศ์เวช รศ. ชวลิต เบนูจางคประเสริฐ
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2546

### บทคัดย่อ

โครงการนี้เป็นการนำความรู้ทางคณิตศาสตร์และการประมวลผลสัญญาณดิจิทัลมาใช้ประโยชน์ในการนำสัญญาณเสียงมาประมวลผลให้ได้รูปแบบอย่างที่ต้องการ โดยในโครงการนี้จะทำในส่วนของกราฟฟิกอิควอไลเซอร์ (Graphic Equalizer) ซึ่งทำงานบนเครื่องคอมพิวเตอร์โดยใช้โปรแกรม Max/MSP ในการสร้าง โดยโปรแกรมประยุกต์กราฟฟิกอิควอไลเซอร์ที่สร้างขึ้นมี 2 แบบ คือ โปรแกรมแบบที่ใช้การประมวลผลทางโดเมนเวลา โดยจะแบ่งย่านเสียงออกเป็น 10 ย่าน ซึ่งสามารถบูสท์หรือคัท(boost/cut) แต่ละย่านได้อย่างอิสระ และสามารถปรับความถี่คัตของตัวกรองแต่ละตัวได้ และโปรแกรมแบบที่ใช้การประมวลผลทางโดเมนความถี่จะสามารถแบ่งความถี่ในการทำอิควอไลเซอร์ได้ถึง 512 ย่าน

Project Title     Digital Audio Graphic Equalizer  
Student            Mr. Werawat Weraprasaertsak     ID. 43010410  
                         Miss Siriwan Suvanaka             ID. 43010434  
Advisor            Assoc. Prof. Dr. Kanok Janchitrapongvej  
                         Assoc. Prof. Chawalit Benjangkprasert  
Graduate Level   Bachelor Degree of Information Engineering  
Department       Information Engineering  
Academic Year    2003

## ABSTRACT

This project propose to apply the knowledge of Mathematic and Digital Signal Processing to utilize for signal processing as is expected by making 10-Band Graphic Equalizers. Users can Boost/Cut digital audio signal independently. This project presents two applications, one using time domain technique and another using frequency domain technique.

Project Title    Digital Audio Graphic Equalizer  
Student         Mr. Werawat Weraprasaertsak    ID. 43010410  
                     Miss Siriwan Suvanaka             ID. 43010434  
Advisor         Assoc. Prof. Dr. Kanok Janchitrapongvej  
                     Assoc. Prof. Chawalit Benjangkprasert  
Graduate Level   Bachelor Degree of Information Engineering  
Department     Information Engineering  
Academic Year   2003

## ABSTRACT

This project propose to apply the knowledge of Mathematic and Digital Signal Processing to utilize for signal processing as is expected by making 10-Band Graphic Equalizers. Users can Boost/Cut digital audio signal independently. This project presents two applications, one using time domain technique and another using frequency domain technique.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี เนื่องด้วยคณะผู้จัดทำได้รับความความอนุเคราะห์จากบุคคลต่างๆหลายฝ่าย ทั้งทางด้านการทำงานและทางด้านกำลังใจ  
ขอขอบพระคุณ รศ.ดร. กนก เจนจิระพงษ์เวช และ รศ. ชวลิต เบญจางคประเสริฐ ซึ่งเป็นอาจารย์ที่ปรึกษา ที่ได้ให้คำแนะนำ คำปรึกษาและ โอกาสในการทำโครงการนี้ ตลอดจนอาจารย์ทุกท่านที่ได้สอนสั่งและให้ความรู้ความเข้าใจในวิชาต่างๆที่คณะผู้จัดทำได้เรียนรู้มา  
ขอขอบคุณเพื่อนๆ พี่ๆทุกคน ขอขอบคุณหอสมุดกลางสำหรับแหล่งค้นคว้าข้อมูล  
ขอขอบพระคุณคุณพ่อคุณแม่ที่ให้การสนับสนุนและกำลังใจ  
ทางคณะผู้จัดทำขอขอบพระคุณท่านเหล่านั้นมา ณ โอกาสนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

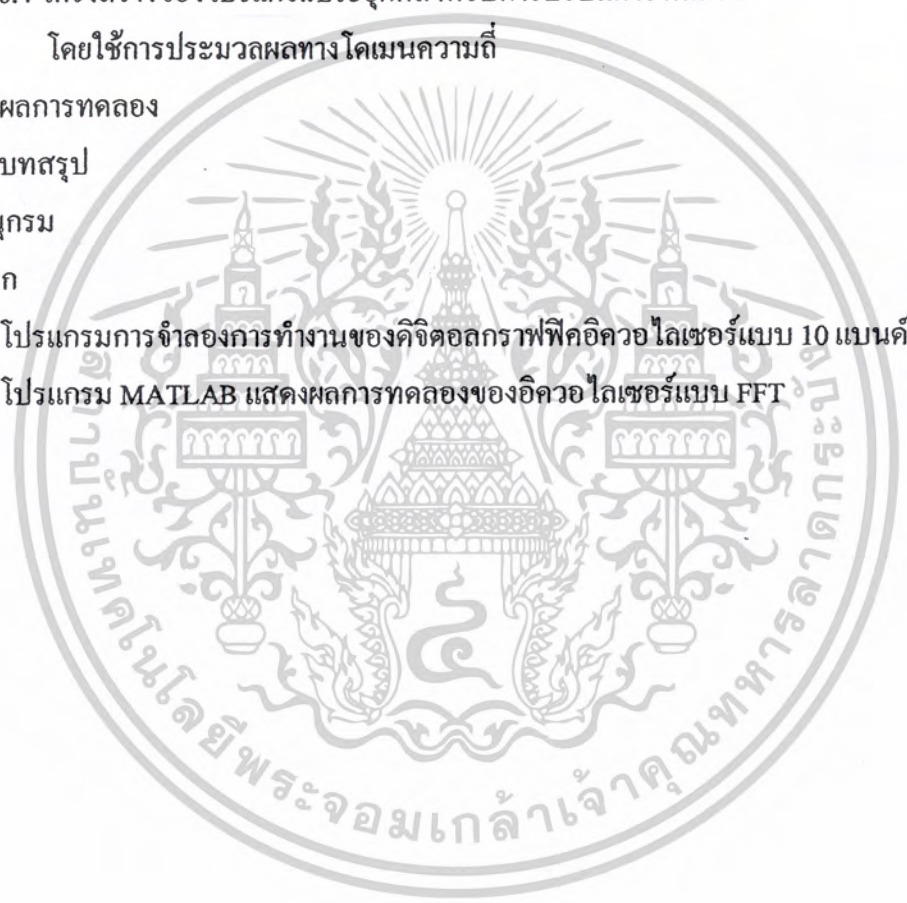
	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ฉ
สารบัญตาราง	ช
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีการประมวลผลสัญญาณดิจิทัล	3
2.1 หลักการสุ่มสัญญาณ (Sampling)	3
2.2 หลักการแบ่งขั้นสัญญาณ (Signal Quantization)	3
2.3 การสร้างสัญญาณคืน (Analog Reconstruction)	3
2.4 ตัวกรองสัญญาณดิจิทัล	4
2.4.1 ข้อดีของตัวกรอง FIR	4
2.4.2 ข้อดีของตัวกรอง IIR	5
2.5 ตัวกรองแบบ FIR	5
2.5.1 คุณสมบัติความสมมาตรของตัวกรองที่มีเฟสเชิงเส้น	6
2.5.2 การออกแบบโดยวิธีหน้าต่าง (Window Method)	7
2.5.3 วิธีการสร้างตัวกรอง FIR (FIR Filter Realization)	10
บทที่ 3 สัญญาณเสียงดิจิทัล	12
3.1 มาตรฐาน MIDI	12
3.2 โปรแกรม Max/MSP	12
บทที่ 4 หลักการออกแบบอิกวอลไลเซอร์	14
บทที่ 5 ผลการทดลองจากการออกแบบด้วยโปรแกรม MATLAB	19
บทที่ 6 การสร้างโปรแกรมประยุกต์	30
6.1 ความสามารถของแบบประมวลผลทางโดเมนเวลา	30
6.2 ความสามารถของแบบประมวลผลทางโดเมนความถี่	32
6.3 โครงสร้างของโปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง	

โดยใช้การประมวลผลทางโดเมนเวลา

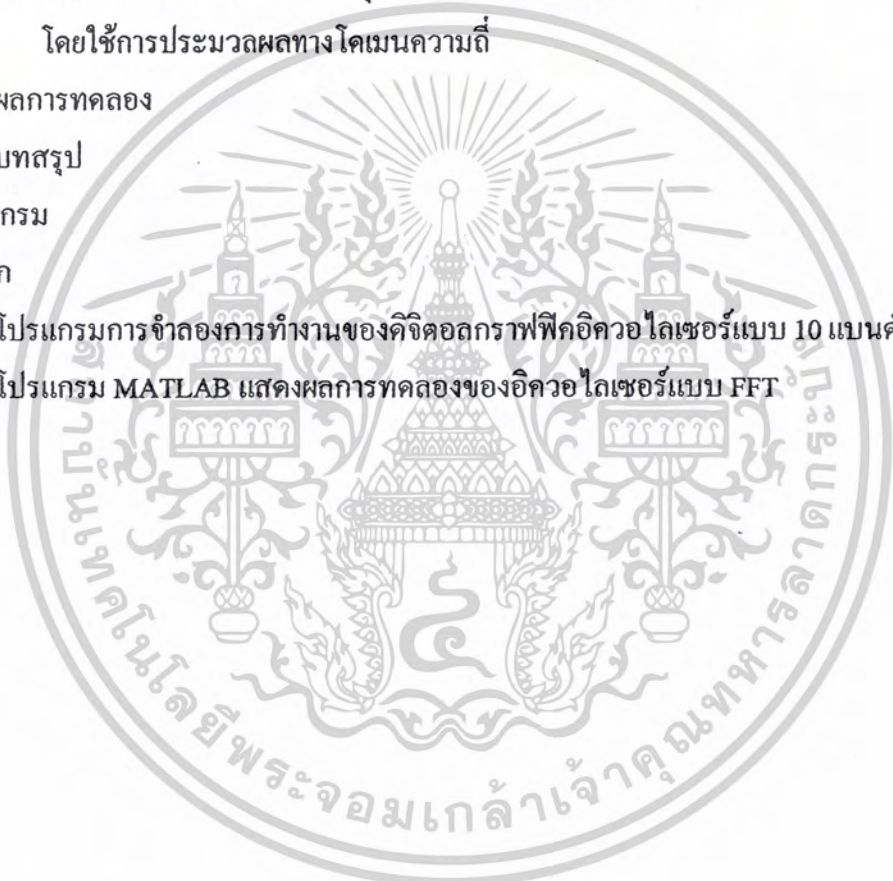
33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
6.3.1. ส่วนของการรับค่าจุดตัดความถี่	33
6.3.2. ส่วนของการออกแบบตัวกรอง	35
6.3.3. ส่วนของการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรอง	39
6.3.4. ส่วนของการควบคุมเกณฑ์และการคูณประสาน (Convolution)	40
6.3.5. ส่วนสัญญาณขาเข้าและสัญญาณขาออกของระบบ	42
6.4 โครงสร้างของ โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง โดยใช้การประมวลผลทาง โดเมนความถี่	43
บทที่ 7 ผลการทดลอง	45
บทที่ 8 บทสรุป	49
บรรณานุกรม	50
ภาคผนวก	51
โปรแกรมการจำลองการทำงานของคิจิตออลกราฟฟิคอิกวอไลเซอร์แบบ 10 แบนด์	51
โปรแกรม MATLAB แสดงผลการทดลองของอิกวอไลเซอร์แบบ FFT	55



	หน้า
6.3.1. ส่วนของการรับค่าจุดตัดความถี่	33
6.3.2. ส่วนของการออกแบบตัวกรอง	35
6.3.3. ส่วนของการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรอง	39
6.3.4. ส่วนของการควบคุมเกณฑ์และการคูณประสาน (Convolution)	40
6.3.5. ส่วนสัญญาณขาเข้าและสัญญาณขาออกของระบบ	42
6.4 โครงสร้างของ โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง โดยใช้การประมวลผลทางโดเมนความถี่	43
บทที่ 7 ผลการทดลอง	45
บทที่ 8 บทสรุป	49
บรรณานุกรม	50
ภาคผนวก	51
โปรแกรมการจำลองการทำงานของคิจิคอลกราฟฟิคอิกวอไลเซอร์แบบ 10 แบนด์	51
โปรแกรม MATLAB แสดงผลการทดลองของอิกวอไลเซอร์แบบ FFT	55



## สารบัญรูป

	หน้า
รูปที่ 1.1 โปรแกรมประยุกต์แบบที่ใช้การประมวลผลทางโดเมนเวลา	2
รูปที่ 1.2 โปรแกรมประยุกต์แบบที่ใช้การประมวลผลทางโดเมนความถี่	2
รูปที่ 2.1 แผนภาพการสร้างตัวกรองแบบ FIR โดยคอนโวลูชันปกติ	10
รูปที่ 2.2 แผนภาพการสร้างตัวกรองแบบ FIR เมื่ออาศัยคุณสมบัติการสมมาตร	11
รูปที่ 2.3 แผนภาพการสร้างตัวกรองแบบ FIR โดยวิธีคอนโวลูชันแบบเร็ว	11
รูปที่ 4.1 แสดง โครงสร้างของ 10 แบนด์	15
รูปที่ 4.2 ผลตอบสนองของตัวกรองทั้ง 10	16
รูปที่ 4.3 แผนภูมิขั้นตอนการดำเนินงานของการจำลองการทำงานของกราฟฟิคอิกวอไลเซอร์	17
รูปที่ 4.4 แผนภูมิขั้นตอนการดำเนินงานการจำลองการทำงานของพารามตริกอิกวอไลเซอร์	18
รูปที่ 5.1 แสดงผลตอบสนองความถี่ของสัญญาณที่จะ ใช้ทดสอบ	19
รูปที่ 5.2 แสดงสัญญาณที่ผ่าน 2400 Hz Lowpass filter	20
รูปที่ 5.3 แสดงสัญญาณที่ผ่าน 2400 -4800 Hz Bandpass filter	21
รูปที่ 5.4 แสดงสัญญาณที่ผ่าน 4800-7200 Hz Bandpass filter	21
รูปที่ 5.5 แสดงสัญญาณที่ผ่าน 7200-9600Hz Bandpass filter	22
รูปที่ 5.6 แสดงสัญญาณที่ผ่าน 9600-12000 Hz Bandpass filter	22
รูปที่ 5.7 แสดงสัญญาณที่ผ่าน 12-14.4 KHz Bandpass filter	23
รูปที่ 5.8 แสดงสัญญาณที่ผ่าน 14.4 -16.8 KHz Bandpass filter	23
รูปที่ 5.9 แสดงสัญญาณที่ผ่าน 16.8-19.2 KHz Bandpass filter	24
รูปที่ 5.10 แสดงสัญญาณที่ผ่าน 19.2-21.6 KHz Bandpass filter	24
รูปที่ 5.11 แสดงสัญญาณที่ผ่าน 21.6-24 KHz Highpass filter	25
รูปที่ 5.12 แสดงผลตอบสนองความถี่	26
รูปที่ 5.13 แสดงผลตอบสนองทางเฟส	27
รูปที่ 5.14 แสดงผลของการคูณกับค่าแกน	28
รูปที่ 5.15 แสดงผลของการคูณกับค่าแกน	29
รูปที่ 6.1 โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียงโดยใช้การประมวลผลทางโดเมนเวลา	31
รูปที่ 6.2 แผนภาพขั้นตอนการทำงานของอิกวอไลเซอร์ที่ใช้เทคนิคการประมวลในทางโดเมนเวลา	31

	หน้า
รูปที่ 6.3 โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง โดยใช้การประมวลผลทางโดเมนความถี่	32
รูปที่ 6.4 แผนภาพขั้นตอนการทำงานของอิกวอลไลเซอร์ที่ใช้เทคนิคการประมวลผลทางโดเมนความถี่	33
รูปที่ 6.5 ส่วนของการรับค่าจุดตัดความถี่	33
รูปที่ 6.6 แนวทางการสร้าง minmax	34
รูปที่ 6.7 ส่วนของการออกแบบตัวกรอง	35
รูปที่ 6.8 หลักการทำงานของตัวกรองแบบผ่านต่ำ	36
รูปที่ 6.9 สัมประสิทธิ์ของตัวกรอง	37
รูปที่ 6.10 หลักการทำงานของตัวกรองแบบผ่านแถบความถี่	38
รูปที่ 6.11 หลักการทำงานของตัวกรองแบบผ่านสูง	38
รูปที่ 6.12 ส่วนของการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรอง	39
รูปที่ 6.13 ตัวอย่างการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรองแบบผ่านสูง	40
รูปที่ 6.14 ส่วนของการรับค่าการเลื่อนเกนและปรับค่าเป็น dB	40
รูปที่ 6.15 ส่วนของการควบคุมเกนและการคูณประสาน	41
รูปที่ 6.16 ส่วนสัญญาณขาเข้าและสัญญาณขาออกของระบบ	42
รูปที่ 6.17 เริ่มต้นการใช้โครงสร้างโปรแกรมประยุกต์	43
รูปที่ 6.18 ส่วนของการเรียก Patch ที่ชื่อ iteproject 1024 ขึ้นมาใช้งาน	43
รูปที่ 6.19 โครงสร้างภายใน iteproject	44
รูปที่ 7.1 แสดงการวาดกราฟเพื่อปรับแต่งเสียง Snare Roll	45
รูปที่ 7.2 แสดงสเปกตรัมของเสียง Snare roll	46
รูปที่ 7.3 แสดงสเปกตรัมของเสียง Snare Roll เมื่อผ่านการปรับแต่งแล้ว	46
รูปที่ 7.4 แสดงสเปกตรัมของสัญญาณขาเข้า	47
รูปที่ 7.5 แสดงสเปกตรัมของสัญญาณขาออกเมื่อถูกปรับแต่งแล้ว	47
รูปที่ 7.6 แสดงสัญญาณขาเข้า เมื่อเทียบกับสัญญาณขาออก ในโดเมนเวลา	48

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

อิกวอไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิก

DIGITAL AUDIO GRAPHIC EQUALIZER



โดย

นาย วีระวัฒน์ วีระประเสริฐศักดิ์

นางสาว ศิริวรรณ สุวะนาคะ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่.....

เลขทะเบียน 55735

วัน,เดือน,ปี 25 พ.ค. 2548

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

DIGITAL AUDIO GRAPHIC EQUALIZER



A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
2003

หัวข้อวิทยานิพนธ์	อิกวอไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิก
TITLE	Digital Audio Graphic Equalizer
โดย	นาย วีระวัฒน์ วีระประเสริฐศักดิ์ รหัสนักศึกษา 43010410 นางสาว ศิริวรรณ สุวะนาคะ รหัสนักศึกษา 43010434
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร. กนก เจนจิระพงศ์เวช รศ. ชวลิต เบญจางคประเสริฐ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2546

วิทยานิพนธ์ฉบับนี้ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาดำเนินการตามหลักสูตร  
วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาด  
กระบัง

.....  
(รศ.ดร. กนก เจนจิระพงศ์เวช)

อาจารย์ผู้ควบคุมวิทยานิพนธ์

.....  
(รศ. ชวลิต เบญจางคประเสริฐ)

อาจารย์ผู้ควบคุมวิทยานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	อิกวอไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิก
ชื่อนักศึกษา	นาย วีระวัฒน์ วีระประเสริฐศักดิ์ รหัสประจำตัว 43010410. นางสาว ศิริวรรณ สุวะนาคะ รหัสประจำตัว 43010434
อาจารย์ที่ปรึกษา	รศ.ดร. กนก เงินจระพงศ์เวช รศ. ชวลิต เบญจางคประเสริฐ
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2546

### บทคัดย่อ

โครงการนี้เป็นการนำความรู้ทางคณิตศาสตร์และการประมวลผลสัญญาณดิจิทัลมาใช้ประโยชน์ในการนำสัญญาณเสียงมาประมวลผลให้ได้รูปแบบอย่างที่ต้องการ โดยในโครงการนี้จะทำในส่วนของกราฟฟิกอิกวอไลเซอร์ (Graphic Equalizer) ซึ่งทำงานบนเครื่องคอมพิวเตอร์โดยใช้โปรแกรม Max/MSP ในการสร้าง โดยโปรแกรมประยุกต์กราฟฟิกอิกวอไลเซอร์ที่สร้างขึ้นมี 2 แบบ คือ โปรแกรมแบบที่ใช้การประมวลผลทางโดเมนเวลา โดยจะแบ่งย่านเสียงออกเป็น 10 ย่าน ซึ่งสามารถบูสท์หรือคัท(boost/cut) แต่ละย่านได้อย่างอิสระ และสามารถปรับความถี่คัตของตัวกรองแต่ละตัวได้ และ โปรแกรมแบบที่ใช้การประมวลผลทาง โดเมนความถี่จะสามารถแบ่งความถี่ในการทำอิกวอไลเซอร์ได้ถึง 512 ย่าน

Project Title    Digital Audio Graphic Equalizer  
Student            Mr. Werawat Weraprasaertsak    ID. 43010410  
                         Miss Siriwan Suvanaka            ID. 43010434  
Advisor            Assoc. Prof. Dr. Kanok Janchitrapongvej  
                         Assoc. Prof. Chawalit Benjangkaprasert  
Graduate Level   Bachelor Degree of Information Engineering  
Department       Information Engineering  
Academic Year    2003

## ABSTRACT

This project propose to apply the knowledge of Mathematic and Digital Signal Processing to utilize for signal processing as is expected by making 10-Band Graphic Equalizers. Users can Boost/Cut digital audio signal independently. This project presents two applications, one using time domain technique and another using frequency domain technique.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี เนื่องด้วยคณะผู้จัดทำได้รับความความอนุเคราะห์จากบุคคลต่างๆหลายฝ่าย ทั้งทางด้านการทำงานและทางด้านกำลังใจ ขอขอบพระคุณ รศ.ดร. กนก เจริญพงศ์เวช และ รศ. ชวลิต เบญจางคประเสริฐ ซึ่งเป็นอาจารย์ที่ปรึกษา ที่ได้ให้คำแนะนำ คำปรึกษาและ โอกาสในการทำโครงการนี้ ตลอดจนอาจารย์ทุกท่านที่ได้สอนสั่งและให้ความรู้ความเข้าใจในวิชาต่างๆที่คณะผู้จัดทำได้เรียนรู้อมา ขอขอบคุณเพื่อนๆ พี่ๆทุกคน ขอขอบคุณหอสมุดกลางสำหรับแหล่งค้นคว้าข้อมูล ขอขอบพระคุณคุณพ่อคุณแม่ที่ให้การสนับสนุนและกำลังใจ ทางคณะผู้จัดทำขอขอบพระคุณท่านเหล่านั้นมา ณ โอกาสนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ฉ
สารบัญตาราง	ช
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีการประมวลผลสัญญาณดิจิทัล	3
2.1 หลักการสุ่มสัญญาณ (Sampling)	3
2.2 หลักการแบ่งขั้นสัญญาณ (Signal Quantization)	3
2.3 การสร้างสัญญาณคืน (Analog Reconstruction)	3
2.4 ตัวกรองสัญญาณดิจิทัล	4
2.4.1 ข้อดีของตัวกรอง FIR	4
2.4.2 ข้อดีของตัวกรอง IIR	5
2.5 ตัวกรองแบบ FIR	5
2.5.1 คุณสมบัติความสมมาตรของตัวกรองที่มีเฟสเชิงเส้น	6
2.5.2 การออกแบบโดยวิธีหน้าต่าง (Window Method)	7
2.5.3 วิธีการสร้างตัวกรอง FIR (FIR Filter Realization)	10
บทที่ 3 สัญญาณเสียงดิจิทัล	12
3.1 มาตรฐาน MIDI	12
3.2 โปรแกรม Max/MSP	12
บทที่ 4 หลักการออกแบบอควอไลเซอร์	14
บทที่ 5 ผลการทดลองจากการออกแบบด้วยโปรแกรม MATLAB	19
บทที่ 6 การสร้างโปรแกรมประยุกต์	30
6.1 ความสามารถของแบบประมวลผลทางโดเมนเวลา	30
6.2 ความสามารถของแบบประมวลผลทางโดเมนความถี่	32
6.3 โครงสร้างของ โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง	

โดยใช้การประมวลผลทางโดเมนเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
6.3.1. ส่วนของการรับค่าจุดตัดความถี่	33
6.3.2. ส่วนของการออกแบบตัวกรอง	35
6.3.3. ส่วนของการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรอง	39
6.3.4. ส่วนของการควบคุมเกณฑ์และการคูณประสาน (Convolution)	40
6.3.5. ส่วนสัญญาณขาเข้าและสัญญาณขาออกของระบบ	42
6.4 โครงสร้างของ โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง โดยใช้การประมวลผลทางโดเมนความถี่	43
บทที่ 7 ผลการทดลอง	45
บทที่ 8 บทสรุป	49
บรรณานุกรม	50
ภาคผนวก	51
โปรแกรมการจำลองการทำงานของคิจิตอกรรภาพิกอิกวอไลเซอร์แบบ 10 แบนด์	51
โปรแกรม MATLAB แสดงผลการทดลองของอิกวอไลเซอร์แบบ FFT	55

## สารบัญรูป

	หน้า
รูปที่ 1.1 โปรแกรมประยุกต์แบบที่ใช้การประมวลผลทางโดเมนเวลา	2
รูปที่ 1.2 โปรแกรมประยุกต์แบบที่ใช้การประมวลผลทางโดเมนความถี่	2
รูปที่ 2.1 แผนภาพการสร้างตัวกรองแบบ FIR โดยคอนโวลูชันปกติ	10
รูปที่ 2.2 แผนภาพการสร้างตัวกรองแบบ FIR เมื่ออาศัยคุณสมบัติการสมมาตร	11
รูปที่ 2.3 แผนภาพการสร้างตัวกรองแบบ FIR โดยวิธีคอนโวลูชันแบบเร็ว	11
รูปที่ 4.1 แสดง โครงสร้างของ 10 แบนด์	15
รูปที่ 4.2 ผลตอบสนองของตัวกรองทั้ง 10	16
รูปที่ 4.3 แผนภูมิขั้นตอนการดำเนินงานของการจำลองการทำงานของกราฟฟิคอิกวอไลเซอร์	17
รูปที่ 4.4 แผนภูมิขั้นตอนการดำเนินงานการจำลองการทำงานของพารามตริกอิกวอไลเซอร์	18
รูปที่ 5.1 แสดงผลตอบสนองความถี่ของสัญญาณที่จะใช้ทดสอบ	19
รูปที่ 5.2 แสดงสัญญาณที่ผ่าน 2400 Hz Lowpass filter	20
รูปที่ 5.3 แสดงสัญญาณที่ผ่าน 2400-4800 Hz Bandpass filter	21
รูปที่ 5.4 แสดงสัญญาณที่ผ่าน 4800-7200 Hz Bandpass filter	21
รูปที่ 5.5 แสดงสัญญาณที่ผ่าน 7200-9600Hz Bandpass filter	22
รูปที่ 5.6 แสดงสัญญาณที่ผ่าน 9600-12000 Hz Bandpass filter	22
รูปที่ 5.7 แสดงสัญญาณที่ผ่าน 12-14.4 KHz Bandpass filter	23
รูปที่ 5.8 แสดงสัญญาณที่ผ่าน 14.4-16.8 KHz Bandpass filter	23
รูปที่ 5.9 แสดงสัญญาณที่ผ่าน 16.8-19.2 KHz Bandpass filter	24
รูปที่ 5.10 แสดงสัญญาณที่ผ่าน 19.2-21.6 KHz Bandpass filter	24
รูปที่ 5.11 แสดงสัญญาณที่ผ่าน 21.6-24 KHz Highpass filter	25
รูปที่ 5.12 แสดงผลตอบสนองความถี่	26
รูปที่ 5.13 แสดงผลตอบสนองทางเฟส	27
รูปที่ 5.14 แสดงผลของการคูณกับค่าเกน	28
รูปที่ 5.15 แสดงผลของการคูณกับค่าเกน	29
รูปที่ 6.1 โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียงโดยใช้การประมวลผลทางโดเมนเวลา	31
รูปที่ 6.2 แผนภาพขั้นตอนการทำงานของอิกวอไลเซอร์ที่ใช้เทคนิคการประมวล	

ในทางโดเมนเวลา

31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 6.3 โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง โดยใช้การประมวลผลทางโดเมนความถี่	32
รูปที่ 6.4 แผนภาพขั้นตอนการทำงานของอิกวอไลเซอร์ที่ใช้เทคนิคการประมวลผลทางโดเมนความถี่	33
รูปที่ 6.5 ส่วนของการรับค่าจุดตัดความถี่	33
รูปที่ 6.6 แนวทางการสร้าง minmax	34
รูปที่ 6.7 ส่วนของการออกแบบตัวกรอง	35
รูปที่ 6.8 หลักการทำงานของตัวกรองแบบผ่านต่ำ	36
รูปที่ 6.9 สัมประสิทธิ์ของตัวกรอง	37
รูปที่ 6.10 หลักการทำงานของตัวกรองแบบผ่านแถบความถี่	38
รูปที่ 6.11 หลักการทำงานของตัวกรองแบบผ่านสูง	38
รูปที่ 6.12 ส่วนของการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรอง	39
รูปที่ 6.13 ตัวอย่างการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรองแบบผ่านสูง	40
รูปที่ 6.14 ส่วนของการรับค่าการเลื่อนเกนและปรับค่าเป็น dB	40
รูปที่ 6.15 ส่วนของการควบคุมเกนและการคูณประสาน	41
รูปที่ 6.16 ส่วนสัญญาณขาเข้าและสัญญาณขาออกของระบบ	42
รูปที่ 6.17 เริ่มต้นการใช้โครงสร้างโปรแกรมประยุกต์	43
รูปที่ 6.18 ส่วนของการเรียก Patch ที่ชื่อ iteproject 1024 ขึ้นมาใช้งาน	43
รูปที่ 6.19 โครงสร้างภายใน iteproject	44
รูปที่ 7.1 แสดงการวาดกราฟเพื่อปรับแต่งเสียง Snare Roll	45
รูปที่ 7.2 แสดงสเปกตรัมของเสียง Snare roll	46
รูปที่ 7.3 แสดงสเปกตรัมของเสียง Snare Roll เมื่อผ่านการปรับแต่งแล้ว	46
รูปที่ 7.4 แสดงสเปกตรัมของสัญญาณขาเข้า	47
รูปที่ 7.5 แสดงสเปกตรัมของสัญญาณขาออกเมื่อถูกปรับแต่งแล้ว	47
รูปที่ 7.6 แสดงสัญญาณขาเข้า เมื่อเทียบกับสัญญาณขาออก ในโดเมนเวลา	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

	หน้า
ตารางที่ 2.1 ผลตอบสนองต่ออิมพัลส์ของตัวกรองอุดมคติต่าง ๆ	8
ตารางที่ 2.2 วิธีหน้าต่างแบบต่าง ๆ และค่าพารามิเตอร์ที่สำคัญ	10
ตารางที่ 6.1 สมการตัวกรองแบบต่างๆ	35



## บทที่ 1

### บทนำ

เนื่องจากระบบการสื่อสารต่างๆ หรือแม้แต่วิธีการบันทึกเสียงได้เปลี่ยนมาใช้ระบบแบบดิจิทัล (digital) และมีผู้ใช้กันอย่างแพร่หลายเนื่องจากมีความสะดวกและแม่นยำกว่า ซึ่งเป็นที่ทราบกันว่าในการส่งผ่านสัญญาณแบบอนาล็อก (analog) นั้น จะเกิดการลดทอนหรือสูญเสียสัญญาณ ซึ่งระบบดิจิทัลมีคุณสมบัติในการรักษาสัญญาณด้วยการตรวจจับและแก้ไขสัญญาณในส่วนที่เสียหายได้ แต่กระนั้นก็ดี อุปกรณ์อย่างอีควอลไลเซอร์ (Equalizer) ได้เข้ามามีบทบาทในการบูสต์หรือคัท (boost/cut) สัญญาณตั้งแต่แบบระบบอนาล็อก และเนื่องจากสัญญาณแบบอนาล็อกมีข้อเสียตามที่ได้กล่าวมาในตอนต้น จึงมีความต้องการที่จะนำมาใช้กับระบบดิจิทัล ซึ่งต้องเปลี่ยนรูปแบบของอีควอลไลเซอร์ใหม่ทั้งหมด

#### จุดประสงค์

1. เพื่อศึกษาการจำลองการประมวลผลสัญญาณดิจิทัลโดยโปรแกรม MATLAB
2. เพื่อศึกษาการทำงาน, การเขียนโปรแกรม และอัลกอริธึม เพื่อนำไปใช้ทำ กราฟฟิกอีควอลไลเซอร์แบบ 10 แบนด์ โดยการประยุกต์ใช้โปรแกรม Max/MSP เพื่อนำไปพัฒนาการประมวลผลสัญญาณในจุดประสงค์อื่น ๆ
3. เพื่อสร้างกราฟฟิกอีควอลไลเซอร์แบบ 10 แบนด์
4. เพื่อนำไปใช้งานเป็นอีควอลไลเซอร์ในย่านความถี่ 20 Hz-20 KHz

#### ขอบเขตของงาน

1. สามารถจำลองการทำงานของอีควอลไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิกแบบ 10 ย่านได้
2. นำรูปแบบของการจำลองมาสร้างเป็นอีควอลไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิกแบบ 10 ย่าน โดยประยุกต์ใช้โปรแกรม Max/Msp
3. สามารถทำการปรับย่านความถี่ 10 ย่านครอบคลุมความถี่ 20 Hz-20 KHz
4. สามารถนำไปประยุกต์ใช้งานในด้านอื่น ๆ ได้ ยกตัวอย่างเช่นการ แยกเสียงมากกว่า 1 เสียง ออกจากกันอิสระ

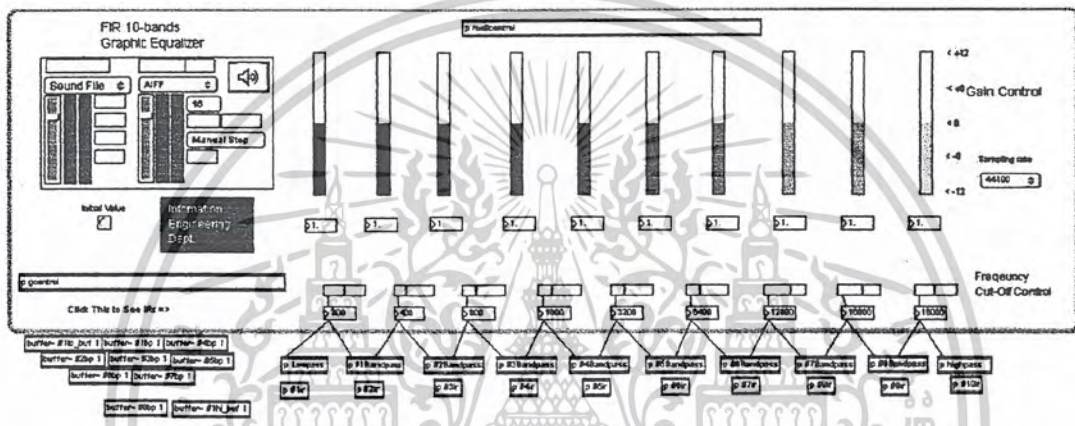
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รูปแบบโปรแกรมประยุกต์ที่สร้างขึ้น

โปรแกรมอิกวไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิคสร้างขึ้นมาจากการประยุกต์ใช้โปรแกรม Max/Msp ซึ่งโปรแกรมกราฟฟิคอิกวไลเซอร์ที่สร้างขึ้นทำงานบนเครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการวินโดวส์ได้ทุกเครื่อง

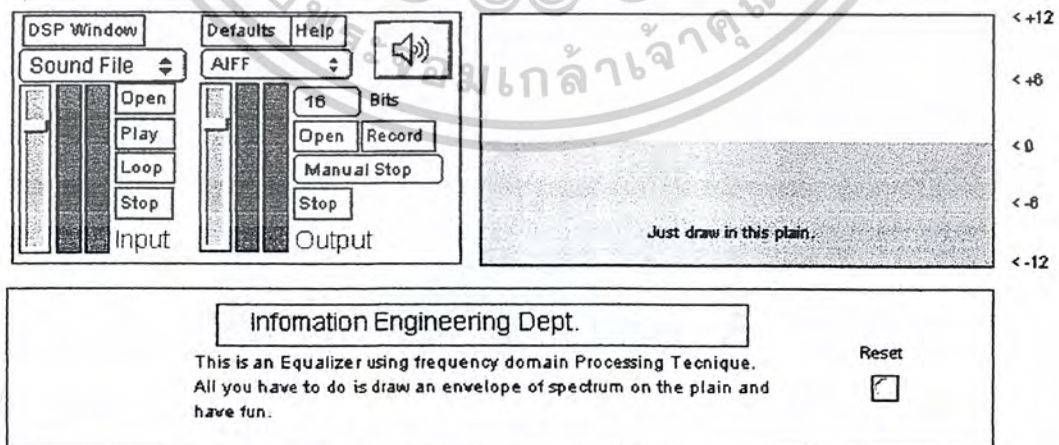
โดยโปรแกรมประยุกต์ที่สร้างขึ้นมี 2 แบบ

1.แบบที่ใช้การประมวลผลทางโดเมนเวลา ซึ่งมีรูปแบบโปรแกรมดังรูปที่ 1.1



รูปที่ 1.1 โปรแกรมประยุกต์แบบที่ใช้การประมวลผลทางโดเมนเวลา

2.แบบที่ใช้การประมวลผลทางโดเมนความถี่ ซึ่งมีรูปแบบโปรแกรมดังรูปที่ 1.2



รูปที่ 1.2 โปรแกรมประยุกต์แบบที่ใช้การประมวลผลทางโดเมนความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีการประมวลผลสัญญาณดิจิทัล

การประมวลผลสัญญาณดิจิทัล สัญญาณที่จะนำมาประมวลผลนั้นต้องแปลงเป็นสัญญาณดิจิทัลก่อน โดยวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล ซึ่งประกอบด้วย 2 กระบวนการคือ วงจรสุ่มสัญญาณ (Sampler) และ วงจรแบ่งขั้นสัญญาณ (Quantizer)

#### 2.1 หลักการสุ่มสัญญาณ (Sampling)

แนวคิดของการสุ่มสัญญาณคือ การนำเอาสัญญาณขาเข้าแบบต่อเนื่องมาผ่านสวิทช์อุดมคติที่ต่อวงจรที่ตำแหน่งเวลาเท่ากับ  $\dots, -2T, -T, 0, T, 2T, 3T, \dots$  และเปิดวงจรที่เวลาอื่น ๆ นั่นคือ มีความถี่ของการตัดต่อวงจรเท่ากับ  $f_s$  ครั้งต่อวินาที หรือมีคาบของการสุ่มเท่ากับ  $T=1/f_s$

ทฤษฎีการสุ่มสัญญาณของไนควิสต์(Nyquist) กำหนดว่า ความถี่ในการสุ่มที่น้อยที่สุดจะต้องมีค่ามากกว่าสองเท่าของความถี่สูงสุดในสัญญาณ หรือ  $f_s > 2 f_{\max}$  ซึ่งความถี่ที่  $2f_{\max}$  เรียกว่า “ความถี่ไนควิสต์” (Nyquist frequency) หรืออัตรา ไนควิสต์

ถ้าเราใช้  $f_s$  ต่ำกว่าค่าความถี่ไนควิสต์ จะทำให้สัญญาณที่สุ่มออกมาจะมีช่วงของความถี่ส่วนปลายที่ซ้อนทับกัน เรียกความถี่ส่วนที่ซ้อนทับกันนี้ว่า aliasing ซึ่งถ้าเกิดซ้อนทับในช่วง  $-f_s/2$  จนถึง  $f_s/2$  แสดงว่าสัญญาณขาเข้าที่สุ่มมาได้มีความผิดเพี้ยน

#### 2.2 หลักการแบ่งขั้นสัญญาณ (Signal Quantization)

การแบ่งขั้นสัญญาณ หมายถึง การแทนค่าสัญญาณที่ถูกสุ่มมาซึ่งมีความละเอียดไม่จำกัด (เพราะมาจากระดับสัญญาณอนาลอก) ด้วยระบบเลขฐานสองที่มีจำนวนบิตจำกัด ในกรณีที่ใช้ระบบเลขจำนวนเต็ม(fixed-point)จะให้ขนาดของขั้นคงที่ ตลอดช่วงที่แทนค่าสัญญาณ การใช้จำนวนบิตยิ่งมากเท่าไรก็จะ ได้การแบ่งขั้นที่ละเอียด และแทนสัญญาณจริงได้ถูกต้องมากขึ้นเท่านั้น ถ้าเราให้  $B$  เป็นจำนวนบิตที่ใช้แทน 1 ค่าของสัญญาณ,  $R$  เป็นช่วงของค่าสัญญาณที่สามารถแทนได้, และ  $Q$  แทนเป็นความกว้างของขั้น จะได้ว่าจำนวนขั้นทั้งหมด  $= R / Q = 2^B$

#### 2.3 การสร้างสัญญาณคืน (Analog Reconstruction)

การสร้างสัญญาณคืน เปลี่ยนจากสัญญาณดิจิทัลเป็นสัญญาณอนาลอก ในทางทฤษฎีทำได้โดยผ่านสัญญาณแบบไม่ต่อเนื่องเข้าไปยังตัวกรอง (อนาลอก) แบบผ่านความถี่ต่ำที่มีความถี่ตัดเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่  $f_c/2$  ตัวกรองจะผ่านเฉพาะสัญญาณในช่วงความถี่ระหว่าง  $-f_c/2$  ถึง  $f_c/2$  หรือช่วงในควิซท์ออกมาผลที่ได้คือ สำเนาของสเปกตรัมที่อยู่รอบความถี่ 0 ซึ่งเป็นสัญญาณอนาล็อกที่มีรูปร่างเป็นขอบของสัญญาณแบบไม่ต่อเนื่องก่อนสร้างกลับ

ในทางปฏิบัติ จะนำสัญญาณ  $y(n)$  มาผ่านวงจรคงค่า (hold) ที่ทำงานเข้าจังหวะกับอัตราสุ่มของสัญญาณ  $y(n)$  เพื่อสร้างเป็นสัญญาณลักษณะขั้นบันไดก่อน จากนั้นจึงใช้สัญญาณขั้นบันไดสร้างเป็นสัญญาณขาออก โดยส่งมันผ่านตัวกรองผ่านค่าเพื่อสร้างสัญญาณคืนอีกทีหนึ่ง

## 2.4 ตัวกรองสัญญาณดิจิทัล

ตัวกรองสัญญาณดิจิทัลมี 2 แบบ คือแบบ FIR (Finite Impulse Response) และ IIR (Infinite Impulse Response)

ระบบไม่ต่อเนื่องที่มีคุณสมบัติเชิงเส้น และไม่แปรตามเวลา สามารถถูกกำหนดลักษณะได้โดย  $h(n)$  ซึ่ง  $h(n)$  ที่แตกต่างกันก็ส่งผลให้เกิดฟังก์ชันของระบบที่แตกต่างกันไป

ระบบแบบ FIR จะมีความยาวของ  $h(n)$  จำกัด เท่ากับ  $N$  จุด ระบบนี้เป็นตัวกรองแบบ FIR ที่มีอันดับ (order) เท่ากับ  $N-1$  ถ้าสมมติว่า  $h(n)$  มีค่าแรกที่  $n=0$  จะได้ว่า  $h(n)$  มีค่าเป็นศูนย์ที่  $n < 0$  และ  $n > N-1$

ระบบแบบ IIR จะมี  $h(n)$  ไม่จำกัด ซึ่ง  $h(n)$  อาจมีความยาวไม่จำกัดในช่วงที่  $n$  เป็นค่าบวกหรือลบ หรือ ทั้งสองด้านก็ได้  $h(n)$  มีค่าไปเรื่อย ๆ จนถึงเวลาที่  $n$  มีค่าเป็นอนันต์

### 2.4.1 ข้อดีของตัวกรอง FIR

1. ให้ผลตอบสนองเชิงความถี่ที่มีเฟสแบบเชิงเส้น โดยสมบูรณ์ตลอดช่วงแถบผ่าน ซึ่งไม่สามารถทำได้โดยตัวกรอง IIR หรือแม้แต่ตัวกรองอนาล็อกใด ๆ (ทำได้แต่ไม่เชิงเส้นสมบูรณ์)
2. ตัวกรอง FIR ไม่มีการป้อนกลับ หรือไม่มีโพล ทำให้มีเสถียรภาพเสมอ
3. ตัวกรอง FIR มีความทนทานดีกว่าต่อความคลาดเคลื่อนจากการประมาณค่าสัมประสิทธิ์และความคลาดเคลื่อนจากการคำนวณ ความคลาดเคลื่อนนี้เกิดมาจากการใช้เลขฐานสองที่มีความยาวจำกัด (finite word length) ในการแทนค่าเลขจริง ๆ ความคลาดเคลื่อนนี้ส่งผลถึงลักษณะต่างๆ ของระบบจากที่ออกแบบไว้ เช่น ความถี่ตัด, รูปร่างของผลตอบสนองความถี่ และรวมถึงเสถียรภาพของระบบด้วย

### 2.4.2 ข้อดีของตัวกรอง IIR

1. ให้ผลตอบสนองเชิงความถี่ที่ดีกว่ามากในด้านความคม (ขนาดของแถบเปลี่ยนเล็กกว่า และความพลิวต่ำกว่า) ที่ขนาดอันดับเท่า ๆ กับตัวกรอง FIR นั่นหมายถึงว่า โดยทั่วไปเราไม่จำเป็นต้องใช้ ตัวกรอง IIR ที่มีอันดับสูง ๆ เหมือน FIR ดังนั้น ตัวกรอง IIR จึงมีความต้องการด้านความเร็วของตัวประมวลผลที่น้อยกว่า

2. สามารถออกแบบโดยอิงจากตัวกรองคั่นแบบอนาลอกได้ ถ้ามีตัวกรองแบบอนาลอกที่เคยใช้งานอยู่แล้ว

ในที่นี้ เราเลือกใช้ ตัวกรองแบบ FIR ในการออกแบบ

### 2.5 ตัวกรองแบบ FIR

สิ่งที่เราต้องการหาในการออกแบบ คือ ค่าของผลตอบสนองต่ออิมพัลส์ หรือ  $h(n)$  ของระบบ สำหรับตัวกรอง FIR ที่มี  $h(n)$  ยาว  $N$  จุด ตัวกรองนี้มีอันดับเท่ากับ  $N-1$  เหตุผลก็คือ มีการใช้สัญญาณขาเข้าในอดีตย้อนหลังไป  $N-1$  ตำแหน่ง หรือ ตัวกำลังสูงสุดที่อยู่ในฟังก์ชัน  $H(z)$  ก็คือ  $z^{-(N-1)}$

คุณสมบัติเฟสแบบเชิงเส้น (linear phase) คือ คุณสมบัติของระบบที่มี ผลตอบสนองทางเฟสมีลักษณะเป็นเชิงเส้น คุณสมบัตินี้เป็นคุณสมบัติที่สำคัญมากของระบบ และเฉพาะตัวกรองแบบ FIR เท่านั้นที่สามารถมีคุณสมบัตินี้ได้อย่างสมบูรณ์ ระบบมีผลตอบสนองเชิงความถี่ที่จัดให้อยู่ในรูปได้ ดังนี้

$$H(e^{j\omega'}) = A(e^{j\omega'}) e^{j\theta(e^{j\omega'})} \quad (2.1)$$

โดยที่  $A$  คือ อัตราขยายหรือลดทอนของระบบ และ  $\theta$  คือเฟสของสัญญาณขาออกที่เปลี่ยนไปจากสัญญาณขาเข้า โดยทั้งคู่มีค่าแปรตามความถี่ ระบบมีเฟสเป็นแบบเชิงเส้นโดยสมบูรณ์ เมื่อ  $\theta$  เป็นฟังก์ชันแบบเชิงเส้นของ  $\omega'$

$$\theta = -a\omega' \quad (2.2)$$

โดยที่  $a$  เป็นค่าคงที่ที่ไม่แปรตามความถี่ เฟสของสัญญาณขาออกมีการเปลี่ยนแปลงที่เป็นเชิงเส้นกับความถี่ของสัญญาณขาเข้า ทำให้สัญญาณขาออกมีความล่าช้าทางเฟส (phase delay) ที่คงที่ตลอดทุก ๆ ความถี่ ซึ่งความล่าช้าทางเฟสมีสมการ คือ

$$T_p = -\theta / \omega \quad (2.3)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในที่นี้จะได้  $T_p$  คงที่เท่ากับ  $a$  การที่ระบบมีเฟสเชิงเส้นมีผลคือ ทำให้ไม่เกิดความผิดเพี้ยนทางเฟส (phase distortion)

