สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การกำจัดเสียงรบกวนด้วยไมโครคอนโทรลเลอร์ dsPIC30F6014

NOISE SUPPRESSION BY USING dsPIC30F6014

โดย

นาย ชัยยุทธ อัศวเมธีรุ่งโรจน์

นาย เอกณรงค์ ใจยงค์

ฐ ใ
ช 2217
2549

เลขหมู่.....
เลขทะเบียน.....72869
วัน,เดือน,ปี...25 มิ.ย. 2550

b.1177373x
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมระบบควบคุม สาขาวิชาวิศวกรรมแมคคาทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชาวิศวกรรมระบบควบคุม สาขาวิชาวิศวกรรมแมคคาทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การกำจัดเสียงรบกวนด้วยไมโครคอนโทรลเลอร์ dsPIC30F6014

NOISE SUPPRESSION BY USING dsPIC30F6 014

ผู้จัดทำ นาย ชัยยุทธ อัศวเมธีรุ่งโรจน์ 46010161

นาย เอกณรงค์ ใจยงค์ 46010987

.....อาจารย์ที่ปรึกษา

(อาจารย์ ธวัชชัย คำศรี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำจัดเสียงรบกวนด้วยไมโครคอนโทรลเลอร์ dsPIC30F6014

โดย
นาย ชัยยุทธ อัศวเมธีรุ่งโรจน์ 46010161
นาย เอกณรงค์ ใจยงค์ 46010987

อาจารย์ที่ปรึกษา
อาจารย์ ธวัชชัย คำศรี

ปีการศึกษา 2549

บทคัดย่อ

ปริญญานิพนธ์เล่มนี้ นำเสนอระบบการกำจัดเสียงรบกวนจากเสียงสนทนา ซึ่งเสียงจะประกอบไปด้วย สัญญาณเสียงสนทนา (audio) ที่ไวต่อสัญญาณรบกวน ในโครงงานนี้มีจุดประสงค์เพื่อที่จะแยกสัญญาณเสียงรบกวนโดยใช้ไมโครคอนโทรลเลอร์ dsPIC30F6014 ขนาด 16 บิต เป็นตัวประมวลผล ในส่วนของระบบจะออกแบบมาให้ทำงานบนฐานเวลาจริงและมีการสุ่มตัวอย่างเสียงสนทนาขนาด 16 บิต

ขั้นตอนการทำงานเริ่มจากการแปลงฟูเรียแบบเร็ว (Fast Fourier Traansfrom : FFT) ซึ่งเป็นฟังก์ชันบนโดเมนความถี่ที่ใช้วิเคราะห์องค์ประกอบความถี่ของสัญญาณ หลังจากนั้นฟังก์ชันการคำนวณแยกแยะระหว่างสัญญาณเสียงพูดและสัญญาณเสียงรบกวน (Voice Activity Detection and Noise Band Energy Computation : VAD) จะพิจารณาว่าเป็นสัญญาณเสียงรบกวน หรือเสียงสนทนา ระบบจะกำจัดเสียงรบกวนนั้น และอัพเดท (update) ทุกครั้งที่ฟังก์ชันนี้ถูกเรียกใช้ สำหรับทุกแถบความถี่ของสัญญาณเสียงจะถูกแบ่งตามสัดส่วนของเสียงรบกวนที่ปนมาในแถบความถี่นั้น ดังนั้นแถบความถี่จึงมีความสำคัญมากในการกำจัดเสียงรบกวน ซึ่งขั้นตอนนี้จะปรับเปลี่ยนตามระดับของสัญญาณเสียงสนทนา และระดับของสัญญาณเสียงรบกวนที่เราไม่ต้องการจากสัญญาณ เสียงอินพุท (input) จากนั้นสัญญาณเสียงจะถูกแปลงกลับฟูเรียแบบเร็ว (Inverse Fast Fourier Traansfrom : IFFT) ทำให้สัญญาณเสียงที่ออกมาปราศจากสัญญาณรบกวนในที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOISE SUPPRESSION BY USING dsPIC30F6014

By

Mr. Chaiyut Asawameiteerungroj

Mr. Akenarong Jaiyong

Advisor

Mr. Thawatchai Kamsri

Academic Year 2006

ABSTRACT

This thesis presents the system to suppress the effect of noise interference on an incoming speech signal. Which consist of speech (audio) signal that is susceptible to ambient noise. The purpose of this thesis is to suppress the noise by using 16 bit microcontroller dsPIC30F6014. The system is designed to operate in real time and sample a speech 16 bit .

The project has been conducted as following steps. Firstly, FFT (Fast Fourier Transform) function is primarily a Frequency Domain algorithm in which is performed so as to analyze the frequency components of the signal. After that, VAD (Voice Activity Detecting) is used to determine if the signal segment is speech or noise . Noise Suppression algorithm maintains a profile of noise and updates it every time when a noise segment is detected. Every frequency band of the input signal is then scaled according to the proportion of noise contained in the frequency band, thereby casing a significant degree of noise contained in the resultant signal. The algorithm can adapt along level of noise in the nature and does not require separate noise reference input. Finally, the signal is transferred by using IFFT (Inverse Fast Fourier Transform) algorithm, so the signal will eliminate unwanted noise and it become clean .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับความช่วยเหลือเป็นอย่างดีจาก อาจารย์ ธวัชชัย คำศรี ที่ได้กรุณาให้คำปรึกษาแนะนำที่ดีมาโดยตลอดตั้งแต่ต้น รวมทั้งเอื้อเฟื้ออุปกรณ์ที่จำเป็น และความช่วยเหลืออื่นๆ ที่เป็นประโยชน์ต่อโครงงาน ผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณอย่างสูง

ขอขอบพระคุณ รศ.ดร.โยธิน เปรมปราณีรัชต์ ที่ให้คำปรึกษาในเรื่องการกำจัดสัญญาณรบกวนด้วยดีตลอดมา

ขอบคุณเพื่อนๆ ทุกคนที่ให้กำลังใจ สนับสนุนอุปกรณ์ที่ขาดเหลือ กระตุ้นเตือน รวมทั้งคอยถามไถ่ความคืบหน้าของโครงงานอยู่เสมอ

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณบิดา มารดา และครอบครัว ที่คอยเป็นกำลังใจที่ดีตลอดมา รวมถึงการสนับสนุนในเรื่องของงบประมาณที่ขาดเหลือ ตลอดจนเป็นแรงบันดาลใจที่ดีที่สุดที่ทำให้โครงงานนี้สำเร็จสมบูรณ์ลงได้

ผู้จัดทำ
นายชัยยุทธ อัสวะเมทีรุ่งโรจน์
นายเอกณรงค์ ใจยงค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา	2
1.3 ขอบเขตของการศึกษา	2
1.4 วัสดุอุปกรณ์ที่ใช้ในการสร้างชิ้นงาน	2
1.5 รายละเอียดของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง	3
2.1 ทฤษฎีการประมวลผลสัญญาณดิจิตอล	3
2.1.1 การประมวลผลข้อมูลโดยการแปลงฟูเรียร์แบบเร็ว	3
2.1.2 ทฤษฎีการสุ่มตัวอย่าง	8
2.1.3 การแบ่งช่วงสัญญาณ	9
2.1.4 การเข้ารหัสของสัญญาณเสียงแบบพัลส์โค้ดมอดูเลชั่น	10
2.1.5 การสังเคราะห์เสียงเชิงคณิตศาสตร์	15
2.1.6 เฟสล็อกลูป	17
2.2 สถาปัตยกรรมไมโครคอนโทรลเลอร์ dsPIC30F6014	28
2.2.1 คุณสมบัติโดยรวมของ dsPIC30F6014	28
2.2.2 โมดูลสำหรับเชื่อมต่ออุปกรณ์อนุกรม แบบ SPI	32
2.2.3 โมดูลสำหรับเชื่อมต่ออุปกรณ์อนุกรม แบบ DCI	33
บทที่ 3 หลักการออกแบบ	37
3.1 หน่วยเข้าและถอดรหัสเสียง	39
3.1.1 คุณสมบัติของ Si3000	39
3.1.2 การเชื่อมต่อแบบดิจิตอล	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
3.1.3 การเลือกความถี่สัญญาณนาฬิกา	43
3.1.4 การกำหนดค่ารีจิสเตอร์สำหรับตัวกำเนิดสัญญาณนาฬิกา	44
3.2 หน่วยแสดงผล	45
3.3 หน่วยประมวลผลกลางและกำจัดเสียงรบกวน	46
3.3.1 กระบวนการกำจัดเสียงรบกวน	46
3.3.2 โปรแกรมการทำงาน	54
บทที่ 4 การทดลอง	67
4.1 วัตถุประสงค์ของการทดลอง	67
4.2 วิธีการทดลอง	68
4.3 ผลการทดลอง	69
4.3 สรุปผลการทดลอง	69
บทที่ 5 บทวิจารณ์และสรุป	77
5.1 สรุปผลการทดลอง	77
5.2 ปัญหาที่พบและแนวทางแก้ไข	78
5.3 ข้อเสนอแนะและแนวทางในการค้นคว้าพัฒนา	78
ภาคผนวก ก ไอซีเข้ารหัสและถอดรหัสเสียง Si3000	80
เอกสารอ้างอิง	90

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้า
1.1 ลักษณะการกำจัดเสียงรบกวนของชิ้นงาน	1
2.1 การเปรียบเทียบจำนวนครั้งของการคำนวณระหว่าง DFT และ FFT	4
2.2 ค่าเอกซ์โพเนนเชียลของ W (แฟกเตอร์การหมุน)	5
2.3 ความสัมพันธ์ระหว่างโดเมนเวลาและโดเมนความถี่	6
2.4 แสดงฟังก์ชันหน้าต่างในแบบต่างๆ	8
2.5 ระบบการประมวลผลสัญญาณแอนะลอกโดยใช้การประมวลผลสัญญาณดิจิตอล	8
2.6 แสดงการแบ่งช่วงสัญญาณ	9
2.7 กระบวนการเข้ารหัสและถอดรหัสของระบบพีซีเอ็ม	10
2.8 การชักตัวอย่างสัญญาณเสียง	11
2.9 สัญญาณแอนะลอกดั้งเดิมก่อนทำการจัดระดับสัญญาณ	11
2.10 สัญญาณดิจิตอลหลังขณะทำการจัดระดับสัญญาณ	14
2.11 การแทรกขั้นตอนของการอัดสัญญาณและการยืดสัญญาณลงในระบบพีซีเอ็ม	14
2.12 คุณลักษณะการคอมเพรสชั่นของไดโอด	14
2.13 แบบอย่างคุณลักษณะของการคอมเพรสชั่น	14
2.14 แสดงการสังเคราะห์สัญญาณเสียง	15
2.15 บล็อกไดอะแกรมของการเปล่งเสียง	16
2.16 แผนผังของเฟสล็อกลูป	17
2.17 แผนผังของหน่วยสังเคราะห์ความถี่	20
2.18 คุณลักษณะ ช่วงพลวัต (dynamic range) ในการเปลี่ยนความถี่เฟสล็อกลูป	21
2.19 ลักษณะสมบัติของส่วนเปรียบเทียบเฟส ชนิดแอนะลอก	23
2.20 ลักษณะสมบัติ ส่วนเปรียบเทียบเฟส ชนิดดิจิตอล	24
2.21 ลักษณะสมบัติของ VCO	24
2.22 วงจรเฟสล็อกลูป	26
2.23 บล็อกไดอะแกรม ของ โมดูล SPI	32
2.24 การเชื่อมต่อโมดูล SPI ระหว่างอุปกรณ์มาสเตอร์ และอุปกรณ์สเลฟ	33
2.25 บล็อกไดอะแกรม ของ โมดูล DCI	34
2.26 แสดงตัวอย่างการเชื่อมต่อกับอุปกรณ์เข้าและถอดรหัส	35
2.27 ไทมิงไดอะแกรมของการทำงานแบบ สเลฟเฟรมซิงค์ ชนิดหลายช่องทาง	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

รูปที่	หน้า
3.1 โครงสร้างการทำงานในการกำจัดสัญญาณเสียงรบกวน	38
3.2 บล็อกไดอะแกรม ของไอซี Si3000	39
3.3 ไทมิงไดอะแกรมของการเชื่อมต่อแบบอนุกรม	40
3.4 การร้องขอเฟรมทุติยภูมิ	42
3.5 การเขียนข้อมูลลงเฟรมทุติยภูมิ	42
3.6 การอ่านข้อมูลจากเฟรมทุติยภูมิ	43
3.7 ระบบการสร้างสัญญาณนาฬิกาด้วยเฟสล็อกลูป	43
3.8 บล็อกไดอะแกรมการกำจัดเสียงรบกวน	47
3.9 แผนผังการทำงานโดยรวมของโปรแกรม	56
3.10 แผนผังการทำงานของฟังก์ชัน InitSPI()	57
3.11 แผนผังการทำงานของฟังก์ชัน Init_DCI()	58
3.12 แผนผังการทำงานของฟังก์ชัน DCI Interrupt()	59
3.13 แผนผังการทำงานของฟังก์ชัน Init_Si3000()	61
3.14 แผนผังการทำงานของโปรแกรมภาษาซี ns_8k.c	62
4.1 เสียงสนทนา ก่อนเข้าสู่ระบบ	69
4.2 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน	69
4.3 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว	70
4.4 เสียงสนทนา ก่อนเข้าสู่ระบบ	70
4.5 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน	71
4.6 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว	71
4.7 เสียงสนทนา ก่อนเข้าสู่ระบบ	72
4.8 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน	72
4.9 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว	73
4.10 เสียงสนทนา ก่อนเข้าสู่ระบบ	73
4.11 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน	74
4.12 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงคุณสมบัติของแต่ละฟังก์ชันหน้าต่าง	7
2.2 ความสัมพันธ์ระหว่างรูปแบบทางกายภาพกับรูปแบบทางคณิตศาสตร์ของสัญญาณเสียง	15
2.3 ย่านสัญญาณความถี่นาฬิกาที่สามารถใช้ร่วมกับวงจรเฟสล็อกลูปภายใน	31
3.1 คุณลักษณะการสวิตช์ชิงของการเชื่อมต่อแบบอนุกรม	40
3.2 โหมดการเชื่อมต่อแบบอนุกรม	41
3.3 ตัวอย่างค่าความถี่จาก MCLK ในการสร้างความถี่สุ่มตัวอย่าง 8 kHz	44
3.4 สัมประสิทธิ์ของวินโดว์ฟังก์ชันต่างๆ	48
3.5 แถบพลังงานของสเปกตรัมความถี่	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

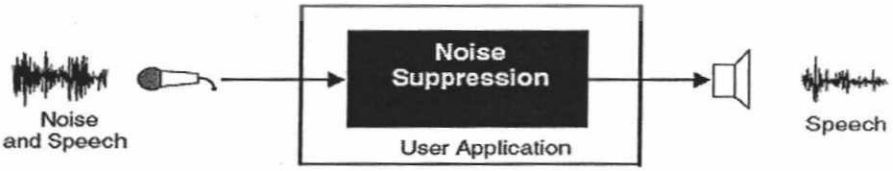
บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เสียงเกิดจากการสั่นสะเทือนของวัตถุ เมื่อวัตถุสั่นสะเทือนก็จะทำให้เกิดการอัดตัว และขยายตัวของคลื่นเสียง สัญญาณเสียงนั้นจะถูกส่งผ่านตัวกลาง เช่น อากาศไปยังอุปกรณ์รับสัญญาณเสียง แต่ถ้าสัญญาณเสียงนั้นประกอบด้วยสัญญาณรบกวนแล้ว สัญญาณเสียงที่ปรากฏที่ตัวรับก็ย่อมรับเอาสัญญาณรบกวนนั้นเข้าไปด้วย ทำให้คุณภาพสัญญาณเสียงต่ำลงไป ปัญหานี้พบได้มากในชีวิตประจำวัน ตัวอย่างเช่น การติดต่อสื่อสารทางโทรศัพท์ ผู้รับสารอาจได้รับสัญญาณเสียงรบกวนจากผู้ส่งสาร หรือเสียงรบกวนจากตัวโทรศัพท์ที่ไม่มีคุณภาพ ตลอดจนอาจเกิดจากระบบการติดต่อสื่อสารที่สร้างคลื่นรบกวนออกมา จะเห็นว่าผู้รับสารหลีกเลี่ยงไม่ได้ที่จะรับสัญญาณเสียงที่ต้องการและเสียงรบกวนที่เข้ามาในระบบในเวลาเดียวกัน

ด้วยเหตุนี้จากอดีตจนถึงปัจจุบัน จึงมีวิธีการกำจัดเสียงที่ไม่พึงประสงค์หลายวิ ธี เริ่มจากวงจรกรองความถี่แบบแอนะลอก (analog filter) แต่วงจรนี้ก็ยังมีข้อจำกัดจากความถี่คัทออฟ (cut off frequency) อยู่ช่วงใดช่วงหนึ่ง ซึ่งไม่เหมาะกับสัญญาณเสียงรบกวนที่มีหลายองค์ประกอบด้วยกัน ; วงจรกรองความถี่แบบดิจิตอล (digital filter) ซึ่งมีการพัฒนาการเขียนอัลกอลิทึม (algorithm) ต่างๆ เป็นภาษาคอมพิวเตอร์แล้วประมวลผลโดยมีหน่วยความจำข้อมูล หน่วยความจำโปรแกรม และหน่วยประมวลผลทางคณิตศาสตร์ แต่วิธีนี้เป็นการประมวลโดยใช้เครื่องคอมพิวเตอร์ ซึ่งมีขนาดใหญ่ไม่เหมาะกับการนำไปใช้ในชีวิตประจำวันจริง

ปริญญานิพนธ์เล่มนี้ มุ่งให้ความสนใจในการกำจัดสัญญาณเสียงรบกวนจาก เสียงสนทนา โดยใช้การประมวลผลสัญญาณเสียงแบบดิจิตอล (Digital Signal Processing : DSP) ซึ่งมีไมโครคอนโทรลเลอร์ (microcontroller) ขนาด 16 บิตเป็นหน่วยประมวลผล โดยมีลักษณะการทำงานภาพรวมในรูปที่ 1.1 วิธีนี้จะทำให้ชิ้นงานมีขนาดเล็กลง แต่ประสิทธิภาพเท่าเทียมกับการประมวลผลโดยใช้เครื่องคอมพิวเตอร์ นอกจากนั้นยังสามารถนำชิ้นงานไปใช้งานได้จริง ตลอดจนต้นทุนในการสร้างมีราคาถูก และผู้ใช้งานไม่จำเป็นที่ต้องมีประสบการณ์มากนัก สิ่งเหล่านี้จึงเป็นเครื่องยืนยันได้อย่างดีว่า "การกำจัดเสียงรบกวนด้วยไมโครคอนโทรลเลอร์ dsPIC30F6014" เป็นคำตอบของการกำจัดเสียงรบกวนอย่างแท้จริง



รูปที่ 1.1 ลักษณะการกำจัดเสียงรบกวนของชิ้นงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1. เพื่อศึกษาระบบกำจัดสัญญาณเสียงรบกวนจากเสียงสนทนาบนฐานเวลาจริง (real time) ด้วยวิธีการประมวลผลสัญญาณแบบดิจิตอล โดยใช้ไมโครคอนโทรลเลอร์ขนาด 16 บิต เป็นตัวประมวลผล
2. เพื่อสร้างชิ้นงานที่มีขนาดเล็กกว่าเครื่องคอมพิวเตอร์ และสามารถนำชิ้นงานไปใช้งานได้จริงในชีวิตประจำวัน โดยที่ผู้ใช้งานไม่จำเป็นต้องมีความรู้มากนัก
3. เพื่อศึกษาระบบการประมวลผลสัญญาณแบบเฟรม (frame)
4. เพื่อศึกษาระบบการเชื่อมต่ออุปกรณ์อนุกรมแบบ DCI
5. เพื่อศึกษาระบบการเชื่อมต่ออุปกรณ์อนุกรมแบบ SPI

1.3 ขอบเขตของการศึกษา

ลดสัญญาณเสียงรบกวนจากเสียงสนทนา และสร้างระบบการประมวลผลขึ้นมาบนฐานเวลาจริง โดยการบันทึกเสียงสนทนาที่มีเสียงสัญญาณรบกวนแล้วนำเสียงสนทนานั้นป้อนเข้าสู่ระบบที่สร้างขึ้นมา เพื่อทำการเปรียบเทียบผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของสัญญาณ เสียงรบกวนก่อนการกำจัดเสียงและหลังการกำจัดเสียง

1.4 วัสดุอุปกรณ์ที่ใช้ในการสร้างชิ้นงาน

1. บอร์ดทดลอง dsPICDEM 1.1
2. เครื่องบันทึกเสียง (MP3)
3. หูฟัง หรือ ลำโพง

1.5 รายละเอียดของปริญญานิพนธ์

เนื้อหาที่จะกล่าวในปริญญานิพนธ์ฉบับนี้ประกอบด้วย

บทที่ 1 บทนำ กล่าวนำถึงความเป็นมาของปัญหา วัตถุประสงค์ตลอดจนขอบเขตการศึกษา พร้อมทั้งรายละเอียดของปริญญานิพนธ์ของแต่ละบท

บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง กล่าวถึงหลักการและทฤษฎีที่เกี่ยวข้องในการสร้างชิ้นงาน แบ่งเป็นสองหมวดคือ ทฤษฎีที่เกี่ยวข้องกับการประมวลผลสัญญาณดิจิตอล และความรู้ที่เกี่ยวข้องในการเขียนโปรแกรม

บทที่ 3 หลักการออกแบบ นำเสนอโครงสร้างโดยรวมของระบบ รวมถึงแนวคิดในการออกแบบระบบกำจัดสัญญาณเสียงรบกวน

บทที่ 4 การทดลอง ทำการเปรียบเทียบผลตอบสนองเชิงความถี่และผลตอบสนองเชิงเวลาของสัญญาณเสียงก่อนและหลังการกำจัดเสียงรบกวน

บทที่ 5 บทวิจารณ์และสรุป จะสรุปผลการดำเนินงาน ปัญหาที่เกิดขึ้น และแนวทางการปรับปรุงพัฒนาโครงงานนี้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและความรู้ที่เกี่ยวข้อง

จากที่ได้กล่าวในบทที่ 1 ก่อนการออกแบบ และสร้างระบบการกำจัดเสียงรบกวนนั้น จำเป็นต้องศึกษาองค์ประกอบต่างๆที่จำเป็นให้เข้าใจเสียก่อน พบว่าระบบ การกำจัดเสียงรบกวนมีส่วนที่สำคัญหลายส่วน ดังนั้นในบทนี้จะศึกษาและอธิบายถึงองค์ประกอบต่างๆ ที่จะนำไปใช้งานในการสร้างชิ้นงานจริง ซึ่งแบ่งออกเป็นสองส่วนคือ ทฤษฎีที่เกี่ยวข้องกับการประมวลผลสัญญาณดิจิตอล และความรู้ที่เกี่ยวข้องในการเขียน โปรแกรม อันจะกล่าวดังต่อไปนี้

2.1 ทฤษฎีการประมวลผลสัญญาณดิจิตอล

ความรู้เกี่ยวกับการประมวลผลสัญญาณดิจิตอลนั้น นับเป็นส่วนสำคัญที่จำเป็นจะต้องศึกษาก่อน เพื่อเป็นพื้นฐานของการออกแบบระบบการกำจัดเสียงรบกวน โดยต้องทำความเข้าใจเกี่ยวกับหลักการประมวลผลสัญญาณดิจิตอลเบื้องต้น ซึ่งประกอบไปด้วยหัวข้อต่างๆ ดังนี้

2.1.1 การประมวลผลข้อมูลโดยการแปลงฟูเรียร์แบบเร็ว

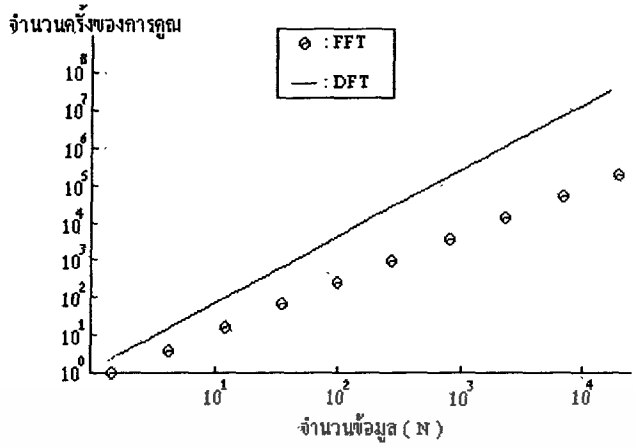
1. การแปลงฟูเรียร์

ข้อมูลส่วนใหญ่จะเป็นเหตุการณ์ที่เปลี่ยนแปลงไปตามเวลา ซึ่งข้อมูลที่แสดงได้เหล่านี้จะสามารถแสดงได้โดยใช้การประกอบกันของคลื่นฟังก์ชันตรีโกณมิติที่มีความถี่ต่างกัน ซึ่งฟังก์ชันที่ได้จากแกนของเวลาสามารถแสดงได้ด้วยฟังก์ชันที่แสดงบนแกนของความถี่ การแปลงแบบนี้เรียกว่า "การแปลงฟูเรียร์" ความสัมพันธ์ระหว่างฟังก์ชันในแกนของเวลา $F(t)$ และ ฟังก์ชันในแกนของความถี่ $F(\omega)$ ได้ดังสมการที่ (2.1) และ (2.2) ดังนี้

$$F(\omega) = \int_{-\infty}^{+\infty} F(t) e^{-j\omega} dt \qquad (2.1)$$

ถ้าทำการแปลงฟูเรียร์กับสัญญาณดิจิตอลจะเรียก การแปลงนั้นว่า การแปลงฟูเรียร์แบบไม่ต่อเนื่อง (Discrete Fourier Transform : DFT) ซึ่งจะแทนการอินทิเกรตจากอนันต์ถึงลบอนันต์ ด้วยการหาค่าผลรวมที่จำกัด (finite summation) การแปลงฟูเรียร์นี้ไม่ต่อเนื่องทั้งโดเมนเชิงเวลา และโดเมนเชิงความถี่ นอกจากนี้ยังมีการพัฒนาการแปลงฟูเรียร์ให้เร็วขึ้น หรือที่เรียกว่า "การแปลงฟูเรียร์แบบเร็ว(Fast Fourier Transform : FFT)" ซึ่งเป็นอัลกอริทึมที่ย่นเวลาในการคำนวณ โดยลดจำนวนครั้งการคำนวณสำหรับแปลงฟูเรียร์แบบไม่ต่อเนื่อง ให้กับข้อมูลที่มีขนาดใหญ่ ดังแสดงในรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 การเปรียบเทียบจำนวนครั้งของการคำนวณระหว่าง DFT และ FFT

2. การแปลงฟูเรียร์แบบเร็ว

สมมุติให้ $F(t)$ เป็นสัญญาณต่อเนื่อง การแปลงฟูเรียร์แบบไม่ต่อเนื่อง จะแสดงดังสมการที่ (2.2) ดังนี้

$$C_k = \frac{1}{N}\sum_{i=1}^{N-1} f(i)\frac{e^{-j2\pi ki}}{N} \qquad (2.2)$$

เมื่อ: C_k คือ ผลรวมการแปลงฟูเรียร์แบบไม่ต่อเนื่อง ณ จุดสัญญาณที่ k

k คือ จุดของข้อมูลตั้งแต่ 0 ถึง N

วิธีคำนวณการแปลงฟูเรียร์แบบไม่ต่อเนื่อง จำเป็นต้องคำนวณผลบวกทั้งหมด N^2 ครั้ง ซึ่งมีขนาดใหญ่มาก ดังนั้นวิธีคำนวณการแปลงฟูเรียร์แบบเร็วจึงพัฒนาขึ้นเพื่อลดจำนวนครั้งของการคูณ โดยทำการแปลงการคูณนั้นให้เป็นการบวกด้วยการรวมพจน์ก่อนเข้าคูณ ดังสมการที่ (2.3)
กำหนดให้

$$W = e^{-j\left(2\pi/N\right)} = \cos\left(\frac{2\pi}{N}\right) - \sin\left(\frac{2\pi}{N}\right) \qquad (2.3)$$

เมื่อ: W คือ แฟกเตอร์การหมุน

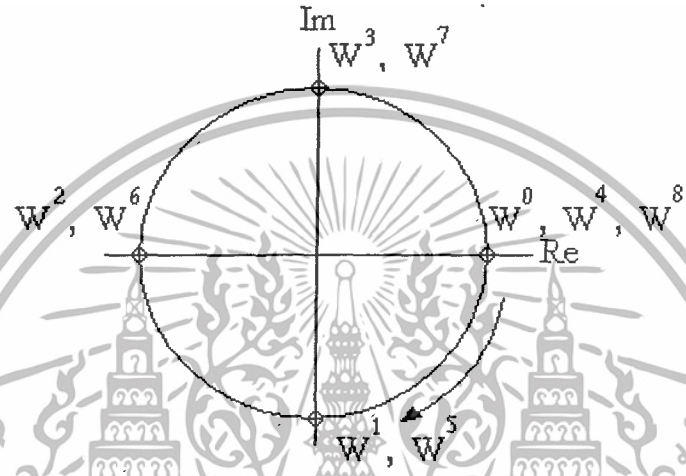
สมการที่(2.2) จะถูกแปลงเป็นสมการที่(2.4) ดังนี้

$$C_k = \frac{1}{N}\sum_{i=1}^{N-1} f(i) * W^{ki} \qquad \text{เมื่อ} \qquad n = ki \qquad (2.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในสมการนี้เรียกว่า W^{ki} ว่า "แฟกเตอร์การหมุนเฟส (phase-rotation factor)"

แนวคิดพื้นฐานการแปลงฟูเรียร์แบบเร็วจะใช้คุณสมบัติ 2 อย่างของ W^n เช่น รูปที่2.2 แสดงการแบ่งวงกลมหนึ่งหน่วย (unit circle) ที่มีจุดศูนย์กลางอยู่ที่จุดกำเนิด (origin) ของระนาบเชิงซ้อน (complex plane) ออกเป็น N ส่วนเท่า ๆ กัน เมื่อมีการเพิ่ม N ก็จะหมุนไปทิศลบ ดังนั้นจากคุณสมบัติการสมมาตร ค่าของ W^6, W^2 จะเท่ากับค่าของ $-W^0$ และจากคุณสมบัติความเป็นรอบ W^4, W^8 จะเท่ากับค่าของ W^0 จึงสามารถแบ่งการคูณได้ดังสมการ ที่ (2.4)



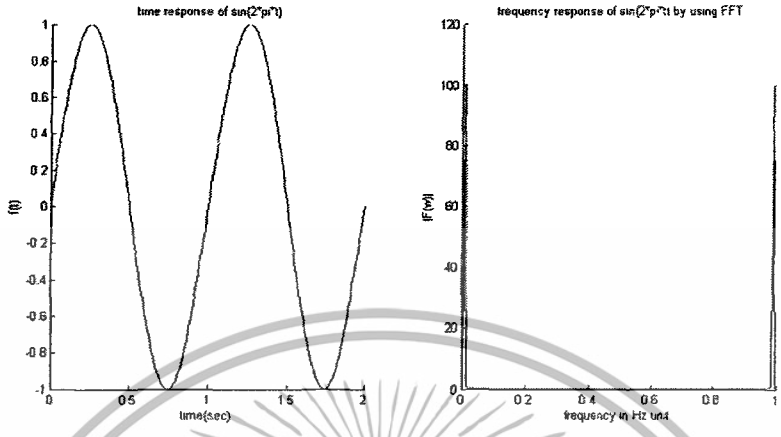
รูปที่ 2.2 ค่าเอกซ์โพเนนเชียลของ W (แฟกเตอร์การหมุน)

3. การประมวลผลแบบใช้วินโดว์

การแปลงฟูเรียร์แบบเร็วจะประมวลผลข้อมูลตัวเลขที่สุ่มมาเป็นส่วนๆจากชุดของข้อมูลที่มีทั้งหมด การตัดข้อมูลหรือคลื่นออกมาส่วนๆแบบนี้เรียกว่า "การใส่วินโดว์ (หน้าต่างเวลา)" แต่เดิมการแปลงฟูเรียร์แบบเร็วถูกนิยามอยู่บนข้อมูลที่มีความยาวไม่จำกัด (วิธีการ DFT ก็เช่นเดียวกัน) วิธีการแปลงฟูเรียร์แบบเร็วจะตัดคลื่นออกเป็นวินโดว์ และสมมุติว่า คลื่นนั้นเป็นคลื่นที่เกิดซ้ำกันไปเรื่อยๆ จากนั้นจะทำการวิเคราะห์สัญญาณ ถ้าความยาวของข้อมูลที่วิเคราะห์นั้นเป็นจำนวนเต็มเท่าของรอบความถี่แต่ละอัน คลื่นที่แปลงฟูเรียร์แบบเร็วนั้นจะเหมือนกับคลื่นอินพุทที่เข้ามา และจะได้ ไลน์สเปกทรัม (line spectrum) อันเดียวดังรูปที่ 2.3 แต่ถ้าความยาวของข้อมูลที่วิเคราะห์นั้นไม่เป็นจำนวนเต็มเท่าของรอบความถี่ (ปลายเริ่มกับปลายปิดไม่เชื่อมต่อกัน) ในกรณีนี้จะต้องจัดการกับคลื่นที่บิดเบี้ยวไป (distorted wave) โดยทฤษฎีแล้วเราไม่สามารถรวมพลังงานไว้ที่สเปกตรัมนั้นแต่จะเกิดการขยายไปข้างๆหรือที่เรียกว่า "ไซด์โลบ (side lobe)" การรั่วของพลังงานเช่นนี้เรียกว่า "ค่าผิดพลาดจากการรั่ว (leakage error)" ตามทฤษฎีแล้ว เพื่อให้ได้มาซึ่งสเปกตรัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อันเดียวจำเป็นต้องใช้ข้อมูลที่ยาวไม่จำกัดแต่โดยทั่วไปแล้ว การแปลงฟูเรียร์แบบเร็วจะประมวลผลสัญญาณข้อมูลในช่วงที่จำกัด ดังนั้นจึงเกิดค่าผิดพลาดจากการรั่ว



รูปที่ 2.3 ความสัมพันธ์ระหว่างโดเมนเวลาและโดเมนความถี่

วิธีหนึ่งในการป้องกันค่าผิดพลาดจากการรั่วที่พบโดยทั่วไปคือ "วิธีประมวลผลด้วยวินโดว์ (window processing)" โดยจะใช้ฟังก์ชันรูปภูเขาที่ปลาย 2 ข้างของเฟรมเป็นศูนย์ ทำการฟิลเตอร์สัญญาณ ซึ่งจะทำให้ปลายเริ่มกับปลายจบเชื่อมต่อกัน ฟังก์ชันแบบนี้เรียกว่า "ฟังก์ชันหน้าต่าง (window function)" การใช้ฟังก์ชันหน้าต่างในการทำชิงโครไนซ์สัญญาณที่วิเคราะห์เรียกว่า "การประมวลผลแบบใช้วินโดว์" ผลลัพธ์ที่ได้จะมีรูปร่างของสเปกตรัมใกล้เคียงกับไลน์สเปกตรัม

