

พารามетริกอีควอลไลเซอร์แบบดิจิทัลโดยใช้ปัญญาประดิษฐ์ช่วยในการปรับจูนและ
การสร้างกราฟฟิคอีควอลไลเซอร์แบบดิจิทัลบนอุปกรณ์ FPGA

Digital Parametric Equalizer with AI-assisted Tuning and FPGA
Implementation of Digital Graphic Equalizer



ปริญญานิพนธ์เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญาานิพนธ์

หัวข้อปริญญาานิพนธ์ พารามตริกอีควอลไลเซอร์แบบดิจิตอลโดยใช้ปัญญาประดิษฐ์ช่วยในการปรับจูน และการสร้างกราฟฟิคอีควอลไลเซอร์แบบดิจิตอลบนอุปกรณ์ FPGA

Thesis Title Digital Parametric Equalizer with AI-assisted Tuning and FPGA Implementation of Digital Graphic Equalizer

นักศึกษา นายภคพล ชิตสกุล, นายอิทธิพล นวอิทธิพร, นายภูวิศ ผลดี

รหัสประจำตัว 5810933, 58011453, 58010995

ปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมดนตรีและสื่อประสม

อาจารย์ที่ปรึกษาปริญญาานิพนธ์ ผศ.ดร. ศรวัฒน์ ชิวปรีชา

คณะกรรมการสอบปริญญาานิพนธ์	ลายมือชื่อ
ผศ.ดร. ศรวัฒน์ ชิวปรีชา	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	พารามetriกรีกอไลเซอร์แบบดิจิทัลโดยใช้ปัญญาประดิษฐ์ช่วยในการปรับจูนและการสร้างกราฟฟิกรีกอไลเซอร์แบบดิจิทัลบนอุปกรณ์ FPGA
นักศึกษา	นายภคพล ชิตสกุล, นายอิทธิพล นวอิทธิพร, นายภูวิช ผลดี
รหัสประจำตัว	58010933, 58011453, 58010995
ปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมดนตรีและสื่อประสม
พ.ศ.	2562
อาจารย์ที่ปรึกษาปริญญานิพนธ์	ผศ.ดร. ศรวัดน์ ชิวปรีชา

บทคัดย่อ

โครงการนี้เป็นการประยุกต์ใช้ตรรกศาสตร์คลุมเครือ (Fuzzy logic) ในการปรับค่า Parameters ของระบบประมวลผลสัญญาณสำหรับ Digital Parametric Equalizer การปรับค่า Parameters ต่างๆ จะอ้างอิงจากคุณลักษณะทางความถี่ของเสียงเครื่องดนตรีชนิดต่างๆ ผนวกกับความรูสึกและประสบการณ์ของผู้ใช้งานหรือผู้เชี่ยวชาญ ซึ่งไม่มีรูปแบบที่เฉพาะเจาะจงขึ้นกับประสบการณ์และความรูสึกของผู้ใช้งานหรือผู้เชี่ยวชาญที่เกี่ยวข้อง ทำให้สามารถนำทฤษฎีตรรกศาสตร์คลุมเครือมาประยุกต์ใช้งานร่วมด้วยได้ ทั้งนี้เพื่อเป็นส่วนช่วยในการปรับค่า Parameters สำหรับผู้ที่เริ่มต้นศึกษาการใช้งานการปรับแต่ง Digital Parametric Equalizer นอกจากนี้ในโครงการนี้ยังได้ทำการออกแบบและสร้าง Digital Graphic Equalizer ให้สามารถทำงานแบบ Real-time เพื่อเป็นการศึกษาการออกแบบและการนำวงจร Digital Filter ไปสร้างและใช้งานในรูปแบบที่เป็นฮาร์ดแวร์ได้ ซึ่งผลลัพธ์ของการดำเนินงานนี้ ในส่วนของ Digital Parametric Equalizer จะทำการสร้างเป็นซอฟต์แวร์ที่ทำงานบน MATLAB และในส่วนของ Digital Graphic Equalizer จะทำการสร้างเป็นฮาร์ดแวร์ที่ทำงานบนอุปกรณ์ FPGA (Field Programmable Gate Array) รวมทั้งผลการทดลองทำงานของระบบ Digital Equalizer ทั้ง 2 รูปแบบที่นำเสนอยังสามารถสาคิตและแสดงให้เห็นผลการดำเนินงาน

Thesis Digital Parametric Equalizer with AI-assisted Tuning and FPGA Implementation of Digital Graphic Equalizer

Student Pakapol Chitsakul, Itthiporn Navaitthiporn, Poowit Pholdee

Student ID. 558010933, 58011453, 58010995

Degree Bachelor of Engineering

Program Music Engineering and Multimedia

Year 2019

Thesis Advisor Asst. Prof. Dr. Sorawat Chivapheecha

ABSTRACT

This project presents an application of fuzzy logic to adjust parameters of signal processing system for digital parametric equalizer. Parameters adjustment are based on frequency characteristics of each musical instruments including with feeling and experience of users or experts, which are not specified but depend on experience and feeling of related users or experts. By this reason, the theory of fuzzy logic can be applied to inference the system that can assist in parameters tuning or adjust for the beginners of digital parametric equalizer users. Moreover, in this project also proposes a design and implementation digital graphic equalizer that can be used in real-time application. The design and application of digital filter, especially in hardware implementation will be studied in this work. Consequently, the software implementation of digital parametric equalizer will be implemented using MATLAB and the hardware implementation of digital graphic equalizer will be implemented on FPGA (Field Programmable Gate Array). Experimental results of the application of both proposed digital equalizers are demonstrated and show the results of their processing tasks.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาปริญญาตรี สำเร็จลุล่วงไปได้ด้วยความกรุณาจาก ผศ.ดร. ศรวัฒน์ ชิวปรีชา ซึ่งเป็นอาจารย์ที่ปรึกษาปริญญาานิพนธ์ที่ได้คำปรึกษา เสนอแนวคิด และให้คำแนะนำ พร้อมทั้งให้กำลังใจทำให้ปริญญาานิพนธ์ครั้งนี้สำเร็จลุล่วงด้วยดี

ขอขอบพระคุณคณาจารย์ และบุคลากรทางการศึกษา สาขาวิศวกรรมดนตรี และสื่อประสม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบังทุกท่านที่คอยให้การสนับสนุนในการทำปริญญาานิพนธ์ในครั้งนี้พร้อมทั้งให้คำปรึกษาในการแก้ไขปรับปรุงให้การทำงานในครั้งนี้มีประสิทธิภาพสูงสุด

และสุดท้ายนี้ขอขอบคุณเพื่อนๆ สาขาวิศวกรรมดนตรีและสื่อประสม ที่คอยให้กำลังใจและช่วยเสนอความคิดเห็นทำให้งานในครั้งนี้ลุล่วง

คณะผู้จัดทำ



สารบัญ

หน้า

บทคัดย่อ.....	i
ABSTRACT.....	ii
กิตติกรรมประกาศ.....	iii
สารบัญ.....	iv
สารบัญรูป.....	vi
สารบัญตาราง.....	x
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์.....	1
1.4 ขอบเขตของการศึกษา.....	2
1.5 ขั้นตอนการศึกษา.....	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....	3
2.1 Digital signal processing (DSP).....	3
2.2 Digital Filter.....	3
2.3 Parametric Equalizer.....	4
2.3.1 Center Frequency.....	5
2.3.2 Quality Factor.....	5
2.3.3 Gain.....	5
2.4 Fuzzy Logic.....	6
2.4.1 Fuzzy set.....	7
2.4.2 Linguistic Variable และ Fuzzy Rules.....	10
2.4.3 Mamdani-Style Inference.....	12
2.5 Graphic Equalizer.....	16
2.6 อุปกรณ์ NI myRIO.....	17
2.6.1 อุปกรณ์ FPGA (Field - Programmable Gate Array).....	18
2.6.2 Laboratory Virtual Instrument Engineer Workbench (LabVIEW).....	18

สารบัญ(ต่อ)

	หน้า
บทที่ 3 วิธีการดำเนินงาน.....	22
3.1 Digital Parametric Equalizer	22
3.1.1 การรับอินพุตจากผู้ใช้งาน.....	23
3.1.2 Pre-processing	25
3.1.3 Fuzzy logic AI.....	28
3.1.4 Peaking Filter	35
3.1.5 การออกแบบหน้าต่าง User Interface.....	36
3.2 Digital Graphic Equalizer.....	38
3.2.1 การออกแบบ.....	38
3.2.2 การเขียนโปรแกรมบน Labview เพื่อควบคุม NI myRIO.....	39
3.2.3 การทดสอบการทำงาน.....	42
บทที่ 4 ผลการออกแบบและทดสอบ.....	44
4.1 Digital Parametric Equalizer	44
4.1.1 ส่วนการรับอินพุต.....	44
4.1.2 ส่วนการทำ Pre-processing	45
4.1.3 Fuzzy logic	50
4.1.4 ผลลัพธ์การทดลอง	52
4.2 Digital Graphic Equalizer.....	61
4.2.1 การแสดงผลลัพธ์ Magnitude response ผ่านหน้าจอ User Interface	61
4.2.2 การแสดงผลลัพธ์ Magnitude response จากอุปกรณ์ Dynamic Signal Analyze.....	63
4.2.3 ค่า spectrum ก่อนและหลัง ทำการ EQ ในแต่ละการทดลอง.....	66
บทที่ 5 สรุปผลและข้อเสนอแนะ.....	71
5.1 สรุปผลการทดลอง	71
5.2 ข้อเสนอแนะ	71
บรรณานุกรม.....	73

สารบัญรูป

หน้า

รูปที่ 2.1 รูปตัวอย่าง plug-in Parametric Equalizer.....	5
รูปที่ 2.2 รูปแสดง Bandwidth ที่เป็นผลลัพธ์ของการปรับ Quality Factor.....	5
รูปที่ 2.3 ขอบเขต Degree of Membership ของ (ก) Boolean Logic และ (ข) Fuzzy Logic.....	7
รูปที่ 2.4 กราฟแสดงความสัมพันธ์ระหว่างข้อมูลความสูงกับ Degree of Membership ของ (ก) Crisp Set และ (ข) Fuzzy Set.....	9
รูปที่ 2.5 กราฟแสดง (ก) Crisp Set และ (ข) Fuzzy Set ของการจัดกลุ่มผู้ชายตัวเตี้ย, มีความสูงตาม มาตรฐาน, และตัวสูง.....	10
รูปที่ 2.6 รูปแสดงกระบวนการ Fuzzification.....	13
รูปที่ 2.7 รูปแสดงกระบวนการ Rule Evaluation.....	13
รูปที่ 2.8 รูปแสดงกระบวนการ Aggregation of the Rule Outputs.....	15
รูปที่ 2.9 รูปแสดงกระบวนการ Defuzzification.....	16
รูปที่ 2.10 ตัวอย่าง Graphic Equalizer Plug-in.....	16
รูปที่ 2.11 อุปกรณ์ NI myRIO.....	17
รูปที่ 2.12 ตำแหน่งพอร์ต Audio IN / Audio OUT ของ NI myRIO.....	18
รูปที่ 2.13 ตัวอย่าง Front panel ของ LabVIEW.....	19
รูปที่ 2.14 ตัวอย่าง Block Diagram ของ LabVIEW.....	20
รูปที่ 2.15 ตัวอย่างการใช้ Function สร้าง Block สมการ.....	21
รูปที่ 3.1 Block Diagram ของ Digital Parametric Equalizer.....	22
รูปที่ 3.2 Block Diagram ส่วน User Input.....	23
รูปที่ 3.3 รูปแสดงตัวอย่างการแบ่งย่านความถี่ของเครื่องดนตรีต่างๆ.....	24
รูปที่ 3.4 รูปแสดง Block Diagram ของส่วนการทำ Pre-processing.....	25
รูปที่ 3.5 รูปแสดงหน้าต่าง FIS Tool ของโปรแกรม MATLAB.....	28
รูปที่ 3.6 รูปแสดงหน้าต่าง FIS Tool ของโปรแกรม MATLAB ส่วนของการคำนวณหาค่า Gain.....	29
รูปที่ 3.7 รูปแสดง Input Membership Function ของค่าคะแนนความต้องการของ User.....	29
รูปที่ 3.8 รูปแสดง Input Membership Function ของค่าคะแนนความต้องการของ User.....	30
รูปที่ 3.9 รูปแสดง Output Membership Function ของระบบ Fuzzy Logic ส่วนการคำนวณหาค่า Gain.....	30
รูปที่ 3.10 รูปแสดงหน้าต่าง Rules ของส่วนการคำนวณ Gain บน FIS Tools.....	33
รูปที่ 3.11 รูปแสดงหน้าต่าง FIS Tool ของโปรแกรม MATLAB ส่วนของการคำนวณหาค่า Quality Factor.....	33

สารบัญรูป(ต่อ)

หน้า

รูปที่ 3.12	รูปแสดง Input Membership Function ของค่า Gain.....	34
รูปที่ 3.13	รูปแสดง Output Membership Function ของระบบ Fuzzy Logic ส่วนการคำนวณหา ค่า Quality Factor.....	34
รูปที่ 3.14	รูป Diagramของ (ก) Cascade Form และ (ข) Equivalent Transfer Function.....	35
รูปที่ 3.15	รูปแสดง หน้าต่างฟังก์ชัน App Designer ของ MATLAB.....	36
รูปที่ 3.16	รูปแสดงแถบ Component Library บน App Designer.....	36
รูปที่ 3.17	รูปแสดงการจัดเรียง Component บน App Designer.....	37
รูปที่ 3.18	รูปแสดงการเขียน Code บน App Designer.....	37
รูปที่ 3.19	รูปแสดงภาพรวม Block Diagram ของระบบ.....	39
รูปที่ 3.20	รูปแสดงส่วนบล็อกคำสั่งอินพุต Audio/in.....	39
รูปที่ 3.21	รูปแสดงบล็อกคำสั่งรับข้อมูลจาก Fader.....	39
รูปที่ 3.22	รูปแสดงบล็อกคำสั่ง MathScript ที่เชื่อมโยงเข้ากับ Fader.....	40
รูปที่ 3.23	รูปแสดงบล็อกคำสั่ง MathScript ที่เชื่อมโยงเข้ากับบล็อก IIR Filter.....	40
รูปที่ 3.24	รูปแสดงการเชื่อมต่อระหว่างบล็อกคำสั่ง IIR Filter.....	41
รูปที่ 3.25	รูปแสดงการเชื่อมต่อเพื่อแสดงกราฟผลการตอบสนองต่อความถี่ของตัวกรองสัญญาณ..	41
รูปที่ 3.26	รูปแสดงหน้าจอ User interface.....	42
รูปที่ 3.27	รูปการทดสอบการทำงานโดยเชื่อมต่ออุปกรณ์ NI myRIO เข้ากับคอมพิวเตอร์.....	42
รูปที่ 3.28	รูปแสดงหน้าจอ User interface เมื่อโปรแกรมทำงาน.....	43
รูปที่ 3.29	รูปการทดสอบโดยการเชื่อมต่ออุปกรณ์ NI my RIO เข้ากับเครื่อง Dynamic Signal Analyzer.....	43
รูปที่ 4.1	รูปแสดงสัญญาณเสียงที่ป้อนเข้าสู่ระบบในการทดสอบ.....	44
รูปที่ 4.2	รูปแสดงตัวอย่างสัญญาณเสียงที่ทำการแบ่งช่องทางดนตรีแล้ว.....	45
รูปที่ 4.3	รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงที่ทำการแบ่งช่องทางดนตรีแล้ว.....	45
รูปที่ 4.4	รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงที่ทำการหาค่าเฉลี่ยแล้ว.....	46
รูปที่ 4.5	รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงกีตาร์ที่ทำการหาค่าเฉลี่ยแล้ว.....	46
รูปที่ 4.6	รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงเบสที่ทำการหาค่าเฉลี่ยแล้ว.....	47
รูปที่ 4.7	รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงเปียโนที่ทำการหาค่าเฉลี่ยแล้ว.....	47
รูปที่ 4.8	รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงร้องหญิงที่ทำการหาค่าเฉลี่ยแล้ว.....	48
รูปที่ 4.9	รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงร้องชายที่ทำการหาค่าเฉลี่ยแล้ว.....	48
รูปที่ 4.10	รูปแสดงตัวอย่างการใช้ FIS Tools ในการหาค่า Gain.....	50

สารบัญรูป(ต่อ)

หน้า

รูปที่ 4.11	รูปแสดงตัวอย่างการใช้ FIS Tools ในการหาค่า Quality Factor.....	51
รูปที่ 4.12	รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงกีตาร์เข้าสู่ระบบ.....	53
รูปที่ 4.13	รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงเบสเข้าสู่ระบบ.....	53
รูปที่ 4.14	รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงเปียโนเข้าสู่ระบบ.....	54
รูปที่ 4.15	รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงร้องหญิงเข้าสู่ระบบ.....	54
รูปที่ 4.16	รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงร้องชายเข้าสู่ระบบ.....	55
รูปที่ 4.17	รูปแสดง Spectrum ของสัญญาณอินพุตเสียงกีตาร์ (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	56
รูปที่ 4.18	รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเบส (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	57
รูปที่ 4.19	รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเปียโน (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	58
รูปที่ 4.20	รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องหญิง (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	59
รูปที่ 4.21	รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องชาย (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	60
รูปที่ 4.22	รูปแสดงผลการทดลองที่ 1 มีค่า Gain แต่ละ Band คือ [12 0 0 12 0 -12 0] dB.....	61
รูปที่ 4.23	รูปแสดงผลการทดลองที่ 2 มีค่า Gain แต่ละ Band คือ [-9 7 10 -7 0 8 -10] dB.....	61
รูปที่ 4.24	รูปแสดงผลการทดลองที่ 3 มีค่า Gain แต่ละ Band คือ [-10 -5 10 0 -5 12 12] dB.....	62
รูปที่ 4.25	รูปแสดงผลการทดลองที่ 4 มีค่า Gain แต่ละ Band คือ [0 -9 1 8 -5 -10 -10] dB.....	62
รูปที่ 4.26	รูปแสดงผลการทดลองที่ 5 มีค่า Gain แต่ละ Band คือ [4 3 -5 7 7 -7 -2] dB.....	63
รูปที่ 4.27	รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 1.....	63
รูปที่ 4.28	รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 2.....	64
รูปที่ 4.29	รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 3.....	64
รูปที่ 4.30	รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 4.....	65
รูปที่ 4.31	รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 5.....	65
รูปที่ 4.32	รูปแสดง Spectrum ของสัญญาณอินพุตเสียงกีตาร์ (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	66

สารบัญรูป(ต่อ)

หน้า

รูปที่ 4.33 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเบส (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	67
รูปที่ 4.34 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเปียโน (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	68
รูปที่ 4.35 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องหญิง (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	69
รูปที่ 4.36 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องชาย (ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล.....	70



สารบัญตาราง

หน้า

ตารางที่ 2.1 ตัวอย่างการกำหนดค่าความเป็นสมาชิกของการจัดเซตของผู้ชายตัวสูง.....	8
ตารางที่ 3.1 ตารางแสดง Rules ที่ใช้ใน Fuzzy Inference System ของการหาค่า Gain.....	31
ตารางที่ 3.2 ตารางแสดง Rules ที่ใช้ในระบบ Fuzzy Logic ของการหาค่า Quality Factor.....	34
ตารางที่ 4.1 ค่า User input ในการทดลองกับสัญญาณอินพุตที่แตกต่างกัน.....	44
ตารางที่ 4.2 ค่าผลลัพธ์ที่ได้จากขั้นตอนการทำ Pre-Processing.....	49
ตารางที่ 4.3 ค่า Gain ที่หาจากการใช้ Fuzzy Logic.....	50
ตารางที่ 4.4 ค่า Quality Factor ที่ได้จากการใช้ Fuzzy logic.....	51
ตารางที่ 4.5 ค่า Parameters ทั้งหมดของ Parametric Equalizer.....	52



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

Digital Equalizer เป็นหนึ่งในอุปกรณ์ที่สำคัญในงานการผสมสัญญาณเสียง มีหน้าที่ช่วยในการแก้ไขข้อผิดพลาดทางความถี่และปรับเปลี่ยนคุณลักษณะทางความถี่ของสัญญาณเสียง เป็นอุปกรณ์ที่ทำงานบนพื้นฐานของ Digital Filter สามารถแบ่งประเภทของ Equalizer ได้เป็น 2 ประเภทตามความสามารถในการปรับค่า Parameters ได้แก่ Graphic Equalizer และ Parametric Equalizer

1.1 ความเป็นมาของปัญหา

ปัจจุบัน ในการทำงานด้านการผสมเสียงเสียง ได้มีการพัฒนาการทำงานจากระบบเดิมที่เป็นอนาล็อกมาสู่ระบบที่เป็น Digital ซึ่งหากพูดถึงการทำงานกับระบบที่เป็นดิจิทัลแล้ว หนึ่งในองค์ความรู้ที่สำคัญที่สุดคือการประมวลผลสัญญาณ Digital Signal Processing ซึ่งอุปกรณ์ที่ใช้งานในงานการผสมเสียงที่เป็น Digital ทั้งหมดนั้น ทำงานบนพื้นฐานองค์ความรู้การทำ Digital Signal Processing และอุปกรณ์ที่เป็นที่นิยมใช้งานที่สุดในงานด้านนี้ คืออุปกรณ์ที่เรียกว่า Equalizer ซึ่งสามารถแบ่งออกได้ 2 ชนิด คือ Parametric Equalizer และ Graphic Equalizer

Parametric Equalizer เป็น Equalizer ชนิดหนึ่งที่เป็นที่นิยมใช้ในงานการผสมเสียง เพราะความสามารถในการปรับค่า Parameters ที่ทำได้อย่างอิสระ แต่ด้วยความอิสระนี้ อาจส่งผลเสียต่อผู้ใช้งานเริ่ม ที่ขาดความเข้าใจในการปรับค่า Parameter นั้น ๆ ตรงจุดนี้เอง ที่เราได้เล็งเห็นถึงช่องทางในการพัฒนาให้ผู้ใช้งาน สามารถใช้งาน Parametric Equalizer ได้ง่ายขึ้น และสามารถเรียนรู้การปรับค่า Parameters ต่าง ๆ อย่างถูกต้องได้ง่ายขึ้น จึงได้มีแนวคิดที่จะนำปัญญาประดิษฐ์มาเป็นตัวช่วยในการปรับค่า Parameter เพื่อประโยชน์ของผู้ใช้งานเริ่มต้นดังที่กล่าวมา

Graphic Equalizer เป็น Equalizer อีกรูปแบบหนึ่งที่แตกต่างจากรูปแบบ Parametric ที่ไม่สามารถปรับค่า Parameter อื่นได้ นอกจากค่ากำลังขยายเพียงอย่างเดียว ซึ่งทำให้ง่ายต่อการใช้งานและการออกแบบ พวกเราจึงแนวคิดที่จะลองนำมาออกแบบเป็นฮาร์ดแวร์ โดยการทำงานบนอุปกรณ์ FPGA (Field Programmable Gate Array) เพื่อพอเป็นแนวทางให้สามารถนำ Graphic Equalizer มาทำงานเป็นฮาร์ดแวร์ที่สามารถประมวลผลแบบ Real Time ได้อีก

1.2 ความมุ่งหมายและวัตถุประสงค์

- 1.2.1 เพื่อศึกษาการออกแบบปัญญาประดิษฐ์โดยใช้ทฤษฎี Fuzzy logic
- 1.2.2 เพื่อศึกษาการนำ Digital Filter มาใช้ในการออกแบบ Digital Equalizer
- 1.2.3 เพื่อเป็นต้นแบบในการออกแบบ Digital Parametric Equalizer ที่มีปัญญาประดิษฐ์เป็นตัวช่วยในการปรับค่า Parameters

1.2.4 เพื่อเป็นต้นแบบในการออกแบบ Digital Graphic Equalizer ที่สามารถประมวลผลสัญญาณเสียงได้ในรูปแบบของการประมวลผลแบบ Real-Time Processing

1.4 ขอบเขตของการศึกษา

1.4.1 สามารถเขียนโปรแกรม Digital Parametric Equalizer ที่มีปัญญาประดิษฐ์ที่ออกแบบโดยใช้ทฤษฎีตรรกะคลุมเครือ มาช่วยในการปรับค่า Parameters ของ Equalizer ให้สามารถทำงานได้บนโปรแกรม MATLAB

1.4.2 สามารถเขียนโปรแกรม Digital Graphic Equalizer ให้สามารถทำงานประมวลผลสัญญาณเสียงบนอุปกรณ์ FPGA (Field Programmable Gate Array) ในรูปแบบการประมวลผลแบบ Real-time

1.5 ขั้นตอนการศึกษา

1.5.1 ศึกษาการออกแบบ Filter เพื่อนำไปใช้ในการสร้าง Equalizer

1.5.2 ศึกษารายละเอียดทฤษฎีของปัญญาประดิษฐ์โดยใช้ทฤษฎี Fuzzy logic และการนำมาประยุกต์ใช้ในการดำเนินงาน

1.5.3 เขียนโปรแกรม Digital Parametric Equalizer โดยใช้โปรแกรม MATLAB

1.5.4 เขียนโปรแกรม Digital Graphic Equalizer โดยใช้โปรแกรม LabVIEW

1.5.5 ทดสอบและตรวจสอบผลการทำงานของโปรแกรม

1.5.6 ทำการแก้ไขปรับปรุงข้อผิดพลาดของโปรแกรมที่พบ

1.5.7 ทำการออกแบบ User Interface ของ Digital Parametric Equalizer และ Digital Graphic Equalizer

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 Digital signal processing (DSP)

Digital Signal Processing คือการใช้ระบบประมวลผลสัญญาณที่ประกอบด้วยระบบปฏิบัติการทางคณิตศาสตร์ โดยมีสัญญาณอินพุตและสัญญาณ output เป็นสัญญาณที่เป็น Discrete Time ในปัจจุบันอุตสาหกรรมดนตรีมีการบันทึกเสียงเป็นข้อมูล Digital จึงมีการใช้การประมวลผลสัญญาณ Digital มาเกี่ยวข้องด้วย 2 ประการ ได้แก่ 1. การแปลงสัญญาณ แปลงสัญญาณจากสัญญาณ Analog ที่เป็นสัญญาณ Continuous Time ไปเป็นสัญญาณ Digital ที่เป็นสัญญาณ Discrete Time เพื่อนำไปบันทึกเป็นข้อมูล Digital 2. การกรองสัญญาณ เป็นการนำ Digital Filter มาใช้กับสัญญาณเสียงเพื่อปรับปรุงคุณภาพของเสียงที่บันทึก เช่น การใช้ Equalizer กับเสียงที่ถูกบันทึก

2.2 Digital Filter

Digital Filter เป็นระบบประมวลผลสัญญาณ Digital ที่เป็น Linear Time-Invariant System ซึ่งสามารถอธิบายระบบได้ด้วย impulse response โดยสัญญาณเอาต์พุต $y(n)$ จะอยู่ในรูปของการ convolution ของสัญญาณอินพุต $x(n)$ กับ Impulse Response $h(n)$ ดังนี้

$$y(n) = h(n) * x(n) \quad (2.1)$$

Impulse Response สามารถนำมาวิเคราะห์และอธิบายบน Frequency Domain เป็น Frequency Response ซึ่งประกอบด้วย Magnitude Response และ phase response โดยสามารถอธิบายบน Z Domain เป็น Transfer Function ดังนี้

$$H(z) = \frac{\sum_{k=0}^{M-1} b_k z^{-k}}{\sum_{k=0}^{N-1} a_k z^{-k}} \quad (2.2)$$

Digital Filter คือระบบประมวลผลสัญญาณ Digital ที่มีการดัดแปลง Frequency Response ในการใช้งานกับสัญญาณเสียงจะให้ความสำคัญกับ Magnitude Response มากกว่า Phase Response ซึ่งสามารถจำแนกได้ตามลักษณะของ Impulse Response เป็น 2 ประเภท ได้แก่ FIR Filter (Finite Impulse Response Filter) และ IIR filter (Infinite Impulse Response Filter)

FIR Filter เป็น Digital Filter ที่มีจำนวนสมาชิกของ Impulse Response จำกัดและ Phase Response มีความสัมพันธ์กับความถี่เป็นปริมาณเชิงเส้น (Linear Phase Response) แต่เพื่อให้ Transition Band มีความแคบเพียงพอจำเป็นต้องใช้ FIR Filter ที่มี Filter Length มาก ซึ่งทำให้สัญญาณ output เกิด Delay และมีการใช้หน่วยความจำค่อนข้างมาก

IIR Filter เป็น Digital Filter มีจำนวนสมาชิกของ Impulse Response เป็นอนันต์ และไม่มีคุณสมบัติ Linear Phase Response แต่ในการออกแบบ Digital Filter ให้มี Transition Band ที่แคบไม่จำเป็นต้องใช้ Filter Length ที่ยาวมากนัก จึงไม่มีปัญหาเรื่องปริมาณการใช้หน่วยความจำ และการเกิด Delay ของสัญญาณ output

การประยุกต์ใช้ Digital Filter ในด้านเสียง เป็นการนำ Digital Filter มาใช้กับสัญญาณเสียง โดยตรงทั้งในรูปแบบของตัวกรอง เช่น Low Pass Filter, High Pass Filter หรือ Equalizer เป็นต้น ซึ่งมีการนำไปใช้อย่างแพร่หลายทั้ง FIR Filter และ IIR Filter โดยขึ้นอยู่กับความจำเป็นของคุณสมบัติ Linear Phase Response และข้อจำกัดของหน่วยความจำของอุปกรณ์ที่ใช้

2.2.1 Peaking Filter

Digital Filter ที่เลือกนำมาใช้ในการสร้าง Digital Equalizer ทั้ง Parametric Equalizer และ Graphic Equalizer เป็น IIR Filter ที่เรียกว่า Peaking Filter ซึ่งมีคุณสมบัติเป็น All Pass Filter เมื่อปรับค่า Gain = 0 dB และจะมีลักษณะเป็น Bell Shape Filter เมื่อทำการปรับเพิ่มหรือลด Gain จาก 0 dB มี Transfer Function ดังนี้

$$H(z) = \frac{\left(\frac{b_0}{a_0}\right) + \left(\frac{b_1}{a_0}\right)z^{-1} + \left(\frac{b_2}{a_0}\right)z^{-2}}{1 + \left(\frac{a_1}{a_0}\right)z^{-1} + \left(\frac{a_2}{a_0}\right)z^{-2}} \quad (2.3)$$

$$\begin{aligned} A &= \sqrt{10^{(G/20)}} & b_0 &= 1 + (\alpha \times A) \\ \omega_c &= 2 \times \pi \times (f_c/f_s) & b_1 &= (-2) \times \omega_c \\ \omega_s &= \sin(\omega_c) & b_2 &= 1 - (\alpha \times A) \\ \omega_c &= \cos(\omega_c) & a_0 &= 1 + (\alpha/A) \\ \alpha &= \omega_s / (2 \times Q) & a_1 &= (-2) \times \omega_c \\ & & a_2 &= 1 - (\alpha/A) \end{aligned}$$

โดย

G คือ Gain

f_s คือ Frequency Sampling Rate

f_c คือ Frequency Center

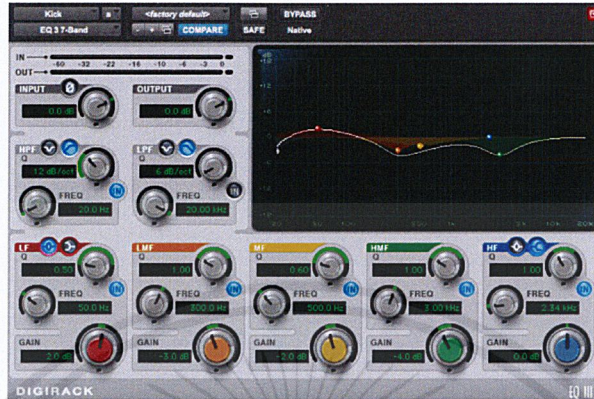
Q คือ Quality Factor

2.3 Parametric Equalizer

Parametric Equalizer เป็น Equalizer ประเภทหนึ่ง ที่มีรูปแบบการปรับค่า Parameters ที่อิสระ สามารถปรับ Parameters ได้ทั้งหมด 3 ตัว ได้แก่ Center Frequency, Quality Factor และ Gain โดย Parameters แต่ละตัวจะมีหน้าที่ที่แตกต่างกัน