ในงานบางอย่าง ไม่จำเป็นที่จะต้องมีการล่าช้าทางเฟส ( $T_p$ ) คงที่ แต่ต้องการเพียงแค่มิการล่าช้าของกลุ่ม (group delay) คงที่

$$T_{group} = d\theta / d\omega' \quad (2.4)$$

ระบบใดที่มีการล่าช้าทางเฟสคงที่ก็จะมีการล่าช้าของกลุ่มคงที่ด้วย ผลตอบสนองทางเฟสของระบบที่มีเฟสเชิงเส้น คือ

$$\theta = -a\omega' + b \quad (2.5)$$

โดยที่  $a$  และ  $b$  เป็นค่าคงที่ที่ไม่แปรตามความถี่

### 2.5.1 คุณสมบัติความสมมาตรของตัวกรองที่มีเฟสเชิงเส้น

ตัวกรอง FIR ที่จะให้ผลตอบสนองเฟสที่เป็นเชิงเส้น ต้องมีเงื่อนไขสมมาตรสำหรับ  $h(n)$  หนึ่งในสี่ชนิด ดังต่อไปนี้

ชนิดที่ 1  $h(n)$  มีความสมมาตรปกติ (symmetric) และ  $N$  เป็นเลขคี่

$$h(n) = h(N-1-n), n=0, 1, \dots, N-1 \quad (2.6)$$

ถ้าให้  $M = (N-1)/2$  ผลตอบสนองเชิงความถี่ของระบบจะอยู่ในรูปของ

$$H(e^{j\omega'}) = H_r(\omega') e^{-j\omega' M} \quad (2.7)$$

$$\text{โดยที่ } H_r(\omega') = h(M) + 2 \sum_{n=0}^{M-1} h(n) \cos[\omega'(M-n)]$$

สังเกตว่า  $H_r(\omega')$  คือส่วนของขนาดซึ่งเป็นค่าจริงเสมอ ส่วน  $-\omega'(N-1)/2$  คือส่วนของเฟสซึ่งเป็นเชิงเส้น ใช้ออกแบบตัวกรองได้ทุกรูปแบบ

ชนิดที่ 2  $h(n)$  มีความสมมาตรปกติ (symmetric) และ  $N$  เป็นเลขคู่

มีสมการเงื่อนไข คือ  $h(n) = h(N-1-n)$  ผลตอบสนองเชิงความถี่จะอยู่ในรูปของ

$$H(e^{j\omega'}) = H_r(\omega') e^{-j\omega'(M+0.5)} \quad (2.8)$$

$$\text{โดยที่ } H_r(\omega') = 2 \sum_{n=0}^{M-1} h(n) \cos[\omega'(M+0.5-n)]$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พบว่าผลตอบสนองทางขนาดของชนิดนี้มีข้อจำกัดว่า ที่  $\omega' = \pi$  จะได้  $H_r(\omega') = 0$  เสมอ ดังนั้น จึงไม่เหมาะสำหรับออกแบบตัวกรองผ่านสูง และตัวกรองผ่านตัดความถี่

ชนิดที่ 3  $h(n)$  มีความสมมาตรแบบตรงข้าม (antisymmetric) และ  $N$  เป็นเลขคี่

$$h(n) = -h(N-1-n), n=0, 1, \dots, N-1 \quad (2.9)$$

ผลตอบสนองเชิงความถี่จะอยู่ในรูปของ

$$H(e^{j\omega'}) = H_r(\omega')e^{j(\omega' M + \pi/2)} \quad (2.10)$$

$$\text{โดยที่ } H_r(\omega') = 2 \sum_{n=0}^{M-1} h(n) \sin[\omega'(M-n)]$$

พบว่าผลตอบสนองทางขนาดของชนิดนี้มีข้อจำกัดว่า ที่  $\omega' = 0$  หรือ  $\pi$  จะได้  $H_r(\omega') = 0$  เสมอ จึงไม่เหมาะสำหรับออกแบบตัวกรองผ่านต่ำ, ผ่านสูง, และตัวกรองตัดแถบความถี่

ชนิดที่ 4  $h(n)$  มีความสมมาตรแบบตรงข้าม (antisymmetric) และ  $N$  เป็นเลขคู่ มีสมการเงื่อนไข คือ  $h(n) = -h(N-1-n)$  ผลตอบสนองเชิงความถี่จะอยู่ในรูปของ

$$H(e^{j\omega'}) = H_r(\omega')e^{j(\omega' (M+0.5) + \pi/2)} \quad (2.11)$$

$$\text{โดยที่ } H_r(\omega') = 2 \sum_{n=0}^{M-1} h(n) \sin[\omega'(M+0.5-n)]$$

พบว่าผลตอบสนองทางขนาดของชนิดนี้มีข้อจำกัดว่า ที่  $\omega' = 0$  จะได้  $H_r(\omega') = 0$  เสมอ ดังนั้น จึงไม่เหมาะสำหรับออกแบบตัวกรองผ่านต่ำ, และตัวกรองตัดแถบความถี่

ตัวกรอง FIR ชนิดที่ 3 และ 4 เหมาะสำหรับออกแบบตัวกรองอนุพันธ์ (differentiator) และตัวกรองฮิลเบิร์ต (Hilbert)

### 2.5.2 การออกแบบโดยวิธีหน้าต่าง (Window Method)

เป็นวิธีพื้นฐานที่สุดวิธีหนึ่งที่จะใช้หาสัมประสิทธิ์ของตัวกรอง เป็นวิธีที่ง่ายต่อการออกแบบ และสามารถใช้ออกแบบตัวกรองแบบต่าง ๆ ได้

เราจะเริ่มออกแบบ โดยใช้ต้นแบบจากตัวกรองอุดมคติ ผลตอบสนองเชิงความถี่ของตัวกรองอุดมคติทั้งสี่ มีความถี่ตัด (cutoff frequency) เท่ากับ  $\omega'_c$  สำหรับแบบผ่านต่ำ และผ่านสูง

ส่วนแบบผ่านแถบความถี่ และตัดแถบความถี่มีความถี่ตัดของแถบความถี่ที่  $\omega'_l$  และ  $\omega'_h$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาดเห็นไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้  $d(n)$  แทนผลตอบสนองต่ออิมพัลส์และ  $D(e^{j\omega'})$  แทนผลตอบสนองเชิงความถี่ สามารถหา  $d(n)$  สำหรับตัวกรองแบบใดๆ ได้โดยใช้ IDTFT กระทำกับผลตอบสนองเชิงความถี่ของตัวกรองนั้น ๆ ตารางที่ 2.1 ได้สรุปค่าของ  $d(n)$  สำหรับตัวกรองแบบต่าง ๆ ไว้ เพื่อใช้อ้างอิงในการออกแบบต่อไป

ชนิดของตัวกรอง	$d(n), -\infty < n < \infty$	$d(0)$
ผ่านต่ำ(LPF)	$(\sin(\omega'_c n)) / \pi n$	$\omega'_c / \pi$
ผ่านสูง(HPF)	$\delta(n) - (\sin(\omega'_c n)) / \pi n$	$1 - (\omega'_c / \pi)$
ผ่านแถบความถี่ (BPF)	$(\sin(\omega'_b n) - \sin(\omega'_s n)) / \pi n$	$(\omega'_b / \pi) - (\omega'_s / \pi)$
ตัดแถบความถี่ (BSF)	$\delta(n) - (\sin(\omega'_b n) - \sin(\omega'_s n)) / \pi n$	$1 - (\omega'_b / \pi) + (\omega'_s / \pi)$

ตารางที่ 2.1 ผลตอบสนองต่ออิมพัลส์ของตัวกรองอุดมคติต่าง ๆ

เราอาจหา  $d(n)$  ของตัวกรองแบบอื่น ๆ ได้จากตัวกรองผ่านต่ำและตัวกรองแบบผ่านทุกความถี่ ตัวกรองแบบผ่านทุกความถี่ในที่นี้คือ  $D(e^{j\omega'})=1$  หรือ  $d(n)=\delta(n)$  แต่อย่างไรก็ตาม  $d(n)$  ที่ได้นี้ไม่สามารถสร้างได้จริงในทางปฏิบัติ เราจะต้องใช้เทคนิควิธีหน้าต่างให้นำ  $d(n)$  เหล่านี้มาใช้ได้

ผลตอบสนองเชิงความถี่ที่เราจะออกแบบ ค่าที่จะใช้กำหนดเป็นคุณลักษณะเฉพาะของวิธีหน้าต่าง ได้แก่

ความพลีวของแถบผ่าน (pass-band ripple,  $\delta_{pass}$ ) คือ ค่าสูงสุดที่ขนาดของแถบผ่าน แกว่งออกห่างจากค่า 1 บางครั้งวัดเป็น dB โดยใช้

$$A_{pass} = 20 \log (1 + \delta_{pass}) / (1 - \delta_{pass}) \quad (\text{dB}) \quad (2.12)$$

การลดทอนของแถบหยุด (stop-band attenuation,  $A_{stop}$ ) คือ จำนวนเท่าที่แถบหยุดลดทอนลงจากค่า 1 วัดเป็น dB โดยมีความสัมพันธ์กับความพลีวของแถบหยุด คือ

$$A_{stop} = -20 \log \delta_{stop} \quad (\text{dB}) \quad (2.13)$$

ความกว้างของแถบเปลี่ยน (transition band width,  $\Delta f'$ )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่ตัด (cutoff frequency,  $f_c$ ) คือ ค่าความถี่ที่ขนาดลดลงประมาณ 0.5 หรืออยู่ที่ประมาณครึ่งหนึ่งของแถบเปลี่ยน นิยามนี้ต่างจากความถี่ตัดของตัวกรองอนาลอก และตัวกรอง IIR ซึ่งความถี่ตัดหมายถึง ความถี่ที่ลดทอนลงเท่ากับ 3 dB

การออกแบบโดยวิธีหน้าต่าง แบ่งเป็นขั้นตอนย่อยได้ดังนี้

1. ใช้ค่าความพลีของแถบผ่าน หรือการลดทอนของแถบหยุดอย่างใดอย่างหนึ่งเพื่อเลือกชนิดของหน้าต่างที่สามารถใช้ได้จากตารางที่ 2.2 ถ้ามีข้อกำหนดทั้งสองอย่าง ให้เปลี่ยน  $A_{stop}$  เป็น  $\delta_{stop}$  ก่อน แล้วเปรียบเทียบ  $\delta_{pass}$  กับ  $\delta_{stop}$  ว่าค่าใดน้อยกว่ากัน ถ้าตัวใดน้อยกว่าให้ใช้ตัวนั้นเป็นตัวเลือกหน้าต่าง โดยมีหลักการว่า  $\delta_{pass}$  ในตารางต้องน้อยกว่าที่ต้องการ และ  $A_{stop}$  ในตารางต้องมากกว่าที่ต้องการ

2. ใช้ค่าความกว้างของแถบเปลี่ยนหาค่าอันดับของตัวกรองที่ต้องใช้ โดยใช้ความสัมพันธ์ระหว่าง  $\Delta f'$  กับ  $N$  ที่แสดงไว้ในตารางที่ 2.2 จากนั้น คำนวณฟังก์ชันหน้าต่างที่ต้องใช้ จะได้

$$w(n), \text{ ที่ } n=0, 1, \dots, N-1$$

จากตารางจะสังเกตเห็นได้ว่า อันดับของตัวกรองจะเป็นสัดส่วนผกผันกับ  $\Delta f'$  นั่นคือ เราสามารถปรับปรุง  $\Delta f'$  ให้แคบลงได้ (ตัวกรองคมขึ้น) โดยการเพิ่มค่า  $N$  แต่เราไม่สามารถปรับปรุงค่า  $\delta_{stop}$  และ  $A_{pass}$  ให้ดีขึ้นได้ เพราะมีขนาดคงที่สำหรับหน้าต่างแบบหนึ่ง ๆ ยกเว้นหน้าต่างแบบ Kaiser

3. ใช้ชนิดของตัวกรองและความถี่ตัดที่ต้องการ เลือกผลตอบสนองต่ออิมพัลส์  $d(n)$  ที่ถูกต้องจากตารางที่ 2.1

4. เลื่อน  $d(n)$  ให้ล้าหลังลง  $M$  ตำแหน่ง โดย  $M=(N-1)/2$  จะได้สัญญาณเป็น  $d(n-M)$  จากนั้นคูณเข้ากับฟังก์ชันหน้าต่าง  $w(n)$  ที่ได้จากข้อ 2 ซึ่งจะได้เป็นผลตอบสนองต่ออิมพัลส์ที่มีความยาว  $N$  จุด และเป็นแบบคอซัล ดังนี้

$$h(n) = d(n - M)w(n), n = 0, 1, \dots, N-1 \quad (2.14)$$

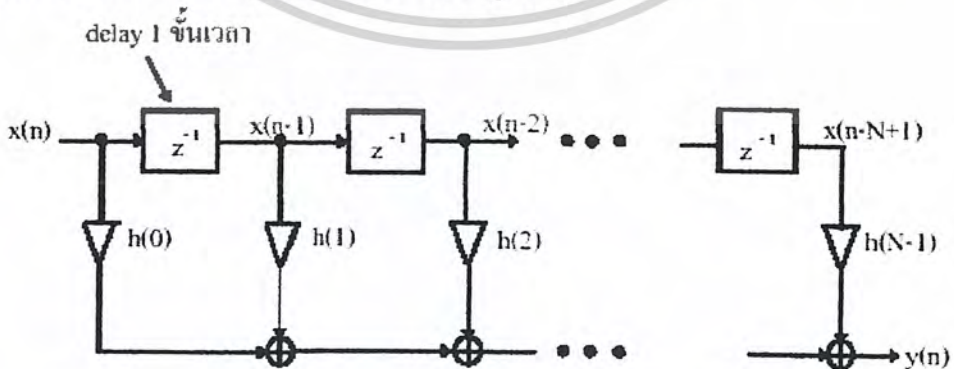
การออกแบบนี้จะต้องใช้  $N$  เป็นจำนวนคู่ ซึ่งจะได้เป็นตัวกรอง FIR ที่มีเฟสเชิงเส้น และมีสมมาตรชนิดที่ 1 เพราะทั้ง  $d(n)$  และ  $w(n)$  มีสมมาตรรอบจุดกึ่งกลางทั้งคู่ สำหรับการออกแบบโดยที่  $N$  เป็นจำนวนคี่ก็ทำได้เช่นกัน

หน้าต่าง	$\delta_{pass}$	$A_{stop}$ $20\log \delta_{stop}$ (dB)	$\Delta f'$ (normalized)	$w(n), n = 0, 1, \dots, N-1$ $M=(N-1)/2$
สี่เหลี่ยม (rectangular)	8.9%	21	$2/N$	1
ฮานนิง (Hanning)	0.63%	44	$4/N$	$0.5 - 0.5\cos(2\pi n/N - 1)$
แฮมมิง (Hamming)	0.22%	53	$4/N$	$0.54 - 0.46\cos(2\pi n/N - 1)$
แบล็กแมน (Blackman)	0.02%	74	$6/N$	$0.42 - 0.5\cos(2\pi n/N - 1)$ $+ 0.08 \cos(4\pi n/N - 1)$
ไคเซอร์ (Kaiser)	ปรับได้	ปรับได้	$(A - 7.95) / 14.36(N-1)$	$I_0(\alpha \sqrt{1 - (n - M)^2 / M^2}) / I_0(\alpha)$

ตารางที่ 2.2 วิธีหน้าต่างแบบต่าง ๆ และค่าพารามิเตอร์ที่สำคัญ

**2.5.3 วิธีสร้างตัวกรอง FIR (FIR Filter Realization)**

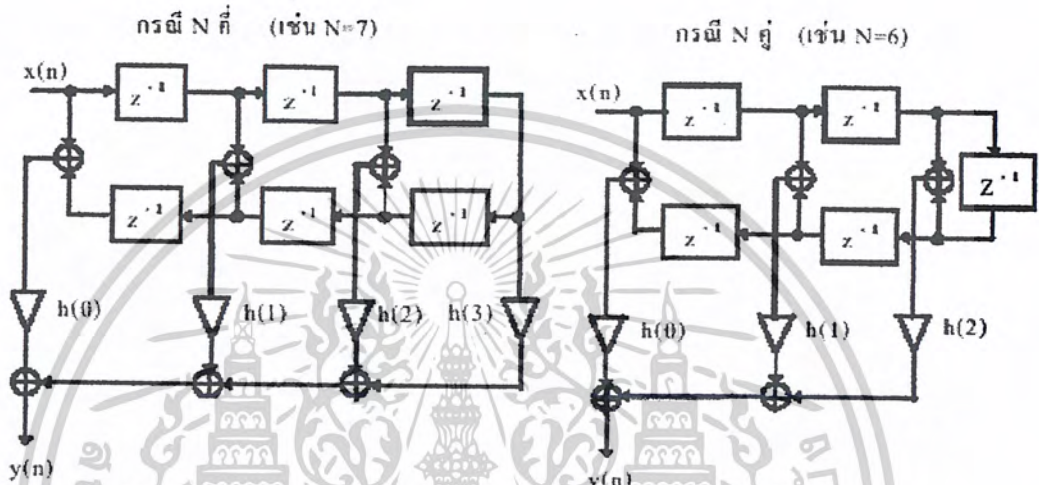
คำว่า realization หมายถึง การนำเอาสิ่งที่ได้ออกแบบแล้ว หรือปรากฏการณ์ทางทฤษฎีไปประยุกต์เป็นอุปกรณ์ที่ใช้งานได้จริง ๆ ขึ้นมา เราจะใช้ผลตอบสนองต่ออิมพัลส์หรือ  $h(n)$  เพื่อสร้างตัวกรอง ซึ่งกระบวนการของตัวกรองในที่นี้ก็คือ การทำคอนโวลูชันระหว่าง  $h(n)$  และสัญญาณขาเข้า  $x(n)$  นั่นเอง ซึ่งสามารถเขียนเป็นแผนภาพได้ดัง รูปที่ 2.1



รูปที่ 2.1 แผนภาพการสร้างตัวกรองแบบ FIR โดยคอนโวลูชันปกติ

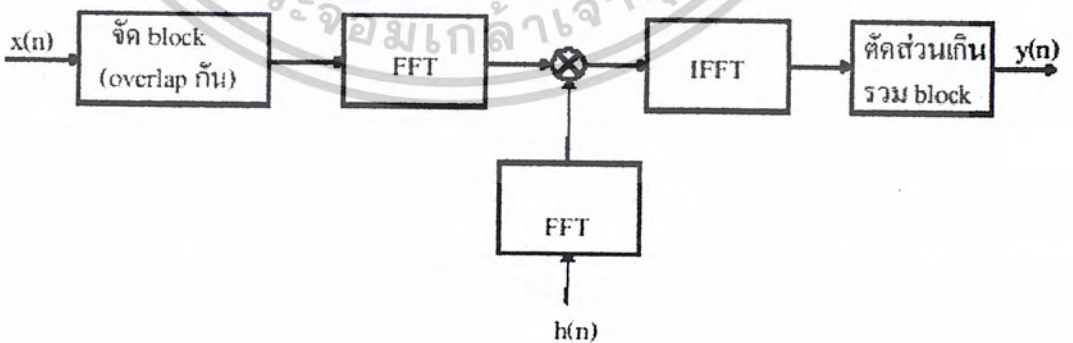
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากแผนภาพนี้เราสามารถนำไปประยุกต์เขียนเป็น โปรแกรมคอมพิวเตอร์ หรือทำเป็น ฮาร์ดแวร์พิเศษเพื่อทำหน้าที่ตัวกรองแบบ FIR โดยตรง ในกรณีที่ตัวกรองเป็นแบบเฟสเชิงเส้น ซึ่ง หมายถึง  $h(n)$  จะมีสมมาตร ณ จุดกึ่งกลาง เราอาจใช้คุณสมบัตินี้ลดโครงสร้างของ FIR ให้เล็กลงได้ ดังแสดงในรูปที่ 2.2



รูปที่ 2.2 แผนภาพการสร้างตัวกรองแบบ FIR เมื่ออาศัยคุณสมบัติการสมมาตร

และโดยอาศัยวิธีคอนโวลูชันแบบเร็ว ก็สามารถใช้แทนคอนโวลูชันปกติเพื่อใช้สร้างตัวกรองแบบ FIR ได้ ดังในรูปที่ 2.3



รูปที่ 2.3 แผนภาพการสร้างตัวกรองแบบ FIR โดยวิธีคอนโวลูชันแบบเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### สัญญาณเสียงดิจิทัล

ในอดีตสัญญาณเสียงต่างๆจะถูกบันทึกและทำการประมวลผลในรูปของอนาลอก แต่ในปัจจุบัน เทคโนโลยีสัญญาณเสียงดิจิทัลได้เข้ามามีบทบาทในเรื่องของเสียง และการบันทึกเสียงอย่างสูง การรักษาสัญญาณให้คงรูปอยู่เสมอ ไม่ว่าจะเปลี่ยนตัวกลางที่เก็บสัญญาณไปเป็นอย่างไร หรือมีระยะการเดินทางไกลแค่ไหน ซึ่งเป็นข้อดีของระบบดิจิทัลที่ทำให้มันเข้ามาแทนที่ระบบอนาลอก และการที่เราสามารถทำสัญญาณเสียงให้อยู่ในรูปดิจิทัลได้ ทำให้เราสามารถ สร้างและแก้ไข คัดแปลงสัญญาณนั้นได้ง่ายโดยการช่วยเหลือจากคอมพิวเตอร์

สัญญาณเสียงในรูปแบบอนาลอกสามารถทำให้เป็นสัญญาณดิจิทัลได้ โดยวิธีการสุ่มสัญญาณและการแบ่งขั้นสัญญาณ ดังที่กล่าว ไปแล้วในบทที่ 2

โดยอัตราในการสุ่มสำหรับสัญญาณเสียง ซึ่งทำให้เราสามารถฟังเสียงที่มีความถี่ได้ถึง 20,000 Hz จะสุ่มในอัตรการสุ่ม 44 kHz ซึ่งความถี่เสียงที่คนเราสามารถได้ยินจะอยู่ที่ระดับ 20 ถึง 20,000 Hz

ในการกำหนดระดับขั้นของสัญญาณ ถ้าหากเรากำหนดแบบ 8 บิต เราจะกำหนดระดับแอมพลิจูดได้ถึง 256 ระดับ ซึ่งจะเพียงพอสำหรับเสียงการพูดคุย แต่สำหรับเสียงเพลง มาตรฐานจะอยู่ที่ 16 บิต ซึ่งจะสามารถกำหนดความดังค่อย ได้ถึง 65,000 ระดับทีเดียว

#### 3.1 มาตรฐาน MIDI

MIDI ย่อมาจาก Musical Instrument Digital Interface เป็นระบบมาตรฐานสำหรับการสื่อสารข้อมูลระหว่างเครื่องดนตรีหรือมาตรฐานสำหรับการเข้ารหัสเสียงจากเครื่องดนตรีให้มาอยู่ในรูปดิจิทัล ซึ่งทำให้คอมพิวเตอร์ เครื่องดนตรี ซินธิไซเซอร์(เครื่องสังเคราะห์เสียงโดยใช้ระบบอิเล็กทรอนิกส์)สามารถติดต่อกันได้

#### 3.2 โปรแกรม Max/MSP

Max คือ โปรแกรมด้านเสียงเพลงและมัลติมีเดียแบบกราฟฟิกสำหรับผู้ที่ประสบกับขีดจำกัดของโปรแกรมเสียงปกติสำหรับอุปกรณ์ MIDI

โดยจะมีรูปแบบการเขียนโปรแกรมประยุกต์ (Application) โดยใช้วัตถุ (object) ซึ่งจะมีลักษณะเป็นกล่องที่สามารถนำมาเชื่อมต่อเข้าด้วยกันด้วยเส้นเชื่อมโยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ความสามารถของ Max

Max ช่วยควบคุมอุปกรณ์ได้ทุกอย่างตามที่ต้องการ โดยสามารถสร้างโปรแกรมประยุกต์ (Application) สำหรับการแต่งเพลง ปรับปรุงทำนองเพลง เตรียมดนตรีคอร์ดให้ในขณะที่คุณกำลังเล่นดนตรี ส่งคำสั่งไปยังเครื่องซินธิไซเซอร์ แก้ไข patch ของ ซินธิไซเซอร์ ทำทุกอย่างได้ด้วยการคลิกเมาส์และ MIDI เพราะ Max เปลี่ยนข้อมูลควบคุมทั้งหมดเป็นรูปแบบกระแสตัวเลข

Max จัดเตรียมภาษาโปรแกรมขั้นสูงแบบกราฟฟิคซึ่งนำเอาความสามารถทางกราฟฟิคและส่วนติดต่อกับผู้ใช้ในรูปแบบสัญลักษณ์ (icon) โปรแกรมจะถูกเขียนโดยใช้วัตถุแบบกราฟฟิคมากกว่าข้อความ ซึ่งช่วยลดการเรียนรู้คำสั่งและซินแทกซ์ (syntax) และมีแนวทางในการเขียนโปรแกรมที่ง่ายและชัดเจน โดยการเชื่อมโยงวัตถุเข้าด้วยกัน