ส่วนของทฤษฎีจะใช้ฟังก์ชันหน้าต่างในการควบคุมค่าของ แถบความถี่เปลี่ยน ,ขนาดของริปเปิ้ล ซึ่งมีอยู่หลายหน้าต่างเช่น หน้าต่างสี่เหลียม , หน้าต่างแฮมมิ่ง ,หน้าต่างฮานนิ่ง , Barllet , Keiser ในแต่ละแบบจะมีผลที่ออกมาไม่เหมือนกัน เช่นการลด ริปเปิ้ลแถบผ่าน และริปเปิ้ลแถบหยุดที่ไม่เท่ากัน คุณลักษณะของฟังก์ชันหน้าต่างในแบบต่างๆ แสดงได้ดังสมการที่ (2.5) ถึง (2.11) และแสดงรูปของหน้าต่างๆ ในรูปที่ 2.4

หน้าต่างสี่เหลียม

$$W(n) = 1 \qquad \text{เมื่อ } -\omega < \omega < \omega_c \tag{2.5}$$
$$= 0 \qquad \text{นอกจากนั้น}$$

หน้าต่างฮานนิ่ง

$$W(n) = 0.5\left[(1) - \cos\left(2 \cdot \pi \cdot \frac{n}{N-1}\right)\right] \qquad \text{เมื่อ } 0 < n < N-1 \tag{2.6}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าต่างแฮมมิ่ง

$$W(n) = 0.54 - 0.46\cos\left(2\cdot\pi\cdot\frac{n}{N-1}\right) \qquad \text{เมื่อ } 0 < n < N-1 \qquad (2.7)$$

หน้าต่างแบร์เร็ต

$$W(n) = 1 - 2\cdot\frac{\left|n - \left(\frac{N-1}{2}\right)\right|}{N-1} \qquad \text{เมื่อ } 0 < n < N-1 \qquad (2.8)$$

หน้าต่างแบล็กแมน

$$W(n) = 0.42 + 0.5\cos\left(2\cdot\pi\cdot\frac{n}{N-1}\right) + 0.08\cos\left(4\cdot\pi\cdot\frac{n}{N-1}\right)$$

$$\text{เมื่อ } 0 < n < N-1 \qquad (2.9)$$

หน้าต่างไคเซอร์

$$W(n) = \begin{cases} \dfrac{Io\left[\beta\cdot\sqrt{1-\left[\dfrac{(n)}{\left(\frac{N-1}{2}\right)}\right]^2}\right]}{Io(\beta)} & \text{เมื่อ } 0 < n < N-1 \\ 0 & \text{นอกจากนั้น} \end{cases} \qquad (2.10)$$

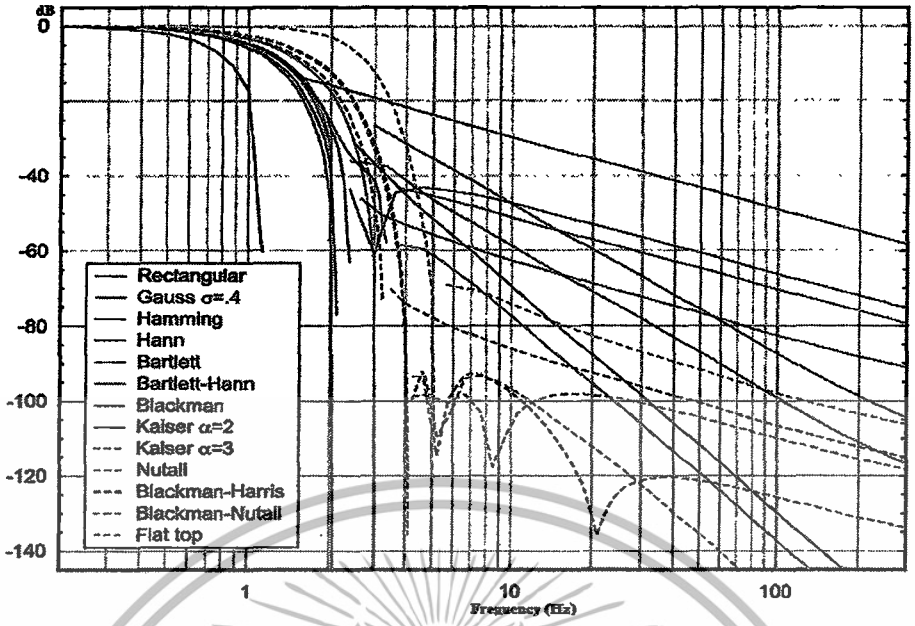
เมื่อ β คือพารามิเตอร์ในการควบคุม แถบความถี่เปลี่ยนและ ริปเปิ้ลแถบผ่าน , ริปเปิ้ลแถบหยุด ดังสมการที่ 2.11

$$Io(x) = 1 + \sum_{k=1}^{L}\left[\left(\frac{1}{k!}\right)\cdot\left(\frac{x}{2}\right)^k\right]^2 \qquad (2.11)$$

ตารางที่ 2.1 แสดงคุณสมบัติของแต่ละฟังก์ชันหน้าต่าง

ฟังก์ชันหน้าต่าง	แถบความถี่เปลี่ยน	การลดทอนของแถบหยุด (dB)
สี่เหลี่ยม	$1.8\pi/N$	20.9
แบร์เร็ต	$6.1\pi/N$	25
ฮานนิ่ง	$6.2\pi/N$	43.9
แฮมมิ่ง	$6.6\pi/N$	54.5
แบล็กแมน	$11\pi/N$	75.3
ไคเซอร์	ปรับค่าได้	ปรับค่าได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 แสดงฟังก์ชันหน้าต่างในแบบต่างๆ

2.1.2 ทฤษฎีการสุ่มตัวอย่าง

อาจกล่าวได้ว่าสัญญาณในธรรมชาติส่วนมาก เช่น สัญญาณเสียง สัญญาณการสั่นสะเทือนของพื้นโลก คลื่นหัวใจ หรือการแปรค่าไปของอุณหภูมิ เหล่านี้เป็นไปในลักษณะที่มีความสัมพันธ์กับเวลาที่เปลี่ยนไป หรือกล่าวได้ว่าเป็นสัญญาณแอนะลอก การนำสัญญาณเหล่านี้ไปประมวลผลในลักษณะของดิจิตอล จะต้องใช้ระบบการประมวลผลดังรูปที่ 2.5

รูปที่ 2.5 ระบบการประมวลผลสัญญาณแอนะลอกโดยใช้การประมวลผลสัญญาณดิจิตอล

ซึ่งตามรูป วงจร S/H เป็นวงจรสุ่มและคงค่าสัญญาณไว้ เพื่อให้วงจร A/D ทำการแปลงสัญญาณจากแอนะลอกเป็นสัญญาณดิจิตอล

ในการแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล ความถี่ในการสุ่มตัวอย่างนั้น จะต้องมากพอที่จะไม่ทำให้สัญญาณสูญเสียข้อมูลที่สำคัญออกไป

ทฤษฎีการสุ่มตัวอย่าง (sampling theory) ของ Shannon กล่าวไว้ว่า ถ้าหากมีสัญญาณแอนะลอก $x(t)$ ที่ค่าการแปลงฟูเรียร์ หรือเพาเวอร์ สเปคตรัม (power spectrum) ของสัญญาณมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

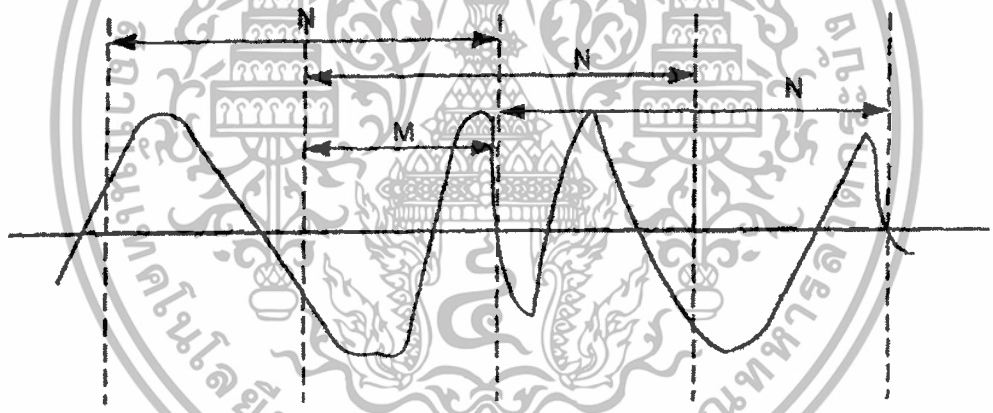
แถบความถี่ปฏิบัติงานเท่ากับ f_0 แล้วเราจะสามารถทำการสุ่มโดยที่สัญญาณนั้นไม่สูญเสียเนื้อหาที่สำคัญก็ต่อเมื่อความถี่ในการสุ่ม f_s มีค่ามากกว่าสองเท่าของความถี่ f_0 ดังสมการที่ (2.12)

$$f_s \geq 2f_0 \tag{2.12}$$

โดยทั่วไปเราอาจสุ่มด้วยความถี่ $f_s = 2f_0$ พอดี ค่าความถี่นี้เรียกว่าความถี่ไนควิสต์ (Nyquist frequency) แต่ในทางปฏิบัติเพื่อหลีกเลี่ยงการไม่เป็นเชิงเส้นของการสุ่ม เราจึงมักใช้ความถี่ในการสุ่มที่มากกว่าความถี่ไนควิสต์ และความถี่ในการสุ่มที่ใช้นั้นขึ้นอยู่กับลักษณะงาน ไม่กำหนดขอบเขตที่แน่นอน ดังนั้นการสุ่มสัญญาณที่จะได้ผลถูกต้องนั้น จำเป็นต้องรู้แถบความถี่ในการปฏิบัติงานของสัญญาณ ดังนั้นในการประมวลผลเราอาจใช้ตัวกรองความถี่ต่ำ (low pass filter) เป็นตัวกำหนดแถบความถี่ของสัญญาณก่อนที่จะเข้าสู่วงจร S/H

2.1.3 การแบ่งช่วงสัญญาณ

ข้อมูลเสียงที่อยู่ในโดเมนเวลา จะถูกแบ่งออกเป็นช่วงสัญญาณ (Block) ซึ่งแต่ละช่วงสัญญาณจะประกอบด้วยสัญญาณเสียง N สัญญาณ และในแต่ละช่วงสัญญาณจะถูกวิเคราะห์โดยการเลื่อนสัญญาณไปครั้งละ M สัญญาณ ดังแสดงในรูปที่ 2.6



รูปที่ 2.6 แสดงการแบ่งช่วงสัญญาณ

จะเห็นได้ว่าถ้าค่า M มีค่าน้อยกว่าค่า N มากเท่าใด จะทำให้การวิเคราะห์สัญญาณมีความแม่นยำมากเท่านั้น และถ้าค่า M มีค่ามากกว่า N แล้วจะทำให้สัญญาณบางส่วนไม่ถูกใช้ในการวิเคราะห์ จะเกิดความผิดพลาดในการวิเคราะห์สัญญาณในส่วนต่อไป

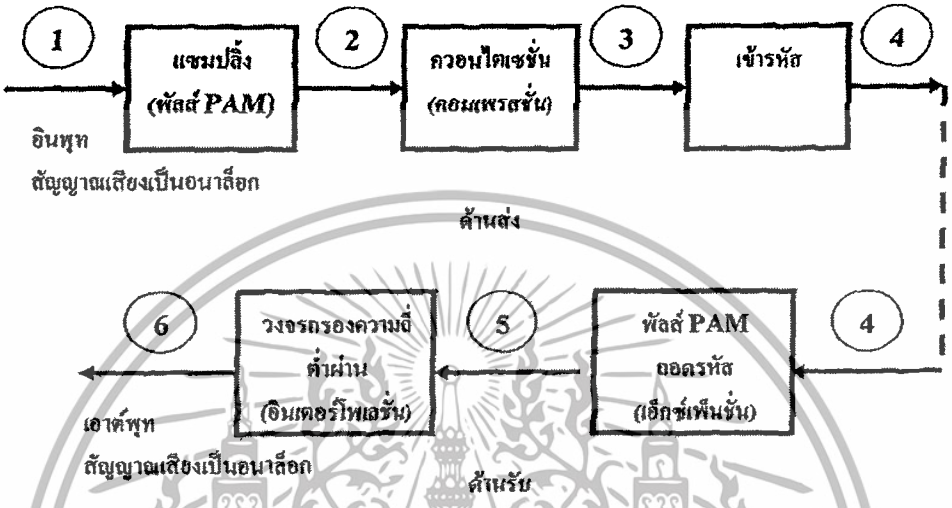
การกำหนดขนาดของช่วงสัญญาณมีเงื่อนไขในการเลือกดังนี้

1. **ค่า M จะต้องสั้นพอที่ทำให้คุณสมบัติของเสียงไม่เปลี่ยนแปลง**
2. **ค่า N จะต้องยาวพอที่จำนวนของตัวอย่างมีเพียงพอต่อการหาสัมประสิทธิ์**
3. **การเลื่อนการวิเคราะห์ (ค่า M) ต้องไม่ข้ามข้อมูล**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 การเข้ารหัสของสัญญาณเสียงแบบพัลส์โค้ดมอดูเลชั่น

วิธีการเข้ารหัสสัญญาณเสียงในระบบสื่อสารแบบดิจิตอลที่ได้รับ การพัฒนา และนำมาใช้อย่างกว้างขวางก็คือ พัลส์โค้ดมอดูเลชั่น (Pulse Code Modulation : PCM) ซึ่งเป็นวิธีการหนึ่งในการเปลี่ยนสัญญาณเสียงจากแอนะลอกให้เป็นดิจิตอล โดยจะประกอบด้วยส่วนประกอบต่างๆ ดังนี้



รูปที่ 2.7 กระบวนการเข้ารหัสและถอดรหัสของระบบพีซีเอ็ม

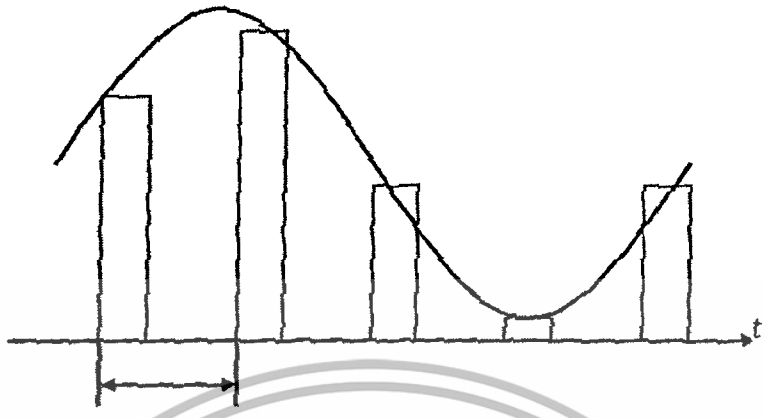
จากรูปที่ 2.7 แสดงขั้นตอนการประมวลผลสัญญาณ เพื่อให้ได้สัญญาณพีซีเอ็ม กล่าวอย่างกว้างๆก็คือ ระบบพีซีเอ็มเป็นการจัดการสัญญาณพัลส์แอมปลิจูดมอดูเลชั่น (Pulse Amplitude Modulation : PAM) โดยการนำสัญญาณพีเอเอ็มไปทำการเข้ารหัส (coding) เป็นสัญญาณดิจิตอล หลังจากนั้นจึงนำสัญญาณดิจิตอลที่ได้นั้นส่งผ่านระบบต่อไปและทางด้านรับก็จะทำการถอดรหัส (decoding) เป็นสัญญาณพีเอเอ็ม แล้วนำไปดีมอดูเลตเพื่อให้ได้สัญญาณเดิมกลับมา

ในการส่งโดยระบบพีซีเอ็มนั้นต้องส่งด้วยจำนวนบิต 8 บิต แต่ยังมีการเข้ารหัสอีกแบบหนึ่งที่ทำการส่งด้วยรหัสน้อยกว่า 8 บิต คือการเข้ารหัสสัญญาณเสียงแบบอแด็พทิฟว์ดิฟเฟอร์เรนเชียลพัลส์โค้ดมอดูเลชั่น ส่งผลให้การและถอดรหัสสัญญาณมีความเร็วมากขึ้น แต่ประสิทธิ์ภาพเท่าเดิม

กระบวนการเข้าและถอดรหัสของระบบพีซีเอ็มสามารถอธิบายองค์ประกอบแต่ละส่วนของระบบได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การชักตัวอย่าง

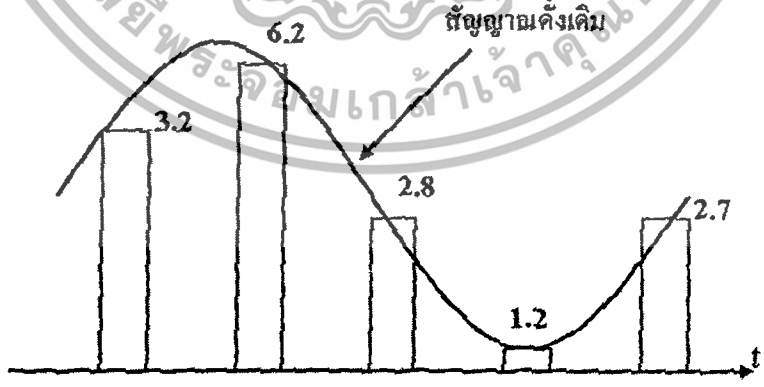


รูปที่ 2.8 การชักตัวอย่างสัญญาณเสียง

การชักตัวอย่าง (sampling) คือการทำสัญญาณซึ่งมีค่าต่อเนื่องให้เป็นแบบดีสครีต ในช่วงเวลาที่เท่าๆกัน ดังรูปที่ 2.8 และจากทฤษฎีการชักตัวอย่าง ถ้าเก็บตัวอย่าง (sample) ด้วยอัตรา 2 เท่าหรือมากกว่าความถี่สูงสุดของสัญญาณแอนะลอกแล้วจะทำให้สัญญาณเดิมกลับคืนมาได้ เนื่องจากสัญญาณเสียงสนทนาที่พบถูกจำกัดให้มีความถี่ระหว่าง 300 - 3400 Hz ดังนั้นอัตราการชักตัวอย่างต่ำสุดจะต้องเท่ากับ 6.8 kHz สำหรับในทางปฏิบัติจะใช้ 8 kHz คือการชักตัวอย่างทุกๆ 125 us

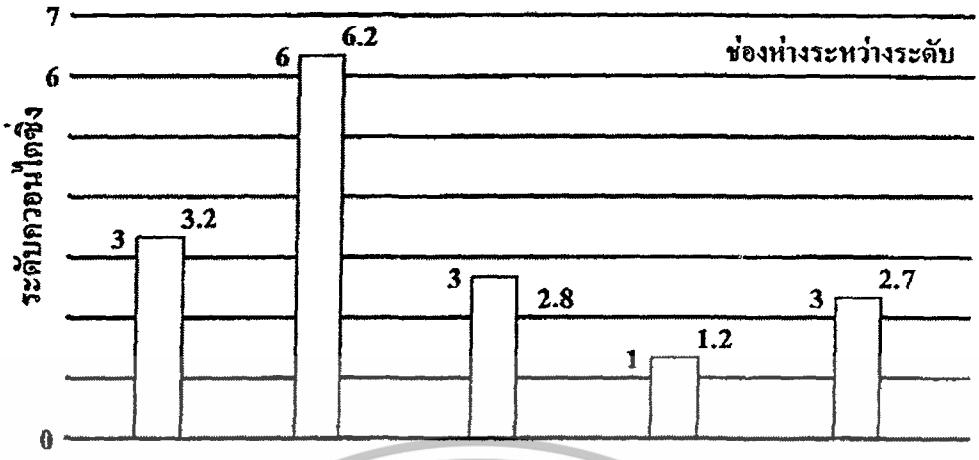
2. การจัดระดับ

ขบวนพัลส์พีเอเอ็มที่ผ่านการชักตัวอย่างมาแล้ว ยังถือว่าเป็น สัญญาณชนิดแอนะลอกอยู่ คือจะมีแอมปลิจูดที่เปลี่ยนแปลงอย่างต่อเนื่องไปกับเวลาที่เป็นช่วง ๆ การจัดระดับคือกระบวนการที่เปลี่ยนแปลงแอมปลิจูดของพัลส์พีเอเอ็มเหล่านั้นให้เป็นตัวเลขแบบดีสครีต ดังแสดงในรูปที่ 2.9



รูปที่ 2.9 สัญญาณแอนะลอกดั่งเดิมก่อนทำการจัดระดับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 สัญญาณดิจิตอลหลังขณะทำการจัดระดับสัญญาณ

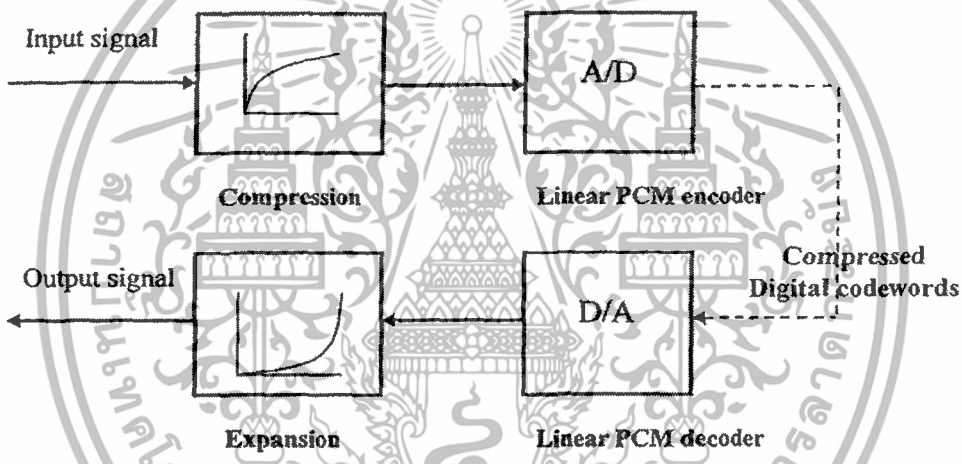
จากรูปที่ 2.10 แอมปลิจูดของการชักตัวอย่างทุกตัวของพีเอเอ็มจะถูกจัดให้เป็นระดับ ซึ่งเรียกว่า ระดับการควอนไตซ์ (quantize level) โดยมีระยะห่างระหว่างระดับข้างเคียง เรียกว่า ควอน-ไตซ์ซิ่งอินเทอร์วัล (quantizing interval) เท่ากัน กรณีนี้เรียกว่าการจัดแบบยูนิฟอร์ม (uniform quantizing) ขนาดของแซมเปิ้ลทุกตัวจะแสดงด้วยระดับการควอนไตซ์ที่ใกล้เคียงที่สุด เช่น ขนาดของแซมเปิ้ลที่ $t = t_1$ คือ 3.2 จะจัดให้เป็นระดับ 3 หรือค่าแซมเปิ้ลที่ $t = t_2$ มีขนาด 6.2 จะจัดให้เป็น 6 เป็นต้น จะเห็นได้ว่าสัญญาณพีเอเอ็มที่ถูกจัดระดับแล้วนี้จะเป็นเพียงค่าโดยประมาณของสัญญาณแอนะลอกเท่านั้น ดังนั้นส่วนเกินและส่วนขาดจากการจัดระดับจึงเป็นค่าผิดพลาดระหว่างสัญญาณเดิมและค่าที่จัดระดับ ซึ่งค่าผิดพลาดนี้เรียกว่า ควอนไตซ์ซิ่งนอยส์ (quantizing noise) หรือค่าความผิดเพี้ยนจากการควอนไตซ์ (quantizing distortion)

จากหลักการที่กล่าวมานี้ ในทางปฏิบัติจะไม่สามารถหลีกเลี่ยงควอนไตซ์ซิ่งนอยส์ได้ แต่เพื่อรักษาคุณภาพเสียงในการสนทนาให้ดี จึงจำเป็นต้องทำการควอนไตซ์ซิ่งนอยส์นี้ให้น้อยลง ในเบื้องต้นคือการลดควอนไตซ์ซิ่งนอยส์อินเทอร์วัลให้แคบลงก็สามารถลดควอนไตซ์ซิ่งนอยส์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. คอมแพนดิ้ง

ตามที่ได้กล่าวมาแล้วว่าเราไม่สามารถ หลีกเลี่ยงควอนไตซ์ซิ่งนอยส์ได้ แต่เราสามารถทำให้มันลดลงได้โดยการควอนไตซ์ซิ่งนอยส์อินเทอร์วัล หรือการเพิ่มจำนวนระดับนั้นเอง แต่เมื่อจำนวนระดับเพิ่มขึ้นแล้วจำนวนบิตที่ใช้ก็เพิ่มขึ้นด้วย จึงจำเป็นต้องใช้ความเร็วในการส่งสัญญาณดิจิตอลสูงขึ้น ตามปกติควอนไตซ์ซิ่งนอยส์จะเกิดขึ้นสม่ำเสมอในทุกอินเทอร์วัลโดยไม่เกี่ยวข้องกับแอมปลิจูดของสัญญาณเดิม ในกรณีที่สัญญาณมีระดับสูง คุณภาพการเข้ารหัสของสัญญาณเสียงจะดีกว่าของสัญญาณซึ่งมีระดับต่ำ ดังนั้นจึงจำเป็นต้องพิจารณาควอนไตซ์ซิ่งนอยส์ในบริเวณ ที่สัญญาณมีระดับต่ำ การจัดแบบนอนยูนิฟอร์ม (non-uniform quantizing) คือบริเวณที่สัญญาณมีแอมปลิจูดต่ำจะควอนไตร์ซิ่งอินเทอร์วัลแคบๆ และในทางตรงกันข้าม บริเวณที่สัญญาณแอมปลิจูดสูงจะใช้ควอนไตซ์ซิ่งนอยส์อินเทอร์วัลกว้างๆ ซึ่งการทำให้เป็นแบบนอนยูนิฟอร์มนั้นจะใช้หลักการคอมแพนดิ้งเข้าช่วย ดังรูปที่ 2.11



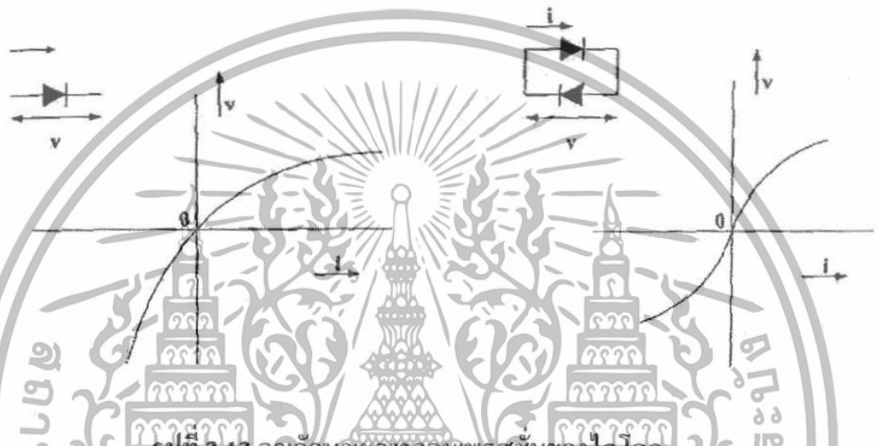
รูปที่ 2.11 การแทรกขั้นตอนของการอัดสัญญาณและการยืดสัญญาณลงในระบบพีซีเอ็ม

การทำการคอมแพนดิ้งเมื่อนำมาใช้ในระบบพีซีเอ็มจะทำให้ประสิทธิ์ภาพในการใช้จำนวนบิตในการเข้ารหัสสูงขึ้น กรณีที่จัดระดับแบบยูนิฟอร์มนั้นจะ ต้องใช้ประมาณ 2,000 ระดับ จึงจะสามารถรักษาคุณภาพของเสียงให้ดี ในการเข้ารหัสต้องใช้ถึง 11 บิตต่อหนึ่งแซมเปิ้ล แต่ถ้าใช้แบบนอนยูนิ-ฟอร์มแล้วจะใช้เพียง 7 บิต ซึ่งจะมีระดับเพียง 128 ระดับเท่านั้น ก็เพียงพอที่จะทำให้อัตราส่วนของสัญญาณต่อการควอนไตซ์ซิ่งนอยส์ใกล้เคียงกับการจัดระดับแบบยูนิฟอร์ม CCITT กำหนดให้ใช้ 8 บิต ต่อแซมเปิ้ล 1 ตัว และระดับการควอนไตซ์เท่ากับ 256 ระดับ ก็จะเป็นการรับรองว่าเสียงพูดจะมีคุณภาพดี

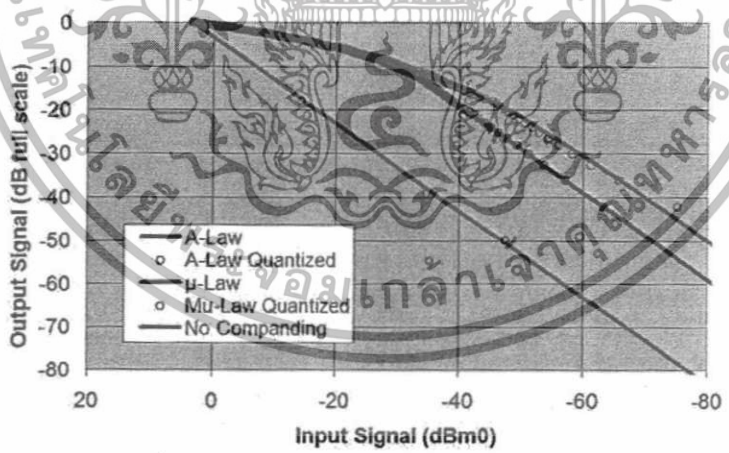
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของการคอมแพรสเซอร์นั้นจะแสดงเป็นแบบลอการิทึม รูปแบบโดยทั่วไปจะใช้คุณสมบัติของ V-I ของไดโอด ตามรูปที่ 2.11 กรณีที่ใช้คอมแพรสเซอร์จะมีกระแส I เป็นอินพุท โวลเตจ V เป็นเอาต์พุท สำหรับกรณีที่ใช้เอกซ์แพนเดอร์จะมีโวลเตจเป็นอินพุท และกระแสเป็นเอาต์พุท

คุณลักษณะของการคอมแพรส์เซอร์ที่ใช้สำหรับการเข้ารหัส สัญญาณเสียงในปัจจุบันคือ $\mu - law$ ซึ่งในทวีปอเมริกาเหนือและญี่ปุ่นเป็นหลัก โดยที่ค่า μ ที่ใช้คือ 255 สำหรับในประเทศยุโรปและประเทศไทยใช้ $A - law$ ซึ่งโดยหลักการจะเหมือนกับกับกฎ $\mu - law$ แต่รายละเอียดจะแตกต่างกัน โดยค่า A ที่ใช้คือ 87.6 คุณลักษณะทั้งสองนี้แสดงไว้ในรูปที่ 2.12 เฉพาะกรณี $\mu = 255$ และ $A = 87.6$



รูปที่ 2.12 คุณลักษณะการคอมเพรสชั่นของไดโอด

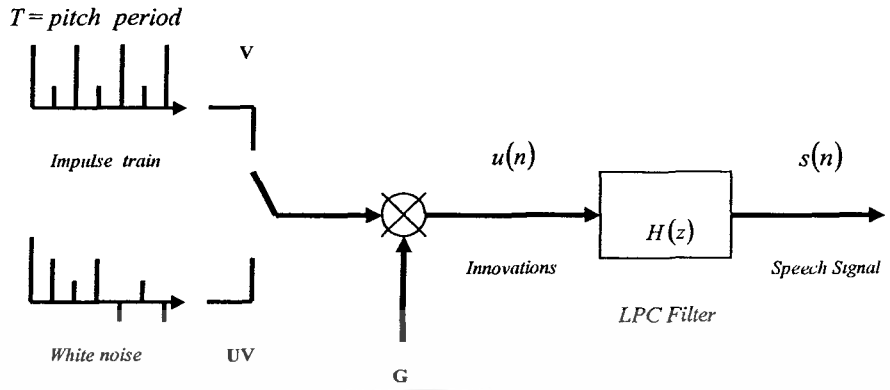


รูปที่ 2.13 แบบอย่างคุณลักษณะของการคอมเพรสชั่น

กรณีที่ $\mu = 100$ จะใช้คอมเพรสเซอร์ตามรูปที่ 2.11 แต่กรณีที่ $\mu = 255$ และ $A = 87.6$ จะใช้วงจรคอมเพรสเซอร์ที่มีคุณลักษณะเป็นเส้นตรงโดยแยกออกเป็นส่วนๆเรียกว่าเซกเมนต์ (segment)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 การสังเคราะห์เสียงเชิงคณิตศาสตร์



รูปที่ 2.14 แสดงการสังเคราะห์สัญญาณเสียง

จากรูปที่ 2.14 แสดงแบบจำลองทางคณิตศาสตร์สำหรับการสังเคราะห์เสียง โดยอธิบายได้ดังนี้

- สัญญาณเสียงที่สังเคราะห์ขึ้นได้จากการนำสัญญาณเสียงที่ มีลักษณะเป็นคาบ (voiced) และสัญญาณเสียงที่ไม่เป็นคาบ (unvoiced) มาทำการมอดูเลชั่น และผ่านวงจรกรอง
- เปรียบเทียบความสัมพันธ์ระหว่างรูปแบบทางกายภาพของเสียงกับรูปแบบทางค ณิตศาสตร์ได้ดังต่อไปนี้

ตารางที่2.2 ความสัมพันธ์ระหว่างรูปแบบทางกายภาพกับรู ปแบบทางคณิตศาสตร์ของสัญญาณเสียง

รูปแบบทางกายภาพ	รูปแบบทางคณิตศาสตร์
ลม (Air)	$u(n)$ (Innovations)
การสั่นของเส้นเสียง (Vocal Cord Vibration)	V (Voiced)
คาบการสั่นของเส้นเสียง (Vocal Cord Vibration Period)	T (Pitch period)
เสียงที่ไม่มีการสั่นของเส้นเสียง (Fricatives and Plosives)	UV (Unvoiced)
ปริมาณของลมที่ออกจากปอด (Air Volume)	G (Gain)

1. การหาสัมประสิทธิ์ของวงจรกรอง

แนวคิดของการหาสัมประสิทธิ์ของวงจรกรอง LP คือการประมาณให้ตัวอย่างของเสียงที่เวลาใดๆ $s(n)$สามารถประมาณค่าได้จากการทำความเป็นไปได้เชิงเส้น (linear combination) ของตัวอย่างสัญญาณเสียงอินพุทก่อนหน้าจำนวน p ตัวอย่าง ดังสมการที่ (2.13)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$s(n) \approx a_1 s(n-1) + a_2 s(n-2) + ...a_p s(n-p) \tag{2.13}$$

