

การควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์

TELEPHONIC CONTROL SYSTEM ELECTRICAL APPLIANCE



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เลขหม.....
เลขทะเบียน..... 37167
วัน, เดือน, ปี..... - 4 ก.ย. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์
TELEPHONIC CONTROL SYSTEM ELECTRICAL APPLIANCE



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2542

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านโทรศัพท์

TELEPHONIC CONTROL SYSTEM ELECTRICAL APPLIANCE

ผู้จัดทำ

- | | | |
|------------------|--------------|----------|
| 1. นางสาวมิณฑรา | ตันสุขานันท์ | 39014410 |
| 2. นางสาวสุทธิดา | ดุจดอนุกิจ | 39014585 |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์

TELEPHONIC CONTROL SYSTEM ELECTRICAL APPLIANCE

โดย นางสาวมณฑรา ตันสุขานันท์ 39014410

นางสาวสุทธิดา ดุลยอนุกิจ 39014585

อาจารย์ที่ปรึกษา ผศ.เกรียงไกร วงศ์โรจนภรณ์

รศ.ดร. สุวิพล สิทธีชีวะภาค

บทคัดย่อ

การควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์สามารถใช้งานได้โดย โทรศัพท์ผ่านชุมสายเข้ามาในบ้าน ซึ่งจะใช้โมเด็ม (Modem) ในการรับโทรศัพท์แล้วส่งงานให้เครื่องใช้ไฟฟ้าเปิดหรือปิดได้ด้วยเสียงของผู้ใช้เอง

หลักการทำงานของระบบการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านโทรศัพท์ คือ เมื่อมีการโทรศัพท์ผ่านชุมสายเข้ามาในบ้านแล้วคอมโพเนนต์ (component) ที่เขียนด้วยภาษาเดลฟี (Delphi) จะทำหน้าที่ในการรับโทรศัพท์อัตโนมัติ และบันทึกเสียงผู้ใช้ไว้ จากนั้นนำไฟล์เสียงที่บันทึกไว้ไปทำการรู้จำเสียงพูดด้วย โปรแกรมรู้จำเสียงพูดซึ่งเขียนภาษาเมทแลบ (MATLAB) จากนั้นนำผลที่ได้จากโปรแกรมรู้จำเสียงพูด(ปริญาณิพนธ์นี้จะใช้เสียงหนึ่งถึงสี่) ไปส่งออกพอร์ตขนานด้วยโปรแกรมภาษาซี เพื่อใช้ควบคุมเครื่องใช้ไฟฟ้าต่อไป

ABSTRACT

Telephonic control system electrical appliance is worked by making a call through a telephone exchange to a house, and uses one word to control the electrical appliances.

When making a call to a house via telephone exchange, delphi component will automatically receive the call and record the control word. The control word will be saved in wave file format. Then the MATLAB speech recognition program will recognize whether the word is. (one, two, three, or four) After that C programming language will send out the result from speech recognition program to the parallel port, to control the electrical appliances.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีหรือหลักการ	2
2.1 โปรแกรมรับโทรศัพท์อัตโนมัติ	3
2.2 โปรแกรมอินเทอร์เฟซระหว่างโปรแกรมรับโทรศัพท์อัตโนมัติกับโปรแกรมรู้จำเสียงพูด	3
2.2.1 เมทแลบคอมไพเลอร์	4
2.3 โปรแกรมรู้จำเสียงพูด	6
2.3.1 การวิเคราะห์ในโดเมนเวลา	7
2.3.1.1 การหาค่าพลังงานของสัญญาณเสียง	7
2.3.1.2 การหาค่าจุดตัดศูนย์	8
2.3.2 การเตรียมสัญญาณ	9
2.3.2.1 การฟรีเอมฟาซิส	9
2.3.2.2 การแบ่งช่วงสัญญาณ	10
2.3.2.3 การทำวินโดว์	10
2.3.2.4 การหาขอบเขตของคำ	11
ขั้นตอนที่ 1	12
ขั้นตอนที่ 2	12
2.3.2.5 การนอร์มอลไลซ์	13
2.3.3 การหาค่าพารามิเตอร์	13
2.3.3.1 การวิเคราะห์การประมาณพหุระเชิงเส้น	13
2.3.3.2 การหาอดีตสหสัมพันธ์	18
สัมประสิทธิ์พาร์คอร์	20
2.3.3.3 การหาอัตราขยาย G	24
2.3.3.4 การเปลี่ยนพารามิเตอร์การประมาณพหุระเชิงเส้นเป็น	25
สัมประสิทธิ์เซปสตรัม	
2.3.3.5 สัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก	26
2.3.4 การเทียบกับรูปแบบอ้างอิง	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4.1	ภาคแบบอ้างอิง	27
2.3.4.2	ฟังก์ชันเรปิ้ง	27
2.3.4.4	สมการไดนามิกโปรแกรมมิ่ง	31
2.3.5	กฎการตัดสินใจ	32
2.3.5.1	กฎ Nearest Neighbor (NN)	33
2.3.5.2	กฎ K-Nearest Neighbor (KNN)	33
2.4	โปรแกรมอินเทอร์เฟสระหว่างโปรแกรมรู้จำเสียงพูดกับโปรแกรมส่งข้อมูลออกพอร์ตนาน	34
2.4.1	ซีเม็กซ์-ไฟล์	34
2.4.2	ส่วนประกอบของโปรแกรมซีเม็กซ์-ไฟล์	34
2.4.2.1	โปรแกรมย่อยสำหรับการประมวลผล	34
2.4.2.2	โปรแกรมย่อยเคเวย์	35
2.5	วงจรแสดงผลออกทางพอร์ตนาน	37
บทที่ 3	การคำนวณและการสร้าง	39
3.1	โปรแกรมส่งข้อมูลออกพอร์ตนาน	41
3.1.1	ฟังก์ชัน cportopen	42
3.1.2	ฟังก์ชัน cportconfig	42
3.1.3	ฟังก์ชัน cportwrite	42
3.2	โปรแกรมการแปลงซีเม็กซ์-ไฟล์ให้เป็นไฟล์ชนิดไดนามิกคอลเลกชันไลบรารี	43
3.2.1	การกำหนดและติดตั้งคอมไพเลอร์	43
3.2.2	การสร้างไฟล์ชนิดไดนามิกคอลเลกชันไลบรารี	43
3.3	โปรแกรมรู้จำเสียงพูด	44
3.3.1	ฟรีโพเซตซิ่ง	45
3.3.2	การหาค่าพารามิเตอร์	55
3.4	โปรแกรมการแปลงเอ็ม-ไฟล์เป็นไฟล์ชนิดเอ็กซ์คิวทีเอเบิล	63
3.5	โปรแกรมรับโทรศัพท์อัตโนมัติ	63
3.6	วงจรแสดงผลออกทางพอร์ตนาน	64
บทที่ 4	การทดลองและผลการทดลอง	66
4.1	การทดลองที่ 1	66
	ผลการทดลอง	66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองที่ 2	71
ผลการทดลอง	71
4.3 การทดลองที่ 3	76
ผลการทดลอง	76
บทที่ 5 บทวิจารณ์และบทสรุป	78
5.1 บทวิจารณ์	78
5.1.1 ส่วนของโปรแกรมรู้จำเสียงพูด	78
5.1.2 ส่วนของโมเด็ม	78
5.1.3 ส่วนของการส่งข้อมูลออกพอร์คขนาน	79
5.2 บทสรุป	79

ภาคผนวก

กิตติกรรมประกาศ

หนังสืออ้างอิง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปลูกภาพ

หน้า

รูปที่ 2.1	หลักการทํางานของการควบคุมเครื่องใช้ไฟฟ้าในบ้านด้วยเสียงพูดผ่านโทรศัพท์	2
รูปที่ 2.2	บล็อกไดอะแกรมของโปรแกรมการควบคุมเครื่องใช้ไฟฟ้าในบ้านด้วยเสียงพูดผ่านโทรศัพท์	3
รูปที่ 2.3	บล็อกไดอะแกรมในการสร้างไฟล์แบบเอ็กซ์คิวทีแอบิลจากเอ็ม-ไฟล์	5
รูปที่ 2.4	หลักการรู้จำเสียงพูด	6
รูปที่ 2.5	การกระจายของค่าจุดตัดศูนย์ของเสียงไม่ก้องและเสียงก้อง	9
รูปที่ 2.6	การแบ่งช่วงที่ใช้ในการวิเคราะห์	10
รูปที่ 2.7	บล็อกไดอะแกรมโมเดลการสร้างสัญญาณเสียงพูดอย่างง่าย	14
รูปที่ 2.8	แสดงให้เห็นลักษณะของฟังก์ชันแรบปีง	28
รูปที่ 2.9	เงื่อนไขการเปลี่ยนความชันของฟังก์ชันแรบปีง	30
รูปที่ 2.10	บล็อกไดอะแกรมการทํางานของซีเม็กซ์-ไฟล์	36
รูปที่ 2.11	พอร์ตชานานของคอมพิวเตอร์	37
รูปที่ 3.1	โพลวซาร์ทของโปรแกรมการทํางานของการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์	39
รูปที่ 3.2	บล็อกไดอะแกรมแสดงการสร้างโปรแกรมการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์	40
รูปที่ 3.3	บล็อกไดอะแกรมการทํางานของฟังก์ชันทั้งหมดในการส่งข้อมูลออกพอร์ตชานาน	42
รูปที่ 3.4	แสดงบล็อกไดอะแกรมของระบบการรับรู้เสียงพูด	44
รูปที่ 3.5	ฟรีโพรเซสซิง	45
รูปที่ 3.6	แสดงขั้นตอนการเตรียมสัญญาณในการวิเคราะห์	46
รูปที่ 3.7	แสดงขั้นตอนการเตรียมสัญญาณในการวิเคราะห์	47
รูปที่ 3.8	แสดงโพลวซาร์ทการหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณ โดยการเปรียบเทียบค่าพลังงาน	48
รูปที่ 3.9	แสดงขั้นตอนการหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงโดยวิธีการเปรียบเทียบอัตราการตัดศูนย์	50
รูปที่ 3.10	แสดงโพลวซาร์ทการหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงโดยวิธีการเปรียบเทียบอัตราการตัดศูนย์	51
รูปที่ 3.11	แสดงขั้นตอนการนอร์มอลไลซ์สัญญาณ	54
รูปที่ 3.12	แสดงโพลวซาร์ทการนอร์มอลไลซ์สัญญาณ	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.13	แสดงโฟลวชาร์ทการหาค่าสัมประสิทธิ์การประมาณเชิงเส้น	56
รูปที่ 3.14	แสดงโฟลวชาร์ทการหาค่าสัมประสิทธิ์เซปสตรีม	59
รูปที่ 3.15	แสดงโฟลวชาร์ทการหาค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนัก	60
รูปที่ 3.16	แสดงขั้นตอนไดนามิกไทม์วาร์ปปีง	61
รูปที่ 3.17	แสดงแผนผังการทำไดนามิกไทม์วาร์ปปีง	62
รูปที่ 3.18	วงจรแสดงผลออกทางพอร์ตขนาน	64
รูปที่ 3.19	วงจรแสดงผลออกทางพอร์ตขนานเมื่อทำการต่อเสร็จเรียบร้อยแล้ว	65
รูปที่ 4.1	แสดงสัญญาณเสียงหนึ่ง	67
รูปที่ 4.2	แสดงสัญญาณเสียงสอง	68
รูปที่ 4.3	แสดงสัญญาณเสียงสาม	69
รูปที่ 4.4	แสดงสัญญาณเสียงสี่	70
รูปที่ 4.5	สัญญาณเสียงหนึ่งเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบค่าพลังงาน	67
รูปที่ 4.6	สัญญาณเสียงหนึ่งเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบอัตราการตัดศูนย์	67
รูปที่ 4.7	สัญญาณเสียงสองเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบค่าพลังงาน	68
รูปที่ 4.8	สัญญาณเสียงสองเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบอัตราการตัดศูนย์	68
รูปที่ 4.9	สัญญาณเสียงสามเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบค่าพลังงาน	69
รูปที่ 4.10	สัญญาณเสียงหนึ่งเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบอัตราการตัดศูนย์	69
รูปที่ 4.11	สัญญาณเสียงสี่เมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบค่าพลังงาน	70
รูปที่ 4.12	สัญญาณเสียงสี่เมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบอัตราการตัดศูนย์	70
รูปที่ 4.13	สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงหนึ่ง	72
รูปที่ 4.14	สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงหนึ่ง	72
รูปที่ 4.15	สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงหนึ่ง	72
รูปที่ 4.16	สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงสอง	73
รูปที่ 4.17	สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงสอง	73
รูปที่ 4.18	สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสอง	73
รูปที่ 4.19	สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงสาม	74
รูปที่ 4.20	สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงสาม	74
รูปที่ 4.21	สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสาม	74
รูปที่ 4.22	สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงสี่	75
รูปที่ 4.23	สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงสี่	75
รูปที่ 4.24	สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสี่	75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงการทำงานและชนิดของพินของพอร์ตขนาน	37
ตารางที่ 4.1 แสดงเปอร์เซ็นต์ความถูกต้องของการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์	76



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

โลกเราในยุคปัจจุบันนี้นับได้ว่าเป็นยุคแห่งโลกาภิวัตน์ ชีวิตของมนุษย์ในสังคมทุกวันนี้ต่างก็มีงาน,หน้าที่ และภาระต่างๆมากมาย แต่สำหรับเครื่องใช้ไฟฟ้านั้น มนุษย์ก็ยังคงต้องทำการปิด-เปิดควบคุมการทำงานด้วยการกดสวิทช์ที่ตัวเครื่องอยู่ดี แม้ว่าในปัจจุบันจะมีการนำรีโมทคอนโทรล (Remote Control) มาใช้ก็ตาม แต่การใช้รีโมทคอนโทรลนั้นก็ยังมีข้อจำกัดอยู่ที่ ระยะทางระหว่างตัวเครื่องใช้ไฟฟ้ากับตัวรีโมทคอนโทรลเช่นกัน ปัญหาเหล่านี้ทำให้เกิดความไม่สะดวกในการควบคุมเครื่องใช้ไฟฟ้า ในยุคปัจจุบันที่เวลาเป็นเงินเป็นทองเช่นนี้

ดังนั้นเพื่อแก้ปัญหาที่ว่าถ้าผู้ใช้อยู่ห่างจากตัวเครื่องใช้ไฟฟ้าเป็นระยะทางมากกว่าขีดจำกัดของรีโมทคอนโทรล และมีความต้องการที่จะปิด-เปิดควบคุมการทำงานของเครื่องใช้ไฟฟ้าแล้วจะทำได้อย่างไร

จึงเป็นแนวคิดให้เกิดการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์ขึ้น

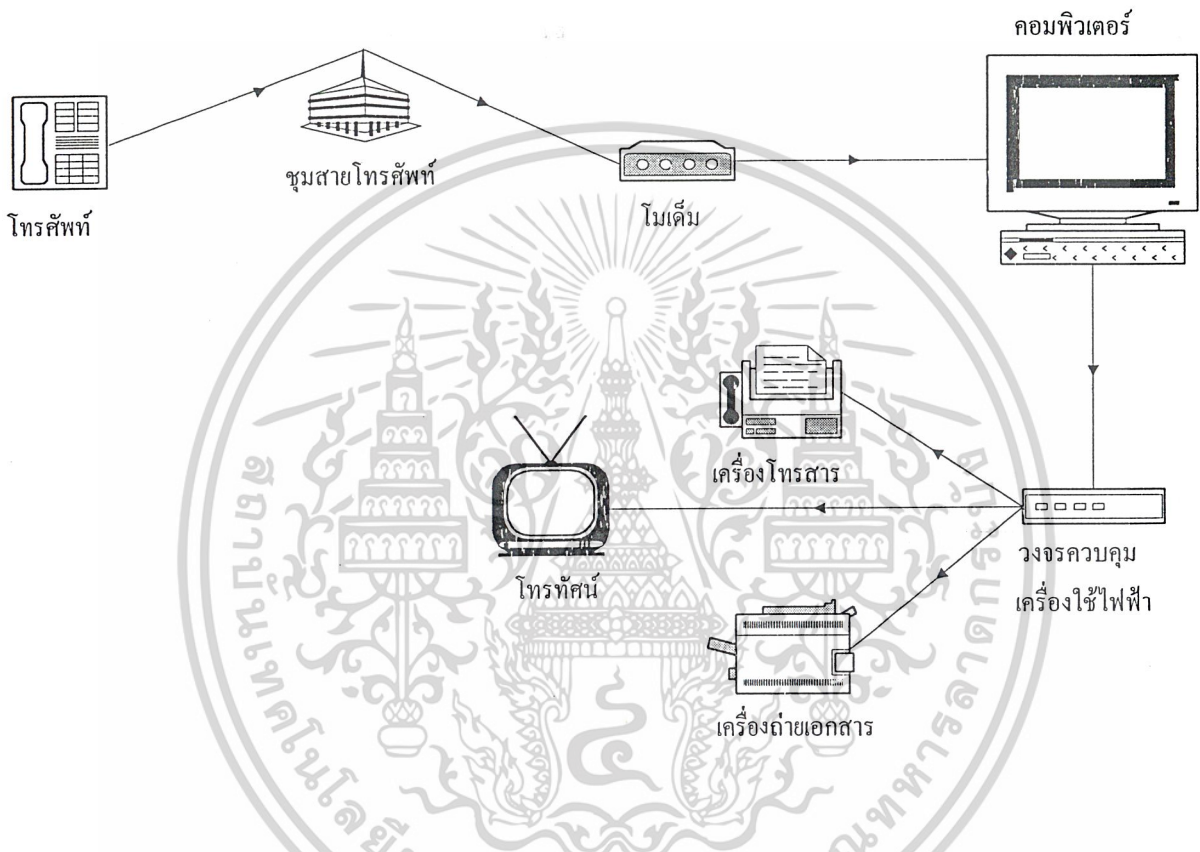
การควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์จะอำนวยความสะดวกให้แก่ผู้ใช้อย่างมาก โดยที่ผู้ใช้สามารถโทรศัพท์จากนอกบ้านเข้ามาในบ้านแล้วสั่งการควบคุมให้เครื่องใช้ไฟฟ้าในบ้านปิด-เปิดได้ด้วยตนเอง ในปฏิญานิพนธ์นี้จะใช้การรู้จำเสียงพูดแบบผู้พูดคนเดียว ซึ่งจะมีประโยชน์ที่ว่าบุคคลอื่นที่ไม่ใช่เจ้าของบ้านจะไม่สามารถโทรศัพท์เข้ามาสั่งงานเครื่องใช้ไฟฟ้าในบ้านได้ถูกต้อง

ในปฏิญานิพนธ์นี้จะใช้การคิด-ดับของแอลอีดี (LED) ท่อนการเปิด-ปิดของเครื่องใช้ไฟฟ้า เนื่องจากให้ความสะดวกในการสร้าง และการทดลองให้เห็นผลมากกว่าเครื่องใช้ไฟฟ้าจริงๆ

บทที่ 2

ทฤษฎีหรือหลักการ

การควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านโทรศัพท์ที่มีหลักการทำงานดังรูปที่ 2.1

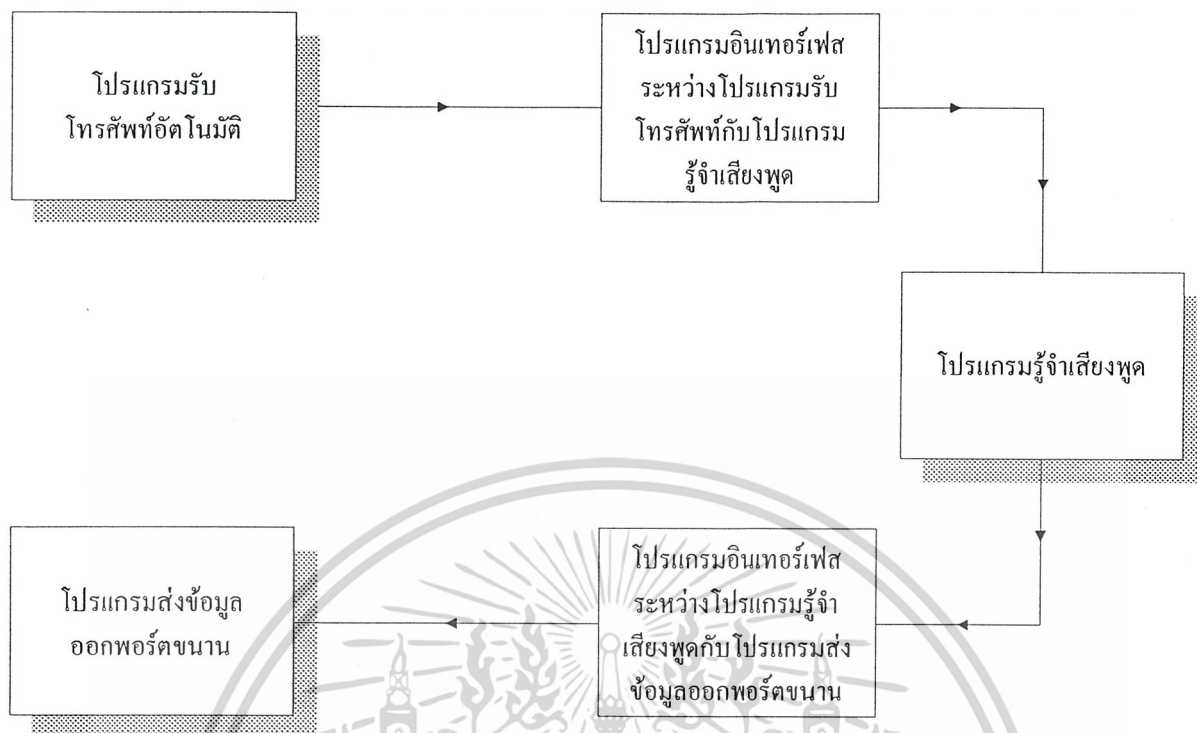


รูปที่ 2.1 หลักการทำงานของ การควบคุมเครื่องใช้ไฟฟ้าในบ้านด้วยเสียงพูดผ่านโทรศัพท์

จากรูปที่ 2.1 ผู้ใช้งานจะโทรศัพท์จากนอกบ้านเข้ามาในบ้าน วอยส์โมเด็ม (Voice Modem) จะคอยรับโทรศัพท์ที่อัตโนมัติ จากนั้นจะส่งไฟล์เสียงสั่งงานเครื่องใช้ไฟฟ้าที่อัดได้ไปยังคอมพิวเตอร์ เพื่อทำการรู้จำเสียงพูด ผลที่ได้รับจากโปรแกรมรู้จำเสียงพูดในคอมพิวเตอร์ จะถูกส่งออกไปยัง วงจรควบคุมเครื่องใช้ไฟฟ้าภายในบ้าน เพื่อควบคุมเครื่องใช้ไฟฟ้าต่างๆต่อไป

ในส่วนของโปรแกรมที่ใช้ในการควบคุมเครื่องใช้ไฟฟ้าในบ้านด้วยเสียงพูดผ่านโทรศัพท์จะมีบล็อกไดอะแกรมดังรูปที่ 2.2 นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 บล็อกโคแอมแกรมของโปรแกรมการควบคุมเครื่องใช้ไฟฟ้าในบ้านด้วยเสียงพูดผ่านโทรศัพท์
 ปรึชญานีพนธ์นี้เราจะเน้นไปที่ตัวโปรแกรมต่างๆที่ใช้ในการควบคุมเครื่องใช้ไฟฟ้าในบ้านด้วยเสียงพูด
 ผ่านโทรศัพท์ที่คั้งนั้นจากรูปที่ 2.2 เราจะสามารแบ่งโปรแกรมทั้งหมดที่เราใช้ออกเป็นส่วนๆได้คั้งนี้

- โปรแกรมรับโทรศัพท์อัตโนมัติ
- โปรแกรมอินเทอร์เฟซ (Interface) ระหว่างโปรแกรมรับโทรศัพท์อัตโนมัติและโปรแกรมรู้จำเสียงพูด
- โปรแกรมรู้จำเสียงพูด
- โปรแกรมอินเทอร์เฟซระหว่างโปรแกรมรู้จำเสียงพูดกับโปรแกรมส่งข้อมูลออกพอร์ตนาน(Parallel Port)
- โปรแกรมส่งข้อมูลออกทางพอร์ตนาน

2.1 โปรแกรมรับโทรศัพท์อัตโนมัติ

เป็นคอมโพเนนท์ (Component) ที่เขียนขึ้นมาจากภาษาเดลไฟ (Delphi) ใช้ในการรอร์บโทรศัพท์อัตโนมัติ
 และบันทึกเสียงที่ผู้โทรศัพท์พูดสั่งงานเครื่องใช้ไฟฟ้าเข้ามา โดยอาศัยความสามารถของโมเด็มภายนอก (External
 Modem) มาช่วยในการรับโทรศัพท์และบันทึกเสียงพูดสั่งงานเครื่องใช้ไฟฟ้าที่โทรศัพท์เข้ามา

2.2 โปรแกรมอินเทอร์เฟซระหว่างโปรแกรมรับโทรศัพท์อัตโนมัติกับโปรแกรมรู้จำเสียงพูด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมนี้จะทำการคอมไพล์ (Compile) เอ็มไฟล์ (M-File) หรือโปรแกรมที่เขียนด้วยเมทแลบ (MATLAB) ซึ่งจะมีนามสกุลของไฟล์เป็น .m ให้เป็นไฟล์แบบเอ็กซีคิวทีเอเบิล (Executable) ที่มีนามสกุลของไฟล์เป็น .exe เพื่อให้เดสก์ท็อปสามารถเรียกใช้โปรแกรมรู้อำนาจเสียงพูดไปใช้ได้ทันทีที่คอม โพนเนทท์ที่ทำหน้าที่รับโทรศัพท์อัตโนมัติทำงานเสร็จลง

ซึ่งในการคอมไพล์เอ็มไฟล์นี้เราจะใช้โปรแกรมที่เรียกว่าเมทแลบคอมไพเลอร์ (MATLAB Compiler) ซึ่งอธิบายการทำงานได้ดังนี้

2.2.1 เมทแลบคอมไพเลอร์

เมทแลบคอมไพเลอร์เป็นโปรแกรมที่ใช้เอ็มไฟล์เป็นอินพุท (Input) และ ผลิตซอร์สโค้ดภาษาซี หรือซี

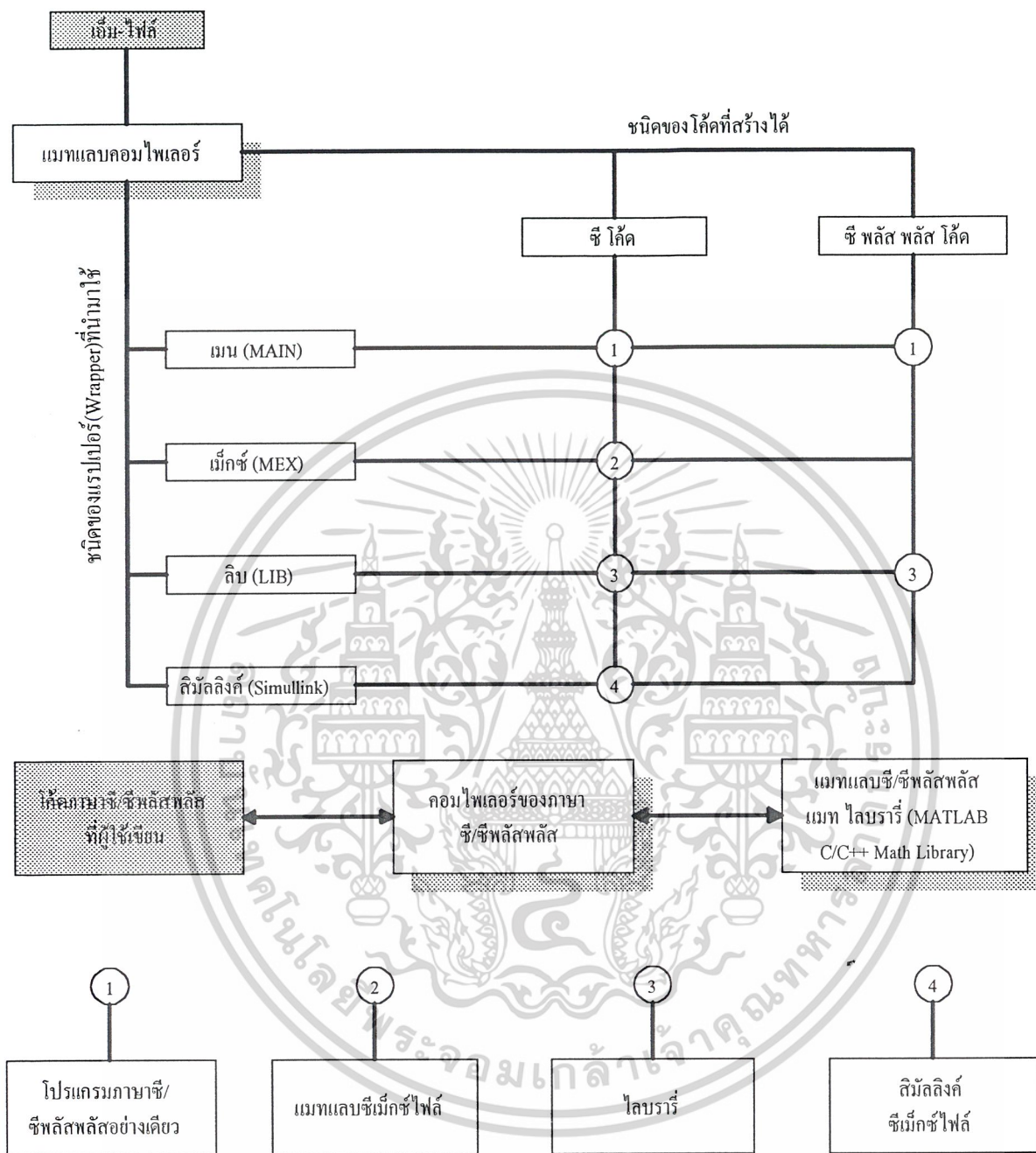
พลัสพลัส (C or C++ Source Code) ออกมาเป็นเอาต์พุท (Output) และจากซอร์สโค้ดภาษาซีที่เมทแลบคอมไพล์ผลิตได้นี้ ยังสามารถนำไปสร้างชีแชรไลบรารี (C Shared Libraries) และ ไฟล์แบบเอ็กซีคิวทีเอเบิลได้อีกด้วย

การสร้างไฟล์แบบเอ็กซีคิวทีเอเบิล จะเริ่มด้วย การสร้างโปรแกรม ภาษาซี / ซีพลัสพลัส ออกมาเป็นชีแชรไลบรารีก่อน จากการใช้ตัวเลือก (Option Flag) ที่เหมาะสม (-m) กับเอ็ม-ไฟล์ที่เราเขียนไว้ หลังจากนั้นจะนำซอร์สโค้ดภาษาซี/ซีพลัสพลัสที่ได้มารวมกับแรปเปอร์ไฟล์ (Wrapper File) ซึ่งเป็นไฟล์ที่บรรจุไปด้วยอินเทอร์เฟซที่จำเป็นระหว่างซอร์สโค้ดภาษาซี/ซีพลัสพลัสที่ได้กับไฟล์แบบเอ็กซีคิวทีเอเบิลที่เราต้องการ

เราต้องเลือกชนิดของแรปเปอร์ไฟล์ให้ตรงกับ ไฟล์แบบเอ็กซีคิวทีเอเบิลที่เราต้องการจะสร้างนั้นๆ ผลของการรวมกันก็จะออกมาเป็นไฟล์แบบเอ็กซีคิวทีเอเบิลที่เราต้องการ

หลักการสร้างไฟล์แบบเอ็กซีคิวทีเอเบิลแสดงไว้ในรูปที่ 2.3

จากรูปที่ 2.3 เริ่มจากการเขียนโปรแกรมด้วยภาษาเมทแลบทำให้ได้เอ็ม-ไฟล์ออกมา หลังจากนั้นเมทแลบคอมไพเลอร์จะทำการแปลงเอ็ม-ไฟล์ที่ได้ไปเป็นซอร์สโค้ดภาษาซี/ซีพลัสพลัส จากนั้นนำซอร์สโค้ดภาษาซี/ซีพลัสพลัสที่ได้ไปรวมกับแรปเปอร์ไฟล์ที่ต้องการ ไฟล์ที่เกิดจากการรวมกันของซอร์สโค้ดภาษาซี/ซีพลัสพลัสกับแรปเปอร์ไฟล์นี้จะถูกนำไปคอมไพล์ด้วยคอมไพเลอร์ของภาษาซี/ซีพลัสพลัส ซึ่งคอมไพเลอร์จะทำการรวมและเชื่อมต่อกับไลบรารีที่เหมาะสม ทำให้เกิดไฟล์แบบเอ็กซีคิวทีเอเบิลที่เราต้องการ



ชนิดของไฟล์ที่ต้องการ

- บล็อกที่เข้มคือ โค้ดที่ใช้เป็นโครงสร้างขึ้น
- บล็อกที่มีเงาคืออุปกรณ์ของเมทเลบ (MATLAB Tool)
- บล็อกที่ไม่มีสีอะไรคือ โค้ดที่เมทเลบได้ทำการสร้างขึ้น

รูปที่ 2.3 บล็อกโคดอะแกรมในการสร้างไฟล์แบบเอ็กซ์คิวทีแอบิลจากเอ็มไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 โปรแกรมรู้จำเสียงพูด

การทำงานของระบบรู้จำเสียงพูดจะประกอบไปด้วยส่วนของการเตรียมสัญญาณ(Signal Preprocessing) ส่วนของการรู้จำเสียงพูด แยกเป็นรายละเอียดได้ดังนี้

การเตรียมสัญญาณ(Signal Preprocessing)

- พรีเมมฟาสิส (Preemphasis)
- การแบ่งช่วงของสัญญาณ (Frame Blocking)
- การวินโดว์ (Windowing) ด้วยแฮมมิงวินโดว์ (Hamming Window)
- การทำอัตโนมัติสัมพันธ์ลำดับ 0 (Autocorrelation Order 0) เพื่อหาสเปกตรัมความหนาแน่นของกำลังงาน (Power Density Spectrum)
- การหาขอบเขตของคำ (Endpoint Detection)

การหาค่าพารามิเตอร์ (Feature Measurement)

- การวิเคราะห์การประมาณพหุเชิงเส้น (LPC Analysis)
- การแปลงสัมประสิทธิ์การประมาณพหุเชิงเส้นเป็นสัมประสิทธิ์เซปสตรัม(LPC Parameter Conversion to Cepstrum Coefficient)
- พารามิเตอร์เวทจัด (Parameter Weighting)

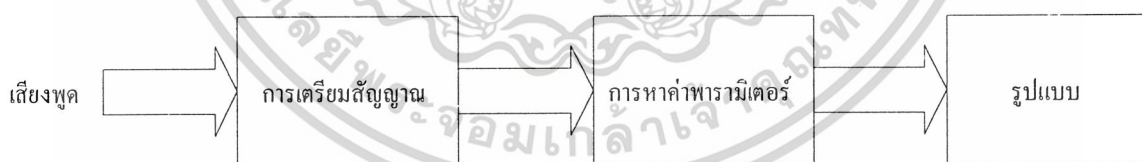
การเทียบกับรูปแบบอ้างอิง (Pattern Classification)

- ไดนามิกไทม์วาร์ปิ้ง (Dynamic Time Warping : DTW)

กฎการตัดสินใจ (Decision Rule)

- เคเอ็นเอ็น (K – Nearest Neighbor : KNN)

ในการรู้จำเสียงพูดนั้นเราจะมีหลักการดังรูปที่ 2.4



รูปที่ 2.4 หลักการรู้จำเสียงพูด

1. การเตรียมสัญญาณ (Preprocessing) เป็นการเตรียมสัญญาณเสียงให้พร้อมสำหรับการทำกระบวนการรู้จำเสียงพูด โดยการทำให้พรีเมมฟาสิสจากนั้นนำมาทำการแบ่งเฟรม แล้วนำไปทำการวินโดว์เพื่อให้สัญญาณมีลักษณะที่ต่อเนื่อง จากนั้นนำไปแยกสัญญาณเสียงพูดออกจากสัญญาณเสียงรบกวน (noise) ด้วยการหาขอบเขตของคำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะที่ต่อเนื่อง จากนั้นนำไปแยกสัญญาณเสียงพูดออกจากสัญญาณเสียงรบกวน (noise) ด้วยการหาขอบเขตของค่า

2. การหาค่าพารามิเตอร์ (Feature Measurement) เป็นการหารูปแบบทดสอบ (Test Pattern) จากสัญญาณอินพุท วิธีการทำมีหลายวิธี ในโครงการนี้เราจะเลือกใช้สัมประสิทธิ์การประมาณพันระเชิงเส้น (Linear Predictive Coding Coefficient), สัมประสิทธิ์เซปสตรัม (Cepstrum Coefficient), สัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก (Weighted Cepstrum Coefficient), สัมประสิทธิ์เซปสตรัม (Cepstrum Coefficient) ร่วมกับ เซปสตรัมเดอริเวทีฟ (Cepstrum Derivative), และสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก (Weighted Cepstrum Coefficient) ร่วมกับเซปสตรัมเดอริเวทีฟ (Cepstrum Derivative)

3. รูปแบบอ้างอิง (Reference Pattern) เป็นการจัดการและจัดกลุ่มเสียงของรูปแบบทดสอบ (Test Pattern) มาเก็บไว้อ้างอิง ผลที่ได้ก็คือรูปแบบอ้างอิง (Reference Pattern) ซึ่งอาจเป็น ตัวอย่าง (exemplar), แบบเทียบ (template), หรือแบบจำลอง (model) ก็ได้

4. การหาความเหมือนของรูปแบบ (Pattern Similarity) ในขั้นตอนนี้จะทำการวัดระยะทาง (Distance) เปรียบเทียบระหว่างรูปแบบทดสอบ (Test Pattern) ซึ่งเราไม่รู้จัก เทียบกับรูปแบบอ้างอิง (Reference Pattern) ที่เก็บไว้เพื่อดูความคล้าย (Similarity)

5. การตัดสินใจ (Decision logic) ค่าที่ได้จากขั้นที่ 3 จะถูกนำมาตัดสินใจว่ารูปแบบอ้างอิง (Reference Pattern) ไດเหมาะสมกับรูปแบบทดสอบ (Test Pattern) มากที่สุด

ก่อนที่เราจะกล่าวถึงทฤษฎีในการรับรู้เสียงพูดเราจะกล่าวถึงการวิเคราะห์ในโดเมนเวลาซึ่งเป็นพื้นฐานในการรับรู้เสียงพูดต่อไป

2.3.1 การวิเคราะห์ในโดเมนเวลา

การวิเคราะห์ในโดเมนเวลา คือ วิธีการประมวลสัญญาณเสียงพูดโดยยึดหลักการวิเคราะห์จากรูปคลื่นสัญญาณที่แปรตามแกนเวลาซึ่งมีอยู่หลายวิธีด้วยกัน ผลการวิเคราะห์สัญญาณเสียงในโดเมนเวลา เช่น ค่าพลังงาน (acoustic energy) ของเสียง ค่าจุดตัดศูนย์ (Zero Crossing) เป็นต้น วิธีต่าง ๆ ที่เป็นพื้นฐานในการวิเคราะห์สัญญาณเสียงในโดเมนเวลาบางวิธีจะกล่าวดังต่อไปนี้

