สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

วงจรกรองความถี่ดิจิตอลแบบจำกัดชนิดความถี่ต่ำผ่านอันดับที่ 8, 12, 16

8, 12, 16 th ORDER FINITE IMPULSE RESPONSE LOW-PASS DIGITAL FILTER

T046486

โดย

นายปริยะ ศิริกุล

นางสาวพนาธิป ภัทรกุลวิวัฒน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....

เลขทะเบียน 46486

วัน, เดือน, ปี - 2 เม.ย. 2546

.b.....

.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรกรองความถี่ดิจิตอลแบบจำกัดชนิดความถี่ต่ำผ่านอันดับที่ 8, 12, 16

8, 12, 16 th ORDER FINITE IMPULSE RESPONSE LOW-PASS DIGITAL FILTER

โดย

นายปริยะ ศิริกุล 41014262

นางสาวพนาธิป ภัทรกุลวิวัฒน์ 41014291

อาจารย์ที่ปรึกษา

รศ.ดร. กอบชัย เดชหาญ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **วงจรกรองความถี่ดิจิตอลแบบจำกัดชนิดความถี่ต่ำผ่านอันดับที่ 8, 12, 16**

8, 12, 16 th ORDER FINITE IMPULSE RESPONSE LOW-PASS DIGITAL FILTER

ผู้จัดทำ

1. นายปริยะ ศิริกุล 41014262

2. นางสาวพนาธิป ภัทรกุลวิวัฒน์ 41014291

(____________________) อาจารย์ที่ปรึกษา

รศ.ดร. กอบชัย เดชหาญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรกรองความถี่ดิจิตอลแบบจำกัด

ชนิดความถี่ต่ำผ่านอันดับที่ 8, 12, 16

8, 12, 16 th ORDER FINITE IMPULSE RESPONSE

LOW-PASS DIGITAL FILTER

โดย นาย ปริยะ ศิริกุล 41014262

นางสาว พนาธิป ภัทรกุลวิวัฒน์ 41014291

อาจารย์ที่ปรึกษา รศ.ดร. กอบชัย เดชหาญ

บทคัดย่อ

วงจรกรองความถี่ดิจิตอลแบบจำกัด ถูกใช้ในการประยุกต์ของการประมวลผลสัญญาณดิจิตอล ได้หลายอย่าง เพราะมีคุณสมบัติสำคัญคือ ให้ความคงที่ของสัญญาณเอาท์พุต ซึ่งจะป้องกันการผิดเพี้ยนของสัญญาณเอาท์พุต

ปริญญานิพนธ์ฉบับนี้จะศึกษาและออกแบบวงจรกรองความถี่ดิจิตอลแบบจำกัดชนิดความถี่ต่ำผ่านอันดับที่ 8, 12 และ16 โดยใช้โปรแกรมในการออกแบบ

ABSTRACT

Finite impulse response digital filter (FIR filter) is used in many digital signal processing (DSP) applications because it has important quality that is stablility of output signal.It protects output signal from distortion.

This project is to study and design 8, 12, 16 th order(tap) finite impulse response low-pass digital filter by using programable.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

Abstract
บทที่ 1 บทนำ 1
บทที่ 2 ทฤษฎีและหลักการ 2
2.1 ประเภทสัญญาณดิจิตอลพื้นฐาน 2
2.1.1 ยูนิตแซมเปิล, ยูนิตสเต็ป และ ยูนิตแรมป์ฟังก์ชันน์ 2
2.2 ระบบและคุณสมบัติของระบบ 4
2.2.1 คุณสมบัติความเป็นเชิงเส้น (Linearity) 4
2.2.2 ไทม์อินแวร์เรียนท์ (Time-Invariance) 5
2.2.3 ระบบลิเนียร์ไทม์อินแวร์เรียนท์ (Linear Time-Invariant (LTI) Systems) 6
2.2.4 ความมีเสถียรภาพ (Stability) 6
2.2.5 ความเป็นเหตุกภาพ (Causality) 7
2.3 ทฤษฎีพื้นฐานของวงจรกรองความถี่แบบดิจิตอล 8
2.3.1 ความหมายของวงจรกรองความถี่แบบดิจิตอล 8
2.3.2 โครงสร้างของวงจรกรองความถี่แบบดิจิตอล 9
2.4 วงจรกรองความถี่ดิจิตอลแบบจำกัด หรือ FIR 10
2.5 วงจรกรองความถี่ดิจิตอลแบบไม่จำกัด หรือ IIR 11
2.6 การเปรียบเทียบวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขกับวงจรกรองความถี่ป้อนกลับเชิงเลข โดยเป็นการเปรียบเทียบที่ตัวกรองอันดับเดียวกัน 13
2.7 การแปลงแซด (Z-Transform) 14
2.8 การออกแบบวงจรกรองความถี่เชิงเลข 15
2.8.1 การออกแบบวงจรกรองความถี่ไม่ป้อนกลับเชิงเลข 16
2.8.2 คุณสมบัติของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลข 16
2.8.2.1(a) วงจรกรองแบบหน่วงเวลาคงที่ (Constant-Delay filters) 16
2.8.2.1(b) ชนิดของ FIR ฟิลเตอร์ (Type of FIR filters) 21
2.8.2.2 ผลตอบสนองความถี่ (Frequency Response) 24
2.8.2.3 ตำแหน่งของซีโร่ (Location of zeros) 26
2.8.3 การออกแบบโดยใช้อนุกรมฟูริเยอร์ 27
2.8.4 การใช้ฟังก์ชั่นวินโดว์ (Use of window functions) 28
2.9 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย 33
2.9.1 ทฤษฎีเลขคณิตกระจาย 34
2.9.2 การนำโครงสร้างเลขคณิตกระจายมาใช้กับวงจรกรองสัญญาณเชิงเลข 38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ประโยชน์ทางการศึกษาเท่านั้น ห้ามนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

บทที่ 3 การคำนวณและการสร้าง	41
3.1 บทนำ	41
3.2 สถาปัตยกรรมภายในของ FLEX EPF 10LC84-3	41
3.3 การหาค่าผลตอบสนองอิมพัลส์หรือค่าสัมประสิทธิ์	47
3.3.1 วงจรกรองความถี่ต่ำผ่านอันดับที่ 8	47
3.3.2 วงจรกรองความถี่ต่ำผ่านอันดับที่ 12	48
3.3.3 วงจรกรองความถี่ต่ำผ่านอันดับที่ 16	49
3.4 การออกแบบวงจรกรองสัญญาณเชิงเลขโดยภาษา VHDL	50
3.4.1 ขั้นตอนในการออกแบบวงจรกรองสัญญาณเชิงเลข	50
3.4.2 การออกแบบแต่ละส่วนย่อย	51
3.4.3 การหาค่าของตารางเปิดดู	52
3.4.3.1 วงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 8	52
3.4.3.2 วงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 12	53
3.4.3.3 วงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 16	55
3.5 การออกแบบสร้างและการทำงานวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลข	57
3.5.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter)	57
3.5.2 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลข	58
3.5.2.1 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 8	58
3.5.2.2 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 12	59
3.5.2.3 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 16	60
3.5.3 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก (D/A Converter)	61
บทที่ 4 การทดลอง และผลการทดลอง	63
4.1 การทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล	63
4.1.1 ขั้นตอนการทดลอง	63
4.1.2 ผลการทดลอง	63
4.2 การทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล	64
4.2.1 ขั้นตอนการทดลอง	64
4.2.2 ผลการทดลอง	64
4.3 การทดลองการทำงานของ FPGA	65
4.3.1 ขั้นตอนการทดลอง	65
4.3.2 ผลการทดลอง	65
4.3.2.1 วงจรกรองความถี่ต่ำผ่านอันดับที่ 8	65
4.3.2.2 วงจรกรองความถี่ต่ำผ่านอันดับที่ 12	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำ... อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้... ของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

4.3.2.3 วงจรกรองความถี่ต่ำผ่านอันดับที่ 16 75
บทที่ 5 บทวิจารณ์และสรุป 80
กิตติกรรมประกาศ
บรรณานุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 (a) ยูนิตสเต็ป (b) ยูนิตอิมพัลส์ 3

รูปที่ 2.2 ฟังก์ชันยูนิตแรมพ์ 4

รูปที่ 2.3 ระบบเวลาเชิงเต็มหน่วย 4

รูปที่ 2.4 ระบบที่มีความเป็นเชิงเส้น 5

รูปที่ 2.5 คุณสมบัติของไทม์อินแวร์เรียนท์ 6

รูปที่ 2.6 คุณสมบัติของระบบลิเนียร์ไทม์อินแวร์เรียนท์ 6

รูปที่ 2.7 แสดงถึงลำดับที่มีเสถียรภาพ (stable sequence) และไม่มีเสถียรภาพ (unstable sequence) 7

รูปที่ 2.8 ความมีเหตุภาพ (causal) และ ความไม่มีเหตุภาพ (noncausal) 7

รูปที่ 2.9 บล็อกไดอะแกรมของวงจรกรองเชิงเลข 9

รูปที่ 2.10 โครงสร้างของวงจรกรองความถี่แบบดิจิตอล 10

รูปที่ 2.11 โครงสร้างวงจรกรองความถี่ดิจิตอลแบบ FIR 11

รูปที่ 2.12 โครงสร้างวงจรกรองความถี่ดิจิตอลแบบ IIR 12

รูปที่ 2.13 รูปแบบของ ROC สำหรับการแปลงแซด 15

รูปที่ 2.14 แสดงคุณสมบัติของตัวกรองหน่วงเวลา 16

รูปที่ 2.15 แมกนิจูดและแอมปลิจูดของเฟสเชิงเส้น FIR ฟิลเตอร์

(a) แมกนิจูดและเฟส (b) แอมปลิจูดและเฟส 18

รูปที่ 2.16 ผลตอบสนองอิมพัลส์สำหรับความหน่วงเฟสและกลุ่มคงที่

(a) N เป็นเลขคู่ (b) N เป็นเลขคี่ 20

รูปที่ 2.17 ผลตอบสนองอิมพัลส์สำหรับความหน่วงกลุ่มคงที่

(a) N เป็นเลขคู่ (b) N เป็นเลขคี่ 21

รูปที่ 2.18 ตัวอย่างผลตอบสนองอิมพัลส์ทั้ง 4 ชนิด ของเฟสเชิงเส้น FIR ฟิลเตอร์ 23

รูปที่ 2.19 ซีโร่และโพลแต่ละแบบของวงจรกรองความถี่ไม่ป้อนกลับแบบหน่วงเวลา 27

รูปที่ 2.20 สเปกตรัมของวินโดว์สี่เหลี่ยม 30

รูปที่ 2.21 สเปกตรัมของวินโดว์ฮานนและแฮมมิ่ง 32

รูปที่ 2.22 แสดงการคูณแบบเลขส่วนเติมเต็มสองโดยใช้บูทอัลกอริธึม 36

รูปที่ 2.23 แสดงโครงสร้างวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 4 40

รูปที่ 3.1 แสดงโครงสร้างของ FPGA ตระกูล FLEX 10K 42

รูปที่ 3.2 แสดงโครงสร้างภายในของ LE 42

รูปที่ 3.3 การใช้งาน LUT เป็นโครงข่ายของลอจิก 43

รูปที่ 3.4 แสดงโครงข่ายของการเชื่อมต่อ 43

รูปที่ 3.5 แสดงโครงสร้างภายในของ LAB 44

รูปที่ 3.6 แสดงโครงสร้างภายใน EAB 45

เอกสารนี้...เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด...เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 3.7 แสดงโครงสร้างภายในของ IOE 45

รูปที่ 3.8 แสดงขั้นตอนการออกแบบวงจรกรองสัญญาณเชิงเลข 46

รูปที่ 3.9 ไทม์มิ่งไดอะแกรมของสัญญาณควบคุมทั้งหมด 52

รูปที่ 3.10 บล็อกไดอะแกรมของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลข 57

รูปที่ 3.11 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล 57

รูปที่ 3.12 โครงสร้างของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 8 58

รูปที่ 3.13 โครงสร้างของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 12 60

รูปที่ 3.14 โครงสร้างของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 16 61

รูปที่ 3.15 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก 61

รูปที่ 3.16 วงจรรวมกับ FPGA ที่ลงโปรแกรมแล้ว 62

รูปที่ 4.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล 63

รูปที่ 4.2 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก 64

รูปที่ 4.3 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 5 กิโลเฮิร์ต 65

รูปที่ 4.4 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 20 กิโลเฮิร์ต 66

รูปที่ 4.5 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 40 กิโลเฮิร์ต 66

รูปที่ 4.6 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 60 กิโลเฮิร์ต 67

รูปที่ 4.7 กราฟแสดงผลตอบสนองทางความถี่จากผล MATLAB 68

รูปที่ 4.8 กราฟแสดงผลตอบสนองทางความถี่จากผลการทดลอง 68

รูปที่ 4.9 กราฟแสดงผลตอบสนองทางเฟสจากผล MATLAB 69

รูปที่ 4.10 กราฟแสดงผลตอบสนองทางเฟสจากผลการทดลอง 69

รูปที่ 4.11 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 5 กิโลเฮิร์ต 70

รูปที่ 4.12 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 20 กิโลเฮิร์ต 70

รูปที่ 4.13 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 40 กิโลเฮิร์ต 71

รูปที่ 4.14 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 60 กิโลเฮิร์ต 71

รูปที่ 4.15 กราฟแสดงผลตอบสนองทางความถี่จากผล MATLAB 73

รูปที่ 4.16 กราฟแสดงผลตอบสนองทางความถี่จากผลการทดลอง 73

รูปที่ 4.17 กราฟแสดงผลตอบสนองทางเฟสจากผล MATLAB 74

รูปที่ 4.18 กราฟแสดงผลตอบสนองทางเฟสจากผลการทดลอง 74

รูปที่ 4.19 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 5 กิโลเฮิร์ต 75

รูปที่ 4.20 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 20 กิโลเฮิร์ต 75

รูปที่ 4.21 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 40 กิโลเฮิร์ต 76

รูปที่ 4.22 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 60 กิโลเฮิร์ต 76

รูปที่ 4.23 กราฟแสดงผลตอบสนองทางความถี่จากผล MATLAB 78

หน้า

รูปที่ 4.24 กราฟแสดงผลตอบสนองทางความถี่จากผลการทดลอง 78
รูปที่ 4.25 กราฟแสดงผลตอบสนองทางเฟสจากผล MATLAB 79
รูปที่ 4.26 กราฟแสดงผลตอบสนองทางเฟสจากผลการทดลอง 79

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงการเปรียบเทียบระหว่าง ตัวกรองป้อนกลับเชิงเลขกับตัวกรองไม่ป้อนกลับเชิงเลข 13
ตารางที่ 2.2 แสดงคุณสมบัติของ $A(\omega)$ สำหรับเฟสเชิงเส้น 24
ตารางที่ 2.3 แสดงผลตอบสนองความถี่ของวงจรกรองความถี่ไม่ป้อนกลับแบบหน่วงเวลา 26
ตารางที่ 2.4 สรุปคุณสมบัติของวินโดว์ 33
ตารางที่ 2.5 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง 36
ตารางที่ 2.6 ค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุต 38
ตารางที่ 3.1 แสดงค่าสัมประสิทธิ์ของวงจรกรองความถี่ต่ำผ่านอันดับที่ 8 47
ตารางที่ 3.2 แสดงค่าสัมประสิทธิ์ของวงจรกรองความถี่ต่ำผ่านอันดับที่ 12 48
ตารางที่ 3.3 แสดงค่าสัมประสิทธิ์ของวงจรกรองความถี่ต่ำผ่านอันดับที่ 16 50
ตารางที่ 3.4 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 4 บิท 52
ตารางที่ 3.5 ตารางเปิดดูของวงจรกรองความถี่ต่ำผ่านอันดับที่ 8 53
ตารางที่ 3.6 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 6 บิท 54
ตารางที่ 3.7 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 3 บิท ตัวที่ 1 54
ตารางที่ 3.8 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 3 บิท ตัวที่ 2 54
ตารางที่ 3.9 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 8 บิท 55
ตารางที่ 3.10 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 4 บิท ตัวที่ 1 56
ตารางที่ 3.11 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 4 บิท ตัวที่ 2 56
ตารางที่ 4.1 ผลการทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล 63
ตารางที่ 4.2 ผลการทดลองวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก 64
ตารางที่ 4.3 ผลการทดลองของวงจรกรองความถี่ต่ำผ่านอันดับที่ 8 67
ตารางที่ 4.4 ผลการทดลองของวงจรกรองความถี่ต่ำผ่านอันดับที่ 12 72
ตารางที่ 4.5 ผลการทดลองของวงจรกรองความถี่ต่ำผ่านอันดับที่ 16 77

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

แต่เดิมเรานิยมใช้ระบบการประมวลผลสัญญาณในรูปแบบระบบสัญญาณอนาลอก (analog signal) ซึ่งมีข้อดีคือ ราคาถูกและออกแบบง่าย แต่มีข้อเสียคือ ประสิทธิภาพต่ำและความแม่นยำในการประมวลผลน้อย ดังนั้นจึงมีการพัฒนาระบบการประมวลผลสัญญาณดิจิตอล (Digital Signal Processing : DSP) หรือเรียกว่า การประมวลผลสัญญาณเชิงเลข ซึ่งให้ความแม่นยำในการคำนวณและประสิทธิภาพดีกว่าการประมวลผลสัญญาณอนาลอก จึงมีการนำมาใช้อย่างกว้างขวางในปัจจุบัน

วงจรกรองความถี่แบบดิจิตอล (digital filter) หรือเรียกว่า ตัวกรองความถี่เชิงตัวเลข ซึ่งเป็นแขนงหนึ่งของการประมวลผลสัญญาณดิจิตอล วงจรกรองความถี่ชนิดนี้ เป็นกระบวนการที่ไปดัดแปลงสเปกตรัมของสัญญาณ ให้มีสเปกตรัมเป็นไปตามข้อกำหนดที่ต้องการ ซึ่งอาจเป็นการเพิ่มค่า หรือลดทอนค่าขนาดของสัญญาณในแถบความถี่ที่กำหนดให้ หรือทำการตัดสัญญาณในแถบความถี่ที่กำหนดให้ออกไป หรือทำการแยกสัญญาณในแถบความถี่ที่ต้องการออกมาใช้งานก็ได้ ซึ่งกระบวนการเหล่านี้มีจุดประสงค์เพื่อ

1. ลดความผิดเพี้ยนของสัญญาณเอาท์พุต (distortion)
2. กำจัดสัญญาณรบกวน (noise)
3. ต้องการแยกสัญญาณออกจากสัญญาณผสมของหลายสัญญาณที่ส่งมาพร้อมกันในสายส่งเดียวกัน
4. ใช้ในการแยกแยะในโดเมนเวลาออกเป็นองค์ประกอบของสัญญาณในโดเมนความถี่
5. ใช้ในการแปลงสัญญาณเชิงเต็มหน่วย (discrete time signal) ออกเป็นองค์ประกอบทางความถี่
6. ใช้ในการดีมอดูเลตสัญญาณ (demodulation)

และจากตามข้อกำหนดที่ต้องการต่างๆ สามารถแบ่งวงจรกรองความถี่แบบดิจิตอลได้หลายชนิด ซึ่งจะแบ่งเป็น 2 ชนิดใหญ่ๆก็คือ

1. วงจรกรองความถี่ดิจิตอลแบบจำกัดผลตอบสนองอิมพัลส์ (Finite Impulse Response Digital Filter : FIR filter) หรือเรียกว่า ตัวกรองไม่ป้อนกลับเชิงเลข
2. วงจรกรองความถี่ดิจิตอลแบบไม่จำกัดผลตอบสนองอิมพัลส์ (Infinite Impulse Response Digital Filter : IIR filter) หรือเรียกว่า ตัวกรองป้อนกลับเชิงเลข

ซึ่งจะอธิบายความหมายและคุณสมบัติของวงจรกรองความถี่ทั้งสองในบทถัดไป โดยปริญญานิพนธ์ฉบับนี้จะสนใจวงจรกรองความถี่ดิจิตอลแบบจำกัดผลตอบสนองอิมพัลส์ โดยใช้โปรแกรมแมทแล็ป (MATLAB) ช่วยในการออกแบบค่าสัมประสิทธิ์ และใช้ภาษา VHDL มาบรรยายการทำงานและออกแบบวงจรขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 ประเภทสัญญาณดิจิตอลพื้นฐาน

ในการศึกษา การประมวลสัญญาณดิจิตอลนั้น จะมีสัญญาณที่เป็นพื้นฐานที่ควรจะต้องรู้จักและทราบถึงคุณสมบัติของมันเพราะสัญญาณเหล่านี้จะถูกอ้างถึงเสมอและใช้ในการวิเคราะห์ระบบ อันได้แก่

1) ยูนิตแซมเปิ้ล (unit sample or unit impulses)
2) ยูนิตสเต็ป (unit step)
3) ยูนิตแรมป์ (unit ramps)
4) เอ็กซ์โปเน็นเชียล (exponentials)

ความจริงแล้วในสัญญาณที่เป็นจริงนั้น จะมีความยุ่งยากมากคือจะไม่เป็นสัญญาณเดี่ยวๆสัญญาณเดียวอย่างกับสัญญาณทั้ง 4 ที่กล่าวมา จะเป็นลักษณะที่ผสมกันมาของสัญญาณหลายสัญญาณ แต่สัญญาณเหล่านี้ก็สามารถจะมองหรือพิจารณาเป็นสัญญาณที่ประกอบจากการรวมของสัญญาณพื้นฐาน หรือกล่าวอีกนัยหนึ่งก็คือ เนื่องจากโดยทั่วไประบบประมวลผลที่ใช้กันจะเป็นแบบเชิงเส้น (linear) นั่นคือหากสัญญาณหลายๆแบบเข้ามาพร้อมกัน (รวมกันมา) การตอบสนองของระบบจะเป็นผลรวมของการตอบสนองของสัญญาณแต่ละสัญญาณที่เข้ามา ดังนั้น หากเราทราบถึงการตอบสนองของระบบเชิงเส้น ต่อสัญญาณพื้นฐานต่างๆนั้น เราก็จะสามารถหาหรือทำนายได้ว่าผลการตอบสนองต่อสัญญาณรวมนั้นเป็นอย่างไร

2.1.1 ยูนิตแซมเปิล, ยูนิตสเต็ป และ ยูนิตแรมป์ฟังก์ชันน์

สัญญาณยูนิตแซมเปิ้ล และสัญญาณยูนิตสเต็ปจะมีความสำคัญมากต่อระบบ DSP เพราะใช้ในการทดสอบระบบ DSP ได้ ส่วนสัญญาณยูนิตแรมป์ก็จะมีความสำคัญรองลงมา และทั้งหมดนี้จะมีความสัมพันธ์เกี่ยวเนื่องกัน

สัญญาณฟังก์ชั่นยูนิตสเต็ปจะมีข้อกำหนดดังนี้

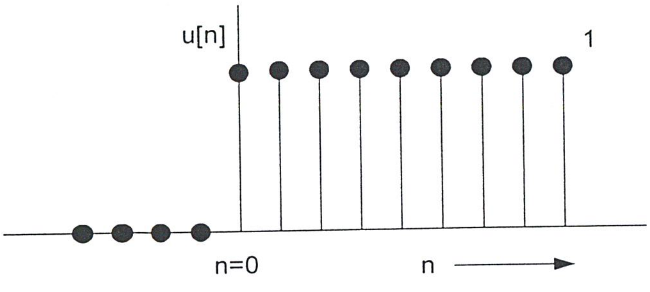
$$u[n] = 0 \qquad ; n < 0$$
$$u[n] = 1 \qquad ; n \geq 0 \tag{2.1}$$

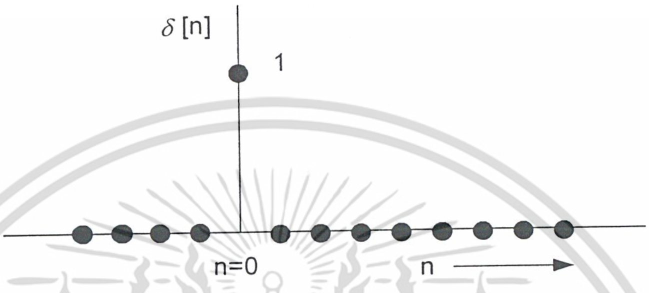
มีลักษณะแสดงไว้ในรูปที่ 2.1 (a) ใช้สัญลักษณ์ $u[n]$ มีความสำคัญในการใช้ทดสอบสัญญาณมากแต่ที่ความสำคัญมากขึ้นไปอีกก็คือ ฟังก์ชั่นยูนิตแซมเปิ้ลซึ่งมีข้อกำหนดดังนี้

$$\delta[n] = 0 \qquad ; n \neq 0$$
$$\delta[n] = 1 \qquad ; n = 0 \tag{2.2}$$

สัญลักษณ์ที่ใช้แทนคือ $\delta[n]$ จะมีค่าเป็น 1 ที่ตรง n=0 เท่านั้น นอกนั้นจะมีค่าเป็นศูนย์หมด หากเราใช้ การแปลงฟูเรียร์แปลงไปสู่โดเมนความถี่ จะทราบว่าประกอบไปด้วยความถี่ตั้งแต่ศูนย์ไปจนถึงอนันต์ทำให้มันมีประโยชน์มากในกรณีใช้หาผลตอบสนองของระบบเพื่อหาคุณสมบัติต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 2.1 (a) ยูนิตสเต็ป (b) ยูนิตอิมพัลส์

$u[n]$ และ $\delta[n]$ ต่างมีความสัมพันธ์ต่อกันและกันมาก เช่น สามารถสร้าง $u[n]$ ได้โดยการใช้ $\delta[n]$ โดยการบวกรวม $\delta[m]$ มาเรื่อยๆจากแกนทางซ้ายมือสุดคือ $m = -\infty$ จนถึง n โดยผลรวมได้ $u[n]$ ดังสมการข้างล่าง

$$u[n] = \sum_{m=-\infty}^{\infty} \delta[m]$$

และสามารถสร้าง $\delta[n]$ ได้อีกโดยการใช้ $u[n]$ ลบกับ $u[n]$ ที่ถูกเลื่อนไปทางขวาหรือหน่วงไป 1 แซมเปิ้ล ($u[n-1]$) นั่นคือ

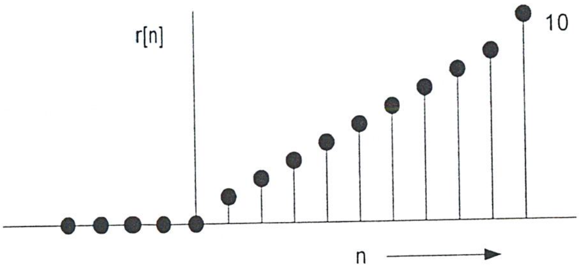
$$\delta[n] = u[n] - u[n-1]$$

ส่วนสัญญาณฟังก์ชั่นยูนิตแรมป์ จะมีลักษณะรูปโครงสัญญาณที่เป็นสัดส่วนกับค่าของ n บางครั้งจะใช้ทดสอบ สัญญาณและสร้างสัญญาณ โดยการกำหนดส่วนของ Straight line section มีข้อกำหนดดังนี้

$$r[n] = n \cdot u[n]$$

จะมีค่าเป็น 0 ที่ค่าของ $n < 0$ ดังแสดงในรูปที่ 2.2 หรืออาจสร้าง $u[n]$ ได้โดยการนำเอา $r[n]$ มาลบกันกับ $r[n-1]$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

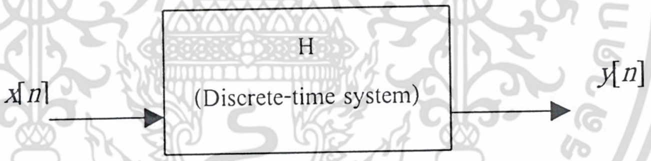


รูปที่ 2.2 ฟังก์ชันยูนิตแรมพ์

2.2 ระบบและคุณสมบัติของระบบ

ระบบเวลาเชิงเต็มหน่วย (discrete-time system) ก็คือหลักการคำนวณหรือสมการที่ใช้คำนวณหาผลที่ได้ออกมา (algorithm) ซึ่งอาจเป็นอุปกรณ์ทางฮาร์ดแวร์ก็ได้ พูดได้อีกอย่างคือ ระบบ (system) จะเป็นตัวปฏิบัติกับลำดับสัญญาณอินพุต $x[n]$ ที่เข้ามา แล้วให้กำเนิดลำดับสัญญาณเอาท์พุต $y[n]$ โดยจะมีกฎของการคำนวณในตัวเองอยู่เพื่อกำหนดผลนั้นออกมา

ดังนั้นหากกำหนดให้สัญญาณอินพุตเป็น $x[n]$ และสัญญาณเอาท์พุตที่ได้เป็น $y[n]$ และแทนสัญลักษณ์ของระบบด้วย H จะสามารถเขียนเป็นบล็อกไดอะแกรมได้



