



การออกแบบวงจรกรองดิจิตอลชนิดเวลาจริง

Real-Time Digital Filter Design

โดย

นายภิญโญ

อิมเอิบ

นายธงชัย

เนียรสวัสดิ์

นายชาญชัย

พฤกษ์พาดิ

วัน เดือน ปี.....	๑	๙๐.	๒๕๔๐
เลขทะเบียน.....	๐๖๖๖๖		
เลขเรียกหนังสือ.....	๖	๓๘๖๖๔	๒๑๕๒๔ ก.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมการวัดคุมทางอุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา ๒๕๓๘

การออกแบบวงจรกรองดิจิตอลชนิดเวลาจริง

Real-Time Digital Filter Design

โดย

นายบุญญ โธม

อิมเอิบ

นายชงชัย เนียรสวัสดิ์

นายชาญชัย พงษ์พาณิชย์

อาจารย์ที่ปรึกษา

รศ.ดร.บุญศักดิ์ ชิวสุวิทย์

(.....)

อาจารย์สักริษา ชิตวงค์

(.....)

บทคัดย่อ

ระบบประมวลผลสัญญาณเชิงเลข หรือดิจิตอลซิกแนลโปรเซสซิง เป็นระบบประมวลผลสัญญาณอีกแบบหนึ่ง ที่ได้รับการพัฒนาอย่างมีประสิทธิภาพ และถูกนำไปประยุกต์ใช้งานในด้านต่างๆ เช่น ระบบควบคุม ระบบสื่อสารข้อมูล และอุปกรณ์การแพทย์ สำหรับการประมวลผลสัญญาณดิจิตอลเบอร์ TMS 320C50 นั้น เป็นชิพที่ บริษัท เท็กซัส อินสตรูเมนต์ จำกัด ได้ผลิตขึ้นมา เพื่อการประมวลผลที่มีความเร็ว สูง สามารถเปลี่ยนแปลงอัลกอริทึม ในการทำงานได้ มีสถาปัตยกรรมภายในไม่ซับซ้อน และราคาไม่สูงมาก จึงเป็นชิพที่เหมาะสมสำหรับผู้เริ่มต้นศึกษา การประมวลผลสัญญาณเชิงดิจิตอลเป็นอย่างมาก

ในโครงการชุดนี้ จะกล่าวถึงการใช้งานของ TMS 320C50 ในการประมวลผลสัญญาณดิจิตอลคือ การกรองเชิงดิจิตอล (Digital Filtering) ซึ่งอัลกอริทึมของการประมวลผลสัญญาณนี้ ไม่ซับซ้อนมาก อีกทั้งยัง ใช้ได้จริงในงานต่างๆ โดยสามารถที่จะนำมาใช้ในการทดลอง และพัฒนาต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSTRACT

Digital signal Processing or DSP in another type of the processing system. DSP system has been developed for modern theories and many applications in control system, telecommunications, Medical instrumentations, Image Processing, etc. TMS 320C50 is Product of the Texas Instruments DSP chips. For real-time signal processing, task flexibility, basic chip architecture, and inexpensive. TMS 320C50 is a good device for the novice in Digital signal processing developing field.

The purpose of this thesis is use the TMS 320C50 to Digital signal Processing for Digital filtering. So further development in high technology, which will be regulated by DSP aspects, can be created successively.

สารบัญ

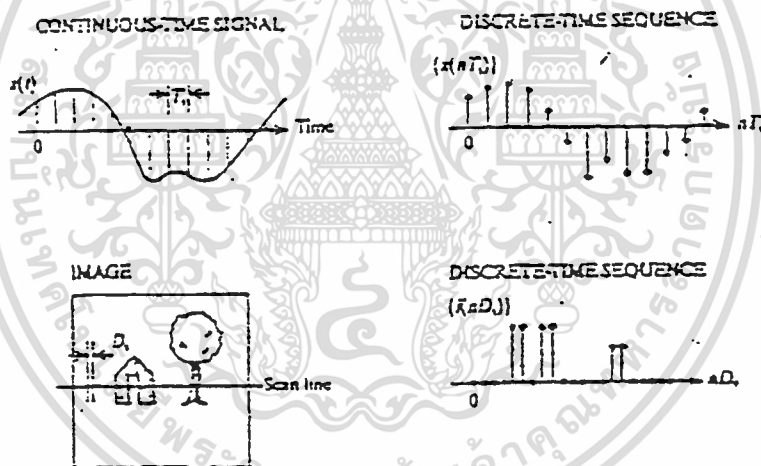
	หน้า
บทที่1 บทนำ	
บทที่2 ทฤษฎี และ หลักการ	2-1
-ทฤษฎีการสุ่มตัวอย่าง	2-2
-Linear Time Invariant System	2-3
-Convolution Sum	2-5
-Stable and Causal of System	2-8
-Fourier Transforms	2-9
-Even and Odd Functions	2-14
-Z-Transform	2-25
-Delay Operator	2-28
บทที่3 การออกแบบ และ การสร้าง	
-การออกแบบโดยใช้ Fourier Series	3-1
-การลด Ripple โดยใช้ Sigma Factor	3-6
-การออกแบบ โดยใช้ Window	3-11
บทที่4 ผลการทดลอง	4-1
บทที่5 สรุปผลการทดลอง	5-1
ภาคผนวก	
เอกสารอ้างอิง	

บทที่ 1

บทนำ (Introduction)

Signal คือ Information ที่อาจอยู่ในรูปแบบต่างๆ กัน เช่น คลื่นวิทยุที่ส่งข่าวสารโดยปกติ สัญญาณจะมีค่าเปลี่ยนไปกับเวลา ในเชิงคณิตศาสตร์ สามารถกล่าวได้ว่าสัญญาณเป็นฟังก์ชันของเวลา เวลาจะเป็นตัวแปรอิสระ (Independent Variable)

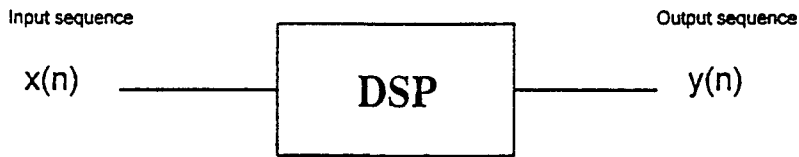
Continuous-time signal คือสัญญาณที่มีค่าต่อเนื่องตลอดเวลา สัญญาณจะมีค่าเสมอทุกช่วงเวลาเมื่อนำเอาขนาดของ Continuous-time signal มาพล็อตกับเวลา จะได้เป็นเส้นที่ต่อเนื่องกัน แต่ Discrete-time signal เป็นสัญญาณที่มีค่าเป็นช่วงๆ สัญญาณจะมีค่าเฉพาะบางช่วงเวลาเท่านั้น ตัวแปรอิสระ เวลา จะมีค่าไม่ต่อเนื่อง มีค่าเฉพาะบางจุด (Discrete) ปกติแล้ว Discrete-time signal จะอยู่ในรูปของ Sequence ของตัวเลขเมื่อขนาดของ Discrete-time signal หรือ Discrete-time Sequence มีค่าได้เฉพาะบางค่าเท่านั้น คือทั้งเวลาและขนาด มีค่า Discrete จะเรียกสัญญาณนี้ว่า Digital Signal Processing สำหรับ Continuous-time Continuous-amplitude signal จะเรียกว่า Analog Signal



รูปที่ 1.1 การหา Discrete-time sequence จาก analog signal

Digital Signal Processing (DSP)

คือกระบวนการแปลงสัญญาณดิจิทัลที่ป้อนเข้ามา ให้อยู่ในรูปที่เหมาะสมสำหรับการนำเอา Information ที่อยู่ในสัญญาณนั้นมาใช้ หรือคือการแปลง Input Sequence โดยใช้ Numerical Procedure หรือ Algorithm บางอย่างให้ได้ Output Sequence ที่มีคุณสมบัติที่ดีกว่าเช่น มี Noise ลดลง



รูปที่ 1.2 DSP เปลี่ยน Input Sequence $x(n)$ ให้เป็น Output Sequence $y(n)$

สำหรับ input sequence นั้นอาจได้มาจากการสุ่ม (Sampling) สัญญาณอนาล็อก ทุกๆ ช่วงเวลาหนึ่ง ช่วงเวลานี้เรียกว่า Sampling period T_s เมื่อสัญญาณอนาล็อกถูกแทนด้วย $x(t)$ ค่าของ discrete-time sequence จะมีค่า

$$X(nT_s) = X(t) \Big|_{t=nT_s}$$

เนื่องจาก Sampling period จะมีค่าคงที่สำหรับงานหนึ่ง จึงสามารถตัดออกไปได้ โดยจะแทน discrete-time sequence ด้วย $x(n)$

การ Sampling นี้คือการเปลี่ยนสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัล ซึ่งทำได้โดยอุปกรณ์ที่เรียกว่า A/D converter เมื่อต้องการแปลง Output sequence ให้กลับไปเป็น สัญญาณอนาล็อก ใช้ D/A converter



รูปที่ 1.3 การเปลี่ยนไปมาระหว่าง analog signal กับ digital signal

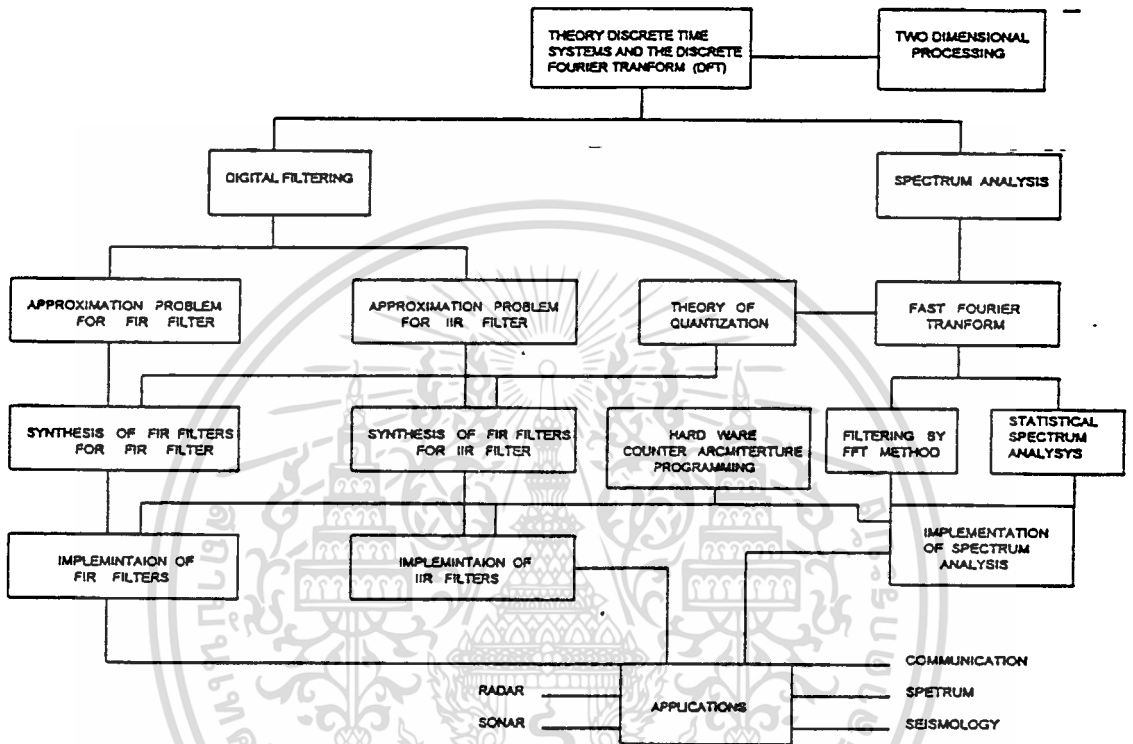
Overview of Digital Signal Processing

พื้นฐานทางทฤษฎีของ Digital Signal Processing คือ Theorie of discrete-time linear time-invariant system งานสามารถแบ่งย่อยเป็น 2 ด้าน ได้แก่ Digital Filtering และ Spectrum analysis ทางด้าน Digital Filtering ยังแบ่งได้ตามชนิดของ filter คือ Finite Impulse Response (FIR) filter และ Infinite Impulse Response (IIR) filter ในงานทาง Spectrum analysis แบ่งตามวิธีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนวณหา Spectrum เป็นการคำนวณโดยใช้ Discrete Fourier Transform (DFT) และโดยวิธีการทางสถิติ เช่นในกรณีของ random signal

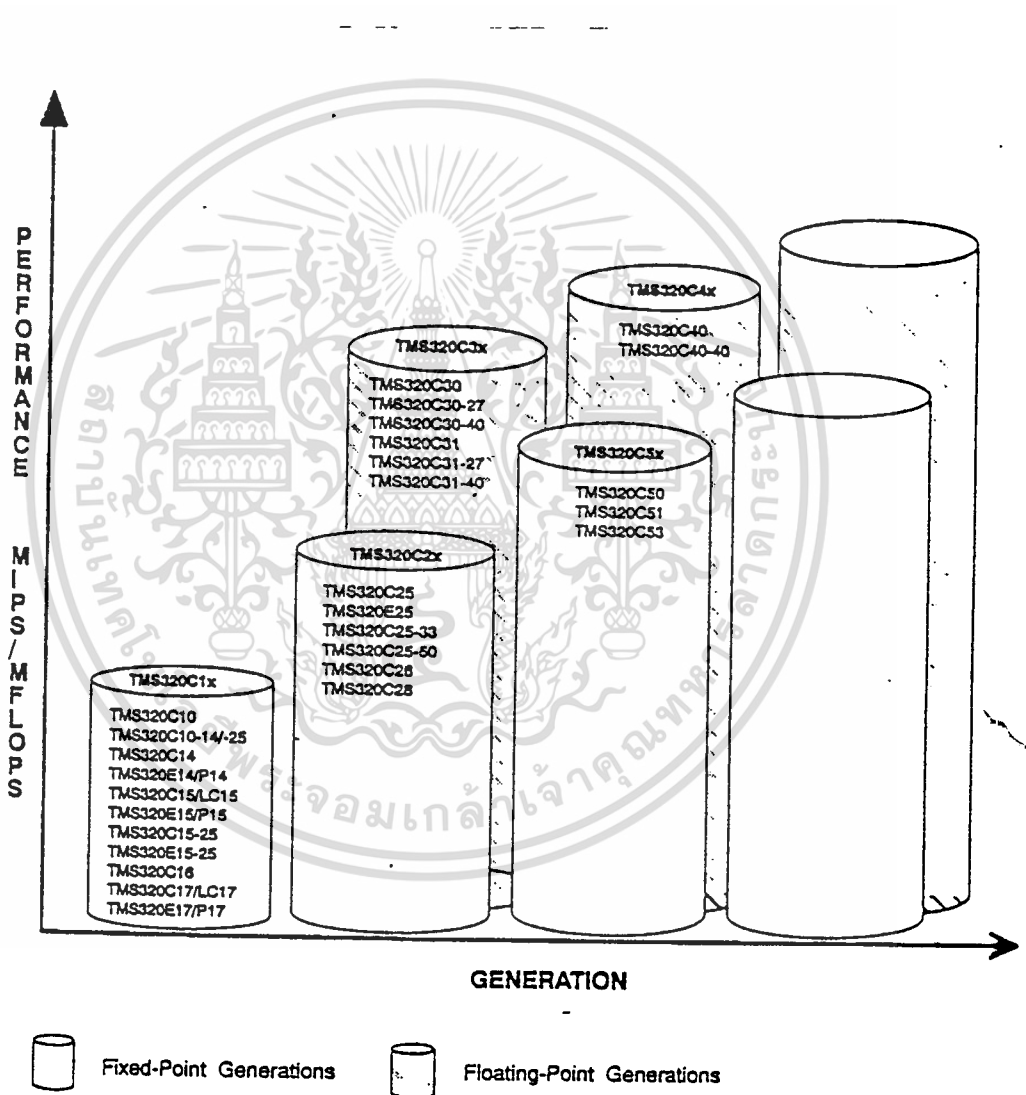
Fast Fourier Transform (FFT) เป็น algorithm ที่ใช้ในการหา DFT ได้อย่างรวดเร็วจึงมีความสำคัญมากสำหรับ Spectrum analysis จากทฤษฎีพื้นฐานเกี่ยวกับ discrete-time ยังได้มีการพัฒนาไปเป็น two-dimensional signal processing ซึ่งขณะนี้กำลังมีการวิจัยทางด้านนี้เป็นจำนวนมาก



รูปที่ 4. Overview of digital signal processing

ความก้าวหน้าทางอิเล็กทรอนิกส์ครั้งใหญ่ เกิดขึ้นเมื่อได้มีการพัฒนาสร้างดิจิทัลซิกแนลโปรเซสเซอร์ แบบความเร็วสูง (High-speed signal Processor) ซึ่งในปัจจุบันมีใช้กันอย่างแพร่หลายในราคาที่ไม่แพงนัก มีขนาดเล็กกว่าเดิมแต่มีความสามารถในการทำงานได้สูงขึ้น

จากรูป 1.5 แสดงวิวัฒนาการของดิจิทัลซิกแนลโปรเซสเซอร์ตระกูล TMS 320x จะพบว่าในระยะแรก ๆ ความสามารถในการทำงานไม่ค่อยสูงนัก เมื่อเทียบกับปัจจุบัน ที่สามารถที่จะทำงานในระบบวงจรรวมขนาดใหญ่ (VLSI) ได้อย่างคล่องตัว



รูปที่ 1.5 วิวัฒนาการของตระกูล TMS 320X

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

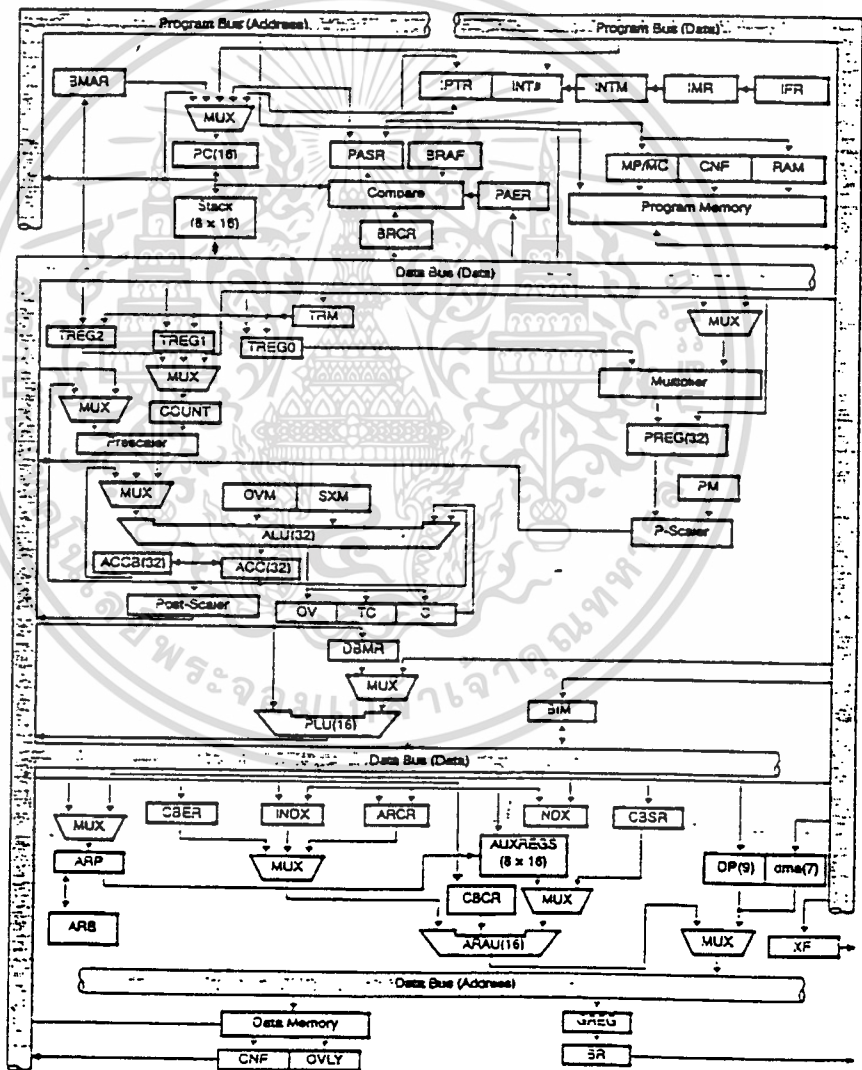
จากคุณสมบัติพิเศษและการประมวลผลแบบเวลาจริง (REAL-TIME) คือเมื่อเราใส่สัญญาณอินพุตเข้าไป ก็จะได้สัญญาณเอาต์พุตออกมาเลย โดยมีเวลาดำเนินการ (Delay) น้อยมากทำให้การประมวลผล ดังกล่าวทำได้อย่างคล่องตัว มีความเร็วในการประมวลผลสูง ซึ่งสามารถประยุกต์ใช้งานได้หลายรูปแบบ ดังแสดงในตารางที่ 1.1

Automotive	Consumer	Control
Adaptive Ride Control Antiskid Brake Cellular Telephone Digital Radio Engine Control Global Positioning Navigation Vibration Analysis Voice Commands	Digital Radio/TV Educational Toys Music Synthesizer Power Tools Radar Detector Solid-State Answering Machines	Disk Drive Control Engine Control Laser Printer Control Motor Control Robotics Control Servo Control
General-Purpose	Graphics/Imaging	Industrial
Adaptive Filtering Convolution Correlation Digital Filtering Fast Fourier Transforms Hilbert Transforms Waveform Generation Windowing	3-D Rotation Animation/Digital Map Homomorphic Processing Pattern Recognition Image Enhancement Image Compression/ Transmission Robot Vision Workstations	Numeric Control Power-Line Monitoring Robotics Security Access
Instrumentation	Medical	Military
Digital Filtering Function Generation Pattern Matching Phase-Locked Loops Seismic Processing Spectrum Analysis Transient Analysis	Diagnostic Equipment Fetal Monitoring Hearing Aids Patient Monitoring Prosthetics Ultrasound Equipment	Image Processing Missile Guidance Navigation Radar Processing Radio Frequency Modems Secure Communications Sonar Processing
Telecommunications		Voice/Speech
1200- to 19200-bps Modems Adaptive Equalizer ADPCM Transcoder Cellular Telephone Channel Multiplexing Data Encryption Digital PBXs Digital Speech Interpolation (DSI)	DTMF Encoding/Decoding Echo Cancellation FAX Line Repeater Speaker Phone Spread Spectrum Communications Video Conferencing X.25 Packet Switching	Speech Enhancement Speech Recognition Speech Synthesis Speaker Verification Speech Vocoding Voice Mail Text-to-Speech

ตารางที่ 1.1 การประยุกต์ใช้งานของ DSP ตระกูล TMS 320X

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 1.6 แสดง Block Diagram ของ (TMS 320C50) ซึ่งแสดงให้เห็นว่ามีโครงสร้างหลักอยู่ 2 ส่วน คือ Program bus และ Data bus โดยโปรแกรม bus จะรับคำสั่ง และมีการส่งข้อมูลต่าง ๆ จากโปรแกรม memory ส่วน databus จะนำข้อมูลที่ได้อมาทำการคำนวณข้อมูลใน CALU และเก็บใน Register ของ Data memory ทั้งสองส่วนนี้ จะเป็นการนำ ข้อมูลจาก Data memory และ Program memory ทั้งภายในหรือภายนอกไปทำการคูณกัน ใน 1 รอบการทำงานคำนวณ



ในโครงการนี้จะใช้ TMS 320C50 มาเป็น DSP สำหรับประมวลผล โดยทำหน้าที่เป็น วงจรกรองเชิงเลข (Digital filter) ซึ่งมีการใช้โปรแกรม (Soft-ware) เข้าช่วยในการประมวลผล มีการประมวลค่า ส.ป.ส. ต่าง ๆ สำหรับการกรองแบบ lowpass, high pass, Band pass และ Band stop ทั้งนี้จะมีการแสดงผลการทดลองและผลลัพธ์ที่ได้โดยการเปรียบเทียบอินพุตและเอาต์พุต อยู่ในบทที่ 4 ซึ่งจะได้กล่าวต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

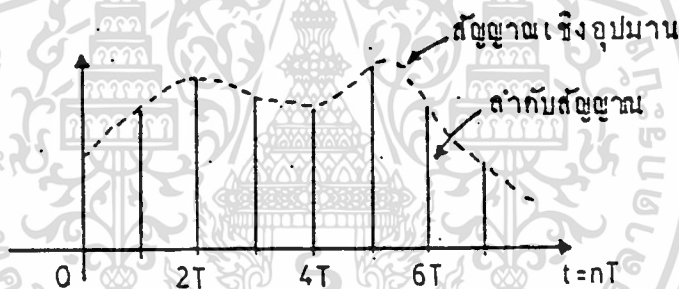
ทฤษฎี และหลักการ

บทนิยามทางคณิตศาสตร์

1. Sequence ถ้าหากเรามี Analog Signal $x(t)$ แล้วเราทำการสุ่มสัญญาณนี้ด้วยช่วงเวลา T วินาที (Sampling Interval) เท่ากันโดยตลอด สิ่งที่ได้ต่อไปนี้จะเรียกว่า Sequence ซึ่งเขียนแทนด้วยสมการ

$$x(t) = x(nT) = x(t) \Big|_{t=nT}$$

โดยที่ $n = 0, 1, 2, \dots$ ซึ่งเป็นค่าคงตัวเต็มหน่วย และ โดยทั่วไปใช้ $T = 1$ สมการนี้แสดงว่าลำดับ $x(n)$ ถูกสุ่มตัวอย่างมาจาก Analog Signal $x(t)$ ทุกคาบเวลา nT วินาที



รูปที่ 2.1 ลำดับที่ได้จากการสุ่มตัวอย่างฟังก์ชันต่อเนื่อง

2. Digital Impulse ซึ่งนิยามได้เป็น

$$\delta(n) = \begin{cases} 1 & ; n = 0 \\ 0 & ; n \neq 0 \end{cases}$$

3. Digital Impulse ที่หน่วงไป k วินาที นิยามได้เป็น

$$\delta(n-k) = \begin{cases} 1 & ; n = k \\ 0 & ; n \neq k \end{cases}$$

4. Unit step Sequence นิยามได้เป็น

$$u(n) = \begin{cases} 1 & ; n \geq 0 \\ 0 & ; n < 0 \end{cases}$$

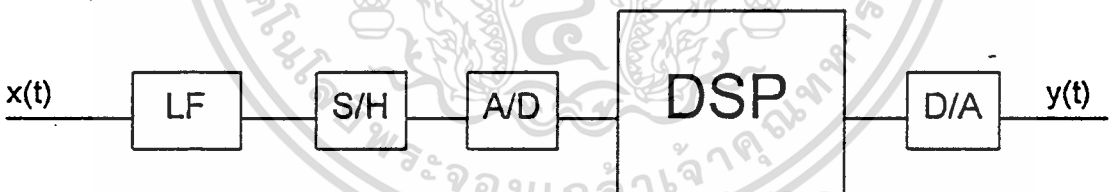
ซึ่ง Unit step Sequence นี้อาจเขียนอยู่ในรูปแบบของผลบวกของ Digital Impulse ได้คือ

$$x(n) = [x(0), x(1), x(2), \dots, x(m)]$$

$$\sum_{m=-\infty}^{\infty} x(m) \delta(n-m)$$

ทฤษฎีการสุ่มตัวอย่าง

อาจกล่าวได้ว่า สัญญาณในธรรมชาติส่วนมาก เช่น สัญญาณเสียง สัญญาณคลื่นสะท้อนของพื้นโลก คลื่นหัวใจ หรือ การแปรค่าไปของอุณหภูมิ เหล่านี้เป็นไปในลักษณะมีความสัมพันธ์กับเวลาที่เปลี่ยนแปลงไป หรือ กล่าวได้ว่าเป็นสัญญาณ Analog การนำสัญญาณเหล่านี้ไปประมวลผลในลักษณะของ Digital หรือ Discrete Signal Processing ได้ จะต้องใช้ระบบการประมวลผลดังรูปที่ 2.2



รูปที่ 2.2 ระบบการประมวลผลสัญญาณ Analog โดยใช้ DSP

ซึ่งตามรูป วงจร S/H เป็นวงจรสุ่มและคงค่าสัญญาณไว้ เพื่อให้วงจร A/D ทำการแปลงสัญญาณจาก Analog เป็น Digital อีกทีหนึ่ง

ในการแปลงสัญญาณ Analog เป็น สัญญาณ Digital ความถี่ในการสุ่มนั้นจะต้องมากพอที่จะไม่ทำให้สัญญาณสูญเสียข้อมูลที่สำคัญไป

ทฤษฎีการสุ่ม (Sampling Theory) ของ Shannon กล่าวไว้ว่า

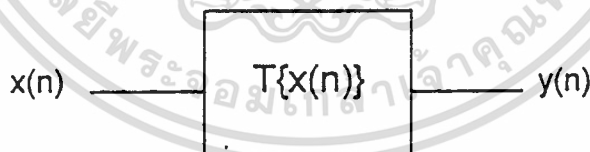
ถ้าหากเรามี Signal Analog $x(t)$ ที่ค่าการแปลง Fourier หรือ Power Spectrum ของสัญญาณ มีแถบความถี่ปฏิบัติงาน เท่ากับ f_0 แล้ว เราจะสามารถทำการสุ่มโดยที่สัญญาณไม่สูญเสียเนื้อหาที่สำคัญ ก็ต่อเมื่อความถี่ในการสุ่ม f_s มีค่ามากกว่าสองเท่าของความถี่ f_0 หรือ

$$f_s \geq 2 f_0$$

โดยทั่วไปเราอาจสุ่มด้วยความถี่ $f_{SN} = 2 f_0$ พอดี ค่าความถี่นี้เรียกว่า Nyquist Frequency แต่ในทางปฏิบัติเพื่อหลีกเลี่ยงการไม่เป็นเชิงเส้นของการสุ่ม เราจึงมักใช้ความถี่ในการสุ่ม ที่มีค่ามากกว่า Nyquist Frequency ขึ้นไป และความถี่ในการสุ่มที่ใช้นั้นขึ้นอยู่กับลักษณะงาน ไม่กำหนดขอบเขตที่แน่นอน ดังนั้นการสุ่มสัญญาณที่จะได้ผลถูกต้องนั้น เราจะต้องรู้แถบความถี่ปฏิบัติงานของสัญญาณ ดังนั้นในการประมวลผลสัญญาณ เราอาจจะใช้ Low pass Filter เป็นตัวกำหนดแถบความถี่ในการปฏิบัติงานของสัญญาณ ก่อนที่สัญญาณจะเข้าสู่วงจร S/H

Linear Time Invariant System

โดยปกติแล้วในการประมวลผลสัญญาณ ระบบนั้นหมายถึง ขั้นตอนวิธี หรือกระบวนการที่ใช้ในการแปลงคุณสมบัติของลำดับสัญญาณอันหนึ่งไปเป็นอีกอันหนึ่ง เราสามารถเขียนแผนภาพได้ดังนี้



รูปที่ 2.3 ตัวต้นแบบเชิงคณิตศาสตร์ของระบบ

สัญญาณออก และทั้งสองลำดับมีความสัมพันธ์กันตามสมการ

$$y(n) = T\{x(n)\}$$

ซึ่ง $T(\cdot)$ เป็น ตัวดำเนินการ (Operator) ที่ใช้แทนขั้นตอนวิธีแบบต่างๆ ซึ่งอาจอยู่ในรูปแบบของโปรแกรม วงจร หรือผลสมกันระหว่างโปรแกรมกับวงจร

คุณสมบัติของ Time Invariant System

1. ความเป็นเชิงเส้น ระบบ $T\{.\}$ เป็น ระบบเชิงเส้น (Linear System) ก็ต่อเมื่อ ถ้า มีลำดับเข้า $x_1(n)$ และ $x_2(n)$ และถ้าให้

$$\begin{aligned} x_1(n) &\longrightarrow y_1(n) \text{ หรือ } T\{x_1(n)\} = y_1(n) \\ \text{และ } x_2(n) &\longrightarrow y_2(n) \text{ หรือ } T\{x_2(n)\} = y_2(n) \end{aligned}$$