ซึ่งสัมประสิทธิ์ของ $a_1, a_2, ..., a_p$ นั้นถือว่ามีค่าคงที่ในช่วงเฟรมที่ทำการวิเคราะห์ เราสามารถแปลงสมการที่ (2.13) จากการประมาณให้เป็นเท่ากับได้โดยเพิ่มสัญญาณกระตุ้นเข้าไปดังสมการที่ (2.14)

$$s(n) = \sum_{i=1}^{p} a_1 s(n-i) + Gu(n) \tag{2.14}$$

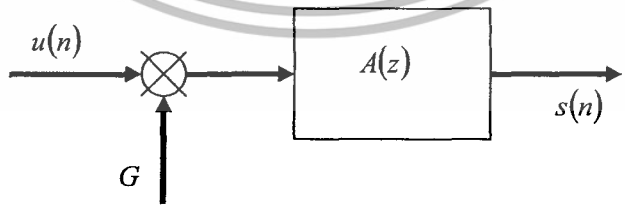
โดยที่ $u(n)$เป็นสัญญาณกระตุ้นที่ถูกทำให้เป็นหนึ่งหน่วย (normalize) แล้ว เมื่อ G เป็นอัตราขยายของสัญญาณกระตุ้น ซึ่งหากพิจารณาสมการดังกล่าวบนโดเมนของ Z จะได้ความสัมพันธ์ดังนี้

$$S(z) = \sum_{i=1}^{p} a_1 z^{-1} S(z) + GU(z) \tag{2.15}$$

ทำให้เราได้ฟังก์ชันถ่ายโอนเป็น

$$H(z) = \frac{S(z)}{GU(z)} = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-i}} = \frac{1}{A(z)} \tag{2.16}$$

ความหมายของสมการนี้แสดงได้ด้วยรูปที่ 2.15 ซึ่งเหล่งกำเนิดของสัญญาณกระตุ้นปล่อยสัญญาณกระตุ้น $u(n)$ออกมา สัญญาณจะถูกขยายด้วยค่า G และกลายเป็นสัญญาณอินพุทที่ป้อนให้กับวงจรกรอง เพื่อสร้างสัญญาณเสียง $s(n)$ออกมา จากความรู้ในปัจจุบันเราพบว่าสัญญาณกระตุ้นของเสียงพูดนั้นเป็นแบบพัลซ์เทรน (pulse train) หรือสัญญาณเสียงที่เป็นคาบ (voiced sound) หรือเป็นสัญญาณแบบสุ่ม (unvoiced sound) ส่วนวงจรกรองนั้นเป็นการทำงานที่ผสมผสานกันของอวัยวะต่างๆที่ใช้ออกเสียง (ช่องคอ, ทางเดินอากาศ, ช่องปาก และลิ้น)



รูปที่ 2.15 บล็อกไดอะแกรมของการเปล่งเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การประมาณความเป็นไปได้เชิงเส้นของสัญญาณเสียงในอดีต โดยนิยามค่าของสัญญาณที่ประมาณจะแสดงด้วยสมการที่ (2.17) ดังนี้

$$\tilde{s}(n) = \sum_{i=1}^{p} a_i s(n-i) \tag{2.17}$$

ค่าความผิดพลาดในการประมาณ $e(n)$แสดงได้โดยสมการที่ (2.18)

$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^{p} a_k (n-k) \tag{2.18}$$

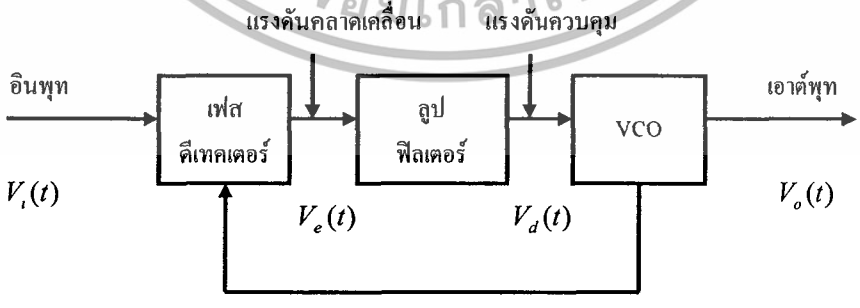
โดยที่มีค่าความผิดพลาดของฟังก์ชันถ่ายโอน (error transfer function) ดังสมการที่ (2.19)

$$A(z) = \frac{E(z)}{S(z)} = 1 - \sum_{k=1}^{p} a_k (n-k) \tag{2.19}$$

ดังนั้นเมื่อให้ $s(n)$เป็นสัญญาณจริง ที่ถูกสร้างขึ้นมาจากระบบเชิงเส้นตามสมการที่ (2.14) จะพบว่า $e(n)$จะมีค่าเท่ากับสัญญาณกระตุ้นที่ถูกขยายแล้ว $Gu(n)$

2.1.6 เฟสล็อกลูป

เฟสล็อกลูปเป็นระบบป้อนกลับ (feedback system) ที่บังคับให้วงจรออสซิลเลเตอร์มีความถี่หรือเฟสเปลี่ยนแปลงไปตามความถี่ หรือเฟสของสัญญาณอ้างอิงภายนอก เฟสล็อกลูปประกอบ ด้วยภาคสำคัญ 3 ภาค คือ ภาคเทียบเฟสหรือเฟสดีเทคเตอร์ ภาคลูปฟิลเตอร์ และภาค VCO (Voltage Controlled Oscillator) ดังรูปที่ 2.16



รูปที่ 2.16 แผนผังของเฟสล็อกลูป

72869
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมุติว่าสัญญาณความถี่อ้างอิงภายนอกเป็นสัญญาณรายคาบมาที่อินพุท ภาคเทียบเฟสทำหน้าที่เปรียบเทียบเฟสระหว่างสัญญาณอ้างอิงกับสัญญาณจาก VCO เอาต์พุทที่ได้จากภาคเฟสดีเทคเตอร์จะเป็นแรงดันที่มีแอมปลิจูดเป็นสัดส่วนกับผลต่างในเฟสของสัญญาณทั้งสอง ที่ทำการเปรียบเทียบแรงดันผลต่างนี้ป้อนไปให้วงจรลูปฟิลเตอร์ซึ่งเป็นฟิลเตอร์ชนิงกรองความถี่ต่ำ ผ่านเอาแต่เฉพาะความถี่ต่างๆที่ต้องการ เพื่อส่งไปควบคุมการออสซิลเลเตอร์ของ VCO ต่อไป

เมื่อลูปอยู่ในสภาวะล็อก (lock) ความถี่ของ VCO จะเท่ากับความถี่ของสัญญาณอินพุทพอดี อาจจะมีเฟลแตกต่างไป แต่ค่าเฟสที่แตกต่างจะมีค่าคงที่ (constant phase difference) ในกรณีที่มีเฟสไม่ตรงกันภาคเฟสดีเทคเตอร์จะจ่ายแรงดันคลาดเคลื่อน (error voltage) ไปควบคุมการทำงานของ VCO เพื่อมิให้เฟสคลาดเคลื่อนจนกว่าจะเข้าสู่สภาวะล็อก เอาต์พุทของ VCO จึงมีแอมปลิจูดคงที่เสมอ แต่ความถี่จะเปลี่ยนแปลงตามความถี่ของสัญญาณอินพุท

จากที่ได้กล่าวมาแล้ว เราอาจจะอธิบายการทำงานในลักษณะสมการทางไฟฟ้าได้อีกดังนี้ คือ เมื่อไม่มีสัญญาณเข้า ($V_i = 0$) แรงดัน V_d และ V_i (ที่ใช้ควบคุมVCO) จะเป็นศูนย์ VCO จะกำเนิดสัญญาณความถี่หนึ่งเรียกว่า ความถี่อิสระ (free running frequency) f_∞ เมื่อใส่สัญญาณเข้าที่ V_i ที่วงจรกรองความถี่ต่ำผ่าน แรงดัน V_d ที่ใช้ควบคุม VCO จะยังคงเป็นศูนย์ และ VCO ยังคงทำงานที่ความถี่ f_∞ ถ้า f_i ไม่ต่างจาก f_∞ มากนัก นั่นคือ $|f_i - f_\infty| \le \Delta f_p$ ($2\Delta f_p$ มีชื่อว่าช่วงดึงเข้าหรือ pull-in range) สัญญาณ V_e จะมีลักษณะไม่สมมาตร กล่าวคือมีองค์ประกอบไฟตรงไม่เท่ากับ 0 V_d จะค่อยๆเพิ่มค่าขึ้นและหลังจากช่วงเวลาหนึ่งเรียกว่า เวลาดึงเข้า (pull-in time) ความถี่ f_d ที่ค่อยๆปรับไปยังค่า f_i จะต่างจาก f_i เพียง $|f_i - f_\infty| = \Delta f_p$ ($2\Delta f_p$ มีชื่อว่าช่วงจับหรือ capture range) ถึงตอนนี้ f_o จะแปรเข้าสู่ค่า f_i อย่างรวดเร็วฉับพลัน เรียกว่าเกิดการล็อก V_d ก็จะเปลี่ยนแปลงฉับพลันเช่นกัน ทั้งนี้สำหรับ VCO โดยทั่วไปความถี่ f_o เป็นปฏิภาคกับการบ่ายเบนจากความถี่อิสระ ดังสมการที่(2.20) นั่นคือ

$$f_0 - f_\infty = k_0 V_d k_0 V_d \qquad (2.20)$$

โดยที่ค่า k_0 เรียกว่าความไวของ VCO มีหน่วยเป็น $\frac{rad/s}{V}$

การได้มาซึ่งการล็อกอาจใช้วิธีได้ดังนี้

1. ณ ขณะหนึ่ง ความถี่ของ VCO ต่างจากความถี่ขาเข้า f_i น้อยกว่า Δf_c จะเกิดการล็อกโดยฉับพลันเรียกว่า เกิดการจับ (capture)
2. ณ ขณะหนึ่ง ความถี่ของVCO ต่างจากความถี่ขาเข้า f_i มากกว่า Δf_c แต่น้อยกว่า Δf_p ความถี่ของ VCO จะค่อยเลื่อนเข้าหาความถี่ f_i เรียกว่า เกิดการดึงเข้า (pull-In)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. นอกช่วงดึงเข้าหรือในกรณีที่การดึงเข้าใช้เวลานานเกินไป อาจเพิ่มวงจรเพื่อควบคุมให้ความถี่ของ VCO กวาดไปเพื่อเสาะหาความถี่ของสัญญาณ หรือ ถ้าเสียงรบกวนน้อย ก็อาจเพิ่มแถบความถี่ของรอบ (tracking) ความถี่ของ VCO จะสามารถตามรอยความถี่ f_i ได้ตราบใดที่ $|f_i - f_\infty| = 2\Delta f_H$ ($2\Delta f_H$ มีชื่อว่าช่วงคงไว้หรือช่วงล็อก)

เราสามารถนำเฟสล็อกลูปไปใช้ในการสังเคราะห์ หรือผลิตความถี่ที่มีความเที่ยงตรง และมีเสถียรภาพเทียบเท่าสัญญาณอ้างอิงได้ วงจรนี้เรียกว่า วงจรสังเคราะห์ความถี่ ระบบสังเคราะห์ความถี่จะช่วยให้เราสามารถสังเคราะห์สัญญาณเอาต์พุท จาก VCO ให้มีความถี่ตามต้องการได้หลายความถี่โดยมีความเที่ยงตรงและเสถียรสูงเทียบเท่าคริสตอลออสซิลเลเตอร์

1. คุณสมบัติของวงจรสังเคราะห์ความถี่

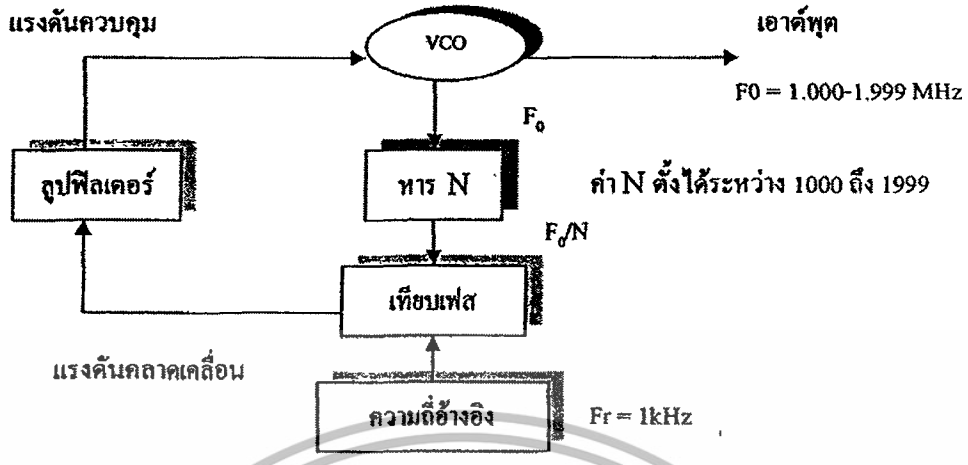
นอกจากวงจรสังเคราะห์ความถี่จะต้องมีคุณสมบัติเกี่ยวกับช่วงความถี่ (frequency range) ที่ต้องผลิตและเรโซลูชั่นระหว่างขั้นแล้ว คุณสมบัติอื่นๆของวงจรสังเคราะห์ความถี่ก็มีความสำคัญเช่นกัน

โดยปกติวงจรสังเคราะห์ความถี่จะสามารถ กำเนิดสัญญาณเพียงสัญญาณเดียวแต่เลือกความถี่ได้หลายค่า ในช่วงความถี่ใช้งาน และมีความละเอียดของความถี่ขึ้นอยู่กับเรโซลูชั่น ในกรณีที่เราเปลี่ยนหลายค่า ในช่วงความถี่ใช้งาน และมีรายละเอียดของความถี่ขึ้นอยู่กับเรโซลูชั่น ในกรณีที่เราเปลี่ยนความถี่จากค่าหนึ่งไปยังอีกค่าหนึ่ง วงจรสังเคราะห์ความถี่จะต้องเปลี่ยนตามได้เร็วทันที กล่าวอีกอย่างหนึ่งคือ ล็อกความถี่ได้ในเวลาอันรวดเร็ว นั่นคือ ช่วงเวลาล็อก (lock-up time) สั้น คุณสมบัติการล็อกความถี่ใหม่ได้รวดเร็วนั้นมีความจำเป็นอย่างยิ่ง

วงจรสังเคราะห์ความถี่ที่ดี จะต้องสามารถผลิตสัญญาณความถี่เดี่ยว โดยปราศจากความถี่แปลกปลอมต่างๆ คุณสมบัตินี้เรียกว่า ความบริสุทธิ์ของสเปกตรัม (spectrum purity) นั่นคือความถี่ฮาร์มอนิกและสปิวเรียสต่างๆนั้น จะต้องถูกกำจัดออกให้เหลือน้อยที่สุด นอกจากนี้นอยส์จากวงจรออสซิลเลเตอร์จะทำให้วงจรสังเคราะห์ความถี่ไม่บริสุทธิ์ ไม่ใช่เพียงความถี่เดียว ในช่วงใกล้เคียงกับความถี่ที่ต้องการ นอยส์ดังกล่าวนี้เรียกว่า เฟสนอยส์ (phase noise)

ความเที่ยงตรง (accuracy) และเสถียรภาพ (stability) ทางความถี่ของวงจรสังเคราะห์ความถี่ขึ้นอยู่กับสัญญาณอ้างอิง โดยทั่วไปสัญญาณ อ้างอิงมักจะเป็นวงจรออสซิลเลเตอร์ชนิดใช้แร่บังคับความถี่ ฉะนั้นวงจรสังเคราะห์ความถี่จะมีเสถียรภาพ และความเที่ยงตรงเทียบเท่ากับคริสตอลออสซิลเลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 แผนผังของหน่วยสังเคราะห์ความถี่

2. การใช้เฟลล็อกลูปในการสังเคราะห์ความถี่

แม้ว่าระบบสังเคราะห์ความถี่จะมีความซับซ้อนเพียงใด เมื่อพิจารณาลึกลงไปแล้วจะพบว่ามีเฟสล็อกลูปเป็นหัวใจในการสังเคราะห์ความถี่เสมอ รูปที่ 2.17 เป็นตัวอย่างของระบบสังเคราะห์ความถี่อย่างง่าย ประกอบด้วย 5 ภาค คือ

1. VCO เป็นออสซิลเลเตอร์กำเนิดสัญญาณเอาต์พุทของระบบสังเคราะห์ความถี่
2. ภาคหารความถี่อ้างอิงคริสตอลจากออสซิลเลเตอร์หรือสัญญาณอื่นๆ
3. ภาคเทียบเฟส
4. ภาคความถี่อ้างอิง
5. ภาคฟิลเตอร์ซึ่งทำหน้าที่กรองเอาเฉพาะความถี่ต่ำไปใช้

ภาคเฟสดีเทคเตอร์

จากรูปจะเห็นว่าสัญญาณอินพุทของภาคเทียบเฟสจาก 2 แหล่งคือ

1. จาก VCO มีความถี่เท่ากับ F_o / N
2. จากสัญญาณอ้างอิงมีความถี่เท่ากับ F_o

เอาต์พุทที่ได้จากการเปรียบเทียบเฟสก็คือ ผลต่างระหว่างสัญญาณ F/N กับ F ซึ่งจะกรองเฉพาะความถี่ต่ำเท่านั้น เพื่อบังคับการออสซิลเลตของวงจร VCO ให้ทำการปรับแก้ความถี่ (หรือเฟส) ให้ตรงจนกว่าความถี่ของสัญญาณทั้งสองจะเท่ากันในสภาวะล็อกความถี่ของ VCO เพื่อผ่านวงจรหาร จะเท่ากับความถี่อ้างอิง นั่นคือ

$$F = N \times F_r \tag{2.21}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคออสซิลเลเตอร์แบบควบคุมความถี่ด้วยระดับแรงดัน (V_{cc})

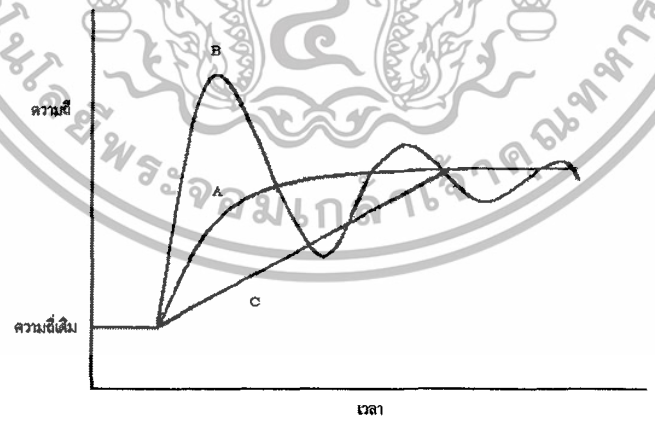
วงจร VCO นับว่ามีความสำคัญมากในระบบเฟสล็อกลูป เพราะจะเป็นตัวกำหนดเสถียรภาพเชิงความถี่ จะต้องมีคุณสมบัติต่อไปนี้คือ

1. การแปลแรงดันเป็นความถี่มีลักษณะเป็นเชิงเส้น
2. มีเสถียรภาพเชิงความถี่
3. ทำงานได้ที่ความถี่สูง
4. ช่วงตามรอยกว้าง
5. ปรับคลื่นได้ง่าย

คุณสมบัติเหล่านี้มักไม่พบใน VCO ที่เป็นตัวเดียวกัน ดั้งนั้นเราต้องเลือกชนิดของ VCO ตามคุณสมบัติที่เราต้องการ

ภาคลูปฟิลเตอร์

เป็นวงจรฟิลเตอร์ชนิดกรองความถี่ต่ำผ่านทำหน้าที่กรองเฉพาะความถี่ต่ำมาควบคุมความถี่ของ VCO และในที่นี้เราใช้ลูปฟิลเตอร์ประเภทพาสซีป ลูปฟิลเตอร์นี้เป็นตัวกำหนดคุณสมบัติการเปลี่ยนแปลงความถี่ก่อนเข้าสู่สภาวะล็อกที่เรียกว่า คุณสมบัติชั่วครู่ ถ้าเลือกอัตราขยายลูปและค่าคงตัวเวลาของลูปไม่เหมาะสม ความถี่ของเฟสล็อกลูปจะไม่ล็อกและจะเปลี่ยนไปเปลี่ยนมา ดังนั้น ค่าคงตัวเวลาของลูปฟิลเตอร์จะต้องไม่มากเกินไป เพื่อว่าทุกครั้งที่เปลี่ยนความถี่เฟสล็อกลูปจะล็อกได้เร็ว โดยไม่มีการสะบัดหรือใช้เวลาเปลี่ยนความถี่อย่างรวดเร็ว แต่ค่าคงตัวเวลาก็ไม่ควรจะน้อยเกินไปจนกระทั่งความถี่สั่นหรือไม่นิ่ง



รูปที่ 2.18 คุณลักษณะ ช่วงพลวัต (dynamic range) ในการเปลี่ยนความถี่เฟสล็อกลูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.18 ซึ่งแสดงการเปลี่ยนความถี่ของ VCO จะเห็นว่าเส้นทางการเปลี่ยนแปลงแรงดันมี 3 เส้นทาง เส้นทาง A เป็นเส้นทางหน่วงวิกฤต (critically damped) ใช้เวลาในการเปลี่ยนสู่ความถี่ใหม่น้อยที่สุด เส้นทาง B เรียกว่า เส้นทางหน่วงขาด (undamped) มีการสะบัด (หรือ ออสซิลเลต) เนื่องจากจากโอเวอร์ชู้ต เส้นทาง C เป็นเส้นทางหน่วงเกิน (overdamped) ไม่มีโอเวอร์ชู้ตแต่เวลาที่ใช้ในการเข้าสู่ความถี่ใหม่จะช้า เส้นทาง A เป็นเส้นทางที่ดีที่สุดในการออกแบบค่าคงตัวเวลาของลูปฟิลเตอร์ เพราะใช้เวลาเปลี่ยนความถี่เร็ว และไม่มีโอเวอร์ชู้ต

ภาคหาร N

ภาคหาร N (programmable divider) เวลาที่ใช้ในการล็อกความถี่เมื่อมีค่าน้อยที่สุดจะไม่เท่ากับเมื่อมีค่ามากที่สุด วงจรหารเกิดจากวงจรนับฐานสิบ (decade counter) หลายชุดมาต่อร่วมกับเกตต่างๆ เพื่อให้สามารถเลือกสั่งให้วงจรนับทำหน้าที่หารความถี่ได้ตามตัวเลขที่ตั้งไว้วงจรหาร N นี้เป็นตัวรับคำสั่งเกี่ยวกับความถี่ไปควบคุม VCO เพื่อให้กำเนิดสัญญาณตามที่ต้องการ ตัว N จะเป็นตัวที่กำหนดย่านความถี่และจำนวนช่องความถี่

ลักษณะการป้อนข้อมูล N ให้กับวงจรหาร N แบบขนาน (parallel) คือข้อมูลแต่ละบิตจะป้อนเข้าพร้อมๆกัน ส่วนลักษณะการป้อนข้อมูล N ให้กับวงจรหาร N แบบอนุกรมแต่ละบิตจะป้อนเข้าไม่พร้อมกัน วงจรหารประเภทนี้มีความซับซ้อนมากกว่า เพราะต้องมีสัญญาณนาฬิกา มีวงจรแลตช์ ในการป้อนข้อมูลวงจรหาร N ประเภทนี้จะควบคุมการทำงานด้วยไมโครคอมพิวเตอร์

ปัญหาสำคัญของซินธิไซเซอร์อีกอย่างหนึ่งก็คือ วงจรหาร N (หรือวงจรหารที่โปรแกรมได้) ไม่สามารถทำงานที่ความถี่สูงกว่า 25 MHz ได้ ฉะนั้นเราจึงต้องหาทางลดทอนความถี่ที่ป้อนแก่วงจรหาร N ลง เพื่อให้วงจรลอจิกของวงจรหาร N ทำงานได้ วิธีต่างๆที่นิยมใช้ได้แก่ ใช้ความถี่จากออสซิลเลเตอร์พิเศษ (บางครั้งเรียกว่า ออสซิลเลเตอร์ PLL) มามิกซ์กับ VCO ให้ความถี่ลดลงก่อนที่จะป้อนให้แก่วงจรหาร

ภาคกำเนิดความถี่อ้างอิง

ภาคกำเนิดความถี่อ้างอิงนิยมใช้คริสตอลออสซิลเลเตอร์ ความถี่ออสซิลเลเตอร์อ้างอิงนี้เป็นตัวกำหนดเรโซลูชั่นและเสถียรภาพของความถี่อ้างอิงที่ดี จึงทำให้สามารถสังเคราะห์ความถี่ที่มีเสถียรภาพที่ดีด้วย

3. การดีมอดูเลตสัญญาณเอฟเอ็มด้วยเฟสล็อกลูป

เฟสล็อกลูปเป็นระบบป้อนกลับ ประกอบด้วยเครื่องกำเนิดสัญญาณที่มีความถี่เปลี่ยนแปลงตามระดับแรงดันขาเข้า (VCO) และส่วนเปรียบเทียบเฟส (phase comparator)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณเข้าจะเปรียบเทียบเฟสกับสัญญาณอ้างอิง (reference signal) ที่กำเนิดจาก VCO ส่วนเปรียบเทียบเฟสจะให้แรงดันขาออกที่มีระดับขึ้นกับความแตกต่างระหว่างเฟสของสัญญาณเข้าทั้งสอง เมื่อผ่านเครื่องกรองความถี่ต่ำผ่านสัญญาณขาออกบางส่วนจะถูกป้อนกลับเพื่อควบคุมการกำเนิดความถี่ของ VCO จนกระทั่งความถี่ทั้งสองสัญญาณเท่ากัน แรงดันที่ป้อ นกลับจะคงที่และทำให้ความถี่ทั้งสองเกาะกันไป โดยมีเฟสต่างกันที่ค่าคงที่ความถี่กลาง (center frequency หรือ free-running frequency)

ต่อไปนี้จะอธิบายส่วนประกอบที่สำคัญของเฟสล็อกลูป

ส่วนเปรียบเทียบเฟส มีอยู่ 2 ชนิดคือ ชนิดแอนะลอก และชนิดดิจิตอล ชนิดแอนะลอกสร้างขึ้นโดยใช้ตัวผสม (mixer) หรือตัวคูณ (multiplier) ดังนี้

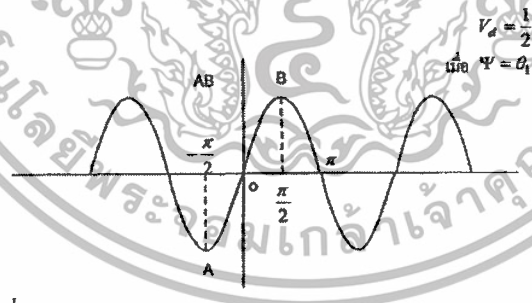
$$V_1(t) = A\sin(\omega_1 t + \theta_1) \tag{2.22}$$

$$V_2(t) = A\sin(\omega_2 t + \theta_2) \tag{2.23}$$

โดย $V_1(t)$ และ $V_2(t)$ เป็นสัญญาณขาเข้า ซึ่งจะได้สัญญาณขาออก $V_d(t)$ ของการคูณดังนี้

$$V_d(t) = \frac{1}{2}AB\cos[(\omega_1 + \omega_2)t + (\theta_1 + \theta_2)] + \frac{1}{2}AB\cos[(\omega_1 - \omega_2)t + (\theta_1 - \theta_2)] \tag{2.24}$$

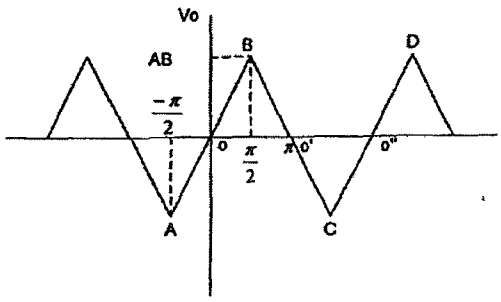
เมื่อผ่านเครื่องกรองความถี่ต่ำผ่านก็จะได้สัญญาณขาออก ที่ขึ้นอยู่กับความแตกต่างของเฟส ซึ่งมีลักษณะสมบัติ ดังรูปที่ 2.19



รูปที่ 2.19 ลักษณะสมบัติของส่วนเปรียบเทียบเฟสชนิดแอนะลอก

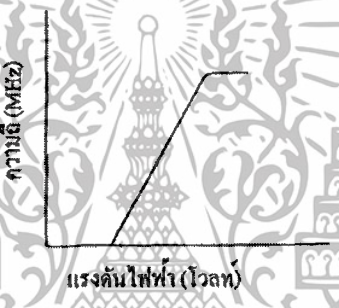
สำหรับอุปกรณ์เปรียบเทียบชนิดดิจิตอลนั้น สร้างขึ้นได้หลายวิธี เช่น ใช้ เอ็กคลูซีฟ-ออร์ หรืออุปกรณ์ชนิดมีความจำ เช่น ใช้ ฟลิป-ฟลอป ซึ่งจะมีลักษณะสมบัติเช่นเดียวกับแบบ แอนะลอก แต่จะต่างกันที่เป็นเส้นตรง ดังรูปที่ 2.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ลักษณะสมบัติ ส่วนเปรียบเทียบเฟสชนิดดิจิตอล

VCO สามารถสร้างขึ้นได้หลายวิธี ไดโอดคาปาซิแตนซ์ชนิดปรับค่าแรงดัน (voltage variable capacitance diode) ซึ่งจะรู้จักกันดีในชื่อ วาแรคเตอร์ (varactor) หรือ มัลติไวเบรเตอร์ (multivibrator) ชนิดต่างๆ ที่สามารถควบคุมความถี่ได้ด้วยแรงดัน รูปที่ 2.21 เป็นลักษณะสมบัติทั่วไปของ VCO



รูปที่ 2.21 ลักษณะสมบัติของ VCO

สมมุติให้สัญญาณเอฟเอ็มเข้าที่ ส่วนเปรียบเทียบเฟส คือ

$$A\cos[\omega_c t + \psi(t)]$$

ให้ความไวทางความถี่ (Frequency Sensitivity) ของ VCO คือ K_0 rad/sec คือค่าของความแตกต่างหรือการเปลี่ยนแปลงของความถี่เชิงมุมขณะใดขณะหนึ่ง ω_1 ที่เกิดจากการเปลี่ยนแปลงแรงดัน V ที่ขาเข้าของ VCO ดังสมการที่ (2.25) นั่นคือ

$$K_0 = \frac{d\omega_1}{dv} \tag{2.25}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนี้จะกระตุ้นให้เกิดสัญญาณจากเครื่องสั่นสะเทือนความถี่ (Oscillator) ใน VCO ดังสมการที่ (2.26)

$$B\cos[\omega_c t + K_0 \int_{-\infty}^{t} V_0(\lambda)d\lambda] \tag{2.26}$$

ถ้าให้สัญญาณขาเข้าและความถี่ของ VCO มีค่าสมดุลเท่ากัน จะได้สมการที่ (2.27)

$$\frac{d\omega(t)}{dt} = \frac{d}{dt} K_0 \int_{-\infty}^{t} V_0(\lambda)d\lambda \tag{2.27}$$

กำหนดให้ $\frac{d\psi(t)}{dt} = \omega$ จะได้สมการที่ (2.28) นั่นคือ

$$V_0(t) = \frac{\omega}{K_0} \tag{2.28}$$

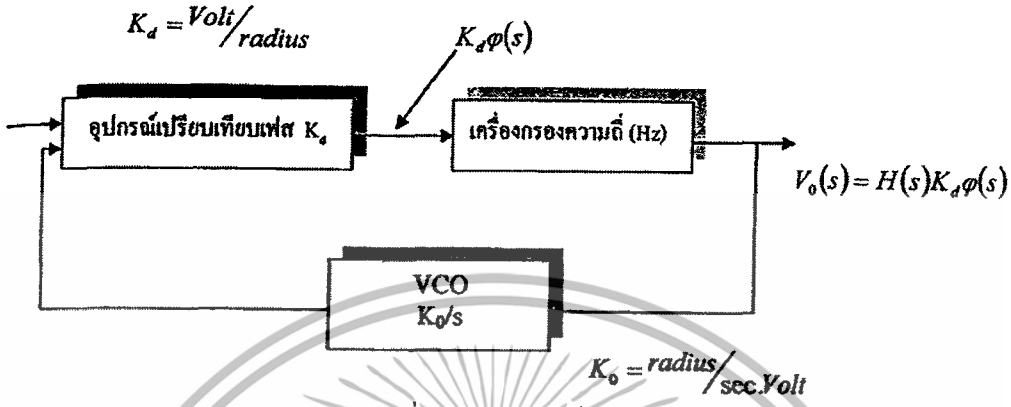
จะเห็นได้ว่าสัญญาณขาออกของ VCO จะผันแปรเป็นสัดส่วนโดยตรงกับการเปลี่ยนแปลงความถี่ที่ต้องการใน เครื่องดีมอดูเลตเอฟเอ็ม ถ้าความถี่ของสัญญาณขาเข้าเปลี่ยนอย่างต่อเนื่องที่อัตราช้าพอควรที่จะให้เฟสล็อกลูปทำงานในสภาพสมดุลได้ สัญญาณขาออกของเฟสล็อกลูปก็จะผันแปรเป็นสัดส่วนโดยตรงอย่างต่อเนื่องกับการเปลี่ยนแปลงของตัวพา นั่นคือ สัญญาณเอฟเอ็มสามารถดีมอดูเลตด้วยเฟสล็อกลูปตามต้องการได้

จากลักษณะสมบัติของ VCO และส่วนเปรียบเทียบเฟส จะพบว่าทั้ง VCO และเฟสมีขอบเขตจำกัดในการทำงาน กล่าวคือ VCO จะมีความสามารถในการกำเนิดความถี่ได้ภายในช่วงจำกัดช่วงหนึ่งและเฟสคอมพาราเตอร์จะให้ระดับแรงดันที่แตกต่างกันได้ เฉพาะช่วงการทำงานที่มีค่าของมุมแตกต่างกันอยู่ในระหว่าง 0–180 องศา ดังนั้นความถี่ที่จะถูกดักจับได้นั้นจึงอยู่ในขอบเขตจำกัด ช่วงความถี่ของสัญญาณขาเข้าซึ่งเฟสล็อกลูปจึงสามารถดักจับตามสัญญาณได้ นับจากเวลาเริ่มต้นที่ดักจับได้เรียกว่า ช่วงล็อก (locked range) ส่วนช่วงความถี่ของสัญญาณขาเข้าซึ่งเฟสล็อกลูปสามารถดักจับได้จากขณะเริ่มต้นความถี่ที่ยังไม่เกาะกัน ดังนั้นช่วงที่ยัง ไม่เกาะกันจะกว้างกว่า**ช่วงที่ดักจับสัญญาณ**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ทรานเฟอร์ฟังชั่นของลูป

สมมุติให้เป็นกรณีที่ขั้วออกของลูปอยู่ที่ขั้วออกของเครื่องกรองความถี่



รูปที่ 2.22 วงจรเฟสล็อกลูป

จากรูปที่ 2.22 ได้แทนคุณลักษณะของส่วนประกอบต่างๆ ในรูปที่ ด้วยฟังชั่น ถ่ายโอนของแต่ละหน่วย ดังนั้นความสัมพันธ์ระหว่างสัญญาณขาเข้าและขาออกเป็นดัง สมการที่ (2.29)

$$\frac{V_0(s)}{\phi(s)} = \frac{K_d K_f(s)}{s + K_d K_0 K_f(s)} \tag{2.29}$$

จากสมการ ออเดอร์ของเฟสล็อกลูปจะขึ้นอยู่กับ ออเดอร์ของเครื่องกรองความถี่ ดังนั้นเสถียรภาพของลูปจะขึ้นอยู่กับการพิจารณาเลือกใช้ลักษณะสมบัติของเครื่องกรองความถี่ กรณีที่เครื่องกรองความถี่เป็นลำดับที่หนึ่ง (First Order) ซึ่งเขียนในรูปทั่วไปดังสมการที่ (2.30)

$$K_f(s) = 1 + \frac{a}{s} \tag{2.30}$$

จะได้ฟังชั่นถ่ายโอนของระบบที่เป็นลำดับที่สอง (Second-order) ดังสมการที่ (2.31)

$$\frac{V_0(s)}{\phi(s)} = \frac{K_d(s+a)}{s^2 + K_d K_0 s + K_d K_0 a} \tag{2.31}$$