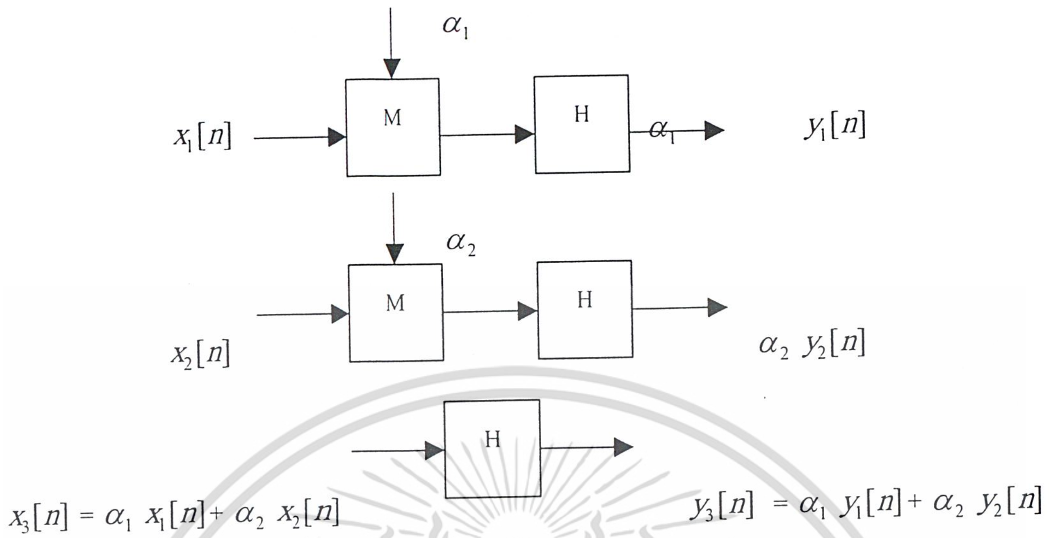
รูปที่ 2.3 ระบบเวลาเชิงเต็มหน่วย

ซึ่งในการประมวลผลสัญญาณ ระบบหรือ H ก็อาจเทียบได้ว่ามันก็คือ วงจรกรองความถี่แบบดิจิตอลนั่นเอง เช่น วงจรกรองความถี่ต่ำผ่าน (Low-pass filter) ก็คือระบบที่สมการของการตัดความถี่สูงออกอยู่ในตัวเอง มันจะให้ลำดับสัญญาณเอาท์พุต โดยเกิดจากการคำนวณของสมการของมันต่อลำดับสัญญาณอินพุต ในการคำนวณโดยทั่วไปของระบบนั้นก็จะมีข้อกำหนดขึ้นมาเพื่อความสะดวกของการคิดพิจารณาระบบ ข้อกำหนดระบบที่เป็นที่นิยมและมักจะนำมาใช้ประโยชน์มากก็คือ ระบบลิเนียร์ไทม์อินแวร์เรียนท์ ซึ่งจะได้กล่าวถึงคุณสมบัติที่สำคัญที่ควรทราบดังต่อไปนี้

2.2.1 คุณสมบัติความเป็นเชิงเส้น (Linearity)

รูปที่ 2.4 แสดงถึงสัญญาณที่เข้าสู่ระบบ จำนวนสองสัญญาณ $x_1[n], x_2[n]$ โดยหลังจากผ่านระบบแล้วก็จะมีการคูณด้วยค่าคงที่คือ α_1, α_2 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 ระบบที่มีความเป็นเชิงเส้น

สัญญาณเอาท์พุตของแต่ละตัวก็คือ $\alpha_1\ y_1[n], \alpha_2\ y_2[n]$ เมื่อรวมสัญญาณ $x_1[n]$และ $x_2[n]$ เข้าด้วยกันและกำหนดให้เป็น $x_3[n]$ป้อนสู่ระบบเดิมอีกครั้งก็จะได้สัญญาณ เอาท์พุตเป็น

$$y_3[n] = \alpha_1\ y_1[n] + \alpha_2\ y_2[n] \tag{2.3}$$

ซึ่งลักษณะของการประมวลผลที่ระบบกระทำต่อสัญญาณแต่ละตัว และที่เอาท์พุตเป็นผลรวมของผลลัพธ์ของแต่ละตัวนี้เรียกว่า "Superposition property" ซึ่งหากระบบใดมีคุณสมบัตินี้อยู่ เรียกได้ว่าระบบนั้นเป็นเชิงเส้น

2.2.2 ไทม์อินแวร์เรียนท์ (Time-Invariance)

เป็นลักษณะของการเลื่อน (shift) ข้อมูลไปข้างหน้าหรือข้างหลัง โดยมีลักษณะการเขียนแบบลำดับ (sequence) ดังนี้

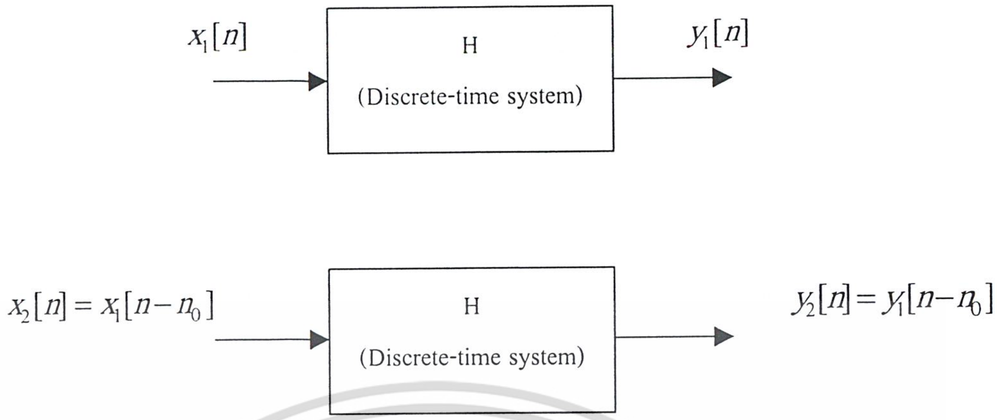
$$x_2[n] = x_1[n - n_0]$$

หมายความว่า ลำดับ $x_2[n]$ คือ $x_1[n]$ ที่ถูกเลื่อนให้ล่าช้าไป (delay) ไปขนาด n_0 แซมเปิ้ล และเมื่อลำดับนี้ป้อนเข้าสู่ระบบจะได้ผลที่เอาท์พุตเป็น

$$y_2[n] = y_1[n - n_0]$$

สำหรับทุกค่าของ n และ n_0 ระบบนี้ จัดได้ว่ามีคุณสมบัติของไทม์อินแวร์เรียนท์ หรือ แสดงได้ดังรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

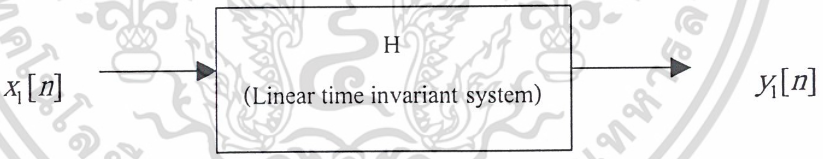


รูปที่ 2.5 คุณสมบัติของไทม์อินแวร์เรียนท์

เมื่อมีการเลื่อนของ ลำดับสัญญาณที่อินพุต ลำดับสัญญาณที่เอาท์พุตที่ได้ก็จะมีการเลื่อนไปด้วยค่าที่เท่ากันเสมอ หรือระบบไม่ขึ้นอยู่กับเวลานั่นเอง

2.2.3 ระบบลิเนียร์ไทม์อินแวร์เรียนท์ (Linear Time-Invariant (LTI) Systems)

ในการวิเคราะห์ระบบทั่วๆ ไปจะวิเคราะห์ด้วยคุณสมบัติที่ระบบจะต้องมีทั้งสองคุณสมบัติหลัก ดังที่กล่าวมานั้นคือลิเนียร์ และไทม์อินแวร์เรียนท์ สามารถเขียนบล็อกไดอะแกรมใหม่ได้เป็น

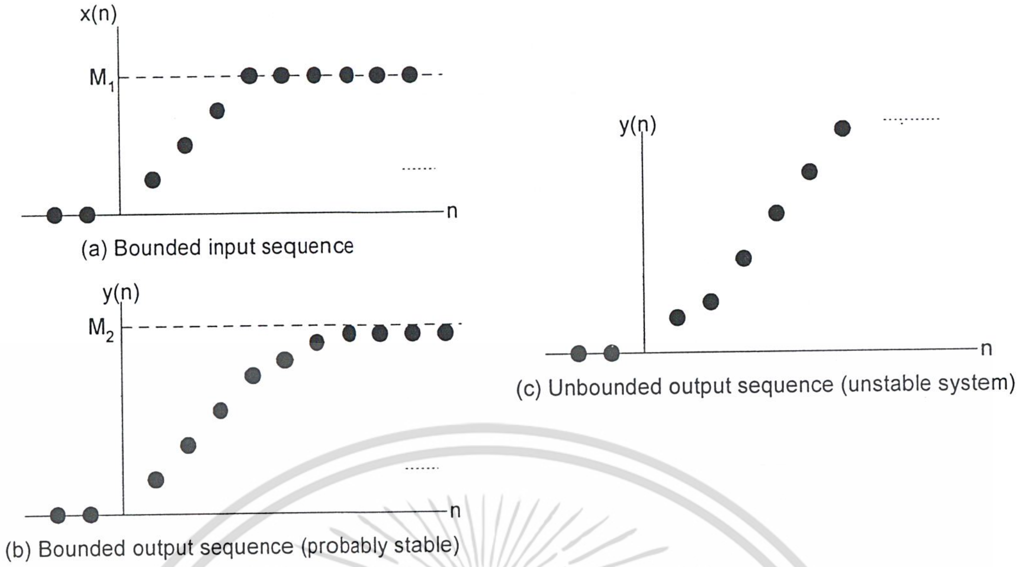


รูปที่ 2.6 คุณสมบัติของระบบลิเนียร์ไทม์อินแวร์เรียนท์

2.2.4 ความมีเสถียรภาพ (Stability)

นับได้ว่าเป็นคุณสมบัติที่สำคัญมากอย่างหนึ่งที่ระบบใดๆควรจะต้องมี เป็นตัวบอกว่าระบบนั้นๆ จะสามารถใช้งานได้ตลอดช่วงยาวนานเท่าในระบบที่เรียกได้ว่ามีคุณสมบัติของความมีเสถียรภาพ อยู่ หมายความว่าเมื่อสัญญาณอินพุต มีแอมปลิจูด ที่อยู่ในช่วงไม่เข้าสู่ ∞ (หรือเรียกว่า Bound input) ระบบนั้นจะต้องให้ค่าของลำดับ ที่เอาท์พุตไม่เข้าสู่ ∞ ด้วย หากค่าที่ได้ที่เอาท์พุตมีค่าเข้าสู่ ∞ ก็ถือได้ว่าระบบนั้น ไม่เสถียรภาพ (unstable)

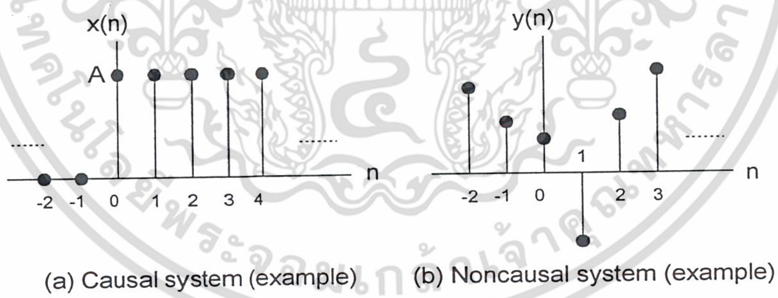
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดงถึงลำดับที่มีเสถียรภาพ (stable sequence) และไม่มีเสถียรภาพ (unstable sequence)

2.2.5 ความเป็นเหตุกภาพ (Causality)

ระบบที่ให้ความเป็นเหตุกภาพหมายถึงระบบที่ให้ลำดับที่เอาท์พุตออกมาที่จุดเริ่มต้นจาก $n = 0$ ไปจนถึง ∞ เท่านั้น (ค่าที่อยู่ต่ำกว่า $n = 0$ จะต้องมีค่าเป็นศูนย์หมด)



รูปที่ 2.8 ความมีเหตุกภาพ (causal) และ ความไม่มีเหตุกภาพ (noncausal)

ซึ่งในปกติแล้วสัญญาณทั่วไปในทางปฏิบัติจะเป็นแบบมีเหตุกภาพ เท่านั้น เอาท์พุตที่เกิดขึ้นใหม่จะมีค่าเท่ากับเอาท์พุตที่ปัจจุบันรวมกับเอาท์พุตที่ผ่านมาเท่านั้น ส่วนระบบที่เป็นไม่มีเหตุกภาพ จะให้ค่าทั้งที่ n อยู่ในช่วงลบและบวก นั้นคือเอาท์พุตที่เกิดขึ้นใหม่จะมีค่าเท่ากับเอาท์พุทที่ปัจจุบันรวมกับเอาท์พุตที่ผ่านมาและเอาท์พุตของอนาคต ซึ่งในระบบแบบนี้จะทำได้ก็ในกรณีที่ต้องทำการบันทึกลำดับสัญญาณนั้นไว้เท่านั้นนอกจากนี้ยังมีคุณสมบัติอื่นๆอีกบ้างเช่น Invertibility, memory, commutation เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ทฤษฎีพื้นฐานของวงจรกรองความถี่แบบดิจิตอล

2.3.1 ความหมายของวงจรกรองความถี่แบบดิจิตอล

วงจรกรองความถี่แบบดิจิตอล คือ กระบวนการที่ไปดัดแปลงสเปกตรัมของสัญญาณ ให้มีสเปกตรัมเป็นไปตามข้อกำหนดที่ต้องการ ซึ่งอาจเป็นการเพิ่มค่า หรือลดทอนค่าขนาดของสัญญาณในแถบความถี่ที่กำหนดให้ ซึ่งในการวิเคราะห์ และสังเคราะห์วงจรนั้น ต้องใช้เครื่องมือพื้นฐานทางคณิตศาสตร์เข้าช่วย ดังนั้นเราจึงนิยมเรียกว่า วงจรกรองความถี่เชิงเลข

การที่วงจรกรองความถี่เชิงเลขมีการนำมาประยุกต์ใช้งานกันอย่างกว้างขวางนั้น อาจมาจากข้อได้เปรียบหลายประการดังต่อไปนี้

1. ผลตอบสนองความถี่ของวงจรกรองความถี่ สามารถออกแบบให้มีความใกล้เคียงกับผลตอบสนองความถี่ที่กำหนดให้ หรือผลตอบสนองความถี่ที่ต้องการได้ นอกจากนี้การออกแบบวงจรกรองความถี่ให้มีผลตอบสนองเชิงเส้นทำได้ง่าย

2. คุณสมบัติของวงจรกรองความถี่ที่ออกแบบและสร้างแล้วจะไม่ขยับเลื่อน (drift) ไปตามสภาพแวดล้อม หรือตามอุณหภูมิ หรือตามระยะเวลาการใช้งาน นอกจากนี้ยังสามารถใช้งานในย่านความถี่ต่ำได้เป็นอย่างดี

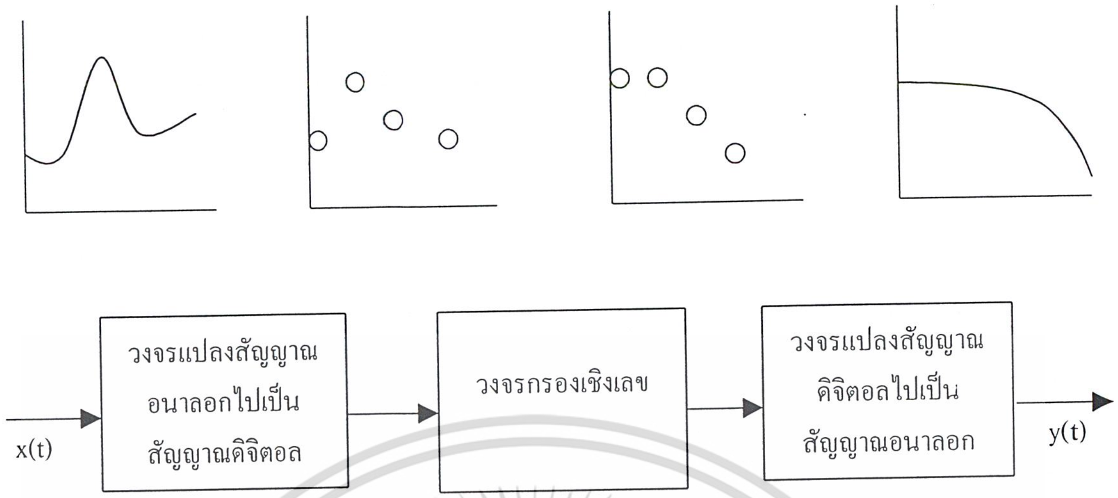
3. การประยุกต์ใช้งานเป็นวงจรกรองความถี่แบบปรับตัวได้ (adaptive filter) ทำได้ง่าย

4. ผู้ออกแบบสามารถออกแบบโดยคำนึงถึงความยาวของคำ (wordlength) ของตัวเลขฐานสองที่ต้องการใช้ และยังสามารถออกแบบให้มีผลตอบสนองความถี่ตามที่ต้องการได้

5. ในปัจจุบัน ถ้าพิจารณาในแง่มุมของเสถียรภาพของวงจรกรองความถี่ ความเชื่อถือได้ ราคา หรือขนาดของวงจรกรองความถี่เชิงเลข สิ่งเหล่านี้กำลังได้รับการพัฒนา และปรับปรุง และมีแนวโน้มว่าจะให้ผลลัพธ์ที่ดีกว่าของวงจรกรองความถี่แบบอนาลอก (analog filter) หรือเรียกว่า วงจรกรองความถี่เชิงอุปมาน

วงจรกรองเชิงเลขสามารถเขียนอธิบายในรูปของบล็อกไดอะแกรมได้ดังรูปที่ 2.9 โดยสัญญาณอินพุตซึ่งเป็นสัญญาณอนาลอกจะถูกสุ่ม (sampled) ด้วยช่วงเวลาที่ค่าคงที่ค่าหนึ่ง และสัญญาณที่ถูกสุ่มนี้จะถูกเปลี่ยนให้อยู่ในรูปเลขฐานสองโดยการแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอลหรือสัญญาณเชิงเลข (analog to digital converter)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 บล็อกไดอะแกรมของวงจรกรองเชิงเลข

หลังจากนั้นเลขฐานสองที่แทนสัญญาณอนาลอกที่เข้ามาทางอินพุทจะถูกกรองโดยวงจรกรองเชิงเลข การกรอง จะเป็นการคำนวณทางตัวเลข ซึ่งจะอาศัยวงจรที่ใช้ในระบบคอมพิวเตอร์ ได้แก่ ตัวบวก (adder) ตัวคูณ (multiplier) รีจิสเตอร์ (shift register) และ อุปกรณ์หน่วยความจำ (memory devices) ต่อมาค่าเอาท์พุทที่ได้จากวงจรกรองเชิงเลขนี้จะถูกแปลงกลับเป็นสัญญาณอนาลอกอีกทีหนึ่งเป็นสัญญาณเอาท์พุทที่นำไปใช้งานได้ วงจรกรองเชิงเลขสามารถแบ่งได้เป็น 2 ประเภท ตามลักษณะของผลตอบสนองอิมพัลส์ ได้แก่

1. วงจรกรองความถี่ดิจิตอลแบบผลตอบสนองอิมพัลส์จำกัดผลตอบสนองอิมพัลส์
2. วงจรกรองความถี่ดิจิตอลแบบผลตอบสนองอิมพัลส์ไม่จำกัดผลตอบสนองอิมพัลส์

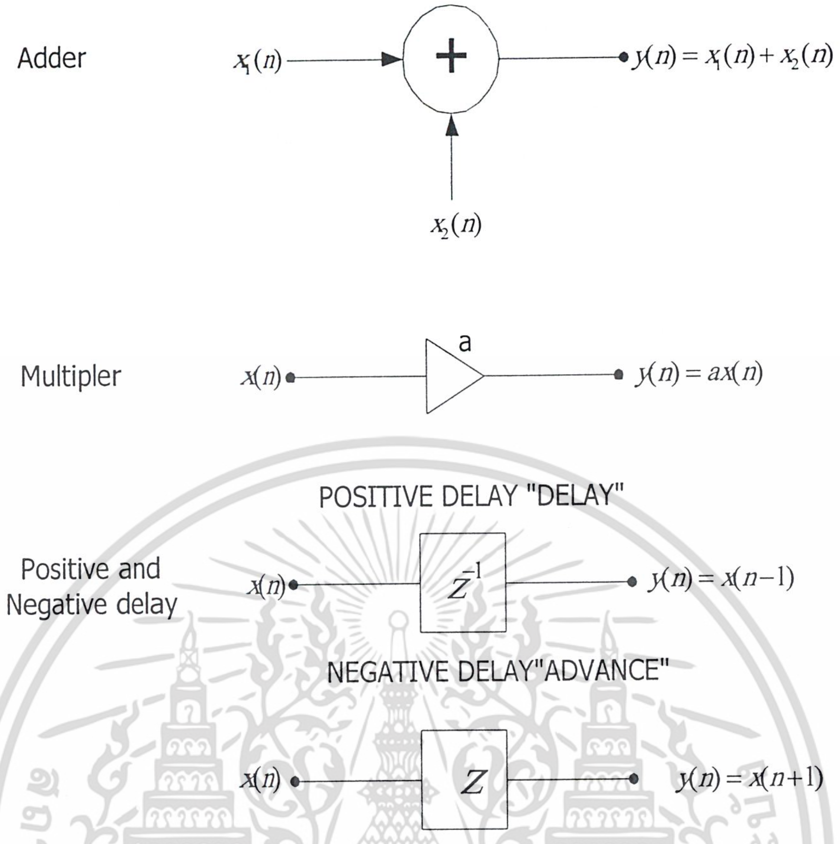
วงจรกรองความถี่ดิจิตอลแบบผลตอบสนองอิมพัลส์จำกัดผลตอบสนองอิมพัลส์ มักเป็นตัวกรองที่ไม่มีการป้อนกลับเป็นวงจรกรองที่มีโครงสร้างง่ายๆ และมีเสถียรภาพที่ดี แต่มีข้อเสียที่จะใช้วงจรกรองที่มีอันดับสูงถึงแม้จะต้องการให้มีลักษณะทางความถี่ที่ง่ายๆก็ตาม

ส่วนวงจรกรองความถี่ดิจิตอลแบบผลตอบสนองอิมพัลส์ไม่จำกัดเป็นตัวกรองที่มีการป้อนกลับเป็นวงจรกรองที่ใช้อันดับต่ำกว่าวงจรกรองแบบผลตอบสนองอิมพัลส์จำกัดที่ความต้องการลักษณะทางความถี่เหมือนกัน แต่การออกแบบจะยุ่งยากกว่า และมีปัญหาในเรื่องความมีเสถียรภาพไม่ดีนัก

2.3.2 โครงสร้างของวงจรกรองความถี่แบบดิจิตอล

วงจรกรองความถี่แบบดิจิตอล ประกอบไปด้วยส่วนที่สำคัญ 3 ส่วน คือ การบวก, การคูณ และการหน่วง (delay) ดังแสดงในรูปที่ 2.10 การบวกและการคูณจะใช้แนวความคิดมาจากตัวเลขในหน่วยคอมพิวเตอร์ ส่วนการหน่วงจะทำให้การถึงข้อมูลในอนาคตมีค่าอย่างต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 โครงสร้างของวงจรกรองความถี่แบบดิจิตอล

การหน่วงจะแบ่งเป็น 2 ส่วน คือ บวก (positive) และลบ (negative) โดยการหน่วงแบบบวกนี้เป็นอุปกรณ์ที่ทำหน้าที่บันทึกความจำของรีจีสเตอร์ (register) จะเก็บค่าได้ตามระยะเวลาที่กำหนดสำหรับการคำนวณครั้งต่อไป การหน่วงแบบบวกจะแทนค่าด้วย Z^{-1} และสามารถอธิบายความสัมพันธ์ได้ด้วยการแปลงแซด การหน่วงแบบลบ ใช้แทนค่าต่อไปในระดับสัญญาณ แทนค่าด้วย Z จะมีชนิดและการใช้งานที่เหมาะสมอย่างไรก็ตาม การใช้งานก็ไม่สามารถใช้งานได้เสมอไป

วงจรกรองความถี่แบบดิจิตอล สามารถออกแบบให้มีค่าของการคูณที่แน่นอน และมีหลักการในการพิจารณาที่ไม่ยุ่งยากซับซ้อน

2.4 วงจรกรองความถี่ดิจิตอลแบบจำกัดผลตอบสนองอิมพัลส์ หรือ FIR

คำว่า FIR ย่อมาจาก Finite Impulse Response ซึ่งแปลว่า ผลตอบสนองอิมพัลส์จำกัด นั่นคือ หากเราป้อนสัญญาณอิมพัลส์ให้กับวงจรนี้แล้ว สัญญาณตอบสนองจะมีค่าจำกัด โดยสัญญาณเอาท์พุตของระบบจะขึ้นอยู่กับสัญญาณอินพุตเท่านั้น จึงเรียกว่า วงจรกรองความถี่ไม่ป้อนกลับ (nonrecusive filter) หรือ ไม่ป้อนกลับเชิงเลข ซึ่งสามารถเขียนสมการได้ดังนี้

$$y(n) = \sum_{k=-\infty}^{\infty} b_k x(n-k) \qquad (2.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ b_k เป็นค่าคงที่ใดๆ ที่แทนค่าสัมประสิทธิ์ (coefficient) ของวงจรกรองความถี่และในทางปฏิบัติค่า k จะมีค่าคงที่ ไม่ถึงกับมีค่าอนันต์ ขึ้นกับอันดับของวงจรกรองความถี่ N ที่ต้องการใช้ จะได้

$$y(n)=\sum_{k=0}^{N-1} b_k x(n-k) \tag{2.5}$$

และจะได้สมการคอนโวลูชั่น (convolution) เป็น

$$y(n)=\sum_{m=0}^{N-1} h(m)x(n-m) \tag{2.6}$$

เปลี่ยนค่าตัวแปร จะได้

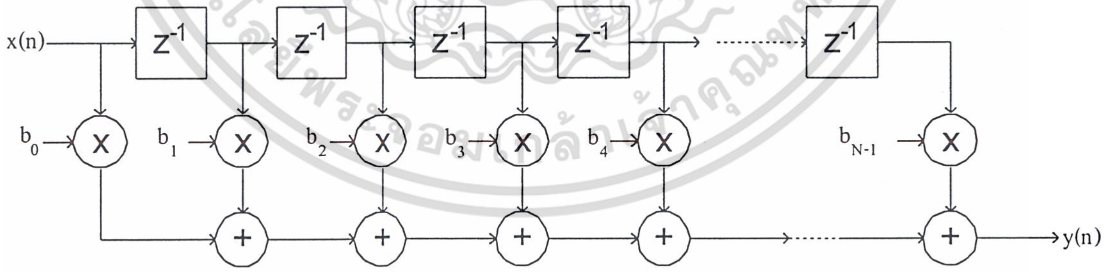
$$y(n)=\sum_{m=n}^{n-N+1} h(n-m)x(m)$$

เมื่อ $x(n)$ เป็นอินพุต และ $h(n)$ เป็นผลตอบสนองอิมพัลส์ลำดับที่ N (length-N impulse response) เมื่อนำมาประยุกต์ใช้งานกับการแปลงแซด จะได้ฟังก์ชั่นถ่ายโอน

$$H(z)=\sum_{n=0}^{N-1} h(n)z^{-n} \tag{2.7}$$

แทนค่า $z=e^{j\omega}$ จะได้ผลตอบสนองความถี่ของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลข

$$H(\omega)=\sum_{n=0}^{N-1} h(n)e^{-j\omega n} \tag{2.8}$$



รูปที่ 2.11 โครงสร้างวงจรกรองความถี่ดิจิตอลแบบ FIR

2.5 วงจรกรองความถี่ดิจิตอลแบบไม่จำกัดผลตอบสนองอิมพัลส์ หรือ IIR

คำว่า IIR ย่อมาจาก Infinite Impulse Response ซึ่งแปลว่า ผลตอบสนองอิมพัลส์ไม่จำกัด หรือถึงอนันต์ (infinite) ทั้งนี้ เพราะวงจรกรองความถี่ชนิดนี้มีคุณสมบัติประจำตัวที่สำคัญ คือ หากเราป้อนสัญญาณอิมพัลส์ให้กับวงจรนี้แล้ว สัญญาณตอบสนองจะไม่สิ้นสุด แต่จะมีไปจนถึงอนันต์ ดังนั้นอาจเรียกวงจรนี้ว่า วงจรกรองความถี่แบบป้อนกลับ (recusive filter) หรือ ป้อนกลับเชิงเลข เพราะสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอาท์พุตจะขึ้นอยู่กับค่าสัญญาณอินพุตที่ป้อนเข้ามา และสัญญาณเอาท์พุตก่อนหน้านั้นโดยทั่วไปถ้าให้สัมประสิทธิ์ของวงจรกรองความถี่เชิงเลขมีจำนวนจำกัด อาจเขียนสมการผลต่างสืบเนื่องที่ M (M[th] order differential equation) สำหรับวงจรกรองความถี่ระบบเวลาจริงได้

$$y(n) = \sum_{k=0}^{M} b_k x(n-k) - \sum_{k=1}^{N} a_k y(n-k) \qquad (2.9)$$

หาผลตอบสนองความถี่สามารถหาได้จาก สมการผลต่าง โดยจาก

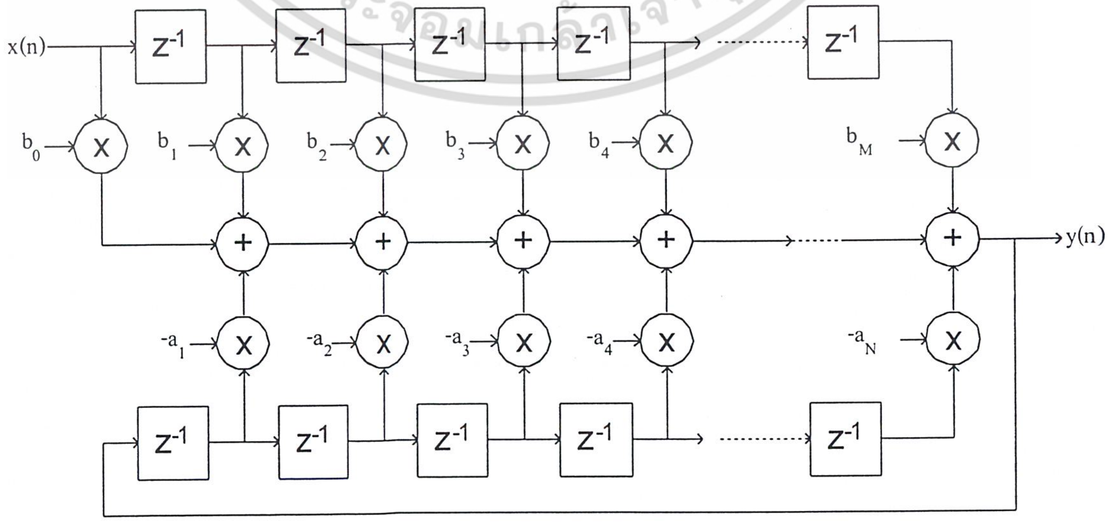
$$y(n) + \sum_{k=1}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