ดังนั้นถ้ามีสัญญาณเข้าเป็น

$$\begin{aligned} a x_1(n) + b x_2(n) &\longrightarrow a y_1(n) + b y_2(n) \\ \text{หรือ } T\{a x_1(n) + b x_2(n)\} &= a y_1(n) + b y_2(n) \end{aligned}$$

โดยที่ a และ b เป็นค่าคงที่ใดๆ

หมายเหตุ คุณสมบัติที่กล่าวมาในข้างต้น ความจริงแล้วก็คือ คุณสมบัติตาม หลักการซ้อนทับ (Superposition Principle) ซึ่งกล่าวไว้ว่า ถ้าหากมี ระบบ หรือ วงจรโครงข่าย ที่ถูกกระตุ้น สัญญาณ Input เข้าหลายๆ สัญญาณ โดยที่แต่ละสัญญาณกระตุ้นมีการคูณค่าคงที่ใดๆ อยู่ด้วยการตอบสนองของระบบสามารถหาได้โดยการนำเอา ผลตอบสนองของระบบที่เกิดจากการแยกกระตุ้นทีละ Input ที่คูณด้วยค่าคงที่มารวมกัน หลักการซ้อนทับ มีอยู่ 2 บทแทรก ที่น่าจะกล่าวถึงคือ

- คุณสมบัติการบวก (Additivity) ถ้า a_1 และ a_2 เป็นค่าใดๆ และให้ $a_1 = a_2 = 1$ ดังนั้น จาก

$$T\{x_1(n) + x_2(n)\} = T\{x_1(n)\} + T\{x_2(n)\} = y_1(n) + y_2(n)$$

- คุณสมบัติการเป็นเอกพันธ์ (Homogeneity) กล่าวไว้ว่า ถ้า $a_1 \neq 0$ และ $a_2 = 0$ ผลตอบสนองของระบบจะเป็น

$$T\{a_1 x_1(n)\} = a_1 T\{x_1(n)\} = a_1 y_1(n)$$

2. ความเป็น Time Invariant ระบบ $T\{.\}$ จะเป็น Time Invariant ก็ต่อเมื่อ

$$\begin{aligned} x(n) &\longrightarrow y(n) \text{ หรือ } T\{x(n)\} = y(n) \\ x(n - n_0) &\longrightarrow y(n - n_0) \text{ หรือ } T\{x(n - n_0)\} = y(n - n_0) \end{aligned}$$

สำหรับทุกๆ ค่า ของ k

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าให้ $h(n)$ เป็น Impulse Response ของระบบหรือ

$$h(n) = T\{\delta(n)\}$$

จะเขียนสมการใหม่ได้เป็น

$$y(n) = \sum_{m=-\infty}^{\infty} x(m).h(n-m)$$

หรือ

$$y(n) = \sum_{m=-\infty}^{\infty} h(m).x(n-m)$$

สมการนี้เรียกว่า Convolution Sum สมการนี้จะมีประโยชน์มากในการประยุกต์ใช้งานในระบบ DSP นี้มาก

จะเห็นได้ว่าเนื่องจากสัญญาณ Input $x(n)$ เป็นลำดับที่ถูก $h(n)$ ทำการแปลงระบบเพื่อให้เป็น $y(n)$ ดังนั้นอาจกล่าวได้ว่า $h(n)$ หรือ Impulse Response เป็นตัวกำหนดคุณสมบัติของ Time Invariant System หรืออาจกล่าวได้ว่า การออกแบบระบบนั้น จะทำได้โดยการกำหนดคุณสมบัติของ $h(n)$ สมการในข้างต้นนี้จะเขียนได้อีกแบบหนึ่งคือ

$$y(n) = x(n) * h(n)$$

โดยที่สัญลักษณ์ * แทนการ Convolution

ตัวอย่าง แสดงการคำนวณ Linear Convolution

ถ้าให้ลำดับ $h(n) = [4, 3, 2, 1, 0]$ และลำดับ $x(n) = [4, 4, 4, 2, 0]$ ให้คำนวณหาผลของ Linear Convolution ของสองลำดับต่อไปนี้

วิธีทำ เนื่องจากข้อมูลมีอยู่ 5 ค่า ตัวแปร n จึงแปลงค่า จาก $n = 0, \dots, 4$ และ

$$y(n) = \sum_{m=-4}^4 x(m).h(n-m)$$

กระจายหาค่า $y(n) = [y(0), y(1), y(2), y(3), y(4), \dots, y(8)]$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(0) = x(0).h(0)$$

$$y(1) = x(0).h(1) + x(1).h(0)$$

$$y(2) = x(0).h(2) + x(1).h(1) + x(2).h(0)$$

$$y(3) = x(0).h(3) + x(1).h(2) + x(2).h(1) + x(3).h(0)$$

$$y(4) = x(0).h(4) + x(1).h(3) + x(2).h(2) + x(3).h(1) + x(4).h(0)$$

$$y(7) = x(3).h(4) + x(4).h(3)$$

$$y(8) = x(4).h(4)$$

ซึ่งเมื่อแทนค่าของลำดับ $x(n)$ และ $h(n)$ จากโจทย์ ทำให้ได้

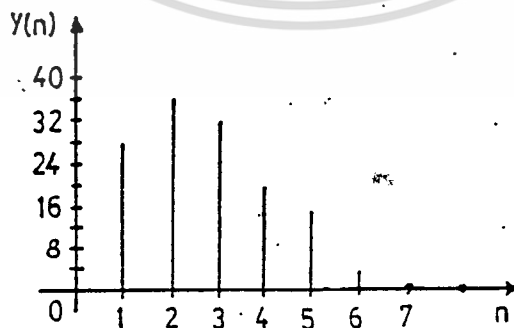
$$y(0) = 16 ; y(1) = 28 ; y(2) = 36 ; y(3) = 32 ; y(4) = 18 ;$$

$$y(5) = 14 ; y(6) = 4 ; y(7) = 0 ; y(8) = 0$$

และผลลัพธ์สามารถนำมา plot graph ได้ดังนี้



(a) ลำดับ $x(n) * h(n)$

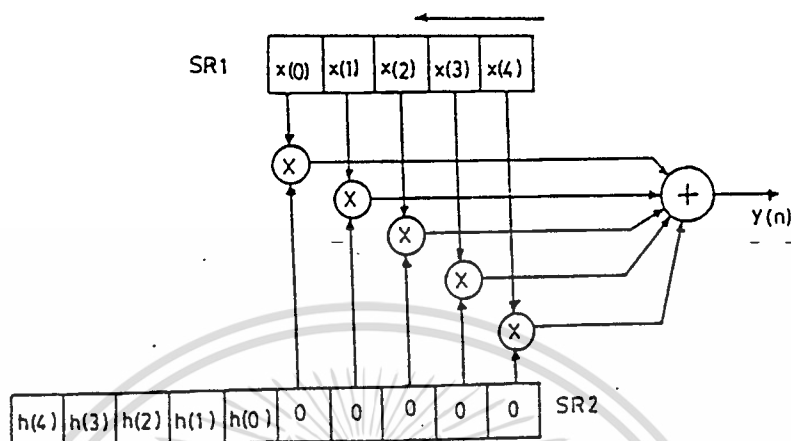


(b) ผลลัพธ์จากการประสานของลำดับ

รูปที่ 2.5 ผลการ Convolution ของลำดับในตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิจารณาจากตัวอย่าง ซึ่งการคำนวณโดยการกระจายสมการ เราอาจเขียนแผนภาพแทนลำดับขั้นตอนการคำนวณได้ดังรูปคือ



รูปที่ 2.6 แผนภาพการ Convolution

Stable and Causal of System

คุณสมบัติอีก 2 ประการที่เป็นเงื่อนไขที่บอกถึงการที่ระบบสามารถนำไปสร้างใช้งานได้คือ เสถียรภาพ (Stability) และ เหตุภาพ (Causality) ของระบบ ซึ่งสามารถแยกพิจารณาได้ว่า

(1) Time Invariant System จะเสถียร ก็ต่อเมื่อ ผลรวมของ Impulse Response ของระบบ จะต้องน้อยกว่าค่า อนันต์ หรือ

$$\sum_{n=-\infty}^{\infty} h(n) < \infty$$

พิสูจน์ เราสามารถพิสูจน์ได้ดังนี้ ถ้าเราจำกัดขนาดของลำดับ $x(n]$ ไม่ให้มีค่าขนาดมากกว่า M หรือให้ $x(n) < M$ สำหรับทุกค่าของ n แล้ว จะได้ว่า

$$y(n) = \sum_{m=-\infty}^{\infty} h(m) x(n-m)$$

และ

$$y(n) \leq M \sum_{m=-\infty}^{\infty} h(m) \leq \infty$$

นั่นคือ ลำดับสัญญาณ Output $y(n]$ จะมีขนาดจำกัดด้วย ถ้าเงื่อนไขในสมการข้างต้นเป็นจริง เอกสารนี้เป็นเอกสารที่สวอนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) Linear Time Invariant System จะเป็นระบบแบบ Causality หรือ สามารถนำไปสร้างใช้งานได้ในระบบเวลาจริง ก็ต่อเมื่อการคำนวณหาสัญญาณ Output $y(n)$ จะขึ้นอยู่กับลำดับ $x(n)$ และ $y(n)$ ที่ย้อนหลังไปเท่านั้น หรือขึ้นกับ $x(n)$ และ $y(n)$ เมื่อ $n \leq n_0$ โดยที่ n_0 เป็นเวลาที่กำลังคำนวณอยู่ นั่นก็คือ Real-Time System เราไม่สามารถรู้ค่าสัญญาณล่วงหน้าได้ด้วยอย่างเช่น Convolution Sum of Real-time และ Causality System อาจเขียนได้ โดยให้ $n_0 = 0$ หรือ

$$y(n) = \sum_{m=0}^N h(n) * x(n-m)$$

คุณสมบัติ 2 ข้อนี้ มีความสำคัญ ต่อระบบประมวลผลสัญญาณในทางปฏิบัติมาก

Fourier Transforms

Fourier Series

ถ้าให้ $x(t)$ เป็นสัญญาณต่อเนื่องใน Time Domain โดยทั่วไปสัญญาณนี้อาจเป็นปริมาณของ ศักย์ไฟฟ้า กระแสไฟฟ้า หรือ ประจุไฟฟ้า ถ้า $x(t)$ มีคุณสมบัติเป็น ฟังก์ชันคาบ ที่มีคาบเป็น T_0 กล่าวคือ $x(t+T_0) = x(t)$ ต่อทุกค่าของเวลา t เราสามารถกระจายหรือเขียนให้อยู่ในรูปของ อนุกรมตรีโกณมิติ คือ

$$x(t) = a_0 + \sum_{n=1}^{\infty} \{ a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t) \}$$

โดยที่

$$a_n = \frac{2}{T} \int_0^{T_0} x(t) \cos(n\omega_0 t) dt$$

$$b_n = \frac{2}{T} \int_0^{T_0} x(t) \sin(n\omega_0 t) dt$$

โดยที่ a_n และ b_n เราจะเรียกว่า Fourier coefficients สมการของ Fourier Series จะเป็นส่วนประกอบในการสร้างสัญญาณของ ฟังก์ชันคาบทุกตัว เมื่อเวลาผ่านไป T วินาที โดยที่ $T=2\pi/\omega_0$

จากสมการ เราสามารถเขียนได้ใหม่โดยที่เป็นรูปแบบของ Phasor เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

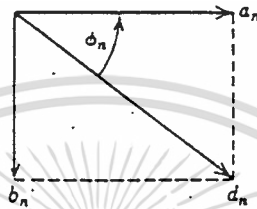
$$x(t) = a_0 + \sum_{n=1}^{\infty} d_n \cos(n\omega_0 t - \phi_n)$$

สัมประสิทธิ์ a_n และ b_n จะถูกแทนที่ว่าเป็น d_n และ ϕ_n โดยที่

$$a_n = d_n \cos \phi_n$$

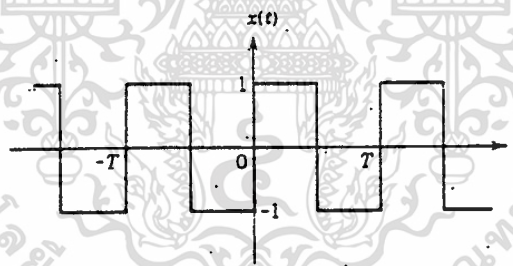
$$b_n = d_n \sin \phi_n$$

เราจะแสดงภาพความสัมพันธ์ของ Phasor และ coefficients ได้ดังรูป



รูปที่ 2.7 แสดงความสัมพันธ์ของ Phasor กับ Coefficients

ตัวอย่าง



Square Wave

จากรูป เราจะหา Fourier Series ของ Square wave ในรูป จากรูป จะได้ $a = 0$ และ

$$a_n = \frac{2}{T} \left[\int_0^{T/2} 1 \cdot \cos(n\omega_0 t) dt + \int_{T/2}^T (-1) \cdot \cos(n\omega_0 t) dt \right] = 0$$

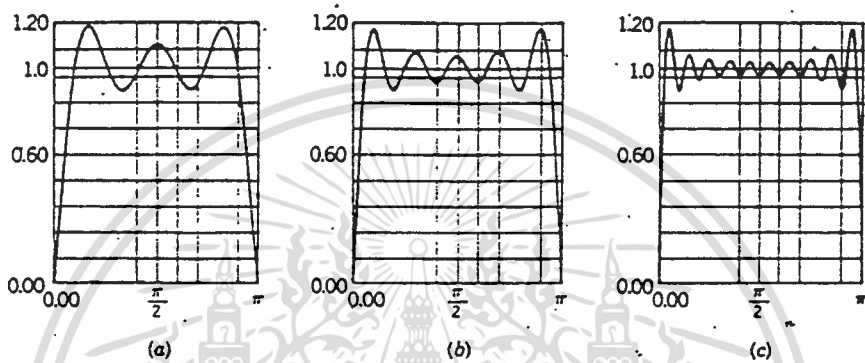
and

$$b_n = \frac{2}{T} \left[\int_0^{T/2} 1 \cdot \sin(n\omega_0 t) dt + \int_{T/2}^T (-1) \cdot \sin(n\omega_0 t) dt \right] = \begin{cases} 0 & \text{for even } n \\ 4/n\pi & \text{for odd } n \end{cases}$$

เราจะได้ Fourier Series ของ Square wave คือ

$$x(t) = \sum_{k=1}^{\infty} \frac{4}{(2k-1)\pi} \sin\{(2k-1)\omega_0 t\}$$

จากสมการ เราจะเห็นว่า Square wave นั้น เกิดจากผลรวมของ Sine wave ในความถี่ต่างๆ หรือ เรียกได้ว่า Square wave นั้น เกิดจากผลรวม Harmonic ของ Sine wave นั้นเอง และสัญญาณที่เป็น พังกซ์ชันคาบทุกตัวด้วย จะแสดงดังรูปที่ 2.8



รูปที่ 2.8 Convergence of the Fourier Series of a square wave. (a) Three terms (b) Five terms (c) Ten terms

EXPONENTIAL FORM OF FOURIER SERIES

เราสามารถเขียน Fourier Series โดยการใช้นำหลักการ Euler เพื่อจะเปลี่ยนให้อยู่ในรูปของ Exponential ได้ จาก

$$\cos(n\omega_0 t) = \frac{\exp(jn\omega_0 t) + \exp(-jn\omega_0 t)}{2}$$

$$\sin(n\omega_0 t) = \frac{\exp(jn\omega_0 t) - \exp(-jn\omega_0 t)}{2j}$$

เราจะนำไปแทนค่าใน Fourier Series ที่เขียนอยู่ในรูปตรีโกณมิติ จะได้

$$x(t) = a_0 + \frac{1}{T} \sum_{n=1}^{\infty} [(a_n - jb_n) \exp(jn\omega_0 t) + (a_n + jb_n) \exp(-jn\omega_0 t)]$$

เราจะสามารถเขียนรูปของสมการได้ใหม่ โดยกำหนดให้

$$c_0 = a_0$$

$$c_n = \frac{(a_n - jb_n)}{2}$$

$$c_{-n} = \frac{(a_n + jb_n)}{2}$$

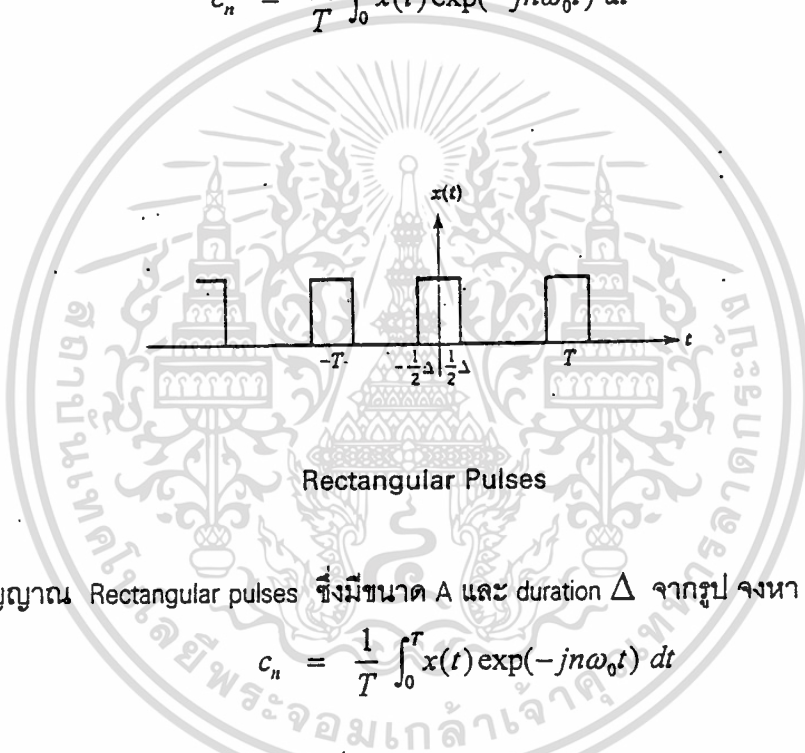
จะได้

$$x(t) = \sum_{-\infty}^{\infty} c_n \exp(jn\omega_0 t)$$

เราจะหาค่า c_n ได้จาก

$$c_n = \frac{1}{T} \int_0^T x(t) \exp(-jn\omega_0 t) dt$$

ตัวอย่าง



สัญญาณ Rectangular pulses ซึ่งมีขนาด A และ duration Δ จากรูป จงหา Fourier Series

จาก

$$c_n = \frac{1}{T} \int_0^T x(t) \exp(-jn\omega_0 t) dt$$

จะได้

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} A \cdot \exp(-jn\omega_0 t) dt$$

$$= \frac{1}{T} \int_{-\Delta/2}^{\Delta/2} A \cdot \exp(-jn\omega_0 t) dt$$

$$= \frac{A}{-jn\omega_0 T} [\exp(-jn\omega_0 \Delta/2) - \exp(jn\omega_0 \Delta/2)]$$

$$= \frac{A\Delta}{2T} \frac{\sin(n\omega_0 \Delta/2)}{n\omega_0 \Delta/2}$$

จากสมการนี้ จะมีกรณีพิเศษ เมื่อ $n = 0$ จะได้

$$c_0 = \frac{A\Delta}{2T}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะนำค่า c_n ไปแทนในสมการ จะได้

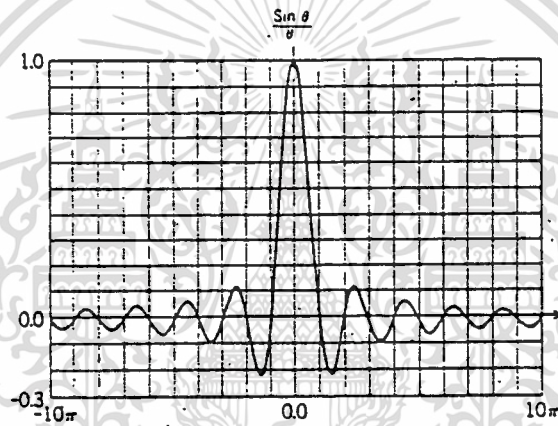
จาก
$$x(t) = \sum_{-\infty}^{\infty} c_n \exp(jn\omega_0 t)$$

จะได้

$$x(t) = \frac{A\Delta}{2T} + \sum_{n=1}^{\infty} \frac{A\Delta}{2T} \frac{\sin(n\omega_0 \Delta/2)}{n\omega_0 \Delta/2} \cos(n\omega_0 t)$$

เราจะกำหนดให้

$$\text{sinc} \triangleq \frac{\sin \theta}{\theta}$$



รูปที่ 2.9 plot of the sinc function

แนวความคิดพื้นฐานนี้สำคัญมากในการวิเคราะห์ระบบวงจรอิเล็กทรอนิกส์ เพราะบอกให้เราเห็นว่า สัญญาณใน Time Domain $x(t)$ สามารถแปลงหรือเขียนกระจายให้อยู่ในรูปของสัญญาณใน Frequency Domain โดยสัญญาณใน Frequency Domain จะประกอบขึ้นด้วยสัญญาณที่มีความถี่เป็นค่า Multiples ของ Fundamental Frequency หรือ $f_0 = 1/T_0$ จำนวนอนันต์ความถี่มารวมกัน เพื่อความสะดวกเราจะเขียน Fundamental Frequency ในรูปของความถี่เชิงมุม $\omega_0 = 2\pi f_0$ ดังนั้น

$$x(t) = \sum_{n=-\infty}^{\infty} C_n(\omega) \exp(jn\omega_0 t)$$

และ

$$C_n(\omega) = (1/T_0) \int_0^{T_0} x(t) \exp(-jn\omega_0 t) dt$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

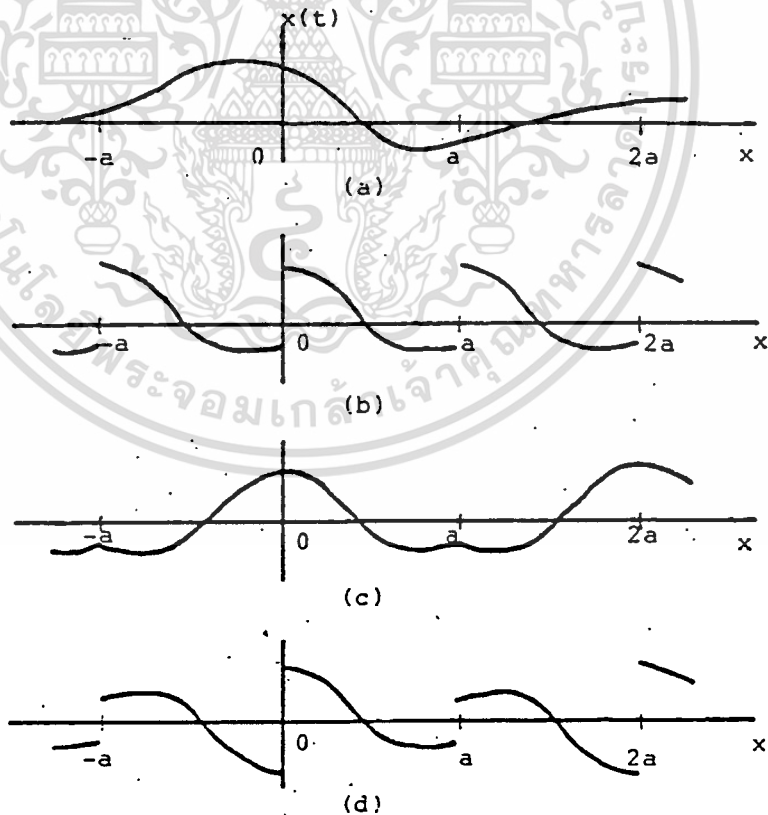
โดยที่ $C_n(\omega)$ เป็นจำนวนเชิงซ้อน และมีชื่อเรียกว่า เป็นส่วนประกอบสเปกตรัมของสัญญาณ $x(t)$ ส่วนความถี่ $n\omega_0$ สำหรับกรณี $n \geq 2$ เรียกว่าเป็น Harmonic Frequency ของสัญญาณ

EVEN AND ODD FUNCTIONS

$x(t)$ จะเป็น Even function ได้ เมื่อ $x(t) = x(-t)$ และจะเป็น Odd function ได้ เมื่อ $x(t) = -x(-t)$ ด้วยเหตุนี้ ฟังก์ชันที่นิยามบนแกนจริง สามารถเขียนได้ในรูปผลบวก ของ Even and Odd functions คือ

$$x(t) = \frac{1}{2}\{x(t)+x(-t)\} + \frac{1}{2}\{x(t)-x(-t)\}$$

เรานำเอา Fourier Series มาเขียนแทน $x(t)$ ที่มีลักษณะไม่เป็นคาบดังตัวอย่างในรูป (a) ได้โดย เรากำหนดช่วงของ t ขึ้นมา โดยที่ $0 < t < T_0$ แล้วใช้สมการของ Fourier Series เขียนแทนในช่วงนี้ เราจะได้สัญญาณใหม่ที่เป็นรูปแบบของฟังก์ชันคาบ ดังรูป (b) ซึ่งเราเขียนแทนด้วย Fourier Series ในช่วงของ $0 < t < a$ ผลที่ได้คือ เราจะได้ฟังก์ชันคาบ ที่มี $T_0 = a$ ดังรูป (b)



รูปที่ 2.10 (a) ตัวอย่างฟังก์ชัน $x(t)$ ที่มี ลักษณะไม่เป็นคาบ รูป(b) เขียนแทนรูป (a) โดยใช้ค่าคาบในคาบ $T_0 = a$ รูป (c) ใช้ค่า $T_0 = 2a$ และให้เป็นฟังก์ชันคี่ ส่วนรูป (d) เป็นการขยายแบบหนึ่งในสี่คลื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น การนำเอกสารนี้ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าหากเราอาศัยคุณสมบัติของฟังก์ชันคู่ และฟังก์ชันคี่ โดยทำการขยาย $x(t)$ ให้มีคาบเป็น $2T_0$ และเลือกฟังก์ชันในระหว่าง T_0 ถึง $2T_0$ เป็นฟังก์ชันคู่ หรือ ให้ $x(t) = x(-t)$ ดังรูป (c) แล้ว Fourier Series ของฟังก์ชันแบบนี้ จะมีเฉพาะพจน์ cosine เท่านั้น คือ

$$x(t) = (A_0/2) + \sum_{n=1}^{\infty} A_n \cos(n\pi t/T_0)$$

by

$$A_n = (1/T_0) \int_0^{T_0} x(t) \cdot \cos(n\pi t/T_0) dt$$

ในลักษณะเดียวกัน ถ้าเราเลือกขยาย $x(t)$ ออกเป็นฟังก์ชันคี่ $x(t) = -x(-t)$ ในช่วง T ถึง $2T$ ดังรูป (d) เราจะได้ Fourier Series ที่มีเฉพาะพจน์ sine เท่านั้น คือ

$$x(t) = \sum_{n=1}^{\infty} B_n \sin(n\pi t/T_0)$$

by

$$B_n = (1/T_0) \int_0^{T_0} x(t) \cdot \sin(n\pi t/T_0) dt$$

วิธีการขยายทั้งสองวิธีที่กล่าวมานี้ เรียกว่า Half-wave expansion โดยที่อนุกรมใน ฟังก์ชันคู่ จะลู่อเข้าหา $x(t)$ บนช่วงปิด $0 \leq t \leq T_0$ และอนุกรมใน ฟังก์ชันคี่ จะลู่อเข้าหา $x(t)$ บนช่วงเปิด $0 < t < T_0$ ส่วนที่จุด $t=0$ และ $t=T_0$ สมการของฟังก์ชันคี่ จะลู่อเข้าหาจุดกึ่งกลาง ณ จุดไม่ต่อเนื่อง ตามรูป (d) คือค่าศูนย์

Fourier Transforms

ถ้าเราให้นิยาม $x(t)$ มีค่าเท่ากับฟังก์ชันบนช่วง $(-T_0, T_0)$ และ นิยามให้มีคาบ $2T_0$ ที่จุดอื่นๆ จากนั้นเขียนแทนฟังก์ชันนี้ด้วย Fourier Series แล้วทำการพิจารณาให้ $T \rightarrow \infty$ เมื่อ T มีค่าสูงมากนั้นทำให้ เครื่องหมายผลบวกรวมยอด (Summation Sign) กลายเป็นค่า Integral ดังนั้น

$$x(t) = (1/2\pi) \int_{-\infty}^{\infty} X(\omega) \exp(j\omega t) d\omega$$

และ

$$X(\omega) = \int_{-\infty}^{\infty} x(t) \exp(-j\omega t) dt$$

สมการแรก $X(\omega)$ เรียกว่าเป็น Fourier Integral หรือ Fourier Transforms ส่วนสมการถัดมา เป็น Inverse Fourier Transforms เงื่อนไขสำหรับ $x(t)$ ที่จะมี $X(\omega)$ ได้นั้นคือ

$$\int_{-\infty}^{\infty} |x(t)| < \infty$$

และจะต้องเป็นจริง $x(t)$ จะต้องสอดคล้องตามเงื่อนไขด้วย ส่วนการแปลงผกผันจะลู่เข้าหา $x(t)$ ณ จุดทุกจุดที่มีความต่อเนื่อง และลู่เข้าหาจุดกึ่งกลาง ณ จุดที่ไม่ต่อเนื่อง สำหรับคู่ของการแปลงระหว่าง $x(t)$ และ $X(\omega)$ อาจเขียนแทนได้เป็น

$$x(t) \Leftrightarrow X(\omega)$$

หรือ

$$X(\omega) = F\{x(t)\} ; x(t) = F^{-1}\{X(\omega)\}$$

โดยที่ $F\{.\}$ แทนการแปลง Fourier และ $F^{-1}\{.\}$ แทนการแปลง Inverse Fourier

CONVOLUTION

เรานิยามการ CONVOLUTION ไว้ว่า $g(t) = x(t) * h(t)$ ของฟังก์ชัน $x(t)$ และ $h(t)$ โดย

$$g(\lambda) = \int_{-\infty}^{\infty} x(t) \cdot h(\lambda - t) dt$$

และยังสามารถพิสูจน์ได้ว่า

$$F\{x(t) * h(t)\} = X(\omega) \cdot H(\omega)$$

สมการเหล่านี้แสดงว่า ถ้าหากใน Time Domain เป็นการนำสัญญาณ 2 สัญญาณ มาทำการ CONVOLUTION กัน ใน Frequency Domain จะเป็นการคูณกันของ สเปกตรัมของแต่ละสัญญาณ ในทางกลับกัน ถ้าใน Time Domain เป็นการคูณกันของสัญญาณ สเปกตรัมที่ได้ก็จะเป็นการ CONVOLUTION ของสเปกตรัมของแต่ละสัญญาณ หรือ สามารถเขียนได้ว่า

$$x(t) * h(t) \Leftrightarrow X(\omega) \cdot H(\omega)$$

และ

$$x(t) \cdot h(t) \Leftrightarrow X(\omega) * H(\omega)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Discrete Fourier Transform : DFT

ถ้าให้ $x_p(t)$ เป็นฟังก์ชันต่อเนื่องและมีคุณสมบัติเป็นคาบ โดยมีคาบเป็น T_p หรือเขียนได้ว่า

$$x_p(t) = x_p(t + mT_p)$$

โดยที่ m เป็นจำนวนเต็มใดๆ ในเชิงคณิตศาสตร์ฟังก์ชันแบบนี้อาจเขียนแทนได้ด้วย Fourier Series ได้ดังนี้

$$x_p(t) = \sum_{k=-\infty}^{\infty} X_p(k) \exp(j2\pi k f_0 t)$$

โดยที่ $f_0 = 1/T_p$ ซึ่งเป็น Fundamental Frequency และ การเขียนแทนนี้เป็นจริงสำหรับในช่วง $0 < t < T_p$ ส่วนสัมประสิทธิ์ของ Fourier Series $X_p(k)$ จะหาได้จาก

$$X_p(k) = (1/T_p) \int_0^{T_p} x_p(t) \cdot \exp(-j2\pi k f_0 t) dt$$

ถ้าหากเราทำการสุ่ม สัญญาณแบบต่อเนื่องและเป็นคาบ $X_p(t)$ การที่จะทำการสุ่มได้นั้น สัญญาณจะต้องมี แแถบความถี่ปฏิบัติงานที่แน่นอน เราอาจทำการกำหนดแถบความถี่ปฏิบัติงานได้ โดยการป้อนผ่าน Low pass Filter ถ้าให้ แถบความถี่ปฏิบัติงาน เป็น f_1 หรือ ให้

$$X_p(k) = 0 ; |k| > k_1$$

โดยที่ k_1 เป็นจำนวนเต็มใดๆ ที่ทำให้ความสัมพันธ์ของความถี่เป็น

$$k_1 \cdot f_0 = f_1$$

หรือกล่าวอีกนัยหนึ่งก็คือ เรากำหนดให้สเปกตรัมของความถี่ที่เกินค่า f_1 มีค่า Amplitude เป็น 0 ดังรูป (b)