Max ดูแลการโปรแกรมในระดับต่ำให้ มันจะดำเนินการติดต่อกับ MIDI และ โปรโตคอลการสื่อสารอื่น และทำการคำนวณทางตรรกะให้ตามเวลาที่คุณต้องการในอนาคต

โปรแกรมประยุกต์ที่สร้างด้วย Max สามารถทำงานได้ในเวลาจริง Max ทำให้คุณสามารถเขียนโปรแกรมที่สร้างเสียงเพลงได้ในทันทีทันใด หรือแก้ไขประสิทธิภาพของเสียงได้ในทันที

Max ใช้หลักการโปรแกรม ภาษา C โดย Max ได้จัดเตรียมภาษาระดับสูงแบบกราฟฟิค ซึ่งใช้งานง่ายสำหรับผู้ที่ยังไม่เคยกับภาษาการเขียน โปรแกรม และผู้ที่ไม่เคยเขียนโปรแกรมมาก่อน และ Max สามารถรวมเข้ากับโปรแกรมภาษาซีที่เขียนขึ้นเองได้ ดังนั้นสิ่งใดที่ต้องการทำแล้ว Max ไม่สามารถทำให้ได้ คุณสามารถเขียนวัตถุ Max ขึ้นมาเองได้โดยใช้ภาษาซี

## บทที่ 4

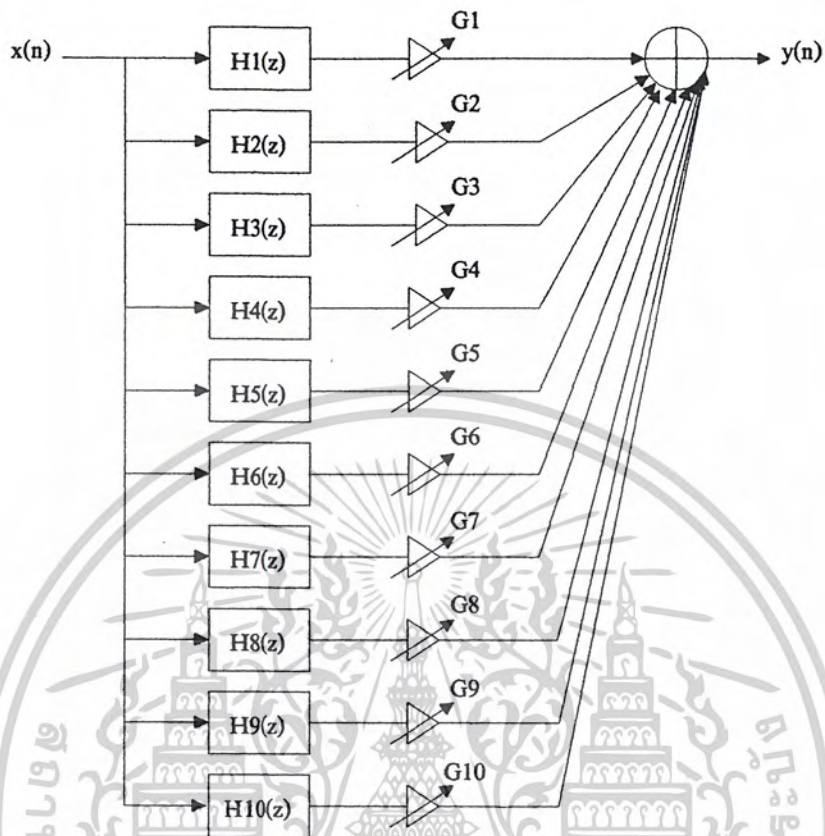
### หลักการออกแบบอีควอไลเซอร์

จากทฤษฎีของการประมวลผลสัญญาณดิจิทัลที่ได้ศึกษามา เราสามารถนำมาประยุกต์ใช้ในการออกแบบอีควอไลเซอร์ได้ ซึ่งจะมีการทดลองผลของการออกแบบด้วยโปรแกรม MATLAB ก่อนจะนำการออกแบบไปสร้างจริง โดยการทดลองนั้นจะสร้าง“m-file” ขึ้นมา โดย m-file นั้นเป็นไฟล์ที่ใช้สำหรับ MATLAB โดยเฉพาะ

อีควอไลเซอร์นั้นมีไว้เพื่อปรับแต่งเสียงในย่านต่าง ๆ ดังนั้นเราจึงต้องกรองสัญญาณออกมาเป็นย่าน ๆ เพื่อนำมาคูณกับเกน(gain) ของแต่ละตัว ก่อนที่จะนำมารวมสัญญาณกันใหม่อีกครั้งเพื่อส่งออกไปใช้งานต่อไป

ในการทดลองนี้เราได้แบ่งออกเป็นสองแบบคือกราฟิกอีควอไลเซอร์ซึ่งจะมีการกำหนดความถี่ตัดที่จุดแบ่งย่านต่างๆ ไว้แล้ว กับพารามตริกอีควอไลเซอร์(Parametric Equalizer) ซึ่งสามารถให้ผู้ใช้งานกำหนดความถี่ตัดของย่านต่างๆเองได้ ในการทดลองจะใช้หลักการออกแบบเดียวกัน แต่ต่างกันตรงที่เราสามารถเปลี่ยนความถี่ตัดได้ ในพารามตริกอีควอไลเซอร์กำหนดให้ความถี่ตัดในแต่ละย่านเป็นดังนี้

- 1)  $H_1(z)$  ผ่านย่านความถี่ 0-2400Hz
- 2)  $H_2(z)$  ผ่านย่านความถี่ 2400-4800 Hz
- 3)  $H_3(z)$  ผ่านย่านความถี่ 4800-7200 Hz
- 4)  $H_4(z)$  ผ่านย่านความถี่ 7200-9600 Hz
- 5)  $H_5(z)$  ผ่านย่านความถี่ 9600-12000 Hz
- 6)  $H_6(z)$  ผ่านย่านความถี่ 12-14.4 KHz
- 7)  $H_7(z)$  ผ่านย่านความถี่ 14.4-16.8 KHz
- 8)  $H_8(z)$  ผ่านย่านความถี่ 16.8-19.2 KHz
- 9)  $H_9(z)$  ผ่านย่านความถี่ 19.2-21.6 KHz
- 10)  $H_{10}(z)$  ผ่านย่านความถี่ 21.6-24 KHz

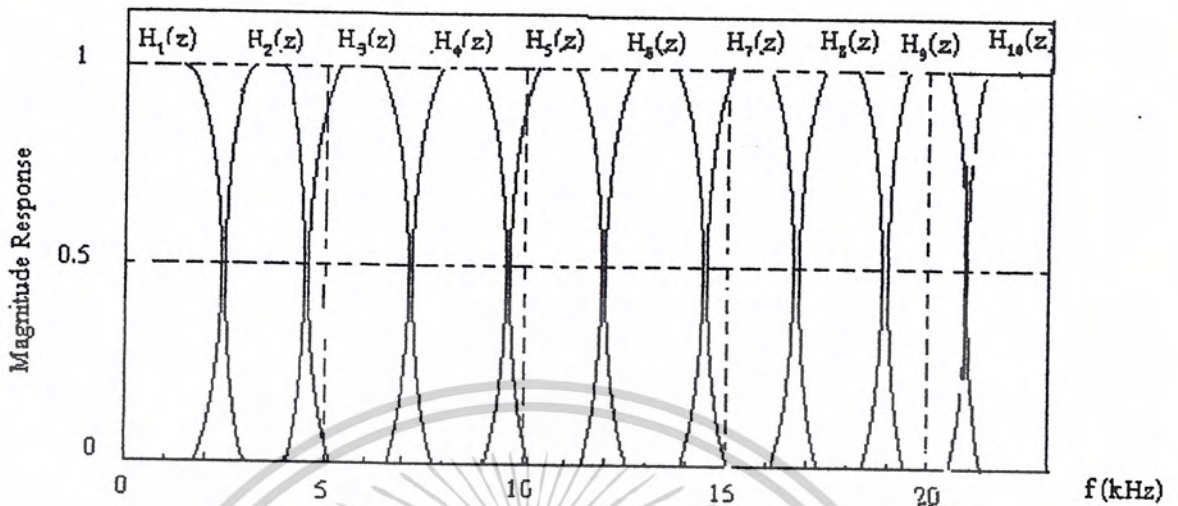


รูปที่ 4.1 แสดง โครงสร้างของ 10 แบนด์

จากรูปที่ 4.1 ฟังก์ชันถ่ายโอนรวมของระบบคือ

$$H_{\text{total}}(z) = G_1H_1(z) + G_2H_2(z) + G_3H_3(z) + G_4H_4(z) + G_5H_5(z) + G_6H_6(z) + G_7H_7(z) + G_8H_8(z) + G_9H_9(z) + G_{10}H_{10}(z)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 ผลตอบสนองของตัวกรองทั้ง 10

ตัวกรองทั้ง 10 ตัวสามารถออกแบบโดยใช้ FIR หรือ IIR ก็ได้ แต่การทดลองนี้เลือกใช้แบบ FIR โดยใช้คำสั่ง FIR1 ใน MATLAB โดยใช้ความถี่ตัดที่ปลายความถี่ดังนี้

$$H = \text{fir1}(N, W_n);$$

โดย H คือ ผลตอบสนองอิมพัลส์

N คือ จำนวน order

$W_n$  คือ ความถี่ตัด โดยจะมีค่าเป็น [L H]

เมื่อ L คือความถี่ตัดที่ปลายความถี่ต่ำ

H คือความถี่ตัดที่ปลายความถี่สูง

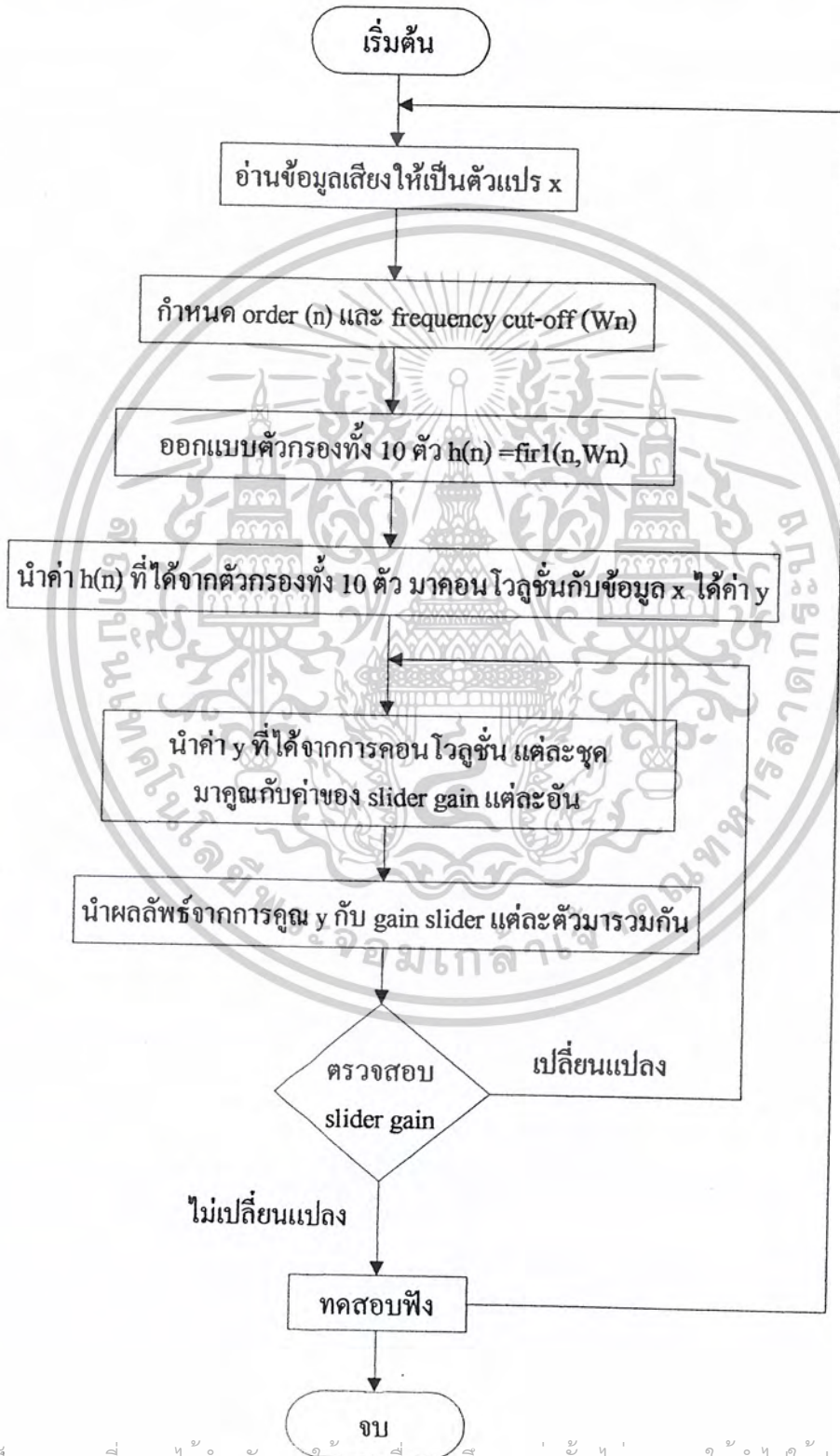
และ  $W_n = 1 = 1/2$  ของ sampling rate

และคำสั่ง FIR1 เป็นการหาผลตอบสนองอิมพัลส์โดยใช้หน้าต่างแบบแฮมมิง

ตัวที่ 1 เป็น LPF ตัวที่ 2-9 เป็น BPF ตัวที่ 10 เป็น HPF

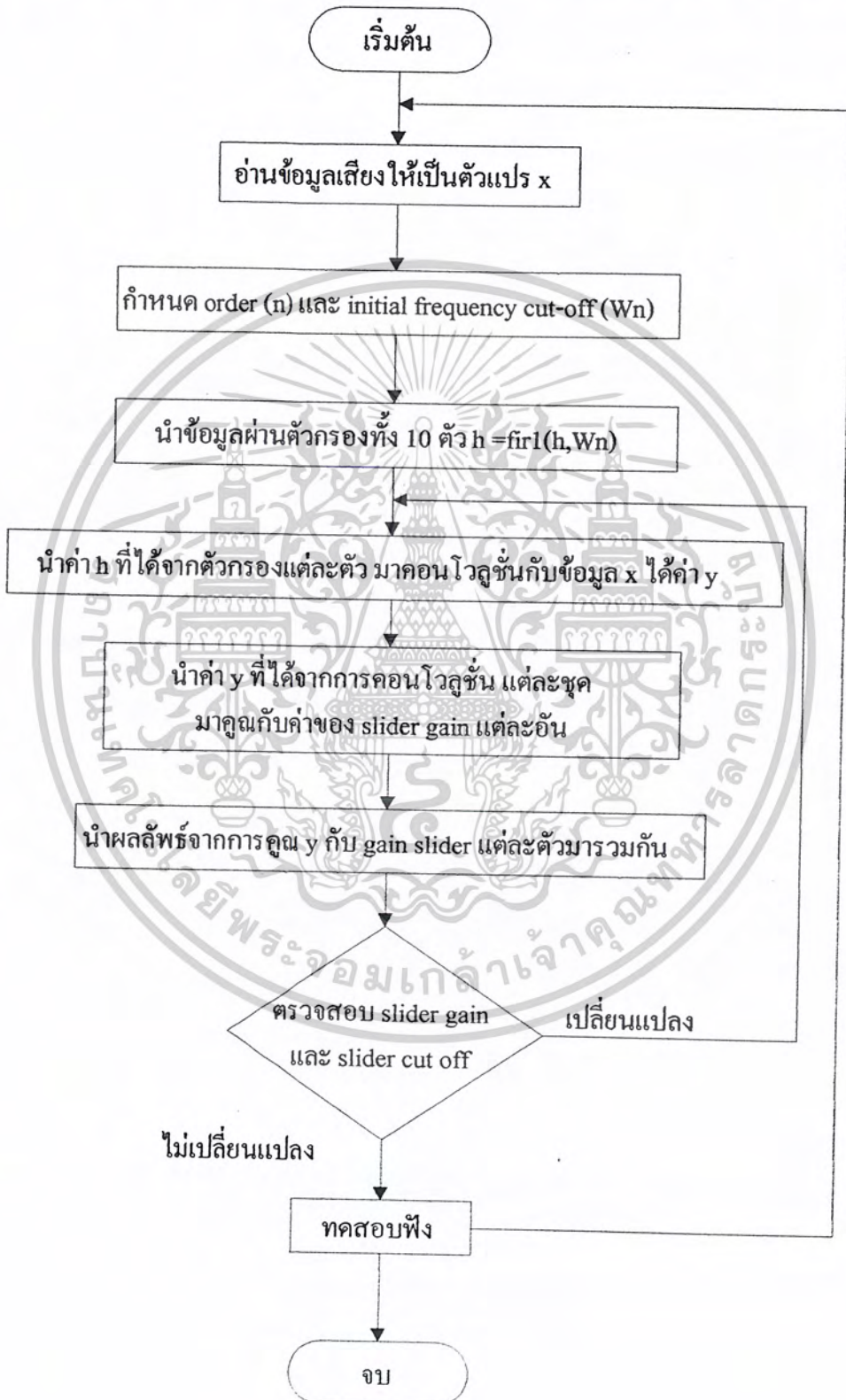
โดยมีขั้นตอนของการทำงานตามนี้

รูป 4.3 แผนภูมิขั้นตอนการดำเนินงานของการจำลองการทำงานของกราฟฟิคอควอไลเซอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 4.4 แผนภูมิขั้นตอนการดำเนินงานการจำลองการทำงานของพารามตริกอีควอไลเซอร์

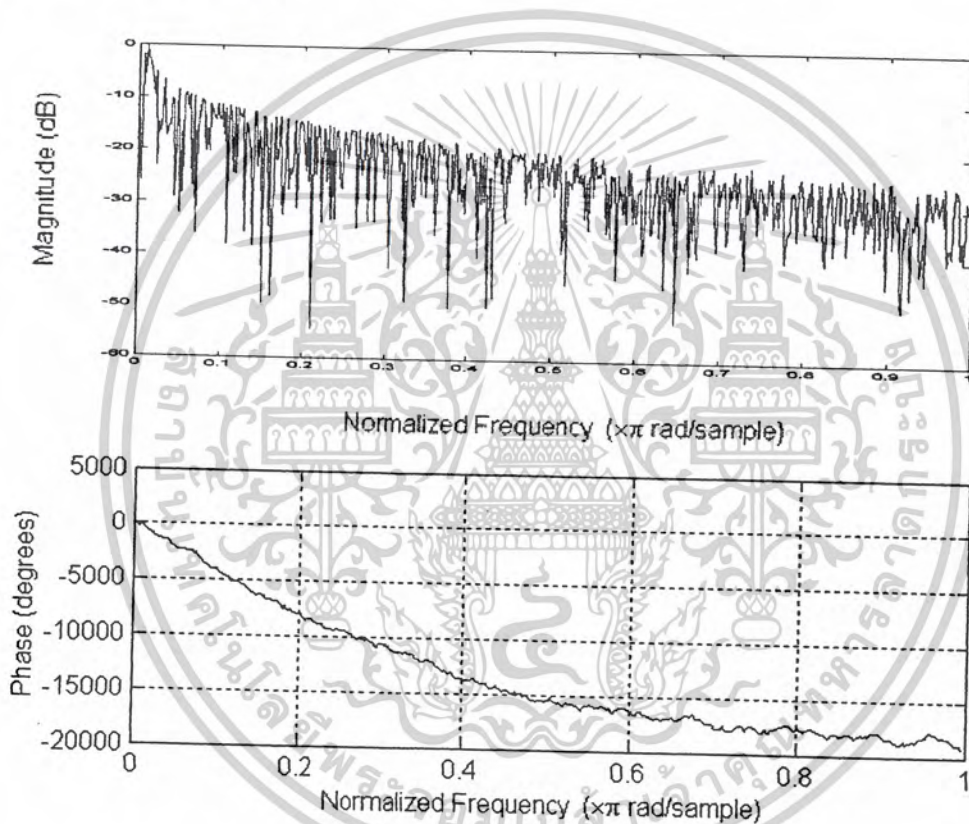


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### ผลการทดลองจากการออกแบบด้วยโปรแกรม MATLAB

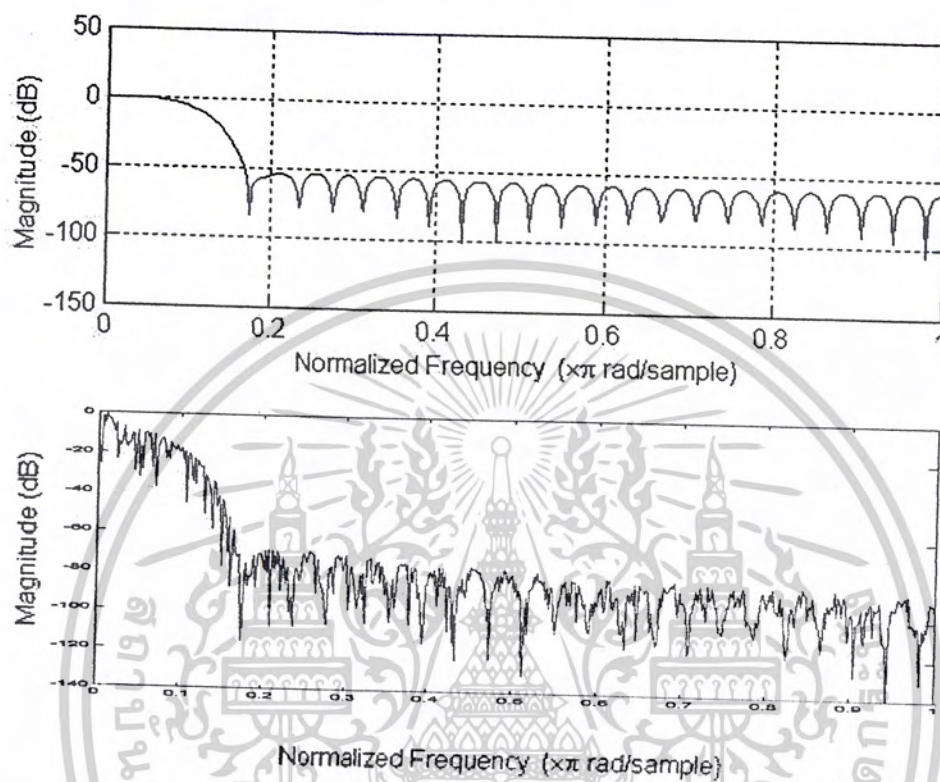
ผลการทดลองจากการออกแบบด้วยโปรแกรม MATLAB ในการทดลองนั้นเพื่อให้ผลที่ออกมาง่ายต่อความเข้าใจ จึงใช้สัญญาณที่กวาดความถี่ตั้งแต่ 20 Hz – 20 KHz โดยมีค่าอัตราการสุ่มเท่ากับ 48 KHz ดังรูป