หรือฟังก์ชั่นถ่ายโอนเป็น

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} \qquad (2.10)$$

โดยในที่นี้ให้ $b_0 = 1$ และโดยการแทนค่าให้ $z = e^{j\omega}$ เราจะได้ผลตอบสนองความถี่ของวงจรกรองความถี่ป้อนกลับเชิงเลขแบบทั่วไป คือ

$$H(\omega) = \frac{\sum_{k=0}^{M} b_k e^{-j\omega k}}{1 + \sum_{k=1}^{N} a_k e^{-j\omega k}} \qquad (2.11)$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้าม...ทุกครั้งที่มีการนำไปใช้

รูปที่ 2.12 โครงสร้างวงจรกรองความถี่ดิจิตอลแบบ IIR

2.6 การเปรียบเทียบวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขกับวงจรกรองความถี่ป้อนกลับเชิงเลข โดยเป็นการเปรียบเทียบที่ตัวกรองอันดับเดียวกัน

ตัวกรองป้อนกลับเชิงเลข	ตัวกรองไม่ป้อนกลับเชิงเลข
1. อาจเกิดการไม่เสถียรได้	1. มีความเสถียรเสมอ
2. ให้ช่วงความถี่ที่ผลตอบสนองความถี่จากผ่านสัญญาณไปเป็นไม่ผ่านสัญญาณ หรือ ทรานซิชั่นแบนด์แคบกว่า	2. ให้ช่วงทรานซิชั่นแบนด์กว้างกว่า
3. ค่าการหน่วงสัญญาณที่จุดออกของสัญญาณ (overall delay) น้อยกว่าจึงเหมาะสำหรับใช้กับระบบเวลาจริง	3. ค่าการหน่วงออกไปของสัญญาณที่จุดสัญญาณออกมากกว่ามาก จึงไม่เหมาะกับระบบเวลาจริง
4. เกิดผลของความผิดเพี้ยนของผลตอบสนองเฟส (phase distortion) สูง	4. สามารถออกแบบให้มีผลตอบสนองเฟสเป็นแบบ เชิงเส้นได้
5. ผลของสัญญาณรบกวนเนื่องมาจากการปัดเศษ (round off error) มีค่ามาก	5. สัญญาณรบกวนจากการปัดเศษมีค่าน้อย
6. การออกแบบง่ายโดยเฉพาะการออกแบบโดยใช้วิธีการแปลงมาจากตัวกรองเชิงอุปมาณ	6. การออกแบบให้มีผลตอบสนองแอมพลิจูดคมทำได้ยากกว่า

ตารางที่ 2.1 แสดงการเปรียบเทียบระหว่าง ตัวกรองป้อนกลับเชิงเลขกับตัวกรองไม่ป้อนกลับเชิงเลข

ในข้อ 1 นั้น เนื่องจากตัวกรอง ไม่ป้อนกลับเป็นระบบที่มีเฉพาะซีโร่ หรืออาจเรียกว่าเป็น (all zeroes filter) ดังนั้นจึงเสถียรเสมอ ผิดกับตัวกรองป้อนกลับที่เป็นระบบที่มีทั้งโพล (pole) และซีโร่ (zero) เมื่อนำตัวกรองไปสร้างใช้งาน โดยการแทนค่าสัมประสิทธ์ของตัวกรองด้วยเลขฐานสองที่มีจำนวนบิตจำกัด ทำให้ค่าสัมประสิทธ์คลาดเคลื่อนไป ตำแหน่งของโพลอาจเลื่อนไปในตำแหน่งที่ทำให้ตัวกรองไม่เสถียรได้

ส่วนในข้อ 2 นั้น เนื่องจากตัวกรองป้อนกลับทำการประเมินค่าผลตอบสนองความถี่ในอุดมคติโดยใช้อัตราส่วนของพหุนามสองพหุนาม หรือแบบตรรกยะ ส่วนตัวกรองแบบไม่ป้อนกลับทำการประเมินค่าโดยใช้พหุนามธรรมดา และโดยทั่วไปการดูว่าตัวกรองเป็นตัวกรองอันดับเท่าใด เราจะดูที่กำลังสูงสุดของพหุนาม ตัวอย่างเช่น ตัวกรองป้อนกลับอันดับ 2 ตัวกรองประมาณค่าโดยใช้พหุนามอันดับ 2 สองพหุนามมาหารกัน ส่วนตัวกรองไม่ป้อนกลับ ใช้พหุนามอันดับ 2 อย่างเดียว ดังนั้นเมื่อเปรียบเทียบตัวกรองในอันดับเดียวกันตัวกรองแบบป้อนกลับจึงทำการประมาณค่าได้แม่นยำกว่า หรือได้ผลตอบสนองแอมปลิจูดได้คมกว่า นั่นก็คือ ช่วงความถี่ แถบเปลี่ยนสถานะ หรือทรานซิชั่นแบนด์ (transition band) หรือ ช่วงความถี่ที่ผลตอบสนองแอมปลิจูดเปลี่ยนจากผ่านสัญญาณเป็นไม่ผ่านสัญญาณ ของตัวกรองป้อนกลับจึงแคบกว่า

เอกสารนี้...ไม่ว่ากรณีใด...ทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนข้อ 3 เป็นผลโดยตรงมาจากข้อ 2 สำหรับตัวกรองที่ให้ผลตอบสนองแอมปลิจูดอย่างเดียว เราสามารถใช้ตัวกรองป้อนกลับที่มีอันดับต่ำกว่า เมื่อใช้อันดับน้อยกว่าโดยทั่วไปในการสร้างตัวกลาง ก็ต้องการอุปกรณ์หน่วงสัญญาณจำนวนน้อยกว่าด้วย ข้อเปรียบเทียบข้อ 4 ตัวกรองป้อนกลับให้ผลตอบสนองเฟสที่ไม่เป็นเชิงเส้น การทำให้ตัวกรองแบบนี้มีผลตอบสนองเฟสเชิงเส้น อาจทำได้โดยในตอนประมวลผล เราทำการประมวลผล 2 ครั้ง ครั้งแรกทำการประมวลผลข้อมูลตามปกติ ส่วนครั้งที่ 2 ทำการประมวลผลในทิศทางย้อนกลับ ซึ่งการทำอย่างนี้ไม่ใช่เป็นการประมวลผลในระบบเวลาจริง

2.7 การแปลงแซด (Z-Transform)

การแปลงแซดเป็นเครื่องมือพื้นฐานทางคณิตศาสตร์ที่ใช้ในการวิเคราะห์ และสังเคราะห์วงจรกรองความถี่แบบดิจิตอล ซึ่งจะใช้ในการอธิบายฟังก์ชั่นระบบ (system function) ที่มีความสัมพันธ์กับผลตอบสนองของวงจรต่อสัญญาณหนึ่งหน่วย (unit sample response) ในระนาบเชิงซ้อนของแซด (complex Z-plane) จะแสดงให้เห็นถึงตำแหน่งของโพล และซีโร่ ของฟังก์ชั่นระบบ และจะมีวงกลมซึ่งมีจุดศูนย์กลางอยู่ที่จุดเริ่มต้น โดยมีรัศมีเท่ากับหนึ่งหน่วย ซึ่งเรียกว่า วงกลมหนึ่งหน่วย (unit cycle)

จากนิยามของการแปลงแซด

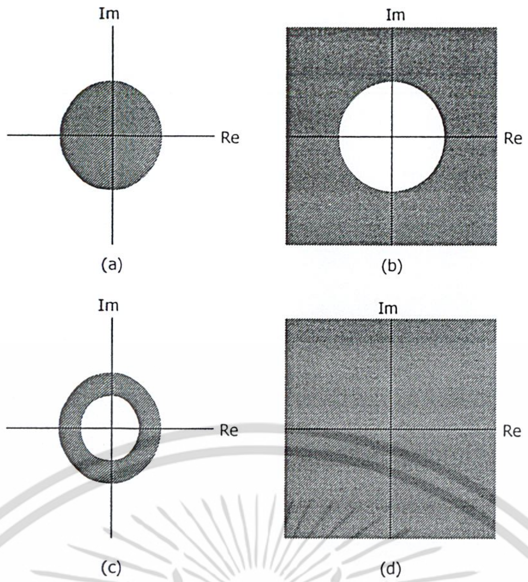
$$X(z) = \sum_{m=-\infty}^{\infty} x(mT)\, z^{-m} \qquad (2.12)$$

เมื่อ z เป็นจำนวนเชิงซ้อน และโดยทั่วไปจะให้ T = 1 เนื่องจากดรรชนี m ของการแปลงแซดในสมการ (2.12) แปรค่าจาก $-\infty$ ถึง ∞ ดังนั้น การแปลงแซดนี้จึงมีชื่อเรียกอีกอย่างว่า ผลการแปลงแซดสองข้าง (two-side Z-transform) แต่ในทางปฏิบัติเรามักจะมีลำดับ $x(m)$ โดยที่ m มีค่ามากกว่า 0 เพราะฉะนั้นจึงเรียกว่า ผลการแปลงแซดข้างเดียว (one-side Z-transform)

$$X(z) = \sum_{m=0}^{\infty} x(m) z^{-m} \qquad (2.13)$$

โดยปกติเมื่อ $x(m)$ เป็นลำดับทั่วไป ผลการแปลงแซดอาจเป็นอนุกรมที่ไม่ลู่เข้า (converge) ต่อทุกค่าของแซดเซตหรือขอบเขตของ z ที่ทำให้ค่าอนุกรมลู่เข้า เราจะเรียกว่า ขอบเขตการลู่เข้า (Region of Convergence) หรือที่เรียกว่า ROC ซึ่งจะปรากฏอยู่ในระนาบแซด ดังรูปที่ 2.13 ซึ่งถูกกำหนดจากผลของการแปลงแซด โดยที่ ROC จะเป็นตัวกำหนดความสัมพันธ์ระหว่างการแปลงแซดกับคุณสมบัติของสัญญาณลำดับดีสครีต ที่อยู่ในโดเมนของเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 รูปแบบของ ROC สำหรับการแปลงแซด

ตำแหน่งของฟังก์ชั่นระบบที่ปรากฏอยู่ในระนาบแซด จะเป็นตัวกำหนดคุณสมบัติต่างๆ เช่น สามารถตรวจสอบความมีเสถียรรูประบบ ถ้าระบบที่มีเสถียรภาพ และเป็นเหตุกภาพ ตำแหน่งของโพลทุกตัวจะอยู่ในวงกลมหนึ่งหน่วย และถ้าระบบที่มีเสถียรรูปแต่ไม่เป็นเหตุกภาพ ตำแหน่งของ โพลทุกตัวจะอยู่ที่อนันต์ เป็นต้น และสัมประสิทธิ์ของวงจรกรองความถี่สามารถกำหนดได้จากตำแหน่งของ โพลและซีโร่ และยังจะได้กล่าวถึงความสัมพันธ์ระหว่างการแปลงแซด การแปลงฟูเรียร์ (fourier transform) รวมถึงการแปลงกลับแซด (inverse Z-transform) ด้วย

2.8 การออกแบบวงจรกรองความถี่เชิงเลข

การที่จะออกแบบให้ตัวกรองมีคุณสมบัติตามต้องการนั้น ในขั้นแรกเราต้องกำหนดคุณสมบัติของตัวกรองขึ้นมา การกำหนดคุณสมบัติอาจอยู่ในรูป ผลตอบสนองความถี่ ซึ่งเป็นคุณสมบัติในโดเมนความถี่ หรืออาจกำหนดในรูปผลตอบสนองอิมพัลส์ ซึ่งเป็นคุณสมบัติใน โดเมนเวลา แล้วจากผลตอบสนองของระบบที่กำหนดขึ้นนี้เราใช้เป็นข้อกำหนดในการออกแบบ ซึ่งก็มีวิธีการออกแบบตัวกรองได้หลายวิธี อย่างไรก็ตามทุกวิธีการออกแบบก็จะได้ผลลัพธ์เป็นฟังก์ชั่นถ่ายโอน และฟังก์ชั่นถายโอนนี้จะเป็นฟังก์ชั่นที่ให้คุณสมบัติการกรองตามต้องการ

ดังนั้นสำหรับการออกแบบสร้างตัวกรองเชิงเลข โดยให้มีคุณสมบัติ ผลตอบสนองความถี่ตามอุดมคติ (ideal response) หรือมีผลตอบสนองตามความถี่ตามต้องการนั้น อาจกล่าวโดยทั่วไปได้ว่า ผู้ออกแบบและสร้างต้องนำองค์ประกอบสำคัญต่างๆมาร่วมพิจารณาด้วย ซึ่งเขียนสรุปได้ดังนี้

1. กำหนดคุณสมบัติของตัวกรองในโดเมนความถี่ เช่น ขนาด หรือค่าความหน่วงกลุ่ม หรือว่าเป็นการกำหนดคุณสมบัติของตัวกรองในโดเมนเวลา เช่น ผลตอบสนองอิมพัลส์

2. สำหรับตัวกรองป้อนกลับเชิงเลข ที่เราเขียนฟังก์ชั่นถ่ายโอนของมันเป็นอัตราส่วนของพหุนาม (polynomial) สองพหุนาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อได้ฟังก์ชั่นถ่ายโอนแล้ว เราต้องทำการเลือกโครงสร้างตัวกรองให้เหมาะสม เพื่อที่ว่าเมื่อนำไปสร้างใช้งานโดยใช้ตัวเลขความยาวจำกัดแล้ว ไม่เกิดปรากฏการณ์ไม่เป็นเชิงเส้นขึ้น หรือถ้าหลีกเลี่ยงไม่ได้ก็ให้เกิดปรากฏการณ์ไม่เป็นเชิงเส้นขึ้นน้อยที่สุด

2.8.1 การออกแบบวงจรกรองความถี่ไม่ป้อนกลับเชิงเลข

ในการออกแบบตัวกรองเชิงอุปมาน หรือ ตัวกรองเชิงเลขโดยทั่วไป เรามักให้ความสำคัญแก่ผลตอบสนองแอมปลิจูดมากกว่า นั่นคือ ออกแบบให้มีผลตอบสนองแอมปลิจูดตามต้องการ โดยยอมให้ผลตอบสนองเฟสเป็นอย่างไรก็ได้ แต่ในงานบางลักษณะนั้น เช่น การประมวลผลสัญญาณภาพ เป็นต้น ผลตอบสนองเฟสมีความสำคัญมาก หรือ ในการประมวลผลสัญญาณอีซีจี (ECG : electrocardiogram) ที่ขนาดและลักษณะรูปคลื่นของสัญญาณ และระยะเวลาในการเกิดสัญญาณมีความสำคัญมาก ในสถานการณ์แบบนี้ ถ้าใช้ตัวกรองที่มีผลตอบสนองเฟสไม่เป็นเชิงเส้นมาทำการประมวลสัญญาณ ก็จะทำให้ค่าขนาด ลักษณะรูปคลื่น หรือเวลาในการเกิดสัญญาณนี้ผิดเพี้ยนไปได้ ในการออกแบบวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขสามารถทำได้ 3 วิธี

1.ออกแบบ โดยใช้อนุกรมฟูริเยอร์ (Fourier series)

2.ออกแบบ โดยใช้การแปลงดีสครีตฟูเรียร์ (Discrete Fourier Transform)

3.ออกแบบ โดยใช้ Optimization techniques

2.8.2 คุณสมบัติของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลข

2.8.2.1(a) วงจรกรองแบบหน่วงเวลาคงที่ (Constant-Delay filters)

เพื่อให้ความกระจ่างเกี่ยวกับ คุณสมบัติการมีผลตอบสนองเป็นเชิงเส้น เรามาทำความรู้จักกับวงจรกรองแบบหนึ่งที่มีชื่อเรียกว่า วงจรหน่วงเวลา วงจรกรองแบบนี้มีคุณสมบัติดังแสดงไว้ในรูปที่ 2.14 คือ ยอมให้สัญญาณผ่านออกไปได้โดยที่ลักษณะของลูกคลื่นสัญญาณออกไปไม่ผิดเพี้ยน เพียงแต่เวลาของการเกิดรูปคลื่นจะมีเวลาหน่วงออกไป τ วินาที โดยที่ τ เป็นค่าเวลาคงตัวที่กำหนดให้ ถ้าให้ $v_2(t)$ เป็นสัญญาณออกจากวงจรกรอง และ $v_1(t)$ เป็นสัญญาณเข้าของวงจรกรอง เราสามารถเขียนความสัมพันธ์ของทั้งสองสัญญาณนี้ได้เป็น

$$v_2(t) = v_1(t-T)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ทำการแก้ไขเนื้อหา และต้องอ้างอิงที่มาของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.14 แสดงคุณสมบัติของตัวกรองหน่วงเวลา

สำนักหอสมุดกลาง_พระจอมเกล้าลาดกระบัง

จากฟังก์ชั่นของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขที่เป็นเหตุกภาพ

$$H(z) = \sum_{n=0}^{N-1} h(nT) z^{-n} \tag{2.14}$$

และเมื่อให้ $z = e^{j\omega}$ จะได้ผลตอบสนองความถี่เป็น

$$H(\omega) = \sum_{n=0}^{N-1} h(nT) e^{-j\omega nT} \tag{2.15}$$

ถ้าให้ $H(\omega)$ เป็นจำนวนเชิงซ้อน โดยส่วนจริง (real part) และส่วนจินตภาพ (imaginary part) ของ $H(\omega)$ จะอยู่ในรูป

$$H(\omega) = R(\omega) + jI(\omega) \tag{2.16}$$

ค่าแมกนิจูด (magnitude) และเฟส (phase) จะได้

$$M(\omega) = |H(\omega)| = \sqrt{R^2 + I^2}$$

$$d(\omega) = \arctan(\frac{1}{R})$$

ฉะนั้น

$$H(\omega) = M(\omega) e^{jd(\omega)} \tag{2.17}$$

ปัญหาทางคณิตศาสตร์เกิดขึ้นเนื่องจาก $M(\omega)$ ไม่สามารถวิเคราะห์ (analytic) และ $d(\omega)$ ไม่ต่อเนื่อง ทางแก้ปัญหา คือ การสร้างฟังก์ชั่นแอมปลิจูดค่าจริง $A(\omega)$ ซึ่งอาจจะเป็นค่าบวก หรือค่าลบ จะได้ผลตอบสนองความถี่

$$H(\omega) = A(\omega) e^{j\theta(\omega)} \tag{2.18}$$

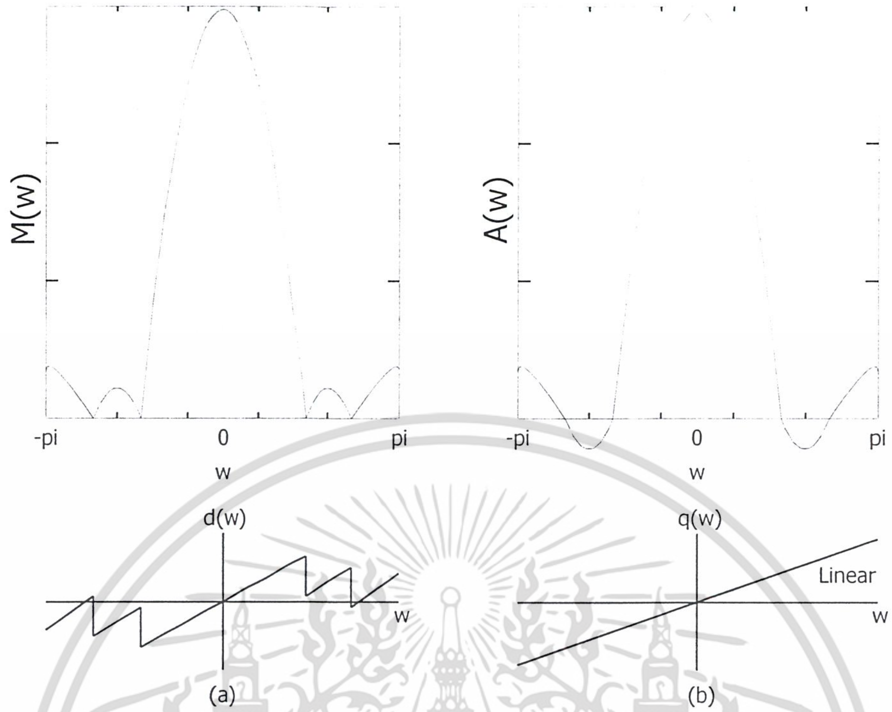
เมื่อ $A(\omega)$ คือ แอมปลิจูด ซึ่งมาจากแมกนิจูด $M(\omega)$ และ $\theta(\omega)$ มีค่าต่อเนื่อง $A(\omega)$ เป็นค่าจริง วิเคราะห์ความสัมพันธ์กับแมกนิจูด

$$A(\omega) = \pm M(\omega)$$

จากนิยามนี้ $A(\omega)$ สามารถทำการวิเคราะห์ และ $\theta(\omega)$ สามารถทำให้ต่อเนื่องได้ ซึ่งคุณสมบัติเหล่านี้นำมาใช้งานง่ายกว่า $M(\omega)$ และ $d(\omega)$ ความสัมพันธ์ระหว่าง $A(\omega)$ กับ $M(\omega)$ และระหว่าง $\theta(\omega)$ กับ $d(\omega)$ แสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

46486



รูปที่ 2.15 แมกนิจูดและแอมปลิจูดของเฟสเชิงเส้น FIR ฟิลเตอร์

(a) แมกนิจูดและเฟส (b) แอมปลิจูดและเฟส

ดังนั้นจะได้

$$A(\omega) = |H(\omega)|$$

และ (2.19)

$$\theta(\omega) = \arg H(\omega)$$

การกล่าวถึงคุณสมบัติของวงจรกรองนั้นยังมีอีกสองคำที่มักใช้ในการบ่งถึงคุณสมบัติของผลตอบสนองเฟสด้วย คำแรกคือ ความหน่วงเฟส(phase delay) จะเขียนแทนด้วย τ_p ซึ่งนิยามให้เป็น อัตราส่วนของเฟสต่อความถี่ หรือ

$$\tau_p = -\frac{\theta(\omega)}{\omega}$$

คำที่สองคือ ความหน่วงกลุ่ม (group หรือ signal หรือ envelope delay) จะเขียนแทนด้วย τ_g ซึ่งนิยามให้เป็นค่า อนุพันธ์ของเฟสเมื่อเทียบกับความถี่ หรือ

$$\tau_g = -\frac{d\theta(\omega)}{d\omega}$$

ความหน่วงเฟสคงที่ (constant phase delay) เทียบเท่ากับความหน่วงกลุ่มคงที่ (constant group delay) ซึ่งผลตอบสนองเฟสต้องเป็นเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\theta(\omega) = -\tau\omega$$

และจากสมการ (2.15) และ (2.19)

$$\theta(\omega) = -\tau\omega = \tan^{-1} \frac{-\sum_{n=0}^{N-1} h(nT)\sin\omega nT}{\sum_{n=0}^{N-1} h(nT)\cos\omega nT}$$

$$\tan\omega\tau = \frac{\sum_{n=0}^{N-1} h(nT)\sin\omega nT}{\sum_{n=0}^{N-1} h(nT)\cos\omega nT}$$

และจาก

$$\sum_{n=0}^{N-1} h(nT)\left(\cos\omega nT\sin\omega\tau - \sin\omega nT\cos\omega\tau\right) = 0$$

หรือ

$$\sum_{n=0}^{N-1} h(nT)\sin\left(\omega\tau - \omega nT\right) = 0$$

แก้สมการออกมาได้ดังนี้

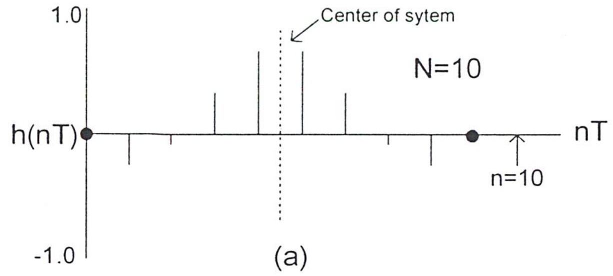
$$\tau = \frac{(N-1)T}{2} \tag{2.20}$$

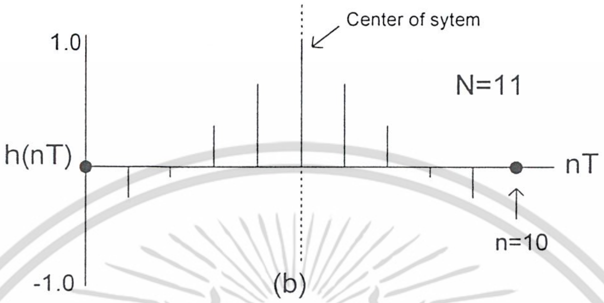
$$h(nT) = h[(N-1-n)T] \qquad for \qquad 0 \le n \le N-1 \tag{2.21}$$

เพราะฉะนั้นวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขจะไม่เหมือนกับวงจรกรองความถี่ป้อนกลับเชิงเลข ซึ่งมีความหน่วงเฟสและกลุ่มคงที่ครอบคลุมเบสแบนด์ทั้งหมด มีผลเพียงเพื่อให้ผลตอบสนองอิมพัลส์นั้นสมมาตรที่จุดกึ่งกลางระหว่างแซมเปิ้ล $(N-2)/2$ และ $N/2$ สำหรับ N เป็นเลขคู่ หรือที่ $(N-1)/2$

สำหรับ N เป็นเลขคี่ สมมาตรที่ต้องการแสดงดังรูปที่ 2.16 เมื่อ N=10 และ 11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 2.16 ผลตอบสนองอิมพัลส์สำหรับความหน่วงเฟสและกลุ่มคงที่
(a) N เป็นเลขคู่ (b) N เป็นเลขคี่

ในหลายๆการประยุกต์อาจจะให้คุณสมบัติเพียงความหน่วงเฟส หรือความหน่วงกลุ่มเพียงอย่างเดียวก็ได้ กรณีที่ต้องการเพียงความหน่วงกลุ่ม จะให้ผลตอบสนองเฟสเป็น

$$\theta(\omega) = \theta_0 - \tau\omega$$

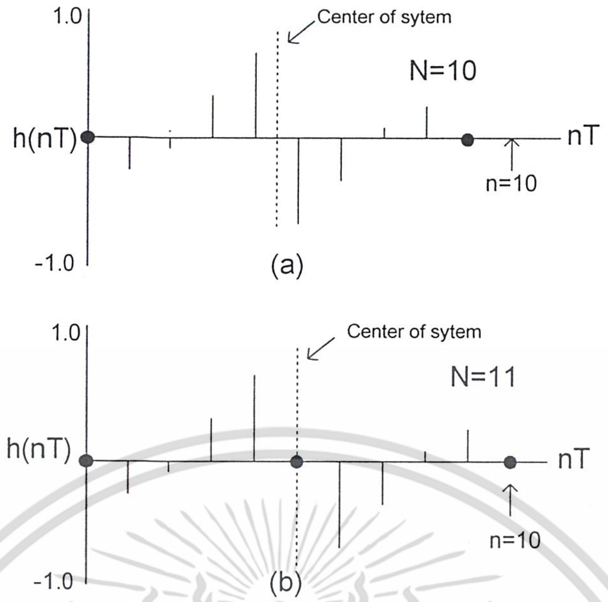
เมื่อ θ_0 เป็นค่าคงที่ ในที่นี้ให้ $\theta_0 = \pm\pi/2$ จะได้

$$\tau = \frac{(N-1)T}{2} \tag{2.22}$$

$$h(nT) = -h[(N-1-n)T] \tag{2.23}$$

ในกรณีนี้ผลตอบสนองอิมพัลส์นั้นจะปฏิสมมาตรที่จุดกึ่งกลางระหว่างแซมเปิ้ล $(N-2)/2$ และ $N/2$ สำหรับ N เป็นเลขคู่ หรือที่ $(N-1)/2$ สำหรับ N เป็นเลขคี่ แสดงดังรูปที่ 2.17 เมื่อ N=10 และ 11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ผลตอบสนองอิมพัลส์สำหรับความหน่วงกลุ่มคงที่
(a) N เป็นเลขคู่ (b) N เป็นเลขคี่

2.8.2.1(b) ชนิดของ FIR ฟิลเตอร์ (Type of FIR filters)