2.3.1.1 การหาค่าพลังงานของสัญญาณเสียง

พลังงานของสัญญาณเป็นตัวแทนอันหนึ่งที่เรามักจะนำมาใช้ในการวิเคราะห์ลักษณะต่าง ๆ ของสัญญาณทั่ว ๆ ไป โดยพลังงานของสัญญาณ $s(n)$ ไດ ๆ ที่แปรตามเวลาสามารถนิยามได้ว่า

$$E = \sum_{n=-\alpha}^{+\alpha} s^2(n) \quad (2.1)$$

แต่สำหรับสัญญาณเสียงซึ่งเป็นสัญญาณที่แปรเปลี่ยนอยู่ตลอด ไม่มีเสถียรภาพตามเวลา เราจะต้องแบ่งสัญญาณออกมาพิจารณาเป็นช่วงเล็ก ๆ ตามแกนเวลาหรือเรียกว่าแบ่งออกเป็นเฟรม เช่น เฟรมละประมาณ 10-30 วินาที ดังนั้นก็จะสามารถหาพลังงานของเสียงในแต่ละเฟรมได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$E(m) = \sum_{n=0}^{N-1} [w(m) s(m-n)]^2 \quad (2.2)$$

โดยที่ $w(m)$ คือ วินโดว์ฟังก์ชัน (windows function) ที่ใช้กำหนดรูปร่างในการพิจารณาของ สัญญาณเสียง $s(n)$ ในหนึ่งเฟรม และ N คือจำนวนตัวอย่างของสัญญาณเสียงที่อยู่ในเฟรมหรือภายในกรอบของ วินโดว์ฟังก์ชัน

การวัดค่าพลังงานดังในสมการที่ (2.2) นั้นมีข้อจำกัดตรงที่ว่ามันจะมีความไวต่อสัญญาณที่มีขนาดใหญ่ ๆ เนื่องจากเราใช้วิธียกกำลังสองค่าของสัญญาณอินพุต ดังนั้นการแก้ปัญหาอย่างหนึ่งก็คือวัดพลังงานของเสียง โดยใช้สมการดังนี้

$$E(m) = \sum_{n=0}^{N-1} [w(m) s(m-n)] \quad (2.3)$$

นั่นคือ การใช้ผลบวกของค่าสัมบูรณ์หรือเรียกว่าค่าแมกนิจูด (magnitude) แทนผลรวมของค่ายกกำลังสอง

2.3.1.2 การหาค่าจุดตัดศูนย์ (Zero Crossing)

การเกิดจุดตัดศูนย์จะเกิดขึ้นจากการที่รูปคลื่นของสัญญาณมีการตัดกับแกนเวลานั้น คือค่าของสัญญาณ จะมีการเปลี่ยนสัญลักษณ์ทางคณิตศาสตร์ (Algebraic Sign) นั่นเอง อัตราการเกิดจุดตัดศูนย์เป็นเครื่องมืออย่าง ง่ายที่ใช้อธิบายการเปลี่ยนแปลงข้อมูลของสัญญาณค่าจุดตัดศูนย์นั้นสามารถนำมาใช้ในการตัดสินใจว่าสัญญาณเสียง นั้นเป็นเสียงก้อง (Voiced) หรือเสียงไม่ก้อง (Unvoiced) เนื่องจากเสียงก้องส่วนใหญ่จะมีค่าพลังงานอยู่ในช่วง ความถี่ต่ำ ส่วนเสียงไม่ก้องจะมีพลังงานอยู่ในช่วงความถี่สูง และค่าจุดตัดศูนย์ก็มีความสัมพันธ์โดยตรงกับค่า ความถี่ของสัญญาณ ดังนั้นจึงอาจสรุปได้ว่า สัญญาณเสียงที่มีค่าจุดตัดศูนย์สูงจะเป็นเสียงไม่ก้อง และสัญญาณที่มีค่าจุดตัดศูนย์ต่ำจะเป็นเสียงก้อง แต่อย่างไรก็ตามการกำหนดขนาดของค่าจุดตัดศูนย์ที่แน่นอนเพื่อจำแนกชนิด ของเสียงนั้น จะต้องอาศัยผลจากการทดลองเป็นหลัก ในรูปที่ 2.16 จะแสดงให้เห็นถึงการกระจายของค่าจุดตัด ศูนย์ของเสียงไม่ก้อง และเสียงก้อง

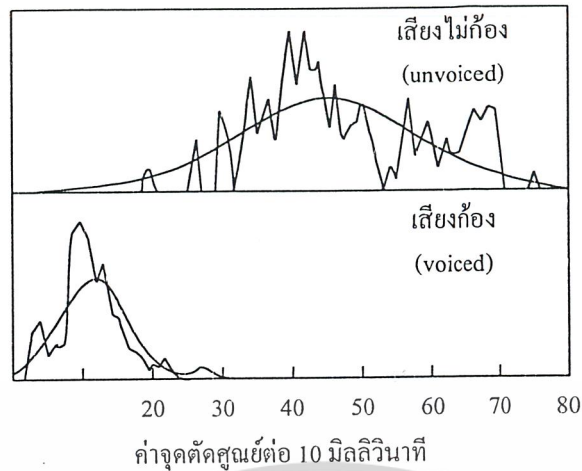
จุดตัดศูนย์สามารถนิยามได้ดังสมการที่ (2.4)

$$Z_s = \int \left\{ \frac{|\operatorname{sgn}\{s(n)\} - \operatorname{sgn}\{(n-1)\}|}{2} \right\} \quad (2.4)$$

เมื่อ

$$\operatorname{sgn}\{s(n)\} = \begin{cases} +1 & s(n) \geq 0 \\ -1 & s(n) < 0 \end{cases} \quad (2.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 การกระจายของค่าจุดตัดศูนย์ของเสียงไม่ก้อง และเสียงก้อง

ในการวิเคราะห์สัญญาณเสียง จำเป็นต้องแบ่งสัญญาณออกมาพิจารณาเป็นเฟรมตามแกนเวลา ดังนั้นเวลาในการหาค่าจุดตัดศูนย์ของสัญญาณในแต่ละเฟรมจะเป็นดังสมการที่ (2.6)

$$Z_s(m) = \frac{1}{N} \sum_{n=m-N+1}^m \frac{|\text{sgn}\{s(n)\} - \text{sgn}\{s(n-1)\}|}{2} w(m-n) \quad (2.6)$$

โดยที่ $w(n)$ คือวินโดว์ฟังก์ชัน (windows function) ที่ใช้กำหนดรูปร่างในการพิจารณาของสัญญาณเสียง $s(n)$ ในหนึ่งเฟรม และ N คือจำนวนตัวอย่างของสัญญาณเสียงที่อยู่ในเฟรมหรือภายในกรอบของวินโดว์ฟังก์ชัน

2.3.2 การเตรียมสัญญาณ

2.3.2.1 การพรีเอมฟาซิส

เนื่องจากสัญญาณเสียงพูด จะมีองค์ประกอบส่วนใหญ่อยู่บริเวณความถี่ต่ำ ดังนั้นเพื่อให้อัตราส่วนสัญญาณเสียงต่อสัญญาณรบกวน มีค่าค่อนข้างคงที่ เราจึงต้องมีการพรีเอมฟาซิส โดยเน้นความถี่สูงให้มีขนาดสูงขึ้น นั่นคือ การพรีเอมฟาซิสเป็นการกรองสัญญาณด้วยวงจรกรองความถี่สูงผ่าน (High pass filter) ซึ่งมักนิยมใช้วงจรกรองอันดับหนึ่ง มีฟังก์ชันถ่ายโอนดังนี้

$$H(z) = 1 - az^{-1} \quad 0.9 \leq a \leq 1.0 \quad (2.7)$$

สำหรับสัญญาณที่ผ่านการพรีเอมฟาซิสนี้ จะมีเอาต์พุตของสัญญาณ $s'(n)$ ซึ่งมีความสัมพันธ์กับสัญญาณอินพุต $s(n)$ โดยสามารถเขียนสมการได้ดังนี้

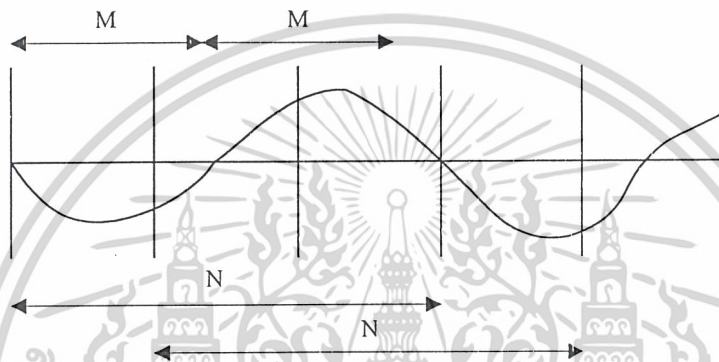
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$s'(n) = s(n) - as(n-1) \quad (2.8)$$

ยิ่งค่า a มีค่าใกล้ 1 เท่าใด ความถี่สูงก็จะถูกขยายมากขึ้นเท่านั้น ค่า a ที่นิยมใช้ในการหาค่าสัมประสิทธิ์การประมาณพันธะเชิงเส้น คือ $a = 15/16 = 0.9375$

2.3.2.2 การแบ่งช่วงสัญญาณ

สัญญาณที่ผ่านการพรีเอมฟาซิสแล้ว $s'(n)$ จะถูกแบ่งออกเป็นช่วง ๆ หรือเฟรมโดยมีช่วงละ N ตัวอย่างสัญญาณ การวิเคราะห์จะวิเคราะห์ทีละช่วงของแต่ละ N ตัวอย่างของสัญญาณ ดังรูปที่ 2.6



รูปที่ 2.6 การแบ่งช่วงที่ใช้ในการวิเคราะห์

โดยช่วงในการวิเคราะห์แต่ละช่วงจะถูกเลื่อนไปเป็นระยะ M ช่วงสัญญาณ จะเห็นได้ว่า ถ้าค่า M ใหญ่กว่า N ในการเลื่อนของช่วงในการวิเคราะห์จะทำให้บางสัญญาณไม่ถูกใช้ในการวิเคราะห์ ก็จะเป็นการสูญเสียส่วนหนึ่งทำให้ผลที่ได้ไม่ถูกต้องเท่าที่ควร ถ้าค่า M เล็กกว่า N จะทำให้ตัวอย่างของสัญญาณทุกตัวถูกนำมาวิเคราะห์ ยิ่งค่า M เล็กเท่าใด ความแม่นยำในการวิเคราะห์ก็จะยิ่งสูงขึ้นเท่านั้นแต่จะทำให้การคำนวณช้าลง

ในการวิเคราะห์หาค่าสัมประสิทธิ์การประมาณพันธะเชิงเส้น ถ้าทำการกำหนดให้สัญญาณเสียงที่นำมาวิเคราะห์ เมื่อผ่านการพรีเอมฟาซิสแล้ว มีการแบ่งช่วงเป็นช่วง ๆ โดยมีอันดับของช่วงเป็น l^{th} โดยสัญญาณในแต่ละเฟรมเป็น $x_l(n)$ และมีจำนวนช่วงทั้งหมดเป็น L เฟรม ดังนั้น

$$x_l(n) = s'(Ml+n) \quad n = 0, 1, \dots, N-1 \quad l = 0, 1, \dots, L-1$$

และทำการกำหนดให้ค่า $N = 240$ ตัวอย่าง และ $M = 80$ ตัวอย่าง ตามลำดับ เมื่อมีการสุ่มตัวอย่างของสัญญาณที่ 8 กิโลเฮิร์ตซ์

2.3.2.3 การทำวินโดว์

ในขั้นนี้เราจะลดความไม่ต่อเนื่องของสัญญาณในบริเวณเริ่มต้น และท้ายสุดของแต่ละเฟรมโดยสมการวินโดว์จะทำการเหลาปลายทั้งสองของสัญญาณในแต่ละเฟรมให้เป็นศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั่นก็คือเราจะต้องแบ่งทำการวิเคราะห์หาพารามิเตอร์ของสัญญาณเสียงพูดในช่วงเวลาสั้น ๆ เหมือนมองผ่านช่องแคบ ๆ ที่เรียกว่า ช็อตไทม์วินโดว์ (Shot-time window) เมื่อเทียบตามเวลาที่เสียงจะมีการเปลี่ยนแปลงได้ เพื่อจะได้มองเห็นเหมือนกับว่า เราหาพารามิเตอร์นั้น ๆ ได้มาจากสัญญาณเสียงที่อยู่ภายในช่องแคบ ๆ และมีความเสถียรภายในช่วงเวลาสั้น ๆ

สัญญาณเสียง โดยการขยับช่องแคบไปเป็นคาบ ๆ ตามแกนเวลา ตามปกติเฟรมเรทจะมีค่าประมาณ 2 เท่าของส่วนกลับของขนาดช่องแคบ นั่นก็คือช่องแคบถี่ ๆ กันไปจะมีการซ้อนทับกัน 50%

การนำฟังก์ชันของช่องแคบที่มีช่วงขนาดจำกัด $w(n)$ มาคูณเข้ากับสัญญาณ $s(n)$ จะทำให้ได้กลุ่มตัวอย่างของเสียงพูดที่ถูกกำหนดน้ำหนักให้แปรไปตามรูปร่างของช่องแคบ รูปแบบของช่องแคบที่ง่ายที่สุดก็คือ กรอบแบบสี่เหลี่ยม (rectangular window) ซึ่งมีนิยามดังนี้

$$w(n) = \begin{cases} 1, & n = 0, 1, \dots, N-1 \\ 0, & n \text{ อื่น ๆ} \end{cases} \quad (2.9)$$

ในสมการนี้ คือการกำหนดช่องของการวิเคราะห์ให้มีจำนวนตัวอย่าง N ตัวอย่าง รูปแบบของช่องแคบในลักษณะนี้ ก็มีฟังก์ชันของช่องแคบหลายลักษณะด้วยกัน ตัวอย่างเช่น แบบลิคแมน (Blackman), บาร์เลตต์ (Barlett), แฮมมิง (Hamming), แฮนนิง (Hanning) เป็นต้น โดยฟังก์ชันของช่องแคบที่นิยมใช้กันมากในการวิเคราะห์สัญญาณเสียงก็คือ ฟังก์ชัน แฮมมิง ซึ่งมีรูปตามลักษณะของพัลส์รูปโคไซน์ (cosine pulse) มีนิยามดังนี้

$$w(n) = \begin{cases} 0.54 - 0.46 \cos \left(\frac{2\pi n}{N-1} \right) & n = 0, 1, \dots, N-1 \\ 0 & n \text{ อื่น ๆ} \end{cases} \quad (2.10)$$

2.3.2.4 การหาขอบเขตของคำ

คือ ขั้นตอนในภาคปริโพเซสซิ่ง โดยจะใช้หลักการหาค่าพลังงานของสัญญาณเสียง และการหาอัตราการตัดศูนย์ในโดเมนเวลา

พื้นฐานของสัญญาณในการรับรู้เสียงพูด คือ สัญญาณจะประกอบด้วยส่วนของเสียงพูด อาจมีสัญญาณรบกวนนำหน้าและต่อท้ายด้วย ดังนั้นเมื่อต้องการเฉพาะส่วนของสัญญาณ เราจึงจำเป็นต้องมีวิธีการแบ่งแยกส่วนของสัญญาณให้ถูกต้อง กระบวนการที่แยกส่วนของสัญญาณเสียงพูดออกจากส่วนที่เป็นสัญญาณรบกวน เรียกว่า การหาขอบเขตของสัญญาณ ความจำเป็นของการหาขอบเขตของสัญญาณในการรับรู้เสียงพูด เนื่องจาก

- ความถูกต้องของการรับรู้เสียงพูด จะขึ้น โดยตรงกับความต้องการของการหาขอบเขตของสัญญาณ
- การหาขอบเขตสัญญาณที่ถูกต้อง จะทำให้เวลาที่ใช้ในการคำนวณน้อยที่สุด

พารามิเตอร์พื้นฐานที่ใช้ในการหาขอบเขตของสัญญาณ คือ พลังงานของสัญญาณ โดยการเปรียบเทียบพลังงานแต่ละเฟรม มีวิธีการดังต่อไปนี้

ขั้นตอนที่ 1 ใช้การวิเคราะห์หาค่าพลังงานของสัญญาณในการหาขอบเขตของคำ ซึ่งมีวิธีการดังนี้

1. คำนวณหาค่าพลังงานของสัญญาณในช่วงเวลาที่ถูกบันทึกทั้งหมด โดยการคำนวณหาค่าพลังงานของสัญญาณซึ่งจะทำการคำนวณเป็นเฟรม กำหนดให้เฟรมหนึ่งมีจำนวนแซมเปิล 100 แซมเปิล โดยจะทำการคำนวณเป็นเฟรมต่อไปเรื่อยๆ จนครบทุกแซมเปิลของสัญญาณ
2. พิจารณาค่าพลังงานสูงสุดของสัญญาณ (E_{max}) และดูว่าที่ค่าพลังงานสูงสุดนี้อยู่ที่เฟรมใด แล้วกำหนดให้เป็นเฟรมที่ m
3. ทำการกำหนดพลังงานอ้างอิง (E_{ref}) ซึ่งพลังงานอ้างอิงสามารถคำนวณได้จาก a เท่าของพลังงานสูงสุด ($a \cdot E_{max}$) ซึ่งในที่นี้ค่า a จะมีค่าเท่ากับ 0.2 พลังงานอ้างอิงนี้จะเป็นตัวบอกว่า จุดเริ่มต้นและจุดสุดท้ายของคำจะอยู่นอกช่วงนี้
4. เมื่อกำหนดพลังงานอ้างอิง เป็นที่เรียบร้อยแล้ว ทำการหาจุดเริ่มต้นของคำ โดยเฟรมที่มีค่าพลังงานสูงสุด จะถูกพิจารณาเป็นเฟรมแรก (เฟรมที่ m) ซึ่งนำค่าพลังงานของสัญญาณเฟรมที่ m ไปเปรียบเทียบกับพลังงานอ้างอิง แล้วเลื่อนเฟรมที่จะเปรียบเทียบกับพลังงานกับพลังงานอ้างอิงไปด้านหลังเรื่อยๆ (เฟรมที่ $m-1, m-2, \dots, 1$) จนกว่าจะพบเฟรมที่มีค่าพลังงานน้อยกว่าพลังงานอ้างอิงเป็นเฟรมแรก ก็จะได้จุดเริ่มต้นของสัญญาณ ($a1$) โดยถือว่าค่าพลังงานของสัญญาณค่าแรกที่มีค่าน้อยกว่าพลังงานอ้างอิง (E_{ref}) เป็นจุดเริ่มต้นของคำ
5. การหาจุดสุดท้ายของคำ จะทำการพิจารณาเช่นเดียวกับการจุดเริ่มต้นของคำ คือเริ่มเปรียบเทียบจากเฟรมที่มีพลังงานสูงสุด (เฟรมที่ m) เช่นกันแต่การเลื่อนเฟรมจะเลื่อนไปทางด้านหน้าแทน (เฟรมที่ $m+1, m+2, \dots$) จนกว่าจะพบเฟรมแรกที่มีค่าพลังงานอ้างอิง ก็จะได้จุดสุดท้ายของสัญญาณ ($b1$) โดยถือว่าค่าพลังงานของสัญญาณค่าแรกที่มีค่าน้อยกว่าพลังงานอ้างอิง เป็นจุดสุดท้ายของคำ

ส่วนพารามิเตอร์อีกตัวหนึ่งที่ใช้ในการหาขอบเขตของสัญญาณเช่นกันคือ อัตราการตัดศูนย์ มีวิธีการดังนี้

ขั้นตอนที่ 2 ใช้การวิเคราะห์หาค่าอัตราการตัดศูนย์ สาเหตุที่ใช้วิธีนี้ช่วยเนื่องจากการหาขอบเขตของคำโดยการเปรียบเทียบค่าพลังงานอาจมีข้อผิดพลาดได้ จึงต้องใช้วิธีหาค่าอัตราการตัดศูนย์ (Zero crossing) เข้าช่วยพิจารณา ซึ่งมีวิธีการดังนี้

1. คำนวณหาค่าอัตราการตัดศูนย์ในช่วงเวลาที่ถูกบันทึกทั้งหมด โดยการคำนวณหาค่าอัตราการตัดศูนย์จะคำนวณเป็นเฟรม กำหนดให้แต่ละเฟรมมีตัวอย่าง 100 ตัวอย่าง และทำการคำนวณจนครบช่วงเวลาของเสียงที่ถูกบันทึกทั้งหมด
2. พิจารณาค่าอัตราการตัดศูนย์สูงสุด (Z_{max}) กำหนดให้เป็นเฟรมที่ p

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กำหนดค่าอัตราการตัดศูนย์อ้างอิง (Z_{ref}) ซึ่งจะมีค่าเป็น a เท่าของค่าอัตราการตัดศูนย์สูงสุด (a^*Z_{ref}) โดยค่า a จะเท่ากับ 0.2 เช่นกัน

4. จากจุดเริ่มต้น (a_1) ที่หาได้จากขั้นตอนที่ 1 จะเป็นเฟรมแรกที่จะพิจารณา โดยจะทำการเปรียบเทียบอัตราการตัดศูนย์ของเฟรมดังกล่าวกับอัตราการตัดศูนย์อ้างอิง ถ้ามีค่ามากกว่า จุดเริ่มต้นจะถูกเปลี่ยนตำแหน่ง โดยจะเลื่อนไปยังจุดแรกที่มีค่าน้อยกว่าอัตราการตัดศูนย์อ้างอิง (a_2) และจะถือว่าจุดนี้เป็นจุดเริ่มต้นที่แท้จริง ซึ่งการเลื่อนเฟรมจะเลื่อนไปด้านหลัง (เฟรม $a_1-1, a_1-2, \dots, 2, \dots, 1$) แต่ในกรณีที่ a_1 น้อยกว่าอัตราการตัดศูนย์อ้างอิง จะถือว่า a_1 ที่หาได้จากขั้นตอนที่ 1 เป็นจุดเริ่มต้นที่ถูกต้องอยู่แล้ว

5. การหาจุดสุดท้าย จะหาจากจุดสุดท้าย (b_1) ที่ได้จากขั้นตอนที่ 1 พิจารณาเหมือนข้อ 3 ก็จะเป็นเฟรมแรกที่จะเปรียบเทียบอัตราการตัดศูนย์กับอัตราการตัดศูนย์อ้างอิง ถ้ามากกว่า จุดสุดท้ายจะเปลี่ยนตำแหน่ง โดยจะเลื่อนเฟรมไปทางด้านหน้า (เฟรม b_1+1, b_1+2, \dots) จนกว่าจะเจอจุดแรกที่มีค่าน้อยกว่าอัตราการตัดศูนย์อ้างอิง (b_2) และจะถือว่าจุดนี้เป็นจุดสุดท้ายที่แท้จริง แต่ถ้าอัตราการตัดศูนย์ที่ b_1 น้อยกว่าอัตราการตัดศูนย์อ้างอิง ก็กล่าวได้ว่า b_1 เป็นจุดสุดท้ายที่ถูกต้องอยู่แล้วเช่นกัน

2.3.2.5 การนอร์มอลไลซ์ (Normalization) เนื่องจากสัญญาณเสียงพูดแต่ละคำมีความยาวไม่เท่ากัน จึงต้องมีการนอร์มอลไลซ์สัญญาณเสียงพูดให้ยาวเท่ากัน

ในการเปรียบเทียบเสียงพูดแต่ละเสียง จะเป็นการเปรียบเทียบกันแบบเฟรมต่อเฟรม แต่เนื่องจากในเสียงพูดแต่ละเสียงมีการใช้เวลาไม่เท่ากัน แม้จะเป็นคำพูดคำเดียวกันและเป็นคนเดียวกันพูดก็จะได้ระยะเวลาไม่เท่ากัน ฉะนั้นจำนวนเฟรมในแต่ละรูปแบบเสียงพูดจึงไม่เท่ากัน ซึ่งเป็นปัญหาในการเปรียบเทียบรูปแบบเสียงระหว่างรูปแบบที่มาจากกลุ่มเสียงทดสอบกับรูปแบบที่มาจากกลุ่มเสียงอ้างอิง วิธีแก้ไขวิธีหนึ่ง คือ เป็นการทำให้จำนวนเฟรมของรูปแบบเสียงที่จะนำมาเปรียบเทียบมีจำนวนเท่ากัน เพื่อง่ายต่อการเปรียบเทียบแบบเฟรมต่อเฟรม

ตัวอย่างของการนอร์มอลไลซ์สัญญาณเสียงพูด การนอร์มอลไลซ์สัญญาณขนาด 5 จุดข้อมูล เป็นสัญญาณขนาด 9 จุดข้อมูล โน้ตขึ้นแรกคำนวณค่า timestep ได้เท่ากับ 0.5 สัญญาณนอร์มอลไลซ์จุดแรกมีค่าเท่ากับสัญญาณเสียงพูดเดิมจุดแรกสัญญาณนอร์มอลไลซ์จุดที่สองซึ่งปรากฏบนแกนเวลาใหม่ที่เวลาที่ 1 มีค่าเท่ากับสัญญาณเสียงพูดเดิมบนแกนเวลาเดิมที่เวลาเท่ากับ timestep เนื่องจากสัญญาณเสียงพูดเดิมเป็นสัญญาณไม่ต่อเนื่อง ทำให้ไม่ทราบค่าสัญญาณที่แท้จริงที่เวลาไม่เท่ากับจำนวนเต็ม จึงทำการประมาณค่าในช่วงเชิงเส้น (linear interpolation) จากสัญญาณเสียงพูดเดิม 2 จุด ที่อยู่ใกล้ที่สุดที่เวลาเป็นจำนวนเต็ม คือสัญญาณเสียงพูดที่เวลา 0 และที่เวลา 1

2.3.3 การหาค่าพารามิเตอร์

2.3.3.1 การวิเคราะห์การประมาณพันธะเชิงเส้น (Linear Predictive Coding Analysis)

จะทำการแปลงข้อมูลในแต่ละเฟรมของ $p+1$ จากการทำอัตโนมัติสัมพันธ์ ไปเป็นชุดประสิทธิภาพการประมาณพันธะเชิงเส้น หรือสัมประสิทธิ์พาร์คอร์ (PARCOR หรือ Reflection Coefficients) หรือสัมประสิทธิ์เชปสตรัม หรือ การแปลงข้อมูลแบบโคแบบหนึ่งของเซตเหล่านี้ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประมาณพหุเชิงเส้น เป็นขบวนการทางคณิตศาสตร์ที่ใช้ในการหาเอกลักษณ์ของเสียง โดยพิจารณาว่าเสียงเกิดจากผลรวมเชิงเส้น (Linear Combination) ของสัญญาณที่ทราบค่าแล้ว โดยใช้วิธี กำลังสองที่น้อยที่สุด (Method of Least Square) ในการเลือกค่าพารามิเตอร์ของระบบ หลักการประมาณพหุเชิงเส้นมีวิธีการใหญ่ 2 วิธี คือ วิธีการหาค่าความแปรปรวนร่วม (Covariance) และวิธีอัตรสหสัมพันธ์

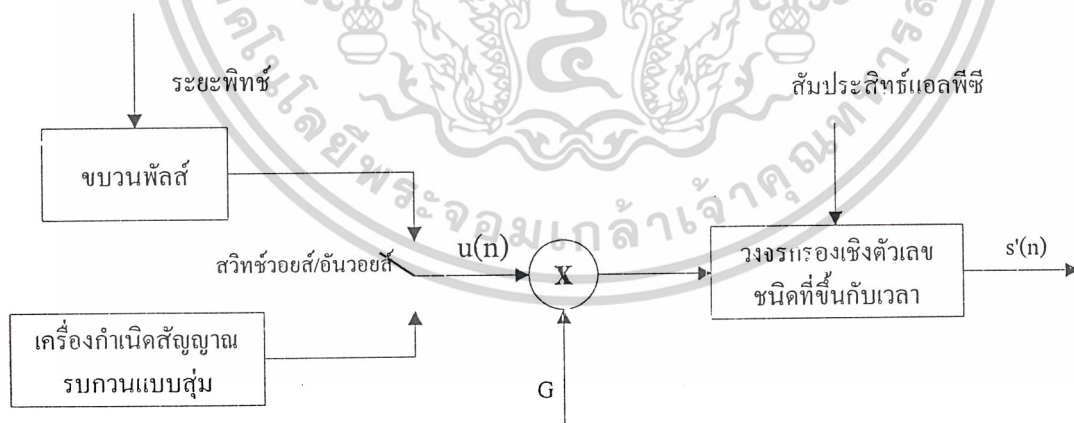
วิธีการประมาณพหุเชิงเส้นสามารถแสดงคุณสมบัติได้ใกล้เคียงกับพื้นฐานโมเดลการกำเนิดเสียงของมนุษย์ ในโครงการนี้การประมาณพหุเชิงเส้น โดยวิธีอัตรสหสัมพันธ์มาหาพารามิเตอร์ของเสียง

จากหลักการพื้นฐานของการประมาณเชิงเส้นคือ การประมาณค่าสัญญาณจากผลรวมเชิงเส้นของสัญญาณก่อนหน้านี้ สมมติว่าสัญญาณเดิม $s(n)$ การประมาณค่าสัญญาณเป็น $s'(n)$ ดังนั้นสามารถอธิบายการประมาณเชิงเส้นได้ด้วยสมการต่อไป

$$S'(n) = \sum_{k=1}^p \alpha_k s(n-k) \tag{2.11}$$

เมื่อ α_k เป็นค่าคงที่ เรียกวิธีการนี้ว่าการประมาณเชิงเส้นอันดับ p โดยมีเงื่อนไขว่า ค่า α_k ที่ใช้ในการประมาณจะต้องทำให้ ผลรวมของกำลังสองของความคลาดเคลื่อน $\{s(n)-s'(n)\}^2$ มีค่าน้อยที่สุดนั่นคือ $\sum e^2(n) = \sum \{s(n)-s'(n)\}^2$ มีค่าต่ำสุด ซึ่งจะทำให้การประมาณเชิงเส้นวิธีอัตรสหสัมพันธ์

จากหลักการพื้นฐานของการประมาณเชิงเส้นและแบบจำลองระบบสร้างสัญญาณเสียงเราสามารถเขียนบล็อกไดอะแกรมการทำการประมาณเชิงเส้น มาสร้างสัญญาณเสียงพูดได้ดังรูปที่ 2.7 (Rebiner and Levinson, 1981)



รูปที่ 2.7 บล็อกไดอะแกรมโมเดลการสร้างสัญญาณเสียงพูดอย่างง่าย

จากรูป 2.7 สามารถเขียนสมการได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$s(n) = G * u(n) + \sum_{k=1}^p a_k s(n-k) \quad (2.12)$$

การประมาณเชิงเส้นโดยใช้สัมประสิทธิ์ $\{\alpha_k\}$ คือ

$$s'(n) = \sum_{k=1}^p \alpha_k s(n-k) \quad (2.13)$$

ดังนั้น ความคลาดเคลื่อนคือ

$$e(n) = s(n) - s'(n) = s(n) - \sum_{k=1}^p \alpha_k s(n-k) \quad (2.14)$$

ฟังก์ชันถ่ายโอนระหว่าง $e(n)$ และ $s(n)$ คือ

$$A(z) = E(z)/S(z) = 1 - \sum_{k=1}^p \alpha_k z^{-k} \quad (2.15)$$

จากสมการ (2.13)–(2.15) จะเห็นได้ว่าถ้า $\{\alpha_k\} = \{a_k\}$ แล้ว

$$e(n) = G * u(n) \quad (2.16)$$

ดังนั้น ค่าผลรวมของกำลังสองของความคลาดเคลื่อน

$$E_n = \sum_0^m e^2(m)$$

$$E_n = \sum_m [s(m) - s'(m)]^2 \quad (2.17)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ n คือ ช่วงที่ n ของสัญญาณที่ใช้คำนวณ เพราะฉะนั้นเพื่อให้ได้ค่า E_n ต่ำที่สุดจะต้องมีเงื่อนไขว่า

$$\frac{\partial E_n}{\partial \alpha_i} = 0$$

จากสมการ 2.20

$$\begin{aligned} \frac{\partial E_n}{\partial \alpha_i} &= -2 S_n(m-i) \sum_m \left[s_n(m) - \sum_{k=1}^p \alpha_k S_n(m-k) \right] \quad \text{เมื่อ } i=1,2,3,\dots,p \\ &= -2 \left[\sum_m s(m)s(m-i) \sum_{k=1}^p \sum_m \alpha_k s(m-k)s(m-i) \right] \quad (2.18) \\ &\quad \frac{\partial E_n}{\partial \alpha_i} = 0 \quad \text{ก็ต่อเมื่อ} \\ &\quad \sum_{k=1}^p \alpha_k \sum_m S_n(m-k)S_n(m-i) = \sum_m S_n(m)S_n(m-i) \\ &\quad \text{เมื่อ } i=1,2,3,\dots,p \end{aligned}$$

ถ้าเรากำหนดให้ $\phi_n(i,j) = \sum_m S_n(m-k)S_n(m-i)$ เพราะฉะนั้น

$$\sum_{k=1}^p \alpha_k \phi_n(i,j) = \phi_n(i,0) \quad (2.19)$$

โดยสมการ (2.17) - (2.18) จะเห็นได้ว่า

$$E_n = \sum_m S_n^2(m) - \sum_{k=1}^p \alpha_k \sum_m S_n(m)S_n(m-k) \quad (2.20)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจาก $\phi_n(i,j) = \sum S_n(m-k)S_n(m-i)$

$$E_n = \phi_n(0,0) - \sum_{k=1}^p \alpha_k \phi_n(0,k) \quad (2.21)$$

สมมติว่าใน 1 เฟรม ของสัญญาณ ที่ตัดมาจำนวนมี N ตัวอย่าง คือ $s_n(0), s_n(1), s_n(2), \dots, s_n(N-1)$ ใน
 ที่นี้เราให้ $s_n(m) = 0$ เมื่อ $m < 0$ หรือ $m > N-1$ เพราะฉะนั้น

$$\begin{aligned} \phi_n(i,j) &= \sum_m S_n(m-k)S_n(m-i) \\ R_n(k) &= \sum_m^{N-1-k} S_n(m)S_n(m+k) \quad \text{เมื่อ } k=0,1,2,\dots,p \\ &= \sum_{m=0}^{N-1-(i-k)} S_n(m)S_n(m+i-k) \quad 0 \leq k \leq p, 1 \leq i \leq p \end{aligned} \quad (2.22)$$

ดังนั้น จากสมการ (2.21) และ (2.22) จะได้ว่า

$$R_n(k) = \sum_{m=0}^{N-1-k} x^1(m)x^1(m+k)$$

จากสมการที่ (2.22) จะได้ว่า

$$\sum_{k=1}^p \alpha_k R_n(i-k) = R_n(i) \quad \text{เมื่อ } i=1,2,3,\dots,p \quad (2.23)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการ 2.23 เขียนให้อยู่ในรูปเมตริกซ์ได้เป็น

$$\begin{pmatrix} R_n(0) & R_n(1) & \dots & R_n(p-1) \\ R_n(1) & R_n(1) & \dots & R_n(p-2) \\ \dots & \dots & \dots & \dots \\ R_n(p-1) & R_n(p-2) & \dots & R_n(0) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \dots \\ \alpha_p \end{pmatrix} = \begin{pmatrix} R_n(1) \\ R_n(2) \\ \dots \\ R_n(p) \end{pmatrix} \quad (2.24)$$

หรือ $R_n \alpha = r_n$

เมื่อ $R_n = \begin{pmatrix} R_n(0) & R_n(1) & \dots & R_n(p-1) \\ R_n(1) & R_n(0) & \dots & R_n(p-2) \\ \dots & \dots & \dots & \dots \\ R_n(p-1) & R_n(p-2) & \dots & R_n(0) \end{pmatrix}, \alpha = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \dots \\ \alpha_p \end{pmatrix}$ (2.25)

และ $r_n = \begin{pmatrix} R_n(1) \\ R_n(2) \\ \dots \\ R_n(p) \end{pmatrix}$

2.3.3.2 การหาอัตสหสัมพันธ์

วิธีอัตสหสัมพันธ์ส่วนย่อย หรือวิธีพาร์คอร์ (Partial Autocorrelation) เป็นวิธีการหาค่าพารามิเตอร์ในการประมาณค่าสัญญาณ ด้วยวิธีประมาณพหุเชิงเส้นวิธีหนึ่ง โดยใช้คุณสมบัติของผลคูณภายใน (Inner Product) และความสัมพันธ์เชิงพิกัดฉาก (Orthogonal Relationship) ของฟังก์ชันพหุนาม (Polynomial Function) ในการเลือกค่าพารามิเตอร์ที่ทำให้เกิดความคลาดเคลื่อนต่ำสุด

กำหนดให้ $x_m^+(n)$ เป็นความคลาดเคลื่อนในการประมาณค่าแบบก้าวหน้า (Forward Prediction) ของสัญญาณ $\{x(n)|n_0 \leq n \leq n_1\}$ โดยใช้สัมประสิทธิ์ในการกะประมาณค่า m ตัว $\{a_m|1 \leq I \leq m\}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$x_m^+(n) = x(n) - \left\{ - \sum_{i=0}^m a_{mi} x(n-1) \right\} \quad (2.26)$$

$$= \sum_{i=0}^m a_{mi} x(n-i) \quad (2.27)$$

ในการทำงานเดียวกันกับความคลาดเคลื่อนในการประมาณค่าแบบถดถอย (Backward Prediction) $x_m^-(n)$ ของสัญญาณ $x(n)|_{n_0 \leq n \leq n_1}$ โดยใช้สัมประสิทธิ์ในการกะประมาณค่า m ตัว $b_{mi}|1 \leq i \leq m$

$$x_m^-(n) = x(n-m-1) - \left\{ - \sum_{i=0}^m b_{mi} x(n-1) \right\} \quad (2.28)$$

$$= \sum_{i=0}^m b_{mi} x(n-i) \quad \text{เมื่อ } b_{m,m+1} = 1 \quad (2.29)$$

ค่าความคลาดเคลื่อนกำลังสองรวม (Total Square Error) แบบก้ำวหน้า α_m และแบบถดถอย β_m เมื่อ $m = 1, 2, \dots, M$ ในช่วง $[n_0, n_1]$ มีค่า

$$\alpha_m = \sum_{n=n_0}^{n_1} \left\{ x_m^+(n) \right\}^2 \quad (2.30)$$

$$\beta_m = \sum_{n=n_0}^{n_1} \left\{ x_m^-(n) \right\}^2 \quad (2.31)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนต่อไปในการประมาณฟังก์ชันเชิงเส้น คือการเลือกค่าสัมประสิทธิ์ในการประมาณค่า $\{a_m\}$ และ $\{b_m\}$ ที่ทำให้ค่า α_m และ β_m มีค่าต่ำสุด