2.3.1 Center Frequency

Center Frequency เป็น Parameter ที่ใช้ในการปรับเลือกความถี่ที่สนใจ โดยผู้ใช้งาน จำเป็นต้องมีความรู้ความเข้าใจว่าการปรับค่าความถี่ในแต่ละย่านจะส่งผลอย่างไรต่อสัญญาณเสียงบ้าง เพื่อให้สามารถเลือกความถี่ที่ถูกต้องตรงตามความต้องการของผู้ใช้งานได้

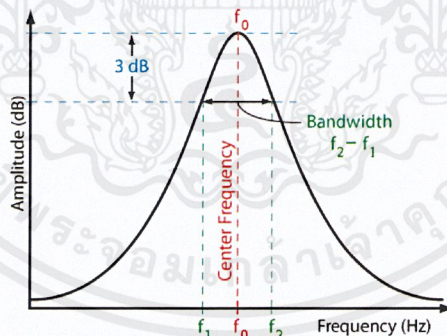


รูปที่ 2.1 รูปตัวอย่าง plug-in Parametric Equalizer

ที่มา : <https://bit.ly/2YDKtZv>

2.3.2 Quality Factor

Quality Factor เป็น Parameter ที่ใช้ในการปรับ Bandwidth หรือความกว้างแคบของช่วงความถี่ เมื่อ Quality Factor มีค่าที่ต่ำช่วงความถี่จะกว้างและช่วงความถี่จะแคบเมื่อ Equalizer มีค่า Quality Factor สูง ซึ่งผู้ใช้งานควรปรับค่า Quality Factor ให้ Quality Factor มี Bandwidth ของแต่ละช่วงความถี่ไม่ทับซ้อนกัน เพื่อให้การปรับค่า Gain ไม่ส่งผลกับช่วงความถี่ข้างเคียงที่ไม่ต้องการ



รูปที่ 2.2 รูปแสดง Bandwidth ที่เป็นผลลัพธ์ของการปรับ Quality Factor

ที่มา : <https://bit.ly/30ufIgT>

2.3.3 Gain

Parameter Gain ใช้ในการควบคุมความดังเบาของช่วงความถี่ที่ทำการปรับค่า ส่งผลให้ Amplitude เกิดการเพิ่มขึ้นหรือลดลงที่ช่วงความถี่ที่ทำการปรับค่า สามารถแสดงบน Frequency Domain ได้ ซึ่งหากผู้ใช้งานทำการปรับค่า Gain มากเกินไป อาจทำให้ Amplitude ของสัญญาณเสียงมีขนาดใหญ่เกิน Dynamic Range ของระบบเสียงที่ใช้งานอยู่รองรับได้ จะเกิด Distortion ขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการอธิบายถึงหลักการปรับค่า Parameters ต่างๆ แสดงให้เห็นว่า การใช้งาน Parametric Equalizer ในแต่ละครั้งมีความยุ่งยากและต้องอาศัยความรู้และประสบการณ์ของผู้ใช้งานในการปรับค่า Parameters ต่างๆ มากพอสมควรซึ่งมักเป็นปัญหาสำหรับผู้เพิ่งเริ่มต้นใช้งาน ทำให้ไม่สามารถปรับค่า Parameters ได้อย่างมีประสิทธิภาพหรือตรงตามความต้องการของตนได้ เราจึงได้นำปัญหาประดิษฐ์มาใช้ในการช่วยคิดค่า Parameters ต่างๆ เพื่อเป็นตัวช่วยในการตัดสินใจและเลือกใช้ค่า Parameters

Parametric Equalizer ที่ออกแบบ จะมีทั้งหมด 3 Bands ที่มีปัญหาประดิษฐ์ช่วยในการปรับ Parameter ทั้ง 3 ตัว ได้แก่ Center Frequency, Quality Factor และ Gain โดยให้ผู้ใช้งานระบุความต้องการเพิ่มหรือลดความดังของแต่ละย่านความถี่ของเสียงที่กำลังปรับใช้แทน เพื่อลดความยุ่งยากในการปรับ Parameters ของ Parametric Equalizer

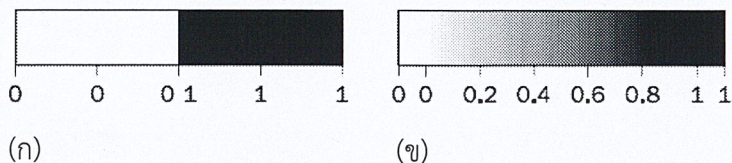
2.4 Fuzzy Logic

Fuzzy Logic เป็นทฤษฎีของการจัดข้อมูลในรูปแบบของ Fuzzy set ซึ่งเป็นเซตของข้อมูลที่มีความคลุมเครือ เป็นหนึ่งในทฤษฎีที่ถูกนำมาใช้ในการสร้างปัญญาประดิษฐ์ ในรูปแบบหนึ่งของ Expert System เป็นตรรกะที่จำลองความคิดของมนุษย์ให้กับเครื่องจักรในเรื่องของ ค่าระดับหรือปริมาณ ให้เครื่องจักรสามารถทำงานได้สอดคล้องกับความคิดของมนุษย์มากขึ้น แตกต่างจาก Expert System รูปแบบเดิมที่ระบบทำงานตามกฎที่อ้างอิงตรรกะถูกและผิดในการทำงาน

หากพิจารณาจากตรรกะของมนุษย์โดยทั่วไป ความรู้สึกจะเป็นหนึ่งในตัวแปรที่มีผลต่อการระบุค่าปริมาณของข้อมูลหรือการแก้ปัญหาใดๆ ที่พบ เช่น เมื่อรู้สึกร้อนก็อาจจะปรับอุณหภูมิของเครื่องปรับอากาศให้ต่ำลงเล็กน้อยเพื่อให้รู้สึกเย็นขึ้น หากว่านี่เป็นการพูดคุยหรืออภิปรายระหว่างมนุษย์ด้วยกันก็ไม่ใช่ว่าเรื่องยากที่จะทำความเข้าใจปัญหาและทำการแก้ไข แต่ถ้าหากโจทย์ปัญหานี้เป็นโจทย์ที่คอมพิวเตอร์ต้องการคำตอบและดำเนินการแก้ไขก็จำเป็นที่จะต้องทำให้คอมพิวเตอร์เข้าใจคำว่า ร้อน ,เล็กน้อย และเย็นขึ้น เพื่อให้สามารถดำเนินการได้อย่างถูกต้อง ณ จุดนี้จึงเป็นที่มาของทฤษฎีที่สอนคอมพิวเตอร์ให้รู้จักกับค่าประมาณหรือความคลุมเครืออันได้แก่ Fuzzy Logic Theorem

ปกติแล้วตรรกะที่เราคุ้นเคยคือ Boolean Logic จะเป็นตรรกะที่มีได้เพียง 2 คำตอบ อาทิ ใช่หรือไม่ เป็นตรรกะที่มีการขีดเส้นระบุความเป็นและไม่เป็นสมาชิกของคำตอบไว้อย่างชัดเจน เช่น การจัดกลุ่มคนตามความสูง หากเราขีดเส้นระบุไว้ว่าคนที่สูงตั้งแต่ 180 เซนติเมตรขึ้นไป จัดเป็นคนที่สูง นั่นหมายความว่าคนที่มีความสูงที่ 179 เซนติเมตรก็จะถูกจัดอยู่ในกลุ่มคนที่เตี้ยไปในทันที ถึงแม้ว่าจะมีความต่างเพียง 1 เซนติเมตรเท่านั้น ซึ่งในทางปฏิบัติแล้วสามารถระบุได้ว่าไม่มีความต่างเลยก็ได้ ด้วยความเป็นเส้นตรงของ Boolean Logic ทำให้เกิดเป็นข้อจำกัดในการใช้งานในหลายๆกรณี ซึ่งเป็นจุดที่ Fuzzy Logic สามารถนำมาใช้เพื่อจัดการกับข้อจำกัดที่ว่านี้ได้ Fuzzy Logic นั้นเป็นตรรกะที่มี

ลักษณะเป็นตรรกะที่มีค่าระดับหลายค่า (multi-valued) แต่ละข้อมูลจะมีค่าระดับความเป็นสมาชิก (Degree of membership) ที่แตกต่างกัน ต่างจาก Boolean Logic ที่มีเพียง 2 ค่า คือเป็นสมาชิก กับไม่เป็นสมาชิก หรือที่เรียกว่า Crisp แต่ Fuzzy Logic จะใช้ความต่อเนื่องระหว่างค่าตรรกะ 0 (ผิดสัมบูรณ์) และ 1 (ถูกสัมบูรณ์) ดังตัวอย่างในรูปที่ 2.3(ข) ได้แสดงถึงการไล่ระดับสีที่แสดงให้เห็นว่า ข้อมูลสามารถเป็นสมาชิกของคำตอบถูกและผิดได้พร้อมๆกัน แต่ข้อมูลจะมีค่าความเป็นสมาชิกของแต่ละฝั่งที่มากน้อยไม่เท่ากัน



รูปที่ 2.3 ขอบเขต Degree of Membership ของ (ก) Boolean Logic และ (ข) Fuzzy Logic

2.4.1 Fuzzy set

หากกล่าวถึงเรื่องของการจัดกลุ่มความเป็นสมาชิกของข้อมูลในทางคณิตศาสตร์ก็ต้องกล่าวถึงเรื่องของเซต หากว่าเป็นเซตในรูปแบบทั่วไปหรือที่เรียกว่า Crisp Set จะกล่าวถึงแค่ความเป็นสมาชิกว่าเป็นหรือไม่เป็นสมาชิกเท่านั้น ไม่สามารถจัดกลุ่มหรือแสดงผลรูปแบบข้อมูลที่มีความคลุมเครือและไม่สามารถจัดการกับข้อมูลที่มีความขัดแย้งในตัวได้ จึงทำให้เกิดเป็นรูปการจัดกลุ่มใหม่ของข้อมูลที่มีความคลุมเครือในรูปของ Fuzzy Set โดยแนวคิดพื้นฐานของ Fuzzy Set คือข้อมูลใดๆที่เป็นสมาชิกของ Fuzzy set จะมีการระบุค่าระดับความเป็นสมาชิก ซึ่งมักจะระบุเป็นจำนวนจริงที่อยู่ในช่วง $[0,1]$ ดังตัวอย่างในตารางที่ 2.1

ตารางที่ 2.1 ตัวอย่างการกำหนดค่าความเป็นสมาชิกของการจัดเซตของผู้ชายตัวสูง

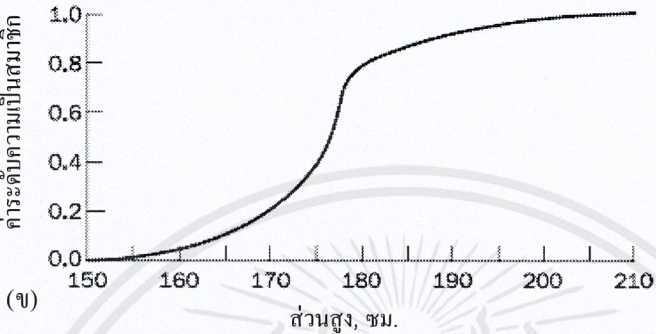
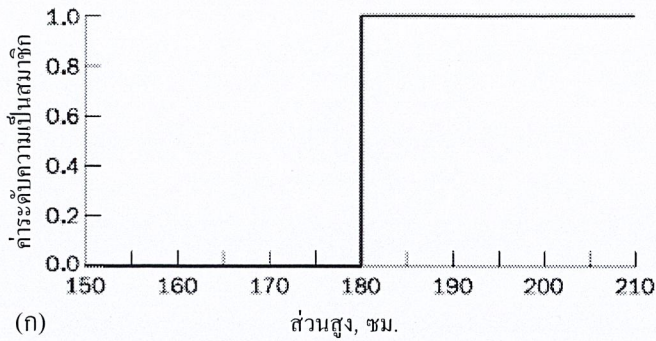
ความสูง, ซม.	ค่าความเป็นสมาชิก	
	Crisp	Fuzzy
208	1	1
208	1	1
198	1	0.98
181	1	0.82
179	0	0.78
172	0	0.24
167	0	0.15
158	0	0.06
155	0	0.01
152	0	0

เมื่อพิจารณาในรูปแบบของ Crisp Set หากตั้งคำถามว่า ‘ผู้ชายที่มีความสูงเท่าไรถึงจะเรียกว่าคนตัวสูง?’ คำตอบจะได้เป็น ผู้ชายที่มีส่วนสูงมากกว่า 180 เซนติเมตรจะถือว่าเป็นคนตัวสูง หากมีส่วนสูงน้อยกว่านั้นจะไม่ถือว่าเป็นคนตัวสูง ในทางกลับกันหากพิจารณาในรูปแบบของ Fuzzy set ถ้าถามว่า ‘ผู้ชายที่ส่วนสูง 181 เซนติเมตร ถือว่าเป็นคนตัวสูงหรือไม่?’ คำตอบจะอยู่ในรูปแบบของค่าความเป็นสมาชิกของ Fuzzy set ซึ่งจะตอบว่า ผู้ชายที่สูง 181 เซนติเมตรจะมีค่าความเป็นสมาชิกของ Fuzzy set อยู่ที่ 0.82 จากข้อมูลในตารางที่ 2.1 สามารถทำไปแสดงความสัมพันธ์ในรูปกราฟดังรูปที่ 2.4

หากนำข้อมูลที่ถูกรวบรวมในรูปแบบของ Crisp Set มาเขียนเป็นฟังก์ชัน กำหนดให้ x เป็นยูนิเวอร์ส และ x เป็นสมาชิกภายในยูนิเวอร์ส Crisp Set A จะอยู่ในรูปของฟังก์ชัน $f_A(x)$ เรียกว่า Characteristic Function ของเซต A

$$f_A(x): X \rightarrow 0,1 \tag{2.4}$$

$$\text{ที่ } f_A(x) = \begin{cases} 1, & \text{เมื่อ } x \in A \\ 0, & \text{เมื่อ } x \notin A \end{cases}$$



รูปที่ 2.4 กราฟแสดงความสัมพันธ์ระหว่างข้อมูลความสูงกับ Degree of Membership ของ
(ก) Crisp Set และ (ข) Fuzzy Set

เซตรูปแบบนี้จะเชื่อมโยงยูนิเวอร์ส X กับเซตที่มีสมาชิก 2 ตัว สำหรับสมาชิก x ใดๆที่อยู่ใน ยูนิเวอร์ส X Characteristic function $f_A(x)$ จะเท่ากับ 1 เมื่อ x เป็นสมาชิกของเซต A และเท่ากับ 0 เมื่อ x ไม่เป็นสมาชิกของเซต A

ถ้าหากข้อมูลถูกจัดให้อยู่ในรูปของ Fuzzy Set จะทำให้ Fuzzy Set A ที่อยู่ภายในยูนิเวอร์ส X จะอยู่ในรูปของฟังก์ชัน $u_A(x)$ ที่ถูกเรียกว่า Membership Function

$$u_A(x): X \rightarrow 0,1 \quad (2.5)$$

ที่

$$u_A(x) = 1 \text{ เมื่อ } x \text{ เป็นสมาชิกของ } A$$

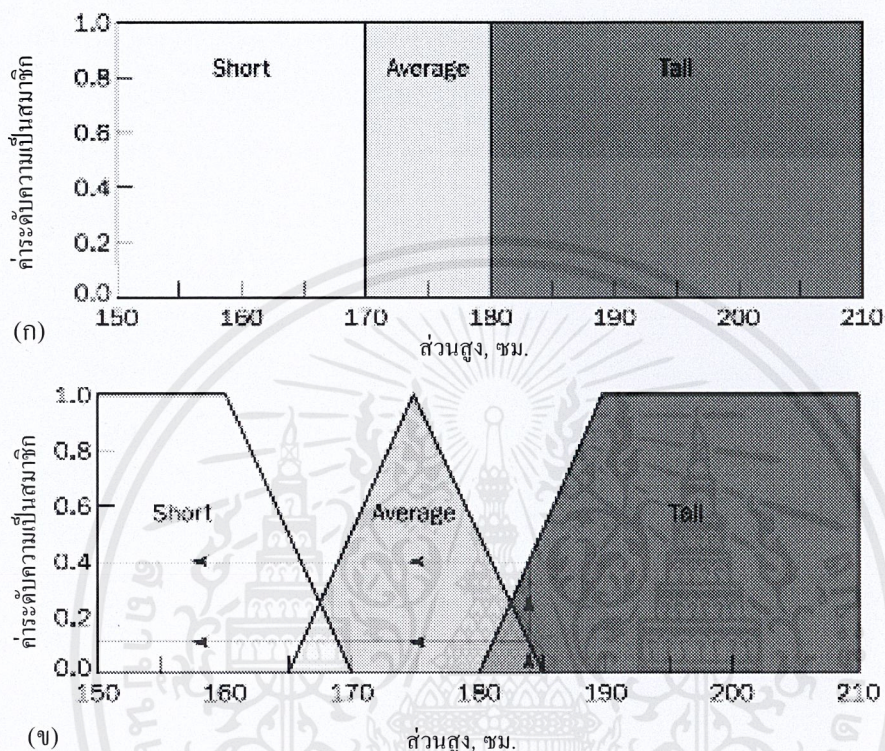
$$u_A(x) = 0 \text{ เมื่อ } x \text{ เป็นสมาชิกของ } A$$

$$0 < u_A(x) < 1 \text{ เมื่อ } x \text{ เป็นสมาชิกบางส่วนของ } A$$

เซตที่อยู่ในรูปของ Fuzzy Set จะมีความต่อเนื่องของข้อมูลที่เป็นไปได้ สำหรับสมาชิก x ใดๆใน ยูนิเวอร์ส X Membership function $u_A(x)$ จะมีค่าเท่ากับค่าระดับความเป็นสมาชิกของเซต A ซึ่งค่าระดับนี้จะเป็ค่าที่อยู่ในช่วงระหว่าง 0 ถึง 1

จากตัวอย่างเดิม ในเรื่องของการจัดกลุ่มความสูงผู้ชาย จากตัวอย่างที่ผ่านมาเราได้แสดง Fuzzy Set ของกลุ่ม *ผู้ชายตัวสูง* ไปแล้ว ในทำนองเดียวกันหากเราสามารถเพิ่ม Fuzzy Set ของ *ผู้ชายตัวเตี้ย* และ *ผู้ชายที่มีส่วนสูงมาตรฐาน* เพิ่มเข้ามาในยูนิเวอร์สได้ ซึ่งผลการจัด Fuzzy Set แสดงอยู่ใน

รูปที่ 2.5 แสดงควมคู่ไปกับการจัด Crisp set ของข้อมูลเดียวกัน การจัดเซตของข้อมูลทั้ง 2 รูปแบบได้แบ่งข้อมูลเป็น 3 เซตภายในยูนิเวอร์สเดียวกัน หากพิจารณาจาก Fuzzy Set จะเห็นได้ว่าผู้ชายที่มีความสูง 184 เซนติเมตร เป็นสมาชิกของเซต *ผู้ชายที่มีส่วนสูงมาตรฐาน* ด้วยค่าระดับความเป็นสมาชิกที่ 0.1 และในเวลาเดียวกันก็เป็นสมาชิกของเซต *ผู้ชายตัวสูง* ที่มีค่าระดับความเป็นสมาชิกที่ 0.4 เช่นกัน กล่าวคือผู้ชายที่มีความสูง 184 เซนติเมตร จะเป็นมีความเป็นสมาชิกอยู่ในทั้ง 2 เซต



รูปที่ 2.5 กราฟแสดง (ก) Crisp Set และ (ข) Fuzzy Set ของการจัดกลุ่มผู้ชายตัวเตี้ย, มีความสูงตามมาตรฐาน, และตัวสูง

การจัดสมาชิกของข้อมูลให้อยู่ใน Fuzzy Set ทำจากการรวบรวมองค์ความรู้จากผู้เชี่ยวชาญในองค์ความรู้นั้นๆ จัดกลุ่มข้อมูลเพื่อนำไปสู่การหา Membership Function เพื่อนำไปใช้ดำเนินการต่อไป

2.4.2 Linguistic Variable และ Fuzzy Rules

ที่มาของ Fuzzy Logic Theorem นั้นเริ่มมาจาก Linguistic Variable ซึ่งเป็น Fuzzy Variable เช่น หากบอกว่า 'อากาศหนาว' จะหมายถึง Linguistic Variable *อากาศ* มีคุณสมบัติตาม Linguistic Variable *หนาว* Linguistic Variable เหล่านี้จะถูกนำมานิยามเป็น Rules ของระบบ ค่าขอบเขตปริมาณที่เป็นไปได้ของ Linguistic Variable จะถูกระบุเอาไว้ในยูนิเวอร์สของตัวแปรนั้นๆ เช่น ยูนิเวอร์สของตัวแปร *ความเร็ว* (Speed) จะเป็นค่าที่อยู่ในช่วงตั้งแต่ 0 ถึง 220 กิโลเมตรต่อชั่วโมง สามารถออกแบ่งช่วงเซตของความเร็วออกได้เป็น *ช้ามาก, ช้า, ปานกลาง, เร็ว, เร็วมาก* เซตเซตแต่ละตัวของ Fuzzy Set จะแสดง Linguistic Quantity ที่เชื่อมโยงกับ Linguistic Variable นั้นๆ

การออกแบบ Rules ที่ใช้ Fuzzy Logic จะเรียกว่า Fuzzy Rules มีความคล้ายคลึงกับ Rules ของระบบ Rules base ทั่วไปที่ทำงานในรูปแบบ IF-THEN

IF x is A
THEN y is B

สิ่งที่ทำให้ Fuzzy Rules แตกต่างจาก Rules ทั่วไปอย่างชัดเจน คือการใช้ Linguistic Quantity

Rule: 1

IF speed is > 100

THEN stopping_distance is long

Rule: 2

IF speed is < 40

THEN stopping_distance is short

ในการออกแบบ Rules หากเป็นกฎทั่วไปจะเป็นดังตัวอย่าง

การออกแบบ Rules ตามหลักทั่วไปจะอ้างอิงรูปแบบตาม Boolean Logic ที่มีการขีดเส้นแบ่งผลลัพธ์เอาไว้อย่างชัดเจน แต่ถ้าหากเป็นการออกแบบ Rules ในรูปแบบ Fuzzy Rules จะเป็นดังตัวอย่าง

Fuzzy Rules จะนำ Linguistic Variable มาสร้างเป็นกฎ ค่าปริมาณของตัวแปรจะถูกระบุไว้ในยูนิเวอร์ส และจะมีการแบ่งปริมาณออกเป็นช่วงๆ ซึ่งรวมไปถึงค่าปริมาณของคำตอบด้วยเช่นกัน คำตอบของกฎเองก็อยู่ในรูปของ Linguistic Variable ที่มีค่าปริมาณระบุเอาไว้ในยูนิเวอร์สของคำตอบและมีการแบ่งคำตอบออกเป็นช่วงๆ ในทำนองเดียวกับตัวตั้งด้วย กล่าวคือ Fuzzy Rules จะมีความสัมพันธ์เชื่อมโยงกับ Fuzzy Set ที่ได้ทำการออกแบบเอาไว้

Rule: 1

IF speed is fast

THEN stopping_distance is long

Rule: 2

IF speed is slow

THEN stopping_distance is short

2.4.3 Mamdani-Style Inference

Fuzzy inference คือวิธีการหา output ของระบบจากอินพุตที่ป้อนเข้าสู่ระบบโดยอ้างอิงจาก Fuzzy Set ซึ่งมีวิธีการทำอยู่ 2 รูปแบบที่เป็นที่นิยม คือ Mamdani-style Inference และ Sugeno-style Inference ในการดำเนินโครงการนี้ได้อ้างอิงวิธีการในรูปแบบ Mamdani-style Inference จึงขอกล่าวถึงเพียงรูปแบบเดียว

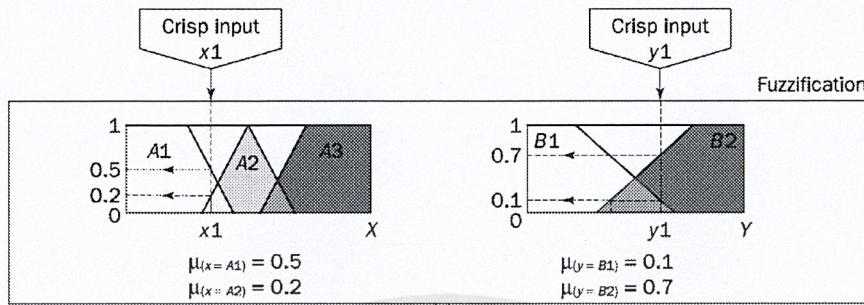
Mamdani-style Inference เป็นรูปแบบการทำ Fuzzy Inference ที่เป็นที่นิยมที่สุด ซึ่งมีวิธีการทำ 4 ขั้นตอน ได้แก่ Fuzzification, Rule Evaluation, Aggregation of the Rule Outputs และ Defuzzification ซึ่งขั้นตอนการดำเนินงานจะแสดงจากตัวอย่างปัญหา การหาค่าความเสี่ยงในการดำเนินงาน ที่มีกฎดังนี้

Rule: 1	Rule:1
IF x is A3	IF project_funding is adequate
OR y is B1	OR project_staffing is small
THEN z is C1	THEN risk is low
Rule: 2	Rule: 2
IF x is A2	IF project_funding is marginal
AND y is B2	AND project_staffing is large
THEN z is C2	THEN risk is normal
Rule: 3	Rule: 3
IF x is A1	IF project_funding is inadequate
THEN z is C3	THEN risk is high

เมื่อ x, y และ z (project funding, project staffing และ risk) เป็น Linguistic Variable และ $A1, A2$ and $A3$ (inadequate, marginal and adequate) เป็น linguistic values

2.4.3.1 Fuzzification

อินพุตที่เข้าสู่ระบบจะอยู่ในรูปของ Crisp Input ซึ่งเป็นค่าจำนวนจริงใดๆ Fuzzification เป็นกระบวนการที่หาว่าอินพุตที่ถูกป้อนเข้ามาเป็นสมาชิกของ Fuzzy Set ใดและมีค่าระดับความเป็นสมาชิกของ Fuzzy Set นั้นเท่าใด

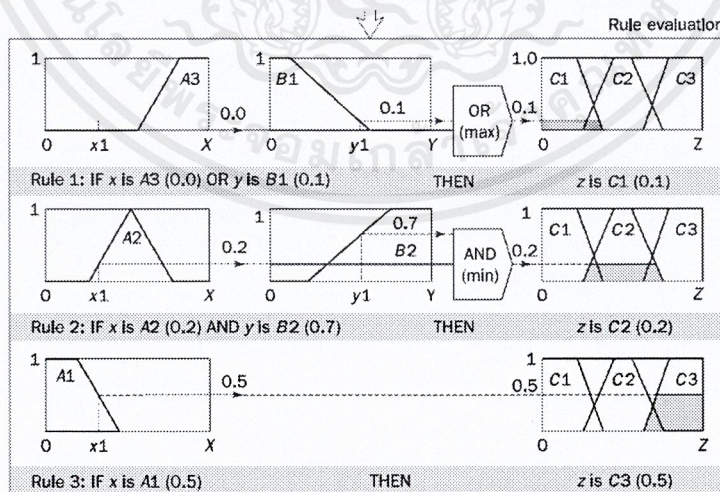


รูปที่ 2.6 รูปแสดงกระบวนการ Fuzzification

จากตัวอย่างในรูปที่ 2.6 Crisp Input x_1 มีค่าอยู่ที่ 35% ซึ่งมี Degree of Membership ของเซต A1 และ A2 อยู่ที่ 0.5 และ 0.2 ตามลำดับ ส่วน Crisp Input y_1 มีค่าอยู่ที่ 60% ซึ่งมี Degree of Membership ของเซต B1 และ B2 อยู่ที่ 0.1 และ 0.7 ตามลำดับ

2.4.3.2 Rule Evaluation

หลังจากหา Degree of Membership ของ Crisp Input เป็นที่เรียบร้อยแล้ว ในกระบวนการ Rule Evaluation เป็นการหาผลลัพธ์จากการดำเนินการตาม Rules ที่กำหนดไว้ ถ้า Rule ที่ระบุไว้เป็น Rule ที่มีความเชื่อมโยงระหว่างยูนิเวอร์สตั้งแต่ 2 ยูนิเวอร์สขึ้นไป จะใช้ Fuzzy Operator AND หรือ OR ในการหา Rule Consequent ออกมาในรูปของค่าคงตัวที่เป็นจำนวนจริง และค่าคงตัวนี้จะถูกนำไประบุ Degree of Membership ของ Rule Consequent นั้น



รูปที่ 2.7 รูปแสดงกระบวนการ Rule Evaluation

เมื่อใช้ Operation OR ในการหา Degree of Membership ของ Rule Consequent จะสามารถหาค่าได้จากสมการ

$$\mu_{(A \cup B)}(x) = \max[\mu_A(x), \mu_B(x)] \quad (2.6)$$

เมื่อใช้ Operation AND ในการหา Degree of Membership ของ Rule Consequent จะสามารถหาค่าได้จากสมการ

$$\mu_{(A \cup B)}(x) = \min[\mu_A(x), \mu_B(x)] \quad (2.7)$$

จากข้อมูลอินพุตที่ผ่านกระบวนการ Fuzzification แล้ว $\mu_{(x=A1)} = 0.5, \mu_{(x=A2)} = 0.2, \mu_{(y=B1)} = 0.1, \mu_{(x=B1)} = 0.7$ นำเข้าไปแทนในกฎได้ผลลัพธ์ดังนี้

Rule: 1

IF x is A3 (0.0)

OR y is B1 (0.1)

THEN z is C1 (0.1)

$$\mu_{(C1)}(z) = \max[\mu_{A3}(x), \mu_{(B1)}(y)] = \max[0.0, 0.1] = 0.1$$

Rule: 2

IF x is A2

AND y is B2

THEN z is C2

$$\mu_{(C2)}(z) = \min[\mu_{A3}(x), \mu_{(B1)}(y)] = \max[0.2, 0.7] = 0.2$$

Rule: 3

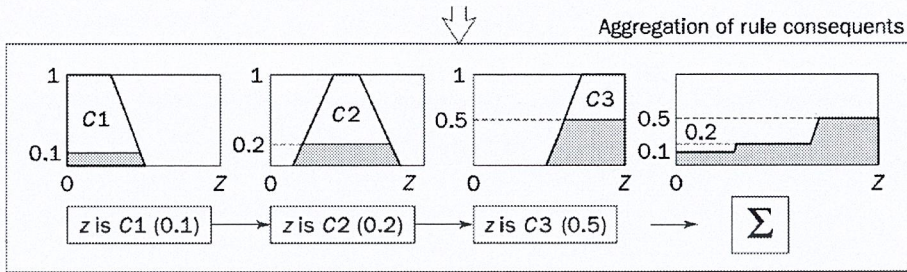
IF x is A1

THEN z is C3

$$\mu_{(C2)}(z) = 0.5$$

2.4.3.3 Aggregation of the Rule Outputs

กระบวนการนี้เป็นกระบวนการที่รวม Rule Consequent ทั้งหมดเข้าด้วยกัน กล่าวคือนำ Degree of Membership ของ Rule Consequent มารวมเข้าด้วยกันเป็น Fuzzy Set เพียงชุดเดียว