นอกจากนี้ยังมีการพัฒนาคุณลักษณะ และคุณสมบัติของเฟสเชิงเส้นฟิลเตอร์ ทำให้เราได้รูปแบบเชิงเส้นทั่วไปสำหรับฟังก์ชั่นเฟส (phase function) ในที่นี้พิจารณาเมื่อผ่านการนอร์มอลไลซ์ (normalize) โดยการให้คาบเวลาในการแซมปลิ้ง (T) เท่ากับ 1

$$\theta(\omega) = K_1 + K_2\omega$$

พิจารณาจากสมการ (2.15) เมื่อทำการนอร์มอลไลซ์แล้ว จะได้สมการ (2.8) ออกมา ซึ่งก็คือ ฟังก์ชั่นของผลตอบสนองความถี่ของ FIR ฟิลเตอร์ อันดับที่ N

$$H(\omega) = \sum_{n=0}^{N-1} h(n)e^{-j\omega n}$$

$$H(\omega) = e^{-j\omega M}\sum_{n=0}^{N-1} h(n)e^{j\omega(M-n)} \qquad (2.24)$$

และ

$$H(\omega) = e^{-j\omega M}[h_0 e^{j\omega M} + h_1 e^{j\omega(M-1)} + \cdots + h_{N-1}e^{j\omega(M-N+1)}] \qquad (2.25)$$

จากสมการ (2.25) สามารถเขียนอยู่ในรูปแบบ

$$H(\omega) = A(\omega)e^{j(K_1+K_2\omega)} \qquad (2.26)$$

ถ้า M (ไม่จำเป็นต้องเป็นจำนวนเต็ม) ถูกกำหนดให้

$$M = \frac{N-1}{2}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือเท่ากับ

$$M = N - M - 1$$

แล้วสมการ (2.25) จะกลายเป็น

$$\begin{aligned} H(\omega) = e^{-j\omega M}\{ & (h_0 + h_{N-1})\cos(\omega M) + j(h_0 - h_{N-1})\sin(\omega M) \\ & + (h_1 + h_{N-2})\cos(\omega(M-1)) \\ & + j(h_1 - h_{N-2})\sin(\omega(M-1)) + \cdots\} \end{aligned} \tag{2.27}$$

เมื่อเรานำสมการ (2.27) มาแทนในสมการ (2.26) แล้ว $A(\omega)$ เป็นค่าจริง คือ $K_1 = 0$ หรือ $K_1 = \pi/2$ กรณีแรกที่ได้ คือ การสมมาตรคู่ หรือ สมมาตร (even sysmetric or sysmetrical) ซึ่งตรงกับสมการ (2.21)

$$h(n) = h(N - n - 1) \tag{2.28}$$

ซึ่งจะได้

$$H(\omega) = A(\omega)e^{-jM\omega}$$

เมื่อ $A(\omega)$ เป็นฟังก์ชั่นค่าจริงของ ω และ $e^{-jM\omega}$ และเมื่อ N เป็นเลขคี่ (odd)

$$A(\omega) = \sum_{n=0}^{M-1} 2h(n)\cos(\omega(M-n)) + h(M) \tag{2.29}$$

เปลี่ยนค่าตัวแปร จะได้

$$A(\omega) = \sum_{n=1}^{M} 2h(M-n)\cos(\omega n) + h(M) \tag{2.30}$$

เมื่อ N เป็นเลขคู่ (even)

$$A(\omega) = \sum_{n=0}^{\frac{N}{2}-1} 2h(n)\cos(\omega(M-n)) \tag{2.31}$$

เปลี่ยนค่าตัวแปร จะได้

$$A(\omega) = \sum_{n=1}^{\frac{N}{2}} 2h(\frac{N}{2} - n)\cos(\omega(n - \frac{1}{2})) \tag{2.32}$$

เมื่อ $K_1 = \pi/2$ ในสมการ (2.26) แล้วรูปแบบของ $h(n)$ ของการสมมาตรคี่ หรือ ปฏิสมมาตร (odd sysmetric or antisysmetrical) ซึ่งจะตรงกับสมการ (2.23)

$$h(n) = -h(N - n - 1) \tag{2.33}$$

สำหรับ N เป็นเลขคี่ แล้ว $H(\omega)$ จะได้เป็น

$$H(\omega) = jA(\omega)e^{-jM\omega}$$

เมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$A(\omega) = \sum_{n=0}^{M-1} 2h(n)\sin(\omega(M-n)) \qquad (2.34a)$$

สำหรับ N เป็นเลขคู่

$$A(\omega) = \sum_{n=0}^{\frac{N}{2}-1} 2h(n)\sin(\omega(M-n)) \qquad (2.34b)$$

สรุปชนิดของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลข ได้ดังนี้

ชนิดที่ 1 ผลตอบสนองอิมพัลส์ที่มี อันดับคี่ และเป็นการสมมาตรคู่ จะอยู่ที่จุดกึ่งกลาง $n = M = (N-1)/2$ ซึ่งต้องการ $h(n) = h(N-n-1)$ และได้สมการ (2.29) และ (2.30)

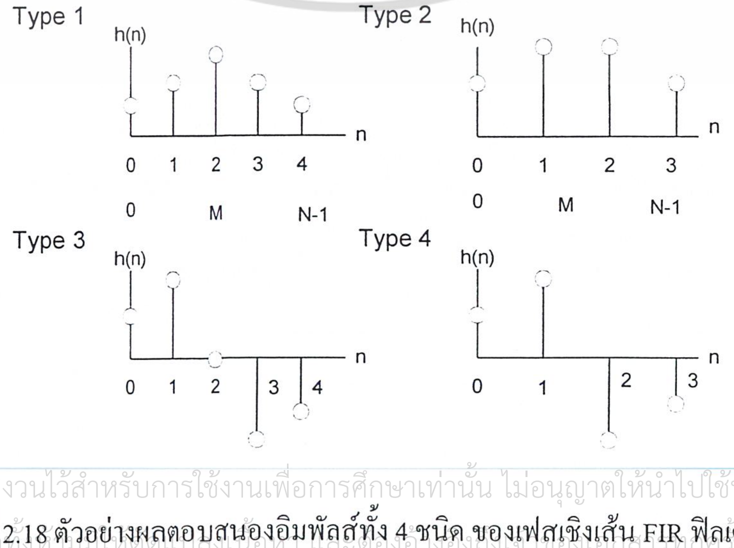
ชนิดที่ 2 ผลตอบสนองอิมพัลส์ที่มี อันดับคู่ และเป็นการสมมาตรคู่ จะอยู่ที่ M แต่ M ไม่เป็นจำนวนเต็ม เพราะฉะนั้นจะไม่มี $h(n)$ ที่จุดสมมาตร จะได้สมการ (2.31) และ (2.32)

ชนิดที่ 3 ผลตอบสนองอิมพัลส์ที่มี อันดับคี่ และเป็นการสมมาตรคี่ จากสมการ (2.33) ให้ผลคูณส่วนจินตภาพสำหรับเฟสเชิงเส้นอยู่ในรูปสมการ (2.34a)

ชนิดที่ 4 ผลตอบสนองอิมพัลส์ที่มี อันดับคู่ และเป็นการสมมาตรคี่ อยู่ในรูปสมการ (2.33) และ (2.34b)

คุณลักษณะที่แสดงออกมาของแต่ละชนิดมีความสำคัญในการออกแบบฟิลเตอร์ เช่น ชนิดที่ 3 และ 4 ที่ $A(0) = 0$ สำหรับทุกสัมประสิทธิ์ $h(n)$ ซึ่งไม่นิยมใช้สำหรับวงจรกรองความถี่ต่ำ ชนิดที่ 2 และ 3 ค่า $A(\pi) = 0$ เสมอ ซึ่งไม่นิยมใช้กับวงจรกรองความถี่สูง เป็นต้น

ระหว่างวงจรกรองความถี่สมมาตรคู่ และสมมาตรคี่ วงจรกรองความถี่ที่มีประโยชน์ใช้งานมาก คือ วงจรกรองความถี่สมมาตรคู่ เพราะค่าของฟังก์ชั่นถ่ายโอนมีเฉพาะค่าจริง ทำให้ผลตอบสนองเฟส อาจเป็นค่าคงที่ หรือค่าศูนย์ หรือเรียกว่า เป็นวงจรกรองความถี่ที่มีผลตอบสนองเฟสที่แปรเป็นเชิงเส้นกับความถี่ได้ นั่นคือ ไม่เกิดความผิดพลาดในทางด้านเฟสของสัญญาณ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 2.18 ตัวอย่างผลตอบสนองอิมพัลส์ทั้ง 4 ชนิด ของเฟสเชิงเส้น FIR ฟิลเตอร์

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้มีการดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Type 1. Odd length, even symmetric $h(n)$	
$A(\omega)$ is even about $\omega = 0$	$A(\omega) = A(-\omega)$
$A(\omega)$ is even about $\omega = \pi$	$A(\pi+\omega) = A(\pi-\omega)$
$A(\omega)$ is periodic with period 2π	$A(\pi+2\omega) = A(\omega)$
Type 2. Even length, even symmetric $h(n)$	
$A(\omega)$ is even about $\omega = 0$	$A(\omega) = A(-\omega)$
$A(\omega)$ is odd about $\omega = \pi$	$A(\pi+\omega) = -A(\pi-\omega)$
$A(\omega)$ is periodic with period 4π	$A(\pi+4\omega) = A(\omega)$
Type 3. Odd length, odd symmetric $h(n)$	
$A(\omega)$ is odd about $\omega = 0$	$A(\omega) = -A(-\omega)$
$A(\omega)$ is odd about $\omega = \pi$	$A(\pi+\omega) = -A(\pi-\omega)$
$A(\omega)$ is periodic with period 2π	$A(\pi+2\omega) = A(\omega)$
Type 4. Even length, odd symmetric $h(n)$	
$A(\omega)$ is odd about $\omega = 0$	$A(\omega) = -A(-\omega)$
$A(\omega)$ is even about $\omega = \pi$	$A(\pi+\omega) = A(\pi-\omega)$
$A(\omega)$ is periodic with period 4π	$A(\pi+4\omega) = A(\omega)$

ตารางที่ 2.2 แสดงคุณสมบัติของ $A(\omega)$ สำหรับเฟสเชิงเส้น

2.8.2.2 ผลตอบสนองความถี่ (Frequency Response)

จากสมการ (2.40) และ (2.43) สำหรับผลตอบสนองอิมพัลส์ที่สมมาตร เมื่อ N เป็นเลขคี่ จะได้ผลตอบสนองความถี่

$$H(e^{j\omega T}) = \sum_{n=0}^{(N-3)/2} h(nT)e^{-j\omega nT} + h\left[\frac{(N-1)T}{2}\right]e^{-j\omega(N-1)T/2} + \sum_{n=(N+1)/2}^{N-1} h(nT)e^{-j\omega nT} \qquad (2.35)$$

จากสมการ (2.21) แทนค่า $N-1-n=m,\ m=n$ แล้วผลรวมสุดท้ายจะอยู่ในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\sum_{n=(N+1)/2}^{N-1} h(nT)e^{-j\omega nT} = \sum_{n=(N+1)/2}^{N-1} h[(N-1-n)T]e^{-j\omega nT}$$
$$= \sum_{n=0}^{(N-3)/2} h(nT)e^{-j\omega(N-1-n)T} \qquad (2.36)$$

จากสมการ (2.35) และ (2.36)

$$H(e^{j\omega T}) = e^{-j\omega(N-1)T/2}\left\{h\left[\frac{(N-1)T}{2}\right] + \sum_{n=0}^{(N-3)/2} 2h(nT)\cos\left[\omega\left(\frac{N-1}{2}-n\right)T\right]\right.$$

และให้ $(N-1)/2-n=k$ จะได้

$$H(e^{j\omega T}) = e^{-j\omega(N-1)T/2}\sum_{k=0}^{(N-1)/2} a_k \cos\omega kT$$

เมื่อ

$$a_0 = h\left[\frac{(N-1)T}{2}\right] \qquad (2.37)$$

$$a_k = 2h\left[\left(\frac{N-1}{2}-k\right)T\right] \qquad (2.38)$$

ผลตอบสนองความถี่ของตัวกรองที่ผลตอบสนองอิมพัลส์สมมาตร เมื่อ เป็นเลขคู่ จะคล้ายกับกรณีที่กล่าวมาข้างต้น และกรณีที่ผลตอบสนองอิมพัลส์ปฏิสมมาตร จะได้ผลตอบสนองความถี่ดังตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

H(nT)	N	$H(e^{j\omega T})$
Symmetrical	Odd	$e^{j\omega\frac{(N-1)T}{2}}\sum_{k=0}^{\frac{N-1}{2}} a_k \cos\omega kT$
	Even	$e^{j\omega\frac{(N-1)T}{2}}\sum_{k=0}^{\frac{N}{2}} b_k \cos[\omega(k-\frac{1}{2})T]$
Antisymmetrical	Odd	$e^{j[\omega\frac{(N-1)T}{2}-\frac{\pi}{2}]}\sum_{k=0}^{\frac{N-1}{2}} a_k \sin\omega kT$
	Even	$e^{j\omega[\frac{(N-1)T}{2}-\frac{\pi}{2}]}\sum_{k=0}^{\frac{N}{2}} b_k \sin[\omega(k-\frac{1}{2})T]$
where	$a_0 = h[\frac{(N-1)T}{2}]$	$a_k = 2h[(\frac{(N-1)}{2}-k)T]$ $b_k = 2h[(\frac{N}{2}-k)T]$

ตารางที่ 2.3 แสดงผลตอบสนองความถี่ของวงจรกรองความถี่ไม่ป้อนกลับแบบหน่วงเวลา

2.8.2.3 ตำแหน่งของซีโร่ (Location of zeros)

จากสมการ (2.14), (2.21) และ (2.23) จะได้ตำแหน่งของซีโร่ เมื่อ N เป็นเลขคี่

$$H(z)=\frac{1}{z^{(N-1)/2}}\sum_{n=0}^{(N-3)/2} h(nT)(z^{(N-1)/2-n}\pm z^{-[(N-1)/2-n]}) + \frac{1}{2}h[\frac{(N-1)T}{2}](z^0\pm z^0) \quad (2.39)$$

ในกรณีผลตอบสนองอิมพัลส์ปฏิสมมาตรเมื่อ $(N-1)/2-n=k$ จากสมการ (2.39) จะได้

$$H(z)=\frac{N(z)}{D(z)}=\frac{1}{z^{(N-1)/2}}\sum_{k=0}^{(N-1)/2}\frac{a_k}{2}(z^k\pm z^{-k})$$

แทนค่า a_0 และ a_k จากสมการ (2.37) และ (2.38) แล้วตำแหน่งซีโร่ของ $H(z)$ จะเป็นรากที่สองของ

$N(z)$ โดย $N(z)=\sum_{k=0}^{(N-1)/2}a_k(z^k\pm z^{-k})$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า z แทนด้วย z^{-1} จะได้

$$N(z^{-1}) = \sum_{k=0}^{(N-1)/2} a_k(z^{-k} \pm z^{k})$$
$$= \pm \sum_{k=0}^{(N-1)/2} a_k(z^{k} \pm z^{-k}) = \pm N(z)$$

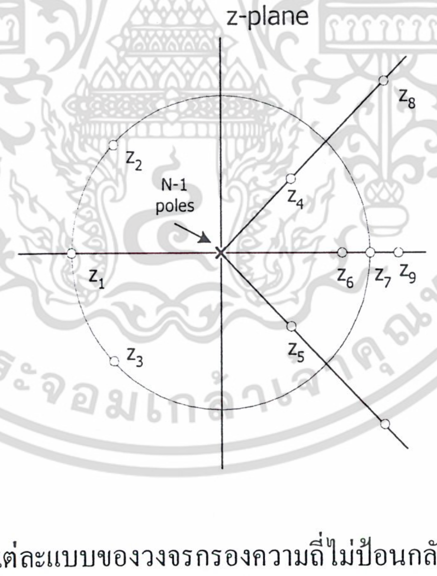
จากความสัมพันธ์คู่เหมือน เพราะฉะนั้นถ้า $z_i = r_i e^{j\theta_i}$ เป็นซีโร่ของ $H(z)$ แล้ว $z_i^{-1} = e^{-j\theta_i}/r_i$ ก็ต้องเป็นซีโร่ของ $H(z)$ ด้วย จะสรุปตำแหน่งซีโร่ได้ ดังต่อไปนี้

1. ถ้า $r_i = 1,\ \theta_i = 0$ หรือ π แล้วซีโร่จะอยู่ที่ $z_i = \pm 1$ ดังนั้น $z_i^{-1} = \pm 1$

2. ถ้า $r_i = 1,\ \theta_i \neq 0$ หรือ π แล้วซีโร่จะเป็นค่าเชิงซ้อนอยู่นอกวงกลมหนึ่งหน่วย มีอยู่ 4 แบบ คือ $z_i,\ z_i^*$ และจำนวนตัวเลขที่เป็นสัดส่วนกันและคูณกันเท่ากับ 1 (reciprocal)

3. ถ้า $r_i \neq 1,\ \theta_i = 0$ หรือ π แล้วซีโร่จะเป็นค่าจริงและเป็นจำนวนตัวเลขที่เป็นสัดส่วนกันและคูณกันเท่ากับ 1 อยู่นอกวงกลมหนึ่งหน่วย

4. ถ้า $r_i \neq 1,\ \theta_i \neq 0$ หรือ π แล้วซีโร่จะเป็นคู่คอนจูเกท (conjugate) กัน และอยู่บนวงกลมหนึ่งหน่วย



รูปที่ 2.19 ซีโร่และโพลแต่ละแบบของวงจรกรองความถี่ไม่ป้อนกลับแบบหน่วงเวลา

2.8.3 การออกแบบโดยใช้อนุกรมฟูริเยอร์

ผลตอบสนองความถี่และผลตอบสนองอิมพัลส์ของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลข หาได้โดยอาศัยการกระจายอนุกรมฟูริเยอร์

$$H(\omega) = \sum_{n=-\infty}^{\infty} h(nT)e^{-j\omega nT} \tag{2.40}$$

และ

$$h(nT) = \frac{1}{\omega_s}\int_{-\omega_s/2}^{\omega_s/2} H(e^{j\omega T})e^{j\omega nT}d\omega \tag{2.41}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และถ้า $e^{j\omega T} = z$ จากสมการ (2.40) จะได้

$$H(z) = \sum_{n=-\infty}^{\infty} h(nT) z^{-n} \tag{2.42}$$

เพราะฉะนั้นจากสมการ (2.41) ฟังก์ชั่นถ่ายโอนจะไม่เป็นเหตุกภาพและมีผลตอบสนองอิมพัลส์ไม่จำกัด จึงมีการตัดปลายอนุกรมฟูริเยอร์ออก เพื่อให้ตัวกรองมีผลตอบสนองอิมพัลส์จำกัด โดยให้

$$h(nT) = 0 \qquad for \qquad |n| > \frac{N-1}{2}$$

ในกรณีนี้ จะได้

$$H(z) = h(0) + \sum_{n=1}^{(N-1)/2} \left[h(-nT) z^{n} + h(nT) z^{-n} \right] \tag{2.43}$$

เพื่อให้เป็นตัวกรองเหตุกภาพ ซึ่งทำได้โดยการคูณ $H(z)$ ด้วย $z^{-(N-1)/2}$ จะได้

$$H'(z) = z^{-(N-1)/2} H(z) \tag{2.44}$$

การคูณด้วยพจน์นี้เข้าไป จะไม่ทำให้ผลตอบสนองแอมปลิจูดเปลี่ยนแปลงไปแต่อย่างใด แต่ทำให้ผลตอบสนองเฟสของตัวกรอง มีคุณสมบัติเป็นเชิงเส้นขึ้น จะได้

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

จากสมการ (2.44) จะเห็นได้ว่า ผลจากการตัดปลายของอนุกรมฟูริเยอร์ จะทำให้เกิดการออสซิลเลชั่น (oscillation) ในช่วงแถบผ่าน (passband) และแถบหยุด (stopband) เนื่องจาก slow convergence ในอนุกรมฟูริเยอร์ ในทางกลับกันก็เป็นสาเหตุจากความไม่ต่อเนื่องที่ขอบแถบผ่าน ซึ่งเราเรียกว่า กิบส์ออสซิลเลชั่น (Gibb's oscillation) เมื่อ N มีค่าเพิ่มขึ้น ความถี่ของการออสซิลเลชั่นก็จะเพิ่มขึ้น และทั้งความถี่ต่ำและความถี่สูง แอมปลิจูดของมันจะลดลง ขนาดของแอมปลิจูดที่ใกล้จุดไม่ต่อเนื่อง คือ จุดระหว่างริพเปิ้ล (ripple) สุดท้ายของแถบผ่าน กับริพเปิ้ลแรกของแถบหยุดจะยังคงไม่แปรเปลี่ยน อย่างไรก็ตาม ถือว่าการออสซิลเลชั่น เป็นสิ่งที่ไม่ต้องการ จึงต้องทำการลดเหตุการณ์นี้ ซึ่งมีอยู่หลายวิธี

2.8.4 การใช้ฟังก์ชั่นวินโดว์ (Use of window functions)

ฟังก์ชั่นวินโดว์เป็นเทคนิคที่นิยมและง่าย ที่ใช้สำหรับการลดผลของการออสซิลเลชั่น โดยการปรับปรุงสัมประสิทธิ์ของอนุกรมฟูริเยอร์ คือ การถ่วงน้ำหนักอนุกรมฟูริเยอร์ด้วยฟังก์ชั่นวินโดว์ที่เหมาะสมสามารถทำให้ลูกคลื่นที่เกิดบนผลตอบสนองความถี่ลดน้อยลงไปได้ โดยทั่วไปการถ่วงน้ำหนักอาจทำได้หลายวิธีซึ่งจะกล่าวรายละเอียดในหัวข้อนี้ จากเดิมเราตัดปลายอนุกรมฟูริเยอร์จำนวนอนันต์พจน์ผ่านหน้าต่างนี้ออกไป อนุกรมนี้จะถูกถ่วงน้ำหนัก หรือตัดปลายตามต้องการ

ฉะนั้นผลตอบสนองอิมพัลส์ ก็คือ ผลคูณของผลตอบสนองอิมพัลส์อุดมคติ $h_d(n)$ กับค่าฟังก์ชั่นวินโดว์ $a_w(n)$

$$h(n) = h_d(n) a_w(n) \tag{2.45}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าให้ $H_d(\omega)$ แทนผลตอบสนองความถี่อุดมคติ และผลตอบสนองนี้สามารถเขียนแทนด้วยอนุกรมฟูริเยอร์ความยาวอนันต์พจน์ โดยจากสมการ (2.40) จะได้ว่า

$$H_d(\omega) \quad = \quad \sum_{n=-\infty}^{\infty} h_d(n) e^{-jk\omega T} \tag{2.46}$$

โดยที่ $h_d(n)$ หาได้จาก

$$h_d(n) \quad = \quad (\frac{1}{2\pi}) \int_{-\pi}^{\pi} H_d(\omega) e^{jk\omega T} d\omega \tag{2.47}$$

หรือ

$$h_d(k) = (\frac{1}{\omega_s}) \int_{-\frac{\omega_s}{2}}^{\frac{\omega_s}{2}} H_d(\omega) e^{jk\omega T} d\omega \tag{2.48}$$

โดยที่ ω_s เป็นค่าความถี่ในการสุ่มตัวอย่าง ตามที่ทราบกันแล้วว่าการนำสมการ (2.46) และ (2.47) ไปใช้ในการออกแบบตัวกรองไม่ป้อนกลับเชิงเลข ซึ่งต้องตัดปลายในอนุกรมในสมการ (2.46) ให้เป็นอนุกรมจำนวนพจน์จำกัด หรือ

$$H_d = \sum_{n=0}^{\frac{N-1}{2}} h_d(n) e^{-jk\omega T} \tag{2.49}$$

ซึ่งผลลัพธ์คือทำให้เกิดลูกคลื่นบนผลตอบสนองแอมปลิจูด แต่ก็ยังไม่สามารถนำสมการ(2.48)ใช้งานได้เพราะว่าผลตอบสนองอิมพัลส์เริ่มจากลำดับที่ $n = (N-1)/2$ ซึ่งสำหรับระบบเวลาจริงไม่สามารถสร้างลำดับเวลาที่เป็นลบได้ ข้อนี้แก้ไขได้โดยเลื่อน หรือหน่วงผลตอบสนองอิมพัลส์ออกไป โดยให้เริ่มจากค่า $n = 0$ หรือก็คือ สมการ (2.15) นั่นเอง

$$H(\omega) = \sum_{n=0}^{N-1} h(n) e^{-jk\omega T}$$

ซึ่งการกระทำแบบนี้มีผลดีคือ ทำให้ตัวกรองมีผลตอบสนองเฟสเป็นเชิงเส้น

ฟังก์ชั่นวินโดว์มีอยู่หลายแบบ ได้แก่

1. วินโดว์สี่เหลี่ยม (Rectangular)
2. วินโดว์วอง ฮาน (von Hann)
3. วินโดว์แฮมมิ่ง (Hamming)
4. วินโดว์แบล็คแมน (Blackman)
5. วินโดว์ดอล์ฟ เชฟบีเชฟ (Dolph-Chebyshev)
6. วินโดว์เคเซอร์ (Kaiser)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ซึ่งในที่นี้เราจะกล่าวถึงวินโดว์ที่นิยมใช้ คือ แบบที่ 1 ถึง 4

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. วินโดว์สี่เหลี่ยม

จากสมการของฟังก์ชั่นสี่เหลี่ยม

$$a_R(nT) = \begin{cases} 1 & for \quad |n| \le \dfrac{N-1}{2} \\ 0 & otherwise \end{cases} \qquad (2.50)$$

จะได้สเปกตรัม (spectrum) ของ $a_R(nT)$ เป็น

$$A_R(e^{j\omega T}) = \sum_{n=-(N-1)/2}^{(N-1)/2} e^{-j\omega nT} = \frac{e^{j\omega(N-1)T/2} - e^{-j\omega(N+1)T/2}}{1-e^{-j\omega T}}$$

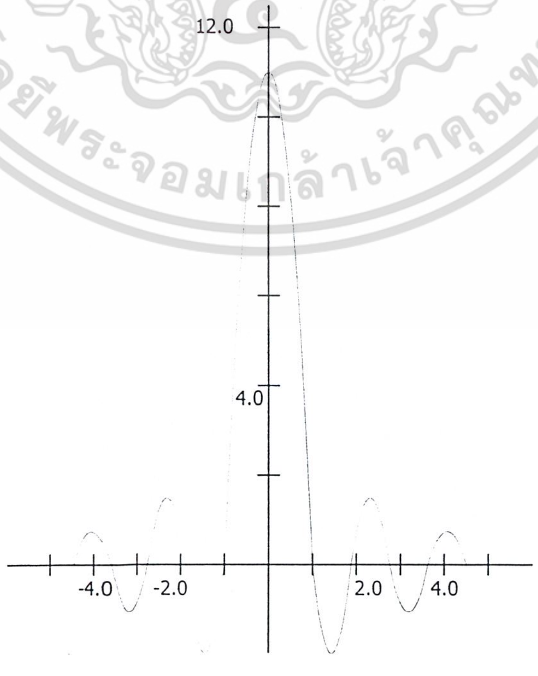
$$= \frac{e^{j\omega NT/2} - e^{-j\omega NT/2}}{e^{j\omega T/2} - e^{-j\omega T/2}} = \frac{\sin(\omega NT/2)}{\sin(\omega T/2)}$$

ผลตอบสนองความถี่ที่เป็นเหตุกภาพ คือ

$$A_R(e^{j\omega T}) = \sum_{n=0}^{N-1} e^{-j\omega T} = e^{-j\omega(N-1)T/2} \frac{\sin(\omega NT/2)}{\sin(\omega T/2)}$$

ผลตอบสนองเฟสเชิงเส้นได้จาก

$$\theta(\omega) = -\omega(N-1)T/2$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 2.20 สเปกตรัมของวินโดว์สี่เหลี่ยม

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. วินโดว์ฮานนและแฮมมิ่ง

จากสมการฟังก์ชั่นฮานนและแฮมมิ่ง

$$a_H(nT) = \begin{cases} \alpha + (1-\alpha)\cos\dfrac{2\pi n}{N-1} & for \quad |n| \le \dfrac{N-1}{2} \\ 0 & otherwise \end{cases} \tag{2.51}$$

ความแตกต่างระหว่างฮานน กับแฮมมิ่ง คือ ค่า α ในกรณีของฮานน $\alpha = 0.5$ และในกรณีของแฮมมิ่ง $\alpha = 0.54$

