

# การรู้จำเสียงพูด

## Speech Recognition



ปริิฤฎยำนั้พนกันั้เป็นส่วนหนึ่งของการศึษาตามหลั้กสู้ตรปรัญญาวิศวะกรรมศาสตร์บั้ฉบั้ฉ

สาขาวิศวะกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2536.

เอกสารนี้้เป็นเอกสารที่้สงวนไว้้สำหรับการใช้้งานเพื่่การศึกษาเท่านั้น ไม่้อนุญาตให้้นำไป้ใช้้ประริ้ษังอื่นั้การใ้  
ไม่้ว่า้กรณี้ใ้ใ้ทุกั้สั้น อี้กั้ทั้งห้ามมิให้้ดัดแปลงเนื้อ้า และต้องอ้างอิงถึงเจ้า้ของเอกสารทุกั้ครั้งที่มี้การนำไป้ใช้้

033407

การรู้จำเสียงพูด  
Speech Recognition

โดย นาย กิติกร พิพัฒนันต์ดา  
นาย สุนจน์ อินทร์ชัยยะ

อาจารย์ที่ปรึกษา อาจารย์ เกียรติกร วงศ์โรจน์ภรณ์

บทคัดย่อ

ในปัจจุบัน วิทยาการต่างๆ ได้ถูกพัฒนาขึ้นอย่างมากรวมทั้งการติดต่อระหว่างมนุษย์กับเครื่องจักร วิทยานิพนธ์นี้ได้กล่าวถึงระบบที่ใช้ในการสื่อสารระหว่างคอมพิวเตอร์กับผู้ใช้แบบคำเดี่ยวโดยสั่งงานได้ เพียงบุคคลเดียว เสียงที่ใช้เป็นตัวเลข 0-9 ซึ่งประกอบด้วยขั้นตอน 5 ขั้นตอนคือ 1) เปลี่ยนเสียงที่เป็นสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัลเพื่อนำไปใช้คำนวณหาพารามิเตอร์ต่างๆต่อไป 2) สัญญาณที่ได้จะถูกตัดค่าเพื่อหาขอบเขตของสัญญาณ โดยใช้ค่าเฉลี่ยพลังงานของสัญญาณและอัตราการตัดศูนย์ 3) หลังจากการตัดค่าแล้วจะนำสัญญาณไปหาพารามิเตอร์ของนิวเคลียร์โดยวิธีลิเนียร์พรีดิกทีฟ ซึ่งมี 8 โพล 4) ขั้นตอนสุดท้ายเป็นการรับรู้โดยใช้เทคนิคไดนามิกโปรแกรมมิ่ง เพื่อหาระยะทางระหว่างแบบอ้างอิงกับแบบทดสอบและกฎการการตัดสินใจได้อาศัยเทคนิคของ K-Nearest Neighbour (KNN)

ABSTRACT

In the present, many civilization had developed for all time include the communication between human with machine. This project discusses the system that used to commute between computer with user, using the speech of digit(0-9) in isolated dependent word. Procedure of system compose of 5 step: 1) Transfer analog speech signal to digital signal in order to compute parameter 2) Signal, which is received, was cut the signal in order to create the range of signal by used average energy of signal and zero crossing technique. 3) After the second step, using the signal to find the parameter of filter by using linear predictive analysis which has 8 pole. 4) The last step, distance between the reference patterns and the test pattern by using Dynamic Programming Technique and The decisionrule are created by using K-nearest Neighbour (KNN) technique

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

|   | หน้า |
|---|------|
| บทคัดย่อภาษาไทย                                   | ง    |
| บทคัดย่อภาษาอังกฤษ                                | ง    |
| บทที่ 1 บทนำ                                      | 1    |
| บทที่ 2 การวิเคราะห์เสียงพูด                      | 4    |
| 2.1 เสียงพูด                                      | 4    |
| 2.1.1 อวัยวะที่ใช้ในการออกเสียง                   | 4    |
| 2.1.2 การเกิดของเสียง                             | 6    |
| 2.2 การวิเคราะห์สัญญาณเสียง                       | 7    |
| 2.2.1 การวิเคราะห์ในโดเมนเวลา                     | 7    |
| 2.2.2 การวิเคราะห์ทางความถี่                      | 15   |
| 2.2.3 การหาขอบเขตของคำ                            | 18   |
| บทที่ 3 การเข้ารหัสแบบลิเนียร์พรีดิคทีฟ           | 27   |
| 3.1 หลักการเบื้องต้นของแอลพีซี                    | 27   |
| 3.2 การหาค่าพารามิเตอร์ฟิลเตอร์โดยวิธีโควาเรียนซ์ | 32   |
| 3.3 การหาพารามิเตอร์ฟิลเตอร์โดยวิธีอีทสหลัมพันธ์  | 33   |
| 3.4 การหาพารามิเตอร์ฟิลเตอร์โดยวิธีนาร์คอร์       | 42   |
| บทที่ 4 การรับรู้เสียงพูด                         | 44   |
| 4.1 การจัดกลุ่มเพื่อสร้างแบบอ้างอิง               | 44   |
| 4.1.1 การจุดศูนย์กลางของกลุ่ม                     | 45   |
| 4.1.2 การจัดกลุ่มใหม่                             | 45   |
| 4.2 การเปรียบเทียบแบบไดนามิกโปรแกรมมิ่ง           | 46   |
| 4.2.1 ฟังก์ชันแวร์ปิง                             | 46   |
| 4.2.2 สมการไดนามิกโปรแกรมมิ่ง                     | 51   |
| 4.3 กฎการตัดสินใจ                                 | 52   |
| 4.3.1 กฎ Nearest Neighbour                        | 52   |
| 4.3.2 กฎ K-Nearest Neighbour                      | 52   |

|  |    |
|--|----|
| บทที่ 5 การออกแบบการบันทึกและแสดงเสียงพูด    | 55 |
| 5.1 การอนาล็อกโปรเซสเซอร์                    | 55 |
| 5.1.1 ภาคสัญญาณเสียงขาเข้า                   | 55 |
| - วงจรขยายสัญญาณจากไมโครโฟน                  | 56 |
| - วงจรกรองความถี่ต่ำ                         | 57 |
| - วงจรยกระดับ                                | 59 |
| 5.2 ภาคอนาล็อกอินเทอร์เฟส                    | 60 |
| 5.2.1 วงจรบัฟเฟอร์                           | 61 |
| 5.2.2 วงจรเลือกพอร์ต                         | 62 |
| 5.2.3 วงจรส่วน สัญญาณนาฬิกา                  | 63 |
| 5.2.4 วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล | 63 |
| 5.2.5 วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณเสียง   | 64 |
| บทที่ 6 การรับรู้เสียงพูดบนไมโครคอมพิวเตอร์  | 66 |
| 6.1 โปรแกรมการบันทึกเสียง                    | 69 |
| 6.2 โปรแกรมการวิเคราะห์                      | 71 |
| 6.2.1 โปรแกรมหาขอบเขตของสัญญาณ               | 71 |
| 6.2.2 โปรแกรมหาค่านารามิเตอร์ของสัญญาณ       | 73 |
| 6.3 โปรแกรมการรับรู้เสียงพูด                 | 75 |
| 6.3.1 โปรแกรมสร้างแฟ้มข้อมูลนารามิเตอร์      | 75 |
| 6.3.2 โปรแกรมคัดเลือกแบบอ้างอิง              | 78 |
| 6.3.3 โปรแกรมการรับรู้                       | 80 |
| บทที่ 7 การทดลองและผลการทดลอง                | 82 |
| บทที่ 8 สรุปและข้อเสนอแนะ                    | 85 |
| เอกสารอ้างอิง                                | 86 |
| กิตติกรรมประกาศ                              | 88 |
| ภาคผนวก                                      | 89 |
| ก. วงจรสมบูรณ์ ของการบันทึกเสียง             | 90 |

เอกสารนี้เป็นเอกสารของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
 ข. ตัวอย่างของสัญญาณภาษาเสียงที่ใช้ในการทดสอบนั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ 92 ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ ๑

## บทนำ

ปัจจุบันเครื่องคอมพิวเตอร์ถูกนำมาใช้กันอย่างแพร่หลาย เพื่อช่วยให้การประมวลผลเป็นไปอย่างถูกต้องและรวดเร็ว โดยปกติการติดต่อระหว่างเครื่องคอมพิวเตอร์กับมนุษย์ จะทำโดยมนุษย์จะเป็นผู้ป้อนคำสั่งให้คอมพิวเตอร์โดยผ่านทางแป้นพิมพ์ และเครื่องคอมพิวเตอร์ก็จะแสดงผลออกทางหน้าจอหรือเครื่องพิมพ์ เพื่อให้มนุษย์ได้ทราบ ซึ่งการติดต่อกับเครื่องโดยผ่านทางแป้นพิมพ์ ต้องอาศัยความชำนาญ และการฝึกฝนเป็นเวลานาน จึงจะสามารถใช้เครื่องคอมพิวเตอร์ได้อย่างมีประสิทธิภาพโดยปกติมนุษย์จะทำการติดต่อสื่อสารกันโดยการใช้เสียงพูด เพราะเป็นวิธีที่สะดวกและรวดเร็ว จึงมีความคิดที่จะใช้เสียงพูดในการติดต่อกันคอมพิวเตอร์แทนการใช้แป้นพิมพ์สำหรับการรับรู้เสียงพูดด้วยคอมพิวเตอร์จะมีปัญหาพื้นฐานอยู่ ๒ ประการ คือ

๑. ในแต่ละคำพูดของคนเราที่เปล่งออกมา จะมีลักษณะเฉพาะตัวเอง แม้ว่าเราจะพยายามเปล่งคำพูดให้มีเสียงใกล้เคียงกันเท่าใดก็ตาม

๒. เนื่องจากเสียงของคนเราที่ใช้ในการสื่อความหมายต่าง ๆ นั้น จะอยู่ในช่วงความถี่ ๔-๕ kHz และจากทฤษฎีการสุ่มตัวอย่าง (Sampling Theory) พบว่าในการที่จะรวบรวมความหมายของเสียงพูดออกมา ๑ วินาที ให้สมบูรณ์นั้น จะต้องใช้อัตราการสุ่มตัวอย่าง อย่างน้อย ๒ เท่าของความถี่เสียง ซึ่งจะเท่ากับ ๘-๑๐ kHz เป็นอย่างต่ำในการรับรู้เสียงพูดของเครื่องนั้นจำเป็นต้องใช้เสียงพูดเป็นหลายวินาทีและพูดหลายคน ดังนั้น จึงมีข้อมูลจำนวนมากที่จะต้องถูกเก็บและนำไปใช้ ซึ่งจะมีผลต่อสื่อที่ใช้ในการบันทึก และเวลาที่ใช้ในการรับรู้ เทคนิคของการวิเคราะห์ที่ได้พารามิเตอร์ของสัญญาณ เพื่อลดจำนวนข้อมูลจึงเป็นทางออกประการหนึ่งในการแก้ปัญหา

เราสามารถแบ่งวิธีการรับรู้เสียงพูดออกได้เป็น ๒ วิธี คือ

๑. แบบพิจารณาทั้งหน่วยภาษาที่เปล่งออกมาทั้งชุด (Isolated Word Recognition) หน่วยภาษาที่พิจารณาในวิธีนี้ อาจจะเป็นพยางค์เดี่ยว คำ กลุ่มคำ วลีหรือประโยค ก็ได้ วิธีนี้มีผลดีคือ จะหลีกเลี่ยงผลกระทบอันเนื่องมาจากรูปแบบของเสียงภายในคำหรือกลุ่มคำนั้น

๒. การรับรู้โดยแยกแยะรายละเอียดของหน่วยเสียง (Phonetic Recognition) ในวิธีนี้จะพิจารณาลักษณะของหน่วยเสียงที่มีขนาดเล็กลงไป เช่น ระดับสระ พยัญชนะ หรือ วารณยุกต์ เป็นต้น โดยจะใช้หน่วยเสียงย่อยเหล่านี้เป็นหลัก ในการเปรียบเทียบวิธีนี้จะมีความสะดวกสำหรับระบบที่จะรับรู้จำนวนคำมาก ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปรียบเทียบข้อดี ข้อเสียของการเลือกหน่วยเสียง แสดงดัง ตารางที่ ๑.๑

| ข้อดี   | ข้อเสีย   |
|---|---|
| <p><u>คำ (Word)</u></p> <ul style="list-style-type: none"> <li>- ไม่มีผลกระทบอันเกิดจากฐานของเสียงภายใน</li> <li>- ใช้ได้กับทุกภาษาและง่าย</li> </ul>               | <ul style="list-style-type: none"> <li>- จำนวนคำจำกัด</li> <li>- ยากต่อการปรับแต่ละคน</li> </ul>                                    |
| <p><u>พยางค์ (Syllable)</u></p> <ul style="list-style-type: none"> <li>- ไม่มีผลกระทบอันเกิดจากฐาน</li> <li>- ง่ายต่อการแยก</li> </ul>                              | <ul style="list-style-type: none"> <li>- จำนวนหน่วยเสียงหลักที่จะเปรียบเทียบเพิ่มขึ้น</li> <li>- ยากต่อการหาขอบเขตพยางค์</li> </ul> |
| <p><u>หน่วยเสียง (Phoneme)</u></p> <ul style="list-style-type: none"> <li>- จำนวนคำเปรียบเทียบในหน่วยความจำน้อย</li> <li>- ง่ายต่อการสร้างด้วยหน่วยเสียง</li> </ul> | <ul style="list-style-type: none"> <li>- ปัญหาการแยกขึ้นกับความหมาย</li> <li>- วิธีการหาและแยกมีมาก</li> </ul>                      |
| <p><u>หน่วยเสียงย่อย (Allophone)</u></p> <ul style="list-style-type: none"> <li>- ง่ายต่อการหาและตรวจพบ</li> </ul>  | <ul style="list-style-type: none"> <li>- จำนวนหน่วยเสียงหลักที่จะเปรียบเทียบสูง</li> </ul>  |

ตาราง ๑.๑ แสดงการเปรียบเทียบข้อดี ข้อเสียของการเลือกหน่วยภาษา

จากวิธีการทั้งสองเราสามารถจำแนกการรับรู้เสียงพูดออกได้ตามลักษณะของคำพูดที่รับรู้ คือ

- ก. การรับรู้คำเดี่ยว (Isolated Words)
- ข. การรับรู้คำพูดต่อเนื่อง (Continuous Words)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจำแนกได้ตามผู้พูด คือ

- ก. การรับรู้เสียงพูดเพียงคนเดียว (Speaker Dependent)
- ข. การรับรู้เสียงพูดแบบต่างบุคคล (Speaker Independent)

ในงานวิจัยมุ่งเน้น การรับรู้ตัวเลข 0-9 แบบบุคคลเพียงคนเดียว ซึ่งจะเกิดปัญหาที่เพิ่มขึ้นอีกหลายประการ เช่น ความเร็วในการออกเสียง การเน้นเสียง สิ่งแวดล้อมในการบันทึกเสียง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ ๒  
การวิเคราะห์เสียงพูด

ในการวิเคราะห์เสียงพูด สิ่งที่สำคัญคือการวิเคราะห์เสียงพูดจริง เพื่อหาคณะสมบัติหรือตัว  
แทนของเสียง ตามหลักเกณฑ์ต่างๆดังต่อไปนี้

2.1 เสียงพูด

คนเราเปล่งเสียงพูดด้วยอวัยวะที่ใช้ในการออกเสียง (organ of speech) ทำเสียงตาม  
ที่มีในระบบภาษาของตน แม้ว่าคนที่อยู่ในสังคมเดียวกันจะใช้ภาษาเดียวกันแต่ถ้าพิจารณาเสียงที่เปล่ง  
ออกมาจริงๆแล้ว แต่ละครั้งก็อาจจะสังเกตลักษณะที่แตกต่างกันได้ เราจึงสามารถจำเสียง จำวิธี  
พูดของคนที่เราคุ้นเคยได้ เสียงพูดที่จะอธิบายด้วยหลักเกณฑ์ทางวิทยาศาสตร์ แม้ว่าในภาษาหนึ่งๆจะมี  
เสียงต่างกันมากบ้างน้อยบ้าง แต่ละเสียงก็สามารถนำมาพิจารณาและอธิบายให้รู้ลักษณะการออกเสียง  
และตำแหน่งที่เกิดเสียงได้คำอธิบายนี้จะทำให้เข้าใจลักษณะเสียงทุกเสียง วิชาที่ว่าด้วยเสียงพูดเรียกว่า  
วิชาสัทศาสตร์ (Phonetics)

ในการศึกษาเสียงพูดแบ่งได้เป็น 2 ลักษณะ คือ

ก. สรีรศาสตร์ (Articulatory Phonetics) เป็นการศึกษาเสียงพูดจากอวัยวะและ  
การเคลื่อนไหวอวัยวะที่ทำให้เกิดเสียงพูด การอธิบายก็จะอธิบายโดยอาศัยลักษณะและอาการเคลื่อนไหว  
ไหวของอวัยวะที่เกี่ยวข้องในการเปล่งเสียงพูดนั้น

ข. กลศาสตร์ (Acoustic Phonetics) เป็นการศึกษาเสียงพูดจากลักษณะคลื่นเสียงที่  
ผู้พูดเปล่งออกมาแล้ว และผู้ฟังได้ยินว่ามีลักษณะทางกลศาสตร์อย่างไร การศึกษาตามแนวนี้ต้องอาศัย  
ความรู้ทางฟิสิกส์และคณิตศาสตร์ช่วยอธิบายลักษณะของคลื่นเสียง

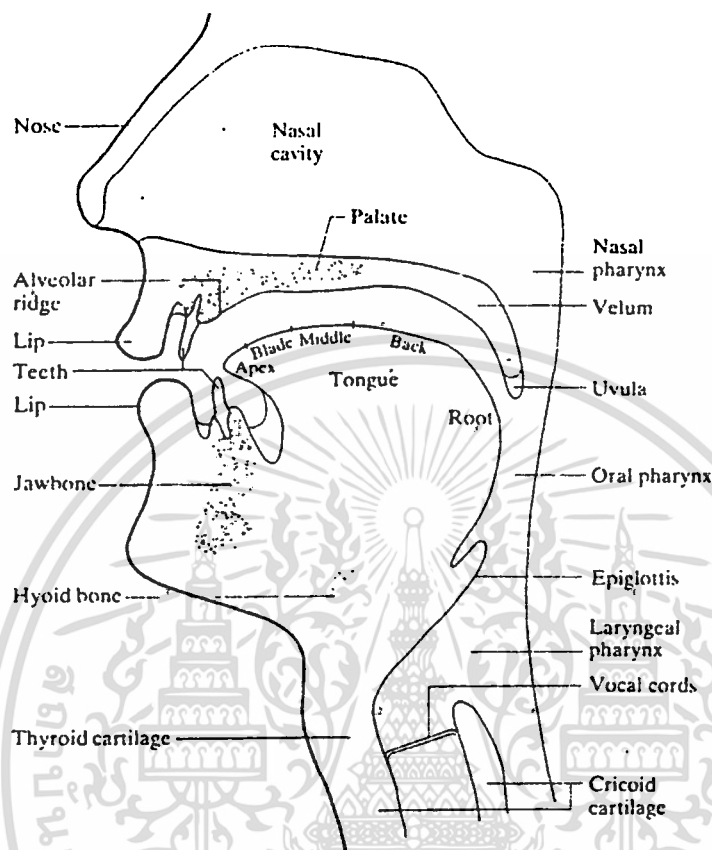
2.1.1 อวัยวะที่ใช้ในการออกเสียง

อวัยวะที่ใช้ในการออกเสียงมีอยู่หลายส่วน แต่ละส่วนสามารถทำให้เสียงพูดแตกต่างกันไปได้  
อวัยวะเหล่านี้มีปากและส่วนต่างๆในปาก ช่องคอ กล้องเสียง ช่องว่างในปากและช่องว่างในจมูก  
ดังในรูปที่ 2.1.1 อวัยวะที่ใช้ในการเปล่งเสียงพูดแบ่งออกเป็น 2 ประเภท คือ

ก. อวัยวะที่ใช้ในการทำอาการ (Articulator) คืออวัยวะที่เคลื่อนไหวเพื่อผลิตลมไปยัง  
ส่วนต่างๆอวัยวะที่สำคัญ คือ ลิ้น ซึ่งเป็นส่วนที่เคลื่อนไหวได้มากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พ. อวัยวะซึ่งเป็นตำแหน่งที่เกิดเสียงต่างๆ (Point Of Articulator) คือ ตำแหน่งที่เกิดเสียงต่างๆ เช่น ริมฝีปาก ฟัน เพดานส่วนต่างๆ เป็นต้น



รูปที่ 2.1.1 อวัยวะที่ใช้ในการออกเสียง

อวัยวะส่วนที่มีหน้าที่ในการออกเสียงโดยตรงมีดังนี้

ก. ริมฝีปาก เป็นอวัยวะส่วนที่เคลื่อนไหวได้มากและทำให้เสียงแตกต่างกันได้มาก เราอาจบังคับให้ริมฝีปากอยู่ติดกัน ห่างกัน ชื่นออกมา เป็นต้น ก็ได้ ลักษณะริมฝีปากแบบต่างๆนี้ ล้วนมีอิทธิพลต่อการออกเสียงและทำให้เสียงแตกต่างกันไปทั้งสิ้น

ข. ฟัน เป็นอวัยวะที่ทำให้เกิดเสียงหลายชนิด เช่น เมื่อฟันกดลงบนริมฝีปากล่างลมที่ผ่านออกมาโดยรูปร่างจะลอดช่องออกมาซึ่งทำให้เกิดเสียงได้

ค. ปุ่มเหงือก เป็นส่วนนูนออกมาอยู่หลังฟันด้านบน ถ้าเอาลิ้นแตะจะรู้สึกว่าเป็นคลื่น ปุ่มเหงือกเป็นบริเวณที่ทำให้เกิดเสียงปุ่มเหงือก (Alveolar Sound)

ง. เพดานแข็งหรือเพดานปาก คือส่วนของเพดานที่เป็นกระดูกแข็ง

จ. เพดานอ่อน คือส่วนเพดานที่ต่อเพดานแข็งไปข้างในมีลักษณะเป็นกระดูกอ่อนที่ขยับขึ้นลงได้ เวลาหายใจ เพดานอ่อนและลิ้นไก่ซึ่งอยู่ปลายเพดานอ่อนลดระดับลงมา เปิดช่องว่างให้ลมออกทางจมูก เวลาพูดส่วนใหญ่ปลายเพดานอ่อนและลิ้นไก่จะถูกยกขึ้นไปจรดหลังคอก้นจากเวลาออกเสียงนาสิกเท่านั้น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าวิธีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เสียงก้อง (voice) เกิดจากเส้นเสียงปิดคลื่นลมไว้ ลมที่ผ่านออกมาจะเพิ่มแรงดันอากาศขึ้นจนเส้นเสียงปิดเปิดสลับกันไป ทำให้เกิดเสียงก้องขึ้นมา ซึ่งสามารถเรียกความถี่ในการปิดเปิดของเส้นเสียงว่า ความถี่มูลฐาน (Fundamental Frequency)

2. เสียงไม่ก้อง (unvoice) เสียงชนิดนี้เส้นเสียงจะเปิดตลอดเวลาที่ลมผ่าน ลมจึงผ่านออกมาได้สะดวกทำให้เกิดเสียงไม่ก้องขึ้น

ขั้นตอนที่ 3 การเปลี่ยนแปลงลักษณะเสียง (Articulation) อวัยวะที่ใช้คือส่วนที่ต่อจากกล่องเสียงจนถึงริมฝีปาก โดยลมที่ผ่านออกจากกล่องเสียงจะทำให้เกิดเสียงในลักษณะต่างๆ ซึ่งจะเกิดจากการเปลี่ยนแปลงของอวัยวะที่กล่าวไว้ในหัวข้อที่แล้ว

## 2.2 การวิเคราะห์สัญญาณเสียง

ในการวิเคราะห์สัญญาณเสียงในขั้นแรกเราจะนำเอาสัญญาณเสียงซึ่งเป็นสัญญาณอนาลอกมาทำการแปลงให้อยู่ในรูปของสัญญาณดิจิทัลก่อน โดยที่การแปลงนี้เราสามารถเรียกอีกอย่างหนึ่งว่า การเข้ารหัสสัญญาณดิจิทัล (Digital Waveform Coding) แบ่งออกเป็น 2 วิธีคือ พัลส์โค้ดโมดูเลชัน (Pulse Code Modulation -PCM [อังกฤษ]) และ ดิฟเฟอเรนเชียลควอนไทเซชัน (Differential Quantization) เมื่อทำการแปลงสัญญาณด้วยวิธีใดวิธีหนึ่งเป็นที่เรียบร้อยแล้วเราจะนำข้อมูลเหล่านี้ไปทำการวิเคราะห์ โดยที่เราสามารถแบ่งการวิเคราะห์ได้เป็น 2 ลักษณะใหญ่ๆ คือ การวิเคราะห์ในโดเมนของเวลา และการวิเคราะห์ในโดเมนความถี่

### 2.2.1 การวิเคราะห์ในโดเมนเวลา (Time Domain Analysis)

เนื่องจากเสียงพูดมีการเปลี่ยนแปลงคุณสมบัติตามเวลา แต่ถ้าเราแบ่งเสียงพูดออกเป็นช่วงๆ (frame) สั้นๆ เสียงจะมีการเปลี่ยนแปลงคุณสมบัติน้อยมาก ทำให้การวิเคราะห์ทำได้ง่ายขึ้นการแบ่งสัญญาณเสียงพูดที่จะนำมาวิเคราะห์ออกเป็นช่วงสั้นๆสามารถเขียนเป็นสมการได้คือ

$$Q_n = \sum_{m=-\alpha}^{\alpha} T[x(m)] * w(n-m) \quad \dots (2.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ข. บาร์ทเลทวินโดว์ (Bartlett window)

$$w(n) = 2n/(N-1) \quad , \quad 0 \leq n \leq (N-1)/2$$

$$= (2-2n)/(N-1) \quad , \quad (N-1)/2 \leq n \leq N-1 \quad \dots (2.3)$$

ค. แฮนนิ่งวินโดว์ (Hanning windows)

$$w(n) = [1 - \cos(2\pi n / (n-1))] / 2 \quad , \quad 0 \leq n \leq N-1 \quad \dots (2.4)$$

ง. แฮมมิงวินโดว์ (Hamming windows)

$$w(n) = 0.54 - 0.46 \cos(2\pi n / (N-1)) \quad , \quad 0 \leq n \leq N-1 \quad \dots (2.5)$$

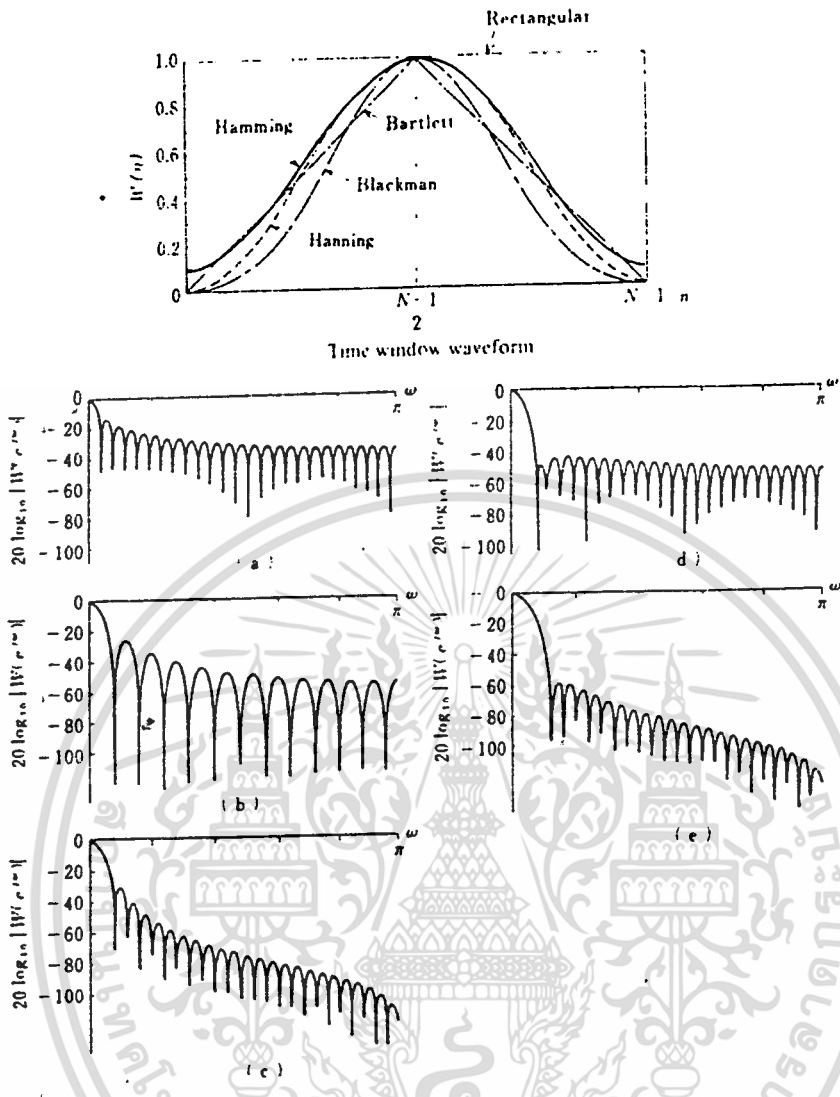
จ. แบล็คแมนวินโดว์ (Blackman windows)

$$w(n) = 0.42 - 0.5 \cos(2\pi n / (N-1)) \quad , \quad 0 \leq n \leq N-1 \quad \dots (2.6)$$

วินโดว์แต่ละชนิดแสดงได้ดังรูป 2.2.2

จากรูปที่ 2.2.2 สามารถเปรียบเทียบสเปกตรัมของวินโดว์แต่ละชนิดได้และสามารถสรุปคุณลักษณะของวินโดว์แต่ละชนิดได้ดังนี้

- ก. วินโดว์รูปสี่เหลี่ยม พบว่ามีเมนโลบกว้างที่สุด แต่ไซด์โลบก็มีค่าสูงที่สุดเช่นเดียวกัน (ประมาณ -20 dB)
- ข. บาร์ทเลทวินโดว์ พบว่ามีเมนโลบแคบแต่ไซด์โลบมีค่าสูง
- ค. แฮนนิ่งวินโดว์ ไซด์โลบแรกและไซด์โลบที่สองมีค่าสูงแต่ที่ไซด์โลบอื่นๆมีค่าต่ำ
- ง. แฮมมิงวินโดว์ พบว่าไซด์โลบมีค่าต่ำ (-40 dB) และเมนโลบแคบพอใช้
- จ. แบล็คแมนวินโดว์ พบว่าไซด์โลบมีขนาดเล็กมาก (< -60 dB) แต่เมนโลบกว้างมาก



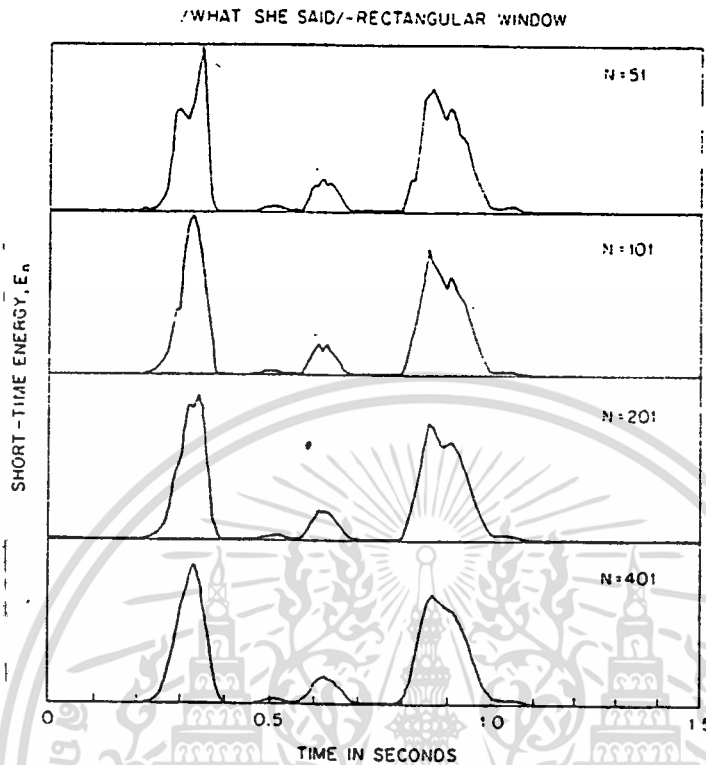
รูปที่ 2.2.2 แสดงโดเมนเวลาและสเปกตรัมของวินโดว์แต่ละชนิด

จากที่พิจารณาลักษณะวินโดว์แต่ละชนิดพบว่า วินโดว์ที่ควรจะถูกใช้ในการวิเคราะห์เสียงคือ **แฮมมิงวินโดว์**

ในการวิเคราะห์เสียงโดยใช้วินโดว์ฟังก์ชัน จะพบว่าสัญญาณเสียงที่ผ่านการกรองโดยวินโดว์ นั้นจะมีการแกว่งขึ้นลงมากน้อยขึ้นอยู่กับช่วงเวลาของวินโดว์ (ความกว้างของวินโดว์) คือถ้าช่วงเวลาของวินโดว์สั้น แอ็กทิวของสัญญาณจะมีการแกว่งขึ้นลงอย่างรวดเร็ว และถ้าช่วงเวลาของวินโดว์มาก แอ็กทิวของสัญญาณจะมีการแกว่งขึ้นลงช้าๆ ดังแสดงในรูปที่ 2.2.3

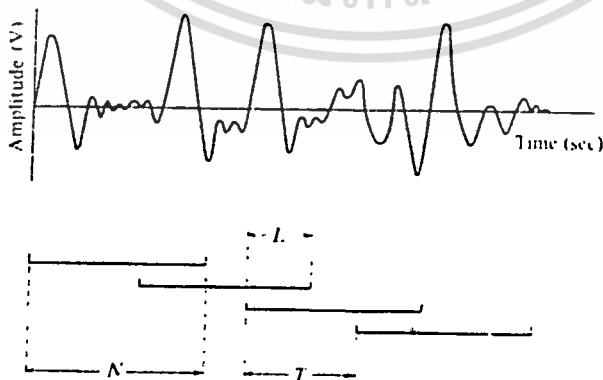
ดังนั้นการเลือกช่วงเวลาของวินโดว์จึงต้องทำการเลือกให้อยู่ในช่วงที่เหมาะสม คือไม่ให้แอ็กทิวของสัญญาณแกว่งช้าหรือเร็วเกินไป ซึ่งจากการทดลองพบว่าอยู่ในช่วงระหว่าง 10-30 มิลลิวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2.3 แสดงการเปลี่ยนแปลงเมื่อใช้ขนาดวินโดว์ต่างกัน

เนื่องจากช่วงเวลาของเสียงพูดมีระยะมากกว่าช่วงเวลาของวินโดว์ ดังนั้นการวิเคราะห์สัญญาณเสียงพูดทั้งหมด จะอาศัยหลักการคาบเกี่ยวของช่วงเวลาของวินโดว์ แสดงดังรูปที่ 2.2.4 คือจุดเริ่มต้นของเฟรมปัจจุบันกับจุดเริ่มต้นของเฟรมที่เกิดขึ้นก่อน 1 เฟรม ต้องอยู่ห่างกัน 10-20 ms ซึ่งเรียกช่วงเวลานี้ว่าช่วงเวลาในการคาบเกี่ยว (interval shift)



รูปที่ 2.2.4 แสดงการคาบเกี่ยวของช่วงเวลาของวินโดว์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์เสียงพูดในโดเมนเวลาช่วงสั้นๆ สามารถทำการวิเคราะห์ได้หลายวิธี เช่น

2.2.1.1 การวิเคราะห์โดยวัดระดับพลังงานและขนาดเฉลี่ยโดยวัดในช่วงเวลาสั้นๆ  
(Short-Time Energy and Average Magnitude)

การพิจารณาขนาดของเสียงในโดเมนของเวลาเราพบว่า ขนาดของสัญญาณเสียงไม่ก้องจะมีค่าต่ำกว่าขนาดของสัญญาณเสียงก้อง เราสามารถใช้ระดับพลังงานของเสียงพูดในช่วงเวลาสั้นๆ ในการพิจารณาขนาดของสัญญาณเสียงเหล่านี้ได้โดยทั่วไปเราสามารถกำหนดสมการพลังงานของเสียงพูดในช่วงเวลาสั้นๆ ได้ดังนี้

$$E_n^\alpha = \sum_{m=-\alpha}^{\alpha} [x(m) * w(n-m)]^2 \quad \dots (2.7)$$

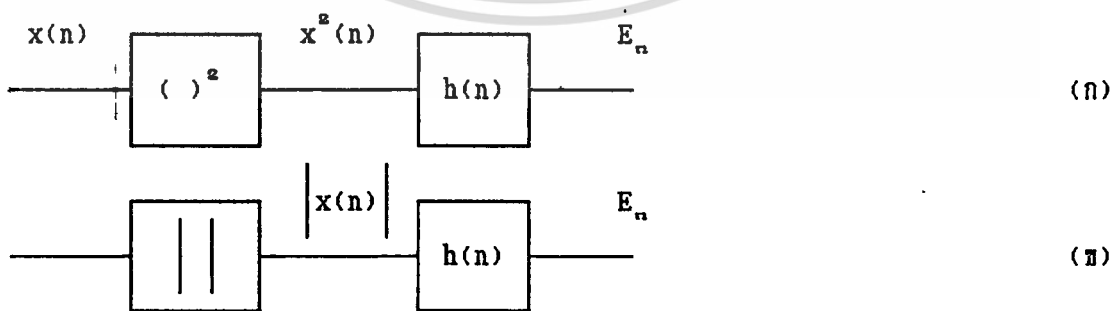
สามารถเขียนได้เป็น

$$E_n^\alpha = \sum_{m=-\alpha}^{\alpha} x^2(m) h(n-m)^2 \quad \dots (2.8)$$

โดยที่

$$h(n) = w^2(n) \quad \dots (2.9)$$

จากสมการ (2.8) สามารถแสดงได้ดังรูปที่ 2.2.5(ก) ซึ่งพบว่าสัญญาณ  $x^2(n)$  จะถูกกรองโดยวงจรกรองความถี่เชิงเส้น (linear filter) ที่มีผลตอบสนอง  $h(n)$  ดังสมการ (2.5) และวินโดว์ที่ใช้ควรเป็นแฮมมิงวินโดว์



รูปที่ 2.2.5 แสดงบล็อกไดอะแกรม (ก) การวิเคราะห์ระดับพลังงานในช่วงเวลาสั้น

(ข) การวิเคราะห์ขนาดเฉลี่ยในช่วงเวลาสั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการ (2.7) มีข้อเสียคือจะมีความไวต่อสัญญาณที่มีขนาดใหญ่เนื่องจากการใช้ยกกำลังสอง ดังนั้นจึงมีการดัดแปลงใหม่โดยใช้ค่าสัมบูรณ์คือ

$$E_n = \sum_{m=n-N+1}^n |x(m)|^2 w(n-m) \quad \dots (2.10)$$

จากสมการ (2.10) สามารถแสดงได้ดังรูป 2.2.4(ข)

### 2.2.1.2 การวิเคราะห์อัตราการตัดศูนย์ (Zero Crossing Rate)

การวิเคราะห์การตัดศูนย์เกิดจากการที่รูปคลื่นของสัญญาณมีการตัดกับแกนของเวลา นั่นคือค่าของสัญญาณจะมีการเปลี่ยนแปลงสัญลักษณ์ทางคณิตศาสตร์ (Algebraic sign) อัตราการตัดศูนย์จะใช้สำหรับวัดความถี่ในการเปลี่ยนแปลงข้อมูลของสัญญาณ โดยการนำเอาค่าตัดศูนย์ที่ได้มาพิจารณาว่าเป็นเสียงก้องหรือไม่ก้อง เนื่องจากเสียงก้องส่วนใหญ่จะมีค่าพลังงานอยู่ที่ความถี่ต่ำ ค่าอัตราการตัดศูนย์ก็จะมีค่าต่ำ ส่วนเสียงไม่ก้อง หลังงานส่วนใหญ่อยู่ที่ความถี่สูง ค่าอัตราการตัดศูนย์ก็จะมีค่าสูงด้วย แต่อย่างไรก็ตามการกำหนดอัตราการตัดศูนย์ที่แน่นอน เพื่อจำแนกชนิดของเสียงนั้นจะอาศัยการทดลองเป็นหลัก รูปที่ 2.2.6 แสดงฮิสโตแกรมของค่าการตัดศูนย์ของสัญญาณเสียงก้องและเสียงไม่ก้อง โดยจะเห็นว่าค่าการตัดศูนย์ของเสียงไม่ก้องมีค่าเป็น 49 ต่อ 10 มิลลิวินาที ส่วนค่าการตัดศูนย์ของสัญญาณเสียงก้องจะอยู่ที่ 14 ต่อ 10 มิลลิวินาที

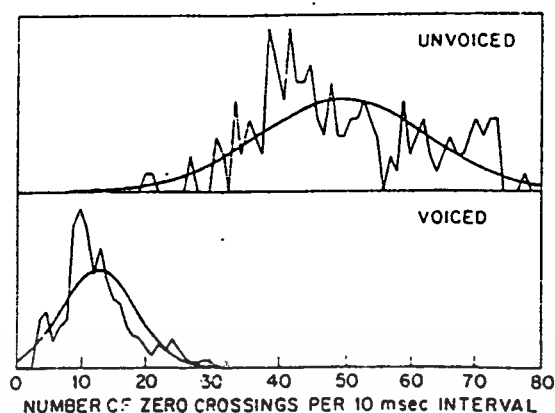
ในการหาค่าอัตราการตัดศูนย์สามารถทำได้โดยสมการดังนี้

$$T[s(n)] = 0.5 \left| \text{Sign}(s(n)) - \text{Sign}(s(n-1)) \right| \quad \dots (2.11)$$

โดยที่

$$Z = \frac{1}{2N} * \sum_{n=1}^N \left| \text{Sign}(s(n)) - \text{Sign}(s(n-1)) \right|$$

$$\begin{aligned} \text{Sign}(s(n)) &= 1 \text{ เมื่อ } s(n) > 0 \\ &= 0 \text{ ที่อื่น ๆ} \end{aligned}$$



รูปที่ 2.2.5 แสดงถึงอีโตรแกรมของอัตราการตัดศูนย์ของสัญญาณเสียงก้องและเสียงไม่ก้อง

### 2.2.1.3 การวิเคราะห์หัตถสัมพันธ์ในช่วงเวลาสั้น (The Short-Time Autocorrelation Function)

ฟังก์ชันหัตถสัมพันธ์เป็นวิธีการที่ใช้ในการวิเคราะห์คุณสมบัติของเสียงในโดเมนเวลา เป็นวิธีหนึ่งที่มีความสะดวกคล่องแคล่ว โดยสมการกำหนดฟังก์ชันหัตถสัมพันธ์ของสัญญาณที่ไม่ต่อเนื่องคือ

$$\phi(k) = \sum_{m=-\alpha}^{\alpha} x(m)x(m+k) \quad \dots(2.12)$$

แต่ถ้าสัญญาณเป็นแรนดอมหรือเป็นคาบ ฟังก์ชันหัตถสัมพันธ์สามารถกำหนดใหม่ได้เป็น

$$\phi(k) = \lim_{N \rightarrow \infty} \frac{1}{(2N+1)} \sum_{m=-N}^N x(m)x(m+k) \quad \dots(2.13)$$

ถ้าสัญญาณมีลักษณะที่เป็นคาบ โดยในหนึ่งคาบมีจำนวนการสุ่ม  $P$  ตัวอย่างจะได้

$$\phi(k) = \phi(h+P) \quad \dots(2.14)$$

จากสมการ (2.14) หมายถึง ถ้าสัญญาณที่มีลักษณะเป็นคาบ  $x(m) = x(m+P)$  ที่  $m$  ใดๆ ที่นำมาคำนวณหาค่าหัตถสัมพันธ์ ค่าหัตถสัมพันธ์ที่จะมีลักษณะเป็นคาบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติที่สำคัญของฟังก์ชันอັคสสัมพันธ์

1. เป็นฟังก์ชันคู่ คือ  $\phi(k) = \phi(-k)$

2. จะได้ค่าสูงสุดที่  $k=0$  คือ  $|\phi(k)| \leq \phi(0)$  ;  $k$  คือค่าใดๆที่ไม่เป็นศูนย์

3. ค่า  $\phi(0)$  เป็นค่าพลังงานที่ใช้ในการพิจารณาสัญญาณหรือเป็นกำลังงานเฉลี่ยสำหรับสัญญาณที่เป็นคาบ

ฟังก์ชันอັคสสัมพันธ์ที่ใช้ในการวิเคราะห์สัญญาณ ถ้าพิจารณาสมการ (2.14) กับคุณสมบัติข้อ (1) และ (2) พบว่าถ้าสัญญาณมีลักษณะเป็นคาบ ค่าอັคสสัมพันธ์จะมีค่าสูงสุดที่แซมเปิลที่ 0,  $P_{2P}$  ซึ่งสามารถหาคาบได้จากการคำนวณระยะห่างของค่าเหล่านี้

เนื่องจากการวิเคราะห์ที่เลื่องลือจำเป็นต้องวิเคราะห์ในช่วงเวลาสั้นๆ ดังนั้นจึงสามารถกำหนดฟังก์ชันอັคสสัมพันธ์ใหม่ได้ดังนี้

$$R_{nn}(k) = \sum_{m=0}^{N-1-k} [x(n+m)w(m)][x(n+m+k)w(m+k)] \quad \dots (2.15)$$

โดยที่วินโดว์ที่นิยมใช้เป็นวินโดว์รูปสี่เหลี่ยม เพราะเมื่อคำนวณค่าอັคสสัมพันธ์แล้วจะได้ตำแหน่งของคาบสัญญาณที่ชัดเจนกว่า

