

การรู้จำข้อมูลเสียงโดยใช้ดิจิทัลนิวรอล

Speech Recognition by Digital Neural



จัดทำโดย

นาย ประพันธ์พงษ์ นทกุล รหัสประจำตัว 40010436

นาย พัสกร พูลสุขศิริ รหัสประจำตัว 40010504

อาจารย์ที่ปรึกษา

ดร. ยุทธนา กิดใจเดียว

เลขหมู่.....  
เลขทะเบียน..... 42272  
วัน, เดือน, ปี..... 16 พ.ค. 2545

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

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของวิชา 01044102 PROJECT II  
ของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประจำภาคเรียนที่ 2 ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ 04274

ปริญญาโท ปีการศึกษา 2543

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การรู้จำข้อมูลเสียงโดยใช้ดิจิทัลนิวรอล

ผู้จัดทำ 1. นาย ประพันธ์พงษ์ นทกุล

2. นาย พัสกร พุดสุขศิริ



..... อาจารย์ที่ปรึกษา

(ดร. ยุทธนา คัดใจเดียว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรู้จำข้อมูลเสียงโดยใช้ดิจิทัลนิวรัล

SPEECH RECOGNITION by DIGITAL NEURAL

นาย ประพันธ์พงษ์ นทกุล 40010436

นาย พัสกร พูลสุขศิริ 40010504



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การรู้จำเสียงโดยใช้ดิจิทัลนิวรอล

นาย ประพันธ์พงษ์ นทกุล

นาย พัสกร พูลสุขศิริ

ดร. ยุทธนา คิดใจเตี๋ยว (อาจารย์ที่ปรึกษา)

ภาคเรียนที่ 2 ปีการศึกษา 2543

### บทคัดย่อ

ปฏิญานิพนธ์ฉบับนี้นำเสนอวิธีการแยกแยะและรู้จำข้อมูลเสียง ด้วยโครงข่ายประสาทเทียมชนิด Weightless Neural Networks เพื่อให้รู้จำเสียงของบุคคลคนเดียว และไม่ต่อเนื่อง โดยเสียงพูดจะถูกบันทึกผ่านเข้ามาทางไมโครโฟน สัญญาณที่ได้จะถูกแปลงเป็นสัญญาณดิจิทัลด้วยการ์ดเสียง (Sound Blaster Card) จากนั้นจะผ่านกระบวนการปริโปรเซสซิ่ง โดยจะเก็บข้อมูลในโคเมนของเวลา และแปลงเป็นข้อมูลในโคเมนของความถี่ ด้วยการแปลงฟาสท์ฟูเรียร์ จากนั้นนำข้อมูลไป Normalize เพื่อเตรียมข้อมูลก่อนที่จะเข้าสู่กระบวนการแยกแยะและรู้จำเสียงของนิวรอลต่อไป

## SPEECH RECOGNITION by DIGITAL NEURAL

Mr.Prapanphong                      Natakul  
Mr.Passakorn                         Poonsuksiri  
Dr.Yuthana Kidjaideaw(Adviser)  
2<sup>nd</sup> Semestor, Education year 2000

### Abstract

This thesis presents the method of separating and recognizing speech signal by Neural Network ; Weightless Neural Network for memorizing discrete signal of one person. Thus the speech signal will be recorded via microphone. The recorded signal will be transformed to digital signal by sound card. After that it passed through Pre-Processing. The signal will be kept into time domain , and transformed to frequency domain by Fast Fourier Transform (FFT). Then it will be normalized for preparing before entering the process of Neural continuously.

# SEPARATE AND RECOGNITION by NEURAL NEURAL

Mr. Prapong Watan

Mr. Pasitorn Boonsuksri

Dr. Willem Widjaidraw (Advisor)

Director, Education year 2008

This thesis presents the method of separating and recognizing speech signal by Neural Network ; Weightless Neural Network for memorizing discrete signal of one person. Thus the speech signal will be recorded via microphone this signal will be transformed to digital signal by sound card. After that it is through Pre-Processing. This signal will be kept into time domain, and transformed to frequency domain by Fast Fourier Transform (FFT). Then it will be normalized for preparing before entering the process of Neural continuously.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
<b>บทที่ 1 บทนำ</b>	
1.1 ความเป็นมาของปัญหา	1
1.2 วัตถุประสงค์	2
1.3 ขั้นตอนและวิธีดำเนินการ	2
<b>บทที่ 2 ทฤษฎีและหลักการ</b>	
2.1 เสียงพูด	3
2.2 อวัยวะที่ใช้ในการออกเสียง (Organs of Speech)	3
2.3 กระบวนการผลิตเสียงพูด	6
2.4 เสียงพูดของมนุษย์	6
2.5 ความถี่ฟอร์แมนท์ (Formant Frequency)	7
2.6 แบบจำลองระบบกำเนิดเสียงพูด	8
2.7 ความเข้าใจ (Perception)	9
2.8 การเตรียมข้อมูลก่อนการประมวลผล	9
2.9 การวิเคราะห์เสียงพูด (Speech Analysis)	12
2.10 นิวรอลเน็ตเวิร์ค (Neural Network)	15
<b>บทที่ 3 การออกแบบ</b>	
3.1 ส่วนปริ้โพรเซสซิ่ง (Pre-Processing)	29
3.2 ส่วนการแยกแยะและตัดสติใจ	30
<b>บทที่ 4 การดำเนินการและทดลอง</b>	
4.1 การดำเนินการ	35
4.2 การทดลอง	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง	
5.1 สรุปผลการทดลอง	44
5.2 วิจารณ์ผลการทดลอง	44
5.3 แนวทางการพัฒนา	45
ภาคผนวก	46
กิตติกรรมประกาศ	47
หนังสืออ้างอิง	48



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

	หน้า
รูปที่ 1.1 แสดงบล็อกไดอะแกรมการทำงานส่วนต่างๆ	2
รูปที่ 2.1 ภาพตัดขวางแสดงอวัยวะในระบบการพูดของมนุษย์	4
รูปที่ 2.2 ภาพกล่องเสียงขณะ (ก) หายใจปกติ (ข) หายใจเข้าลึกๆ (ค) กำลั้งส่งเสียง (ง) ส่งเสียงกระซิบหรือเสียงแผ่ว	5
รูปที่ 2.3 (ก) ตัวอย่างรูปคลื่นของสัญญาณเสียงก้อง /a/ (ข) ตัวอย่างรูปคลื่นของสัญญาณเสียงไม่ก้อง /sh/	7
รูปที่ 2.4 สเปกตรัมของเสียง /a/ ที่ได้จากตัวประสิทธิ์ LPC	8
รูปที่ 2.5 แผนภาพกรอบจำลองระบบกำเนิดเสียงเริ่มต้น	8
รูปที่ 2.6 กราฟแสดงระดับความเท่ากันของระดับความดัง	9
รูปที่ 2.7 แสดงโครงสร้างเวฟไฟล์ที่มีรูปแบบของไฟล์ RIFF	11
รูปที่ 2.8 ตัวอย่างเซลล์ประสาทชีวภาพ	16
รูปที่ 2.9 ไดอะแกรมของนิวนอล 1 หน่วย	17
รูปที่ 2.10 ซิกมอยด์ ฟังก์ชัน (Sigmoid function)	18
รูปที่ 2.11 Hyperbolic tangent function	19
รูปที่ 2.12 โครงข่ายประสาทเทียมแบบชั้นเดียว	19
รูปที่ 2.13 Linear Separable (สามารถแบ่งได้เชิงเส้น)	19
รูปที่ 2.14 Non Linear Separable (ไม่สามารถแบ่งได้เชิงเส้น)	20
รูปที่ 2.15 โครงข่ายประสาทเทียมแบบหลายชั้น	20
รูปที่ 2.16 ไดอะแกรมของ Backpropagation Neural Network แบบ 2 ชั้น	21
รูปที่ 2.17 (a) A RAM-based neuron (b) 4 RAMs discriminators with 3 inputs each	24
รูปที่ 2.18 A schematic view of a basic GNU	26
รูปที่ 3.1 แสดงรูปสัญญาณที่ถูกรบกวนมาไลซ์	30
รูปที่ 3.2 แสดงการนำข้อมูลเสียงมาวางซ้อนกัน	30
รูปที่ 3.3 แสดงกราฟสมการเกาส์เซียน	32
รูปที่ 3.4 แสดงการสุ่มจับพิกัดของขานิวรอลแต่ละตัว	32
รูปที่ 4.1 แสดงหน้าจอของเมนูหลัก	35
รูปที่ 4.2 แสดงรูปแบบการใส่ Password ของโปรแกรม	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.3 แสดงหน้าจอการฝึกสอนของโปรแกรม	36
รูปที่ 4.4 แสดงสัญญาณเสียงพูด “ศูนย์”	37
รูปที่ 4.5 แสดงสัญญาณเสียงพูด “หนึ่ง”	37
รูปที่ 4.6 แสดงสัญญาณเสียงพูด “สอง”	37
รูปที่ 4.7 แสดงสัญญาณเสียงพูด “สาม”	38
รูปที่ 4.8 แสดงสัญญาณเสียงพูด “สี่”	38
รูปที่ 4.9 แสดงสัญญาณเสียงพูด “ห้า”	38
รูปที่ 4.10 แสดงสัญญาณเสียงพูด “หก”	39
รูปที่ 4.11 แสดงสัญญาณเสียงพูด “เจ็ด”	39
รูปที่ 4.12 แสดงสัญญาณเสียงพูด “แปด”	39
รูปที่ 4.13 แสดงสัญญาณเสียงพูด “เก้า”	40
รูปที่ 4.14 แสดงหน้าจอของส่วนทดสอบ	40
รูปที่ 4.15 แสดงหน้าจอผลลัพธ์ที่ได้จากการทดสอบ	41
รูปที่ 4.16 กราฟแสดงผลการทดลอง	43

## สารบัญตาราง

	หน้า
ตารางที่ 2.1 Main parameters of a GNU	26
ตารางที่ 3.1 แสดงการ Optimize ข้อมูล	31
ตารางที่ 3.2 แสดงการหา Hamming Distance	33
ตารางที่ 3.3 แสดงการให้คะแนนข้อมูล	34
ตารางที่ 4.1 ผลการทดลอง	42



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาของปัญหา

ปัจจุบันเครื่องคอมพิวเตอร์ถูกนำมาใช้กันอย่างแพร่หลาย เพื่อช่วยให้การประมวลผลเป็นไปอย่างรวดเร็วและถูกต้อง วิธีการใช้คอมพิวเตอร์จึงถูกพัฒนาอย่างต่อเนื่อง จากการติดต่อที่ผู้ใช้จะต้องพิมพ์คำสั่งลงไปทีเรียกว่าเป็นการติดต่อแบบ Text Based เปลี่ยนมาเป็นการใช้สัญลักษณ์ที่เป็นภาพกราฟิก ที่เรียกว่าเป็นการติดต่อที่เป็น Graphics User Interface การพัฒนาสิ่งเหล่านี้ก็เพื่อที่จะแก้ปัญหา หรือลดความยุ่งยากในการใช้เครื่องคอมพิวเตอร์ลง นอกจากความพยายามในการแก้ปัญหาในแง่การออกแบบซอฟต์แวร์ และการออกแบบฮาร์ดแวร์ เช่น การนำอุปกรณ์อินพุทที่เป็น Pointing Device เช่น เมาส์ จอระบบสัมผัส หรือการพัฒนาคอมพิวเตอร์ระบบปากกา เพื่อให้สามารถใช้เครื่องคอมพิวเตอร์ได้อย่างสะดวกมีประสิทธิภาพ ในปัจจุบันการพัฒนาเทคโนโลยีด้านมัลติมีเดียเป็นไปอย่างต่อเนื่อง ทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ สามารถบันทึกเสียง ภาพ วิดีโอ เพื่อใช้งานร่วมกันในคอมพิวเตอร์ได้ สิ่งหนึ่งที่น่าสนใจติดตามคือ การพัฒนาวิธีการควบคุมหรือสั่งงานคอมพิวเตอร์ด้วยเสียง

ด้วยความคิดฝันที่จะควบคุมการทำงานของเครื่องคอมพิวเตอร์ด้วยเสียงพูด ทำให้มีการวิจัยและพัฒนาเทคโนโลยีเกี่ยวกับเสียงมาเป็นระยะเวลาหลายปี แต่ส่วนใหญ่จะเน้นการใช้งานกับเครื่องใหญ่ เพื่อควบคุมระบบโทรคมนาคม หรือเป็นการพัฒนาเพื่อการรักษาความปลอดภัยทางการทหาร สำหรับเครื่องไมโครคอมพิวเตอร์เมื่อมีการผลิต Sound Card ขึ้นมาใช้เพื่อเป็นอุปกรณ์บันทึกเสียงใช้กับ โปรแกรมมัลติมีเดีย นักวิจัยทั้งหลายก็เริ่มมองเห็นความฝันที่เป็นจริงนั้นขึ้นมาในการที่จะประยุกต์เอาข้อมูลเสียงมาควบคุมการทำงานของเครื่องคอมพิวเตอร์ นับว่าเป็นงานวิจัยที่ได้รับความสนใจมาก

ในการควบคุมคอมพิวเตอร์ด้วยเสียงนั้นจะพบว่ามีปัญหาที่เกิดขึ้น ซึ่งปัญหาพื้นฐาน 2 ประการที่ควรคำนึงถึงนั้น คือ

ประการแรก ในแต่ละคำพูดของคนเราที่เปล่งออกมาจะมีลักษณะเฉพาะ แม้ว่าเราจะพยายามเปล่งคำพูดให้มีเสียงใกล้เคียงกันเท่าใดก็ตาม เราจะพบถึงความแตกต่างที่เกิดขึ้นได้อย่างชัดเจน

ประการที่สอง เนื่องจากเสียงคนเราที่ใช้ในการสื่อความหมายต่าง ๆ นั้น จะอยู่ในช่วงความถี่ 4 - 5 กิโลเฮิรตซ์ และจากทฤษฎีการสุ่มตัวอย่าง (Sampling Theorem) พบว่า ในการที่จะรักษาความหมายของเสียงที่พูดออกมา 1 วินาทีให้สมบูรณ์นั้น จะต้องใช้อัตราการสุ่มตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างน้อย 2 เท่าของความถี่เสียง (Carlson A.B., 1975) ซึ่งจะเท่ากับ 8,000 - 10,000 ค่าเป็นอย่างน้อย ในการรับรู้เสียงพูดของเครื่องคอมพิวเตอร์นั้นจำเป็นต้องใช้เสียงพูดเป็นหลายวินาทีและผู้พูดหลายคน ดังนั้นจึงมีข้อมูลจำนวนมากที่จะต้องถูกเก็บและนำไปใช้ ซึ่งจะมีผลต่อสื่อที่ใช้ในการบันทึก และเวลาที่ใช้ในการรับรู้ เทคนิคของการวิเคราะห์ให้ได้พารามิเตอร์ของสัญญาณ เพื่อลดจำนวนข้อมูลลงจึงเป็นทางออกประการหนึ่งในการแก้ปัญหา

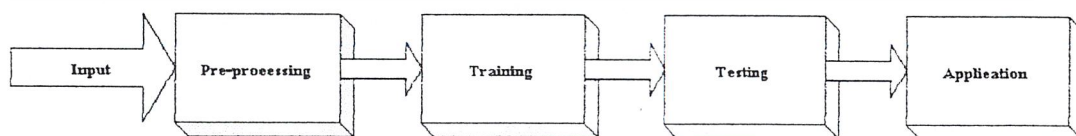
## 1.2 วัตถุประสงค์

1. เพื่อพัฒนาระบบให้สามารถรับรู้เสียงพูดแบบบุคคลเดียว และไม่ต่อเนื่อง
2. เพื่อหาอัลกอริทึมที่เหมาะสมสำหรับเพิ่มอัตราความถูกต้องในการรับรู้
3. เพื่อประยุกต์ทฤษฎีต่างๆเพื่อช่วยในการรับรู้ เช่น ไดนามิกโปรแกรมมิ่ง การสร้างแบบอ้างอิง เป็นต้น

## 1.3 ขั้นตอนและวิธีดำเนินการ

1. ศึกษาค้นคว้าลักษณะเบื้องต้นของสัญญาณเสียงพูดคำโดด
2. ศึกษาทฤษฎีและหลักการทำงานของ Digital Neural Network ชนิด Weightless
3. ศึกษาและวิเคราะห์สัญญาณเสียงในกระบวนการ Pre-Processing เช่น การบันทึกเสียงพูดผ่านไมโครโฟน การแปลงสัญญาณเสียงในแกนเวลาให้เป็นแกนความถี่ การนอร์มาไลซ์สัญญาณ เป็นต้น
4. เขียนโปรแกรมสร้างแบบการทำงานของ Digital Neural Network ชนิด Weightless
5. ทำการทดสอบความถูกต้องในการจดจำสัญญาณเสียงพูดคำโดดจำนวน 10 คำว่าเป็นเสียงของคำใด

โครงสร้างการทำงานของปริญาณิพจน์ฉบับนี้แบ่งเป็นส่วนๆ ได้ ดังรูปที่ 1.1



รูปที่ 1.1 แสดงบล็อกไดอะแกรมการทำงานส่วนต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 เสียงพูด

คนเราเปล่งเสียงพูดด้วยอวัยวะที่ใช้ในการออกเสียง (Organs of Speech) ทำเสียงตามที่มีอยู่ในระบบภาษาของตน แม้ว่าคนที่อยู่ในสังคมเดียวกันจะใช้ภาษาเดียวกัน แต่ถ้าพิจารณาเสียงที่เปล่งออกมาจริงๆ ในแต่ละครั้งแล้ว ก็อาจจะสังเกตลักษณะที่แตกต่างกันได้ เราจึงสามารถจำเสียง จำวิธีพูดของคนที่เราคุ้นเคยได้ โดยเสียงพูดนี้มีลักษณะที่จะอธิบายได้ด้วยหลักเกณฑ์ทางวิทยาศาสตร์ซึ่งเป็นสากล แม้ว่าภาษาหนึ่งๆจะมีเสียงแตกต่างกันไปมากน้อยบ้าง แต่ละเสียงก็สามารถจะนำมาพิจารณา และอธิบายให้รู้ลักษณะการออกเสียง และตำแหน่งที่เกิดของเสียงได้ คำอธิบายนี้จะทำให้เข้าใจลักษณะเสียงทุกเสียง ซึ่งวิชาที่ว่าด้วยเสียงพูดเราเรียกว่า “ สัทศาสตร์ (Phonetics) ” ในการศึกษาเรื่องเสียงพูดสามารถแบ่งออกได้เป็น 2 ลักษณะ คือ

1. สรีรศาสตร์ (Articulatory Phonetics) เป็นการศึกษาเสียงพูดจากอวัยวะ และการเคลื่อนไหวของอวัยวะที่ทำให้เกิดเสียงพูด การอธิบายก็จะอธิบายโดยอาศัยลักษณะและการเคลื่อนไหวของอวัยวะที่เกี่ยวข้องในการเปล่งเสียงพูดนั้น
2. กกลศาสตร์ (Acoustic Phonetics) เป็นการศึกษาเสียงพูดจากลักษณะคลื่นเสียงที่ผู้พูดเปล่งออกมาแล้ว และผู้ฟังได้ยินว่ามีลักษณะทางกลศาสตร์เป็นอย่างไร การศึกษาตามแนวนี้ต้องอาศัยความรู้ทางฟิสิกส์ และคณิตศาสตร์เข้าช่วยอธิบายลักษณะของคลื่นเสียง

จากการศึกษาด้านกายวิภาคศาสตร์ของมนุษย์ วิชาที่ว่าด้วยเสียงพูด (Phonetics) และศาสตร์ทางด้านเสียง (Acoustic) ช่วยให้เข้าใจขั้นตอนการทำงานร่วมกันของอวัยวะต่างๆ ในการเปล่งเสียงพูด ตลอดจนลักษณะทางกายภาพของเสียงพูด เพื่อนำมาวิเคราะห์และสร้างแบบจำลองเลียนแบบเสียงพูดของมนุษย์

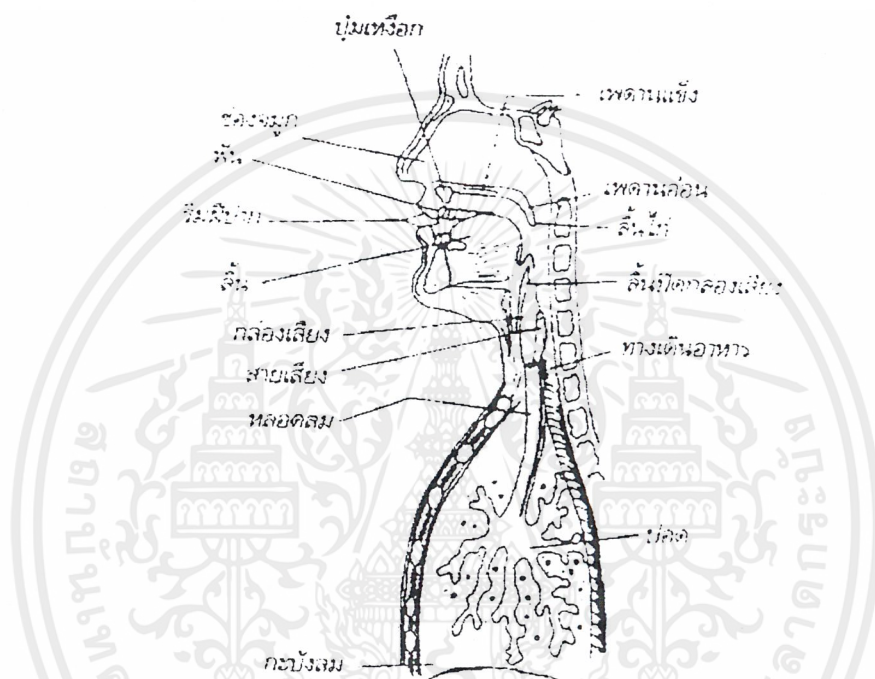
#### 2.2 อวัยวะที่ใช้ในการออกเสียง (Organs of Speech)

การทำให้เกิดเสียงเป็นหน้าที่หนึ่งของระบบหายใจ การออกเสียงหรือการพูดของมนุษย์แต่ละครั้งจะต้องมีการทำงานร่วมกันของอวัยวะต่างๆ ของร่างกาย ดังรูปที่ 2.1 อวัยวะที่ใช้ในการออกเสียงพูดมีอยู่หลายส่วน ซึ่งแต่ละส่วนสามารถทำให้เสียงพูดแตกต่างกันไปได้ อวัยวะเหล่านี้มีปาก และส่วนต่างๆ ภายในปาก ช่องคอ กล่องเสียง ช่องปาก และช่องจมูก เป็นต้น

อวัยวะที่ใช้ในการออกเสียงแบ่งได้เป็น 2 พวก คือ

1. อวัยวะที่ใช้ในการกระทำอาหาร (Articulator) หมายถึง อวัยวะส่วนที่เคลื่อนไหวเพื่อผลิตลมไปยังส่วนต่างๆ อวัยวะตัวผลักดันอาหารที่สำคัญคือ ลิ้น ซึ่งเป็นส่วนที่เคลื่อนไหวได้มากที่สุด

