

อะแดปทีฟอัลกอริทึมสำหรับดีเทกต์สัญญาณไซน์คลื่นเดียวโดยใช้
วงจรรีโอดีอาร์นอ้ทช์ฟิลเตอร์

Adaptive Algorithm for a Single Sinusoidal Detection using IIR Notch Filter



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต

ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
จัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อะแคปทีฟอัลกอริทึมสำหรับดีเทกต์สัญญาณชายน้คลื่นเดียวโดยใช้
วงจรวจรไอไออาร์นอทซ์ฟิลเตอร์

โดย

นาย ชัยพร ยาทิพย์

นาย นคร ไทยพิทักษ์

อาจารย์ที่ปรึกษา

ผศ. ชวลิต เบญจางคประเสริฐ

บทคัดย่อ

บทวิจัยนี้เป็นการนำเสนออะแคปทีฟอัลกอริทึมแบบใหม่เพื่อใช้สำหรับดีเทกต์สัญญาณชายน้คลื่นเดียวโดยใช้วิธีการปรับค่าคิวเฟลคเตอร์ ด้วยวงจรวจรไอไออาร์นอทซ์ฟิลเตอร์ โดยมีขั้นตอนการออกแบบดังนี้ ขั้นแรกออกแบบอะแคปทีฟอัลกอริทึมแบบมีค่าคิวเฟลคเตอร์คงที่ ขั้นที่สองออกแบบอะแคปทีฟอัลกอริทึมแบบปรับค่าคิวเฟลคเตอร์ได้ และหลังจากนั้นปรับปรุงอะแคปทีฟอัลกอริทึมที่ได้ เพื่อลดอุปสรรคของวงจรรให้น้อยลง แต่สามารถทำงานได้เร็วกว่าเดิม ซึ่งสามารถยืนยันได้โดยการซิมูเลตด้วยโปรแกรมคอมพิวเตอร์ และจากการทดลองจริงบนบอร์ด DSP TMS320C50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Adaptive Algorithm for a Single Sinusoidal Detection using IIR Notch Filter

By

Mr. Chaiyaporn Yathip

Mr. Nakorn Thaipitak

Adviser

Asst.Prof. Chawalit Benjangkprasert

ABSTRACT

This paper presents a new adaptive algorithm for a single sinusoidal detection base on an IIR Notch filter and a variable Q-factor is proposed. The procedure is as follow: First, Adaptive IIR Notch filter with a fixed Q-factor is reviewed. Secondly, a new adaptive algorithm with a variable Q-factor is realized. In order to reduce the amount of computational complexity and the processing speed up. We propose a simplified adaptive algorithm. Finally, The computer simulation results and the experiment on DSP TMS320C50 board are given to demonstrate the convergence performance.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

ในเครื่องมือที่มีการใช้ความถี่ของสัญญาณเพื่อนำไปประมวลผล เช่น เครื่องตรวจสอบการทำงานของหัวใจ เครื่องมือวัดความถี่ ความสามารถของวงจรความถี่ทั่วไปของเครื่องมือเหล่านี้มีสัญญาณรบกวนต่างๆผ่านไปได้ปริิญาณิพนธ์นี้จึงได้สร้างวงจรเพื่อใช้ควบคุมวงจรของความถี่ให้สามารถลดทอนสัญญาณรบกวนได้มากยิ่งขึ้น การออกแบบวงจรความถี่ที่ใช้อะแดปทีฟอัลกอริทึมประยุกต์เข้ากับวงจรความถี่แบบไอโออาร์นอทซ์ฟิลเตอร์ ผลของการออกแบบจะได้รับการพิสูจน์จากโปรแกรมที่เขียนมาเพื่อซิมูเลทสมการต่างๆ และเมื่อจะนำไปใช้งานจริงทำได้โดยเราสามารถเขียนโปรแกรมบนชิพ DSP เพื่อทำการทดลองได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การที่โครงการ Adaptive Algorithm for a Single Sinusoidal Detection using IIR Notch Filter สำเร็จลงด้วยความเรียบร้อยเป็นอย่างดีนั้น ทางคณะผู้จัดทำขอขอบพระคุณอย่างสูงต่อท่านทั้งหลายที่กรุณาให้ความอนุเคราะห์ ความร่วมมือ ให้คำแนะนำและคำปรึกษา และอำนวยความสะดวกโดยเฉพาะ ผศ. ชวลิต เบญจางคประเสริฐ ในการให้คำปรึกษาชี้แนะและแก้ไขปัญหาต่างๆ ในการจัดทำโครงการและในการเขียนปริญญาานิพนธ์และ ผศ. อรลภก แสงอรุณ อ.มยุรี เลิศเวชกุล อ.พิทักษ์ ธรรมวาริน ในการเอื้อเฟื้ออุปกรณ์และเครื่องมือต่างๆ ในการทำโครงการ



คณะผู้จัดทำ

ชัยพร ยาทิพย์

นคร ไทยพิทักษ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบายสัญลักษณ์

- k : จำนวนเวลาการ Sampling (จำนวนเต็ม)
- $x(k)$: สัญญาณทางด้าน Input
- $y(k)$: สัญญาณทางด้าน Output
- $n(k)$: Noise แบบเกาส์
- $H(z)$: Transfer Function ของวงจรกรองความถี่แบบ IIR อันดับสอง
- $\alpha_0(k)$: สัมประสิทธิ์ คงที่สำหรับกำหนดค่า Q-Factor ของวงจรกรองความถี่มีค่าระหว่าง $(0 < \alpha_0 < 1)$
- $\alpha_1(k)$: สัมประสิทธิ์แปรค่าใช้กำหนดความถี่กลาง ω_0 มีค่าระหว่าง $(-1 < \alpha_1 < 1)$
- ψ_{01}, ψ_{02} : สัญญาณควบคุม Adaptive ของ $\alpha_1(k)$
- μ_0, μ_1 : Step Size Parameter จะเป็นสัดส่วนโดยตรงกับค่า $y(k)$ และ $\psi(k)$ ซึ่งต่างก็มาจาก $H(z)$ เนื่องจากว่า $\alpha_1(k)$ จะติดตามความถี่เร็วหรือไม่นั้นก็ขึ้นอยู่กับค่าของ $y(k) \cdot \psi(k)$
- $\alpha_{0,opt}(k)$: ค่า $\alpha_0(k)$ ที่จะทำให้ $\alpha_1(k)$ มีค่าสูงสุด
- $G(z)$: Transfer Function ของ $\psi(k)$ ที่อยู่ในรูป Z-Transform
- $|H(\alpha_0(k))|^2$: เป็นฟังก์ชันของ α_0 ซึ่งจะเป็นสัดส่วนโดยตรงกับค่าเฉลี่ยกำลังสองของ $[y(k)]^2$
- $\Pi(\alpha_0)$: มีค่าเท่ากับ $\frac{\partial |H(\alpha_1(k))|}{\partial \alpha_1(k)}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 ทฤษฎีเบื้องต้น	
1.1 Introduction to Digital Filters	1
1.2 Application in Digital Filter Design	2
1.3 Realization Structures for Digital Filter	2
1.4 Type of Digital Filter	4
1.5 Summary of key Characteristic Features of FIR	5
1.6 Summary of the Basic Features of IIR Filter	6
1.7 Adaptive Digital Filter	7
1.8 Concepts of Adaptive Digital Filter	8
1.9 ตัวอย่างการใช้งาน Adaptive Filter ในงานต่างๆ	9
1.10 บอร์ด DSP	21
บทที่ 2 การวิเคราะห์และออกแบบ	
- ขั้นตอนการออกแบบ	27
2.1 Adaptive Algorithm โดยการใช้ฟิลเตอร์แบบค่า Q-Factor คงที่	30
- การแปลงจาก Analog Notch Filter เป็น Digital Notch Filter	30
2.2 Adaptive Algorithm โดยการใช้ฟิลเตอร์ปรับค่า Q-Factor ได้	34
2.3 Simplified Adaptive Algorithm	42
บทที่ 3 โปรแกรมและกราฟแสดงผลการทดลอง	
3.1 โปรแกรม	53
3.2 กราฟแสดงผลการทดลอง	64
บทที่ 4 การเขียนโปรแกรมบน Chip DSP	
4.1 Frequency Response of Interface Chip	78
4.2 IIR Notch Filter	84
4.3 Adaptive IIR Notch Filter	95
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง	108

บรรณานุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญโปรแกรม

หน้า

- แสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ ω	53
- แสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ α_1	54
- แสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีที่ยังไม่ได้ลัดตัวแปร	55
- แสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีที่ปรับตัวแปรแล้ว	55
- แสดงคุณสมบัติของ $ H(\alpha_0(k)) $ เทียบกับ α_0	56
- แสดงคุณสมบัติของ $ G(z) , G_{01}(z) $ และ $ G_{02}(z) $ เทียบกับ ω	57
- แสดงคุณสมบัติของ $H(e^{j\omega})$ เทียบกับ α_0	59
- แสดงคุณสมบัติของ $ \Pi(\alpha_0(k)) $ เทียบกับ α_0	60
- Adaptive Algorithm กราฟแสดงการ Converge ของ $\alpha_1(k)$	61
- Simplified Adaptive Algorithm กราฟแสดงการ Converge ของ $\alpha_1(k)$	62

สารบัญกราฟ

- กราฟแสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ ω	64
- กราฟแสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ α_1	65
- กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีที่ยังไม่ลัดตัวแปร	66
- กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีที่ปรับตัวแปรแล้ว	67
- กราฟแสดงคุณสมบัติของ $ H(\alpha_0(k)) $ เทียบกับ α_0	68
- กราฟแสดงคุณสมบัติของ $ G(z) , G_{01}(z) $ และ $ G_{02}(z) $ เทียบกับ ω	69
- กราฟแสดงคุณสมบัติของ $H(e^{j\omega})$ เทียบกับ α_0	70
- กราฟแสดงคุณสมบัติของ $ \Pi(\alpha_0(k)) $ เทียบกับ α_0	71
- กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Simplify Adaptive IIR Notch Filter (2)	72
- กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Simplify Adaptive IIR Notch Filter (2)	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Simplify Adaptive IIR Notch Filter (2) 74
- กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Adaptive IIR Band Pass Filter (2) 75
- กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Adaptive IIR Band Pass Filter (2) 76
- กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Adaptive IIR Band Pass Filter (2) 77



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

1.1 Introduction to Digital Filter

Digital Filter เป็นสิ่งที่สำคัญมากที่สุดอย่างหนึ่งในการนำไปใช้ในระบบ DSP (Digital Signal Processing) เป็นการนำ Algorithm ทางคณิตศาสตร์มาใช้ในระบบ Hardware หรือ Software กล่าวคือ สัญญาณ Digital Input จะทำการสร้างสัญญาณ Digital Output โดยใช้ Filtering Algorithm เพื่อให้ได้รับสัญญาณจากการ Filtering ตามความต้องการเพื่อนำไปใช้ใน Digital Filter สำหรับชนิดของ Filter ที่สำคัญจะถูกกำหนดโดย

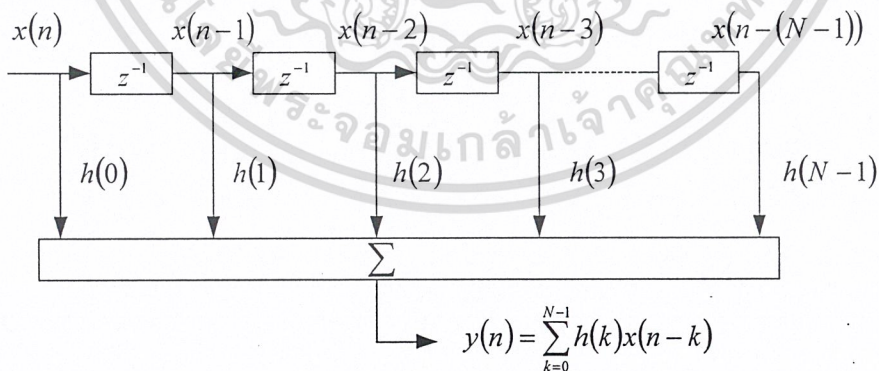
$$y(n) = \sum_{k=0}^{N-1} h(k) \cdot x(n-k) \quad 1.1$$

โดยที่ $h(k)$ คือ สัมประสิทธิ์ ของ Filter $k=0,1,\dots,n-1$

$x(n)$ คือ Input Filter

$y(n)$ คือ Output Filter

สำหรับ Filter ที่ให้มาค่าสัมประสิทธิ์จะกำหนด Characteristic ของ Filter และ Filtering นี้จะอยู่กับการ Convolution ที่แท้จริงของสัญญาณและการตอบสนองต่ออิมพัลส์ของ Filter ใน Time Domain ซึ่งก็คือ $h(k)$

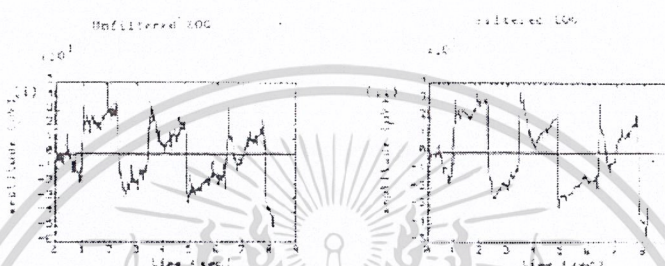


รูปที่ 1 Block Diagram ของ Transversal Filter โดยที่มี $h(k)$, ($k=0,1,\dots,N-1$) เป็นสัมประสิทธิ์ของ Filter และ z^{-1} แสดงถึง Delay ของช่วงเวลาในการสุ่มตัวอย่างหนึ่งครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะแสดงถึง Block Diagram ซึ่งจะแสดงถึงนิยามของ Digital Filter ที่กำหนดไว้ก่อนหน้านี โดยที่รูปแบบ Filter ชนิดนี้เป็นที่รู้จักอย่างแพร่หลาย เรียกว่า Transversal Filter จากรูปจะแสดงถึง Delay ของช่วงเวลาในการสุ่มตัวอย่างหนึ่งครั้ง

วัตถุประสงค์ของ Filter โดยรวมๆ แล้วเป็นการกำจัดหรือลด Noise ในระบบอื่นๆ อาจจะปะปนมาด้วยดังนั้นเราจึงทำการกรองสัญญาณที่แท้จริงออกมาเพื่อที่จะสามารถนำไปใช้ในการวิเคราะห์ได้ถูกต้องดังรูป



รูปที่ 2 แสดงถึงผลของการใช้ Digital Low Pass Filtering ในการลดสัญญาณ Distortion ที่ความถี่สูงในทางการแพทย์

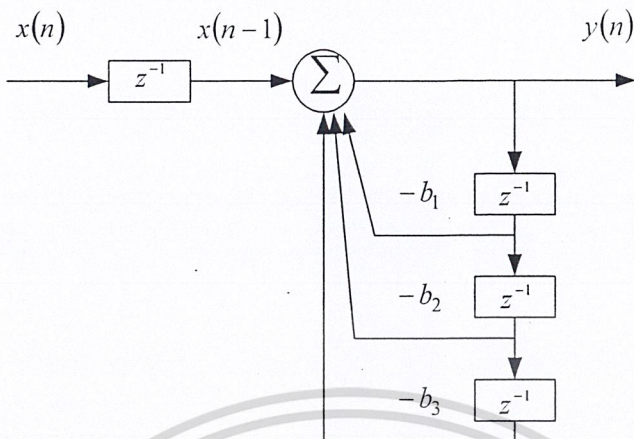
1.2 Application in Digital Filter Design

Z-Transform เป็นสิ่งที่สำคัญที่สุดอย่างหนึ่งในระบบ DSP นั่นคือใช้ในการออกแบบเพื่อวิเคราะห์สัญญาณ Error ใน Digital Filter โดยเฉพาะในระบบ IIR Filter ซึ่งมักจะถูกใช้ในการคำนวณค่าสัมประสิทธิ์ของ Digital Filter และวิเคราะห์ผลของการ Quantization Errors ต่างๆ ในโครงสร้างที่ใช้ใน Digital Filter และสิ่งที่สำคัญอย่างหนึ่งที่ใช้ Z-Transform ในการออกแบบ Discrete Filter คือใช้ในการแสดงโครงสร้างของ Digital Filter

1.3 Realization Structures for Digital Filter

Discrete Time Filter จะถูกนำมาแสดงบ่อยๆ ในรูปแบบของ Block Diagram หรือ Signal Flow Graph ซึ่ง Block Diagram จะเป็นการสะดวกกว่าที่ใช้ในการแสดง Differential Equations หรือ Equivalent ของ Transfer Function ซึ่งสามารถพิจารณาดังตัวอย่าง Discrete-Filter ด้วย Difference Equation

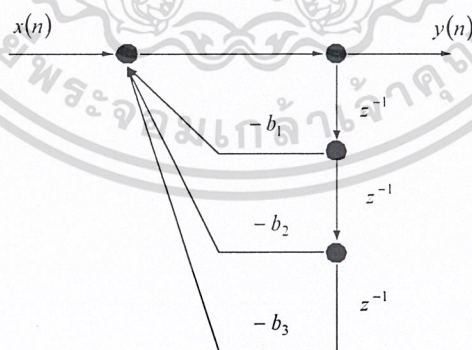
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3 การวิเคราะห์ห้สมการ Difference Equation ในรูปแบบ Block Diagram โดยสมมติสมการดังนี้

$$y(n) = x(n-1) - b_1 y(n-1) + b_2 y(n-2) + b_3 y(n-3) \tag{1.2}$$

Block Diagram แสดงถึงสมการดังกล่าวจากรูปสัญลักษณ์ จะแสดงถึง Delay ของช่วงเวลาหนึ่งหน่วย ซึ่งอาจจะพิจารณาใน Node ต่างๆ ลูกศรแสดงถึงการ Multipliers และค่าคงที่ตัวถูกทำมาจาก การคูณกับตัวประกอบ ความสัมพันธ์ระหว่าง Difference Equation และ Block Diagram จะค่อยๆ ปรากฏขึ้น และ Signal Flow Graph Diagram ก็ใช้แสดงกับ Difference Equation เช่นเดียวกัน ดังรูปที่ 4



รูปที่ 4 การวิเคราะห์ห้สมการ Difference Equation ในรูปแบบ Signal Flow Graph Diagram โดยสมมติสมการดังนี้

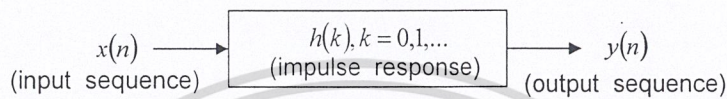
$$y(n) = x(n-1) - b_1 y(n-1) + b_2 y(n-2) + b_3 y(n-3) \tag{1.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 TYPE of Digital Filter

Digital Filter ที่ใช้อย่างกว้างขวางแบ่งออกเป็น 2 ชนิด

1. Finite Impulse Response (FIR)
2. Infinite Impulse Response (IIR)



รูปที่ 5 Block Diagram แสดงถึงหลักการของ Digital Filter

Filter แต่ละชนิด จะมีรูปแบบพื้นฐานแต่ละตัวซึ่งสามารถแสดงลำดับการตอบสนองต่ออิมพัลส์ $h(k)$, ($k = 0, 1, 2, \dots$) ดังรูปที่ 5 สัญญาณ Input และ Output ที่ต่อ Filter จะถูกนำมาเกี่ยวข้องใช้โดยผลรวมของการ Convolution ซึ่งสมการแต่ละชนิดแสดงได้ดังต่อไปนี้

$$y(n) = \sum_{k=0}^{\infty} h(k) \cdot x(n-k) : IIR \quad 1.4$$

$$y(n) = \sum_{k=0}^{N-1} h(k) \cdot x(n-k) : FIR$$

จากสมการข้างต้นสำหรับ IIR Filter นั้นผลการตอบสนองต่ออิมพัลส์จะอยู่ในช่วงที่ไม่จำกัด แต่สำหรับ FIR Filter จะอยู่ในช่วงที่กำหนด โดยเริ่มที่ a_k สำหรับ FIR จะมีเพียง N ค่า ส่วนในการคำนวณหาค่า Output ของ IIR ไม่สามารถเป็นไปได้ เนื่องจากสมการมีความยาวของผลการตอบสนองต่ออิมพัลส์นั้นยาวมาก (Infinity) ดังนั้นเราจึงแทนสมการของ IIR Filter ในรูปของ Recursive Form

$$y(n) = \sum_{k=0}^{\infty} h(k) \cdot x(n-k) = \sum_{k=0}^{\infty} a_k \cdot x(n-k) - \sum_{k=1}^{\infty} b_k \cdot x(n-k) \quad 1.5$$

โดยที่ a_k และ b_k เป็นค่าสัมประสิทธิ์ของ Filter ดังนั้นจากสมการข้างต้นคือสมการที่แตกต่างกันระหว่าง FIR Filter และ IIR Filter จากสมการเหล่านั้นจะมีค่า $h(k)$ สำหรับ FIR Filter และ a_k และ b_k สำหรับ IIR Filter ซึ่งสมการทั้งสองเป็นสิ่งที่สำคัญมากที่ใช้ในการออกแบบ Filter เป็นส่วนใหญ่ จากฟังก์ชันการสุ่มตัวอย่าง Output $y(n)$ เป็นฟังก์ชันของ Output ที่ผ่านมา ซึ่งค่าที่ถูกต้องจะเท่ากับค่าตัวอย่าง Input ที่สุ่มมา ทั้งในอดีตและปัจจุบัน ซึ่งนั่นคือ IIR Filter จะมี

เอกสการ Feedback ซึ่งจะเปรียบเทียบกับสมการ FIR Filter แล้ว การสุ่มตัวอย่าง Output $y(n)$ จะเป็นการคำนวณไม่เพียงฟังก์ชันของค่า Input ของในอดีตและปัจจุบันเท่านั้นเอง แต่อย่างไรก็ตามเมื่อ มีค่าเท่ากับศูนย์ใช้

สมการ IIR Filter ก็คือ สมการ FIR Filter นั้นเองและจากสมการด้านล่าง นี้ก็คือ Transfer Function ของ FIR และ IIR Filter ซึ่งค่า Transfer Function เหล่านี้จะมีประโยชน์มาก ในการคำนวณหาค่าใน Frequency Response

$$H(z) = \sum_{k=0}^{N-1} h(k) \cdot z^{-k} : \text{FIR}$$

$$H(z) = \frac{\sum_{k=0}^N a_k \cdot z^{-k}}{1 + \sum_{k=1}^M b_k \cdot z^{-k}} : \text{IIR}$$
1.6

1.5 Summary of key Characteristic Features of FIR Filter

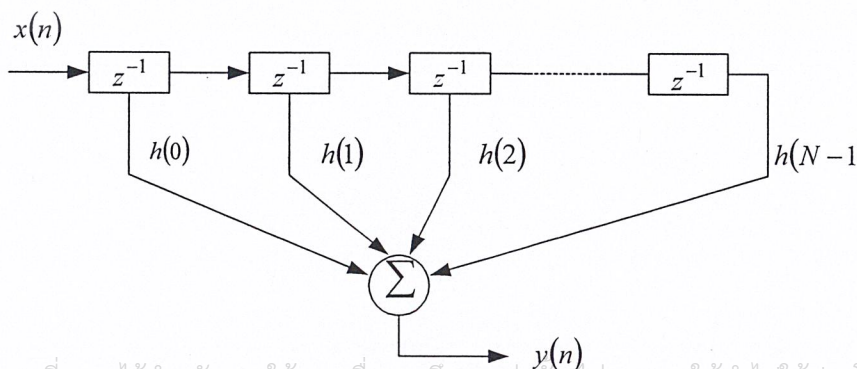
1. Characteristic ของ FIR Filter จะแสดงดังสมการ

$$y(n) = \sum_{k=0}^{N-1} h(k) \cdot x(n-k)$$

$$H(z) = \sum_{k=0}^{N-1} h(k) \cdot z^{-k}$$
1.7

โดยที่ $h(k), k=0,1,\dots,n-1$ เป็นค่าสัมประสิทธิ์ในการตอบสนองต่ออิมพัลส์ของ Filter
 $H(z)$ เป็น Transfer Function ของ Filter
 N เป็น Filter Length

นั่นก็คือจำนวนของสัมประสิทธิ์ของ Filter ของสมการแรกเป็น FIR Difference Equation ซึ่งมันจะเป็นสมการใน Time Domain และจะพิจารณา FIR Filter ในรูปของ Non-Recursive Form ซึ่งจะทำให้การคำนวณ Output $y(n)$ เป็นฟังก์ชันเพียงค่าของ Input ในอดีตและปัจจุบันเท่านั้น เมื่อ Filter ถูกนำไปใช้ในรูปแบบนี้ นั่นคือ จะเป็นการหาค่าโดยตรง จะทำให้ได้ค่าคงที่ (Stable) และ Transfer Function ของ Filter จะใช้ในการวิเคราะห์และหาค่า Filter ในทาง Frequency Response



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ไปยังสื่ออื่นและต้องขออนุญาตทุกครั้งที่มีการนำไปใช้

รูปที่ 6 โครงสร้างของ FIR Digital Filter

2. FIR Filter สามารถมีความเป็น Linear Phase Response ที่แท้จริง

3. FIR Filter ย่างที่จะนำไปใช้ในระบบ DSP Processors ทั้งหมดมีความเหมาะสมที่จะใช้กับ FIR Filtering Non-Recursive FIR Filter จะมีข้อผิดพลาดน้อยกว่าผลของความยาวที่จำกัดกว่า IIR Filter

1.6 Summary of The Basic Features of IIR Filter

Characteristic ของ IIR Digital Filter ถูกกำหนดตาม Recursive Equation ดังนี้

$$y(n) = \sum_{k=0}^{\infty} h(k) \cdot x(n-k) = \sum_{k=0}^{\infty} a_k \cdot x(n-k) - \sum_{k=1}^{\infty} b_k \cdot x(n-k)$$

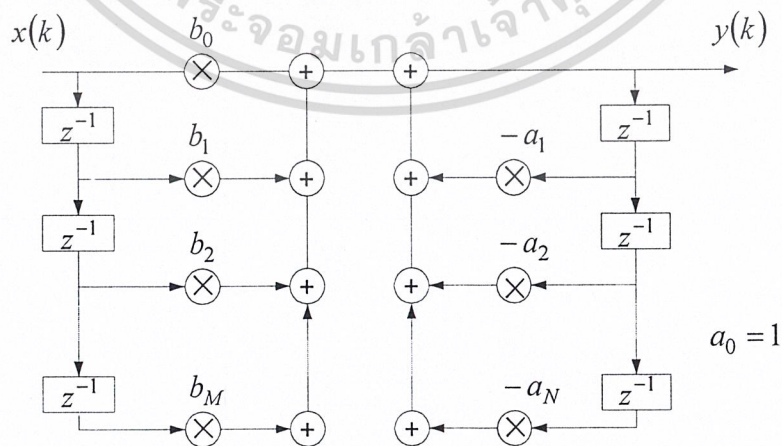
โดยที่ $h(k)$ เป็นผลการตอบสนองต่ออิมพัลส์ของ Filter ซึ่งจะมีช่วงเวลาที่ไม่มีจำกัด เป็นสัมประสิทธิ์ของ Filter

$x(n)$ และ $y(n)$ เป็น Input และ Output ของ Filter

Transfer Function สำหรับ IIR Filter ถูกกำหนดโดย

$$H(z) = \frac{a_0 + a_1 z^{-1} + \dots + a_n z^{-N}}{1 + b_1 z^{-1} + \dots + b_m z^{-M}} = \frac{\sum_{k=0}^N a_k \cdot z^{-k}}{1 + \sum_{k=1}^M b_k \cdot z^{-k}} \quad 1.8$$

ส่วนที่สำคัญในการออกแบบ IIR Filter คือการหาค่าที่เหมาะสมสำหรับค่าสัมประสิทธิ์ และให้สอดคล้องกับลักษณะของ Filter ใน Frequency Response ดังนั้นสมการดังกล่าวทั้งสอง จึงเป็น Characteristic Equation สำหรับ IIR Filter



รูปที่ 7 โครงสร้าง IIR Digital Filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ 1.5 กล่าวได้ว่าค่าการสุ่มตัวอย่าง Output $y(n)$ เป็นฟังก์ชันของ Output ที่ผ่านมา $y(n)$ ซึ่งจะเท่ากับค่าสมการสุ่มตัวอย่าง Input ทั้งในปัจจุบันและอดีต นั่นคือ IIR Filter เป็นระบบ Feedback ซึ่งปกติแล้ว IIR Filter ต้องการเพียงสัมประสิทธิ์ 2-3 ตัวมากกว่า FIR Filter สำหรับข้อกำหนดเดียวกันซึ่ง IIR Filter จะถูกใช้เมื่อจุดคัทออฟ ที่แคบและมี Throughput ที่สูงเป็นความต้องการที่สำคัญ เหตุผลที่จะทำให้ IIR Filter ไม่เสถียร

1.7 Adaptive Digital Filter

Adaptive Digital Filter เป็นสิ่งที่จำเป็นในการกรองความถี่แบบ Digital ที่มีลักษณะการปรับค่าต่างๆ ได้ด้วยตัวเอง โดยทำการปรับค่าอย่างอัตโนมัติ เพื่อที่จะเป็นค่าของสัญญาณ Input ในครั้งต่อไป

Adaptive Digital Filter เราหมายถึงอุปกรณ์ซึ่งถูกออกแบบให้สามารถปรับค่าเองได้ในความคิดที่ว่า อุปกรณ์จะบรรจุกลุ่มของ Parameters ที่ปรับค่าได้ และค่านั้นจะถูกกำหนดโดยอัตโนมัติขึ้นอยู่กับการประมาณค่าคุณสมบัติทางสถิติของสัญญาณที่เกี่ยวข้อง ฉะนั้นทฤษฎีของ Adaptive Filter จะมีความสัมพันธ์ใกล้ชิดกับการออกแบบ Filter ที่ดีที่สุดในปัญหาการออกแบบความต้องการที่จะจัดกลุ่มที่ดีที่สุดของ Filter Parameters จากความรู้ของคุณสมบัติของสัญญาณที่เกี่ยวข้อง ให้เหมาะสมกับบรรทัดฐาน และ ปัญหาของ Adaptive Filter คือ ความต้องการที่จะหา Algorithm (ขบวนการในการแก้ปัญหา) สำหรับปรับค่า Filter Parameter ในสถานการณ์ที่ความรู้เกี่ยวกับคุณสมบัติของสัญญาณที่เกี่ยวข้องไม่สมบูรณ์ ดังนั้นการทำงานของ Adaptive Filter จะเข้าสู่สู่ลักษณะ Filter ที่ดีที่สุด หลังจากผ่านขบวนการซ้ำๆ เกี่ยวกับตัวเลขจำนวนมากของ Algorithm

Adaptive Filter มีลักษณะแตกต่างกันดังนี้

1. Adaptive Filter ลักษณะ Open Loop หรือไม่กระทำซ้ำ ประกอบด้วยขบวนการสองขั้นโดยที่ขบวนการแรก คือ การเรียนรู้สถิติของสัญญาณที่เกี่ยวข้อง แล้วนำผลที่ได้ใส่เข้าไปใน Algorithm ที่ไม่มีการกระทำซ้ำ (Non-Recursive Algorithm) ลักษณะเช่นนี้จะมีข้อเสียที่ต้องการความละเอียดมากและ Hardware ราคาแพง
2. Adaptive Filter ลักษณะ Closed Loop หรือ กระทำซ้ำ ที่เรื่องของสถิติไม่ได้ถูกและมาอย่างเด่นชัด แต่ว่าการออกแบบของ Adaptive Filter จะทำได้ในขบวนการเดียวโดยการเฉลี่ยของ Algorithm ที่กระทำซ้ำ (Recursive Algorithm) ซึ่งจะเปลี่ยนค่า Filter Parameters โดยอัตโนมัติด้วยข้อมูลที่สุ่มเข้ามาใหม่ในแต่ละครั้ง ในแต่ละการกระทำซ้ำของ Adaptive Filter

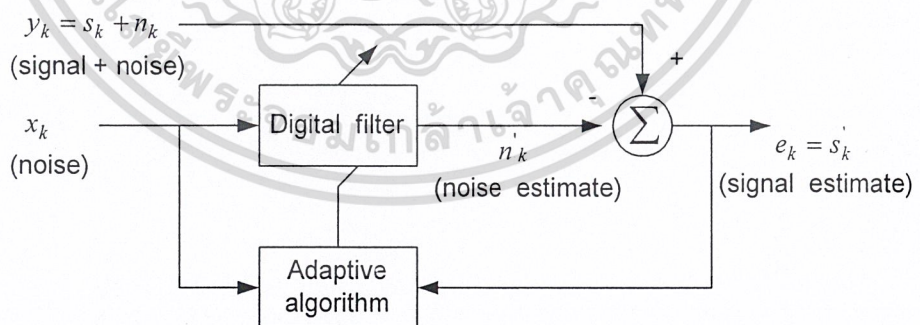
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะค่อยๆ เรียนรู้เกี่ยวกับสถิติของสัญญาณที่เกี่ยวข้องและการปรับปรุงกลุ่มของค่าปัจจุบันของ Filter ที่เปลี่ยนค่า Parameters ได้ จะถูกคำนวณโดยใช้ข่าวสารใหม่และ Adjust Table Filter (Filter ที่เปลี่ยนค่า Parameters) ค่า Parameters ของมันจะเพิ่มขึ้นและการกระทำซ้ำครั้งต่อไป จะมีฐานจากการปฏิบัติการของ Filter ด้วยกลุ่มของค่าที่ปรับปรุงแล้วในลักษณะของ Closed Loop การเรียนรู้และการปฏิบัติการคำนวณจะรวมกันเป็นขบวนการเดียวกันซึ่งดูเหมือนว่าจะมี เครื่องมือที่ง่ายกว่าลักษณะ Open Loop

Adaptive Digital Filter มีคุณสมบัติใน Frequency Response จะทำการปรับ หรือแก้ไขโดยอัตโนมัติ เพื่อที่จะปรับปรุงโครงสร้างให้สอดคล้องกับเกณฑ์บางส่วนในการที่จะให้ Filter ทำการปรับและเปลี่ยนค่าลักษณะสัญญาณ Input เนื่องจากการปรับค่าของตัวเอง และมีความง่ายในการสร้าง Adaptive Digital Filter จะพบว่าถูกนำมาใช้ในการหลายๆ อย่าง เช่น Telephone Echo Canceling, Radar Signal Processing, Navigator System Equation of Communication Channels และ Biomedical Signal Enhancement

1.8 Concepts of Adaptive Digital Filter

Adaptive Filter as a Noise Canceller ประกอบด้วยสองส่วนที่แตกต่างกัน Digital Filter กับสัมประสิทธิ์ที่ใช้ในการปรับค่า และ Adaptive Algorithm ซึ่งจะใช้ในการปรับค่าหรือปรับปรุงค่าสัมประสิทธิ์ของ Filter ดังรูป



รูปที่ 8 Adaptive Noise Canceller

สัญญาณ Input 2 สัญญาณ y_k และ x_k ถูกป้อนอย่างต่อเนื่องไปยัง Adaptive Filter สัญญาณ y_k ประกอบด้วยสัญญาณที่ต้องการ s_k และ Noise ปะปนอยู่ สมมติว่าสัญญาณทั้งสองเกี่ยวข้องซึ่งกันและกัน โดยที่ x_k เป็นสัญญาณที่ใช้ในการตรวจสอบสัญญาณที่ปะปนมา ซึ่งจะเกี่ยวข้องกับ Noise (n_k) สัญญาณ x_k จะเป็นสัญญาณที่ผ่านกระบวนการ Digital Filter เพื่อที่จะประมาณค่าของ

Noise ซึ่งให้เป็น n_k ของ n_k ดังนั้น ค่าประมาณของสัญญาณที่ต้องการจะหาได้จากการหักล้างของ Output ของ Digital Filter จากสัญญาณ y_k

$$s'_k = y_k - n_k = s_k + n_k - n'_k \quad 1.9$$

วัตถุประสงค์ที่สำคัญใน Noise Canceling ก็เพื่อจะสร้างค่าประมาณที่ดีที่สุดของ Noise ใน s'_k สัญญาณที่ปะปนกัน และค่าประมาณที่ดีที่สุดของสัญญาณซึ่งจะทำได้โดยการใช้ s'_k ในระบบการ Feedback เพื่อทำการปรับค่าสัมประสิทธิ์ของ Digital Filter และ Adaptive Algorithm ที่เหมาะสมเพื่อลด Noise ให้มากที่สุด ใน s'_k สัญญาณ Output s_k จะนำไปใช้สองส่วนคือ

1. ใช้ในการประมาณค่าสัญญาณที่ต้องการ
2. สัญญาณที่ผิดพลาดจะถูกใช้ในการปรับค่าสัมประสิทธิ์ของ Filter

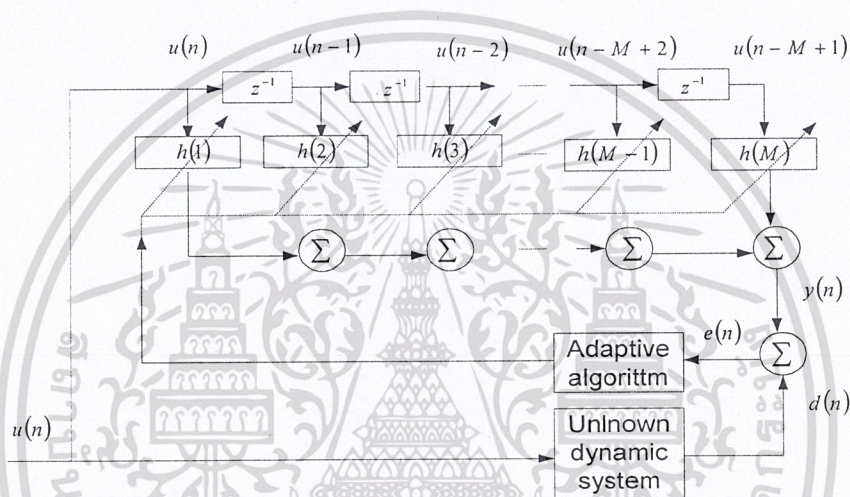
1.9 ตัวอย่างการใช้งาน Adaptive Filter ในงานต่างๆ

