



เครื่องกำจัดเสียงก้อง

ACOUSTIC ECHO CANCELLER



ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่ใช้การนำไปใช้

040414

เครื่องกำจัดเสียงก้อง
ACOUSTIC ECHO CANCELLER



โดย
นายจเด็จ พลจรัส 39013044
นายณัฐศักดิ์ เกิดศรีเหล็ก 39013048
อาจารย์ที่ปรึกษา
รศ. ณรงค์ เหมกรณ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2541

ภาควิชาวิศวกรรมโทรคมนาคม

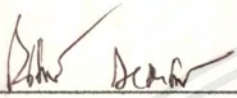
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องกำจัดเสียงก้อง

ACOUSTIC ECHO CANCELLER

ผู้จัดทำ

1. นายเจด็จ พลจรัส 39013044
2. นายณัฐศักดิ์ เกิดศรีเหล็ก 39013048



อาจารย์ที่ปรึกษา

(รศ.ณรงค์ เหมกর্ণ)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องกำจัดเสียงก้อง

ACOUSTIC ECHO CANCELLER

โดย นายจเด็จ พลจรัส 39013044

นายณัฐศักดิ์ เกิดศรีเหล็ก 39013048

อาจารย์ที่ปรึกษา รศ.ณรงค์ เหมกรณ์

บทคัดย่อ

ในระบบการสื่อสาร โทรคมนาคมแบบสองทิศทาง(Full Duplex) ความชัดเจนของเสียงพูดที่ใช้ในการสื่อสารกัน เป็นสิ่งสำคัญอย่างมาก เพราะมันหมายถึงว่าการสื่อสารนั้นมีประสิทธิภาพดีหรือไม่เพียงพอ ถ้าเกิดมีการสะท้อนกลับของเสียงที่ส่งมาซึ่งด้านรับกลับไปด้านส่งอีกครั้งก็จะเกิดเป็นเสียงก้องหรือเสียงสะท้อนขึ้นทำให้การรับฟังไม่ชัดเจนและเกิดการผิดพลาดในการสื่อสารได้ ดังนั้น เครื่องกำจัดเสียงก้องหรือเสียงสะท้อน จึงมีความจำเป็นในการสื่อสารแบบสองทิศทาง เนื้อหาปริญญาบัตรฉบับนี้นำเสนอ ทฤษฎีการออกแบบ การสร้าง และการใช้งานเครื่องกำจัดเสียงก้อง ที่ทำงานในย่านความถี่เสียงด้วยการวิเคราะห์โดยใช้วิธีประมวลผลสัญญาณทางดิจิทัล จากโปรเซสเซอร์ประมวลผลทางดิจิทัล เพื่อกำจัดเสียงก้องหรือเสียงสะท้อน

ABSTRACT

Voice clearness in Duplex voice communication system is important thing extremely. That show how does the system have a good efficiency?If it has echo from a receiver back to a transmitter, echo error of communication will occur. Therefore,an acoustic echo canceller is useful for the Duplex voice communication system. This project is consist of Design Theory, Creating and Using acoustic echo canceller in frequency voice band. The Digital Signal Processing is used for getting rid of the echo.

สารบัญ

		หน้า
บทที่ 1	บทนำ	1
บทที่ 2	ทฤษฎีและหลักการ	2
	2.1 Acoustic echo canceller	2
	2.2 คอมแพนเคอร์	11
	2.3 LMS Algorithm	16
	2.4 NORMALIZED LMS ALGORITHM	19
	2.5 การวิเคราะห์ภายในช่วงเวลาสั้นๆ (Short-Time)	24
	2.6 ความมีเสถียรภาพ,การลู่เข้า และสถานะคงที่ของค่าเฉลี่ยกำลังสอง	26
	2.7 วงจรสุ่มตัวอย่างและการคงค่าสัญญาณ (Sample And Hold)	29
	2.8 TMS320C50	36
	2.9 ส่วนแปลงสัญญาณอนาลอกเป็นดิจิทัล	56
	2.10 หลักการทั่วไปของ Acoustic Echo Canceler ที่ใช้ อัลกอริทึม FBAF	58
	2.11 การกรองความถี่ที่มีลำดับของสัญญาณยาวๆ	63
บทที่ 3	การคำนวณและการสร้าง	69
	3.1 TMS 320C50 DSP Starter Kit	69
	3.2 การคำนวณหาค่าในส่วนของวงจรแสดงผลที่ใช้ IC 4066	83
	3.3 การออกแบบวงจรกรองความถี่ต่ำ	85
	3.4 การออกแบบวงจรหน่วงเวลา(delay Circuit)	89
บทที่ 4	การทดลองและผลการทดลอง	97
	4.1 การทดลองวงจรกรองความถี่ต่ำ	97
	4.2 ผลการทดลองวงจร Digital to Analog Conversion (DAC)	99
	4.3 ผลการทดลองวงจร Analog to Digital Conversion (ADC)	102
	4.4 ผลการทดลองวงจร COMPANDER	104
	4.5 ผลการทดลองวงจรการกำเนิดสัญญาณสี่เหลี่ยมที่ใช้ IC 4066 ของวงจรแสดงผล	106
	4.6 การเขียน โปรแกรมคำสั่งภาษาแอสเซมบลีของ DSP	108
	4.7 ผลการทดลองเครื่องกำจัดเสียงก้อง(Acoustic Echo Canceller)	110
	4.7.1 ผลการทดลองสุ่มรับส่งข้อมูล โดยที่ไม่ต้องผ่านชุดอัลกอริทึม	111
	4.7.2 ผลการทดลองการกำจัดเสียงสะท้อนระหว่างไมโครโฟนกับลำโพง	114
	4.7.3 ผลการทดลองโดยใช้ชุด Delay Equalizer	118
	4.7.4 ผลการทดลองโดยใช้ชุด Delay Equalizer ที่ใช้อินพุทเป็น Sinewave	122

บทที่ 5 บทวิจารณ์ และ บทสรุป

125

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

ภาคผนวก

กิตติกรรมประกาศ

หนังสืออ้างอิง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1.1	อธิบายถึงการสื่อสารทางเคียวที่ใช้ระบบ Hand-free ออดิโอเทอร์มินอล	4
2.1.2	โครงสร้างของเส้นทางเดินเสียงของมนุษย์	6
2.1.3	แสดงภาพจำลองโครงสร้างของเส้นทางเดินเสียงของมนุษย์	6
2.1.4	แสดงสเปคโตรแกรมของสัญญาณเสียง ในการสื่อสารทางค่านดิจิตอล	7
2.1.5	แสดงให้เห็นถึงค่าขนาดของสเปคตรัมของสัญญาณเสียง	7
2.1.6	Impulse response : teleconference room	8
2.1.7	Impulse response : hands-free telephone	8
2.1.8	AVS และ AVR	9
2.2.1	ลักษณะของระดับเสียง	12
2.2.2	การจัดวางตำแหน่งของการคอมแพนเคอร์	13
2.2.3	โครงสร้างภายในของ NE 570/571	14
2.2.4	รูปฟอร์มของการกระจัดสัญญาณเสียง	15
2.2.5	แสดงการต่อวงจรเพื่อขยายกว้างออกของสัญญาณ	16
2.3.1	บล็อกไดอะแกรมของ Widrow's LMS Algorithm สำหรับ Transversal Filter	18
2.4.1	ตัวอย่างของค่าการเปลี่ยนแปลงมาตรฐาน โดยใช้สมการ (2.7.1)	23
2.4.2	ค่า variance ที่คำนวณได้จากสมการ (2.7.3), $x(n)$ และ $\sigma(n)$ ที่ $\alpha = 0.9$, $x(n) G(n)$	23
2.5.1	แสดงการวัดค่าต่างๆ ในช่วงเวลาสั้นๆ	25
2.7.1	แสดงพื้นฐานของการ Sample-Hold	29
2.7.2	แสดงวงจร Sample And Hold ที่ใกล้เคียงกับวงจรใช้งานจริง	30
2.7.3	ขนาดของ ฟังก์ชันการตอบสนองความถี่ของอุปกรณ์การ โสที่ค่า $T=0.2$	31
2.7.4	แสดงวงจรพื้นฐานแบบ Open-Loop	32
2.7.5	Current Source Sample-Hold	32
2.7.6	แสดง Low-Frequency Sample-Hold with Feedback	33
2.7.7	Low-Frequency Sample-Hold with Integrator Feedback	33
2.7.8	แสดงช่วงเวลาที่สำคัญของวงจร Sample And Hold	34
2.7.9	(a) สเปคตรัมสัญญาณที่มีความต่อเนื่องทางด้านเวลา, (b) สเปคตรัมสัญญาณที่เป็นดิสครีต	35
2.8.1	บล็อกไดอะแกรมของ TMS320C50	37
2.8.2	แสดงการจัดตำแหน่งของหน่วยความจำของ TMS320C50	44
2.8.3	แสดงการอินเทอร์รัพต์เฟลท	51
2.8.4	แสดงการต่อพอร์คอนุกรมกับพอร์คภายนอก	51
2.8.5	แสดงบล็อกไดอะแกรมการทำงานของพอร์คอนุกรม	52

สารบัญรูป(ต่อ)

รูปที่	หน้า
2.8.6 แสดงบล็อกโคอะแกรมของไทม์เมอร์	53
2.10.1 การแบ่งผลการตอบสนองทางอิมพัลส์	59
2.10.2 อัลกอริทึม แบบ FBAF	62
2.11.1 กระบวนการของ Fast Convolution	64
2.11.2 แสดง Overlap- Save Method	65
2.11.3 แสดง Overlap-Add method	67
2.11.4 แสดงผลลัพธ์จากสมการ (2.11.4)	68
3.1.1 บล็อกโคอะแกรมของ TMS320C5X DSK	69
3.1.2 แสดงหน่วยความจำภายใน TMS320C50 DSK	70
3.1.3 แสดงการเชื่อมต่อระหว่าง DSK และ PC ผ่านทาง RS232	70
3.1.4 แสดงขั้นตอนการสร้างโปรแกรม	71
3.1.5 แสดงการกำหนดค่าใน PMST	72
3.1.6 แสดงฟังก์ชันบล็อกโคอะแกรม ของ TLC 32040	75
3.1.7 แสดงการเชื่อมต่อระหว่าง TLC32040 กับ TMS320C50	81
3.1.8 แสดงช่วงเวลาในการส่งและรับข้อมูลระหว่าง TLC32040 กับ TMS320C50	82
3.2.1 วงจรกำเนิดสัญญาณรูปสี่เหลี่ยม โดยใช้ IC#4066	83
3.2.2 วงจรสมมูลย์	84
3.3.1 วงจรกรองความถี่ต่ำแบบ Active	85
3.3.2 วงจรกรองความถี่ต่ำแบบ BUTTERWORTH อันดับที่ 4 มีความถี่คัทออฟที่ 3.4kHz	88
3.3.3 วงจรกรองความถี่ต่ำแบบ BUTTERWORTH อันดับที่ 2 มีความถี่คัทออฟที่ 3.4 kHz	88
3.4.1 แสดงระบบการหน่วงเวลาที่ใช้ในวงจรอีควอไลเซชัน(delay equalization)	89
3.4.2 แสดงการป้อนสัญญาณสามเหลี่ยมที่มีสัญญาณรบกวนความถี่สูงรวมอยู่กับรูปสัญญาณ	89
3.4.3 แสดงผลการตอบสนองต่อการหน่วงเวลา	89
3.4.4 แสดงลักษณะแพทเทิร์นของโพลและซีโร ของวงจรที่ให้สัญญาณผ่าน ไม้คัตลอค	90
3.4.5 แสดงค่าของการหน่วงเวลาที่อยู่ในฟังก์ชันความถี่	91
3.4.6 วงจรออปแอมป์	91
3.4.7 แสดงค่าการหน่วงเวลารวมของวงจรที่ใช้วงจรหน่วงเวลาลำดับที่ 1 ต่ออนุกรมกับ วงจรกรองความถี่ต่ำแบบบัตเตอร์เวิร์คลำดับที่ n	92
3.4.8 แสดงค่าการหน่วงเวลามาพล็อตกราฟเทียบกับค่า Q	94
3.4.9 แสดง โพลและซีโรใน s-plane	94
3.4.10 วงจรของ Deliyannnis	95

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.7.3.5 แสดงการทดลองตามรูปที่ 4.7.3.1(b) โคชวัดสัญญาณที่ Input2(Ch1) กับ Output(Ch2) ในช่วงที่ไม่มีเสียงพูดจากไมโครโฟน	120
4.7.3.6 แสดงการทดลองตามรูปที่ 4.7.3.1(b) โคชวัดสัญญาณที่ Input2(Ch1) กับ Output(Ch2) ในช่วงที่เริ่มต้นการพูดจากไมโครโฟน	121
4.7.3.7 แสดงการทดลองตามรูปที่ 4.7.3.1(b) โคชวัดสัญญาณที่ Input2(Ch1) กับ Output(Ch2) ในช่วงที่มีการพูดจากไมโครโฟนอย่างต่อเนื่อง	121
4.7.4.1 แสดงการต่อวงจรในการทดลองชุด Delay Equalizer ที่มีการป้อนอินพุตเป็น Sinewave	122
4.7.4.2 แสดงรูปสัญญาณที่ตำแหน่ง Input2 ซึ่งเป็นสัญญาณขาชนที่ป้อนเป็นสัญญาณอินพุต	123
4.7.4.3 แสดงรูปสัญญาณที่ตำแหน่ง Output1 ซึ่งเป็นสัญญาณเอาต์พุต	123
4.7.4.4 แสดงรูปสัญญาณที่ตำแหน่ง Output2(Ch1) กับ Input1(Ch2) ซึ่งเป็นสัญญาณ อินพุตและเอาต์พุต ของ ชุด Delay Equalizer	124
4.7.4.5 แสดงรูปสัญญาณที่ตำแหน่ง Output1(Ch1) กับ Input1(Ch2) ซึ่งสัญญาณ Output1(Ch1) เป็นสัญญาณ เอาต์พุต	124

สารบัญตาราง

ตารางที่	หน้า	
2.8.1	แสดงตำแหน่งขาและหน้าที่การทำงานของ TMS320C50	38
2.8.2	แสดงการเชื่อมต่อรีจิสเตอร์ควบคุมหน่วยความจำเพื่อใช้หน่วยความจำข้อมูล	45
2.8.3	แสดง ตำแหน่งแอดเดรสข้อมูล เทง 0	45
2.8.4	แสดงการกำหนดค่าสำหรับใช้หน่วยความจำโปรแกรม	48
2.8.5	แสดงตำแหน่งของอินเทอร์รัคต์เวกเตอร์	48
2.8.6	แสดงขาสัญญาของพอร์ตคอนนุกรม	51
2.8.7	แสดงรีจิสเตอร์พอร์ตคอนนุกรม	52
2.8.8	แสดงรีจิสเตอร์ควบคุมไทม์เมอร์ (TCR)	53
2.8.9	แสดงความสัมพันธ์ระหว่างรีจิสเตอร์ กับ พื้นที่ของการรอ	54
2.8.10	Wait-State Field Values and Wait States as a Function of CWSR Bit n	55
2.8.11	Space Controlled by CWSR Bit n	55
2.9.1	แสดงการทำงานของ ADC908 ในโหมด RAM	56
2.9.2	แสดงการทำงานของ ADC908 ในโหมด ROM	57
2.9.3	แสดงการทำงานของ ADC908 ในโหมด Slow-Memory	57
3.1.1	แสดงตำแหน่งขาและหน้าที่การทำงานของแต่ละขา	76
3.1.2	แสดงการกำหนดค่าใน D1 และ D0 ในการทำงานในโหมดต่างๆของ AIC	80
3.1.3	แสดงเงื่อนไขของการตอบสนอง AIC	82
4.1.1	แสดงผลการทดลองวงจรกรองความถี่ต่ำอันดับสี่ แบบ BUTTERWORTH	97
4.1.2	แสดงผลการทดลองวงจรกรองความถี่ต่ำอันดับสอง แบบ BUTTERWORTH	98
4.2.1	แสดงผลการทดลองวงจร DAC	100
4.3.1	แสดงผลการทดลองวงจร ADC	103
4.4.1	แสดงผลการทดลองวงจร COMPANDER	104

บทที่ 1

บทนำ

ในปัจจุบันการสื่อสารระบบสองทิศทาง(Full Duplex) ได้แก่ การประชุมทางไกล, ระบบโทรศัพท์ทางไกล หรือแม้กระทั่งในระบบ Room Acoustic คุณภาพเสียงและความชัดเจนของเสียง จะเป็นสิ่งสำคัญอย่างมาก เพราะจะเป็นตัวกำหนดประสิทธิภาพของการสื่อสารว่าดีหรือไม่เพียงใด ถ้าเกิดมีการสะท้อนกลับของเสียงที่ส่งมา จะทำให้เกิดเสียงก้องหรือเสียงสะท้อนขึ้นในระหว่างการสื่อสารกัน ทำให้คุณภาพเสียงแหล่งและการรับฟังเสียงที่ติดต่อกับสื่อสารก็จะไม่ชัดเจน ทำให้เกิดการผิดพลาดในการติดต่อสื่อสารกันขึ้น ดังนั้นอุปกรณ์ที่สามารถใช้ในการกำจัดเสียงก้องหรือเสียงสะท้อนที่เกิดขึ้นในระบบการสื่อสารจึงมีความจำเป็น เพื่อให้การรับฟังเกิดความชัดเจน ทำให้การสื่อสารมีประสิทธิภาพดีขึ้น ดังนั้นในการทดลองนี้จึงเสนอ เครื่องกำจัดเสียงก้อง ซึ่งใช้ในย่านความถี่เสียง ด้วยการวิเคราะห์โดยใช้วิธีการประมวลผลสัญญาณทางดิจิทัล จากโปรเซสเซอร์ประมวลผลสัญญาณดิจิทัล ก่อนที่จะมีการส่งสัญญาณเสียงนั้นไปยังระบบการส่งสัญญาณแบบต่างๆต่อไป



บทที่ 2

ทฤษฎีและหลักการ

2.1 Acoustic echo canceller

เมื่อพูดถึงการสื่อสารแบบมือถือ(Hand-free)การพูดและการฟังผ่านทางอุปกรณ์เทอร์มินอล โดยที่ไม่มีชุดแฮนด์เซตเป็นความต้องการที่เกิดขึ้นเมื่อหลายปีที่ผ่านมา จะเป็นปัญหาอย่างมาก โดยเฉพาะเรื่อง การควบคุมเสียงสะท้อนในการทำงานในโหมดของ hand-free ของอุปกรณ์เทอร์มินอลของระบบเสียง อาทิจำ ระบบ hand-free ในชุดโทรศัพท์, ระบบการประชุมทางไกล, การส่งสัญญาณภาพและเสียงทางไกล(video conferencing) เป็นต้น

การควบคุมการสะท้อนสามารถกระทำอย่างง่าย ๆ ได้โดยใช้การกดระดับสัญญาณเสียงสะท้อน (echo suppressors) ซึ่งจะเป็นการใช้การเปลี่ยนแปลงระดับการลดทอนของสัญญาณในช่วงของการรับและ ส่ง ซึ่งจะไม่สามารถทำให้เป็นการสื่อสารแบบพูด-ดูเพล็กซ์ได้และยัง ไม่สามารถทำงานได้ในสถานะที่มี สัญญาณเต็มไปด้วการรบกวน เพื่อเป็นการแก้ปัญหาจึงได้มีการใช้ตัวกำจัดเสียงสะท้อนที่สามารถ ปรับเปลี่ยนค่าได้เอง(Adaptive echo canceller) โดยลักษณะการทำงานจะเป็นการสร้างเลียนแบบการตอบสนองต่ออิมพัลส์เรสพอร์น ในทางเดินของเสียงสะท้อนจริงโดยการใช้การฟิลเตอร์ที่ปรับค่าได้เอง ซึ่งจะ คล้ายคลึงกับเสียงสะท้อนที่เป็นกระแสไฟฟ้าที่เกิดขึ้นจากการไม่แมทช์กันของตัวแปลงสัญญาณจาก 2 wire-4 wire (hybrid) ซึ่งจะต้องคิดคั้งตัวกำจัดเสียงสะท้อนไว้ที่ตัวรวมสาย เพื่อที่จะให้ทำความเข้าใจใน เรื่องของเสียงสะท้อน จะเริ่มต้นการเกิดเสียงสะท้อนจากเสียงที่เดินทางออกไปจากอุปกรณ์เทอร์มินอลซึ่งเป็นลำโพงผ่านทางเดินของเสียงที่เกิดขึ้นจากลำโพง ไปยังลำโพง โดยเสียงที่เดินทางไปในอากาศจะมีทางเดินของเสียงมากมาย ทำให้การเดินทางของเสียงที่กลับเข้ามายังไม่โคร โฟนมีเวลาไม่เท่ากันอันเนื่องมาจากระยะทางที่ใช้ไม่เท่ากัน ทำให้เกิดการรบกวนกับสัญญาณเสียงปกติเมื่อเป็นการส่งสัญญาณที่มีความล่าช้าของระบบมากกว่า 20-30 มิลลิวินาที เพื่อให้เกิดความมิเสถียรภาพของรูปการส่งสัญญาณจะต้องมีการ ป้องกันการใช้ความถี่สูงซึ่งจะมีประโยชน์ในการประยุกต์ใช้ในบางประเภท

อีกทั้งเมื่อมีการใช้ไมโคร โฟนและลำโพงที่มีคุณภาพสูงจะทำให้ระบบเกือบมีความเป็นเชิงเส้น มากยิ่งขึ้นโดยพิจารณาจากการตอบสนองต่ออิมพัลส์คือเส้นทางเดินของเสียง นั่นคือการตอบสนองต่ออิมพัลส์จะแสดงถึงจำนวนของค่าสัมประสิทธิ์ ซึ่งอาจจะมีค่าได้ในช่วงหลายร้อยไปจนถึงหลายพัน ซึ่งจะขึ้น กับงานแต่ละประเภท ยิ่งไปกว่านั้นค่าการตอบสนองต่ออิมพัลส์สามารถที่จะเปลี่ยนแปลงได้ตลอดเวลา อันเป็นผลมาจากการเปลี่ยนแปลงคุณสมบัติทางด้านอะคูสติคของห้อง เช่นการเปิดหน้าต่าง, การเคลื่อนที่ ของสิ่งต่างๆที่เกิดขึ้นภายในห้อง เป็นต้น โมเดลของของเส้นทางเดินของเสียงจะเกิดขึ้นมาจากปรากฏ การณ์ทางด้านฟิสิกส์ที่มีอยู่ภายในห้องซึ่งจะเป็นส่วนหนึ่งในการนำเอามาช่วยแก้ปัญหา แต่โดยทั่วไป แล้วจะนิยมนำเอาการตอบสนองต่ออิมพัลส์มาใช้เป็นพื้นฐาน ในการกำจัดเสียงสะท้อน(Acoustic Echo Cancellation)

ปัญหาที่เหมือนกันของระบบการกำจัดเสียงสะท้อน(Acoustic Echo Canceller)ก็คือ ความไม่คงที่ ของการตอบสนองต่ออิมพัลส์และความแรงของสัญญาณรบกวนที่เอาท์พุทจะมีค่ามาก การแก้ไขโดยมาก

จะใช้ทฤษฎีของกระบวนการฟิลเตอร์แบบปรับค่าเอง ซึ่งโดยมากจะใช้กระบวนการการประมวลผลทางตัวเลขเข้ามาประยุกต์ใช้

การประยุกต์ใช้งานของระบบควบคุมสัญญาณสะท้อน

ในการสื่อสารที่ใช้เสียงเป็นสัญญาณข้อมูลนั้นเราสามารถแบ่งออกได้ 2 กลุ่มคือ กลุ่มออดิโอเทอร์มินอล และกลุ่มที่ใช้ระบบ Hand-free ในโทรศัพท์

- กลุ่มออดิโอเทอร์มินอล จะรวมทั้งระบบการประชุมทางไกล(teleconference) และระบบออดิโอคอนเฟอร์เรนซ์

ในระบบการประชุมทางไกลจะมีการคิดคั้งอุปกรณ์ที่มีราคาแพงเอาไว้ในห้องพิเศษเฉพาะ และมีความมุ่งหมายต้องการคุณภาพของเสียงที่สูง โดยทั่วไปทั้งภาพและเสียงจะถูกส่งออกไปด้วยกัน (video conference) และเครื่องมือที่ใช้ในการสื่อสาร โดยทั่วไปคุณภาพของอุปกรณ์จะมีผลต่อความเป็นธรรมชาติของสัญญาณเสียงซึ่งจะมีผลต่อองค์ประกอบของสัญญาณเสียงอัน ได้แก่ค่าแบนด์วิดท์, monophonic Transmission, การควบคุมเสียงสะท้อน ดังนั้นการที่จะให้สัญญาณเสียงที่มีคุณภาพจึงจำเป็นต้องมีการควบคุมเสียงสะท้อนที่ดีด้วย

ในระบบออดิโอคอนเฟอร์เรนซ์จะมีขนาดของอุปกรณ์จะเล็กกว่าที่ใช้ในระบบการประชุมทางไกลและจะคิดคั้งเอาไว้ภายในห้อง คุณภาพของเสียงจะต้องดีเพียงพอต่อการสนทนาและการประชุม ดังนั้นการออกแบบทางด้านอะคูสติกและระบบการควบคุมสัญญาณสะท้อนกลับจะต้องมีความปราณีตในการออกแบบ ตัวอย่างของชุดออดิโอคอนเฟอร์เรนซ์ที่มีในท้องตลาดที่ใช้ส่งในระบบ ISDN ซึ่งจะส่งไปในช่วงแบนด์กว้าง ได้ดีเท่าการส่งในแบนด์แคบ

- กลุ่มของระบบ Hand-free ที่ใช้ในกิจการโทรศัพท์(เหมาะกับการใช้งานเฉพาะและจะมีความแตกต่างกันกับกลุ่มที่กล่าวมาแล้วข้างต้น) เราสามารถที่จะแบ่งแยกย่อยสำหรับการใช้งาน ในกิจการ โทรศัพท์ได้เป็น การใช้งานสำหรับที่สำนักงานหรือที่บ้าน , สำหรับวิทยุเคลื่อนที่ และวีดีโอ โฟน

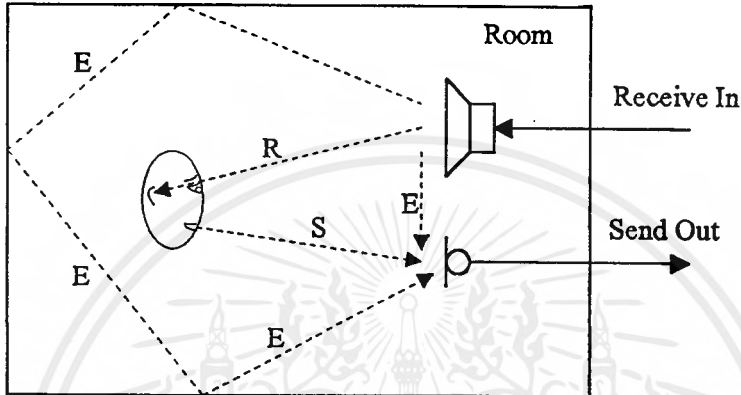
ในระบบโทรศัพท์ไร้สายซึ่งก็อาจจะต้องมีการควบคุมเสียงสะท้อนเนื่องจากว่าในชุดรับส่งนั้น ไมโครโฟนจะต้องอยู่ใกล้กับส่วนของหูฟัง จริงแล้วในส่วนของแฮนด์เซตจะมีชุดอุปกรณ์การฟังอยู่อิสระเพื่อส่งไปยังผู้ฟัง นั่นคือสัญญาณเสียงถูกส่งแพร่ออกไป และจะมีเสียงอีกส่วนที่เดินทางไปเข้ายังไมโครโฟน ซึ่งจะทำให้คุณภาพของเสียงที่ต่ำกว่ากลุ่มของออดิโอเทอร์มินอลที่มีจัดระบบที่ดีเมื่อใช้เทียบกับคุณภาพของเสียงที่เป็นธรรมชาติของระบบโทรศัพท์มาตรฐานในช่วงเวลาสั้นๆ(ไม่กี่วินาที) อย่างไรก็ตามก็ยังมีคามจำเป็นในการควบคุมเสียงสะท้อนที่มีประสิทธิภาพเพียงพอ อีกปัญหาหนึ่งที่สำคัญในการสื่อสารของชุด โทรศัพท์มาตรฐานก็คือด้านที่มีความไวในการรับเสียงสัญญาณสะท้อนสูงจะต้องมีการลดทอนของสัญญาณเสียงสะท้อนที่มากด้วยเมื่อมีการพูดและการฟังผ่านชุดโทรศัพท์ ซึ่งเราจะแสดงปัญหาในการประยุกต์ใช้งานแต่ละชนิด

- วิดีโอโฟน ความสัมพันธ์กันระหว่างจังหวะปากของผู้พูดและเสียง จำเป็นที่จะต้องมีการชดเชยการหน่วงเวลาของวิดีโอไลค์เตอร์ซึ่งโดยทั่วไปจะมีค่ามาก(150-300 msec ในระบบทั่วไป) โดยค่าของสัญญาณเสียงสะท้อนที่เหลื่ออยู่จะต้องมีค่าที่เหมาะสม ตามการจัดวางระบบการควบคุมเสียงสะท้อนที่เหมาะสมเพื่อให้มีระดับสัมพันธ์ของสัญญาณเสียงสะท้อนมีค่าต่ำเนื่องจากการหน่วงในระบบ

- ระบบโทรศัพท์ hand-free ในวิทยุเคลื่อนที่ การหน่วงเวลาในการส่งข้อมูล 2 ทางในระบบ(เป็นไปถึง 180 ms ใน GSM)จึงจำเป็นต้องมีการควบคุมเสียงสะท้อน ตัวอย่างเช่น การใช้โทรศัพท์ในรถยนต์ ที่วิ่งไปด้วยความเร็ว อัลกอริทึมที่ใช้ในระบบการควบคุมเสียงสะท้อนจะต้องออกแบบให้มีการจัดการเกี่ยวกับค่า acoustic noise pick up ของไมโครโฟน โดยเฉพาะอย่างยิ่งถ้าเป็นอุปกรณ์ที่มีการใช้งานอย่างแพร่หลายจึงต้องออกแบบให้ระบบมีราคาถูกและมีคุณภาพ

การวางแผนของระบบการควบคุมเสียงสะท้อน

คุณสมบัติของเสียงสะท้อน



รูปที่ 2.1.1 เป็นการช่วยอธิบายให้เห็นถึงการสื่อสารทางเคเบิลที่ใช้ระบบ Hand-free ออดีโอเทอร์มินอล โดยจะประกอบไปด้วย 3 ส่วนคือ

- ส่วนการส่ง (คือส่วนที่เกิดขึ้นระหว่างปากของผู้พูด ไปยังไมโครโฟน)
- ส่วนการรับ(คือส่วนที่เกิดขึ้นจาก ไมโครโฟน ไปยังหูฟังของผู้พูดเอง)
- ส่วนการสะท้อน(คือส่วนที่เกิดขึ้นจากลำโพง ไปยังไมโครโฟน)

ยกตัวอย่างเช่น เราถือว่าไม่ให้มีการเปลี่ยนแปลงในการส่งและรับสัญญาณเสียง ความมุ่งหมายในส่วนนี้จะประมาณว่าขึ้นอยู่กับทางเลือกชนิดและการวางตำแหน่งของตัวแปลงสัญญาณ(ลำโพงและไมโครโฟน)

ในส่วนของทางเดินของเสียงสะท้อนจะมีทั้งการแพร่ของเสียงทั้งทางตรงจากลำโพง ไปยังไมโครโฟนและส่วนของสัญญาณที่มาจากลำโพงแล้วเดินทางไปตกกระทบและสะท้อนกับผนังของห้องหรือจากส่วนต่างๆที่อยู่ภายในห้อง เช่น เฟอร์นิเจอร์, คน เป็นต้น แล้วเดินทางมาตกกระทบไปที่ไมโครโฟน คุณสมบัติของเส้นทางการเดินของเสียงจะขึ้นอยู่กับองค์ประกอบหลายๆประการ เช่น ยานความถี่ในช่วงผ่าน, ทิศทางการวางของไมโครโฟนและลำโพง, ระยะห่างระหว่างของอุปกรณ์ทั้งสอง, คุณสมบัติทางอะคูสติกของห้อง(เช่นค่า Reverberation time) เป็นต้น เนื่องจากการยากที่จะหาโมเดลที่มีความถูกต้องของปรากฏการณ์การสะท้อน แต่เราสามารถที่จะลดการสะท้อนลงได้โดยใช้ตัวกำจัดเสียงสะท้อน(Echo Canceller) ที่มีความเชื่อถือสูง โดยทำการสมมุติให้เป็นโมเดลของกล่องดำ(black box model) ซึ่งจะช่วยให้มองเห็นความสัมพันธ์ต่อผลการตอบสนองต่ออิมพัลส์มากขึ้น ซึ่งพอสรุปได้ดังนี้

- จะมีจำนวนของสัมประสิทธิ์จำนวนมากของการตอบสนองต่ออิมพัลส์ โดยจะมีช่วงตั้งแต่ 200 สำหรับระบบโทรศัพท์แบบ hand-free ในรถยนต์(มีค่าประมาณ 30 ms เมื่อทำการสุ่มตัวอย่างที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่ 8 KHz เนื่องจากปริมาตรปิดของรถยนต์จะมีค่า reverberation time ต่ำๆ) ไปจนถึง 8000 ในห้องประชุมขนาดใหญ่(ประมาณ 500 ms เมื่อทำการสุ่มตัวอย่างที่ 16 KHz)

- ผลการตอบสนองต่ออิมพัลส์จะเปลี่ยนแปลงไปตามเวลา ที่เกิดขึ้นจากการเปลี่ยนแปลงเกิดขึ้นภายในห้อง จึงทำให้คุณสมบัติของเส้นทางเดินของเสียงเปลี่ยนแปลงไปจากเดิม
- สัญญาณเสียงที่รับเข้ามาจะเป็นสัญญาณที่สถานะไม่คงที่ของสัญญาณในลักษณะ large spectral dynamics (กำลังงานส่วนใหญ่ของสัญญาณเสียงจะอยู่ที่ช่วงความถี่ต่ำๆ)
- ช่องสัญญาณ ในการส่งสัญญาณเสียงจะต้องมีลดการรบกวนที่เกิดจากเสียงที่มาจากลำโพงและ background noise

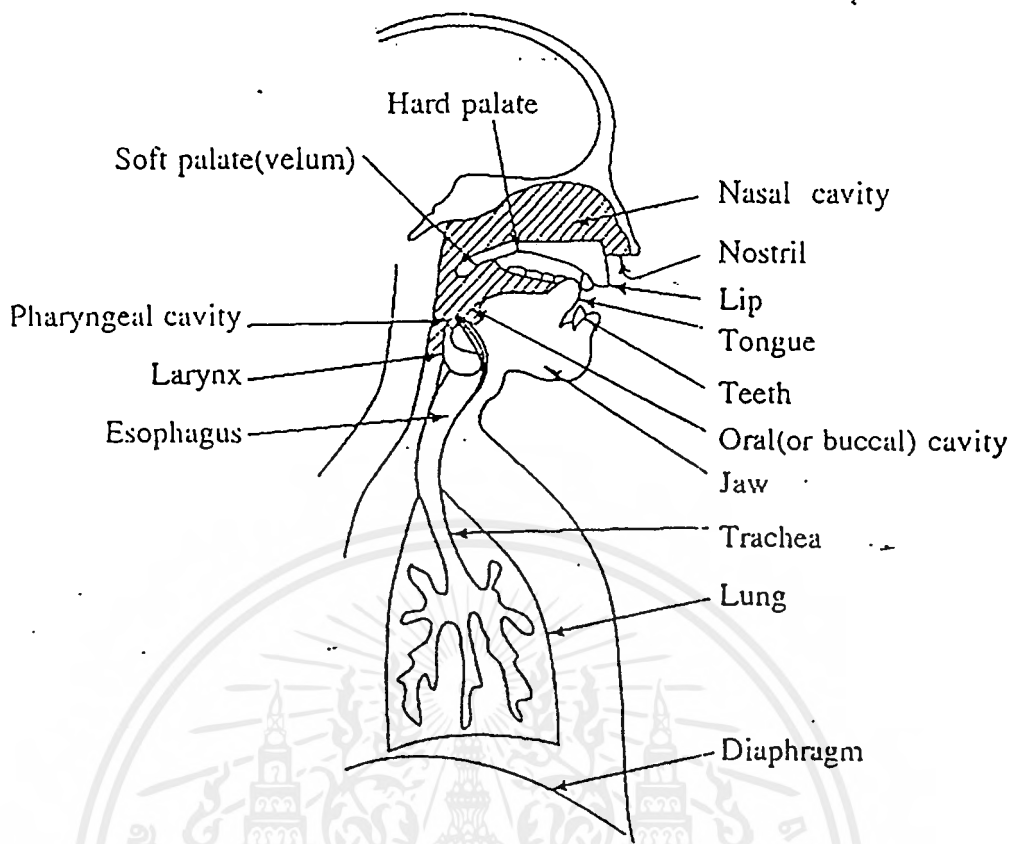
โดยปกติสัมผัสทั้ง 5 ของมนุษย์ ซึ่ง ได้แก่ การมองเห็น, การได้กลิ่น, การรับรู้รสสัมผัสจากลิ้น, การได้ยินเสียง และ ความรู้สึกสัมผัสจากผิวหนัง และจากประสาทสัมผัสทั้ง 5 นี้ เราได้มีการนำมาใช้ในการติดต่อสื่อสารกันซึ่ง ได้แก่ การมองเห็นภาพ, การแสดงออกทางการสื่อสารทางอวัจนะภาษาซึ่งใช้ภาษามือ ในการเป็นสื่อกลางในการติดต่อสื่อสาร และการสื่อสารทางเสียงซึ่งเราจะพบเจอและได้ใช้ในชีวิตประจำวัน อยู่เสมอ และในปัจจุบันแม้ว่าการสื่อสารจะได้พัฒนาไปมากอันด้วยความเจริญทางด้านเทคโนโลยีแต่ก็ปฏิเสธไม่ได้ที่จะต้องยังคงมีการสื่อสารที่ผ่านทางประสาทสัมผัสจากการมองเห็นและการได้ยิน

สัญญาณเสียงพูด

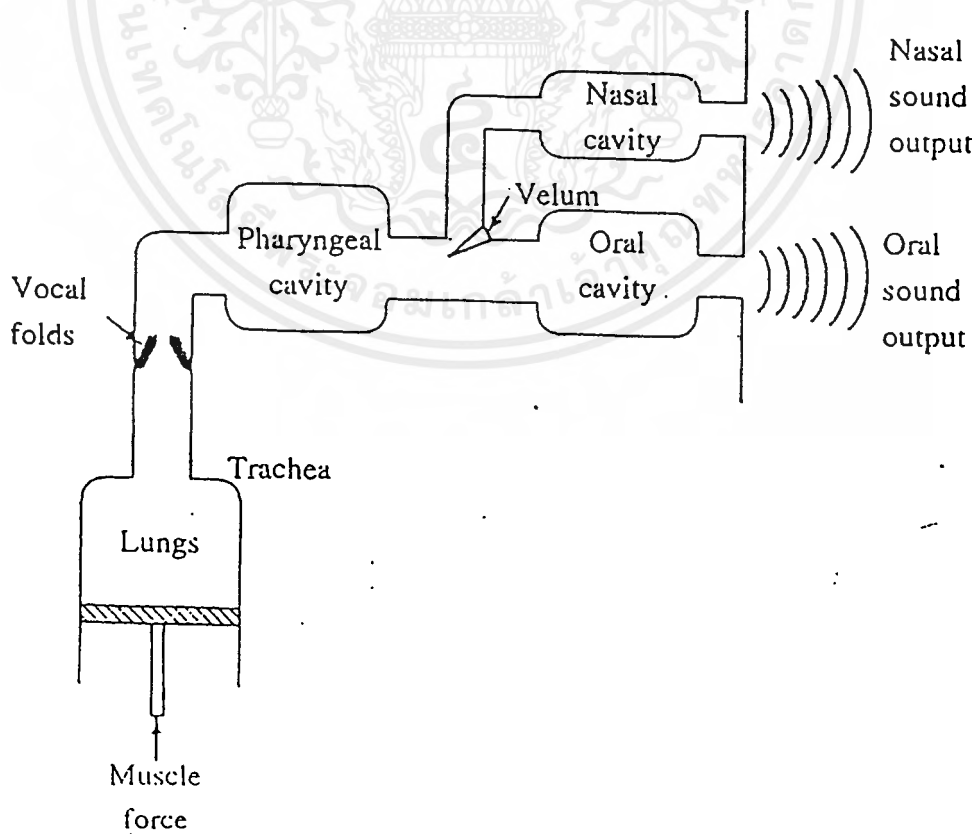
สัญญาณเสียงของมนุษย์จะมีส่วนสำคัญอยู่ 2 ส่วน ก็คือ ช่วงที่มีสัญญาณ และช่วงที่ไม่มีสัญญาณ โดยสัญญาณเสียงกำเนิดขึ้นมาได้จากการสั่นสะเทือน มนุษย์เราอาศัยกระบังลมให้ลมในปอดผลักลมออกมาผ่านทางหลอดลมสู่กล่องเสียง และที่กล่องเสียงนี้จะมีเส้นเสียงสำหรับกำเนิดเสียงเบื้องต้นของมนุษย์ และเสียงที่ออกมาจากกล่องเสียงนี้ จะผ่าน โครงสร้างของช่องปาก ซึ่งประกอบไปด้วย เพดานอ่อน(ลิ้นไก่), เพดานแข็ง, ลิ้น เหงือก และ ริมฝีปาก นอกจากนี้แล้วเสียงบางส่วนยังออกไปทางช่องจมูก โดยการควบคุมจากลิ้น ไก่ออกไปยังจมูกดังแสดงในรูปที่ 2.1.2

การศึกษาในการกำหนดสัญญาณอะคูสติกของเสียงนั้น ได้มาจากเส้นเสียงที่อยู่ในลำคอซึ่งจะมีสัญญาณที่เป็นฮาร์โมนิกส์ของสัญญาณเสียงเบื้องต้น และเนื่องจากความไม่สม่ำเสมอของรูปร่างของหูที่รับสัญญาณเสียงดังนั้นจึงทำให้ขนาดของสเปกตรัมของสัญญาณเสียงมีการกระจายออกไปอย่างไม่สม่ำเสมอออกไปจากจุดที่รับฟัง ได้อย่างชัดเจน นั่นคือสัญญาณเสียงที่รับฟังผ่านทางหูจะมีการเปลี่ยนแปลงทางด้านสเปกตรัม

ในการสื่อสารทางจิตตอลนั้นการจำกัดค่าของแบนด์วิธของสัญญาณเสียงนับว่าเป็นสิ่งที่มีความจำเป็นมาก และโดยปกติเมื่อวิเคราะห์ถึงขนาดของสัญญาณเสียงในแต่ละสเปกตรัมแล้วจะเห็นได้ว่ามีค่าที่ลดน้อยลงเมื่อเทียบกับที่ความถี่สัญญาณเสียงพื้นฐาน และสังเกตได้ว่าความแรงของสัญญาณเสียงที่มีขนาดของสัญญาณที่ยังมีผลต่อการได้ยินจะอยู่ในช่วงที่ต่ำกว่า 4000 Hz ซึ่งค่าความถี่ที่มีขนาดของสัญญาณที่มีผลต่อการรับฟังเราจะเรียกว่า "Formants" ซึ่งจะมีอยู่ 3 Formant ดังที่แสดงในรูปที่ 2.1.4 แสดงให้เห็นเมื่อเริ่มต้นการเปล่งเสียงและช่วงห่างระหว่าง Formant ที่ 1 และที่ 2 จะประมาณ 600-1500 Hz ซึ่งช่วงนี้จะมีค่ากำลังงานของสัญญาณเสียงมีค่าน้อย ซึ่งเรียกว่า Spectral gap

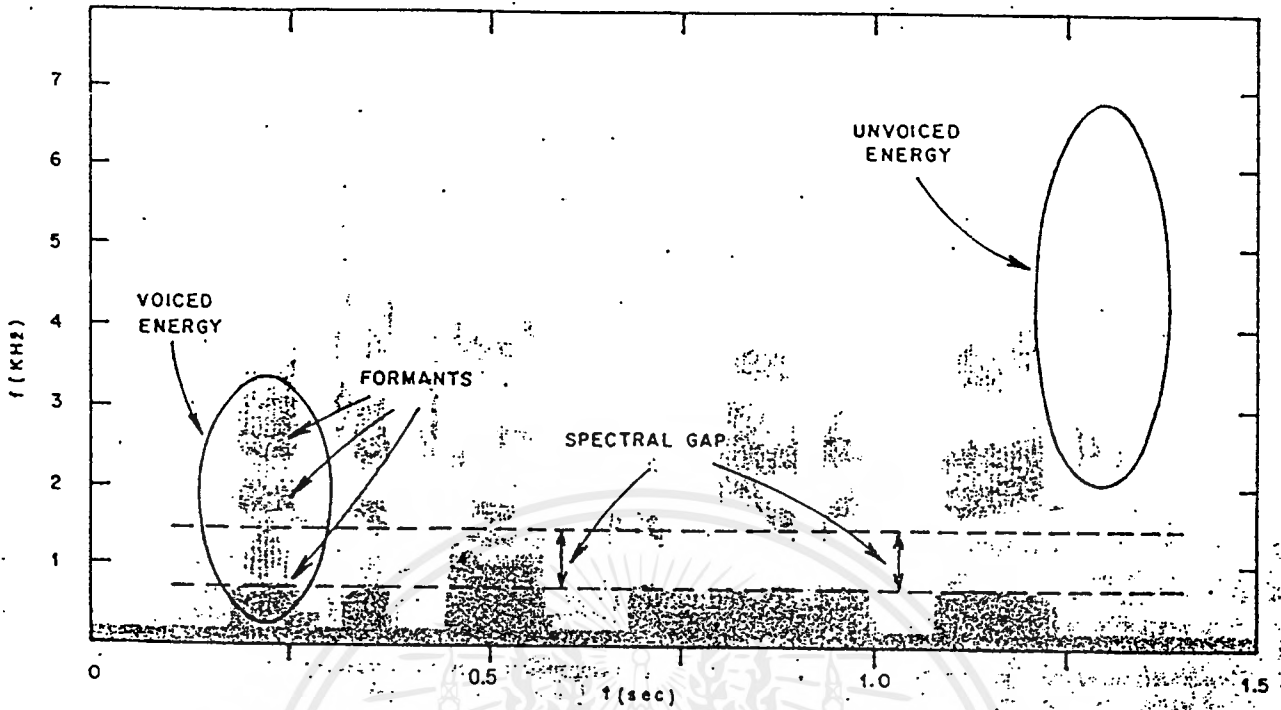


รูปที่ 2.12 โครงสร้างของเส้นทางเดินเสียงของมนุษย์



รูปที่ 2.13 แสดงภาพจำลองโครงสร้างของเส้นทางเดินเสียงของมนุษย์

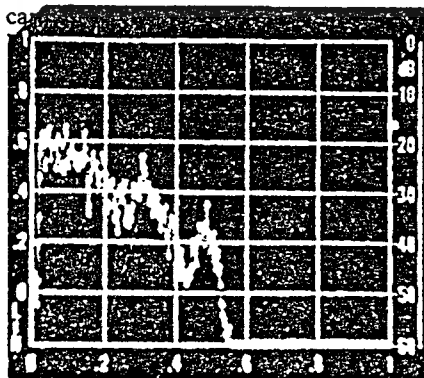
เอกสารนี้เป็นเอกสารลิขสิทธิ์ส่วนตัวที่จัดทำขึ้นเพื่อการศึกษาค้นคว้าเท่านั้น เมื่อผู้ใดเห็นสมควรขอใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1.4 แสดงสเปกโตรแกรมของสัญญาณเสียง ในการสื่อสารทางคำณวจิตตอล

ในขณะที่ไม่มีสัญญาณเสียง(Unvoiced) ที่เกิดขึ้นในสภาวะของการเปล่งเสียงนั้นซึ่งจะเกิดขึ้นเมื่อผู้พูดใช้ลิ้น, ริมฝีปากและฟัน เกิดการกระทบกัน และเนื่องจากการกระจายทางขนาดของสเปกตรัมทางความถี่ของช่วง Unvoice นั้นจะเกิดขึ้นที่ความถี่ที่สูงกว่า 1500 Hz และมีลักษณะที่เหมือนกับสัญญาณรบกวนซึ่งจะไม่มีลักษณะของสัญญาณที่ไม่เป็นคาบ

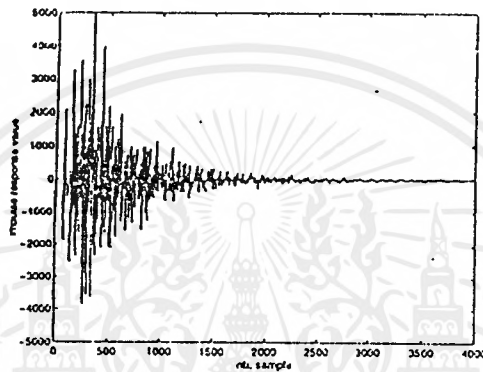
ความสำคัญของสัญญาณเสียงจะมีรูปสเปกตรัมของสัญญาณเสียงซึ่งจะมีค่าที่แตกต่างกันออกไปของแต่ละบุคคล โดยในช่วงของทั้ง 3 Formants ในรูปที่ 2.1.4 ซึ่งมีค่าช่วงของความถี่ที่ต่ำกว่า 2.5 KHz สัญญาณเสียงที่ประกอบไปด้วย 3 Formants จะมีคุณสมบัติที่ดีทั้งในด้านความเป็นภาษาและคุณสมบัติของการรับฟัง(intelligibility and listenability)



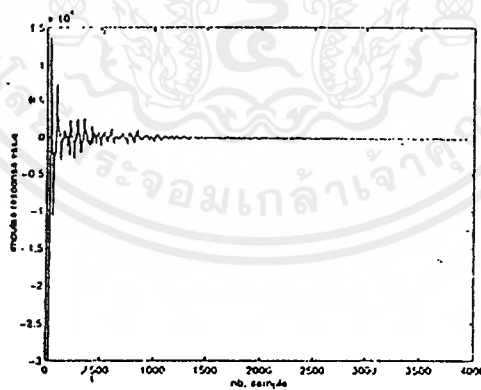
รูปที่ 2.1.5 แสดงให้เห็นถึงค่าขนาดของสเปกตรัมของสัญญาณเสียง

ถ้ารับในรูปที่ 2.1.5 แสดงให้เห็นถึงค่าขนาดของสเปกตรัมของสัญญาณเสียง โดยที่สังเกตได้ว่ามี 3 Formants ที่แสดงให้เห็นถึงจุดที่มีค่าความแรงของแพทเทอร์นในรูป นอกจากนั้นส่วนประกอบของสัญญาณที่สูงกว่า 2.5 KHz นั้นซึ่งดูได้จากรูปจะเห็นที่สเกลตรงกลางตามแนวนอน นั้นจะมีขนาดที่ลดลงมาเป็น 20 -25 dB เมื่อเทียบกับที่ความถี่ 2.5 KHz

ตัวอย่างของเส้นทางเดินของเสียงที่มีการตอบสนองต่ออิมพัลส์ที่แสดงในรูปที่ 2.1.6 (การประชุมทางไกล)และ ในรูปที่ 2.1.7 (โทรศัพท์แบบ Hand-free) โดยผลของการตอบสนองต่ออิมพัลส์จะถูกคุมตัวอย่างที่ความถี่ 16 KHz (สัญญาณเสียงในช่วงกว้าง)



รูปที่ 2.1.6 Impulse response : teleconference room



รูปที่ 2.1.7 Impulse response : hands-free telephone

เป็นที่น่าสังเกตว่ารูปร่างของผลการตอบสนองต่ออิมพัลส์ทั้ง 2 รูปจะมีความแตกต่างกันคือ

ในรูปที่ 2.1.6 จะเป็นของห้องประชุมทางไกล ซึ่งจะรวมเอาลำโพงและไมโครโฟนเข้ากับอุปกรณ์ในการประชุมทางไกล สังเกตเห็นว่าขนาดความแรงของสัญญาณสูงสุดนั้นจะไม่สูงมาก เนื่องมาจากทำให้ความสนใจในการจัดวางตำแหน่งของไมโครโฟนกับลำโพง ซึ่งจะมีผลให้มีทิศทางการเดินทาง

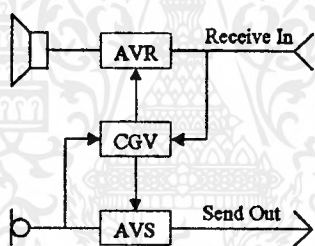
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของเสียงที่สะท้อนเข้ามาค่า และจะมีค่าลดลงตามคุณสมบัติของค่า Reverberation time ของห้อง ส่วนรูปที่ 2.1.7 เป็นของโทรศัพท์ระบบ hand-free ในสำนักงาน ซึ่งจะเห็นว่าค่าความแรงของสัญญาณอิมพัลส์จะสูงมากในช่วง เริ่มต้นความถี่ของเสียงสะท้อนจากลำโพง ไปยัง ไมโครโฟน แล้วค่าการตอบสนองจะลดต่ำลงเช่นเดียวกับของการประชุมทางไกลแต่การลดค่าลงจะมีเร็วกว่า สังเกตว่าค่าสูงสุดของผลการตอบสนองต่ออิมพัลส์ที่แตกต่างกันแม้ว่าจะส่งและรับที่ระดับเดียวกัน ผลที่ตามมาของระบบโทรศัพท์ hand-free จะเกิดเสียงหอนเกิดขึ้นเราเรียกว่า "Larsen effect" มากกว่าระบบการประชุมทางไกล จากตัวอย่างทั้งสองนี้แสดงให้เห็นถึงคุณสมบัติของเส้นทางเดินของสัญญาณเสียงที่บางที่จะแตกต่างกันตามชนิดของเทอร์มินอล

เทคนิคการควบคุมสัญญาณเสียงสะท้อน

1. การก่ระดับสัญญาณสะท้อนลง

มีหลายเทคนิคที่ใช้ในการจัดการที่เกี่ยวกับการควบคุมเสียงสะท้อน วิธีหนึ่งที่ย่าง ก็คือ เทคนิคการก่ระดับของสัญญาณลง โดยการปรับเปลี่ยนค่าเกณฑ์การขยายหรือการใส่ค่าการลดทอน ทางด้านรับหรือทางด้านส่งเพื่อทำการควบคุมสัญญาณเสียงสะท้อนแสดงให้เห็น ดังรูป



รูปที่ 2.1.8 AVS และ AVR จะปรับเปลี่ยนค่าการลดทอนอย่างต่อเนื่องโดยการควบคุมของ CGV ความระดับของสัญญาณที่รับได้ทั้งทางด้านรับและทางด้านส่ง

ข้อเสียของวิธีนี้ก็คือ ในช่วงที่ผู้พูดเริ่มต้นการสื่อสาร โดยช่วงเริ่มต้นของคำที่จะพูดนั้นจะเกิดการตัด (clip) รูปสัญญาณและความยากในการ Double talk โดยเฉพาะเมื่อมีการเกิดเสียงสะท้อนที่มีเวลานานมากๆ เนื่องจากเรามีความต้องการที่จะหลีกเลี่ยงไม่ให้มีเสียงสะท้อนจึงจำเป็นต้องให้มีการลดทอนที่มากด้วย

2. ตัวกำจัดเสียงสะท้อนที่สามารถปรับค่าเองได้

การทำงานของตัวกำจัดเสียงสะท้อนในงานทางด้านอะคูสติกจะใช้หลักการของการปรับเปลี่ยนสร้างรูปแบบจำลองของเส้นทางเดินของเสียงสะท้อน โดยการปรับเปลี่ยนให้ค่า Input-output มีความสัมพันธ์กัน โดยนำเอาหลักการกรองความถี่ที่เป็นเชิงเส้นมาแก้ปัญหา เมื่อพิจารณาผลการตอบสนองต่ออิมพัลส์ที่มีค่านานมากๆ โดยโมเดลจะเป็นของระบบจะเป็นการกรองความถี่แบบ IIR FILTER อย่างไรก็ตามถ้าเราพิจารณาผลการตอบสนองต่ออิมพัลส์โดยเลือกช่วงที่จำกัด, ที่มีความพอเพียง, สามารถให้ความแม่นยำถูกต้องได้แล้ว เราก็สามารถที่จะเปลี่ยนการพิจารณาการกรองสัญญาณแบบ IIR โดยการเลือกพิจารณาผลการตอบสนองต่ออิมพัลส์ที่มีค่าจำกัดเท่านั้น

ปัญหาพิเศษบางประการของการควบคุมสัญญาณเสียงสะท้อน

ในงานของการควบคุมสัญญาณเสียงสะท้อนจะใช้อัลกอริทึมในการเปลี่ยนแปลงค่า(Adaptive Algorithm) ซึ่งจะต้องมีความเหมาะสมในบางประการดังนี้

- การมีภูมิคุ้มกันต่อสัญญาณรบกวน(Robustness) รวมถึง ความมีเสถียรภาพของระบบ และจะต้องมีการลู่เข้าที่รวดเร็ว(good convergence speed) และความสามารถในการแทรกซึมที่ค่าผิดพลาดน้อยที่สุด(ในบางครั้งจำเป็นที่จะต้องนำเอาค่า trade-off มาพิจารณาด้วย) เป็นที่น่าสังเกตว่าสัญญาณผิดพลาดที่เหลืออยู่นี้เป็นสิ่งที่ไม่ต้องการและจะเป็นการยากที่เราพบกับสัญญาณที่มีสภาวะที่แน่นอน ค่าความผิดพลาดนี้จะเป็นผลมาจากความยาวของการกรองสัญญาณ แต่อย่างไรก็ตามจะมีคุณสมบัติอยู่ 2 ประการ ที่เป็นการยากที่จะทำให้ได้สัญญาณความผิดพลาดตามที่ต้องการ (เมื่อสัญญาณที่เข้ามาเป็นสัญญาณเสียง{speech}และการสร้างวงจรกรองสัญญาณที่มีความยาวมากๆ)
- คุณภาพและการหน่วงเวลา(Quality and Delay) เมื่อมีความต้องการคุณภาพของสัญญาณเสียงที่พอเพียงต่อการใช้งานในแต่ละประเภทเพื่อให้เป็นไปตามมาตรฐานเดียวกัน ตัวอย่างเช่น ในงานทางด้านโทรศัพท์ ค่าความผิดเพี้ยนของสัญญาณจะต้องอยู่ในขอบเขตที่สามารถยอมรับได้ ส่วนการหน่วงเวลาที่ขมขมของระบบการสื่อสารแต่ละประเภทนับเป็นอีกปัจจัยหนึ่งที่สำคัญมาก โดยจะต้องมีการกำหนดค่าการหน่วงเวลามาตรฐานในงานแต่ละประเภท และอุปกรณ์ในการควบคุมการสะท้อนจะต้องทำให้มีการหน่วงเวลามีค่าต่ำสุดเท่าที่เป็นไปได้ ดังเช่น ในการส่งสัญญาณเสียงที่ใช้สำหรับการสื่อสารทางดิจิทัลแบบ End to End (ในตัวอย่างจะรวมถึงระบบ ISDN)จะต้องมีการหน่วงเวลาได้ไม่เกิน 16 msec และไม่เกิน 2 msec สำหรับระบบที่ใช้การเชื่อมต่อเป็นแบบสัญญาณเชิงอุปทาน(อนาล็อก)
- ราคา(Cost) การคำนวณที่มีความยุ่งยากน้อยนับว่าเป็นสิ่งที่ต้องการเนื่องจากจะทำให้อุปกรณ์ที่สามารถใช้งานได้มีราคาถูก ดังนั้นจึงจำเป็นจะต้องเลือกใช้อุปกรณ์ที่ง่ายในการใช้งานกับอัลกอริทึมที่สามารถปรับค่าเองได้ตามต้องการและการเลือกอุปกรณ์ DSP ที่เหมาะสมในการใช้งานแต่ละประเภทโดยคำนึงถึงต้นทุนการผลิต

ระบบควบคุมการคืนเสียงสะท้อน THE ACOUSTIC ECHO CONTROL "GLUE"

อุปกรณ์อย่างหนึ่งที่สามารถใช้เสริมเพื่อป้องกันการความเลวร้ายที่อาจเกิดขึ้นในช่วงที่เกิดการเปลี่ยนแปลงอย่างรวดเร็วของเส้นทางเดินของสัญญาณสะท้อนดังที่ได้กล่าวมาแล้ว อุปกรณ์นี้จะใช้หลักการของการเลื่อนทางความถี่ (frequency shifting) หรือใช้เทคนิคของการปรับปรุงสเกลทางเวลา (time-scale modification techniques) นอกจากนี้ถ้าระดับสัญญาณรบกวนที่เรียกว่า แบ็กกราวด์นอยส์ (background noise) มีค่าน้อยเพียงพอ สัญญาณเสียงสะท้อนที่เหลืออยู่ก็สามารถที่จะกำจัดออกไปได้สมบูรณ์โดยใช้อุปกรณ์แบบไม่เป็นเชิงเส้น เช่น central-clippers ซึ่งจะกดระดับความแรงของสัญญาณลง เพื่อที่จะเพิ่มประสิทธิภาพในการกำจัดเสียงสะท้อน ตัวอะแดปทีฟของตัวกำจัดเสียงสะท้อนโดยปกติจะมีส่วนของตัวกดระดับเสียงสะท้อน ซึ่งจะทำการกำจัดสัญญาณสะท้อนง่ายมากขึ้น ในขณะที่การสูญเสียที่เกิดจากตัวกดระดับสัญญาณสะท้อนมีค่าน้อย จึงสามารถนำมาใช้งานในการสื่อสารแบบ 2 ทิศทาง (full-duplex communication)

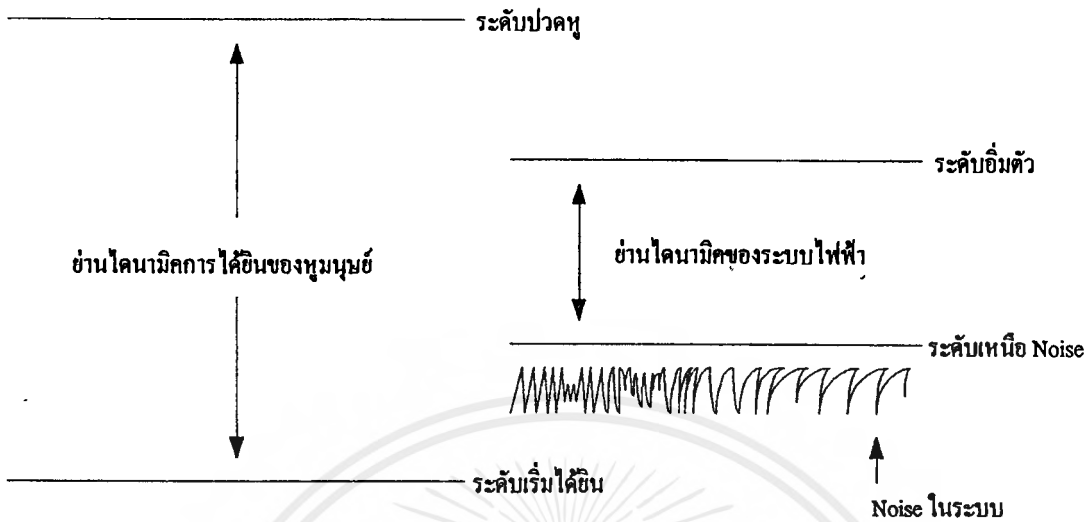
การประยุกต์ใช้งานของ Acoustic Echo Cancellation

เทคโนโลยี ของ AEC ซึ่งแต่เดิมได้มีการนำมาใช้งานในระบบ Hands-Free ตัวอย่างเช่น Speakerphones จากนั้นจึงได้มีการประยุกต์ใช้งานอื่นๆ ความมามีมากมายเช่น

- 1.High performance, full-duplex speakerphones สำหรับระบบ audioconferencing.
- 2.Group and desktop videoconferencing systems.
- 3.Digital simultaneous voice and data modems (DSVD).
- 4.Internet phones.
- 5.Computer telephony.
- 6.LAN-based workstation or PC-based multimedia terminals.
- 7.Remote or Tele-gaming over the internet or the phone network.
- 8.Digital telephone answering devices (DTADs).