และเขียนเป็นรูปแบบทั่วไป ตามหลักวิชาวิศวกรรมระบบควบคุม ได้ ดังสมการที่ (2.32)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{V_0(s)}{\phi(s)} = \frac{2\xi\omega_n s + \omega_n^2 s}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{2.32}$$

ในที่นี้ ξ คือ ปัจจัยที่ทำให้เกิดการหน่วง (damping factor) และ ω_n คือ ความถี่ธรรมชาติ (Natural Frequency)

5. ข้อดีและข้อเสียของวงจรเฟสล็อกลูป

วงจรเฟสล็อกลูปนี้มีข้อดีอยู่หลายประการสำหรับการใช้งานบางอย่าง ซึ่งไม่มีวงจรอื่นที่จะเทียบเท่า อย่างไรก็ดีในการใช้งานบางอย่างนั้นก็อาจใช้วงจรกรองแบบ LC หรือ RC แทนได้ ข้อดีของวงจรเฟสล็อกลูปเมื่อเทียบกับวงจรกรองแบบ RC ได้แก่

1. ทำงานได้ดีที่ความถี่สูง สามารถทำงานได้ที่ความถี่สูงกว่า 100 MHz ส่วนวงจรกรองแบบทำได้แค่ความถี่ประมาณ 10 MHz
2. อำนาจการเลือกสรร (selectivity) และความถี่กลางไม่ขึ้นต่อกัน กล่าวคือความถี่กลางกำหนดโดยความถี่อิสระของ VCO ส่วนอำนาจการเลือกสรรขึ้นอยู่กับลักษณะของวงจรกรองความถี่ต่ำผ่าน จึงไม่มีปัญหาการเรียงคลื่น (alignment)
3. องค์ประกอบภายนอกวงจรประมวลมีน้อยและปรับคลื่นได้ง่าย โดยทั่วไปความถี่อิสระของ VCO กำหนดโดย C ตัวเดียว หรือ C กับ R ซึ่งจะปรับคลื่นได้ตั้งแต่ค่าต่ำกว่า 1 kHz ถึง 100 MHz

อย่างไรก็ดีวงจรเฟสล็อกลูปก็มีข้อเสียเช่นกัน ได้แก่

1. ไม่ให้ข้อมูลเกี่ยวกับช่วงสูงวงจรเฟสล็อกลูปสนองตอบต่อความถี่เท่านั้นตราบเท่าที่สัญญาณเข้ามีขนาดใหญ่ที่จะทำให้เกิดการล็อกวงจรจะไม่สนองตอบต่อช่วงสูงของสัญญาณเข้า
2. สนองตอบต่อฮาร์มอนิกหรือฮาร์มอนิกย่อยของสัญญาณเข้าทำให้การชดสัญญาณแทรกแซงที่มีความถี่เป็นอัตราส่วนจำนวนเต็มกับสัญญาณไม่ สู้ได้ผลนัก
3. ขาดเทคนิคการสังเคราะห์ เนื่องจากลักษณะการจับของวงจรเฟสล็อกลูปเป็นแบบไม่เชิงเส้น เทคนิคการสังเคราะห์ให้ได้วงจรที่มีลักษณะเชิงความถี่ตามข้อกำหนดจึงเป็นเรื่องที่ยากมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 สถาปัตยกรรมไมโครคอนโทรลเลอร์ dsPIC30F6014

สำหรับโครงงานนี้ได้เลือกใช้ไมโครคอนโทรลเลอร์ dsPIC30F6014 ในการประมวลผลสัญญาณเสียง dsPIC คือไมโครคอนโทรลเลอร์ 16 บิต และไมโครคอนโทรลเลอร์นี้ ถูกจัดให้อยู่ในอนุกรม DSC (Digital Signal Controller) ซึ่งเป็นไมโครคอนโทรลเลอร์ที่ได้รับการออกแบบมาเป็นพิเศษเพื่องานประมวลผลสัญญาณดิจิตอลสำหรับสร้างระบบควบคุมอัตโนมัติที่มีความสามารถสูง และมีโมดูลพิเศษต่างๆที่ช่วยในการประมวลผลทางด้านเสียง

2.2.1 คุณสมบัติโดยรวมของ dsPIC30F6014

ในหัวข้อนี้จะขออธิบายถึงคุณสมบัติและความสามารถต่างๆภายในไมโครโทรลเลอร์ รวมถึง โมดูลพิเศษที่ใช้สำหรับเชื่อมต่อกับ อุปกรณ์ภายนอกในการทำโครงงานนี้

1. คุณสมบัติของซีพียู

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ ซีพียู แบบ RISC (Reduced Instruction Set Computer : RISC) ถูกออกแบบมาให้มีคำสั่งในการทำงานจำนวนน้อย ทำให้โครงสร้างภายในไม่ซับซ้อน สามารถประมวลผลได้อย่างมีประสิทธิภาพ อีกทั้งคำ สั่งในการทำงานไม่ซับซ้อน ทำให้สามารถทำงานได้รวดเร็ว สามารถใช้กับสัญญาณนาฬิกาความถี่สูงมากๆ ได้ การทำงานถูกออกแบบมาให้ใช้รีจิสเตอร์ภายใน จึงจำเป็นต้องมีการถ่ายเทข้อมูลจากหน่วยความจำไปไว้ในรีจิสเตอร์ก่อน
- ออกแบบด้วย สถาปัตยกรรมแบบฮาร์วาร์ดที่ถูกปรับปรุงใหม่ (modified harvard architecture) เป็นโพรเซสเซอร์ที่สร้างขึ้นมาให้เหมาะกับการประมวลผลสัญญาณแบบเรียลไทม์ มีการแยกบัสระหว่างบัสข้อมูล กับบัสคำสั่งโปรแกรม เพื่อหลีกเลี่ยงการแย่งไปป์ไลน์ (pipeline) ที่เป็นสาเหตุให้ระบบช้าลง โดยปกติแล้วสถาปัตยกรรมฮาร์วาร์ดเป็นโครงสร้างของ DSP (Digital Signal Processor : DSP)
- คำสั่งการคำนวณและบวก (Multiply and Accumulation command : MAC) มีการทำงานแบบไปป์ไลน์ ทำให้ไม่เกิดการว่างการทำงานเนื่องจากต้องรอผลของส่วนอื่น จึงทำให้การคำนวณการแปลงฟูเรียร์แบบเร็วทำได้เร็วขึ้น
- มี 84 คำสั่งมาตรฐาน สามารถรองรับรูปแบบการอ้างแอดเดรสได้อย่างอิสระ
- ชุดคำสั่งมีขนาด 24 บิต สามารถประมวลผลข้อมูลได้ 16 บิต
- มีขนาดพื้นที่หน่วยความจำโปรแกรมแบบแฟลช ความจุ 144 Kbytes ลบและเขียนใหม่ได้ไม่น้อยกว่า 10,000 ครั้ง สามารถโปรแกรมหน่วยความจำโปรแกรมได้โดยทางซอฟต์แวร์ บรรจุชุดคำสั่งได้จริง 48 K words
- มีขนาดพื้นที่หน่วยความจำข้อมูลทั้งหมด มีขนาด 64 Kbytes (32K words) ภายในจะมีพื้นที่หน่วยความจำข้อมูลแรม 8 Kbytes แบ่งออกเป็น 2 ส่วนได้แก่ หน่วยความจำข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรม X และ Y แต่ละส่วนมีขนาด 16 บิต ความจุ 4 Kbytes ในแต่ละส่วนจะมีตัวกำหนดแอดเดรส (Address Generation Unit : AGU) แยกออกจากกัน คำสั่งคูณและบวกของชุดคำสั่งการประมวลผลสัญญาณดิจิตอล (DSP instructions) จาก 2 แหล่ง สามารถทำงานผ่าน ตัวกำหนดแอดเดรส X และ Y

- มีหน่วยความจำข้อมูลแรม 8 Kbytes
- ความเร็วในการทำงานสูงถึง 30 ล้านคำสั่งต่อวินาที (Million Instruction Per Second : MIPS)
- มีรีจิสเตอร์ W จัดอยู่ในรูปแบบอะเรย์ (working register array) มีขนาด 16 บิต จำนวน 16 ตัว เป็นรีจิสเตอร์หลักที่ใช้ในการทำงาน เนื่องจากโครงสร้างเป็นอะเรย์ จึงสามารถรองรับทั้งข้อมูล, ค่าแอดเดรส หรือค่าของรีจิสเตอร์ใดๆ ที่ต้องนำมาประมวลผล รีจิสเตอร์ W ตัวที่ 0-14 (W0 – W14) ใช้ในการประมวลผลหลัก ส่วนรีจิสเตอร์ W ตัวที่ 15 (W15) ใช้ทำงานร่วมกับตัวชี้สแต็ก (software stack pointer) ในการทำงานของโปรแกรมย่อย และบริการอินเตอร์รัปต์ (interrupt service)
- รองรับแหล่งกำเนิดสัญญาณอินเตอร์รัปต์ได้สูงสุด 41 แหล่ง โดยสามารถกำหนดระดับความสำคัญ (selectable priority levels)ได้ 8 ระดับ
- รับความถี่สัญญาณนาฬิกาจากภายนอกได้ ตั้งแต่ 4 MHz - 10 MHz ในกรณีใช้งานร่วมกับวงจรเฟสล็อกลูปภายใน เลือกปรับ ตัวคูณได้ 3 ระดับคือ 4, 8 หรือ 16 เท่า

2. คุณสมบัติด้านการประมวลผลสัญญาณดิจิตอล

- มีวงจรเฟตช์ข้อมูลคู่ (Dual data fetch) ทำให้สามารถประมวลผลข้อมูลได้อย่างรวดเร็ว
- สามารถเข้าถึงหน่วยคำจำข้อมูลได้ 2 รูปแบบพิเศษ คือ แบบวนรอบ และ แบบกลับบิต
    - การเข้าถึงหน่วยความจำข้อมูล แบบวนรอบ (modulo addressing) สามารถใช้ได้ในพื้นที่หน่วยความจำข้อมูล X และ Y วิธีการนี้ใช้เพื่อลดการทำงานแบบวนรอบ (loop program) สำหรับ DSP algorithms เป็นหลัก
    - การเข้าถึงหน่วยความจำข้อมูล แบบกลับบิต (bit-reversed addressing) สามารถใช้ได้ใน X AGU เท่านั้น เพื่อการเข้าถึงข้อมูลอย่างมีประสิทธิภาพ โดยการจัดเรียงข้อมูล เพื่อเข้าสู่ radix-2 FFT algorithms ทำได้ง่ายขึ้น
- มีแอกคิวมูเลเตอร์ (accumulator) ขนาด 40 บิต 2 ตัว ใช้เก็บข้อมูลสำหรับประมวลผลทางคณิตศาสตร์
    - เมื่อประมวลผลเชิงเศษส่วน สามารถเก็บข้อมูลได้ตั้งแต่ -256.0 ถึง 256 -2^{-31}
    - เมื่อประมวลผลเลขจำนวนเต็ม สามารถเก็บข้อมูลได้ตั้งแต่ -549,755,813,888 ถึง 549,755,813,887

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีหน่วยประมวลผลทางคณิตศาสตร์และลอจิก (Arithmetic Logic Unit : ALU) ขนาด 40บิต
- มีหน่วยประมวลผลด้านการคูณเลขจำนวนเต็มและเศษส่วน 17x17 บิต ในรูปของฮาร์ดแวร์ รองรับการคำนวณค่าตัวเลขทั้งที่ คิดและไม่คิดเครื่องหมาย โดยบิตที่ 16 จะเป็นบิตเครื่องหมาย สำหรับการให้ผลลัพธ์สามารถให้ผลได้ทั้งเลขจำนวนเต็มขนาด 32 บิต และ เลขทศนิยม ใช้เวลาในการทำงานเพียง 1 รอบการทำงาน (Instruction Cycle Period)
  - เมื่อทำงานในโหมดเลขจำนวนเต็ม จะให้ผลลัพธ์เป็น ค่า two's complement แบบคิดเครื่องหมาย ค่าที่ได้จะอยู่ในช่วง -2^{N-1} to $2^{N-1} - 1$
    - สำหรับเลขจำนวนเต็มขนาด 16 บิต จะมีค่าอยู่ระหว่าง -32768 (0x8000) ถึง 32767 (0x7FFF)
    - สำหรับเลขจำนวนเต็มขนาด 32 บิต จะมีค่าอยู่ระหว่าง -2,147,483,648 ถึง 2,147,483,645
  - เมื่อทำงานในโหมดเลขเศษส่วน จะให้ผลลัพธ์เป็น ค่า two's complement ของเลขเศษส่วนแบบคิดเครื่องหมาย ค่าที่ได้จะอยู่ในช่วง -1.0 ถึง $(1 - 2^{1-N})$
- สำหรับเลขทศนิยมขนาด 16 บิต จะมีค่าอยู่ระหว่าง -1.0 ถึง 0.999969482 และมีค่าความละเอียด 3.01518×10^{-5}
- ชุดคำสั่งการประมวลผลสัญญาณดิจิตอลทั้งหมดใช้เวลาในการทำงานเพียง 1 รอบการทำงาน
- ตัวเลื่อนข้อมูลสองทิศทางแบบบาร์เรล (barrel shifter) ขนาด 40 บิต สามารถเลื่อนไปทางซ้ายหรือขวาได้ภายใน 1 รอบการทำงาน เมื่อได้รับข้อมูลที่กำหนดเครื่องหมาย โดยถ้าเป็นค่าบวก จะเลื่อนข้อมูลไปทางขวา ถ้าเป็นค่าลบ จะเลื่อนข้อมู ลไปทางซ้าย ให้ผลลัพธ์ขนาด 40 บิต

3. คุณสมบัติด้านการเชื่อมต่อกับอุปกรณ์ภายนอก

- สามารถจ่ายกระแสออกทางขาพอร์ต ได้ 25 mA ทั้งแบบกระแสซิงก์และซอร์ส
- ไทเมอร์/เคาน์เตอร์ ขนาด 16 บิต จำนวน 5 ตัว สามารถต่อใช้งานร่วมกันเป็น ไทเมอร์ 32 บิตได้
- มีส่วนเชื่อมต่อกับอุปกรณ์อนุกรมในแบบ SPI (Serial Peripheral Interface : SPI)
- มีส่วนเชื่อมต่อกับอุปกรณ์เข้ารหัสและถอดรหัสเสียง ในแบบ DCI (Data Converter Interface : DCI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. คุณสมบัติพิเศษด้านอื่นๆ

- มีพาวเวอร์ออนรีเซต (POR), พาวเวอร์อัพไทเมอร์ (PWRT) และ ออสซิเลเตอร์สตาร์ทอัพไทเมอร์ (OST)
- มีวอตช์ด็อกไทเมอร์แบบโปรแกรมทางซอฟแวร์ได้ (Programmable Watchdog Timer : WDT) ทำหน้าที่รีเซตไมโครคอนโทรลเลอร์ในกรณีที่เกิดการทำงานทางซอฟแวร์ผิดพลาดจนไม่สามารถทำงานอื่นต่อได้
- รองรับการโปรแกรมในวงจรแบบอนุกรม (In-Circuit Serial Programming : ICSP)

5. การรองรับการหารเลข

รองรับการหารเลขอย่างเต็มรูปแบบ โดยไม่ต้องสร้างโปรแกรมย่อยขึ้นมาจัดการ แบ่งออกเป็น 5 รูปแบบ ดังนี้

1. การหารเลขเศษส่วนหรือทศนิยม 16 บิตแบบคิดเครื่องหมาย
2. การหารเลขจำนวนเต็มที่มีตัวตั้ง 32 บิต ด้วยตัวหาร 16 บิตแบบคิดเครื่องหมาย
3. การหารเลขจำนวนเต็มที่มีตัวตั้ง 32 บิต ด้วยตัวหาร 16 บิตแบบไม่คิดเครื่องหมาย
4. การหารเลขจำนวนเต็มที่มีตัวตั้ง 16 บิต ด้วยตัวหาร 16 บิตแบบคิดเครื่องหมาย
5. การหารเลขจำนวนเต็มที่มีตัวตั้ง 16 บิต ด้วยตัวหาร 16 บิตแบบไม่คิดเครื่องหมาย

6. ระบบสัญญาณนาฬิกา

- สามารถเลือกแหล่งกำเนิดสัญญาณนาฬิกาจากภายในหรือภายนอกได้
- มีวงจรเฟสล็อกลูป อยู่ภายในเพื่อเพิ่มความถี่ในการทำงานของสัญญาณนาฬิกา ดังแสดงในตารางที่ 2.3
- มีส่วนตรวจสอบความผิดพลาดของสัญญาณนาฬิกาเพื่อเปลี่ยนแหล่งกำเนิดสัญญาณนาฬิกาได้

ตารางที่ 2.3 ย่านสัญญาณความถี่นาฬิกาที่สามารถใช้ร่วมกับวงจรเฟสล็อกลูปภายใน

ความถี่ขาเข้า	ตัวคูณเฟสล็อกลูป	ความถี่ขาออก
4 MHz -10 MHz	x4	16 MHz -40 MHz
4 MHz -10 MHz	x8	32 MHz -80 MHz
4 MHz -7.5 MHz	x16	64 MHz -120 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

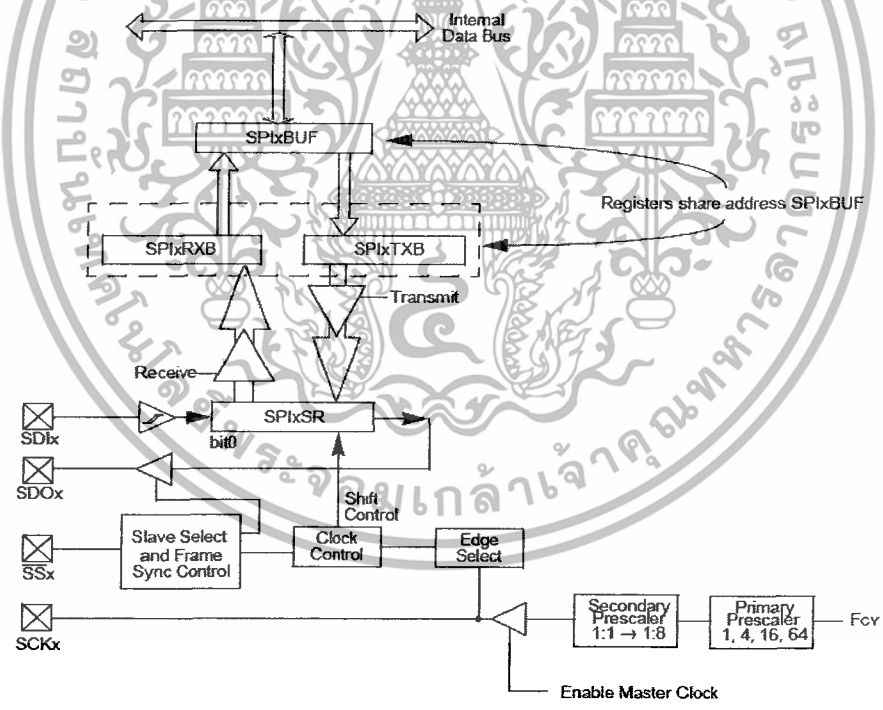
2.2.2 โมดูลสำหรับเชื่อมต่ออุปกรณ์อนุกรม แบบ SPI

การติดต่อแบบ SPI เป็นการติดต่ออนุกรมแบบเป็นจังหวะ (synchronous serial interface) อย่างหนึ่ง ช่วยเพิ่มความสะดวกในการติดต่อกับอุปกณ์ภายนอก ที่มีระบบการติดต่อแบบ SPI เหมือนกัน เช่น หน่วยความจำภายนอก EEPROM, ไอซีวัดอุณหภูมิ, อุปกรณ์แปลงสัญญาณแอนะลอกเป็นดิจิตอล, อุปกรณ์แปลงสัญญาณดิจิตอลเป็นแอนะลอก หรือ ไมโครคอนโทรลเลอร์ตัวอื่น สำหรับในโปรเจคนี้จะใช้สำหรับติดต่อกับ ไอซีขับหน้าจอแสดงผล LCD

โมดูล SPI มีขาสำหรับต่อใช้งาน 4 ขา ซึ่งใช้ร่วมกับขาพอร์ตปกติด้วย ประกอบด้วย

1. SDIx Pin (Serial Data Input) เป็นขารับข้อมูลอนุกรม
2. SDOx Pin (Serial Data Output) เป็นขาส่งข้อมูลอนุกรม
3. SCKx Pin (Shift Clock input or output) เป็นขาสัญญาณนาฬิกาอนุกรม
4. SSx Pin (active low slave select or frame synchronization I/O pulse) เป็นขาเลือกอุปกรณ์สเลฟ ทำงานที่ลอจิก "0" หรือใช้เป็นขารับส่งสัญญาณพัลส์สำหรับกำหนดจังหวะของเฟรมข้อมูลอนุกรม

แต่อย่างไรก็ตาม ปกติแล้วจะใช้เพียง 3 ขาหลักคือ SDI, SDO และ SCK ขึ้นอยู่กับการกำหนดโหมดการทำงาน ในรูปที่ 2.23 แสดงบล็อกไดอะแกรมการทำงานของโมดูล SPI ใน dsPIC

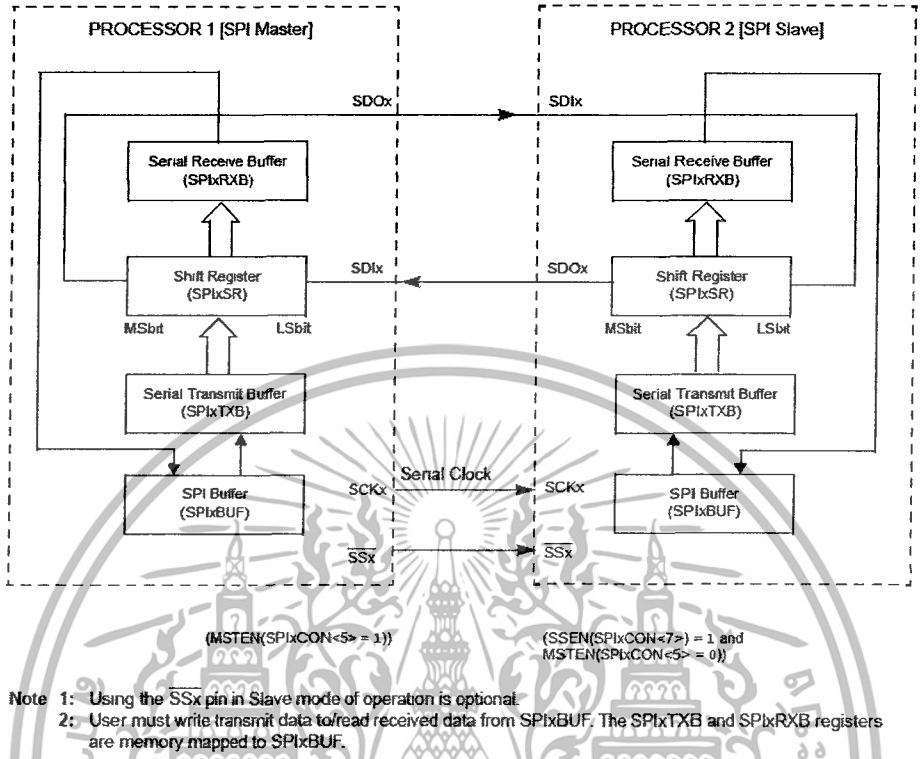


Note: The SPIxTXB and SPIxRXB registers are memory mapped to the SPIxBUF register.

รูปที่ 2.23 บล็อกไดอะแกรม ของ โมดูล SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับโครงงานนี้ได้กำหนดให้โมดูล SPI ทำงานในโหมด มาสเตอร์ และมีรูปแบบการเชื่อมต่อระหว่างไมโครคอนโทรเลอร์กับไอซีเข้าและถอดรหัสเสียง ดังรูปที่ 2.24



รูปที่ 2.24 การเชื่อมต่อโมดูล SPI ระหว่างอุปกรณ์มาสเตอร์ และอุปกรณ์สเลฟ

2.2.3 โมดูลสำหรับเชื่อมต่ออุปกรณ์อนุกรม แบบ DCI

โมดูล DCI ช่วยให้การเชื่อมต่อกับอุปกรณ์ภายนอกทำได้ง่ายขึ้น อาทิเช่น อุปกรณ์เข้าและถอดรหัสเสียง (audio codecs) หรือ อุปกรณ์แปลงสัญญาณแอนะลอกเป็นดิจิตอล, อุปกรณ์แปลงสัญญาณดิจิตอลเป็นแอนะลอก สำหรับโปรเจคนี้จะใช้สำหรับติดต่อกับ ไอซี Si3000

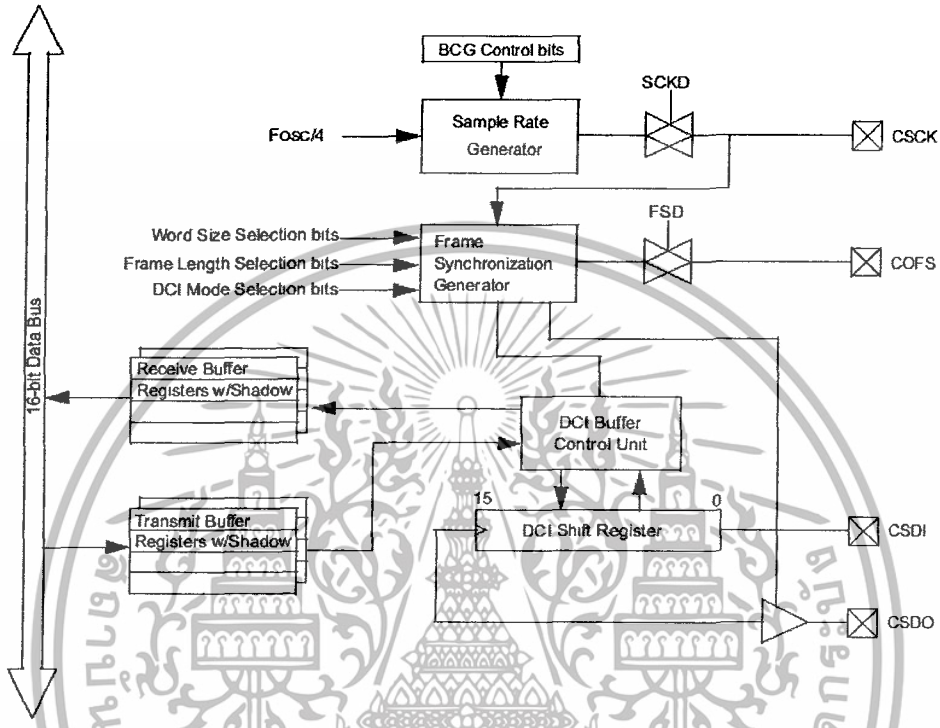
1. คุณสมบัติทั่วไปของ โมดูล DCI

- สามารถกำหนดขนาดเวิร์ดข้อมูล (data word size) ได้สูงสุดที่ขนาด 16 บิต ด้วยวิธีการโปรแกรม
- รองรับการทำงานได้สูงสุด 16 ไทม์สลอต (time slot) ใน เฟรมข้อมูล (data frame) และรองรับขนาดเฟรม (frame size) ได้สูงสุด 256 บิต โดยมีบิตควบคุมอยู่ในทุกๆไทม์สลอตในเฟรมข้อมูล ซึ่งจะคอยพิจารณาว่าเมื่อใด DCI จะรับหรือส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีแหล่งที่พักสำรองข้อมูล (data buffer) สามารถเก็บข้อมูลได้ถึง 4 เวิร์ดข้อมูล โดยไม่แย่งทรัพยากรของ CPU

ในรูปที่ 2.25 จะแสดงถึงลักษณะการทำงานของส่วนประกอบต่างๆภายในโมดูล DCI ซึ่งแสดงถึงการทำงานรับส่งข้อมูลระหว่างโมดูล DCI กับ บัสข้อมูล (data bus)



รูปที่ 2.25 บล็อกไดอะแกรม ของ โมดูล DCI

2. ขาสำหรับต่อใช้งาน โมดูล DCI

โมดูล DCI มีขาสำหรับต่อใช้งาน 4 ขา ประกอบด้วย

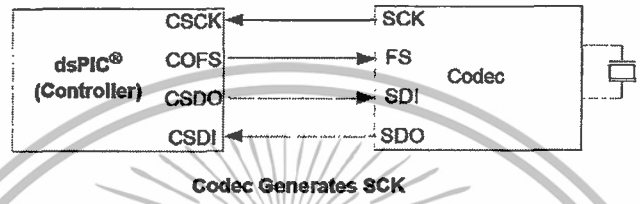
1. CSCK Pin (Serial Clock input or output) ทำหน้าที่ เชื่อมต่อสัญญาณนาฬิกาเข้าด้วยกัน สามารถกำหนดให้ไมโครคอนโทรลเลอร์เป็น ตัวรับสัญญาณนาฬิกาจากอุปกรณ์ภายนอกหรือเป็นตัวส่งสัญญาณนาฬิกา ผ่านทางบิตควบคุม
2. CSDO Pin (Serial Data Output) ถูกกำหนดให้เป็นขาส่งข้อมูลออก เมื่อโมดูล DCI ถูกเอนเอเบิล (enable) สามารถกำหนดว่าจะส่งข้อมูลออกหรือไม่ ผ่านทางบิต ควบคุม
3. CSDI Pin (Serial Data Input) ถูกกำหนดให้เป็นขารับข้อมูลเข้า เมื่อโมดูล DCI ถูกเอนเอเบิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. COFS Pin (Frame Synchronization) ทำหน้าที่ ซิงโครไนซ์ (synchronize) การถ่ายโอนข้อมูล ซึ่งเกิดขึ้นบน CSDO และ CSDI สามารถกำหนดได้ว่าจะรับหรือส่งพัลส์ (frame synchronization pulse) ผ่านทางบิตควบคุม

ในรูปที่ 2.26 นี้ ไมโครคอนโทรเลอร์เป็นตัวส่งพัลส์ไปยังไอซีเข้าและถอดรหัสเสียงจึงจัดว่า โมดูล DCI ทำงานในโหมดมาสเตอร์

- ในกรณีที่ไมโครคอนโทรเลอร์สร้างพัลส์ ส่งให้อุปกรณ์ภายนอกโมดูล DCI จะเป็น มาสเตอร์
- ในกรณีที่ไมโครคอนโทรเลอร์รับพัลส์ จากอุปกรณ์ภายนอก โมดูล DCI จะเป็น สเลฟ



ข้อสังเกต : วงจรออสซิเลเตอร์ของอุปกรณ์เข้าและถอดรหัส ทำหน้าที่ สร้างสัญญาณ SCK

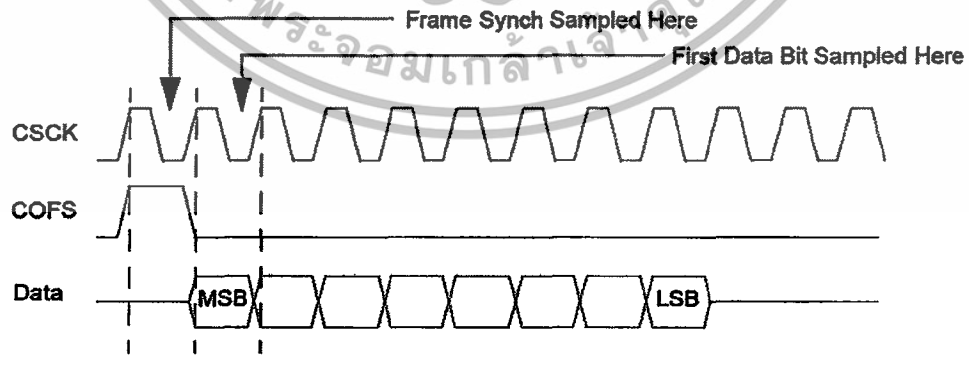
รูปที่ 2.26 แสดงตัวอย่างการเชื่อมต่อกับอุปกรณ์เข้าและถอดรหัส

3. ลักษณะการเชื่อมต่อ โมดูล DCI

โมดูล DCI มีลักษณะการเชื่อมต่ออยู่ 3 รูปแบบใหญ่ๆดังนี้

1. แบบสื่อสารอนุกรม (framed synchronous serial Transfer (single or m ulti-Channel)
2. แบบระบบบัส I^2S (Inter-IC Sound interface)
3. แบบ AC-Link compliant mode

สำหรับในโครงงานนี้ กำหนดให้ การสื่อสารอนุกรม DCI ทำงานแบบ สเลฟเฟรมซิงค์ชนิดหลายช่องทาง (Slave Frame Sync & Multi-Channel mode) โดยรูปที่ 2.27 แสดงถึงจังหวะการส่งสัญญาณนาฬิกาเพื่อทำงานซิงโครนัสกับเฟรมข้อมูล



รูปที่ 2.27 ไทมิงไดอะแกรมของการทำงานแบบ สเลฟเฟรมซิงค์ ชนิดหลายช่องทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในบทที่ 2 นี้ ได้กล่าวถึง ทฤษฎีที่เกี่ยวข้องกับการประมวลผลสัญญาณดิจิตอล และความรู้ทางด้านไมโครคอนโทรเลอร์ที่เกี่ยวข้องในการเขียน โปรแกรม รวมถึงโมดูลพิเศษต่างๆ เพื่อในไปใช้ในการทำโครงงานนี้ ในบทที่ 3 นั้นจะเป็นการอธิบายถึงการนำความรู้และทฤษฎีในบทที่ 2 มาใช้ในการออกแบบโครงงานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการออกแบบ

การออกแบบระบบกำจัดเสียงรบกวน ในเบื้องต้นนั้นต้องเข้าใจหลักการทำงานในภาพรวมของระบบก่อนว่ามีหลักการทำงานอย่างไร แล้วจึงพิจารณาส่วนย่อยต่างๆ อันประกอบไปด้วย หน่วยประมวลผลกลางและกำจัดเสียงรบกวน ,หน่วยเข้าและถอดรหัสเสียง , และหน่วยแสดงผล โดยหน่วยย่อยทั้งสามนี้จะทำงานร่วมกัน ภายใต้ระบบการทำงานที่ถูกออกแบบไว้ ท้ายที่สุดจึงเป็นการนำเสนอโครงสร้างการทำงาน ในส่วนของโปรแกรม ซึ่งอธิบายโครงสร้างของระบบกำจัดเสียงรบกวนได้ตามรูปที่ 3.1 อันประกอบด้วยส่วนประกอบต่างๆ ดังนี้