1. Modeling of an Unknown Dynamic System

สมมติว่ามีระบบการเปลี่ยนแปลงที่ไม่เป็นที่รู้จัก และมีกลุ่มของเครื่องมือวัด Discrete Time ใช้พิจารณาความหลากหลายของสัญญาณ Output ของระบบที่ตอบสนองกับสัญญาณคงที่ป้อนให้กับ Input ของระบบ สมมติว่าระบบนี้ไม่เปลี่ยนแปลงไปตามเวลาและมีคุณสมบัติเชิงเส้น และต้องการที่จะพัฒนา Model (แบบจำลอง) สำหรับระบบนี้ในรูปแบบของ Tapped-Delay-Line Filter ซึ่งประกอบด้วยกลุ่มของอุปกรณ์ Delay-Line แต่ละอันจะแสดงด้วยตัวปฏิบัติการ Unit-Delay กลุ่มของค่าสัมประสิทธิ์ที่ปรับค่าได้ที่มีการต่อภายในดังรูปที่ 9 ที่เวลา n สัญญาณที่มีอยู่ประกอบด้วยกลุ่มของ $u(n), u(n-1), \dots, u(n-M+1)$ ตัวอย่างเหล่านี้ถูกคูณด้วยกลุ่มที่ตรงกันของค่าสัมประสิทธิ์ที่ปรับค่าได้ มีชื่อว่า $h(1), h(2), \dots, h(M)$ เพื่อที่จะสร้างสัญญาณ Output $y(n)$ ให้ Output แท้จริงของระบบที่ไม่เป็นที่รู้จักมีสัญลักษณ์ $d(n)$ Output ของ Adaptive Filter ถูกเปรียบเทียบกับ Output ของระบบที่ไม่เป็นที่รู้จัก $d(n)$ เพื่อสร้างสัญญาณผิดพลาด ซึ่งเป็นความแตกต่างระหว่างทั้งสอง

ถึงตอนนี้ อาจจะบอกได้ว่าสถานะการทำงานของ Adaptive Filter ได้ดังนี้ ด้วย Output ของระบบที่ไม่เป็นที่รู้จัก $d(n)$ จะเรียกว่าการตอบสนองที่ได้ทำการออกแบบ (desired response) ต้องการที่จะพัฒนาขบวนการ Adaptive สำหรับการปรับค่าสัมประสิทธิ์ของ Tapped-Delay-Line Filter เพื่อที่จะทำให้สัญญาณผิดพลาด $e(n)$ มีน้อยที่สุดในลักษณะนี้ บรรทัดฐานซึ่งถูกใช้บ่อยๆในทางปฏิบัติสำหรับการทำให้น้อยที่สุดคือบรรทัดฐานค่าผิดพลาดกำลังสองเฉลี่ย ซึ่งเหมาะสมกับ Filter ที่ปรับค่าสัมประสิทธิ์ได้ เพื่อให้ได้ค่าเฉลี่ยกำลังสองน้อยที่สุดของสัญญาณผิดพลาด ในบาง

กรณีเมื่อสัญญาณ Input และ Output ของระบบที่ไม่เป็นที่รู้จักนั้นคงที่ ซึ่งเป็นผลมาจากการสมมติให้ระบบไม่เปลี่ยนแปลงไปตามเวลา หรือ Time-Invariant จะทำให้สัญญาณผิดพลาดนั้นคงที่ด้วย และค่าเฉลี่ยกำลังสองของ $e(n)$ เป็นฟังก์ชันอันดับสองของสัมประสิทธิ์ Filter ค่าผิดพลาดเฉลี่ยกำลังสองที่ขึ้นอยู่กับสัมประสิทธิ์ Filter ที่ไม่เป็นที่รู้จัก นั้นจะอยู่ในรูปของ Multidimensional Parabolic ที่มีจุดต่ำสุดที่ถูกกำหนดไว้จุดเดียว หรือ Minimum Point (จุดที่มีค่าต่ำสุด) Adaptive Filter จะทำงานในการค้นหาจุดที่ต่ำที่สุดของพื้นที่นี้อย่างต่อเนื่อง จึงต้องทำให้การทำงานของอุปกรณ์ที่ดีที่สุดเท่าที่เป็นไปได้



รูปที่ 9 แบบจำลองของระบบที่ไม่เป็นที่รู้จักโดยใช้ Tapped-Delay-Line Filter

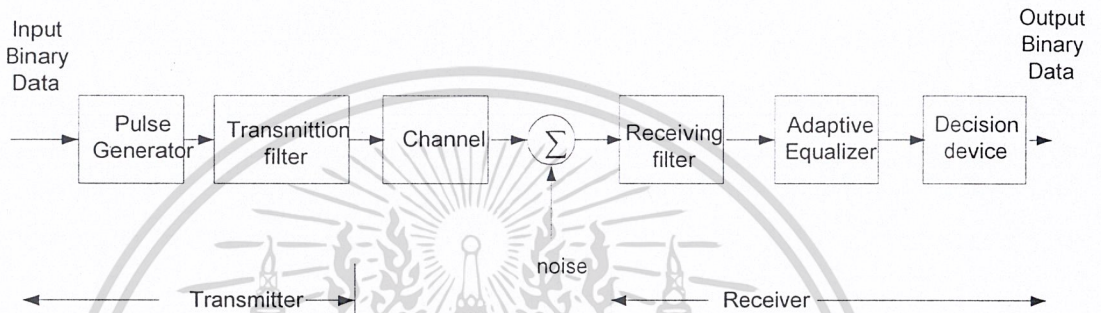
เมื่อระบบที่ไม่เป็นที่รู้จักนี้เปลี่ยนแปลงไปตามเวลา เป็นผลให้ Output ของระบบรวมถึงการตอบสนองที่ต้องการที่ได้จาก Adaptive Tapped-Delay-Line Filter จะเป็นแบบไม่คงที่ ทำนองเดียวกัน การเปลี่ยนแปลงพื้นที่ของประสิทธิภาพการผิดพลาดเปลี่ยนแปลงไปตามเวลาด้วย ในกรณีนี้ Adaptive Algorithm ใช้ปรับค่าสัมประสิทธิ์ของ Tapped-Delay-Line Filter จะถูกเพิ่มค่าสำหรับติดตามจุดต่ำสุดของประสิทธิภาพการผิดพลาดอย่างต่อเนื่อง

2. Adaptive Equalization for Data Transmission

ในการศึกษาระบบการส่งข้อมูลซึ่งใช้ประโยชน์จากช่องสัญญาณที่มีอยู่ให้มีประสิทธิภาพ จุดมุ่งหมายคือ ออกแบบระบบสำหรับการส่งข้อมูลด้วยอัตราสูงที่สุดเท่าที่เป็นไปได้ ภายใต้ความน่าเชื่อถือที่กำหนดซึ่งปกติถูกวัดอยู่ในรูปของ อัตราผิดพลาด หรือความผิดพลาดความน่าจะเป็นเฉลี่ยของสัญลักษณ์ (Average probability of symbol error) การส่งข้อมูล Digital ผ่านช่องทางการสื่อสารเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า สื่อสารแบบเชิงเส้นถูกจำกัดด้วยองค์ประกอบสองอย่าง คือ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Intersymbol Interference (ISI) เกิดจากการแพร่กระจายของรูป Pulse ที่ถูกส่ง ซึ่งเป็นผลมาจากการเบี่ยงเบนของการตอบสนองความถี่ในช่องทาง จากลักษณะทางความคิดที่ความสูงคงที่และมี Phase เชิงเส้น เช่น constant delay
2. Addition Thermal Noise กำเนิดจากภาคแรกสุดของกลุ่มรับสัญญาณ สำหรับช่องทางที่จำกัด Bandwidth เช่น ช่องทางโทรศัพท์ปกติจะพบว่า Intersymbol Interference เป็นองค์ประกอบสำคัญที่ใช้พิจารณาในการออกแบบระบบอัตราส่งข้อมูลสูง



รูปที่ 10 Block diagram of a baseband data transmission system

รูปแสดงแบบจำลองเทียบเท่าของระบบข้อมูลสื่อสาร รูปแบบของลำดับข้อมูล $\{a(n)\}$ ไม่เพียงขึ้นข้อมูล Binary ที่จะส่งออกไปเท่านั้น แต่ยังขึ้นอยู่กับชนิดของการ Modulate ที่ใช้ด้วย ดังตัวอย่างในกรณีของ M-array Phase Shift Keying $a(n)$ ได้มาจากค่า M จาก $\exp[j2\pi(m-1)/M]$ เมื่อ $m=1,2,\dots,M$ โดยปกติ M จะเท่ากับจำนวนเต็มยกกำลังสอง เมื่อไม่มี Noise ครอบคลุม Output ของ Filter เครื่องรับในรูปเท่ากับ

$$\tilde{u}(t) = \sum_n a_n \tilde{p}(t-nT) \tag{1.10}$$

เมื่อ T เป็นช่วงเวลาของคาบสัญญาณและ $\tilde{p}(t)$ เป็นการตอบสนอง Impulse ของจุดต่อลดหลักของ Filter ภาคส่ง ของสัญญาณ และ Filter ภาครับ โดยการสุ่มตัวอย่าง $u(t)$ ให้เข้ากันกับภาคส่งและนิยาม $\tilde{u}(n) = \tilde{u}(nT)$ และ $\tilde{p}(n) = \tilde{p}(nT)$ ด้านขวามือของสมการ กำหนดสัญลักษณ์ที่ต้องการ และอนุกรมที่เหลือแสดงถึง Intersymbol Interference ที่เป็นผลมาจากการกระทำรวมของ Filter ภาคส่งช่องสัญญาณ

$$\begin{aligned} u(k) &= \sum_n a_n p(k-n) \\ &= a_k p(0) + \sum_{n \neq k} a_n p(k-n) \end{aligned}$$

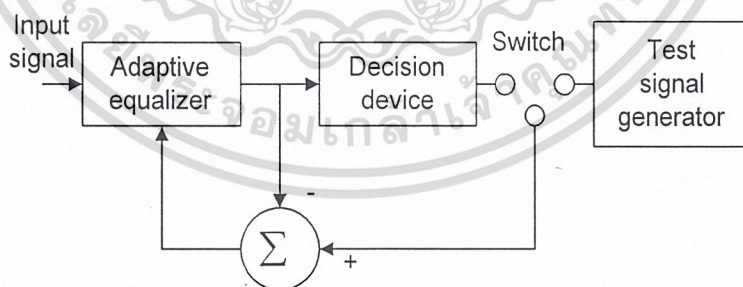
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Filter ภาครับ Intersymbol Interference นี้ ถ้าปล่อยไว้โดยไม่ตรวจสอบ อาจจะเป็นผล สำหรับการตัดสินใจหาข้อผิดพลาดเมื่อสัญญาณที่ถูกส่งที่ Filter ภาครับถูกเปรียบเทียบกับขอบเขต ระดับย่อยที่กำหนดได้บางตัวโดยได้มาจากอุปกรณ์ตัดสินใจ

เพื่อจะแก้ปัญหา Intersymbol Interference ต้องมีการควบคุมฟังก์ชันของเวลา $\tilde{p}(t)$ เฉพาะ เรื่องนี้ ถ้าลักษณะของช่องส่งสัญญาณเป็นที่รู้อย่างแน่นอนแล้ว จะเสมือนว่ามีทางสำหรับออกแบบ ทั้ง Filter ภาครับและภาครับ ซึ่งจะทำให้ผลของ Intersymbol Interference เวลาส่งก่อนข้างน้อย และที่เวลาเดียวกันจะจำกัดผลของ Noise ภาครับที่เพิ่มเข้าไป โดยลดความผิดพลาดความเป็นไปได้ เหลือของสัญญาณให้น้อยที่สุด อย่างไรก็ตามในทางปฏิบัติ เราพบว่าช่องสื่อสารนั้นเป็นแบบสุ่ม ในกลุ่มของช่องสัญญาณที่เป็นไปได้

ฉะนั้นการใช้คู่ตายตัวของ Filter ภาครับและภาครับที่ถูกออกแบบโดยคุณลักษณะช่อง สัญญาณเฉลี่ย อาจจะไม่เพียงพอที่จะลด Intersymbol Interference จึงเป็นสิ่งจำเป็นที่ต้องมี Adaptive Equalizer ซึ่งจะทำให้ความสามารถของการส่งช่องสัญญาณให้เต็มที่ให้เป็นจริงได้ คำว่า Equalizer ถูกใช้อธิบาย Filter ที่ใช้บนช่องสัญญาณโทรศัพท์เพื่อทำให้คุณลักษณะความสูงและการ หน่วงเวลาของช่องส่งสัญญาณราบเรียบ

โดยพื้นฐานสำหรับ Equalization (การทำให้เท่ากัน) ของระบบการส่งข้อมูลที่มีอยู่ คือ Pre-Equalization ที่ภาครับและ Post-Equalization ที่ภาครับ และเนื่องจากวิธีอย่างแรกต้องใช้ทางย้อน กลับ (Feedback Path) เราจึงจะพิจารณา Equalization ที่ภาครับ Adaptive Equalizer วางหลังจาก Filter ภาครับดังรูปที่ 11



รูปที่ 11 Adaptive Filter with a Decision-Directed Mode of Operation

จากรูปแสดง Block diagram ของ Adaptive Equalizer การทำงานที่เกี่ยวข้องกับ Training Mode แล้วถูกตามโดย Training Mode ในช่วง Training Mode สัญญาณทดสอบที่ทราบค่าจะถูกส่ง เพื่อวัดช่องสัญญาณสัญญาณทดสอบที่ใช้กันอย่างกว้างขวางประกอบด้วย Maximal-Length Shift Register หรือ Pseudo-Noise (PN) Sequence พร้อมกับ spectrum ที่กว้างและลงตัว สัญญาณทดสอบ

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของสำนักงานส่งเสริมการค้าในต่างประเทศ ณ นครเชียงใหม่ โดยสำนักงานส่งเสริมการค้าในต่างประเทศ ณ นครเชียงใหม่ ขอสงวนสิทธิ์ในเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะต้องมีความยาวอย่างน้อยเท่ากับตัว Equalizer เพื่อให้แน่ใจว่า Spectrum ของสัญญาณที่ส่งมีความหนาแน่นเพียงพอใน Bandwidth ของช่องส่งสัญญาณในเครื่องรับ Adaptive Equalizer จะถูกจ่ายด้วยการตอบสนองที่ต้องการ Output ของ Equalizer ถูกลบออกจากการตอบสนองนี้เพื่อสร้างสัญญาณผิดพลาด ซึ่งใช้ในการปรับสัมประสิทธิ์ของ Equalizer ให้ไปสู่ค่าที่ดีที่สุด ชนิดของ Adaptive Algorithm ที่เป็นที่ยอมรับมากที่สุดที่ใช้สำหรับการปรับนี้เกี่ยวข้องกับการพัฒนาให้ดีขึ้นของสัมประสิทธิ์แต่ละตัวของ Equalizer ในแต่ละตัวช่วงของ Symbol โดยเริ่มจากค่าเริ่มต้นที่ตั้งไว้

เมื่อช่วง Training เริ่มต้นสำเร็จลง ค่าสัมประสิทธิ์ของ Adaptive Equalizer อาจจะถูกเปลี่ยนแปลงต่อไปอีกในลักษณะของ Decision-Directed Mode (Modeของการตัดสินใจโดยตรง) ใน Mode นี้ สัญญาณผิดพลาดจะได้อาจจากการประมาณของภาครับช่วงสุดท้าย (ซึ่งไม่จำเป็นต้องเป็นสัญญาณที่ถูกต้อง) ของลำดับการส่ง การประมาณของเครื่องรับได้มาจากการให้ Output ของ Adaptive equalizer ไปยังอุปกรณ์ตัดสินใจ (Decision Device) ดังแสดงในรูป ในการปฏิบัติการปกติ อุปกรณ์ตัดสินใจจะมีความถูกต้องที่มีความเป็นไปได้สูง ดังนั้นการประมาณของสัญญาณผิดพลาดจะถูกต้องบ่อยครั้ง เพียงพอที่จะยอมให้ Adaptive equalizer สามารถปรับค่าสัมประสิทธิ์ของมันได้อย่างถูกต้อง ความสามารถที่น่าสนใจอื่นๆของ Decision Direct Adaptive Equalizer คือจริงๆแล้วมันสามารถติดตามการเปลี่ยนแปลงอย่างช้าในคุณลักษณะหรือการกระทำซ้ำๆของช่องส่งสัญญาณในภาครับด้านหน้า อย่างเช่น การสั่นซ้ำๆของ Phase

3. Digital Representation of Speech

ในช่วง 20 ปีก่อนมีการใช้ขบวนการทาง Digital มากขึ้นสำหรับการเข้ารหัสและการส่งสัญญาณเสียงอย่างมีประสิทธิภาพ องค์ประกอบหลักสองอย่างสำหรับงานนี้คือ อย่างแรกขบวนการ Digital มีความเป็นไปได้ที่จะควบคุมผลของ Noise และการรบกวนข้างเคียง ในการสื่อสาร จึงเป็นการปรับปรุงความน่าเชื่อถือของระบบอย่างสำคัญ อย่างที่สอง การปฏิวัติทางเทคโนโลยี Digital การสร้าง Very-Large-Scale Integration VLSI ได้ลดทั้งขนาดและราคาของอุปกรณ์อย่างต่อเนื่อง

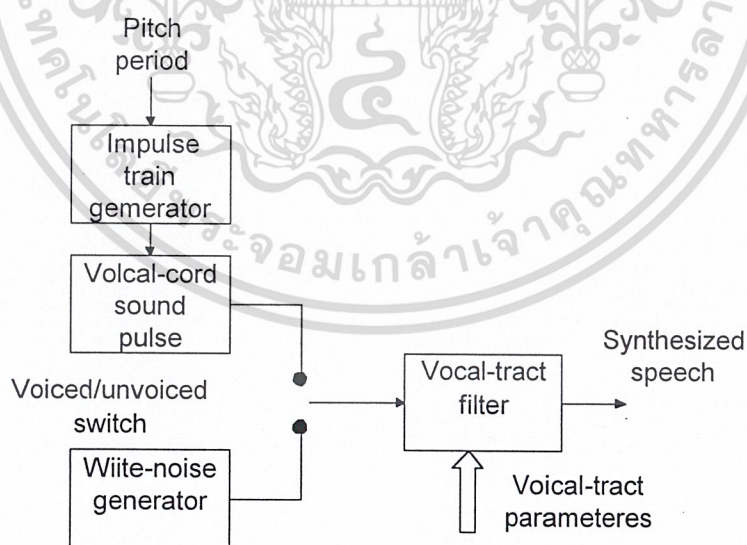
Coders (ส่วนเข้ารหัส) สำหรับการแสดงสัญญาณเสียงแบบ Digital แบ่งออกเป็น 2 ชนิดใหญ่ คือ Source Coders (ส่วนเข้ารหัสของแหล่งจ่าย) และ Waveform Coders (ส่วนเข้ารหัสของรูปคลื่น) Source Coders นั้นขึ้นอยู่กับแบบจำลอง ซึ่งพวกนี้ใช้ความรู้ก่อนสำหรับสร้างสัญญาณเสียงที่แหล่งจ่าย Source Coders สำหรับการพูดมักจะเรียกกันว่า Vocoder (คำย่อของvoice code) พวกนี้สามารถปฏิบัติการที่ 4.8 Kbit/s หรือต่ำกว่า อย่างไรก็ตาม พวกมันให้คุณภาพที่ประกอบขึ้นด้วยสัญญาณเสียงที่ขาดความเป็นธรรมชาติ ในอีกทางหนึ่ง Waveform Coders ต้องใช้สัญญาณรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งหากมีการนำไปใช้

คลื่นเสียงที่ส่งมาสร้างขึ้นมาใหม่ ในจุดนี้ Coders ตัวนี้จึงไม่ขึ้นอยู่กับสัญญาณ และอาจจะถูกออกแบบเพื่อสร้างคุณภาพโทรศัพท์สำหรับการพูดที่อัตราการเข้ารหัสต่ำๆ เท่ากับ 16 Kbit/s

แบบจำลองของกระบวนการสร้างเสียง รูปที่ 12 แสดง Block Diagram อย่างง่ายของแบบจำลองเก่า สำหรับการสร้างเสียง ซึ่งสมมติให้ขบวนการกำเนิดเสียงเป็นแบบเชิงเส้น เช่น แหล่งของการกระตุ้น และแยกออกจาก Intelligence-Modulating Vocal-Tract Filter ได้ รูปแบบที่แน่นอนของการกระตุ้นขึ้นอยู่กับว่าเสียงพูดเป็นเสียงคนพูดหรือไม่ดังที่อธิบายดังนี้

1. Voiced Speech Sound ถูกกำเนิดจากเสียงของหลอดเสียงที่ลากต่อกัน เครื่องกำเนิดขบวนการของ Impulse สร้างอันดับของ Impulses (เป็น pulse ที่สั้นมากๆ) ซึ่งเว้นว่างด้วยช่วงพื้นฐาน เท่ากับช่วงของเสียงสูง ในสัญญาณนี้จะกระตุ้น Filter เชิงเส้นที่มีการตอบสนอง Impulse เท่ากับ Pulse ของเสียงจากหลอดเสียง
2. Unvoiced Speech Sound ถูกกำเนิดจากเสียงแบบสุ่มจากอากาศที่ไหลและสั้นอย่างรวดเร็ว ในกรณีนี้ตัวกระตุ้นประกอบด้วยง่ายๆเพียง White Noise Source ความเป็นไปได้ในการกระจายของตัวอย่าง Noise นี้ไม่เป็นจุดสำคัญเลย การตอบสนองความถี่ของ Vocal-Tract Filter สำหรับ Unvoiced Speech หรือ กลุ่มของเสียงนั้นถูกคูณด้วย Spectrum ของ Vocal-Cord Sound Pulses (Pulses ของเสียงจากหลอดเสียง) และได้เป็น Short-Time Spectral Envelope ของสัญญาณเสียง



รูปที่ 12 Block Diagram ของแบบจำลองอย่างง่ายสำหรับขบวนการผลิตเสียงพูด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้ารหัสคาคหมายแบบเชิงเส้น (Linear Predictive Coding) ขบวนการของ Linear Predictive Coding; LPC เป็นตัวอย่างของ Source Coding (การเข้ารหัสแหล่งจ่าย) ขบวนการนี้สำคัญมากเพราะว่ามันไม่เพียงแต่ให้เทคนิคที่มีอำนาจมากสำหรับการส่งเสียงแบบ Digital ที่อัตราบิตต่ำแล้ว มันยังประมาณค่า Parameters พื้นฐานของการพูดอย่างถูกต้องด้วย

การพัฒนาของ LPC ขึ้นอยู่กับแบบจำลองของรูป สำหรับขบวนการผลิตเสียงพูดการตอบสนองความถี่ของ Vocal Tract (กลุ่มของเสียง) สำหรับ Unvoiced Speech หรือ Vocal Tract ที่ถูกคูณโดย Spectrum ของ Vocal-Cord Sound Pulse สำหรับเสียงพูด สามารถเขียนเป็น Transfer Function

$$H(z) = \frac{G}{1 + \sum_{k=1}^M a(k) \cdot z^{-k}} \quad 1.11$$

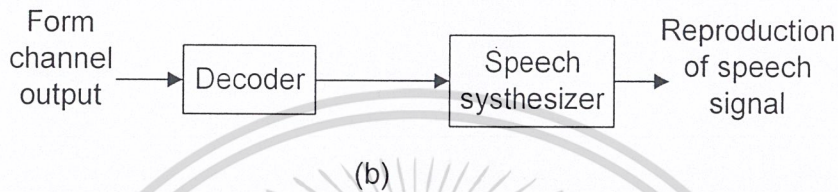
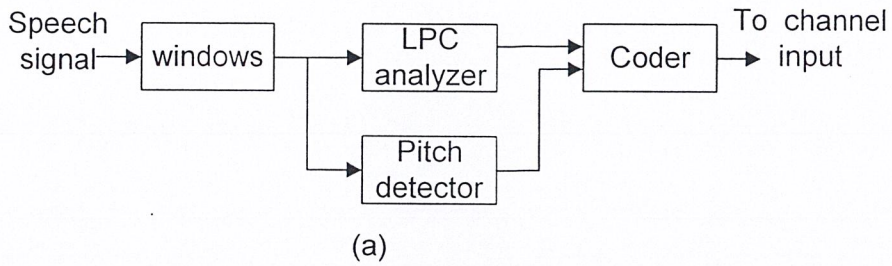
เมื่อ G เป็น Gain parameter และ z^{-1} เป็น Unit-Delay Operator รูปแบบของตัวกระตุ้นที่ให้กับ Filter นี้ถูกเปลี่ยนโดย Switching ระหว่าง Voiced และ Unvoiced Sounds ฉะนั้น Filter ที่มี Transfer Function $H(z)$ ถูกกระตุ้นด้วยอันดับของ Impulses เพื่อสร้างเสียงหรือ White-Noise Sequence เพื่อสร้าง Unvoiced Sounds ใน LPC Linear Prediction (การคาคหมายแบบเชิงเส้น) ถูกใช้เพื่อประมาณค่า Parameters ของเสียง ให้กลุ่มของตัวอย่างในอดีตของสัญญาณเสียงให้ชื่อว่า $u(n-1), u(n-2), \dots, u(n-M)$ การคาคหมายเชิงเส้นของ $u(n)$ ที่เป็นค่าตัวอย่างปัจจุบัน ถูกกำหนดโดย

$$\tilde{u}(n, n-1, \dots, n-M) = \sum_{k=1}^M h(k) \cdot u(n-k) \quad 1.12$$

สัมประสิทธิ์ของตัวคาคหมาย $h(1), h(2), \dots, h(M)$ ถูกทำให้ดีที่สุดโดยทำให้ค่าเฉลี่ยกำลังสองของความผิดพลาดในการคาคหมาย $e(n)$ มีค่าต่ำที่สุด โดยที่ $e(n)$ กำหนดจากความแตกต่างระหว่าง $u(n)$ และ $\tilde{u}(n, n-1, \dots, n-M)$ การใช้บรรทัดฐานความผิดพลาดเฉลี่ยกำลังสองต่ำที่สุดสำหรับการทำให้ส่วนคาคหมายดีที่สุด อาจจะต้องถูกปรับเพื่อให้เข้ากับเหตุผลพื้นฐานสองประการ

1. ถ้าสัญญาณเสียงพูดสามารถใช้ได้กับแบบจำลองที่อธิบายไว้ในสมการ และถ้าค่าเฉลี่ยกำลังสองของสัญญาณผิดพลาด $e(n)$ ถูกทำให้ต่ำที่สุดแล้ว เราพบว่า $e(n)$ เท่ากับตัวกระตุ้น $x(n)$ คูณกับ Gain Parameter G ในแบบจำลองในรูปที่ 13 และ $a(k) = -h(k), k=1, 2, \dots, M$ ฉะนั้นสัญญาณผิดพลาด $e(n)$ ประกอบด้วยขบวนการของ Impulses ในกรณีของ Unvoiced Sound ในกรณีทั้งคู่ สัญญาณผิดพลาด $e(n)$ อาจจะมีขนาดเล็กมากตลอดเวลา
2. การใช้บรรทัดฐานผิดพลาดเฉลี่ยกำลังสองต่ำสุดจะนำไปสู่คณิตศาสตร์ที่หาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

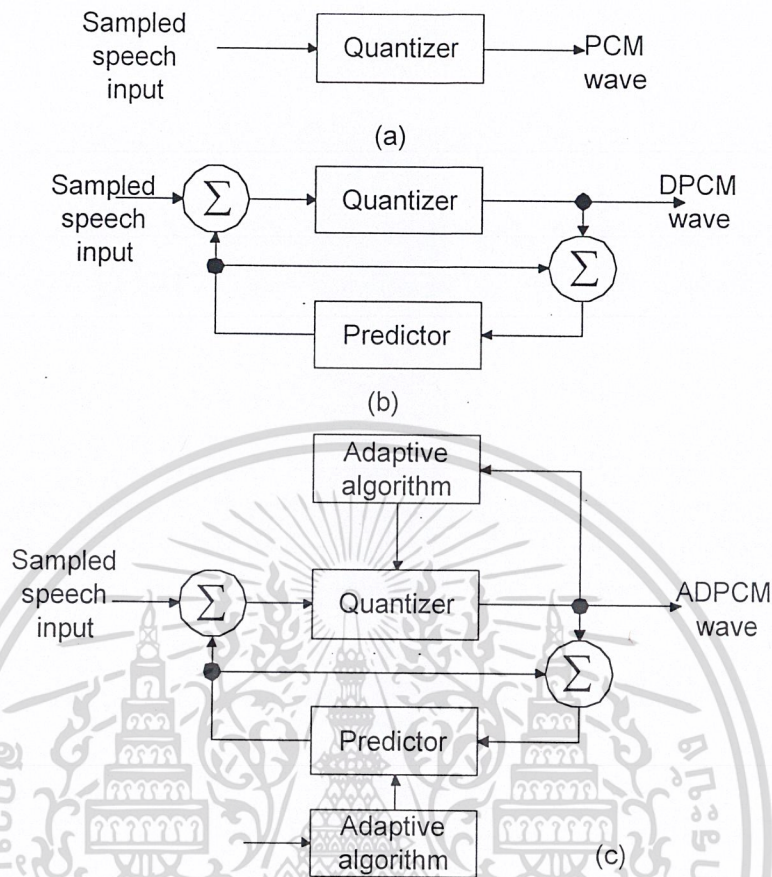


รูปที่ 13 Block Diagram of LPC Vocoder (a) Transmitter (b) Receiver

รูปแสดง Block diagram ของ LPC Vocoder ประกอบด้วยเครื่องส่งและเครื่องรับเครื่องส่งแรกสุดจะส่ง window (โดยปกติมีขนาด 10-30 ms) เข้ากับสัญญาณเสียง Input และกลายเป็น Block ของตัวอย่างเสียงพูดสำหรับเข้าขบวนการ window นี้จะสั้นพอที่ทำให้รูปร่างกลุ่มเสียงเกือบจะคงที่ ดังนั้น Parameter ของการสร้างเสียงพูดในแบบจำลอง อาจจะถือว่าเป็นค่าคงที่ที่สำคัญสำหรับช่วงเวลาของ window จากนั้นเครื่องส่งจะวิเคราะห์สัญญาณเสียงพูด Input ในลักษณะของ Adaptive block ต่อ block โดยการกระทำการคาดหมายอย่างเชิงเส้นและการตรวจจับความสูงของเสียง ท้ายสุดมันจะสร้างรหัส Parameters: (1) กลุ่มของสัมประสิทธิ์ของส่วนคาดหมาย (2) ช่วงความสูงของเสียง (3) Gain Parameter และ (4) Voiced/Unvoiced Parameter สำหรับการส่งไปทั่วช่องส่งสัญญาณ เครื่องรับจะทำปฏิบัติการกลับกัน โดยแรกสุดถอดรหัส Parameter ที่เข้ามา ในจุดนี้มันจะคำนวณค่าของสัมประสิทธิ์ส่วนคาดหมาย ช่วงความสูงของเสียงพูด และ Gain Parameter และพิจารณาว่าส่วนที่สนใจเป็นเสียงแบบ Voiced หรือ Unvoiced สุดท้าย เครื่องรับจะใช้ค่า Parameters เหล่านี้ในการสังเคราะห์สัญญาณเสียงพูดโดยใช้แบบจำลอง

การเข้ารหัสรูปคลื่น (Waveform Coding) การเข้ารหัสรูปคลื่นที่กระทำบนสัญญาณเสียงถูกออกแบบเพื่อรักษารูปร่างของสัญญาณ โดยเฉพาะการปฏิบัติการนี้รวมถึง Sampling (การสุ่ม Time discrete) และ Quantization (การประกอบ Amplitude Discrete) อัตราส่วนของการสุ่มจะเป็นตามคุณสมบัติพื้นฐานของสัญญาณเสียงพูดทั้งหมด ที่มีชื่อว่า bandlimited นี้หมายถึงว่าสัญญาณเสียงพูดสามารถสุ่มในเวลาอัตราจำกัด ที่เหมาะสมกับทฤษฎีการสุ่ม ตัวอย่าง เช่น เครื่องข่ายธุรกิจ โทรศัพท์ที่ถูกออกแบบสำหรับส่งสัญญาณเสียงพูดที่ใช้ Bandwidth(300-3,200Hz)เพื่อจะเป็นไปตาม

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี การนำเอกสารนี้ไปใช้โดยไม่ผ่านการอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้โดยไม่ผ่านการอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้โดยไม่ผ่านการอนุญาตถือว่าผิดกฎหมาย



รูปที่ 14 ตัวเข้ารหัสรูปคลื่น (a) PCM (b) DPCM (c) ADPCM

ทฤษฎี อัตราการสุ่มที่คงไว้ 8 kHz จะถูกใช้โดยทั่วไปในทางปฏิบัติ Quantization ถูกปรับบนพื้นฐานเหล่านี้ และแม้ว่าสัญญาณเสียงพูดจะมีช่วงต่อเนื่องของความสูง (และแน่นอนตัวอย่างของเสียงนี้ก็มีช่วงความสูงที่ต่อเนื่อง) ไม่ว่าอย่างไร มันก็ไม่จำเป็นที่จะส่งความสูงจริงๆของตัวอย่าง โดยพื้นฐานแล้ว หูของคนเป็นเครื่องรับท้ายสุด สามารถตรวจความแตกต่างของความสูงได้จำกัดเท่านั้น

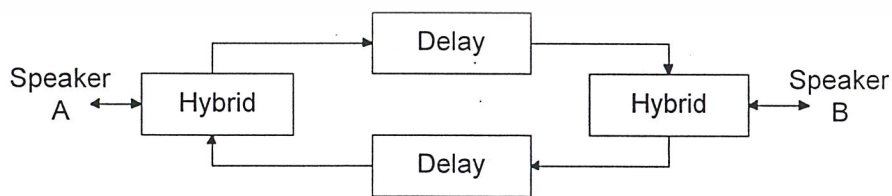
ตัวอย่างของการเข้ารหัสรูปคลื่นรวมถึง Pulse-Code Modulation; PCM และ Differential Pulse-Code Modulation DPCM ใน PCM ดังที่ถือใช้ในการโทรศัพท์ สัญญาณเสียงพูด หลังจากผ่าน Low Pass Filter ไปแล้ว จะถูกสุ่มที่อัตรา 8 kHz เป็นแบบ Quantized ไม่เชิงเส้น และจากนั้นเข้ารหัสเป็น 8-bit words ดังรูป เป็นผลที่ได้จะเป็น Signal to Quantization Noise Ratio ที่ดีในช่วงเปลี่ยนแปลงของระดับสัญญาณ Input DPCM จะเกี่ยวกับการใช้ส่วนคาคงหมายดังในรูป (b) ส่วนคาคงหมายถูกออกแบบเพื่อใช้ประโยชน์จากความสัมพันธ์ซึ่งมีอยู่ระหว่างตัวอย่างข้างเคียงของสัญญาณเสียง เพื่อที่จะทำการลดจำนวนของบิตที่ถูกใช้สำหรับการส่งแต่ละตัวอย่างของสัญญาณเสียงเป็นจริง และยังคงคุณภาพของคุณสมบัติตามที่กำหนดไว้ สิ่งนี้สามารถทำได้โดย Quantizing

และจากนั้นเข้ารหัสความผิดพลาดของการคาดหมาย ซึ่งเป็นผลจากการลบกันจาก Output ของส่วนคาดหมายจากสัญญาณ Input ถ้าส่วนคาดหมายถูกทำให้ดีที่สุดแล้วการเปลี่ยนแปลงของความผิดพลาดของส่วนคาดหมายจะน้อยกว่าของสัญญาณ Input ถูก Quantized โดยตรงในระบบ PCM มาตรฐาน เหมือนกันสำหรับความผิดพลาดของ Quantizing ของการเปลี่ยนแปลงที่กำหนดได้ DPCM ต้องการจำนวนของระดับ Quantizing น้อยกว่า และแน่นอนว่ามีอัตรา bit น้อยกว่า PCM

DPCM ใช้ Quantizer คงที่ และส่วนคาดหมายที่คงที่ การส่งสามารถทำได้ต่อไปโดยใช้ Adaptive Quantizer และ Adaptive Predictor ดังแสดงในรูป (c) การเข้ารหัสของรูปคลื่นชนิดนี้ถูกเรียกว่า Adaptive Differential Pulse-Code Modulation; ADPCM Adaptive Predictor ถูกใช้เพื่อที่จะบันทึกธรรมชาติที่ไม่คงที่ของสัญญาณเสียง

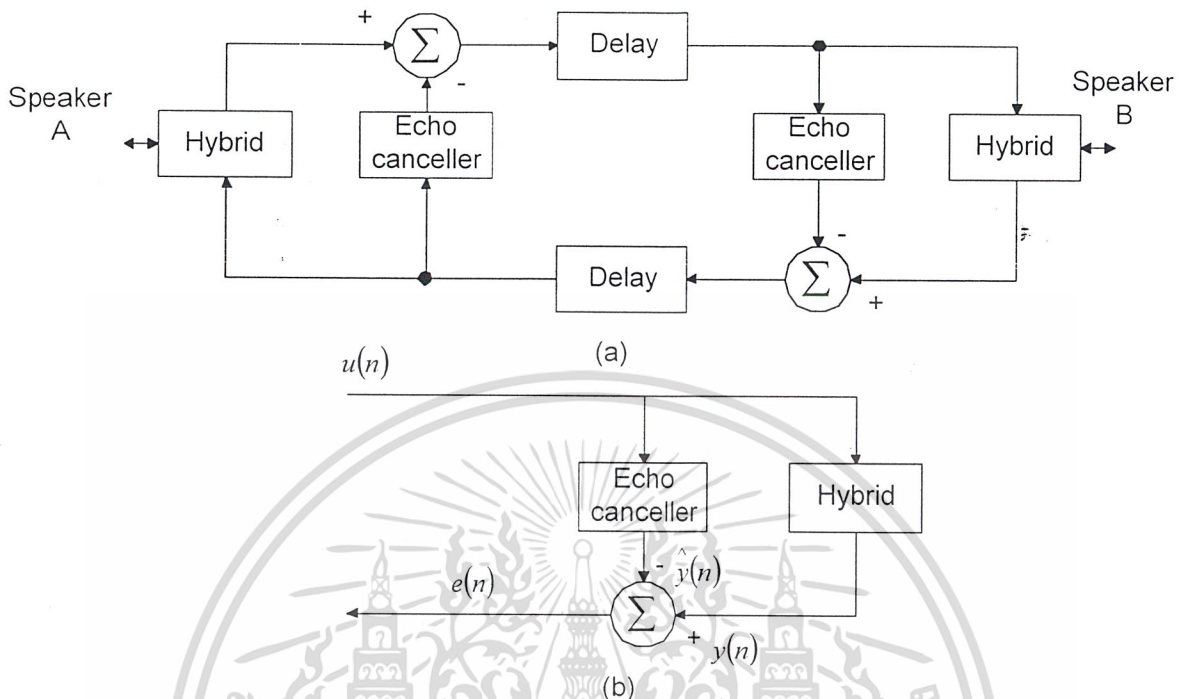
4. Echo Cancellation

ในการต่อโทรศัพท์ซึ่งเกี่ยวข้องทั้งการส่งแบบสี่สายและสองสาย Echo (เสียงก้อง) นั้นกำเนิดที่ Hybrid ซึ่งต่ำกับการส่งแบบสี่สายไปยังสองสาย เมื่อมีการเรียกโทรศัพท์ทางไกล เช่น โดยการใช้ดาวเทียมประจำที่ Geostationary Satellites Echo จะแสดงถึงการลดทอนซึ่งอาจจะเป็นสิ่งที่น่ารำคาญมากกว่าเสียงเบาหรือมี Noise รูปที่ 15 แสดงว่าวงจรดาวเทียมซึ่งไม่มีการป้องกัน Echo Hybrids ที่ปลายทั้งสองข้างของวงจรเปลี่ยนการส่งแบบสองสายที่ใช้บน Customer Loops และกลุ่มของสายส่งแบบสี่เส้นเป็นสิ่งจำเป็นสำหรับส่งวงจรพาหะ ช่วงเวลาสูงสุดของดาวเทียมการหน่วงเวลา 270 ms จะเกิดขึ้นในแต่ละทางบนสายแบบสี่สาย โดยแนวความคิด เมื่อมีนาย ก พุดอยู่ทางซ้าย เสียงพุดของเขาควรจะตามทางสายส่งด้านบนไปยัง Hybrid ทางด้านขวาและจะเข้าสู่วงจรสองสายอย่างไรก็ตามในทางปฏิบัติพลังงานเสียงไม่ทั้งหมดที่จะตรงเข้าวงจรสองสาย เป็นผลให้สัญญาณบางส่วนย้อนกลับทางสี่สายด้านล่างทำให้ได้ยิน โดยบุคคลด้านซ้ายจึงเป็น Echo ที่มี การหน่วงเวลาเท่ากับ 540 ms