รูปที่ 5.1 แสดงผลตอบสนองของความถี่ของสัญญาณที่จะใช้ทดสอบ

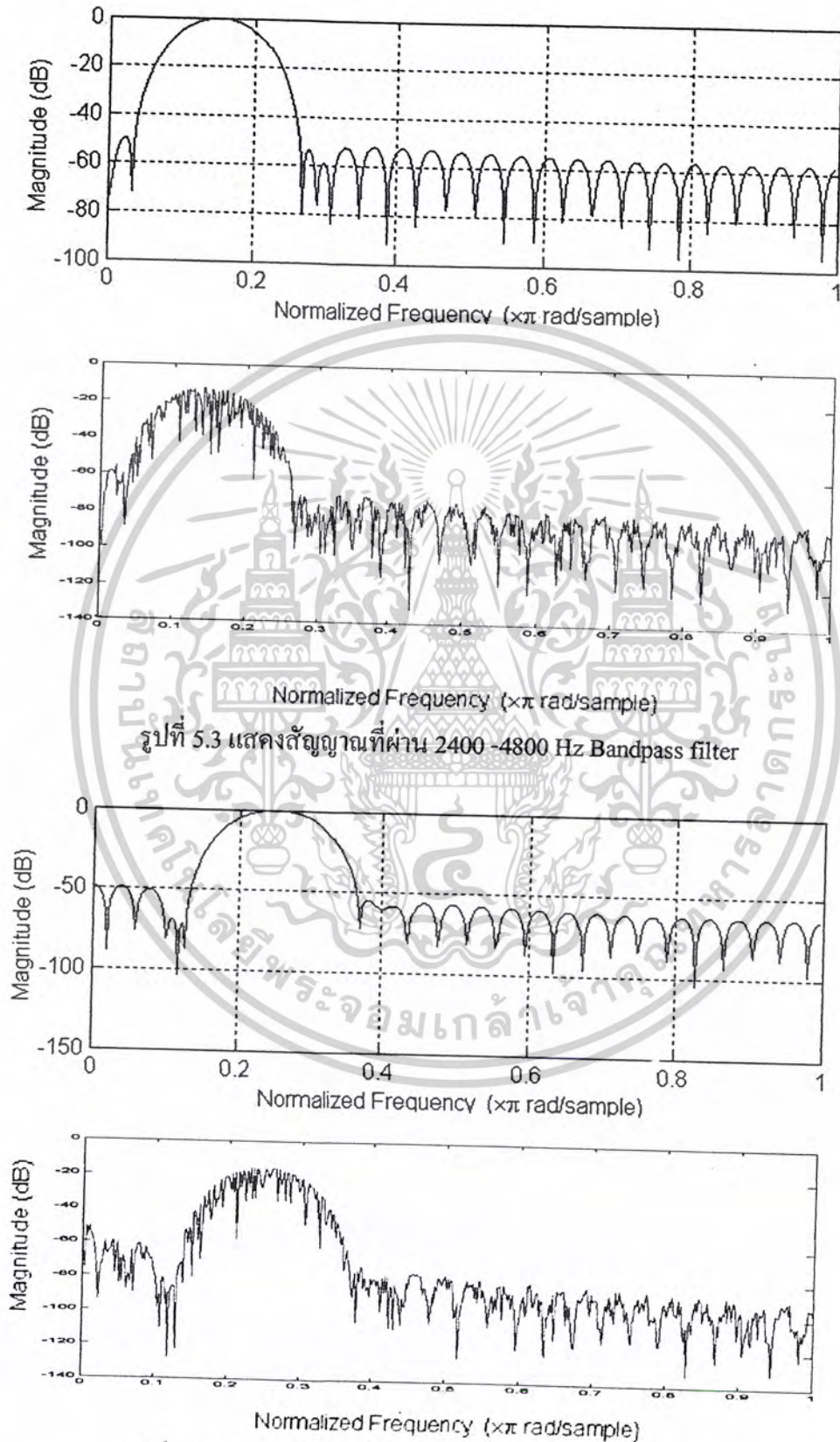
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อมีการจับสัญญาณที่ผ่านตัวกรองจุดต่าง ๆ ก็จะได้ผลตอบสนองตามนี้



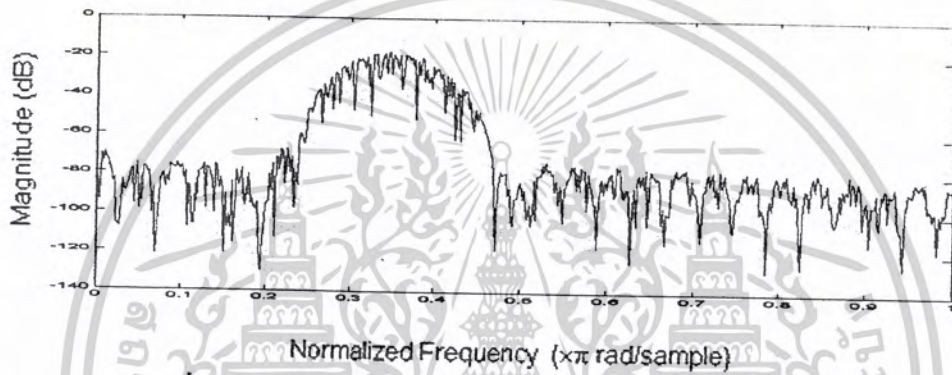
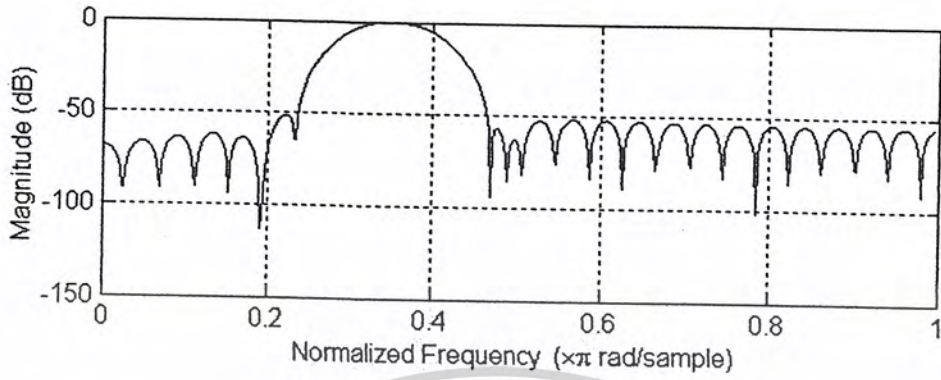
รูปที่ 5.2 แสดงสัญญาณที่ผ่าน 2400 Hz Lowpass filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

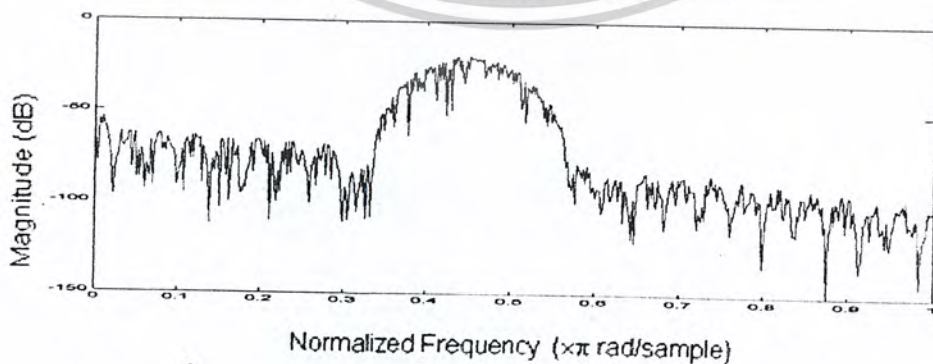
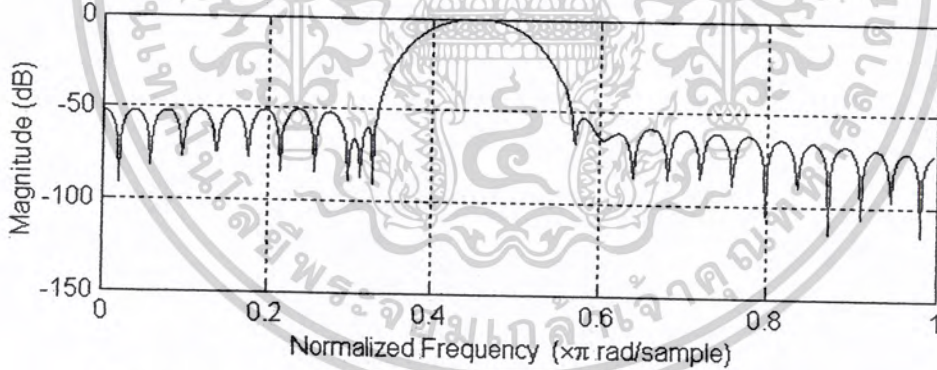


รูปที่ 5.4 แสดงสัญญาณที่ผ่าน 4800-7200 Hz Bandpass filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

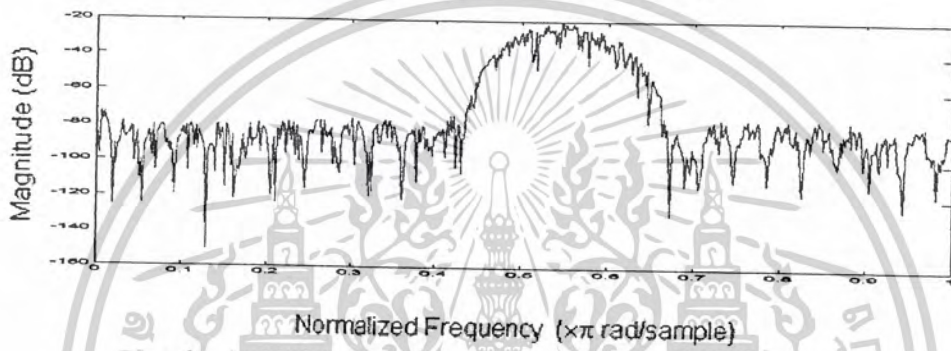
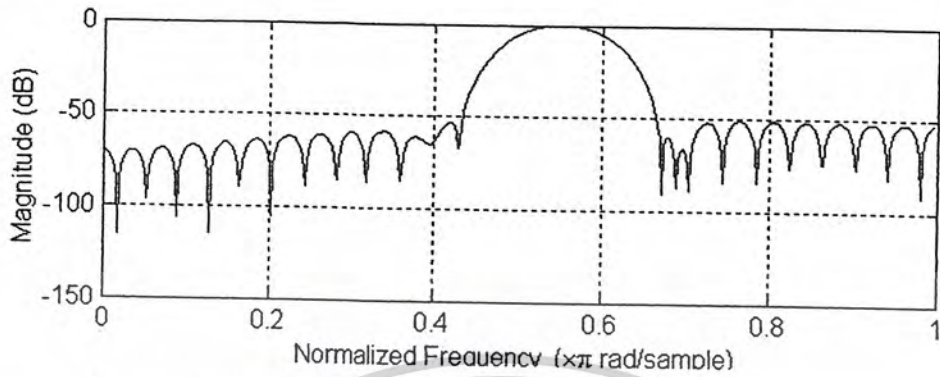


รูปที่ 5.5 แสดงสัญญาณที่ผ่าน 7200-9600Hz Bandpass filter

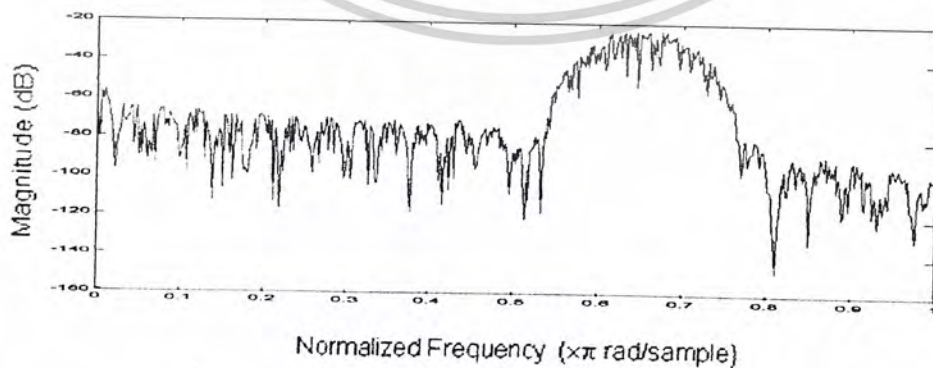
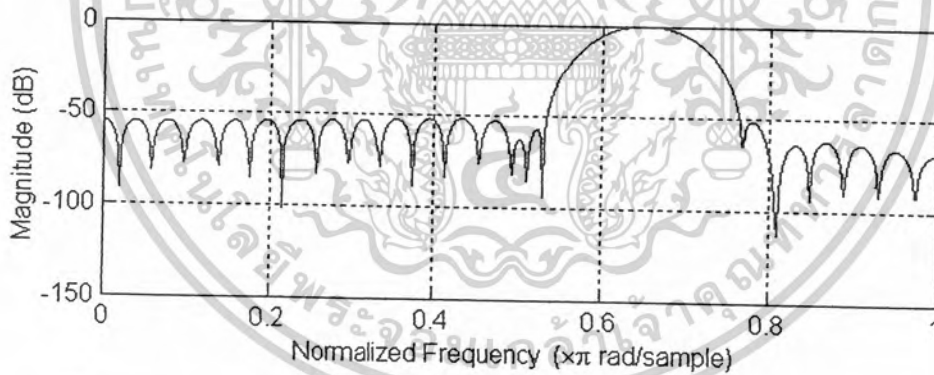


รูปที่ 5.6 แสดงสัญญาณที่ผ่าน 9600-12000 Hz Bandpass filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

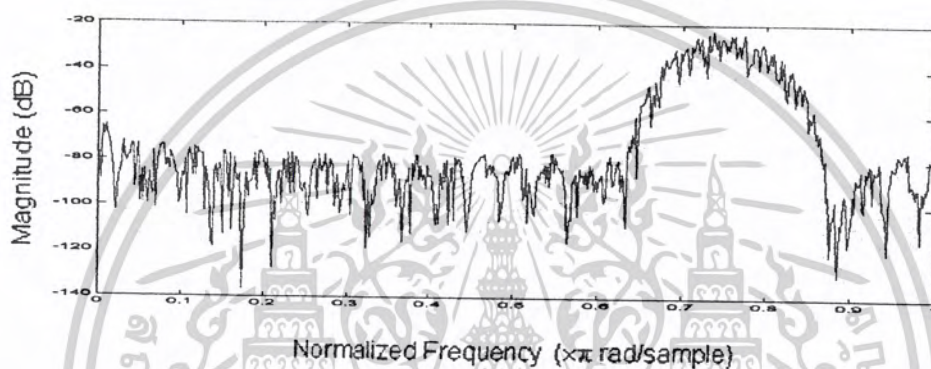
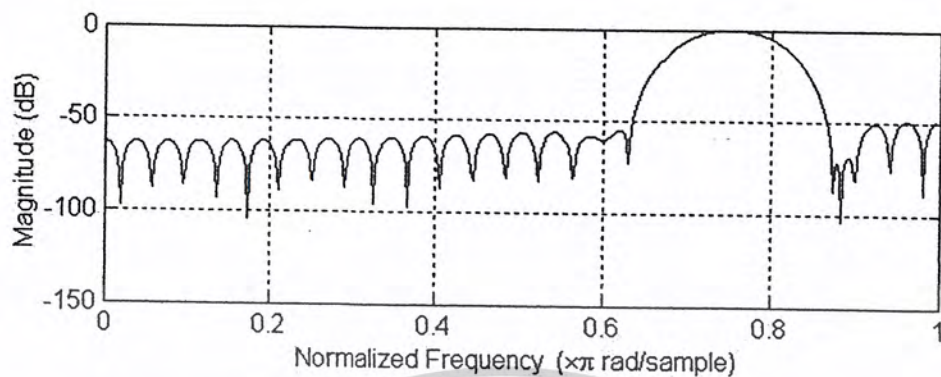


รูปที่ 5.7 แสดงสัญญาณที่ผ่าน 12-14.4 KHz Bandpass filter

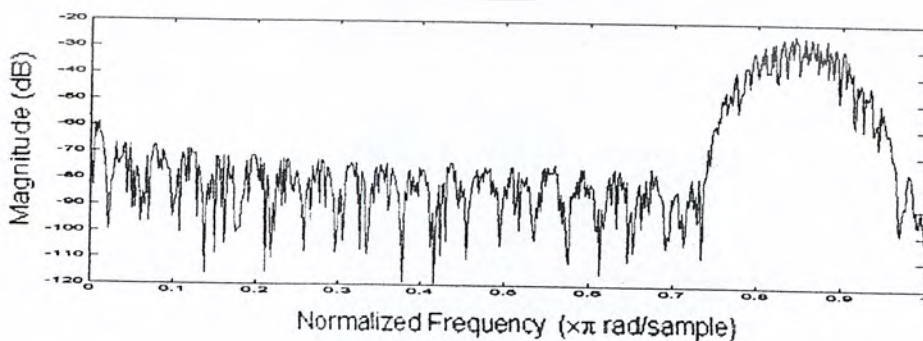
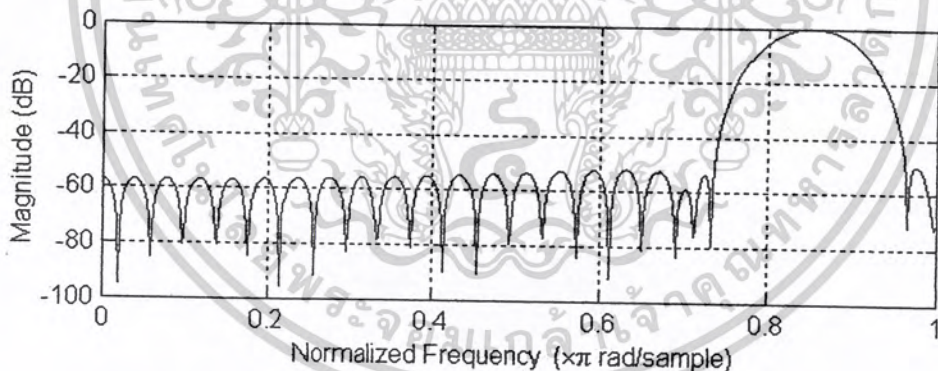


รูปที่ 5.8 แสดงสัญญาณที่ผ่าน 14.4 -16.8 KHz Bandpass filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

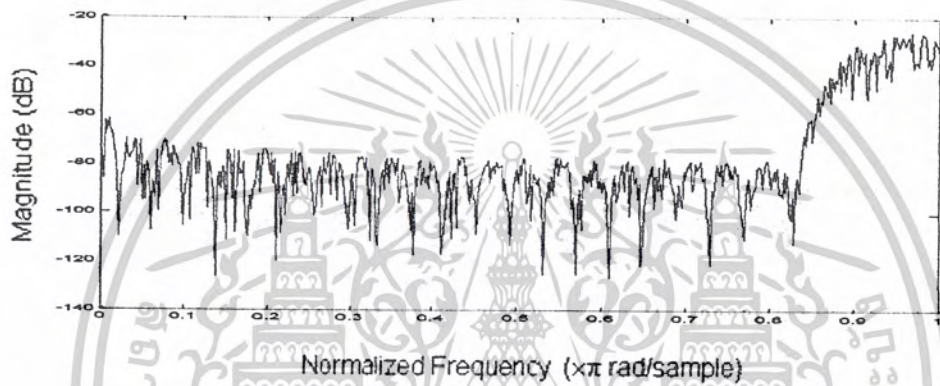
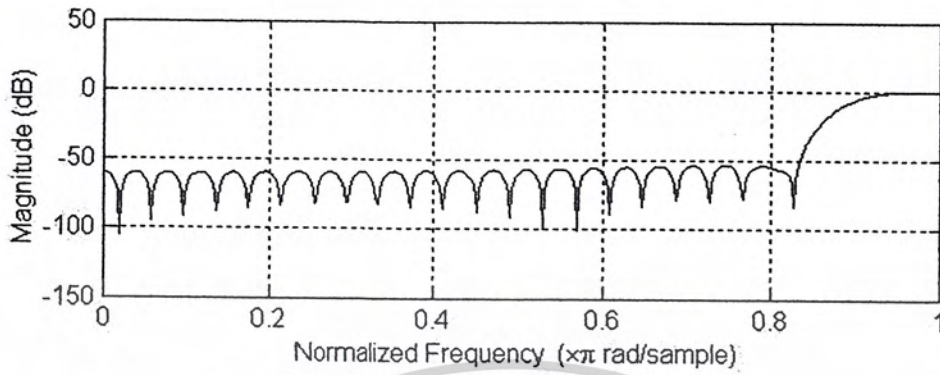


รูปที่ 5.9 แสดงสัญญาณที่ผ่าน 16.8-19.2 KHz Bandpass filter



รูปที่ 5.10 แสดงสัญญาณที่ผ่าน 19.2-21.6 KHz Bandpass filter

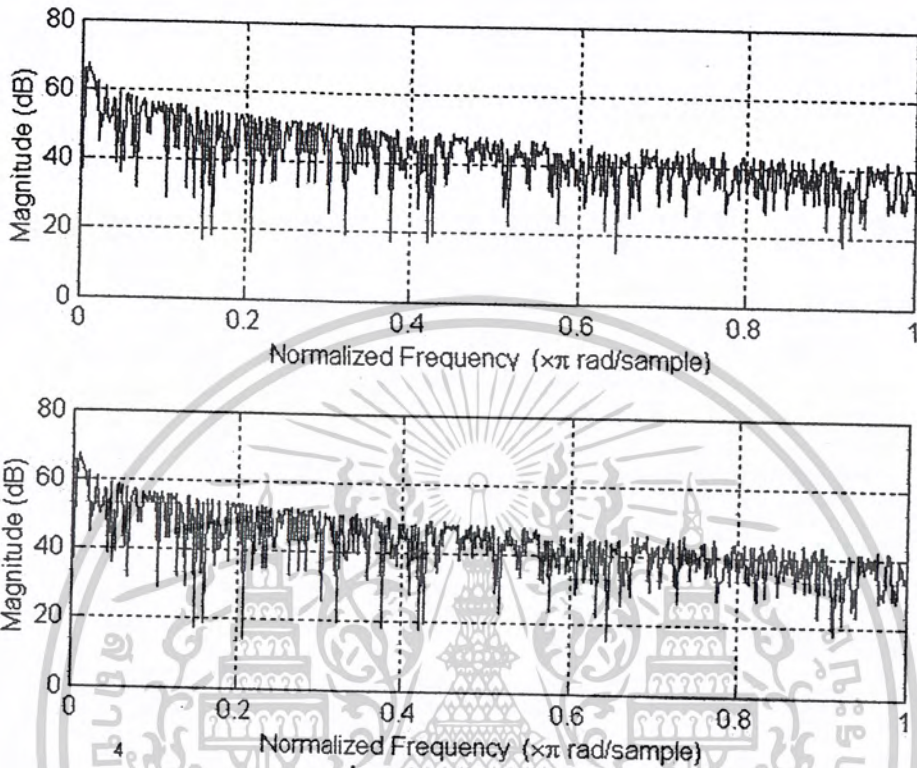
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.11 แสดงสัญญาณที่ผ่าน 21.6-24 KHz Highpass filter

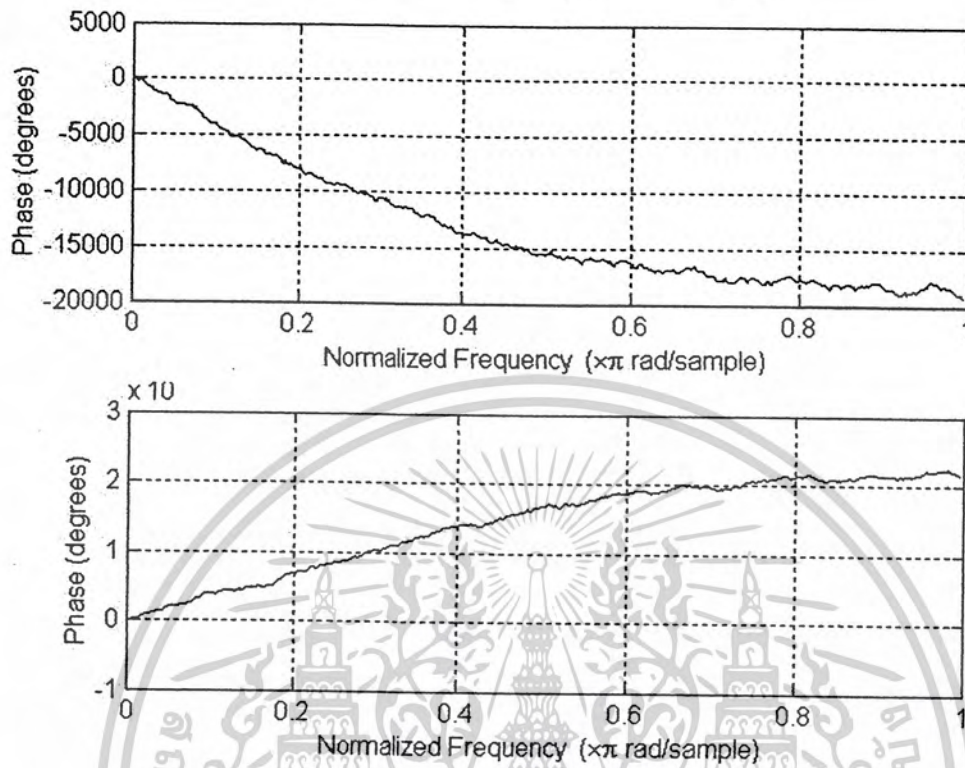
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำสัญญาณที่ได้จากแต่ละตัวกรองมารวมกันก็จะ ได้สัญญาณตามรูป