รูปที่ 2.8 รูปแสดงกระบวนการ Aggregation of the Rule Outputs

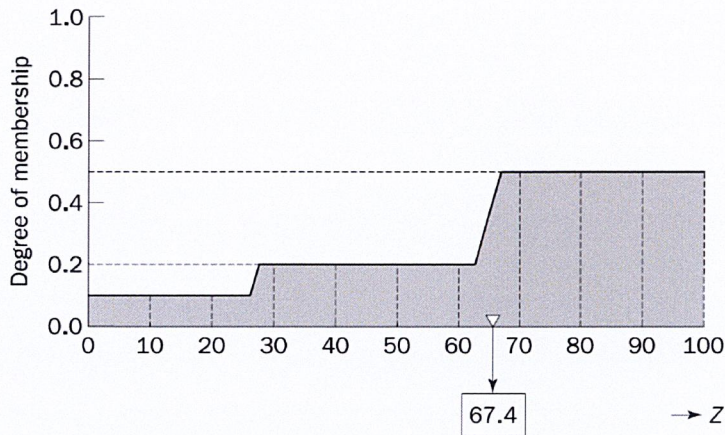
2.4.3.4 Defuzzification

ขั้นตอนสุดท้ายของกระบวนการคือการทำให้ Defuzzification โดยผลลัพธ์ที่ออกจาก Fuzzy System จะเป็นค่าคงตัวที่เป็นจำนวนจริงที่เรียกว่า Crisp number ซึ่งวิธีการได้มาซึ่งค่า output ของระบบนั้นเรียกว่า Centroid Technique เป็นวิธีการหาจุดบนแกนราบของกราฟที่ลากเส้นลงมาตามแกนตั้งจะแบ่งผลรวมเซตออกเป็น 2 ส่วนเท่าๆกัน หรือในทางคณิตศาสตร์เรียกว่า Centre of gravity (COG) เป็นไปตามสมการที่ 2.6

$$COG = \frac{\int_a^b \mu_A(x) x dx}{\int_a^b \mu_A(x) dx} \quad (2.8)$$

ตามทฤษฎีค่า COG หรือ output ของ Fuzzy System คำนวนจากความต่อเนื่องของจุดบน Membership function ของ Aggregated Rule Consequent แต่ในทางปฏิบัตินั้นสามารถคำนวณจากตัวอย่างจุดตามแกนราบได้ ซึ่งในกรณีจะดำเนินการโดยใช้สมการที่ 2.3.4 และแสดงผลลัพธ์การดำเนินการในรูปที่ 2.7

$$COG = \frac{\sum_{x=a}^b \mu_A(x) x}{\sum_{x=a}^b \mu_A(x)} \quad (2.9)$$

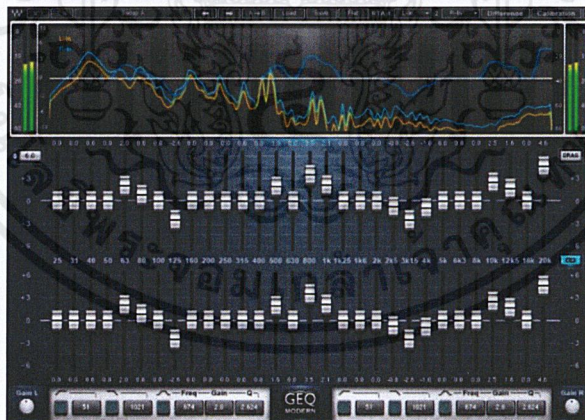


รูปที่ 2.9 รูปแสดงกระบวนการ Defuzzification

ผลลัพธ์ Crisp Output z_1 ได้ค่าเป็น 67.4 นั้นหมายถึงค่าความเสี่ยงที่คาดคะเนจากการใช้ Fuzzy Logic อยู่ที่ 67.4%

2.5 Graphic Equalizer

Graphic Equalizer เป็น Equalizer ประเภทหนึ่งที่สามารถปรับค่า Parameter ได้เพียงอย่างเดียว คือ Gainของแต่ละช่วงความถี่เท่านั้น ส่วน Parameters อื่นๆจะถูกกำหนดไว้โดยผู้ออกแบบ ซึ่งรวมไปถึงจำนวนช่วงความถี่ที่สามารถปรับค่าด้วยเช่นกัน ทำให้มีความเรียบง่ายในการใช้งาน เราจึงนำมาทดลองออกแบบ Graphic Equalizer ที่สามารถประมวลผลสัญญาณเสียงได้ในรูปแบบของการประมวลผลแบบ Real Time บนฮาร์ดแวร์



รูปที่ 2.10 ตัวอย่าง Graphic Equalizer Plug-in

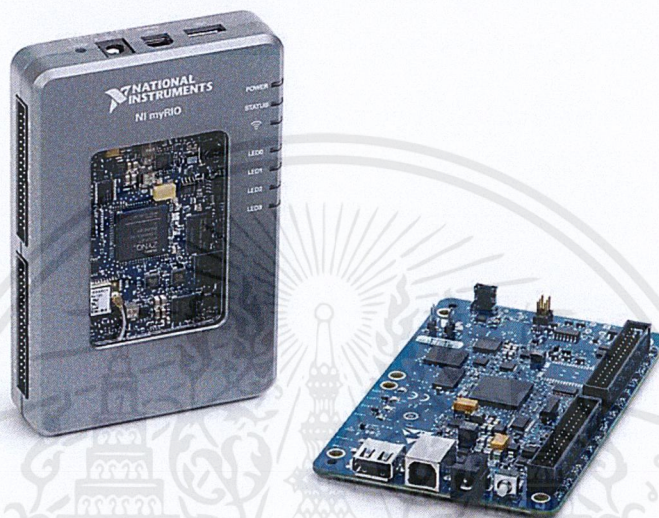
ที่มา : <https://bit.ly/2Or9HGJ>

Graphic Equalizer ที่ออกแบบ จะมีทั้งหมด 7 Bands แต่ละ Band จะมี Quality Factor เท่ากับ 6 และมีค่า Center Frequency ดังนี้ 125, 250, 500, 1000, 2000, 4000 และ 8000 Hz โดยจะออกแบบบนอุปกรณ์ FPGA ชื่อ NI myRIO ซึ่งสามารถออกแบบโปรแกรมได้ โดยใช้ซอฟต์แวร์ NI LabVIEW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 อุปกรณ์ NI myRIO

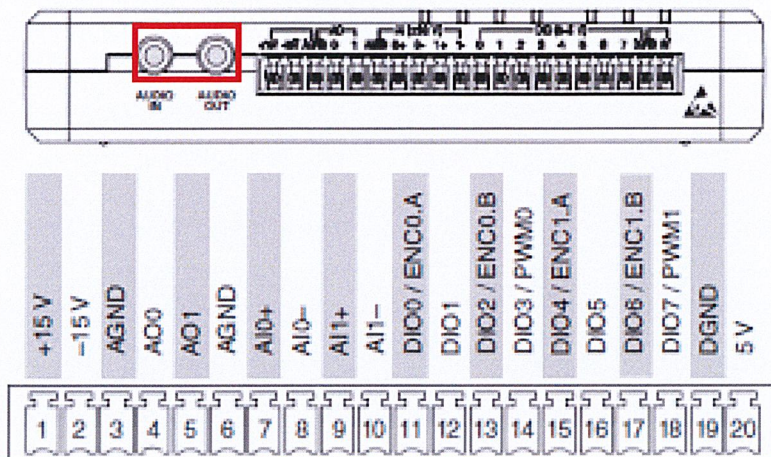
อุปกรณ์ NI myRIO คืออุปกรณ์ประมวลผลขนาดเล็กที่ใช้สถาปัตยกรรม RIO (Reconfigurable I/O) ควบคู่กับ FPGA (Field – Programmable Gate Array) ซึ่งสามารถเขียนโปรแกรมได้ด้วยซอฟต์แวร์ NI LabVIEW และสามารถปรับเปลี่ยนอินพุตและ output ได้ โดยภายในประกอบด้วย Dual – Core ARM® Cortex-A9 Real – Time Processor และ Xilinx FPGA ที่สามารถเชื่อมต่อกับอุปกรณ์ภายนอกต่างๆและเซนเซอร์ภายนอกได้



รูปที่ 2.11 อุปกรณ์ NI myRIO

ที่มา : <https://bit.ly/2U4sqhl>

อุปกรณ์ NI myRIO ประกอบด้วยฟังก์ชันสำคัญ เช่น WiFi, Accelerometer, LEDs, USB, และ Audio Input/Output เป็นต้น เราสามารถเขียนโปรแกรมแบบ Graphical Programming บนซอฟต์แวร์ LabVIEW เพื่อควบคุมการทำงานของ NI myRIO โดยเรามีการใช้พอร์ต Audio In และ Audio Out ดังในรูป 2.5.2 เป็นอินพุตและ output ของฮาร์ดแวร์ Graphic Equalizer ซึ่งพอร์ตทั้งสองสามารถรับและส่งสัญญาณที่เป็น Analog ได้ โดยมี Digital to Analog converter และ Analog to Digital converter ภายในอุปกรณ์ทำหน้าที่แปลงสัญญาณ Analog เป็น Digital ไปใช้กับ Equalizer แล้วแปลงสัญญาณ output ที่เป็นสัญญาณ Digital กลับเป็นสัญญาณ Analog เพื่อให้สามารถเชื่อมต่อกับลำโพงหรือหูฟังที่รับสัญญาณเป็น Analog ได้



รูปที่ 2.12 ตำแหน่งพอร์ต Audio IN / Audio OUT ของ NI myRIO

ที่มา : <https://bit.ly/2EklaDV>

2.6.1 อุปกรณ์ FPGA (Field - Programmable Gate Array)

อุปกรณ์ FPGA เป็นอุปกรณ์ที่ถูกพัฒนาเพื่อให้มีประสิทธิภาพและมีปริมาณความหนาแน่นของ Logic Gate สูง โดยทั่วไปมีความจุตั้งแต่ 1,000 Gates จนถึง 1,000,000 Gates และสามารถกำหนดฟังก์ชันได้ตามความต้องการของผู้ออกแบบ

โครงสร้างภายในของ FPGA สามารถโปรแกรมให้มีหน้าที่การทำงานเหมือน Logic Gate พื้นฐาน เช่น OR gate, AND gate, NOT Gate และ Flip-Flop เป็นต้น เป็นวงจรรวมที่มีขนาดใหญ่ Combinational Logic และมีความซับซ้อน

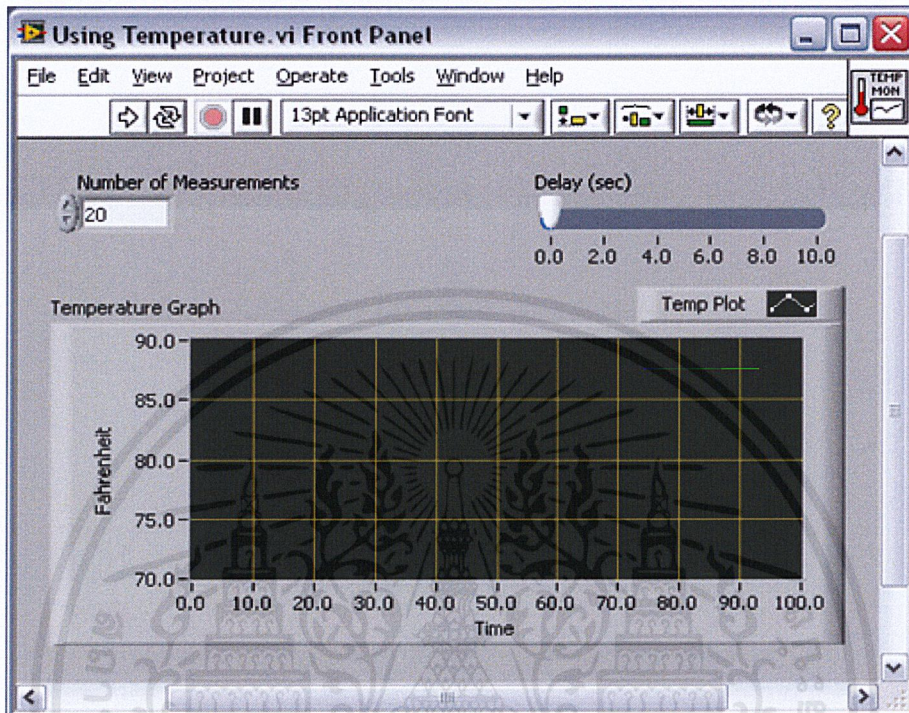
2.6.2 Laboratory Virtual Instrument Engineer Workbench (LabVIEW)

LabVIEW เป็นโปรแกรมคอมพิวเตอร์ที่สร้างเพื่อนำมาใช้ในด้านการจัดการวัดและเครื่องมือวัดสำหรับงานทางวิศวกรรม ซึ่งเป็นโปรแกรมที่เขียนด้วยวิธีการ Graphical Programming ทำให้ผู้พัฒนาโปรแกรมไม่จำเป็นต้องเขียน code หรือคำสั่งใดๆ ซึ่งภาษาที่ใช้ในโปรแกรมคือภาษา G (Graphical Language) หรือภาษารูปภาพที่มาแทนการเขียนโปรแกรมเป็นภาษาพื้นฐาน เช่น C, BASIC หรือ FORTRAN ด้วยรูปภาพหรือสัญลักษณ์ทั้งหมด ช่วยอำนวยความสะดวกและสามารถลดเวลาในการเขียนโปรแกรมลงไปได้มาก โดยเฉพาะในงานเขียนโปรแกรมคอมพิวเตอร์ เพื่อเชื่อมต่อกับอุปกรณ์ภายนอกเช่น Port หรือ Card ต่างๆรวมถึงการจัดวางตำแหน่งในหน่วยความจำเพื่อที่จะสามารถรวบรวมข้อมูลมาใช้ในการคำนวณและเก็บข้อมูลให้ได้ประโยชน์สูงสุด

โปรแกรมที่พัฒนาขึ้นโดย LabVIEW จะเรียกว่า Virtual Instrument (VI) มีส่วนประกอบ 3 ส่วน ดังนี้

2.6.2.1 Front panel

Front Panel เป็นส่วน User Interface ที่ใช้สื่อสารระหว่างผู้ใช้กับโปรแกรม มีลักษณะเหมือนหน้าต่างของเครื่องมือหรืออุปกรณ์ที่ใช้งานการวัดทั่วไป จึงเปรียบเสมือน Graphical User Interface ของโปรแกรมหรือ VI มีลักษณะดังรูป 2.5.3



รูปที่ 2.13 ตัวอย่าง Front panel ของ LabVIEW

ที่มา : <https://bit.ly/2Jsr3Dm>

Front Panel มีส่วนประกอบสำคัญ 2 ส่วนคือ

2.6.2.1.1 Control

Control มีหน้าที่เป็นตัวควบคุมการรับค่าอินพุตจากผู้ใช้ในการไปกำหนดค่าหรือแหล่งข้อมูลต่างๆ

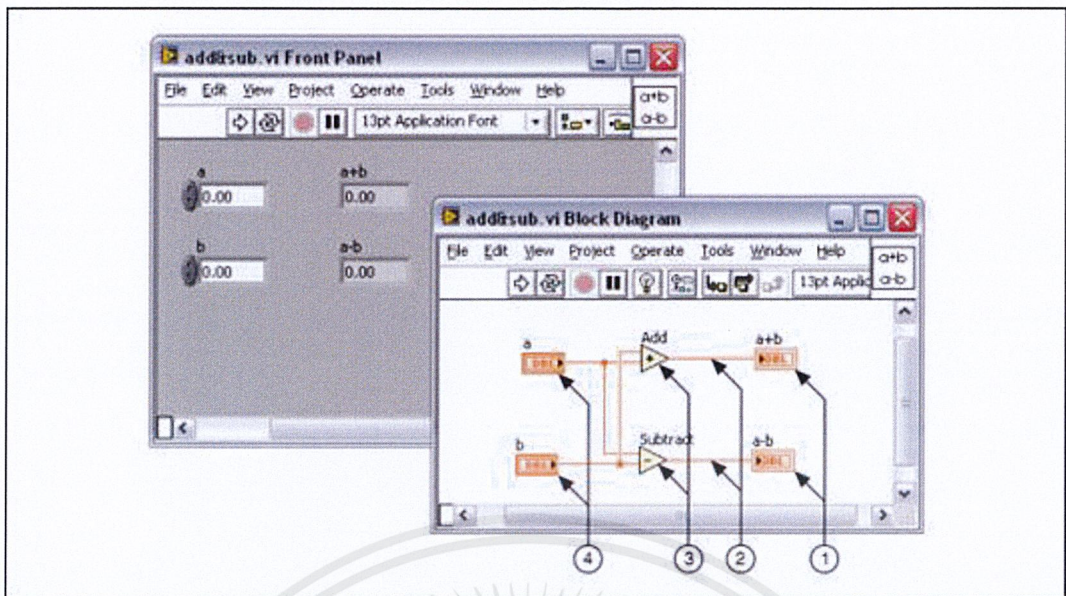
2.6.2.1.2 Indication

Indication มีหน้าที่เป็นตัวแสดงผล โดยการรับค่าที่ได้จากแหล่งข้อมูลมาแสดงผลในรูปแบบต่างๆ เช่น กราฟ, ตัวเลข หรือ เข็ม เป็นต้น เปรียบเสมือนเป็น output ของผู้ใช้งาน ซึ่งผู้ใช้งาน ไม่สามารถปรับค่าต่าง ๆ ที่แสดงผลบน Indication ได้โดยตรง

2.6.2.2 Block Diagram

Block Diagram เป็นส่วนที่เสมือนกับ Source Code ที่พัฒนาขึ้นโดย LabVIEW ซึ่งอยู่ในรูปภาษา G โดย LabVIEW มีการตรวจสอบข้อผิดพลาดของโปรแกรมตลอดเวลา เมื่อเกิดข้อผิดพลาด โปรแกรมจะไม่สามารถทำงานได้โดยผู้ใช้งานสามารถดูรายละเอียดข้อผิดพลาดเพื่อให้เกิดความสะดวกต่อการเขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 ตัวอย่าง Block Diagram ของ LabVIEW

ที่มา : <https://bit.ly/2JQv8AD>

สำหรับ Block Diagram จะมีส่วนที่ทำหน้าที่ส่งผ่านข้อมูล (Data Flow) และกำหนดวิธีการประมวลผลข้อมูลอยู่ 4 ส่วนดังนี้

2.6.2.2.1 Terminal

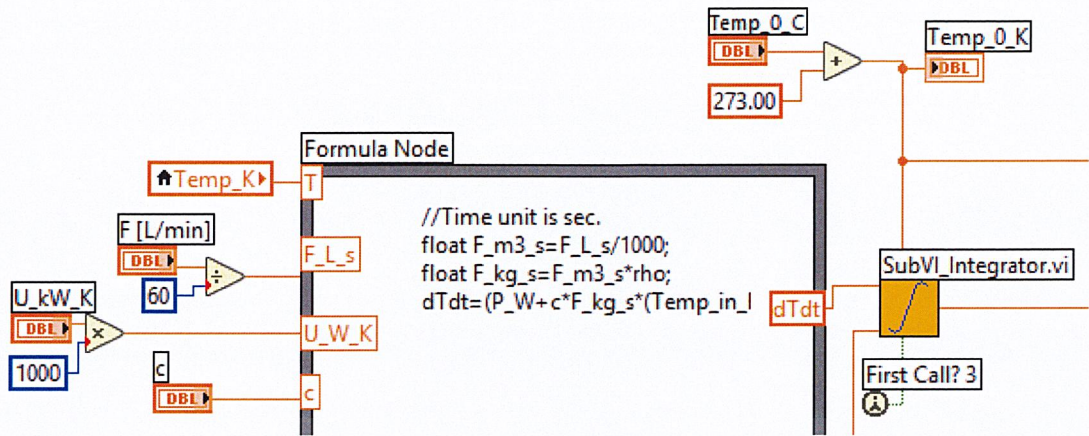
Terminal เป็นสถานีของข้อมูล ถ้าเป็นสถานีต้นทางของข้อมูลจะเป็น Control Terminal (เลข (4) ใน 2.5.4) และถ้าเป็นสถานีปลายทางของข้อมูลจะเป็น Indicator Terminal (เลข (1) ในรูป 2.5.4)

2.6.2.2.2 Node

Node เป็นจุดต่อใน Block Diagram แสดงเป็นอินพุตหรือ output จะทำงานเมื่อสั่ง RUN VI แสดงใน เลข (3) ในรูป 2.5.4

2.6.2.2.3 Function

Function เป็นตัวดำเนินการสำเร็จรูป เช่น Sine, Cos, Tan และ Log เป็นต้น นอกจากนี้ยังสามารถใช้สร้างบล็อกเพื่อสร้างสมการขึ้นเองได้ด้วย



รูปที่ 2.15 ตัวอย่างการใช้ Function สร้าง Block สมการ

ที่มา : <https://bit.ly/2QaBwE5>

2.6.2.2.4 Wire

Wire เป็นการเชื่อมต่อการรับส่งข้อมูลระหว่าง Terminal หรือ Node ต่างๆ ที่มีใน Block Diagram เข้าด้วยกันโดย Wire จะทำหน้าที่กำหนดเส้นทางของข้อมูลว่ามีลำดับเป็นอย่างไรและสุดท้ายจะมีการแสดงผลที่ Terminal ไต แสดงใน เลข (2) ในรูป 2.5.4

2.6.2.3 Icon และ Connector

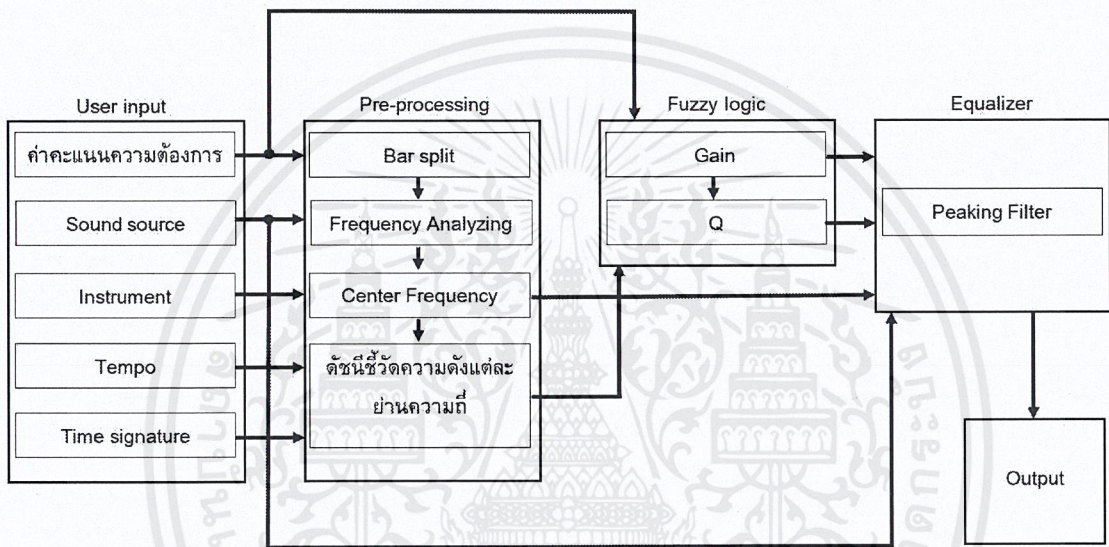
เสมือนเป็นโปรแกรมย่อย (Procedure) ประเภท Subroutine ในโปรแกรมทั่วไป ซึ่ง Icon จะหมายถึง Block Diagram ที่มีการย่อส่วนข้อมูลเข้าและออกผ่านทาง Connector ซึ่งใน LabVIEW เรียก Subroutine นี้ว่า subVI

บทที่ 3

วิธีการดำเนินงาน

วิธีการดำเนินงาน แบ่งออกเป็น 2 ส่วนหลักๆ คือ ในส่วนของ การออกแบบ Digital Parametric Equalizer ที่มีปัญญาประดิษฐ์ที่ออกแบบโดยใช้ Fuzzy Logic Theorem มาช่วยในการปรับค่า Parameters และการออกแบบ Digital Graphic Equalizer ที่สามารถประมวลผลสัญญาณเสียงได้ในรูปแบบ Real Time Processing

3.1 Digital Parametric Equalizer



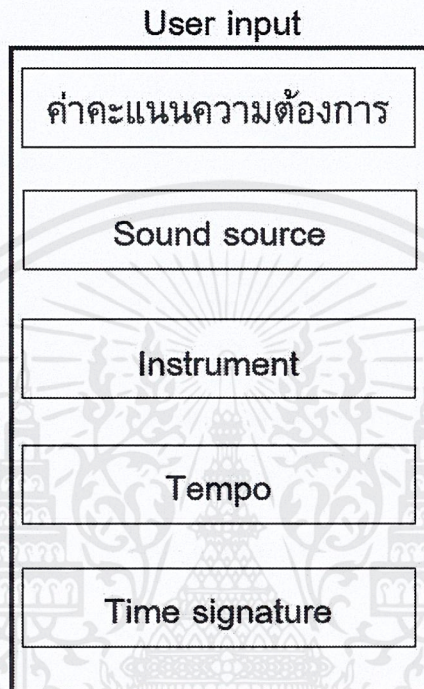
รูปที่ 3.1 รูปแสดงBlock Diagram ของ Digital Parametric Equalizer

Parametric Equalizer มี Parameter ที่ทั้งหมดที่สามารถปรับค่าได้ประกอบไปด้วย Center Frequency, Quality Factor และ Gain ซึ่ง Parameters ทั้ง 3 ตัวนี้มีหน้าที่ที่แตกต่างกันดังที่ได้กล่าวถึงในหัวข้อที่ 2.2 โดย Parameters ทั้ง 3 นี้จะสร้างความสับสนให้กับผู้ใช้งานที่ยังขาดความชำนาญ เราจึงได้มีการออกแบบวิธีการหาค่า Parameters ทั้ง 3 และนำปัญญาประดิษฐ์เข้ามาช่วยประยุกต์ใช้เพื่อเป็นตัวช่วยในการปรับค่า Parameters นี้

Digital Filter ที่ใช้ในการออกแบบ Parametric Equalizer จะใช้เป็น Peaking Filter ซึ่ง Equalizer ที่ทำการออกแบบนี้จะมีทั้งหมด 3 Band ประกอบด้วย Low Frequency Band, Mid Frequency Band และ High Frequency Band กล่าวคือจะใช้ Peaking Filter ทั้งหมด 3 ตัว นำ Digital Filter ทั้งหมดมาเชื่อมต่อกันแบบ Cascade ให้ผลลัพธ์สุดท้ายเป็น Digital Filter รวมสำหรับ Equalizer

ในแต่ละ Frequency Band จะมีการกำหนดขอบเขตค่าความถี่สูงสุดและต่ำสุดเอาไว้ตามคุณลักษณะทางความถี่ของสัญญาณเสียงที่ผู้ใช้เป็นผู้เลือกเข้าสู่ระบบ และค่า Parameters ต่างๆจะ ได้จากการประมวลผลสัญญาณอินพุตรวมกับการใช้ปัญญาประดิษฐ์ ซึ่งค่า Center Frequency จะหาได้ในขั้นตอน Pre-Processing ส่วนค่า Quality Factor และค่า Gain จะใช้ปัญญาประดิษฐ์ในการหาค่า

3.1.1 การรับอินพุตจากผู้ใช้งาน



รูปที่ 3.2 รูปแสดง Block Diagram ส่วน User Input

ผู้ใช้งานจะต้องป้อนอินพุตเข้าสู่ระบบดังนี้

3.1.1.1 ค่าคะแนนความต้องการ

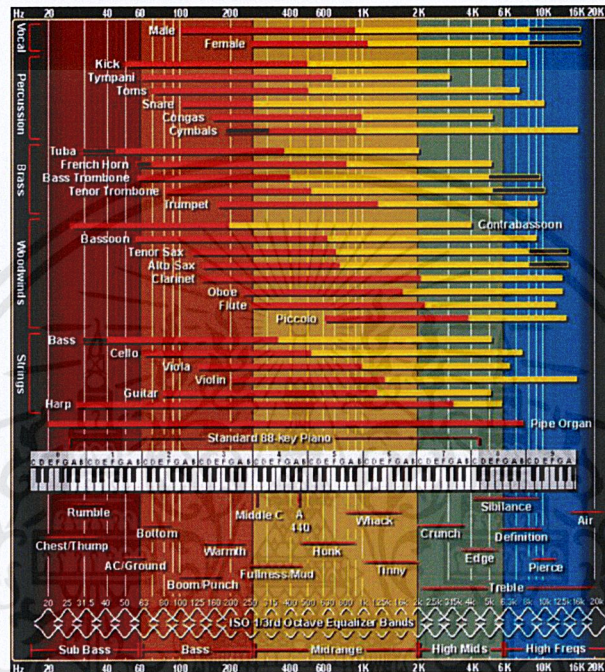
ในส่วนนี้ผู้ใช้งานจะต้องฟังเสียงที่ต้องการทำการปรับความถี่ และให้คะแนนในแต่ละย่านลงไป โดยถ้าให้คะแนนที่ติดลบจะหมายความว่าผู้ใช้ต้องการจะลดความดังของความถี่ย่านนั้นๆ แต่หากให้คะแนนที่เป็นบวกจะหมายความว่าผู้ใช้งานต้องการที่จะเพิ่มความดังของย่านความถี่นั้นๆ

3.1.1.2 Sound Source

ผู้ใช้งานต้องเลือกไฟล์เสียงที่ต้องการทำการปรับค่าความถี่ โดยเสียงที่นำมาจะต้องเป็นสัญญาณเสียงของเครื่องดนตรีเพียงชิ้นเดียวเท่านั้น เพื่อให้การปรับ Parameters มีประสิทธิภาพสูงสุด เพราะโดยทั่วไปในการผสมเสียงนั้นการปรับ Equalizer จะทำทีละ Track หรือทีละชิ้นเครื่องดนตรีเป็นส่วนมาก เราจึงนำแนวคิดนี้มาใช้ใน Digital Parametric Equalizer ของเรา

3.1.1.3 ชนิดของเครื่องดนตรี (Instrument)

เนื่องจากเครื่องดนตรีแต่ละชนิดนั้น จะมีคุณลักษณะทางความถี่ที่แตกต่างกันไปตามชนิดของเครื่องดนตรี ซึ่งส่งผลอย่างมากในการหาค่า Center Frequency ของ Frequency Band ต่างๆ จึงต้องให้ผู้ใช้ต้องเลือกชนิดเครื่องดนตรีโดยอ้างอิงจาก Sound Source ที่ผู้ใช้งานต้องการจะปรับความถี่ เพื่อให้สามารถแบ่งความถี่ให้อยู่ใน Frequency Range อันประกอบด้วย Low Frequency Range, Mid Frequency Range และ High Frequency Range ได้