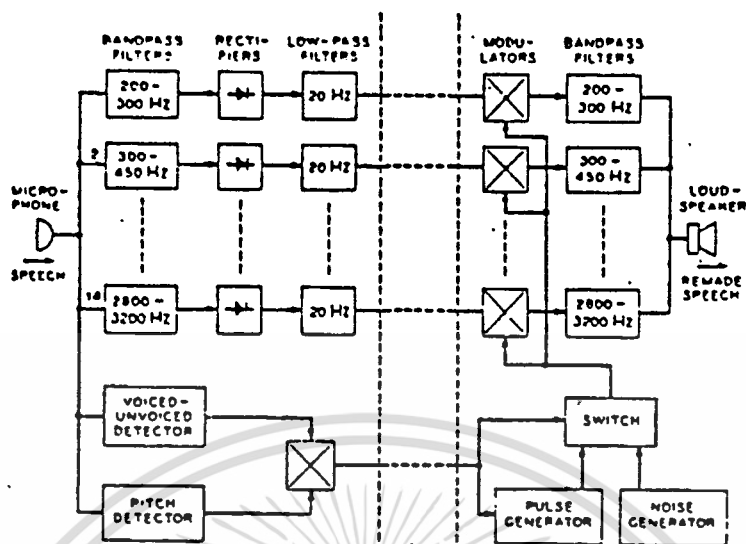
## 2.2.2 การวิเคราะห์ทางความถี่หรือสเปกตรัม (Frequency Domain Analysis)

หลักการนำอาศัยการวิเคราะห์ส่วนประกอบทางความถี่ (Spectrum) ของคลื่นเสียงได้แก่

### 2.2.2.1 Channel Vocoder

channel vocoder ประกอบด้วยสองภาค คือ ภาควิเคราะห์ (Analyzer) และภาคสังเคราะห์ (Synthesizer) ดังในรูปที่ 2.2.6 ภาควิเคราะห์ประกอบด้วย วงจรแบนด์พาสฟิลเตอร์คลุมย่านความถี่ระหว่าง 200-3200 Hz จำนวน 14 ช่อง ในแต่ละช่องจะมีวงจร เรกติฟายเออร์ (Rectifier) เพื่อดึงเอาเฉพาะ Envelope ของสัญญาณที่ผ่าน แบนด์พาสฟิลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2.6 ส่วนประกอบของ Channel Vocoder

ตัวตรวจสอบเสียงก้อง (Voice-Unvoice Detector) จะทำหน้าที่ตัดสินใจว่าเสียงพูดขณะนั้นเป็นเสียงก้อง หรือไม่ก้อง ตัวตรวจสอบคาบ (Pitch Detector) จะทำหน้าที่หาความถี่หลักมูล (Fundamental Frequency) ของเสียงขณะนั้น ภาคสังเคราะห์ประกอบด้วย มอดคูลเลเตอร์ ทำหน้าที่คูณหรือปรับขนาดของสัญญาณจากตัวกำเนิดพัลส์ หรือ สัญญาณรบกวน เพื่อป้อนเข้าสู่แบนด์พาสฟิลเตอร์ในแต่ละช่อง ตัวคูณ คือค่าที่ส่งมาจากภาควิเคราะห์ เสียงพูดจะได้จากสัญญาณที่ผ่าน แบนด์พาสฟิลเตอร์ทุกช่องนำมารวมกัน วิธีการนี้ถูกนำไปสร้างเป็นเครื่องวิเคราะห์-สังเคราะห์เสียงพูดสำเร็จเป็นครั้งแรกเมื่อ ปี 1939 โดย Homer W. Dudley ไม่นานมานี้ได้มีการประยุกต์วิธีการโดยหันมาใช้ Short-time Fast Fourier Transform ทำการคำนวณด้วยเครื่องคอมพิวเตอร์ ซึ่งสามารถใช้งานได้ทั้งวิเคราะห์และสังเคราะห์ แต่มีข้อเสียคือต้องมีการคำนวณเป็นจำนวนมากและข้อมูลเสียงยังมีจำนวนมาก ปัจจุบันนิยมใช้วิธีการนี้สำหรับวิเคราะห์เสียงพูด เช่น ใช้ในเครื่อง Sono Graph หรือ Spectrum Analyzer

### 2.2.2.2 การวิเคราะห์ฟูเรียร์ในช่วงเวลาสั้น (Short-time Fourier Analysis)

เป็นเทคนิคที่ใช้ในการหาความสัมพันธ์ระหว่าง ฟังก์ชันโดเมนเวลากับฟังก์ชันโดเมนความถี่ โดยเริ่มจากการหาการแปลงของฟูเรียร์สำหรับสัญญาณต่อเนื่องในรูปของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X(f) = \int_{-\alpha}^{\alpha} x(t) e^{-2\pi f t} dt \quad \dots (2.16ก)$$

$$x(t) = \int_{-\alpha}^{\alpha} X(f) e^{2\pi f t} df \quad \dots (2.16ข)$$

เมื่อ  $-\alpha < f < \alpha$ ,  $-\alpha < t < \alpha$  และ  $i = -1$  โดยที่  $X(f)$  จะหมายถึงฟังก์ชันโดเมนความถี่ และ  $x(t)$  หมายถึงฟังก์ชันของโดเมนเวลา ในลักษณะเดียวกัน ถ้าเราทำการแปลงรูปคลื่นในโดเมนเวลาที่มีลักษณะเป็นฟังก์ชันแบบไม่ต่อเนื่อง และมีช่วงที่จำกัด ลักษณะการแปลงแบบนี้ เรียกว่า การแปลงแบบดิสครีตฟูริเยร์ (Discrete Fourier Transform) สามารถเขียนได้ในรูปของ

$$X(j) = (1/N) \sum_{k=0}^{N-1} x(k) e^{-j2\pi k/M} \quad \dots (2.17ก)$$

$$x(k) = \sum_{j=0}^{N-1} X(j) e^{j2\pi k/M} \quad \dots (2.17ข)$$

เมื่อ  $j=0,1,\dots,N-1$  และ  $k=0,1,\dots,N-1$  โดยที่ทั้ง  $X(j)$  และ  $x(k)$  เป็นอนุกรมเชิงซ้อน เราสามารถเขียนสมการข้างต้นใหม่ โดยการให้  $W_N = e^{2\pi j/M}$  คือ

$$X(f) = (1/N) \sum_{j=0}^{N-1} x(k) W_N^{-jk} \quad \dots (2.18ก)$$

$$x(k) = \sum_{j=0}^{N-1} X(j) W_N^{jk} \quad \dots (2.18ข)$$

ซึ่งสามารถแสดงตัวอย่างได้ คือ ถ้าเราจะแปลงรูปของข้อมูล 256 จุด ( $N$ ) ของสัญญาณที่มีอัตราการสุ่มที่ 8 กิโลเฮิรตซ์ เราจะได้ 256 ช่วงความถี่ระหว่าง 0 ถึง 8 กิโลเฮิรตซ์ ดังแสดงในตารางที่ 2.1 จากตารางจะเห็นว่า ทุกๆ 31 เฮิรตซ์ในโดเมนความถี่ หรือ ทุกๆ 32 มิลลิวินาทีของเสียงที่จะถูกแปลงรูป ได้ข้ออาศัยความสัมพันธ์ข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| โดเมนเวลา   |                      | โดเมนความถี่ |            |
|-------------|----------------------|--------------|------------|
| ตัวอย่างที่ | เวลา $\mu\text{sec}$ | ตัวอย่างที่  | ความถี่ Hz |
| 0           | 0                    | 0            | 0          |
| 1           | 125                  | 1            | 31         |
| 2           | 250                  | 2            | 62         |
| ...         |                      | ...          |            |
| 254         | 31,750               | 254          | 7938       |
| 255         | 31,875               | 255          | 7969       |

ตารางที่ 2.1 แสดงถึงตัวอย่างความสัมพันธ์ของโดเมนความถี่กับโดเมนเวลา

### 2.2.3 การหาขอบเขตของคำ (EndPoint detection)

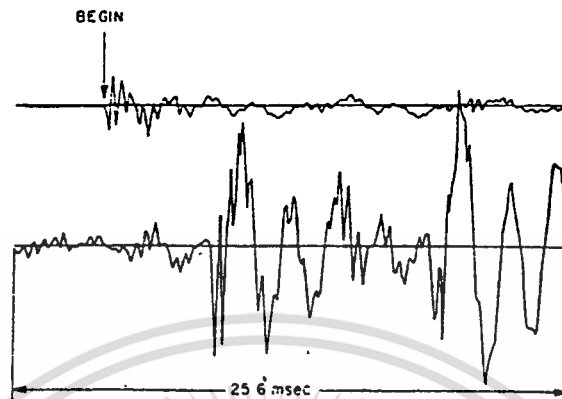
จากการบันทึกเสียงพูดลงไปในหน่วยความจำคอมพิวเตอร์ ข้อมูลที่ปรากฏอยู่ในหน่วยความจำจะประกอบไปด้วย ส่วนของเสียงพูดที่แท้จริง กับส่วนของเสียงรบกวน (background noise) ดังนั้นก่อนที่จะนำข้อมูลเหล่านี้ไปทำการวิเคราะห์พารามิเตอร์ จะต้องทำการแยกเสียงพูดที่แท้จริงออกจากเสียงรบกวน ซึ่งการแยกเสียงพูดที่เราสามารถเรียกอีกอย่างได้ว่าเป็นการหาขอบเขตของคำพูดที่แท้จริง สาเหตุที่ต้องทำการหาขอบเขตของคำ ก็เพื่อที่จะสามารถวิเคราะห์พารามิเตอร์ของเสียงพูดได้อย่างรวดเร็วและถูกต้อง และทำให้การตรวจรู้มีความถูกต้องมากยิ่งขึ้น

การหาขอบเขตของคำของเสียงพูดที่มีอัตราส่วนของสัญญาณเสียงต่อสัญญาณเสียงรบกวน (S/N) สูงคือแทบจะไม่มีเสียงรบกวนเลย เช่น เสียงที่ทำการบันทึกในห้องบันทึกเสียงจะพบว่าระดับพลังงานของเสียงที่ต่ำที่สุด จะมีค่ามากกว่าเสียงรบกวนอย่างเห็นได้ชัด ดังนั้นจึงสามารถทำการหาขอบเขตของคำๆ นั้นได้อย่างง่ายดาย ดังแสดงในรูป 2.2.7

แต่ในความเป็นจริงแล้ว การบันทึกเสียงพูดจะมีเสียงรบกวนอยู่เสมอ ทำให้การหาขอบเขตของคำทำได้ยากขึ้น และสิ่งในกรณีที่จุดเริ่มต้นของเสียงและจุดสุดท้ายของเสียงมีลักษณะ

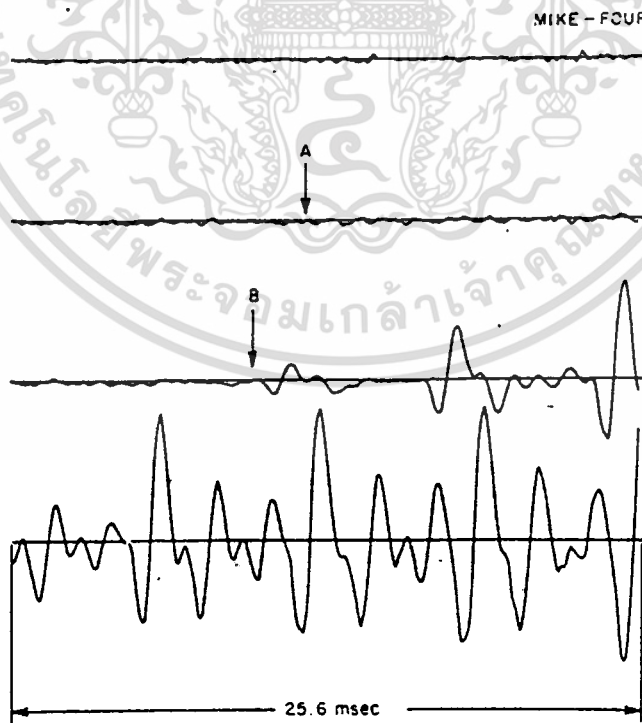
1. เป็นเสียงแทรก (fricative) อย่างอ่อน เช่น /f/, /th/, /h/
2. เป็นเสียงระเบิด (plosive) อย่างอ่อน เช่น /p/, /t/, /k/
3. เสียงนาสิก (nasal)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.2.7 แสดงจุดเริ่มต้นของเสียง /eight/

ทำให้การพิจารณาขอบเขตของคำเหล่านั้นยากขึ้นไปอีก ดังจะเห็นได้ดังรูปที่ 2.2.8 ซึ่งแสดงสัญญาณเสียงของคำว่า /four/ จะเห็นได้ว่า จุดเริ่มต้นของคำๆนี้ เป็นเสียงแทรกอย่างอ่อน (/f/) ทำให้การหาจุดเริ่มต้นและจุดสุดท้ายของเสียงทำได้ยากมาก เมื่อพิจารณาตำแหน่ง A และ B ในรูป จะพบว่า จุด B น่าจะเป็นจุดเริ่มต้นของคำ แต่ที่จริงแล้วจุดเริ่มต้นของคำจะอยู่ที่จุด A



รูปที่ 2.2.8 แสดงจุดเริ่มต้นของคำว่า /four/

การหาขอบเขตของคำ สามารถแยกได้เป็น 3 ประเภท เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การหาขอบเขตแบบเอ็กพลีซิท (explicit)
2. การหาขอบเขตแบบอิมพลีซิท (implicit) [IEEE endpoint]
3. การหาขอบเขตของค่าแบบไฮบริดจ์ (hybridge) [IEEE endpoint]

แต่ในวิกิฮานินท์ฉบับนี้ จะใช้การหาค่าแบบเอ็กพลีซิท ซึ่งจะเป็นการพิจารณาหาขอบเขตของค่าในโดเมนเวลาซึ่งมีขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 ใช้การวิเคราะห์ขนาดสัญญาณ ดังหัวข้อ (2.2.1.1) ในการหาขอบเขตค่า ซึ่งมีวิธีการดังต่อไปนี้

1. ค่ารวมหาขนาดของสัญญาณในช่วงเวลาที่ถูกบันทึกทั้งหมด โดยการคำนวณหาขนาดของสัญญาณจะคำนวณเป็นเฟรม กำหนดให้ frame หนึ่งมีขนาด 10 msec และทำการคำนวณไปเรื่อยๆจนครบช่วงเวลาของเสียงที่ถูกบันทึกทั้งหมด แสดงวิธีคำนวณจากรูป 2.2.9

2. กำหนดระดับอ้างอิง (ITL) ซึ่งระดับอ้างอิงนี้สามารถคำนวณได้จาก 100 ms แรกของช่วงเวลาทำการบันทึกทั้งหมด ซึ่งเราถือว่าช่วง 100 ms แรกเป็นช่วงที่ไม่มีสัญญาณเสียงรบกวนอยู่เลย

3. กำหนดระดับอ้างอิง (ITU) โดยที่

$$ITU = 3 * ITL$$

ระดับอ้างอิงนี้จะเป็นตัวบอกว่า จุดเริ่มต้นและจุดสุดท้ายของค่าจะอยู่นอกช่วงเวลา

4. เมื่อกำหนดระดับอ้างอิง ITU เป็นที่เรียบร้อยแล้ว ก็จะทำการศึกษาจุดเริ่มต้นของค่า โดยพิจารณาขนาดของสัญญาณค่าแรกที่มีค่ามากกว่าระดับอ้างอิง ITU จากนั้นก็จะทำการพิจารณาขนาดของสัญญาณย้อนกลับ จนกว่าขนาดของสัญญาณจะมีค่าน้อยกว่าระดับอ้างอิง ITL โดยจะถือว่าขนาดของสัญญาณค่าแรกที่มีค่าน้อยกว่าระดับอ้างอิง ITL เป็นจุดเริ่มต้นของสัญญาณ (N1)

5. การหาจุดสุดท้ายก็ทำเช่นเดียวกับข้อ 4 แต่แทนที่จะพิจารณาย้อนกลับก็จะทำการพิจารณาไปข้างหน้า โดยเราจะถือว่าขนาดของสัญญาณแรกที่มีค่าน้อยกว่าระดับอ้างอิง ITL เป็นจุดสุดท้าย (N2)

จากขั้นตอนทั้งหมดสามารถพิจารณารูป 2.2.11 ประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 ใช้การวิเคราะห์หาค่าอัตราการตัดศูนย์ (2.2.1.2) สาเหตุที่ใช้วิธีนี้มาช่วยในการหาขอบเขตของค่าเนื่องจาก การใช้ขนาดของสัญญาณเพียงอย่างเดียวในการหาขอบเขตของค่ายังทำได้ไม่ถูกต้องมากนัก โดยเฉพาะในกรณีที่มีเสียงแทรกอยู่ที่จุดเริ่มต้นและจุดสุดท้ายของค่า ซึ่งวิธีนี้จะทำให้ขอบเขตของค่าที่หาได้มีความถูกต้องมากขึ้น ซึ่งมีวิธีการดังต่อไปนี้

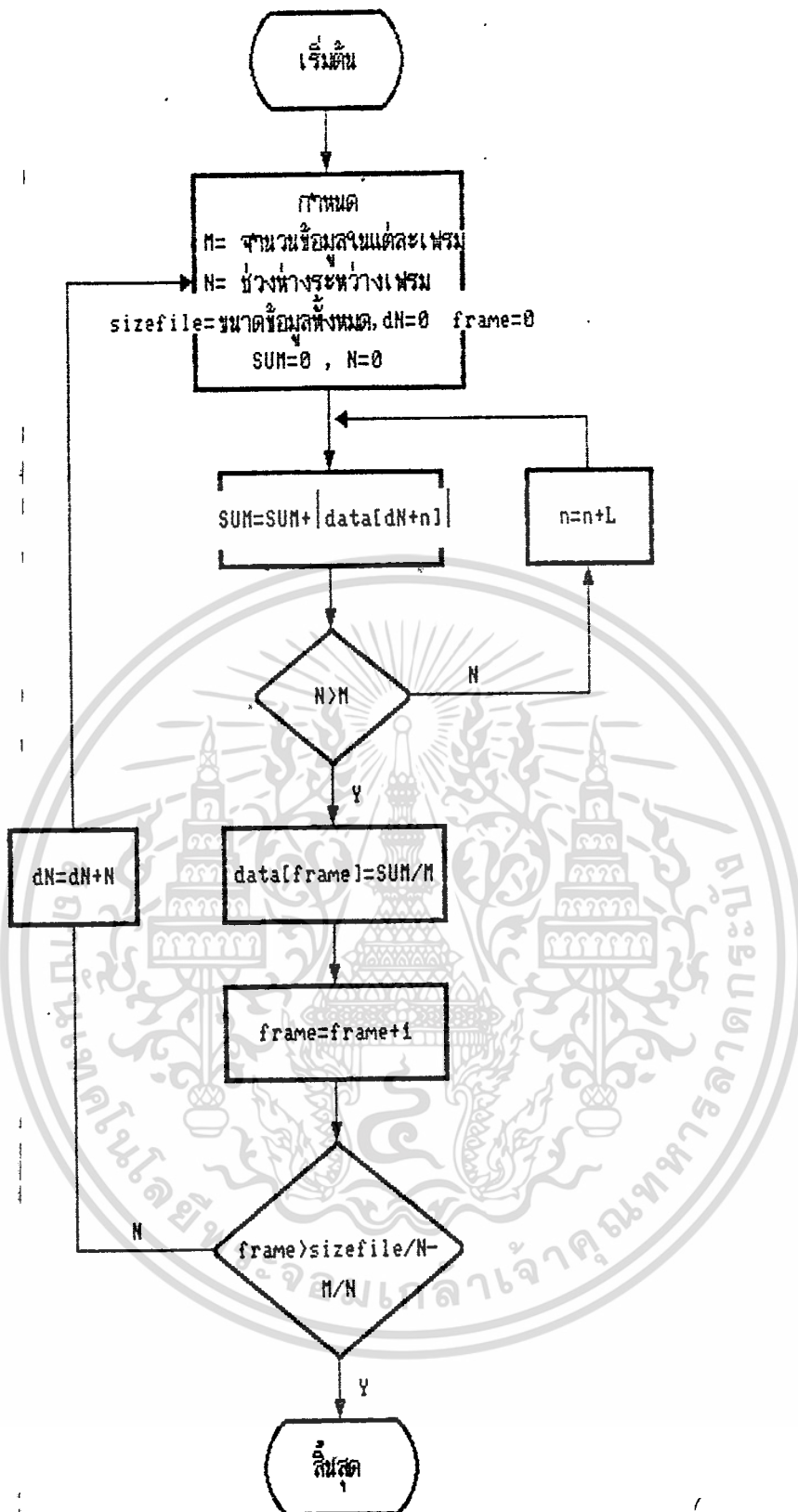
1. คำนวณหาอัตราการตัดศูนย์ในช่วงเวลาที่ถูกบันทึกทั้งหมด โดยการคำนวณหาอัตราการตัดศูนย์ จะคำนวณเป็นเฟรม กำหนดให้เฟรมหนึ่งๆมีขนาด 10 มิลลิวินาที และทำการคำนวณไปเรื่อยๆจนครบช่วงเวลาของเสียงที่ถูกบันทึกทั้งหมด พิจารณาได้จากรูป 2.2.10

2. กำหนดระดับอ้างอิง (IZCT) ซึ่งเป็นการคำนวณหาอัตราการตัดศูนย์ในช่วงเวลา 100 ms แรกของช่วงเวลาที่ทำการบันทึกทั้งหมด

3. จากจุดเริ่มต้น (N1) ที่หาได้จากขั้นตอนที่ 1 เราจะทำการพิจารณาค่าอัตราการตัดศูนย์ย้อนกลับไป 25 ค่าจากจุด N1 แล้วพิจารณาว่าที่จุดนั้นค่าของอัตราการตัดศูนย์มีค่ามากกว่า 3 เท่าของระดับอ้างอิง IZCT หรือไม่ ถ้ามีค่ามากกว่า N1 จะถูกเปลี่ยนตำแหน่ง โดยจะถูกเลื่อนไปยังจุดแรกที่มีค่ามากกว่าระดับอ้างอิง IZCT และจะถือว่าจุดนี้เป็นจุดเริ่มต้นที่แท้จริง แต่ในกรณีที่มามีค่าน้อยกว่า จะถือว่า N1 ที่หาได้จากขั้นตอนที่ 1 เป็นจุดเริ่มต้นที่แท้จริง

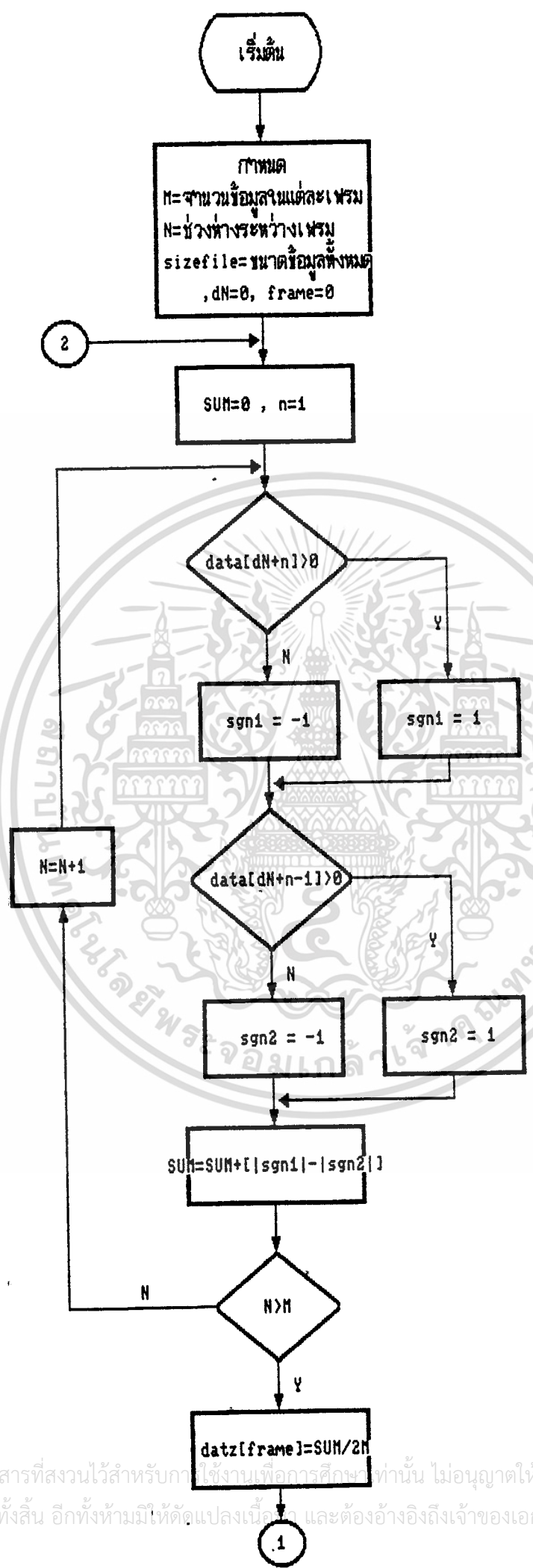
4. จากจุดสุดท้าย (N2) ที่หาได้จากขั้นตอนที่ 1 จะพิจารณาเหมือนข้อ 3 แต่จะพิจารณาข้อมูลค่าอัตราการตัดศูนย์ไปข้างหน้า 25 ค่า จากจุด N2

โดยที่การทำงานของทั้งสองขั้นตอนสามารถแสดงได้ดังรูปที่ 2.2.11

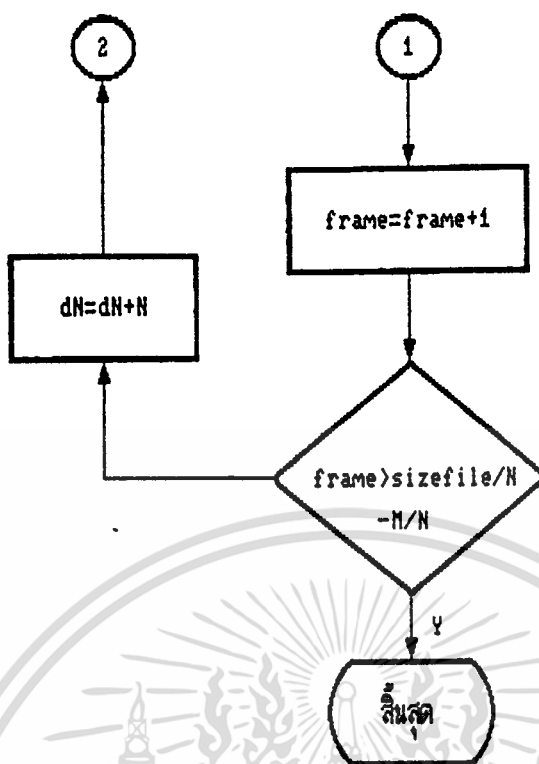


รูปที่ 2.2.9 แผนผังแสดงการคำนวณหาขนาดเฉลี่ยของสัญญาณในแต่ละเฟรม

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การใช้งานเพื่อการค้าหรือการเผยแพร่โดยไม่ได้รับอนุญาตเป็นการฝ่าฝืน  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

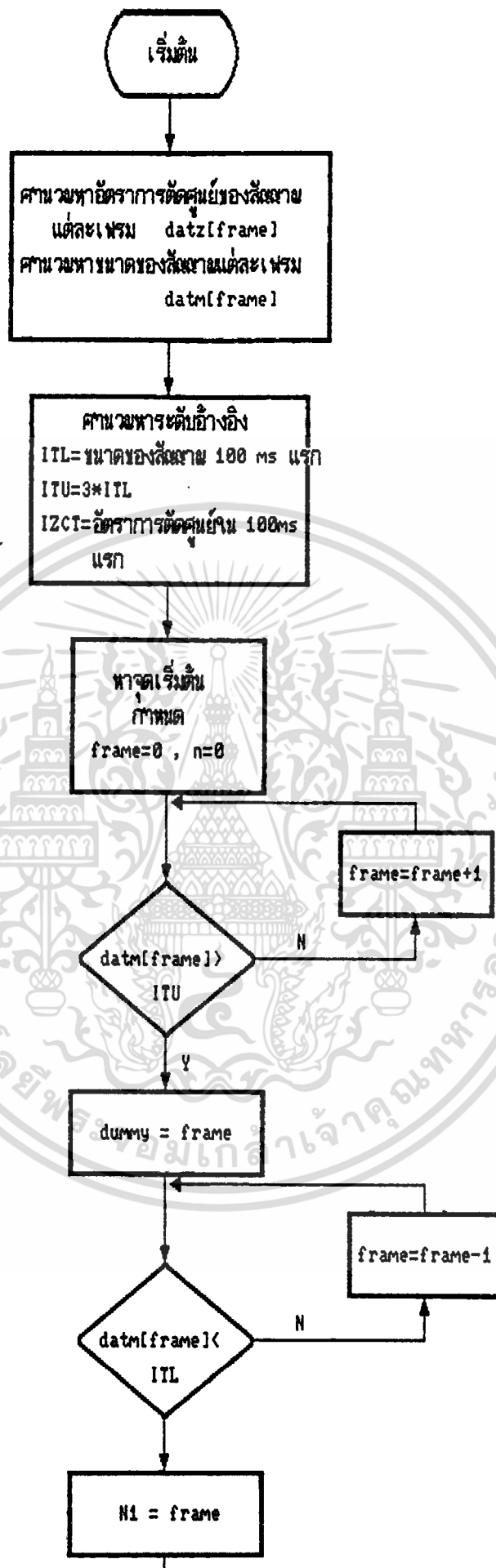


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาก่อน และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2.10 แผนผังแสดงการคำนวณหาอัตราการตัดศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานทางการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ ๓

#### การเข้ารหัสแบบลิเนียร์พรีดิกทีฟ

#### (LINEAR PREDICTIVE CODING, LPC)

การวิเคราะห์แบบลิเนียร์พรีดิกทีฟเป็นการวิเคราะห์เสียงพูดที่คิที่สุดวิธีหนึ่ง โดยการวิเคราะห์วิธีนี้ใช้ในการวิเคราะห์หาค่าพารามิเตอร์พื้นฐานของเสียง เช่น พิตช์(pitch) ฟอร์แมน(formant) และ สเปกตรัม(spectra) เป็นต้น ซึ่งพารามิเตอร์ที่หาได้จากวิธีนี้จะมีความถูกต้องแม่นยำและใช้เวลาในการคำนวณไม่มากนัก

#### 3.1 หลักการเบื้องต้นของแอลพีซี

หลักการเบื้องต้นของการทำนายเชิงเส้นคือ สัญญาณปัจจุบันสามารถประมวลได้จากผลรวมเชิงเส้นของสัญญาณในอดีต โดยให้  $s(n)$  คือสัญญาณเสียงที่สุ่มที่  $n$  ใดๆ  $s(n)$  คือสัญญาณที่  $n$  ที่ได้จากการทำนาย  $a_i$  คือสัมประสิทธิ์การทำนายโดยใช้สัญญาณในอดีตทั้งหมด  $N$  ตัวหรือการทำนายมีออร์เดอร์ (order) เท่ากับ  $M$  สมการการทำนายเชิงเส้นคือ

$$\tilde{s}(n) = \sum_{i=1}^M a_i * s(n-1) \quad \dots (3.1)$$

Z-transform ของสมการ (3.1) คือ

$$\tilde{S}(z) = F(z)S(z) \quad \dots (3.2)$$

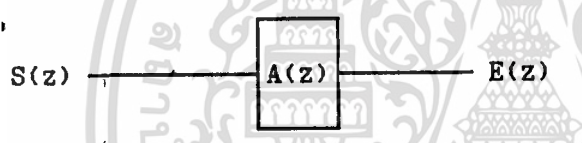
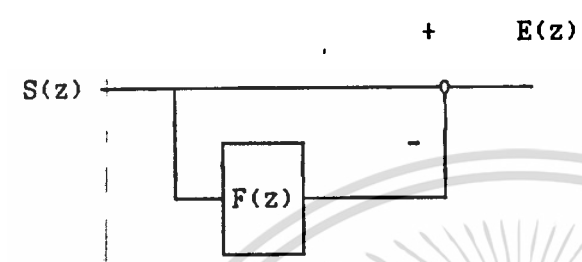
โดยให้

$$F(z) = -\sum_{i=1}^M a_i * z^{-i} \quad \dots (3.3)$$

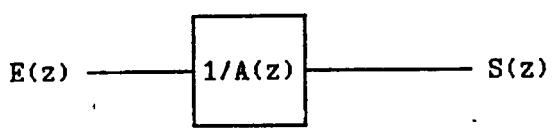
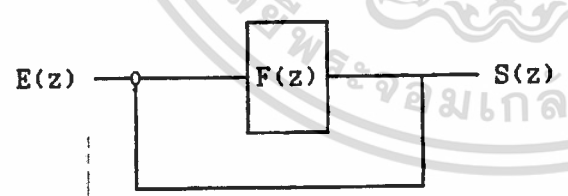
$F(z)$  มีชื่อว่าฟิลเตอร์แห่งการทำนาย(Predictor Filter) โดยถ้าเอาสัญญาณจริงลบด้วยสัญญาณที่ทำนาย จะได้ค่าผิดพลาดของการทำนาย  $e(n)$  เท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างสัญญาณตามสมการ(3.10) มีชื่อเฉพาะว่า All Pole Model เพราะฟิลเตอร์  $(1/A(z))$  มีแต่ Pole ล้วนๆ ทำให้การคำนวณง่ายขึ้น รูป (3.1) แสดงแบบจำลองของวิเคราะห์และสังเคราะห์ตามสมการ (3.8) และ (3.9) ในการวิเคราะห์เราจะแบ่งสัญญาณออกเป็นสั้นๆ เร็วกว่าเฟรมแต่ละเฟรมกินเวลาประมาณ 10-20 มิลลิวินาที ซึ่งช่วงเวลาดังกล่าวเราสมมติให้คุณสมบัติของเสียงไม่เปลี่ยนแปลงตามเวลา  $a_n$  จะเป็นค่าคงที่



(ก) รูปแบบการวิเคราะห์



(ข) รูปแบบการสังเคราะห์

รูป (3.1) แบบจำลองการวิเคราะห์และการสังเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$e(n) = s(n) - \tilde{s}(n) \quad \dots(3.4)$$

$$\begin{aligned} &= s(n) + \sum_{i=0}^M a_i * s(n-i) \\ &= \sum_{i=0}^M a_i * s(n-i) \quad ; a_0=1 \end{aligned} \quad \dots(3.5)$$

Z-transform ของสมการ (3.5) คือ

$$E(z) = S(z)\{1-F(z)\} \quad \dots(3.6)$$

1-F(z) ในสมการที่(3.6) มีชื่อว่าเป็นอินเวอร์สฟิลเตอร์(Inverse Filter)โดยให้สัญลักษณ์ A(z)

$$A(z) = 1-F(z) = 1 + \sum_{i=1}^M a_i * z^{-i} \quad \dots(3.7)$$

ดังนั้น

$$E(z) = S(z)A(z) \quad \dots(3.8)$$

สมการ (3.8) คือส่วนของการวิเคราะห์ โดยเมื่อเรารู้ค่าของ  $a_i$  ก็สามารหาค่าผิดพลาดของการทำนาสได้ ในการสังเคราะห์เราสามารถหาค่า S(z) ได้ ก็รู้ชุดของ  $a_i$  และ E(z) จากความสัมพันธ์

$$\begin{aligned} s(z) &= \frac{E(z)}{1-F(z)} \\ &= \frac{E(z)}{A(z)} \end{aligned} \quad \dots(3.9)$$

หรือ

$$s(n) = e(n) - \sum_{i=1}^M a_i * s(n-1) \quad \dots(3.10)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการหาค่าพารามิเตอร์ เราจะใช้เงื่อนไขค่าผิดพลาดยกกำลังสองในหนึ่งเฟรมที่มีค่าน้อยที่สุด สมมติให้เฟรมที่ทำการวิเคราะห์อยู่ระหว่าง  $n=n_0$  กับ  $n=n_1$  ค่าผิดพลาดยกกำลังสองรวมมีค่าเท่ากับ

$$E = \sum_{n=n_0}^{n_1} [e(n)]^2 \quad \dots (3.11)$$

แทนค่า  $e(n)$  จากสมการที่ (3.4) ได้

$$\begin{aligned} E &= \sum_{n=n_0}^{n_1} [s(n) - \tilde{s}(n)]^2 \\ &= \sum_{n=n_0}^{n_1} [s(n) - \sum_{i=1}^M a_i s(n-i)]^2 \\ &= \sum_{n=n_0}^{n_1} \left[ \sum_{i=1}^M [s(n) + \sum_{i=1}^M a_i s(n-i)] \right] \left[ \sum_{i=1}^M [s(n) + \sum_{i=1}^M a_i s(n-i)] \right] \\ &= \sum_{n=n_0}^{n_1} \left[ \sum_{i=1}^M s^2(n) + \sum_{i=1}^M \sum_{j=1}^M a_i a_j s(n-i) s(n-j) \right] \end{aligned}$$

กำหนดเงื่อนไขค่าผิดพลาดยกกำลังสองรวมในเฟรมหนึ่งที่มีค่าน้อยที่สุด ค่าอนุพันธ์เมื่อเทียบกับ  $a_i$  จะมีค่าเท่ากับศูนย์ ( $\frac{\partial E}{\partial a_i} = 0$ )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{\partial E}{\partial a_1} = \sum_{n=n_0}^{n_1} \left[ \begin{matrix} M \\ 0 + 0 + s(n-j)*s(n) + \sum_{i=1}^M a_i * s(n-j)*s(n-i) \end{matrix} \right]$$

$$0 = \sum_{n=n_0}^{n_1} \left[ \begin{matrix} M & n_1 \\ s(n)*s(n-j) + \sum_{i=1}^M a_i * s(n-i) \end{matrix} \right] \quad \dots(3.12)$$

ถ้ากำหนดให้  $C(i,j) = \sum_{n=n_0}^{n_1} s(n-i)*s(n-j) \quad \dots(3.13)$

แทนลงในสมการ (3.12) จะได้

$$\sum_{i=1}^M a_i C(i,j) = -C(0,j) \quad ; \quad j=1,2,3,\dots,M \quad \dots(3.14)$$

แต่เมื่อ  $j=0$   $\frac{\partial E}{\partial a_1} = 0$  ดังนั้นจะได้

$$E = \sum_{i=0}^M \sum_{n=n_0}^{n_1} a_i * s(n) * s(n-i) \quad \dots(3.15)$$

หรือ  $E = \sum_{i=0}^M a_i C(i,j)$  โดยที่ E เรียกว่า prediction error  $\dots(3.16)$

ค่าของ  $a_i$  ที่ได้จากการแก้สมการ (3.14) สามารถนำไปหาค่าผิดพลาดได้จากสมการที่ (3.15) ซึ่งสามารถนำไปวิเคราะห์เสียงพูดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแก้สมการหาค่า  $a_i$  แบ่งได้เป็น 3 วิธีสำคัญคือ

- 1) วิธีโควาเรียนซ์ (Covariance Method)
- 2) วิธีออโตโครีเลชัน (Autocorrelation)
- 3) วิธีพาร์คอร์ (PARCOR)

### 3.2 การหาค่าตอบด้วยวิธีโควาเรียนซ์

จากสมการ (3.11) ถ้ากำหนดค่าผิดพลาดยกกำลังสองรวมในช่วง  $n_0=M$  ถึง  $n_1=N-1$  จากสมการ (3.13) เราต้องทราบค่าของสัญญาณจริงทั้งหมด  $N$  ตัว คือ  $\{s(n)\} = \{s(0), s(1), \dots, s(N-1)\}$  จึงจะคำนวณค่า  $C(i,j)$  โดย  $i=0,1,2,\dots,M$  และ  $j=1,2,3,\dots,M$  ชุดของสมการได้เป็น

$$\sum_{i=1}^M a_i * C(i,j) = -C(0,j) \quad ; j=1,2,3,\dots,M \quad \dots (3.17)$$

โดยที่

$$C(i,j) = \sum_{n=M}^{N-1} s(n-i)*s(n-j) \quad ; i=0,1,2,\dots,M \quad \dots (3.18)$$

$$j=1,2,3,\dots,M$$

หรือเขียนอยู่ในรูปเมตริกซ์ได้เป็น

$$\begin{bmatrix} C(1,1) & C(1,2) & \dots & C(1,M) \\ C(2,1) & C(2,2) & \dots & C(2,M) \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ C(3,1) & C(3,2) & \dots & C(3,M) \end{bmatrix} = - \begin{bmatrix} C(0,1) \\ C(0,2) \\ \vdots \\ \vdots \\ \vdots \\ C(0,3) \end{bmatrix} \quad \dots (3.19)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งค่า  $C(i,j)$  มีความคล้ายคลึงกับฟังก์ชันโควาเรียนซ์ของสัญญาณ  $s(n)$  จึงเรียกเมตริกซ์ของ  $C(i,j)$  ว่า โควาเรียนซ์เมตริกซ์ (covariance Matrix) จากสมการ (3.16) พบว่า  $C(i,j)=C(j,i)$  ดังนั้นเมตริกซ์นี้จึงมีคุณสมบัติของการสมมาตร โดยค่าในเส้นทแยงมุมมีความสัมพันธ์คือ

$$C(i+1,j+1) = C(i,j) + S(-i-1)*s(-j-1) - s(n-1-i)*s(n-1-j) \quad \dots(3.20)$$

การหาค่าพหามิเตอร์  $a_i$  โดยสมการ (3.19) เรียกว่าวิธีโควาเรียนซ์ (Covariance Method) ซึ่งสามารถแก้สมการนี้ได้โดยมีประสิทธิภาพด้วย Cholesky Decomposition Method [Thomas Parson]

### 3.3 การคำนวณหาค่าตอบโดยวิธีอีตส์สัมพันธ์

จากสมการ (3.13) ถ้ากำหนดให้  $n=-\alpha$  ถึง  $n=\alpha$  และให้  $s(n)=0$  เมื่อ  $n<0$  และ  $n>N$  สมการของ  $C(i,j)$  จะกลายเป็น

$$\begin{aligned} C(i,j) &= \sum_{n=-\alpha}^{\alpha} s(n-i)*s(n-j) \\ &= \sum_{n=-\alpha}^{\alpha} s(n)*s(n+|i-j|) \\ &= \sum_{n=0}^{N-1-|i-j|} s(n)*s(n+|i-j|) \\ &= r(|i-j|) \quad ; \quad \begin{matrix} i=0,1,2,\dots,M \\ j=1,2,3,\dots,M \end{matrix} \quad \dots(3.21) \end{aligned}$$

ซึ่งกรณีนี้  $C(i,j)$  จะตรงกับฟังก์ชัน Short Time Autocorrelation  $r(|i-j|)$  กำหนดเงื่อนไขขอบก่าดังสองรมน้อยที่สุดในช่วง  $n=-\alpha$  ถึง  $n=\alpha$  หรือให้ผลจริงในช่วง  $n=0$  ถึง  $N-M+1$  จะได้ชุดของสมการเชิงเส้นจำนวน  $M$  สมการคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\sum_{i=1}^M a_i * r(|i-j|) = -r(j) \quad ; \quad j=1,2,3...M \quad \dots(3.22)$$

หรือ

$$\sum_{i=0}^M a_i * r(|i-j|) = 0$$

เมื่อเขียนในรูปเมตริกซ์จะได้

$$\begin{bmatrix} r(0) & r(1) & r(2) & \dots & r(M-1) \\ r(1) & r(2) & r(3) & \dots & r(M-2) \\ r(2) & r(3) & r(4) & \dots & r(M-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r(M-1) & r(M-2) & r(M-3) & \dots & r(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_M \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ r(3) \\ \vdots \\ r(M) \end{bmatrix} \quad \dots(3.23)$$

โดยที่

$$r(l) = \sum_{n=0}^{N-1+l} s(n)*s(n+1) \quad ; \quad (l=|i-j|)$$

ในที่นี้  $[r(l)]$  คือเมตริกซ์ของค่าอัตโนมัติสัมพันธ์ซึ่งมีลักษณะเป็น Toeplitz Matrix คือมีคุณสมบัติสมมาตรและองค์ประกอบในแนวเส้นทแยงมุมมีค่าเท่ากัน วิธีหาค่าตอบข้างมีประสิทธิภาพสามารถทำได้จากวิธีของ Levinson และ Durbin

Durbin ได้แก้ไขโดยใช้หวนการรีเคอร์ซีฟ (Recursive solution) ซึ่งเป็นวิธีการที่ใช้ผลลัพธ์ที่หาได้ก่อนมาคำนวณหาผลลัพธ์ถัดไป และทำไปเรื่อยๆจนได้ผลลัพธ์ทั้งหมด ตัวอย่างเช่น การหาผลลัพธ์ของสัมประสิทธิ์การทำนาย(prediction coefficient,  $a_n$ ) เราจะใช้  $a_0$  ในการหา  $a_1$  และ  $a_0, a_1$  ในการหา  $a_2$  ทำอย่างนี้จนได้สัมประสิทธิ์ของตัวทำนายทั้งหมด  $n$  ตัวออกมา

วิธีแก้ปัญหานี้แบบรีเคอร์ซีฟสามารถทำได้โดย การนำสูตรการผิดพลาดของการทำนาย (prediction-error formula) มาใช้ร่วมกับสมการ (3.22) โดยที่สมการผิดพลาดของการทำนายคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$E = \sum_{i=0}^M a_i C(i, j)$$

หรือ

$$E = \sum_{i=0}^M a_{c_i} * r(i)$$

เมื่อนำมา (3.22) และ (3.24) มาเขียนอยู่ในรูปเมทริกซ์จะได้

$$\begin{bmatrix} r(0) & r(1) & r(2) & \dots & r(M) \\ r(0) & r(1) & r(2) & \dots & r(M) \\ r(0) & r(1) & r(2) & \dots & r(M) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r(M-1) & r(M-2) & r(M-3) & \dots & r(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{c_1} \\ a_{c_2} \\ \vdots \\ \vdots \\ a_{c_M} \end{bmatrix} = \begin{bmatrix} E \\ 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix} \quad \dots (3.25)$$

จะเห็นว่าเมทริกซ์ที่ได้เป็น Toeplitz Matrix อยู่เช่นเดิม

สมมติว่าเราสามารถหาค่าสัมประสิทธิ์ของการทำนายที่ลำดับ (order) เป็น 2 (M=2)  $a_1$  และ  $a_2$  ซึ่งสามารถเขียนเมทริกซ์ของการทำนายได้เป็น

$$\begin{bmatrix} r(0) & r(1) & r(2) \\ r(1) & r(2) & r(0) \\ r(2) & r(1) & r(0) \end{bmatrix} = \begin{bmatrix} E \\ 0 \\ 0 \end{bmatrix} \quad \dots (3.26)$$

และเราจะใช้ (3.26) ในการหาค่าสัมประสิทธิ์ของการทำนายที่ลำดับเป็น 3 (M=3) ซึ่งสามารถเขียนเมทริกซ์ได้เป็น

$$\begin{bmatrix} r(0) & r(1) & r(2) & r(3) \\ r(1) & r(2) & r(3) & r(0) \\ r(2) & r(3) & r(0) & r(1) \\ r(3) & r(2) & r(1) & r(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{c_1} \\ a_{c_2} \\ a_{c_3} \end{bmatrix} = \begin{bmatrix} E \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \dots (3.27)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการสำคัญคือการอินเวอร์สเมตริกซ์ จากสมการ (3.26) จะได้

$$\begin{bmatrix} r(0) & r(1) & r(2) \\ r(1) & r(2) & r(3) \\ r(2) & r(3) & r(1) \end{bmatrix} \begin{bmatrix} a_o(2) \\ a_e(1) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ E_1 \end{bmatrix}$$