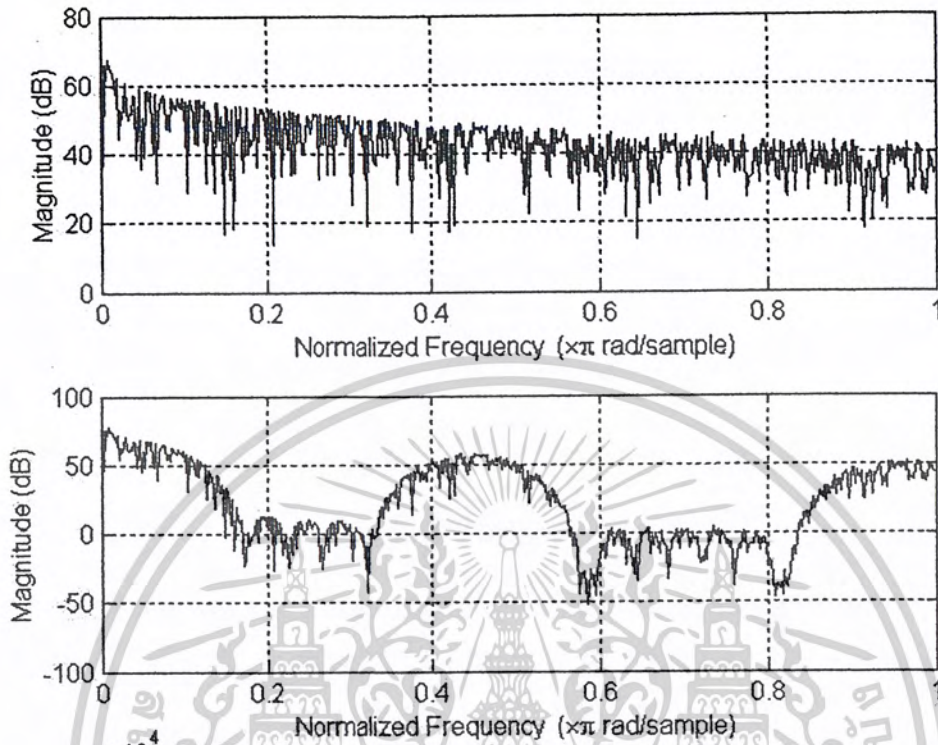
รูปที่ 5.12 แสดงผลตอบสนองความถี่

รูปบนคือสัญญาณเริ่มต้น รูปล่างคือการนำสัญญาณจากแต่ละตัวกรองมารวมกันใหม่



รูปที่ 5.13 แสดงผลตอบสนองทางเฟส

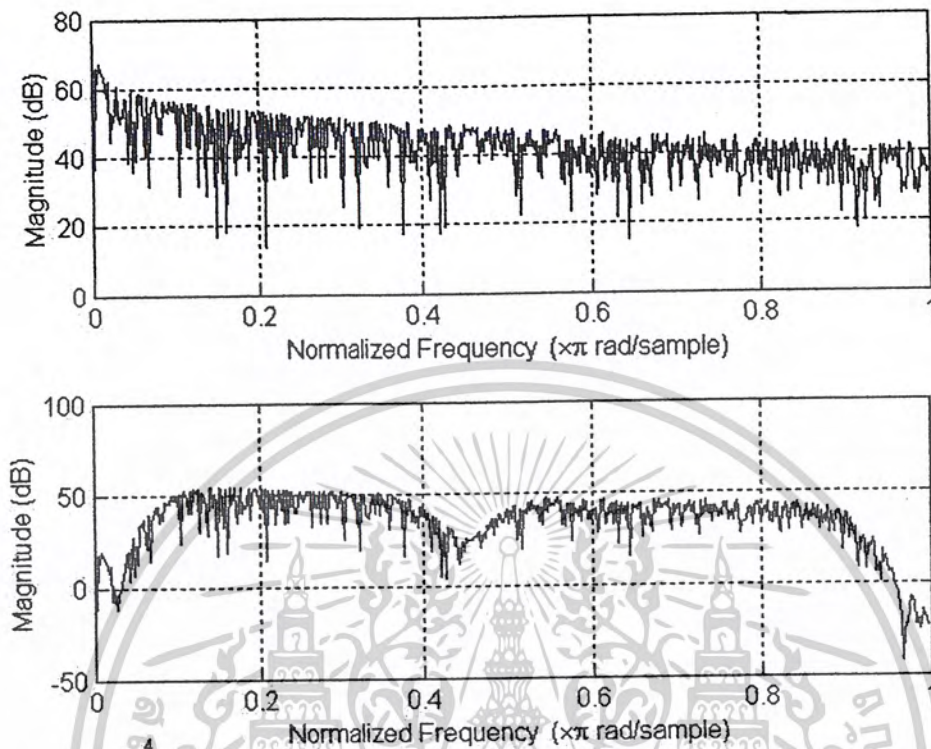
จากรูปที่ 5.13 จะเห็นได้ว่า phase ของสัญญาณเริ่มต้นกับสัญญาณที่ผ่านตัวกรองมีความแตกต่างกัน



รูปที่ 5.14 แสดงผลของการคูณกับค่าเกิน

จากรูป 5.14 รูปบนเป็นสัญญาณเริ่มต้น ขณะที่รูปล่างเกิดจากการ บูลท์ในย่าน 2400 Hz, 9.6-12 KHz และ 21.6-24 KHz และที่เหลือจะตัดทิ้งทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.15 แสดงผลของการคูณกับค่าเกิน

จากรูป 5.15 รูปบนเป็นสัญญาณเริ่มต้น รูปล่างเป็นการตัดย่าน 2400 Hz, 9.6-12 KHz และ 21.6-24 KHz โดยที่เหลือไม่เปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

### การสร้างโปรแกรมประยุกต์

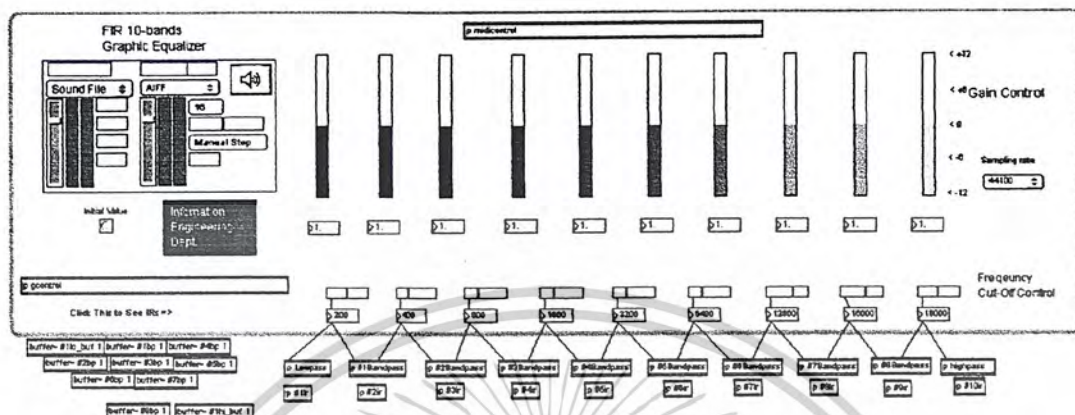
โปรแกรมประยุกต์จะนำเทคนิคของการประมวลผลทางโดเมนของเวลากับการประมวลผลทางโดเมนของความถี่ซึ่งจะได้โปรแกรมประยุกต์ 2 แบบที่ใช้เทคนิคต่างกัน

#### 6.1 ความสามารถของแบบประมวลผลทางโดเมนเวลา

แบบประมวลผลทางโดเมนเวลาจะมีความสามารถหลักอย่างทีระบุไว้ในตอนต้น และเพิ่มความสามารถบางอย่างซึ่งใช้ความได้เปรียบของการประมวลด้วยคอมพิวเตอร์ดังนี้

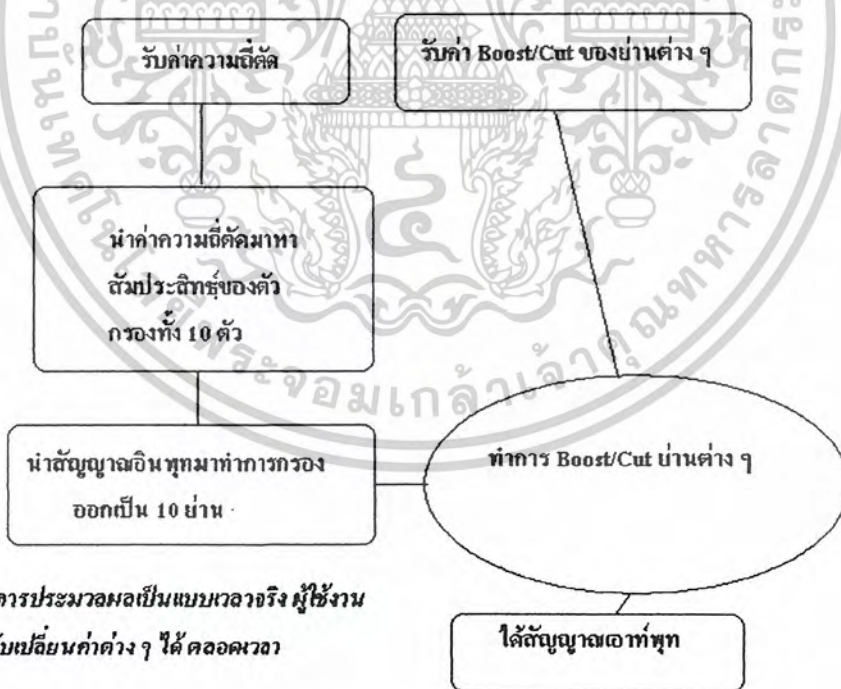
- 1.สามารถทำการเพิ่มหรือลดเกนของย่านต่าง ๆ ได้ 10 ย่าน โดยค่าเกนสามารถปรับได้ตั้งแต่ -12 dB ถึง 12 dB
- 2.สามารถเปลี่ยนจุดตัดความถี่ของย่านต่าง ๆ ได้แบบเวลาจริง
- 3.สามารถควบคุมพารามิเตอร์ต่าง ๆ ได้ด้วยตัวควบคุม MIDI
- 4.สามารถจะใช้แหล่งเสียงทั้งจากอินพุทของการ์ดเสียงและไฟล์ข้อมูลเสียง โดยไฟล์ข้อมูลเสียงจะสนับสนุนไฟล์ประเภท WAV ซึ่งเป็นรูปแบบของไฟล์เสียงมาตรฐานใช้กับระบบปฏิบัติการวินโดวส์ AFFI เหมือนกับ WAV แต่เป็นรูปแบบที่ใช้บนระบบปฏิบัติการแมคอินทอช SDII เป็นรูปแบบไฟล์ที่ใช้กับโปรแกรม Sound Designer ผลิตโดย Digidesign
- 5.สามารถบันทึกเสียงที่ปรับแต่งได้ที่ความละเอียดต่าง ๆ ตามที่การ์ดเสียงสนับสนุน
- 6.สามารถแสดงรูปคลื่นของผลตอบสนองทางอิมพัลส์ได้ทุกย่าน
- 7.สามารถเพิ่มความสามารถได้ด้วยตัวเอง รวมไปถึงการปรับแต่งให้เหมาะสมกับการใช้งานแบบต่าง ๆ ได้อิสระ

โดยมีรูปแบบ โปรแกรมแสดงดังรูปที่ 6.1



รูปที่ 6.1 โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียง โดยใช้การประมวลผลทาง โดเมนเวลา

ขั้นตอนการทำงานของ โปรแกรมที่ใช้การประมวลผลทาง โดเมนเวลาแสดงดังรูปที่ 6.2



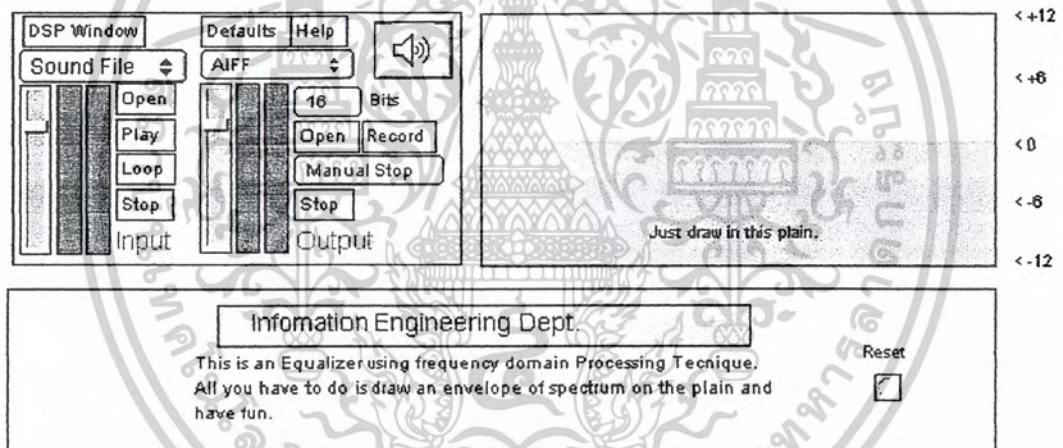
รูปที่ 6.2 แผนภาพขั้นตอนการทำงานของอิกวไลเซอร์ที่ใช้เทคนิคการประมวลในทางโดเมนเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.2 ความสามารถของแบบประมวลผลทางโดเมนความถี่

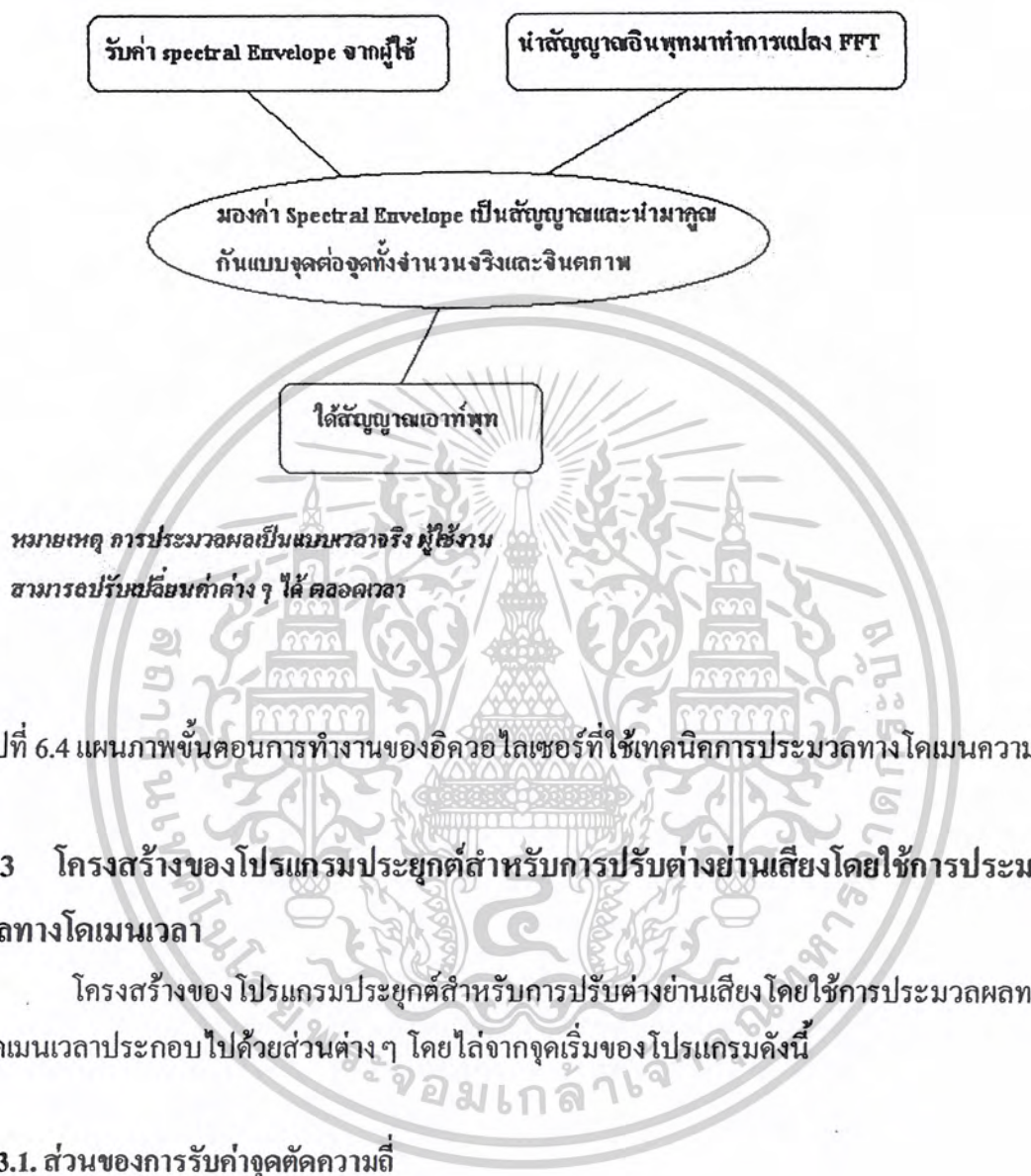
- 1.สามารถเพิ่มหรือลดความถี่ของย่านต่าง ๆ ได้ 512 ย่าน โดยใช้การควบคุมด้วยการวาดกราฟ โดยค่าเกณฑ์สามารถปรับได้ตั้งแต่  $-12$  dB ถึง  $12$  dB
- 2.สามารถที่จะใช้แหล่งเสียงทั้งจากอินพุตของการ์ดเสียงหรือไฟล์ข้อมูลเสียงหรือใช้ทั้งสองอย่างพร้อมกันเหมือนกับแบบประมวลผลทางโดเมนเวลา
- 3.สามารถบันทึกเสียงที่ปรับแต่งได้ที่ความละเอียดต่าง ๆ ตามที่การ์ดเสียงสนับสนุน
- 4.สามารถเพิ่มความสามารถได้ด้วยตัวเอง รวมไปถึงการปรับแต่งให้เหมาะสมกับการใช้งานแบบต่าง ๆ ได้อิสระ

โดยมีรูปแบบโปรแกรมแสดงดังรูปที่ 6.3



รูปที่ 6.3 โปรแกรมประยุกต์สำหรับการปรับแต่งย่านเสียงโดยใช้การประมวลผลทางโดเมนความถี่

ขั้นตอนการทำงานของโปรแกรมที่ใช้การประมวลผลทาง โดเมนความถี่แสดงดังรูปที่ 6.4

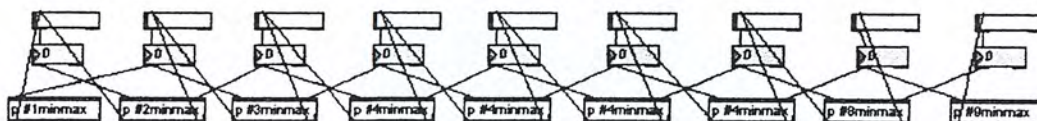


รูปที่ 6.4 แผนภาพขั้นตอนการทำงานของอิกวอลไลเซอร์ที่ใช้เทคนิคการประมวลผลทาง โดเมนความถี่

### 6.3 โครงสร้างของโปรแกรมประยุกต์สำหรับการปรับต่างย่านเสียงโดยใช้การประมวลผลทางโดเมนเวลา

โครงสร้างของโปรแกรมประยุกต์สำหรับการปรับต่างย่านเสียงโดยใช้การประมวลผลทางโดเมนเวลาประกอบไปด้วยส่วนต่างๆ โดยไล่จากจุดเริ่มของโปรแกรมหดังนี้

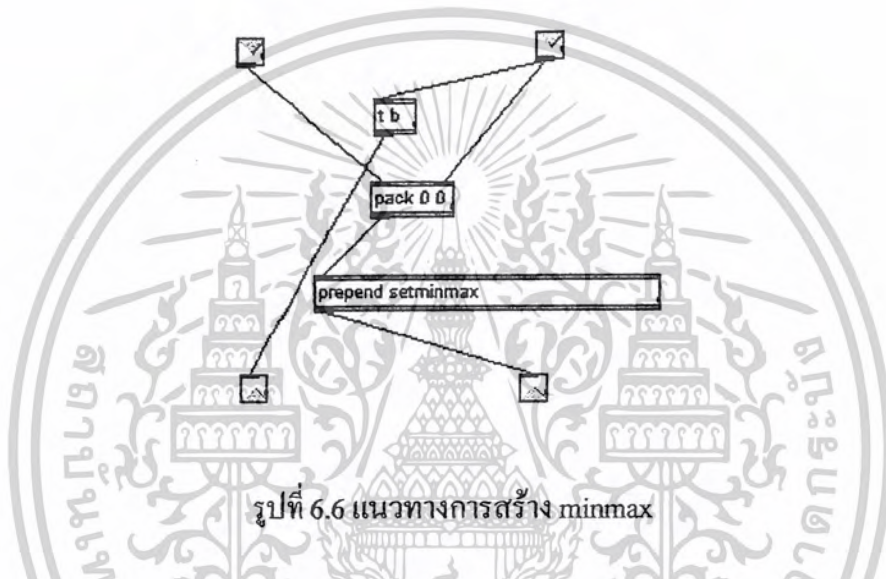
#### 6.3.1 ส่วนของการรับค่าจุดตัดความถี่



รูปที่ 6.5 ส่วนของการรับค่าจุดตัดความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำหน้าที่ในการรับความถี่ต่าง ๆ โดยผู้ใช้งานทำการเปลี่ยนย่านด้วยการเลื่อนแถบเลื่อน (มีชื่อเรียกใน Max/Msp ว่า MultiSlider แบบแนวนอน) ก่อนที่จะส่งไปยัง Number Box เพื่อแสดงผลออกมาให้ผู้ใช้งาน โดยมี Subpatch ที่ชื่อ “minmax” (ในการใช้งานปกติจะมองไม่เห็น) ไว้คอยควบคุมไม่ให้เกิดการเกินกันระหว่างตัวเลื่อน ที่อยู่สูงกว่าและต่ำกว่า โดยจะไม่ให้ค่าออกมาสูงกว่าตัวเลื่อน ที่อยู่สูงกว่าและไม่ให้ค่าออกมาต่ำกว่าตัวเลื่อน ที่อยู่ต่ำกว่า ซึ่งจะทำให้การประมวลผลผิดพลาด โดยใน minmax มีแนวทางการสร้างดังนี้



รูปที่ 6.6 แนวทางการสร้าง minmax

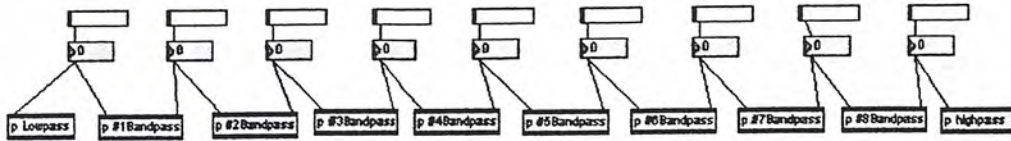
กล่องที่วางอยู่ 4มุม จะเรียก Inlet/Outlet โดยกล่องด้านบนจะเป็น Inlet กล่องด้านล่างจะเป็น Outlet ทั้ง Inlet และ Outlet จะทำหน้าที่เป็น Input/Output ที่เข้ามายัง Subpatch นั้น เพื่อนำข้อมูลเข้ามาประมวลผลใน Patch ก่อนที่จะส่งข้อมูลออกไปจาก Subpatch

Inlet ด้านซ้ายมือจะรับข้อมูลจุดตัดความถี่จากแถบที่อยู่ต่ำกว่า Inlet ด้านขวาจะรับข้อมูลจุดตัดความถี่จากแถบที่อยู่สูงกว่า ข้อมูลจาก Inlet ทั้ง 2 ตัว จะถูกส่งไปยัง pack เพื่อทำการรวมข้อมูลให้เป็นก้อนก่อนจะส่งไป prepend เพื่อจะใส่ค่านำหน้าว่า “setminmax” ยกตัวอย่างเช่นข้อมูลที่ส่งจาก Inlet ซ้ายคือ 30 ขวาคือ 50 เมื่อเข้ามาที่ pack จะเป็น 30 50 และเมื่อเข้ามาที่ prepend จะเป็น setminmax 30 50 เป็นต้น แล้วส่งไปยังตัวเลื่อน เพื่อเป็นการบังคับไม่ให้ตัวเลื่อนตัวนั้นเลื่อนค่าต่ำกว่า 30 และไม่สูงกว่า 50

ส่วนข้อมูลจาก Inlet ด้านขวาที่ไปเข้า t นั้น จะทำการส่งข้อมูลออกไป Trigger (ใน Max/Msp จะเรียก bang) เพื่อส่งไปยังตัวเลื่อน ที่อยู่ต่ำกว่าให้รับทราบการทำงานของตัวเลื่อน ตัวนั้นเพื่อแก้ปัญหาความผิดพลาดที่ไม่มีการรับทราบข้อมูลจากตัวเลื่อน ด้านขวามือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.2. ส่วนของการออกแบบตัวกรอง



รูปที่ 6.7 ส่วนของการออกแบบตัวกรอง

จากที่เห็นในรูป ส่วนของการออกแบบตัวกรองจะเป็น Subpatch ซึ่งรับค่าจุดตัดความถี่มา จากส่วนที่แล้ว โดยจะมี Subpatch 10 ตัว ซึ่งทำหน้าที่ออกแบบตัวกรองครอบคลุมย่านเสียง 20Hz ถึง 20 KHz โดยมีเป็นตัวกรองแบบผ่านต่ำ 1 ตัว แบบผ่านแถบความถี่ 8 ตัวและ ผ่านสูง 1 ตัว

