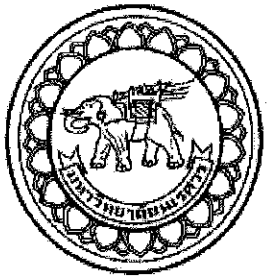
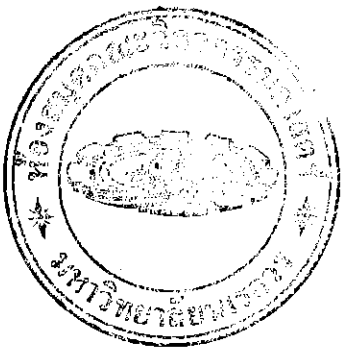




ระบบตรวจการได้ยิน

A Prototype of Hearing Loss Inspection System

นายชัชฌ์ ปราชญ์วิทยาการ รหัส 47361886

นายไพโรจน์ ตั้งจิตวิบูลย์กุล รหัส 47362330

I 4233761

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 17 พ.ย. 51
เลขทะเบียน 5100041
เลขเรียกหนังสือ ว.ร.
มหาวิทยาลัยนเรศวร ร 3515

2551

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2551

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	ระบบตรวจการได้ยิน
ผู้ดำเนินโครงงาน	นายชัชฌ์ ปราชญ์วิทยาการ รหัส 47361886
	นายไพโรจน์ ตั้งจิตวิบูลย์กุล รหัส 47362330
อาจารย์ที่ปรึกษา	ดร. อัครพันธ์ วงศ์กังแห
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2551

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ
(ดร. อัครพันธ์ วงศ์กังแห)

.....กรรมการ
(ดร.สุรเดช จิตประไพกุลศาล)

.....กรรมการ
(ดร.พนมขวัญ ริยะมงคล)

.....กรรมการ
(ดร.ไพศาล มุณีสว่าง)

หัวข้อโครงงาน	ระบบตรวจการได้ยิน	
ผู้ดำเนินโครงงาน	นายชัชฌ์ ปราชญ์วิทยาการ	รหัส 47361886
	นายไพโรจน์ ตั้งจิตวิบูลย์กุล	รหัส 47362330
อาจารย์ที่ปรึกษา	ดร. อัครพันธ์ วงศ์กังแห	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์	
ปีการศึกษา	2551	

บทคัดย่อ

ในปัจจุบัน การให้บริการ การตรวจวัดการได้ยินในที่ห่างไกลของประเทศไทย ตามสถานพยาบาลของประเทศไม่มีความสามารถให้บริการการตรวจได้เนื่องจากเครื่องมือมีราคาแพง จึงมีแนวคิดที่จะสร้างโปรแกรมตรวจการได้ยินที่มีการใช้งานง่ายและใช้ได้กับคอมพิวเตอร์ทั่วไปที่ใช้บนระบบปฏิบัติการ Window XP Service Pack 2 โดยพัฒนาขึ้นเป็นโปรแกรมต้นแบบตรวจสอบการได้ยินด้วย MATLAB 7

ผลที่ได้รับคือโปรแกรมต้นแบบที่สามารถตรวจสอบการได้ยินที่ความถี่ต่างๆในเบื้องต้นและสามารถแสดงผลการตรวจออกมาในรูปแบบกราฟ รวมทั้งสามารถจัดเก็บผลการตรวจเพื่อที่จะนำไปวินิจฉัยทางการแพทย์ต่อไป

Project Title	A Prototype of Hearing Loss Inspection System
Name	Mr. Chat Prachvidhayakarn ID. 47361886
	Mr. Phairoj Thangjitwiboonkul ID. 47362330
Project Advisor	Mr. Akaraphunt Vongkunghae, Ph.D.
Major	Computer Engineering.
Department	Electrical and Computer Engineering.
Academic Year	2008

ABSTRACT

At the present time in the remote area in Thailand, there is no service to inspect hearing loss in a hospital because the instruments for that are expensive. Therefore, it cause an idea for the researchers to create a program to inspect hearing loss that is easy to use and able to be applied with general computers using Window XP Service Pack 2 operating system. The program is improved to be the model to inspect hearing loss with MATLAB 7.

The result of this project is the program which can primarily check hearing at any rate of frequency, show the outcome of audibility examination by graph and save the outcome for further medical diagnosis.

กิตติกรรมประกาศ

โครงงานฉบับนี้ จะไม่สามารถสำเร็จลงได้หากไม่ได้รับความสนับสนุนจากบุคคลจำนวนมาก ก่อนอื่นผู้จัดทำโครงงานใคร่ขอขอบคุณ ดร.อัครพันธ์ วงศ์กังแห อาจารย์ที่ปรึกษาโครงงานด้านวิศวกรรมศาสตร์ ที่ให้ความสนับสนุนในทำโครงงาน ตลอดจนให้คำชี้แนะและมอบแนวทางการแก้ไขปัญหาของโครงงานเพื่อให้โครงงานสามารถดำเนินต่อไปได้ด้วยดีตลอดมา

ผู้จัดทำโครงการใคร่ขอขอบพระคุณ อาจารย์ ดร.ไพศาล มุณีสว่าง , อาจารย์ ดร.พนมขวัญ ริยะมงคล และ อาจารย์ ดร.สุรเดช จิตประไพกุลศาล ที่กรุณาเสียสละเวลาอันมีค่าในการตรวจสอบเนื้อหาของโครงงานฉบับนี้ และให้ความกรุณาเป็นกรรมการในการสอบโครงงาน

สุดท้ายนี้ขอขอบพระคุณอาจารย์ทุกท่านและทุกๆ คนที่ยังไม่ได้เอ่ยนามที่คอยให้การสนับสนุนผู้ดำเนินโครงงาน ให้สามารถทำโครงงานนี้จนสำเร็จลุล่วงไปได้ด้วยดี

สารบัญ

หน้า

บทคัดย่อ ..... ก

Abstract ..... ข

กิตติกรรมประกาศ ..... ค

สารบัญ ..... ง

สารบัญตาราง ..... ฉ

สารบัญรูปภาพ ..... ช

บทที่ 1 บทนำ

1.1 ที่มาของโครงงาน 1

1.2 วัตถุประสงค์ 1

1.3 ขอบข่ายการทำงาน 1

1.4 แผนการดำเนินงาน 1

1.5 ผลที่คาดว่าจะได้รับ 2

1.6 งบประมาณ 2

บทที่ 2 บทความที่เป็นประโยชน์ต่อโครงงานและทฤษฎีที่เกียวข้อง

2.1 การตรวจสมรรถภาพการได้ยิน 3

2.2 เทคนิควิธีการตรวจการได้ยิน 10

2.3 ความรู้ทั่วไปเกี่ยวกับเสียง 11

2.4 คุณสมบัติของเสียง 14

บทที่ 3 วิธีการดำเนินงาน

3.1 ความประสงค์ 17

3.2 การออกแบบโปรแกรม 18

3.3 ขั้นตอนการทำงานของโปรแกรมตรวจสอบการได้ยิน 19

3.4 รูปร่างและส่วนประกอบของตัวโปรแกรม 21

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการทดลอง

4.1 ทดสอบการเล่นเสียง 22

4.2 การทดสอบการตอบรับการได้ยินเสียง 24

4.3 การลบและบันทึกข้อมูลของการทดสอบ 26

4.4 ผลการทดสอบการทำงานของโปรแกรม 27

4.5 การเปรียบเทียบประสิทธิภาพระหว่างเครื่องตรวจการได้ยินกับโปรแกรมต้นแบบการตรวจสอบการได้ยิน 28

บทที่ 5 บทสรุป

5.1 วิเคราะห์ผลการทดลอง 30

5.2 ปัญหาและแนวทางแก้ไข 31

5.3 สรุปผลการทดลอง 31

5.4 ข้อเสนอแนะ 31

เอกสารอ้างอิง 32

ภาคผนวก ก Code โปรแกรมตรวจสอบการได้ยิน 33

ภาคผนวก ข การติดตั้งโปรแกรม MATLAB 50

ประวัติผู้เขียนโครงงาน 51

สารบัญตาราง

ตารางที่	หน้า
1.1 แผนการดำเนินการ	1
2.1 Time weighted average	5
3.1 Business Event	19
4.1 ผลการทดสอบการทำงานของโปรแกรม	27
5.1 สรุปผลการทดสอบการทำงานของโปรแกรมตาม Requirement	30

สารบัญรูปภาพ

รูปที่	หน้า
2.1 โครงสร้างของหู	4
2.2 โครงสร้างภายในของคอคเคลีย	4
2.3 บันทึกผลการตรวจ (Audiogram)	11
3.1 Context Diagram	18
3.2 ขั้นตอนการทำงานของโปรแกรมตรวจสอบการได้ยิน	19
3.3 Flow chart	20
3.4 รูปร่างของโปรแกรมตรวจสอบการได้ยิน	21
4.1 แถบปรับความถี่ (Hz)	22
4.2 แถบปรับระดับความดังของเสียง (dB)	23
4.3 Radio box เลือกหูข้างที่ต้องการทดสอบ	24
4.4 ปุ่ม Play sound	24
4.5 เส้นแสดงการตอบรับการได้ยินของหูซ้าย	25
4.6 เส้นแสดงการตอบรับการได้ยินของหูขวา	25
4.7 ปุ่มลบข้อมูลการทดสอบ	26
4.8 ปุ่มบันทึกข้อมูลการทดสอบ	26
4.9 กล่องข้อความสำหรับกรอกชื่อของไฟล์	26
4.10 กราฟที่ได้ทำการบันทึก	27
4.11 กราฟจากเครื่องตรวจสอบการได้ยิน	29
4.12 กราฟจากโปรแกรมตรวจสอบการได้ยิน	29

บทที่ 1
บทนำ

ที่มาของโครงงาน

ในปัจจุบัน การให้บริการตรวจวัดการได้ยินในที่ห่างไกลของประเทศไทย ตามสถานพยาบาลของประเทศไม่มีความสามารถให้บริการการตรวจได้เนื่องจากเครื่องมือมีราคาแพง ผู้เสนอโครงการจึงมีแนวคิดที่จะสร้างโปรแกรมตรวจการได้ยินที่มีการใช้งานง่ายและใช้ได้กับเครื่องคอมพิวเตอร์ทั่วไปที่ใช้ระบบปฏิบัติการ Windows XP Service Pack 2 ผู้เสนอโครงการคาดหวังว่าผลจากการสร้างซอฟแวร์ จะเป็นต้นแบบของระบบตรวจสอบการได้ยิน ซึ่งสามารถ

1. ตรวจสอบ อาการผิดปกติการรับรู้เสียงในช่วงความถี่ต่างๆ ในเบื้องต้นของผู้ป่วย และสามารถแสดงผลออกมาในรูปกราฟ
2. เก็บข้อมูลการได้ยินของบุคคลปกติ เพื่อใช้ในการหาจุดอ้างอิงที่ 0 db ทุกช่วง ความถี่มนุษย์สามารถรับรู้

วัตถุประสงค์

1. สร้างโปรแกรมต้นแบบตรวจสอบการได้ยิน
2. ศึกษาและประยุกต์ใช้หลักการในวิชา การออกแบบซอฟแวร์ บน Windows XP Service Pack 2
3. เพื่อช่วยลดค่าใช้จ่ายในการสั่งซื้ออุปกรณ์ ซึ่งอุปกรณ์ที่ใช้อยู่มีราคาแพงมาก

ขอบข่ายการทำงาน

1. ตัวโปรแกรมต้นแบบตรวจสอบการได้ยิน
2. ฐานข้อมูลการทดสอบของบุคคลที่ได้ทำการตรวจสอบ

แผนการดำเนินการ

ตารางที่ 1.1 แผนการดำเนินการ

กิจกรรม	2551		
	มีนาคม	เมษายน	พฤษภาคม
ศึกษาความรู้ที่เกี่ยวข้องกับการทำโปรแกรม	←——→		
พัฒนาโปรแกรม		←——→	
ทำการทดสอบและเก็บข้อมูลบุคคล			←——→

ผลที่คาดว่าจะได้รับ

1. โปรแกรมต้นแบบตรวจสอบการได้ยิน และสามารถใช้งานได้จริง
2. มีความรู้ ความเข้าใจในการเขียนโปรแกรมมากขึ้น
3. ลดค่าใช้จ่ายในการสั่งซื้ออุปกรณ์ ซึ่งอุปกรณ์ที่ใช้อยู่มีราคาแพงมาก

งบประมาณ

1.	ค่าถ่ายเอกสารและค่าเข้าเล่มโครงงาน	เป็นเงิน	400	บาท
2.	ค่าหนังสือ	เป็นเงิน	600	บาท
3.	ค่าวัสดุสำนักงาน	เป็นเงิน	400	บาท
4.	ค่าวัสดุคอมพิวเตอร์	เป็นเงิน	600	บาท
		รวมเป็นเงินทั้งสิ้น	2,000	บาท

หมายเหตุ ถัวเฉลี่ยทุกรายการ

บทที่ 2
บทความที่เป็นประโยชน์ต่อ โครงงานและทฤษฎีที่เกี่ยวข้อง

การทำงานของเครื่องตรวจวัดการได้ยินที่ใช้อยู่ในปัจจุบันนี้ ตัวเครื่องจะทำการสร้างคลื่นเสียงในช่วงความถี่ต่างๆ ออกมาทางหูฟัง ผู้ที่รับการตรวจจะต้องใส่หูฟังเพื่อฟังเสียงที่สร้างออกมา พอได้ยินเสียงแล้วจึงกดปุ่ม ตัวเครื่องก็จะทำการเก็บข้อมูลของผู้ที่รับการตรวจไว้ ว่าสามารถรับรู้ได้ที่ระดับความถี่และระดับเสียงที่เท่าไร และเมื่อทำการตรวจวัดจนครบถ้วนทุกกระบวนการแล้ว เครื่องจะทำการแสดงผลการตรวจวัดออกมาในรูปแบบกราฟ

ดังนั้นโครงการนี้จึงใช้หลักการและทฤษฎีจากงานด้านคลื่นเสียงและด้าน GUI (Graphic User Interface) โดยเลือกโปรแกรม MATLAB 7 เพื่อใช้ให้การสร้างต้นแบบโปรแกรมการตรวจวัดการได้ยินขึ้นมา

2.1 การตรวจสมรรถภาพการได้ยิน

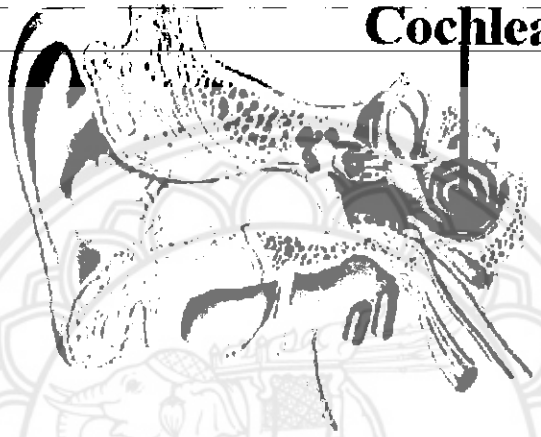
การตรวจสมรรถภาพการได้ยิน หรือ Audiography เป็นการตรวจการได้ยินเสียง ณ ความถี่ต่างๆ ตั้งแต่ระดับความถี่เสียงสนทนาจนถึงเสียงเครื่องจักร ซึ่งเป็นความถี่ที่ไม่ได้ยินกันในชีวิตประจำวัน หรือคนทั่วไปที่ไม่ได้มีหน้าที่เกี่ยวข้องโดยตรงจะไม่ได้มีโอกาสสัมผัส โดยการตรวจจะนำข้อมูลไปสร้างเป็นกราฟ เรียกว่า ออดิโอแกรม (audiogram) ซึ่งการแปลผลว่ามีสมรรถภาพการได้ยินเป็นอย่างไรนั้นจะดูจากกราฟนี้

การที่เราจะเข้าใจว่าการตรวจสมรรถภาพการได้ยินมีประโยชน์อย่างไร ควรทราบถึงกลไกการได้ยินและอันตรายของเสียงดังก่อนดังนี้

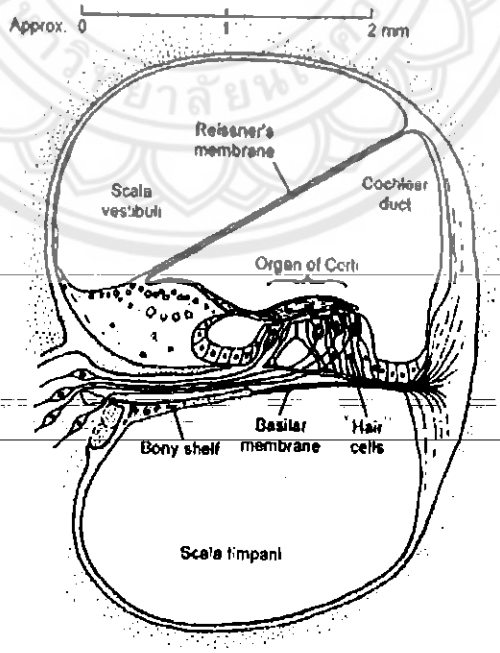
2.1.1 กลไกการได้ยินและอันตรายของเสียงดังต่อมนุษย์