รูปที่ 15 วงจรดาวเทียมที่ไม่มีการป้องกันเสียงสะท้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 16(a) วงจรความถี่รวมพร้อมทั้งคู่ของเครื่องกำจัดเสียงสะท้อน (b) นิยามของสัญญาณ

เพื่อจะชนะปัญหานี้ Echo canceller (เครื่องตัดเสียงสะท้อน) จึงถูกติดตั้งไว้ในเครือข่ายเป็นคู่ดังแสดงในรูปที่ 16 การกำจัดนี้จะทำได้โดยการประมาณค่าของการสะท้อนและลบจากสัญญาณที่กลับมา การสมมติข้างล่างนี้เป็นทางกลับของเสียงสะท้อน จากจุดที่ Canceller ต่อที่จุดที่ค่าประมาณเสียงสะท้อนถูกลบ เป็นเชิงเส้นและไม่ขึ้นอยู่กับเวลา

ฉะนั้นโดยการอ้างอิง Canceller เดียวในรูป เมื่อกำหนดสัญญาณที่ส่งกลับที่เวลา n อาจจะได้เป็น

$$y(n) = \sum_{k=0}^{\infty} h(k) \cdot u(n-k) + v(n) \quad 1.13$$

เมื่อ $u(n), u(n-1), \dots, u(n-M)$ เป็นตัวอย่างของเสียงพูดที่ไกลสุดจากผู้พูด $v(n)$ เป็นเสียงพูดที่ไกลที่สุดจากผู้พูด x บวกเข้ากับ Noise ที่เพิ่มเข้ามาที่เวลา n และ $h(k)$ เป็นการตอบสนอง Impulse ของทางเสียงสะท้อน เครื่องตัดเสียงสะท้อนจะทำหน้าที่ประมาณ $h(k)$ ของการตอบสนอง Impulse ของเส้นทางสะท้อน และจากนั้นประมาณเสียงสะท้อนออกมาเป็น Convolution Sum (ผลรวมผสม)

$$\hat{y}(n) = \sum_{k=0}^M \hat{h}(k) \cdot u(n-k) \quad 1.14$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในมหาวิทยาลัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

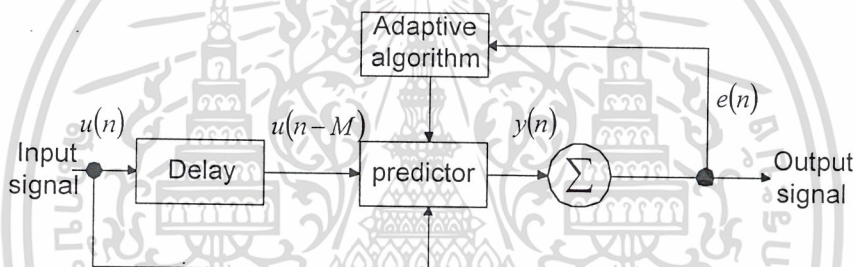
ซึ่งสามารถทำได้โดยใช้ Tapped-Delay-Line Filter กับสัมประสิทธิ์ $\hat{h}(0), \hat{h}(1), \dots, \hat{h}(M)$ สัญญาณผิดพลาด $e(n)$ จะถูกสร้างโดยการลบค่าประมาณ $\hat{y}(n)$ จากสัญญาณย้อนกลับ $y(n)$ ดังแสดงโดย

$$e(n) = y(n) - \hat{y}(n) \quad 1.15$$

สัญญาณผิดพลาด $e(n)$ ใช้สำหรับควบคุมการเปลี่ยนแปลงของสัมประสิทธิ์ของ Canceller $\hat{h}(0), \hat{h}(1), \dots, \hat{h}(M)$ ดังนั้นหลังจากที่ผ่านการคำนวณซ้ำๆ เล็กน้อยแล้ว จะรู้สึกรู้ว่าผลของการสะท้อนจะน้อยที่สุด

5. Adaptive Line Enhancer

Adaptive Line Enhancer; ALE เป็นอุปกรณ์ที่สามารถใช้ตรวจสอบคลื่น Sine ระดับต่ำที่ถูกฝังไว้ในส่วนหลังของ Noise ที่เพิ่มขึ้นมาบน Spectrum ที่มี Wire Band



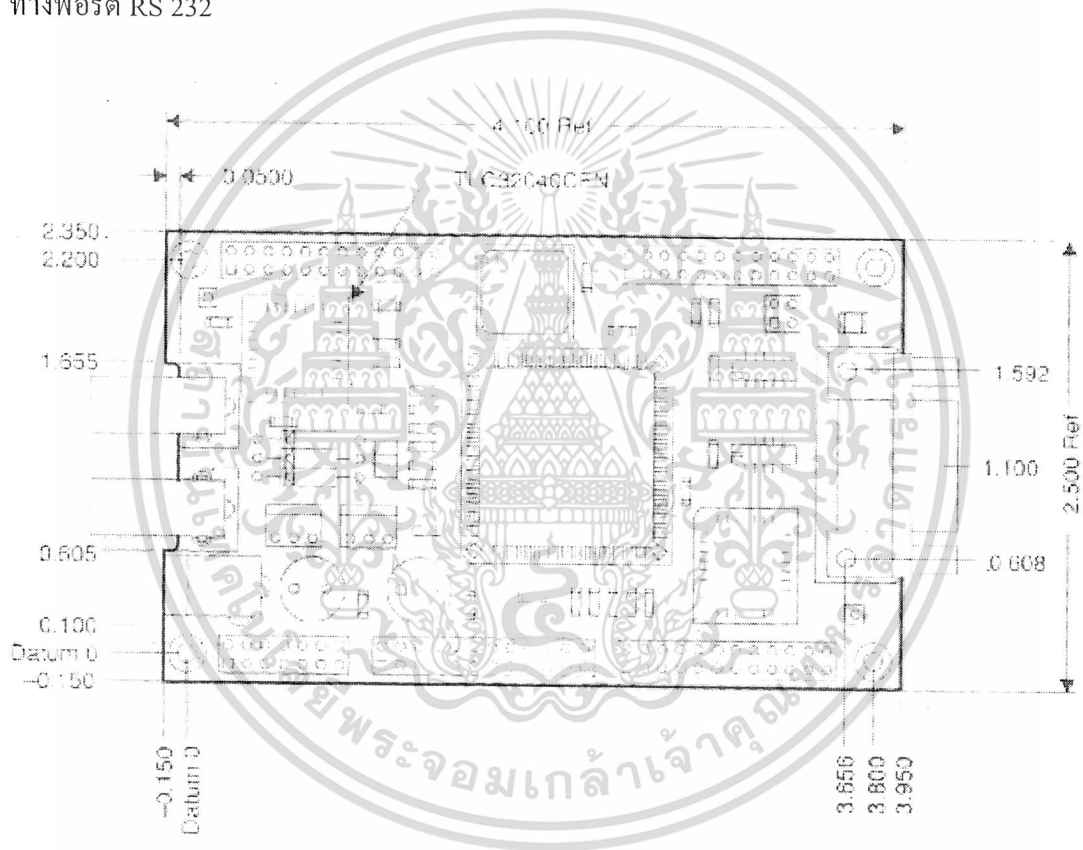
รูปที่ 17 Adaptive Line Enhancer

ALE ประกอบด้วยอุปกรณ์หน่วยเวลาและ Linear Predictor (ส่วนคาดหมายเชิงเส้น) Output ของ Predictor $y(n)$ จะถูกลบกับสัญญาณ $u(n)$ เพื่อสร้างสัญญาณผิดพลาด $e(n)$ สัญญาณนี้จะถูกใช้เพื่อควบคุมการเปลี่ยนแปลงของสัมประสิทธิ์ Predictor เท่ากับ $u(n-D)$ สัญญาณ Input ดั้งเดิม $u(n)$ ถูกหน่วงด้วย D วินาที เมื่อ D นั้นเท่ากับหรือมากกว่าช่วงของการสุ่ม หน้าที่หลักของ Delay-Parameter D คือ ตัดความสัมพันธ์ซึ่งอาจจะมีอยู่ระหว่างองค์ประกอบของ Noise ใน Predictor Input ที่หน่วงเวลาแล้ว $u(n-D)$ สำหรับเหตุผลนี้ ค่า Delay Parameter D จึงถูกเรียกอีกชื่อ ว่า Deceleration Parameter ของ ALE ฉะนั้น ALE อาจจะถูกมองเป็น Adaptive Filter ซึ่งถูกออกแบบเพื่อตัดองค์ประกอบแบบ Wire Band เช่น Sine Waves ถูกส่งด้วยการลดทอนเล็กน้อย พุดได้อีกอย่างว่า มันสามารถถูกใช้เพื่อเพิ่มความประสิทธิภาพการปรากฏของรากของ Sine wave ซึ่ง Spectrum ประกอบด้วย Harmonic Lines

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

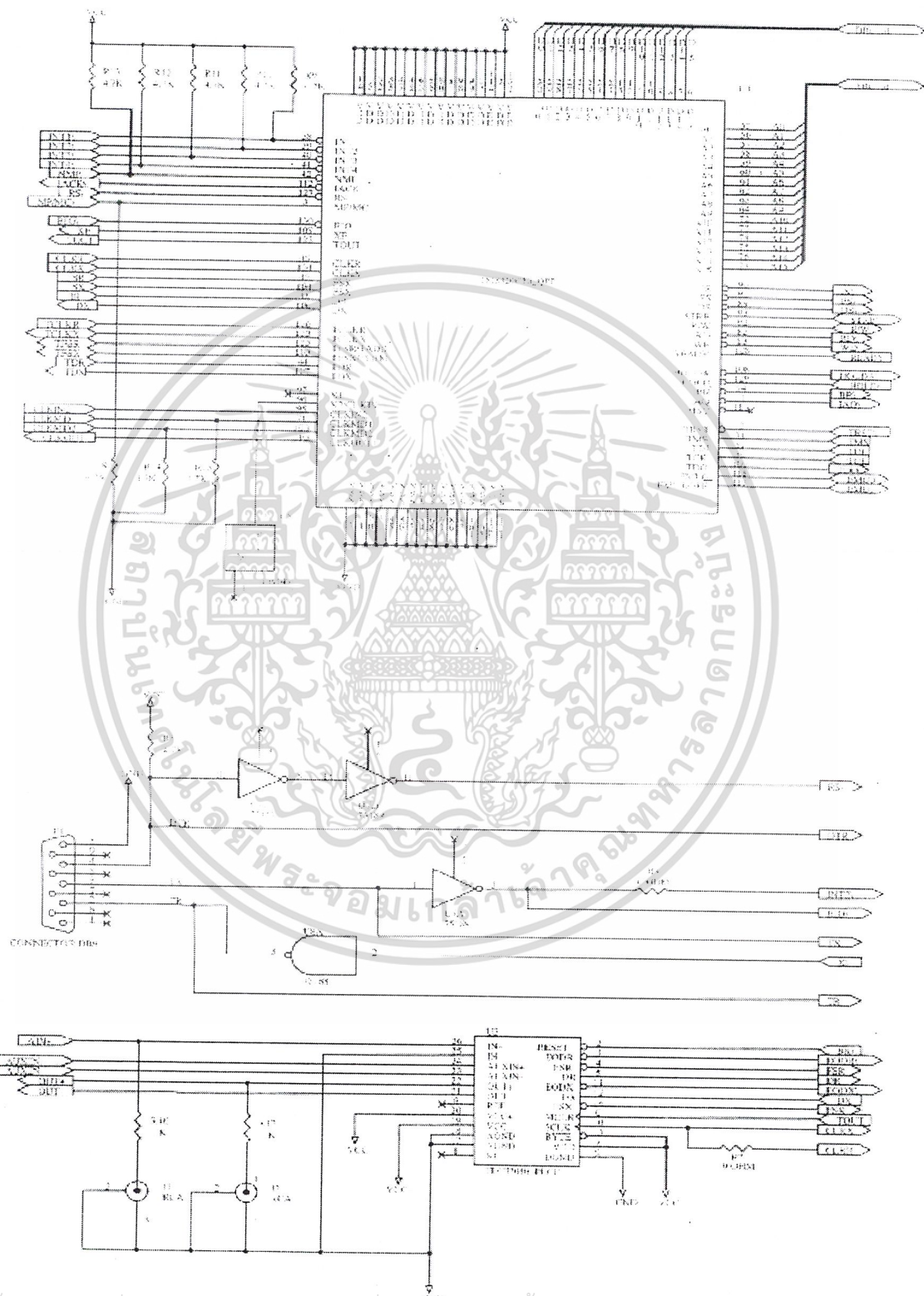
10.บอร์ด DSP

ระบบที่ต้องการ คอมพิวเตอร์ที่จะใช้กับชุด DSK จะต้องมีคุณสมบัติ คือ เป็นเครื่อง IBM PC/AT หรือเทียบเท่า หน่วยความจำ RAM น้อยที่สุด 640 Kbytes มี Floppy Disc ขนาด 5.25 นิ้ว และตัว DSP บอร์ดต้องการไฟฟ้าเลี้ยง $9V_{AC}$ (กระแสขั้นต่ำ 250 ma) พร้อมขั้วต่อขนาด 3.1 มิลลิเมตรและสายเชื่อมกับคอมพิวเตอร์ผ่านทางพอร์ต RS 232 และ DSP บอร์ดต้องการไฟฟ้าเลี้ยง $9V_{AC}$ (กระแสขั้นต่ำ 250 ma) พร้อมขั้วต่อขนาด 3.1 มิลลิเมตรและสายเชื่อมกับคอมพิวเตอร์ผ่านทางพอร์ต RS 232



รูปที่ 18 รูปแบบและขนาดของบอร์ด DSK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของบอร์ด DSK รูปร่างภายนอกของบอร์ด DSK แสดงดังรูปที่ 18 อุปกรณ์โดยส่วนใหญ่เป็นแบบติดตั้งบนผิวหน้าลงแผ่นวงจรพิมพ์ จะมีเพียงตัวเก็บประจุค่าสูงและไอซี Regulator ที่เป็นแบบธรรมดา บนบอร์ด DSK จะมีชิพที่เป็นตัวหลักๆ อยู่ 2 อย่างคือ TMS320C5x ซึ่งเป็น DSP Processor และ Digital to Analog ชิพเบอร์ TLC32040CFN ขนาดของบอร์ดอยู่ที่ 2.5×3.5 นิ้ว มีขั้วต่อชนิด D เพื่อใช้ในการเชื่อมต่อกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม RS-232 1 ชุด มีขั้วต่อ RCA สำหรับรับและส่งสัญญาณทาง Analog

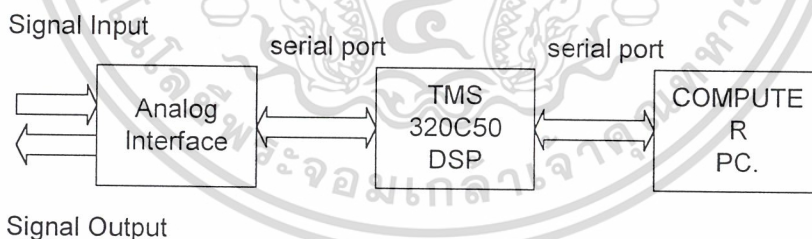
วงจรมบูรณ บอร์ด DSK มีวงจรมบูรณแสดงในรูปที่ 19 ในส่วนบนเป็นวงจรมบูรณของการเชื่อมตอระหวางชิพ DSP กับอุปกรณ์ตอรวมอื่น ๆ ในส่วนกลางเป็นภาคจ่ายไฟของบอร์ด DSK ตัวสำคัญของบอร์ดคือ ชิพ DSP เบอร์ TMS320C5x เป็นไมโครโปรเซสเซอร์ตัวหนึ่งที่ใช้งานด้านการประมวลผลสัญญาณดิจิทัลโดยเฉพาะทำหน้าที่ประมวลผลข้อมูลทั้งหมดบนบอร์ด DSK นี้ โดยต้องทำงานร่วมกับ Software ซึ่งจะอธิบายในส่วนต่อไป ชิพ DSP ทำงานได้โดยอาศัยสัญญาณนาฬิกาจากโมดูลกำเนิดสัญญาณนาฬิกา 40 MHz ชิพ DSP โดยมี Ceramic Resonator ช่วยควบคุมความถี่ของสัญญาณนาฬิกาให้มีความเที่ยงตรงอีกที ชิพ ADC/DAC เบอร์ TLC320C40 ทำหน้าที่รับสัญญาณ Analog ทาง Input ที่จ่ายเข้ามาทาง J_1 แล้วแปลงเป็นข้อมูลทาง Digital ส่งไปประมวลผลต่อที่ ชิพ DSP และเมื่อต้องการส่งสัญญาณออกไปที่ ชิพ ADC/DAC เบอร์ TLC320C40 จะส่งข้อมูล Digital เพื่อทำการแปลงเป็นสัญญาณ Analog จ่ายออกไปทาง J_2 รายละเอียดของการแปลงสัญญาณ Digital จะมีขนาด 14 บิต สามารถกำหนดอัตรา Sampling ใน 1 วินาที สำหรับการติดต่อกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม RS-232C นั้น จะเป็นหน้าที่ของเบอร์ 75188D และเบอร์ 75189D โดย เบอร์ 75188D ทำหน้าที่ส่งข้อมูลจากบอร์ด DSK ออกไปทาง P_1 ขั้วต่อชนิด DB9 ขาที่ต่อเข้ากับพอร์ตอนุกรม RS-232C ส่วน เบอร์ 75188D ทำหน้าที่เป็น Buffer กำหนดข้อมูลจากคอมพิวเตอร์ ส่วนภาคจ่ายไฟบอร์ด DSK ต้องการไฟสลับ 9 โวลต์ จากภายนอกจ่ายเข้ามาทาง J_3, D_1, D_2, C_1 และ C_3 ทำหน้าที่เป็นวงจรมบูรณแรงดันแล้วแบ่งแรงดันเป็นแรงดันบวกและแรงดันลบจ่ายให้แก่ไอซีเรกูเรเตอร์ U_1 และ U_2 ได้เป็นแรงดัน ± 5 โวลต์ไปเลี้ยงบอร์ด DSK

เกี่ยวกับชิพ TMS320C5x เป็นตัวที่สำคัญที่สุดในบอร์ด DSK เป็น Microprocessor ตัวหนึ่งที่ถูกออกแบบมาให้ใช้งานด้าน DSP โดยเฉพาะมีคาบเวลาของไซเคิลของคำสั่งเป็น 50 n sec ในการทำงานคำสั่งแต่ละตัวจะใช้เวลาหนึ่งไซเคิล ทำให้สามารถประมวลผลได้ถึง 10 ล้านคำสั่งต่อวินาที จึงส่งผลให้ชิพนี้สามารถทำการประมวลผลแบบเวลาจริง (Real Time) ได้ทัน ในตัว TMS320C5x มีหน่วยความจำเร็วสูงขนาด 9 K word โดยที่แต่ละ Word จะมีขนาด 16 bit และมี ROM ภายในขนาด

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ห้ามการใชวงมเพื่อการค้าโดยไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้า
2 K word สามารถอ้างอิงหน่วยความจำภายนอกได้ถึง 128 k word แบ่งเป็นหน่วยความจำข้อมูล
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และหน่วยความจำโปรแกรมอย่างละ 64 K word ALU (Arithmetic Logic Unit) และ Accumulator ขนาด 32 bit ที่สามารถคูณเลขขนาด 32 bit ได้ภายใน 1 Machine Cycle ทั้งนี้เนื่องจากตัว ALU มี วงจรคูณแบบ 16×16 บิต Hardwire นอกจากนี้ยังมีตัวตั้งเวลาและส่วนกำเนิดสัญญาณนาฬิกาอยู่ ภายในชิพ ส่วนของการ Interface เป็นแบบขนาน 16 บิต แบ่งเป็น Input 16 Channel Output 16 Channel วงจรติดต่อหน่วยความจำร่วม (Global Data Memory Interface) ตัว TMS320C5x สามารถต่อใช้งานเพียงตัวเดียวหรือจะต่อขนานกันหลายตัวที่เรียกว่าเป็นระบบ Multi-Processor System ก็ได้ ในการใช้งานกับหน่วยความจำภายนอกไม่ว่าจะเป็น EPROM หรือ ROM ตัวชิพจะ ทำงานได้ช้ามาก จึงได้ใช้หน่วยความจำแบบภายในชิพเป็นที่เก็บโปรแกรมแทน ในการใช้งานให้ โหลดโปรแกรมจากหน่วยความจำภายนอกมาเก็บในชิพแล้วจึงทำการรันโปรแกรมทำงาน โดย TMS320C5x ได้เพิ่มคำสั่ง “Block Transfer” ที่ใช้ในการโหลดข้อมูลจากหน่วยความจำภายนอกมา ที่ละ block แล้วมาเก็บไว้ใน RAM ของชิพเอง ด้วยวิธีการนี้จะช่วยให้การทำงานเร็วมากยิ่งขึ้น การอ้างแอดเดรสของ TMS320C5x มีด้วยกัน 2 โหมดคือ โดยตรง (Direct) ทางอ้อม (Indirect) และทันทีทันใด (Immediate) นอกจากนี้มีรีจิสเตอร์เสริมอีก 8 ตัว

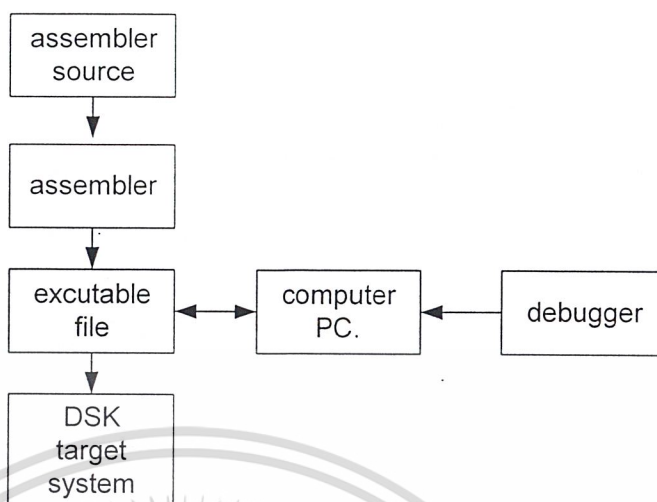
ขั้นตอนการพัฒนาโปรแกรม ขั้นตอนต่างๆ ได้แสดงไว้ในรูปที่ 21 จุดประสงค์ของการ พัฒนาคือ ต้องการสร้าง Software ที่สามารถใช้งานกับบอร์ด DSK ในชุด DSK จะมาพร้อมกับ โปรแกรม Assembler และ Debugger ใช้ในการเขียน แก้ไขทดสอบและปรับปรุง Software ให้ได้ ผลตามต้องการ



รูปที่ 20 การต่อใช้งานบอร์ด DSP

Assembler มีหน้าที่ในการแปลไฟล์ข้อมูลภาษา Assembly ให้เป็นไฟล์ภาษาเครื่องหรือที่มี ชื่ออีกอย่างว่า Object code ที่ทำให้ชิพสามารถนำข้อมูลรหัสไปปฏิบัติงานได้โดยตรง Assembler ของ DSK เป็นแบบที่สามารถกำหนด “directive” ก่อนที่จะทำการ Assembly ได้ ทำให้สามารถ กำหนดตำแหน่งแอดเดรสของโปรแกรมเป็นแบบสัมพัทธ์ (คือเป็นแอดเดรสมาลอยๆ ที่สามารถ กำหนดตำแหน่งที่แน่นอนได้ในภายหลัง) ทำให้ไม่จำเป็นต้องมีขั้นตอนของการ Linker

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 21 ขั้นตอนการพัฒนาซอฟต์แวร์

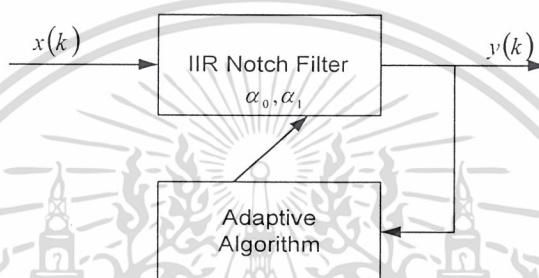
Debugger มีหน้าที่ใช้ช่วยในการหาตำแหน่งและแก้ไขข้อผิดพลาด (Error หรือ Bug) ต่างๆ ที่อาจเกิดในตัวโปรแกรม การทำงานของโปรแกรมจะแบ่งการทำงานในส่วนต่างๆ แยกเป็นคณละหน้าต่างที่ง่ายต่อการใช้ *Debugger* จะมีหน้าที่พื้นฐานที่สำคัญ ได้แก่ Single Step Execution, Breakpoint Setting, และ Execution Halt เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

การวิเคราะห์และออกแบบ

โครงการนี้เป็นการศึกษาประยุกต์ Adaptive Algorithm เข้ากับ IIR Digital Notch Filter อันดับ 2 ที่มีค่า Q-Factor คงที่ เพื่อออกแบบฟิลเตอร์ที่มีความสามารถปรับค่า Q-Factor และ Simplified Adaptive Algorithm ที่มีการปรับค่า Q-Factor เพื่อความรวดเร็วและสามารถติดตามสัญญาณทาง Input ตามความถี่ที่ใช้งานได้เร็วยิ่งขึ้น



รูปที่ 23 Block Diagram ของ Adaptive IIR Notch Filter

จาก Block Diagram ในรูปที่ 23 สัญญาณ Input ประกอบด้วยสัญญาณคลื่นไซน์และ Noise $x(k) = A \cdot \cos(k \cdot \omega) + n(k)$ วงจรกรองความถี่ IIR Notch Filter จะถูกปรับเปลี่ยนค่าโดย Adaptive Algorithm สัญญาณควบคุมได้มาจากสัญญาณ Output ในอดีต และจากการประมวลผลจะได้ Output ที่เป็นสัญญาณรูปคลื่นไซน์ก็ต่อเมื่อเต็มรูปคลื่นก็ต่อเมื่อ ค่า $\alpha_1(k) = \cos(\omega_0)$

ขั้นตอนการออกแบบ

1. แปลง Transfer Function ของ Analog Notch Filter เป็น Digital Notch Filter โดยใช้ วิธี Bilinear Transform
2. ทำการศึกษา Parameters ในสมการ Transfer Function มีผลอย่างไรต่อสัญญาณโดยการทำ Transfer Function ไปพลอตกราฟแสดงได้ดังกราฟรูปที่ 1
3. ค่าเฉลี่ยกำลังสอง ของ $[y(k)]^2$ จะเป็นสัดส่วนโดยตรงกับ $|H(e^{j\omega})|^2$ ที่ $z = e^{j\omega}$ เมื่อใช้วิธีการของ Stochastic Gradient ค่า $\alpha_1(k)$ ที่ทำให้ $[y(k)]^2$ มีค่าสูงสุด Adaptive Algorithm สามารถกำหนดได้ดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}\alpha_1(k+1) &= \alpha_1(k) - \frac{\mu_1}{2} \frac{\partial y^2(k)}{\partial \alpha_1(k)} \\ &= \alpha_1(k) - \mu_1 y(k) \cdot \psi(k)\end{aligned}$$

พิสูจน์ว่า $|H(\omega)|$ เทียบกับ α_1 จะมีจุดต่ำสุดเพียงจุดเดียว

4. ทำการพิสูจน์ว่าคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k ดังกราฟที่ 3 ขณะยังไม่มีกรแยกโดยใช้ Transfer Function และกราฟที่ 4 ขณะที่ได้ทำการแยกโดยใช้ Transfer Function ว่ามีลักษณะเหมือนกัน

5. การเปลี่ยนแปลงค่าของ $\alpha_1(k)$ จะเร็วขึ้นได้ โดยการปรับค่า α_0 ให้เปลี่ยนแปลงไปด้วยวิธีการ Stochastic Gradient เนื่องจากว่า $\alpha_1(k)$ จะติดตามความถี่ได้เร็วหรือไม่ขึ้นอยู่กับค่าของ $y(k) \cdot \psi(k)$ เรียกว่า Update Function และค่าของ $y(k)$ และ $\psi(k)$ ต่างก็มาจาก $H(z)$ ดังนั้น $[y(k) \cdot \psi(k)]^2$ จะเป็นสัดส่วนโดยตรงกับ $\frac{\partial [H(\alpha_1(k))]^2}{\partial \alpha_1(k)}$ เพื่อใช้วิธีการ Stochastic Gradient จะต้อง

พิสูจน์เทอม $\frac{\partial [H(\alpha_1(k))]^2}{\partial \alpha_1(k)}$ เมื่อพลอตกับ α_0 แล้วต้องมียอดสูงสุดของกราฟเพียงยอดเดียวแสดงได้ดังกราฟที่ 5

6. จากกราฟมียอดสูงสุดยอดเดียวจึงสามารถใช้วิธี Stochastic Gradient ได้ Adaptive Algorithm สำหรับ $\alpha_0(k)$ สามารถกำหนดได้ดังสมการ

$$\alpha_0(k+1) = \alpha_0(k) + \frac{\mu_0}{2} \frac{\partial [y(k) \cdot \psi(k)]^2}{\partial \alpha_0(k)}$$

$$\alpha_0(k+1) = \alpha_0(k) + \mu_0 y(k) \cdot \psi(k) [\psi_{01}(k) + \psi_{02}(k) y(k)]$$

โดยที่ $\psi_{01}(k) = \frac{\partial y(k)}{\partial \alpha_0(k)}$ และ $\psi_{02}(k) = \frac{\partial \psi(k)}{\partial \alpha_0(k)}$

ψ_{01}, ψ_{02} : เป็นสัญญาณควบคุมแบบ Adaptive ของ $\alpha_0(k)$

μ_0 : เป็น Step Size Parameters

7. จากสมการ $\alpha_0(k+1)$ จะเห็นว่าต้องใช้สัญญาณควบคุมแบบ Adaptive 2 ตัวคือ

$\psi_{01}(k), \psi_{02}(k)$ และจากสมการ $\psi_{01}(k)$ ต้องใช้สัญญาณ Input 2 ตัวคือ $x(k), y(k)$ เพื่อสร้างสัญญาณควบคุมแบบ Adaptive $\psi_{01}(k)$ จากสมการ $\psi_{02}(k)$ ต้องใช้สัญญาณ Input 3 ตัว คือ $y(k), \psi(k), \psi_{01}(k)$ เพื่อสร้างสัญญาณควบคุมแบบ Adaptive $\psi_{02}(k)$ ซึ่งวงจรที่ได้ยังซับซ้อนและยากในการออกแบบ แสดงได้ใน Adaptive IIR Notch Filter

8. สำหรับ $\alpha_0(k)$ ที่จะทำให้ $\alpha_1(k)$ มีค่าสูงสุด จะหาได้จาก $\alpha_{0,opt} = \frac{\partial H_1(\alpha_0(k))}{\partial \alpha_0(k)} = 0$

$$\alpha_{0,opt}(k) = \frac{\sin(\omega) - \cos(\omega) + \alpha_1(k)}{\sin(\omega) + \cos(\omega) - \alpha_1(k)} \quad \text{เมื่อ } \alpha_1(k) < \cos(\omega)$$

$$\alpha_{0,opt}(k) = \frac{\sin(\omega) + \cos(\omega) - \alpha_1(k)}{\sin(\omega) - \cos(\omega) + \alpha_1(k)} \quad \text{เมื่อ } \alpha_1(k) > \cos(\omega)$$

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของภาควิชาวิศวกรรมไฟฟ้า ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. ฟังก์ชันของ $\psi(k), \psi_{01}(k), \psi_{02}(k)$ จะขึ้นอยู่กับฟังก์ชันหลายตัว ไม่ขึ้นกับฟังก์ชันตัวใดตัวหนึ่ง ถ้าสามารถเปลี่ยนฟังก์ชันเหล่านี้ให้ขึ้นอยู่กับฟังก์ชันตัวใดตัวหนึ่งได้ ก็จะทำให้ห้วงจรมีความง่ายขึ้น ซึ่งทำได้โดยการหา Transfer Function ของฟังก์ชันเหล่านี้

$$\text{Transfer Function ของ } \psi(k) \text{ เขียนสัญลักษณ์ด้วย } G(z) = \frac{\psi(z)}{y(z)}$$

$$\text{Transfer Function ของ } \psi_{01}(k) \text{ เขียนสัญลักษณ์ด้วย } G_{01}(z) = \frac{\psi_{01}(z)}{y(z)}$$

$$\text{Transfer Function ของ } \psi_{02}(k) \text{ เขียนสัญลักษณ์ด้วย } G_{02}(z) = \frac{\psi_{02}(z)}{y(z)}$$

เมื่อได้ Transfer Function ทั้ง 3 แล้ว ก็จะนำไปเขียนโคอะแกรมได้ เนื่องจากฟังก์ชันแต่ละตัวขึ้นอยู่กับฟังก์ชันใดฟังก์ชันหนึ่งเท่านั้น จึงสามารถเขียนโคอะแกรมแสดงใน Adaptive IIR Notch Filter คุณสมบัติ $G(z), G_{01}(z), G_{02}(z)$ เทียบกับ Amplitude แสดงได้ดังกราฟที่ 6

10. คุณลักษณะของ $|H(\alpha_0)|$ เทียบกับ α_0 แสดงได้ดังกราฟรูปที่ 7 จากรูปจะเห็นได้ว่ากราฟไม่มียอดสูงสุด ดังนั้นค่าสูงสุดของค่าเฉลี่ยของ $y(k)*\psi(k)$ สามารถประมาณให้เท่ากับค่าเฉลี่ยของ $\psi(k)$ ได้ ดังนั้นค่าเฉลี่ยของ $\psi(k)$ เป็นสัดส่วนโดยตรงกับ $\frac{\partial H(\alpha_0(k))}{\partial \alpha_0(k)}$

11. จากสมการ $\frac{\partial H(\alpha_0(k))}{\partial \alpha_0(k)}$ ซึ่งเป็นฟังก์ชันของ α_0 สามารถเขียนใหม่ให้อยู่ในรูปที่ง่ายขึ้นได้ดังสมการ

$$\Pi(\alpha_0) = \frac{\partial H(\alpha_0(k))}{\partial \alpha_0(k)}$$

คุณลักษณะของ $|\Pi(\alpha_0)|$ เทียบกับ α_0 แสดงได้ดังกราฟรูปที่ 8 จะเห็นได้ว่ามียอดสูงสุดเพียงยอดเดียวเหมือนกับกราฟที่ 5 ดังนั้นสามารถตัดแปลงหรือปรับ Algorithm สำหรับ Update ของ $\alpha_0(k)$ ได้ดังสมการ

$$\alpha_0(k+1) = \alpha_0(k) + \frac{\mu_0}{2} \frac{\partial \psi(k)}{\partial \alpha_0(k)}$$

$$\alpha_0(k+1) = \alpha_0(k) + \mu_0 \psi(k) \psi_{02}(k)$$

$$\text{โดยที่ } \psi_{02}(k) = \frac{\partial \psi(k)}{\partial \alpha_0(k)}$$

μ_0 : เป็น Step Size Parameters

$\psi_{02}(k)$ เป็นสัญญาณควบคุมแบบ Adaptive ของ $\alpha_0(k)$

$\psi_{02}(k)$ จะเห็นได้ว่าใช้สัญญาณควบคุมเพียงตัวเดียว สามารถแสดงได้ในโคอะแกรมของ

Simplified Adaptive IIR Notch Filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 Adaptive Algorithm โดยการใช้ฟิลเตอร์แบบค่า Q-Factor คงที่

กำหนดให้ช่วงเวลาการแซมปลิง (Sampling) เป็น T วินาที ถูกระบุด้วยค่า ω (Normalize) อยู่ที่ 1 ตัวแปร k (เลขจำนวนเต็ม) เป็นเวลาการแซมปลิง และสัญญาณทางค่าน Input เขียนได้ดังสมการ

$$X(k) = A \cdot \cos(k \cdot \omega) + n(k) \quad 2.1$$

โดยที่ $X(k)$ คือสัญญาณทางค่าน Input ซึ่งประกอบด้วยคลื่นไซน์

$A \cdot \cos(k \cdot \omega)$ A เป็นค่าคงที่

ω (radian/sec) ความเร็วเชิงมุม มีค่าในช่วง $0 < \omega < \pi$

$n(k)$ คือสัญญาณรบกวนแบบเกาส์

สำหรับ Transfer Function ของวงจรกรองความถี่ IIR Digital Notch Filter อันดับสองที่มีความถี่กลางเป็น ω_0 คือ

$$H(z) = \frac{1 + \alpha_0}{2} \left(\frac{1 - 2\alpha_1 z^{-1} + z^{-2}}{1 - \alpha_1(1 + \alpha_0)z^{-1} + \alpha_0 z^{-2}} \right) \quad 2.2$$

โดยที่ $\alpha_1(k)$ มีค่าระหว่าง $-1 < \alpha_1(k) < 1$ เป็นสัมประสิทธิ์แปรค่าที่ใช้สำหรับ