แล้วนำเมตริกซ์ที่ได้นี้กับเมตริกซ์จากสมการ (3.26) มาทำการรวมกันเพื่อให้ได้สมการ (3.27) คือ

$$\begin{bmatrix} 1 \\ a_o(1) \\ a_o(2) \\ a_o(3) \end{bmatrix} = \begin{bmatrix} 1 \\ a_e(1) \\ a_e(2) \\ 0 \end{bmatrix} + k_3 \begin{bmatrix} 1 \\ a_e(2) \\ a_e(1) \\ 0 \end{bmatrix} \quad \dots (3.28)$$

โดย  $k_3$  ถูกเรียกว่า reflection coefficient

นำ สมการ (3.28) แทนลงในสมการ (3.27) จะได้

$$\begin{bmatrix} r(0) & r(1) & r(2) & r(3) \\ \hline r(1) & r(0) & r(1) & r(2) \\ r(2) & r(1) & r(0) & r(1) \\ r(3) & r(2) & r(1) & r(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_e(1) \\ a_e(2) \\ 0 \end{bmatrix} + k_3 \begin{bmatrix} 0 \\ a_e(2) \\ a_e(1) \\ 1 \end{bmatrix} = \begin{bmatrix} E_2 \\ 0 \\ 0 \\ q \end{bmatrix} + k_3 \begin{bmatrix} q \\ 0 \\ 0 \\ E_e \end{bmatrix} \quad \dots (3.29)$$

เมื่อเปรียบเทียบเมตริกซ์ทางด้านขวากับทางด้านซ้ายของสมการ (3.27) กับสมการ (3.29) จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$1) \quad q + E_e k_3 = 0$$

$$k_3 = -q/E_e$$

q สามารถหาได้จากสมการ (3.29) คือ

$$q = \sum_{i=0}^2 a(i)r_{(3-i)}$$

ดังนั้น

$$k_3 = -\frac{1}{E_e} \sum_{i=0}^2 a(i)r_{(3-i)}$$

$$2) \quad E_3 = E_e + k_3 q$$

$$= E_e (1 - k_3^2) \dots (3.30)$$

โดย  $E_3$  คือความผิดพลาดของการทำนายตัวใหม่

ค่า  $k$  ในสมการ(3.30) จะต้องมีค่าน้อยกว่าหรือเท่ากับ 1 เสมอ เมื่อพิจารณาสมการแล้วพบว่าค่าความผิดพลาดของการทำนายตัวใหม่ที่ทำได้ จะมีน้อยกว่าค่าความผิดพลาดของการทำนายตัวก่อนหน้านั้นเสมอ แต่จะมีค่ามากกว่า 0 ซึ่งจะเห็นได้ว่าค่าความถูกต้องจะมีค่าเพิ่มขึ้นตามออร์เดอร์ที่ใช้ในการทำนาย

จากสมการที่กล่าวถึงข้างต้น เราสามารถเขียนสูตรทั่วไปของ recursive ที่  $n$ , ได้ดังนี้

$$\begin{bmatrix} a_n(0) \\ a_n(1) \\ \dots \\ a_n(n-1) \\ a_n(n) \end{bmatrix} = \begin{bmatrix} a_{n-1}(0) \\ a_{n-1}(1) \\ \dots \\ a_{n-1}(2) \\ a_{n-1}(3) \end{bmatrix} + k_n \begin{bmatrix} 0 \\ a_{n-1}(n-1) \\ \dots \\ a_{n-1}(1) \\ a_{n-1}(0) \end{bmatrix} \dots (3.31a)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือเขียนใหม่ได้เป็น

$$a_n(i) = a_{n-1} + k_n a_{n-1}(n-i) \quad \dots (3.21b)$$

โดยที่  $a_n(0) = 1$

$$k_n = \frac{-1}{E_n} \sum_{i=0}^{n-1} a_{n-1}(i)r(n-i) \quad \dots (3.31c)$$

$$E_n = E_{n-1}(1-k_n^2) \quad \dots (3.31d)$$

การคำนวณแบบรีเคอร์ซีฟสามารถเขียนเป็นอัลกอริทึมได้ดังนี้

1) เริ่มจาก  $n=0$  จะได้  $\begin{bmatrix} r(0) \\ 1 \end{bmatrix} = \begin{bmatrix} E_0 \end{bmatrix}$

ดังนั้น  $E_0 = r(0)$

2) เมื่อ  $n > 0$  จะทำการหาค่า

$$2.1 \quad k_n = \frac{-1}{E_{n-1}} \sum_{i=0}^{n-1} a_{n-1}(i)r(n-i)$$

$$2.2 \quad \text{โดยที่ } 0 < i < n \quad \text{ถ้า } i=0 \quad a_n(i) = 1 \quad \text{ถ้า } i=n \quad a_n(n) = k_n$$

$$0 < i < n \quad a_n(i) = a_{n-1}(i) + k_n a_{n-1}(n-i)$$

$$2.3 \quad E_n = E_{n-1}(1-k_n^2)$$

3) กลับไปทำขั้นตอนที่ 2 จน  $n=๓$  (๓ คือจำนวนออร์เดอร์ของการทำนาย)

ข้อดีของวิธีอัลกอริทึมนี้คือการคำนวณเพื่อแก้สมการที่มีค่าน้อยกว่าวิธีโควาเรียนซ์และมีความแน่นอนด้านเสถียรภาพ แต่สัญญาณจริงที่ใช้คำนวณ  $r(l)$  ต้องผ่าน smoothing window เพื่อลดความเอนทางด้านความถี่ นอกจากนั้นการใช้ rectangular window ยังทำให้ค่าความผิดพลาด  $e(n)$  มีค่ามากในช่วงท้ายของแต่ละเฟรม สัญญาณ  $s(n)$  ที่ใช้คำนวณ  $r(l)$  จึงมีค่าเท่ากับ  $s(n) = s_1(n)*w(n)$  ซึ่งค่า  $s_1(n)$  คือสัญญาณของเสียง  $w(n)$  คือฟังก์ชันของวินโดว์

โดยทั่วไปใช้ Hamming window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากทฤษฎีที่ได้กล่าวไว้ในตอนต้น สามารถนำมาเขียนเป็นโปรแกรมย่อย LPCA เพื่อใช้ในการวิเคราะห์เสียงพูด หาพารามิเตอร์ของฟิลเตอร์ในแต่ละเฟรม โดยสามารถกำหนดตัวแปรที่ใช้ในโปรแกรมย่อย LPCA ได้ดังนี้ คือ

| ตัวแปรในโปรแกรมย่อย | ความหมาย   |
|---------------------|--|
| N                   | จำนวนสัญญาณสุ่มที่ใช้ในการวิเคราะห์ในหนึ่งเฟรม                 |
| x(n)                | สัญญาณสุ่มที่ใช้ในการวิเคราะห์ในหนึ่งเฟรม                      |
| iorder              | จำนวนออร์เดอร์   |
| r[i]                | ค่าสัมประสิทธิ์อัตโนมัติสัมพันธ์ (autocorrelation coefficient) |
| rφ                  | กำลังงานของสัญญาณในหนึ่งเฟรม                                   |
| rc[k]               | ค่าสัมประสิทธิ์การสะท้อนกลับ (reflection coefficients)         |
| a[k]                | ค่าสัมประสิทธิ์การทำนาย (predictor coefficient)                |
| pe                  | ค่าความผิดพลาดของการทำนาย (prediction error)                   |

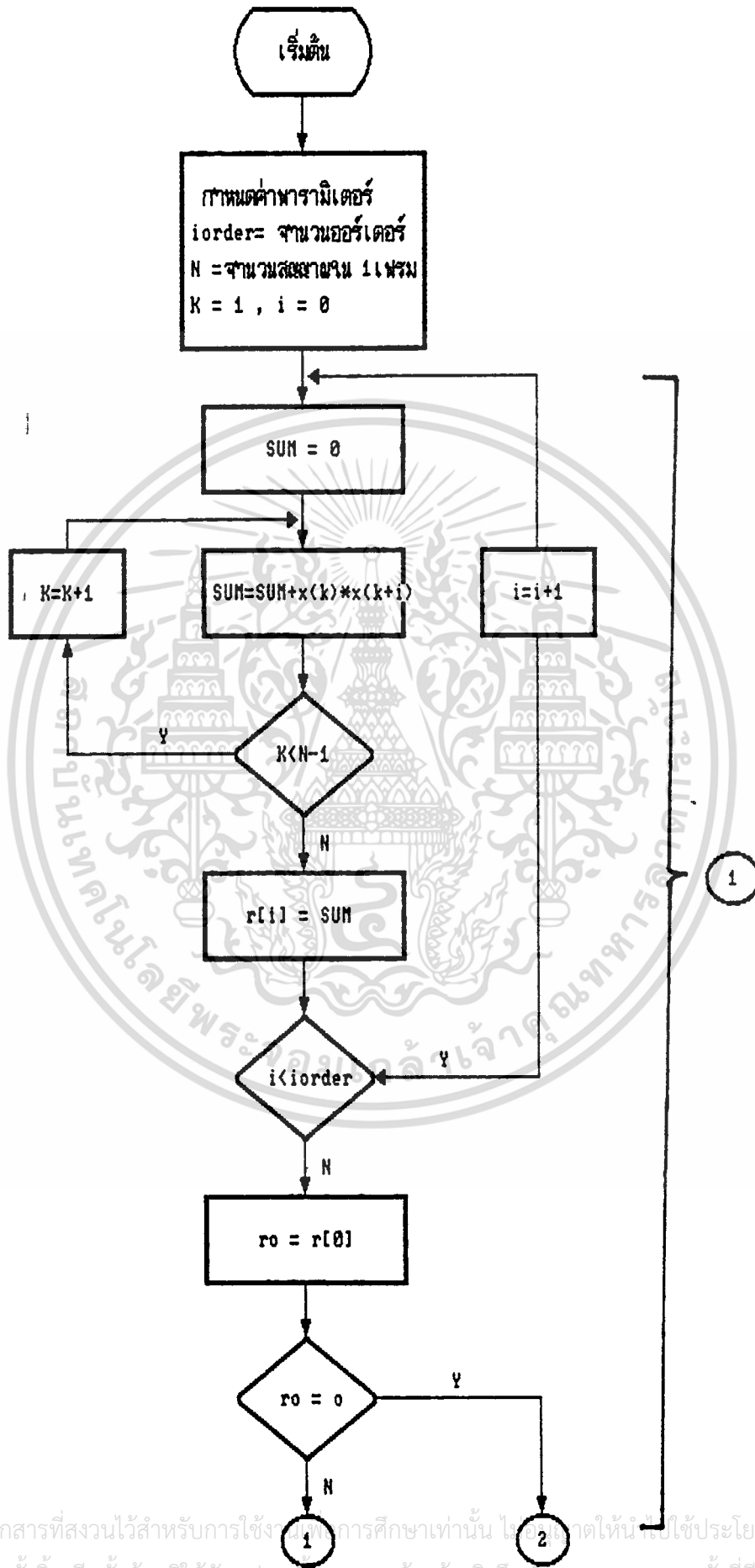
ตารางที่ 3.2 แสดงตัวแปรที่ใช้ในโปรแกรมย่อยและความหมาย

ซึ่งโปรแกรมย่อย LPCA สามารถแบ่งได้เป็น 2 ส่วน ดังนี้

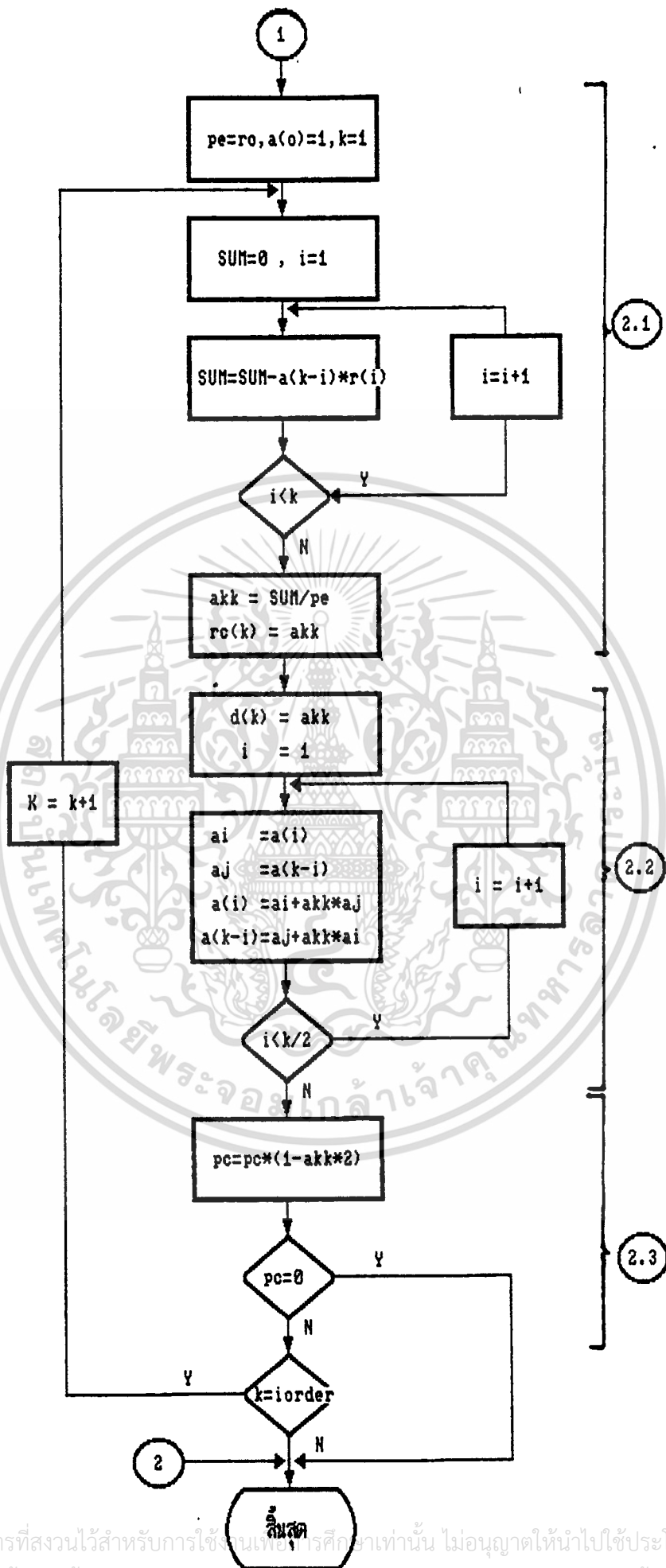
1. คำนวณสัมประสิทธิ์อัตโนมัติสัมพันธ์  $r[i]$  หลังจากหาค่าสัมประสิทธิ์อัตโนมัติสัมพันธ์ค่าแรกได้ ( $r[0]$ ) จะทำการตรวจสอบค่า  $r[0]$  ถ้ามีค่าเท่ากับศูนย์จะออกจากโปรแกรมทันที
2. คำนวณหาค่าสัมประสิทธิ์การทำนายเชิงเส้น (ทำการคำนวณแบบรีเคอร์ซีฟ)
  - 2.1 คำนวณสัมประสิทธิ์การสะท้อนกลับ (reflection coefficient:  $rc[k]$ )
  - 2.2 คำนวณสัมประสิทธิ์การทำนาย (predictor coefficient:  $a[k]$ )
  - 2.3 คำนวณความผิดพลาดของการทำนาย (prediction error:  $pe$ ) หลังจากคำนวณได้ค่า  $pe$  เป็นที่เรียบร้อยแล้ว ก็ทำการตรวจสอบ ถ้าค่า  $pe$  มีค่าเป็นลบจะออกจากโปรแกรมทันที

สามารถการฝังการทำงานได้ดังรูป 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



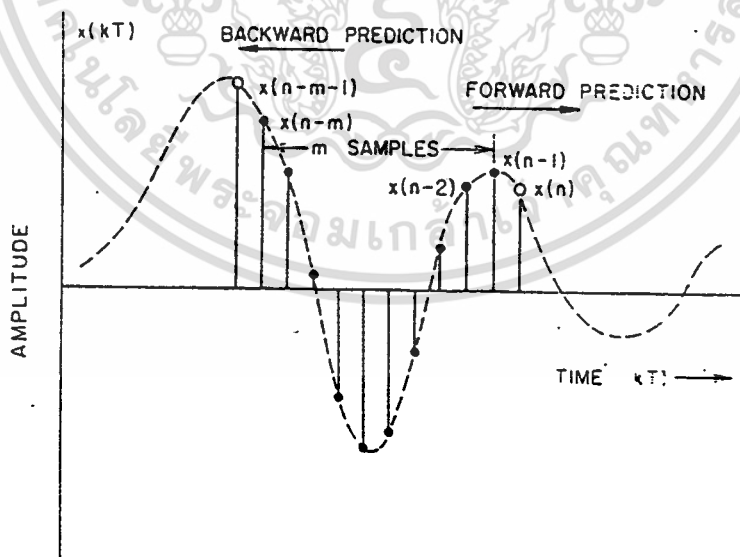
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.2 แผนผังแสดงโปรแกรมย่อย LPCA

### 3.4 วิธีพาร์คอร์ (PACOR)

หลักการเบื้องต้นของวิธีพาร์คอร์ (PACOR) ซึ่งย่อมาจาก Partial Correlation (อัตราสัมพันธ์บางส่วน) คือการทำนายด้วยตัวทำนายเชิงเส้นที่มีอันดับเท่ากับ  $m$  ทั้งสองทาง คือทำนายไปข้างหน้าและทำนายย้อนกลับ จากรูป 3.3 แสดงชุดสัญญาณสุ่ม  $x(kT)$  จำนวน  $m+2$  ค่า การทำนายไปข้างหน้าหมายถึง การทำนายค่า  $x(n)$  จากค่า  $x(n-1), x(n-2), \dots, x(n-m)$  ค่าความผิดพลาดของการทำนายที่  $n$  แทนด้วย  $x_{m+}(n)$  เท่ากับ

$$\begin{aligned}
 x_{m+}(n) &= x(n) - \left\{ -\sum_{i=1}^m a_{m,i} * x(n-i) \right\} \\
 &= \sum_{i=0}^m a_{m,i} * x(n-i) \quad ; a_{m,0} = 1 \quad \dots (3.32)
 \end{aligned}$$



รูป 3.3 แสดงการทำนายไปข้างหน้าและย้อนกลับจำนวน  $m$  ลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำนายค่าย้อนหลังคือ การทำนายค่า  $x(n-m-1)$  จากค่า  $x(n-m), x(n-m+1), \dots, x(n-1)$  ค่าความผิดพลาดของการทำนายที่ออร์เดอร์  $m$  แทนด้วย  $x_{m-}(n)$  เท่ากับ

$$\begin{aligned} x_{m-}(n) &= x(n-m-1) - \{-\sum_{i=1}^m b_{m,i} * x(n-i)\} \\ &= \sum_{i=1}^{m+1} b_{m,i} * x(n-i) \quad ; \quad b_{m,m+1} = 1 \end{aligned} \quad \dots (3.33)$$

จากนั้นใช้เงื่อนไขค่าความผิดพลาดยกกำลังสองผิดพลาดน้อยที่สุดกับการทำนายทั้งสองทิศทางพร้อมกัน ให้  $\alpha_m$  และ  $\beta_m$  คือค่าความผิดพลาดยกกำลังสองรวมของการทำนายไปข้างหน้าและการทำนายไปข้างหลังตามลำดับ

$$\alpha_m = \sum_{n=n_0}^{n_1} [x_{m+}(n)]^2 \quad \dots (3.34)$$

$$\beta_m = \sum_{n=n_0}^{n_1} [x_{m-}(n)]^2 \quad \dots (3.35)$$

โดย  $m$  เป็นตัวแปรที่มีค่าเท่ากับ  $1, 2, 3, \dots, M$  เมื่อ  $M$  เป็นจำนวนออร์ทั้งหมดของตัวทำนายที่ใช้เงื่อนไขค่าความผิดพลาดยกกำลังสองรวมน้อยที่สุดดังนั้น

$$\frac{\partial \alpha_m}{\partial a_{m,i}} = 0 \quad , \quad \frac{\partial \beta_m}{\partial b_{m,i}} = 0 \quad i, m = 1, 2, 3, \dots, M$$

จากเงื่อนไขนี้ยังไม่สามารถนำไปตั้งสมการหาค่าสัมประสิทธิ์หึงตัวแปรได้โดยตรง จำเป็นต้องมีเงื่อนไขเพิ่มเติม กรรมวิธีในการหาเงื่อนไขเพิ่มเติมซึ่งพัฒนาโดย F. Itakura จะมีคำตอบอยู่ในรูปสัมประสิทธิ์ของพาร์คอร์ (PARCOR coefficient) แทนด้วยสัญลักษณ์  $k_m$  มีรายละเอียดอยู่ใน [3] ส่วนวิธีของ J.D. Markel และ A.H. Gray ก็เป็นการหาค่าตอบในรูปของ  $k_m$  เช่นเดียวกัน แต่ด้วยวิธีที่แตกต่างกัน Markel และ Gray อาศัยหลักการของอินเนอร์โพรดักส์และคุณสมบัติของโถโกนัลของตัวทำนายนำไปใช้ในการหาค่า  $k_m$  มีรายละเอียดอยู่ใน [6]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ ๘ การรับรู้เสียงพูด

การรับรู้เสียงพูด เป็นลักษณะหนึ่งของการรับรู้รูปแบบ (Pattern Recognition) คือจะเป็นการเปรียบเทียบระหว่าง แบบทดสอบ (Test Pattern) กับ แบบอ้างอิง (Reference Pattern) ซึ่งเป็นรูปแบบที่เราทราบและเก็บไว้ล่วงหน้า

ขั้นตอนในการรับรู้แบบแบ่งเป็น 2 ขั้นตอน คือ (Parson T.W., 1986)

ก. ขั้นเรียนรู้ (Learning) จะเป็นการสร้างกลุ่มของแบบอ้างอิง ในการรับรู้เสียงพูดในขั้นนี้จะทำการวิเคราะห์เสียงพูดก่อน แล้วเก็บลักษณะของเสียงในรูปของพารามิเตอร์ พร้อมกับป้ายกำกับ เพื่อให้เปรียบเทียบในขั้นต่อไป รายละเอียดของการวิเคราะห์เสียงพูดได้กล่าวไว้ในบทที่ 2

ข. ขั้นรับรู้ (Recognition) จะเป็นการทดสอบการรับรู้ ระหว่างแบบอ้างอิง กับแบบทดสอบ โดยจะทำการเปรียบเทียบพารามิเตอร์ของแบบทดสอบ กับแบบอ้างอิงทั้งหมด แบบอ้างอิงที่เลือก คือ แบบอ้างอิงที่มีพารามิเตอร์ใกล้เคียงกับแบบทดสอบที่สุด เทคนิคในการเปรียบเทียบมีอยู่หลายวิธี ดังจะได้อีกต่อไป

### 4.1 การจัดกลุ่มเพื่อสร้างแบบอ้างอิง

การสร้างกลุ่มของแบบอ้างอิงเป็นขั้นตอนในส่วนของขั้นเรียนรู้ โดยเริ่มต้นจากการนำข้อมูลสัญญาณเสียงที่ได้จากส่วนของวิเคราะห์ มาทำการสร้างเป็นกลุ่มของแบบอ้างอิง ในระบบการรับรู้เสียงพูดแบบต่างบุคคล จะใช้แบบอ้างอิงของคำหนึ่ง ๆ จากผู้พูดจำนวนมาก เพื่อที่จะได้ครอบคลุมถึงความแปรปรวนต่าง ๆ ที่จะเกิดขึ้นระหว่างผู้พูดแต่ละคน แต่อย่างไรก็ตามผลที่เกิดขึ้นตามมา เมื่อจำเป็นจะต้องมีข้อมูลจำนวนมากในกลุ่มของแบบอ้างอิง คือ

ก. เวลาที่ใช้ในการตอบสนอง ในการรับรู้เสียงพูด ขั้นตอนการเปรียบเทียบแบบทดสอบกับแบบอ้างอิงทั้งหมดที่มีเป็นขั้นตอนที่สำคัญในการรับรู้ เมื่อแบบอ้างอิงมีจำนวนมาก การเปรียบเทียบจึงจำเป็นที่จะต้องอาศัยเวลาเพิ่มขึ้นด้วย

ข. เนื้อหาในหน่วยความจำสำรองที่ใช้ในการเก็บแบบอ้างอิง เมื่อมีแบบอ้างอิงจำนวนมากที่จำเป็นต้องใช้ในการเปรียบเทียบ เนื้อหาในหน่วยความจำสำรองก็จำเป็นต้องมีเพิ่มขึ้นด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค. ความถูกต้องในการรับรู้ เนื่องจากจำนวนแบบอ้างอิงในแต่ละคำไม่ได้เป็นสัดส่วนโดยตรงกับความถูกต้องในการรับรู้ กล่าวคือเมื่อเราเพิ่มจำนวนแบบอ้างอิงในแต่ละคำไปจนถึงระดับหนึ่ง ความถูกต้องในการรับรู้จะเริ่มคงที่ และในบางครั้งอาจจะมีค่าลดลงด้วย (Rebiner L.R., 1979)

จากเหตุผลดังกล่าว จึงทำให้มีความพยายามที่จะจัดกลุ่มของแบบอ้างอิงในแต่ละคำใหม่เพื่อให้ได้แบบอ้างอิงจำนวนที่พอเหมาะ และสามารถใช้เป็นตัวแทนของแบบอ้างอิงที่มีอยู่ทั้งหมดได้ อัลกอริทึมที่ใช้ในการแบ่งกลุ่มมีอยู่จำนวนมาก สำหรับอัลกอริทึมที่จะกล่าวถึง คือการหาค่า K (K-mean algorithm) ซึ่งประกอบไปด้วยขั้นตอนต่างๆ ดังนี้

#### 4.1.1 การหาจุดศูนย์กลางของกลุ่ม (Center Cluster)

ในขั้นตอนนี้จะทำการกำหนดจำนวนกลุ่มของแบบอ้างอิงให้มีค่าเท่ากับ K และเริ่มต้นจากแบบอ้างอิงที่มีความยาวมากที่สุดเป็นจุดศูนย์กลางของกลุ่มแรก จากนั้นจะหาจุดศูนย์กลางของคำถัดไปได้จากแบบอ้างอิงที่มีระยะทางห่างจากจุดศูนย์กลางของกลุ่มแรกมากที่สุด และจุดศูนย์กลางของคำถัดไปจะหาได้จาก แบบอ้างอิงที่มีระยะทางห่างจากจุดศูนย์กลางของกลุ่มทั้งสองมากที่สุด ขั้นตอนนี้จะถูกทำซ้ำจนได้กลุ่มของแบบอ้างอิงครบตามจำนวน K กลุ่ม หรือค่าระยะทางที่มากที่สุดมีค่าน้อยกว่าระดับที่กำหนดไว้ซึ่งเรียกว่า Farthest Neighbor

#### 4.1.2 การจัดกลุ่มใหม่ (Reclassification)

แบบอ้างอิงที่เหลืออยู่ในแต่ละคำ จะถูกจัดให้เข้าในแต่ละกลุ่ม โดยมีเงื่อนไขคือ ระยะทางของแบบอ้างอิงนี้กับแบบอ้างอิงในกลุ่มจะมีค่าน้อยที่สุด หลังจากที่แบบอ้างอิงทุกแบบถูกจัดเข้ากลุ่มเรียบร้อยแล้ว จะทำการหาจุดศูนย์กลางของแต่ละกลุ่มใหม่ ขั้นตอนนี้จะทำซ้ำจนกว่าจุดศูนย์กลางของแต่ละกลุ่มจะไม่มี การเปลี่ยนแปลง อย่างไรก็ตามจากขั้นตอนดังกล่าวจะเห็นว่า การกำหนดค่าจำนวนกลุ่มที่เหมาะสมจะทำได้ลำบาก ดังนั้นจึงได้มีการปรับปรุงเทคนิคนี้เป็น การกำหนดค่าระยะทางสูงสุดที่จะทำให้เกิดกลุ่มใหม่ขึ้น คือเมื่อแบบอ้างอิงใดที่ถูกเลือกให้เข้ากลุ่มด้วยระยะทางที่น้อยที่สุดแล้ว แต่ระยะทางนี้ยังคงมีค่ามากกว่าระดับที่กำหนด แบบอ้างอิงนี้จะถูกนำไปสร้างเป็นจุดศูนย์กลางของกลุ่มใหม่ทันที

## 4.2 การเปรียบเทียบแบบไดนามิกโปรแกรมมิ่ง

เป็นที่ทราบกันดีอยู่แล้วว่า การเปลี่ยนแปลงอัตราเร็วในการเปล่งเสียง เป็นสาเหตุของการแกว่งไปมา ของรูปสัญญาณเสียงบนแกนเวลา การกำจัดผลของการแกว่งนี้เป็นปัญหาสำคัญประการหนึ่งในการรับรู้เสียงพูด ในยุคเริ่มแรกได้มีการเสนอวิธีปรับรูปร่างเชิงเส้น (Linear Normalization) เพื่อที่จะกำจัดความแตกต่างของเวลา ระหว่างรูปแบบของสัญญาณเสียง 2 รูปแบบ โดยทำการเปลี่ยนรูปร่างเชิงเส้น (Linear Transformation) บนแกนเวลา แต่อย่างไรก็ตาม วิธีนี้ยังไม่ประสบความสำเร็จในการปรับปรุงความถูกต้องของการรับรู้เสียงพูด

เทคนิคของไดนามิกโปรแกรมมิ่งเป็นแนวความคิดใหม่ ที่จะใช้ในการกำจัดความแตกต่างของเวลาระหว่างรูปแบบสัญญาณ 2 รูปแบบ โดยการสร้างแบบจำลองของฟังก์ชันที่ไม่เป็นเชิงเส้น โดยที่ชื่อว่า ฟังก์ชันแรปปิง (Wrapping Function) ความแตกต่างของเวลาจะถูกกำจัดออกไปได้โดยอาศัยแบบจำลองฟังก์ชันนี้

### 4.2.1 ฟังก์ชันแรปปิง (Rabiner L.R., 1978)

เราสามารถแสดงสัญญาณเสียงออกมาให้อยู่ในรูปของเวกเตอร์ได้ คือ

$$\begin{aligned} A &= a_1, a_2, \dots, a_i, \dots, a_I \\ B &= b_1, b_2, \dots, b_j, \dots, b_J \end{aligned} \quad \dots (4.2.1)$$

ในการที่จะแก้ปัญหาความแตกต่างของเวลา ระหว่างรูปแบบของสัญญาณเสียง 2 รูปแบบ นั้น สามารถทำได้โดย แทนที่จะอธิบายสัญญาณเสียงด้วยธรรมชาติของการแกว่งบนแกนเวลาหรือความแตกต่างของเวลา เราจะอธิบายด้วยระนาบ  $i-j$  ดังแสดงในรูปที่ 4.2.1 โดยรูปแบบ A และ B จะถูกแสดงบนแกน  $i$  และ  $j$  ตามลำดับ จากรูปจะเห็นว่า ความแตกต่างของเวลาสามารถแสดงได้ด้วยลำดับของจุด  $c = (i, j)$  จะได้ว่า

$$F = c(1), c(2), \dots, c(k), \dots, c(K) \quad \dots (4.2.2)$$

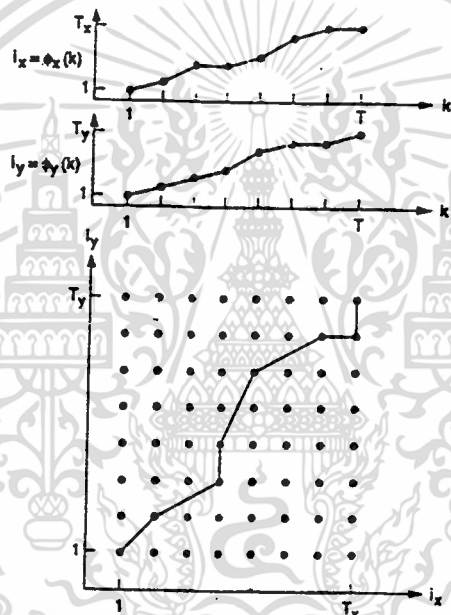
โดย  $c(k) = (i(k), j(k))$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากลำดับนี้จะสามารถมองเป็นฟังก์ชันซึ่งจะโยงแกนเวลาของรูปแบบสัญญาณเสียง A ไปยัง B เราเรียกฟังก์ชันนี้ว่า ฟังก์ชันแรปปิง (Wrapping Function) ดังแสดงในรูป 4.2.1 ถ้าไม่มีความแตกต่างของเวลาระหว่างสัญญาณทั้งสองแล้ว จะได้ว่า ฟังก์ชันแรปปิงจะมีลักษณะเป็นเส้นทแยงมุม  $i=j$  ฟังก์ชันนี้จะมีการเปลี่ยนไปตามความแตกต่างของเวลาที่เกิดขึ้น

ให้ความแตกต่างระหว่างเวกเตอร์  $a_i$  และ  $b_j$  เป็น

$$d(c) = d(i,j) = \left\| a_i - b_j \right\| \dots\dots (4.2.3)$$



รูปที่ 4.2.1 แสดงให้เห็นลักษณะของฟังก์ชันแรปปิง

ดังนั้นจะได้ว่า ผลต่างรวมของฟังก์ชันแรปปิง F คือ

$$E(F) = \sum_{k=1}^K d(c(k)) \cdot w(k) \dots\dots (4.2.4)$$

โดยที่  $w(k)$  เป็นสัมประสิทธิ์น้ำหนัก (Weighting Coefficient) ซึ่งจะทำให้การวัด  $E(F)$  มีความยืดหยุ่นขึ้น ในการหาทางเดินของฟังก์ชันแรปปิงที่ดีที่สุดนั้น จะได้จากค่าผลต่างรวมที่น้อยที่สุดตามทางเดินนั้น และเพื่อที่จะให้ได้ค่าคงตัวบนแกนเวลา เราสามารถเขียนระยะทางระหว่างรูปแบบสัญญาณเสียง A และ B ได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$D(A,B) = \min_F \frac{\sum_{k=1}^K d(c(k)) \cdot w(k)}{\sum_{k=1}^K w(k)} \quad \dots (4.2.5)$$

โดยที่ตัวหาร  $\sum w(k)$  จะเป็นตัวชดเชยผลของค่า  $K$  คือ จำนวนจุดของฟังก์ชันเรปปีง จากสมการที่ (4.2.5) จะได้รูปทั่วไปของระยะทางการปรับตัวตามแนวแกนเวลา (Time-normalized Distance) เราเรียกวิธีการที่จะแก้สมการนี้ได้อย่างมีประสิทธิภาพว่า **ไดนามิกโปรแกรมมิ่ง (Dynamic Programming)** อย่างไรก็ตามการหาฟังก์ชันเรปปีงที่เหมาะสมนั้น ซึ่งขึ้นอยู่กับข้อกำหนดของฟังก์ชันเรปปีง และการระบุถึงประสิทธิภาพนั้นก็อีกด้วย

ในการหาค่าฟังก์ชันเรปปีง ได้มีการกำหนดเงื่อนไขดังนี้ คือ

ก. เงื่อนไขโมโนโทนิก (Monotonic Condition)

$$i(k-1) < i(k) \text{ และ } j(k-1) < j(k)$$

ข. เงื่อนไขความต่อเนื่อง (Continuity Condition)

$$i(k) - i(k-1) < 1 \text{ และ } j(k) - j(k-1) < 1$$

จากเงื่อนไขทั้งสองนี้ ทำให้ได้ความสัมพันธ์ระหว่าง 2 จุดคือ

$$c(k-1) = \begin{bmatrix} (i(k), j(k-1)) \\ (i(k-1), j(k-1)) \\ \text{หรือ } (i(k-1), j(k)) \end{bmatrix} \quad \dots (4.2.6)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค. เงื่อนไขขอบเขต (Boundary Condition)

$$i(1) = 1, j(1) = 1 \text{ และ}$$

$$i(K) = I, j(K) = J \quad \dots(4.2.7)$$

ง. เงื่อนไขการปรับขอบเขต (Boundary Adjustment)

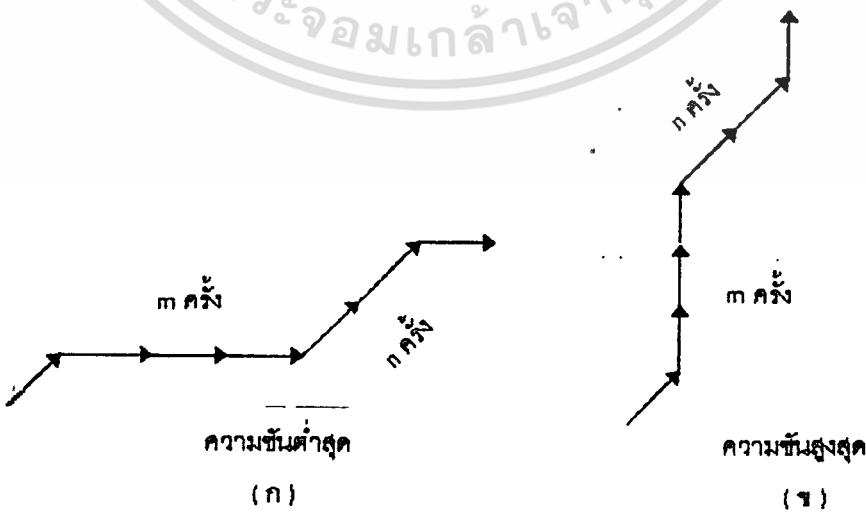
$$|i(k) - j(k)| < r \quad \dots(4.2.8)$$

โดยที่  $r$  เป็นความกว้างของหน้าต่าง ซึ่งจะมีค่าเป็นเลขจำนวนเต็มบวก

จ. เงื่อนไขการเปลี่ยนความชัน (Slope Constraint Condition)

เนื่องจากความแปรปรวนของความยาวในค่าพูด สามารถเกิดขึ้นได้มาก เช่นเมื่อเราเปรียบเทียบคำพูด  $g$  ซึ่งสั้นมากเมื่อเทียบกับคำพูด  $x$  ในกรณีนี้ทำให้ได้ฟังก์ชันแรมป์บั้งที่ผิดเพี้ยนไปมาก ดังนั้นจึงมีการกำหนดให้มีเงื่อนไขการเปลี่ยนความชันขึ้น โดยเงื่อนไขนี้จะกำหนดให้จุด  $c(k)$  สามารถเคลื่อนที่ไปตามแกน  $i$  หรือ  $j$  โดยไม่เปลี่ยนทิศทางได้  $m$  ครั้ง จากนั้นจุด  $c(k)$  จะไม่สามารถเคลื่อนที่ไปในทิศทางเดิมได้อีกอย่างน้อย  $n$  ครั้ง ดังแสดงในรูปที่ (4.2.2ก) และ(4.2.2ข) เงื่อนไขนี้จะถูกเขียนออกมาในรูปของ

$$P = n/m \quad \dots(4.2.9)$$



รูปที่ 4.2.2 เงื่อนไขการเปลี่ยนความชันของฟังก์ชันแรมป์บั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ (4.2.5) ถ้าเราให้ตัวหาร ซึ่งเรียกว่าสัมประสิทธิ์การปรับตัว (Normalize Coefficient) ไม่ขึ้นอยู่กับฟังก์ชันแรบปีง จะทำให้เขียนสมการนี้ใหม่ได้เป็น

$$D(A,B) = (1/N) \text{Min} \left[ \sum_{k=1}^K d(c(k)) \cdot w(k) \right] \quad \dots(4.2.10)$$

โดย

$$N = \sum_{k=1}^K w(k) \quad \dots(4.2.11)$$

เพื่อที่จะทำให้สามารถแก้สมการที่ (4.2.10) ได้ง่ายขึ้น เราจึงมีการกำหนดค่าของ  $w(k)$  ซึ่งเราเรียกว่า สัมประสิทธิ์น้ำหนักแบ่งออกได้เป็น 2 แบบ คือ

ก. แบบแกนเวลาสมมาตร (Symmetric) ในแบบนี้แกนเวลาทั้งสองจะถูกเปลี่ยนรูปให้อยู่แกนชั่วคราว ซึ่งกำหนดเป็นแกนร่วม ดังนั้น ผลรวมของสัมประสิทธิ์น้ำหนักก็คือ ผลรวมของแกนร่วมนั่นเอง ( $1=i+j$ )

$$w(k) = (i(k)-i(k-1)) + (j(k)-j(k-1)) \quad \dots(4.2.12)$$

จะได้ว่า  $N = I + J \quad \dots(4.2.13)$

โดย  $I$  และ  $J$  เป็นความยาวของสัญญาณ  $A$  และ  $B$  ตามลำดับ

ข. แบบแกนเวลาไม่สมมาตร (Asymmetric) ในแบบนี้แกนเวลาจะถูกเปลี่ยนรูปจากแกนเวลาหนึ่ง ไปอยู่บนอีกแกนเวลาหนึ่ง ดังนั้นผลรวมของสัมประสิทธิ์น้ำหนัก ก็คือผลรวมของแกน  $i$  หรือ  $j$  เท่านั้น

$$w(k) = i(k) - i(k-1) \quad \dots(4.2.14)$$

จะได้ว่า  $N = I \quad \dots(4.2.15)$

หรือ  $w(k) = j(k) - j(k-1) \quad \text{จะได้ว่า} \quad N = J$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$D(A,B) = \underset{F}{\text{Min}} \left[ \begin{array}{c} K \\ \sum_{k=1} d(c(k)).w(k) \\ \hline K \\ \sum_{k=1} w(k) \end{array} \right] \dots\dots(4.2.5)$$

โดยที่ตัวหาร  $\sum w(k)$  จะเป็นตัวชดเชยผลของค่า  $K$  คือ จำนวนจุดของฟังก์ชันแรบปีง จากสมการที่ (4.2.5) จะได้รูปทั่วไปของระยะทางการปรับตัวตามแนวแกนเวลา (Time-normalized Distance) เราเรียกวิธีการที่จะแก้สมการนี้ได้อย่างมีประสิทธิภาพว่า **ไดนามิกโปรแกรมมิ่ง (Dynamic Programming)** อย่างไรก็ตามการหาฟังก์ชันแรบปีงที่เหมาะสมนั้น ยังขึ้นอยู่กับข้อกำหนดของฟังก์ชันแรบปีง และการระบุถึงประสิทธิภาพที่หนักอึ้งด้วย

ในการหาค่าฟังก์ชันแรบปีง ได้มีการกำหนดเงื่อนไขดังนี้ คือ

ก. เงื่อนไขโมโนโทนิก (Monotonic Condition)

$$i(k-1) < i(k) \text{ และ } j(k-1) < j(k)$$

ข. เงื่อนไขความต่อเนื่อง (Continuity Condition)

$$i(k) - i(k-1) < 1 \text{ และ } j(k) - j(k-1) < 1$$

จากเงื่อนไขทั้งสองนี้ ทำให้ได้ความสัมพันธ์ระหว่าง 2 จุดคือ

$$c(k-1) = \left[ \begin{array}{c} ( i(k), j(k-1) ) \\ ( i(k-1), j(k-1) ) \\ \text{หรือ } ( i(k-1), j(k) ) \end{array} \right] \dots\dots(4.2.6)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมาตร และแกนเวลาไม่สมมาตร กับการเปลี่ยนแปลงเงื่อนไขความชื้น

ในรูปที่ 4.2.3 จะแสดงผังของโปรแกรมย่อย DYNAMIC ซึ่งเป็นส่วนที่ใช้ในการเปรียบเทียบพารามิเตอร์ของสัญญาณเสียงพูด ระหว่างสัญญาณอ้างอิงกับสัญญาณทดสอบ

#### 4.3 กฎการตัดสินใจ (Decision Rules)

ผลที่ได้จากการเปรียบเทียบแบบไดนามิกโปรแกรมมี คือ ระยะทางแบบทดสอบกับแบบอ้างอิง แต่ละแบบระยะที่ได้ทั้งหมดจะนำมาผ่านขั้นตอนการตัดสินใจ เพื่อหาผลลัพธ์ของการรับรู้ กฎการตัดสินใจ (Decision Rules) ที่ใช้ในการรับรู้เสียงพูดมีอยู่ 3 แบบ คือ

##### 4.3.1 กฎ Nearest Neighbor (NN)

ผลลัพธ์ของการรับรู้จากกฎการตัดสินใจนี้ ได้แก่ แบบอ้างอิงที่มีระยะทางจากแบบทดสอบน้อยที่สุด กฎนี้เหมาะสำหรับระบบการรับรู้เสียงพูดแบบบุคคลเดียว

##### 4.3.2 กฎ K-Nearest Neighbor (KNN)

เป็นกฎที่ดัดแปลงมาจากกฎ NN อีกทีหนึ่ง เพื่อให้เหมาะสมกับการรับรู้เสียงพูดแบบต่างบุคคล ซึ่งมีแบบอ้างอิงในแต่ละคำมากกว่า 1 แบบ กฎนี้แบ่งออกเป็น 2 แบบ คือ

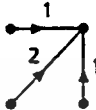
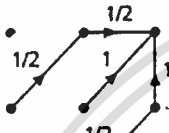
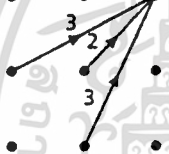
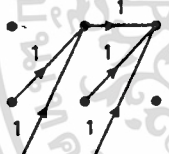
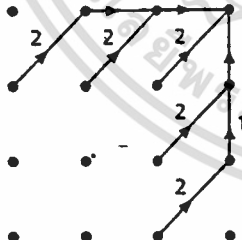
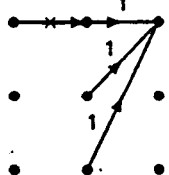
ก. แบบที่ 1 จะทำการหาแบบอ้างอิงจำนวน  $k$  แบบ ที่มีระยะทางห่างจากแบบทดสอบน้อยที่สุด ผลลัพธ์ของการรับรู้จะได้จาก แบบอ้างอิงที่มีจำนวนมากที่สุดในจำนวน  $K$  แบบ ที่ได้รับการเลือกมา