2. อวัยวะที่เป็นตำแหน่งที่เกิดเสียงต่างๆ (Point of Articulator) หมายถึง ตำแหน่งหรือฐานกรณ์ที่เกิดเสียงต่างๆ เช่น เพดานส่วนต่างๆ ริมฝี ปาก ฟัน เป็นต้น



รูปที่ 2.1 ภาพตัดขวางแสดงอวัยวะในระบบการพูดของมนุษย์

ซึ่งอวัยวะต่างๆที่ช่วยในการออกเสียงมีรายละเอียดดังต่อไปนี้

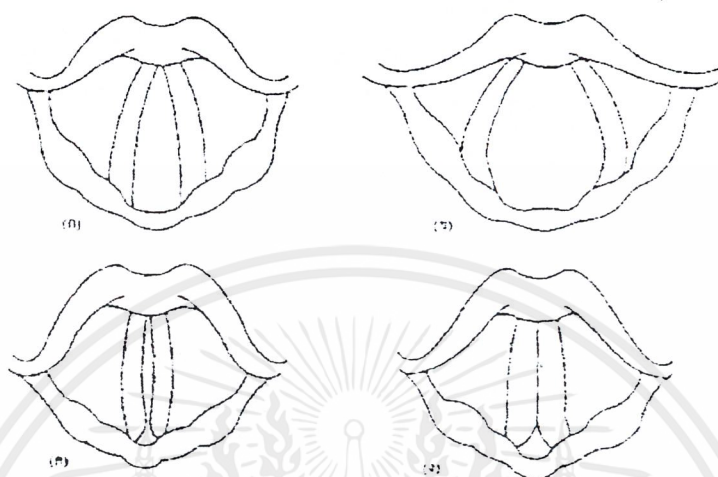
1. ปอดและกระบังลม ทำหน้าที่สำคัญในการหายใจ และเป็นต้นกำเนิดการไหลของอากาศในกระบวนการผลิตเสียง

2. หลอดลม ทำหน้าที่นำอากาศจากปอดผ่านกล่องเสียง และเป็นอวัยวะที่อยู่ด้านหน้าของหลอดอาหาร

3. กล่องเสียง เป็นอวัยวะพิเศษที่ทำหน้าที่เป็นทางเดินอากาศเวลาหายใจ และเป็นตัวผลิตพัลส์ (pulse) ของอากาศขณะเปล่งเสียง ซึ่งประกอบด้วย เส้นเสียง (Vocal Cords) เส้นเสียงเป็นอวัยวะสำคัญที่เกิดของเสียง มีลักษณะเป็นกล้ามเนื้อ 2 แผ่นภายในกล่องเสียงปิดขวางอยู่ปากช่องหลอดลมจากด้านหลังมาด้านหน้า ระหว่างเส้นเสียงจะมีช่องว่างซึ่งเป็นทางให้ลมผ่านเข้าไปถึงปอด และออกมาจากปอดได้โดยช่องนี้เรียกว่า ช่องว่างระหว่างเส้นเสียง (Glottis) เส้นเสียงทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สองสามารถจะดึงออกให้ห่างจากกัน หรือดึงเข้ามาชิดกันก็ได้ เส้นเสียงเป็นส่วนที่สำคัญที่ทำให้เกิดเสียงพูด รูปร่างของกล่องเสียงและเส้นเสียงในลักษณะต่างๆ แสดงในรูปที่ 2.2



รูปที่ 2.2 ภาพกล่องเสียงขณะ (ก) หายใจปกติ (ข) หายใจเข้าลึกๆ  
(ค) กำลำส่งเสียง (ง) ส่งเสียงกระซิบหรือเสียงแผ่ว

4. ช่องคอ (Pharynx) หมายถึง ช่องคอซึ่งอยู่ถัดจากช่องปากลงไปจนถึงเส้นเสียง
5. แผ่นเนื้อปากหลอดลม (Epiglottis) เป็นก้อนเนื้อเล็กๆ คล้ายลิ้นไก่ อยู่ต่อจากโคนลิ้นลงไปในลำคอ มีหน้าที่ปิดช่องลมเมื่อรับประทานอาหาร และเปิดช่องลมเมื่อพูด
6. ฟัน เป็นอวัยวะที่เกิดของเสียงหลายชนิด เช่น เมื่อฟันบนกดลงบนริมฝี ปากล่าง หรือกดกับฟันล่าง ลมที่ผ่านออกมาโดยแรงจะลอดช่องที่พอฟ่านได้ออกมา ทำให้เกิดเป็นเสียงชนิดที่เรียกว่า เสียงเสียดแทรกที่เกิดที่ฟัน เป็นต้น
7. ปุ่มเหงือก เป็นส่วนนูนออกมาอยู่หลังฟันค้ำบน ถ้าเอาลิ้นแตะจะรู้สึกรูปร่างว่ามีลักษณะเป็นคลื่น ปุ่มเหงือกเป็นบริเวณที่เกิดเสียง(Alveolar Sound) ปุ่มเหงือกนั้นเป็นตำแหน่งสำคัญในการอธิบายเรื่องเสียงตำแหน่งหนึ่ง
8. เพดานแข็งหรือเพดานปาก คือ ส่วนเฉพาะเพดานที่โค้งเป็นกระดูกแข็ง
9. เพดานอ่อน คือ ส่วนของเพดานที่อยู่ต่อจากเพดานแข็งไปข้างใน มีลักษณะเป็นกระดูกอ่อนที่ขยับขึ้นลงได้ เวลาหายใจเพดานอ่อน และลิ้นไก่ซึ่งอยู่ปลายเพดานอ่อนจะลดระดับลงมาเปิดช่องให้ลมออกไปทางจมูก เวลาพูดส่วนใหญ่ปลายเพดานอ่อน และลิ้นไก่จะถูกยกขึ้นไปจรดกับหลังคอ นอกจากเวลาออกเสียงนาสิกเท่านั้น ที่เพดานอ่อนจะลดระดับลงมา
10. ลิ้นไก่ เป็นก้อนเนื้อเล็กๆ อยู่ต่อจากปลายเพดานอ่อนตรงกลางปาก สั้นเร็วได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ลิ้น ลิ้นเป็นส่วนที่เคลื่อนไหวมากที่สุดในการออกเสียงพูด จึงต้องแบ่งออกเป็น 3 ส่วนด้วยกัน ตามหน้าที่ที่มีในการออกเสียง

- ปลายลิ้น (Blade of the Tongue) คือ ลิ้นส่วนปลาย ซึ่งสามารถจะยกขึ้นไปแตะอวัยวะส่วนต่างๆในปากตอนบนได้ง่าย

- หน้าลิ้น (Front of the Tongue) คือ ลิ้นส่วนที่อยู่ตรงข้ามกับเพดานแข็ง จะวางลิ้นราบกับปากอย่างในขณะที่ไม่พูด

- หลังลิ้น (Back of the Tongue) คือ ส่วนของลิ้น ซึ่งถ้าวางลิ้นราบกับปาก ตามปกติจะอยู่ตรงข้ามกับเพดานอ่อน

12. ช่องจมูก (Nasal Cavity) หมายถึง โพรงในช่องจมูกซึ่งอยู่เหนือลิ้น ใกล้เคียงกับเป็นช่องที่ลมซึ่งผ่านเส้นเสียงขึ้นมาจะผ่านออกไปทางจมูกได้เมื่อเวลาหายใจ และเวลาออกเสียงนาสิก ในเวลาพูดเสียงอื่นๆ ลิ้นใกล้เคียงจะยกขึ้นไปปิดช่องจมูก เพื่อให้ลมออกมาทางปาก

### 2.3 กระบวนการผลิตเสียงพูด

จากระบบเสียงพูด สามารถจำแนกกลไกการสร้างเสียงพูดของมนุษย์ได้ 3 แบบ ดังนี้

1. อากาศไหลจากปอดจะถูกมอดูเลต (Modulate) โดยการสั่นของเส้นเสียงทำให้เกิดคลื่นเสียงลักษณะคล้ายพัลส์ที่มีความยาวแบบควอไซ (Quasi-periodic pulse-like excitation)

2. อากาศที่ไหลจากปอดถูกทำให้ปั่นป่วน ด้วยการบังคับให้ไหลผ่านช่องแคบอันเกิดจากการบีบตัวของอวัยวะในช่องปาก ทำให้เกิดเสียงลักษณะเสียงรบกวน (Noise-like excitation)

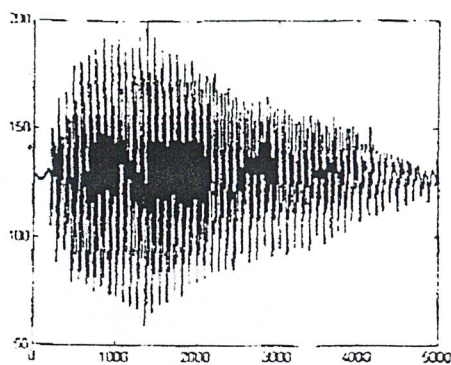
3. อากาศที่ไหลถูกกัก และเกิดแรงดันอยู่ภายในส่วนของช่องปากที่ปิด จากนั้นจึงปล่อยให้อากาศที่มีแรงดันพุ่งออกไปอย่างรวดเร็ว ทำให้เกิดการกระตุ้นเป็นเสียงในช่วงเริ่มต้น (Transient excitation)

### 2.4 เสียงพูดของมนุษย์

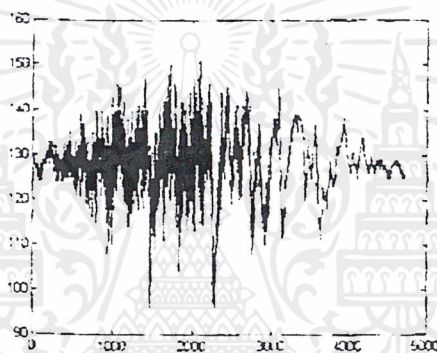
เสียงพูดเป็นคลื่นตามยาว (Longitudinal Wave) เกิดจากการสั่นของอนุภาคตัวกลางนั้น คือ อากาศ และทิศทางการสั่นของอนุภาคจะอยู่ในทิศเดียวกันกับทิศทางของการเคลื่อนที่ คลื่นเสียงเป็นคลื่นที่เปลี่ยนแปลงไปตามเวลา เสียงพูดแบ่งออกได้เป็น 2 ชนิดตามการกำเนิดเสียงหรือโหมด (Mode) การกระตุ้น คือ

1. เสียงก้องหรือเสียงโฆษะ (Voiced) เกิดจากการบังคับอากาศให้ผ่านช่องสายเสียง ทำให้มีการเปลี่ยนแปลงความตึงหย่อนของเส้นเสียง โดยเส้นเสียงจะสั่นและเกิดเป็นพัลส์ (pulse) ของอากาศไปกระตุ้นอวัยวะกำหนดเกิดเป็นเสียงก้อง ตัวอย่างเสียงก้อง ได้แก่ เสียงสระ เสียงพยัญชนะ ที่ต้องออกเสียงจากคำค่อ (Voiced Consonants) เช่น เสียง /a/ ดังรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 (ก) ตัวอย่างรูปคลื่นของสัญญาณเสียงก้อง /a/



รูปที่ 2.3 (ข) ตัวอย่างรูปคลื่นของสัญญาณเสียงไม่ก้อง /sh/

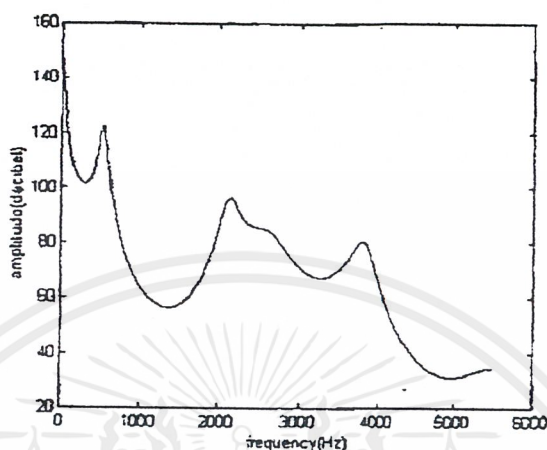
2. เสียงไม่ก้องหรือโหมยะ (Unvoiced หรือ Voiceless) เป็นเสียงที่เกิดในช่องปากหรือโพรงจมูก โดยอวัยวะภายในช่องปาก ริมฝีปาก ขวางการไหลของอากาศให้ผ่านได้เป็นช่องเล็กๆ อากาศจึงไหลผ่านอย่างรวดเร็ว และปั่นป่วนจนกระทั่งสร้างเป็นเสียงรบกวนช่วงความถี่กว้าง (Broad-spectrum noise) ตัวอย่างเสียงไม่ก้องได้แก่ เสียงพยัญชนะที่ไม่ได้เกิดจากคำคอ (Voiceless consonants) เช่น เสียง /sh/ ดังรูปที่ 2.3 (ข)

## 2.5 ความถี่ฟอร์แมนท (Formant Frequency)

ความถี่ฟอร์แมนท คือความถี่กำทอน (Resonance Frequency) ของกลุ่มอวัยวะกำทอนเสียง ความถี่ฟอร์แมนทของเสียงจะมีค่าเท่าใดขึ้นอยู่กับกระบังค้ำขนาดและรูปร่างของอวัยวะส่วนนี้ อันเกิดเป็นเสียงความถี่ที่แตกต่างกันนั้นคือ เสียงพูดต่าง ๆ นั้นเอง ความถี่ฟอร์แมนทสำหรับเสียงพูดคำหนึ่งๆอาจมีได้หลายค่า ดังรูปที่ 2.4 เป็นสเปกตรัมของเสียง /a/ ซึ่งมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่ฟอร์แมนท์ที่ 1,2,3 และ 4 อยู่ ณ ตำแหน่งความถี่ที่ 550 ,2150 ,2600 และ 3750 เฮิรตซ์ ตามลำดับ

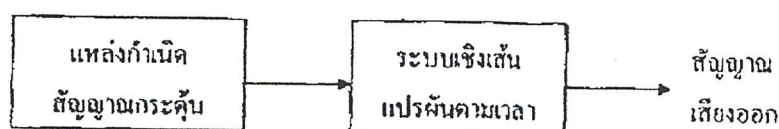


รูปที่ 2.4 สเปกตรัมของเสียง /a/ ที่ได้จากสัมประสิทธิ์ LPC

## 2.6 แบบจำลองระบบกำเนิดเสียงพูด

จากภาพรูปที่ 2.5 สามารถแสดงแผนภาพกรอบจำลองระบบกำเนิดเสียงเบื้องต้น มีการแยกภาคแหล่งกำเนิดสัญญาณกระตุ้น ออกจากส่วนกำเนิดเสียง ซึ่งแทนด้วยระบบเชิงเส้นแปรผันตามเวลา

แหล่งกำเนิดสัญญาณกระตุ้นทำหน้าที่แทนการทำงานของปอด และกล่องเสียงส่วนนี้จะผลิตขบวนพัลส์ที่มีคาบเวลาพิชัขณะเปล่งเสียงว้อยซ์ และให้กำเนิดเสียงซึ่งคล้ายเสียงรบกวนขณะเปล่งเสียงอโฆมะ (Unvoiced) ส่วนที่สองเป็นท่อนำทอนเสียง จะแทนการทำงานของช่องปากและโพรงจมูก ทำหน้าที่เสมือนตัวกรองสัญญาณ (Filter) ที่ยอมให้ความถี่ฟอร์แมนท์ผ่านได้ ซึ่งสามารถแทนด้วยระบบเชิงเส้นแปรผันตามเวลา (Time-varying linear system)

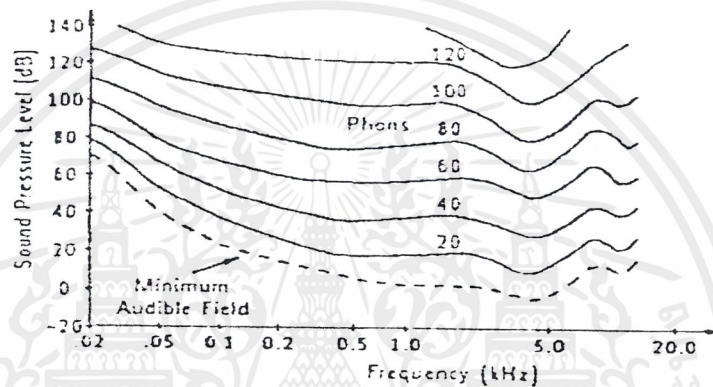


รูปที่ 2.5 แผนภาพกรอบจำลองระบบกำเนิดเสียงเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.7 ความเข้าใจ (Perception)

หน้าที่ของการรับเสียงและความสามารถ โดยปกติจะอยู่ในช่วง 16 Hz ถึง 16 KHz ในเด็กอาจถึง 20 KHz ยังมีอยู่ การรับฟังจะลดสมรรถภาพลงตามอายุ อาจได้ถึง 10 KHz การได้ยินความถี่ในช่วงย่านต่ำสัญญาณที่ได้ยินจะเป็นแบบขบวนสัญญาณ (Pulse train) ที่ความถี่ด้านสูงความถี่จะค่อยๆ ลดลงจนไม่ได้ยิน ช่วงของความเข้มจะอยู่ในช่วงที่ 1 - 130 dB ที่ความเข้มต่ำเสียงจะเบาจนไม่ได้ยิน ส่วนที่ความเข้มสูงอาจทำให้แก้วหูเจ็บปวดได้



รูปที่ 2.6 กราฟแสดงระดับความเท่ากันของระดับความดัง

## 2.8 การเตรียมข้อมูลก่อนการประมวลผล

### 2.8.1 การบันทึกเสียง

ในการบันทึกเสียงจะมีรูปแบบการบันทึก ซึ่งใช้ขบวนการพัลส์โค้ดมอดูเลชัน (PCM : Pulse Code Modulation) รูปแบบการบันทึกมีดังต่อไปนี้

1. บันทึกแบบสเตอริโอ (Stereo) หรือ โมโน (Mono)
  - แบบสเตอริโอ จะมีการแยกเสียงที่เข้ามาว่ามาจากไมโครโฟนซ้ายหรือไมโครโฟนขวา
  - แบบโมโน จะเป็นการบันทึกแบบธรรมดา คือเสียงเข้ามาจากไมโครโฟนตัวเดียว
2. อัตราการแซมปลิงต่างๆ (Sampling Rate) เช่น 11.025 KHz , 22.050 KHz , 44.100 KHz เมื่ออัตราการแซมปลิงยิ่งสูงจะยิ่งเก็บรายละเอียดของเสียงได้มาก
3. จำนวนบิตในการบันทึกต่อ 1 แซมเปิล (Sample) เช่น 8 บิตต่อ 1 แซมเปิล , 16 บิตต่อ 1 แซมเปิล เมื่อจำนวนบิตในการบันทึกต่อ 1 แซมเปิลมาก จะสามารถลดควอนไทซ์ซึ่งนอยส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Quantizing Noise) ในขบวนการ PCM ได้ การบันทึกสามารถเก็บรายละเอียดของสัญญาณเสียงได้มากด้วยเช่นกัน

### รูปแบบของไฟล์เสียง

ในการบันทึกเสียงพูดเป็นข้อมูลดิจิทัล โดยใช้การรูดเสียง จะทำให้ได้เวฟไฟล์ที่มีรูปแบบของไฟล์ RIFF (Resource International File Format) ซึ่งเป็นรูปแบบมาตรฐานที่ใช้กันอย่างกว้างขวาง ไฟล์ RIFF นั้นมีโครงสร้างเชิงซ้อนคือ มีลักษณะเป็นกลุ่ม (Chunk) ซึ่งภายในกลุ่มนี้ยังประกอบด้วยกลุ่มย่อยซึ่งมีโครงสร้างคล้ายกัน ดังแสดงในรูปที่ 2.7

จากรูปที่ 2.7 จะเห็นว่า กลุ่ม RIFF (RIFF Chunk) จะเป็นกลุ่มใหญ่ที่สุด ซึ่งประกอบด้วยกลุ่มรูปแบบของข้อมูล (Format Data Chunk) และกลุ่มข้อมูลจากการแซมปลิง (Sample Data Chunk) ซึ่งในแต่ละกลุ่มจะมีโครงสร้างที่ประกอบด้วย

1. ชื่อกลุ่ม (CKID : Chunk Identification) เช่น RIFF, fmt\_, data ซึ่งมีกฎกำหนดว่าต้องมี 4 ตัวอักษร (FCC : Four Character Code) ถ้ามี 3 ตัวอักษรก็ต้องเพิ่มช่องว่าง (Blank) เข้าไปอีก 1 ตัวอักษรให้รวมเป็น 4 ตัวอักษรในกรณี fmt\_

2. ตัวเลขบอกขนาดข้อมูลของกลุ่ม (CKSize : Chunk Size) มีหน่วยเป็นไบต์ (Byte) โดยจะไม่รวมไบต์ที่ใช้ไปในการเก็บชื่อกลุ่ม (CKID) แต่จะเก็บตัวเลขบอกขนาดของกลุ่มเองด้วย ตัวเลขบอกขนาดของกลุ่มจะแสดงด้วยข้อมูล 4 ไบต์ โดยไบต์ที่มีความสำคัญต่ำสุด (Least Significant Byte) จะถูกเก็บเข้าไปก่อน

3. ข้อมูลของกลุ่ม (CKData : Chunk Data) มีจำนวน ไบต์เท่ากับตัวเลขที่แสดงในข้อที่ 2 เช่น ในกลุ่มรูปแบบของข้อมูล (Format Data Chunk) จะมีข้อมูลของกลุ่มเป็นรูปแบบในการบันทึกเสียง ยกตัวอย่างเช่น อัตราการแซมปลิง, จำนวนบิตต่อ 1 แซมเปิล เป็นต้น หรือในกลุ่มข้อมูลจากการแซมปลิง (Sample Data Chunk) จะมีข้อมูลของกลุ่มเป็นแอมพลิจูด (Amplitude) ที่ได้จากการแซมปลิง



## 2.9 การวิเคราะห์เสียงพูด (Speech Analysis)

ในการวิเคราะห์เสียงพูดทั้งในโดเมนเวลา (กระทำกับเสียงพูดที่เก็บไว้โดยตรง) และในโดเมนความถี่ (ผ่านการแปลง Fast Fourier) เพื่อหาตัวแทนพารามิเตอร์ในรูปที่เหมาะสมที่สุดสามารถนำไปใช้งานได้ และมีข่าวสารข้อมูลครบถ้วน การวิเคราะห์ในโดเมนเวลาจะถูกจำกัดไว้ได้เฉพาะพารามิเตอร์ง่ายๆ แต่การวิเคราะห์ในโดเมนความถี่จะให้ค่าพารามิเตอร์ที่มีประสิทธิภาพมากกว่า

### 2.9.1 การวิเคราะห์เสียงพูดในช่วงสั้น (Short Time Speech Analysis)

สัญญาณเสียงพูดเป็นสัญญาณที่เปลี่ยนแปลงไปตามเวลาโดยเกิดในลักษณะแบบสุ่ม (Random) แต่ก็ขึ้นกับการควบคุมเสียงของผู้พูดด้วย เพราะเสียงที่เปล่งออกมาในช่วงเวลาหนึ่งนั้น จะขึ้นอยู่กับรูปร่างของท่อกำจรเสียง (Vocal tract) และลักษณะการสั่นของเส้นเสียง (Vocal cord) สัญญาณที่เป็นเสียงพูดจึงเป็นคาบเวลาชั่วขณะ หมายความว่า สัญญาณเสียงพูดมีคาบเวลาที่ในระยะเวลานั้น และมีการเปลี่ยนแปลงในระหว่างเวลานั้น ในการพูดโดยทั่วไปลักษณะการเปลี่ยนแปลงของเสียงพูดอยู่ในช่วงค่าเฉลี่ยประมาณ 80 ms ในการวิเคราะห์เราจะใช้ windows วิเคราะห์สัญญาณเป็นช่วงๆ หรือเรียกว่าการวิเคราะห์เฟรม (Frame) ช่วงเวลาของวินโดว์มีค่าแน่นอนขึ้นอยู่กับอัตราส่วนการสุ่มสัญญาณ (Sampling rate)