สเปกตรัมของวินโดว์ฮานนและแฮมมิ่ง สามารถเขียนในพจน์ของวินโดว์สี่เหลี่ยมได้คือ

$$a_H(n) \quad = \quad a_R\{\alpha + (1-\alpha)\cos(\frac{2\pi n}{(N-1)})\}$$

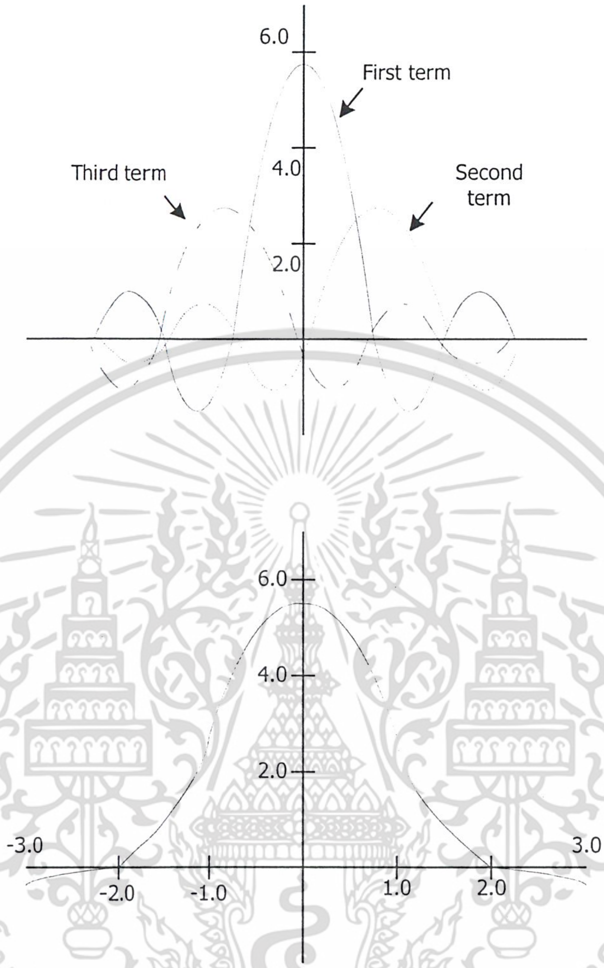
เมื่อประยุกต์ใช้ผลการแปลง Z ได้ผลตอบสนองความถี่เป็น

$$\begin{aligned} A_H(e^{j\omega T}) &= Z[a_H(nT)] \\ &= A_R(e^{j\omega T}) + \frac{1-\alpha}{2} A_R(e^{j[\omega T - 2\pi/(N-1)]}) \\ &\quad + \frac{1-\alpha}{2} A_R(e^{j[\omega T + 2\pi/(N-1)]}) \end{aligned}$$

ฉะนั้นจะได้

$$\begin{aligned} A_H(e^{j\omega T}) &= \frac{\alpha \sin(\omega NT/2)}{\sin(\omega T/2)} + \frac{1-\alpha}{2} \frac{\sin[\omega NT/2 - N\pi/(N-1)]}{\sin[\omega T/2 - \pi/(N-1)]} \\ &\quad + \frac{1-\alpha}{2} \frac{\sin[\omega NT/2 + N\pi/(N-1)]}{\sin[\omega T/2 + \pi/(N-1)]} \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 สเปกตรัมของวินโดว์ฮานนและแฮมมิ่ง

3. วินโดว์แบล็คแมน

จากฟังก์ชั่นแบล็คแมน

$$a_B(nT) = \begin{cases} 0.42 + 0.5\cos\dfrac{2\pi n}{N-1} + 0.08\cos\dfrac{4\pi n}{N-1} & \text{for} \quad |n| \le \dfrac{N-1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{2.52}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิดของวินโดว์	Main-Lobe Width	Ripple ratio(%) N=31	Peak Amplitude Of Side Lobe (dB)	Minimum Stop Band Attennuation(dB)
สี่เหลี่ยม	2 w_s/N	21.80	-13	-21
ฮาน	4 w_s/N	2.67	-31	-44
แฮมมิ่ง	4 w_s/N	0.82	-41	-53
แบล็คแมน	6 w_s/N	0.12	-57	-74

วินโดว์	Ripple Ration		
	N=11	N=21	N=31
สี่เหลี่ยม	22.34	21.89	21.8
ฮาน	2.62	2.67	2.67
แฮมมิ่ง	1.67	0.93	0.82
แบล็คแมน	0.08	0.12	0.12

ตารางที่ 2.4 สรุปคุณสมบัติของวินโดว์

โดยที่ ω_s คือความถี่ในการสุ่ม

2.9 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย

โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) หรือเรียกอย่างย่อๆว่า “DA” เป็นการจัดรูปแบบทางคณิตศาสตร์ของสมการที่อยู่ในรูปของผลบวกของผลคูณ (Sum of Product) ของเลขฐานสิบให้กระจายออกเป็นบิทหรือในรูปแบบของเลขฐานสอง เพื่อให้การคำนวณทางคณิตศาสตร์สามารถแปลงเป็นวงจรดิจิตอลอิเล็กทรอนิกส์ได้ โดยจะปรากฏอยู่ในงานทางด้านการประมวลผลสัญญาณเชิงเลข หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจายที่นำมาใช้งานด้านการประมวลผลสัญญาณเชิงเลข คือการแปลงฟังก์ชันถ่ายโอน (Transfer Function) ซึ่งเป็นสมการที่อยู่ในรูปผลบวกของผลคูณระหว่างสัญญาณอินพุทกับค่าสัมประสิทธิ์ของวงจรกรอง โดยในการคูณกันระหว่างค่าสัมประสิทธิ์ของวงจรกรองกับสัญญาณอินพุท จะใช้การคูณเลขฐานสองแบบส่วนเติมเต็มสอง(2’s complement) และการคูณจะใช้แบบเปิดตาราง (Look-up table หรือ LUT) โดยค่าผลบวกของผลคูณระหว่างสัมประสิทธิ์และสัญญาณอินพุทจะถูกเก็บไว้ในหน่วยความจำ EPROM และจะใช้สัญญาณอินพุทเป็นแอดเดรสของ EPROM โดยตรง ทั้งค่าสัมประสิทธิ์ของวงจรกรองและสัญญาณอินพุทจะถูกแทนด้วยเลขส่วนเติมเต็มสอง ดังนั้นโครงสร้างเลขคณิตกระจายจึงมีทฤษฎีพื้นฐานอยู่บนการคูณแบบเลขส่วนเติมเต็มสอง (2’s complement multiplication)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.1 ทฤษฎีเลขคณิตกระจาย

จากที่กล่าวมาแล้วว่า โครงสร้างเลขคณิตกระจายมีพื้นฐานอยู่บนการคูณเลขส่วนเติมเต็มสอง ดังนั้นในหัวข้อนี้จะได้อธิบายถึงหลักการของการคูณเลขส่วนเติมเต็มสอง

ให้เลขส่วนเติมเต็มสองของ X ซึ่งแทนด้วย $\overline{X}$ และนิยามโดย

$$\overline{X} = \begin{cases} X & , X \geq 0 \\ 2 - |X| & , X < 0 \end{cases} \tag{2.53}$$

โดย X เป็นเลขที่เป็นเศษส่วน (fractional number)

ในระบบเลขส่วนเติมเต็มสองจะใช้บิทที่มีนัยสำคัญสูงสุด (MSB) เป็นบิทแสดงเครื่องหมาย โดยถ้าเป็นบวกแทนด้วย "0" และถ้าเป็นลบแทนด้วย "1" ถ้าให้ X แทนด้วยเลขฐานสองขนาด L+1 บิท ดังนั้นรูปแบบของเลขส่วนเติมเต็มสองจะเขียนได้ดังนี้

$$\overline{X} = X_0.X_1X_2...X_L \tag{2.54}$$

ค่าของ $\overline{X}$ ในรูปของเลขฐานสิบสามารถหาได้ดังนี้

$$X = -X_0 + \sum_{i=1}^{L} X_i 2^{-i} \tag{2.55}$$

จากนั้นพิจารณาผลคูณต่อไปนี้

$$Y = Xm \tag{2.56}$$

ให้ $\overline{Y}$, $\overline{X}$ และ $\overline{m}$ เป็นส่วนเติมเต็มสองของ Y, X และ m ตามลำดับ จากนั้นพิจารณาจากสมการ (2.55) และสมการ (2.56) จะได้

$$\begin{aligned} Y &= -Y_0 + \sum_{i=1}^{L} Y_i 2^{-i} \\ &= -X_0 m + \sum_{i=1}^{L} X_i m 2^{-i} \end{aligned} \tag{2.57}$$

ดังนั้น

$$\begin{aligned} \overline{Y} &= \text{ส่วนเติมเต็มสองของ } \left[-X_0 m + 2^{-1} X_1 m + 2^{-2} X_2 m + 2^{-3} X_3 m + ... + 2^{-L} X_L m\right] \\ &= \text{ส่วนเติมเต็มสองของ } \left[-X_0 m + 2^{-1}(X_1 m + ... + 2^{-1}(X_{L-1} m + 2^{-1}(X_L m)))\right] \end{aligned} \tag{2.58}$$

ต่อไปพิจารณาส่วนเติมเต็มสองของ $2^{-1}U$ โดย

$$\overline{U} = U_0.U_1U_2...U_M$$

สำหรับ $U \geq 0$ (หรือ $U_0 = 0$)

$$\text{ส่วนเติมเต็มสองของ } (2^{-1}U) = 2^{-1}\overline{U}$$

และสำหรับ $U < 0$ (หรือ $U_0 = 1$)

$$\text{ส่วนเติมเต็มสองของ } (2^{-1}U) = 2 - \left|2^{-1}U\right| = 1 + 2^{-1}(2 - |U|) = 1 + 2^{-1}\overline{U}$$

ดังนั้นสรุปได้ว่า

$$\text{ส่วนเติมเต็มสองของ } (2^{-1}U) = \begin{cases} 2^{-1}\overline{U} & , U_0 = 0 \\ 1 + 2^{-1}\overline{U} & , U_0 = 1 \end{cases} \tag{2.59}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการ (2.59) นี้แสดงให้เห็นได้ว่า ส่วนเติมเต็มสองของ $(2^{-1}U)$ เป็นการเลื่อนข้อมูลของ $\overline{U}$ ไปทางขวา 1 บิท ดังนั้น

$$\text{ส่วนเติมเต็มสองของ } (2^{-1}U) = 2_2^{-1}\ \overline{U} \quad (2.60)$$

โดย $2_2^{-1}\overline{U}$ แสดงถึงการเลื่อนข้อมูลของ $\overline{U}$ ไปทางขวา 1 บิท แบบเลขส่วนเติมเต็มสอง ซึ่งสัญลักษณ์ 2_2^{-1} (ซึ่งโดยทั่วไปนิยมเขียนเป็น 2^{-1}) เป็นการแสดงว่า ในกรณีที่ $\overline{U}$ เป็นเลขบวก ซึ่งบิทเครื่องหมายจะเป็นเลขศูนย์ โดยหลังจากเลื่อนข้อมูลไปทางขวา 1 บิทแล้ว บิทที่มาแทนบิทเครื่องหมายก็ยังคงเป็นเลขศูนย์ ส่วนในกรณีที่ $\overline{U}$ เป็นเลขลบ หลังจากเลื่อนข้อมูลไปทางขวา 1 บิทแล้ว บิทที่มาแทนบิทเครื่องหมายจะเป็นเลขหนึ่ง (จาก $1+2^{-1}\overline{U}$) ซึ่งในการสร้างเพื่อใช้งานจริงจำเป็นจะต้องมีวงจรที่ใช้ในการตรวจสอบบิทเครื่องหมาย (sign digit) ทุกครั้งที่มีการเลื่อนข้อมูล

จากนั้นพิจารณาสมการ (2.58) และสมการ (2.59) จะได้ว่า

$$\overline{Y} = -X_0\overline{m} + 2^{-1}X_1\overline{m} + 2^{-2}X_2\overline{m} + 2^{-3}X_3\overline{m} + \ldots + 2^{-L}X_L\overline{m}$$

$$= -X_0\overline{m} + 2^{-1}(X_1\overline{m} + \ldots + 2^{-1}(X_{L-1}\overline{m} + 2^{-1}(X_L\overline{m}))) \quad (2.61)$$

ซึ่งจากสมการ (2.61) จะเห็นได้ว่าผลคูณจากสมการ (2.56) สามารถหาได้โดยการใช้หลักการเลื่อนและบวก (shift and add) หรือ บูทอัลกอริธึม (Booth's algorithm) โดยผลลัพธ์ที่ได้จากการคูณแบบเลขส่วนเติมเต็มสอง สามารถหาได้ตามขั้นตอน ดังนี้

1. เคลียร์ค่าข้อมูลในแอคคิวมูเลเตอร์รีจีสเตอร์ (Accumulator Register)
2. บวก $X_L\overline{m}$ กับค่าที่อยู่ในแอคคิวมูเลเตอร์รีจีสเตอร์
3. เลื่อนค่าที่อยู่ในแอคคิวมูเลเตอร์รีจีสเตอร์ไปทางขวา 1 บิท
4. ทำซ้ำข้อ 2 และ 3 สำหรับค่า $X_{L-1},\ldots, X_1$
5. ลบค่า $X_0\overline{m}$ ออกจากค่าที่อยู่ในแอคคิวมูเลเตอร์รีจีสเตอร์ (ลบแบบเลขส่วนเติมเต็มสอง)

ตัวอย่างการทำงานตามอัลกอริธึมนี้

$Y = Xm = 0.8125(-0.390625)$ โดยสมมติให้ใช้แอคคิวมูเลเตอร์รีจีสเตอร์ขนาด 12 บิท

พิจารณาหาค่า และ ได้ดังนี้

$m = -0.39625$ $\qquad X = 0.8125 = \overline{X}$ (เพราะ X เป็นเลขบวก)

$\overline{m} = 2 - |m|$ (เพราะ m เป็นเลขลบ)

$= 2 - 0.390625$

$= 1.609375$

ดังนั้นจะได้ค่า $\overline{m} = 1.100111$ $\overline{X} = 0.1101 = X_0.X_1X_2X_3X_4$

โดยมีขั้นตอนการทำงาน ดังตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การดำเนินการ	ข้อมูลในแอคคิวมูเลเตอร์รีจีสเตอร์
เคลียร์ ACC	0.000 0000 0000
ACC+ $X_4\overline{m}$	1.100 1110 0000
เลื่อนข้อมูลใน ACC ไปทางขวา 1 บิท	1.110 0111 0000
ACC+ $X_3\overline{m}$	1.110 0111 0000
เลื่อนข้อมูลใน ACC ไปทางขวา 1 บิท	1.111 0011 1000
ACC+ $X_2\overline{m}$	1.100 0001 1000
เลื่อนข้อมูลในACCไปทางขวา 1 บิท	1.110 0000 1100
ACC+ $X_1\overline{m}$	1.010 1110 1100
เลื่อนข้อมูลใน ACC ไปทางขวา 1 บิท	1.101 0111 0110
ACC- $X_0\overline{m}$	1.101 0111 0110

ตารางที่ 2.5 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง

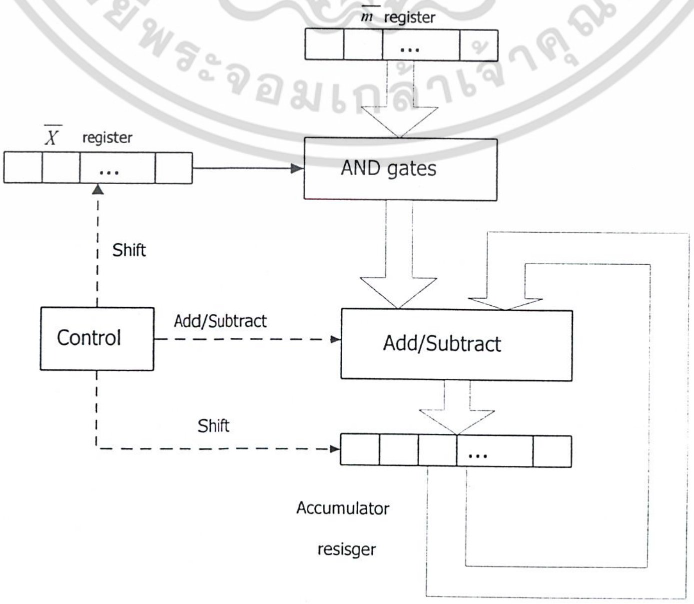
ฉะนั้นจะได้

$$\overline{Y} = Y_0.Y_1Y_2\ldots Y_{11} = 1.101\ 0111\ 0110$$

หาค่าในรูปเลขฐานสิบและจากอัลกอริธึมนี้ สามารถออกแบบการทำงานและสร้างวงจร แสดงได้ดังรูป

$$Y = -Y_0 + \sum_{i=1}^{11} Y_i 2^{-i}$$

$$= -1 + \left[2^{-1} + 2^{-3} + 2^{-5} + 2^{-6} + 2^{-7} + 2^{-9} + 2^{-10}\right] = -0.3173828125$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งต้องมีการอ้างอิงแหล่งที่มาทุกครั้งที่มีการนำไปใช้

รูปที่ 2.22 แสดงการคูณแบบเลขส่วนเติมเต็มสองโดยใช้บูทอัลกอริธึม

ที่ผ่านมาเป็นหลักการคูณแบบเลขส่วนเติมเต็มสอง ส่วนทฤษฎีเลขคณิตกระจายก็อาศัยหลักการดังกล่าวมาใช้ โดยทำการกระจายสมการที่อยู่ในรูปผลบวกของผลคูณให้แตกออกมาอยู่ในระดับบิท (bit level) ซึ่งนำเสนอโดย A.Peled และ B.Liu ดังนี้

พิจารณาผลบวกของผลคูณต่อไปนี้

$$Y = \sum_{i=0}^{N} m_i X_i \tag{2.62}$$

โดย m_i ค่าสัมประสิทธิ์ซึ่งมีค่าคงที่

X_i ข้อมูลอินพุต

ถ้า X_i แต่ละค่าเป็นเลขส่วนเติมเต็มสอง โดย $|X_i| < 1$ สามารถแสดง X_i แต่ละค่าได้ดังนี้

$$X_i = -X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \tag{2.63}$$

โดย X_{ij} บิทต่างๆของข้อมูล X_i มีค่าเป็น 0 หรือ 1

X_{i0} บิทแสดงเครื่องหมาย

X_{iL} บิทที่มีนัยสำคัญต่ำสุด (LSB)

$L+1$ จำนวนบิทที่แทนข้อมูลอินพุต

แทนค่า X_i ในสมการ (2.63) ลงในสมการ (2.62) จะได้

$$Y = \sum_{i=0}^{N} m_i \left[-X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \right]$$

จัดเทอมผลบวกใหม่จะได้

$$\begin{aligned} Y &= -X_{i0} \sum_{i=0}^{N} m_i + \sum_{j=1}^{L} X_{ij} 2^{-j} \sum_{i=0}^{N} m_i \\ &= -\sum_{i=0}^{N} X_{i0} m_i + \sum_{i=0}^{N} \sum_{j=1}^{L} X_{ij} m_i 2^{-j} \end{aligned} \tag{2.64}$$

จากนั้นทำการกระจายออกให้เป็นระดับบิท ได้ดังนี้

$$\begin{aligned} Y = &-(X_{00}m_0 + X_{10}m_1 + X_{20}m_2 + \ldots + X_{N0}m_N) \\ &+ 2^{-1}(X_{01}m_0 + X_{11}m_1 + X_{21}m_2 + \ldots + X_{N1}m_N) \\ &+ 2^{-2}(X_{02}m_0 + X_{12}m_1 + X_{22}m_2 + \ldots + X_{N2}m_N) \\ &+ 2^{-L}(X_{0L}m_0 + X_{1L}m_1 + X_{2L}m_2 + \ldots + X_{NL}m_N) \end{aligned} \tag{2.65}$$

สมการ (2.65) นี้ ถูกกระจายออกให้อยู่ในรูปผลบวกของผลคูณระหว่างสัมประสิทธิ์กับข้อมูลอินพุตในระดับบิท ซึ่งนิยามของการคำนวณแบบเลขคณิตกระจาย และเมื่อเปรียบเทียบสมการ (2.65) กับสมการ (2.61) จะเห็นว่าการคำนวณหาค่า Y ก็ใช้บูทอัลกอริธึมนั่นเอง เพียงแต่เรานำค่าผลคูณย่อย (partial product) ที่คำนวณล่วงหน้าแล้วสำหรับแต่ละค่าที่สอดคล้องกับแต่ละบิทไปเก็บไว้ในตารางเปิดดู ซึ่งเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำ EPROM และใช้ข้อมูลอินพุตเป็นแอดเดรสของหน่วยความจำ เพื่อนำค่าในตารางเปิดดูมาผ่านขั้นตอนการคำนวณตามบูทอัลกอริธึม ซึ่งค่าในตารางเปิดดู สามารถแสดงได้ดังนี้

Bit pattern ของข้อมูลอินพุต $X_{Nj} \ldots\ldots X_{2j}\ X_{1j}\ X_{0j}$	ผลคูณย่อยที่เก็บไว้ในตารางเปิดดู
0 ……… 0 0 0	0
0 ……… 0 0 1	m_0
0 ……… 0 1 0	m_1
0 ……… 0 1 1	$m_1 + m_0$
0 ……… 1 0 0	m_2
0 ……… 1 0 1	$m_2 + m_0$
0 ……… 1 1 0	$m_2 + m_1$
0 ……… 1 1 1	$m_2 + m_1 + m_0$
.	.
.	.
.	.
1 ……… 1 1 1	$m_N + m_{N-1} + \ldots + m_2 + m_1 + m_0$

ตารางที่ 2.6 ค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุต

2.9.2 การนำโครงสร้างเลขคณิตกระจายมาใช้กับวงจรกรองสัญญาณเชิงเลข

ในการนำโครงสร้างเลขคณิตกระจายมาใช้ในการออกแบบวงจรกรองสัญญาณเชิงเลขแบบ FIR โดยนำสมการ (2.5) มาพิจารณาและสามารถกระจายพจน์ของอนุกรมที่ $N = 4$ ได้ดังนี้

$$Y(n) = b_0 X(n) + b_1 X(n-1) + b_2 X(n-2) + b_3 X(n-3) \qquad (2.66)$$

แทนลำดับสัญญาณอินพุต $X(n)$ ด้วยเลขส่วนเติมเต็มสองได้ดังนี้

$$\overline{X}(n) = X_0(n).X_1(n)X_2(n)\ldots X_L(n)$$

และให้ $\overline{b}_i$ เป็นเลขส่วนเติมเต็มสองของ b_i

$X(n)$ สามารถแสดงได้ดังนี้

$$X(n) = -X_0(n) + \sum_{i=1}^{L} X_i(n) 2^{-i}$$

นำค่า $X(n)$ แทนลงในสมการ (2.66) ได้

$$Y(n) = \sum_{i=1}^{L} 2^{-i}\left[b_0 X_i(n) + b_1 X_i(n-1) + b_2 X_i(n-2) + b_3 X_i(n-3)\right]$$

$$-\left[b_0 X_0(n) + b_0 X_0(n-1) + b_0 X_0(n-2) + b_0 X_0(n-3)\right] \qquad (2.67)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คูณ 2^{-1} ทั้งสองข้างของสมการจะได้

$$2^{-1}Y(n)=\sum_{i=1}^{L}2^{-i}\left[2^{-1}b_0X_i(n)+2^{-1}b_1X_i(n-1)+2^{-1}b_2X_i(n-2)+2^{-1}b_3X_i(n-3)\right]$$
$$-\left[2^{-1}b_0X_0(n)+2^{-1}b_0X_0(n-1)+2^{-1}b_0X_0(n-2)+2^{-1}b_0X_0(n-3)\right] \quad (2.68)$$

พิจารณาสมการ (2.68) และสมการ (2.60) จะได้

$$2^{-1}\overline{Y}(n)=\sum_{i=1}^{L}2_2^{-1}\left[2^{-1}\overline{b}_0X_i(n)+2^{-1}\overline{b}_1X_i(n-1)+2^{-1}\overline{b}_2X_i(n-2)+2^{-1}\overline{b}_3X_i(n-3)\right]$$
$$-\left[2^{-1}\overline{b}_0X_0(n)+2^{-1}\overline{b}_0X_0(n-1)+2^{-1}\overline{b}_0X_0(n-2)+2^{-1}\overline{b}_0X_0(n-3)\right] \quad (2.69)$$

เพราะฉะนั้น

$$\overline{Y}(n)=\sum_{i=1}^{L}2_2^{-1}F_i-F_0 \quad (2.70)$$

โดย

$$F_i=\overline{b}_0X_i(n)+\overline{b}_1X_i(n-1)+\overline{b}_2X_i(n-2)+\overline{b}_3X_i(n-3) \quad (2.71)$$

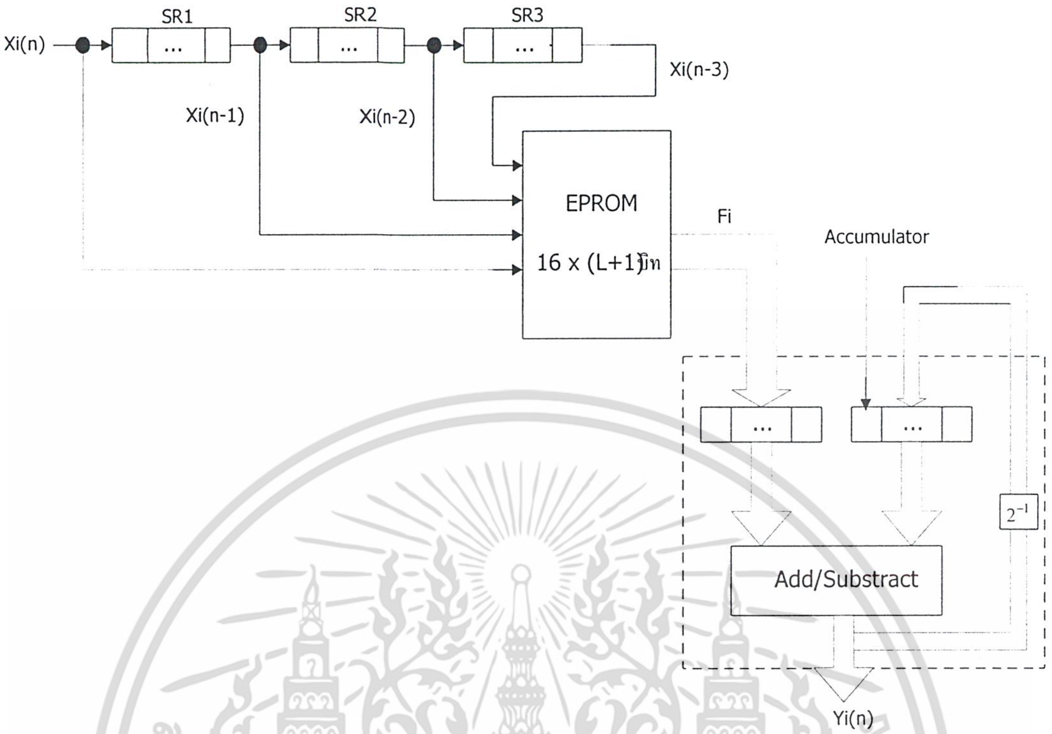
$$F_0=\overline{b}_0X_0(n)+\overline{b}_1X_0(n-1)+\overline{b}_2X_0(n-2)+\overline{b}_3X_0(n-3) \quad (2.72)$$

ส่วนเติมเต็มสองของ สามารถหาได้โดยใช้อัลกอริธึมดังนี้

1. เคลียร์ค่าของข้อมูลในแอคคิวมูเลเตอร์รีจีสเตอร์
2. คำนวณค่า F_i สำหรับ $i=L$
3. บวกค่า F_i กับค่าที่บรรจุอยู่ในแอคคิวมูเลเตอร์รีจีสเตอร์
4. เลื่อนค่าที่บรรจุอยู่ในแอคคิวมูเลเตอร์รีจีสเตอร์ ไปทางขวา 1 บิท (เลื่อนข้อมูลแบบส่วนเติมเต็มสอง)
5. ทำซ้ำข้อ 2 ถึง 4 สำหรับ $i=L-1, L-2,\ldots,1$
6. คำนวณค่า F_0
7. ลบค่า F_0 ออกจากค่าที่บรรจุอยู่ในแอคคิวมูเลเตอร์รีจีสเตอร์ (ลบแบบเลขส่วนเติมเต็มสอง)

โดยอัลกอริธึมที่กล่าวมาสามารถออกแบบการทำงานและสร้างวงจรได้ ดังแสดงในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 แสดงโครงสร้างวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 4

จากรูปที่ 2.23 ส่วนที่อยู่ในเส้นประนิยมเรียกกันว่า สเกลลิ่งแอคคิวมูเลเตอร์รีจิสเตอร์ (Scaling Accumulator) โดยค่าที่อยู่ในแอคคิวมูเลเตอร์รีจิสเตอร์ก่อนที่จะส่งไปบวกกับค่าผลลัพธ์จาก EPROM (หรือ partial sum ตัวต่อไป) จะต้องทำการเลื่อนข้อมูลไป 1 บิทก่อน ดังที่กล่าวมาแล้ว ซึ่งการเลื่อนข้อมูลไปทางขวา 1 บิทนี้ เขียนแทนด้วยการคูณด้วย 2^{-1} และผลจากสมการ (2.71) นำมาสร้างเป็นตารางเปิดดูบรรจุไว้ใน EPROM ค่าในตารางเปิดดูเป็นค่าของ F_i ซึ่งเกิดจากตัวแปรที่เป็นลำดับสัญญาณอินพุต 4 ตัว ดังนั้นค่าของ F_i จะมีค่า $2^4 = 16$ ค่า โดยขนาดของ EPROM จะมีขนาด $16 \times (L+1)$ บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การคำนวณและการสร้าง

3.1 บทนำ

การออกแบบวงจรกรองสัญญาณเชิงเลข มีการพัฒนาไปอย่างมาก โดยการใช้ภาษามาบรรยายการทำงานของวงจร (Hardware Description Languge : HDL) เช่น Verilog หรือ VHDL (VHSIC Hardware Description Languge) โดย VHSIC ย่อมาจาก Very High Speed Integrated Circuits ซึ่งในโครงงานนี้จะใช้ภาษา VHDL (เป็นโปรแกรมภาษาที่สามารถสร้างรูปแบบการทำงานของระบบสัญญาณดิจิตอลเท่านั้น) มาทำการออกแบบวงจรกรองสัญญาณเชิงเลข