คนเราสามารถได้ยินเนื่องจากคลื่นเสียงเคลื่อนที่จากหูชั้นนอกเข้าสู่หูชั้นกลางแล้วเข้าสู่หูชั้นใน การทำงานของหูในช่วงตั้งแต่ใบหู รูหู กระดูกหูชั้นกลาง จัดเป็นการนำเสียงผ่านโมเลกุลของอากาศ ซึ่งจะส่งต่อไปยังหูชั้นกลางในหูชั้นกลางจะมีกระดูกหู 3 ชิ้นส่งคลื่นเสียงเข้าไปสู่หูชั้นใน ที่กระดูกนี้จะมีกล้ามเนื้ออยู่ซึ่งเมื่อมีเสียงดังมากเกินไปผ่านเข้ามา ร่างกายจะมีกลไกป้องกัน โดยให้กล้ามเนื้อนี้จะหดตัวอัตโนมัติจะช่วยลดระดับเสียงที่จะผ่านเข้าไปสู่หูชั้นในได้ประมาณ 30 – 40 dB ที่หูชั้นใน จะมีอวัยวะรูปก้นหอยเรียกคอคเคลีย (Cochlea) ภายในกลวงบรรจุของเหลวไว้ พื้นของคอคเคลียจะบุด้วยเซลล์ขน (Hair cell) ซึ่งทำหน้าที่รับความรู้สึกสั่นสะเทือนแปลงเป็นคลื่นประสาทส่งไปสมองเพื่อแปลความหมายของเสียงที่ได้ยิน การนำคลื่นเสียงจากอากาศมาสู่หูชั้นกลางเรียกว่า Conductive Function และการนำคลื่นเสียงจากการสั่นสะเทือนแล้วแปลเป็นกระแสประสาทเพื่อส่งไปรับรู้ที่สมองจะเรียกว่า Sensorineural Function ตัวเซลล์ขนนี้จะมีความยาวไม่

เท่ากันและมีความจำเพาะเจาะจงต่อความถี่ใดความถี่หนึ่ง ถ้าเสียงดังมาก เกินกว่าการป้องกันโดยธรรมชาติของร่างกายคือถ้ามีระดับความเข้มของเสียงสูงกว่า 85 dB เมื่อคลื่นเสียงเดินทางมาถึงเซลล์ขนจะทำให้มีการสั่นสะเทือนอย่างต่อเนื่องเป็นเวลานานติดต่อกัน เซลล์ขนจะไม่สามารถปรับสภาพคืนสู่ปกติ และหลุดร่วงไป ก็จะเกิดการขาดช่วงการเดินทางของเสียงที่ไปยังสมอง เกิดการสูญเสียการได้ยินขึ้น และก็จะเป็นเฉพาะความถี่ของเสียงดังนั้นๆ คอคเคลียอยู่ในหูชั้นใน ดังรูปที่ 2.1 และเมื่อนำไปส่องกล้องจุลทรรศน์ ภายในจะกลวงและมีเซลล์ขนดังรูปที่ 2.2



รูปที่ 2.1 โครงสร้างของหู



รูปที่ 2.2 โครงสร้างภายในคอคเคลีย

การที่เซลล์ขนถูกทำลาย ทำให้เกิดหูตึงได้ 2 ลักษณะ คือ

2.1.1.1 Acoustic trauma คือ การสูญเสียการได้ยินอย่างฉับพลันเมื่อได้ยินเสียงดังมาก เช่น เสียงระเบิด เสียงปืน ฯลฯ

2.1.1.2 Noise induced hearing loss คือ การสูญเสียการได้ยินแบบค่อยเป็นค่อยไป เกิดขึ้นในผู้ที่ทำงานอยู่ในที่ที่มีเสียงดังเป็นเวลานานๆ เช่น อุตสาหกรรมสิ่งทอ, อุตสาหกรรมเครื่องเรือน, อุตสาหกรรมถลุงเหล็ก, อุตสาหกรรมเครื่องแก้ว, อุตสาหกรรมเครื่องเหล็ก, โรงเลื่อย, ขับเรือหางยาว, ขับรถสามล้อเครื่อง, ตำรวจจราจร, นักจัดรายการดนตรี จากการศึกษาวิจัยพบว่า ในกลุ่มคนงานที่ทำงานสัมผัสกับเสียงที่ดังกว่า 85 dBA นาน 8 ชั่วโมง/วัน ติดต่อกันนาน 5 ปี มีโอกาสที่จะทำให้สมรรถภาพการได้ยินเสียไปดังตารางนี้

ตารางที่ 2.1 Time weighted average

Duration per day, hours	Sound level dBA slow response
8	90
6	92
4	95
3	97
2	100
1.5	102
1	105
0.5	110
¼ or Less	115

2.1.2 อันตรายของเสียงต่อสุขภาพทั่วไป

2.1.2.1 ทำให้การทำงานของระบบการไหลเวียนโลหิต ระบบประสาทและระบบต่อมไร้ท่อทำงานผิดปกติ

2.1.2.2 ทำให้สมดุลร่างกายเปลี่ยนแปลงโดยทำให้ความดันโลหิตสูงขึ้นกว่าปกติ การเต้นของหัวใจผิดปกติ และการหดตัวของเส้นเลือดผิดปกติ

2.1.3 อันตรายของเสียงต่อความปลอดภัยในการทำงาน

2.1.3.1 ทำให้พฤติกรรมส่วนบุคคลเปลี่ยนแปลง เช่น เชื่องช้าต่อการตอบสนองต่อสัญญาณต่างๆ และเกิดความว้าวุ่นใจในการทำงาน ทำให้การทำงานผิดพลาดจนเกิดอุบัติเหตุได้

2.1.3.2 รบกวนการทำงานทำให้ประสิทธิภาพการทำงานลดลง ลักษณะของเสียงที่พบว่ามีผลต่อการลดประสิทธิภาพการปฏิบัติงาน ได้แก่

- เสียงดังๆ หยุดๆ เป็นช่วง (Transient noise)
- เสียงที่มีความถี่สูงกว่า 2,000 Hz.
- เสียงที่ดังต่อเนื่องเป็นระยะเวลานาน (Continuous noise)
- เสียงที่มีลักษณะต่างๆ ข้างต้นผสมผสานกัน

2.1.3.3 รบกวนการนอนหลับ ทำให้เกิดความอ่อนเพลีย เมื่อปฏิบัติงานอาจเกิดความผิดพลาดได้ง่าย

2.1.3.4 รบกวนการติดต่อสื่อสาร

2.1.4 ผลกระทบของเสียงดังต่อสุขภาพ

มีหลักฐานชัดเจนเฉพาะในหัวข้อแรก (สูญเสียการได้ยิน)

2.1.4.1 สูญเสียการได้ยิน (Noise Induced Hearing Loss)

2.1.4.1.1 สูญเสียการได้ยินอย่างเฉียบพลันจากเสียงที่ดังมาก

2.1.4.1.2 สูญเสียการได้ยินแบบถาวร วัดได้หลายวิธี ที่ใช้บ่อยคือวัดจากการสูญเสียการได้ยินเฉลี่ยในช่วงความถี่ 1, 2, 3 และ 4 kHz ถ้าไม่ได้ยินเสียงที่ 25 dBA ให้ถือว่ามีความผิดปกติของการได้ยิน การสูญเสียการได้ยินนี้อาจเกิดขึ้นเร็วมาก แต่โดยทั่วไปถ้าสัมผัสเสียงดัง 90 dBA การสูญเสียจะเป็นดังนี้

- ใน 1-2 ปี เสียการได้ยินที่ 4 kHz = 10 dB
- ใน10 ปี เสียการได้ยินระหว่าง 3-6 kHz = 20 dB
- ใน 30 ปี เสียการได้ยิน ระหว่าง 2-6 kHz => 40 dB
- ผู้สัมผัสเสียงดัง 85 dBA เป็นเวลา 40 ปี ก็มีโอกาสสูญเสียการได้ยินได้ประมาณ 8%
- มีเสียงดังรบกวนในหู (Tinnitus)

2.1.4.1.3 สูญเสียการได้ยินชั่วคราว Temporary Threshold Shift แม้จะมีผลตามตารางที่2.1 แต่ได้ยินเสียงดังเป็นช่วงๆ การสูญเสียการได้ยินชั่วคราวนี้อาจกลายเป็นสูญเสียถาวรได้

2.1.4.2 ผลกระทบนอกเหนือจากการได้ยิน ได้แก่

- ผลกระทบต่ออารมณ์ (Psychological Stress) หงุดหงิด โมโหง่าย
- กล้ามเนื้อเเข็งตึง
- ระบบย่อยอาหารผิดปกติ
- ความดันโลหิตสูง

- ระบบต่อมไร้ท่อแปรปรวน
- หัวใจขาดเลือด

2.1.5 ประเภทของความสูญเสียการได้ยิน

ประเภทของความสูญเสียการได้ยิน แบ่งได้ 5 ประเภท ดังนี้

2.1.5.1 การนำเสียงบกพร่อง (Conductive hearing loss) ความผิดปกติเกิดขึ้นในหูชั้นนอกและชั้นกลาง แต่ ประสาทหูยังดีอยู่

อาการ

มีของเหลวออกจากช่องหูอาจจะเป็นเลือดหรือหนอง มีประวัติการอักเสบของช่องหูมาก่อน การพูดคุยมักพูดเสียงเบาทุ้มนุ่มนวล การได้ยินจะดีชัดเจนเมื่ออยู่ในที่จอแจแต่ไม่ค่อยดีในที่เงียบๆ มักมีปัญหาในการฟังเสียงขณะเคี้ยวอาหาร บางรายมีเสียงรบกวนในหู (Tinnitus) เป็นเสียงต่ำๆ การพูดจาชัดเจนออกเสียงได้ตามปกติ ตรวจการได้ยินพบการสูญเสียในช่วงความถี่ต่ำๆ และมักไม่มากกว่า 60 dBA

สาเหตุ

- โรคหรือความผิดปกติที่หูชั้นนอก : หูพิการตั้งแต่กำเนิด สิ่งแปลกปลอมทำให้เกิดการอุดตันในช่องหู ขี้หูอุดตัน ผนังช่องหูอักเสบบวม จนช่องหูตีบตัน โรคเนื้องอกในช่องหูชั้นนอก ช่องหูพับลง
- โรคหรือความผิดปกติที่แก้วหู : มีรูทะลุที่เยื่อแก้วหู แก้วหูอักเสบ เยื่อแก้วหูหนา
- โรคหรือความผิดปกติในหูชั้นกลาง : มีเลือดออกในหูชั้นกลาง, โรคหูน้ำหนวก (ทั้งชนิดมีน้ำไหลและแห้ง), โรคหูชั้นกลางมีหินปูนจับแข็ง, ภาวะแทรกซ้อนจากการติดเชื้อไวรัส, กระดูก 3 ชิ้นแตกหรือหัก

2.1.5.2 ประสาทรับฟังเสียงบกพร่อง (Sensorinural hearing loss) ความผิดปกติที่เกิดขึ้นในหูชั้นใน (cochlea) หรือที่ประสาทรับฟังเสียง (acoustic nerve)

อาการ

ถ้ามีการสูญเสียของประสาทหูมากทั้ง 2 ข้างและเป็นเวลานาน เสียงพูดจะดังมากกว่าปกติ เพราะไม่ได้ยินเสียงตัวเอง มีเสียงรบกวนในหูเป็นเสียงสูงๆ จะฟังเสียงพูดได้ดีเมื่ออยู่ในที่สงบและจะไม่ค่อยเข้าใจคำพูดเมื่ออยู่ในที่จอแจ มักไม่ค่อยเข้าใจคำพูดแม้ว่าเสียงพูดนั้นดังถึงระดับการได้ยินปกติแล้วก็ตาม มักมีอาการเวียนศีรษะแบบบ้านหมุนร่วมด้วย ถ้าประสาทหูเสียมากทั้ง 2 ข้าง หรือเป็นมาแต่กำเนิดมักจะพูดไม่ชัดหรือพูดไม่ได้ ไม่มีประวัติของการปวดหู หรือมีของเหลวไหลออกจากหู ตรวจการได้ยินพบการสูญเสียในช่วงความถี่สูงๆ

สาเหตุ

- ประสาทรับฟังเสียงบกพร่องแต่กำเนิด : ขาดออกซิเจนขณะอยู่ในครรภ์หรือระหว่างคลอด, ติดเชื้อแต่กำเนิดหรือหลังคลอด เช่น ซิฟิลิส หัด หัดเยอรมัน คางทูม สุกใส ไข้หวัดใหญ่ ปอดอักเสบ, การอักเสบของเยื่อหุ้มสมองหรือหูชั้นใน
- ประสาทรับฟังเสียงบกพร่องจากยา : ผู้ป่วยจะมีการสูญเสียการได้ยินของหูทั้ง 2 ข้างพร้อมๆ กัน ยาบางชนิดทำให้มีอาการชั่วคราว เมื่อหยุดยาการได้ยินอาจกลับคืนมาได้ แต่ยาบางชนิดทำให้มีอาการถาวรรักษาไม่หาย เช่น kanamycin, streptomycin
- ประสาทรับฟังเสียงบกพร่องจากเสียงดัง (Noise induced hearing loss)

โรคที่เกิดจากความผิดปกติเกี่ยวกับปริมาณของของเหลวในหูชั้นใน (Meniere's disease) ทำให้มีอาการหูอื้อ เวียนศีรษะ บ้านหมุน คลื่นไส้อาเจียน และมีเสียงรบกวนในหู อาจเป็นหูเดียวหรือสองหูก็ได้ อาการของโรคจะเป็นซ้ำๆ กัน มีอาการเป็นๆ หายๆ

- ประสาทหูพิการจากการจับแข็งของกระดูกในหูชั้นใน
- ประสาทหูบกพร่องในวัยชรา (Presbycusis hearing loss) ความผิดปกติเกิดขึ้นจากเซลล์ขนที่อยู่บริเวณฐานของก้นหอยในหูชั้นในมีการเสื่อมไปตามอายุ ทำให้การรับฟังเสียงสูงๆ ได้ไม่ดี มักมีเสียงดังในหูเป็นเสียงสูงๆ ตรวจช่องหูไม่พบสิ่งผิดปกติ มีความผิดปกติของการได้ยินของหูทั้งสองข้าง มักพบในคนที่อายุ 40 ปีขึ้นไป
- ศีรษะทูกกระทบกระเทือน ทำให้ประสาทรับฟังเสียงบกพร่องเล็กน้อยไปจนถึงระดับรุนแรง

2.1.5.3 การรับฟังเสียงบกพร่องแบบผสม (Mixed hearing loss) เป็นภาวะที่เกิดจากความผิดปกติในระบบการนำเสียงร่วมกับประสาทรับฟังเสียงบกพร่อง พบในโรคที่มีความพิการที่หูชั้นนอก ชั้น-กลาง และชั้นในร่วมกัน เช่น โรคหูน้ำหนวกเรื้อรังซึ่งอาการลุกลามเข้าไปในหูชั้นใน โรคหินปูนจับแข็งที่กระดูกโกลน

2.1.5.4 ความผิดปกติทางจิต (Functional or Psychological hearing loss)

2.1.5.5 ความบกพร่องที่สมองส่วนกลาง (Central Hearing Impairment) สมองไม่สามารถรับและแปลความหมายได้ จึงไม่สามารถเข้าใจความหมายของเสียงที่ได้ยิน เช่น โรคเส้นเลือดในสมองแตก ทำให้ศูนย์การรับฟังไม่สามารถใช้การได้

2.1.6 การสูญเสียการได้ยินเนื่องจากเสียงดัง (Noise Induced Hearing Loss)

ประเภทการสูญเสียการได้ยินเนื่องจากเสียงดัง แบ่งได้ออกเป็น 2 ประเภทดังนี้

2.1.6.1 ประสาทหูผิดปกติเนื่องจากเสียงดังรบกวน แบ่งได้อีก 2 ประเภทดังนี้

2.1.6.1.1 การสูญเสียความสามารถในการได้ยินชั่วคราว (Temporary thresholds shift: TTS) มีสาเหตุเกิดจากเซลล์ประสาทรับการได้ยินมีอาการล้าจากการสัมผัสเสียงดังต่อเนื่องเป็นเวลานานๆ ไม่สามารถแปลสัญญาณการสั่นสะเทือนเป็นคลื่นประสาทได้ เกิดอาการหูตึงชั่วคราว (Auditory fatigue) อาการหูตึงนี้มักร่วมกับมีเสียงดังในหู (tinnitus) ในกรณีสงสัยว่าจะสูญเสียความสามารถในการได้ยินชั่วคราว ควรให้พนักงานพักจากการฟังเสียงที่ต่ำกว่า 70 dBA อย่างน้อย 48 ชั่วโมง

2.1.6.1.2 การสูญเสียความสามารถในการได้ยินถาวร (Permanent threshold shift: PTH) เมื่อผู้ป่วยมีอาการล้าของเซลรับเสียงจนไม่สามารถได้ยินเสียงในระดับปกติ หากยังสัมผัสกับเสียงดังต่อเนื่องอีกก็จะทำให้เซลรับเสียงถูกทำลายอย่างถาวร (Degenerative change of hair cell) ในระยะแรกการสูญเสียการได้ยินจะเริ่มเสียที่ช่วงความถี่ของเสียง 3,000 – 6,000 Hz. และจะพบเสมอว่าจะเสียที่ความที่ของการได้ยินที่ 4,000 Hz. ก่อนความถี่อื่นๆ จากนั้นจะเริ่มมีเสียงดังรบกวนในหู ความไวของหูในการรับเสียงลดลง แต่พอเลิกงานไม่ได้อยู่ในที่ที่มีเสียงดังจะรู้สึกว่าการได้ยินดีขึ้น อาจมีอาการปวดหูหรือเวียนศีรษะร่วมด้วย

เมื่อทำงานในที่มีเสียงดังเป็นระยะเวลานานๆ จะมีการสูญเสียการได้ยินไปทีละน้อย โดยไม่รู้สึกตัว จนลุกลามไปถึงช่วงความถี่ของการพูดคุย (500–2,000 Hz.) ทำให้การรับฟังเสียงคำพูดไม่เข้าใจ ถ้าผิดปกติมากจะไม่ทราบทิศทางของเสียงที่ได้ยิน การตรวจภายในช่องหูไม่พบสิ่งผิดปกติ ตรวจวัดการได้ยินด้วยเครื่องตรวจวัดการได้ยิน จะได้กราฟลักษณะเส้นประสาทหูผิดปกติ