สัมประสิทธิ์พาร์ตอร์ การเลือกค่าพารามิเตอร์ของระบบกำเนิดเสียงโดยวิธี พาร์ตอร์ คือการหาค่าของสัมประสิทธิ์ $\{k_m | 1 \leq m \leq M\}$ ที่ทำให้สามารถแสดงฟังก์ชันพหุคูณ $A_m(z)$ ในรูปผลรวมเชิงเส้นของ $A_{m-1}(z)$ และ B_{m-1} โดยเริ่มจาก

$$A_0(z) = 1 \quad (2.32)$$

และ

$$B_0(z) = z^{-1} \quad (2.33)$$

และสำหรับค่า m ใดๆ ที่มากกว่าศูนย์

$$A_m(z) = A_{m-1}(z) + k_m B_{m-1}(z) \quad (2.34)$$

ผลคูณภายใน

$$(A_m(z), z^{-m}) = (A_{m-1}(z), z^{-m}) + k_m (B_{m-1}(z), z^{-m}) \quad (2.35)$$

จากคุณสมบัติเชิงพิกัดฉากของฟังก์ชันพหุคูณ

$$(A_m(z), z^{-m}) = 0 \quad (2.36)$$

$$\begin{aligned} (A_m(z), z^{-m}) &= (A_{m-1}(z), B_{m-1}(z)) \\ &= (1, B_{m-1}(z)) \end{aligned} \quad (2.37)$$

และ

$$\begin{aligned} (B_m(z), z^{-m}) &= (B_{m-1}(z), B_{m-1}(z)) \\ &= \|1, B_{m-1}(z)\|^2 \\ &= \beta_{m-1} \end{aligned} \quad (2.38)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แทนค่าในสมการ (2.30)

$$k_m = \frac{-1}{\beta_{m-1}} (A_{m-1}(z), B_{m-1}(z))$$

$$= \frac{-1}{\beta_{m-1}} \left\{ \sum_{n=n_0}^{n_1} \sum_{i=0}^{m-1} a_{mi} x^{(n-i)} \right\}$$

$$= \left\{ \sum_{n=n_0}^{n_1} \sum_{i=0}^{m-1} b_{mi} x^{(n-j)} \right\}$$

$$\frac{-1}{\beta_{m-1}} \left\{ \sum_{n=n_0}^{n_1} \frac{x - (n)}{m} \right\} \left\{ \sum_{n=n_0}^{n_1} \frac{x - (n)}{m} \right\} \quad (2.39)$$

จาก

$$A_m(z) = \sum_{i=0}^m a_{mi} z^{-i} \sum_{i=0}^m a_{mi} z^{-i} \quad (2.40)$$

ให้ $j = (i-1)n-1$

$$z^{-(m+1)} A_m(z^{-1}) = z^{-(m+1)} \sum_{i=0}^m a_{mi} z^i$$

$$= \sum_{i=0}^m a_{mi} z^{i-m-1} \quad (2.41)$$

$$z^{-(m+1)} A_m(z^{-1}) = \sum_{j=1}^{m+1} a_{m,m=1-j} z^{-j} \quad (2.42)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ

$$b_{mj} = a_{m,m+1+j} \quad (2.43)$$

$$B_m(z) = z^{-(m+1)} B_m(z^{-1}) \quad (2.44)$$

และ

$$A_m(z) = z^{-(m+1)} B_m(z^{-1}) \quad (2.45)$$

จากสมการ (2.41)

$$z^{-(m+1)} A_m(z^{-1}) = z^{-1} \{z^{-m} A_{m-1}(z^{-1}) + k_m z^m B_{m-1}(z^{-1})\} \quad (2.46)$$

$$B_m(z) = z^{-1} \{k_m A_{m-1}(z) + B_{m-1}(z)\} \quad (2.47)$$

จากสมการ (2.41) และ (2.43)

$$x_m^+(z) = A_m(z)x(z)$$

$$x_m^-(z) = B_m(z)x(z) \quad (2.48)$$

แทนค่าทั้งสองลงในสมการ (2.27) และ (2.40)

$$x_m^+(z) = x_{m-1}^+(z) + k_m x_{m-1}^-(z)$$

$$x_m^-(z) = z^{-1} \{k_m x_{m-1}^-(z) + x_{m-1}^-(z)\} \quad (2.49)$$

หรือแสดงความสัมพันธ์ในเชิงเวลา

$$x_m^-(z) = x_{m-1}^+(n) + k_m x_{m-1}^-(n)$$

$$x_m^-(z) = k_m x_{m-1}^+(n-1) + x_{m-1}^-(n-1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากความสัมพันธ์ทั้งหมดที่กล่าวมาทำให้สามารถหาค่าพารามิเตอร์ของการวิเคราะห์สัญญาณ ซึ่งต่อไปจะเรียกพารามิเตอร์ที่ได้ด้วยวิธีนี้ว่า สัมประสิทธิ์พาร์คอร์หรือสัมประสิทธิ์การสะท้อน (Reflection Coefficient) จากสมการ (2.27) เมื่อ $A_0(z) = 1$

$$A_m(z) = 1 + \sum_{i=1}^m k_i B_{i-1}(z) \quad (2.50)$$

$$\|A_m(z) - 1\|^2 = \sum_{i=1}^m k_i^2 \beta_{i-1} \quad (2.51)$$

และ

$$\begin{aligned} \|A_m(z) - 1\|^2 &= \|A_m(z)\|^2 - 2(A_m(z), 1) + \|1\|^2 \\ &= \alpha_m - 2\alpha_m + \|1\|^2 \\ &= \|1\|^2 - \alpha_m \end{aligned} \quad (2.52)$$

แทนค่าดังกล่าวลงในสมการ (2.52)

$$\alpha_m = \|1\|^2 - \sum_{i=1}^m k_i^2 \beta_{i-1} \quad (2.53)$$

และ

$$\alpha_{m+1} = \|1\|^2 - \sum_{i=1}^m k_i^2 \beta_{i-1} \quad (2.54)$$

จากผลต่างของสมการทั้งสอง

$$\alpha_m = \alpha_{m+1} - k_{m+1}^2 \beta_m \quad (2.55)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีที่จะใช้ในโครงการนี้ คือ การแปลงจากสัมประสิทธิ์อัตรสหสัมพันธ์ ไปเป็นชุดตัวแปรของการประมาณพหุเชิงเส้น ในกรณีที่เป็นวิธีอัตรสหสัมพันธ์ นั้นจะใช้ ซึ่งจะทำให้เราได้สัมประสิทธิ์การประมาณพหุเชิงเส้น และสัมประสิทธิ์พาร์คอร์ หรือสัมประสิทธิ์การสะท้อน ซึ่งจะถูกใช้ในขั้นตอนต่อไปได้

เราจะละ 1 ใน $r_1(m)$ เป็น $r(m)$

$$E^{(0)} = r(0) \quad (2.56)$$

$$k_i = \{r(i) - \sum_{j=1}^{L-1} \alpha_j^{(i-1)} r(i-j)\} / E^{(i-1)} \quad (2.57)$$

$$G^2 = \frac{R_n(0) - \sum_{k=1}^p \alpha_k R_n(k)}{\sum_{m=0}^{N-1} u^2(m)}$$

$$\alpha_j^{(i)} = \alpha_j^{(i-1)} - k_i \alpha_{i-j}^{(i-1)} \quad (2.58)$$

$$E^{(0)} = (1 - k_i^2) E^{(i-1)} \quad (2.59)$$

เฉพาะในสมการที่สองจะไม่คำนวณค่าที่ $I = 1$ เซตของสมการที่เหลือจะสามารถหาได้จากการทำซ้ำเมื่อ $I = 1, 2, \dots, p$ จะทำให้ได้ค่าต่าง ๆ ดังนี้

$$a_m = \text{LPC coefficients} = \alpha_m^{(p)} \quad 1 \leq m \leq p$$

$$k_m = \text{PARCOR coefficients}$$

$$g_m = \text{log area ratio coefficients} = \log [(-k_m) / (1+k_m)]$$

2.3.3.3 การหาอัตราขยาย G

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อนำไปใช้ในการเปลี่ยนพารามิเตอร์การประมาณพหุระเชิงเส้น เป็นสัมประสิทธิ์เซปสตรีม

$$e(n) = G * u(n)$$

$$E_n = \sum_{m=0}^{N-1} e^2(m) = G \sum_{m=0}^{N-1} u^2(m)$$

$$\begin{aligned} E_n &= \phi_n(0,0) - \sum_{k=1}^p \alpha_k \phi_n(0,k) \\ &= R_n(0) - \sum_{k=1}^p \alpha_k R_n(k) \end{aligned} \quad (2.60)$$

และจากสมการ 2.60 เราสามารถหาค่า G โดยตรงจาก

$$G^2 = \frac{R_n(0) - \sum_{k=1}^p \alpha_k R_n(k)}{\sum_{m=0}^{N-1} u^2(m)} \quad (2.61)$$

2.3.3.4 การเปลี่ยนพารามิเตอร์การประมาณพหุระเชิงเส้นเป็นสัมประสิทธิ์เซปสตรีม

ในการรู้จักเสียงพูดนั้น สัมประสิทธิ์เซปสตรีมนี้เป็นพารามิเตอร์ที่มีลักษณะน่าเชื่อถือได้ดีกว่า สัมประสิทธิ์ LPC ทั้งยังมีความสัมพันธ์ใกล้ชิดกับการรับรู้เสียง ตามความรู้สึกรของมนุษย์โดยแท้จริงสัมประสิทธิ์เซปสตรีมสามารถหาได้โดยตรงจากสัมประสิทธิ์การประมาณพหุระเชิงเส้น ดังนี้

$$C_0 = \ln G$$

$$C_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) C_k a_{m-k}, \quad 1 \leq m \leq p$$

$$C_m = \sum_{k=1}^{m-1} \left(\frac{k}{m} \right) C_k a_{m-k}, \quad m > p \quad (2.62)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Q \approx \frac{3}{2}p$$

2.3.3.5 สัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก (Weighted Cepstrum Coefficient)

เนื่องจากความไว (sensitivity) ของสัมประสิทธิ์เซปสตรัมที่มีลำดับต่ำต่อความชันของสเปกตรัมทั้งหมด และความไวของสัมประสิทธิ์เซปสตรัมที่มีออร์เดอร์สูง ต่อสัญญาณรบกวนและการแปรผันอื่น ๆ ที่คล้ายกัน จึงทำให้เกิดเทคนิคมาตรฐานที่จะทำการถ่วงน้ำหนักสัมประสิทธิ์เซปสตรัมโดยการใช้หน้าต่างสำหรับหลายปลายของสัญญาณ (Tapered Window) เพื่อลดความไวเหล่านี้ เมื่อพิจารณาการแสดงฟูเรียร์ของค่าลอการิทึมของขนาดของสเปกตรัม (log magnitude spectrum) เทียบความถี่ของค่าลอการิทึมของขนาดของสเปกตรัม (differentiated log magnitude spectrum) จะได้ว่า

$$\log |S(e^{j(i)})| = \sum_{m=-\infty}^{\infty} c_m e^{-jm(i)} \quad (2.63)$$

$$\frac{\partial}{\partial \omega} [\log |S(e^{j(i)})|] = \sum_{m=-\infty}^{\infty} (-jm) c_m e^{-jm(i)} \quad (2.64)$$

เมื่อพิจารณาการคูณด้วย $(-jm)$ ในการแสดงอนุพันธ์ของค่าลอการิทึมของขนาดของสเปกตรัมเป็นรูปแบบหนึ่งของการถ่วงน้ำหนัก จะได้ว่า

$$\frac{\partial}{\partial \omega} [\log |S(e^{j(i)})|] = \sum_{m=-\infty}^{\infty} c_m e^{-jm(i)} \quad (2.65)$$

$$C_m = C_m (-jm) \quad (2.66)$$

เพื่อให้ได้ความรวดเร็ว (robustness) ในการแก้ปัญหา สำหรับ m มาก ๆ จึงทำการถ่วงน้ำหนักค่าน้อย ๆ เมื่อ m เข้าใกล้ Q และการตัดทอนการคำนวณไม่รู้จบจากสมการ (2.65) เราจะพิจารณาค่าการถ่วงน้ำหนักที่ใช้ทั่วไปในรูปของ

$$C_m = W_m C_m \quad 1 \leq m \leq Q \quad (2.67)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และค่าการถ่วงน้ำหนักที่เหมาะสมนี้เป็นวงจรกรองความถี่ผ่าน (bandpass filter) ในโดเมนของเซปสตรัม (Copstrum domain) เป็น

$$W_m = \left[1 + \frac{Q}{2} \sin\left(\frac{\pi m}{Q}\right) \right] \quad (2.68)$$

ฟังก์ชันการถ่วงน้ำหนักนี้จะตัดทอนการคำนวณและทำการแก้การเน้น (De-emphasizes) ของ c_m เมื่อ $m = 1$ และ $m = Q$

2.3.4 การเทียบกับรูปแบบอ้างอิง

2.3.4.1 ภาคแบบอ้างอิง

คือส่วนที่เก็บค่าพารามิเตอร์ของสัญญาณ (สัมประสิทธิ์แบบต่างๆ) ที่ใช้เป็นสัญญาณเลข 0-9 มาทำการหาค่าเฉลี่ยของแบบอ้างอิงแต่ละแบบคังสมการ

$$R^i = \frac{\sum_{j=1}^{J-1} R_j^i}{J} \quad (2.69)$$

โดยที่ j จะแทนจำนวนบุคคลที่จะนำมาสร้างแบบอ้างอิง
 i แทนหมายเลขแบบอ้างอิง
 R_j^i แทนรูปแบบของเสียงที่ i ของคนที่ j
 R^i เป็นแบบอ้างอิงที่ i

กล่าวคือ เมื่อมีสัญญาณเสียงทดสอบเข้ามาและผ่านภาคการหาขอบเขตของคำ และค่าพารามิเตอร์ของสัญญาณแล้ว ก็นำมาเปรียบเทียบกับความคล้ายคลึงโดยวิธีไดนามิกไทม์วาร์ปกับสัญญาณอ้างอิง 0-9 ในภาคไดนามิกไทม์วาร์ปไป

2.3.4.3 ฟังก์ชันแรปปิง (Wrapping Function)

เราสามารถแสดงสัญญาณเสียงออกมาให้อยู่ในรูปของเวกเตอร์ได้ คือ

$$\begin{aligned} A &= a_1, a_2, \dots, a_1, \dots, a_1 \\ B &= b_1, b_2, \dots, b_1, \dots, b_1 \end{aligned} \quad (2.70)$$

ในการที่จะแก้ปัญหาความแตกต่างของเวลา ระหว่างรูปแบบของสัญญาณเสียง 2 รูปแบบนั้น สามารถทำได้โดย แทนที่จะอธิบายสัญญาณด้วยธรรมชาติของการแกว่งบนแกนเวลาหรือความแตกต่างของเวลา เราจะอธิบายด้วยระนาบ $i-j$ ดังแสดงในรูปที่ 2.8 โดยรูปแบบสามารถแสดงได้ด้วยลำดับของจุด $c = (i,j)$ จะได้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

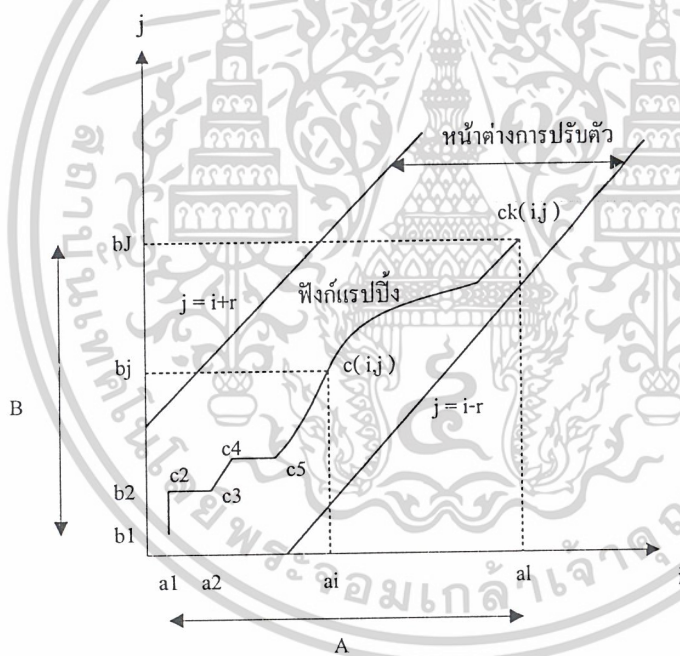
$$F = c(1), c(2), \dots, c(k), \dots, c(K) \tag{2.71}$$

โดย $c(k) = (i(k), j(k))$

จากคู่ลำดับนี้จะสามารถมองเป็นฟังก์ชัน ซึ่งจะโยงแกนเวลาของรูปแบบสัญญาณเสียง A ไปยัง B เราเรียกฟังก์ชันนี้ว่า ฟังก์ชันแรปปิง (Wrapping Function) ดังแสดงในรูปที่ 2.8 ถ้าไม่มีความแตกต่างของเวลา ระหว่างสัญญาณทั้งสองแล้ว จะได้ว่าฟังก์ชันแรปปิงจะมีลักษณะเป็นเส้นทะแยงมุม $i=j$ ฟังก์ชันนี้จะมีการเปลี่ยนแปลงไปตามความแตกต่างของเวลาที่เกิดขึ้น

ให้ความแตกต่างระหว่างเวกเตอร์ a_i และ b_j เป็น

$$d(c) = d(i,j) = ||a_i - b_j|| \tag{2.72}$$



รูปที่ 2.8 แสดงให้เห็นลักษณะของฟังก์ชันแรปปิง

ดังนั้นจะได้ว่า ผลต่างรวมของฟังก์ชันแรปปิง F คือ

$$EF = \sum_{k=1}^K d(c(k)).w(k) \tag{2.73}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

$$D(a, B) = \text{Min} \left[\frac{\sum_{k=1}^K d(c(k)) \cdot w(k)}{\sum_{k=1}^k w(k)} \right] \quad (2.74)$$

โดยที่ตัวหาร $\sum w(k)$ จะเป็นตัวชดเชยผลของค่า K คือ จำนวนจุดของฟังก์ชันแรปปิ่ง จากสมการที่ (2.77) จะได้รูปทั่วไปของระยะทางการปรับตัวตามแกนเวลา (Time-normalized Distance) เราเรียกวิธีการที่จะแก้สมการนี้ได้ว่ามีประสิทธิภาพว่า ไดนามิกโปรแกรมมิ่ง (Dynamic Programming) อย่างไรก็ตามการหาฟังก์ชันแรปปิ่งที่เหมาะสมนั้น ยังขึ้นอยู่กับข้อกำหนดของฟังก์ชันแรปปิ่ง และการระบุถึงสัมประสิทธิ์น้ำหนักอีกด้วย

ในการหาค่าฟังก์ชันแรปปิ่ง ได้มีการกำหนดเงื่อนไขไว้ดังนี้ คือ

ก. เงื่อนไขโมโนโทนิก (Monotonic Condition)

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

ข. เงื่อนไขความต่อเนื่อง (Continuity Condition)

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

จากเงื่อนไขทั้ง 2 นี้ ทำให้ได้ความสัมพันธ์ระหว่างจุด 2 จุดคือ

$$c(k-1) = \begin{cases} (i(k), j(k)-1) \\ (i(k)-1, j(k)-1) \\ \text{หรือ } (i(k)-1, j(k)) \end{cases} \quad (2.75)$$

ค. เงื่อนไขขอบเขต (Boundary Condition)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}
 i(1) &= 1, & j(1) &= 1 \\
 i(K) &= I, & j(K) &= J
 \end{aligned}
 \tag{2.76}$$

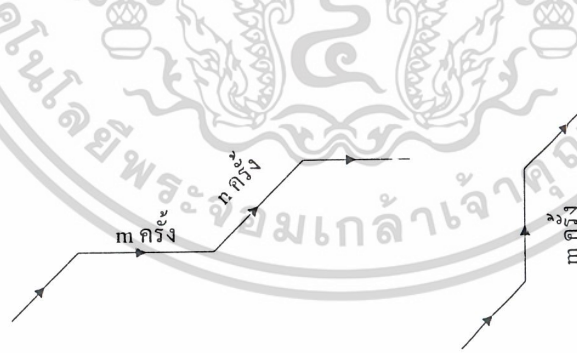
ง. เงื่อนไขการปรับขอบเขต (Boundary Adjustment)

$$|i(k) - j(k)| \leq r \tag{2.77}$$

โดยที่ r เป็นความกว้างของหน้าต่าง ซึ่งจะมีค่าเป็นเลขจำนวนเต็มบวก

จ. เงื่อนไขการเปลี่ยนความชัน (Slope Constraint Condition) เนื่องจากความแปรปรวนของความยาวในคำพูด สามารถเกิดขึ้นได้มาก เช่น เมื่อเราเปรียบเทียบคำพูด g ซึ่งสั้นมากเมื่อเทียบกับคำพูด x ในกรณีนี้จะทำให้ได้ฟังก์ชันแรบปีงที่มีค่าผิดเพี้ยนไปมา ดังนั้นจึงมีการกำหนดให้มีเงื่อนไขของการเปลี่ยนความชันขึ้น โดยเงื่อนไขนี้จะกำหนดให้จุด $c(k)$ สามารถเคลื่อนที่ไปตามแกน i หรือ j โดยไม่เปลี่ยนทิศทางได้ m ครั้ง จากนั้นจุด $c(k)$ จะไม่สามารถเคลื่อนที่ไปในทิศทางเดิมได้อย่างน้อย n ครั้ง ดังแสดงในรูปที่ 2.9ก และ 2.9ข เงื่อนไขนี้จะถูกเขียนออกมาในรูปของ

$$P = \frac{n}{m} \tag{2.78}$$



(ก) ความชันต่ำสุด (ข) ความชันสูงสุด

รูปที่ 2.9 เงื่อนไขการเปลี่ยนความชันของฟังก์ชันแรบปีง

จากสมการ (2.74) ถ้าเราให้ตัวหาร ซึ่งเรียกว่าสัมประสิทธิ์การปรับตัว(Normalized Coefficient) ไม่ขึ้นอยู่กับฟังก์ชันแรบปีง จะทำให้เขียนสมการนี้ใหม่ได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$D(A, B) = (1/N) \frac{\text{Min}}{F} \left[\sum_{k=1}^K d(c(k).w(k)) \right] \quad (2.79)$$

โดย

$$N = \sum_{k=1}^K w(k) \quad (2.80)$$

เพื่อที่จะทำให้สามารถแก้สมการที่ (2.82) ได้ง่ายขึ้น เราจึงมีการกำหนดค่าของ $w(k)$ ซึ่งเราเรียกว่า สัมประสิทธิ์น้ำหนักแบ่งออกได้เป็น 2 แบบ คือ

ก. แบบแกนเวลาสมมาตร (Symmetric) ในแบบนี้แกนเวลาทั้งสองจะถูกเปลี่ยนรูปให้อยู่บนแกนชั่วคราว ซึ่งกำหนดเป็นแกนร่วม ดังนั้น ผลรวมของสัมประสิทธิ์น้ำหนักก็คือผลรวมของแกนร่วมนั่นเอง ($i=j$)

$$w(k) = (i(k)-i(k-1)) + (j(k)-j(k-1)) \quad (2.81)$$

จะได้ว่า

$$N = I+J \quad (2.82)$$

โดย I และ J เป็นความยาวของสัญญาณ A และ B ตามลำดับ

ข. แบบแกนเวลาไม่สมมาตร (Asymmetric) ในแบบนี้แกนเวลาจะถูกเปลี่ยนรูปจากแกนเวลาหนึ่ง ไปอยู่บนอีกแกนหนึ่ง ดังนั้นผลรวมของสัมประสิทธิ์น้ำหนัก ก็คือ ผลรวมของแกน i หรือ j เท่านั้น

$$w(k) = i(k) - i(k-1) \quad (2.83)$$

จะได้ว่า

$$N = I \quad (2.84)$$

หรือ $w(k) = j(k) - J(k-1)$ จะได้ว่า $N = J$

2.3.4.4 สมการไดนามิกโปรแกรมมิ่ง

จากสมการที่ (2.77) สามารถแก้ได้โดยอาศัยทฤษฎีของไดนามิกโปรแกรมมิ่งดังนี้
เงื่อนไขเริ่มต้น

$$g_1(c(1)) = d(c(1)).w(1) \quad (2.85)$$

สมการไดนามิกโปรแกรมมิ่ง

$$g_k(c(k)) = \min_{c(k-1)} \left[g_{k-1}(c(k-1)) + d(c(k)).w(k) \right] \quad (2.86)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระยะทางการปรับตัวตามแกนเวลา

$$D(A,B) = (1/N) g_K(c(K)) \quad (2.87)$$

จากสมการทั้งหมดนี้ สมมติให้ $c(0) = (0,0)$ และคิดสัมประสิทธิ์น้ำหนักแบบสมมาตร $w(1) = 2$ แทนค่าสมการ (2.76) ลงในสมการ (2.81) จะทำให้สามารถหาค่าต่างๆ ได้ เช่น ถ้ากำหนดให้เงื่อนไขของการเปลี่ยนความชัน $P=0$ จะได้ว่า

เงื่อนไขเริ่มต้น

$$g(1,1) = 2d(1,1) \quad (2.88)$$

สมการไดนามิก โปรแกรมมิ่ง

$$g(i,j) = \min \begin{cases} g(i,j-1) + d(i,j) \\ g(i-1,j-1) + 2d(i,j) \\ g(i-1,j) + d(i,j) \end{cases} \quad (2.89)$$

ระยะทางการปรับตัวตามแกนเวลา

$$D(A,B) = (1/N) g(I,J) \quad \text{โดย } N = I+J \quad (2.90)$$

ในตารางที่ 2.1 จะสรุปให้เห็นถึงสมการไดนามิกโปรแกรมมิ่งในลักษณะต่าง ๆ ทั้งแบบแกนเวลาสมมาตร และแกนเวลาไม่สมมาตร กับการเปลี่ยนแปลงเงื่อนไขความชัน

2.3.5 กฎการตัดสินใจ (Decision Rules)

เนื่องจากผลที่ได้จากการเปรียบเทียบแบบไดนามิกไทม์วาร์ปปีง จะได้ระยะทางระหว่างแบบทดสอบและแบบอ้างอิงแต่ละแบบ ดังนั้นจำเป็นจะต้องนำไปผ่านขั้นตอนตัดสินใจ ที่จะเลือกรูปแบบที่เหมาะสมที่สุด สำหรับแบบทดสอบที่นำมาทดสอบ

เมื่อทำการเปรียบเทียบความคล้ายคลึงระหว่างสัญญาณแบบทดสอบกับสัญญาณแบบอ้างอิง 0-9 แล้ว หลังจากนั้นก็ทำการเลือกผลที่ได้จากการเปรียบเทียบในภาคที่เล็กที่น้อยที่สุด ตามกฎการตัดสินใจ นั่นคือการตัดสินใจในการจำแนกเสียงพูดว่าเสียงที่นำมาทดสอบเป็นเลขอะไร

สำหรับแบบทดสอบที่นำมาทดสอบมีหลายชนิด เช่น

2.3.5.1 กฎ Nearest Neighbor (NN)

ผลลัพธ์ของการรับรู้จากกฎการตัดสินใจนี้ ได้แก่ แบบอ้างอิงที่มีระยะทางจากแบบทดสอบน้อยที่สุด กฎนี้เหมาะสำหรับระบบการรับรู้เสียงพูดแบบบุคคลเดียว

2.3.5.2 กฎ K-Nearest Neighbor (KNN)

เป็นกฎที่ดัดแปลงมาจากกฎ NN อีกทีหนึ่ง เพื่อให้เหมาะสมกับระบบการรับรู้เสียงพูดแบบต่างบุคคล ซึ่งมีแบบอ้างอิงในแต่ละคำมากกว่า 1 แบบ กฎนี้ได้แบ่งออกเป็น 2 แบบ คือ

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

ข. แบบที่ 2 จะทำการหาแบบอ้างอิงจำนวน K แบบในแต่ละคำ โดยที่ทั้ง K แบบมีระยะทางห่างจากแบบทดสอบน้อยที่สุด ผลลัพธ์ของการเรียนรู้จะได้จาก แบบอ้างอิงที่มีค่าเฉลี่ยของระยะทางในจำนวน K แบบที่ได้รับเลือกน้อยที่สุด วิธีการนี้จะให้ผลลัพธ์ที่ถูกต้องมากขึ้นโดยกำหนดค่า K เป็น 2 หรือ 3 เมื่อเทียบกับกฎ NN (Rabiner L.R., 1979) อย่างไรก็ตามวิธีนี้จะทำให้เวลาที่ใช้ในการรับรู้เพิ่มขึ้นด้วย

ในระบบการรับรู้เสียงพูดทั่วไป จะมีการกำหนดค่าระยะทางมากที่สุดที่ยอมรับได้ (Reject Value) เพื่อเพิ่มความถูกต้องของการรับรู้ในกรณีที่ผลลัพธ์ที่ได้จากการตัดสินใจมีค่าใกล้เคียงกันมาก วิธีการกำหนดค่านี้ต้องใช้ความระมัดระวังเป็นอย่างมาก เนื่องจากถ้าเรากำหนดค่านี้สูงเกินไป หรือไม่ได้กำหนด โอกาสที่จะทำให้ผลลัพธ์ของการรับรู้ผิดพลาดไปจะมีได้มาก แต่ถ้าเรากำหนดค่านี้ต่ำเกินไป จะทำให้ผลลัพธ์ของการรับรู้สำหรับแบบอ้างอิงที่ถูกต้องเกิดขึ้นได้ยาก ดังนั้นโดยทั่วไปค่าระยะทางนี้มักจะได้มาจากการทดลอง เพื่อหาค่าที่เหมาะสมสำหรับกลุ่มของแบบอ้างอิงหนึ่ง ๆ เท่านั้น

ในโครงการนี้เราจะใช้กฎการตัดสินใจแบบ

กฎ K-Nearest Neighbor (KNN)

ใช้ในกรณีที่รูปแบบอ้างอิงแต่ละรูปแบบ (pattern) มีด้วยกันหลายชุด

กำหนดให้รูปแบบอ้างอิงแต่ละ V แบบมีอยู่ P ชุดคือ R^i โดยที่

$$i = 1, 2, 3, \dots, V \text{ และ } j = 1, 2, 3, \dots, p$$

D^j เป็น Distance ของรูปแบบที่ i จำนวน P ชุด

$$D^{i(1)} \leq D^{i(2)} \leq \dots \leq D^{i(p)} \quad (2.91)$$

โดยที่ KNN rule จะหา Average Distance จากจำนวน K ชุดที่มีค่าระยะทางรวมน้อยที่สุด

$$r^i = (1/K) \sum_{K=1}^k D^{i(k)} \quad (2.92)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยสามารถเลือกผลของการรู้จำได้จาก

$$i = \arg \min_i [r^i] \quad (2.93)$$

2.4 โปรแกรมอินเทอร์เฟซระหว่างโปรแกรมรู้จำเสียงพูดกับโปรแกรมส่งข้อมูลออกพอร์ตขนาน

แมทแลบเป็นโปรแกรมภาษาที่ใช้ในการคำนวณสมการต่างๆที่มีความซับซ้อนได้ง่ายขึ้น เนื่องจากว่าในตัวภาษาของแมทแลบนั้นมีการเขียนฟังก์ชันต่างๆให้ผู้ใช้สามารถเรียกใช้ได้เลย โดยไม่ต้องมานั่งเขียนโปรแกรมของฟังก์ชันเหล่านั้นให้ยุ่งยาก แม้ว่าแมทแลบจะสามารถช่วยอำนวยความสะดวกในด้านการคำนวณให้แก่ผู้ใช้ได้ แต่ว่าแมทแลบก็มีข้อเสียที่ว่า มันไม่สามารถจะอินเทอร์เฟซกับฮาร์ดแวร์พอร์ต (Hardware Port) ใดๆได้โดยตรง

ดังนั้นเราจึงต้องทำการเขียนโปรแกรมส่งข้อมูลออกพอร์ตขนานด้วยซีเม็กซ์-ไฟล์ (C MEX-File) ซึ่งมีความสามารถในการอินเทอร์เฟซกับฮาร์ดแวร์พอร์ต แล้วคอมไพล์โปรแกรมซีเม็กซ์ไฟล์นั้นเป็นไฟล์ประเภทไคนามิกคอลลิ่งคิไลบรารีส์ เพื่อให้แมทแลบสามารถเรียกใช้ได้ จากในโปรแกรมของแมทแลบโดยตรง

2.4.1 ซีเม็กซ์-ไฟล์ (C MEX-File)

ซีเม็กซ์-ไฟล์เป็นโปรแกรมย่อยที่เขียนขึ้นมาด้วยภาษาซี แต่มันจะมีลักษณะเป็น โปรแกรมย่อยชนิดไคนามิกคอลลิ่งคิไลบรารีส์ และมีนามสกุลของไฟล์เป็น .dll ดังนั้นแมทแลบจะสามารถเรียกและเอ็กซีคิวท์ (Execute) ซีเม็กซ์-ไฟล์นี้ได้เหมือนกับมันเป็นเอ็ม-ไฟล์ (โปรแกรมที่เขียนด้วยแมทแลบ) ซึ่งมีนามสกุลของไฟล์เป็น .m และฟังก์ชันในแมทแลบเลย

ซีเม็กซ์-ไฟล์จะถูกสร้างขึ้นมาจากการใช้สคริปท์ (Script) คำว่า mex ไปคอมไพล์ซอร์สโค้ดภาษาซีที่รวมอยู่กับชุดคำสั่งเอพีไอ (API: Application Program Interface) อยู่แล้ว

แมทแลบเอพีไอ (MATLAB API) มีหน้าที่ในการช่วยให้แมทแลบสามารถอินเทอร์เฟซกับสิ่งแวดล้อมภายนอกแมทแลบได้ อุปกรณ์แมทแลบเอพีไอนั้นจะมีมาให้ในแมทแลบอยู่แล้ว เราก็เพียงแค่หาคอมไพเลอร์ของภาษาซีมาติดตั้งเพื่อใช้ในการคอมไพล์ซีเม็กซ์-ไฟล์นั้นๆ

2.4.2 ส่วนประกอบของโปรแกรมซีเม็กซ์-ไฟล์

ซอร์สโค้ดของซีเม็กซ์-ไฟล์ประกอบด้วย 2 ส่วนแยกจากกัน

2.4.2.1 โปรแกรมย่อยสำหรับการประมวลผล (Computational Routine)

ในส่วนนี้จะบรรจุไปด้วยโค้ดสำหรับการประมวลผลทางคอมพิวเตอร์ ที่เราต้องการให้เกิดในซีเม็กซ์-ไฟล์ของเรา ในส่วนนี้อาจจะเป็นการคำนวณทางตัวเลข หรือ การนำข้อมูลเข้าและออกก็ได้

2.4.2.2 โปรแกรมย่อยเกตเวย์ (Gateway Routine)

เป็นส่วนที่จะทำการอินเทอร์เฟสโปรแกรมย่อยสำหรับการประมวลผล กับเมทแลบ โดยการใช้นิพจน์ `mexFunction` และพารามิเตอร์ของมันอันได้แก่ `prhs`, `nrhs`, `plhs` และ `nlhs`

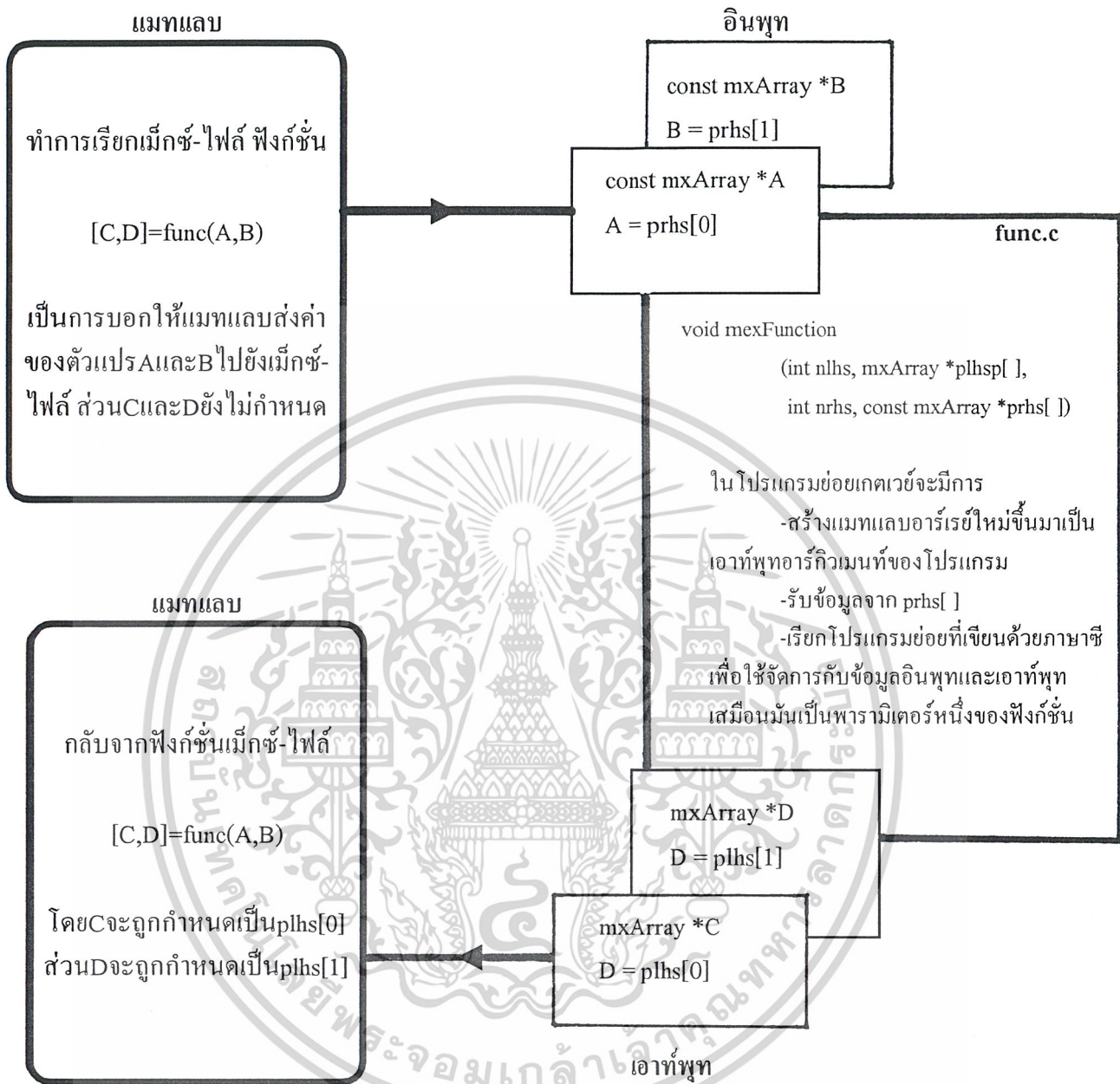
`prhs` และ `plhs` คือ อาร์เรย์ (Array) ของอินพุทอาร์กิวเมนต์ (Input Argument) ซึ่งจะอยู่ทางด้านขวาของฟังก์ชัน และเอาต์พุทอาร์กิวเมนต์ (Output Argument) ซึ่งจะอยู่ทางด้านซ้ายของฟังก์ชันตามลำดับ