2.2 คอมแพนเดอร์

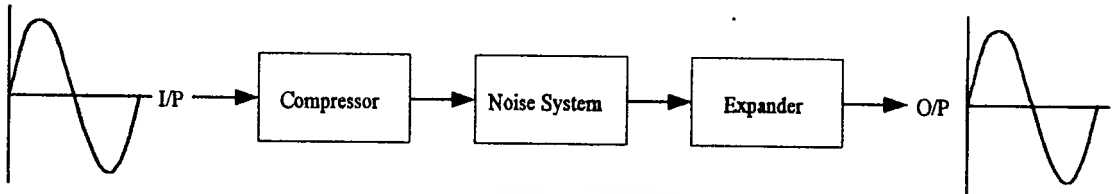
เสียงที่หูมนุษย์ได้ยินนั้นจะมีการเปลี่ยนแปลงทั้งขนาดความดังและความถี่ของสัญญาณที่มีการเปลี่ยนแปลงอยู่ตลอดเวลา โดยช่วงย่านของความถี่ที่หูได้ยินในทางทฤษฎีจะมีค่าอยู่ในช่วง 20 Hz -20 KHz ส่วนความดังจะอยู่ในช่วงระหว่างระดับเทรสโลที่หูสามารถรับฟังได้และขณะที่ความดังของสัญญาณเสียงที่มีค่ามากจะทำให้เกิดความรู้สึกปวดหู ซึ่งถ้าความดังของสัญญาณมากเกินไปจะทำลายโสตประสาทการรับฟังซึ่งเป็นอันตรายถึงขั้นหูอาจจะหนวกได้ ดังนั้นช่วงที่มีความดังของสัญญาณเสียงที่เหมาะสมแก่การรับฟังเราเรียกว่าช่วงย่านไดนามิกส์ (Dynamic Rang) เมื่อมีการแปลงสัญญาณเสียงให้เป็นสัญญาณไฟฟ้าโดยผ่านทรานสดิวเซอร์ซึ่งได้แก่ ไมโครโฟน แล้วส่งผ่านเขาสู่ระบบเพื่อทำการประมวลผลต่างๆ ในการสื่อสาร ซึ่งจะทำให้ช่วง ไดนามิกส์เปลี่ยนไปบ้างขึ้นอยู่กับตัวระบบ ซึ่งจะมีขีดจำกัดคือ ในช่วงย่านไดนามิกส์ด้านต่ำจะถูกจำกัดด้วยระดับของสัญญาณรบกวน (Noise) ที่เกิดขึ้นในระบบ ส่วนทางด้านสูงของช่วง ไดนามิกส์จะถูกจำกัดด้วยจุดอิ่มตัว (Saturation) ของตัวระบบ



รูปที่ 2.2.1 ลักษณะของระดับเสียง

จากรูปจะเห็นได้ว่าย่าน ไดนามิกส์ของหูมนุษย์จะมีความกว้างกว่าของระบบคิงที่กล่าวมาแล้วข้างต้น ดังนั้นในการแปลงสัญญาณ ให้เสียงให้เป็นสัญญาณ ไฟฟ้าช่วงความดังต่ำสุดจะต้องสามารถที่จะกลบสัญญาณเสียงรบกวนที่เกิดขึ้นจากระบบได้และที่ความดังมากๆ จะเกิดความผิดเพี้ยนเกิดขึ้นเมื่อเข้าใกล้จุดอิมิตัวของระบบ เพื่อเป็นการแก้ปัญหานี้สามารถทำได้โดยการควบคุมระดับความแรงของสัญญาณที่จะป้อนเข้าสู่ระบบ โดยที่ช่วงระดับที่มีสัญญาณต่ำๆ ก็จะต้องสามารถที่จะขยายระดับสัญญาณ ให้มีความแรงเหนือกว่าสัญญาณรบกวนในระบบ และที่ระดับสัญญาณที่เข้ามีความแรงมากเกินไปก็ต้องกดระดับให้มีความดังที่ต่ำกว่าจุดอิมิตัวของระบบ เราเรียกอุปกรณ์ที่ใช้ควบคุมความแรงของสัญญาณ โดยอัตโนมัติเพื่อให้ช่วงไดนามิกส์ของสัญญาณเข้ากับช่วง ไดนามิกส์ของระบบได้อย่างพอดีนี้ว่า คอมเพรสเซอร์ (Compressor) ส่วนที่ทางด้านรับจะต้องแปลงระดับช่วงย่าน ไดนามิกส์ของสัญญาณ ให้กลับเป็นเหมือนอย่างเดิมมิฉะนั้นแล้วจะทำให้สัญญาณเสียงที่รับได้จะผิดจากความเป็นจริงคือที่เสียงดังจะไม่ดังจริงและเสียงที่ค่อยจะไม่ค่อยจริง ในส่วนการแปลงช่วง ไดนามิกส์กลับนี้เราเรียกว่า เอ็กแพนเดอร์ (Expander) ซึ่งจะทำหน้าที่ควบคุมสัญญาณอย่างอัตโนมัติให้ได้ย่าน ไดนามิกส์กลับกว้างเหมือนดังเดิมคือ สัญญาณแรงที่ระดับหนึ่งก็จะขยายมากขึ้น และระดับความดังต่ำที่ระดับหนึ่งก็ลดให้ต่ำลง ไปอีก เมื่อรวมเราทั้ง 2 ส่วนเข้าด้วยกันเราจะได้กรรมวิธีในการที่จะบีบและขยายช่วงย่าน ไดนามิกส์นี้ว่า Compandor ซึ่งเป็นการรวมกันระหว่าง การ Compressor และ Expander

หลักการพื้นฐานของคอมแพนเดอร์



รูปที่ 2.2.2 การจัดวางตำแหน่งของการคอมแพนเดอร์

- การ Compressing

ขบวนการคอมเพรสชันที่ใช้กับเสียงพูดหรือเสียงดนตรี คือการลดช่วงไดนามิกส์ของสัญญาณเสียงขาเข้าก่อนป้อนเข้าสู่ระบบ ที่รับขนาดของสัญญาณได้ไม่ใหญ่นัก (ช่วงสัญญาณก่อนการเกิดโอเวอร์โพลด์มีค่าอยู่ในระดับที่ต่ำ) แล้วค่อยไปเพิ่มขนาดของสัญญาณที่ทางด้านออกเพื่อคืนสัญญาณออกมา ทางด้านอินพุทของวงจรคอมเพรสเซอร์ได้มีการขยายสัญญาณระดับต่ำให้สูงขึ้นและสัญญาณที่มีขนาดสูงๆ ก็จะถูกลดทอนลง อย่างไรก็ตามการทำเช่นนี้อาจก่อให้เกิดความผิดเพี้ยนของสัญญาณเสียงขึ้นบ้าง ซึ่งอยู่กับการออกแบบวงจรว่ายอมรับให้มีการผิดเพี้ยนได้มากน้อยเพียงใด

รายละเอียดอื่นๆ ที่สำคัญและมีความสัมพันธ์กับการคอมเพรสชัน เช่น

ค่าเวลา Attack Time หมายถึงเวลาที่วงจรใช้ในการตรวจสอบและได้ตอบสนองผลการเปลี่ยนแปลงทันทีทันใดของขนาดของสัญญาณที่รับเข้ามา ว่าทำได้รวดเร็วเพียงใด ทั้งนี้ขึ้นอยู่กับชนิดของสัญญาณเสียงด้วย(เสียงพูดหรือเสียงดนตรี) และคุณภาพการทำงานที่ต้องการ ค่า Attack time โดยทั่วไปในวงจรคอมเพรสเซอร์ควรมีค่าอยู่ในช่วง 0.5- 5 มิลลิวินาที

ค่า Release Time หมายถึงเวลาที่วงจรจะกลับคืนสู่ค่าเริ่มต้นเดิมได้อย่างรวดเร็วเพียงใดก่อนที่ขนาดสัญญาณจะเริ่มปรากฏเพิ่มขนาดใหญ่อีกครั้ง ปกติมีค่าหลายวินาที ซึ่งถ้าหากใช้ค่า Release Time สั้นเกินไปไปพฤติกรรมการลดทอนขนาดของสัญญาณในวงจรคอมเพรสเซอร์ไปรบกวนต่อองค์ประกอบสัญญาณในช่วงความถี่ต่ำๆ และถ้าเลือกใช้นานเกินไปจะทำให้เสียงจะผิดธรรมชาติไป ดังนั้นในบางวงจรจะต้องมีตัวปรับค่า Release Time โดยอัตโนมัติเมื่อไม่ให้เสียงผิดเพี้ยนไป

เนื่องจากวงจรคอมเพรสเซอร์จะทำการบีบช่วง ไดนามิกส์ให้มีค่าแคบลง โดยเป็นที่น่าสังเกตว่าการทำงานของวงจรคอมเพรสเซอร์จะบีบสัญญาณที่มีขนาดเล็กๆ ที่เดิมมีระดับสัญญาณที่ต่ำกว่าระดับของสัญญาณรบกวน เมื่อถูกบีบแล้วจะทำให้มีระดับที่สูงกว่าระดับสัญญาณรบกวน ดังนั้น ในการติดตั้งวงจรคอมเพรสเซอร์จะติดตั้งไว้ทางด้านส่งเสมอ

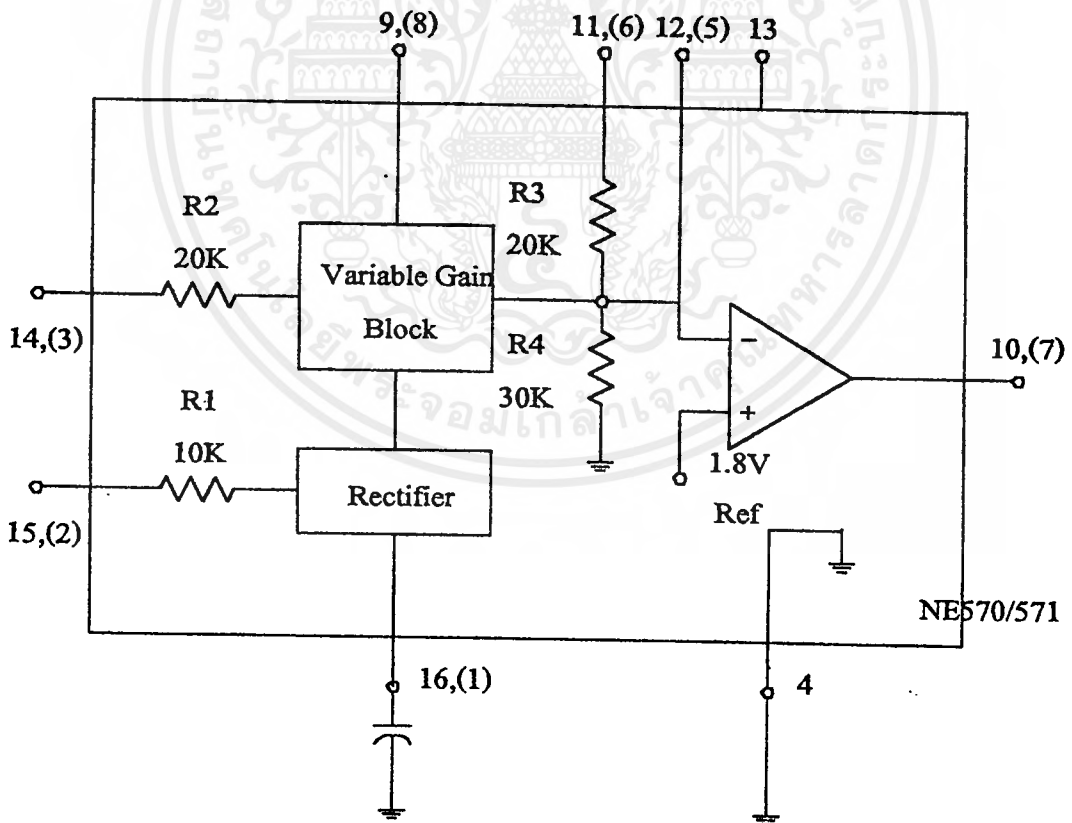
- Expanding

จะทำหน้าที่ขยายช่วง ไดนามิกส์ของสัญญาณให้มีค่าเท่าเดิมโดยการขยายสัญญาณที่มีขนาดที่แรงให้แรงยิ่งขึ้นคืออัตราเคียวกับกับที่ถูกบีบมาจากด้านส่ง ขณะเคียวกันก็จะลดสัญญาณเสียงที่ค่อยให้ค่อยลงไปอีก ซึ่งรวมถึงระดับของสัญญาณรบกวนซึ่งมีขนาดค่าก็จะถูกลดขนาดลงทำให้เป็นการปรับปรุงค่าของอัตราส่วนของสัญญาณคือสัญญาณรบกวน (Signal to Noise Ratio: S/N) การติดตั้งวงจรเอ็กแพนเดอร์จะติดตั้งไว้ทางด้านรับ

หลักการทํางานของวงจรคอมเพรสเซอร์

จากรูป ส่วนสำคัญของวงจรการคอมเพรสเซอร์จะประกอบไปด้วย Op-Amp ซึ่งจัดการขยายเป็นแบบอินเวอร์ต โดยอัตราการขยายจะขึ้นอยู่กับอัตราส่วนของค่าความต้านทานในส่วนของการป้อนกลับ และมีไดโอดคํออยู่ในส่วนของการป้อนกลับดังแสดงดังรูป เพื่อก่อให้เกิดการเปลี่ยนแปลงทางด้านเกณฑ์การขยายของวงจร ได้ทั้งที่เป็นซิกบวกและซิกลบ ในขณะที่ระดับของสัญญาณมีระดับแรงคั้นที่คํากว่าระดับที่จะทำให้ไดโอดทํางาน ได้ ทำให้ไม่มีผลคํอการทํางานของวงจรมันคืออัตราการขยายจะยังคงเป็นเหมือนปกติ แต่เมื่อระดับสัญญาณที่เข้ามามีค่าที่จะทำให้ไดโอดสามารถนํากระแส ได้ซึ่งเป็นผลทำให้ผลรวมของค่าความคํานทานทางด้านที่ป้อนกลับมีค่าลดลง ทำให้อัตราการขยายของวงจรมีค่าลดลง

ตัวอย่าง ไอซีคอมเพนเตอร์ เช่น IC #NE570/571



รูปที่ 2.2.3 โครงสร้างภายในของ NE 570/571

ภายในประกอบด้วยวงจรเดียวกัน แต่มีอยู่ด้วยกัน 2 ชุด ประกอบด้วยวงจรปรับค่าอัตราขยายด้วยแรงดันปูลเวฟเร็คติฟายเออร์ ในส่วนของภาซขยายจะมีแรงดันอ้างอิง 1.8 V และชุดความต้านทานปกคิสัญญาณอินพุตจะถูกต่อผ่านตัวเก็บประจุที่ขา 2 (15) เพื่อส่งเข้าภาคปูลเวฟเร็คติฟายเออร์ ขณะที่ขา 1 (16) จะต่อเข้ากับตัวเก็บประจุภายนอก เพื่อกรองแรงดันให้เรียบ กลายเป็นระดับสัญญาณไปควบคุมอัตราขยายในส่วน Variable Gain Block

ขณะเดียวกันสัญญาณที่ป้อนเข้าที่ขา 3 (14) เข้าสู่วงจรที่มีอัตราขยายถูกควบคุมด้วยแรงดันจากขา 1 (16) ความถี่ของสัญญาณสามารถปรับให้น้อยได้โดยป้อนแรงดันที่ขา 8 (9) อย่างเหมาะสม เอ้าท์พุทของวงจรในภาคนี้จะถูกส่งไปเข้ายังขาอินเวอร์ตติ้งของ Op-Amp เป็นภาคสุดท้ายโดยมี R3 และ R4 ต่อร่วมกันที่ขาอินเวอร์ตติ้ง ในส่วนของ Op-Amp จะมีการชดเชยความถี่อยู่ภายใน และที่ขาอนอินเวอร์ตติ้งจะต่ออยู่กับระดับแรงดันอ้างอิงคงที่ 1.8 โวลท์

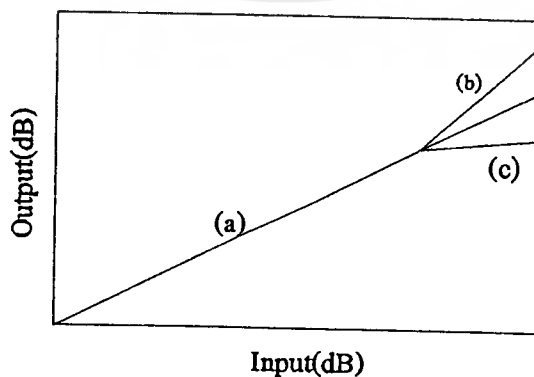
การกกระดัดและการขยายกว้างออกของสัญญาณที่มีความต่อเนื่องโดยการใช้อุปกรณ์ที่ไม่เป็นเชิงเส้น

(Analog Compression and Expansion Processing with Nonlinear Element)

เนื่องจากคุณสมบัติของสัญญาณเสียงจะมีช่วงย่านไดนามิกที่กว้าง เมื่อสัญญาณเสียงหรือเสียงดนตรีที่ทำการบันทึกหรือทำการส่งซึ่ง โดยปกติแล้วจะมีการลดระดับไดนามิกเร้นซลงซึ่งจะทำให้สัญญาณเสียงที่มีความนุ่มนวลลดลง (soft to loud) ซึ่งจะมีการใช้กันอย่างมากในการสื่อสารวิทยุ และโดยปกติแล้วการกกระดัด (compression) จะมีความสำคัญเป็นอย่างมากในการนำเอาไปใช้กับสัญญาณที่มีช่วงไดนามิกที่กว้างกว่าและช่วยการนำเอาอุปกรณ์ที่ไม่เป็นเชิงเส้นมาเป็นตัวกกระดัดของสัญญาณซึ่งจะมีผลทำให้เกิดความผิดเพี้ยนที่เกิดจากสัญญาณฮาร์โมนิกส์เพิ่มมากขึ้น

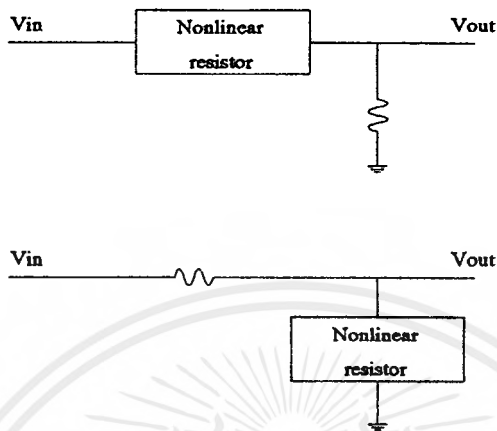
ในบางครั้งผลที่ไม่ต้องการของการกกระดัดของสัญญาณเราสามารถที่จะสามารถลดลงได้เมื่อทางภาครับได้มีการออกแบบวงจรที่มีการขยายกว้างออกของไดนามิกเร้นซและจะต้องมีคุณสมบัติของเกณฑ์ที่เหมือนกันด้วย (คุณสมบัติของเกณฑ์หรือที่เรียกกันว่า Transfer Function) ซึ่งจะมีหน้าที่เปลี่ยนคุณสมบัติทางเกณฑ์ของการส่งข้อมูลที่มีการกกระดัดของสัญญาณ

ในรูปฟอร์มของการกกระดัดสัญญาณเสียงและการขยายกว้างออกของสัญญาณจะเป็นโมเดลของการขยายที่เป็นเชิงเส้นที่มีการนำเอาไดโอดที่ไม่เป็นเชิงเส้น



รูปที่ 2.24 รูปฟอร์มของการกกระดัดสัญญาณเสียง

-สำหรับวงจรการกระดึบของสัญญาณเสียงนั้น ไคโอดหรือหลอดจะต่อเพื่อให้เอาท์พุทลดลงอย่างไม่มีเป็นเชิงเส้นเมื่อขนาดแรงดันของสัญญาณอินพุทมีค่าเพิ่มมากขึ้นซึ่งสามารถดูได้จากรูปที่ 2.5.4 และในรูปที่ 2.5.5 แสดงการต่อวงจรเพื่อขยายกว้างออกของสัญญาณซึ่งวงจรของไคโอดจะให้ค่าของเกนที่จะทำให้ขนาดของสัญญาณเอาท์พุทมีค่าเพิ่มมากขึ้นเมื่ออินพุทมีค่าแรงดันเพิ่มมากขึ้น



รูปที่ 2.5.5 แสดงการต่อวงจรเพื่อขยายกว้างออกของสัญญาณ

สมการสำหรับการกระดึบของสัญญาณและการขยายกว้างออกของสัญญาณจะเป็นดังนี้

$$V_{out} = A V_{in} \text{ หรือ } A = V_{out}/V_{in} \text{ (Transfer function)}$$

สำหรับการกระดึบของสัญญาณ สมการข้างต้นจะมีทรานสเฟอร์ฟังก์ชันเป็น

$$V_{out} = A^k V_{in}$$

เมื่อ k แทนความไม่เป็นเชิงเส้นของเกนซ์ สำหรับการกระดึบของสัญญาณค่า $k < 1$

และในส่วนของวงจรการขยายออกของสัญญาณสมการข้างต้นจะเป็น

$$V_{out} = A^k V_{in} \quad \text{เมื่อ } k > 1$$

2.3 LMS Algorithm

ในระบบการสื่อสารทางไกลนั้นเสียงสะท้อนนับเป็นปัญหาที่สำคัญอย่างยิ่งในการสื่อสารเนื่องจากจะมีผลต่อคุณภาพในการรับส่งสัญญาณ กระบวนการในการกำจัดสัญญาณเสียงสะท้อนจะต้องอาศัยกระบวนการทางด้านการประมวลผลทางดิจิทัลซึ่งจะมีวิธีการในการกำจัดเสียงสะท้อนโดยอาศัยวงจรฟิลเตอร์ที่สามารถปรับค่าได้เองอย่างอัตโนมัติ (Adaptive Filter) และเราสามารถพิจารณาการทำงานของวงจรได้ทั้งด้านโดเมนทางเวลา (Time Domain Adaptive Filter) และด้านโดเมนทางความถี่ (Frequency Domain Adaptive Filter)

-Time Domain Adaptive Filter

จะอาศัยวงจรฟิลเตอร์ชนิด ทรานสเวอร์ซอลฟิลเตอร์ (FIR Transversal Filter) ในการใช้ป็นเครื่องมือช่วยในการพิจารณาทางด้านโดเมนทางเวลา ซึ่งจะอาศัยอัลกอริทึมของค่าประมาณแบบกำลังสองต่ำสุด (Least Mean Square: LMS Algorithm) สัญญาณอะคูสติคที่ป้อนกลับโดยมีเส้นทางเดินของเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากลำโพงไปยัง ไมโคร โฟนภายในห้องส่ง จะทำให้เกิดเสียงสะท้อนเกิดขึ้นซึ่งจะมีค่ามากขึ้นอยู่กับขนาดของห้องและองค์ประกอบทางด้านอะคูสติกของอุปกรณ์ต่างๆที่อยู่ภายในห้อง และสามารถมีการเปลี่ยนแปลงได้โดยง่ายเช่น การเปิด/หน้าต่างภายในห้อง, การเคลื่อนไหวของคนที่อยู่ภายในห้อง เป็นต้น ดังนั้นจึงต้องมีการ ใช้ตัวควบคุมเสียงสะท้อนที่สามารถปรับค่าได้เองอย่างอัตโนมัติเพื่อให้มีความยืดหยุ่นในการทำงานและจะต้องสามารถทำงาน ได้อย่างต่อเนื่องและเหมาะสมเมื่อเส้นทางเดินของสัญญาณมีการเปลี่ยนแปลงและมีการ ใช้ สัมประสิทธิ์ของการหน่วง Tap-Coefficients เพื่อจำลองเสียงสะท้อนที่เกิดขึ้นจริงในระบบและ จะนำเข้ากระบวนการของ FIR Filter โดยจะให้ค่า Weight แต่ละ Tap มีการปรับค่าเป็นแบบ Adaptive FIR Filter

$$y(n) = \sum_{i=0}^L w_i(n) \cdot x(n-i)$$

จากสมการของ steepest descent

$$w(n+1) = w(n) - \left(\frac{\mu}{2}\right) \nabla \xi \quad (2.3.1)$$

เมื่อค่า μ เป็นแฟคเตอร์ของการก้าวเข้าหรือ step size

ซึ่งเราสามารถเห็น ได้ว่าการเปลี่ยนแปลงจาก $w(n)$ ไป $w(n+1)$ จะมีทิศทางที่ลาดในทิศทางที่ลดลง ดังนั้นการเทร็คกิ่งของค่าน้ำหนักจะเป็น ไปตามการประมาณของ steep descent ตามคุณสมบัติของพื้นผิว(performance surface) แต่อย่างไรก็ตามในทางปฏิบัติค่าของ $d(n)$ และ $x(n)$ จะมีค่าในทางปฏิบัติเป็นค่าที่ไม่ทราบค่า ดังนั้นวิธีการของ steep descent จึงไม่สามารถใช้งานได้โดยตรง WIDROW จึงได้ใช้ค่าของ instantaneous squared error, $e^2(n)$ เพื่อนำมาใช้ให้ค่า mean-square error เป็น $\xi(n) = e^2(n)$

ดังนั้นการลดลงอย่างเหมาะสมโดยการใช้ LMS จึงใช้ instantaneous gradient ของ single square error sample

$$\nabla \xi(n) = 2[\nabla e(n)] e(n)$$

เมื่อ $e(n) = d(n) - w^T(n)x(n)$

$$\nabla e(n) = -x(n)$$

ดังนั้นค่าของการลดลงอย่างเหมาะสมจึงเป็น

$$\nabla \xi(n) = -2 x(n) e(n)$$

ดังนั้นเมื่อเรานำไปแทนในสมการของ steep descent algorithm สมการ(2.6.1)เราจะ ได้เป็น

$$w(n+1) = w(n) + \mu x(n) e(n)$$

ซึ่งเราเรียกว่าเป็น LMS Algorithm หรือ stochastic gradient algorithm ซึ่งเป็นอัลกอริทึมที่ง่ายและ ไม่ต้องการชกกำลังสองสมการเฉลี่ยหรือการทำอนุพันธ์

โดยหลักการโคแอดแกรมของ Widrow's LMS Algorithm สำหรับ Transversal Filter จะเป็น ไปดังรูปที่ 2.3.1 ซึ่งเราสามารถที่จะสรุปได้ดังนี้

1. การเลือกค่าพารามิเตอร์และการกำหนดค่าเริ่มต้น ได้แก่ $L, \mu, w(0)$

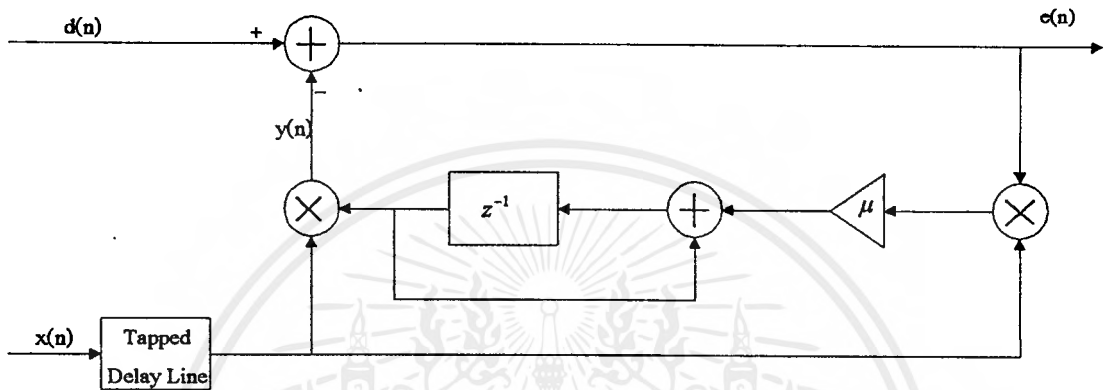
เมื่อ L เป็นลำดับของฟิลเตอร์

μ คือค่าของ step size

$w(0)$ เป็นค่าเริ่มต้นของเวกเตอร์ของน้ำหนักที่เวลา $n=0$

2. คำนวณหาค่า Output ของวงจร Filter

$$y(n) = \sum_{i=0}^L w_i(n) \cdot x(n-i)$$



รูปที่ 2.3.1 บล็อกไดอะแกรมของ Widrow's LMS Algorithm สำหรับ Transversal Filter

3. คำนวณหา Error signal

$$e(n) = d(n) - y(n)$$

4. ทำการปรับค่าของ เวกเตอร์ของน้ำหนักจาก $w(n)$ ไปเป็น $w(n+1)$ โดยการใช้ LMS Algorithm

$$w_i(n+1) = w_i(n) + \mu e(n) x(n-i)$$

การควบคุมให้เกิดความมีเสถียรภาพของ LMS Algorithm

เมื่อค่าเฉลี่ยของการลู่เข้าของค่าน้ำหนักใน LMS Algorithm จากค่าในสถานะเริ่มต้น $w(0)$ ไปยังค่าที่เหมาะสม w^0 ซึ่งเราสามารถเขียนสมมุติได้ดังนี้

$$v'_i(n) \longrightarrow 0 \quad ; i=0,1,\dots,L-1$$

และจากความสัมพันธ์ใน LMS Algorithm ได้ว่า

$$|1 - \mu \lambda| < 1$$

$$0 < \mu < 2 / \lambda_i$$

จากอัลกอริทึมของ LMS อัตราการลู่เข้าจะเปลี่ยนจาก $w(0)$ ไปยัง w^0

ดังนั้นเราจะได้ว่า

$$0 < \mu < 2 / \lambda_{\max}$$

เมื่อ λ_{\max} เป็นค่าไอเก้นที่มากที่สุดของสัญญาณอินพุตที่กระทำ Autocorrelation matrix R โดยแท้จริงแล้วอัตราการลู่เข้าที่เร็วที่สุด (fastest convergence) ของโหมดครั้งที่ l จะเกิดขึ้นได้สำหรับค่าวิกฤตของการแกมมา (critical damping) เป็น

$$\mu = 1 / \lambda_l$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คั้งนั้นอัตราการเรียนรู้ที่เร็วที่สุดของ **โหมดเด่น (Dominant Mode)** จะมีคั้งนี้

$$\mu = 1/\lambda_{\max}$$

คั้งนั้นข้อบ้งคั้งเพื่อให้เกิดความมีเสถียรภาพ นั้นไม่เหมาะที่จะนำมาใช้เนื่องจากการคำนวณเพื่อหาค่าของ λ_{\max} จะเป็นสิ่งที่ทำได้ยากมากเมื่อค่า L มีค่ามากคั้งนั้นในทางปฏิบัติการออกแบบหาค่าเพื่อให้มีค่าที่เหมาะสมกับค่าของ λ_{\max} โดยการ ใช้ simple method

$$t_r[R] = Lrxx(0) = \sum_{l=0}^{L-1} \mathcal{M}$$

$\text{tr}[R]$ แสดง Trace of Matrix R

$$\lambda_{\max} \leq \sum_{l=0}^{L-1} \mathcal{M} = Lrxx(0) = LPx$$

เมื่อ $Px \equiv rxx(0) = E[x^2(n)]$

ถ้าแสดงอยู่ในเทอมของกำลังงานเราสามารถเขียนใหม่ได้ว่า

$$0 < \mu < 2/LPx \quad (2.3.2)$$

จากสมการข้างบนเราสามารถหาคุณสมบัติในการเลือกใช้ค่าของ μ ได้คั้งนี้

1. ในข้อจำกัดค่า μ ด้านบน จะเป็นส่วนกลับกับอัตราส่วนของ L ถ้า L มีค้าน้อยแสดงว่าค่าของ μ ที่ถูกใช้สำหรับวงจรกรองที่มีค่าคั้งคั้งที่มากค้ว (Larg-Order filter)
2. เมื่อค่าของ μ เป็นส่วนกลับกับค่าของกำลังงานของสัญญาณอินพุทที่เข้ามา ถ้าสัญญาณที่เข้ามา มีระดับของกำลังงานที่ต่ำ ก็จะทำให้ได้ค่า μ ที่สูงค้ว ในทางกลับกันถ้าสัญญาณที่เข้ามา มีระดับของกำลังงานที่สูงก็จะทำให้ได้ค่า μ ต่ำลง ประโยชน์ก็เพื่อที่จะทำการนอร์มอลไลซ์ค่า μ ค้วอัตราส่วนของกำลังของสัญญาณอินพุท Px ซึ่งเราเรียกอัลกอริทึมใหม่นี้ว่า normalized LMS Algorithm
3. ในทางปฏิบัติจะนิยมใช้ค่า

$$0.01/LPx < \mu < 0.1/LPx$$

2.4 NORMALIZED LMS ALGORITHM

เนื่องจากการคำนวณเสถียรภาพ(stability) เวลาในการลู่เข้า(convergence time), และการแกว่งไกวของ LMS adaptation process จะมีความเกี่ยวข้องกับค่าของ step size μ และค่าของกำลังงานของสัญญาณที่ใช้อ้างอิง ตามสมการที่(2.3.2) ซึ่งค่าสูงสุดของของ μ ที่ยังคงมีความเป็นเสถียรภาพนั้นจะเป็นอัตราส่วนกลับกับค่าของกำลังงานของสัญญาณอ้างอิงและค่า L (filter order L) เทคนิคอันหนึ่งที่มีความสำคัญต่อค่าของความเร็วในการลู่เข้าโดยใช้คุณสมบัติของการออกแบบในสภาวะที่คงค้ว(steady-state performance) ซึ่งแยกออกมาต่างหากกับค่าของกำลังงานของสัญญาณอ้างอิง ซึ่งเป็นของ normalized LMS Algorithm ซึ่งสามารถเขียนเป็นสมการ ได้คั้งนี้

$$w(n+1) = w(n) + \mu(n)x(n)e(n)$$

โดยที่ค่า $\mu(n)$ คือ Adaptive step-size ซึ่งสามารถคำนวณได้จาก

$$\mu(n) = \alpha / L P(n)$$

เมื่อ $P(n)$ เป็นค่ากำลังงานที่เหมาะสมของสัญญาณ $x(n)$ ที่เวลา n และค่า α เป็นค่าของ step-size ที่ถูกนอร์มอลไลซ์แล้วซึ่งจะมีค่าวิกฤตดังนี้

$$0 < \alpha < 2$$

การหาค่ากำลังงานที่เหมาะสมของสัญญาณจะมีอยู่ 2 วิธีคือ

1. ใช้ Rectangular Moving Window Technique ซึ่งสามารถเขียนเป็นสมการได้ดังนี้

$$\begin{aligned} P_x(n) &= \frac{1}{M} \sum_{m=0}^{M-1} x^2(n-m) \\ &= P_x(n-1) + \frac{x^2(n) - x^2(n-M)}{M} \end{aligned}$$

เมื่อ M คือความยาวของ Moving Window

ผลลัพธ์ที่ได้จะมีค่ากำลังงานที่เหมาะสมมากแต่อย่างไรก็ตามจำเป็นจะต้องเก็บค่า $M+1$ ไว้

2. การใช้เอ็กโพเนนเชียลวินโดว์ (exponential window or first-order IIR Filter) ซึ่งต้องการเก็บค่าไว้เพียงตัวเดียวเท่านั้น

$$P_x(n) = (1-\beta) P_x(n-1) + \beta x^2(n)$$

ซึ่งค่า β เป็นค่าสัมประสิทธิ์เชิงเรขาคณิตซึ่งมีค่าอยู่ในช่วง 0 ถึง 1 ซึ่งจะมีความสัมพันธ์กับค่าของความยาวของวินโดว์เป็น

$$M \equiv 1/\beta$$

ซึ่งโดยวิธีการนี้จะให้ค่าที่หยากกว่าการใช้วิธีแรก

และถ้าความยาวของวินโดว์ในวิธีแรกค่าเท่ากับจำนวนของสัมประสิทธิ์ของตัวอะแด็ปทีฟซึ่งมีค่าเป็น L แล้ว

$$P_x(n) = \frac{x^T(n)x(n)}{M}$$

และจาก

$$\mu(n) = \frac{\alpha}{L P_x(n)}$$

ทำให้สามารถลดได้เป็น

$$\mu(n) = \frac{\alpha}{x^T(n)x(n)}$$

โดยทั่วไปแล้ว สำหรับทั้งสองวิธีที่ได้เสนอมาข้างบนนั้นค่า Smoother power estimate จะเกิดขึ้นได้เมื่อค่าความยาวของวินโดว์มีความยาวมากๆ (M is long) แต่ตัวฟิลเตอร์จะไม่สามารถที่จะตอบสนองต่อการเปลี่ยนแปลงกำลังอย่างทันทีทันใด ดังนั้นถ้าสัญญาณอินพุตเป็น Stationary จะใช้ M ที่มีความยาวมาก ในขณะที่ถ้าสัญญาณอินพุตเป็น nonstationary ค่าของ M คำน้อยๆจะถูกนำมาใช้เพื่อการติดตามหาค่ากำลังงานที่เปลี่ยนแปลง

เทคนิคที่เป็นประโยชน์ในการใช้จะมีดังนี้

1. เลือก $P_x(0)$ ค่าที่ดีที่สุดของวิธีการประมาณกำลังของสัญญาณอินพุต
2. เมื่อไม่ต้องการให้ค่าของกำลังสัญญาณ $P_x(n)$ มีค่าเป็นศูนย์ หรือมีค่าน้อยมากๆ ดังนั้นจะต้องใช้โปรแกรมเข้ามาควบคุมเพื่อให้ค่าดังนี้

$$\mu(n) = \frac{\alpha}{L_{\max} [P_x(n), P_{\min}]}$$

เมื่อ P_{\min} เป็นสถานะที่ต่ำของกำลังงาน ซึ่งสมการนี้ทำให้แน่ใจได้ว่า $\mu(n)$ จะต้องมีขอบเขตเสมอ ถ้า $P_x(n)$ มีค่าน้อยมากๆ เมื่อสัญญาณเกิดการขาดหายไปเป็นเวลานาน

ในแผนงานของ feed-forward ซึ่งมันจะเป็นผลที่จะช่วยในการพิจารณาบางอย่าง เช่น ในระบบโดยส่วนมากแล้วจะต้องสามารถที่จะให้ความเหมาะสมในการเปลี่ยนแปลงค่าได้ตามค่าเวลาที่เปลี่ยนแปลงไป ดังนั้นค่าของ step size จะต้องทำให้มีอัตราส่วนที่เหมาะสมกับค่าของการเปลี่ยนแปลงมาตรฐาน (Standard Deviation) หรือค่าของอัตราการขยายที่ใช้คู่กับสัญญาณอินพุตซึ่งสามารถที่จะทำได้โดยใช้อัตราส่วนกลับของค่าการเปลี่ยนแปลงมาตรฐาน

โดยพื้นฐานของการดูเข้าเพื่อที่จะสมมติว่าค่าของ Variance ที่เป็นอัตราส่วนกับค่าของกำลังงานของสัญญาณในช่วงเวลานั้นๆ ซึ่งเป็นเอาต์พุตของการกรองความถี่ต่ำ ซึ่งป้อนอินพุตเป็น $x^2(n)$ ซึ่งจะมีสมการดังนี้

$$\sigma^2(n) = \sum_{m=-\infty}^{\infty} x^2(m)h(n-m) \quad (2.4.1)$$

เมื่อ $h(n)$ เป็นผลการตอบสนองทางความถี่ของการกรองความถี่ต่ำ เมื่อสัญญาณอินพุตมีสถานะคงตัวแล้วเราสามารถที่จะแสดงได้ดังข้างขึ้นถึงค่าความคาดหวังของกำลังงานของสัญญาณ $\sigma^2(n)$ ซึ่งเป็นอัตราส่วนโดยตรงกับค่า Variance $\sigma^2 x(n)$

$$\begin{aligned} \text{เช่น } h(n) &= \alpha^{n-1} & n \geq 1 \\ &= 0 & \text{otherwise} \end{aligned}$$

โดยการใช้สมการที่ (2.4.1) เราจะได้ว่า

$$\sigma^2(n) = \sum_{m=-\infty}^{n-1} x^2(m)\alpha^{n-m-1} \quad (2.4.2)$$

และจากสมการที่(2.7.2) เราจะได้ว่า

$$\sigma^2(n) = \alpha\sigma^2(n-1) + x^2(n-1) \quad (2.4.3)$$

เพื่อให้เกิดเสถียรภาพเราจะต้องทำให้ค่า $0 < \alpha < 1$

การเลือกค่าพารามิเตอร์ของ α เพื่อที่จะควบคุมค่า Effective interval ที่จะนำไปสู่การให้ค่าของ Variance ที่เหมาะสม ดังรูปที่ 2.4.1 เป็นค่าของการจัดระดับของสัญญาณในระบบ DPCM

จากรูปที่ 2.4.1 (a) แสดงให้เห็นถึงค่าการเปลี่ยนแปลงมาตรฐาน (Standard Deviation) ที่เหมาะสมกับรูปสัญญาณซึ่งใน CASE นี้ค่า $\alpha = 0.99$

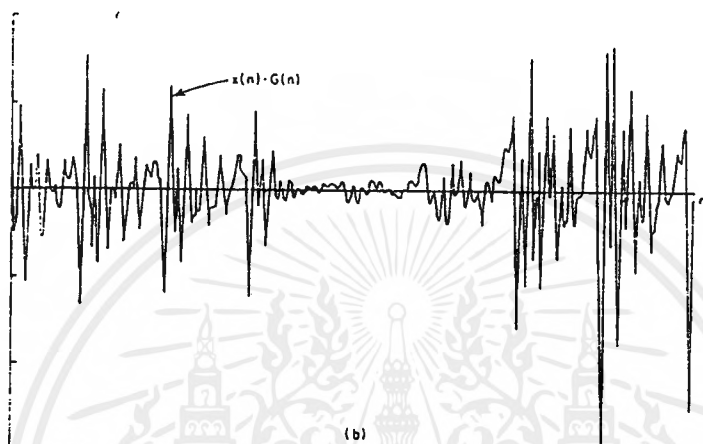
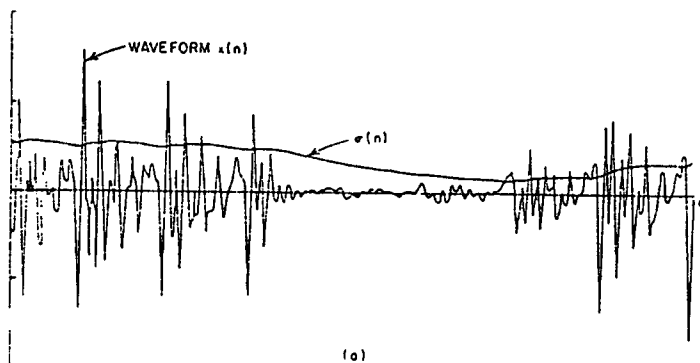
ในรูปที่ 2.4.1(b) จะแสดงค่าของผลคูณของ $y(n) = x(n)G(n)$ ซึ่งการเลือกค่าของ α จะลดขนาดของสัญญาณ $x(n)$

ในรูปแรกด้วยค่า $\alpha = 0.99$ ช่วงเวลาที่คงที่ Time Constant (เวลาที่ถูกหน่วง ไปของสัญญาณ การเวทคิงมีค่า e^{-1} หรือประมาณ 100 ตัวอย่างข้อมูลเมื่อทำการสุ่มด้วยความถี่ 8 kHz

และในรูปที่ 2.4.2 ค่า $\alpha = 0.9$ ซึ่งค่า Time Constant มีค่าเพียง 9 ตัวอย่างเท่านั้น หรือประมาณ 1 mSec ที่ 8 kHz ดังนั้นจึงต้องคำนึงถึงความเหมาะสมในการจัดระบบให้มีความถูกต้อง โดยระบบที่ใช้ค่า $\alpha = 0.99$ เป็นส่วนของระบบที่เกี่ยวข้องกับพยางค์ และระบบที่ใช้ค่า $\alpha = 0.9$ จะเป็นระบบที่มีการเปลี่ยนแปลงอย่างรวดเร็ว

พิจารณาให้เห็น ได้อย่างเด่นชัดดังแสดงในรูป 2.4.1(a) และ 2.4.2(a) ซึ่งมีค่ามาตรฐานของการเปลี่ยนแปลงที่เหมาะสมเหมือนกันทั้งที่ขั้นที่มีการเปลี่ยนแปลงอย่างช้าๆ เมื่อเปรียบเทียบกับสัญญาณเสียงต้นกำเนิด โดยอัตราการขยายหรือค่า Step Size เพื่อควบคุมสัญญาณ ซึ่งจะถูกกำหนด โดยค่าของ Bandwidth ของการกรองความถี่ต่ำ

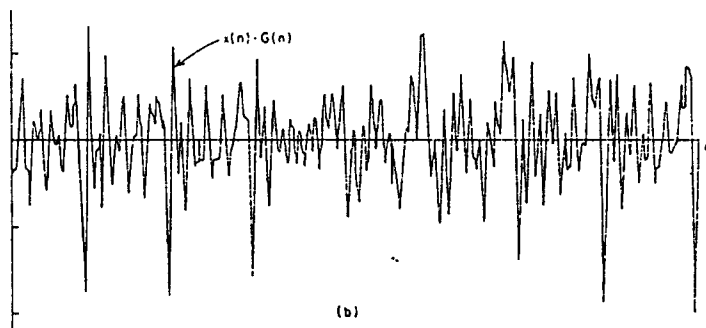
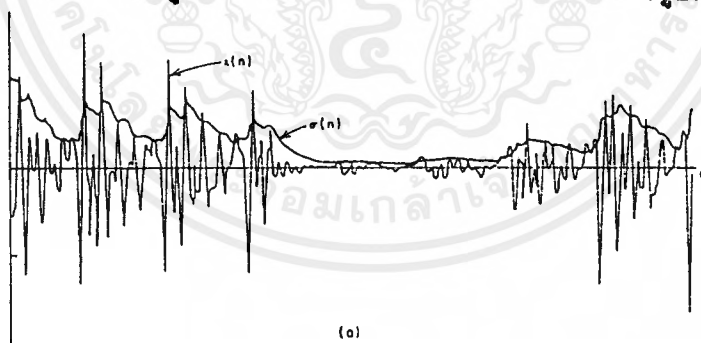
และเนื่องจากค่าของ SNR มีความสำคัญดังนั้นจึงสามารถที่จะปรับปรุงได้โดยผลการศึกษาของ Noll ซึ่งเขาได้พิจารณา ดังนี้



รูปที่ 2.4.1 ตัวอย่างของค่าการเปลี่ยนแปลงมาตรฐานโดยใช้สมการ (2.7.1)

(a) รูปสัญญาณ $x(n)$ และค่าการเปลี่ยนแปลงมาตรฐาน $\sigma(n)$ ที่ $\alpha = 0.99$

(b) จะแสดงค่าของผลคูณของ อัตราขยายที่เปลี่ยนแปลงตามเวลาและรูปสัญญาณ



รูปที่ 2.4.2 ค่า Variance ที่คำนวณได้จากสมการ (2.7.3); (a) $x(n)$ และ $\sigma(n)$ ที่ $\alpha = 0.9$; (b) $x(n) - G(n)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในระบบ Feed-forward ซึ่งมีค่า variance ที่เหมาะสมเป็น

$$\sigma^2(n) = \frac{1}{M} \sum_{m=n}^{n+M-1} x^2(m)$$

- ในระบบ Feedback

$$\sigma^2(n) = \frac{1}{M} \sum_{m=n-M}^{n-1} x^2(m)$$

2.5 การวิเคราะห์ภายในช่วงเวลาสั้นๆ (Short-Time)

การวิเคราะห์ในช่วงเวลาสั้นๆ ในกระบวนการทางค่านิจิตอลที่เกี่ยวข้องกับสัญญาณเสียง ซึ่งโดยทั่วไปแล้ว สัญญาณที่จะนำมาวิเคราะห์นั้นควรมีลักษณะที่ไม่มีการเปลี่ยนแปลงไปกับเวลา (Time-Invariant) และแม้ว่าสัญญาณเสียงจะเป็นสัญญาณที่มีการเปลี่ยนแปลงไปตามเวลา (Time Varying) และเนื่องมาจาก สัญญาณเสียงจะมีความสัมพันธ์กับการเปลี่ยนแปลงที่เป็นไปอย่างช้าๆ เมื่อเราวิเคราะห์สัญญาณเสียงเพียง ช่วงเวลาสั้นๆ นั้นซึ่งมีความสัมพันธ์ที่แยกอิสระออกจากกันได้และมีลักษณะที่ไม่มีการเปลี่ยนแปลงทาง คำนวณได้

การวัดค่าต่างๆ ในช่วงเวลาสั้นๆ

ฟังก์ชันในทอมที่จะวิเคราะห์หาค่าต่างๆ ในช่วงเวลาสั้นๆ นั้นและเนื่องจากรูปแบบของสัญญาณ เสียงที่เราจะนำมาวิเคราะห์นั้นจะมีช่วงของสัญญาณที่มีขอบเขตอยู่อย่างจำกัด ซึ่งค่ามาตรฐานที่ใช้วัด สัญญาณก็คือ

- ค่า แอมป์ริจูดขนาดสูงที่สุด

ซึ่งเป็นขนาดสัมบูรณ์ของสัญญาณที่เกิดขึ้น ในช่วงเฟรมที่จะทำการวิเคราะห์และจะต้องอาศัยการ แยกช่วงที่มีสัญญาณและช่วงที่ไม่มีสัญญาณ (Voice and Unvoice) หรืออาศัยวิธีการนี้เพื่อใช้ สำหรับการนอร์มัลไลซ์ในกระบวนการของ Fixe Point Processing

-กำลังงาน

กำลังงานในช่วงเวลาสั้นๆ นั้น จะหมายถึงการรวมเอาค่าของสัญญาณที่นำมาชกกำลังสองให้ ครอบคลุมช่วงของเฟรมที่จะใช้ในการนำมาวิเคราะห์ (จะมี N ตัวอย่าง และสิ้นสุดที่ตัวอย่างที่ n)

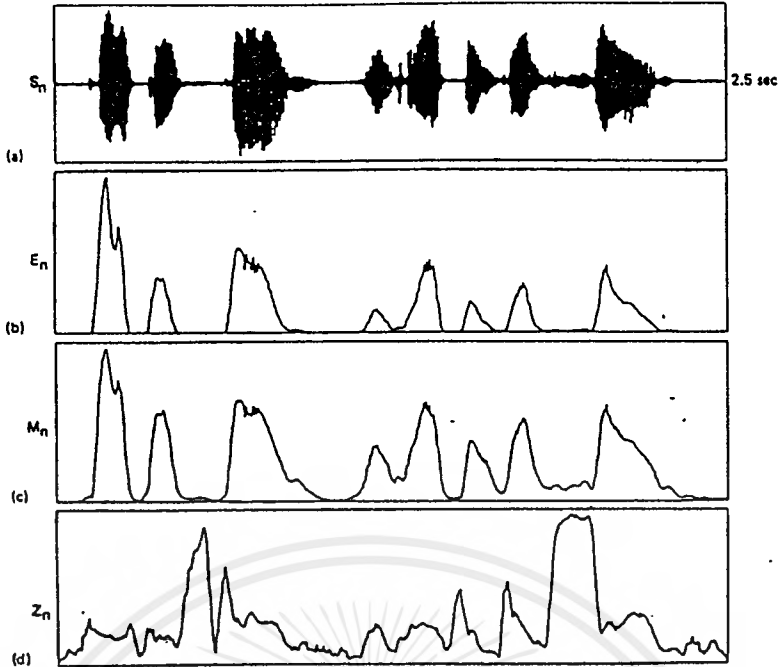
$$E_n = \sum_{m=n-N+1}^n s^2(m)$$

ฟังก์ชันที่แสดงในรูปที่ 2.5.1 (b) สำหรับสัญญาณเสียงในรูปที่ (a) ในบางครั้งเราจะเขียนรูปแบบ ทั่วๆ ไป โดยอาศัยการคูณของ Window function ซึ่งสามารถเขียนใหม่ให้อยู่ในรูปทั่วไปดังนี้

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

$$= 0 \quad \text{otherwise}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5.1 แสดงการวัดค่าต่างๆในช่วงเวลาสั้นๆ

เมื่อ $w(n)$ ในที่นี้จะเป็นที่ฟังก์ชันรูปสี่เหลี่ยม (rectangular window) ซึ่งมีความยาวเป็น N และมีขนาดของสัญญาณเป็น 1 (ขนาดของสัญญาณอาจจะถูกนอร์มอลไลซ์ได้โดยการให้มีความเป็น $\frac{1}{N}$)

ถ้าค่าของ N มีค่าสั้นมากๆ แล้วฟังก์ชันของกำลังงานจะมีการเปลี่ยนแปลงให้เห็นได้อย่างรวดเร็วและชัดเจน ซึ่งเป็นผลอันเนื่องมาจากการแบ่งช่วงของ pitch ของสัญญาณเสียง ในขณะที่ถ้าค่าของ N มีค่ายาว แล้วการเปลี่ยนแปลงของกำลังงานนั้นจะมีลักษณะที่มีความราบเรียบมากขึ้น อย่างไรก็ตามถ้า N มีค่ามากเกินไปแล้วฟังก์ชันของกำลังงานก็จะมีค่าที่มีการเปลี่ยนแปลงน้อยมากๆ และจะไม่มีผลเมื่อมีการเปลี่ยนแปลงทางด้านคุณสมบัติของสัญญาณเสียง ดังที่แสดงในรูปที่ 2.5.1 ซึ่งเป็นการคำนวณด้วยค่าของ window ขนาด 30 msec และจะถูกเลื่อนออกไป 5 msec

- ค่าเฉลี่ยของขนาดของสัญญาณ

ซึ่งจะมีลักษณะที่คล้ายกับฟังก์ชันของกำลังงานของสัญญาณแต่จะแตกต่างกันตรงจะต้องที่ไม่มีกำลังสอง (อันสืบเนื่องมาจากการที่ฟังก์ชันของกำลังงานของสัญญาณจะให้ค่าที่มีความไวต่อขนาดของสัญญาณที่มีค่ามากๆ)

$$M_n = \sum_{m=-\infty}^{\infty} |s(m)w(n-m)|$$

โดยที่ทั้งฟังก์ชันของกำลังงานและของค่าเฉลี่ยของขนาดของสัญญาณ (ดูในรูปที่ 2.8.1(c)) ซึ่งจะมีประโยชน์ในการตัดสินใจช่วงของสัญญาณที่มีสัญญาณเสียงหรือไม่มีสัญญาณเสียง (Voice/Unvoiced Decision)

-Zero Crossing rate (อัตราการตัดผ่านศูนย์)

ซึ่งสามารถที่จะวัดได้โดยการตรวจนับการเปลี่ยนเครื่องหมายที่เกิดขึ้นในเฟรมที่ต้องการวิเคราะห์

$$Z_n = \sum_{m=-\infty}^{\infty} [(\text{sign changes})w(n-m)]$$

และอัตราการตัดผ่านศูนย์นั้นจะเป็นตัวบ่งชี้ถึงความบ่อยของการเกิดการเปลี่ยนแปลงเช่นในช่วงที่เกิดการเปลี่ยนแปลงการตัดผ่านศูนย์ด้วยอัตราที่สูงแล้วจะแสดงว่าเป็นช่วงการการที่ไม่มีเกิดสัญญาณเสียง(ไม่มีการเปล่งเสียง) ซึ่งจะให้อัตราของ Zero Crossing ที่สูง ซึ่งสามารถดูได้ดังในรูปที่ 2.8.1(d) ซึ่งจะนำมาใช้ในการตรวจนับการเกิด Voiced/Unvoiced/Silence (ช่วงเวลาที่เกิดการเงียบ)

2.6 ความมีเสถียรภาพ, การลู่เข้า และสถานะคงที่ของค่าเฉลี่ยกำลังสองของค่าความผิดพลาด ใน FDAF (Stability, Convergence and Steady-state Mean-Square Error of the FADF)

1. การลู่เข้าของค่าเฉลี่ย (Convergence in the mean)

เมื่อพิจารณา $x(n), d(n)$ ให้เป็นค่าที่มีสถานะคงที่ (Stationary) เราจะได้สถานะของความมีเสถียรภาพและการลู่เข้าของค่าเฉลี่ย ซึ่งโดยปกติแล้วเราจะให้ค่าสัมประสิทธิ์ของการฟิลเตอร์ที่ค่าเริ่มต้นให้เป็นศูนย์

$$F_0(k) = 0; k = 0, 1, \dots, L-1$$

เมื่อเอาท์พุทในแต่ละ Frequency bin สามารถคำนวณได้ซึ่งมีค่าเป็น

$$Y_i(k) = F_i(k) X_i(k); k = 0, 1, \dots, L-1$$

เมื่อ $F_i(k)$ คือตัวที่ i ที่มีการเปลี่ยนแปลง(ซึ่งมีความสัมพันธ์กับ บล็อกตัวที่ i ของข้อมูล) ค่าสัมประสิทธิ์ของการฟิลเตอร์และสอดคล้องกับคอมโพเนนต์ทางความถี่ตัวที่ k และค่าความผิดพลาดของคอมโพเนนต์ทางความถี่ตัวที่ k และเราเขียนได้เป็น

$$E_i(k) = D_i(k) - Y_i(k); k = 0, 1, \dots, L-1$$

สำหรับค่าของสัมประสิทธิ์ของการฟิลเตอร์เราสามารถแทนได้ตามอัลกอริทึมของ LMS คือ

$$F_{i+1}(k) = F_i(k) + \alpha E_i(k) X_i^*(k)$$

เมื่อแทนค่า $E_i(k)$ จะได้เป็น

$$\begin{aligned} F_{i+1}(k) &= F_i(k) + \alpha \{D_i(k) - Y_i(k)\} X_i^*(k) \\ &= \{1 - \alpha |X_i^*(k)|^2\} F_i(k) + \alpha D_i(k) X_i^*(k) \end{aligned} \quad (2.6.1)$$

$$E\{F_{i+1}(k)\} = \{1 - \alpha R(k)\} E\{F_i(k)\} + \alpha G(k)$$

$$\text{เมื่อ } R(k) = E\{|X_i^*(k)|^2\} \text{ และ } G(k) = E\{D_i(k) X_i^*(k)\}$$

และเมื่อค่า R และ G ถูกกำหนดบน Frequency Bin ที่ k ซึ่งเป็นผลที่เกิดมาจากหลังของสัญญาณ $x(n)$ และ $d(n)$ เมื่อใช้ Z-Transform สมการ (2.6.1)

$$F^{(k)}(z) = \{1 - \alpha R(k)\} z^{-1} F^{(k)}(z) + \frac{\alpha G(k)}{(1 - z^{-1})}$$

$$\text{เมื่อ } F^{(k)}(z) = \sum E\{F_i^{(k)}(z)\}$$

ทำให้เราได้ค่า Steady-state mean value $F(k)$ โดยการใช้ทฤษฎีการหาค่าสุดท้ายของ Z-Transform คือ

$$\begin{aligned} E\{F_\infty(k)\} &= \lim_{z \rightarrow 1} \{(z-1)F^{(k)}(z)\} \\ &= \frac{G(k)}{R(k)} = \frac{E\{D_i(k) X_i^*(k)\}}{E\{|X_i^*(k)|^2\}} \\ &= \frac{E\{D_i(k) X_i^*(k)\}}{E\{|X_i^*(k)|^2\}} \end{aligned}$$

จะเห็นได้ว่าค่าของสัมประสิทธิ์ในสภาวะ steady state ของอัลกอริทึมทางด้านความถี่จะคล้ายคลึงกับคำตอบของสมการปกติ(normal equation solution) สำหรับ Least-squares filter ในทางแกนเวลาถึงแม้ว่าจะไม่เหมือนกันหมดเสียทีเดียว เราสามารถได้สภาวะสำหรับการเข้าสู่ซึ่งค่านี้ได้ถ้าเรากำหนดให้ค่าเฉลี่ยของสัมประสิทธิ์ของการฟิลเตอร์ที่เป็นความแตกต่างให้เป็น $U_i(k)$

$$U_i(k) = E\{f_i(k)\} - E\{F_\infty(k)\}$$

จะได้

$$U_{i+1}(k) = \{1 - \alpha R(k)\} U_i(k)$$

สุดท้ายเนื่องจากมีความสัมพันธ์

$$\lim_{i \rightarrow \infty} U_i(k) \rightarrow 0 \quad ; \quad k=0,1, \dots, L-1$$

ซึ่งทำให้ได้

$$|1 - \alpha R(k)| < 1$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$-1 < 1 - \alpha R(k) < 1$$

$$-2 < -\alpha R(k) < 0$$

$$0 < \alpha < \frac{2}{R(k)}$$

หรือ

$$0 < \alpha < \frac{2}{E\{|X_i(k)|^2\}}$$

โดยปกติสภาวะของการลู่เข้าและความมีเสถียรภาพ จะขึ้นอยู่กับกำลังของสัญญาณอินพุท

b.) ความไม่สม่ำเสมอของการลู่เข้า (Non-Uniform Convergence)

จากสมการข้างบนแสดงให้เห็นว่าอัตราของการลู่เข้าจะขึ้นอยู่กับตัวแปรที่เป็นกำลังของสัญญาณในแต่ละ Frequency band ในรูปของเวกเตอร์ $F_{i+1}(k) = F_i(k) + \alpha E_i(k) X_i^*(k)$ ซึ่งอัตราของการลู่เข้าจะขึ้นอยู่กับค่าไอเก็นซของเมทริกซ์

$$R = \text{diag} \{R(0), R(1), \dots, R(L-1)\}$$

กระบวนการของการลู่เข้าสามารถถูกแทนได้ด้วยค่าช่วงเวลาคงตัว (Time Constant) ซึ่งเราจะกำหนดค่า time constant สำหรับการเปลี่ยนแปลงค่าความแตกต่างของสัมประสิทธิ์ที่เวลาที่ต้องการสำหรับส่วนประกอบนั้น (Component) กับค่าการลดลงด้วยค่า $1/e$ ซึ่งเราสามารถประมาณค่าคงตัวของเวลาได้เป็น

$$t_j \approx \frac{1}{\alpha \lambda_j}$$

$$\text{เมื่อ } \lambda_j = R(j) = E\{|X_i(k)|^2\}$$

ดังนั้นในโหมคทางด้านความถี่อัตราการลู่เข้าจะเป็น

$$t_{\text{blk}} \approx \frac{1}{\alpha E\{|X_i(k)|^2\}}$$

นั่นคือจะมีการเปลี่ยนแปลง 1 ครั้งต่อ 1 บล็อก (once - per block update) อย่างไรก็ตามจะมีสัญญาณอยู่ L ตัวอย่างต่อการเปลี่ยนแปลง ดังนั้น

$$t_k \approx \frac{L}{\alpha E\{|X_i(k)|^2\}}$$

และเมื่อพิจารณากำลังงานซึ่งจะมีความแตกต่างกันระหว่างช่วงแบนด์ความถี่ (frequency band) และในอัลกอริทึมทางด้านความถี่เป็นปัญหากับความไม่เหมือนกันของผลกระทบจากการลู่เข้าซึ่งจะคล้ายคลึงกับปัญหาของการแพร่กระจายของค่าไอเก็นซทางด้านแกนเวลา แต่อย่างไรก็ตามอัลกอริทึมทางด้านแกนความถี่เราจะทำให้ง่ายขึ้น โดยการนอร์มอลไลซ์ค่าการลู่เข้าในแต่ละแบนด์ โดยการปรับปรุงดังนี้

$$F_{i+1}(k) = F_i(k) + \frac{\alpha E_i(k) X_i^*(k)}{E\{|X_i(k)|^2\}}$$

หรือในทางปฏิบัติเราจะใช้ค่าการประมาณของ $E\{|X_i(k)|^2\}$ ทำให้ได้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$F_{i+1}(k) = F_i(k) + \frac{\alpha E_i(k) X_i^*(k)}{\frac{1}{N} \sum_{j=I-N+1}^I |X_j(k)|^2}$$

C) ค่าเฉลี่ยของสัญญาณความแตกต่างกำลังสอง

โดยการใช้สมมติฐานอิสระในการพิจารณาทำให้เราสามารถประมาณหาค่าความสัณฐานได้

Excess mean-squared error ได้เป็น

$$\text{Excess mse} = \frac{\alpha L}{2} \text{ Power of the input}$$

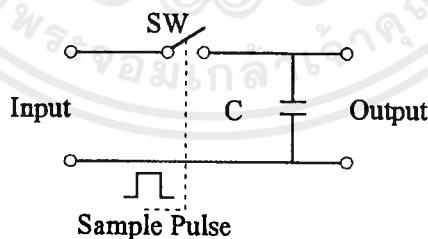
2.7 วงจรสุ่มตัวอย่างและการคงค่าสัญญาณ (Sample And Hold)

เป็นอุปกรณ์ที่ใช้ในการจัดการกับสัญญาณอินพุต, เอาท์พุตและการควบคุมระบบทางอินพุต ซึ่งจะมีการทำงานอยู่ 2 โหมดด้วยกัน คือ

- โหมดการสุ่มตัวอย่างหรือการติดตามสัญญาณ (Sample or Track Mode) ในโหมดนี้จะมีการทำงานเมื่อมีสัญญาณทางอินพุตและจะติดตามการเปลี่ยนแปลงเปลี่ยนแปลงของสัญญาณจนกว่าจะอยู่ในสภาวะการควบคุมของการคงค่า (Hold)

- โหมดของการคงค่าระดับสัญญาณ (Hold Mode)

เวลาส่วนใหญ่ที่เกี่ยวข้องจะไปอยู่ที่โหมดของการติดตามสัญญาณ การ Sample-Hold โดยปกติจะมีเกนที่มีค่าเป็น 1 (Unity Gain) และเป็นแบบ นอนอินเวอร์ต ขาอินพุตการควบคุมสามารถใช้ได้กับระบบ TTL ซึ่งโดยทั่วไปลอจิก " 1 " จะเป็นสัญญาณการควบคุมของโหมดการสุ่มตัวอย่าง (Sample) และลอจิก " 0 " จะเป็นสัญญาณการควบคุมของโหมดการคงค่า (Hold)



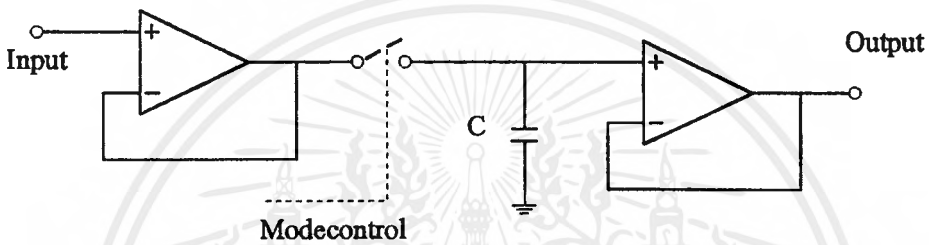
รูปที่ 2.7.1 แสดงพื้นฐานของการ Sample-Hold

วงจรสุ่มค่าสัญญาณและคงค่าไว้เป็นวงจรที่ประยุกต์มาจากวงจรของอะนาล็อกสวิตช์และวงจรคงค่า วงจรนี้จะสุ่มตัวอย่างสัญญาณอนาล็อกที่เข้ามาในรูปของค่าความต่างศักย์และจากนั้นก็จะนำเอาค่าที่สุ่มมาได้มาทำการคงค่าเอาไว้เพื่อให้ไหลออกทางเอาท์พุต ซึ่งการสวิตช์จะถูกควบคุมจากพัลส์สัญญาณควบคุมและเวลาในการประจุนแรงดันถึงค่าที่สุ่มมานั้นจะเรียกว่า อพอร์เจอร์ไทม์ (Aperture Time) เมื่อสวิตช์ปิดวงจรก็จะอยู่ในสภาวะการติดตามข้อมูลตามค่าของแรงดันที่มีการเปลี่ยนแปลง และจะอยู่ในสภาวะของการคงค่าเมื่อสวิตช์เปิดวงจรและจะเก็บค่าของแรงดันของสัญญาณเอาไว้โดยตัวเก็บประจุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถึงซึ่งคาบของการทำงานใหม่ ซึ่งจะขึ้นอยู่กับค่าขนาดของตัวเก็บประจุและค่าความต้านทานที่ก่อให้เกิดการรั่วไหลของสัญญาณ ในทางปฏิบัตินั้นวงจร Sampling And Hold จะต้องมีบัฟเฟอร์แอมพลิฟายเออร์ (Buffer Amplifier) ทั้งทางอินพุตและเอาต์พุต โดยทางด้านเอาต์พุตบัฟเฟอร์แอมพลิฟายเออร์จะต้องให้ผลกระทบต่อกรร่วไหลของสัญญาณน้อยมากๆ ซึ่งโดยส่วนใหญ่วงจรทางด้านอินพุตจะใช้ FET และในส่วนของอิเล็กทรอนิกส์สวิตช์จะต้องมีการรั่วไหลของสัญญาณต่ำด้วย

อุปกรณ์ Sample And Hold นี้มีใช้ เนื่องจากวงจรแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัล (ADC) ไม่สามารถยอมรับการเปลี่ยนแปลงมากๆ ของสัญญาณอนาล็อกที่เข้าไปได้ในขณะที่กำลังทำการแปลงข้อมูล (conversion time) ดังนั้นการแก้ปัญหาจึงต้องใช้วงจรสุ่มและคงค่าไว้ให้คงที่ภายในช่วงเวลาที่กำลังแปลงข้อมูล



รูปที่ 2.7.2 แสดงวงจร Sample And Hold ที่ใกล้เคียงกับวงจรใช้งานจริง

การทำงานภายใต้สถานะของการโฮลด์

ในส่วนของการฟิลเตอร์เราจะเห็น ได้ว่ายังมีอีกหลายๆ วิธีของการสร้างสัญญาณที่มีความต่อเนื่องทางเวลา $x(t)$ จากสัญญาณการสุ่มตัวอย่าง $x_s(t)$ ในที่นี้เราจะพิจารณาของสถานะของการ โฮลด์ซึ่งจะให้เอาต์พุตออกมาเป็น

$$x(t) = x(nT) \quad nT \leq t < nT+T$$

พิจารณาภายใต้สถานะของการ โฮลด์ซึ่งจะยังคงค่าของสัญญาณที่ทำการสุ่มตัวอย่าง ณ เวลา nT จนกระทั่งรับเอาสัญญาณสุ่มตัวอย่างค่าถัดมาที่เวลา $nT+T$ และค่าของเอาต์พุตของอุปกรณ์การ โฮลด์ ซึ่งเป็น PIECEWISE - CONSTANT ของสัญญาณอนาล็อก นั่นคือ $x(t)$ เป็นค่าคงที่ซึ่งในแต่ละช่วง T วินาที ภายใน $nT \leq t < nT+T$ เมื่อขนาดของสัญญาณ $x(t)$ ซึ่งเป็นค่าคงที่ตลอดช่วงเวลา T ในบางครั้งเราจะเรียกอุปกรณ์ที่ทำการโฮลด์นั้นว่า "Zero Order Hold"

ในทางกลับกันอุปกรณ์ของการ โฮลด์จะมีความสัมพันธ์กับการกรองความถี่ต่ำ โดยการแสดงการคำนวณให้เห็นถึงผลการตอบสนองทางความถี่ของอุปกรณ์การ โฮลด์ และโดยการใช้อนุกรมฟูเรียร์ทรานสฟอร์ม $X_s(w)$ ของสัญญาณที่ถูกสุ่มตัวอย่างคือ $x_s(t) = x(t)p(t)$ โดยการใช้นิยามของอนุกรมฟูเรียร์ทรานสฟอร์มซึ่งเราจะได้ว่า

$$\begin{aligned} X_s(\omega) &= \int x_s(t) e^{-j\omega t} dt \\ &= \sum x(t) \delta(t - nT) e^{-j\omega t} dt \\ &= \sum x(nT) e^{-j\omega nT} \end{aligned}$$

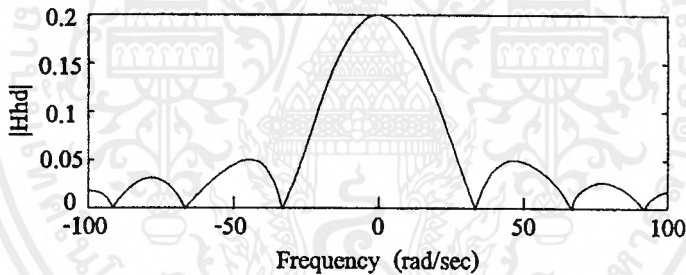
และเมื่อพิจารณาฟูริเยร์ทรานสฟอร์มของเอาต์พุตของสัญญาณที่ได้มาจากอุปกรณ์ของการโฮลซึ่งจะได้ว่า

$$\begin{aligned}
 X_s(\omega) &= \int x'(t)e^{-j\omega t} dt \\
 &= \sum \int x(nT)e^{-j\omega t} dt \\
 &= \sum \left[\int e^{-j\omega t} dt x(nT) \right] \\
 &= \sum \left[\left(\frac{-1}{j\omega} \right) e^{-j\omega t} \right] x(nT) \\
 &= \frac{(1 - e^{-j\omega T})}{j\omega} \sum e^{-j\omega nT} x(nT) \\
 X(\omega) &= \frac{(1 - e^{-j\omega T})}{j\omega} X(\omega)
 \end{aligned}$$

ดังนั้นเราจะได้ทรานสฟอร์มฟังก์ชันของอุปกรณ์การโฮลเป็น

$$H_{hd}(\omega) = \frac{(1 - e^{-j\omega T})}{j\omega}$$

ซึ่งเราสามารถแสดงให้เห็นดังในรูปซึ่งมีค่า $T=0.2$ ดังแสดงในรูป



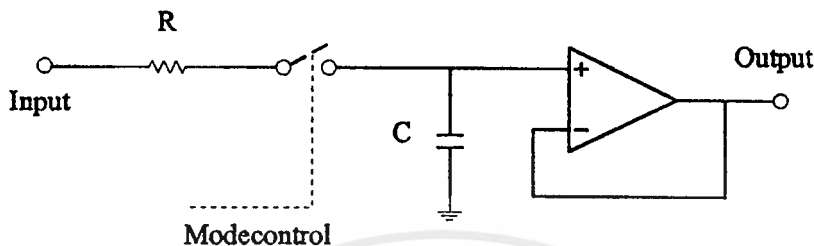
รูปที่ 2.7.3 ขนาดของ ฟังก์ชันการตอบสนองความถี่ของอุปกรณ์การโฮลที่ค่า $T=0.2$

เทคนิคหลายๆ อย่างได้ถูกนำมาใช้ในวงจร Sample And Hold เราสามารถแบ่งได้ออกเป็น 2 ประเภทใหญ่ คือ

1. ระบบเปิด (Open-Loop Circuit) แสดงดังรูปที่ 2.7.3 ซึ่งเป็นวงจรโคซพื้นฐานของเมื่อสวิทช์ปิดวงจร ตัวเก็บประจุก็จะเริ่มชาร์จเก็บประจุในลักษณะของเอ็ทโพเนนเชียลกับสัญญาณอินพุตซึ่งจะเป็นไปตามค่าของ Time Constant ของวงจร และวงจรขยายทางเอาต์พุตจะให้ค่าเอาต์พุตออกมาตามค่าแรงดันที่คคร่อมตัวเก็บประจุ เมื่อสวิทช์เปิดวงจร ตัวเก็บประจุก็ยังคงมีค่าแรงดันค้างอยู่ ซึ่งช่วงเวลาในการเข้าถึงสัญญาณอินพุตของตัวเก็บประจุจะขึ้นอยู่กับค่าของตัวต้านทานที่ต่ออยู่อนุกรมและค่าของกระแสที่ไหลผ่านตัวเก็บประจุ เมื่อมีการเก็บประจุเต็มแล้ว สวิทช์จึงถูกเปิดวงจรออก แต่เมื่อผ่านวงจรขยายทางค่านเอาต์พุตจะทำให้ได้ค่าเอาต์พุตที่มีค่าไม่สม่ำเสมอ และทำให้ช่วงเวลาของการเซตคิ่งมีค่ามากขึ้น โดย

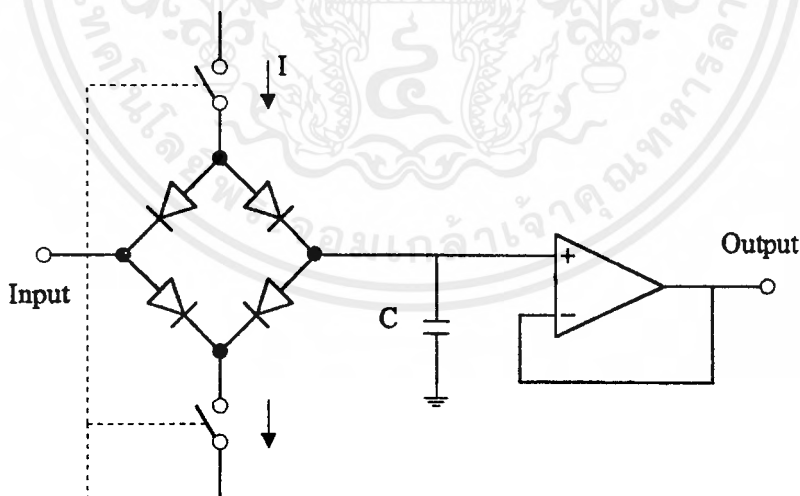
ปราศจากผลกระทบจากโคจรของสวิทช์ทั่วไปซึ่งโคจรทั่วไปจะใช้ FET และวงจรวงศาขยทางค้ำนเอาร์ทุคจะใช้ FET เป็นอินพุท

วงจรมีข้อเสียคือ โทลคของตัวเก็บประจุจะขึ้นอยู่กับอินพุทของอุปกรณ์แหล่งก้ำเนคสัญญาณซึ่งอาจทำให้เกิดการออสซิลเลชันได้หรือทำให้ค่าของกระแสที่ไหลผ่านตัวเก็บประจุได้ไม่เพียงพอซึ่งจะมีผลต่อค่าความเร็วในการเก็บประจุ



รูปที่ 2.7.4 แสดงวงจรถิ่นฐานแบบ Open-Loop

ต่อมาได้มีการปรับปรุงเพื่อแยกผลกระทบระหว่างวงจรถิ่นฐานกับแหล่งก้ำเนค สัญญาณ (Isolate Source) เพื่อให้การซำคประจุทำได้อย่างรวดเร็วที่สภาวะปิดวงจรถิ่นฐานด้วยค่าของ Linear Slew Rate ของวงจรถิ่นฐาน ซึ่งได้แสดงดังรูป 2.7.5 ค่าของกระแสจากแหล่งก้ำเนคจะไหลผ่านสวิทช์ที่ปิดวงจรถิ่นฐานเข้าไปยังตัวเก็บประจุเก็บประจุ ถ้าววงจรถิ่นฐานและกระแสมีความสมดุลกัน (Balance) ค่าของกระแสจะหยุดไหลเมื่อระดับแรงค้ำนของตัวเก็บประจุจะมีค่าเท่ากับค้ำแรงค้ำนทางอินพุทที่เข้ามาซึ่งวงจรมีข้อดีในเรื่องการปรับปรุงค้ำของเวลาการเข้าถึงข้อมูลและเวลาในการเซตติง (Settling Time)

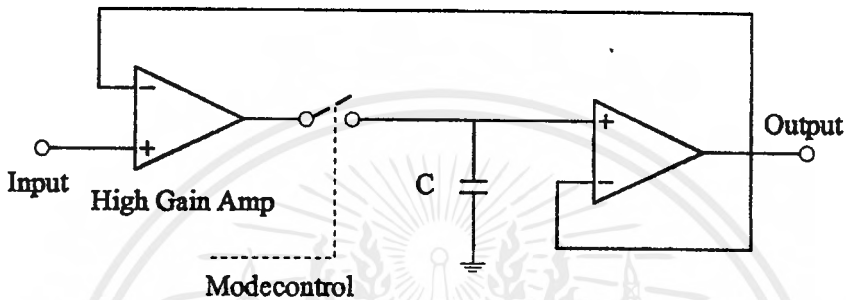


รูปที่ 2.7.5 Current Source Sample-Hold

2. ระบบปิด (Closed-Loop Circuit) วงจรถิ่นฐานแล้วข้างค้ำนยังคงเป็นระบบการควบคุมแบบเปิดอยู่ แต่ถ้ำการติดตามสัญญาณที่ความถี่ค้ำๆ แล้วค้ำความถูกต้องจะเป็นถิ่งที่มีความข้ำเป็นมากกว่าค้ำความเร็ว ดังนั้นจึงอ้ำคัยการค้ำวงจรถิ่นฐานแบบค้ำสแคคกันและมีการบื้อนกลับของสัญญาณเอาร์ทุค โดยจะมีวงจรถิ่นฐานทางอินพุทที่มีเกณช์การขยที่สูงตำหรับใช้ในการติดตามสัญญาณให้มีความถูกต้อง