1. หน่วยเข้าและถอดรหัสเสียง

ทำหน้าที่สองอย่างคือ เมื่อรับสัญญาณเสียงแอนะลอกจากไมโครโฟน ไอซี Si3000 จะแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล และนำสัญญาณดิจิตอลนั้นส่งไปประมวลผลในรูปแบบเฟรมข้อมูล ด้วยการเชื่อมต่อแบบอนุกรม (DCI) ในทางกลับกันเมื่อไมโครคอนโทลเลอร์กำจัดเสียงรบกวนเสร็จแล้ว ไอซี*Si3000* จะแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกและส่งเสียงแอนะลอกนั้นทางลำโพง หน่วยเข้าและถอดรหัสเสียง สามารถแสดงได้ดังรูปที่ 3.1(A)

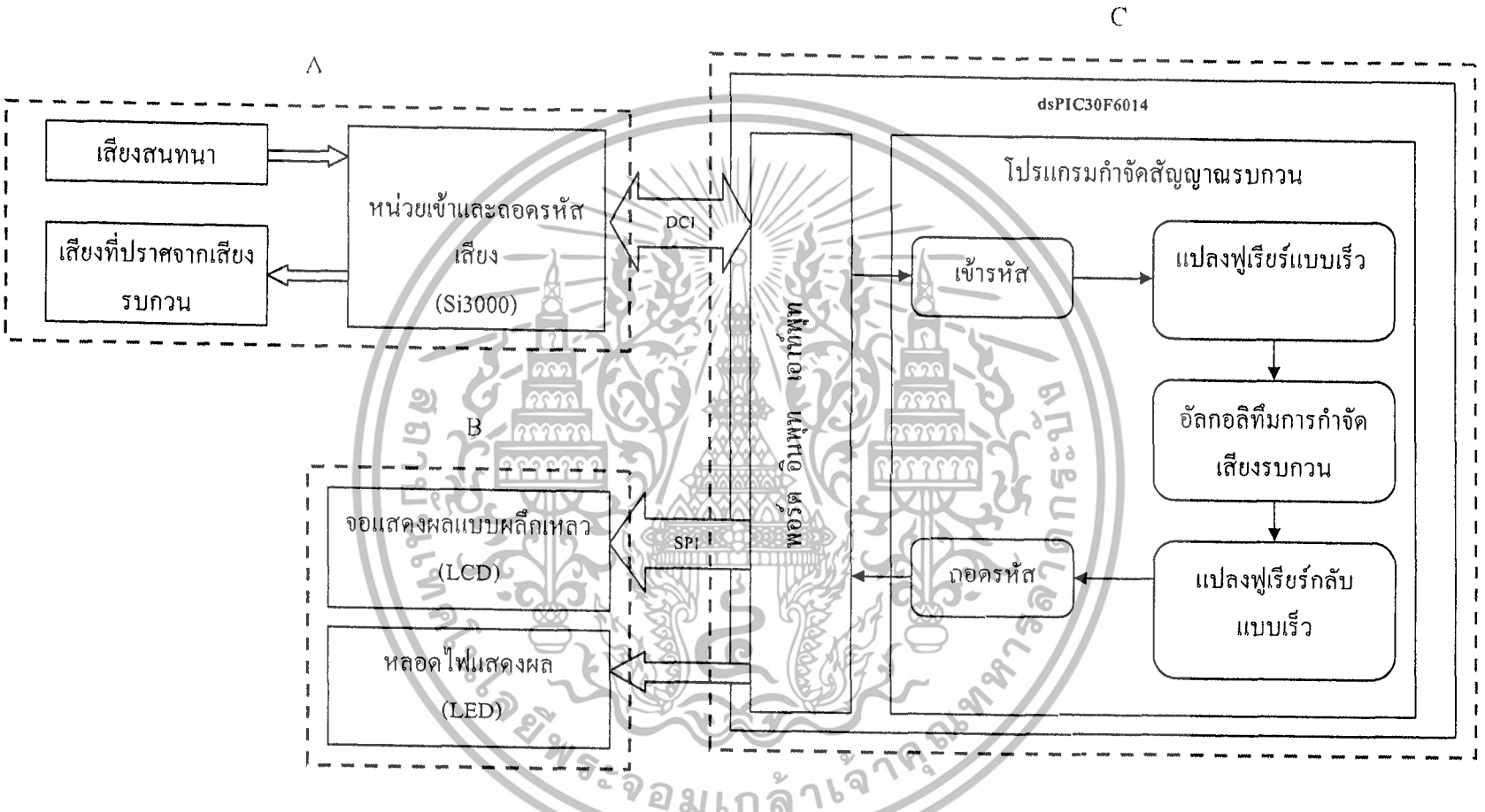
2. หน่วยแสดงผล

ในโครงงานนี้ผู้จัดทำได้ใช้จอแสดงผลในการแสดงค่าอัตราส่วนของสัญญาณเสียงสนทนาต่อเสียงสัญญาณรบกวน SNR และใช้หลอดไฟ LED ในการแสดงโหมดการทำงานของโปรแกรม ดังรูปที่ 3.1(B) และเนื่องจากผู้จัดทำใช้บอร์ดทดลอง dsPICDEM 1.1 ในการพัฒนาโปรแกรม ซึ่งในบอร์ดนี้มีส่วนแสดงผลอยู่แล้วอันประกอบด้วยจอแสดงผลแบบ ผลึกเหลว (Liquid Crystal Display : LCD) และหลอดไฟ ในส่วนของจอแสดงผล เป็นแบบดอทเมตริกซ์ (dot-matrix)122x32 ขนาด 4 บรรทัด 20 ตัวอักษร ใช้ไอซี PIC18F242 LCD เป็นคอนโทรลเลอร์ ติดต่อกับ LCD ในโหมด 8 บิต และ dSPIC30F6014 ซึ่งเป็นไมโครคอนโทรลเลอร์ตัวหลักจะทำการเชื่อมต่อกับไอซี PIC18F242 LCD ผ่านการสื่อสารแบบอนุกรม SPI โดยผู้จัดทำกำหนดให้ dSPIC30F6014 เป็นมาสเตอร์ และไอซี PIC18F242 LCD เป็นสเลฟ

3.หน่วยประมวลผลกลางและกำจัดเสียงรบกวน

เป็นหน่วยที่มีความสำคัญมากที่สุด ในโครงงานนี้ใช้ ไมโครคอนโทลเลอร์ dsPIC30F6014 ขนาด 32 บิตเป็นตัวประมวลผล ภายในได้สร้างระบบการกำจัดเสียงรบกว นโดยใช้ภาษาซี ร่วมกับภาษาแอสแซมบีในการพัฒนาโปรแกรม และใช้ MPLAB C30 เป็นตัวคอมไพรเลอร์ดังรูปที่ 3.1(C) ซึ่งบล็อกไดอะแกรมการทำงานของระบบกำจัดเสียงรบกวนจะได้กล่าวต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 โครงสร้างการทำงานในการกำจัดสัญญาณเสียงรบกวน

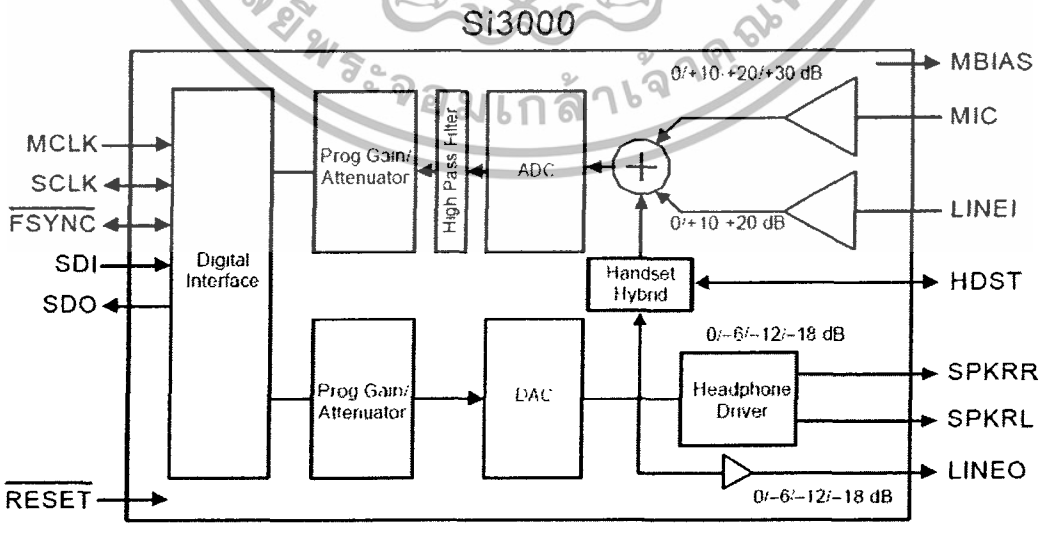
3.1 หน่วยเข้าและถอดรหัสเสียง

ในโครงงานนี้จะใช้หน่วยเข้าและถอดรหัสเสียง Si3000 ซึ่งเป็นไอซีสำเร็จรูปที่มีฟังก์ชันการทำงานกับเสียงหลากหลาย และที่สำคัญคือไอซี Si3000 สามารถรองรับการประมวลผลแบบดิจิตอลซึ่งสอดคล้องกับความต้องการของโครงงาน สำหรับคุณสมบัติของไอซี Si3000 จะอธิบายได้ดังต่อไปนี้

3.1.1 คุณสมบัติของ Si3000

- แปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลและแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกขนาด 10 บิต
- สามารถปรับเพิ่ม,ลดสัญญาณเสียงในช่วง –34.5 ถึง12 dB โดยการปรับค่าทางโปรแกรม
- สามารถรับสัญญาณแอนะลอกได้ 3 แหล่ง
- เชื่อมต่อแบบอนุกรมกับไมโครคอนโทรลเลอร์
- ขยายสัญญาณเสียงอินพุท ได้สูงสุด 30 dB
- รองรับความถี่สุ่มตัวอย่างได้ 4 ถึง12 kHz

การทำงานของไอซี Si3000 เริ่มจากรับเสียงแอนะลอกจากไมโครโฟน หรือ จากสายออดิโอ (audio)โดยตรง จากนั้นสัญญาณจะถูกขยายในช่วง 0 ถึง 30 เดซิเบล สัญญาณแอนะลอกนั้นจะถูกแปลงเป็นสัญญาณดิจิตอล แล้วผ่านตัวกรองความถี่สูง และโปรแกรมเกนตามลำดับ ข้อมูลดิจิตอลนั้นจะติดต่อสื่อสารกับหน่วยประมวลผลผ่านการเชื่อมต่อแบบอนุกรม DCI ในทางกลับกันข้อมูลดิจิตอลจะถูกส่งออกมาที่ไอซี Si3000 ผ่านโปรแกรมเกนเอ้าพุท และทำการแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกกลับ และส่งสัญญาณ นั้นออกทางสายออดิโอ หรือลำโพงซึ่งในที่นี้ขึ้นอยู่กับการกำหนดค่าทางโปรแกรม การทำงานของไอซี Si3000 อธิบายได้ดังรูปที่ 3.2



รูปที่ 3.2 บล็อกไดอะแกรมของไอซี Si3000

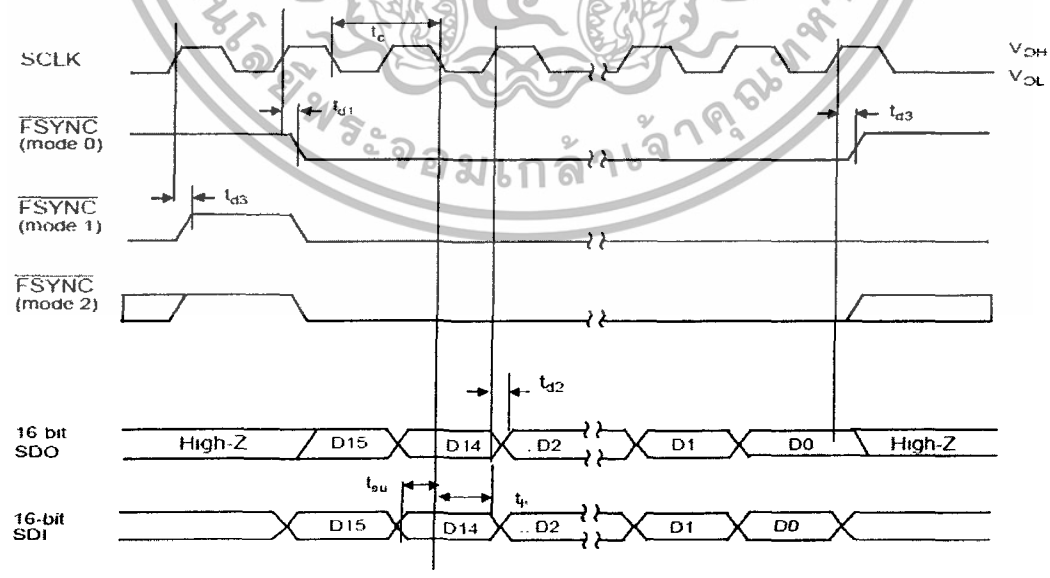
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 การเชื่อมต่อแบบดิจิตอล

การเชื่อมต่อแบบดิจิตอลแบ่งออกเป็น 3 โหมด ในแต่ละโหมดจะมีช่วงจังหวะการรับส่งข้อมูลในเฟรมที่ขึ้นกับจังหวะสัญญาณนาฬิกา SCLK ดังรูปที่ 3.3 โดยมีคุณลักษณะการสวิตช์ชิงของการเชื่อมต่อแบบอนุกรมดังตารางที่ 3.1

ตารางที่ 3.1 คุณลักษณะการสวิตช์ชิงของการเชื่อมต่อแบบอนุกรม

Parameter	Symbol	Min	Type	Max	Unit
Cycle Time, SCLK	t_c	354	1/256 Fs	-	ns
SCLK Duty Cycle	t_{dty}	-	50	-	%
Delay Time, SCLK ↑ to FSYNC ↓	t_{d1}	-	-	10	ns
Delay Time, SCLK ↑ to SDO Valid	t_{d2}	-	-	20	ns
Delay Time, SCLK ↑ to FSYNC ↑	t_{d3}	-	-	10	ns
Setup Time, SDI, before SCLK ↓	t_{su}	25	-	-	ns
Hold Time, SDI, after SCLK ↓	t_h	20	-	-	ns
Setup Time, FSYNC (mode 2) before MCLK ↓	t_{su}	25	-	-	ns
Hold Time, FSYNC (mode 2) after MCLK ↓	t_h	20	-	-	ns



รูปที่ 3.3 ไทมิงไดอะแกรมของการเชื่อมต่อแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกโหมดการรับส่งข้อมูล โดยใช้ตัวต้านทาน50 kΩ พลูอัพ/ดาวน์ ที่ขา SDO และ SDI ตามตารางที่ 3.2 สำหรับในโครงงานนี้จะกำหนดให้รับส่งข้อมูลแบบอนุกรมในโหมดที่ 1 โดยให้ไอซี Si3000 เป็นมาสเตอร์

ตารางที่ 3.2 โหมดการเชื่อมต่อแบบอนุกรม

โหมด	SCLK*	SDO*	รายละเอียด
0	0	0	$\overline{FSYNC}$ เฟรมข้อมูล
1	0	1	$\overline{FSYNC}$ เป็นสัญญาณขั้นบันไดเริ่มเฟรมข้อมูล
2	1	0	สเลฟโหมด
3	1	0	ไม่ใช้งาน
*เมื่อ : พลูอัพ/พลูดาวน์			

เมื่อกำหนดให้ไอซี Si3000 เป็นมาสเตอร์ (master clock - MCLK) จะเป็นอินพุท และ(serial data clock : SCLK) จะเป็นเอาท์พุท และ SCLK จะทำงานที่ 256 บิต ต่อเฟรม เมื่อเฟรมเรท (frame rate) มีค่าเท่ากับความถี่สุ่มตัวอย่าง (Fs) โดยการส่งและรับข้อมูลจะแบ่งออกเป็น 2 เฟรม คือ

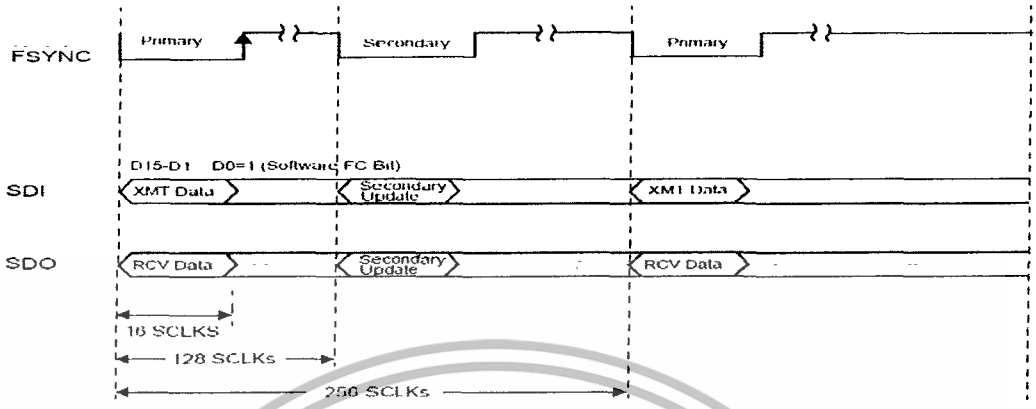
- เฟรมปฐมภูมิ จะใช้สำหรับเก็บจุดตัวอย่างข้อมูลสัญญาณเสียงดิจิตอล
- เฟรมทุติยภูมิ จะใช้ติดต่อกับรีจิสเตอร์ ภายในของ ไอซี Si3000 และจะเกิดขึ้นเฉพาะเวลาถูกร้องขอเท่านั้น

การส่งข้อมูลในเฟรมปฐมภูมิ (จาก dsPIC ไปยังไอซี Si3000) จะใช้ LSB เป็นบิตร้องขอเฟรมทุติยภูมิ ดังนั้นข้อมูลเสียงที่ส่งมาจะมีเพียง 15 บิตเท่านั้น และเมื่อเฟรมทุติยภูมิ ไม่ปรากฏก็จะไม่เกิดการส่งข้อมูลในไทม์สลอตนี้

การรับมูลในเฟรมปฐมภูมิ (จากไอซี Si3000 ไปยัง dsPIC) ในโหมด 0 หรือ1 จะรับทางขา SDO พร้อมกับข้อมูลเสียง เมื่อ dsPIC ได้รับเฟรมปฐมภูมิ แล้วจะตรวจสอบ LSB ว่าจะรับข้อมูลจากไอซี Si3000 ต่อ หรือส่งเฟรมทุติยภูมิออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการส่งและรับข้อมูลในเฟรมทุติยภูมิ จะส่งและรับข้อมูลแบบ 16 บิต ดังรูปที่ 3.4



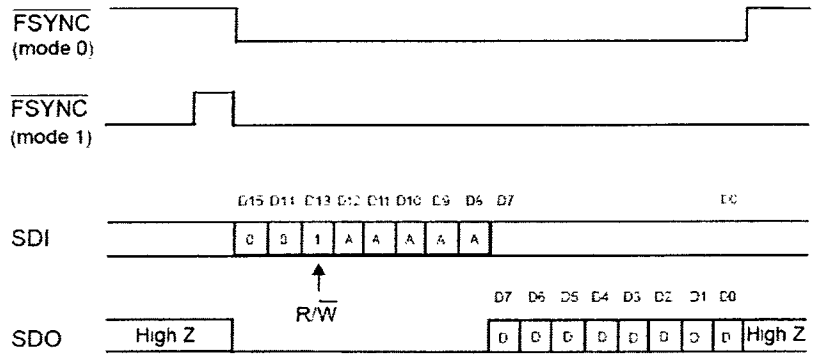
รูปที่ 3.4 การร้องขอเฟรมทุติยภูมิ

ระหว่างการเขียนข้อมูลบิต R/W จะถูกเคลียร์ ตามด้วย แอดเดรสของรีจิสเตอร์ที่เขียนขนาด 5 บิตแล้วตามด้วย ข้อมูลขนาด 8 บิต และจะมีรีจิสเตอร์เพียงตัวเดียวเท่านั้นที่จะถูกอ่านหรือเขียนในเฟรมทุติยภูมิดังรูปที่ 3.5

รูปที่ 3.5 การเขียนข้อมูลลงเฟรมทุติยภูมิ

ระหว่างการอ่านข้อมูล บิต R/W จะถูกกำหนดให้เป็น 1 ตามด้วย แอดเดรสของรีจิสเตอร์ที่อ่านขนาด 5 บิต ส่วนข้อมูลขนาด 8 บิต ที่เหลือจะใช้กับ รีจิสเตอร์ควบคุม (control register) ซึ่งข้อมูลจะถูกเขียนผ่านขา SDO ดังรูปที่ 3.6

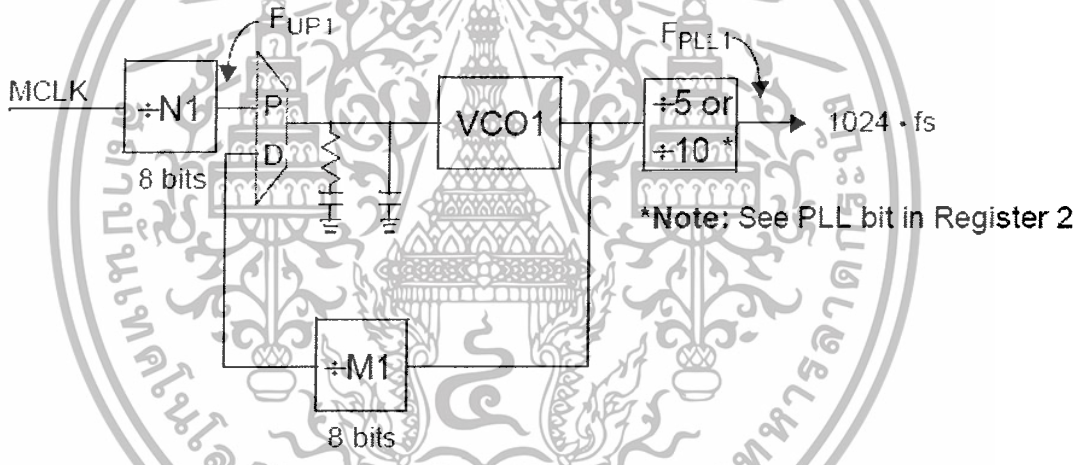
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 การอ่านข้อมูลจากเฟรมทุติยภูมิ

3.1.3 การเลือกความถี่สัญญาณนาฬิกา

ไอซี Si3000 สามารถสร้างความถี่สุ่มตัวอย่าง โดยใช้ออสซิเลเตอร์จากภายนอก (MCLK) เป็นตัวกำหนด



รูปที่ 3.7 ระบบการสร้างสัญญาณนาฬิกาด้วยเฟสล็อกลูป

สัญญาณนาฬิกาที่ใช้ในการสุ่มตัวอย่างจะถูกสร้างมาจากเฟสล็อกลูป ซึ่งจากรูปที่ 3.7 F_{PLL1} จะเป็นตัวกำหนดค่าความถี่สุ่มตัวอย่าง สถาปัตยกรรมระบบนี้เป็นแบบวงปิด ที่ใช้รีจิสเตอร์ 2 ตัว เป็นตัวกำหนดความถี่สุ่มตัวอย่าง ซึ่งใช้วิธีกำหนดค่าทางโปรแกรม นอกจากนั้นความถี่จาก MCLK ยังสามารถรองรับในช่วงความถี่ 1 MHz ถึง 60 MHz โดยตัวกำเนิดสัญญาณนาฬิกา (clock generator) จะสร้างความถี่สัญญาณนาฬิกา ที่1024*Fs เมื่อ Fs คือความถี่สุ่มตัวอย่างที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 การกำหนดค่ารีจิสเตอร์สำหรับตัวกำเนิดสัญญาณนาฬิกา

ประกอบด้วยรีจิสเตอร์ 2 ตัวได้แก่

Register 3 - N1 divider, 8 bits

Register 4 - M1 divider, 8 bits

เมื่อ N1 และ M1 เป็นรีจิสเตอร์ ขนาด 8 บิต แบบไม่คิดเครื่องหมาย

เราสามารถคำนวณค่า N1 และค่า M1 ได้ตามสมการความสัมพันธ์ ระหว่างความถี่จาก MCLK และความถี่สุ่มตัวอย่าง ดังสมการที่ (3.21)

$$\frac{M1}{N1} = \frac{5 * 1024 * Fs}{MCLK} \tag{3.21}$$

เมื่อ: Fs คือ ความถี่สุ่มตัวอย่าง

ตารางที่ 3.3 ตัวอย่างค่าความถี่จาก MCLK ในการสร้างความถี่สุ่มตัวอย่าง 8 kHz

ความถี่ออสซิเลเตอร์ MCLK (MHz)	N1	M1
1.8432	9	200
4.000	25	256
4.0960	1	10
5.2800	33	256
5.7600	9	64
6.1440	3	20
8.1920	1	5
9.2160	9	40
10.0800	63	256
10.5600	33	128
11.0592	27	100
12.288	3	10
14.7456	9	25
16.000	25	64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับการเขียนรีจิสเตอร์ มีความสำคัญมากสำหรับความถี่ที่ได้จากระบบนี้ N1 จะต้องถูกเขียนก่อน M1 เสมอ

<u>ข้อสังเกต</u> ค่า N1 และ ค่า M1 ที่ถูกเขียนจริงจะต้องน้อยกว่าการคำนวณอยู่ 1 เสมอ

ขั้นตอนสุดท้ายของการออกแบบ คือการปรับแก้อัตราความถี่ (update rate) ตามสมการที่ (3.22) และ (3.23)

$$\text{PLL lock time} < 1 \text{ ms} \tag{3.22}$$

$$F_{up} = \frac{F_{MCLK}}{N1} \geq 144kHz \tag{3.23}$$

เมื่อ: F_{up} คือ ความถี่ที่เข้ามาเพื่อทำการปรับแก้ ดังแสดงในรูปที่ 3.7

เมื่อทราบความถี่จาก MCLK และความถี่สุ่มตัวอย่างที่ออกแบบ จะทำให้สามารถคำนวณค่ารีจิสเตอร์ M1 และ N1 ได้ สิ่งที่ต้องพิจารณาอีกสิ่งหนึ่ง คือช่วงของแต่ละรีจิสเตอร์ ควรเป็นช่วงที่น้อยที่สุดของการปรับแก้อัตราความถี่ของเฟสล็อกลูป นั่นหมายความว่า M1 และ N1 นั้นต้องเป็นอัตราส่วนอย่างต่ำ นอกจากนั้น ค่าที่คำนวณออกมาต้องลบออก “1” ในเวลาเขียนข้อมูลลงรีจิสเตอร์เสมอ เนื่องจากการเก็บค่าในรีจิสเตอร์ จะเริ่มตั้งแต่ 0

3.2 หน่วยแสดงผล

ในโครงงานนี้ผู้จัดทำได้ใช้จอแสดงผลในการแสดงค่าอัตราส่วนของสัญญาณเสียงสนทนาต่อเสียงสัญญาณรบกวน SNR และใช้หลอด LED ในการแสดงโหมดการทำงานของโปรแกรม และเนื่องจากผู้จัดทำใช้บอร์ดทดลอง dsPICDEM 1.1 ในการพัฒนาโปรแกรม ซึ่งในบอร์ดนี้มีส่วนแสดงผลอยู่แล้วอันประกอบด้วย จอแสดงผล LCD และหลอดไฟในส่วนของจอแสดงผล LCD เป็นแบบดอทเมตริกซ์ (dot-matrix)122x32 ขนาด 4 บรรทัด 20 ตัวอักษร ใช้ไอซี PIC18F242 LCD เป็นคอนโทรลเลอร์ ติดต่อกับ LCD ในโหมด 8 บิต และ dSPIC30F6014 ซึ่งเป็นไมโครคอนโทรลเลอร์ตัวหลักจะทำการเชื่อมต่อกับ ไอซี PIC18F242 LCD ผ่านการสื่อสารแบบอนุกรม SPI โดยผู้จัดทำกำหนดให้ dSPIC30F6014 เป็นมาสเตอร์ และไอซี PIC18F242 LCD เป็น สเลฟ ดังที่ได้กล่าวถึงลักษณะการเชื่อมต่อมาแล้วในบทที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 หน่วยประมวลผลกลางและกำจัดเสียงรบกวน

ในที่นี้ผู้จัดทำจะแยกอธิบายถึงกระบวนการกำจัดเสียงรบกวน และขั้นตอนการทำงานของโปรแกรม (flow chart) โดยรวมเพื่อให้ง่ายต่อการเข้าใจดังนี้

3.3.1 กระบวนการกำจัดเสียงรบกวน

หลักการทำงานในภาพรวมของการกำจัดเสียงรบกวน สามารถแสดงได้ดังในรูปที่ 3.8 ซึ่งประกอบด้วยส่วนประกอบต่างๆ ดังนี้

- ตัวกรองความถี่สูงผ่าน (High-Pass Filter)
- การแปลงฟูเรียร์แบบเร็ว(Fast Fourier Transform)
- การคำนวณระดับพลังงาน (Band Energy Computation)
- การคำนวณแถบ SNR (Band SNR Computation)
- การคำนวณสเกลแฟกเตอร์ (Scale Factor Computation)
- การคำนวณแยกแยะระหว่างสัญญาณเสียงพูดและสัญญาณเสียงรบกวน (Voice Activity Detection and Noise Band Energy Computation : VAD)
- การปรับสเกลของแถบความถี่ (Scaling of Frequency Bands)
- การแปลงกลับของสัญญาณในโดเมนของเวลา (Time Domain Conversion)

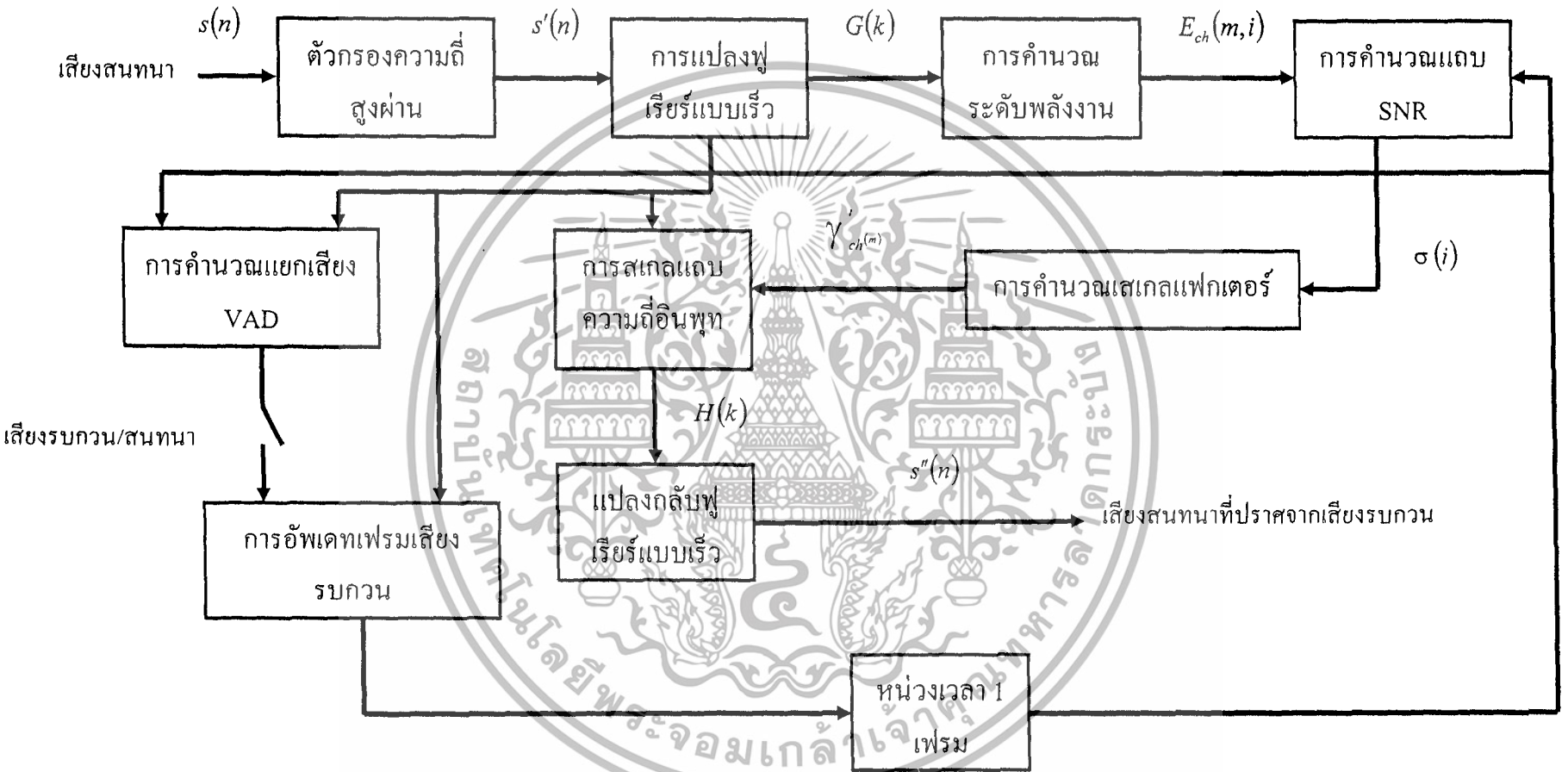
ตัวกรองความถี่สูงผ่าน

ตัวกรองความถี่สูงผ่าน (High-Pass Filter : HPF) จะมีความถี่คัทออฟที่ 80 Hz ใช้ในการตัดองค์ประกอบความถี่ต่ำของสัญญาณ ไฟกระแสตรง และความถี่อื่นๆที่ใกล้เคียงกับสัญญาณเสียงพูด ผลลัพธ์ของการกรองนี้จะได้ $s'(n)$ ตามบล็อกไดอะแกรมในรูปที่ 3.8

ทรานเฟอร์ฟังก์ชัน (transfer function) ของตัวกรองความถี่สูงผ่านแสดงดังสมการที่ (3.24)

$$H(z) = 0.5\frac{0.92727435 - 1.8544941z^{-1} + 0.92727435z^{-2}}{1 - 1.9059465z^{-1} + 0.9114024z^{-2}} \qquad (3.24)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 บล็อกไดอะแกรมการกำจัดเสียงรบกวน

การแปลงฟูเรียร์แบบเร็ว

สัญญาณเสียงที่เข้ามาจะถูกทำการเน้นเสียงเน้นคำ (pre – emphasized) ดังสมการที่ (3.25)

$$d(n) = s(n) + \xi s'(n-1) \tag{3.25}$$

เมื่อ: $\xi = -0.8$

การเน้นเสียงเน้นคำสัญญาณนั้นจะทำทุก 10 ms และจะเกิดการเหลื่อมกันของสัญญาณในเฟรมก่อนหน้าอีก 3 ms ดังนั้นจาก 104 ตัวอย่างสัญญาณ ผลของการเน้นเสียงเน้นคำจะทำให้เกิด zero - padded ขึ้นมา 24 ตัว ด้วยเหตุนี้จะเกิดสัญญาณตัวอย่าง 128 ตัว สัญญาณที่ได้จะถูกนำมาประมวลผลโดยใช้วินโดว์ (buffer window coefficients) ตามตารางที่ 3.4

ตารางที่ 3.4 สัมประสิทธิ์ของวินโดว์ฟังก์ชันต่างๆ (BUFFER WINDOW COEFFICIENTS)

สัมประสิทธิ์ของวินโดว์ฟังก์ชัน	เงื่อนไข
$\sin^2(\pi(n+0.5)/2D)$	$0 \le n < D$
1	$D \le n < L$
$\sin^2(\pi(n-L+D+0.5)/2D)$	$L \le n < D+L$
0	$D+L \le n < M$

