ภาควิชาวิศวกรรมไฟฟ้า

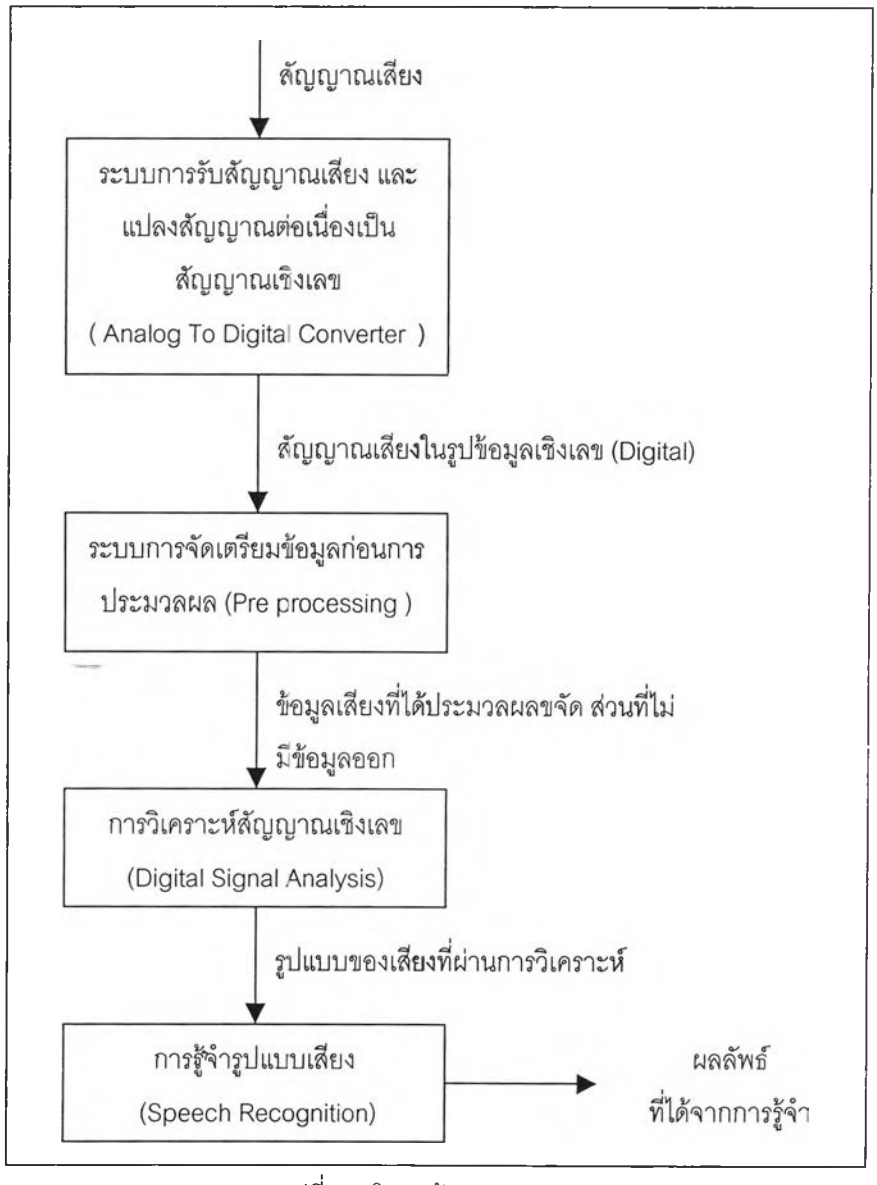
บทที่ 3

ระบบวิเคราะห์และรู้จำเสียงพูด

ในงานวิจัยฉบับนี้ ได้อาศัยโครงสร้างระบบวิเคราะห์และรู้จำเสียงพูด ที่มีการใช้โดยทั่วไป โดยกำหนดเป็น 4 ขั้นตอนคือ ขั้นตอนการรับสัญญาณเสียง หรือการแปลงสัญญาณต่อเนื่องเป็นสัญญาณเชิงเลข (Analog to Digital Converter : ADC) ขั้นตอนการจัดเตรียมข้อมูลก่อนการประมวลผล (Pre processing) ขั้นตอนการวิเคราะห์สัญญาณเชิงเลข (Digital Signal Analysis) และขั้นตอนการรู้จำรูปแบบเสียง (Speech Recognition) แต่ในรายละเอียดของแต่ละขั้นตอน ก็จะมีการกำหนดที่แตกต่างกันออกไป เพื่อให้ได้รูปแบบที่เหมาะสม

3.1 โครงสร้างระบบวิเคราะห์และรู้จำเสียงพูด

เราสามารถกำหนดโครงสร้างระบบได้ดังรูป 3.1 ซึ่งแสดงให้เห็นองค์ประกอบ และการเชื่อมต่อระหว่างส่วนต่างๆ ในการทำงาน



รูปที่ 3.1 โครงสร้างระบบ

I 20325137

3.2 การทำงานของระบบวิเคราะห์และรู้จำเสียงพูด

การทำงานของส่วนต่างๆ จะมีรายละเอียด และข้อกำหนดต่างๆ เพื่อให้การทำงานเป็นไป อย่างเหมาะสม และให้ได้ผลลัพธ์ ส่งต่อไปยังส่วนอื่น เพื่อให้การทำงานเป็นไปอย่างถูกต้องและได้ผลลัพธ์ที่ถูกต้องที่สุด ซึ่งเราสามารถสรุปการทำงานของส่วนต่างๆ ได้ดังนี้

3.2.1 ระบบการรับสัญญาณเสียง และแปลงสัญญาณต่อเนื่องเป็นสัญญาณเชิงเลข (Analog To Digital Converter)

เนื่องจากในวิทยานิพนธ์ฉบับนี้ เป็นการพัฒนางานวิจัยบนเครื่อง คอมพิวเตอร์ ที่ทำงานภายใต้ระบบปฏิบัติการ WINDOWS 98 ดังนั้นการรับสัญญาณเสียงจึงใช้อุปกรณ์มาตราฐานของเครื่องคอมพิวเตอร์คือ ไมล์ (Microphone) ต่อกับการ์ดเสียง (Sound Card) และในการพัฒนาโปรแกรมก็ใช้คำสั่งของ ระบบปฏิบัติการ Windows 98 (Windows API) ในการนำเสียงที่เป็นสัญญาณ ต่อเนื่อง เข้าสู่การประมวลผลเชิงเลข (Digital)