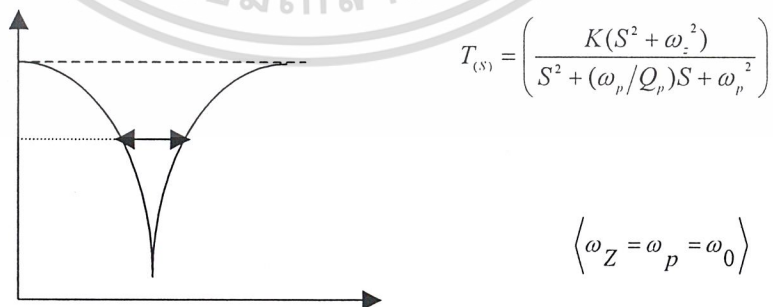
กำหนด ω_0 ซึ่ง $\alpha_1(k) = \cos(\omega_0)$

α_0 มีค่าอยู่ระหว่าง $0 < \alpha_0 < 1$ เป็นสัมประสิทธิ์คงที่สำหรับกำหนดค่า Q-Factor ของวงจรกรองความถี่

วิธีการแปลงจาก Analog Notch Filter เป็น Digital Notch Filter

ใช้วิธี Bilinear Transform ทำได้ดังนี้

Analog Notch Filter



$$\left\langle \alpha = \frac{\sqrt{1 - G_B^2}}{G_B} (1 + \Omega_0^2) \tan\left(\frac{\Delta\omega}{2}\right) \right\rangle \quad \left\langle \Omega_0 = \tan\left(\frac{\omega_0}{2}\right) \right\rangle \quad H(s) = \left(\frac{s^2 + \Omega_0^2}{s^2 + \alpha s + \Omega_0^2} \right)$$

Bilinear Transform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}
H_{(Z)} = H_{(s)} \Big|_{s=\left(\frac{1-Z^{-1}}{1+Z^{-1}}\right)} &= \left[\frac{\left(\frac{1-Z^{-1}}{1+Z^{-1}}\right)^2 + \Omega_0^2}{\left(\frac{1-Z^{-1}}{1+Z^{-1}}\right) + \alpha\left(\frac{1-Z^{-1}}{1+Z^{-1}}\right) + \Omega_0^2} \right] \\
&= \left[\frac{(1-Z^{-1})^2 + \Omega_0^2(1+Z^{-1})^2}{(1-Z^{-1})^2 + \alpha(1-Z^{-1})(1+Z^{-1}) + \Omega_0^2(1+Z^{-1})^2} \right] \\
&= \left[\frac{(1+\Omega_0^2) - 2(1-\Omega_0^2)Z^{-1} + (1+\Omega_0^2)Z^{-2}}{(1+\alpha+\Omega_0^2) + 2(1-\Omega_0^2)Z^{-1} + (1-\alpha+\Omega_0^2)Z^{-2}} \right] \\
&= \left[\left(\frac{1+\Omega_0^2}{1+\alpha+\Omega_0^2} \right) \frac{1-2\left(\frac{1-\Omega_0^2}{1+\Omega_0^2}\right)Z^{-1}+Z^{-2}}{1-2\left(\frac{1-\Omega_0^2}{1-\alpha+\Omega_0^2}\right)Z^{-1}+\left(\frac{1-\alpha+\Omega_0^2}{1+\alpha+\Omega_0^2}\right)Z^{-2}} \right] \\
\langle \alpha = (1+\Omega_0^2)\beta \rangle &= \left[\left(\frac{1+\Omega_0^2}{(1+\Omega_0^2)+(1+\Omega_0^2)\beta} \right) \frac{1-2\left(\frac{1-\Omega_0^2}{1+\Omega_0^2}\right)Z^{-1}+Z^{-2}}{1-2\left(\frac{1-\Omega_0^2}{(1+\Omega_0^2)+(1+\Omega_0^2)\beta}\right)Z^{-1}+\left(\frac{(1+\Omega_0^2)-(1+\Omega_0^2)\beta}{(1+\Omega_0^2)+(1+\Omega_0^2)\beta}\right)Z^{-2}} \right] \\
\left\langle \frac{1-\Omega_0^2}{1+\Omega_0^2} \right\rangle = \left\langle \frac{1-\tan^2\left(\frac{\omega_0}{2}\right)}{1+\tan^2\left(\frac{\omega_0}{2}\right)} \right\rangle = \cos\omega_0 & \quad \left\langle \beta = \frac{1-\alpha_0}{1+\alpha_0} \right\rangle \\
&= \left(\frac{1+\alpha_0}{(1+\alpha_0)+(1-\alpha_0)} \right) \frac{1-2\cos\omega_0 Z^{-1}+Z^{-2}}{1-2\frac{(1+\alpha_0)}{(1+\alpha_0)+(1-\alpha_0)}\cos\omega_0 Z^{-1}+\left(\frac{(1+\alpha_0)-(1-\alpha_0)}{(1+\alpha_0)+(1-\alpha_0)}\right)Z^{-2}} \\
&= \left(\frac{1}{1+\beta} \right) \left[\frac{1-2\cos\omega_0 Z^{-1}+Z^{-2}}{1-2\frac{\cos\omega_0}{(1+\beta)}Z^{-1}+\left(\frac{1-\beta}{1+\beta}\right)Z^{-2}} \right]
\end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$H(z) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{1 - 2 \cos \omega_0 Z^{-1} + Z^{-2}}{1 - (1 + \alpha_0) \cos \omega_0 Z^{-1} + \alpha_0 Z^{-2}} \right) \quad 2.3$$

กำหนดให้ $\cos \omega_0 = \alpha_1$

$$H(z) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{1 - 2\alpha_1 Z^{-1} + Z^{-2}}{1 - \alpha_1(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}} \right)$$

เพื่อนำ Adaptive Algorithm ไปใช้งานจึงต้องมีการศึกษาค่า Parameters แต่ละตัวของ $H(z)$ โดยแทน $z = e^{j\omega}$ มีค่าระหว่าง $0 < \omega < \pi$ แล้วพลอตกราฟของ $|H(e^{j\omega})|$ เทียบกับแกน ω พบว่าค่าของ α_0 มีค่าอยู่ระหว่าง $0 < \alpha_0 < 1$ จะมีผลต่อการเปลี่ยนแปลงของ Q-Factor โดยมีความสัมพันธ์ที่แปรผันต่อกันดังนี้

1. เมื่อค่า α_0 (เข้าใกล้ศูนย์) มีค่าน้อย ค่า Q-Factor มีค่าน้อยลง กราฟของ $H(e^{j\omega})$ จะมีความกว้างมาก
2. เมื่อค่า α_0 (เข้าใกล้หนึ่ง) มีค่ามาก ค่า Q-Factor มีค่ามากขึ้น กราฟของ $H(e^{j\omega})$ จะมีความกว้างน้อย ค่าของ α_1 จะมีผลทำให้ความถี่กลาง ω_0 เบี่ยงไปโดยที่
 - 2.1 ค่าของ α_1 เท่ากับศูนย์ จะทำให้ ω_0 อยู่ตรงกลางพอดี
 - 2.2 ค่าของ α_1 เป็นลบ ความถี่ ω_0 จะเบี่ยงเบนไปทางซ้าย
 - 2.3 ค่าของ α_1 เป็นบวก ความถี่ ω_0 จะเบี่ยงเบนไปทางขวา

เพื่อหาการเปลี่ยนแปลงของ $\alpha_1(k)$ เมื่อมีสัญญาณ Input ป้อนเข้ามา จึงต้องหาเอาที่พหุ $y(k)$ เมื่อ $Y(z)$ เป็น Z-Transform ของ $y(k)$ ความสัมพันธ์ของ Transfer Function จะเป็น $H(z) = \frac{Y(z)}{X(z)}$ หรือ $Y(z) = H(z) \cdot X(z)$ เมื่อ $Y(z) = H(z) \cdot X(z)$

$$Y(z) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{1 - 2\alpha_1(k)Z^{-1} + Z^{-2}}{1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}} \right) X(z)$$

$$Y(z)(1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}) = \left(\frac{1 + \alpha_0}{2} \right) (1 - 2\alpha_1(k)Z^{-1} + Z^{-2}) X(z)$$

$$(Y(z) - \alpha_1(k)(1 + \alpha_0)Y(z)Z^{-1} + \alpha_0 Y(z)Z^{-2}) = \left(\frac{1 + \alpha_0}{2} \right) (X(z) - 2\alpha_1(k)X(z)Z^{-1} + X(z)Z^{-2})$$

Take Inverse Z-Transform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$(y(k) - \alpha_1(k)(1 + \alpha_0)y(k-1) + \alpha_0 y(k-2)) = \left(\frac{1 + \alpha_0}{2}\right)(x(k) - 2\alpha_1(k)x(k-1) + x(k-2))$$

$$y(k) = \left(\frac{1 + \alpha_0}{2}\right)(x(k) - 2\alpha_1(k)x(k-1) + x(k-2)) + \alpha_1(k)(1 + \alpha_0)y(k-1) - \alpha_0 y(k-2) \quad 2.4$$

ค่าเฉลี่ยกำลังสอง ของ $[y(k)]^2$ จะเป็นสัดส่วนโดยตรงกับ $|H(e^{j\omega})|^2$ ที่ $z = e^{j\omega}$ เมื่อใช้วิธี

การของ Stochastic Gradient ค่า $\alpha_1(k)$ ที่ทำให้ $[y(k)]^2$ มีค่าสูงสุด Adaptive Algorithm สามารถกำหนดได้ดังสมการ

$$\alpha_1(k+1) = \alpha_1(k) - \frac{\mu_1}{2} \frac{\partial y^2(k)}{\partial \alpha_1(k)} \quad 2.5$$

$$= \alpha_1(k) - \mu_1 y(k) \cdot \psi(k)$$

เมื่อ $\psi(k) = \frac{\partial y(k)}{\partial \alpha_1(k)}$ เป็นสัญญาณควบคุม Adaptive ของ $\alpha_1(k)$
 μ_1 เป็น Step Size Parameters

$y(k)$ ในแต่ละครั้งมีน้อย จึงสามารถประมาณสัญญาณควบคุม Adaptive ได้ดังนี้

$$\psi(k-i) = \frac{\partial y(k-i)}{\partial \alpha_1(k-i)} \approx \frac{\partial y(k-i)}{\partial \alpha_1(k)} \quad 2.6$$

เมื่อ i เป็นจำนวนเต็มบวกใดๆ

ดังนั้นจึงเขียนสมการได้ดังนี้

$$\psi(k) = -(1 + \alpha_0)x(k-1) + (1 + \alpha_0)y(k-1) + (1 + \alpha_0)\alpha_1(k)\psi(k-1) - \alpha_0\psi(k-2) \quad 2.7$$

พิจารณาความเร็วในการ Converge ของ $\alpha_1(k)$ โดยการพลอตกราฟ $\alpha_1(k)$ เทียบกับแกนเวลา k จะพบว่า

1. ความละเอียดของค่า μ_1 จะส่งผลต่อการเกิดสัญญาณรบกวน ถ้าค่านี้มีความละเอียดมากสัญญาณรบกวนจะลดลง แต่การ Converge ก็ช้าลงด้วย
2. ความเร็วเชิงมุม ω_0 ที่มีค่ามากจะทำให้การ Converge เร็วกว่าค่า ω_0 ที่มีค่าน้อย
3. ความสูงของสัญญาณ Input ที่เข้ามาก็มีผลต่อการ Converge โดยที่ถ้าความแรงของสัญญาณ Input มีมากก็จะทำให้การ Converge เร็วขึ้นด้วย
4. α_0 ค่ามาก(เข้าใกล้ศูนย์) จะทำให้มีการแกว่งของสัญญาณ แต่การ Converge จะเร็วเมื่อ เข้าใกล้สถานะคงที่ และเมื่อค่า α_0 ค่าน้อย(เข้าใกล้หนึ่ง) การ Converge ในช่วงเริ่มต้นเร็วและแทบไม่มีการแกว่งของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 Adaptive Algorithm โดยใช้ฟิลเตอร์ที่สามารถปรับค่า Q-Factor ได้

การเปลี่ยนแปลงค่าของ $\alpha_1(k)$ จะเร็วขึ้นได้โดยการปรับค่า α_0 ให้เปลี่ยนแปลงไปด้วย วิธีการ Stochastic Gradient เนื่องจากว่า $\alpha_1(k)$ จะติดตามความถี่ได้เร็วหรือไม่ขึ้นอยู่กับค่าของ $y(k)*\psi(k)$ เรียกว่า Update Function และค่าของ $y(k)$ และ $\psi(k)$ ต่างก็มาจาก $H(z)$ ดังนั้น $[y(k)*\psi(k)]^2$ ก็จะเป็นสัดส่วนโดยตรงกับ $\frac{\partial[H(\alpha_1(k))]^2}{\partial\alpha_1(k)}$

เพื่อใช้วิธีการ Stochastic Gradient จะต้องพิสูจน์เพิ่มเติม $\frac{\partial[H(\alpha_1(k))]^2}{\partial\alpha_1(k)}$ เมื่อพลอตกับ α_0 แล้วต้องมียอดสูงสุดของกราฟเพียงยอดเดียวเท่านั้น จึงต้องหาค่าของสมการนี้ เพื่อนำมาเขียนกราฟจาก

$$H(z) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{1 - 2\alpha_1 z^{-1} + z^{-2}}{1 - \alpha_1(1 + \alpha_0)z^{-1} + \alpha_0 z^{-2}} \right)$$

แทนค่า $z = e^{j\omega}$

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{1 - 2\alpha_1(e^{j\omega})^{-1} + (e^{j\omega})^{-2}}{1 - 2\alpha_1(1 + \alpha_0)(e^{j\omega})^{-1} + \alpha_0(e^{j\omega})^{-2}} \right) \quad 2.8$$

คูณด้วย $z = e^{j\omega}$ ทั้งเศษและส่วน

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{(e^{j\omega}) - 2\alpha_1 + (e^{-j\omega})}{(e^{j\omega}) - \alpha_1(1 + \alpha_0) + \alpha_0(e^{-j\omega})} \right) \quad 2.9$$

ใช้สูตร EULER $e^{\pm j\omega} = \cos \omega \pm j \sin \omega$

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{\cos(\omega) + j \sin(\omega) + \cos(\omega) - j \sin(\omega) - 2\alpha_1}{\cos(\omega) + j \sin(\omega) + \alpha_0(\cos(\omega) - j \sin(\omega)) - \alpha_1(1 + \alpha_0)} \right)$$

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{2 \cos(\omega) - 2\alpha_1}{(1 + \alpha_0) \cos(\omega) + (1 - \alpha_0) \sin(\omega) - \alpha_1(1 + \alpha_0)} \right) \quad 2.10$$

$$H(e^{j\omega}) = \left(\frac{\cos(\omega) - \alpha_1}{\cos(\omega) + j \sin\left(\frac{1 - \alpha_0}{1 + \alpha_0}\right) - \alpha_1} \right)$$

หาค่าสัมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}
 |H(e^{j\omega})| &= \sqrt{\frac{(\cos(\omega) - \alpha_1)^2}{\left(\sin(\omega) \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right)\right)^2 + (\cos(\omega) - \alpha_1)^2}} \\
 &= \frac{1}{\sqrt{\left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right)\right)^2 + 1}}
 \end{aligned} \tag{2.11}$$

ยกกำลังสองทั้งสองข้าง

$$|H(e^{j\omega})|^2 = \left[\left\{ \frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right) \right\}^2 + 1 \right]^{-1} \tag{2.12}$$

แต่ $H(\alpha_0)$ มีค่าเท่ากับ

$$\begin{aligned}
 H(\alpha_0) &= \frac{\partial |H(e^{j\omega})|^2}{\partial \alpha_1} \\
 &= (-1) \left[\left\{ \frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right) \right\}^2 + 1 \right]^{-2} \frac{\partial}{\partial \alpha_1} \left[\left\{ \frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right) \right\}^2 + 1 \right] \\
 &= (-1) \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_1}\right)^2 \frac{\partial}{\partial \alpha_1} \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1}\right)^2 \right]}{\left[\left\{ \frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right) \right\}^2 + 1 \right]^2} \\
 &= (-2) \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_1}\right)^2 \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1}\right) \frac{\partial}{\partial \alpha_1} \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1}\right) \right]}{\left[\left\{ \frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right) \right\}^2 + 1 \right]^2} \\
 &= (-2) \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_1}\right)^2 \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1}\right) (-1)(-1)\sin(\omega)(\cos(\omega) - \alpha)^{-2} \right]}{\left[\left\{ \frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \left(\frac{1 - \alpha_0}{1 + \alpha_0}\right) \right\}^2 + 1 \right]^2}
 \end{aligned} \tag{2.13}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$H(\alpha_0) = (-2) \frac{\left[\left(\frac{1-\alpha_0}{1+\alpha_1} \sin(\omega) \right)^2 (\cos(\omega) - \alpha)^{-3} \right]}{\left[\left\{ \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1-\alpha_0}{1+\alpha_0} \right) \right\}^2 + 1 \right]^2} \quad 2.14$$

เมื่อนำไปเขียนกราฟ α_0 เปลี่ยนจาก 0 ถึง 1 พบว่ากราฟที่ได้จะมียอดสูงที่สุดเพียงหนึ่งยอดเท่านั้น ดังนั้นสามารถใช้ Stochastic Gradient เพื่อหาสมการเปลี่ยนแปลง α_0 ได้

Adaptive Algorithm สำหรับ $\alpha_0(k)$ สามารถกำหนดได้ดังสมการ

$$\alpha_0(k+1) = \alpha_0(k) + \frac{\mu_0}{2} \frac{(\partial y(k), \psi(k))^2}{\partial \alpha_0(k)} \quad 2.15$$

$$\alpha_0(k+1) = \alpha_0(k) + \mu_0 y(k) \psi(k) [\psi(k), \psi_{01}(k) + \psi_{02}(k) \cdot y(k)] \quad 2.16$$

โดยที่ $\psi_{01}(k) = \frac{\partial y(k)}{\partial \alpha_0(k)}$ และ $\psi_{02}(k) = \frac{\partial \psi(k)}{\partial \alpha_0(k)}$

ψ_{01}, ψ_{02} : เป็นสัญญาณควบคุมแบบ Adaptive ของ $\alpha_0(k)$

μ_0 : เป็น Step Size Parameter

$$y(k) = \left(\frac{1+\alpha_0}{2} \right) (x(k) - 2\alpha_1(k)x(k-1) + x(k-2)) + \alpha_1(k)(1+\alpha_0)y(k-1) - \alpha_0 y(k-2)$$

$$\psi_{01}(k) = \frac{\partial y(k)}{\partial \alpha_0(k)}$$

$$\begin{aligned} \psi_{01}(k) = & \frac{1}{2} (x(k) - 2\alpha_1(k)x(k-1) + x(k-2)) + \alpha_1(k)y(k-1) - y(k-2) \\ & + \alpha_1(k)(1+\alpha_0(k))\psi_{01}(k-1) - \alpha_0(k)\psi_{01}(k-2) \end{aligned} \quad 2.17$$

$$\psi(k) = -(1+\alpha_0(k))x(k-1) + (1+\alpha_0(k))y(k-1) + (1+\alpha_0(k))\alpha_1(k)\psi(k-1) - \alpha_0(k)\psi(k-2)$$

$$\psi_{02}(k) = \frac{\partial \psi(k)}{\partial \alpha_0(k)}$$

$$\begin{aligned} \psi_{02}(k) = & y(k-1) - x(k-1) + \alpha_1(k)\psi(k-1) - \psi(k-2) + (1+\alpha_0(k))\alpha_1(k)\psi_{01}(k-1) \\ & + (1+\alpha_0(k))\alpha_1(k)\psi_{02}(k-1) - \alpha_0(k)\psi_{02}(k-2) \end{aligned} \quad 2.18$$

เมื่อนำทั้งสองฟังก์ชันไปแทนใน $\alpha_0(k)$ สำหรับค่าของ μ_0 ในฟังก์ชัน $\alpha_0(k)$ จะต้องมีค่ามากกว่าค่า μ_0 ของฟังก์ชัน $\alpha_1(k)$ เพื่อที่จะทำให้การ Converge ของฟังก์ชัน $\alpha_0(k)$ เร็วกว่า $\alpha_1(k)$ แต่ต้องคำนึงถึงเสถียรภาพ เพราะถ้าค่า μ_0 ทั้งสองแตกต่างกันมากเกินไปจะทำให้มี Ripple เกิดขึ้น

เมื่อฟังก์ชัน $\alpha_1(k)$ Converge และค่าของ $\alpha_0(k)$ ที่หาได้จากฟังก์ชันนี้นั้น จะมีค่าอยู่ระหว่าง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$-1 < \alpha_0(k) < 1$ แต่ว่าค่า $\alpha_0(k)$ ที่ใช้กับ IIR Notch Filter สำหรับโครงงานนี้จะมีค่าอยู่ระหว่าง $-1 < \alpha_0(k) < 1$ เท่านั้น จึงต้องมีการควบคุมค่าตัวแปรนี้

จากสมการ $\alpha_0(k+1)$ จะเห็นว่าต้องใช้สัญญาณควบคุมแบบ Adaptive 2 ตัว คือ $\psi_{o1}(k), \psi_{o2}(k)$ และจากสมการ $\psi_{o1}(k)$ ต้องใช้สัญญาณ Input 2 ตัว คือ $x(k), y(k)$ เพื่อสร้างสัญญาณควบคุมแบบ Adaptive $\psi_{o1}(k)$ จากสมการ $\psi_{o2}(k)$ ต้องใช้สัญญาณ Input 3 ตัว คือ $y(k), \psi(k), \psi_{o1}(k)$ เพื่อสร้างสัญญาณควบคุมแบบ Adaptive $\psi_{o2}(k)$ ซึ่งวงจรที่ได้ยังซับซ้อนและยากในการออกแบบ สำหรับ $\alpha_0(k)$ ที่จะทำให้ $\alpha_1(k)$ มีค่าสูงสุดจะหาได้จาก $\alpha_{0,opt} = \frac{\partial H_1(\alpha_0(k))}{\partial \alpha_0(k)} = 0$ จาก

$$H(\alpha_0) = \frac{\partial |H(e^{j\omega})|^2}{\partial \alpha_1(k)} = 2 \cdot \frac{\left\{ \frac{(1-\alpha_0)(\sin(\omega))}{(1+\alpha_0)(\cos(\omega)-\alpha_1(k))} \right\}^2 \left\{ \frac{1}{\cos(\omega)-\alpha_1(k)} \right\}}{\left[1 + \left\{ \frac{(1-\alpha_0)(\sin(\omega))}{(1+\alpha_0)(\cos(\omega)-\alpha_1(k))} \right\}^2 \right]^2} \quad 2.19$$

กำหนดให้ $h(\alpha_0) = \frac{1+\alpha_0}{1-\alpha_0}$, $A = \frac{1}{\sin(\omega)}$ และ $B = \{\cos(\omega) - \alpha_1(k)\}$ จะได้

$$\begin{aligned} H(\alpha_0) &= \frac{2 \cdot \left\{ \frac{1}{h(\alpha_0)^2 A^2 B^3} \right\}}{\left[\left\{ \frac{1}{h(\alpha_0)^2 A^2 B^2} \right\} + 1 \right]^2} \\ &= \frac{2 \cdot \left\{ \frac{1}{h(\alpha_0)^2 A^2 B^3} \right\}}{\left[\frac{1+h(\alpha_0)^2 A^2 B^2}{h(\alpha_0)^2 A^2 B^2} \right]^2} \\ H(\alpha_0) &= \frac{2 \cdot h(\alpha_0)^2 A^2 B}{\left[1+h(\alpha_0)^2 A^2 B^2 \right]^2} \quad 2.20 \end{aligned}$$

ดังนั้น

$$\alpha_{0,opt}(k) = \frac{\partial H(\alpha_0(k))}{\partial (\alpha_0(k))} = 0 \quad 2.21$$

$$\begin{aligned} \frac{\partial}{\partial \alpha_0(k)} \left[\frac{h(\alpha_0)^2 A^2 B}{\left[1+h(\alpha_0)^2 A^2 B^2 \right]^2} \right] &= 0 \\ \alpha_{0,opt}(k) &= 2 \cdot \frac{\left[(1+h(\alpha_0)^2 A^2 B^2)^2 (2 \cdot A^2 B h(\alpha_0) h'(\alpha_0)) - (h(\alpha_0)^2 A^2 B) (4 \cdot A^2 B^2 h(\alpha_0) h'(\alpha_0)) \right]}{\left[1+A^2 B^2 h(\alpha_0)^2 \right]^4} \end{aligned}$$

$$\alpha_{0,opt}(k) = \frac{[4 \cdot A^2 B h(\alpha_0) h'(\alpha_0)] [1 - A^2 B^2 h(\alpha_0)^2]}{\left[1 + A^2 B^2 h(\alpha_0)^2 \right]^4} \quad 2.22$$

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะมีอยู่ 2 กรณีคือ $1 - A^2 B^2 h(\alpha_0)^2 = 0$, $4 \cdot A^2 \cdot B \cdot h(\alpha_0)h'(\alpha_0) = 0$ 2.23

จาก

$$4 \cdot A^2 \cdot B \cdot h(\alpha_0)h'(\alpha_0) = 0$$

$$h(\alpha_0)h'(\alpha_0) = 0$$

$$h'(\alpha_0) = \frac{\partial h(\alpha_0)}{\partial \alpha_0} = \frac{\partial}{\partial \alpha_0} \left(\frac{1 + \alpha_0}{1 - \alpha_0} \right) = \frac{2}{(1 - \alpha_0)^2}$$

$$\therefore h(\alpha_0)h'(\alpha_0) = \frac{2}{(1 - \alpha_0)^2} \cdot \left(\frac{1 + \alpha_0}{1 - \alpha_0} \right) = 0 \quad 2.24$$

จะได้

$$\left[\frac{(1 - \alpha_0^2)}{(1 - \alpha_0)^4} \right] = 0$$

$$1 - \alpha_0^2 = 0$$

$$\therefore \alpha_0 = \pm 1$$

และจาก

$$1 - A^2 B^2 h^2(\alpha_0) = 0$$

$$A^2 B^2 h(\alpha_0) = 1$$

$$A^2 B^2 \left[\frac{1 + \alpha_0}{1 - \alpha_0} \right]^2 = 1$$

$$A^2 B^2 (1 + \alpha_0)^2 = (1 - \alpha_0)^2$$

$$A \cdot B \cdot (1 + \alpha_0) = \pm (1 - \alpha_0)$$

กรณีที่ 1

$$A \cdot B \cdot (1 + \alpha_0) = (1 - \alpha_0)$$

$$A \cdot B + A \cdot B \cdot \alpha_0 = 1 - \alpha_0$$

$$\alpha_0 = \frac{1 - A \cdot B}{1 + A \cdot B}$$

$$A \cdot B = \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}$$

$$1 - \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}$$

$$\alpha_0 = \frac{\sin(\omega)}{1 + \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}$$

$$\therefore \alpha_0 = \frac{\sin(\omega) - \cos(\omega) + \alpha_1(k)}{\sin(\omega) + \cos(\omega) - \alpha_1(k)}$$

2.26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณี ที่ 2

$$A \cdot B \cdot (1 + \alpha_0) = (\alpha_0 - 1)$$

$$A \cdot B + A \cdot B \cdot \alpha_0 = \alpha_0 - 1$$

$$\alpha_0 = \frac{1 + A \cdot B}{1 - A \cdot B}$$

$$A \cdot B = \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}$$

2.27

$$1 + \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}$$

$$\alpha_0 = \frac{\sin(\omega)}{1 - \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}$$

$$\therefore \alpha_0 = \frac{\sin(\omega) + \cos(\omega) - \alpha_1(k)}{\sin(\omega) - \cos(\omega) + \alpha_1(k)}$$

ทดสอบโดยการกำหนดให้ $\alpha_1(k) > \cos(\omega)$ และ $\alpha_1(k) < \cos(\omega)$ พบว่า

$$\alpha_{0,up}(k) = \frac{\sin(\omega) - \cos(\omega) + \alpha_1(k)}{\sin(\omega) + \cos(\omega) - \alpha_1(k)} \quad \text{เมื่อ } \alpha_1(k) < \cos(\omega)$$

$$\alpha_{0,down}(k) = \frac{\sin(\omega) + \cos(\omega) - \alpha_1(k)}{\sin(\omega) - \cos(\omega) + \alpha_1(k)} \quad \text{เมื่อ } \alpha_1(k) > \cos(\omega)$$

ลักษณะฟังก์ชันของ $\psi(k), \psi_{01}(k), \psi_{02}(k)$ จะขึ้นอยู่กับฟังก์ชันหลายตัว ไม่ขึ้นกับฟังก์ชันตัวใดตัวหนึ่ง ถ้าสามารถเปลี่ยนฟังก์ชันเหล่านี้ให้ขึ้นอยู่กับฟังก์ชันตัวใดตัวหนึ่งได้ ก็จะทำให้วงจรมีความง่ายขึ้น ซึ่งทำได้โดยการหา Transfer Function ของฟังก์ชันเหล่านี้

Transfer Function $\psi(k)$ เขียนสัญลักษณ์ด้วย $G(z)$ ในรูป Z-Transform หาได้จาก

Transfer Function ของ $\psi(k)$ คือ $G(z)$

$$\psi(k) = -(1 + \alpha_0(k))x(k-1) + (1 + \alpha_0(k))y(k-1) + (1 + \alpha_0(k))\alpha_1(k)\psi(k-1) - \alpha_0(k)\psi(k-2) \quad 2.28$$

Take Z-Transform

$$\psi(z) = -(1 + \alpha_0(k))X(z)Z^{-1} + (1 + \alpha_0(k))Y(z)Z^{-1} + (1 + \alpha_0(k))\alpha_1(k)\psi(z)Z^{-1} - \alpha_0(k)\psi(z)Z^{-2}$$

$$G(z) = \frac{\psi(z)}{Y(z)} (1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}) = \left((1 + \alpha_0) - (1 + \alpha_0) \frac{X(z)}{Y(z)} \right) Z^{-1}$$

เปลี่ยน $X(z)$ ให้อยู่ในรูปของ $Y(z)$

$$\frac{X(z)}{Y(z)} = \left(\frac{2}{1 + \alpha_0} \right) \left(\frac{1 - \alpha_1(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}}{1 - 2\alpha_1 Z^{-1} + Z^{-2}} \right) \quad 2.29$$

แทนค่าในสมการ $G(z)$

$$G(z) = \frac{\psi(z)}{Y(z)} (1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}) = (1 + \alpha_0) \left(1 - \left(\frac{2}{1 + \alpha_0} \right) \left(\frac{1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}}{1 - 2\alpha_1(k)Z^{-1} + Z^{-2}} \right) \right) Z^{-1}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า
จะได้ $G(z)$ ดังนี้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$G(z) = \frac{\psi(z)}{Y(z)} = \left(\left(\frac{(1 + \alpha_0)Z^{-1}}{(1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2})} \right) - \left(\frac{2Z^{-1}}{1 - 2\alpha_1(k)Z^{-1} + Z^{-2}} \right) \right)$$

การนำไปเขียนโปรแกรมคอมพิวเตอร์ จะต้องเป็นโครงสร้างแบบตรง I จะได้

$$G(z) = \frac{\psi(z)}{Y(z)} = \left(\frac{(\alpha_0 - 1)(Z^{-1} - Z^{-3})}{(1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2})(1 - 2\alpha_1(k)Z^{-1} + Z^{-2})} \right)$$

$$G(z) = \frac{\psi(z)}{Y(z)} = \left(\frac{(\alpha_0 - 1)(Z^{-1} - Z^{-3})}{(1 - \alpha_1(k)(3 + \alpha_0)Z^{-1} + (1 + 2(1 + \alpha_0)\alpha_1^2(k) + \alpha_0)Z^{-2} - \alpha_1(k)(1 + 3\alpha_0)Z^{-3} + \alpha_0 Z^{-4})} \right)$$

$$G_{01}(z)$$

Transfer Function ของ $\psi_{01}(k)$ คือ $G_{01}(z)$

$$\psi_{01}(k) = \frac{1}{2} \cdot [x(k) - 2\alpha x(k-1) + x(k+2)] + \alpha_1(k)y(k-1) - y(k-2) + (1 + \alpha_0)\alpha_1(k)\psi_{01}(k-1) - \alpha_0\psi_{01}(k-2) \quad 2.30$$

เปลี่ยน $x(k)$ ให้อยู่ในรูป $y(k)$ จากสมการ $y(k)$ จะได้

$$\frac{1}{2}[x(k) - 2\alpha_1(k)x(k-1) + x(k-2)] = \frac{y(k) - (1 + \alpha_0)\alpha_1(k)y(k-1) + \alpha_0 y(k-2)}{(1 + \alpha_0)} \quad 2.31$$

แทนค่า

$$\psi_{01}(k) = y(k) - (1 + \alpha_0)\alpha_1(k)y(k-1) + \alpha_0 y(k-2) + \alpha_1(k)y(k-1) - y(k-2) + (1 + \alpha_0)\alpha_1(k)\psi_{01}(k-1) - \alpha_0\psi_{01}(k-2)$$

$$\psi_{01}(k) - (1 + \alpha_0)\alpha_1(k)\psi_{01}(k-1) + \alpha_0\psi_{01}(k-2) = \frac{y(k)}{(1 + \alpha_0)} - \alpha_1 y(k-1) + \frac{\alpha_0 y(k-2)}{(1 + \alpha_0)} + \alpha_1(k)y(k-1) - y(k-2)$$

Take Z-Transform

$$\psi_{01}(z)[1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2}] = \frac{y(z)}{(1 + \alpha_0)} + \frac{\alpha_0 y(z)z^{-2}}{(1 + \alpha_0)} - y(z)z^{-2}$$

$$\psi_{01}(z)[1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2}] = \frac{(1 - z^{-2})y(z)}{(1 + \alpha_0)}$$

$$G_{01}(z) = \frac{\psi_{01}(z)}{y(z)} = \left(\frac{1}{(1 + \alpha_0)} \right) \left(\frac{1 - z^{-2}}{1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2}} \right)$$

นอกจากนี้เรายังสามารถทำให้ ขึ้นอยู่กับตัวแปรเพียงตัวแปรเดียวได้ดังนี้

$$\frac{\psi_{01}(z)}{y(z)} = \left[\frac{(\alpha_0 - 1)(z^{-1} - z^{-3})}{(1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2})(1 - 2\alpha_1(k)z^{-1} + z^{-2})} \right]$$

$$y(z) = \frac{[1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2}][1 - 2\alpha_1(k)z^{-1} + z^{-2}]}{z^{-1}(\alpha_0 - 1)(1 - z^{-2})} \quad 2.32$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 แทนค่าใน $G_{01}(z)$
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$G_{01}(z) = \frac{\psi_{01}(z)}{\psi(z)} = \frac{1}{(1+\alpha_0)} \left(\frac{1-z^{-2}}{1-(1+\alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2}} \right) \left(\frac{(1-(1+\alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2})(1-2\alpha_1(k)z^{-1} + z^{-2})}{z^{-1}(\alpha_0-1)(1-z^{-2})} \right)$$

$$G_{01}(z) = \frac{\psi_{01}(z)}{\psi(z)} = \frac{1}{(\alpha_0^2-1)} \left(\frac{1-2\alpha_1(k)z^{-1} + z^{-2}}{z^{-1}} \right)$$

$$G_{02}(z)$$

Transfer Function ของ $\psi_{02}(k)$ คือ $G_{02}(z)$

$$\begin{aligned} \psi_{02}(z) &= y(k-1) - x(k-1) + \alpha_1(k)\psi(k-1) - \psi(k-2) + (1+\alpha_0)\psi_{01}(k-1) \\ &+ (1+\alpha_0)\alpha_1(k)\psi_{02}(k-1) - \alpha_0\psi_{02}(k-2) + (1+\alpha_0)\alpha_1(k)\psi_{02}(k-2) - \alpha_0\psi_{02}(k-2) \end{aligned} \quad 2.33$$

เปลี่ยน $y(k-1) - x(k-1)$ ให้อยู่ในรูป

$$\psi_{02}(k) = \frac{\psi(k) - (1+\alpha_0)\alpha_1(k)\psi(k-1) + \alpha_0\psi(k-2)}{(1+\alpha_0)} + \alpha_0\psi(k-1) - \psi(k-2) + (1+\alpha_0)\psi_{01}(k-1)$$

แทนค่าในสมการ

$$y(k-1) - x(k-1) = \frac{\psi(k) - (1+\alpha_0)\alpha_1(k)\psi(k-1) + \alpha_0\psi(k-2)}{(1+\alpha_0)} \quad 2.34$$

Take Z-Transform

$$\begin{aligned} \psi_{02}(z) &= \frac{\psi(z)}{(1+\alpha_0)} - \alpha_1(k)\psi(z)z^{-1} + \frac{\alpha_0\psi(z)z^{-2}}{(1+\alpha_0)} + \alpha_1(k)\psi(z)z^{-1} - \psi(z)z^{-2} + (1+\alpha_0)\psi_{01}(z)z^{-1} \\ &+ (1+\alpha_0)\alpha_1(k)\psi_{02}(z)z^{-1} - \alpha_0\psi_{02}(z)z^{-2} - \alpha_0\psi_{02}(z)z^{-2} \end{aligned}$$

$$\psi_{02}(z) - (1+\alpha_0)\alpha_1(k)\psi_{02}(z)z^{-1} + \alpha_0\psi_{02}(z)z^{-2} = \frac{\psi(z)}{(1+\alpha_0)} - \frac{\psi(z)z^{-2}}{(1+\alpha_0)} + (1+\alpha_0)\psi_{01}(z)z^{-1}$$

เปลี่ยนให้อยู่ในรูป $G_{01}(z)$

$$\psi_{01}(z) = \frac{\psi(z)}{(1+\alpha_0)} \left(\frac{1-2\alpha_1(k)z^{-1} + z^{-2}}{z^{-1}} \right) \quad 2.35$$