2.1.6.2 ประสาทหูผิดปกติเนื่องจากมีเสียงดังมาก ๆ

- หูอื้อทันทีหลังจากได้รับเสียงดัง
- มีเสียงดังในหูตลอดเวลา
- มักฟังคำพูดเข้าใจดี เนื่องจากการได้ยินไม่เสียที่บริเวณความถี่ของการพูดคุย
- เมื่อตรวจวัดการได้ยินพบว่ามีลักษณะความผิดปกติ
- ตรวจภายในช่องหูพบว่า ช่องหูชั้นนอกปกติ แต่อาจมีแก้วหูทะลุร่วมด้วย

2.2 เทคนิควิธีการตรวจการได้ยิน

2.2.1 วิธีการตรวจการได้ยิน

2.2.1.1 Descending Technique โดยการปล่อยระดับเสียงที่ดัง เพื่อให้ผู้รับการตรวจได้ยินก่อนแล้วค่อยๆลดความดังลงทีละน้อย ทีละ 10 dBA จนถึงจุดหนึ่งที่ผู้รับการตรวจไม่ได้ยินเสียง ให้เพิ่มระดับเสียงจากจุดที่ไม่ได้ยิน ทีละ 5 dBA หากไม่ได้ยินก็ให้เพิ่มอีก 5 dBA จนเริ่มได้ยิน แล้วลดลงไปอีก 10 dBA เมื่อแน่ใจว่าผู้รับการตรวจได้ยินแน่ชัดที่จุดนั้นๆ ให้ลดลง 10 dBA อีกครั้ง ถ้าไม่ได้ยิน ให้เพิ่มขึ้น 5 dBA ทำกลับไปกลับมาจนได้จุดที่ผู้รับการตรวจได้ยินโดยใช้ระดับเสียงเบาที่สุดที่ผู้ถูกตรวจสามารถตอบสนองได้ร้อยละ 50 ถึง 70 ของจำนวนครั้งที่ให้สัญญาณ จุดนั้นคือ hearing threshold

2.2.1.2 Ascending Technique ใช้ในกรณีที่ผู้รับการตรวจอายุน้อย หรือหูหนวกมากๆ รวมทั้งผู้ที่ไม่แน่ใจว่าจะแสร้งทำเป็นหูหนวกหรือไม่ วิธีนี้เริ่มจากความตั้งใจที่ผู้รับการตรวจไม่ได้ยินก่อน แล้วเพิ่มความดังทีละ 10 dBA จนถึงจุดที่ผู้รับการตรวจเริ่มได้ยินเสียงเบาที่สุด แล้วลดเสียงลง 5 dBA ทำกลับไปกลับมาจนได้จุดที่ผู้รับการตรวจได้ยินเสียงบ้างไม่ได้ยินเสียงบ้าง จุดนั้นคือ hearing threshold

2.2.1.3 Combination Technique ใช้วิธีผสมระหว่างวิธีที่ 1 และที่ 2 โดยใช้ระดับเสียงดัง-เบาสลับกันไป

2.2.2 การเตรียมผู้รับการตรวจ

ก่อนตรวจผู้รับการตรวจควรงดรับฟังเสียงดังเกิน 80 dBA เป็นเวลา 8 - 16 ชั่วโมง เพื่อหลีกเลี่ยงการเกิดหูตึงแบบชั่วคราว หากไม่สามารถทำได้ในทางปฏิบัติ ต้องสวมใส่ที่ครอบหูลดเสียงตลอดเวลาที่สัมผัสเสียงก่อนการทดสอบการตรวจการได้ยินโดยการนำเสียงผ่านทางอากาศ

2.2.2.1 ให้ผู้รับการตรวจนั่งในห้อง ที่มีระดับเสียงในห้องตามมาตรฐานกำหนดไม่เกิน 40 dBA ในทุกความถี่

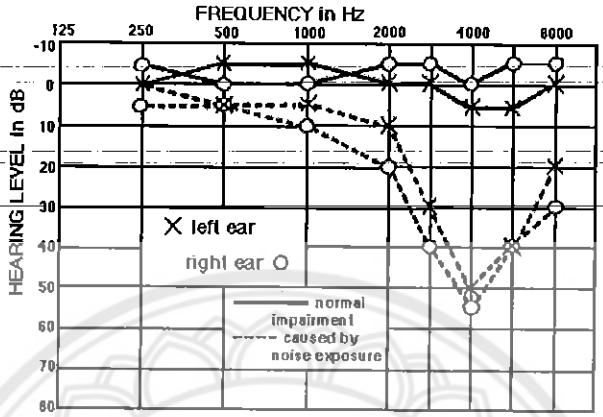
2.2.2.2 อธิบายให้ผู้รับการตรวจเข้าใจถึงเสียงสัญญาณที่จะได้ยินและการกดสวิทซ์สัญญาณตอบรับ

2.2.2.3 ให้ผู้รับการตรวจนั่งหันหลังให้ผู้ทำการทดสอบและใช้หูฟัง (Head phone) ครอบหูทั้งซ้ายและขวา

2.2.2.4 การสอบถามผู้รับการตรวจ และทำการทดสอบในหูข้างที่ดีก่อน เริ่มทดสอบ hearing threshold ที่ความถี่ 1,000 Hz. แล้วหาต่อไปที่ 2,000 3,000 4,000 6,000 และ 8,000 Hz. แล้วกลับมาทดสอบซ้ำที่ 1,000 Hz. ใหม่ แล้วหาต่อไปที่ 500, 250 Hz. ตามลำดับ ทำการตรวจการได้ยินของหูอีกข้างตามวิธีข้างต้น

2.2.3 การบันทึกผลการตรวจ

การตรวจการได้ยินโดยการนำเสียงผ่านทางอากาศ ให้ใช้เครื่องหมายสำหรับการบันทึกผลการตรวจการได้ยินตามหลักสากล โดยใช้เครื่องหมาย O สีแดง สำหรับหูข้างขวา X สีน้ำเงินสำหรับหูข้างซ้าย การลากเส้นใช้เส้นทึบดังรูป



รูปที่ 2.3 บันทึกผลการตรวจ (Audiogram)

2.3 ความรู้ทั่วไปเกี่ยวกับเสียง

2.3.1 ความหมายที่เกี่ยวกับเสียง

เสียง คือ พลังงานที่เกิดจากการสั่นสะเทือนของโมเลกุลของตัวกลางที่เสียงเคลื่อนที่ผ่าน (ตัวกลางนี้อาจเป็นอากาศ ของแข็ง ของเหลว หรือก๊าซก็ได้) เป็นเหตุให้เกิดการอดและขยายตัวของอากาศสลับกันไป ซึ่งมีผลทำให้ความดันบรรยากาศเปลี่ยนแปลงสูงต่ำสลับกันเป็นคลื่น เช่นเดียวกับการอัดขยายของอากาศ คลื่นที่เกิดขึ้นนี้เรียกว่า คลื่นเสียง เสียงที่มนุษย์ได้ยินเป็นผลจากการเปลี่ยนแปลงอย่างรวดเร็วของความดันอากาศจากการสั่นสะเทือนของวัตถุคลื่นในอากาศมากระทบหูระบบประสาทหูและสมอง มนุษย์สามารถรับฟังคลื่นเสียงที่มีความถี่ตั้งแต่ 20 – 20,000 รอบ/วินาที ช่วงความถี่ของคลื่นเสียงที่มนุษย์ได้ยินนี้ เรียกว่า "คลื่นออดิเบิล" (audible waves) คลื่นความถี่ต่ำกว่า audible waves เรียกว่า infrasonic waves และคลื่นความถี่สูงกว่า audible waves เรียกว่า ultra sonic waves

2.3.2 องค์ประกอบของคลื่นเสียง

คลื่นเสียงประกอบด้วยองค์ประกอบ 3 อย่าง

2.3.2.1 ความถี่ (frequency) คือ อัตราการเคลื่อนไหวของคลื่นที่นับเป็นรอบ/วินาที หรือมีหน่วยเป็นเฮิรตซ์ (Hertz) นิยมใช้ตัวย่อ Hz

2.3.2.2 แอมปลิจูด (amplitude) คือ ปริมาณความแรงของคลื่นที่อัดอากาศจากศูนย์กลางการสั่นของวัตถุต่อเนื่องกันไป หน่วยเป็น Pascal นิยมใช้ตัวย่อ Pa

2.3.2.3 เฟส (phase) คือ ส่วนของคลื่นในพิกัดของเวลาต่างๆ สำหรับคลื่นเสียง ส่วนของคลื่นครบรอบ คือ 360 องศา

2.3.3 การวัดความแรงเสียง ปริมาณที่ใช้แสดงความแรงของคลื่นเสียงในอากาศ คือ แอมปลิจูด ซึ่งสามารถวัดได้ ดังนี้

2.3.3.1 ความดันเสียงและระดับความดันเสียง (sound pressure and sound pressure level) คือ ค่าความดันของคลื่นเสียงที่เปลี่ยนไปจากความดันบรรยากาศปกติหน่วยที่ใช้อาจเป็น นิวตัน/ตารางเมตร (N/m^2) หรือ Pascal (Pa) ความดันเสียงต่ำสุดที่หูคนปกติสามารถได้ยินที่ความถี่ 1,000 Hz คือ 2×10^{-5} N/m^2 ค่าความดันอ้างอิงในการตรวจวัดระดับความดันเสียงซึ่งมีหน่วยในการวัดเสียงเป็นเดซิเบล ความดันเสียง 2×10^{-5} N/m^2 = 0 เดซิเบล

2.3.3.2 ความเข้มเสียง (sound intensity) คือ พลังงานเสียงเฉลี่ยต่อหนึ่งหน่วยเวลาที่ผ่านพื้นที่หนึ่งหน่วยตามทิศทางการกระจายตัวของเสียงมีหน่วยเป็นวัตต์/ตารางเมตร (W/m^2) ระดับความเข้มของเสียง (sound intensity level) เป็นค่าความเข้มของเสียงที่วัดได้เปรียบเทียบกับความเข้มของเสียงอ้างอิง ซึ่งมีค่า 10^{-12} W/m^2 ซึ่งเป็นความเข้มเสียงต่ำสุดที่หูคนปกติสามารถได้ยิน เมื่อต้องการวัดความเข้มของเสียงในรูประดับความเข้มของเสียงมีหน่วยเป็นเดซิเบล ใช้ระดับความเข้มอ้างอิง จะได้ว่าความเข้มเสียง 10^{-12} W/m^2 = ระดับความเข้มของเสียง 0 เดซิเบล

2.3.3.3 สื่อและความเร็วเสียง (the medium and the speed of sound) โดยทั่วไปเสียงทุกความถี่จะเคลื่อนที่ไปด้วยความเร็วที่เท่ากันเสมอ ในสื่อชนิดหนึ่งๆ ความเร็วของเสียงจะแปรผันตามความอัด (compessibillity) ความหนาแน่นของสื่อจะมีค่าคงที่ในอุณหภูมิที่กำหนดให้

2.3.3.4 มาตราเดซิเบล (Decibel, dB) เป็นหน่วยของระดับความดันและระดับความเข้มของเสียง ซึ่งถูกคิดค้นมาเพื่อความสะดวกในการวัดความดันและความเข้มของเสียงและเนื่องจากการตอบสนองของหูต่อความดันเสียงมีลักษณะเป็นล็อกกาลิทึ่ม ทำให้ได้สูตรในการคำนวณระดับความดันและระดับความเข้มของเสียง ดังนี้

ระดับความดัน (เดซิเบล) = $\log (P/P_{xf})^2$ [2]

เมื่อ P = ความดันเสียง (N/m^2)

P_{xf} = ความดันเสียงอ้างอิง (2×10^{-5} N/m^2)

ระดับความเข้ม (เดซิเบล) = $10 \log (I/I_{xf})$ [2]

เมื่อ I = ความดันเสียง (W/m^2)

I_{xf} = ความดันเสียงอ้างอิง (10^{12} W/m^2)

2.3.4 มาตรฐานอ้างอิงกับการวัดเสียง มีดังนี้

- dBSPL หมายถึง ระดับความดันเสียง ที่เปรียบเทียบกับความดันอ้างอิง 0.00002 Pa หรือ 2×10^{-5} N/m^2

- dBIL หมายถึง ระดับความเข้มเสียง ที่เปรียบเทียบกับความเข้มอ้างอิง 10^{-12} W/m^2
- dBA หมายถึง ระดับความดันเสียงที่เป็นอัตราส่วนเปรียบเทียบกับค่าความดันทุกความถี่ที่วัดได้จากมาตรวัดระดับเสียง (sound level meter) ซึ่งใช้วงจรถ่วงน้ำหนัก A (A-weighting network) ประกอบเข้าในเครื่อง วงจรนี้สามารถตอบสนองและชดเชยความถี่ต่ำและความถี่สูงได้ดีเป็นพิเศษและเป็นที่นิยมใช้วัดระดับเสียงรบกวนและงานวิจัยด้านมลพิษของเสียง (noise pollution) เนื่องจากมีลักษณะการตอบสนองต่อเสียงใกล้เคียงกับหูมนุษย์มากที่สุด
- dBSL หมายถึง ระดับความดันเสียงที่เปรียบเทียบกับความดันเฉพาะความถี่ของเครื่องตรวจการได้ยิน (audiometer) ซึ่งเป็นค่าเฉลี่ยของขีดเริ่มการได้ยินของผู้รับฟังนั้น
- dBHL หมายถึง ระดับความดันเสียงที่เปรียบเทียบกับความดันเฉพาะความถี่ของเครื่องตรวจการได้ยิน (audiometer) ซึ่งเป็นค่าเฉลี่ยของขีดเริ่มการได้ยินของคนปกติ (absolute threshold) ที่ความถี่ต่างๆ

2.3.5 ประเภทของเสียง แบ่งได้ 3 ชนิด ดังนี้

- เสียงบริสุทธิ์ (pure tone) คือ เสียงที่มีความถี่คงที่เพียงความถี่เดียว เช่น เสียงจากส้อมเสียง เป็นต้น
- เสียงผสม (complex tone) คือ กลุ่มเสียงที่เกิดจากเสียงบริสุทธิหลายๆความถี่รวมกัน เช่น เสียงดนตรี เสียงพูด เสียงร้องเพลง เป็นต้น
- เสียงรบกวน (noise) คือ เสียงที่ไม่พึงประสงค์ของผู้ฟัง อาจเป็นเสียงบริสุทธิหรือเสียงผสม เช่น เสียงเครื่องจักร เสียงจากการจราจร เป็นต้น เสียงรบกวนเหล่านี้หากรับฟังติดต่อกันนานๆ ทำให้ประสาทหูเสื่อมได้

2.4 คุณสมบัติของเสียง

2.4.1 Frequency, wavelength and propagation speed

คลื่นเสียง (Sound Waves) เกิดจากการเปลี่ยนแปลงความดัน ความถี่ ความเร็วของการแพร่ (propagation) หรือการสั่นสะเทือนในตัวกลางเช่น อากาศ ทำให้เกิดความดันเป็นคลื่นส่งต่อจากแหล่งกำเนิดคลื่นเสียงที่แพร่ออกจากแหล่งเสียงจะมีลักษณะเป็นทรงกลม (spherical wave) เป็นคลื่นสามมิติที่มีจุดกำเนิดเป็นจุด แนวหน้าคลื่นครอบคลุมไปรอบทุกทิศทางเสมือนกับลูกทรงกลมหลายลูกที่มีจุดศูนย์กลางร่วมกัน ความยาวของคลื่นในการแพร่ 1 รอบ มีความสัมพันธ์ดังสมการ [2]

$$\lambda = c / f$$

λ = wavelength (m), c = propagation speed (m/s) และ f = frequency (Hz)

นั่นคือค่าความยาวคลื่นจะลดลงเมื่อความถี่สูงขึ้น ในทางกลับกันความยาวคลื่นจะเพิ่มขึ้นเมื่อความถี่ลดลง

Propagation speed ความเร็วของการแพร่ของเสียงในอากาศขึ้นกับอุณหภูมิดังสมการ [2]

$$C = 20.04[T' + 273.16]^{1/2}$$

C = propagation speed เช่น speed of sound (m/s), T' = air temperature (C^{o})

คลื่นเสียงจะแพร่ที่ความเร็วแตกต่างกันในของแข็งและของเหลว

2.4.2 Sound power and sound pressure

Sound power เป็นกำลังงานที่มาจากอากาศสั่นสะเทือน ซึ่ง sound power level เป็นลักษณะอำนาจในการแผ่พลังงานเสียงออกมาจากแหล่งกำเนิดเสียง โดยมีค่าอ้างอิงที่ยอมรับได้ทั่วโลก คือ 10^{-12} watt และมีสมการ sound power level (L_w) ดังนี้ [2]

$$L_w = 10 \log (W/W_{re}) \quad \text{(dB)}$$

เมื่อ W_{re} เป็น reference power ของ 10^{-12} watt และ W เป็นพลังเสียงที่แผ่ออกมาจากแหล่งกำเนิด

ความดันเสียง (sound pressure) หมายถึง ค่าความดันของคลื่นเสียงที่เปลี่ยนไปจากความดันบรรยากาศปกติ ค่าความดันที่เปลี่ยนแปลงมากที่สุดคือ ค่าแอมปลิจูด หน่วยที่ใช้คือ นิวตันต่อตารางเมตร (N/m^2) หรือปาสคาล การหาระดับความดังเสียงต้องเทียบกับความดันอ้างอิงที่ 20 ไม-โครปาสคาล ซึ่งเป็นความดันเสียงต่ำสุดที่หูคนหนุ่มสาวปกติสามารถได้ยินที่ความถี่ 1,000 Hz และระดับความดันเสียงที่หาได้จากสมการ [2]

$$L_p = 10 \log (P/P_{re})^2 \quad \text{(dB)}$$

P_{re} คือ 20×10^{-6} pascal หรือ 20 micro Pa

เนื่องจากการหาระดับความดันเสียง โดยการวัดความดังเสียงก่อนเป็นสิ่งที่ค่อนข้างยุ่งยาก จึงได้มีการใช้เครื่องมือที่ใช้วัดระดับความดันเสียงโดยตรง เครื่องมือชนิดนี้เรียกว่า มาตรวัดระดับเสียง (Sound level meter) หรือที่เรียกทั่วไปว่าเครื่องวัดเสียง