#### การวินโดว์ (Windowing)

รูปแบบของการเฉลี่ยโดยปกติมีหลายวิธี เพื่อให้ได้เส้นของพารามิเตอร์ที่เป็นฟังก์ชันของเวลาได้อย่างถูกต้อง

1. วินโดว์จะต้องมีขนาดเล็กพอๆกับลักษณะของคำพูด
2. วินโดว์จะต้องยาวพอที่จะใช้ในการคำนวณหาค่าพารามิเตอร์
3. วินโดว์ที่ดีจะต้องไม่สั้นไปจนข้ามบางช่องคำพูด การวิเคราะห์จะทำได้เป็นคาบๆ ไปซ้ำกันตลอดที่ได้สัญญาณ

เงื่อนไขข้อนี้จะขึ้นกับ Frame Rate มากกว่าขนาดของวินโดว์ โดยปกติ Frame Rate ประมาณ 2 เท่า ของความถี่เพื่อให้สัญญาณซ้อนทับกัน 50% การทำวินโดว์เป็นการคูณสัญญาณเสียงด้วยวินโดว์ที่มีช่วงเวลาจำกัด  $w(n)$  ซึ่งกลุ่มของสัญญาณที่สุ่มมาจะถูกให้น้ำหนักโดยวินโดว์

### 2.9.2 ค่าพารามิเตอร์ในโดเมนความถี่

ค่าพารามิเตอร์ที่สำคัญจำนวนมากถูกพบในโดเมนความถี่ เสียงพูดคำเดียวกันของคนๆ เดียวในแต่ละครั้งเมื่อพิจารณาในโดเมนเวลาจะเห็นผลของความแตกต่างของสัญญาณ แต่เมื่อพิจารณาในโดเมนความถี่จะมีค่าใกล้เคียงกัน ระบบการได้ยินของมนุษย์จะตอบสนองต่อรูปร่าง

ของสัญญาณเสียง หรือขนาดที่กระจายอยู่ในโดเมนความถี่มากกว่าที่เฟสของสัญญาณในโดเมนเวลา ดังนั้นการวิเคราะห์ในโดเมนความถี่จึงดีกว่าวิเคราะห์ในโดเมนเวลา

### การวิเคราะห์ฟูรีเยร์ (Fourier Analysis)

การวิเคราะห์ฟูรีเยร์ช่วงสั้นเป็นเทคนิคการหาความถี่ที่ใช้กันมานานแล้ว การวิเคราะห์ฟูรีเยร์ให้ตัวแทนของสัญญาณเสียงพูดเป็นฟังก์ชันความถี่ในเทอมของขนาดและเฟส เนื่องจากเสียงพูดไม่ได้นิ่งตลอดเวลา ดังนั้นจึงจำเป็นต้องทำการวิเคราะห์ในช่วงสั้นๆ โดยใช้วินโดว์ การแปลงฟูรีเยร์ช่วงสั้นมีสมการว่า

$$X_n(e^{-j\omega}) = \sum x(m)[\exp(-j\omega m)w(n-m)] \quad (2.1)$$

ในการคำนวณต้องใช้ DFT แทนการแปลงฟูรีเยร์แบบต่อเนื่อง โดยใช้ฟังก์ชันวินโดว์ลดข้อมูลแบบไม่ต่อเนื่องทั้งหมดให้เหลือจำนวน  $N$  ตัว ( $N$  คือ ช่วงเวลาหรือขนาดของวินโดว์ที่ใช้ในการแปลง DFT) ข่าวดสารต่างๆ ใน  $X_n(e^{-j\omega})$  จะไม่สูญหายไปจากข้อมูลเดิม ถ้าการแปลงนั้นสัมพันธ์ด้วยความถี่สูงเพียงพอ และวินโดว์  $w(n)$  ไม่มีจุดศูนย์กลางตลอดช่วง  $N$  ตัวแปร  $N$  เป็นตัวแปรที่จะต้องระวังมากเป็นพิเศษในการวิเคราะห์ความถี่ช่วงสั้น ถ้าค่าของ  $N$  ค่าจะทำให้ความละเอียดในโดเมนความถี่จะหายากมาก เพราะจะให้ผลที่ดีในโดเมนเวลา เพราะการเฉลี่ยถูกทำในช่วงสั้นๆเท่านั้น ในทางตรงกันข้าม ถ้า  $N$  มีขนาดใหญ่ จะให้ผลของความละเอียดที่แย่ในโดเมนเวลา แต่จะทำให้โดเมนความถี่มีความละเอียดสูงกว่า

### การแปลงฟาสต์ฟูรีเยร์ (Fast Fourier Transform)

โดยทั่วไปการแปลงฟูรีเยร์ (Discrete Fourier Transform) เป็นการคำนวณที่ใช้เวลาค่อนข้างมาก ลำดับขั้นตอนในการคำนวณ DFT ให้เร็วขึ้น เรียกว่าฟาสต์ฟูรีเยร์ (Fast Fourier Transform หรือ FFT) โดยการแปลง FFT แบ่งได้เป็น 2 ชนิดใหญ่ๆ คือ ชนิดลดทอนทางเวลา (Decimation In Time หรือ DIT) และชนิดลดทอนทางความถี่ (Decimation In Frequency หรือ DIF)

### วิธีลดทอนทางเวลา(Decimation In Time หรือ DIT)

วิธีนี้เป็นการจัดแบ่งกลุ่มของลำดับสัญญาณในโดเมนเวลา  $X(n)$  ขนาด  $N$  จุดออกเป็น 2 ลำดับ สัญญาณขนาด  $N/2$  จุดเท่ากันคือ ลำดับคู่และลำดับคี่ โดยที่ลำดับคู่เกิดจากการเอาลำดับในตำแหน่งคู่มาเรียงกัน ที่เหลือเป็นลำดับคี่ ดังนั้นจะได้

$$\begin{aligned} x_c(m) &= x(2n) & ; \quad m = 0,1,2,\dots,(N/2)-1 \\ x_o(m) &= x(2n+1) & ; \quad m = 0,1,2,\dots,(N/2)-1 \end{aligned} \quad (2.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าให้  $W_N$  เท่ากับ  $\exp(-2j/N)$  จะทำให้การคำนวณ DFT ของลำดับ  $X(n)$  ที่ยาว  $N$  จุด สามารถเขียนใหม่ได้เป็น

$$X(k) = \sum x_e(2m) W_N + \sum x(2m+1) W_N \quad (2.3)$$

โดยที่

$$W_N = \{\exp[j2\pi/N]\} = \exp[j2\pi/N/2] = W_{N/2} \quad (2.4)$$

ซึ่ง  $W_{N/2}$  เป็นค่า  $W$  ลำดับความยาว  $N/2$  จุด สมการ 2.3 สามารถเขียนใหม่ได้เป็น

$$X(k) = \sum x_e(m) W_N + \sum x(m) W_{N/2} \quad (2.5)$$

การนำผลการแปลง DFT ขนาด 2 จุด จำนวน  $N/2$  ภาคมารวมกัน เพื่อให้เป็นการคำนวณ DFT ขนาด  $N$  จุด จะต้องมีหลักเกณฑ์ที่ถูกต้องด้วย จากสมการ 2.5 ถ้าเขียนให้อยู่ในช่วง  $0 \leq k \leq N/2 - 1$  สามารถเขียนใหม่ได้เป็น

$$\begin{aligned} X(k) &= X_e(k) + (W_N) X_o(k) \quad ; 0 \leq k \leq N/2 - 1 \\ &= X_e(k - N/2) + (W_N) X_o(k - N/2) \quad ; 0 \leq k \leq N/2 - 1 \end{aligned} \quad (2.6)$$

เทอม  $W_N$  เรียกว่า ตัวประกอบหมุน (Twiddle Factor) ซึ่งใช้ร่วมกับ DFT ขนาด 2 จุด หรือขนาด  $N/2$  จุด ในการนำมาประกอบเป็น DFT ขนาด  $N$  จุดได้เหมือนเดิม และจากความสัมพันธ์

$$(W)^{k-N/2} = -(W)^k$$

จะได้

$$\begin{aligned} X(k) &= X_1(k) + (W)^k X_2(k) \quad ; 0 \leq k \leq N/2 - 1 \\ &= X_1(k - N/2) + (W)^{k-N/2} X_2(k - N/2) \quad ; N/2 \leq k \leq N/2 - 1 \end{aligned} \quad (2.7)$$

ผลจากสมการนี้สามารถนำไปใช้สร้างหน่วยคำนวณที่เรียกว่า หน่วยผีเสื้อ (Butterfly Unit) โดยมีข้อมูลเข้าคือ  $A$  และ  $B$  และข้อมูลออกคือ  $X$  และ  $Y$  เป็น

$$\begin{aligned} X &= A + (W)^k B \\ Y &= A + (W)^k B \end{aligned} \quad (2.8)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.10 นิวรอลเน็ตเวิร์ค (Neural Network)

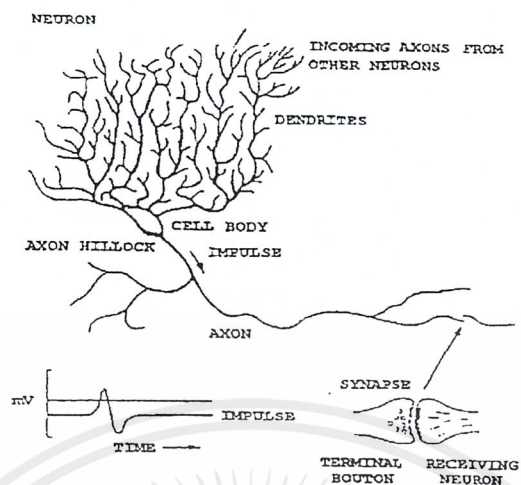
### 2.10.1 ความรู้เบื้องต้นเกี่ยวกับนิวรอลเน็ตเวิร์ค

ในช่วงระยะเวลา 8-9 ปีที่ผ่านมา ในต่างประเทศมีการตื่นตัวในการวิจัยและพัฒนาเกี่ยวกับโครงข่ายประสาทเทียม (Artificial Neural Network : ANN) อย่างกว้างขวางทั้งทางด้านทฤษฎีและการประยุกต์ใช้งาน จนกระทั่งถึงปัจจุบัน ได้มีการประยุกต์นำนิวรอลเน็ตเวิร์คมาใช้อุปกรณ์เครื่องใช้ต่างๆ มากขึ้น เช่น เครื่องมือหาปลา (Sonar) ที่มีความฉลาดมากขึ้น เช่น สามารถบอกได้ว่าฝูงปลาที่กำลังตรวจจับอยู่นั้นเป็นปลาชนิดใด จำนวนเท่าไร เครื่องโทรศัพท์แบบที่สามารถเรียกเลขหมายปลายทางให้อัตโนมัติ (Voice Phone) เพียงยกหูและพูดชื่อของผู้ที่จะติดต่อเท่านั้น เครื่องอ่านตัวอักษร (OCR) ที่สามารถเปลี่ยนภาพตัวอักษรให้เป็นรหัสตัวอักษรแบบแอสกี ระบบนักบินอัตโนมัติ (Auto Pilot Aircraft) ระบบการคาดเดาอนาคตจากข้อมูลในอดีต (Forecasting, Prediction) ฯลฯ ซึ่งเครื่องมือและอุปกรณ์ต่างๆ ที่นำเอา ANN มาใช้ช่วยวิเคราะห์นั้น จะมีความฉลาดมากขึ้นและมีระบบความคิดที่มีการทำงานในลักษณะคล้ายกับมนุษย์

นิวรอลเน็ตเวิร์ค หมายถึง โครงข่ายใยประสาทที่เชื่อมต่อกันระหว่างเซลล์ประสาทจำนวนมากมายมหาศาล มีความสามารถประมวลผลสูงบรรจุอยู่ในสมอง สมองชีวภาพที่เป็นจุดศูนย์กลางการควบคุมกิจกรรมของการดำเนินชีวิต การวิจัยสร้างโครงข่ายประสาทเทียม (Artificial Neural Network) มีแนวคิดเลียนแบบการทำงานของสมองชีวภาพเพื่อกำหนดแนวทางสำหรับการสร้างแบบจำลองขึ้นมา แล้วพยายามสมมติฐานลักษณะการทำงาน โดยจำลองเป็นโมเดลคณิตศาสตร์ที่มีลักษณะเดียวกันแล้วดำเนินการคำนวณ โดยใช้คอมพิวเตอร์

### 2.10.2 นิวรอลเน็ตเวิร์คชีวภาพ

ระบบการคิดของมนุษย์มีโครงสร้างพื้นฐานจากเซลล์สมอง ที่เรียกว่านิวรอล (Neural) เรียงเป็นชั้นจำนวนมากประมาณหมื่นล้านนิวรอล เชื่อมโยงถึงกันประมาณพันล้านล้านจุด แต่ละนิวรอลจะมีคุณลักษณะต่างกันไป แต่จะมีรูปแบบการทำงานคล้ายๆ กันคือ รับค่าเข้ามาประมวลผล และส่งสัญญาณไฟฟ้าเคมีออกไป



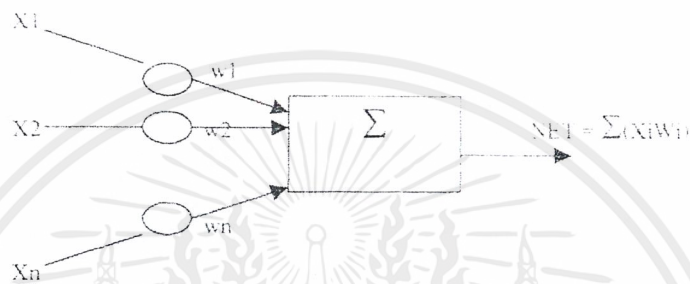
รูปที่ 2.8 ตัวอย่างเซลล์ประสาทชีวภาพ

จากภาพแขนงที่ขยายออกไปยังเซลล์อื่นๆเพื่อรับสัญญาณ เรียกว่า เดนไดรต์ (Dendrites) จะมีจุดเชื่อมต่อกับเซลล์ประสาทอื่นๆ เรียกว่า ซิแนปส์ (synapse) และส่วนที่ใช้ส่งสัญญาณออกไป เรียกว่า แอกซอน (Axon) แต่ละนิวรอนจะมีคุณสมบัติในการเพิ่มขยาย หรือลดทอนความเข้มของสัญญาณที่เข้ามายังเดนไดรต์ของเซลล์ อาจกระตุ้นหรือยับยั้งตัวเซลล์ก็ได้ เนื่องจากเซลล์ประสาทหนึ่งเซลล์มีเดนไดรต์จำนวนมาก ดังนั้นสัญญาณกระตุ้นเดนไดรต์ที่รับเข้ามาจากเซลล์อื่นๆจะถูกนำมารวมกันที่ตัวเซลล์ประสาท ที่เซลล์ประสาทจะมีค่าเทรชโฮลด์ (Threshold) ค่าหนึ่ง หากผลรวมของสัญญาณไฟฟ้าเคมี (Electrochemical) มีค่ามากกว่าค่าเทรชโฮลด์ เซลล์ประสาทจะส่งสัญญาณค่าหนึ่งผ่านทางแอกซอนไปยังนิวรอนอื่นๆต่อไป การจัดเรียงเป็นชั้น (Layer) และลักษณะการเชื่อมโยงระหว่างนิวรอนในสมองนั้นมีการจัดเรียงที่ซับซ้อน สอดคล้องกับหน้าที่การทำงานเฉพาะส่วน มีการเจริญเติบโตเรียนรู้สัมพันธ์กับสิ่งแวดล้อมตลอดเวลา จึงเป็นการยากที่จะสามารถสร้างโมเดลเลียนแบบการทำงานของสมองชีวภาพได้ทั้งหมด ปัจจุบันยังคงทำได้เพียงการจำลองเลียนแบบการทำงานเฉพาะส่วนบางส่วน ของโครงข่ายประสาทเทียม มาใช้เฉพาะกับงานใดงานหนึ่ง และมีขอบเขตจำกัดการใช้งานด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.10.3 โครงข่ายประสาทเทียม (Artificial Neural Network)

การออกแบบสร้างประสาทเทียมมีสมมติฐานจากโครงข่ายประสาทชีวภาพ ซึ่งมีความนำสัญญาณไฟฟ้าเคมีต่างกัน โมดูลโครงข่ายประสาทเทียมจึงต้องมีการถ่วงน้ำหนักก่อนไปใช้งาน เรียกว่า ไซแนปติกเวกท์ ปริมาณข้อมูลจะถูกนำมารวมกัน และตัดสินใจด้วยระดับความสนใจของนิวรอล (Activation level) แล้วส่งไปเอาต์พุตออกไปยังนิวรอลอื่นๆ



รูปที่ 2.9 ไดอะแกรมของนิวรอล 1 หน่วย

จากภาพเป็น ไดอะแกรมที่จำลองจากแนวความคิดของเซลล์สมองทางชีวภาพ สัญญาณอินพุตคือ  $X_1, X_2, \dots, X_n$  จะถูกป้อนเข้ามายังนิวรอล เปรียบได้กับสัญญาณไฟฟ้าเคมีเข้ามายังไซแนปส์ของเซลล์ประสาท ค่าอินพุตเหล่านี้จะคูณด้วยค่าถ่วงน้ำหนัก (weight) ที่มีค่าตั้งแต่ 0.0-1.0 ผลรวมของสัญญาณทั้งหมดจะส่งออกมาโดย

$$\begin{aligned} \text{NET} &= X_1W_1 + X_2W_2 + \dots + X_nW_n \\ &= \sum (X_iW_i) \end{aligned}$$

จากนั้นจะถูกตัดสินใจโดยหน่วยเซลล์ด้วยฟังก์ชันการตัดสินใจ (Activation Function) และได้ค่าเอาต์พุตออกมา

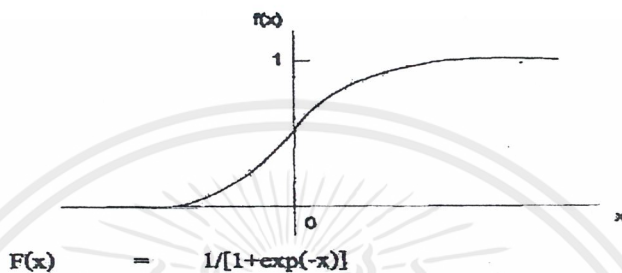
$$\text{OUT} = F(\text{NET})$$

ฟังก์ชันตัดสินใจอาจเป็น hard linear function โดย

$$\begin{aligned} \text{OUT} &= 1 \text{ ถ้า } \text{NET} > T \\ &= 0 \text{ ถ้า } \text{NET} \text{ มีค่าเป็นกรณีอื่น} \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T เป็นค่าคงที่เทรชโฮล (Threshold) หรืออาจเป็นฟังก์ชันอื่นๆที่เลียนแบบคุณสมบัติไม่คงที่ของเซลล์ประสาทได้ดีกว่า และใช้เป็นฟังก์ชันให้กับโครงข่ายทั่วไปได้ ฟังก์ชันตัดสินใจที่นิยมใช้กันทั่วไปคือ สแควชซิ่ง ฟังก์ชัน (Squashing function) ซึ่งมีรูปร่างคล้ายตัว "S" และมีสมการทางคณิตศาสตร์ดังนี้

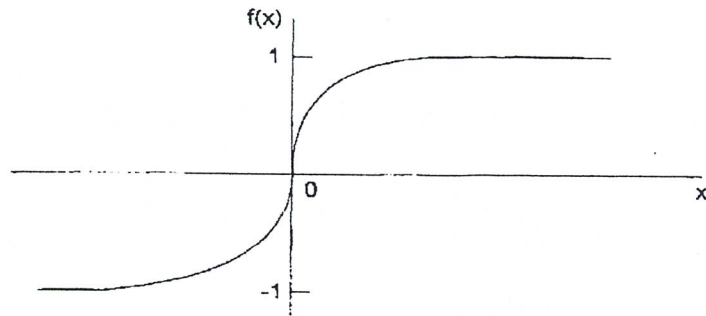


รูปที่ 2.10 ซิกมอยด์ ฟังก์ชัน (Sigmoid function)

การใช้ซิกมอยด์ ฟังก์ชันจะทำให้เทรชโฮลฟังก์ชันมีลักษณะ non linear function ทำให้ได้ค่าอินพุตที่ไวต่อสัญญาณขนาดเล็กๆ และเฉื่อยต่อสัญญาณแรงๆ คือสัญญาณไปทางบวกเล็กน้อยก็จะมีค่าเอาต์พุตใกล้เคียง "1" และสัญญาณเป็นลบค่าต่างๆใกล้เคียง "0" ขณะเดียวกันสัญญาณบวกขนาดใหญ่ก็ยังคงให้ค่าเอาต์พุตใกล้เคียง "1" และสัญญาณลบค่ามากๆก็ยังคงให้เอาต์พุตใกล้เคียง "0" ได้ คือมีคุณสมบัติ non linear gain นั้นเอง ซึ่ง คลอสเบอร์ก (Grossberg,1973) พบว่าสามารถแก้ปัญหา Noise Saturation Dilemma ได้และทำให้นิวรอลสามารถทำงานได้กว้างขวางขึ้น

นอกจากนี้ยังมีฟังก์ชันอื่นๆอีกคือ ไฮเพอร์โบลิค ฟังก์ชัน (Hyperbolic function) ซึ่งมีลักษณะคล้ายกับ logistic function คือ

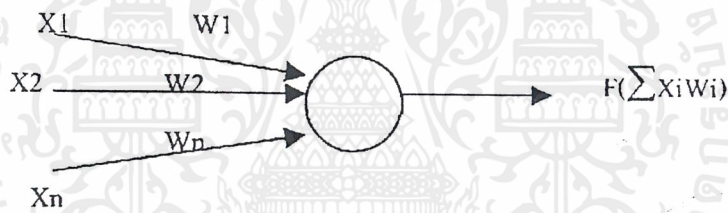
$$OUT = \tanh(X)$$



รูปที่ 2.11 hyperbolic tangent function

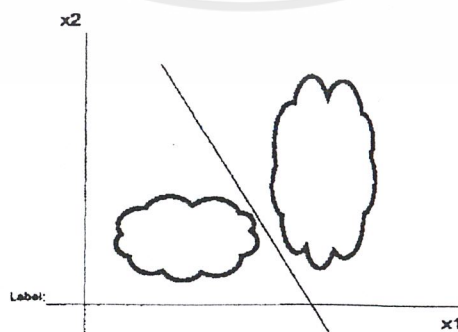
2.10.4 โครงข่ายประสาทเทียมแบบชั้นเดียว (Single Layer Artificial Neural Network)