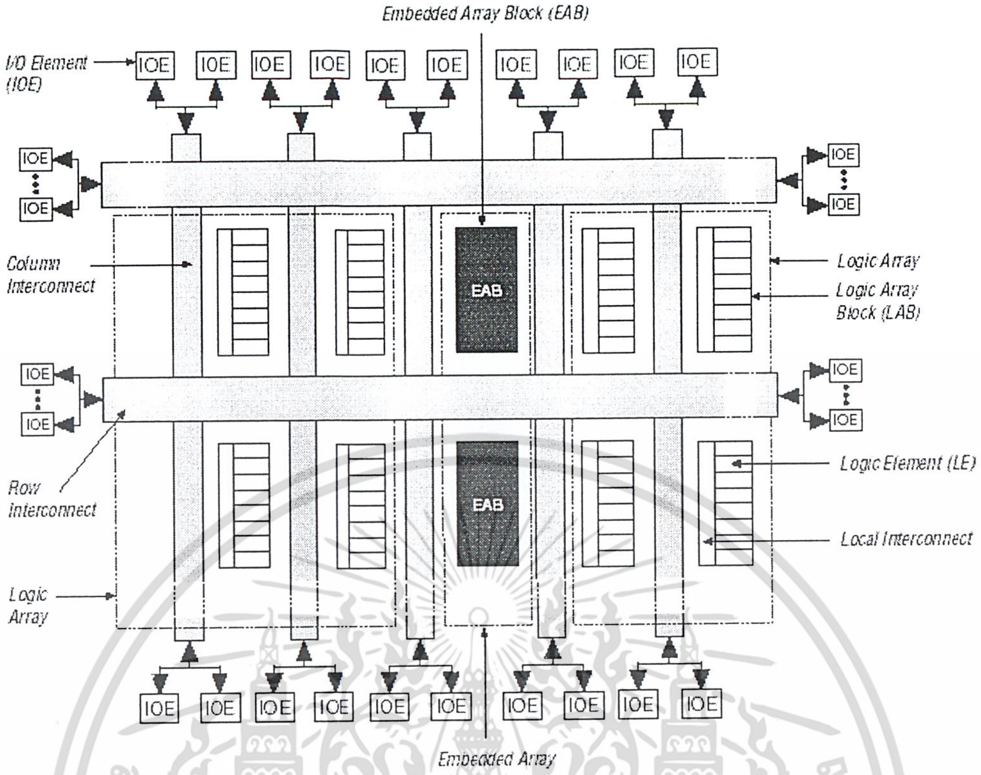
นอกจากนี้ยังนำโปรแกรม Max + Plus II มาใช้งานในการออกแบบวงจรกรองสัญญาณเชิงเลขควบคู่ไปกับการเขียนโปรแกรม VHDL ซึ่งเป็นโปรแกรมที่ทั้งคอมไพล์ (compile) ซิมูเลต (simulate) สังเคราะห์ (synthesis) และสามารถแม็ป (map) วงจรที่ได้จากการสังเคราะห์แล้ว ลงไปยัง FPGA (Field Programmable Gate Array) เพื่อเป็นชิปต้นแบบสำหรับการนำไปทดสอบได้ นั่นคือชิปนั้นจะมีคุณสมบัติตามที่เราเขียนโปรแกรมลงไป

ชิปนั้นมีอยู่หลายแบบ และหลายบริษัท ซึ่งแต่ละแบบก็มีโครงสร้างภายในแตกต่างกัน แต่ในโครงงานนี้เราเลือกใช้ชิปของ Altera ตระกูล FLEX EPF 10LC84-3 ซึ่งสามารถใช้งานกับโปรแกรม Max + Plus II ได้เป็นอย่างดี

3.2 สถาปัตยกรรมภายในของ FLEX EPF 10LC84-3

FPGA ของบริษัท Altera ตระกูล FLEX 10 K เป็นอุปกรณ์ที่มีความหนาแน่นเกทประมาณตั้งแต่ 10,000 – 250,000 เกท โดยการจัดโครงสร้าง (Configuration) จะใช้วิธีโหลดโครงสร้างเข้าไปใน SRAM ภายใน ซึ่งหมายความว่าถ้าไม่ได้มีการจ่ายไฟเลี้ยงให้ โครงสร้างที่จัดเอาไว้ก็จะหายไป FPGA ประเภทนี้จะสามารถโปรแกรมซ้ำได้ไม่จำกัดจำนวนครั้ง และการทำงานตามลอจิกฟังก์ชั่นจะใช้วิธีการเปิดตารางเปิดดูโดยโครงสร้างของ FPGA ตระกูล FLEX 10 K แสดงดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

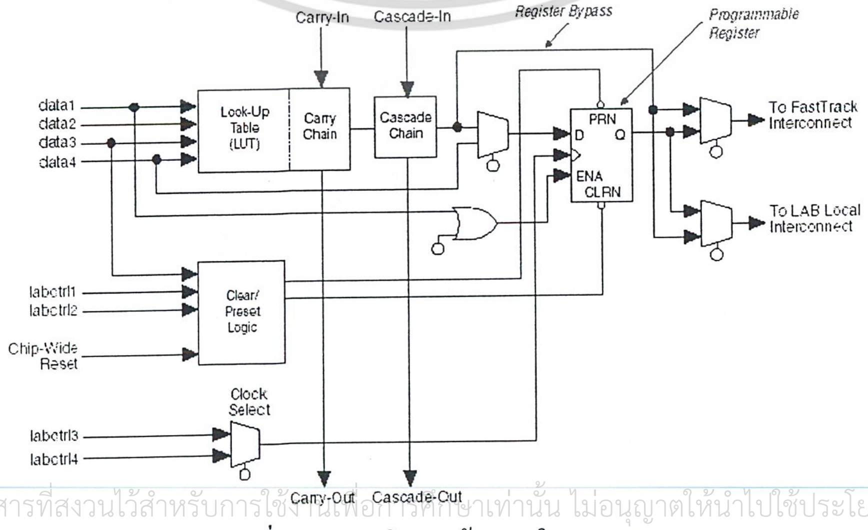


รูปที่ 3.1 แสดงโครงสร้างของ FPGA ตระกูล FLEX 10K

ในโครงสร้างของ FPGA ตระกูล FLEX 10 K สามารถที่จะแบ่งเป็นส่วนต่างๆ ได้ดังนี้

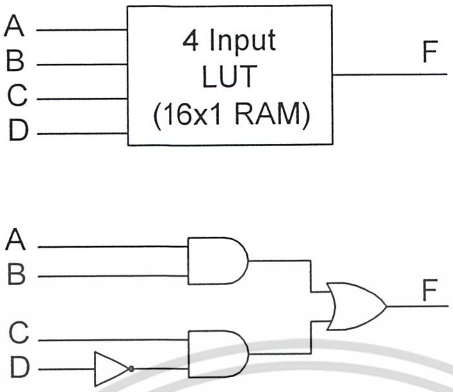
1. Logic Element (LE)

ในรูปที่ 3.2 แสดงโครงสร้างภายในของ LE โดยการกระทำทางบูลีนของลอจิกเกทจะสร้างด้วยวิธีการ LUT โดย LUT คือ 1x16 SRAM ซึ่ง LUT เพียงตัวเดียวสามารถนำมาทำโครงข่ายของลอจิกเกทที่มี 4 อินพุท และ 1 เอาท์พุท โดยโครงข่ายของลอจิกเกทจะถูกแปลงไปเป็นตารางค่าความจริง (Truth Table) ดังแสดงในรูปที่ 3.3



รูปที่ 3.2 แสดงโครงสร้างภายในของ LE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

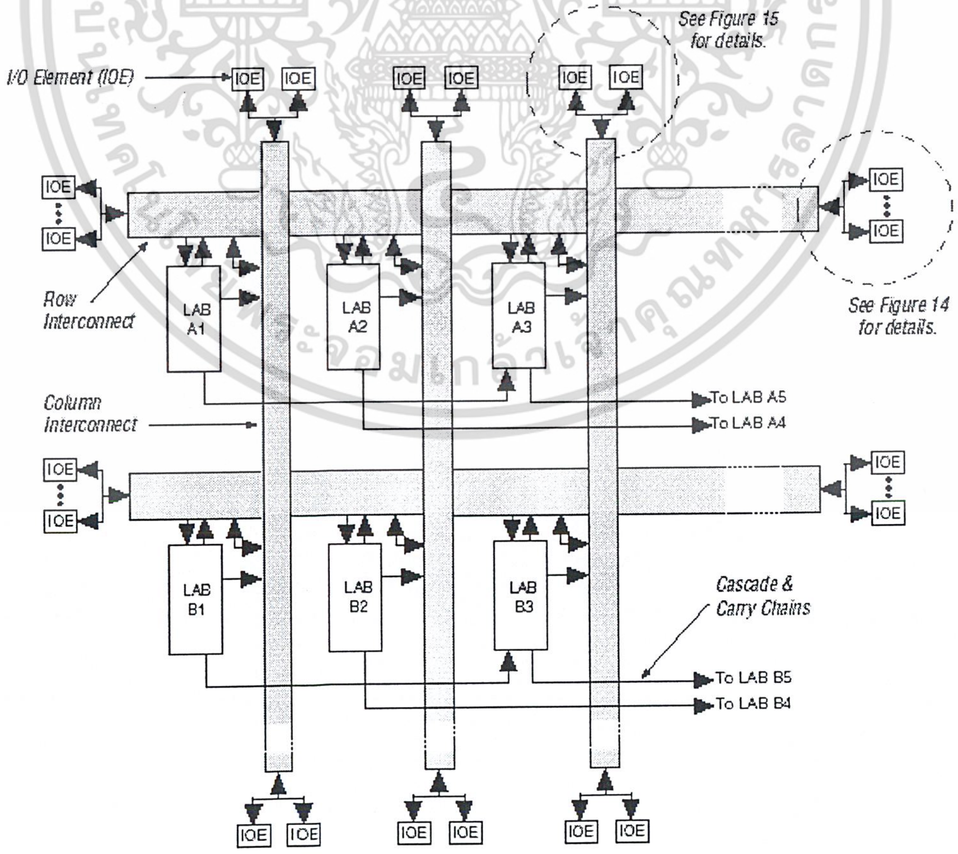


RAM Contents

Address				Data
A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

รูปที่ 3.3 การใช้งาน LUT เป็นโครงข่ายของลอจิก

ถ้าโครงข่ายของลอจิกเกทมีความซับซ้อนขึ้นจะต้องใช้ LUT ของแต่ละ LE เป็นจำนวนหลายตัว โดยเอาท์พุทของ LUT จะส่งต่อไปยังฟลิปฟลอปและต่อไปยังโครงข่ายการเชื่อมต่อ (Interconnection Network) ดังแสดงในรูปที่ 3.4



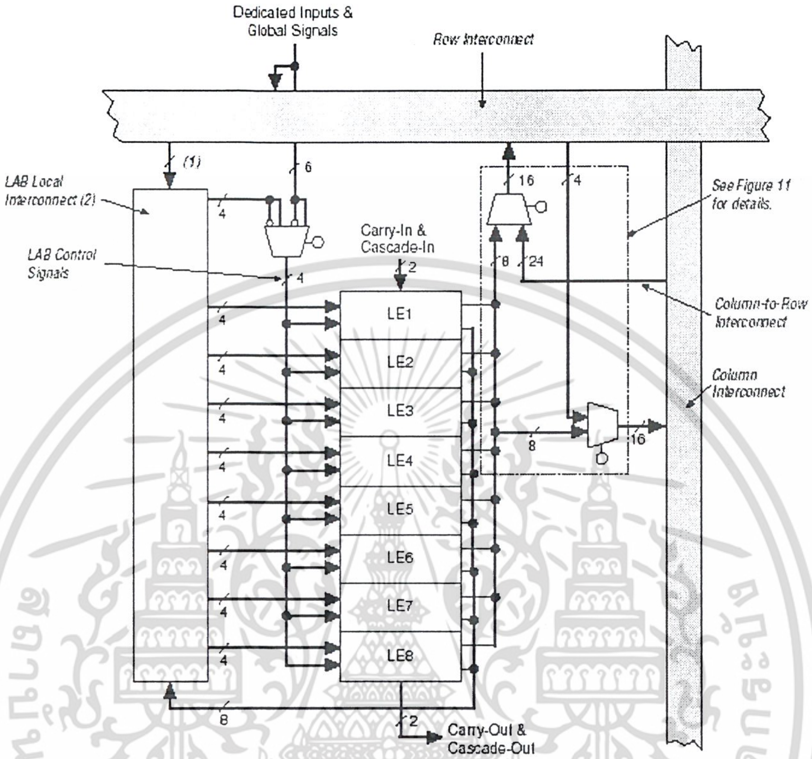
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 3.4 แสดงโครงข่ายของการเชื่อมต่อ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Logic Array Block (LAB)

LAB 1 ตัว จะประกอบไปด้วย 8 LE ดังแสดงในรูปที่ 3.5

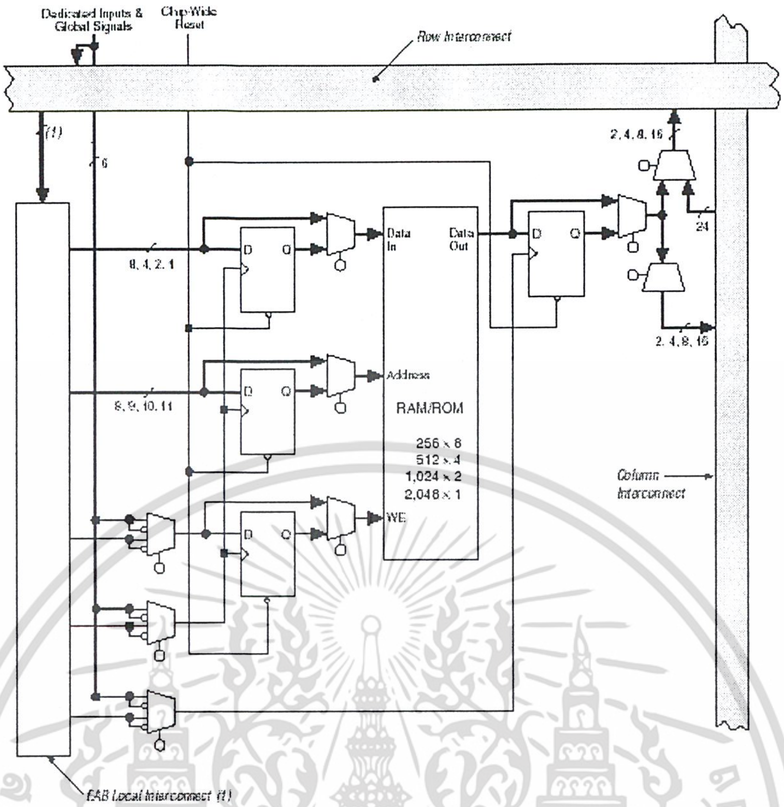


รูปที่ 3.5 แสดงโครงสร้างภายในของ LAB

3. Embedded Array Block (EAB)

สถาปัตยกรรมโดยทั่วไปของ FLEX 10 K จะมีลักษณะของ LAB ที่มีการจัดเรียงแบบเมตริกซ์ และ EAB ซึ่งมีการเชื่อมต่อผ่านทางแถวและคอลัมน์ โดยในแต่ละแถวจะมี 1 EAB ซึ่ง 1 EAB จะมีขนาด 2048 บิท และสามารถกำหนดความกว้าง (Width) ความลึก (Depth) ของ EAB ได้โดยไม่ส่งผลต่อความเร็ว

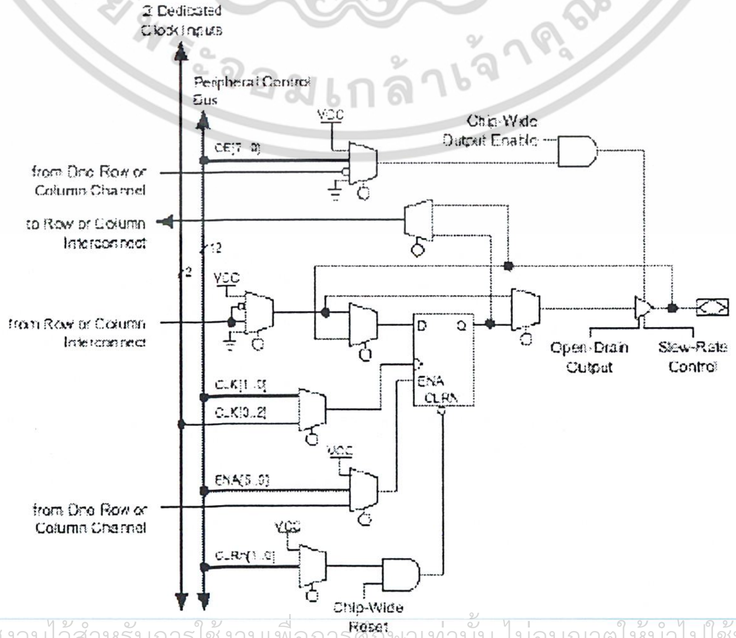
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แสดงโครงสร้างภายใน EAB

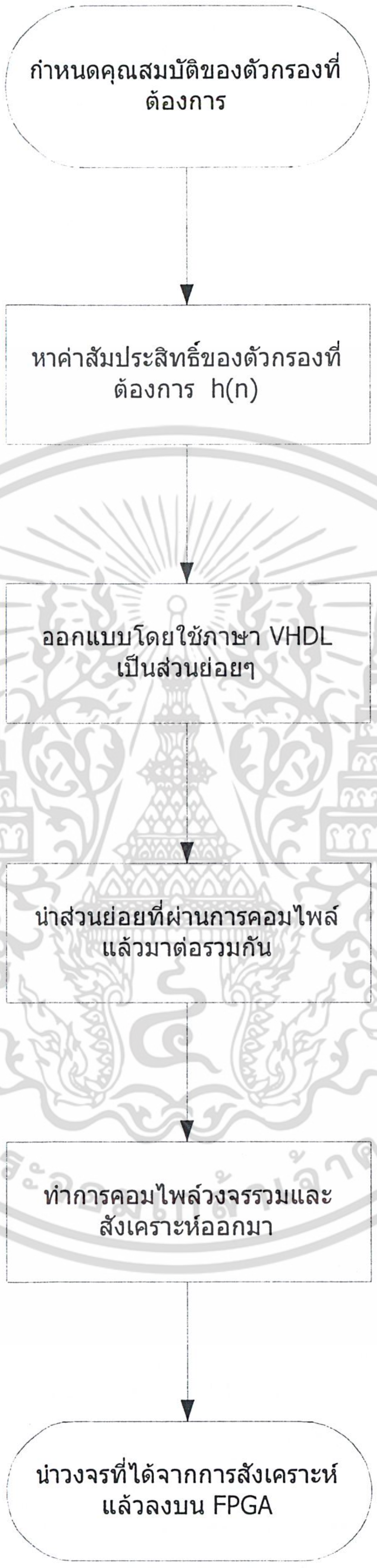
4. Input Out Element (IOE)

IOE จะถูกต่ออยู่กับขา I/O โดยจะประกอบด้วยส่วนของวงจรที่เป็น Tri State และส่วนที่เป็นฟลิปฟลอป ซึ่งเป็น option ดังแสดงในรูปที่ 3.7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.7 แสดงโครงสร้างภายในของ IOE



รูปที่ 3.8 ขั้นตอนการออกแบบวงจรกรองสัญญาณเชิงเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การหาค่าผลตอบสนองอิมพัลส์หรือค่าสัมประสิทธิ์

3.3.1 วงจรกรองความถี่ต่ำผ่านอันดับที่ 8

เมื่อต้องการวงจรกรองความถี่ต่ำผ่านอันดับที่ 8 ที่มีคุณสมบัติดังนี้

- มีการสมมาตรคู่ นั่นคือ เป็น FIR ฟิลเตอร์ชนิดที่ 2 (มีอันดับคู่และเป็นการสมมาตรคู่)
- ความถี่คัทออฟ (ω_c) เท่ากับ 20 กิโลเฮิร์ต
- ความถี่ในการสุ่ม (ω_s) เท่ากับ 250 กิโลเฮิร์ต

ดังนั้น สามารถหาค่าสัมประสิทธิ์ได้โดยใช้โปรแกรม MATLAB ซึ่งเป็นการหาค่าสัมประสิทธิ์ที่ผ่านการถ่วงน้ำหนักอนุกรมฟูริเยอร์ด้วยฟังก์ชั่นวินโดว์ที่เหมาะสมแล้ว ในที่นี้เราเลือกใช้วินโดว์แบล็คแมน จะได้ค่าสัมประสิทธิ์ดังนี้

h(0) = 0
h(1) = 0.0245
h(2) = 0.1493
h(3) = 0.3262
h(7) = 0
h(6) = 0.0245
h(5) = 0.1493
h(4) = 0.3262

เนื่องจากค่า h(n)ที่ได้มาเป็นค่าจุดทศนิยม เราสามารถทำให้เป็นค่าเป็นจำนวนเต็มที่มีค่า –127 ถึง +127 ก่อน เพื่อง่ายต่อการแปลงเป็นเลขฐานสองได้ โดยใช้สูตร

$$round\left[\frac{127\times h(n)}{h_{\max}}\right]$$

หลังจากนั้นจึงทำการแปลงค่า h(n)ให้อยู่ในรูปของเลขฐานสอง จะได้ค่าดังตารางที่ 3.1

n	h(n)		
0	0	0	0000 0000
1	0.0245	10	0000 1010
2	0.1493	58	0011 1010
3	0.3262	127	0111 1111
4	0.3262	127	0111 1111
5	0.1493	58	0011 1010
6	0.0245	10	0000 1010
7	0	0	0000 0000

ตารางที่ 3.1 แสดงค่าสัมประสิทธิ์ของวงจรกรองความถี่ต่ำผ่านอันดับที่ 8

และเนื่องจากคุณสมบัติของตัว FIR ฟิลเตอร์ชนิดที่ 2 ที่มีความสมมาตรกันดังนั้นจากสมการ (2.28) $h(n) = h(N-n-1)$ ทำให้สามารถเลือกใช้ค่า h(n) เพียง 4 ค่า คือ h(0), h(1), h(2) และ h(3)ได้

3.3.2 วงจรกรองความถี่ต่ำผ่านอันดับที่ 12

เมื่อต้องการวงจรกรองความถี่ต่ำผ่านอันดับที่ 12 ที่มีคุณสมบัติดังนี้

- มีการสมมาตรคู่ นั่นคือ เป็น FIR ฟิลเตอร์ชนิดที่ 2 (มีอันดับคู่และเป็นการสมมาตรคู่)
- ความถี่คัทออฟ (ω_c) เท่ากับ 20 กิโลเฮิร์ต
- ความถี่ในการสุ่ม (ω_s) เท่ากับ 250 กิโลเฮิร์ต

ดังนั้น สามารถหาค่าสัมประสิทธิ์ได้โดยใช้โปรแกรม MATLAB และเลือกใช้วินโดว์แบล็คแมน จะได้ค่าสัมประสิทธิ์ดังนี้

h(0) = 0 h(7) = 0

h(1) = 0.0027 h(6) = 0.0027

h(2) = 0.0219 h(5) = 0.0219

h(3) = 0.0769 h(8) = 0.0769

h(4) = 0.1639 h(7) = 0.1639

h(5) = 0.2346 h(6) = 0.2346

หลังจากนั้นทำการแปลงค่า h(n) เป็นเลขฐานสิบก่อนจะทำให้อยู่ในรูปของเลขฐานสอง จะได้ค่าดงตารางที่ 3.2

n	h(n)		
0	0	0	0000 0000
1	0.0027	1	0000 0001
2	0.0219	12	0000 1100
3	0.0769	42	0010 1010
4	0.1639	89	0101 1001
5	0.2346	127	0111 1111
6	0.2346	127	0111 1111
7	0.1639	89	0101 1001
8	0.0769	42	0010 1010
9	0.0219	12	0000 1100
10	0.0027	1	0000 0001
11	0	0	0000 0000

ตารางที่ 3.2 แสดงค่าสัมประสิทธิ์ของวงจรกรองความถี่ต่ำผ่านอันดับที่ 12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเนื่องจากความสมมาตรกันดังนั้นจากสมการ (2.28) ทำให้สามารถเลือกใช้ค่า h(n) เพียง 6 ค่า คือ h(0), h(1), h(2), h(3), h(4) และ h(5) ได้

3.3.3 วงจรกรองความถี่ต่ำผ่านอันดับที่ 16

เมื่อต้องการวงจรกรองความถี่ต่ำผ่านอันดับที่ 16 ที่มีคุณสมบัติดังนี้

- มีการสมมาตรคู่ นั่นคือ เป็น FIR ฟิลเตอร์ชนิดที่ 2 (มีอันดับคู่และเป็นการสมมาตรคู่)
- ความถี่คัทออฟ (ω_c) เท่ากับ 20 กิโลเฮิร์ต
- ความถี่ในการสุ่ม (ω_s) เท่ากับ 250 กิโลเฮิร์ต

ดังนั้น สามารถหาค่าสัมประสิทธิ์ได้โดยใช้โปรแกรม MATLAB และเลือกใช้วินโดว์แบล็คแมน จะได้ค่าสัมประสิทธิ์ดังนี้

h(0) = 0
h(1) = -0.0001
h(2) = 0.0020
h(3) = 0.0136
h(4) = 0.0437
h(5) = 0.0947
h(6) = 0.1531
h(7) = 0.1930
h(15) = 0
h(14) = -0.0001
h(13) = 0.0020
h(12) = 0.0136
h(11) = 0.0437
h(10) = 0.0947
h(9) = 0.1531
h(8) = 0.1930

หลังจากนั้นทำการแปลงค่า h(n) เป็นเลขฐานสิบก่อนจะทำให้อยู่ในรูปของเลขฐานสอง จะได้ค่าดังตารางที่ 3.3

n	h(n)		
0	0	0	0000 0000
1	-0.0001	0	0000 0000
2	0.0020	1	0000 0001
3	0.0136	9	0000 1001
4	0.0437	29	0001 1101
5	0.0947	62	0011 1110
6	0.1531	101	0110 0101
7	0.1930	127	0111 1111
8	0.1930	127	0111 1111
9	0.1531	101	0110 0101
10	0.0947	62	0011 1110
11	0.0437	29	0001 1101
12	0.0136	9	0000 1001
13	0.0020	1	0000 0001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

n	h(n)		
14	-0.0001	0	0000 0000
15	0	0	0000 0000

ตารางที่ 3.3 แสดงค่าสัมประสิทธิ์ของวงจรกรองความถี่ต่ำผ่านอันดับที่ 16

และเนื่องจากความสมมาตรกันดังนั้นจากสมการ (2.28) ทำให้สามารถเลือกใช้ค่า h(n) เพียง 6 ค่า คือ h(0), h(1), h(2), h(3), h(4), h(5), h(6) และ h(7)ได้

3.4 การออกแบบวงจรกรองสัญญาณเชิงเลขโดยภาษา VHDL

ในการใช้ภาษา VHDL มาช่วยในการออกแบบวงจรกรองสัญญาณเชิงเลข ซึ่งมีความแตกต่างเป็นอย่างมากเมื่อเปรียบเทียบกับการออกแบบในอดีตที่ผ่านมา คือผู้ออกแบบจะต้องนำเอาอุปกรณ์แต่ละตัวที่ทำการออกแบบมาทำการต่อทดลองในแผงวงจรจริง และทำการทดสอบวงจรเพื่อหาข้อผิดพลาด ซึ่งต้องใช้เวลานานกับการแก้ปัญหาแต่ละอย่างที่เกิดขึ้น แต่ในการออกแบบด้วยภาษา VHDL ผู้ออกแบบเพียงแต่เขียนโค้ด (source code) มาบรรยายการทำงานของวงจร ไม่ต้องสร้างวงจรจริง หลังจากนั้นก็ทำการคอมไพล์แล้วซิมูเลตดูว่าได้ไทม์มิ่ง (Timing) ตามที่ต้องการหรือไม่ จากนั้นก็นำโค้ด (source code) ที่ได้ไปทำการสังเคราะห์ด้วยโปรแกรมสังเคราะห์ (Synthesis tool) แล้วทำการโปรแกรมลงชิปเพียงตัวเดียว ก็จะทำให้ชิปนั้นทำงานตามวงจรที่เราต้องการได้ ซึ่งมีข้อดีคือ ลดความยุ่งยากในการนำอุปกรณ์มาเชื่อมต่อกันให้เป็นวงจร รวมทั้งลดเวลาในการออกแบบและทดสอบการทำงาน

ภาษา VHDL สามารถเขียนได้ 3 แบบ คือ

1. Structural เป็นวิธีการออกแบบโดยการแตกองค์ประกอบออกเป็นบล็อกย่อยๆ ซึ่งแต่ละบล็อกนี้จะเชื่อมต่อเป็นวงจรสมบูรณ์ แต่ละบล็อกจะประกอบไปด้วยคอมโพเน้นท์ (component) ที่มีอยู่
2. Data Flow เป็นวิธีการออกแบบที่อธิบายว่า สัญญาณอินพุตและเอาท์พุตนั้นเชื่อมต่อกันอย่างไร หรือพูดได้อีกแบบหนึ่งว่า สัญญาณต่างๆมีการไหลอย่างไรในวงจร
3. Behavioral เป็นวิธีการออกแบบที่ไม่ต้องสนใจถึงวิธีการสร้างวงจรสำหรับใช้งานจริงๆ มักจะใช้เพื่อประโยชน์ 2 ประการ ดังนี้ 1.) ทดสอบการทำงานในระบบที่ซับซ้อน 2.) เป็นการออกแบบที่สะดวกและใช้งานง่าย