ความสัมพันธ์ระหว่าง sound power level และ sound pressure level

ถ้าแต่ละแหล่งมีการแผ่ของเสียงเท่ากันทุกๆทิศทาง พลังงานจะแพร่ขยายในบริเวณที่เพิ่มขึ้นตามระยะทางจากแหล่งที่เพิ่มขึ้น พลังงานเสียงที่ได้รับต่อพื้นที่ 1 หน่วยจะเป็นสัดส่วนที่ตรงกันข้ามกันต่อตารางของระยะทางจากแหล่ง ถ้าระยะทางจากแหล่งเพิ่มเป็น 2 เท่า ความหนาแน่นพลังงานจะลดลง 1 ใน 4 ตัวอย่างเช่น การลดลงในระดับของ 6 dB สำหรับทุกๆสองเท่าของระยะทาง sound pressure จะตกลงตามระยะทางจากแหล่ง ดังนั้นแต่ละสองเท่าของระยะทางความดันจะลดลงหนึ่งในสี่ [2]

2.4.3 Propagation and transmission of sound

Sound intensity (ความเข้มของเสียง) หมายถึง กำลังเสียงต่อ 1 หน่วยพื้นที่ ความเข้มเสียงขึ้นกับทิศทางการกระจายตัว และระยะทางของเสียง มีหน่วยเป็นวัตต์ต่อตารางเมตร (W/m^2) ความดังเสียงลดลงถ้าระยะทางระหว่างแหล่งเสียงและผู้รับเพิ่มขึ้น ตัวอย่างเช่นการวัดความดันเสียง (Sound pressure) ของเสียงจราจรบน motorway ที่ระยะทาง 7 เมตร Sound pressure level (SPL) 78 dB และที่ 14 เมตร SPL 75 dB แต่ที่ 28 เมตร SPL 72 dB [2]

2.4.3.1 ความดันเสียงลดลงเนื่องจากระยะทางจากแหล่งกำเนิดเสียงไม่เคลื่อนที่ (Attenuation Due to Distance from a Point Source) แสดงได้ตามสมการดังนี้ [2]

$$I = W/4\pi d^2$$

I = sound intensity

D = distance from a point sound source

W = acoustic power in free field

ดังนั้น sound intensity เป็นสัดส่วนผกผันกับระยะทาง sound intensity หรือ pressure level ที่จุดนี้สามารถแสดงเป็นหน่วยเดซิเบลได้ดังนี้ [2]

$$L = L_w - 11 - 20\log_{10}d \qquad \text{(dB)}$$

Sound power level (L_w) = 10 $\log_{10}$ ($W/10^{-12}$)

2.4.3.2 ความดันเสียงลดลงเนื่องจากระยะทางจากแหล่งเส้นทางคมนาคม (Attenuation Due to Distance from a Line Source) แหล่งเส้นทางคมนาคมที่ไม่มีขอบเขต (infinite line source) จะมีแหล่งเสียงมากมายจากถนนที่มีบ้านเป็นแถว เช่น ยานพาหนะบน motorway แหล่งเสียงจะต่อเนื่องไปตลอดเส้นทางคมนาคม ในขั้นตอนการสุ่ม บางคลื่นเสียงสามารถละเว้นได้ ใน free filed คลื่นเสียงจะแพร่เป็นรูปทรงกระบอก (cylindrical) จากรอบๆ line source เป็นคลื่นสองมิติ มีจุดเริ่มต้นจากแกนทรงกระบอก แนวหน้าคลื่นจะเป็นรูปทรงกระบอกขนาดต่างๆกันที่มีแนวแกนร่วมกัน เมื่อพลังงานเสียงต่อหน่วยความยาวของ line source เป็น W และ sound intensity (I) ที่ระยะทาง d แสดงได้ดังสมการ [2]

$$I = W/2\pi d$$

พลังงานเสียงจะกระจายเหนือผิวหน้าเป็นรูปทรงกระบอก (Cylinder) ด้วยรัศมี d จากสมการ I ผกผันต่อ d เมื่อ sound power level (L_w) ต่อหน่วยความยาว ดังนั้น sound intensity หรือ pressure level แสดงเป็นหน่วย dB ได้ดังสมการ [2]

$$L = L_w - 8 - 10\log_{10}d \qquad \text{(dB)}$$

$$L_w = 10\log_{10} (W/10^{-12})$$

2.4.3.3 ความดังและระดับความดังของเสียง (Loudness and Loudness Level) เสียงอาจจะถูกบรรยายในลักษณะของความดัง (loudness) ซึ่งจะอธิบายถึงความรู้สึกการได้ยินของผู้รับต่อเสียงนั้น ความดังเป็นพารามิเตอร์ที่มีความสัมพันธ์ระหว่างระดับความดันเสียงและความถี่เสียง เสียงบริสุทธิ์ที่มีระดับความดัน 20 dB และความถี่ 1,000 Hz จะฟังชัด ขณะเดียวกันเสียงซึ่งมีระดับความดันเท่ากัน แต่มีความถี่ 100 Hz จะฟังไม่ได้ยินเพราะต่ำกว่าขีดเริ่มของการได้ยิน Loudness Level มีหน่วยเป็นโฟน (Phon) คือ ระดับความดังของเสียงใดๆมีนิยามว่าเป็นค่าตัวเลขเท่ากับระดับความดันเป็นเดซิเบลของเสียง 1,000 Hz เนื่องด้วยความดังของเสียงที่เรารู้สึกนั้นมิได้เป็นสัดส่วนโดยตรงกับระดับความดังของเสียง จึงมีการตั้งหน่วยใหม่ให้ตรงกับความรู้สึกของคน โดยหน่วยเป็นซอน (sone) [2]

บทที่ 3
วิธีการดำเนินงาน

ในบทนี้จะกล่าวถึงขั้นตอนการทำงาน เพื่อให้ได้มาซึ่งโปรแกรมตรวจสอบการได้ยิน โดยมีเอาท์พุตท์ให้ผู้ถูกทดสอบได้ยิน ซึ่งเป็นเสียงที่มีความถี่ (Hz) และระดับความดังของเสียง (dBA) ที่กำหนดได้

3.1 ความประสงค์ (Requirement)

3.1.1 รูปแบบการทำงาน (Function)

- สร้างเสียงที่มีความถี่ (Hz) และระดับความดังของเสียง (dBA) ตามที่ได้กำหนดไว้
- สามารถแยกการทดสอบของหูข้างซ้ายและ หูข้างขวา
- รับอินพุตท์จากผู้ที่ได้รับการทดสอบ เมื่อผู้ทดสอบได้ยินเสียง
- บันทึกผลการทดสอบ เก็บไว้ในรูปแบบไฟล์

3.1.2 ข้อจำกัด (Constraints)

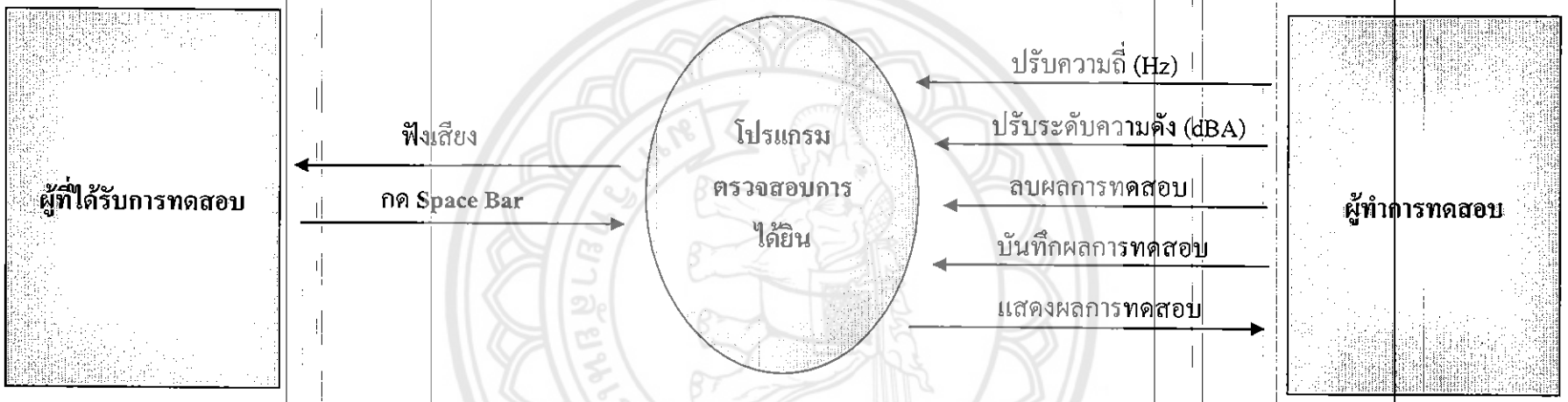
- ผู้ใช้ควรมีความรู้พื้นฐานเกี่ยวกับการทดสอบสมรรถภาพการได้ยิน
- ผู้ใช้ต้องมีความเข้าใจโปรแกรมนี้ก่อนใช้งาน
- ผู้รับการทดสอบจะสามารถได้ยินเสียงผ่านทางหูฟังเท่านั้น
- การบันทึกข้อมูลการทดสอบสามารถบันทึกเป็นไฟล์ .jpg เท่านั้น
- โปรแกรมนี้ทำงานอยู่บนโปรแกรม Matlab 7 ขึ้นไป

3.1.3 รายละเอียด (Specification)

- สร้างเสียงที่มีความถี่ (Hz) และระดับความดังของเสียง (dBA) อย่างถูกต้อง ตามที่ได้กำหนดไว้
- สามารถแยกการทดสอบของหูข้างซ้ายและ หูข้างขวาได้
- รับอินพุตท์จากผู้ที่ได้รับการทดสอบ เมื่อผู้ทดสอบได้ยินเสียงได้
- บันทึกผลการทดสอบ เก็บไว้ในรูปแบบไฟล์ได้

3.2 การออกแบบโปรแกรม

3.2.1 Context Diagram



รูปที่ 3.1 Context Diagram

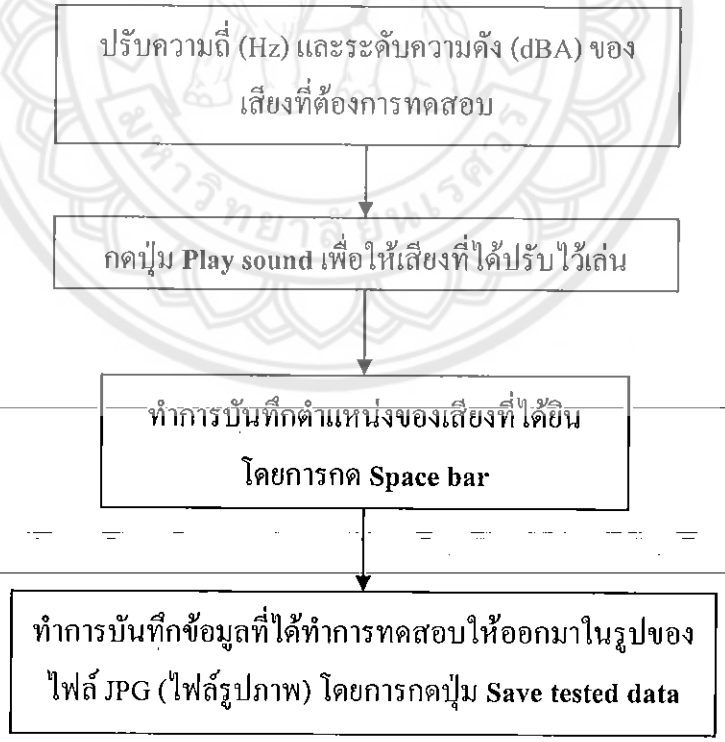
จาก **รูปที่ 3.1** Context Diagram เริ่มต้นทำงานจากตัวโปรแกรมตรวจสอบการได้ยินจะทำการเล่นเสียง ซึ่งผู้ทำการทดสอบได้ทำการตั้งค่าความถี่และ ระดับความดังให้กับตัวโปรแกรมไว้ก่อนแล้ว เมื่อผู้ที่ได้รับการทดสอบได้ยินเสียง จะมีการตอบรับกลับมายังโปรแกรมโดยการกด **Space bar** จากนั้นตัวโปรแกรมจะแสดงผลการตอบรับของผู้ที่ได้รับการทดสอบให้กับผู้ทำการทดสอบ ตัวโปรแกรมสามารถที่จะลบและ บันทึกผลการทดสอบได้

3.2.2 Business Event

ตารางที่ 3.1 Business Event

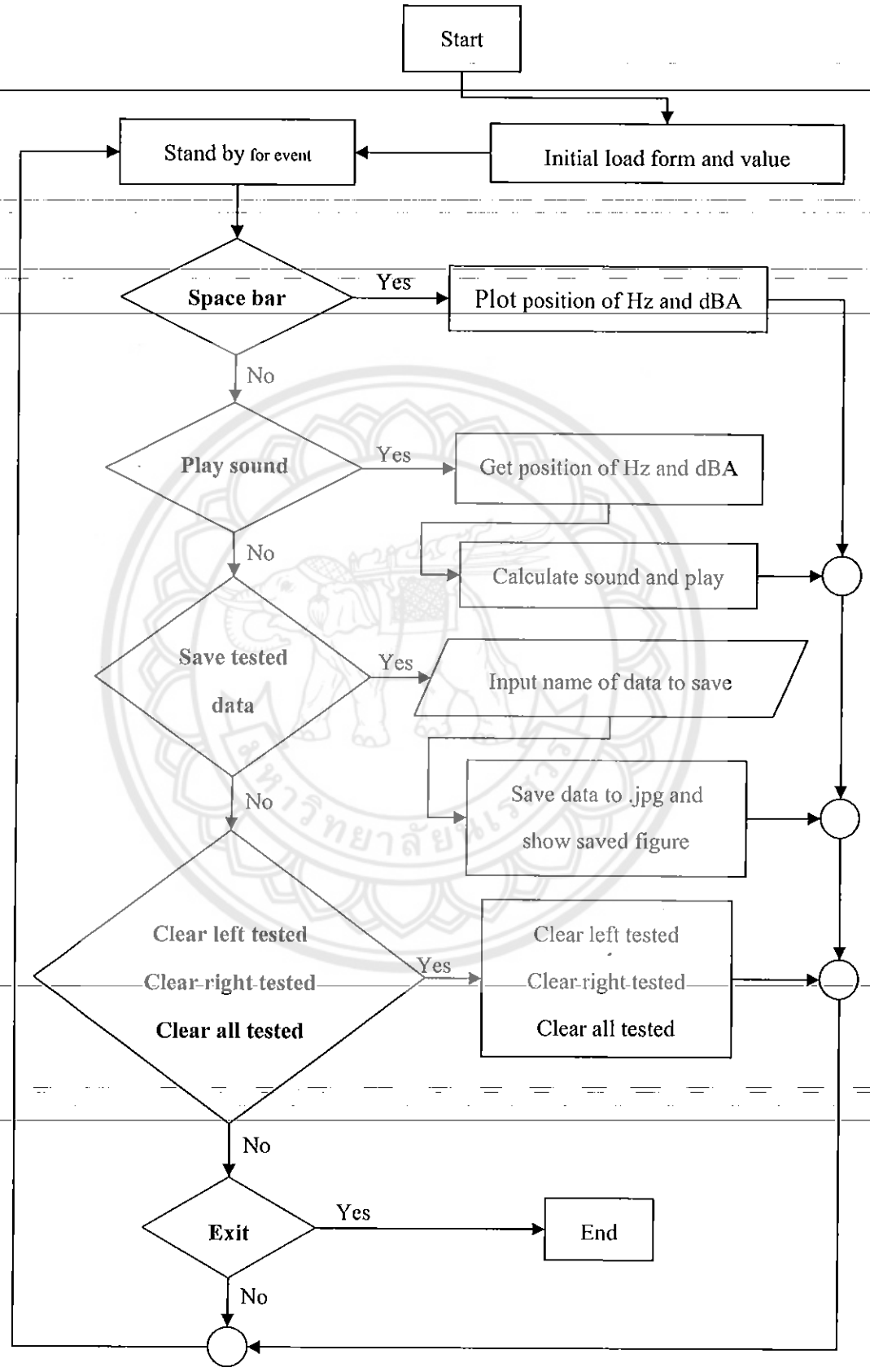
Event	Flow
1. ผู้ที่ได้รับการทดสอบได้ฟังเสียง	เสียง (Out)
2. ผู้ที่ได้รับการทดสอบกด Space Bar	Space Bar (In)
3. ผู้ทำการทดสอบปรับความถี่ (Hz)	ความถี่ (In)
4. ผู้ทำการทดสอบปรับระดับความดัง (dBA)	ระดับความดัง (In)
5. ผู้ทำการทดสอบลบผลการทดสอบ	ผลการทดสอบ (In)
6. ผู้ทำการทดสอบบันทึกผลการทดสอบ	ผลการทดสอบ (In)
7. โปรแกรมตรวจสอบการได้ยินแสดงผลการทดสอบ	ผลการทดสอบ (Out)