เมื่อสัญญาณเสียงถูกสร้างขึ้นจากการพูด โดยจะอย่างในรูปของพลังคลื่นเสียง เราจึงใช้ ไมล์ (Micro phone) เป็นอุปกรณ์ในการเปลี่ยนพลังงานเสียงให้เป็นพลังงานไฟฟ้า โดยพลังงานไฟฟ้าที่ได้ก็จะเปลี่ยนแปลงตามระดับและความถี่ของเสียงพูด แต่ก็ยังอยู่ในรูปของสัญญาณต่อเนื่องเช่นเดิม เมื่อเราต้องการนำสัญญาณดังกล่าวมาประมวลผลในระบบคอมพิวเตอร์ เราต้องมีการเปลี่ยนแปลงรูปแบบสัญญาณโดยอาศัยอุปกรณ์ที่ เรียกว่า วงจรแปลงสัญญาณต่อเนื่องเป็นสัญญาณเชิงเลข (Analog To Digital Converter) ซึ่งในส่วนนี้ปัจจัยสำคัญที่จำเป็นต้องพิจารณาคือ ความเร็วในการดึงข้อมูลหรือสุ่มข้อมูล (Sampling Rate) และความละเอียดของข้อมูลที่ดึง (Data Sampling) โดยในส่วนของความเร็วในการดึงข้อมูล ก็อาศัยหลักการที่ว่า ในการสุ่มว่าความถี่ในการสุ่มควรเป็นอย่างน้อย 2 เท่าของความถี่สูงสุดที่เราต้องการให้ระบบสามารถตอบสนองได้ โดยในที่นี้ความถี่สูงสุดที่เราต้องการให้ระบบตอบสนองคือ 4,000 Hz แต่ก็มีข้อจำกัดในส่วนของอุปกรณ์ การ์ดเสียง เนื่องสามารถการ์ดเสียงถูกกำหนดให้มีอัตราการสุ่มต่ำสุดที่ 11,024 Hz และที่อัตราการสุ่มดังกล่าวก็สามารถตอบสนองความถี่ที่เราต้องการได้ดังนั้นจึงเลือกใช้อัตราสุ่มที่ 11 KHz และในส่วนของความละเอียดของข้อมูลที่สุ่มก็มีอยู่ 2 ระดับคือ ระดับ 8 หลัก (Bit) และระดับ 16 หลัก (Bit) โดยความละเอียดของข้อมูล จะมีผลต่อความถูกต้องในการ เปลี่ยนแปลงของข้อมูล ดังนั้นจึงเลือกใช้ความละเอียดระดับ 16 Bit เพื่อความถูกต้องและใกล้เคียงกับเสียงจริงๆ

3.2.2 ระบบการจัดเตรียมข้อมูลก่อนการประมวลผล (Pre processing)

เมื่อเราสามารถนำสัญญาณเสียง เข้าสู่ระบบและจัดให้อยู่ในรูปแบบของข้อมูลเชิงเลข (Digital) แล้วเราก็สามารถนำข้อมูลไปคำนวณได้ แต่สัญญาณที่รับเข้ามาเป็นข้อมูลเสียงพูดที่ไม่มีการควบคุมดังนั้นก่อนจะสามารถนำไปประมวลผล จึงจำเป็นต้องมีการจัดข้อมูลให้ถูกต้องและสมบูรณ์ ก่อนโดยจะมีการเตรียมการ 2 เรื่องคือ

3.2.2.1 การตรวจหาจุดเริ่มต้นเสียง

เนื่องจากระบบที่พัฒนาเป็นระบบทำงานตลอดเวลา (On Line) และการประมวลผลแบบเวลาจริง (Real Time) ดังนั้น ระบบจะรอรับสัญญาณเสียงอยู่ตลอดเวลา ดังนั้นระบบต้องสามารถแยกความแตกต่างระหว่างความเงียบ สัญญาณรบกวน และสัญญาณเสียง ได้ ซึ่งในงานวิจัยนี้ใช้หลักการคำนวณด้านกำลัง คือ ถ้ามีพลังไฟฟ้า ที่ถูกเปลี่ยนจากสัญญาณเสียง ในช่วงเวลาหนึ่งมากกว่าระดับอ้างอิงก็ให้ถือว่าเป็นจุดเริ่มต้นของสัญญาณเสียง และให้ระบบเริ่มบันทึกเสียงเป็นความยาว 1 วินาที (เนื่องจากเสียงที่ต้องการรู้จำเป็นคำที่มีความยาวไม่เกิน 1 วินาที)

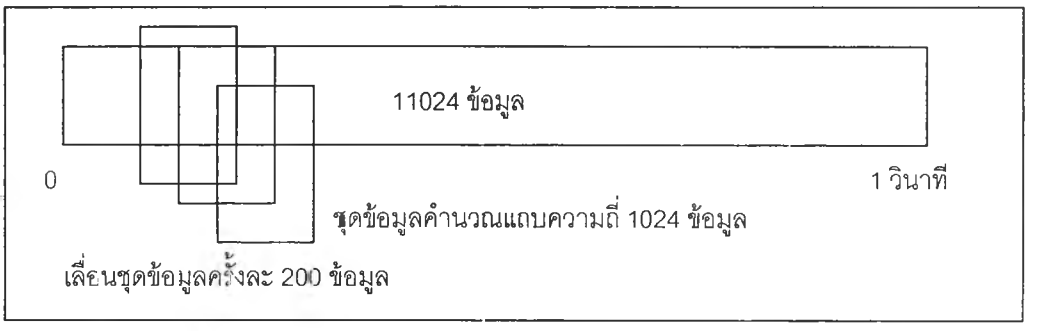
3.2.2.2 การจัดระดับเสียง

เสียงที่เข้าสู่ระบบเป็นสัญญาณเสียงที่ไม่ได้ถูกควบคุมดังนั้นอาจมีระดับที่ไม่คงที่ดังนั้นจึงมีการจัดระดับสัญญาณเสียงที่เข้าในแต่ละเสียงพูดให้สม่ำเสมอ โดยการหาค่าที่ดังที่สุดชุดข้อมูล และหาอัตราส่วนที่เหมาะสมในการคำนวณแถบความถี่ จากนั้นทำการปรับระดับของข้อมูลทั้งหมดในอัตราส่วนเดียวกัน