ค่าความถี่ของสัญญาณสุ่มตัวอย่าง อย่างน้อยที่สุด จะต้องเป็น 2 เท่าของ แถบความถี่ปฏิบัติงาน หรือ ในกรณีนี้ต้องเท่ากับ $2f_1$ ดังนั้น ช่วงเวลาสุ่มตัวอย่าง T ที่ใช้ในการสุ่มตัวอย่าง สัญญาณนี้ มีค่าเท่ากับ

$$T = 1/2f_1 = 1/2k_1f_0 = T_p/2k_1$$

และถ้าให้ในหนึ่งคาบ ของสัญญาณที่สุ่มออกมา เป็น N ช่วง ด้วยกัน ทำให้ได้ว่า $2k_1 = N$

ดังนั้นสัญญาณ $x_p(t)$ ที่อยู่ในช่วง $0 < t < T_p$ เมื่อถูกสุ่มออกเป็น N ช่วงนี้ สามารถเขียนสมการแทนได้เป็น

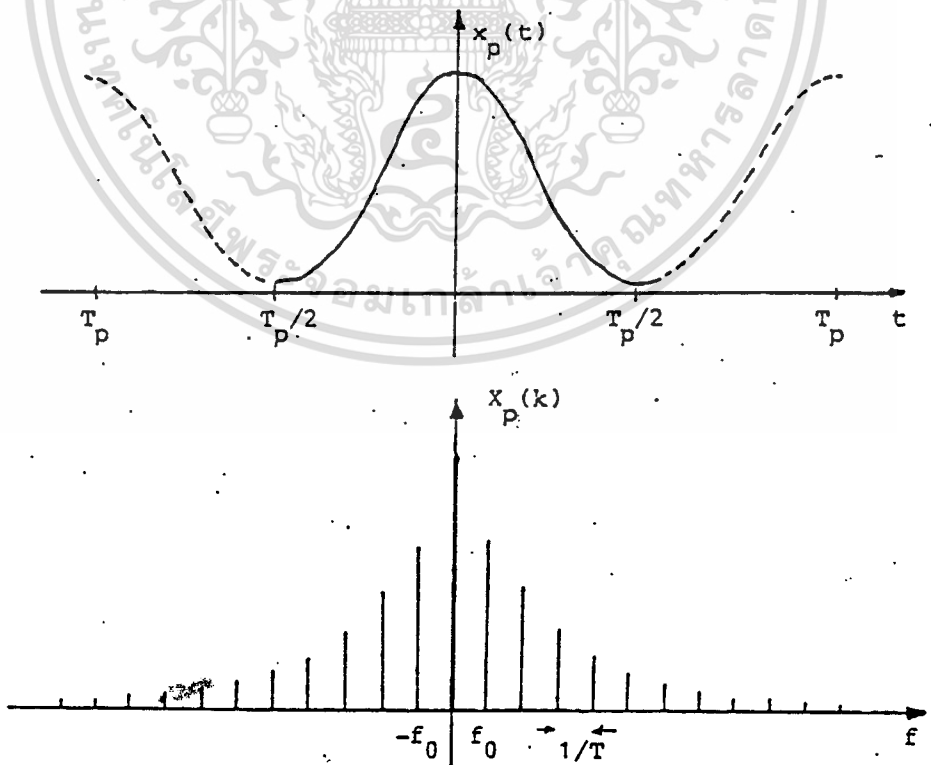
$$x(t/T) = \sum_{n=0}^{N-1} x_p(t/T) \cdot \delta((t/T) - n)$$

โดยที่ t/T เป็น Normalized Time ซึ่งนิยามในช่วง $0 \leq t < T_p$ หรือ $0 \leq t/T < N$ เนื่องจากสัญญาณ $x(t/T)$ เนื่องจากสัญญาณ $\overline{x(t/T)}$ เป็นสัญญาณแบบฟังก์ชันคาบเช่นกัน ดังนั้นจึงอาจหาสัมประสิทธิ์ของสัญญาณได้คือ

$$X(k) = (1/N) \int_{n=0}^{N-1} x(t/T) \cdot \exp\{-j2\pi k/N \cdot (t/T)\} d(t/T)$$

และเมื่อแทนค่า $x(t/T)$ ลงในสมการ จะได้

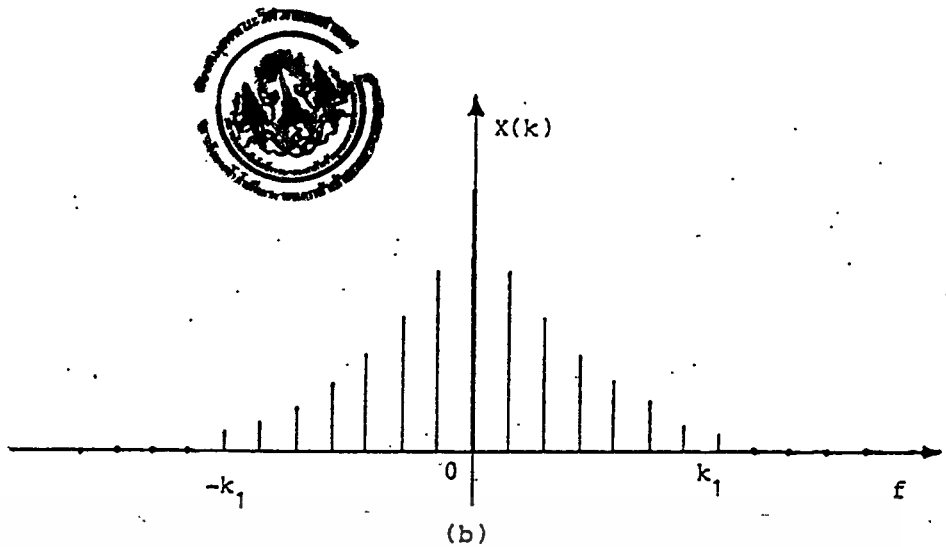
$$X(k) = (1/N) \int_0^{N-1} \left\{ \sum_{n=0}^{N-1} x_p(t/T) \cdot \delta((t/T) - n) \cdot \exp\{-j2\pi k/N \cdot (t/T)\} \right\} d(t/T)$$



สเปกตรัมของสัญญาณเป็นคาบ $x_p(t)$

(a).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 แสดงสเปกตรัมของสัญญาณที่มีแถบความถี่ปฏิบัติงานจำกัด (a) สัญญาณเป็นคาบ (b) เป็นสเปกตรัมของสัญญาณ (c) สเปกตรัมเมื่อจำกัดความถี่ปฏิบัติงาน

และจากคุณสมบัติของ $\delta(t)$ ที่นิยามว่า

$$\int_{-\infty}^{\infty} f(t/T) \cdot \delta((t/T) - n) d(t/T) = f(n)$$

เพราะฉะนั้นทำให้ได้

$$X'(k) = (1/N) \sum_{n=0}^{N-1} x_p(n) \cdot \exp(-j2\pi kn/N)$$

และโดยความจริงแล้ว ที่จุดที่ทำการสุ่มสัญญาณ สัญญาณที่สุ่มมา $x(n)$ มีค่าเท่ากับ $X_p(n)$ ดังนั้น

$$X'(k) = (1/N) \sum_{n=0}^{N-1} x(n) \cdot \exp(-j2\pi kn/N)$$

แต่โดยนิยามทั่วไปให้

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot \exp(-j2\pi kn/N)$$

สมการนี้เรียกว่า Discrete-Time Fourier Transform (DFT) ของลำดับสัญญาณ $x(n)$ และสำหรับ Inverse Discrete Fourier Transform (IDFT) จะได้ว่า

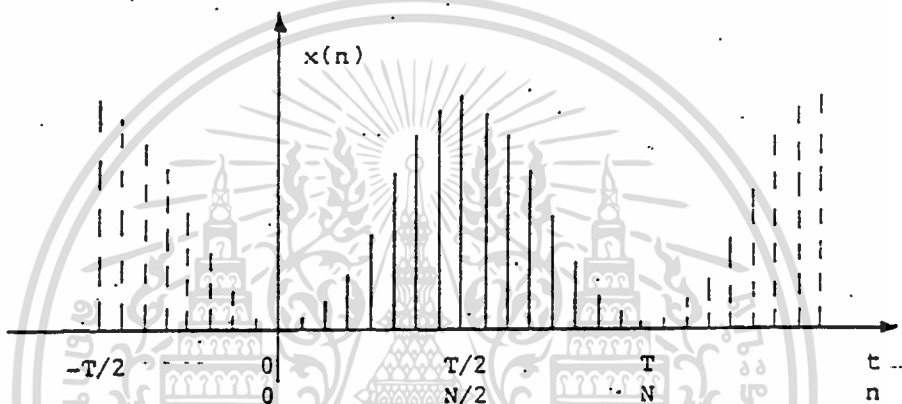
$$x(n) = (1/N) \sum_{k=0}^{N-1} X(k) \cdot \exp(j2\pi kn/N)$$

โดยที่ลำดับสัญญาณและลำดับของสเปกตรัมต้องมีคุณสมบัติความเป็นคาบด้วย หรือ

$$x(n) = x(n+mN) \text{ เมื่อ } N \text{ เป็นจำนวนเต็มใดๆ}$$

$$X(k) = X(k+mN) \text{ เมื่อ } N \text{ เป็นจำนวนเต็มใดๆ}$$

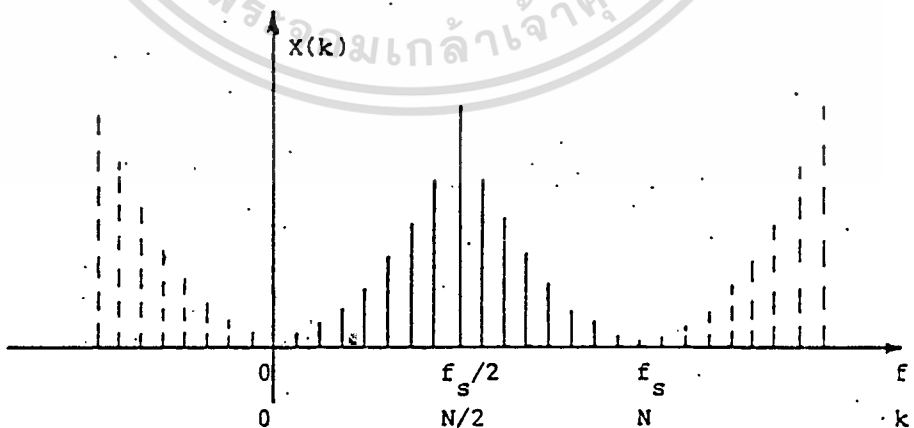
คุณสมบัติความเป็นคาบของลำดับ $x(n)$ นั้น มาจากสมมติฐานตอนแรกที่เราทำการสุ่มสัญญาณที่เป็นคาบ ส่วนคุณสมบัติความเป็นคาบ $X(k)$ เป็นผลมาจากว่า สัญญาณที่ถูกสุ่มตัวอย่างมา สเปกตรัมของมันมีความเป็นคาบด้วย



$$X(k) = (1/N) \sum_{n=0}^{N-1} x(n) \cdot \exp(-j2\pi nk/N)$$

รูปที่ 2.12 (a) แสดงสัญญาณที่ทำการสุ่มจากสัญญาณต่อเนื่อง

(a)



$$x(n) = \sum_{k=0}^{N-1} X(k) \cdot \exp(j2\pi k n/N)$$

รูปที่ 2-12 (b) สเปกตรัมของรูป (a)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของการแปลง Fourier โดยสรุป

จะมีการแปลง Fourier อยู่ 4 รูปแบบด้วยกัน แต่ละรูปแบบมีประโยชน์ในการประยุกต์ใช้งานที่ต่างกันอย่างออกไป และโดยมากในการใช้งาน มักจะเกี่ยวข้องกับ ตัวแปรเวลา t และตัวแปรความถี่ f ดังนั้นจะแสดงความสัมพันธ์ใน Domain ทั้ง 2 นี้

การแปลงอินทิกรัล

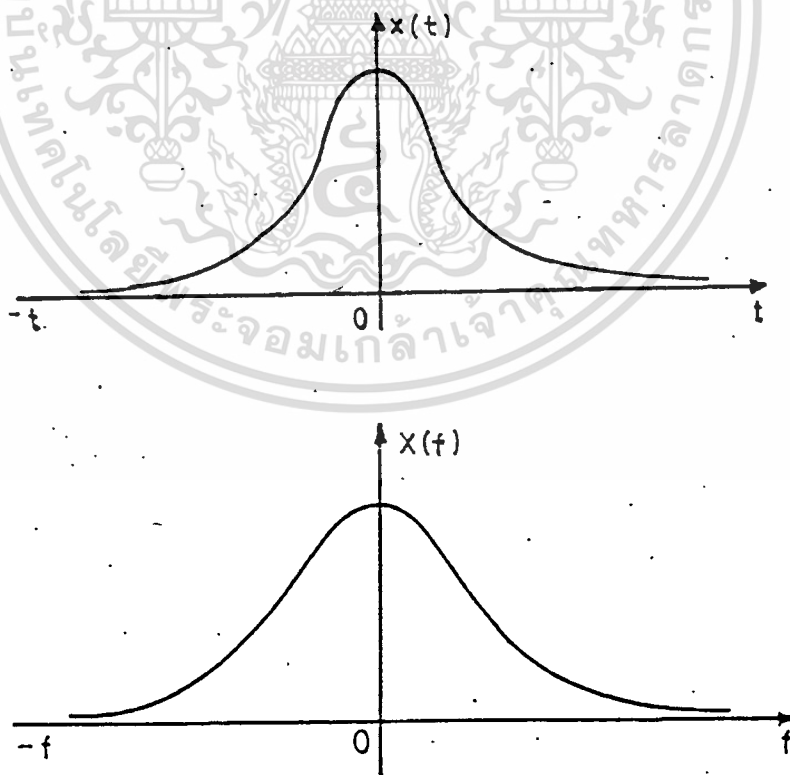
การแปลงรูปแบบนี้ ใช้กับสัญญาณใน Time Domain เป็นสัญญาณที่ต่อเนื่อง และมีคาบเป็นอนันต์ $-\infty < t < \infty$ ส่วนใน Frequency Domain ก็มีลักษณะเป็นฟังก์ชันต่อเนื่อง และมีคาบเป็นอนันต์ด้วย การแปลง Fourier $X(f)$ ของสัญญาณ $x(t)$ เป็น

$$X(f) = \int_{-\infty}^{\infty} x(t) \exp(-j2\pi ft) dt$$

ส่วนการแปลงผกผัน เป็น

$$x(t) = \int_{-\infty}^{\infty} X(f) \exp(j2\pi ft) df$$

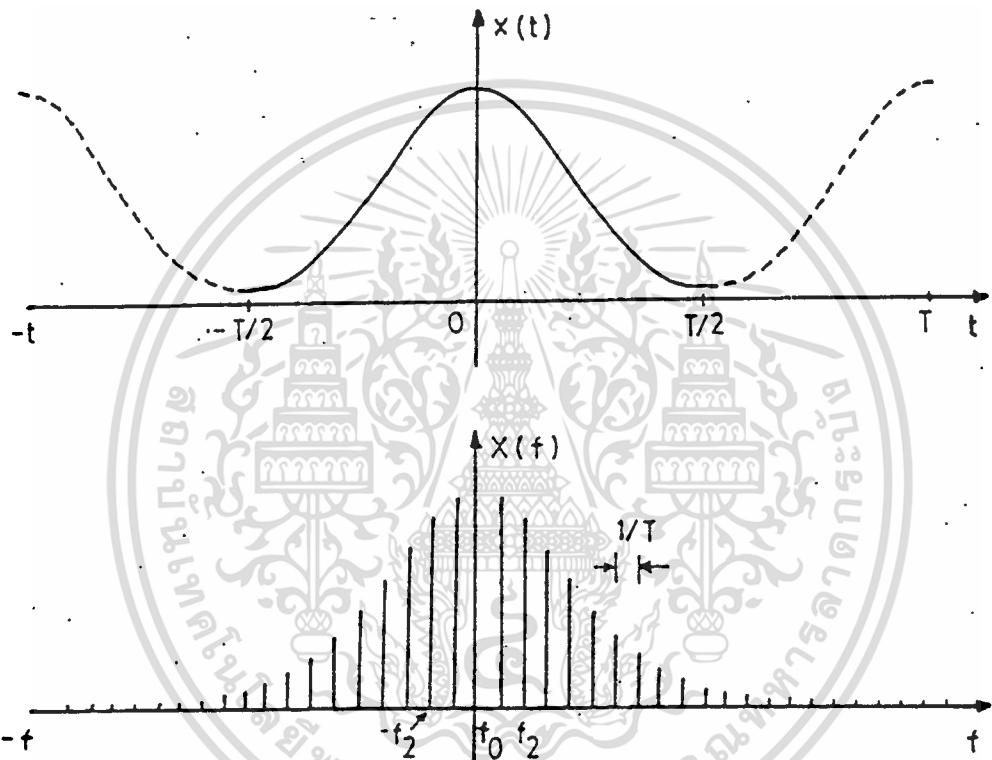
รูปแบบของสัญญาณฟังก์ชันของเวลา และผลการแปลงของมันได้แสดงไว้ดังรูปที่ 2-13



รูป 2-13 แสดงผลการแปลงฟูริเยอร์อินทิกรัล ของสัญญาณที่ต่อเนื่องและมีคาบเป็นอนันต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล่าวโดยสรุปก็คือ เป็นการแปลงสำหรับสัญญาณแบบไม่เป็นคาบที่เป็นฟังก์ชันไม่ต่อเนื่องกับเวลา โดยที่สเปกตรัมของสัญญาณมีลักษณะไม่เป็นคาบและเป็นฟังก์ชันต่อเนื่องกับความถี่ การแปลงนี้เป็นการแปลงอุดมคติ สำหรับสัญญาณที่เป็นฟังก์ชันของเวลาได้ทุกรูปแบบ อย่างไรก็ตามการหาค่าของการแปลง หรือการแปลงผกผันนี้ ต้องทำการอินทิเกรตค่าของฟังก์ชันตลอดช่วงของเวลา และความถี่ เป็นลบนันต์ ถึง บวกอนันต์ ดังนั้นการแปลงนี้จึงเหมาะที่จะใช้กับสัญญาณที่มีการนิยามค่าในช่วงแคบๆ เท่านั้น เพราะการอินทิเกรตทำได้โดยง่าย



รูป 2-14 แสดงสเปกตรัมจากการประยุกต์ใช้อินTEGRUMฟูริเยอร์กับสัญญาณต่อเนื่องและเป็นคาบ

Fourier Series

รูปแบบนี้มีการใช้ทั่วไป โดยที่ใช้ในการแปลงสำหรับสัญญาณประเภท เป็นคาบและต่อเนื่อง ใน Time Domain ส่วนสเปกตรัมใน Frequency Domain มีลักษณะไม่เป็นคาบ และเป็นฟังก์ชันเต็มหน่วย ถ้าเราให้ $x(t)$ แทนสัญญาณเป็นคาบและต่อเนื่องกับเวลาและมีค่าคาบเป็น T_0 และให้มีผล การแปลง Fourier เป็นฟังก์ชันเต็มหน่วยกับความถี่ หรือ $x(f_k)$ ดังนั้นเราสามารถเขียนการแปลงได้ เป็น

$$X(f_k) = \int_{-T_0/2}^{T_0/2} x(t) \exp(-j2\pi f_k t) dt$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแปลงผกผันเป็น

$$x(t) = \left(1/T_p\right) \sum_{m=-\infty}^{\infty} X(f_k) \exp(j2\pi f_k t) dt$$

ฟังก์ชันเชิงสุ่มตัวอย่าง

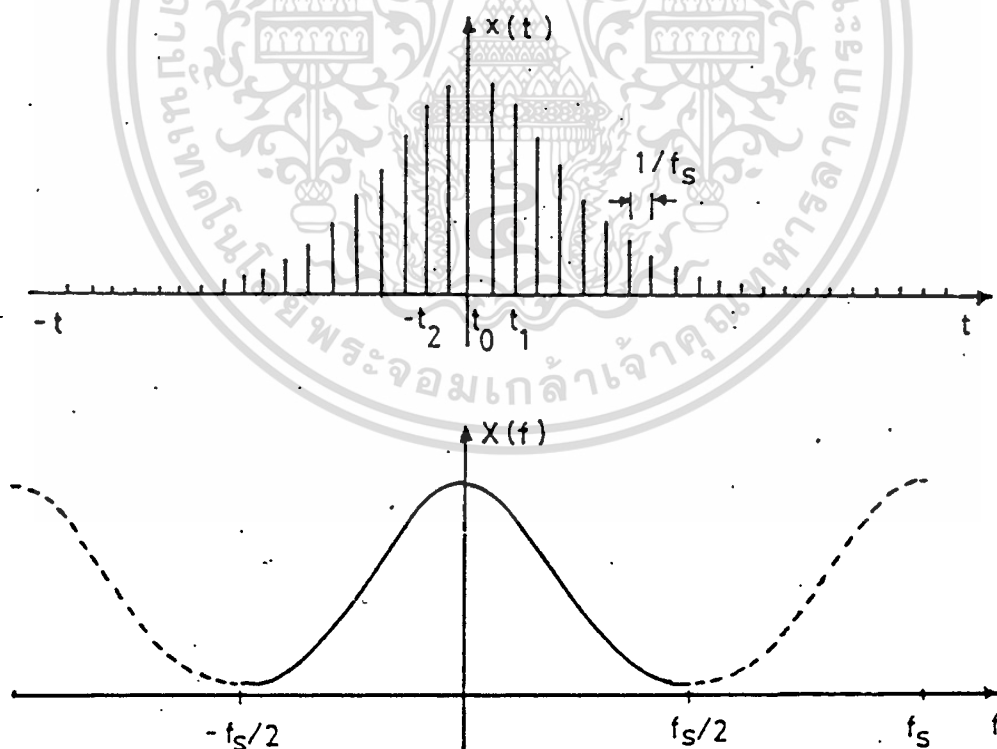
การแปลงแบบนี้ใช้กับสัญญาณใน Time Domain ที่เป็นแบบไม่เป็นคาบ และเป็นฟังก์ชันเต็มหน่วย หรือเขียนแทนด้วย $x(t_n)$ ส่วนผลการแปลงของมันจะเป็น ฟังก์ชันแบบต่อเนื่องและเป็นคาบ หรือเขียนแทนด้วย $X(f)$ โดยที่

$$X(f) = \sum_{n=-\infty}^{\infty} x(nT) \exp(-j2\pi f t_n)$$

and

$$x(t_n) = (1/f_s) \int_{-f_s/2}^{f_s/2} X(f) \exp(j2\pi f t_n) dt$$

โดยที่การอินทิเกรตเป็นการหาค่าในหนึ่งช่วงคาบของ $X(f)$ และ รูปลักษณะของฟังก์ชันทั้งสองได้แสดงไว้ดังรูปที่ 2-15



รูปที่ 2-15 สัญญาณเชิงสุ่มตัวอย่าง และสเปกตรัมของมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของผลการแปลงแบบนี้อาจกล่าวเทียบเคียงได้ว่า คือการทำกรหาผลการแปลงซัด และการแปลงซัดผกผัน บนเส้นรอบวงของวงกลมหนึ่งหน่วยนั่นเอง และสภาวะการเป็นคาบของสเปกตรัมนั้น เกิดมาจากการที่เราทำการสุ่มตัวอย่างสัญญาณใน โดเมนเวลานั้นเอง

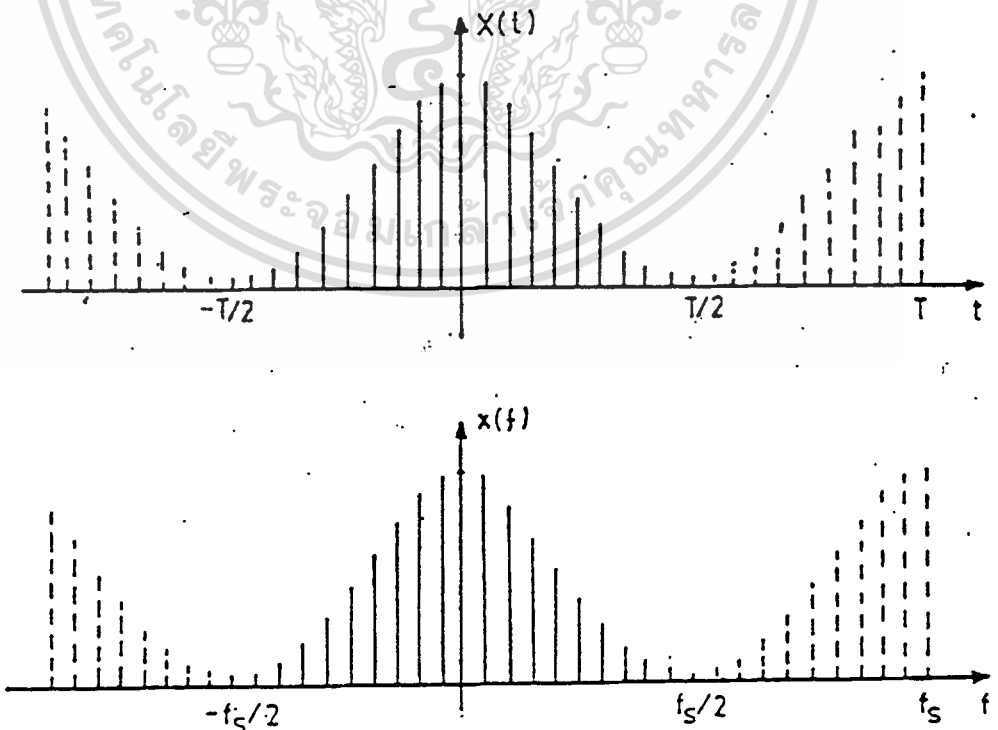
Discrete Time Fourier Transform

การแปลงแบบนี้ใช้กับลักษณะของสัญญาณใน Time Domain เป็นฟังก์ชันคาบ และเป็นฟังก์ชันเต็มหน่วย ส่วนใน Frequency Domain สเปกตรัมจะมีลักษณะเป็นคาบและเป็นฟังก์ชันเต็มหน่วยของความถี่ ถ้าให้ $x(t_n)$ แทนสัญญาณใน Time Domain และ $X(f_k)$ แทนสเปกตรัมของความถี่ เราสามารถเขียนความสัมพันธ์ของสองสัญญาณนี้ได้เป็น

$$X(f_k) = \int_{n=0}^{N-1} x(t_n) \exp(-j2\pi t_n f_k / N)$$

$$x(t_n) = (1/N) \int_{m=0}^{N-1} X(f_k) \exp(j2\pi t_n f_k / N)$$

โดยที่เครื่องหมายรวมยอดในสมการแรก เป็นการหาค่าในหนึ่งคาบของสัญญาณ $x(t_n)$ และเครื่องหมายการรวมยอดในสมการถัดมา เป็นการคำนวณหาค่าในหนึ่งคาบของ $X(f_k)$ และค่าความถี่ในการสุ่มตัวอย่างสัญญาณ $f_s = 1/T$ ในรูปได้แสดงภาพเพื่อเป็นการเปรียบเทียบให้เห็นถึงผลการแปลง



รูปที่ 2-16 สัญญาณเชิงเต็มหน่วย และสเปกตรัมของมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังที่กล่าวมาได้สรุปถึงคุณลักษณะของการแปลงทั้ง 4 แบบ มีข้อนำสังเกตอีก 3 ประการ คือ

1. ถ้าตัวแปรในโดเมนหนึ่ง มีคุณสมบัติภาวะเป็นคาบ ตัวแปรในโดเมนของผลการแปลงจะมีคุณสมบัติเป็น ฟังก์ชันเต็มหน่วย หรือ ฟังก์ชันเชิงเส้นตัวอย่าง
2. ในทางตรงกันข้าม ถ้าในโดเมนหนึ่งตัวแปรเป็นฟังก์ชันแบบเชิงเส้นตัวอย่าง ผลการแปลงก็จะมีคุณสมบัติของภาวะเป็นคาบ
3. ค่าของคาบในโดเมนหนึ่ง จะเท่ากับส่วนกลับของค่าระยะห่างของลำดับสัญญาณในอีกโดเมนหนึ่ง

Z - TRANSFORM

DEFINITION OF THE Z - TRANSFORM

Model ของสัญญาณสุ่มตัวอย่างดังที่กล่าวไว้ในข้างต้น คือ

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT) \delta(t-nT)$$

ถ้า $\delta(t-nT)$ มีค่าเป็น ศูนย์ ยกเว้นที่ $t = nT$ จะมีค่าเป็น 1 $x(t)$ สามารถแทนได้โดย $x(nT)$ ถ้า $x(t)$ ต่อเนื่องที่ $t = nT$ ถ้าเราให้ $x(t) \equiv 0 ; t < 0$ เราจะเขียนสมการได้ใหม่ เป็น

$$x(t) = \sum_{n=0}^{\infty} x(nT) \delta(t-nT)$$

เราจะทำการแปลง Laplace Transform จะได้สมการเป็น

$$\begin{aligned} X(s) &= \int_0^{\infty} \sum_{n=0}^{\infty} x(nT) \delta(t-nT) e^{-st} dt \\ &= \sum_{n=0}^{\infty} x(nT) \int_0^{\infty} \delta(t-nT) e^{-st} dt \end{aligned}$$

จากคุณสมบัติของ Delta Function

$$\int_{t_1}^{t_2} x(t) \cdot \delta(t-t_0) dt = \begin{cases} x(t_0) & ; t_1 < t_0 < t_2 \\ 0 & ; \text{otherwise} \end{cases}$$

จะได้สมการเป็น

$$X(s) = \sum_{n=0}^{\infty} x(nT) e^{-snT}$$

กำหนดให้ ตัวแปรเชิงซ้อน z เป็น

$$z = e^{sT}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะได้

$$X(z) = \sum_{n=0}^{\infty} x(nT) z^{-n}$$

ตัวแปร s นั้นก็คือ จำนวนเชิงซ้อนค่าหนึ่ง จะเขียนได้ว่า

$$s = \sigma + j\omega$$

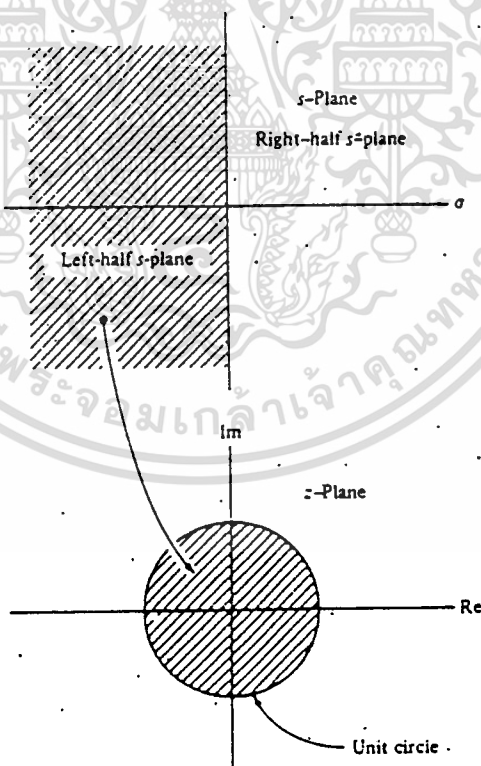
เราจะเขียน z ได้ใหม่เป็น

$$z = e^{\sigma T} e^{j\omega T}$$

เพราะฉะนั้น เราจะเขียน ขนาดของ z ได้โดย

$$|z| = e^{\sigma T}$$

ดังนั้น เราจะได้ว่า ทางขวามือของ s -plane คือ $\sigma > 0$ จะมีค่าเท่ากับ $|z| > 1$ และทางซ้ายมือคือ $\sigma < 0$ จะมีค่าเท่ากับ $|z| < 1$ เราจะเห็นได้ว่า ทางซ้ายมือทั้งหมดของ s -plane คือพื้นที่ในวงกลมหนึ่งหน่วยของ z -plane และทางขวามือของ s -plane คือพื้นที่นอกวงกลมหนึ่งหน่วยของ z -plane เราจะแสดงตัวแปร s บน z -plane ได้ดังรูปที่ 2-17