ซึ่งจะสามารถแก้ปัญหาฟังก์ชันเชิงเส้น (Linear function) ได้เท่านั้น



รูปที่ 2.12 โครงข่ายประสาทเทียมแบบชั้นเดียว

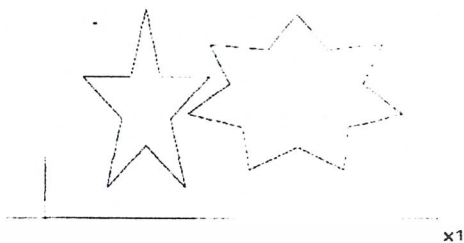
โครงข่ายประสาทแบบชั้นเดียวที่ไม่มีการไหลย้อนกลับของค่าสัญญาณเอาต์พุต จะมีความสามารถในขอบเขตจำกัด คือใช้ตัดสินใจกับปัญหาที่เป็นฟังก์ชันเชิงเส้นเท่านั้น



รูปที่ 2.13 Linear Separable (สามารถแบ่งได้เชิงเส้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

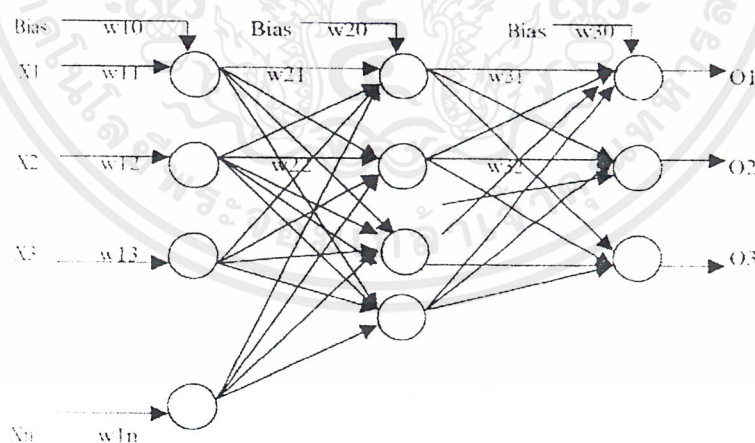
x2



รูปที่ 2.14 Non Linear Separable (ไม่สามารถแบ่งได้เชิงเส้น)

### 2.10.5 โครงข่ายประสาทเทียมแบบหลายชั้น (Multilayer Artificial Neural Network)

ซึ่งประกอบด้วยอินพุทเลเยอร์ (Input Layer), เอาต์พุทเลเยอร์ (Output Layer) และชั้นซ่อน (Hidden Layer) จำนวนตั้งแต่ 1 ชั้นขึ้นไป ทำให้ความสามารถของโครงข่ายดีขึ้น คือสามารถแก้ปัญหาที่ไม่เป็นเชิงเส้นได้ ปัญหาที่มักเกิดขึ้นจากการออกแบบโครงข่ายประสาทแบบหลายชั้นจะเป็นในแง่ของการกำหนดขนาดจำนวนหน่วยในแต่ละชั้น จำนวนชั้นซ่อนที่เหมาะสม และลักษณะการเชื่อมของแต่ละหน่วย แต่ละชั้น



รูปที่ 2.15 โครงข่ายประสาทเทียมแบบหลายชั้น

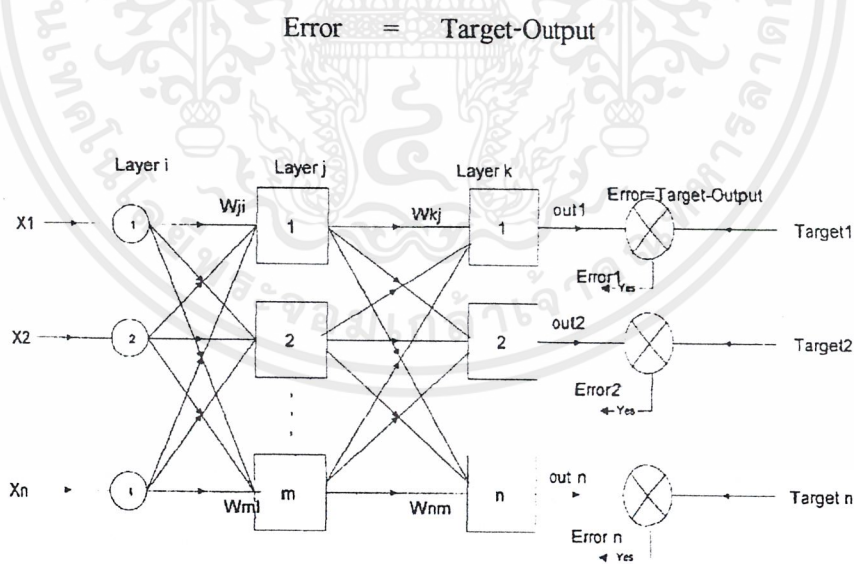
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.10.6 การฝึกสอนให้กับโครงข่าย(Training Algorithm)

เทรนนิ่งอัลกอริทึมถูกจัดเป็น 2 ประเภท คือ แบบควบคุม (Supervise Training) และแบบอิสระ (Unsupervise) โดยเทรนนิ่งแบบควบคุมต้องการข้อมูล 2 ชุดคือ อินพุตกับชุดเป้าหมาย เรียกเทรนนิ่งเพอร์ (Training Pairs) เช่น การเรียนรู้แบบแพร์ย้อนกลับ ส่วนการเทรนนิ่งแบบอิสระสร้างขึ้น โดยเปรียบเทียบการทำงานของสมองที่ไม่จำเป็นต้องมีผู้มาคอยคิดค่าเป้าหมายให้ก่อน โครงข่ายจะรับเพียงค่าอินพุตเข้าไปเท่านั้น เทรนนิ่งอัลกอริทึมจะเปลี่ยนแปลงค่าถ่วงน้ำหนักของโครงข่าย เพื่อสร้างเอาต์พุตที่มั่นคง และเมื่อค่าอินพุตมีการเปลี่ยนแปลงไปเพียงเล็กน้อย โครงข่ายจะยังคงสามารถชี้ได้ว่าเป็นข้อมูลเดิม เนื่องจากเอาต์พุตของโครงข่ายไม่มีการกำหนดมาก่อน ส่วนใหญ่จะถูกแปรรูปให้เข้าใจได้ จึงไม่สามารถใช้ตัดสินใจปัญหาที่มีความยากได้ มักใช้ในางานง่ายๆ เช่น การเปรียบเทียบเอกลักษณ์, รูปภาพ, รูปแบบที่สัมพันธ์กันระหว่างอินพุต-เอาต์พุต

## 2.10.7 ขั้นตอนการเรียนรู้แบบแพร์ย้อนกลับ (Backpropagation Learning)

การฝึกสอนให้กับโครงข่ายประสาทแบบหลายชั้นใช้การเรียนรู้แบบแพร์ย้อนกลับ โดยโครงข่ายจะรับค่าอินพุต X และค่าเป้าหมาย (Target) ไว้ เมื่อคำนวณค่าเอาต์พุตในโครงข่ายแล้ว จะเปรียบเทียบค่าผิดพลาดของเอาต์พุตที่คำนวณได้กับค่าเป้าหมาย



รูปที่ 2.16 โครงข่ายของ Backpropagation Neural Network แบบ 2 ชั้น

ค่าผิดพลาดนี้จะนำไปใช้ปรับค่าถ่วงน้ำหนักของโครงข่ายเพื่อให้สอดคล้องกับค่าเป้าหมาย การเรียนรู้จะทำได้เรื่อยๆจนโครงข่ายสามารถแยกแยะข้อมูลได้

ขั้นตอนการฝึกสอนสามารถสรุปเป็นขั้นๆได้ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. สร้างโครงข่ายประสาทที่มีจำนวนอินพุต และเอาต์พุตเท่ากับจำนวนอินพุต และเอาต์พุตของปัญหา ส่วนจำนวนชั้นซ่อนอาจเริ่มที่ค่ามากพอสมควรก่อน
2. ให้ค่าเริ่มต้นแก่ค่าถ่วงน้ำหนักทุกเส้นในโครงข่าย โดยควรมีค่าในช่วง -1 ถึง 1
3. คำนวณค่า  $\sum X_i W_i = U_i$  สำหรับแต่ละนิวรอน  $j$  ของชั้นซ่อน
4. คำนวณค่ากระตุ้น (Activation) สำหรับแต่ละนิวรอน  $j$  ของชั้นซ่อน

$$Y_j = f(U_j)$$

โดยฟังก์ชันที่นิยมใช้คือ sigmoid function  $f(x) = 1/[1+\exp(-x)]$

5. คำนวณค่า  $\sum Y_i W_i = V_i$  สำหรับแต่ละนิวรอน  $k$  ของชั้นเอาต์พุต
6. คำนวณค่ากระตุ้น (Activation) ของแต่ละนิวรอน  $k$

$$\text{OUT} = Z_k = f(V_k)$$

7. นำค่า  $Z_k$  มาเปรียบเทียบกับเป้าหมายถ้าค่าผิดพลาดที่คำนวณได้น้อยกว่าระดับที่กำหนดไว้ก็จบการเรียนรู้ มิฉะนั้นก็ทำต่อไป

8. คำนวณค่าปรับน้ำหนักสำหรับแต่ละค่าถ่วงน้ำหนัก

8.1 สำหรับเส้นเชื่อมชั้นซ่อนกับชั้นเอาต์พุต

$$\Delta W_{jk} = \alpha \delta_k Y_j$$

$$\text{โดยที่ } \delta_k = (\text{Target } k - Z_k) f'(V_k)$$

8.2 สำหรับเส้นเชื่อมชั้นอินพุตกับชั้นซ่อน

$$\Delta W_{ij} = \alpha \delta_j X_i$$

$$\text{โดยที่ } \delta_j = \sum \delta_k W_{jk} f'(U_j)$$

9. ปรับค่าน้ำหนักเส้นเชื่อมจากนิวรอน  $r$  ไปนิวรอน  $s$  จะได้

$$W_{rs}(\text{new}) = W_{rs}(\text{old}) + \Delta W_{rs}$$

10. กลับไปทำข้อ 3

ค่า  $\alpha$  : อัตราการเรียนรู้ (Learning rate) มีค่าในช่วง 0 ถึง 1

การปรับความเร็วในการเรียนรู้อาจใช้การปรับค่า Gain ของ sigmoid function ในส่วนดีกรีของ exponential การปรับอัตราการเรียนรู้ ขนาดจำนวนหน่วยในชั้นซ่อน และการเพิ่ม โมเมนต์คัมแพกเตอร์เพื่อคำนวณทิศทาง การปรับค่าถ่วงน้ำหนัก นอกจากนี้ยังเลือกใช้อัตราการเรียนรู้เป็นฟังก์ชันที่มีความเหมาะสมได้อีกด้วย

ฟังก์ชัน sigmoid เป็นที่นิยมในการใช้เทรนนิ่งเนื่องจากมีความใกล้เคียงกับฟังก์ชันขั้นบันได และสามารถคำนวณค่าอนุพันธ์ได้ง่ายอีกด้วย โดยอยู่ในรูปนิพจน์ของตัวฟังก์ชันเอง

$$f'(x) = f(x)[1-f(x)]$$

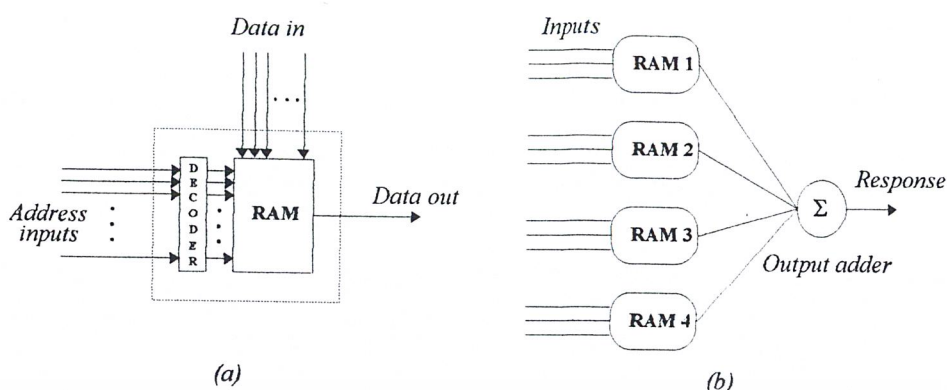
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.10.8 Weightless Neural Networks

Weightless models คือ RAMที่เป็นพื้นที่สำหรับเก็บข้อมูลการกระตุ้นของอินพุทและผลตอบสนองต่อความสัมพันธ์ใน look-up tables ไปในรูปแบบของปฏิบัติการทางตรรกศาสตร์ของนิรอล ค้นแบบของความคิดนี้ถูกเสนอ โดย Bledsoe และ Browning (1959) ซึ่งรู้จักกันในชื่อของกระบวนการจำแนกข้อมูลแบบ n-tuple (n-tuple recognition process) tuple แต่ละยูนิตจะรับข้อมูลเพียงส่วนเล็กๆของข้อมูลทั้งหมดและตอบสนองโดยผลที่ได้ของแต่ละยูนิตเป็นอิสระต่อกัน การจำแนกข้อมูลใน n-tuple machine สามารถทำได้โดยเปรียบเทียบลักษณะที่คล้ายกันระหว่างรูปแบบของข้อมูลที่เก็บภายในความจำของคอมพิวเตอร์ซึ่งมาจาก tuples ทั้งหมดกับรูปแบบของข้อมูลที่ป้อนเข้าไป โดยให้ลักษณะที่คล้ายคลึงกันออกมาในรูปของคะแนน ซึ่งข้อได้เปรียบของเทคนิคนี้คือข้อมูลหลายตัวสามารถมีผลตอบสนองร่วมกันใน tuple หนึ่งยูนิตและสามารถแยกรูปแบบของข้อมูลแบบไม่เป็นเชิงเส้น(non-linearly) ออกมาได้

ระหว่างทศวรรษที่ 70 การพัฒนาเทคโนโลยีของวงจรรวมหรือ (Integrated Circuit Technology) สามารถสร้าง Random Access Memories (RAMs) ได้ซึ่งเป็นการนำไปสู่การค้นพบนิรอลเน็ตเวิร์คที่ใช้หน่วยความจำแบบ RAMs เป็นที่เก็บข้อมูลของ n-tuple machines โดย Aleksander และ Stonham (1979) หลังจากนั้นหลักการเดียวกันนี้ถูกนำไปใช้ในการออกแบบ WISARD (Aleksander, Thomas and Bowden 1984, Aleksander 1984)

เมื่อพิจารณาการใช้นิรอลเน็ตเวิร์คแบบที่ใช้RAMแล้วพบว่า มีข้อดีในการเรียนรู้ algorithm เนื่องจากเป็น โหนดแต่ละ โหนดอิสระต่อกันระหว่างการอ้างอิงข้อมูลจาก look-up table เว้นแต่ในส่วนของ weighted models ในส่วนที่สองซึ่งน้ำหนักของข้อมูลต้องขึ้นกับการเปลี่ยนแปลงในส่วนแรก ส่วนข้อดีอื่น ๆ นั้นคือฮาร์ดแวร์ซึ่งประกอบด้วย RAMs มีราคาถูกลงและสามารถหาได้ง่ายโดยบล็อกไดอะแกรมพื้นฐานของนิรอลเน็ตเวิร์คแบบที่ใช้ RAM นี้สามารถแสดงได้ดังรูปที่ 2.17 (a)



รูปที่ 2.17 (a) A RAM-based neuron (b) 4 RAMs discriminators with 3 inputs each

โมเดลของนิวรอนแบบไร้น้ำหนัก (weightless neural model หรือ WISARD) คือระบบที่มีการป้อนไปข้างหน้าแบบชั้นเดียวที่ประกอบไปด้วยเซตสัญลักษณ์การแบ่งแยกคลาสของ discriminators รูปแบบของข้อมูลตัวแบ่งแยกทั้งหมดจะเริ่มค้นหาที่เซตของศูนย์และค่อยๆเปลี่ยนไปตามการป้อนข้อมูล (training) ระหว่างการป้อนข้อมูลรูปแบบของข้อมูลแต่ละอันก็จะถูกเก็บอยู่แค่ใน RAM ของการตอบสนองการแบ่งแยกหลังจากการป้อนข้อมูลเสร็จสิ้นก็จะได้ผลการแบ่งแยกของแต่ละคลาส การจำแนกข้อมูล (recognition) ก็จะเริ่มโดยป้อนรูปแบบของข้อมูลที่ไม่ทราบค่าให้กับ discriminators ทุกๆ ตัว เพื่อหาผลตอบสนองที่มีค่ามากที่สุด โดยหลักการทั่วไปของการคิดคือไม่นำมาคิดแค่เพียงแต่ละ โหนดแต่คิดทั้งระดับของเน็ตเวิร์ค

เนื่องจากการทำงานเป็นเส้นตรงของ WISARD จะมีความคลุมเครือระหว่างสถานะ 0 ที่เกิดขึ้นระหว่างการเก็บรูปแบบของข้อมูลและสถานะ 0 ที่จุดเริ่มต้นก่อนจะเกิดการป้อนข้อมูลหรือ training เพื่อแก้ปัญหาข้อนี้ Kan และ Aleksander (1988) ได้นำเสนอการใช้ Probabilistic Logic Node (PLN) ซึ่งแต่ละ โหนดจะยังคงมีเอาต์พุตเป็นเลขฐานสองแต่จะมีสถานะการทำงานเป็น 3 ระดับ โดยสถานะที่เพิ่มขึ้นมาก็คือ u-state ซึ่งจะแทนตรงส่วนที่ค่าเริ่มต้นหรือในช่วงที่ข้อมูลยังไม่ถูกเก็บ ซึ่งโอกาสที่เอาต์พุตของ โหนดจะเป็น 1 และ 0 จะมีค่า 50% ใน recalling mode

เมื่อการที่ข้อมูลจะเข้าไปได้มีเพียงแค่การป้อนข้อมูล (trained) สำหรับการประยุกต์การเรียนรู้ข้อมูล แต่ใน PLNs ข้อมูลจะมาจาก delay เมื่อเซตของการป้อนข้อมูลมีค่าน้อยเช่น 2 หรือ 3 รูปแบบ เนื่องจากบนตำแหน่งส่วนใหญ่จะยังคงเหลืออยู่ภายใน u-states. อุปสรรคอีกอย่างหนึ่งที่ต้องระวังก็คือการ generalisation ยังแสดงผลออกมาเป็นระดับต่างๆ เนื่องจากปัญหานี้ Aleksander (1990a) ได้เสนอเทคนิคที่จะไม่เห็นอินพุตเป็นลักษณะระดับของ โหนดเราเรียกวิธีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นี้ว่า spreading process ซึ่งจะกล่าวต่อไปในบทอื่นๆ อัลกอริทึมของมันคือระบุตำแหน่งของ u-state ที่มาจากเน็ตเวิร์คข้างเคียงที่ใกล้ที่สุดโดยหลักของ Hamming distance หากข้อมูลไม่ตรงกันหรือเกิดความขัดแย้งระหว่างเน็ตเวิร์คที่ใกล้ที่สุดกันเอง ตำแหน่งของ u-state. จะไม่เปลี่ยนแปลง การ generalisation ของเน็ตเวิร์คจะขึ้นกับรัศมีของการกระจายชั้นของแต่ละโหนดของเน็ตเวิร์ค ถ้ารัศมีมีค่าเท่ากับผลรวมจำนวนของอินพุตที่เป็นเลขฐานสองหมายความว่าเน็ตเวิร์คจะมีการกระจายเต็มที่เราเรียกเน็ตเวิร์คที่มีคุณสมบัติเช่นนี้ว่า Generalising RAMs หรือ (GRAMs)

ความจุของการ generalisation ใน GRAMs สามารถหาได้โดยจำนวนของรูปแบบของการเรียนรู้ข้อมูล (training patterns) เนื่องจากการกระจายของชั้นของเน็ตเวิร์คจะมีค่าน้อยลงเมื่อมีจำนวนของรูปแบบเพิ่มขึ้น นอกจากนี้ยังมีอัลกอริทึมของการกระจายหรือ spreading ในแบบอื่นๆ อีกเช่น the Combined Generalisation Algorithm (CGA) ค้นพบโดย Aleksander Clarke และ Braga (1994) โดยเป็นวิธีที่ผสมผสานกันระหว่างการเรียนรู้แบบ Hebbian และ Weightless

### 2.10.9 The General Neural Unit (GNU)

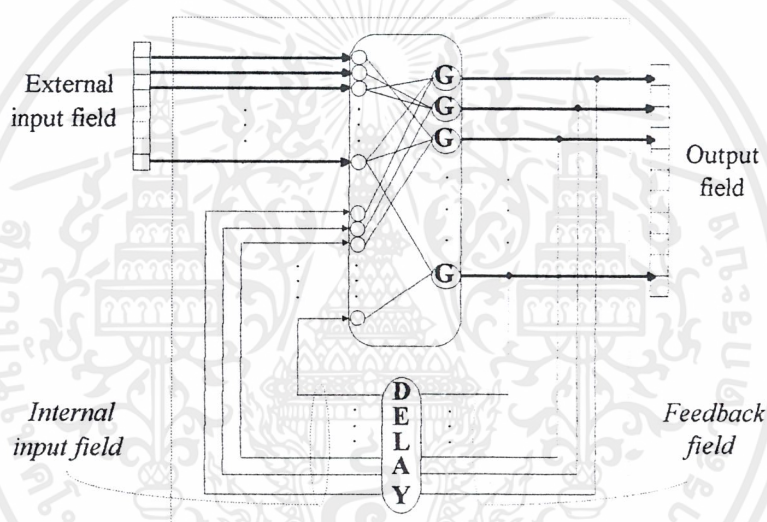
The General Neural Unit (GNU) จะเกิดขึ้นเมื่อมีการเชื่อมโยงโดยใช้ GRAM-based neurons ในเน็ตเวิร์ค โมเดล GNU สามารถที่จะ mapping ได้ทั้งแบบอัตโนมัติและแบบ hetero-associative mode (Zurada 1992) กำหนดให้แต่ละโหนดมีจำนวนรวมของอินพุตเป็น I ประกอบด้วยจำนวนอินพุตที่ได้จากการสุ่ม N จำนวน ค่อยอยู่กับบิตจำนวน W บิตของ External input field และ F แทนอินพุตที่มาจากกลุ่มซึ่งค้อยอยู่กับ Feedback input Field จำนวน K บิต สามารถเขียนความสัมพันธ์ได้ดังนี้

$$I = N + F .$$

GNU จะมีคุณสมบัติเป็น fully connected หากอัตราส่วน  $N/W$  และ  $F/K$  มีค่าเท่ากับ 1 และจะมีคุณสมบัติเป็น partially connected เมื่ออัตราส่วนของจำนวนดังกล่าวมีค่าน้อยกว่า 1 เราสามารถสรุปพารามิเตอร์หลักของ GNU ได้ดังตารางที่ 2.1 และโคอะแกรมของรูปแบบการทำงานดังรูปที่ 2.18

$K$	The number of nodes of a GNU
$W$	The number of external inputs
$I$	Total number of inputs to each neuron
$N$	The number of inputs connected from the external field
$F$	The number of inputs connected from the feedback field