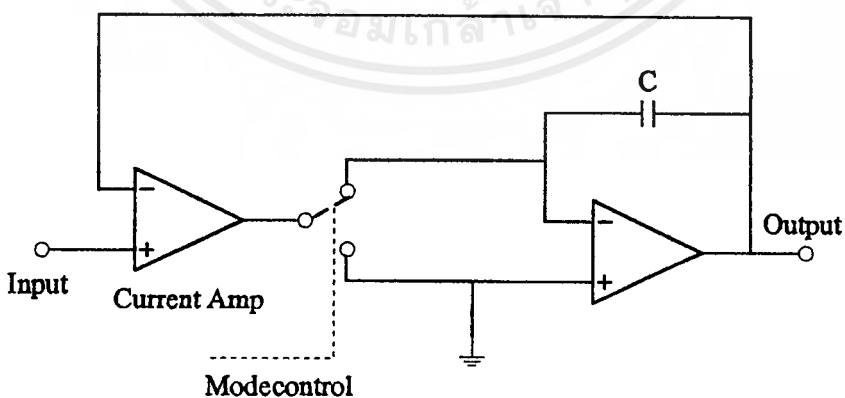
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังแสดงในรูปที่ 2.7.6 เมื่อสวิตช์ปิดวงจร เอาท์พุทจะถูกแสดงด้วยค่าของประจุที่มีในตัวเก็บประจุ และจะถูกนำมาใช้ในการติดตามค่าของสัญญาณอินพุทที่วงจรขยายทางด้านอินพุท ค่าคอมมอนโหมดและค่าออฟเซต (Common Mode and Offset Error) ที่วงจรทางด้านเอาท์พุทจะได้รับการชดเชยโดยค่าประจุในตัวเก็บประจุ และการป้อนกลับจะถูกสวิตช์ไปยังสภาวะของการคงค่า (Hold) ก่อนที่เอาท์พุทจะมีค่าที่แน่นอนเท่ากับค่าทางอินพุทซึ่งทำให้เกิดความผิดพลาดเกิดขึ้นได้ เมื่ออยู่ในสภาวะการคงค่า (Hold) สวิตช์จะเปิดวงจรจะต้องให้ค่าที่ถูกต้องแน่นอนในการกลับสู่สภาวะการ Sample เมื่ออินพุทไม่มีการเปลี่ยนแปลง



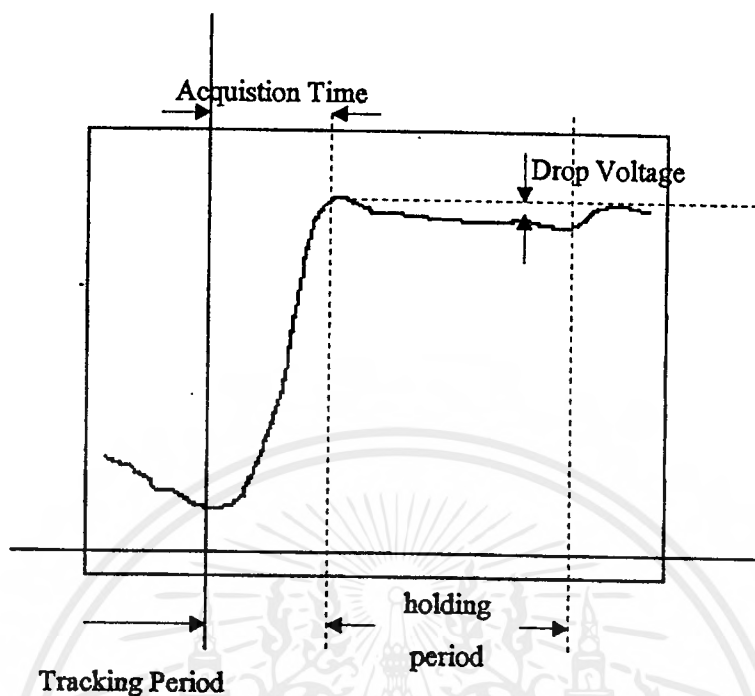
รูปที่ 2.7.6 แสดง Low-Frequency Sample-and-Hold with Feedback

วงจรในรูปที่ 2.7.7 ที่วงจรทางด้านเอาท์พุทจะเป็นวงจรอินทิเกรเตอร์ โดยจะยอมให้มีการสวิตช์ผ่านกราวด์ได้และเพื่อหลีกเลี่ยงปัญหาของค่าการรั่วไหลของกระแสที่เกิดจากอิเล็คตรอนิกส์ สวิตช์วงจรทั้งสองจะมีค่าเวลาในการเข้าถึงข้อมูลและเวลาเซตติงที่เท่ากัน เมื่อการเก็บประจุของตัวต้านทางจะถูกควบคุมจากระดับแรงดันเอาท์พุทเพื่อให้ได้ค่าการสุ่มเหมือนกับสัญญาณอินพุทที่รับเข้ามา



รูปที่ 2.7.7 Low-Frequency Sample-and-Hold with Integrator Feedback

คุณสมบัติของวงจร Sample And Hold



รูปที่ 2.7.8 แสดงช่วงเวลาที่สำคัญของวงจร Sample And Hold

ในทางอุดมคติของการ Sample And Hold การติดตามสัญญาณอินพุตจะต้องปราศจากความผิดพลาด (Error-Free) , การเข้าถึงข้อมูลจะเกิดขึ้นอย่างรวดเร็วโดยไม่มีค่าเวลาในการเซตลิ่ง และช่วงเวลาในการคงค่าจะสามารถรักษาค่าเอาไว้ได้อย่างไม่จำกัดเวลาซึ่งก็คือไม่มีค่าการรั่วไหลของกระแส แต่ในการใช้งานจริงจะมีค่าพารามิเตอร์ที่ทำให้เกิดความแตกต่างจากอุดมคติดังนี้

1. Acquisition Time (ช่วงเวลาการเข้าถึงข้อมูล) เป็นช่วงเวลาที่เกิดขึ้นเมื่อมีการสุ่ม (Sample) ไปจนถึงจุดที่ได้รับสัญญาณเอาต์พุตมีค่าเท่ากับสัญญาณที่ติดตามทางอินพุต
2. Aperture Time (อเพอร์เจอร์ไทม์) เป็นช่วงเวลาระหว่างการควบคุมให้เกิดการคงค่า (Hold) ไปจนถึงจุดที่สวิตช์ปิดวงจรอย่างสมบูรณ์ ซึ่งเราเรียกว่าเวลาการหยุดการทำงาน (Turn-Off Time)
3. Aperture Uncertainty Time เป็นช่วงเวลาที่เกิดขึ้นใน อเพอร์เจอร์ไทม์ (Aperture Time) ซึ่งวัดจากช่วงเวลาที่มีการเปลี่ยนแปลงของสัญญาณเอาต์พุตมากที่สุด ไปจนถึงช่วงที่มีการเปลี่ยนแปลงของสัญญาณเอาต์พุตน้อยที่สุด
4. Decay Rate เป็นอัตราการลดลงของแรงดันทางด้านเอาต์พุตในเวลาที่อยู่ใน โหมดการคงค่า
5. Feedthrough แสดงปริมาณของสัญญาณอินพุตที่ทำให้สัญญาณเอาต์พุตมีค่าเป็นหนึ่ง ในช่วงเวลาของโหมดการคงค่า (Hold) โดยจะมีค่าเปลี่ยนแปลงที่ขึ้นอยู่กับความถี่ของสัญญาณและในบางครั้งจะแทนด้วยอัตราการลดทอนในหน่วยเดซิเบล (dB)
6. Settling Time (เซตลิ่ง ไทม์) เป็นช่วงเวลาที่เกิดขึ้นจากการควบคุมให้มีการเปลี่ยนตำแหน่งจนกระทั่งเอาต์พุตมีค่าคงตัว ภายใต้ช่วงที่ก่อให้เกิดความผิดพลาดที่น้อยที่สุด

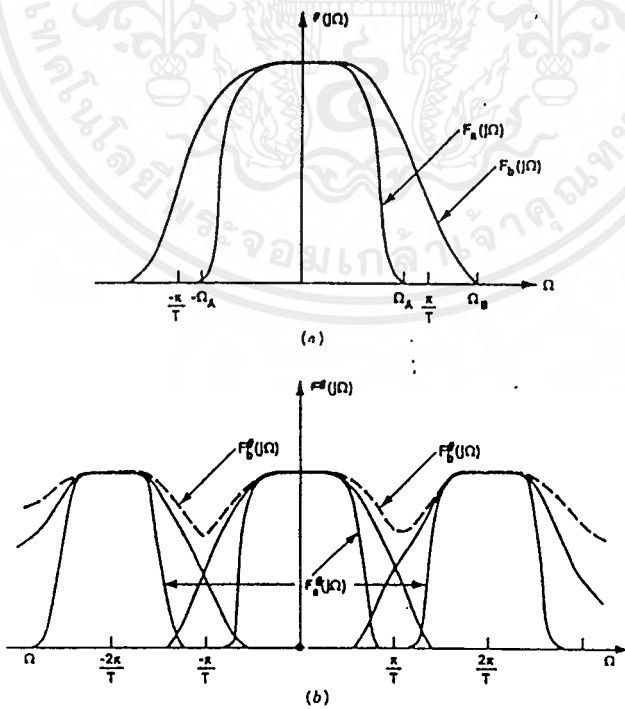
จาก

$$F(j\Omega) = (1/T) \sum F(j\Omega - j2\pi k/T)$$

ดังนั้นสเปกตรัมของ Discrete-time signal เป็นสัญญาณที่เป็นฟังก์ชันของ Ω และจะเกิดขึ้นซ้ำๆกันทุก $2\pi/T$ เทอมสัญญาณทางค่านอะนาล็อกจะมีสเปกตรัมที่ไม่เป็นคาบ

ความต้องการการกรองความถี่ที่ป้องกันการเกิดการเหลื่อมล้ำกันทางด้านสเปกตรัมซึ่งได้แสดงไว้ในสมการข้างต้น ในทางปฏิบัติการพิจารณาสัญญาณที่มีความต่อเนื่องกันทางด้านเวลาซึ่งมีสเปกตรัมเป็น $F(j\Omega)$ ซึ่งแสดงในรูป 10-4 ซึ่งจะมียูนิฟอร์มสเปกตรัมของสัญญาณที่มีแบนด์จำกัด (bandlimited) และมีความถี่อยู่ในช่วง $-\Omega_A$ ถึง Ω_A เมื่อ Ω_A มีค่าน้อยกว่า π/T และจะไม่มีการซ้อนของสัญญาณที่อยู่นอกช่วงความถี่นี้โดยหลักการกรองความถี่ที่มีความเป็นไปเหมือนกับ bandlimiting นั้น ในทางปฏิบัติไม่สามารถที่จะทำได้จริงๆ แต่เราสามารถที่จะใช้ค่าของการประมาณ โดยการใช้มาตรฐานทางด้านเทคนิคซึ่งเป็นอะนาล็อกของวงจรกรองความถี่ต่ำ

เมื่อสัญญาณที่มีความต่อเนื่องทางด้านเวลา ได้ถูกทำการสุ่มตัวอย่างและเปลี่ยนเป็นสัญญาณที่ไม่มีความต่อเนื่องทางด้านเวลา(สัญญาณที่เป็นดิสครีต) ซึ่งจะทำได้สเปกตรัมของสัญญาณจะเกิดขึ้นทุกๆ $2\pi/T$ และถูกแทนด้วย $F_a(j\Omega)$ ซึ่งจะเกิดการเหลื่อมล้ำกันทางด้านสเปกตรัมดังที่ได้แสดงไว้ในรูปที่ 2.7.9 ซึ่งแสดงสเปกตรัมของแต่ละตัวจะมีความสัมพันธ์กันกับสเปกตรัมดั้งเดิมที่เป็นสัญญาณที่มีความต่อเนื่องทางด้านเวลาและความคิดเทียบทางสเปกตรัมของสัญญาณที่เป็นดิสครีตสามารถที่ลดลงได้โดยการใช้วงจรกรองความถี่ต่ำเพื่อที่จะทำการกู้สัญญาณกลับคืนมาเป็นสัญญาณที่มีความต่อเนื่องทางด้านเวลาได้อย่างถูกต้องและให้ความคิดเทียบที่ลดน้อยลงด้วย



รูปที่ 2.7.9 (a) สเปกตรัมสัญญาณที่มีความต่อเนื่องทางด้านเวลา

(b) สเปกตรัมสัญญาณที่เป็นดิสครีต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพิจารณาสัญญาณที่มีความต่อเนื่องทางด้านเวลา ที่ให้ $F_b(j\Omega)$ ที่ไม่ได้จำกัดเรื่องแบนด์วิดธ์ ให้มีค่าอยู่ในช่วง $\pm\pi/T$ ดังที่ได้แสดงในรูปที่ 2.7.9 (a) ซึ่งจะมีผลทำให้ในทางคิสิกส์ของสัญญาณจะแทนด้วยสเปกตรัมที่ทำให้เกิดการเหลื่อมล้ำกันของสเปกตรัมที่อยู่ใกล้เคียงกันเป็นผลทำให้ผลรวมทางด้านสเปกตรัมทำให้ไม่สามารถที่จะกู้สัญญาณกลับคืนมาได้ ซึ่งเราเรียกว่าเป็นการเกิด ALIASING ซึ่งเป็นผลที่ไม่ต้องการ

ดังนั้นภายใต้สภาวะการณนี้เราสามารถที่จะกำหนดอัตราการสุ่มตัวอย่างเพื่อที่จะกำจัดผลของ Aliasing นี้ได้คือ

$$\pi/T > \Omega_A \quad (2.7.1)$$

และเราจะรู้จักขอบเขตที่ด้านค่าสุดของความถี่ในการสุ่มสัญญาณ (Ω_A/π) นี้ว่า Nyquist Rate บนพื้นฐานทางด้านวิศวกรรมนั้นสมการ (2.7.1) ค่าค่าที่สุ่มของ T และ Ω_A ซึ่งเป็นแบนด์วิดธ์ที่จำกัด

2.8 TMS320C50

TMS320C50 เป็นโปรเซสเซอร์ที่ทำงานในประมวลผลเชิงตัวเลขหรือ DSP (Digital Signal Processing) ซึ่งตระกูล TMS320 จะมีอยู่หลายเบอร์ด้วยกัน เช่น TMS320C50, TMS320C51 และ TMS320C53 ซึ่งเป็นการนำเอาสถาปัตยกรรมของ TMS320C25 มาปรับปรุงให้ดีขึ้นทั้งในด้านความสะดวกในการใช้งานและในเรื่องความเร็วในการประมวลผล

จุดเด่นของ TMS320C50

1. มีหน่วยความจำข้อมูล(RAM) บนบอร์ดขนาด 10 กิโลไบต์
2. เวลาที่ใช้ในการประมวลผลจะเร็วขึ้นคือ 30-50 นาโนวินาทีต่อคำสั่ง
3. สามารถนำมาใช้งานแทน TMS320C1X และ TMS320C2X ได้
4. มีหน่วยความจำข้อมูล(RAM)สำหรับโปรแกรม/ข้อมูลขนาด $9K \times 16$ บิต
5. มี ROM ขนาด $2K \times 16$ บิต ไว้สำหรับการบูต
6. สามารถต่อหน่วยความจำข้างนอกได้ถึง $224 K \times 16$ บิต ซึ่งประกอบด้วย หน่วยความจำสำหรับเก็บโปรแกรม 64K หน่วยความจำเก็บข้อมูล 64K, สำหรับอินพุต/เอาต์พุตและอื่นๆ อีก 64 K
7. มี ALU(Arithmetic Logic Unit), ACC(Accumulator) และ ACCB(Accumulator Buffer) ขนาด 32 บิต
8. มี PLU (Parallel Logic Unit) ขนาด 16 บิต
9. มีคำสั่งในการคูณ 16 บิตที่ทำงานใน 1 ไจเกิล
10. มีรีจิสเตอร์ถึง 8 ตัวในการคำนวณและเก็บค่า
11. มีสแต็ค (STACK)
12. มีคำสั่งสำหรับการเลื่อนบิตตั้งแต่ 0-16 บิต
13. มีการอ้างแอดเดรสแบบวนรอบ(Circular) โดยการใช้บัฟเฟอร์แบบวนรอบ(Circular Buffer)
14. มีคำสั่งสำหรับการทำซ้ำ โดยเฉพาะสำหรับการเคลื่อนย้ายบล็อกละหว่างหน่วยความจำ
15. มีคำสั่งเคลื่อนย้ายข้อมูลระหว่างหน่วยความจำข้อมูลและหน่วยความจำโปรแกรม

เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี หากมีการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย

สถาปัตยกรรมของ TMS320C50 สร้างขึ้นมาเพื่อให้มีความเร็วในการทำงานด้านการประมวลผล สัญญาณเชิง คณิต (DSP) และเพื่อให้การทำงานของบัสไม่ขึ้นต่อกัน จึงแยกออกเป็น บัสโปรแกรม (Program Bus) และ บัสข้อมูล (Data Bus) ออกจากกัน โดยบัสโปรแกรมจะเป็นทางเข้าของรหัสคำสั่ง และโอเพอร์เรนด์ของคำสั่ง ส่วนบัสข้อมูลจะเชื่อมต่อโดยตรงระหว่างหน่วยความจำที่ใช้เก็บข้อมูลกับ วงจรการทำงานประมวลผล เช่น ALU และ AR0-AR7 ซึ่งโครงสร้างการคำนวณทางคณิตศาสตร์นี้ สืบหลักการให้ทำงานอย่างมีประสิทธิภาพ เช่น การเลื่อนบิต , การคูณ และคำสั่งทางลอจิก

ตำแหน่งขาและหน้าที่การทำงานของ TMS320C50 แสดงดังในตารางที่ 2.8.1

ตารางที่ 2.8.1 แสดงตำแหน่งขาและหน้าที่การทำงานของ TMS320C50

สัญญาณ	ขา	สถานะ	การทำงาน
กลุ่มบัสแอดเดรสและบัสข้อมูล(Address and Data Bus)			
A15(MSB)	77	I/O/Z	เป็นบัสแบบขนาน (Parallel Address Bus) ที่ใช้สำหรับชี้ตำแหน่งของ หน่วยความจำข้อมูลและหน่วยความจำโปรแกรม หรือ I/O ภายนอก เมื่ออยู่ในโหมดโฮลด์ (Hold Mode) ก็จะอยู่สถานะของอิมพีแดนซ์สูง (High Impedance) สัญญาณเหล่านี้ใช้เป็นอินพุต สำหรับ DMA ภายนอกของแรมภายใน (On-Chip Single Access RAM) ซึ่งจะเป็อินพุต เมื่อ \overline{HOLDA} และ \overline{BR} ถูกขับ (Drive) ให้มีสถานะต่ำ (Low)
A14	76		
A13	75		
A12	74		
A11	73		
A10	72		
A9	64		
A8	63		
A7	62		
A6	61		
A5	60		
A4	59		
A3	58		
A2	57		
A1	56		
A0 (LSB)	55		

สัญญาณ	ขา	สถานะ	การทำงาน
D15(MSB)	6	I/O/Z	เป็นบัสข้อมูลแบบขนาน (Parallel Data Bus) ใช้ส่งผ่านข้อมูลระหว่างซีพียูหลัก (Core CPU) กับหน่วยความจำข้อมูล/โปรแกรมภายนอก หรือ อุปกรณ์ I/O เมื่อไม่มีเอาต์พุตสัญญาณเหล่านี้จะเป็นอิมพีแดนซ์สูง หรือ เมื่อขา RS หรือ HOLD อยู่ในสถานะต่ำ (Active Low) และขา OFF เป็นสถานะต่ำ นอกจากนี้ยังใช้สำหรับ DMA ภายนอกของแรม(Single Access RAM)
D14	7		
D13	8		
D12	9		
D11	10		
D10	11		
D9	12		
D8	13		
D7	23		
D6	24		
D5	25		
D4	26		
D3	27		
D2	28		
D1	29		
D0 (LSB)	30		
กลุ่มสัญญาณควบคุมหน่วยความจำ			
\overline{DS}	89	O/Z	เลือกหน่วยความจำข้อมูล/โปรแกรมและ I/O ปกติมีสถานะสูงแต่เมื่อเป็นสถานะต่ำจะเป็นการติดต่อกับภายนอกเมื่อขา \overline{OFF} อยู่ในสถานะต่ำ ขาสัญญาณควบคุมจะอยู่ในสถานะอิมพีแดนซ์สูง
\overline{PS}	91		
\overline{IS}	90		
READY	128	I	สัญญาณข้อมูลพร้อม (Data Ready Input) ให้แสดงเมื่ออุปกรณ์ภายนอกส่งข้อมูลเรียบร้อยแล้ว และเมื่อยังทำงานไม่เสร็จ (READY = 0) จะต้องมีการรอ 1 ไซเคิล (Wait 1 Cycle) และเช็คขา READY อีกครั้ง ในสภาวะปกติขา READY จะทำงานหลังจากที่มีสัญญาณ BR

สัญญาณ	ขา	สถานะ	การทำงาน
กลุ่มสัญญาณควบคุมหน่วยความจำ			
R/\overline{W}	92	I/O/Z	สัญญาณอ่าน/เขียน (Read/write Signal) เป็นสัญญาณควบคุมการอ่านและเขียนข้อมูล จะเป็นสถานะอิมพีแดนซ์สูง เมื่ออยู่ในโหมดการโฮลด์ ถูกใช้ใน DMA ของ RAM ภายนอก เมื่อ \overline{HOLDA} และ \overline{IAQ} อยู่ในสถานะต่ำ ใช้แสดงทิศทางของบัสข้อมูล สำหรับ DMA อ่าน (Read อยู่ในสถานะสูง) และ เขียน (Write อยู่ในสถานะต่ำ)
\overline{STRB}	93	I/O/Z	สัญญาณสโตรบ (Strob Signal) ปกติมีอยู่ในสถานะสูง จะเป็นสถานะต่ำ เมื่อบัสภายนอกเป็นอิมพีแดนซ์สูง ในโหมดการโฮลด์เมื่อ \overline{HOLDA} และ \overline{IAQ} แอคทีฟสัญญาณนี้จะใช้เลือกการเข้าถึงหน่วยความจำ
RD	82	O/Z	สัญญาณเลือกอ่าน (Read Select) ขานี้จะทำงานเมื่อมีการอ่าน จะต่อโดยตรงกับ OE ของอุปกรณ์ภายนอก สัญญาณนี้จะใช้ในการอ่านหน่วยความจำโปรแกรม/ข้อมูล และ I/O ภายนอก เป็นอิมพีแดนซ์สูงเมื่ออยู่ในโหมดการโฮลด์
กลุ่มสัญญาณมัลติโพรเซสซิง(Multiprocessing)			
\overline{HOLD}	129	I	สัญญาณโฮลด์ (Hold Input) เป็นสัญญาณที่ใช้แสดงว่ากำลังมีการติดต่อกับบัสตำแหน่ง , บัสข้อมูล และบัสควบคุม เมื่อถูกตอบรับ (Acknowledge) โดย TMS320C5X จะอยู่ในสภาวะอิมพีแดนซ์สูง
\overline{HOLDA}	108	O/Z	สัญญาณตอบรับสัญญาณโฮลด์ (Hold Acknowledge Signal) ใช้แสดงว่าวงจรอยู่ในสภาวะโฮลด์ (Hold State) ทั้งบัสตำแหน่ง , บัสข้อมูล และบัสควบคุมอยู่ในสภาวะอิมพีแดนซ์สูง
\overline{BR}	94	I/O/Z	สัญญาณการขอใช้บัส (Bus Request Signal) แสดงเมื่อมีการติดต่อกับหน่วยความจำข้อมูล สัญญาณจากขานี้ใช้กับหน่วยความจำข้อมูลที่ว่างได้ 32 กิโลเวิร์ด เมื่อ \overline{HOLDA} อยู่ในสถานะต่ำ สัญญาณจากขานี้ใช้กับ DMA ของ แรมภายนอก \overline{BR} จะเป็นสถานะต่ำ เมื่อติดต่อกับแรมภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ	ขา	สถานะ	การทำงาน
\overline{IAQ}	1	O/Z	สัญญาณรับคำสั่ง (Instruction Acquisition Signal) จะแสดงค่าสถานะต่ำ เมื่อมีการเข้าถึงคำสั่งที่อยู่บนแอสเซมบลี ใช้กับ DMA ของ RAM ภายนอก เมื่อ <i>HOLDA</i> อยู่ในสถานะต่ำ
\overline{BIO}	130	I	สัญญาณควบคุมบรานซ์ (Branch Control Input) ถ้าเป็นสถานะต่ำ จะเป็นการให้ทำคำสั่งที่เป็นเงื่อนไข สัญญาณนี้จะทำงานเมื่อมีการเพ็ชคำสั่งที่เป็นเงื่อนไข
XF	109	O/Z	สัญญาณติดต่อกาภายนอก (External Flag Output) ถูกเซตให้เป็นสถานะสูงหรือสถานะต่ำ โดยคำสั่งพิเศษหรือโดยโพลลค่าในสเตจรีจิสเตอร์ (State Register(ST1)) เมื่อรีเซตขานี้จะเป็นสถานะสูง
\overline{LACK}	112	O/Z	สัญญาณตอบรับการอินเทอร์รัพต์ (Interrupt Acknowledge Signal) แสดงค่าเมื่อรับค่าการอินเทอร์รัพต์ (Interrupt Vector) ซึ่งกำหนด โดย A15 - A0
การอินิเชียลไลซ์ (Initialization), อินเทอร์รัพต์, คำตั้งรีเซต			
INT4 INT3 INT2 INT1	41 40 39 38	I	สัญญาณอินเทอร์รัพต์จากผู้ใช้ภายนอก (External User Interrupt Input) กำหนดโดยรีจิสเตอร์ควบคุมการอินเทอร์รัพต์ (Interrupt Mask Register) และ บิตอินเทอร์รัพต์โหมด (Interrupt Mode bit) สามารถรีเซตผ่านรีจิสเตอร์บอกรินเทอร์รัพต์
\overline{NMI}	42	I	สัญญาณอินเทอร์รัพต์ที่ไม่สามารถควบคุมได้ (Non-Maskable Interrupt) เป็นอินเทอร์รัพต์ภายนอก ไม่สามารถควบคุมโดย \overline{INTM} หรือ MR เมื่อ <i>MNI</i> ทำงานจะมีการอินเทอร์รัพต์
MC / MP	5	I	ขาคเลือกโหมด ไมโคร โปรเซสเซอร์/ไมโครคอมพิวเตอร์ (Microprocessor/Microcomputer Mode) ถ้าเป็นสถานะต่ำ (Microcomputer Mode) จะทำให้โปรแกรมรวมภายในส่งไปยังหน่วยความจำโปรแกรมในโหมดไมโคร โปรเซสเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ	ขา	สถานะ	การทำงาน															
สัญญาณออสซิลเลเตอร์/ไทม์เมอร์ และ CLKIN1/2																		
CLKOUT1	110	O/Z	สัญญาณนาฬิกาส่งออก (Master Clock Output Signal หรือ CLKIN2 Frequency) มีค่าไซเคิลเท่ากับอัตราแมชชีนไซเคิล (Machine-Cycle) ของซีพียู															
CLKMD1 CLKMD2	71 103	I	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;"><u>CLKMD1</u></th> <th style="text-align: center;"><u>CLKMD2</u></th> <th></th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>สัญญาณนาฬิกาภายนอก เป็นสัญญาณนาฬิกาเข้าจากขา X2/CLKIN ทำให้ออสซิลเลเตอร์ (Oscillator) ภายใน และ PLL Disable</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td>สำหรับตรวจสอบ</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>เป็นสัญญาณนาฬิกาอินพุท (Input Clock) สำหรับ CLKIN2 ทำให้ออสซิลเลเตอร์ภายในหยุดทำงานและ PLL ทำงานแทน</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td>เป็นสัญญาณนาฬิกาอินพุท สำหรับขา X2/CLKIN1 ทำให้ออสซิลเลเตอร์ภายในทำงานและ PLL ภายในไม่ทำงาน</td> </tr> </tbody> </table>	<u>CLKMD1</u>	<u>CLKMD2</u>		0	0	สัญญาณนาฬิกาภายนอก เป็นสัญญาณนาฬิกาเข้าจากขา X2/CLKIN ทำให้ออสซิลเลเตอร์ (Oscillator) ภายใน และ PLL Disable	0	1	สำหรับตรวจสอบ	1	0	เป็นสัญญาณนาฬิกาอินพุท (Input Clock) สำหรับ CLKIN2 ทำให้ออสซิลเลเตอร์ภายในหยุดทำงานและ PLL ทำงานแทน	1	1	เป็นสัญญาณนาฬิกาอินพุท สำหรับขา X2/CLKIN1 ทำให้ออสซิลเลเตอร์ภายในทำงานและ PLL ภายในไม่ทำงาน
<u>CLKMD1</u>	<u>CLKMD2</u>																	
0	0	สัญญาณนาฬิกาภายนอก เป็นสัญญาณนาฬิกาเข้าจากขา X2/CLKIN ทำให้ออสซิลเลเตอร์ (Oscillator) ภายใน และ PLL Disable																
0	1	สำหรับตรวจสอบ																
1	0	เป็นสัญญาณนาฬิกาอินพุท (Input Clock) สำหรับ CLKIN2 ทำให้ออสซิลเลเตอร์ภายในหยุดทำงานและ PLL ทำงานแทน																
1	1	เป็นสัญญาณนาฬิกาอินพุท สำหรับขา X2/CLKIN1 ทำให้ออสซิลเลเตอร์ภายในทำงานและ PLL ภายในไม่ทำงาน																
X2/CLKIN	122	O	ขาอินพุทสำหรับออสซิลเลเตอร์ภายใน (Input Pin to Internal Oscillator From Crystal) ถ้าออสซิลเลเตอร์ภายในไม่ถูกต้อง สัญญาณนาฬิกาจะเป็นอินพุทสำหรับอุปกรณ์บนขานี้ แมชชีนไซเคิลภายในเป็นครึ่งหนึ่งของอัตรา CLK															
X1			เป็นขาเอาต์พุทของออสซิลเลเตอร์ภายในสำหรับคริสตอล ถ้าไม่ใช้ออสซิลเลเตอร์ภายในจะไม่มีการต่อกับขานี้															
CLKIN2			เป็นอินพุทสำหรับสัญญาณนาฬิกาสำหรับอัตราแมชชีน (Machine Rate)															
TOUT			เอาต์พุทไทม์เมอร์ (Timer Output) ขานี้ให้สัญญาณพัลส์เมื่อไทม์เมอร์ภายใน (On-Chip Timer) นับถึง 0 ความกว้างพัลส์เท่ากับ CLKOUT 1 ไซเคิล															

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

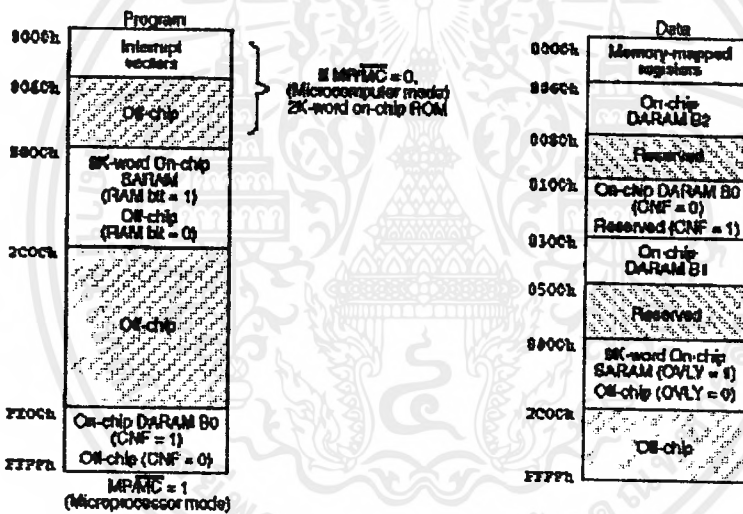
สัญญาณ	ขา	สถานะ	การทำงาน
สัญญาณพอร์ตอนุกรม			
CLKR	46	I	เป็นขาที่รับสัญญาณนาฬิกาจากข้างนอกเพื่อกำหนดให้การรับข้อมูล (DR/TDR) เข้าไปเก็บไว้ที่ RSR (Serial Port Receive Shift Register) แต่ถ้าขานี้ไม่ใช่สามารถที่จะใช้เป็นขาอินพุทของ IN0 ของ SPC/TSPC รีจิสเตอร์ได้
CLKRT	126	I	
CLKX	124	I/O/Z	เป็นขาที่แสดงสัญญาณนาฬิกาจากข้างนอกเพื่อกำหนดให้ DR/TDR ส่งข้อมูลไปที่ DX/TDX CLKX จะเป็นอินพุท ถ้า MCM บิต ที่การควบคุมพอร์ตอนุกรม (Serial Port Control) มีค่าเป็น 0 และอาจจะับความถี่เป็น 1/4(CLKOUT1) เมื่อ MCM เป็น 1 ถ้าขานี้ไม่ใช่ สามารถที่จะทำเป็นอินพุทของบิต INT1 ของ SPC/TSPC รีจิสเตอร์
TCLKX	123	I/O/Z	
DR	43	I	เป็นขาที่รับสัญญาณข้อมูล ซึ่งเมื่อรับเข้ามาแล้วจะเก็บไว้ที่ RSR (Serial Port Transmit Shift Register)
TDR	44	I	
DX	106	O/Z	เป็นขาเพื่อส่งสัญญาณข้อมูล ซึ่งข้อมูลจะส่งจาก XSR (Serial Port Receive Shift Register)
TDX	107	O/Z	
FSR	45	I	แสดงสัญญาณการพร้อมของเฟรม (Frame Synchronization) สำหรับรับสัญญาณอินพุท TFSR จะเป็นได้ทั้งอินพุท/เอาต์พุทเมื่อ TSM ถูกเซตให้เป็น 1
TFSR/TSDD	125	I/O/Z	
FSX	104	I/O/Z	แสดงสัญญาณการพร้อมของเฟรม (Frame Synchronization) สำหรับส่งสัญญาณขานี้จะเลือกได้โดยการใช้ซอฟต์แวร์ และจะเป็นเอาต์พุทเมื่อ TSM ถูกเซตให้เป็น 1
TFSX/TFRM	105	I/O/Z	
TCK	34	I	สัญญาณนาฬิกาตรวจสอบ JTAG (JTAG Test Clock) เป็นสัญญาณนาฬิกาแบบฟรีรันนิ่ง (Free-Running) ซึ่งมีค่าควิตซ์ไซเคิล 50% การเปลี่ยน TAP (Test Access Port) ซึ่งเป็นอินพุทสัญญาณนาฬิกาจะควบคุม TAP รีจิสเตอร์คำสั่ง (Instruction Register) หรือเลือกการทดลองรีจิสเตอร์ข้อมูล (Data Register) ที่ขอขาขึ้นของ TCK ในการเปลี่ยน TAP จะเป็นสัญญาณเอาต์พุทจะปรากฏที่ขอบาลลงของ TCK
TDI	67	I	เป็นสัญญาณนาฬิกาเพื่อเลือกรีจิสเตอร์ในขอบขาขึ้นของ TCK
TDO	100	O/Z	เป็นการทดสอบเอาต์พุทข้อมูลในขอบขาลงของ TCK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ	ขา	สถานะ	การทำงาน
DMS	31	I	เป็นการเลือกโหมดทดสอบ JTAG และเป็นสัญญาณนาฬิกาอินพุตที่ทดสอบพอร์ต ตรวจสอบการเข้าถึง (Test Access Port (TAP)) จะทำงานที่ขอบขาขึ้นของ TCK
\overline{TRST}	2	I	ทดสอบการรีเซ็ตจะเป็นสถานะสูง
EMU0	118	I/O/Z	ขาอิมูเลเตอร์ 1 / ขาหยุดการทำงานของเอาต์พุต (Emulator Pin 1 / Disable all Output) จะทำงานที่สถานะต่ำ
EMU1/OFF	119		

การจัดหน่วยความจำของ TMS320C50

TMS320C50 จะมีการแบ่งหน่วยความจำเป็นส่วนของหน่วยความจำโปรแกรม และหน่วยความจำข้อมูลดังรูปที่ 2.10.2



รูปที่ 2.8.2 แสดงการจัดตำแหน่งของหน่วยความจำของ TMS320C50

หน่วยความจำข้อมูล(Data Memory)

ในการออกแบบของ TMS320C5X จะใช้สถาปัตยกรรมแบบฮาร์เวิร์ด (Harward) ซึ่งเป็นการแยกหน่วยความจำข้อมูล และหน่วยความจำโปรแกรมแยกออกจากกัน ซึ่งจะมีผลดีในเรื่องความเร็ว

หน่วยความจำข้อมูล ของ TMS320C5X สามารถขยายได้ถึง $64K \times 16$ บิต ซึ่ง TMS320C51 มีแรมบนชิปขนาด 1056 เวิร์ด บนชิป ซึ่งจะมีข้อดีคือ

1. มีการปฏิบัติการได้เร็ว ไม่มีการรอ (Wait State)
2. ทำงานโดยวิธีการ ไปป์ไลน์ ทำให้มีการทำงานได้อย่างรวดเร็ว
3. ช่วยลดค่าใช้จ่ายเรื่องการต่อหน่วยความจำภายนอก
4. ช่วยประหยัดไฟ ซึ่งถ้าให้การต่อหน่วยความจำภายนอกจะเปลืองกว่า

ในการกำหนดการใช้หน่วยความจำข้อมูลจะต้องมีการเซตค่าของบิตของรีจิสเตอร์ควบคุมหน่วยความจำ (Control State Register) ดังนี้

ตารางที่ 2.8.2 แสดงการเซตบิตรีจิสเตอร์ควบคุมหน่วยความจำเพื่อใช้หน่วยความจำข้อมูล

CNF	OVLV	DRAM B0	DRAM B1	DRAM B2	SARAM	Off-Chip
0	0	100h-2FFh	300h-4FFh	60h-7Fh		800h-FFFFh
0	1	100h-2FFh	300h-4FFh	60h-7Fh	800h-2BFFh	2C00h-FFFFh
1	0	-	300h-4FFh	60h-7Fh		800h-FFFFh
1	1	-	300h-4FFh	60h-7Fh	800h-2BFFh	2C00h-FFFFh

เนื่องจากการติดต่อกับรีจิสเตอร์ต่างๆ และการส่งค่าต่างๆ จะกระทำบนพยางค์ 0 ซึ่งจะเป็นการใช้แบบส่งแบบรีจิสเตอร์ (Register Memory Map) ดังตารางที่ 2.8.3

ตารางที่ 2.8.3 แสดง ตำแหน่งแอดเดรสข้อมูล พยางค์ 0

Memory-Mapped Core Processor Registers			
Name	Address		Description
	Dec	Hex	
----	0-3	0-3	Reserved
IMR	4	5	Interrupt Mask Register
GREG	5	5	Global Memory Allocation Register
IFR	6	6	Flag Interrupt Register
PMST	7	7	Processor Mode Status Register
RPTC	8	8	Repeat Counter Register
BRCR	9	9	Block Repeat Counter Register
PASR	10	A	Block Repeat Program Address Start Register
PAER	11	B	Block Repeat Program Address End Register
TREG0	12	C	Temporary Register Used for Multiplicand
TREG1	13	D	Temporary Register Used for Dynamic Shift Count(5 bits only)
TREG2	14	E	Temporary Register Used as Bit Pointer in Dynamic Bit Test(4 bits only)
DBMR	15	F	Dynamic Bit Manipulation Register
AR0	16	10	Auxiliary Register Zero
AR1	17	11	Auxiliary Register One
AR2	18	12	Auxiliary Register Two
AR3	19	13	Auxiliary Register Three
AR4	20	14	Auxiliary Register Four

Name	Address		Description
	Dec	Hex	
AR5	21	15	Auxiliary Register Five
AR6	22	16	Auxiliary Register Six
AR7	23	17	Auxiliary Register Seven
INDX	24	18	Index Register
ARCR	25	19	Auxiliary Register Compare Register
CBSR1	26	1A	Circular Buffer 1 Start Register
CBER1	27	1B	Circular Buffer 1 End Register
CBSR2	28	1C	Circular Buffer 2 Start Register
CBER2	29	1D	Circular Buffer 2 End Register
CBCR	30	1E	Circular Buffer Control Register
BMAR	31	1F	Block Move Address Register
Memory-mapped Peripheral Register			
DRR	32	20	Data Receive Register
DXR	33	21	Data Transmit Register
SPC	34	22	Serial Port Control Register
----	35	23	Reserved
TIM	36	24	Timer Register
PRD	37	25	Period Register
TCR	38	26	Timer Control Register
----	39	27	Reserved
PDWSR	40	28	Program/Data S/W Wait-State Register
IOWSR	41	29	I/O S/W Wait-State Control Register
CWSR	42	2A	S/W Wait-State Control Register
-	43-47	2B-2F	Reserved
TRCV	48	30	TDM Data Receive Register
TDXR	49	31	TDM Transmit Data Register
TSPC	50	32	TDM Serial Port Control Register
TCSR	51	33	TDM Channel Select Register
TRTA	52	34	TDM Receive/Transmit Address Register
TRAD	53	35	TDM Received Address Register
-----	54-79	36-4F	Reserved

Name	Address		Description
	Dec	Hex	
Memory-Mapped I/O Ports			
PA0	80	50	I/O Port 50h
PA1	81	51	I/O Port 51h
PA2	82	52	I/O Port 52h
PA3	83	53	I/O Port 53h
PA4	84	54	I/O Port 54h
PA5	85	55	I/O Port 55h
PA6	86	56	I/O Port 56h
PA7	87	57	I/O Port 57h
PA8	88	58	I/O Port 58h
PA9	89	59	I/O Port 59h
PA10	90	5A	I/O Port 5Ah
PA11	91	5B	I/O Port 5Bh
PA12	92	5C	I/O Port 5Ch
PA13	93	5D	I/O Port 5Dh
PA14	94	5E	I/O Port 5Eh
PA15	95	5F	I/O Port 5Fh

หน่วยความจำโปรแกรม (Program Memory)

ในการใช้งานของหน่วยความจำโปรแกรม ซึ่งสามารถขยายได้ถึง 64 K ซึ่ง TMS320C50 จะมี ROM (ROM), SARAM และ DARAM ซึ่งมีความเร็วสูง โดยไม่ต้องมีสภาวะการรอ (Wait State)

ในการทำงานของ TMS320C50 สามารถใช้ร่วมกับรอมภายในขนาด 4 K ซึ่งสามารถโปรแกรมจากโรงงาน มีความเร็วในการทำงานเต็มที่ ในการเลือกใช้จะต้องกำหนดทิศทางของ $\overline{MP}/\overline{MC}$ หากขาดังกล่าวมีสภาวะสูง หรือเป็น 1 ตำแหน่ง 4 K แรก จะเป็นหน่วยความจำภายนอกชิพ แต่ถ้ามีสถานะต่ำหรือเป็น 0 ตำแหน่ง 4 K เวิร์ดแรกก็จะเป็นรอมภายในชิพดังแสดงให้เห็นในรูปการจัดแบ่งหน่วยความจำ

การกำหนดสถานะ (Status) ของค่าต่างๆ จะกำหนดตามตารางที่ 2.8.4

ตารางที่ 2.8.4 แสดงการกำหนดค่าสำหรับใช้หน่วยความจำโปรแกรม

CNF	RAM	MP/MC	ROM	SARAM	DRAM B0	Off-Chip
0	0	0	0000h-07FFh			0800h-FFFFh
0	0	1				0000h-FFFFh
0	1	0	0000h-07FFh	2000h-23FFh		2C00-FFFFh
0	1	1		2000h-23FFh		0000h-FFFFh 2C00-FFFFh
1	0	0	0000h-07FFh		FE00h-FFFFh	0800h-FDFFh
1	0	1			FE00h-FFFFh	0000h-FDFFh
1	1	0	0000h-07FFh	2000h-23FFh	FE00h-FFFFh	2C00-FDFFh
1	1	1		2000h-23FFh	FE00h-FFFFh	0000h-07FFh 2C00-FDFFh

การใช้งานของการอินเทอร์รัพต์ จะต้องกำหนดแอดเดรสของแอดเดรสในหน่วยความจำโปรแกรม ซึ่งอินเทอร์รัพต์แอดเดรสจะแสดงดังตารางที่ 2.10.5

ตาราง 2.8.5 แสดงตำแหน่งของอินเทอร์รัพต์แอดเดรส

Name	Location		Priority	Function
	Dec	Hex		
RS	0	0	1(highest)	External reset signal
INT1	2	2	3	External user interrupt #1
INT2	4	4	4	External user interrupt #2
INT3	6	6	5	External user interrupt #3
TINT	8	8	6	Internal timer interrupt
RINT	10	A	7	Serial port receive interrupt
XINT	12	C	8	Serial port transmit interrupt
TRNT	14	E	9	TDM port receive interrupt
TXNT	16	10	10	TDM port transmit interrupt
INT4	18	12	11	External user interrupt #4
-	20-33	14-21	N/A	Reserved
TRAP	34	22	N/A	Software trap instruction
NMI	36	24	2	Nonmaskable interrupt
-	38-41	26-29	N/A	Reserved for emulation and test
-	42-47	2A-2F	N/A	Software interrupts

โหมดการเข้าถึงหน่วยความจำ (Memory Addressing Mode)

การอ้างแอดเดรสสามารถอ้างได้ถึง 64 K สำหรับหน่วยความจำโปรแกรม และ 96 K สำหรับหน่วยความจำข้อมูล TMS320C50 มีวิธีการอ้างแอดเดรสได้ 8 วิธี ดังนี้

1. โดยการเข้าถึงโดยตรง (Direct Address Bus) โดยการใช้แอดเดรสบัสโดยตรง ซึ่งต้องมีรีจิสเตอร์ชี้เพจของข้อมูล (Data Page Pointer (DP)) บอกคี่ว่าเป็นเพจใด DP สามารถอ้างได้ 512 เพจ แต่ละเพจจะมีค่าเท่ากับ 128 เวิร์ด ตัวอย่างเช่น ADD 01h หมายถึง เอาข้อมูลที่รีจิสเตอร์ตำแหน่ง 010h (ซึ่ง DP ชี้บอกว่าจะอยู่ที่เพจใด) นำมาบวกกับ ACC

2. โดยการเข้าถึงหน่วยความจำ (Memory Map) จะเหมือนกับกับวิธีการแรกแต่จะใช้เฉพาะเพจ 0 เท่านั้น เช่น

LAMM PMST โดยที่ PMST เป็นรีจิสเตอร์

3. โดยการใช้ ออกซิลารีรีจิสเตอร์ (Auxiliary Register) โดยการอ้างแอดเดรสที่บรรจุใน AR0 - AR7 เช่น ADD* ซึ่งจะเป็นตัวบอกว่า AR ตัวที่เท่าใด ตัวอย่างเช่น

ARP	4	4
AR4	0302h	0302h
หน่วยความจำข้อมูล		
302h	2h	2h
ACC	2h	4h

จะเห็นว่าแอดเดรสที่ได้อยู่ใน AR4 (0302h)

4. โดยการใช้รีจิสเตอร์คำสั่ง (Instruction Register) ซึ่งเป็นการเอาแอดเดรสโดยตรงมาจากคำสั่งเลข เช่น

ADD #0FFFh

เป็นการนำเอาข้อมูลที่แอดเดรส 0FFh มาบวกกับ ACC

โดยวิธีนี้จะอ้างได้แบบช่วงสั้น (Short Immediate) คือค่าแอดเดรส จะเป็น ได้ตั้งแต่ 00h ไปจนถึง 0FFh

5. โดยการใช้ PC (Program Counter) จะเหมือนการใช้รีจิสเตอร์คำสั่ง แต่สามารถอ้างแบบช่วงยาวได้ (Long Immediate (0000h-0FFFFh)) เช่น

ADD #0FFFh

6. โดยการเข้าถึงรีจิสเตอร์โดยตรง (Register Access) เป็นการใช้อินสตรักชันที่ทำงานโดยเฉพาะ เช่น TREG0, TREG1, TREG2, ARCR, DBMR ตัวอย่างการใช้คำสั่งเช่น

APL *, AR7

7. โดยการใช้คำสั่งตัวที่ 2 เป็นตัวชี้แอดเดรส เช่น คำสั่ง BLDD #02345h จะหมายถึงให้นำเอาข้อมูลที่รีจิสเตอร์แอดเดรส 02345h ไปเก็บไว้ที่แอดเดรส 012h

8. โดยการอ้างถึงหน่วยความจำเป็นบล็อก (Block Memory Address Register) ซึ่ง BMAR จะเป็นตัวชี้ข้อมูล ตัวอย่างการใช้งานคำสั่งเช่น

BLDD BMAR, 012h

อุปกรณ์เสริม (Peripherals)

TMS320C50 สามารถที่จะต้องอินเทอร์เฟสกับภายนอกได้ 8 วิธี คือ

1. การอินเทอร์รัพต์
2. พอร์ตอนุกรม (Serial Port)
3. พอร์ตอนุกรมแบบอาศัยการแบ่งช่วงทางเวลา (TDM Serial Port)
4. ไทม์เมอร์
5. รีจิสเตอร์ใช้โปรแกรมให้เกิดสภาวะการรอ (Software -Programable wait State)
6. พอร์ตติดต่อภายนอก (I/O Port)
7. รีจิสเตอร์หารสัญญาณนาฬิกา (Divide-by-Clock)
8. XF และ BIO

ซึ่งทั้งหมดนี้จะควบคุมผ่านรีจิสเตอร์ควบคุม โดยการอาศัยวิธีการอ้างหน่วยความจำ ในการใช้งานจะเป็นการใช้โดยการส่งลงที่แอดเดรสของหน่วยความจำโปรแกรมและข้อมูล

ในที่นี้จะขอกล่าวเฉพาะหัวข้อที่สำคัญๆ

การอินเทอร์รัพต์

TMS320C50 มีการอินเทอร์รัพต์ทั้งหมด 16 อินเทอร์รัพต์ (INT16-INT1) แต่จะไม่ใช่พร้อมกันทั้งหมด (โดยปกติจะใช้เพียง 9)

การรีเซต (Reset) จะเป็นการอินเทอร์รัพต์จากภายนอกแบบไม่สามารถหลีกเลี่ยงได้ (Nonmaskable External Interrupt) เมื่อมีการรีเซตเกิดขึ้นจะทำให้เกิดผลดังนี้

1. CNF = 0 หมายถึง การใช้หน่วยความจำข้อมูลบนชิพ
2. PC = 0000h
3. บิต INTM = 1 และ IFR = 0 ทำให้อินเทอร์รัพต์ทุกตัวไม่ทำงาน
4. บิตสถานะ (Status Bit) จะเป็นดังนี้

0 → OV	1 → XF	1 → SXM	0 → PM	1 → HM
0 → BRAI	0 → TRM	0 → NDX	0 → CENM1	0 → CENM2
0 → IPTR	0 → OVLY	0 → AVIS	0 → RAM	0 → BIG
0 → CNF	1 → INTM	MP/MC(Pin) → PMST(MP/MC)	1 → C	
5. GREG = 000000
6. RPTC = 00
7. LACK จะแสดงสภาวะการเก็บการรีเซต

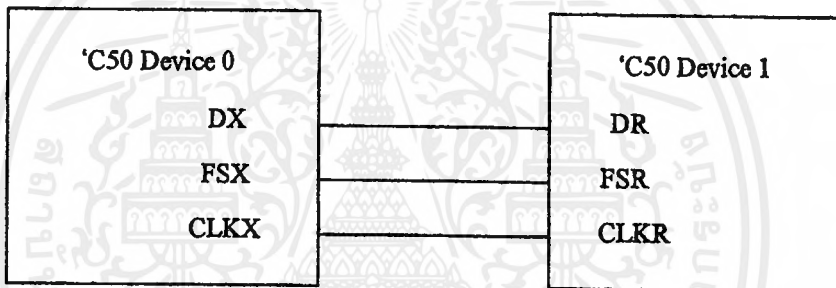
ในการควบคุมการอินเทอร์รัพต์จะมี IFR (Interrupt Flag Register) แสดงสภาวะการอินเทอร์รัพต์ และ IMR (Interrupt Mask Register) แสดงการ Mask การอินเทอร์รัพต์ ดังรูปที่ 2.8.3

15	9	8	7	6	5	4	3	2	1	0
Reserved	INT4	TXNT	TRNT	XINT	RINT	TINT	INT3	INT2	INT1	

รูปที่ 2.8.3 แสดงการอินเทอร์รัพต์แฟล็ก

ในการกำหนดให้ใช้อินเทอร์รัพต์จะต้องทำการเคลียร์ที่ INTM บิต ด้วย พอร์ตอนุกรม (serial Port)

Serial Port ของ TMS320C50 จะเป็นแบบฟูลดuplek คือสามารถรับและส่งได้ภายในเวลาเดียวกัน ทำให้สามารถติดต่อกับอุปกรณ์ภายนอกอื่นๆ ได้ไม่ว่าจะเป็น CODEC A/D หรือระบบอื่นๆ ในการต่อขาต่างๆจะเป็นตามรูปที่ 2.10.4



รูปที่ 2.8.4 แสดงการต่อพอร์ตอนุกรมกับพอร์ตภายนอก

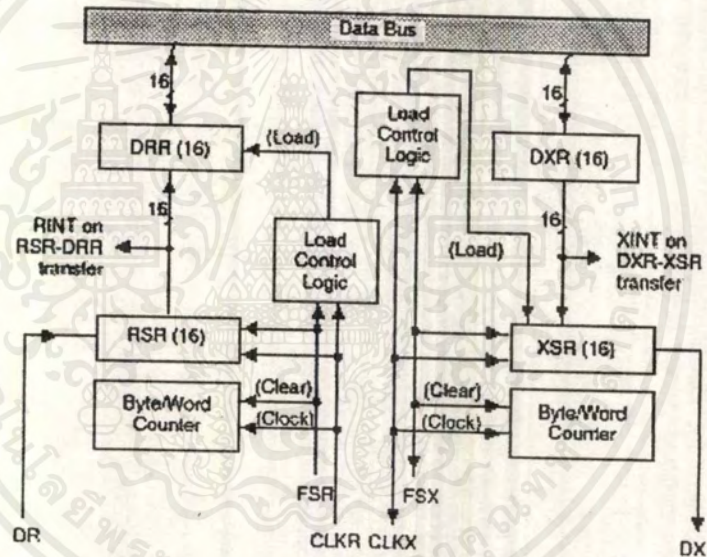
ตารางที่ 2.8.6 แสดงขาสัญญาณของพอร์ตอนุกรม

ขาสัญญาณ	ความหมาย
CLKX	สัญญาณนาฬิกาที่ใช้ในการส่ง
CLKR	สัญญาณนาฬิกาที่ใช้ในการรับ
DX	สัญญาณข้อมูลอนุกรมในการส่ง
DR	สัญญาณข้อมูลอนุกรมในการรับ
FSX	สัญญาณการซิงโครไนซ์เฟรมการส่ง
FSR	สัญญาณการซิงโครไนซ์เฟรมการรับ

ในการควบคุมพอร์ตอนุกรมจะมีรีจิสเตอร์ที่จำเป็นตามตารางที่ 2.8.7

ตารางที่ 2.8.7 แสดงรีจิสเตอร์พอร์ตอนุกรม

รีจิสเตอร์	ความหมาย
SPC	รีจิสเตอร์ควบคุมพอร์ตอนุกรม
DXR	รีจิสเตอร์ส่งข้อมูล
DRR	รีจิสเตอร์รับข้อมูล
XSR	รีจิสเตอร์เลื่อนการส่ง
RSR	รีจิสเตอร์เลื่อนการรับ

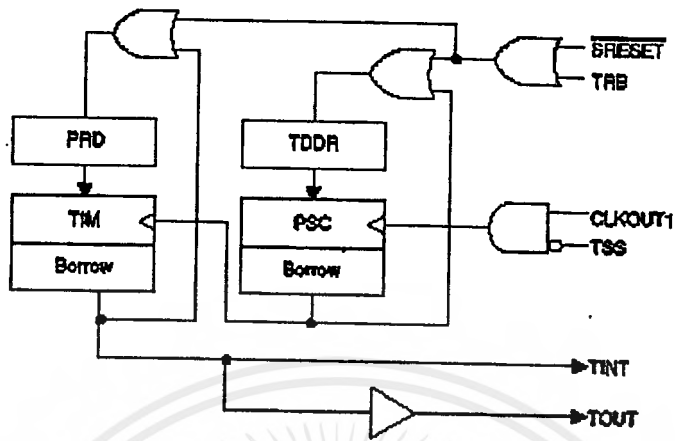


รูปที่ 2.8.5 แสดงบล็อกไออะแกรมการทำงานของพอร์ตอนุกรม

ในการส่งข้อมูลทำได้โดยการเขียนข้อมูลลงใน DXR (Data Transmit Register) โดย XSR (Transmit Shift Register) จะเป็นตัวเลื่อนบิตตามสัญญาณนาฬิกาจนครบไปคาบขา DX ส่วนการรับข้อมูลจะรับเข้ามาทางขา DR เข้ามาที่ RSR (Receive Shift Register) เมื่อครบแล้วจะทำการอินเทอร์รัพต์บอกให้ทราบ แล้วสามารถรับข้อมูลโดยการอ่านข้อมูลจาก DRR (Data Receive Register)

ไทม์เมอร์(Timer)

ในการใช้ไทม์เมอร์ของ TMS320C50 ซึ่งสามารถตั้งคาบเวลาของสัญญาณนาฬิกาได้ โดยสัญญาณจะออกที่ขา T_{OUT} ของ TMS320C50



รูปที่ 2.8.6 แสดงบล็อกโคะแกรมของไทม์เมอร์
ซึ่งในการคำนวณคาบเวลา จะเป็นตามสูตร

$$T = \frac{1}{t_0 \times (TDDR + 1) \times (PRD + 1)}$$

ซึ่ง t_0 เป็นคาบเวลาของสัญญาณนาฬิกาที่จ่ายให้ที่ขา CLKOUT1

TDDR (Timer Division Down Ratio)

PRD (Period Register) เป็นรีจิสเตอร์ที่สามารถใส่ค่าคาบเวลาได้

ในการทำงานจะต้องมีการใส่ค่าต่างๆ ลงบน TCR (Timer Control Register) ดังตารางที่ 2.8.8

ตารางที่ 2.8.8 แสดงรีจิสเตอร์ควบคุมไทม์เมอร์ (TCR)

15-12	11	10	9-6	5	4	3-0
Reserved	SOFT	FREE	PSC	TRB	TSS	TDDR

Software-Programmable Wait-State Generators

การกำเนิดสภาวะรอโดยใช้การโปรแกรมซอฟต์แวร์ สามารถใช้ขีด ไขเคลือบข้อมูลภายนอกขึ้นไปได้ถึง 7 machine cycles นี้หมายถึงการจัดการความสะดวกสำหรับการอินเตอร์เฟสอุปกรณ์ภายนอกซึ่งเวลาในการเข้าถึงข้อมูลเต็มความเร็วไม่พอกับความต้องการของ 'C5x อุปกรณ์ที่ต้องการมากกว่า 7 wait states สามารถถูกต่อเข้ากับทางด้านฮาร์ดแวร์ ที่สาย READY เมื่ออุปกรณ์ภายนอกทั้งหมดถูกทำให้มีรูปแบบสำหรับ zero wait state, สัญญาณนาฬิกาภายในที่เป็นตัวกำเนิดสภาวะรอ ก็จะถูกปิดเสีย, ยอมให้อุปกรณ์ถูก run ในโหมด lower power ของการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำเนิดสภาวะรอ โดยใช้การ โปรแกรมซอฟต์แวร์ถูกควบคุมโดยรีจิสเตอร์ wait state ขนาด 16 บิต 2 ตัว (PDWSR และ IOWSR) และรีจิสเตอร์ควบคุม ขนาด 5 บิต (CWSR) พื้นที่ภายนอกทั้งสาม(program ,data และ I/O spaces) มีการกำหนดขอบเขต ไว้ในรีจิสเตอร์ software wait-state

Wait-states สำหรับพื้นที่โปรแกรมและข้อมูล ถูกระบุไว้ใน lower และ upper halves ของ PDWSR โดยลำดับ. Wait-states สำหรับพื้นที่ I/O ถูกระบุไว้ใน IOWSR. บิตของ CWSR ควบคุมการแมประหว่างค่าที่อยู่ในรีจิสเตอร์ Wait-state และจำนวนของ Wait-states.

พื้นที่โปรแกรมและข้อมูลแต่ละอย่างจะประกอบด้วย 64K address แต่ละ 64K สามารถถูกมองเห็นเป็นประกอบด้วย 16K-word จำนวน 4 บล็อก แต่ละ 16K address ที่ถูกแบ่งออกในพื้นที่โปรแกรมและข้อมูลถูกทำให้มีความสัมพันธ์กับบริเวณ 2 บิต ใน รีจิสเตอร์ PDWSR , ดังแสดงในตาราง 2.8.9 บริเวณค่าจำนวน 2 บิตใน รีจิสเตอร์ PDWSR จะระบุด้วยจำนวนของค่า Wait-states ที่ถูกใส่สำหรับการเข้าถึงข้อมูล ในพื้นที่ที่กำหนดให้และขอบเขตของ Address

ตาราง 2.8.9 แสดงความสัมพันธ์ระหว่างรีจิสเตอร์ กับ พื้นที่ของการรอ

Register	Bits	Space	Address Range	
PDWSR	0-1	Program	0000h-3FFFh	
	2-3		4000h-7FFFh	
	4-5		8000h-0BFFFh	
	6-7		0C000h-0FFFFh	
	8-9	Data	0000h-3FFFh	
	10-11		4000h-7FFFh	
	12-13		8000h-0BFFFh	
	14-15		0C000h-0FFFFh	
IOWSR		I/O	BIG=0	BIG=1
	0-1		Port 0/1,Port 10/11, etc.	0000h-1FFFh
	2-3		Port 2/3,Port 12/13, etc.	2000h-3FFFh
	4-5		Port 4/5,Port 14/15, etc.	4000h-5FFFh
	6-7		Port 6/7,Port 16/17, etc.	6000h-7FFFh
	8-9		Port 8/9,Port 18/19, etc.	8000h-9FFFh
	10-11		Port 0A/0B,Port 1A/1B, etc.	0A000h-0BFFFh
	12-13		Port 0C/0D,Port 1C/1D, etc.	0C000h-0DFFFh
	14-15		Port 0E/0F,Port 1E/1F, etc.	0E000h-0FFFFh

รีจิสเตอร์ I/O space wait-state (IOWSR) สามารถถูกแมปในที่อื่นๆของ 2 ทาง,เป็นการระบุโดย บิต BIG ในรีจิสเตอร์ CWSR , ถ้าบิต BIG = 0 , แต่ละเส้นของ 8 เส้นของ Memory-mapped I/O ports จะมีการกำหนดขอบเขตใน 2 บิต ใน รีจิสเตอร์ IOWSR

Note. นั่นแน่แค่เมื่อ $BIG = 0$, พื้นที่ I/O ทั้งหมดถูกทำให้มีรูปแบบเข้ากับ Wait-state บนบริเวณ 2 – Word (เช่น port 0/1, port 10/11, และ port 20/21 ทั้งหมดนี้จะมีตัวเลขของ Wait state แบบเดียวกัน) รูปแบบนี้จัดให้ได้ค่าสูงสุดเมื่อ I/O bus cycles มีการเข้าถึงข้อมูลภายนอก เช่น อุปกรณ์ D/A และ A/D อย่างไรก็ตามถ้าอุปกรณ์ I/O ที่ใช้อ่าน และ/หรือ เขียน นั้นเป็น address (เช่น RAM ภายนอก), บิต BIG สามารถถูกเซ็ตไปที่ 1. ในกรณีนี้พื้นที่ I/O ขนาด 64 K จะสามารถแบ่งเป็น 8 address บล็อกๆละ 8K-word และแต่ละบล็อกจะมีอิสระในการโปรแกรม ค่าตัวเลขของ Wait-state

Note. คำกำเนิด Wait-state นั้นมีผลต่อการเข้าถึงข้อมูลภายนอกเพียงอย่างเดียว, การเข้าถึงข้อมูลภายในจะมีค่า Wait-state เป็น "0" เสมอ

ค่า 4 บิตใน รีจิสเตอร์ CWSR จะยอมให้ผู้ใดเลือกหนึ่งใน สอง ของการแอมระหว่าง 2 บิต ของ Wait-state และ ตัวเลขของ Wait-state สำหรับพื้นที่ที่ตรงกัน ดังแสดงในตาราง 2.8.10 ถ้าบิตของ CWSR เป็น "0" การแอมระหว่างค่าขอบเขตของ Wait-state และตัวเลขที่เป็นผลของ Wait-state จะเป็น ไปโดย ตรง. ตัวเลขของ Wait-state สำหรับการเข้าถึงข้อมูลภายนอกในพื้นที่จะสัมพันธ์กับบิตควบคุมซึ่งมีค่าเท่ากับค่าขอบเขตของ Wait-state ถ้าบิตควบคุมของ รีจิสเตอร์ CWSR เป็น "1" ค่าตัวเลขของ Wait-state จะ ถูกกำหนด โดยการแอมดังแสดงในตาราง 2.8.10

ตาราง 2.8.11 แสดง แบบของ รีจิสเตอร์ CWSR ในรีจิสเตอร์ PDWSR และ IOWSR คุณควร โปรแกรมรีจิสเตอร์ CWSR ก่อนที่จะกำหนดรูปแบบที่ รีจิสเตอร์ PDWSR และ IOWSR ที่ควรหลีกเลี่ยง รูปแบบหน่วยความจำกับค่า Wait-state เล็กน้อยในระหว่างการติดตั้งค่าของรีจิสเตอร์ Wait-state

ตาราง 2.8.10 Wait-State Field Values and Wait States as a Function of CWSR Bit n

Wait-State Field of PDWSR or IOWSR (Binary Value)	No. of Wait States (CWSR Bit n=0)	No. of Wait-States (CWSR Bit n=1)
00	0	0
01	1	1
10	2	3
11	3	7

ตาราง 2.8.11 Space Controlled by CWSR Bit n

N(Bit Position in CWSR)	Space
0	Program
1	Data
2	I/O(lower-half:Port 0-Port 7 if $BIG=0$, 0000h-7FFFh if $BIG=1$)
3	I/O(upper-half:Port 8-Port F if $BIG=0$, 8000h-0FFFFh if $BIG=1$)
4	BIG mode bit

2.9 ส่วนแปลงสัญญาณอนาลอกเป็นดิจิทัล

ส่วนแปลงสัญญาณอนาลอกเป็นดิจิทัลนี้จะป็นอุปกรณ์ที่ใช้บน Port Header ซึ่งจะเป็นพอร์ตรับข้อมูลเข้าอย่างหนึ่งของระบบประมวลผล

เราจะเลือกใช้ ADC เบอร์ ADC908 ซึ่งมีคอนเวอร์ชันไทม์(Conversion Time) เท่ากับ 6 ms ข้อมูล 8 บิต สำหรับเหตุผลที่เลือกใช้ ADC เบอร์นี้ ก็คือการทำงานของ TMS นั้นมีความเร็วสูงมาก ดังนั้นจึงต้องการ ADC ที่ทำงานเร็วทัน และการเลือกใช้ ADC ที่มีข้อมูล 16 บิต นั้นจะมีราคาแพง

จากสัญญาณอินพุตซึ่งมีความถี่สูง จึงต้องมีวงจรส่วนแซมเปิล แอนด์ โฮลด์ (Sample and Hold) ซึ่งจะใช้ IC LF398 ซึ่งจากค่า บั๊ก ของ LF398 เราจะสามารถเลือกค่าตัวเก็บประจุที่จะใช้ได้จากกราฟ Acquisition Time ในค่า บั๊ก จะ ได้ค่า ตัวเก็บประจุที่ต้องใช้เท่ากับ 1 นาโนฟารัด เพื่อความเร็วที่สูงที่สุด

สำหรับสัญญาณที่จะควบคุม LF398 ว่าจะให้ แซมเปิล หรือ โฮลด์ ก็นำมาจากขาสัญญาณ ~BUSY ของ ADC908 ซึ่งจากการทำงานของ ADC908 นี้ เมื่อ ADC กำลังทำการแปลงสัญญาณอยู่ที่ขา ~BUSY นี้จะมีลอจิก "0" ซึ่งในช่วงนี้ LF398 ก็จะต้องโฮลด์ข้อมูลไว้ จะเห็นได้ว่าสัญญาณที่จะควบคุม LF398 ก็ต้องการลอจิก "0" เช่นเดียวกัน ดังนั้นจึงสามารถต่อขาสัญญาณที่จะควบคุม LF398 เข้ากับขา ~BUSY ของ ADC908 ได้โดยตรง

ADC908 นี้จะมีสัญญาณอินพุตอยู่ 2 โหมด คือ โหมดยูนิโพลาร์(Unipolar Mode) และ โหมดไบโพลาร์(Bipolar Mode) ในโครงงานนี้เราจะเลือกการใช้งานในโหมดไบโพลาร์เพื่อให้สัญญาณอินพุตอนาลอกสามารถเป็นได้ทั้ง บวก และลบ มีขนาดสัญญาณไม่เกิน + 10 โวลต์และ -10 โวลต์ ดังนั้นจะต้องใช้แรงดันไฟเบรียบเทียบ(VREF) ขนาด -10 โวลต์ซึ่งสามารถสร้างได้โดยใช้ไอซี LM337 ซึ่งเป็นไอซีที่สามารถปรับค่าแรงดันได้ และสำหรับสัญญาณ BOFS เพื่อทำการปรับออฟเซ็ทของ ADC908

จากการทำงานของ ADC908 ซึ่งจะมีการทำงานอยู่ 3 โหมด คือ

- โหมดการทำงานแบบ RAM
- โหมดการทำงานแบบ ROM
- โหมดการทำงานแบบ Slow - Memory

ตารางการทำงาน ของทั้ง 3 โหมด แสดงดังตารางที่ 2, 3 และ 4

ตารางที่ 2.9.1 แสดงการทำงานของ ADC908 ในโหมด RAM

อินพุต		เอาต์พุต		การทำงาน
~CS	~RD	~BUSY	บัคข้อมูล	
L	H	H	Z	เริ่มทำการแปลงข้อมูล
L	H-L	H	Z-D	อ่านข้อมูล
L	L-H	H	D-Z	รีเซทตัวแปลงข้อมูล
H	X	X	Z	ไม่ใช่
L	H	L	Z	ไม่ใช่
L	H-L	L	Z	ไม่ใช่
L	L-H	L	Z	ไม่อนุญาตให้เกิดขึ้น

ตารางที่ 2.9.2 แสดงการทำงานของ ADC908 ในโหมด ROM

อินพุต		เอาต์พุต		การทำงาน
\sim CS	\sim RD	\sim BUSY	บัสข้อมูล	
L	H-L	H	Z-D	อ่านข้อมูล รีเซ็ตและเริ่มแปลงข้อมูล ไม่ใช่ ไม่อนุญาตให้เกิดขึ้น
L	L-H	H-L	D-Z	
L	H-L	L	Z	
L	L-H	L	Z	

ตารางที่ 2.9.3 แสดงการทำงานของ ADC908 ในโหมด Slow-Memory

อินพุต		เอาต์พุต		การทำงาน
\sim CS	\sim RD	\sim BUSY	บัสข้อมูล	
H	H	H	Z	ไม่ใช่
H-L	H-L	H-L	Z	เริ่มทำการแปลงข้อมูล
L	L	L	Z	รอการแปลงข้อมูล
L	L	L-H	Z-D	แปลงข้อมูลเสร็จ, อ่านข้อมูลได้
L-H	L-H	H	D-Z	รีเซ็ตและยกเลิกการใช้

(Z : High Impedance , H-L : ขอบขาลง , L-H : ขอบขาขึ้น , X : ไม่สนใจ , D : มีข้อมูล)

จากตารางการทำงานจะเห็นได้ว่า ในโหมด RAM นั้น จะต้องมีสัญญาณในการควบคุม ADC908 มาก ซึ่งยุ่งยากในการออกแบบวงจร และในโหมด Slow-Memory ก็จะทำให้ TMS เสียเวลาออกการทำงานของ ADC908 ในช่วงของการแปลงสัญญาณ แต่การใช้งาน ADC908 ในโหมด ROM นั้นดูจะมีความเหมาะสมมากที่สุด

การที่จะใช้งาน ADC908 ในโหมด ROM นั้นก็เพียงแต่ป้อนลอจิก "0" เข้าที่ขา \sim CS และเมื่อเราต้องการจะอ่านข้อมูลจาก ADC908 ก็ป้อนลอจิก "0" เข้าที่ขา \sim RD หลังจากที่เราทำการอ่านข้อมูลเสร็จแล้ว ADC908 ก็จะทำการแปลงข้อมูลชุดต่อไป

จาก ADC908 ที่ข้อมูล 8 บิต จะต้องอินเทอร์เฟสกับ ค่าคำบัสของ TMS ที่เป็นแบบ 16 บิต เราจะกำหนดให้ข้อมูล 8 บิตล่างมีค่าเท่ากับข้อมูลที่ได้จาก ADC908 ส่วนข้อมูล 8 บิตบนจะมีค่าเท่ากับ "00000000" โดยใช้ 74LS244 ที่ต่อขาอินพุตทั้งหมดลงกราวด์

และจาก Timing diagram การทำงานของ ADC908 ในโหมด ROM และจากข้อมูลในคำคำบัสของ ADC908 จะเห็นได้ว่าเมื่อเราทำการอ่านข้อมูลจาก ADC908 แล้วเลิกอ่านข้อมูลนั้นจะยังคงค้างอยู่ที่

ขาต้านของ ADC908 อีกช่วงระยะเวลาหนึ่ง ซึ่งระยะเวลานี้มีค่านานเกินไปสำหรับขาต้านของ TMS ดังนั้นจึงต้องใช้ 74F373 มาทำการเชื่อมต่อระหว่างขาต้านของ ADC908 กับขาต้านของ TMS 8 บิตล่าง เมื่อรวมส่วนต่างๆ ดังที่กล่าวมาข้างต้น ก็จะได้อะไหล่ของส่วนแปลงสัญญาณ ดังรูปในภาคผนวก ก ซึ่งสามารถอธิบายการทำงานได้ดังนี้