`nrhs` และ `nlhs` คือ จำนวนของอินพุทอาร์กิวเมนต์ทางด้านขวา และ เอาต์พุทอาร์กิวเมนต์ทางด้านซ้ายตามลำดับ

โปรแกรมย่อยเกตเวย์จะถือว่าโปรแกรมย่อยสำหรับการประมวลผลเป็นโปรแกรมย่อยของมันอีกทีหนึ่ง บล็อกโค๊ดแอสเซมบลีของการทำงานในซีเม็กซ์-ไฟล์แสดงในรูปที่ 2.10



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 บล็อกไดอะแกรมทำงานของซีเม็กซิค-ไฟล์

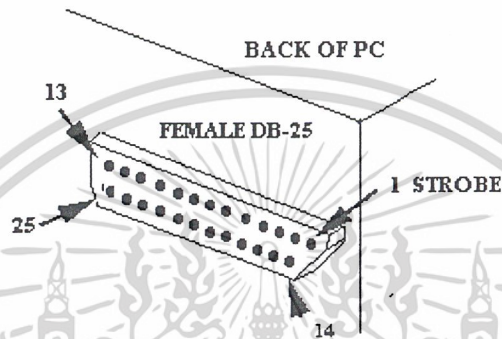
จากบล็อกไดอะแกรมจะเห็นว่าซีเม็กซิค-ไฟล์นั้นจะทำงานได้โดยเริ่มจากการเมทแล็บทำการเรียกฟังก์ชันที่เขียนด้วยซีเม็กซิค-ไฟล์ (ฟังก์ชันซีเม็กซิค-ไฟล์จะประกอบไปด้วยโปรแกรมย่อยสำหรับการประมวลผลและโปรแกรมย่อยเกตเวย์) จากนั้นจะส่งค่าอินพุทที่กำหนดจากโปรแกรมเมทแล็บไปยังโปรแกรมย่อยสำหรับประมวลผลผ่านทางโปรแกรมย่อยเกตเวย์ เมื่อโปรแกรมย่อยสำหรับประมวลผลทำการคำนวณและประมวลผลเสร็จแล้ว ก็ จะส่งค่าเอาต์พุทที่ได้ผ่านทางโปรแกรมย่อยเกตเวย์ออกมายังโปรแกรมเมทแล็บอีกทีหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 วงจรแสดงผลออกทางพอร์ตขนาน

เมื่อเราได้ส่งผลที่ได้จากการรู้จำเสียงออกทางพอร์ตขนานแล้ว เราก็จะนำผลที่ได้นั้นมาควบคุมเครื่องใช้ไฟฟ้าอีกที ซึ่งในปริยญาณิพนธ์นี้เราจะใช้การติด-ดับของแอลอีดี (LED) แสดงผลแทนการเปิด-ปิดเครื่องใช้ไฟฟ้าภายในบ้าน

พอร์ตขนานของคอมพิวเตอร์พีซี (PC Computer) ทั่วไปจะเป็นพอร์ตตัวเมีย 25 พิน (Pin) ดังรูปที่ 2.11



รูปที่ 2.11 พอร์ตขนานของคอมพิวเตอร์

ซึ่งทั้ง 25 พินจะมีการใช้งานดังตารางข้างล่างนี้

ตารางที่ 2.1 แสดงการทำงานและชนิดของพินของพอร์ตขนาน

หมายเลขของพิน	หน้าที่	ชนิด
1	สโตรบ (STROB)	ควบคุม / CONTROL
2	ข้อมูลบิตที่ 0	เอาต์พุต / OUTPUT
3	ข้อมูลบิตที่ 1	เอาต์พุต / OUTPUT
4	ข้อมูลบิตที่ 2	เอาต์พุต / OUTPUT
5	ข้อมูลบิตที่ 3	เอาต์พุต / OUTPUT
6	ข้อมูลบิตที่ 4	เอาต์พุต / OUTPUT
7	ข้อมูลบิตที่ 5	เอาต์พุต / OUTPUT
8	ข้อมูลบิตที่ 6	เอาต์พุต / OUTPUT
9	ข้อมูลบิตที่ 7	เอาต์พุต / OUTPUT
10	แอกโนเลจ (ACKNOWLEDGE)	สถานะ / STATUS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11	พอร์ตไม่ว่าง	สถานะ / STATUS
12	ไม่มีกระดาษ	สถานะ / STATUS
13	ปริ้นท์เตอร์พร้อมใช้งาน	สถานะ / STATUS
14	ขึ้นบรรทัดใหม่อัตโนมัติ	ควบคุม / CONTROL
15	ปริ้นท์เตอร์เออร์เรอร์ (ERROR)	สถานะ / STATUS
16	เตรียมปริ้นเตอร์ให้พร้อม	ควบคุม / CONTROL
17	เลือก / ไม่เลือก ปริ้นท์เตอร์	ควบคุม / CONTROL
18-25	ไม่ใช้ / กราวนด์ (GROUND)	

จากตารางจะเห็นว่าทั้ง 25 พินนั้น แบ่งออกเป็น 4 ชนิด คือ : พินข้อมูล 8 พิน , พินควบคุม 4 พิน , พินสถานะ 5 พิน และ พินกราวนด์ 8 พิน ในปฏิญญาพันธันนี้เราจะใช้เพียงพินข้อมูล 8 พิน ในการควบคุมเครื่องใช้ไฟฟ้าในบ้าน ซึ่งสามารถนำไปใช้ควบคุมเครื่องใช้ไฟฟ้าได้ถึง 8 เครื่องทีเดียว

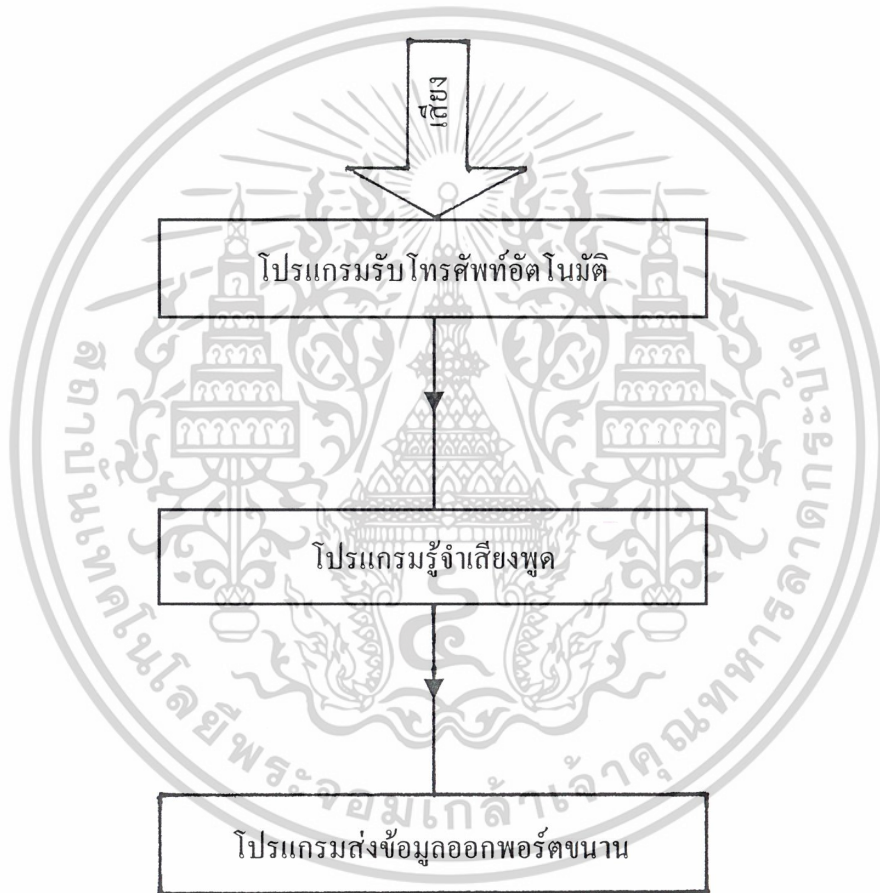


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การคำนวณและการสร้าง

จากทฤษฎีและหลักการทำให้สามารถแบ่งการทำงานของ การควบคุมเครื่องใช้ไฟฟ้าในบ้านด้วยเสียงพูดได้เป็น 2 ส่วน คือ ส่วนของ โปรแกรม (Software) และ ส่วนของ วงจรควบคุม (Hardware)

ปริญญานิพนธ์นี้ได้เน้นไปยังส่วนของการเขียนโปรแกรม ซึ่ง โฟลว์ชาร์ทของ โปรแกรมที่ใช้ในการควบคุมจะมีลักษณะดังนี้



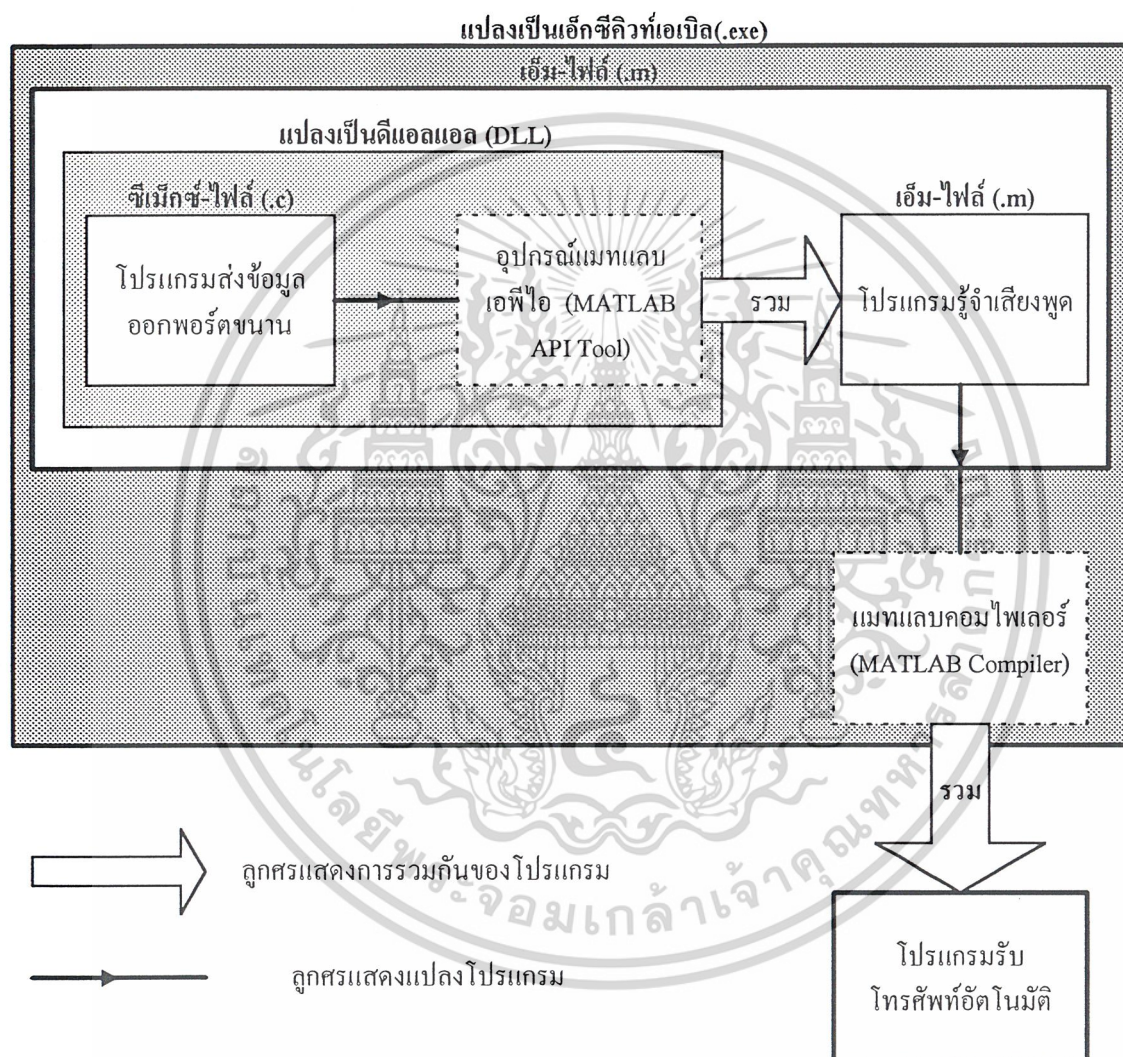
รูปที่ 3.1 โฟลว์ชาร์ทของ โปรแกรมการทำงานของ การควบคุมเครื่องใช้ไฟฟ้าภายในบ้านด้วยเสียงพูดผ่านโทรศัพท์

เมื่อเสียงที่เราโทรศัพท์เข้าบ้านมาสั่งงานเครื่องใช้ไฟฟ้าภายในบ้านผ่านโมเด็มเข้ามาแล้ว มันจะเข้าสู่โปรแกรมรับโทรศัพท์อัตโนมัติ ซึ่งจะอัดเสียงที่เราพูดสั่งงานไว้ จากนั้นจะมีการเรียกโปรแกรมรู้จำเสียงพูดให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขึ้นมาไฟล์เสียงที่อัดไว้ไปทำกระบวนการรู้จำเสียง ผลที่ได้จากการรู้จำเสียงจะถูกส่งออกพอร์ตขนานด้วยโปรแกรมส่งข้อมูลออกพอร์ตขนาน

เนื่องจากว่าโปรแกรมรู้จำเสียงพูดที่เราใช้นั้นเขียนด้วยแมทแลบ โดยพื้นฐานของภาษาแมทแลบเองแล้วมันจะไม่สามารถอินเทอร์เฟสกับสิ่งแวดล้อมภายนอกโปรแกรมของมันได้โดยตรง ดังนั้นเราจึงต้องทำการอินเทอร์เฟสโปรแกรมส่วนต่างๆให้มีการเข้าถึง และทำงานสอดคล้องกันได้ ด้วยวิธีการตามบล็อกไดอะ



แกรมในรูปแบบที่ 3.2 นี้

รูปที่ 3.2 บล็อกไดอะแกรมแสดงการสร้างโปรแกรมการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านโทรศัพท์

จากรูปที่ 3.2 โปรแกรมการควบคุมเครื่องใช้ไฟฟ้าภายในบ้านด้วยเสียงพูดผ่านโทรศัพท์ทั้งหมด จะเริ่มจากโปรแกรมส่งข้อมูลออกพอร์ตขนานก่อน โดยโปรแกรมส่งข้อมูลออกพอร์ตขนานนี้จะเขียนด้วยภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซี มีรูปแบบเป็นซีเม็กซ์-ไฟล์ มีนามสกุลของไฟล์เป็น .c จากนั้นเราจะนำซีเม็กซ์ไฟล์ที่เขียนได้ไปผ่านกระบวนการคอมไพล์โดยอุปกรณ์แมทเลบเอพีไอ ผลลัพธ์ที่ได้ออกมาจะเป็นไฟล์ชนิดไคนามิกคอลลิงค์ไลบรารีส์ที่มีนามสกุลของไฟล์เป็น .dll

โปรแกรมแมทเลบจะสามารถเรียกไฟล์ชนิดไคนามิกคอลลิงค์ไลบรารีส์นี้มาใช้งานได้ทันทีเหมือนกับว่ามันเป็นฟังก์ชันของแมทเลบเลย ดังนั้นในตัวโปรแกรมรู้จำเสียงพูดที่เขียนด้วยแมทเลบ เมื่อได้ผลลัพธ์ของการรู้จำออกมาเป็นเช่นไร ก็สามารถเรียกไฟล์ชนิดไคนามิกคอลลิงค์ไลบรารีส์นี้ให้สามารถทำงานส่งข้อมูลผลลัพธ์นั้นออกพอร์ตขนานได้ทันที ในส่วนนี้ทั้งหมดของโปรแกรมจะเขียนด้วยแมทเลบ และได้เป็นไฟล์ที่เรียกว่าเอ็ม-ไฟล์ มีนามสกุลเป็น .m

จากนั้นนำเอ็ม-ไฟล์ที่ได้ มาผ่านแมทเลบคอมไพเลอร์ เพื่อทำการคอมไพล์เอ็ม-ไฟล์ให้เป็นไฟล์ชนิดเอ็กซ์คิวทีเอเบิลอีที เพื่อให้โปรแกรมที่เขียนด้วยภาษาเซลไฟสามารถเรียกไปใช้งานได้

ส่วนสุดท้ายที่จะทำการสร้างก็คือส่วนของโปรแกรมรับโทรศัพท์อัตโนมัติ เป็นคอมโพเนนท์ที่เขียนขึ้นมาด้วยภาษาเซลไฟ เมื่อทำการรับโทรศัพท์เสร็จแล้วโปรแกรมเซลไฟจะทำการเรียกโปรแกรมรู้จำเสียงพูดที่ถูกแปลงเป็นไฟล์ชนิดเอ็กซ์คิวทีเอเบิลแล้วขึ้นมาทำการรู้จำเสียงพูดนั้นๆแล้วส่งผลที่ได้ออกทางพอร์ตขนานต่อไป

เราจะแบ่งการสร้างโปรแกรมทั้งหมดนี้ออกเป็นส่วนๆดังนี้

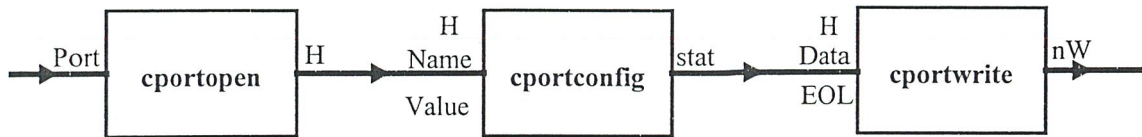
- โปรแกรมส่งข้อมูลออกพอร์ตขนาน (ซีเม็กซ์-ไฟล์)
- โปรแกรมการแปลงซีเม็กซ์-ไฟล์เป็นไฟล์ชนิดไคนามิกคอลลิงค์ไลบรารีส์
- โปรแกรมรู้จำเสียงพูด (เอ็ม-ไฟล์)
- โปรแกรมการแปลงเอ็ม-ไฟล์เป็นไฟล์ชนิดเอ็กซ์คิวทีเอเบิล
- โปรแกรมรับโทรศัพท์อัตโนมัติ

3.1 โปรแกรมส่งข้อมูลออกพอร์ตขนาน (ซีเม็กซ์-ไฟล์)

โปรแกรมส่งข้อมูลออกพอร์ตขนานเขียนด้วยโปรแกรมภาษาบอร์แลนด์ ซี (Borland C) ลงในเท็กซ์อิดิต (Text Edit) ในรูปแบบซีเม็กซ์-ไฟล์ ในการจะส่งข้อมูลออกพอร์ตขนานนี้ต้องใช้ฟังก์ชันถึง 3 ฟังก์ชันในการทำงานได้แก่

- cportopen
- cportconfig
- cportwrite

ซึ่งมีบล็อกไดอะแกรมการทำงานของฟังก์ชันทั้ง 3 ดังนี้



รูปที่ 3.3 บล็อกไดอะแกรมการทำงานของฟังก์ชันทั้งหมดในการส่งข้อมูลออกพอร์ตขนาน

3.1.1 ฟังก์ชัน cportopen

ฟังก์ชัน cportopen ทำหน้าที่ในการเปิดพอร์ตที่เราต้องการ รูปแบบของฟังก์ชันคือ

$$H = \text{cportopen}(\text{Port})$$

โดย Port เป็นอินพุตค่าสตริง (String) บอกถึงพอร์ตที่เราต้องการจะใช้ สามารถกำหนดให้มีค่าได้เป็น 'com1' ถึง 'com8' และ 'lpt1' ถึง 'lpt3' ในการเปิดพอร์ตขนานเราจะใช้ค่า 'lpt1'

H เป็นค่าเอาต์พุตแฮนเดิล (Handle) ที่ได้กลับมาจากการเปิดพอร์ตแต่ละพอร์ต ค่านี้จะคงที่ตลอดการใช้งานของโปรแกรมส่งข้อมูลออกพอร์ตขนานนี้

3.1.2 ฟังก์ชัน cportconfig

ฟังก์ชัน cportconfig ทำหน้าที่กำหนดคุณสมบัติและลักษณะต่างๆของพอร์ตที่ได้เปิดไว้ในฟังก์ชัน cportopen มาก่อนแล้ว มีรูปแบบของฟังก์ชันคือ

$$\text{stat} = \text{cportconfig}(H, \text{Name}, \text{Value}, \text{Name}, \text{Value}, \dots)$$

H เป็นค่าแฮนเดิลที่ได้จากการเปิดพอร์ตโดยฟังก์ชัน cportopen

Name เป็นอินพุตสตริงของตัวอักษรที่บอกชื่อของพารามิเตอร์ต่างๆที่ต้องการจะกำหนดคุณสมบัติ

Value เป็นอินพุตสตริงของตัวเลขหรือตัวอักษรที่บอกถึงค่าที่เราจะกำหนดให้แก่พารามิเตอร์แต่ละตัวของ Name

stat เป็นเอาต์พุตที่ได้มาจากการทำงานของฟังก์ชัน cportconfig เรียบร้อยแล้ว มันจะแสดงถึงชื่อของพารามิเตอร์และค่าของพารามิเตอร์ทั้งหมดในการติดต่อกับพอร์ต ไม่ว่าค่าที่เราจะกำหนดไว้หรือไม่ก็ตาม

รายละเอียดของพารามิเตอร์ที่ต่างๆ และการกำหนดค่าของมัน สามารถอ่านได้จากภาคผนวกของปฏิญานินพนธ์ฉบับนี้

3.1.3 cportwrite

ฟังก์ชัน cportwrite ทำหน้าที่ในการส่งข้อมูลออกไปยังพอร์ตที่เราได้เปิดไว้แต่แรก มีรูปแบบของฟังก์ชันดังนี้

$$nW = \text{cportwrite}(H, \text{Data}, \text{EOL})$$

H เป็นค่าแฮนเดิลที่ได้จากการเปิดพอร์ตโดยฟังก์ชัน cportopen

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data เป็นอาร์เรย์อินพุทของตัวอักษรที่เราต้องการ จะส่งออกพอร์ต มันจะถูกแปลงเป็นรหัสแอสกี (ASCII) ก่อนที่จะถูกส่งออกพอร์ต

EOL เป็นอินพุทที่ทำหน้าที่กำหนดว่าจะให้สัญลักษณ์ที่แสดงการถึงจุดสิ้นสุดของบรรทัดว่าเป็น 'CR' หรือ 'LF' หรือ 'CRLF'

nW คือเอาท์พุทที่ได้จากการทำงานของฟังก์ชัน `fprintf` มันจะแสดงถึงจำนวนของตัวอักษรที่ถูกส่งออกไปยังพอร์ตที่เราต้องการ

3.2 โปรแกรมการแปลงซีเม็กซ์-ไฟล์ให้เป็นไฟล์ชนิดไดนามิกคอลลิคัลไลบรารีส์

ในหัวข้อ 3.1 เราสามารถสร้างโปรแกรมส่งข้อมูลออกพอร์ตขนานในรูปแบบซีเม็กซ์-ไฟล์ได้แล้ว ในหัวข้อนี้จะกล่าวถึงการนำซีเม็กซ์-ไฟล์เหล่านั้นมาเรียกใช้โดยใช้เมทเลบเรียกได้โดยตรง

ขั้นตอนในการแปลงซีเม็กซ์-ไฟล์ให้เป็นไฟล์ชนิดไดนามิกคอลลิคัลไลบรารีส์แบ่งเป็น 2 ขั้นตอนดังนี้

3.2.1 การกำหนดและติดตั้งคอมไพเลอร์

ก่อนที่จะทำการคอมไพล์ซีเม็กซ์-ไฟล์ให้เป็นไฟล์ชนิดไดนามิกคอลลิคัลไลบรารีส์ จะต้องทำการกำหนดและติดตั้งคอมไพเลอร์ของภาษาซีให้กับเมทเลบเสียก่อน

เริ่มจากการเปลี่ยนไดเรกทอรี (Directory) ที่เมทเลบทำงานอยู่ให้เป็นไดเรกทอรีที่ซีเม็กซ์-ไฟล์ที่เราต้องการใช้งานนั้นเก็บอยู่ ด้วยคำสั่ง

```
>> cd ([matlabroot 'pathname'])
```

จากนั้นก็เริ่มทำการติดตั้งคอมไพเลอร์ด้วยคำสั่ง

```
>> mex -setup
```

เมื่อกดเอนเทอร์ (Enter) แล้วที่คอมมานด์วินโดว์ (Command Window) ของเมทเลบ จะมีหน้าต่างภาษาเพิร์ล (Perl) ขึ้นมาให้จัดการกำหนดคอมไพเลอร์ต่างๆ ได้ตามที่เราต้องการ โดยคอมไพเลอร์ที่เรากำหนดไว้นั้นต้องเป็นคอมไพเลอร์ที่อยู่ภายในชนิดที่เมทเลบได้กำหนดไว้ให้ และต้องติดตั้งโปรแกรมคอมไพเลอร์นั้นไว้ในไดเรกทอรีเดียวกับไดเรกทอรีที่เมทเลบอยู่ด้วย ในปัจจุบันนี้ใช้โปรแกรมบอร์แลนด์ ซี 5.0 เป็นคอมไพเลอร์

ทำการกำหนดเสร็จแล้วเราก็สามารถใช้ที่เรากำหนดไว้นั้นมาคอมไพล์ซีเม็กซ์-ไฟล์ให้เป็นไฟล์ชนิดไดนามิกคอลลิคัลไลบรารีส์ได้

3.2.2 การสร้างไฟล์ชนิดไดนามิกคอลลิคัลไลบรารีส์

ในการคอมไพล์เพื่อสร้างไฟล์ชนิดไดนามิกคอลลิคัลไลบรารีส์ ทำได้ด้วยคำสั่ง

```
>> mex filename.c
```

เมื่อใช้คำสั่งนี้เสร็จแล้วคอมไพเลอร์ภาษาซีและอุปกรณ์เมทเลบเอพีไอจะช่วยกันคอมไพล์และสร้างไฟล์ชนิดไดนามิกคอลลิคัลไลบรารีส์ `filename.dll` จากซีเม็กซ์-ไฟล์ออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

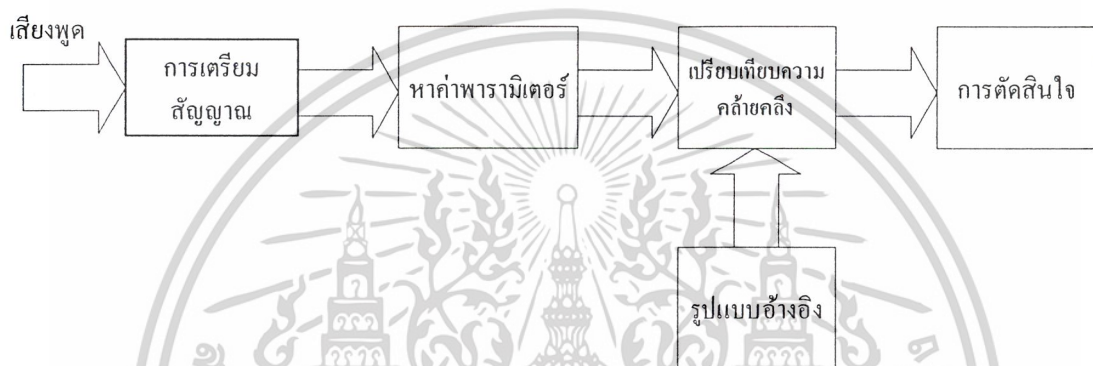
เมื่อทำสำเร็จถึงขั้นตอนนี้แล้วเราก็เรียกใช้ซีเม็กซ์-ไฟล์นี้ได้ง่ายๆจากเมทแลบคอมมานด์วินโดว์ได้เลย
ดังนี้

>> filename

จะเห็นว่ามันมีการเรียกใช้เหมือนมันเป็นฟังก์ชันตัวหนึ่งที่เขียนด้วยเอ็ม-ไฟล์เลย

3.3 โปรแกรมรู้จำเสียงพูด

การทำงานของระบบการรับรู้เสียงพูดเป็นไปตามบล็อกไดอะแกรมดังนี้



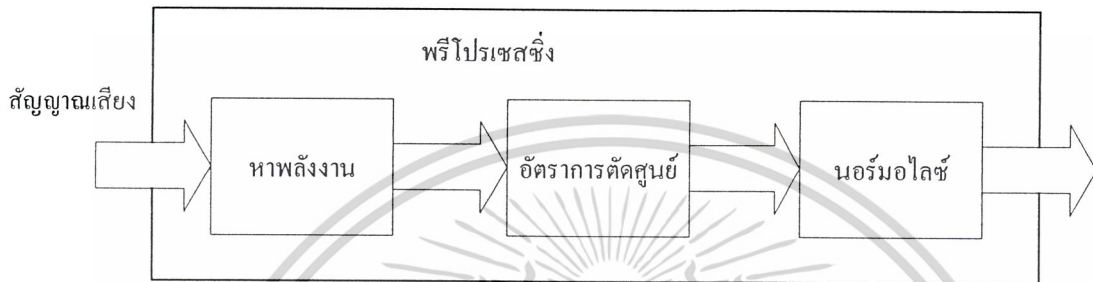
รูปที่ 3.4 แสดงบล็อกไดอะแกรมของระบบการรับรู้เสียงพูด

ซึ่งเราจะแบ่งภาคการทำงานเป็น 5 ภาคใหญ่ ๆ คือ 프리โพรเซสซิ่ง (Preprocessing), การหาค่าพารามิเตอร์, การเปรียบเทียบความคล้ายคลึง, รูปแบบอ้างอิงและการตัดสินใจ แต่ละภาคจะมีการทำงานดังขั้นตอนต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

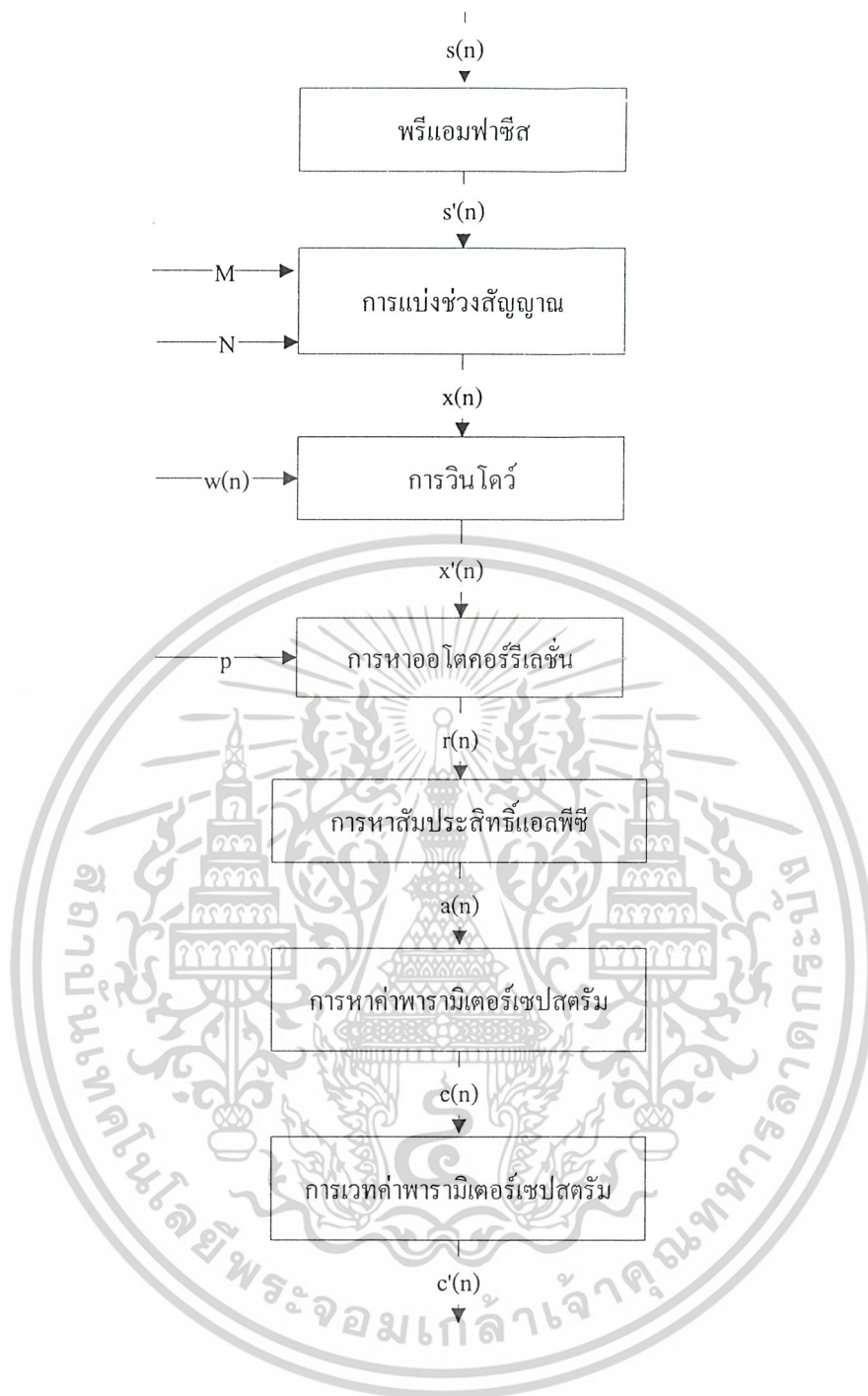
3.3.1 พรีโพรเซสซิ่ง (Preprocessing)

Preprocessing มีวิธีการทำงานตามที่ได้กล่าวไว้แล้วในบทที่ 2 คือเมื่อนำสัญญาณผ่านกระบวนการ PCM แล้ว ก็จะทำกรพลังงานและอัตราการตัดศูนย์เพื่อหาขอบเขตของสัญญาณ จากนั้นก็จะนำขอบเขตของแต่ละค่าไปทำการนอร์มอลไลซ์



รูปที่ 3.5 พรีโพรเซสซิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แสดงขั้นตอนการเตรียมสัญญาณในการวิเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

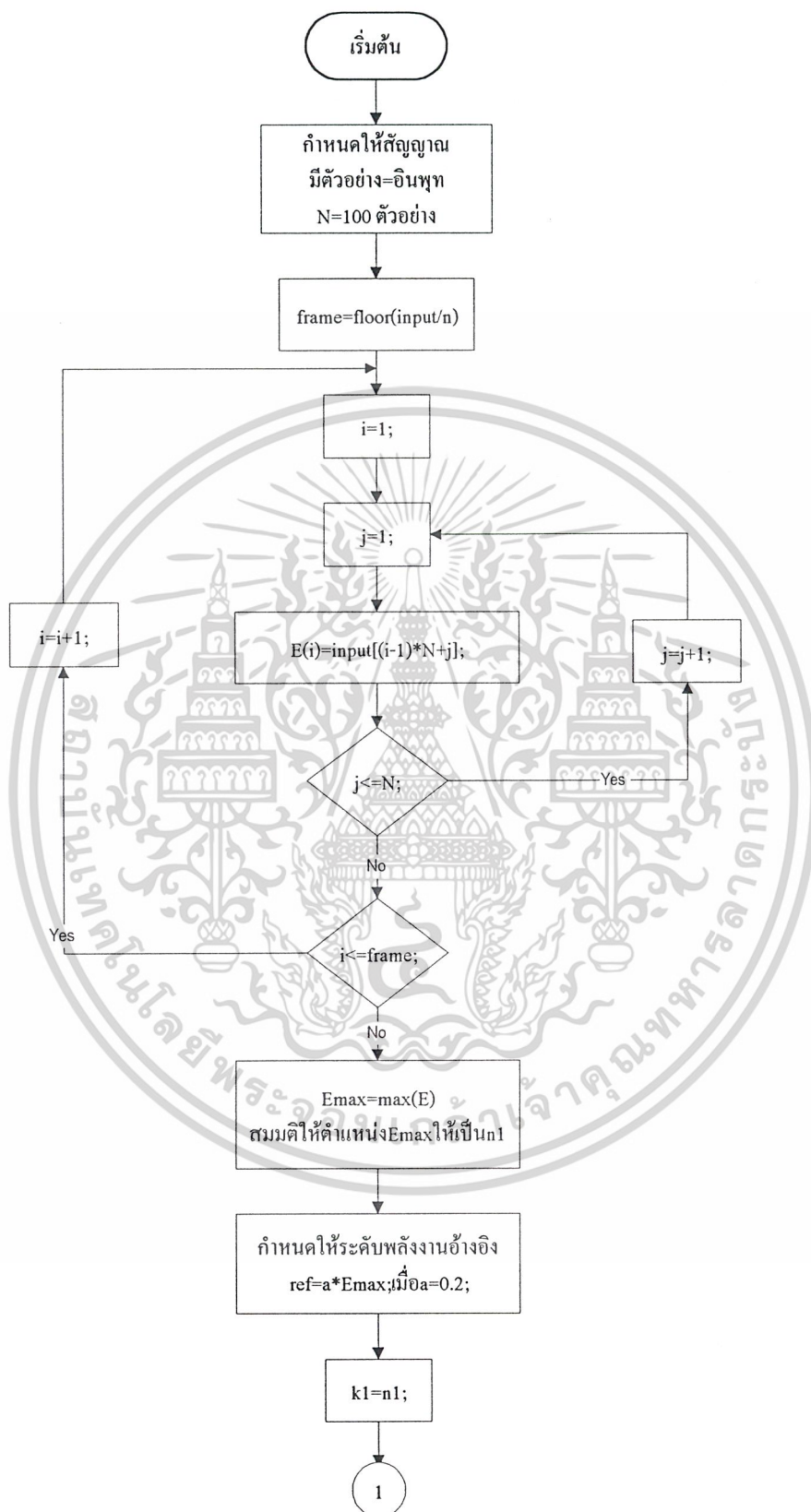
การหาจุดเริ่มต้นและจุดสุดท้ายของเสียงด้วยวิธีเปรียบเทียบค่าพลังงานของเสียงมีดังรูปที่ 3.7