ตารางที่ 2.1 Main parameters of a GNU



รูปที่ 2.18 A schematic view of a basic GNU

การศึกษา GNU ในรูปแบบของ Boolean network ซึ่งทำการวิจัยโดย Wong และ Sherrington (1988, 1989) ส่วนงานวิจัยของ Lucy (1991) แสดงให้เห็นว่าการที่ GNU มีการเชื่อมต่อในแบบ fully connected auto-associative mode หรือ ( $W = 0$ ) จะสามารถเก็บข้อมูลได้เป็นจำนวน  $2^K$  รูปแบบ ผลลัพธ์ที่ดีที่สุดของการดำเนินงานซึ่งศึกษาโดย Aleksander และ Morton (1991) โดยกำหนดให้จำนวนของการเชื่อมต่อมีค่าระหว่างจำนวนอินพุตกับจำนวน feedback หรือ ( $N/W = F/K$ ). การคาดคะเนของความซับซ้อน GNU ทำการวิจัยโดย Braga (1993; 1994). ส่วนโปรแกรมที่ทำการจำลอง GNUs กับข้อมูลจริงๆมีชื่อว่า MAGNUS (Multi Automata General Neural Units) ร่วมกันพัฒนาโดย Aleksander, Evans และ Penney (1993)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.10.10 การทำงานของ GNUs

เราสามารถอธิบาย GNU โดยแบ่งการทำงานเป็นส่วนหนึ่งของ GRAMs ได้เป็น 3 ขั้นตอน คือ storage spread และ retrieval process ในการทำงานแบบ storage process จะเป็นการ mapping ของเน็ตเวิร์คระหว่างอินพุท (ทั้ง external inputs และ internal representations) และ เอาท์พุท โดยการสร้างสถานะเสถียร (stable state) ภายใน Boolean space สถานะของ n-dimensional Boolean space สำหรับ โมเดลของ single GNU สามารถแสดงได้โดย two-dimensional Boolean transition table ในแบบพื้นฐาน

การทำงานแบบ storage process มีหลักการคือความสัมพันธ์ของอินพุทและเอาท์พุทใน ทุกๆ โหนดของเน็ตเวิร์ค ภายใน GRAM ซึ่งจะเริ่มต้นจาก u-states จากนั้นก็จะมีกรอกแบบ ตำแหน่งของสถานะของเอาท์พุท รูปแบบลำดับของการทำงานในแบบ storing นั้น เป็นการ representations ภายใน เชื่อม ไปถึงลำดับการทำงานเอาท์พุทซึ่งเปรียบเสมือนอ่าง

storage ต่อไปจะเป็นกระบวนการ spreading ซึ่งช่วยในการลดจำนวน delay ของเน็ตเวิร์ค เนื่องจากมีหลายๆตำแหน่งยังคงเหลืออยู่ภายใน u-states ซึ่งการกระจายจะนำที่ว่างในแต่ละ โหนดแยกออกเป็นอิสระต่อกัน การกระจายใน GNUs จะคล้ายคลึงกับแนวคิดในการสร้างพลังงานที่น้อยที่สุดของ Hopfield network โดยมีความแตกต่างกันคือ auto-associative ใน Hopfield model จะมีความเป็นไปได้ที่จะเกิดความผิดพลาดในการป้อนกลับ เนื่องจากความสัมพันธ์ ระหว่างข้อมูลที่เข้ามาใหม่และข้อมูลที่ถูเก็บไว้ก่อนหน้านั้น ขณะที่ โมเดลของ GNUs มีการ เชื่อมต่อแบบ จะไม่เกิดความผิดพลาดระหว่างความสัมพันธ์ของรูปแบบของข้อมูล นอกจากนั้น การกระจายใน GNUs จะสร้าง attractors ด้วยการเปลี่ยนแปลงในแบบ steeper ผลที่ได้คือจะใช้ เวลาในการดำเนินการน้อยกว่า อย่างไรก็ตามบางครั้งก็เกิดความผิดพลาดในการสร้าง attractor หากการสร้างนั้นมีขนาดกว้างเกินไปเนื่องจากรูปแบบของข้อมูลที่เก็บมีค่าน้อยเกินไป นอกจาก นั้นการ storage ของความสัมพันธ์ระหว่างรูปแบบของข้อมูล สามารถเกิดความขัดแย้งกันได้ ใน โมเดลของ GNU ที่มีการเชื่อมโยงกันน้อยเกินไป reprocess จะทำงานได้ดีเมื่อทั่วทั้งเน็ตเวิร์คใน ทุกๆ โหนดสามารถเปลี่ยนแปลงข้อมูลได้พร้อมๆกันภายในเวลาเดียวกัน ดังเช่นการแสดงภาพจะ เกิดขึ้นพร้อมๆกันกับอินพุทและ ในช่วงเริ่มต้นของสถานะภายใน จำนวนที่เพิ่มขึ้นของ attractors จะมีผลต่อการทำงานเน็ตเวิร์คให้มีความรวดเร็วและต่อเนื่องมากขึ้น อย่างไรก็ตาม ถ้าหากจำนวน ของ attractors มากเกินไปเมื่อเทียบกับความจุของเน็ตเวิร์คก็อาจจะทำให้การเรียกกลับ (recall) ล้มเหลวก็เป็นได้ เนื่องจากมีสถานะที่ไม่เสถียรมากเกินไปทำให้เกิดความขัดแย้งกัน จำนวนรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบการเก็บข้อมูลจะต้องมีน้อยกว่าความสามารถของเน็ตเวิร์คเพื่อการป้อนกลับที่มีความถูกต้องสูง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบ

การทำงานของปฏิญาณนิพนธ์นี้ประกอบด้วย 2 ส่วนหลัก คือ

1. ส่วนปริ้โปรเซสซิ่ง (Pre-Processing)
2. ส่วนการแยกแยะและตัดสินใจของนิรอรล

#### 3.1 ส่วนปริ้โปรเซสซิ่ง (Pre-Processing)

##### - ส่วนการอัดเสียง

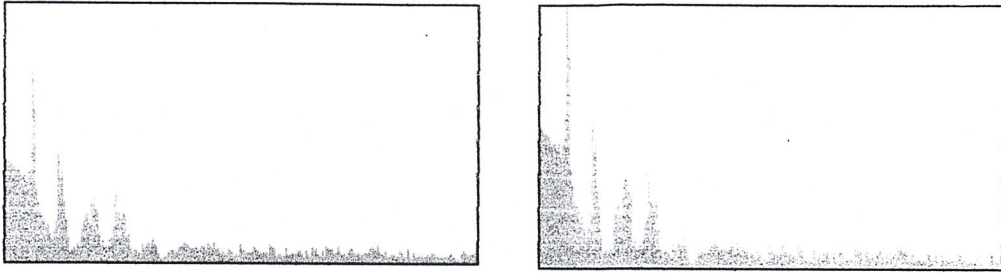
ส่วนนี้จะทำหน้าที่รับเสียงเข้ามาจากไมโครโฟน ผ่านทางการ์เคเสียง โดยใช้สัญญาณเสียงพูดคำโคค คำว่า ศูนย์, หนึ่ง, สอง, ..., เก็บเสียงที่ได้จะถูกเก็บในรูปของเวฟไฟล์ (\*. Wav) ซึ่งเป็นการอัดเสียงแบบ โมโน ด้วยความถี่สุ่ม 11 kHz ขนาด 8 บิต

##### - ส่วนการแปลง FFT

การแปลง FFT จะใช้คอมโปเนนท์ TFastFourier ของ Borland Delphi 5 ซึ่งเป็นแซร์แวร์ ดาวน์โหลดมาจาก [www.lohninger.com](http://www.lohninger.com) โดยกำหนดขนาดของข้อมูลให้มีขนาด 4096 ตัว ถ้าข้อมูลมีขนาดมากกว่า 4096 ตัวก็ทำการรับข้อมูลมาแค่ 4096 ตัว แต่ถ้าจำนวนข้อมูลน้อยกว่า 4096 ตัว ก็ทำการเติม 0 ให้ครบ 4096 ตัว ก่อนที่จะแปลง FFT

##### - ส่วนการนอร์มาไลซ์

การนอร์มาไลซ์ คือ การทำข้อมูลให้อยู่ในมาตราส่วนเดียวกัน โดยถ้าข้อมูลใดมีขนาดใหญ่มากกว่ากรอบข้อมูลที่กำหนดไว้ก็จะทำการลดขนาดลงเท่ากับที่กำหนดไว้ (รายละเอียดของข้อมูลยังคงเหมือนเดิม) แต่ถ้าข้อมูลใดมีขนาดเล็กกว่ากรอบข้อมูลที่กำหนดไว้ก็จะทำการเพิ่มขนาดขึ้นให้เท่ากับที่กำหนดไว้ ดังรูปที่ 3.1 โดยนำข้อมูลที่ผ่านมาการแปลง FFT แล้วมาทำการนอร์มาไลซ์ทางแอมพลิจูดทุกๆข้อมูล



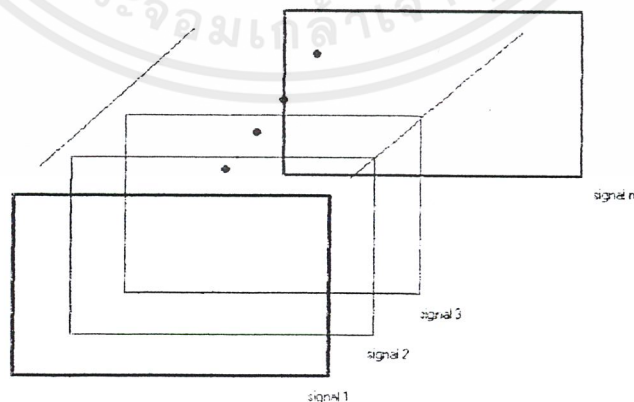
รูปที่ 3.1 แสดงรูปสัญญาณที่ถูกรบกวนมาไลซ์

## 3.2 ส่วนการแยกแยะและตัดสินใจ

### 3.2.1 ขั้นตอนการฝึกสอน

ส่วนนี้จะใช้คิจัดลนิวรอล ชนิด Weightless Neural Networks ในการแยกแยะและตัดสินใจเสียงพูดที่เข้ามาว่าเป็นเสียงของคำใดใน 10 คำ คือ ศูนย์, หนึ่ง, สอง, ..., เก้า ซึ่งขั้นตอนการฝึกสอนให้กับนิวรอล จะใช้สัญญาณเสียงของผู้พูดคนเดียว เสียงละ 10 ครั้ง รวมเป็น 100 เสียง โดยมีขั้นตอนดังนี้

1. นำข้อมูลเสียงที่ทำการแปลงฟูริเยร์อย่างรวดเร็วมาหาจุดที่นิวรอลควรจับคือจุดที่มีความแตกต่างมากที่สุด โดยนำข้อมูลเสียงที่ทำการแปลงฟูริเยร์อย่างรวดเร็วของทุกข้อมูลมาซ้อนกัน ดังรูปที่ 3.2



รูปที่ 3.2 แสดงการนำข้อมูลเสียงมาวางซ้อนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทำการหาค่าความแตกต่างของทุกพิกัดของทุกข้อมูล วิธีการนี้เรียกว่าการ Optimize มีหลักการดังนี้

### หลักการ Optimize ข้อมูล

เราจะหาตำแหน่งของข้อมูลที่เราจะให้นิวรอลจับคือ ตำแหน่งที่มีข้อมูลที่แตกต่างกันมากที่สุดเอาไปเก็บในนิวรอลแต่ละตัว

ตารางที่ 3.1 แสดงการ Optimize ข้อมูล

ข้อมูล	ค่าแตกต่างของข้อมูล
1 0 1 0 1 1 1 0 0 1 0 1 0 0 0 1 0 1 0 1	14
1 0 1 0 0 0 1 0 1 0 0 1 1 1 0 1 0 1 0 1	14
0 0 0 1 0 1 0 1 0 0 0 1 1 1 0 1 0 1 1 1	11
1 1 1 0 1 0 1 1 1 0 0 1 0 1 0 0 0 1 1 1	10
0 1 0 1 0 0 0 0 1 0 1 0 0 1 1 1 0 1 0 1	12
1 1 0 1 1 0 0 1 1 1 1 1 0 0 1 0 1 1 0 1	10
0 1 0 0 1 0 0 1 1 1 1 0 1 0 1 0 1 0 1 0	14
0 0 1 0 1 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1	7
0 1 0 1 0 1 1 1 0 1 0 1 0 1 0 1 1 1 0 1	15

จากตัวอย่างเราจะนับค่าความต่างของข้อมูลจาก 1 เป็น 0 และจาก 0 เป็น 1 แล้วเก็บค่าดังกล่าวไว้ เสร็จแล้วก็คัดเลือกตำแหน่งที่การเปลี่ยนแปลงมากที่สุดเท่ากับจำนวนขาของนิวรอลเพื่อจะเป็นตำแหน่งที่จะไว้เก็บข้อมูลต่อไป

2. เมื่อได้ค่าความแตกต่างของข้อมูลทุกข้อมูลในแต่ละพิกัดก็จะทำการสุ่มเลือกแต่ละพิกัดมาเป็นแต่ละขาของนิวรอลแต่ละตัว โดยจะทำการสุ่มตามหลักสถิติโดยใช้ สมการเกาส์เซียน (Gaussian Equation) แสดงดังรูป 3.3

$$f(x) = ce^{-(x-\mu)^2/2\sigma^2} \quad ; \quad c = 1/\sigma\sqrt{2\pi}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

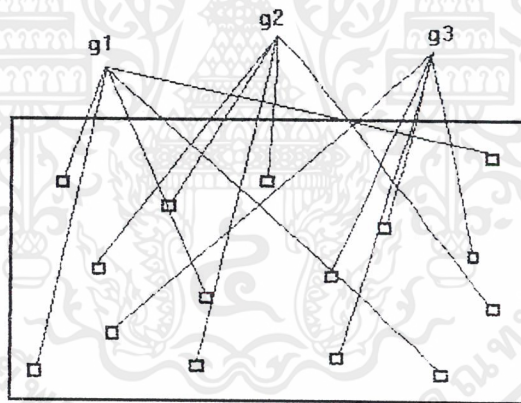


รูปที่ 3.3 แสดงกราฟของสมการเกาส์เซียน

และมีหลักการคำนวณจำนวนขาของนิวรอลโดยมีค่าน้อยตามสมการ

$$\text{จำนวนขา} = \log_2 (\text{จำนวนข้อมูลที่ฝึกสอน})$$

และทำการสุ่มเลือกพิกัด ดังรูป 3.4 หลังจากได้แต่ละพิกัดที่นิวรอลแต่ละตัวจะจับแล้วก็นำค่าแต่ละพิกัดมาเก็บใส่ Grams ของนิวรอลแต่ละตัวของแต่ละข้อมูล



รูปที่ 3.4 แสดงการสุ่มจับพิกัดของขานิวรอลแต่ละตัว

### 3.2.2 ขั้นตอนการทดสอบ

เริ่มจากโหลดเสียงที่จะนำมาทดสอบ ที่ผ่านกระบวนการปริโปรเซสซิ่งแล้ว โดยมีขั้นตอนการแยกแยะดังต่อไปนี้

1. นำข้อมูลเสียงทดสอบที่ผ่านกระบวนการปริโปรเซสซิ่งแล้ว ทำการเก็บค่าพิกัดที่พิกัดเดียวกับข้อมูลที่เราสอนนิวรอล
2. เก็บค่าตามพิกัดไว้ใน Grams ของข้อมูลทดสอบ
3. นำ Grams ของข้อมูลทดสอบที่ได้ไปเทียบกับ Grams ของข้อมูลที่เราสอนให้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิรอรล โดยเราจะหาว่าแต่ละนิรอรลแตกต่างมากน้อยเพียงใด เราเรียกวิธีนี้ว่า Hamming Distance มีหลักการดังนี้

#### หลักการหา Hamming Distance

การหา Hamming Distance คือ การหาค่าความแตกต่างของข้อมูล ซึ่งเราจะหา Hamming Distance ระหว่าง Grams ข้อมูลที่นำมาทดสอบกับข้อมูลที่อยู่ใน Grams ของข้อมูลที่ฝึกสอนแต่ละข้อมูล

ตารางที่ 3.2 แสดงการหา Hamming Distance

ข้อมูล		ค่า Hamming Distance
ทดสอบ	เทียบ	
1010101110	0101010001	10
1010101110	0101101011	6
1010101110	0101000101	8
1010101110	0111010101	8
1010101110	0111010111	7
1010101110	0101010111	8
1010101110	0001100011	6
1010101110	0010111010	3
1010101110	1011110001	6
1010101110	1101011101	7

เมื่อทราบค่า Hamming Distance ของแต่ละข้อมูลแล้ว ก็ต้องเลือกค่าที่มีค่าน้อยที่สุด หมายถึง ข้อมูลที่มีความแตกต่างน้อยที่สุดนั่นเอง

4. พอทราบค่าความแตกต่างของนิรอรลแต่ละตัว ให้นิรอรลแต่ละตัวของทุกข้อมูลให้คะแนน โดยข้อมูลที่สอนให้นิรอรลใดแตกต่างกับข้อมูลที่นำมาทดสอบน้อยที่สุด ก็จะทำให้คะแนน โดยมีหลักการให้คะแนนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### หลักการให้คะแนน (Score)

เราจะทำการให้คะแนนแก่นิวรอลแต่ละตัว โดยการเปรียบเทียบค่า Hamming Distance ของนิวรอลแต่ละตัว ถ้านิวรอลตัวใดมีค่า Hamming Distance น้อยที่สุด (แสดงว่าใกล้เคียงกับข้อมูลที่ทดสอบมากที่สุด) ก็จะให้คะแนนแก่นิวรอลตัวนั้น

ตารางที่ 3.3 แสดงการให้คะแนนข้อมูล

Neural	ค่า Hamming Distance ข้อมูล		คะแนน	
	1	2	1	2
$G_1$	5	6	1	0
$G_2$	2	6	1	0
$G_3$	6	6	1	1
$G_4$	4	1	0	1
$G_5$	5	2	0	1
$G_6$	7	5	0	1
$G_7$	1	4	1	0
$G_8$	2	7	1	0
$G_9$	3	8	1	0
$G_{10}$	2	2	1	1
คะแนนรวม			7	5

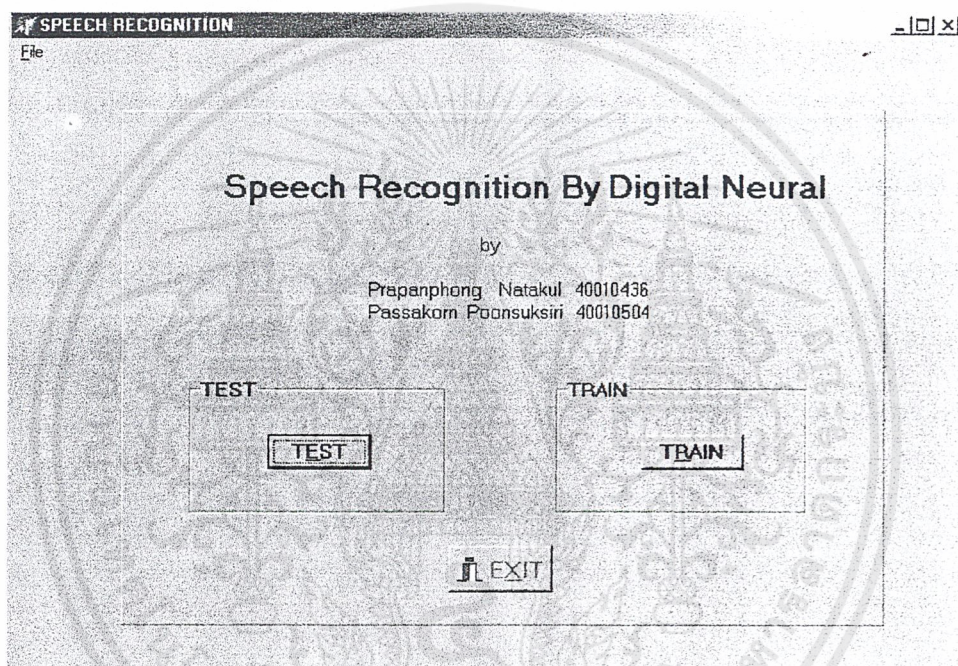
เมื่อนิวรอลให้คะแนนครบทุกตัวแล้ว ข้อมูลเสียงใดมีคะแนนมากที่สุด ก็จะแสดงว่าข้อมูลที่ทดสอบนั้น คือข้อมูลที่มีคะแนนสูงที่สุดนั่นเอง

## บทที่ 4

### การดำเนินการและทดลอง

#### 4.1 การดำเนินการ

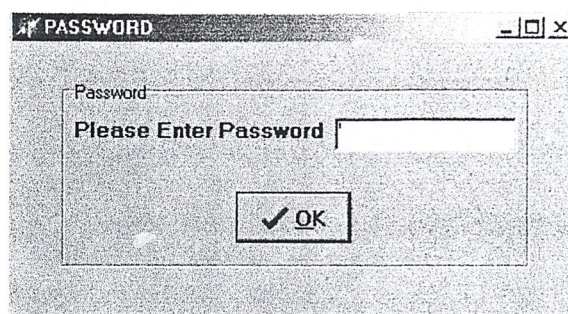
ในการออกแบบ โปรแกรมการทำงาน เมื่อเริ่มเปิดโปรแกรมขึ้นมาจะแสดงหน้าจอเป็นเมนูหลักสำหรับการเลือกที่จะฝึกสอนหรือทดสอบ ดังรูปที่ 4.1



รูปที่ 4.1 แสดงหน้าจอของเมนูหลัก

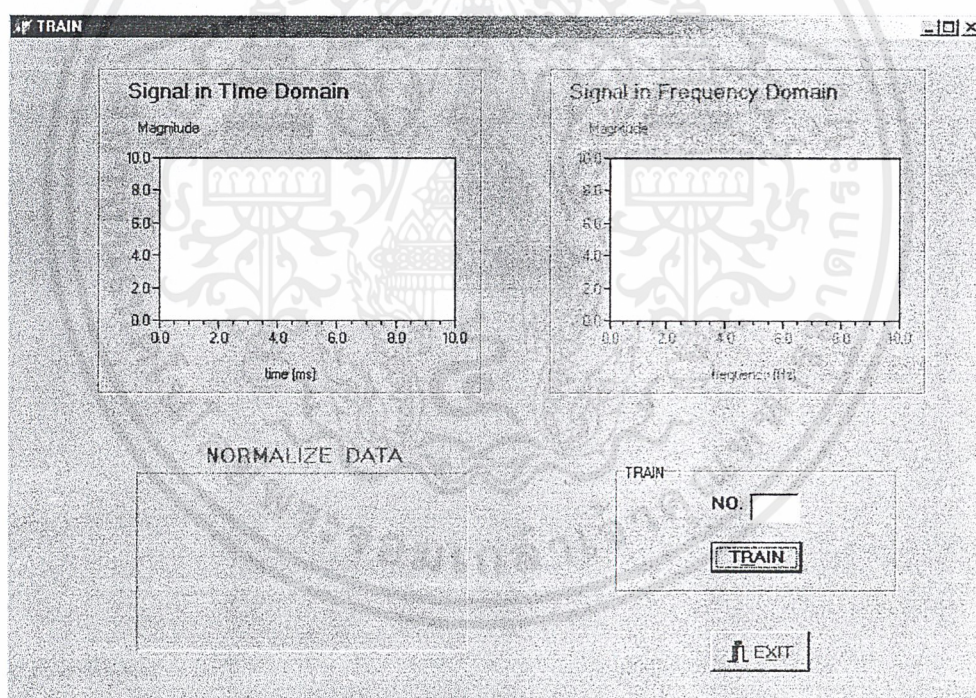
##### 4.1.1 การฝึกสอน

จากเมนูหลักดังรูปที่ 4.1 ให้เลือกโหมดการฝึกสอน โดยคลิกไปที่ปุ่ม TRAIN จากนั้นโปรแกรมจะให้ใส่ Password เพื่อตรวจสอบผู้ใช้ที่จะนำข้อมูลเสียง ไปฝึกสอนให้กับนิรอล ในกรณีที่น่าโปรแกรม ไปประยุกต์ใช้งานด้านอื่นๆ ดังรูปที่ 4.2