- เมื่อเกิดการรีเซ็ต D Flip Flop จะถูกเคลียร์ ($Q=0$, $\sim Q=1$) ซึ่งจะทำให้ ADC อยู่ในสภาวะอ่านข้อมูลสัญญาณ $\sim BUSY$ จะเป็น "1" และจาก $\sim Q=0$ 74LS373 ก็จะรับข้อมูลมาจาก ADC
- เมื่อมีสัญญาณอ่านพอร์ท ($\sim PS$) เข้ามา เมื่อ $\sim PS$ เป็นศูนย์ 74AS74 จะถูกปริเซ็ต (PRESET) $Q=1$, $\sim Q=0$ เมื่อ Q เปลี่ยนจาก "0" เป็น "1" ADC ก็จะเริ่มทำการแปลงข้อมูล สัญญาณ $\sim BUSY$ จะเปลี่ยนจาก "1" เป็น "0" ทำให้ LF398 โสตสัญญาณอนาล็อกไว้ และจาก $\sim Q=0$ ดังนั้น 74LS373 ก็จะไม่รับข้อมูลจาก ADC เมื่อการแปลงข้อมูลเสร็จสิ้น สัญญาณ $\sim BUSY$ จะเปลี่ยนจาก "0" เป็น "1" ทำให้ 74AS74 รับข้อมูลจากขา D มาออกที่ขา Q ซึ่งข้อมูลที่ D="0" ดังนั้น $Q="0"$, $\sim Q="1"$ ซึ่งก็จะเป็นการอ่านข้อมูล ข้อมูลจึงถูกส่งไปเก็บใน 74LS373 แล้วก็คอยการอ่านเข้าไปประมวลผลครั้งต่อไป

2.10 หลักการทั่วไปของ Acoustic Echo Canceler ที่ใช้ อัลกอริทึม FBAF

ในระบบการสื่อสารทางไกลนั้นเสียงสะท้อนนับเป็นปัญหาที่สำคัญอย่างยิ่งในการสื่อสารเนื่องจากจะมีผลต่อคุณภาพในการรับส่งสัญญาณ กระบวนการในการกำจัดสัญญาณเสียงสะท้อนจะต้องอาศัยกระบวนการทางด้านการประมวลผลทางดิจิทัลซึ่งจะมีวิธีการในการกำจัดเสียงสะท้อนโดยอาศัยวงจรฟิลเตอร์ที่สามารถปรับค่าได้เองอย่างอัตโนมัติ (Adaptive Filter) และเราสามารถพิจารณาการทำงานของวงจรได้ทั้งด้านโดเมนทางเวลา (Time Domain Adaptive Filter) และด้านโดเมนทางความถี่ (Frequency Domain Adaptive Filter)

-Frequency Domain Adaptive Filter

จะอาศัยวงจรฟิลเตอร์ชนิด ทรานสเวอร์ซอลฟิลเตอร์ (FIR Transversal Filter) ในการใช้เป็นตัวช่วยในการพิจารณาทางด้านโดเมนทางเวลา ซึ่งจะอาศัยอัลกอริทึมของค่าประมาณแบบกำลังสองต่ำสุด (Least Mean Square: LMS Algorithm) และเราสามารถที่จะแปลงสัญญาณที่อยู่ในเทอมของเวลาให้อยู่ในเทอมทางความถี่ได้โดยอาศัย DFT (Discrete Fourier Transform) เมื่อผ่านกระบวนการแล้วเสร็จจึงแปลงกลับให้อยู่ในเทอมทางเวลา ซึ่งการพิจารณาทางด้านความถี่จะมีประโยชน์อยู่ 2 ประการคือ

1. สามารถที่จะใช้ การแปลงฟูเรียร์อย่างรวดเร็วหรือ FFT (Fast Fourier Transform) เพื่อลดการคำนวณในส่วนการคูณค่าสัมประสิทธิ์ของการฟิลเตอร์เมื่อเทียบกับการใช้ทรานสเวอร์ซอลอะแด็ปทีฟฟิลเตอร์ทางด้านโดเมนทางเวลา ในทางปฏิบัตินั้นถ้าข้อมูลถูกจัดแบ่งเป็นบล็อกที่มีขนาด L จุดในทางด้านโดเมนทางเวลานั้นจะต้องทำการคูณทั้งหมดถึง $2L^2+L$ และในทางด้านโดเมนทางความถี่จะประกอบไปด้วย FFT ขนาด L จุด 3 จุด และจะมีการคูณกันเพียง $2L$ (เมื่อเป็นการคูณเลขเชิงซ้อน)
2. ด้วยทฤษฎีการนอร์มอลไลน์ค่าที่เหมาะสมจะเป็นการกำจัดหรือลดค่าที่ไม่สม่ำเสมอของค่าไอเก้นของสัญญาณ (Eigenvalue)

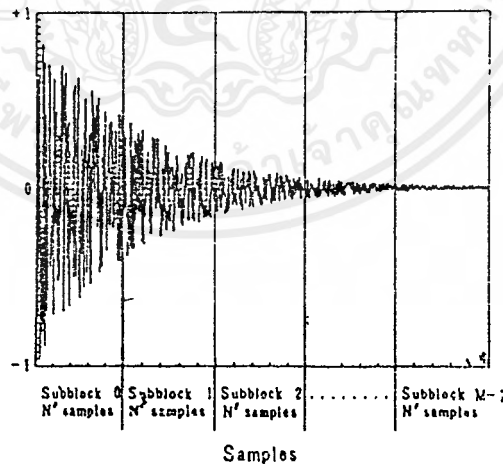
ด้วยการประมวลผลทางด้านความถี่นั้นจะกระทำในลักษณะที่เป็นบล็อกซึ่งเราเรียกว่า

" Block Processing " ซึ่งจะกระทำในลักษณะบล็อกต่อบล็อกและจะทำการปรับค่าของสัมประสิทธิ์ของการฟิลเตอร์จะมีการเปลี่ยนแปลงเพียงหนึ่งครั้งต่อบล็อกข้อมูลขนาด L จุด

และเนื่องจากผลของ Circular Convolution Effect ซึ่งเป็นผลมาจากการใช้ DFT เพื่อให้ผลลัพธ์ที่ได้มีความถูกต้องตรงกับค่าที่เกิดจาก Linear Convolution ในด้านโดเมนทางเวลา ซึ่งจะต้องอาศัยการเติมศูนย์ (zero-padded) ซึ่งอยู่ภายในกระบวนการของ Overlap-save หรือ Overlap-add

สัญญาณอะคูสติกที่ป้อนกลับโดยมีเส้นทางเดินของเสียงจากลำโพงไปยัง ไมโคร โฟนภายในห้องส่ง จะทำให้เกิดเสียงสะท้อนเกิดขึ้นซึ่งจะมีค่าน้อยขึ้นอยู่กับขนาดของห้องและองค์ประกอบทางด้านอะคูสติกของอุปกรณ์ต่างที่อยู่ภายในห้อง และสามารถมีการเปลี่ยนแปลงได้โดยง่ายเช่น การเปิด/หน้าต่างภายในห้อง, การเคลื่อนไหวของคนที่อยู่ภายในห้อง เป็นต้น ดังนั้นจึงต้องมีการใช้ตัวควบคุมเสียงสะท้อนที่สามารถปรับค่าได้เองอย่างอัตโนมัติเพื่อให้มีความยืดหยุ่นในการทำงานและจะต้องสามารถทำงานได้อย่างต่อเนื่องและเหมาะสมเมื่อเส้นทางเดินของสัญญาณมีการเปลี่ยนแปลงและเนื่องจากความเหมาะสมในด้านการวิเคราะห์ทางด้านความถี่จะทำให้ลดการคำนวณลงไปได้มากเมื่อมีการใช้ สัมประสิทธิ์ของการห้วงจำนวนมากๆ เช่น หลายร้อย/หลายพัน Tap-Coefficients เพื่อจำลองเสียงสะท้อนที่เกิดขึ้นจริงในระบบและเมื่อนำเอากระบวนการทางบล็อกก็สามารถทำให้การห้วงเวลาในการประมวลผลนั้นมีค่าน้อยมากโดยปราศจากผลกระทบต่อการคำนวณค่าสัมประสิทธิ์ โดยเราเรียก อัลกอริทึมใหม่นี้ว่า Frequency Bin Adaptive Filter โดยแต่ละส่วนของ Frequency Bin จะถูกนำเข้าสู่กระบวนการของ FIR Filter แยกอิสระจากกัน โดยจะให้ค่า Weight แต่ละ Tap เหมือนกับใน FDAF อัลกอริทึม

ที่มาของ FBAF อัลกอริทึม



รูปที่ 2.10.1 การแบ่งผลการตอบสนองทางอิมพัลส์

สมมติว่าความยาวของผลการตอบสนองทางอิมพัลส์ของห้อง (Acoustic Impulse Response) มี N Sample เราจะแบ่งผลการตอบสนองทางอิมพัลส์ของห้องนี้เป็น M บล็อกย่อย และในแต่ละบล็อกจะมี N' Sample นั่นคือ $N = M \cdot N'$ เมื่อ M เป็นจำนวนเต็มใดๆ ดังแสดงในรูปที่ 2.2.1

ค่าเอาต์พุตคือ $y(.)$ ของ Echo Canceller จะเป็นการผลของการทำการผลการประสาน(Convolution) ระหว่างสัญญาณอินพุต $x(.)$ และค่าของตัวถ่วงน้ำหนัก $w(.)$ ซึ่งสามารถเขียนเป็นสมการ ได้ดังนี้

$$y(kN'+r) = \sum_{m=0}^{M-1} \sum_{i=0}^{N'-1} w_m^i(k) \cdot x(kN'+r-mN'-i) \tag{2.10.1}$$

เมื่อ k เป็นตัวเลขแสดงตำแหน่งของบล็อกย่อยๆ ซึ่งภายในบล็อกจะประกอบด้วย N' Sample
 r เป็นตัวเลขแสดงตำแหน่งของ Sample ที่อยู่ในแต่ละบล็อกย่อยๆ
 ส่วนค่า m ตัวยกด้านบนและค่า i ตัวที่ห้อยท้ายไว้ด้านล่างของตัววงน้ำหนัก แสดงถึงค่าตัวที่ i
 ในบล็อกย่อยที่ m

ต่อมาเมื่อพิจารณาในด้าน โดเมนทางความถี่โดยใช้ FFT ทั้ง 2 ข้างของสมการ

$$\text{FFT}[y(kN'+r)] = \sum_{m=0}^{M-1} \text{FFT} \left[\sum_{i=0}^{N'-1} w_m^i(k) \cdot x(kN'+r-mN'-i) \right] \tag{2.10.2}$$

และเพื่อแก้ปัญหาการเกิด Circular Convolution Effect จึงได้มีการแก้ปัญหาโดยการใช่วิธี
 Overlap-Save ซึ่งจะกระทำในส่วนการใช้ FFT กับ ตัววงน้ำหนักนั้นเราจะต้องทำการเติมศูนย์เข้าไปเป็น
 จำนวน N' ตัว ของแต่ละบล็อกย่อยๆ เพื่อที่จะลดผลกระทบอันนี้ เมื่อแปลงค่าของการวงน้ำหนักไป
 พิจารณาด้าน โดเมนทางความถี่เราจะได้

$$W_m^T(k) = \text{FFT} [w_m^0(k), w_m^1(k), \dots, w_m^{N'-1}(k), 0, 0, \dots, 0] \tag{2.10.3}$$

$$X_m^T(k) = \text{FFT} [x(kN' - N' - mN'), x(kN' - (N'-1) - mN'), \dots, x(kN' - 1 - mN'), \\ x(kN' - mN'), \dots, x(kN' + N' - 1 - mN')] \tag{2.10.4}$$

และตัว T ยกด้านบน แสดงถึงการทรานสโพอส

เอาท์พุท $y(kN' + r)$ ในสมการที่ 2.10.1 ซึ่งเป็นผลรวมของผลการประสาน (Convolution)
 ระหว่างค่าสัมประสิทธิ์ของการ Tap กับสัญญาณอินพุทในแต่ละบล็อกย่อยๆ แล้วนำเอาส่วนที่เป็นการ
 คูณของแต่ละส่วน(Element-by-Element Multiplication (\otimes)) มาเข้า IFFT

$$[y(kN'), y(kN' + 1), \dots, y(kN' + N' - 1)]^T = \sum_{m=0}^{M-1} W_m(k) \otimes X_m(k) \tag{2.10.5}$$

จากสมการที่ 2.10.5 จะเลือกเอาเฉพาะในส่วน N' เทอมหลังเท่านั้น และจะเห็นได้ว่าเอาท์พุทเพียง
 N' Sample จะถูกคำนวณโดย IFFT และค่าของการหน่วงเวลาในการส่งเอาท์พุทออกไปจะถูกหน่วงเอา
 ไว้เพียง N' Sample เท่านั้น สำหรับการปรับค่าของสัมประสิทธิ์ของการเทปเราสามารถ
 ที่จะแสดงในแต่ละบล็อกย่อยๆ ได้ดังนี้

$$w_m^i(k+1) = w_m^i(k) - 2\mu \sum_{r=0}^{N'-1} e(kN'+r)x(kN'+r-i-mN') \tag{2.10.6}$$

เมื่อ $i = 0, 1, \dots, N-1$ (แสดงองค์ประกอบตัวที่ i ในบล็อกย่อยที่ m)

μ = เฟคเตอร์ของค่าการถ่วงน้ำหนัก

$e(.)$ = สัญญาณผลต่างผิดเพี้ยน (Error Signal) ซึ่งได้มาจาก

$$e(kN' + r) = y(kN' + r) - d(kN' + r) \quad (2.10.7)$$

เมื่อ $d(.)$ เป็นสัญญาณที่ได้มาจากการทำงานของระบบที่ได้ออกแบบ

จากสมการที่ (2.10.6) ในส่วนของการทำคอร์ริเลชัน (Correlation) ระหว่างสัญญาณผลต่างผิดเพี้ยนกับสัญญาณอินพุท ซึ่งจะแสดงโดยส่วนของเกรเดียนต์ $\nabla_m(k)$ ซึ่งสามารถเป็นจริงได้โดยการใช้การแปลงทางฟูเรียร์ดังนี้

$$\nabla_m(k) = \text{IFFT} [E(k) \otimes X_m^*(k)] \quad (2.10.8)$$

จากสมการที่ (2.10.8) เราจะเลือกเอาผลลัพธ์ที่ได้เพียง N' Sample ในส่วนทอมแรกเท่านั้น ซึ่งค่าผลต่างผิดเพี้ยนที่ทำการทราสฟอร์มแล้วสามารถแสดงให้เห็นได้ดังนี้

$$E^T(k) = \text{FFT} [0, 0, \dots, 0, e(kN'), e(kN'+1), \dots, e(kN'+N'-1)] \quad (2.10.9)$$

| ← N' → |

ดังนั้นเวกเตอร์ของตัวถ่วงน้ำหนักของบล็อกที่ m ซึ่งได้มีการปรับโดยการเติมศูนย์ลงไปรวมกับส่วนของเกรเดียนต์และมีการแปลงให้อยู่ในโดเมนทางความถี่ได้ดังนี้

$$W_m^T(k+1) = W_m^T(k) - 2\mu \text{FFT} [\nabla_m(k), 0, 0, \dots, 0] \quad (2.10.10)$$

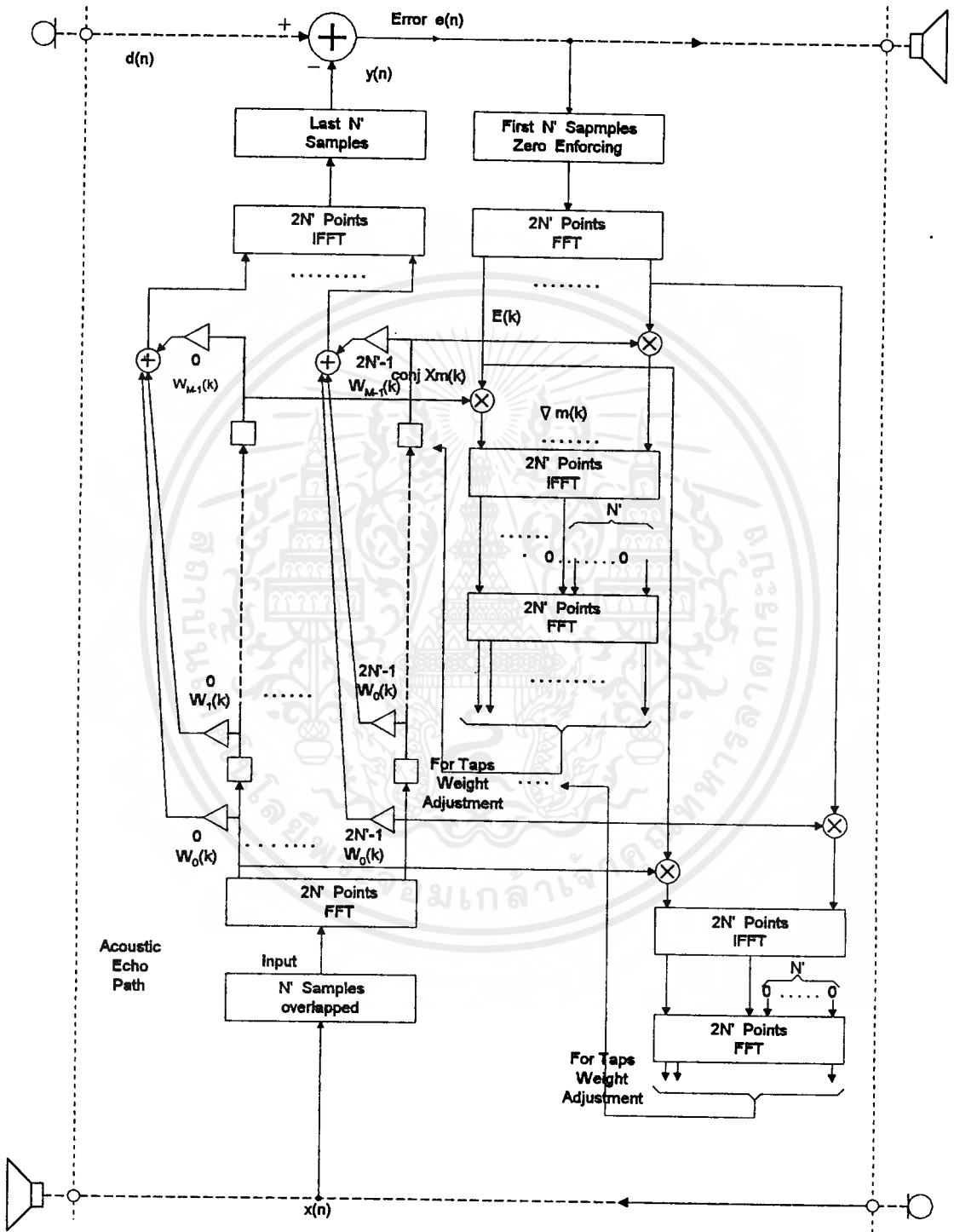
| ← N' → |

เราสามารถที่จะสรุปการทำงานของวงจรได้ดังรูปที่ 2.10.2

สัญญาณอินพุท $x(kN' + r - i - mN')$ ในแต่ละบล็อกย่อยจะมีอยู่ N' Sample ที่มีค่าเหลื่อมซ้อนกันทับกัน ตามกระบวนการ Overlap-Save ดังนั้นในโดเมนทางความถี่บล็อกสัญญาณที่ต่อเนื่องกันจึงถูกแบ่งกระบวนการออกจากกันด้วยค่าการหน่วงเวลาค่าหนึ่งในแต่ละ Frequency Bin ดังรูปที่ 2.10.2

ดังนั้นในแต่ละบล็อกย่อยเราจะเลือกค่า N' ที่มีขนาดเล็กๆ แบ่งออกเป็น M บล็อกย่อยๆ เมื่อเราทราบค่าความยาวทั้งหมดเป็น N Sample ดังนั้นเฟคเตอร์ของการต่อเข้าจะถูกจำกัดอยู่ที่ค่า $1/N$ เท่าของส่วนกลับของค่าไอเก้นสูงสุด (λ_{\max}) ที่ได้มาจากการทำคอร์ริเลชันระหว่างสัญญาณอินพุทกับตัวของมันเอง

$$0 < \mu < (1/N)(1/\lambda_{\max}) \quad (2.10.11)$$



รูปที่ 2.10.2 อัลกอริทึม แบบ FBAF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 การกรองความถี่ที่มีลำดับของสัญญาณยาวๆ

ในทางปฏิบัติการใช้งานทำให้ต้องใช้ตัวกรองแบบ Linear Filter เมื่อลำดับของสัญญาณอินพุต $x(n)$ จะมีลำดับที่มีความยาวมากๆ ซึ่งจะเห็นได้ในงานที่ใช้แบบเวลาจริง (Real Time Signal Processing) ซึ่งจะมีความเกี่ยวข้องกับทฤษฎีการมอดูเลชันและการวิเคราะห์สัญญาณ

ด้วยเหตุที่ Linear Filter กระทำผ่าน DFT ทำให้ต้องกระทำบน Block ของข้อมูลซึ่งจำเป็นที่ต้องกำหนดขนาดของหน่วยความจำ เมื่อลำดับของสัญญาณยาวจะต้องจัดแบ่งออกเป็นบล็อกย่อยๆ ที่มีขนาดของบล็อกที่แน่นอน ก่อนที่จะเข้าสู่กระบวนการอื่นต่อไป เมื่อการฟิลเตอร์เป็นชนิดเชิงเส้นสัญญาณที่ถูกแบ่งออกเป็นบล็อกย่อยๆ ที่มีความต่อเนื่องกันจะถูกดำเนินการอันเดียวกันโดยผ่านทาง DFT และ Output ของบล็อกถูกทำให้เหมาะสมในรูปของสัญญาณที่เป็นซีแควน

มีอยู่ 2 กระบวนการที่ใช้งานกับ Linear FIR Filtering ที่มีลำดับของสัญญาณยาวๆ บนมาตรฐานในการกระทำแบบบล็อกต่อบล็อก โดยการใช้ DFT สัญญาณอินพุตในแต่ละส่วนที่อยู่ภายในบล็อกจะถูกดำเนินการ โดย DFT และ IDFT เพื่อผลิตสัญญาณออกมาเป็นบล็อก

(Block Output Data) เมื่อนำเอาผลลัพธ์ทั้งหมดที่ได้มาเป็นบล็อกมาจัดทำเป็นสัญญาณที่มีลำดับที่ถูกต้องการจะได้สัญญาณซึ่งเหมือนกันกับที่ได้มาจากการกระทำการผลคูณประสานของสัญญาณที่มีความยาวมากๆ ที่ถูกกระทำโดยกระบวนการทางเวลา Time - Domain Convolution

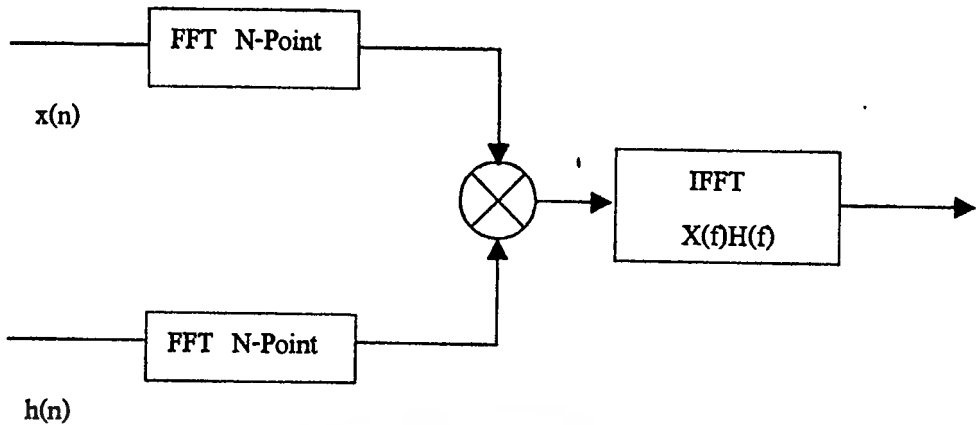
เนื่องจากกระบวนการของ Circular Convolution มีการนำเอาผลลัพธ์ที่ได้มาจากการดำเนินการ DFT ที่ทำกับสัญญาณอินพุต 2 สัญญาณ นำมาเข้าสู่กระบวนการกระทำ IDFT ซึ่งจะให้เกิดผลที่ผิดพลาดเกิดขึ้นเราเรียกผลอันนั้นว่า "Circular Effect" เพื่อแก้ผลกระทบดังกล่าวเพื่อให้ได้ผลลัพธ์เหมือนกันในผลการประสานชนิดเชิงเส้น (Linear Convolution) ซึ่งจำเป็นที่จะต้องทำการเพิ่มเติม "0" กับสัญญาณที่รับเข้ามา

FIR Filter จะทำงานเป็น Linear Convolution ระหว่าง Filter Impulse Response กับลำดับของสัญญาณที่ค่าจำกัด สำหรับการประยุกต์ใช้งานหลายประเภทลำดับของสัญญาณอินพุตที่มีความยาวมากๆ ในระหว่างการเปรียบเทียบกับ Filter Impulse Response ดังนั้นในการกระทำกระบวนการในทางด้าน Frequency Domain จะใช้ FFT และ IFFT โดยลำดับของสัญญาณอินพุตจะถูกแบ่งออกเป็นส่วนๆ โดยมีขนาดที่ขึ้นอยู่กับการทำงานของ Filtering Response และ Hardware/Software ที่ใช้ทำกระบวนการ FFT ของ Impulse Response สามารถที่จะกระทำกระบวนการคำนวณและเก็บข้อมูลไว้ก่อน (Precalculated and Stored) ซึ่งกระบวนการของ Fast Convolution จะเป็นดังรูปที่ 2.11.1

มีอยู่ 2 วิธีที่ใช้งานในทางปฏิบัติคือ

1. Overlap-Save Method
2. Overlap-Add Method

โดยเราสามารถว่า FIR Filter อยู่ระหว่าง M (ข้อมูลทางอินพุตที่เป็นซีแควน) ในแต่ละส่วนในบล็อกจะมีอยู่ L จุด โดยเราสามารถให้ $L \gg M$ โดยปราศจาก Loss ที่มีเกิดขึ้นจริง



รูปที่ 2.11.1 กระบวนการของ Fast Convolution

Overlap-Save Method

ขนาดของสัญญาณอินพุตที่ถูกจัดเป็นบล็อกๆจะมีขนาดเป็น $N = L + M - 1$ และขนาดของ DFT และ IDFT จะมีความยาวเป็น N ด้วย ในแต่ละบล็อกจะประกอบด้วยส่วนท้ายที่มีขนาดเป็น $M - 1$ จุดของบล็อกข้อมูลก่อนตามด้วยบล็อกข้อมูลใหม่ที่มีขนาดเป็น L จุด ซึ่งเมื่อรวมกันแล้วเราจะได้บล็อกข้อมูลที่มีความยาวเป็น N และ DFT ที่มีขนาดเป็น N จุดจะใช้ในการคำนวณข้อมูลในแต่ละบล็อก Impulse Response ของ FIR จะมีขนาดความยาวที่เพิ่มขึ้นด้วยโดยการใส่ " 0 " จำนวน $L - 1$ จุดแบบที่ด้านท้าย และ N Point DFT ของลำดับสัญญาณจะถูกคำนวณครั้งหนึ่งและเก็บค่าไว้ การคูณของสัญญาณ 2 ตัวที่มีขนาดเป็น N จุด ใน DFT คือ $\{H(k)\}$ และ $\{X_m(k)\}$ เมื่อ m แสดงถึงบล็อกข้อมูลที่ m

$$Y_m(k) = H(k) X_m(k) \quad (2.11.1)$$

เมื่อ $k = 0, 1, 2, 3, 4, 5, \dots, N - 1$

และที่เอาท์พุทของ N -Point IDFT

$$Y'_m(n) = \{y'_m(0) \ y'_m(1) \ \dots \ y'_m(M-1) \ y'_m(m) \ \dots \ y'_m(N-1)\} \quad (2.11.2)$$

เมื่อพิจารณาข้อมูลที่ปรากฏที่เอาท์พุทที่มีความยาวเป็น N จุด โดยในเทอมแรกคือ $M-1$ จุดซึ่งเป็นข้อมูลทำให้เกิดการเหลื่อมกันเราเรียกว่าเกิดการ Aliasing ซึ่งในจะต้องตัดทิ้งไป (Discard) และส่วนที่สองจะมีขนาด N จุด ซึ่งจะให้ผลลัพธ์เหมือนกับกับผลจากการทำการประสานแบบเชิงเส้น (Linear Convolution) ซึ่งเราสามารถเขียนได้เป็น

$$y'_m(n) = y_m(n) \quad (2.11.3)$$

เมื่อ $n = M, M+1, \dots, N-1$

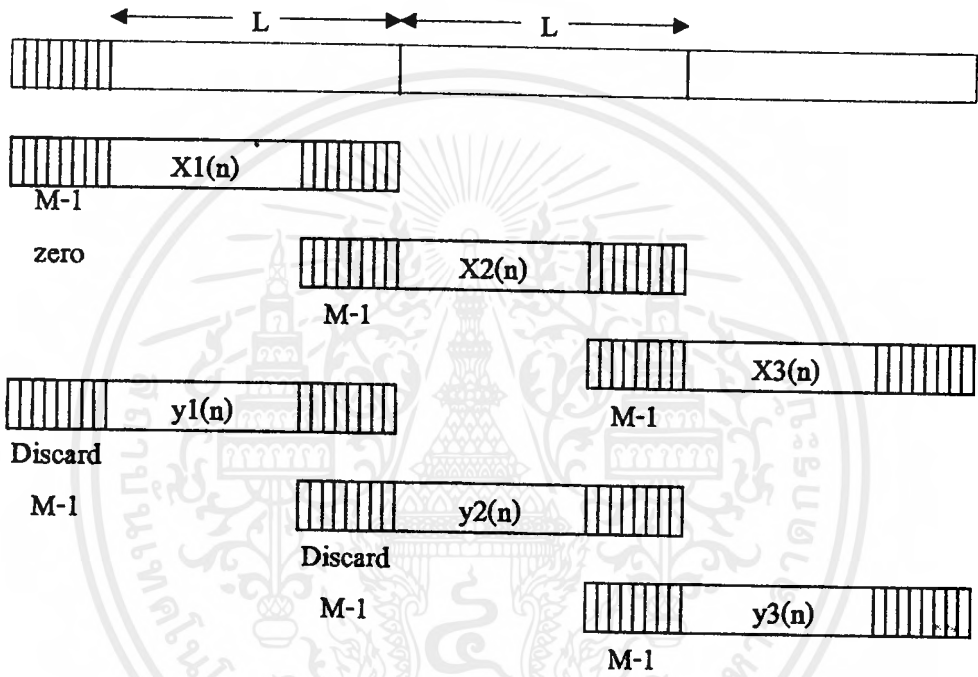
เพื่อหลีกเลี่ยงความผิดพลาดของข้อมูลในระหว่างการเกิด Aliasing จึงต้องทำให้ $M-1$ จุด ในส่วนแรกของแต่ละ Data Record จะถูก save และข้อมูลเหล่านี้จะกลายเป็น $M-1$ จุดในส่วนข้อมูลแรกของ Subsequence Record โดยเมื่อเริ่มต้นกระบวนการเราจะต้องทำการตั้งค่าในส่วนเทอมแรกของ $M-1$ จุดของ First Record ดังนั้นเราสามารถเขียนเป็นบล็อกข้อมูลได้ดังนี้คือ

$$x_1(n) = \{0, 0, 0, 0, \dots, 0, x(0), x(1), \dots, x(L-1)\}$$

$$x_2(n) = \{x(L-M+1), \dots, x(L-1), x(L), \dots, x(2L-1)\}$$

$$x_3(n) = \{x(2L-M+1), \dots, x(2L-1), x(2L), \dots, x(3L-1)\}$$

เมื่อนำเอาบล็อกข้อมูลไปผ่านกระบวนการเราจะได้อะไรที่ออกมาจาก IDFT ได้ผลลัพธ์ออกมาเหมือนกับกับสมการที่ 2.11.2 โดยเทอม $M-1$ จุดแรกจะถูกตัดทิ้งอันเนื่องมาจากการเหลื่อมกันของข้อมูลในเทอมที่เหลือนซึ่งมีขนาด L จุด ที่ต้องการอันเป็นผลมาจาก Linear Convolution โดยการแบ่งอินพุตเช่นนี้และการปรับปรุงค่าของข้อมูลทางด้านเข้าที่ทุกตัวพร้อมกันซึ่งเราสามารถแสดงได้ดังรูป



รูปที่ 2.11.2 แสดง Overlap- Save Method

เราสามารถอธิบายกระบวนการทำงานได้ดังนี้

1. คุณสมบัติของ N-Point FFT ของ FIR Impulse Response $h(n)$

$$h(n) = h(n) \quad \text{เมื่อ } n = 0, 1, 2, \dots, M-1$$

$$= 0 \quad \text{เมื่อ } n = M, M+1, \dots, N-1$$

2. เลือก N จุด จากลำดับของสัญญาณอินพุตเป็นบล็อกข้อมูลได้ดังนี้

$$x_m(n) = x\{n + (m-1)(N-M+1)\}$$

เมื่อ $n = 0, 1, \dots, N-1$

$m = 1, 2, 3, 4, \dots$

$N =$ ความยาวของบล็อกข้อมูลใน FFT

$M =$ Impulse Response Length

3. ทำการแปลงสัญญาณจากโดเมนทางเวลามาเป็นโดเมนทางความถี่โดยการใช้ FFT ที่มีขนาด N จุด
4. ทำการคูณสัญญาณทั้งสองที่ผ่านการแปลงมาอยู่ในโดเมนทางความถี่ ของสัญญาณอินพุตในบล็อกที่ m
5. นำเอาข้อมูลที่ได้จาก Step ที่ 4 มาผ่าน IFFT ขนาด N จุด ซึ่งจะเป็นการแปลงสัญญาณโดเมนทางความถี่มาเป็นโดเมนทางเวลา
6. คัดส่วนเทอมแรกขนาด $M-1$ จุด จากแต่ละเอาต์พุตของทุกๆบล็อกที่ได้มาจากกระบวนการ IFFT ทำให้ได้ลำดับของสัญญาณเอาต์พุตดังนี้คือ

$$\begin{aligned}
 & y_1(n) \\
 & y_2\{n - (N-M+1)\} \\
 & y_3\{n - 2(N-M+1)\} \\
 & \vdots \\
 & \vdots \\
 & y_m\{n - (m-1)(N-M+1)\}
 \end{aligned}$$

สุดท้ายเราจะได้

$$y(n) = y_1(n), y_2(n), y_3(n), \dots, y_m(n), \dots$$

Overlap-Add Method

ขนาดของข้อมูลทางอินพุตจะถูกแบ่งออกเป็นบล็อกข้อมูลที่มีขนาดความยาวเป็น L จุด และขนาดของ DFT และ IDFT จะค่าเป็น N จุด ซึ่ง $N = L + M - 1$ โดยแต่ละบล็อกของข้อมูลจะทำการใส่ "0" เข้าไปเป็นจำนวน $M-1$ จุด ไว้ในเทอมสุดท้ายของบล็อกข้อมูลอินพุต ซึ่งเราสามารถเขียนได้เป็น

$$\begin{aligned}
 x_1(n) &= \{x(0), x(1), \dots, x(L-1), 0, 0, \dots, 0, 0\} \\
 x_2(n) &= \{x(L), x(L+1), \dots, x(2L-1), 0, 0, \dots, 0, 0\} \\
 x_3(n) &= \{x(2L), x(2L+1), \dots, x(3L-1), 0, 0, \dots, 0, 0\}
 \end{aligned}$$

และเอาต์พุตที่ได้จากการคูณกันโดยการกระทำผ่านทาง DFT เป็น

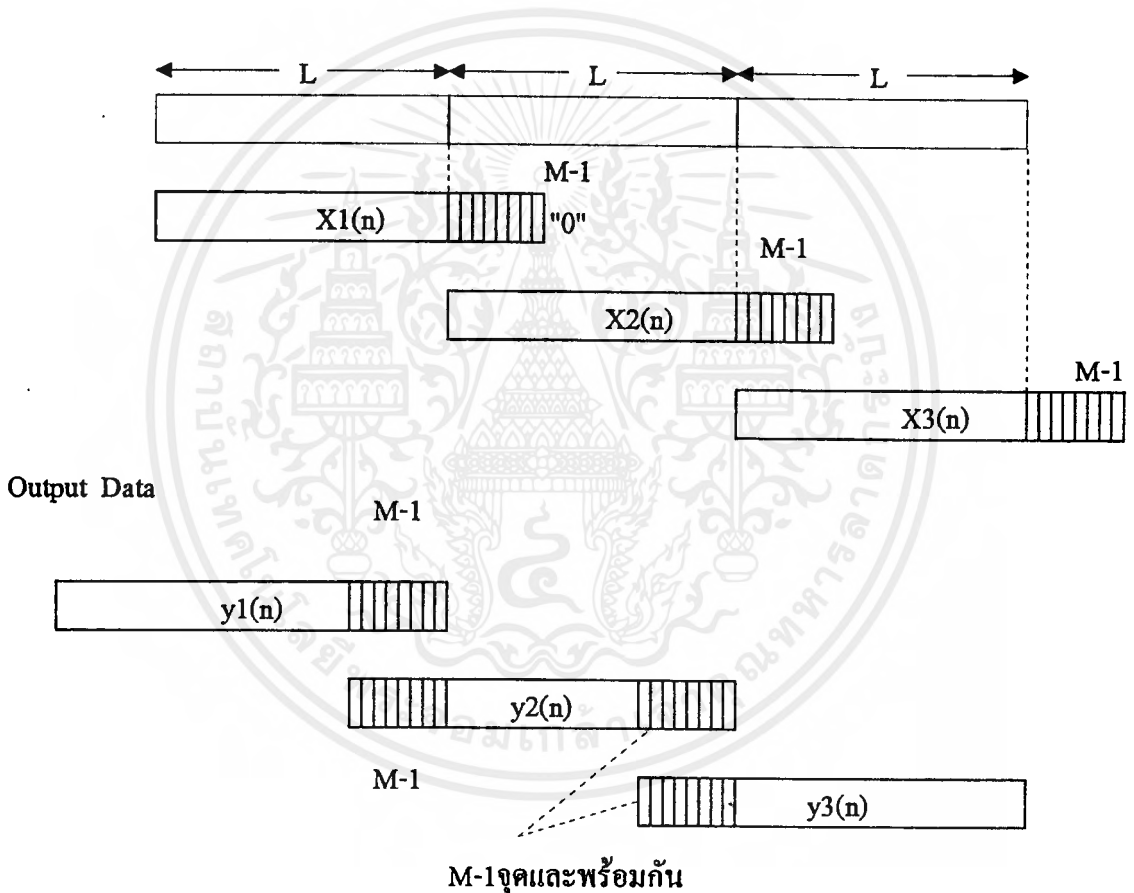
$$Y_m(k) = H(k)X_m(k)$$

เมื่อ $k = 0, 1, 2, 3, 4, 5, \dots, N-1$

เมื่อมีการเติม " 0 " ลงไปในแต่ละบล็อกจำนวน $M-1$ ตัวในแต่ละบล็อก ดังนั้นทางด้านเข้าที่พูดจะการเหลื่อมล้ำกันของข้อมูลและจะอยู่ที่ส่วนท้ายของบล็อกด้วย ดังนั้นเราจึงเรียกว่า Overlap-Add และจะให้เอาท์พุทเมื่อผ่าน IDFT ที่มีขนาดเป็น N จุด ออกมาเป็น

$$y(n) = \{y_1(0), y_1(1), \dots, y_1(L-1), y_1(L) + y_2(0), y_1(L+1) + y_2(1), \dots, \dots, y_1(N-1) + y_2(M-1), y_2(M), \dots\}$$

ในการแบ่งส่วนของสัญญาณอินพุต ไปเป็นบล็อกนั้นเพื่อให้เกิดความเหมาะสมกับ Output Data Block เราสามารถแบ่งได้ดังรูป



รูปที่ 2.11.3 แสดง Overlap-Add method

ขั้นตอนของวิธีการ Overlap-Add มีดังนี้คือ

1. ที่ FFT ขนาด N จุด ของ FIR Impulse Response ประกอบด้วยส่วนที่มีการเติม " 0 " ให้กับ $h(n)$ ที่มีลำดับเป็น N จุด
2. เลือก $N-M+1$ จุด จากลำดับของสัญญาณอินพุตตามสมการข้างล่างคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$x_m(n) = x\{r+(m-1)(N-M+1)\} \quad 0 \leq r \leq N-M$$

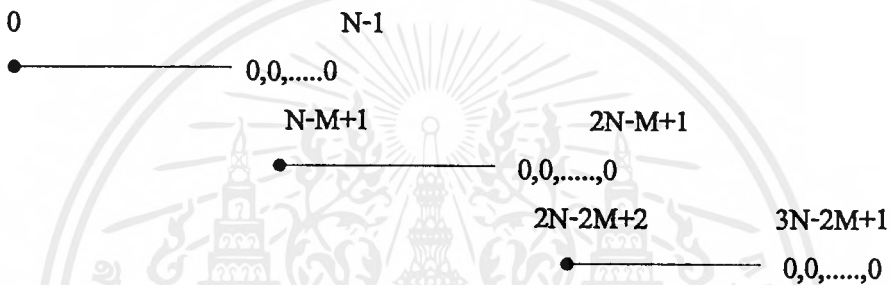
$$= 0 \quad N-M < r \leq N-1 \quad (2.11.4)$$

เมื่อ $n = 0, 1, \dots, N-1$
 $m = 1, 2, 3, 4, \dots$
 $N =$ ความยาวของบล็อกข้อมูลใน FFT
 $M =$ Impulse Response Length

และ

$$h(n) = h(n) \quad n = 0, 1, 2, \dots, M-1$$

$$= 0 \quad n = M, M+1, \dots, N-1$$



รูปที่ 2.11.4 แสดงผลลัพธ์จากสมการ (2.11.4)

จากรูปที่ 2.11.4 แสดงให้เห็นถึงผลลัพธ์จากสมการ (2.11.4) สำหรับอินพุตที่ถูกส่งเข้า FFT จะเกิดขึ้นอย่าง ต่อเนื่อง

3. ทำการแปลงสัญญาณจากโดเมนทางเวลามาเป็น โดเมนทางความถี่ โดยการใช้ FFT ที่มีขนาด N จุด
4. ทำการคูณสัญญาณทั้งสองที่ผ่านการแปลงมาอยู่ใน โดเมนทางความถี่ ของสัญญาณอินพุตใน บล็อกที่ m
5. นำเอาเอาท์พุทที่ได้จาก Step ที่ 4 มาเข้ากระบวนการแปลงจากข้อมูลใน โดเมนทางความถี่ เป็น โดเมนทางเวลาโดยใช้ IFFT ขนาด N จุด
6. ในเทอมข้อมูลส่วนท้ายขนาด M-1 จุด ของข้อมูลเอาท์พุทบล็อกก่อนหน้านี้กับเอาท์พุทที่ได้ ใน Step ที่ 5 ในเทอมที่อยู่ส่วนหน้า ทำให้ได้ข้อมูลที่ได้เป็น $y(n)$ ดังนี้คือ

$$y(n) = y(n) + y_m(r) \quad r = 0, 1, 2, \dots, M-2$$

$$= y_m(r) \quad M-1 \leq r \leq N-1$$

$$n = (m-1)(N-M+1) + r$$

บทที่ 3

การคำนวณและการสร้าง

3.1 TMS 320C50 DSP Starter Kit

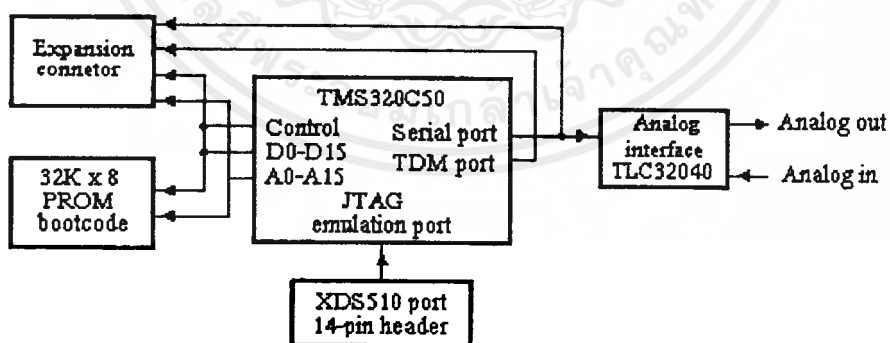
โครงสร้างของ TMS 320C50 DSP Starter Kit

ลักษณะโดยทั่วไปของตัวมาตรฐานบอร์ด

- ใช้โปรเซสเซอร์เบอร์ TMS320C50 ซึ่งใช้งาน DSP โดยเฉพาะ
- คำสั่งจะใช้เวลาประมาณ 50 นาโนวินาที (Instruction Cycle Time)
- หน่วยความจำชนิด PROM ขนาด 32 กิโลไบต์
- สามารถใช้ใน ช่วงความถี่เสียง โดยติดต่อผ่านทาง TLC32040
- ใช้มาตรฐาน RCA คอนเน็คเตอร์ สำหรับสัญญาณอนาล็อกที่เข้าที่อินพุตและเอาต์พุต เพื่อให้สามารถต่อเข้ากับไมโคร โฟนและลำโพง ได้โดยตรง
- ในการติดต่อกับส่วนการควบคุมจะมีคอนเน็คเตอร์ XDS510 ติดต่อกับคอมพิวเตอร์ทางพอร์ตอนุกรม
- สามารถขยาย I/O บัสได้เพื่อสำหรับการออกแบบภายนอก

รูปที่ 3.1.1 แสดงบล็อกไดอะแกรมของบอร์ดซึ่งประกอบไปด้วย โอสอินเทอร์เฟส,

อนาล็อกอินเทอร์เฟส และอิมูเลชันพอร์ต ทำให้สามารถติดต่อกับพีซีได้โดยตรงผ่านทาง RS232 นอกจากนี้ยังมี PROM ขนาด 32 กิโลไบต์ ที่ใช้เก็บเคอร์เนลโปรแกรมเพื่อนำมาใช้ในการเริ่มต้นการบูตเครื่อง ในบอร์ดยังมี TLC32040 วงจรอินเทอร์เฟสสัญญาณอนาล็อก (Analog Interface Circuit:AIC) ที่มี RCA คอนเน็คเตอร์ 2 ตัวสำหรับอินพุตและเอาต์พุต



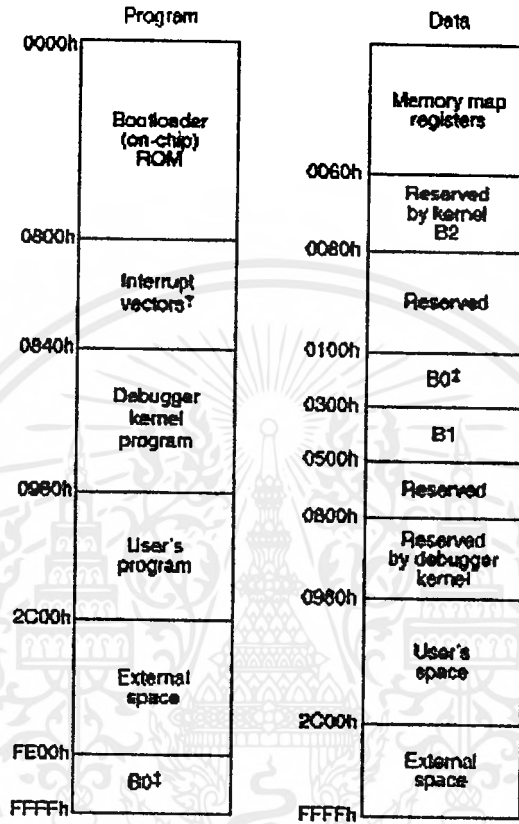
รูปที่ 3.1.1 บล็อกไดอะแกรมของ TMS320C5X DSK

หน่วยความจำใน TMS320C50 DSK นั้น จะแบ่งออกเป็นหน่วยความจำสำหรับโปรแกรมและหน่วยความจำสำหรับข้อมูลซึ่งในแต่ละแอดเดรสก็จะถูกใช้งานต่างๆ ตามแสดงในรูปที่ 3.1.2 และผู้ใช้สามารถใช้หน่วยความจำภายในได้ขนาด 10 K โดยไม่จำเป็นต้องต่อภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

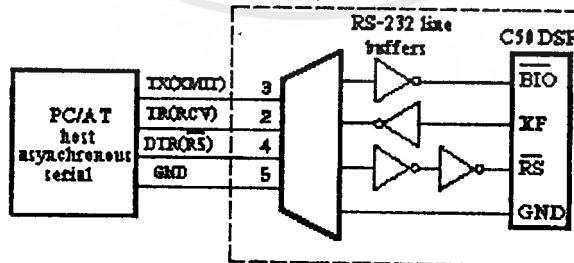
คุณสมบัติของ TLC32040 บนบอร์ด มีดังนี้

- สามารถใช้ในการแปลงสัญญาณดิจิทัลเป็นสัญญาณอะนาล็อก(DAC) และการแปลงสัญญาณอะนาล็อกเป็นสัญญาณดิจิทัล(ADC)
 - สามารถเปลี่ยนอัตราการแซมปลิงทั้งของ ADC , DAC และ ความถี่ฟิลเตอร์ได้
- การติดต่อระหว่าง AIC และ TMS320C50 จะผ่านทางพอร์ตอนุกรมภายในบอร์ด



รูปที่ 3.1.2 แสดงหน่วยความจำภายใน TMS320C50 DSK

การต่อใช้งาน TMS320C50 กับ PC จะใช้ขา XF และ /BIO ที่ต่อผ่านพอร์ตอนุกรม RS232 ดังแสดงในรูปที่ 3.1.3



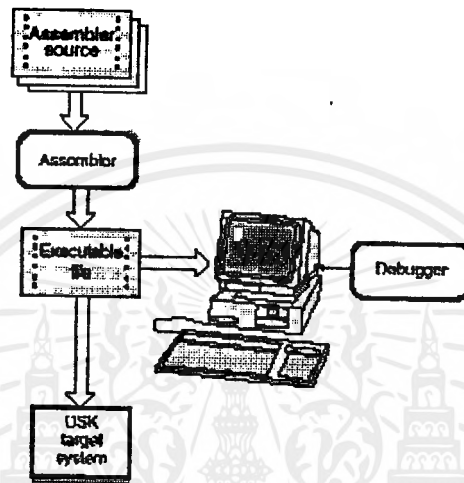
รูปที่ 3.1.3 แสดงการเชื่อมต่อระหว่าง DSK และ PC ผ่านทาง RS232

นอกจากนั้น DSK ยังมีแอสแซมเบอ์ และคิบัคเกอร์ของมันเอง ซึ่งจะมีประโยชน์มากเนื่องจากภาษาแอสแซมบีที่ใช้ใน DSK นั้นจะสนับสนุนการทำงานด้านการประมวลผลสัญญาณ (Signal Processing) และคิบัคเกอร์ก็มีความสามารถที่จะทำงานในแบบทีละขั้น (Single-Step) และเบรคพอยน์เพื่อช่วยในการแก้ไขโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างโปรแกรมเพื่อใช้กับ DSK

1. ทำการสร้าง ไฟล์หลัก (Source File) สำหรับโปรแกรม เช่น EAGLE.ASM
2. ทำการแปลงไฟล์หลัก โดยใช้ DSK แอสเซมเบอร์ ซึ่งมีคำสั่งดังนี้
DSK5A EAGLE.ASM
3. หากต้องการตรวจสอบแก้ไข โปรแกรมโดยใช้ดีบักเกอร์ก็ทำได้โดยใช้โปรแกรม DSK5D.EXE ซึ่งได้ให้มากับบอร์ดแล้ว



รูปที่ 3.1.4 แสดงขั้นตอนการสร้างโปรแกรม

การเขียนโปรแกรมและคำสั่ง

- การกำหนดสถานะเริ่มต้นสำหรับ TMS320C50

ทุกครั้งที่มีการใช้งาน TMS320C50 จำเป็นที่จะต้องมีการกำหนดสถานะเริ่มต้น (Initialize) ตัวโปรเซสเซอร์เสียก่อนเสมอ ซึ่งจะต้องทำการเตรียมสถานะของโปรเซสเซอร์ให้พร้อม โดยปกติจะทำให้เมื่อเกิดการรีเซต ซึ่งการรีเซตจะเกิดขึ้นเมื่อขา \overline{RS} อยู่ในสถานะต่ำ ซึ่งจะทำให้ IPTR (Interrupt Pointer) ซึ่งอยู่ใน PMST (Processor Mode Status Register) เกิดการเคลียร์ ทำให้เกิดการ Map เวกเตอร์ไปที่ Page 0 ในการเกิดการอินเตอร์รัพต์นั้นเมื่อรีเซตจะทำให้ Disable

ในการกำหนดสถานะเริ่มต้นสำหรับโปรเซสเซอร์นั้นจะประกอบด้วย

- การ Map หน่วยความจำ และรีจิสเตอร์ควบคุม
- การเซตโครงสร้างของอินเตอร์รัพต์
- โหมดการควบคุมต่างๆ (OVM, SXM, PM, AVIS, NDX, TRM)
- การควบคุมหน่วยความจำ (RAM, OVLY, CNF)
- การกำหนดรีจิสเตอร์ช่วย (Auxiliary Register) และตัวชี้รีจิสเตอร์ช่วย
- การกำหนดตัวชี้เพจข้อมูล (Data Page Pointer :DP)

การกำหนดค่าใน PMST (Processor Mode Status Register)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPTR	0	0	0	AVIS	0	OVLY	RAM	MP/MC	NDX	IRM	BRAT				

รูปที่ 3.1.5 แสดงการกำหนดค่าใน PMST

ค่า IPTR เราจะกำหนดที่ IPTR = 00001B หรือเป็นที่เพจ 1 ซึ่งตรงกับโปรแกรมตำแหน่ง 800h ส่วนการกำหนดการใช้ RAM ภายใน จะต้องใช้ RAM 9 K ซึ่งอยู่ที่ตัวชิพ กำหนด RAM, OVLY เป็น " 1 " RAM ที่ใช้จะเป็น SARAM โดยที่หน่วยความจำโปรแกรมอยู่ที่ 0800h-2BFFh และหน่วยความจำข้อมูลอยู่ที่ 0800h-2BFFh

การเขียนโปรแกรมทางด้านคณิตศาสตร์และลอจิก (Logic and Arithmetic Operations)

TMS 320C50 มี PLU (Parallel Logic Unit) ซึ่งจะมีการกระทำทางลอจิก (Logic) ที่มีความเร็วมาก ในส่วนนี้มีรีจิสเตอร์ซึ่งใช้ในการประมวลผล ประกอบไปด้วย

- 1.) สเกลลิงชิฟเตอร์ (scaling shifter) 16 บิต
- 2.) ตัวคูณแบบขนาน 16×16 บิต
- 3.) อริทเมติก ลอจิก ยูนิท (arithmetic logic unit: ALU) 32 บิต
- 4.) แอคคิวมูเลเตอร์ (accumulator : ACC) 32 บิต
- 5.) แอคคิวมูเลเตอร์บัฟเฟอร์ (accumulator buffer : ACCB) 32 บิต

ซึ่งมีประสิทธิภาพในการประมวลผล ทางด้าน คณิตศาสตร์ และ ลอจิก ที่จำเป็นในการใช้งานทางด้านการประมวลผล ทางดิจิทัล

การใช้สเกลลิงชิฟเตอร์ (scaling shifter) โปรเซสเซอร์ตัวนี้สามารถประมวลผลแล้วทำการเลื่อนบิตได้ ทำให้สะดวกสำหรับการคูณในรูปแบบของ 2^x ตัวอย่างเช่น

LACC #01111h, 8

ซึ่งเป็นการ โหลดค่าเข้า ACC ด้วยค่า 01111h แต่จะเลื่อนไป 8 บิต ผลคือ ACC = 0111100000000B

คำสั่งของ TMS 320C50 ที่สำคัญ

ในการใช้งานของโปรเซสเซอร์ TMS320C50 จะต้องเขียนโปรแกรมเป็นภาษาแอสเซมบลีจึงจำเป็นต้องทราบคำสั่งและวิธีใช้ซึ่งมีอยู่มากมาย สามารถสรุปได้ดังหัวข้อต่อไปนี้

- 1.) คำสั่งที่เกี่ยวกับแอคคิวมูเลเตอร์ และ หน่วยความจำ ซึ่งเป็นคำสั่งที่ใช้บ่อย และสำคัญ

LACC โหลด ACC โดยการชิฟต์เข้ามา

LACL โหลดค่าเข้ามาในเวิร์ดต่ำของ ACC

ซึ่งคำสั่งเป็นการนำค่ามาเก็บไว้ที่แอคคิวมูเลเตอร์ซึ่งมีขนาด 32 บิต จะมีทั้งนำมาเก็บทั้ง 32 บิต หรือ ขนาด 16 บิต คำสั่งจะอำนวยความสะดวกในการใช้งานเพราะมีการเลื่อน ข้อมูลเข้ามาด้วย

SACH เก็บค่าในเวิร์ดสูงของ ACC โดยการชิฟต์เข้ามา

SACL เก็บค่าในเวิร์ดต่ำของ ACC โดยการชิฟต์เข้ามา

จะตรงข้ามกับคำสั่งที่ผ่านมา ซึ่งจะมีทั้ง 32 บิต และ 16 บิต ส่วนอีกคำสั่งหนึ่งเป็นการติดต่อระหว่าง ACC กับ เมมโมรีรีจิสเตอร์แมป (memory register map) คือ

SAMM ส่งค่า ACCL ไปยังเมมโมรีแมปรีจิสเตอร์

LAMM โหลด ACC ด้วยค่าในเมมโมรีแมปรีจิสเตอร์

2.) คำสั่งเกี่ยวกับ รีจิสเตอร์ช่วย และ เพจข้อมูล

LAR โหลด AR

LDP โหลดเพจข้อมูล

MAR ขยายรีจิสเตอร์ช่วย (Modify Auxiliary Register)

ซึ่งจะมีรีจิสเตอร์ช่วยถึง 8 ตัวคือ AR0-AR7 ในการใช้งานจะต้องกำหนด ARP (Auxiliary Register Pointer) ด้วย ส่วนในการกำหนดเพจ ของข้อมูลมีทั้งหมด 512 เพจ เพจละ 128 เวิร์ด ซึ่งจะกล่าวในเรื่องหน่วยความจำอีกครั้งหนึ่ง

3.) คำสั่งเกี่ยวกับ พาราเลลลอจิกยูนิต (Parallel Logic Unit)

TMS320C50 มีคำสั่งเกี่ยวกับ logic ซึ่งจำเป็นในการใช้งานถึงระดับบิต ที่สำคัญคือ

SPLK เก็บค่าลงบนหน่วยความจำข้อมูล

ซึ่งเป็นคำสั่งการเก็บค่าลงบนหน่วยความจำข้อมูล ซึ่งกระทำในระดับบิต ส่วนมากจะใช้ในการควบคุมบิตต่างๆ ในรีจิสเตอร์ ที่จะเป็นไม่ว่าจะเป็น PMST (Processor Mode Status Register), IMR (Interrupt Mask Register), BRCCR (Block Repeat Countor Register) เป็นต้น

4.) คำสั่งในการคูณ

ในการประมวลผล การคูณมีความจำเป็นอย่างมาก TMS320C50 จึงมีคำสั่งที่ใช้รองรับเกี่ยวกับการประมวลผล ทางคิจิตอลไว้มาก ไม่ว่าจะเป็นการ คอนโวลูชัน (convolution), คอร์รีเลชัน (correlation)

LT โหลด TREGO

MACD การคูณ / แอคคิวมูเลต (accumulate) โดยรีพีตข้อมูล

ZPR Zero Product Register

คำสั่ง MACD เป็นการคูณแล้วเลื่อนข้อมูล ไปในตัวซึ่งสะดวกในการทำ คอนโวลูชัน

5.) คำสั่งสำหรับการกระโดด

B กระโดดโดยไม่มีเงื่อนไข

BANZ กระโดด ถ้า ARX = 0

BCND กระโดดโดยมีเงื่อนไข

CALL เรียก โดยการไม่มีเงื่อนไข

RETI รีเทอร์นอินเตอร์รัพท์ (Return Interrupt)

RET รีเทอร์น (Return)

จะเห็นว่าคำสั่ง BCND (Branch Condition) จะเป็นคำสั่งที่สำคัญเพราะมีเงื่อนไขมากดังนี้

ACC = 0	EQ
ACC ≠ 0	NEQ
ACC < 0	LT
ACC ≤ 0	LEQ
ACC > 0	GT
ACC ≥ 0	GEQ
C = 0	NC
C = 1	C
OV = 0	NOV
OV = 1	OV
BIO low	BIO
TC = 0	NTC
TC = 1	TC
Uncondition	UNC

6.) คำสั่งที่เกี่ยวข้องกับการควบคุมระบบ

จากที่ได้กล่าวในพาราลลอลจิกยูนิต (Parallel Logic Unit) ซึ่งกระทำระดับบิต จำเป็นสำหรับการควบคุมระบบแล้ว ยังมีคำสั่งซึ่งเจาะจงระดับบิตมากขึ้น ดังนี้

SETC	เซตบิต
CLRC	เคลียร์บิต
BIT	ทดสอบบิต

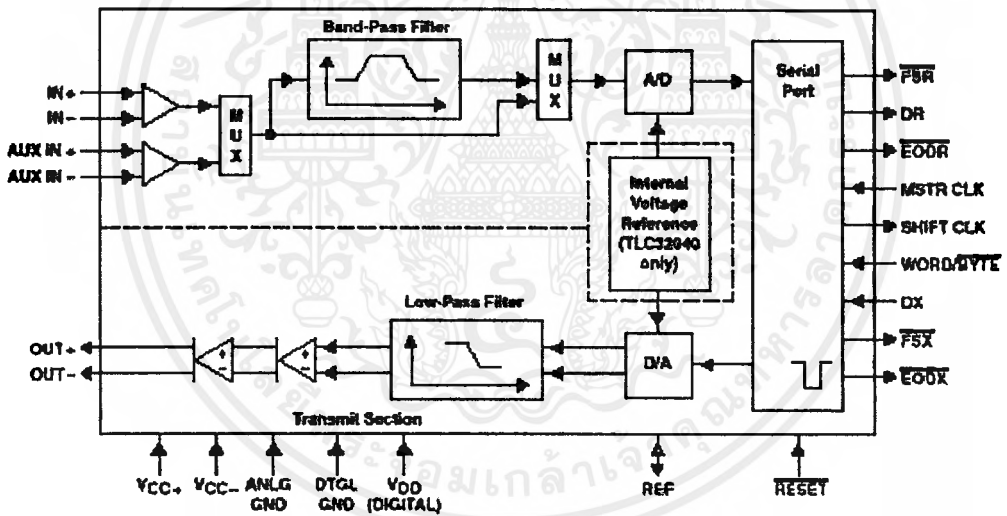
ซึ่งคำสั่ง BIT จะใช้มากเพราะมีเงื่อนไขหลายอย่างเช่น BIT 0h, 15 เลข 15 จะเป็นตัวกำหนดบิต ซึ่งมีได้ตั้งแต่ 0 ถึง 15 นั้นหมายถึงเราจะทดสอบบิต ได้ทุกตำแหน่งในหน่วยความจำด้วย

วงจรอินเทอร์เฟซสัญญาณอนาล็อก TLC32040

TLC 32040 เป็น ชิพซีมอส (CMOS Chip) ที่ใช้เชื่อมต่อกับ TMS 320C50 ลักษณะสำคัญของ TL320C40 มีดังนี้

- ใช้เทคโนโลยีการผลิต Advance Lin CMOSTM Silicon-Gate Process
- ความละเอียดของ ADC และ DAC เป็น 14 บิต
- สามารถเปลี่ยนแปลงอัตราการสุ่มข้อมูลของ ADC และ DAC ได้ถึง 19,200 ครั้ง/วินาที
- มี Switched-Capacitor Antialiasing Input Filter และ Output-Reconstruction Filter
- มีพอร์ตควบคุมสำหรับการติดต่อได้โดยตรงกับ TMS3211, TMS320C17, TMS320C25
- สามารถปรับอัตราการเปลี่ยนแปลงของ ADC และ DAC ทั้งแบบซิงโครนัส หรือ อะซิงโครนัส โดยอาศัยโปรแกรมช่วยในการควบคุม

ฟังก์ชันบล็อกไดอะแกรม ของ TLC 32040



รูปที่ 3.1.6 แสดงฟังก์ชันบล็อกไดอะแกรม ของ TLC 32040

ตำแหน่งขาและหน้าที่การทำงานของแต่ละขา

ตารางที่ 3.1.1 แสดงตำแหน่งขาและหน้าที่การทำงานของแต่ละขา

ชื่อ	หมายเลข	I/O	รายละเอียด
ANLG GND	17,18		กราวด์อนาล็อก
AUX IN+	24	I	ขาอินพุตเสริมแบบ นอน-อินเวอร์ต
AUX IN-	23	I	ขาอินพุตเสริมแบบ อินเวอร์ต
DGTL GND	9		กราวด์ดิจิตอล
DR	5	O	ใช้สำหรับส่งเอาต์พุต ADC จาก AIC (Analog Interface Circuit) ไปยัง TMS320C50 ผ่านทางพอร์ตอนุกรม ซึ่งการส่งผ่านพอร์ตอนุกรมจะต้องซิงค์กับ Shift Clock
DX	12	I	ใช้รับอินพุต DAC หรือคำสั่งการควบคุมจาก TMS320C50 ผ่านทางพอร์ตอนุกรมและจะต้องซิงค์กับสัญญาณ Shift Clock
EODR	3	O	สัญญาณหยุดรับข้อมูล(End of Data Receiver): ในการติดต่อกันผ่านทางพอร์ตอนุกรมในโหมดเวิร์ด (Word Mode) สัญญาณ /EODR จะอยู่ในสถานะต่ำทันทีเมื่อทั้ง 16 บิต ของเอาต์พุต A/D ได้ถูกส่งจาก AIC ไปยัง TMS320C50 ซึ่งสามารถใช้สัญญาณนี้ในการอินเทอร์รัพท์ไมโคร โปรเซสเซอร์ให้ทราบว่าสิ้นสุดการติดต่อแล้ว หรือใช้สไตรบและให้รีจิสเตอร์เลื่อนข้อมูลออก(Enable External Serial-to-Parallel Shift Register) ก็ได้ แต่ถ้าเป็นโหมดไบต์ (Byte Mode) สัญญาณ /EODR จะอยู่ในสถานะต่ำหลังจากไบต์แรกได้ส่งไปยัง TMS320C50 แล้วและยังคงรักษาสถานะต่ำจนกระทั่งไบต์ที่สองได้ส่งไป
EODX	11	O	สัญญาณหยุดส่งข้อมูล(End of Data Transmit) ก็คล้ายกับ /EODR ซึ่งจะบอกให้ทราบว่า การติดต่อกับ TMS320C50 ไปยัง AIC นั้นเสร็จแล้วทั้งในโหมดเวิร์ด และ โหมดไบต์ ก็คล้ายกับ /EODR

ชื่อ	หมายเลข	I/O	คำอธิบาย
FSR	4	O	สัญญาณซิงค์การรับ (Frame Sync Receiver): ในการติดต่อทางพอร์ตอนุกรม /FSR จะอยู่ในสถานะค่าตลอดการส่งจาก AIC ไปยัง TMS320C50 ซึ่งบิตแรกที่จะส่งจะต้องพร้อมอยู่ที่ขา DR ก่อน /FSR จะอยู่ในสถานะค่า
FSX	14	O	สัญญาณซิงค์การส่ง (Frame Sync Transmit) : เมื่อสัญญาณนี้อยู่ในสถานะค่า พอร์ตอนุกรม TMS320C50 จะส่งบิตไปยัง AIC โดยมาที่ขา DX ในการติดต่อพอร์ตอนุกรมทุกโหมด /FSR จะอยู่ในสถานะค่าตลอดการส่ง
IN+	26	I	ขาอินพุทชนิด นอน-อินเวอร์ตติ้ง
IN-	25	I	ขาอินพุทชนิด อินเวอร์ตติ้ง
MST CLK	6	I	สัญญาณนาฬิกามาสเตอร์(Master Clock) จะใช้ในการควบคุมทุกส่วนภายใน AIC ไม่ว่าจะเป็นสัญญาณนาฬิกาเลื่อน(Shift Clock),สัญญาณนาฬิกาควบคุมฟิลเตอร์(Switched-Capacitor Filter Clock), A/D และ D/A Timing
OUT+	22	O	ขาเอาต์พุทชนิด นอน-อินเวอร์ตติ้ง
OUT-	21	O	ขาเอาต์พุทชนิด อินเวอร์ตติ้ง
REF	8	I/O	สำหรับ TLC32040 และ TLC32040 แรงดันอ้างอิงภายในจะถูกส่งออกมาที่ขานี้ แต่ถ้าเป็น TLC32040 และ TLC32040 แรงดันอ้างอิงจากภายนอกจะต่อเข้าที่ขานี้
RESET	2	I	รีเซ็ตเพื่อตั้งค่า TA , TA' , TB ,RA , RA' , RB และรีจิสเตอร์ควบคุม (Control Register) ให้เป็นค่าดีฟอลท์ (Default) ซึ่งจะบอกภายหลัง
SHIFT CLK	10	O	สัญญาณนาฬิกาเลื่อนเกิดจากการหารความถี่ของสัญญาณนาฬิกามาสเตอร์ด้วย 4 ซึ่งสัญญาณนี้จะใช้ในการติดต่อทางพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ	หมายเลข	I/O	คำอธิบาย
VDD	7		ขาไฟเลี้ยงส่วนวงจรดิจิทัล(Digital Supply Voltage) $5V \pm 5\%$
VCC+	20		ขาไฟเลี้ยงส่วนวงจรมอนาโลกที่เป็นบวก(Positive Analog Supply Voltage) $5V \pm 5\%$
VCC-	19		ขาไฟเลี้ยงส่วนวงจรมอนาโลกที่เป็นลบ(Negative Analog Supply Voltage) $5V \pm 5\%$
WORD/BYTE	13	O	ขานี้จะทำงานร่วมกับรีจิสเตอร์ควบคุมเพื่อใช้ในการเลือกโหมดการติดต่ออนุกรม ซึ่งมี 4 แบบ ดังนี้คือ <ul style="list-style-type: none"> - การติดต่อแบบ อะซิงโครนัส - ในโหมดไบต์ (WORD/BYTE = สถานะต่ำ) <p>พอร์ตอนุกรมจะติดต่อโดยตรงกับ TMS320C50 และจะติดต่อทีละ 8 บิต 2 ครั้ง ซึ่งมีขั้นตอนการทำงานดังนี้</p> <ol style="list-style-type: none"> 1. /FSX หรือ /FSR อยู่ในสถานะต่ำ 2. 8 บิตแรกถูกส่งออกไป หรือรับเข้ามา 3. /EODX หรือ /EODR อยู่ในสถานะต่ำ 4. /FSX หรือ /FSR อยู่ในสถานะสูงประมาณ 4 สัญญาณนาฬิกาที่ใช้เลื่อนข้อมูลแล้วอยู่ในสถานะต่ำ 5. 8 บิตต่อมา (ของไบต์ที่สอง)ถูกส่งหรือรับเข้ามา 6. /EODX หรือ /EODR อยู่ในสถานะสูง 7. /FSX หรือ /FSR อยู่ในสถานะสูง - ในโหมดเวิร์ด <p>พอร์ตอนุกรมจะต่อตรงกับพอร์ตอนุกรมของ TMS320C50 และมีการติดต่อครั้งเดียว 16 บิต</p>

ชื่อ	หมายเลข	I/O	คำอธิบาย
			<p>ซึ่งมีขั้นตอนดังนี้</p> <ol style="list-style-type: none"> 1. /FSX หรือ /FSR อยู่ในสถานะต่ำ 2. 16 บิต ถูกส่งหรือรับเข้ามา 3. /FSX หรือ /FSR อยู่ในสถานะสูง 4. /EODX หรือ /EODR อยู่ในสถานะต่ำ <p>- การติดต่อแบบซิงโครนัส</p> <p>ในกรณี แบนด์พาสฟิลเตอร์ (Bandpass Filter) และอัตราการแปลง A/D(A/D Conversion Timing) จะถูกกำหนดจาก TX เคนเตอร์ A ,TX เคนเตอร์ B และ TA , TA' และ TB แทนส่วนในการติดต่อทั้งในแบบโหมคเวีร็อคและโหมคไบคั้นมีขั้นตอนเหมือนกับการติดต่อแบบอะซิงโครนัส</p>

การทำงานของ TLC320C40

อินพุททางอนาล็อก (Analog Input)