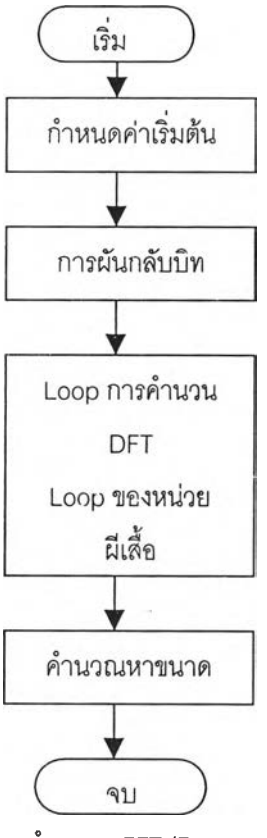
3.2.3 การวิเคราะห์สัญญาณเชิงเลข (Digital Signal Analysis)

ในการวิเคราะห์สัญญาณเชิงเลข (Digital Signal Analysis) เป็นการอาศัยการคำนวณทางคณิตศาสตร์ ซึ่งเป็นวิชาที่ว่าด้วยการประมวลผลสัญญาณเชิงเลข (Digital Signal Processing) และในการคำนวณก็มีหลาย วิธีที่สามารถคำนวณเพื่อหาคุณลักษณะของสัญญาณได้ และจากความรู้พื้นฐานที่เรารู้ว่าเสียงมีองค์ประกอบด้านความถี่ที่เป็นลักษณะเฉพาะของเสียงต่างๆ ดังนั้น เราจึงใช้การคำนวณที่สามารถแสดงให้เห็นถึงองค์ประกอบด้านความถี่ หรือแถบความถี่ (Spectrum) และการคำนวณโดยใช้ อนุกรมฟูริเยอร์ (Fourier serial) และสมการ ดิสครีทฟูริเยอร์ ทรานสฟอร์ม (Discrete Fourier Transform) ซึ่งเป็นสมาการที่ใช้ในการคำนวณ หาแถบความถี่ของสัญญาณในช่วงเวลาหนึ่งๆ แต่ในทางปฏิบัติ การคำนวณ ดิสครีทฟูริเยอร์ ทรานสฟอร์ม เป็นการคำนวณที่ใช้เวลามาก จึงมีการพัฒนาเทคนิคการลดทอนด้านเวลา โดยใช้ การคำนวนแบบหน่วยผีเสื้อ (Butterfly) เข้ามาช่วย เพื่อให้การคำนวณแถบความถี่ใช้เวลาน้อยลง เราเรียกการคำนวณแบบนี้ว่า ฟาร์สฟูเรีย ทรานสฟอร์ม (Fast Fourier Transform : FFT) ซึ่งมีหลักการตามโฟลว์ชาร์ต ดังรูปที่ 3.3

ในการคำนวณแถบความถี่จากชุดข้อมูลสัญญาณเสียงที่รับเข้ามาเป็นเวลา 1 วินาที ซึ่งจะมีข้อมูลประมาณ 11024 ข้อมูล และให้ให้การคำนวนแถบความถี่รวดเร็วขึ้นจึงมีการลดขนาดของชุดข้อมูลในการคำนวนเป็นชุดย่อย 1024 ข้อมูลต่อชุด และทำการเลื่อนชุดข้อมูลแบบคาบเกี่ยวกัน โดยเลื่อนข้อมูลครั้งละ 200 ข้อมูลเพื่อขจัดความไม่ต่อเนื่องของการคำนวณแถบความถี่ และทำการคำนวณ 50 รอบ นำผลของแถบความถี่ทั้ง 50 รอบมารวมแบบ ทับซ้อนกัน ก็จะทำให้เราได้แถบความถี่ของสัญญาณเสียงที่บันทึกเข้ามาในช่วง 1 วินาที



รูปที่ 3.2 การจัดชุดข้อมูลเพื่อคำนวณ FFT (Fast Fourier Transform)



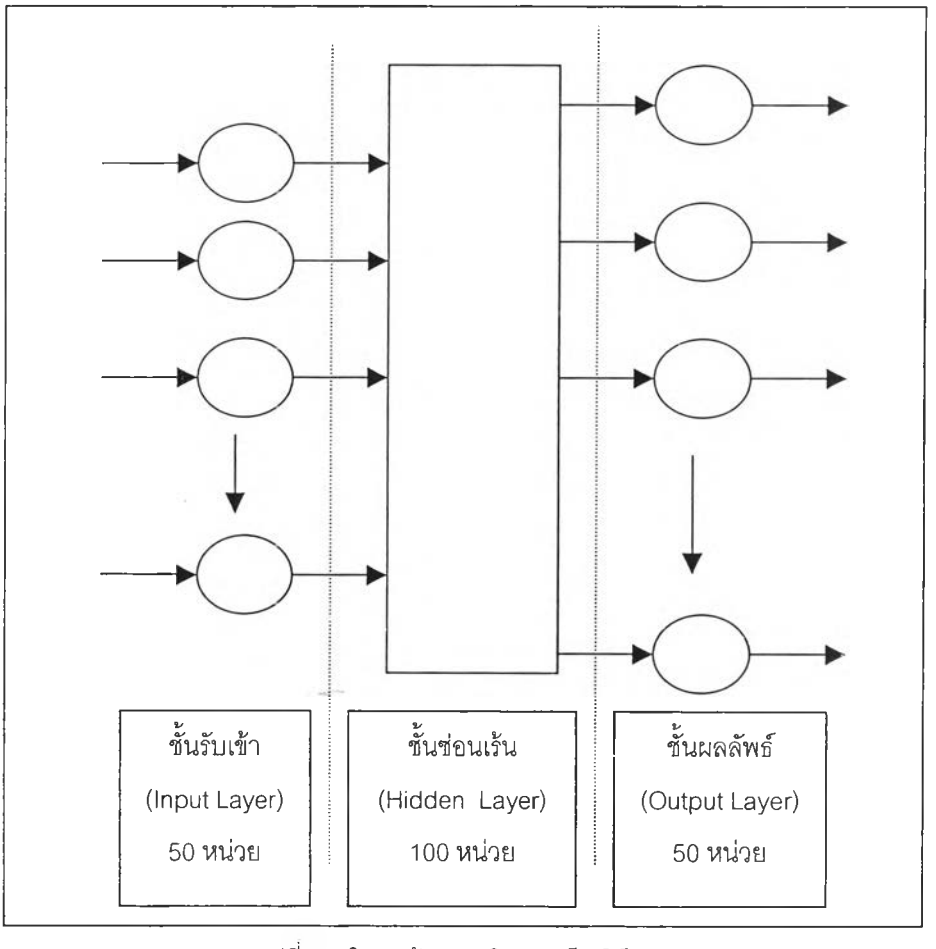
รูปที่ 3.3 การคำนวณ FFT (Fast Fourier Transform)