แทนค่า

$$\frac{\psi_{02}(z)}{\psi(z)} (1 - (1+\alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2}) = \frac{1}{(1+\alpha_0)} \left(1 - z^{-2} + \frac{(1+\alpha_0)^2}{(\alpha_0^2-1)} (1 - 2\alpha_1(k)z^{-1} + z^{-2}) \right)$$

$$\frac{\psi_{02}(z)}{\psi(z)} (1 - (1+\alpha_0)\alpha_1(k)z^{-1} + \alpha_0 z^{-2}) = \frac{((\alpha_0^2-1) - (\alpha_0^2-1)z^{-2} + (1+2\alpha_0 + \alpha_0^2)(1 - 2\alpha_1(k)z^{-1} + z^{-2}))}{(1+\alpha_0)(\alpha_0^2-1)}$$

$$\text{แก้สมการ } ((\alpha_0^2-1) - (\alpha_0^2-1)z^{-2} + (1+2\alpha_0 + \alpha_0^2)(1 - 2\alpha_1(k)z^{-1} + z^{-2}))$$

$$= (1 - 2\alpha_1(k)z^{-1} + z^{-2} + 2\alpha_0 - 4\alpha_0\alpha_1(k)z^{-1} + 2\alpha_0 z^{-2} + \alpha_0^2 - 2\alpha_0^2\alpha_1(k)z^{-1} + \alpha_0^2 z^{-2}) + (\alpha_0^2-1) - (\alpha_0^2-1)z^{-2}$$

$$= (2\alpha_0 + 2\alpha_0^2) + (-2\alpha_1(k) - 4\alpha_0\alpha_1(k) - 2\alpha_1(k)\alpha_0^2)z^{-1} + (1 + 2\alpha_0 + \alpha_0^2 - (\alpha_0^2-1))z^{-2}$$

$$= 2\alpha_0(1+\alpha_0) - 2\alpha_1(k)(1-2\alpha_0 + \alpha_0^2)z^{-1} + (2+2\alpha_0)z^{-2}$$

$$= 2(1+\alpha_0)(\alpha_0 - \alpha_1(k)(1+\alpha_0)z^{-1} + z^{-2})$$

เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาตจากทางมหาวิทยาลัย
 งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่า **แทนค่า** ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{\psi_{02}(z)}{\psi(z)}(1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0z^{-2}) = \frac{2}{(\alpha_0^2 - 1)}(\alpha_0 - \alpha_1(k)(1 + \alpha_0)z^{-1} + z^{-2})$$

$$\psi_{02}(z)(1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0z^{-2}) = \frac{2}{(\alpha_0^2 - 1)}(\alpha_0 - \alpha_1(k)(1 + \alpha_0)z^{-1} + z^{-2})\psi(z)$$

$$G_{02}(z) = \frac{\psi_{02}(z)}{\psi(z)} = \left(\frac{2(\alpha_0 - \alpha_1(k)(1 + \alpha_0)z^{-1} + z^{-2})}{(\alpha_0^2 - 1)(1 - (1 + \alpha_0)\alpha_1(k)z^{-1} + \alpha_0z^{-2})} \right)$$

จาก $G(z)$, $G_{01}(z)$, $G_{02}(z)$ นำไปเขียนเป็น Algorithm ได้สมการดังนี้

$$y(k) = \left(\frac{1 + \alpha_0}{2} \right) (x(k) - 2\alpha_1(k)x(k-1) + x(k-2)) + \alpha_1(k)(1 + \alpha_0)y(k-1) - \alpha_0y(k-2)$$

$$\frac{\psi(z)}{Y(z)} = \left(\left(\frac{(1 + \alpha_0)Z^{-1}}{(1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0Z^{-2})} \right) - \left(\frac{2Z^{-1}}{1 - 2\alpha_1(k)Z^{-1} + Z^{-2}} \right) \right)$$

$$\psi(k) = (\alpha_0 - 1)(y(k-1) - y(k-3)) + \alpha_1(k)(3 + \alpha_0)\psi(k-1) - (1 + 2(1 + \alpha_0)\alpha_1^2(k) + \alpha_0)\psi(k-2) + \alpha_1(k)(1 + 3\alpha_0)\psi(k-3) - \alpha_0\psi(k-4)$$

$$\alpha_1(k+1) = \alpha_1(k) - \mu_0 y(k)\psi(k)$$

$$\psi_{01}(k) = \left(\frac{1}{1 + \alpha_0(k)} \right) (y(k) - y(k-2)) + \alpha_1(k)(1 + \alpha_0(k))\psi_{01}(k-1) - \alpha_0\psi_{01}(k-2)$$

$$\psi_{02}(k) = (\psi(k) - \alpha_1(k)(1 + \alpha_0(k))\psi(k-1) + \alpha_0\psi(k-2) + \alpha_0(1 + \alpha_0)\psi(k-1) - (1 + \alpha_0)\psi(k-2) + (1 + \alpha_0)^2\psi_{01}(k-1))$$

$$\alpha_0(k+1) = \alpha_0(k) + \mu_0 y(k)\psi(k)[\psi(k)\psi_{01}(k) + \psi_{02}(k)y(k)]$$

2.3 Simplified Adaptive Algorithm

จากหัวข้อที่ 2 เราจะทำการแก้ปัญหาดังกล่าวโดยการตัดแปลงและปรับปรุงสมการใหม่ โดยพิจารณาสิ่งดังต่อไปนี้

ค่าเฉลี่ยกำลังสองของ $[y(k)]^2$ เป็นสัดส่วนโดยตรงกับค่าของ $|H(\alpha_1(k))|^2$ ซึ่งเป็นฟังก์ชันของ α_0 จะได้สมการดังต่อไปนี้

จาก Transfer Function ของวงจกรองความถี่แบบ IIR Notch Filter อันดับสองที่มีความถี่กลางเป็น ω_0 เขียนเป็นสมการดังต่อไปนี้

$$H(z) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{1 - 2\alpha_1 Z^{-1} + Z^{-2}}{1 - \alpha_1(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2}} \right)$$

แทนค่า $Z = e^{j\omega}$ เป็นส่วนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรรมใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แทนค่า $Z = e^{j\omega}$

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{1 - 2\alpha_1(e^{j\omega})^{-1} + (e^{j\omega})^{-2}}{1 - 2\alpha_1(1 + \alpha_0)(e^{j\omega})^{-1} + \alpha_0(e^{j\omega})^{-2}} \right)$$

คูณด้วย $e^{j\omega}$ ทั้งเศษและส่วน

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{(e^{j\omega}) - 2\alpha_1 + (e^{-j\omega})}{(e^{j\omega}) - \alpha_1(1 + \alpha_0) + \alpha_0(e^{-j\omega})} \right)$$

ใช้สูตร Euler 's Formula $e^{\pm j\omega} = \cos \omega \pm j \sin \omega$

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{\cos(\omega) + j \sin(\omega) + \cos(\omega) - j \sin(\omega) - 2\alpha_1}{\cos(\omega) + j \sin(\omega) + \alpha_0(\cos(\omega) - j \sin(\omega)) - \alpha_1(1 + \alpha_0)} \right)$$

$$H(e^{j\omega}) = \left(\frac{1 + \alpha_0}{2} \right) \left(\frac{2 \cos(\omega) - 2\alpha_1}{(1 + \alpha_0) \cos(\omega) + (1 - \alpha_0) \sin(\omega) - \alpha_1(1 + \alpha_0)} \right)$$

$$H(e^{j\omega}) = \frac{\cos(\omega) - \alpha_1}{\cos(\omega) + j \sin(\omega) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) - \alpha_1}$$

หาค่าสัมบูรณ์

$$|H(e^{j\omega})| = \frac{(\cos(\omega) - \alpha_1)^2}{\left(\sin(\omega) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right)^2 + (\cos(\omega) - \alpha_1)^2}$$

$$\therefore |H(e^{j\omega})| = \frac{1}{\sqrt{\left(\left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right)^2 + 1}}$$

คุณลักษณะของ $|H(e^{j\omega})|$ เทียบกับ α_0 จากการพลอตกราฟ จะเห็นได้ว่ากราฟไม่มีขด ดังนั้นค่าสูงสุดของค่าเฉลี่ยของ $y(k) * \psi(k)$ สามารถประมาณให้เท่ากับค่าเฉลี่ยของ $\psi(k)$ ได้ ดังนั้นค่าเฉลี่ยของ $\psi(k)$ เป็นสัดส่วนโดยตรงกับ $\frac{\partial H(\alpha_1(k))}{\partial \alpha_1(k)}$ ซึ่งจะนำไปตามสมการดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}
\frac{\partial |H(e^{j\omega})|}{\partial \alpha_1(k)} &= \frac{\partial}{\partial \alpha_1(k)} \left[\left\{ \left(\frac{\sin \omega}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right]^{\frac{1}{2}} \\
&= \left(-\frac{1}{2} \right) \left[\left\{ \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right]^{-\frac{3}{2}} \frac{\partial}{\partial \alpha_1} \left[\left\{ \left(\frac{\sin \omega}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right] \\
&= \left(-\frac{1}{2} \right) \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_0} \right)^2 \frac{\partial}{\partial \alpha_1} \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right)^2 \right]}{\left[\left\{ \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right]^{\frac{3}{2}}} \\
&= (-1) \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_0} \right)^2 \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \frac{\partial}{\partial \alpha_1} \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \right]}{\left[\left\{ \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right]^{\frac{3}{2}}} \\
&= (-1) \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_1} \right)^2 \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) (-1)(-1) \sin(\omega) (\cos(\omega) - \alpha)^{-2} \right]}{\left[\left\{ \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right]^{\frac{3}{2}}} \\
&= (-1) \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_0} \right)^2 \sin(\omega) (\cos(\omega) - \alpha)^{-3} \right]}{\left[\left\{ \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right]^{\frac{3}{2}}} \\
\left| \frac{\partial |H(e^{j\omega})|}{\partial \alpha_1(k)} \right|^2 &= \frac{\left[\left(\frac{1 - \alpha_0}{1 + \alpha_0} \right)^4 \sin(\omega)^2 (\cos(\omega) - \alpha)^{-6} \right]}{\left[\left\{ \left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1} \right) \left(\frac{1 - \alpha_0}{1 + \alpha_0} \right) \right\}^2 + 1 \right]^3}
\end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}
&= \left[\frac{\left(\left(\frac{1-\alpha_0}{1+\alpha_0} \right) \sin(\omega) \right)^4 (\cos(\omega)-\alpha)^{-6}}{(\cos(\omega)-\alpha_1)^{-6} \left[\left\{ \left(\sin(\omega) \right) \left(\frac{1-\alpha_0}{1+\alpha_0} \right) \right\}^2 + (\cos(\omega)-\alpha_1)^2 \right]^3} \right] \\
&= \left[\frac{\left(\left(\frac{1-\alpha_0}{1+\alpha_0} \right) \sin(\omega) \right)^4}{\left[\left\{ \left(\sin(\omega) \right) \left(\frac{1-\alpha_0}{1+\alpha_0} \right) \right\}^2 + (\cos(\omega)-\alpha_1)^2 \right]^3} \right] \\
\frac{\partial H(\alpha_1(k))}{\partial \alpha_1(k)} &= \frac{\left(\left(\frac{1-\alpha_0}{1+\alpha_0} \right) \cdot \sin(\omega) \right)^2}{\left[\left\{ \left(\sin(\omega) \right) \cdot \left(\frac{1-\alpha_0}{1+\alpha_0} \right) \right\}^2 + (\cos(\omega)-\alpha_1) \right]^{\frac{3}{2}}} \quad 2.36
\end{aligned}$$

จากสมการซึ่งเป็นฟังก์ชันของ α สามารถเขียนใหม่ให้อยู่ในรูปที่ง่ายขึ้น ได้ดังสมการ

$$\Pi(\alpha_0) = \frac{\partial H(\alpha_1(k))}{\partial \alpha_1(k)} \quad 2.37$$

คุณลักษณะของ $|\Pi(\alpha_0)|$ เทียบกับ α_0 จากการพลอตจะเห็นได้ว่า $|\Pi(\alpha_0)|$ มียอดเดียว เหมือนกับวงจรรองความถี่กลางผ่าน ดังนั้นสามารถตัดแปลงหรือปรับปรุง Algorithm สำหรับการ Update ของ $\alpha_0(k)$ ขึ้นใหม่ได้ดังสมการ

$$\alpha_0(k+1) = \alpha_0(k) + \frac{\mu_0}{2} \frac{\partial \psi(k)^2}{\partial \alpha_{0(k)}} \quad 2.38$$

$$\alpha_0(k+1) = \alpha_0(k) + \mu_0 \psi(k) \psi_{02}(k) \quad 2.39$$

$$\text{โดยที่ } \psi_{02}(k) = \frac{\partial \psi(k)}{\partial \alpha_0(k)}$$

μ_0 : เป็น Step Size Parameter

และ $\psi_{02}(k)$ เป็นสัญญาณควบคุมแบบ Adaptive ของ $\alpha_0(k)$ จะเห็นได้ว่าใช้เพียงตัวเดียวเท่านั้น จากสมการ $\psi_{02}(k)$ จะเห็นว่าสัญญาณควบคุมแบบ Adaptive $\psi_{02}(k)$ ต้องการสัญญาณ Input $\psi(k)$ เพียงตัวเดียวเท่านั้น ดังนั้นโครงสร้างวงจรจะง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า สำหรับค่า $\alpha_0(k)$ ที่จะทำให้ $\alpha_1(k)$ มีค่าสูงสุดถูกแทนด้วย $\alpha_{0, \text{opt}}(k)$ ซึ่งสามารถหาได้จาก ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{สมการ } \alpha_{0, \text{opt}(k)} = \frac{\partial H(\alpha_0(k))}{\partial \alpha_0(k)} = 0 \text{ จาก}$$

$$|\Pi(\alpha_0)| = \left| \frac{\partial |H(\alpha_1(k))|}{\partial \alpha_1(k)} \right|^2$$

$$= \frac{\left(\left(\frac{1-\alpha_0}{1+\alpha_0} \right) \sin(\omega) \right)^4 (\cos(\omega) - \alpha_1(k))^{-6}}{\left\{ \left[\left(\frac{\sin(\omega)}{\cos(\omega) - \alpha_1(k)} \right) \left(\frac{1-\alpha_0}{1+\alpha_0} \right) \right]^2 + 1 \right\}^3}$$

กำหนดให้ $h(\alpha_0) = \frac{1+\alpha_0}{1-\alpha_0}$, $A = \frac{1}{\sin(\omega)}$ และ $B = \{\cos(\omega) - \alpha_1(k)\}$ จะได้

$$|\Pi(\alpha_0)| = \frac{1}{A^4 \cdot B^6 h^4(\alpha_0)} \left\{ \left[\frac{1}{A^2 B^2 h^2(\alpha_0)} \right] + 1 \right\}^3$$

2.40

$$\frac{\partial}{\partial \alpha_0(k)} \left[\frac{1}{A^4 B^6 h^4(\alpha_0)} \left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1 \right)^3 \right] = 0$$

$$= \frac{\left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1 \right)^2}{\left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1 \right)^6} \left[\left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1 \right) \left(\frac{-4A^4 B^6 h^3(\alpha_0) h'(\alpha_0)}{A^8 B^{12} h^8(\alpha_0)} \right) - 6 \left(\frac{1}{A^4 B^6 h^4(\alpha_0)} \right) \left(\frac{-2A^2 B^2 h(\alpha_0) h'(\alpha_0)}{A^4 B^4 h^4(\alpha_0)} \right) \right]$$

$$= \frac{1}{\left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1 \right)^4} \left[\left(\frac{1+A^2 B^2 h^2(\alpha_0)}{A^2 B^2 h^2(\alpha_0)} \right) \left(\frac{-4h'(\alpha_0)}{A^4 B^8 h^5(\alpha_0)} \right) + 5 \left(\frac{1}{A^4 B^6 h^4(\alpha_0)} \right) \left(\frac{h'(\alpha_0)}{A^4 B^4 h^4(\alpha_0)} \right) \right]$$

$$= \frac{1}{\left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1 \right)^4} \left[\left(\frac{-4h'(\alpha_0) - 4A^2 B^2 h^2(\alpha_0) h'(\alpha_0)}{A^6 B^8 h^7(\alpha_0)} \right) + \left(\frac{6h'(\alpha_0)}{A^6 B^8 h^7(\alpha_0)} \right) \right]$$

$$= \frac{1}{\left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1 \right)^4} \left[\frac{2h'(\alpha_0) - 4A^2 B^2 h^2(\alpha_0) h'(\alpha_0)}{A^6 B^8 h^7(\alpha_0)} \right]$$

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= \frac{1}{\left(\frac{1}{A^2 B^2 h^2(\alpha_0)} + 1\right)^4} \left[\frac{2h'(\alpha_0)(1 - 2A^2 B^2 h^2(\alpha_0))}{A^6 B^8 h^7(\alpha_0)} \right]$$

$$= \frac{1}{\left(\frac{1 + A^2 B^2 h^2(\alpha_0)}{A^2 B^2 h^2(\alpha_0)}\right)^4} \left[\frac{2h'(\alpha_0)(1 - 2A^2 B^2 h^2(\alpha_0))}{A^6 B^8 h^7(\alpha_0)} \right]$$

$$= \frac{1}{\frac{(1 + A^2 B^2 h^2(\alpha_0))^4}{A^8 B^8 h^8(\alpha_0)}} \left[\frac{2h'(\alpha_0)(1 - 2A^2 B^2 h^2(\alpha_0))}{A^6 B^8 h^7(\alpha_0)} \right]$$

$$|\Pi(\alpha_0)| = \frac{(2A^2 h(\alpha_0) h'(\alpha_0))(1 - 2A^2 B^2 h^2(\alpha_0))}{(1 + A^2 B^2 h^2(\alpha_0))^4} = 0 \quad 2.41$$

จะมีอยู่ 2 กรณี คือ

$$2A^2 h(\alpha_0) h'(\alpha_0) = 0 \quad 2.42$$

$$1 - 2A^2 B^2 h^2(\alpha_0) = 0 \quad 2.43$$

จาก

$$2 \cdot A^2 \cdot h(\alpha_0) h'(\alpha_0) = 0$$

$$h(\alpha_0) h'(\alpha_0) = 0$$

$$h'(\alpha_0) = \frac{\partial h(\alpha_0)}{\partial \alpha_0} = \frac{\partial}{\partial \alpha_0} \left(\frac{1 + \alpha_0}{1 - \alpha_0} \right) = \frac{2}{(1 - \alpha_0)^2}$$

จะได้

$$h(\alpha_0) h'(\alpha_0) = \frac{2}{(1 - \alpha_0)^2} \cdot \left(\frac{1 + \alpha_0}{1 - \alpha_0} \right) = 0$$

$$\left[\frac{(1 - \alpha_0^2)}{(1 - \alpha_0)^4} \right] = 0$$

$$\therefore \alpha_0 = \pm 1$$

และจาก

$$1 - 2A^2 B^2 h^2(\alpha_0) = 0$$

$$2A^2 B^2 h^2(\alpha_0) = 1$$

$$A^2 B^2 \left[\frac{1 + \alpha_0}{1 - \alpha_0} \right]^2 = \frac{1}{2} \quad 2.44$$

$$A^2 B^2 (1 + \alpha_0)^2 = \frac{1}{2} (1 - \alpha_0)^2$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ A · B · (1 + α₀) = ± $\frac{1}{\sqrt{2}}$ (1 - α₀) นั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 1

$$A \cdot B \cdot (1 + \alpha_0) = \frac{1}{\sqrt{2}} (1 - \alpha_0)$$

$$A \cdot B + A \cdot B \cdot \alpha_0 = \frac{1}{\sqrt{2}} - \frac{\alpha_0}{\sqrt{2}}$$

$$\alpha_0 = \frac{\frac{1}{\sqrt{2}} - A \cdot B}{\frac{1}{\sqrt{2}} + A \cdot B}$$

$$A \cdot B = \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)} \quad 2.45$$

$$\alpha_0 = \frac{\frac{1}{\sqrt{2}} - \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}{\frac{1}{\sqrt{2}} + \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}$$

$$\therefore \alpha_0 = \frac{\sin(\omega) - \sqrt{2} \cos(\omega) + \sqrt{2} \alpha_1(k)}{\sin(\omega) + \sqrt{2} \cos(\omega) - \sqrt{2} \alpha_1(k)}$$

กรณีที่ 2

$$A \cdot B \cdot (1 + \alpha_0) = \frac{1}{\sqrt{2}} (\alpha_0 - 1)$$

$$A \cdot B + A \cdot B \cdot \alpha_0 = \frac{\alpha_0}{\sqrt{2}} - \frac{1}{\sqrt{2}}$$

$$\alpha_0 = \frac{\frac{1}{\sqrt{2}} + A \cdot B}{\frac{1}{\sqrt{2}} - A \cdot B}$$

$$A \cdot B = \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)} \quad 2.47$$

$$\alpha_0 = \frac{\frac{1}{\sqrt{2}} + \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}{\frac{1}{\sqrt{2}} - \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}$$

$$\therefore \alpha_0 = \frac{\sin(\omega) + \sqrt{2} \cos(\omega) - \sqrt{2} \alpha_1(k)}{\sin(\omega) - \sqrt{2} \cos(\omega) + \sqrt{2} \alpha_1(k)}$$

ทดสอบโดยการกำหนดให้ $\alpha_1(k) > \cos(\omega)$ และ $\alpha_1(k) < \cos(\omega)$ พบว่า

$$\alpha_{0,opt}(k) = \frac{\sin(\omega) - \sqrt{2} (\cos(\omega) + \sqrt{2} \alpha_1(k))}{\sin(\omega) + \sqrt{2} (\cos(\omega) - \sqrt{2} \alpha_1(k))} \quad \text{เมื่อ } \alpha_1(k) < \cos(\omega)$$

$$\alpha_{0,opt}(k) = \frac{\sin(\omega) + \sqrt{2} (\cos(\omega) - \sqrt{2} \alpha_1(k))}{\sin(\omega) - \sqrt{2} (\cos(\omega) + \sqrt{2} \alpha_1(k))} \quad \text{เมื่อ } \alpha_1(k) > \cos(\omega)$$

เอกสารนี้เป็นเอกสารที่สํานักงานเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและข้อมูลอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการที่นำมาออกแบบวงจรเป็นวงจร Adaptive Algorithm ที่ใช้สำหรับตีเทคต์สัญญาณ
ขาอินพุตเดี่ยว

$$y(k) = \left(\frac{1 + \alpha_0}{2} \right) (x(k) - 2\alpha_1(k)x(k-1) + x(k-2)) + \alpha_1(k)(1 + \alpha_0)y(k-1) - \alpha_0 y(k-2)$$

$$\frac{\psi(z)}{Y(z)} = \left(\left(\frac{(1 + \alpha_0)Z^{-1}}{(1 - \alpha_1(k)(1 + \alpha_0)Z^{-1} + \alpha_0 Z^{-2})} \right) - \left(\frac{2Z^{-1}}{1 - 2\alpha_1(k)Z^{-1} + Z^{-2}} \right) \right)$$

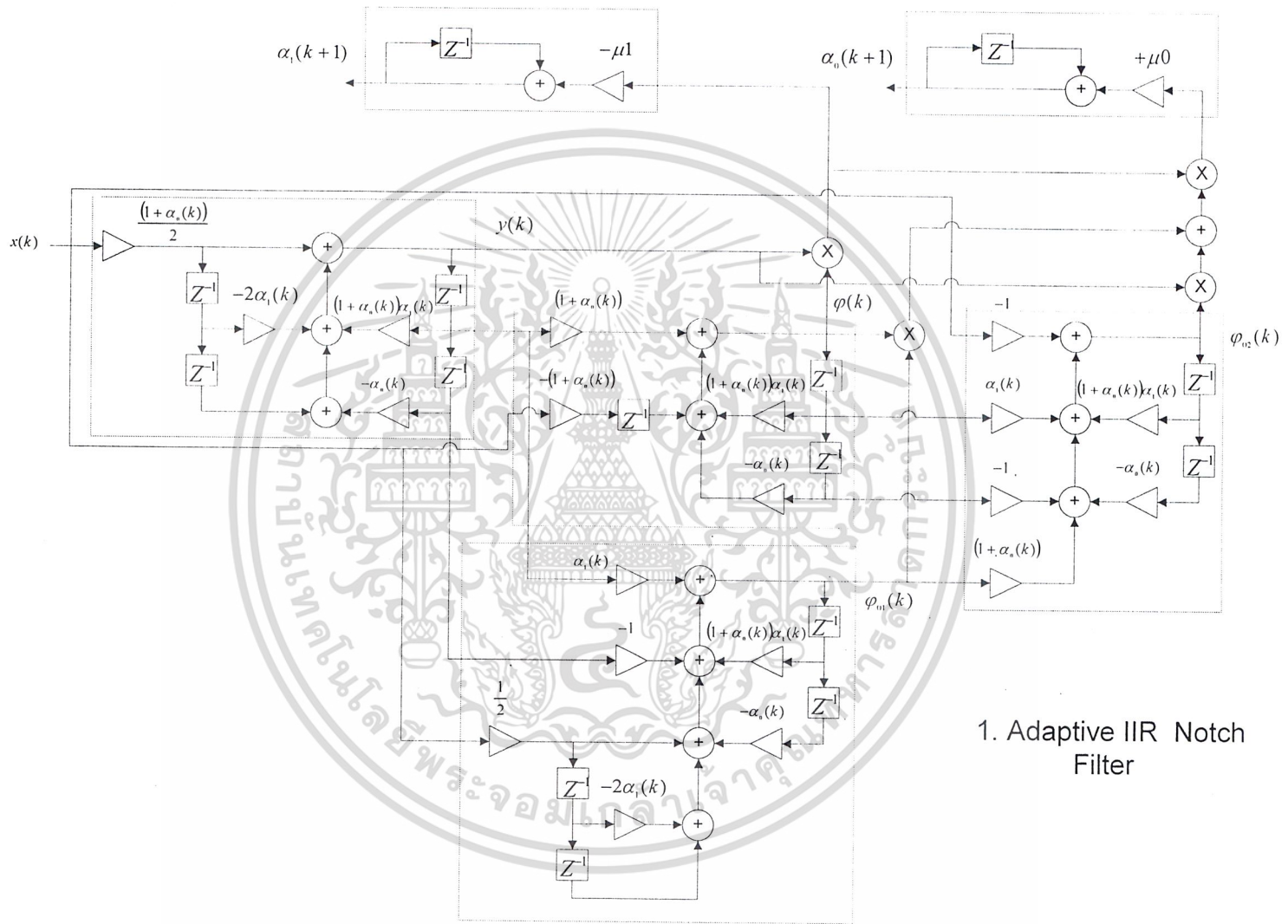
$$\psi(k) = (\alpha_0 - 1)(y(k-1) - y(k-3)) + \alpha_1(k)(3 + \alpha_0)\psi(k-1) - (1 + 2(1 + \alpha_0)\alpha_1^2(k) + \alpha_0)\psi(k-2) \\ + \alpha_1(k)(1 + 3\alpha_0)\psi(k-3) - \alpha_0\psi(k-4)$$

$$\alpha_1(k+1) = \alpha_1(k) - \mu_0 y(k)\psi(k)$$

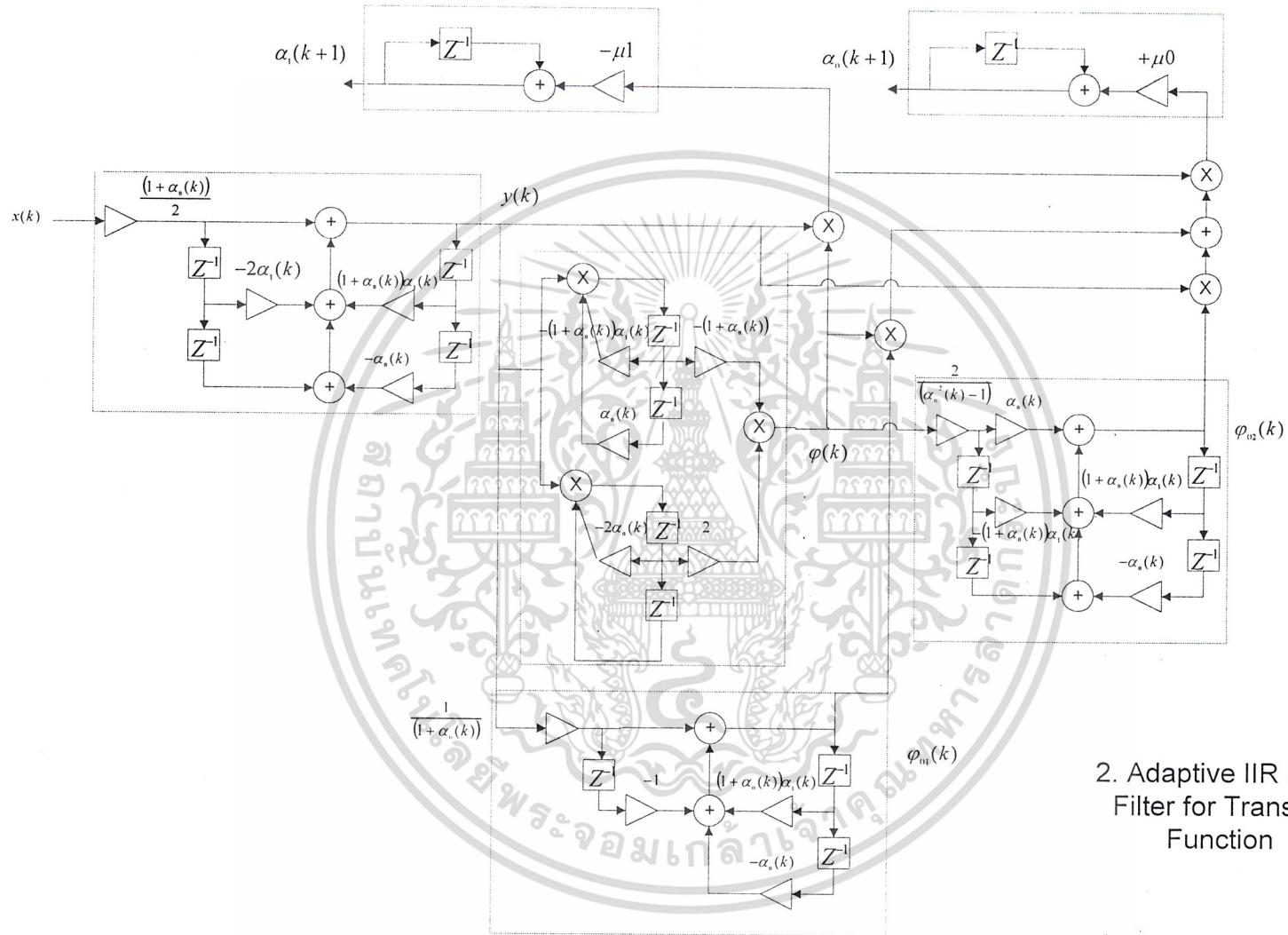
$$\psi_{02}(k) = \left(\frac{2}{\alpha_0^2(k) - 1} \right) (\alpha_0\psi(k) - \alpha_1(k)(1 + \alpha_0(k))\psi(k-1) + \psi(k-2)) + (1 + \alpha_0(k))\alpha_1(k)\psi_{02}(k-1) - \alpha_0(k)\psi_{02}(k)$$

$$\alpha_0(k+1) = \alpha_0(k) + \mu_0 \psi(k)\psi_{02}(k)$$

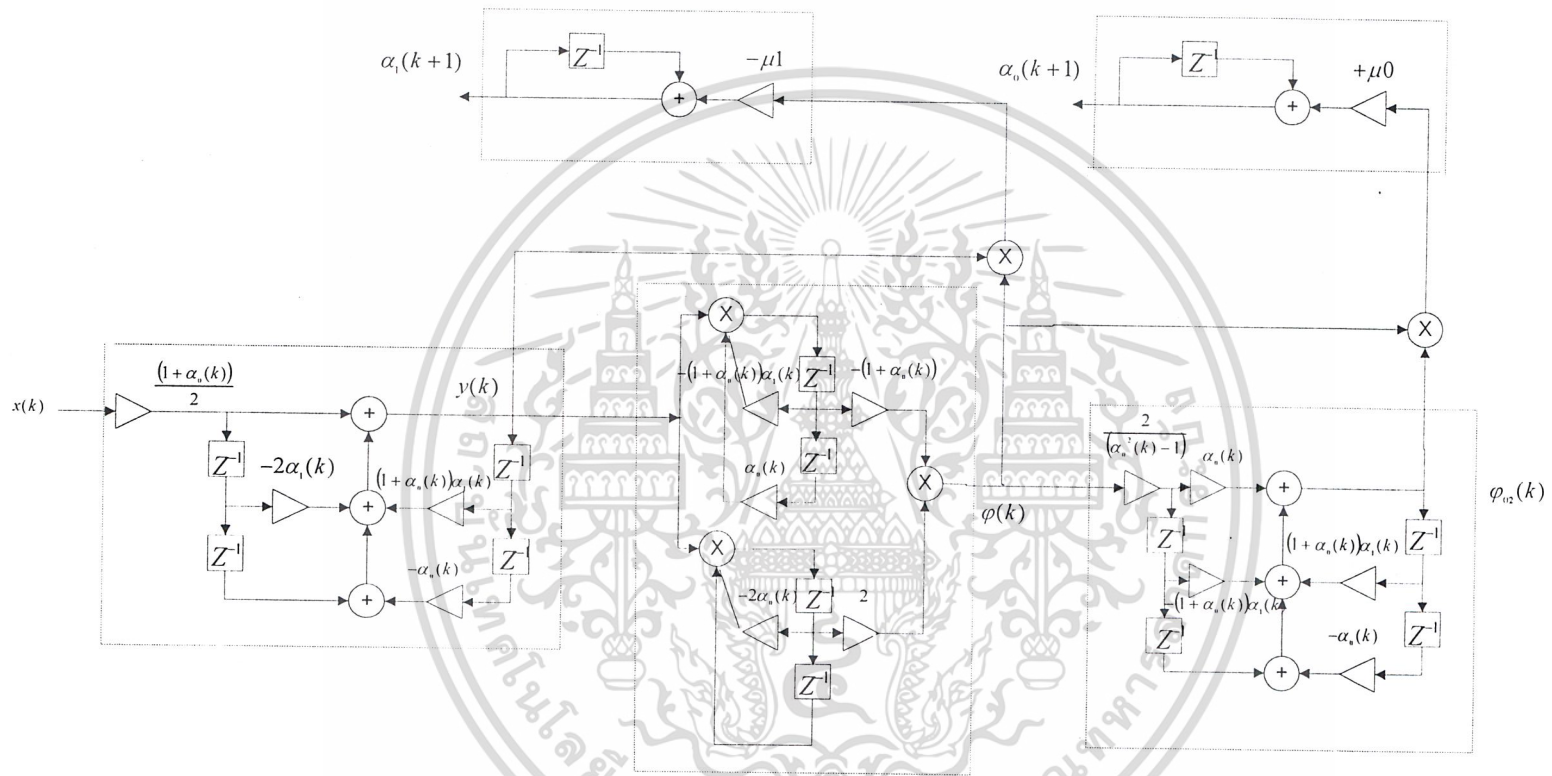
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



1. Adaptive IIR Notch Filter



2. Adaptive IIR Notch Filter for Transfer Function



3. Simplified Adaptive IIR Notch Filter

บทที่ 3

3.1 โปรแกรม

```
%%Plot 1 IIR Notch Filter กราฟแสดงคุณสมบัติของ |H(w)| เทียบกับ w
clc
alpha0=-1;
while(alpha0<0 | alpha0>1) %กำหนดค่า Q-Factor ของวงจกรองความถี่
alpha0=input('Q factor (0<alpha0<1) \n (alpha0= )');
end
omega=-1;
while (omega<0 | omega>180) %กำหนดค่า W ของวงจกรองความถี่
omega=input('W (0<w<pi) (degree) \n (W= )');
end
alpha1=cos((pi/180)*omega);
b1=1;
b2=-(2*alpha1);
b3=1;
a1=1;
a2=-(1+alpha0)*alpha1;
a3=alpha0;
k=(1+(alpha0))/2;
n=[b1,b2,b3];
d=[a1,a2,a3];
[Tz,ws]=freqz(n,d,200);
plot(ws,abs(Tz*k),'r'); %กราฟ |H(w)| เทียบกับ w
hold on
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Plot2 กราฟแสดงคุณสมบัติของ $|H(w)|$ เทียบกับ α_1

clc

alpha0=-1;

while(alpha0<0 | alpha0>1) %กำหนดค่า Q-Factor ของวงจรกรองความถี่

alpha0=input('Q factor (0<alpha0<1) \n (alpha0=)');

end

omega=-1;

while (omega<0 | omega>180) %กำหนดค่า w ของวงจรกรองความถี่

omega=input('W (0<w<pi) (degree) \n (W=)');

end

alpha1=linspace(0,1,100);%กำหนดค่าการเปลี่ยนแปลงของการตอบสนองของ Transfer Function

H=zeros(size(alpha1));

H=abs([cos(omega)-alpha1]./[[cos(omega)-alpha1]+j.*[sin(omega).*[[1-alpha0]./[1+alpha0]]]]);

plot(alpha1,H); %กราฟ alpha1 เทียบกับ Magnitude H(w)

xlabel('alpha1(k)',ylabel('|H|'))

hold on

Plot3 กราฟแสดงคุณสมบัติของ α_1 เทียบกับ k กรณีที่ยังไม่ลดตัวแปร

w=-1;

while (w<0 | w>180) %กำหนดค่า w ของวงจรกรองความถี่

w=input('W (0 to pi) \n (W=)');

end

n=10000; %จำนวนการสุ่มตัวอย่างทั้งหมด

k=0:n-1; %กำหนดลำดับการสุ่มตัวอย่างของสัญญาณ

data=cos((pi*w/180)*k+randn(1,1))+0.1*randn(1,n);

alp0=0.3; %กำหนดค่า Q-Factor ของวงจรกรองความถี่

u=0.0006; %Step Size Parameter

x=zeros(n+2,1); %กำหนดลำดับการสุ่มของสัญญาณอินพุท

x(3:n+2)=data;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

w1=zeros(n+2,1);           %กำหนดลำดับการสุ่มของค่าความถี่
alp1=zeros(n+2,1);        %กำหนดลำดับการสุ่มของการตอบสนองของสัญญาณ
y=zeros(n+2,1);           %กำหนดลำดับการสุ่มของสัญญาณเอาท์พุท
for k=3:n+2;
    y(k)=(1+alp0)/2*(x(k)-(2*alp1(k)*x(k-1))+x(k-2))+(1+alp0)*alp1(k)*y(k-1)-alp0*y(k-2);
    w1(k)=(1+alp0)*y(k-1)-(1+alp0)*x(k-1)+(1+alp0)*alp1(k)*w1(k-1)-alp0*w1(k-2);
    alp1(k+1)=alp1(k)-u*y(k)*w1(k);
end;
plot(alp1)                 %กราฟการ Converge ของ alpha1(k) เทียบกับ k
grid
xlabel('k');ylabel('alp1(k)');
title('FIG');
hold on
clear
%%%%%%Plot4 กราฟแสดงคุณสมบัติของ alpha1 เทียบกับ k กรณีที่ปรับตัวแปรแล้ว%%%%%%%%
clear;
w=-1;
while(w<0 | w>180)        %กำหนดค่า W ของวงจรรองความถี่
w=input('W (0 to 180) \n (W=)');
end
n=10000;                  %จำนวนการสุ่มตัวอย่างทั้งหมด
k=0:n-1;                  %กำหนดลำดับการสุ่มตัวอย่างของสัญญาณ
data=cos((pi*w/180)*k+randn(1,1))+0.1*randn(1,n);
u1=0.0006;                %Step Size Parameter
x=zeros(n+4,1);          %กำหนดลำดับการสุ่มของสัญญาณอินพุท
x(5:n+4)=data;
w1=zeros(n+4,1);         %กำหนดลำดับการสุ่มของค่าความถี่
w01=zeros(n+4,1);        %กำหนดลำดับการสุ่มของค่าความถี่

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

alp1=zeros(n+4,1);           %กำหนดค่าค้ำบการสุมของการคอบสนองของสัญญาณ
alp0=0.3;                    %กำหนดค่า Q-Factor ของวงจรงรองคววมถึ
y=zeros(n+4,1);
for k=5:n+4;
    y(k)=(1+alp0)/2*(x(k)-(2*alp1(k)*x(k-1))+x(k-2))+(1+alp0)*alp1(k)*y(k-1)-alp0*y(k-2);
    w1(k)=[3+alp0]*alp1(k)*w1(k-1)-[1+alp0+2*(alp1(k)^2)*(1+alp0)]*w1(k-2)+[3*alp0+1]*alp1
(k)*w1(k-3)-alp0*w1(k-4)+[alp0-1]*[y(k-1)-y(k-3)];
alp1(k+1)=alp1(k)-u1*y(k)*w1(k);
end;
plot(alp1,'r') %กราฟ alpha1 เทียบกับ k ขณะทำการแยกโดยการใ้ Transfer Function แล้ว
grid
xlabel('k');ylabel('alp1(k)');
title('FIG');
hold on;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Plot5 กราฟแสดงคุณสมบัติของ H(alpha_0) เทียบกับ alpha_0 %%%%%%%%%%%%%%%
clc
omega=-1;
while(omega<0 | omega>180) %กำหนดค่า w ของวงจรงรองคววมถึ
omega=input(' W 0 to pi\n (omega= ');
end
alpha1=-2;
while(alpha1<-1 | alpha1>1) %กำหนดค่า alpha1 ของวงจรงรองคววมถึ
alpha1=input(' alpha1 -1 to 1 \n (alpha1= ');
end
w=((pi/180)*omega);
alpha0=0:0.01:1; %กำหนดค่าการเปลี่นแปลงของค่า Q-Factor ของวงจรงรองคววมถึ