โดยในการออกแบบตัวกรองเราใช้ตัวกรองแบบ FIR ในการออกแบบ โดยตัวกรองที่ใช้มี อันดับเท่ากับ 30 โดยใช้วิธีหน้าต่างสี่เหลี่ยมโดยตัวกรองตัวที่หนึ่งเป็นแบบผ่านต่ำ ตัวกรองตัวที่ สองถึงเก้าเป็นแบบผ่านแถบความถี่ ตัวกรองตัวที่สิบเป็นแบบผ่านสูง โดยมีผลตอบสนองต่ออิมพัลส์ของแต่ละตัวกรองดังนี้

$$h(n) = w(n) d(n-M) \quad \text{เมื่อ } M = (N-1)/2 = 15$$

เมื่อค่าของ  $d(n)$  จะ ได้มาจากการแปลง IDTFT กับผลตอบสนองเชิงความถี่ของตัวกรองนั้นๆ

โดยแสดงค่าของ  $d(n-M)$  มีค่าตามตารางที่ 6.1

ชนิดของตัวกรอง	$d(n-M), -15 < n-M < 15$	$d(0)$
ผ่านต่ำ(LPF)	$(\sin(\omega'_c(n-M))) / \pi(n-M)$	$\omega'_c / \pi$
ผ่านสูง(HPF)	$\delta(n-M) - (\sin(\omega'_c(n-M))) / \pi(n-M)$	$1 - (\omega'_c / \pi)$
ผ่านแถบความถี่ (BPF)	$(\sin(\omega'_b(n-M)) - \sin(\omega'_s(n-M))) / \pi(n-M)$	$(\omega'_b / \pi) - (\omega'_s / \pi)$

ตารางที่ 6.1 สมการตัวกรองแบบต่างๆ

โดยมีหลักการทำงานดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.8 หลักการทำงานของตัวกรองแบบผ่านต่ำ

สำหรับผ่านต่ำจะรับความถี่แค่เพียงความถี่เดียว เข้ามาที่ t ซึ่งแยกออกเป็นข้อมูลสองส่วน คือ Trigger (bang) และ Floating point

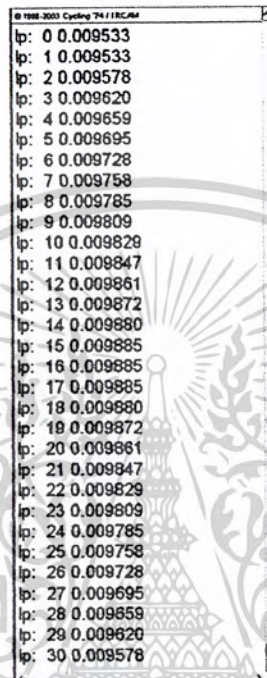
Bang จะวิ่งที่ uzi ซึ่งเป็นวัตถุที่จะยิง Bang ออกไปตามตัวเลขที่ระบุ นำไปเข้า Counter ซึ่ง จะทำการนับจำนวน Bang ตั้งแต่ 0-30 นำค่า 0-30 มาทำการลบกับ 15 ที่ expr ที่จุดนี้เราจะได้อุณหภูมิของตัวเลข จาก -15 ถึง 15 ซึ่งเป็นค่า n ที่ใช้แทนในสมการตัวกรองแบบต่าง ๆ ตามตารางที่ 6.1

Floating Point จะเข้าไปทำการคำนวณกับสมการของตัวกรองแบบผ่านต่ำ โดยใน expr ตัวแรกจะเป็นการแปลงค่าจุดตัดความถี่ให้เป็นเรเดียน จากนั้นจึงนำค่าที่ได้มาคำนวณ ซึ่งจะมี 2 กรณี ในทุก ๆ สมการคือ d(n) และ d(0) จะเห็นได้ว่ามีการใช้วัตถุแบบ expr 2 ตัวเพื่อทำการคำนวณทั้ง 2 กรณี โดยมีวัตถุ Select 0 ไว้ทำการเลือกค่า d(n) และเมื่อวัตถุ = ทำการสวิตช์ (Switch) ไปในกรณี d(0) เมื่อ n เป็น 0

ค่าที่ได้จากการคำนวณจะนำมา Pack รวมกับ Counter ซึ่งรับค่า 0-30 เป็นดัชนีในการชี้ สัมประสิทธิ์ของตัวกรอง ก่อนจะส่งไปยัง Peek~ เพื่อทำการเขียนข้อมูลลงบัฟเฟอร์ (buffer) ที่ชื่อ #llo\_buf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

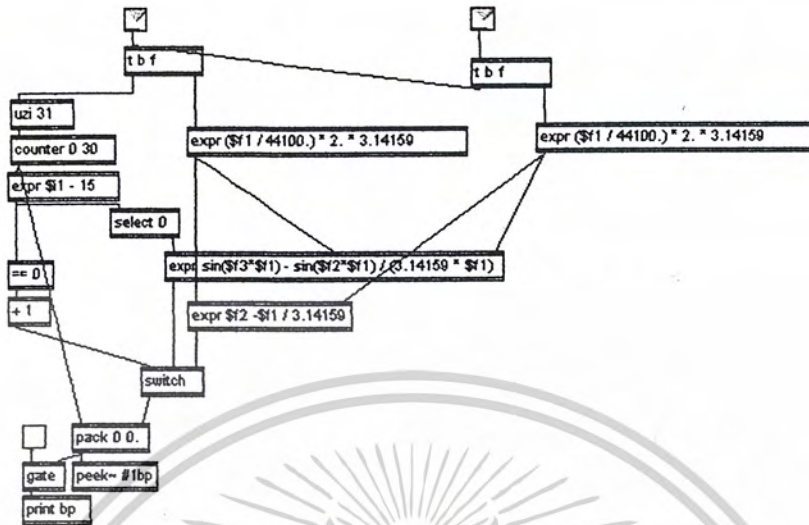
หากเราต้องการให้โปรแกรมแสดงค่าสัมประสิทธิ์ของตัวกรอง ก็ให้คลิกที่กล่องสี่เหลี่ยม และจะส่งค่า “1” ไปยัง Gate เพื่อเปิดประตูให้ค่าที่ Pack แล้ว ส่งไปยัง print เพื่อทำการแสดงสัมประสิทธิ์ของตัวกรอง ดังรูป



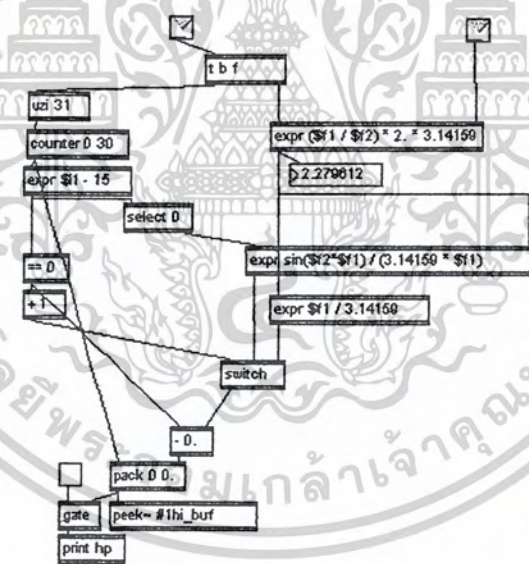
lp: 0 0.009533
lp: 1 0.009533
lp: 2 0.009578
lp: 3 0.009620
lp: 4 0.009659
lp: 5 0.009695
lp: 6 0.009728
lp: 7 0.009758
lp: 8 0.009785
lp: 9 0.009809
lp: 10 0.009829
lp: 11 0.009847
lp: 12 0.009861
lp: 13 0.009872
lp: 14 0.009880
lp: 15 0.009885
lp: 16 0.009885
lp: 17 0.009885
lp: 18 0.009880
lp: 19 0.009872
lp: 20 0.009861
lp: 21 0.009847
lp: 22 0.009829
lp: 23 0.009809
lp: 24 0.009785
lp: 25 0.009758
lp: 26 0.009728
lp: 27 0.009695
lp: 28 0.009659
lp: 29 0.009620
lp: 30 0.009578

รูปที่ 6.9 สัมประสิทธิ์ของตัวกรอง

ใน Subpatch ที่สร้างขึ้นมาเพื่อออกแบบตัวกรองแบบผ่านแถบความถี่และผ่านสูงจะใช้หลักการเหมือนกัน แต่จะต่างกันตรงสมการที่ใช้ในการคำนวณ ซึ่งมีรูปแบบตามตารางที่ 6.1 ซึ่งได้กล่าวไปแล้ว



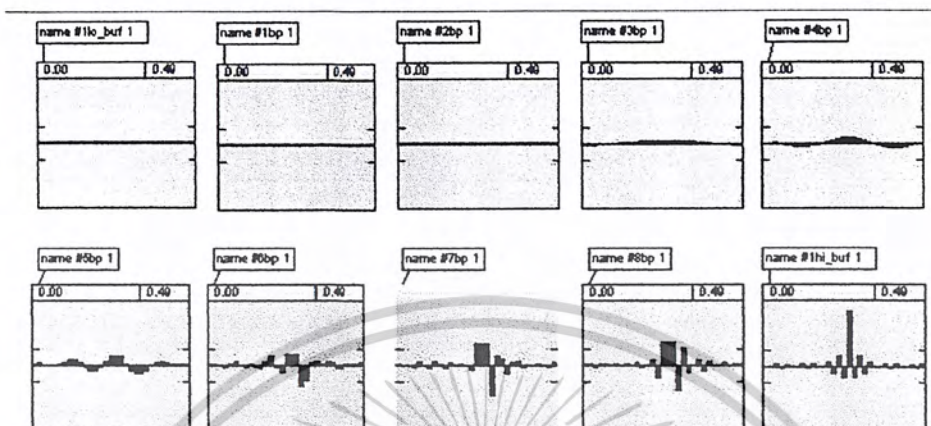
รูปที่ 6.10 หลักการทำงานของตัวกรองแบบผ่านแถบความถี่



รูปที่ 6.11 หลักการทำงานของตัวกรองแบบผ่านสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

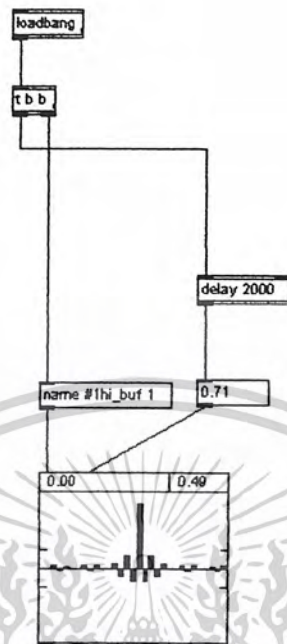
### 6.3.3. ส่วนของการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรอง



รูปที่ 6.12 ส่วนของการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรอง

ในส่วนที่แล้วได้มีการเขียนข้อมูลของผลตอบสนองทางอิมพัลส์ของทุก ๆ ตัวกรอง ลงในบัฟเฟอร์ส่วนนี้จะเป็นการนำข้อมูลนั้นมาแสดงในรูปของโดเมนเวลา

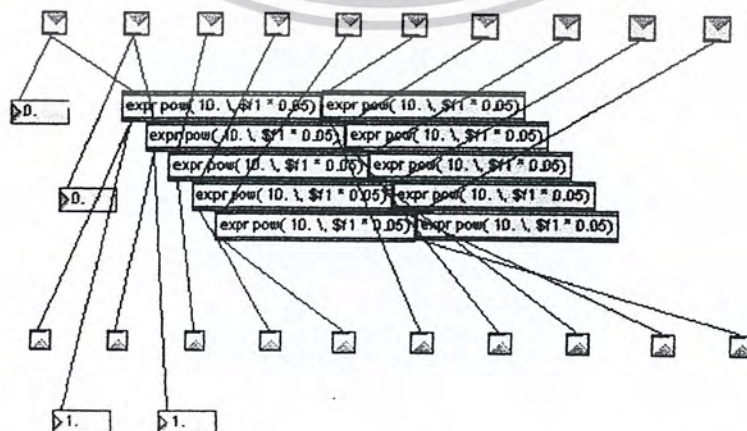
หลักการสร้าง Subpatch ถ้าหรับใช้ในการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรองแต่ละตัวนั้นจะเหมือนกัน ซึ่งจะขอยกตัวอย่างการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรองแบบผ่านสูงได้ดังนี้



รูปที่ 6.13 ตัวอย่างการแสดงผลตอบสนองทางอิมพัลส์ของตัวกรองแบบผ่านสูง

load ang ใช้ในการส่ง bang เมื่อโปรแกรมถูกโหลดขึ้นมา เมื่อส่ง bang เข้า t จะทำการส่ง bang ออกไปสองวัตถุคือ name ซึ่งเอาไว้ส่งค่าที่อยู่ในบัฟเฟอร์ที่ชื่อ #1hi\_buf ไปแสดงผลที่ waveform~ และ delay 2000 ซึ่งจะทำหน้าที่หน่วงเวลาของการส่ง bang ออกไป 2 วินาทีก่อนจะส่งไปยัง 0.71 ซึ่งจะส่งค่าไปบอกกับ waveform~ ว่าจะแสดงผลแค่ 0.71 มิลลิวินาที ซึ่งเท่ากับ 31 จุด ในกรณีที่ใช้ความถี่ของการสุ่มที่ 44100 Hz นั้นเอง

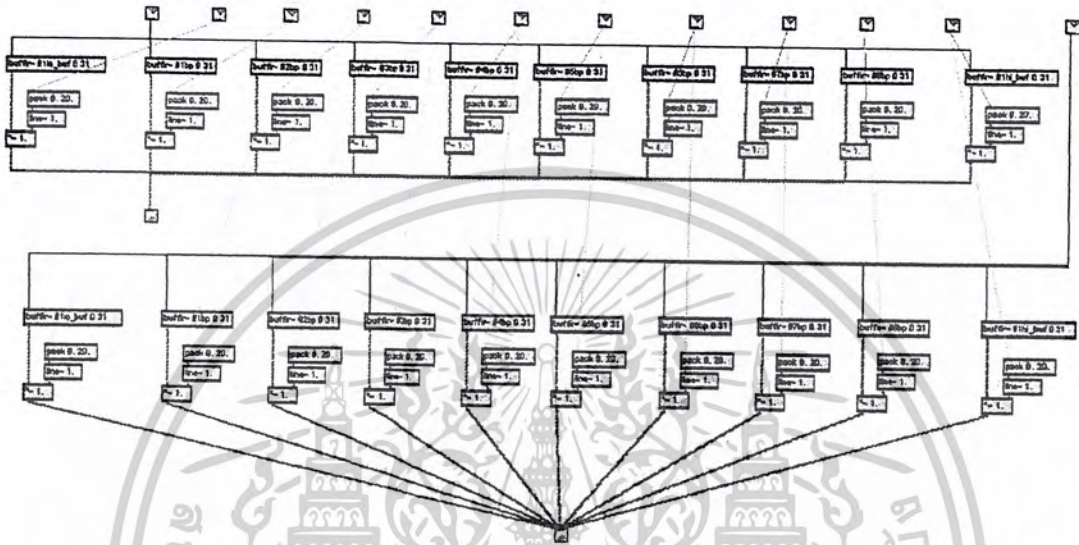
#### 6.3.4. ส่วนของการควบคุมเกณฑ์และการคูณประสาน (Convolution)



รูปที่ 6.14 ส่วนของการรับค่าการเลื่อนเกณฑ์และปรับค่าเป็น dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้เห็นาเปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของการรับค่าการเลื่อนแกนในรูปแบบที่ 6.14 จะแสดงการรับค่าการเลื่อนแกนและนำค่าที่ได้มาทำการปรับให้เป็นค่า dB และส่งไปยัง Subpatch ที่ชื่อ gcontrol ซึ่งทำในส่วนของ การควบคุมแกนและการคูณประสาน



รูปที่ 6.15 ส่วนของการควบคุมแกนและการคูณประสาน

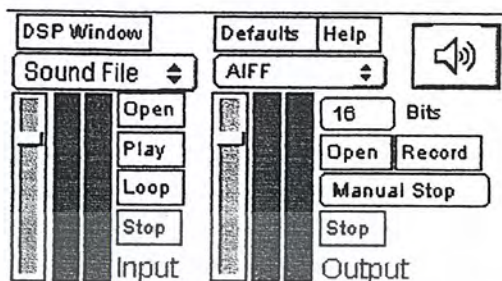
gcontrol จะมี inlet ทั้งหมด 12 ตัว เพื่อรับข้อมูลเสียงแบบสเตอริโอ (Stereo) จะอยู่ด้านซ้ายสุดและขวาสุด อีก 10 ตัวรับค่าจากตัวเลื่อนแกนเพื่อนำมาปรับแกนของย่านต่าง ๆ และมี outlet 2 ตัวสำหรับข้อมูลเสียงแบบสเตอริโอ

ข้อมูลเสียงที่เข้ามาจะนำมาทำคอนโวลูชันกับผลตอบสนองทางอิมพัลส์ของตัวกรองทุกตัว เพื่อทำการแยกเสียงออกเป็น 10 ย่านดังที่เห็นในรูป ข้อมูลสำหรับการควบคุมแกนจะนำมา pack เข้ากับ 20 (เช่น  $x \cdot 20$ ) เพื่อนำค่าที่ได้ มาใส่ให้ line ก่อนจะนำมาคูณกัน โดย line จะทำหน้าที่ให้ผลของการคูณราบรื่น ไม่มีสะดุด ตัวเลข 20 หมายถึง 20 มิลลิวินาที ในการเพิ่มหรือลดจากจุดต้น ไปยังจุดปลาย หากไม่ใส่ line ลงไปผลลัพธ์ของการคูณจะเปลี่ยนแปลงอย่างทันทีทันใด ซึ่งอาจทำให้เกิดการคลิกของเสียงได้

สุดท้าย สัญญาณเสียงจะมารวมกันที่ outlet ก่อนจะส่ง ไปยังส่วนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6.3.5. ส่วนสัญญาณขาเข้าและสัญญาณขาออกของระบบ



รูปที่ 6.16 ส่วนสัญญาณขาเข้าและสัญญาณขาออกของระบบ

ส่วนนี้มีแถบมาให้ใช้งาน โดย Max/Msp ไม่จำเป็นต้องสร้างเอง มีไว้สำหรับสัญญาณขาเข้า และสัญญาณขาออกของระบบที่เราสร้าง

DSP Window จะให้เราทำการปรับแต่งระบบของการ์ดเสียง (Soundcard)

Defaults คือการใช้ค่าเริ่มต้นที่กำหนดมาโดยโปรแกรม

Help มีไว้อธิบายการใช้งาน Patch นี้ เป็นภาษาอังกฤษ

Speaker มีไว้เปิดระบบเสียงให้ทำงาน

Sound File เป็นการเลือกที่จะใช้งานไฟล์เสียงหรือสัญญาณขาเข้าที่รับมาจากการ์ดเสียง

AIFF คือประเภทของไฟล์ที่ใช้ในการบันทึก

Open จะใช้เปิดไฟล์เสียง

Play ใช้เล่นไฟล์เสียง

Loop ใช้เล่นไฟล์เสียงวนลูป

Stop หยุดเล่นไฟล์เสียง

16 คือเลือกความลึกของบิต (Bit Depth) ของเสียงที่เราจะใช้ทำการบันทึก ทั้งนี้ก็ขึ้นอยู่กับ การรองรับของการ์ดเสียงด้วย

Open (ด้านสัญญาณขาออก) คือการเก็บข้อมูลเสียง

Record คือเริ่มการบันทึก

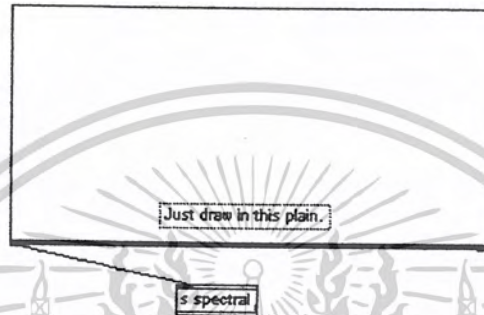
Manual Stop คือการหยุดโดยผู้ใช้อย่าง

IO จะทำการรับส่งค่าจากส่วนที่แล้ว โดยส่งสัญญาณเสียงเข้า gcontrol และสัญญาณขาออกที่ออกมาจาก gcontrol ก็จะกลับเข้าไปเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.4 โครงสร้างของโปรแกรมประยุกต์สำหรับการปรับต่างย่านเสียงโดยใช้การประมวลผลทางโดเมนความถี่

โครงสร้างของโปรแกรมประยุกต์สำหรับการปรับต่างย่านเสียงโดยใช้การประมวลผลทางโดเมนความถี่ มีความเรียบง่ายไม่ซับซ้อน โดยเริ่มจาก



รูปที่ 6.17 เริ่มต้นการใช้โครงสร้างโปรแกรมประยุกต์

จะทำการรับค่าจากผู้ใช้งานซึ่งจะใช้ MultiSlider 512 ตัวเพื่อการนี้ เมื่อรับค่าต่าง ๆ จากผู้ใช้แล้ว ส่งออกไปโดยใช้ S เป็นวัตถุที่ใช้ช่วยส่งค่า โดยมีตัวแปรที่ชื่อ Spectral

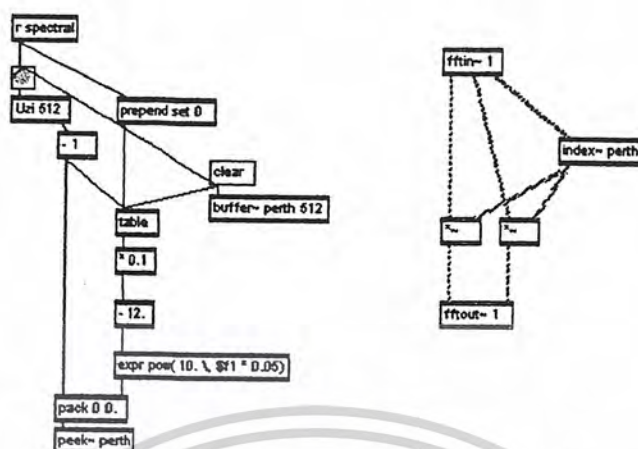
ค่านี้จะส่งไปยัง pfft~ ซึ่งเป็นวัตถุที่ใช้ประมวลผลทางโดเมนความถี่ โดยเราต้องเขียน Patch ขึ้นมาก่อน แล้วใช้ pfft~ เรียก patch นั้นขึ้นมาใช้งาน

pfft~ iteproject 1024

รูปที่ 6.18 ส่วนของการเรียก Patch ที่ชื่อ iteproject 1024 ขึ้นมาใช้งาน

จากในรูปจะเป็นการเรียก Patch ที่ชื่อ iteproject ตัวเลข 1024 หมายถึงขนาดของ FFT ซึ่งภายใน iteproject จะมีโครงสร้างดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.19 โครงสร้างภายใน iteproject

ข้อมูลที่ส่งจาก s จะรับด้วย r เข้ามา โดย list ของข้อมูลจะส่งไปยังตารางโดยผ่าน prepend set 0 ก่อน เพื่อเป็นการใส่เลขลงใน Address ศูนย์ ซึ่งเป็น Address แรก และจะทำการ Address ค้าง แต่ 0-511 โดย uzi ที่ต้องทำการ Address ด้วย uzi ก็เพราะต้องการให้ผู้ใช้งาน Envelope ของสเปกตรัมได้โดยตลอด ไม่ต้องทำการเปิดโปรแกรมใหม่ และจากตารางจะส่งไป pack โดยก่อน pack จะทำการนำมาคูณกับ 0.01 เพราะค่าเริ่มต้นเป็น 100 ต้องนำมาทำการนอร์มัลไลซ์ (Normalize) ให้เป็น 1 จาก pack จะส่งไป peek เพื่อทำการเกิดค่าจากตารางลงบัฟเฟอร์นั่นเอง

fftin~ เป็นวัตถุพิเศษสำหรับใช้เป็นสัญญาณขาเข้าใน pfft เท่านั้นซึ่งจะทำการแปลง FFT สัญญาณขาเข้าที่เข้ามายัง pfft ให้อยู่ในโดเมนของความถี่ เรานำค่านั้นมาคูณกับ index~ ซึ่งรับค่าดัชนีจาก fftin แล้วนำมาคูณกับค่าที่อยู่ในบัฟเฟอร์ที่ชื่อ perth แบบจุดต่อจุด ที่เห็นการคูณสองที่นั่นเป็นการคูณของส่วนจริงและส่วนจินตภาพเรียงจากซ้ายไปขวา และนำผลที่ได้จากการคูณนี้ไปออกที่ fftout~ ซึ่งเป็นวัตถุพิเศษสำหรับ pfft เหมือนกับ fftin แต่ใช้เป็นสัญญาณขาออกของ pfft

pfft จะรับข้อมูลที่ออกและเข้าจาก IO เหมือนกับแบบที่แล้ว ซึ่งเราจะใช้ความสามารถของ IO ในการบันทึกและเล่นไฟล์เสียงเหมือนเดิมทุกอย่าง แต่เนื่องจาก pfft iteproject มีสัญญาณขาเข้าและขาออกเป็นแบบทางเดียว (mono) จึงต้องใช้ 2 ตัวในการประมวลผลเป็นแบบสเตอริโอ