3.4.1 ขั้นตอนในการออกแบบวงจรกรองสัญญาณเชิงเลข

การออกแบบวงจรด้วยภาษา VHDL นั้นนำหลักการของโครงสร้างเลขคณิตกระจายมาพิจารณา มีขั้นตอนในการออกแบบดังนี้

1. ทำการออกแบบแต่ละส่วนย่อย (module) ของวงจรกรองสัญญาณเชิงเลข ซึ่งประกอบด้วย

   1.1 รีจีสเตอร์ RX (Parallel in Serial out shift register : PISO)

   1.2 รีจีสเตอร์ SR (Serial in Serial out shift register : SISO)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 หน่วยความจำ (EPROM)

1.4 วงจรบวก/ลบ (ADD/SUB)

1.5 แอคคิวมูเลเตอร์ (Accumulator : ACC)

1.6 บัฟเฟอร์ (Buffer)

1.7 หน่วยควบคุม (Control Unit)

1.8 วงจรบวกบิทซีเรียล (Bit-serial adder)

2. ทำการทดสอบเพื่อดูผลการทำงานของแต่ละส่วนย่อย
3. ทำการรวมทุกๆ ส่วนย่อยเข้าด้วยกันเป็นวงจรกรองสัญญาณเชิงเลข โดยเชื่อมต่อสัญญาณเข้าด้วยกัน นั่นคือเป็นการเขียนโค้ดในลักษณะ Structural
4. ทำการทดสอบการทำงานหลังจากรวมทุกๆส่วนย่อยเข้าด้วยกัน
5. ทำการสังเคราะห์วงจรด้วยโปรแกรมการสังเคราะห์
6. ทำการอิมพลีเมนต์ (implement) วงจรลงบนอุปกรณ์ FPGA
7. ทำการโปรแกรมข้อมูลลงบนอุปกรณ์ FPGA

3.4.2 การออกแบบแต่ละส่วนย่อย

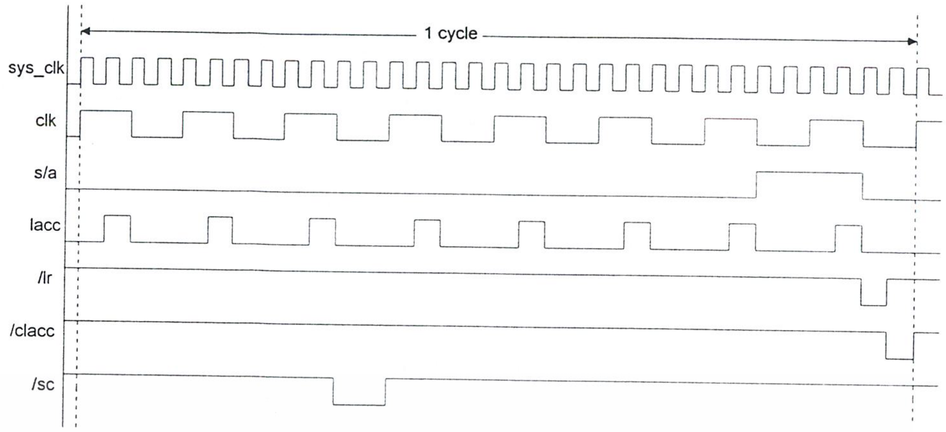
การออกแบบแต่ละส่วนย่อยทำให้ง่ายต่อการตรวจสอบข้อผิดพลาด

1. รีจีสเตอร์ RX จะทำหน้าที่เป็นวงจรแปลงสัญญาณขนานเป็นอนุกรมขนาด 8 บิท โดยภายในจะประกอบไปด้วย D ฟลิบฟลอบ (D flip-flop) และวงจรเลื่อนบิท
2. รีจีสเตอร์ SR จะทำหน้าที่เป็นวงจรเลื่อนบิทขนาด 8 บิท โดยภายในจะประกอบไปด้วย D ฟลิบฟลอบ จำนวน 8 ตัวต่ออนุกรมกัน
3. หน่วยความจำ จะเป็นแบบเก็บค่าไว้ในตารางเปิดดู สามารถเรียกไปใช้ได้โดยหน่วยการออกแบบเอนทิตี้ (Entity Design Unit) และ หน่วยการออกแบบสถาปัตยกรรม (Architecture Design Unit) โดยในที่นี้จะใช้รอม (ROM) ของโปรแกรม Max + Plus II เอง เพราะจะทำงานดีกว่ารอม ที่เราเขียนเอง สามารถคำนวณหาค่าในตารางเปิดดูได จากตารางที่ 2.6
4. วงจรบวก/ลบ เป็นการบวกแบบเลขส่วนเติมเต็มสองเลขขนาด 8 บิท
5. แอคคิวมูเลเตอร์ เป็นวงจรเก็บค่าขนาด 9 บิท

ค่าที่อยู่ในแอคคิวมูเลเตอร์ ก่อนที่จะส่งไปบวกกับค่าผลลัพธ์ที่ได้จาก EPROM จะต้องทำการเลื่อนข้อมูลไป 1 บิทก่อน เพื่อป้องกันการเกิดการล้น (overflow) เรียกส่วนนี้ว่า สเกลลิ่งแอคคิวมูเลเตอร์

6. บัฟเฟอร์ เป็นวงจรที่ทำหน้าที่ค้างค่าเอาท์พุตไว้ เพื่อรอแปลงสัญญาณเชิงเลขกลับเป็นสัญญาณเชิงอุปมาน มีขนาด 8 บิท
7. หน่วยควบคุม จะใช้การเขียนในลักษณะของ FSM (Finite State Machine) เพื่อผลิตสัญญาณควบคุม clk, /lr, /sc, l_acc, /cl_acc, s_a ในการควบคุมวงจรรวม สามารถแสดงด้วยไทม์มิ่งไดอะแกรม (Timing Diagram) ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 ไทม์มิ่งไดอะแกรมของสัญญาณควบคุมทั้งหมด

8. วงจรบิทอนุกรม จะประกอบไปด้วย 2 ส่วน คือ วงจรบวก (full adder) และ D ฟลิปฟลอป โดยข้อมูลจะถูกส่งมาทีละ 1 บิท โดยส่งบิทที่มีนัยสำคัญต่ำสุด (least-significant bit) เข้ามาก่อน แล้วทำการบวกทีละบิท ในส่วนของตัวทด (carries out) จะถูกเก็บไว้ที่ D ฟลิปฟลอป เพื่อฟีดแบ็ก (feedback) กลับไปทำการบวกกับบิทถัดไปที่ส่งเข้ามา สถานะเริ่มต้นของ D จะต่างกันสำหรับการบวกกับการลบ D สำหรับการบวก จะเริ่มต้นที่ 0 ส่วนการลบจะเริ่มต้นที่ 1 ก่อนวงจรบิทอนุกรม จะทำให้ขนาดของหน่วยความจำเล็กลง โดยอาศัยคุณสมบัติการสมมาตรของตัวกรอง $h(n) = h(N-n-1)$

3.4.3 การหาค่าของตารางเปิดดู

3.4.3.1 วงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 8

ถ้าต้องการหาค่าของตารางเปิดดูของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 8 ซึ่งเลือกใช้ค่า h(n) เพียง 4 ค่า คือ h(0), h(1), h(2) และ h(3) ทำให้ใช้ EPROM ที่มีแอดเดรสขนาด 4 บิท สามารถหาค่าได้โดยเทียบจากตารางที่ 2.6 จะได้รูปแบบตารางดังนี้

Address	Data	Address	Data
0000	0	1000	h(3)
0001	h(0)	1001	h(3)+h(0)
0010	h(1)	1010	h(3)+h(1)
0011	h(1)+h(0)	1011	h(3)+h(1)+h(0)
0100	h(2)	1100	h(3)+h(2)
0101	h(2)+h(0)	1101	h(3)+h(2)+h(0)
0110	h(2)+h(1)	1110	h(3)+h(2)+h(1)
0111	h(2)+h(1)+h(0)	1111	h(3)+h(2)+h(1)+h(0)

ตารางที่ 3.4 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 4 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงแหล่งที่มาของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่าสัมประสิทธิ์ h(n) จากตารางที่ 3.1 มาแทนค่าในตารางที่ 3.4 จะไดค่าที่เก็บอยู่ในตารางเปิดดูทั้งหมด 2^4 ค่า แต่ค่าที่นำมาบวกกันทำให้เกิดการล้น คือ ผลบวกที่ได้มีบิทข้อมูลไปทับบิทเครื่องหมาย ทำให้ถ้านำไปใช้จะเกิดความผิดพลาดได้ จึงต้องนำค่าที่คำนวณได้ทุกค่ามาหาร 2 หรือ คือการเลื่อนบิทไปทางขวา 1 บิท นั่นเอง

2^{-1}

0000 0000	=>	0000 0000
0000 1010	=>	0000 0101
0001 1101	=>	0000 1110
0100 0100	=>	0010 0010
0111 1111	=>	0011 1111
1000 1001	=>	0100 0100
1011 1001	=>	0101 1100
1100 0011	=>	0110 0001

ดังนั้นจะได้ตารางเปิดดูของวงจรกรองความถี่ต่ำผ่านที่สามารถนำไปใช้งานได้ ดังตารางที่ 3.5

Address	Data	Address	Data
0000	0000 0000	1000	0011 1111
0001	0000 0000	1001	0011 1111
0010	0000 0101	1010	0100 0100
0011	0000 0101	1011	0100 0100
0100	0001 1101	1100	0101 1100
0101	0001 1101	1101	0101 1100
0110	0010 0010	1110	0110 0001
0111	0010 0010	1111	0110 0001

ตารางที่ 3.5 ตารางเปิดดูของวงจรกรองความถี่ต่ำผ่านอันดับที่ 8

ข้อสังเกต จำไว้ว่า เมื่อมีการหารออก ก็ต้องมีการคูณกลับ หรือมีการเลื่อนบิทไปทางขวา 1 บิท ก็ต้องเลื่อนบิทกลับไปทางซ้าย 1 บิทเช่นกัน โดยจะทำเป็นขั้นตอนสุดท้าย ก่อนออกไปเป็นเอาท์พุต

3.4.3.2 วงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 12

ถ้าต้องการหาค่าของตารางเปิดดูของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 12 ซึ่งเลือกใช้ค่า h(n) เพียง 6 ค่า คือ h(0), h(1), h(2), h(3), h(4) และ h(5) ทำให้ใช้ EPROM ที่มีแอดเดรสขนาด 6 บิท สามารถหาค่าได้โดยเทียบจากตารางที่ 2.6 จะได้รูปแบบตารางดังนี้

เอกสารนี้เป็นเอกสารที่สงวนสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Data
00 0000	0
00 0001	h(0)
00 0010	h(1)
00 0011	h(0)+h(1)
00 0100	h(2)
00 0101	h(2)+h(0)
00 0110	h(2)+h(1)
. .	. .
11 1111	h(5)+h(4)+h(3)+h(2)+h(1)+h(0)

ตารางที่ 3.6 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 6 บิท

จะพบว่าต้องใช้ EPROM ที่มีแอดแดรสขนาด 6 บิท ซึ่งจะได้ค่าในตารางเปิดดูทั้งหมด $2^6 = 64$ ค่า เราสามารถทำการลดขนาดของ EPROM ลงได้อีก โดยแบ่งเป็น EPROM ที่มีแอดแดรสขนาด 3 บิท 2 ตัว จะได้ตารางเปิดดูของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 12 ดังนี้

Address	Data	Address	Data
000	0	100	h(2)
001	h(0)	101	h(2)+h(0)
010	h(1)	110	h(2)+h(1)
011	h(1)+h(0)	111	h(2)+h(1)+h(0)

ตารางที่ 3.7 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 3 บิท ตัวที่ 1

Address	Data	Address	Data
000	0	100	h(5)
001	h(3)	101	h(5)+h(3)
010	h(4)	110	h(5)+h(4)
011	h(4)+h(3)	111	h(5)+h(4)+h(3)

ตารางที่ 3.8 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 3 บิท ตัวที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่าสัมประสิทธิ์ h(n) จากตารางที่ 3.2 มาแทนค่าในตารางที่ 3.7 และ 3.8 แล้วถ้าเกิดการล้นก็ทำเช่นเดียวกับการหาค่าของตารางเปิดดูของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 8

3.4.3.3 วงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 16

ถ้าต้องการหาค่าของตารางเปิดดูของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 16 ซึ่งเลือกใช้ค่า h(n) เพียง 8 ค่า คือ h(0), h(1), h(2), h(3), h(4), h(5), h(6) และ h(7) ทำให้ใช้ EPROM ที่มีแอดเดรสขนาด 8 บิท สามารถหาค่าได้โดยเทียบจากตารางที่ 2.6 จะได้รูปแบบตารางดังนี้

Address	Data
0000 0000	0
0000 0001	h(0)
0000 0010	h(1)
0000 0011	h(0)+h(1)
0000 0100	h(2)
0000 0101	h(2)+h(0)
0000 0110	h(2)+h(1)
0000 0111	h(2)+h(1)+h(0)
0000 1000	h(3)
. . .	. . .
1111 1111	h(7)+h(6)+h(5)+h(4)+h(3)+h(2)+h(1)+h(0)

ตารางที่ 3.9 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 8 บิท

จะพบว่าต้องใช้ EPROM ที่มีแอดแดรสขนาด 8 บิท ซึ่งจะได้ค่าในตารางทั้งหมด $2^8 = 256$ ค่า เราสามารถลดขนาดของ EPROM ลงได้อีก โดยแบ่งเป็น EPROM ที่มีแอดแดรสขนาด 4 บิท 2 ตัว จะได้ตารางเปิดดูของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 16 ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Data	Address	Data
0000	0	1000	h(3)
0001	h(0)	1001	h(3)+h(0)
0010	h(1)	1010	h(3)+h(1)
0011	h(1)+h(0)	1011	h(3)+h(1)+h(0)
0100	h(2)	1100	h(3)+h(2)
0101	h(2)+h(0)	1101	h(3)+h(2)+h(0)
0110	h(2)+h(1)	1110	h(3)+h(2)+h(1)
0111	h(2)+h(1)+h(0)	1111	h(3)+h(2)+h(1)+h(0)

ตารางที่ 3.10 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 4 บิท ตัวที่ 1

Address	Data	Address	Data
0000	0	1000	h(7)
0001	h(4)	1001	h(7)+h(4)
0010	h(5)	1010	h(7)+h(5)
0011	h(5)+h(4)	1011	h(7)+h(5)+h(4)
0100	h(6)	1100	h(7)+h(6)
0101	h(6)+h(4)	1101	h(7)+h(6)+h(4)
0110	h(6)+h(5)	1110	h(7)+h(6)+h(5)
0111	h(6)+h(5)+h(4)	1111	h(7)+h(6)+h(5)+h(4)

ตารางที่ 3.11 ตารางเปิดดูของ EPROM ที่มีแอดแดรสขนาด 4 บิท ตัวที่ 2

นำค่าสัมประสิทธิ์ h(n) จากตารางที่ 3.3 มาแทนค่าในตารางที่ 3.10 และ 3.11 แล้วถ้าเกิดการล้นก็ทำเช่นเดียวกับการหาค่าของตารางเปิดดูของวงจรกรองความถี่ไม่ป้อนกลับเชิงเลขอันดับที่ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

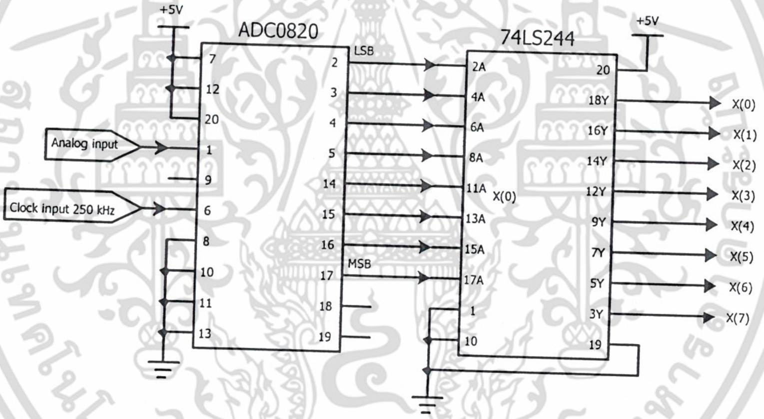
3.5 การออกแบบสร้างและการทำงานวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลข

สามารถแบ่งออกเป็น 3 ส่วนใหญ่ ดังนี้

รูปที่ 3.10 บล็อกไดอะแกรมของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลข

3.5.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter)

จะทำการแปลงสัญญาณเชิงอุปมาน $x(t)$ ให้เป็นลำดับสัญญาณเชิงเลข $x(n)$ ขนาด 8 บิท เป็นการถ่ายโอนข้อมูลแบบขนาน ซึ่งถูกควบคุมด้วยสัญญาณ /sc



รูปที่ 3.11 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

จากรูปที่ 3.11 สามารถอธิบายการทำงานของวงจรได้ดังนี้

1. การสุ่มตัวอย่าง (Sampling) จะแปลงสัญญาณต่อเนื่องไปเป็นสัญญาณเวลาช่วง โดยดึงตัวอย่าง (sample) ของสัญญาณเวลาเป็นช่วงที่ช่วงเวลาที่เท่ากัน
2. การควอนไทซ์ (Quantization) จะเปลี่ยนสัญญาณจากสัญญาณเวลาต่อเนื่องขนาดเวลาต่อเนื่อง ซึ่งเป็นสัญญาณ สุ่ม (Sample signal) เป็นสัญญาณเวลาเป็นช่วง ขนาดเป็นช่วง (ดิจิตอล) ค่าของแต่ละตัวอย่างของสัญญาณจะถูกแทนด้วยค่าที่เลือกจากเซตของค่าที่เป็นไปได้ที่จำกัด
3. การเข้ารหัส (Coding) ของแต่ละค่าที่ผ่านการควอนไทซ์แล้ว จะถูกแทนด้วยลำดับ (sequence) ของเลขฐานสองจำนวน 8 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลข

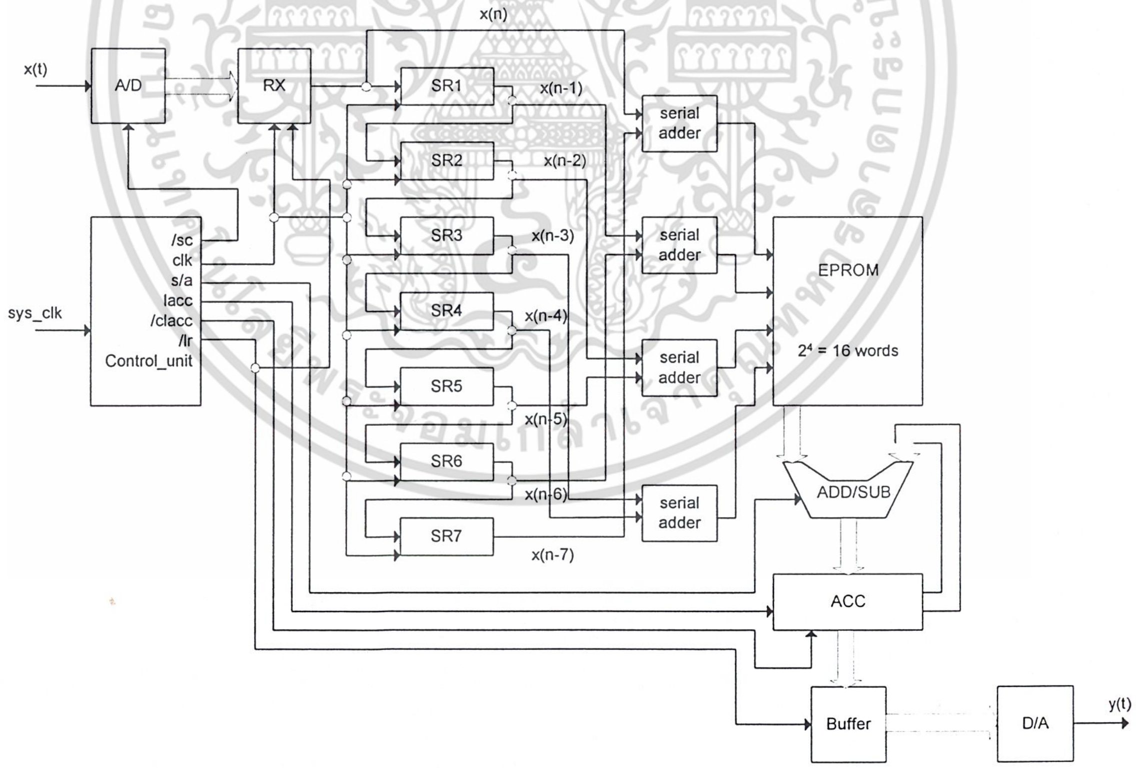
3.5.2.1 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 8

ซึ่งจะใช้หน่วยย่อยต่างๆ ดังนี้

1. รีจีสเตอร์ RX จำนวน 1 ตัว
2. รีจีสเตอร์ SR1, SR2, SR3, SR4, SR5, SR6, SR7 จำนวน 7 ตัว
3. หน่วยความจำ ขนาด $2^4 \times 8$ จำนวน 1 ตัว
4. วงจรบวก/ลบ จำนวน 1 ตัว
5. แอคคิวมูเลเตอร์ จำนวน 1 ตัว
6. บัฟเฟอร์ จำนวน 1 ตัว
7. หน่วยควบคุม จำนวน 1 ตัว
8. วงจรบิทอนุกรม 4 ตัว

สามารถเขียนโครงสร้างของวงจรตามหลักการของโครงสร้างเลขคณิตกระจายได้ ดังนี้

$$Y(n) = b_0 X(n) + b_1 X(n-1) + b_2 X(n-2) + b_3 X(n-3) + b_4 X(n-4) + b_5 X(n-5) + b_6 X(n-6) + b_7 X(n-7)$$



รูปที่ 3.12 โครงสร้างของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.12 สามารถอธิบายการทำงานเป็น 4 ขั้นตอน ดังนี้

1. เมื่อสัญญาณถูกแปลงให้อยู่ในรูปลำดับสัญญาณด้วยวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแล้ว จะถูกโหลดข้อมูลเข้าไปเก็บไว้ในรีจีสเตอร์ RX (ซึ่งถูกควบคุมด้วยสัญญาณ lr)

2. จากนั้นสัญญาณนาฬิกา (clk) จะทำการเลื่อนข้อมูลภายในรีจีสเตอร์ RX, SR1, SR2, SR3, SR4, SR5, SR6 และ SR7 ไปครั้งละ 1 บิท และผลที่ได้จากการเลื่อนข้อมูลแต่ละบิทของรีจีสเตอร์ คือ x(n), x(n-1), x(n-2), x(n-3), x(n-4), x(n-5), x(n-6) และ x(n-7) ตามลำดับ จะไปกำหนดแอดเดรสของ EPROM ซึ่งต้องใช้ EPROM ที่มีแอดเดรสขนาด 8 บิท ทำให้สิ้นเปลือง จึงนำเอาท์พุตจากการเลื่อนข้อมูลผ่านวงจรบิทอนุกรม (โดยจับคู่ตัวที่มีสัมประสิทธิ์เหมือนกัน เช่น x(n) กับ x(n-7), x(n-1), x(n-6) จะได้เอาท์พุตเหลือ 4 บิทไปเป็นแอดเดรสของ EPROM

3. เอาท์พุตจาก EPROM จะถูกส่งไปบวกกับค่าที่อยู่ในแอคคิวมูเลเตอร์ ด้วยวงจรบวก / ลบ (ซึ่งถูกควบคุมด้วยสัญญาณ s/a) ผลลัพธ์ที่ได้จะถูกโหลดเข้าเก็บไว้ในแอคคิวมูเลเตอร์ด้วยสัญญาณ lacc

4. เมื่อสัญญาณ clk เลื่อนข้อมูลในรีจีสเตอร์ไปอีก 1 บิท แล้วกระทำซ้ำข้อ 2 จนกระทั่ง clk เลื่อนข้อมูลไปถึงบิทที่ 8 จึงนำค่าที่ได้จากเอาท์พุทของ EPROM ไปลบออกจากค่าที่อยู่ในแอคคิวมูเลเตอร์ ผลลัพธ์ที่ได้จากการคำนวณของ EPROM จะถูกโหลดเข้าไปเก็บไว้ในบัฟเฟอร์ (ด้วยสัญญาณ /lr) เพื่อทำการแปลงสัญญาณเชิงเลขให้เป็นสัญญาณเชิงอุปมาน ด้วยวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก จากนั้นทำการลบข้อมูลภายในแอคคิวมูเลเตอร์ (ด้วยสัญญาณ /clacc) และจะวนกลับ ไปทำงานซ้ำในขั้นตอนที่ 1, 2, 3, 4 ตามลำดับ

3.5.2.2 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 12

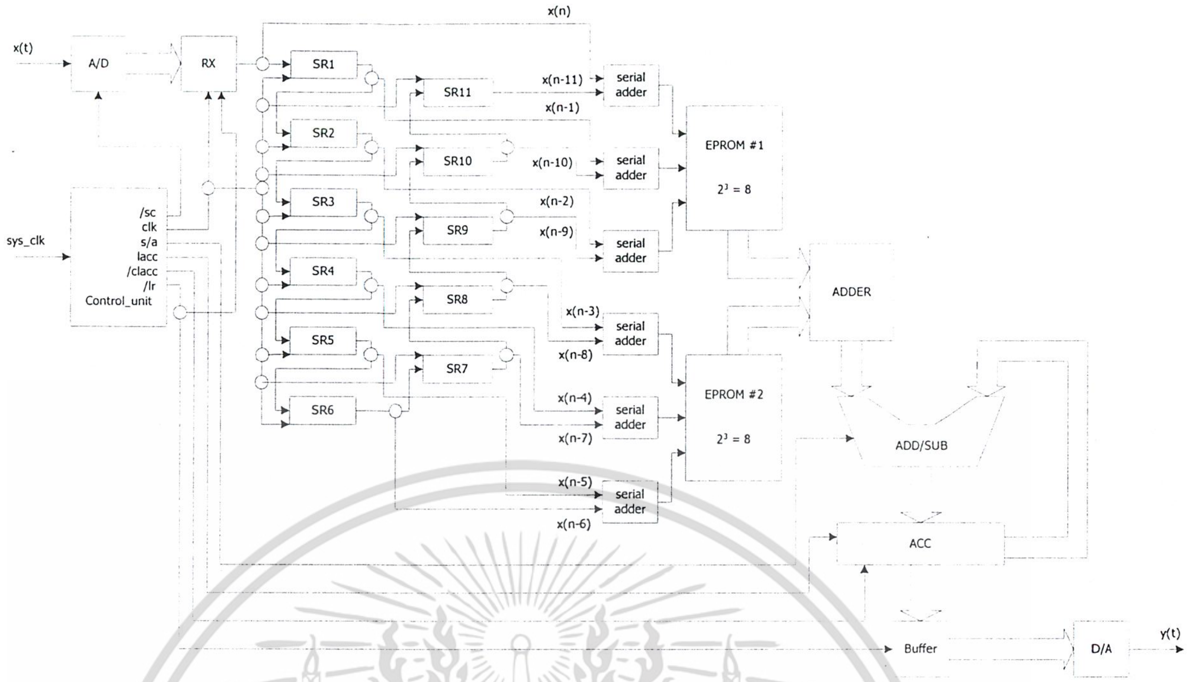
ซึ่งจะใช้หน่วยย่อยต่างๆ ดังนี้

1. รีจีสเตอร์ RX จำนวน 1 ตัว
2. รีจีสเตอร์ SR1, SR2, SR3, SR4, SR5, SR6, SR7, SR8, SR9, SR10, SR11 จำนวน 11 ตัว
3. หน่วยความจำ ขนาด $2^3 \times 8$ จำนวน 2 ตัว
4. วงจรบวก/ลบ จำนวน 1 ตัว
5. แอคคิวมูเลเตอร์ จำนวน 1 ตัว
6. บัฟเฟอร์ จำนวน 1 ตัว
7. หน่วยควบคุม จำนวน 1 ตัว
8. วงจรบิทอนุกรม 6 ตัว
9. วงจรบวกขนาด 8 บิท จำนวน 1 ตัว

สามารถเขียนโครงสร้างของวงจรตามหลักการของโครงสร้างเลขคณิตกระจายได้ ดังนี้

$$\begin{aligned}Y(n) = {} & b_0 X(n) + b_1 X(n-1) + b_2 X(n-2) + b_3 X(n-3) \\ & + b_4 X(n-4) + b_5 X(n-5) + b_6 X(n-6) + b_7 X(n-7) \\ & + b_8 X(n-8) + b_9 X(n-9) + b_{10} X(n-10) + b_{11} X(n-11)\end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 โครงสร้างของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 12