เมื่อ: D คือ จำนวนจุดสัญญาณที่เกิดการเหลื่อมกันในเฟรมก่อนหน้า = 24

L คือ จำนวนจุดสัญญาณในหนึ่งเฟรม = 80

M คือ ความยาวของการประมวลผลแบบ FFT = 128

จำนวนเชิงซ้อนจากการแปลงฟูเรียร์แบบเร็ว 64 จุด จะถูกแปลงให้อยู่ในรูปของจำนวนจริง 128 จุด และกระบวนการนี้เองจะให้ผลลัพธ์เป็นเวกเตอร์จำนวนจริง 128 จุด ซึ่งกระบวนการนี้จะอธิบายต่อไป

ขั้นตอนของ FFT สามารถจัดการกับสัญญาณอินพุทเชิงซ้อนได้โดยการกระจายให้อยู่ในรูปการคำนวณของ DFT ในรูปจำนวนจริง 2 จำนวน : $x_1(n), x_2(n)$ บนความยาว N และ $x(n)$ คือจำนวนเชิงซ้อน ดังแสดงในสมการที่ (3.26)

$$x(n) = x_1(n) + jx_2(n) \qquad \text{สำหรับ } 0 \le n \le N-1 \tag{3.26}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการแปลงฟูเรียร์แบบไม่ต่อเนื่อง จะได้สมการที่ (3.27)

$$X(k) = X_1(k) + jX_2(k) \tag{3.27}$$

กระจาย $x_1(n)$ และ $x_2(n)$ ในรูปของ $x(n)$ ดังสมการที่ (3.28)

$$\begin{aligned} x_1(n) &= (x(n) + x^*(n))/2 \\ x_2(n) &= (x(n) - x^*(n))/2j \end{aligned} \tag{3.28}$$

ดังนั้น DFT ของ $x_1(n)$ และ $x_2(n)$ จะอยู่ในรูปดังสมการที่ (3.29)

$$\begin{aligned} X_1(n) &= \{DFT(x(n)) + DFT(x^*(n))\}/2 \\ X_2(n) &= \{DFT(x(n)) - DFT(x^*(n))\}/2j \end{aligned} \tag{3.29}$$

ให้ DFT ของ $x^*(n)$ คือ $X^*(N-k)$ จะได้สมการใหม่ดังสมการที่ (3.30)

$$\begin{aligned} X_1(k) &= \{X(k) + X^*(N-k)\}/2 \\ X_2(k) &= \{X(k) - X^*(N-k)\}/2j \end{aligned} \tag{3.30}$$

ให้ $g(n)$ เป็นค่าจริงของ $2n$ จุด เราจะแบ่ง $g(n)$ ให้อยู่ในจำนวน 2 จำนวนได้ดังสมการที่ (3.31)

$$\begin{aligned} x_1(n) &= g(2n) \\ x_2(n) &= g(2n+1) \end{aligned} \tag{3.31}$$

ดังนั้นค่าจริง $2n$ จุด จะสามารถกระจายให้อยู่ในรูปจำนวนจริงสองจำนวนตามวิธี การข้างต้น โดยให้ $x(n)$เป็นจำนวนเชิงซ้อน n จุดนั้นจะได้สมการที่ (3.31) และ สมการที่ (3.32) ตามลำดับ

$$x(n) = x_1(n) + jx_2(n) \tag{3.31}$$

$$\begin{aligned} X_1(k) &= \{X(k) + X^*(N-k)\}/2 \\ X_2(k) &= \{X(k) - X^*(N-k)\}/2j \end{aligned} \tag{3.32}$$

ท้ายที่สุดเราต้องกระจาย DFT $2n$ จุด ในรูป N จุด : $X_1(k)$ และ $X_2(k)$ ซึ่งในกระบวนการนี้เราจะใช้ FFT จัดการเพื่อประสิทธิ์ภาพในการคำนวณ ดังสมการที่ (3.33)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$G(k)=\sum_{n=0}^{n=N-1} g(2n)W_{2N}^{2nk}+\sum_{n=0}^{n=N-1} g(2n+1)W_{2N}^{(2n+1)k}$$
$$G(k)=\sum_{n=0}^{n=N-1} x_1(n)W_N^{nk}+W_{2N}^{k}\sum_{n=0}^{n=N-1} x_2(n)W_N^{nk} \qquad (3.33)$$

ดังนั้น จะได้

$$G(k)=X_1(k)+W_{2N}^{k}X_2(k)$$
$$G(k+N)=X_1(k)-W_{2N}^{k}X_2(k) \qquad \text{เมื่อ } k=0,1,...,N-1 \qquad (3.34)$$

เนื่องจาก FFT เป็นฟังก์ชันของเหตุการณ์จริง ดังนั้นผลลัพธ์เชิงซ้อน 128 จุดนั้น เราจึงสามารถคำนวณแค่ 67 จุดแรกได้

การคำนวณระดับพลังงาน

สเปกตรัมความถี่ (frequency spectrum) ของสัญญาณเสียงที่ทำ FFT แล้ว จะถูกแบ่งแถบพลังงานออกมาเป็น 16 ระดับ ซึ่งแต่ละระดับจะถูกคำนวณโดยใช้การรวมพลังงานที่รั่ว (leak integration) ดังแสดงในตารางที่ 3.5

ตารางที่ 3.5 แถบพลังงานของสเปกตรัมความถี่ (BAND ENERGY SIGNAL SPECTRUM)

แถบพลังงาน	ความถี่ขอบบน (FFT Index) FL(i)	ความถี่ขอบล่าง (FFT Index) FH(i)
1	2	3
2	4	5
3	6	7
4	8	9
5	10	11
6	12	13
7	14	16
8	17	19
9	20	22
10	23	26
11	27	30
12	31	35
13	36	41
14	42	48
15	49	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถบพลังงาน 'i' แต่ละระดับจะขึ้นอยู่กับการคำนวณในสมการที่ (3.35) ดังนี้

$$E_{ch}(m,i) = \max\left\{E_{\min}, \alpha_{ch} E_{ch}(m-1,i) + \left(1-\alpha_{ch}\right)\frac{\sum_{k=f_L(i)}^{k=f_H(i)} |G(k)|^2}{f_H(i) - f_L(i) + 1}\right\} \quad (3.35)$$

เมื่อ : α_{ch} = 0 สำหรับข้อมูลเฟรมแรก และ เท่ากับ 0.55 ในเฟรมอื่นๆ

$E_{\min}$ = 0.0625

$G(k)$ คือ FFT ของสัญญาณอินพุต

m คือ เฟรมข้อมูล

การคำนวณแถบ SNR

แต่ละแถบความถี่จะถูกคำนวณอัตราส่วนระหว่างสัญญาณเสียงพูด ต่อเสียงรบกวน (Signal to Noise Ratio: SNR) โดยแถบพลังงานจะถูกคำนวณแบบขั้นบันได และสัญญาณรบกวนแต่ละแถบความถี่จะถูกนำไปใช้ในการคำนวณ สเกลแฟกเตอร์ (scale factor) ซึ่งสัญญาณที่ได้นี้จะถูกอัพเดทในเฟรมสัญญาณรบกวนต่อไป

ถ้า $E_n(m,i)$ คือ พลังงานของเสียงรบกวนสำหรับเฟรม m แล้ว สเกลแฟกเตอร์ ของแถบพลังงาน i จะคำนวณได้ตามสมการที่ (3.36)

$$\sigma(i) = \max\left\{0, round\left(10\log\left(\frac{E_{ch}(m,i)}{E_n(m,i)}\right)\right)\right\} \quad (3.36)$$

การคำนวณสเกลแฟกเตอร์

*สเกลแฟกเตอร์*ของสัญญาณจะถูกนำไปคูณกับแถบความถี่ที่คำนวณเอาไว้ และทุกแถบความถี่ สเกลแฟกเตอร์จะแทนด้วยค่าคงที่ตั้งแต่ 15 dB จนถึงแถบ SNR ของแต่ละแถบความถี่ซึ่งสเกลแฟกเตอร์นี้ จะถูกเปลี่ยนจากสเกลในรูปของลอกการิทึม (logaithmic scale)ไปเป็นสเกลในรูปเชิงเส้น (linear scale)

ถ้า i เป็นแถบความถี่ของพลังงานในแต่ละระดับ สเกลแฟกเตอร์ของสัญญาณจะถูกคำนวณในสเกลลอกการิทึม ตามสมการที่ (3.37)

$$\gamma_{dB}^{i}(m) = \min(\sigma(i) - 15, 0) \quad (3.37)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และสมารถเปลี่ยนให้อยู่ในรูปของสเกลเชิงเส้น ตามสมการที่ (3.38)

$$\gamma_{ch}^{i}(m) = 10^{\gamma_{dB}^{i}(m)/20} \tag{3.38}$$

การคำนวณแยกแยะระหว่างเสียง สนทนาและเสียงรบกวน

แถบพลังงานในเฟรม และแถบพลังงานของเสียงรบกวนจะถูกนำมาเปรียบเทียบ เพื่อที่จะจัดจำแนกเฟรมนั้นว่าเป็น เฟรมของเสียงรบกวน หรือเฟรมของเสียงพูด เฟรมข้อมูล 6 เฟรม จะถูกนำไปใช้ในการส่งเฟรมจากเฟรมสัญญาณเสียงพูดถึงเฟรมสัญญาณเสียงรบกวน และถ้าในเฟรมนั้นถูกดำเนินการโดยฟังก์ชัน VAD แล้วแถบพลังงานของเสียงรบกวนจะถูกอัพเดท

สำหรับข้อมูล 12 เฟรมแรก แถบพลังงานของเสียงรบกวนจะถูกอัพเดทอย่างต่อเนื่องโดยไม่คำนึงถึงเสียงรบกวนและเสียงพูดระหว่างเฟรม

ตามขั้นตอนนี้จะถูกพิจารณาไม่ว่าเฟรมปัจจุบันนั้น จะเป็นเสียงรบกวนหรือไม่โดย

1. ให้ตัวนับเป็น 0
2. ถ้าในแถบพลังงานของเสียงมากกว่า 1.5 เท่าของแถบพลังงานเสียงรบกวน ตัวนับจะเพิ่มขึ้น 1 ค่า
3. ถ้าตัวนับมีค่ามากกว่า 5 ในเฟรมปัจจุบัน (เช่น ถ้า 5 ของ 16 แถบพลังงานมากกว่า 1.5 เท่าเหมือนกับแถบพลังงานเสียงรบกวน)แล้วเฟรมปัจจุบันนั้นจะลดค่าเป็นเฟรม สนทนา

นอกจากนั้นเฟรมปัจจุบันจะถูกพิจารณาเป็นเฟรมของสัญญาณรบกวน และ แถบพลังงานของสัญญาณรบกวน ซึ่งทั้งคู่จะถูกอัพเดท โดยใช้วิธี รวมพลังงานที่รั่ว (leaky integration) ดังสมการที่ (3.39)

$$E_n(m+1,i) = \max\{E_{\min}, (\alpha_n E_n(m,i) + (1-\varepsilon_n)E_{cn}(m,i))\} \tag{3.39}$$

เมื่อ $0 \le i < Nc$

Nc คือ ลำดับสุดท้ายของแถบพลังงาน i

การปรับสเกลของแถบความถี่

ผลลัพธ์ของการแปลง FFT ที่ยังไม่ได้จัดจำแนกแถบพลังงานจะถูกคูณด้วยอัตราส่วนที่เหมาะสมของสเกลแฟกเตอร์ และจากกระบวนการนี้เองจะเป็นการกรองเสียงรบกวนสำหรับทุกองค์ประกอบของความถี่ k ที่ยังไม่ได้จัดจำแนกแถบพลังงาน i ผลลัพธ์ของการสเกลจะได้ $H(k)$ ดังสมการที่ (3.40)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned} H(k) &= \gamma_{ch}(i)G(k); && f_L(i) \le k \le f_H(i), 0 \le i < Nc \\ H(k) &= G(k); && 0 \le f_L(0), f_H(Nc-1) < k \le m/2 \end{aligned} \quad (3.40)$$

จากสมการที่ (3.40) จะสังเกตว่า $H(k)$ จะไม่เปลี่ยนตามสเกลแฟกเตอร์ ยิ่งไปกว่านั้น เนื่องจาก $H(k)$ ต้องการขนาดซึ่งเป็นฟังก์ชันจำนวนคู่ และเฟสเป็นฟังก์ชันจำนวนคี่ ซึ่งตามข้อจำกัดนี้จะถูกนำมาดัดแปลงมาใช้ในการคำนวณข้างล่างดังนี้

$$H(M-k) = H^*(k); \quad (3.41)$$

เมื่อ $0 \le k < M/2$

การแปลงกลับของสัญญาณในโดเมนของเวลา

ท้ายที่สุดองค์ประกอบความถี่ของสัญญาณจะถูกแปลงกลับ ในโดเมนของเวลาตามเดิม โดย 48 จุดแรกสัญญาณของ FFT จะถูกเพิ่มเข้าไปใน 48 จุดสัญญาณของเฟรมสุดท้าย การแปลงกลับของสัญญาณ จะอธิบายได้ดังสมการที่ (3.42) ดังนี้

$$\begin{aligned} h'(n) &= h(m,n) + h(m-1, n+L) && 0 \le n < M-L \\ h'(n) &= h(m,n) && M-L \le n < L \\ s''n &= h'(n) + \xi s''(n-1) && 0 \le n < L \end{aligned} \quad (3.42)$$

เมื่อ $s''(n)$ คือ สัญญาณเสียงพูดที่กำจัดสัญญาณรบกวนแล้ว

ξ คือ ค่าคงที่เท่ากับ 0.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 โปรแกรมการทำงาน

สำหรับโครงงานนี้ เราสามารถรับเสียงจากไฟล์ในคอมพิวเตอร์ หรือทางไมโครโฟน สัญญาณเสียงที่มีเสียงรบกวนจะถูกสุ่มตัวอย่างผ่านทางไอซี Si3000 และโมดูล DCI ของไมโครคอนโทรลเลอร์ dsPIC30F6014 ส่วนสัญญาณเสียงที่กำจัดเสียงรบกวนแล้ว จะส่งสัญญาณออกมาผ่านทางโมดูล DCI และไอซี Si3000

ในหัวข้อนี้จะแสดงรูปแผนผังการทำงานของโปรแกรมโดยรวม และรูปแสดงการทำงานภายในโปรแกรมอย่างละเอียด ในรูปที่ 3.9 แสดงถึงภาพรวมในการทำงานของฟังก์ชัน main() ซึ่งจัดเป็นฟังก์ชันหลักของโปรแกรม ขั้นตอนการทำงานหลักภายในโปรแกรมแบ่งออกเป็น 4 ขั้นตอน ได้แก่

1. ตรวจสอบการกดปุ่ม SW 1-4 เพื่อเลือกโหมดการทำงานได้แก่ โหมด A และโหมด B
2. กำหนดค่าเริ่มต้นในการกำจัดสัญญาณเสียงรบกวน ด้วยฟังก์ชัน NoiseSuppressionInit()
3. กำหนดค่าเริ่มต้นในการเชื่อมต่ออุปกรณ์ภายนอกได้แก่
   - ฟังก์ชัน InitSPI() ใช้ในการกำหนดค่าเริ่มต้นโมดูล SPI เพื่อทำการเชื่อมต่อกับ LCD
   - ฟังก์ชัน Init_DCI()ใช้ในการกำหนดค่าเริ่มต้นโมดูล DCI เพื่อทำการเชื่อมต่อกับ Si3000
   - ฟังก์ชัน Init_Si3000()ใช้ในการกำหนดค่าเริ่มต้นภายในรีจิสเตอร์ของ Si3000
4. เริ่มต้นการทำงานเข้าและถอดรหัสเสียง ภายในจะมี ฟังก์ชันวนรอบไม่รู้จบ while(start) ตัวแปร intr_count จะถูกนับค่าเพิ่ม ก็ต่อเมื่อโมดูล DCI เกิดการอินเตอร์รัปต์ เมื่ออินเตอร์รัปต์ครบ 80ครั้ง ก็จะเข้าสู่การตรวจสอบโหมดการทำงาน ในกรณีที่ทำงานในโหมด A ตัวแปร runnsflag จะถูกกำหนดให้เป็น “1” ถ้าทำงานในโหมด B ตัวแปร runnsflag จะยังคงเท่ากับ “0” นั่นหมายความว่า โหมด A จะประมวลผลสัญญาณเสียงแล้วกำจัดเสียงรบกวนออกด้วยฟังก์ชัน NoiseSuppression() ในขณะที่โหมด B จะเป็นเพียงการเข้าและถอดรหัสเสียง

ในรูปที่ 3.10 ถึง 3.14 จะแสดงถึงแผนผังการทำงานอย่างละเอียดของโปรแกรม ได้แก่ ฟังก์ชัน InitSPI(), ฟังก์ชัน Init_DCI(), ฟังก์ชัน DCIInterupt(), ฟังก์ชัน Init_Si3000() และโปรแกรมภาษาซี ns_8k.c ตามลำดับ

โปรแกรมภาษาซีนี้ ทำงานบน dsPIC30F6014 ใช้ระบบสัญญาณนาฬิกาในโหมด XT ร่วมกับ ตัวคูณเฟสล็อกลูป 8 เท่า

ไฟล์ ns_8k.c จะบรรจุฟังชันหลักไว้ภายใน และเรียกตัวแปรทุกตัวรวมถึงตัวแปรอะเรย์ที่อยู่ในหน่วยความจำข้อมูลซึ่งจำเป็นต่อการรับส่งข้อมูลผ่านทางโมดูล DCI ตัวแปร CodecRxBuffer และCodecTxBuffer เป็นบัฟเฟอร์ในการรับและส่งข้อมูลสัญญาณเสียงที่เข้ารหัส แต่ละตัวแปรมีขนาด 240 เวิร์ด (ขนาดเฟรม 10 มิลลิวินาที จำนวน 3 เฟรม สุ่มจุดตัวอย่างที่ความถี่ 8 kHz ตัวแปร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CodecTxRIndex, CodecTxWIndex, CodecRxRIndex และCodecRxWIndex เป็น ดรรชนีการเขียนและอ่าน สำหรับการส่งและรับสัญญาณเสียงที่เข้ารหัส ดรรชนีการอ่านจะตามหลังดรรชนีการเขียนอยู่ 160 จุดตัวอย่างสัญญาณ เพื่อให้มั่นใจว่าข้อมูลถูกอ่านหลังจากข้อมูลถูกเขียน

โปรแกรมจะเริ่มเข้าสู่ฟังก์ชัน NoiseSuppressionInit() สำหรับการกำหนดค่าเริ่มต้นในการเข้าสู่กระบวนการกำจัดสัญญาณเสียงรบกวน หลังจากนั้นจะเข้าสู่ฟังก์ชัน Init_DCI() เพื่อกำหนดค่าเริ่มต้นของโมดูล DCI โดยโมดูล DCI จะทำหน้าที่เป็นอุปกรณ์สเลฟ ส่วนไอซี Si3000 จะทำหน้าที่เป็นอุปกรณ์มาสเตอร์ และป้อนสัญญาณนาฬิกาให้โมดูล DCI ที่ถูกกำหนดให้ทำงานในแบบสเลฟเฟรมซิงค์ชนิดหลายช่องทาง กำหนดขนาดเวิร์ดข้อมูล เป็น 16 บิต จำนวน 16 ไทม์สลอต ต่อหนึ่งเฟรม ซึ่งในโครงงานนี้ ใช้เพียง 1 สลอตส่งข้อมูล(transmit slot) และ 1 สลอตรับข้อมูล(receive slot)

ต่อมาจะเข้าสู่ฟังก์ชัน Init_Si3000() ซึ่งถูกใช้ในการกำหนดค่าเริ่มต้นให้กับไอซี Si3000 ดังนี้ ความถี่สุ่มตัวอย่างเท่ากับ 8 kHz MIC Gain และ Receive Gain ถูกกำหนดเป็น 0 dB Transmit Gain เป็น 12 dB พารามิเตอร์ Analog Attenuation ถูกกำหนดเป็น 0 dB หลังจากที่กำหนดค่าในรีจิสเตอร์เสร็จสิ้นแล้ว จะทำการหน่วงเวลาเพื่อให้เกิดการ calibrate และท้ายที่สุดจะทำการ เอนเอเบิล อินเตอร์รัปต์จาก DCI (DCI Interrupt)

ตัวแปร intr_count จะถูกนับเพิ่มค่าขึ้นทุกๆครั้งที่เกิดอินเตอร์รัปต์จาก DCI เมื่อนับครบ 80 ครั้ง แล้ว สิ่งที่บรรจุอยู่ในแหล่งพักสำรองข้อมูล (codec data buffer) จะถูกคัดลอกลงไปในตัวแปรอะเรย์ sig_in และฟังชัน NoiseSuppression() จะนำค่าที่เก็บไว้ในตัวแปร sig_in เป็น เฟรมข้อมูลอินพุท ภายในการประมวลผล ฟังก์ชัน NoiseSuppression เสร็จสิ้นแล้ว ข้อมูลเอาต์พุทจะถูกเก็บไว้ในตัวแปร sig_in และเสียงจะถูกขับออกมาทางลำโพง

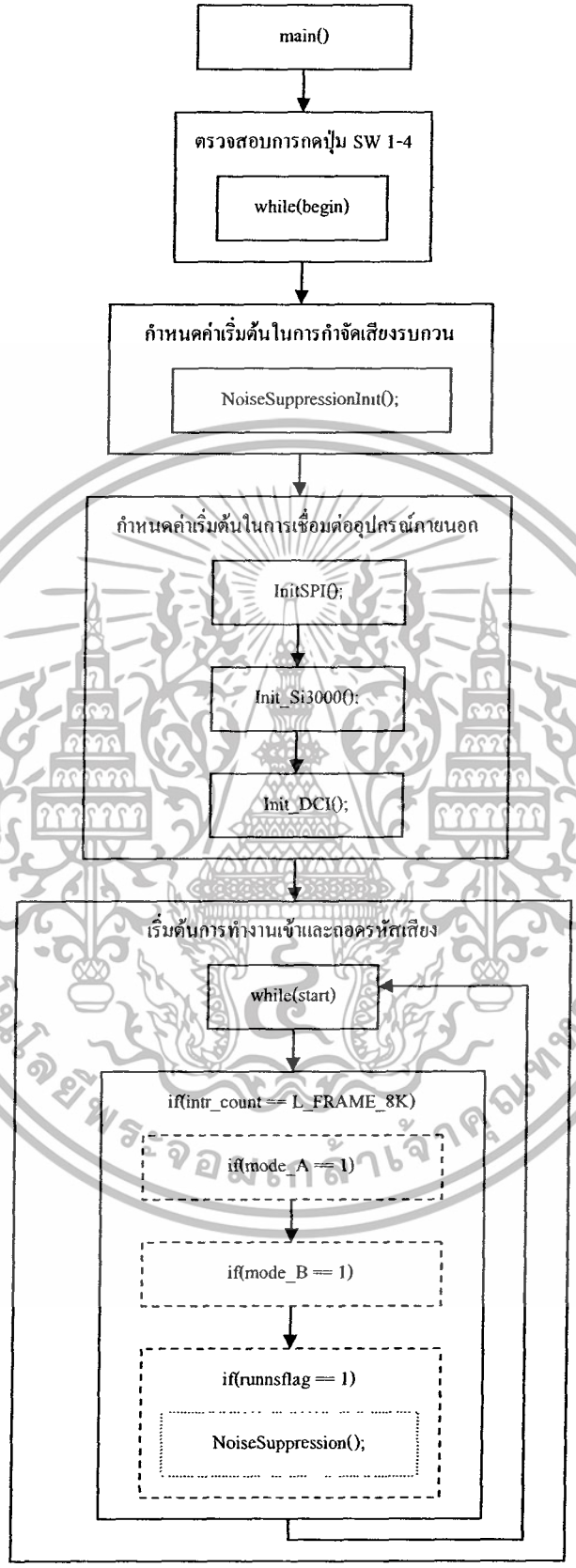
ฟังก์ชัน NoiseSuppression() จะทำงานก็ต่อเมื่อกำหนดให้โปรแกรมทำงานในโหมด A เมื่อเริ่มการทำงาน LED ดวงที่ 1 จะสว่าง เสียงสนทนาที่ได้ออกมาจะมีชัดเจนกว่าสัญญาณเสียงเดิม

ในการแสดงผลข้อความทาง LCD จำต้องกำหนดค่าเริ่มต้นผ่านทางฟังก์ชัน Init_SPI() ก่อนทำการแสดงข้อความใดๆบนหน้าจอ LCD

ตัวแปรบิต runnsflag จะถูกตรวจสอบว่า พร้อมในการเข้าสู่ฟังก์ชัน NoiseSuppression() หรือไม่ ถ้าบิตนี้ถูกกำหนดเป็น “1” สัญญาณเสียงที่ออกมาจะเป็นสัญญาณเสียงที่ถูกกำจัดเสียงรบกวนออกไปแล้ว

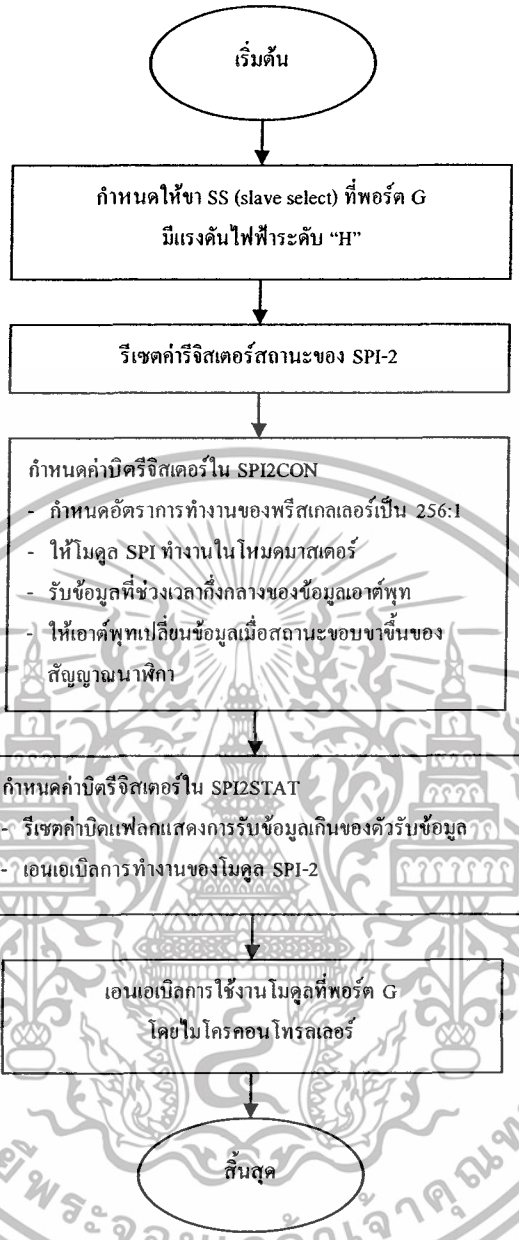
สำหรับในโหมด B จะเป็นเพียงการเข้าและถอดรหัสเสียงที่ไม่ได้เข้าสู่กระบวนการกำจัดเสียงรบกวน และโปรแกรมหลัก main() นี้จะวนอยู่ในลูป while(start) ไม่รู้จบ เพื่อทำการประมวลผลสัญญาณเสียงอย่างต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แผนผังการทำงานโดยรวมของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

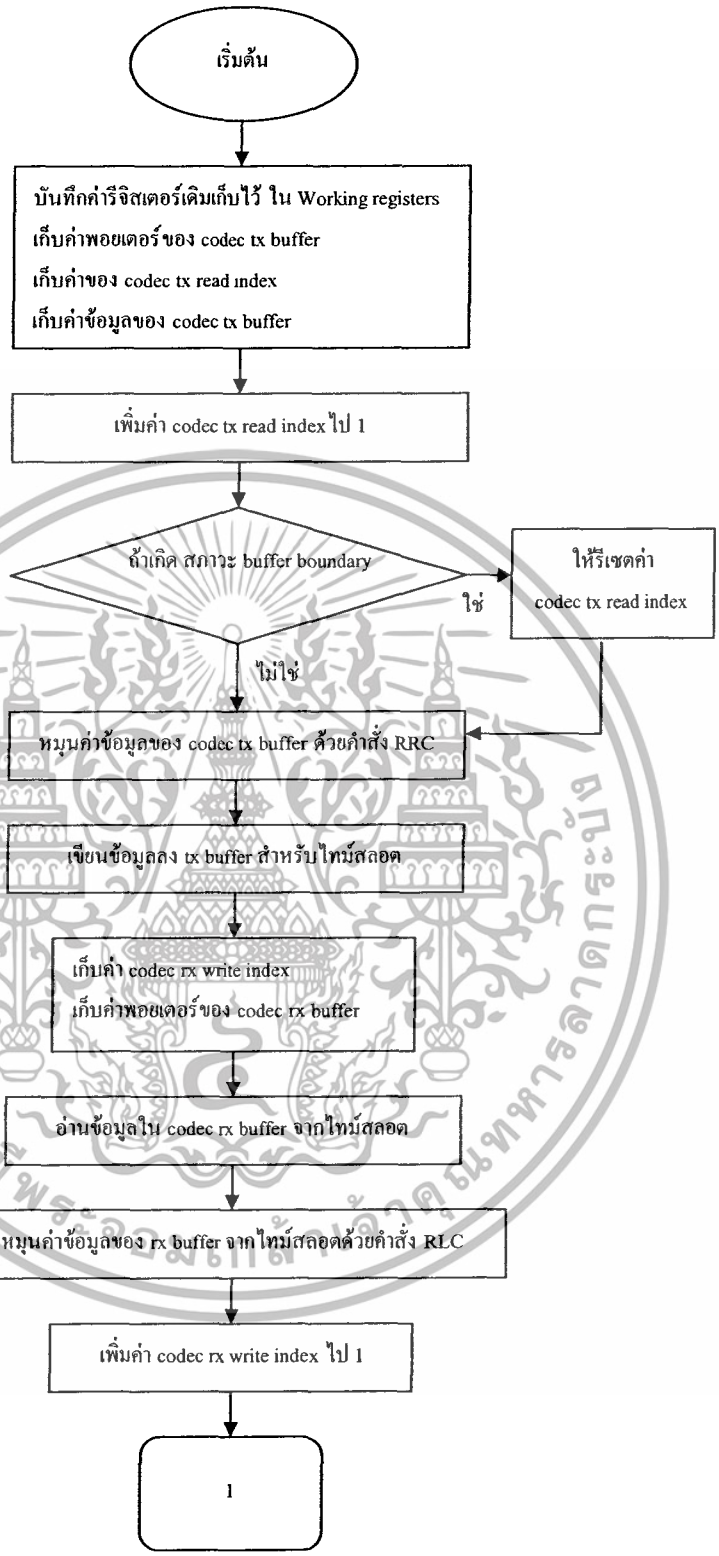


รูปที่ 3.10 แผนผังการทำงานของฟังก์ชัน InitSPI()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

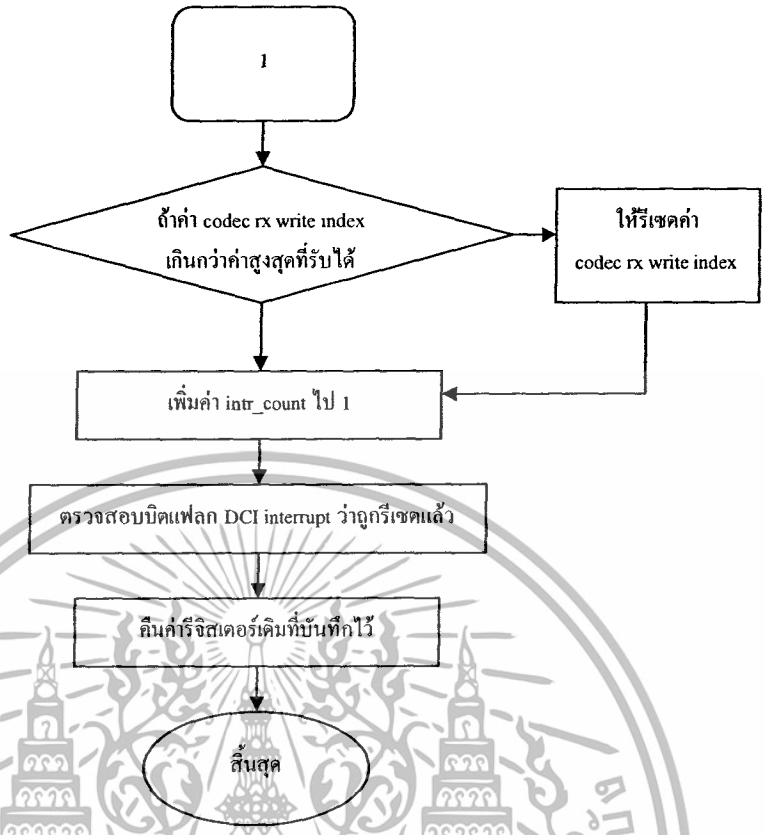
รูปที่ 3.11 แผนผังการทำงานของฟังก์ชัน Init_DCI()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



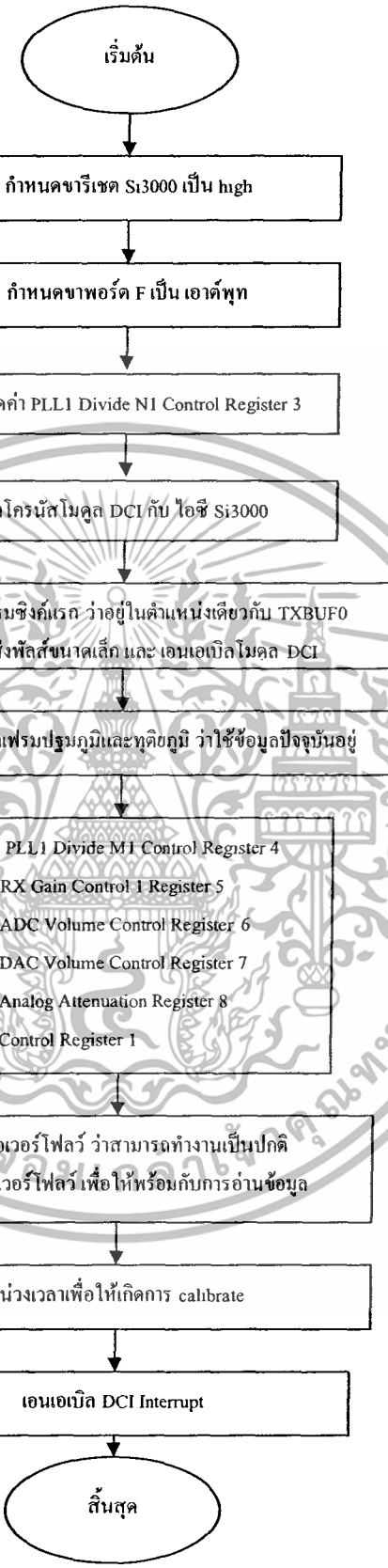
รูปที่ 3.12(A) แผนผังการทำงานของฟังก์ชัน DCI Interrupt()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