สำหรับอนาล็อกอินพุทจะมีอยู่ 2 กลุ่ม คือ IN+ , IN- และ AUX IN+ ,AUX IN- ซึ่งสามารถเลือกใช้งานกลุ่มใดกลุ่มหนึ่ง โดยใช้ในแบบ ดิเฟอเรนเชียล หรือซิงเกิล-เอนด์ ค่าแกนสำหรับอินพุท IN+ ,IN- ,AUX IN+ และ AUX IN- สามารถจะใช้โปรแกรมตั้งค่าได้(มี 3 ค่าคือ 1,2 หรือ 4)การเลือกใช้กลุ่มอินพุทใดทำได้โดยอาศัยการ โปรแกรมควบคุม

A/D Bandpass Filter, A/D Bandpass Filter Clocking และ A/D Conversion Timing

สำหรับ A/D Bandpass Filter เราสามารถที่จะกำหนดให้สามารถใช้งาน ได้โดยอาศัยตัวโปรแกรมควบคุมความถี่ของสัญญาณพิกาคควบคุมการฟิลเตอร์จะเป็นตัวกำหนดทรานเฟอร์ฟังก์ชันของฟิลเตอร์โดยคิดอัตราส่วนจากความถี่สัญญาณพิกาคควบคุมฟิลเตอร์ 28 KHz ที่ความถี่ต่ำที่เริ่มมีลักษณะเป็นความถี่สูงผ่านจะมีความถี่เป็น 300 Hz อัตราการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อก ก็หาได้จากความถี่ที่หาร 228 KHz ด้วย RX เคนเตอร์ B

Analog Output

ขาที่ให้สัญญาณเอาต์พุทออกมาเป็นสัญญาณอนาล็อกจะมีวงจรขยายซึ่งจะให้เอาต์พุทออกมาได้ทั้งแบบไม่มีการกลับขั้วของสัญญาณ(Nominverting)และแบบมีการกลับขั้วของสัญญาณ(inverting)ทำให้สามารถที่จะขับ ทรานส์ฟอร์มเมอร์ชนิดไฮ-บริดจ์ (Transformer Hybrid) หรือที่โหลดอิมพีแดนซ์ต่ำได้ทั้งแบบดิเฟอเรนเชียลหรือ ซิงเกิล-เอนด์

วงจรรองความถี่ต่ำของ การแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อก ,วงจรรองความถี่แบบแบนด์พาสของการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อก ,สัญญาณพิกาคควบคุมรอง

ความถี่ต่ำ และอัตราการแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก(D/A Lowpass Filter,D/A Lowpass Filter Clcking และ D/A Conversion Time)

เช่นเดียวกันกับ การกรองความถี่ของการแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัลโดยมีทราสเฟอร์ฟังก์ชันจะถูกกำหนดจากอัตราส่วนกับความถี่ 228 KHz และอัตราการแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อกก็หาได้จากความถี่ 228 KHz หารด้วย TX เคาน์เตอร์ B

การต่อกลับ(Loopback)

การต่อกลับจะให้ผู้ใช้สามารถตรวจสอบวงจรได้โดย OUT+ และ OUT- จะต่อเข้าภายในกับ IN+ และ IN- ดังนั้นบิตข้อมูลของการแปลงสัญญาณดิจิทัลให้เป็นสัญญาณอนาล็อก D15-D2 จะถูกส่งไปยังขา DX และเปรียบเทียบกับบิตข้อมูลของการแปลง

ตารางที่ 3.1.2 แสดงการกำหนดค่าใน D1 และ D0 ในการทำงานในโหมดต่างๆของ AIC

D1	D0	การทำงาน
0	0	จะทำการโหลดค่าในรีจิสเตอร์ TA และ RA ไปยัง TX และ RX เคาน์เตอร์ A ตามลำดับ และจะทำการโหลด TX และ RX เคาน์เตอร์ ด้วยค่าในรีจิสเตอร์ TB และ RB
0	1	TX และ RX เคาน์เตอร์ A จะถูกโหลดด้วยค่า TA+TA' และ RA+RA' แต่จะทำการ โหลด TX และ RX เคาน์เตอร์ ด้วยค่าในรีจิสเตอร์ TB และ RB
1	0	TX และ RX เคาน์เตอร์ A จะถูกโหลดด้วยค่า TA-TA' และ RA-RA' แต่จะทำการ โหลด TX และ RX เคาน์เตอร์ ด้วยค่าในรีจิสเตอร์ TB และ RB
1	1	TX และ RX เคาน์เตอร์ A จะถูกโหลดด้วยค่าในรีจิสเตอร์ TA และ RA ส่วน TX และ RX เคาน์เตอร์ B จะถูกโหลดด้วยค่าในรีจิสเตอร์ TB และ RB หลังจากนั้นจะรอประมาณ 4 สัญญาณนาฬิกาเลื่อนก็จะทำการส่งครั้งที่ 2 (Secondary Communication) เพื่อไปโปรแกรม AIC ให้ทำงานตามต้องการ

ฟังก์ชันรีเซต(Reset)

ฟังก์ชันรีเซตจะเป็นการตั้งค่าอัตราการเปลี่ยนแปลงของการแปลง สัญญาณดิจิทัลเป็นสัญญาณอนาล็อกและการแปลง สัญญาณอนาล็อกเป็นสัญญาณดิจิทัล ให้เป็น 8 KHz ถ้าใช้สัญญาณนาฬิกา 5.184 MHz โดยค่าในรีจิสเตอร์จะถูกตั้งค่าเริ่มต้นเป็น

TA = RA = 9, TA' = RA' = 1, TB = RB = 24 และบิตในรีจิสเตอร์ควบคุมเป็น D7 = 1, D6 = 1, D5 = 1, D4 = 0, D3 = 0, D2 = 1

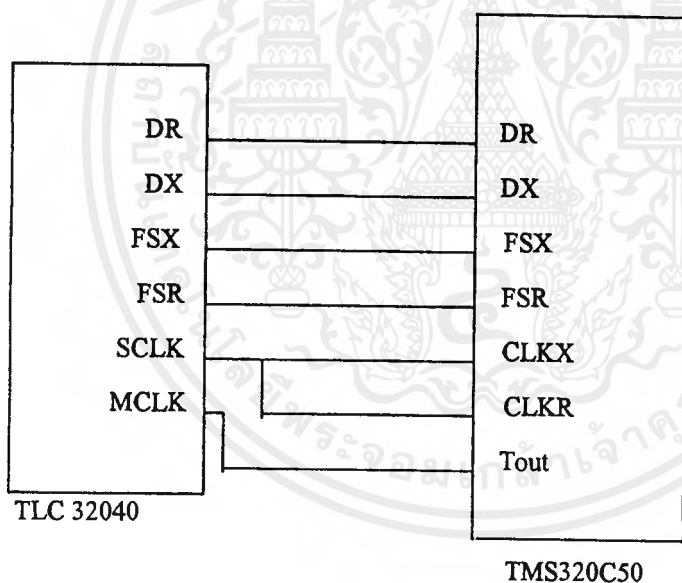
ข้อบังคับในการกำหนดค่าในรีจิสเตอร์ของ AIC

1. รีจิสเตอร์ TA ในโหมดเวิร์ค ต้องมีค่า ≥ 4
2. รีจิสเตอร์ TA ในโหมดเวิร์ค ต้องมีค่า ≥ 5
3. รีจิสเตอร์ TA' สามารถเป็นได้ทั้งค่า บวก, ลบ และ ศูนย์
4. รีจิสเตอร์ RA ในโหมดเวิร์ค ต้องมีค่า ≥ 4
5. รีจิสเตอร์ RA ในโหมดเวิร์ค ต้องมีค่า ≥ 5
6. รีจิสเตอร์ RA สามารถเป็นได้ทั้งค่า บวก, ลบ และ ศูนย์
7. $(TA \pm TA')$ จะต้องมีค่ามากกว่า 1
8. $(RA \pm RA')$ จะต้องมีค่ามากกว่า 1
9. TB จะต้องมีค่ามากกว่า 1

ถ้าไม่เป็นไปตามเงื่อนไขทั้ง 9 ข้อนี้ AIC ก็จะทำตามเงื่อนไขดังตารางที่ 3.1.3

การอินเตอร์เฟสระหว่าง TMS320C50 และ TLC32024

การใช้อินเตอร์เฟซของพอร์อนุกรม



รูปที่ 3.1.7 แสดงการเชื่อมต่อระหว่าง TLC32040 กับ TMS320C50

ตารางที่ 3.1.3 แสดงเงื่อนไขของการตอบสนอง AIC

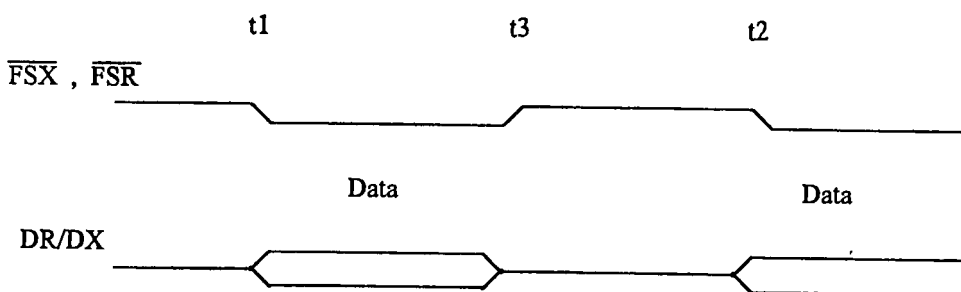
เงื่อนไข	การตอบสนองของ AIC
1. ถ้า $TA + TA' = 0$ หรือ 1 ถ้า $TA - TA' = 0$ หรือ 1	จะทำการโหลดค่าในรีจิสเตอร์ TA ลงใน TX เคนเตอร์ A แทน
2. ถ้า $TA + TA' < 0$	MODULO 64 จะถูกใช้เพื่อให้แน่ใจว่า ค่าที่บวกถูกโหลดไปไว้ที่ TX เคนเตอร์ A แทน $TA + TA' + 40h$ โหลดไปไว้ที่ TX เคนเตอร์ A
3. ถ้า $RA + RA' = 0$ หรือ 1 ถ้า $RA - RA' = 0$ หรือ 1	จะทำการโหลดค่าในรีจิสเตอร์ RA ลงใน RX เคนเตอร์ A
4. ถ้า $RA + RA' = 0$ หรือ 1	MODULO 64 ถูกใช้เหมือนข้อ 2 เพียงแต่โหลดไปที่ RX เคนเตอร์ A
5. ถ้า $TA = 0$ หรือ 1 หรือ $RA = 0$ หรือ 1	AIC หยุดทำงาน
6. ถ้า $TA < 4$ ในโหมดเวิร์ด $TA < 5$ ในโหมดเวิร์ด $RA < 4$ ในโหมดเวิร์ด $RA < 4$ ในโหมดเวิร์ด	การติดต่อทางเทอร์มินัลของ AIC ไม่ทำงาน
7. ถ้า $TB = 0$ หรือ 1	โหลดค่า TB ใหม่ด้วยค่า 24h
8. ถ้า $RA = 0$ หรือ 1	โหลดค่า RB ใหม่ด้วยค่า 24h
9. ถ้า AIC และ DSP ไม่สามารถติดต่อกันได้	จะเก็บค่า DAC Output ไว้

TMS320C50

- การส่ง โดยการใช้ DXR จะเป็นการขออินเทอร์รัพท์ของ XINT (โดยไม่มีมารด์)
- การรับ เมื่อ RSR → DRR เต็มจะเกิดการอินเทอร์รัพท์ของ RINT (โดยไม่มีมารด์)

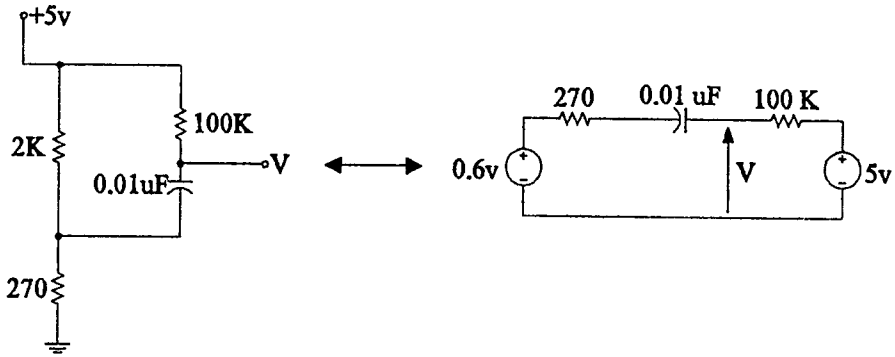
TLC32040

เนื่องจากการส่งระหว่าง TLC32040 กับ TMS320C50 เป็นแบบซิงโครนัส TLC32040 จะส่งและรับทุกๆ เฟรม-ซิงโครนัส (Frame Synchronous)



รูปที่ 3.1.8 แสดงช่วงเวลาในการส่งและรับข้อมูลระหว่าง TLC32040 กับ TMS320C50

เอกสารนี้เป็นเอกสารทบทวนวิชาสำหรับการใช้งานเพื่อการศึกษารายละเอียดเกี่ยวกับเงื่อนไขและข้อกำหนดด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2.2 วงจรสมมูล

ในสถานะเริ่มต้นให้ตัวทรานซิสเตอร์ชนิด NPN (T1) ทำงานอยู่ในสถานะ ON เราสามารถที่จะหาค่าแรงดันที่ตกคร่อมที่ขาควบคุมของ T2 ได้

เมื่อกำหนดให้ค่า $C=10\text{nF}$ และ $R=100\text{K}$

$$i=C \frac{dV_c}{dt} = \frac{5-0.6-V_c}{100\text{K}}$$

$$\frac{1}{V_c-4.4} dV_c = -10^3 dt$$

$$\int_{V_c=V(0)}^{V_c(t)} \frac{1}{V_c-4.4} dV_c = \int_{t=0}^t -10^3 dt$$

$$\ln \frac{V_c(t)-4.4}{V_c(0)-4.4} = -10^3 t$$

$$t = -10^3 \ln \frac{V_c(t)-4.4}{V_c(0)-4.4}$$

$$V_c(t) = 4.4 - (4.4 - V(0)) e^{-10^3 t} \quad (1)$$

$$i=C \frac{dV_c}{dt} = (4.4 - V(0)) \times 10^{-9} \times 10^3 e^{-10^3 t}$$

$$V = 5 - (4.4 - V(0)) \times e^{-10^3 t} \quad (2)$$

$$t = 10^3 \ln \frac{V(0) - 4.4}{V - 5} \quad (3)$$

ค่าแรงดันเริ่มต้นของ C มีค่าเริ่มต้นเป็น 0 V ดังนั้นเราสามารถหาค่าของกระแสได้เป็นเมื่อการตรวจจับแรงดันที่ขาควบคุมจะตรวจสอบแรงดันให้ได้ 2.75 V ดังนั้น

จากสมการที่ 1 และ 3 จะได้

$$t = 10^3 \ln \frac{-4.4}{2.75 - 5} = 0.67 \text{ mSec}$$

$$V_c(0.67 \text{ mSec}) = 2.15 \text{ V}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และในช่วงเวลาต่อมา T1(off) และ T2(on) ทำให้ $V_{c2}(0) = 0 \text{ V}$

$$t = 10^{-3} \ln \frac{-4.4}{2.75 - 5} = 0.67 \text{ mSec}$$

ในขณะที่ T2(on) อยู่ นั้น ค่าแรงดันที่ตกคร่อมที่ C1 จะมีค่าน้อยมากๆ ประมาณ 0 V เนื่องจาก C1 ได้ทำการคายประจุ ผ่านความต้านทานที่ต่ออยู่กับชุดการขับทรานซิสเตอร์และผ่าน R 100K ครบวงจร รวมถึงการคายประจุผ่านตัว IC

จะเห็นได้ว่าค่าแรงดันที่ตกคร่อม $V_c(t)$ จะอยู่ในช่วง ประมาณ 0 ถึง 2.15 V

ดังนั้นความถี่ของการออสซิลเลตจึงเป็น

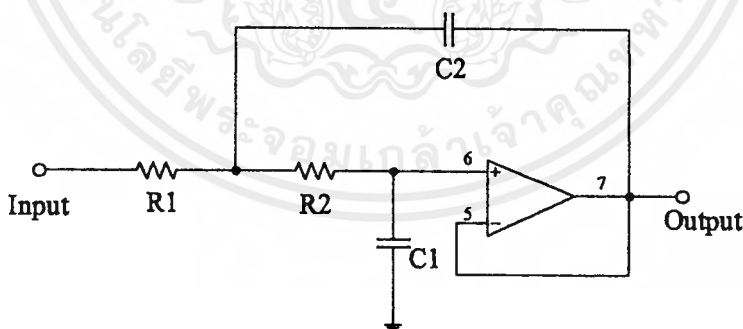
$$f = \frac{1}{2t} = \frac{1}{2 \times 0.67 \text{ mSec}}$$

$$= 745.5518 \text{ Hz}$$

3.3 การออกแบบวงจรกรองความถี่ต่ำ

วงจรกรองความถี่ต่ำผ่านในการออกแบบจะใช้โอปแอมป์รวมกันกับค่าความต้านทานซึ่งในส่วน
ของวงจรในส่วนก่อนการต่อตัวอย่างของข้อมูลจะต้องอาศัยการกำหนดค่าความถี่ที่สูงที่สุด ซึ่งในการสื่อสารเรทราบกันอยู่แล้วว่าสัญญาณเสียงในช่วง 300-3400 Hz เป็นช่วงที่มีสเปคตรัมที่เพียงพอในการสื่อสารความหมาย ดังนั้นในโครงการนี้จึงได้มีการกำหนดให้ความถี่สูงสุดของสัญญาณเสียงมีค่าเพียง

3.4 KHz เท่านั้น การออกแบบวงจรจึงได้นำเอาวิธีการกรองความถี่แบบแอกทีฟมาประยุกต์ใช้กับวงจรกรองความถี่แบบ BUTTERWORTH ซึ่งจะให้ผลการตอบสนองทางความถี่ที่ราบเรียบสม่ำเสมอในช่วงของย่านความถี่ที่ผ่าน



รูปที่ 3.3.1 วงจรกรองความถี่ต่ำแบบ Active

การคำนวณ

$$\text{ที่ Node ที่ 1} \quad \frac{V_1 - V_{in}}{R_1} + \frac{V_1 - V_{in}}{R_2} + (V_1 - V_o)SC_1 = 0 \quad (3.3.1)$$

$$\text{ที่ Node ที่ 2} \quad \frac{V_o - V_1}{R_2} + V_oSC_2 = 0 \quad (3.3.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ 3.3.2 เรานำมาจัดใหม่ได้เป็น

$$V_o \left\{ \frac{1}{R_2} + SC_2 \right\} = \frac{V_1}{R_2} \quad (3.3.3)$$

นำสมการ (3.3.3) ที่ได้มาแทนในสมการที่ (3.3.1) แล้วจัดรูปใหม่จะ ได้เป็น

$$V_o \left\{ (1 + SR_2C_2) \left(\frac{1}{R_1} + \frac{1}{R_2} + SC_1 \right) - SC_1 - \frac{1}{R_2} \right\} = \frac{V_{in}}{R_1}$$

$$\frac{V_o}{V_{in}} = \frac{1}{S^2 R_1 R_2 C_1 C_2 + S [R_2 C_2 + R_1 C_2] + 1} \quad (3.3.4)$$

เมื่อทรานสเฟอ์ฟังก์ชันของ วงจรกรองความถี่ต่ำเป็น

$$T(S) = \frac{w_o^2}{S^2 + S \left(\frac{w_o}{Q} \right) + w_o^2} \quad (3.3.5)$$

เมื่อค่าอัตราขยายมีค่าเท่ากับ 1 และ จัดเทอมของสมการที่ (3.3.4) ให้เหมือนกับสมการที่ (3.3.5)

$$T(S) = \frac{1}{(R_1 R_2 C_1 C_2)} \frac{1}{S^2 + S \left\{ \frac{R_2 C_2 + R_1 C_2}{R_1 R_2 C_1 C_2} \right\} + \frac{1}{R_1 R_2 C_1 C_2}} \quad (3.3.6)$$

เมื่อจับเทียบสัมประสิทธิ์เราจะได้ว่า

$$w_o^2 = \frac{1}{R_1 R_2 C_1 C_2} \quad Q = \frac{\sqrt{(R_1 R_2 C_1 C_2)}}{C_2 (R_1 + R_2)} \quad (3.3.7)$$

โดยปกติในการคำนวณเพื่อให้ง่ายขึ้นเราจะทำการนอร์มอลไลซ์ค่า w_o ให้มีค่าเป็น 1 เรเดียน/วินาทีจากนั้นจึงค่อยนำเอาวิธีการของการทำสเกลลิง (Magnitude Scaling และ frequency Scaling) มาช่วยทำให้ได้ค่าในการออกแบบที่ง่ายขึ้น โดย

$$C_{new} = \frac{C_{old}}{(k_f k_m)}$$

เมื่อทำการนอร์มอลไลซ์ค่าของ w_o , ค่าของ R_1 และ R_2 ให้มีค่า เป็น 1 เราสามารถหาค่าของตัวเก็บประจุได้ในเทอมของค่า Q ได้จากสมการที่ (7)

$$w_o^2 = w_o = R_1 = R_2 = 1, \quad Q = 1/2(C_2)$$

$$C_1 = 2Q \text{ และ } C_2 = 1/2Q$$

ดังนั้นถ้าเราทราบเพียงค่าความถี่คัทออฟที่มีระดับสัญญาณเอาต์พุตลดลงมาเมื่อเทียบกับอินพุต เป็น - 3 dB และทราบค่าของทรานสเฟอ์ฟังก์ชันเราก็สามารถได้ทราบค่าอุปกรณ์ต่างๆ ในวงจรกรองความถี่ต่ำได้

วงจรกรองความถี่ต่ำแบบ BUTTERWORTH จะมีค่าทรานสเฟอ์ฟังก์ชันของวงจรเป็น

$$\text{- ที่วงจรอันดับที่ 4 } T(S) = \frac{1}{(S^2 + 0.76536686S + 1)(S^2 + 1.84775907S + 1)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากเป็นวงจรกรองความถี่ต่ำอันดับที่ 4 เราจะสามารถแยกส่วนของโพลได้เป็น 2 ส่วน และสามารถนำหลักการต่อวงจรแบบคาสเคสกันเพื่อให้ได้ทรานสเฟอร์ฟังก์ชันที่ถูกต้องแล้วตามสมการข้างบน

เราจะแยกพิจารณา การเกิดโพลคู่แรกคือ ที่ตำแหน่ง $S^2 + 0.76536686 S + 1 = 0$

จากที่กล่าวมาแล้วข้างต้นเราจะได้
$$Q = \frac{1}{0.76536686}$$

ดังนั้นค่าของอุปกรณ์ในวงจรส่วนนี้จะมีค่าเป็น

$$R_1 = R_2 = 1 \quad C_1 = 2Q = 2.613125946 \quad C_2 = (2Q)^{-1} = 0.38268343$$

เรามีความสัมพันธ์ $k_f = 2 \times \pi \times \text{Frequency} = 2 \times \pi \times 3.4 \text{ KHz} = 21362.83004$

และ
$$C_{new} = \frac{C_{old}}{(k_f k_m)}$$

โดยค่า k_m จะถูกแทนด้วยค่าของความต้านทานที่จะใช้ในวงจร

เราจะเลือกค่า ให้ $R = 3 \text{ k}\Omega$

$$\begin{aligned} C_{new} &= \frac{C_{old}}{(k_f k_m)} = \frac{2.613125946}{21362.83004 \times 3K} \\ &= 40.7737168 \text{ nF} \end{aligned}$$

ในชิ้นงานนี้เราใช้ค่า $C_1 = 40.7 \text{ nF}$

และในทำนองเดียวกันค่าของ C_2 ก็จะสามารถหาค่าได้ด้วยวิธีเดียวกันดังนี้คือ

$$\begin{aligned} C_{new} &= \frac{C_{old}}{(k_f k_m)} = \frac{0.38268343}{21362.83004 \times 3K} \\ C_{2new} &= 5.971172503 \text{ nF} \end{aligned}$$

ในชิ้นงานนี้เราใช้ค่า $C_2 = 6 \text{ nF}$

พิจารณา การเกิดโพลคู่ต่อมาคือ ที่ตำแหน่ง $S^2 + 1.84775907 S + 1 = 0$

ก็จะได้ค่าเป็น

$$R_1 = R_2 = 1 \quad C_1 = 2Q = 1.082392197 \quad C_2 = (2Q)^{-1} = 0.923879535$$

เรามีความสัมพันธ์ $k_f = 2 \times \pi \times \text{Frequency} = 2 \times \pi \times 3.4 \text{ KHz} = 21362.83004$

และ
$$C_{new} = \frac{C_{old}}{(k_f k_m)}$$

โดยค่า k_m จะถูกแทนด้วยค่าของความต้านทานที่จะใช้ในวงจร

เราจะเลือกค่า ให้ $R = 3 \text{ k}\Omega$

$$\begin{aligned} C_{1new} &= \frac{C_{1old}}{(k_f k_m)} = \frac{1.082392197}{21362.83004 \times 3K} \\ &= 16.88902633 \text{ nF} \end{aligned}$$

ในชิ้นงานนี้เราใช้ค่า $C_1 = 16.8 \text{ nF}$

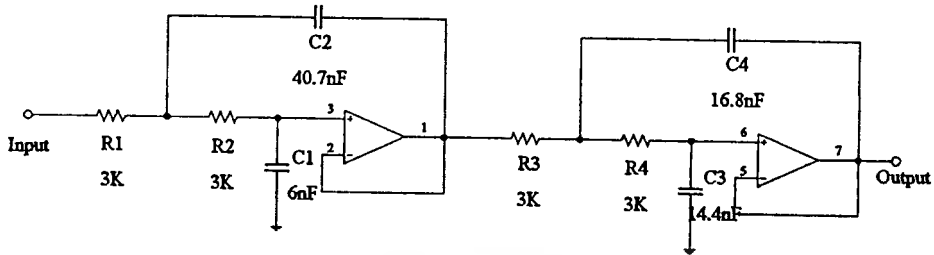
และในทำนองเดียวกันค่าของ C_2 ก็จะสามารถหาค่าได้ด้วยวิธีเดียวกันดังนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$C_{2new} = \frac{C_{2old}}{(k_f k_m)} = \frac{0.923879535}{21362.83004 \times 3K}$$

$$C_{2new} = 14.41568577 \text{ nF}$$

ในชิ้นงานนี้เราใช้ค่า $C_2 = 14.4 \text{ nF}$



รูปที่ 3.3.2 วงจรกรองความถี่ต่ำแบบ BUTTERWORTH อันดับที่ 4 มีความถี่คัทออฟที่ 3.4 kHz

- ที่วงจรอันดับที่ 2 $T(S) = \frac{1}{(S^2 + 1.41421356 S + 1)}$

โพลจะอยู่ที่ตำแหน่ง $(S^2 + 1.41421356 S + 1) = 0$

$$R_1 = R_2 = 1$$

$$C_1 = 2Q = 1.41421356$$

$$C_2 = (2Q)^{-1} = 0.707106781$$

เราจะเลือกค่าให้ $R = 3 \text{ k}\Omega$

$$C_{1new} = \frac{C_{1old}}{(k_f \times k_m)} = \frac{1.41421356}{21362.83004 \times 3K} = 22.06657634 \text{ nF}$$

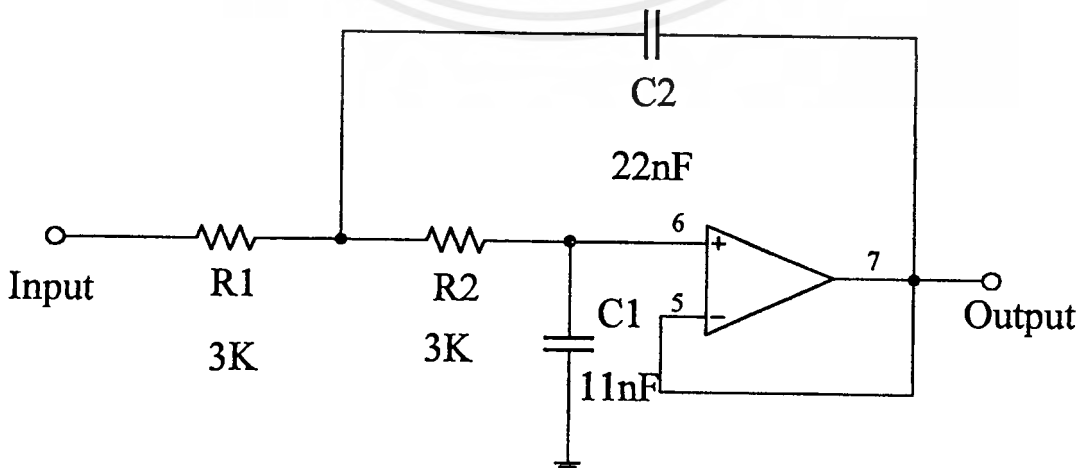
ในชิ้นงานนี้เราใช้ค่า $C_1 = 22 \text{ nF}$

และในทำนองเดียวกันค่าของ C_2 ก็จะสามารหาค่าได้ด้วยวิธีเดียวกันดังนี้คือ

$$C_{2new} = \frac{C_{2old}}{k_f \times k_m} = \frac{0.707106781}{21362.83004 \times 3K}$$

$$C_{2new} = 11.03328819 \text{ nF}$$

ในชิ้นงานนี้เราใช้ค่า $C_2 = 11 \text{ nF}$



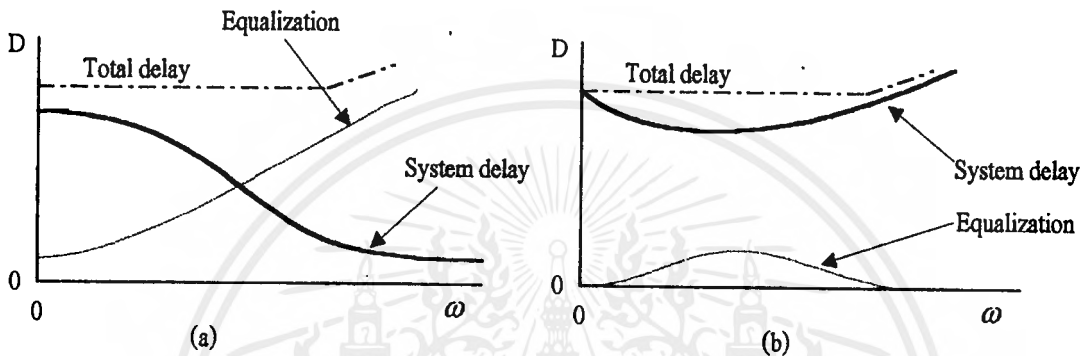
รูปที่ 3.3.3 วงจรกรองความถี่ต่ำแบบ BUTTERWORTH อันดับที่ 2 มีความถี่คัทออฟที่ 3.4 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

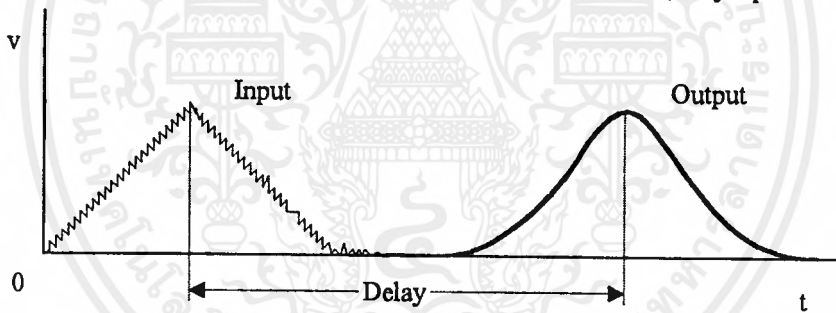
3.4 การออกแบบวงจรหน่วงเวลา(delay Circuit)

1. การกรองความถี่ที่ได้ออกแบบเพื่อที่จะออกแบบถึงผลการตอบสนองทางแอมปริจูดด้วย
2. ผลทางเฟสของการกรองความถี่ที่ต้องการจะเป็นส่วนที่เสริมให้ครอบคลุมตลอดช่วงย่าน

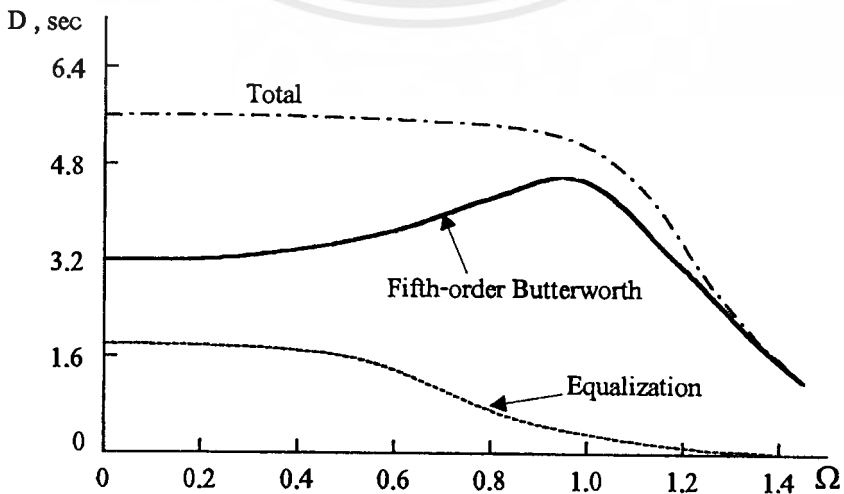
ความถี่เพราะมีความสำคัญอย่างยิ่งต่อการต่อแบบอนุกรมกัน(cadcade)ของวงจรหน่วงเวลาซึ่งเป็นแบบวงจรกรองความถี่ให้ผ่าน ได้ตลอดช่วง(All Pass Filter)แบบออร์เคอร์ที่ 1 และ แบบออร์เคอร์ที่ 2 ซึ่งไม่มีการลดทอน แต่เนื่องจากผลรวมของการหน่วงเวลาทั้งหมดนั้นจะต้องมีค่าโดยประมาณที่คงที่ครอบคลุมตลอดช่วงความถี่ที่ต้องการ ซึ่งในระบบการหน่วงเวลาที่ใช้ในวงจรอีควอไลเซชัน(delay equalization)จะเป็นตัวอย่างที่ดีในการหน่วงเวลาซึ่งแสดงได้ดังในรูปที่ 3.4.1



รูปที่ 3.4.1 แสดงระบบการหน่วงเวลาที่ใช้ในวงจรอีควอไลเซชัน(delay equalization)



รูปที่ 3.4.2 แสดงการป้อนสัญญาณที่มีรูปสัญญาณเป็นสามเหลี่ยมที่มีสัญญาณรบกวนความถี่สูงรวมอยู่กับรูปสัญญาณ

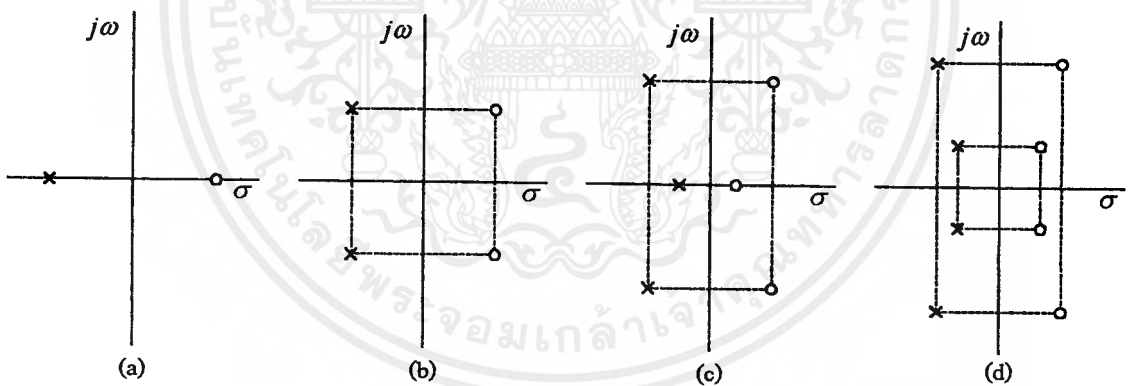


รูปที่ 3.4.3 แสดงผลการตอบสนองต่อการหน่วงเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อพิจารณากระบวนการคั้งในรูปที่ 3.4.2 โคซทำการป้อนสัญญาณที่มีรูปสัญญาณเป็นสามเหลี่ยม (triangular waveform) ที่มีสัญญาณรบกวนความถี่สูงรวมอยู่กับรูปสัญญาณ โดยใช้การกรองความถี่จะทำให้สามารถที่จะกำจัดสัญญาณรบกวนความถี่สูงออกไปได้ และยังคงรักษารูปสัญญาณที่ต้องการเอาไว้ได้ แต่ก็ได้เป็นเหมือนอย่าง ในอุดมคติแต่ที่เราต้องการ ในทางปฏิบัติก็คือจะต้อง ให้ความมิตเพื่อช่นของ สัญญาณที่น้อย และถ้าสมมติให้สัญญาณรบกวนนั้น ได้ถูกกำจัดออกโดยการใช้วงจรกรองความถี่แบบบัตเตอร์เวิร์คในออร์เดอร์ที่ 5 ซึ่งจะมีการผลการตอบสนองต่อกรหน่วงเวลาเป็น ไปดังรูปที่ 3.4.3 และถ้า รวมเอากรหน่วงเวลาที่เกิดขึ้นในวงจรอีควอไลเซอร์ ซึ่งจะทำให้ผลรวมของกรหน่วงเวลาจะมีค่าโดย ประมาณว่าคงที่อยู่ในช่วง $\omega=0$ ถึง $\omega=0.9$ (ความถี่ที่ถูกรวมอลไลซ์แล้ว) ผลก็คือจะทำให้เอาท์พุทที่ได้ จะไม่เป็นรูปสัญญาณที่เป็นสามเหลี่ยมแต่ก็ยังคงเค้า โครงรูปสัญญาณสามเหลี่ยมเดิมอยู่

ในการใช้งานวงจรหน่วงเวลานั้นเราสามารถที่จะนำเอามาประยุกต์ใช้กับงานการติดต่อสื่อสาร ได้ อีกมาก แต่ในหัวข้อของ โครงงานนั้นเราได้นำเอาวงจรหน่วงเวลาใช้ในการทดสอบเพื่อให้เห็น ได้ถึง การทำงานของ โครงงานที่ได้ทำขึ้น ซึ่งมีการออกแบบเหมือนกับวงจรอีควอไลเซอร์ซึ่งต้องอาศัยหลาย วงจรมาต่ออนุกรมเข้าด้วยกันซึ่งค่าการหน่วงเวลาจะมีค่าที่เปลี่ยนแปลงไปกับค่าของความถี่ แต่จะต้องไม่ มีการลดทอนที่เป็นฟังก์ชันของความถี่ ซึ่งมีวงจรที่ให้คุณสมบัติตามที่กล่าวซึ่งก็คือวงจรที่ให้สัญญาณผ่าน ได้ตลอด (all pass filter) ซึ่งจะมีลักษณะแพทเทิร์นของ โพลและซีโร (pole-zero pattern) ดังแสดงในรูปที่ 3.4.4



รูปที่ 3.4.4 แสดงลักษณะแพทเทิร์นของ โพลและซีโร (pole-zero pattern) ของวงจรที่ให้สัญญาณผ่าน ได้ตลอด (all pass filter)

ในรูป 3.4.4(a) แสดงให้เห็นถึงการมีโพลและซีโรอย่างละหนึ่งตัวและมีระยะห่างจากจุดกำเนิดที่ เท่าๆ กัน

ในรูป 3.4.4(b) จะมีอยู่ทั้งสี่ควอดเรนต์ที่สมมาตรกัน โดยจำนวนของโพลและซีโรจะต้องมี จำนวนที่เท่ากัน โดยตำแหน่งของโพลและซีโรจะอยู่ในตำแหน่งที่ตรงกันข้ามทั้ง ในส่วนของจำนวนจริง และค่าจินตภาพ

ในรูปที่ 3.4.4 (c) เป็นผลรวมในรูปที่ 3.4.4 (a) และ 3.4.4 (b)

ในรูปที่ 3.4.4 (d) เป็นผลมาจากรูปที่ 3.4.4 (a) และ 3.4.4 (b) ซึ่งมีระยะห่างจากจุดกำเนิดที่ต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปเผยแพร่ในวงการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรมห้วงเวลาลำดับที่ 1

เมื่อพิจารณาทรานสเฟอริงฟังก์ชันของวงจรรองความถี่ที่ให้สัญญาณผ่านได้ตลอดจะเป็นดังนี้

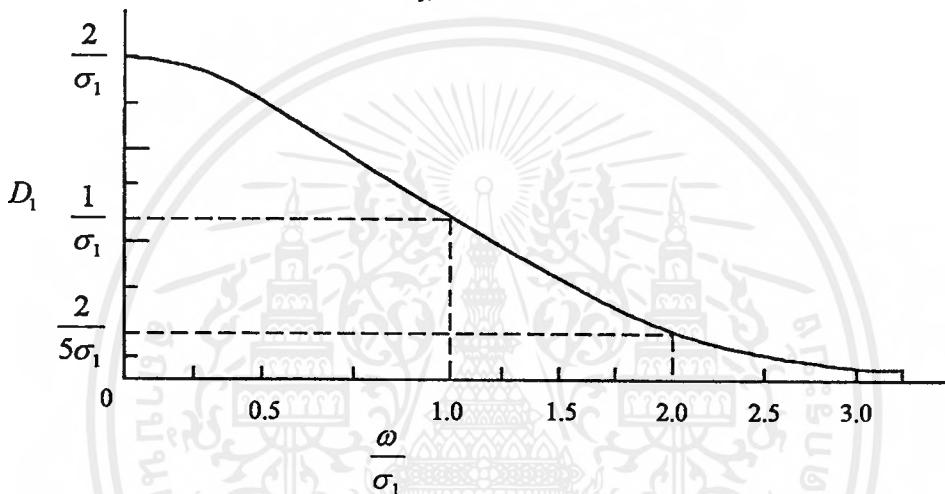
$$T_1(s) = K \frac{s - \sigma_1}{s + \sigma_1}$$

ซึ่งจะมีแพทเทิร์นของโพลและซีโรคั้งในรูปที่ 3.4.4 (a) ซึ่งเราสามารถสรุปได้เป็น

$$T_1(j\omega) = K \text{ และ } \theta = -2 \tan^{-1} \left(\frac{\omega}{\sigma_1} \right)$$

เราสามารถหาค่าของการหน่วงได้จาก θ ดังนี้

$$D_1(\omega) = -\frac{d}{d\omega} \theta(\omega) = \frac{2/\sigma_1}{1 + (\omega/\sigma_1)^2}$$

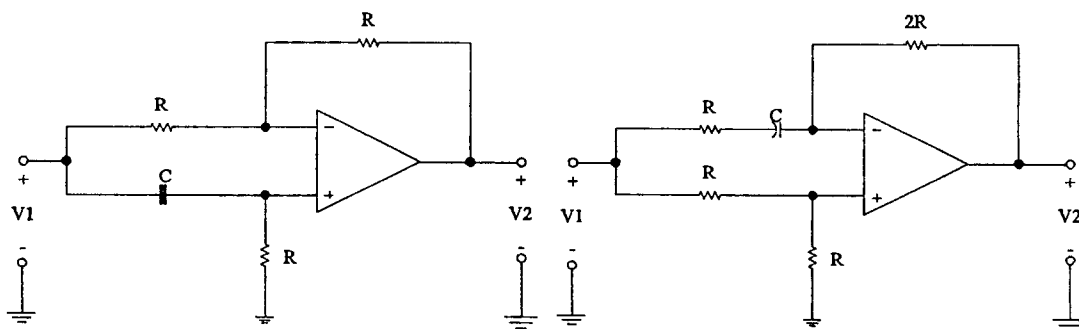


รูปที่ 3.4.5 แสดงค่าของการหน่วงเวลาที่อยู่ในฟังก์ชันความถี่

ซึ่งค่าของการหน่วงเวลาจะเป็นฟังก์ชันของความถี่ดังแสดงในรูปที่ 3.4.5 ซึ่งแสดงให้เห็นว่าถ้าความถี่มีค่ามากขึ้นจะทำให้ค่าของเวลาในการหน่วงจะมีค่าน้อยลง

$$D_{max} = D_1(0) = \frac{2}{\sigma_1}$$

จากสมการของทรานสเฟอริงฟังก์ชันคอนตันเราสามารถที่จะนำมาสร้าง โดยการใช้โอปแอมป์ดังแสดงในรูปที่ 3.4.6



รูปที่ 3.4.6 วงจรรอปแอมป์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 3.4.6 (a) เราจะมีตัวแปรคือ $K=1$ และ $\sigma=1/RC$ (เป็นวงจรแบบนอนอินเวอร์ต)

ในรูปที่ 3.4.6 (b) เราจะมีตัวแปรคือ $K=-1/2$ และ $\sigma=1/RC$ (เป็นวงจรแบบอินเวอร์ต)

และถ้าต่อวงจรกรองความถี่ต่ำเข้าร่วมกับวงจรนี้จะทำให้เกิดมีการหน่วงเวลาอยู่ 2 ส่วน ซึ่งเราจะได้ค่าของการหน่วงเวลารวมเป็น

$$D_t(\omega) = D_B(\omega) + D_1(\omega)$$

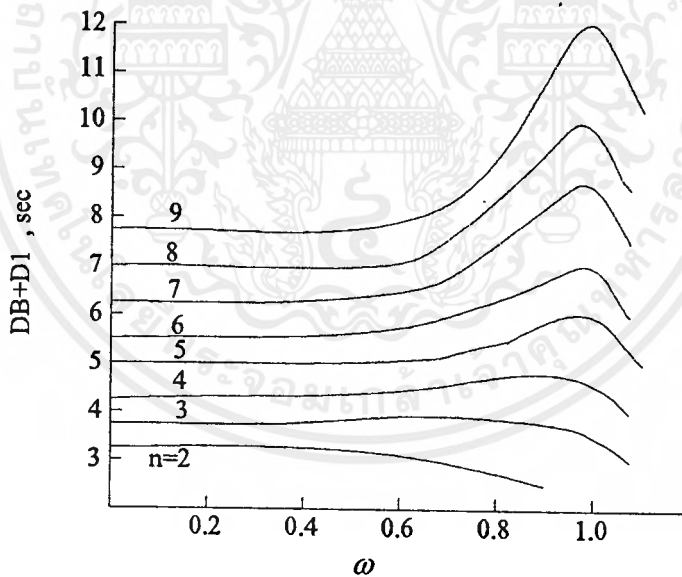
ซึ่งจากรูปแสดงผลของ $D_1(\omega)$ เราจะเห็นได้ว่าค่าของการหน่วงเวลาสูงที่สุดจะอยู่ที่ $\omega=0$ ดังนั้นจุดประสงค์ในการใช้งานของวงจรหน่วงเวลาลำดับที่ 1 ก็คือให้เกิดการหน่วงเวลาที่ความถี่ต่ำรวมอยู่ด้วย และเราสามารถที่จะทำให้ค่าของ $D_1(\omega)$ เข้าง่าย โดยจัดให้มีค่า

$$\sigma_1 = \left(2 \sin \frac{3\pi}{2n} \right)^{\frac{1}{3}}$$

ดังนั้นเราจะได้ค่าการหน่วงเวลาที่ความถี่เป็น 0 ได้ดังนี้

$$D_t(0) = \frac{1}{\sin(\pi/2n)} + \frac{2}{\sigma_1}$$

ค่าเวลาการหน่วงรวมสามารถที่จะแสดงให้เห็นได้ดังรูปที่ 3.4.7



รูปที่ 3.4.7 แสดงค่าการหน่วงเวลารวมของวงจรที่ใช้วงจรหน่วงเวลาลำดับที่ 1 ต่ออนุกรมกับวงจรกรองความถี่ต่ำแบบบัตเตอร์เวิร์คลำดับที่ n

การหน่วงเวลาโดยการใช้วงจรกรองความถี่ให้ผ่านได้ตลอดช่วงลำดับที่ 2 (Delay and Second-Order Allpass Function)

ค่าทรานสเฟอ์ฟังก์ชันของวงจรกรองความถี่ให้ผ่านได้ตลอดจะเป็นดังนี้

$$T_2(s) = \frac{s^2 - (\omega_o/Q)s + \omega_o^2}{s^2 + (\omega_o/Q)s + \omega_o^2}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการเราจะเห็นได้ว่ามีตัวแปรอยู่ 2 ตัว คือ ω_0 และ Q ซึ่งจะเป็นตัวกำหนดค่าของการ
 หน่วงเวลา

เฟสฟังก์ชันของสมการคือ

$$\arg\{T_2(s)\} = \theta_2(\omega) = -2 \tan^{-1}\left(\frac{\omega_0/Q}{\omega_0^2 - \omega^2}\right)$$

ทำการจัดรูปเพื่อให้ง่ายขึ้น โดยการให้ $\Omega = \omega/\omega_0$ เราจะได้เป็น

$$\theta_2(\Omega) = -2 \tan^{-1}\left(\frac{\Omega/Q}{1 - \Omega^2}\right)$$

ค่าของการหน่วงเวลาเป็นค่าที่ได้จากการดิฟเฟอเรนเชียล (differentiating the phase function)

$$D_2(\Omega) = \frac{d\theta_2}{d\Omega} \frac{d\Omega}{d\omega} = \frac{2}{\omega_0 \Omega} \frac{1 + \Omega^2}{(1 - \Omega^2) + \Omega^2/Q^2}$$

ที่ความเร็วเชิงมุมเป็น 0 หรือ ความถี่เป็น 0 จะทำให้ค่าการหน่วงเวลามีค่าเป็น

$$D_2(0) = \frac{2}{Q\omega_0}$$

หาค่าวิกฤตที่ทำให้เกิดค่าการหน่วงเวลาสูงที่สุดโดยการดิฟเฟอเรนเชียล $D_2(\Omega)$ แล้วจับให้เท่ากับ 0
 เราจะได้

$$\Omega^4 + 2\Omega^2 + \left(\frac{1}{Q^2} - 3\right) = 0$$

และคำตอบของสมการที่ได้เป็น

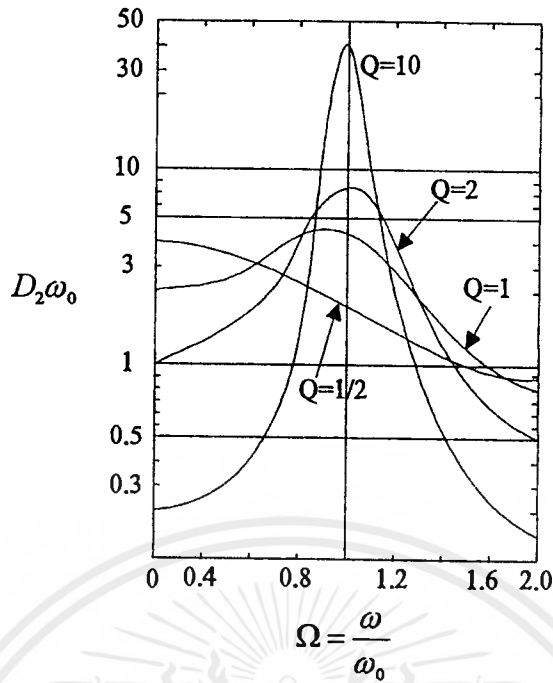
$$\Omega_{\max} = \sqrt{-1 + \sqrt{4 - \frac{1}{Q^2}}}$$

เมื่อเราให้ค่า $Q > 1$ แล้ว $\Omega_{\max} \approx 1$ หรือ $\omega_{\max} = \omega_0$ และถ้า Ω มีค่ามากๆ แล้วเราจะได้ว่า

$$\omega_0 D_2(\infty) \longrightarrow 0 \quad (\text{slope} = -12\text{dB per octave})$$

เมื่อเรานำค่าการหน่วงเวลามาพล็อตกราฟเทียบกับค่า Q เราจะได้ดังรูปที่ 3.4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



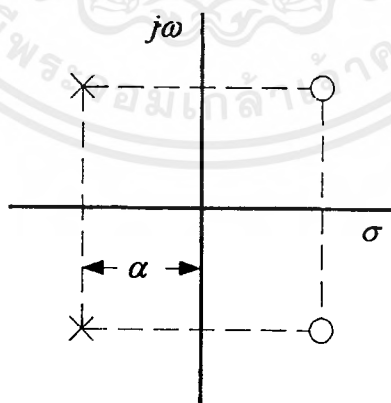
รูปที่ 3.4.8 แสดงค่าการหน่วงเวลามาพล็อตกราฟเทียบกับค่า Q

จากรูปเมื่อเราแก้สมการ โดยการแทนค่า $\Omega_{\max} = 1$ แล้วจะได้

$$(D_2\omega_0)_{\max} = 4Q$$

ความสัมพันธ์ของสมการนี้จะมีความเกี่ยวข้องกับตำแหน่งของโพลและซีโรซึ่งในตำแหน่งที่เป็นค่าจำนวนจริงนั้นจะมีระยะห่างจากจุดกำเนิดเป็น α ดังในรูปที่ 3.4.9 และจากรูปเรารู้ว่าในส่วนค่าจำนวนจริงนี้สามารถหาค่าได้เป็น $\alpha = \omega_0 / 2Q$ และทำให้ได้ค่าการหน่วงเวลาสูงที่สุดเป็น

$$D_{2\max} = \frac{2}{\alpha}$$



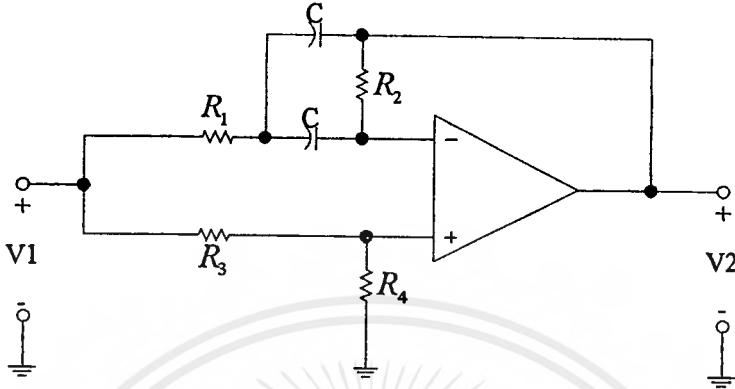
รูปที่ 3.4.9 แสดง โพลและซีโรใน s-plane

ข้อสังเกต ความสัมพันธ์ระหว่างค่าของการหน่วงเวลาสูงที่สุดและ s-plane แสดงตำแหน่งของโพลและซีโร โดยจะสามารถใช้ได้กับค่าของ Q ที่มากกว่า 1 เท่านั้น

วงจรกรองความถี่ผ่านตลอดลำดับที่ 2

วงจรที่ใช้งานเพื่อให้ได้ทรานสเฟอร์ฟังก์ชันของวงจรกรองความถี่ผ่านตลอดนั้นสามารถที่จะสร้างได้ทั้งวงจรที่เป็นแอคทีฟและพาสซีฟ แต่ในที่นี้จะใช้วงจรของ Delyiannis ดังที่ได้แสดงให้เห็นดัง ในรูปที่

3.4.10



รูปที่ 3.4.10 วงจรของ Delyiannis

แรงดันที่ขานอนอินเวอร์ตของออปแอมป์จะมีค่าเป็น

$$V_c = \frac{R_4}{R_3 + R_4} V_1 = kV_1$$

และ $V_c = V_B$ เราจะใช้การวิเคราะห์โดยการใช่วิธีของ โนด พิจารณาที่ โนด A และ โนด B ที่ โนด A ใช้กฎกระแสของเคอร์ชอฟท์ ซึ่งได้สมการออกมาดังนี้

$$\left(2Cs + \frac{1}{R_1}\right)V_A + (-Cs)V_2 - \left(\frac{1}{R_1} + kCs\right)V_1 = 0$$

ในทำนองเดียวกันที่ โนด B

$$-(Cs)V_A + \left(-\frac{1}{R_2}\right)V_2 + \left[k\left(Cs + \frac{1}{R_2}\right)\right]V_1 = 0$$

และเมื่อเรากำจัดตัวที่ไม่ทราบค่า (V_A) และทำการแก้สมการเพื่อหาค่าอัตราส่วน V_2/V_1

$$\frac{V_2}{V_1} = k \frac{s^2 + [2/R_2C + (1-1/k)/R_1C]s + 1/R_1R_2C^2}{s^2 + (2/R_2C)s + 1/R_1R_2C^2}$$

จากรูปทั่วไปของวงจรกรองความถี่ที่ผ่านตลอดเราจึงจับค่าที่อยู่หน้า s ทั้งตัวบนและตัวล่างให้เท่ากันดังนี้

$$\frac{2}{R_2C} + \frac{1-1/k}{R_1C} = -\frac{2}{R_2C}$$

เราสามารถจัดให้ง่ายเข้าดังนี้

$$\frac{R_2}{R_1} = 4 \frac{R_4}{R_3}$$

หรือ

$$\frac{R_4}{R_3} = \frac{1}{4} \frac{R_2}{R_1}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อให้ $\omega_o = 1$ และ $R_1 = 1$ และ $R_2 = 4Q^2$

นั่นคือ $\frac{R_4}{R_3} = Q^2$

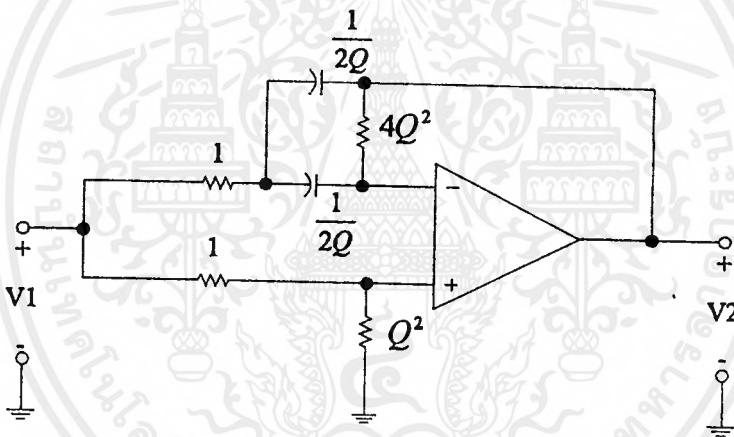
และถ้าเราให้ $R_4 = Q^2$ และ $R_3 = 1$

และเมื่อเราพิจารณาที่ความเร็วเชิงมุมเป็น 0 หรือที่ความถี่เป็น 0 ดังนั้นจึงมีทรานสเฟอ์ฟังก์ชันเป็น

$$T(0) = k = \frac{R_4}{R_3 + R_4} = \frac{Q^2}{1 + Q^2}$$

ดังนั้นเราสามารถที่จะสรุปทรานสเฟอ์ฟังก์ชันออกมาเป็น

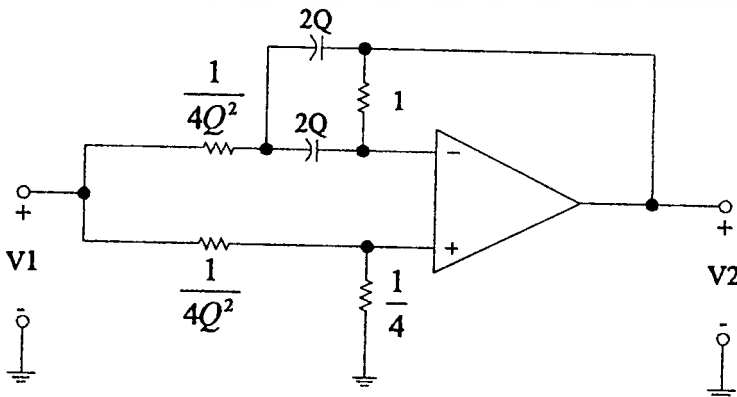
$$T(s) = \frac{Q^2}{1 + Q^2} \frac{s^2 - (1/Q)s + 1}{s^2 + (1/Q)s + 1}$$



รูปที่ 3.4.11 วงจรการหน่วงเวลาที่นอร์มอลไลซ์แล้วที่ $\omega_o = 1$

และถ้าเราทำการใช้สเกลถึงทางขนาด(magnitude Scaling)ด้วยค่า $k_m = 1/4Q^2$

เราจะได้ $R_1 = R_3 = \frac{1}{4Q^2}$, $R_2 = 1$, $R_4 = \frac{1}{4}$ และ $C = 2Q$



รูปที่ 3.4.12 วงจรการหน่วงเวลาที่นอร์มอลไลซ์แล้วที่ $\omega_o = 1$ และถูกสเกลถึง $k_m = 1/4Q^2$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองวงจรกรองความถี่ต่ำ

ลำดับขั้นการทดลอง

1. ค่่วงจรตามรูปที่ 3.3.2
2. ปรับแหล่งกำเนิดสัญญาณตามค่าความถี่ในตาราง โดยปรับระดับแรงดันแอมพลิจูดของสัญญาณไว้ที่ 1Vp-p บันทึกค่าแรงดันเอาต์พุตทุกลงในตาราง
3. ทำการปรับค่าความถี่วัดแรงดันอินพุตและเอาต์พุตบันทึก พร้อมทั้งคำนวณผลที่ได้ให้อยู่ในรูปเดซิเบล ลงในตาราง
4. เปรียบเทียบค่าที่ได้จริงกับค่าผลลัพธ์ที่ได้จากการ Simulate ในโปรแกรม Pspice
5. ค่่วงจรตามรูปที่ 3.3.3 แล้วดำเนินการจากขั้นที่ 2 ไปจนถึงขั้นที่ 4 บันทึกผลและทำการเปรียบเทียบ

ตารางที่ 4.1.1 แสดงผลการทดลองวงจรกรองความถี่ต่ำอันดับสี่ แบบ BUTTERWORTH

ความถี่ (Frequency) Hz	ระดับแรงดัน อินพุต(Vin) Volt	ระดับแรงดัน เอาต์พุต (Vout) Volt	เกณฑ์การขยาย (GAIN) เท่า	เกณฑ์การขยาย (GAIN) dB
500	1 Vp-p	1.040 Vp-p	1.040	0.3406
1000	1 Vp-p	1.024 Vp-p	1.024	0.2059
1500	1 Vp-p	1.008 Vp-p	1.008	0.0692
2000	1 Vp-p	0.992 Vp-p	0.992	-0.0697
2500	1 Vp-p	0.944 Vp-p	0.944	-0.5005
3000	1 Vp-p	0.816 Vp-p	0.816	-1.7662
3100	1 Vp-p	0.784 Vp-p	0.784	-2.1136
3200	1 Vp-p	0.732 Vp-p	0.732	-2.7097
3300	1 Vp-p	0.692 Vp-p	0.692	-3.1978
3400	1 Vp-p	0.656 Vp-p	0.656	-3.6619
3500	1 Vp-p	0.632 Vp-p	0.632	-3.9856
3600	1 Vp-p	0.572 Vp-p	0.572	-4.8521
3700	1 Vp-p	0.540 Vp-p	0.540	-5.3521
3800	1 Vp-p	0.496 Vp-p	0.496	-6.0903

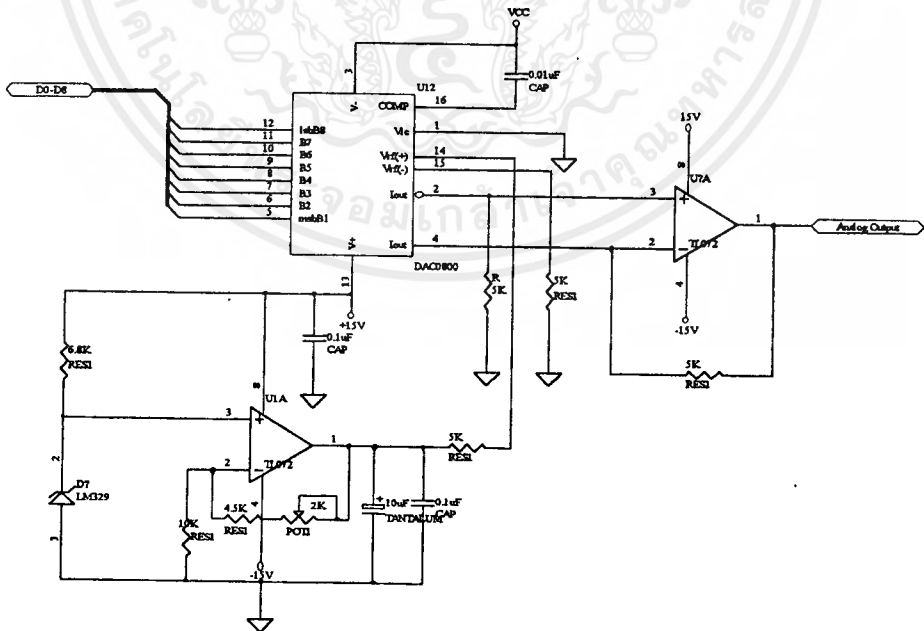
ความถี่ (Frequency) Hz	ระดับแรงดัน อินพุต(Vin) Volt	ระดับแรงดัน เอาต์พุต (Vout) Volt	เกนจ์การขยาย (GAIN) เท่า	เกนจ์การขยาย (GAIN) dB
3900	1 Vp-p	0.456 Vp-p	0.456	-6.8207
4000	1 Vp-p	0.424 Vp-p	0.424	-7.4527
4200	1 Vp-p	0.360 Vp-p	0.360	-8.8739
4400	1 Vp-p	0.312 Vp-p	0.312	-10.1169
4600	1 Vp-p	0.264 Vp-p	0.264	-11.5679
4800	1 Vp-p	0.232 Vp-p	0.232	-12.6902
5000	1 Vp-p	0.200 Vp-p	0.200	-13.9794
6000	1 Vp-p	0.104 Vp-p	0.104	-19.6593
7000	1 Vp-p	0.054 Vp-p	0.054	-25.3521
8000	1 Vp-p	0.036 Vp-p	0.036	-28.8739

ตารางที่ 4.1.2 แสดงผลการทดลองวงจรกรองความถี่ต่ำอันดับสอง แบบ BUTTERWORTH

ความถี่ (Frequency) Hz	ระดับแรงดัน อินพุต(Vin) Volt	ระดับแรงดัน เอาต์พุต (Vout) Volt	เกนจ์การขยาย (GAIN) เท่า	เกนจ์การขยาย (GAIN) dB
500	1 Vp-p	1.0000 Vp-p	1.0000	0.0000
1000	1 Vp-p	0.9938 Vp-p	0.9938	-0.0540
1500	1 Vp-p	0.9688 Vp-p	0.9688	-0.2753
2000	1 Vp-p	0.9313 Vp-p	0.9313	-0.6182
2500	1 Vp-p	0.8625 Vp-p	0.8625	-1.2848
3000	1 Vp-p	0.7812 Vp-p	0.7812	-2.1447
3100	1 Vp-p	0.7562 Vp-p	0.7562	-2.4272
3200	1 Vp-p	0.7375 Vp-p	0.7375	-2.6447
3300	1 Vp-p	0.7250 Vp-p	0.7250	-2.7932
3400	1 Vp-p	0.7062 Vp-p	0.7062	-3.2545
3500	1 Vp-p	0.6875 Vp-p	0.6875	-3.2545
3600	1 Vp-p	0.6625 Vp-p	0.6625	-3.5763
3700	1 Vp-p	0.6500 Vp-p	0.6500	-3.7417

ความถี่ (Frequency) Hz	ระดับแรงดัน อินพุท (Vin) Volt	ระดับแรงดัน เอาต์พุท (Vout) Volt	เกนซ์การขยาย (GAIN) เท่า	เกนซ์การขยาย (GAIN) dB
3800	1 Vp-p	0.6250 Vp-p	0.6250	-4.0824
3900	1 Vp-p	0.6125 Vp-p	0.6125	-4.2578
4000	1 Vp-p	0.5937 Vp-p	0.5937	-4.5286
4200	1 Vp-p	0.5500 Vp-p	0.5500	-5.1927
4400	1 Vp-p	0.5125 Vp-p	0.5125	-5.8061
4600	1 Vp-p	0.4844 Vp-p	0.4844	-6.2959
4800	1 Vp-p	0.4563 Vp-p	0.4563	-6.8149
5000	1 Vp-p	0.4281 Vp-p	0.4281	-6.5802
6000	1 Vp-p	0.3188 Vp-p	0.3188	-7.3691
7000	1 Vp-p	0.2438 Vp-p	0.2438	-12.259
8000	1 Vp-p	0.1938 Vp-p	0.1938	-14.253

4.2 ผลการทดลองวงจร Digital to Analog Conversion (DAC)



รูปที่ 4.2.1 แสดงวงจร Digital to Analog conversion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2.1 แสดงผลการทดลองวงจร DAC

ระดับ LOGIC	ระดับแรงดัน	ระดับ LOGIC	ระดับแรงดัน	ระดับ LOGIC	ระดับแรงดัน
0000 0000	-9.94	0001 1100	-7.72	0011 1000	-5.51
0000 0001	-9.96	0001 1101	-7.64	0011 1001	-5.43
0000 0010	-9.78	0001 1110	-7.56	0011 1010	-5.35
0000 0011	-9.70	0001 1111	-7.48	0011 1011	-5.27
0000 0100	-9.62	0010 0000	-7.40	0011 1100	-5.19
0000 0101	-9.54	0010 0001	-7.33	0011 1101	-5.12
0000 0110	-9.46	0010 0010	-7.25	0011 1110	-5.04
0000 0111	-9.39	0010 0011	-7.17	0011 1111	-4.96
0000 1000	-9.30	0010 0100	-7.09	0100 0000	-4.87
0000 1001	-9.22	0010 0101	-7.02	0100 0001	-4.79
0000 1010	-9.14	0010 0110	-6.94	0100 0010	-4.71
0000 1011	-9.07	0010 0111	-6.86	0100 0011	-4.63
0000 1100	-8.99	0010 1000	-6.77	0100 0100	-4.55
0000 1101	-8.91	0010 1001	-6.70	0100 0101	-4.48
0000 1110	-8.83	0010 1010	-6.62	0100 0101	-4.40
0000 1111	-8.75	0010 1011	-6.54	0100 0110	-4.32
0001 0000	-8.66	0010 1100	-6.47	0100 0111	-4.23
0001 0001	-8.59	0010 1101	-6.39	0100 1000	-4.16
0001 0010	-8.51	0010 1110	-6.31	0100 1001	-4.08
0001 0011	-8.43	0010 1111	-6.23	0100 1010	-4.00
0001 0100	-8.35	0011 0000	-6.14	0100 1011	-3.92
0001 0101	-8.27	0011 0001	-6.06	0100 1100	-3.84
0001 0110	-8.19	0011 0010	-5.98	0100 1101	-3.76
0001 0111	-8.12	0011 0011	-5.90	0100 1110	-3.68
0001 1000	-8.03	0011 0100	-5.82	0100 1111	-3.60
0001 1001	-7.95	0011 0101	-5.74	0101 0000	-3.52
0001 1010	-7.88	0011 0110	-5.67	0101 0001	-3.44
0001 1011	-7.80	0011 0111	-5.59	0101 0010	-3.36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

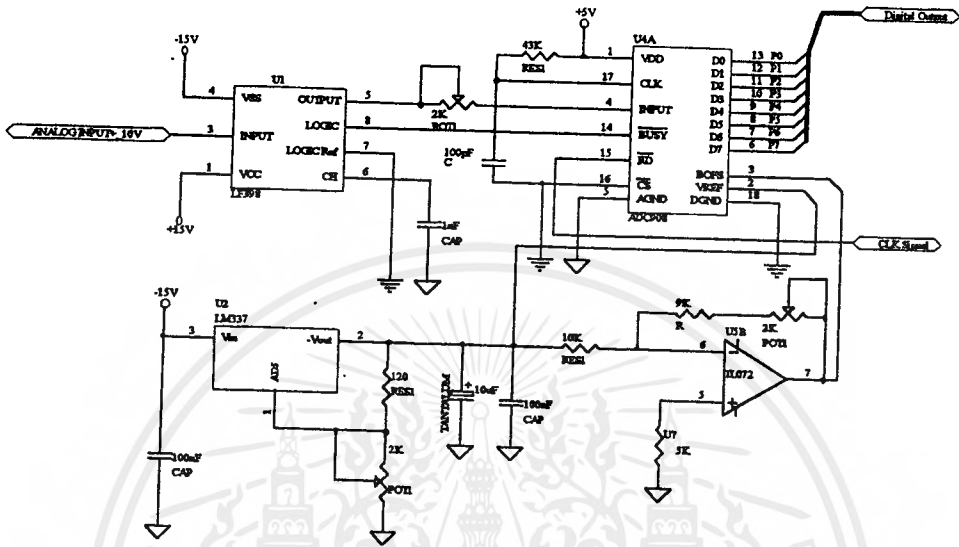
ระดับ LOGIC	ระดับแรงดัน	ระดับ LOGIC	ระดับแรงดัน	ระดับ LOGIC	ระดับแรงดัน
0101 0011	-3.28	0111 0000	-1.190	1000 1101	1.071
0101 0100	-3.20	0111 0001	-1.112	1000 1110	1.150
0101 0101	-3.13	0111 0010	-1.034	1000 1111	1.228
0101 0110	-3.05	0111 0011	-0.956	1001 0000	1.317
0101 0111	-2.96	0111 0100	-0.878	1001 0001	1.395
0101 1000	-2.89	0111 0101	-0.800	1001 0010	1.473
0101 1001	-2.81	0111 0110	-0.721	1001 0011	1.552
0101 1010	-2.81	0111 0111	-0.643	1001 0100	1.629
0101 1011	-2.73	0111 1000	-0.559	1001 0101	1.708
0101 1100	-2.65	0111 1001	-0.481	1001 0110	1.786
0101 1101	-2.57	0111 1010	-0.402	1001 0111	1.865
0101 1110	-2.50	0111 1011	-0.324	1001 1000	1.948
0101 1111	-2.42	0111 1100	-0.247	1001 1001	1.91
0110 0000	-2.34	0111 1101	-0.1809	1001 1010	1.98
0110 0001	-2.26	0111 1110	-0.1021	1001 1011	2.06
0110 0010	-2.18	0111 1111	-0.0241	1001 1100	2.14
0110 0011	-2.10	1000 0000	0.0612	1001 1101	2.22
0110 0100	-2.02	1000 0001	0.1393	1001 1110	2.30
0110 0101	-1.94	1000 0010	0.206	1001 1111	2.38
0110 0110	-1.87	1000 0011	0.284	1010 0000	2.45
0110 0111	-1.91	1000 0100	0.361	1010 0001	2.53
0110 1000	-1.827	1000 0101	0.440	1010 0010	2.61
0110 1001	-1.749	1000 0110	0.518	1010 0011	2.69
0110 1010	-1.670	1000 0111	0.596	1010 0100	2.76
0110 1011	-1.592	1000 1000	0.680	1010 0101	2.85
0110 1100	-1.514	1000 1001	0.759	1010 0110	2.92
0110 1101	-1.436	1000 1010	0.838	1010 0111	3.00
0110 1110	-1.357	1000 1011	0.916	1010 1000	3.09
0110 1111	-1.279	1000 1100	0.993	1010 1001	3.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลองวงจร Analog to Digital Conversion (ADC)