รูปที่ 4.2 แสดงรูปแบบการใส่ Password ของ โปรแกรม

เมื่อ Password ถูกต้อง โปรแกรมจะเข้าสู่การฝึกสอน โดยมีรูปแบบหน้าจอการฝึกสอน ดังรูปที่ 4.3



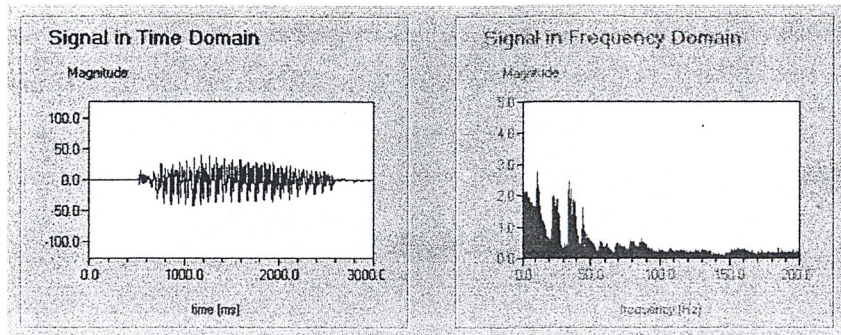
รูปที่ 4.3 แสดงหน้าจอการฝึกสอนของ โปรแกรม

เมื่อคลิกปุ่ม TRAIN โปรแกรมจะทำการฝึกสอนให้กับนิรอรอด โดยนิรอรอดจะเก็บข้อมูลที่ได้จากการสอนเป็นไฟล์

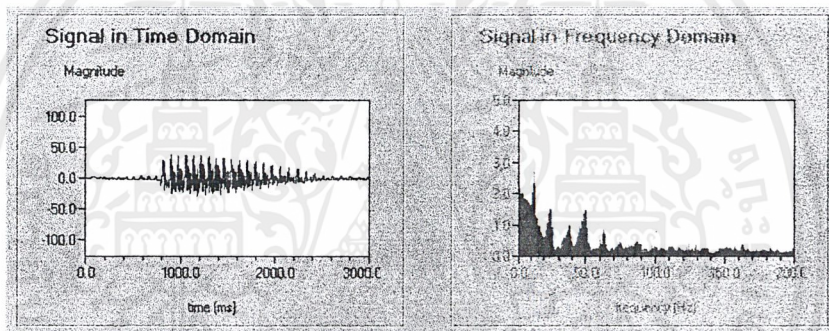
สัญญาณเสียงที่ใช้ในการฝึกสอนให้กับนิรอรอดเป็นเสียงพูดคำว่า “ศูนย์” ถึงคำว่า “เก้า”

โดยแสดงรูปสัญญาณเสียงต่างๆ ในโดเมนเวลา และ โดเมนความถี่ที่ได้จากการแปลง FFT ดังรูป

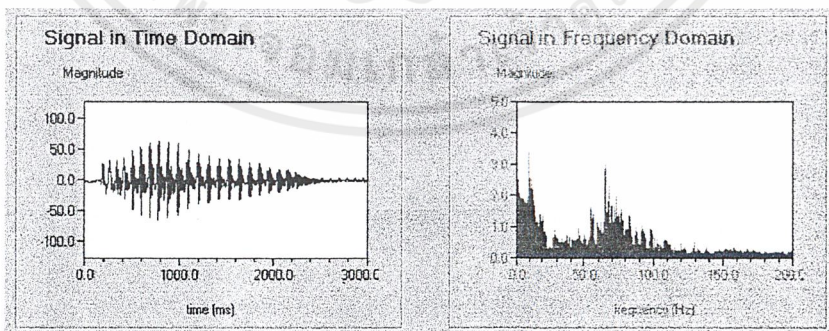
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงสัญญาณเสียงพูด “ศูนย์”

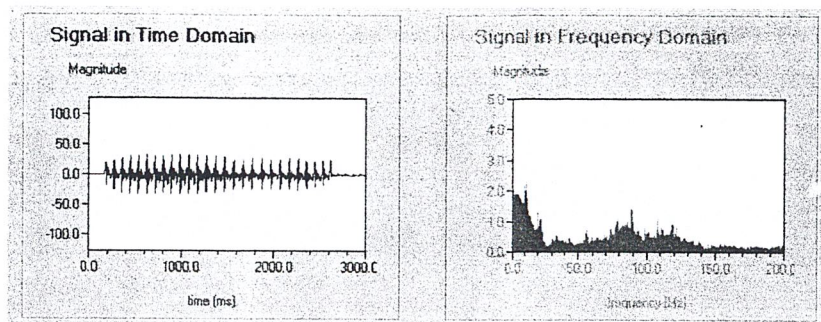


รูปที่ 4.5 แสดงสัญญาณเสียงพูด “หนึ่ง”

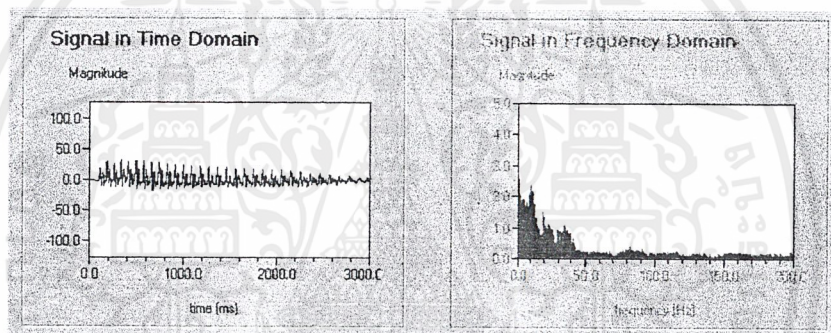


รูปที่ 4.6 แสดงสัญญาณเสียงพูด “สอง”

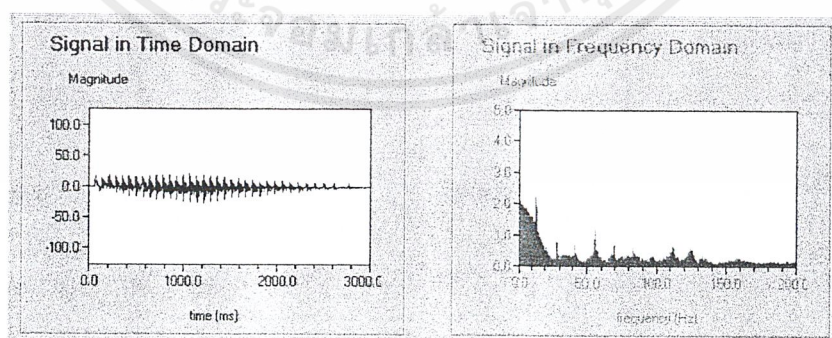
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงสัญญาณเสียงพูด “สาม”

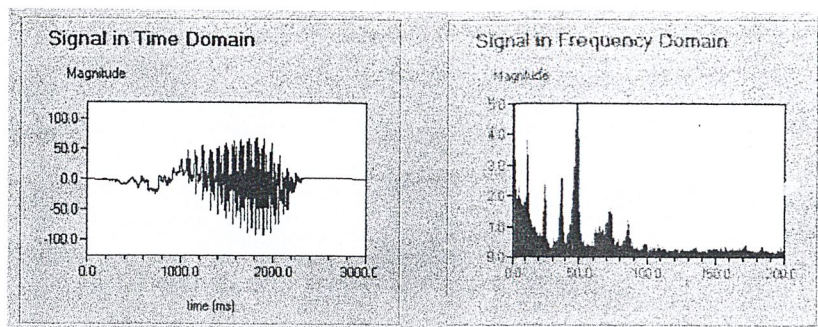


รูปที่ 4.8 แสดงสัญญาณเสียงพูด “สี่”

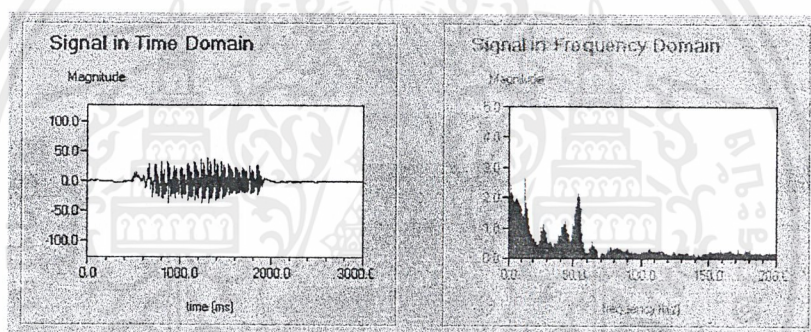


รูปที่ 4.9 แสดงสัญญาณเสียงพูด “ห้า”

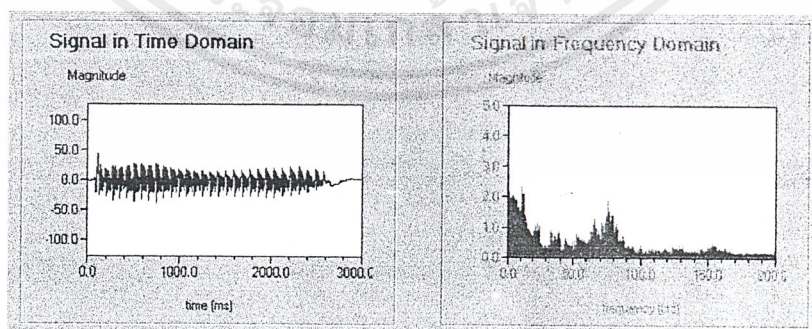
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 แสดงสัญญาณเสียงพูด “ห”

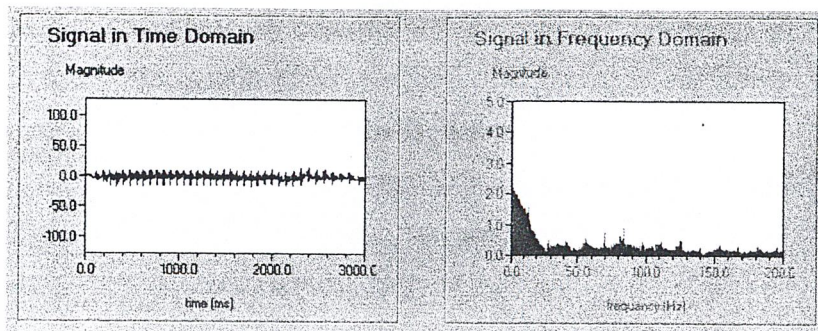


รูปที่ 4.11 แสดงสัญญาณเสียงพูด “เจ็ด”



รูปที่ 4.12 แสดงสัญญาณเสียงพูด “แปด”

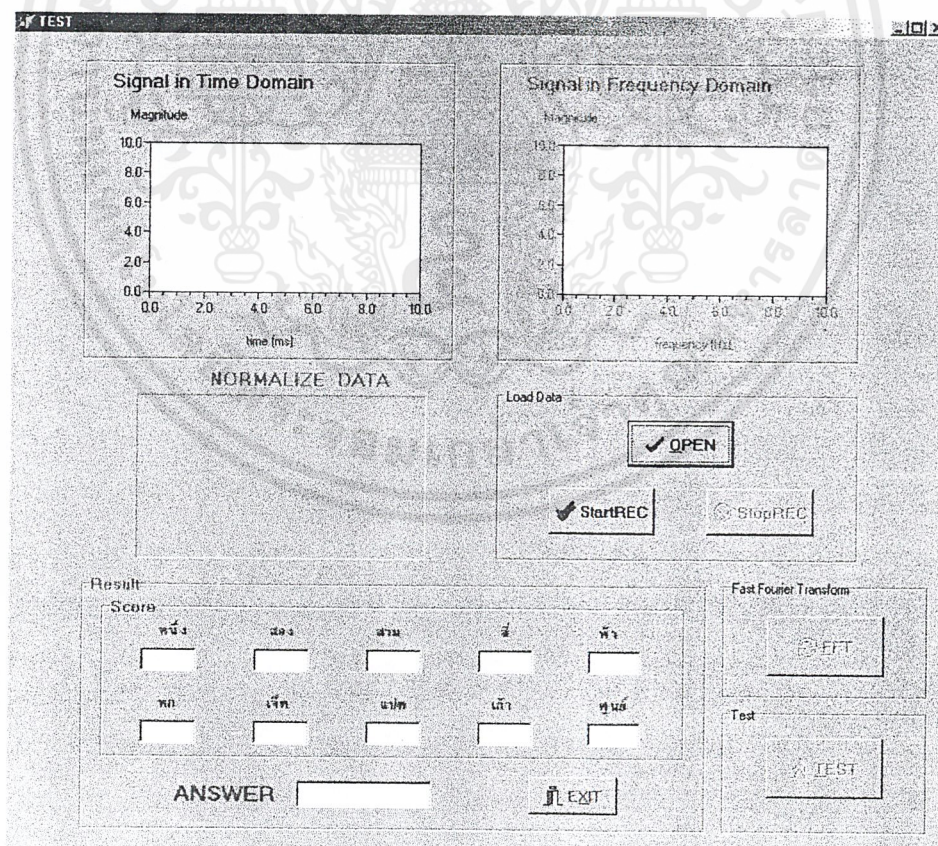
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แสดงสัญญาณเสียงพูด “เก้า”

#### 4.1.2 การทดสอบ

จากเมนูหลัก จะเริ่มทดสอบเสียงโดยคลิกที่ปุ่ม TEST โปรแกรมจะแสดงหน้าจอส่วนการทดสอบดังรูปที่ 4.14

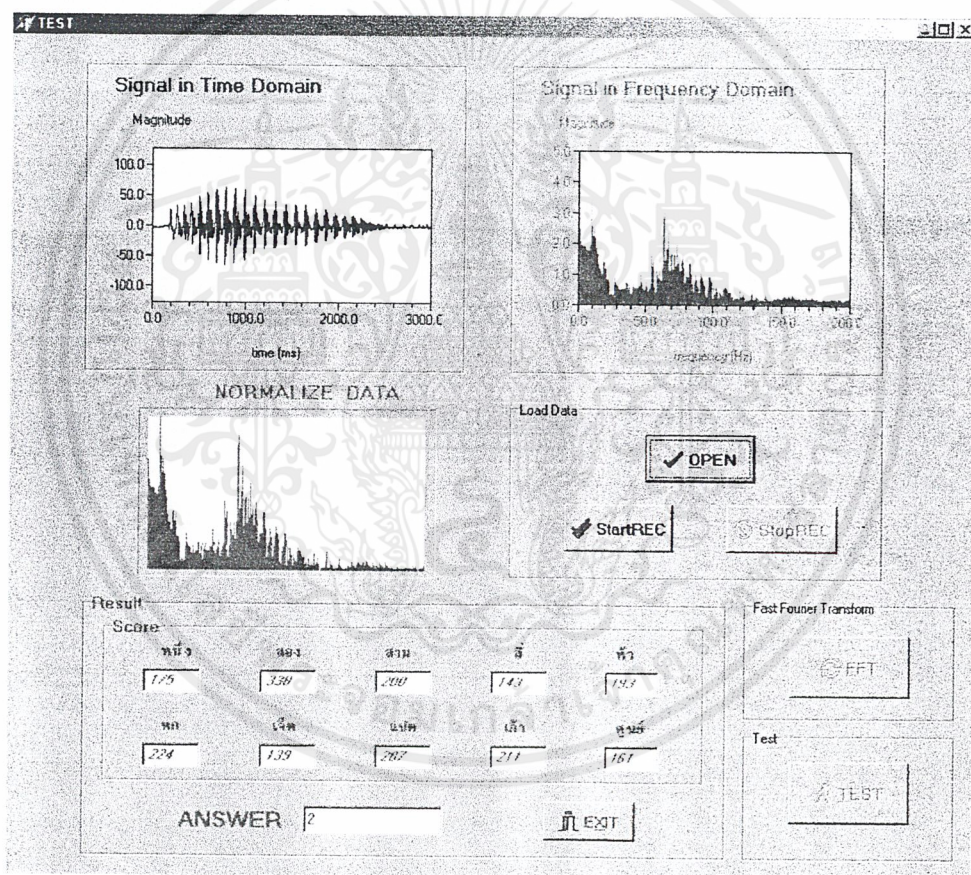


รูปที่ 4.14 แสดงหน้าจอของส่วนทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการโหลดข้อมูลเสียงที่จะทดสอบ โดยคลิกที่ปุ่ม OPEN แล้วเลือกสัญญาณเสียงที่ต้องการ จะ ได้รูปสัญญาณเสียงแสดงใน โดเมนเวลา ทำการแปลง FFT โดยคลิกที่ปุ่ม FFT จะ ได้สัญญาณเสียงแสดงใน โดเมนความถี่

หลังจากนั้นจะทำการทดสอบ โดยคลิกที่ปุ่ม TEST สัญญาณเสียงที่ทดสอบจะเข้าสู่กระบวนการแยกแยะและรู้จำของนิวรรต นิวรรตทุกตัวของทุกข้อมูลเสียงที่ทำการฝึกสอน จะทำการโหวตคะแนนให้กับสัญญาณเสียงที่นำมาทดสอบว่าตรงกับเสียงที่ถูกฝึกสอนเสียงใด ผลลัพธ์ที่ได้คือ เสียงที่มีคะแนนสูงสุด แสดงดังรูป 4.15



รูปที่ 4.15 แสดงหน้าจอผลลัพธ์ที่ได้จากการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 การทดลอง

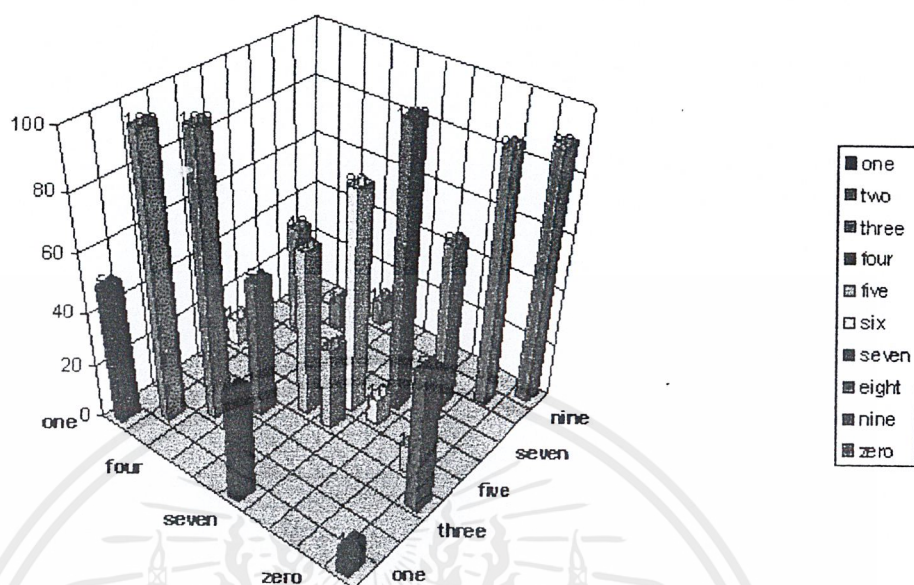
วิธีการทดลองเริ่มจากนำข้อมูลเสียง เสียงละ 10 ข้อมูลที่ไม่ได้ถูกฝึกสอนมาทดสอบกับ  
นิวรอล ได้ผลการทดลองดังนี้

ตารางที่ 4.1 ผลการทดลอง

เสียง ทดสอบ	เอาท์พุท										% ความ ถูกต้อง
	หนึ่ง	สอง	สาม	สี่	ห้า	หก	เจ็ด	แปด	เก้า	ศูนย์	
หนึ่ง	5					4				1	50
สอง		10									100
สาม			10								100
สี่				5						5	50
ห้า					6	3				1	60
หก	1					8	1				80
เจ็ด							10				100
แปด		4						6			60
เก้า			1							9	90
ศูนย์				1						9	90

เปอร์เซ็นต์ความถูกต้องเฉลี่ย = 78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 กราฟแสดงผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปและวิจารณ์ผลการทดลอง

#### 5.1 สรุปผลการทดลอง

1. ผลการรู้จำอยู่ในระดับปานกลาง
2. เสียงของคำว่า สอง สาม เจ็ด มีผลการรู้จำสูงที่สุด เนื่องจากลักษณะของสัญญาณในโดเมนความถี่มีลักษณะที่แตกต่างจากเสียงอื่นมาก
3. ถ้าใช้ตัวอย่างเสียงในการทดลองมากๆ จะทำให้อัตราการรู้จำสูงขึ้น แต่ก็ใช้เวลาในการฝึกสอนนานขึ้น
4. เสียงที่ใช้ในการฝึกสอนจะต้องมีความหลากหลาย เพื่อเพิ่มความยืดหยุ่นในการรู้จำ แต่เสียงของคำเดียวกันจะต้องไม่แตกต่างกันมากนัก เพราะจะทำให้เกิดความผิดพลาด

#### 5.2 วิจารณ์ผลการทดลอง

ในการทดสอบการแยกแยะและรู้จำเสียงพูดจะขึ้นอยู่กับปัจจัยต่างๆ ดังนี้

1. ความสามารถในการหาคุณลักษณะที่แตกต่างกันของเสียงที่ใช้ในการทดลองของกระบวนการปริ-โปรเซสซึ่ง ถ้าสามารถทำให้เห็นความแตกต่างของเสียง ได้มากเท่าไร อัตราการรู้จำก็จะมีมากขึ้นเท่านั้น ซึ่งอาจเพิ่มการรู้จำได้โดยเปลี่ยนพารามิเตอร์ในแกนความถี่ใหม่ ให้สอดคล้องกับลักษณะเด่นของคำที่ใช้
2. ลักษณะการเปล่งเสียงของผู้ที่ใช้ฝึกสอน ว่าสามารถเปล่งเสียงของคำแต่ละคำที่ต่างกัน ได้ต่างกันมากน้อยเพียงใด ถ้าเปล่งเสียง ได้ต่างกันมากอัตราการรู้จำก็จะมากขึ้น
3. เสียงที่ใช้ในการฝึกสอน ถ้าเสียงที่เป็นคำเดียวกันมีความแตกต่างกันน้อย ในการฝึกสอนจะได้ค่าผิดพลาดลดลง เมื่อทดสอบด้วยเสียงเดิมที่ใช้ในการฝึกสอนจะได้ค่าของเอาท์พุทของคำที่ถูกต้องสูง แต่ถ้าทดสอบด้วยเสียงที่แตกต่างจากเสียงที่ใช้ในการฝึกสอนมาก ค่าผิดพลาดก็จะมากขึ้น ดังนั้นเพื่อความถูกต้องของเอาท์พุท จึงควรใช้เสียงในการฝึกสอนที่มีความแตกต่างกันมากพอสมควรเพื่อครอบคลุมการเปลี่ยนแปลงในการเปล่งเสียงของคำเดียวกันได้