รูปที่ 3.3 รูปแสดงตัวอย่างการแบ่งย่านความถี่ของเครื่องดนตรีต่างๆ

ที่มา : <https://bit.ly/2w9kahl>

3.1.1.4 Tempo

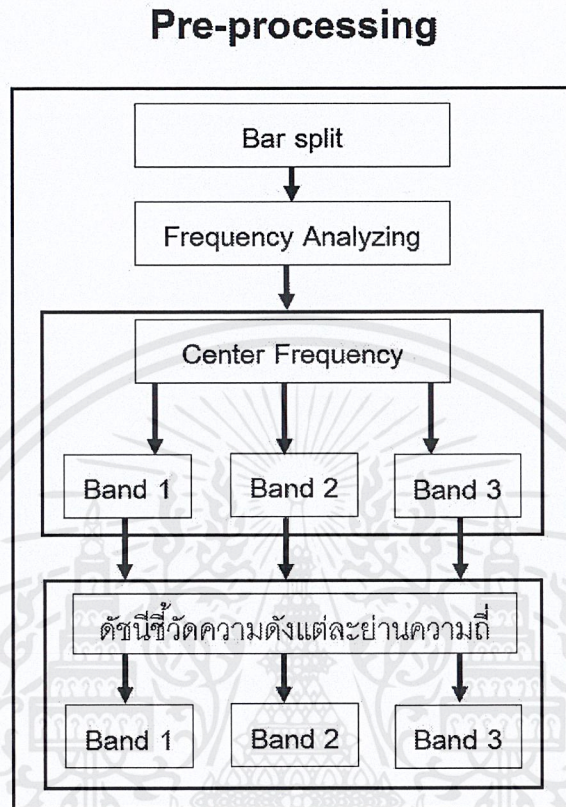
Tempo คือความเร็วในการเล่นโน้ตดนตรีซึ่งระบุเป็นค่า BPM (Beats per Minute) หมายถึงจะมีจังหวะทั้งหมดกี่จังหวะใน 1 นาที ในงานประพันธ์ดนตรีส่วนมากมักใช้โน้ตตัวดำเป็นโน้ตอ้างอิงจังหวะ เช่น โน้ตตัวดำ = 120 BPM หมายความว่าใน 1 นาทีจะมีการเล่นโน้ตตัวดำทั้งหมด 120 ตัว เป็นต้น และจะมีการกำหนด Tempo ไว้อย่างชัดเจนในทุกชิ้นงาน ซึ่งค่า Tempo เป็นอินพุตสำคัญที่ใช้ในส่วน Pre-processing ในขั้นตอนการแบ่งส่วนสัญญาณเพื่อใช้ในการประมวลผล

3.1.1.5 Time Signature

Time Signature คือเครื่องหมายกำหนดจังหวะ ที่ถูกกำหนดไว้เพื่อบอกว่าใน 1 ห้องเพลงมีกี่จังหวะ และใช้โน้ตใดเป็นเกณฑ์ 1 จังหวะ เช่น 4/4 มีความหมายว่าใน 1 ห้องเพลงมี 4 จังหวะและกำหนดให้โน้ตตัวดำมีค่าเท่ากับ 1 จังหวะ เป็นต้น ซึ่งในงานประพันธ์ดนตรีจะมีการกำหนด

ไว้อย่างชัดเจนในทุกชิ้นงาน ซึ่งค่า Time Signature เป็นอินพุตสำคัญที่ใช้ในส่วน Pre-processing ในขั้นตอนการแบ่งส่วนสัญญาณเพื่อใช้ในการประมวลผล

3.1.2 Pre-processing



รูปที่ 3.4 รูปแสดง Block Diagram ของส่วนการทำ Pre-processing

ที่มา : <https://bit.ly/2w9kahl>

นำอินพุตจากผู้ใช้งานและสัญญาณอินพุตมาแปลงเป็นข้อมูลที่จำเป็นสำหรับปัญหาประติษฐ์ มีขั้นตอนดังนี้

3.1.2.1 การแบ่งสัญญาณเป็นช่วง (Bar Split)

แบ่งสัญญาณอินพุตเป็นชุดขนาดซึ่งแต่ละชุดมีขนาดเท่ากับ 1 ห้องดนตรี โดยใช้ Parameters, Tempo และ Time Signature ที่ผู้ใช้งานป้อนในส่วนอินพุตจากผู้ใช้งาน ซึ่งขั้นตอนการแบ่งห้องเริ่มจากการหาความยาวของ 1 ห้องดนตรีในหน่วยเวลาเป็นวินาทีจากสมการ

$$\text{Barlength} = \frac{60}{(\text{Tempo} \times \text{Time Signature})} \quad (3.1)$$

เมื่อ Tempo อยู่ในหน่วย Beat per second (BPS)

จากนั้นนำผลลัพธ์ที่ได้มาคำนวณหา Sample Length ใน 1 ห้อง โดยนำค่า Sampling Frequency มาคำนวณในสมการ

$$\text{Samplelength} = \text{Barlength} \times F_s \quad (3.2)$$

เมื่อใด Sample Length ใน 1 ห้องแล้ว จึงนำข้อมูลมาจัดกลุ่มเป็นห้องตามลำดับจนครบทุก Sample

$$\text{ห้องแรก } \text{bar}\{i\} = x(1 : (F_s \times \text{Barlength}) - 1)$$

$$\text{ห้องต่อไป } \text{bar}\{i\} = x(F_s \times (\text{Barlength} \times j) : (F_s \times (\text{Barlength} \times (j + 1)) - 1))$$

$$\text{ห้องสุดท้าย } \text{bar}\{i\} = x(F_s \times (\text{Barlength} \times j) : \text{length}(x))$$

$$\text{โดยที่ } i = 1, 2, 3, \dots, \frac{\text{length}(x)}{F_s \times \text{Barlength}} - 1 \text{ และ } j = 1, 2, 3, \dots, \frac{\text{length}(x)}{F_s \times \text{Barlength}} - 1$$

3.1.2.2 Frequency Analyzing

นำชุดข้อมูลสัญญาณอินพุตที่ถูกแบ่งเป็นห้องแล้วมาวิเคราะห์คุณสมบัติทางความถี่ด้วย Discrete Fourier Transform

$$X(k) = \sum_{k=0}^{N-1} x(k) e^{\frac{-j2\pi mk}{N}} \quad (3.3)$$

โดยที่ $x(k)$ คือ เฟรมของสัญญาณเสียงที่ผ่านแฮมมิงวินโดว์แล้ว

$X[k]$ คือ Spectrum ของเฟรมสัญญาณเสียง $x(k)$

N คือ ขนาดของเฟรมสัญญาณเสียง $x(k)$

m คือ ตำแหน่งของสัญญาณเสียงอินพุตซึ่งมีค่าตั้งแต่ $m = 0$ ถึง $N - 1$

Discrete Fourier Transform นั้นเป็นวิธีการแปลงลำดับสัญญาณทางเวลาที่มีจำนวน N ข้อมูล ให้ได้ค่าผลลัพธ์เป็น Frequency Spectrum ที่มีจำนวนจุดข้อมูลเท่ากับ N ตัว แต่ในการคำนวณ Discrete Fourier Transform แต่ละครั้งจะต้องมีการคูณตัวเลขทั้งสิ้น $4N^2$ ครั้งและมีการบวกตัวเลขอีก $N(4N-2)$ ครั้ง ดังนั้นจะเห็นว่าการแปลง Discrete Fourier Transform แต่ละครั้งจะต้องใช้เวลาในการคำนวณเพิ่มขึ้นเป็นสัดส่วนโดยตรงกับ N^2 ซึ่งมี N จำนวนมากการคำนวณในแต่ละครั้งจะต้องใช้เวลานานขึ้นตามไปด้วย ดังนั้นในทางปฏิบัติจึงใช้ Fast Fourier Transform แทนซึ่งจะทำให้การทำ Fourier Transform นั้นทำได้อย่างมีประสิทธิภาพมากขึ้นและรวดเร็วขึ้น

จากนั้นนำชุดข้อมูลทั้งหมดของแต่ละห้องมาเฉลี่ยเพื่อวิเคราะห์ Amplitude ของแต่ละความถี่ตลอดทั้งสัญญาณโดยอ้างอิงจากสมการ

$$\text{average}(n) = \frac{P_1(1)+P_2(2)+P_3(3)+\dots+P_{n-1}(\text{length}(P))}{\text{length}(P)-1} \quad (3.4)$$

เมื่อ $n = 1, 2, 3, \dots$, จำนวนห้องที่แบ่งได้

เมื่อหาค่าเฉลี่ยได้เป็นที่เรียบร้อย จะนำค่าทั้งหมดไปทำการ Normalize ให้ค่าสูงสุด มีค่าเป็น 1 จากสมการ

$$average_nor(n) = \frac{average(n)}{\max(average)} \quad (3.5)$$

เมื่อ $n = 1, 2, 3, \dots, length(average)$

3.1.2.3 การหาค่า Center Frequency ของแต่ละย่านความถี่

เครื่องดนตรีแต่ละชนิดจะมี Frequency Range ที่แตกต่างกัน เช่น เสียงกีตาร์มี Low Frequency Range อยู่ในช่วงความถี่ 80-250 Hz หรือเสียงเปียโนมี Mid Frequency Range อยู่ในช่วงความถี่ 251-2000 Hz เป็นต้น ในขั้นตอนการวิเคราะห์คุณสมบัติทางความถี่เราสามารถระบุความถี่ที่มี Amplitude สูงที่สุดได้ ซึ่งความถี่ที่ถูกระบุนั้นจะเป็น Center Frequency ของ Digital Filter ตัวแรก และการระบุว่าเป็น Digital Filter ของย่านความถี่ใดจะพิจารณาอ้างอิงจากสัญญาณ อินพุตว่าเป็นสัญญาณอะไรด้วยคามแตกต่างของคุณลักษณะทางความถี่ที่ตั้งที่อธิบายไปข้างต้น และในการหาค่า Center Frequency ของย่านความถี่อีก 2 ย่านที่เหลือจะต้องพิจารณาเป็นกรณีได้ 3 กรณี ดังนี้

กรณีที่ 1 ค่า Center frequency ของ Digital Filter ตัวแรกอยู่ใน Mid Frequency Range ของสัญญาณเสียง Digital Filter ตัวนี้จะเป็น Mid Frequency Band ของ Parametric Equalizer ไปโดยปริยาย ส่วน Center Frequency ของ Low Frequency Band และ High Frequency Band จะคำนวณจากวิธีการลด Octave และการเพิ่ม Octave ตามลำดับ การเพิ่ม Octave คือการนำค่า Center Frequency ของ Digital Filter ตัวแรกมาคูณ 2 ขึ้นไปเรื่อยๆ จนได้ค่าที่สูงที่สุดที่ไม่เกิน High Frequency Range ของสัญญาณเสียง และการทำ Down Octave คือการนำค่า Center Frequency ของ Digital Filter ตัวแรกมาหาร 2 ลงไปจนได้ค่าที่ต่ำที่สุดที่ไม่เกิน Low Frequency Range ของสัญญาณเสียง

กรณีที่ 2 ค่า Center Frequency ของ Digital Filter ตัวแรกอยู่ใน Low Frequency Range ของสัญญาณเสียง Digital Filter ตัวนี้จะเป็น Low Frequency Band ของ Parametric Equalizer ไปโดยปริยาย ส่วน Center Frequency ของ Low Frequency Band และ High Frequency Band จะคำนวณจากการเพิ่ม Octave ซึ่งเป็นการนำค่า Center Frequency ของ Digital Filter ตัวแรกมาคูณ 2 ขึ้นไปเรื่อยๆ โดยค่า Center Frequency ของ Mid Frequency Band จะเท่ากับค่าความถี่สูงสุดที่ไม่เกิน Mid Frequency Range ของสัญญาณเสียง และค่า Center Frequency ของ High Frequency Band จะเท่ากับค่าความถี่สูงสุดที่ไม่เกิน High Frequency Range ของสัญญาณเสียง

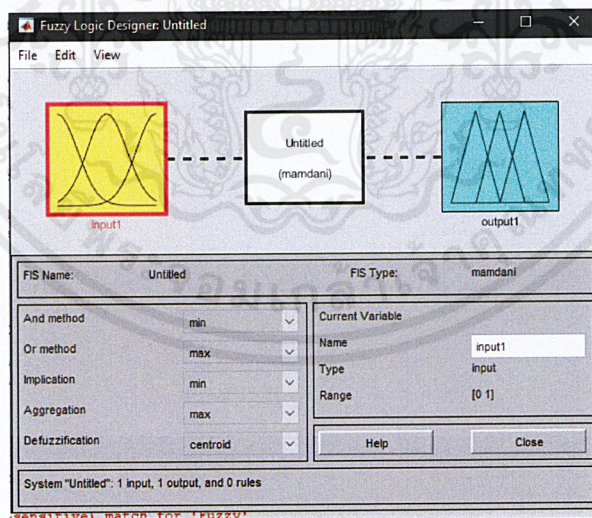
กรณีที่ 3 ค่า Center Frequency ของ Digital Filter ตัวแรกอยู่ใน High Frequency Range ของสัญญาณเสียง Digital Filter ตัวนี้จะเป็น High Frequency Band ของ Parametric Equalizer ไปโดยปริยาย ส่วน Center Frequency ของ Low Frequency Band และ Mid Frequency Band จะคำนวณจากการลด Octave ซึ่งเป็นการนำค่า Center Frequency ของ Digital Filter ตัวแรกมาหาร 2 ลงไปเรื่อยๆ โดยค่า Center Frequency ของ Mid Frequency Band จะเท่ากับค่าความถี่ต่ำสุดที่ไม่เกิน Mid Frequency Range ของสัญญาณเสียง และค่า Center Frequency ของ Low Frequency Band จะเท่ากับค่าความถี่ต่ำสุดที่ไม่เกิน Low Frequency Range ของสัญญาณเสียง

3.1.2.4 การหาดัชนีชี้วัดความดังของแต่ละย่านความถี่

ทำการแบ่งข้อมูลที่ได้จากขั้นตอนที่ 3.1.2.2 ที่เป็นชุดข้อมูล Frequency Range ตามชนิดเครื่องดนตรีที่ผู้ใช้งานกำหนด จากนั้นนำข้อมูลของแต่ละความถี่ในแต่ละย่านมาหาค่าเฉลี่ย ผลลัพธ์ที่ได้จะเป็นค่าดัชนีชี้วัดความดังของแต่ละย่านความถี่ของสัญญาณอินพุต ซึ่งถูกนำไปใช้ใน ส่วน Processing ของปัญญาประดิษฐ์

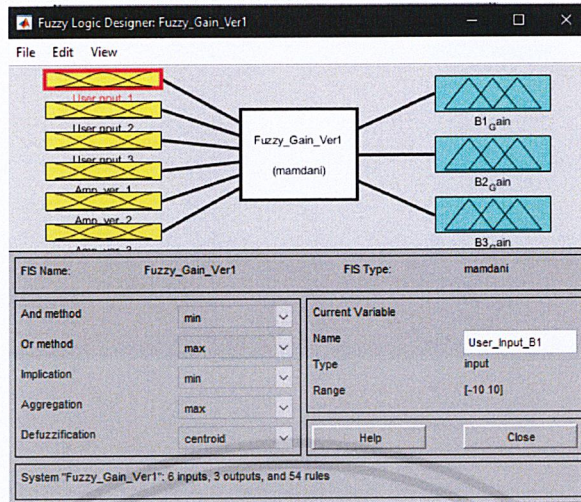
3.1.3 Fuzzy logic AI

ปัญญาประดิษฐ์ที่นำมาใช้ในการหาค่า Parameters, Quality Factor และ Gain คือ Fuzzy Logic System รูปแบบ Mamdani ซึ่งใช้ FIS Tools ของโปรแกรม MATLAB ในการ ออกแบบ



รูปที่ 3.5 รูปแสดงหน้าต่าง FIS Tool ของโปรแกรม

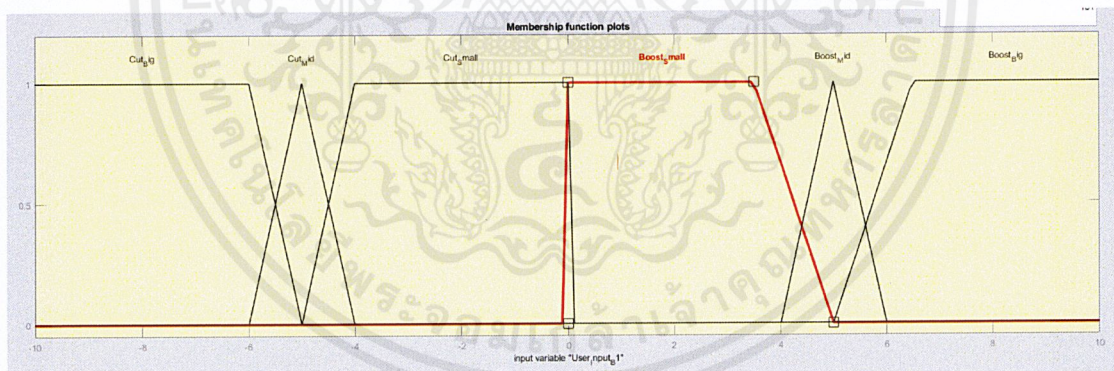
3.1.3.1 การหาค่า Gain



รูปที่ 3.6 รูปแสดงหน้าต่าง FIS Tool ของโปรแกรม MATLAB ส่วนของการคำนวณหาค่า Gain

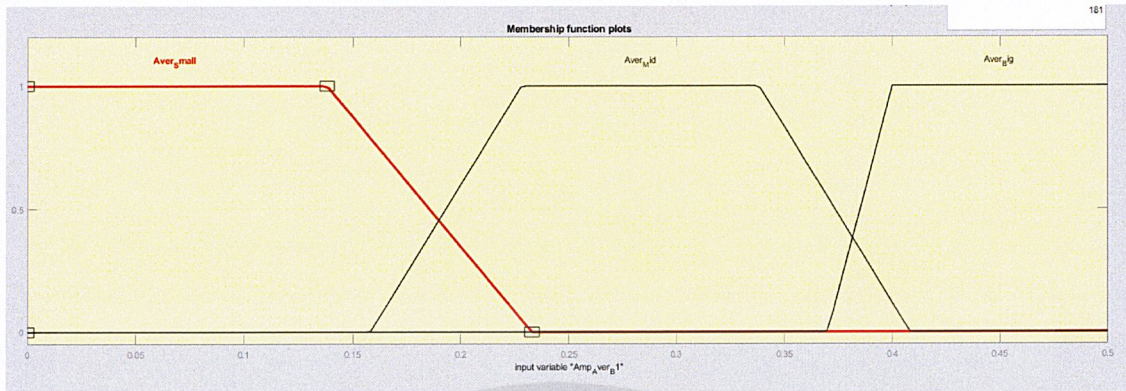
อินพุตที่ป้อนเข้าสู่ Fuzzy Logic System คือ ค่าคะแนนความต้องการที่ผู้ใช้งานเป็นผู้ป้อนค่าทั้ง 3 Frequency Band และค่าดัชนีความดังของแต่ละย่านความถี่ที่คำนวณได้จากในขั้นตอน Pre processing

Input Membership Function ของค่าคะแนนความต้องการ ใช้เป็นฟังก์ชันรูปแบบ Triangle Function และ Trapezoid Function ผสมกันตามที่แสดงในรูปที่ 3.7



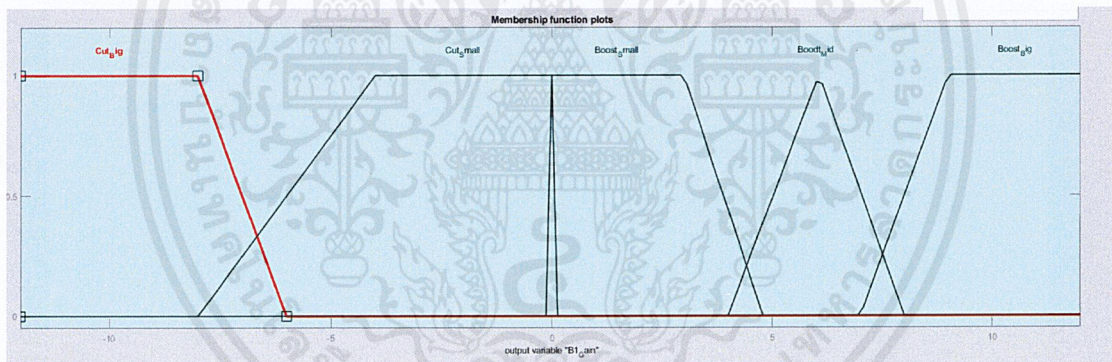
รูปที่ 3.7 รูปแสดง Input Membership Function ของค่าคะแนนความต้องการของ User

Input Membership Function ของค่าดัชนีความดังของย่านความถี่ ใช้เป็นฟังก์ชันรูปแบบ Trapezoid Function ตามที่แสดงในรูปที่ 3.8



รูปที่ 3.8 รูปแสดง Input Membership Function ของค่าคะแนนความต้องการของ User

Output Membership Function ของ Fuzzy Logic System ในการคำนวณค่า Gain ใช้เป็นฟังก์ชันรูปแบบ Triangle Function และ Trapezoid Function ผสมกันตามทีแสดงในรูปที่ 3.9 และขอบเขต Output ที่จะเป็นค่า Gain ของ Parametric Equalizer จะอยู่ในช่วง -12 ถึง +12 dB



รูปที่ 3.9 รูปแสดง Output Membership Function ของระบบ Fuzzy Logic ส่วนการคำนวณค่า

Rules ของ Fuzzy Logic System ส่วนการหาค่า Gain แสดงอยู่ในตารางที่ 3.1

ตารางที่ 3.1 ตารางแสดง Rules ที่ใช้ใน Fuzzy Inference System ของการหาค่า Gain

IF	Then
User_Input is Cut_Big AND Amplitude_Average is Small	Band1 Gain is Cut_Small Band2 Gain is Boost_Small Band3 Gain is Boost_Small
User_Input is Cut_Big AND Amplitude_Average is Medium	Band1 Gain is Cut_Big
User_Input is Cut_Big AND Amplitude_Average is Big	Band1 Gain is Cut_Big
User_Input is Cut_Medium AND Amplitude_Average is Small	Band1 Gain is Cut_Small Band2 Gain is Boost_Medium Band3 Gain is Boost_Medium
User_Input is Cut_Medium AND Amplitude_Average is Medium	Band1 Gain is Cut_Big
User_Input is Cut_Medium AND Amplitude_Average is Big	Band1 Gain is Cut_Big
User_Input is Cut_Small AND Amplitude_Average is Small	Band1 Gain is Cut_Small
User_Input is Cut_Small AND Amplitude_Average is Medium	Band1 Gain is Cut_Small
User_Input is Cut_Small AND Amplitude_Average is Big	Band1 Gain is Cut_Small
User_Input is Boost_Small AND Amplitude_Average is Small	Band1 Gain is Boost_Small

ตารางที่ 3.1 (ต่อ) ตารางแสดง Rules ที่ใช้ใน Fuzzy Inference System ของการหาค่า Gain

User_Input is Boost_Small AND Amplitude_Average is Medium	Band1 Gain is Boost_Small
User_Input is Boost_Small AND Amplitude_Average is Big	Band1 Gain is Boost_Small
User_Input is Boost_Medium AND Amplitude_Average is Small	Band1 Gain is Boost_Small
User_Input is Boost_Medium AND Amplitude_Average is Medium	Band1 Gain is Boost_Medium
User_Input is Boost_Medium AND Amplitude_Average is Big	Band1 Gain is Boost_Small Band2 Gain is Cut_Small Band3 Gain is Cut_Small
User_Input is Boost_Big AND Amplitude_Average is Small	Band1 Gain is Boost_Big
User_Input is Boost_Big AND Amplitude Average is Medium	Band1 Gain is Boost_Small Band2 Gain is Cut_Small Band3 Gain is Cut_Small
User_Input is Boost_Big AND Amplitude_Average is Big	Band1 Gain is None Band2 Gain is Cut_Big Band3 Gain is Cut_Big

โดย

User_Input หมายถึง ค่าคะแนนความต้องการ

Cut_Small หมายถึง ลดน้อย, Cut_Medium หมายถึง ลดปานกลาง, Cut_Big หมายถึง ลดมาก

Boost_Small หมายถึง เพิ่มน้อย, Boost_Medium หมายถึง เพิ่มปานกลาง, Boost_Big หมายถึง เพิ่มมาก

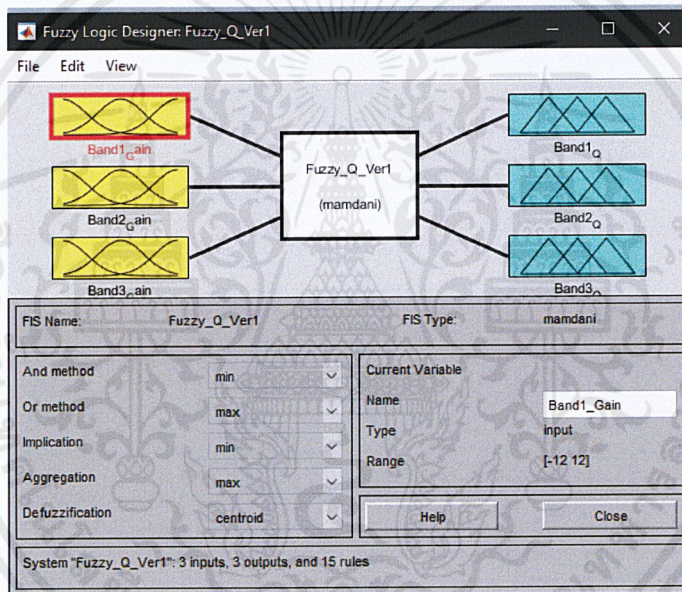
Amplitude_Average หมายถึง ค่าดัชนีชี้วัดความดังของย่านความถี่

Small หมายถึง น้อย, Medium หมายถึง ปานกลาง และ Big หมายถึง มาก



รูปที่ 3.10 รูปแสดงหน้าต่าง Rules ของส่วนการคำนวณ Gain บน FIS Tools

3.1.3.2 การหาค่า Quality Factor

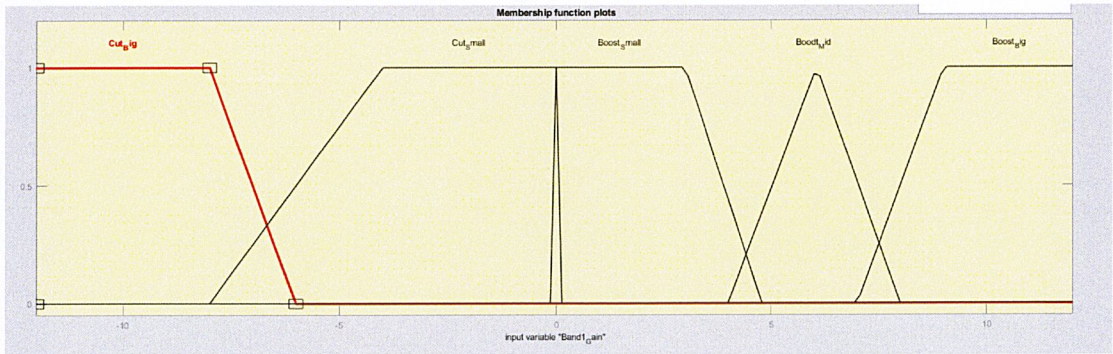


รูปที่ 3.11 รูปแสดงหน้าต่าง FIS Tool ของโปรแกรม MATLAB

ส่วนของการคำนวณหาค่า Quality Factor

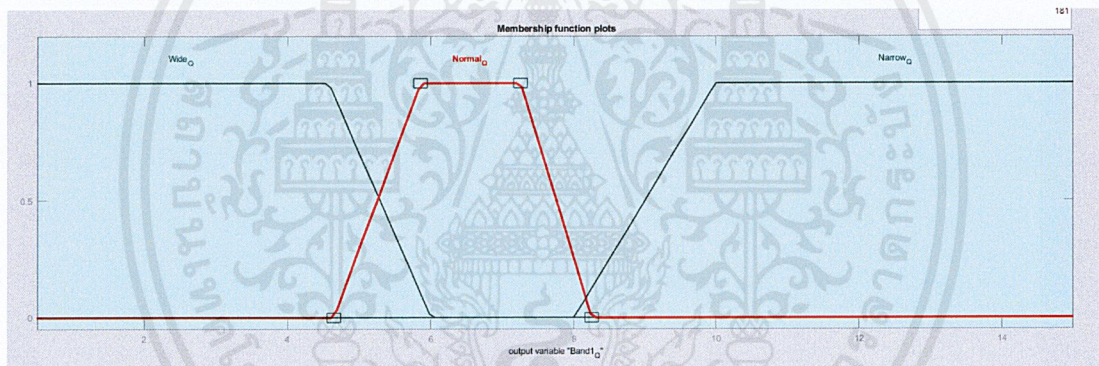
อินพุตที่ป้อนเข้า Fuzzy Logic System คือค่า Gain ที่หาค่าได้จาก Fuzzy Logic System ของค่า Gain

Input Membership Function ของค่า Gain ใช้เป็นฟังก์ชันรูปแบบ Triangle Function และ Trapezoid Function ผสมกันตามที่แสดงในรูปที่ 3.12



รูปที่ 3.12 รูปแสดง Input Membership Function ของค่า Gain

Output Membership Function ของ Fuzzy Logic System ในการคำนวณค่า Quality Factor ใช้เป็นฟังก์ชันรูปแบบ Trapezoid Function ตามที่แสดงในรูปที่ 3.13 และขอบเขต Output ของค่า Quality Factor ของ Parametric Equalizer จะอยู่ในช่วง 1 ถึง 15



รูปที่ 3.13 รูปแสดง Output Membership Function ของระบบ Fuzzy Logic ส่วนการคำนวณหาค่า Quality Factor

Rules ของ Fuzzy Logic System ส่วนการหาค่า Quality Factor แสดงอยู่ในตารางที่ 3.2

ตารางที่ 3.2 ตารางแสดง Rules ที่ใช้ในระบบ Fuzzy Logic ของการหาค่า Quality Factor

IF	Then
Gain is Cut_Big	Q is Narrow
Gain is Cut_Small	Q is Wide
Gain is Boost_Small	Q is Wide
Gain is Boost_Medium	Q is Normal
Gain is Boost_Big	Q is Wide

Cut_Small หมายถึง ลดน้อย, Cut_Medium หมายถึง ลดปานกลาง และ Cut_Big หมายถึง ลดมาก

Narrow หมายถึง แคบ, Normal หมายถึง ปกติ และ Wide หมายถึง กว้าง

3.1.4 Peaking Filter

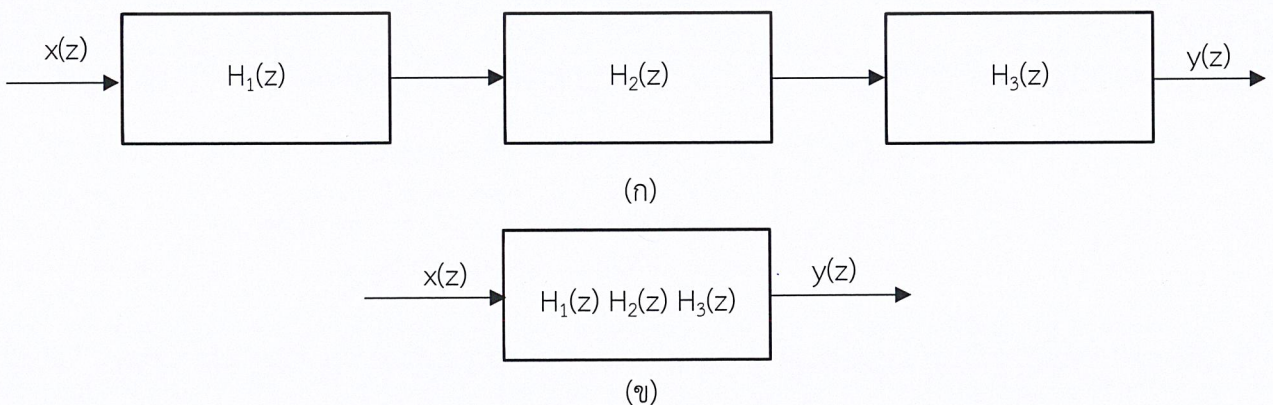
ค่า Parameters ทั้ง 3 ได้แก่ Center Frequency, Gain และ Quality Factor ที่หาได้ในขั้นตอน Pre processing และการใช้ Fuzzy Logic จะถูกนำมาใช้ในการหาค่า Filter Coefficient ของ Peaking Filter ซึ่งสรุปได้ตามสมการ

$$H(z) = \frac{\left(\frac{b_0}{a_0}\right) + \left(\frac{b_1}{a_0}\right)z^{-1} + \left(\frac{b_2}{a_0}\right)z^{-2}}{1 + \left(\frac{a_1}{a_0}\right)z^{-1} + \left(\frac{a_2}{a_0}\right)z^{-2}} \quad (3.6)$$

$$\begin{aligned} A &= \sqrt{10^{(G/20)}} & b_0 &= 1 + (\alpha \times A) \\ \omega_c &= 2 \times \pi \times (f_c / f_s) & b_1 &= (-2) \times \omega C \\ \omega S &= \sin(\omega_c) & b_2 &= 1 - (\alpha \times A) \\ \omega C &= \cos(\omega_c) & a_0 &= 1 + (\alpha / A) \\ \alpha &= \omega S / (2 \times Q) & a_1 &= (-2) \times \omega C \\ & & a_2 &= 1 - (\alpha / A) \end{aligned}$$