ลำดับขั้นการทดลอง ADC

1. ค่วงจรดังรูปที่ 4.3.1



รูปที่ 4.3.1 แสดงการต่อวงจร Analog to Digital Conversion

2. เปิดสวิตช์ POWER ของแหล่งจ่ายไฟฟ้า
3. ค่ สัญญาณนาฬิกาจากภายนอกเข้าสู่วงจร ADC ที่ขา RD เพื่อให้อยู่ในกระบวนการของการอ่านข้อมูลโดยการทำงานของวงจร ADC จะอยู่ในโหมดของรอม (ROM MODE) และจะต่อใช้งานแบบ Offset Binary Operation
4. ทำการปรับแต่งค่าเพื่อใช้ในการทดลอง ซึ่งมีขั้นตอนดังนี้
 - ปรับแรงดันอ้างอิง $V_{ref} = 10\text{ V}$ ที่ขา 2
 - ปรับค่าความต้านทานแบบปรับค่าได้ R_2 และ R_3 ไว้ที่ค่าบริเวณกลางๆ
 - ป้อนแรงดันขนาด $+10\text{ V}$ เข้าที่ขา อินพุท ปรับค่า R_2 จน LED ทุกตัวที่ต่ออยู่กับขาสัญญาณเอาต์พุตติดสว่างทุกตัว
 - ป้อนแรงดันขนาด 0 Volt เข้าที่ อินพุทแล้วจะต้องได้ค่าเป็น 1000 0000 หรือ 01111 1111
 - ป้อนแรงดันขนาด -10 Volt เข้าที่อินพุท ปรับค่า R_3 จน LED ทุกตัวดับสนิท
 - ทดลองป้อนค่าแรงดัน -9.92 Volt จะต้องให้ผลลัพธ์ออกมาเป็น 1111 1110 $\pm 1\text{ Bit}$ ถ้าไม่ได้ให้ไปเริ่มต้นการปรับแต่งใหม่
5. ทดลองสุ่มค่าแรงดันป้อนเข้าสู่วงจร โดยให้เพิ่มแรงดันเป็นขั้นๆ ทีละ 0.5V บันทึกผลลงในตาราง

ตารางที่ 4.3.1 แสดงผลการทดลองวงจร ADC

ระดับแรงดัน	ระดับ LOGIC	ระดับแรงดัน	ระดับ LOGIC
10 V	1111 1111	-10 V	0000 0000
9.5 V	1111 1001	-9.5 V	0000 0111
9.0 V	1111 0011	-9.0 V	0000 1110
8.5 V	1110 1100	-8.5 V	0001 0100
8.0 V	1110 0110	-8.0 V	0001 1010
7.5 V	1110 0000	-7.5 V	0010 0010
7.0 V	1101 1001	-7.0 V	0010 0111
6.5 V	1101 0011	-6.5 V	0010 1110
6.0 V	1100 1100	-6.0 V	0011 0011
5.5 V	1100 0111	-5.5 V	0011 1010
5.0 V	1100 0000	-5.0 V	0100 0000
4.5 V	1011 1101	-4.5 V	0100 0110
4.0 V	1011 0011	-4.0 V	0100 1100
3.5 V	1010 1100	-3.5 V	0101 0011
3.0 V	1010 0110	-3.0 V	0101 1001
2.5 V	1001 1111	-2.5 V	0110 0000
2.0 V	1001 1001	-2.0 V	0110 0110
1.5 V	1001 0011	-1.5 V	0110 1100
1.0 V	1000 1100	-1.0 V	0111 0011
0.5 V	1000 0110	-0.5 V	0111 1001
0.0 V	1000 0000		

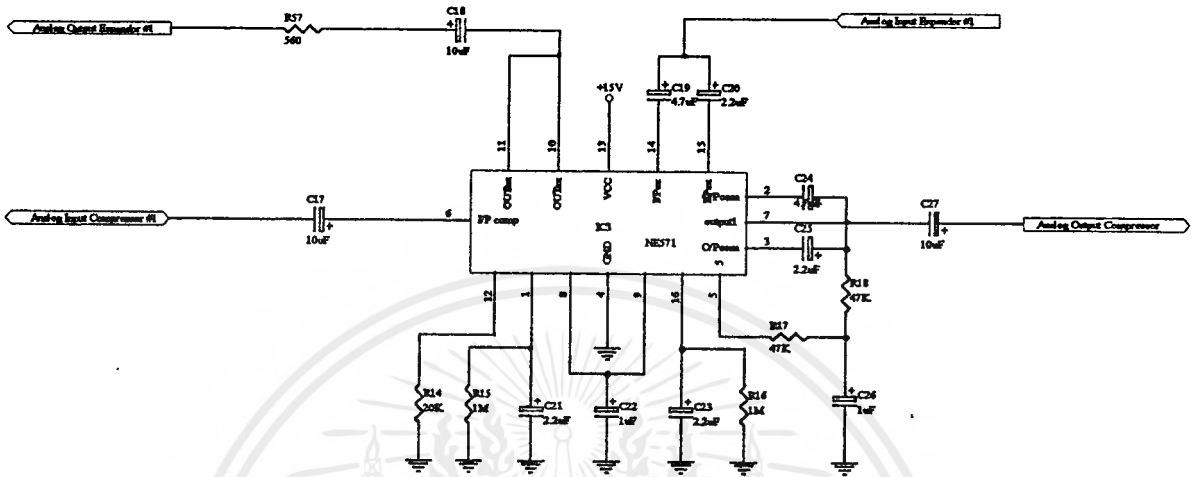
เราสามารถทดสอบเปรียบเทียบผลการทดลองได้โดยวิธีง่ายๆ คือ

เนื่องจาก Data Sheet ของ ADC908 ในจะให้แรงดันในแต่ละขั้นที่แตกต่างกัน 80 mV แล้วเทียบหาค่าระดับของ Step ที่แตกต่างกัน เช่น ที่แรงดันอินพุตเป็น 2.5 Volt จะมีระดับที่แตกต่างกันอยู่คือ $(2.5/0.8) = 31$ ขั้น แต่เรารู้ว่าที่ระดับแรงดันเป็นกราวด์เอาท์พุตที่ได้จะเป็น 1000 0000 เราสามารถที่จะบวกด้วยค่าจำนวนขั้นได้เป็น $(1000\ 0000B + 31d = 1001\ 1111B)$

4.4 ผลการทดลองวงจร COMPANDER

ลำดับขั้นการทดลอง

1. ค่วงจรดังรูปที่ 4.4.1



รูปที่ 4.4.1 แสดงการต่อวงจร COMPANDER

2. เปิดสวิตช์ POWER ของแหล่งจ่ายไฟที่
3. บ้อน สัญญาณความถี่ 1 KHz ที่ทางด้าน INPUT ทำการวัดค่าทาง OUTPUT แล้วทำการเปลี่ยนแอมพลิจูดของสัญญาณ INPUT บันทึกค่า OUTPUT ที่ได้แต่ละค่าลงในตารางตามลำดับ

ตารางที่ 4.4.1 แสดงผลการทดลองวงจร COMPANDER

INPUT	OUTPUT	
	Compressor	Expander
14 Vp-p	5.92 Vp-p	
13 Vp-p	5.52 Vp-p	
12 Vp-p	5.28 Vp-p	
11 Vp-p	5.12 Vp-p	
10 Vp-p	4.8 Vp-p	(เพิ่มขึ้น)
9 Vp-p	4.64 Vp-p	
8 Vp-p	4.32 Vp-p	
7 Vp-p	4.08 Vp-p	
6 Vp-p	3.76 Vp-p	
5 Vp-p	3.36 Vp-p	

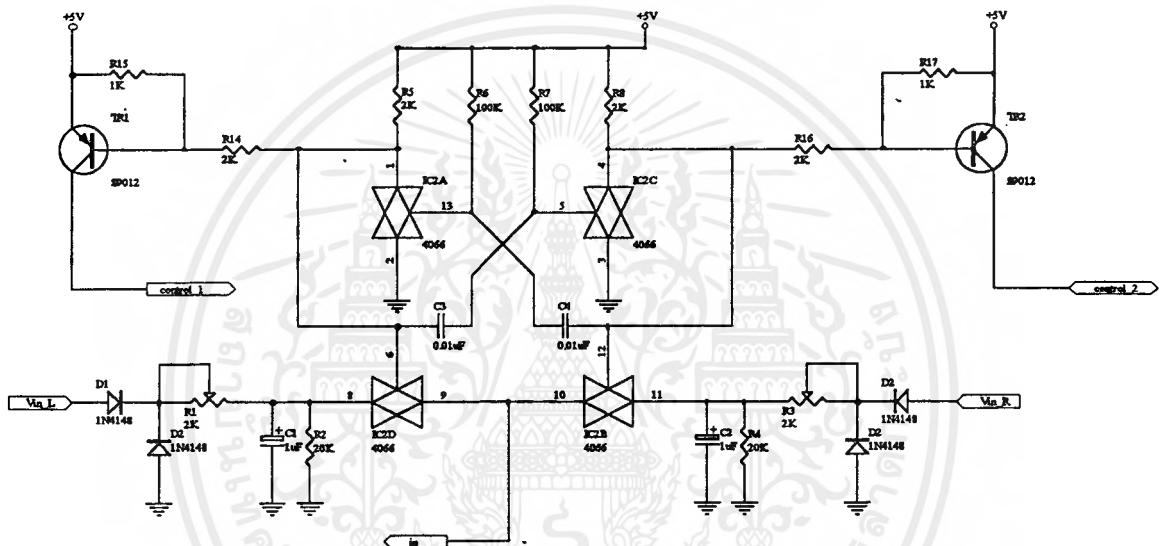
INPUT	OUTPUT	
	Compressor	Expander
4 Vp-p	3.04 Vp-p	6.62 Vp-p
3 Vp-p	2.68 Vp-p	3.6 Vp-p
2 Vp-p	2.16 Vp-p	1.67 Vp-p
1 Vp-p	1.52 Vp-p	372 mVp-p
950 mVp-p	1.48 Vp-p	323 mVp-p
900 mVp-p	1.44 Vp-p	292 mVp-p
850 mVp-p	1.4 Vp-p	264 mVp-p
800 mVp-p	1.36 Vp-p	236 mVp-p
750 mVp-p	1.31 Vp-p	227 mVp-p
700 mVp-p	1.27 Vp-p	211 mVp-p
650 mVp-p	1.21 Vp-p	187 mVp-p
600 mVp-p	1.18 Vp-p	166 mVp-p
550 mVp-p	1.12 Vp-p	148 mVp-p
500 mVp-p	1.08 Vp-p	121 mVp-p
450 mVp-p	1.02 Vp-p	83 mVp-p
400 mVp-p	960 mVp-p	62 mVp-p
350 mVp-p	900 mVp-p	48 mVp-p
300 mVp-p	810 mVp-p	37 mVp-p
250 mVp-p	740 mVp-p	28 mVp-p
200 mVp-p	680 mVp-p	20 mVp-p
150 mVp-p	580 mVp-p	14 mVp-p

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ผลการทดลองของวงจรการกำเนิดสัญญาณสี่เหลี่ยมที่ใช้ IC 4066 ของวงจรแตรวงผล

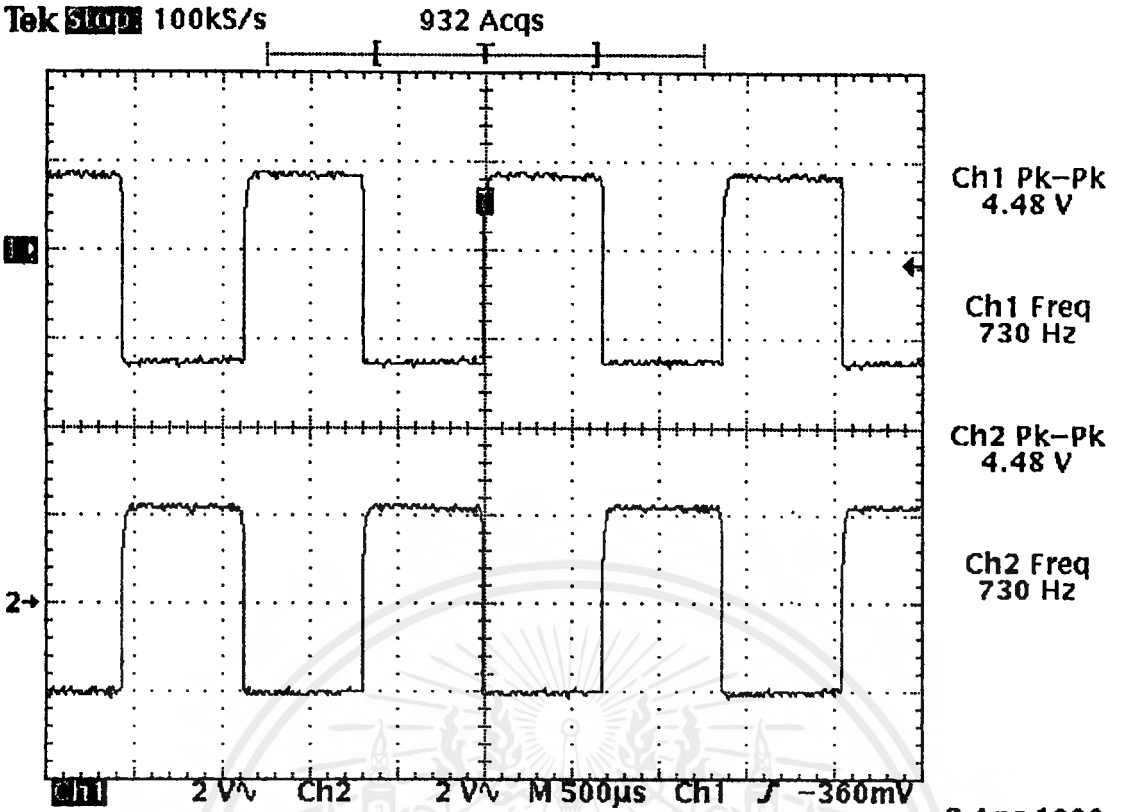
ลำดับขั้นการทดลอง

1. ค่่วงจรดังรูปที่ 4.5.1



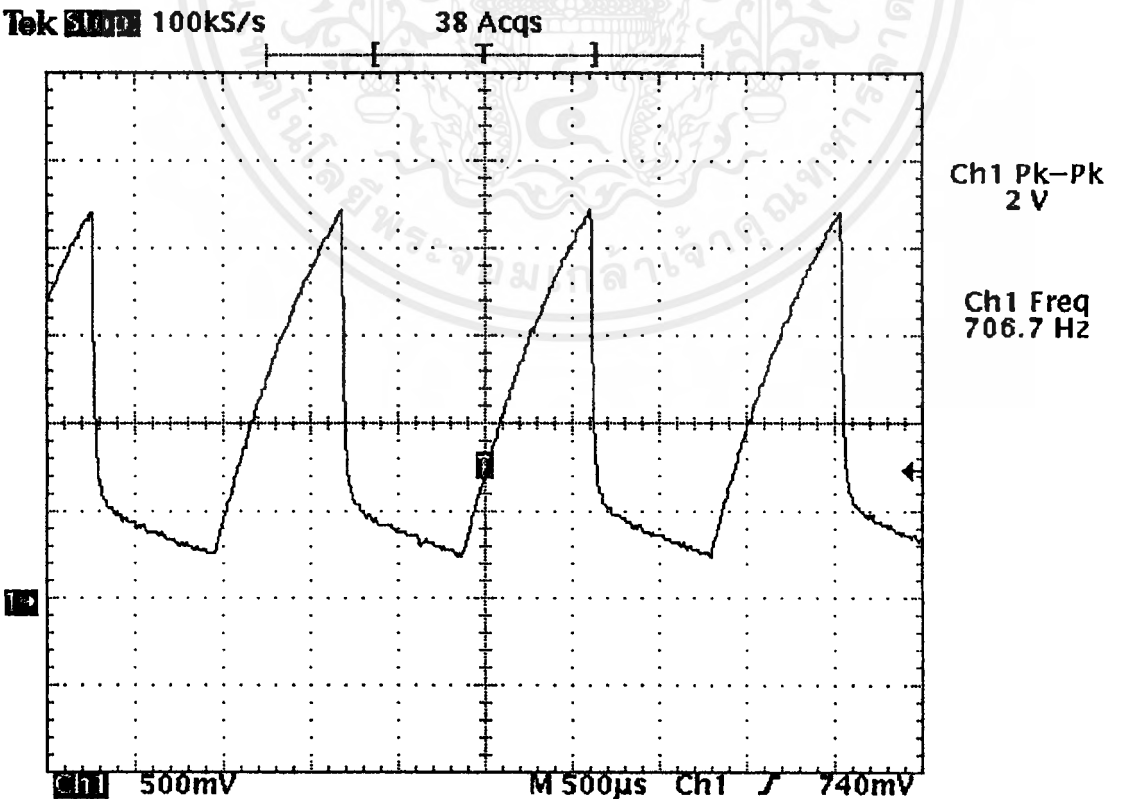
รูปที่ 4.5.1 แสดงการค่่วงจรกำเนิดสัญญาณสี่เหลี่ยมที่ใช้ IC 4066

2. เปิดสวิตช์ POWER ของแหล่งจ่ายไฟฟ้า
3. ใช้ Oscilloscope วัดค่าสัญญาณเอาต์พุตที่เกิดขึ้น แล้วทำการบันทึกรูปสัญญาณเอาต์พุต



2 Apr 1999 03:20:17

รูปที่ 4.5.2 แสดงสัญญาณเอาต์พุตที่ได้จากวงจรกำเนิดสัญญาณ IC 4066



2 Apr 1999 03:27:25

เอกสารนี้เป็นเอกสารที่สงวนรูปที่ 4.5.3 แสดงรูปสัญญาณที่ตรวจพร้อมตัวเก็บประจุ C3 หรือ C4 ั้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การเขียนโปรแกรมคำสั่งภาษาแอสเซมบลีของ DSP

การทำงานของโปรแกรมคำสั่งภาษาแอสเซมบลี

(.set) ส่วนคั่นของโปรแกรมเป็นการ define ค่าตัวแปรต่างๆ ในโปรแกรม

(.mmregs) เป็นการให้ Assembler เข้าใจถึงตัวแปรให้ตรงกับ memory map register

(.ps) program start

- (.ps) ส่วนแรกเป็น Interrupt vector
- (.ps) ส่วนที่สองเป็น save run

Routine 1 Initial โปรแกรมเซอร์ DSP โดยการเซตค่ารีจิสเตอร์ในการคำนวณและรีจิสเตอร์ที่ใช้ในการเซตค่าพื้นที่หน่วยความจำ และเซตค่าในหน่วยความจำให้เป็นไปตามที่ต้องการให้เริ่มต้น

Routine 2 Initial อุปกรณ์ ADC เพื่อให้พร้อมในการรับข้อมูล

Routine 3 กำหนดชนิดของการเกิดอินเตอร์รัพท์ ว่าต้องการให้เกิดอินเตอร์รัพท์ประเภทใดบ้าง และเซตให้อินเตอร์รัพท์เริ่มทำงาน

Routine 4 รอการอินเตอร์รัพท์ ถ้ามีการเกิดอินเตอร์รัพท์ให้กระโดดไปรับข้อมูลแล้วทำงานในโปรแกรมหลัก

งานหลักที่มีความสำคัญมากคือคุณจะต้องกระทำก่อนเริ่มการประยุกต์ใช้งาน คือ การ Initial DSK การ Initial ถูกกระทำในช่วง Software ก่อนหน้าที่เข้าไปในการใช้งานโปรแกรมของคุณ และจะต้องถูกทำการ Initial ถ้าอุปกรณ์ AIC ถูกใช้งาน

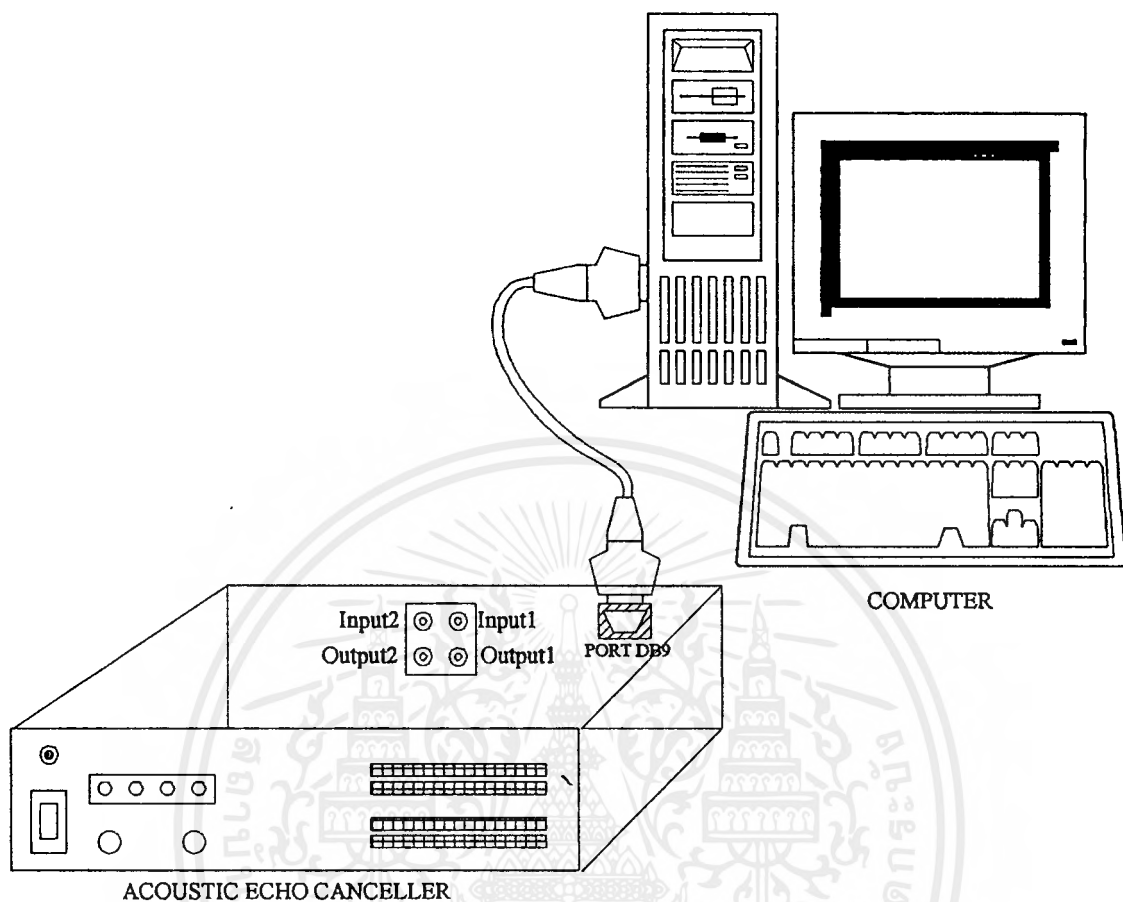
ถ้าการ Initial นี้ไม่ถูกกระทำ การประยุกต์ใช้งานโปรแกรม DSK ของคุณจะไม่มีการ RUN อย่างแท้จริง ในช่วงเวลาอื่นการประยุกต์ใช้งานโปรแกรมดูเหมือนจะมีการ RUN แต่ไม่มีอะไรจะเกิดขึ้น มี

2 อย่างที่จะต้องถูก Initial

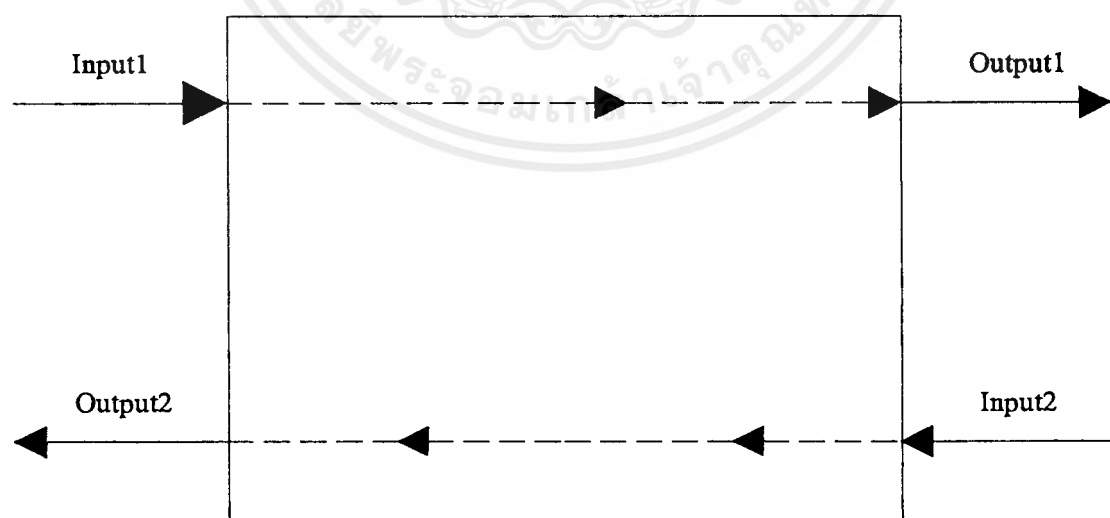
1. The TMS320C50 register ต่างๆ ที่ใช้งาน
2. The TMS320C5x on-chip timer
3. การเกิดอินเตอร์รัพท์ ที่ต้องการใช้งาน

สำหรับโปรแกรมหลัก Acoustic Echo Canceller ได้ถูกเขียนขึ้น โดยมีหลักการทำงานดังที่แสดงในโฟลว์ชาร์ต ซึ่งแสดงขั้นตอนต่างๆ ในการทำงานของโปรแกรม

4.7 ผลการทดลองเครื่องกำจัดเสียงก้อง(Acoustic Echo Canceller)

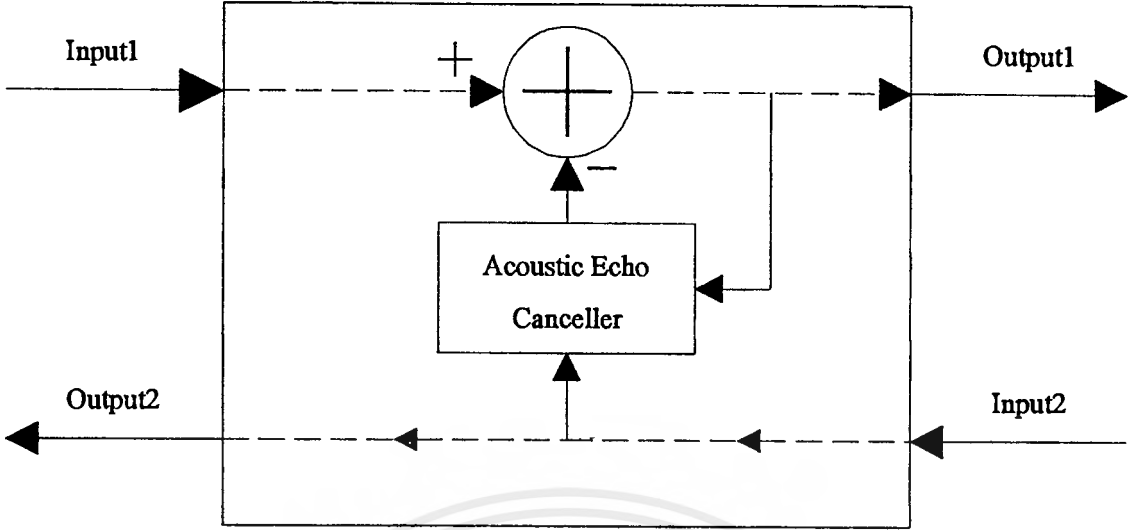


รูปที่ 4.7.1 แสดงการต่อใช้งานของเครื่องกำจัดเสียงก้อง(Acoustic Echo Canceller)



(a) เครื่องกำจัดเสียงก้องที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้อง

รูปที่ 4.7.2 (a) แสดงบล็อก ไดอะแกรมการทำงานของเครื่องที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้อง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



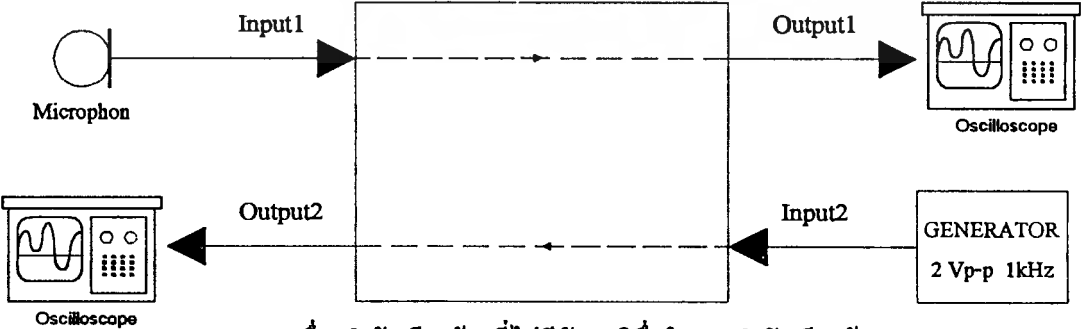
(b) เครื่องกำจัดเสียงก้องที่มีอัลกอริทึมในการกำจัดเสียงก้อง

รูปที่ 4.7.2 (b) แสดงบล็อก ไดอะแกรมการทำงานของเครื่องที่มีการรับอัลกอริทึมในการกำจัดเสียงก้อง

4.7.1 ผลการทดลองชุดรับส่งข้อมูลโดยที่ไม่ต้องผ่านชุดอัลกอริทึมการกำจัดเสียงสะท้อน



(a) เครื่องกำจัดเสียงก้องที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้อง



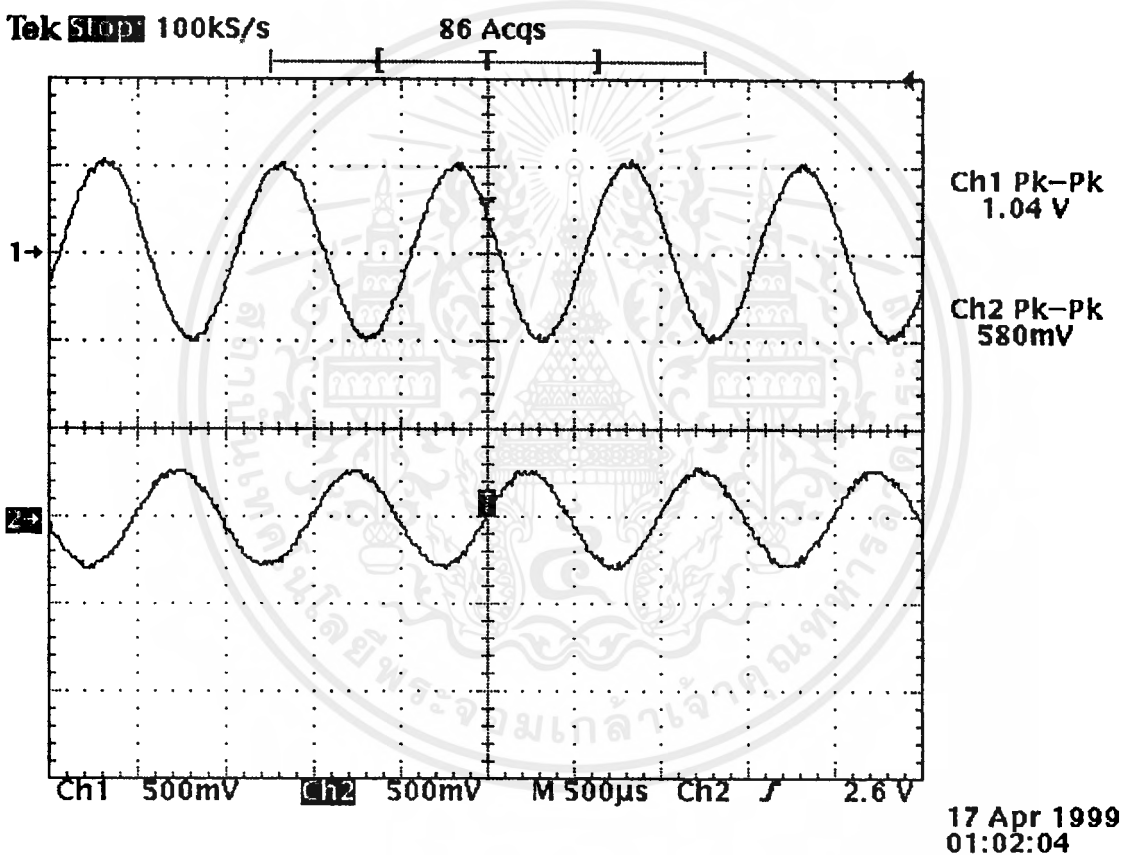
(b) เครื่องกำจัดเสียงก้องที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้อง

รูปที่ 4.7.1.1 บล็อก ไดอะแกรมการต่อทดลองการรับส่งข้อมูลของเครื่องกำจัดเสียงสะท้อนที่ไม่ได้ผ่านชุดอัลกอริทึมในการกำจัดเสียงก้อง

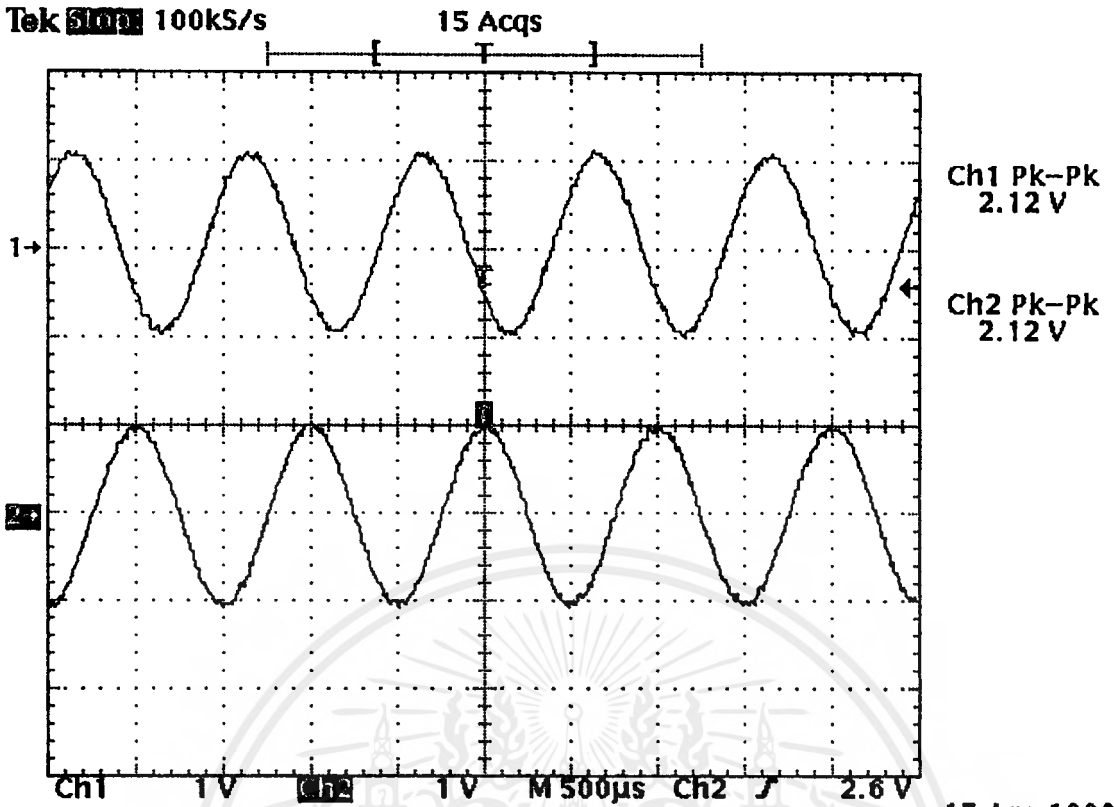
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทดลอง

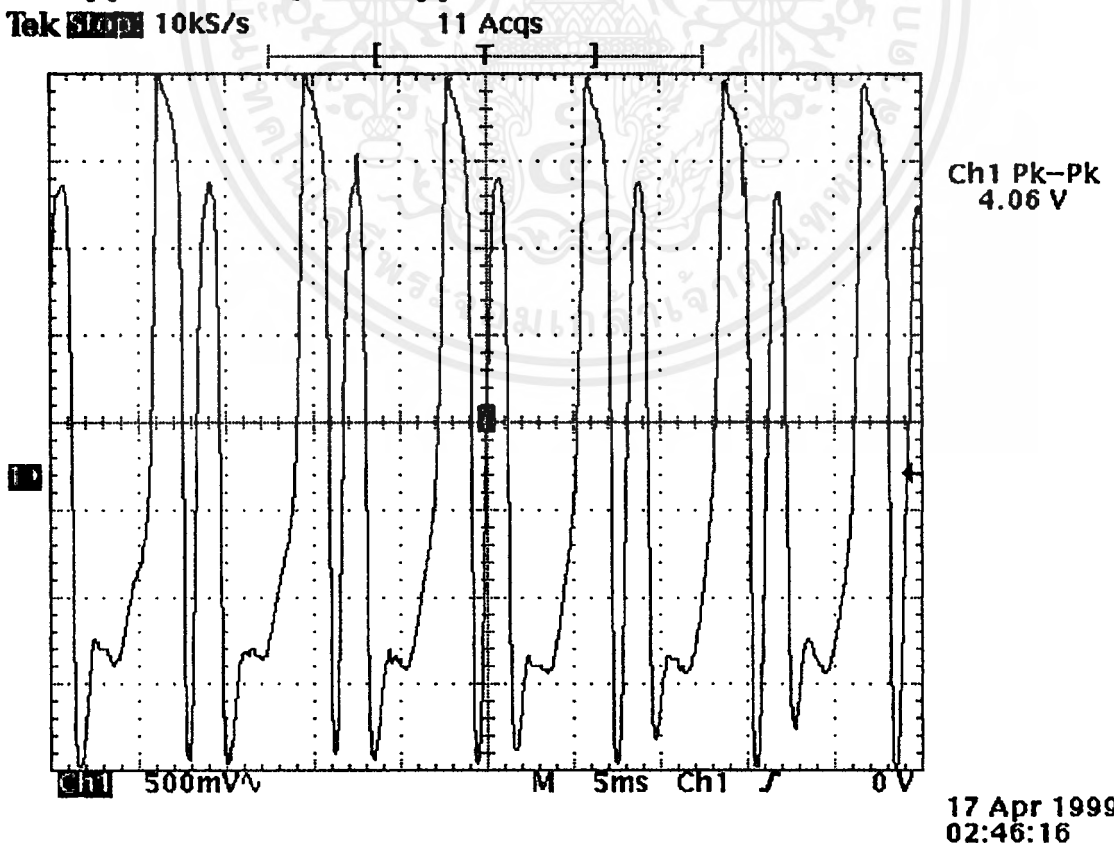
1. จากรูปที่ 4.7.1.1(a) ให้ต่อสัญญาณ sinewave ขนาด 2 Vp-p 1 kHz เข้าที่ Input2 และ สัญญาณ Sinewave 1 Vp-p 1kHz เข้าที่ Input1
2. ถ่ายไฟให้กับเครื่องกำเนิดเสียงก้อง แล้วทำการรัน โปรแกรมที่มีเพียงการรับส่งข้อมูลแต่ไม่มีชุดอัลกอริทึมในการกำเนิดเสียงก้อง ตามที่แสดง ในรูปที่ 4.7.1.1(a) แล้วทำการวัดสัญญาณที่ตำแหน่ง Output1 และ Output2 แล้วทำการบันทึกผล
3. เปลี่ยนสัญญาณ Sinewave ที่เข้าที่ Input1 เป็นสัญญาณจาก ไมโคร โฟนแทน ตามที่ได้แสดงดังรูป 4.7.1.1(b) แล้วทำการวัดสัญญาณที่ตำแหน่ง Output1 แล้วทำการบันทึกผล



รูปที่ 4.7.1.2 แสดงผลการทดลองที่ได้จากการวัดสัญญาณที่ตำแหน่ง Input1(Ch1) กับ Output1(Ch2) เมื่อป้อนสัญญาณที่ตำแหน่ง Input1 เป็นสัญญาณ Sinewave



รูปที่ 4.7.13 แสดงผลการทดลองที่ได้จากการวัดสัญญาณที่ตำแหน่ง Input2(Ch1) กับ Output2(Ch2) เมื่อป้อนสัญญาณที่ตำแหน่ง Input1 เป็นสัญญาณ Sinewave

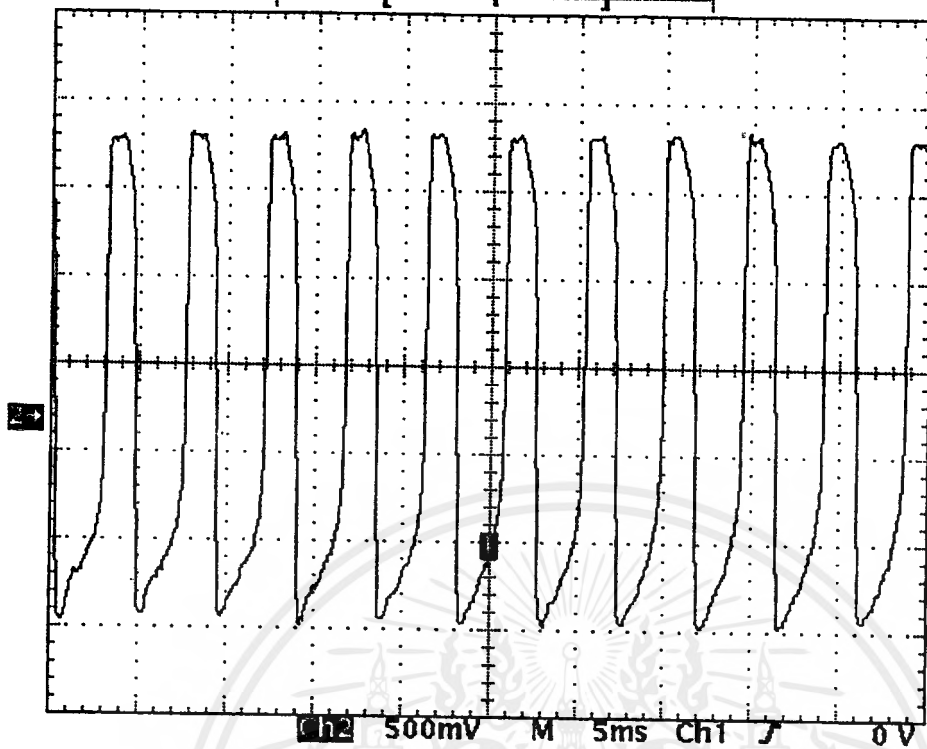


รูปที่ 4.7.14 แสดงรูปสัญญาณที่ Input1 โดยป้อนสัญญาณจากไมโครโฟน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tek Stop 10kS/s

10 Acqs



Ch2 Pk-Pk
2.82 V

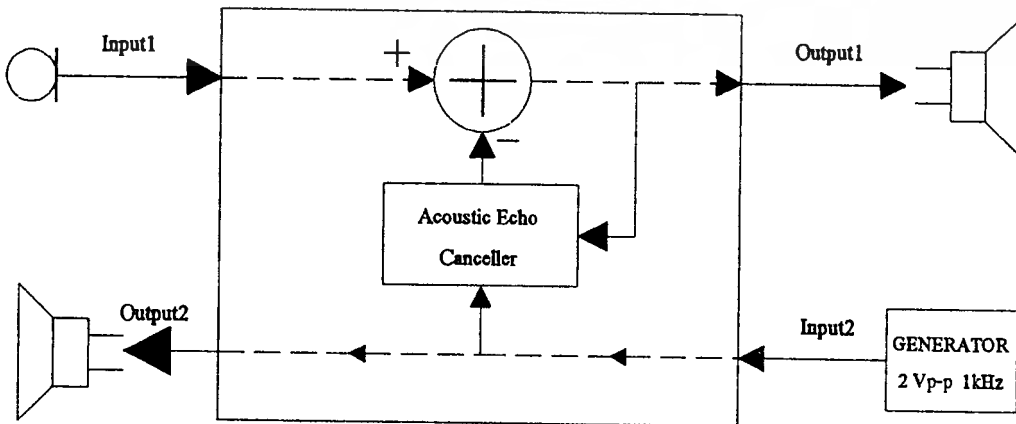
17 Apr 1999
02:55:18

รูปที่ 4.7.1.5 แสดงรูปสัญญาณที่ Output1 โดยป้อนสัญญาณจากไมโครโฟน

4.7.2 ผลการทดลองการกำจัดเสียงสะท้อน(เสียงหอน) ระหว่างไมโครโฟนกับลำโพง



(a) เครื่องกำจัดเสียงก้องที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้อง



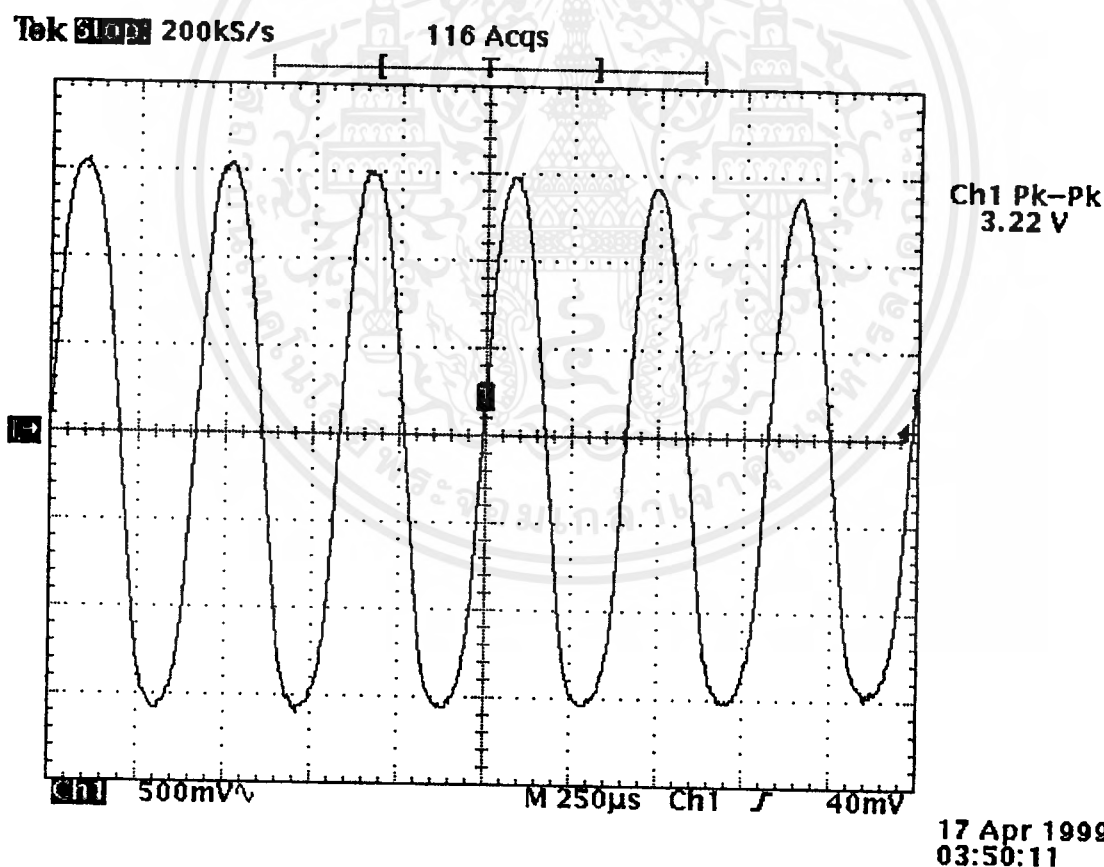
(b) เครื่องกำจัดเสียงก้องที่มีอัลกอริทึมในการกำจัดเสียงก้อง

รูปที่ 4.7.2.1 บล็อกโคโอดแกรมแสดงการต่อทดลองการกำจัดเสียงสะท้อน ระหว่างไมโครโฟนกับลำโพง

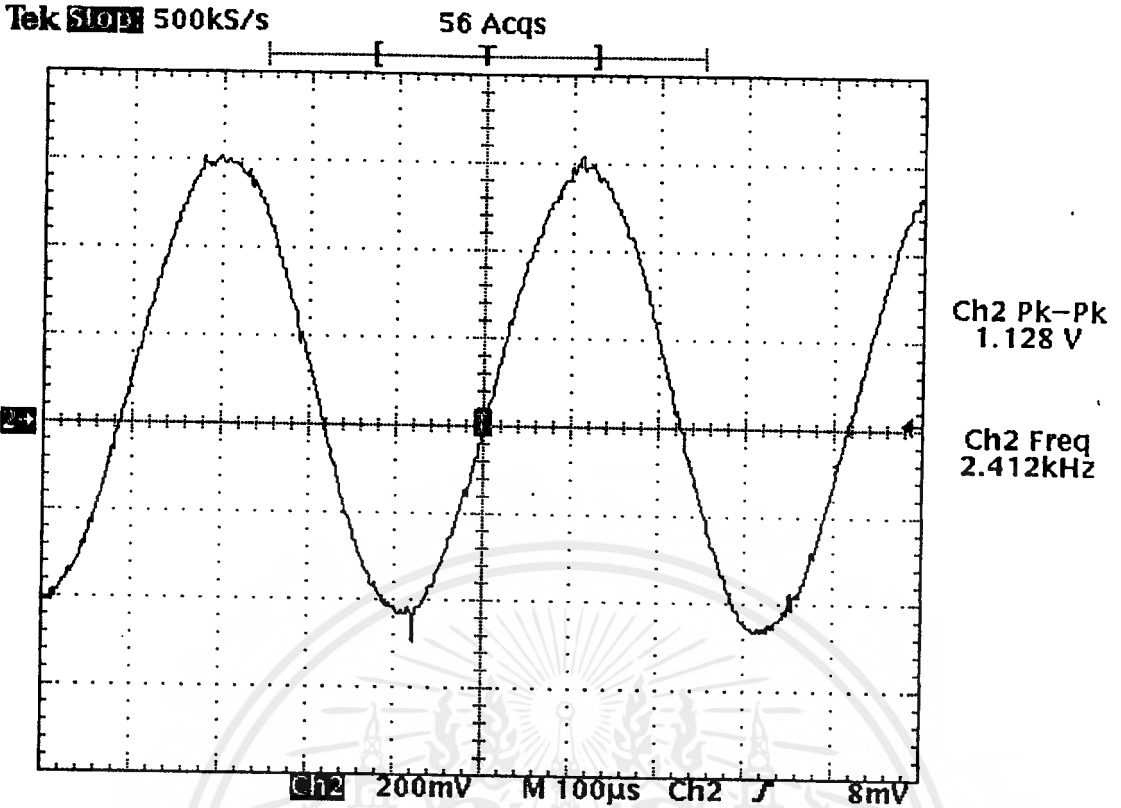
เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ผู้ใช้ต้องรับผิดชอบต่อการใช้งานการคัดลอกหรือการนำข้อมูลไปใช้โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทดลอง

1. จ่ายไฟให้กับเครื่องกำเนิดเสียงก้อง
2. ป้อนไมโครโฟนเข้าที่ Input1 และต่อลำโพงที่ Output1 และป้อนสัญญาณ Sinewave 2 Vp-p 1 kHz เข้าที่ Input2 ดังรูปที่ 4.7.2.1 (a)
3. ข้อมไมโครโฟนที่ Input1 เข้าไปหาลำโพง Output1
4. ทำการรันโปรแกรมที่มีเพียงการรับส่งข้อมูลแต่ไม่มีอัลกอริทึมในการกำจัดเสียงสะท้อนดังรูปที่ 4.7.2.1(a) แล้วทำการวัดสัญญาณที่ตำแหน่ง Output1 บันทึกผลการทดลอง
5. เปลี่ยนมาทำการรันโปรแกรมที่มีอัลกอริทึมในการกำจัดเสียงสะท้อนดังรูปที่ 4.7.2.1(b) แล้วทำการวัดสัญญาณที่ตำแหน่ง Output1 บันทึกผลการทดลอง

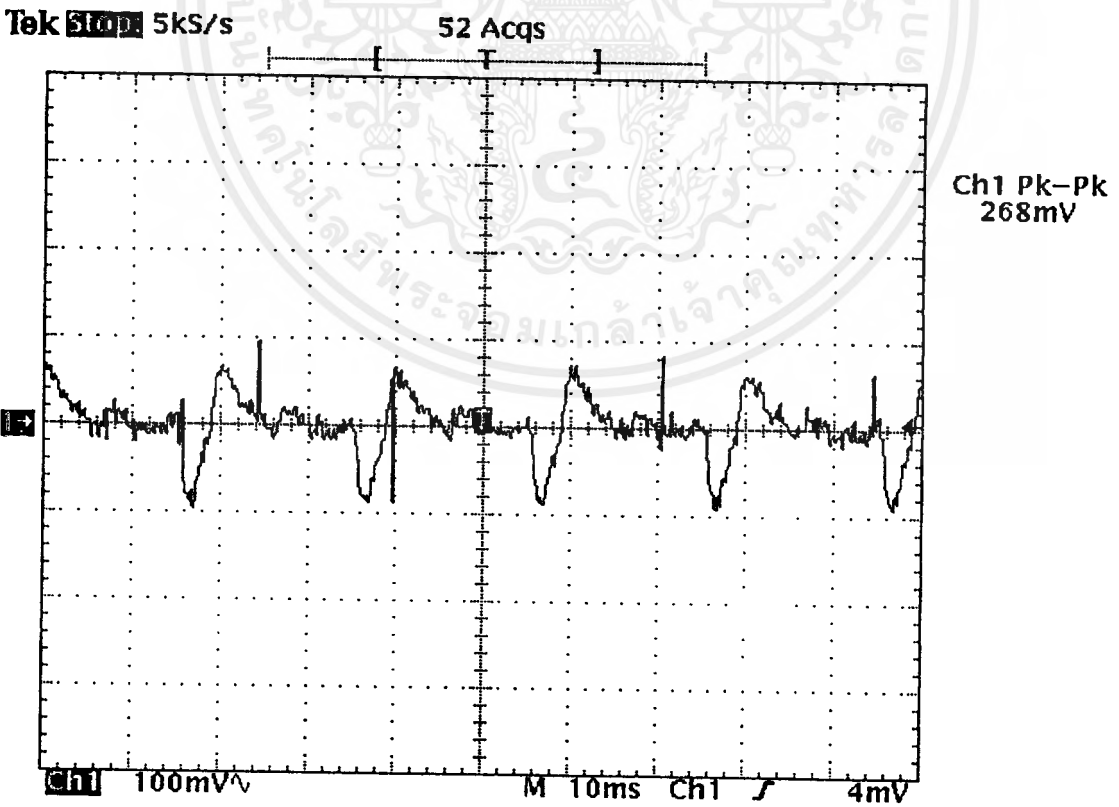


รูปที่ 4.7.2.2 แสดงรูปสัญญาณที่ตำแหน่ง Input1 เมื่อทำการทดลอง ตามรูปที่ 4.7.2.1(a)



17 Apr 1999 03:40:00

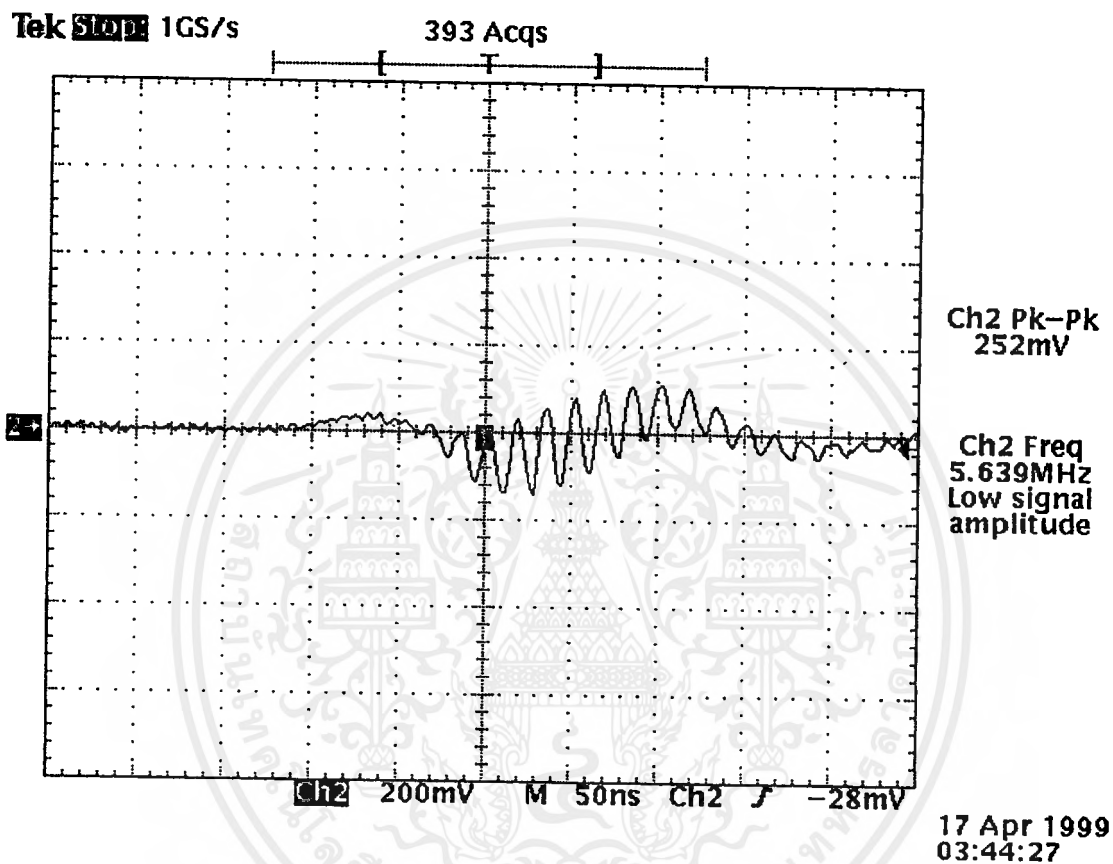
รูปที่ 4.7.2.3 แสดงรูปสัญญาณที่ตำแหน่ง Output1 เมื่อทำการทดลอง ตามรูปที่ 4.7.2.1(a)



17 Apr 1999 03:53:02

รูปที่ 4.7.2.4 แสดงรูปสัญญาณที่ตำแหน่ง Input1 เมื่อทำการทดลอง ตามรูปที่ 4.7.2.1(b)

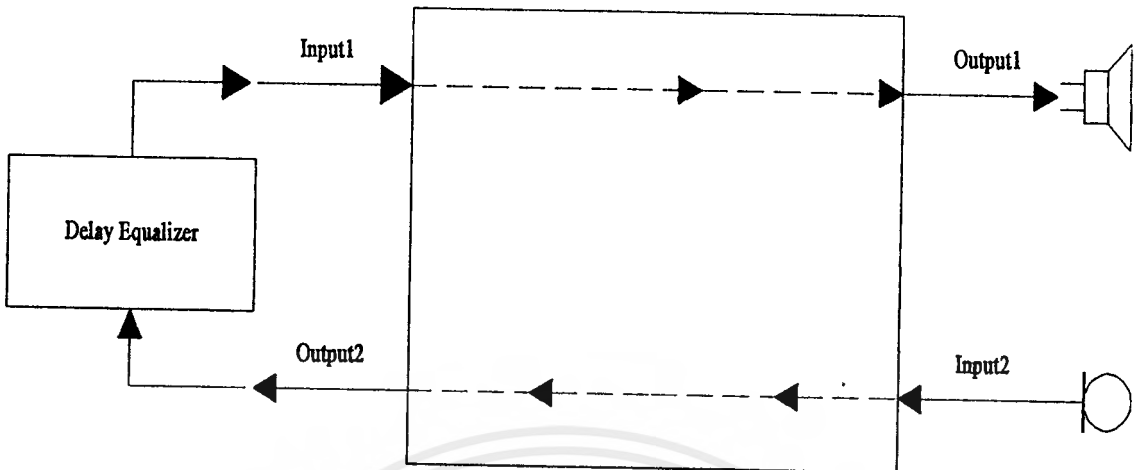
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



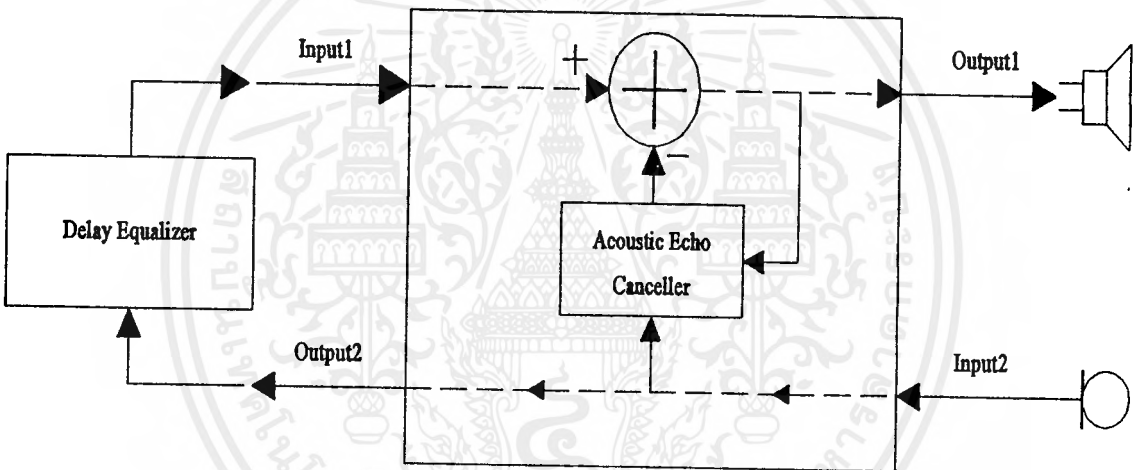
รูปที่ 4.7.2.5 แสดงรูปสัญญาณที่ตำแหน่ง Output1 เมื่อทำการทดลอง ตามรูปที่ 4.7.2.1(b)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7.3 ผลการทดลองโดยใช้ชุด Delay Equalizer



(a) เครื่องกำจัดเสียงก้องที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้อง



(b) เครื่องกำจัดเสียงก้องที่มีอัลกอริทึมในการกำจัดเสียงก้อง

รูปที่ 4.7.3.1 บล็อก โคอะแกรมแสดงการทดลองการกำจัดเสียงสะท้อน โดยใช้ชุดทดลอง Delay Equalizer

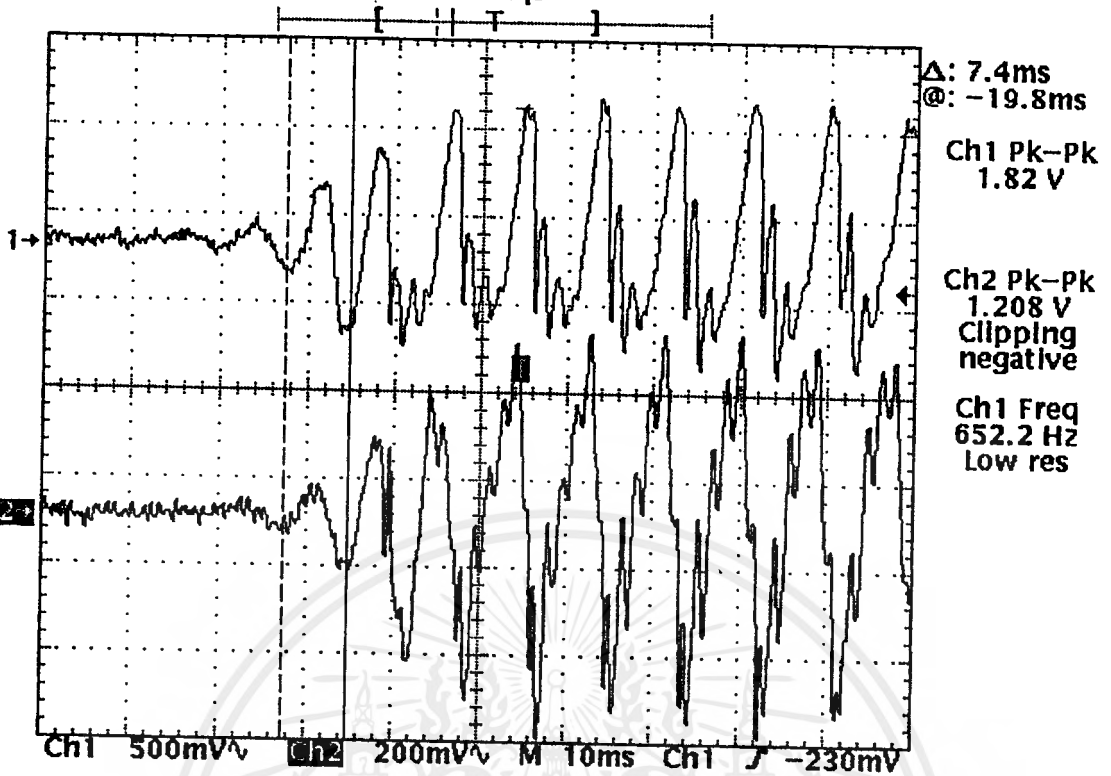
ขั้นตอนการทดลอง

1. ต่อชุดทดลอง Delay Equalizer เข้ากับ Input1 และ Output 2 ของเครื่องกำจัดเสียงก้อง(Acoustic Echo Canceller) ต่อโมโครโฟนเข้ากับ Input2 ของเครื่อง และต่อลำโพงเข้ากับ Output1ของเครื่องกำจัดเสียงก้อง ตามที่ได้แสดงผังรูปที่ 4.7.3.1
2. จ่ายไฟให้กับเครื่องกำจัดเสียงก้อง
3. ทำการรัน โปรแกรมที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้องดัง ในรูปที่ 4.7.3.1(a) แล้วทำการวัดสัญญาณที่ ตำแหน่ง Input2 , Output2 , Input1, และ Output1 บันทึกผลที่ได้จากการทดลอง
4. เปลี่ยนมาทำการรัน โปรแกรมที่มีอัลกอริทึมในการกำจัดเสียงก้องดังแสดงในรูปที่ 4.7.3.1(b) แล้วทำการวัดสัญญาณที่ตำแหน่ง Output1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tek **STOP** 5kS/s

3 Acqs

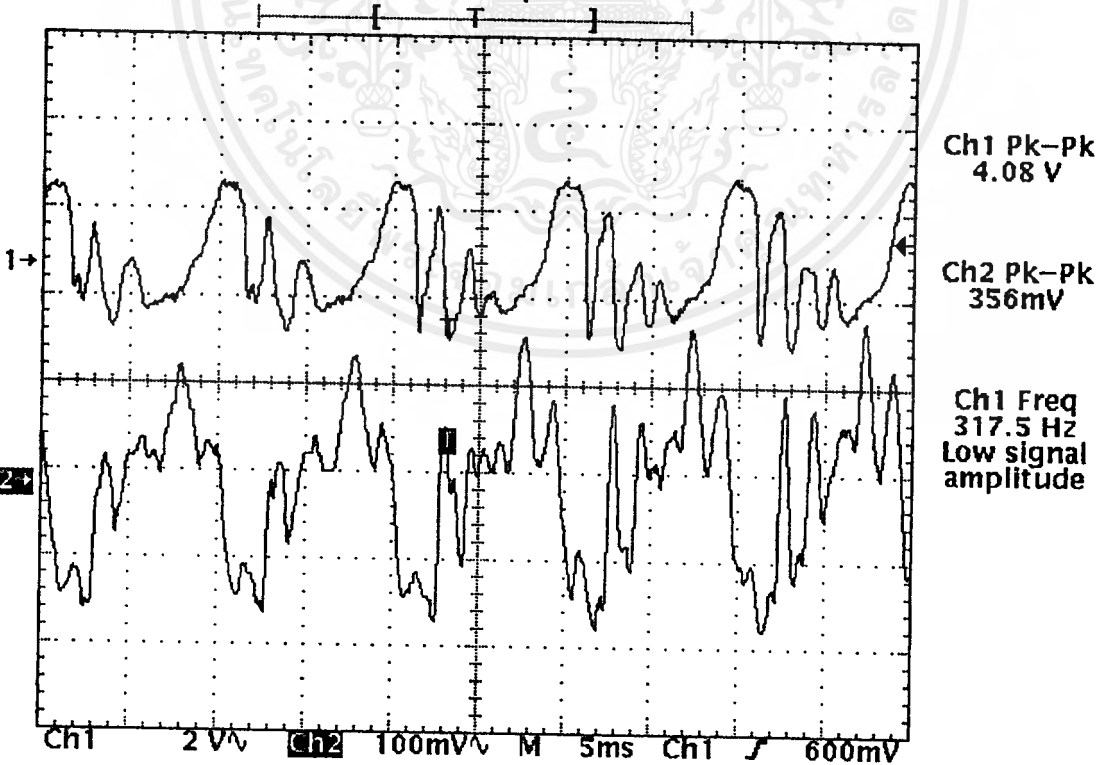


22 Apr 1999 04:52:42

รูปที่ 4.7.3.2 แสดงการทดลองตามรูปที่ 4.7.3.1(a) โดยวัดที่ตำแหน่ง Output2 (Ch1) กับ Input1(Ch2) เพื่อตรวจสอบการ delay สัญญาณ ของชุด Delay Equalizer