3.3 ขั้นตอนการทำงานของโปรแกรมตรวจสอบการได้ยิน



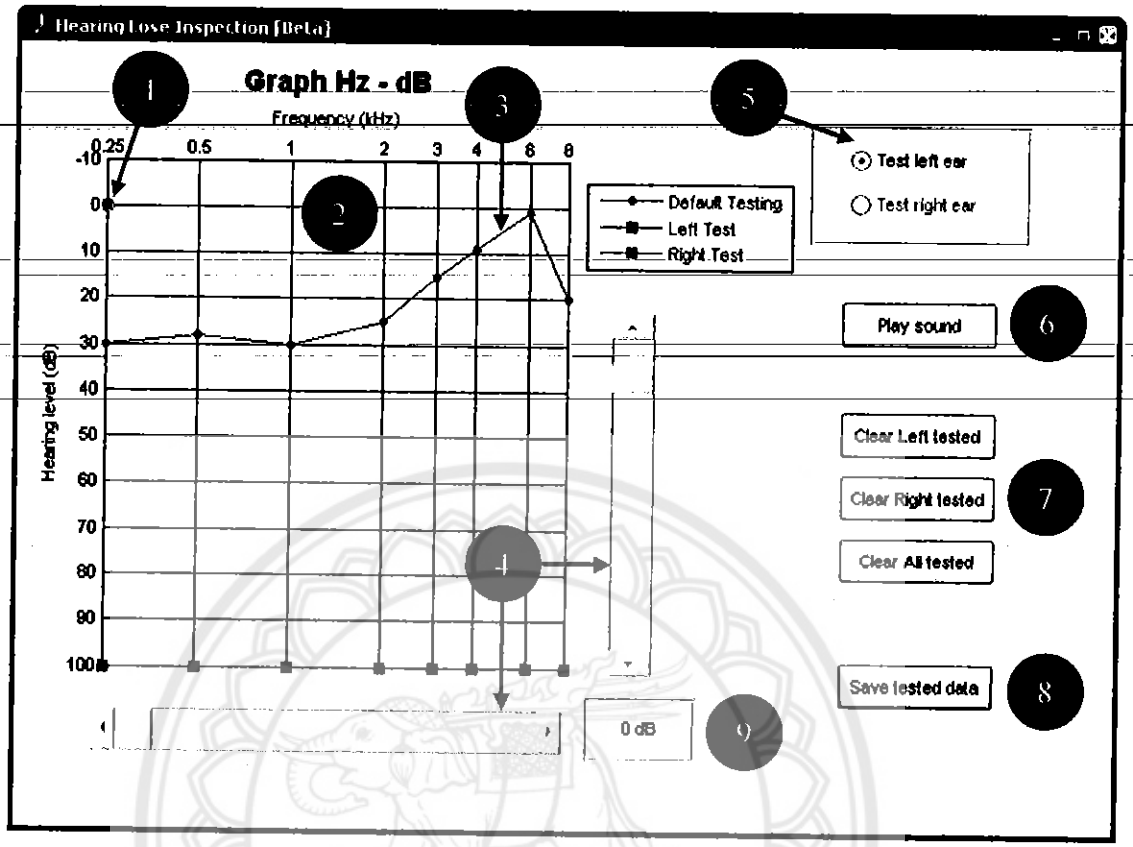
รูปที่ 3.2 ขั้นตอนการทำงานของโปรแกรมตรวจสอบการได้ยิน

3.3.1 Flow chart



รูปที่ 3.3 Flow chart

3.4 รูปร่างและส่วนประกอบของตัวโปรแกรม



รูปที่ 3.4 รูปร่างของโปรแกรมตรวจสอบการได้ยิน

ส่วนประกอบของโปรแกรมตรวจสอบการได้ยิน

หมายเลข 1 หมายถึง ตัวบอกตำแหน่งของ ความถี่ (Hz) และ ระดับความดัง (dBA)

หมายเลข 2 หมายถึง กราฟแสดงความถี่ (Hz) และ ระดับความดัง (dBA)

หมายเลข 3 หมายถึง เส้นแสดงค่าเฉลี่ยของการทดสอบ ที่ทดสอบกับคนที่มีการได้ยินปกติ

หมายเลข 4 หมายถึง ตัวปรับค่าความถี่ (แกนนอน) และตัวปรับค่าระดับความดัง (แกนตั้ง)

หมายเลข 5 หมายถึง ตัวเลือกการทดสอบหูซ้าย และหูขวา

หมายเลข 6 หมายถึง ปุ่มเล่นเสียง

หมายเลข 7 หมายถึง ปุ่มลบค่าที่ได้ทำการบันทึกไว้บนกราฟ

Clear left tested คือ ลบข้อมูลของหูซ้าย

Clear right tested คือ ลบข้อมูลของหูขวา

Clear all tested คือ ลบข้อมูลทั้งหมด

หมายเลข 8 หมายถึง ปุ่มบันทึกข้อมูลออกมาให้รูปไฟล์ JPG (ไฟล์รูปภาพ)

บทที่ 4

ผลการทดลอง

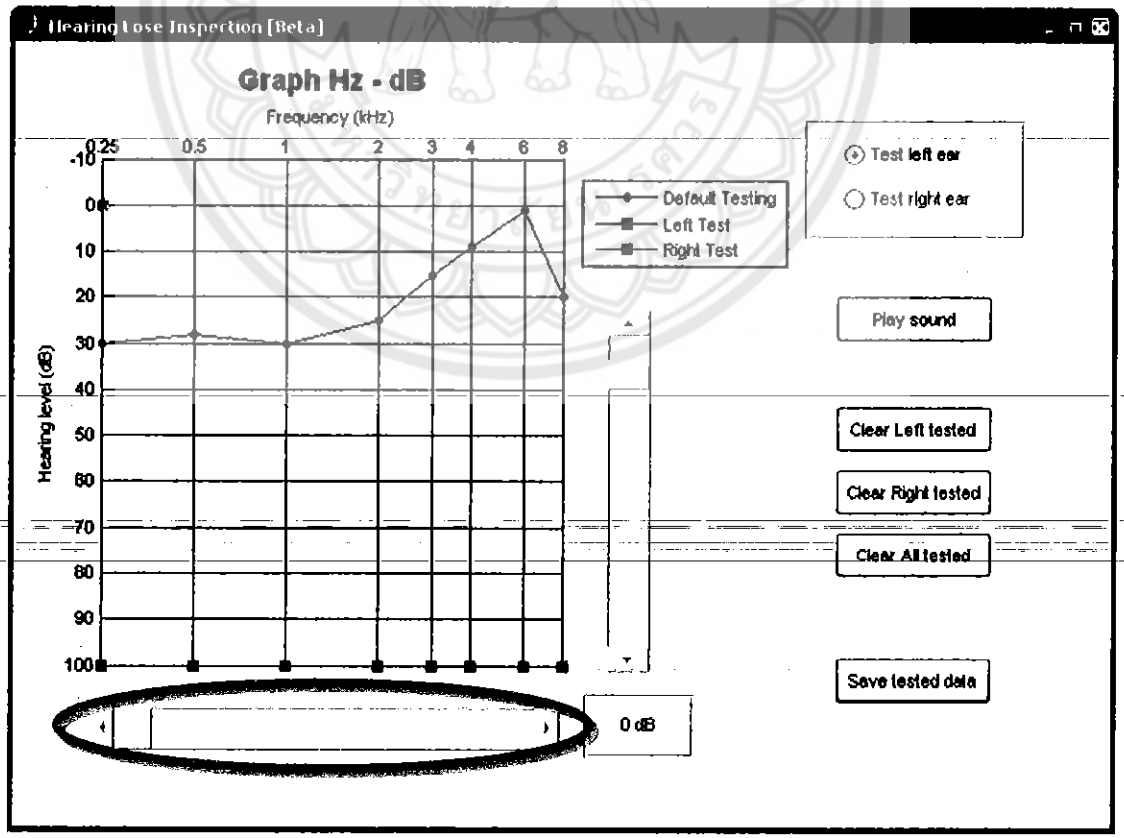
ในบทนี้จะกล่าวถึงการทดสอบโปรแกรมและแสดงผลของการทดสอบโปรแกรม โดยสามารถแบ่งได้ทั้งหมด 3 ขั้นตอน ดังนี้ ขั้นตอนแรกเป็นการทดสอบการเล่นเสียง ขั้นตอนที่สองเป็นการทดสอบการตอบรับการได้ยินเสียงโดยการกด Space Bar และขั้นตอนสุดท้ายเป็นการลบและบันทึกข้อมูลของการทดสอบ

4.1 การทดสอบการเล่นเสียง

การทดสอบนี้จะทดสอบการเล่นเสียงที่มีความถี่ (Hz) และระดับความดังของเสียง (dBA) ตามที่ได้กำหนดได้อย่างถูกต้อง

เปิดโปรแกรม Matlab 7 จากนั้น ทำการตั้ง Path directory ไปยังตำแหน่งของโปรแกรม ตรวจสอบการได้ยิน พิมพ์ "initDisplay" ในช่อง Command line และกด Enter

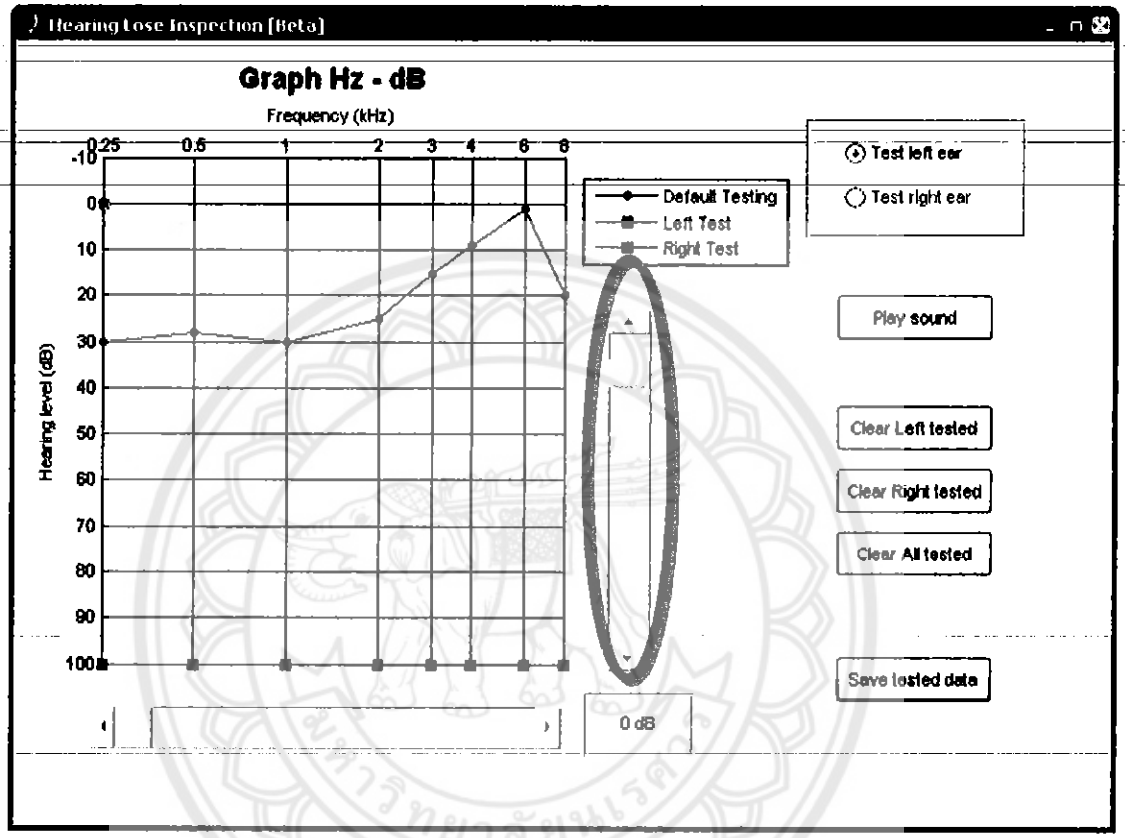
การกำหนดความถี่ (Hz) นั้นจะใช้แทบ Bar ทางด้านล่าง หรือแนวนอนเป็นตัวกำหนด



รูปที่ 4.1 แถบปรับความถี่ (Hz)

เมื่อทำการเลื่อนแถบตัวนี้ ตัวบอกตำแหน่งจะทำการเลื่อนตามแนวนอน ตามระยะที่ได้เลื่อนแถบปรับความถี่ (Hz)

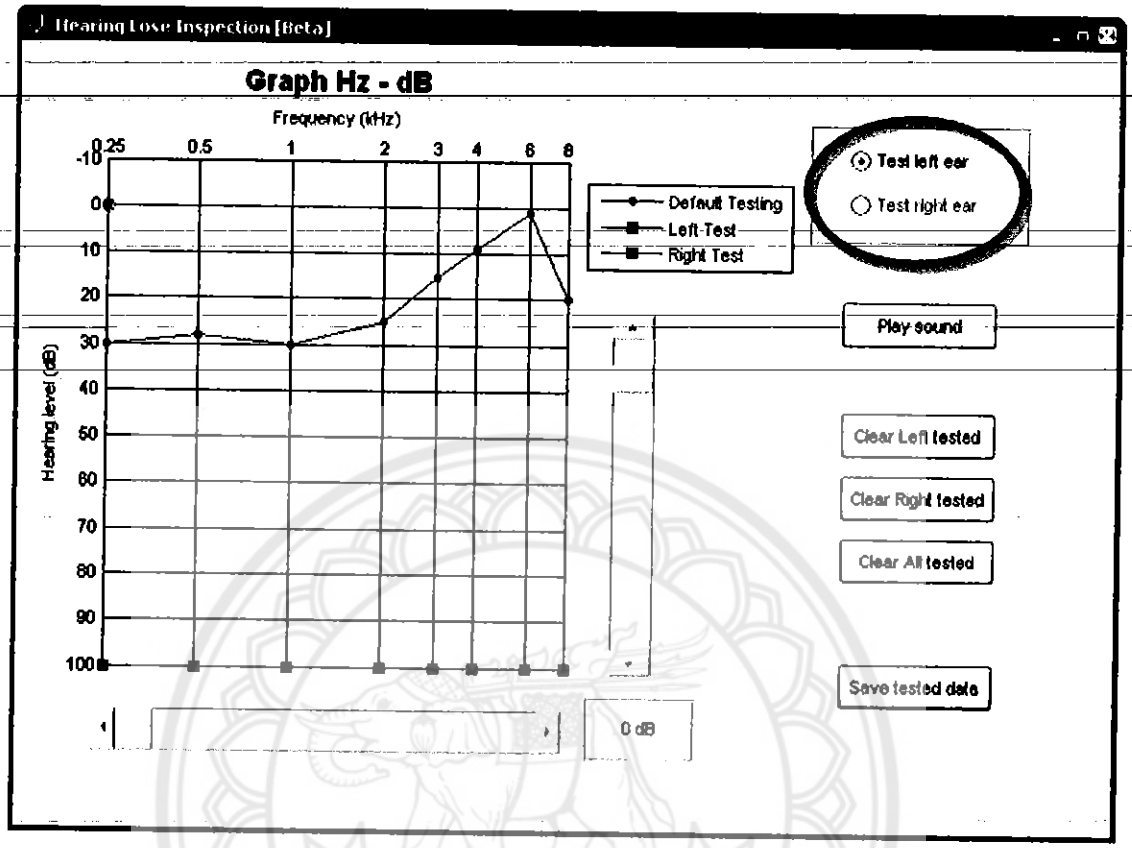
การกำหนดระดับความดังของเสียง (dBA) นั้นจะใช้แถบ Bar ด้านข้างทางขวาของกราฟเป็นตัวกำหนด



รูปที่ 4.2 แถบปรับระดับความดังของเสียง (dB)

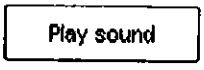
เมื่อทำการเลื่อนแทบตัวนี้ ตัวบอกตำแหน่งจะทำการเลื่อนตามแนวตั้ง ตามระยะที่ได้เลื่อนแถบปรับระดับความดังของเสียง (dBA) และจะมีตัวเลขระบุค่าละเอียดของระดับความดังของเสียงอยู่ข้างล่างแถบปรับตัวนี้

หลังจากนั้นทำการเลือกหูข้างที่ต้องการจะทำการทดสอบ



รูปที่ 4.3 Radio box เลือกหูข้างที่ต้องการทดสอบ

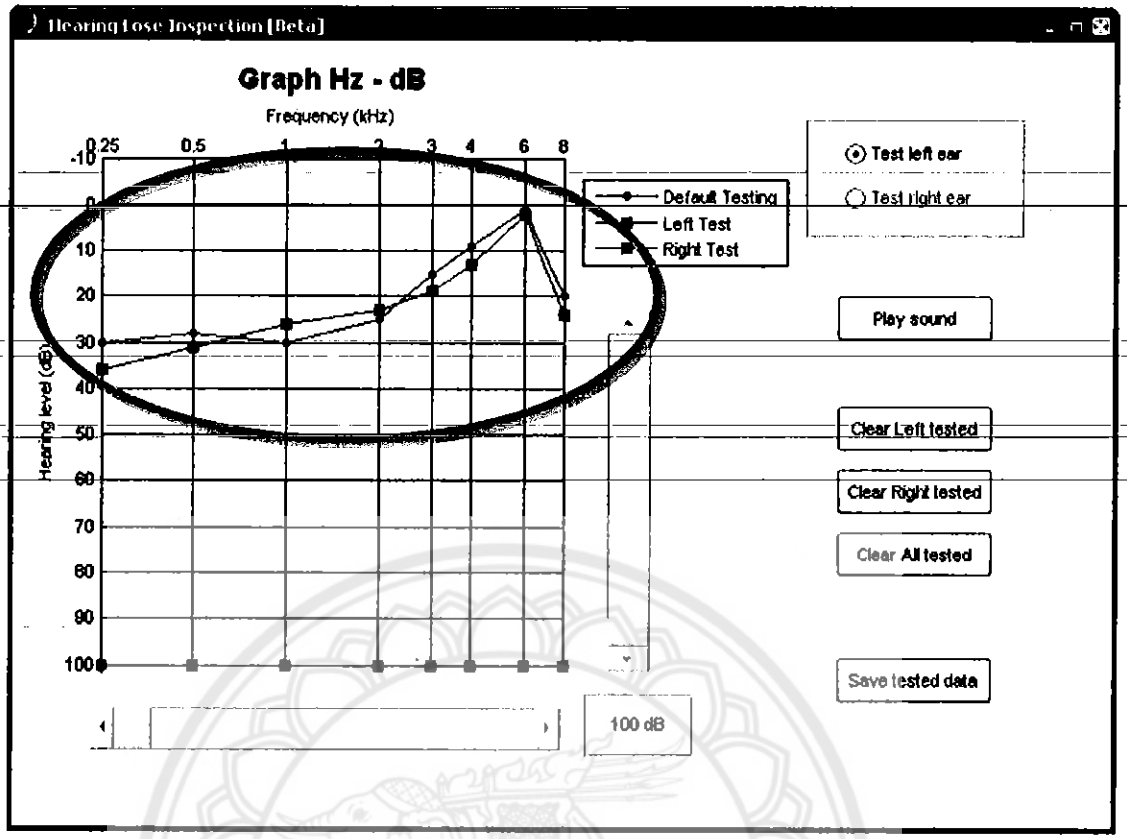
เมื่อทำการปรับความถี่ (Hz) ระดับความดังของเสียง (dBA) และเลือกหูข้างที่ต้องการทดสอบ ได้ตามที่ต้องการจะทดสอบแล้ว การเริ่มทดสอบนั้นจะต้องทำการกดปุ่มที่มีชื่อว่า "Play sound"