- โดย
- G คือ Gain ที่หาได้จากการใช้ Fuzzy Logic
 - f_s คือ Frequency Sampling rate ที่กำหนดตามไฟล์ของสัญญาณอินพุต
 - f_c คือ Frequency Center ที่หาค่าได้จากในขั้นตอน Pre Processing
 - Q คือ Quality Factor ที่หาได้จากการใช้ Fuzzy Logic

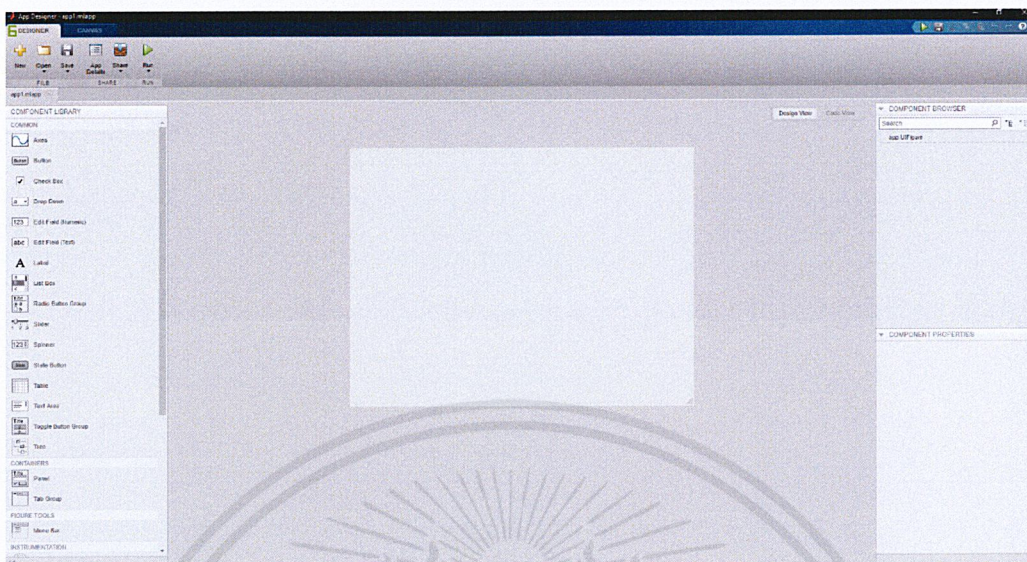
เนื่องจากการทำ Parametric Equalizer ที่มีทั้งหมด 3 Band นั้นหมายความว่าเราต้องสร้าง Digital Filter 3 ตัวที่เชื่อมต่อกันด้วยการนำ Transfer Function ของ Digital Filter ทั้ง 3 ตัวมาเชื่อมต่อกันแบบอนุกรม หรือที่เรียกว่า Cascade Form



รูปที่ 3.14 รูป Diagram ของ (ก) Cascade Form และ (ข) Equivalent Transfer Function

3.1.5 การออกแบบหน้าต่าง User Interface

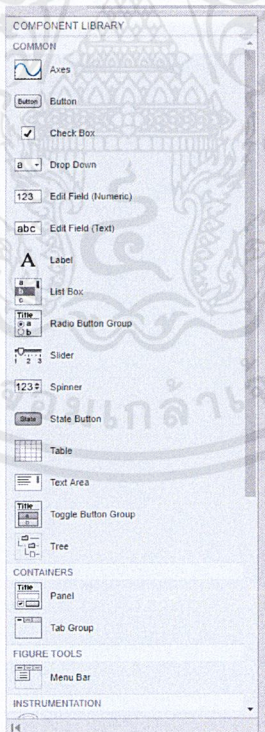
หน้าต่าง User Interface ออกแบบโดยการใช้ฟังก์ชัน App Designer ของ MATLAB



รูปที่ 3.15 รูปแสดง หน้าต่างฟังก์ชัน App Designer ของ MATLAB

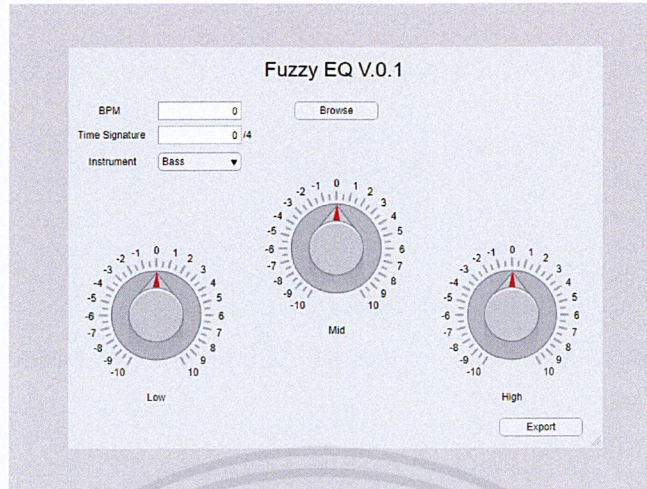
ขั้นตอนการออกแบบ

- 1) เลือก Component ที่ต้องใช้ในโปรแกรมบนแถบ Component Library



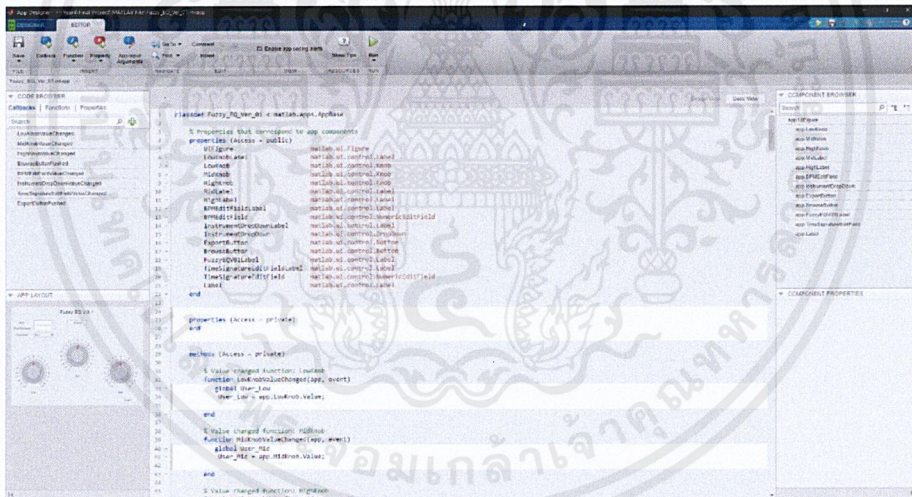
รูปที่ 3.16 รูปแสดงแถบ Component Library บน App Designer

2) ทำการจัดวาง Component ตามรูปแบบที่ต้องการ



รูปที่ 3.17 รูปแสดงการจัดเรียง Component บน App Designer

3) ทำการเขียน Code เพื่อเชื่อมโยงแต่ละ Component เข้ากับคำสั่งต่างๆ และทำการรวมโปรแกรมส่วน Fuzzy Logic System และส่วนของ Digital Filter เข้าสู่โปรแกรมเป็นขั้นตอนสุดท้าย



รูปที่ 3.18 รูปแสดงการเขียนCodeบน App Designer

3.2 Digital Graphic Equalizer

3.2.1 การออกแบบ

Transfer function ที่ใช้ในการออกแบบ Graphic Equalizer คือฟังก์ชัน Peaking Filter เช่นเดียวกันกับที่ใช้ในการออกแบบ Parametric Equalizer ดังที่แสดงอยู่ในสมการที่ 3.6 Graphic Equalizer มี Parameter ที่ปรับค่าได้มีเพียง Gain เท่านั้น มีการกำหนดขอบเขตการเพิ่ม-ลดค่า กำลังขยายเอาไว้ที่ ± 12 dB และได้ออกแบบให้มี Band ที่สามารถปรับค่าได้รวมทั้งหมด 7 Band ซึ่งมีค่า Center Frequency อยู่ที่ 125, 250, 500, 1000, 2000, 4000 และ 8000 Hz ตามลำดับ และแต่ละย่านจะมีค่า Quality factor = 6 ซึ่งจาก Parameters ที่กำหนดไว้ทำให้ได้ Transfer function ของแต่ละ Band เป็น

$$\text{ย่านที่ 1 } H_1(z) = \frac{\frac{1+(0.0015 \times A_1)}{1+(0.0015 \div A_1)} + \frac{(-2) \times (0.9998)}{1+(0.0015 \div A_1)} z^{-1} + \frac{1+(0.0015 \times A_1)}{1+(0.0015 \div A_1)} z^{-2}}{1 + \frac{(-2) \times (0.9998)}{1+(0.0015 \div A_1)} z^{-1} + \frac{1-(0.0015 \div A_1)}{1+(0.0015 \div A_1)} z^{-2}}$$

$$\text{ย่านที่ 2 } H_2(z) = \frac{\frac{1+(0.0030 \times A_2)}{1+(0.0030 \div A_2)} + \frac{(-2) \times (0.9994)}{1+(0.0030 \div A_2)} z^{-1} + \frac{1+(0.0030 \times A_2)}{1+(0.0030 \div A_2)} z^{-2}}{1 + \frac{(-2) \times (0.9994)}{1+(0.0030 \div A_2)} z^{-1} + \frac{1-(0.0030 \div A_2)}{1+(0.0030 \div A_2)} z^{-2}}$$

$$\text{ย่านที่ 3 } H_3(z) = \frac{\frac{1+(0.0059 \times A_3)}{1+(0.0059 \div A_3)} + \frac{(-2) \times (0.9975)}{1+(0.0059 \div A_3)} z^{-1} + \frac{1+(0.0059 \times A_3)}{1+(0.0059 \div A_3)} z^{-2}}{1 + \frac{(-2) \times (0.9975)}{1+(0.0059 \div A_3)} z^{-1} + \frac{1-(0.0059 \div A_3)}{1+(0.0059 \div A_3)} z^{-2}}$$

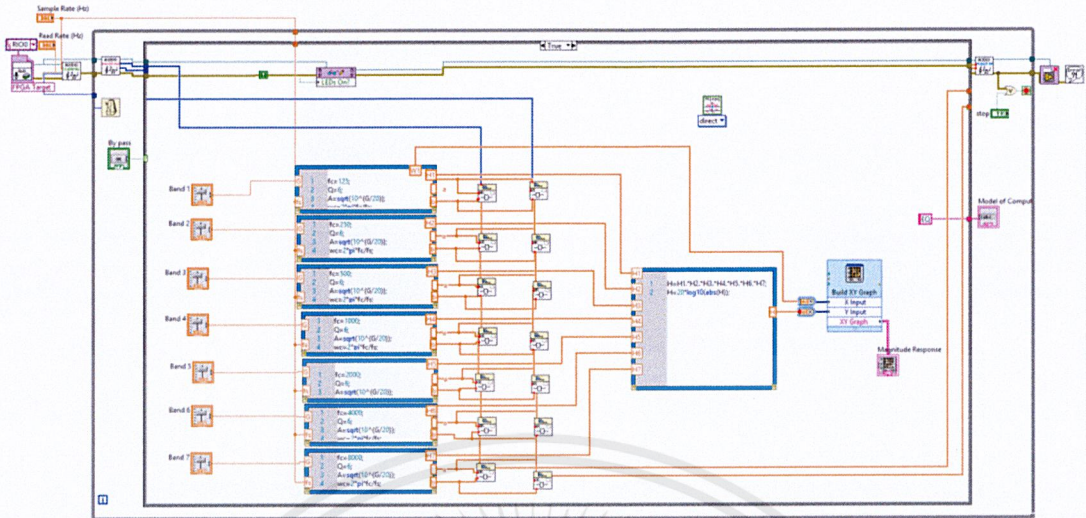
$$\text{ย่านที่ 4 } H_4(z) = \frac{\frac{1+(0.0118 \times A_4)}{1+(0.0118 \div A_4)} + \frac{(-2) \times (0.9899)}{1+(0.0118 \div A_4)} z^{-1} + \frac{1+(0.0118 \times A_4)}{1+(0.0118 \div A_4)} z^{-2}}{1 + \frac{(-2) \times (0.9899)}{1+(0.0118 \div A_4)} z^{-1} + \frac{1-(0.0118 \div A_4)}{1+(0.0118 \div A_4)} z^{-2}}$$

$$\text{ย่านที่ 5 } H_5(z) = \frac{\frac{1+(0.0234 \times A_5)}{1+(0.0234 \div A_5)} + \frac{(-2) \times (0.9597)}{1+(0.0234 \div A_5)} z^{-1} + \frac{1+(0.0234 \times A_5)}{1+(0.0234 \div A_5)} z^{-2}}{1 + \frac{(-2) \times (0.9597)}{1+(0.0234 \div A_5)} z^{-1} + \frac{1-(0.0234 \div A_5)}{1+(0.0234 \div A_5)} z^{-2}}$$

$$\text{ย่านที่ 6 } H_6(z) = \frac{\frac{1+(0.0450 \times A_6)}{1+(0.0450 \div A_6)} + \frac{(-2) \times (0.8420)}{1+(0.0450 \div A_6)} z^{-1} + \frac{1+(0.0450 \times A_6)}{1+(0.0450 \div A_6)} z^{-2}}{1 + \frac{(-2) \times (0.8420)}{1+(0.0450 \div A_6)} z^{-1} + \frac{1-(0.0450 \div A_6)}{1+(0.0450 \div A_6)} z^{-2}}$$

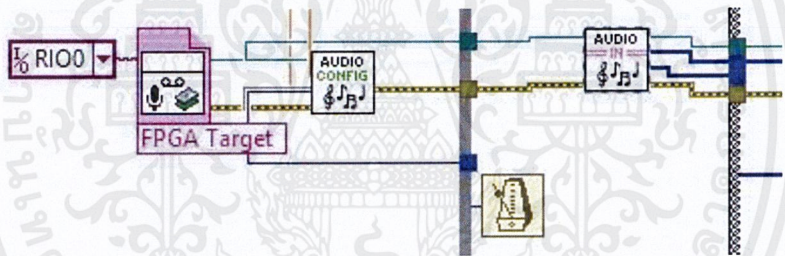
$$\text{ย่านที่ 7 } H_7(z) = \frac{\frac{1+(0.0757 \times A_7)}{1+(0.0757 \div A_7)} + \frac{(-2) \times (0.4178)}{1+(0.0757 \div A_7)} z^{-1} + \frac{1+(0.0757 \times A_7)}{1+(0.0757 \div A_7)} z^{-2}}{1 + \frac{(-2) \times (0.4178)}{1+(0.0757 \div A_7)} z^{-1} + \frac{1-(0.0757 \div A_7)}{1+(0.0757 \div A_7)} z^{-2}}$$

3.2.2 การเขียนโปรแกรมบน LabVIEW เพื่อควบคุม NI myRIO



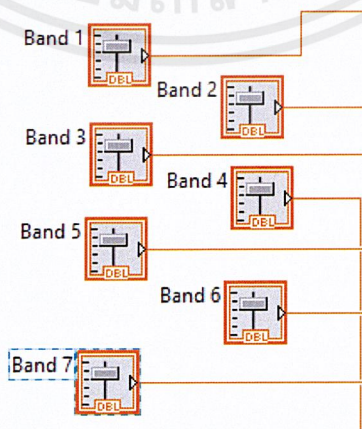
รูปที่ 3.19 รูปแสดงภาพรวม Block Diagram ของระบบ

1) นำเข้าชุดคำสั่งควบคุมอินพุต เอาต์พุต สัญญาณเสียง ให้สัญญาณอินพุตรับเข้าผ่านทางช่อง Audio/in และให้สัญญาณเอาต์พุตส่งออกทางช่อง Audio/out ของบอร์ด NI myRIO



รูปที่ 3.20 รูปแสดงส่วนบล็อกคำสั่งอินพุต Audio/in

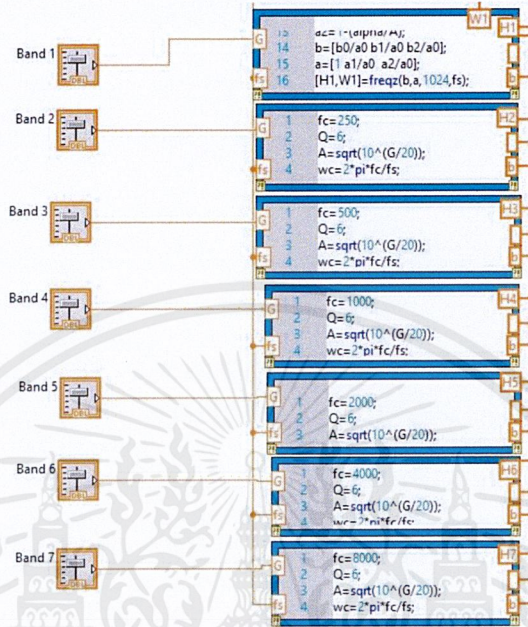
2) สร้าง Fader 7 อันเพื่อสำหรับปรับค่า Parameter Gain โดยกำหนดช่วงการปรับค่าให้อยู่ในช่วง -12 ถึง +12 dB



รูปที่ 3.21 รูปแสดงบล็อกคำสั่งรับข้อมูลจาก Fader

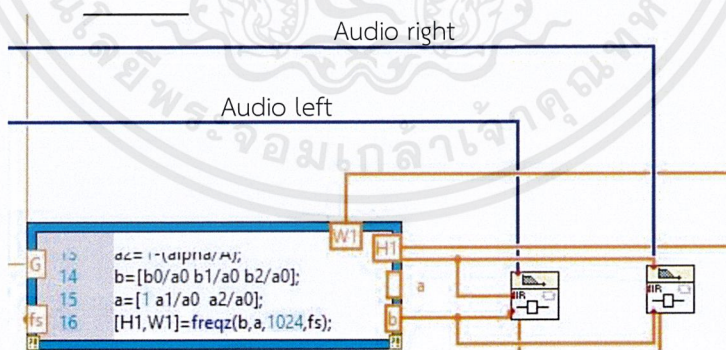
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ³⁹ ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) สร้าง Block รับคำสั่ง MathScript 7 บล็อกเพื่อใช้ในการเขียนคำสั่งสร้าง Transfer Function ของย่านความถี่ทั้ง 7 Band ที่กำหนดได้และโยงเส้นเชื่อมต่อกับ Fader แต่ละอันที่สร้างเตรียมไว้ไปยังกล่องคำสั่งแต่ละอันเพื่อใช้ควบคุมค่า G (Gain) ในช่วงที่กำหนดไว้บน Fader และโยงเส้นนำค่า Sampling Frequency เข้าสู่บล็อกเพื่อใช้ในการคำนวณด้วยเช่นกัน



รูปที่ 3.22 รูปแสดงบล็อกคำสั่ง MathScript ที่เชื่อมโยงเข้ากับ Fader

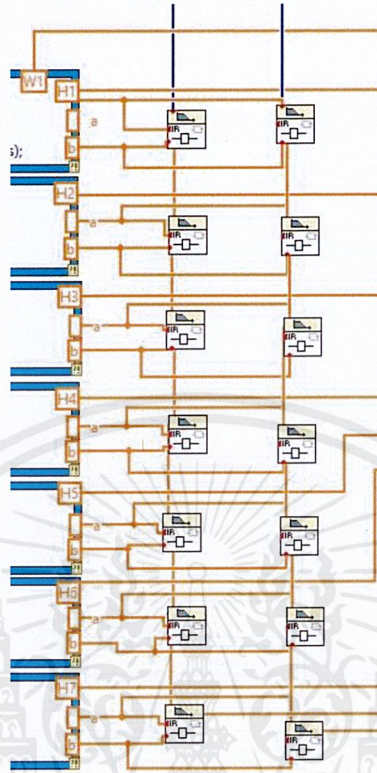
4) ส่งออกค่า Filter Coefficient จาก Block คำสั่ง MathScript ไปป้อนเข้าสู่กล่องคำสั่ง IIR Filter ทั้งนี้จำเป็นต้องทำซ้ำเป็น 2 ชุดเพื่อประมวลผลสัญญาณจากสัญญาณอินพุตฝั่งซ้ายและฝั่งขวาแยกจากกัน



รูปที่ 3.23 รูปแสดงบล็อกคำสั่ง MathScript ที่เชื่อมโยงเข้ากับบล็อก IIR Filter

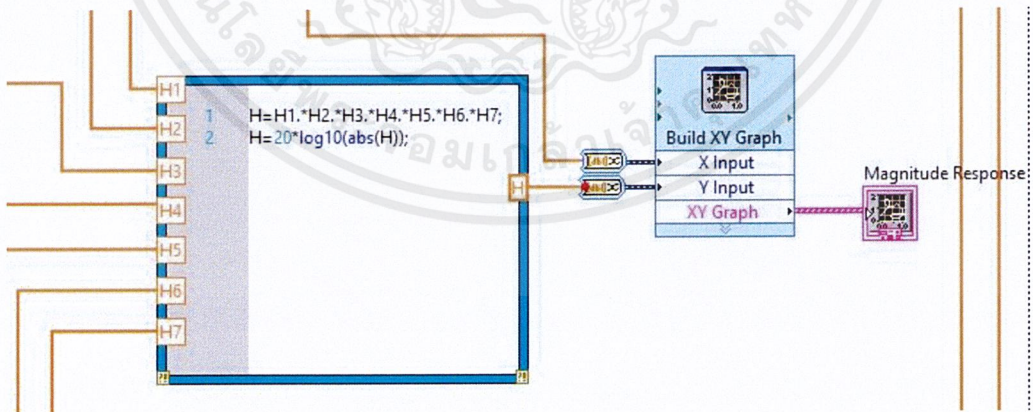
5) เชื่อมต่อค่าสัญญาณอินพุตจาก Audio/in เข้าที่ส่วนสัญญาณอินพุตของ Block IIR Filter และเชื่อมต่อเอาต์พุตของ Block แรกเข้ากับอินพุตของบล็อกต่อไปและทำในทำนองเดียวกัน

กับทุกบล็อกให้เชื่อมต่อกันในรูปแบบ Cascade และนำเอาต์พุตของบล็อกสุดท้ายให้เชื่อมโยงไปยังส่วนเอาต์พุต Audio/out ของบอร์ด NI myRIO



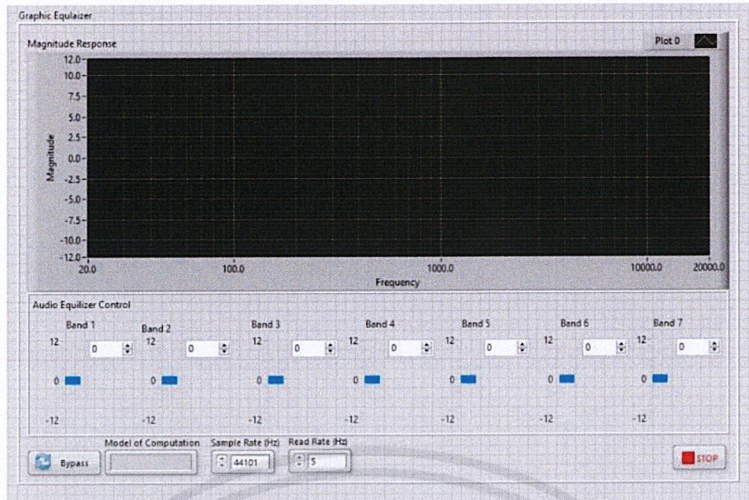
รูปที่ 3.24 รูปแสดงการเชื่อมต่อระหว่างบล็อกคำสั่ง IIR Filter

6) ทำการส่งค่าผลการตอบสนองต่อความถี่ของตัวกรองสัญญาณที่เชื่อมต่อ Cascade กันแล้วไปแสดงผลผ่าน Block คำสั่งกราฟ

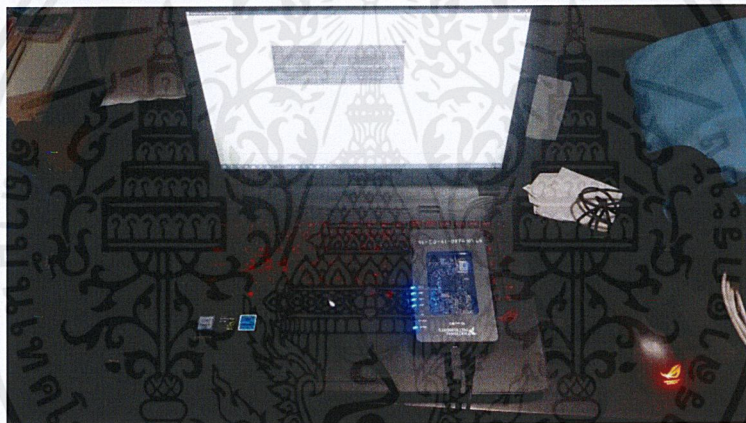


รูปที่ 3.25 รูปแสดงการเชื่อมต่อเพื่อแสดงกราฟผลการตอบสนองต่อความถี่ของตัวกรองสัญญาณ

7) ทำการจัดเรียงหน้าต่างควบคุมของผู้ใช้ให้ง่ายต่อการใช้งาน



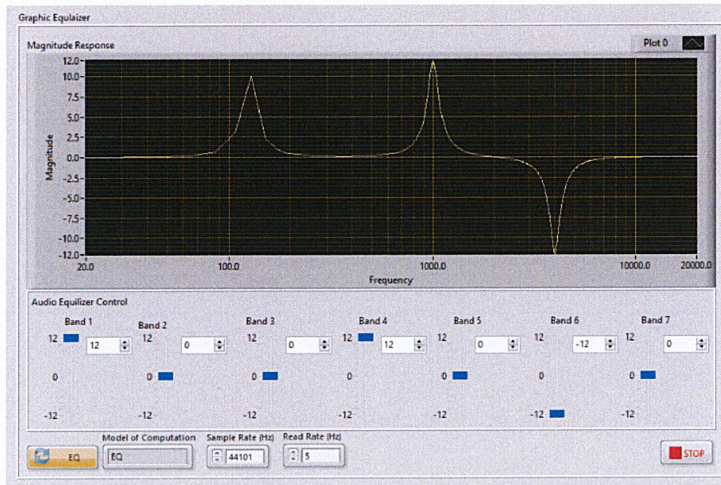
รูปที่ 3.26 รูปแสดง หน้าจอ User interface



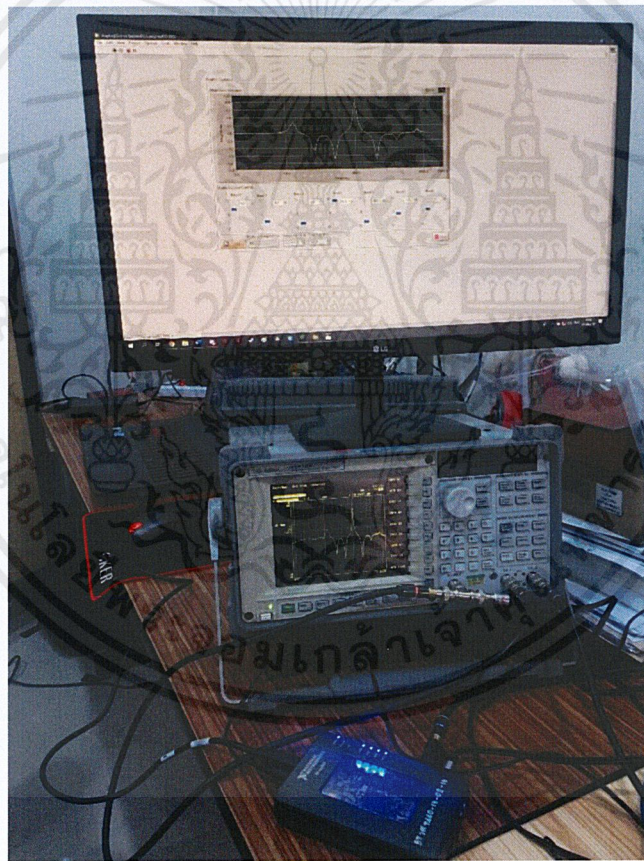
รูปที่ 3.27 รูปการทดสอบการทำงานโดยเชื่อมต่ออุปกรณ์ NI my RIO เข้ากับคอมพิวเตอร์

3.2.3 การทดสอบการทำงาน

ในการทดสอบการทำงาน ในขั้นเบื้องต้นได้ทดสอบโดยใช้การฟังเพื่อหาความเปลี่ยนแปลงของสัญญาณเสียง และได้ทำการแสดงผล Magnitude Response อยู่บนหน้าต่าง User Interface เพื่อทำการตรวจสอบเบื้องต้นว่าโปรแกรมสามารถทำงานได้ในแบบ Real Time และเพื่อเป็นการยืนยันผลว่า Magnitude Response ที่แปลงผลบนหน้าต่าง User Interface มีความถูกต้องจริง จึงได้นำอุปกรณ์ NI myRIO ไปเชื่อมต่อเข้ากับอุปกรณ์ Dynamic Spectrum Analyzer เพื่อวัดผล



รูปที่ 3.28 รูปแสดงหน้าจอ User interface เมื่อโปรแกรมทำงาน



รูปที่ 3.29 รูปแสดงการทดสอบโดยการเชื่อมต่ออุปกรณ์ NI myRIO
เข้ากับเครื่อง Dynamic Signal Analyzer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ⁴³ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการออกแบบและทดสอบ

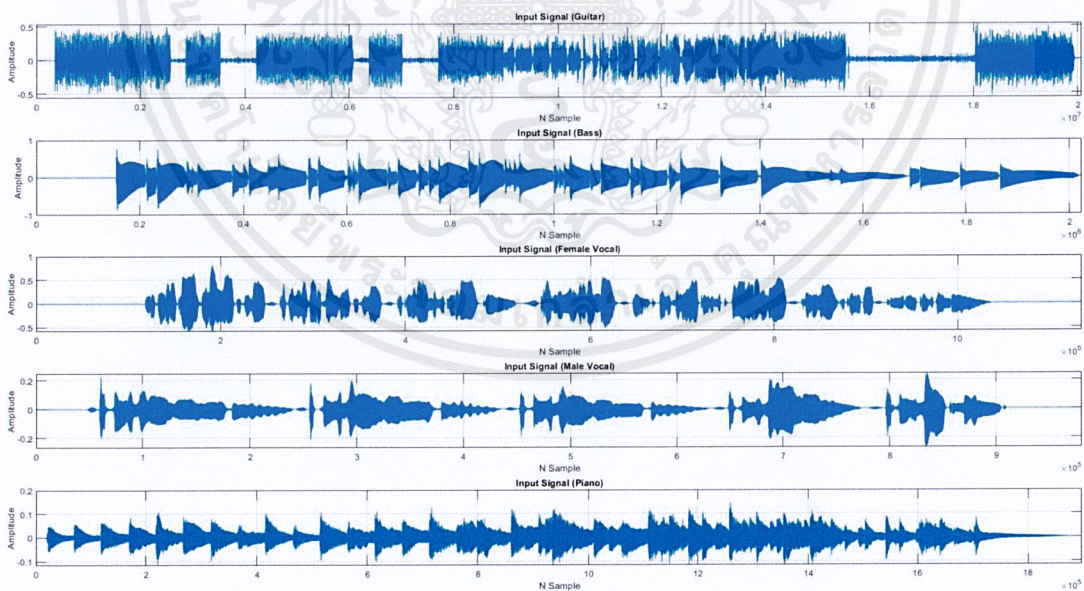
4.1 Digital Parametric Equalizer

4.1.1 ส่วนการรับอินพุต

ค่าอินพุตจาก User ที่ใช้ในการทดสอบผลลัพธ์จากการทดลองแสดงบนตารางที่ 4.1 และสัญญาณเสียงที่ป้อนเข้าสู่ระบบในการทดสอบแสดงอยู่ในรูปที่ 4.1