3.2.4 การรู้จำรูปแบบเสียง (Speech Recognition)

เมื่อเราสามารถคำนวณเพื่อแสดงคุณลักษณะของสัญญาณเสียงได้แล้ว ในขั้นต่อไปคือ การเปรียบเทียบลักษณะของเสียงนั้นว่าเป็นเสียงใด ซึ่งเราต้องมีต้นแบบของเสียงต่างๆ เพื่อใช้ในการเปรียบเทียบ และในการเปรียบเทียบข้อมูลไม่ใช้การเปรียบเทียบแบบ หนึ่งต่อหนึ่ง แต่เป็นการเปรียบเทียบชุดข้อมูล ดังนั้นในการประมวลผลจึงต้องใช้ขั้นตอนที่ซับซ้อนกว่า โดยใช้ โครงข่ายประสาทเทียม หรือ นิวรอลเน็ตเวิร์ก (Artificial Neural network) เข้ามาช่วยในการคำนวณเปรียบเทียบชุดข้อมูล

โครงสร้างของนิวรอลเน็ตเวิร์ก ประกอบด้วยชุดนิวรอล หรือ เพอร์เซ็บตรอน (Perceptron) ที่เชื่อมต่อกันเป็นชั้นๆ เรียกว่า นิวรอลเน็ตเวิร์ก (Multi layer neural networks) โดยเรากำหนด นิวรอลเน็ตเวิร์ก เป็น 3 ชั้น ประกอบด้วย ชั้นข้อมูลเข้า (Input Layer) ชั้นซ้อนเร้น (Hidden Layer) และชั้นข้อมูลออก หรือผลลัพธ์ (Output

Layer) และในการกำหนดขนาดของ เน็ตเวิร์ก (Networks) ของแต่ละชั้นก็มีข้อกำหนดคือ ในส่วนของชั้นข้อมูลเข้า ขึ้นอยู่กับจำนวนหลักของข้อมูลที่ต้องการป้อนเข้าสู่โครงข่ายประสาท ส่วนชั้นข้อมูลออก (Output Layers) ก็ขึ้นอยู่กับจำนวน ผลลัพธ์ที่ต้องการให้ระบบตอบสนอง หรือรู้จำ (Recognize) และในส่วนของชั้นซ้อนเร้น จะกำหนดจำนวนนิวรอล ตามสามารถของเน็ตเวิร์ก และความเร็วในการเรียนรู้ (Leaning) และการรู้จำ (Recognize) โดยจำนวนของนิวรอล จะมีผลต่อค่าผิดพลาด (Error) ของระบบด้วย ซึ่งก็ต้องอาศัยการทดลองหาค่าที่เหมาะสม



รูปที่ 3.4 โครงสร้างของนิวรอลเน็ตเวิร์ก

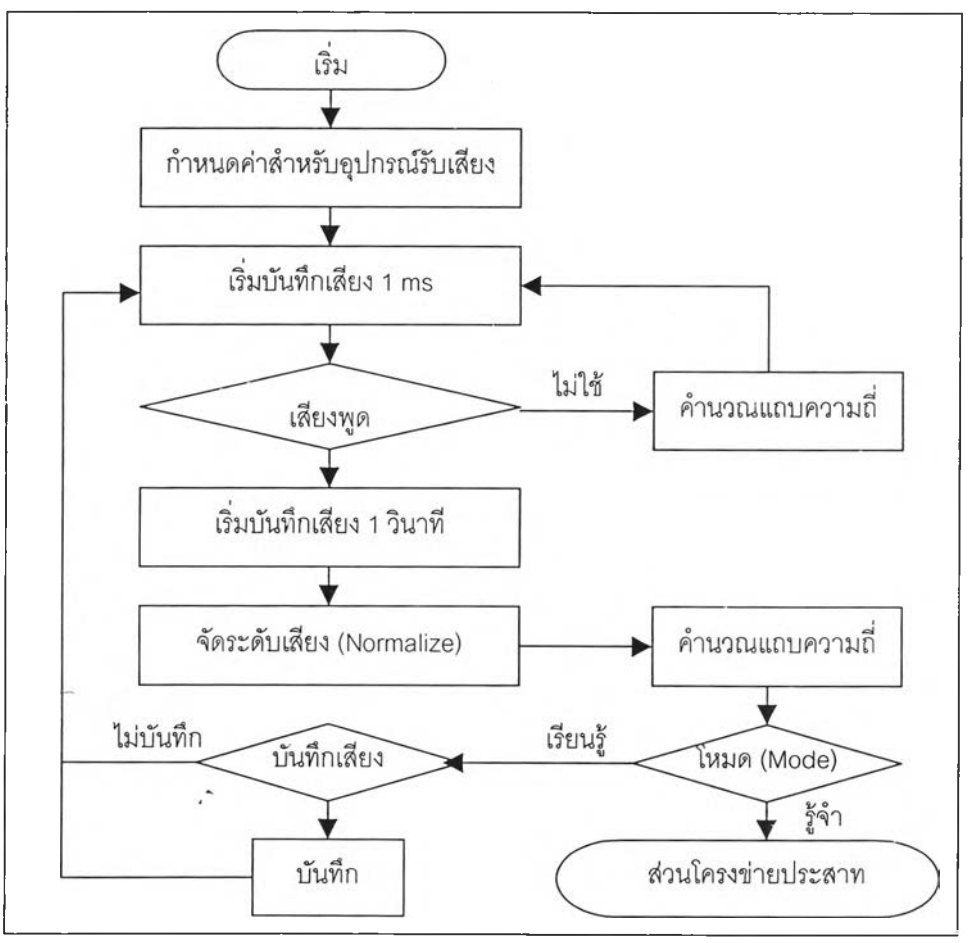
3.3 โปรแกรมวิเคราะห์และรู้จำเสียงพูด

ในการวิจัย ได้สร้างระบบเพื่อทำงานตามหลักการที่ได้ศึกษามา โดยใช้โปรแกรมภาษา วิชวล เบสิก รุ่น 6 (Visual Basic V. 6.0) เป็นภาษาในการพัฒนาโปรแกรม โดยแบ่งโปรแกรมออกเป็นส่วนๆ ตามหน้าที่การทำงานดังนี้คือ

- โปรแกรมส่วนรับเสียง และการประมวลผลขั้นต้น
- โปรแกรมส่วนวิเคราะห์แถบความถี่ (Fast fourier transform : FFT)
- โปรแกรมส่วนของนิวรอลเน็ตเวิร์ก (Neural Networks)
- โปรแกรมส่วนควบคุมโปรแกรม