```

```

H0=(-2).*(((1.-alpha0)/(1.+alpha0)).*sin(w)).^2).*(1/(cos(w)-alpha1)^3);

```

```

H1=(((sin(w)/(cos(w)-alpha1)).*((1.-alpha0)/(1.+alpha0)).^2)+1).^2);

```

ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
H=H0./H1;
plot(alpha0,abs(H))           %กราฟ alpha0 เทียบกับ Magnitude H(alpha0)
title('H(alpha0)')
xlabel('alpha0'),ylabel('|H(alpha0)|')
hold on;
```

```
%%%%%%%%%Plot6 กราฟแสดงคุณสมบัติของ |G(z)|, |G01(z)| และ |G02(z)| เทียบกับ w%%%%%%%%%
```

```
clc
omega=-1;
while(omega<0 | omega>180)   %กำหนดค่า w ของวงจรรองความถี่
omega=input(' W input 0 to pi\n (omega= ');
end
alpha0=-1;
while(alpha1<0 | alpha1>1)   %กำหนดค่าการตอบสนองของ Transfer Function
alpha0=input(' alpha0 0 to 1 \n (alpha0= ');
end
alpha1=cos(pi*omega/180);
%G(z)
c1=0;
c2=1;
c3=0;
c4=-1;
c5=0;
d1=1;
d2=-2*(alpha0+3);
d3=(1+2*(1+alpha0)*alpha1^2+alpha0);
d4=-(alpha0*(1+3*alpha0));
d5=alpha0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 $k=(\alpha_0-1)$;
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
v=[c1,c2,c3,c4,c5];
w=[d1,d2,d3,d4,d5];
[G,ws]=freqz(v,w,200);
plot(ws,abs(k*G));           %กราฟ w เทียบกับ Magnitude [G(z)]
hold on;
%G01(z)
```

```
b1=1;
b2=0;
b3=-1;
a1=1;
a2=-(1+alpha0)*alpha1;
a3=alpha0;
k=1/(1+(alpha0));
n=[b1,b2,b3];
d=[a1,a2,a3];
[Tz,ws]=freqz(n,d,200);
plot(ws,abs(Tz*k),'r');     %กราฟ w เทียบกับ Magnitude [G01(z)]
hold on;
%G02(z)
b1=alpha0;
b2=-(1+alpha0)*alpha1;
b3=1;
a1=1;
a2=-(1+alpha0)*alpha1;
a3=alpha0;
k=2/((alpha0)^2-1);
n=[b1,b2,b3];
d=[a1,a2,a3];
```

```
[Tz,ws]=freqz(n,d,200);
plot(ws,abs(Tz*k),'g');     %กราฟ w เทียบกับ Magnitude [G02(z)]
```

```

hold on;
title('G(Z) , G01(Z) , G02(Z)')
xlabel('W'),ylabel('amplitude')
clc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Plot7 กราฟแสดงคุณสมบัติของ  $H(e^{j\omega})$  เทียบกับ  $\alpha_0$  %%%%%%%%%%%%%%%
clc
omargar=-1;
while(omega<0 | omega>180) %กำหนดค่า W ของวงจรรองความถี่
omega=input(' W (0 to pi)\n (omega= ');
end
alpha1=-2;
while(alpha1<-1 | alpha1>1) %กำหนดค่าการตอบสนองของ Transfer Function
alpha1=input(' alpha1 (-1 to 1)\n (alpha1= ');
end
w=((pi/180)*omega);
alpha0=0:0.01:1; %กำหนดค่าการเปลี่ยนแปลงของค่า Q-Factor ของวงจรรองความถี่
H0=(cos(w)-alpha1)*(1+alpha0);
H1=(1+alpha0)*cos(w)+j*sin(w)*(1-alpha0)-alpha1*(1+alpha0);
H=H0./H1;
plot(alpha0,abs(H)) %กราฟ alpha0 เทียบกับ Magnitude (alpha0)
title('H(alpha0)')
xlabel('alpha0'),ylabel('|H(alpha0)|')
hold on;
clc

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%%Plot8 กราฟแสดงคุณสมบัติของ |H(alpha0)| เทียบกับ alpha0
clc

omega=-1;

while(omega<0 | omega>180)           %กำหนดค่า w ของวงจรรองความถี่
omega=input(' W input 0 to pi\n (omega= ');
end

alpha1=-2;

while(alpha1<-1 | alpha1>1)         %กำหนดค่าการตอบสนองของ Transfer Function
alpha1=input(' alpha1 -1 to 1 \n (alpha1= ');
end

w=((pi/180)*omega);
alpha0=0:0.01:1;                    %กำหนดการเปลี่ยนแปลงค่า Q-Factor ของวงจรรองความถี่
a0=((1.-alpha0)/(1.+alpha0)).^4;
a1=(sin(w))^4;
b0=(cos(w)-alpha1)^2;
b1=(sin(w)*((1.-alpha0)/(1.+alpha0))).^2;
H0=a0.*a1;
H1=(b0+b1).^3;
H=H0./H1;
plot(alpha0,abs(H))                 %กราฟแสดง alpha0 เทียบกับ Magnitude H(alpha0)
title('H(alpha0)')
xlabel('alpha0'),ylabel('|H(alpha0)|')

hold on;

clc

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%%%%%%%%%%Plot9 Adaptive Algorithm กราฟแสดงการ Converge ของ alpha1 %%%%%%%%%%%
a=1;
if a==1
clear;
w=-1;;
while (w<0 | w>180)
    w=input('W (0 to 180) \n (W= )');
end
n=10000;
knt=0:n-1;
data=cos(pi/180*w*knt)+0.1*randn(1,n);
u1=0.0003;
u0=0.0006;
x=zeros(n+4,1);
x(5:n+4)=data;
w1=zeros(n+4,1);
w01=zeros(n+4,1);
w02=zeros(n+4,1);
alp1=0.1*zeros(n+4,1);
alp0=0.12*zeros(n+4,1);
y=zeros(n+4,1);
for k=5:n+4;
    y(k)=(1+alp0(k))/2*(x(k)-(2*alp1(k)*x(k-1))+x(k-2))+(1+alp0(k))*alp1(k)*y(k-1)-alp0(k)*y
(k-2);
    w1(k)=(1+alp0(k))*y(k-1)-(1+alp0(k))*x(k-1)+(1+alp0(k))*alp1(k)*w1(k-1)-alp0(k)*w1(k-2);
    w01(k)=1/2*(x(k)-2*alp1(k)*x(k-1)+x(k-2))+alp1(k)*y(k-1)-y(k-2)+(1+alp0(k))*alp1(k)*w01
(k-1)-alp0(k)*w01(k-2);
    w02(k)=y(k-1)-x(k-1)+alp1(k)*w1(k-1)-w1(k-2)+(1+alp0(k))*w01(k-1)+(1+alp0(k))*alp1
(k)*w02(k-1)-alp0(k)*w02(k-2);
    alp1(k+1)=alp1(k)-u1*y(k)*w1(k);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ภายใต้การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

alp0(k+1)=alp0(k)+u0*y(k)*w1(k)*(w1(k)*w01(k)+w02(k)*y(k));
if alp0(k+1)>=0.7
    alp0(k+1)=0.7;
end;
end;
disp(alp1(n+4))
plot(alp1)
xlabel('k');ylabel('alpha1(k)');
title('FIG');
hold on;
cos(pi/180*w)
end;

%%%%%%%%%Plot10 Simplify Adaptive Algorithm กราฟแสดงการ Converge ของ alpha1%%%%%%%%%
a=1;
if a==1
w=-1;
while (w<0 | w>180)
w=input('W (0 to 180)\n (W=) ');
end
n=10000;
knt=0:n-1;
data=cos(pi/180*w*knt)+0.1*randn(1,n);
u1=0.0003;
u0=0.0006;
x=zeros(n+4,1);
x(5:n+4)=data;
w1=zeros(n+4,1);
w01=zeros(n+4,1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

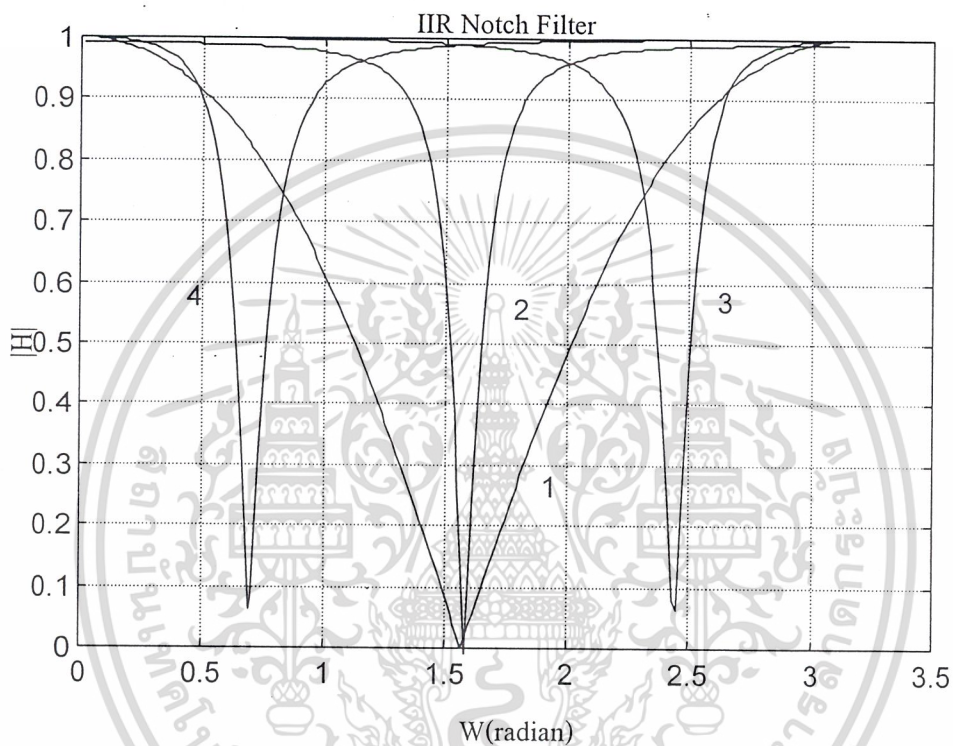
w02=zeros(n+4,1);
alp1=0.1*zeros(n+4,1);
alp0=0.12*zeros(n+4,1);
y=zeros(n+4,1);
for k=5:n+4;
    y(k)=(1+alp0(k))/2*(x(k)-(2*alp1(k)*x(k-1))+x(k-2))+(1+alp0(k))*alp1(k)*y(k-1)-alp0(k)*y
(k-2);
    w1(k)=[3+alp0(k)]*alp1(k)*w1(k-1)-[1+alp0(k)+2*(alp1(k)^2)*(1+alp0(k))]*w1(k-2)+[3*alp0
(k)+1]*alp1(k)*w1(k-3)-alp0(k)*w1(k-4)+[alp0(k)-1]*[y(k-1)-y(k-3)];
    w002=[alp0(k)*w1(k)-alp1(k)*(1+alp0(k))*w1(k-1)+w1(k-2)];
    w02(k)=[(1+alp0(k))*alp1(k)*w02(k-1)-alp0(k)*w02(k-2)]+[2/((alp0(k)^2)-1)]*w002;
    alp1(k+1)=alp1(k)-u1*y(k)*w1(k);
    alp0(k+1)=alp0(k)+u0*w1(k)*w02(k);
    if alp0(k+1)>=0.7
        alp0(k+1)=0.7;
    end;
end;
disp(alp1(n+4))
plot(alp1,'r')
grid
xlabel('k');ylabel('alp1(k)');
title('FIG');
hold on;
cos(pi/180*w)
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 กราฟแสดงผลการทดลอง

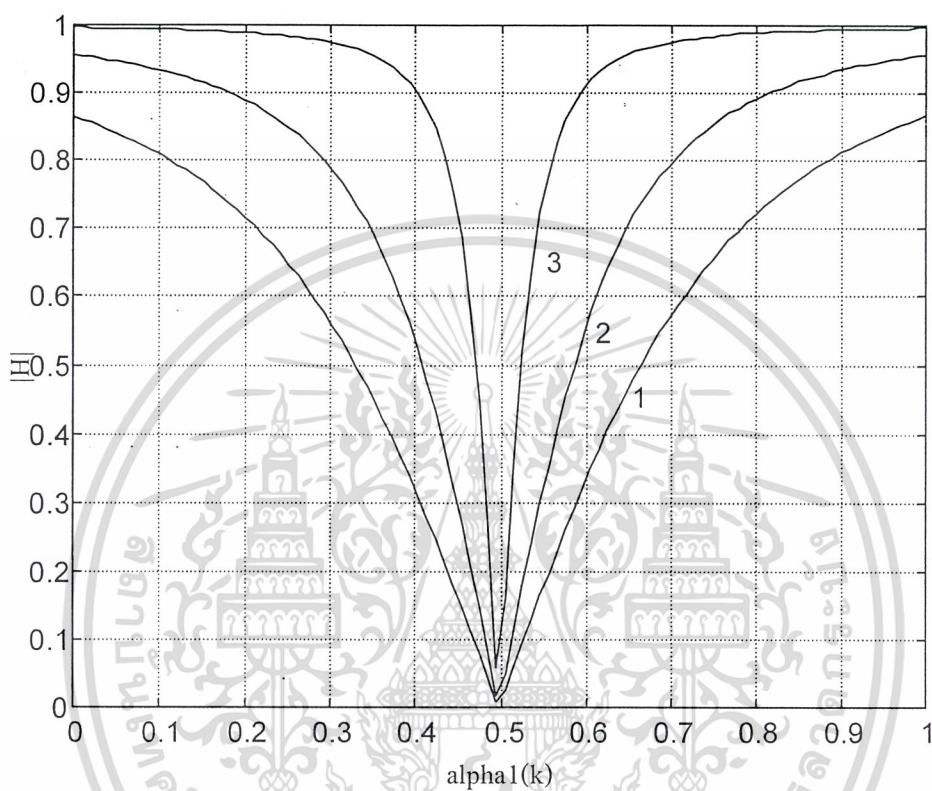
1. กราฟแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ ω



1. $\alpha_0 = 0.1, \omega = \pi/2$
2. $\alpha_0 = 0.8, \omega = \pi/2$
3. $\alpha_0 = 0.8, \omega = 3\pi/4$
4. $\alpha_0 = 0.8, \omega = \pi/4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

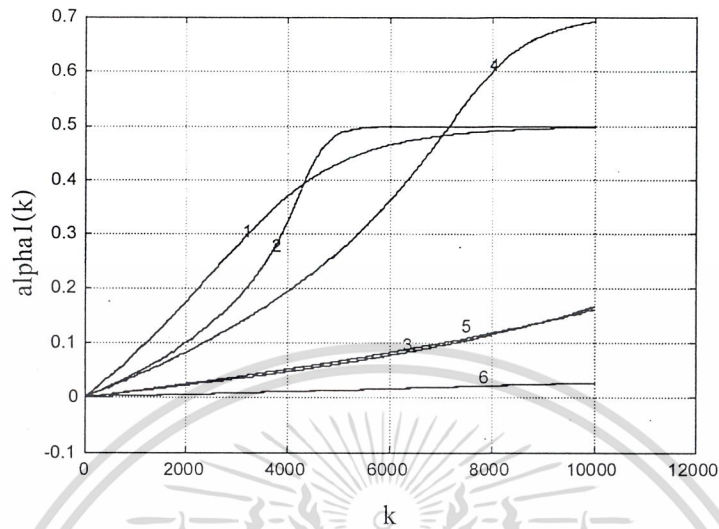
2. กราฟแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ α_1



1. $\alpha_0 = 0.5, \omega = \pi/3$
2. $\alpha_0 = 0.7, \omega = \pi/3$
3. $\alpha_0 = 0.9, \omega = \pi/3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

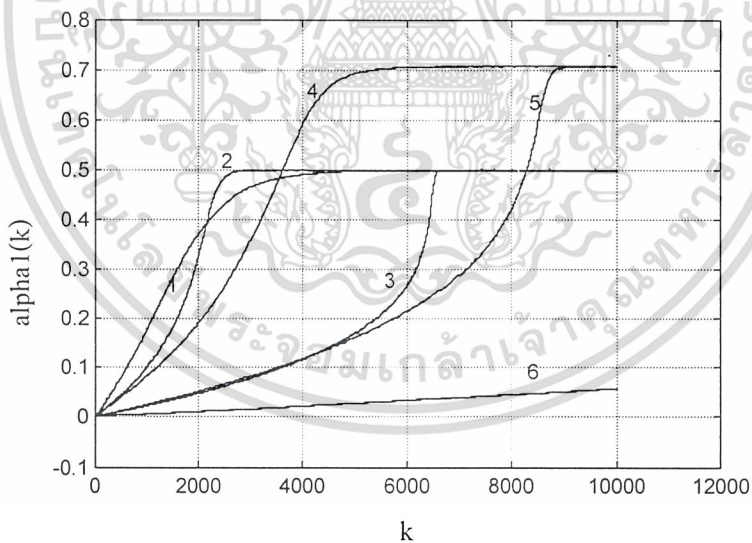
3. กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีที่ยังไม่ลดตัวแปร



1. $\mu = 0.0003, \omega = \pi/3, \alpha_0 = 0.3$ 2. $\mu = 0.0003, \omega = \pi/3, \alpha_0 = 0.6$

3. $\mu = 0.0003, \omega = \pi/3, \alpha_0 = 0.8$ 4. $\mu = 0.0003, \omega = \pi/4, \alpha_0 = 0.3$

5. $\mu = 0.0003, \omega = \pi/4, \alpha_0 = 0.6$ 6. $\mu = 0.0003, \omega = \pi/4, \alpha_0 = 0.8$



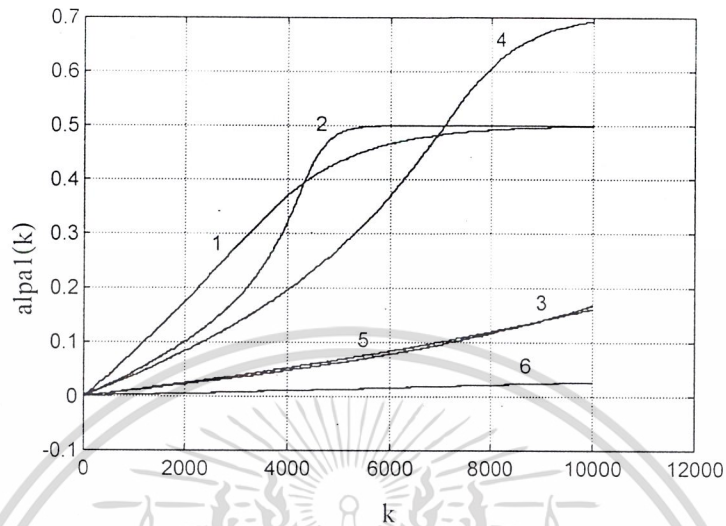
1. $\mu = 0.0006, \omega = \pi/3, \alpha_0 = 0.3$ 2. $\mu = 0.0006, \omega = \pi/3, \alpha_0 = 0.6$

3. $\mu = 0.0006, \omega = \pi/3, \alpha_0 = 0.8$ 4. $\mu = 0.0006, \omega = \pi/4, \alpha_0 = 0.3$

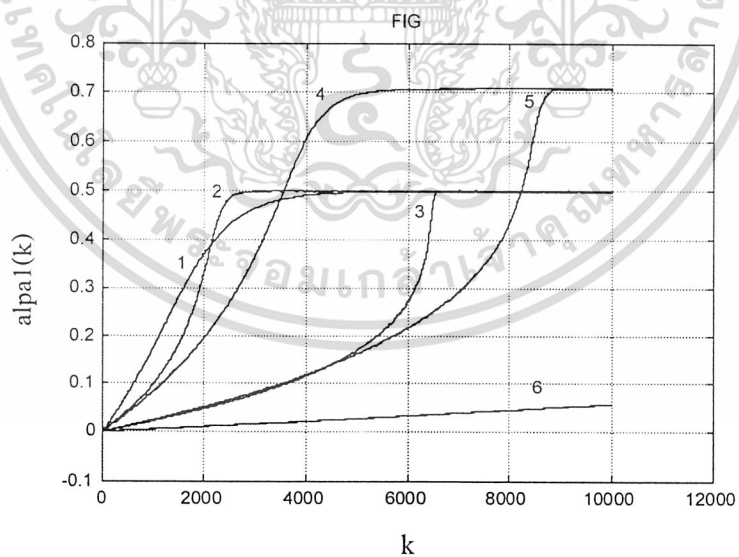
5. $\mu = 0.0006, \omega = \pi/4, \alpha_0 = 0.6$ 6. $\mu = 0.0006, \omega = \pi/4, \alpha_0 = 0.8$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีที่ปรับตัวแปรแล้ว



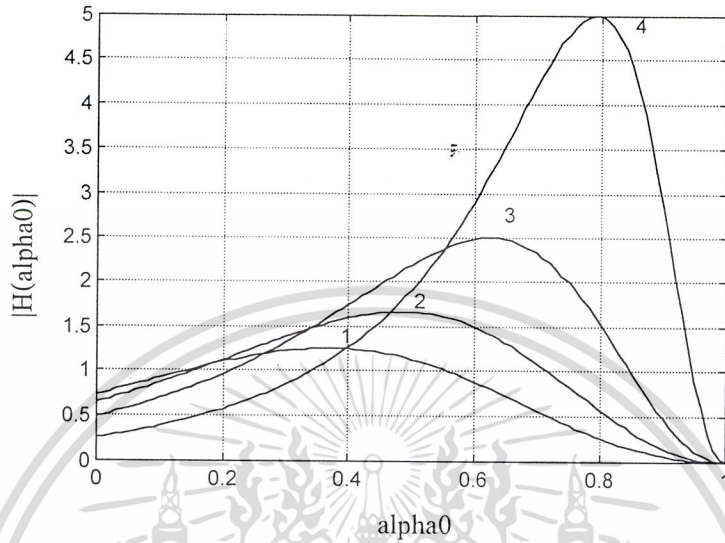
1. $\mu = 0.0003, \omega = \pi/3, \alpha_0 = 0.3$ 2. $\mu = 0.0003, \omega = \pi/3, \alpha_0 = 0.6$
 3. $\mu = 0.0003, \omega = \pi/3, \alpha_0 = 0.8$ 4. $\mu = 0.0003, \omega = \pi/4, \alpha_0 = 0.3$
 5. $\mu = 0.0003, \omega = \pi/4, \alpha_0 = 0.6$ 6. $\mu = 0.0003, \omega = \pi/4, \alpha_0 = 0.8$



1. $\mu = 0.0006, \omega = \pi/3, \alpha_0 = 0.3$ 2. $\mu = 0.0006, \omega = \pi/3, \alpha_0 = 0.6$
 3. $\mu = 0.0006, \omega = \pi/3, \alpha_0 = 0.8$ 4. $\mu = 0.0006, \omega = \pi/4, \alpha_0 = 0.3$
 5. $\mu = 0.0006, \omega = \pi/4, \alpha_0 = 0.6$ 6. $\mu = 0.0006, \omega = \pi/4, \alpha_0 = 0.8$

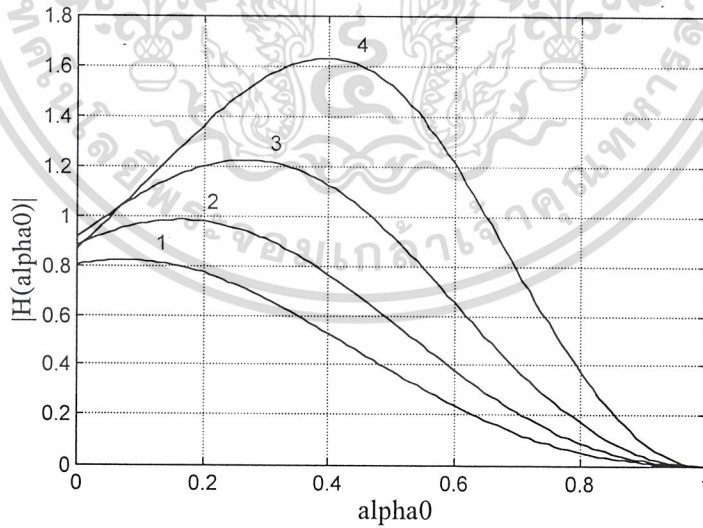
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. กราฟแสดงคุณสมบัติของ $H(\alpha_0)$ เทียบกับ α_0



1. $\omega = \pi/3, \alpha_0 = 0.1$ 2. $\omega = \pi/3, \alpha_0 = 0.2$

3. $\omega = \pi/3, \alpha_0 = 0.3$ 4. $\omega = \pi/3, \alpha_0 = 0.4$



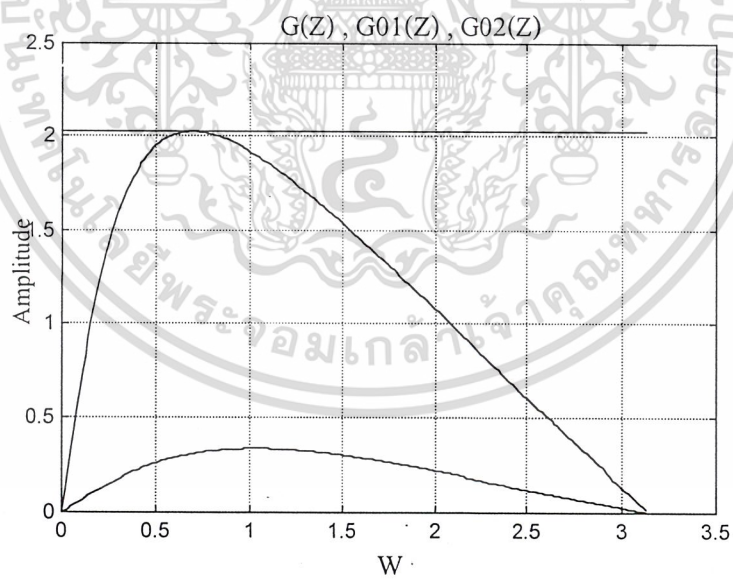
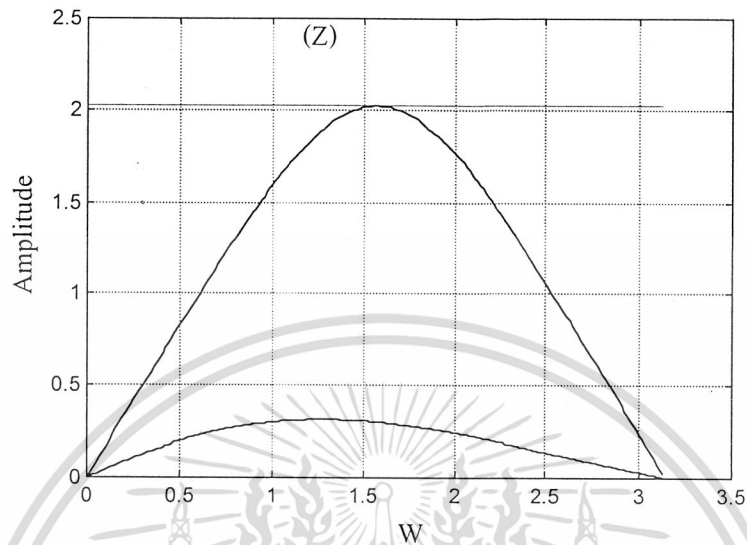
1. $\omega = \pi/4, \alpha_0 = 0.1$ 2. $\omega = \pi/4, \alpha_0 = 0.2$

3. $\omega = \pi/4, \alpha_0 = 0.3$ 4. $\omega = \pi/4, \alpha_0 = 0.4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

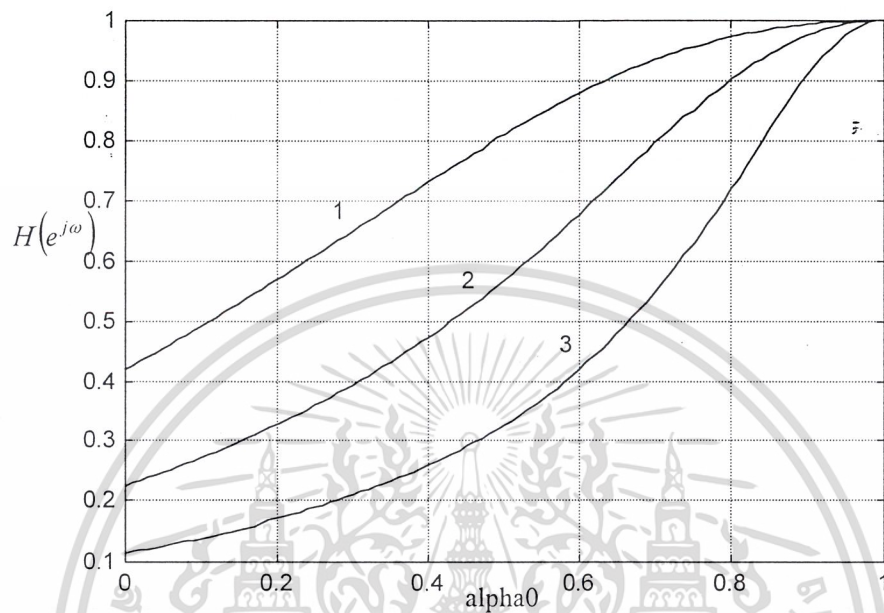
6. กราฟแสดงคุณสมบัติของ $G(z), G01(z), G02(z)$ เทียบกับ Amplitude

$G(Z) , G01(Z) , G02$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

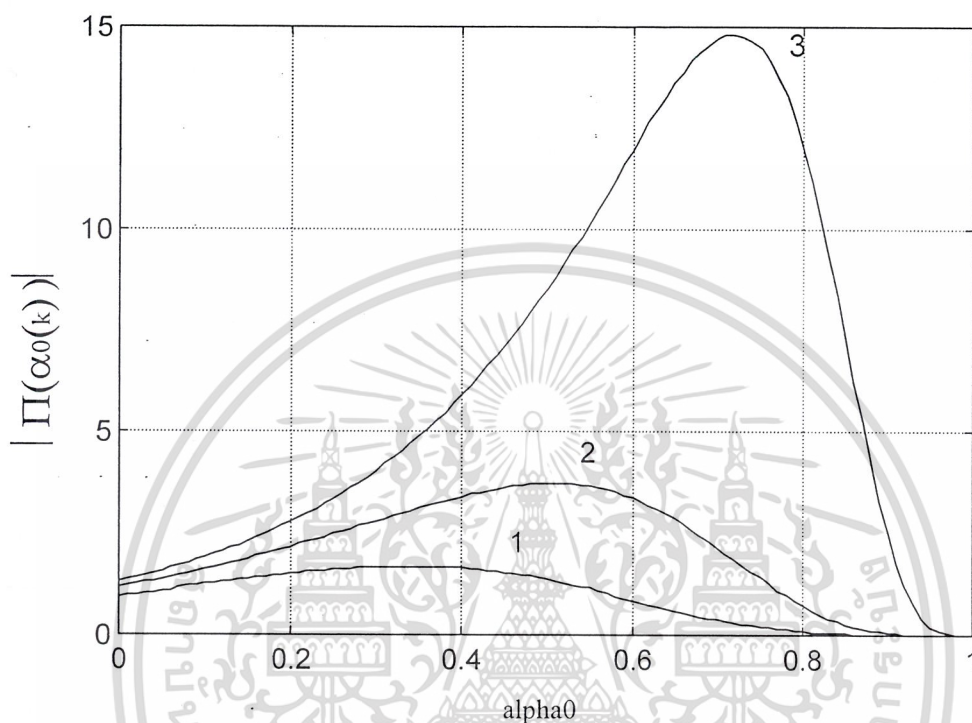
7. กราฟแสดงคุณสมบัติของ $H(e^{j\omega})$ เทียบกับ α_0



1. $\omega = \pi/3$, $\alpha_0 = 0.2$
2. $\omega = \pi/3$, $\alpha_0 = 0.3$
3. $\omega = \pi/3$, $\alpha_0 = 0.4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. กราฟแสดงคุณสมบัติของ $|\Pi(\alpha_0(k))|$ เทียบกับ α_0



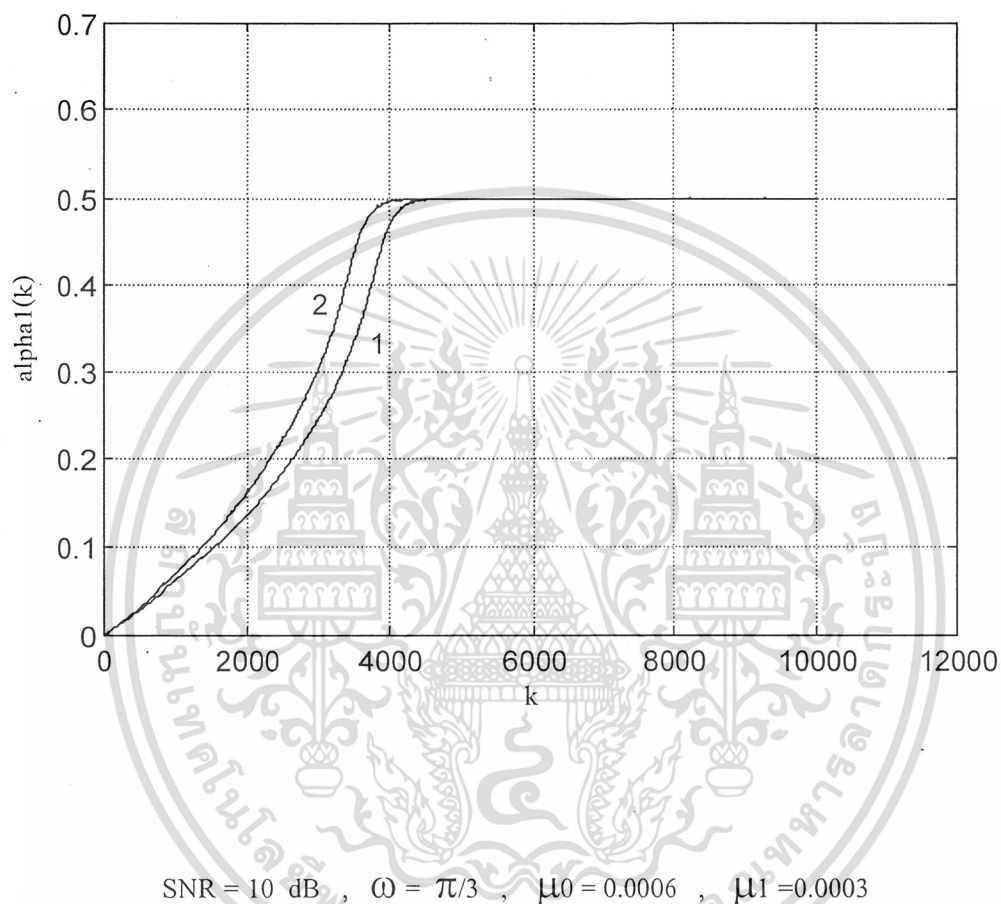
1. $\omega = \pi/3$, $\alpha_1 = 0.2$

2. $\omega = \pi/3$, $\alpha_1 = 0.3$

3. $\omega = \pi/3$, $\alpha_1 = 0.4$

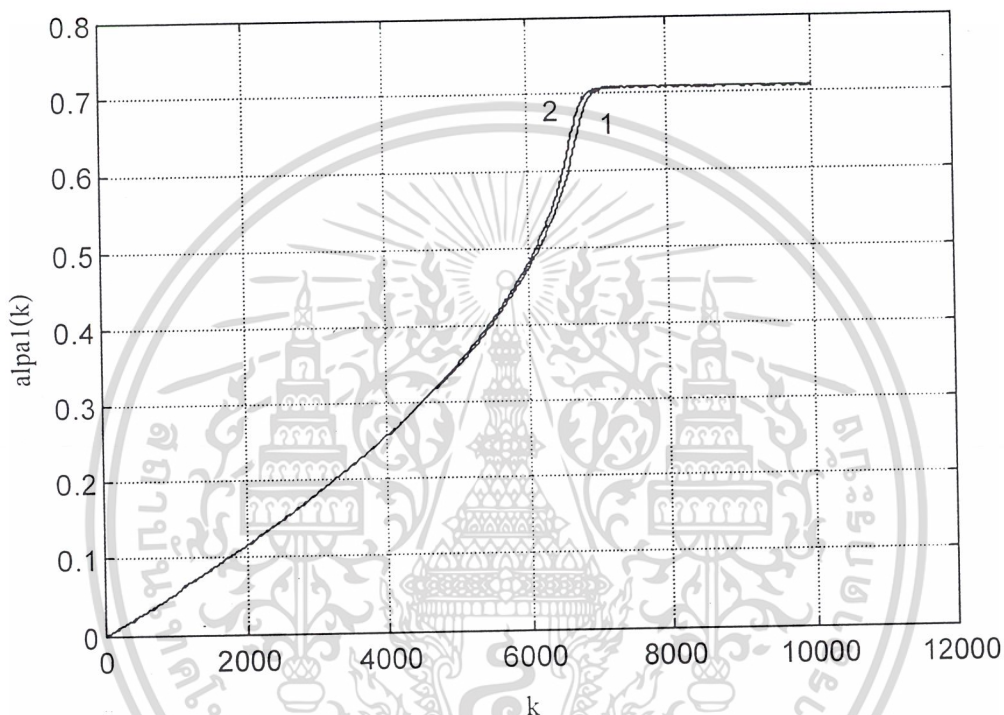
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Simplify Adaptive IIR Notch Filter (2)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