รูปที่ 2-17 การส่งผ่าน s -domain ไปยัง z -domain

คุณสมบัติ ของ z - Transform

1. Linearity

$$\zeta [a_1 x_1(n) + a_2 x_2(n)] = a_1 X_1(z) + a_2 X_2(z)$$

2. Shift in time

$$\zeta [x(n+1)] = z [X(z) - x(0)]$$

3. Multiplication by n

$$\zeta [n x(n)] = -z \frac{d}{dz} X(z)$$

4. Multiplication by r^n

$$\zeta [r^n x(n)] = X\left(\frac{z}{r}\right)$$

5. Convolution

$$\zeta \left[\sum_{k=0}^{\infty} x_1(k) x_2(n-k) \right] = X_1(z) X_2(z)$$

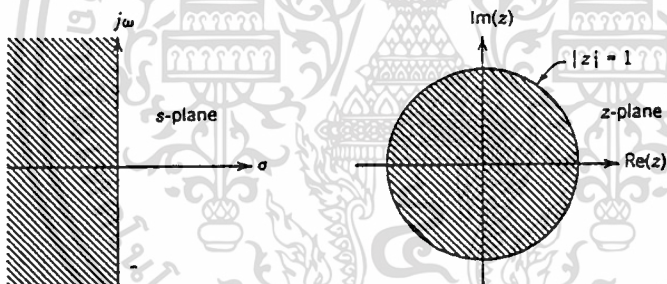
6. Initial value

$$x(0) = \lim_{z \rightarrow \infty} X(z)$$

7. Final value

$$\lim_{n \rightarrow \infty} x(n) = \lim_{z \rightarrow 1} [(z-1)X(z)]$$

Stability of Linear Discrete Time Systems



1. Discrete Time system จะเสถียรได้ก็ต่อเมื่อ Poles ของระบบจะต้องอยู่ภายใน Unit Circle ของ z - plane

2. Discrete Time system จะเสถียรได้ก็ต่อเมื่อ Impulse Response Sequence $h(n)$ จะต้องเป็นไปตามคุณสมบัติ

$$\lim_{n \rightarrow \infty} |h(n)| = 0$$

3. Discrete Time system จะเสถียรได้ก็ต่อเมื่อ Impulse Response Sequence $h(n)$ จะต้องเป็นไปตามคุณสมบัติ

$$\sum_{n=0}^{\infty} |h(n)|^p < \infty \quad ; \text{ for any } p \geq 1$$

4. Discrete Time system จะเสถียรได้ก็ต่อเมื่อ รากของ polynomial จะต้องน้อยกว่า 1

Delay Operator

Delay Operator เป็นพื้นฐานของการวิเคราะห์ในระบบ Digital หรือ ระบบ Discrete Time เช่น ถ้าเรามีลำดับสัญญาณ $x(nT)$ ซึ่งถูก Delay โดย K sample periods เขียนให้อยู่ในรูปของ z -domain ได้รูปผลคูณของ $X(z)$ กับ z^{-K} เราสามารถพิสูจน์ได้จาก

$$X(z) = \sum_{n=0}^{\infty} x(nT) z^{-n}$$

ลำดับ $x(nT - KT)$ ในรูปของ z -Transform จะได้เป็น

$$\zeta[x(nT - KT)] = \sum_{n=0}^{\infty} x(nT - KT) z^{-n}$$

ถ้าให้ $m = n - K$ จะได้

$$\zeta[x(nT - KT)] = \sum_{m=-K}^{\infty} x(mT) z^{-m-K}$$

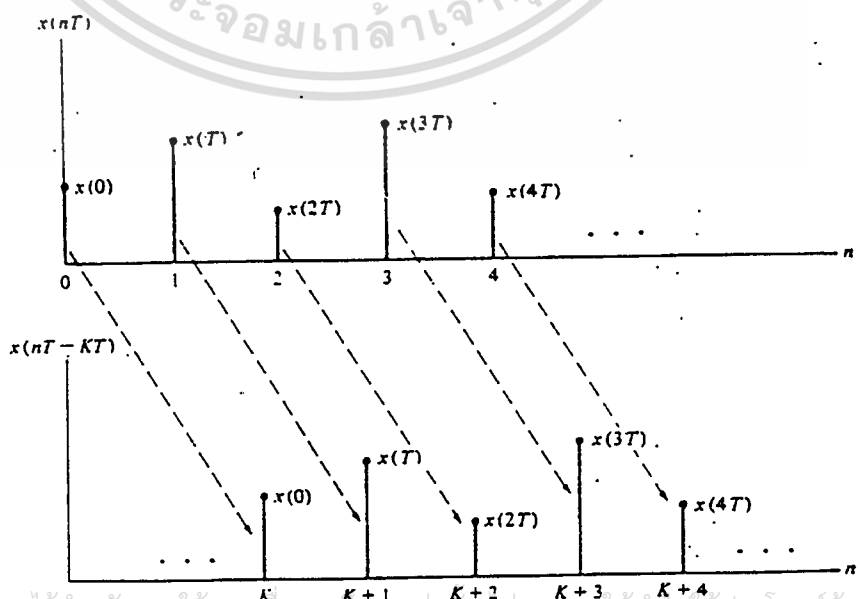
$x(mT)$ จะมีค่าเป็นศูนย์ เมื่อ $m < 0$ เราจะได้

$$\zeta[x(nT - KT)] = \sum_{m=0}^{\infty} x(mT) z^{-m-K}$$

or

$$\zeta[x(nT - KT)] = z^{-K} \sum_{m=0}^{\infty} x(mT) z^{-m} = z^{-K} X(z)$$

เราจะแสดงภาพของ Sequence ที่ถูก delay ไป K sample periods ได้ดังนี้



ยกตัวอย่าง ให้ $Y(z) = X(z) z^{-2}$ เราจะทำการแปลงกลับจะได้เป็น $y(nT) = x[(n-2)T]$
 และ ถ้าให้ $Y(z) = X(z) z^{-2} + X(z) z^{-1} + X(z) z^0 + X(z) z^1 + X(z) z^2$ เราจะทำการ
 แปลงกลับได้เป็น $y(n) = x(n-2) + x(n-1) + x(n) + x(n+1) + x(n+2)$



บทที่ 3

การออกแบบและสร้าง

การออกแบบ Digital Filter โดยใช้ Fourier Series

Non-recursive Filter จะสามารถเขียนสมการได้ตามสมการของ Fourier Series ซึ่งในการคำนวณจะเห็นว่า ความยาวของ Filter นั้นมีจำกัด ถ้าให้ $x(n)$ เป็น Input Signal Sequence และให้ $y(n)$ เป็น Output Signal Sequence เพราะฉะนั้น เราจะเขียนสมการได้ดังนี้

$$\begin{aligned}y(n) &= \sum_{k=-M}^M C_k \cdot x(n-k) \\&= C_0 x(n) + \sum_{k=-1}^{-M} C_k \cdot x(n-k) + \sum_{k=1}^M C_k \cdot x(n-k) \\&= C_0 x(n) + \sum_{k=1}^M \{C_{-k} \cdot x(n+k) + C_k \cdot x(n-k)\}\end{aligned}$$

ทำการแปลง z -Transform เพื่อเขียน Transfer Function จะได้

$$H(z) = C_0 + \sum_{k=1}^M \{C_{-k} z^k + C_k z^{-k}\} = \sum_{k=-M}^M \{C_k z^{-k}\}$$

or

$$H(\omega) = C_0 + \sum_{k=1}^M \{C_{-k} e^{j\omega k} + C_k e^{-j\omega k}\} = \sum_{k=-M}^M \{C_k e^{-j\omega k}\}$$

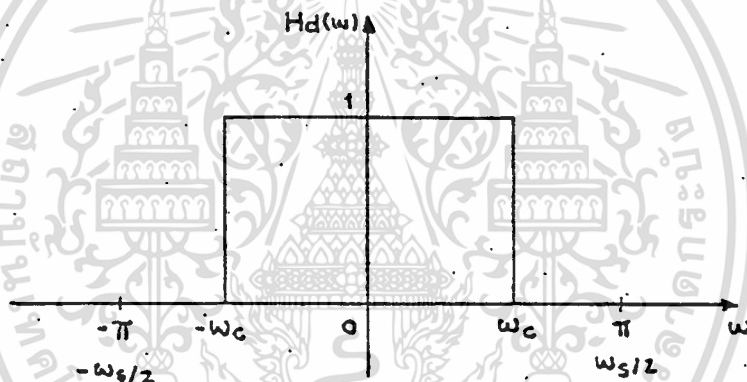
ถ้าพิจารณากรณี Filter แบบสมมาตรคู่ หรือ $C_k = C_{-k}$

$$\begin{aligned}H(\omega) &= C_0 + \sum_{k=1}^M C_k \{e^{j\omega k} + e^{-j\omega k}\} \\&= C_0 + \sum_{k=1}^M 2C_k \left\{ \frac{e^{j\omega k} + e^{-j\omega k}}{2} \right\} \\H(\omega) &= a_0 + \sum_{k=1}^M a_k \cos(\omega k)\end{aligned}$$

โดยที่กำหนดให้ $a_0 = C_0$ and $a_k = 2C_k$ and $k > 0$

ผลที่ได้จากสมการ จะเห็นได้ว่า Non-recursive Filter แบบสมมาตรคู่ นั้น สามารถเขียนอธิบายได้ด้วย Fourier Series ที่ประกอบด้วยสัมประสิทธิ์เฉพาะพจน์ Cosine เท่านั้น และ ค่าสัมประสิทธิ์ของ Fourier Series α_k ก็จะเป็นค่าสัมประสิทธิ์ของ Filter ด้วย ในลักษณะเดียวกัน เราอาจแสดงได้ว่า สำหรับ Non-recursive Filter แบบสมมาตรคี่ สามารถเขียนแทนด้วย Fourier Series ที่มีเฉพาะพจน์ Sine เท่านั้น และ ค่าสัมประสิทธิ์จะเป็นค่า จินตภาพ เสมอ

โดยปกติแล้วการออกแบบตัวกรองสัญญาณนั้น กระทำโดยการทำการประมาณค่า ให้ Filter ที่ออกแบบมีผลตอบสนองความถี่ได้ใกล้เคียงกับ ผลตอบสนองอุดมคติดังรูปที่ 3.1 ให้มากที่สุด



รูปที่ 3.1 ผลตอบสนอง Amplitude อุดมคติ

โดยตามรูป ω_c เป็นค่า Cut off Frequency จากผลตอบสนองความถี่อุดมคติ $H_d(\omega)$ สามารถกระจายให้อยู่ในพจน์ของ Time Domain โดยการประยุกต์ใช้ Fourier Transform ได้

$$H_d(\omega) = \sum_{k=-\infty}^{\infty} h_d(k) e^{-jk\omega T}$$

โดยที่ Impulse Response $h_d(k)$ หาได้จาก

$$h_d(k) = (1/2\pi) \int_{-\omega_s/2}^{\omega_s/2} H_d(\omega) e^{jk\omega T} d\omega$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการข้างต้น Frequency Response ของ Non-recursive Digital Filter มีค่าเป็น

$$H(\omega) = \sum_{k=-M}^M C_k e^{-jk\omega T}$$

เมื่อนำสมการมาเปรียบเทียบ จะเห็นได้ว่า Frequency Response ของ Filter จะใกล้เคียงกับผลตอบ
สนองอุดมคติ ก็ต่อเมื่อ $M = \infty$ แต่ในทางปฏิบัติเป็นไปได้ยาก เราจึงทำการตัดปลาย Fourier
Series ให้เหลือเท่าที่จำเป็น และจัดให้

$$C_k = h_d(k)$$

โดยที่ $k = -M, \dots, 0, \dots, M$

ส่วนความยาวของตัว $N = (2M+1) \rightarrow M = (N-1)/2$

ตัวอย่าง จงออกแบบ Filter โดยใช้ Fourier Series โดยให้ได้ Frequency Response ดังรูป

โดยมีค่า ความถี่ตัด $\omega_c = \pi/2$

จากสมการจะได้

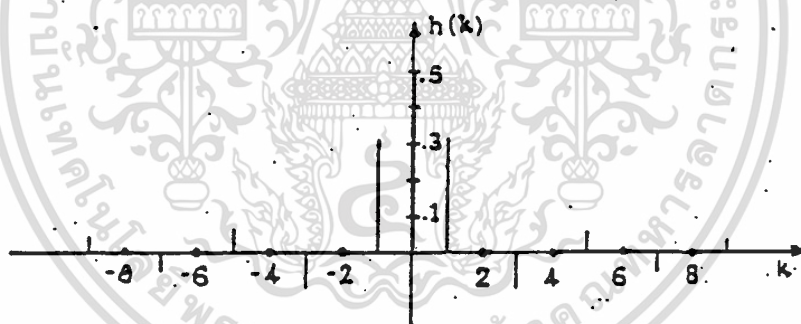
$$\begin{aligned} h_d(k) &= (1/2\pi) \int_{-\pi}^{\pi} H_d(\omega) e^{j\omega k} d\omega \\ &= (1/2\pi) \int_{-\pi/2}^{\pi/2} 1 \cdot e^{jn\omega} d\omega \\ &= (1/2\pi) \cdot (1/jn) \cdot e^{jn\omega} \Big|_{-\pi/2}^{\pi/2} \\ &= (1/2\pi) \cdot (1/jn) [\exp(jn\pi/2) - \exp(-jn\pi/2)] \\ h_d(k) &= (1/k\pi) \sin(\pi k/2) \end{aligned}$$

ดังนั้น สัมประสิทธิ์ของ Non-recursive Digital Filter เป็น

$$C_k = h_d(k)$$

ค่าสัมประสิทธิ์ของ Filter จะแสดงอยู่ในตาราง

$ k $	$h_d(k) = C_k$
0	0.5
1	0.3183
2	0
3	-0.1061
4	0
5	0.06
6	0
7	-0.05
8	0
9	0.04

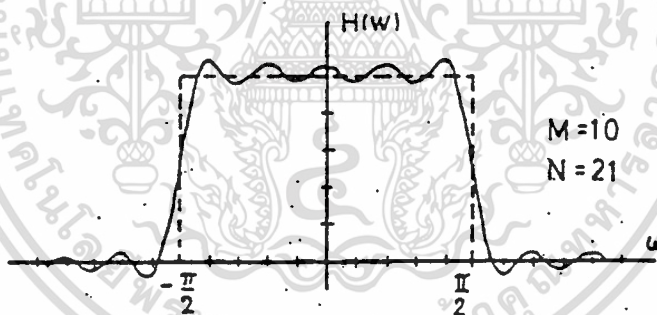
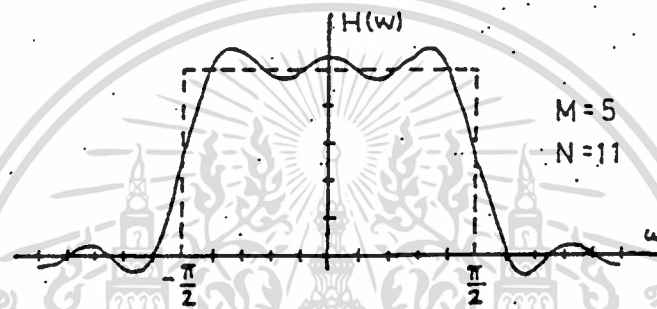
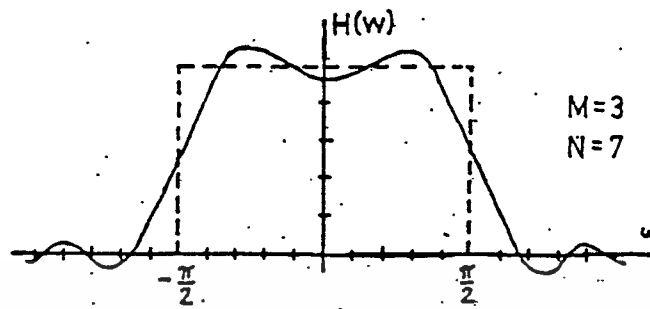


รูปที่ 3.2 Impulse Response ของตัวอย่าง

สัมประสิทธิ์ของ Non-recursive Filter สำหรับที่ $M = 9$ ได้แสดงไว้ในตาราง จากรูปที่ 3.2 จะเห็นว่า Filter ตัวนี้ มีความเสถียรดี เพราะ Impulse Response มีค่าเข้าใกล้ศูนย์ เมื่อ N เข้าใกล้อนันต์ จากรูปที่ 3.3 ได้แสดงผลตอบสนอง Amplitude เมื่อเปรียบเทียบกรณี $M = 3, 5, 10$ ตามลำดับ จากรูปทั้ง 3 จะสังเกตเห็นว่า ถ้า M หรือ N มีค่ามาก มีผลทำให้

1. ผลตอบสนอง Amplitude ใกล้เคียงผลตอบสนองในอุดมคติมากขึ้น
2. จำนวน Ripple มากขึ้นโดยที่คาบของลูกคลื่นลดลง
3. ผลตอบสนอง Amplitude มีความคมมากขึ้น หรือ แถบเปลี่ยนสถานะ แคบลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ผลตอบสนอง Amplitude สำหรับ Non-recursive Filter จากตัวอย่าง

ปรากฏการณ์ที่บริเวณใกล้จุดเปลี่ยนจาก Amplitude หนึ่งเป็นศูนย์ เกิดมี Ripple เราเรียกว่า Gibb's phenomena ผลนี้เกิดจากการที่ผลตอบสนองอุดมคติถูกแทนด้วย Fourier Series จำนวนจำกัด ซึ่งหมายถึงถูกแทนด้วยเซตของฟังก์ชัน ความถี่ของ Damping จำนวนจำกัดด้วย ซึ่งจำนวนนี้ไม่เพียงพอที่ผลรวมจะรวมกันแล้วเป็นเส้นตรงได้ ส่วนการที่ผลตอบสนองไม่เปลี่ยนจากเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าหนึ่งไปศูนย์ทันที แต่เกิดมีแถบเปลี่ยนสถานะขึ้น เพราะผลตอบสนองความถี่อุดมคติ ถูกเขียนแทนด้วยเซตของสัญญาณความถี่จำนวนจำกัด หรือ

$$H(\omega) = \sum_{k=-M}^M C_k \{ \cos(k\omega) - j \sin(k\omega) \}$$

ซึ่ง $H(\omega)$ ประกอบขึ้นจากสัญญาณ Sine และ Cosine จำนวนจำกัดมารวมกัน ความถี่สูงสุดของสัญญาณจะเป็น $M\omega$ ดังนั้นสัญญาณใน Time Domain ช่วงเวลา T_r ซึ่งเป็นช่วงเวลาที่สัญญาณจะเปลี่ยนได้เร็วที่สุดจะเป็น

$$T_r = \frac{1}{(M \cdot \omega)}$$

ซึ่งจะเห็นได้ว่า ถ้า M มากขึ้น แถบเปลี่ยนสถานะ ก็จะแคบลง

โดยทั่วไปในการออกแบบ Non-recursive Filter เราจะต้องการ

1. ลดจำนวน Ripple ให้น้อยที่สุด
2. ให้มีช่วงแถบเปลี่ยนสถานะแคบ

แต่ทั้งสองข้อนี้ไม่สามารถทำได้พร้อมกัน คือ ถ้าลดจำนวน Ripple แถบเปลี่ยนสถานะจะกว้างขึ้น ดังนั้นจึงจำเป็นต้องเลือกเอาอย่างใดอย่างหนึ่ง

เราจะสรุปการออกแบบ Non-recursive Filter โดยวิธี Fourier Series เป็นขั้นตอนได้ คือ

1. กำหนดผลตอบสนองอุดมคติที่ต้องการ โดยการกำหนดความถี่ตัด
2. ทำการอินทิเกรตผลตอบสนองอุดมคติ เพื่อหาลำดับ Impulse Response

$$h_d(k)$$

3. เลือกให้สัมประสิทธิ์ $C_k = h_d(k)$

การลด Ripple โดยใช้ Sigma Factor

เพื่อที่จะลด ขนาดของ Ripple บน Frequency Response ของ Filter เราจำเป็นต้องเปลี่ยนแปลงสัมประสิทธิ์ของ Fourier Series ที่ถูกตัดปลายเสียใหม่ หรือเรียกว่าการถ่วงน้ำหนัก (Weighted) เป็นวิธีการลด Ripple วิธีหนึ่งทำได้โดยการหาค่าประมาณ หรือค่าเฉลี่ยของ Ripple ที่เกิดขึ้น M Harmonic ซึ่งก็เท่ากับเป็นการหาค่า

$$A(\omega) = (M/2\pi) \int_{\omega-\pi/M}^{\omega+\pi/M} H(\omega) d\omega$$

เอกสาร โดยที่ $H(\omega)$ เป็นผลตอบสนองความถี่ดังสมการข้างต้น จากเมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$H(\omega) = a_0 + \sum_{k=1}^M a_k \cos(\omega k)$$

จะได้

$$\begin{aligned} A(\omega) &= (M/2\pi) [a_0] \left| \frac{\omega + \pi/M}{\omega - \pi/M} \right| + (M/2\pi) \sum_{k=1}^M [a_k \sin(\omega k)/k] \left| \frac{\omega + \pi/M}{\omega - \pi/M} \right| \\ &= a_0 + (M/2\pi) \sum_{k=1}^M (a_k/k) \left\{ \sin[(\omega + \pi/M)k] - \sin[(\omega - \pi/M)k] \right\} \\ &= a_0 + \sum_{k=1}^M \left\{ [\sin(\pi k/M)] / (\pi k/M) \right\} \cdot a_k \cos(\omega k) \\ &= a_0 + \sum_{k=1}^M \sigma(M, k) \cdot a_k \cos(\omega k) \end{aligned}$$

ซึ่งจะเห็นว่าผลที่ได้จากการเฉลี่ยค่านี้นี้ คือ Fourier Series ที่พจน์ Cosine ถูกคูณด้วย Sigma Factor $\sigma(M, k)$ และค่าสัมประสิทธิ์ Filter จะกลายเป็น

$$\begin{aligned} C_0 &= a_0 \\ C_k &= (1/2) \sigma(M, k) \cdot a_k \end{aligned}$$

โดยที่ C_k, C_0 เป็น Fourier Series ข้อสังเกต เทอม $\sigma(N, N) = (\sin \pi)/\pi = 0$ เพราะฉะนั้นเทอมที่ N จะมีค่าเป็นศูนย์ Filter ตามสมการจึงมีเทอมสัมประสิทธิ์ เท่ากับ $2(M-1)+1=2M-1$ เทอม หรืออีกนัยหนึ่งก็คือ ต้องใช้ $(2M-1)$ ตัว วิธีการลด Ripple ที่กล่าวมานี้ เป็นวิธีการลด Ripple โดยใช้ Window แบบหนึ่ง โดยมีชื่อเรียกว่า Lanczos Windowing

ตัวอย่าง แสดงสัมประสิทธิ์ของ Filter จากการให้ Sigma Factor โดย ถ้าให้

$$\begin{aligned} H(f) &= 1 && ; 0 < |f| < 0.2 \\ &= 0 && ; 0.2 < |f| < 0.5 \end{aligned}$$

จาก $a_k = 2C_k$ จะได้

$$\begin{aligned}
 a_k &= \left(\frac{1}{2\pi} \right) \int_{-\pi}^{\pi} H(\omega) e^{jk\omega T} d\omega \\
 &= \left(\frac{2}{2\pi} \right) \int_0^{\pi} H(\omega) \left\{ \frac{e^{j\omega k} + e^{-j\omega k}}{2} \right\} d\omega \\
 &= \left(\frac{2}{\pi} \right) \int_0^{\pi} H(\omega) \cos(k\omega) d(k\omega) \\
 &= 4 \int_0^{1/2} H(f) \cos(2\pi kf) df \\
 &= \left(\frac{2}{\pi k} \right) \sin(0.4(\pi k))
 \end{aligned}$$

$$H(f) = 0.4 + 2 \left\{ \sum_{k=1}^{\infty} \left\{ \frac{\sin(0.4(\pi k))}{\pi k} \right\} \cos(2\pi kf) \right\}$$

ถ้าเราทำการตัดปลายให้เหลือเพียง 5 พจน์ และ คำนวณ $\sigma(5, k)$ จะได้

$$\sigma(5, k) = \frac{\sin(\pi k/5)}{(\pi k/5)}$$

เพราะฉะนั้น

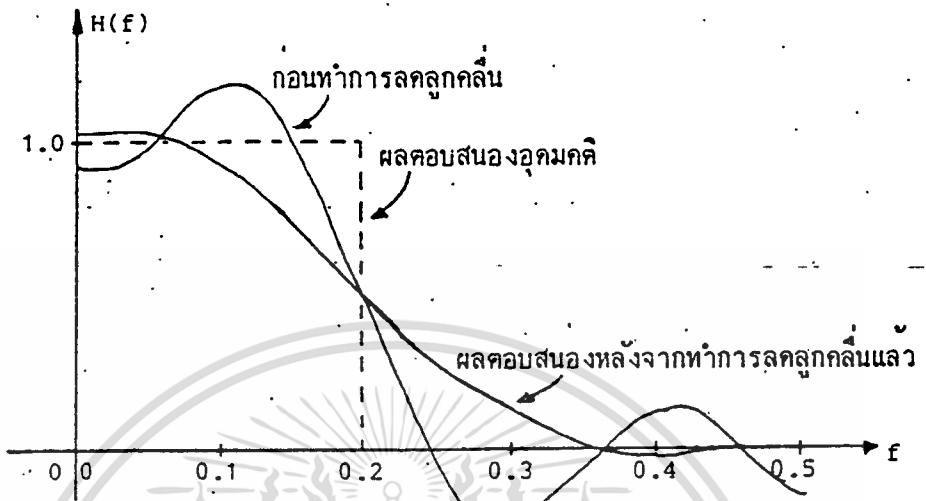
$$\begin{aligned}
 C_k &= (1/2) \left\{ \frac{2 (\sin(\pi k/5))}{(\pi k/5)} \right\} \left\{ \frac{\sin(0.4 \pi k)}{(\pi k)} \right\} \\
 &= \left\{ \frac{(\sin(\pi k/5))}{(\pi k/5)} \right\} \left\{ \frac{\sin(0.4 \pi k)}{(\pi k)} \right\}
 \end{aligned}$$

จะได้สัมประสิทธิ์ดังตาราง

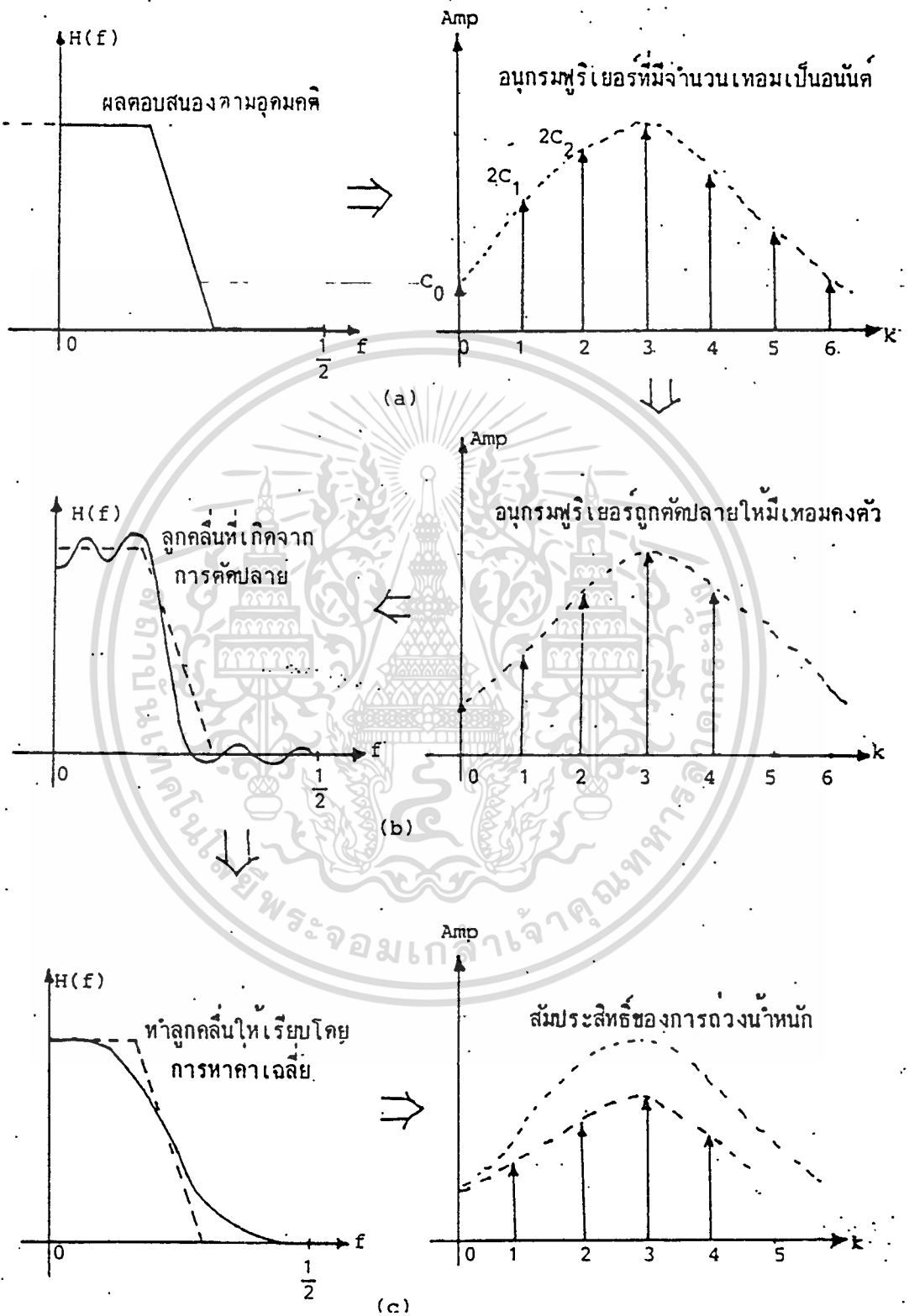
k	$\frac{\sin(\pi k/5)}{(\pi k/5)}$	$\frac{\sin(0.4 \pi k)}{(\pi k)}$	C_k
0	-	-	0.5
1	0.935	0.303	0.2833
2	0.757	0.093	0.0704
3	0.504	-0.062	-0.0312
4	0.234	-0.076	-0.0178

Frequency Response ของ Filter ที่ได้ ก่อนที่จะทำการลด Ripple และ หลังจากทำการลด Ripple ได้

เอกสารนี้จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 Frequency Response เมื่อพจน์ Cosine ของ Fourier Series ถูกถ่วงน้ำหนักด้วย Sigma Factor



รูปที่ 3.5 ผลของการแทน ผลตอบสนองความถี่ ด้วย Fourier Series จำนวนพจน์คงตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบโดยใช้ Window (Window Methods)