### 5.3 แนวทางการพัฒนา

1. เลือกว่าพารามิเตอร์ใหม่ที่สามารถแสดงคุณลักษณะของเสียงพูดมนุษย์เสียงต่างๆ ได้ชัดเจนมากขึ้น
2. พัฒนาอัลกอริทึมในส่วนของการแยกแยะและรู้จำข้อมูลของนิรอรตให้ดีขึ้น
3. พัฒนาโปรแกรมให้สามารถประมวลผลแบบ Real Time เพื่อไปใช้งานจริงได้สะดวกมากขึ้น
4. พัฒนาให้มีการรู้จำเสียงพูดแบบต่อเนื่องเพื่อไปใช้งานได้หลากหลายมากขึ้น
5. นำโปรแกรมไปประยุกต์ใช้กับงานด้านต่างๆ เช่น การนำไปใช้งานด้านระบบการรักษาความปลอดภัย, การควบคุมอุปกรณ์ต่างๆ โดยใช้เสียง เช่น โทรศัพท์ เป็นต้น ฯลฯ

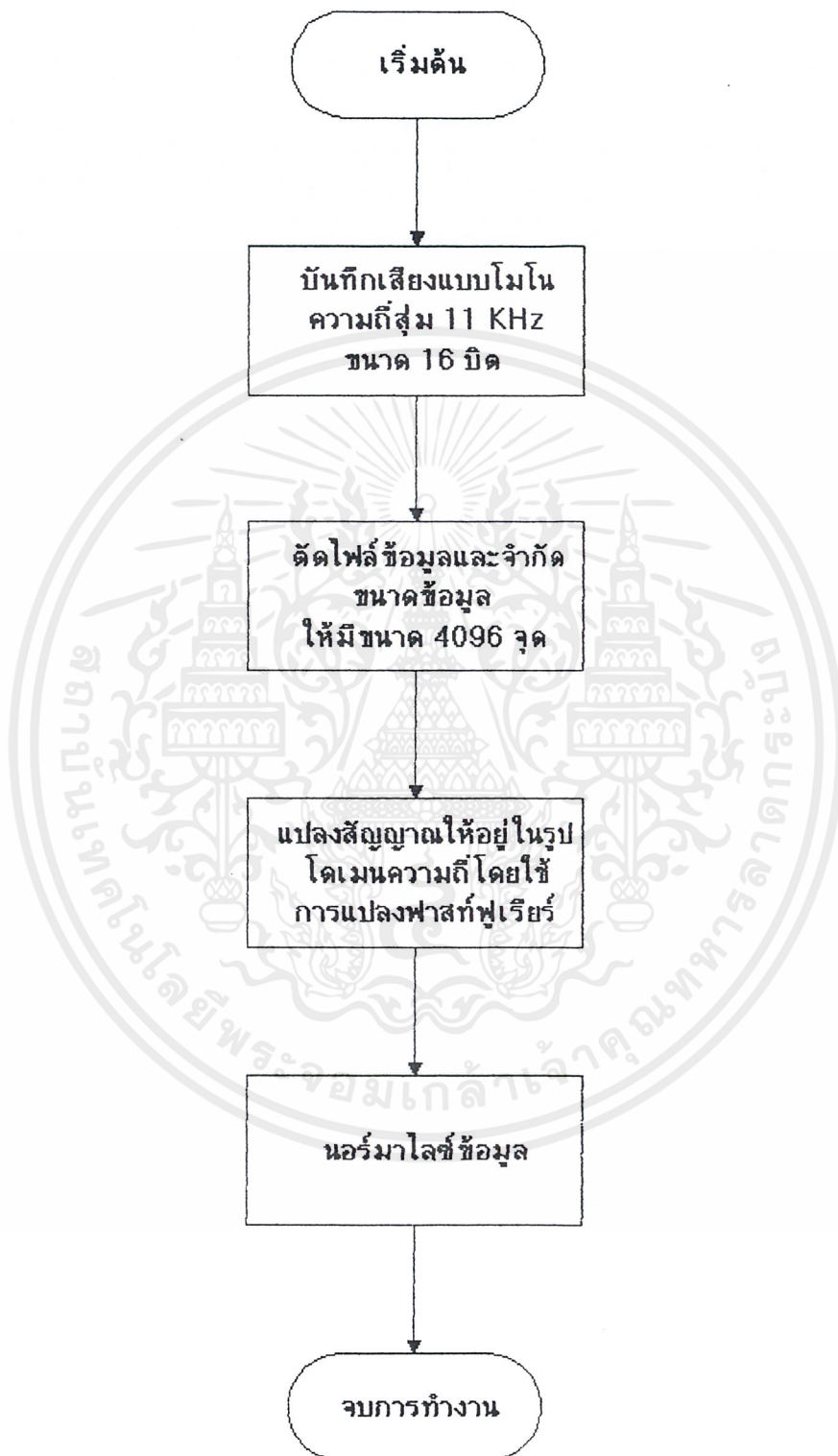


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



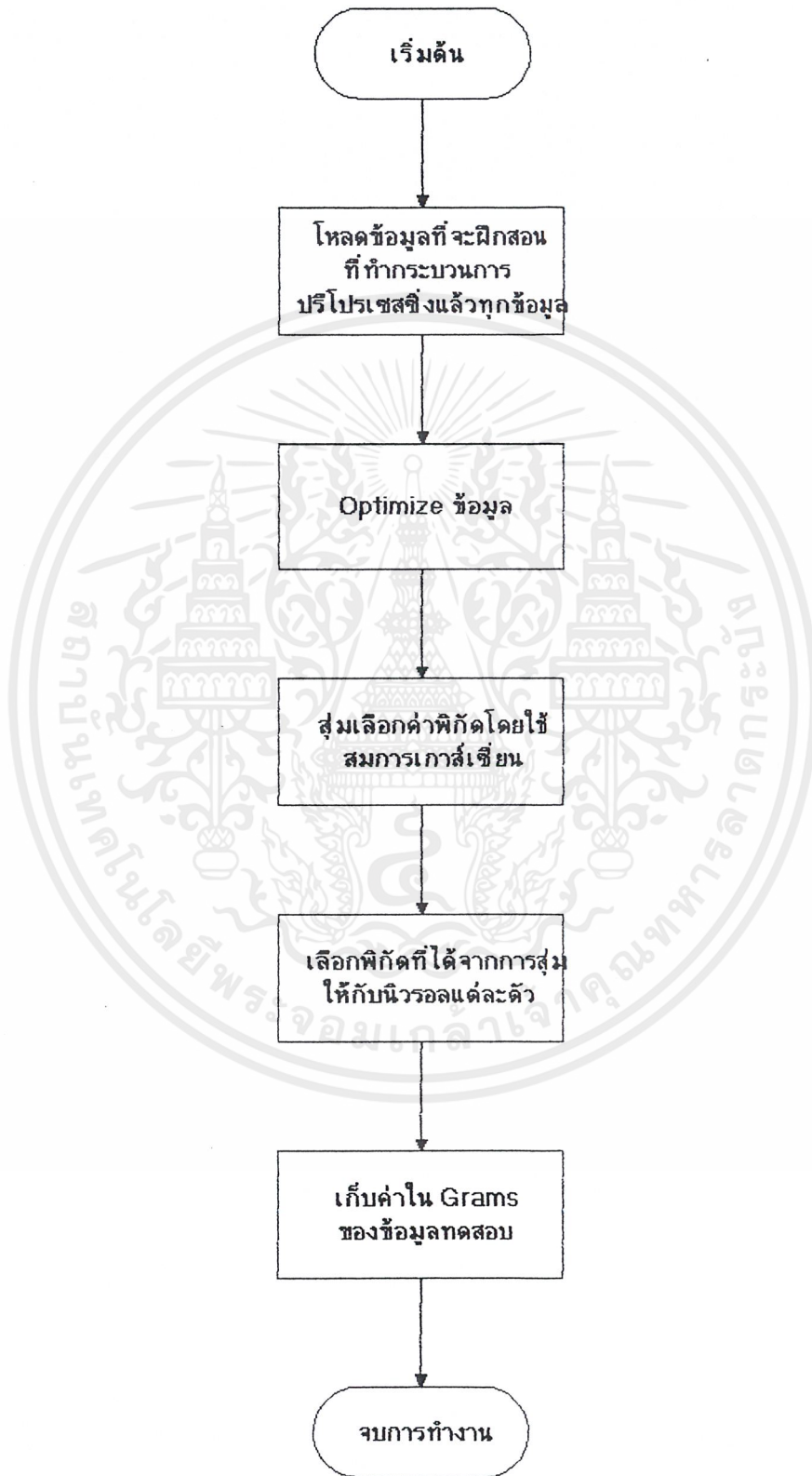
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Flowchart แสดงการทำงานของส่วน Pre-Processing



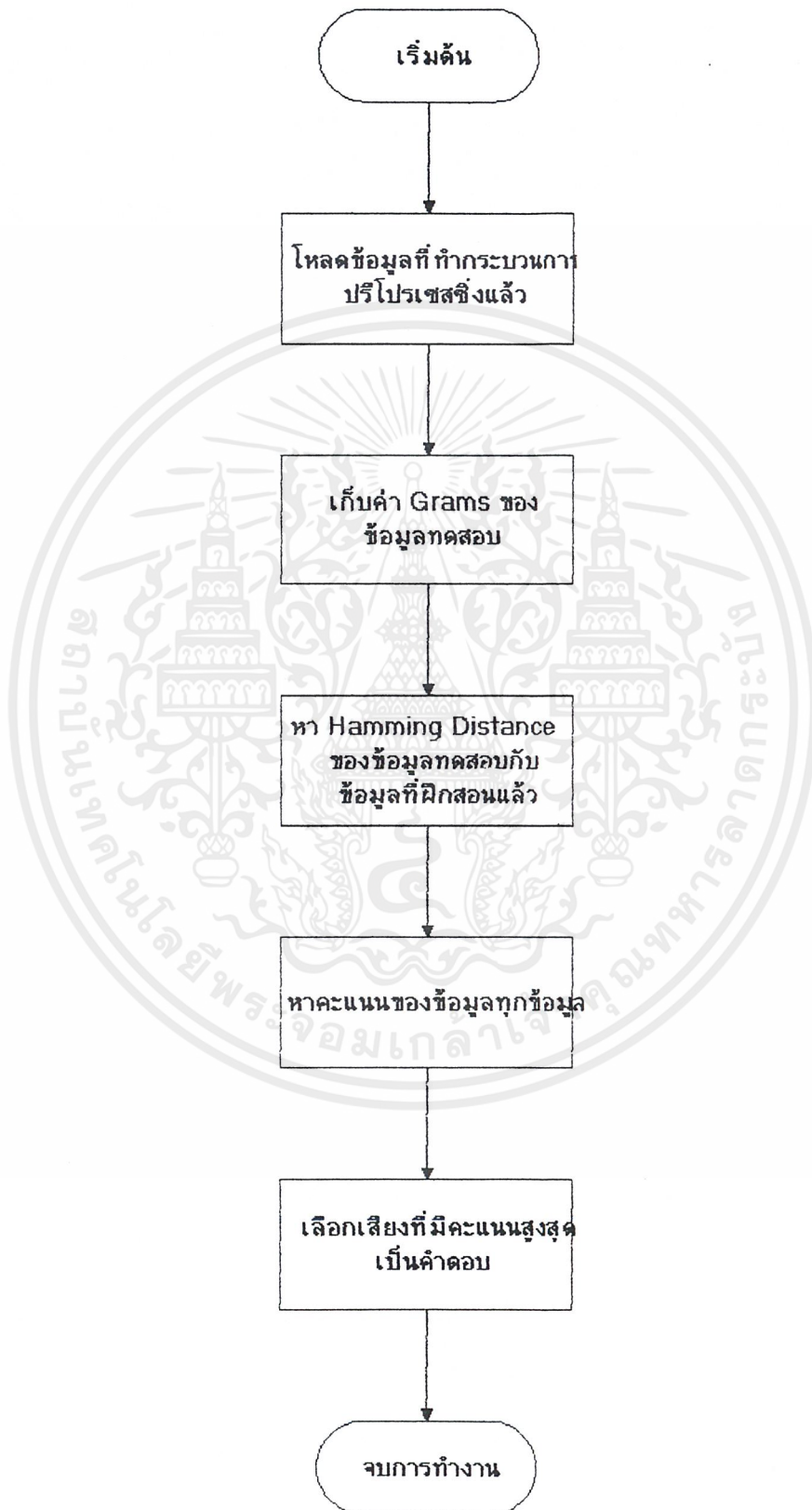
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Flowchart แสดงการทำงานของส่วนการฝึกสอน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Flowchart แสดงการทำงานของส่วนการทดสอบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Code ของโปรแกรมการทำงาน

**unit fool1;**

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, ExtCtrls, Menus, Buttons;

type

TForm1 = class(TForm)

Panel1: TPanel;

GroupBox1: TGroupBox;

GroupBox2: TGroupBox;

TEST: TButton;

TRAIN: TButton;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

MainMenu1: TMainMenu;

File1: TMenuItem;

Train1: TMenuItem;

Test1: TMenuItem;

Exit1: TMenuItem;

BitBtn1: TBitBtn;

procedure TRAINClick(Sender: TObject);

procedure TESTClick(Sender: TObject);

procedure Train1Click(Sender: TObject);

procedure Test1Click(Sender: TObject);

procedure Exit1Click(Sender: TObject);

procedure BitBtn1Click(Sender: TObject);

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation
uses fool2, fool4;

{$R *.DFM}
procedure TForm1.TRAINClick(Sender: TObject);
begin
    form2.show;
end;

procedure TForm1.TESTClick(Sender: TObject);
begin
    form4.show;
end;

procedure TForm1.Train1Click(Sender: TObject);
begin
    form2.show;
end;

procedure TForm1.Test1Click(Sender: TObject);
begin
    form4.show;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end;

**procedure TForm1.Exit1Click(Sender: TObject);**

begin

    application.Terminate;

end;

**procedure TForm1.BitBtn1Click(Sender: TObject);**

begin

    application.Terminate;

end;

end.

**unit fool2;**

interface

uses

    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
    StdCtrls, Buttons;

type

TForm2 = class(TForm)

    Password: TGroupBox;

    Edit1: TEdit;

    Label1: TLabel;

    OK: TBitBtn;

    procedure OKClick(Sender: TObject);

private

    { Private declarations }

public

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    { Public declarations }

end;

var
    Form2: TForm2;

implementation
uses fool3;

{$R *.DFM}

procedure TForm2.OKClick(Sender: TObject);
begin
    if Edit1.text = 'a' then
    begin
        form3.show;
        form2.Visible:=false;
    end
    else showmessage('Please Enter Password Again');

end;

end.

unit fool3;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
    ExtCtrls, StdCtrls, Fourier, RChart, Buttons;

const leg = 7; {leg of neural}
        neu = 100; {number of neural}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
nd = 100; {number of data}
xfinish = 238; {width of data-1}
yfinish = 130; {height of data-1}
```

type

```
dd = ^ddd;
ddd = array[1..nd, 0..xfinish, 0..yfinish] of byte;
bb = array[1..neu, 1..leg] of integer;
xx = array[1..neu, 1..leg] of integer;
yy = array[1..neu, 1..leg] of integer;
gg = array[1..nd, 1..neu, 1..leg] of byte;
rr = ^rrr;
rrr = array[0..xfinish, 0..yfinish] of integer;
TForm3 = class(TForm)
  GroupBox4: TGroupBox;
  TRAIN: TButton;
  FastFourier1: TFastFourier;
  RChart1: TRChart;
  RChart2: TRChart;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  EXIT: TBitBtn;
  Image1: TImage;
  Label7: TLabel;
  Edit1: TEdit;
  Label8: TLabel;
  Bevel1: TBevel;
  Bevel2: TBevel;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Bevel3: TBevel;
procedure TRAINClick(Sender: TObject);
procedure MENUClick(Sender: TObject);
procedure EXITClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form3: TForm3;
implementation
uses fool1;

{$R *.DFM}
procedure TForm3.TRAINClick(Sender: TObject);
var
  f_point : file of word;
  data : word;
  l,a,b,a1,b1,c,v,x,y,pd,red,white,count,up,down : integer;
  sum,ssqr,amp1,gau1 : integer;
  mean,std,amp,gauss:real;
  pg,offset,z,chk,m,n,min,max : integer;
  y1 : double;
  voice:array[1..nd]of string;
  d:dd;
  r:rr;
  g:gg;
  codinatex:array[1..700{..}] of integer;
  codinatey:array[1..700{..}] of integer;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

fixcox:xx;
x_point:file of xx;
fixcoy:yy;
y_point:file of yy;
g_point:file of gg;
tt:array[0..xfinish,0..yfinish] of byte;
Bitmap : TBitmap;

begin
new(d); new(r);
sum:=0; ssqr:=0;
voice[1]:= 'D:\Student\um&fung\voice\new\am1.1.wav';
voice[2]:= 'D:\Student\um&fung\voice\new\am1.2.wav';
voice[3]:= 'D:\Student\um&fung\voice\new\am1.3.wav';
voice[4]:= 'D:\Student\um&fung\voice\new\am1.4.wav';
voice[5]:= 'D:\Student\um&fung\voice\new\am1.5.wav';
voice[6]:= 'D:\Student\um&fung\voice\new\am1.6.wav';
voice[7]:= 'D:\Student\um&fung\voice\new\am1.7.wav';
voice[8]:= 'D:\Student\um&fung\voice\new\am1.8.wav';
voice[9]:= 'D:\Student\um&fung\voice\new\am1.9.wav';
voice[10]:= 'D:\Student\um&fung\voice\new\am1.10.wav';

voice[11]:= 'D:\Student\um&fung\voice\new\am2.1.wav';
voice[12]:= 'D:\Student\um&fung\voice\new\am2.2.wav';
voice[13]:= 'D:\Student\um&fung\voice\new\am2.3.wav';
voice[14]:= 'D:\Student\um&fung\voice\new\am2.4.wav';
voice[15]:= 'D:\Student\um&fung\voice\new\am2.5.wav';
voice[16]:= 'D:\Student\um&fung\voice\new\am2.6.wav';
voice[17]:= 'D:\Student\um&fung\voice\new\am2.7.wav';
voice[18]:= 'D:\Student\um&fung\voice\new\am2.8.wav';
voice[19]:= 'D:\Student\um&fung\voice\new\am2.9.wav';
voice[20]:= 'D:\Student\um&fung\voice\new\am2.10.wav';

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

voice[21]:= 'D:\Student\um&fung\voice\new\am3.1.wav';  
voice[22]:= 'D:\Student\um&fung\voice\new\am3.2.wav';  
voice[23]:= 'D:\Student\um&fung\voice\new\am3.3.wav';  
voice[24]:= 'D:\Student\um&fung\voice\new\am3.4.wav';  
voice[25]:= 'D:\Student\um&fung\voice\new\am3.5.wav';  
voice[26]:= 'D:\Student\um&fung\voice\new\am3.6.wav';  
voice[27]:= 'D:\Student\um&fung\voice\new\am3.7.wav';  
voice[28]:= 'D:\Student\um&fung\voice\new\am3.8.wav';  
voice[29]:= 'D:\Student\um&fung\voice\new\am3.9.wav';  
voice[30]:= 'D:\Student\um&fung\voice\new\am3.10.wav';

voice[31]:= 'D:\Student\um&fung\voice\new\am4.1.wav';  
voice[32]:= 'D:\Student\um&fung\voice\new\am4.2.wav';  
voice[33]:= 'D:\Student\um&fung\voice\new\am4.3.wav';  
voice[34]:= 'D:\Student\um&fung\voice\new\am4.4.wav';  
voice[35]:= 'D:\Student\um&fung\voice\new\am4.5.wav';  
voice[36]:= 'D:\Student\um&fung\voice\new\am4.6.wav';  
voice[37]:= 'D:\Student\um&fung\voice\new\am4.7.wav';  
voice[38]:= 'D:\Student\um&fung\voice\new\am4.8.wav';  
voice[39]:= 'D:\Student\um&fung\voice\new\am4.9.wav';  
voice[40]:= 'D:\Student\um&fung\voice\new\am4.10.wav';

voice[41]:= 'D:\Student\um&fung\voice\new\am5.1.wav';  
voice[42]:= 'D:\Student\um&fung\voice\new\am5.2.wav';  
voice[43]:= 'D:\Student\um&fung\voice\new\am5.3.wav';  
voice[44]:= 'D:\Student\um&fung\voice\new\am5.4.wav';  
voice[45]:= 'D:\Student\um&fung\voice\new\am5.5.wav';  
voice[46]:= 'D:\Student\um&fung\voice\new\am5.6.wav';  
voice[47]:= 'D:\Student\um&fung\voice\new\am5.7.wav';  
voice[48]:= 'D:\Student\um&fung\voice\new\am5.8.wav';

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

voice[49]:= 'D:\Student\um&fung\voice\new\am5.9.wav';  
voice[50]:= 'D:\Student\um&fung\voice\new\am5.10.wav';

voice[51]:= 'D:\Student\um&fung\voice\new\am6.1.wav';  
voice[52]:= 'D:\Student\um&fung\voice\new\am6.2.wav';  
voice[53]:= 'D:\Student\um&fung\voice\new\am6.3.wav';  
voice[54]:= 'D:\Student\um&fung\voice\new\am6.4.wav';  
voice[55]:= 'D:\Student\um&fung\voice\new\am6.5.wav';  
voice[56]:= 'D:\Student\um&fung\voice\new\am6.6.wav';  
voice[57]:= 'D:\Student\um&fung\voice\new\am6.7.wav';  
voice[58]:= 'D:\Student\um&fung\voice\new\am6.8.wav';  
voice[59]:= 'D:\Student\um&fung\voice\new\am6.9.wav';  
voice[60]:= 'D:\Student\um&fung\voice\new\am6.10.wav';

voice[61]:= 'D:\Student\um&fung\voice\new\am7.1.wav';  
voice[62]:= 'D:\Student\um&fung\voice\new\am7.2.wav';  
voice[63]:= 'D:\Student\um&fung\voice\new\am7.3.wav';  
voice[64]:= 'D:\Student\um&fung\voice\new\am7.4.wav';  
voice[65]:= 'D:\Student\um&fung\voice\new\am7.5.wav';  
voice[66]:= 'D:\Student\um&fung\voice\new\am7.6.wav';  
voice[67]:= 'D:\Student\um&fung\voice\new\am7.7.wav';  
voice[68]:= 'D:\Student\um&fung\voice\new\am7.8.wav';  
voice[69]:= 'D:\Student\um&fung\voice\new\am7.9.wav';  
voice[70]:= 'D:\Student\um&fung\voice\new\am7.10.wav';

voice[71]:= 'D:\Student\um&fung\voice\new\am8.1.wav';  
voice[72]:= 'D:\student\um&fung\voice\new\am8.2.wav';  
voice[73]:= 'D:\student\um&fung\voice\new\am8.3.wav';  
voice[74]:= 'D:\student\um&fung\voice\new\am8.4.wav';  
voice[75]:= 'D:\student\um&fung\voice\new\am8.5.wav';  
voice[76]:= 'D:\student\um&fung\voice\new\am8.6.wav';