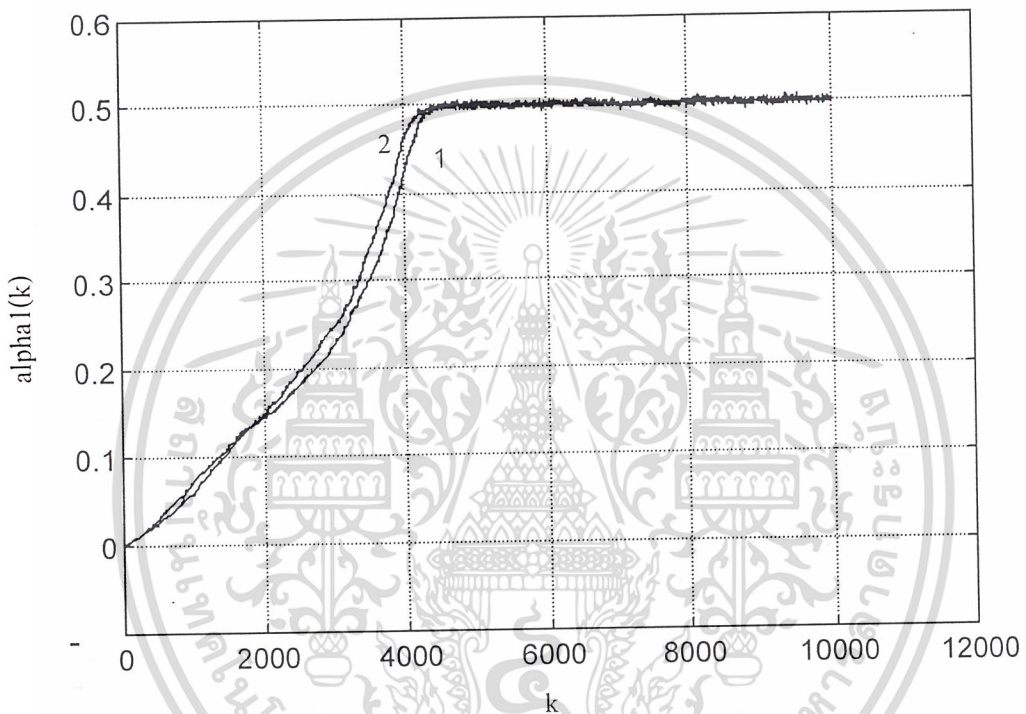
10. กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Simplify Adaptive IIR Notch Filter (2)



$$\text{SNR} = 10 \text{ dB} , \omega = \pi/4 , \mu_0 = 0.0006 , \mu_1 = 0.0003$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

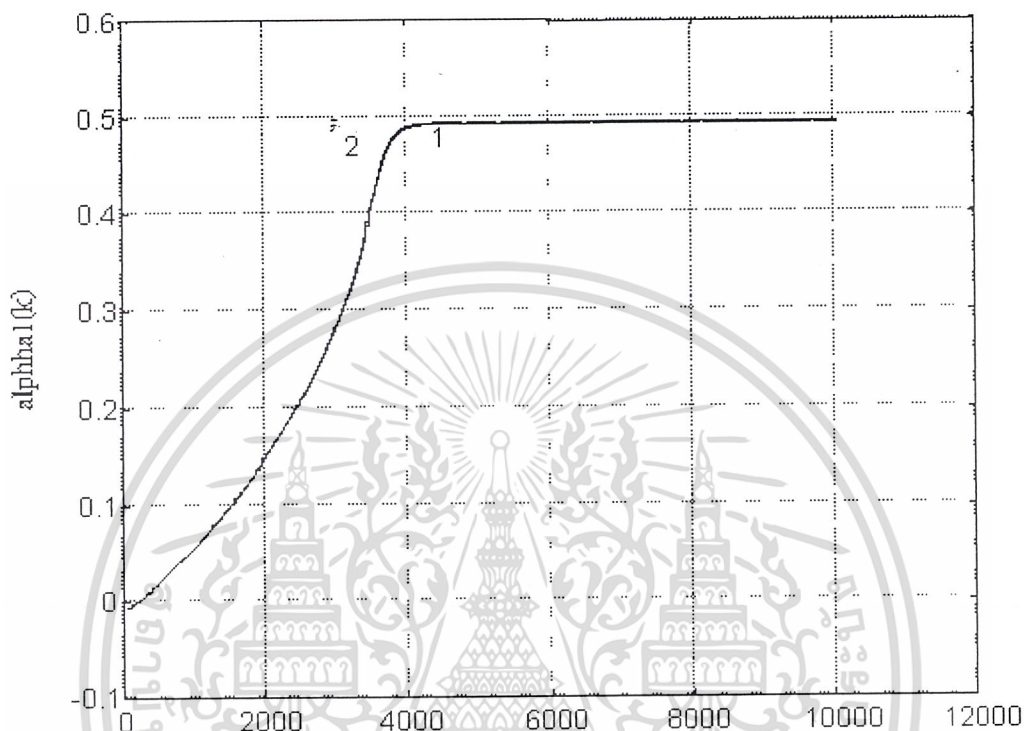
11. กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter (1) กับ Simplify Adaptive IIR Notch Filter (2)



$$\text{SNR} = 1.8 \text{ dB} , \quad \omega = \pi/3 , \quad \mu_0 = 0.0006 , \quad \mu_1 = 0.0003$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

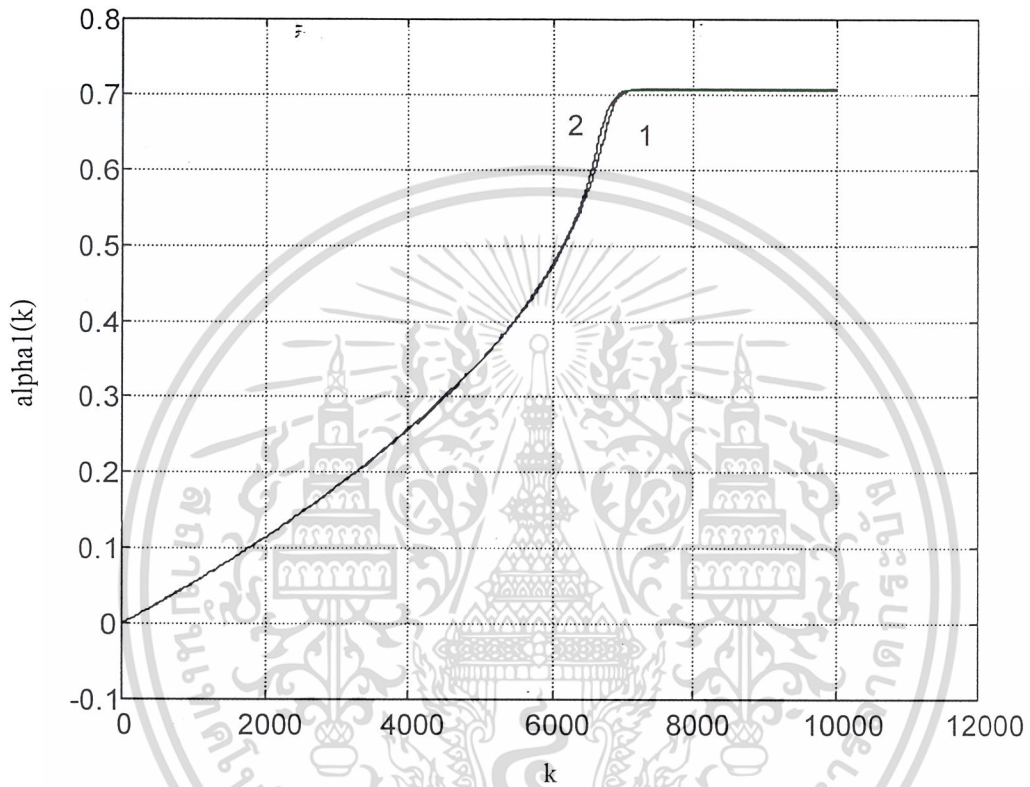
12. กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Simplify Adaptive IIR Notch Filter (1) กับ
Simplify Adaptive IIR Band Pass Filter (2)



SNR = 10 dB, $\omega = \pi/3$, $\mu_0 = 0.0006$, $\mu_1 = 0.0003$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Simplify Adaptive IIR Notch Filter (1) กับ
Simplify Adaptive IIR Band Pass Filter (2)

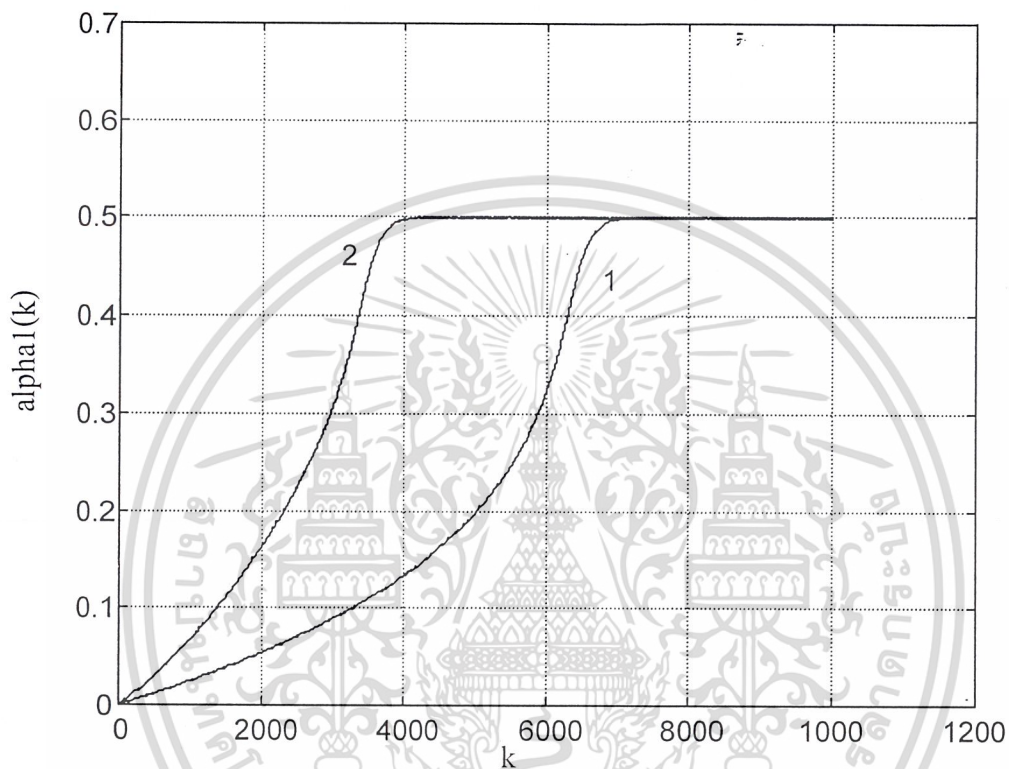


SNR = 10 dB , $\omega = \pi/4$, $\mu_0 = 0.0006$, $\mu_1 = 0.0003$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14. กราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Simplify Adaptive IIR Notch Filter

ให้ค่า Q-Factor คงที่ (1) ค่า Q-Factor แปรค่าได้ (2)



SNR = 10dB , $\omega = \pi/3$, $\mu_0 = 0.0006$, $\mu_1 = 0.0003$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การเขียนโปรแกรมบน Chip DSP

จากสมการที่ทำการ Simulation ในโปรแกรม MATLAB ในขั้นตอนนี้จะทำการเขียนโปรแกรมลงในชิพ DSP เพื่อดูการตอบสนองต่อสัญญาณจริงๆ

4.1 Frequency Response of Interface Chip

ก่อนที่จะทำการโปรแกรมบนชิพ เพื่อให้ทำงานเป็น Filter และทำการวิเคราะห์การตอบสนองในรูปแบบต่าง ๆ เราต้องทราบว่าที่ภาคสุดท้ายในส่วนของ Digital to Analog ใน IC Analog Interface (TLC32040CNF) จะมี Low Pass Filter (Reconstruction Filter) อยู่ทำให้การวิเคราะห์การตอบสนองความถี่ของ Filter ที่เราจะโปรแกรมลงบน Chip DSP เกิดการผิดพลาดได้จะนั้นก่อนอื่นเราจะต้องทำการวิเคราะห์การตอบสนองความถี่ของส่วนนี้เสียก่อน โดยทำการเขียนโปรแกรมรับค่าเข้ามาแล้วทำการส่งออกไปเลยโดยไม่ต้องทำการคำนวณใดๆเลย โดยโปรแกรมสามารถแสดงได้ดังนี้

```
*****  
;   
; SAMPLING FREQUENCY = 39.0625 kHz to test response of low pass Filter in digital  
; analog output in analog interface  
*****  
  
.MMREGS  
  
.ds 0f00h  
  
TA .word 16  
RA .word 16 ; This set up of AIC registers give  
; a sampling frequency of 39.0625 kHz  
  
TB .word 16  
RB .word 16  
  
AIC_CTR .word 08h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
rint: B RECEIVE
```

```
xint: B TRANSMIT
```

```
.ps 0a00h
```

```
.entry
```

```
SETC INTM
```

```
LDP #0
```

```
OPL #0834h,PMST
```

```
LACC #0
```

```
SAMM CWSR
```

```
SAMM PDWSR
```

```
SETC SXM ; SXM MUST BE SET
```

```
SPLK #022h,IMR ; This turns on receive interrupt only
```

```
CALL AICINIT
```

```
SPLK #12h,IMR ; This turns on receive interrupt only
```

```
CLRC OVM
```

```
SPM 0
```

```
CLRC INTM
```

```
WAIT: ; a main program would go here
```

```
B WAIT
```

```
RECEIVE:
```

```
LDP #TEMP
```

```
CLRC INTM ; EENABLE INTERRUPTS FOR DEBUGGING DSK PURPOSES
```

```
LAMM DRR ; load accumulator with word received from AIC
```

```
SAMM DXR ; write output word to transmit register
```

```
RETE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSMIT:

RETE

```

;*****
; DESCRIPTION: This routine initializes the TMS320C50 serial port and the TLC320C40's
; (AIC) TA,RA,TB,RB and control registers
;*****

```

AICINIT: SPLK #20h,TCR ; To generate 10 MHz from Tout

SPLK #01h,PRD ; for AIC master clock

MAR *,AR0

LACC #0008h ; Non continuous mode

SACL SPC ; FSX as input

LACC #00c8h ; 16 bit words

SACL SPC

LACC #080h ; Pulse AIC reset by setting it low

SACH DXR

SACL GREG

LAR AR0,#0FFFFh

RPT #10000 ; and taking it high after 10000 cycles

LACC *,0,AR0 ; (0.5ms at 50ms)

SACH GREG

LDP #TA

SETC SXM

LACC TA,9 ; Initialize TA and RA register

ADD RA,2

CALL AIC_2ND

LDP #TB

LACC TB,9 ; Initialize TB and RB register

ADD RB,2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ADD #02h
CALL AIC_2ND
;-----
LDP #AIC_CTR
LACC AIC_CTR,2 ; Initialize control register
ADD #03h
CALL AIC_2ND
RET

AIC_2ND:
LDP #0 ; Data page point is (MM reg.)
SACH DXR ; send ACC hi 00
CLRC INTM ; enable interrupt
IDLE ; wait for interrupt
ADD #6h.15 ; 0000 0000 0000 0011 XXXX XXXX XXXX XXXX b
SACH DXR ; send ACC Hi to initiate secondary protocol
IDLE ; wait for interrupt
SACL DXR ; send the T register data
IDLE ; wait for interrupt
LACL #0 ; clear ACC low
SACL DXR ; send another to make sure 1st word got sent
IDLE ; wait for interrupt
SETC INTM
RET
.END

```

โดยความถี่ Sampling ที่ระบุเปลี่ยนแปลงในการทดลองนี้จะไม่มีผลกับ Low pass Filter เพราะเราจะทำการเปลี่ยนแปลงเฉพาะค่าใน Register TB, RB ซึ่งจะไม่มีผลกับการตอบสนอง ความถี่ของ Low pass Filter (Switch Capacitor) เพราะ Clock ที่ใช้ในส่วนนี้จะมาจากการหาร เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

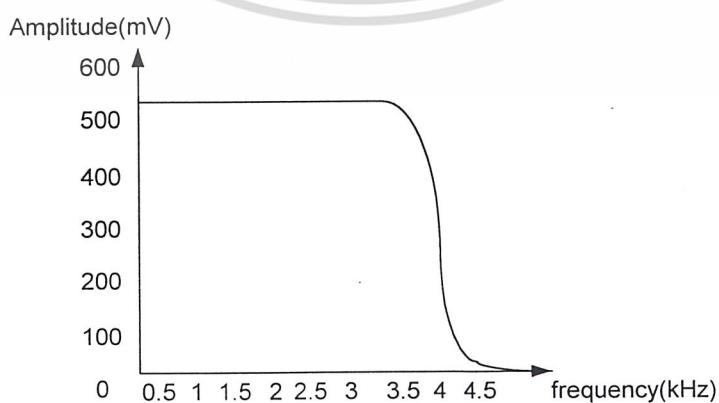
Master Clock กับค่าใน Register TA, TB (ดูรายละเอียดการทำงานของชิพ Analog Interface จาก DSK Starter Kit User Guide)

จากนั้นทำการรันโปรแกรมข้างต้นสามารถเก็บค่าการตอบสนองที่ความถี่ต่างๆ ได้ดังนี้

Frequency	Volt Output
500 Hz	560 mV
1.00 kHz	560 mV
2.00 kHz	560 mV
3.00 kHz	560 mV
3.40 kHz	544 mV
3.60 kHz	440 mV
3.82 kHz	395 mV
3.85 kHz	320 mV
3.90 kHz	232 mV
4.00 kHz	120 mV
4.50 kHz	16 mV

ตารางที่ 4.1 การตอบสนองความถี่

ซึ่งสามารถ Plot กราฟได้ดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามรูปที่ 4.1 กราฟที่แสดงการตอบสนองความถี่เอกสารทุกครั้งที่มีการนำไปใช้

จากกราฟจะเห็นว่าความถี่ Cutoff ของ Low Pass Filter จะอยู่ที่ 3.82 kHz ซึ่งเวลาทดลองจะต้องอยู่ในช่วงไม่เกินความถี่นี้ อีกอย่างคือการตอบสนองเชิง Phase ซึ่งสามารถแสดงโดยรูปของการตอบสนองความถี่ความถี่ต่างๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปแสดงถึงการตอบสนองทาง Amplitude และ Phase จะเห็นว่ามีการ Shift Phase ของแรงดัน Output ไปโดยจากการทดลองเปลี่ยนค่าความถี่พบว่าจะมีการ Shift Phase ไปประมาณ 90 องศาที่ความถี่ประมาณทุกๆ 600 Hz

จากข้อมูลข้างต้นจะทำให้สามารถวิเคราะห์การตอบสนองแบบต่างๆของ Filter ที่เราจะทำการทดลองได้ถูกต้องขึ้นทั้งทาง Amplitude และ Phase

4.2 IIR Notch Filter

ในขั้นตอนนี้จะนำเอาสมการของ IIR Notch Filter มาทำการเขียนโปรแกรมเพื่อดูผลตอบสนองของ Transfer Function ที่เราทำการ Simulate ใน MATLAB โดยดูผลการตอบสนองว่าจะเหมือนที่ทำการ Simulate หรือไม่ จากสมการของ IIR Notch Filter คือ

$$y(k) = \left(\frac{1 + \alpha_0}{2} \right) (x(k) - 2\alpha_1(k)x(k-1) + x(k-2)) + \alpha_1(k)(1 + \alpha_0)y(k-1) - \alpha_0y(k-2)$$

โดยกำหนดความถี่การ Sampling เท่ากับ 15,625 Hz (กำหนดจาก ค่า TA, RA, TB, RB ในโปรแกรม) หาได้จากสูตร

$$f_s = \frac{f_{MCLK}}{2 \times TA \times TB} \quad 4.1$$

โดยที่ f_s ความถี่ในการ Sampling

f_{MCLK} (AIC Master Clock) มีค่า 10.368 MHz

AIC Shift Clock มีค่า 2.592 MHz หรือ Instruction Cycles 7.72 MHz

Master Clock มีค่า 20 MHz หรือ Instruction Cycles 20 MHz

TA คือ จำนวนเต็มที่มีค่า ($4 < TA < 31$)

TB คือ จำนวนเต็มที่มีค่า ($2 < TB < 63$)

ทำการแทนค่า

$$\begin{aligned} f_s &= \frac{f_{MCLK}}{2 \times TA \times TB} \\ &= \frac{10 \text{ MHz}}{2 \times 16 \times 20} \end{aligned}$$

$$\therefore f_s = 15,625 \text{ Hz}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้ได้ ความถี่กลาง ใน Time Domain สามารถหาได้จากสมการจาก $\omega = \frac{2\pi}{N}$ radian /sample

$$\text{และ } \Omega_0 = 2\pi f_0 = \frac{2\pi}{T_0} \text{ radian /sec}$$

$$T = \frac{\omega}{\Omega} \quad 4.2$$

โดยที่ T เป็นช่วงเวลาของการ Sampling แต่ละครั้ง

N เป็นจำนวน Sample

ω เป็นจำนวน Sample ต่อรอบ

Ω เป็นจำนวนรอบต่อหน่วยเวลา

แทนค่าตัวแปรในสมการจะได้

$$T = \frac{\omega}{\Omega}$$

$$= \frac{\omega}{\Omega_0}$$

$$T = \frac{2\pi/N}{2\pi/T_0}$$

$$\frac{1}{T_0} = \frac{2\pi/N}{2\pi T}$$

$$\therefore f_0 = \frac{f_s}{N}$$

กำหนดให้ $N = 40$ แทนค่าในสมการจะได้

$$f_0 = \frac{f_s}{N}$$

$$= \frac{15625}{40} \text{ Hz}$$

$$\therefore f_0 = 390.625 \text{ Hz}$$

ดังนั้นใน Time Domain มีค่าเท่ากับ 390.625 Hz

ความถี่คutoff ด้านต่ำโดยประมาณหาได้จากสูตร

$$f_c = \frac{f_{MCLK} (kHz)}{288 kHz} \times \frac{1}{2 \times TA} \times 3.6 kHz \quad 4.3$$

ความถี่คutoff ด้านสูงโดยประมาณหาได้จากสูตร

$$f_{CHP} = \frac{f_{MCLK} (kHz)}{288 kHz} \times \frac{1}{2 \times TA} \times 300 \text{ Hz} \quad 4.4$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการแทนค่าความถี่คัทออฟด้านต่ำ

$$\begin{aligned} f_c &= \frac{f_{MCLK} (kHz)}{288kHz} \times \frac{1}{2 \times TA} \times 3.6kHz \\ &= \frac{10,000kHz}{288kHz} \times \frac{1}{2 \times 16} \times 3.6kHz \end{aligned}$$

$$\therefore f_c = 3906Hz$$

ทำการแทนค่าความถี่คัทออฟด้านสูง

$$\begin{aligned} f_{cHP} &= \frac{f_{MCLK} (kHz)}{288kHz} \times \frac{1}{2 \times TA} \times 300Hz \\ &= \frac{10,000kHz}{288kHz} \times \frac{1}{2 \times 16} \times 300Hz \end{aligned}$$

$$\therefore f_{cHP} = 326Hz$$

ดังนั้นความถี่คัทออฟมีค่า $326Hz, 3906Hz$ แต่ในทางปฏิบัติความถี่คัทออฟควรจะถูก Sampling อย่างน้อยด้วยความถี่ Sampling ประมาณ 2 เท่าของความถี่สูงสุดที่เราจะทำการประมวลผลจะหาได้จาก ($f_s \geq 2f_1$)

$$\text{โดยที่กำหนดให้ } \alpha_1 = \cos(\pi / 20) = 0.999996$$

$$\alpha_0 = 0.92$$

จากสมการด้านบนเราทำการแทนค่าคงที่ต่างๆ ได้ดังนี้

$$y(k) = 0.96x(k) - 1.896x(k-1) + 0.96x(k-2) + 1.896y(k-1) - 0.96y(k-2)$$

4.2.1 การแปลงค่าจำนวนทศนิยมมาทำการคำนวณในรูปจำนวนเต็ม

ค่าคงที่ต่างๆ ในแต่ละเทอมต้องทำการแปลงก่อนที่จะทำการเขียนลงบนชิพ เพราะค่านี้จะเป็นค่าที่อยู่ในรูปของจำนวนทศนิยม (Floating-Point) ส่วนในชิพ ตระกูล C50 จะสามารถคำนวณในรูปจำนวนเต็ม (Fixed-Point) ได้เท่านั้น โดยการคำนวณนั้นจะต้องทำการนำค่าเหล่านี้มาทำการคูณกับค่าที่ A/D แปลงมาซึ่งจะอยู่ในรูปของจำนวนเต็มเช่นกันแต่ระดับแรงดันใน Domain ของ Analog นั้นจะเป็นค่าทศนิยมเช่นกัน ซึ่งในการคูณกันนั้นผลออกมาจะไม่ถูกต้องตามความจริง เพราะการคูณค่าทศนิยมกับจำนวนใดๆ นั้นจะทำให้คำตอบที่มีค่าน้อยลง เพราะฉะนั้นการทำการเก็บค่าคำตอบของการคูณนั้นต้องทำการ Scale ค่าของการคำนวณในรูป Binary คือ การทำการ Shift

นั่นเองโดยการ Shift ขวาไปหนึ่ง bit หมายถึงการหารสองในฐาน 10 ส่วนการ Shift ซ้ายไปหนึ่ง bit คือการคูณด้วยสองในฐาน 10 โดยในการ Shift ซ้าย หรือการหารนั้นจะเป็นค่าที่ bit LSB ไม่นำมา

คิดเพราะการ Shift ซ้ายนั้นจะตัด LSB ทิ้ง หรือถ้าจะมองได้อีกอย่างคือการหารด้วยการ Shift นั้นจะได้ค่าที่ถูกตัดก็ต่อเมื่อ จำนวน bit ทางด้าน LSB นั้นจะต้องเป็น 0 เท่ากับจำนวน bit ที่ทำการ Shift ซึ่งใน Chip DSP โดยทั่วไปนั้นจะมี Shifter ที่ทำหน้าที่ในส่วนนี้อยู่แล้ว ซึ่งส่วนมากจะทำในขั้นตอนของการเก็บและโหลดค่า

โดยในการแปลงค่าสัมประสิทธิ์ของ Filter ในที่นี้จะทำการแปลงโดยอ้างอิงกับการแปลงแรงดันของ A/D มาอยู่ในรูป Binary ที่จะทำการคำนวณเพราะจะเป็นการง่ายในการ Scale ค่าหลังการคำนวณ โดยจะอ้างอิงกับแรงดันอ้างอิงในการแปลงค่าของ A/D ซึ่งก็คือ 2 Volt แต่ในการแปลงนั้นจะมีทั้งส่วนของบวกและลบโดยการแปลงในส่วนของแต่ละค่าจะเป็น 14 bit ซึ่งทำให้ได้รายละเอียดเท่ากับ $2^{14} = 16384$ ระดับ ซึ่งค่านี้จะเท่ากับ 1 ใน Domain ของแรงดัน

ซึ่งทำให้สามารถคำนวณค่าคงที่ต่างๆ ได้ดังนี้

$$0.96 \times 16384 = 15728$$

$$-1.896 \times 16384 = -31064$$

$$0.96 \times 16384 = 15728$$

$$1.896 \times 16384 = 31064$$

$$-0.92 \times 16384 = -15073$$

สามารถเขียนสมการกับค่าคงที่ในหน่วยของค่าที่ทำการ Scale แล้วได้ดังนี้

$$y(k) = 15728x(k) - 31064x(k-1) + 15728x(k-2) + 131064y(k-1) - 15073y(k-2)$$

โดยสามารถเขียน โปรแกรมเพื่อให้ Chip DSP ทำการคำนวณหาแรงดัน Output แต่ละขณะเวลาเพื่อส่งออกไปที่ Digital to Analog เพื่อทำการแปลงกลับเป็น Analog โปรแกรมสามารถแสดงได้ดังนี้

```

;*****
;
;           IIR Notch Filter
; This program uses an order2 IIR Notch filter to implement a filter
; The center frequency is 390 Hz
;*****
.MMREGS
.ds 0f00h

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

TA .word 16
RA .word 16      ; This set up of AIC registers give
                  ; a sampling frequency of 15,625 Hz

TB .word 20
RB .word 20
AIC_CTR .word 08h
OUTPUT .word 0

;-----
; Coefficient Generator Notch Filter
;-----
a2 .word 15728   ; 0.96 X(n-2)
a1 .word -31064 ; 1.896 X(n-1)
a0 .word 15728   ; 0.96 X(n)
b2 .word -15073 ; 1.896 Y(n-2)
b1 .word 31064   ; 0.92 Y(n-1)

;-----
; Storage delay line (Y(n-1),...,Y(n-N),X(n),...,X(n-N))
;-----
YN .word 0,0
XN .word 0,0
XNL .word 0

        .ps 0080ah      ; Program initial Address for interrupt

rint: B RECEIVE
xint: B TRANSMIT

        .ps 0a00h      ; Program initial Address

```

```
.entry
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

-----
SETC INTM          ; Disable All interrupt to modify internal register
LDP #0             ; Data Page Pointer point to memory map register
OPL #0834h,PMST   ; Write to PMST register
LACC #0           ; Assign to value to software wait state generate
SAMB CWSR
SAMB PDWSR
SETC SXM          ; Enable sign extension mode (SXM bit=1)
SPLK #022h,IMR
CALL AICINIT
SPLK #12h,IMR     ; This turns on receive interrupt only
CLRC OVM
SPM 0
CLRC INTM

WAIT:             ; a main program
    B WAIT

RECEIVE:
LDP #XN
CLRC INTM        ; Enable interrupts for debugging DSK purposes
LAMB DRR        ; load accumulator with word received from AIC
SACL XN         ; Next line should be commented out if the store
LAR AR0,#XNL    ; store value of received word to a variable
ZAP             ; zero ACC & Product registers
MAR *,AR0      ; AR0 is the current AR register.
RPT #4         ; Repeat the next instruction 5 times through
MACD #a2,*-    ; the complete coefficient table
APAC           ; Accumulate last product

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SACH OUTPUT,1      ; ACC->OUTPUT. Get rid of extra sign bit.
LACC OUTPUT
SFL                ; left shift
SACL YN
AND #0fffch        ; Two LSB's must be zero for AIC
SAMM DXR           ; write output word to transmit register
RETE

```

TRANSMIT:

```

RETE
;*****
; DISCRIPTION: This routine initializes the TMS320C50 serial port and the TLC320C40's (AIC)
; TA,RA,TB,RB and control registers
;*****
AICINIT: SPLK #20h,TCR ; To generate 10 MHz from Tout (TCR=timer control register)
SPLK #01h,PRD        ; for AIC master clock
MAR *,AR0
LACC #0008h          ; Non continuous mode
SACL SPC             ; FSX as input (SPC=serial port control register)
LACC #00c8h
SACL SPC
LACC #080h           ; Pulse AIC reset by setting it low
SACH DXR
SACL GREG            ; generate reset signal with access to external memory
LAR AR0,#0FFFFh
RPT #10000
LACC *,0,AR0         ; (0.5ms at 50ns)
SACH GREG
;-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 LDP #TA
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SETC SXM
LACC TA,9           ; initialize TA and RA register
ADD RA,2
CALL AIC_2ND
;-----
LDP #TB
LACC TB,9
ADD RB,2           ; initialize TB and RB register
ADD #02h
CALL AIC_2ND
;-----
LDP #AIC_CTR
LACC AIC_CTR,2     ; initialize control register
ADD #03h
CALL AIC_2ND
RET
AIC_2ND:
LDP #0             ; Data page point is 0 (MM reg.)
SACH DXR          ; send ACC Hi 00
CLRC INTM        ; enable interrupts
IDLE              ; wait for interrupt
ADD #6h,15        ; 0000 0000 0000 0011 XXXX XXXX XXXX XXXX b
SACH DXR          ; send ACC high to initiate secondary protocol
IDLE              ; wait for interrupt
SACL DXR          ; send the T register data
IDLE              ; wait for interrupt
LACL #0           ; clear ACC low
SACL DXR          ; send another to make sure 1st word got sent
IDLE              ; wait for interrupt

```

RET

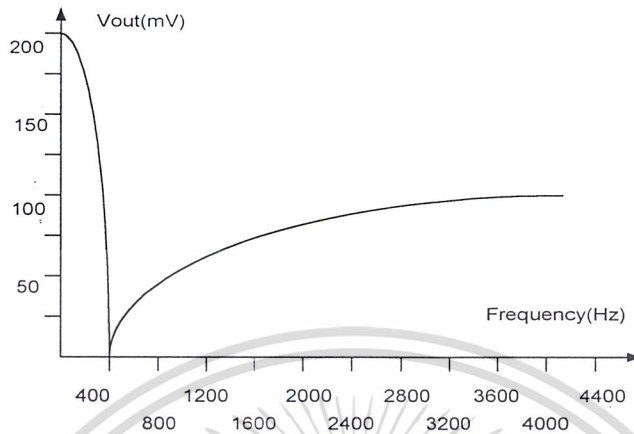
.END

จากการรันโปรแกรมข้างต้นแล้วทำการป้อนสัญญาณ Input ความถี่ต่างๆ แล้วทำการวัดค่าการตอบสนองทาง Amplitude ที่ความถี่ต่างๆ ดังตาราง

Frequency	Volt Output(mV)
50 Hz	200
100 Hz	200
200 Hz	200
250 Hz	190
300 Hz	160
330 Hz	140
360 Hz	130
390 Hz	0
420 Hz	40
450 Hz	44
515 Hz	50
600 Hz	54
650 Hz	60
700 Hz	62
800 Hz	67
900 Hz	70
1 kHz	74
2 kHz	80
3 kHz	90
4 kHz	100
5 kHz	110

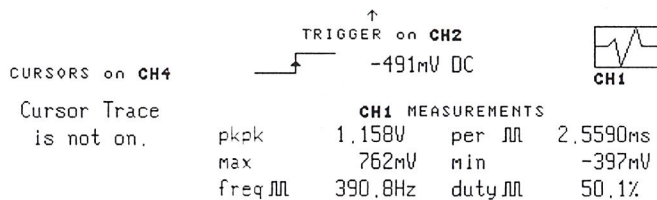
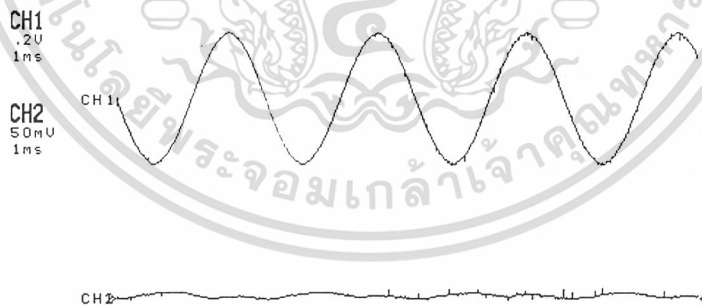
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ตารางที่ 4.2 การตอบสนองความถี่ของ IIR Notch Filter
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถพลอตกราฟได้ดังนี้



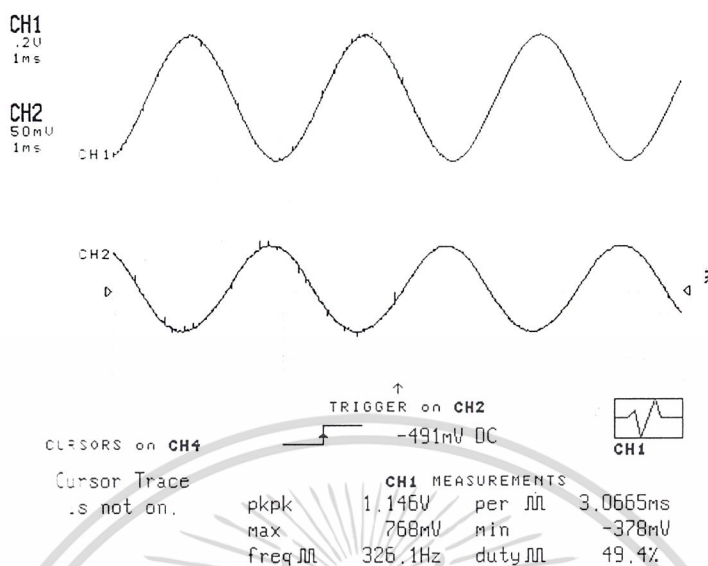
รูปที่ 4.4 กราฟการตอบสนอง ของ IIR Notch Filter

จากกราฟการตอบสนองของ IIR Notch Filter จะเห็นได้ว่าการตอบสนองความถี่เป็นไปตามที่คำนวณไว้คือค่า Amplitude ที่ความถี่ 390 Hz มีค่าน้อยที่สุด และเหตุที่สัญญาณด้านความถี่ต่ำมีความชันมากกว่าเพราะที่บอร์ด DSP ด้าน Input และ Output จะมีวงจร Low Pass Filter จึงทำให้สัญญาณมีลักษณะดังรูปที่ 4.4 ซึ่งไม่ได้เกิดจากความผิดพลาดของการประมวลผล โดยรูปคลื่นที่ความถี่กลางและความถี่ Cutoff ทั้งสองด้านสามารถแสดงดังนี้

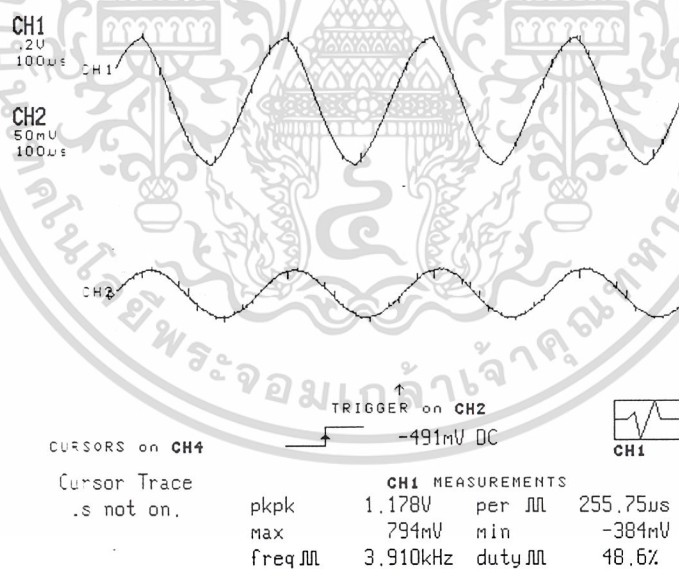


เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับอาจารย์ผู้สอนเท่านั้น การนำเอกสารนี้ไปเผยแพร่โดยไม่ขออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.5 การตอบสนองที่ความถี่กลางของ IIR Notch Filter