จากที่ผ่านมาเราได้แสดงให้เห็นว่าการถ่วงน้ำหนัก Fourier Series ให้มีความเหมาะสมสามารถทำให้ Ripple ที่เกิดขึ้นบน Frequency Response ลดน้อยลงไปได้ โดยทั่วไปการถ่วงน้ำหนักอาจทำได้หลายวิธี และเพื่อความสะดวกในการอธิบายขอนำขั้นตอนในการออกแบบมากล่าวถึงอีกครั้งหนึ่ง จากเดิมที่เราตัดปลาย Fourier Series เพื่อให้ได้ความยาวตามต้องการนั้น เปรียบเสมือนกับที่เราทำการเจาะช่องคล้ายเป็น Window ที่มีรูปร่างต่างๆ และเมื่อ Fourier Series จำนวนอนันต์พจน์ผ่าน Window นี้ออกไป ก็จะถูกถ่วงน้ำหนัก หรือ ถูกตัดปลาย ตามต้องการ

เราจะเริ่มจาก $H_d(\omega)$ แทนผลตอบสนองความถี่อุดมคติ และ ผลตอบสนองนี้ สามารถเขียนแทนด้วย Fourier Series ยาวอนันต์พจน์ ได้คือ

$$H_d(\omega) = \sum_{n=-\infty}^{\infty} h_d(n) e^{-jn\omega T}$$

โดยที่ $h_d(n)$ หาได้จาก

$$h_d(n) = (1/2\pi) \int_{-\pi}^{\pi} H_d(\omega) e^{jn\omega T} d\omega$$

or

$$h_d(n) = (1/\omega_s) \int_{-\omega_s/2}^{\omega_s/2} H_d(\omega) e^{jn\omega T} d\omega$$

โดยที่ ω_s เป็นค่าความถี่ในการสุ่มตัวอย่าง ในการออกแบบที่ผ่านมาเราจะเกิดปัญหาอยู่ 2 ประการ คือ

ประการแรก เราจะต้องทำการตัดปลายอนุกรม ให้เป็นอนุกรมจำนวนพจน์จำกัด หรือ

$$H_d(\omega) = \sum_{n=-(N-1)/2}^{(N-1)/2} h_d(n) e^{-jn\omega T}$$

ผลลัพธ์ก็คือ ทำให้เกิด Ripple บนผลตอบสนอง Amplitude

ประการที่สอง เมื่อตัดปลายอนุกรมแล้ว ยังไม่สามารถนำไปใช้งานได้ เพราะ Impulse Response เริ่มจากลำดับที่ $n=-(N-1)/2$ ซึ่งในระบบเวลาจริง ไม่สามารถสร้างลำดับที่เวลาเป็นลบได้ เราจึงทำการแก้ไขโดยการ ห่วง Impulse Response ออกไปโดยให้เริ่มจากค่า $n=0$ หรือ

$$H_d(\omega) \doteq \sum_{n=0}^{(N-1)/2} h_d(n) e^{-jn\omega T}$$

จะเห็นได้ว่าการที่เราตัดปลายของ Fourier Series ก็คือ การที่เราทำการเลือกให้

$$h(n) = \begin{cases} h_d(n) & ; 0 \leq n \leq N-1 \\ 0 & ; otherwise \end{cases}$$

ซึ่งสมการนี้ ถ้าหากเขียนให้อยู่ในรูปของสมการทั่วไปก็คือ การนำเอาลำดับ $h_d(n)$ มาคูณค่ากับลำดับจำกัด $w(n)$ หรือ

$$h(n) = h_d(n) \cdot w(n)$$

โดยที่ $w(n)$ แทนลำดับของ Window แบบต่างๆ สำหรับสมการที่ผ่านมานั้น เป็นแบบ Rectangular Window ที่ลำดับ $w(n)$ มีคุณสมบัตินิยามได้เป็น

$$w_R(n) = \begin{cases} 1 & ; 0 \leq n \leq N-1 \\ 0 & ; otherwise \end{cases}$$

ซึ่งต่อไปเราจะอธิบายถึงคุณสมบัติของ Frequency Response ของ Rectangular Window

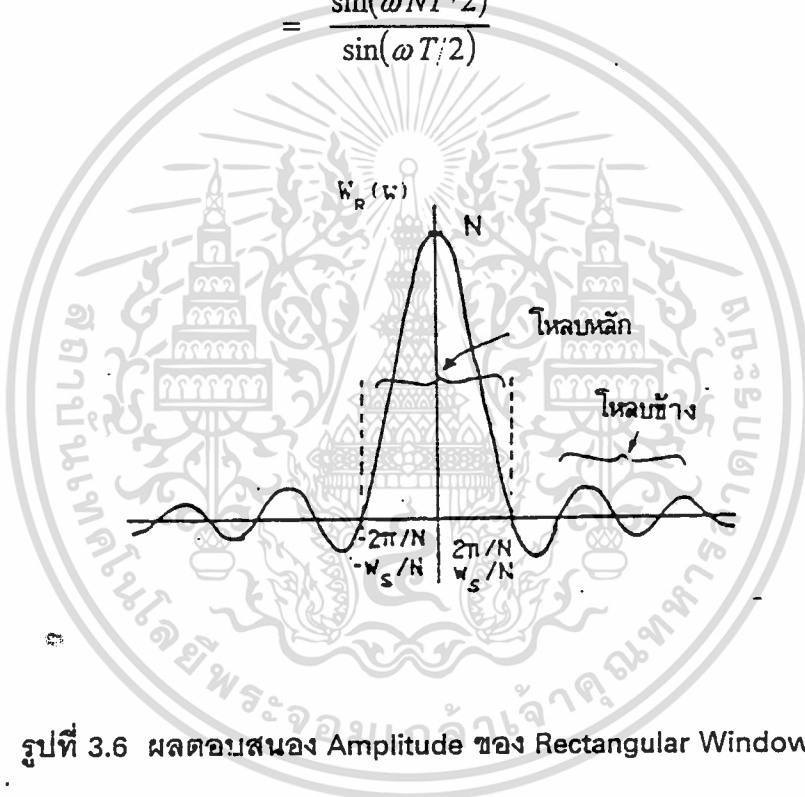
คุณสมบัติของ Rectangular Window

สำหรับ Rectangular Window อาศัยนิยามโดยทั่วไป เป็น

$$w_R(n) = \begin{cases} 1 & ; |n| \leq (N-1)/2 \\ 0 & ; otherwise \end{cases}$$

Frequency Response ของ Window แบบนี้ หาได้โดยการประยุกต์ใช้ ผลการแปลง z เข้ากับสมการ แล้วทำการแทนค่าให้ $z = e^{-j\omega T}$ เมื่อจัดรูปสมการใหม่ จะได้

$$\begin{aligned}
 R(\omega) &= \sum_{n=-(N-1)/2}^{(N-1)/2} e^{-j\omega nT} \\
 &= \frac{\{e^{j\omega(N-1)T/2} - e^{-j\omega(N+1)T/2}\}}{\{1 - e^{-j\omega T}\}} \\
 &= \frac{\{e^{j\omega NT/2} - e^{-j\omega NT/2}\}}{\{e^{j\omega T/2} - e^{-j\omega T/2}\}} \\
 &= \frac{\sin(\omega NT/2)}{\sin(\omega T/2)}
 \end{aligned}$$



รูปที่ 3.6 ผลตอบสนอง Amplitude ของ Rectangular Window

จากรูป แสดงผลตอบสนอง Amplitude ของ Rectangular Window ซึ่งเห็นได้ว่ามีคุณสมบัติดังนี้

1. ความกว้างของ Main Lobe มีค่า $4\pi/N$ และมีค่าขนาดเป็น N
2. เมื่อ N มีค่ามากขึ้น ความกว้างของ Main Lobe ลดลง
3. Side Lobe มีการแกว่งโดยผ่านจุดศูนย์ทุกๆ ค่าความถี่ $2\pi/N$ และมีขนาดลดลงไปเรื่อยๆ และ $R(\omega) = 0$ ณ ค่าความถี่ $\omega = m \omega_s/N$ เมื่อ m ค่าบวกและลบ ของ $1, 2, \dots$ ผลนี้เห็นได้ว่า ถ้า N มีค่ามาก Side Lobe จะเข้ามารวมกันใกล้ ความถี่ $\omega = 0$ มากขึ้น

4. ถ้านิยามให้ Ripple Ratio (RR) เป็น

$$RR = \frac{\text{Maximum Main Lobe}}{\text{Maximum Side Lobe}} * 100$$

การที่เรานำเอาลำดับ Impulse Response มาคูณกับลำดับของ Window ซึ่งเป็นปฏิบัติการใน Time Domain เมื่อเราพิจารณาใน Frequency Domain ก็เท่ากับเป็นการนำเอา ผลตอบสนองความถี่จุดมคติ มาทำการ Convolution กับ Frequency Response ของ Window หรือ

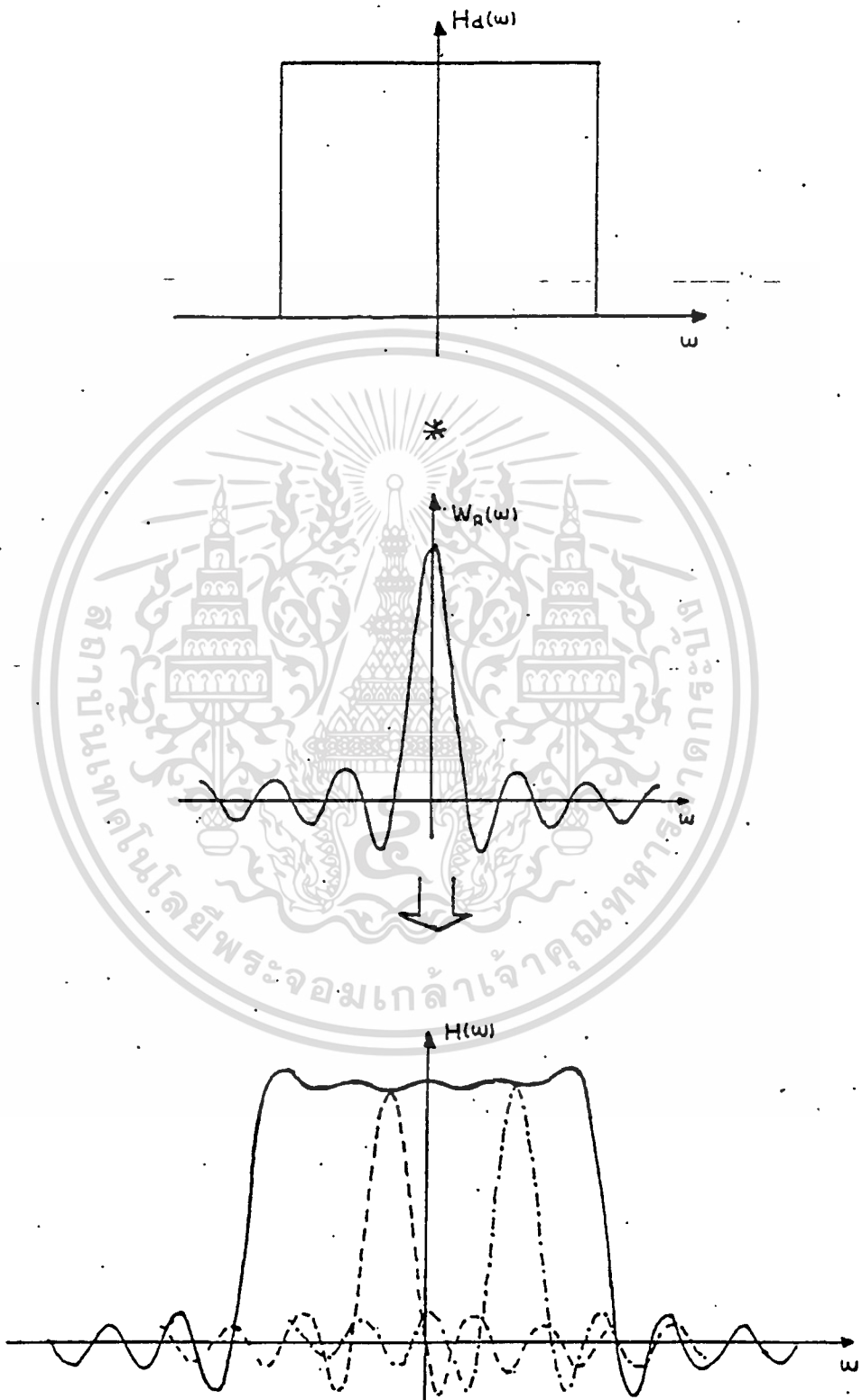
$$H(\omega) = (1/2\pi) \int_{-\pi}^{\pi} H_d e^{j\theta} \cdot W_R e^{j(\omega-\theta)} d\theta$$

การ Convolution นี้ อธิบายได้โดยใช้แผนภาพ รูปที่ 3.7 ซึ่งจะเห็นได้ชัดว่า

1. ค่าความชันของ Main Lobe ของสเปกตรัม ของ Window ทำให้เกิดแถบเปลี่ยนสถานะของ Frequency Response เพราะฉะนั้น ถ้า N มาก ก็จะทำให้แถบสถานะแคบลง
2. Ripple บน Frequency Response เป็นผลมาจาก Side Lobe ของสเปกตรัม ของ Window ดังนั้นถ้าเลือก N ค่ามาก Side Lobe จะเลื่อนเข้าใกล้ความถี่ $\omega=0$ มากขึ้น เป็นผลทำให้เกิด Ripple บนยอดของ Frequency Response มากขึ้น

ด้วยผลทั้งสองประการนี้ เราจึงต้องหา Window หรือดัดแปลง Rectangular Window เพื่อให้มีคุณสมบัติ คือ

1. ให้ Main Lobe ของ Frequency Response แคบ แต่ภายใน Main Lobe สามารถครอบคลุม พลังงานของสัญญาณทั้งหมดไว้ให้ได้มากที่สุด
2. ให้ Frequency Response ของ Window มีค่าขนาดของ Side Lobe น้อย และมีค่าขนาดลดลงอย่างรวดเร็ว เมื่อมีความถี่สูงขึ้น



รูปที่ 3.7 แสดงการปฏิบัติการทำ Window ใน Frequency Domain

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Hann, Hamming และ Blackman Window

Hann Window และ Hamming Window สามารถเขียนเป็นสมการของลำดับได้ คือ

$$w_H(n) = \begin{cases} \alpha + (1-\alpha) \cos\left(\frac{2\pi n}{N-1}\right) & ; |n| \leq (N-1)/2 \\ 0 & ; \text{otherwise} \end{cases}$$

โดยที่ Hann Window $\alpha = 0.5$ และ Hamming Window $\alpha = 0.54$ Frequency Response ของ Window ทั้งสองแบบ สามารถหาได้ตามลำดับดังนี้

เรานำสมการของ Window ทั้งสองแบบ มาเขียนในพจน์ของ Rectangular Window ได้คือ

$$\begin{aligned} H(n) &= W_R(n) \left\{ \alpha + (1-\alpha) \right\} \cos\left(\frac{2\pi n}{N-1}\right) \\ &= \alpha W_R(n) + \left\{ \frac{1-\alpha}{2} \right\} W_R(n) \left[\exp\left\{ \frac{j 2\pi n}{N-1} \right\} + \exp\left\{ \frac{-j 2\pi n}{N-1} \right\} \right] \end{aligned}$$

เมื่อประยุกต์ใช้ผลการแปลง z กับสมการ จะได้ Frequency Response เป็น

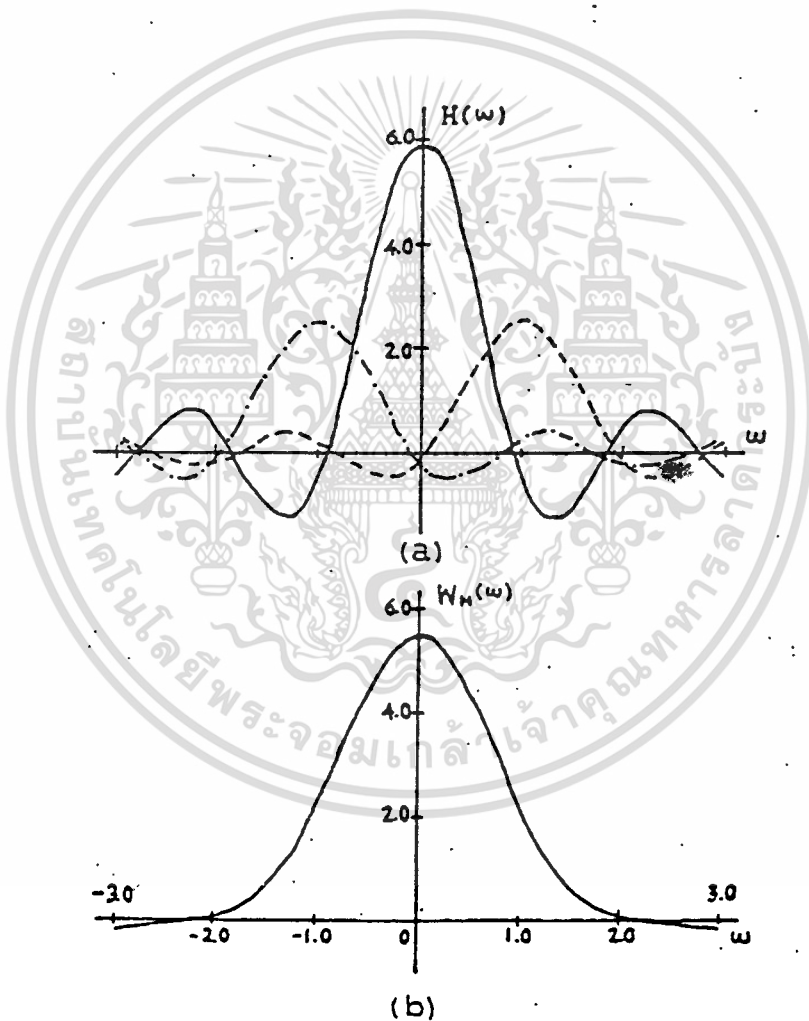
$$\begin{aligned} H\{\exp(j\omega T)\} &= \zeta[W_H(n)] \Big|_{z=e^{j\omega T}} \\ &= \alpha W_R\{\exp(j\omega T)\} + \left[\left(\frac{1-\alpha}{2} \right) W_R \left\{ \exp\left\{ j \left(\omega T - \frac{2\pi}{N-1} \right) \right\} \right\} \right] \\ &\quad + \left[\left(\frac{1-\alpha}{2} \right) W_R \left\{ \exp\left\{ j \left(\omega T + \frac{2\pi}{N-1} \right) \right\} \right\} \right] \end{aligned}$$

แทนค่าที่ $W_R\{\exp(j\omega T)\}$ ลงในสมการ จะได้

$$\begin{aligned} H\{\exp(j\omega T)\} &= \alpha \frac{\sin(\omega N/2)}{\sin(\omega/2)} + \frac{1-\alpha}{2} \frac{\sin\{(\omega N/2) - (N\pi/(N-1))\}}{\sin\{(\omega/2) - (\pi/(N-1))\}} \\ &\quad + \frac{1-\alpha}{2} \frac{\sin\{(\omega N/2) + (N\pi/(N-1))\}}{\sin\{(\omega/2) + (\pi/(N-1))\}} \end{aligned}$$

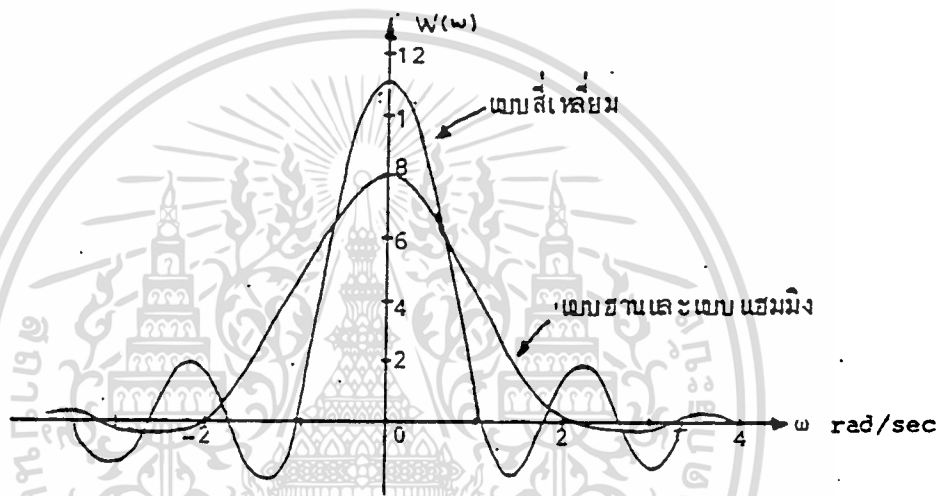
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเปกตรัมของ Window นี้ได้วาดไว้ในรูปที่ 3.8 ซึ่งเห็นได้ว่า เทอมที่ 2 และ 3 ของสมการ เป็นการถ่วงน้ำหนัก $r(e^{j\omega x})$ แล้วเลื่อนไปทางซ้ายและขวา $2\pi/(N-1)T$ ลำดับ การทำอย่างนี้จุดประสงค์เพื่อไปหักล้าง กับ Side Lobe ทั้งสองข้าง แล้วไปเพิ่มขนาดความกว้างของ Main Lobe ขึ้นมา ผลการรวมดังรูป 3.8 (b) เห็นได้ชัดว่า Main Lobe เพิ่มขึ้น และ Ripple ลดลงเป็นอย่างมาก ความกว้างของ Main Lobe ของ Window นี้มีค่าประมาณ $4\omega_s/N$



รูปที่ 3.8 สเปกตรัมของ Hann และ Hamming

ในรูปที่ 3.9 นั้นเป็นการเปรียบเทียบกันระหว่างสเปกตรัมความถี่ของ Rectangular Window Hann Window และ Hamming Window ซึ่งเห็นได้ชัดว่าแบบ Rectangular พลังงานในส่วนความถี่นอก Main Lobe มีอยู่มาก ส่วนในแบบ Hann และ Hamming พลังงานได้ถ่ายเข้าไป Main Lobe เกือบ 99 %



รูปที่ 3.9 สเปกตรัมของ Rectangular Window และ Hann ,Hamming
กรณี $N=11$

สำหรับลำดับ Blackman Window สามารถเขียนได้ คือ

$$w_B(n) = \begin{cases} 0.42 + 0.5 \cos\left(\frac{2n\pi}{N-1}\right) + 0.08 \cos\left(\frac{4n\pi}{N-1}\right) & ; |n| \leq (N-1)/2 \\ 0 & ; \text{otherwise} \end{cases}$$

สเปกตรัมความถี่ไม่ได้แสดงการหาไว้ แต่ Window นี้มีคุณสมบัติที่ดีกว่า แบบ Hann และ Hamming คือสามารถถ่ายพลังงานเข้าไป Main Lobe ได้มากกว่า เราจะสรุปคุณสมบัติของ Window ต่างๆ เพื่อเป็นการเปรียบเทียบไว้ในตาราง

ชนิดของ Window	Main Lobe Width	Ripple ratio (%) N = 31	Peak-Amp of Side Lobe (dB)	Min-stopband att ⁿ (dB)
Rectangular	$2\omega_s/N$	21.80	- 13	- 21
Hann	$4\omega_s/N$	2.67	- 31	- 44
Hamming	$4\omega_s/N$	0.82	- 41	- 53
Blackman	$6\omega_s/N$	0.12	- 57	- 74

Window	Ripple ratio (%)		
	N = 11	N = 21	N = 31
Rectangular	22.34	21.89	21.80
Hann	2.62	2.67	2.67
Hamming	1.47	0.93	0.82
Blackman	0.08	0.12	0.12

โดยที่ ω_s คือความถี่สุ่ม ($= 2\pi$)

การออกแบบ Filter โดยใช้ Window

วิธีการออกแบบโดยใช้ Window ใน Time Domain Frequency Response ของ Low pass Filter อุดมคติ จะเป็นดังรูปที่ 3.10 (a) หรือ

$$H(e^{j\omega T}) = \begin{cases} 1 & ; |\omega| \leq \omega_c \\ 0 & ; \omega_c \leq |\omega| \leq \omega_s/2 \end{cases}$$

การหาค่าความสัมพันธ์ของ Frequency Response ของ Filter กับ Impulse Response ของ Filter ทำโดยการอาศัยการกระจาย Fourier Series คือ

$$H(e^{j\omega T}) = \sum_{n=-\infty}^{\infty} h(nT) e^{-jn\omega T}$$

and

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรนำออกไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อแทนค่าจากสมการ ผลตอบสนองความถี่อุดมคติ ลงในสมการ จะได้

$$\begin{aligned} h(nT) &= (1/\omega_c) \int_{-\omega_c}^{\omega_c} e^{j n \omega T} d\omega \\ &= \left(\frac{1}{n\pi} \right) \{ \sin(n\omega_c T) \} \end{aligned}$$

สมการที่ได้นี้ เป็น Impulse Response ของ Low pass Filter

จากสมการ
$$H(e^{j\omega T}) = \sum_{n=-\infty}^{\infty} h(nT) e^{-j n \omega T}$$

ทำการเปลี่ยนให้อยู่ในรูปของ z - Transform โดยแทน $z = e^{j\omega T}$ จะได้

$$H(z) = \sum_{n=-\infty}^{\infty} h(nT) z^{-n}$$

ดังที่กล่าวมาแล้วในทางปฏิบัติ Filter นี้มี Impulse Response จำกัด เพราะฉะนั้นเราจะทำการตัดปลาย Fourier Series โดยให้

$$h(nT) = 0 \quad ; |n| > (N-1)/2$$

จะได้สมการเป็น

$$H(z) = h(0) + \sum_{n=1}^{(N-1)/2} \{ h(-nT)z^n + h(nT)z^{-n} \}$$

แต่เนื่องจากเราจะต้องทำให้ Filter เป็น Causal ซึ่งทำได้โดย การคูณสมการด้วยพจน์ $z^{-(N-1)/2}$ และถ้าให้ $H'(z)$ เป็น Filter ที่ได้ หรือ

$$H'(z) = z^{-(N-1)/2} \cdot H(z)$$

การคูณด้วยพจน์นี้เข้าไปจะไม่ทำให้ ผลตอบสนอง Amplitude เปลี่ยนแปลงแต่อย่างใด แต่ทำให้ Phase Response ของ Filter มีคุณสมบัติเป็น Linear มากขึ้น เมื่อทำการแทนสมการ จะได้

$$H'(z) = z^{-(N-1)/2} \left\{ \sum_{n=0}^{(N-1)/2} h(nT) (z^n + z^{-n}) \right\}$$

ในทำนองเดียวกัน เรานำสมการซึ่งเป็น Frequency Response ของ Filter ที่ได้จากการตัดปลาย Fourier Series โดยใช้ Window Function จะได้

$$\begin{aligned} H'(z) &= z^{-(N-1)/2} \left\{ \sum_{n=0}^{(N-1)/2} w(nT) h(nT) (z^n + z^{-n}) \right\} \\ &= z^{-(N-1)/2} \left\{ \sum_{n=0}^{(N-1)/2} (a_n/2) (z^n + z^{-n}) \right\} \end{aligned}$$

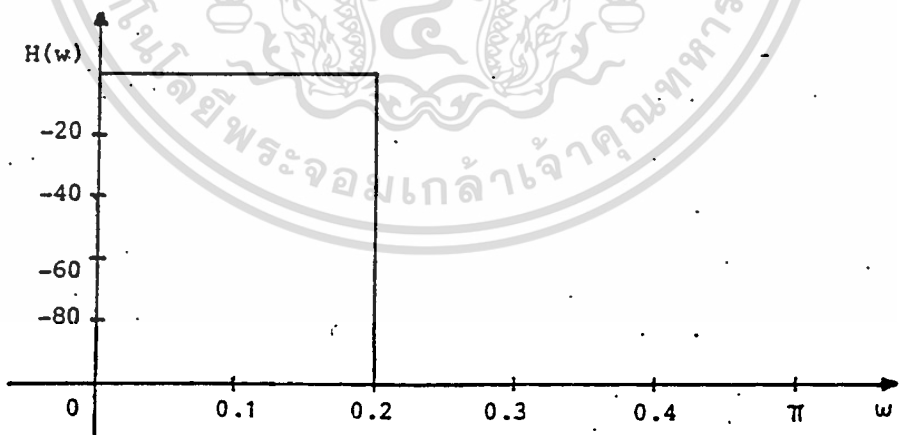
โดยการเปรียบเทียบสมการทั้งสอง โดยให้ a_n เป็นสัมประสิทธิ์ของ Non-recursive Filter ที่ต้องการ ซึ่งจะมีค่าคือ

$$a_0 = w(0).h(0)$$

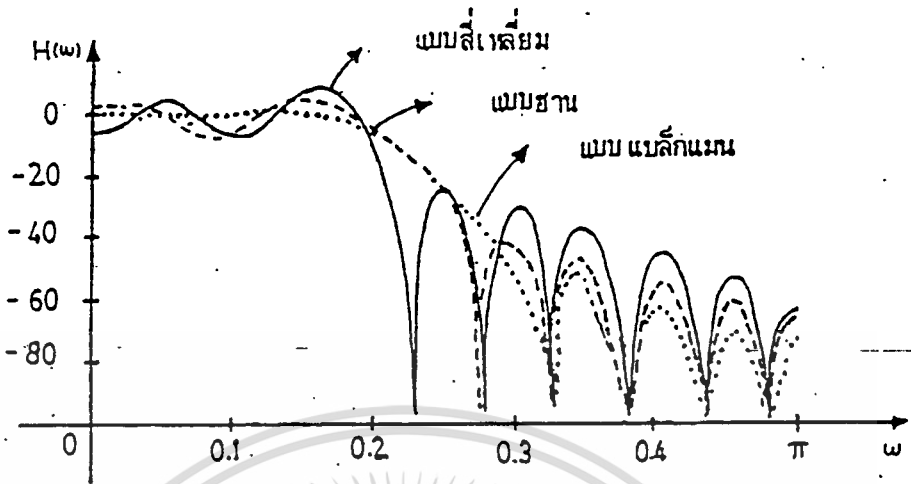
and

$$a_n = 2.w(nT).h(nT)$$

โดยที่ $w(nT)$ Impulse Response ของ Window แบบต่างๆ



(a) ผลตอบสนองแอมพลิจูดของตัวกรองความถี่ต่ำผ่านอวกมคติ



(b) ผลตอบสนองแอมพลิจูดโดยใช้วินโดว์แบบต่าง ๆ

รูปที่ 3.10 Frequency Response จากการใช้ Window แบบต่าง ๆ

กรณี $\omega_c = 2 \text{ rad/s}$, $\omega_s = 10 \text{ rad/s}$ and $N = 2$

จากสมการผลตอบสนอง Amplitude จะมีค่าเท่ากับ

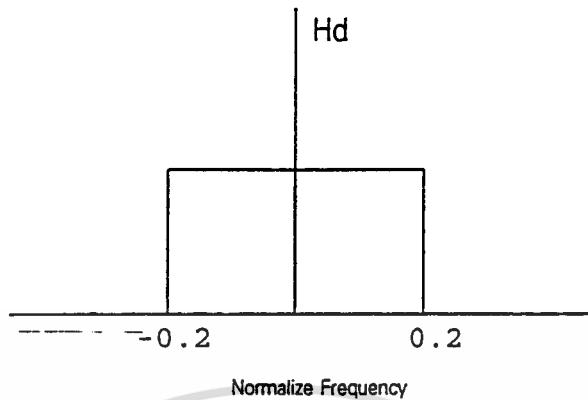
$$M(\omega) = \left| \sum_{n=0}^{(N-1)/2} a_n \cos(n\omega T) \right|$$

เราจะแสดงผลตอบสนอง Amplitude ของ Window แบบต่าง ๆ ไว้ในรูปที่ 3.10 (b) จะเห็นได้จากการเปรียบเทียบว่า เมื่อมีการทำการถ่วงน้ำหนักด้วย Window Ripple ในแถบผ่านความถี่ จะลดลง แต่ค่าลดทอนน้อยสุดในแถบหยุด (Minimum Stopband Attenuation) และแถบเปลี่ยนสถานะจะเพิ่มขึ้น