รูปที่ 3.7 แสดงขั้นตอนการเตรียมสัญญาณในการวิเคราะห์

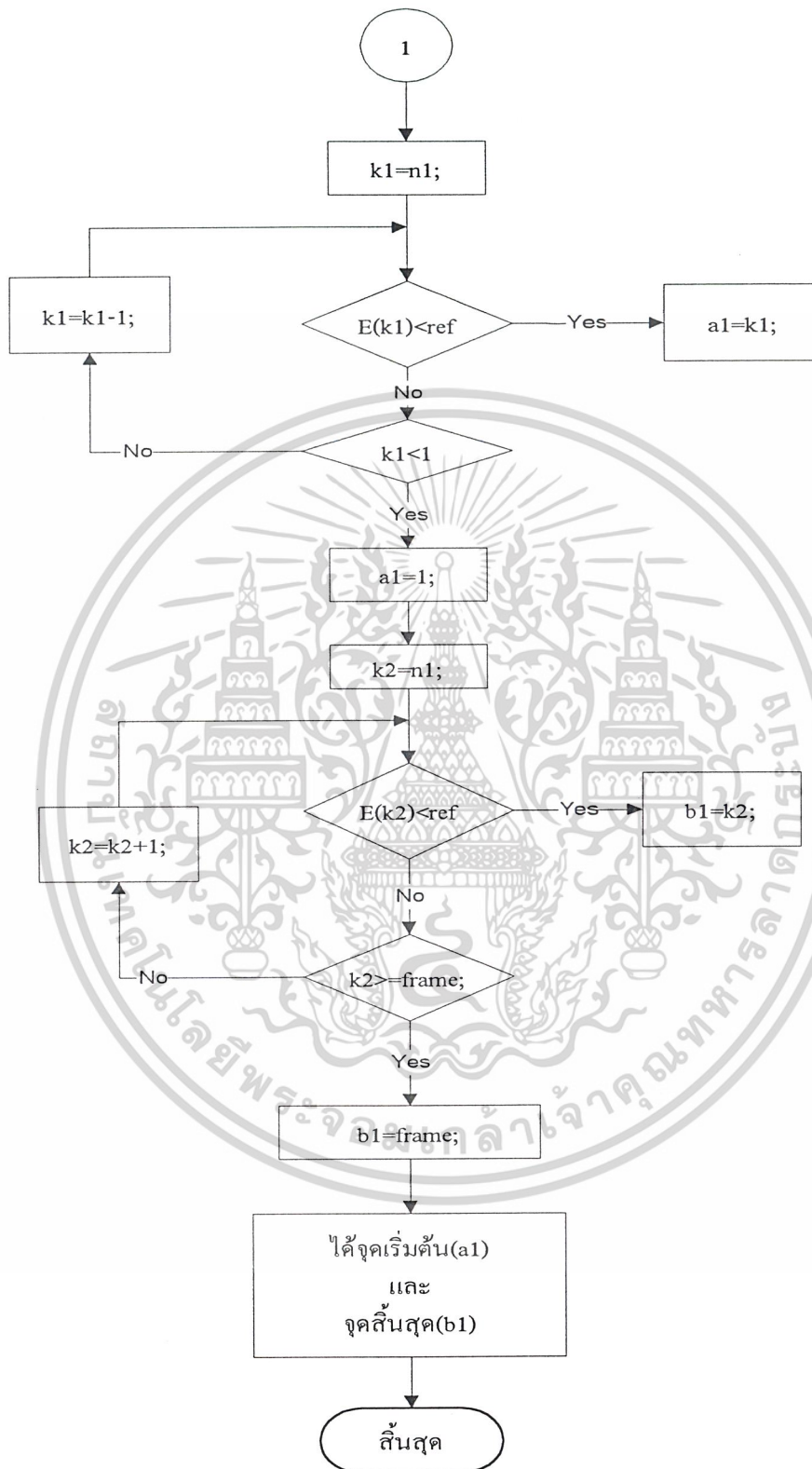
ซึ่งมีการทำเป็นโพลวชาร์ตดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงโพลซาร์ทการหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณ โดยการเปรียบเทียบค่าพลังงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงด้วยวิธีเปรียบเทียบอัตราการตัดศูนย์มีขั้นตอนดังรูปที่

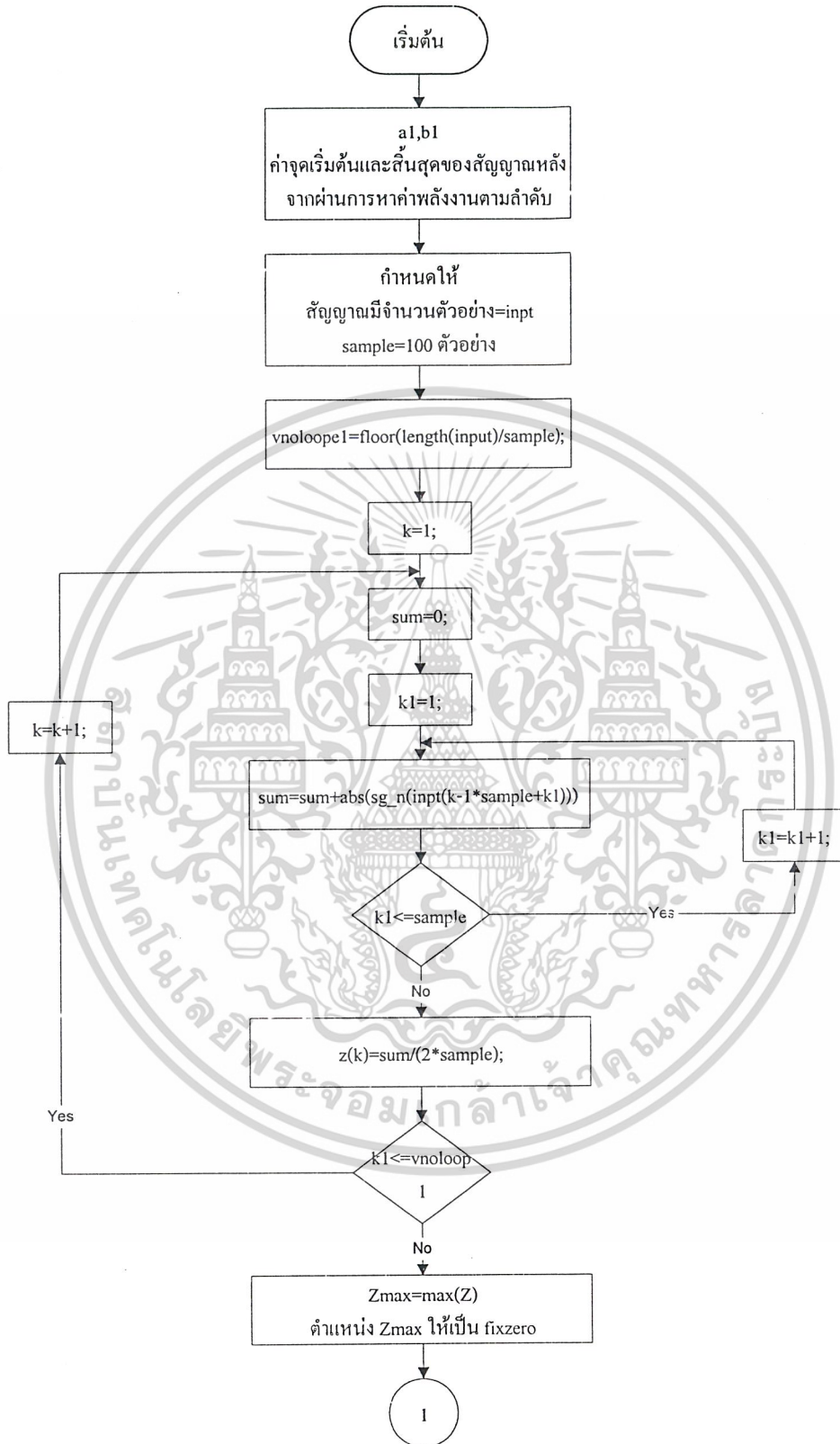
3.9



รูปที่ 3.9 แสดงขั้นตอนการหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงโดยวิธีการเปรียบเทียบอัตราการตัดศูนย์

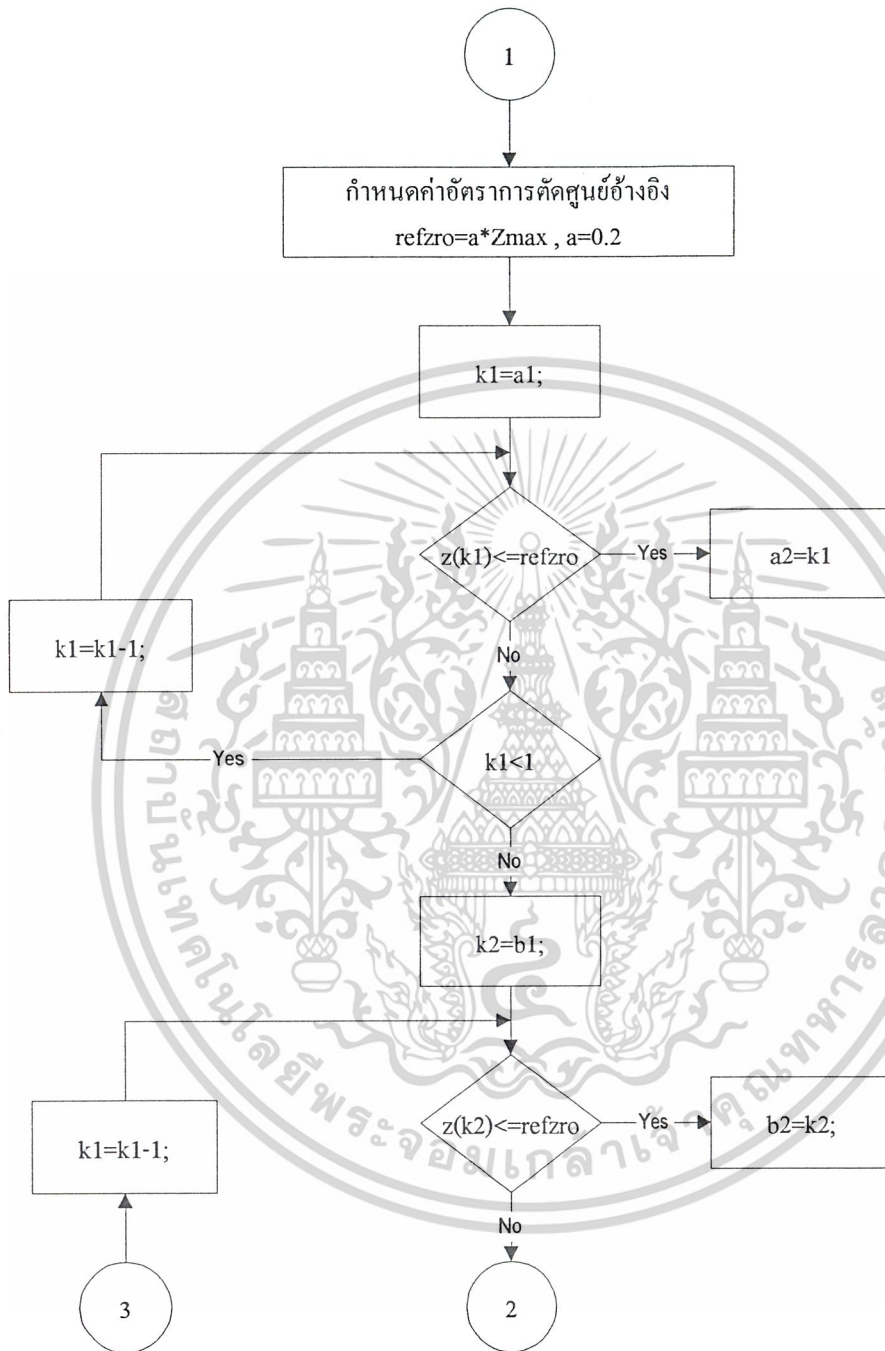
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมีโฟลวชาร์ตดังรูปที่ 3.10



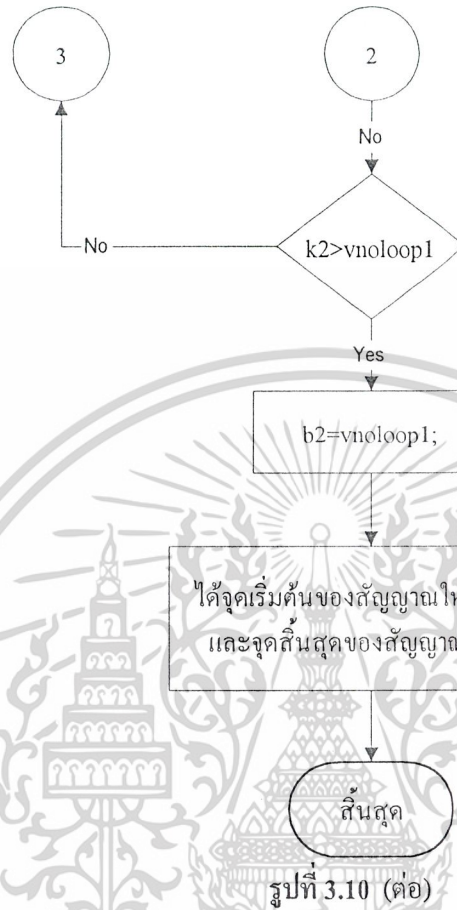
รูปที่ 3.10 แสดงโฟลวชาร์ตการหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงด้วยวิธีการหาอัตราการตัด
ศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



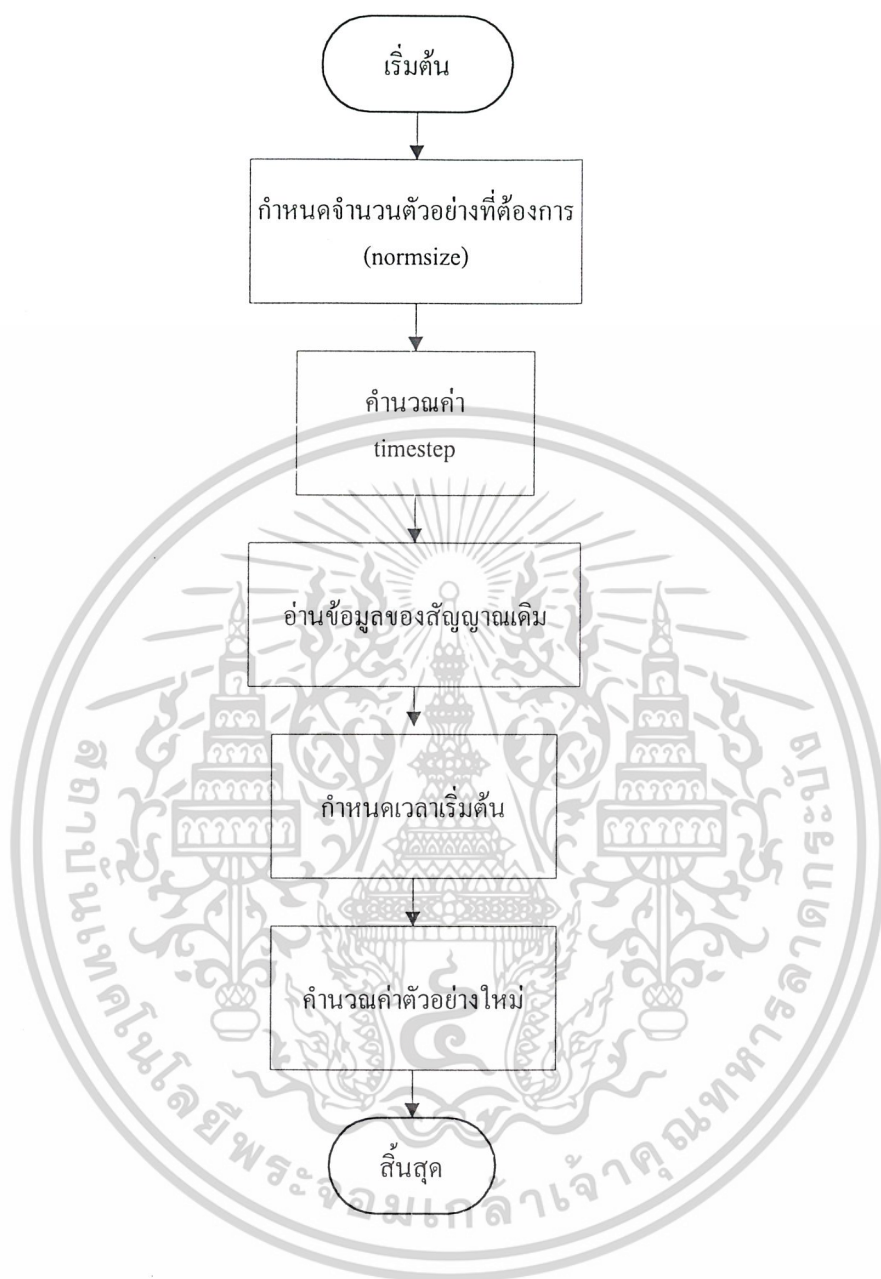
รูปที่ 3.10 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การนอร์มอลไลซ์มีขั้นตอนดังรูปที่ 3.11

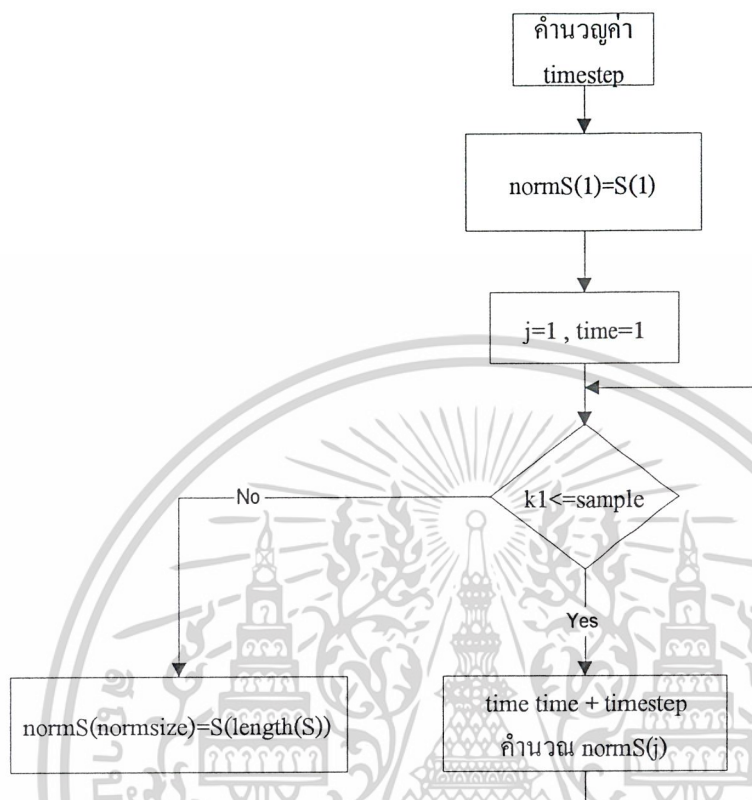
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แสดงขั้นตอนการนอร์มอลไลซ์สัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมีโฟลวชาร์ตดังรูปที่ 3.12



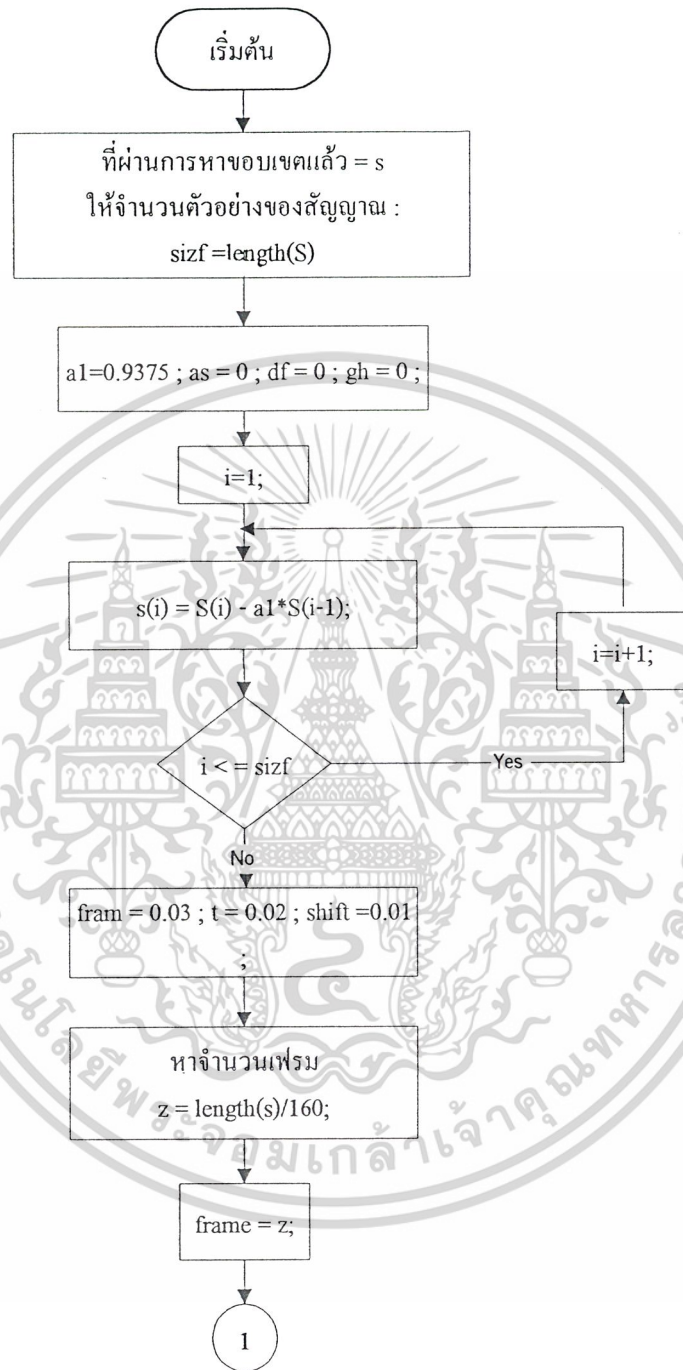
รูปที่ 3.12 แสดงโฟลวชาร์ตการนอร์มอลไลซ์ของสัญญาณ

3.3.2 การหาค่าพารามิเตอร์

ในการหาพารามิเตอร์นั้น เราจะแยกเป็น 3 ชนิดดังนี้

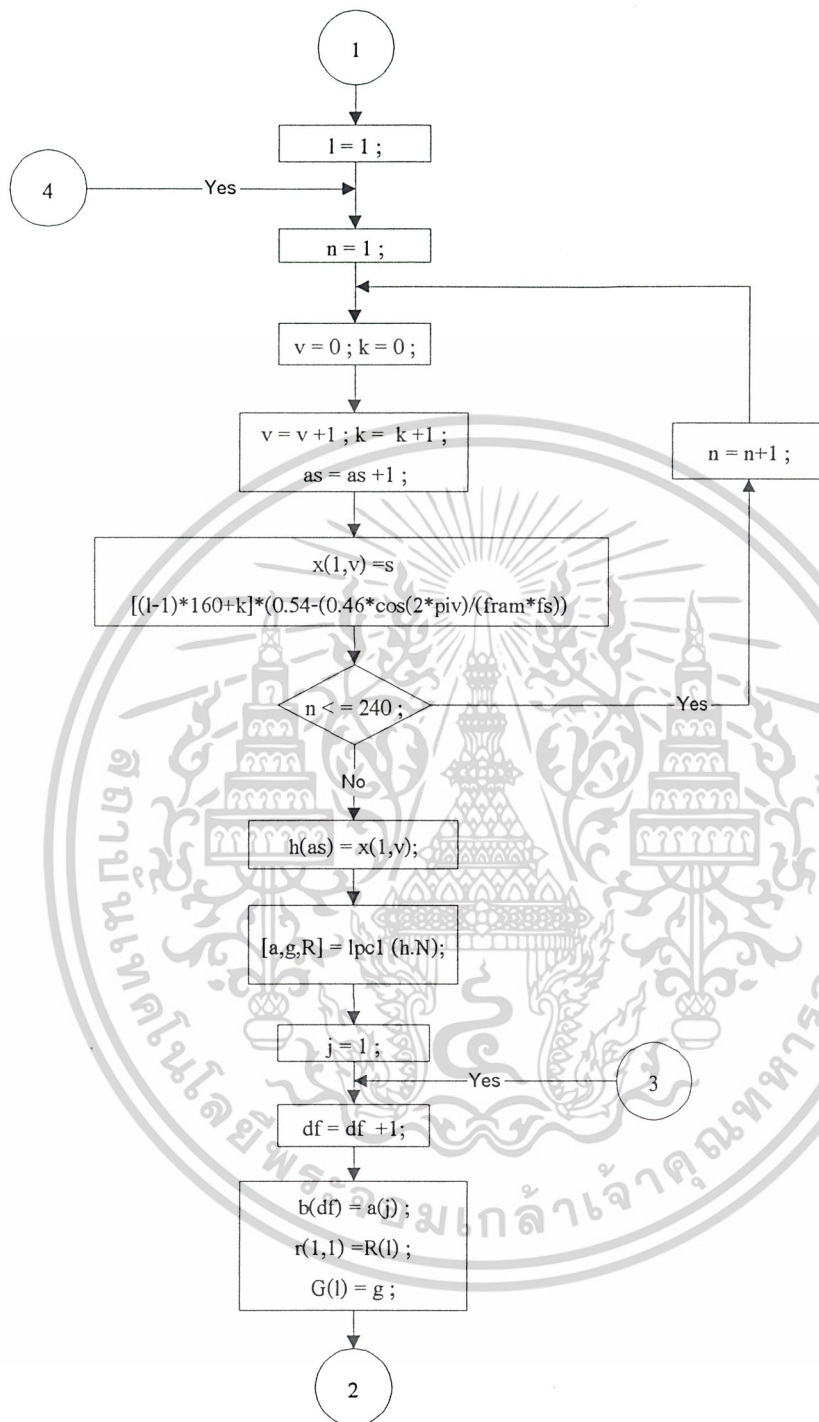
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าสัมประสิทธิ์การประมาณเชิงเส้น



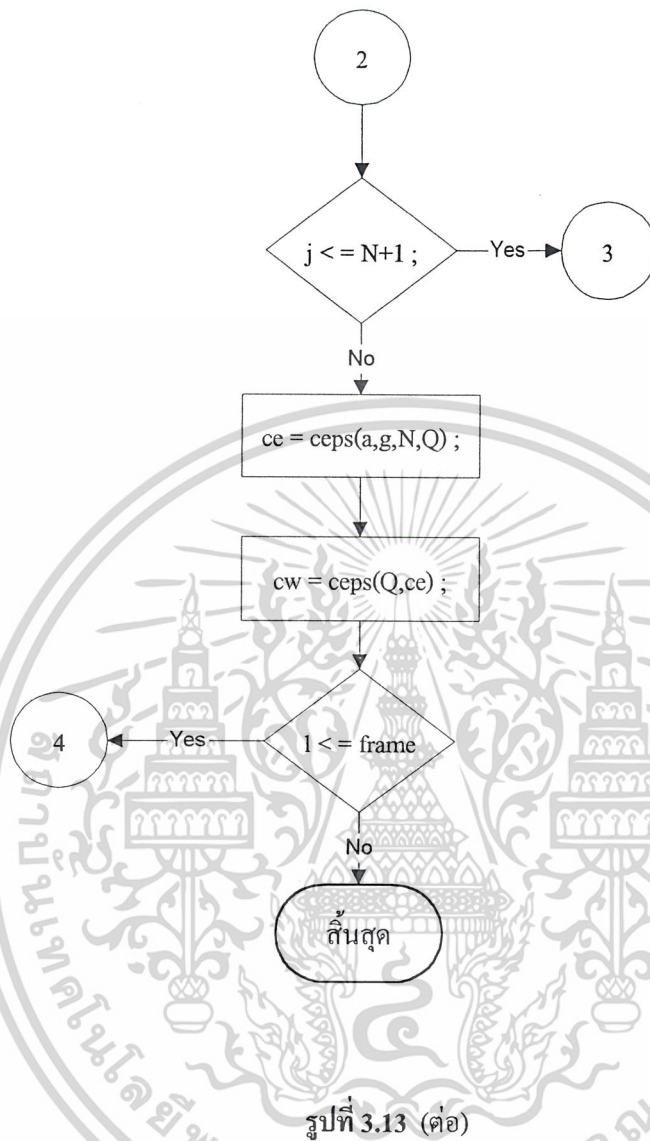
รูปที่ 3.13 แสดงโฟลวชาร์ตการหาค่าสัมประสิทธิ์การประมาณเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



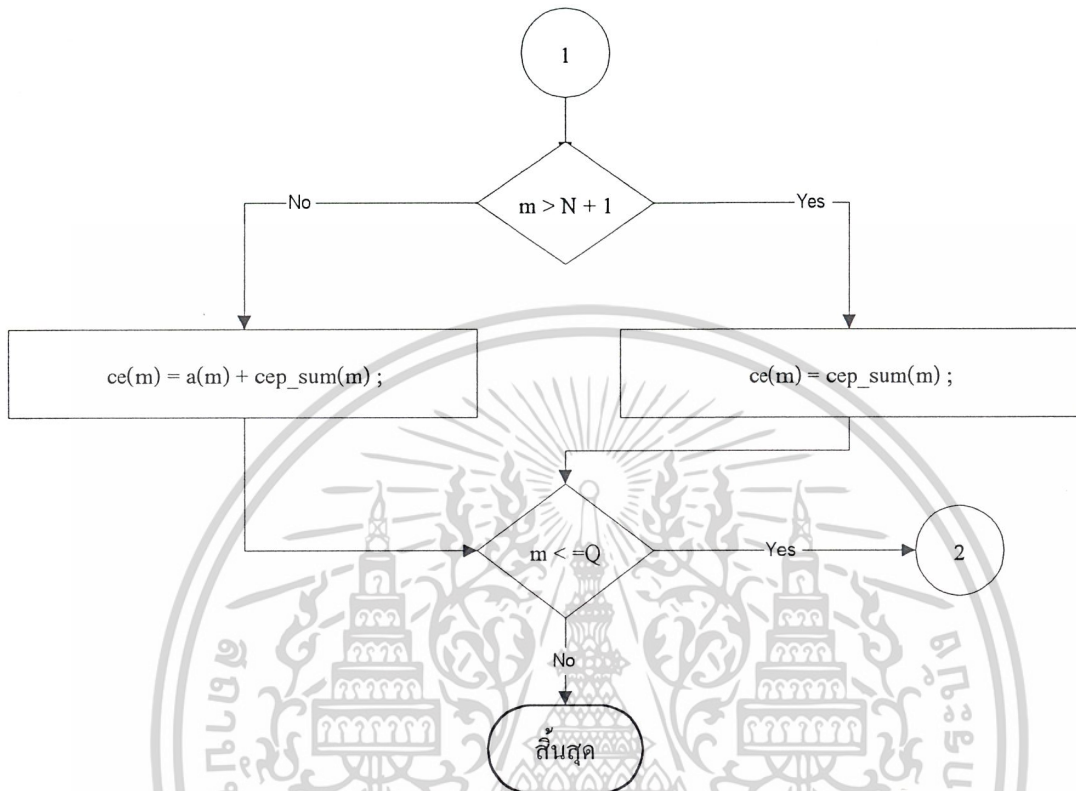
รูปที่ 3.13 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

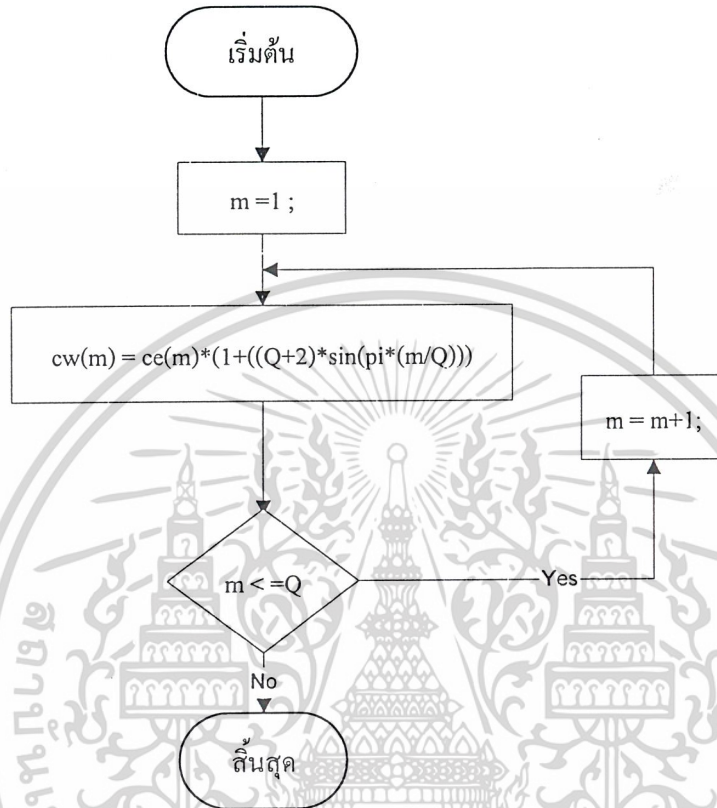
การหาค่าสัมประสิทธิ์เซปสตรีม



รูปที่ 3.14 แสดงโฟลวชาร์ทการหาค่าสัมประสิทธิ์เซปสตรีม

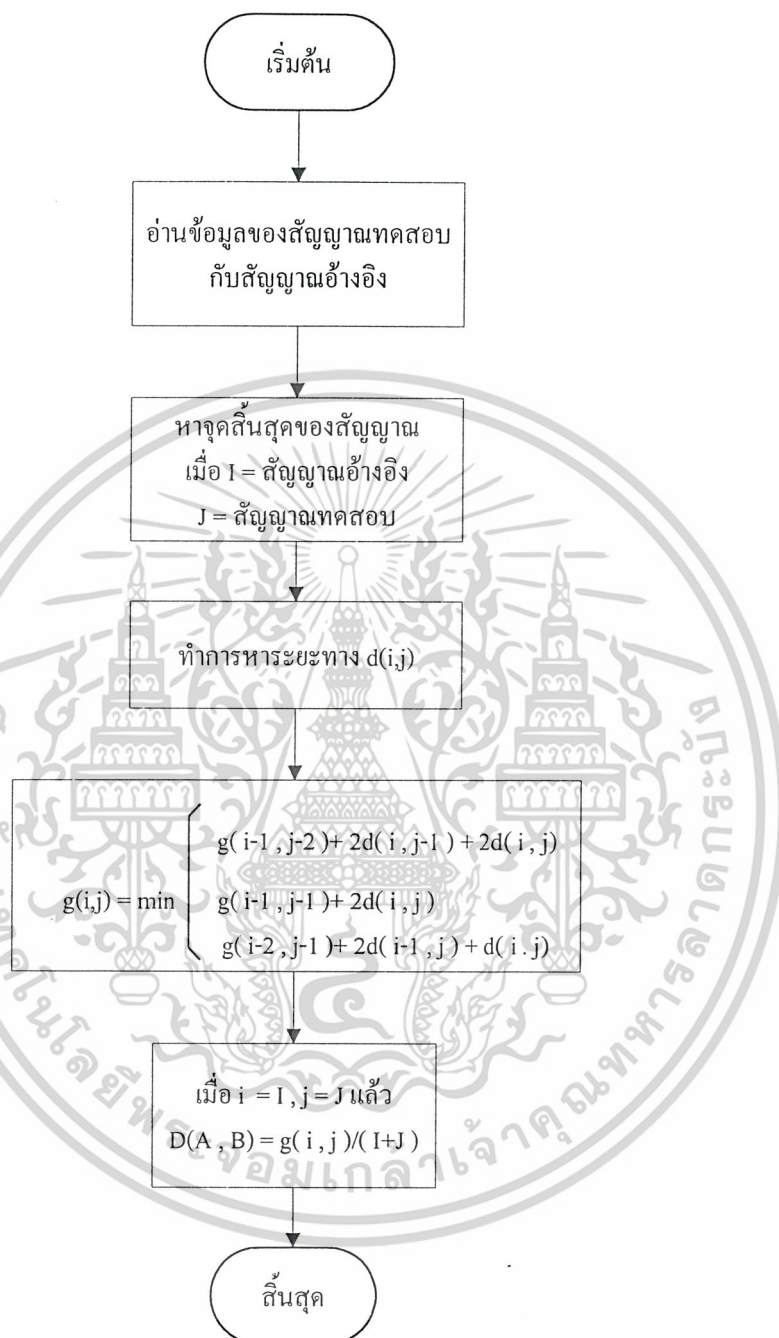
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนัก



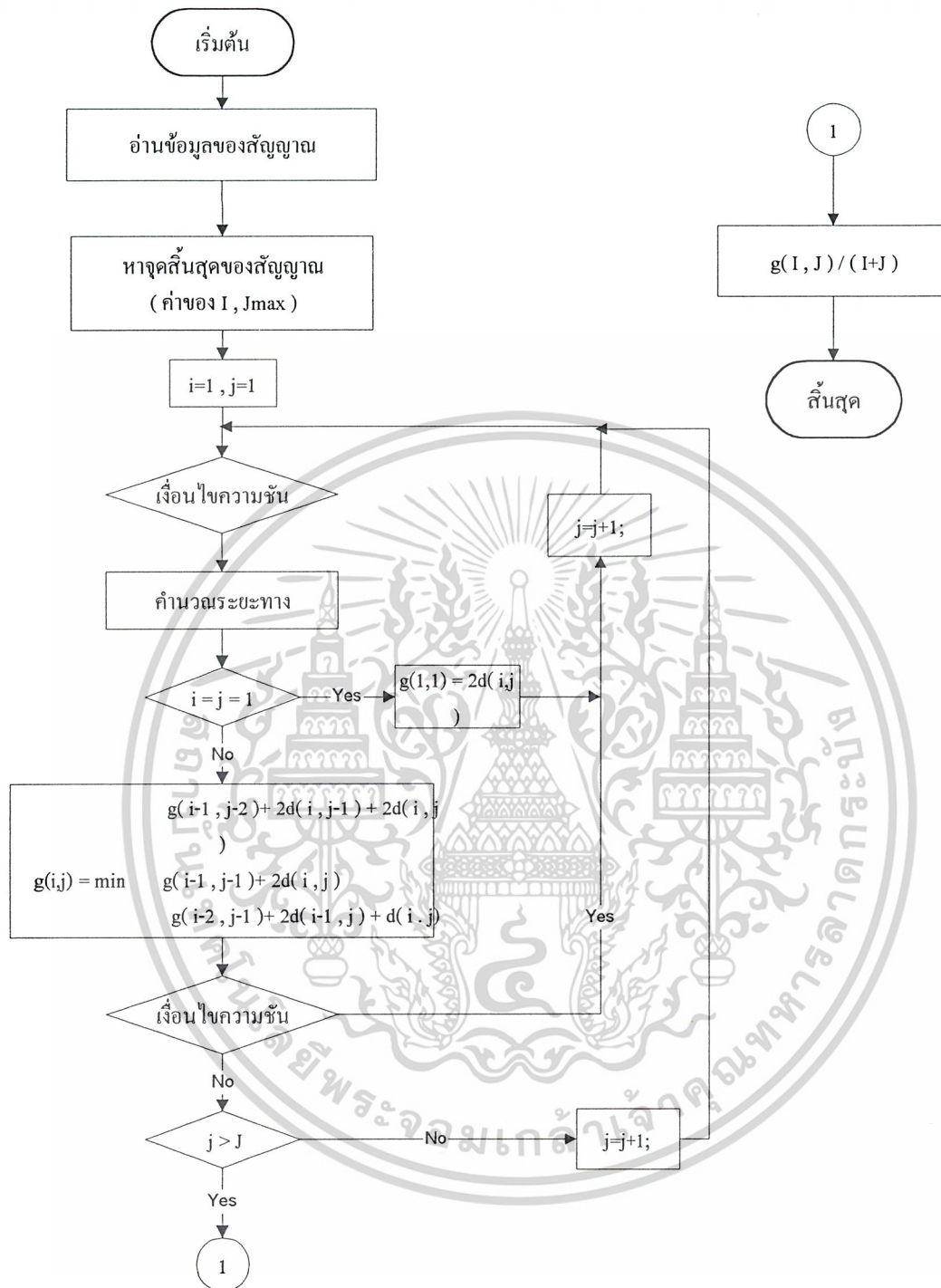
รูปที่ 3.15 แสดงโฟลวชาร์ทการหาค่าสัมประสิทธิ์เซปสตรีมถ่วงน้ำหนัก