ข. แบบที่ 2 จะทำการหาแบบอ้างอิงจำนวน  $K$  แบบในแต่ละคำ โดยที่ทั้ง  $K$  แบบมีระยะทางห่างจากแบบทดสอบน้อยที่สุด ผลลัพธ์ของการรับรู้จะได้จาก แบบอ้างอิงที่ค่าเฉลี่ยของระยะทางในจำนวน  $K$  แบบที่ได้รับการเลือกมาน้อยที่สุด วิธีนี้จะให้ผลลัพธ์ที่ถูกต้องมากขึ้น โดยกำหนดค่า  $K$  เป็น 2 หรือ 3 เมื่อเทียบกับกฎ NN (Rabiner L.R., 1979) อย่างไรก็ตามวิธีนี้จะทำให้เวลาที่ใช้ในการรับรู้เพิ่มขึ้นด้วย

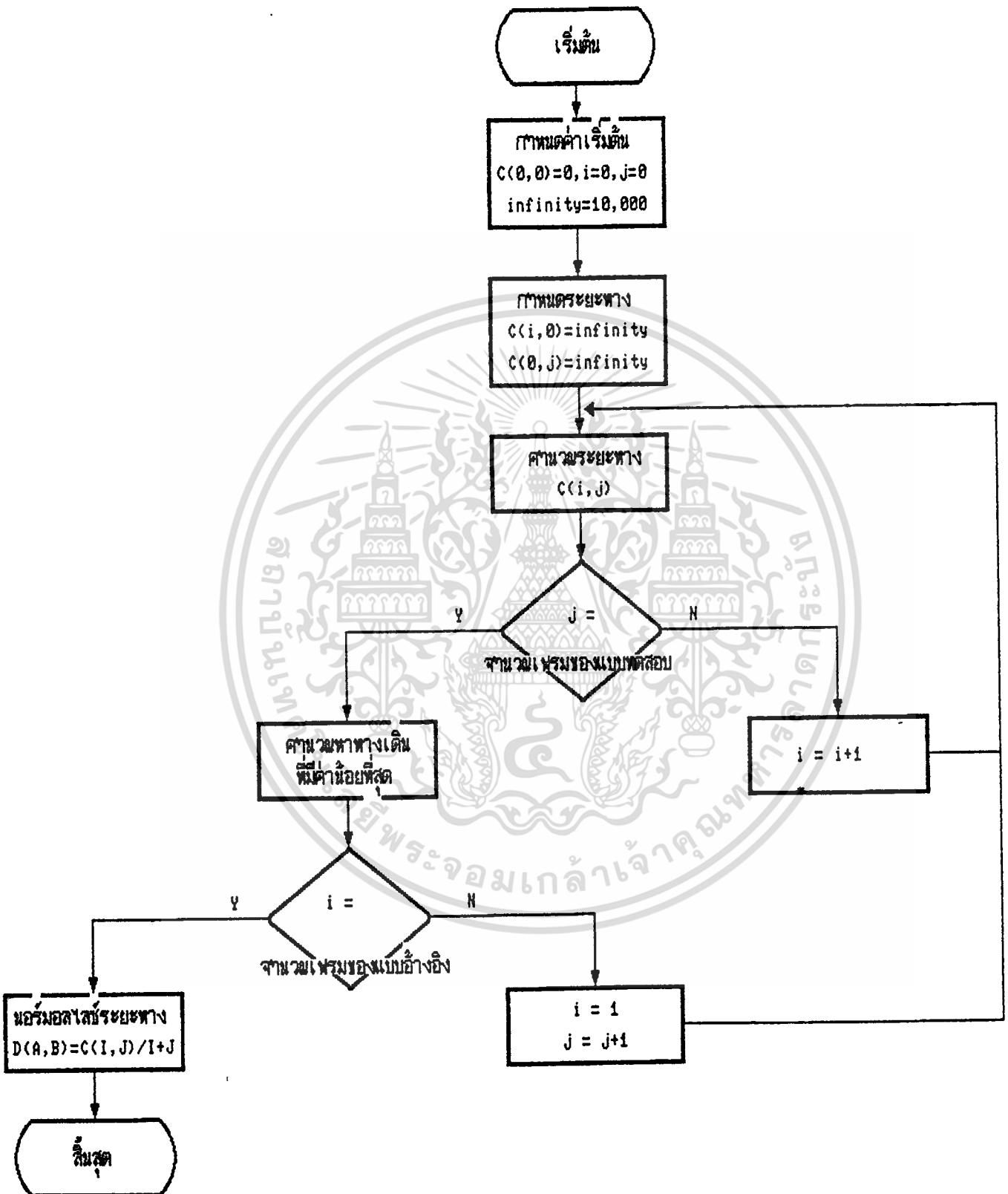
ในระบบการรับรู้เสียงพูดทั่วไป จะมีการกำหนดค่าระยะทางมากที่สุดที่ยอมรับได้ (Reject Value) เพื่อเพิ่มความถูกต้องของการรับรู้ในกรณีที่ผลลัพธ์ที่ได้จากการตัดสินใจมีค่าใกล้เคียงกันมาก วิธีการกำหนดค่านี้ต้องใช้ความระมัดระวังเป็นอย่างมาก เนื่องจากถ้าเรากำหนดค่านี้สูงเกินไป หรือไม่ได้กำหนด โอกาสที่จะทำให้ผลลัพธ์ของการรับรู้ผิดพลาดไปจะมีมาก แต่ถ้าเรากำหนดค่านี้ต่ำเกินไป จะทำให้ผลลัพธ์ของการรับรู้สำหรับแบบอ้างอิงที่ถูกต้องเกิดขึ้นได้ยาก ดังนั้นโดยทั่วไปค่าระยะทางนี้มักจะได้มาจากการทดลอง เพื่อหาค่าที่เหมาะสมสำหรับกลุ่มของแบบอ้างอิงหนึ่งๆเท่านั้น (Holtzman J., 1984)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE 4.7. Summary of sets of local constraints, slope weights, and DP recursion formulas

| Local Constraints & Slope Weights   | DP Recursion Formula  |
|---|---|
|    | $\min \left\{ \begin{array}{l} D(i_x - 1, i_y) + d(i_x, i_y), \\ D(i_x - 1, i_y - 1) + 2d(i_x, i_y), \\ D(i_x, i_y - 1) + d(i_x, i_y) \end{array} \right\}$   |
|    | $\min \left\{ \begin{array}{l} D(i_x - 2, i_y - 1) + \frac{1}{2}[d(i_x - 1, i_y) + d(i_x, i_y)], \\ D(i_x - 1, i_y - 1) + d(i_x, i_y), \\ D(i_x - 1, i_y - 2) + \frac{1}{2}[d(i_x, i_y - 1) + d(i_x, i_y)] \end{array} \right\}$  |
|   | $\min \left\{ \begin{array}{l} D(i_x - 2, i_y - 1) + 3d(i_x, i_y), \\ D(i_x - 1, i_y - 1) + 2d(i_x, i_y), \\ D(i_x - 1, i_y - 2) + 3d(i_x, i_y), \end{array} \right\}$  |
|  | $\min \left\{ \begin{array}{l} D(i_x - 2, i_y - 1) + d(i_x - 1, i_y) + d(i_x, i_y), \\ D(i_x - 2, i_y - 2) + d(i_x - 1, i_y) + d(i_x, i_y), \\ D(i_x - 1, i_y - 1) + d(i_x, i_y), \\ D(i_x - 1, i_y - 2) + d(i_x, i_y), \end{array} \right\}$   |
|  | $\min \left\{ \begin{array}{l} D(i_x - 3, i_y - 1) + 2d(i_x - 2, i_y) + d(i_x - 1, i_y) + d(i_x, i_y), \\ D(i_x - 1, i_y - 1) + 2d(i_x - 1, i_y) + d(i_x, i_y), \\ D(i_x - 1, i_y - 1) + 2d(i_x, i_y), \\ D(i_x - 1, i_y - 2) + 2d(i_x, i_y - 1) + d(i_x, i_y), \\ D(i_x - 1, i_y - 2) + 2d(i_x, i_y - 1) + d(i_x, i_y), \\ D(i_x - 1, i_y - 3) + 2d(i_x, i_y - 2) + d(i_x, i_y - 1) + d(i_x, i_y), \end{array} \right\}$ |
|  | $\min \left\{ \begin{array}{l} D(i_x - 1, i_y)g(k) + d(i_x, i_y), \\ D(i_x - 1, i_y - 1) + d(i_x, i_y), \\ D(i_x - 1, i_y - 2) + d(i_x, i_y), \end{array} \right\}$ <p>with <math display="block">g(k) = \begin{cases} 1 &amp; \phi(k-1) \neq \phi(k-2) \\ \infty &amp; \phi(k-1) = \phi(k-2) \end{cases}</math></p>  |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ **ตารางที่ 4.2.1 แสดงสมการไดนามิกโปรแกรมมิ่งต่างๆ** ด้านการคำนวณด้านค่า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา **เมื่อเงื่อนไขการเปลี่ยนความชันเปลี่ยนไป** นำไปใช้

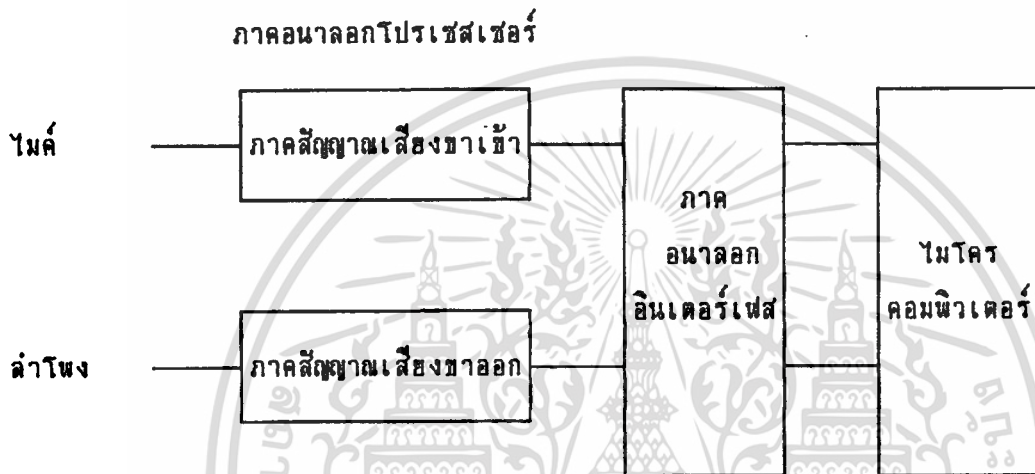


เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ที่ 4.2.3 แสดงผังของโปรแกรม DYNAMIC ที่ให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ ๕

#### การออกแบบการ์ดบันทึกและแสดงเสียงพูด

การ์ดที่ออกแบบนี้จะนำไปใช้ร่วมกับไมโครคอมพิวเตอร์ โดยจะใช้การ์ดนี้สำหรับการบันทึกสัญญาณเสียงลงบนแผ่นดิสก์ในรูปแบบสัญญาณดิจิทัล และสามารถที่จะนำเอาเสียงที่ถูกบันทึกไว้นั้นแสดงให้อยู่ในรูปแบบของสัญญาณเสียงดั้งเดิม ซึ่งส่วนประกอบในการทำงานสามารถแสดงได้ดังรูปที่ 5.1



รูปที่ 5.1

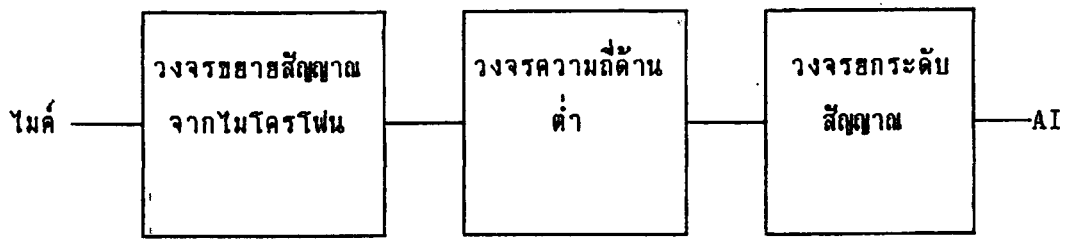
จากรูปที่ 5.1 ส่วนประกอบของการ์ดมีอยู่สองส่วน คือ ภาคอนาล็อกโปรเซสเซอร์ (analog processor) และ ภาคอนาล็อกอินเทอร์เฟซ(analog interface AI) ซึ่งสามารถแยกอธิบายได้ดังนี้

#### 5.1 ภาคอนาล็อกโปรเซสเซอร์ สามารถแบ่งได้เป็นสองส่วน

##### 5.1.1 ภาคสัญญาณเสียงขาเข้า

ภาคสัญญาณเสียงขาเข้าทำหน้าที่เปลี่ยนคุณสมบัติทางด้านความถี่และขนาดของสัญญาณเสียง (สัญญาณอนาล็อก) ให้เหมาะสมก่อนที่จะส่งสัญญาณเข้าไปในภาคอนาล็อกอินเทอร์เฟซ ซึ่งภาคนี้จะทำการแปลงสัญญาณสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัล ดังรูปที่ 5.2 แสดงส่วนประกอบของภาคสัญญาณเสียงขาเข้า ซึ่งประกอบไปด้วย ไมโครโฟน วงจรขยายสัญญาณจากไมโครโฟน (pre-amplifier) วงจรกรองความถี่ต่ำ (Low pass filter) และวงจรยกระดับสัญญาณ (Shifting amplifier)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ส่วนประกอบของภาคสัญญาณเสียงขาเข้า

ก. วงจรขยายสัญญาณจากไมโครโฟน

เนื่องจากขนาดสัญญาณเสียงที่ได้จากไมโครโฟนนั้นมีขนาดของสัญญาณต่ำมาก จึงต้องทำการขยายสัญญาณเสียงเหล่านี้ก่อนดังรูปที่ 5.3 แสดงวงจรขยายสัญญาณจากไมโครโฟน โดยใช้ไอซีออปแอมป์ LM318 นำมาต่อเป็นวงจรขยายแบบไม่กลับเฟส (non-inverting amplifier) ซึ่งตัวเก็บประจุ  $C_u$  มีหน้าที่ป้องกันแรงดันไฟตรงจากภาคก่อนหน้านี้ มิเช่นนั้นแรงดันไฟตรงนี้จะถูกขยายและทำให้เอาต์พุตของวงจรขยายมีค่าเกิน  $+V_{cc}$  (แรงดันอ้อมตัว) เป็นผลให้สัญญาณเสียงผิดพลาดจากความไม่เป็นจริง และ  $C_u$  กับ  $R_u$  ยังใช้เป็นตัวกำหนดความถี่ของสัญญาณอินพุตที่สามารถผ่านเข้าสู่ขั้วสัญญาณอินพุตได้ ซึ่งสามารถหาได้จาก

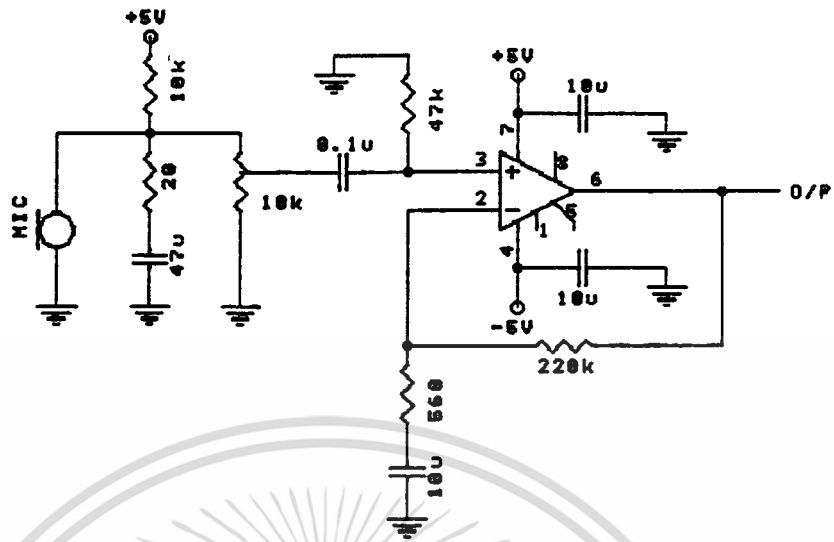
$$\begin{aligned}
 f_c &= 1/2\pi R_u C_u \\
 &= 1/2\pi 47k * 0.1\mu F \\
 &= 33.8 \text{ Hz}
 \end{aligned}$$

และมีอินพุตอิมพีแดนซ์ประมาณ  $R_u$

จากวงจรสามารถคำนวณหาค่าอัตราขยายได้จาก

$$\begin{aligned}
 V_o/V_i &= R_f + R_{in}/R_{in} \\
 &= 1 + R_f/R_{in} \\
 &= 1 + 220k/560 = 393.86 \\
 &= 20 \log(393.86) = 51.9 \text{ dB}
 \end{aligned}$$

LM318

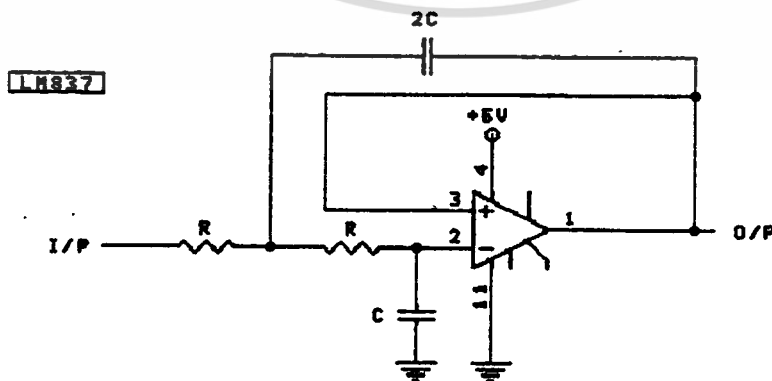


รูปที่ 5.3 วงจรขยายสัญญาณ

ข. วงจรกรองความถี่ต่ำ

วงจรกรองความถี่ต่ำจะยอมให้สัญญาณที่มีความถี่ต่ำกว่าหรือเท่ากับความถี่ที่ต้องการผ่านเข้ามา ความถี่เรียกว่า ความถี่คัตออฟ ( $f_c$ ) โดยที่สัญญาณที่มีความถี่สูงกว่า  $f_c$  จะถูกจำกัดออกไป ในรูปที่

5.4 แสดงวงจรกรองความถี่ต่ำ ใช้ไอซีออปแอมป์ LM837 ซึ่งเป็นวงจรกรองความถี่ต่ำอันดับที่ 2 ( $2^{nd}$  order low pass filter) โดยมีความชันประมาณ  $-40$  dB/decade



รูปที่ 5.4 วงจรกรองความถี่ต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในภาคสัญญาณเสียงขาเข้านี้จะประกอบไปด้วยวงจรรองความถี่ทั้งหมด 3 ชุด โดยแต่ละจะมีความถี่คutoffที่แตกต่างกัน คือ 3kHz, 4kHz, 5kHz เหตุที่มีวงจรรองความถี่ทั้งหมด 3 ชุด ก็เพื่อให้ผู้ใช้เลือกบันทึกเสียงที่มีช่วงความถี่ที่แตกต่างกัน ซึ่งเราสามารถคำนวณหาความถี่คutoffได้จาก

$$f_c = 0.1125/RC$$

วงจรรองความถี่ต่ำที่มี  $f_c = 3 \text{ kHz}$  หาได้จาก

$$\begin{aligned} f_c &= 0.1125/1.7\text{k}\cdot 22\text{nF} \\ &= 3.008 \text{ kHz} \end{aligned}$$

วงจรรองความถี่ต่ำที่มี  $f_c = 4 \text{ kHz}$  หาได้จาก

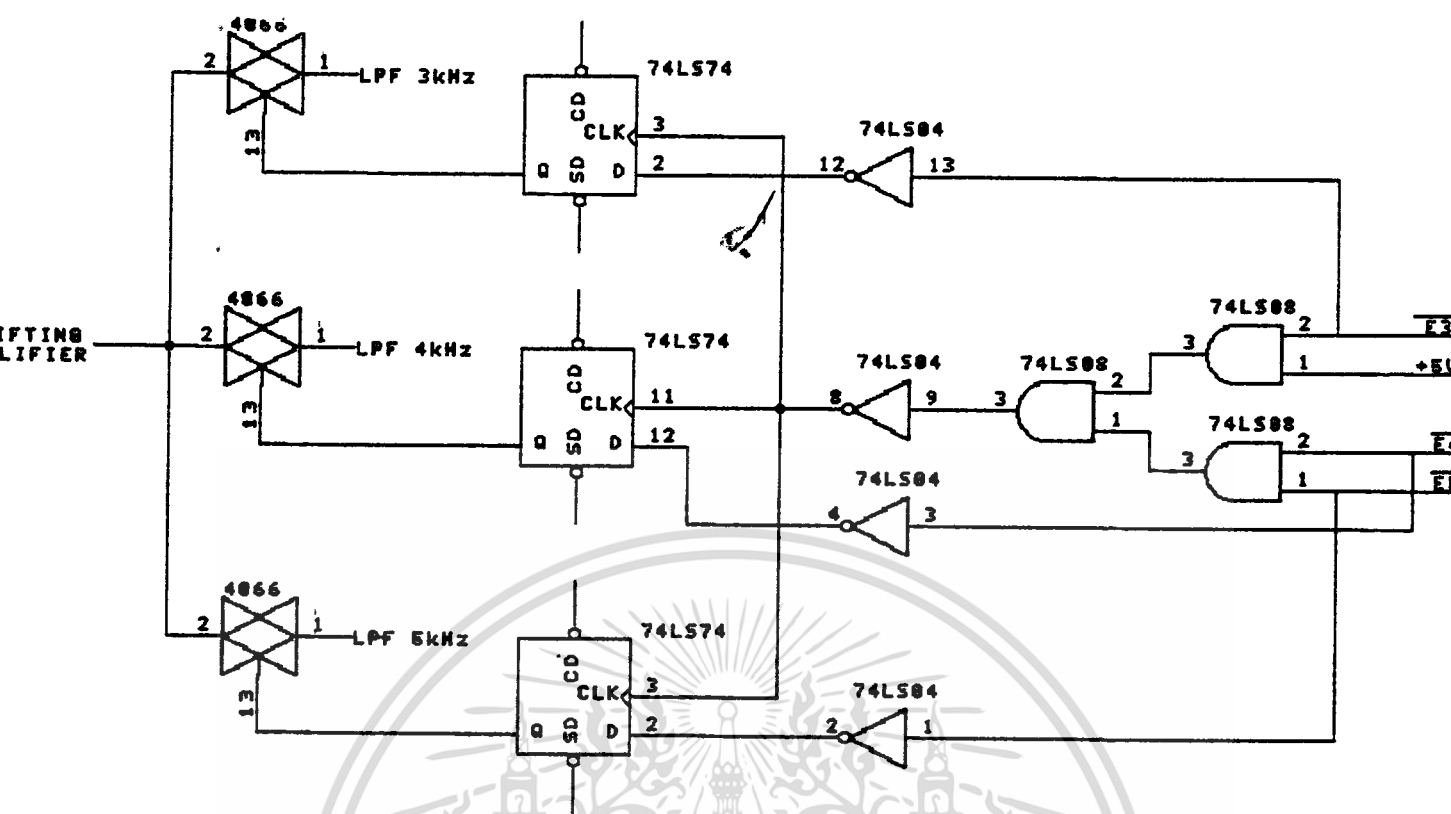
$$\begin{aligned} f_c &= 0.1125/1.9\text{k}\cdot 15\text{nF} \\ &= 3.95 \text{ kHz} \end{aligned}$$

วงจรรองความถี่ต่ำที่มี  $f_c = 5 \text{ kHz}$  หาได้จาก

$$\begin{aligned} f_c &= 0.1125/1.5\text{k}\cdot 15\text{nF} \\ &= 5 \text{ kHz} \end{aligned}$$

การเลือกวงจรรองความถี่ต่ำทั้ง 3 ชุดนี้สามารถทำได้โดยการใช้ซอฟต์แวร์ โดยจะมีการกำหนดพอร์ตสำหรับวงจรรองความถี่ต่ำแต่ละชุด (เบอร์พอร์ตสามารถพิจารณาได้ในส่วนของวงจรเลือกพอร์ต) และเมื่อมีการเลือกวงจรรองความถี่ชุดใดชุดหนึ่ง พอร์ตของวงจรรองความถี่นั้นจะแอกทีฟ อยู่ในสถานะต่ำ "0" และส่งต่อไปยัง 74LS04 เพื่อเปลี่ยนสถานะให้อยู่ในสถานะสูง "1" แล้วส่งต่อไปยัง 74LS74 ซึ่งทำหน้าที่เป็นตัวควบคุม 4066 คือว่าถ้ามีสัญญาณสถานะสูง "1" ผ่านเข้ามา 74LS74 จะส่งสัญญาณที่มีสถานะสูงไปยัง ทรานซิสเตอร์ของ 4066 เพื่อเป็นการกำหนดให้วงจรรองความถี่ต่ำที่ต่ออยู่กับ 4066 ตัวนั้นสามารถทำงานได้ ดังแสดงในรูปที่ 5.5

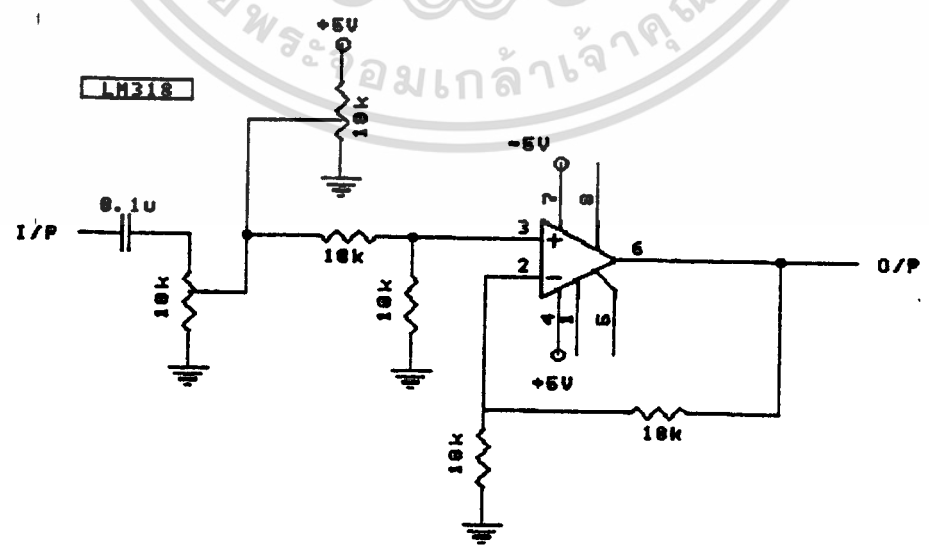
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.5 ส่วนควบคุมการเลือกวงจรรองความถี่

ค. วงจรยกระดับสัญญาณ

ดังรูปที่ 5.6 เป็นวงจรยกระดับสัญญาณ เป็นวงจรที่ต่อจากวงจรรองความถี่ต่ำ มีไว้เพื่อทำการปรับระดับสัญญาณให้มีขนาดอยู่ระหว่าง 0-5 โวลต์ ก่อนที่จะทำการป้อนเข้าสู่ส่วนของการแปลงสัญญาณ



รูปที่ 5.6 วงจรยกระดับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.1.2 ภาคสัญญาณเสียงขาออก

เมื่อผู้ใช้ต้องการแสดงสัญญาณเสียงที่ถูกเก็บในรูปแบบสัญญาณดิจิทัล ให้กลับไปเป็นสัญญาณอนาลอกตามเดิม : ภาคอนาลอกอินเทอร์เฟซจะทำการแปลงสัญญาณนี้ แล้วส่งไปยังภาคสัญญาณเสียงขาออก ซึ่งภาคนี้จะทำการปรับปรุงคุณภาพของสัญญาณเสียงเพื่อที่จะให้เสียงที่ถูกแปลงออกมานั้นใกล้เคียงกับเสียงที่ถูกทำการบันทึกมากที่สุด ดังรูปที่ 5.7 แสดงส่วนประกอบของภาคสัญญาณเสียงขาออก ซึ่งประกอบไปด้วย วงจรรองความถี่ต่ำ วงจรขยายสัญญาณ (Power amplifier) และ ลำโพง



รูปที่ 5.7 ภาคสัญญาณเสียงขาออก

#### ก. วงจรรองความถี่ต่ำ

วงจรรองความถี่ต่ำจะมีให้เลือกอยู่ทั้งหมด 3 ชุด โดยแต่ละชุดจะมีความถี่คัตออฟที่แตกต่างกัน ซึ่งวงจรรองความถี่และส่วนควบคุมวงจรรองความถี่จะเหมือนกับส่วนของภาคสัญญาณเสียงขาเข้า เหตุที่ต้องผ่านวงจรรองความถี่ก่อนก็เพื่อให้ได้รูปคลื่นของสัญญาณเสียงที่แท้จริงออกมา

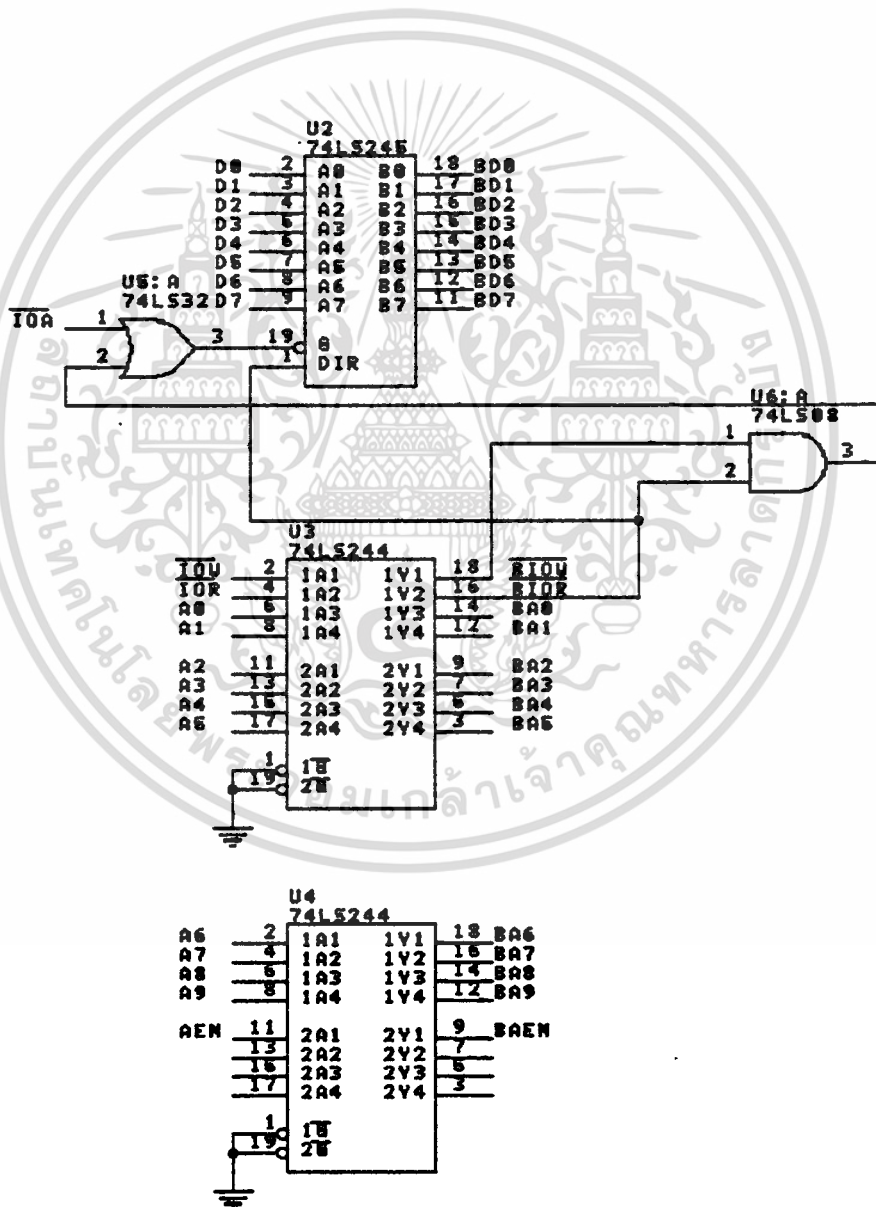
### 5.2 ภาคอนาลอกอินเทอร์เฟซ

เป็นส่วนที่ใช้ในการเชื่อมต่อเข้ากับไมโครคอมพิวเตอร์ ซึ่งมีหน้าที่ทำการแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัลเพื่อใช้สำหรับการบันทึกเสียงพูด และทำการแปลงสัญญาณดิจิทัลให้เป็นสัญญาณอนาลอกสำหรับการแสดงเสียงพูดให้ออกทางลำโพง ซึ่งในภาคนี้จะประกอบไปด้วยวงจรต่างๆดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.1 วงจรบัฟเฟอร์

ในรูปที่ 5,8 แสดงวงจรบัฟเฟอร์ ซึ่งเป็นส่วนที่ต่อโดยตรงกับ บัสข้อมูล(data bus) บัสแอดเดรส (address bus) และสัญญาณควบคุมต่างๆ เช่น BIOR, BLOW, CLK และ AEN ในรูปประกอบไปด้วย ไอซี 74LS245 เป็นบัฟเฟอร์แบบสองทิศทาง(Bi-direction)ซึ่งใช้เป็นบัฟเฟอร์สำหรับบัสข้อมูล (BD0-BD7) และ ไอซี 74LS244 เป็นบัฟเฟอร์แบบทิศทางเดียว ใช้เป็นบัฟเฟอร์สำหรับบัสแอดเดรส (A0-A9) และสัญญาณควบคุมอื่นๆ



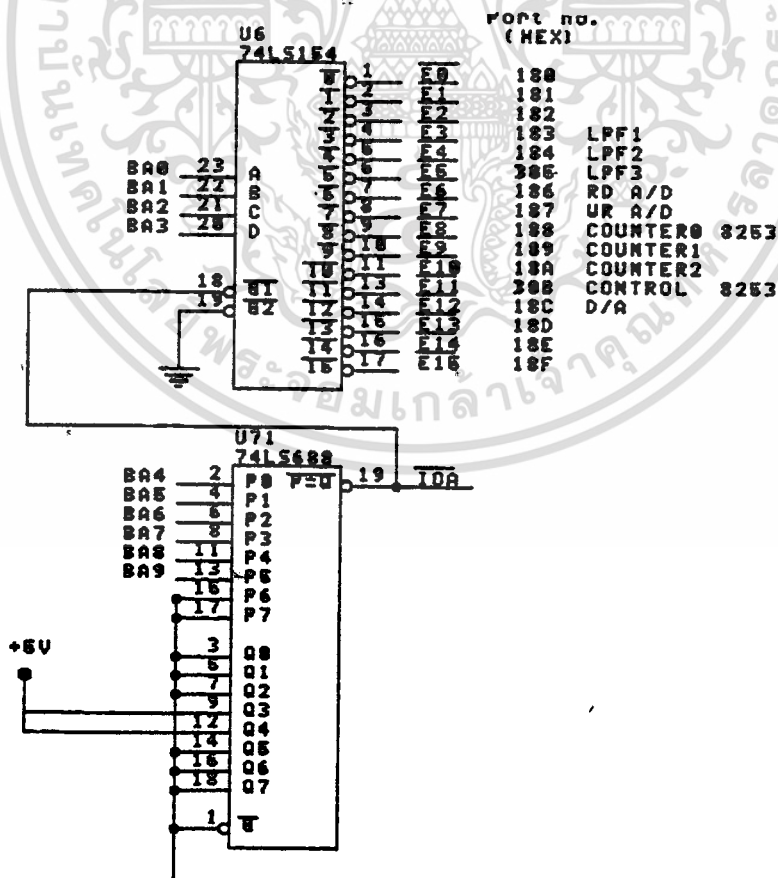
รูปที่ 5.8 วงจรบัฟเฟอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.2 วงจรเลือกพอร์ต

ในรูปที่ 5.9 แสดงวงจรรวมการเลือกพอร์ต ใช้ไอซี 74LS154 เป็นวงจรถอดรหัสแบบ 4 เป็น 16 ซึ่งทำให้สามารถกำหนดพอร์ตได้มากที่สุด 16 พอร์ต และไอซี 74LS688 เป็นคอมพาราเตอร์ มีไว้สำหรับกำหนดเบอร์พอร์ตที่แน่นอนเพื่อที่จะควบคุมอุปกรณ์ต่างๆบนการ์ด ซึ่งพอร์ตต่างๆถูกกำหนดไว้ใช้งานดังนี้

- พอร์ต 183H เป็นพอร์ตเลือกวงจรถอดความถี่ต่ำ 3kHz
- พอร์ต 184H เป็นพอร์ตเลือกวงจรถอดความถี่ต่ำ 4kHz
- พอร์ต 185H เป็นพอร์ตเลือกวงจรถอดความถี่ต่ำ 5kHz
- พอร์ต 186H เป็นพอร์ตเลือกการอ่านเอทีดี (RD A/D)
- พอร์ต 187H เป็นพอร์ตเลือกการเขียนเอทีดี (WR A/D)
- พอร์ต 188H เป็นพอร์ตเลือกวงจรมับตัวที่ 0 ของ 8253
- พอร์ต 189H เป็นพอร์ตเลือกวงจรมับตัวที่ 1 ของ 8253
- พอร์ต 18AH เป็นพอร์ตเลือกวงจรมับตัวที่ 2 ของ 8253
- พอร์ต 18BH เป็นพอร์ตเลือกรีจิสเตอร์ควบคุม 8253
- พอร์ต 18CH เป็นพอร์ตเลือกคิกเอ (D/A)

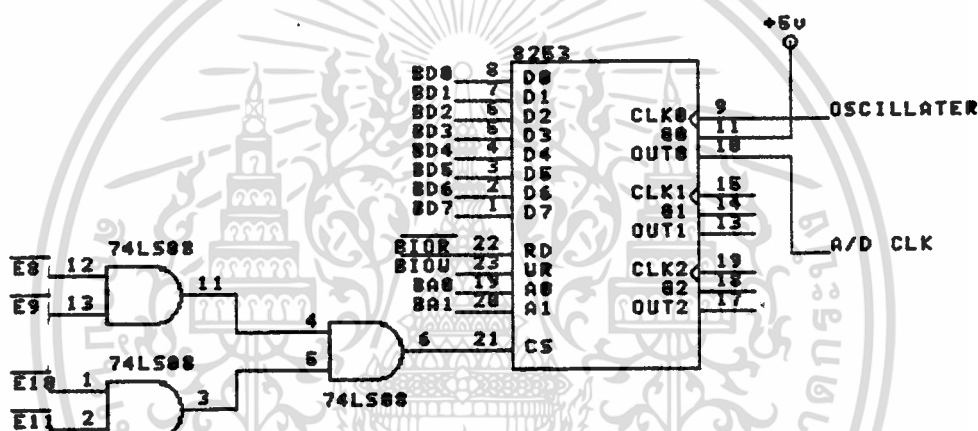


รูปที่ 5.9 วงจรเลือกพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3 วงจรส่วนสร้างสัญญาณนาฬิกา

ในรูปที่ 5.10 แสดงวงจรส่วนสร้างสัญญาณนาฬิกา ซึ่งประกอบไปด้วย วงจรออสซิลเลเตอร์ที่ใช้คริสตัล 2 MHz ซึ่งวงจรนี้จะทำการกำเนิดสัญญาณนาฬิกาความถี่ 2 MHz ออกมา ต่อจากนั้นจะทำการผ่านเข้าสู่ขา CLK 0 ของตัวนับที่ 0 (COUNTER 0) ของไอซี 8253 แล้วทำการโปรแกรม 8253 ให้ตัวนับที่ 0 อยู่ในโหมดที่ 2 ซึ่งเป็นโหมดที่ใช้สำหรับหารความถี่ของสัญญาณที่เข้ามา โดยจะทำการหารสัญญาณความถี่ 2 MHz ด้วย 3 จะทำให้ได้สัญญาณออกมาจากขา OUT 0 ซึ่งสัญญาณนี้จะมีค่าถี่เหลือเพียง 666 kHz แล้วนำไปเป็นสัญญาณนาฬิกาของตัวแปลงสัญญาณอนาลอกเป็นดิจิทัล



รูปที่ 5.10 วงจรส่วนสร้างสัญญาณนาฬิกา

### 5.4 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

ในรูปที่ 5.11 แสดงวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล โดยการใช้ไอซี ADC 0804 ไอซีชนิดนี้ใช้สำหรับการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลที่มีขนาด 8 บิตจากรูปจะเห็นที่เนอร์โคโดด LM 336 (2.5V) ต่อไว้ที่ขา  $V_{ref}/2$  ก็เพื่อที่จะเป็นตัวกำหนดโวลเตจที่ป้อนไปยังขา  $V_{ref}/2$  ของตัวแปลงสัญญาณ ซึ่งโวลเตจนี้ใช้เป็นตัวกำหนดช่วงของระดับสัญญาณที่ใช้ในการแปลง (span) ให้กับตัวแปลงสัญญาณ ซึ่งในวงจรนี้จะกำหนดให้มี  $span = 5V$  คือช่วงสัญญาณที่จะถูกทำการแปลงจะต้องอยู่ระหว่าง 0-5 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

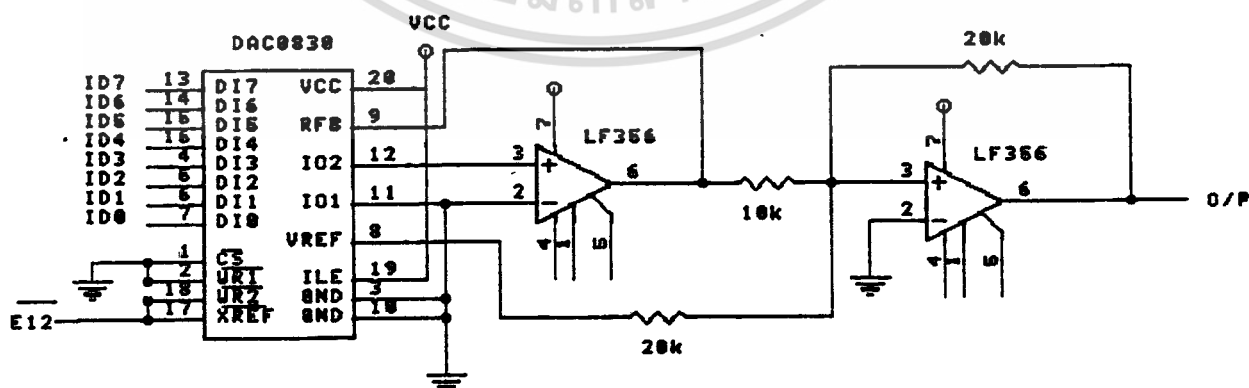
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรแปลงสัญญาณจะเริ่มทำงานเมื่อมีสัญญาณจากพอร์ต 187 ที่มีสถานะต่ำส่งไปยังขา WR ของตัวแปลง ซึ่งสัญญาณนี้จะไปทำการรีเซ็ตรีจิสเตอร์ภายใน จากนั้นก็จะทำการเปลี่ยนสัญญาณเป็นสัญญาณดิจิทัล สัญญาณที่ถูกแปลงจะไปรออยู่ที่ TRI-STATE OUTPUT LATCHES ของตัวแปลงสัญญาณ ที่ขาของ INTR ของตัวแปลงสัญญาณ จะมีสัญญาณ "0" ถูกส่งออกมา โดยสัญญาณนี้จะไปผ่านคอนเวอร์เตอร์ 74LS04 ซึ่งจะทำการทำให้สัญญาณเป็น "1" ต่อจากนั้นสัญญาณจะถูกส่งไปยังฟลิปฟลอป 74LS74 และบัฟเฟอร์ 74LS125 โดยที่ผู้ใช้จะคอยตรวจสอบที่บิตที่ 0 (BD0) ของพอร์ต 18DH ว่ามีลอจิกเป็น "1" หรือไม่ ถ้าเป็น "1" แสดงว่าการแปลงข้อมูลสิ้นสุดแล้ว ถ้าเป็น "0" แสดงว่าเอทูดูอยู่ระหว่างการแปลงสัญญาณข้อมูล ถ้าพบลอจิกเป็น "1" ที่พิน 8088 ก็จะทำการอ่านข้อมูลเข้าจากพอร์ต 186H ทำให้มีการส่งสัญญาณ RD ออกมาซึ่งจะมารีเซ็ตฟลิปฟลอป 74LS74 ทำให้ BD0 กลับเป็น 0 จนกว่าข้อมูลตัวต่อไปจะถูกแปลงเสร็จ เป็นเช่นนี้ไปเรื่อย ๆ

สัญญาณ "0" จากอินเทอร์รัพท์ของเอทูดีกส่วนจะถูกส่งผ่านอินเวอร์ตเตอร์เพื่อนำไปควบคุม 4066 (อนาล็อกสวิตช์) เพื่อทำการสลับสัญญาณอนาล็อกเข้ามาใหม่ จนกว่าจะมีคำสั่งเขียนพอร์ต 187 เพื่อรีเซ็ตเอทูดูโดยจะมีผลทำให้อินเทอร์รัพท์มีค่าเป็น "1" เมื่อผ่านอินเวอร์ตเตอร์จะเปลี่ยนเป็น "0" ทำให้หยุดการสลับสัญญาณอนาล็อกทาง 4066 และเอทูดูก็กจะเริ่มการแปลงสัญญาณ เป็นเช่นนี้ไปเรื่อย ๆ