ตัวอย่าง FIR Lowpass Filter ให้ออกแบบ Lowpass Filter of order 11 โดยให้ Cutoff Frequency = 1 kHz และมีค่าความถี่สุ่มเป็น 10 kHz

วิธีทำ จาก

$$\begin{aligned} C_n &= \int_0^{\gamma} H_d(\gamma) \cos(n\pi \gamma) d\gamma \\ &= \frac{\sin(n\pi \gamma)}{n\pi} \Big|_0^{\gamma} \\ &= \frac{\sin(0.2n\pi)}{n\pi} \end{aligned}$$



โดยค่า γ หาได้จาก $\gamma = f_c/f_N = 2f_c/f_s = 2(10^3)/10^4 = 0.2$

$$C_0 = 0.2$$

$$C_n = \frac{\sin(0.2n\pi)}{n\pi} \quad n=1, \dots, N$$

$$C_0 = 0.2$$

$$C_1 = 0.1592$$

$$C_5 = 0$$

จะได้

$$h_0 = C_{-5} = C_5 = 0$$

$$h_1 = C_{-4} = C_4 = 0.0467744$$

$$h_2 = C_{-3} = C_3 = 0.1009102$$

$$h_3 = C_{-2} = C_2 = 0.1513653$$

$$h_4 = C_{-1} = C_1 = 0.1870978$$

$$h_5 = 0.2$$

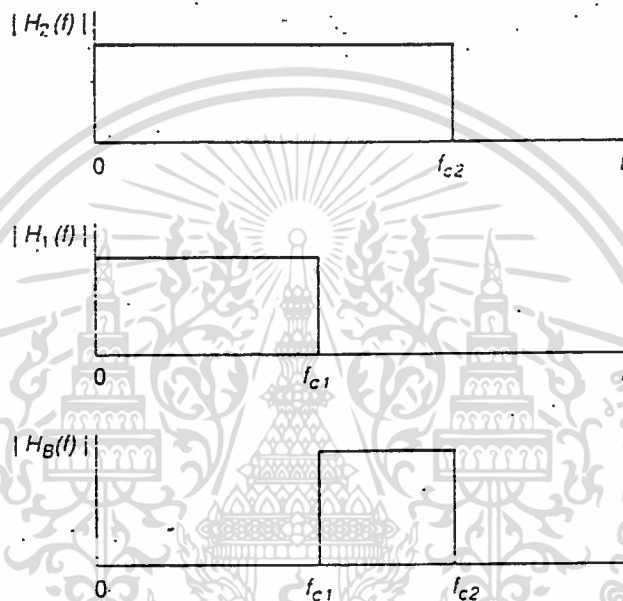
โดยที่ $h_i = C_{Q-i}$ and $C_{-n} = C_n$

การออกแบบ Bandpass, Highpass and Bandstop Filters

การออกแบบ Filter แบบอื่นๆ เช่น Bandpass, Highpass and Bandstop นั้น สามารถทำได้
ในเงื่อนไขของ Lowpass จากสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ $C_n = \frac{\sin[(2f_c/f_s)n\pi]}{n\pi}$ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะยกตัวอย่างการสร้าง Bandpass Filter เราสามารถสร้าง Bandpass Filter ได้โดยการสร้าง Lowpass Filter ที่ครอบคลุมถึงช่วง Upper Cutoff Frequency แล้วจึงสร้าง Lowpass Filter ตัดความถี่ช่วง Lower Cutoff Frequency เราก็จะได้ Bandpass Filter ตามที่ต้องการ เราจะสามารถแสดงการ Cutoff ความถี่บนแกน Normalize Frequency ได้ดังรูป

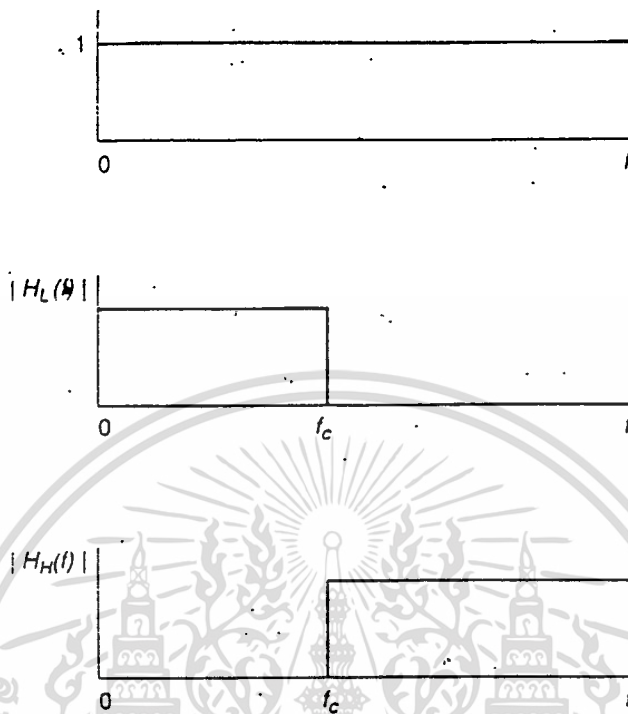


Graphical description of the bandpass calculation

เราจะสามารถหาค่า C_n ของ Bandpass Filter ได้โดยการนำค่าสัมประสิทธิ์ C_n ของ Lowpass Filter ทั้งสอง มาทำการลบกัน เราสามารถเขียนสมการหาค่า C_n ของ Bandpass Filter ได้ดังนี้

$$C_n = \frac{\sin[(2f_{c2}/f_s)n\pi]}{n\pi} - \frac{\sin[(2f_{c1}/f_s)n\pi]}{n\pi}$$

ต่อจากนี้ เราจะแสดงการออกแบบ Highpass Filter เราสามารถออกแบบ Highpass Filter ได้จาก Lowpass Filter โดยนำเงื่อนไขของ Lowpass Filter มาทำการลบกับค่าคงที่ ของผลตอบสนอง Amplitude เราสามารถแสดงได้ดังรูป



Graphical description of the highpass calculation

ใน Time Sequence เราจะให้ค่าคงที่ของ ผลตอบสนอง Amplitude เป็น $\delta(n)$ ดังนั้นเราจะเขียนสมการการหาค่า C_n ของ Highpass Filter ได้ดังนี้

$$C_n = \delta(n) - \frac{\sin[(2f_c - f_s)n\pi]}{n\pi}$$

ในการสร้าง Bandstop Filter ก็สามารทำได้ในวิธีเดียวกัน โดยใช้วิธีแผนภาพ โดยการสร้าง Lowpass Filter และสร้าง Highpass Filter ให้ Lowpass Filter Cutoff ในช่วง Lower Cutoff Frequency และให้ Highpass Filter Cutoff ในช่วง Upper Cutoff Frequency แล้วนำช่วงของ Filter ทั้งสอง มาทำการบวกกัน เราก็จะสามารถสร้าง Bandstop Filter ได้ ค่าที่ได้จากการคำนวณจากการสร้าง Filter แบบต่างๆ สามารถนำไปใช้กับ Window Function ต่างๆ ได้ตามปกติ

Kaiser Window

Kaiser Window เป็น Window ที่ถูกพัฒนาให้มีคุณสมบัติที่ดีขึ้น สามารถปรับค่าได้ และการออกแบบสามารถทำได้เป็นระบบมากขึ้น โดยการใช้ ฟังก์ชันทรงคล้ายทรงกลมแบนข้าง (Prolate Spheroidal Function) มาทำการประมาณค่า ผลที่ได้ก็คือ ได้ Window ที่มีคุณสมบัติ แลกเปลี่ยน ระหว่าง ค่าขนาดของ Main Lobe กับค่าขนาดของ Side Lobe โดยมีพารามิเตอร์ α เป็นตัวควบคุมค่าเพื่อปรับค่าขนาด Side Lobe เมื่อเปรียบเทียบค่าขนาดของ Main Lobe ได้ ส่วน ความกว้างของ Main Lobe ก็เหมือนกับ Window อื่นๆ คือ ปรับค่าได้โดยการเลือก N Kaiser Window นี้ อาจเรียกได้ว่า มีคุณสมบัติ เกือบเหมาะสมที่สุด อย่างไรก็ตาม ฟังก์ชันทรงคล้ายทรงกลมแบนข้าง เป็นฟังก์ชันที่ไม่รู้จักกันทั่วไป และยังทำการคำนวณได้ยาก Kaiser จึงได้พัฒนาขั้นตอนการออกแบบขึ้นมา

ลำดับของ Kaiser Window ที่อยู่ในรูปของฟังก์ชันทรงคล้ายทรงกลมแบนข้าง คือ

$$w_K(nT) = \begin{cases} I_0(\beta)/I_0(\alpha) & ; |n| \leq (N-1)/2 \\ 0 & ; \text{otherwise} \end{cases}$$

โดยที่ α เป็นพารามิเตอร์อิสระ ที่สัมพันธ์กับพารามิเตอร์ β โดย

$$\beta = \alpha \cdot \sqrt{1 - \left(\frac{2n}{N-1}\right)^2}$$

และ $I_0(x)$ เป็น Zero th - order Bessel Function ชนิดที่หนึ่ง ที่สามารถคำนวณได้โดยใช้ Power Series คือ

$$\begin{aligned} I_0(x) &= 1 + \sum_{k=1}^{\infty} \left\{ \left(\frac{1}{k!}\right) \cdot \left(\frac{x}{2}\right)^k \right\}^2 \\ &= 1 + \left(\frac{x}{2}\right)^2 + \left\{ \frac{(x/2)^4}{(2!)^2} \right\} + \left\{ \frac{(x/2)^6}{(3!)^2} \right\} + \left\{ \frac{(x/2)^8}{(4!)^2} \right\} + \dots \end{aligned}$$

ในการใช้งานทั่วไป ความละเอียดในการคำนวณตามสมการ Power Series ใช้เพียง 15 ถึง 25 พจน์ ก็เพียงพอ

เราจะพิจารณาพจน์ α ที่มีผลต่อคุณสมบัติของ Kaiser Window พิจารณากรณี $\alpha = 0$ จะเห็นได้ว่า พจน์เศษ และ พจน์ส่วน ของสมการ Window มีค่าเป็นหนึ่ง นั่นคือ $w_K(nT)$ จะทำตัวเป็น Rectangular Window และอาจพิสูจน์ได้ว่า ถ้า $\alpha = 0.54414$ แล้ว $w_K(nT)$ จะให้คุณสมบัติเป็น Hamming Window ถ้าหาก α มีค่าเพิ่มมากขึ้น Main Lobe ของ Window จะกว้างขึ้น ส่วนค่าขนาดของ Side Lobe จะมีค่าลดลง จากการพิจารณานี้ จะเห็นได้ว่า พจน์ α สามารถใช้เป็นพารามิเตอร์ ควบคุมคุณสมบัติของ Kaiser Window ให้มี Ripple น้อยกว่า Blackman Window จนมีอัตรา Ripple เท่ากับ Rectangular Window ได้

สเปกตรัมของ Kaiser Window หาได้โดย การประยุกต์ผลการแปลง z กับสมการของ Kaiser Window และให้ $z = e^{j\omega T}$ ได้ผลเป็น

$$K(e^{j\omega T}) = W_K(0) + 2 \cdot \sum_{n=1}^{(N-1)/2} w_K(nT) \cos(n\omega T)$$

ซึ่งสามารถแสดงได้ว่า

$$K(\omega) = \left(\frac{2}{I_0(\alpha)} \right) \sin \left(\frac{\tau \sqrt{\omega^2 - \omega_b^2}}{\sqrt{\omega^2 - \omega_b^2}} \right)$$

โดยที่ให้ $\omega_b = \alpha/\tau$ และ $\tau = (N-1)T/2$ และ ω_b เป็นค่าที่กำหนดขึ้นมาเพื่อเป็นการแลกเปลี่ยนกันระหว่างความกว้างของ Main Lobe กับค่าของ Side Lobe

จากสมการ เราสามารถหาค่าความกว้างของ Main Lobe โดยการหาดำแหน่งที่ $(j\omega) = 0$ ครั้งแรก แล้วคูณค่าด้วย 2 ได้เป็นผล

$$C_w = 2\omega_b \sqrt{1 + \left(\frac{\pi}{\alpha} \right)^2}$$

โดยที่ค่า C_w คือค่าความกว้างของ Main Lobe และจากสมการ เราสามารถหาค่า $(j\omega)$ ที่ตำแหน่งที่ค่าอนุพันธ์ของ $(j\omega)$ จะเป็นศูนย์ครั้งแรกแล้วนำไปคิดเป็นเปอร์เซ็นต์ กับค่า (0) จะได้ค่าอัตรา Ripple เป็น

$$RR = \frac{0.218 \alpha}{\sinh h(\alpha)} * 100$$

จากตาราง 3.1 ได้แสดงค่าของสมการการหาค่า C_w และ RR โดยแปรค่า α อยู่ระหว่าง 4 ถึง 9 ซึ่งเป็นค่า α ที่ใช้งานโดยทั่วไป จะเห็นได้ว่า RR แปรจากค่า 3.2% ถึง 0.048% ดังนั้น โดยทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มักใช้ค่า α ระหว่าง 4 ถึง 9 ส่วนในตาราง 3.2 ได้แสดงค่าขนาด peak to peak ของ Main Lobe ซึ่งค่านี้เป็นผลทำให้เกิด Ripple ที่บริเวณจุดไม่ต่อเนื่องของ Frequency Response จะเห็นได้ว่าเมื่อ α มีค่ามากขึ้น จะทำให้ Main Lobe มีขนาดลดลง

ตารางที่ 3.1 ความกว้างของ Main Lobe และ Ripple ratio ของ Kaiser Window
เมื่อให้ $\omega_b = 1$ และ เปลี่ยนค่า α

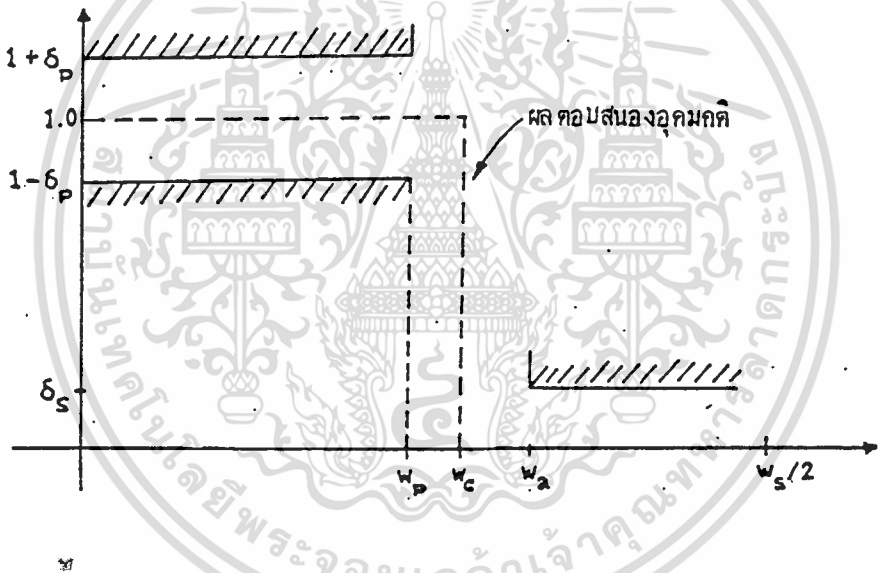
α	ความกว้างของ Main Lobe	Ripple ratio
4	2.54	3.19%
5	2.36	1.47%
6	2.26	0.648%
7	2.19	0.278%
8	2.15	0.117%
9	2.12	0.0483%

ตารางที่ 3.2 ค่า peak to peak ของ Ripple ที่เป็นผลมาจาก Side Lobe ของ Kaiser Window

α	ค่า peak to peak เป็นเปอร์เซ็นต์ของ Ripple
0.0	17.898%
5.0	0.329%
6.0	0.140%
6.5	0.083%
7.0	0.050%
7.5	0.029%
8.0	0.017%
8.5	0.010%

การออกแบบโดยใช้ Kaiser Window

จากรูปที่ 3.11 เป็นการแสดงถึงข้อกำหนดในการออกแบบ วงจร Low pass Filter โดยทั่วไป พารามิเตอร์ δ_p และ δ_s เป็นตัวกำหนด คุณสมบัติของค่าขนาดของ Ripple ใน แถบผ่าน และ คุณสมบัติของค่าขนาดของ Ripple ในแถบหยุด ตามลำดับ ถ้ากำหนดให้ A_p เป็นค่าขนาดของ Ripple ในแถบผ่าน A_s เป็นค่าลดทอนในแถบหยุดน้อยที่สุด และ B , เป็นค่าความกว้างของแถบ เปลี่ยนสถานะ เราสามารถเขียนความสัมพันธ์ได้ว่า



รูปที่ 3.11 ข้อกำหนดของ Frequency Response ที่ต้องการออกแบบ

$$A_p = 20 \log \left\{ \frac{1 + \delta_p}{1 - \delta_p} \right\}$$

$$A_s = -20 \log \delta_s$$

$$B = \omega_s - \omega_p$$

ในการเขียนต่อไปนี ถ้าสมมุติให้ว่า A'_p และ A'_s เป็นค่าขนาดของ Ripple ในแถบผ่าน และค่าลดทอนในแถบหยุดน้อยที่สุด ที่กำหนดมาจากผลตอบสนอง รูป 3.11 และให้ A_p และ A_s เป็นค่าที่ได้จากการออกแบบ ดังนั้นขั้นตอนในการออกแบบ Filter โดยใช้ Kaiser Window เป็นดังนี้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. หา Impulse Response $h(n)$ โดยการประยุกต์ใช้ Fourier Series กับผลตอบสนองอุดมคติ คือ

$$H_d(\omega) = \begin{cases} 1 & ; |\omega| \leq \omega_c \\ 0 & ; \omega_c < |\omega| \leq \omega_s/2 \end{cases}$$

โดยจากรูปที่ 3.11

$$\omega_s = (1/2) (\omega_p + \omega_a)$$

2. เลือกพารามิเตอร์จากสมการ จนได้ค่าที่ทำให้

$$A_p \leq A'_p \quad ; \quad A_a \geq A'_a$$

และเลือก

$$\delta = \min(\delta_p, \delta_s)$$

การเลือก δ ตามสมการ $\delta = \min(\delta_p, \delta_s)$ เนื่องจากว่า การเลือกตามเงื่อนไขสมการ

$A_p = 20 \log \left\{ \frac{1+\delta_p}{1-\delta_p} \right\}$, $A_a = -20 \log \delta_s$ ทำให้เกิดข้อขัดแย้งคือ ได้ Filter ที่ทำให้

$\delta = \delta_p = \delta_s$ เสมอ

3. คำนวณหา A_a จากสมการ $A_a = -20 \log \delta_s$ โดยใช้ค่า δ จาก

$$\delta = \min(\delta_p, \delta_s)$$

4. เลือกพารามิเตอร์ α ให้สอดคล้องกับสมการข้างล่างนี้ คือ

$$\alpha = \begin{cases} 0 & ; A_a \leq 21 \\ 0.5842 (A_a - 21)^{0.4} + 0.07886 (A_a - 21) & ; 21 < A_a \leq 50 \\ 0.1102 (A_a - 8.7) & ; A_a > 50 \end{cases}$$

5. เลือกพารามิเตอร์ D ให้สอดคล้องกับ

$$D = \begin{cases} 0.9222 & ; A_a \leq 21 \\ (A_a - 7.95)/14.36 & ; A_a > 21 \end{cases}$$

6. คำนวณหาค่า N ที่เป็นเลขคี่ที่ต่ำที่สุดที่ทำให้ได้เงื่อนไข

$$N \geq \left(\frac{\omega_s \cdot D}{B_r} \right) + 1$$

7. คำนวณหา $w_K(nT)$ จากสมการของ Kaiser Window และตัดแปลง Impulse Response ตามสมการ จะได้เป็น

$$h(n) = h_d(n) \cdot w_K(n) \quad ; |n| \leq (N-1)/2$$

8. Transfer Function ของ Filter เขียนได้เป็น

$$H(z) = z^{-(N-1)/2} \left\{ h(0) + 2 \sum_{n=0}^{(N-1)/2} h(n) (z^n + z^{-n}) \right\}$$

$$\begin{aligned} \text{by } h(0) &= h_d(0) \cdot w_K(0) \\ h(n) &= h_d(n) \cdot w_K(n) \end{aligned}$$

และผลตอบสนอง Amplitude ของวงจร Filter หาได้จาก

$$|H(\omega)| = h(0) + 2 \cdot \sum_{n=1}^{(N-1)/2} h(n) \cos(2n\pi fT)$$

ตัวอย่าง การออกแบบ วงจร Low pass Filter

โดยกำหนดให้ ช่วงแถบผ่านความถี่อยู่ในช่วง 1 ถึง 5 rad/s และมีค่าขนาดของ Ripple ในแถบผ่านความถี่เป็น $A_p \leq 0.1 \text{ dB}$ และให้ แถบหยุดความถี่อยู่ในช่วง 2.5 ถึง 5.0 rad/s และมีค่าลดทอนในแถบหยุดน้อยสุด $A_s \leq 40 \text{ dB}$ โดยที่ให้ใช้ค่าความถี่ในการสุ่มตัวอย่าง 10 rad/s จงออกแบบตัวกรองนี้

วิธีทำ การคำนวณทำตามขั้นตอนในการออกแบบตามลำดับ ดังนี้

ขั้นตอนที่ 1 ได้
$$h(nT) = \frac{\sin(\omega_c nT)}{n\pi}$$

ให้ค่า
$$\omega_c = \frac{(1.5+2.5)}{2} = 2.0 \text{ rad/s}$$

ขั้นตอนที่ 2 จากสมการ $A_p = 20 \log \left\{ \frac{1 + \delta_p}{1 - \delta_p} \right\}$

จะได้ $\delta_1 = \left\{ \frac{(10^{0.05(0.1)} - 1)}{(10^{0.05(0.1)} + 1)} \right\} = 5.7564 \times 10^{-3}$

จากสมการ $A_s = -20 \log \delta_s$

จะได้ $\delta_2 = 10^{-0.05(40)} = 0.0$

เราจะเลือก $\delta = \min(\delta_1, \delta_2) = 5.7564 \times 10^{-3}$

ขั้นตอนที่ 3 ถึง 6 ได้ $A_s = 44.979 \text{ dB}$

ขั้นตอนที่ 4 และ 5 ให้ $\alpha = 3.9524$; $D = 2.566$

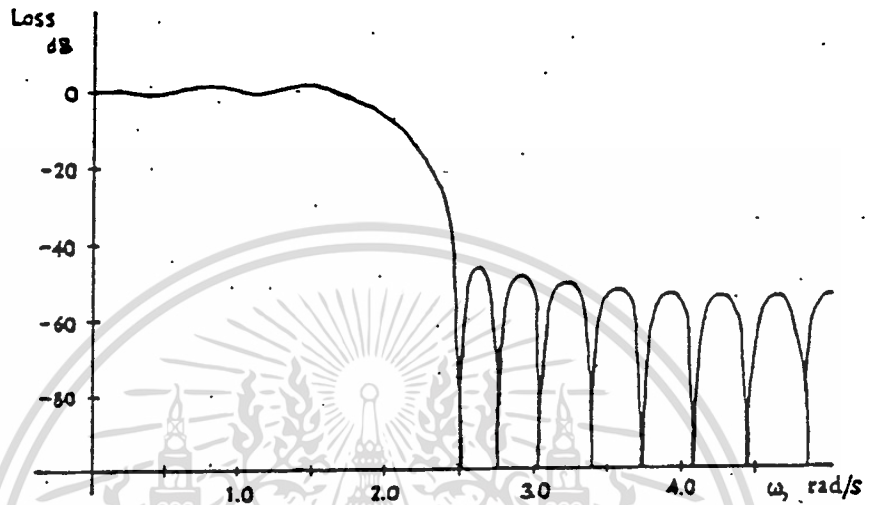
และ $N \geq \left\{ \frac{10(2.566)}{1} \right\} + 1 = 26.6 \approx 27$

ขั้นตอนที่ 6 และ 7 ได้ สัมประสิทธิ์ของ Filter ซึ่งจะเขียนเป็นตัวอย่างให้ 7 ตัวแรก ดังนี้

n	$h(nT)$	$w_k(nT) h(nT)$
0	0.4	0.4
1	0.302731	0.299692
2	0.0935489	0.0898359
3	-0.062366	-0.0569018
4	-0.0756827	-0.0642052
5	0	0
6	0.0504551	0.0345003
7	0.0267283	0.0157769

และ Frequency Response เป็นดังรูปที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 ผลตอบสนอง Amplitude ของ Kaiser Window ในตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

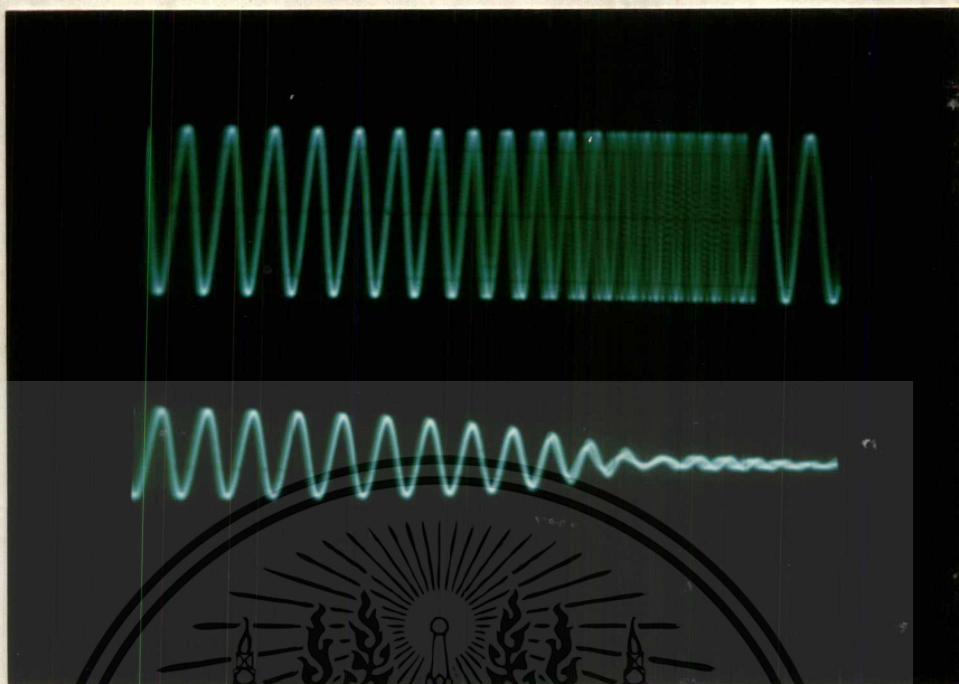
บทที่ 4

ผลการทดลอง

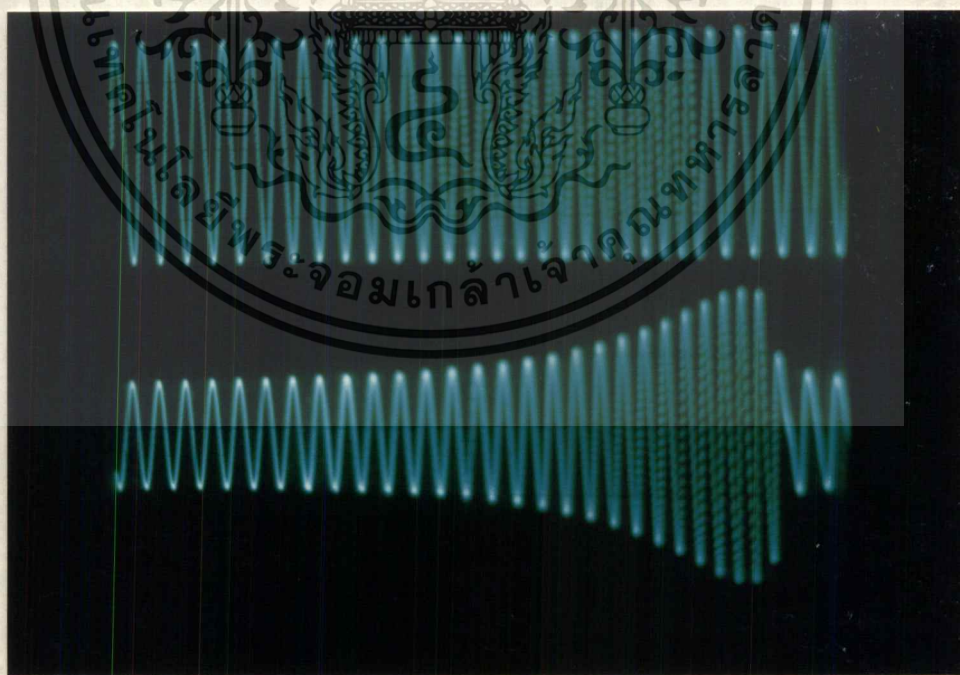
ในการทดลอง Real time digital filter นี้ได้แบ่งการทดลองออกเป็น 2 ตอน

- ตอนที่ 1 จะทดลองผลการทำงานของ Board TMS 320C50 เมื่อทำงานเป็น filter ต่างๆ ว่าจะได้ตาม ideal หรือไม่ โดยใช้ Function generater เป็นตัวจ่ายสัญญาณ Sweep sine เป็น input และใช้ oscilloscope เป็นตัวแสดงผล โดยทำการจับสัญญาณทั้ง input และ output เพื่อที่จะได้มองเห็น และสามารถเปรียบเทียบสัญญาณทั้งสองได้ง่าย โดยให้ chanal 1 จับสัญญาณ input (input จะอยู่ด้านบนของจอ oscilloscope) และให้ chanal 2 จับสัญญาณ output (output จะอยู่ด้านล่างของจอ oscilloscope) แล้วทำการ load program จาก PC. สู่ Board TMS 320C50 โดยให้ Board TMS 320C50 ทำงานเป็น Low pass filter ที่มีความถี่ cutoff ที่ 1K Hz และเป็น High pass filter ที่มีความถี่ cutoff ที่ 1K Hz ซึ่งผลที่ได้จะแสดงดัง รูปที่ 4.1 และ รูปที่ 4.2

- ตอนที่ 2 จะทำการทดลองกรองสัญญาณ Square wave และ Saw tooth เพื่อจะดูสัญญาณพื้นฐาน (Fundamental signal) ของสัญญาณทั้งสอง โดยใช้ Function generater เป็นตัวกำเนิดสัญญาณและใช้เป็น input เพื่อป้อนเข้าสู่ Board TMS 320C50 ส่วนภาคแสดงผลยังคงใช้ oscilloscope เป็นตัวแสดงผลเหมือนตอนที่ 1 วิธีการทดลอง จะ load program จาก PC. สู่ Board TMS 320C50 เพื่อให้ Board TMS 320C50 นี้ทำงานเป็น Low pass filter ที่มีความถี่ cutoff 1K Hz และ ที่ความถี่ cutoff 2K Hz แล้วป้อนสัญญาณทั้งสองตามลำดับ ผลการทดลองจะแสดงดัง รูปที่ 4.3 ถึง รูปที่ 4.4 (เมื่อ input เป็น Square wave) และ รูปที่ 4.9 ถึง รูปที่ 4.10 (เมื่อ input เป็น Saw tooth) ต่อมาจะทำการเปลี่ยนโหมดการทำงานของ Board TMS 320C50 เป็น Band pass filter ที่มีความถี่ cutoff ต่างๆ กันดังนี้ 1) ที่ความถี่ cutoff 0.9K Hz ถึง 1K Hz, 2) ที่ความถี่ cutoff 1.9K Hz ถึง 2K Hz, 3) ที่ความถี่ cutoff 2.9K Hz ถึง 3K Hz, 4) ที่ความถี่ cutoff 3.9K Hz ถึง 4K Hz แล้วทำการป้อนสัญญาณทั้งสองตามลำดับ ผลการทดลองที่ได้จะแสดงดัง รูปที่ 4.5 ถึง รูปที่ 4.8 (เมื่อ input เป็น Square wave) และ รูปที่ 4.11 ถึงรูปที่ 4.14 (เมื่อ input เป็น Saw tooth)