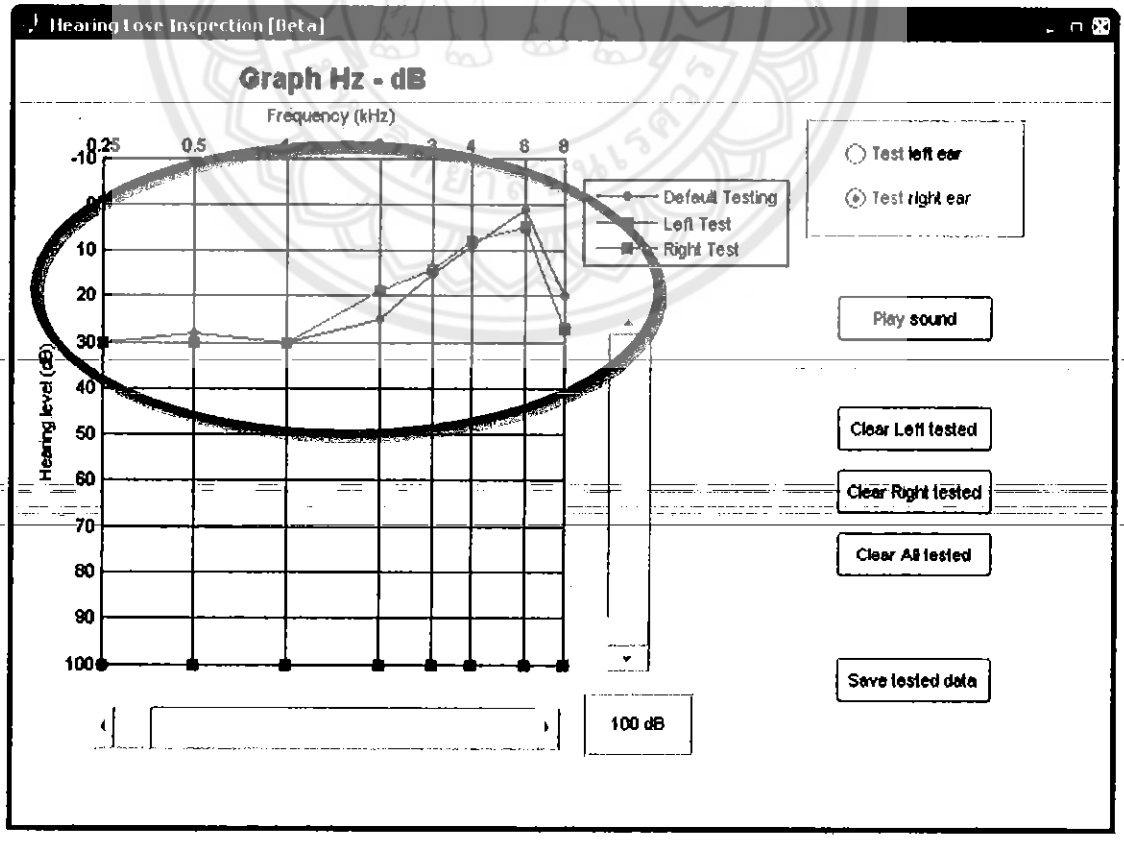
รูปที่ 4.4 ปุ่ม Play sound

4.2 การทดสอบการตอบรับการได้ยินเสียง

การตอบรับการได้ยินเสียงจะมีฟังชันก์ที่ชื่อว่า "getKeyPress" ในการรอรับค่าการกด **Space Bar** ของผู้ที่ได้รับการทดสอบ เมื่อมีการกด **Space Bar** ของผู้ที่ได้รับการทดสอบจะทำให้มีการบันทึกความถี่ (Hz) และระดับความดังของเสียง (dBA) ณ ตำแหน่งนั้นไว้บนกราฟ เส้นสีน้ำเงินจะหมายถึง การทดสอบของหูซ้าย และเส้นสีแดงจะหมายถึง การทดสอบของหูขวา



รูปที่ 4.5 เส้นแสดงการตอบรับการได้ยินของหูซ้าย



รูปที่ 4.6 เส้นแสดงการตอบรับการได้ยินของหูขวา

4.3 การลบและบันทึกข้อมูลของการทดสอบ

4.3.1 การลบข้อมูลการทดสอบ

การลบข้อมูลที่ได้ทำการทดสอบ ทำได้ด้วยกัน 3 ปุ่ม ซึ่งแต่ละปุ่มนั้นจะมีความแตกต่างกัน ดังนี้

รูปที่ 4.7 ปุ่มลบข้อมูลการทดสอบ

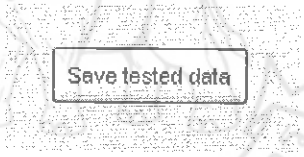
รูปทางซ้ายสุดคือ ปุ่มลบข้อมูลการทดสอบของหูซ้าย

รูปตรงกลางคือ ปุ่มลบข้อมูลการทดสอบของหูขวา

รูปทางขวาสุดคือ ปุ่มลบข้อมูลการทดสอบทั้งหมด

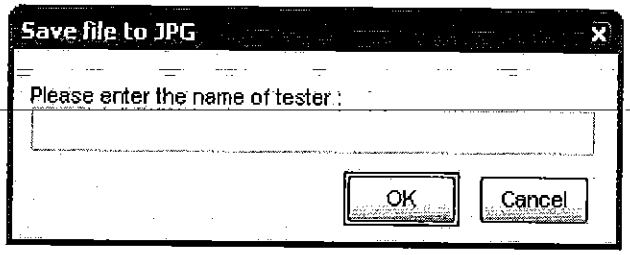
4.3.2 การบันทึกข้อมูลการทดสอบ

การบันทึกข้อมูลที่ได้ทำการทดสอบ ทำได้ด้วยการกดปุ่มชื่อมีชื่อว่า "Save tested data"



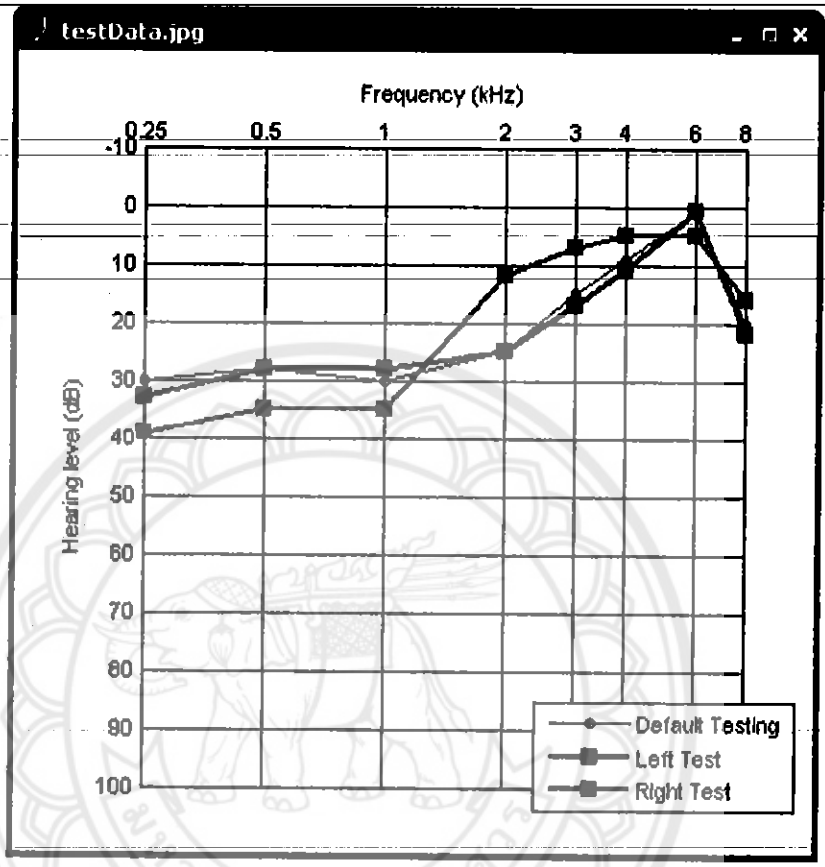
รูปที่ 4.8 ปุ่มบันทึกข้อมูลการทดสอบ

เมื่อทำการกดปุ่ม Save tested data แล้วจะมีกล่องข้อความขึ้นมาให้กรอกชื่อของไฟล์ที่ต้องการจะบันทึกลงไป



รูปที่ 4.9 กล่องข้อความสำหรับกรอกชื่อของไฟล์

หลังจากที่ทำการกรอกชื่อของไฟล์ที่ต้องการบันทึกเสร็จแล้วกดยืนยันการบันทึก โปรแกรมจะแสดงกราฟที่ได้ทำการบันทึก



รูปที่ 4.10 กราฟที่ได้ทำการบันทึก

4.4 ผลการทดสอบการทำงานของโปรแกรม

ตารางที่ 4.1 ผลการทดสอบการทำงานของโปรแกรม

รายการ	โปรแกรมตรวจสอบการได้ยิน
การทดสอบการเล่นเสียง	
ปรับแทบความถี่จาก 0.25 kHz ถึง 8 kHz	✓
ปรับแทบระดับความดังของเสียงจาก -10 dB ถึง 100 dB	✓
เสียงทางหูข้างซ้าย	✓
เสียงทางหูข้างขวา	✓

เสียงความถี่ 0.25 kHz ถึง 8 kHz ในระดับความดังที่ต่ำกว่า 0 dB	✘

ตารางที่ 4.1 (ต่อ)

รายการ	โปรแกรมตรวจสอบการได้ยิน
การทดสอบการเล่นเสียง (ต่อ)	
เสียงความถี่ 0.25 kHz ถึง 8 kHz ในระดับความดัง 0 dB ถึง 60 dB	✓
เสียงความถี่ 0.25 kHz ถึง 8 kHz ในระดับความดังที่มากกว่า 60 dB	✘
การทดสอบการตอบรับการได้ยินเสียง	
เส้นแสดงการได้ยินของหูข้างซ้าย (สีน้ำเงิน)	✓
เส้นแสดงการได้ยินของหูข้างขวา (สีแดง)	✓
การลบและบันทึกข้อมูลของการทดสอบ	
ลบข้อมูลการทดสอบของหูข้างซ้าย (สีน้ำเงิน)	✓
ลบข้อมูลการทดสอบของหูข้างขวา (สีแดง)	✓
ลบข้อมูลการทดสอบของหูทั้ง 2 ข้าง	✓
บันทึกข้อมูลการทดสอบและแสดงผลการบันทึก	✓

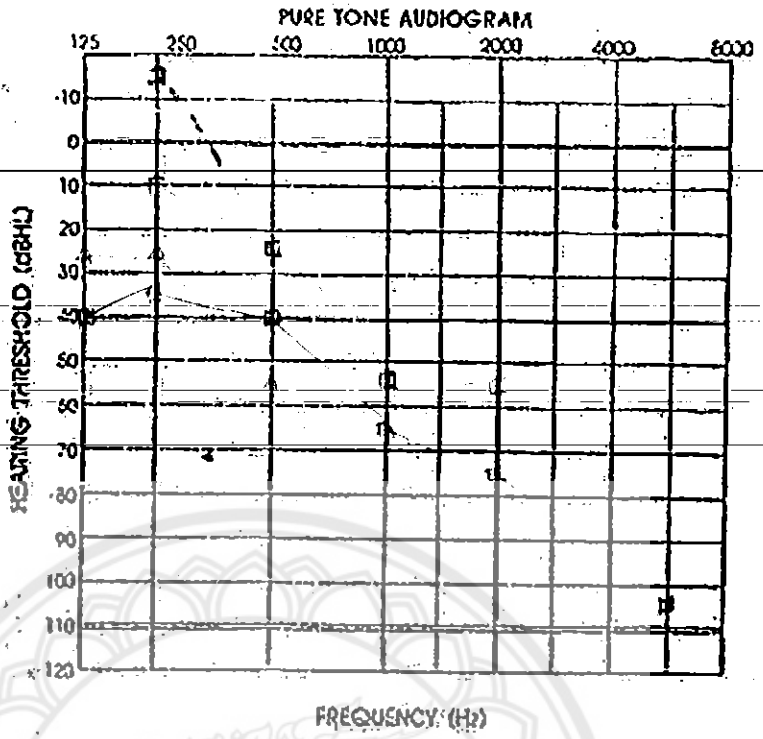
*** **หมายเหตุ :** เครื่องหมาย ✓ หมายถึง ผ่าน และเครื่องหมาย ✘ หมายถึง ไม่ผ่าน

สรุปผลการทดสอบการทำงานของโปรแกรม

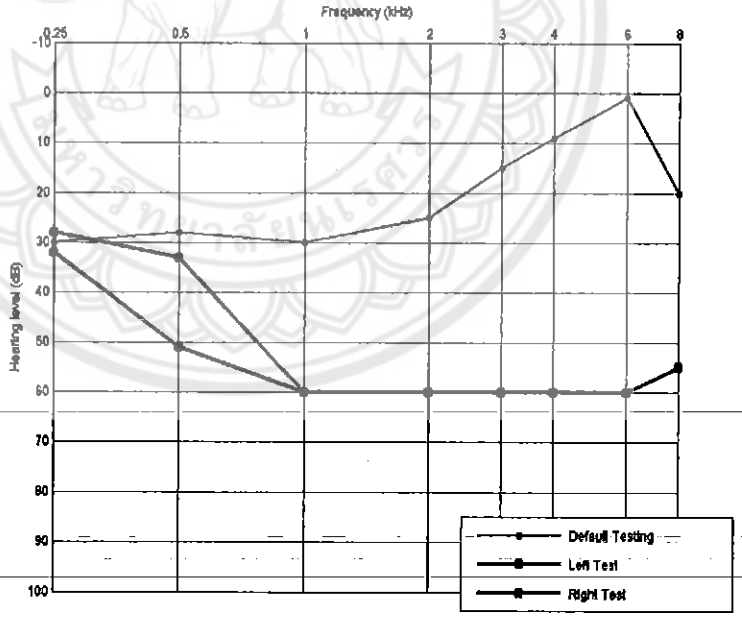
การทดสอบการทำงานของโปรแกรมนั้นได้ทำการแบ่งการทดสอบออกเป็น 3 หัวข้อใหญ่ ซึ่งแต่ละการทดสอบได้ผลออกมาเป็นที่น่าพึงพอใจ ยกเว้นแต่การทดสอบการเล่นเสียงที่มีความดังในระดับน้อยกว่า 0 dBA และที่มีความดังมากกว่า 60 dBA ไม่สามารถทำการเล่นได้ตามที่ได้ระบุไว้ตาม Requirement ดังที่ได้ทำการออกแบบโปรแกรมไว้

4.5 การเปรียบเทียบประสิทธิภาพระหว่างเครื่องตรวจการได้ยินกับโปรแกรมต้นแบบการตรวจสอบการได้ยิน

ทำการเปรียบเทียบระหว่างผลการทดสอบของเครื่องตรวจการได้ยินกับโปรแกรมต้นแบบการตรวจสอบการได้ยิน โดยใช้ผู้รับการทดสอบคนเดียวกันซึ่งได้ผลออกมาดังนี้



รูปที่ 4.11 กราฟจากเครื่องตรวจสอบการได้ยิน



รูปที่ 4.12 กราฟจากโปรแกรมตรวจสอบการได้ยิน

สรุปผลการเปรียบเทียบ

กราฟมีแนวโน้มสอดคล้องในช่วงความถี่ต่ำ แต่เนื่องจากมีข้อจำกัดของคอมพิวเตอร์ฮาร์ดแวร์ ทำให้โปรแกรมที่สร้างไม่สามารถปรับให้ได้มาตรฐาน กราฟที่ได้จึงมีความคลาดเคลื่อน

บทที่ 5
บทสรุป

5.1 วิเคราะห์ผลการทดลอง

จากผลการทดลองในบทที่ 4 ได้ทำการทดสอบการทำงานของโปรแกรมด้วยกัน 3 ขั้นตอน ดังนี้ ขั้นตอนแรกเป็นการทดสอบการเล่นเสียง ขั้นตอนที่สองเป็นการทดสอบการตอบรับการได้ยินเสียงโดยการกด **Space Bar** และขั้นตอนสุดท้ายเป็นการลบและบันทึกข้อมูลของการทดสอบ ซึ่งได้ผลสรุปตาม Requirement ดังนี้

ตารางที่ 5.1 สรุปผลการทดสอบการทำงานของโปรแกรมตาม Requirement

รายการ	โปรแกรมตรวจสอบการได้ยิน
สร้างเสียงที่มีความถี่ (Hz) และระดับความดังของเสียง (dB) อย่างถูกต้องตามที่ได้กำหนดไว้	✗
สามารถแยกการทดสอบของหูข้างซ้ายและ หูข้างขวาได้	✓
รับอินพุตท์จากผู้ที่ได้รับการทดสอบ เมื่อผู้ทดสอบได้ยินเสียงได้	✓
บันทึกผลการทดสอบ เก็บไว้ในรูปแบบไฟล์ได้	✓