ตารางที่ 4.1 ค่า User input ในการทดลองกับสัญญาณอินพุตที่แตกต่างกัน

การทดลองที่	Instrument	Tempo	Time Signature	Low	Mid	High
1	Guitar	120	4/4	-3	7	4
2	Bass	68	4/4	3	-1	6
3	Piano	107	4/4	-10	8	5
4	Female Vocal	84	4/4	-2	4	-8
5	Male Vocal	108	4/4	-6	6	4



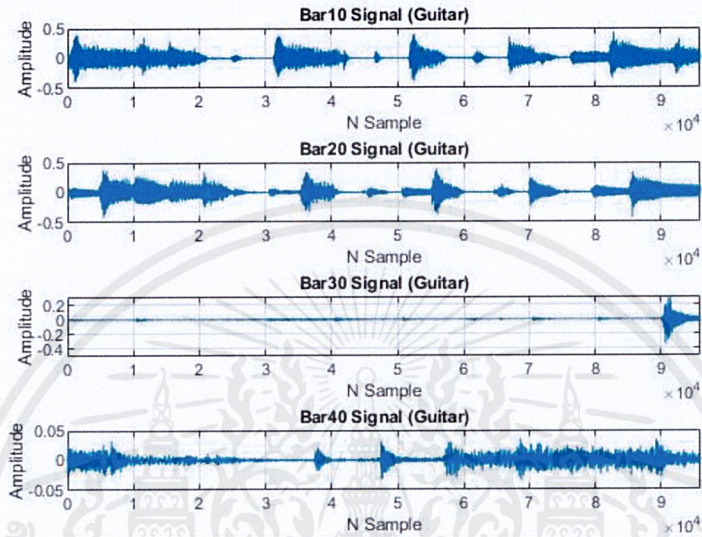
รูปที่ 4.1 รูปแสดงสัญญาณเสียงที่ป้อนเข้าสู่ระบบในการทดสอบ

4.1.2 ส่วนการทำ Pre-processing

4.1.2.1 การแบ่งสัญญาณเสียงเป็นช่วง

ตัวอย่างสัญญาณเสียงที่ทำการแบ่งสัญญาณออกเป็นห้องทางดนตรีแสดงในรูปแบบที่

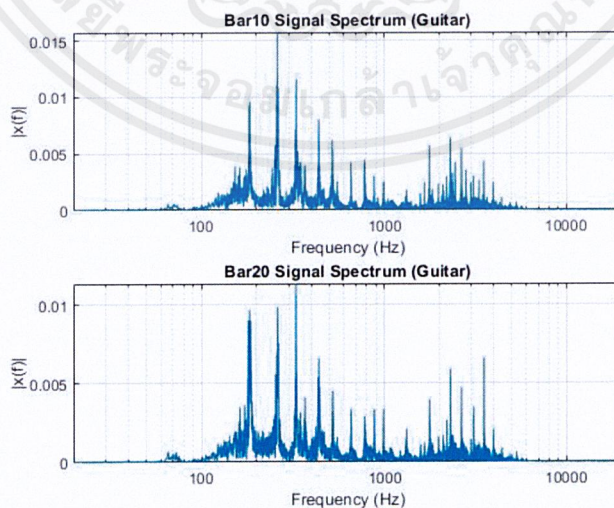
4.2 สุ่มจากข้อมูลสัญญาณห้องที่ 10, 20, 30 และ 40 ตามลำดับจากห้องทั้งหมดที่ถูกแบ่งจำนวน 209 ห้องของสัญญาณเสียงอินพุตเสียงกีตาร์



รูปที่ 4.2 รูปแสดงตัวอย่างสัญญาณเสียงที่ทำการแบ่งห้องทางดนตรีแล้ว

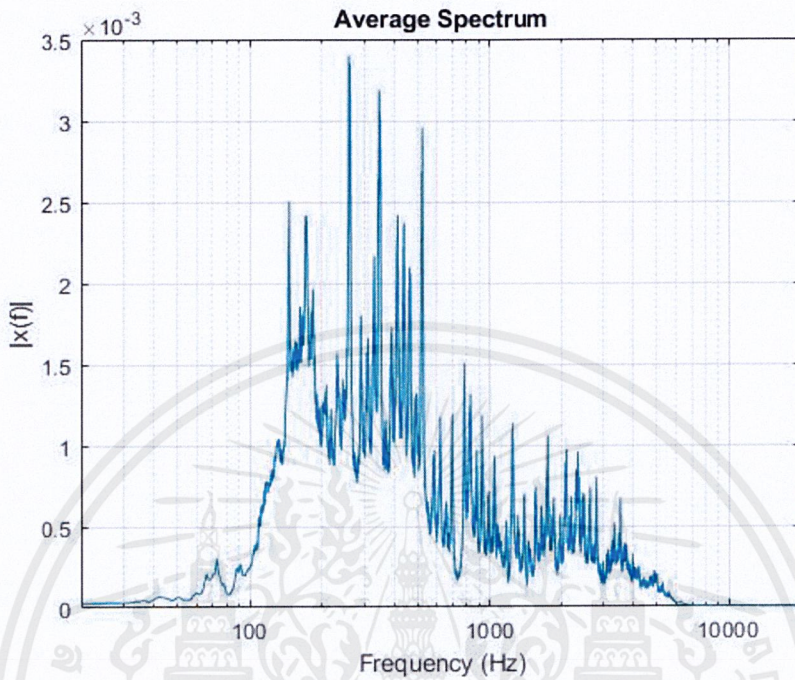
4.1.2.2 Frequency Analyzing

ตัวอย่างสัญญาณเสียงที่ผ่านขั้นตอนการแบ่งช่วงในรูปแบบของ Spectrum แสดงอยู่ในรูปที่ 4.3 สุ่มจากข้อมูลสัญญาณห้องที่ 10 และ 20 ตามลำดับ ของสัญญาณเสียงอินพุตเสียงกีตาร์



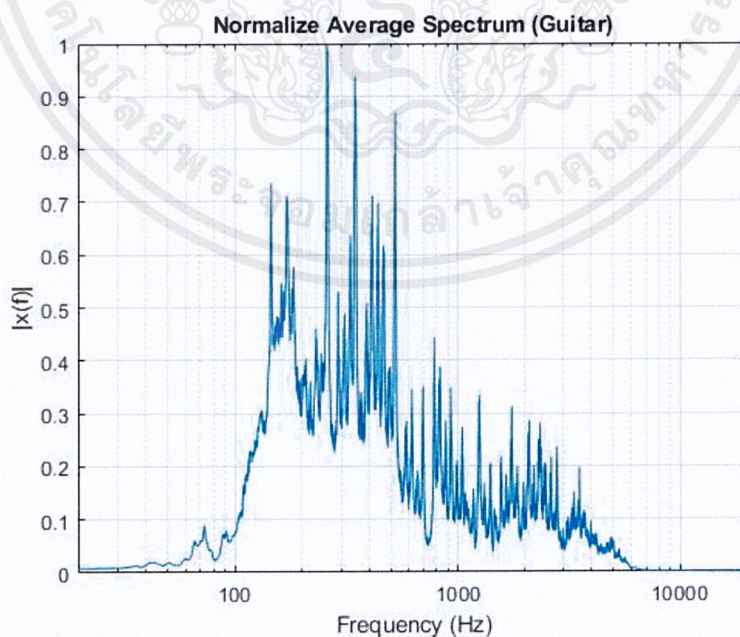
รูปที่ 4.3 รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงที่ทำการแบ่งห้องทางดนตรีแล้ว

ผลลัพธ์การนำข้อมูลที่ถูกแบ่งช่วงตามห้องทางดนตรีทั้งหมดมาหาค่าเฉลี่ยตามตำแหน่งข้อมูลบน Frequency Domain แสดงอยู่ในรูปที่ 4.4 จากข้อมูลสัญญาณเสียงอินพุตเสียงกีตาร์

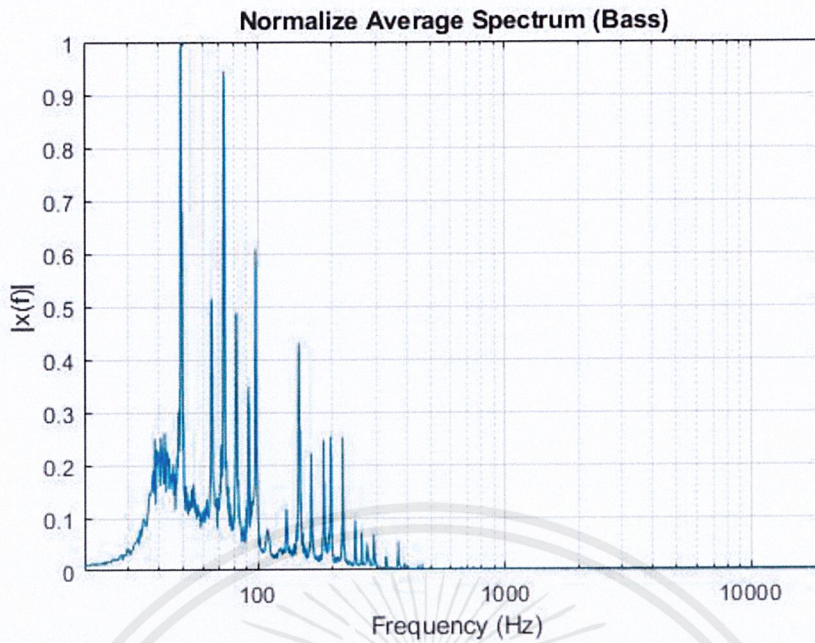


รูปที่ 4.4 รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงที่ทำการหาค่าเฉลี่ยแล้ว

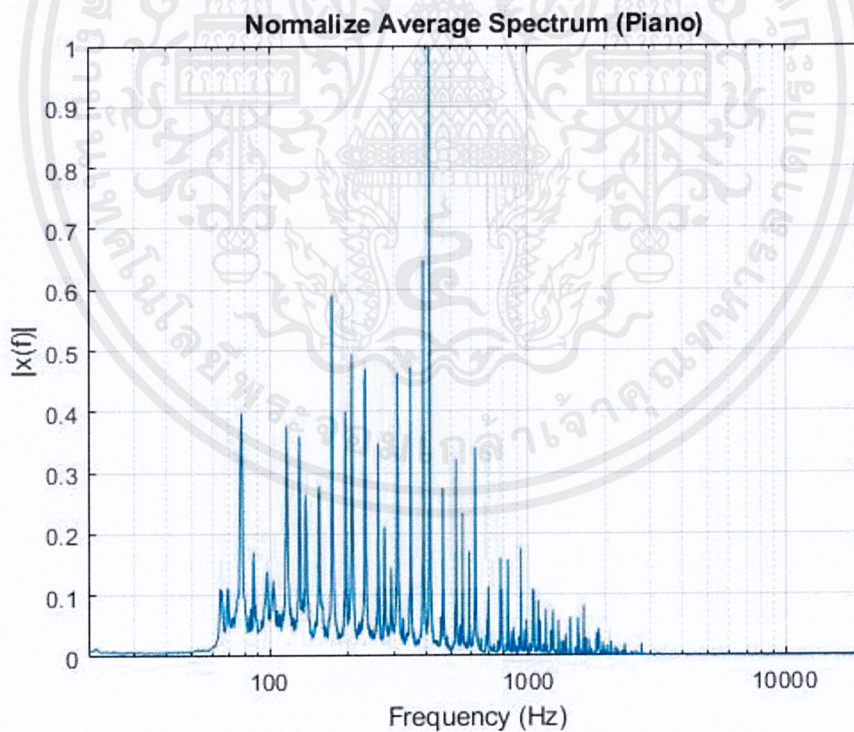
ผลลัพธ์จากการนำข้อมูลมาค่าเฉลี่ยของ Spectrum มาทำการ Normalize ให้ค่า Amplitude สูงสุดที่ค่าเป็น 1 แสดงในรูปที่ 4.5 ถึง 4.9



รูปที่ 4.5 รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงกีตาร์ที่ทำการหาค่าเฉลี่ย และ Normalized แล้ว

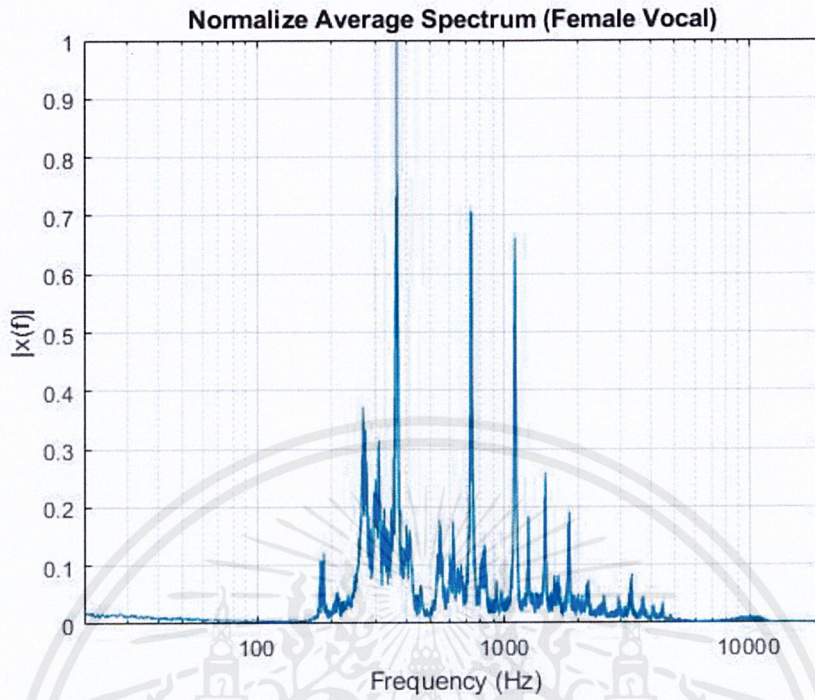


รูปที่ 4.6 รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงเบส ที่ทำการหาค่าเฉลี่ย และ Normalized แล้ว

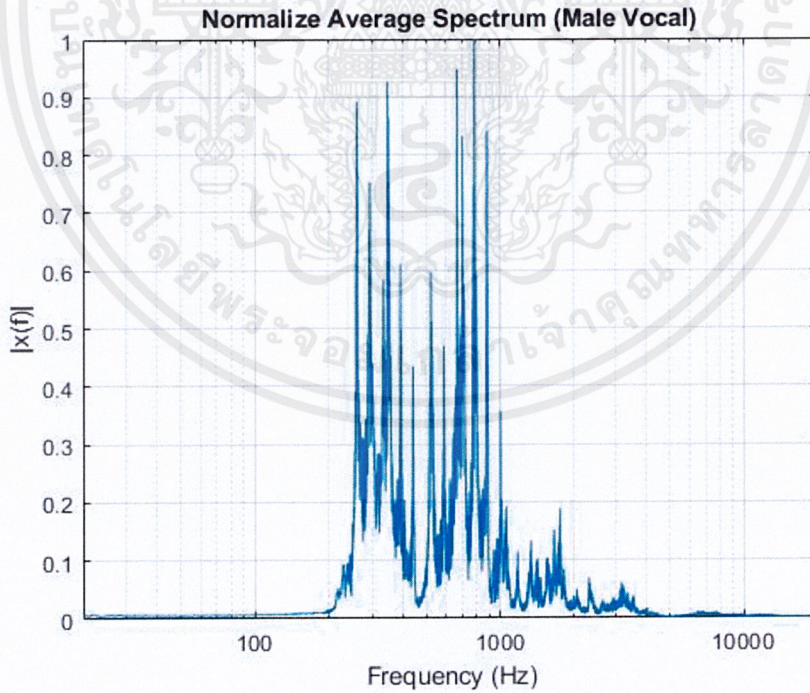


รูปที่ 4.7 รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงเปียโน ที่ทำการหาค่าเฉลี่ย และ Normalized แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ 47 เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงร้องหญิง ที่ทำการหาค่าเฉลี่ย และ Normalized แล้ว



รูปที่ 4.9 รูปแสดงตัวอย่าง Spectrum ของสัญญาณเสียงร้องชาย ที่ทำการหาค่าเฉลี่ย และ Normalized แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ⁴⁸ชานเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์จากขั้นตอนการทำ Pre-processing จากการทดลองทั้งหมด แสดงอยู่บนตารางที่

4.2

ตารางที่ 4.2 ค่าผลลัพธ์ที่ได้จากขั้นตอนการทำ Pre-Processing

Instrument	Frequency Length (Hz)			Center Frequency (Hz)			Frequency Average		
	low	mid	high	low	mid	high	low	mid	high
Guitar	80 - 250	250 - 1250	1250 - 5000	129.5	259.5	4159	0.3186	0.2279	0.0921
Bass	40 - 80	80 - 320	320 - 5000	49.01	196.91	3154.9	0.1903	0.0495	0.0005
Piano	27 - 250	250 - 2000	2000 - 10000	51	415.96	6662.1	0.0826	0.0237	0.0004
Female Vocal	240 - 1200	1200 - 8000	8000 - 16000	368.2	5896	11793	0.1016	0.0151	0.0034
Male Vocal	120 - 900	900 - 8000	8000 - 16000	787.95	6306.8	12614	0.2121	0.0169	0.0027

จากขั้นตอนการทำ Pre-processing ทำให้ทราบ Center Frequency และดัชนีชี้วัดความดังในแต่ละย่านความถี่ของแต่ละชนิดเครื่องดนตรีที่นำมาทดสอบ ได้ผลลัพธ์ตามตารางที่ 4.2 เช่น เสียงกีตาร์ที่นำมาทดสอบ มีย่านความถี่ที่ดังที่สุดใน Low Frequency Band โดยมีดัชนีชี้วัดความดังเท่ากับ 0.3186 และมี Center Frequency เท่ากับ 129.5 Hz และมี Mid Center Frequency ที่ 259.5 Hz ด้วยดัชนีชี้วัดความดังเท่ากับ 0.2279 และ High Center Frequency ที่ 4159 Hz ด้วยดัชนีชี้วัดความดังเท่ากับ 0.0921 ซึ่งมีดัชนีชี้วัดความดังที่ลดหลั่นกันไปเป็น Low Frequency Band มีความดังมากที่สุด ตามด้วย Mid Frequency Band ที่มีความดังปานกลาง และมี High Frequency Band มีความดังน้อยที่สุด เป็นต้น

4.1.3 Fuzzy logic

4.1.3.1 ส่วนการคำนวณค่า Gain

ตัวอย่างผลลัพธ์การหาค่า Gain ของสัญญาณอินพุตเสียงกีตาร์โดยใช้ Fuzzy Logic ผ่าน FIS Tools ของ MATLAB แสดงอยู่ในรูปที่ 4.10



รูปที่ 4.10 รูปแสดงตัวอย่างการใช้ FIS Tools ในการหาค่า Gain ผลลัพธ์ค่า Gain ที่ได้จากการทดลองสัญญาณอินพุตต่างๆ แสดงอยู่ในตารางที่ 4.3

ตารางที่ 4.3 ค่า Gain ที่หาจากการใช้ Fuzzy Logic

Instrument	Gain (dB)		
	low	mid	high
Guitar	-3.05	2.42	-1.44
Bass	2.13	-3.05	9.88
Piano	-3.05	6.02	3.27
Female Vocal	-1.11	1.92	-3.05
Male Vocal	-7.81	7.90	2.05

4.1.3.2 การหาค่า Quality Factor

ตัวอย่างผลลัพธ์การหาค่า Gain ของสัญญาณอินพุตเสียงกีตาร์โดยใช้ Fuzzy Logic ผ่าน FIS Tools ของ MATLAB แสดงอยู่ในรูปที่ 4.11



รูปที่ 4.11 รูปแสดงตัวอย่างการใช้ FIS Tools ในการหาค่า Quality Factor

ผลลัพธ์ค่า Quality Factor ที่ได้จากการทดลองสัญญาณอินพุตต่างๆ แสดงอยู่ในตารางที่ 4.4

ตารางที่ 4.4 ค่า Quality Factor ที่ได้จากการใช้ Fuzzy logic

Instrument	Q		
	low	mid	high
Guitar	2.8758	4.6836	5.9984
Bass	2.8758	1.7858	1.7858
Piano	2.8758	5.9982	5.9996
Female Vocal	2.8758	4.6836	5.9984
Male Vocal	11.6	2.63	6

4.1.4 ผลลัพธ์การทดลอง

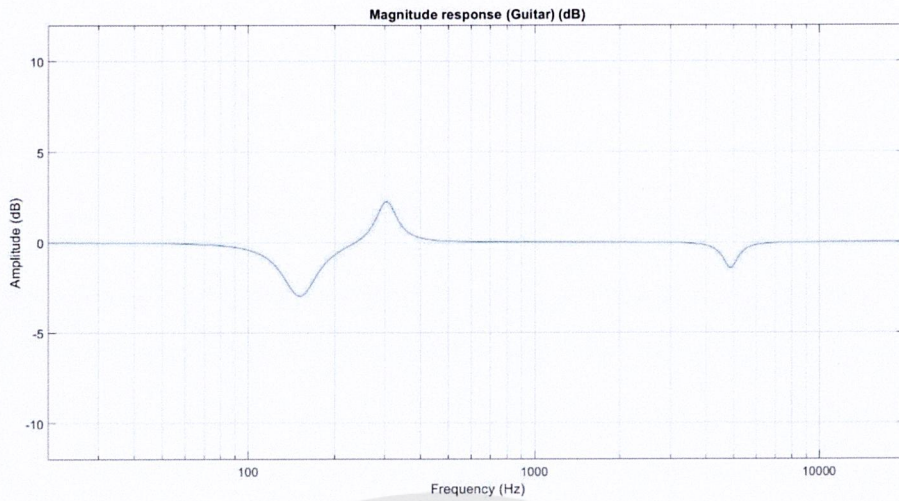
จากการบวนการทั้งหมดของระบบจะได้ค่า Parameters ของ Parametric Equalizer ดังที่แสดงในตารางที่ 4.5

ตารางที่ 4.5 ค่า Parameters ทั้งหมดของ Parametric Equalizer

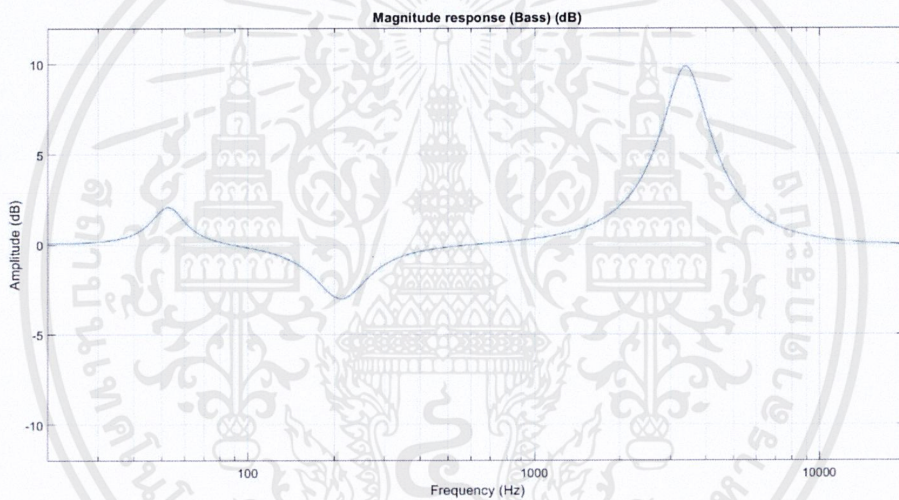
Instrument	Center Frequency (Hz)			Gain (dB)			Q		
	low	mid	high	low	mid	high	low	mid	high
Guitar	129.5	259.5	4159	-3.05	2.42	-1.44	2.8758	4.6836	5.9984
Bass	49.01	196.91	3154.9	2.13	-3.05	9.88	2.8758	1.7858	1.7858
Piano	51	415.96	6662.1	-3.05	6.02	3.27	2.8758	5.9982	5.9996
Female Vocal	368.2	5896	11793	-1.11	1.92	-3.05	2.8758	4.6836	5.9984
Male Vocal	787.95	6306.8	12614	-7.81	7.90	2.05	11.6	2.63	6

จากขั้นตอนการทำ Pre-processing และใช้ Fuzzy Logic ทำให้ได้ค่า Parameters ของ Parametric Equalizer ที่เหมาะสมกับสัญญาณเสียงที่ใช้ทดสอบซึ่งได้ผลลัพธ์ดังนี้ เสียงกีตาร์ มีการปรับค่า Parameters ดังนี้ Low Frequency Band มี Center Frequency = 129.5 Hz, Gain = -3.05 dB และ Quality Factor = 2.8758, Mid Frequency Band มี Center Frequency = 259.5 Hz, Gain = 2.42 dB และ Quality Factor = 4.6836 และมี High Frequency Band มี Center Frequency = 4159, Gain = -1.44 dB และ Quality Factor = 5.9984 ซึ่งให้ผลลัพธ์เป็นไปตามที่ผู้ทดสอบต้องการจากการปรับค่าคะแนนความต้องการ เช่นเดียวกับเสียงเครื่องดนตรีชิ้นอื่นๆ ที่นำมาทดสอบ

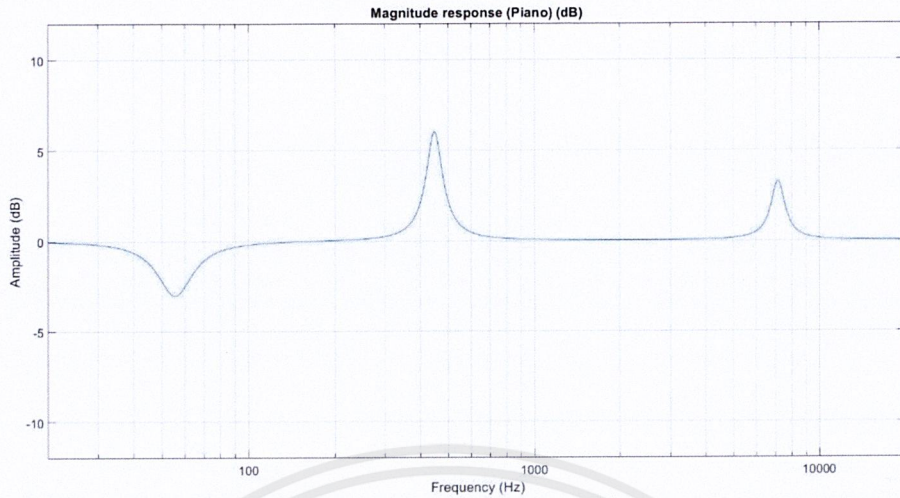
จากค่า Parameters ที่ได้ นำไปใช้ในการออกแบบ Filter ซึ่งผลลัพธ์ Magnitude Response ของ Transfer Function ของ Filter แสดงอยู่ในรูปที่ 4.12 ถึง 4.16



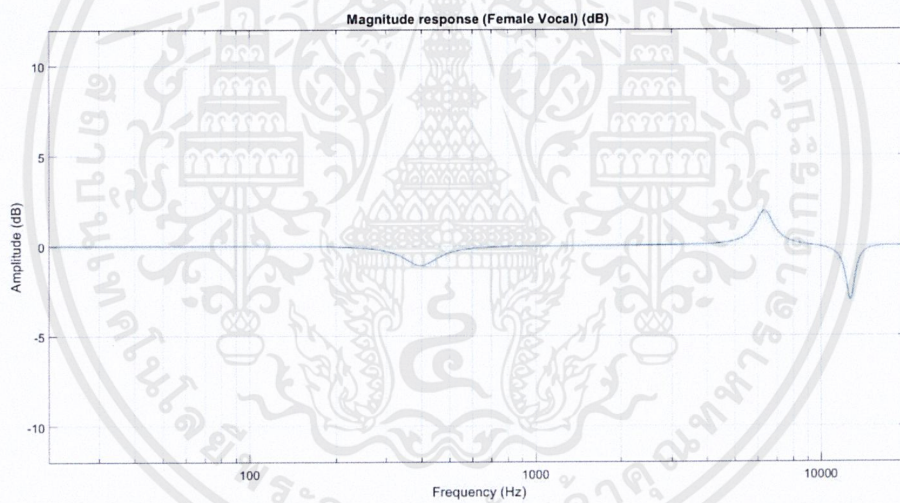
รูปที่ 4.12 รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงกีตาร์เข้าสู่ระบบ



รูปที่ 4.13 รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงเบสเข้าสู่ระบบ

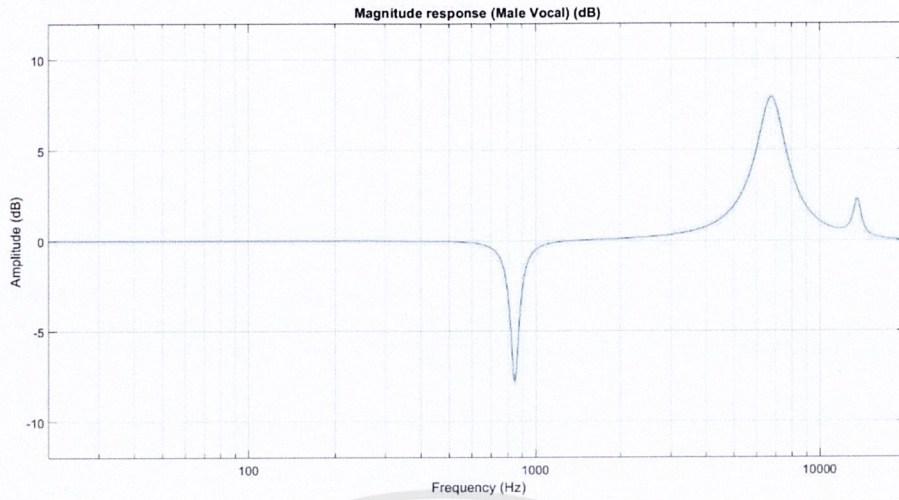


รูปที่ 4.14 รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงเปียโนเข้าสู่ระบบ