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
voice[77]:= 'D:\student\um&fung\voice\new\am8.7.wav';  
voice[78]:= 'D:\student\um&fung\voice\new\am8.8.wav';  
voice[79]:= 'D:\student\um&fung\voice\new\am8.9.wav';  
voice[80]:= 'D:\student\um&fung\voice\new\am8.10.wav';
```

```
voice[81]:= 'D:\student\um&fung\voice\new\am9.1.wav';  
voice[82]:= 'D:\student\um&fung\voice\new\am9.2.wav';  
voice[83]:= 'D:\student\um&fung\voice\new\am9.3.wav';  
voice[84]:= 'D:\student\um&fung\voice\new\am9.4.wav';  
voice[85]:= 'D:\student\um&fung\voice\new\am9.5.wav';  
voice[86]:= 'D:\student\um&fung\voice\new\am9.6.wav';  
voice[87]:= 'D:\student\um&fung\voice\new\am9.7.wav';  
voice[88]:= 'D:\student\um&fung\voice\new\am9.8.wav';  
voice[89]:= 'D:\student\um&fung\voice\new\am9.9.wav';  
voice[90]:= 'D:\student\um&fung\voice\new\am9.10.wav';
```

```
voice[91]:= 'D:\student\um&fung\voice\new\am0.1.wav';  
voice[92]:= 'D:\student\um&fung\voice\new\am0.2.wav';  
voice[93]:= 'D:\student\um&fung\voice\new\am0.3.wav';  
voice[94]:= 'D:\student\um&fung\voice\new\am0.4.wav';  
voice[95]:= 'D:\student\um&fung\voice\new\am0.5.wav';  
voice[96]:= 'D:\student\um&fung\voice\new\am0.6.wav';  
voice[97]:= 'D:\student\um&fung\voice\new\am0.7.wav';  
voice[98]:= 'D:\student\um&fung\voice\new\am0.8.wav';  
voice[99]:= 'D:\student\um&fung\voice\new\am0.9.wav';  
voice[100]:= 'D:\student\um&fung\voice\new\am0.10.wav';
```

```
for pd:=1 to nd do
```

```
begin
```

```
Edit1.text:=IntToStr(pd);
```

```
RChart1.ClearGraf;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RChart1.DataColor := clBlue;
RChart1.RangeLoX := 0;
RChart1.RangeHiX := 3000;
RChart1.RangeLoY := -128;
RChart1.RangeHiY := 127;
RChart1.MoveTo (0,0);
FastFourier1.Clear;
x := 0;
assignfile(f_point,voice[pd]);
reset(f_point);

!Cut Head file;
for a := 1 to 22 do
  if Eof(f_point) then break
  else read(f_point,data);

!Plot graph!
for a:=1 to FastFourier1.SpectrumSize do
  begin
    if Eof(f_point) then break;
    read(f_point,data);
    b1 := integer(data);
    b1 := b1 div 256;
    if b1 < 125 then b1 := b1;
    if b1 > 125 then b1 := b1-256;
    if x < b1 then
      for c := x to b1 do
        begin
          y := c;
          FastFourier1.RealSpec[a] := y;
          RChart1.DrawTo(a,y);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    end;
  if x > b1 then
    for c := b1 to x do
      begin
        y := c;
        FastFourier1.RealSpec[a] := y;
        RChart1.DrawTo(a,y);
      end;
    x := b1;
  end;
closefile(f_point);
RChart1.ShowGraf;
FastFourier1.Transform;
RChart2.ClearGraf;
RChart2.RangeLoY := 0;
RChart2.RangeHiY := 5;
RChart2.RangeLoX := 0;
RChart2.RangeHiX := 200;
RChart2.DataColor := clRed;
RChart2.MoveTo (FastFourier1.FreqOfLine(1,0.001),0);
for a:=1 to (FastFourier1.SpectrumSize div 2) do
  begin
    y1:=FastFourier1.Magnitude[a];
    while y1 >=0 do
      begin
        RChart2.DrawTo (FastFourier1.FreqOfLine(a,0.001),y1);
        y1:=y1-0.05;
      end;
    end;
  end;
RChart2.ShowGraf;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{Normalization}
x:=0; y:=0; up:=0; down:=0;
while y < yfinish do
begin
  while x < xfinish do
  begin
    if form3.canvas.pixels[491+x,97+y] = clred then
    begin
      up:=y; x:=xfinish+1; y:=yfinish+1;
    end
    else x:=x+1;
  end;
  y:=y+1;x:=0;
end;
x:=0; y:=yfinish;
while y >= 0 do
begin
  while x < xfinish do
  begin
    if form3.canvas.pixels[491+x,97+y] = clwhite then
    begin
      down:=y; x:=xfinish+1; y:=0;
    end
    else x:=x+1;
  end;
  y:=y-1;x:=0;
end;

{Clear old Output}
Bitmap := TBitmap.Create;
Bitmap.Width := xfinish+1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Bitmap.Height := down-up+1;
Image1.Picture.Graphic := Bitmap;
for y:=up to down do
  for x:=0 to xfinish do
    begin
      image1.canvas.pixels[x,y-up]:=form3.canvas.pixels[491+x,97+y];
    end;
  image1.Width:=xfinish+1;
  image1.Height:=yfinish+1;
  form3.Update;
  image1.Stretch:=true;
  form3.update;
  form3.Update;

{Mean}
x:=0;y:=0;
while x<xfinish+1 do
  begin
    while y<yfinish+1 do
      begin
        if form3.canvas.pixels[120+x,360+y] = clred then
          begin
            sum := sum+y;
            ssqr:=ssqr+(y*y);
            y:=yfinish+1;
          end
        else y:=y+1;
      end;
    x:=x+1; y:=0;
  end;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{Keep data}
for y:=0 to yfinish do
for x:=0 to xfinish do
begin
if form3.canvas.pixels[120+x,360+y] = cired then d^[pd,x,y]:=1
else d^[pd,x,y]:=0;
end;
end;

{Mean & Standard Deviation}
mean:=0; std:=0;
mean:=sum /((xfinish+1)*nd);
std:=sqrt((ssqr div ((xfinish+1)*nd))-mean*mean);

{Find Coordinate}
for n:=1 to neu do
for l:=1 to leg do
begin
fixcox[n,l]:=0; fixcoy[n,l]:=0;
end;
for y:=0 to yfinish do
for x:=0 to xfinish do
begin
tt[x,y]:=0;
end;
amp:=(1/(std*sqrt(2*3.14)))*1000000;
amp1:=round(amp);
n:=1;
while n <= neu do
begin
l:=1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while l <= leg do
begin
a:=0; b:=0; randomize;
a:=random(238);
b:=random(130);
gauss:=((1/(std*sqrt(2*3.14)))*exp(-(sqr(b-mean))/(2*std*std)))*1000000;
gaul:=round(gauss);
a1:=random(amp1);
if tt[a,b] > 2 then
begin
fixcox[n,l]:=a;
if a1 < gaul then
begin
fixcoy[n,l]:=b;
l:=l+1;
tt[a,b]:=2;
end;
end;
end;
n:=n+1;
end;

{Keep GRams}
for pd:=1 to nd do
for n:=1 to neu do
for l:=1 to leg do
begin
g[pd,n,l]:=d^[pd,fixcox[n,l],fixcoy[n,l]];
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{Save file of GRams}  
assignfile(g_point,'D:\student\um&fung\file\grams4.uf');  
rewrite(g_point);  
write(g_point,g);  
CloseFile(g_point);
```

```
{Save file of Coordinate}  
assignfile(x_point,'D:\student\um&fung\file\cox4.uf');  
rewrite(x_point);  
write(x_point,fixcox);  
CloseFile(x_point);  
assignfile(y_point,'D:\student\um&fung\file\coy4.uf');  
rewrite(y_point);  
write(y_point,fixcoy);  
CloseFile(y_point);
```

```
end;
```

```
procedure TForm3.MENUClick(Sender: TObject);
```

```
begin
```

```
    form3.Visible:=false;
```

```
    form1.show;
```

```
end;
```

```
procedure TForm3.EXITClick(Sender: TObject);
```

```
begin
```

```
    application.Terminate;
```

```
end;
```

```
end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### **unit fool4;**

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, Fourier, RChart, ExtCtrls, WavRec;

const leg = 7; {leg of neural}

neu = 100; {number of neural}

nd = 100; {number of data}

fd = 10; {frequency of data in 1 speech data}

nv = 10; {number of voice = nd/fd}

xfinish = 238; {width of data-1}

yfinish = 130; {height of data-1}

type

gg=array[1..nd,1..neu,1..leg] of byte;

xx=array[1..neu,1..leg] of integer;

yy=array[1..neu,1..leg] of integer;

TForm4 = class(TForm)

FastFourier1: TFastFourier;

GroupBox4: TGroupBox;

GroupBox5: TGroupBox;

OPEN: TBitBtn;

TEST: TBitBtn;

GroupBox6: TGroupBox;

GroupBox7: TGroupBox;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Edit4: TEdit;

Edit5: TEdit;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Edit6: TEdit;  
Edit7: TEdit;  
Edit8: TEdit;  
Edit9: TEdit;  
Edit10: TEdit;  
Label5: TLabel;  
Label6: TLabel;  
Label7: TLabel;  
Label8: TLabel;  
Label9: TLabel;  
Label10: TLabel;  
Label11: TLabel;  
Label12: TLabel;  
Label13: TLabel;  
Label14: TLabel;  
OpenDialog1: TOpenDialog;  
Edit11: TEdit;  
Label15: TLabel;  
RChart1: TRChart;  
RChart2: TRChart;  
Label1: TLabel;  
Label2: TLabel;  
Label3: TLabel;  
Label4: TLabel;  
Label16: TLabel;  
Label17: TLabel;  
EXIT: TBitBtn;  
Image1: TImage;  
Label18: TLabel;  
Bevel1: TBevel;  
Bevel2: TBevel;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Bevel3: TBevel;
GroupBox1: TGroupBox;
WavRec1: TWavRec;
StartREC: TBitBtn;
FFT: TBitBtn;
StopREC: TBitBtn;
procedure TESTClick(Sender: TObject);
procedure OPENClick(Sender: TObject);
procedure FFTClick(Sender: TObject);
procedure MENUClick(Sender: TObject);
procedure EXITClick(Sender: TObject);
procedure StartRECClick(Sender: TObject);
procedure StopRECClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form4: TForm4;
implementation
uses fool1;

{$R *.DFM}

procedure TForm4.TESTClick(Sender: TObject);
var
  g:gg;
  g_point:file of gg;
  fixcox:xx;
  fixcoy:yy;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

x_point:file of xx;
y_point:file of yy;
x,y,z,n,l,pg,pd,pv,offset,max,result,up,down : integer;
t:array[0..xfinish,0..yfinish] of byte;
gt:array[1..neu,1..leg] of byte;
h:array[1..nd,1..neu] of byte;
min:array[1..neu] of integer;
score:array[1..nd,1..neu] of integer;
tscore:array[1..nd] of integer;
stscore:array[1..nv] of integer;
Bitmap : TBitmap;
begin
assignfile(g_point,'D:\student\um&fung\file\grams4.uf');
reset(g_point);
read(g_point,g);
CloseFile(g_point);
assignfile(x_point,'D:\student\um&fung\file\cox4.uf');
reset(x_point);
read(x_point,fixcox);
CloseFile(x_point);
assignfile(y_point,'D:\student\um&fung\file\coy4.uf');
reset(y_point);
read(y_point,fixcoy);
CloseFile(y_point);

{Normalization}
x:=0; y:=0; up:=0; down:=0;
while y < yfinish do
begin
while x < xfinish do
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if form4.canvas.pixels[491+x,97+y] = clred then
begin
  up:=y; x:=xfinish+1; y:=yfinish+1;
end
else x:=x+1;
end;
y:=y+1;x:=0;
end;
x:=0; y:=yfinish;
while y >= 0 do
begin
  while x < xfinish do
  begin
    if form4.canvas.pixels[491+x,97+y] = clwhite then
    begin
      down:=y; x:=xfinish+1; y:=0;
    end
    else x:=x+1;
  end;
  y:=y-1;x:=0;
end;

{Clear old Output}
Bitmap := TBitmap.Create;
Bitmap.Width := xfinish+1;
Bitmap.Height := down-up+1;
Image1.Picture.Graphic := Bitmap;
for y:=up to down do
  for x:=0 to xfinish do
    begin
      image1.canvas.pixels[x,y-up]:=form4.canvas.pixels[491+x,97+y];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;
image1.Width:=xfinish+1;
image1.Height:=yfinish+1;
image1.Stretch:=true;
form4.Update;

```

```
{Keep Test data}
```

```

for y:=0 to yfinish do
  for x:=0 to xfinish do
    begin
      if form4.Canvas.Pixels[128+x,328+y] = clred then t[x,y]:=1
      else t[x,y]:=0;
    end;
  end;
end;

```

```
{Keep GRans out of Test data}
```

```

for n:=1 to neu do
  for l:=1 to leg do
    gt[n,l]:=t[fixcox[n,l],fixcoy[n,l]];
  end;
end;

```

```
{Hamming distance}
```

```

for pg:=1 to neu do
  for pd:=1 to nd do
    h[pd,pg]:=0;
  end;
end;
for pd:=1 to nd do
  for n:=1 to neu do
    begin
      for l:=1 to leg do
        begin
          if gt[n,l] <> g[pd,n,l] then
            h[pd,n]:=h[pd,n]+1;
          end;
        end;
      end;
    end;
  end;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end;

{Minimum of Hamming distance}

for n:=1 to neu do

min[n]:=leg+1;

for n:=1 to neu do

for pd:=1 to nd do

begin

if min[n] > h[pd,n] then

min[n]:= h[pd,n];

end;

{Score of GRams}

for n:=1 to neu do

for pd:=1 to nd do

score[pd,n]:=0;

for pd:=1 to nd do

for n:=1 to neu do

begin

if h[pd,n] = min[n] then score[pd,n]:=score[pd,n]+1;

end;

{Total Score}

for pd:=1 to nd do tscore[pd]:=0;

for pd:=1 to nd do

for n:=1 to neu do

tscore[pd]:=tscore[pd]+score[pd,n];

for pv:=1 to nv do stscore[pv]:=0;

for pv:=0 to nv-1 do

for pd:=1 to fd do

stscore[pv+1]:=stscore[pv+1]+tscore[pd+(pv\*fd)];

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{Resulttest}
    max:=0;
    for pv:=1 to nv do
    begin
        if max < stscore[pv] then
            max:= stscore[pv];
        end;
    for pv:=1 to nv do
    begin
        if max = stscore[pv] then result:=pv;
    end;
    edit1.text:=IntToStr(stscore[1]);
    edit2.text:=IntToStr(stscore[2]);
    edit3.text:=IntToStr(stscore[3]);
    edit4.text:=IntToStr(stscore[4]);
    edit5.text:=IntToStr(stscore[5]);
    edit6.text:=IntToStr(stscore[6]);
    edit7.text:=IntToStr(stscore[7]);
    edit8.text:=IntToStr(stscore[8]);
    edit9.text:=IntToStr(stscore[9]);
    edit10.text:=IntToStr(stscore[10]);
    if result = 10 then result:=0;
    edit11.text:=IntToStr(result);
    TEST.enabled:=False;
    OPEN.enabled:=true;
end;

```

**procedure TForm4.OPENClick(Sender: TObject);**

var

CurrentFile:string;

f\_point : file of word;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

data : word;
i,a,b1,c,x,y : integer;
y1 : double;
Bitmap : TBitmap;

begin
edit1.Clear; edit2.clear; edit3.Clear; edit4.clear; edit5.Clear; edit6.clear;
edit7.Clear; edit8.clear; edit9.Clear; edit10.clear; edit11.Clear;

{Clear old Output}
Bitmap := TBitmap.Create;
Bitmap.Width := xfinish+1;
Bitmap.Height := yfinish+1;
Image1.Picture.Graphic := Bitmap;
if OpenFileDialog1.Execute then CurrentFile := OpenFileDialog1.FileName;
RChart1.ClearGraf;
RChart2.ClearGraf;
RChart1.DataColor := clBlue;
RChart1.RangeLoX := 0;
RChart1.RangeHiX := 3000;
RChart1.RangeLoY := -128;
RChart1.RangeHiY := 127;
RChart1.MoveTo (0,0);
FastFourier1.Clear;
x := 0;
assignfile(f_point,CurrentFile);
reset(f_point);

{Cut Head file}
for a := 1 to 22 do
    if Eof(f_point) then break
    else read(f_point,data);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{Plot graph}
for a:=1 to FastFourier1.SpectrumSize do
  begin
    if Eof(f_point) then break;
    read(f_point,data);
    b1 := integer(data);
    b1 := b1 div 256;
    if b1 < 125 then b1 := b1;
    if b1 > 125 then b1 := b1-256;
    if x < b1 then
      for c := x to b1 do
        begin
          y := c;
          FastFourier1.RealSpec[a] := y;
          RChart1.DrawTo(a,y);
        end;
    if x > b1 then
      for c := b1 to x do
        begin
          y := c;
          FastFourier1.RealSpec[a] := y;
          RChart1.DrawTo(a,y);
        end;
    x := b1;
  end;
closefile(f_point);
RChart1.ShowGraf;
FFT.Enabled := True;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm4.FFTClick(Sender: TObject);
```

```
var
```

```
  a,x,y : integer;
```

```
  y1 : double;
```

```
begin
```

```
FastFourier1.Transform;
```

```
RChart2.ClearGraf;
```

```
RChart2.RangeLoY := 0;
```

```
RChart2.RangeHiY := 5;
```

```
RChart2.RangeLoX := 0;
```

```
RChart2.RangeHiX := 200;
```

```
RChart2.DataColor := clRed;
```

```
RChart2.MoveTo (FastFourier1.FreqOfLine(1,0.001),0);
```

```
for a:=1 to (FastFourier1.SpectrumSize div 2) do
```

```
begin
```

```
  y1:=FastFourier1.Magnitude[a];
```

```
  while y1 >=0 do
```

```
  begin
```

```
    RChart2.DrawTo (FastFourier1.FreqOfLine(a,0.001),y1);
```

```
    y1:=y1-0.05;
```

```
  end;
```

```
end;
```

```
RChart2.ShowGraf;
```

```
FFT.Enabled := false;
```

```
TEST.Enabled := true;
```

```
end;
```

```
procedure TForm4.MENUClick(Sender: TObject);
```

```
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
form4.Visible:=false;
form1.show;
end;
```

```
procedure TForm4.EXITClick(Sender: TObject);
```

```
begin
    application.Terminate;
end;
```

```
procedure TForm4.StartRECClick(Sender: TObject);
```

```
begin
    wavrec1.WAVformat.channels:=Mono;
    wavrec1.WAVformat.SamplesPerSec:=11025;
    wavrec1.WAVformat.BitsPerSample:=16;
    wavrec1.BeginRecording;
    StopREC.enabled:=True;
end;
```

```
procedure TForm4.StopRECClick(Sender: TObject);
```

```
var
    f_point : file of word;
    data : word;
    i,a,b1,c,x,y : integer;
    y1 : double;
    Bitmap : TBitmap;
begin
    wavrec1.StopRecording;
    edit1.Clear; edit2.clear; edit3.Clear; edit4.clear; edit5.Clear; edit6.clear;
    edit7.Clear; edit8.clear; edit9.Clear; edit10.clear; edit11.Clear;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{Clear old Output}
Bitmap := TBitmap.Create;
Bitmap.Width := xfinish+1;
Bitmap.Height := yfinish+1;
Image1.Picture.Graphic := Bitmap;
RChart1.ClearGraf;
RChart2.ClearGraf;
RChart1.DataColor := clBlue;
RChart1.RangeLoX := 0;
RChart1.RangeHiX := 3000;
RChart1.RangeLoY := -128;
RChart1.RangeHiY := 127;
RChart1.MoveTo (0,0);
FastFourier1.Clear;
x := 0;
assignfile(f_point,'D:\Student\um&fung\voice\test\fung.wav');
reset(f_point);

{Cut Head file}
for a := 1 to 22 do
    if Eof(f_point) then break
    else read(f_point,data);

{Plot graph}
for a:=1 to FastFourier1.SpectrumSize do
    begin
        if Eof(f_point) then break;
        read(f_point,data);
        b1 := integer(data);
        b1 := b1 div 256;
        if b1 < 125 then b1 := b1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if b1 > 125 then b1 := b1-256;
if x < b1 then
  for c := x to b1 do
    begin
      y := c;
      FastFourier1.RealSpec[a] := y;
      RChart1.DrawTo(a,y);
    end;
  if x > b1 then
    for c := b1 to x do
      begin
        y := c;
        FastFourier1.RealSpec[a] := y;
        RChart1.DrawTo(a,y);
      end;
    x := b1;
  end;
closefile(f_point);
RChart1.ShowGraf;
FFT.Enabled := True;
StopREC.enabled:=False;
erase(f_point);
end;

end.

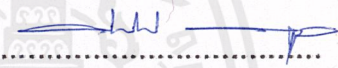
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ปริญญาบัตรฉบับนี้ สำเร็จลงได้ด้วยดีด้วยดีจากความช่วยเหลือจากหลายๆ ด้าน โดยเฉพาะ ดร. ยุทธนา คิทธิใจเดียว (อาจารย์ที่ปรึกษา) ที่คอยให้คำปรึกษาและชี้แนะแนวทาง ทฤษฎีเกี่ยวกับนิวรอลเน็ตเวิร์ค (Weightless Neural Network) และ ผศ.ดร. สุรพันธ์ เอื้อไพบูลย์ ที่กรุณาให้ใช้คอมพิวเตอร์ในการเขียนโปรแกรม ทางผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูง ขอขอบคุณ รุ่นพี่ปริญญาโทที่คอยให้คำแนะนำและข้อมูลต่างๆ และเพื่อนๆ ที่คอยช่วยเหลือตลอดมา

สุดท้ายนี้ ขอกราบขอบพระคุณ บิดา มารดา ที่ให้การสนับสนุน และให้กำลังใจ จนปริญญาบัตรนี้สำเร็จลงด้วยดี

  
.....  
(นาย ประพันธ์พงษ์ นทกุล)

  
.....  
(นาย พัสกร พูลสุขศิริ)

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หนังสืออ้างอิง

1. นาย เสรี ปานซาง ,วิทยานิพนธ์ เรื่อง “การรู้จำเสียงพูดคำไทยแบบไม่ขึ้นกับผู้พูดด้วยนิเวศน์เน็ตเวิร์ค” โดย นาย เสรี ปานซาง , นักศึกษาวิศวกรรมศาสตรมหาบัณฑิต สาขา วิศวกรรมไฟฟ้า สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง พ.ศ. 2540
2. เสรี ปานซาง และชม กิมปาน , “การรู้จำเสียงพูดคำไทยเฉพาะบุคคลด้วยนิเวศน์เน็ตเวิร์ค” , การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18 , ชลบุรี : มหาวิทยาลัยมหานคร, 982-987 , พ.ศ. 2538
3. ศ.ดร.วัลลภ สุระกำพลธร , “การประมวลผลสัญญาณเชิงเลข” สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 116 หน้า , ตุลาคม 2533
4. Wiley , “Digital Signal Processing” , Defatta Lucas
5. ชันวา ศรีประโมง และชม กิมปาน , “การวิเคราะห์เสียงภาษาไทยในแกนความถี่ฮาร์โมนิค” , การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 15 , พ.ศ. 2535