Tek **STOP** 10kS/s

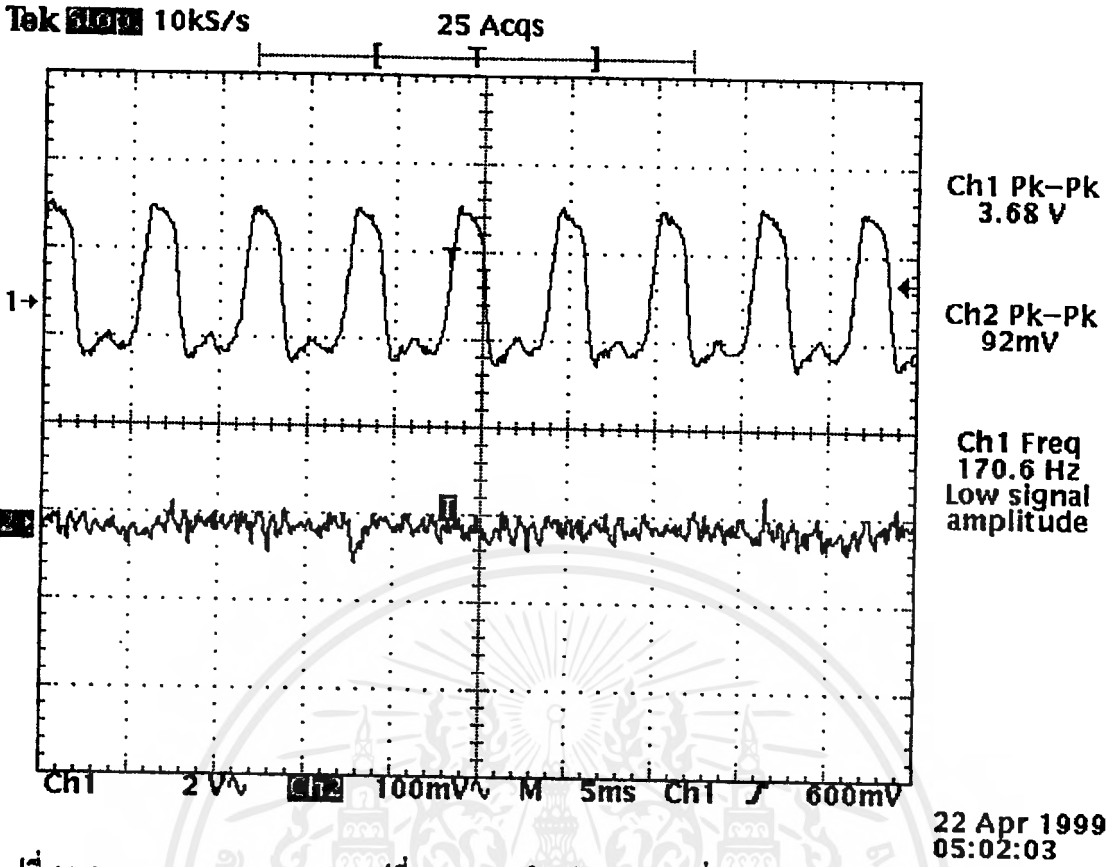
10 Acqs



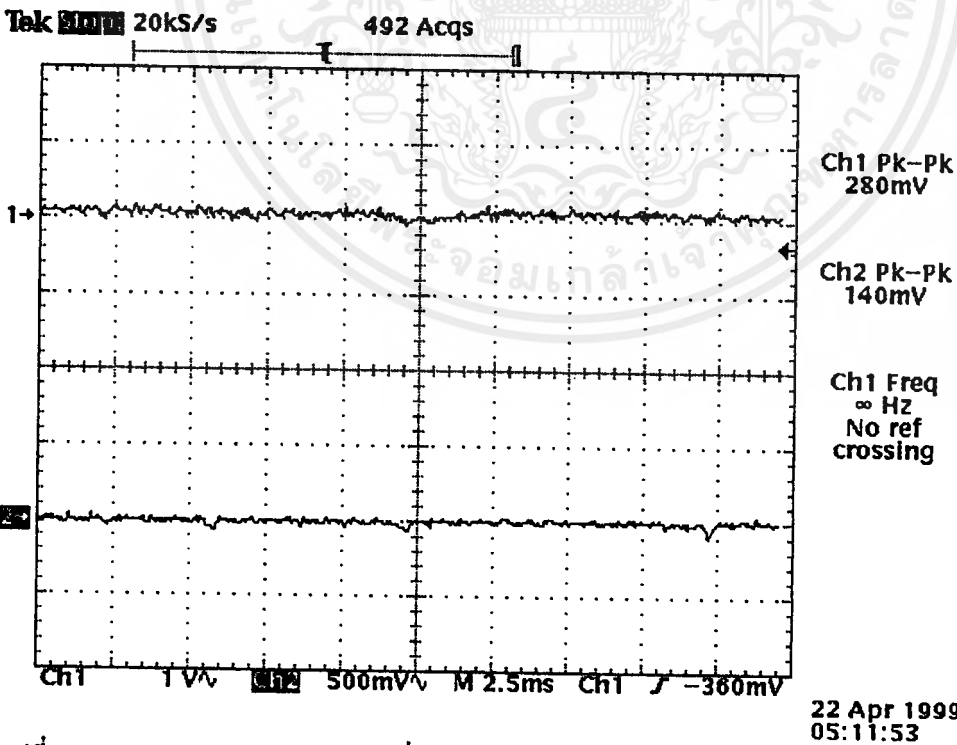
22 Apr 1999 04:57:50

รูปที่ 4.7.3.3 แสดงการทดลองตามรูปที่ 4.7.3.1(a) โดยวัดที่ Input2(Ch1)กับ Output1(ch2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

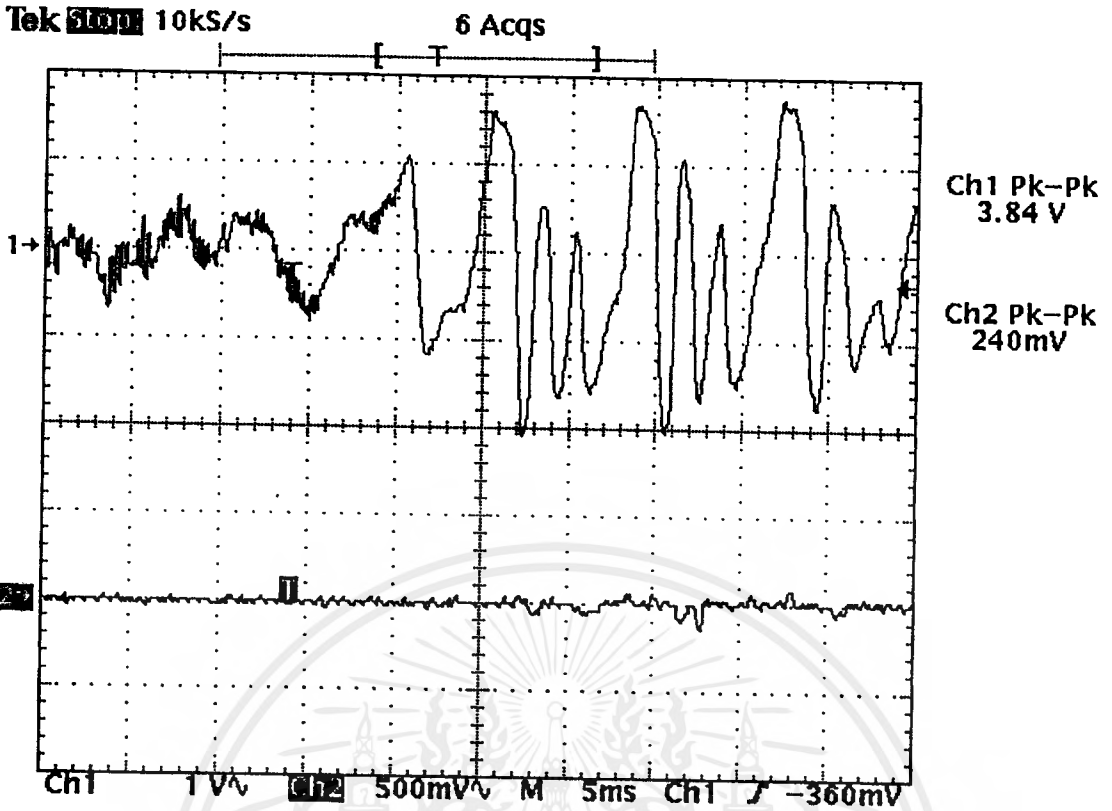


รูปที่ 4.7.3.4 แสดงการทดลองตามรูปที่ 4.7.3.1(b) โดยวัดสัญญาณที่ Input2(Ch1) กับ Output(Ch2) เพื่อแสดงผลการกำจัดเสียงของการ delay ที่เกิดขึ้นจากชุดทดลอง Delay Equalizer



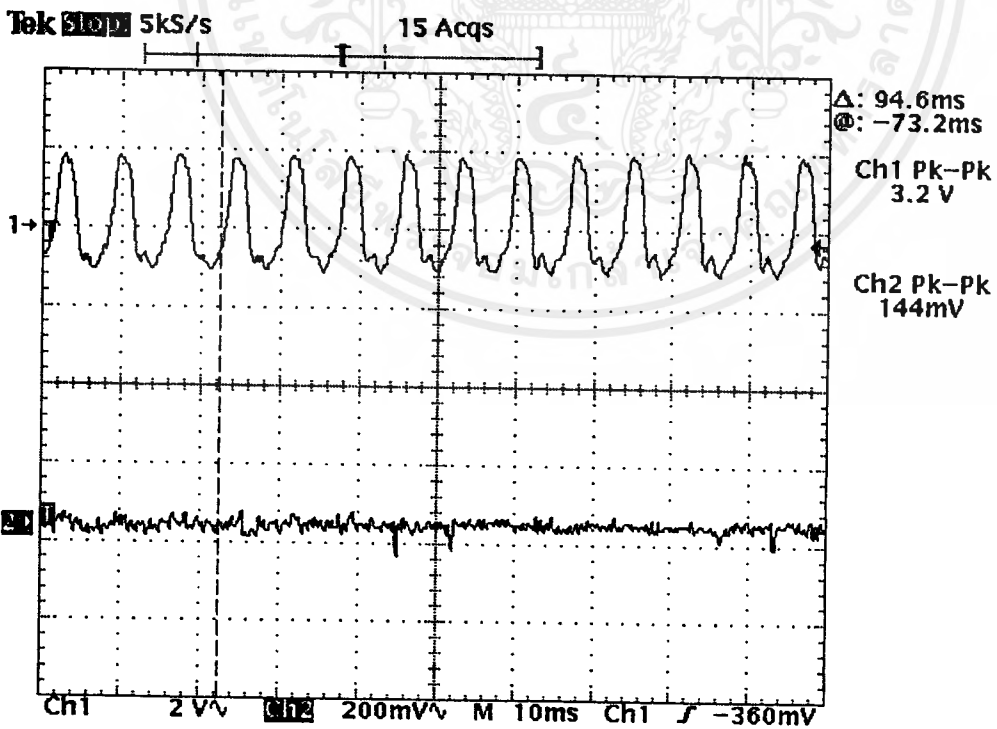
รูปที่ 4.7.3.5 แสดงการทดลองตามรูปที่ 4.7.3.1(b) โดยวัดสัญญาณที่ Input2(Ch1) กับ Output(Ch2) ในช่วงที่ไม่มีเสียงพูดจากไมโครโฟน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



22 Apr 1999 05:17:04

รูปที่ 4.7.3.6 แสดงการทดลองตามรูปที่ 4.7.3.1(b) โดยวัดสัญญาณที่ Input2(Ch1) กับ Output(Ch2) ในช่วงที่เริ่มดำเนินการทดสอบจากไมโคร โฟน

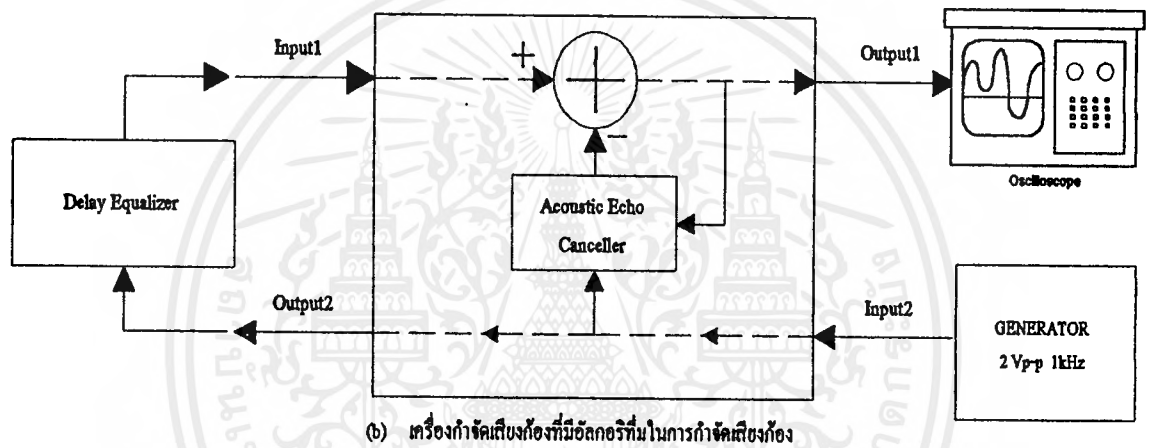
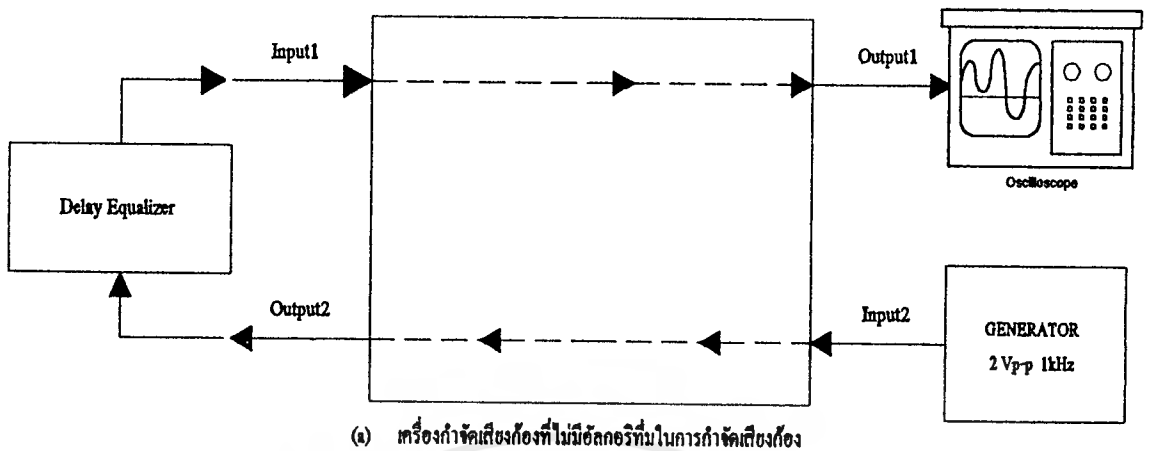


22 Apr 1999 05:41:05

รูปที่ 4.7.3.7 แสดงการทดลองตามรูปที่ 4.7.3.1(b) โดยวัดสัญญาณที่ Input2(Ch1) กับ Output(Ch2) ในช่วงที่มีการทดสอบจากไมโคร โฟนอย่างต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7.4 ผลการทดลองโดยใช้ชุด Delay Equalizer ที่มีการป้อนอินพุตเป็นสัญญาณ Sinewave

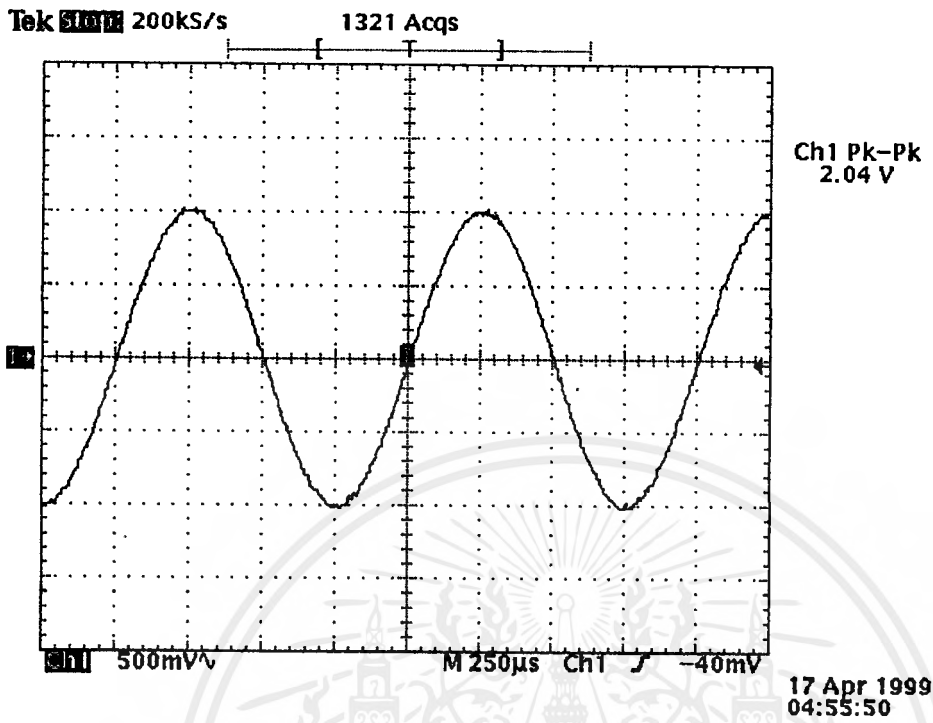


รูปที่ 4.7.4.1 แสดงการต่อวงจร ในการทดลองชุด Delay Equalizer ที่มีการป้อนอินพุตเป็นสัญญาณ Sinewave

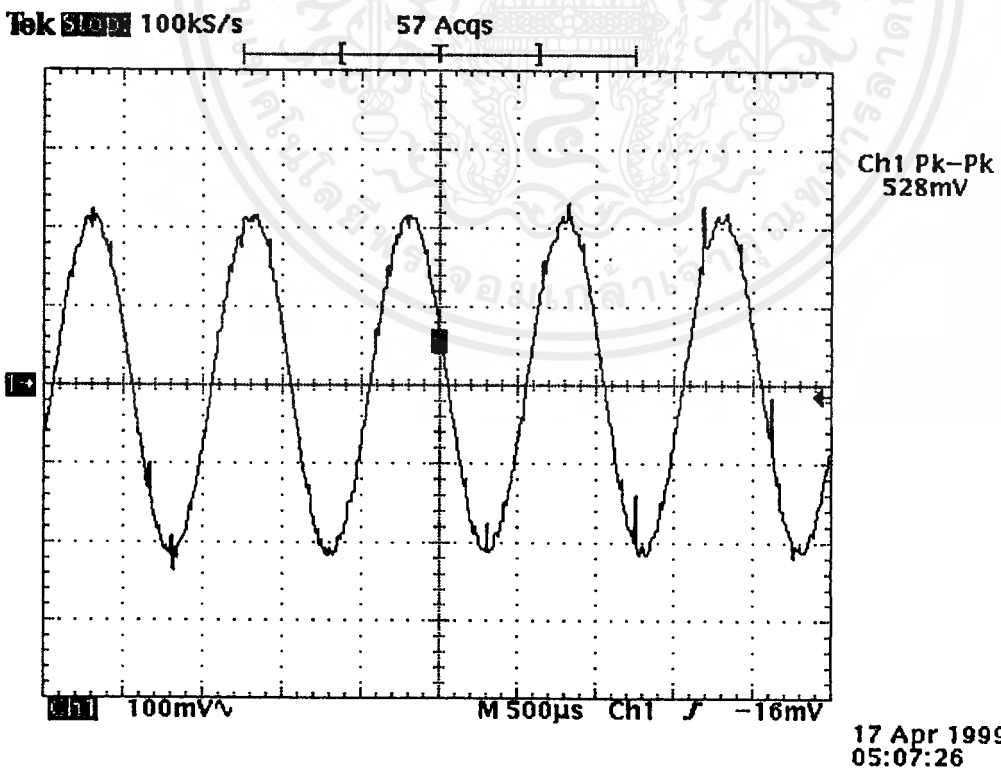
ขั้นตอนการทดลอง

1. ต่อชุดทดลอง Delay Equalizer เข้ากับ Input1 และ Output 2 ของเครื่องกำจัดเสียงก้อง(Acoustic Echo Canceller) ต่อ เครื่องกำเนิดสัญญาณซาวน์เข้ากับ Input2 ของเครื่องกำจัดเสียงก้อง และต่อออสซิลโลสโคปเข้ากับ Output1ของเครื่องกำจัดเสียงก้อง ตามที่ได้แสดงดังรูปที่ 4.7.4.1
2. จ่ายไฟให้กับเครื่องกำจัดเสียงก้อง
3. ทำการรัน โปรแกรมที่ไม่มีอัลกอริทึมในการกำจัดเสียงก้องดังในรูปที่ 4.7.4.1(a) แล้วทำการวัดสัญญาณที่ ตำแหน่ง Input2 , Output2 , Input1, และ Output1 บันทึกผลที่ได้จากการทดลอง
4. เปลี่ยนมาทำการรัน โปรแกรมที่มีอัลกอริทึมในการกำจัดเสียงก้องดังแสดงในรูปที่ 4.7.4.1(b) แล้วทำการวัดสัญญาณที่ตำแหน่ง Output1

ผลการทดลองวงจรในรูปที่ 4.7.4.1(a)



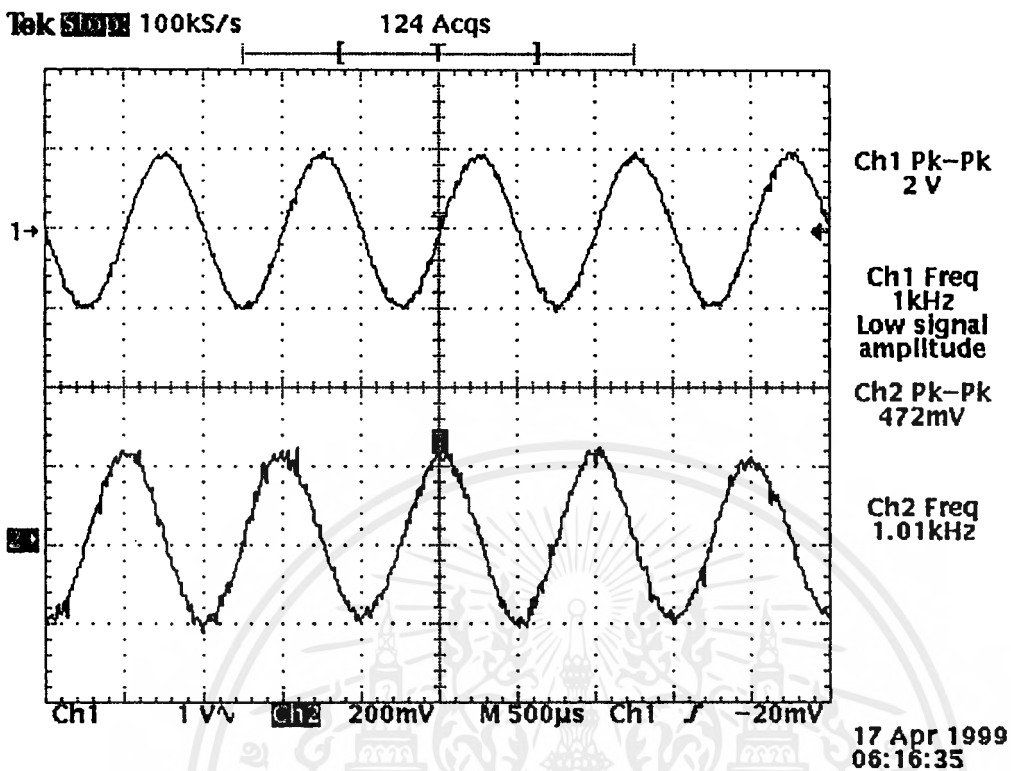
รูปที่ 4.7.4.2 แสดงรูปสัญญาณที่ตำแหน่ง Input2 ซึ่งเป็นสัญญาณขาอินพุตที่ป้อนเป็นสัญญาณอินพุต



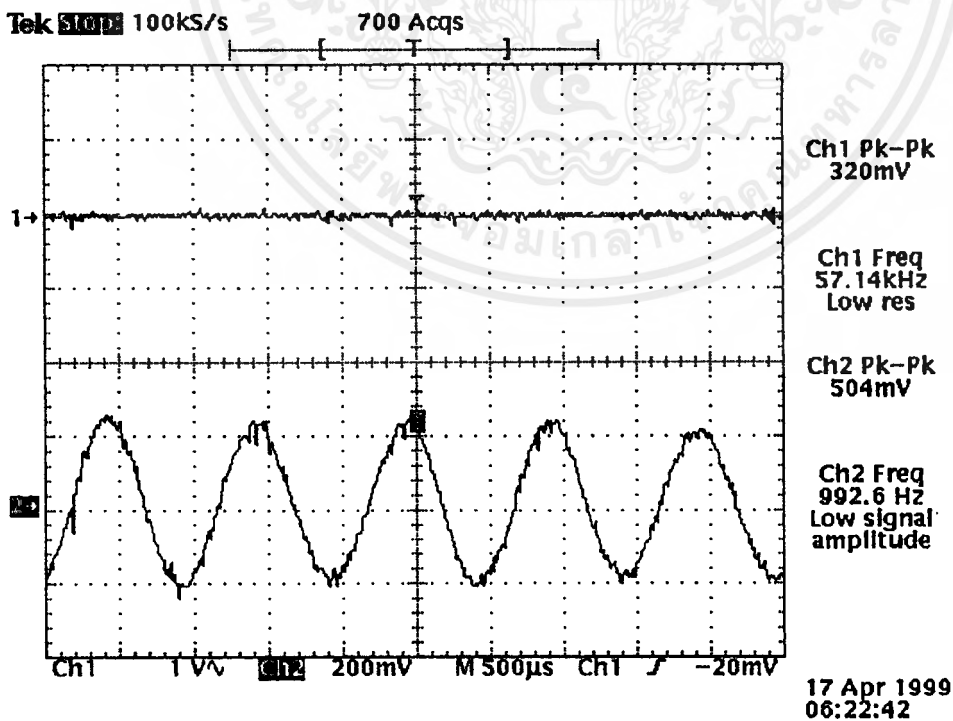
รูปที่ 4.7.4.3 แสดงรูปสัญญาณที่ตำแหน่ง Output1 ซึ่งเป็นสัญญาณเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลองวงจรในรูปที่ 4.7.4.1(b)



รูปที่ 4.7.4.4 แสดงรูปสัญญาณที่ตำแหน่ง Output2(Ch1) กับ Input1(Ch2)
ซึ่งเป็นสัญญาณ อินพุตและเอาต์พุต ของ ชุด Delay Equalizer



รูปที่ 4.7.4.5 แสดงรูปสัญญาณที่ตำแหน่ง Output1(Ch1) กับ Input1(Ch2)
ซึ่งสัญญาณ Output1(Ch1) เป็นสัญญาณ เอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทวิจารณ์ และ บทสรุป

เนื่องด้วยในทอมคันนั้นทางกลุ่มโปรเจ็คมีเจตจำนงที่จะนำเอาอัลกอริทึมที่มีประโยชน์ในการวิเคราะห์เพื่อจำลองรูปสัญญาณเพื่อเอามาใช้หักล้างกับสัญญาณเสียงสะท้อนโดยการทำให้ค่า Weight ของตัว Adaptive filter มีลักษณะเป็นไปตามค่าของเส้นทางเดินของเสียงสะท้อน(Echo path) ซึ่งจะมีการใช้งานอยู่ 2 ระบบ หรือ 2 อัลกอริทึมเป็นประเภทใหญ่ๆ คือ การวิเคราะห์ทางด้านโดเมนที่เกี่ยวกับเวลา time domain algorithms ซึ่งเส้นทางเดิน ของเสียงมีผลต่อการตอบสนองในอัตราที่มากแล้วจะทำให้การคำนวณในการหาค่าความสัมพันธ์ระหว่างสัญญาณนั้นเป็นไปในจำนวนอัตราที่สูงด้วยซึ่งจะต้องอาศัยการห้วงเวลาเพื่อที่จะอาศัยเอาค่าของสัญญาณที่ใช้อ้างอิงมาทำการปรับน้ำหนัก ซึ่งจำนวนของการห้วงเวลานั้นถ้ามีค่ามากแล้วจะมีผลต่อการคำนวณซึ่งใช้เวลาในการคำนวณที่มากด้วย ซึ่งเป็นข้อเสียของการใช้อัลกอริทึมทางโดเมนทางเวลา แต่ก็มีข้อดีคือถ้าจำนวนการห้วงเวลาหรือจำนวน tap มีค่าไม่มากแล้วและก็จะทำให้การคำนวณก็จะมีประสิทธิภาพที่ดี และด้วยโครงสร้างทางอัลกอริทึมที่ง่ายจึงเป็นข้อที่ได้เปรียบด้วย ในงานที่ทำโปรเจ็คนี้เราได้ใช้บอร์ดทดลอง DSK START KID ซึ่งจะมีประสิทธิภาพในการทำงานทางด้านเวลาได้เป็นอย่างดี ซึ่งทำให้เราสามารถนำมาใช้งานในโปรเจ็ค แต่ก็มีข้อเสียที่เรื่องจำนวน tap ของอัลกอริทึม ดังที่ได้กล่าวมาแล้ว และในส่วนของ Hardware นั้นเราได้สร้างชุดการติดต่อทางด้านอะนาล็อกขึ้นมาเอง เนื่องจากจะต้องมีการติดต่อกับชุด DSK ถึง 2 ชุด โดยประกอบไปด้วยส่วนที่จัดการกับสัญญาณที่เป็นอะนาล็อก ซึ่งจะต้องมีการลดความผิดเพี้ยนที่เกิดขึ้นกับส่วนนี้ด้วย

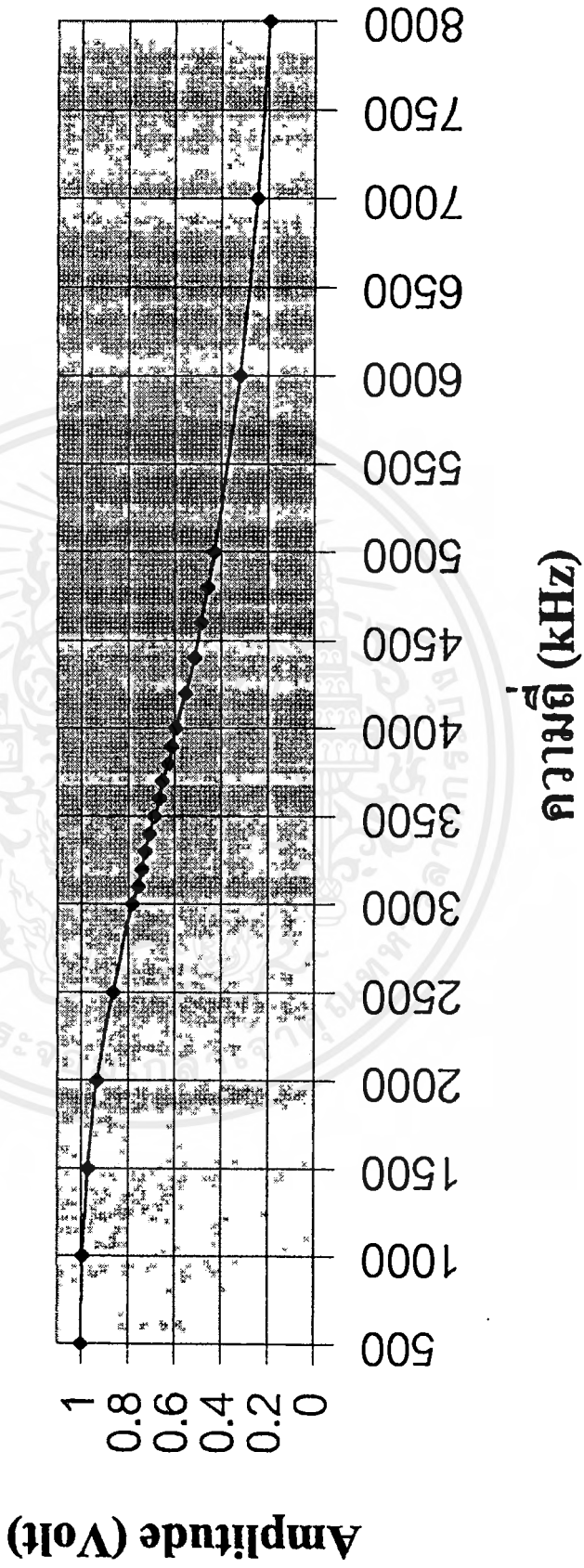
แนวทางในการปรับปรุงนั้นเนื่องจากหากต้องการที่จะใช้งานในการกำจัดผลการตอบสนองของห้องที่มีเส้นทางเดินของเสียงสะท้อนที่มากแล้วเราจะต้องมีการคำนวณในอัตราที่มากด้วย แต่ก็มีอัลกอริทึมทางด้านความถี่ที่สามารถที่จะลดการคำนวณโดยอาศัยการใช้ การแปลงฟูเรียร์อย่างรวดเร็ว และการแปลงกลับฟูเรียร์อย่างรวดเร็ว แต่ก็ต้องอาศัยการทำงานของหน่วยประมวลผลในแบบขนานซึ่งจะยากขึ้นไปอีก แต่ก็เหมาะกับระบบของ Room Acoustic Echo canceller ซึ่งมีเส้นทางของเสียงสะท้อนที่สูง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

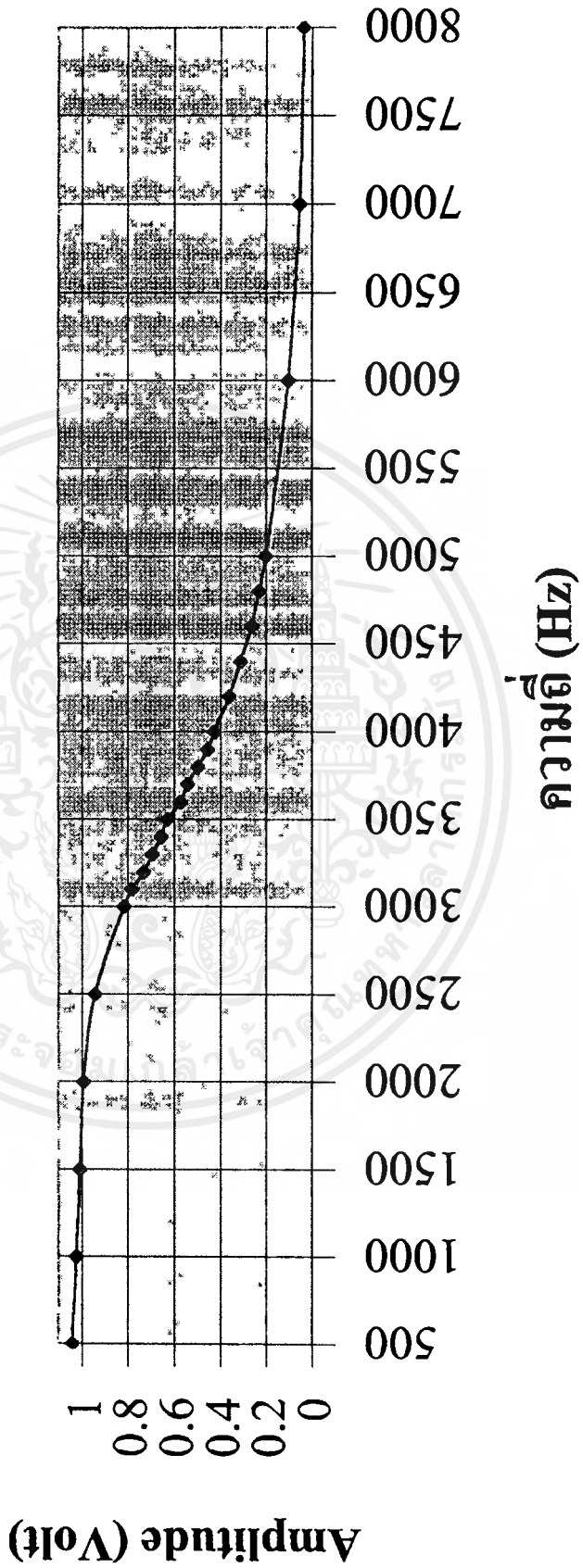
ผลการทดลองวงจรความถี่ต่ำ

Butterworth Order 2 $f_c = 3.4$ kHz



ผลการทดลองวงจรกรองความถี่ต่ำผ่าน

Butterworth Order 4 $f_c = 3.4$ kHz

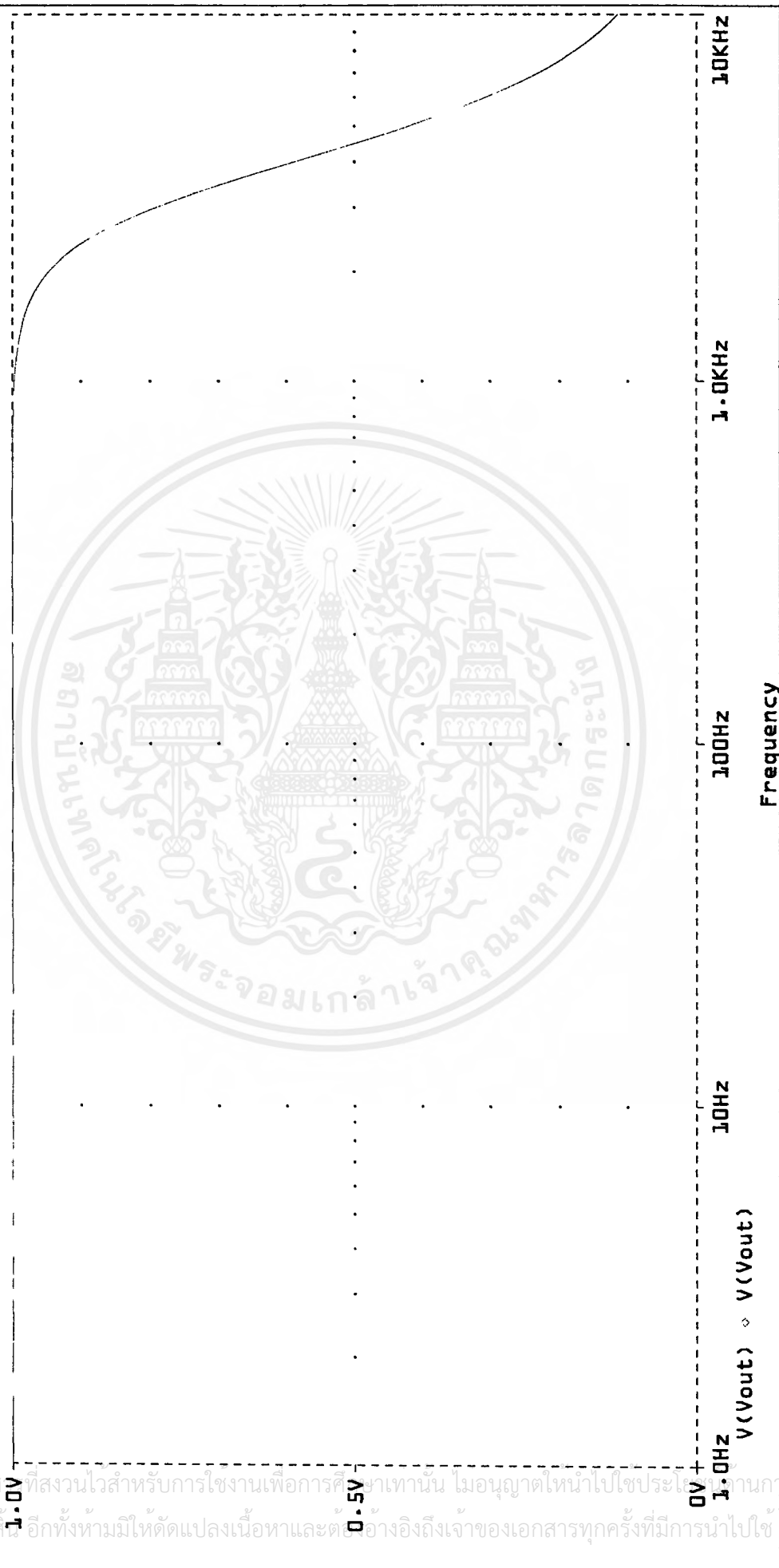


* C:\TEMP\disk\3.4K2.sch

Date/Time run: 04/23/99 06:20:14

Temperature: 27.0

(A) 3.4K2.dat



AA1:(3.4065K,707.826m) A2:(1.0000,1.0000) DIFF(A):(3.4055K,-292.135m)

Date: April 23, 1999 Butterworth Lowpass Filter order2 fc = 3.4 kHz

Time: 06:23:47

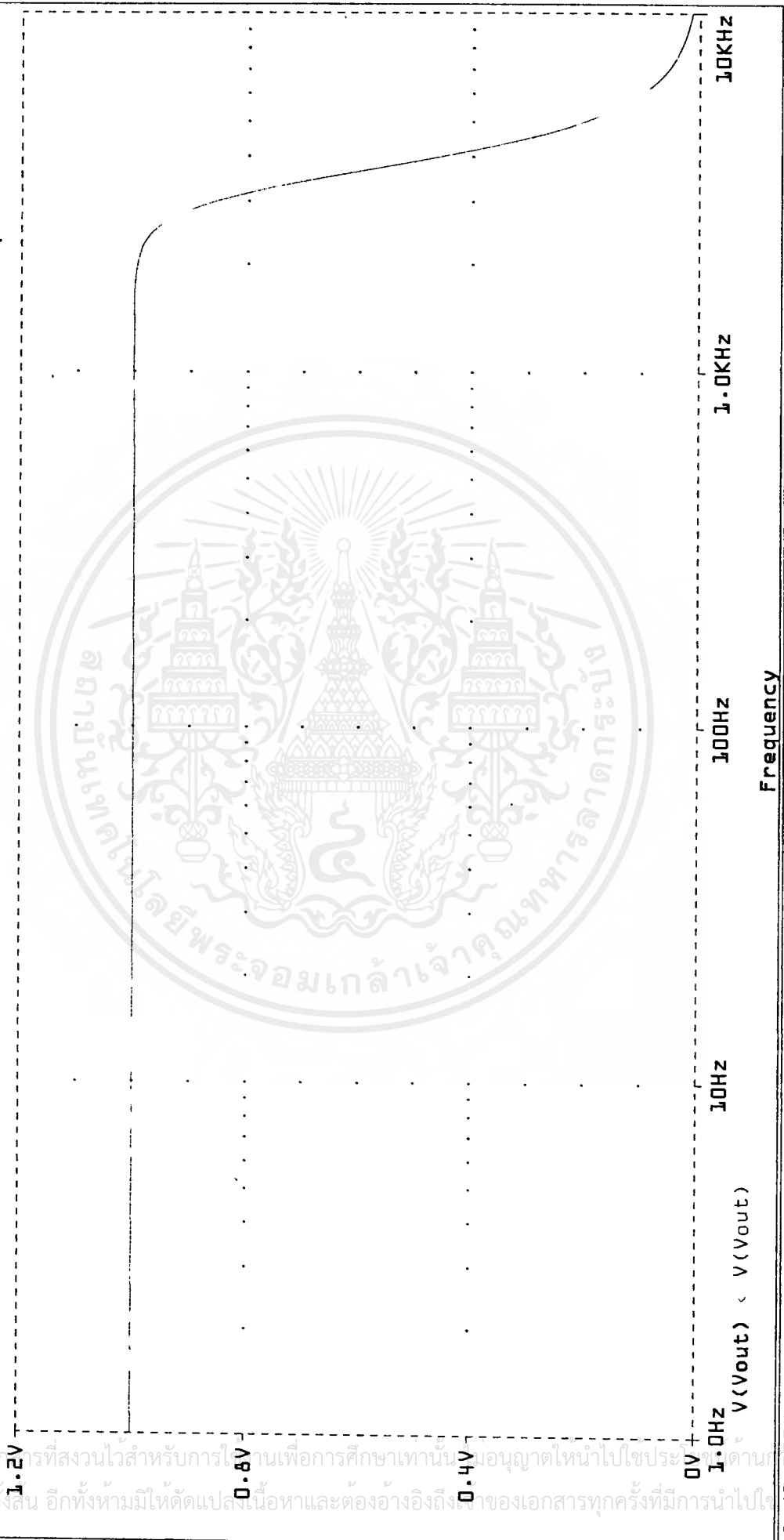
เอกสารนี้เป็นเอกสารต้นฉบับที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในทางอื่นโดยไม่ได้รับอนุญาต หากท่านมีเหตุเปลี่ยนแปลงเนื้อหาและต้นฉบับอื่นใด กรุณาแจ้งถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

* C:\TEMP\disk\3.4K.sch

Date/Time run: 04/23/99 05:53:35

Temperature: 27.0

(A) 3.4K.dat



AA1:(3.3927K,707.366m) A2:(1.0000,0.9999) DIFF(A):(3.3917K,-292.553m)

Date: April 23, 1999

Butterworth Lowpass Filter order4 fc = 3.4 kHz

Time: 06:16:07

โปรแกรมหลัก Acoustic Echo Canceller ที่ใช้กับ DSP TMS320C50

```

*****
***   ACOUSTIC ECHO CANCELLER 128 Tap   *****
*****

        .mmregs
*****
*       ALGORITHM CONSTANTS           *
*****
THRES0   .set   0800h           ;* SUPRESS THRES = 1/32 (-30DB)
*         *       = 2**(-THRES0 (-6DB * THRES0)
HANGT0   .set   600             ;* HANG OVER TIME = 75 MSECS
*         *       = 125 USECS * HANGT0
NER       .set   1              ;* NEAR END SPEECH THRES = -6DB
*         *       = -6DB * NER
CUTOF0   .set   0102h          ;* UPDATE CUTOFF = -60DB
*         *       = -6DB * (13 - LN(CUTOF0))
;-----;
*****
***** PAGE 0           *****
*****
IN_SAMP   .set   0x75           ;* STORAGE FOR Sn and Yn
OUTPUT   .set   0x76           ;* STORAGE FOR En
*****
***** DP 20h (1000h) *****
*****
Y         .set   0x1            ;Input reference non convert
ONE       .set   0x4            ;Contant one. (DP = 20).
En        .set   0x5            ;Signal error. (DP = 20).
ERRF      .set   0x6            ;Error function. (DP = 20).
Rn        .set   0x7            ;Filter output. (DP = 20).
Sn        .set   0x8            ;* NEAR-END SAMPLE MSBS
H         .set   0x9
THRES     .set   0x0A           ;* RESIDUAL OUTPUT SPRS THRESHOLD
HANGT     .set   0x0B           ;* HANG OVER COUNTER RESET VALUE
HCNTR     .set   0x0C           ;* HANG OVER COUNTER
CUTOFF    .set   0x0D           ;* ABSY CUTOFF LVL FOR NO UPDATE
POWS      .set   0x0E           ;* Power Sn=Sn^2
LPOW      .set   0x0F           ;* LONG input power
MU        .set   0x10           ;* 1/INPOW
INPOW     .set   0x11           ; Short input power
M0        .set   0x12           ;* LOCAL MAXIMA (K=0)
M1        .set   0x13
M2        .set   0x14
M3        .set   0x15
M4        .set   0x16
M5        .set   0x17
M6        .set   0x18
M7        .set   0x19
M8        .set   0x1A           ;* LOCAL MAXIMA (K=8)
ADM7      .set   0x1019
OUT_En    .set   0x1B
ABSYn     .set   0x1C
ABSSn     .set   0x1D
CUN0      .set   0x1E
SONE      .set   0x1F

```

```

*****
***** DP 22h (1100h) *****
*****
DY .set 0x1100
FIRST_Y .set 0x1100 ;Next newest data sample.
LAST_Y .set 0x117F ;Oldest data sample.
Yn .set 0x0 ;* REFERENCE SAMPLE (K=0)
Y1 .set 0x1
Y16 .set 0x10
Y127 .set 0x7F ;* REFERENCE SAMPLE (K=127)
*****
***** DP 23h (1180h) *****
*****
Y128 .set 0x0
;-----;
COEFFP .set 02000h ;Program memory address of the coeff. in S/A RAM
COEFFD .set 02000h ;Data memory address of the coeff. in S/A RAM
LAST_C .set 0207Fh
ADDRS .set 0E00h ;Program memory address of the conver
*****
* INTERRUPT BRANCHES *
*****
.ps 0x800
B BEGIN ;* ON HARDWARE RESET GO TO INIT
.ps 0x804
INT2: B COMM
.ps 0x808
TINT B TIMER ;* ON TINT GO TO TIMER
RINT: B RXRT ;* ON RINT GO TO RXRT
XINT: B TXRT ;* ON TINT GO TO TXRT
*****
* PROCESSOR INITIALIZATION ROUTINE *
*****
.ps 0x0a00
.entry
BEGIN call INIT ;DSP Initialization
call TIM_INIT ;TIMER Initialization
call Enab_INT ;ENABLE Interrupts
*****
* CYCLE END ROUTINE *
*****
LOOP ldp #0
idle
nop
b START
nop
b LOOP
*****
* Initial DSP Processor and Memory *
*****
INIT ldp #0 ;* INITIALIZE ST0 AND ST1
setc INTM
splk #0830h,PMST ;9K on-chip RAM as Data, No ROM
lacl #0 ; Set Wait State Control Register
samm PDWSR ; for 0 waits in pgm & data memory
lacl #04h ;for 7 waits in I/O
samm CWSR ;for port 0 - port 7
lacl #0Fh ;set waits state for Port 0/1
samm IOWSR ;for waits state for I/O
setc SXM
setc OVM

```

```

    spm    0
    clrc   CNF
;-----;
    mar    *,ar1
    lar    AR1,#1000h  ;* LOWEST PAGE 0 LOCATION -> AR1
    lacl   #0          ;* 0 -> ACC
    rpt    #5ffh
    sacl   *+          ;* ZERO PAGE 0
    lar    AR1,#2000h
    rpt    #0ffh
    sacl   *+          ;* ZERO PAGE 0
*
    ldp    #0
    lacc   #0h
    sacl   IN_SAMP
    sacl   OUTPUT
*
    ldp    #20h
    lacl   #1
    sacl   ONE
*
    lacc   #HANGT0
    sacl   HANGT
*
    lacc   #0400h      ;* >400 = 1/8 OF MAX ABSY
    sacl   INPOW
    sacl   M0
    sacl   LPOW
*
    lacc   #020h
    sacl   Mu
*
    lacc   #THRES0
    sacl   THRES
*
    lacc   #CUTOF0
    sacl   CUTOFF
*
    lacc   #4FFFh
    sacl   SONE
*
    ret
;-----;
*****
***   TIMER INTERRUPT INITIALIZATION   *
*****
TIM_INIT ldp    #0
        splk   #2499,PRD          ;Load timer period for 125 us period.
        splk   #0A,IMR           ;Set timer interrupt mask bit.
        splk   #20h,TCR          ;reload and start timer.
        splk   #10h,IFR          ;Clear any pending timer interrupts.
        ret
*****
***   ENABLE INTERRUPTS   ***
*****
Enab_INT ldp    #0
        splk   #0Ah,IMR          ;Enable TINT
        clrc   INTM             ;Enable Interrupt
        ret
;-----;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

*****
***** MAIN PROGRAM *****
*****
START ldp #0
*****
* NEAR END SAMPLE (Sn) *
*****
    lacl IN_SAMP
    and #0FFh
    sub #80h ;convert data
    ldp #20h
    sacl Sn
*****
* INPUT REFERENCE SAMPLE (Yn) *
*****
    ldp #0
    lacc IN_SAMP,8
    and #0FF00h,8
    ldp #20h
    sach Y
    lacl Y
    sub #80h ;convert data
    ldp #22h
    sacl Yn ;Y is converted
*****
* FIR ADAPTIVE FILTER *
*****
ADPFIR zpr ;Clear P register.
    lacc #1,14 ;Load output rounding bit.
    mar *,AR1
    lar AR1,#LAST_Y ;Point to oldest sasmple.
FIR rpt #127
    macd COEFFFP,*- ;128-tap FIR filter.
    apac
    ldp #20h
    sach Rn,1 ;Store the filter output.
    lacc Sn
    sub Rn ; Sn - Rn
    sacl En ;err(n) = Sn - Rn
*****
*
* RESIDUAL OUTPUT SUPPRESSION ROUTINE *
*
*****
SPRS lt Mu ;* Mu -> T REG
    mpy LPOW ;* LPOW * Mu -> P REG
;-----
;
    lacc HCNTR ;* NEAR END SPEECH FLAG -> ACC
    bcnd WOUT,GT ;* IF N.E. SPEECH NO SPRS
    pac
    sub THRES ;* ACC - THRES -> ACC
    bcnd WOUT,GT ;acc>0;* IF THRES EXCEEDED SKIP SPRS
    lacl #0h ;* 0 -> ACC
    sacl En ;* ACC -> OUTPUT
WOUT lt En ;* OUTPUT -> T REG (FOR UNO)

```

```

*****
*   TRANSMIT DATA TO MEMORY OUTPUT   *
*****
    lacl  En                ;* En -> ACC
    and   #0FFh
    add   #80h
    and   #0FFh
    sacl  OUT_En            ;ACC -> DXR
;-----;
;   OUTPUT Port 1           ;
;-----;

    zap
    lacc  Y,8
    and   #0FF00h
    or    OUT_En
    ldp   #0
    and   #0FFFFh
    sacl  OUTPUT
;-----;
*****
*   Find Long Input_Power (Y1 - Y128) *
*****
LONGPOW:
    zap
    ldp   #23h
    sqra  Y128
    ldp   #22h
    sqrs  Y1
    apac
    ldp   #20h
    add   LPOW
    sacl  LPOW
*****
*   Find Short Input_Power before speaker (Y1-Y16) *
*****
    zap                ;PREG = 0 , ACC=0
    ldp   #22h         ;DP=1880h
    sqra  Y16         ;PREG = x(k-N)^2
    sqrs  Y1          ;accu = - (x(k-N)^2)
*                ;PREG = x(k)^2
    apac              ;accu = x(k)^2 - x(k-N)^2
    ldp   #20h
    add   INPOW ;accu = INS(k-1) + x(k)^2 - x(k-N)^2
    add   CUTOFF   ;accu = INS(k-1) + x(k)^2 - x(k-N)^2+CUTOFF
    sacl  INPOW ;store Short input power
;-----;

    lacc  ONE,16
    rpt   #14
    subc  INPOW
    sacl  Mu                ;store 1/INPOW
*****
*   Find Input_Power after Microphone *
*****
    zap                ;PREG = 0
    sqra  Sn           ;PREG = d(n)^2
    spl   POWS         ;store d(n)^2 in POWS

```

```

*****
*   UPDATE MODULO 16 COUNTER (H)   *
*****
    lacc  H           ;* H -> ACC
    add   ONE         ;* ACC + 1 -> ACC
    and   #0fh        ;* IF ACC = 16 THEN 0 -> ACC
    sac1  H           ;* ACC -> H
    bcnd  NESP1,GT    ;* IF H > 0 THEN GO TO NESP1
*****
*   MOVE M0,M1,...,M7 TO NEXT HIGHER MEMORY LOCATION   *
*****
    lar   ar1,#ADM7   ;* ADM7 -> AR1
    rpt   #7          ;* K=7,6,...,0
    dmov  *-          ;* M(K) -> M(K+1)
    dmov  INPOW       ;* ABSYOF -> M0
    b     NESP3       ;* ON MEMORY MOVES SKIP DETECTION
*****
*   UPDATE MOST RECENT LOCAL MAXIMA (M0)   *
*****
NESP1 lacc  INPOW     ;* ABSYOF -> ACC
      sub   M0        ;* ACC - M0 -> ACC;
      bcnd  NESP2,LEQ ;acc<=0 IF M0 > INPOW THEN NO UPDATE
      dmov  INPOW     ;* INPOW -> M0
*****
*   COMPARE REFERENCE POWER TO NEAR-END POWER   *
*****
NESP2 lacc  POWS      ;* POWS -> ACC
      sub   M0        ;* ACC - M0 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M0 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M1        ;* ACC - M1 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M1 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M2        ;* ACC - M2 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M2 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M3        ;* ACC - M3 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M3 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M4        ;* ACC - M4 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M4 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M5        ;* ACC - M5 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M5 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M6        ;* ACC - M6 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M6 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M7        ;* ACC - M7 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M7 > ABSSOF
      lacc  POWS      ;* POWS -> ACC
      sub   M8        ;* ACC - M8 -> ACC
      bcnd  NESP3,LEQ ;acc<=0 NO N.E. SPEECH IF M8 > ABSSOF
*****
*   NEAR-END SPEECH DETECTED SET HANGOVER COUNTER (HCNTR)   *
*****
      dmov  HANGT     ;* HANGT -> HCNTR

```

```

*****
*   CHECK AND UPDATE HANGOVER COUNTER   *
*****
NESP3 lacc  HCNTR          ;* HCNTR -> ACC
      bcnd  NESP4,EQ        ;acc=0 IF HCNTR = 0 THEN GO TO NESP4
      sub   ONE            ;* ACC - 1 -> ACC
      sacl  HCNTR          ;* ACC -> HCNTR
      b     LOOP           ;* GO TO CYCLE END
*****
*   CHECK IF LTAU REFERENCE POWER ESTIMATE IS BELOW CUTOFF   *
*****
NESP4 lacc  INPOW          ;* INPOW -> ACC
      sub   CUTOFF         ;* ACC - CUTOFF -> ACC
      bcnd  LOOP,LEQ       ;acc<=0 IF INPOW < CUTOFF THEN LOOP
*****
*   LMS Adaption of Filter Coefficients.   *
*****
      mpy   Mu              ;P = beta*err(i)
      pac   #1,14           ;errf(i) = beta * err(i)
      sach  ERRF,1         ;Round the results.
      *
      lacc  #127
      samm  BRCR           ;128 coefficients to update
      *
      lar   AR2,#COEFFD    ;Point to the coefficients.
      lar   AR1,#Last_Y    ;Point to the data sample.
      lt    ERRF
      *
      rptb  end_block2
      mpy   *,,AR2         ;P = 2*beta*err(i)*x(i-255)
      pac
      bcnd  pl,geq
ne:    setc  SXM
      rpt   #17
      sfr
      bit   *,0 ;check W is +/-
      bcnd  nepl,ntc
nene:  add   *
      bcnd  end_add,lt
      lacc  #8000h,15
      rol
      b     end_add
nepl:  add   *
      b     end_add
pl:    clrc  SXM
      rpt   #17
      sfr
      bit   *,0 ;check W is +/-
      bcnd  plpl,ntc
plne:  add   *
      b     end_add
plpl:  add   *
      bcnd  end_add,geq
      lacc  #7fffh,15
      rol
      b     end_add
end_add: sacl  *,0,ar1
end_block2: NOP
      b     LOOP

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

*****
TIMER
    ldp    #0          ;* 0 -> PAGE POINTER
    out    OUTPUT,PA2 ;transmit data to D/A
    in     IN_SAMP,PA0 ;receive data Y,S from A/D
    RETE                   ;* INTERRUPTS ENABLED
                          ;* RETURN TO PROGRAM
*****
RXRT RETE ;* INTERRUPTS ENABLED
*****
TXRT RETE ;* INTERRUPTS ENABLED
*****
COMM: RETE ;* INTERRUPTS ENABLED
*****
    .end

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%*****
%
%                               Program design
%
%                               Telecommunication Engineering
%
%                               TIME DOMAIN ACOUSTIC ECHO CANCELLER Project 3T/2
%
%                               With adaptive for sinusoidal signal test at 1KHz
%
%                               Amplitude sinusoidal reference 1 Volt
%
%                               Amplitude sinusoidal replica 1/4 Volt
%
%                               Step size = 0.00002
%
%                               tap = 4
%*****
%*****
%***** Generate Signal Test *****
%*****
t = 0:0.0001:2 ;
d = sin(2*pi*1000*t) ;
energy = 0 ;
firX = 0 ;
Y = 0 ;
s = 5 ;
error = 0 ;
w = zeros(s,1) ;
fir = 0 ;
b = 1 ;
%*****
%***** CYCLE OF START *****
%*****
while b <= 2000
%*****
%***** Adaptive Transversal Filter *****
%*****
z = s-b;
if z >= 0

```

```

        firX(z+1:s,1)= d(1:b,1);
    else
        firX(1:s,1) = d(1-z:b,1);
    end

    fir= firX.*w;

%*****
%*****      Replica signal      *****
%*****
    Y(b,1) = sum(fir(1:s,1));

%*****
%*****      difference signal (reference signal-replica signal) *****
%*****
    error(b,1) = (d(b,1)/4)-Y(b,1);

%*****
%*****      Weight Adaptive Adjustment      *****
%*****

    if z >= 0
        for n1 = 1:b
            w(s-n1+1,1) = w(s-n1+1,1) + (0.00002).*error(b,1).*d(b-n1+1,1);
        end
    else
        for n1=1:s
            w(s-n1+1,1) = w(s-n1+1,1) + (0.00002).*error(b,1).*d(b-n1+1,1);
        end
    end

    b=b+1;

end

subplot(311);plot(d(1:500)/4) ; grid on ;
title(' reference signal '); xlabel('(samples)'); ylabel('(Amplitude)');
subplot(312),plot(Y(1:500)) ; grid on ;
title('replica signal'); xlabel('(samples)') ; ylabel('(Amplitude)');
subplot(313);plot(error(1:500)/4) ; grid on ;

```

```
title('error signal') ; xlabel('(samples)') ; ylabel('(Amplitude)');
```

```
%*****
```

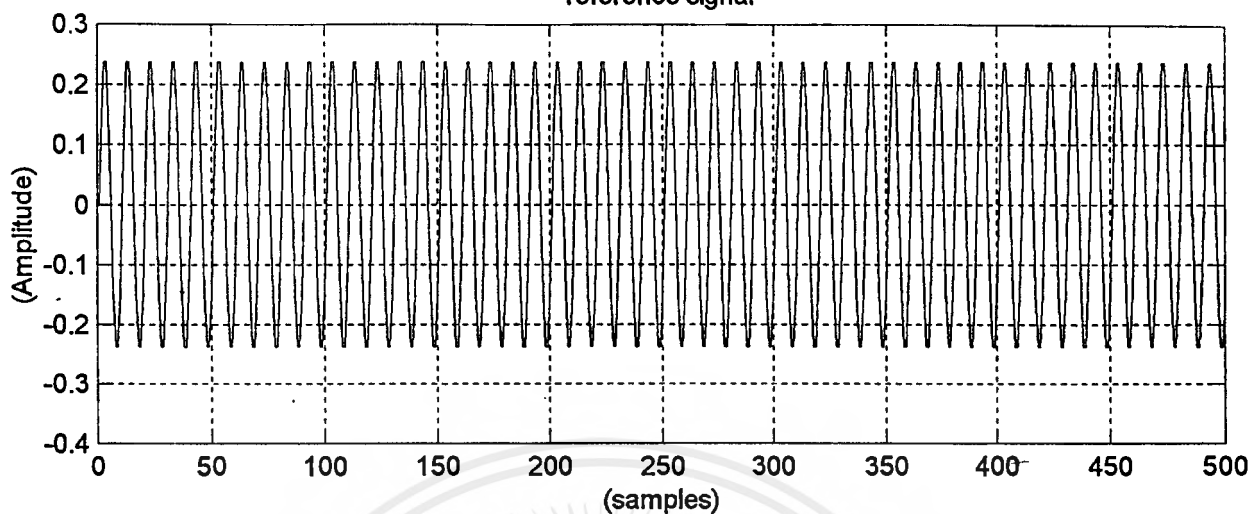
```
%***** END OF PROGRAM *****
```

```
%*****
```

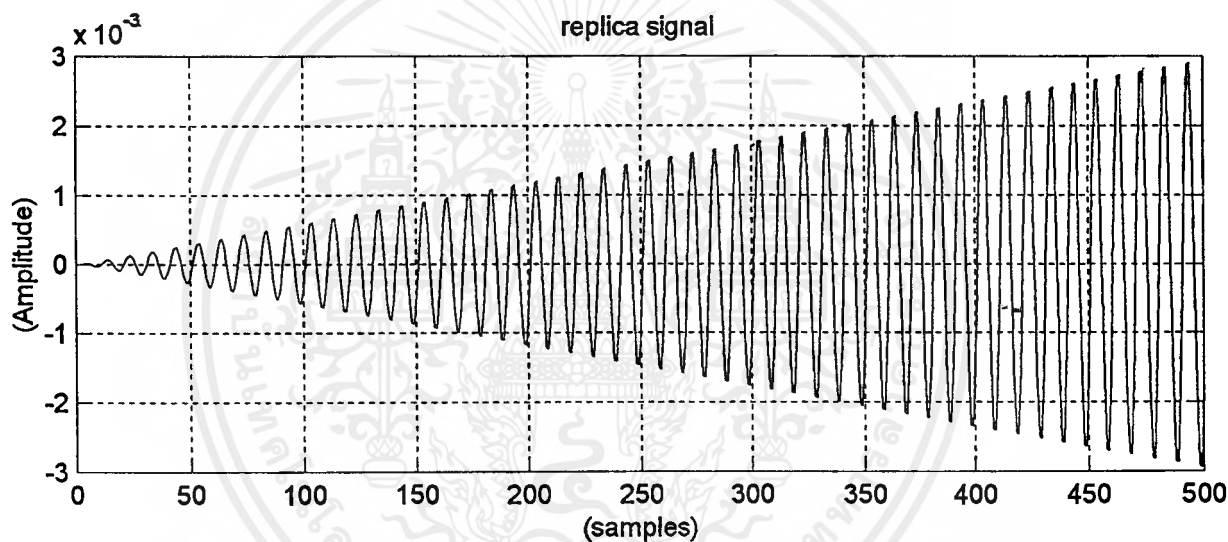


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

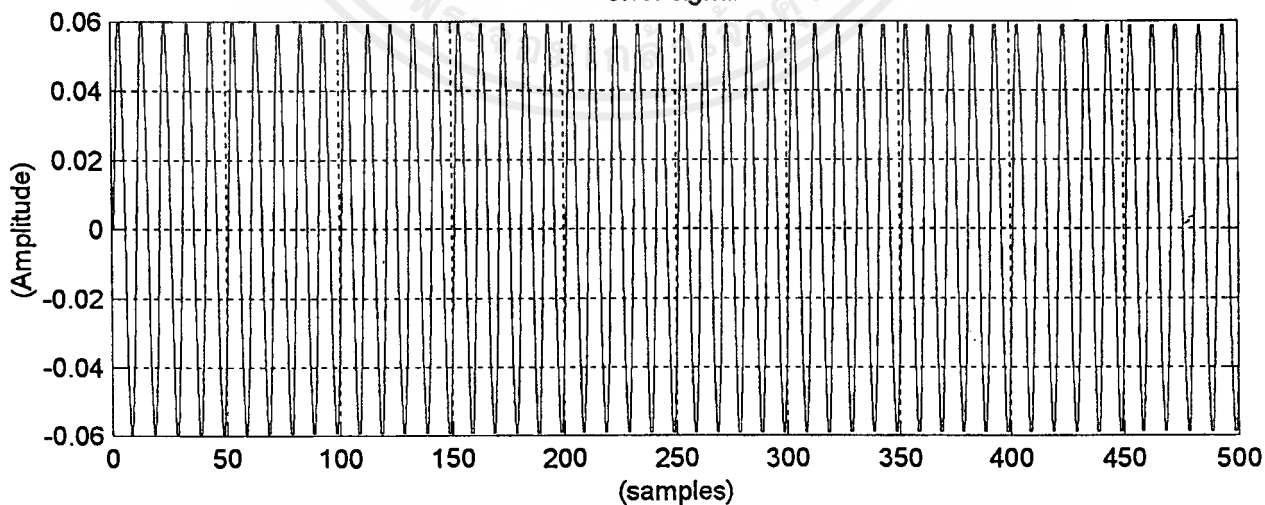
reference signal

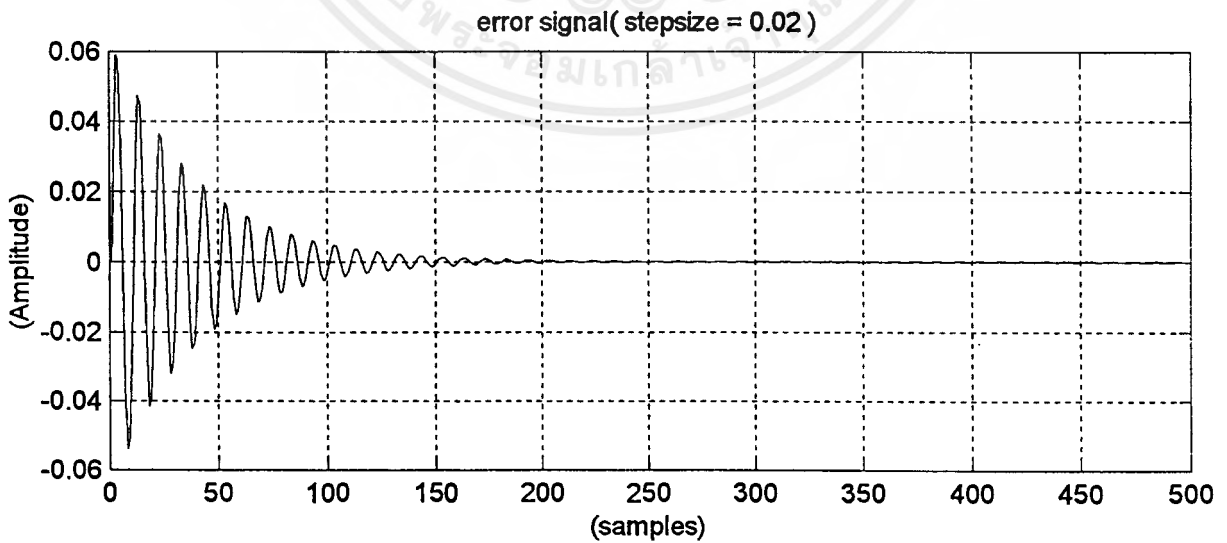
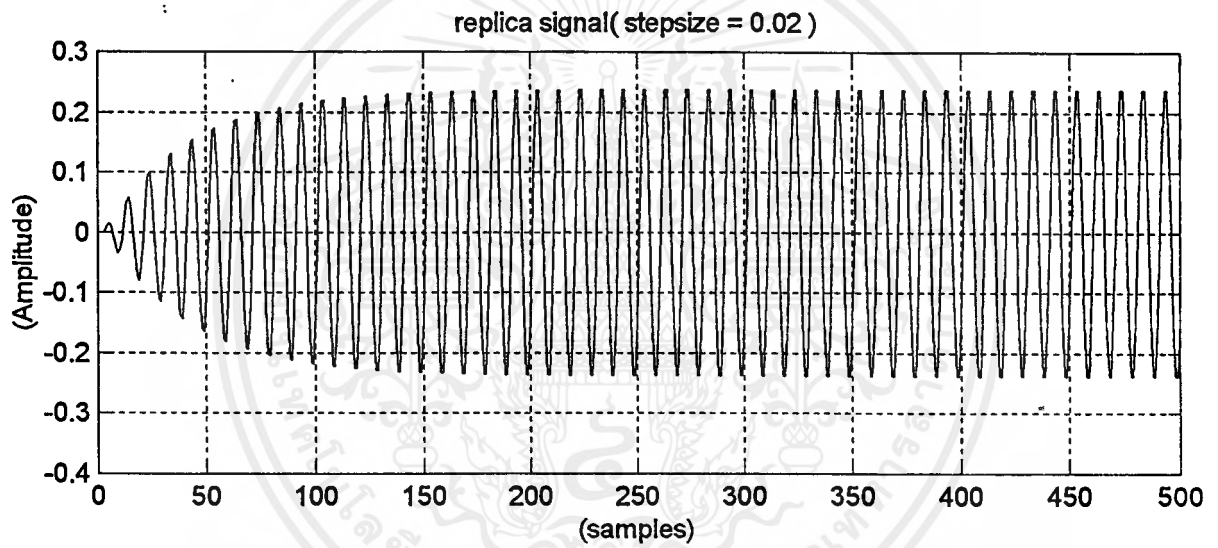
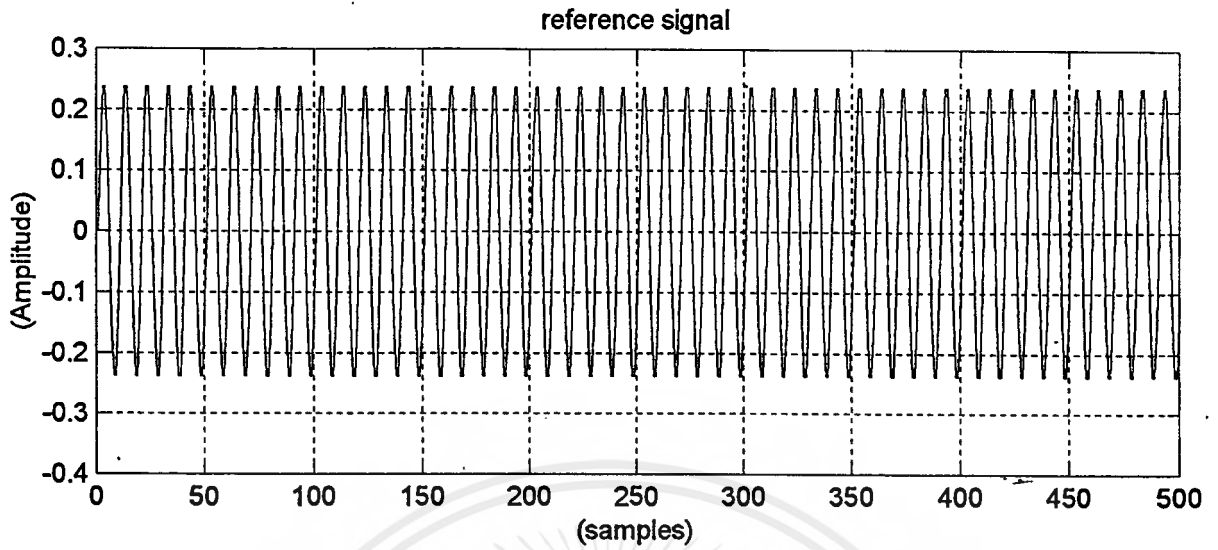


replica signal



error signal



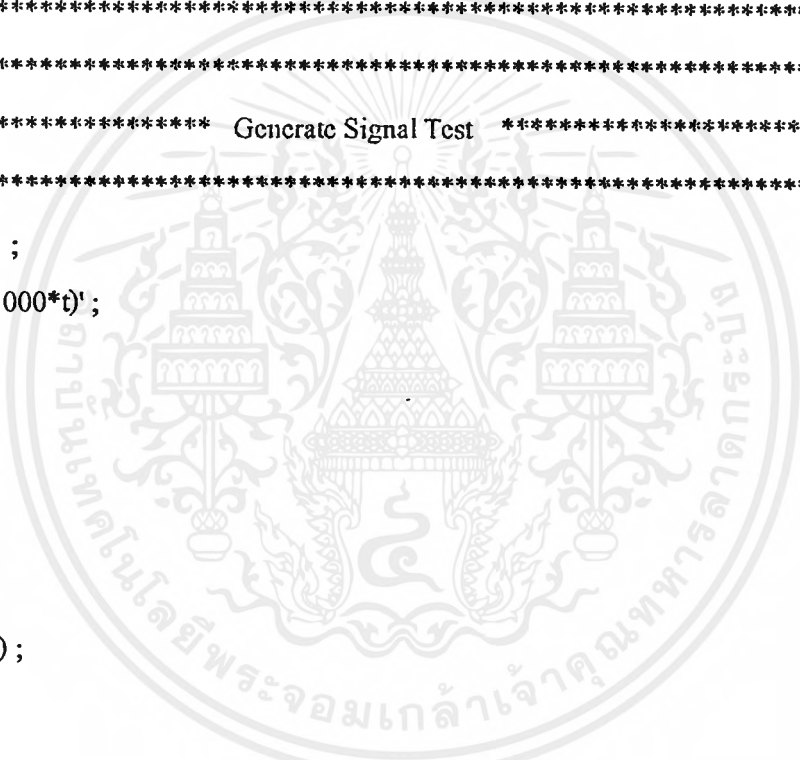


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%*****
%
%                               Program design
%
%                               Telecommunication Engineering
%
%                               TIME DOMAIN ACOUSTIC ECHO CANCELLER Project 3T/2
%
%                               With adaptive for sinusoidal signal test at 1KHz
%
%                               Amplitude sinusoidal reference 1 Volt
%
%                               Amplitude sinusoidal replica 1/4 Volt
%
%                               Short time energy = 16 samples
%
%                               tap = 4
%*****
%*****
%***** Generate Signal Test *****
%*****
t = 0:0.0001:2 ;
d = sin(2*pi*1000*t) ;
energy = 0 ;
firX = 0 ;
Y = 0 ;
s = 5 ;
error = 0 ;
w = zeros(s,1) ;
fir = 0 ;
b = 1 ;
%*****
%***** CYCLE OF START *****
%*****
while b <= 2000
%*****
%***** Check of Energy of signal Reference *****
%*****
if b <= 16
energy(b,1) = sum(d(1:b,1).^2);

```