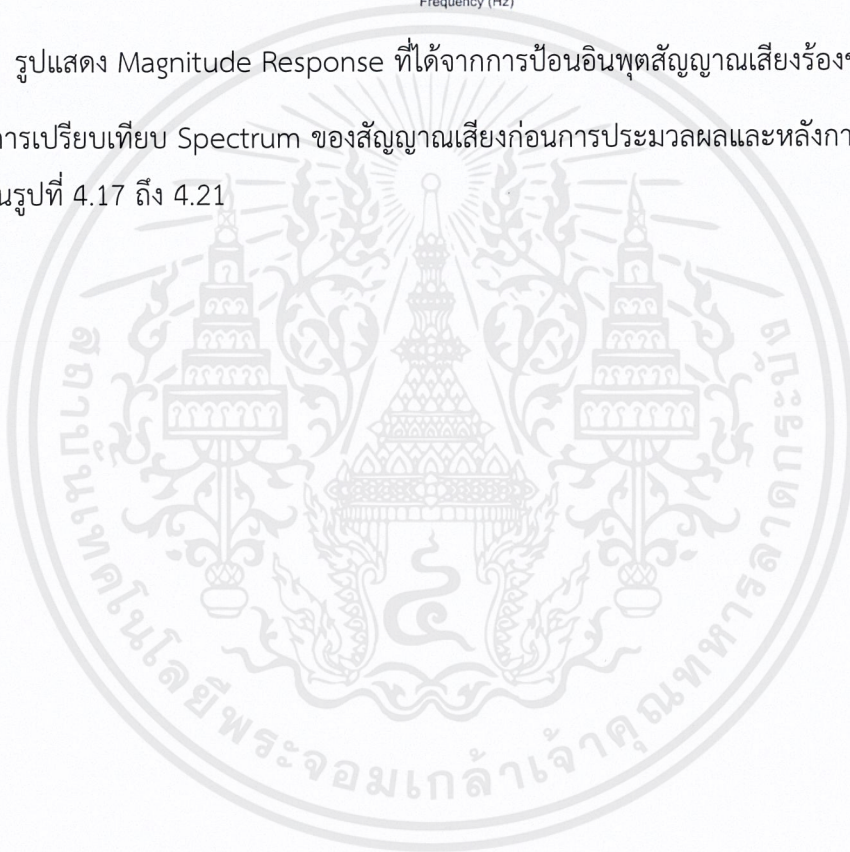
รูปที่ 4.15 รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงร้องหญิงเข้าสู่ระบบ

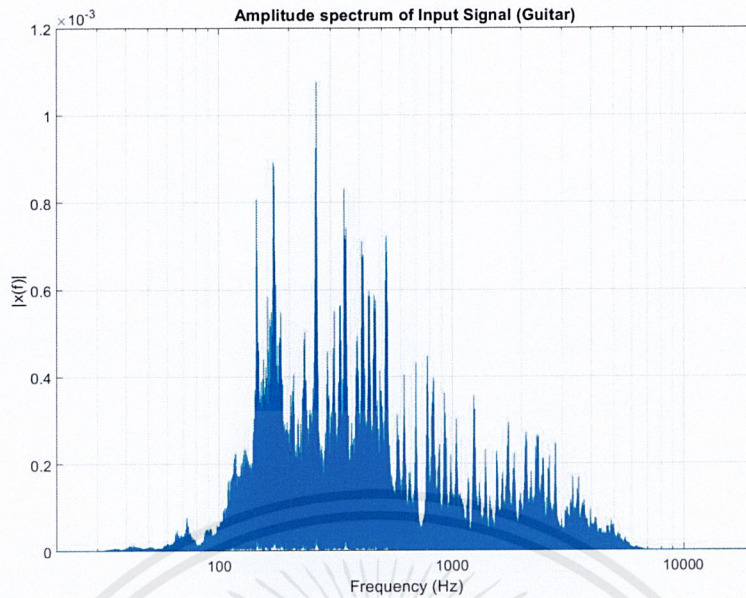
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ 54 เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



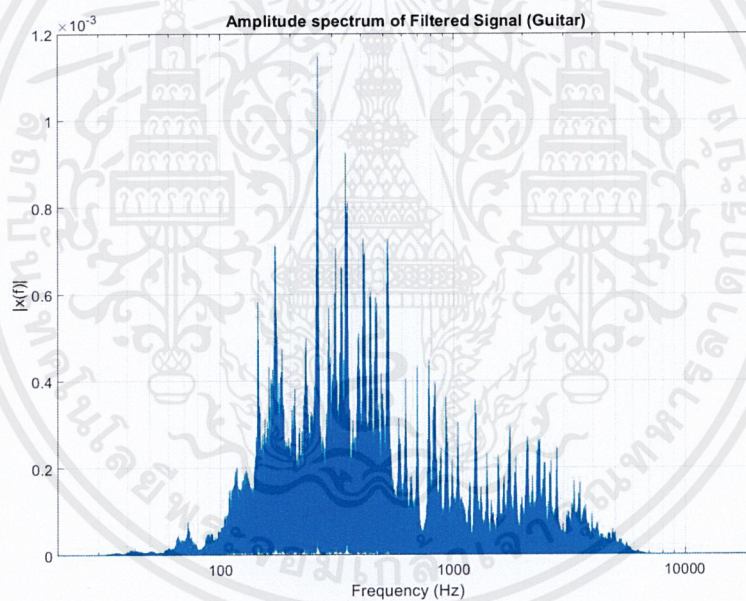
รูปที่ 4.16 รูปแสดง Magnitude Response ที่ได้จากการป้อนอินพุตสัญญาณเสียงร้องชายเข้าสู่ระบบ

ผลการเปรียบเทียบ Spectrum ของสัญญาณเสียงก่อนการประมวลผลและหลังการประมวลผล แสดงอยู่ในรูปที่ 4.17 ถึง 4.21





(ก)

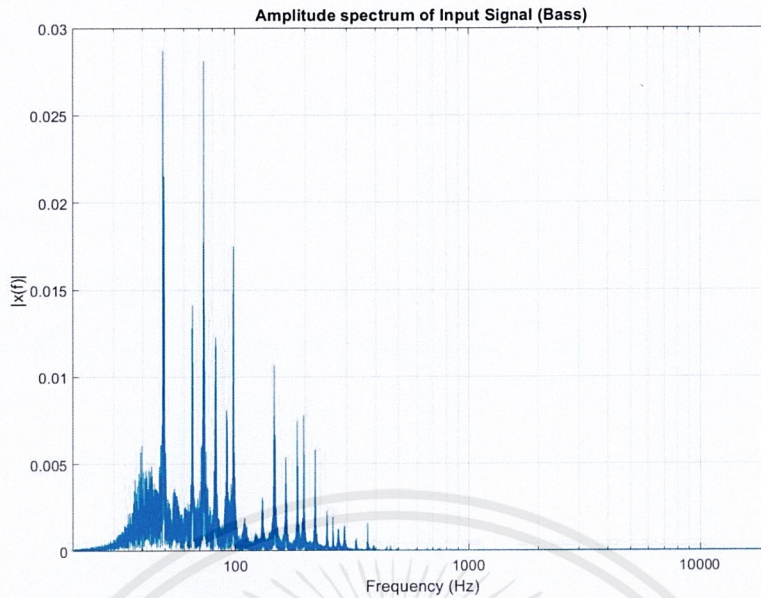


(ข)

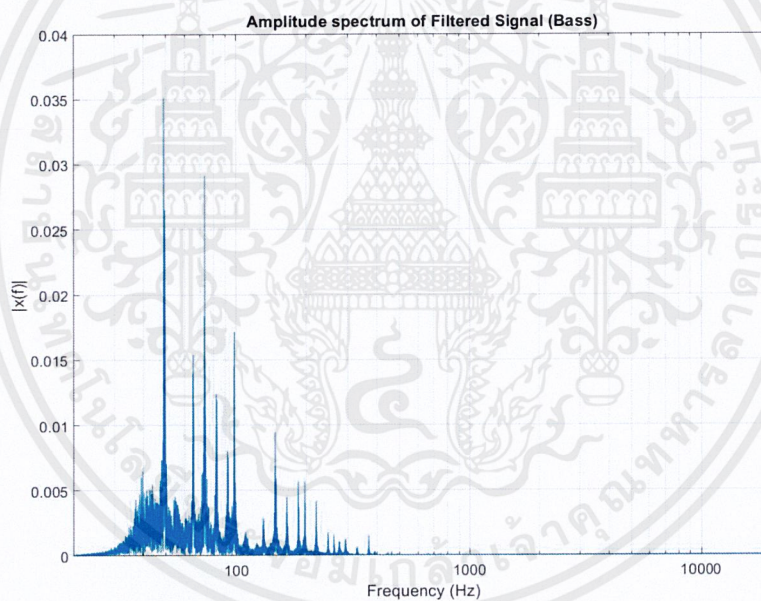
รูปที่ 4.17 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงกีตาร์

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

หลังผ่านการประมวลผลแล้ว Spectrum ของเสียงกีตาร์แสดงให้เห็นว่ามีความถี่ในช่วง 260 Hz เพิ่มขึ้นอย่างชัดเจน และในช่วงความถี่ 130 Hz มีการลดลงอย่างสังเกตเห็นได้ชัด ซึ่งเป็นไปตาม Transfer Function ของ Parametric Equalizer ที่นำมาประมวลผลกับเสียงกีตาร์



(ก)

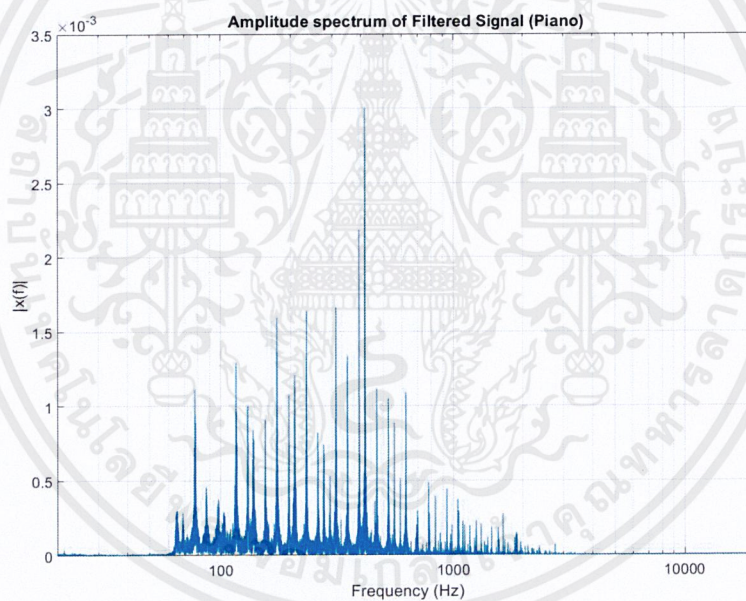
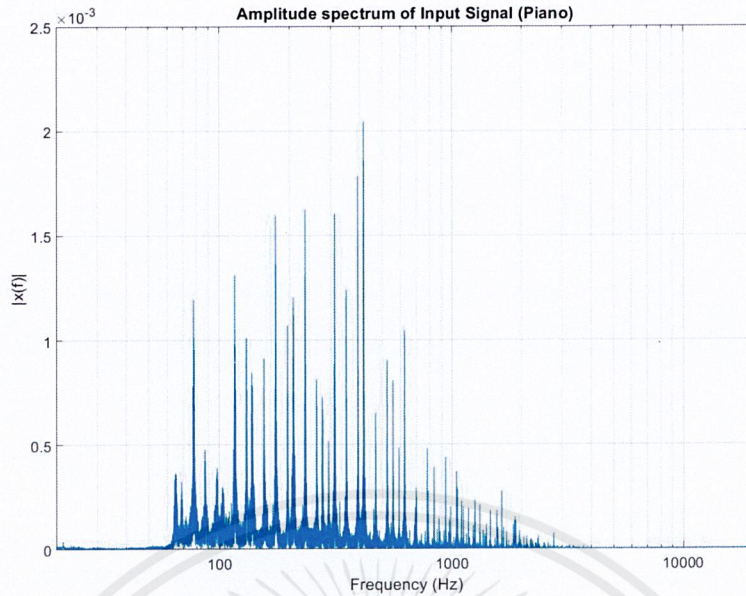


(ข)

รูปที่ 4.18 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเบส

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

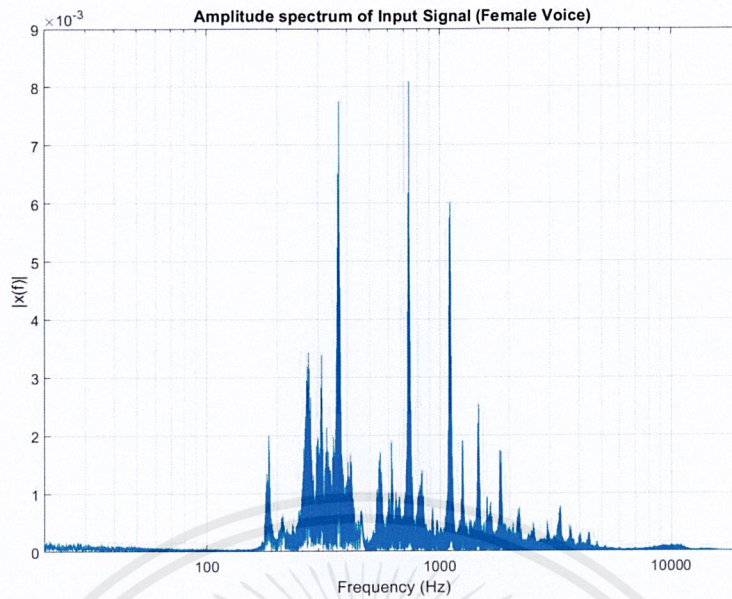
หลังผ่านการประมวลผลแล้ว Spectrum ของเสียงเบสแสดงให้เห็นว่ามีความถี่ประมาณ 40 Hz มีการเพิ่มขึ้น และช่วงความถี่ 200 Hz มีการลดลงอย่างเห็นได้ชัด ซึ่งเป็นไปตาม Transfer Function ของ Parametric Equalizer ที่นำมาประมวลผลกับเสียงเบส



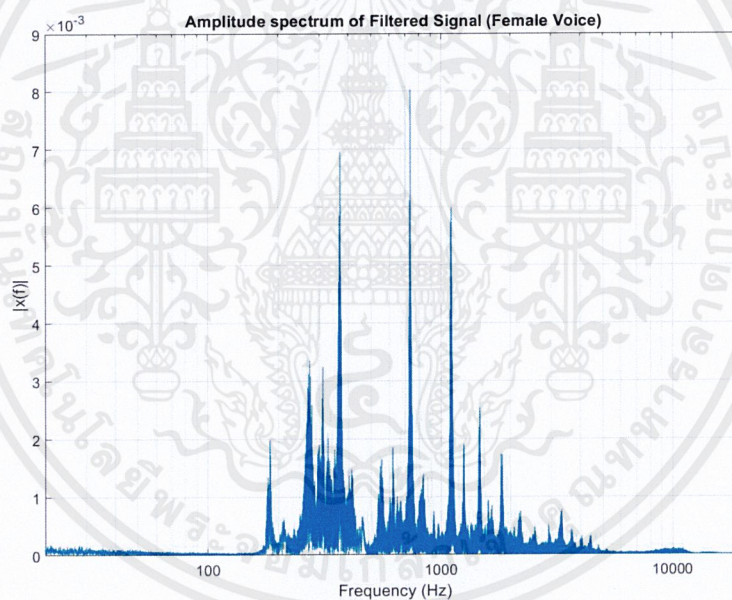
รูปที่ 4.19 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเปียโน

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

หลังผ่านการประมวลผลแล้ว Spectrum ของเสียงเปียโนแสดงให้เห็นว่ามีความถี่ประมาณ 300 Hz มีการเพิ่มขึ้นอย่างมาก และที่ความถี่ 50 Hz มีการลดลงเล็กน้อย ซึ่งเป็นไปตาม Transfer Function ของ Parametric Equalizer ที่นำมาประมวลผลกับเสียงเปียโน



(ก)

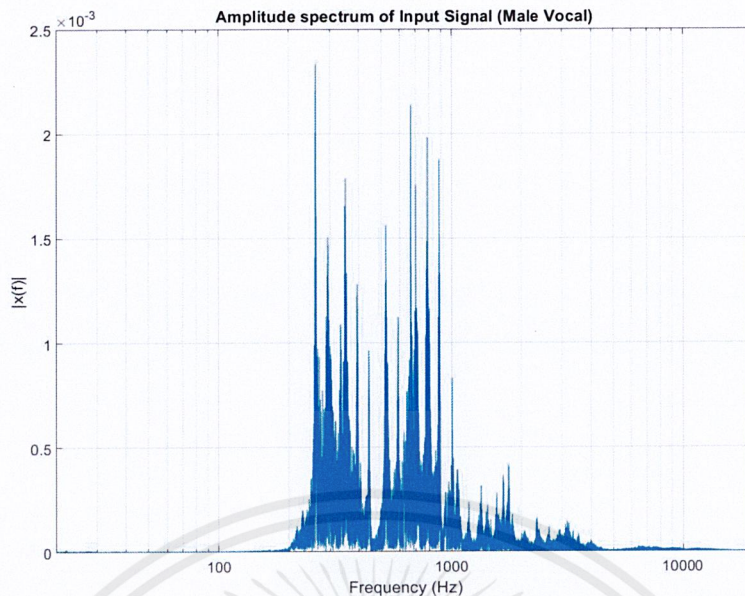


(ข)

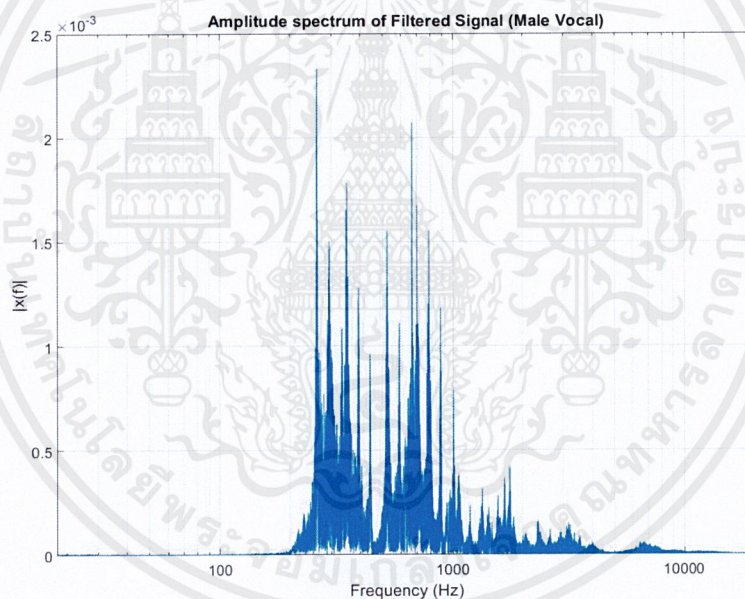
รูปที่ 4.20 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องหญิง

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

หลังผ่านการประมวลผลแล้ว Spectrum ของเสียงร้องผู้หญิงแสดงให้เห็นว่ามีความถี่ประมาณ 200 Hz มีการลดลงอย่างเห็นได้ชัด ซึ่งเป็นไปตาม Transfer Function ของ Parametric Equalizer ที่นำมาประมวลผลกับเสียงร้องผู้หญิง



(ก)



(ข)

รูปที่ 4.21 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องชาย

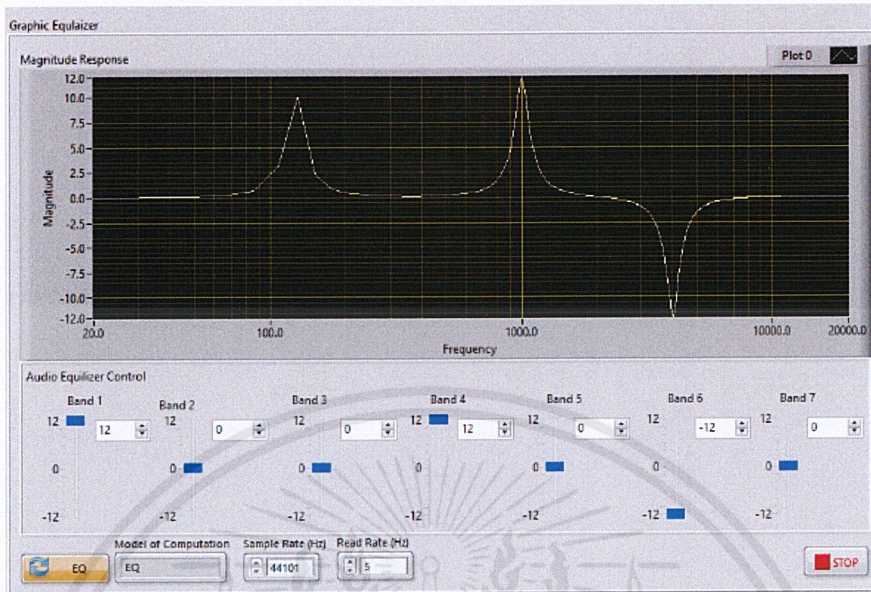
(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

หลังผ่านการประมวลผลแล้ว Spectrum ของเสียงร้องชายแสดงให้เห็นว่ามีความถี่ประมาณ 800 Hz มีการลดลงอย่างสังเกตเห็นได้และ 7000 Hz มีการเพิ่มขึ้นอย่างสังเกตเห็นได้ ซึ่งเป็นไปตาม Transfer Function ของ Parametric Equalizer ที่นำมาประมวลผลกับเสียงร้องชาย

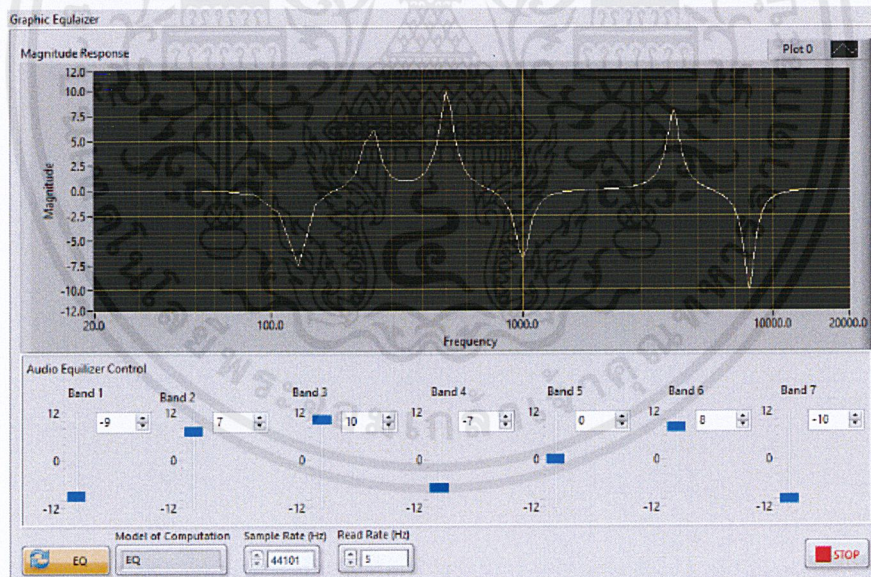
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ 60 ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 Digital Graphic Equalizer

4.2.1 การแสดงผลลัพธ์ Magnitude response ผ่านหน้าจอ User Interface

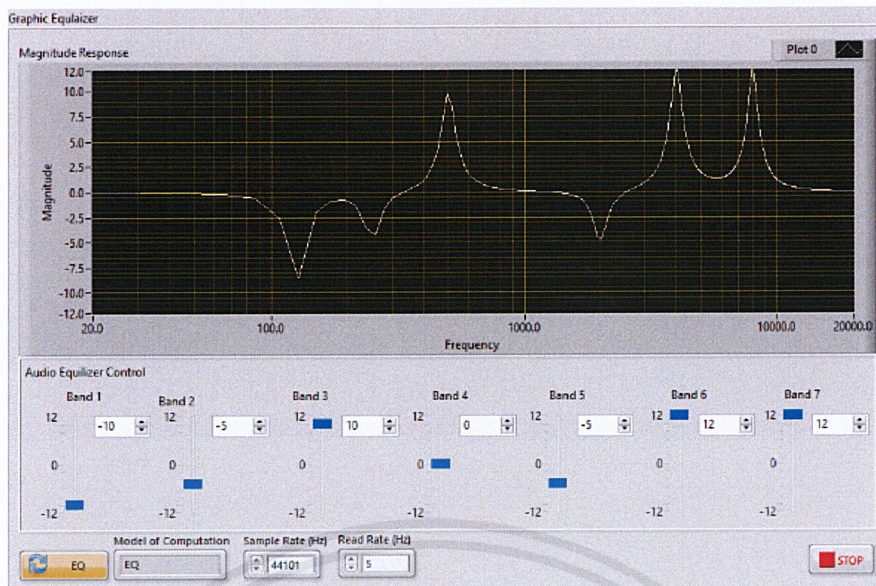


รูปที่ 4.22 รูปแสดงการทดลองที่ 1 มีค่า Gain แต่ละ Band คือ [12 0 0 12 0 -12 0] dB

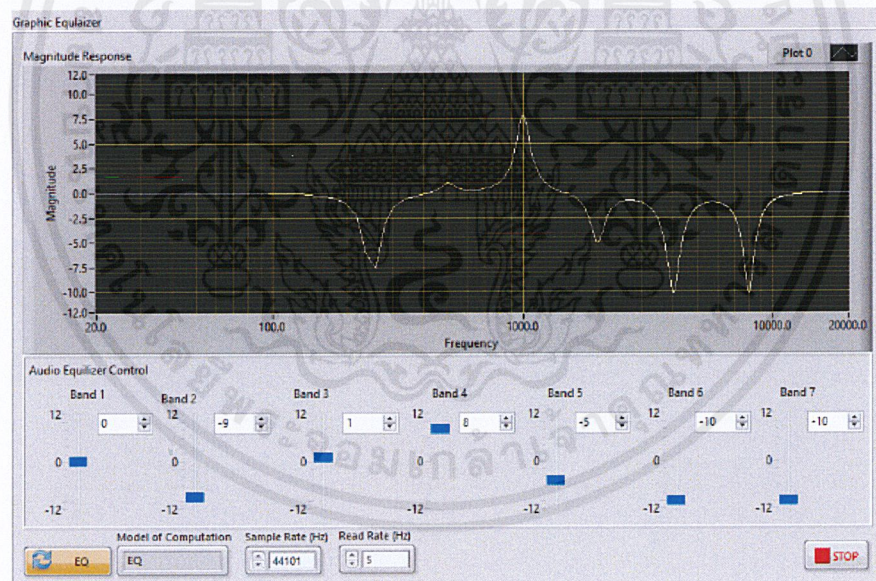


รูปที่ 4.23 รูปแสดงการทดลองที่ 2 มีค่า Gain แต่ละ Band คือ [-9 7 10 -7 0 8 -10] dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ⁶¹ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

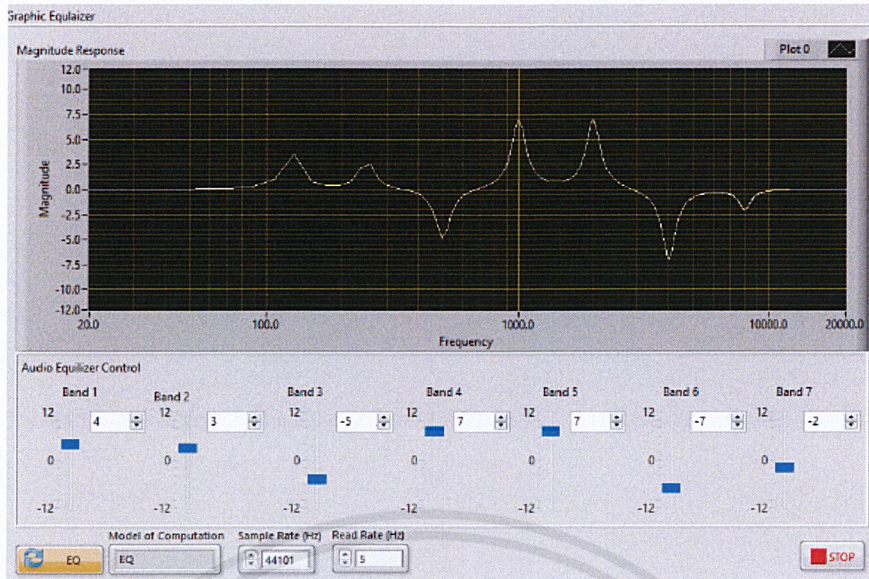


รูปที่ 4.24 รูปแสดงการทดลองที่ 3 มีค่า Gain แต่ละ Band คือ [-10 -5 10 0 -5 12 12] dB



รูปที่ 4.25 รูปแสดงการทดลองที่ 4 มีค่า Gain แต่ละ Band คือ [0 -9 1 8 -5 -10 -10] dB

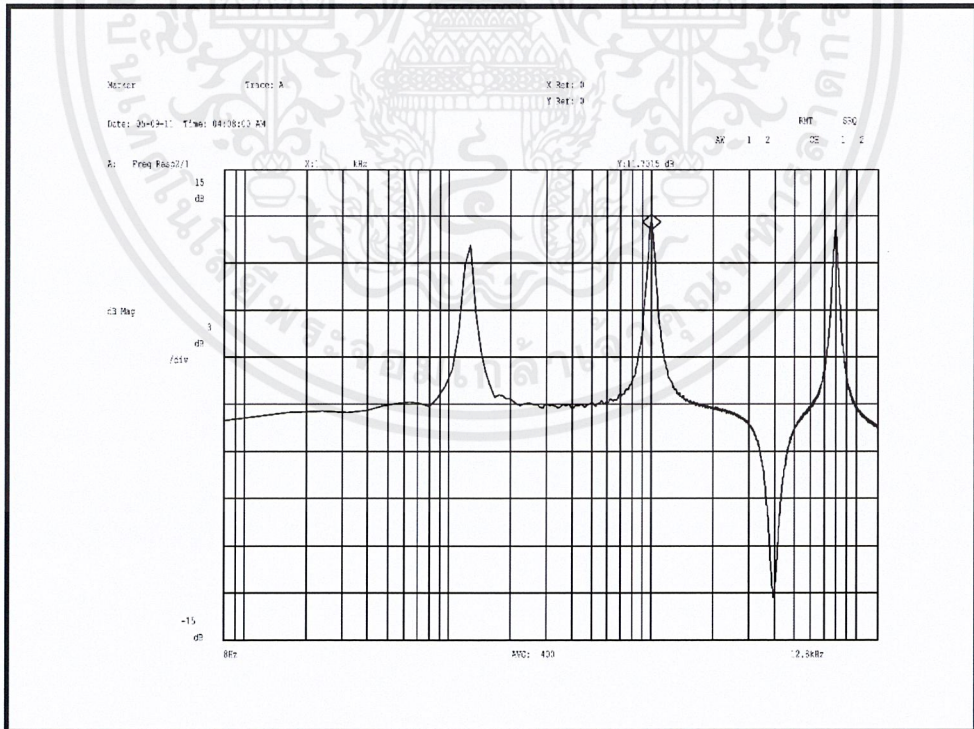
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ⁶²ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.26 รูปแสดงการทดลองที่ 5 มีค่า Gain แต่ละ Band คือ [4 3 -5 7 7 -7 -2] dB

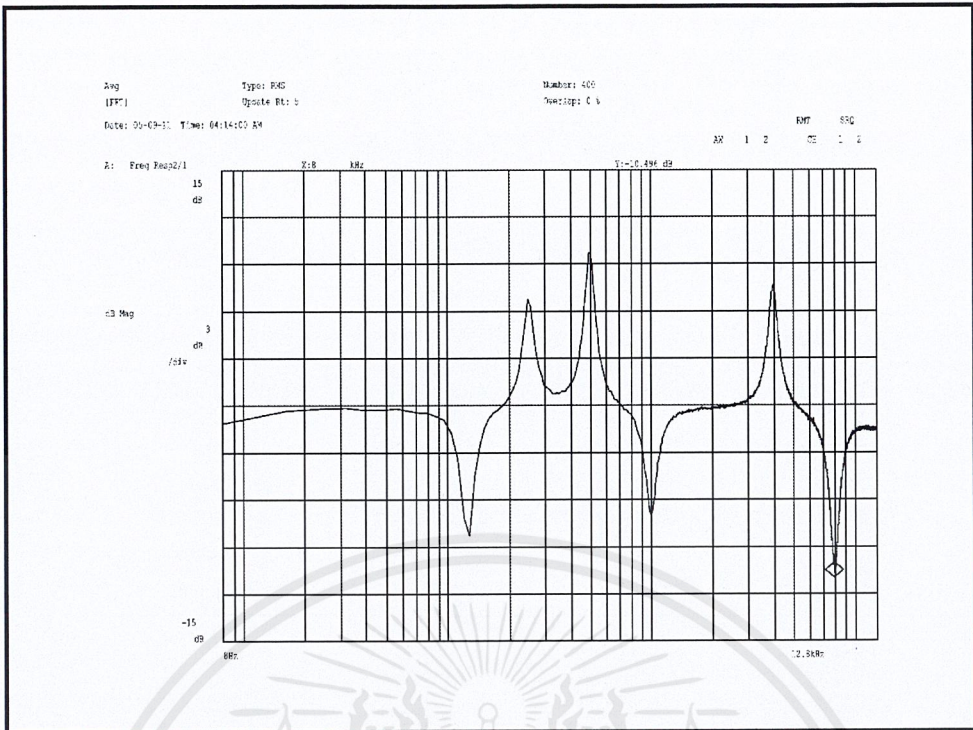
4.2.2 การแสดงผลลัพธ์ Magnitude response จากอุปกรณ์ Dynamic Signal