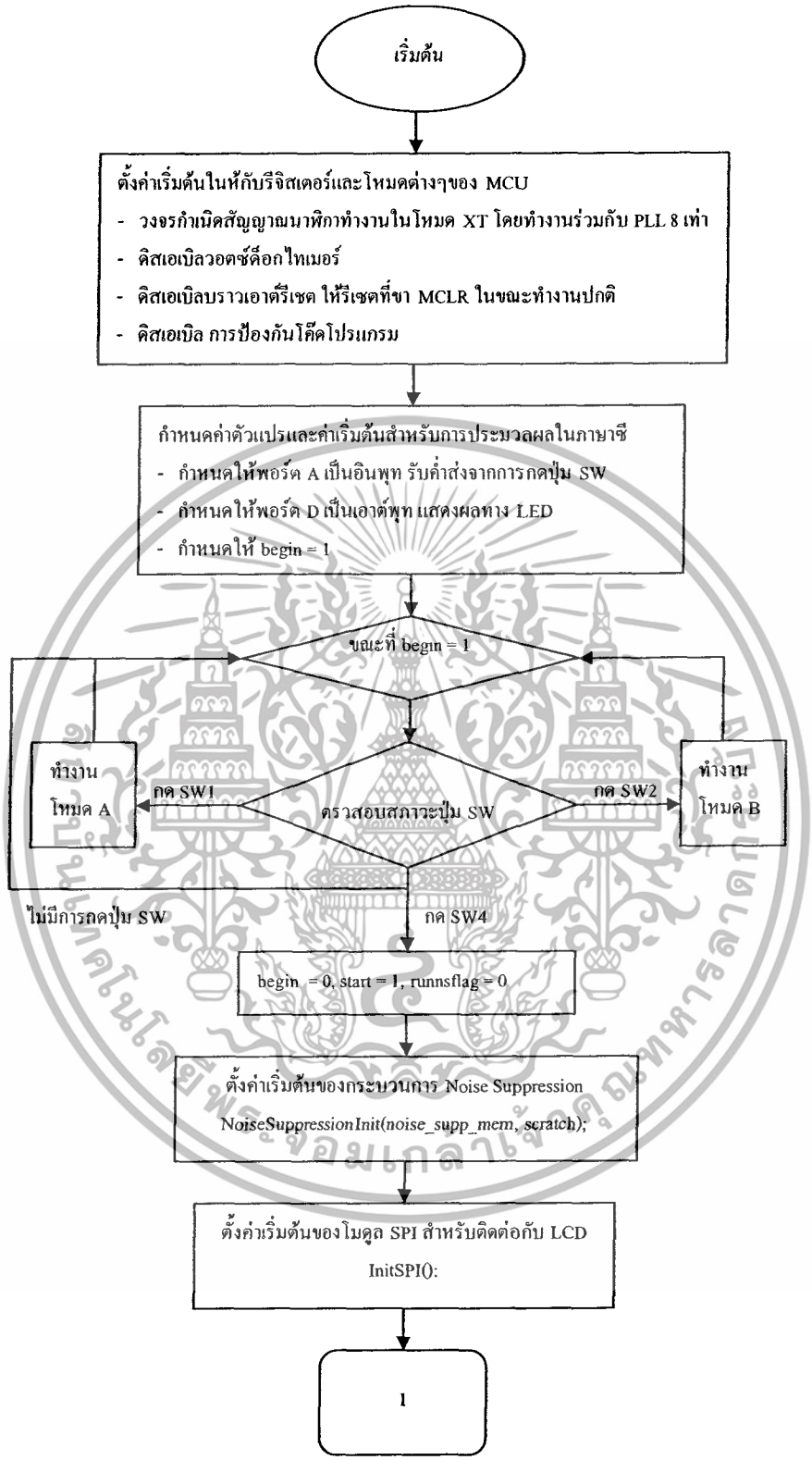
รูปที่ 3.12(B) แผนผังการทำงานของฟังก์ชัน DCI Interrupt()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



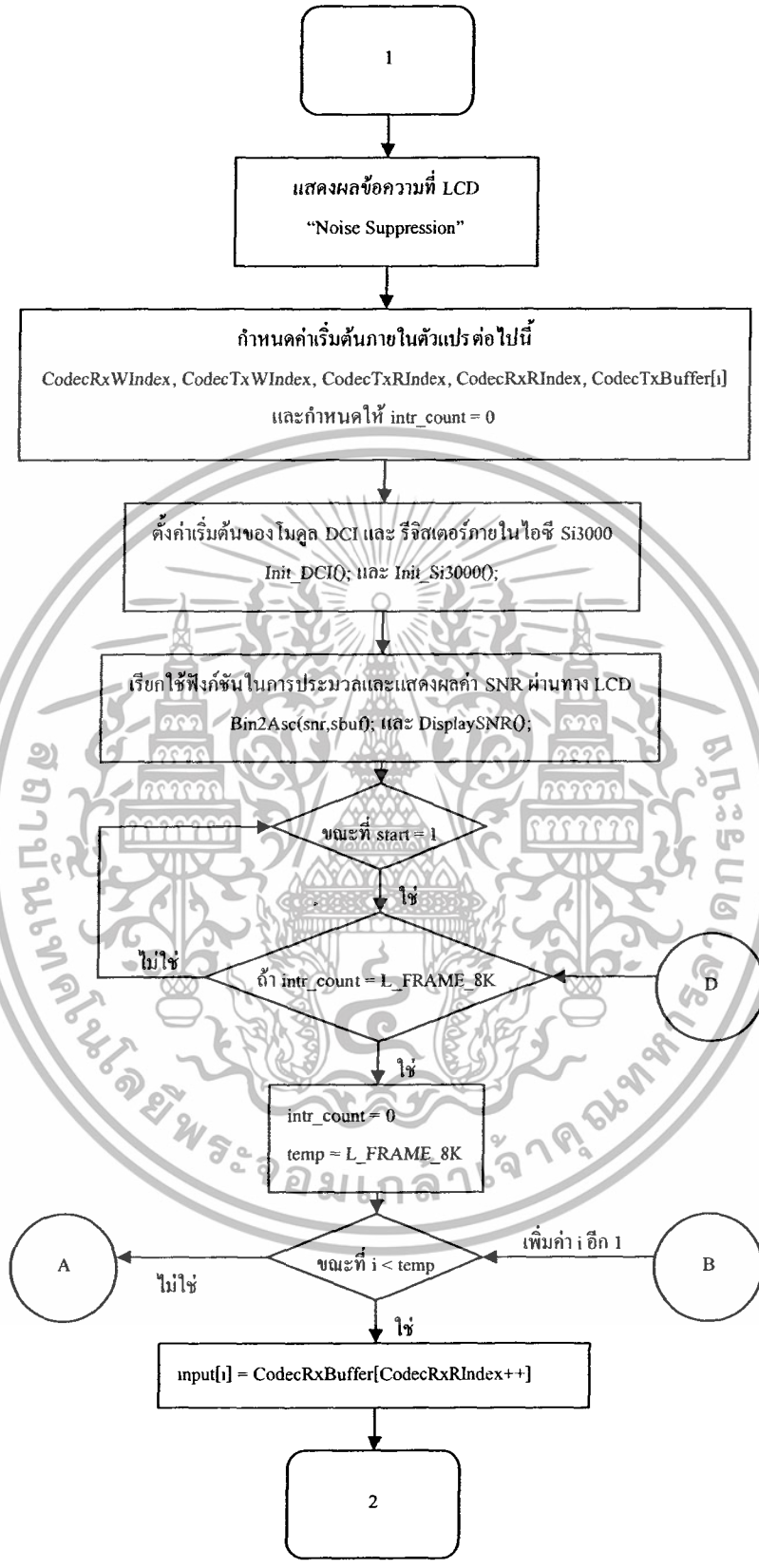
รูปที่ 3.13 แผนผังการทำงานของฟังก์ชัน Init_Si3000()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



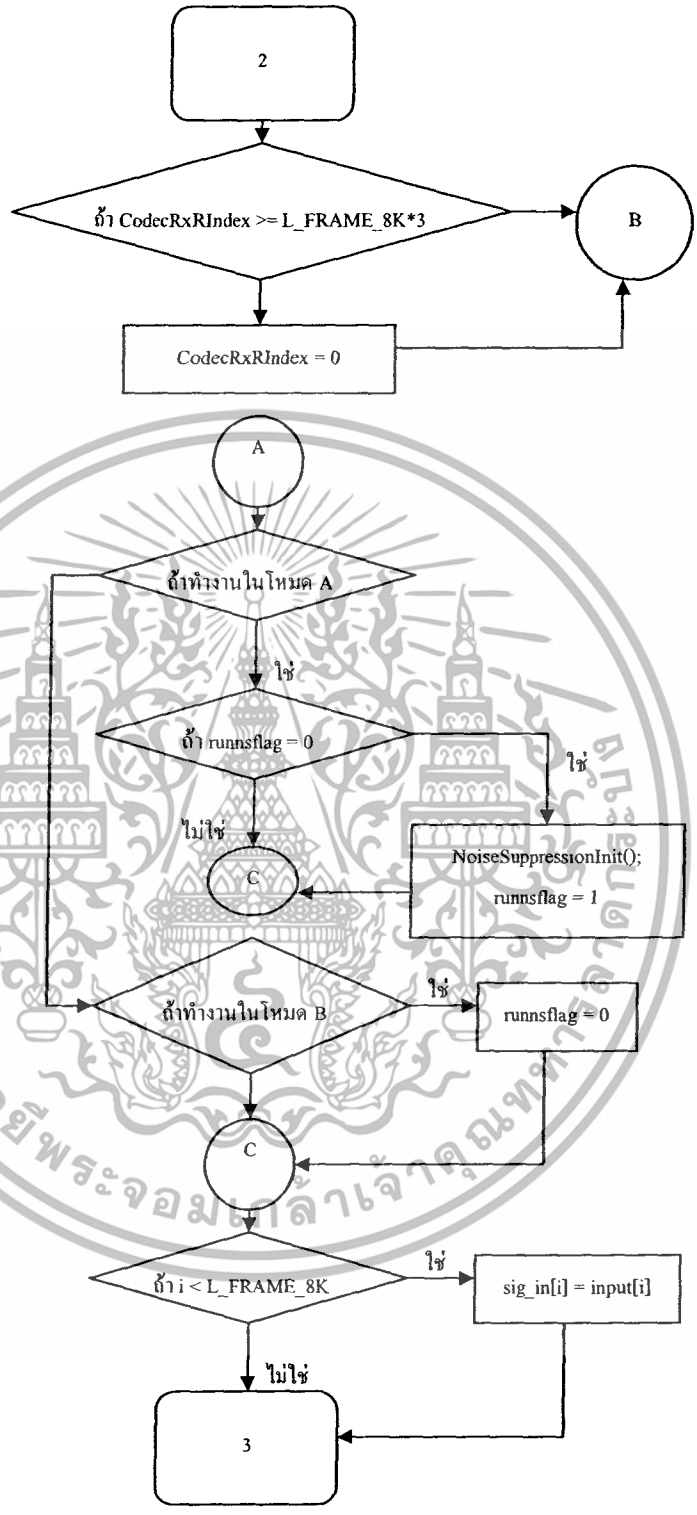
รูปที่ 3.14(A) แผนผังการทำงานของโปรแกรมภาษาซี ns_8k.c

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14(B) แผนผังการทำงานของโปรแกรมภาษาซี ns_8k.c

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

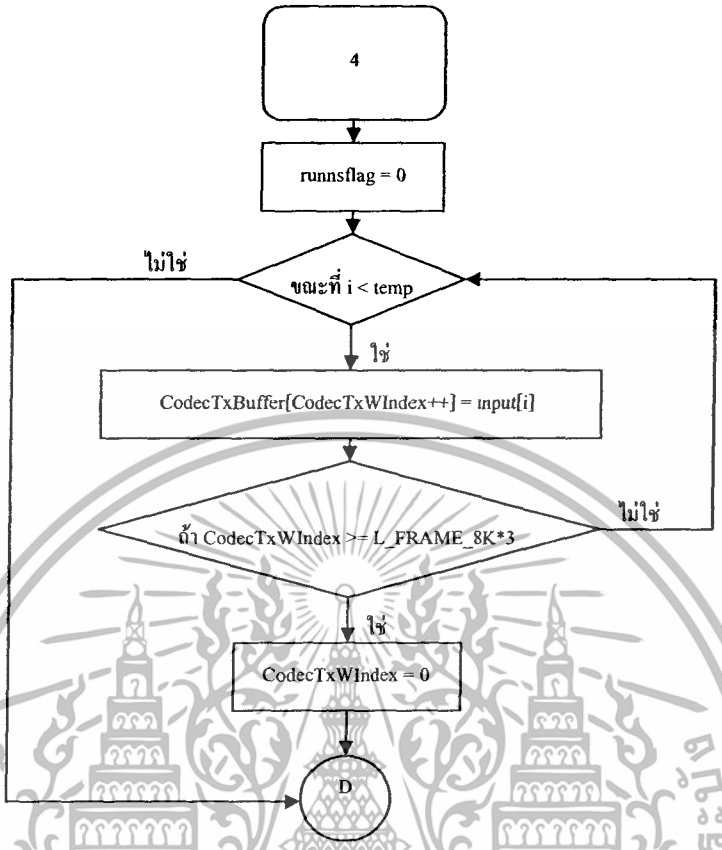


รูปที่ 3.14(C) แผนผังการทำงานของโปรแกรมภาษาซี ns_8k.c

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.14(D) แผนผังการทำงานของโปรแกรมภาษาซี ns_8k.c

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14(E) แผนผังการทำงานของโปรแกรมภาษาซี ns_8k.c

สำหรับในบทที่ 3 นี้ ได้นำเสนอถึงหลักการออกแบบโครงงาน ซึ่งต้องพิจารณาถึงความสามารถต่างๆทางด้านฮาร์ดแวร์ก่อนเริ่มการเขียนโปรแกรมใช้งานจริง สำหรับในบทต่อไปจะอธิบายถึงขั้นตอนการทดลองการทำงานของโปรแกรมและผลลัพธ์ที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลอง

ในบทนี้จะกล่าวถึงการทดลองและผลการทดลอง ซึ่งในโครงงานนี้ ผู้จัดทำได้เก็บตัวอย่างเสียงสนทนาที่มีเสียงรบกวนสองเสียง คือ เสียงสนทนาที่มีเสียงรบกวนจากเสียงรถไฟ และเสียงสนทนาที่มีเสียงรบกวนจากรถมอเตอร์ไซด์ จากนั้นทำการวิเคราะห์ตัวอย่างเสียงสนทนานั้นด้วยโปรแกรม " matlab " ซึ่งเป็นโปรแกรมที่นิยมใช้กันอย่างแพร่หลาย เพื่อพล็อตผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนาทั้งสอง ผู้จัดทำได้ทำการศึกษาเปรียบเทียบ ตามวัตถุประสงค์ดังนี้

4.1 วัตถุประสงค์ของการทดลอง

1. ศึกษาและเปรียบเทียบผลของการเปลี่ยนค่าเฟสล็อกลูป ซึ่งใช้ในการเชื่อมต่อแบบ DCI ระหว่าง ไอซีเข้าและถอดรหัสเสียง Si3000 กับไมโครคอนโทรลเลอร์ โดยเลือกใช้ออสซิลเลเตอร์จากภายนอก และกำหนดค่ารีจิสเตอร์ควบคุมจากการคำนวณ ในบทที่ 3 ดังแสดงในตารางที่ 4.1

ตารางที่ 4.1 ความถี่ออสซิลเลเตอร์ และค่ารีจิสเตอร์ควบคุมที่ใช้ในการเปลี่ยนค่าเฟสล็อกลูป

ความถี่ออสซิลเลเตอร์ (MHz)	รีจิสเตอร์ควบคุม N1	รีจิสเตอร์ควบคุม M1
16.384	4	10
16.384	2	5

2. ศึกษาและเปรียบเทียบผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลา ของเสียงสนทนา ระหว่างเสียงสนทนาก่อนเข้าสู่ระบบที่ถูกบันทึกโดยตรง และเสียงสนทนาที่ถูกบันทึกไว้หลังผ่านระบบแต่ยังไม่มีการกำจัดเสียงรบกวน

3. ศึกษาและเปรียบเทียบผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนา ระหว่างเสียงสนทนาหลังเข้าสู่ระบบที่ถูกบันทึกโดยตรง และเสียงสนทนาที่ถูกบันทึกไว้หลังผ่านระบบที่มีการกำจัดเสียงรบกวนแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้จัดทำได้ทำการออกแบบการทดลองออกเป็นสองตอน เพื่อศึกษาผลของการเปลี่ยนค่าเฟสล็อกลูปต่อผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนา เมื่อใช้ความถี่สุ่มตัวอย่างที่ 8 KHz ดังนี้

ตอนที่ 1 .ใช้ความถี่ออสซิลเลเตอร์16.384MHz รีจิสเตอร์ควบคุม N1 : M1 คือ 4:10

ตอนที่ 2 .ใช้ความถี่ออสซิลเลเตอร์16.384MHz รีจิสเตอร์ควบคุม N1 : M1 คือ 2: 5

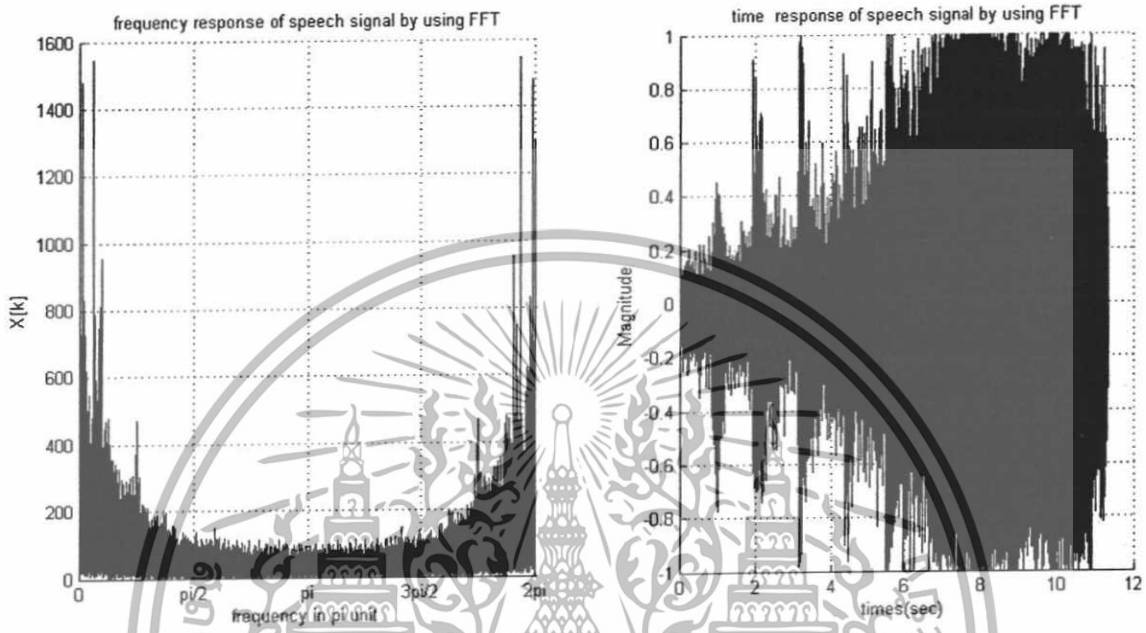
4.2 วิธีการทดลอง

1. บันทึกเสียงสนทนาที่มีเสียงสัญญาณรบกวน 2 เสียง โดยใช้โปรแกรม Wave Pad V2.0 บันทึกเป็นไฟล์เสียง PCM ออดิโอแซมเปิ้ล 16 บิต ความถี่สุ่มตัวอย่าง 8 kHz
2. เปิดเสียงสนทนาที่บันทึกไว้ในข้อ 1 จากนั้นต่อสายออดิโอของเครื่องเล่นเสียงเข้ากับบอร์ดทดลอง (mic in : J16)
3. เลือกโปรแกรมในโหมดที่ 1 (ยังไม่เปิดระบบกำจัดเสียงรบกวน) โดยใช้ความถี่ออสซิลเลเตอร์16.384MHz รีจิสเตอร์ควบคุม N1 : M1 คือ 4:10
4. พล็อตผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนา ด้วยโปรแกรม “matlab”
5. บันทึกเสียงสนทนาด้วยโปรแกรม Wave Pad เช่นเดียวกับข้อ 1
6. เลือกโปรแกรมในโหมดที่ 2 (เปิดระบบกำจัดเสียงรบกวน) จากนั้นทำการทดลองในข้อ 4,5
7. เปลี่ยนรีจิสเตอร์ควบคุม N1 : M1 เป็น 2:5 จากนั้นทำการทดลองในข้อ 3,4,5,6 อีกครั้ง

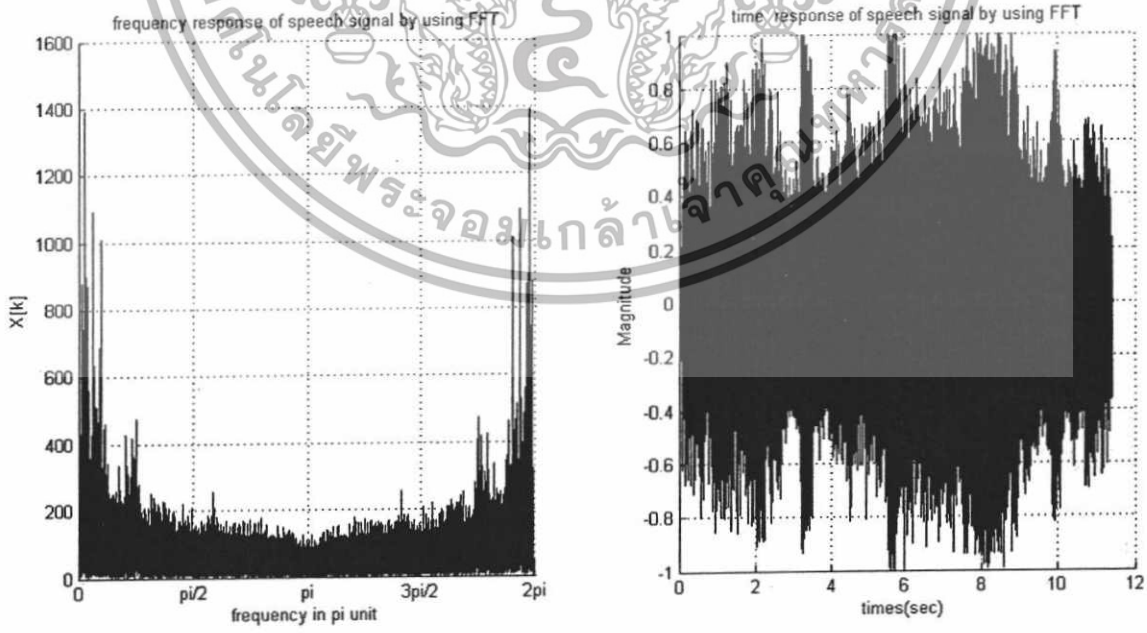
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลอง

ตอนที่ 1 ศึกษาผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนาที่มีเสียงรบกวนจากเสียงรถไฟ เมื่อกำหนดให้ความถี่ออสซิลเลเตอร์16.384 MHzรีจิสเตอร์ควบคุม N1:M1คือ 4:10

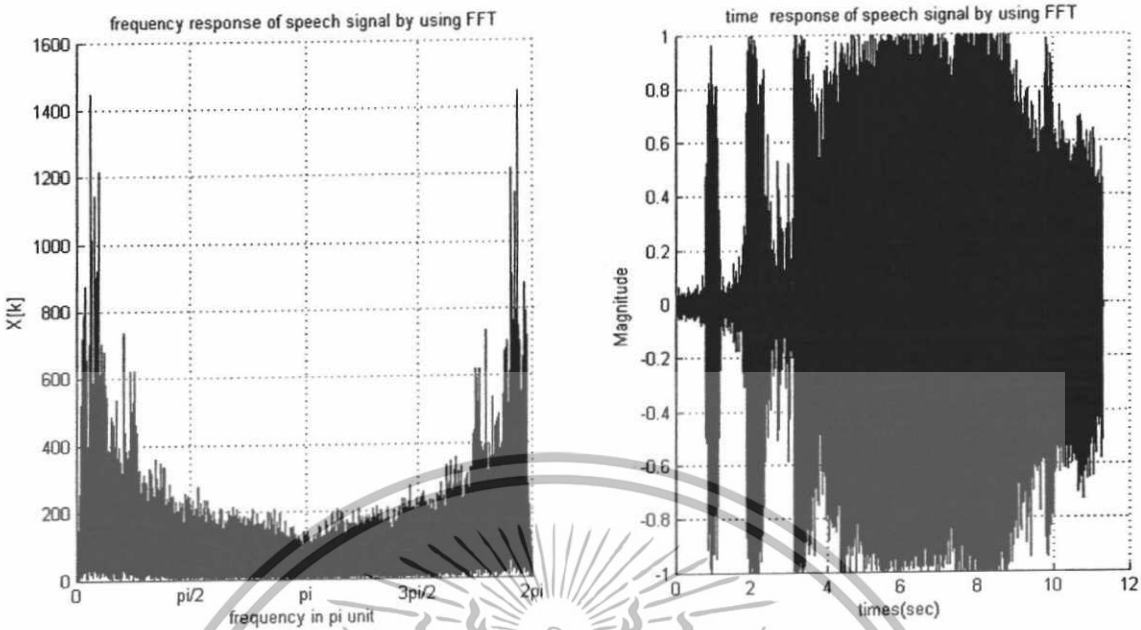


รูปที่ 4.1 เสียงสนทนาก่อนเข้าสู่ระบบ



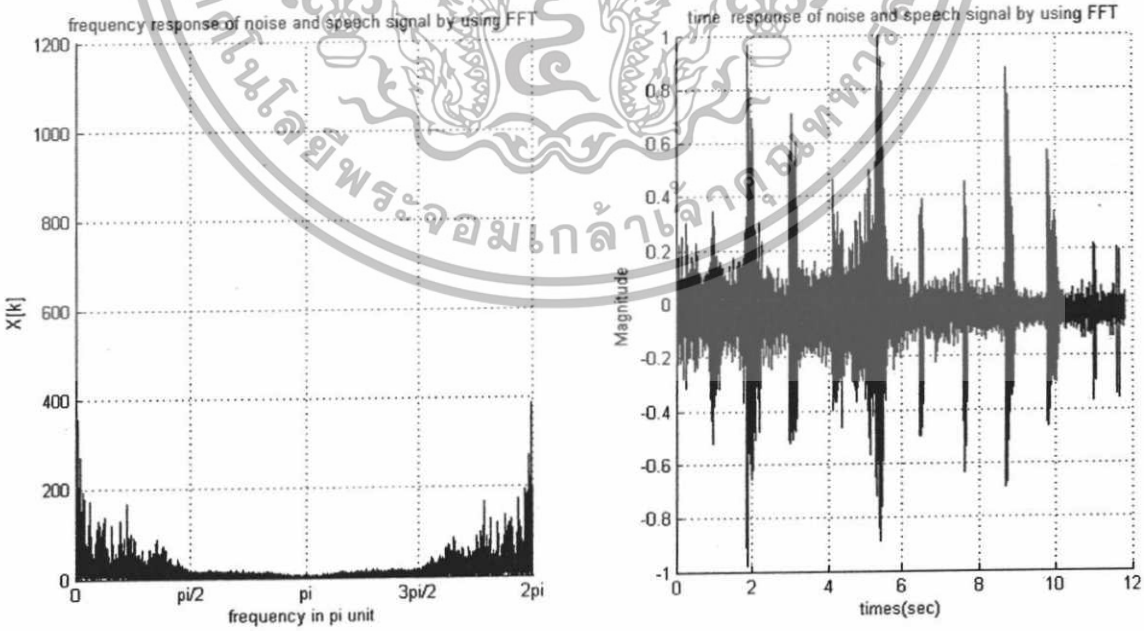
รูปที่ 4.2 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



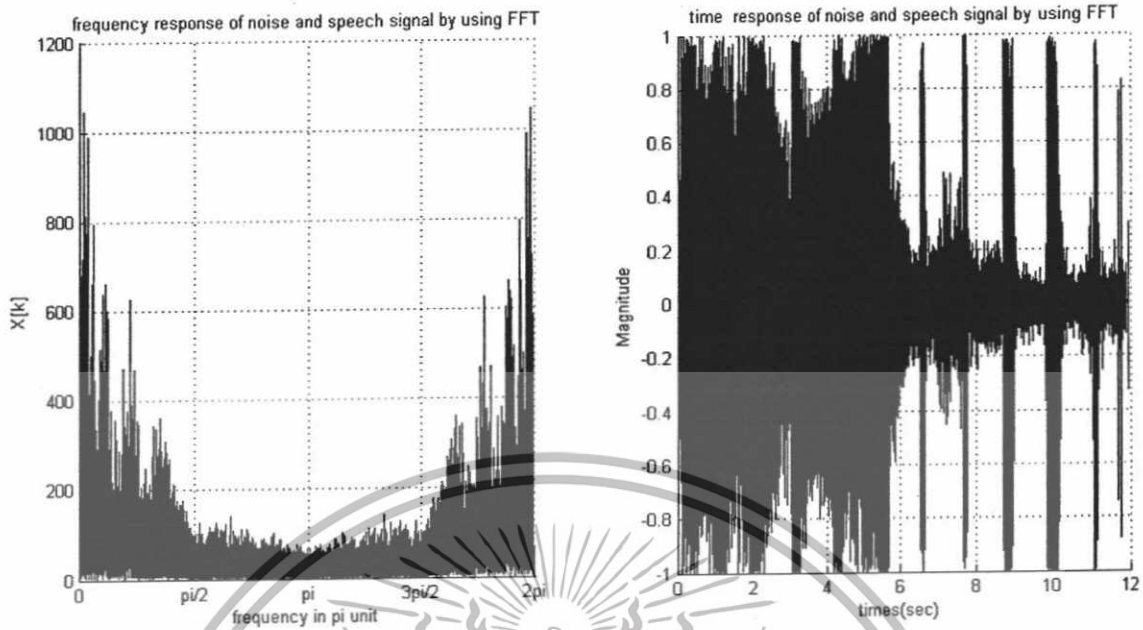
รูปที่ 4.3 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว

ตอนที่ 1(ต่อ) ศึกษาผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลา ของเสียงสนทนาที่มีเสียงรบกวนจากรถมอเตอร์ไซค์ เมื่อความถี่ออสซิลเลเตอร์ 16.384 MHz รีจิสเตอร์ควบคุม N1:M1คือ 4:10

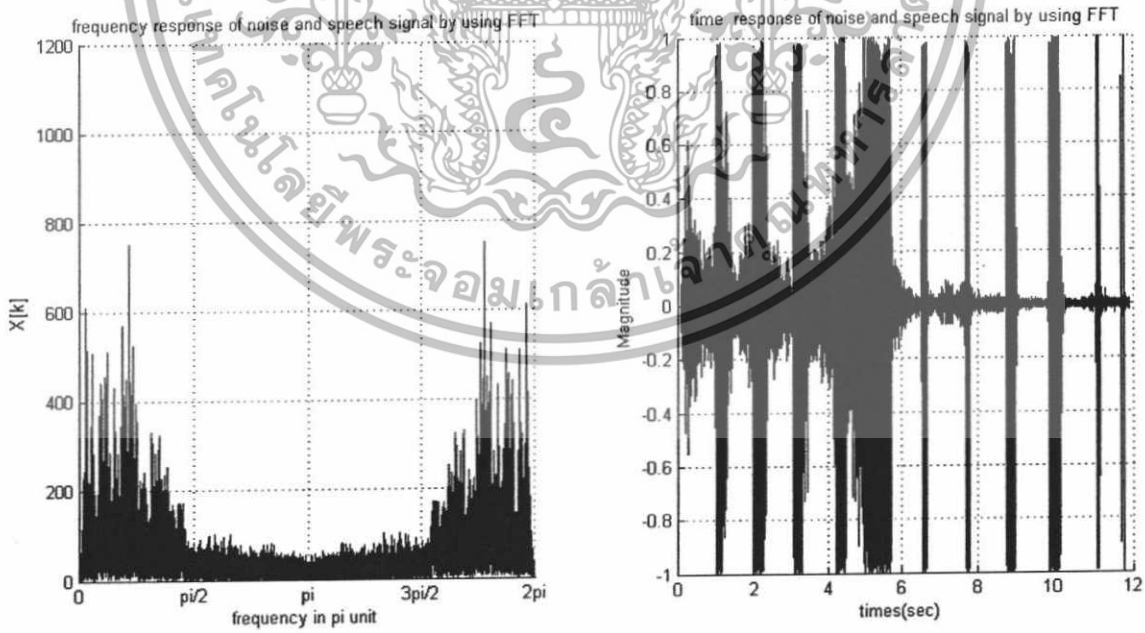


รูปที่ 4.4 เสียงสนทนา ก่อนเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



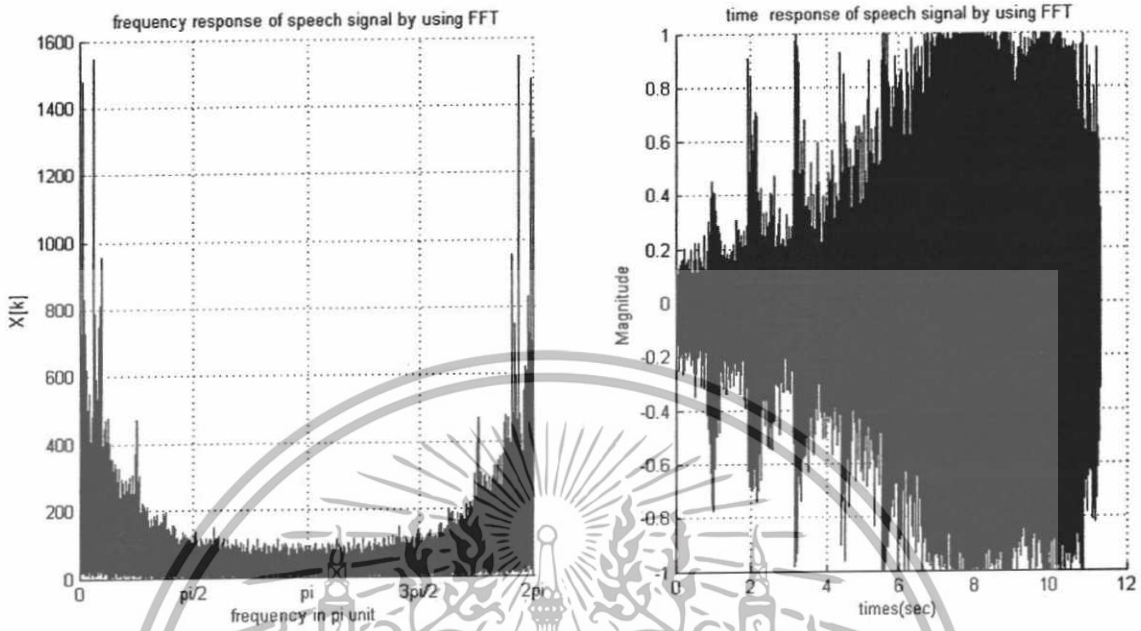
รูปที่ 4.5 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน



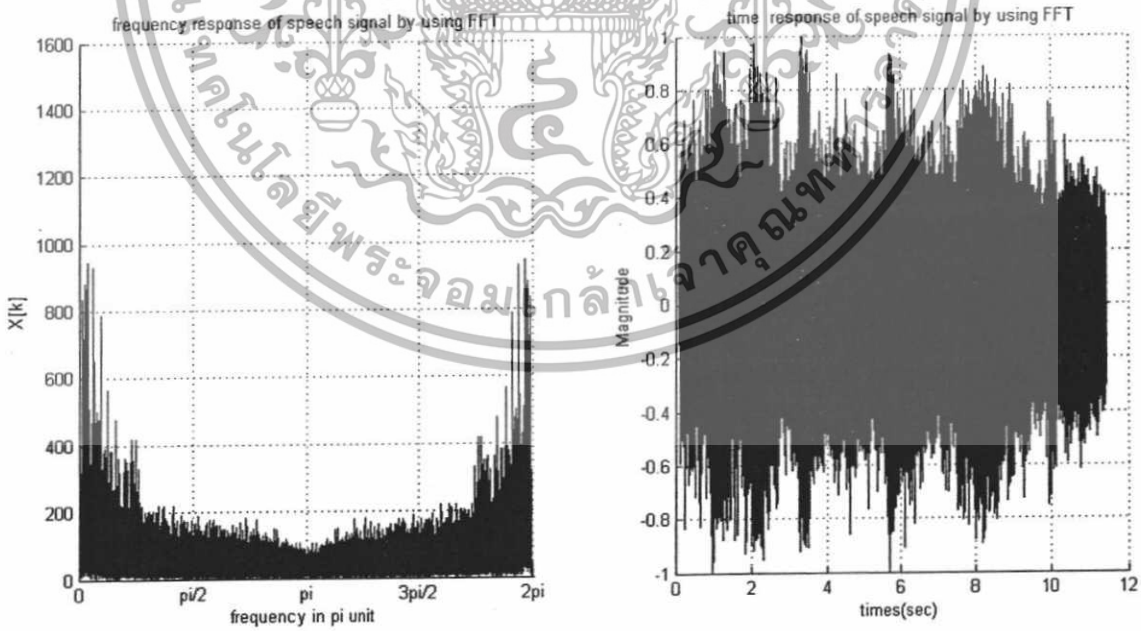
รูปที่ 4.6 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนที่ 2 ศึกษาผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนาที่มีเสียงรบกวนจากเสียงรถไฟ เมื่อกำหนดให้ความถี่ออสซิลเลเตอร์ 16.384 MHz รีจิสเตอร์ควบคุม N1:M1คือ2:5

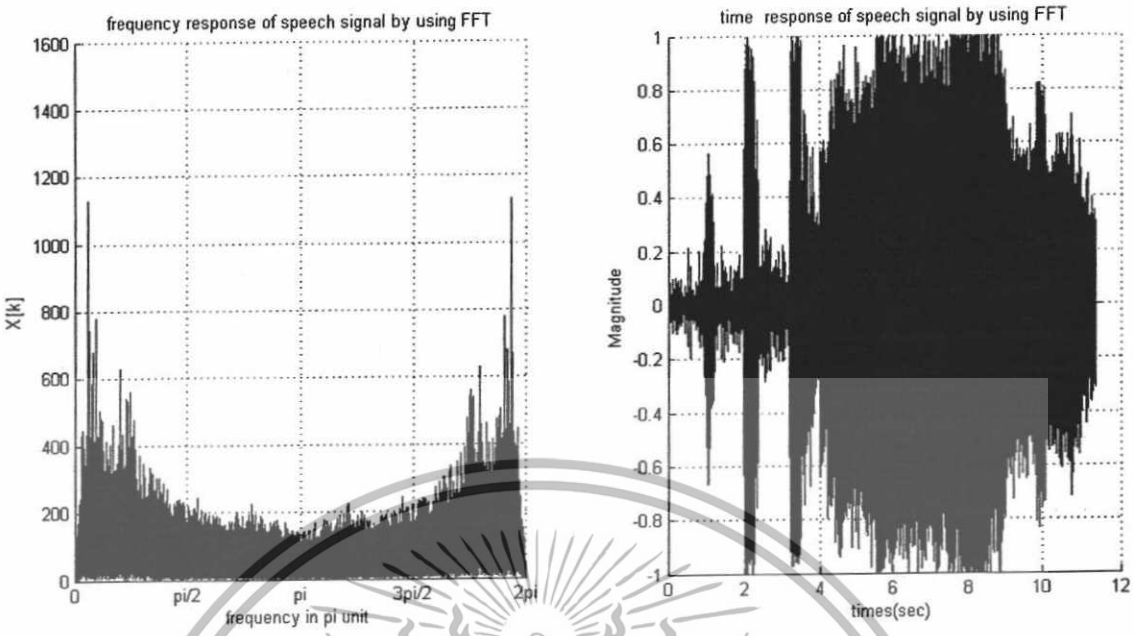


รูปที่ 4.7 เสียงสนทนา ก่อนเข้าสู่ระบบ



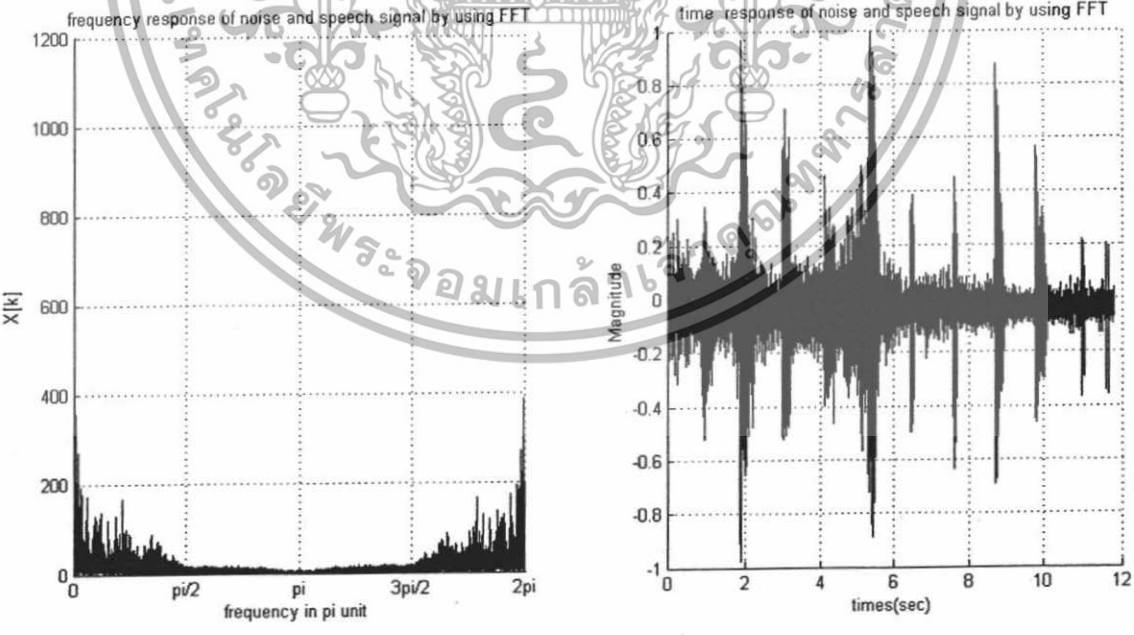
รูปที่ 4.8 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



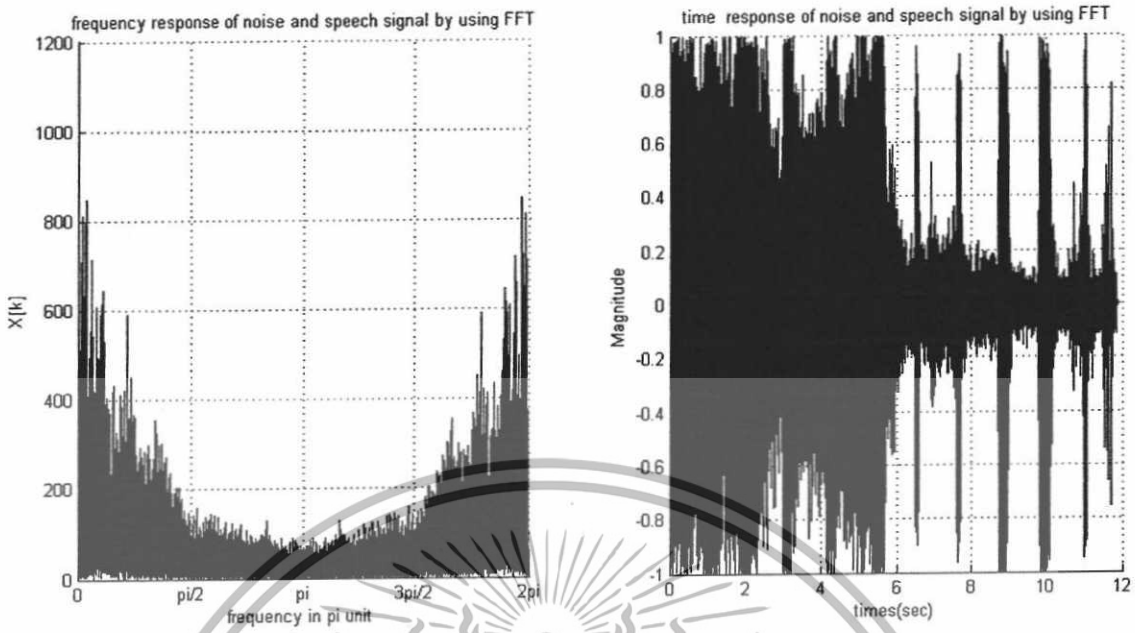
รูปที่ 4.9 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว

ตอนที่ 2(ต่อ) ศึกษาผลตอบสนองเชิงความถี่และผลตอบสนองเชิงเวลาของเสียงสนทนาที่มีเสียงรบกวนจากรถมอเตอร์ไซด์ เมื่อความถี่ออสซิลเลเตอร์16.384 MHzรีจิสเตอร์ควบคุม N1:M1 คือ2:5

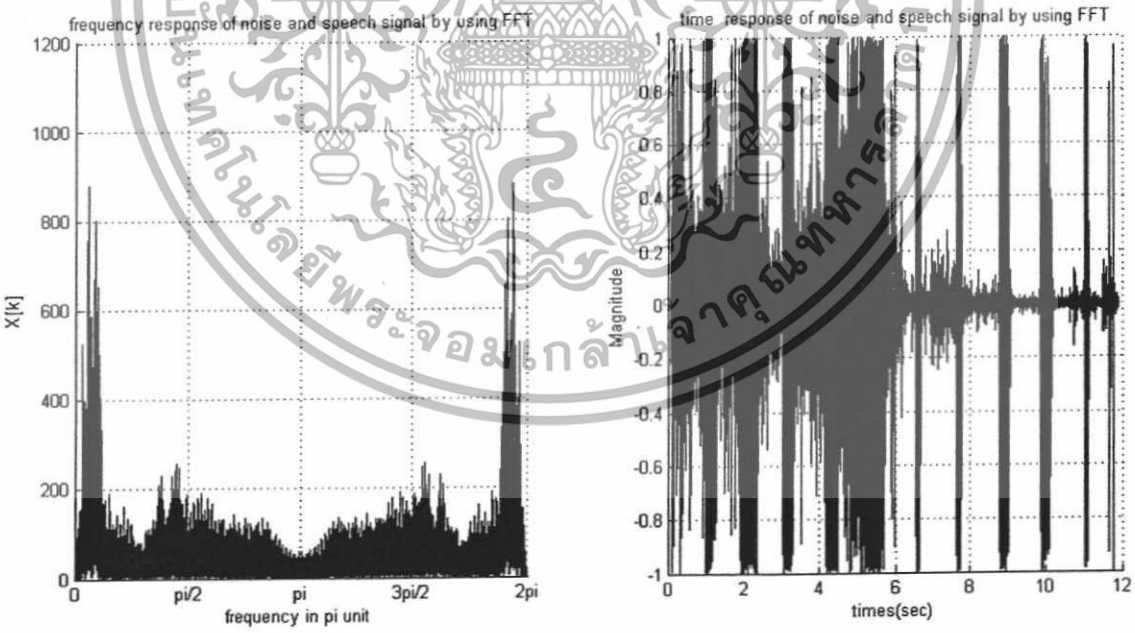


รูปที่ 4.10 เสียงสนทนา ก่อนเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 เสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน



รูปที่ 4.12 เสียงสนทนาหลังเข้าสู่ระบบที่มีการกำจัดเสียงรบกวนแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 สรุปผลการทดลอง

1.ศึกษาผลการเปลี่ยนค่าเฟสล็อกลูประหว่างไอซีเข้าและถอดรหัสเสียง กับไมโครคอนโทรลเลอร์ โดยเลือกใช้ออสซิลเลเตอร์จากภายนอก และมีการกำหนดค่ารีจิสเตอร์ควบคุมจากการคำนวณ

เมื่อเปลี่ยนค่ารีจิสเตอร์ควบคุม N1:M1 จากเดิม 4:10 เป็น 2:5 พบว่าไอซี Si3000 จะใช้เวลาส่งเฟรมปฐมภูมิไปยังไมโครคอนโทรลเลอร์นานกว่า นั่นคือรีจิสเตอร์ควบคุม 2:5 จะใช้เวลาในการสร้างความถี่สุ่มตัวอย่างนานกว่า ซึ่งสอดคล้องกับการเขียนค่าเฟสล็อกลูปใน Si3000 ที่ค่าจริงจากการคำนวณจะต้องน้อยกว่าค่าเดิมหนึ่งค่าเวลาเขียนโปรแกรม ในกรณีนี้จากเดิมคำนวณได้ 2:5 เวลาเขียนโปรแกรมจริงจึงกลายเป็น 1:4 ซึ่งเป็นอัตราส่วนที่น้อยเกินไปไม่เหมาะที่จะนำมาเขียนโปรแกรม แต่ในทางกลับกันเมื่อ Si3000 สามารถสร้างความถี่สุ่มตัวอย่างได้สมบูรณ์แล้วจะสรุปผลตอบสนองของสัญญาณได้ดังนี้

ผลตอบสนองเชิงความถี่ของเสียงสนทนาที่ใช้รีจิสเตอร์ควบคุม 4:10 พบว่าขนาด $|H(j\omega)|$ จะมีขนาดสูงกว่า เสียงสนทนาที่ใช้รีจิสเตอร์ควบคุม 2:5 ดังแสดงในรูปที่ 4.2 กับ 4.8 , รูปที่ 4.3กับ 4.9 และรูปที่ 4.5 กับ 4.11 เป็นต้น

ผลตอบสนองเชิงเวลาของเสียงสนทนาที่ใช้รีจิสเตอร์ควบคุม 4:10 พบว่าสามารถลดทอนสัญญาณรบกวน และขยายขนาดของเสียงสนทนาได้ดีกว่าการใช้รีจิสเตอร์ควบคุม 2:5 ดังแสดงในรูปที่ 4.3 กับ 4.9 ,รูปที่ 4.5 กับ 4.11 และรูปที่ 4.6 กับ 4.12

2.ศึกษาเปรียบเทียบผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนา ระหว่างเสียงสนทนาก่อนเข้าสู่ระบบที่ถูกบันทึกโดยตรง และเสียงสนทนาที่ถูกบันทึกไว้หลังผ่านระบบแต่ยังไม่มีการกำจัดเสียงรบกวน

ผลตอบสนองเชิงความถี่ของเสียงสนทนา ที่ถูกบันทึกโดยตรงจะมีขนาดของโลบหลักสูง แต่ภายหลังเมื่อเสียงสนทนาเข้าสู่ระบบที่ยังไม่มีการกำจัดเสียงรบกวน โลบหลักจะมีขนาดลดลงแต่จะมีขนาดของโลบข้างที่กว้างขึ้น นั่นคือจะมีเสียงสัญญาณรบกวนเพิ่มขึ้น ดังแสดงในรูปที่ 4.1 กับ 4.2 , รูปที่ 4.4 กับ 4.5, และรูปที่ 4.7 กับ 4.8 เป็นต้น

ผลตอบสนองเชิงเวลาของเสียงสนทนาหลังผ่านระบบที่ยังไม่ได้กำจัดเสียงรบกวน พบว่าทั้งขนาดของเสียงสนทนาและเสียงรบกวนจะถูกขยาย แต่ในช่วงที่มีขนาดของสัญญาณเสียงรบกวนมากกว่า 1 เสียงที่ผ่านระบบนี้จะมีขนาดถี่ลดลง ดังแสดงในรูปที่ 4.1 กับ 4.2 และรูปที่ 4.7 กับ 4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.ศึกษาเปรียบเทียบผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของเสียงสนทนา ระหว่างเสียงสนทนาหลังเข้าสู่ระบบที่ถูกบันทึกโดยตรง และเสียงสนทนาที่ถู กบันทึกไว้หลังผ่านระบบที่มีการกำจัดเสียงรบกวนแล้ว

ผลตอบสนองเชิงความถี่ของเสียงสนทนาหลัง จากผ่านระบบ แต่ยังไม่มีการกำจัดเสียงรบกวนพบว่าโลบหลักจะมีขนาดต่ำลงไป และโลบจะเลื่อนไปทางขวามากขึ้น เกิดเป็นโลบรอง ที่กว้างขึ้น ดังแสดงในรูปที่ 4.5 กับ 4.6 , รูปที่ 4.8 กับ 4.9 และรูปที่ 4.11 กับ 4.12

ผลตอบสนองเชิงเวลาของเสียงสนทนาหลังจากผ่านระบบแต่ยังไม่มีการกำจัดเสียงรบกวน พบว่า ขนาดของเสียงสนทนาจะถูกขยายอย่างเห็นได้ชัด แต่ช่วงที่มีเสียงสัญญาณรบกวนระบบจะลดทอนเสียงนั้นให้มีแมกนิจูดที่ต่ำลง ดังรูปที่ 4.5 กับ 4.6 และรูปที่ 4.11 กับ 4.12 ซึ่งสามารถอธิบายได้จากกระบวนการเน้นเสียงนั้นคำที่ เกิดขึ้นหลังจากการตัดองค์ประกอบความถี่ต่ำ นั้นคือ ไม่ว่าเฟรมนั้นจะเป็นของเสียงรบกวน หรือเสียงสนทนาก็ตาม สัญญาณในเฟรมนั้นก็จะถูกขยายดังที่กล่าวไว้แล้วในบทที่ 3 ต่อมาในช่วงที่มีเสียงรบกวน ฟังก์ชัน VAD จะทำงานและเมื่อ VAD อัพเดทสัญญาณเสียงรบกวนจึงถูกกำจัดลง

จากการทดลองทำให้ทราบว่าการกำจัดสัญญาณเสียงรบกวนโดยใช้ค่าเฟสล็อกลูป 4:10 ให้ผลของการกำจัดเสียงรบกวนที่ดีที่สุด และเมื่อผู้จัดทำได้ทดลองนำเสียงสนทนาที่กำจัดเสียงรบกวนแล้วให้เป็นอินพุทกับระบบ ใหม่อีกครั้ง พบว่าเสียงที่ผ่านระบบครั้งนี้ จะมีเสียงสนทนาที่ชัดเจนขึ้นและเสียงสัญญาณรบกวนจะลดลงจากเดิม แต่ในทางตรงกันข้ามหากนำ เสียงสนทนาที่ผ่านระบบการกำจัดเสียงรบกวนของเสียงที่มีขนาดของเสียงรบกวนมากกว่าหนึ่ง ป้อนเป็นอินพุทให้กับระบบใหม่อีกครั้ง พบว่าเสียงรบกวนจะถูกขยายมากกว่าเดิม ดังนั้นแนวทางที่ดีที่สุดคือ การพิจารณาขนาดของสัญญาณรบกวนต้องมีขนาดบนโดเมนของเวลาที่น้อยกว่าหนึ่ง ให้เป็นสัญญาณอินพุทที่ป้อนเข้าสู่ระบบ เอ้าพุทที่ได้ควรนำไปป้อนกลับเป็นสัญญาณอินพุทให้กับระบบใหม่ เสียงสนทนาที่ได้ก็จะชัดเจนและปราศจากเสียงรบกวนมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง

ปริญญานิพนธ์ฉบับนี้นำเสนอ ระบบกำจัดเสียงรบกวนประกอบ ด้วยหน่วยย่อยสามส่วน คือ หน่วยประมวลผลกลาง หน่วยเข้ารหัสและถอดรหัสเสียง และหน่วยแสดงผล สำหรับหน่วยประมวลผลกลางนั้นได้ใช้ไมโครคอนโทรลเลอร์ dsPIC30F6014 ขนาด 16 บิต ในการประมวลผลสัญญาณเสียงแทนเครื่องคอมพิวเตอร์ ซึ่ง dsPIC30F6014 นั้นมีความสามารถรองรับการประมวลผลสัญญาณเสียงเป็นอย่างดี และมีโมดูลพิเศษต่างๆที่ช่วยในการทำงานได้ดีกว่าไมโครคอนโทรลเลอร์ชนิดอื่นๆ เมื่อเชื่อมโยงหน่วยย่อยทั้งสามเข้ากัน ก็จะกลายเป็นระบบกำจัดเสียงรบกวน ซึ่งได้อธิบายรายละเอียดแล้วในบทที่ 3 เมื่อทำการทดลองกับสัญญาณเสียงรบกวนแล้ว ผู้จัดทำได้พล็อตผลตอบสนองเชิงความถี่ และผลตอบสนองเชิงเวลาของสัญญาณเสียงสนทนาออกมา และเปรียบเทียบการเปลี่ยนค่ารีจิสเตอร์ควบคุมในการสร้างความถี่สุ่มตัวอย่างพบว่า เสียงสนทนาที่มีเสียงสัญญาณรบกวนก่อนเข้าสู่ระบบ และเสียงสนทนาหลังเข้าสู่ระบบที่ยังไม่กำจัดสัญญาณเสียงรบกวน สัญญาณเสียงนั้นจะถูกขยายทั้งเสียงที่ต้องการและเสียงที่ไม่ต้องการ จนกระทั่งเมื่อเปิดระบบกำจัดสัญญาณเสียงรบกวนแล้ว ฟังก์ชัน VAD จะถูกอัพเดทและจะเปรียบเทียบอัตราส่วนของระดับพลังงานในเสียงสนทนากับเสียงสัญญาณรบกวน เพื่อตัดสินใจว่าเฟรมข้อมูลนั้นเป็นเฟรมของเสียงสนทนาหรือเสียงสัญญาณรบกวน จากนั้นสัญญาณที่ถูกอัพเดทจะนำมาคูณกับสเกลแฟกเตอร์ และแปลงฟูเรียกลับ กลายเป็นสัญญาณเสียงที่ปราศจากเสียงรบกวนออกมา และเมื่อผู้จัดทำทดลองเปลี่ยนค่ารีจิสเตอร์ควบคุม N1:M1 ให้มีค่าที่น้อยลงจะส่งผลให้ระบบอันใหม่นี้สร้างความถี่สุ่มตัวอย่างได้ช้า และประสิทธิภาพในการกำจัดสัญญาณเสียงรบกวนลดลง ซึ่งสามารถสรุปได้ว่าค่ารีจิสเตอร์ควบคุม จะต้องเป็นจำนวนเต็มที่เป็นอัตราส่วนกัน แต่ในที่นี้จะต้องคำนึงถึงความจริงในการเขียนโปรแกรม ที่นับค่าเริ่มต้นตั้งแต่ 0 จนถึง n ดังนั้นค่าที่คำนวณได้จึงต้องลบออก 1 เพื่อชดเชยการนับค่าเริ่มต้นของโปรแกรม ดังนั้นจึงไม่ควรออกแบบค่ารีจิสเตอร์ควบคุมเป็นจำนวนที่มีค่าน้อยเกินไป อันจะทำให้การสร้างความถี่สุ่มตัวอย่างคลาดเคลื่อน และจะส่งผลให้ขาดประสิทธิภาพในการกำจัดสัญญาณเสียงรบกวน ในที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาที่พบและแนวทางแก้ไข

1. เสียงสนทนาที่ผ่านระบบกำจัดเสียงรบกวนแล้วยังคงมีเสียงรบกวนแฝงอยู่ และเมื่อผู้ทดลองได้เปลี่ยนค่าค่ารีจิสเตอร์ควบคุม N1:M1 พบว่าสัญญาณเสียงรบกวนนั้นจะลดทอนลงมา จึงต้องหาค่าค่ารีจิสเตอร์ควบคุมนี้ที่เหมาะสมต่อไป
2. การพัฒนาโปรแกรมแบบฝังตัว ต่างจากการเขียนโปรแกรมบนเครื่องคอมพิวเตอร์มาก เพราะต้องคำนึงถึงขีดจำกัดของการเข้าถึงในหน่วยความจำ ด้วยเหตุนี้ ผู้จัดทำจึงต้องจองพื้นที่หน่วยความจำทุกครั้งก่อนการเขียนโปรแกรม

5.3 ข้อเสนอแนะและแนวทางในการค้นคว้าพัฒนา

เพื่อประสิทธิภาพในการกำจัดสัญญาณเสียงรบกวนที่มากขึ้นอาจมีการพัฒนาโปรแกรมให้มีการสุ่มตัวอย่างในความถี่ที่สูงขึ้น และเพื่อลดปัญหาในส่วนของขนาดหน่วยความจำภายในอาจมีการต่อหน่วยความจำไว้ภายนอกเพิ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] ธีรพงษ์ เจนวิริยะกุล, นรินทร์ทิพย์ เชาว์ทอง, ชญาดา ชั้นบุญใส, "การเข้ารหัสเสียงโดยวิธี เอ ดี พี ซี เอ็ม", ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2547.

[2] พรเทพ ศรีแสงอยู่, วีระศักดิ์ การุณย์, สุเวศ นวนวัน, "วงจรกรองความถี่เชิงเลข", ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2546.

[3] รวินทร์ วินทร์ วิรัชพินทุ, สุวารี เหลือเพิ่มพร, "ระบบสั่งงานด้วยเสียง", ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2546.

[4] ธนพล ตั้งชูพงศ์, ธนสิทธิ์ ชวลิตธารา, "การเข้ารหัสสัญญาณเสียงแบบ Real Time สำหรับมาตรฐาน ITU-T G.729 บน DSP chip TMS 320C6713" ,ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ , สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2547.

[5] ณัฐพล สมบัติภิญโญ, ณัฐยา อินทร์แก้ว, นัฐวัชร มหาวงศ์, "การส่งสัญญาณเสียงโดยใช้แสง" , ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2547.

[6] นคร ภักดีชาติ, ชัยวัฒน์ ลิ้มพรจิตรวิไล, **คู่มือการทดลอง dsPIC Microcontroller เบื้องต้นด้วยโปรแกรมภาษา C กับ MPLAB C30,** กรุงเทพนคร : อินโนเวตีฟ เอ็กเพอริเมนต์.

[7] ธนารักษ์ ธีระมั่นคง, **เทคโนโลยีสมองกลฝังตัว,** พิมพ์ครั้งที่ 1. กรุงเทพนคร : สมาคมส่งเสริมเทคโนโลยี(ไทย-ญี่ปุ่น), 2549.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

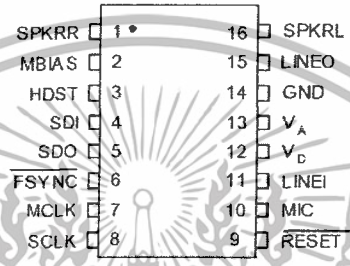
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ไอซีเข้าและถอดรหัสเสียง Si3000

ไอซีเข้าและถอดรหัสเสียง Si3000 เป็นอุปกรณ์แปลงสัญญาณดิจิตอลเป็นแอนะล็อก และแอนะลอกเป็นดิจิตอล 16 บิต มีรายละเอียดต่างๆดังนี้

1. Pin Descriptions: Si3000



Pin #	Pin Name	Description
1	SPKRR	**Speaker Right Output.** Analog output capable of driving a 60 Ω load.
2	MBIAS	**Microphone bias output.**
3	HDST	**Handset Input/Output.** Handset analog input/output.
4	SDI	**Serial Port Data In.** Serial communication and control data that is generated by the Si3000 to the system DSP.
5	SDO	**Serial Port Data Out.** Serial communication data that is provided by the Si3000 to the system DSP.
6	$\overline{\text{FSYNC}}$	**Frame Sync Output.** Data framing signal that is used to indicate the start and stop of a communication data frame.
7	MCLK	**Master Clock Input.** High speed master clock input. Generally supplied by the system crystal clock or DSP.
8	SCLK	**Serial Port Bit Clock Input/Output.** Controls the serial data on SDO and latches the data on SDI. This pin is an input in slave mode and an output in master mode.
9	$\overline{\text{RESET}}$	**Reset.** An active low input that is used to reset all control registers to a defined initialized state. Also used to bring the Si3000 out of sleep mode.
10	MIC	**MIC Input.** Microphone level or line level input. This input contains selectable gain of 0, 10, 20, or 30 dB with a full scale input level of 1 V_{RMS}.
11	LINEI	**Line Input.** Line level input with selectable gain of 0, 10, or 20 dB. The full scale input level is 1 V_{RMS}.
12	V_D	**Digital Supply Voltage.** Provides the digital supply voltage to the Si3000. Nominally either 5 or 3.3 V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin #	Pin Name	Description
14	GND	**Ground.** Connects to the system digital ground.
15	LINEO	**Line Output.** Line level analog output with a 1 V_{RMS} full scale output level.
16	SPKRL	**Speaker Left Output.** Analog output capable of driving a 60 Ω load.

2. Control Registers

Note: Any register not listed here is reserved and should not be written. Any register bit labelled reserved should be written to zero during writes to the register. Register 0 can be read (always returns 0) and written safely.

Table 1. Register Summary

Register	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Control 1	SR			SPD	LPD	HPD	MPD	CPD
2	Control 2				HPFD	PLL	DL1	DL2	
3	PLL1 Divide N1	Divider N1							
4	PLL1 Multiply M1	Multiplier M1							
5	RX Gain Control 1	LIG		LIM	MCG		MCM	HIM	IIR
6	ADC Volume Control	RXG						LOM	HOM
7	DAC Volume Control	TXG						SLM	SRM
8	Status Report	SLSC	SRSC	LOSC					
9	Analog Attenuation					LOT		SOT	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 1. Control 1

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	SR			SPD	LPD	HPD	MPD	CPD
Type	R/W			R/W	R/W	R/W	R/W	R/W

Reset settings = 0000_0000

Bit	Name	Function
7	SR	**Software Reset**. 1 = Sets all registers to their reset value. 0 = Enables chip for normal operation. **Note:** Bit will automatically clear after being set.
6:5	Reserved	Read returns zero.
4	SPD	**Speaker Drive Power Down.** 1= Normal operation 0 = Power down left and right speaker drive.
3	LPD	**Line Drive Power Down.** 1 = Normal operation 0 = Power down line driver.
2	HPD	**Handset Drive Power Down.** 1 = Normal operation 0 = Power down handset driver.
1	MPD	**MIC Bias Power Down.** 1 = Power down MIC bias buffer. 0 = Normal operation
0	CPD	**Chip Power Down**. 1 = Puts Si3000 into power down mode. 0 = Normal operation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 2. Control 2

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name				HPFD	PLL	DL1	DL2	
Type				R/W	R/W	R/W	R/W	

Reset Settings = 0000_0000

Bit	Name	Function
7:5	Reserved	Read returns zero.
4	HPFD	**High Pass Filter (HPF) Disable.** 1 = HPF disabled 0 = HPF enabled
3	PLL	**PLL Divide by 10.** 1 = Sets final stage of PLL to divide by 10. 0 = Sets final stage of PLL to divide by 5.
2	DL1	**Digital Loopback.** 1 = Enables digital loopback (DAC analog out → ADC analog in). 0 = Normal operation
1	DL2	**Digital Loopback.** 1 = Enables digital loopback (DAC one bit → ADC one bit). 0 = Normal operation
0	Reserved	Read returns zero.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 3. PLL1 Divide N1

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	Divider N1							
Type	R/W							

Reset settings = 0000_0000

Bit	Name	Function
7:0	N1	N1. Contains the (value – 1) for determining the output frequency on PLL.

Register 4. PLL1 Multiply M1

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	Multiplier M1							
Type	R/W							

Reset settings = 0000_0000

Bit	Name	Function
7:0	M1	M1. Contains the (value – 1) for determining the output frequency on PLL.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 5. RX Gain Control 1

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	LIG		LIM	MCG		MCM	HIM	IIR
Type	R/W		R/W	R/W		R/W	R/W	R/W

Reset settings = 0100_0111

Bit	Name	Function
7:6	LIG	Line in Gain. 11 = 20 dB gain 10 = 10 dB gain 01 = 0 dB gain 00 = Reserved
5	LIM	Line in Mute. 1 = Line input muted 0 = Line input goes to mixer
4:3	MCG	MIC Input Gain. 11 = 30 dB gain 10 = 20 dB gain 01 = 10 dB gain 00 = 0 dB gain
2	MCM	MIC Input Mute. 1 = Mute MIC input 0 = MIC input goes into mixer.
1	HIM	Handset Input Mute. 1 = Mute handset input 0 = Handset input goes into mixer.
0	IIR	IIR Enable. 1 = Enables IIR filter 0 = Enables FIR filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 6. ADC Volume Control

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name		RXG					LOM	HOM
Type	R/W						R/W	R/W

Reset settings = 0101_1100

Bit	Name	Function
7	Reserved	Read returns zero.
6:2	RXG	RX PGA Gain Control. 11111 = 12 dB 10111 = 0 dB 00000 = –34.5 dB LSB = 1.5 dB
1	LOM	Line Out Mute. 0 = Mute 1 = Active
0	HOM	Handset Out Mute. 0 = Mute 1 = Active

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 7. DAC Volume Control

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name		TXG					SLM	SRM
Type	R/W						R/W	R/W

Reset settings = 0101_1100

Bit	Name	Function
7	Reserved	Read returns zero.
6:2	TXG	TX PGA Gain Control. 11111 = 12 dB 10111 = 0 dB 00000 = –34.5 dB LSB = 1.5 dB
1	SLM	SPKR_L Mute. 0 = Mute 1 = Active
0	SRM	SPKR_R Mute. 0 = Mute 1 = Active

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 8. Status Report

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	SLSC	SRSC	LOSC					
Type	R	R	R					

Reset settings = 0000_0000

Bit	Name	Function
7	SLSC	SPK_L Short Circuit. 1 = Indicate short circuit status is detected at left speaker. 0 = Normal mode
6	SRSC	SPK_R Short Circuit. 1 = Indicate short circuit status is detected at right speaker. 0 = Normal mode
5	LOSC	Line Out Short Circuit. 1 = Indicate short circuit status is detected at line out. 0 = Normal mode
4:0	Reserved	Read returns zero.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register 9. Analog Attenuation

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name					LOT		SOT	
Type					R/W		R/W	

Reset settings = 0000_0000

Bit	Name	Type
7:4	Reserved	Read returns zero.
3:2	LOT	Line Out Attenuation. 11 = –18 dB analog attenuation on Line Output. 10 = –12 dB analog attenuation on Line Output. 01 = –6 dB analog attenuation on Line Output. 00 = 0 dB analog attenuation on Line Output.
2:0	SOT	Speaker Out Attenuation. 11 = –18 dB analog attenuation on Speaker Output. 10 = –12 dB analog attenuation on Speaker Output. 01 = –6 dB analog attenuation on Speaker Output. 00 = 0 dB analog attenuation on Speaker Output.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้