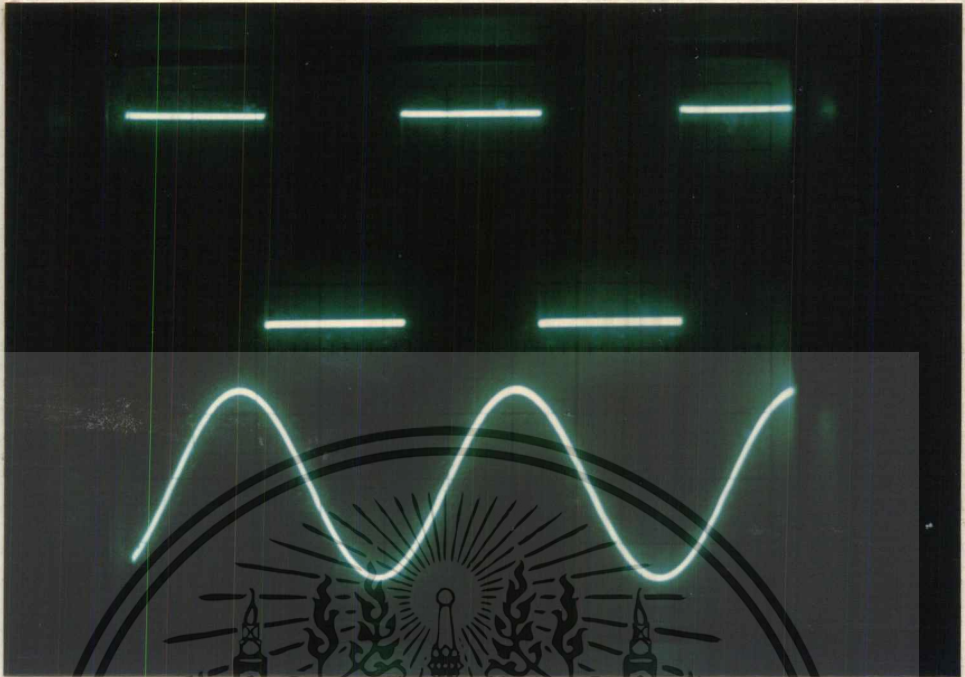


รูปที่ 4.1 เมื่อ input เป็น Sweep sine ที่มีความถี่ 1.4K Hz ผ่าน Low pass filter ที่มีความถี่ cut off 1K Hz โดยใช้สเกล T/D 1ms, Vo/D 2v, Vi/D 2v

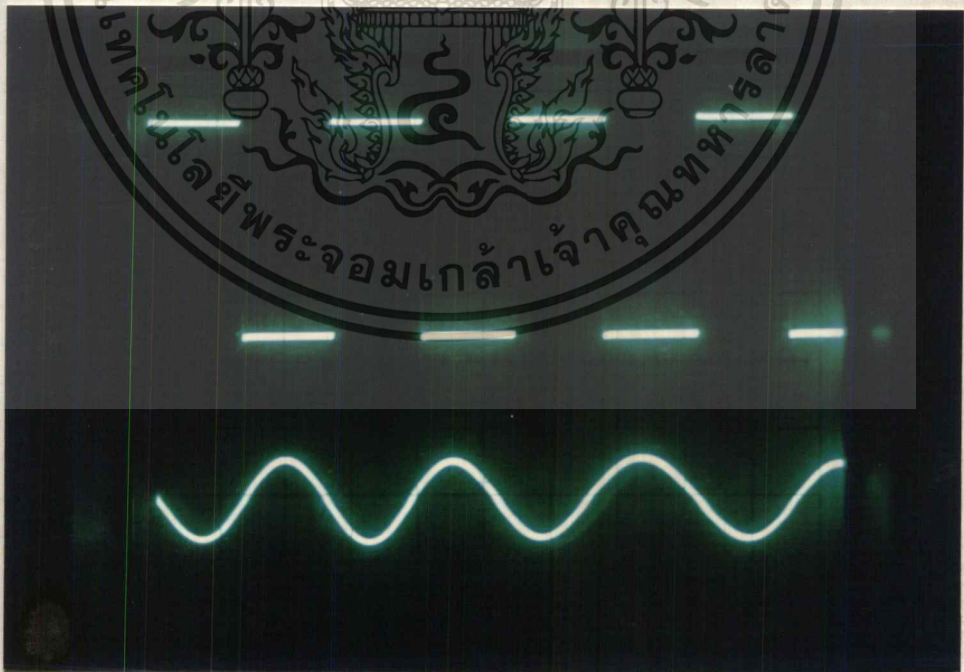


รูปที่ 4.2 เมื่อ input เป็น Sweep sine ที่มีความถี่ 1.2K Hz ผ่าน High pass filter ที่มีความถี่ cut off 1K Hz โดยใช้สเกล T/D 2ms, Vo/D 0.5v, Vi/D 2v

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

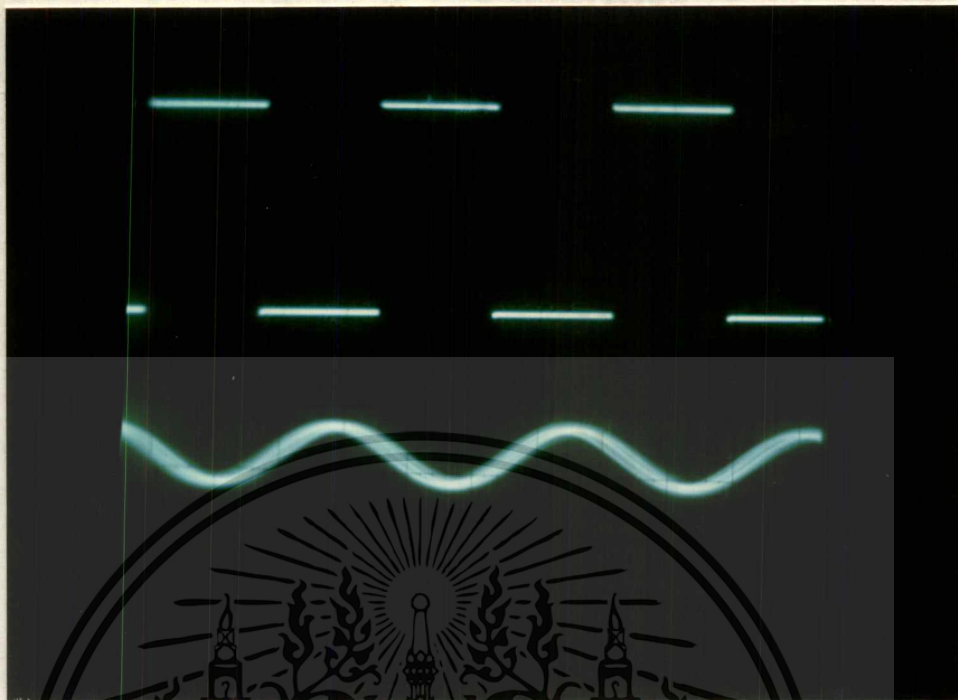


รูปที่ 4.3 เมื่อ input เป็น Square wave ที่มีความถี่ 1K Hz ผ่าน Low pass filter ที่มีความถี่ cut off 1K Hz โดยใช้สเกล T/D 2ms, Vo/D 2v, Vi/D 2v

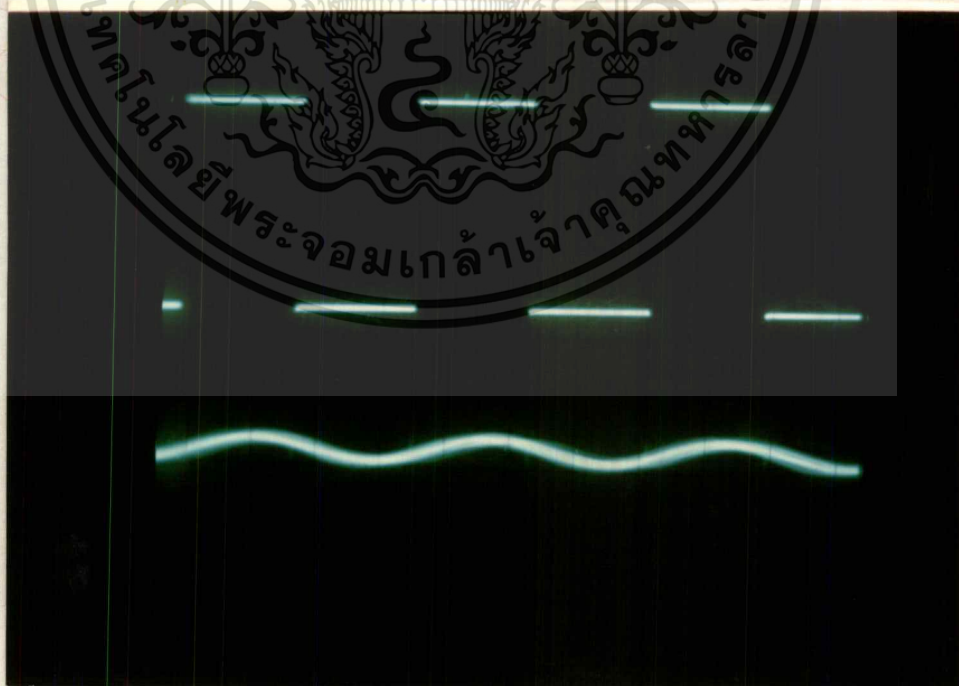


รูปที่ 4.4 เมื่อ input เป็น Square wave ที่มีความถี่ 2K Hz ผ่าน Low pass filter ที่มีความถี่ cut off 2K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.5v, Vi/D 2v

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

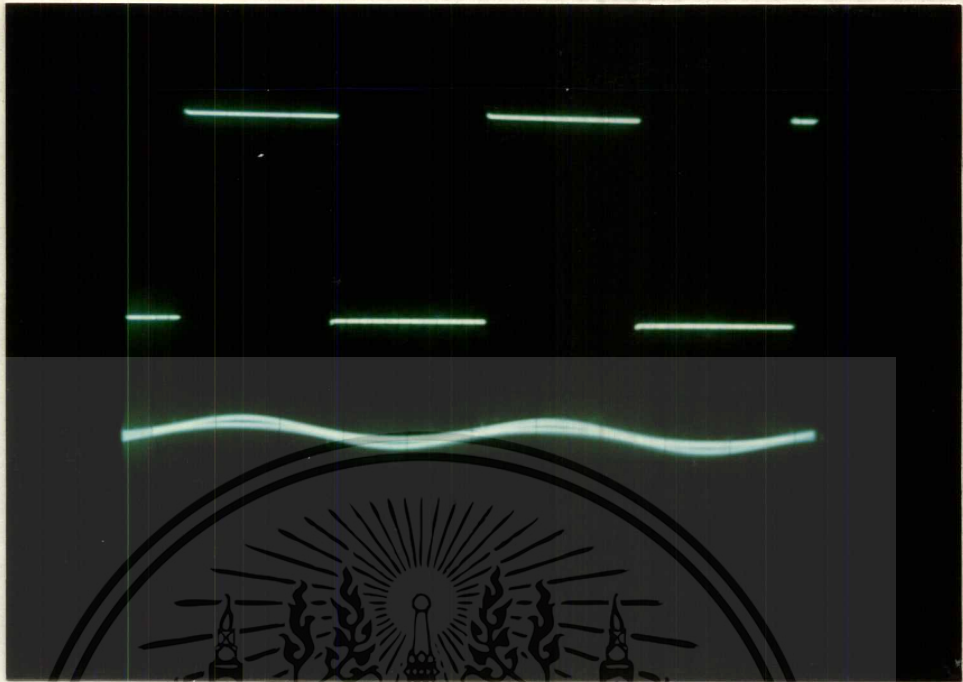


รูปที่ 4.5 เมื่อ input เป็น Square wave ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cut off 0.9K Hz ถึง 1K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

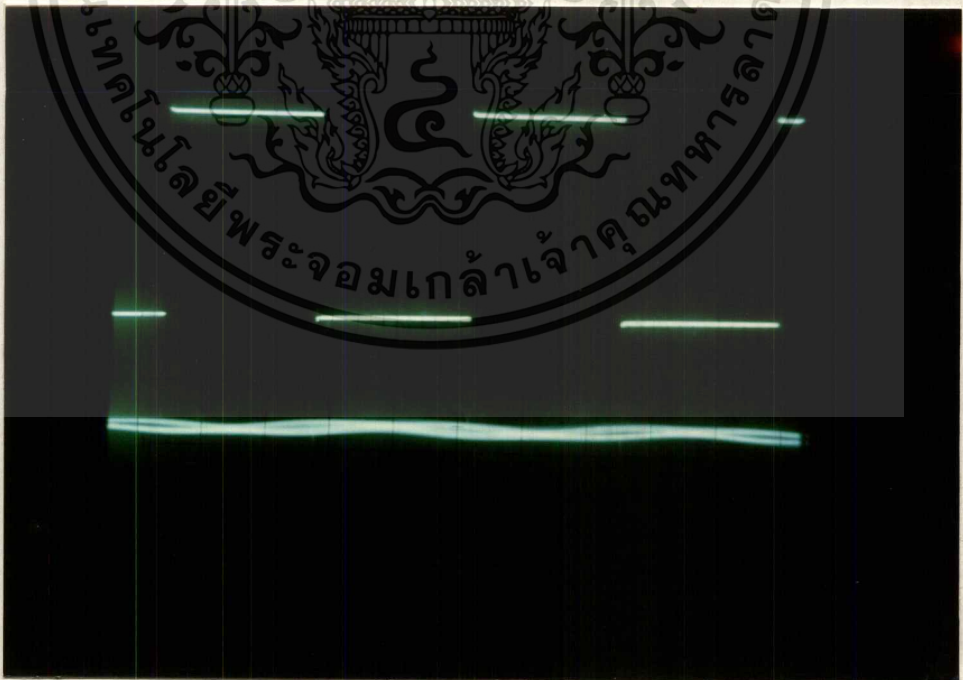


รูปที่ 4.6 เมื่อ input เป็น Square wave ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cut off 1.9K Hz ถึง 2K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนสิทธิ์ในชื่อของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้เผยแพร่โดยไม่ขออนุญาตจากฝ่ายวิชาการ
ไม่วารณใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

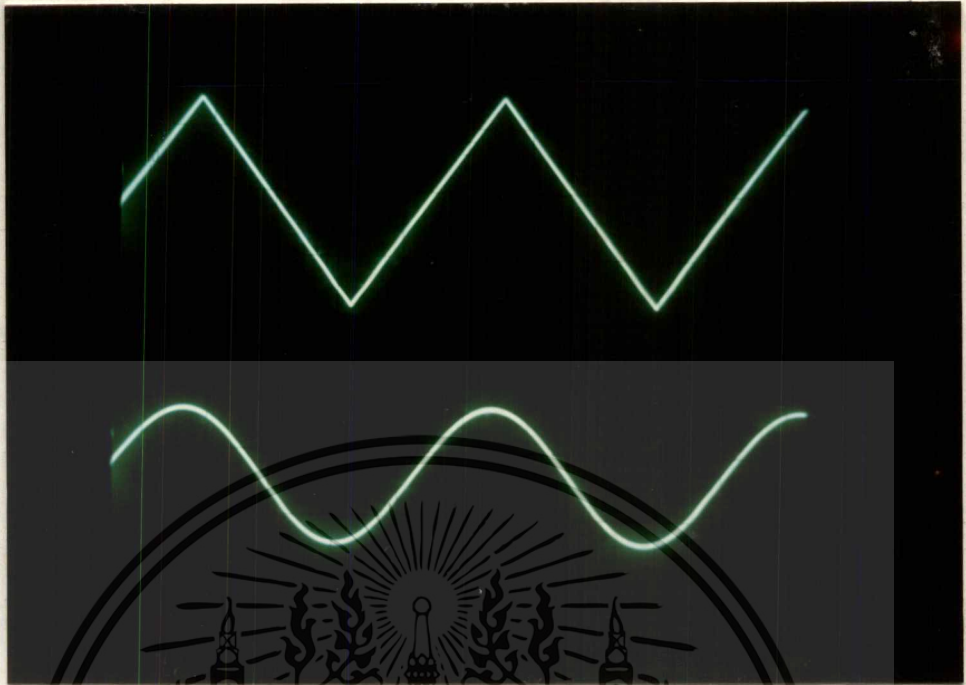


รูปที่ 4.7 เมื่อ input เป็น Square wave ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cutoff 2.9K Hz ถึง 3K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

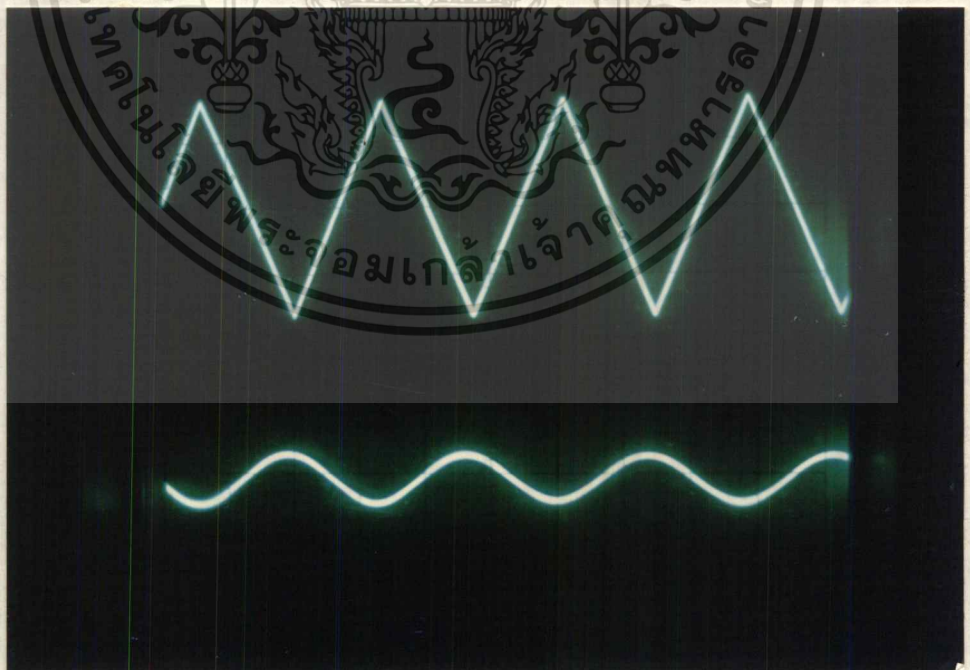


รูปที่ 4.8 เมื่อ input เป็น Square wave ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cut off 3.9K Hz ถึง 4K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดก็ตาม ห้ามนำไปใช้เพื่อวัตถุประสงค์อื่น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

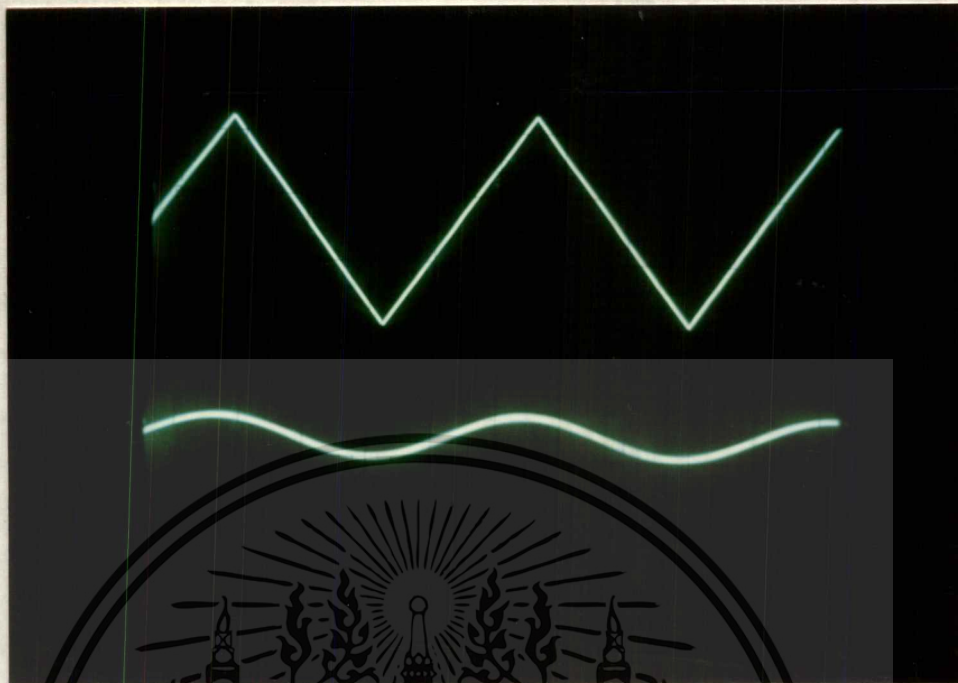


รูปที่ 4.9 เมื่อ input เป็น Saw tooth ที่มีความถี่ 1K Hz ผ่าน Low pass filter ที่มีความถี่ cut off 1K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

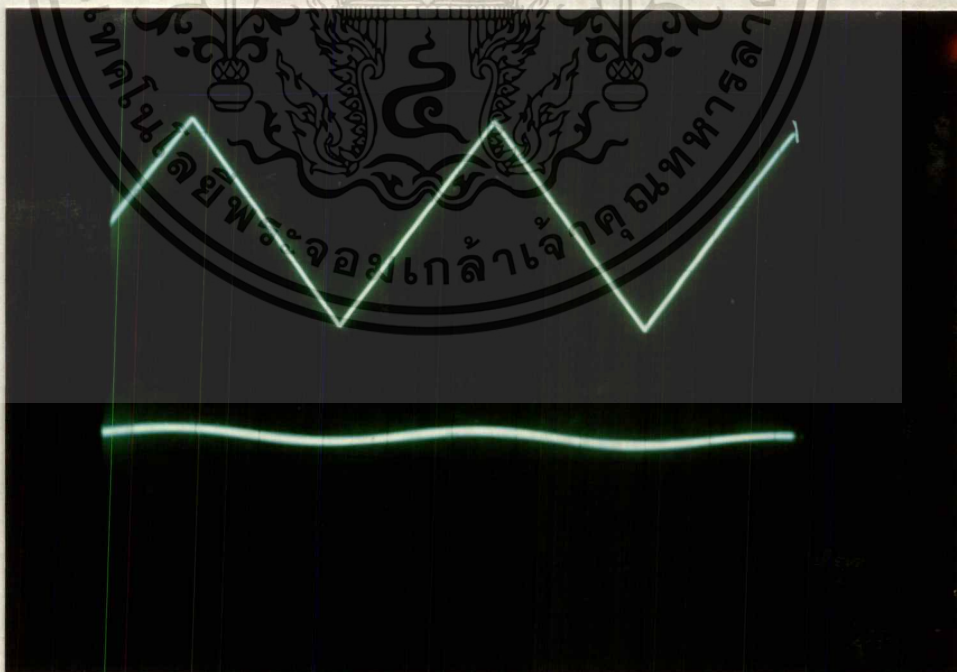


รูปที่ 4.10 เมื่อ input เป็น Saw tooth ที่มีความถี่ 2K Hz ผ่าน Low pass filter ที่มีความถี่ cut off 2K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

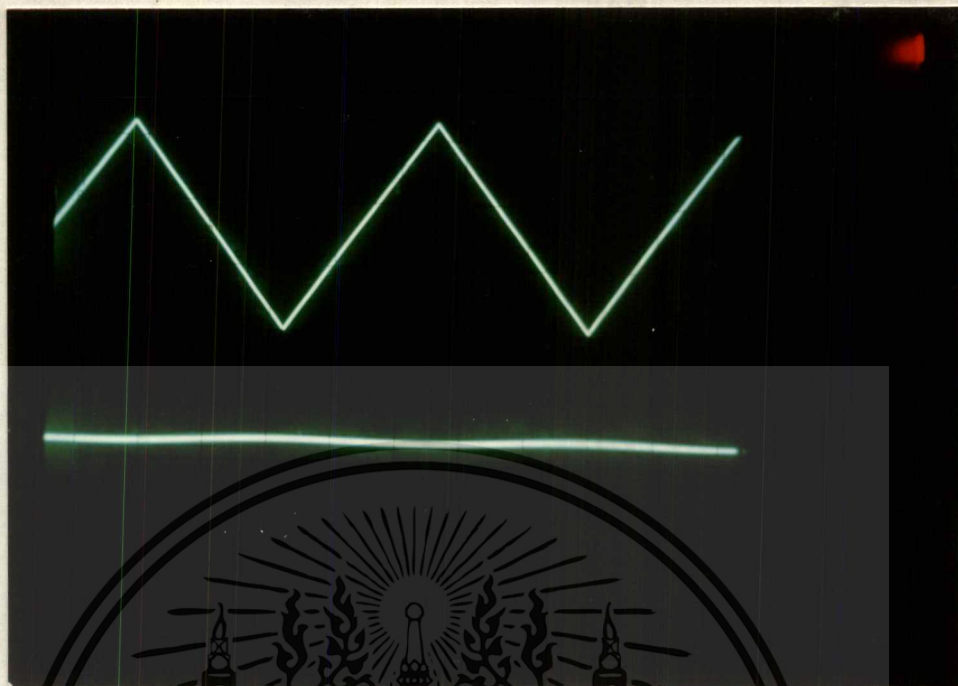


รูปที่ 4.11 เมื่อ input เป็น Saw tooth ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cut off 0.9K Hz ถึง 1K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v



รูปที่ 4.12 เมื่อ input เป็น Saw tooth ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cut off 1.9K Hz ถึง 2K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 เมื่อ input เป็น Saw tooth ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cut off 2.9K Hz ถึง 3K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v



รูปที่ 4.14 เมื่อ input เป็น Saw tooth ที่มีความถี่ 1K Hz ผ่าน Band pass filter ที่มีความถี่ cut off 3.9K Hz ถึง 4K Hz โดยใช้สเกล T/D 0.2ms, Vo/D 0.2v, Vi/D 2v

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

จากการทดลองในตอน 1 เราจะเห็นกราฟคุณสมบัติของ filter ชนิดต่าง ๆ แต่กราฟที่ได้จะไม่ได้ตาม Ideal เนื่องจาก Order ของ filter ตัวนี้ยังไม่สูงพอ และข้อจำกัดของ Board DSP นี้ที่จะยอมให้ใช้ Sampling frequency สูงสุดไม่เกิน 8 KHz

ส่วนในตอน 2 กรณีที่อินพุตเป็นสัญญาณ Square Wave และ Saw tooth เราพบว่าสัญญาณ Square Wave และ Saw tooth นั้นเกิดจากความถี่ Harmonic รูป Sine Wave ที่อันดับต่างๆ มารวมกัน ดังนั้นเมื่อผ่านวงจร Filter ทำให้เอาที่พุดที่ได้นั้นออกมาเป็นรูป Sine Wave สาเหตุเนื่องมาจากความถี่ Harmonic บางค่าหายไปพร้อมกับการ Filter จึงทำให้รูปสัญญาณเอาต์พุดไม่เป็น Square Wave เหมือนกับอินพุตทั้งยังทำให้ความถี่เปลี่ยนไปเนื่องจากความถี่หายไป ความถี่ของเอาต์พุดจึงลดลงด้วย ผลการทดลองที่ได้นั้นค่อนข้างที่จะถูกต้องตามทฤษฎีทุกอย่าง แต่ทางผู้จัดทำยังขาดเครื่องมือที่ใช้ตรวจจับสัญญาณที่มีคุณภาพ จึงทำให้ผลการทดลองที่ได้ไม่ดีเท่าที่ควร จึงขออภัยมา ณ ที่นี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include<stdio.h>
#include<math.h>
#include<conio.h>
#include<dos.h>

float C[200],M[200],H[200],freq,frec1,frec2;
float freq,freq1,freq2,rate,srate;
float n,gamma,gamma1,gamma2,PI=3.1415926;
int i,sfil,legth,hex,swin;
char pm;

main()
{
    textmode(3);
    textbackground(0);
    clrscr();
    for(;;)
    {
        textcolor(1);
        textbackground(7);
        gotoxy(11,2);
        printf("    KING MONGKUT INSTUTUTE OF TECHNOLOGY NORTH BANGKOK    ");
        gotoxy(11,3);
        printf(" DEPARTMENT OF INDUSTRIAL PHYSICS AND MEDICAL INSTRUMENTATION ");
        gotoxy(16,5);
        textcolor(4);
        textbackground(11);
        printf(" CALCULATE COEFFICEINT FIR FILTER BY WINDOWS METHOD \n");
        textcolor(14);
        textbackground(0);

```

```

cprintf("RECTANGULAR WINDOW");
gotoxy(33,10);
cprintf("HANNING WINDOW");
gotoxy(33,12);
cprintf("HAMMING WINDOW");
gotoxy(33,14);
cprintf("BLACKMAN WINDOW");
gotoxy(33,16);
cprintf("QUIT PROGRAM TO DOS");
textcolor(13);
gotoxy(28,8);
cprintf(" (1)");
gotoxy(28,10);
cprintf(" (2)");
gotoxy(28,12);
cprintf(" (3)");
gotoxy(28,14);
cprintf(" (4)");
gotoxy(28,16);
cprintf(" (5)");
textcolor(12);
gotoxy(11,19);
cprintf("SELECT WINDOWS FUNCTION: ");
textcolor(14);
cscanf("%d",&swin);
clrscr();
textcolor(1);
textbackground(7);

```



เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยราชภัฏบรียรัมย์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

case 1:

```
gotoxy(28,1);  
cprintf(" RECTANGULAR WINDOW ");  
filter();  
rectan();  
disp();  
break;
```

case 2:

```
gotoxy(30,1);  
cprintf(" HANNING WINDOW ");  
filter();  
hanning();  
disp();  
break;
```

case 3:

```
gotoxy(30,1);  
cprintf(" HAMMING WINDOW ");  
filter();  
hamming();  
disp();  
break;
```

case 4:

```
gotoxy(30,1);  
cprintf(" BLACKMAN WINDOW ");  
filter();  
blackman();  
disp();  
break;
```

```

        case 5:
            textcolor(7);
            textbackground(0);

            clrscr();

            exit(0);

        break;
    }

    getch();
}
}

filter()
{
    textcolor(14);
    textbackground(0);
    gotoxy(33,4);
    cprintf("LOWPASS FILTER");
    gotoxy(33,6);
    cprintf("HIGHPASS FILTER");
    gotoxy(33,8);
    cprintf("BANDPASS FILTER");
    gotoxy(33,10);
    cprintf("STOPBAND FILTER");
    gotoxy(33,12);
    cprintf("GOTOO MAIN MENU");
    textcolor(13);
    gotoxy(28,4);

```

เอกสารนี้ `cprintf(" (1)");` สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะ `gotoxy(28,6);` อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

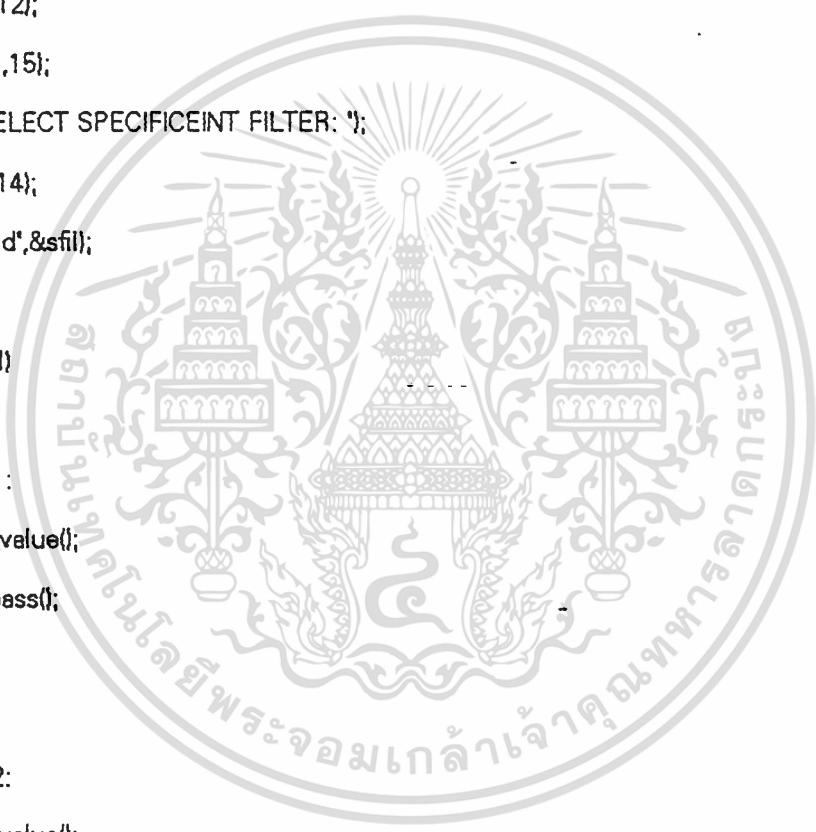
printf" (2)");
gotoxy(28,8);
printf" (3)");
gotoxy(28,10);
printf" (4)");
gotoxy(28,12);
printf" (5)");
textcolor(12);
gotoxy(11,15);
printf("SELECT SPECIFIC EINT FILTER: ");
textcolor(14);
scanf("%d",&sfil);

switch(sfil)
{
case 1:
    lh_value();
    lopass();
    break;

case 2:
    lh_value();
    hipass();
    break;

case 3:
    bs_value();
    bpass();
    break;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
case 4: ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        bs_value();
        sband();
        break;

    case 5:
        main();
        break;
}
}

lopass()
{
    M[0]=gamma;
    for(i=1;i<=(legth/2);i++)
    {
        n=i;
        M[i]=(sin(gamma*n*PI))/(n*PI);
    }
}

hipass()
{
    M[0]=1-gamma;
    for(i=1;i<=(legth/2);i++)
    {
        n=i;
        M[i]=-sin(gamma*n*PI)/(n*PI);
    }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม้ว่าใครเห็นใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
M[0]=gamma2-gamma1;
for(i=1;i<=(legth/2);i++)
{
n=i;
M[i]=((sin(gamma2*n*P))/(n*P))-((sin(gamma1*n*P))/(n*P));
}
}