### ข้อดีของดิจิทัลออลิควอลไลเซอร์

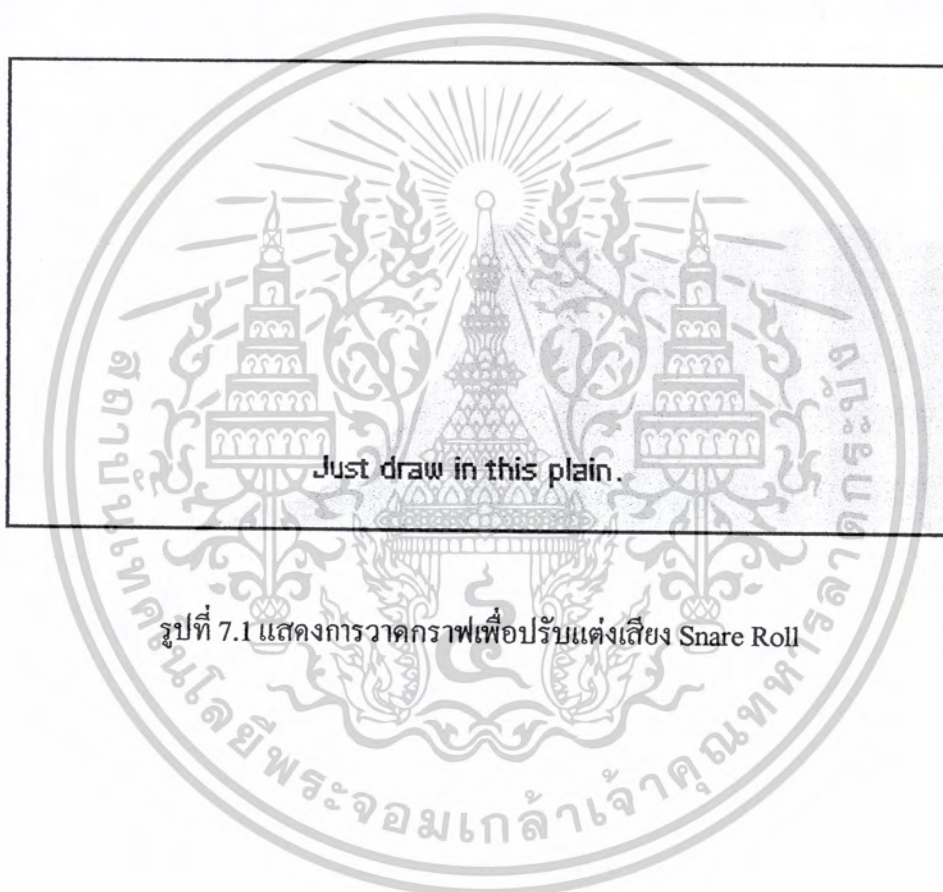
จะมีทั้งข้อดีที่มาจากไฟล์เสียง ซึ่งแม้ว่าจะทำการแปลงจากอนาลอกมาดีเพียงใด ก็ไม่สามารถสร้างคืนมาได้เหมือนเดิม 100% และข้อดีที่มาจากความหน่วงของอินเทอร์เฟซสัญญาณเสียง (Audio Interface) ซึ่งการประมวลผลในระบบดิจิทัลจำเป็นต้องมีความหน่วงเกิดขึ้น ขึ้นอยู่กับการออกแบบวงจรและโปรแกรมไครเวอร์ของอินเทอร์เฟซสัญญาณเสียงนั้นว่าจะให้ความหน่วงออกมาอย่างน้อยแค่ไหน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 7

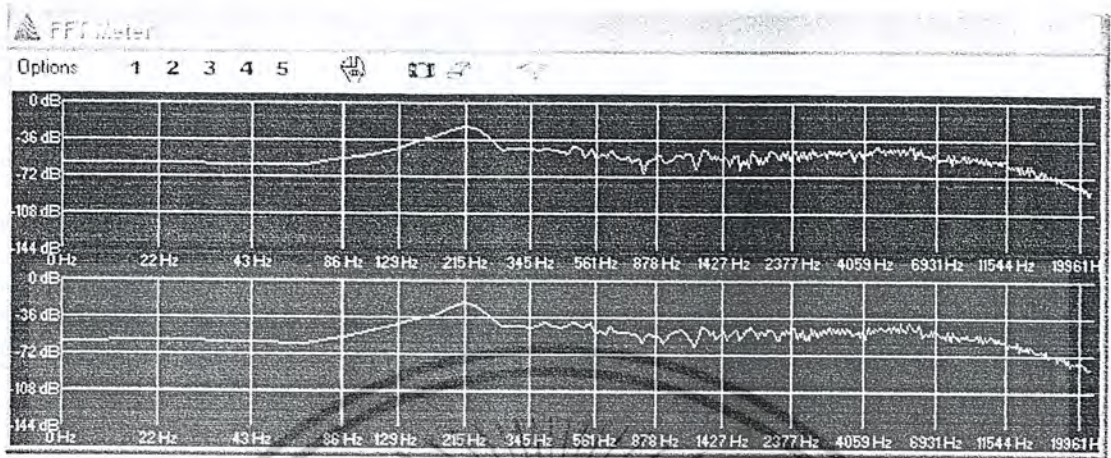
### ผลการทดลอง

ผลการทดลองการทำงานของอิควอไลเซอร์แบบ FFT ด้วยโปรแกรม MATLAB  
โปรแกรม MATLAB แสดงในภาคผนวก  
ทดลองปรับแต่งเสียงของเสียง Snare Roll (ผลการทดลองอย่างละเอียด แสดงอยู่ใน CD-ROM)

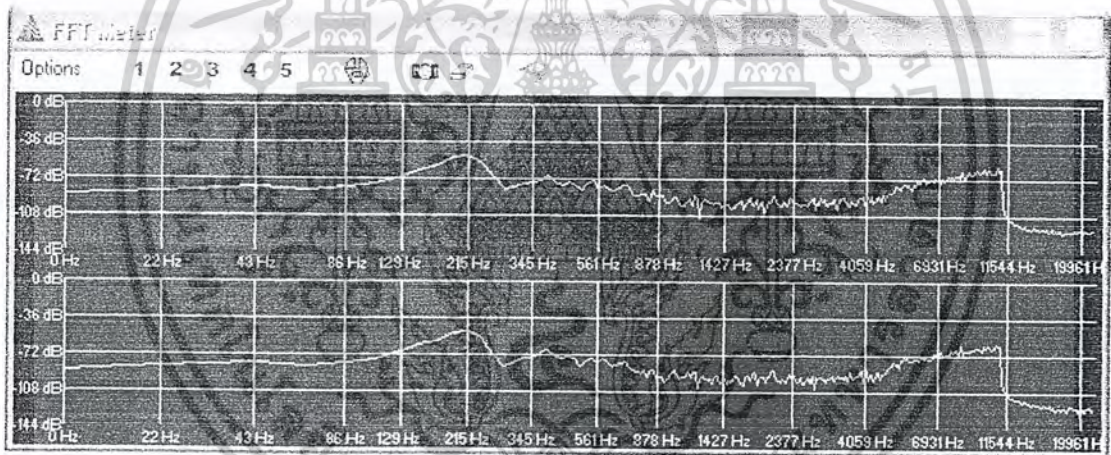


รูปที่ 7.1 แสดงการวาดกราฟเพื่อปรับแต่งเสียง Snare Roll

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

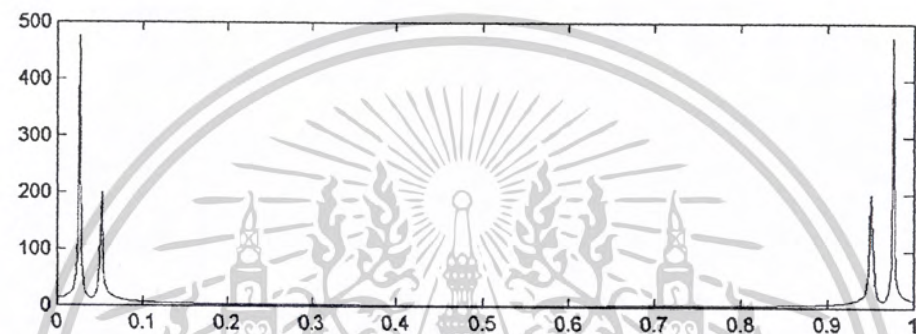
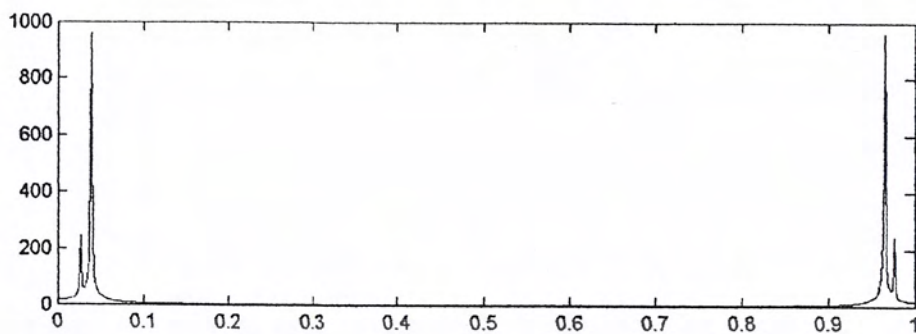


รูปที่ 7.2 แสดงสเปกตรัมของเสียง Snare roll

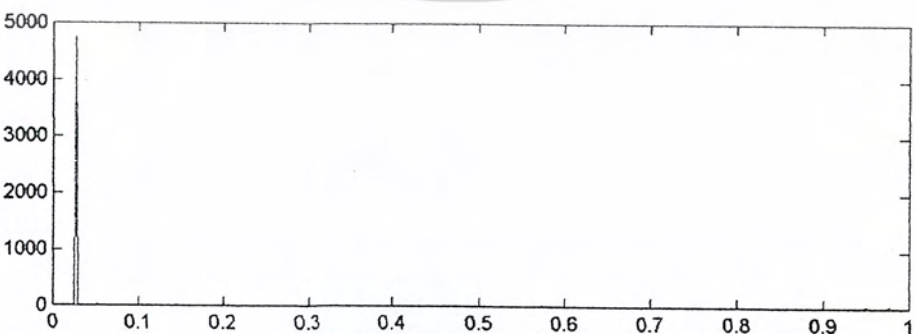
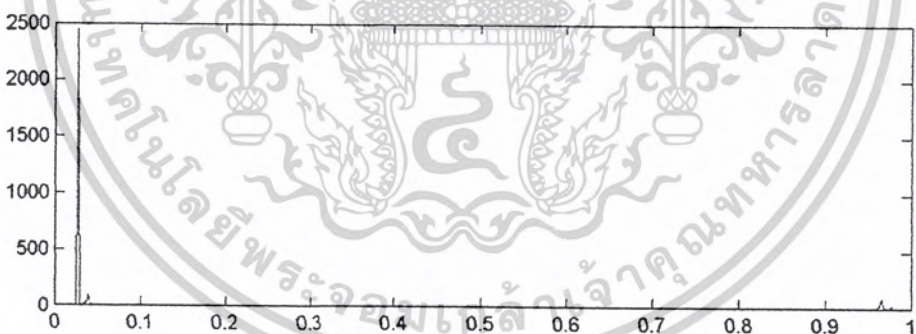


รูปที่ 7.3 แสดงสเปกตรัมของเสียง Snare Roll เมื่อผ่านการปรับแต่งแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

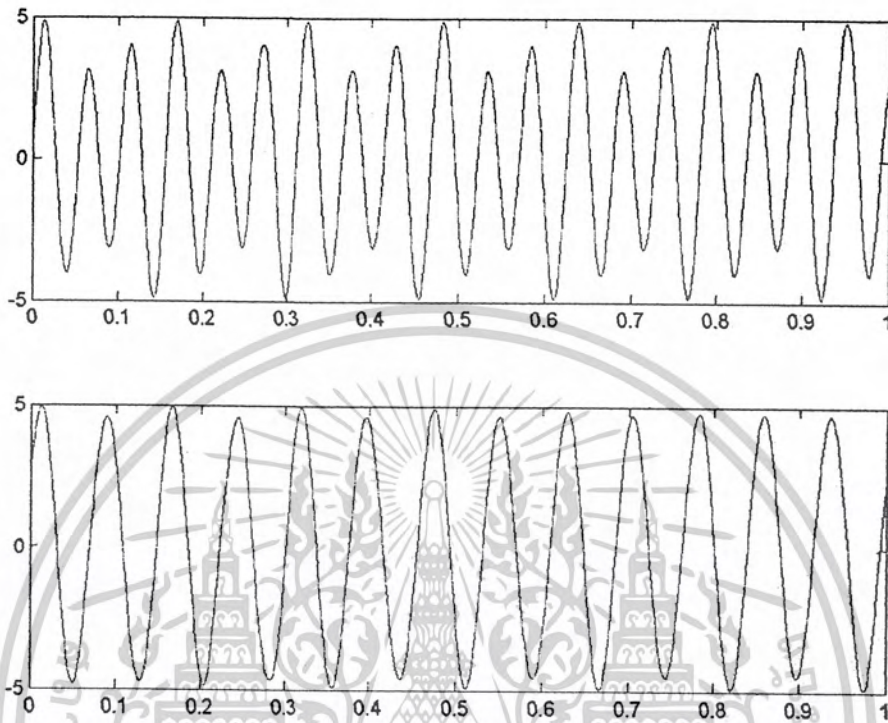


รูปที่ 7.4 แสดงสเปกตรัมของสัญญาณขาเข้า



รูปที่ 7.5 แสดงสเปกตรัมของสัญญาณขาออกเมื่อถูกปรับแต่งแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.6 แสดงสัญญาณขาเข้า เมื่อเทียบกับสัญญาณขาออก ในโดเมนเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 8

### บทสรุป

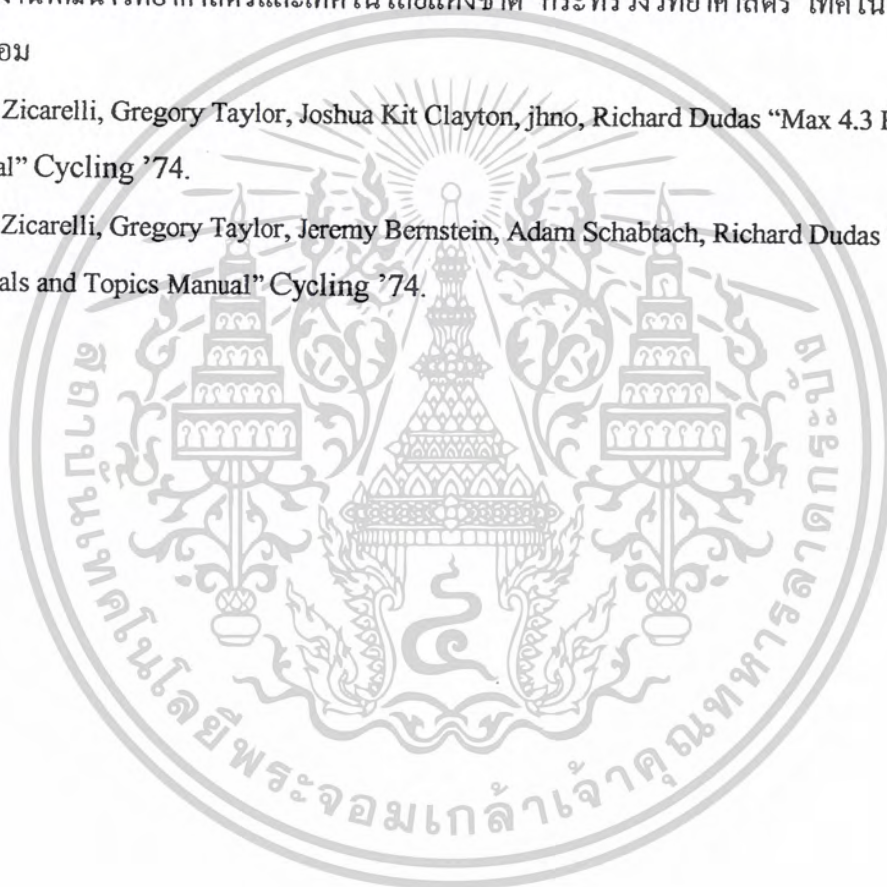
โปรแกรมประยุกต์อ็ควอไลเซอร์สัญญาณเสียงดิจิทัลแบบกราฟฟิก ที่ได้ออกแบบโดยใช้วิธีการประมวลผลทางโดเมนของเวลานี้ เป็นกราฟฟิกอ็ควอไลเซอร์ที่สามารถบูสท์และคัทความถี่ในแต่ละย่านได้อย่างอิสระต่อกัน โดยแบ่งย่านความถี่ออกเป็น 10 ย่าน และสามารถปรับค่าความถี่คัทในแต่ละย่านได้ และ โปรแกรมประยุกต์ที่ออกแบบโดยใช้การประมวลผลทางความถี่สามารถแบ่งความถี่ได้ถึง 512 ย่าน โดยรับค่าจากผู้ใช้ผ่านการวาดกราฟ

จากการทดลองพบว่าสามารถบูสท์และคัทสัญญาณเสียงดิจิทัลได้ตามต้องการ สามารถนำไปใช้ในการปรับแต่งเสียงได้อย่างมีประสิทธิภาพ

โดยมีข้อดีของดิจิทัลอ็ควอไลเซอร์ในการสร้างเสียงขึ้นมาซึ่งมีผลมาจากไฟล์เสียง ถึงแม้ว่าจะแปลงสัญญาณจากอนาลอกมาดีเพียงใด ก็ไม่สามารถสร้างขึ้นมาได้ 100% และมีการหน่วงของเวลา ซึ่งในการประมวลผลสัญญาณดิจิทัลจำเป็นต้องมีการหน่วงเกิดขึ้น

## บรรณานุกรม

1. Rulph Chassaing "DSP Application Using C and the TMS320C6x DSK" John Wiley & Sons
2. อรรถสิทธิ์ หล้าสกุล "Digital Signal Processing" แผนกตำราคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
3. ไพรัช รัชชพงษ์ "การประมวลสัญญาณดิจิทัล: ตอนการออกแบบวงจรกรองดิจิทัล = Digital signal processing : digital filter design" ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ กระทรวงวิทยาศาสตร์ เทคโนโลยีและสิ่งแวดล้อม
4. David Zicarelli, Gregory Taylor, Joshua Kit Clayton, John, Richard Dudas "Max 4.3 Reference Manual" Cycling '74.
5. David Zicarelli, Gregory Taylor, Jeremy Bernstein, Adam Schabtach, Richard Dudas "Max 4.3 Tutorials and Topics Manual" Cycling '74.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก

% โปรแกรมการจำลองการทำงานของคิซิจิตอลกราฟฟิคอควอไลเซอร์แบบ 10 แบนด์

[x, fs] = wavread('J:\Loop\sonicfoundry cinematic vol2\Individual

Instruments and Grooves\Drums and Percussion\NYPD Drum 01.WAV');

x1=x(:,1); %แยกเสียงออกมา 1 ch

n=50; % อันดับเท่ากับ 50

wn0=[0.1]; % เป็นการเลือกจุดตัดความถี่ โดย  $W_n = 1.0$

wn1 = [0.1 0.2]; % จะมีความถี่เป็น 1/2 ของอัตราการสุ่ม

wn2 = [0.2 0.3]; % ในกรณีนี้จะเลือกมา 10 ย่านเพื่อกรองเอาความถี่

wn3 = [0.3 0.4]; % 10 ย่านที่ต้องการ

wn4 = [0.4 0.5];

wn5 = [0.5 0.6];

wn6 = [0.6 0.7];

wn7 = [0.7 0.8];

wn8 = [0.8 0.9];

wn9 = [0.9];

h0 = fir1(n,wn0); % นำค่าอันดับและจุดตัดความถี่มาผ่านตัวกรองแบบ fir

h1 = fir1(n,wn1); % โดยใช้หน้าต่างแบบแฮมมิง

h2 = fir1(n,wn2);

h3 = fir1(n,wn3);

h4 = fir1(n,wn4);

h5 = fir1(n,wn5);

h6 = fir1(n,wn6);

h7 = fir1(n,wn7);

h8 = fir1(n,wn8);

h9 = fir1(n,wn9,'high');

y0 = conv (x1,h0); % นำผลตอบสนองที่ได้มาคอนโวลูชันกับ

y1 = conv (x1,h1); % สัญญาณเสียงต้นแบบเพื่อต้องการสัญญาณปลายทาง

y2 = conv (x1,h2); % ที่ถูกกรองแล้ว 10 สัญญาณที่กรองมาจากตัวกรองต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

y3 = conv (x1,h3);           % 10 ตัว
y4 = conv (x1,h4);
y5 = conv (x1,h5);
y6 = conv (x1,h6);
y7 = conv (x1,h7);
y8 = conv (x1,h8);
y9 = conv (x1,h9);
sd1 = uicontrol(gcf,...
    'style','slider',...
    'position',[100 300 20 120],...
    'min',0,'max',2,'value',0.1);
    % เป็นการควบคุมพารามิเตอร์ของตัวเลื่อน
    % 'callback','get(sd1,"value")
sd2 = uicontrol(gcf,...
    'style','slider',...
    'position',[150 300 20 120],...
    'min',0,'max',2,'value',0.1);
sd3 = uicontrol(gcf,...
    'style','slider',...
    'position',[200 300 20 120],...
    'min',0,'max',2,'value',1);
sd4 = uicontrol(gcf,...
    'style','slider',...
    'position',[250 300 20 120],...
    'min',0,'max',2,'value',1);
sd5 = uicontrol(gcf,...
    'style','slider',...
    'position',[300 300 20 120],...
    'min',0,'max',2,'value',1);
sd6 = uicontrol(gcf,...
    'style','slider',...

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'position',[350 300 20 120],...
'min',0,'max',2,'value',1.5);
sd7 = uicontrol(gcf,...
'style','slider',...
'position',[400 300 20 120],...
'min',0,'max',2,'value',1);
sd8 = uicontrol(gcf,...
'style','slider',...
'position',[450 300 20 120],...
'min',0,'max',2,'value',1.5);
sd9 = uicontrol(gcf,...
'style','slider',...
'position',[500 300 20 120],...
'min',0,'max',2,'value',1);
sd10 = uicontrol(gcf,...
'style','slider',...
'position',[550 300 20 120],...
'min',0,'max',2,'value',1);

output1=get(sd1,'value')*y0; %เป็นการนำผลที่ได้จากการกรองมาคูณกับแกน
output2=get(sd2,'value')*y1; %ที่ได้จากตัวเลื่อน เพื่อใช้ในการบวสท์/ลัทย่านต่างๆ
output3=get(sd3,'value')*y2;
output4=get(sd4,'value')*y3;
output5=get(sd5,'value')*y4;
output6=get(sd6,'value')*y5;
output7=get(sd7,'value')*y6;
output8=get(sd8,'value')*y7;
output9=get(sd9,'value')*y8;
output10=get(sd10,'value')*y9;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

output = output1 + output2 + output3 + ...           % นำสัญญาณที่ได้จากการ
+output4 + output5 + output6 + ...                 % บุสต์/คัท มารวมกันเพื่อเป็น
+output7 + output8 + output9 + output10;           % สัญญาณใหม่

```

```

bt1 = uicontrol(gcf,...
    'style','pushbutton',...
    'string','Play',...
    'callback','sound(output,fs)');

```

```

sound(output,fs);
freqz(output);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

%โปรแกรม MATLAB แสดงผลการทดลองของอีควอไลเซอร์แบบ FFT

clear all

fs = 40960;

f = 1024;

for n=1:512

$x(n) = \sin(2\pi n f / fs) + 4 \sin(2\pi 1.5 n f / fs);$

$x2(n) = 2 \sin(2\pi n f / fs) + \sin(4\pi n f / fs);$

end

%sound(x,fs);

X=fft(x,512);

X2=fft(x2,512);

t=1/512:1/512:1;

figure(1);

subplot(2,1,1)

plot(t,abs(X))

subplot(2,1,2)

plot(t,abs(X2))

for n=1 :512

$G(n)=0.1 \cos(45) + 0.1 i \sin(45);$

end

$G(14)=10 \cos(45) + 10 i \sin(45);$

figure(2)

$XX2=X2.*G;$

$XX=X.*G;$

subplot(2,1,1)

plot(t,abs(XX));

subplot(2,1,2)

plot(t,abs(XX2));

$x3=ifft(XX,512);$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
x4=ifft(XX2,512);
```

```
figure(3)
```

```
subplot(2,1,1)
```

```
plot(t,x);
```

```
subplot(2,1,2)
```

```
plot(t,x3);
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้