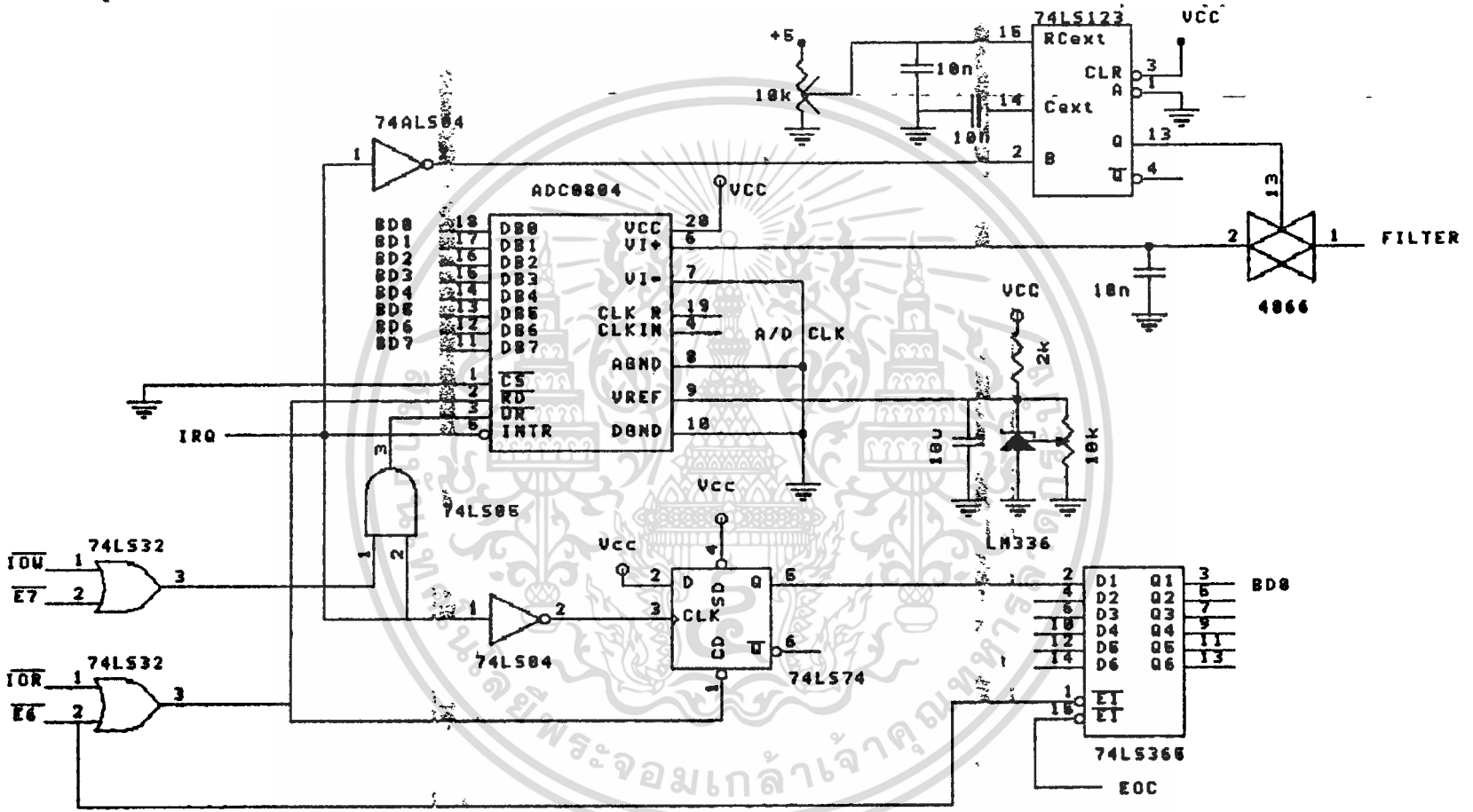
### 5.5 วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก

ในรูปที่ 5.12 แสดงวงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก โดยการใช้ไอซี DAC0830 ซึ่งเป็นตัวแปลงขนาด 8 บิต วงจรนี้จะเริ่มการแปลงเมื่อมีสัญญาณ "0" ที่ส่งจากพอร์ต 18CH เพื่อไปควบคุมการทำงาน WR2 และ XFER ของ DAC0830 ข้อมูลที่ถูกเก็บในบัฟเฟอร์ในรูปของสัญญาณดิจิทัลก็จะถูกแปลงเป็นสัญญาณอนาล็อก สัญญาณอนาลอกนี้จะถูกแปลงออกมาในรูปของกระแส จึงต้องนำไปผ่านออปแอมป์ LF356 เพื่อแปลงให้อยู่ในรูปศักย์ค่าไฟฟ้ (โวลต์เดจ) ก่อนที่จะนำมาออกสู่โอเอาต์พุต



รูปที่ 5.12 แสดงวงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

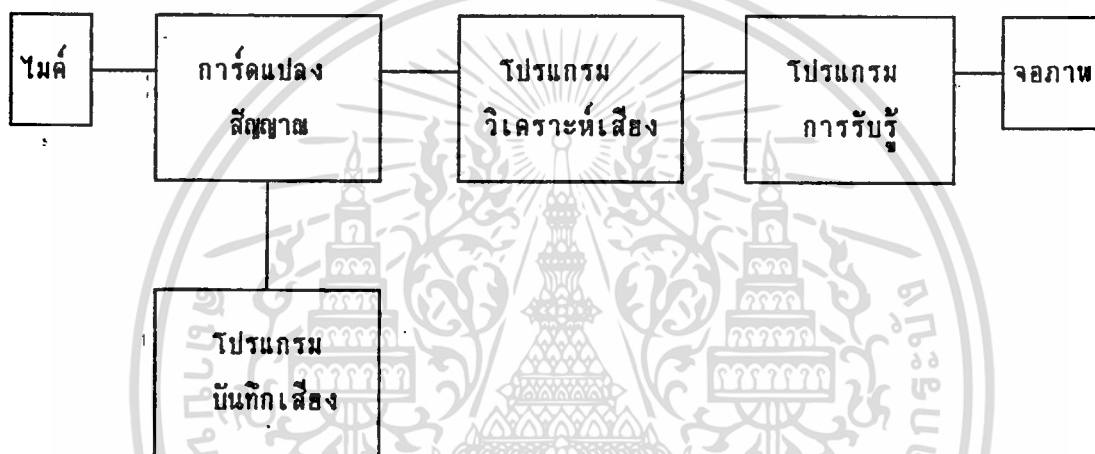


รูปที่ 5:11 แสดงวงจรสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

## บทที่ ๖

### การรับรู้เสียงพูดบนไมโครคอมพิวเตอร์

ขั้นตอนการรับรู้เสียงพูดจะประกอบไปด้วย ส่วนใหญ่ ๆ 2 ส่วน คือ 1) ส่วนของฮาร์ดแวร์ จะประกอบไปด้วยไมโครคอมพิวเตอร์ ไมโครโฟน และการ์ด ที่ใช้ในการแปลงสัญญาณเสียงให้เป็นสัญญาณดิจิทัล ซึ่งมีรายละเอียดอยู่ในบทที่ 5 2) ส่วนของซอฟต์แวร์ จะประกอบไปด้วย โปรแกรมที่ใช้ในการบันทึกเสียง โปรแกรมการวิเคราะห์เสียงพูด และ โปรแกรมการรับรู้ ซึ่งสามารถแสดงความสัมพันธ์ของทั้ง 2 ส่วนได้ ดังรูปที่ 6.1



รูปที่ 6.1 แสดงความสัมพันธ์ของส่วนฮาร์ดแวร์และซอฟต์แวร์

เพื่อให้เห็นขั้นตอนการรับรู้เสียงพูดได้ชัดเจนมากขึ้น เราจะนำส่วนของซอฟต์แวร์ที่ได้กล่าวไว้ข้างต้นมาแสดงให้ละเอียดลงไปอีกดังต่อไปนี้

1) โปรแกรมบันทึกเสียง เป็นโปรแกรมที่ใช้ในการควบคุมการ์ดบันทึกเสียงให้สามารถทำการแปลงสัญญาณเสียงให้เป็นสัญญาณดิจิทัลเก็บลงในแฟ้มข้อมูลได้

2) โปรแกรมการวิเคราะห์เสียงพูด เป็นโปรแกรมที่นำเอาเสียงพูดที่ถูกบันทึกมาทำการวิเคราะห์ โดยแบ่งได้เป็น

ก. โปรแกรมหาขอบเขตของสัญญาณ เป็นโปรแกรมที่ใช้สำหรับทำการหาขอบเขตของสัญญาณเสียงที่แท้จริง และทำการตัดสัญญาณส่วนที่ไม่ใช่เสียงทิ้งไป

ข. โปรแกรมหาค่าพารามิเตอร์ของสัญญาณ เป็นการท่านาสเชิงเส้นโดยวิธีอีต-สหสัมพันธ์ ซึ่งจะได้ค่าพารามิเตอร์ของสัญญาณ เพื่อนำไปใช้ในโปรแกรมถัดไป

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

๑) โปรแกรมการรับรู้เสียงพูด เป็นโปรแกรมที่นำค่าพารามิเตอร์ที่หาได้จาก 2ข. มาใช้ในการรับรู้เสียงพูด ซึ่งจะประกอบไปด้วย

ก. โปรแกรมสร้างแฟ้มข้อมูลพารามิเตอร์ มีหน้าที่นำค่าพารามิเตอร์ที่หาได้ในแต่ละเฟรมของข้อมูลแบบทดสอบมารวมกันอยู่ในแฟ้มข้อมูลเดียวกัน และมีหน้าที่นำค่าพารามิเตอร์ที่จะใช้เป็นแบบอ้างอิงของแต่ละคำมารวมกันอยู่ในแฟ้มข้อมูลเดียวกัน

ข. โปรแกรมคัดเลือกแบบอ้างอิง มีหน้าที่นำแฟ้มข้อมูลแบบอ้างอิงมาทำการคัดเลือกแบบอ้างอิงที่ดีที่สุด เพื่อนำมาใช้เป็นแบบอ้างอิงในโปรแกรมการรับรู้ต่อไป

ค. โปรแกรมการรับรู้ มีหน้าที่ในการนำเสียงที่เป็นแบบทดสอบมาเปรียบเทียบกับแบบอ้างอิงแต่ละคำ และจะทำการตัดสินใจว่าแบบทดสอบตรงกับแบบอ้างอิงใด โดยพิจารณาจากระยะทางระหว่างแบบทดสอบทั้งสอง ซึ่งมีระยะห่างกันน้อยที่สุด

ในรูปที่ 6.2 แสดงขั้นตอนในระบบการรับรู้เสียงพูด แบ่งได้เป็น 2 ขั้นตอน คือ

#### ก. ขั้นตอนการสร้างแบบอ้างอิง

1. หนึ่เสียงผ่านไมโครโฟนซึ่งต่ออยู่กับการบันทึกเสียง โดยที่โปรแกรมการบันทึกเสียงจะทำหน้าที่ควบคุมการดีให้สามารถแปลงสัญญาณเสียงให้เป็นข้อมูลดิจิทัล แล้วบันทึกข้อมูลเหล่านั้นลงแฟ้มข้อมูลเสียง

2. นำข้อมูลในแฟ้มข้อมูลเสียงนี้ไปหาขอบเขตของสัญญาณ เพื่อตัดส่วนของสัญญาณรบกวนออกไปให้เหลือแต่เพียงส่วนที่เป็นสัญญาณเสียงจริงเท่านั้น โดยการใช้โปรแกรมการหาขอบเขตของคำ

3. นำข้อมูลที่เป็นสัญญาณเสียงจริงที่หาได้จาก 2 นำไปคำนวณหาค่าพารามิเตอร์ โดยใช้โปรแกรมการคำนวณหาค่าพารามิเตอร์

4. นำค่าพารามิเตอร์ที่ได้ ทำการบันทึกลงแฟ้มข้อมูลพารามิเตอร์

5. เมื่อได้แบบอ้างอิงเพียงพอกับความต้องการ ก็จะทำการเลือกแบบอ้างอิงเพื่อใช้ในการรับรู้ต่อไป

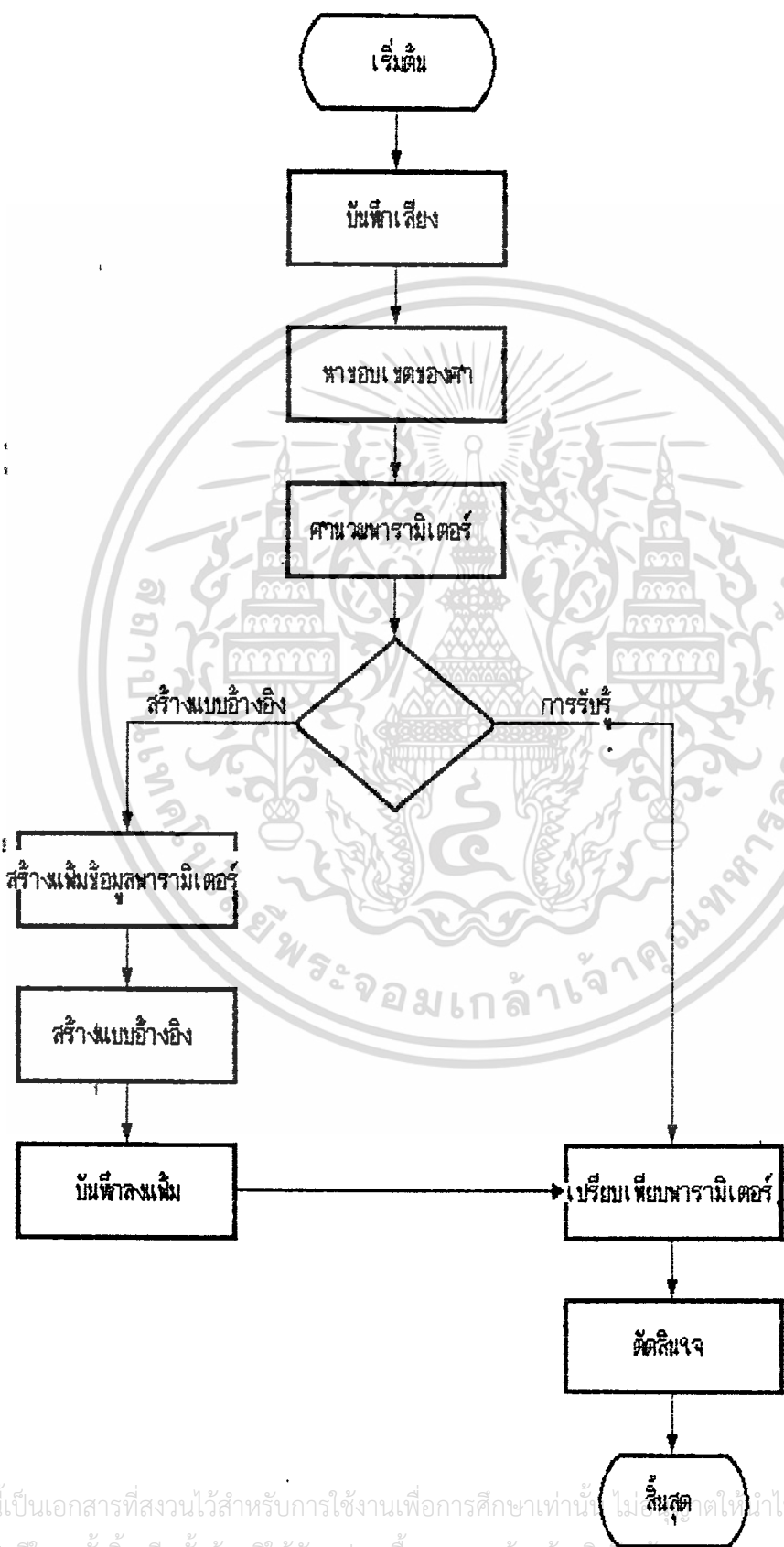
#### ข. ขั้นตอนการรับรู้

1. ขั้นตอนที่ 1, 2 และ 3 จะเหมือนกับขั้นตอนการสร้างแบบอ้างอิง

2. นำพารามิเตอร์ที่ได้ มาทำการเปรียบเทียบกับแบบอ้างอิงที่ได้สร้างไว้แล้ว โดยใช้

Dynamic Programming ในการคำนวณหาระยะทางระหว่างสัญญาณทั้งสอง เป็นตัวเปรียบเทียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
3. เมื่อทำการเปรียบเทียบจนครบทุกแบบอ้างอิงแล้ว ก็จะทำการตัดสินใจว่าการเลือกแบบอ้างอิงที่มีระยะทางสั้นที่สุด เมื่อเทียบกับแบบทดสอบ เป็นค่าที่ถูกต้อง



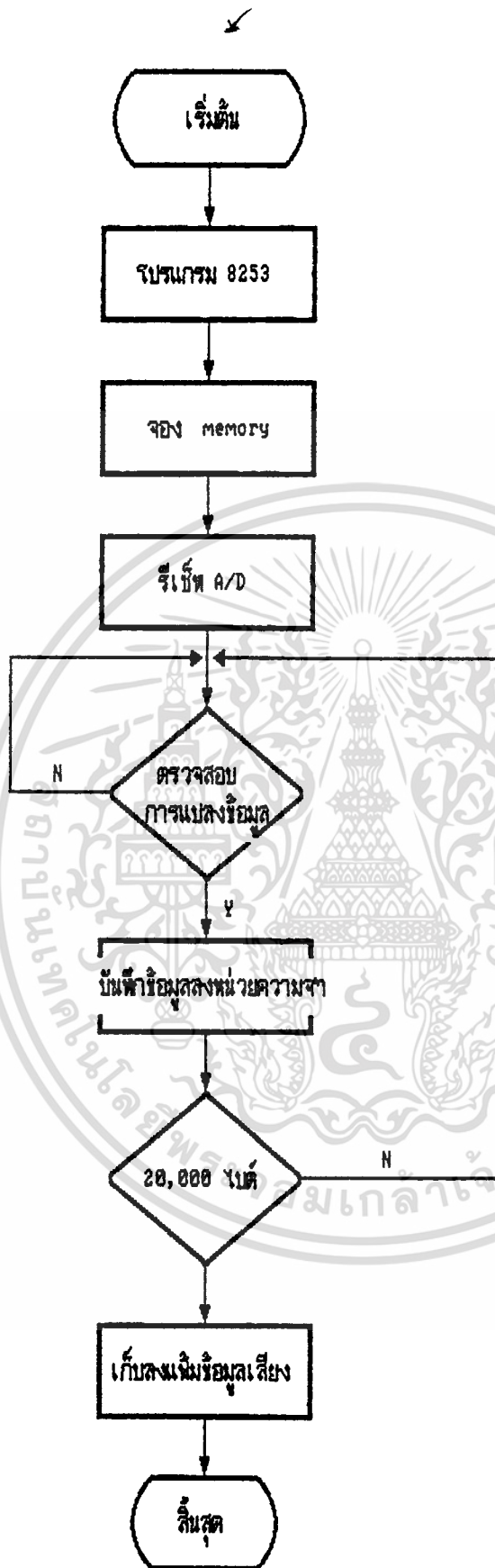
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.2 แผนผังแสดงขั้นตอนของการรับรู้เสียงบนไมโครคอมพิวเตอร์

## 6.1 โปรแกรมการบันทึกเสียง

โปรแกรมนี้อาจทำหน้าที่ควบคุมการ์ดให้สามารถทำงานร่วมกับไมโครคอมพิวเตอร์ ในการแปลงสัญญาณเสียงพูดที่พูดเข้าทางไมโครโฟนให้เป็นสัญญาณดิจิทัล แล้วบันทึกลงแฟ้มข้อมูล เพื่อนำไปใช้เป็นข้อมูลในการทำงานขั้นตอนต่อไป ซึ่งมีขั้นตอนการทำงานดังรูปที่ 6.3

1. ทำการโปรแกรม 8253 ที่อยู่บนการ์ดให้ทำงานใน Mode ที่ 2 เพื่อใช้ในการหารสัญญาณนาฬิกาจาก CPU และนำสัญญาณนาฬิกาที่หารได้นี้ไปป้อนให้กับตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล (บทที่ 5)
2. จากนั้นทำการจองเนื้อที่หน่วยความจำไว้ 20,000 ไบต์ เพื่อใช้ในการเก็บข้อมูลที่ได้รับการแปลง
3. ทำการรีเซ็ต A/D เพื่อให้เริ่มทำการแปลงสัญญาณ
4. โปรแกรมจะทำการตรวจสอบว่า A/D ทำการแปลงสัญญาณแต่ละไบต์เสร็จเรียบร้อยแล้วหรือยัง ถ้าเสร็จเรียบร้อยแล้วจะทำการบันทึกลงบนหน่วยความจำ
5. ทำการตรวจสอบว่าแปลงข้อมูลครบ 20,000 ไบต์หรือยัง ถ้าครบแล้วจะนำข้อมูลที่เก็บอยู่ในหน่วยความจำเก็บลงในแฟ้มข้อมูลสัญญาณเสียง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าในรูปแบบที่ ๑.๓ แผนผังแสดงขั้นตอนการทำงานของโปรแกรมการบันทึกเสียง

## 6.2 โปรแกรมการวิเคราะห์

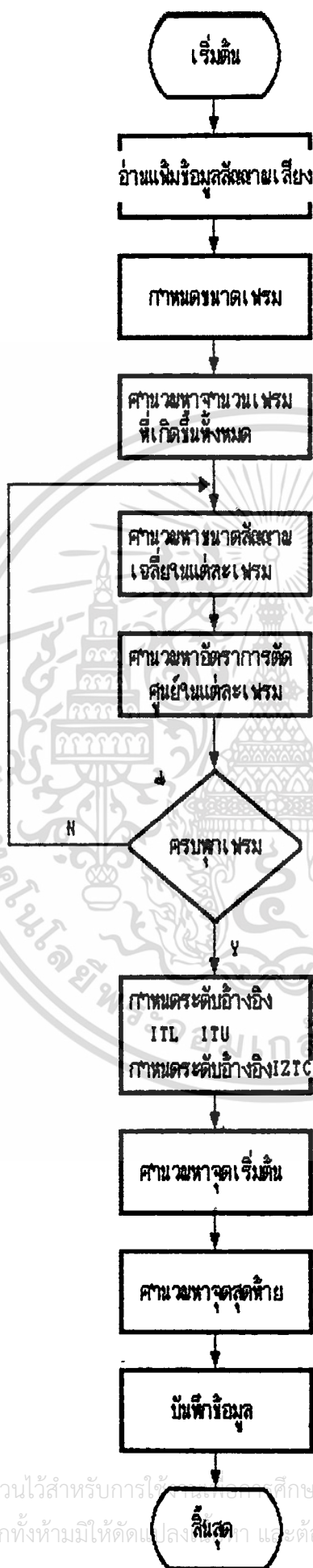
### 6.2.1 โปรแกรมหาขอบเขตของสัญญาณ

เป็นโปรแกรมที่ใช้ตรวจสอบหาขอบเขตของสัญญาณ โดยจะทำการตัดส่วนที่เป็นเสียงรบกวนที่ได้จากการบันทึกให้เหลือแต่เพียงส่วนของเสียงพูดจริงๆ โดยใช้วิธีการคำนวณขนาดเฉลี่ยของสัญญาณเสียง มาใช้ในการกำหนดจุดเริ่มต้นและจุดสุดท้ายของเสียง และเพื่อความถูกต้องยิ่งขึ้นของการหาจุดเริ่มต้นและจุดสุดท้ายของเสียง เราจะใช้วิธีคำนวณอัตราการตัดศูนย์มาช่วยอีกแรง ซึ่งวิธีการหาขอบเขตของสัญญาณเสียง สามารถศึกษาได้ในบทที่ 2 อีกครั้งหนึ่ง ขั้นตอนการทำงานของโปรแกรมรูปที่ 6.4

1. อ่านแฟ้มข้อมูลของสัญญาณเสียงลงหน่วยความจำ
2. กำหนดขนาดความกว้างของเฟรมที่ใช้สำหรับแบ่งสัญญาณเสียงออกเป็นส่วนๆ
3. คำนวณหาจำนวนเฟรมที่เกิดขึ้นได้ทั้งหมดจากสัญญาณเสียง
4. คำนวณหาขนาดของสัญญาณในแต่ละเฟรม จนครบทุกเฟรม
5. คำนวณหาอัตราการตัดศูนย์ในแต่ละเฟรม จนครบทุกเฟรม
6. คำนวณหาระดับอ้างอิง ITL ซึ่งระดับอ้างอิงนี้เป็นการคำนวณขนาดเฉลี่ยของ 100 ms แรกของช่วงเวลาที่ทำการบันทึกทั้งหมด ซึ่งเราถือว่าช่วง 100 ms แรกเป็นช่วงที่ไม่มีสัญญาณเสียงอยู่เลย
7. กำหนดระดับอ้างอิง ITU โดยที่  $ITU = 3 * ITL$  ระดับอ้างอิงนี้เป็นตัวบอกจุดเริ่มต้นและจุดสุดท้ายจะอยู่นอกช่วงเวลา
8. กำหนดระดับอ้างอิง IZCT เป็นการคำนวณหาอัตราการตัดศูนย์ในช่วงเวลา 100 ms ของช่วงเวลาที่ทำการบันทึกทั้งหมด
9. การคำนวณหาจุดเริ่มต้น จะนำเอาขนาดของสัญญาณที่ทำได้ในแต่ละเฟรม ตั้งแต่เฟรมแรก เปรียบเทียบระดับอ้างอิง ITU ถ้าขนาดของสัญญาณในแต่ละเฟรมมีขนาดมากกว่า ITU ก็จะทำการศึกษาขนาดของสัญญาณย้อนกลับ ไปจนกว่าขนาดของสัญญาณจะมีค่าน้อยกว่าระดับอ้างอิง ITL โดยจะถือว่าขนาดของสัญญาณค่าแรกที่มีค่าน้อยกว่าระดับอ้างอิง ITL เป็นจุดเริ่มต้นของสัญญาณ(N1) แต่เพื่อให้หาจุดเริ่มต้นได้ถูกต้องมากยิ่งขึ้น เราจะนำค่าอัตราการตัดศูนย์มาใช้ คือ จะนำค่าอัตราการตัดศูนย์ของ 25 เฟรมก่อนเฟรมที่ N1 มาพิจารณาว่า ณ.จุดนั้นค่าอัตราการตัดศูนย์มีค่ามากกว่า 3 เท่าของระดับอ้างอิง IZCT หรือไม่ ถ้าน้อยกว่าจะถือว่า N1 เป็นจุดเริ่มต้นที่แท้จริง แต่ในกรณีที่มากกว่า ตำแหน่งของ N1 จะถูกเปลี่ยนแปลง โดยจะทำการเลื่อนค่าอัตราการตัดศูนย์ ไปยังจุดที่มีค่ามากกว่าระดับอ้างอิง IZCT อยู่ 1 จุด และถือว่าจุดนี้เป็นจุดเริ่มต้นที่แท้จริง
10. การคำนวณหาจุดสุดท้าย จะทำการคำนวณเหมือนในข้อ 3 แต่แทนที่จะเก็บการพิจารณาสัญญาณย้อนกลับเหมือนในข้อ 9 แต่จะเป็นการพิจารณาสัญญาณไปข้างหน้า ซึ่งจะได้จุดท้ายออกมา
11. ทำการเก็บข้อมูลเฉพาะจุดเริ่มต้นและจุดสุดท้ายเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



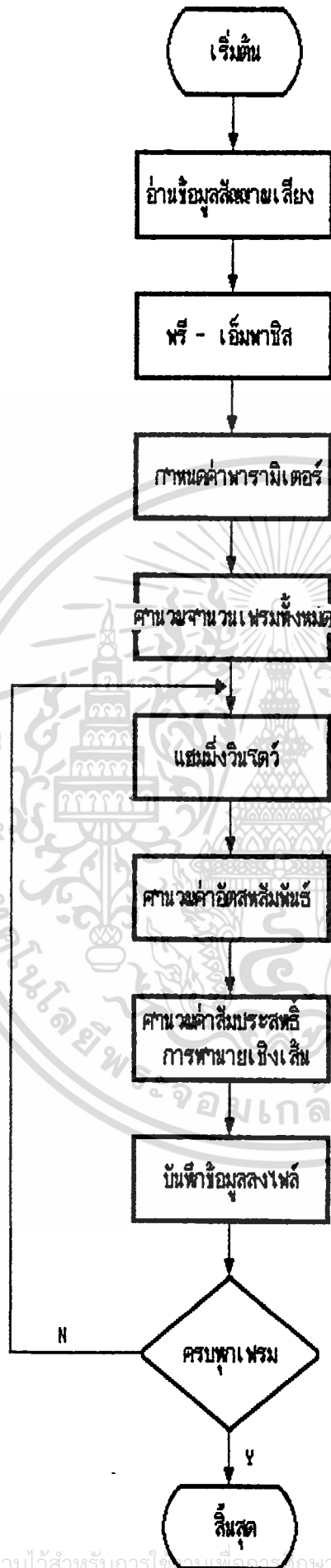
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ของนักศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.4 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมการหาขอบเขตของเสียง

## 6.2.2 โปรแกรมหาค่าพารามิเตอร์ของสัญญาณ

เป็นโปรแกรมที่ใช้ในการคำนวณหาค่าพารามิเตอร์ของสัญญาณเสียงโดยอาศัยวิธีการหาค่าสัมประสิทธิ์ออสสัมพันธ์ของสัญญาณในแต่ละเฟรม แล้วจึงนำค่าสัมประสิทธิ์ออสสัมพันธ์ในแต่ละเฟรมนี้ไปทำการคำนวณหาค่าพารามิเตอร์ฟิลเตอร์ต่อไป (อธิบายอย่างละเอียดในบทที่ 3.3) ขั้นตอนการทำงานของโปรแกรมแสดงดังรูปที่ 6.5

1. อ่านข้อมูลสัญญาณเสียง ซึ่งเป็นข้อมูลของเสียงที่ได้ผ่านการหาขอบเขตของคำมาแล้ว
2. สัญญาณเสียงทั้งหมดจะถูกพรี-เอมพาซิส โดยการใช้ดิซิทอลฟิลเตอร์ออร์หนึ่ง  $H(z) = 1 - az^{-1}$  ซึ่งกำหนดค่า  $a = 0.95$  จากการศึกษา [13] พบว่า พรี-เอมพาซิสช่วยลดความเริ่นซ์ของการคำนวณระยะทางซึ่งถูกใช้ในกระบวนการรับรู้ เมื่อพารามิเตอร์ LPC ถูกนำมาใช้ในการคำนวณ
3. กำหนดพารามิเตอร์ในการคำนวณ คือ ความยาวของเฟรม ระยะห่างระหว่างเฟรม คือ ระยะห่างระหว่างตัวอย่างแรกของเฟรมปัจจุบันกับตัวอย่างแรกของเฟรมก่อนหน้า และจำนวนออร์เดอร์ของฟิลเตอร์
4. ทำการคำนวณหาจำนวนเฟรมทั้งหมดของสัญญาณเสียงข้อมูล
5. นำสัญญาณทีละเฟรม มาทำการคูณด้วยฟังก์ชันแฮมมิงวินโดว์ ซึ่งขนาดของแฮมมิงวินโดว์ต้องมีขนาดเท่ากับขนาดของเฟรมของสัญญาณด้วย
6. นำสัญญาณที่ได้จากข้อที่ 5 มาทำการคำนวณหาค่าสัมประสิทธิ์ออสสัมพันธ์ ซึ่งค่าสัมประสิทธิ์ออสสัมพันธ์ที่ได้ จะมีจำนวนเท่ากับจำนวนออร์เดอร์ที่กำหนดมา โดยที่ค่าสัมประสิทธิ์ออสสัมพันธ์เหล่านี้จะถูกแปลงเป็นสัมประสิทธิ์การทำนายเชิงเส้น (พารามิเตอร์ฟิลเตอร์) โดยการใช้วิธีออสสัมพันธ์ (หัวข้อ 3.3)
7. ทำการบันทึกค่าพารามิเตอร์ที่ได้ลงในแฟ้มข้อมูล
8. ทำการตรวจสอบว่าโปรแกรมทำการคำนวณหาค่าพารามิเตอร์ของสัญญาณครบทุกเฟรมหรือยัง ถ้ายังไม่ครบจะกลับไปทำตั้งแต่ข้อ 4 - 7 ใหม่ แต่ถ้าทำครบทุกแล้ว จะถือว่าเป็นการสิ้นสุดการทำงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในของนักศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.5 แผนผังแสดงขั้นตอนการทำงานของโปรแกรม

การหาค่าพารามิเตอร์ของสัญญาณ

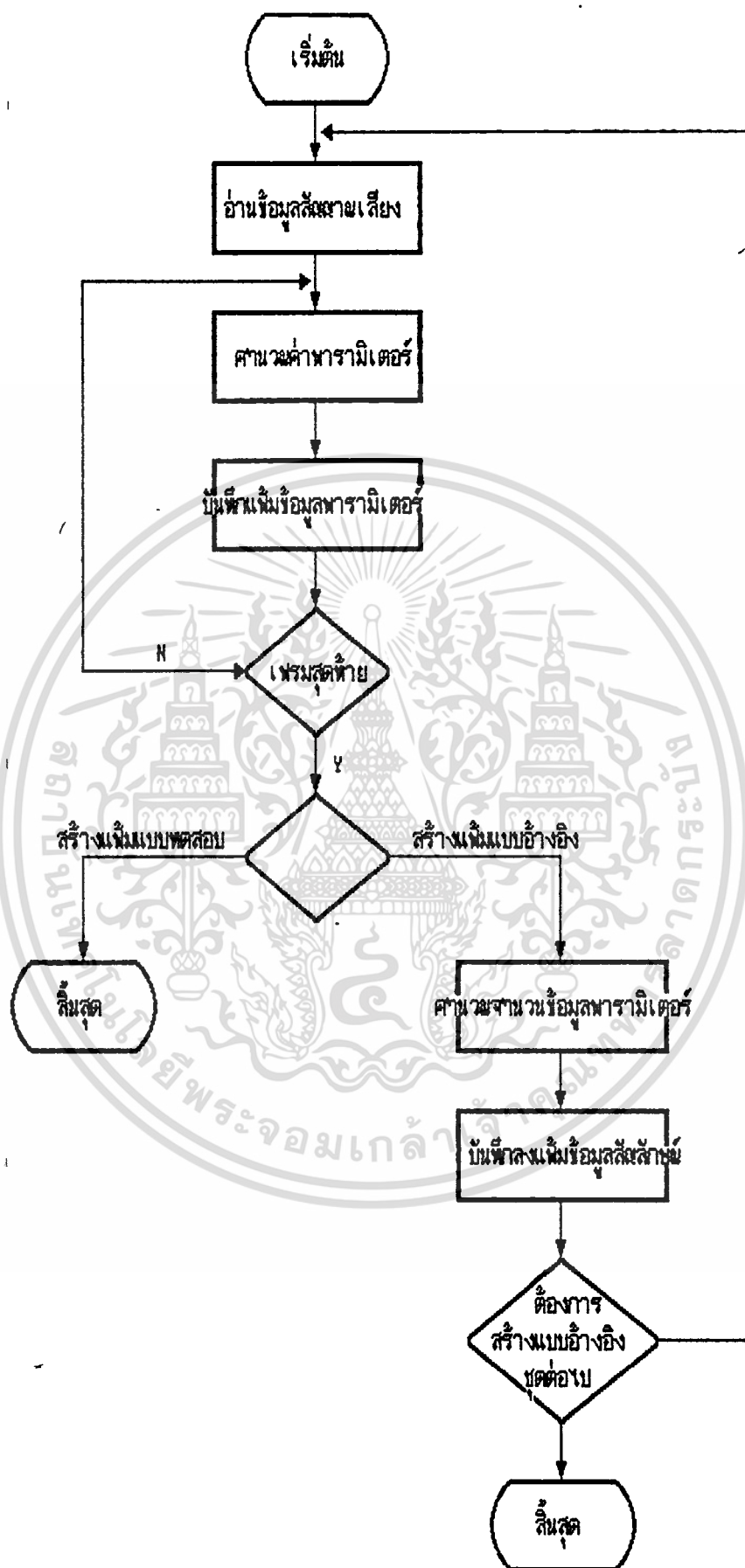
## 6.3 โปรแกรมการรับรู้เสียงพูดประกอบด้วย

### 6.3.1 โปรแกรมสร้างแฟ้มข้อมูลพารามิเตอร์

เป็นโปรแกรมที่ใช้ในการบันทึกทุกค่าพารามิเตอร์ที่หาได้โปรแกรมหาค่าพารามิเตอร์ของสัญญาณ โดยการนำค่าพารามิเตอร์ที่หาได้ในแต่ละเฟรมบันทึกต่อกันตามลำดับเฟรม จนกระทั่งเฟรมสุดท้ายของข้อมูลสัญญาณเสียง ถ้าต้องการสร้างแฟ้มข้อมูลพารามิเตอร์เพื่อนำไปใช้เป็นแบบอ้างอิง โปรแกรมจะนำค่าพารามิเตอร์ที่หาได้จากสัญญาณเสียงชุดใหม่ นำมาต่อท้ายกับข้อมูลที่มีอยู่เดิม พร้อมทั้งคำนวณจำนวนข้อมูลพารามิเตอร์ทั้งหมดของสัญญาณเสียงแต่ละค่า แล้วบันทึกลงในข้อมูลสัญลักษณ์ (Label file) เพื่อใช้ในการรับรู้ต่อไป โดยจะนำเอาจำนวนข้อมูลพารามิเตอร์ของสัญญาณในแฟ้มข้อมูลสัญลักษณ์ มาทำการคำนวณหาตำแหน่งเริ่มต้น ของข้อมูลพารามิเตอร์ของค่าแต่ละค่า ซึ่งสามารถแสดงโครงสร้างแฟ้มข้อมูลทั้งสอง คือ แฟ้มข้อมูลของพารามิเตอร์ และแฟ้มข้อมูลสัญลักษณ์ ดังรูปที่ 6.6 แต่ถ้าต้องการสร้างแฟ้มข้อมูลเพื่อนำไปใช้เป็นแบบทดสอบ โปรแกรมจะทำการบันทึกข้อมูลพารามิเตอร์ของสัญญาณเสียงที่ต้องการ เพียงสัญญาณเดียวเท่านั้น ซึ่งมีขั้นตอนการทำงานของโปรแกรมดังรูปที่ 6.7

1. ทำการอ่านข้อมูลสัญญาณเสียงที่ต้องการจะนำไปใช้เป็นแบบทดสอบหรือแบบอ้างอิง ลงในหน่วยความจำ
2. ทำการคำนวณหาค่าพารามิเตอร์ โดยใช้โปรแกรมหาค่าพารามิเตอร์ โดยทำการคำนวณทีละเฟรม และทำการบันทึกเฟรมต่อเฟรม ลงในแฟ้มข้อมูลพารามิเตอร์ตามลำดับ จนกระทั่งเฟรมสุดท้าย
3. ถ้าต้องการสร้างแฟ้มข้อมูลพารามิเตอร์ เพื่อนำไปใช้เป็นแบบทดสอบ ก็จะทำกรปิดแฟ้มข้อมูลทันที
4. ถ้าต้องการสร้างแฟ้มข้อมูลพารามิเตอร์ เพื่อนำไปใช้เป็นแบบอ้างอิง ก็จะทำกรคำนวณจำนวนข้อมูลของพารามิเตอร์ทั้งหมดของสัญญาณเสียงชุดนั้น
5. จากนั้นทำการบันทึกจำนวนพารามิเตอร์ที่คำนวณได้ข้อที่ 4 ลงในแฟ้มข้อมูลสัญลักษณ์ เพื่อนำไปใช้ในการรับรู้ต่อไป
6. ถ้าต้องการสร้างแบบอ้างอิงเพิ่ม โปรแกรมจะกลับไปทำงานตั้งแต่ข้อ 1 - 5 แต่จะข้ามข้อ 3 ไป ซึ่งการบันทึกจำนวนพารามิเตอร์ในข้อ 5 ครั้งใหม่ จะทำการบันทึกต่อจากจำนวนข้อมูลที่มีอยู่ก่อนแล้วในแฟ้มข้อมูลสัญลักษณ์นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น รูปที่ 6.7 มีแผนผังแสดงขั้นตอนการทำงานของโปรแกรมสารทุกครั้งที่มีการนำไปใช้  
การสร้างเพิ่มข้อมูลพารามิเตอร์.

ข้อมูลพารามิเตอร์ของสัญญาเสียงแต่ละชุด

|                               | ลำดับ                | ข้อมูล            |             |
|-------------------------------|----------------------|-------------------|-------------|
|                               | 1                    | RC(1)             |             |
|                               | 2                    | RC(2)             | ค่าที่ 1    |
|                               | ⋮                    | ⋮                 | เฟรมที่ 1   |
|                               | P                    | RC(P)             |             |
|                               | ⋮                    | ⋮                 | ค่าที่ 1    |
|                               | ⋮                    | ⋮                 | เฟรมที่ 2   |
|                               | ⋮                    | ⋮                 | ค่าที่ 1    |
|                               | ⋮                    | ⋮                 | เฟรมสุดท้าย |
| <u>เพิ่มข้อมูลพารามิเตอร์</u> |                      |                   |             |
|                               | ตำแหน่งในเพิ่มข้อมูล | ข้อมูล            |             |
|                               | 0                    | ข้อมูลพารามิเตอร์ |             |
| L1                            | AAA                  |                   | ค่าที่ 1    |
|                               | ⋮                    | ข้อมูลพารามิเตอร์ |             |
| L2                            | BBB                  |                   | ค่าที่ 2    |
|                               | ⋮                    | ข้อมูลพารามิเตอร์ |             |
| LN                            |                      |                   | ค่าที่ N    |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## แฟ้มข้อมูลสัญลักษณ์

| หมายเลขสัญลักษณ์ | จำนวนข้อมูลพารามิเตอร์ |
|------------------|------------------------|
| 1                | L1                     |
| 2                | L2                     |
| ⋮                | ⋮                      |
| ⋮                | ⋮                      |

รูปที่ 6.6 แสดงโครงสร้างของแฟ้มข้อมูลพารามิเตอร์และแฟ้มข้อมูลสัญลักษณ์

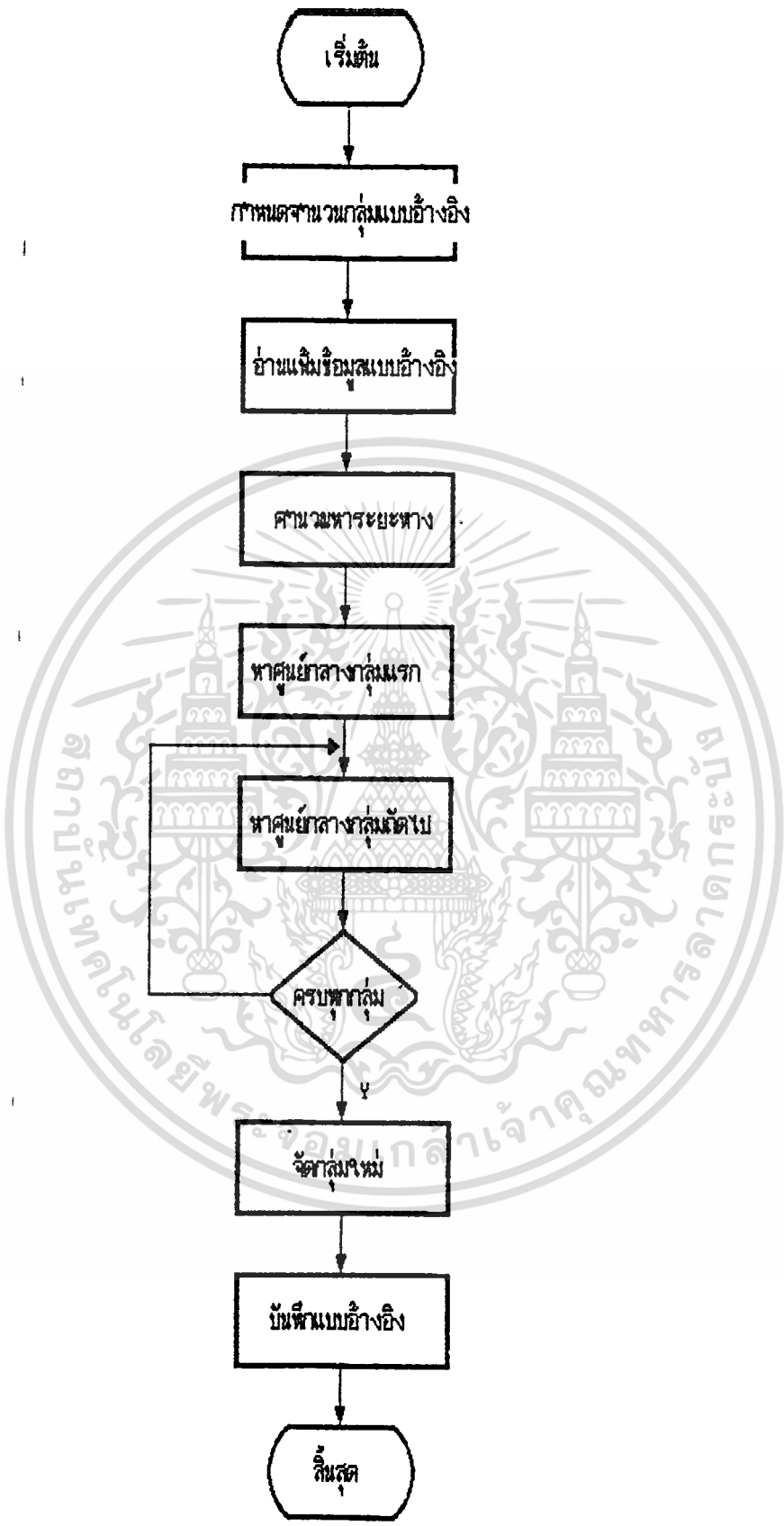
### 6.3.2 โปรแกรมคัดเลือกแบบอ้างอิง

เป็นโปรแกรมที่ใช้ในการคัดเลือกแบบอ้างอิงจากแฟ้มข้อมูลแบบอ้างอิงที่สร้างโดยโปรแกรมสร้างแฟ้มข้อมูลพารามิเตอร์ ซึ่งแฟ้มข้อมูลแบบอ้างอิงนี้จะถูกแบ่งออกเป็นจำนวน 10 แฟ้ม ได้แก่ แฟ้มข้อมูลของเสียงศูนย์ ถึง แฟ้มข้อมูลของเสียงเก้า ในแต่ละแฟ้มข้อมูลจะประกอบไปด้วยข้อมูลของเสียงที่เกิดจากการพูดหลายครั้ง โดยโปรแกรมนี้จะทำการคัดเลือกแบบอ้างอิงที่ดีที่สุด เพื่อนำไปใช้ในโปรแกรมการรับรู้ต่อไป ซึ่งมีขั้นตอนการทำงานของโปรแกรมดังรูปที่ 6.8

1. กำหนดค่าจำนวนแบบอ้างอิงที่ต้องการในแต่ละคำให้มีค่าเท่ากับ  $K$  กลุ่ม
2. ทำการอ่านแบบอ้างอิงในแฟ้มข้อมูลแบบอ้างอิงทั้งหมดของแต่ละเสียงลงบนหน่วยความจำ
3. โปรแกรมจะทำการหาค่าระยะทาง ระหว่างแบบอ้างอิงทุกแบบที่มีอยู่ในแฟ้มข้อมูลแบบอ้างอิง
4. ทำการหาจุดศูนย์กลางของกลุ่มแรก โดยใช้แบบอ้างอิงที่มีความยาวมากที่สุด เป็นจุดศูนย์กลางของกลุ่มแรก
5. ทำการหาจุดศูนย์กลางของกลุ่มถัดไปจาก แบบอ้างอิงที่มีระยะห่างจากจุดศูนย์กลางมากที่สุด
6. จุดศูนย์กลางของกลุ่มถัดไปจะหาได้จาก แบบอ้างอิงที่มีระยะห่างจากจุดศูนย์กลางของกลุ่มทั้งสองมากที่สุด ทำซ้ำจนได้กลุ่มแบบอ้างอิงครบจำนวน  $K$  กลุ่ม
7. ทำการจัดกลุ่มใหม่ อธิบายในหัวข้อ 3.1.2