การเปรียบเทียบความคล้ายคลึงของรูปแบบ โคนามิคไทม์วาร์ปปีง (Dynamic Time Wrapping)



รูปที่ 3.16 แสดงขั้นตอนไดนามิกไทม์วาร์ปิ้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.17 แสดงผังงานการทำไดนามิกโปรแกรมอาร์บี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 โปรแกรมการแปลงเอ็ม-ไฟล์เป็นไฟล์ชนิดเอ็กซ์คิวทีเอเบิล

เมื่อนำโปรแกรมส่งข้อมูลออกพอร์ตที่ถูกแปลงเป็นไดนามิกคอลเลกชันไลบรารีส์เข้ามารวมไว้ในโปรแกรมรู้จำเสียงพูด เพื่อให้ผลที่ได้จากการรู้จำเสียงพูดสามารถแสดงผลออกทางพอร์ตที่ต้องการได้แล้ว

ขั้นต่อไปก็คือการนำเอาโปรแกรมแมทแลบ(เอ็ม-ไฟล์)ที่ได้นั้น มาแปลงอีกครั้งเพื่อให้ความสะดวกแก่โปรแกรมภาษาเคลไพในการเรียกใช้งาน

การแปลงเอ็ม-ไฟล์ให้เป็นไฟล์ชนิดเอ็กซ์คิวทีเอเบิลนั้นทำได้โดยไม่ต้องทำการเซ็ทอัพ (Set Up) ใดๆ เหมือนกับที่ทำในเม็กซีไฟล์เลย ใช้เพียงฟังก์ชัน mcc ของแมทแลบคอมไพเลอร์ ในการแปลงเท่านั้น

โดยมีวิธีการแปลงดังนี้

```
>> mcc -m filename(เป็นเอ็ม-ไฟล์)
```

คำสั่งนี้จะต้องพิมพ์ลงไปบนคอมมานด์วินโดว์ของแมทแลบเพื่อให้แมทแลบคอมไพเลอร์ซึ่งเป็นตัวคอมไพเลอร์ในแมทแลบ ทำการเลือกเรปเปอร์ไฟล์ที่เหมาะสมและทำการคอมไพล์เอ็ม-ไฟล์นั้นๆ ให้เป็นเอ็กซ์คิวทีเอเบิลไฟล์ที่ต้องการ

ข้อควรระวังสำหรับขั้นตอนนี้ก็คือตัวโปรแกรมในเอ็ม-ไฟล์ที่เราใช้นั้นถ้ามีคำสั่ง load อยู่ ต้องระวังให้มีการใช้คำสั่ง load ให้อยู่ในรูปแบบที่ถูกต้องดังนี้

```
>> load ('filename.mat','variable')
```

โดยที่ 'variable' ก็คือตัวแปรที่เราต้องการอ่านค่าออกมานั่นเอง

ถ้าใช้วิธี load แบบธรรมดาคือ

```
>> load variable
```

เมื่อทำการแปลงเอ็ม-ไฟล์นี้เป็นไฟล์แบบเอ็กซ์คิวทีเอเบิลแล้วก็จะทำให้เกิดการเออร์เรอร์ขึ้นในกระบวนการแปลงได้

3.5 โปรแกรมรับโทรศัพท์อัตโนมัติ

โปรแกรมรับโทรศัพท์ที่ใช้เป็นคอมโพเนนท์ที่เขียนด้วยภาษาเคลไพเรียกคอมโพเนนท์นี้ว่า วอยส์โมเด็มไลบรารี (Voice Modem Library)

วอยส์โมเด็มไลบรารีเป็นเซ็ทของคอมโพเนนท์ที่เขียนด้วยภาษาเคลไพ ซึ่งคอมโพเนนท์นี้จะใช้ในการสร้างเครื่องรับโทรศัพท์อัตโนมัติ และ ประโยชน์อีกหลายอย่าง โดยใช้ความสามารถของโมเด็มภายนอกที่เราใช้เป็นหลัก

วอยส์โมเด็มไลบรารีเป็นคอมโพเนนท์ที่มีความสามารถในการ เล่นและอัดเสียงของผู้ที่โทรศัพท์เข้ามา, รับความถี่ทีเอ็มเอฟ (DTMF) ที่ถูกส่งมาตามสายโทรศัพท์ , รับข้อมูลและแฟกซ์ (fax) , หมุนโทรศัพท์เพื่อการโทรออก, รอจนมีผู้มารับสาย, ส่งความถี่เฉพาะไปให้คู่สนทนาอีกฝั่งให้ได้ยิน ได้ หรือ ส่งสัญญาณ คีทีเอ็มเอฟออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

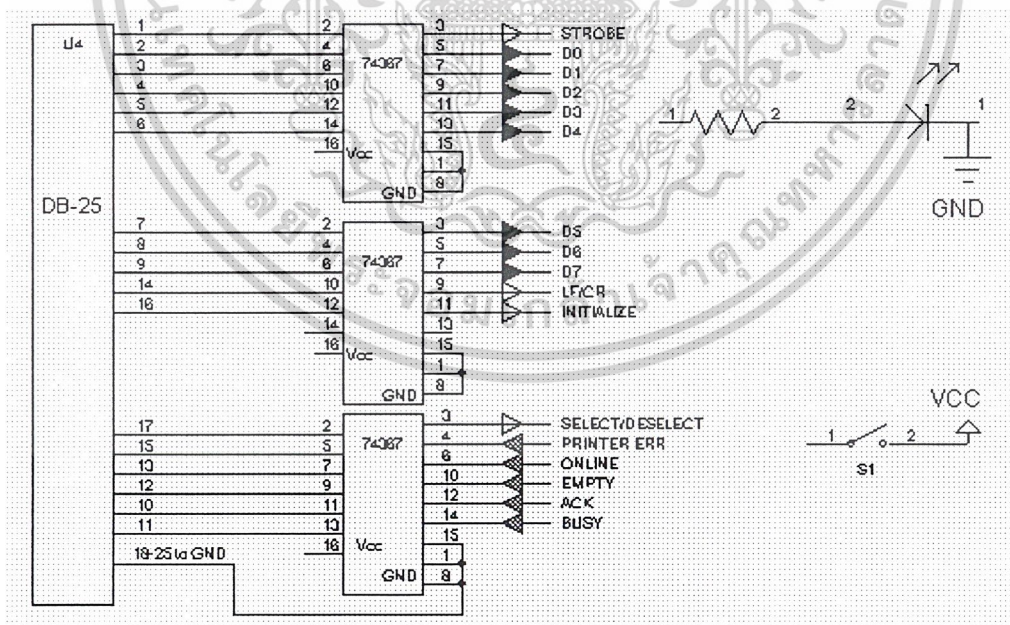
ในการใช้งานของคอมพิวเตอร์นี่ก็เพียงแต่ติดตั้ง (Install) คอมพิวเตอร์หลังจากนั้นก็เลือกใช้เดโมรับโทรศัพท์อัตโนมัติ (AutoAnswer Demo) จากนั้นก็ทำการตั้งค่าของโมเด็มที่เราใช้ว่าเป็นชนิดและยี่ห้ออะไร ทำการกำหนดพอร์ตของโมเด็ม และมีรายละเอียดในการ โปรแกรมอีกมากมายซึ่งสามารถศึกษาได้ในส่วนของภาคผนวก

ปัญหาอย่างหนึ่งของการใช้คอมพิวเตอร์โมเด็มไลบรารีคือ เสียงที่คอมพิวเตอร์อัดได้จะเป็นเสียงที่อัดได้ผ่านทางโมเด็ม ซึ่งโมเด็มที่ใช้ในการทำปฏิญาณพนธ์นี้จะให้ไฟล์เสียงที่อัดได้เป็นไฟล์นามสกุล .mmd แต่ในส่วนของการโปรแกรมรับเสียงพูดที่ใช้นั้นต้องใช้ไฟล์เสียงที่อยู่ในรูปของไฟล์นามสกุล .wav ทำให้ต้องมีการนำไฟล์เสียงที่เข้ามาจากสายโทรศัพท์สู่โมเด็มนี้ส่งต่อไปยังการ์ดเสียงเพื่อทำการบันทึกเสียงเป็นไฟล์นามสกุล .wav ซึ่งสามารถทำได้ด้วยการต่อสายไลน์อิน (line in) จากตัวโมเด็มเข้าสู่การ์ดเสียงโดยตรง แล้วเขียนโปรแกรมไปคอมพิวเตอร์ทำให้การ์ดเสียงทำการอ่านข้อมูลเสียงนี้ต่อจากโมเด็ม ไปเก็บไว้ในรูปของไฟล์นามสกุล .wav เพื่อให้โปรแกรมรับเสียงพูดสามารถนำไปใช้ได้

3.6 วงจรแสดงผลออกทางพอร์ตขนาน

วงจรแสดงผลออกทางพอร์ตขนานที่ใช้ในปฏิญาณพนธ์นี้สามารถนำไปใช้กับพอร์ตขนานของคอมพิวเตอร์ได้โดยไม่ต้องมีการ โปรแกรมหรือว่าเซ็ทอะไรเพิ่มเติม

การต่อวงจรแสดงดังรูปที่ 3.18



รูปที่ 3.18 วงจรแสดงผลออกทางพอร์ตขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

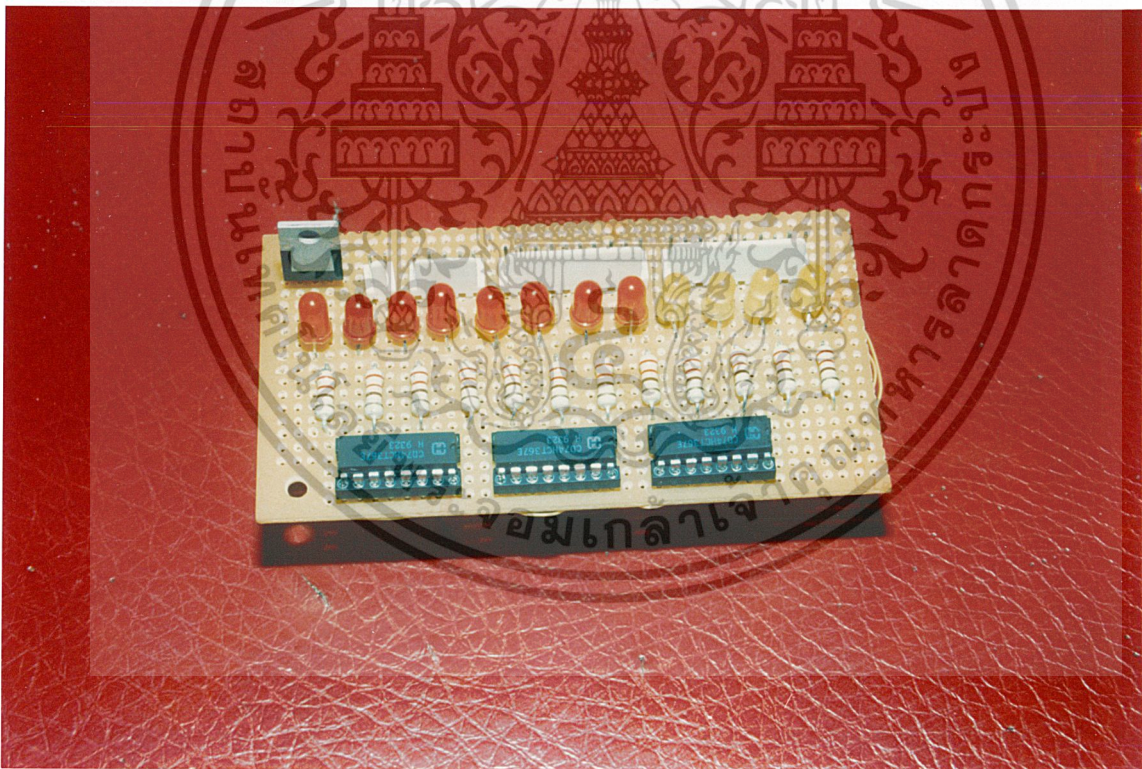
จากรูปที่ 3.18 DB-25 คือ หัวพอร์ตตัวผู้แบบ 25 พิน หัวลูกศรสีดำแสดงถึงขาข้อมูลของพอร์ตขนานเมื่อผ่าน 74367 ออกมาแล้ว ลูกศรหัวสีขาวแสดงถึงขาควบคุมของพอร์ตขนานเมื่อผ่าน 74367 ลูกศรหัวสีเทาแสดงถึงขาแสดงสถานะของพอร์ตขนานซึ่งจะผ่าน 74367 กลับเข้าสู่เครื่องคอมพิวเตอร์

ดังนั้นในการต่อวงจรจริงจะทำการต่อลูกศรที่แสดงในรูปวงจรด้วยแอลอีดี (แสดงทางขวามือของรูปวงจร) ผ่านตัวต้านทาน 220 โอห์ม

ไอซี (IC) 74367 ทำหน้าที่เป็นบัฟเฟอร์ป้องกัน เนื่องจากว่าเมนบอร์ด (Main Board) ในคอมพิวเตอร์นั้นจะสามารถทนต่อกระแสไฟฟ้าได้เพียง 5 มิลลิแอมป์ (milliAmp.) ดังนั้นจึงไม่ควรที่จะต่อพอร์ตขนานของคอมพิวเตอร์เข้ากับเครื่องใช้ไฟฟ้าโดยตรง ควรจะต่อผ่าน 74367 เสียก่อน เพราะเมื่อเกิดกรณีที่มีกระแสมาเกินไปไหลเข้าสู่เมนบอร์ดเกิดขึ้น 74367 ก็จะทำหน้าที่ตัดไฟทันที เปรียบไปก็เหมือนกับว่า 74367 ทำหน้าที่เป็นฟิวส์ (Fuse) ทางอิเล็กทรอนิกส์นั่นเอง

เมื่อเราทำการต่อวงจรตามรูปที่ 3.18 เรียบร้อยแล้วก็สามารถนำไปต่อกับพอร์ตขนานที่ด้านหลังของปริ้นท์เตอร์ได้โดยตรงเลย

วงจรแสดงผลออกทางพอร์ตขนานเมื่อต่อเสร็จเรียบร้อยแล้วจะแสดงดังรูปที่ 3.19



รูปที่ 3.19 วงจรแสดงผลออกทางพอร์ตขนานเมื่อทำการต่อเสร็จเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองที่ 1

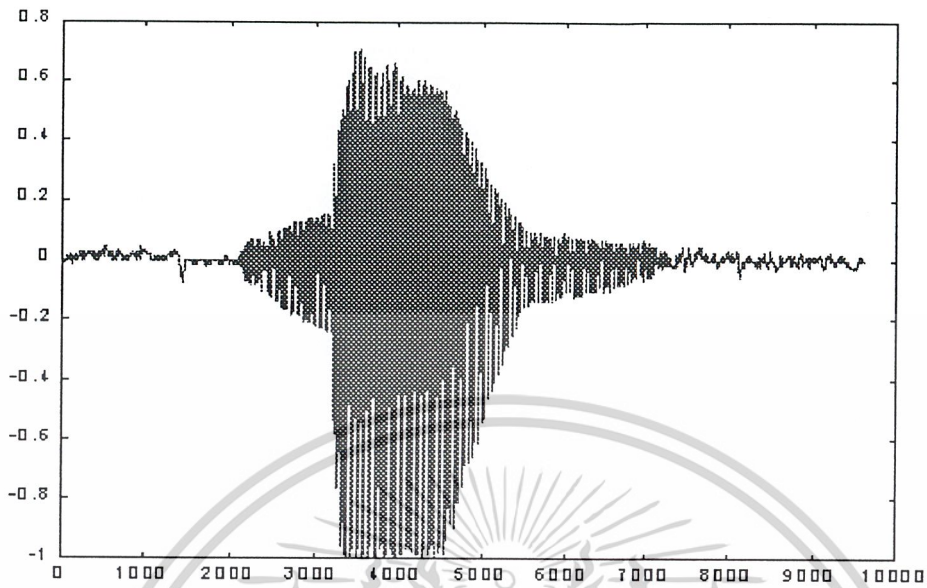
เป็นการทดลองเพื่อแสดงถึงความแตกต่างของรูปสัญญาณเสียงหนึ่งถึงสี่ และแสดงแสดงถึงผลของสัญญาณเสียงที่ถูกจำกัดขอบเขตด้วยการเปรียบเทียบพลังงานและการตัดศูนย์ของเสียงหนึ่งถึงสี่

โดยการนำเสียงที่อัดไว้ด้วยคอมพิวเตอร์ที่อัดเสียงอัตโนมัติด้วยเคลป์ เป็นไฟล์ .wav มาอ่านด้วยโปรแกรม Cool Edit จากนั้นนำผลที่ได้มาแสดงเป็นรูปสัญญาณ แล้วนำสัญญาณเสียงหนึ่งถึงสี่ที่อัดไว้นั้นมาผ่านขั้นตอนการจำกัดขอบเขตด้วยวิธีการเปรียบเทียบค่าพลังงานโดยฟังก์ชัน detect1 และ วิธีการเปรียบเทียบอัตราการตัดศูนย์โดยฟังก์ชัน zrocrs1

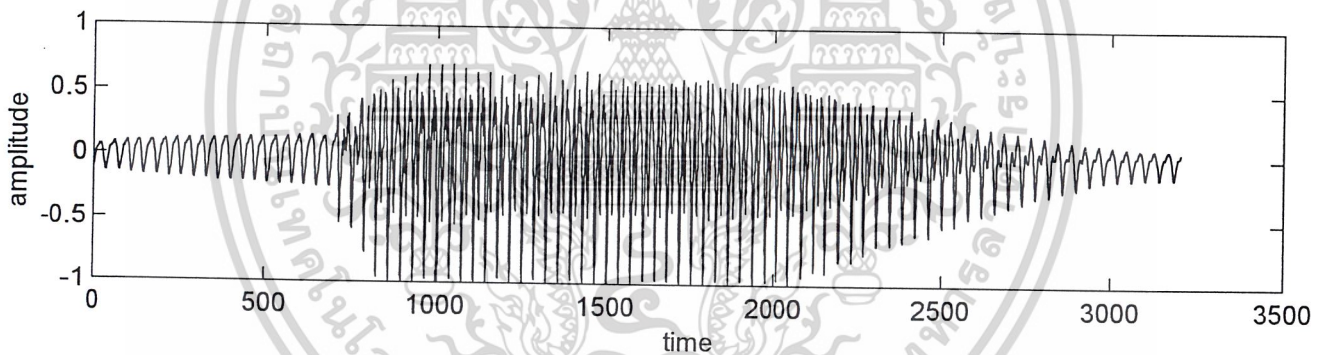
ผลการทดลอง



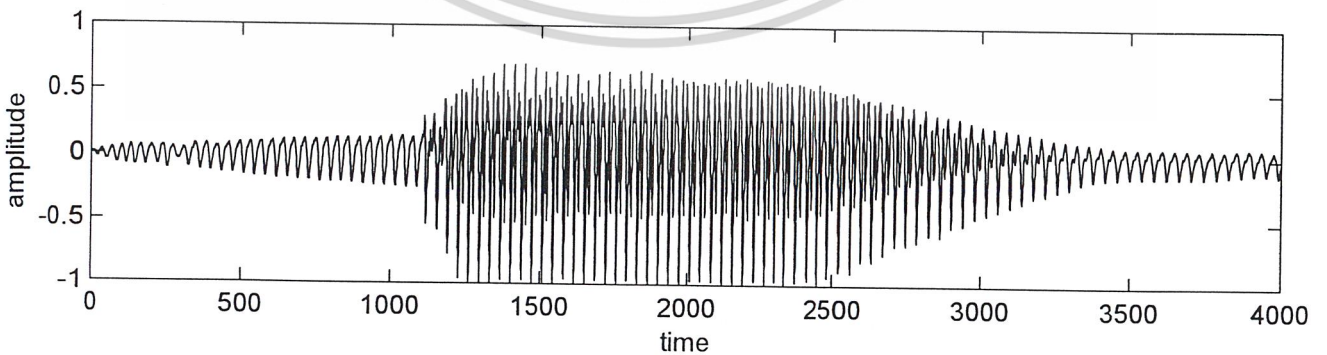
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 แสดงสัญญาณเสียงหนึ่ง

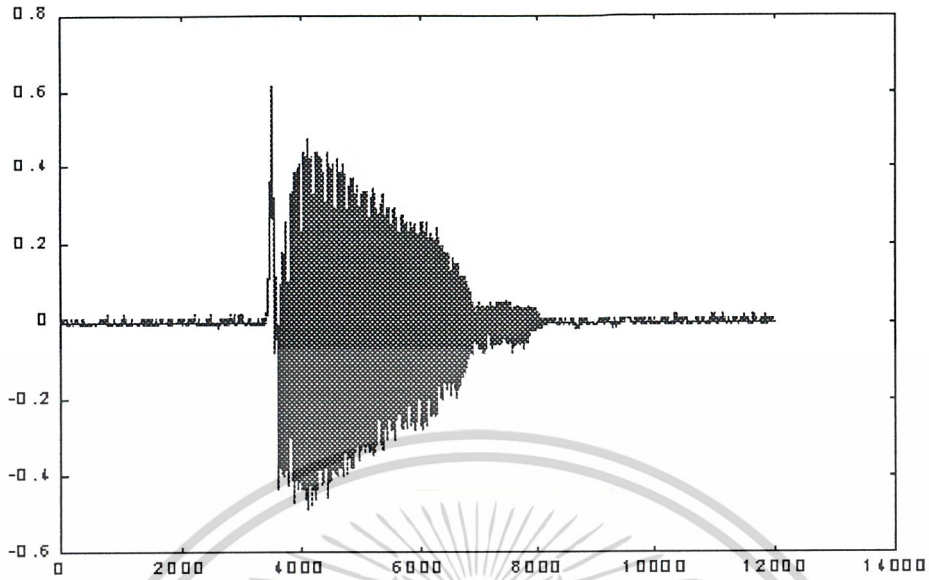


รูปที่ 4.5 สัญญาณเสียงหนึ่งเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบค่าพลังงาน

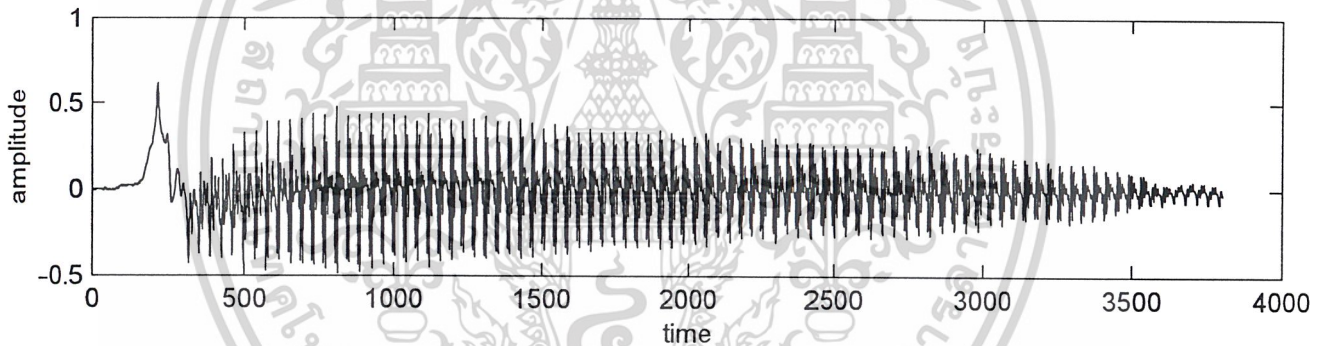


รูปที่ 4.6 สัญญาณเสียงหนึ่งเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบอัตราการตัดศูนย์

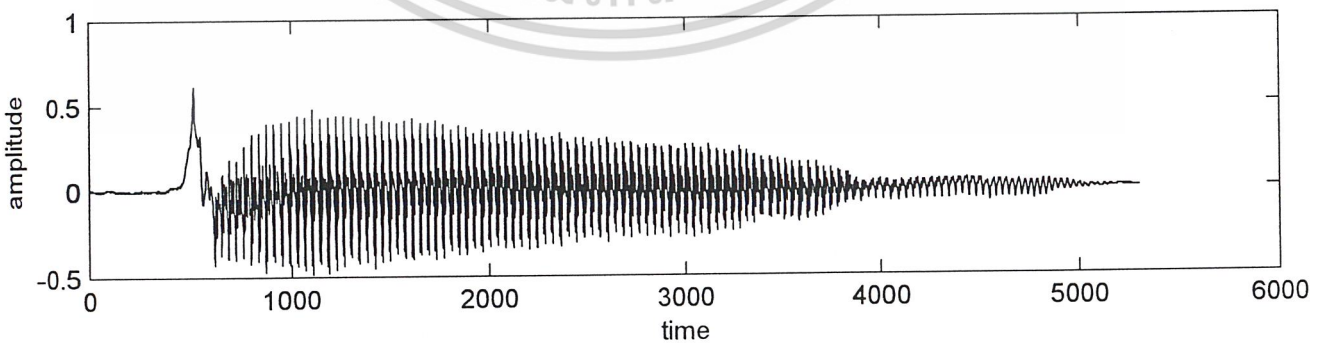
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงสัญญาณเสียงสอง

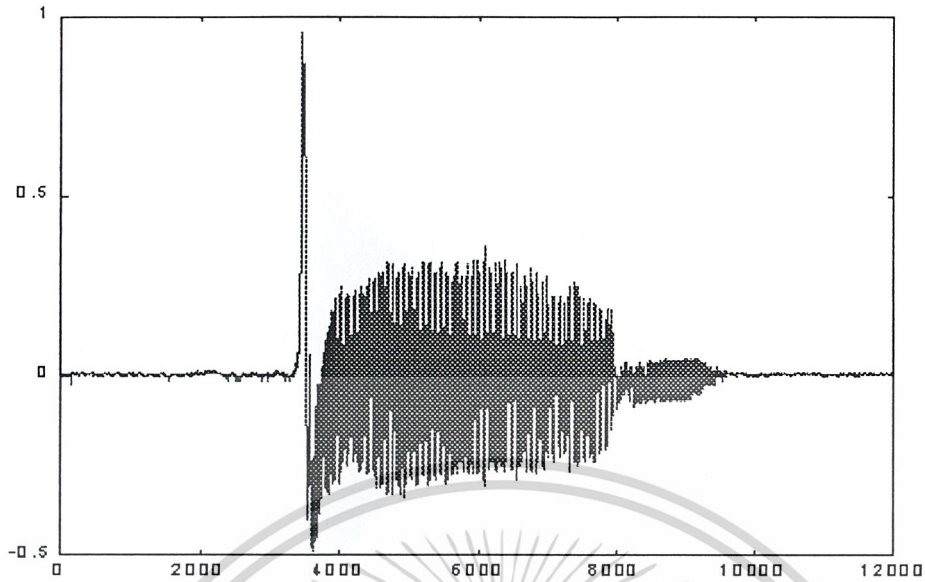


รูปที่ 4.7 สัญญาณเสียงสองเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบค่าพลังงาน

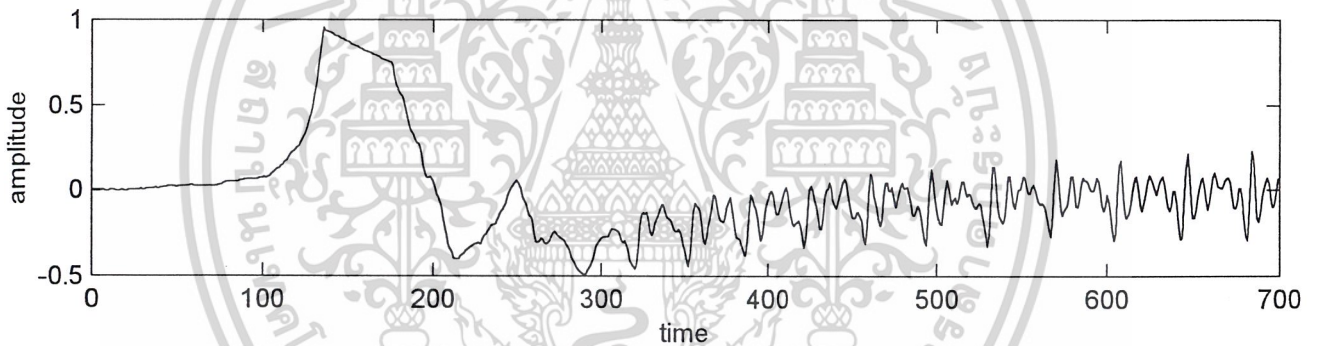


รูปที่ 4.8 สัญญาณเสียงสองเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบอัตราการตัดศูนย์

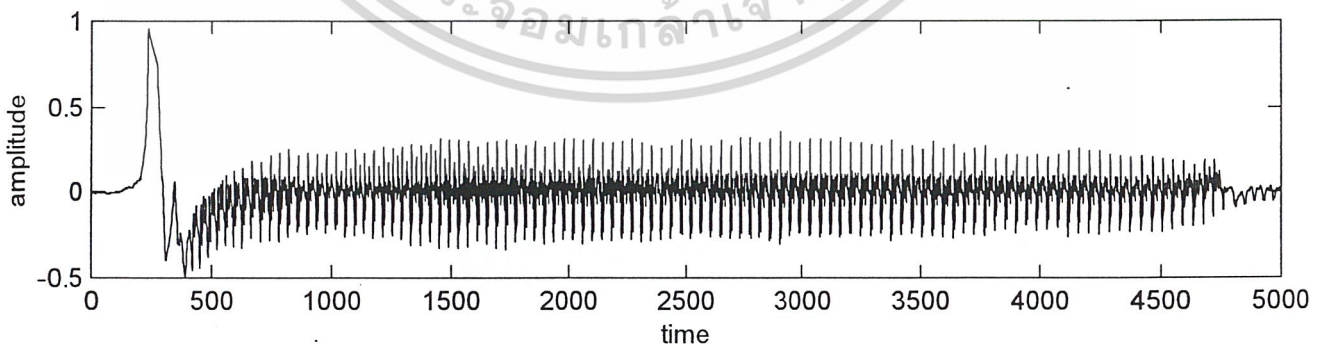
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงสัญญาณเสียงสาม



รูปที่ 4.9 สัญญาณเสียงสามเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบค่าพลังงาน



รูปที่ 4.10 สัญญาณเสียงสามเมื่อถูกกำหนดจุดเริ่มต้นและจุดสิ้นสุดด้วยวิธีการเปรียบเทียบอัตราการตัดศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองที่ 2

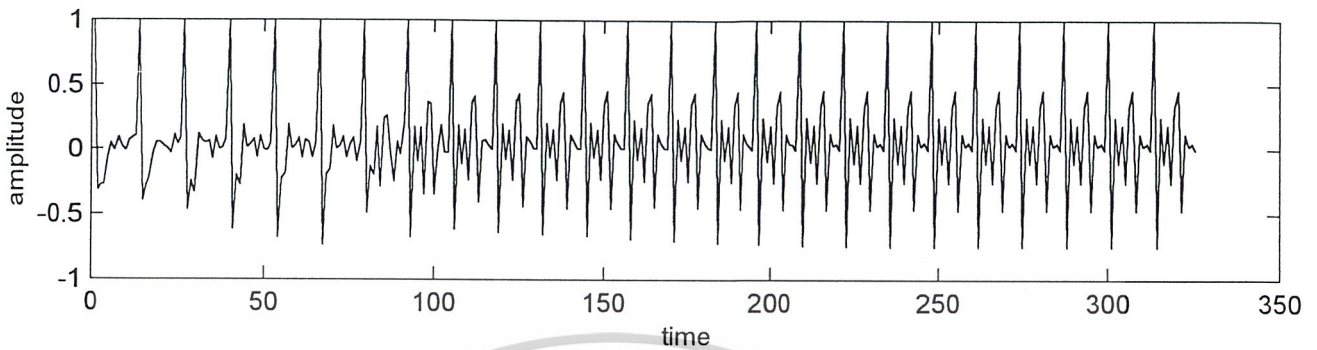
เป็นการทดลองเพื่อแสดงถึงความแตกต่างของสัมประสิทธิ์การประมาณเชิงเส้น, สัมประสิทธิ์เซปสตรีม และ สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนัก ของเสียงหนึ่งถึงสี่

โดยการนำไฟล์เสียงหนึ่งถึงสี่ที่ได้อัปไว้ไปผ่านกระบวนการของฟังก์ชัน detect1, zocrs1, normalize, และ levin-durbin ตามลำดับเพื่อให้ได้ค่าสัมประสิทธิ์ทั้งสาม แล้วจึงนำค่าสัมประสิทธิ์ทั้งสามของแต่ละเสียงมาทำการพล็อตเป็นกราฟดังนี้

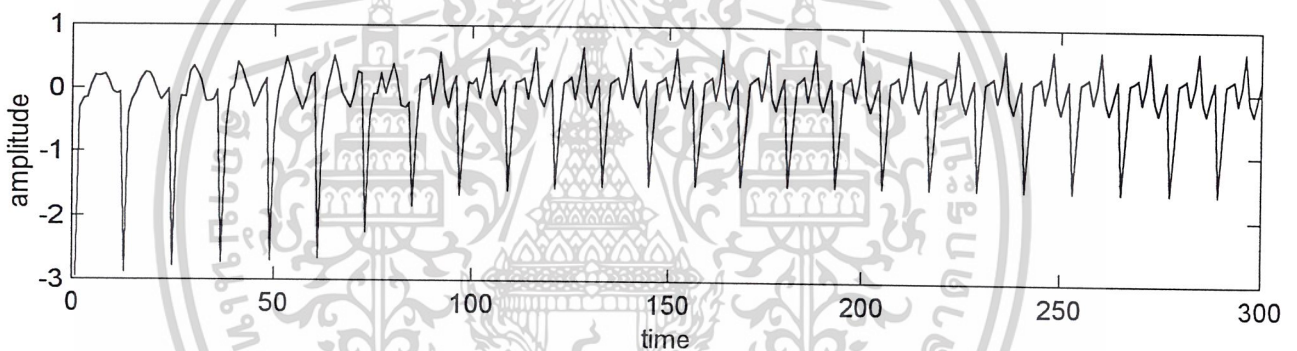
ผลการทดลอง



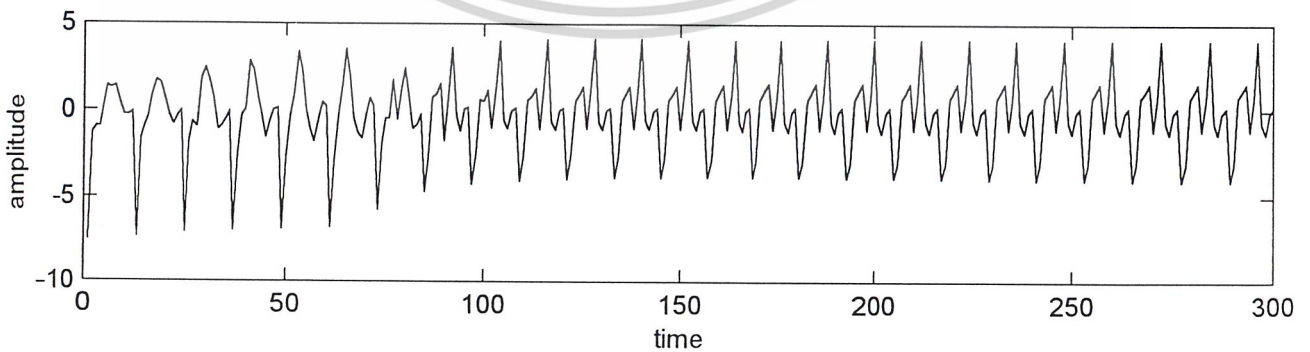
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงหนึ่ง

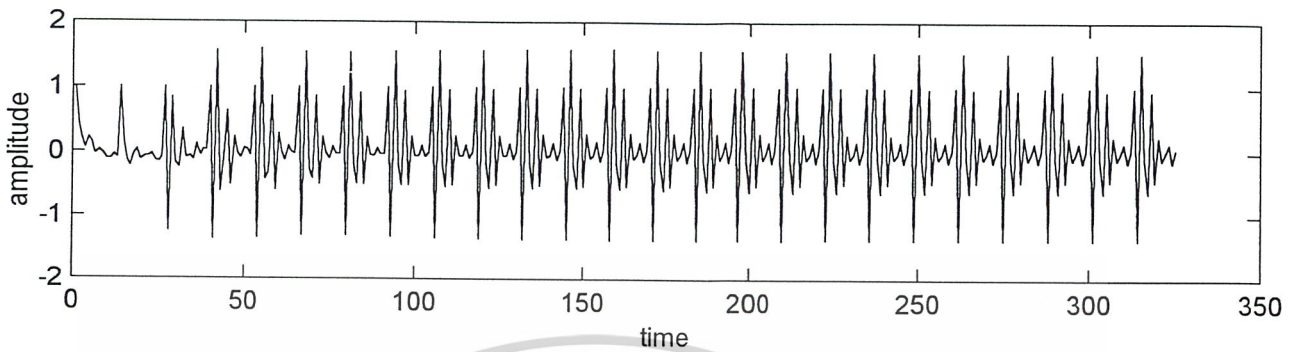


รูปที่ 4.14 สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงหนึ่ง

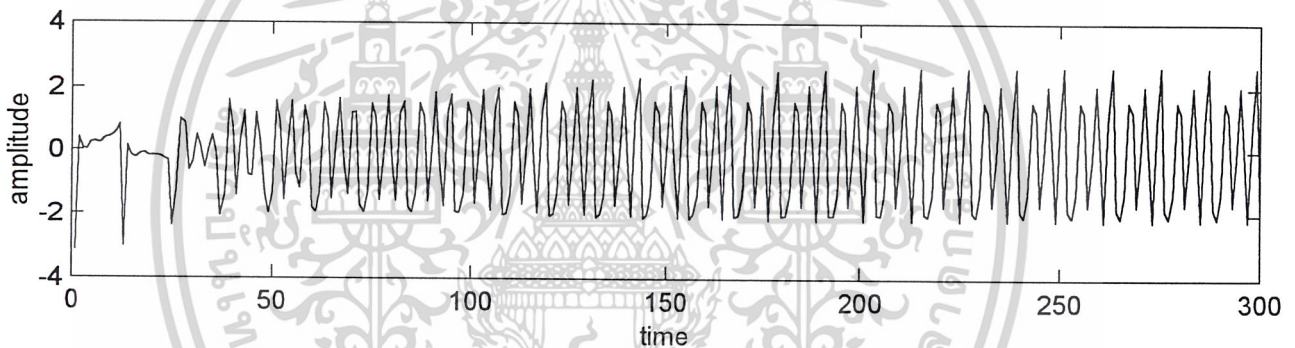


รูปที่ 4.15 สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงหนึ่ง

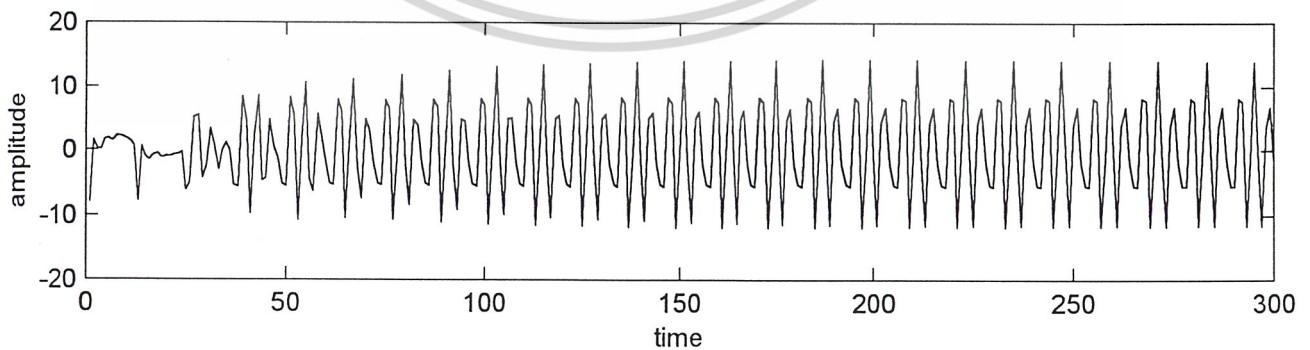
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงสอง

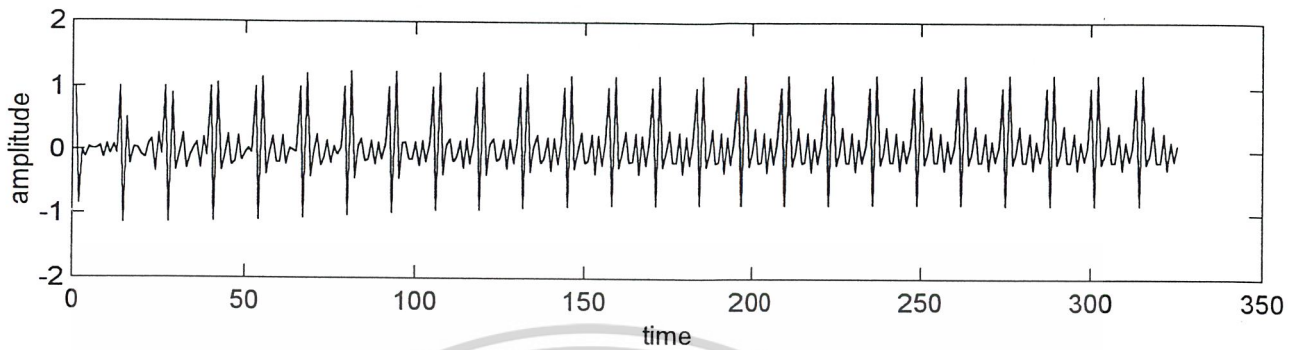


รูปที่ 4.17 สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงสอง

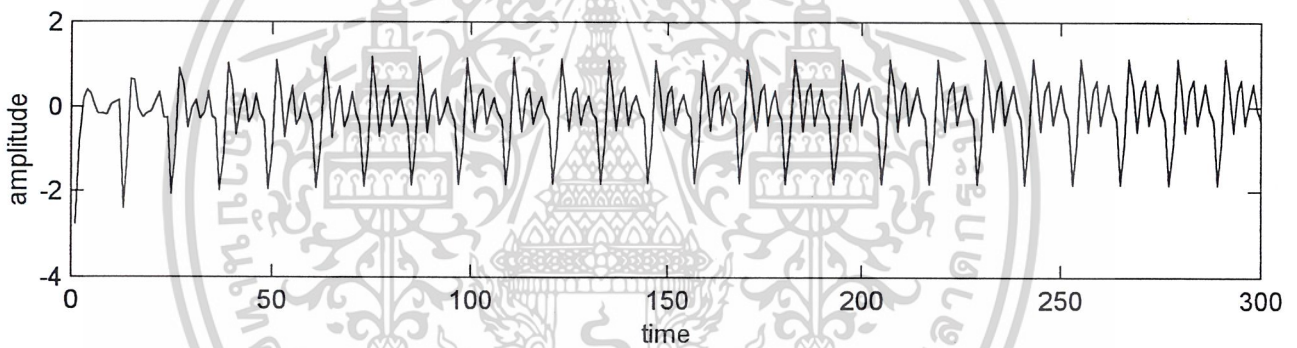


รูปที่ 4.18 สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสอง

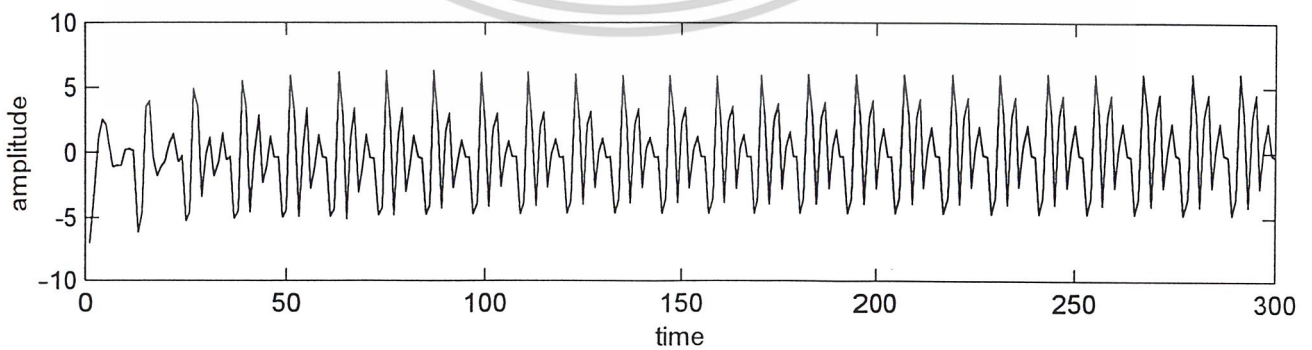
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงสาม

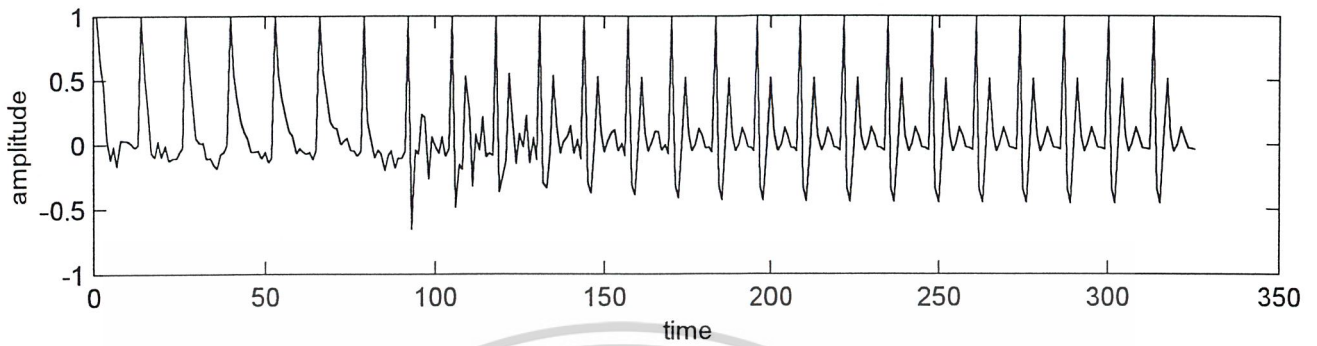


รูปที่ 4.20 สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงสาม

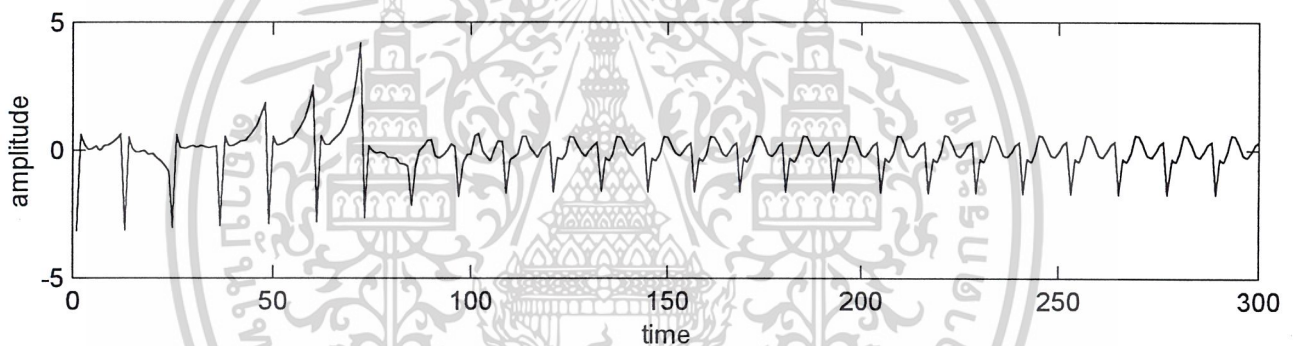


รูปที่ 4.21 สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสาม

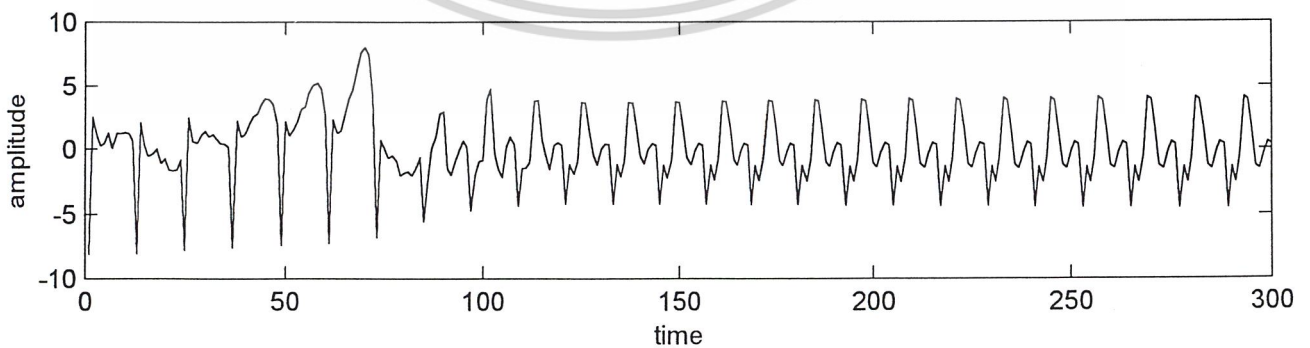
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 สัมประสิทธิ์การประมาณเชิงเส้นของสัญญาณเสียงสี่



รูปที่ 4.23 สัมประสิทธิ์เซปสตรีมของสัญญาณเสียงสี่



รูปที่ 4.24 สัมประสิทธิ์เซปสตรีมถ่วงน้ำหนักของสัญญาณเสียงสี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองที่ 3

เป็นการทดลองเพื่อหาถึงอัตราส่วนความถูกต้องในการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์

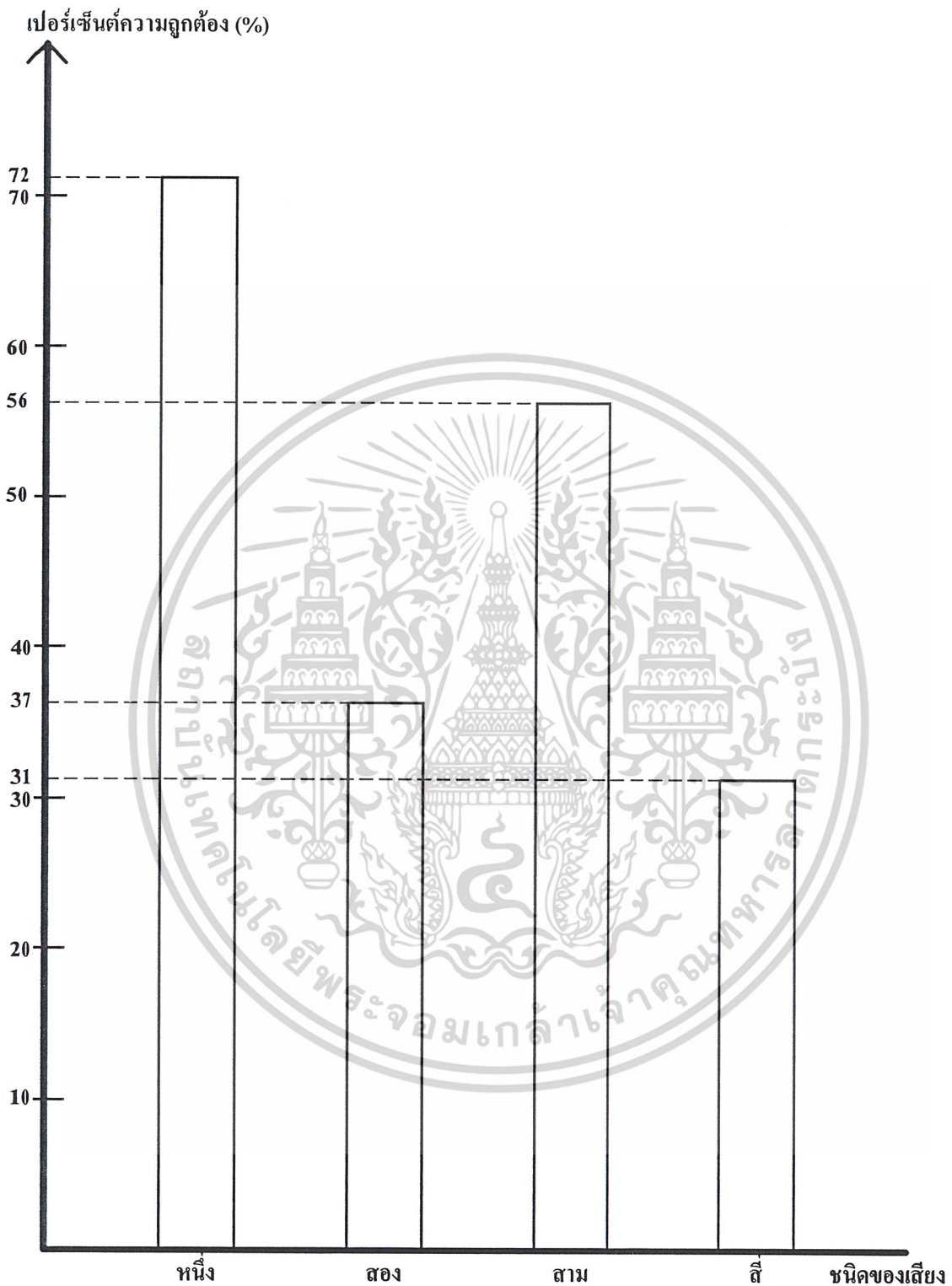
โดยทำการเก็บค่าสัมประสิทธิ์สัญญาณเสียงอ้างอิงจากผู้พูดคนเดียวจำนวน 20 เสียงตัวอย่าง จากนั้น ทำการทดสอบโดยให้ผู้พูดคนเดิมโทรศัพท์เข้ามาสั่งงานเครื่องใช้ไฟฟ้าผ่านทางโทรศัพท์เป็นจำนวน 100 ครั้งแล้ว บันทึกความถูกต้องของการพูดสั่งงานและผลที่ได้จากตัวโปรแกรมควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์

ผลการทดลอง

เสียงที่ทดสอบ	จำนวนครั้งที่ทำการทดสอบ	จำนวนครั้งที่ถูกต้อง	เปอร์เซ็นต์ (%) ความถูกต้อง	เปอร์เซ็นต์ความถูกต้องเฉลี่ย
1	100	72	72 %	50 %
2	100	37	37 %	
3	100	56	56 %	
4	100	31	31 %	

ตารางที่ 4.1 ตารางแสดงเปอร์เซ็นต์ความถูกต้องของการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์

ผลจากการทดลองที่ 4.3 จะนำมาพล็อตเป็นกราฟได้ดังรูปที่ 4.25



รูปที่ 4.25 กราฟแสดงเปอร์เซ็นต์ความถูกต้องของการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพุดผ่านเครื่องโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และบทสรุป

5.1 บทวิจารณ์

จากการทำงานพบปัญหาและข้อจำกัดในการทำงานมากมาย จะอธิบายและแยกเป็นหัวข้อได้ดังนี้

5.1.1 ส่วนของโปรแกรมรู้จำเสียงพูด

จากผลการทดลองพบว่าสัญญาณเสียงที่แตกต่างกัน (ในปริภูมิตวินนัมที่คือเสียงหนึ่งถึงสี่) นั้นจะให้ผลของจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงที่แตกต่างกัน นอกจากนั้นค่าสัมประสิทธิ์การประมาณเชิงเส้น, สัมประสิทธิ์เซปสตรัม, และสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก ก็มีค่าแตกต่างกันไปในแต่ละเสียงด้วย

และถึงแม้ว่าจะเป็นเสียงเดียวกันแต่บุคคลคนละครั้งก็ได้ค่าจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียง, สัมประสิทธิ์การประมาณเชิงเส้น, สัมประสิทธิ์เซปสตรัม, และสัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก ที่ไม่เหมือนกันด้วย

นอกจากนี้วิธีการหาจุดสิ้นสุดของเสียงทั้งสองวิธี คือ วิธีหาค่าพลังงานและวิธีหาอัตราการตัดศูนย์ก็ให้ผลของจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงที่แตกต่างกันด้วย

ในส่วนของการลดความถูกต้องของการใช้งานเครื่องใช้ไฟฟ้านั้น พบว่าค่อนข้างต่ำ สาเหตุน่าจะมาจากตัวโปรแกรมในการรู้จำเสียงพูดเอง เนื่องจากว่าค่าของ

1. ลำดับของแอลพีซี (Order of Lpc)
2. ลำดับของเซปสตรัม (Order of Cepstrum)
3. จำนวนของการนอร์มอลไลซ์ (Number of Normalize)

นั้นต่างก็มีผลต่อความถูกต้องของการรู้จำเสียงพูดทั้งสิ้น ดังนั้นต้องทำการทดลองและเลือกค่าทั้ง 3 นี้ให้เหมาะสม ก็จะมีส่วนให้การรู้จำเสียงมีอัตราความถูกต้องสูงขึ้นด้วย

ส่วนที่สองที่มีผลต่อความถูกต้องของการรู้จำเสียงพูดก็คือ จำนวนของเสียงที่เก็บเป็นตัวอย่างในการรู้จำเสียงพูด จากโปรแกรมที่ใช้ในปริภูมิตวินนัมมีเสียงตัวอย่างเพียงค่าละ 20 เสียงซึ่งอาจจะให้ความถูกต้องได้ไม่สูงนัก ดังนั้นถ้าเพิ่มจำนวนของเสียงตัวอย่างอาจจะทำให้ความถูกต้องเพิ่มขึ้นได้

อีกส่วนที่มีผลต่อความถูกต้องในการใช้งานเครื่องใช้ไฟฟ้าก็คือ ตัวของโมเด็มที่ใช้ เนื่องจากว่าในการที่จะบันทึกเสียงเป็นไฟล์เวฟ (wave file) ใต้นั้น ต้องกระทำการผ่านสายไลน์อินของโมเด็ม ซึ่งถ้าคุณสมบัติในการอ่านเสียงของโมเด็มไม่ดีพอ ก็จะทำให้เสียงที่ได้มีความพร่าเพี้ยนสูงซึ่งก็จะส่งผลถึงการรู้จำเสียงพูดด้วยเช่นกัน

5.1.2 ส่วนของโมเด็ม

เนื่องจากว่าเสียงที่ใช้ควบคุมโทรศัพท์ที่เข้ามานั้นต้องได้รับการบันทึกจากสองอุปกรณ์ตามลำดับ คือ โมเด็มก่อน แล้วตามด้วยการ์ดเสียง ดังนั้นในการพูดสั่งงานนั้นถ้าพูดนานเกินไปหรือวางหูเข้าไป จะทำให้การเข้ามาเก็บเสียงของทั้งสองอุปกรณ์เกิดการชนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงต้องระวังในการสั่งงานให้เสร็จภายใน 5-8 วินาที

5.1.3 ส่วนของการส่งข้อมูลออกพอร์ตขนาน

ในส่วนของการส่งข้อมูลออกพอร์ตขนานที่เขียนด้วยภาษาบอร์แลนดซ์นั้น เราต้องสร้างเป็นฟังก์ชัน cportopen, cportconfig, และ cportwrite ขึ้นมาแทนที่จะใช้คำสั่ง output หรือ outputb ก็เนื่องจากว่า ภาษาบอร์แลนดซ์นั้นไม่สามารถจะใช้คำสั่ง output หรือ outputb ได้เหมือนกับภาษาซีอื่นๆที่รันบนคอส (DOS)

บอร์แลนดซ์ที่รันบนวินโดวส์และมีแพลตฟอร์ม (Platform) เป็นแบบ 32 บิต (bit) ในขณะที่ภาษาซีที่รันบนคอสมีแพลตฟอร์มแบบ 16 บิต ซึ่งสามารถเข้าถึงพอร์ตแอปพลิเคชัน (Port Application) ได้นั่นเอง

5.2 บทสรุป

แนวทางในการพัฒนาการควบคุมเครื่องใช้ไฟฟ้าด้วยเสียงพูดผ่านเครื่องโทรศัพท์ได้ โดยจะเน้นไปที่การเลือกค่า ลำดับของแอลพีซี, ลำดับของเซปสตรีม, จำนวนของการนอร์มอลไลซ์ ที่เหมาะสม และทำการเก็บตัวอย่างของเสียงที่จะนำไปใช้เทียบที่มากขึ้น

และในส่วนของวงจรที่แสดงการคิด-คียบของแอลอีดีนั้น สามารถนำผลการคิด-คียบนั้นไปประยุกต์ใช้กับเครื่องใช้ไฟฟ้าจริงๆได้ ซึ่งไม่ใช่เรื่องยากที่จะทำการพัฒนาต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. โปรแกรมส่งข้อมูลออกพอร์ตขนาน

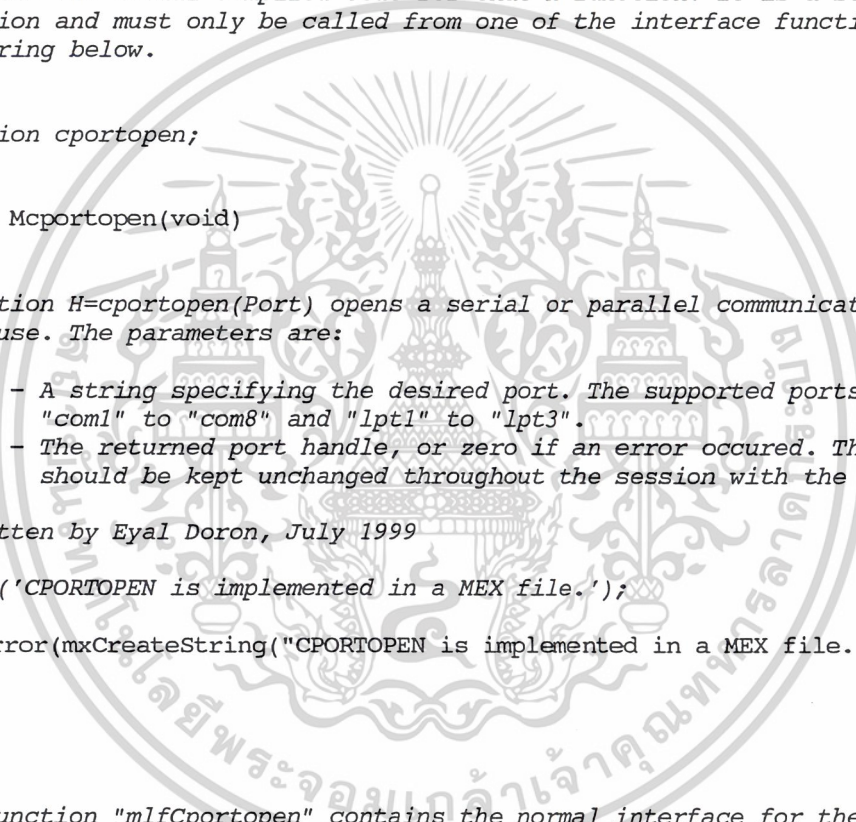
ฟังก์ชัน `cportopen` : เปิดพอร์ตที่ต้องการ

```
/*
 * MATLAB Compiler: 2.0
 * Date: Fri Mar 24 19:35:38 2000
 * Arguments: "-m" "NanDll"
 */

#include "cportopen.h"

/*
 * The function "Mcpportopen" is the implementation version of the "cportopen"
 * M-function from file "c:\matlabr11\cport\cportopen.m" (lines 1-14). It
 * contains the actual compiled code for that M-function. It is a static
 * function and must only be called from one of the interface functions,
 * appearing below.
 */
/*
 * function cportopen;
 */

static void Mcpportopen(void)
{
/*
 * %function H=cportopen(Port) opens a serial or parallel communication port
 * %for use. The parameters are:
 * %
 * %Port - A string specifying the desired port. The supported ports are
 * % "com1" to "com8" and "lpt1" to "lpt3".
 * %H - The returned port handle, or zero if an error occurred. This variable
 * % should be kept unchanged throughout the session with the port.
 *
 * % Written by Eyal Doron, July 1999
 *
 * error('CPOROPEN is implemented in a MEX file.');
```



```
*/
    mlfError(mxCreateString("CPOROPEN is implemented in a MEX file.));
/*
 *
 */
}
/*
 * The function "mlfCportopen" contains the normal interface for the
 * "cportopen" M-function from file "c:\matlabr11\cport\cportopen.m" (lines
 * 1-14). This function processes any input arguments and passes them to the
 * implementation version of the function, appearing above.
 */

void mlfCportopen(void)
{
    mlfEnterNewContext(0, 0);
    Mcpportopen();
    mlfRestorePreviousContext(0, 0);
}

/*
 * The function "mlxCportopen" contains the feval interface for the "cportopen"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

* M-function from file "c:\matlabr11\cport\cportopen.m" (lines 1-14). The
* feval function calls the implementation version of cportopen through this
* function. This function processes any input arguments and passes them to the
* implementation version of the function, appearing above.
*/
void mlxCportopen(int nlhs, mxArray * plhs[], int nrhs, mxArray * prhs[])
{
    if (nlhs > 0)
    {
        mlfError(
            mxCreateString(
                "Run-time Error: File: cportopen Line: 1 Column:"
                " 0 The function \"cportopen\" was called with m"
                "ore than the declared number of outputs (0)"));
    }
    if (nrhs > 0)
    {
        mlfError(
            mxCreateString(
                "Run-time Error: File: cportopen Line: 1 Column:"
                " 0 The function \"cportopen\" was called with m"
                "ore than the declared number of inputs (0)"));
    }
    mlfEnterNewContext(0, 0);
    Mcportopen();
    mlfRestorePreviousContext(0, 0);
}

```

ฟังก์ชัน **cportconfig** : กำหนดค่าพารามิเตอร์ต่างๆของพอร์ตที่เปิดไว้

```

/*
* MATLAB Compiler: 2.0
* Date: Fri Mar 24 19:35:38 2000
* Arguments: "-m" "NanDll"
*/
#include "cportconfig.h"

/*
* The function "Mcportconfig" is the implementation version of the
* "cportconfig" M-function from file "c:\matlabr11\cport\cportconfig.m" (lines
* 1-121). It contains the actual compiled code for that M-function. It is a
* static function and must only be called from one of the interface functions,
* appearing below.
*/
/*
* function cportconfig;
*/
static void Mcportconfig(void)
{
    /*
    * %function stat=cportconfig(H,Name,Value,Name,Value,Name,Value...)
    * %configures a serial or parallel port previously opened by cportopen,
    * %and/or returns the current or modified configuration. The parameters are:
    * %
    * %Input:
    * %-----
    * %H      - The port handle returned by cportopen.
    * %Name   - A character string containing a configuration parameter name. It
    */
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

* %      is enough to specify just enough characters to make a unique
* %      identification, e.g. "Ba" instead of "BaudRate".
* %Value - A scalar or character string containing the new parameter value.
* %
* %Output:
* %-----
* %stat - Optional. If present, contains the port configuration in a record
* %      whose fields are the parameter names and whose values are the
* %      configuration values. The parameter names and values are the same
* %      as for the input parameters.
* %
* %* zero or more input parameter pairs may be specified.
* %* 1/ON/YES/TRUE are synonymous, as are 0/OFF/NO/FALSE.
* %* All strings may be replaced by their equivalent numbers, as defined
* %* in Windows.h.
* %* The commands are case insensitive.
* %
* %Supported parameters and their allowed values are given below
* %
* %Port Configuration:
* %-----
* %      BaudRate:      Specifies the device communication rate in bits per
* %                     second.
* %      ByteSize:      Specifies the bits per byte transmitted and received,
* %                     can be 4-8.
* %      StopBits:      Specifies the number of stop bits. Allowed values are
* %                     1, 1.5 and 2. The common value is 1.
* %      Parity:         Specifies the parity scheme: NONE,ODD,EVEN,MARK,SPACE (0-
* %                     4).
* %                     Do not confuse this member with the fParity member, which
* %                     turns parity on or off. Because parity is rarely used,
* %                     this member is usually NONE.
* %      fParity:        ON/OFF toggles parity checking.
* %      fOutX, fInX:    ON/OFF toggles software flow control (XON/XOFF) using the
* %                     XonChar/XoffChar characters.
* %      fRtsControl:    ON/OFF toggles the RTS line on and off during the
* %                     connection. HANDSHAKE turns on RTS handshaking. TOGGLE
* %                     sets the RTS line high if bytes are available for
* %                     transmission, later putting it to low.
* %      fOutxCtsFlow:   ON/OFF Turn the CTS flow control on and off. To use
* %                     RTS/CTS flow control, specify ON for this member and
* %                     HANDSHAKE for
* %                     the fRtsControl member.
* %      fOutxDsrFlow:   ON/OFF toggles the DSR flow control. DSR flow control is
* %                     rarely used. A typical port configuration is to set this
* %                     member to OFF, while enabling the DTR line.
* %      fDtrControl:    ON/OFF turns the DTR line on and off during the
* %                     connection. HANDSHAKE turns on DTR handshaking.
* %      fDsrSensitivity: ON/OFF specifies whether the comm driver is sensitive
* %                     to the state of the DSR line. If ON, the driver ignores
* %                     any bytes received while the DSR line is low.
* %      fTXContinueOnXoff: ON/OFF Specifies whether transmission stops when the
* %                     input buffer is full and the driver has transmitted
* %                     the XOFF character. .
* %      fErrorChar:     ON/OFF specifies whether bytes received with parity errors
* %                     are replaced by the ErrorChar character. This will be
* %                     operative only if fParity is also ON.
* %      fNull:          ON/OFF specifies whether null bytes are discarded.
* %      fAbortOnError:  ON/OFF specifies whether read and write operations are
* %                     terminated if an error occurs. The driver will then not

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

* %          accept further communication until it is cleared using
* %          CPORTRESET.
* % XonLim:    The minimum number of bytes allowed in the input buffer
* %           before flow control is activated to inhibit the sender.
* %           This assume that either Xon/Xoff, RTS, or DTR input flow
* %           control is specified in fInX, fRtsControl or fDtrControl.
* %           Note that the sender may transmit characters after the
* %           flow control signal has been activated, so this value
* %           should never be zero.
* % XoffLim:   The maximum number of bytes allowed in the input buffer
* %           before flow control is activated to allow transmission by
* %           the sender. This assumes that either Xon/Xoff, RTS, or DTR
* %           input flow control is specified in fInX, fRtsControl or
* %           fDtrControl. The maximum number of bytes allowed is
* %           calculated by subtracting this value from the size, in
* %           bytes, of the input buffer.
* % XonChar,XoffChar: The ASCII value of the characters used for Xon/Xoff
* %                   software flow control.
* % ErrorChar:  The ASCII value of the character used to replaced bytes
* %             received with a parity error.
* % EofChar:    The ASCII value of the character used to signal the end of
* %             data.
* % EvtChar:    The ASCII value of the character used to signal an event.
* %
* %Port timeouts:
* %-----
* % ReadIntervalTimeout:
* %           The maximum number of milliseconds that can elapse between
* %           two characters without a time-out occurring with the
* %           ReadIntervalTimeout member.
* % ReadTotalTimeoutMultiplier:
* %           For each read operation, this number is multiplied by the
* %           number of bytes that the read operation expects to
* %           receive.
* % ReadTotalTimeoutConstant:
* %           This member is the number of milliseconds added to the
* %           result of multiplying the total number of bytes to read by
* %           ReadTotalTimeoutMultiplier. The result is the number of
* %           milliseconds that must elapse before a time-out for the
* %           read operation occurs.
* % WriteTotalTimeoutMultiplier:
* %           For each write operation, this number is multiplied by the
* %           number of bytes that the write operation expects to
* %           receive.
* % WriteTotalTimeoutConstant:
* %           This member is the number of milliseconds added to the
* %           result of multiplying the total number of bytes to write
* %           by WriteTotalTimeoutMultiplier. The result is the number
* %           of milliseconds that must elapse before a time-out for the
* %           write operation occurs.
* %
* %To assist with multitasking, it is common practice to configure so that
* %read
* %operations immediately return with the characters to be read. To do this,
* %set ReadIntervalTimeout to -1 and set both ReadTotalTimeoutMultiplier and
* %ReadTotalTimeoutMultiplier to zero.
*
* % Written by Eyal Doron, July 1999
*
* error('CPORTCONFIG is implemented in a MEX file.');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    */
    mlfError(mxCreateString("CPORTCONFIG is implemented in a MEX file.));
    /*
    *
    */
    }

    /*
    * The function "mlfCportconfig" contains the normal interface for the
    * "cportconfig" M-function from file "c:\matlabr11\cport\cportconfig.m" (lines
    * 1-121). This function processes any input arguments and passes them to the
    * implementation version of the function, appearing above.
    */
    void mlfCportconfig(void)
    {
        mlfEnterNewContext(0, 0);
        Mcportconfig();
        mlfRestorePreviousContext(0, 0);
    }

    /*
    * The function "mlxCportconfig" contains the feval interface for the
    * "cportconfig" M-function from file "c:\matlabr11\cport\cportconfig.m" (lines
    * 1-121). The feval function calls the implementation version of cportconfig
    * through this function. This function processes any input arguments and
    * passes them to the implementation version of the function, appearing above.
    */
    void mlxCportconfig(int nlhs, mxArray * plhs[], int nrhs, mxArray * prhs[]) {
        if (nlhs > 0)
        {
            mlfError(
            mxCreateString(
            "Run-time Error: File: cportconfig Line: 1 Column"
            ": 0 The function \"cportconfig\" was called with"
            " more than the declared number of outputs (0)");
        }
        if (nrhs > 0)
        {
            mlfError(
            mxCreateString(
            "Run-time Error: File: cportconfig Line: 1 Column"
            ": 0 The function \"cportconfig\" was called with"
            " more than the declared number of inputs (0)");
        }
        mlfEnterNewContext(0, 0);
        Mcportconfig();
        mlfRestorePreviousContext(0, 0);
    }
}

```

ฟังก์ชัน cportwrite : ส่งค่าที่กำหนดออกพอร์ตที่เปิดไว้

```

/*
*
* CPORTWrite.C
*
* The calling syntax is:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

*
* nWritten=cportWrite(Port, Data, EOL)
*
*/

#include <mex.h>
#include <windows.h>

/* Input Arguments */

#define PORT_IN prhs[0]
#define DATA_IN prhs[1]
#define EOL_IN prhs[2]

/* Output Arguments */

#define NW plhs[0]

BOOL CPortWrite(
    HANDLE *hp,
    void *Data,
    DWORD nWrite,
    DWORD *nWritten,
    char *EOL
)
{
    char EOLStr[3];
    DWORD nWEOL;

    if (!WriteFile(*hp, Data, nWrite, nWritten, NULL)) return FALSE;
    /* mexPrintf("%s",Data); */
    if (EOL != NULL)
    {
        {
            EOLStr[0] = '\\0'; EOLStr[1] = '\\0'; EOLStr[2] = '\\0';
            if ((strcmp(EOL,"cr")==0) || (strcmp(EOL,"CR")==0 )) EOLStr[0] = 13;
            if ((strcmp(EOL,"lf")==0) || (strcmp(EOL,"LF")==0 )) EOLStr[0] = 10;
            if ((strcmp(EOL,"crlf")==0) || (strcmp(EOL,"CRLF")==0))
            {
                EOLStr[0] = 13; EOLStr[1] = 10;
            }
        }
        nWrite = strlen(EOLStr);
        if (!WriteFile(*hp, EOLStr, nWrite, &nWEOL, NULL)) return FALSE;
        *nWritten += nWEOL;
        /* mexPrintf("%s",EOLStr); */
    }
    return TRUE;
}

void mexFunction(int nlhs, mxArray *plhs[], int nrhs, const mxArray *prhs[])

{
    void *Data;
    HANDLE *hp;
    DWORD nWritten, nWrite;
    double *dnWritten, nW;
    char *EOL;

    /* Check for proper number of arguments */
    if (nrhs < 2)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        mexErrMsgTxt("CPORTWRITE requires at least two input arguments.");
    }
    if (nrhs > 3)
    {
        mexErrMsgTxt("CPORTWRITE requires at most three input arguments.");
    }
    if (nlhs > 1)
    {
        mexErrMsgTxt("CPORTWRITE requires one output argument at most.");
    }

    /* Check the type of PORT_IN - should be double */
    if (!mxIsDouble(PORT_IN))
    {
        mexErrMsgTxt("CPORTWRITE first argument should be a port handle.");
    }

    /* Allocate and translate the input data string */
    if (mxIsChar(DATA_IN))
    {
        Data = (void *)mxArrayToString(DATA_IN);
        nWrite = mxGetNumberOfElements(DATA_IN);
    }
    else
    {
        Data = mxGetData(DATA_IN);
        nWrite = mxGetNumberOfElements(DATA_IN) * mxGetElementSize(DATA_IN);
    }
    nWritten = 0;

    /* EOL */
    EOL=NULL;
    if (nrhs==3)
    {
        if (!mxIsChar(EOL_IN))
            mexErrMsgTxt("CPORTWRITE third argument should be a string.");
        EOL=mxArrayToString(EOL_IN);
    }

    /* Assign pointers to the various parameters */
    hp = (HANDLE *)mxGetData(PORT_IN);

    /* Do the actual computations in a subroutine */
    if (CPortWrite(hp, Data, nWrite, &nWritten, EOL))
        nW = nWritten;
    else
        nW = -1;
    if (nlhs > 0)
    {
        NW = mxCreateDoubleMatrix(1, 1, mxREAL);
        dnWritten = (double *)mxGetPr(NW);
        *dnWritten = nW;
    }
    return;
}

```

ฟังก์ชันทั้งหมดได้มาจาก <ftp://ftp.mathworks.com/pub/contrib/v5/tools/cport/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. โปรแกรมรู้จำเสียงพูด

ฟังก์ชัน detect1 : ฟังก์ชันหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงด้วยการเปรียบเทียบ