*** **หมาย**เหตุ : เครื่องหมาย ✓ หมายถึง ผ่าน และเครื่องหมาย ✗ หมายถึง ไม่ผ่าน

จากตารางที่ 5.1 สรุปผลการทดสอบการทำงานของโปรแกรมตาม Requirement นั้น รายการ สร้างเสียงที่มีความถี่ (Hz) และระดับความดังของเสียง (dB) อย่างถูกต้อง ตามที่ได้กำหนดไว้ นั้นสามารถทำการออกแบบให้โปรแกรมทำงานได้อย่างถูกต้อง แต่มีข้อผิดพลาดในช่วงมากกว่า 60 dB

เนื่องจากข้อจำกัดของอุปกรณ์ที่ใช้เล่นเสียง ที่ไม่สามารถผลิตเสียงที่มีความดังมากกว่า 60 dB ได้ ทำให้ไม่สามารถสร้างเสียงที่มีความถี่ (Hz) และระดับความดังของเสียง (dB) ตามที่กำหนดไว้ได้อย่างถูกต้องทั้งหมด

5.2 ปัญหาและแนวทางแก้ไข

ในระหว่างการทำโครงงานนี้ได้ ประสบปัญหาทางด้านการเขียนโปรแกรมเป็นส่วนใหญ่ ได้แก่

1. ปัญหาในการสร้าง figure โดยใช้ตัวช่วยอัตโนมัติของโปรแกรม Matlab ไม่สามารถปรับแต่งเพื่อมาใช้งานกับโครงงานนี้ได้ แก้ด้วยการหาตัวอย่างโครงงานที่ทำการสร้าง figure เอง มาศึกษาและนำมาประยุกต์ใช้
2. ปัญหาในการเรียกค่าของตัวแปรข้าม Function ไม่ได้ แก้ด้วยการกำหนดให้เป็นค่าตัวแปร Global โดยการกำหนดใช้ "set(gcf, 'UserData', figData);" และการเรียกค่าตัวแปรมาใช้ "figData = get(gcf, 'UserData');"

5.3 สรุปผลการทดลอง

โปรแกรมตรวจสอบการได้ยินนี้ เป็นเพียงต้นแบบของเครื่องตรวจสอบการได้ยินทางการแพทย์ สามารถใช้ในการตรวจการได้ยินและแสดงผลออกมาในรูปแบบกราฟ รวมทั้งยังสามารถจัดเก็บผลการตรวจไว้ในรูปแบบไฟล์ได้ แต่ยังมีจุดที่ต้องนำไปพัฒนาต่อคือ ในการทำงานของโปรแกรม ไม่สามารถผลิตเสียงที่มีระดับความดังของเสียงที่สูงกว่า 60 dB

5.4 ข้อเสนอแนะ

1. โปรแกรมนี้สามารถทำการตรวจสอบ การได้ยินที่อยู่ช่วงระดับความดังของเสียงตั้งแต่ 0 dB ถึง 60 dB เท่านั้น
2. ในการพัฒนาต่อไป อาจจะเพิ่มการ calibrate หูฟังเพื่อความสะดวกในกรณีที่มีการเปลี่ยนอุปกรณ์หูฟัง เพื่อความแม่นยำในของผลการตรวจ
3. ในการพัฒนาต่อไป อาจจะพัฒนาการตรวจสอบการได้ยินแบบอัตโนมัติเพื่อความแม่นยำและรวดเร็วยิ่งขึ้น
4. สามารถนำไปพัฒนาสร้างขึ้นเป็น Embedded Machine ต่อไปได้

เอกสารอ้างอิง

[1] รศ.ดร.มนัส สังวรศิลป์, วรรัตน์ ภัทรอมรกุล, 2543. **คู่มือโปรแกรม MATLAB ฉบับสมบูรณ์** กรุงเทพฯ: สำนักพิมพ์ดวงกมล

[2] ธิติยา รักษ์ศรี. **"การติดตามสมรรถภาพการได้ยินและสภาพเสียงดังจากการทำงานในคนงานโรงพยาบาลสงขลานครินทร์ เปรียบเทียบย้อนหลัง 3 ปี"** วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาอนามัยสิ่งแวดล้อม มหาวิทยาลัยสงขลานครินทร์. 2545

[3] The MathWork, 1999. Signal Processing Toolbox User's Guide. The MathWork Inc.

[4] i-medipro.com. **"การตรวจสมรรถภาพการได้ยิน"**. [Online]. Available: http://www.i-medipro.com/index.php?lay=show&ac=article&Id=70931&Ntype=4

ภาคผนวก ก.

Code โปรแกรมตรวจสอบการได้ยิน

1. initDisplay.m

```
clc;
clear all;
close all;

scrSize = get(0,'ScreenSize');

% Set figure window size [width height].
figureSize = [700 500];

% Create figure window.
figureWindow = figure('Name', 'Hearing Lose Inspection [Beta]', ...
                      'NumberTitle', 'off', 'MenuBar', 'none', ...
                      'Position', ...
                      [(scrSize(3)-figureSize(1))/2 (scrSize(4)-figureSize(2))/2 figureSize(1)
figureSize(2)], ...
                      'Resize', 'off', 'KeyPressFcn', @getKeyPress);
clf;

% Create Axes graph.
axesGraph = axes('Parent', figureWindow, 'Position', [.08 .2 .42 .65], ...
                 'YDir', 'reverse', 'XAxisLocation', 'top' , 'NextPlot', 'add', ...
                 'FontSize', 8, 'Box', 'on', 'XTick', [0.25 0.5 1 2 3 4 6 8], ...
                 'YTick' , [-10 0 10 20 30 40 50 60 70 80 90 100], 'XScale', 'log', ...
                 'GridLineStyle', '-');
title('\bfGraph Hz - dB', 'Color', 'Blue', 'FontSize', 14);
grid on;
```

```
xlabel('Frequency (kHz)', 'Color', 'Red');
ylabel('Hearing level (dB)', 'Color', 'Red');
axis([0.25 8 -10 100]);

% Create radio button group.
buttonGroup = uibuttongroup('Position', [.72 .75 .2 .15]);
radioButton1 = uicontrol('Parent', buttonGroup, 'Style', 'radiobutton', ...
                'String', 'Test left ear', 'Position', [25 40 100 30], ...
                'ForegroundColor', 'Blue');
radioButton2 = uicontrol('Parent', buttonGroup, 'Style', 'radiobutton', ...
                'String', 'Test right ear', 'Position', [25 12 100 30], ...
                'ForegroundColor', 'Red');

% Create button.
playButton = uicontrol('String', 'Play sound', 'Style', 'pushbutton', ...
                'Position', [525 310 100 30], 'Callback', @playTestSound);
clearLeftTested = uicontrol('String', 'Clear Left tested', 'Style', 'pushbutton', ...
                'Position', [525 240 100 30], 'Callback', @clearTested);
clearRightTested = uicontrol('String', 'Clear Right tested', 'Style', 'pushbutton', ...
                'Position', [525 200 100 30], 'Callback', @clearTested);
clearAllTested = uicontrol('String', 'Clear All tested', 'Style', 'pushbutton', ...
                'Position', [525 160 100 30], 'Callback', @clearTested);
saveButton = uicontrol('String', 'Save tested data', 'Style', 'pushbutton', ...
                'Position', [525 80 100 30], 'Callback', @saveToJPG);

% Create slider.
sliderButtonX = uicontrol('Style', 'slider', 'Position', [50 50 300 27], ...
                'Min', 1, 'Max', 8, 'SliderStep', [0.143 0.1], ...
                'Value', 1, 'Callback', @update_HzText);
sliderButtonY = uicontrol('Style', 'slider', 'Position', [380 100 27 230], ...
```

```
                    'Min', -10, 'Max', 100, 'SliderStep', [0.0092 0.1], ...
                    'Value', 90, 'Callback', @update_dBText);

% Create text of dB.
buttonTextGroup = uibuttongroup('Position', [.52 .09 .1 .08]);
dBTextButton = uicontrol('Parent', buttonTextGroup, 'Style', 'text', ...
                    'Position', [10 8 50 20], 'String', '0 dB', ...
                    'HorizontalAlignment', 'center');

% Plot default testing.
left_dBTest = [100 100 100 100 100 100 100 100];
right_dBTest = [100 100 100 100 100 100 100 100];

dtX = [0.25 0.5 1 2 3 4 6 8];
dtY = [30 28 30 25 15 9 1 20];
defaultTestingPlot = plot(axesGraph, dtX, dtY, 'Color', 'black', ...
                    'Marker', 'o', 'MarkerFaceColor', 'black', 'MarkerSize', 2);
leftPlot = plot(axesGraph, dtX, left_dBTest, 'Color', 'blue', ...
                    'Marker', 's', 'MarkerFaceColor', ...
                    'blue', 'MarkerSize', 5, 'Tag', 'leftPlot');
rightPlot = plot(axesGraph, dtX, right_dBTest, 'Color', 'red', ...
                    'Marker', 's', 'MarkerFaceColor', ...
                    'red', 'MarkerSize', 5, 'Tag', 'rightPlot');

% Legend line.
legend(axesGraph, 'Default Testing', 'Left Test', 'Right Test', ...
    'Location', [0.56 0.72 0.1 0.1]);

% Default plot user select point.
scatter(axesGraph, 0.25, 0, 'Marker', 'o', 'Tag', 'SelectByUser', ...
    'MarkerEdgeColor', 'magenta', 'MarkerFaceColor', 'magenta');
```

```
)

        % For check key press input.
        bCheckKey = 0;

        % Store global variables in figure data field.
        % Note: This is used to pass variables between functions.
        figData.axesGraph = axesGraph;
        figData.radioButton1 = radioButton1;
        figData.radioButton2 = radioButton2;
        figData.playButton = playButton;
        figData.clearLeftTested = clearLeftTested;
)       figData.clearRightTested = clearRightTested;
        figData.clearAllTested = clearAllTested;
        figData.saveButton = saveButton;
        figData.sliderButtonX = sliderButtonX;
        figData.sliderButtonY = sliderButtonY;
        figData.dBTextButton = dBTextButton;
        figData.left_dBTest = left_dBTest;
        figData.right_dBTest = right_dBTest;
        figData.bCheckKey = bCheckKey;
)       figData.valueX = 0.25;
        figData.valueY = 0;

        % Set figure data to User data.
        set(gcf, 'UserData', figData);
```

2. update_HzText.m

```
function update_HzText(obj, event)

% Get figure data from initDisplay.
figData = get(gcf, 'UserData');

str1 = get(figData.sliderButtonY, 'Value');

% Find user select point and delete.
delete(findobj('Tag', 'SelectByUser'));

% Estimate value of sliderButtonY.
if (get(figData.sliderButtonY, 'Value') < 0)
   figData.valueY = 90 - round(str1);
else
   figData.valueY = 90 - floor(str1);
end

% Map value of Hz.
switch (round(get(figData.sliderButtonX, 'Value')))
   case 1
      figData.valueX = .25;
   case 2
      figData.valueX = .5;
   case 3
      figData.valueX = 1;
   case 4
      figData.valueX = 2;
   case 5
      figData.valueX = 3;
   case 6
```

```
            figData.valueX = 4;
        case 7
            figData.valueX = 6;
        case 8
            figData.valueX = 8;
    end

    %Plot user select point.
    scatter(figData.axesGraph, figData.valueX, figData.valueY, 'Marker', 'o', 'Tag',
    'SelectByUser', ...
        'MarkerEdgeColor', 'magenta', 'MarkerFaceColor', 'magenta');

    % Set figure data to User data.
    set(gcf, 'UserData', figData);
```

3. update_dBText.m

```
function update_dBText(obj, event)

% Get figure data from initDisplay.
figData = get(gcf, 'UserData');

str1 = get(figData.sliderButtonY, 'Value');
str2 = ' dB';

% Find user select point and delete.
delete(findobj('Tag', 'SelectByUser'));

% Estimate value of sliderButtonY.
if (get(figData.sliderButtonY, 'Value') < 0)
   figData.valueY = 90 - round(str1);
else
   figData.valueY = 90 - floor(str1);
end

% Map value of Hz.
switch (round(get(figData.sliderButtonX, 'Value')))
  case 1
     figData.valueX = .25;
  case 2
     figData.valueX = .5;
  case 3
     figData.valueX = 1;
  case 4
     figData.valueX = 2;
  case 5
     figData.valueX = 3;
```

```
        case 6
            figData.valueX = 4;
        case 7
            figData.valueX = 6;
        case 8
            figData.valueX = 8;
    end

    %Plot user select point.
    scatter(figData.axesGraph, figData.valueX, figData.valueY, 'Marker', 'o', 'Tag',
    'SelectByUser', ...
        'MarkerEdgeColor', 'magenta', 'MarkerFaceColor', 'magenta');

    % Set dB text display.
    set(figData.dBTextButton, 'String', strcat(num2str(figData.valueY), str2));

    % Set figure data to User data.
    set(gcf, 'UserData', figData);
```

4. saveToJPG.m

```
function saveToJPG(obj, event)

% Get figure data from initDisplay.
figData = get(gcf, 'UserData');

%Ask the name of file.
ansFromUser = inputdlg('Please enter the name of tester :', 'Save file to JPG');

% Check input the name of file.
if (isequal(ansFromUser, {''}))
   warndlg('The file must be name.', 'Warning !!!');
else
   % Create figure window.
   scrSize = get(0,'ScreenSize');
   figureSave = figure('Name', strcat(ansFromUser{1,1}, '.jpg'), 'NumberTitle', 'off', ...
                'MenuBar', 'none', 'Resize', 'off', ...
                'Position', ...
                [(scrSize(3)-400)/2 (scrSize(4)-400)/2 400 400]);

   % Create Axes graph.
   axesSave = axes('Parent', figureSave, 'Position', [.16 .08 .75 .8], ...
                'YDir', 'reverse', 'XAxisLocation', 'top' , 'NextPlot', 'add', ...
                'FontSize', 8, 'Box', 'on', 'XTick', [0.25 0.5 1 2 3 4 6 8], ...
                'YTick' , [-10 0 10 20 30 40 50 60 70 80 90 100], 'XScale', 'log', ...
                'GridLineStyle', '-');
   grid on;
   xlabel('Frequency (kHz)', 'Color', 'Red');
   ylabel('Hearing level (dB)', 'Color', 'Red');
   axis([0.25 8 -10 100]);
```

```
% Plot default testing and results of testing.
dtX = [0.25 0.5 1 2 3 4 6 8];
dtY = [30 28 30 25 15 9 1 20];
plot(axesSave, dtX, dtY, 'Color', 'black' , 'LineWidth', 1, ...
    'Marker', 'o', 'MarkerFaceColor', 'black', 'MarkerSize', 4);
plot(axesSave, dtX, figData.left_dBTest, 'Color', 'blue', 'LineWidth', 2, ...
    'Marker', 's', 'MarkerFaceColor' , ...
    'blue', 'MarkerSize', 5, 'Tag', 'leftPlot');
plot(axesSave, dtX, figData.right_dBTest, 'Color', 'red', 'LineWidth', 2, ...
    'Marker', 's', 'MarkerFaceColor' , ...
    'red', 'MarkerSize', 5, 'Tag', 'rightPlot');
% Legend line.
legend(axesSave, 'Default Testing', 'Left Test', 'Right Test', ...
     'Location', [.65 .054 .325 .137]);

saveas(figureSave, ansFromUser{1,1}, 'tif');
end
```