8. เมื่อได้แบบอ้างอิงตามกำหนด จะทำการบันทึกลงแฟ้มข้อมูล เพื่อนำไปใช้ในการรับรู้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่สามารถนำข้อมูลไปใช้ในเชิงพาณิชย์ได้  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



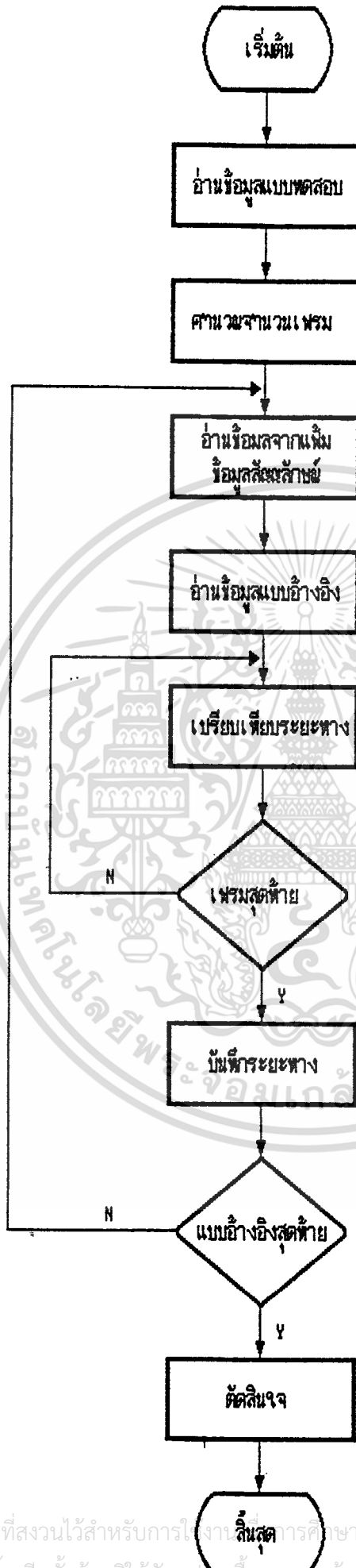
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 รูปที่ 6-8 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมการคัดเลือกแบบอ้างอิง  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6.3.3 โปรแกรมการรับรู้

เป็นโปรแกรมที่ใช้ในการรับรู้ โดยจะทำการเปรียบเทียบข้อมูลพารามิเตอร์ของเสียงที่เป็นแบบทดสอบ กับข้อมูลพารามิเตอร์ของเสียงที่เป็นแบบอ้างอิงทีละคำ โดยอาศัยการเปรียบเทียบแบบไดนามิกโปรแกรมมิ่ง (รายละเอียดอยู่ในหัวข้อ 4.2) ซึ่งผลลัพธ์ที่ได้จากการคำนวณโดยใช้โปรแกรมนี้จะแสดงเป็นระยะทางระหว่างแบบทดสอบกับแบบอ้างอิงแต่ละคำ จากระยะทางที่ได้นำไปผ่านกฎการตัดสินใจเพื่อเลือกค่าที่ถูกต้องที่สุด โดยสามารถแสดงขั้นตอนการทำงานของโปรแกรมการรับรู้ ดังรูปที่

6.9

1. อ่านค่าพารามิเตอร์จากแฟ้มข้อมูลแบบทดสอบ ลงในหน่วยความจำ
2. ทำการคำนวณหาจำนวนแฟ้มทั้งหมดจากแฟ้มข้อมูลแบบทดสอบ เพื่อที่จะนำไปใช้ในการเปรียบเทียบระยะทางต่อไป
3. ทำการอ่านข้อมูลจากแฟ้มข้อมูลสัญลักษณ์ เพื่อใช้ในการกำหนดจุดเริ่มต้นและจุดสิ้นสุดของข้อมูลแบบอ้างอิงในแต่ละคำ
4. เมื่อทำการหาจุดเริ่มต้นและจุดสิ้นสุดของข้อมูลแบบอ้างอิงในแต่ละคำเรียบร้อยแล้ว ก็ทำการอ่านข้อมูลของแบบอ้างอิงของคำๆนั้นลงในหน่วยความจำ
5. ทำการเรียกโปรแกรมย่อยไดนามิก (รูปที่ 4.2.3) เพื่อที่จะนำเอาพารามิเตอร์ของข้อมูลแบบทดสอบและพารามิเตอร์ของข้อมูลแบบอ้างอิงมาทำการเปรียบเทียบกัน
6. โปรแกรมจะกลับไปทำงานในหัวข้อ 5 อีกครั้งจนกระทั่งทำการเปรียบเทียบกันครบทุกแฟ้ม
7. ทำการบันทึกค่าระยะทางที่คำนวณได้จากการเปรียบเทียบลงในหน่วยความจำ
8. โปรแกรมจะกลับไปทำงานตั้งแต่ข้อ 3 ถึง 7 จนถึงแบบอ้างอิงสุดท้ายที่ปรากฏอยู่ในแฟ้มข้อมูลอ้างอิง
9. ใช้กฎการตัดสินใจเพื่อคัดเลือกผลลัพธ์ที่ถูกต้อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.9 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมการรับรู้

บทที่ ๗

การทดลองและผลการทดลอง

7.1 การทดลองและผลการทดลอง

ในการทดลองนี้ ได้ทำการเก็บเสียงตัวเลข 0-9 โดยเก็บค่าละ 5 ครั้ง เพื่อนำไปสร้างแบบอ้างอิง ในส่วนของข้อมูลที่ใช้ในการทดสอบคือ จะให้บุคคลเดีวกันนั้นทำการพูดเสียงตัวเลข 0-9 ใหม่ อีกค่าละ 15 ครั้ง เมื่อได้ข้อมูลที่เป็นแบบทดสอบและแบบอ้างอิงเรียบร้อยแล้ว ก็จะนำแบบทดสอบที่ได้ทั้งหมดไปทำการเปรียบเทียบกับแบบอ้างอิง โดยใช้เทคนิคไดนามิคโปรแกรมมิ่งเพื่อทดสอบค่าที่ถูกต้องต่อไป

ในการทดลองนี้แบ่งได้เป็น 2 ส่วนคือ

**ส่วนที่ 1** ทำการทดสอบแบบอ้างอิงที่จะนำไปใช้แบบอ้างอิงเป็นแบบอ้างอิงในโปรแกรมการรับรู้ โดยกำหนดให้มี 1 แบบอ้างอิงในแต่ละค่า โดยแบบอ้างอิงนี้จะทำการคัดเลือกจากแบบอ้างอิงที่ได้บันทึกไว้ในตอนต้น โดยผ่านกระบวนการคัดเลือกแบบอ้างอิง หลังจากนั้นก็นำค่าที่เป็นข้อมูลแบบทดสอบทั้งหมดมาผ่านารรับรู้ และใช้กฎการตัดสินใจแบบ Nearest Neighbour ซึ่งจะได้อัตราความถูกต้องในการรับรู้เสียงพูดของตัวเลข 0-9 เฉลี่ยเป็นร้อยละ 79.3 แสดงดังรูปที่ 7.1

**ส่วนที่ 2** ทำการกำหนดจำนวนแบบอ้างอิงที่จะนำไปใช้เป็นแบบอ้างอิงในโปรแกรมการรับรู้ โดยกำหนดให้มีจำนวน 2 แบบในแต่ละค่า โดยแบบอ้างอิงที่จะนำมาใช้นี้ จะทำการคัดเลือกแบบอ้างอิงที่ได้บันทึกไว้ข้างต้น โดยผ่านกระบวนการคัดเลือกแบบอ้างอิง หลังจากนั้นก็จะนำข้อมูลที่เป็นแบบทดสอบทั้งหมดมาผ่านโปรแกรมการรับรู้และใช้กฎการตัดสินใจแบบ K-Nearest Neighbour (KNN) แบบที่ 2 ซึ่งจะได้อัตราความถูกต้องในการรับรู้เสียงพูดของตัวเลข เฉลี่ยเป็นร้อยละ 82.5 แสดงดังรูป 7.2

## แบบอ้างอิง

แบบทดสอบ

|   | 1  | 2  | 3  | 4  | 5  | 6  | 7 | 8  | 9  | 0  | %    |
|---|----|----|----|----|----|----|---|----|----|----|------|
| 1 | 13 | -  | -  | -  | 2  | -  | - | -  | -  | -  | 86.6 |
| 2 | -  | 10 | 3  | -  | -  | -  | - | -  | -  | 2  | 66.6 |
| 3 | -  | -  | 12 | 2  | -  | -  | - | -  | -  | 1  | 80.0 |
| 4 | -  | 1  | 1  | 12 | -  | -  | - | -  | -  | 1  | 80.0 |
| 5 | -  | -  | 1  | -  | 14 | -  | - | -  | -  | -  | 93.3 |
| 6 | 5  | -  | -  | -  | -  | 10 | - | -  | -  | -  | 66.6 |
| 7 | 1  | -  | -  | 2  | -  | -  | 9 | 3  | -  | -  | 60.0 |
| 8 | -  | -  | 1  | -  | -  | -  | - | 14 | -  | -  | 93.3 |
| 9 | -  | -  | 1  | -  | -  | -  | - | -  | 12 | 2  | 80.0 |
| 0 | -  | -  | 2  | -  | -  | -  | - | -  | -  | 13 | 86.6 |

รูปที่ 7.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## แบบอ้างอิง

แบบทดสอบ

|   | 1  | 2 | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 0  | %    |
|---|----|---|----|----|----|----|----|----|----|----|------|
| 1 | 13 | - | -  | -  | 2  | -  | -  | -  | -  | -  | 86.6 |
| 2 | -  | 9 | 6  | -  | -  | -  | -  | -  | -  | -  | 60.0 |
| 3 | -  | - | 15 | -  | -  | -  | -  | -  | -  | -  | 100  |
| 4 | -  | - | 1  | 14 | -  | -  | -  | -  | -  | -  | 93.3 |
| 5 | -  | - | -  | -  | 15 | -  | -  | -  | -  | -  | 100  |
| 6 | -  | - | -  | -  | -  | 15 | -  | -  | -  | -  | 100  |
| 7 | 2  | - | -  | 1  | -  | -  | 12 | -  | -  | -  | 80.0 |
| 8 | -  | - | 4  | -  | 1  | -  | -  | 10 | -  | -  | 66.6 |
| 9 | -  | - | 3  | -  | 1  | -  | -  | -  | 11 | -  | 73.3 |
| 0 | -  | - | 5  | -  | -  | -  | -  | -  | -  | 10 | 66.6 |

รูปที่ 7.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ ๘

### สรุปผลการทดลองและข้อเสนอแนะ

#### 8.1 สรุปผลการทดลอง

ระบบการรับรู้เสียงพูดในผลงานชิ้นนี้ ได้ใช้เสียงของตัวเลข 0-9 ในการทดสอบโดยใช้เสียงของบุคคลเดียวในการสร้างแบบอ้างอิงและแบบทดสอบ จากการทดลองที่ได้แบ่งเป็น 2 การทดลองคือการทดลองที่ใช้แบบอ้างอิง 1 แบบในแต่ละค่าในการเปรียบเทียบกับแบบทดสอบ ผลที่ได้ปรากฏว่า อัตราความถูกต้องของแต่ละค่าแสดงดังรูป 7.1 และการทดลองที่ใช้แบบอ้างอิง 2 แบบในแต่ละค่าในการเปรียบเทียบกับแบบทดสอบ ผลที่ได้แสดงดังรูป 7.2

เมื่อเปรียบเทียบผลการทดลองทั้งสองอันแล้วพบว่าอัตราความถูกต้องในการตรวจรู้ของเสียงตัวเลข 0-9 ของการทดลองที่ 2 จะมีความถูกต้องมากกว่าการทดลองที่ 1 เนื่องจากการทดลองที่ 2 ได้ใช้กฎการตัดสินใจแบบ K-Nearest Neighbour โดยค้นหาแบบอ้างอิงที่มีระยะทางน้อยที่สุดของแต่ละค่า จำนวน k แบบ (การทดลองที่ 2 ใช้ 2 แบบและนำค่าเฉลี่ยของระยะทางนำมาใช้ในการตัดสินใจ ซึ่งจะทำให้อัตราความถูกต้องในการตรวจรู้สูงกว่าการใช้ nearest response เช่นในการทดลองที่ 1

#### 8.2 ข้อเสนอแนะ

1. เรื่องความเร็วของระบบการทำงาน เนื่องจากผลงานชิ้นนี้ ใช้ไมโครคอมพิวเตอร์และใช้ซอฟต์แวร์ในการหาค่าพารามิเตอร์ในวิธีการทำนายเชิงเส้น การตรวจรู้โดยใช้เทคนิคไดนามิกโปรแกรมมิ่งและกฎการตัดสินใจแบบ K-Nearest Neighbour (KNN) ซึ่งต้องมีการคำนวณเป็นจำนวนมาก ทำให้เกิดปัญหาด้านความเร็วซึ่งต้องใช้เวลาค่อนข้างมากกว่าจะได้ผลออกมา
2. ในการบันทึกเสียงที่จะนำไปใช้เป็นแบบอ้างอิงหรือแบบทดสอบ ควรเลือกสถานที่ที่เหมาะสม กล่าวคือควรเป็นสถานที่เงียบ ไม่มีเสียงรบกวน เพื่อให้ได้เสียงที่มีความเป็นจริงมากที่สุด ซึ่งจะส่งผลให้อัตราความถูกต้องของการตรวจรู้มากขึ้นด้วย
3. ถ้าต้องการเพิ่มอัตราความถูกต้องของแต่ละตัวเลข สามารถทำได้โดยการใช้กฎการตัดสินใจแบบ K-Nearest Neighbour (KNN) โดยเพิ่มจำนวน k แบบให้มีจำนวนมากขึ้น แต่มีข้อเสียคือ จะใช้เวลาในการคำนวณมากตามไปด้วย

## เอกสารอ้างอิง

- [1] ไพศาล ธรรมไพทอง. "ระบบการรับรู้เสียงพูดแบบต่างบุคคล." วิทยานิพนธ์ปริญญาโทมหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2533.
- [2] ทวี ประทุมทอง. "การตรวจรู้เสียงพูดภาษาไทยโดยใช้หน่วยพยางค์." วิทยานิพนธ์ปริญญาโทมหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2530.
- [3] กฤษดา เรืองเส. "ต้นแบบเครื่องสังเคราะห์เสียงพูดด้วยวิธีเข้ารหัสแบบลิเนียร์พรีดิคทีฟ." วิทยานิพนธ์ปริญญาโทมหาบัณฑิต ภาควิชาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2530.
- [4] Lawrence R. Rabiner and Ronald W. Schafer. "Digital Processing of Speech Signals." Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1978.
- [5] Thomas W. Parsons. "Voice and Speech Processing." McGraw-Hill Book Company, New York, 1987.
- [6] J.D. Markel and A.H. Grey, Jr. "Linear Prediction of Speech." Springer-Verlag, Berlin Heidelberg, New York, 1976.
- [7] Shozo Saito and Kazuo Nakata. "Fundamentals of Speech Signal Processing." ACADEMIC PRESS, INC. New York, 1985.
- [8] Panos E. Papamichalis. "Practical Approaches to Speech Coding." Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 1987.
- [9] Lawrence Rabiner and Bing-Hwang Juang. "Fundamentals of Speech Recognition." PTR Prentice Hall Englewood Cliffs, New Jersey, 1993.
- [10] W.A. Ainsworth. "Speech Recognition by Machine." Billing and Sons Ltd. London, 1988.
- [11] H. Sakoe and S. Chiba. "Dynamic Programming Algorithm Optimization Spoken Word Recognition." IEEE TRANSACTIONS ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING, VOL, ASSP-26, NO.1, FEBRUARY 1978.
- [12] LORI F. LAMEL, LAWRENCE R. RABINER, AARON E. ROSENBERG AND JAY G. WILPON. "An Improved Endpoint Detector for Isolated Word Recognition" IEEE TRANSACTIONS ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [13] LAWRENCE R. RABINER, STEPHEN E. LEVINSON, AARON E. ROSENBERG, AND JAY G. WILPON. "Speaker-Independent Recognition of Isolated Word Using Clustering Techinics." IEEE TRANSACTION ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, VOL, ASSP-17, NO.4, AUGUST 1979.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### กิติกรรมประกาศ

ปริญญาโทฉบับนี้ สามารถสำเร็จลงไปได้ดีด้วยคำแนะนำทางด้านความรู้ และชี้แนวทางในการศึกษาจากอาจารย์ที่ปรึกษา จึงขอกราบขอบพระคุณอาจารย์ เกียรติเกรียงไกร วงศ์โรจน์ภรณ์ ไร่ ณ. ที่นี้ด้วย

ขอขอบพระคุณ อาจารย์ ดร. สุวิมล สิทธิชีวภาค, อาจารย์ ปราโมทย์ วาดเขื่อน, อาจารย์ ดร. อภิชาติ ตั้งทางธรรม และอาจารย์ภาคโทรคมนาคมและภาคอื่นๆทุกท่าน ที่ได้ให้คำปรึกษาและให้เครื่องมือ อุปกรณ์ต่างๆและห้องในการวิจัย

นอกจากนี้การวิจัยครั้งนี้คงไม่สามารถสำเร็จลงไปได้ถ้าขาด เจียร ทุม กฤษณ์และเพื่อนๆ ที่กรุณาให้ใช้เสียงและให้คำที่ปรึกษาในการวิจัย ทางผู้จัดทำจึงขอขอบคุณไว้ ณ. ที่นี้ด้วย

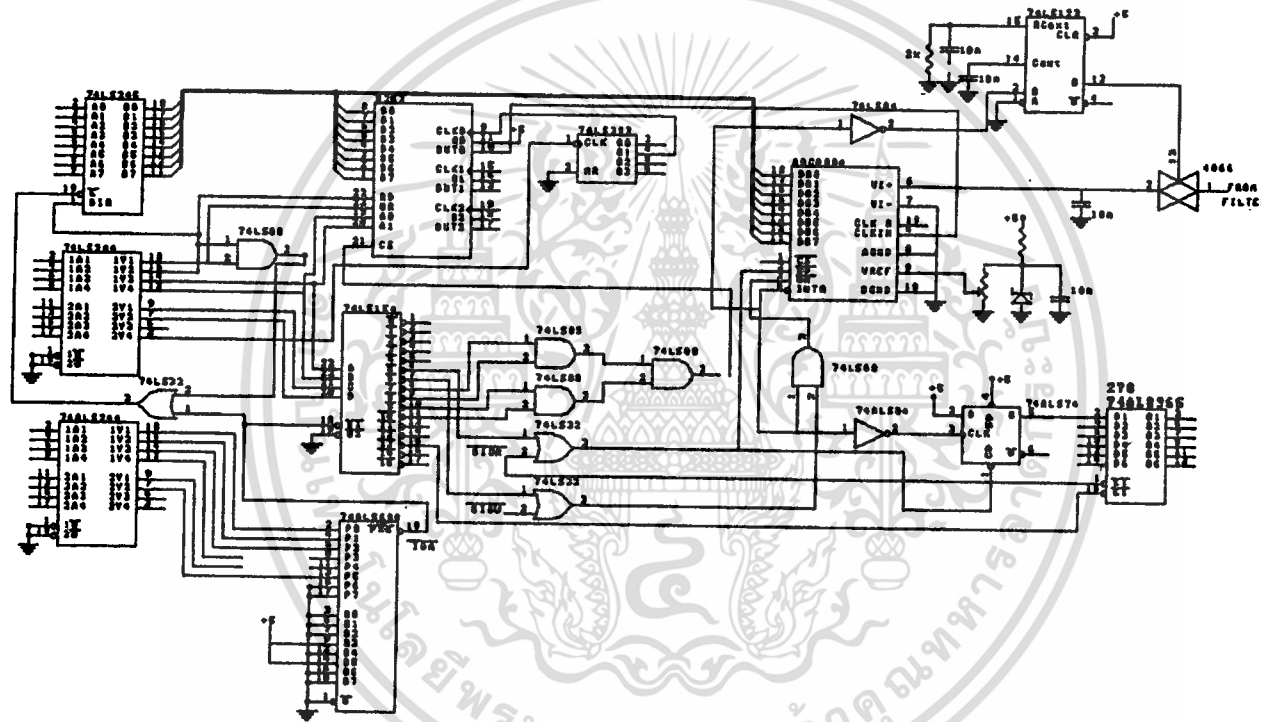


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

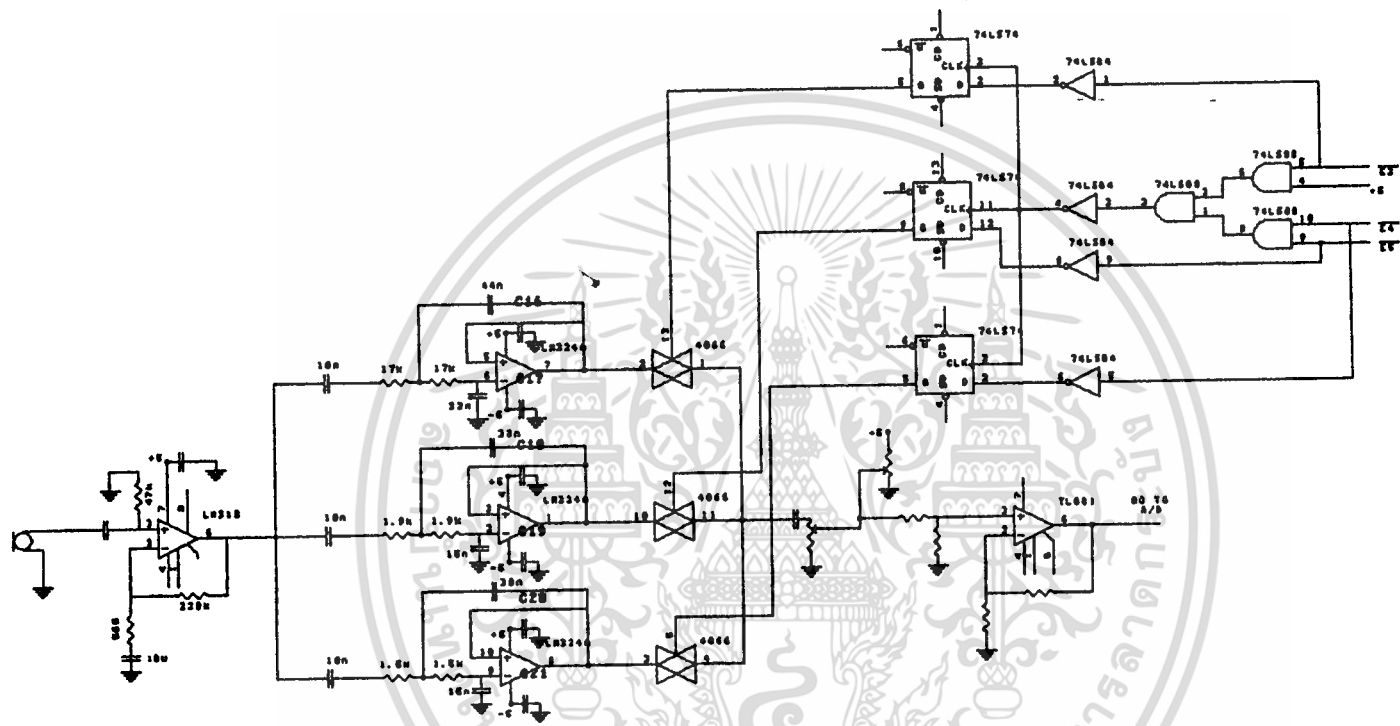


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

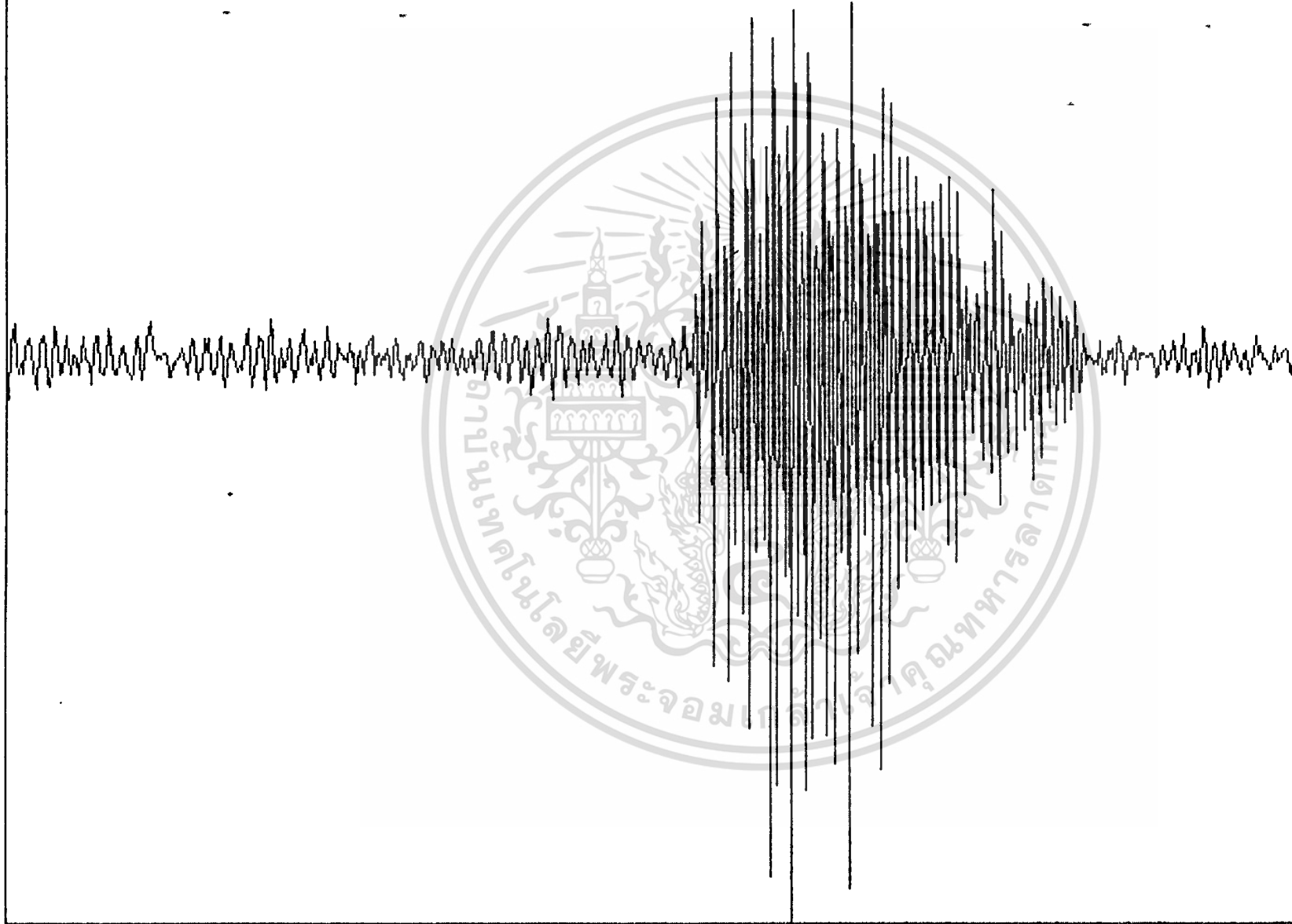
ภาคผนวก ก.