3.3.1 โปรแกรมส่วนรับเสียง และการประมวลผลขั้นต้น

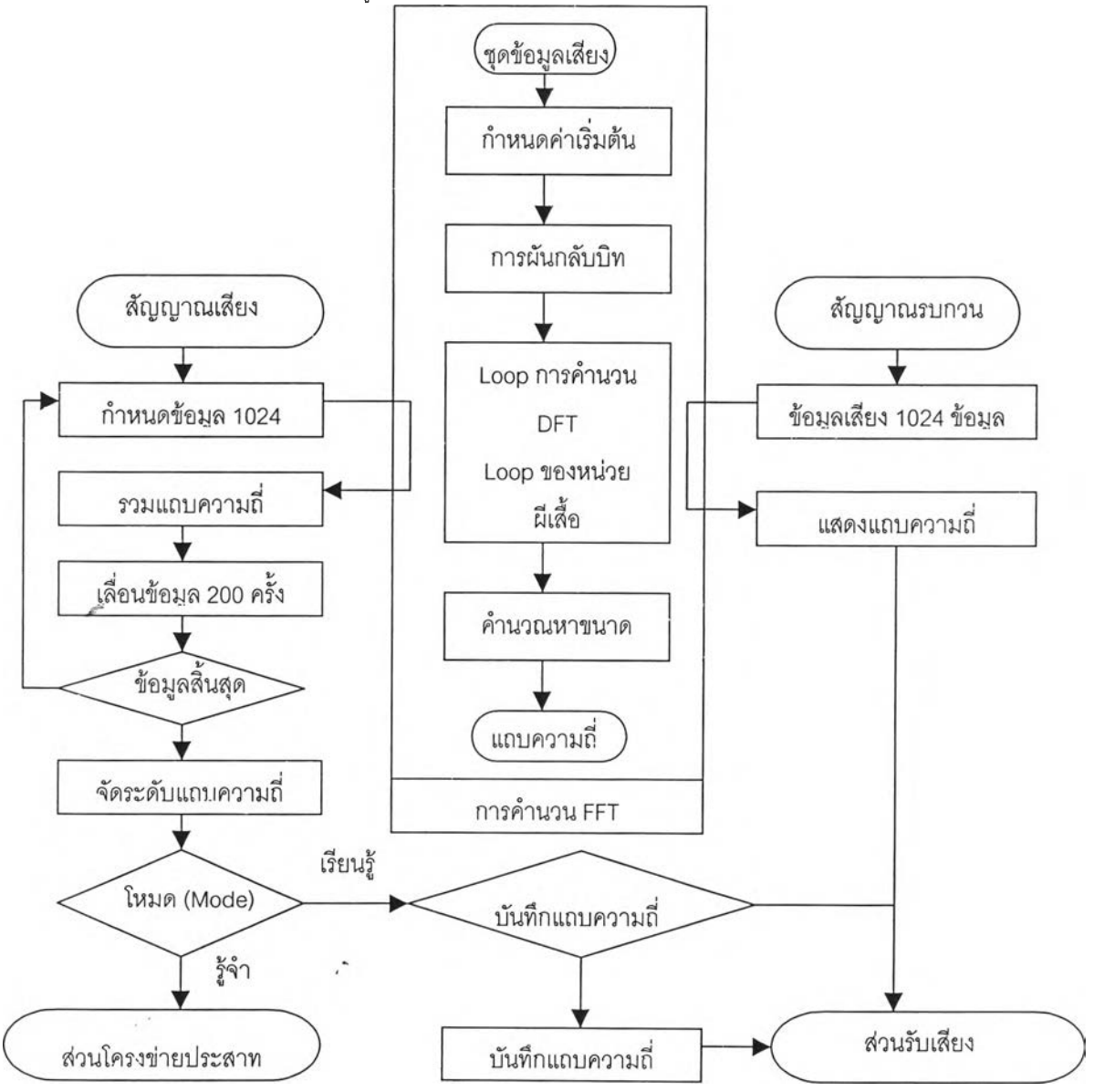
ส่วนนี้ทำหน้าที่กำหนดค่าของอุปกรณ์ด้านเสียงของระบบปฏิบัติการ กำหนดหน่วยความจำในการเก็บข้อมูลเสียง และเป็นอุปกรณ์รับเสียง เพื่อเริ่มบันทึกเสียง และปรับระดับเสียงให้อยู่ในระดับปกติ (Normalize) โดยจะบันทึกเสียงครั้งละ 1 ส่วน 10 วินาที (1 ms) นอกจากนั้นยังเป็นส่วนที่ประมวลผลเพื่อตรวจสอบว่าข้อมูลเสียงที่รับเข้ามาเป็นเสียงพูด หรือสภาวะเงียบ หรือสัญญาณรบกวน และถ้าคำนวณแล้ว ปรากฏว่าเป็นเสียงพูด โปรแกรมก็จะสร้างหน่วยความจำขึ้นมาอีกชุด ซึ่งสามารถบันทึกเสียงได้ประมาณ 1 วินาที โดยค่าประมาณ 20,480 ตำแหน่ง แต่ในความเป็นจริงหน่วยความจำนี้จะถูกแบ่งออกเป็น 2 ส่วนเพื่อบันทึกเสียงแบบ สเตริโอ (Stereo) แต่ในโปรแกรมวิเคราะห์และรู้จำเสียงพูดใช้การบันทึกเสียงแบบ โมโน (Mono) แต่ก็ต้องกำหนดหน่วยความจำทิ้งไว้ และระบบปฏิบัติการจะเป็นผู้จัดการหน่วยความจำเอง โดยสามารถแสดงขั้นตอนการทำงานดังรูปที่ 3.5



รูปที่ 3.5 ขั้นตอนการทำงานของส่วนรับเสียง และการประมวลผลขั้นต้น

3.3.2 โปรแกรมส่วนวิเคราะห์แถบความถี่ (Fast Fourier Transform : FFT)