```

sband()

```

{
M[0]=(1-gamma2)+gamma1;
for(i=1;i<=(legth/2);i++)
{
n=i;
M[i]=(-(sin(gamma2*n*P))/(n*P))+((sin(gamma1*n*P))/(n*P));
}
}

```

rectan()

```

{
for(i=0;i<=((legth-1)/2);i++)
{
C[i]=M[i];
}
}

```

hanning()

```

{
for(j=0;j<=((legth-1)/2);j++)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

n=i;
C[i]=M[i]*(0.5+0.5*cos((2*n*PI)/(legth-1)));
}
}

```

hamming()

```

---{ ---
for(i=0;i<=((legth-1)/2);i++)

```

```

{
n=i;
C[i]=M[i]*(0.54+0.46*cos((2*n*PI)/(legth-1)));
}
}

```

blackman()

```

{
for(i=0;i<=((legth-1)/2);i++)
{
n=i;
C[i]=M[i]*(0.42+0.5*cos((2*n*PI)/(legth-1))+0.08*cos((4*n*PI)/(legth-1)));
}
}

```

lh_value()

```

{
title();
tfi();
textbackground(0);
textcolor(11);

```

```

gotoxy(1,5);
cprintf("SAMPLING RATE FREQUENCY      :");
gotoxy(1,7);
cprintf("FREQUENCY CUTOFF (KHz):");
gotoxy(1,9);
cprintf("LENGTH  ");
textcolor(13);
gotoxy(25,5);
cprintf("(KHz)");
gotoxy(18,7);
cprintf("(KHz)");
gotoxy(7,9);
cprintf(" >>");
gotoxy(32,5);
textcolor(14);
cscanf("%f",&rate);
gotoxy(25,7);
cscanf("%f",&freq);
gotoxy(11,9);
cscanf("%d",&legth);
srate=rate*1000;
frec=freq*1000;
gamma=2*frec/srate;
}

```

```
bs_value()
```

```
{
```

```
title();
```

```
tfil();
```

```
textbackground(0);
```

```
textcolor(11);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

gotoxy(1,5);
printf("SAMPLING RATE FREQUENCY :");
gotoxy(1,7);
printf("LOWER FREQUENCY CUTOFF (KHz):");
gotoxy(1,9);
printf("UPPER FREQUENCY CUTOFF (KHz):");
gotoxy(1,11);
printf("LENGTH>>");
textcolor(13);
gotoxy(25,5);
printf("(KHz)");
gotoxy(24,7);
printf("(KHz)");
gotoxy(24,9);
printf("(KHz)");
gotoxy(7,11);
printf(" >>");
textcolor(14);
gotoxy(32,5);
scanf("%f",&rate);
gotoxy(32,7);
scanf("%f",&freq1);
gotoxy(32,9);
scanf("%f",&freq2);
gotoxy(11,11);
scanf("%d",&legth);
rate=rate*1000;
freq1=freq1*1000;
freq2=freq2*1000;

```



```

}

disp()
{
    textcolor(14);
    for(i=0;i<=((legth-1)/2);i++)
    {
        H(((legth-1)/2)-i)=C[i];
    }
    gotoxy(1,13);
    textcolor(4);
    textbackground(7);
    cprintf(" COEFFICIENT FILTER ");
    gotoxy(1,15);
    for(i=0;i<=((legth-1)/2);i++)
    {
        hex=H[i]*32768;
        printf("H[%d] = %e = H[%d] = %x\n",i,H[i],(legth-1)-i,hex);
    }
    pcoeff();
}

title()
{
    clrscr();
    textcolor(1);
    textbackground(7);

    switch(swin)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

gotoxy(28,1);
printf(" RECTANGULAR WINDOW ");
break;

case 2:
gotoxy(30,1);
printf(" HANNING WINDOW ");
break;

case 3:
gotoxy(30,1);
printf(" HAMMING WINDOW ");
break;

case 4:
gotoxy(30,1);
printf(" BLACKMAN WINDOW ");
break;
}
}

tfil()
{
textcolor(1);
textbackground(10);

switch(sfil)
{
case 1:
gotoxy(1,3);
printf(" LOWPASS FILTER ");

```



```

break;
case 2:
    gotoxy(1,3);
    printf(" HIGHPASS FILTER ");
break;
case 3:
    gotoxy(1,3);
    printf(" BANDPASS FILTER ");
break;
case 4:
    gotoxy(1,3);
    printf(" STOPBAND FILTER ");
break;
}
}
pcoeff()
{
textcolor(13);
textbackground(0);
printf("\nYOU WANT PRINT COEFFICIENT FILTER (YN) : ");
scanf("%c",&pm);
switch(pm)
{
case 'y':
    ptile();
    ptfil();
    pvalue();
    fprintf(stdpm, "\nCOEFFICIENT FILTER\n");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใด ๆ ทั้งสิ้น หากมีข้อผิดพลาดประการใด ขออภัยเป็นอย่างสูง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        hex=H[i]*32768;
        fprintf(stdpm,"H[%d] = %e = H[%d] = %x\n",i,H[i],(legth-1-i),hex);
    }
    main();
    break;

case 'n' :
    main();

default :
    main();
}
}

ptitle()
{
    switch(swin)
    {
        case 1:
            fprintf(stdpm,"RECTANGULAR WINDOW\n");
            break;

        case 2:
            fprintf(stdpm,"HANNING WINDOW\n");
            break;

        case 3:
            fprintf(stdpm,"HAMMING WINDOW\n");
            break;
    }
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น ขอสงวนสิทธิ์ในสิ่งที่ปรากฏและขอสงวนสิทธิ์ในสิ่งที่ปรากฏอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

break;

case 4:
    fprintf(stdpm, "\n\nBLACKMAN WINDOW\n\n");
    break;
}
]

```

```

ptfil()
{
switch(sfil)
{
case 1:
    fprintf(stdpm, "\nLOWPASS FILTER\n\n");
    break;
case 2:
    fprintf(stdpm, "\nHIGHPASS FILTER\n\n");
    break;
case 3:
    fprintf(stdpm, "\nBANDPASS FILTER\n\n");
    break;
case 4:
    fprintf(stdpm, "\nSTOPBAND FILTER\n\n");
    break;
}
}

```

```

pvalue()
{

```

```

if(sfil==1 || sfil==2)

```

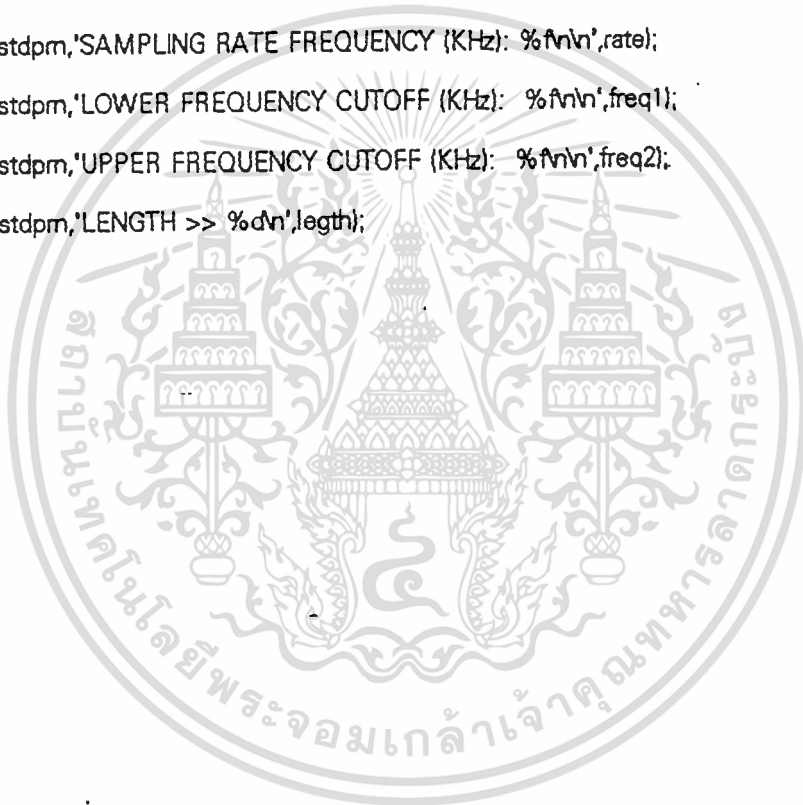
เอกสารนี้เป็นเอกสารลิขสิทธิ์ส่วนตัวไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
fprintf(stdpm,'SAMPLING RATE FREQUENCY (KHz): %f\n',rate);  
fprintf(stdpm,'FREQUENCY CUTOFF (KHz): %f\n',freq);  
fprintf(stdpm,'LENGTH >> %d\n',legth);  
}
```

```
if(sfil==3 || sfil==4)
```

```
{  
    fprintf(stdpm,'SAMPLING RATE FREQUENCY (KHz): %f\n',rate);  
    fprintf(stdpm,'LOWER FREQUENCY CUTOFF (KHz): %f\n',freq1);  
    fprintf(stdpm,'UPPER FREQUENCY CUTOFF (KHz): %f\n',freq2);  
    fprintf(stdpm,'LENGTH >> %d\n',legth);  
}
```

```
}
```



```
; DIGITAL FILTER USED TMS320C50
; DIGITAL SIGNAL PROCESSING
;ORDER 11
```

```
; FINITE IMPULSE RESPONSE (FIR)
```

```
; SELECT TYPE OF FILTER * ONE TYPE ONLY
```

```
.mmregs
```

```
;.include "lp.asm" ; LOW-PASS FILTER Fc = 1 KHZ
;.include "hp.asm" ; HIGH-PASS FILTER Fc = 1 KHZ
.include "bp.asm" ; BAND-PASS FILTER Fl = 500 Hz,Fh = 1 KHZ
;.include "sb.asm" ; STOP-BAND FILTER Fl = 500 Hz,Fh = 1 KHZ
```

```
.ds 320h
```

```
X0 .word 0000h
X1 .word 0000h
X2 .word 0000h
X3 .word 0000h
X4 .word 0000h
X5 .word 0000h
X6 .word 0000h
X7 .word 0000h
X8 .word 0000h
X9 .word 0000h
X10 .word 0000h
IMI .word 0000h
```

```
; Solutions Register
```

```
.ds 0f00h
```

```
TA .word 24 ; Fcut = 8 KHZ
RA .word 24 ; Fcut = 8 KHZ
TAP .word 31 ;
RAP .word 31 ;
TB .word 18 ; Fs = 2*Fcut
RB .word 18 ; Fs = 2*Fcut
AIC_CTR .word 90h

ACC_lo .word 0 ;
ACC_hi .word 0 ;
STAT0 .word 0 ; STAT0 storage
STAT1 .word 0 ; STAT1 storage
TEMP .word 0 ; location of TEMPorary storage
TEMP1 .word 0 ;
```

```
*****
* Set up the ISR vector *
*****
```

```
.ps 080ah
rint: B RECEIVE ;0A; Serial prot receive interrupt RINT.
xint: B TRANSMIT ;0C; Serial port transmit interrupt XINT.
```

```
*
*****
* TMS32C05X INITIALIZATION *
*****
```

```
.ps 0a00h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;-----
LDP      #TB
LACC     TB,9           ; Initialized TB and RB register
ADD      RB,2           ;
ADD      #02h          ;
CALL     AIC_2ND       ;
;-----
LDP      #AIC_CTR
LACC     AIC_CTR,2     ; Initialized control register
ADD      #03h          ;
CALL     AIC_2ND       ;
RET      ;

```

AIC_2ND:

```

LDP      #0
SACH     DXR           ;
CLRC     INTM
IDLE
ADD      #6h,15       ; 0000 0000 0000 0011 XXXX XXXX XXXX XXXX
SACH     DXR           ;
IDLE
SACL     DXR           ;
IDLE
LACL     #0           ;
SACL     DXR           ; make sure the word got sent
IDLE
SETC     INTM
RET      ;

```

.end

```

; DIGITAL FILTER USED TMS320C50
; DIGITAL SIGNAL PROCESSING
;ORDER 11

; FINITE IMPULSE RESPONSE (FIR)

; SELECT TYPE OF FILTER * ONE TYPE ONLY

.mmregs

;.include "lp.asm"      ; LOW-PASS FILTER Fc = 1 KHz
;.include "hp.asm"      ; HIGH-PASS FILTER Fc = 1 KHz
.include "bp.asm"       ; BAND-PASS FILTER Fl = 500 Hz, Fh = 1 KHz
;.include "sb.asm"      ; STOP-BAND FILTER Fl = 500 Hz, Fh = 1 KHz

```

```

.ds      320h
X0       .word    0000h
X1       .word    0000h
X2       .word    0000h
X3       .word    0000h
X4       .word    0000h
X5       .word    0000h
X6       .word    0000h
X7       .word    0000h
X8       .word    0000h
X9       .word    0000h
X10      .word    0000h
IMI      .word    0000h           ; Solutions Register

```

```

.ds      0f00h

```

```

TA       .word    24           ; Fcut = 8 KHz
RA       .word    24           ; Fcut = 8 KHz
TAP      .word    31           ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าในรูปแบบใดทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา หรือทำซ้ำโดยไม่ได้รับอนุญาตของเอกสารทุกครั้งที่มีการนำไปใช้

```
.entry
B      START
```

```
START:  SETC    INTM          ; Disable interrupts
        LDP     #0           ; Set data page pointer
        OPL    #0834h,PMST
        LACC   #0
        SAMP   CWSR          ; Set software wait state to 0
        SAMP   PDWSR        ;
```

```
* Reset AIC by writing to PA2 (address >52) to DSK
```

```
        SPLK   #022h,IMR      ; Using XINT syn TX & RX
        CALL   AICINIT        ; initialize AIC and enable interrupts
```

```
*
*****
* This routine enables serial port rx interrupts & configures *
* TLC32040 for the frame sync. When RINT is triggered, read a *
* dummy data word from the AIC then generate a sine wave to  *
* send out. *
*****
```

```
        CLRC   OVM           ; OVM = 0
        SPM    0             ; PM = 0
        SPLK   #012h,IMR
        SETC   CNF
        CLRC   OVLX
        CLRC   INTM          ; enable
```

```
;
WAIT    NOP                ; wait for a transmit or receive
        NOP
        NOP
        B      WAIT
```

```
;----- end of main program -----;
```

```
;
; RECIEVER INTERRUPT SERVICE ROUTINE
```

```
;
RECEIVE:
```

```
        LDP    #6
        LAMM   DRR
        SACL   X0
        ZAP                    ; SET ACC = 0 & P REGISTER = 0
```

```
        LTD   X10          ; LD X10 IN P REGISTER
        MPY   H0           ; H0*X10
```

```
        LTD   X9          ;MOVE X9 TO X10 ; ADD IN ACC ; LD X9 IN P REGISTER
        MPY   H1
```

```
        LTD   X8
        MPY   H2
```

```
        LTD   X7
        MPY   H3
```

```
        LTD   X6
        MPY   H4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LTD    X5
MPY    H5

LTD    X4
MPY    H6

LTD    X3
MPY    H7

LTD    X2
MPY    H8

LTD    X1
MPY    H9

LTD    X0
MPY    H0-----
APAC

```

```

OO:    SACH    IMI,1
        LACC    IMI,15
        AND     #0FFFCh,15
        RPT     #14
        SFR
        SAMP    DXR
        RETE

```

```

; TRANSMIT INTERRUPT SERVICE ROUTINE
;
TRANSMIT:
    RETE

```

```

*
*****
* DESCRIPTION: This routine initializes the TLC320C40, for          *
* a 8Khz sample rate with a gain setting of 1 *
*****
* aic initialization data
*

```

```

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        ; (.5ms at 50ns)
          SACH    GREG
          ;-----
          LDP     #TA              ;
          SETC    SXM              ;
          LACC    TA,9            ; Initialized TA and RA register
          ADD     RA,2              ;
          CALL    AIC_2ND          ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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    ; (.5ms at 50ns)
SACH     GREG

```

```

-----
LDP      #TA        ;
SETC     SXM        ;
LACC     TA,9       ; Initialized TA and RA register
ADD      RA,2       ;
CALL     AIC_2ND    ;
-----

```

```

LDP      #TB        ;
LACC     TB,9       ; Initialized TB and RB register
ADD      RB,2       ;
ADD      #02h       ;
CALL     AIC_2ND    ;
-----

```

```

LDP      #AIC_CTR   ;
LACC     AIC_CTR,2  ; Initialized control register
ADD      #03h       ;
CALL     AIC_2ND    ;
RET      ;

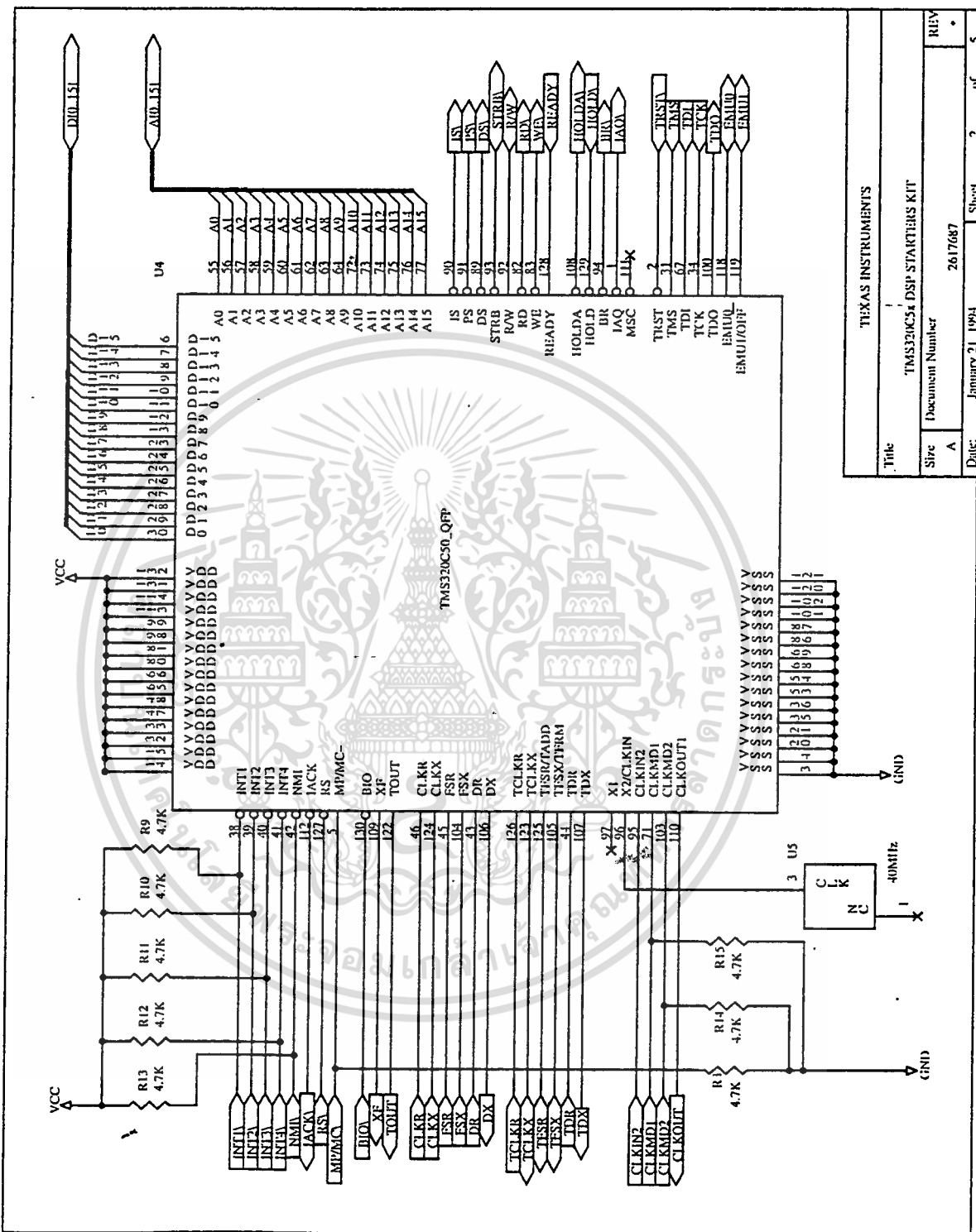
```

AIC_2ND:

```

LDP      #0
SACH     DXR        ;
CLRC     INTM
IDLE
ADD      #6h,15     ; 0000 0000 0000 0011 XXXX XXXX XXXX XXXX
SACH     DXR        ;
IDLE
SACL     DXR        ;
IDLE
LACL     #0         ;
SACL     DXR        ; make sure the word got sent
IDLE
SETC     INTM
RET      ;
.end

```

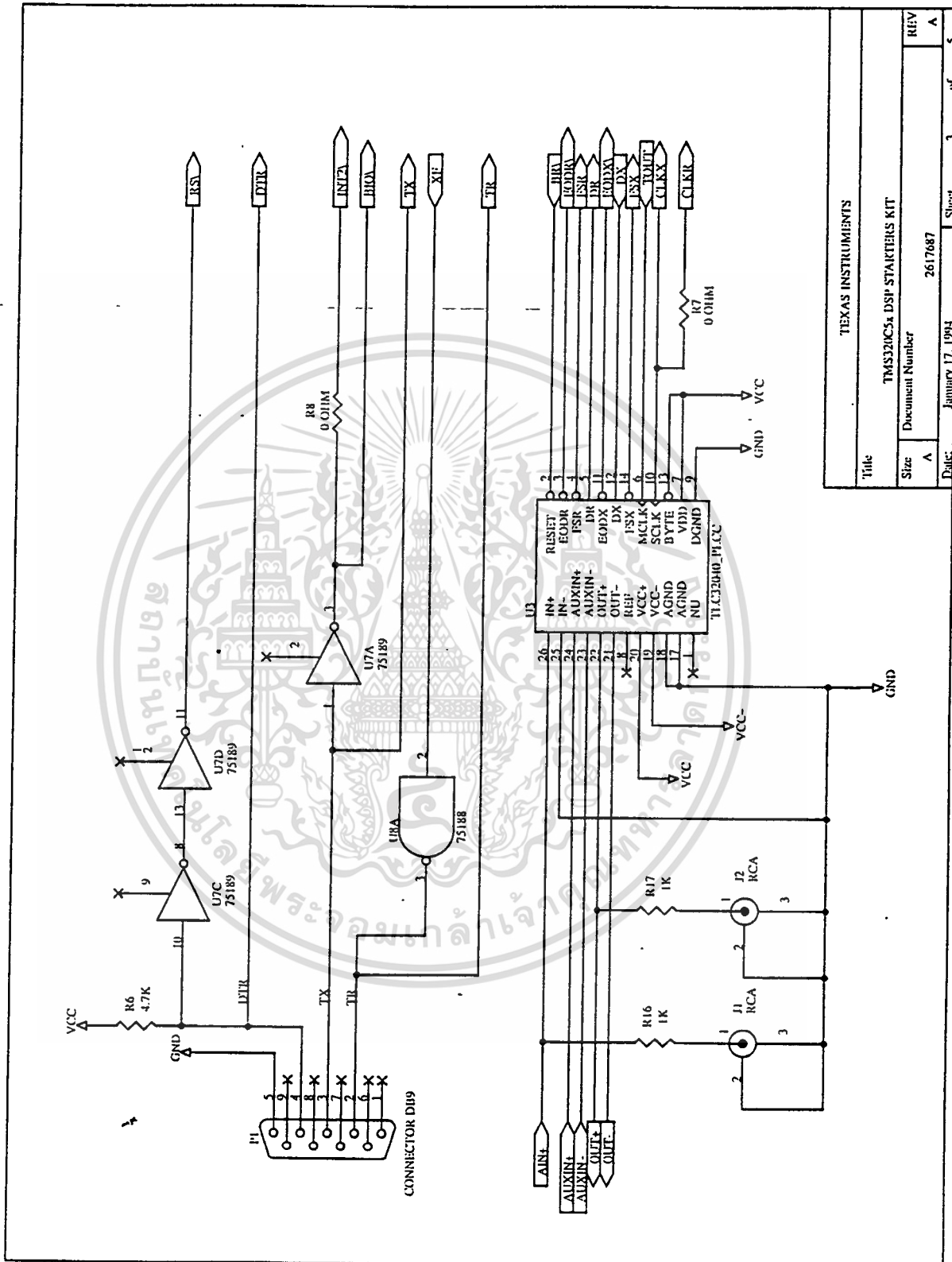


TEXAS INSTRUMENTS	
Title	TMS320C50 DSP STARTERS KIT
Size	A
Document Number	2617087
Date	January 21, 1984
Sheet	2 of 5

DSK Circuit Board Dimensions and Schematic Diagrams

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

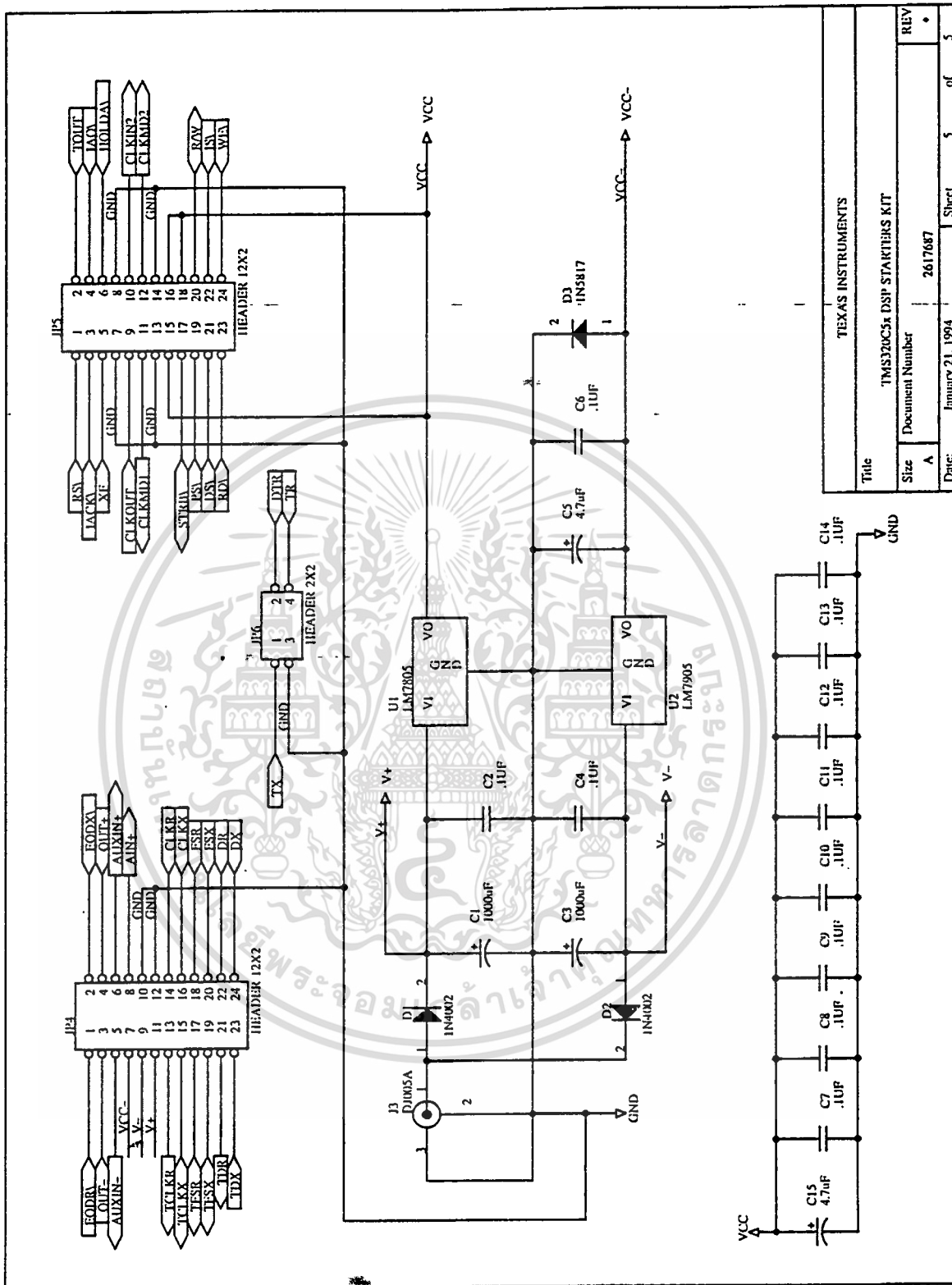
Schematic Diagrams



A-4

Title		TEXAS INSTRUMENTS	
Size		A	
Document Number		2617687	
Date	January 17, 1994	Sheet	1 of 5
REV		A	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



A-6

Title		TEXAS INSTRUMENTS	
Size		A	
Document Number		TMS320C5x DSP STARTERS KIT	
REV		•	
Date:	January 21, 1994	Sheet	5 of 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงลงไปได้ต้องขอขอบคุณ อาจารย์ที่ปรึกษา คือ อาจารย์ พุคต์ดี ชิวสุวิทย์ และอาจารย์ สักรียา ชิตวงศ์ ที่ช่วยให้คำแนะนำมาโดยตลอด ขอขอบคุณมิตา มารดา ที่ช่วยเหลือค่าใช้จ่ายต่าง ๆ และสุดท้ายต้องขอขอบคุณเพื่อน ๆ ทุกคนที่มีส่วนช่วยให้โครงการนี้สำเร็จลงได้ด้วยดี

คณะผู้จัดทำ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. Texas Instrument "user's guide TMS 320C5x digital signal process (DSP)" .
2. Louis Nashelshy Editor, "Introduction to Digital Technology"
3. David J. Defatta, Joseph G. Lucas, william S. Hodgkiss, "Digital signal processing: A system Design Approach", JOHN WILEY 7 SONS, 1998
4. ดร.ไพรัช ธัชยพงษ์, "การประมวลผลสัญญาณดิจิทัล : ตอนการออกแบบวงจรของดิจิทัล", NECTEC, 2535
5. สันติ แซ่ย่อ, สุชาดา ดีชัยยะ, สุทธินันต์ พรศิริกุล "ปริญญานิพนธ์ ปีการศึกษา 2535 สาขาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สจล. เรื่อง ดิจิตอล ซิกแนล โปรเซสซิง"



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้