รูปที่ ก.1 แสดงวงจรรวมของภาคแปลงสัญญาณ

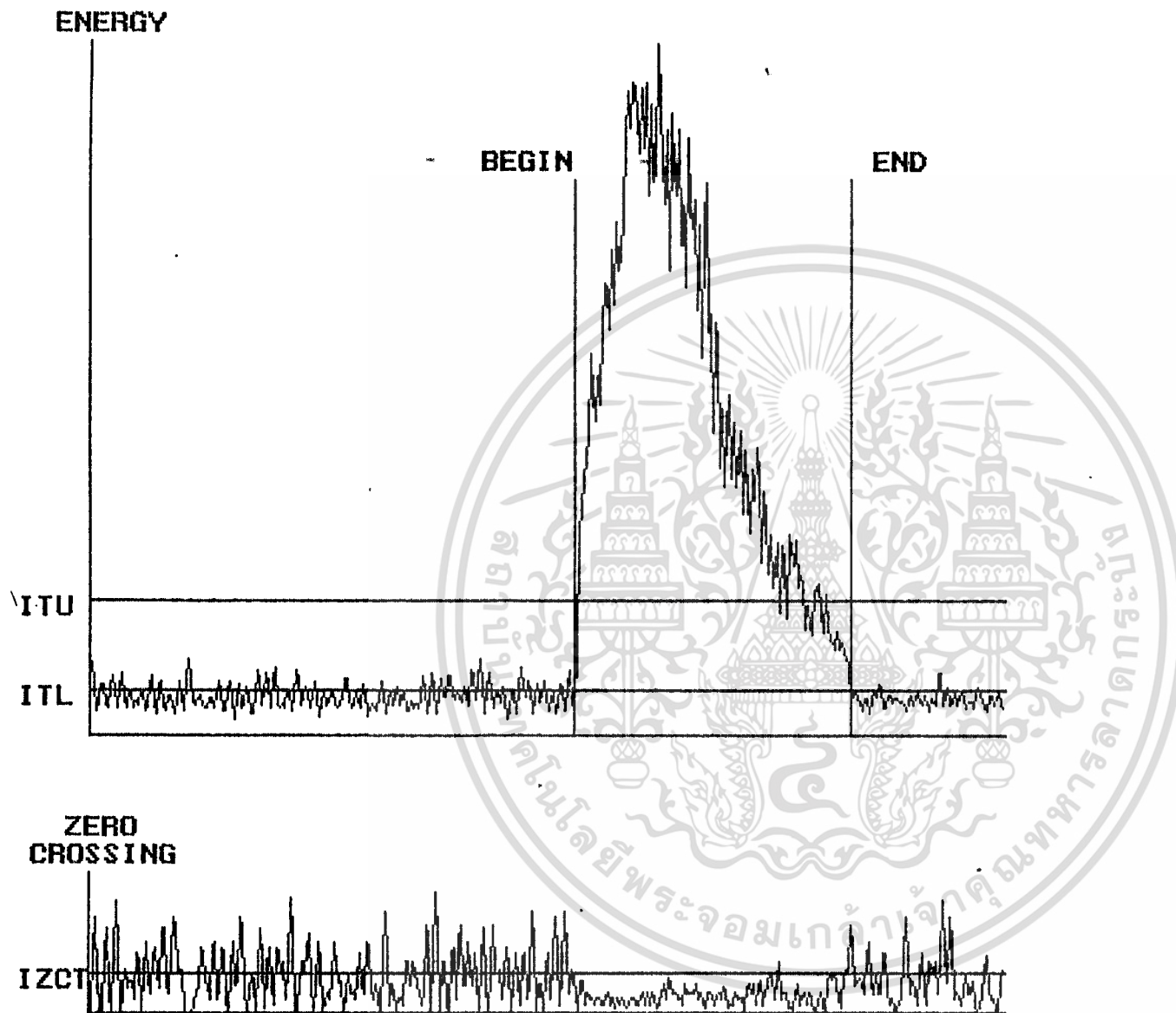


รูปที่ ก.2 แสดงวงจรรวมภาคอนุถก



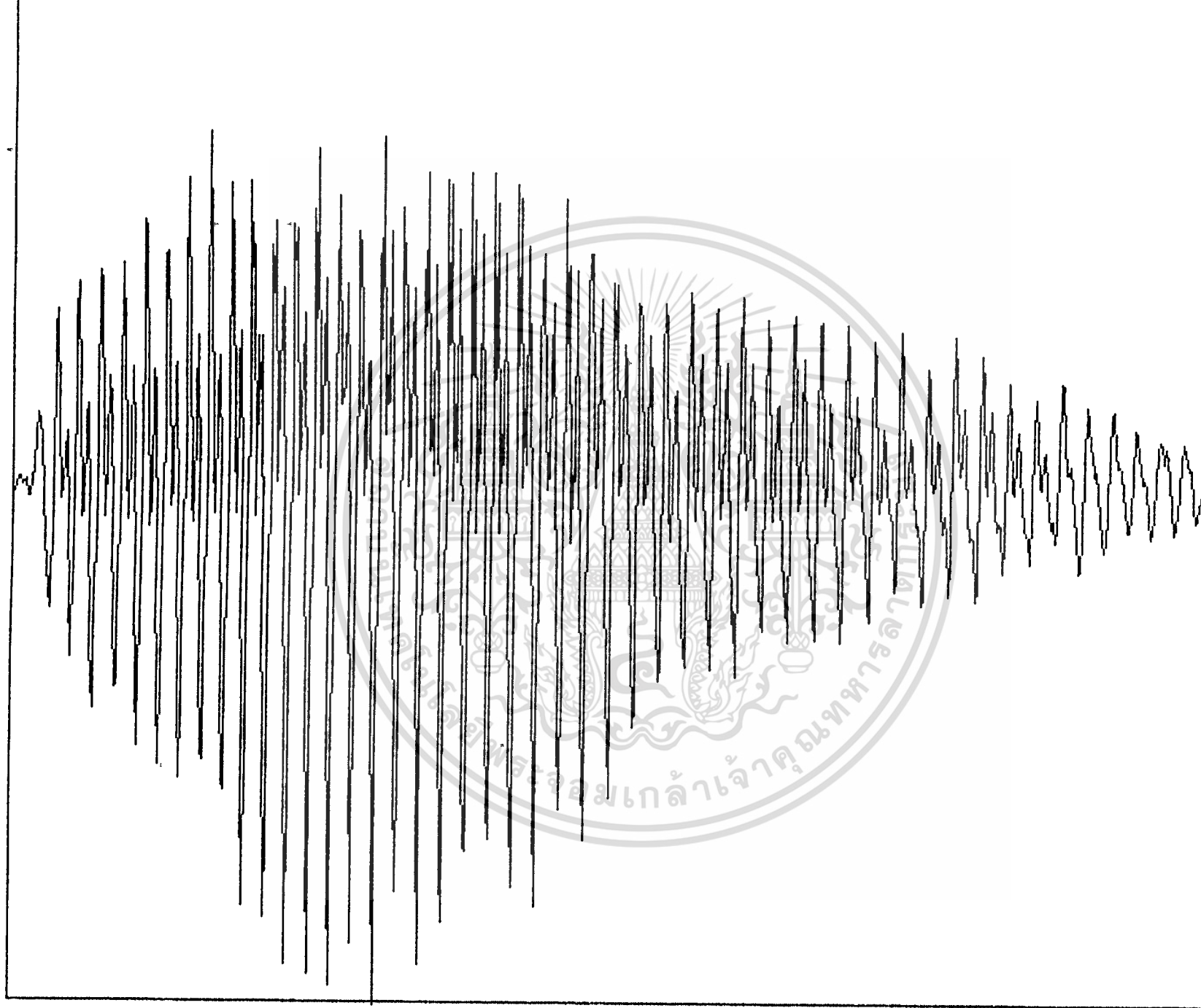
รูปที่ ข.1 แสดงภาพสัญญาณเสียง 1

PRESS ANY KEY TO CONTINUE



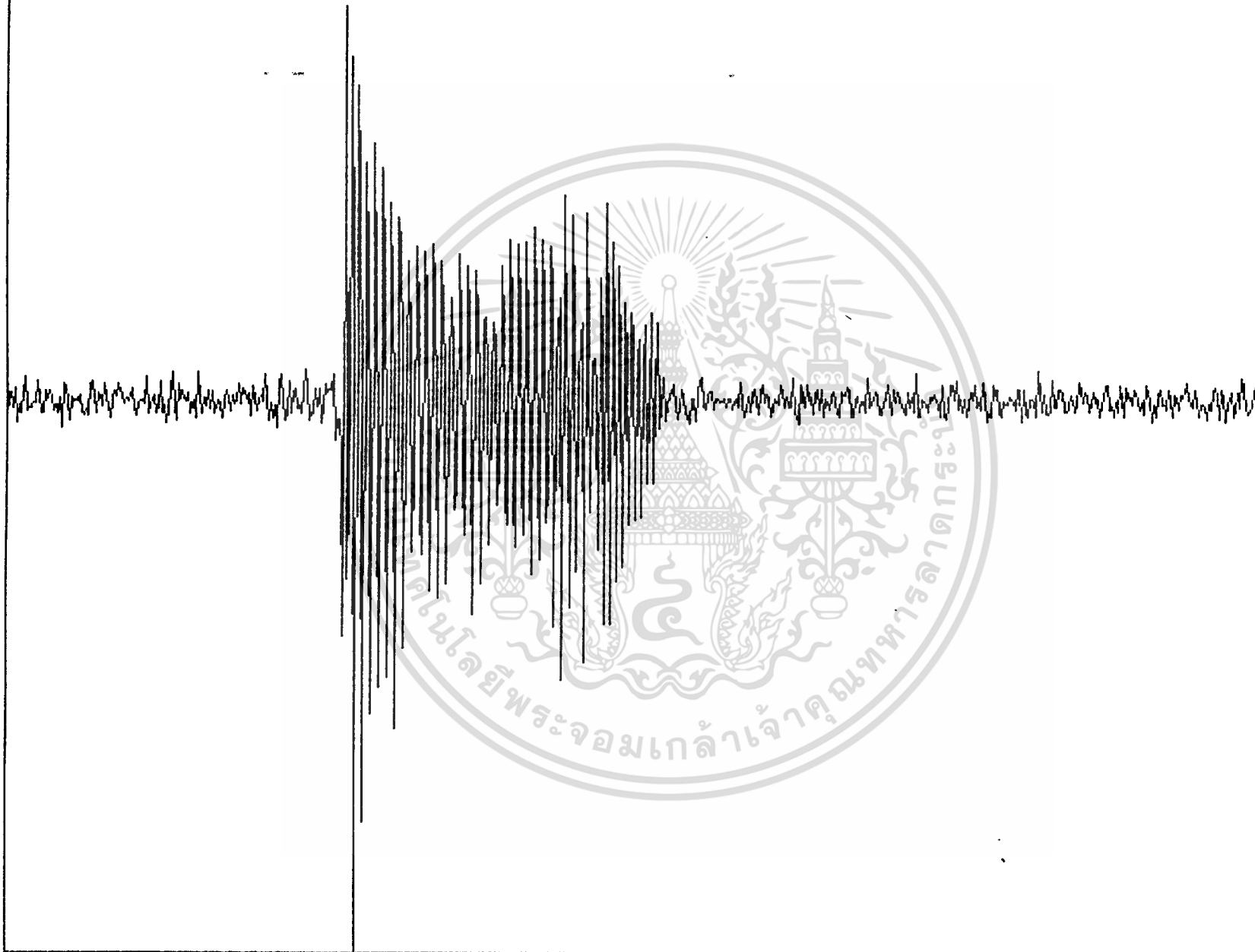
PRESS ANY KEY SEE TIME-DOMAIN

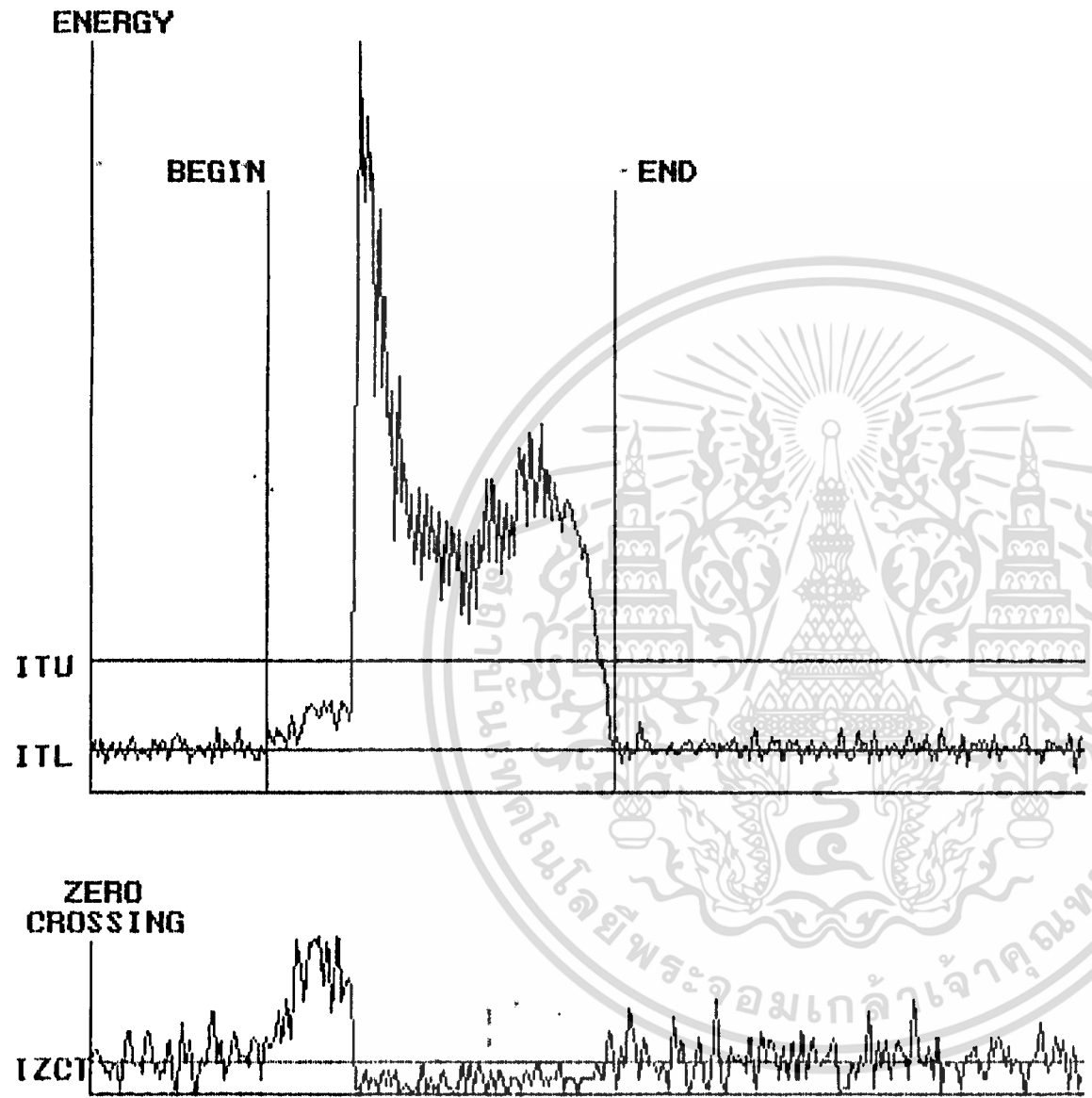
รูปที่ ข.2 แสดงภาพการหาขอบเขตของเสียง 1



รูปที่ ข.3 แสดงภาพสัญญาณเสียง 1 เมื่อได้รับการตัดคำแล้ว

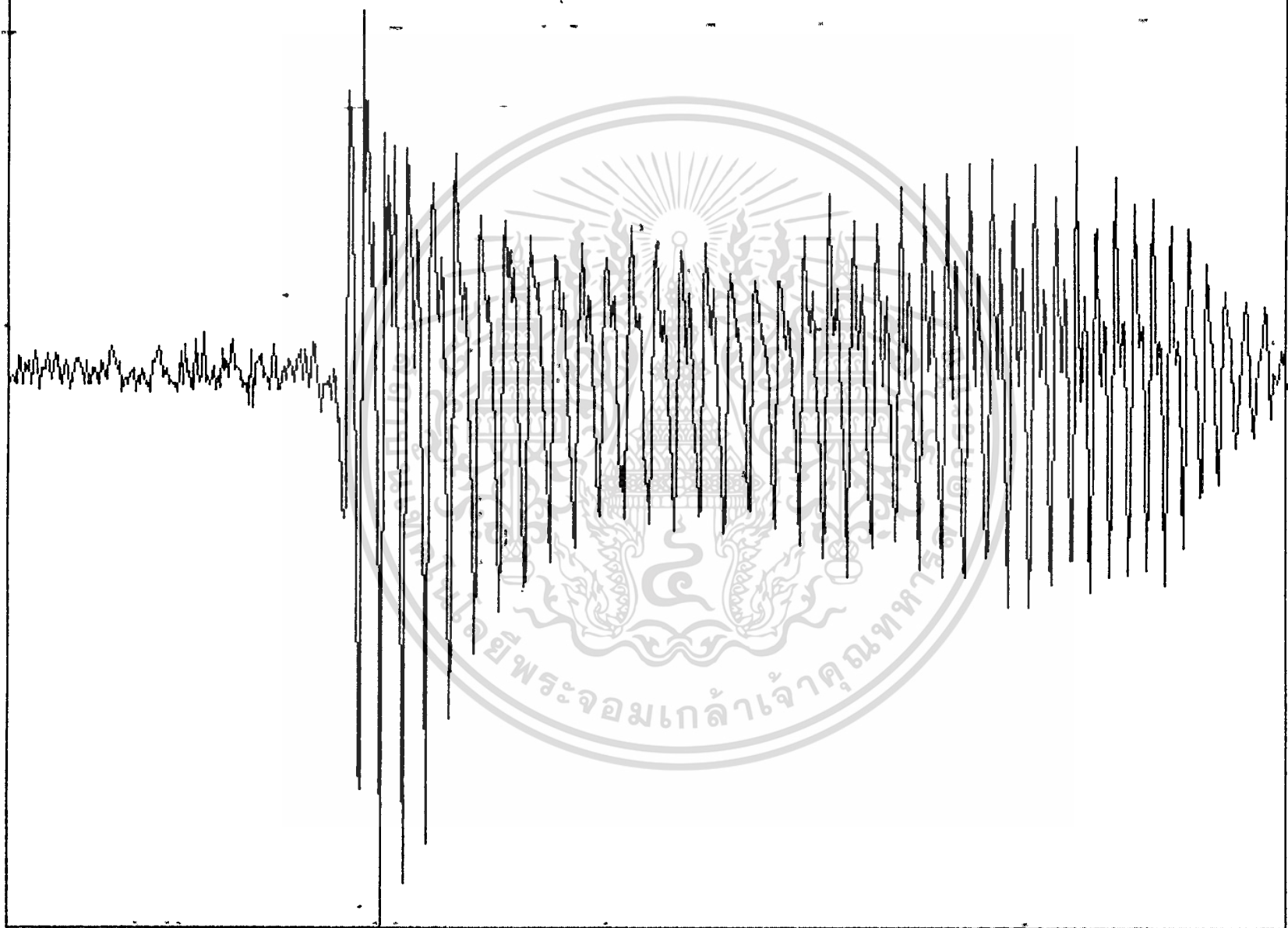
PRESS ANY KEY TO CONTINUE





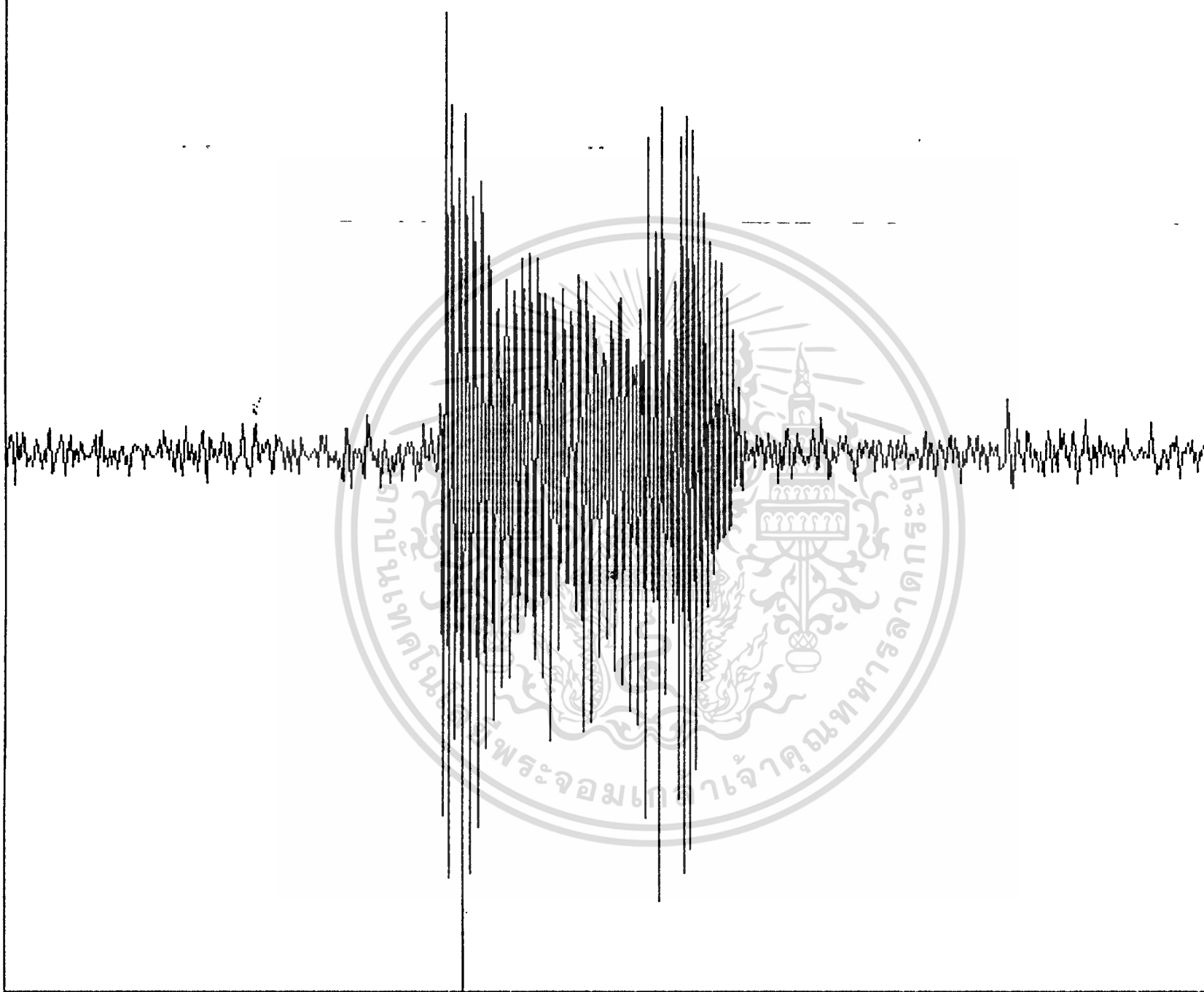
รูปที่ ข.5 แสดงภาพการหาขอบเขตของเสียง 2

PRESS ANY KEY SEE TIME-DOMAIN



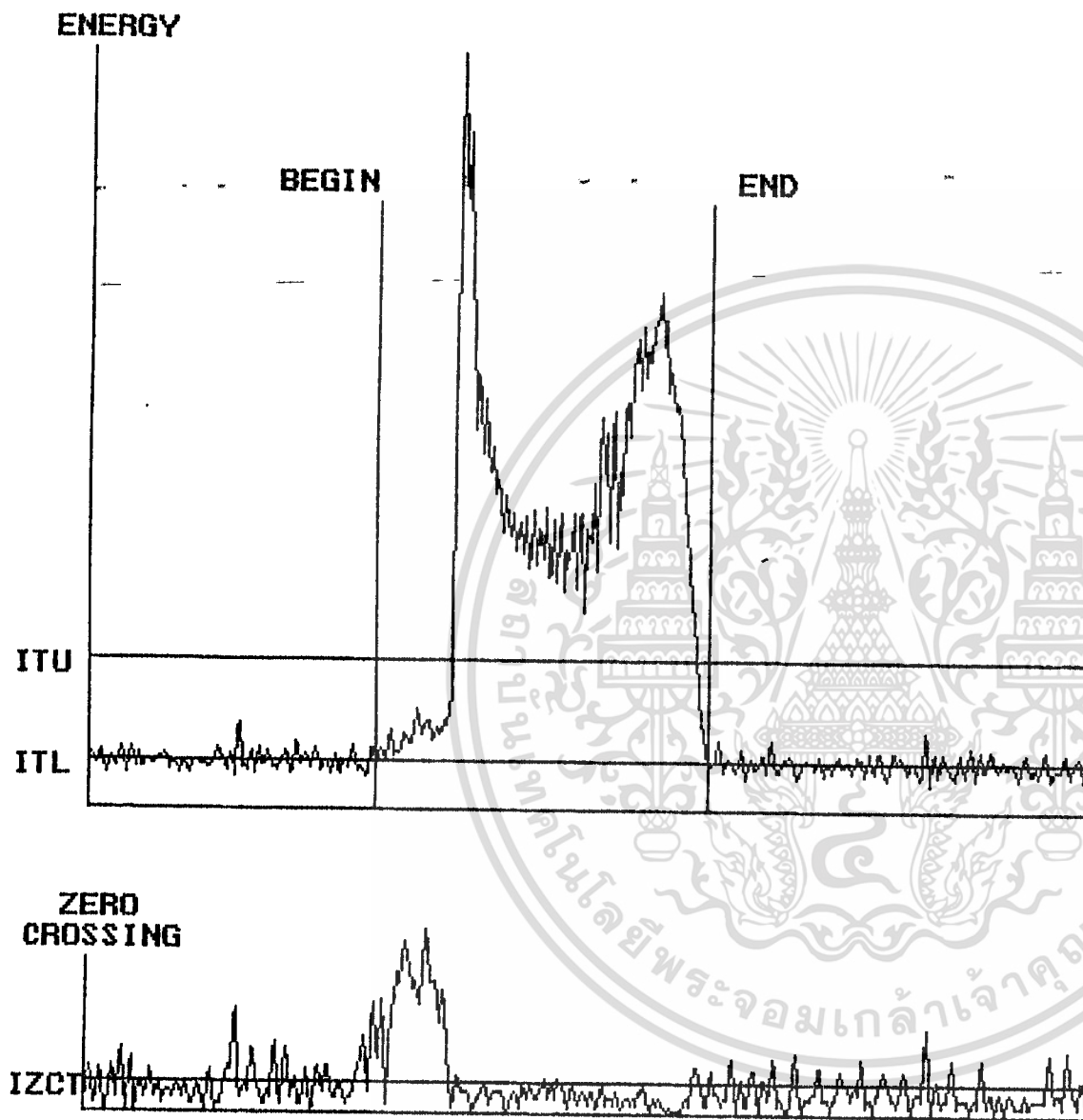
รูปที่ ข.6 แสดงภาพสัญญาณเสียง 2 เมื่อได้รับการตัดคำแล้ว

PRESS ANY KEY TO CONTINUE



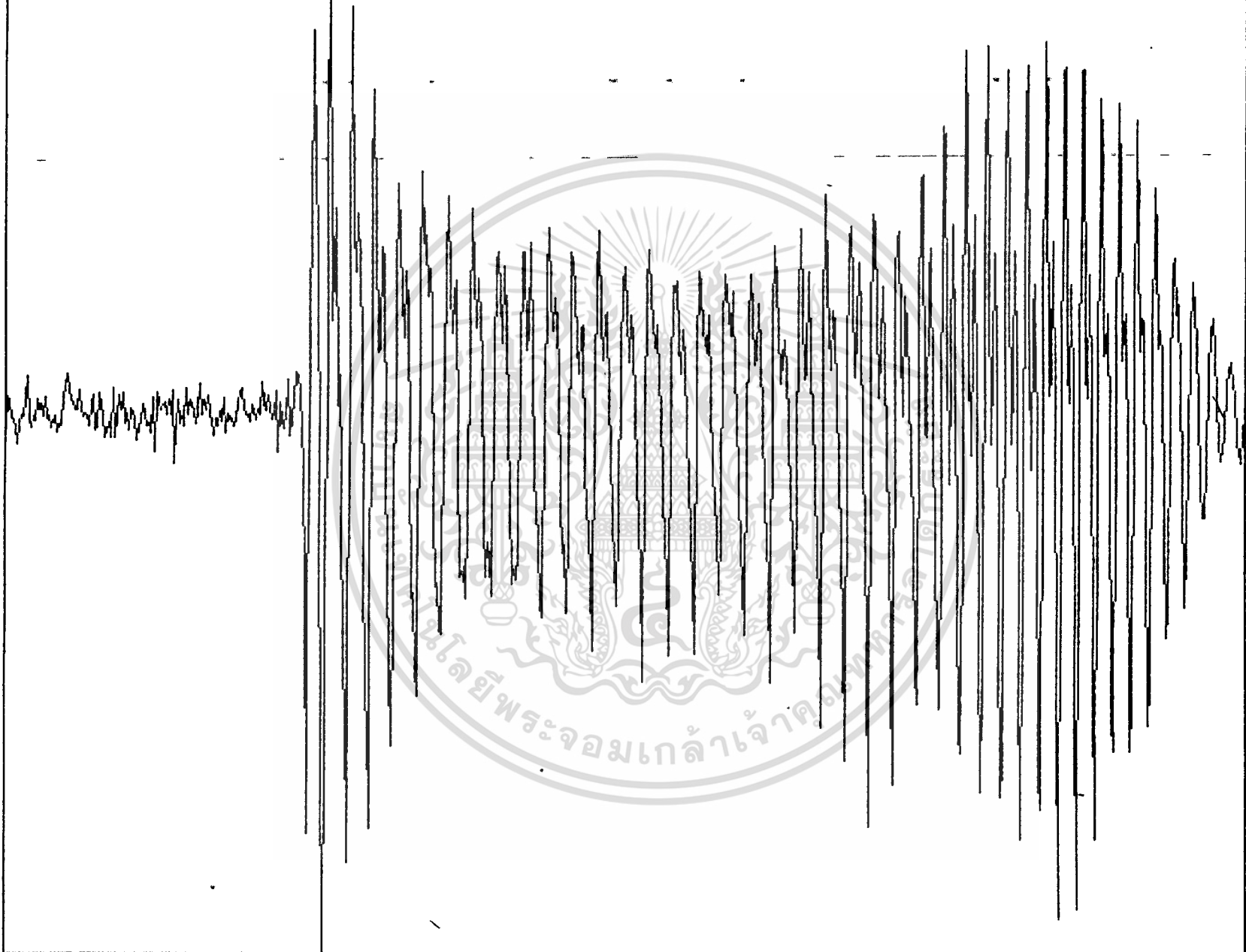
รูปที่ ข.7 แสดงภาพสัญญาณเสียง 3

PRESS ANY KEY TO CONTINUE



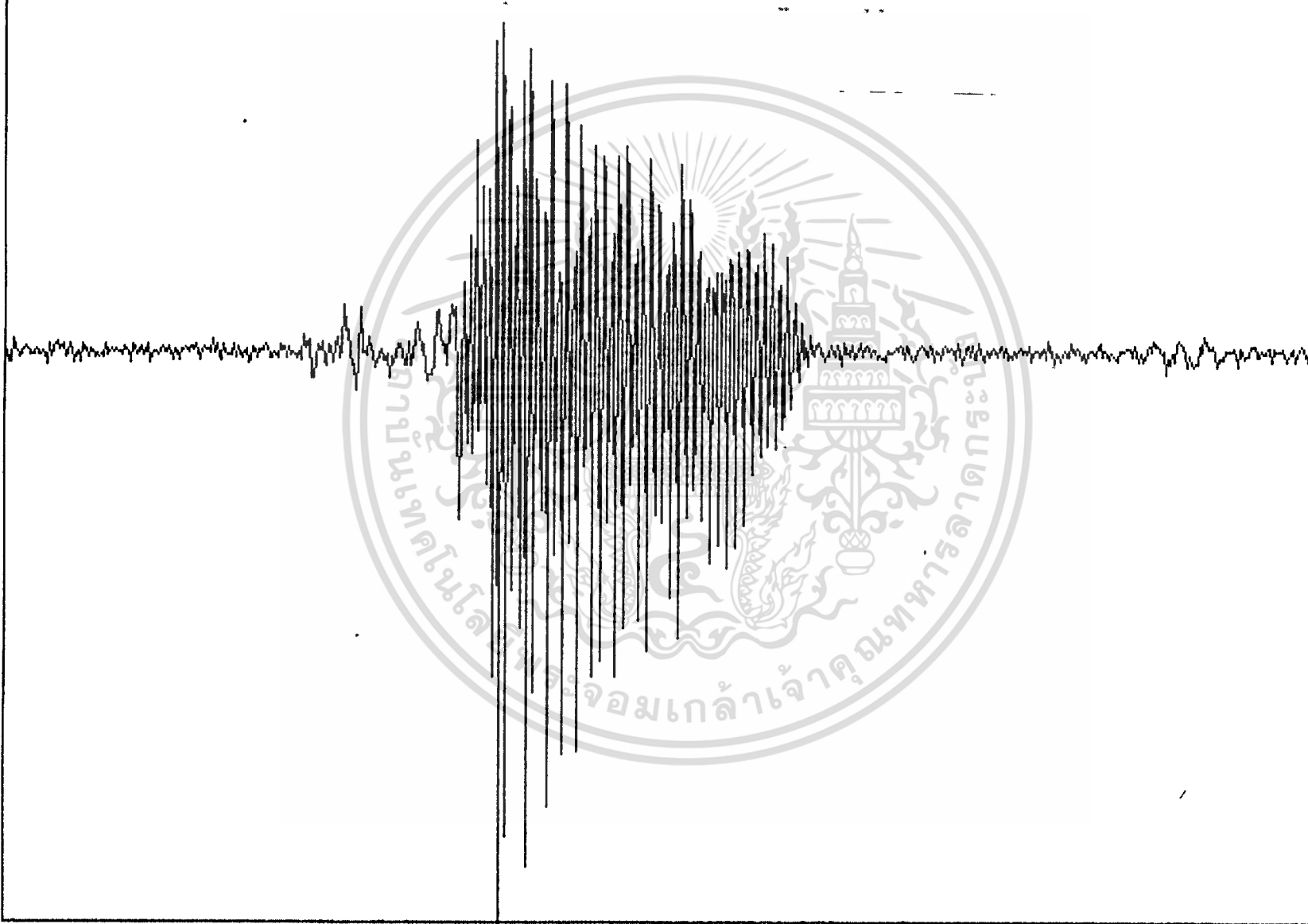
รูปที่ ข.8 แสดงภาพการหาขอบเขตของเสียง 3

PRESS ANY KEY SEE TIME-DOMAIN



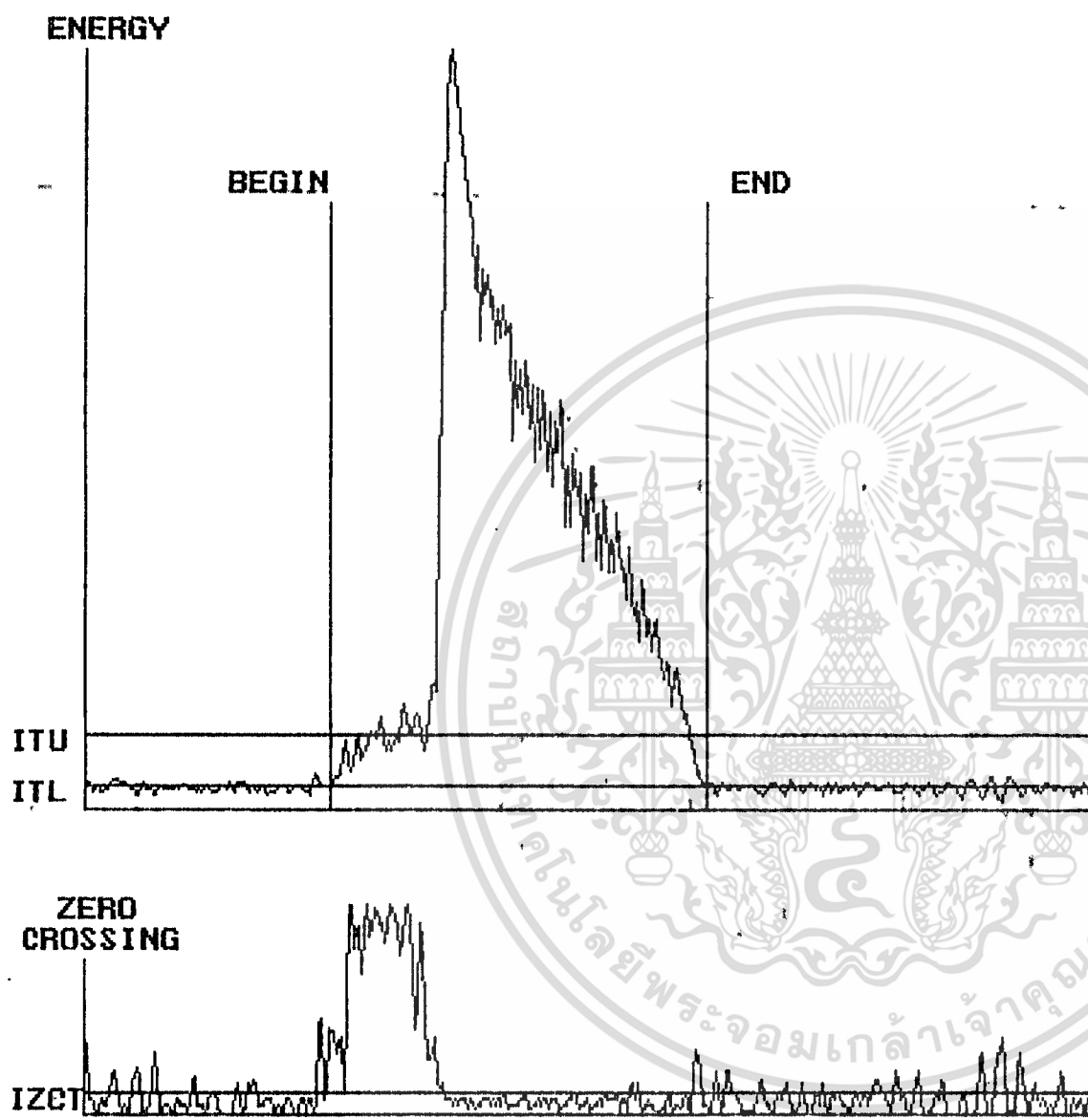
รูปที่ ข.9 แสดงภาพสัญญาณเสียง ๑ เมื่อได้รับการตัดคำแล้ว

PRESS ANY KEY TO CONTINUE



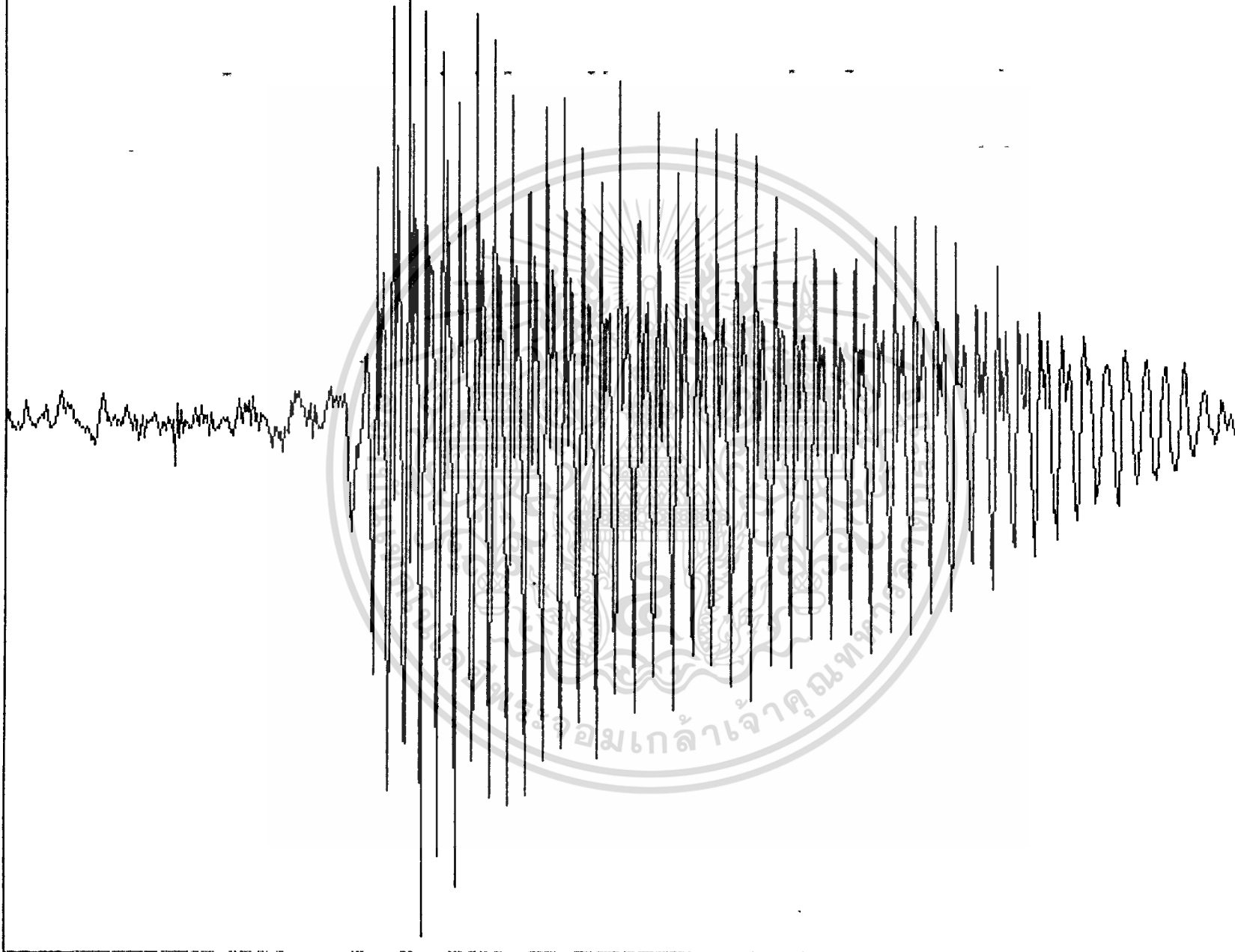
รูปที่ ข.10 แสดงภาพสัญญาณเสียง 4

PRESS ANY KEY TO CONTINUE



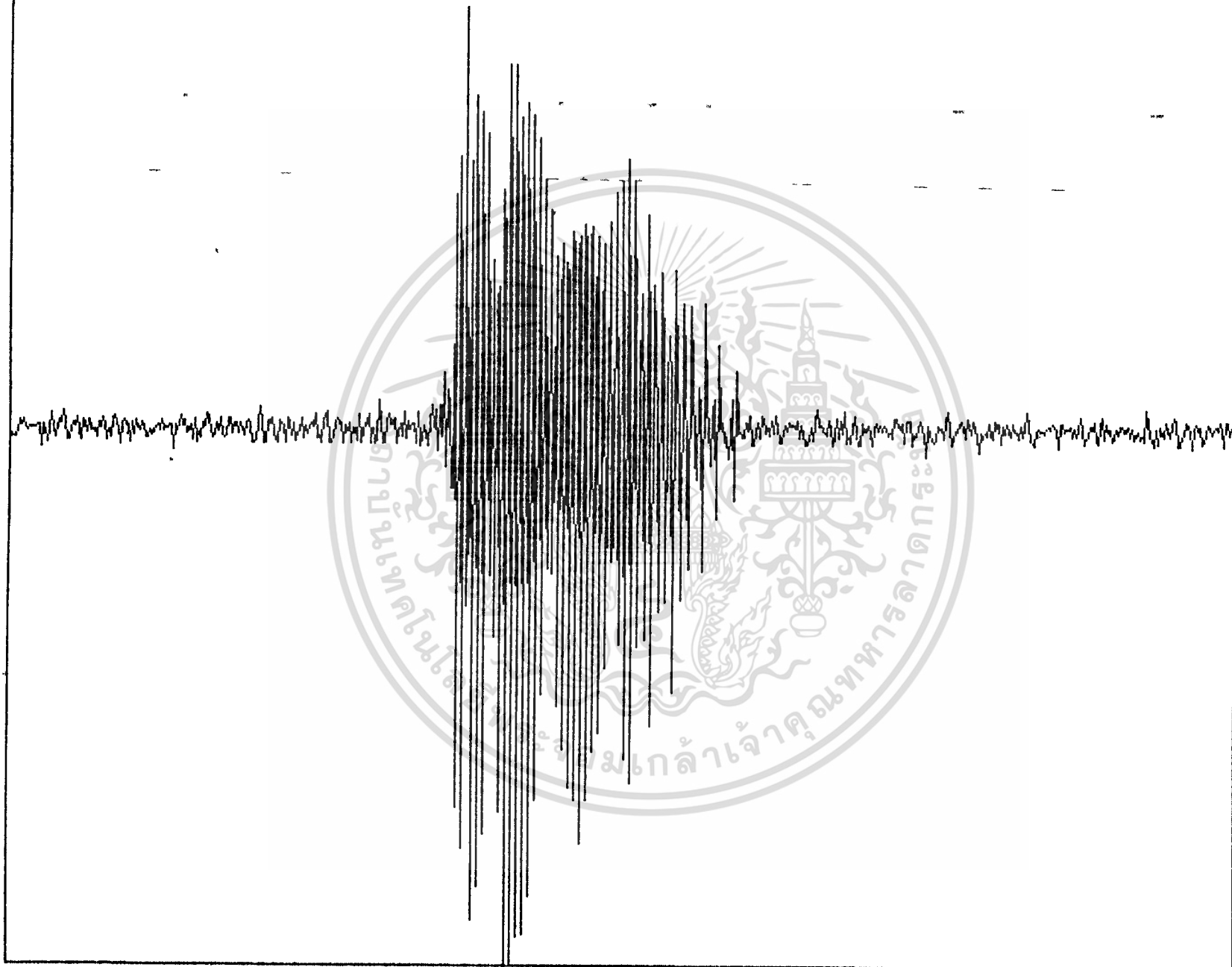
รูปที่ ข.11 แสดงภาพการหาขอบเขตของเสียง 4

PRESS ANY KEY SEE TIME-DOMAIN



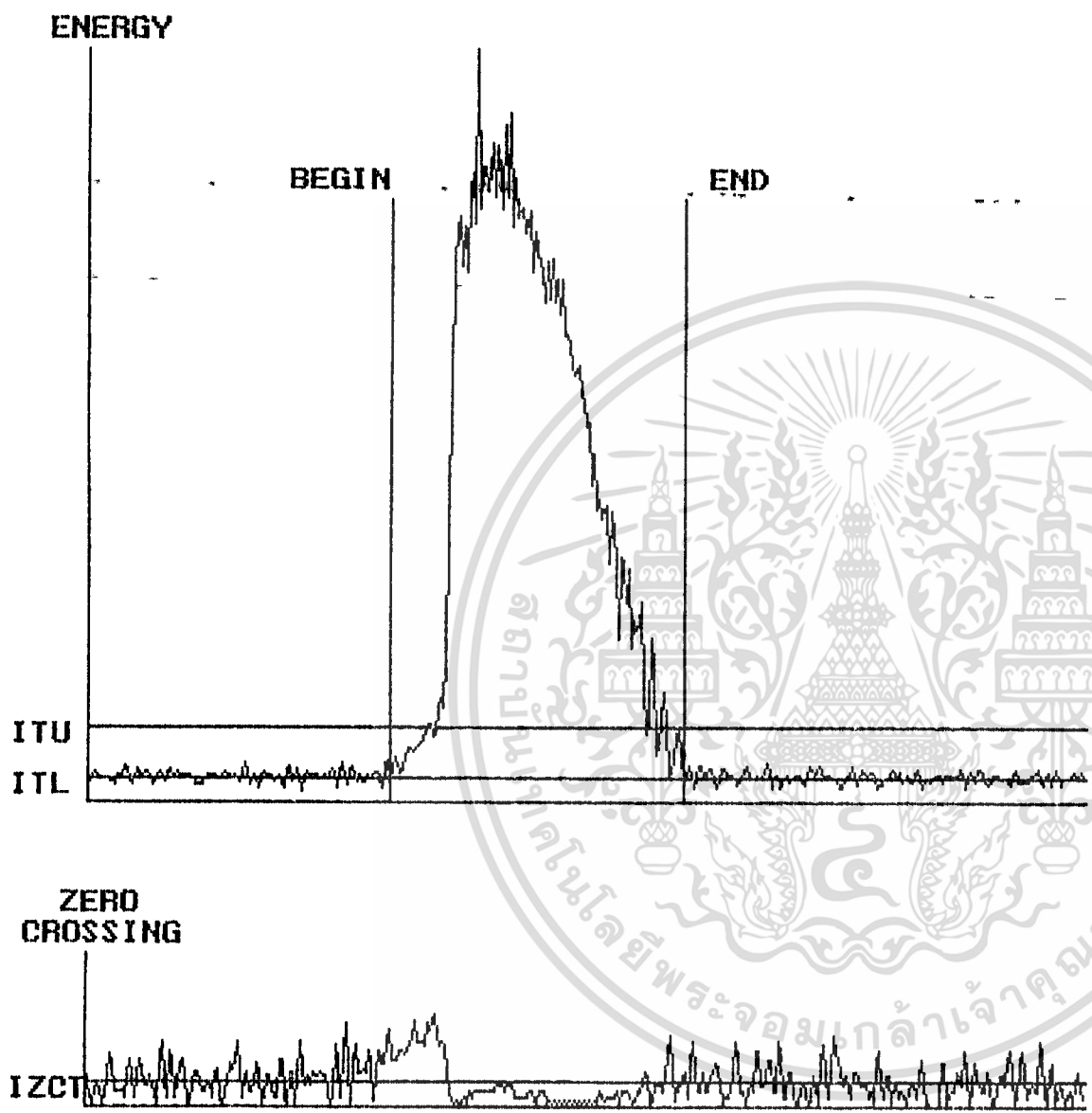
รูปที่ ข.12 แสดงภาพสัญญาณเสียง 4 เมื่อได้รับการตัดค่านแล้ว

PRESS ANY KEY TO CONTINUE



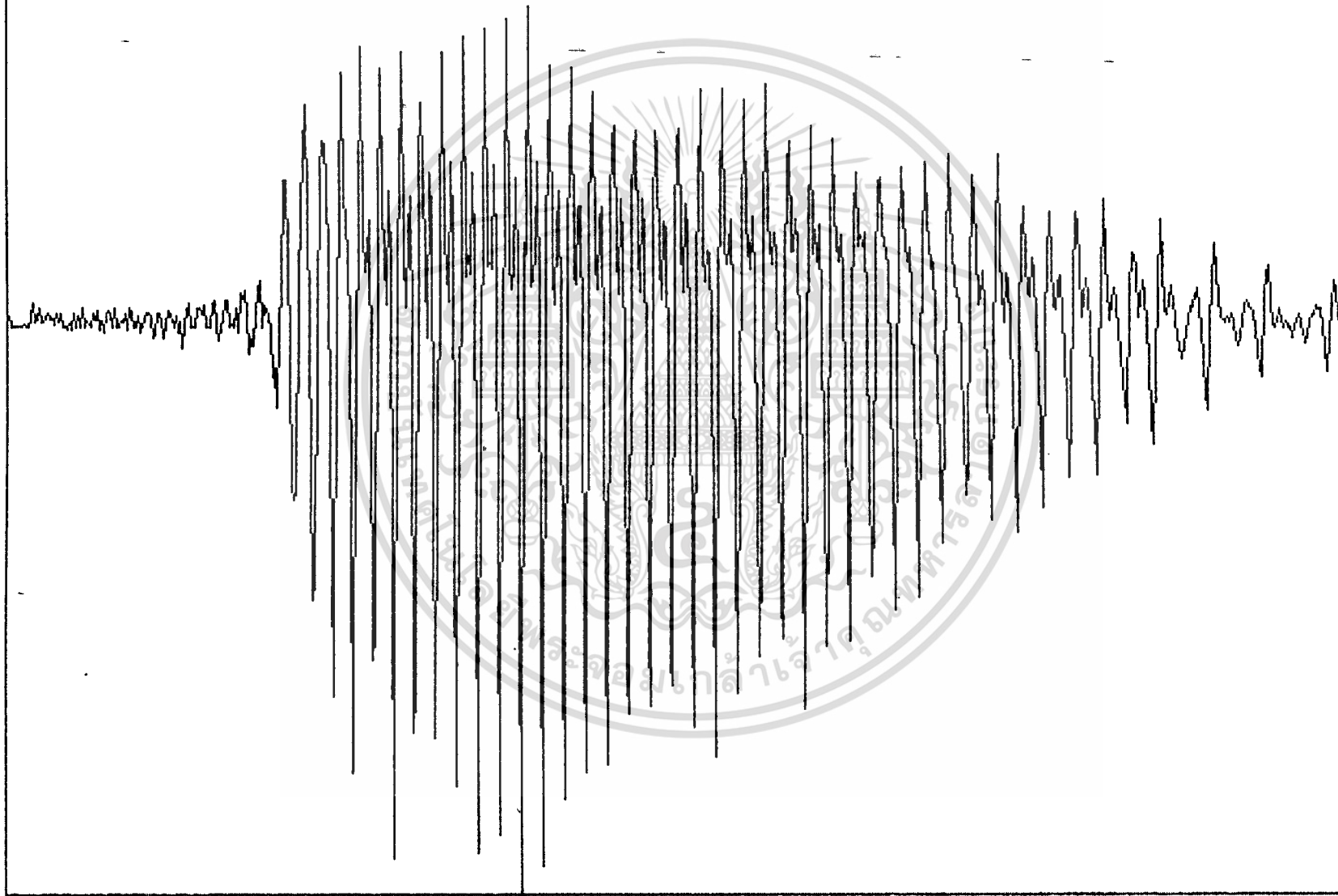
รูปที่ ข.13 แสดงภาพสัญญาณเสียง 5

PRESS ANY KEY TO CONTINUE



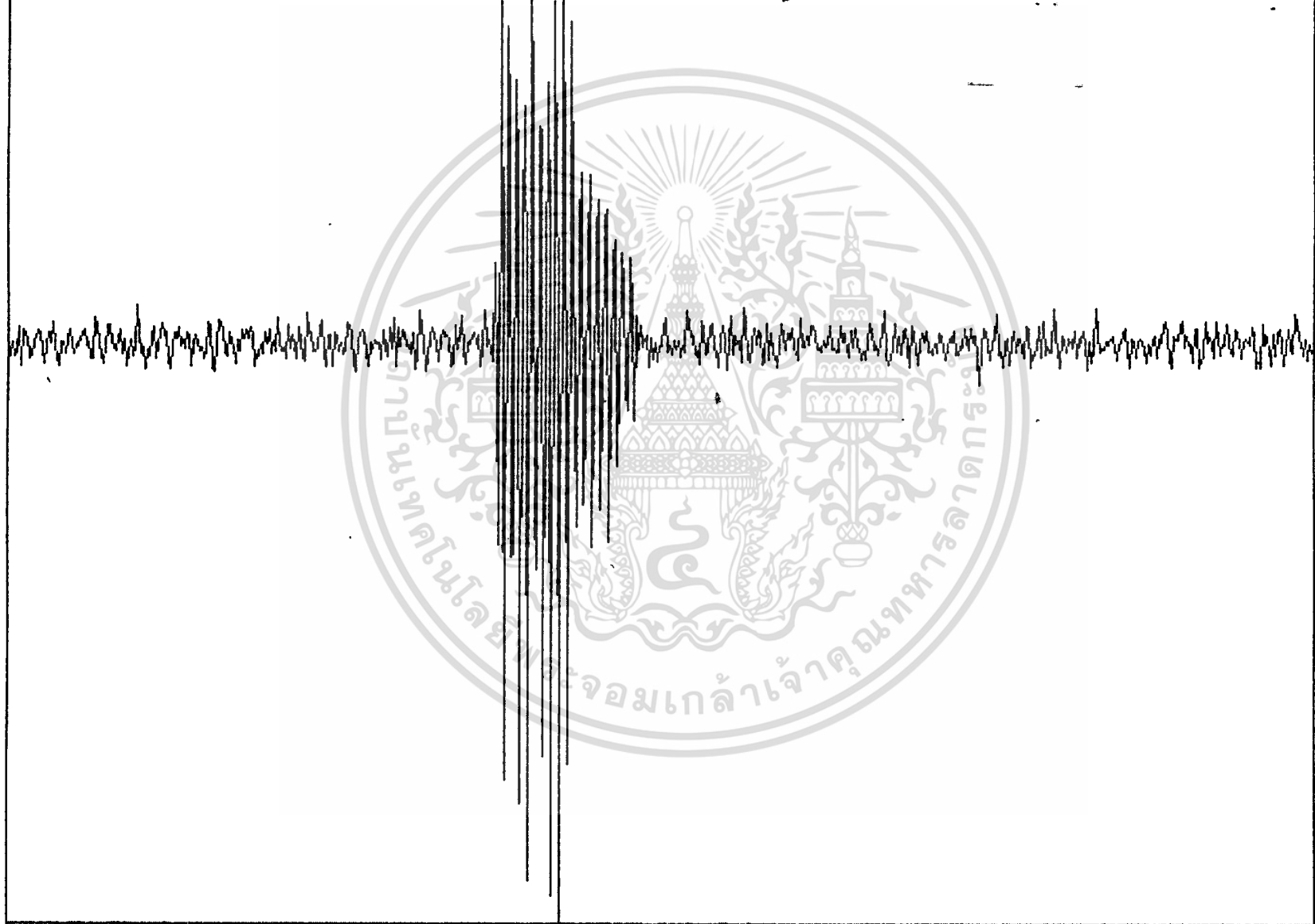
รูปที่ ข.14 แสดงภาพการหาขอบเขตของเสียง 5

PRESS ANY KEY SEE TIME-DOMAIN



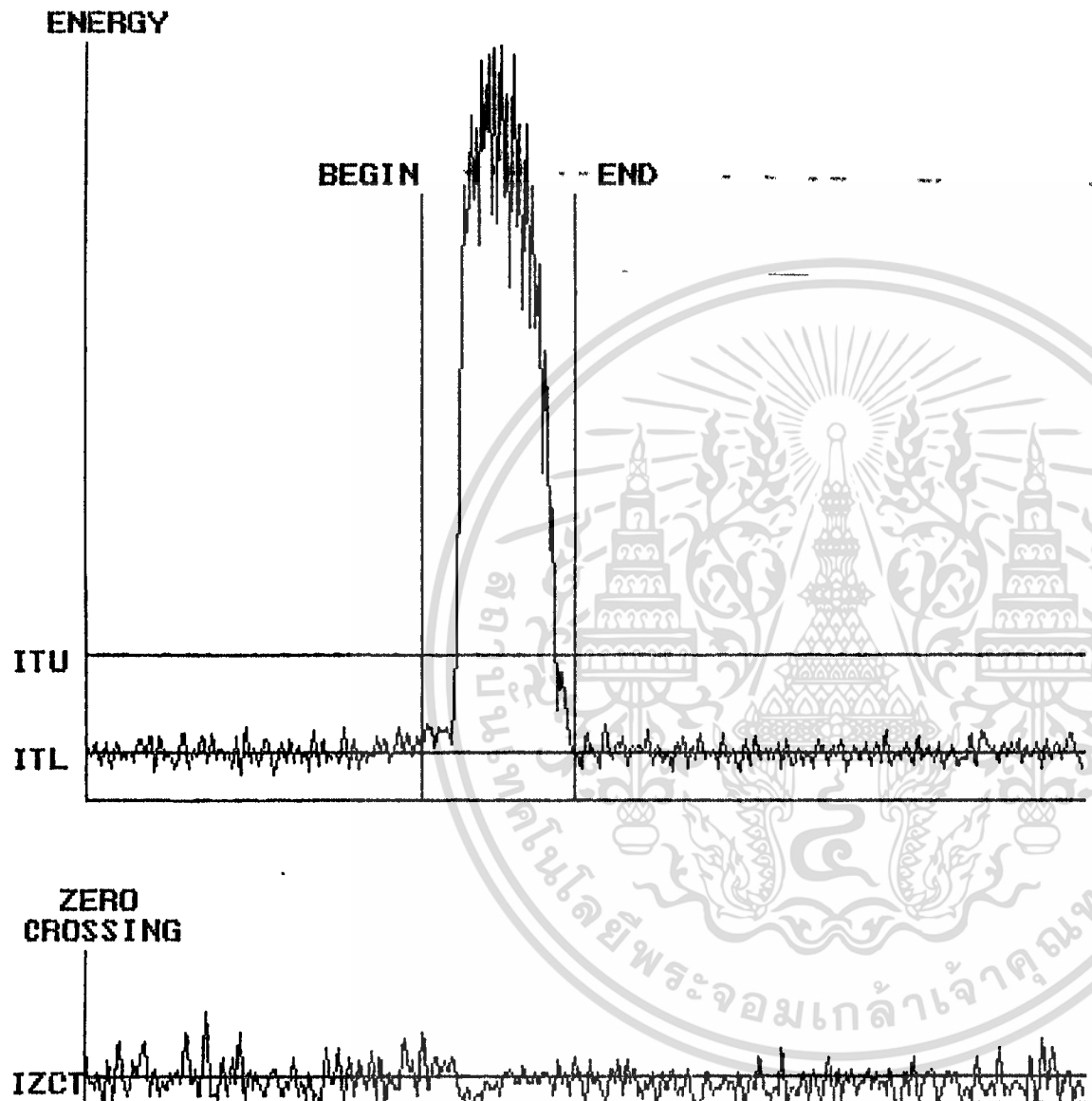
รูปที่ ข.15 แสดงภาพสัญญาณเสียง 5 เมื่อได้รับการตัดค่าแล้ว

PRESS ANY KEY TO CONTINUE



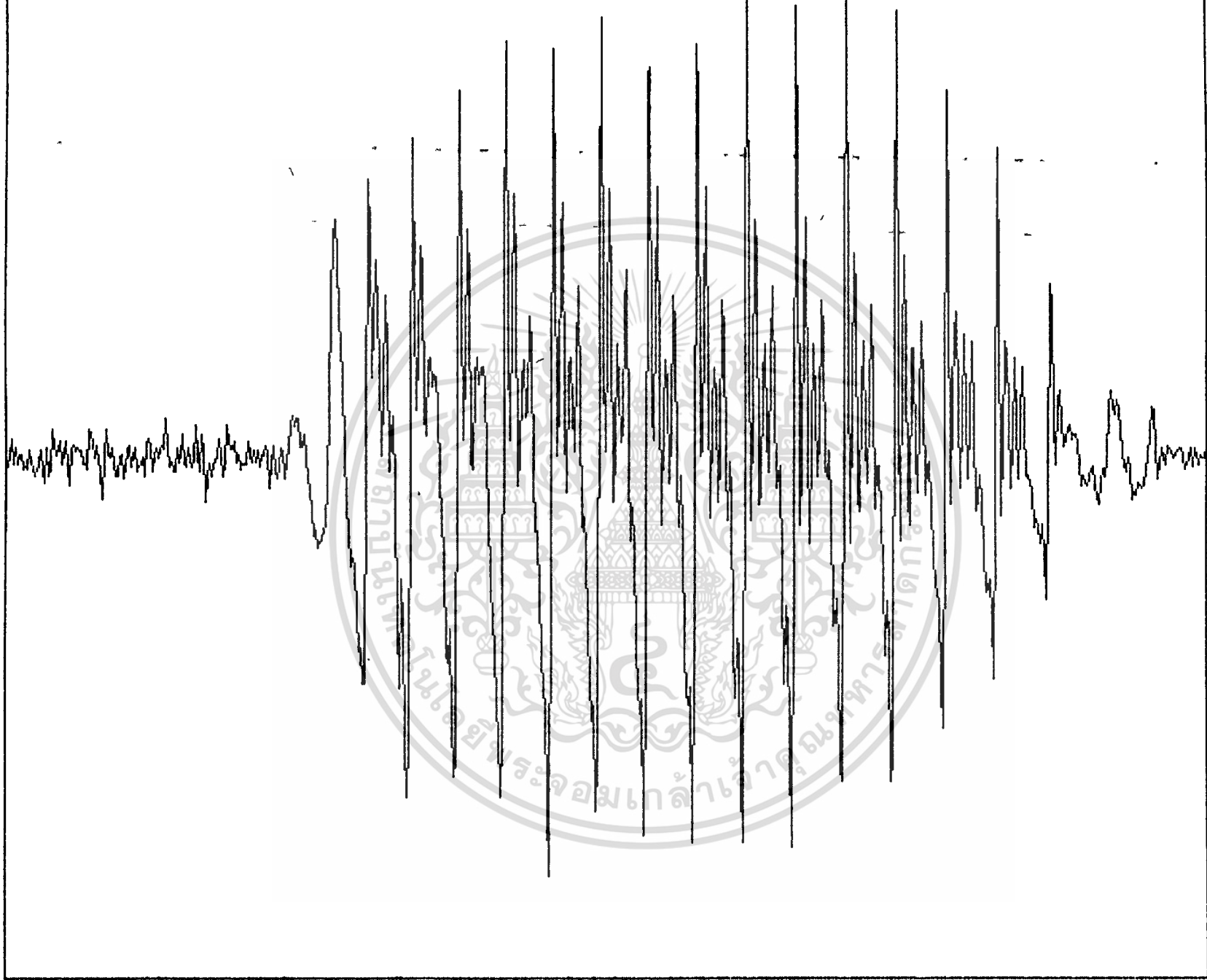
รูปที่ ข.16 แสดงภาพสัญญาณเสียง 6

PRESS ANY KEY TO CONTINUE



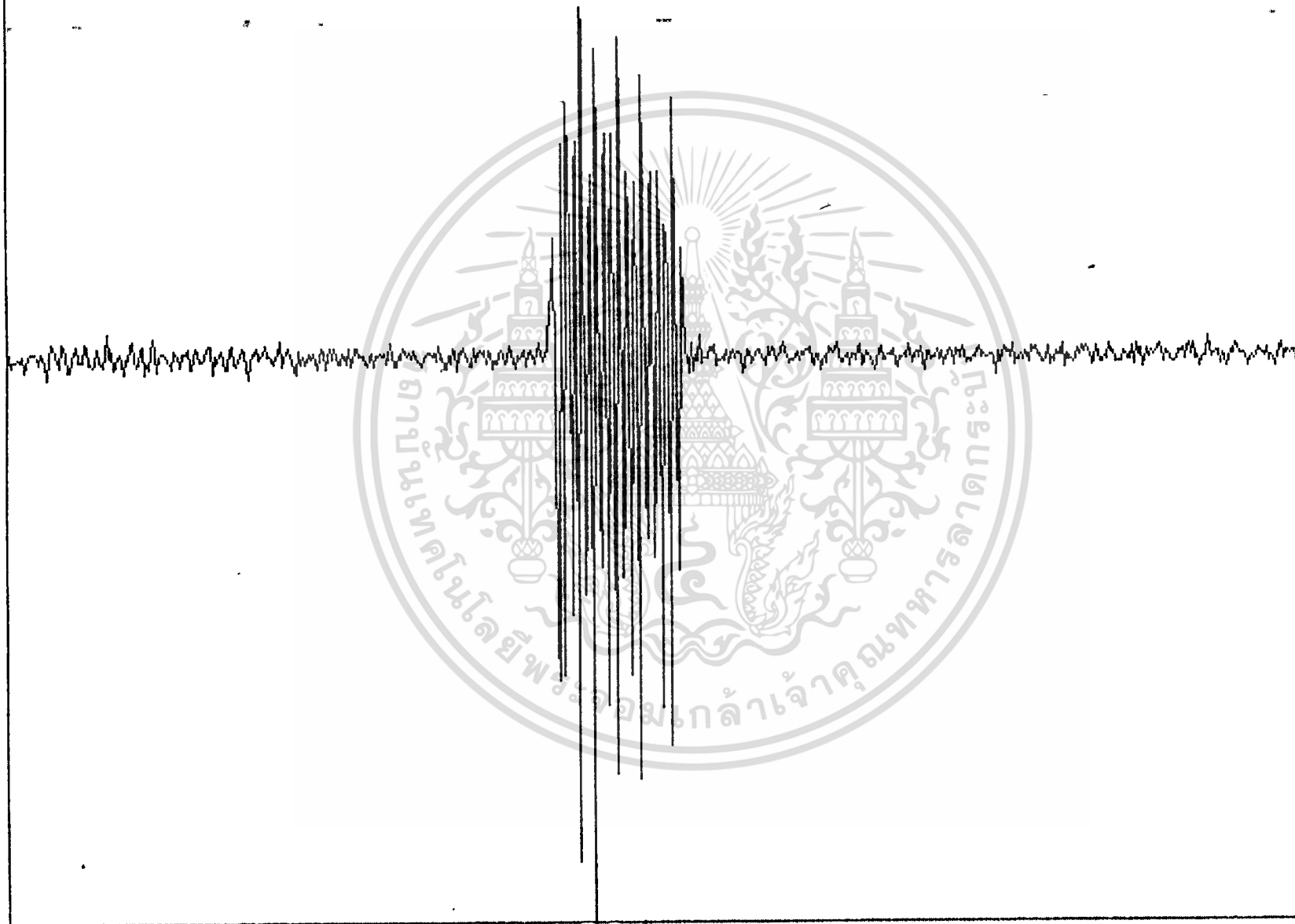
รูปที่ ข.17 แสดงภาพการหาขอบเขตของเสียง 6