ในส่วนนี้เป็นส่วนที่ต้องรับข้อมูลเสียงที่อยู่ในรู้แบบสัญญาณเชิงเลข ที่ผ่านมาจากส่วนรับเสียง และการประมวลผลขั้นต้น เนื่องจากการคำนวณ ฟูเรีย เป็นการคำนวณแบบเป็นช่วงๆ ดังนั้นจำจำเป็นต้องมีการกำหนดขนาดของชุดข้อมูล โดยในโปรแกรมกำหนดให้เป็น 1024 ซึ่งเป็นค่าที่ใช้เวลาน้อย และเหมาะสมกับระยะเวลารับเสียงเพื่อการวิเคราะห์ โดยจะทำให้โปรแกรมสามารถคำนวณและแสดงผลแบบเวลาจริงได้มากที่สุด (Real Time) ซึ่งถ้าการตรวจสอบพบว่าเป็นเสียงพูด จะทำให้มีข้อมูลที่ต้องคำนวนหาแถบความถี่มากขึ้นคือประมาณ 10,240 ข้อมูล ทำให้ต้องแบ่งการคำนวนเป็นแฟรม (Frame) โดยใช้แฟรมขนาด 1024 แต่การคำนวนแถบความถี่ในลักษณะหลายๆ แฟรมต่อเนื่องกัน จะทำให้มีข้อมูลบ้างอย่างสูญหายไป จึงทำการคำนวณแบบ แฟรมคร่อมแฟรม โดยจะให้แฟรมขนาด 1024 เลื่อนไปครั้งละ 200 ข้อมูล ซึ่งจะสามารถลดการสูญหายของแถบความถี่ได้ และไม่ใช้เวลานานมากนักในการคำนวณ ดังแสดงในรูป 3.2 ซึ่งก็จะได้จำนวนแฟรมของแถบความถี่มา 50 แฟรมเช่นกัน จากนั้นทำการรวมแบบทับซ้อนกัน ทั้ง 50 แฟรมก็จะได้ผลรวมของแถบความถี่ โดยขั้นตอนการคำนวณ FFT สามารถแสดงได้ดังรูป 3.6



รูปที่ 3.6 แสดงขั้นตอนการทำงานของส่วนวิเคราะห์แถบความถี่ (Fast Fourier Transform : FFT)

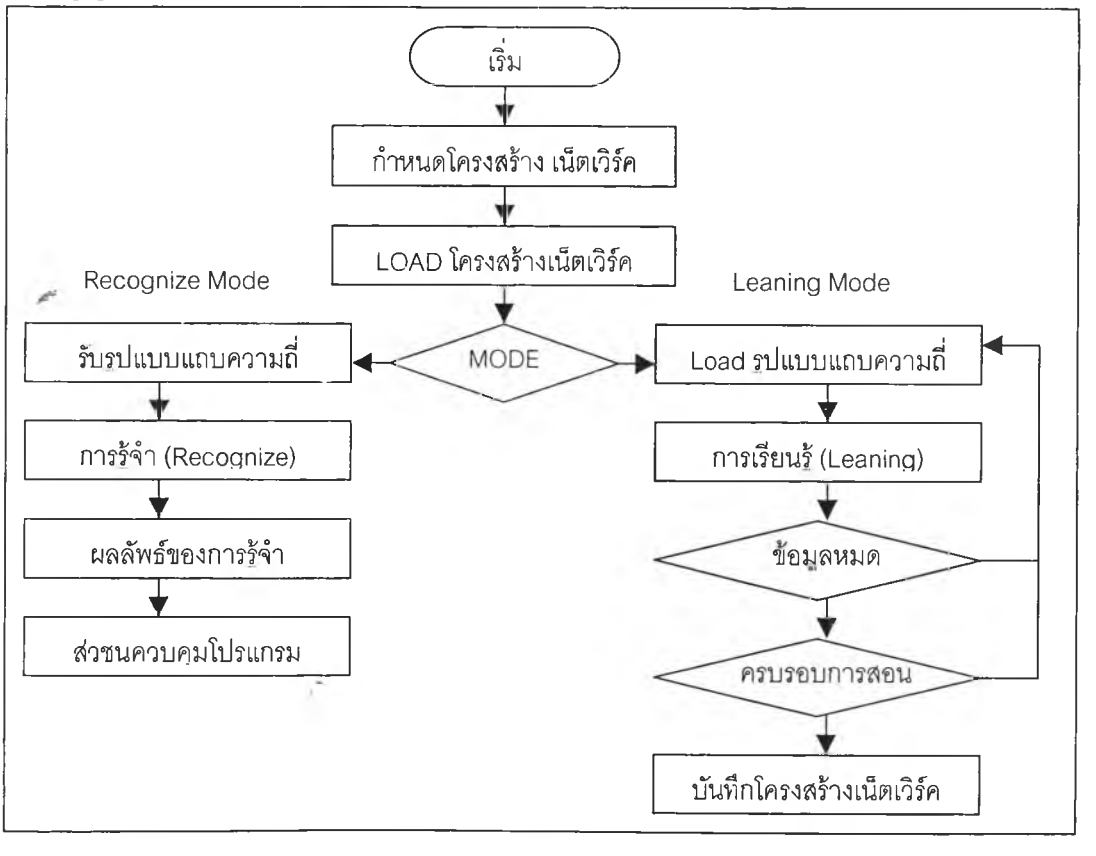
3.3.3 โปรแกรมส่วนของนิวรอลเน็ตเวิร์ค (Neural Networks)

โปรแกรมของนิวรอลเน็ตเวิร์คจะทำงานได้ 2 อย่างคือ การเรียนรู้ (Leaning) และการรู้จำ (Recognize) โดยก่อนอื่น ต้องทำการสร้างโครงสร้างของเน็ตเวิร์ค โดยใช้ ฟังก์ชัน ConstructNN(Input,Hidden,Output) และโปรแกรมจำกำหนดค่าเริ่มต้นให้ และถ้าเรามีฐานข้อมูลโครงสร้างของเน็ตเวิร์คอยู่เราก็สามารถ เรียกค่าของเน็ตเวิร์ค เดิมมาใช้โดยใช้ฟังก์ชัน ImportNN(ชื่อแฟ้มที่เก็บโครงสร้าง) และในการทำงานก็เลือกว่าเราจะทำงานในแบบ เรียนรู้ (Leaning) หรือ รู้จำ (Recognize)