```

else
    energy(b,1) = d(b-1).^2 + energy(b-1,1)-(d(b-16).^2);
end
if energy(b,1) == 0
    energy(b,1)=1 ;
end
%*****
%***** Adaptive Transversal Filter *****
%*****
z = s-b;
if z >= 0
    firX(z+1:s,1)= d(1:b,1);
else
    firX(1:s,1) = d(1-z:b,1);
end
fir= firX.*w;
%*****
%***** Replica signal *****
%*****
Y(b,1) = sum(fir(1:s,1));
%*****
%***** difference signal (reference signal-replica signal) *****
%*****
error(b,1) = (d(b,1)/4)-Y(b,1);
%*****
%***** Weight Adaptive Adjustment *****
%*****
if z >= 0
    for n1 = 1:b
        w(s-n1+1,1) = w(s-n1+1,1) + (1/energy(b,1)).*error(b,1).*d(b-n1+1,1);
    end
else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for n1=1:s
```

```
    w(s-n1+1,1) = w(s-n1+1,1) + (1/energy(b,1)).*error(b,1).*d(b-n1+1,1);
```

```
end
```

```
end
```

```
b=b+1;
```

```
end
```

```
subplot(311);plot(d(1:500)/4) ; grid on ;
```

```
title(' reference signal '); xlabel('(samples)'); ylabel('(Amplitude)');
```

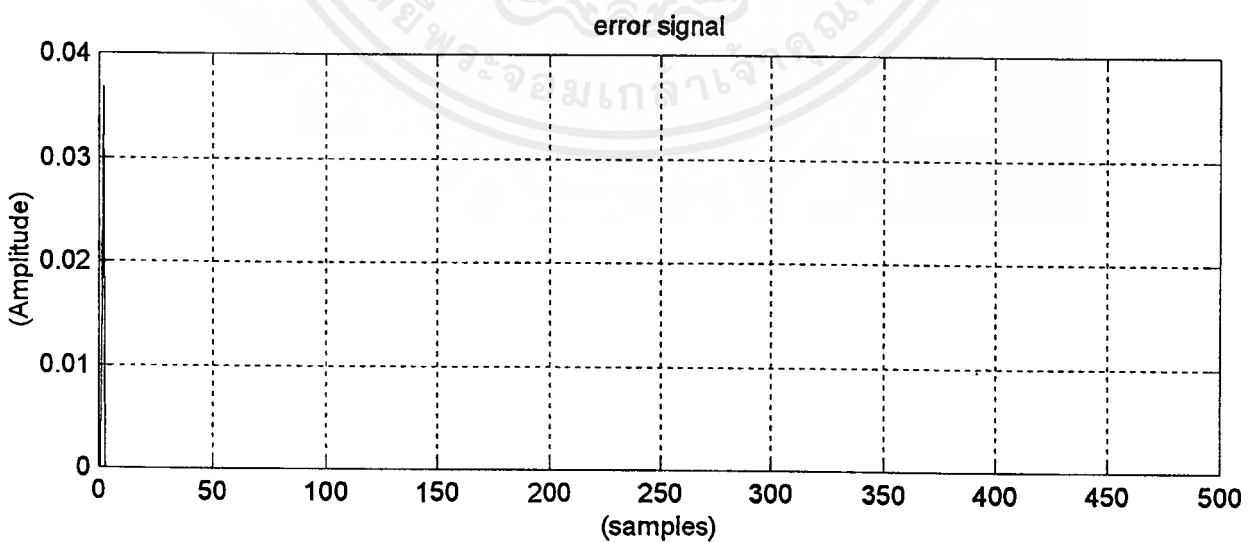
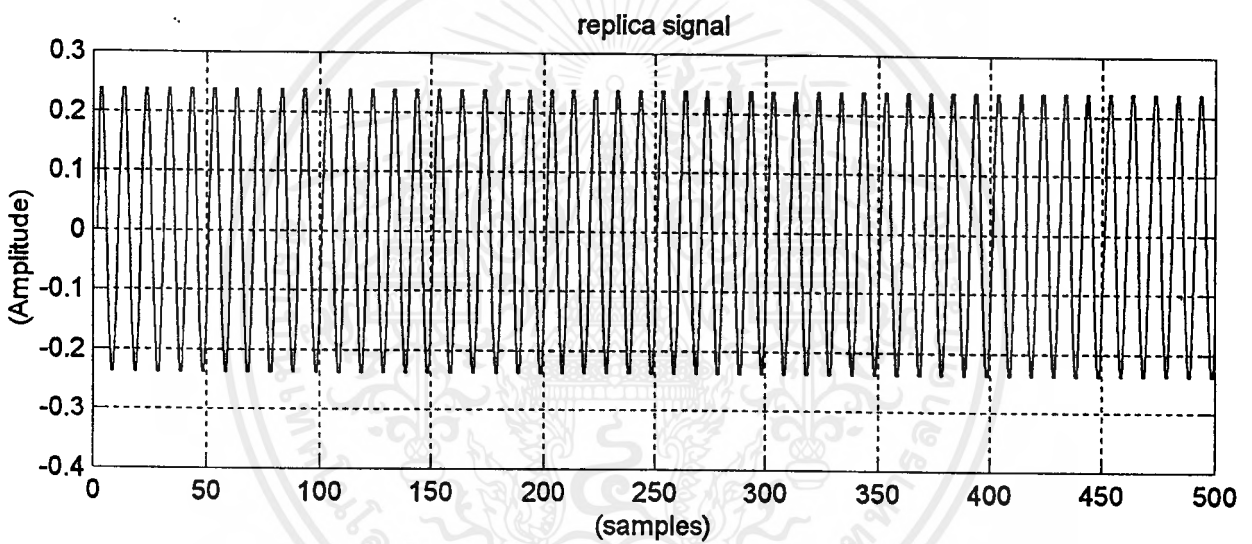
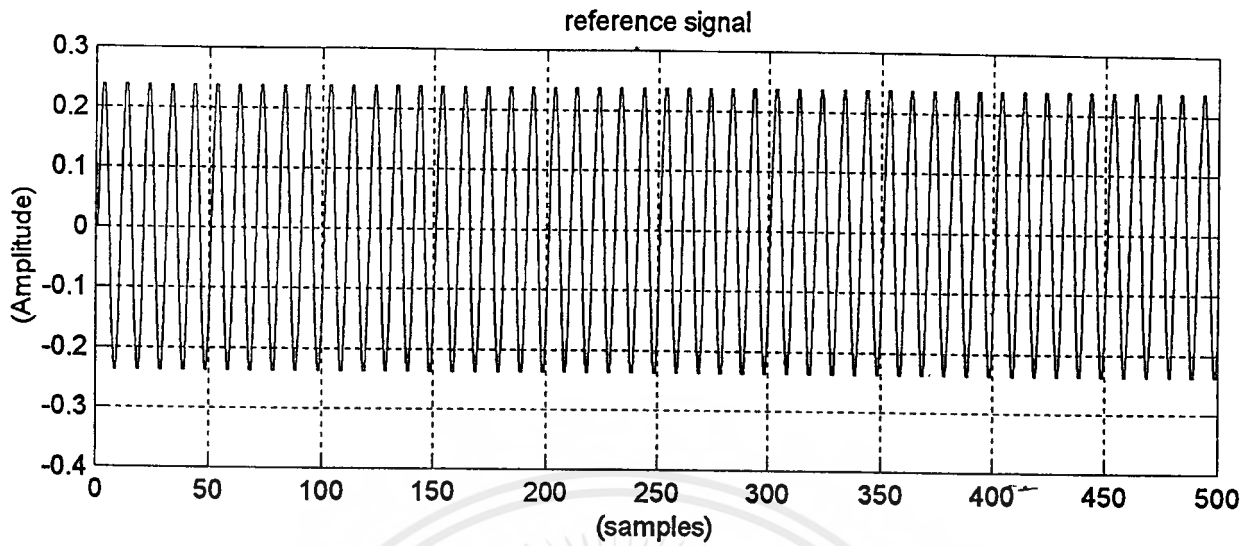
```
subplot(312),plot(Y(1:500)) ; grid on ;
```

```
title('replica signal'); xlabel('(samples)') ; ylabel('(Amplitude)');
```

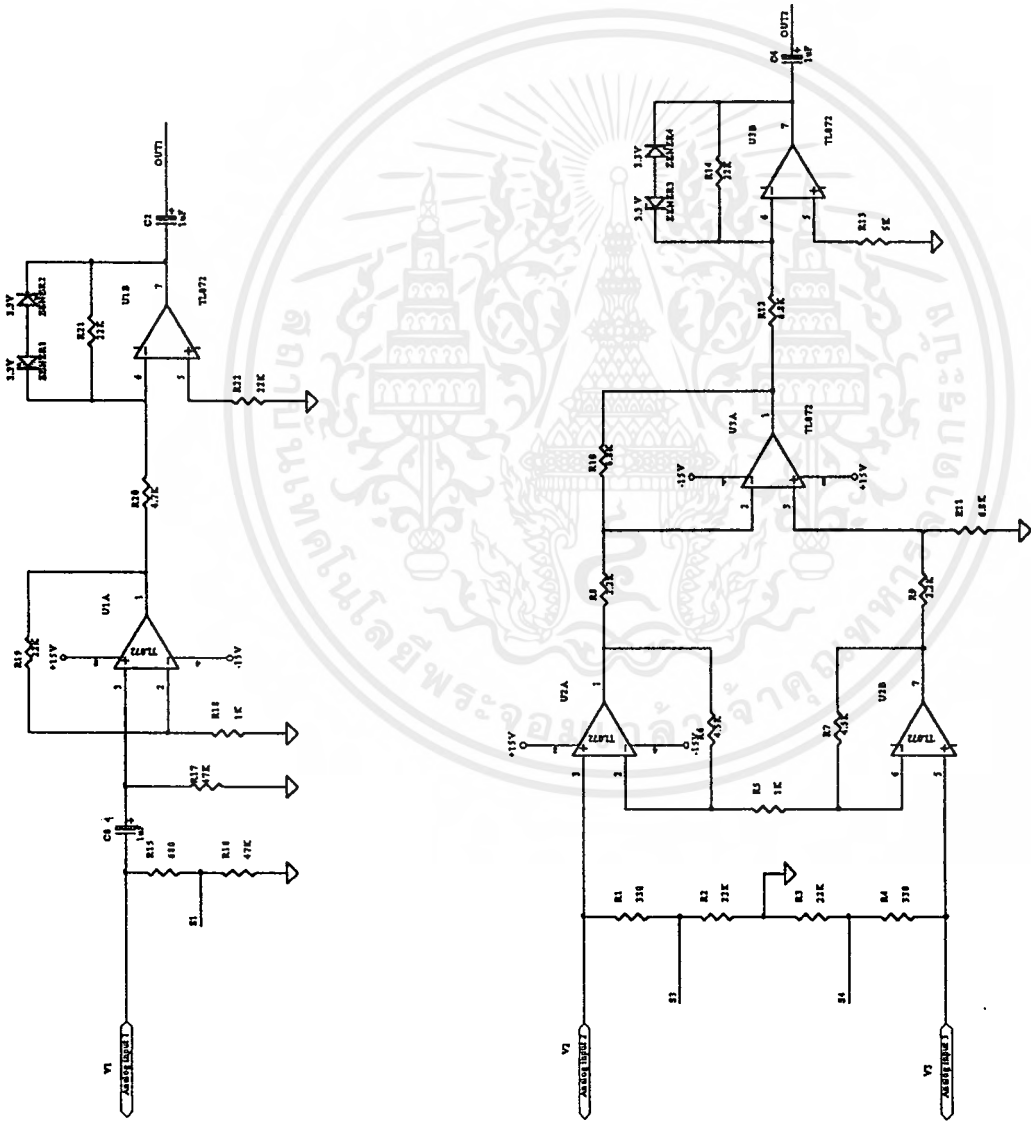
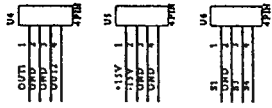
```
subplot(313);plot(error(1:500)/4) ; grid on ;
```

```
title('error signal') ; xlabel('(samples)') ; ylabel('(Amplitude)');
```

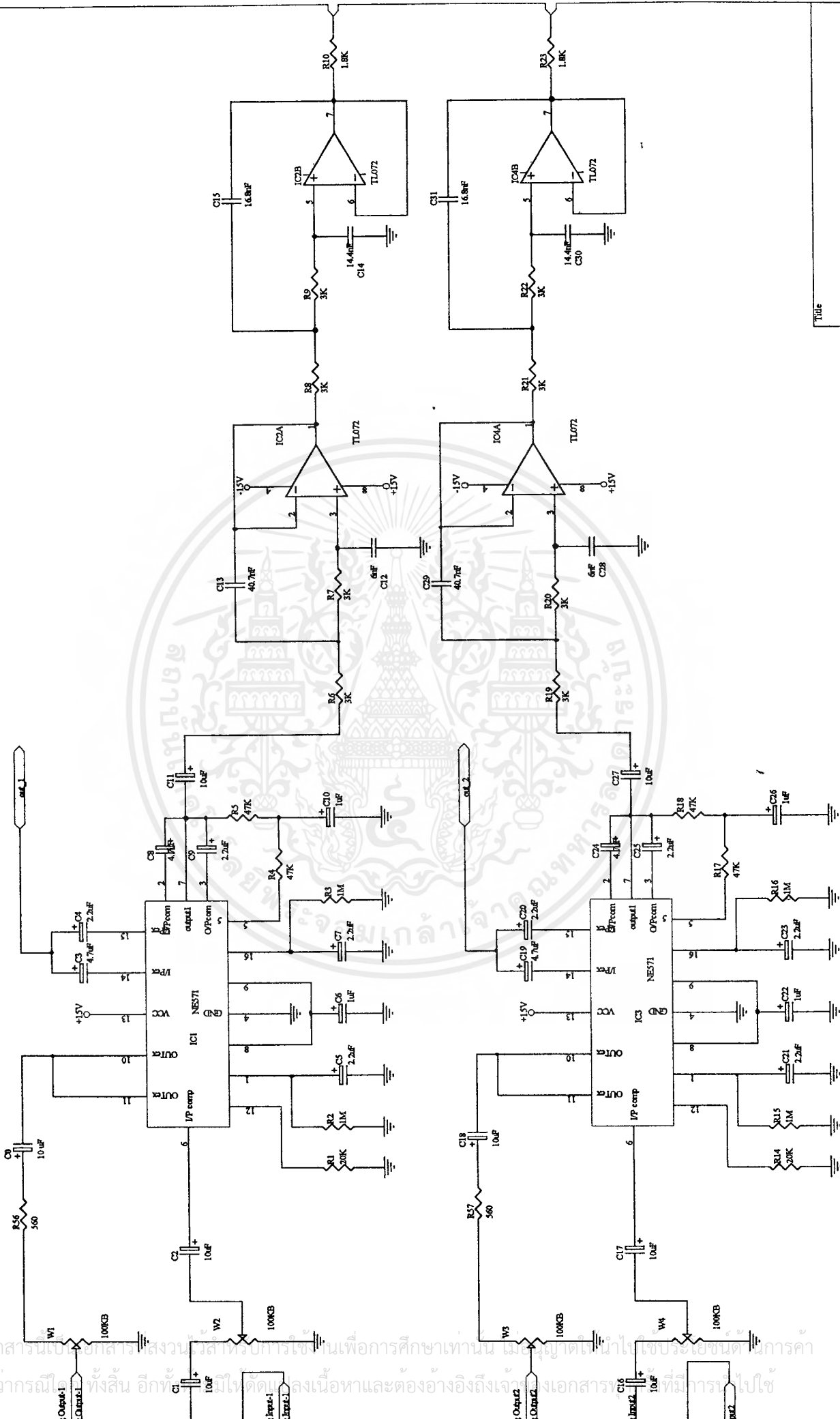
```
%*****  
%***** END OF PROGRAM *****  
%*****
```



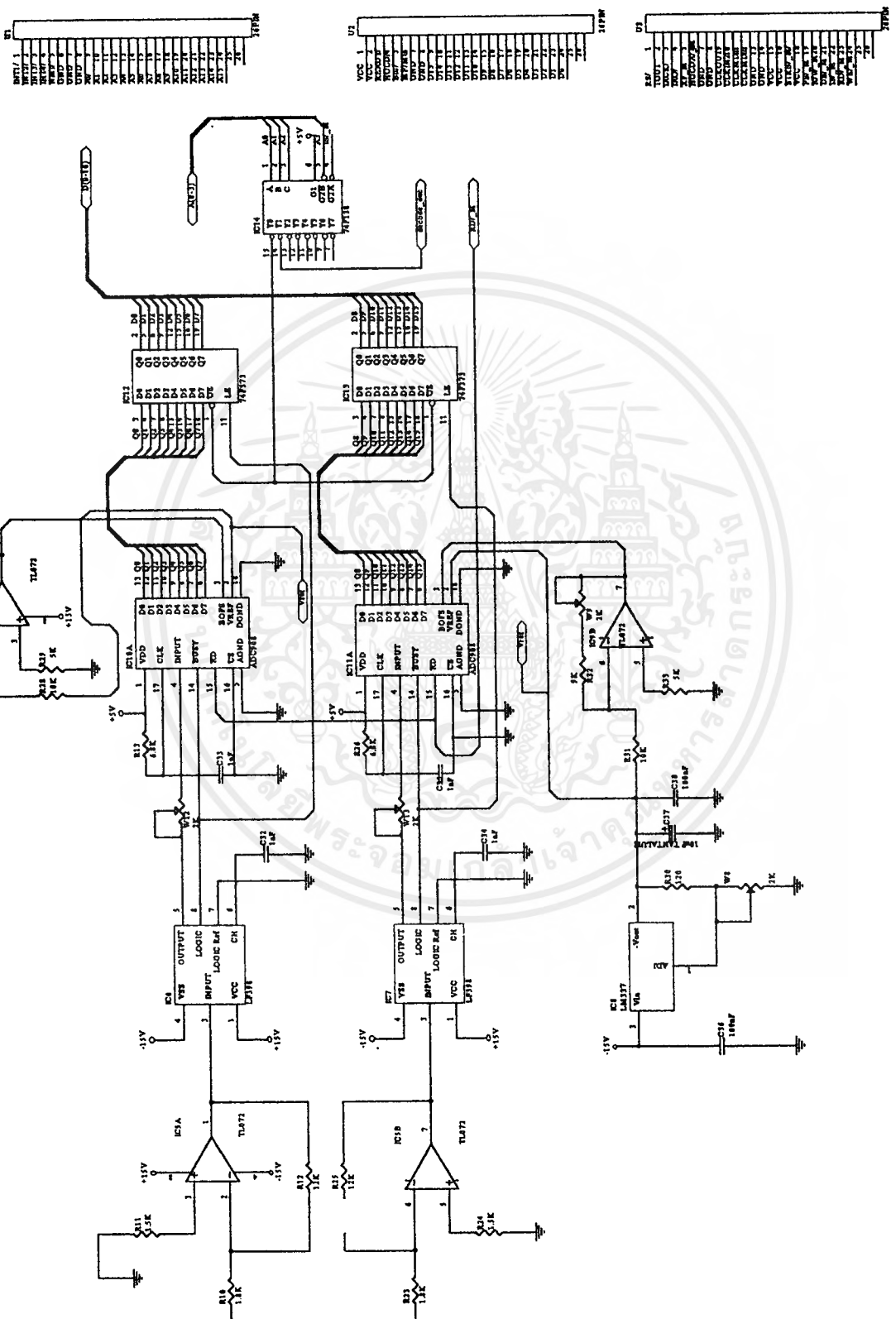
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



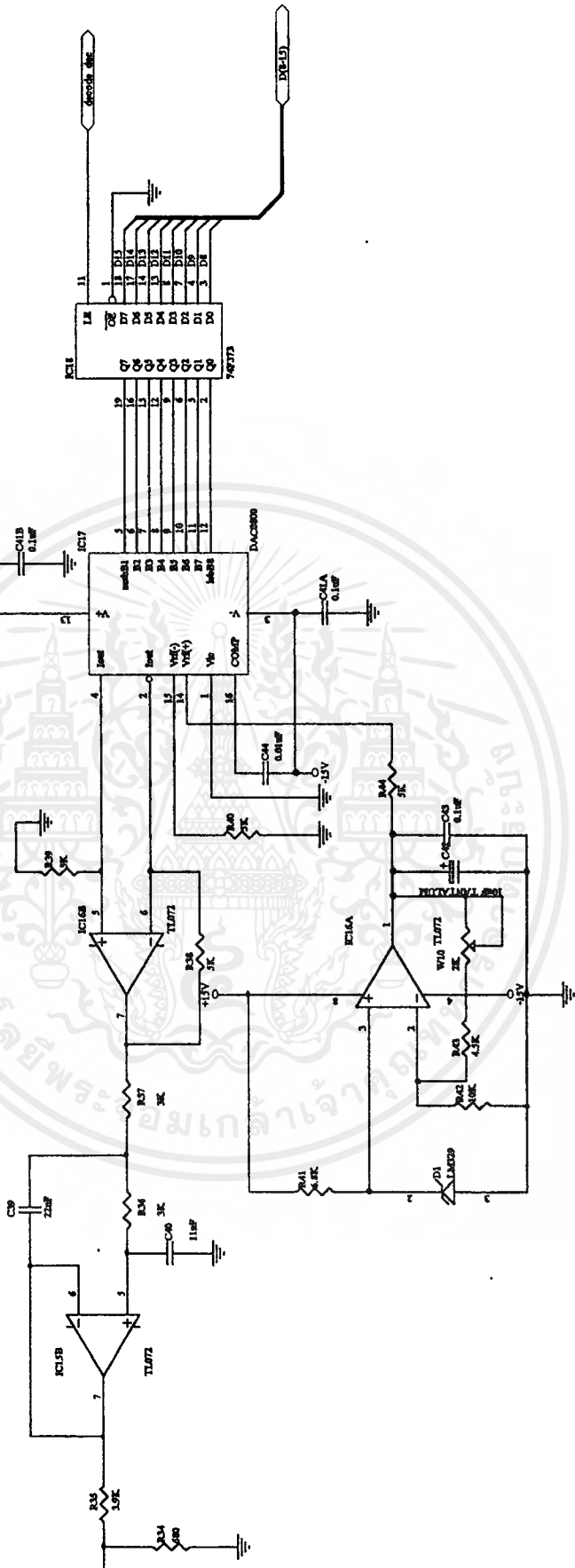
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์หรือการใช้งานเพื่อการศึกษานาน ในประเทศไทยภายใต้ลิขสิทธิ์ของเอกสารนี้
 ไม่สามารถนำออกให้คนอื่นได้โดยไม่ขออนุญาตจากเจ้าของเอกสารนี้
 หากต้องการข้อมูลเพิ่มเติมหรือข้อสงสัย กรุณาติดต่อฝ่ายบริการลูกค้าของเรา

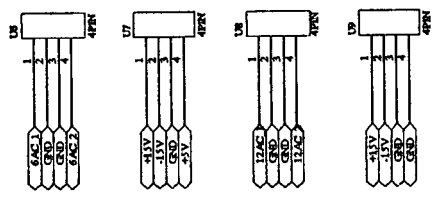
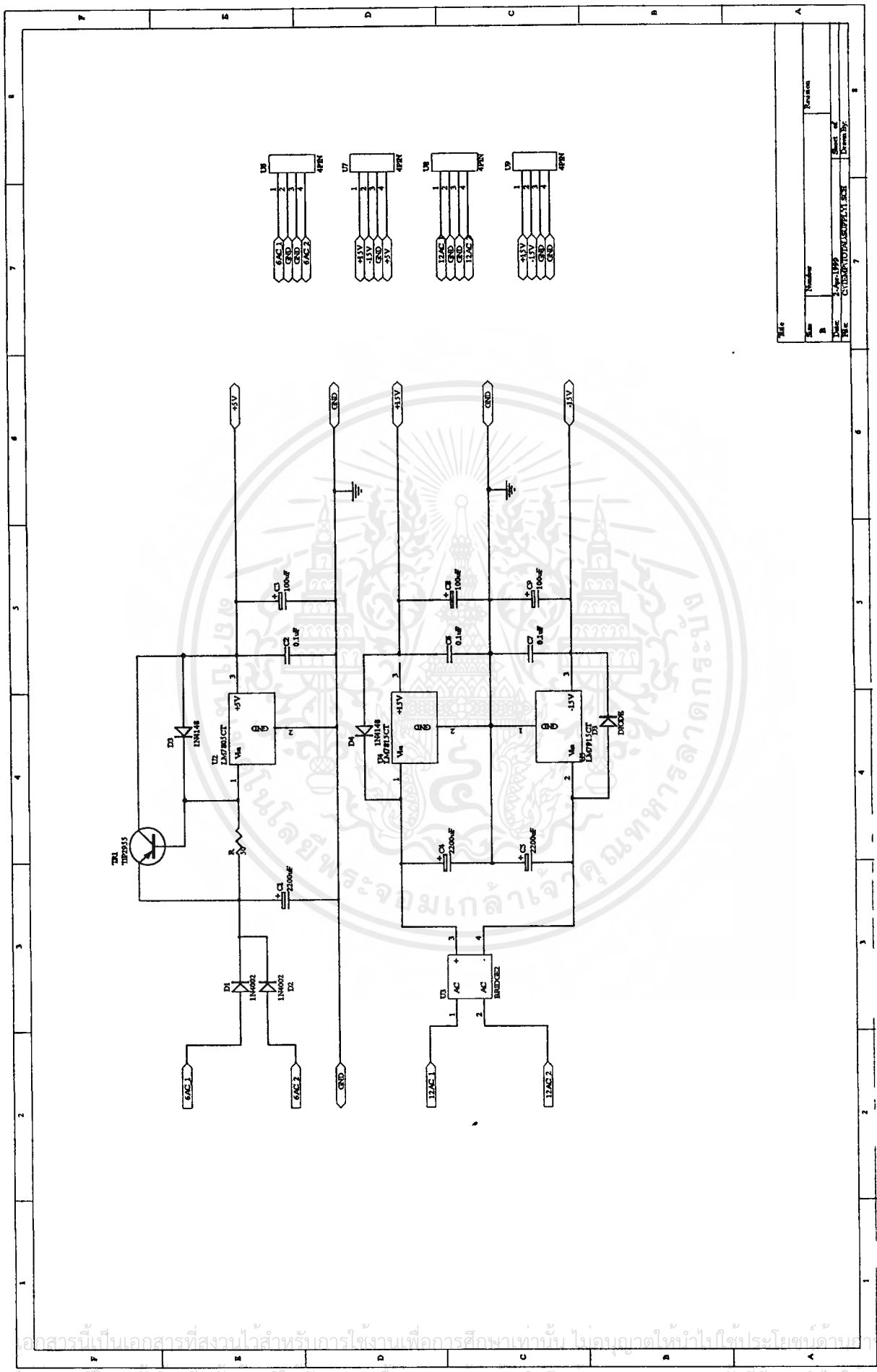


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



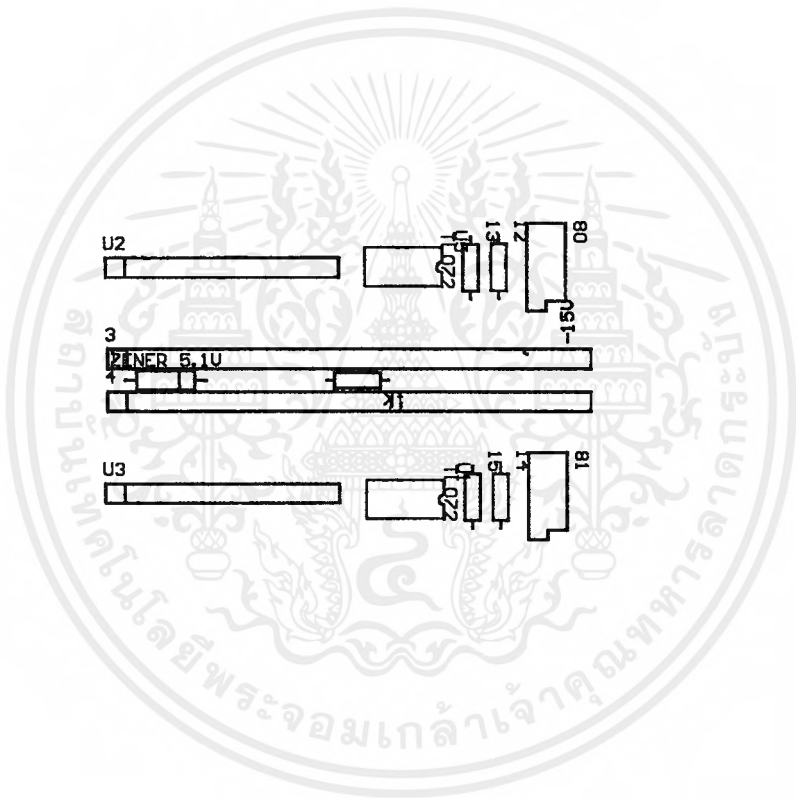
Title		Revision	
Rev	Number	Rev	Reason
1	1	1	Initial
Date: 25-Aug-1993		Sheet of 2	
File: C:\TEMP\85810\DAC.SCH		Drawn By:	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



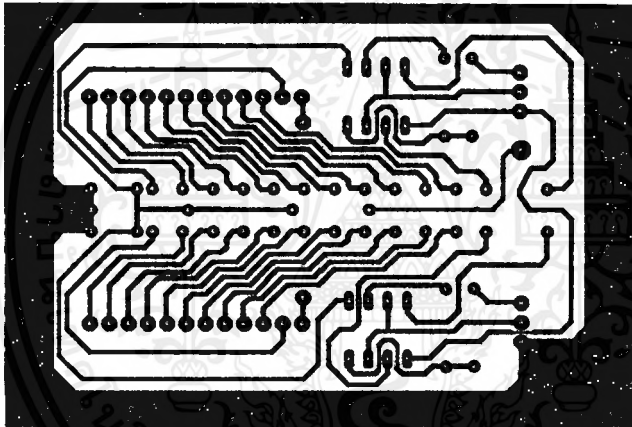
Title		Revision	
Rev	Number	Rev	Revision
B		2	2
Doc	2-AC-1008	Drawn By	
File	C:\MSD\POWER\SUPPLY.DWG	Checked By	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้วางใจใช้ประโยชน์นอกเหนือจากนี้
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



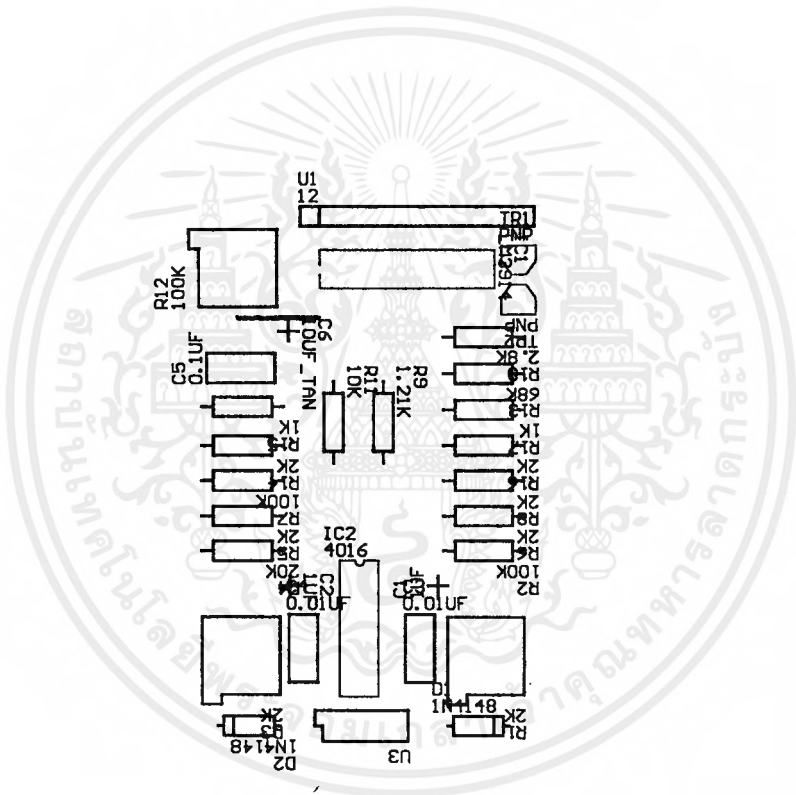
รูปถ่ายอุปกรณ์วงจร Display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



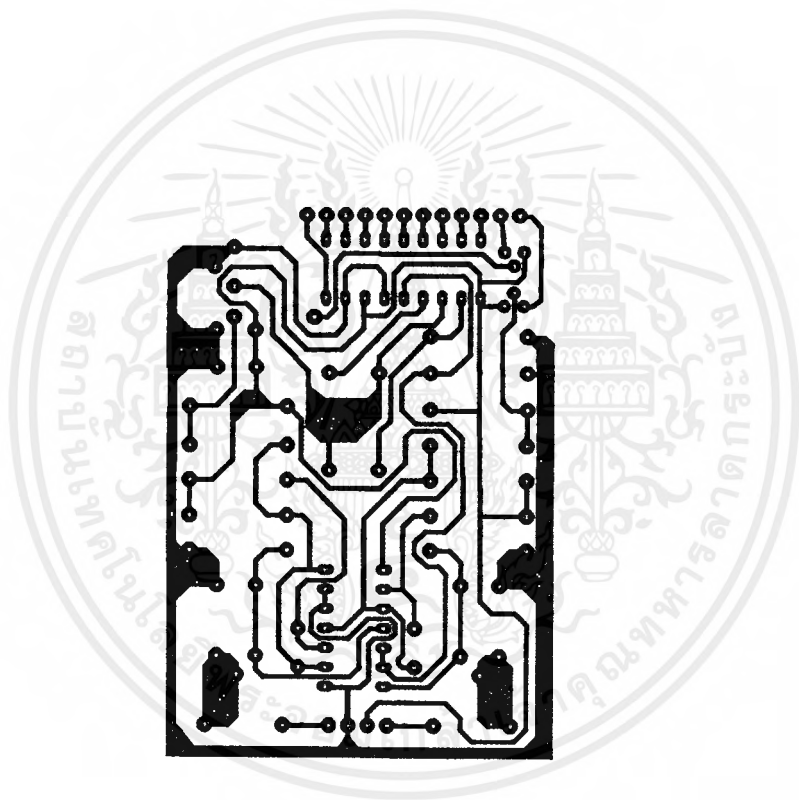
รูปลายวงจรภาค Display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



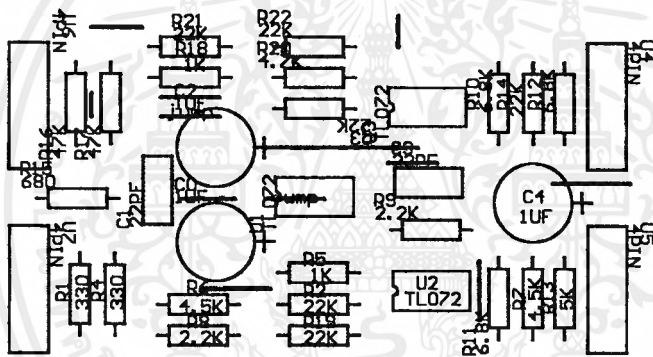
รูปลายอุปกรณ์วงจรภาค Drive display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



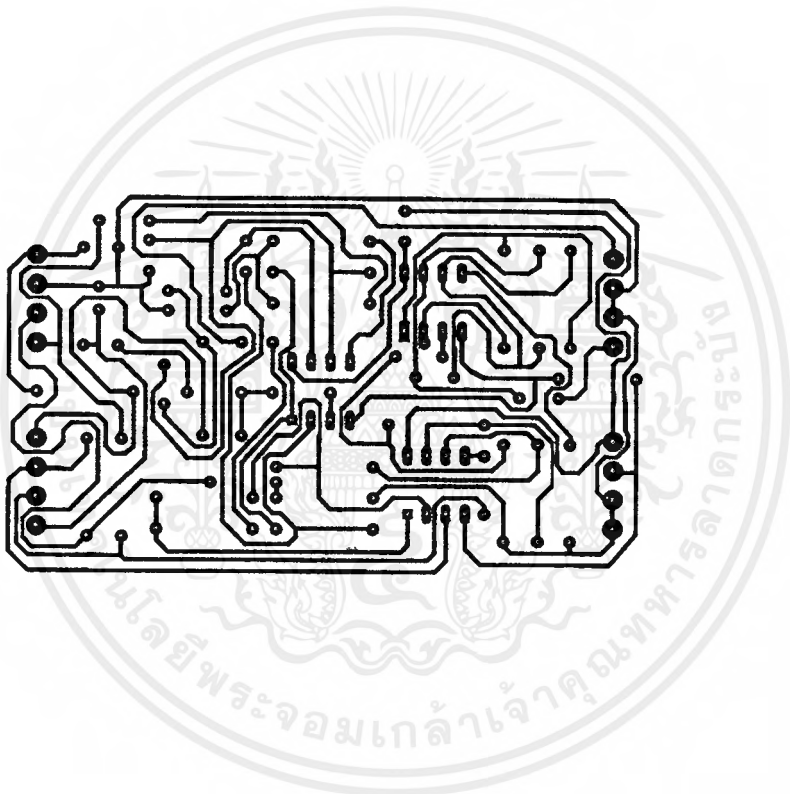
รูปลายวงจรของภาค Drive display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



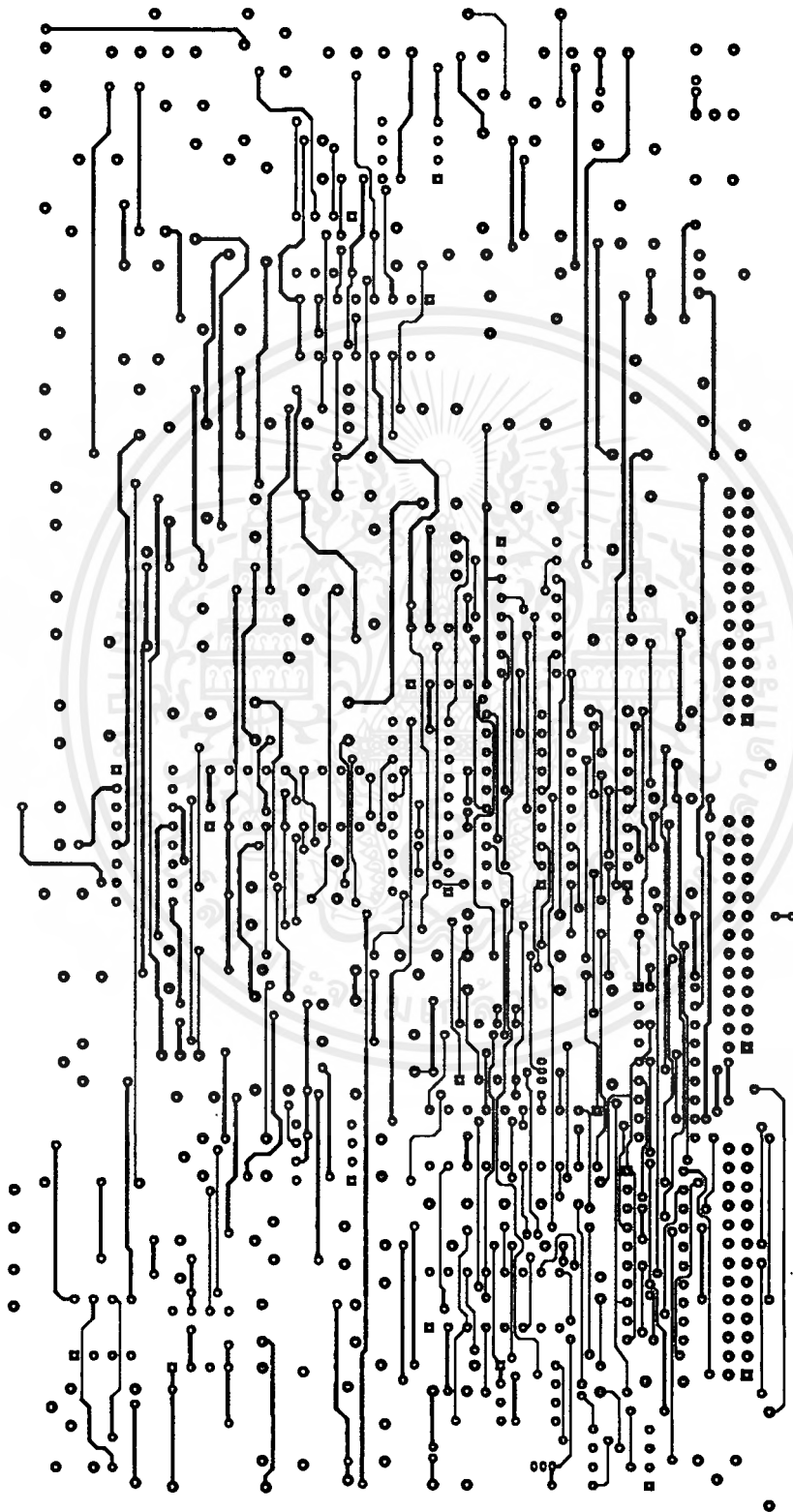
รูปลายอุปกรณ์ของวงจรภาค Analog input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



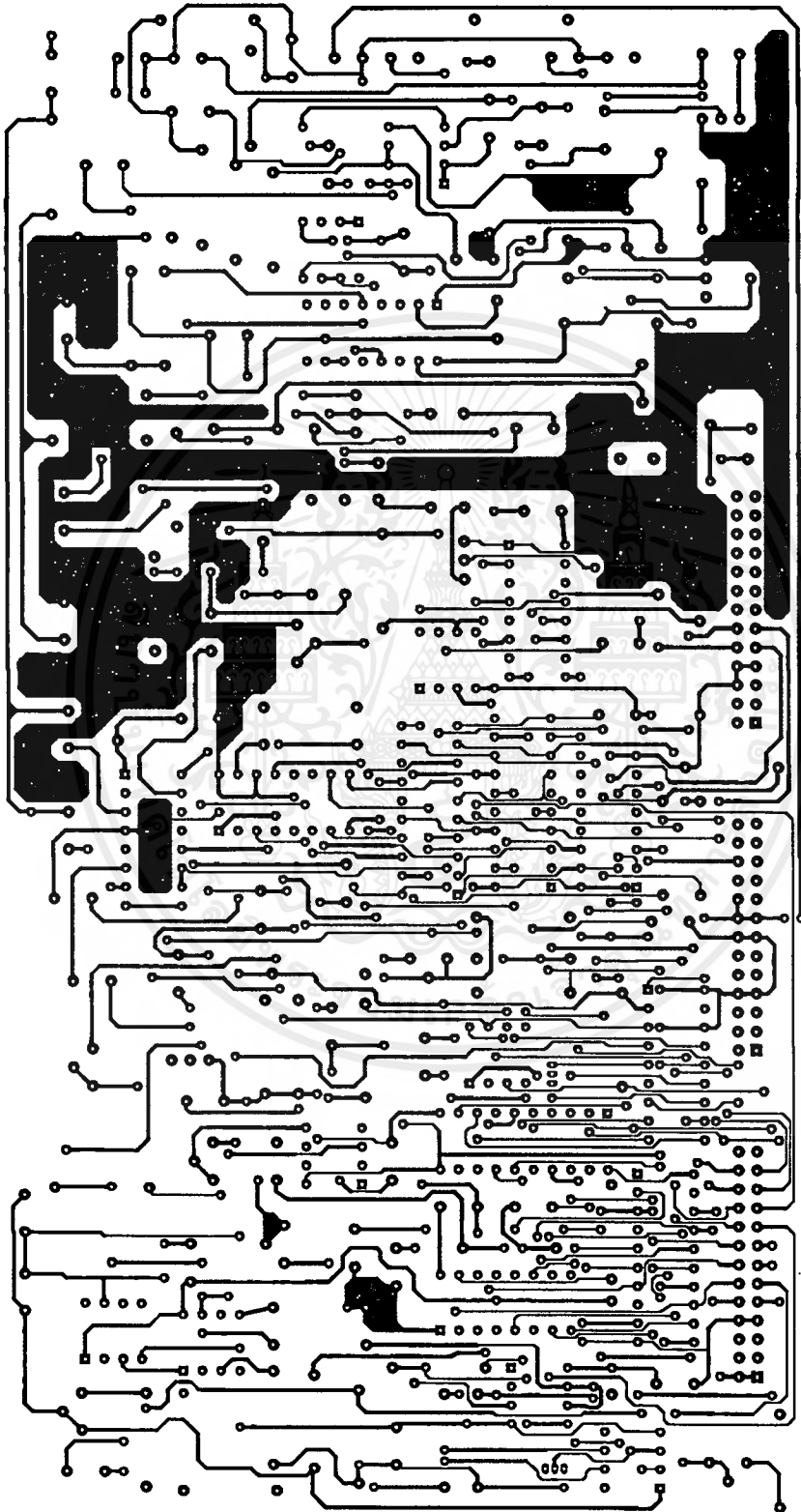
รูปลายวงจรของอุปกรณ์ภาค Analog input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



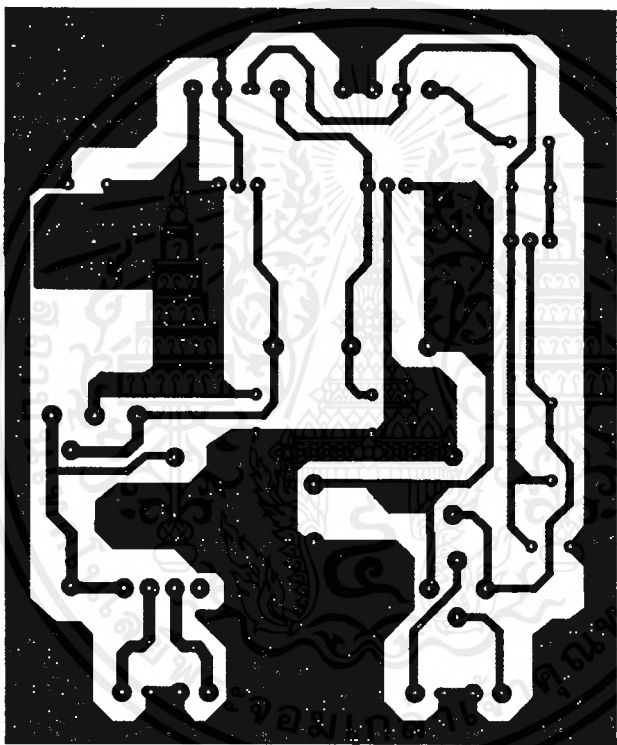
รูปลายวงจรค่านบนของอุปกรณ์ภาค Analog interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปลายวงจรด้านล่างของอุปกรณ์ภาค Analog interface

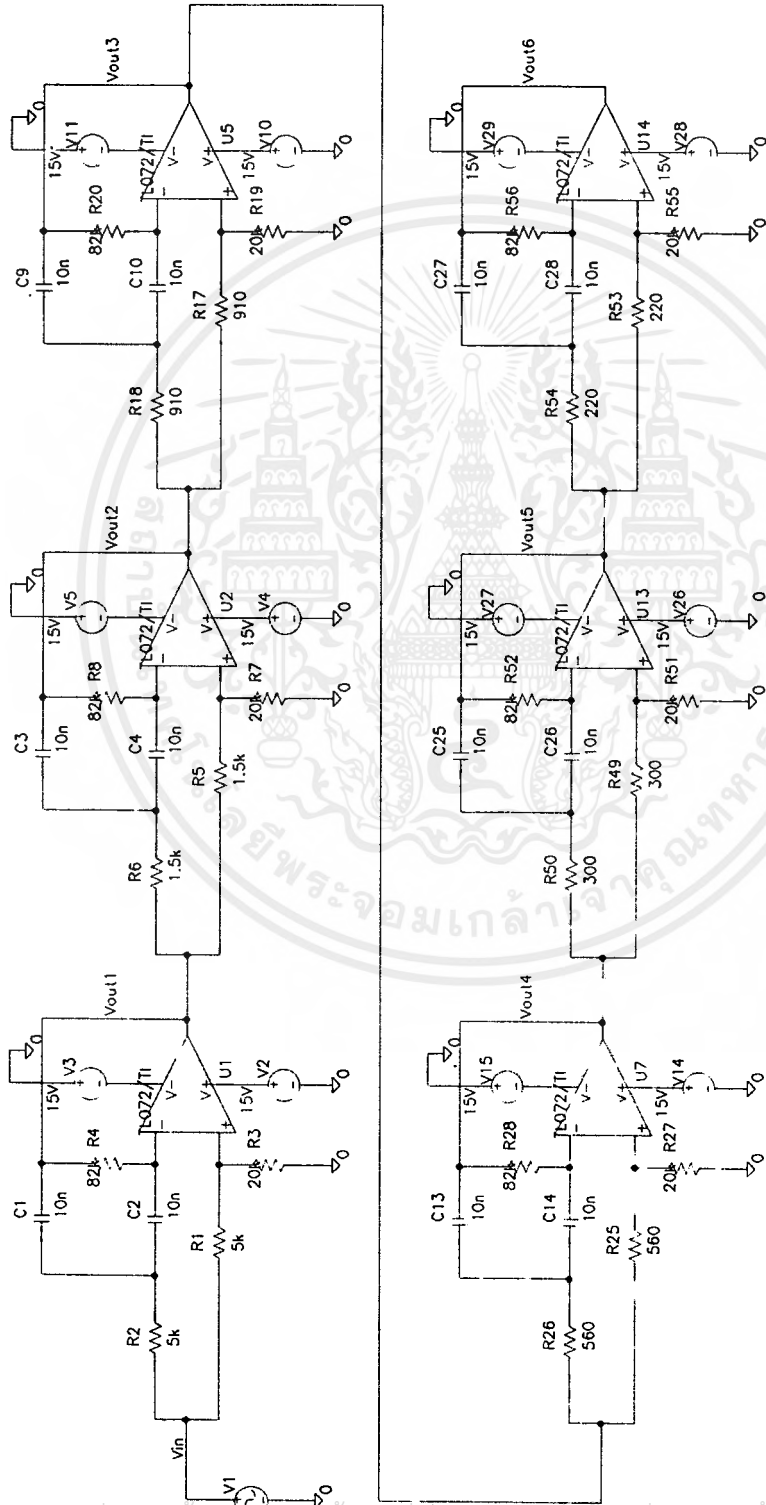
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปลายวงจรของอุปกรณ์ภาคจ่ายไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

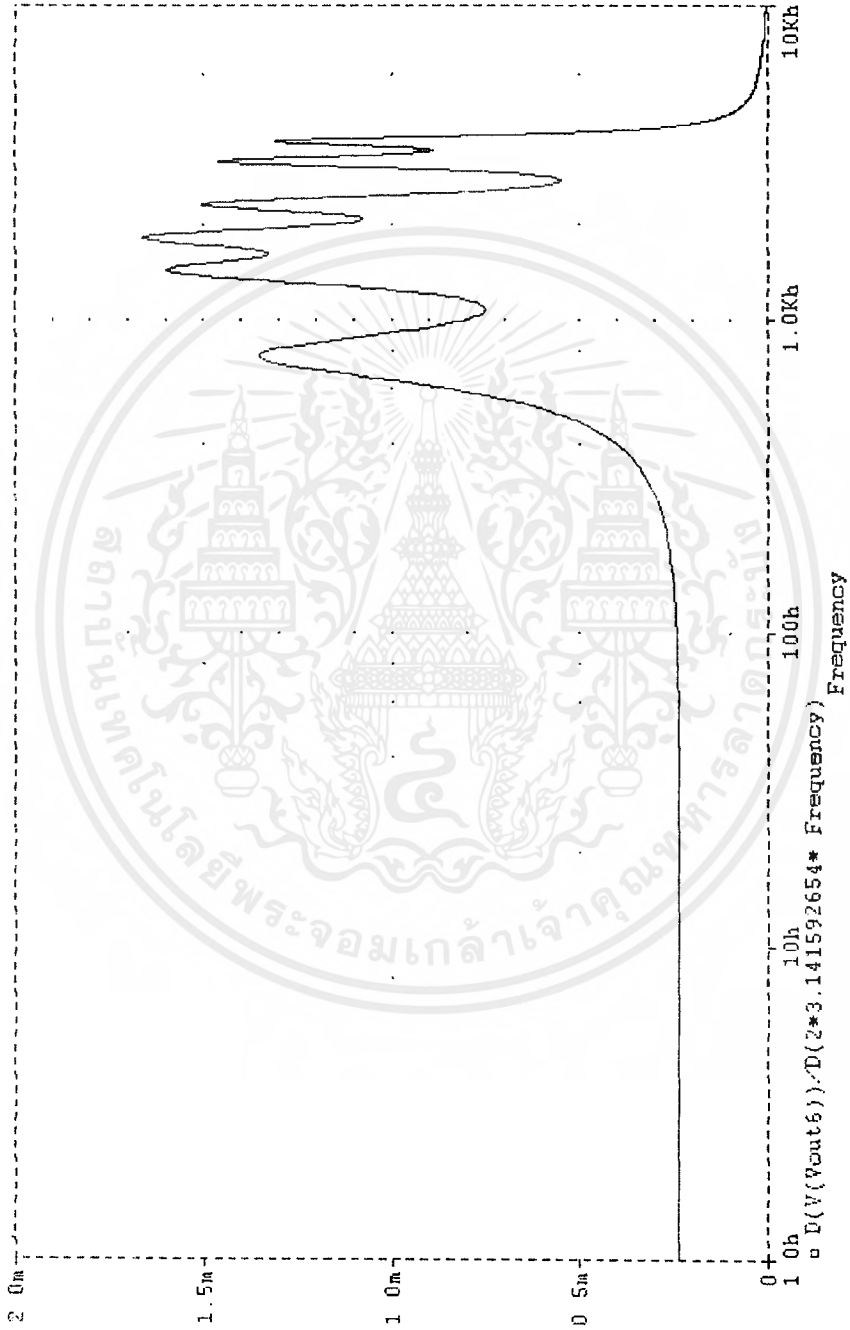
รูปวงจรของชุด Delay Equalizer ที่ใช้ในการทดลองกับสัญญาณ Sinewave



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

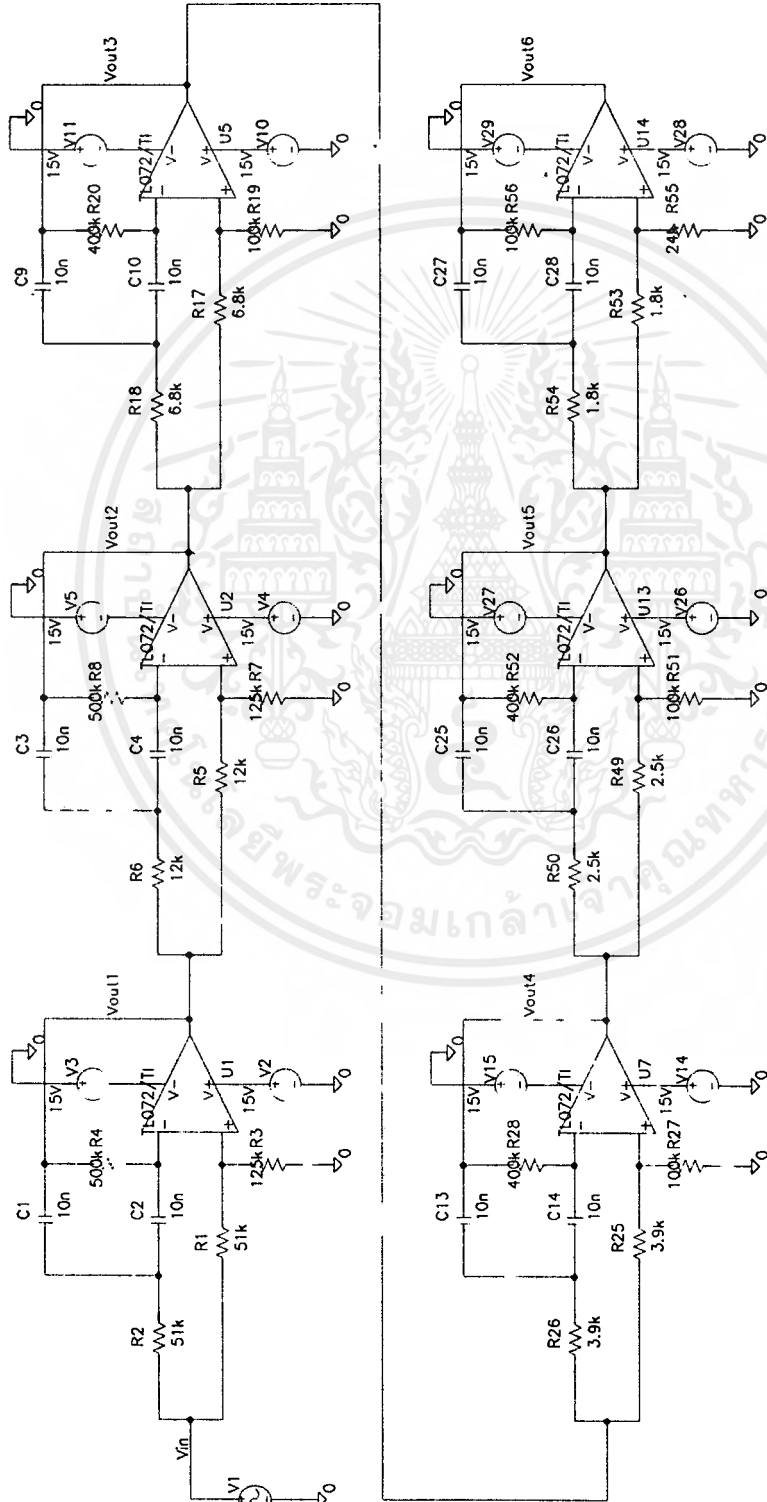
ผลการ Simulation ของวงจรชุด Delay Equalizer ที่ใช้ในการทดลองกับสัญญาณ Sinewave

untitled.bmp (800x600x16M bmp)



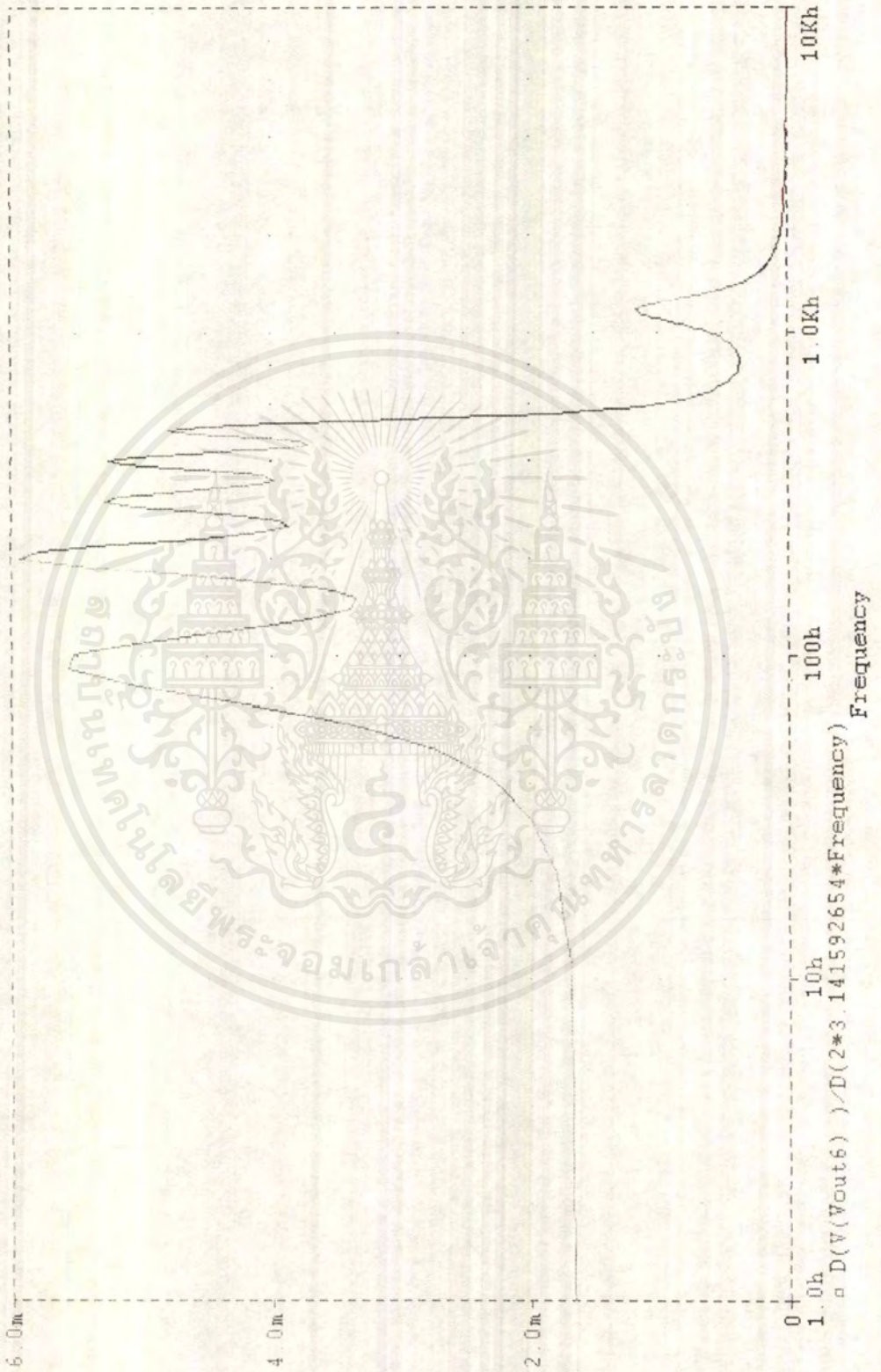
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปวงจรของชุด Delay Equalizer ที่ใช้ในการทดลองกับสัญญาณจากไมโครโฟน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

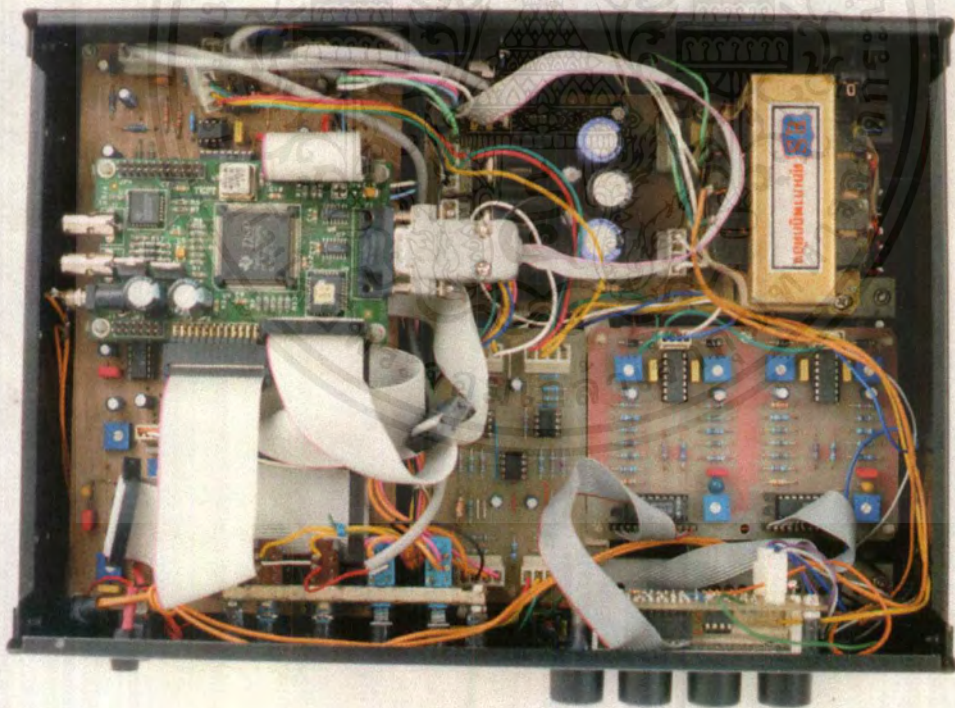
ผลการ Simulation ของวงจรชุด Delay Equalizer ที่ใช้ในการทดลองกับสัญญาณ ไมโครโฟน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

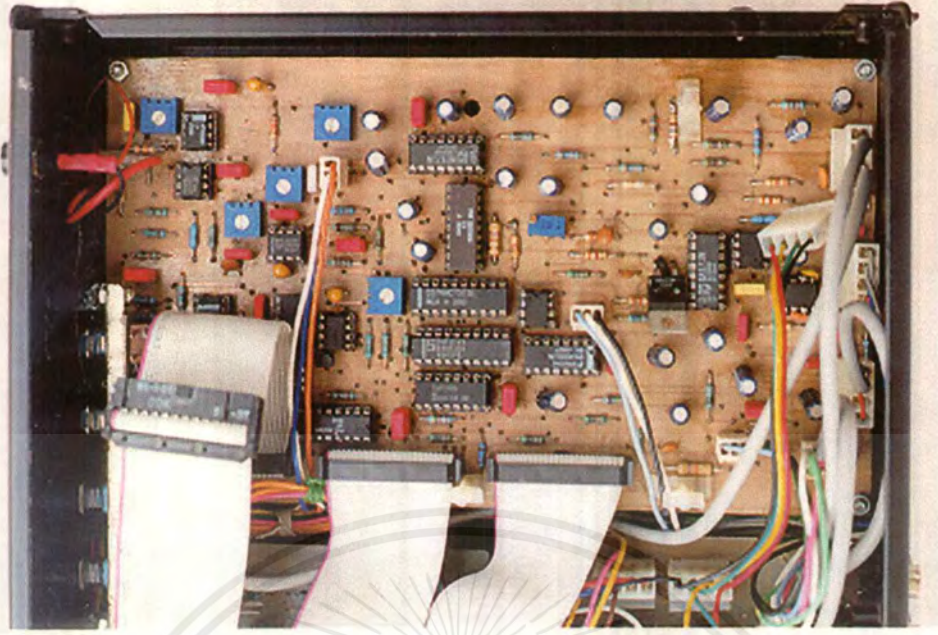


รูปถ่าย เครื่องกำจัดเสียงก้อง (Acoustic Echo Canceller)

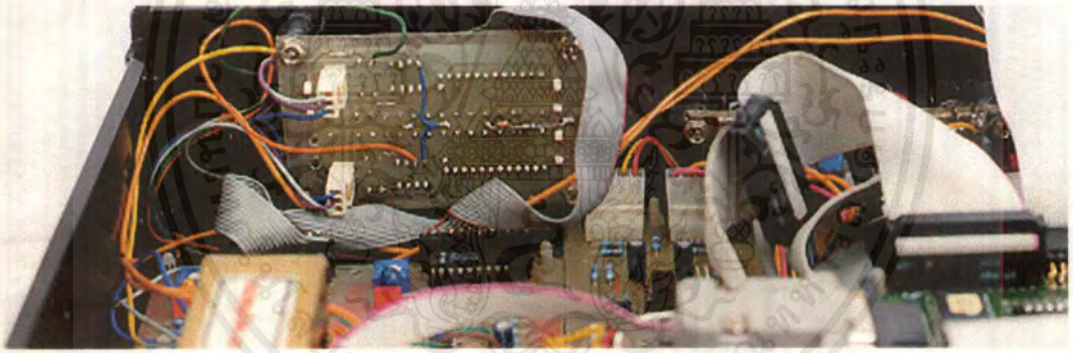


รูปถ่าย อุปกรณ์ภายในโดยรวมของเครื่องกำจัดเสียงก้อง (Acoustic Echo Canceller)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปถ่าย วงจรภาค Analog interface เพื่อติดต่อกับบอร์ด DSK



รูปถ่าย แผ่นปริ๊นซ์ของวงจรภาค Display



รูปถ่าย เครื่องกำจัดเสียงก้อง (Acoustic Echo Canceller) และ ชุดทดลอง Delay Equalizer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาดเห็นาไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณเป็นอย่างสูง สำหรับ

อาจารย์ที่ปรึกษา รศ. ณรงค์ เหมกรณ์

อาจารย์ทุกๆท่านที่กรุณาให้คำปรึกษาอันมีคุณค่าและประโยชน์เป็นอย่างมาก

คุณพ่อและคุณแม่ที่ให้การสนับสนุนเป็นอย่างดี

เพื่อนๆทุกคนที่ให้การช่วยเหลือและแนะนำสิ่งต่างๆที่เป็นประโยชน์เป็นอย่างดี

ขอขอบคุณที่ได้ให้การสนับสนุนในการทำวิทยานิพนธ์นี้เป็นอย่างสูง

จาก คณะผู้จัดทำ



หนังสืออ้างอิง

1. Texas Instrument, TMS320C26 User's Guide, 1993
2. Texas Instrument, TMS320C50 User's Guide, 1993
3. Texas Instrument, TMS320C26 DSP Starter Kit User's Guide, 1993
4. Texas Instrument, TMS320C50 DSP Starter Kit User's Guide, 1993
5. Anibal R. Figueiras-Viaal (Ed.), "Digital Signal Processing in Telecommunication" European Project cost#229 Technical Contributions Springer Verlag, Berlin .
6. Rinay K. Ingle, John G. Proakis, "DSP Using MATLAB V.4 " Bookware Companion Series PWS Publishing Company And International Thomson Publishing Company ISBN 0-534-93805-1
7. Kishan Shenoj , Rentice hall , "DSP in Telecommunication", 1995 ISBN 0-13-096751-3
8. N.Kalouptsidis, John Wiley & Sons "Signal Processing System Theory and Design", 1997.
9. Marvin E. Frerking, "DSP in Communication Systems", Van Nostrand Reinhold is an international Thomsom Publishing Company ISBN 0-442-01616-6
10. Bernard Widrow , Samuel. D. Stearns, "Adaptive Signal Processing", 1985 ISBN 013-00402901
11. Glenn Zelniker, Fred J. Taylor Marcel Dekker , "Advanced Digital Signal Processing Theory and Applications".
12. Simon Haykin , "Adaptive Filter Theory Third Edition.", 1996 ISBN 0-13-397985-7
13. "Optimum Signal Processing An Introduction second Edition", Sophocles J. Orfanidis McGrew-Hill International Editions Electrical engineering Series
14. "Adaptive System Identification and Signal Processing Algorithms", Prentice-Hall edited by N.Kalouptsid , S.Theodoridis ISBN 0-13-006545-5
15. JAE S.LIM, ALAN V. Oppenheim, "Advanced Topics in Signal Processing", Prentice-Hall
16. Peter M. Clarkson , "Optimal and Adaptive Signal Processing", CRC Press , INC.
17. Johnny R. Johnson , "Introduction to Digital Signal Processing", Prentice-Hall.
18. Richard A.Haddad , Thomas W Parsons, "Digital Signal Processing Theory, Applications and Hardware", Computer Science Press An imprint of W.H. Freeman and Company New York.
19. Douglas G.MYERS , "Digital Signal Processing /Efficient Convolution and Fourier Transform Techniques", Prentice-Hall
20. Lonnie C.Ludeman, "Fundamental of Digital Signal Processing", Harper&Row, Publishers, New York.
21. Alan V.Oppenheim, Ronald W.Schafer, "Discrete-Time Signal Processing", Prentice-Hall International, Inc.