PRESS ANY KEY SEE TIME-DOMAIN



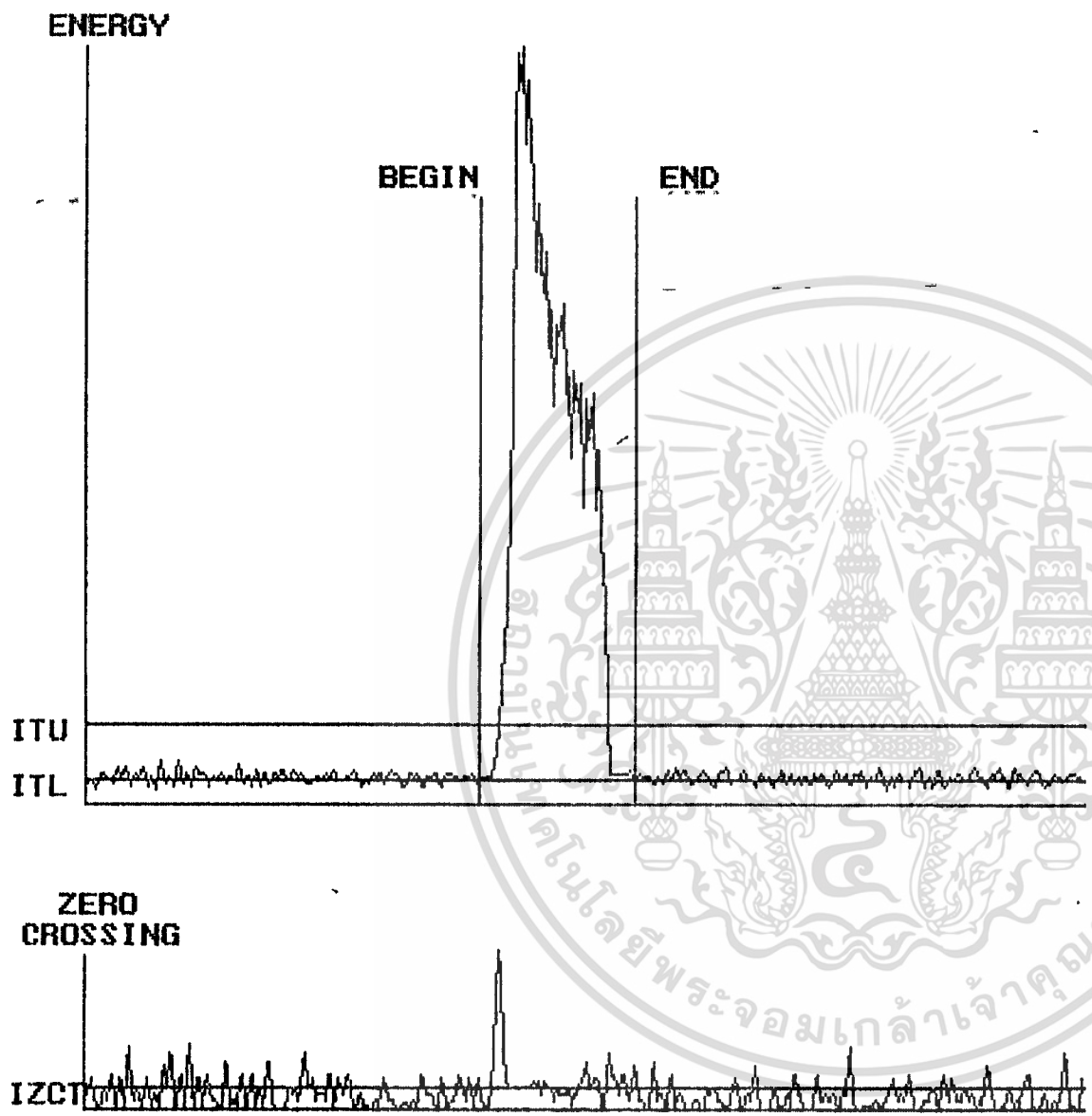
รูปที่ ข.18 แสดงภาพสัญญาณเสียง 6 เมื่อได้รับการตัดค่าแล้ว

PRESS ANY KEY TO CONTINUE



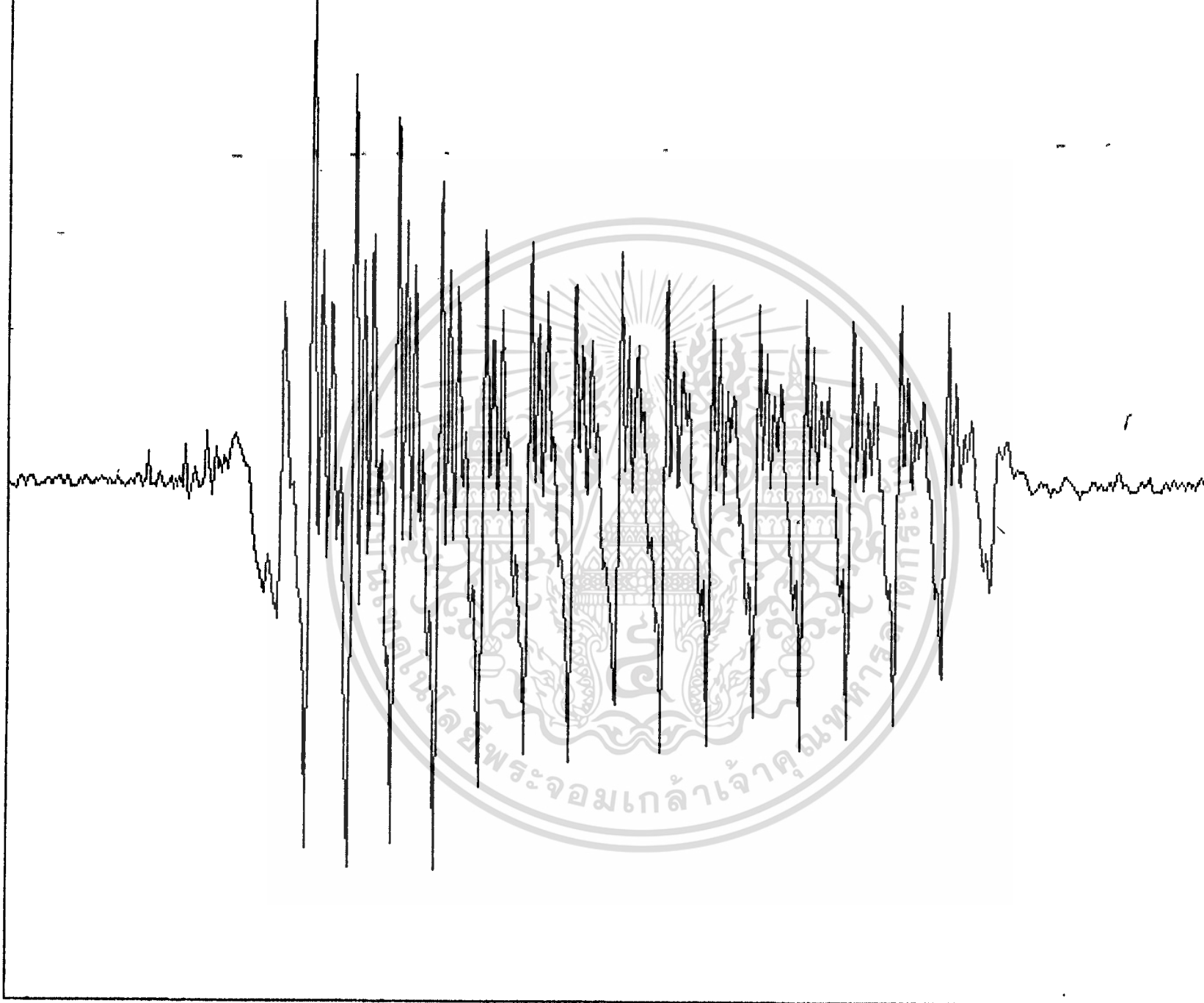
รูปที่ ข.19 แสดงภาพสัญญาณเสียง 7

PRESS ANY KEY TO CONTINUE



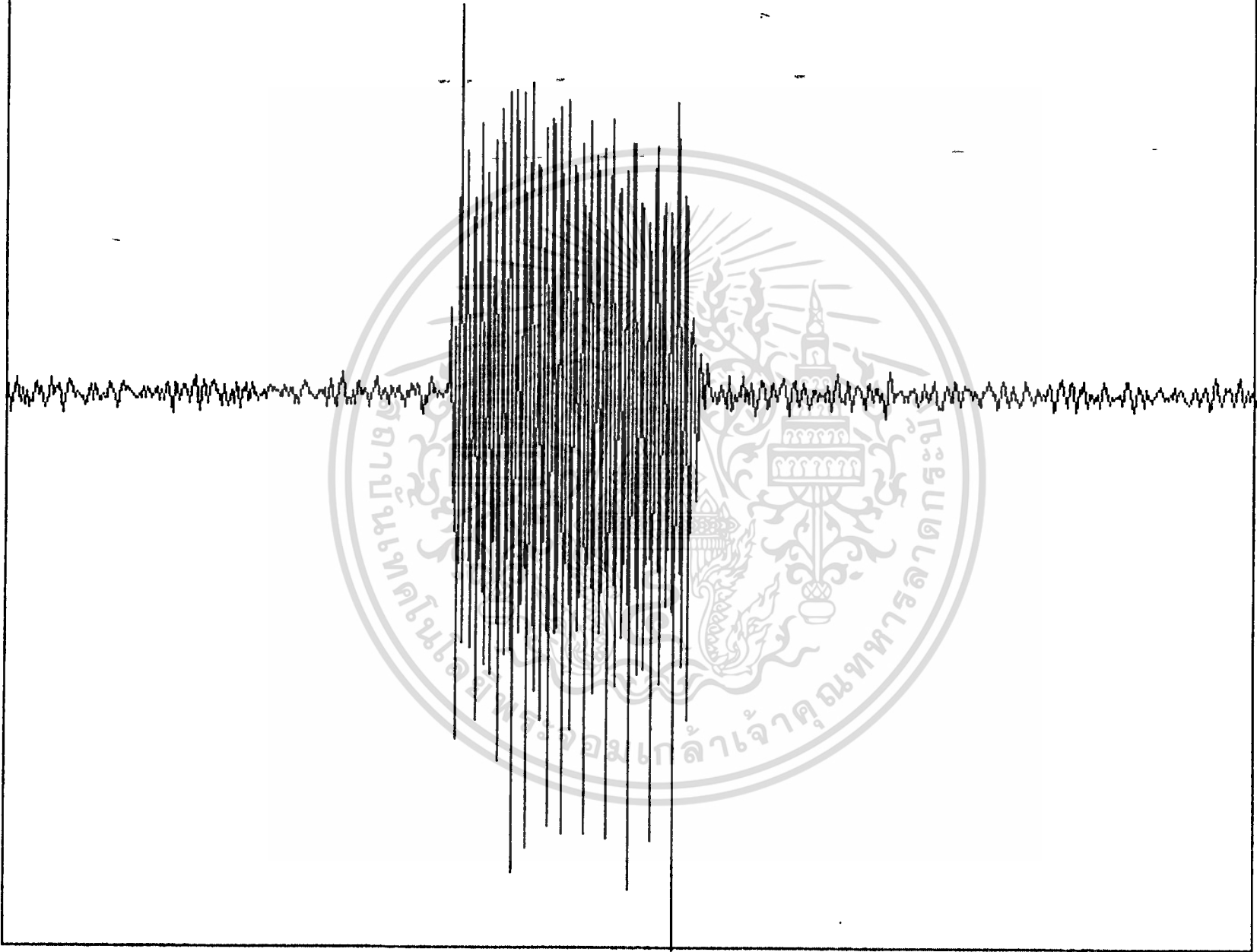
รูปที่ ข.20 แสดงภาพการหาขอบเขตของเสียง 7

PRESS ANY KEY SEE TIME-DOMAIN



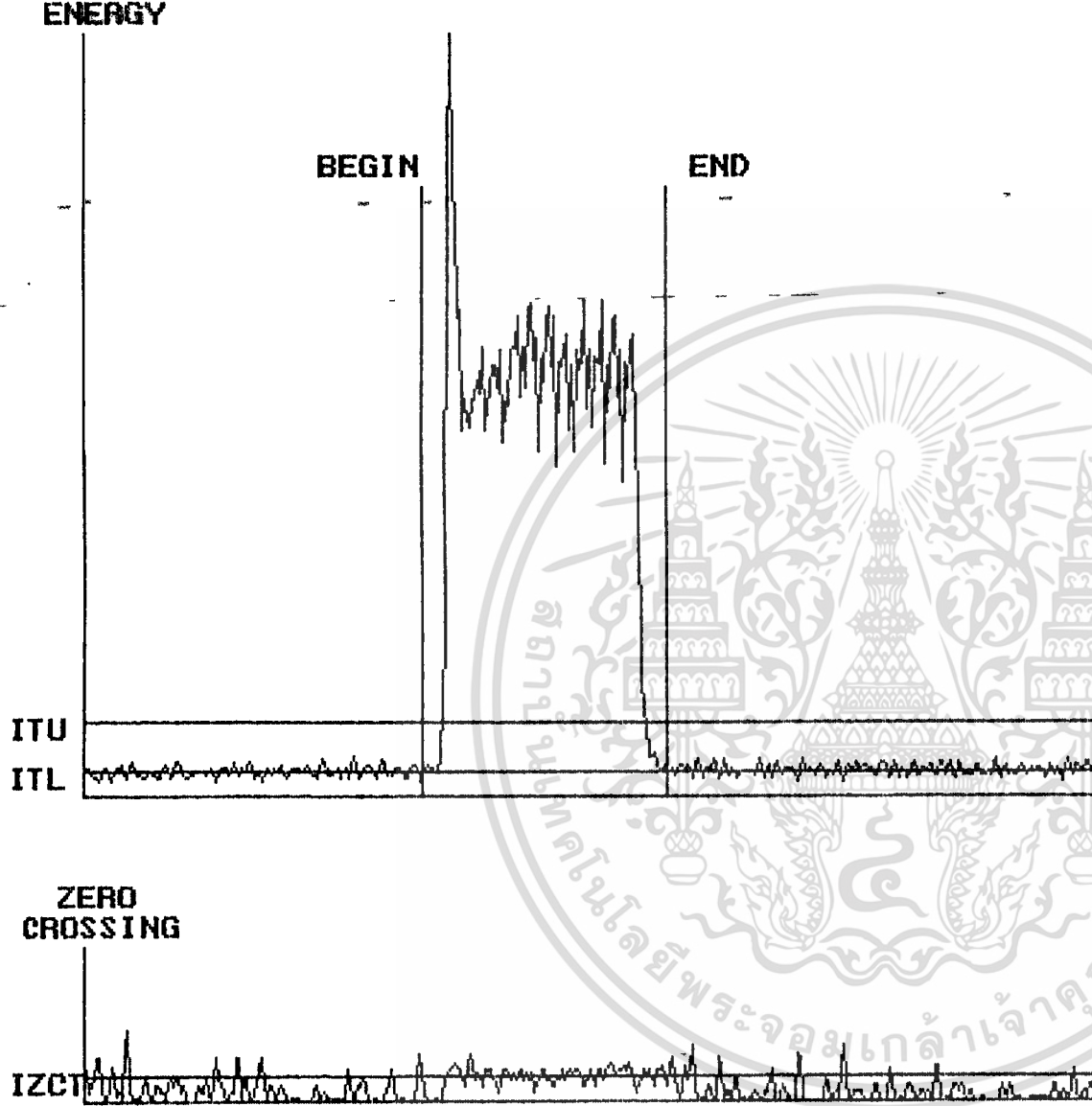
รูปที่ ข.21 แสดงภาพสัญญาณเสียง 7 เมื่อได้รับการตัดค่าแล้ว

PRESS ANY KEY TO CONTINUE



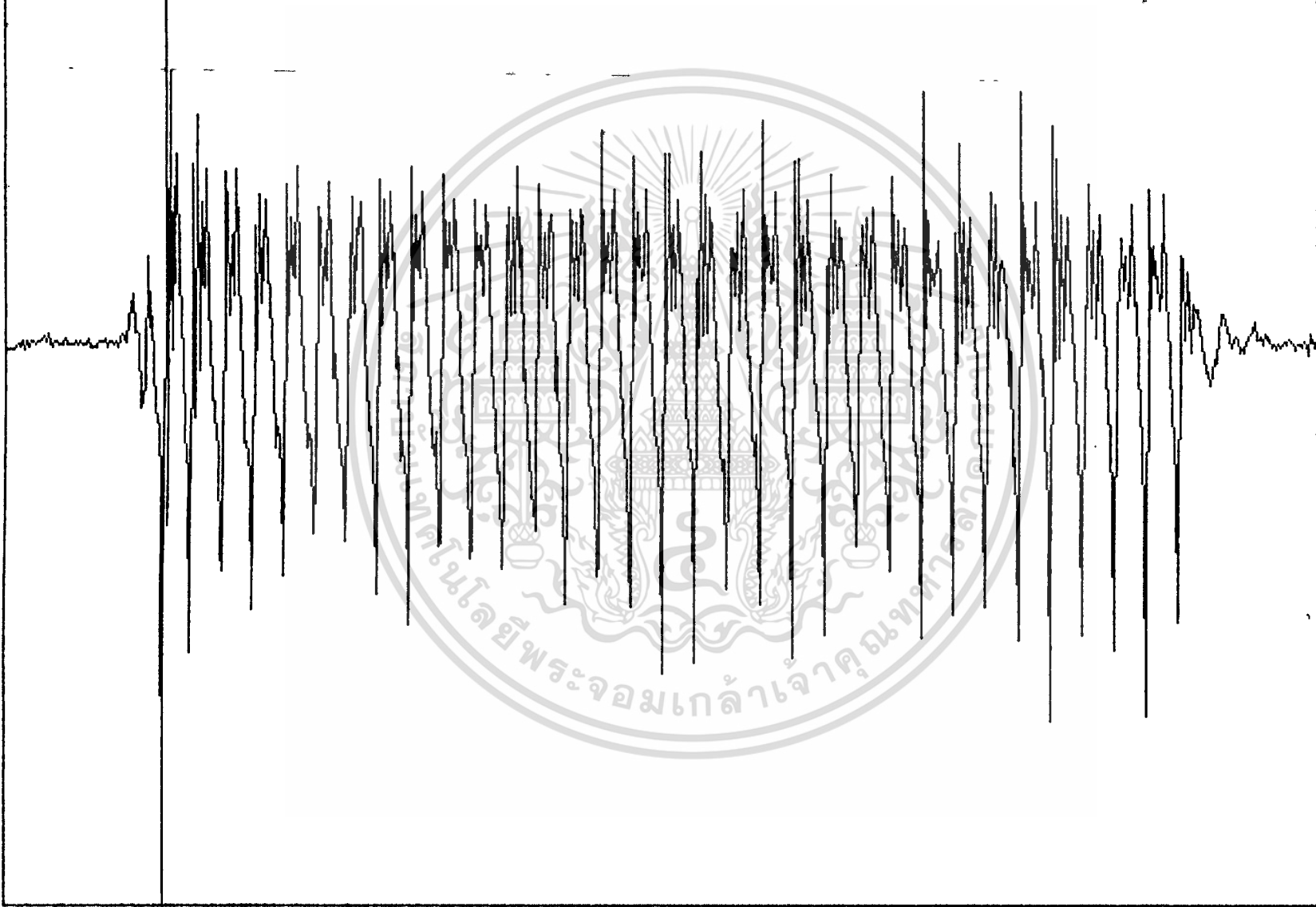
รูปที่ ข.22 แสดงภาพสัญญาณเสียง ๘

PRESS ANY KEY TO CONTINUE



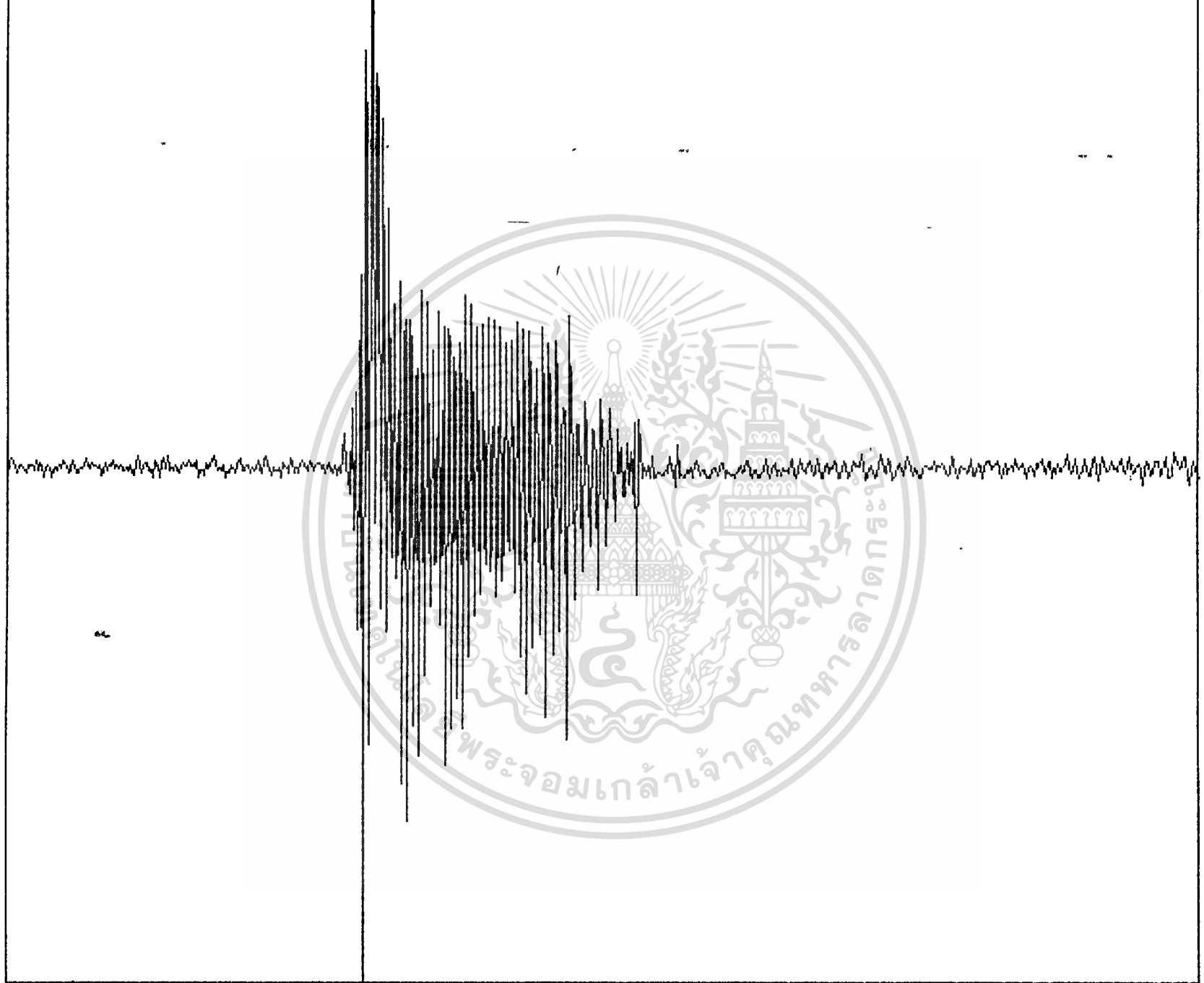
รูปที่ ข.23 แสดงภาพการหาขอบเขตของเสียง 8

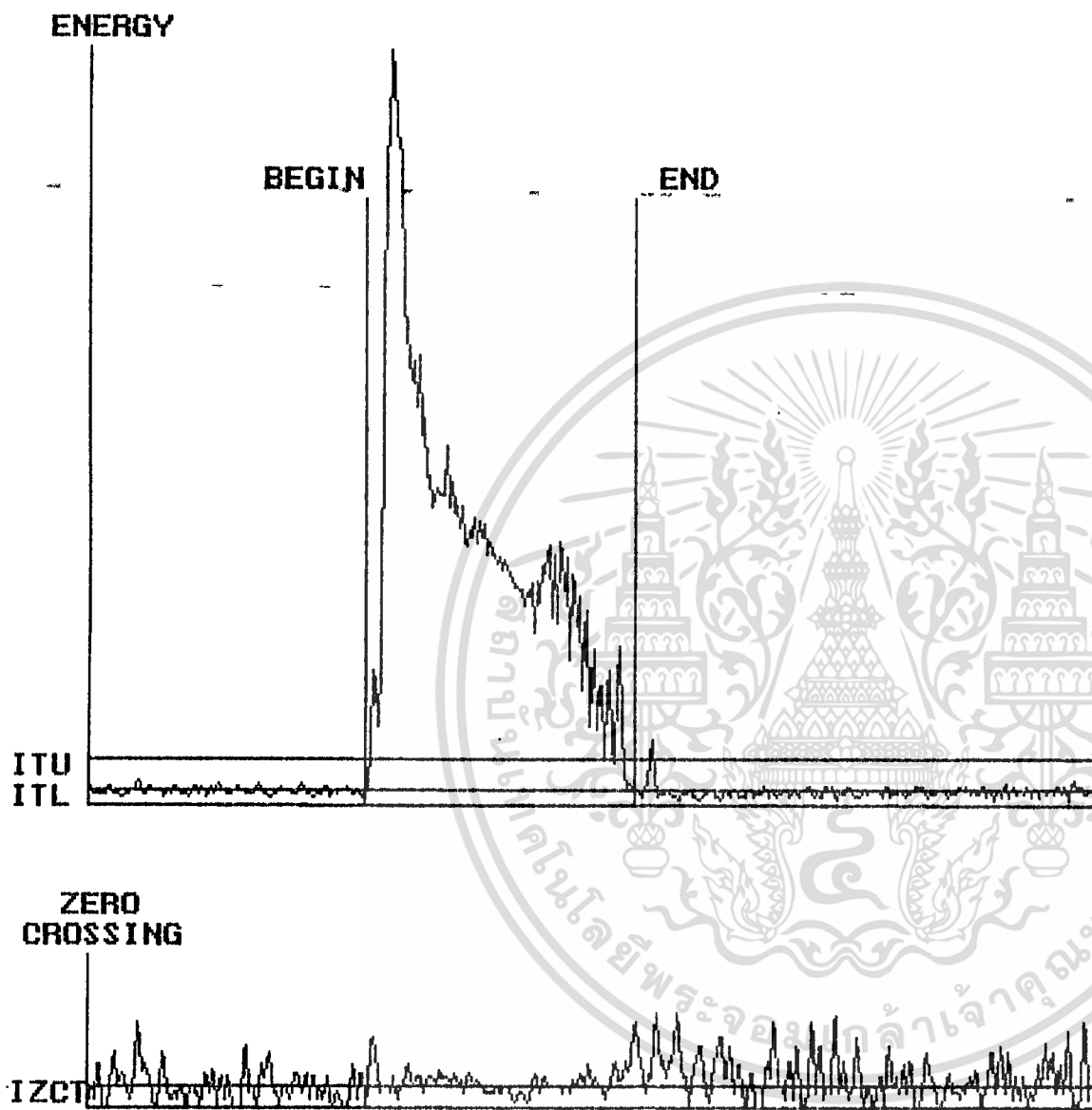
PRESS ANY KEY SEE TIME-DOMAIN



รูปที่ ข.24 แสดงภาพสัญญาณเสียง 8 เมื่อได้รับการตัดคำแล้ว

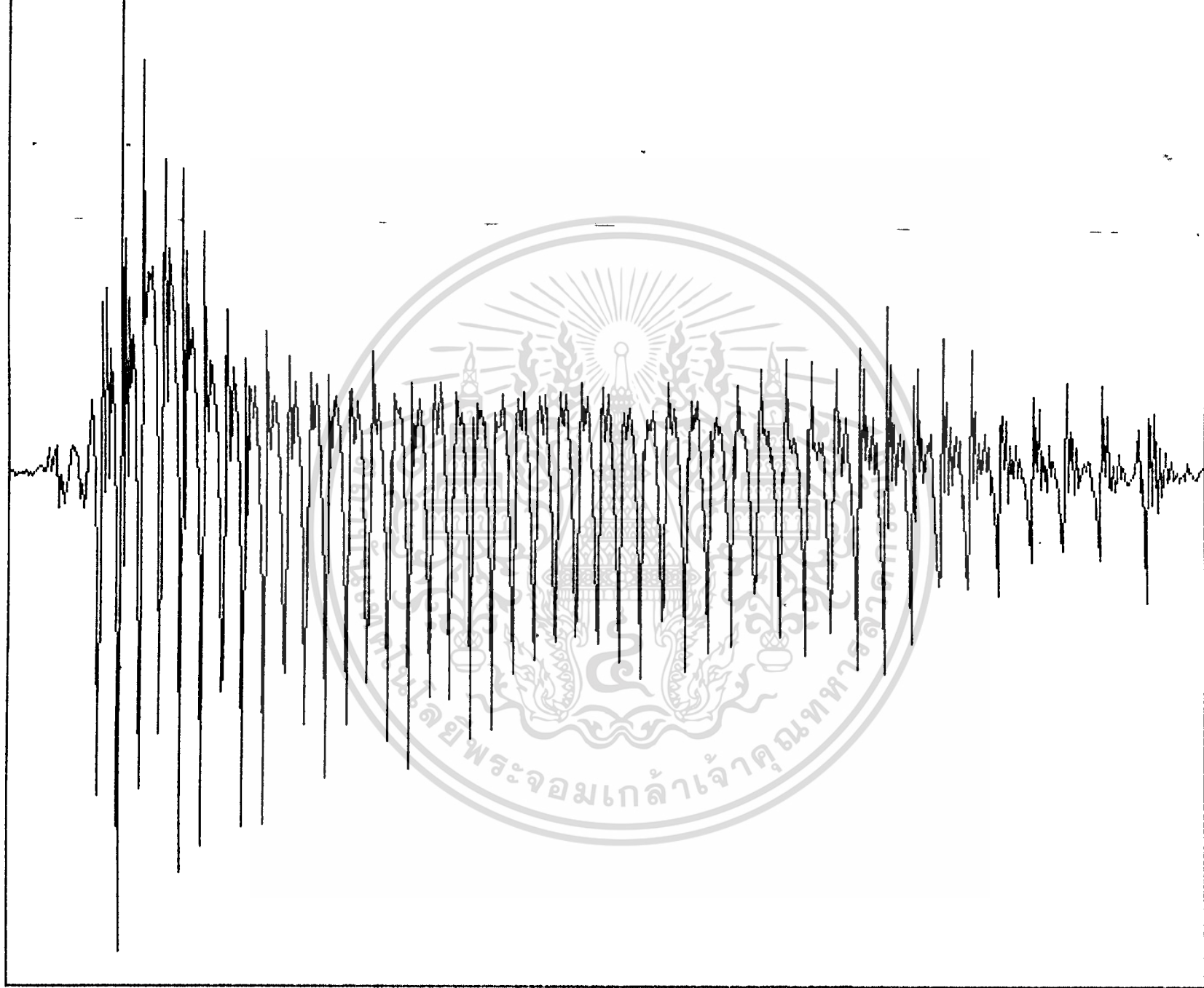
PRESS ANY KEY TO CONTINUE





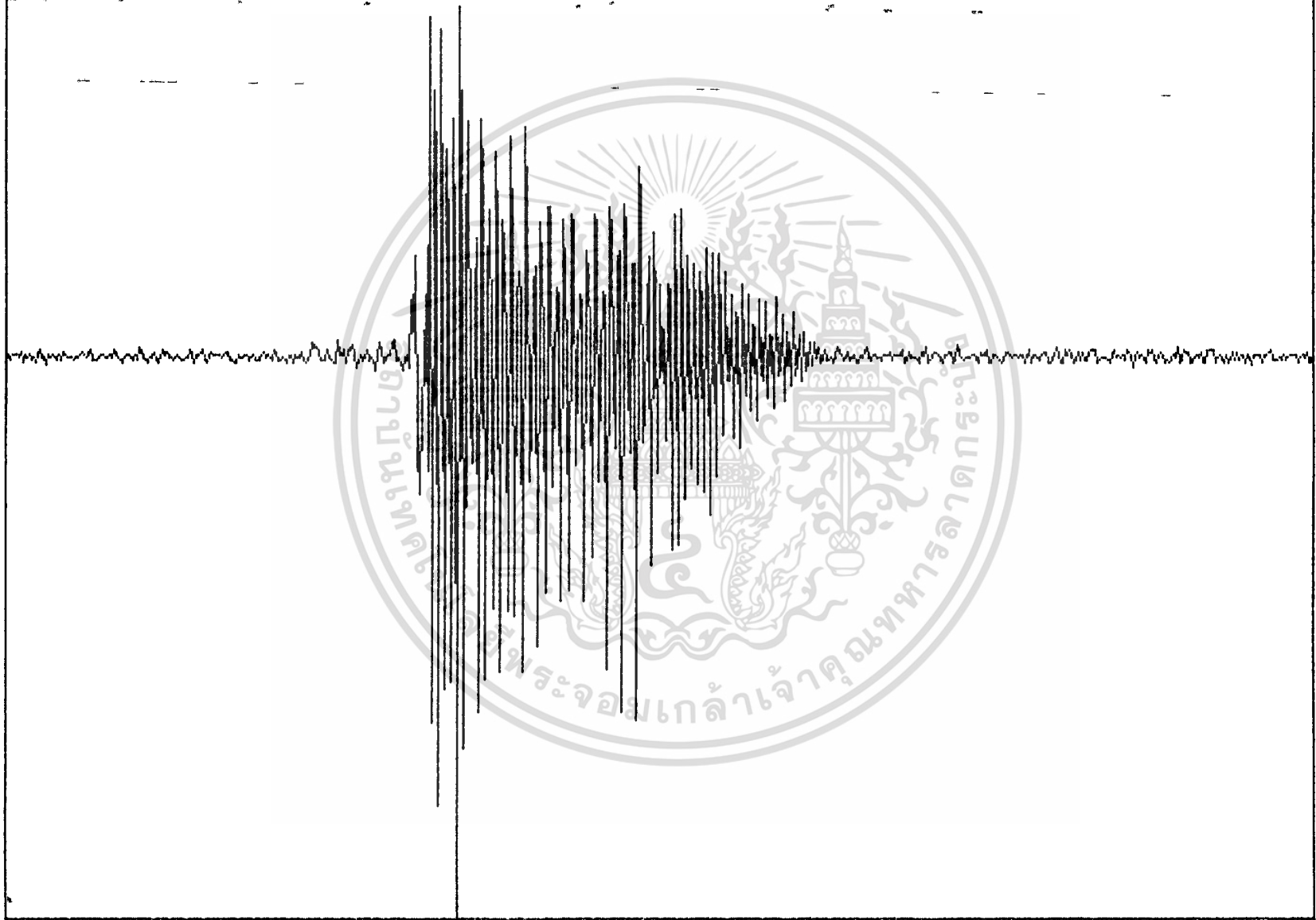
รูปที่ ข.26 แสดงภาพการหาขอบเขตของเสียง 9

PRESS ANY KEY SEE TIME-DOMAIN



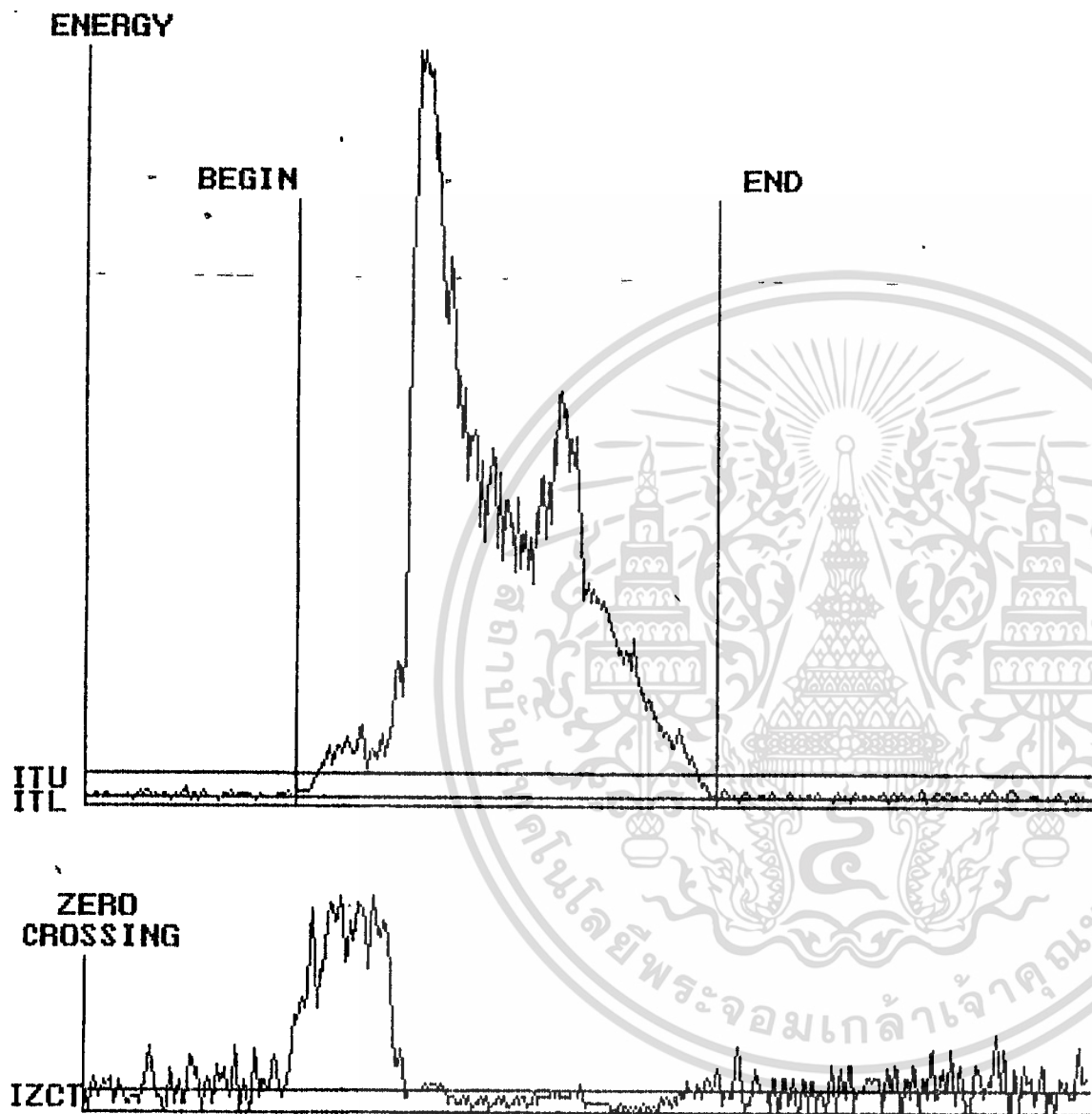
รูปที่ ข.27 แสดงภาพสัญญาณเสียง 9 เมื่อได้รับการตัดค่าแล้ว

PRESS ANY KEY TO CONTINUE



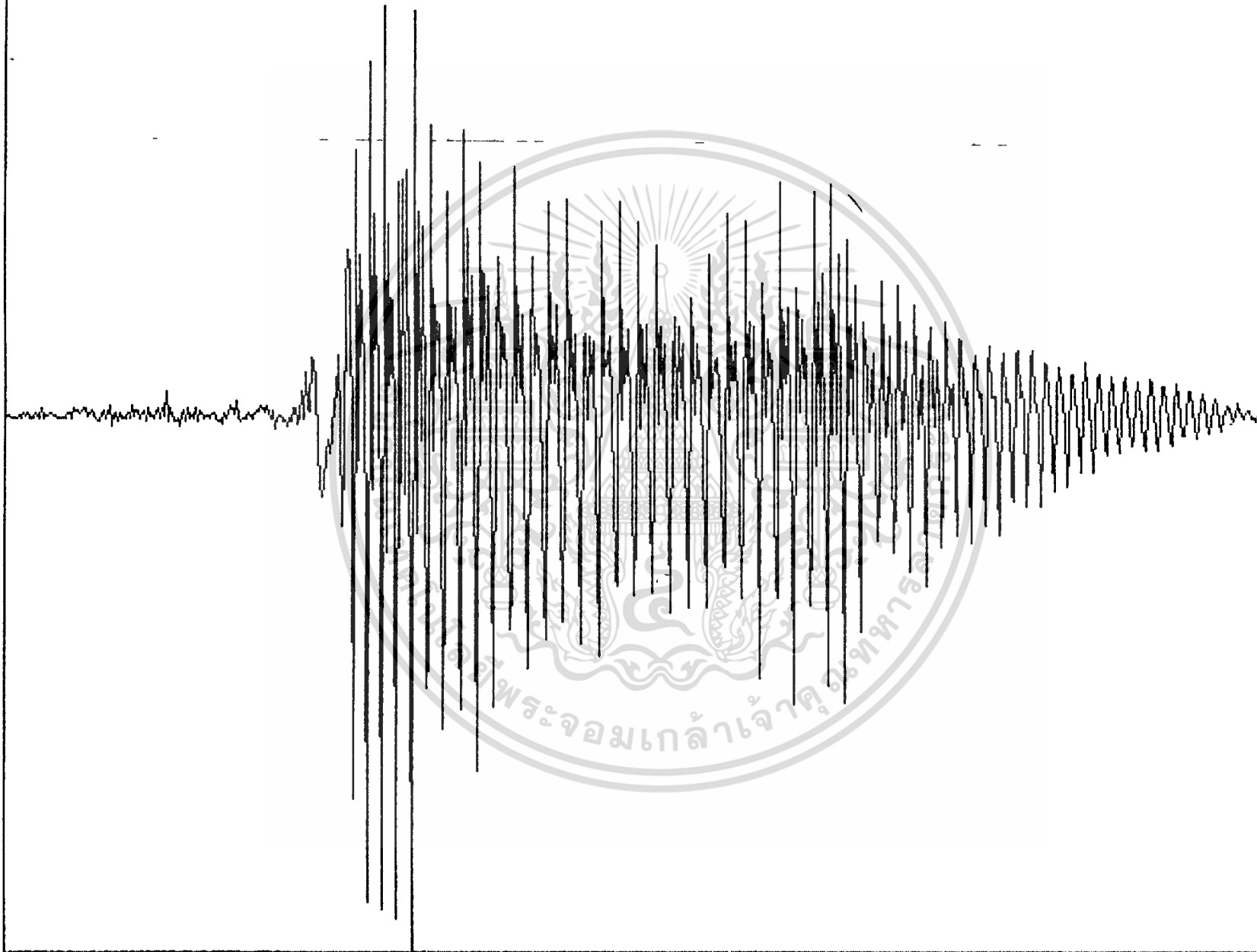
รูปที่ ข.28 แสดงภาพสัญญาณเสียง 0

PRESS ANY KEY TO CONTINUE



รูปที่ ข.29 แสดงภาพการหาขอบเขตของเสียง 0

PRESS ANY KEY SEE TIME-DOMAIN



รูปที่ ข.30 แสดงภาพสัญญาณเสียง 0 เมื่อได้รับการตัดคำแล้ว

PRESS ANY KEY TO CONTINUE