การเรียนรู้ (Leaning) เราจะต้องมีต้นแบบความรู้ที่เราต้องการจะให้ นิวรอลเน็ตเวิร์ค เรียน โดยในที่นี้ก็คือ รูปแบบของแถบความถี่ของเสียงแต่ละเสียง ซึ่งเราก็จะมีต้นแบบบันทึกไว้ในฐานข้อมูล โดยที่ในฐานข้อมูลก็จะมี รูปแบบของแถบความถี่ (Spectrum) ผลลัพธ์ของเน็ตเวิร์ค และคำสั่งที่กำหนดไว้ และระบบก็จะดึงข้อมูลครั้งละข้อมูลมาสอนให้เน็ตเวิร์ค จนหมดฐานข้อมูล และจำนวนรอบการสอนก็สามารถกำหนดได้จากโปรแกรม และเมื่อการเรียนรู้สิ้นสุด เน็ตเวิร์คก็พร้อมที่จะทำงานในแบบ การรู้จำ (Recognize) จากนั้นเราก็สามารถบันทึกโครงสร้างของเน็ตเวิร์ค ไว้ในฐานข้อมูลเพื่อการใช้งานในครั้งต่อไปเราไม่จำเป็นต้อง สอน (Train) ใหม่เพื่อประหยัดเวลา โดยในโปรแกรมเรียกใช้ฟังก์ชัน ExportNN(ชื่อแฟ้ม)

การรู้จำ (Recognize) เมื่อ เน็ตเวิร์ค ได้รับการสอน (Train) หรือเรียกโครงสร้างเน็ตเวิร์คที่บันทึกไว้ มาใช้แล้ว นิวรอลเน็ตเวิร์ค ก็พร้อมที่จะตอบสนองต่อรูปแบบข้อมูลที่ได้จากการวิเคราะห์แถบความถี่ เพื่อหาว่ารูปแบบแถบความถี่ที่ส่งเข้ามาทดสอบ มีความคล้ายคลึงกับรูปแบบข้อมูลที่สอนไว้ชุดใด และส่งผลลัพธ์ที่ได้ไปยังส่วนควบคุมโปรแกรม

ในส่วนการทำงานของ นิวรอลเน็ตเวิร์ค เราสามารถแสดงขั้นตอนการทำงานได้ดังแสดงในรูปที่ 3.7 โดยรับข้อมูลรูปแบบแถบความถี่มาจากส่วนวิคราะห์แถบความถี่



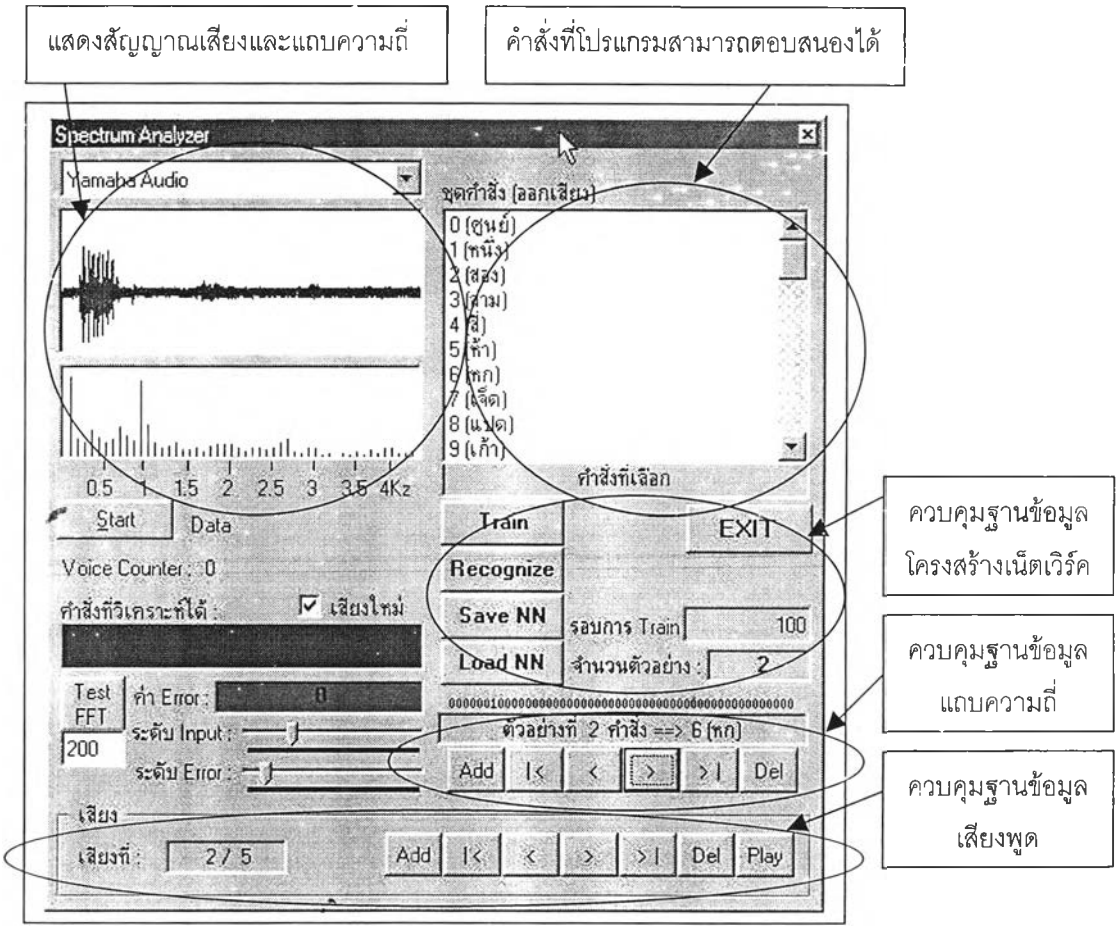
รูปที่ 3.7 การทำงานในส่วนของ นิวรอลเน็ตเวิร์ค

3.3.4 โปรแกรมส่วนควบคุมโปรแกรม

ส่วนควบคุมโปรแกรมจะรับผลลัพธ์ของการ รู้จำ (Recognition) ว่าแถบความถี่เสียงที่วิเคราะห์ได้นั้นมีรูปแบบ เหมือน หรือคล้ายกับรูปแบบแถบความถี่ที่มีการสอน (Training) ไว้ชุดใด และทำการตรวจสอบว่ารูปแบบดังกล่าวถูกกำหนดให้โปรแกรมตอบสนองอย่างได้ ซึ่งก็จะเป็นการตอบสนองจากโปรแกรมวิเคราะห์และรูปจำเสียงพูด ไปยังโปรแกรมที่กำลังทำงานอยู่ หรือระบบปฏิบัติการ โดยรูปแบบของการตอบสนองจะมีอยู่ 2 รูปแบบคือ ตอบสนองโดยการส่งรหัส คีย์บอร์ด ไปยังโปรแกรมที่ทำงาน (Active) อยู่ หรือตอบสนองโดยการเรียกโปรแกรมสำเร็จรูปอื่นๆ ในกรณีที่เป็นการส่งรหัส คีย์บอร์ด จะใช้ฟังก์ชัน SendKeys(รหัส คีย์บอร์ด) และในกรณีที่เป็นการเรียกใช้โปรแกรมสำเร็จรูปอื่นๆ ก็จะใช้ฟังก์ชัน Shell(โปรแกรมสำเร็จรูป)