5. playTestSound.m

```
function playTestSound(obj, event)

% Get figure data from initDisplay.
figData = get(gcf, 'UserData');

% Typical rates: {8.000, 11.025, 22.050, 44.100} kHz.
Fs = 44.1e3;

% Set test tone durations (in seconds).
dT = 2;

% Determine sample times.
t = (1 / Fs) * [1:round(dT * Fs)];

% Generate 1000 Hz test tone.
fc = (figData.valueX * 1000);

% Level of dB.
if (figData.valueX == 0.25 && figData.valueY > 0 && figData.valueY <= 50)
   x250 = [0.3 0.6 0.8 1 1.2 1.5 1.8 2 2.3 2.6 2.9 3.3 3.7 4.3 4.8 5.4 6.1 6.7 7.4 8.6 9.7
10.9 12 13.2 14.6 16.4 19 21.5 23.8 26.4 30 33 38 43 48 54 60 66 76 88 98 110 120 135
148 165 190 220 260 320];
   dbA = x250(figData.valueY)
elseif (figData.valueX == 0.5 && figData.valueY > 0 && figData.valueY <= 55)
   x500 = [0.1 0.2 0.3 0.3 0.4 0.4 0.5 0.6 0.7 0.8 0.9 1 1.2 1.3 1.5 1.7 1.9 2.1 2.3 2.6 2.9
3.3 3.7 4.1 4.6 5.1 5.8 6.5 7.4 8.5 9.5 10.5 11.6 13 14.6 16.3 18.4 20.5 23 26.2 29 32.7
36.5 41.2 46.2 52 58.2 65.2 73 82 92 104 126 158 210];
   dbA = x500(figData.valueY)
elseif (figData.valueX == 1 && figData.valueY > 0 && figData.valueY <= 60)
```

```
    x1k = [0.1 0.1 0.1 0.1 0.2 0.2 0.2 0.3 0.3 0.4 0.4 0.5 0.6 0.6 0.7 0.8 0.9 1 1.1 1.3 1.4
1.6 1.8 2 2.2 2.5 2.8 3.1 3.5 3.9 4.4 4.9 5.5 6.2 6.9 7.8 8.7 9.7 10.9 12.3 13.8 15.4 17.3
19.4 21.7 24.6 27.6 30.8 34.5 39 43.6 48.8 54.7 61.5 69 77 87 98 115 170];
    dbA = x1k(figData.valueY)
elseif (figData.valueX == 2 && figData.valueY > 0 && figData.valueY <= 60)
    x2k = [0 0.1 0.1 0.1 0.1 0.2 0.2 0.2 0.3 0.3 0.3 0.4 0.4 0.5 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2
1.4 1.5 1.7 1.9 2.1 2.4 2.7 3 3.3 3.7 4.2 4.7 5.2 5.9 6.6 7.4 8.3 9.3 10.4 11.7 13.2 14.8
16.5 18.6 20.8 23.3 26.2 29.2 33 37 41.5 46.5 52.2 58.5 66 74 83 93];
    dbA = x2k(figData.valueY)
elseif (figData.valueX == 3 && figData.valueY > 0 && figData.valueY <= 60)
    x3k = [0 0.1 0.1 0.1 0.1 0.2 0.2 0.2 0.3 0.3 0.3 0.4 0.4 0.5 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2
1.3 1.5 1.7 1.9 2.1 2.4 2.7 3 3.3 3.7 4.2 4.7 5.2 5.8 6.5 7.3 8.2 9.2 10.3 11.5 13 14.7 16.4
18.4 20.5 23 26 29 32.5 36.8 41 46 51.5 58.5 65 73 82 92];
    dbA = x3k(figData.valueY)
elseif (figData.valueX == 4 && figData.valueY > 0 && figData.valueY <= 60)
    x4k = [0 0.1 0.1 0.1 0.1 0.2 0.2 0.2 0.3 0.3 0.4 0.4 0.5 0.5 0.6 0.6 0.7 0.8 0.9 1 1.1 1.3
1.4 1.6 1.8 2 2.2 2.5 2.8 3.1 3.5 3.9 4.4 4.9 5.5 6.2 6.9 7.8 8.7 9.8 10.9 12.3 13.9 15.5
17.4 19.5 21.8 24.4 27.4 30.8 34.5 38.7 43.5 49 55 61.5 69 77 87 98];
    dbA = x4k(figData.valueY)
elseif (figData.valueX == 6 && figData.valueY > 0 && figData.valueY <= 60)
    x6k = [0 0.1 0.1 0.1 0.2 0.2 0.2 0.3 0.3 0.4 0.4 0.5 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.4 1.6
1.8 2 2.2 2.4 2.7 3.1 3.4 3.8 4.3 4.8 5.4 6 6.7 7.6 8.5 9.5 10.8 12 13.6 15.2 17 19.2 21.5
24 27.1 30.2 34 38.2 42.5 48 53.5 60.5 68 76 85 96 112 180];
    dbA = x6k(figData.valueY)
elseif (figData.valueX == 8 && figData.valueY > 0 && figData.valueY <= 55)
    x8k = [0.1 0.1 0.1 0.2 0.2 0.3 0.3 0.4 0.4 0.5 0.6 0.6 0.7 0.8 0.9 1 1.2 1.3 1.5 1.6 1.8 2
2.3 2.5 2.8 3.2 3.6 4 4.5 5 5.6 6.3 7.1 8 8.9 10 11.3 12.6 14.1 15.9 17.8 20 22.5 25 28.3
31.5 35.5 40 44.5 50 56.5 63 71 79.5 89];
    dbA = x8k(figData.valueY)
else
    dbA = 0
```

```
end
A = dbA/100;

y = sin(2 * pi * fc * t);

% Size of sound.
[m n] = size(y);

% Check radio button for play left or right.
if (get(figData.radioButton1, 'Value') == 1)
tempSound = [y' zeros(n, 1)];
else
tempSound = [zeros(n, 1) y'];
end

% Play sound.
sound(A * tempSound, Fs);

disp('You push Play_sound button and sound is playing.');
```

6. getKeyPress.m

```
function getKeyPress(obj, event)

% Get figure data from initDisplay.
figData = get(gcf, 'UserData');

switch event.Key
    case 'space'
        dtX = [0.25 0.5 1 2 3 4 6 8];
        XPosition = find(dtX == figData.valueX);

        % Check radio button for play left or right.
        if (get(figData.radioButton1, 'Value') == 1)

            % Find user select point and delete.
            delete(findobj('Tag', 'leftPlot'));

            figData.left_dBTest(XPosition) = figData.valueY;
            leftPlot = plot(figData.axesGraph, dtX, figData.left_dBTest, ...
                    'Color', 'blue', 'Marker', 's', 'MarkerFaceColor', ...
                    'blue', 'MarkerSize', 5, 'Tag', 'leftPlot');
        else

            % Find user select point and delete.
            delete(findobj('Tag', 'rightPlot'));

            figData.right_dBTest(XPosition) = figData.valueY;
            leftPlot = plot(figData.axesGraph, dtX, figData.right_dBTest, ...
                    'Color', 'red', 'Marker', 's', 'MarkerFaceColor', ...
                    'red', 'MarkerSize', 5, 'Tag', 'rightPlot');
        end
```

```
        beep;
    end

    % Set figure data to User data.
    set(gcf, 'UserData', figData);
```

7. clearTested.m

```
function clearTested(obj, event)

% Get figure data from initDisplay.
figData = get(gcf, 'UserData');

% Default value.
dtX = [0.25 0.5 1 2 3 4 6 8];
lineDefault = [100 100 100 100 100 100 100 100];

% Compare obj for get what button is clicked.
switch (obj)
   case figData.clearLeftTested
      compareButton = ' Left Tested';
   case figData.clearRightTested
      compareButton = ' Right Tested';
   case figData.clearAllTested
      compareButton = ' All Tested';
end

ansFromUser = questdlg(strcat('Do you want to delete', compareButton, ' ?'), ...
                       'Warning !!!', 'Yes', 'No', 'No');

switch (ansFromUser)
   case 'Yes'

      % Find data tested, delete and build it to default.
      switch (obj)
         case figData.clearLeftTested
            delete(findobj('Tag', 'leftPlot'));
            figData.left_dBTest = lineDefault;
```

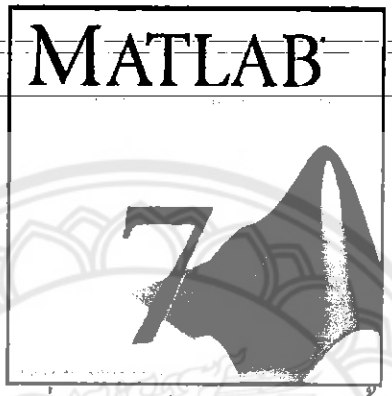
```
                figData.leftPlot = plot(figData.axesGraph, dtX, lineDefault, ...
                                'Color', 'blue', 'Marker', 's', 'MarkerFaceColor', ...
                                'blue', 'MarkerSize', 5, 'Tag', 'leftPlot');
            case figData.clearRightTested
                delete(findobj('Tag', 'rightPlot'));
                figData.right_dBTest = lineDefault;
                figData.rightPlot = plot(figData.axesGraph, dtX, lineDefault, ...
                                'Color', 'red', 'Marker', 's', 'MarkerFaceColor', ...
                                'red', 'MarkerSize', 5, 'Tag', 'rightPlot');
            case figData.clearAllTested
                delete(findobj('Tag', 'leftPlot'));
                delete(findobj('Tag', 'rightPlot'));
                figData.left_dBTest = lineDefault;
                figData.right_dBTest = lineDefault;
                figData.leftPlot = plot(figData.axesGraph, dtX, lineDefault, ...
                                'Color', 'blue', 'Marker', 's', 'MarkerFaceColor', ...
                                'blue', 'MarkerSize', 5, 'Tag', 'leftPlot');
                figData.rightPlot = plot(figData.axesGraph, dtX, lineDefault, ...
                                'Color', 'red', 'Marker', 's', 'MarkerFaceColor', ...
                                'red', 'MarkerSize', 5, 'Tag', 'rightPlot');
            end
            disp(strcat('Delete', compareButton, ' complete.'));
        case 'No'
            disp('You cancle to clearing.');
    end

    % Set figure data to User data.
    set(gcf, 'UserData', figData);
```

ภาคผนวก ข.

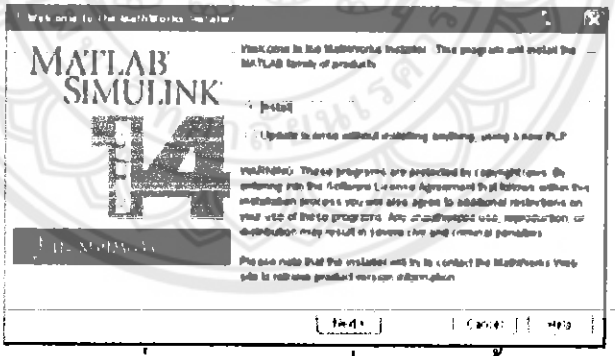
การติดตั้งโปรแกรม MATLAB

1. ดับเบิ้ลคลิกที่ Setup.exe เพื่อเริ่มต้นการติดตั้งโปรแกรม



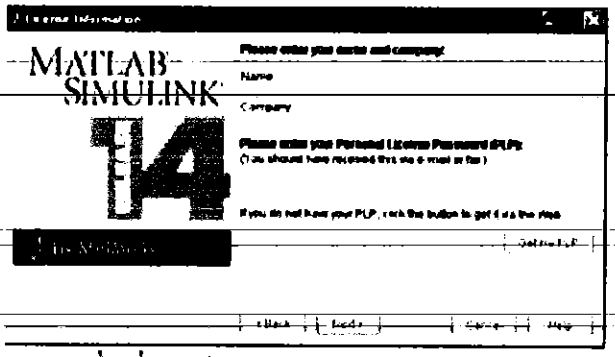
รูปที่ 1 ภาพหน้าแรกของการติดตั้ง

2. เลือก installs เพื่อทำการติดตั้ง จากนั้นคลิก next



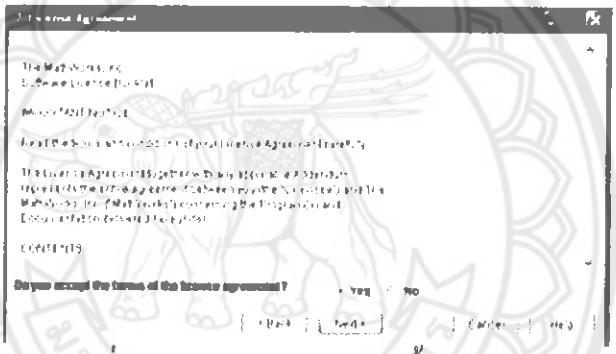
รูปที่ 2 ภาพเลือกชนิดที่ต้องการติดตั้ง

3. ใส่ชื่อ ชื่อองค์กร และ Personal License Password (PLP) จากนั้นคลิก next



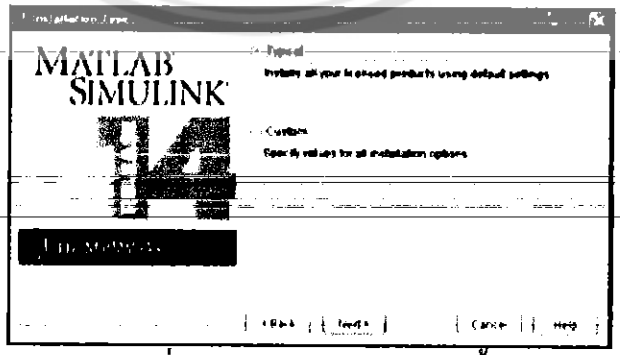
รูปที่ 3 ภาพใส่ชื่อ ชื่อองค์กร และ Personal License Password (PLP)

4. ข้อตกลงในการติดตั้งโปรแกรม คลิกที่ yes จากนั้นต่อด้วย next



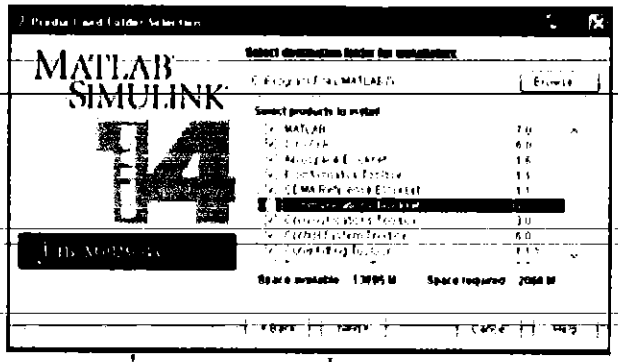
รูปที่ 4 ภาพข้อตกลงในการติดตั้งโปรแกรม

5. เลือกประเภทการติดตั้งโปรแกรม เลือก custom จากนั้นคลิก next



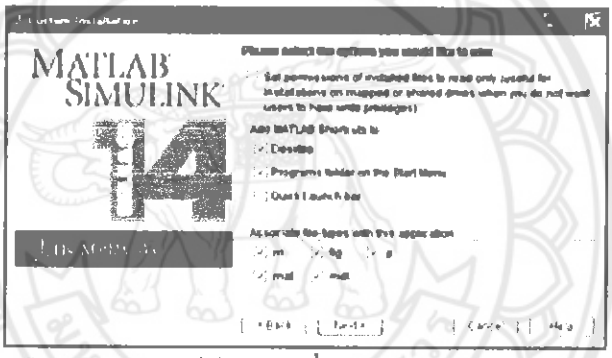
รูปที่ 5 ภาพประเภทของการติดตั้ง

6. เลือกไดเรคทอรี่และองค์ประกอบที่ต้องการติดตั้งโปรแกรมลงในเครื่อง จากนั้นคลิก next



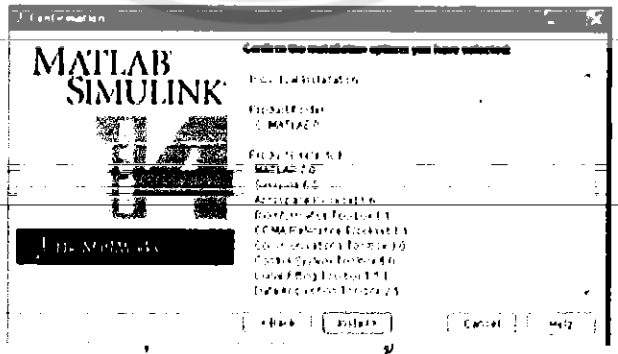
รูปที่ 6 ภาพไดเรคทอรี่และองค์ประกอบ

7. เลือกตระกูลไฟล์และส่วนที่ต้องการให้แสดงการเข้าสู่โปรแกรม จากนั้นคลิก next



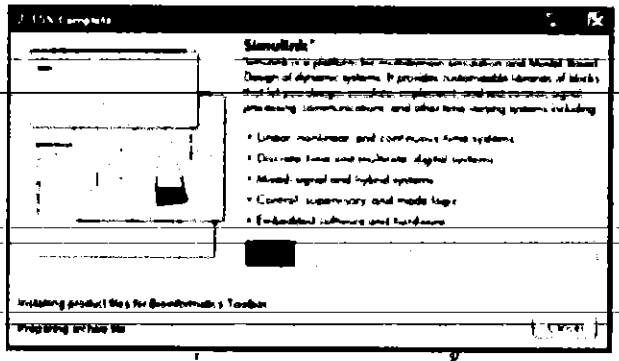
รูปที่ 7 การเลือกตระกูลไฟล์และส่วนที่ต้องการให้แสดงการเข้าสู่โปรแกรม

8. ทำการยืนยันการติดตั้งโปรแกรม จากนั้นคลิก install



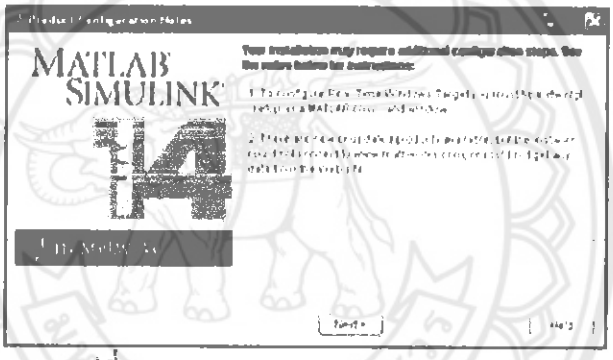
รูปที่ 8 ภาพยืนยันการติดตั้งโปรแกรม

9. สถานะการติดตั้งที่แสดงให้ทราบว่าการดำเนินการติดตั้งเป็นไปกี่เปอร์เซนต์แล้ว



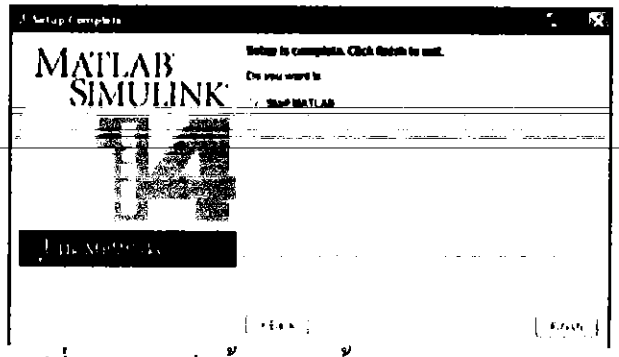
รูปที่ 9 ภาพสถานะการติดตั้ง

10. แสดงค่าการติดตั้งโปรแกรม จากนั้นคลิก next



รูปที่ 10 ภาพ Product Configuration Notes

12. เมื่อทำการติดตั้งสมบูรณ์แล้วจะเลือกว่าต้องการเริ่มทำงานโปรแกรมหรือไม่ จากนั้นคลิก Finish



รูปที่ 11 ภาพเสร็จสิ้นการติดตั้งโปรแกรม MATLAB

ประวัติผู้เขียนโครงงาน

ชื่อ นายชัชฌ์ ปราชญ์วิทยาการ

ภูมิลำเนา 244 ม.7 ต.พิชัย อ.เมือง จ.ลำปาง

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนอัสสัมชัญลำปาง
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 5

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-mail : **khawtaan@hotmail.com**

ชื่อ นายไพโรจน์ ตั้งจิตวิบูลย์กุล

ภูมิลำเนา 961 ต.ในเมือง อ.เมือง จ.พิษณุโลก

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนพิษณุโลกพิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 5

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-mail : **tae_ki@hotmail.com**