ค่าพลังงาน

```
function[E,a1,b1,ref,s] = detect1(filename);
% [E,a1,b1,ref,s]=detect1(filename);
% This function for find endpoint detection by ENERGY
% when filename : file.wav
% E : energy each frame
% s : signal when pass detection by energy

[inpt1,fs] = wavread(filename);
inpt = inpt1-mean(inpt1);
% number of frame
sample = 100;
vnoloop = floor(length(inpt)/sample);
% find energy each of frame
for k = 1:vnoloop
    sum = 0;
    for k1 = 1:sample
        sum = sum + abs (inpt((k-1)*sample+k1));
    end
    E(k) = sum;
end
% find maximum energy
Emax = max(E);
% set value
a = 1/5;
% begin process
ref = a*Emax;
for k = 1:vnoloop
    if E(k)==Emax
        fixenergy = k;
        break;
    end
end
for k = fixenergy:-1:1
    if E(k)<=ref
        a1 = k;
        break;
    end
end
for k1 = fixenergy:vnoloop
    if E(k1)<=ref
        b1 = k1;
        break;
    end
end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

num = 0;
if isempty(b1)==1|b1==vnoloop
    b1 = vnoloop-1;
    for i = (a1-1):b1
        for i1 = 1:sample
            num = num+1;
            s(num) = inpt(i*sample+i1);
        end
    end
elseif (b1~=vnoloop-1)
    for i = (a1-1):(b1+1)
        for i1 = 1:sample
            num = num+1;
            s(num) = inpt(i*sample+i1);
        end
    end
else b1==vnoloop-1;
    for i = (a1-1):b1
        for i1 = 1:sample
            num = num+1;
            s(num) = inpt(i*sample+i1);
        end
    end
end
end

```

ฟังก์ชัน zrocrs1 : ฟังก์ชันหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงโดยการเปรียบเทียบอัตราการตัดศูนย์

```

function [S,z,a2,b2,zmax] = zrocrs1(filename,a1,b1);
% [S,z,a2,b2,Zmax] = zrocrs1(filename,a1,b1)
% when filename : file.wav
% a1,b1 : initial and finish point at detect by energy
% S : signal when pass detection by zero_crossing

[inpt1,fs] = wavread(filename);
inpt = inpt1-mean(inpt1);
% find number of frame
sample = 100;
vnoloop1 = floor(length(inpt)/sample);
% find zero_crossing each of frame
for k = 1:vnoloop1
    sum = 0;
    for k1 = 1:sample
        if k1==1
            sum = sum+abs(sg_n(inpt((k-1)*sample+k1)));
        else
            sum = sum+abs(sg_n(inpt((k-1)*sample+k1))-...
                sg_n(inpt((k-1)*sample+(k1-1))));
        end
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        end
    end
    z(k) = sum/(2*sample);
end
% find maximum zero_crossing
zmax = max(z);
a = 1/5;
% find position of maximum zero_crossing
for k = 1:vnoloop1
    if z(k)==zmax
        fixzero = k;
        break;
    end
end
refzro = a*zmax;
% process find bound signal
for k1 = a1:-1:1
    if z(k1)<=refzro
        a2 = k1;
        break;
    end
end
for k2 = b1:vnoloop1
    if z(k2) <= refzro
        b2 = k2;
        break;
    end
end
% plot signal
n = 0;
if isempty(b2)==1|b2==vnoloop1|isempty(a2)==1;
    if isempty(a2)==1;
        a2 = a1-1;
        b2 = vnoloop1-1;
        for k3 = a2:b2
            for k4 = 1:sample
                n = n+1;
                S(n) = inpt((k3*sample)+k4);
            end
        end
    end
else
    b2 = vnoloop1-1;
    for k3 = a2:b2
        for k4 = 1:sample
            n = n+1;
            S(n) = inpt((k3*sample)+k4);
        end
    end
end
else
    for k5 = a2:b2
        for k6 = 1:sample

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        n = n+1;
        S(n) = inpt((k5*sample)+k6);
    end
end
end

```

```

function zerc = sg_n(s);
% This function for sign
% zerc = sg_n(s);
% s : is signal when pass detection by energy

for i = 1:length(s)
    if s(i)>0
        zerc = 1;
    elseif s(i)==0
        zerc = 0;
    else
        zerc = -1;
    end
end
end

```

ฟังก์ชัน normalize : ฟังก์ชันนอร์มอลไลซ์สัญญาณเสียงให้ได้จำนวนตัวอย่าง 4000 ตัว

```

function normS = normalize(S,normsize);
% normS = normalize(S,normsize);
%
%***** input *****
% normsize = 4000;
% when S = signal when zerocrossing;
% normsize = number of normalize;
%***** output *****
% normS = signal when pass normalize;
%*****

normS(1) = S(1);
timestep = (length(S))/(normsize);
time = 1;
j = 2;
while j<normsize
    time = time+timestep;
    lowtime = floor(time);
    residue = time-lowtime;
    if (lowtime+1>length(S))
        S(lowtime+1) = 0;
    end
    normS(j) = S(lowtime)+residue*(S(lowtime+1)-S(lowtime));
    j = j+1;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
normS(j) = S(length(S));
```

ฟังก์ชัน `levin_durbin` : ฟังก์ชันหาค่าสัมประสิทธิ์การประมาณเชิงเส้น, สัมประสิทธิ์เซป
สตรัม, สัมประสิทธิ์เซปสตรัมถ่วงน้ำหนัก และค่าเกณฑ์ของการ
ประมาณเชิงเส้น

```
function [b,c,w,G] = levin_durbin(S,N,Q);  
% [b,c,w,G] = levin_durbin(S,N,Q);  
  
%***** input *****  
% when S = speech waveform  
% N = order of lpc11  
% Q = order of cepstrum  
%***** output *****  
% when b = lpc11 parameter  
% c = cepstrum parameter  
% w = weight cepstrum parameter  
% G = gain  
%*****  
  
fs = 8000;  
sizf = length(S);  
% initial pre emphasis *****  
format long;  
a1 = 0.9375;  
as = 0;  
df = 0;  
gh = 0;  
for i = 1:sizf  
    if i==1  
        s(i) = S(i);  
    else  
        s(i) = S(i)-a1*S(i-1);  
    end  
end  
  
% end pre-emphasis *****  
  
% initial divide frame *****  
  
fram = (240/fs);%0.03  
t = (160/fs); %0.02  
shift = (80/fs);%0.01  
z = length(s)/160;  
le = length(s)-((floor(z)-1)*160);  
if le==fram*fs % (fram*fs = 240)  
    frame = floor(z);  
else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    frame = floor(z)-1;
end

% end divide frame *****

% initial windowing *****

for l = 1:frame
    v = 0;
    for n = (((l-1)*160)+1):(((l-1)*160)+240)
        v = v+1;
        as = as+1;
        x(l,v) = s(n)*(0.54-(0.46*cos(2*pi*v/(fram*fs))));
        h(as) = x(l,v);
    end
    % end windowing *****
    % initial auto correlation & lpc11 analysis
    [a,g,R] = lpc11(h,N); %N : order of lpc11
    for j = 1:N+1
        df = df+1;
        b(df) = a(j);
        r(l,1) = R(1);
        G(l) = g;
    end
    [ce,cw] = ceps(a,g,N,Q);
    for il = 1:Q
        gh = gh+1;
        c(gh) = ce(il);
        w(gh) = cw(il);
    end
end

% end auto correlation & lpc11 analysis
% when final frame not 240 *****
% initial auto correlation & lpc11 analysis
if le<240
    v = 0;
    for n = (frame*160)+1:(frame*160)+le
        v = v+1;
        as = as+1;
        x(frame+1,v) = s(n)*(0.54-(0.46*cos(2*pi*v/le)));
        h(as) = x(frame+1,v);
    end
    frame = frame+1;
    l = l+1;

    % end windowing *****

    % initial auto correlation & lpc11 analysis
    [a,g,R] = lpc11(h,N);
    for j = 1:N+1
        df = df+1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        b(df) = a(j);
        r(1,1) = R(1);
        G(1) = g;
    end
    [ce,cw] = ceps(a,g,N,Q);
    for i = 1:Q
        gh = gh+1;
        c(gh) = ce(i);
        w(gh) = cw(i);
    end
end

function [ce,cw] = ceps(a,G,N,Q);
ce(1) = 0.5*log(G);
ce(2) = a(2);
for i = N+2:Q;
    a(i) = 0;
end
for m = 3:Q
    sum = 0;
    for k = 1:m-1
        sum = sum+(k*ce(k)*a(m-k));
    end
    cep_sum(m) = (1/m)*sum;
    if m<=N+1
        ce(m) = a(m)+cep_sum(m);
    else
        ce(m) = cep_sum(m);
    end
end
cw = cepsw(Q,ce);

% ***** weight cepstrum function *****

function cw = cepsw(Q,ce);
for m = 1:Q
    cw(m) = ce(m)*(1+((Q/2)*sin(pi*(m/Q))));
end

function [a,g,R] = lpc11(h,N);
error(nargchk(1,2,nargin))
h = h;
[r,c] = size(h);
if (c>1)&(r==1)
    h = h(:);
end
numsig = size(h,2);
if nargin<2 ,
    N = size(h,1)-1;
end
if (N>size(h,1)-1),

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

h(N+1,:) = zeros(1,numsigs);
end
R = flipud(fftshift(h,conj(flipud(h))))/length(h);
% R is the autocorrelation vector
% equivalent code (for the single signal case);
% R = xcorr(h,'biased');
% M = length(h);
% R(1:M-1) = [];

a = levinson(R,N);
g = real(sqrt(sum((a.').*R(1:N+1,:)))));

```

ฟังก์ชัน distance1 และ grad1 : ฟังก์ชันหาค่าระยะทางระหว่างรูปแบบอ้างอิงกับรูปแบบ

ทดสอบ

```

function [d,A,B]= distance1(s,s1);
% [d,A,B] = distance1(s,s1);
% when s : signal for test recognition
% S1 : reference coefficient when pass weight cepstrum

format long;
A = length(s);
B = length(s1);

for i = 1:A
    top = ceil((i+1)/2)+(B-1)-(A-2)/2;
    bottom = ceil((2*i)+(B-2)-2*(A-1))-2;
    ifbottom<=0
        for j = 1:top
            d(i,j) = abs(s(i)-s1(j));
        end
    else
        if top<=max(A,B)
            for j = bottom:top
                d(i,j) = abs(s(i)-s1(j));
            end
        else
            for j = bottom:max(A,B)
                d(i,j) = abs(s(i)-s1(j));
            end
        end
    end
end
end

function [D,g] = grad1(d,A,B)
% [D,g] = grad1(d,A,B);
format long;
for i = 1:A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

top = round(((i+1)/2)+(B-1)-(A-2)/2);
bottom = round((2*i)+(B-2)-2*(A-1))-2
ifbottom<=0
    bottom = 1;
end
iftop>max(A,B);
    top = max(A,B);
end
for j = bottom:top
    if i==1|j==1
        g(i,j) = 2*d(i,j);
    elseif (i==2)&(j==2)
        g(i,j) = g(i-1,j-1)+2*d(i,j);
    elseif (j>2)&(i==2)
        g(i,j) = min(g(i-1,j-1)+2*d(i,j),...
            g(i-1,j-2)+2*d(i,j-1)+d(i,j));
    elseif (i>2)&(j==2)
        g(i,j) = min(g(i-1,j-1)+2*d(i,j),...
            g(i-2,j-1)+2*d(i-1,j)+d(i,j));
    elseif (j>2)&(i>2)
        gradian = [g(i-2,j-1)+2*d(i-1,j)+...
            d(i,j),g(i-1,j-1)+2*d(i,j),...
            g(i-1,j-2)+2*d(i,j-1)+d(i,j)];
        g(i,j) = min(gradian);
    end
end
end
D = g(A,B)/(A+B);

```

ฟังก์ชัน FindRef, FindWav : ฟังก์ชันหาค่าสัมประสิทธิ์ที่ใช้ในระบบการรู้จำเสียงพูด

```

function [b,c,w,G]=FindRef(filename);
N = 12; Q = 12;
[E,a1,b1,ref,s] = detect1(filename);
[S,z,a2,b2,zmax] = zrocrs1(filename,a1,b1);
normS = normalize(S,4000);
[b,c,w,G] = levin_durbin(normS,12,12);

function [b,c,w,G,S,s]=FindWav(filename);
N = 12; Q = 12;
[E,a1,b1,ref,s] = detect1(filename);
[S,z,a2,b2,zmax] = zrocrs1(filename,a1,b1);
normS = normalize(S,4000);
[b,c,w,G] = levin_durbin(normS,12,12);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน SaveRef : ฟังก์ชันเก็บฐานข้อมูลของเสียงพูด

```
function SaveRef_All;
```

```
[b101,c101,w101,G101]=FindRef('c:\users\voice\n1_01.wav');  
fprintf('***** Pass 1.01 *****\n');  
[b102,c102,w102,G102]=FindRef('c:\users\voice\n1_02.wav');  
fprintf('***** Pass 1.02 *****\n');  
[b103,c103,w103,G103]=FindRef('c:\users\voice\n1_03.wav');  
fprintf('***** Pass 1.03 *****\n');  
[b104,c104,w104,G104]=FindRef('c:\users\voice\n1_04.wav');  
fprintf('***** Pass 1.04 *****\n');  
[b105,c105,w105,G105]=FindRef('c:\users\voice\n1_05.wav');  
fprintf('***** Pass 1.05 *****\n');  
[b106,c106,w106,G106]=FindRef('c:\users\voice\n1_06.wav');  
fprintf('***** Pass 1.06 *****\n');  
[b107,c107,w107,G107]=FindRef('c:\users\voice\n1_07.wav');  
fprintf('***** Pass 1.07 *****\n');  
[b108,c108,w108,G108]=FindRef('c:\users\voice\n1_08.wav');  
fprintf('***** Pass 1.08 *****\n');  
[b109,c109,w109,G109]=FindRef('c:\users\voice\n1_09.wav');  
fprintf('***** Pass 1.09 *****\n');  
[b110,c110,w110,G110]=FindRef('c:\users\voice\n1_10.wav');  
fprintf('***** Pass 1.10 *****\n');  
[b111,c111,w111,G111]=FindRef('c:\users\voice\n1_11.wav');  
fprintf('***** Pass 1.11 *****\n');  
[b112,c112,w112,G112]=FindRef('c:\users\voice\n1_12.wav');  
fprintf('***** Pass 1.12 *****\n');  
[b113,c113,w113,G113]=FindRef('c:\users\voice\n1_13.wav');  
fprintf('***** Pass 1.13 *****\n');  
[b114,c114,w114,G114]=FindRef('c:\users\voice\n1_14.wav');  
fprintf('***** Pass 1.14 *****\n');  
[b115,c115,w115,G115]=FindRef('c:\users\voice\n1_15.wav');  
fprintf('***** Pass 1.15 *****\n');  
[b116,c116,w116,G116]=FindRef('c:\users\voice\n1_16.wav');  
fprintf('***** Pass 1.16 *****\n');  
[b117,c117,w117,G117]=FindRef('c:\users\voice\n1_17.wav');  
fprintf('***** Pass 1.17 *****\n');  
[b118,c118,w118,G118]=FindRef('c:\users\voice\n1_18.wav');  
fprintf('***** Pass 1.18 *****\n');  
[b119,c119,w119,G119]=FindRef('c:\users\voice\n1_19.wav');  
fprintf('***** Pass 1.19 *****\n');  
[b120,c120,w120,G120]=FindRef('c:\users\voice\n1_20.wav');  
fprintf('***** Pass 1.20 *****\n');
```

```
save c:\users\result\b101 b101; save c:\users\result\b102 b102;  
save c:\users\result\b103 b103; save c:\users\result\b104 b104;  
save c:\users\result\b105 b105; save c:\users\result\b106 b106;  
save c:\users\result\b107 b107; save c:\users\result\b108 b108;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
[b207,c207,w207,G207]=FindRef('c:\users\voice\n2_07.wav');
fprintf('***** Pass 2.07 *****\n');
[b208,c208,w208,G208]=FindRef('c:\users\voice\n2_08.wav');
fprintf('***** Pass 2.08 *****\n');
[b209,c209,w209,G209]=FindRef('c:\users\voice\n2_09.wav');
fprintf('***** Pass 2.09 *****\n');
[b210,c210,w210,G210]=FindRef('c:\users\voice\n2_10.wav');
fprintf('***** Pass 2.10 *****\n');
[b211,c211,w211,G211]=FindRef('c:\users\voice\n2_11.wav');
fprintf('***** Pass 2.11 *****\n');
[b212,c212,w212,G212]=FindRef('c:\users\voice\n2_12.wav');
fprintf('***** Pass 2.12 *****\n');
[b213,c213,w213,G213]=FindRef('c:\users\voice\n2_13.wav');
fprintf('***** Pass 2.13 *****\n');
[b214,c214,w214,G214]=FindRef('c:\users\voice\n2_14.wav');
fprintf('***** Pass 2.14 *****\n');
[b215,c215,w215,G215]=FindRef('c:\users\voice\n2_15.wav');
fprintf('***** Pass 2.15 *****\n');
[b216,c216,w216,G216]=FindRef('c:\users\voice\n2_16.wav');
fprintf('***** Pass 2.16 *****\n');
[b217,c217,w217,G217]=FindRef('c:\users\voice\n2_17.wav');
fprintf('***** Pass 2.17 *****\n');
[b218,c218,w218,G218]=FindRef('c:\users\voice\n2_18.wav');
fprintf('***** Pass 2.18 *****\n');
[b219,c219,w219,G219]=FindRef('c:\users\voice\n2_19.wav');
fprintf('***** Pass 2.19 *****\n');
[b220,c220,w220,G220]=FindRef('c:\users\voice\n2_20.wav');
fprintf('***** Pass 2.20 *****\n');
```

```
save c:\users\result\b201 b201; save c:\users\result\b202 b202;
save c:\users\result\b203 b203; save c:\users\result\b204 b204;
save c:\users\result\b205 b205; save c:\users\result\b206 b206;
save c:\users\result\b207 b207; save c:\users\result\b208 b208;
save c:\users\result\b209 b209; save c:\users\result\b210 b210;
save c:\users\result\b211 b211; save c:\users\result\b212 b212;
save c:\users\result\b213 b213; save c:\users\result\b214 b214;
save c:\users\result\b215 b215; save c:\users\result\b216 b216;
save c:\users\result\b217 b217; save c:\users\result\b218 b218;
save c:\users\result\b219 b219; save c:\users\result\b220 b220;
```

```
save c:\users\result\c201 c201; save c:\users\result\c202 c202;
save c:\users\result\c203 c203; save c:\users\result\c204 c204;
save c:\users\result\c205 c205; save c:\users\result\c206 c206;
save c:\users\result\c207 c207; save c:\users\result\c208 c208;
save c:\users\result\c209 c209; save c:\users\result\c210 c210;
save c:\users\result\c211 c211; save c:\users\result\c212 c212;
save c:\users\result\c213 c213; save c:\users\result\c214 c214;
save c:\users\result\c215 c215; save c:\users\result\c216 c216;
save c:\users\result\c217 c217; save c:\users\result\c218 c218;
save c:\users\result\c219 c219; save c:\users\result\c220 c220;
```

```
save c:\users\result\w201 w201; save c:\users\result\w202 w202;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
save c:\users\result\w203 w203; save c:\users\result\w204 w204;
save c:\users\result\w205 w205; save c:\users\result\w206 w206;
save c:\users\result\w207 w207; save c:\users\result\w208 w208;
save c:\users\result\w209 w209; save c:\users\result\w210 w210;
save c:\users\result\w211 w211; save c:\users\result\w212 w212;
save c:\users\result\w213 w213; save c:\users\result\w214 w214;
save c:\users\result\w215 w215; save c:\users\result\w216 w216;
save c:\users\result\w217 w217; save c:\users\result\w218 w218;
save c:\users\result\w219 w219; save c:\users\result\w220 w220;
```

```
save c:\users\result\G201 G201; save c:\users\result\G202 G202;
save c:\users\result\G203 G203; save c:\users\result\G204 G204;
save c:\users\result\G205 G205; save c:\users\result\G206 G206;
save c:\users\result\G207 G207; save c:\users\result\G208 G208;
save c:\users\result\G209 G209; save c:\users\result\G210 G210;
save c:\users\result\G211 G211; save c:\users\result\G212 G212;
save c:\users\result\G213 G213; save c:\users\result\G214 G214;
save c:\users\result\G215 G215; save c:\users\result\G216 G216;
save c:\users\result\G217 G217; save c:\users\result\G218 G218;
save c:\users\result\G219 G219; save c:\users\result\G220 G220;
```

```
[b301,c301,w301,G301]=FindRef('c:\users\voice\n3_01.wav');
fprintf('***** Pass 3.01 *****\n');
[b302,c302,w302,G302]=FindRef('c:\users\voice\n3_02.wav');
fprintf('***** Pass 3.02 *****\n');
[b303,c303,w303,G303]=FindRef('c:\users\voice\n3_03.wav');
fprintf('***** Pass 3.03 *****\n');
[b304,c304,w304,G304]=FindRef('c:\users\voice\n3_04.wav');
fprintf('***** Pass 3.04 *****\n');
[b305,c305,w305,G305]=FindRef('c:\users\voice\n3_05.wav');
fprintf('***** Pass 3.05 *****\n');
[b306,c306,w306,G306]=FindRef('c:\users\voice\n3_06.wav');
fprintf('***** Pass 3.06 *****\n');
[b307,c307,w307,G307]=FindRef('c:\users\voice\n3_07.wav');
fprintf('***** Pass 3.07 *****\n');
[b308,c308,w308,G308]=FindRef('c:\users\voice\n3_08.wav');
fprintf('***** Pass 3.08 *****\n');
[b309,c309,w309,G309]=FindRef('c:\users\voice\n3_09.wav');
fprintf('***** Pass 3.09 *****\n');
[b310,c310,w310,G310]=FindRef('c:\users\voice\n3_10.wav');
fprintf('***** Pass 3.10 *****\n');
[b311,c311,w311,G311]=FindRef('c:\users\voice\n3_11.wav');
fprintf('***** Pass 3.11 *****\n');
[b312,c312,w312,G312]=FindRef('c:\users\voice\n3_12.wav');
fprintf('***** Pass 3.12 *****\n');
[b313,c313,w313,G313]=FindRef('c:\users\voice\n3_13.wav');
fprintf('***** Pass 3.13 *****\n');
[b314,c314,w314,G314]=FindRef('c:\users\voice\n3_14.wav');
fprintf('***** Pass 3.14 *****\n');
[b315,c315,w315,G315]=FindRef('c:\users\voice\n3_15.wav');
fprintf('***** Pass 3.15 *****\n');
[b316,c316,w316,G316]=FindRef('c:\users\voice\n3_16.wav');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
save c:\users\result\G319 G319; save c:\users\result\G320 G320;
```

```
[b401,c401,w401,G401]=FindRef('c:\users\voice\n4_01.wav');  
fprintf('***** Pass 4.01 *****\n');  
[b402,c402,w402,G402]=FindRef('c:\users\voice\n4_02.wav');  
fprintf('***** Pass 4.02 *****\n');  
[b403,c403,w403,G403]=FindRef('c:\users\voice\n4_03.wav');  
fprintf('***** Pass 4.03 *****\n');  
[b404,c404,w404,G404]=FindRef('c:\users\voice\n4_04.wav');  
fprintf('***** Pass 4.04 *****\n');  
[b405,c405,w405,G405]=FindRef('c:\users\voice\n4_05.wav');  
fprintf('***** Pass 4.05 *****\n');  
[b406,c406,w406,G406]=FindRef('c:\users\voice\n4_06.wav');  
fprintf('***** Pass 4.06 *****\n');  
[b407,c407,w407,G407]=FindRef('c:\users\voice\n4_07.wav');  
fprintf('***** Pass 4.07 *****\n');  
[b408,c408,w408,G408]=FindRef('c:\users\voice\n4_08.wav');  
fprintf('***** Pass 4.08 *****\n');  
[b409,c409,w409,G409]=FindRef('c:\users\voice\n4_09.wav');  
fprintf('***** Pass 4.09 *****\n');  
[b410,c410,w410,G410]=FindRef('c:\users\voice\n4_10.wav');  
fprintf('***** Pass 4.10 *****\n');  
[b411,c411,w411,G411]=FindRef('c:\users\voice\n4_11.wav');  
fprintf('***** Pass 4.11 *****\n');  
[b412,c412,w412,G412]=FindRef('c:\users\voice\n4_12.wav');  
fprintf('***** Pass 4.12 *****\n');  
[b413,c413,w413,G413]=FindRef('c:\users\voice\n4_13.wav');  
fprintf('***** Pass 4.13 *****\n');  
[b414,c414,w414,G414]=FindRef('c:\users\voice\n4_14.wav');  
fprintf('***** Pass 4.14 *****\n');  
[b415,c415,w415,G415]=FindRef('c:\users\voice\n4_15.wav');  
fprintf('***** Pass 4.15 *****\n');  
[b416,c416,w416,G416]=FindRef('c:\users\voice\n4_16.wav');  
fprintf('***** Pass 4.16 *****\n');  
[b417,c417,w417,G417]=FindRef('c:\users\voice\n4_17.wav');  
fprintf('***** Pass 4.17 *****\n');  
[b418,c418,w418,G418]=FindRef('c:\users\voice\n4_18.wav');  
fprintf('***** Pass 4.18 *****\n');  
[b419,c419,w419,G419]=FindRef('c:\users\voice\n4_19.wav');  
fprintf('***** Pass 4.19 *****\n');  
[b420,c420,w420,G420]=FindRef('c:\users\voice\n4_20.wav');  
fprintf('***** Pass 4.20 *****\n');
```

```
save c:\users\result\b401 b401; save c:\users\result\b402 b402;  
save c:\users\result\b403 b403; save c:\users\result\b404 b404;  
save c:\users\result\b405 b405; save c:\users\result\b406 b406;  
save c:\users\result\b407 b407; save c:\users\result\b408 b408;  
save c:\users\result\b409 b409; save c:\users\result\b410 b410;  
save c:\users\result\b411 b411; save c:\users\result\b412 b412;  
save c:\users\result\b413 b413; save c:\users\result\b414 b414;  
save c:\users\result\b415 b415; save c:\users\result\b416 b416;  
save c:\users\result\b417 b417; save c:\users\result\b418 b418;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
load c:\users\result\b119; load c:\users\result\b120;
```

```
%***** 2 *****
```

```
load c:\users\result\b201; load c:\users\result\b202;  
load c:\users\result\b203; load c:\users\result\b204;  
load c:\users\result\b205; load c:\users\result\b206;  
load c:\users\result\b207; load c:\users\result\b208;  
load c:\users\result\b209; load c:\users\result\b210;  
load c:\users\result\b211; load c:\users\result\b212;  
load c:\users\result\b213; load c:\users\result\b214;  
load c:\users\result\b215; load c:\users\result\b216;  
load c:\users\result\b217; load c:\users\result\b218;  
load c:\users\result\b219; load c:\users\result\b220;
```

```
%***** 3 *****
```

```
load c:\users\result\b301; load c:\users\result\b302;  
load c:\users\result\b303; load c:\users\result\b304;  
load c:\users\result\b305; load c:\users\result\b306;  
load c:\users\result\b307; load c:\users\result\b308;  
load c:\users\result\b309; load c:\users\result\b310;  
load c:\users\result\b311; load c:\users\result\b312;  
load c:\users\result\b313; load c:\users\result\b314;  
load c:\users\result\b315; load c:\users\result\b316;  
load c:\users\result\b317; load c:\users\result\b318;  
load c:\users\result\b319; load c:\users\result\b320;
```

```
%***** 4 *****
```

```
load c:\users\result\b401; load c:\users\result\b402;  
load c:\users\result\b403; load c:\users\result\b404;  
load c:\users\result\b405; load c:\users\result\b406;  
load c:\users\result\b407; load c:\users\result\b408;  
load c:\users\result\b409; load c:\users\result\b410;  
load c:\users\result\b411; load c:\users\result\b412;  
load c:\users\result\b413; load c:\users\result\b414;  
load c:\users\result\b415; load c:\users\result\b416;  
load c:\users\result\b417; load c:\users\result\b418;  
load c:\users\result\b419; load c:\users\result\b420;
```

```
[E,a1,b1,ref,s] = detect1(filename);  
[S,z,a2,b2,zmax] = zrocrs1(filename,a1,b1);  
normS = normalize(S,normsize);  
S1 = normS;  
[lp,ce,w2,g] = levin_durbin(S1,12,12);  
s0 = lp;
```

```
%file average
```

```
s1_1 = [b101;b102;b103;b104;b105  
        b106;b107;b108;b109;b110  
        b111;b112;b113;b114;b115  
        b116;b117;b118;b119;b120];  
s11 = mean(s1_1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

s2_1 = [b201;b202;b203;b204;b205
        b206;b207;b208;b209;b210
        b211;b212;b213;b214;b215
        b216;b217;b218;b219;b220];
s21 = mean(s2_1);

s3_1 = [b301;b302;b303;b304;b305
        b306;b307;b308;b309;b310
        b311;b312;b313;b314;b315
        b316;b317;b318;b319;b320];
s31 = mean(s3_1);

s4_1 = [b401;b402;b403;b404;b405
        b406;b407;b408;b409;b410
        b411;b412;b413;b414;b415
        b416;b417;b418;b419;b420];
s41 = mean(s4_1);

%Dynamic Time Warpping
%One
[d,A,B] = distance1(s0,s11);
[D1,g] = grad1(d,A,B);

%Two
[d,A,B] = distance1(s0,s21);
[D2,g] = grad1(d,A,B);

%Three
[d,A,B] = distance1(s0,s31);
[D3,g] = grad1(d,A,B);

%Four
[d,A,B] = distance1(s0,s41);
[D4,g] = grad1(d,A,B);

value = [D1 D2 D3 D4];
minvalue = min(value);

if (D1 == minvalue)
    D=D1;
    [H]=cportopen('lpt')
    [stat]=cportconfig(H,Name,Value,Name,Value,Name,Value...)
    [nW]=cportwrite(H,{'1'},'cr')
elseif (D2 == minvalue)
    D=D2;
    [H]=cportopen('lpt')
    [stat]=cportconfig(H,Name,Value,Name,Value,Name,Value...)
    [nW]=cportwrite(H,{'2'},'cr')
elseif (D3 == minvalue)
    D=D3;
    [H]=cportopen('lpt')
    [stat]=cportconfig(H,Name,Value,Name,Value,Name,Value...)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
[nW]=cportwrite(H,{'3'},'cr')
elseif (D4 == minvalue)
    D=D4;
    [H]=cportopen('lpt')
    [stat]=cportconfig(H,Name,Value,Name,Value,Name,Value...)
    [nW]=cportwrite(H,{'4'},'cr')
else
    disp('Not a command');
end;
```

3. โปรแกรมเดสไพล์

คำสั่งที่ใช้เรียกไฟล์ชนิดเอกซ์คิวเทเบิลขึ้นมารัน

```
winexec('outport4.exe',1)
```

คอมโพเนนท์รับโทรศัพท์อัตโนมัติได้จาก www.torry.ru



Table 8-1: MEX Script Switches (Continued)

Switch	Function
-F <file>	(UNIX) Use <file> as the options file. <file> is searched for in the following order: The file that occurs first in this list is used: <ul style="list-style-type: none"> • ./<filename> • \$HOME/matlab/<filename> • \$TMW_ROOT/bin/<filename>
-F <file>	(Windows) Use <file> as the options file. (Not necessary if you use the <code>--setup</code> option.) <file> is searched for in the current directory first and then in the same directory as <code>mex.bat</code> .
-g	Build an executable with debugging symbols included.
-h[elp]	Help; lists the switches and their functions.
-I<pathname>	Include <pathname> in the compiler include search path.
-l<file>	(UNIX) Link against library <code>lib<file></code> .
-L<pathname>	(UNIX) Include <pathname> in the list of directories to search for libraries.
<name>=<def>	(UNIX) Override options file setting for variable <name>.
-n	No execute flag. Using this option causes the commands that would be used to compile and link the target to be displayed without executing them.
-output <name>	Create an executable named <name>. (An appropriate executable extension is automatically appended.)
-O	Build an optimized executable.

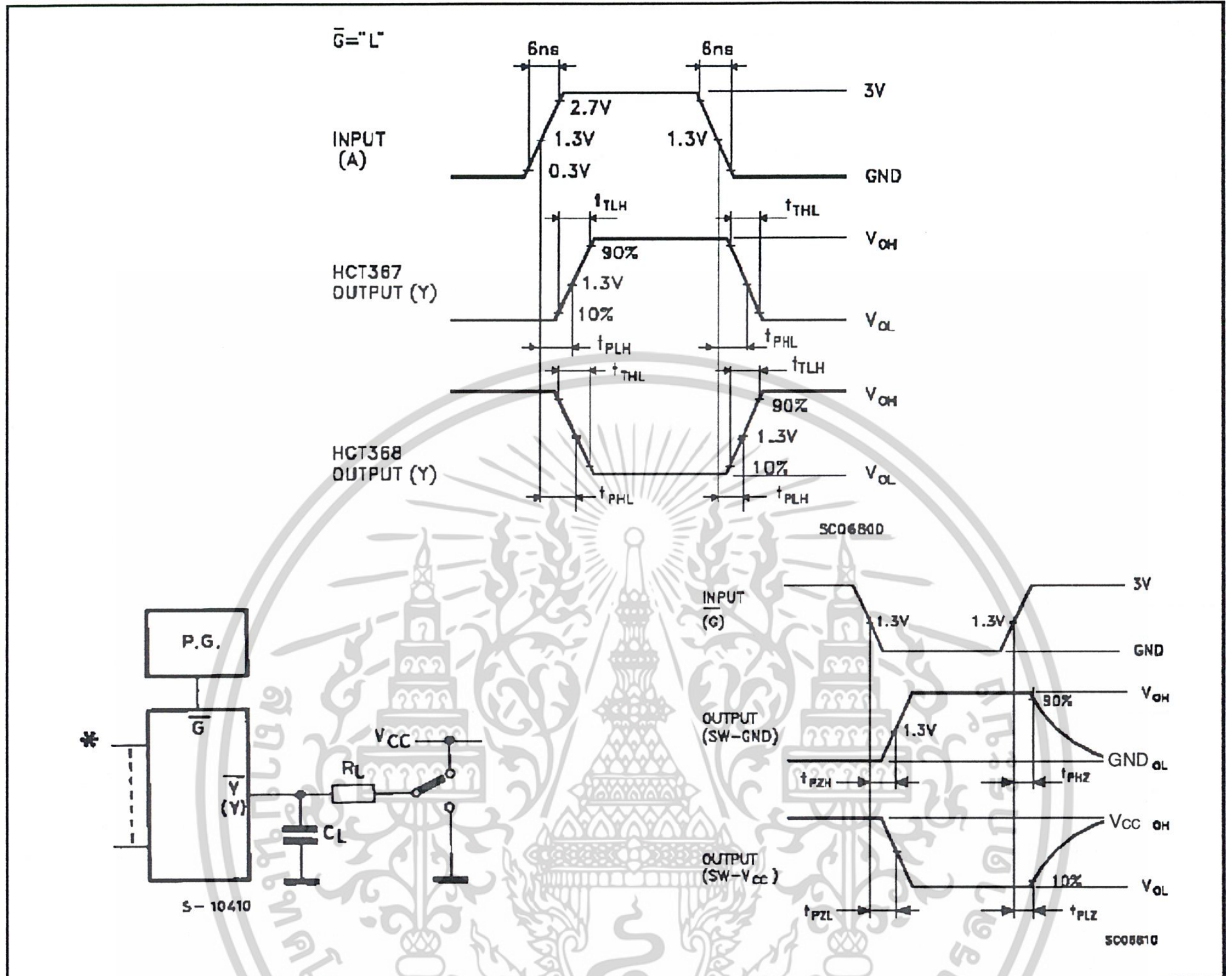
mcc (Compiler 2.0)

Bold entries in the Comment/Options column indicate default values.

Option	Description	Comment/Options
A <i>annotation:type</i>	Controls M-file code/ comment inclusion in generated C/C++ source	<i>type</i> = all comments none
A <i>debugline:setting</i>	Controls the inclusion of source filename and line numbers in run-time error messages	<i>setting</i> = on off
A <i>line:setting</i>	Controls the #line preprocessor directives included in the generated C/C++ source	<i>setting</i> = on off
B <i>filename</i>	Replaces -B <i>filename</i> on the mcc command line with the contents of <i>filename</i>	The file should contain only mcc command line options.
c	Generates C code only	Overrides -T option; equivalent to -T codegen
d <i>directory</i>	Places output in specified <i>directory</i>	
f <i>filename</i>	Uses the specified options file, <i>filename</i>	mex -setup and mbuild -setup are recommended.
F <i>option</i>	Specifies format parameters	<i>option</i> = list expression-indent: <i>n</i> page-width: <i>n</i> statement-indent: <i>n</i>
g	Generates debugging information	

Option	Description	Comment/Options
V1.2	Invokes MATLAB Compiler 1.2	Not supported in stand-alone mode
V2.0	Invokes MATLAB Compiler 2.0	
w <i>option</i>	Displays warning messages	<i>option</i> = list level level:string where level = disable enable error No w option displays only serious warnings (default).
W <i>type</i>	Controls the generation of function wrappers	<i>type</i> = mex main simulink lib:string none
x	Macro to generate MEX-function	Equivalent to: -t -W mex -L C -T link:mex
y <i>number</i>	Specifies number of outputs for Simulink S-function	
Y <i>licensefile</i>	Uses <i>licensefile</i> when checking out a Compiler license	
z <i>path</i>	Specifies <i>path</i> for library and include files	
?	Displays help message	

SWITCHING CHARACTERISTICS TEST WAVEFORM



Note : Such a logic level shall be applied to each input that the output voltage stays in the opposite side to the switch connection level, when the output is enabled.

หนังสืออ้างอิง

- [1] Duane Hanselman and Bruce Littlefield , "Mastering MATLAB 5" : Prentice Hall : Maine , 1998.
- [2] Delores M. Etter , "Engineering Problem Solving with MATLAB Second Edition" : Prentice Hall , International , Inc. : New Jersey , 1998.
- [3] Nathan Gurewich and Ori Gurewich , "Programming Sound for DOS and Windows" : SAMS Publishing : Indiana , 1988.
- [4] Muhammad Ali Mazidi and Janice Gillispie Mazidi , "The 80x86 IBM PC & Compatible Computers" : Prentice Hall : New Jersey , 1996.
- [5] จักรพงษ์ สุขประเสริฐ, กนกพร ภาวสุทธิกุล และ ศัจจะ จรัสรุ่งรวีร์ , "คู่มือการเขียนโปรแกรมด้วยเคลไฟ 4.0 ฉบับสมบูรณ์" : อินโฟเพลส : กรุงเทพฯ , พ.ศ. 2542.
- [6] นุกูล กระจาย , "การเขียนโปรแกรมในคอสและวินโดวส์ด้วยบอร์แลนด์ C++ 5.0" : บริษัท ซีเอ็ดดูเคชั่น จำกัด(มหาชน) : กรุงเทพฯ , พ.ศ. 2540.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้