3.4 การใช้งานโปรแกรมวิเคราะห์และรู้จำเสียงพูด

เมื่อเรียกใช้โปรแกรมวิเคราะห์และรู้จำเสียงพูด จะปรากฏโปรแกรมดังรูปที่ 3.8 ซึ่งในโปรแกรมสามารถแบ่งออกเป็นส่วนๆ ได้ดังรูป



รูปที่ 3.8 แสดงรูปแบบของโปรแกรมและการแบ่งส่วนต่างๆ ของโปรแกรม

3.4.1 การสอน (Training)

เมื่อ Run โปรแกรม ให้ Click ที่ [เสียงใหม่] และกดปุ่ม [Start] โปรแกรมจะรับสัญญาณเสียง และแสดงสัญญาณที่รับผ่าน Mic ในช่องสัญญาณเสียง ในขณะเดียวกันโปรแกรมจะคำนวนหาองค์ประกอบทางความถี่ของสัญญาณปัจจุบัน โปรแกรมจะทำการรับสัญญาณไปจนกว่า การตรวจสอบจุดเริ่มของคำจะเป็นจริง โดยการหาผลรวมด้านพลังของสัญญาณถึงจุดที่กำหนดไว้ เมื่อพบจุดเริ่มต้นของเสียงก็จะทำการบันทึกสัญญาณเสียงเป็นเวลา 1 วินาที และหาองค์ประกอบด้านความถี่ (Spectrum) ของสัญญาณที่รับมา และคำนวณหาผลรวมขององค์ประกอบทางความถี่ เพื่อแสดงในช่องความถี่ เมื่อได้องค์ประกอบด้านความถี่แล้ว ให้เลือกคำสั่งที่ต้องการจากรายการคำสั่ง และกดปุ่ม [Add] ค่าของจำนวนตัวอย่างจะเพิ่มขึ้น จากนั้นให้กดปุ่ม [Train] โปรแกรมจะแสดงรอบการสอนและตัวอย่างที่จะสอน นิวรอลเน็ตเวิร์ค (ในช่วงนี้โปรแกรมจะหยุดทำงานระยะหนึ่งขึ้นกับจำนวนตัวอย่าง) เมื่อโปรแกรมสอน นิวรอลเน็ตเวิร์ค ให้รู้จักตัวอย่างทั้งหมดแล้ว โปรแกรมจะแสดงกรอบโต้ตอบ แสดงว่าโปรแกรมได้สอน นิวรอลเน็ตเวิร์ค เรียบร้อย หลังจากนั้นให้ทำการบันทึก นิวรอลเน็ตเวิร์ค โดยการกดปุ่ม [Save NN] เมื่อโปรแกรมบันทึกเรียบร้อย โปรแกรมจะแสดงกรอบโต้ตอบ

3.4.2 การรู้จำเสียงพูด (Recognition)

เมื่อรันโปรแกรม (Run) ให้ตรวจสอบว่า ตัวเลือก [เสียงใหม่] ไม่ถูกเลือกไว้ และให้โหลด (Load) โครงสร้างของ นิวรอลเน็ตเวิร์ค โดยการกดที่ปุ่ม [Load NN] เมื่อโปรแกรมโหลดข้อมูล นิวรอลเน็ตเวิร์ก เรียบร้อย โปรแกรมจะแสดงกรอบโต้ตอบ และกดปุ่ม [Start] โปรแกรมจะทำการรับข้อมูลสัญญาณเสียงจากวงจร เปลี่ยนสัญญาณอนาล็อกเป็น ดิจิตอล (Analog to Digital Converter) และแสดงรูปแบบของสัญญาณอนาล็อก (Analog) ในช่องสัญญาณเสียง ในขณะเดียวกันก็จะคำนวณหาองค์ประกอบด้านความถี่ และแสดงในช่องสัญญาณความถี่ และโปรแกรมจะตรวจสอบจุดเริ่มของเสียงพูด โดยใช้การหาผลรวมด้านพลังของสัญญาณถึงจุดที่กำหนดไว้ เมื่อพบจุดเริ่มต้นของเสียงก็จะทำการบันทึกสัญญาณเสียงเป็นเวลา 1 วินาที และหาองค์ประกอบด้านความถี่ (Spectrum) ของสัญญาณที่รับมา และคำนวณหาผลรวมขององค์ประกอบทางความถี่เพื่อแสดงในช่องความถี่ เมื่อได้องค์ประกอบด้านความถี่แล้ว โปรแกรมจะส่งชุดข้อมูลด้านความถี่ เข้าไปทดสอบใน นิวรอลเน็ตเวิร์ค และ นิวรอลเน็ตเวิร์ค จะให้คำตอบของแบบที่เข้ากันได้หรือใกล้เคียงกับข้อมูลทดสอบ เมื่อได้คำตอบจาก นิวรอลเน็ตเวิร์ค แล้วก็ตรวจสอบว่าโปรแกรมถูกกำหนดให้ทำคำสั่งอะไร และจะส่งคำสั่งนั้นไปให้โปรแกรมที่ทำงานเป็นหน้าฉาก (Front End) เมื่อส่งคำสั่งไปแล้วก็จะกลับมารอรับสัญญาณเสียงพูดต่อไป

<u>คำสั่งอื่นๆ</u>

[Exit] - เมื่อต้องการออกจากโปรแกรม

[Recognize] – เมื่อต้องการให้โปรแกรมรู้จำแถบความถี่ตำแหน่งปัจจุบัน

[Add] – เพิ่มตัวอย่างเสียง หรือตัวอย่างแถบความถี่ ลงในฐานข้อมูล (Database)

[I<] , [>I] – เมื่อต้องการไปยังตำแหน่งเริ่มต้น หรือ สิ้นสุด ของข้อมูลเสียงพูด หรือข้อมูลแถบความถี่

[<] , [>] – เพื่อต้องการเลื่อนตำแหน่งของข้อมูลเสียง หรือข้อมูลแถบความถี่ ขึ้นลง 1 ตำแหน่ง

[Del] – เมื่อต้องการลบตัวอย่างข้อมูลที่ตำแหน่งปัจจุบัน

[Play] – เมื่อต้องการเล่นกลับเสียงที่บันทึกไว้ และโปรแกรมจะทำการ รู้จำเสียงดังกล่าวด้วย