Analyzer

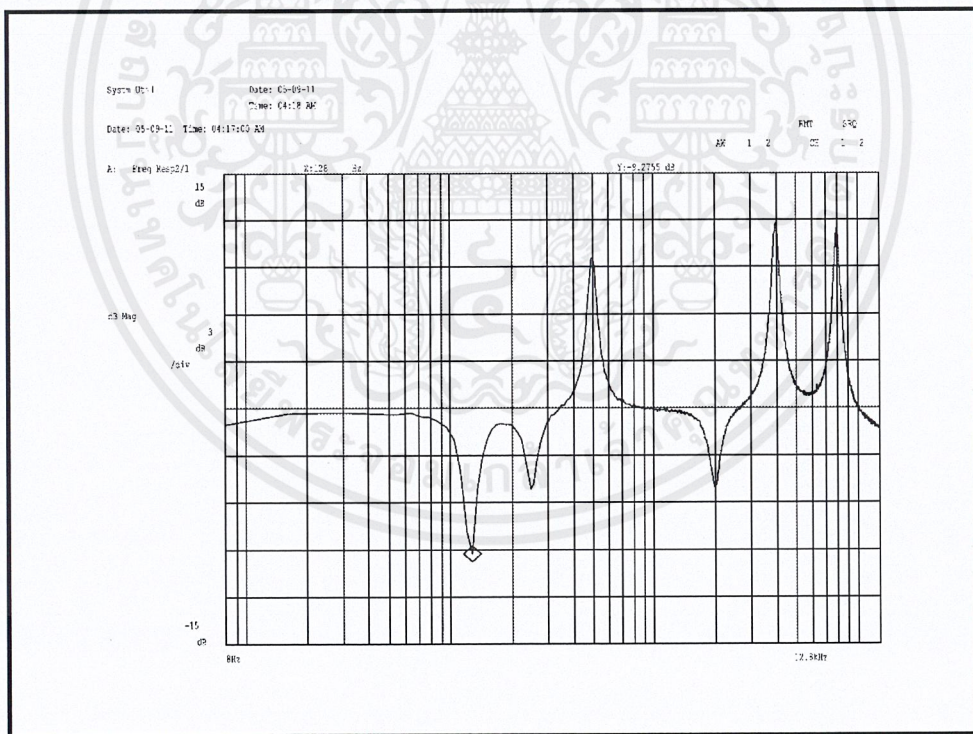


รูปที่ 4.27 รูปแสดง Magnitude response ที่ได้จากการทดลองที่

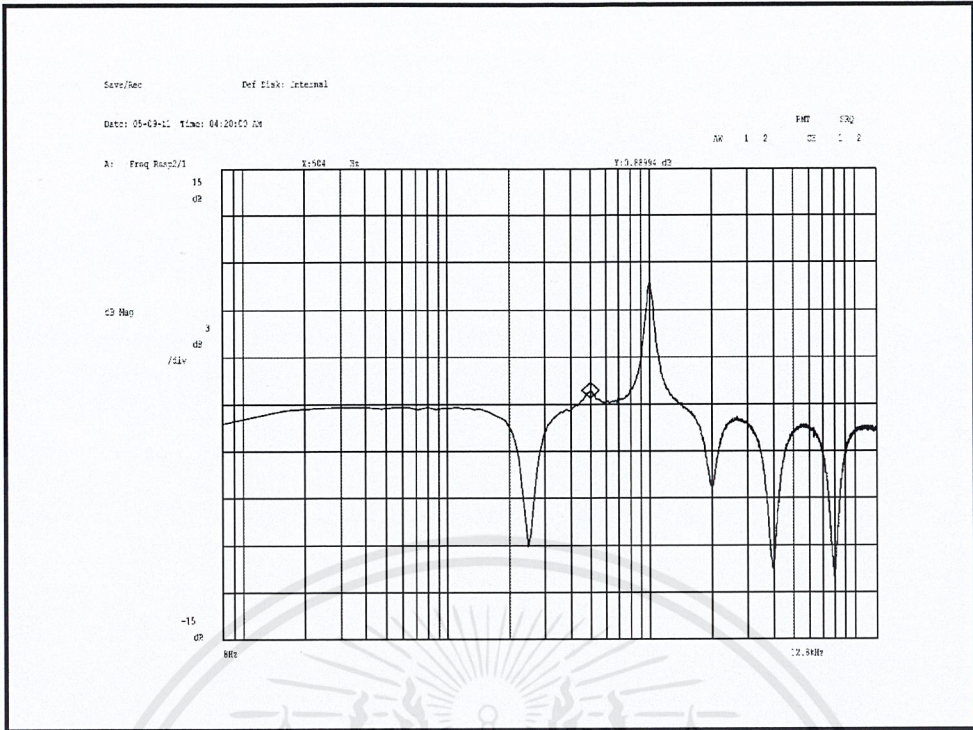
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ⁶³ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



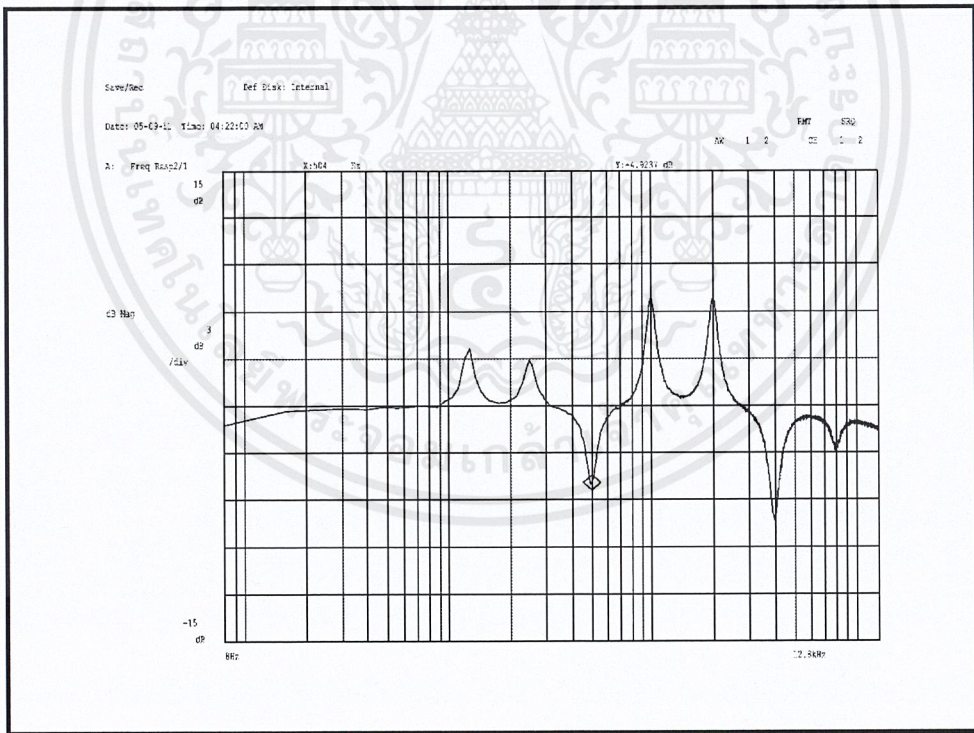
รูปที่ 4.28 รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 2



รูปที่ 4.29 รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 3



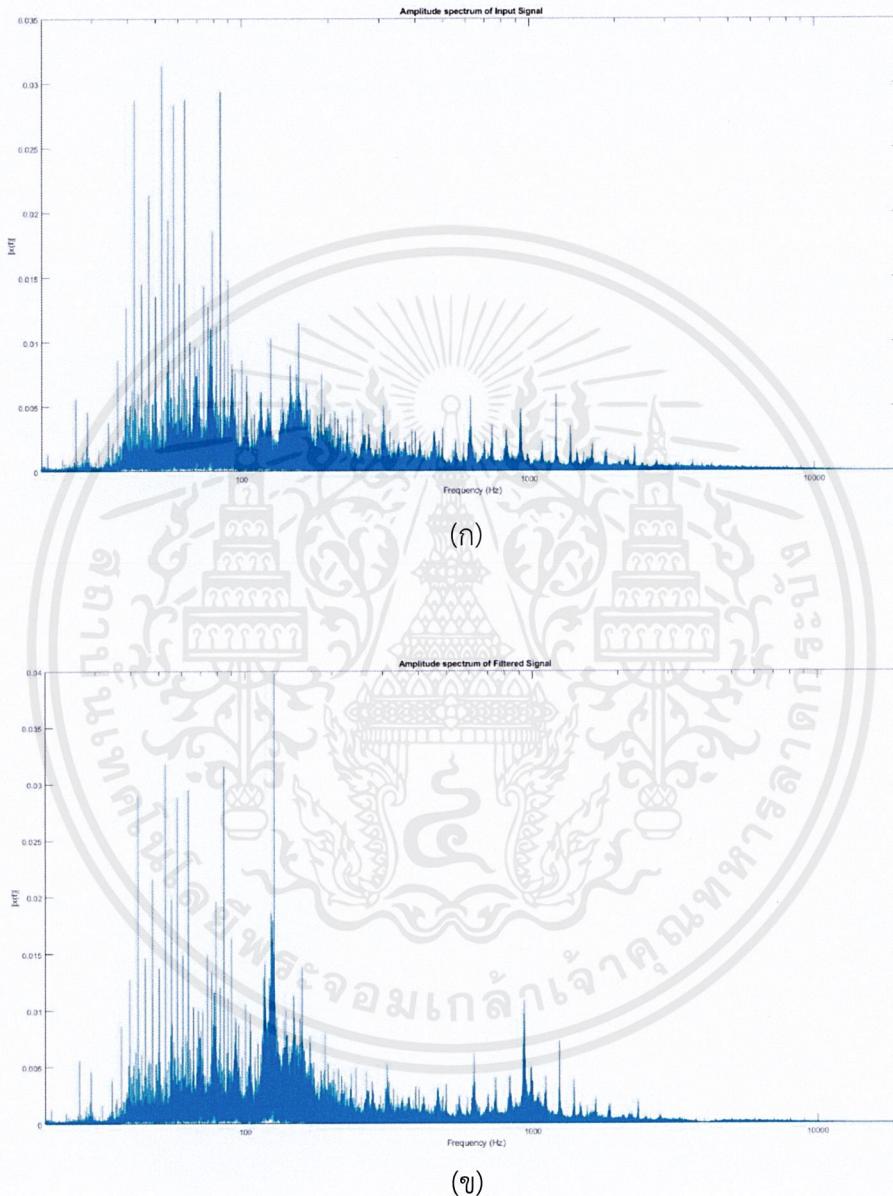
รูปที่ 4.30 รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 4



รูปที่ 4.31 รูปแสดง Magnitude response ที่ได้จากการทดลองที่ 5

การแสดงผลของ Magnitude Response ที่แสดงผลบนหน้าจอ User Interface และที่วัดได้จากอุปกรณ์ Dynamic Signal Analyzer ของการทดลองที่ 1 ถึง 5 มีความเหมือนกัน แสดงว่า Graphic Equalizer ที่ออกแบบสามารถทำงานได้อย่างถูกต้อง

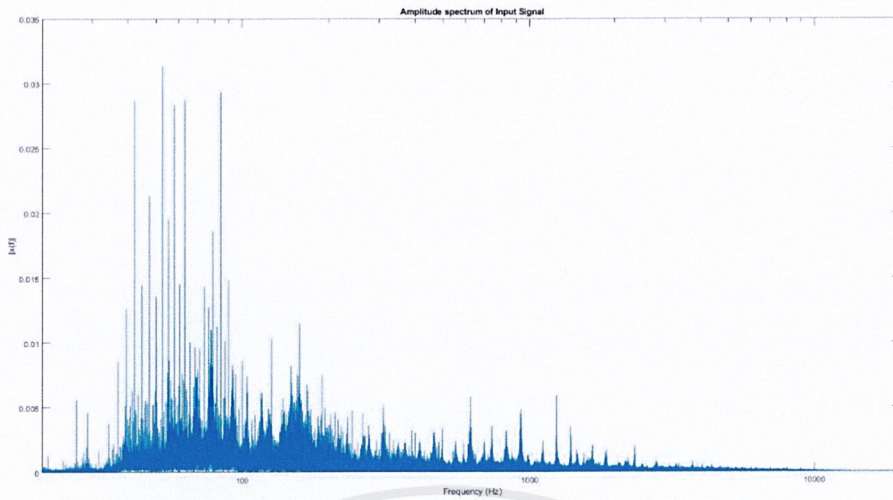
4.2.3 ค่า spectrum ก่อนและหลัง ทำการ EQ ในแต่ละการทดลอง



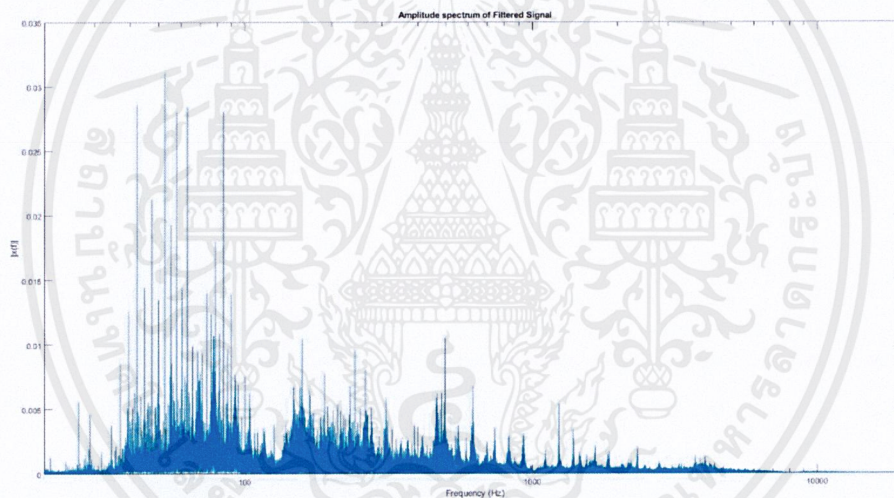
รูปที่ 4.32 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเพลง

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

Spectrum ของสัญญาณอินพุตเสียงเพลงหลังการประมวลผลในการทดลองครั้งที่ 1 มีลักษณะความถี่ในช่วงประมาณ 125 และ 1000 Hz เพิ่มขึ้นอย่างมากตรงตามการปรับ Graphic Equalizer ในการทดลองครั้งที่ 1



(ก)

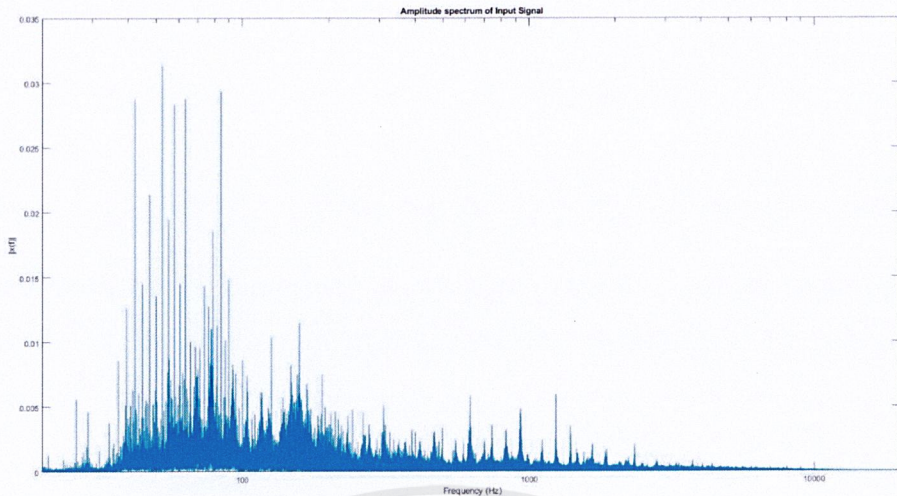


(ข)

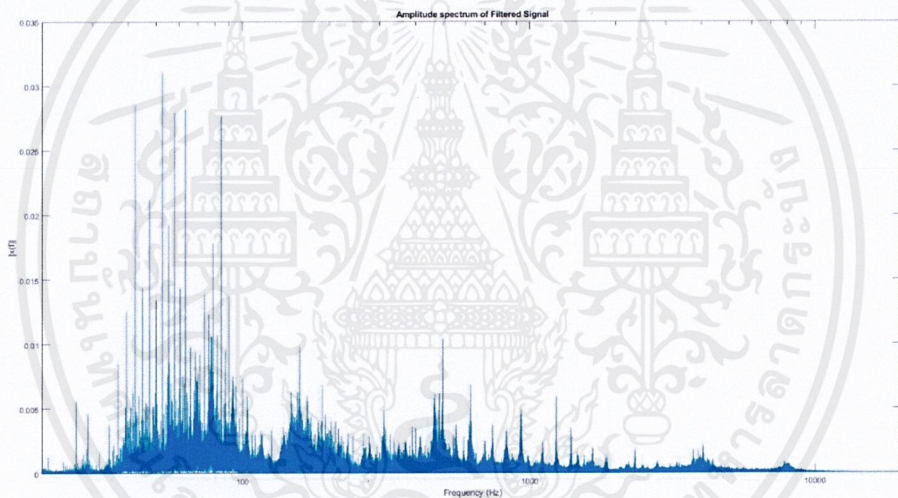
รูปที่ 4.33 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงเพลง

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

Spectrum ของสัญญาณอินพุตเสียงเพลงหลังการประมวลผลในการทดลองครั้งที่ 2 มีลักษณะความถี่ในช่วงประมาณ 125 ลดลง และความถี่ในช่วง 500 และ 4000 Hz เพิ่มขึ้นอย่างเห็นได้ชัดตรงตามการปรับ Graphic Equalizer ในการทดลองครั้งที่ 2



(ก)



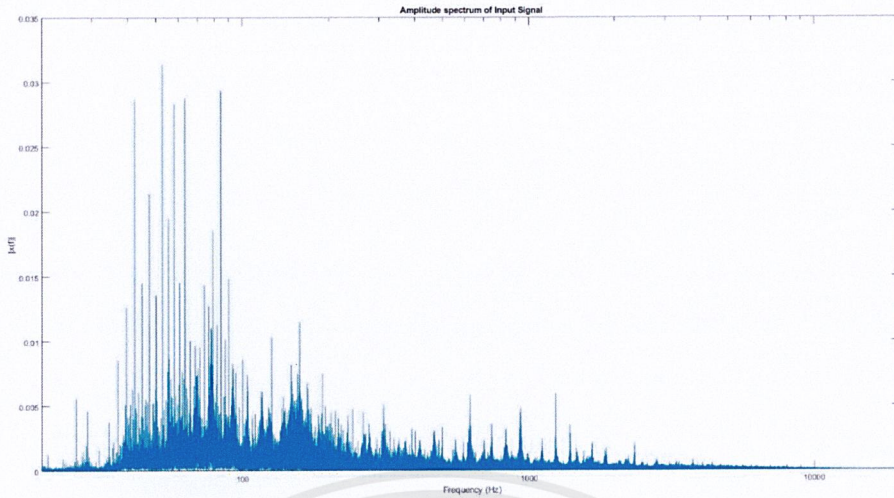
(ข)

รูปที่ 4.34 รูปแสดงSpectrumของสัญญาณอินพุตเสียงเพลง

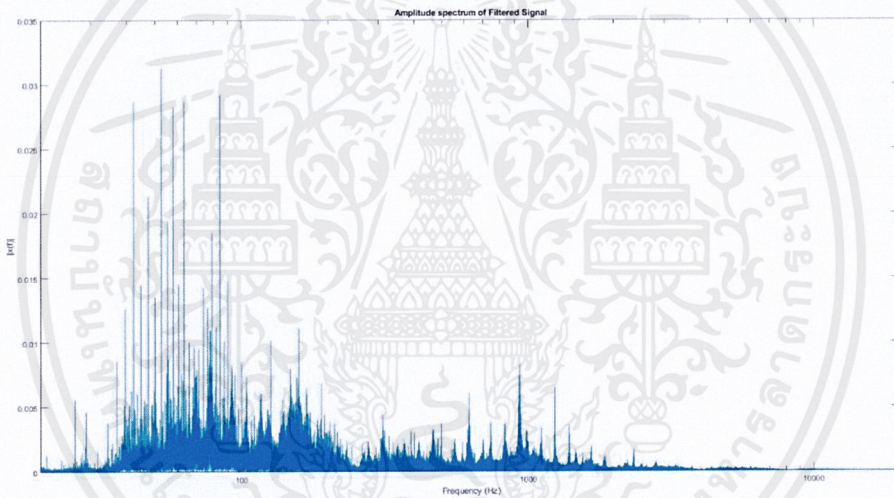
(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

Spectrum ของสัญญาณอินพุตเสียงเพลงหลังการประมวลผลในการทดลองครั้งที่ 3 มีลักษณะความถี่ในช่วงประมาณ 125 Hz ลดลง และความถี่ในช่วง 500, 4000 และ 8000 Hz เพิ่มขึ้นอย่างเห็นได้ชัดตรงตามการปรับ Graphic Equalizer ในการทดลองครั้งที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ 68 ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)

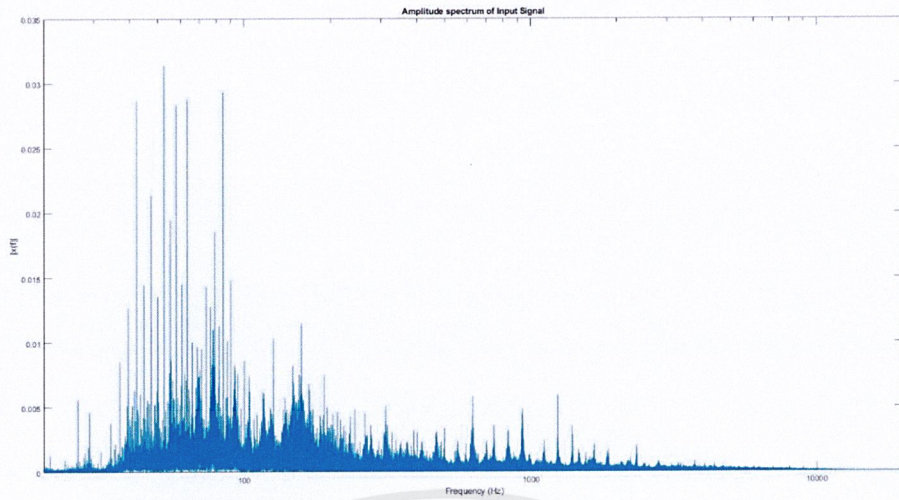


(ข)

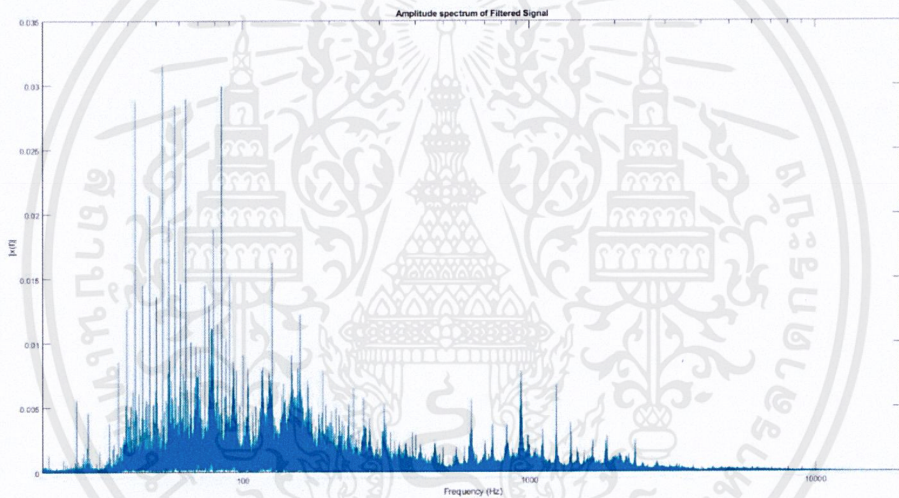
รูปที่ 4.35 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องหญิง

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

Spectrum ของสัญญาณอินพุตเสียงเพลงหลังการประมวลผลในการทดลองครั้งที่ 4 มีลักษณะความถี่ในช่วงประมาณ 1000 Hz เพิ่มขึ้นอย่างเห็นได้ชัดและมีในย่านอื่นๆ มีการลดลงอย่างเล็กน้อยตรงตามการปรับ Graphic Equalizer ในการทดลองครั้งที่ 4



(ก)



(ข)

รูปที่ 4.36 รูปแสดง Spectrum ของสัญญาณอินพุตเสียงร้องชาย

(ก) ก่อนการประมวลผล และ (ข) หลังผ่านการประมวลผล

Spectrum ของสัญญาณอินพุตเสียงเพลงหลังการประมวลผลในการทดลองครั้งที่ 5 มีลักษณะความถี่ในช่วงประมาณ 1000 และ 2000 Hz เพิ่มขึ้นเล็กน้อยตรงตามการปรับ Graphic Equalizer ในการทดลองครั้งที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ 70 ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

โครงการนี้เป็นกรอกแบบและสร้าง Digital Equalizer ซึ่งเป็นระบบประมวลผลสัญญาณ Digital ในรูปแบบของวงจรกรองสัญญาณ Digital ชนิดหนึ่งที่ใช้ในอุตสาหกรรมดนตรี ทั้งหมด 2 ประเภท ได้แก่ Digital Parametric Equalizer ที่มีการนำระบบตรรกศาสตร์คลุมเครือมาช่วยปรับ Parameters และ Digital Graphic Equalizer ที่ประมวลผลแบบ Real-time บนอุปกรณ์ FPGA

5.1.1 Digital Parametric Equalizer

ซอฟต์แวร์ Digital Parametric Equalizer ที่ออกแบบ มีระบบตรรกศาสตร์คลุมเครือมีความสามารถในการช่วยปรับค่า Parameters, Gain และ Quality Factor ได้อย่างมีประสิทธิภาพดังที่อนุญาตไว้ และสามารถระบุ Frequency Center ที่เหมาะสมกับสัญญาณเสียงนั้นๆได้

5.1.2 Digital Graphic Equalizer

ฮาร์ดแวร์ Digital Graphic Equalizer ที่ทำงานบนอุปกรณ์ FPGA สามารถประมวลผลแบบ Real-time ได้ โดยทำงานที่ Sampling Frequency = 44100 Hz และ Bit Depth = 12 Bits สามารถใช้ได้ทั้งหมด 7 ย่านความถี่ ซึ่งมี Center Frequency ดังนี้ 125, 250, 500, 1000, 2000, 4000 และ 8000 Hz สามารถปรับ Gain ได้ตั้งแต่ -12 ถึง 12 dB บน User Interface ที่แสดงบนหน้าจอและมีการแสดง Magnitude Response ที่กำลังปรับค่าอยู่ด้วย

5.2 ข้อเสนอแนะ

5.2.1 Digital Parametric Equalizer

5.2.1.1 ซอฟต์แวร์ที่ออกแบบทำงานอยู่บน MATLAB เท่านั้น จำเป็นต้องพัฒนาให้เป็น plug-ins เพื่อให้สามารถใช้นโปรแกรม Digital Audio Workstation ที่ถูกใช้งานในอุตสาหกรรมดนตรีได้

5.2.1.2 ระบบการเปิดฟังเพื่อตรวจสอบการทำงานของ Equalizer ที่ออกแบบไม่สามารถใช้งานในรูปแบบ Real-time ได้ จำเป็นต้องแก้ไขเมื่อนำไปต่อยอดให้สามารถใช้งานได้จริง

5.2.2 Digital Graphic Equalizer

5.2.2.1 ในส่วน Converter ที่ใช้แปลงสัญญาณอินพุตและเอาต์พุตของฮาร์ดแวร์ที่เลือกให้มีข้อจำกัดที่ Bit Depth สูงสุดที่ 12 Bits ซึ่งไม่เพียงพอในการนำไปใช้ในอุตสาหกรรมดนตรีที่ปัจจุบันมีมาตรฐานที่ 24 Bits หากต้องการนำฮาร์ดแวร์นี้ไปใช้งานจริงมีความจำเป็นต้องเปลี่ยนส่วนของ Converter ให้มีความละเอียดตามมาตรฐานของอุตสาหกรรม

5.2.2.2 Memory ของฮาร์ดแวร์ไม่เพียงพอที่จะใช้แสดง Magnitude Response ของ Equalizer ในย่านความถี่ที่ต่ำกว่า 500 Hz สามารถเลือกใช้ฮาร์ดแวร์ที่มี Memory สูงกว่านี้เพื่อแสดงผล Magnitude Response ในย่านความถี่ต่ำให้ถูกต้องแม่นยำกว่านี้ได้ แต่ปัญหานี้เป็นเพียงส่วนแสดงผลเท่านั้น ไม่มีความเกี่ยวข้องโดยตรงกับส่วนประมวลผลสัญญาณ

5.2.2.3 การปรับค่า Parameters ของ Equalizer สามารถทำได้ด้วยการปรับค่า Gain บน User Interface ที่แสดงบนหน้าจอคอมพิวเตอร์ ทำให้ในการใช้งานจำเป็นต้องเชื่อมต่อกับคอมพิวเตอร์เพื่อควบคุม สามารถปรับเปลี่ยนไปใช้ Potentiometer มาทดแทนการใช้ User Interface บนคอมพิวเตอร์เพื่อให้สามารถทำงานในอุปกรณ์ฮาร์ดแวร์ชิ้นเดียวได้



บรรณานุกรม

- [1] Julius O. Smith III. *“Introduction to Digital Filters with Audio Application.”* Center of Computer Research in Music and Acoustics (CCRMA), Stanford University, 2019.
- [2] J.-S. Roger Jang, Ned Gully. *“MATLAB: Fuzzy Logic Toolbox User’s Guide.”* The MathWorks, Inc., 1997.
- [3] Michael Negnevitsky, *“Artificial Intelligence: A Guide to Intelligent System.”* 2nd ed. England, Pearson Education, 2005.
- [4] Udo Zolzer, *“Digital Audio Signal Processing.”* 2nd ed. Stuttgart: John Wiley & Son, 2008.
- [5] Vesa Välimäki, Joshua D. Reiss. *“All About Audio Equalization: Solutions and Frontiers.”* Department of Signal Processing and Acoustics, Aalto University, 2016.
- [6] Apogee Technology, Inc. *“Filter Design Equations.”*
- [7] James M. Montgomery. *“Implementing of 10-Band Stereo Equalizer on the DSP56311 EVM Board.”* Freescale Semiconductor, 2005.
- [8] Jonas Ekeroot. *“Implementing a Parametric EQ plug-in in C++ using the Multi-Platform VST Specification.”* Division of Sound Recording, Lulea University of Technology, 2003.
- [9] Dennis Rane. *“Constant-Q Graphic Equalizers.”* <https://bit.ly/2VNl9xC>.
- [10] Mark Wickert. *“How to Characterize the Peaking Filter for an Audio Graphic Equalizer.”* <https://bit.ly/30rfdyJ>.
- [11] *“Interactive Instrument Frequency Chart.”* <https://bit.ly/1uuZUik>.