รูปที่ 4.6 การตอบสนองที่ความถี่ Cutoff ด้านสูง



รูปที่ 4.7 การตอบสนองที่ความถี่ Cutoff ด้านต่ำ

จากรูปคลื่นที่แสดงให้เห็นว่าที่ความถี่กลาง 390 Hz จะมีค่าแรงดันต่ำสุด 0 V และที่ความถี่ Cutoff การด้านสูงจะมีแรงดัน 200 mV และที่ความถี่ Cutoff ด้านต่ำจะมีแรงดัน 100 mV จะเห็นได้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ว่าเป็นไปตามที่ได้ออกแบบไว้ แต่การตอบสนองทาง Phase ของ Filter นี้จะไม่ตรงกับทางทฤษฎี ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ผลและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพราะว่ามีผลตอบสนอง ของ Low Pass Filter ที่ภาคสุดท้ายของ D/A มาเกี่ยวข้องด้วย ซึ่งจริงๆ ที่ความถี่กลาง จะไม่มีความเพี้ยนทาง Phase เกิดขึ้น

4.3 Adaptive IIR Notch Filter

ส่วนของการโปรแกรมจะเหมือนกับของ Adaptive IIR Notch Filter จะทำการเขียนโปรแกรมเพื่อดูผลการตอบสนองต่อการเปลี่ยนแปลงความถี่ของสัญญาณ Input (การเปลี่ยนแปลงค่า $\alpha(k)$) ว่าจะติดตามสัญญาณ Input อย่างไรรวมทั้งการแยกสัญญาณชานซ์ ที่เป็นความถี่หลักออก จากสัญญาณรบกวน

4.3.1 การโปรแกรม

ส่วนของการโปรแกรมจะเหมือนกับของ IIR Notch Filter ในส่วนแรก แต่จะมีส่วนของการ Update ค่าความถี่กลางของ Filter ในกรณีที่สัญญาณ Input มีการเปลี่ยนแปลงความถี่เพิ่มขึ้นมา คือการคำนวณค่าในส่วนของการ $\alpha(k+1)$ และ $\psi(k)$ โดยสามารถเขียน Flowchart ขั้นตอนการคำนวณของโปรแกรมห่างที่ได้แสดงไว้

โดยจากสมการทาง Time Domain ของ IIR Notch Filter และสมการการ Update ค่าความถี่กลางที่ได้ทำการ Simulate ดูการตอบสนองค่าต่างๆกับ MATLAB มาแล้วในตอนแรกคือ

$$\begin{aligned}
 y(k) &= \left(\frac{1+\alpha_0}{2}\right)(x(k)-2\alpha_1(k)x(k-1)+x(k-2))+\alpha_1(k)(1+\alpha_0)y(k-1)-\alpha_0y(k-2) \\
 \alpha_1(k+1) &= \alpha_1(k)-\mu_0\psi(k)\psi(k) \\
 \psi(k) &= (\alpha_0-1)(y(k-1)-y(k-3))+\alpha_1(k)(3+\alpha_0)\psi(k-1)-(1+2(1+\alpha_0)\alpha_1^2(k)+\alpha_0)\psi(k-2) \\
 &\quad +\alpha_1(k)(1+3\alpha_0)\psi(k-3)-\alpha_0\psi(k-4)
 \end{aligned}$$

ในขั้นตอนนี้จะทำการเขียนโปรแกรมลงบน Chip DSP เพื่อทำการแยกเอาสัญญาณชานซ์ ที่มี Noise โดยมี SNR เท่ากับ 10 dB เหมือนกับที่ทำการ Simulate

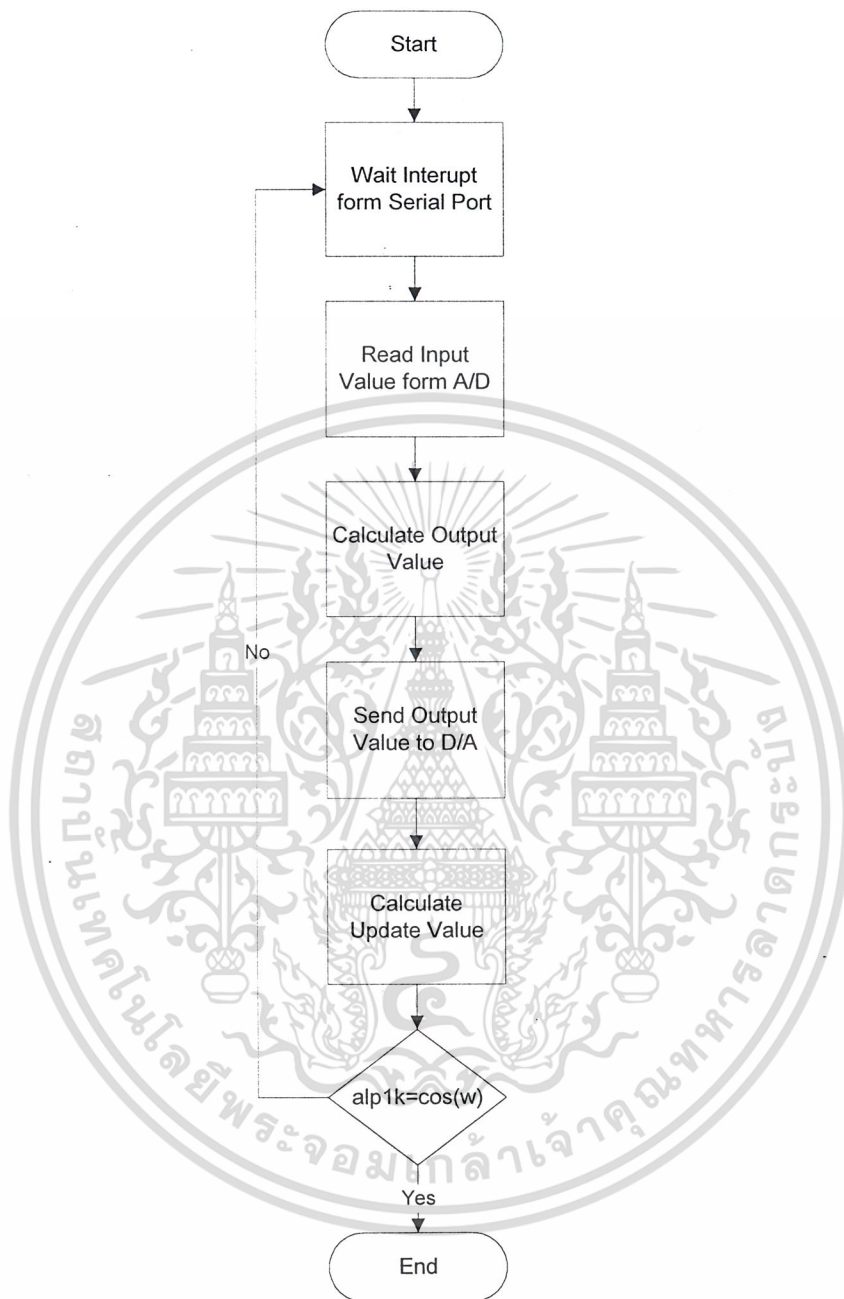
ในขั้นตอนนี้เราจะกำหนดค่าความถี่ของสัญญาณ Sampling เท่ากับ 15,625 kHz ค่า Q-Factor (α_0) เท่ากับ 0.7 (เหตุที่กำหนดมากไม่ได้เพราะจะทำให้การติดตามการเปลี่ยนแปลงความถี่ของสัญญาณ Input ทำได้ช้าแต่ถ้ากำหนดค่า Q น้อยเกินไปก็จะทำให้ความถี่ที่เราไม่ต้องการออกมามาก) ส่วนค่าของความถี่กลาง $\alpha_1(k)$ จะเริ่มต้นที่ค่า 0 ส่วนค่า Step Size Parameter (μ) กำหนดไว้ที่ 0.0003 ดังนั้นสามารถเขียนสมการ $\psi(k)$ ใหม่ได้เป็น

$$y(k) = 0.85(x(k) - x(k-2)) - 0.7y(k-2)$$

$$\alpha_1(k+1) = \alpha_1(k) + 0.0003y(k) \cdot \psi(k)$$

$$\psi(k) = -0.3y(k-1) - y(k-3) - 1.7\psi(k-2) - 0.7\psi(k-4)$$

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี หากมีการนำไปใช้โดยไม่ขออนุญาต หรือมีการดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



จะเห็นว่ามีเทอมที่เป็นศูนย์อยู่คือเทอมที่มีการคูณกับ $\alpha_1(k)$ แต่พอทำการ Update ค่า $\alpha_1(k)$ เพื่อติดตามสัญญาณ Input ก็จะมีค่ามากขึ้นมาเอง ซึ่งสามารถคำนวณค่าคงที่ต่างๆ โดยอ้างอิงกับ A/D และ D/A Converter ขนาด 14 bit ซึ่งได้อธิบายไปแล้วในหัวข้อที่ผ่านมา หลังจากนั้นทำการเขียนโปรแกรมของ Adaptive Algorithm IIR Notch Filter เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
*****
```

```
; ADAPTIVE IIR NOTCH FILTER
```

```
*****
```

```
.MMREGS
.ds 0f00h
TA .word 16
RA .word 16 ; This set up of AIC registers give
; a sampling freq of 15,625 Hz
TB .word 20
RB .word 20
AIC_CTR .word 08h
OUTPUT .word 0
templ .word 0
;-----
; Coefficient Generator Notch Filter
;-----
mue .word -5
alp1k .word 0
a2 .word -13926
a1 .word 0
a0 .word 13926
b2 .word -11469
b1 .word 0
d4 .word -11459
d3 .word 0
d2 .word 27853
d1 .word 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
f2 .word 60620
```

```
f1 .word 50790
```

```
-----
```

```
; Storage delay line
```

```
-----
```

```
YN1 .word 0
```

```
YN2 .word 0
```

```
XN .word 0,0
```

```
XNL .word 0
```

```
BUF .word 0
```

```
YNx .word 0
```

```
PHI1 .word 0
```

```
PHI2 .word 0
```

```
.ps 0080ah
```

```
rint: B RECEIVE
```

```
xint: B TRANSMIT
```

```
.ps 0a00h
```

```
.entry
```

```
-----
```

```
SETC INTM
```

```
LDP #0
```

```
OPL #0834h,PMST
```

```
LACC #0
```

```
SAMM CWSR
```

```
SAMM PDWSR
```

```
SETC SXM ; SXM MUST BE SET
```

```
SPLK #022h,IMR ; This turns on receive interrupt only
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CALL AICINIT
SPLK #12h,IMR
CLRC OVM
SPM 0
CLRC INTM

```

WAIT: ; a main program would go here

```
B WAIT
```

RECEIVE:

```

LDP #XN
CLRC INTM ; ENABLE INTERRUPTS FOR DEBUGGING DSK PURPOSES
LAMM DRR ; load accumulator with word received from AIC
SACL XN ; store the value of received word to a variable
EXAR
LAR AR0,#XNL ; load AR0 with address of last delay element
ZAP ; ZERO ACC AND PRODUCT REGISTERS
MAR *,AR0 ; AR0 is the current AR register.

RPT #4 ; Repeat the next instruction 5 times through
MACD #a2,*- ; the complete coeff table.
APAC ; Accumulate last product

SACH OUTPUT,1 ; ACC -> OUTPUT. Get rid of extra sign bit.
LACC OUTPUT
SFL
SACL YN1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 AND #0fffch ; Two LSB's must be zero for AIC
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SAMM DXR ; write output word to transmit register

LACC YN2 ;Adaptation section
SACL YNx
LAR AR1,#PHI2 ;AR1 point to add off PHI2
MAR *,AR1 ;AR1 is current auxiliary register
ZAP ;Zero ACC &Product register
RPT #6 ;Repeat next instruction 7 time
MACD #d4,*- ;calculate PHI1
APAC
SACH PHI1 ;store PHI1
LAR AR2,#YN1 ;calculate alp1(k+1)
MAR *,AR2 ;AR2 is current AR register
ZAP
MAC #PHI1,*,AR3 ;yn*PHI1
APAC
SACH temp1
LAR AR3,#mue
ZAP
LACC alp1k,16
MAC #temp1,*,AR4 ;Calculate Alp1(k)
APAC ;(yn*PHI1)*mue + alp1k
SACH alp1k
LAR AR4,#alp1k ;calculate new cefficient for next time (b2)
ZAP
MAC #d2,* ; alp1k(1+alp0)
APAC
SACH b1,2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 LACC b1
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ADDC  *,ffff          ;-alp1k(1+alp0)
SACL  a1,2
      LACC  alp1k
      MAR  *,AR6
      MAC  #f2,*       ; alp1k(3+alp0)
      APAC
SACH  d1,2

      LACC  alp1k
      MAR  *,AR7       ; alp1k(1+3alp0)
      MAC  #f1,*
      APAC
SACL  d3,2
      LACC  b1
      MAC  #alp1k,*    ; (2*alp1k*alp1k(1+alp0))+1+alp0
      SFL
      ADD  *,6ccc
      SACH  d2,2
      RETE

```

TRANSMIT:

RETE

```

;*****
; DESCRIPTION: This routine initializes the 'C50 serial port and the TLC320C40's (AIC)
; TA, RA, TB, RB and control registers
;*****

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 AICINIT: SPLK #20h,TCR ; To generate 10 MHz from Tout
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SPLK #01h,PRD      ; for AIC master clock
MAR  *,AR0
LACC #0008h        ; Non continuous mode
SACL SPC           ; FSX as input
LACC #00c8h        ; 16 bit words
SACL SPC
LACC #080h         ; Pulse AIC reset by setting it low
SACH DXR
SACL GREG
LAR  AR0,#0FFFFh
RPT  #10000        ; and taking it high after 10000 cycles
LACC *,0,AR0       ; (0.5ms at 50ns)
SACH GREG
;-----
LDP  #TA
SETC .SXM
LACC TA,9          ; Initialize TA and RA register
ADD  RA,2
CALL AIC_2ND
;-----
LDP  #TB
LACC TB,9          ; Initialize TB and RB register
ADD  RB,2
ADD  #02h
CALL AIC_2ND
;-----
LDP  #AIC_CTR
LACC AIC_CTR,2     ; Initialize control register
ADD  #03h
CALL AIC_2ND

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RET

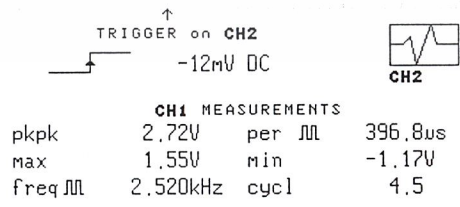
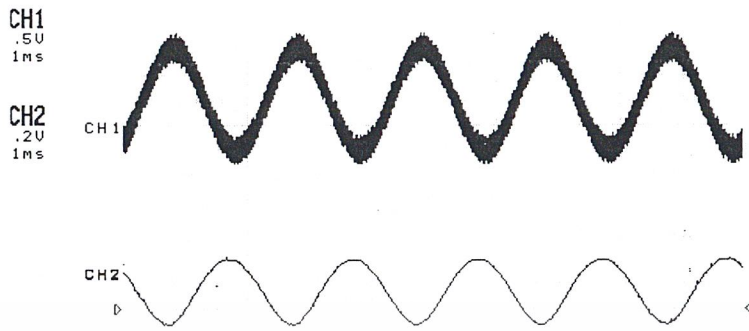
AIC_2ND:

```

LDP #0 ; Data page point is 0 (MM reg.)
SACH DXR ; send ACC high 00
CLRC INTM ; enable interrupts
IDLE ; wait for interrupt
ADD #6h,15 ; 0000 0000 0000 0011 XXXX XXXX XXXX XXXX b
SACH DXR ; send ACC high to initiate secondary protocol
IDLE ; wait for interrupt
SACL DXR ; send the T register data
IDLE ; wait for interrupt
LACL #0 ; clear ACC low
SACL DXR ; send another to make sure 1st word got sent
IDLE ; wait for interrupt
SETC INTM
RET
.END

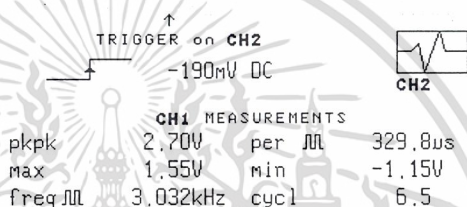
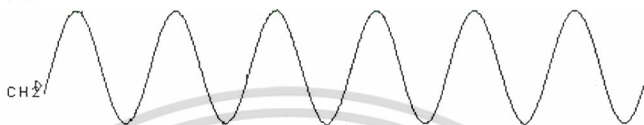
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

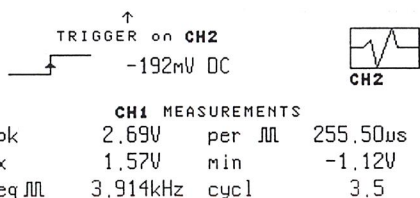
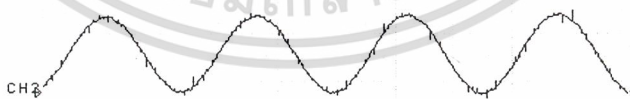
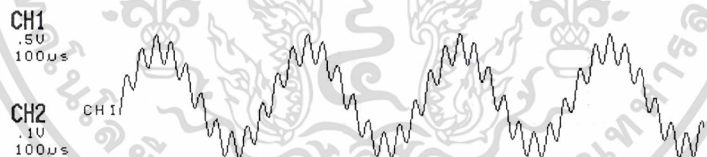


รูปที่ 4.9 ความถี่ Input 2,520 Hz สัญญาณรบกวน 38 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

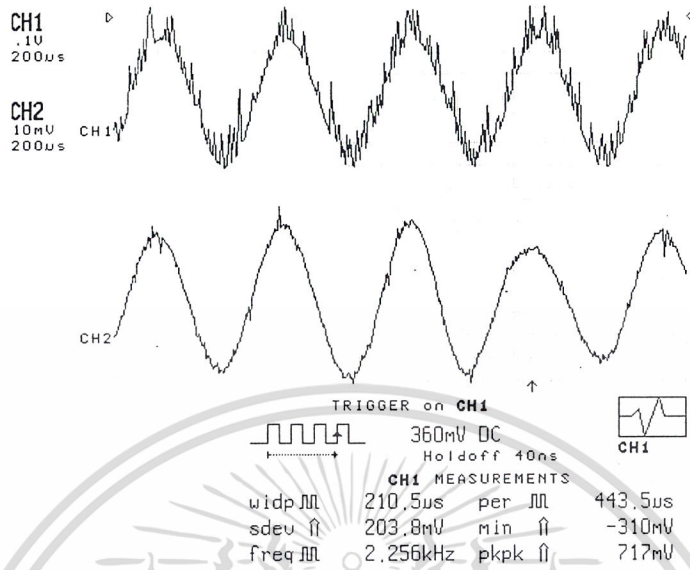


รูปที่4.10 ความถี่ Input 3,032 Hz สัญญาณรบกวน 38 kHz

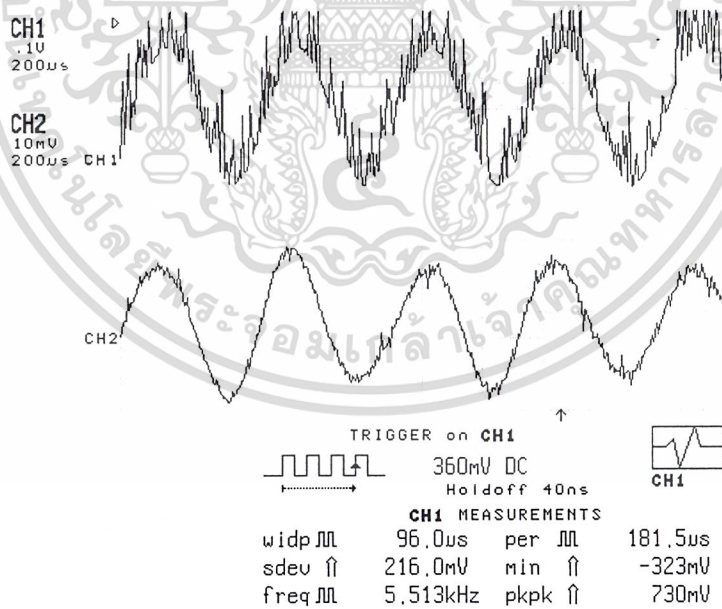


รูปที่4.11 ความถี่ Input 3,914 Hz สัญญาณรบกวน 38 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 ความถี่ Input 2,252 Hz สัญญาณรบกวนแบบเกาส์ S/N 20dB



รูปที่ 4.13 ความถี่ Input 2,252 Hz สัญญาณรบกวนแบบเกาส์ S/N 10dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ได้ทำการบันทึกมาจาก Storage Oscilloscope จะเห็นการกำจัดสัญญาณรบกวนออกจากสัญญาณขาขึ้นอยู่ในขั้นที่ดีมาก ความถี่ที่ใช้งานจะอยู่ในช่วง 500 Hz ถึง 4,000 Hz ในรูปที่ 8 ถึงรูปที่ 11 ใช้สัญญาณรบกวนที่คงที่ แต่ในรูปที่ 12 และรูปที่ 13 ใช้สัญญาณรบกวนแบบเกาส์ และกำหนดให้สัญญาณต่อสัญญาณรบกวนมีค่า 20dB และ 10 dB แต่ในรูปที่ได้บันทึกมาอ่านความถี่ได้ 2,256 kHz และ 5,513 kHz อาจเกิดจากการผิดพลาดจาก Storage Oscilloscope



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

สรุปผลการทดลอง

จากการศึกษาวิเคราะห์และออกแบบเพื่อหา Adaptive Algorithm แบบใหม่ที่ใช้สำหรับ ดีเทคต์สัญญาณชานด์คลื่นเดียว โดยใช้วงจรกรองความถี่ไอโออาร์นอทซ์ฟิลเตอร์ และสามารถปรับค่า Q-Factor ได้ ซึ่งสามารถควบคุมค่า Q-Factor เป็นไปได้อย่างต่อเนื่อง โดยตอนเริ่มต้นจะให้ค่า Q-Factor ต่ำ และเมื่อ Adaptive Algorithm เริ่ม Converge แล้ว จะให้ค่า Q-Factor สูง จากผลของการวิเคราะห์ทำให้สามารถออกแบบ Adaptive Algorithm ขึ้นใหม่ได้ วงจรจะมีส่วนประกอบของอุปกรณ์น้อยลงและง่ายขึ้น และจากผลการทดลองด้วยการ Simulate ด้วยโปรแกรมคอมพิวเตอร์ จะเห็นได้ว่า Adaptive Algorithm แบบใหม่นี้ ยังสามารถติดตามค่าจัดสัญญาณรบกวนทาง Input ตามความถี่ที่ใช้งานได้เร็วกว่า Adaptive Algorithm แบบเดิมอีกด้วย

จากกราฟที่ 1 เป็นกราฟแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ ω พบว่าค่าของ α_0 มีผลทำให้ค่า Q-Factor หรือความกว้างของการตอบสนองของ $|H(\omega)|$ เปลี่ยนแปลงไปโดยค่า Q-Factor จะแปรผันตรงกับค่าของ α_0 และค่าของ α_1 เป็นค่าที่กำหนดจุดการตอบสนองสูงสุดของ $|H(\omega)|$ โดยจุดตอบสนองสูงสุดจะเบี่ยงเบนไปด้านซ้ายถ้า α_1 มีค่าน้อยกว่าศูนย์ และเบี่ยงเบนไปด้านขวาถ้า α_1 มีค่ามากกว่าศูนย์

กราฟที่ 2 เป็นกราฟแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ α_1 พบว่ามีจุดต่ำสุดเพียงจุดเดียวเท่านั้น จึงสามารถใช้วิธี Stochastic Gradient ได้

กราฟที่ 3 เป็นกราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีที่ยังไม่ลดตัวแปร พบว่าค่า μ จะมีผลทำให้การลู่เข้าของกราฟเร็วขึ้น โดยที่ถ้าค่า μ มีค่ามากก็จะลู่เข้าเร็ว แต่ถ้าค่าน้อยก็จะลู่เข้าช้า ค่า μ เป็นความละเอียดของการลู่เข้าถ้ามีค่ามากจะทำให้ลู่เข้าได้เร็ว และถ้ามากเกินไปจะทำให้ลู่เข้าอย่างขาดเสถียรภาพ ค่า ω จะต้องลู่เข้าที่จุด $\alpha_1 = \cos(\omega)$ และค่าของ α_0 จะทำให้การลู่เข้าช้าลงถ้าค่าของ α_0 มีค่ามาก (เข้าใกล้หนึ่ง) และจะเร็วถ้ามีค่าน้อย (เข้าใกล้ศูนย์)

กราฟที่ 4 เป็นกราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k กรณีปรับตัวแปรแล้ว พบว่ามีลักษณะเหมือนกันกับกราฟที่ 3 จึงนำสมการไปคำนวณได้

กราฟที่ 5 เป็นกราฟแสดงคุณสมบัติของ $|H(\alpha_0(k))|$ เทียบกับ α_0 จะพบว่าไม่ว่าจะเปลี่ยนค่าตัวแปร α_1 และ ω เป็นค่าใดก็ตามกราฟการตอบสนองของ $|H(\alpha_0(k))|$ ก็ยังมีความสูงอยู่ยอดเดียวเท่านั้น จึงสามารถใช้วิธี Stochastic Gradient ได้

กราฟที่ 6 เป็นกราฟแสดงคุณสมบัติของ $|G(z)|, |G_{01}(z)|$ และ $|G_{02}(z)|$ เทียบกับ ω เป็นเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า การแสดงคุณสมบัติของสัญญาณควบคุมแต่ละตัว ไม่ว่าการณ์ใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กราฟที่ 7 เป็นกราฟแสดงคุณสมบัติของ $|H(e^{j\omega})|$ เทียบกับ α_0 จะพบว่ากราฟไม่มีขดสูงสุด ทำให้เราวิเคราะห์ได้ว่าสามารถประมาณให้ $y(k) * \varphi(k) \approx \varphi(k)$ ทำให้สามารถปรับปรุงระบบให้ดีขึ้นได้

กราฟที่ 8 เป็นกราฟแสดงคุณสมบัติของ $|\Pi(\alpha_0(k))|$ เทียบกับ α_0 แสดงคุณสมบัติ Transfer Function เมื่อได้ผ่านการปรับปรุงและตัดแปลงสมการใหม่ โดยให้ $y(k) * \varphi(k) \approx \varphi(k)$ แล้วทำการพลอต จะเห็นได้ว่ามีขดสูงเพียงขดเดียว เหมือนกับกราฟในข้อที่ 5 ทำให้ทราบได้ว่าสามารถปรับปรุงให้ $y(k) * \varphi(k) \approx \varphi(k)$ ได้

กราฟที่ 9 เป็นกราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter กับ Simplify Adaptive IIR Notch Filter พบว่าสัญญาณ Simplify Adaptive แบบใหม่ที่สามารถ Convergence สัญญาณได้เร็วกว่าแบบเดิม โดยกำหนดสัมประสิทธิ์ของตัวแปรต่างๆ เหมือนกันทุกประการ โดยกำหนดให้ ω มีค่า 60 และ $(\alpha_0(k))$ และ $\alpha_1(k)$ เป็นสัมประสิทธิ์แปรค่าได้

กราฟที่ 11 เป็นกราฟแสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter กับ Simplify Adaptive IIR Notch Filter พบว่าสัญญาณ Simplify Adaptive แบบใหม่สามารถ Convergence สัญญาณได้เร็วกว่าแบบเดิม โดยกำหนดสัมประสิทธิ์ของตัวแปรต่างๆ เหมือนกันทุกประการ โดยกำหนดให้ ω มีค่า 60 และ SNR มีค่า 1.8 จะมีสัญญาณรบกวนมากกว่ากราฟที่ 9 และกราฟที่ 10 เพราะมีค่า SNR น้อย

กราฟที่ 12 แสดงการ Convergence ของ $\alpha_1(k)$ ของ Adaptive IIR Notch Filter เปรียบเทียบกับสมการ Adaptive IIR Band Pass Filter จะเห็นได้ว่า Adaptive IIR Band Pass Filter สามารถ Converge ได้เร็วกว่า เนื่องจากสมการมีตัวแปรน้อยกว่า

จากการ Simulate จะเห็นได้ว่าสัญญาณรบกวนที่ป้อนเข้ามากับสัญญาณ Input สามารถกำจัดได้โดย Adaptive IIR Notch Filter และ Output ที่ได้มีลักษณะสัญญาณใกล้เคียงกับสัญญาณทาง Input จริงเมื่อ α_0 คู่เข้าสู่ค่าคงที่ และจากการเขียนโปรแกรมบนบอร์ด DSP เพื่อทำการแยกสัญญาณชาน์นออกจากสัญญาณรบกวนโดยกำหนดให้ค่า Q-Factor คงที่ จะเห็นได้ว่าสามารถกำจัดสัญญาณรบกวนได้ในระดับที่ดีโดยไม่จำเป็นต้องรู้ว่าความถี่ของสัญญาณ Input มีค่าเท่าใดแต่สัญญาณ Input ต้องอยู่ในย่าน Nyquist Frequency ความถี่ที่ใช้งานอยู่ในช่วง 600 Hz ถึง 4,000 Hz และค่า SNR ประมาณ 0 dB ขึ้นไป

แนวทางในการทำการพัฒนา

1. เขียนโปรแกรม Simplify Adaptive IIR Notch Filter บนบอร์ด DSP
2. ทำให้อะแดปทีฟสามารถทำงานได้ในย่านความถี่สูงซึ่งต้องสร้าง A/D ตลอดจนอุปกรณ์ประมวลผลสัญญาณดิจิทัลที่มีความไวสูงหรือให้ทำงานในแบบขนาน
3. ทำการทดลองโดยการสร้างการ์ด Interface กับคอมพิวเตอร์ PC ที่มีรูปแบบการคำนวณในรูปของ 32 bit และใช้การเขียนโปรแกรมด้วยภาษาชั้นสูงในคอมพิวเตอร์ โดยเลือกใช้ D/A และ A/D ที่มี Sampling Rate สูงๆ จะสามารถที่ทำการทดลองง่ายขึ้นได้กว้างขึ้น เช่นสามารถทดลองในช่วงความถี่กว้างขึ้นมีการคำนวณในรูปที่ทำการประมวลค่าน้อยลงและสามารถที่จะนำไปใช้งานจริงได้
4. ทำการทดลองเขียนโปรแกรมบน Chip DSP ที่มีการคำนวณในรูปทศนิยม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- (1) Widrow B., Stearn S. “ **Adaptive Signal Processing**”, Prentice-Hall, Inc.Englewood Cliffs, 1985
- (2) Richard G. Lyons “**Understand Digital Signal Processing**”, IEEE Proc.,pp.561-751, April 1978
- (3) W. Kenneth Jenkins, Andrew W. Hull, Jeffrey C. Strait, Bernard A. Schnauffer and Xiaohui Li “**Advanced Concepts In Adaptive Signal Processing**”
- (4) Simon Haykin “**Introduction To Adaptive Filter**”, Macmillan Publishing Company, Newyork
- (5) Alan V. Oppenheim and Ronald W. Schafer “**Digital Signal Processing**”, Prentice-Hall International, Alan V. Oppenheim and Bell Telephone Laboratories, Inc., 1975
- (6) Alan, V.Oppenheim and Ronald, W.Schafer, “ **DISCREAT-TIME SIGNAL PROCESSING** “ ,Prentice-Hall , Englewood Cliffs , NJ , 1989
- (7) D.M.Etter , “ **ENGINEERING PROBLEM SOLVING WITH MATLAB** “ , Prentice-Hall Simon & Schuster (Asia) . 1996
- (8) Digital Signal Processing Products , “ **TMS320C5x User ‘s Guide** “ , Texas Instrument : 1993
- (9) Microprocessor Development Systems, “ **TMS320C5x DSP Starter Kit User ‘s Guide**”, Texas Instruments: 1994
- (10)Sanjit,K.Mitra , “ **Handbook for Digital Signal Processing** “ , John Wiley & Sons , Inc ,California, 1993
- (11)C.Benjangkprasert, K.Kikuchi, N.Takahashi, T.Takabe.,”An Implementation of High-Speed Adaptive Noise canceller with Parallel Block Structure”,Proc.APCCAS’94,pp.376-381
- (12)อิสราวุฒิ ชัยศิริ, ชูชาติ วงศ์ศิลป์, วรวิทย์ สมหา, ปัญญา ฐิติมัทธินิมา “ **ตัวกรองความถี่ชนิดปรับค่าได้ที่เวลาจริงโดยใช้ TMS320C50** “, การประชุมวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 20 ภาค วิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย วันที่ 13-14 พฤศจิกายน 2540 หน้า 313-317

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TLC32040C, TLC32040I, TLC32041C, TLC32041I
TLC32042C, TLC32042I
ANALOG INTERFACE CIRCUITS

SLAS014D - D2964, SEPTEMBER 1987 - REVISED MAY 1991

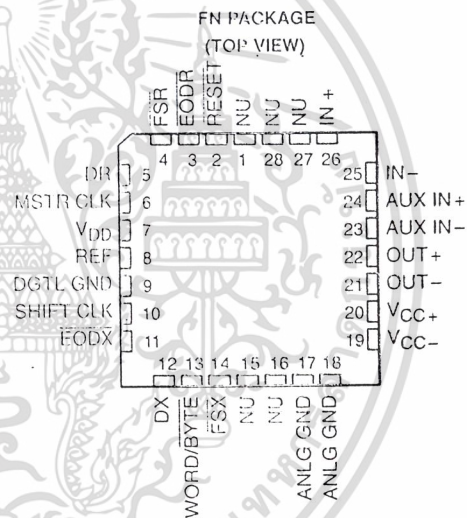
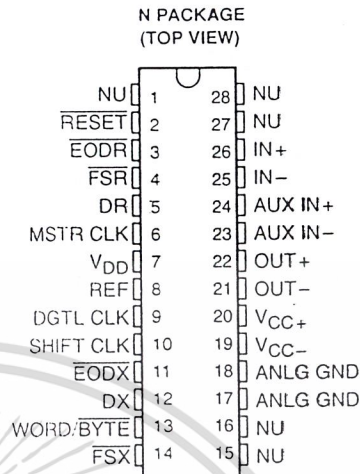
- Advanced LinCMOS™ Silicon-Gate Process Technology
- 14-Bit Dynamic Range ADC and DAC
- Variable ADC and DAC Sampling Rate Up to 19,200 Samples per Second
- Switched-Capacitor Antialiasing Input Filter and Output-Reconstruction Filter
- Serial Port for Direct Interface to TMS32011, TMS320C17, TMS32020, and TMS320C25 Digital Signal Process
- Synchronous or Asynchronous ADC and DAC Conversion Rate With Programmable Incremental ADC and DAC Conversion Timing Adjustments
- Serial Port Interface to SN74299 Serial-to-Parallel Shift Register for Parallel Interface to TMS32010, TMS320C15, or Other Digital Processors
- 600-Mil Wide N Package (C_L to C_L)

PART NUMBER	DESCRIPTION
TLC32040	Analog interface circuit with internal reference. Also a plug-in replacement for TLC32041.
TLC32041	Analog interface circuit without internal reference
TLC32042	Identical to TLC32040, but has a slightly wider bandpass filter bandwidth

description

The TLC32040, TLC32041, and TLC32042 are complete analog-to-digital and digital-to-analog input/output systems, each on a single monolithic CMOS chip. This device integrates a bandpass switched-capacitor antialiasing input filter, a 14-bit-resolution A/D converter, four microprocessor-compatible serial port modes, a 14-bit-resolution D/A converter, and a low-pass switched-capacitor output-reconstruction filter. The device offers numerous combinations of master clock input frequencies and conversion/sampling rates, which can be changed via digital processor control.

Typical applications for this integrated circuit include modems (7.2-, 8-, 9.6-, 14.4-, and 19.2-kHz sampling rate), analog interface for digital signal processors (DSPs), speech recognition/storage systems, industrial process control, biomedical instrumentation, acoustical signal processing, spectral analysis, data acquisition, and instrumentation recorders. Four serial modes, which allow direct interface to the TMS32011, TMS320C17, TMS32020, and TMS320C25 digital signal processors, are provided. Also, when the transmit and receive



NU - Nonusable; no external connection should be made to these pins.

Advanced LinCMOS™ is a trademark of Texas Instruments Incorporated

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1991, Texas Instruments Incorporated



POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

B-3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

description (continued)

sections of the analog interface circuit (AIC) are operating synchronously, it will interface to two SN74299 serial-to-parallel shift registers. These serial-to-parallel shift registers can then interface in parallel to the TMS32010, TMS320C15, other digital signal processors, or external FIFO circuitry. Output data pulses are emitted to inform the processor that data transmission is complete or to allow the DSP to differentiate between two transmitted bytes. A flexible control scheme is provided so that the functions of this integrated circuit can be selected and adjusted coincidentally with signal processing via software control.

The antialiasing input filter comprises seventh-order and fourth-order CC-type (Chebyshev/elliptic transitional) low-pass and high-pass filters, respectively and a fourth-order equalizer. The input filter is implemented in switched-capacitor technology and is preceded by a continuous time filter to eliminate any possibility of aliasing caused by sampled data filtering. When no filtering is desired, the entire composite filter can be switched out of the signal path. A selectable, auxiliary, differential analog input is provided for applications where more than one analog input is required.

The A/D and D/A converters each have 14 bits of resolution. The A/D and D/A architectures ensure no missing codes and monotonic operation. An internal voltage reference is provided on the TLC32040 and TLC32042 to ease the design task and to provide complete control over the performance of this integrated circuit. The internal voltage reference is brought out to a pin and is available to the designer. Separate analog and digital voltage supplies and grounds are provided to minimize noise and ensure a wide dynamic range. Also, the analog circuit path contains only differential circuitry to keep noise to an absolute minimum. The only exception is the DAC sample and hold, which utilizes pseudo-differential circuitry.

The output-reconstruction filter is a seventh-order CC-type (Chebyshev/elliptic transitional low-pass filter followed by a fourth-order equalizer) and is implemented in switched-capacitor technology. This filter is followed by a continuous-time filter to eliminate images of the digitally encoded signal.

The TLC32040C, TLC32041C, and TLC32042C are characterized for operation from 0°C to 70°C, and the TLC32040I, TLC32041I and TLC32042I are characterized for operation from -40°C to 85°C.