3.5.2.3 วงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 16

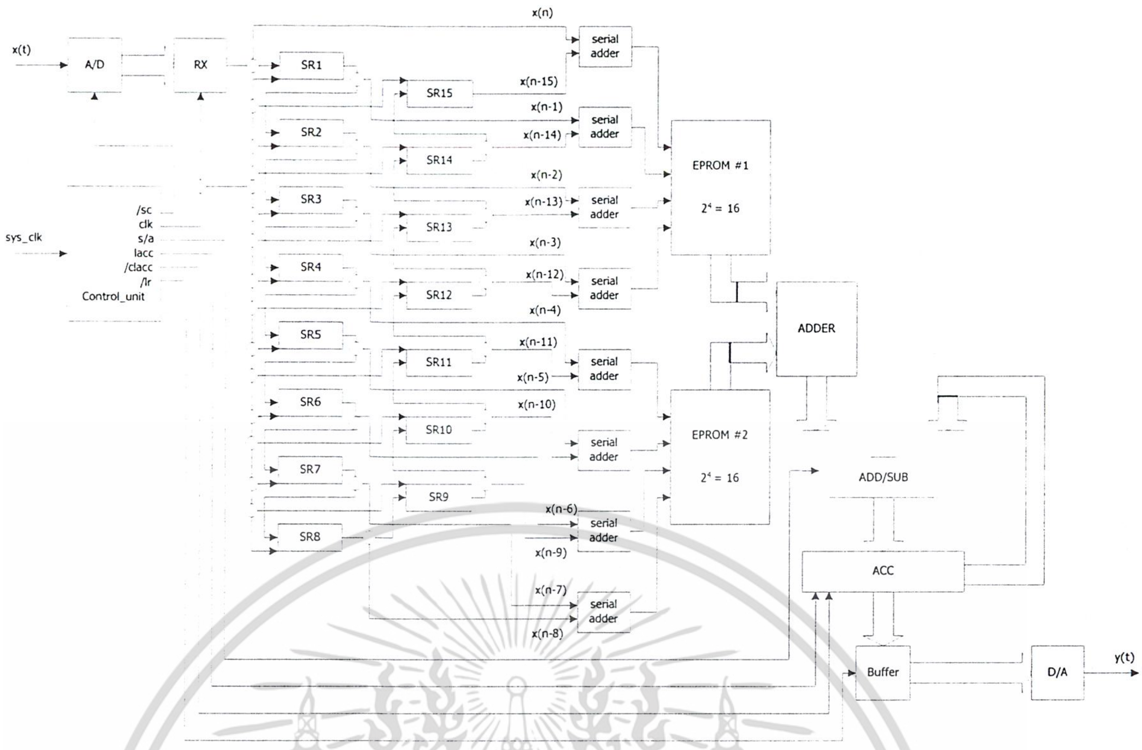
ซึ่งจะใช้หน่วยย่อยต่างๆ ดังนี้

10. รีจีสเตอร์ RX จำนวน 1 ตัว
11. รีจีสเตอร์ SR1, SR2, SR3, SR4, SR5, SR6, SR7, SR8, SR9, SR10, SR11, SR12, SR13, SR14, SR15 จำนวน 15 ตัว
12. หน่วยความจำ ขนาด $2^4 \times 8$ จำนวน 2 ตัว
13. วงจรบวก/ลบ จำนวน 1 ตัว
14. แอคคิวมูเลเตอร์ จำนวน 1 ตัว
15. บัฟเฟอร์ จำนวน 1 ตัว
16. หน่วยควบคุม จำนวน 1 ตัว
17. วงจรบิทอนุกรม 8 ตัว
18. วงจรบวกขนาด 8 บิท จำนวน 1 ตัว

สามารถเขียนโครงสร้างของวงจรตามหลักการของโครงสร้างเลขคณิตกระจายได้ ดังนี้

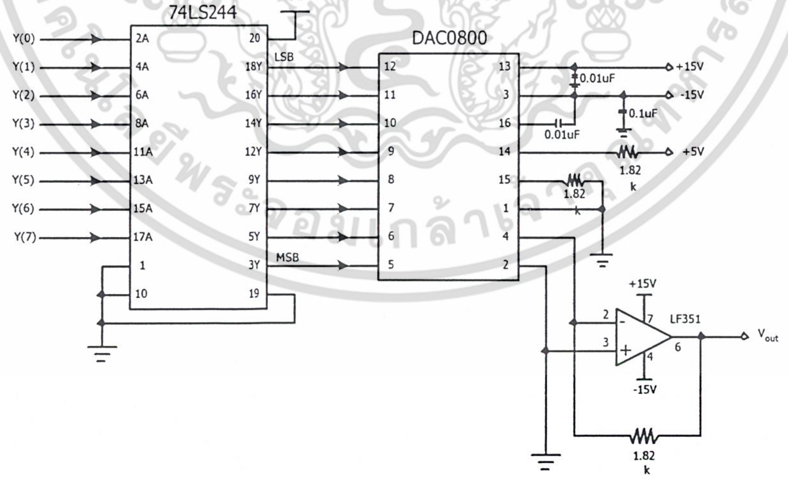
$$\begin{aligned} Y(n) = {} & b_0 X(n) + b_1 X(n-1) + b_2 X(n-2) + b_3 X(n-3) \\ & + b_4 X(n-4) + b_5 X(n-5) + b_6 X(n-6) + b_7 X(n-7) \\ & + b_8 X(n-8) + b_9 X(n-9) + b_{10} X(n-10) + b_{11} X(n-11) \\ & + b_{12} X(n-12) + b_{13} X(n-13) + b_{14} X(n-14) + b_{15} X(n-15) \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 โครงสร้างของวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 16

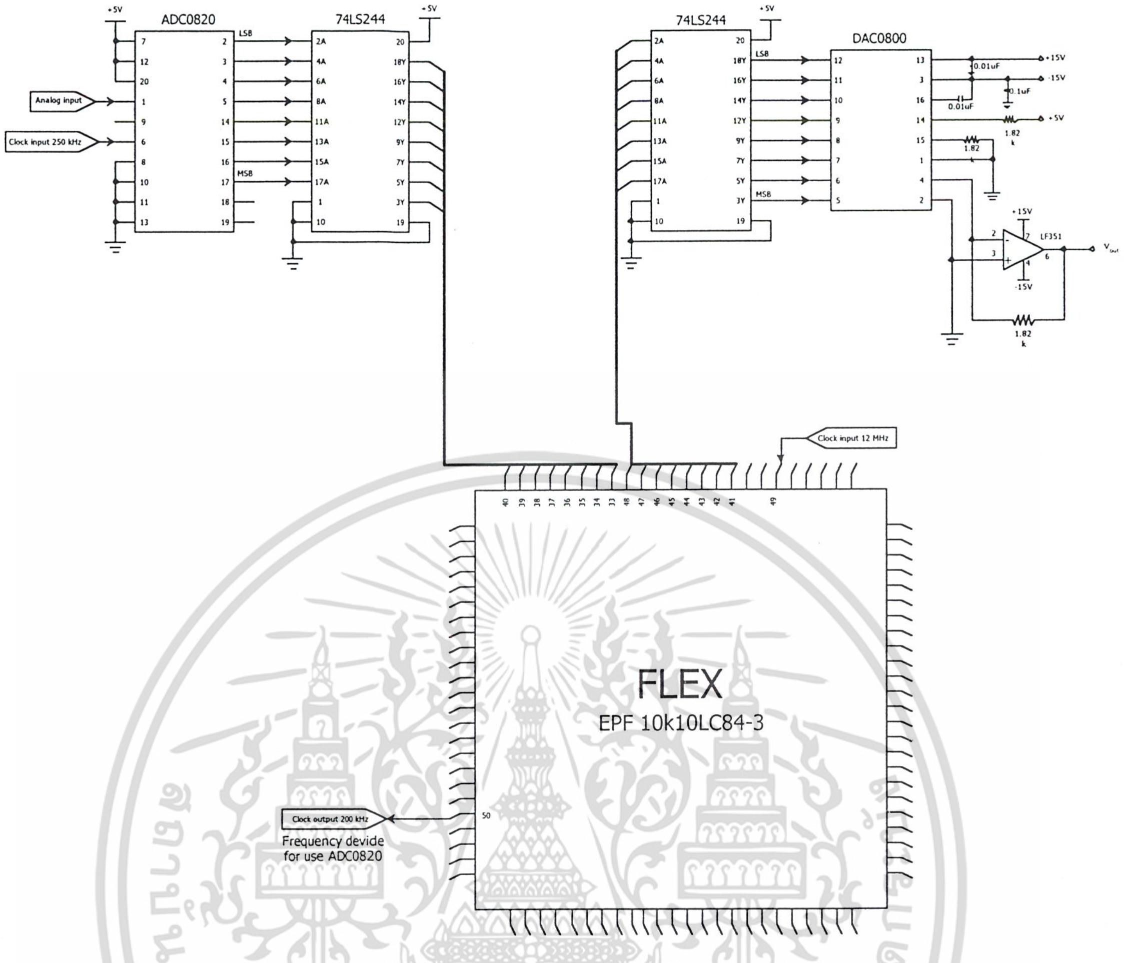
3.5.3 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก (D/A Converter)
จะทำการแปลงสัญญาณดิจิตอลกลับไปเป็นสัญญาณอนาลอกเช่นเดิม



รูปที่ 3.15 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก

เมื่อนำวงจรมาต่อรวมจะได้ดังรูปที่ 3.16 ซึ่งชิปต้นแบบที่ผ่านการโหลดโปรแกรมแล้วจะทำงานเสมือนวงจรกรองสัญญาณไม่ป้อนกลับเชิงเลขอันดับที่ 8, 12, 16 ที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 วงจรรวมกับ FPGA ที่ลงโปรแกรมแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

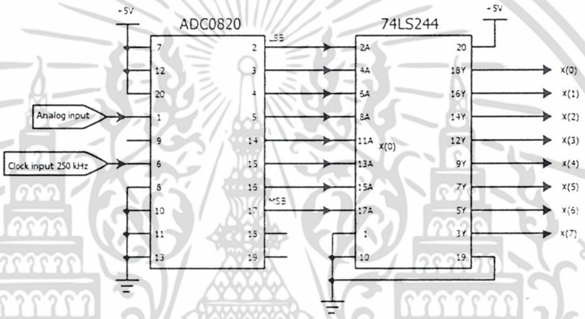
การทดลอง และผลการทดลอง

บทนี้จะกล่าวถึงการทดลองและผลการทดลองที่ได้จากการทำงานของวงจรกรองความถี่แบบดิจิตอลชนิด FIR การทำงานของวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล การทำงานของวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก

4.1 การทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

4.1.1 ขั้นตอนการทดลอง

1. ต่อวงจรตามรูปที่ 4.1
2. ต่อแหล่งจ่ายไฟที่สามารถปรับค่าได้ตั้งแต่ –2.5 โวลต์ จนถึง +2.5 โวลต์
3. ทำการปรับแหล่งจ่ายไฟตามค่าที่กำหนดในตารางที่ 4.1
4. วัดเอาท์พุตที่ได้จากผลการทดลอง



รูปที่ 4.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

4.1.2 ผลการทดลอง

อินพุต (โวลต์)	เอาท์พุต
-2.5	0000 0000
-2.0	0001 1001
-1.5	0011 0011
-1.0	0100 1100
-0.5	1100 0110
0	0111 1111
+0.5	1001 1001
+1.0	1011 0010
+1.5	1100 1000
+2.0	1110 1111
+2.5	1111 1111

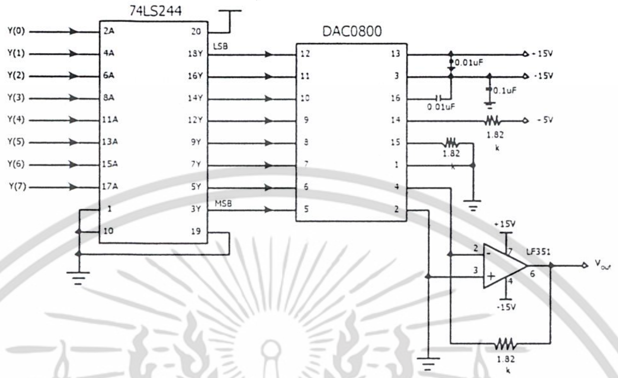
ตารางที่ 4.1 ผลการทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

4.2.1 ขั้นตอนการทดลอง

1. ต่อวงจรตามรูปที่ 4.2
2. ต่อแหล่งจ่ายไฟที่สามารถปรับค่าได้ตั้งแต่ ±2.5 โวลต์ให้กับวงจร D/A
3. ปรับค่าอินพุตตามค่าที่กำหนดในตารางที่ 4.1 ให้วงจร D/A
4. วัดเอาท์พุตที่ได้จากผลการทดลอง



รูปที่ 4.2 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก

4.2.2 ผลการทดลอง

อินพุต	เอาท์พุต (โวลต์)
0000 0000	1.2
0010 1111	1.7
0100 1111	2
0110 1111	2.3
1000 1111	0.1
1010 1111	0.5
1100 1111	1.1
1111 1111	1.3

ตารางที่ 4.2 ผลการทดลองวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองการทำงานของ FPGA

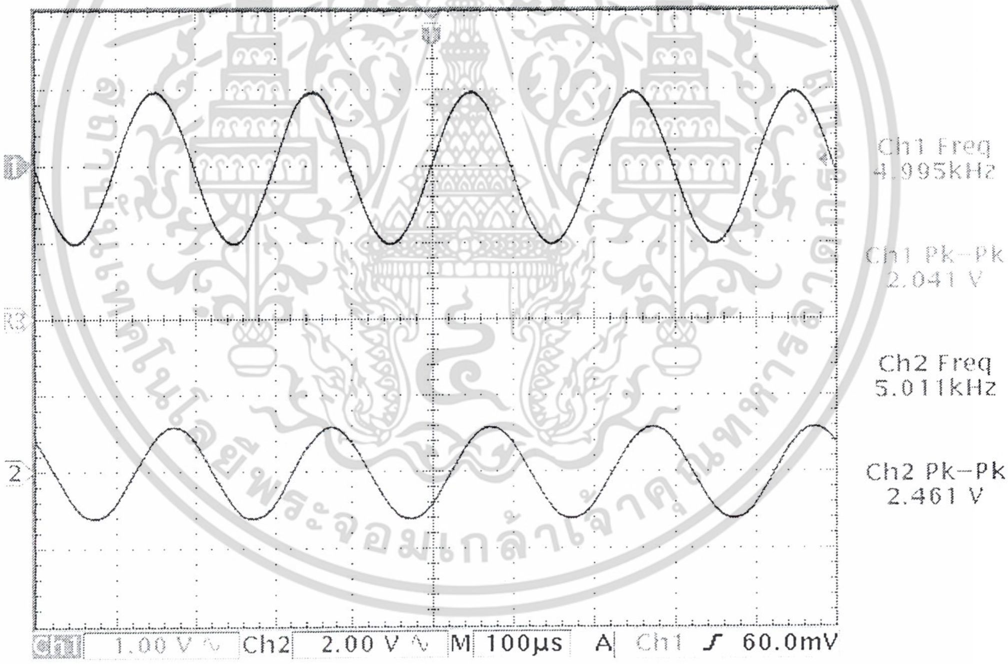
4.3.1 ขั้นตอนการทดลอง

1. ต่อวงจรรวมเข้าด้วยกัน
2. โหลดโปรแกรมลงบน FPGA โดยใช้โปรแกรม Max + Plus II
3. ป้อนสัญญาณอินพุตที่แรงดันเท่ากับ 2 Vp-p และที่ความถี่ต่างๆกัน
4. ใช้ออสซิลโลสโคปวัดสัญญาณอินพุตและเอาท์พุต โดยที่ Ch1 เป็นอินพุต และ Ch2 เป็นเอาท์พุต แล้วบันทึกผลการทดลอง
5. หลังจากนั้นนำค่าแรงดันเอาท์พุต ใส่ตารางแล้วนำมาพล็อตกราฟผลตอบสนองทางความถี่
6. วัดเฟสของสัญญาณอินพุตและเอาท์พุต แล้วนำมาพล็อตกราฟ

4.3.2 ผลการทดลอง

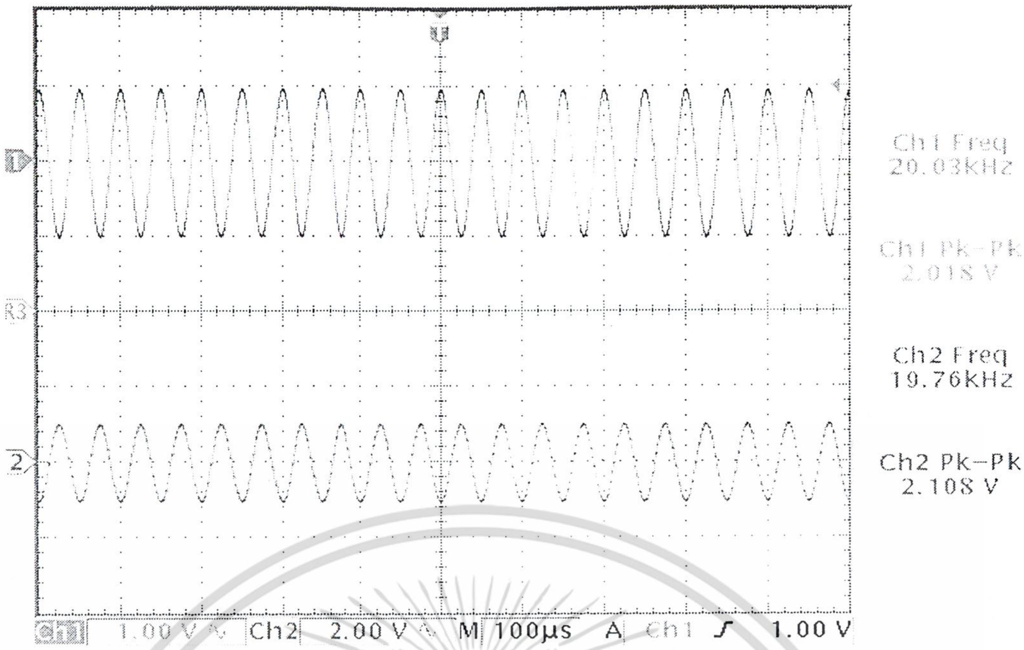
4.3.2.1 วงจรกรองความถี่ต่ำผ่านอันดับที่ 8

- ความถี่คัทออฟ (ω_c) เท่ากับ 20 กิโลเฮิร์ต
- ความถี่ในการสุ่ม (ω_s) เท่ากับ 250 กิโลเฮิร์ต

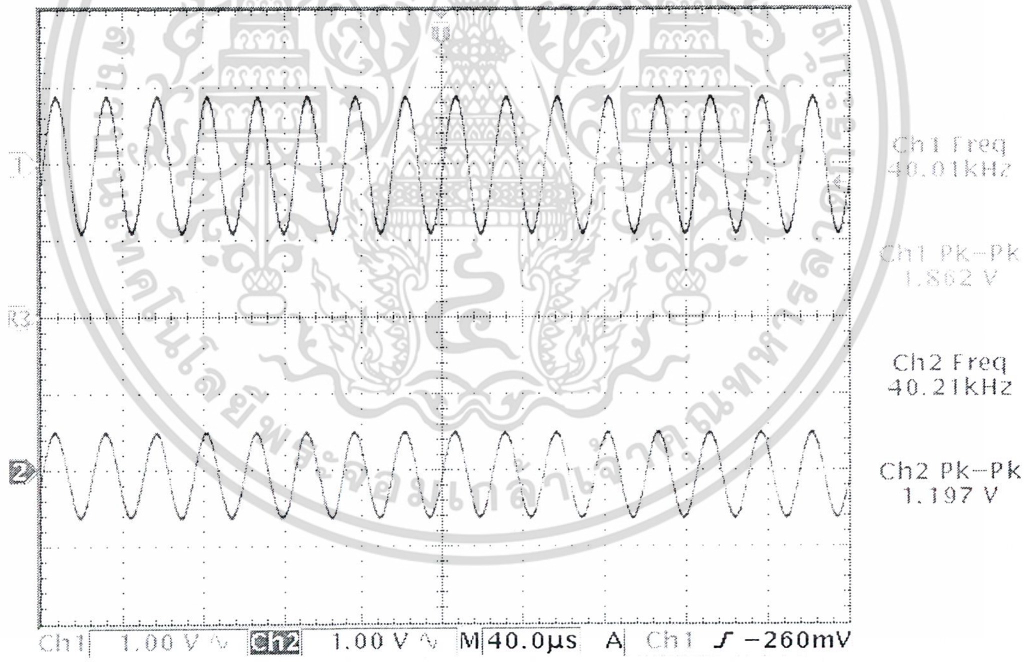


รูปที่ 4.3 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 5 กิโลเฮิร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

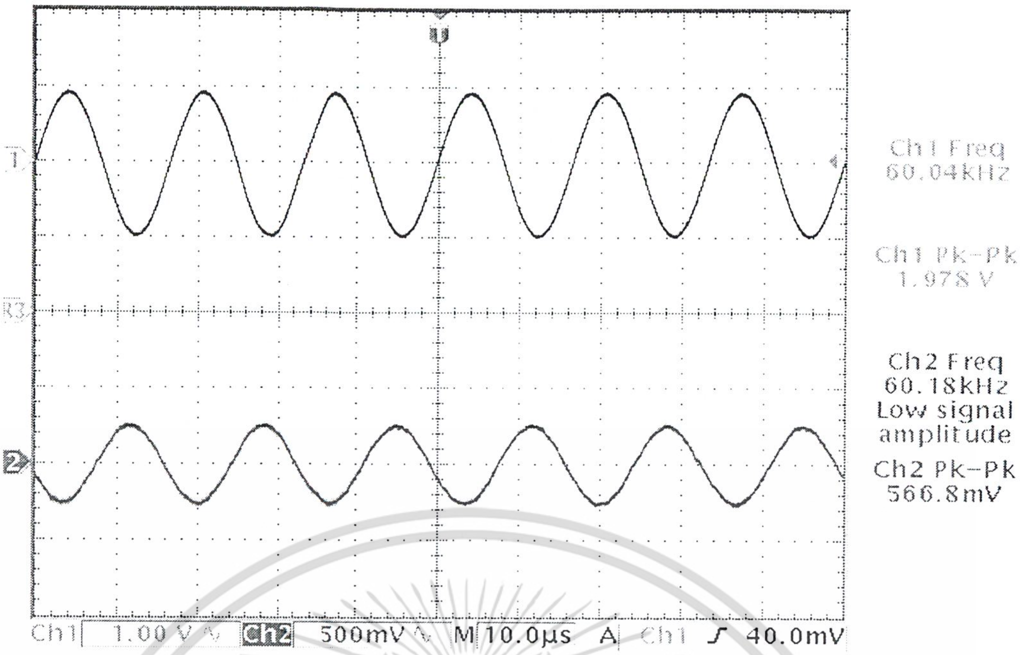


รูปที่ 4.4 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 20 กิโลเฮิร์ต



รูปที่ 4.5 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 40 กิโลเฮิร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



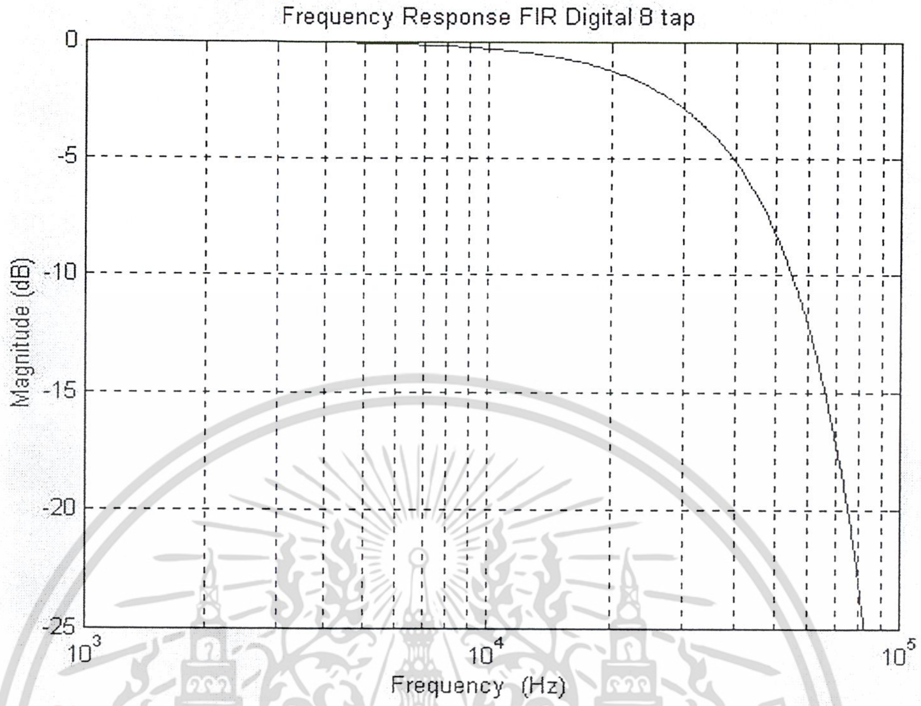
รูปที่ 4.6 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 60 กิโลเฮิร์ต

ความถี่ (kHz)	ค่าแรงดันเอาท์พุต (Vp-p)	ความต่างเฟส(องศา)
1	2.521	-14.4
5	2.461	-76.2
10	2.357	-142.2
15	2.285	-223.1
17	2.224	-248.0
19	2.139	-282.0
20	2.108	-300.8
22	2.040	-328.8
25	1.951	-370.4
30	1.698	-448.4
35	1.496	-524.2
40	1.297	-601.4
50	9.890m	-752.6
60	566.8m	-902.4
70	353.9m	-1045.5
80	203.2m	-1185.2

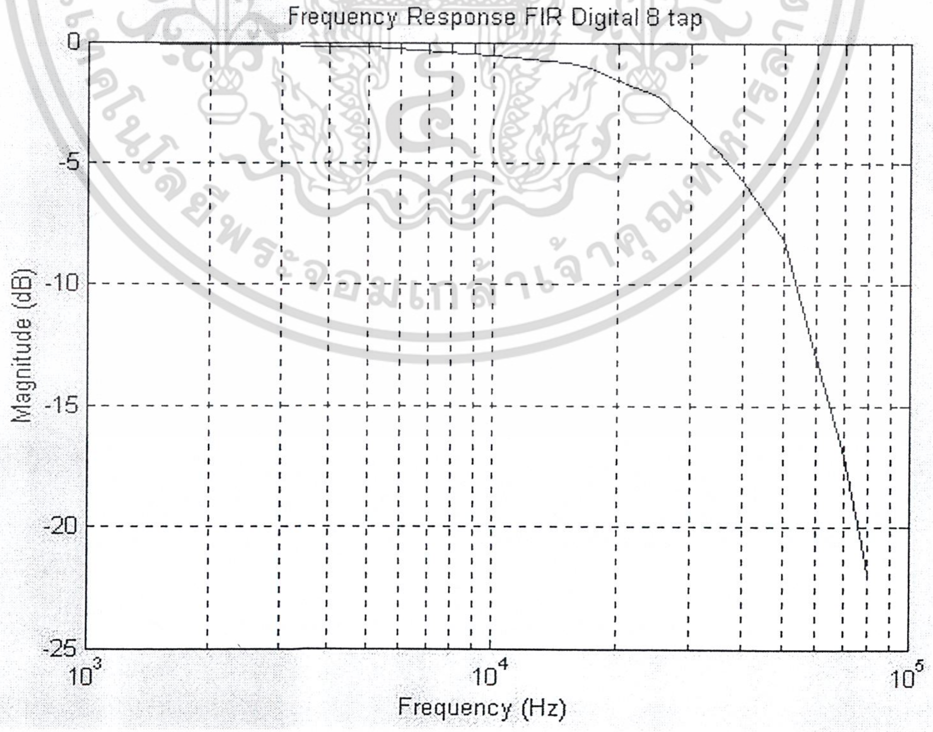
ตารางที่ 4.3 ผลการทดลองของวงจรกรองความถี่ต่ำผ่านอันดับที่ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบกราฟแสดงผลตอบสนองทางความถี่ระหว่างผลที่ได้จาก MATLAB กับผลจากการทดลอง



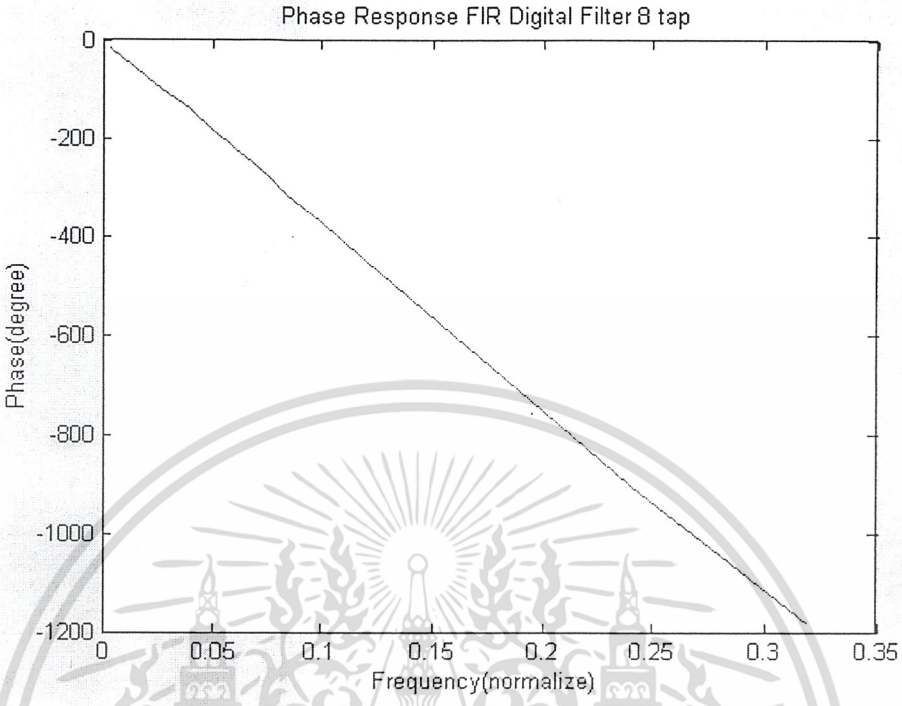
รูปที่ 4.7 กราฟแสดงผลตอบสนองทางความถี่จากผล MATLAB



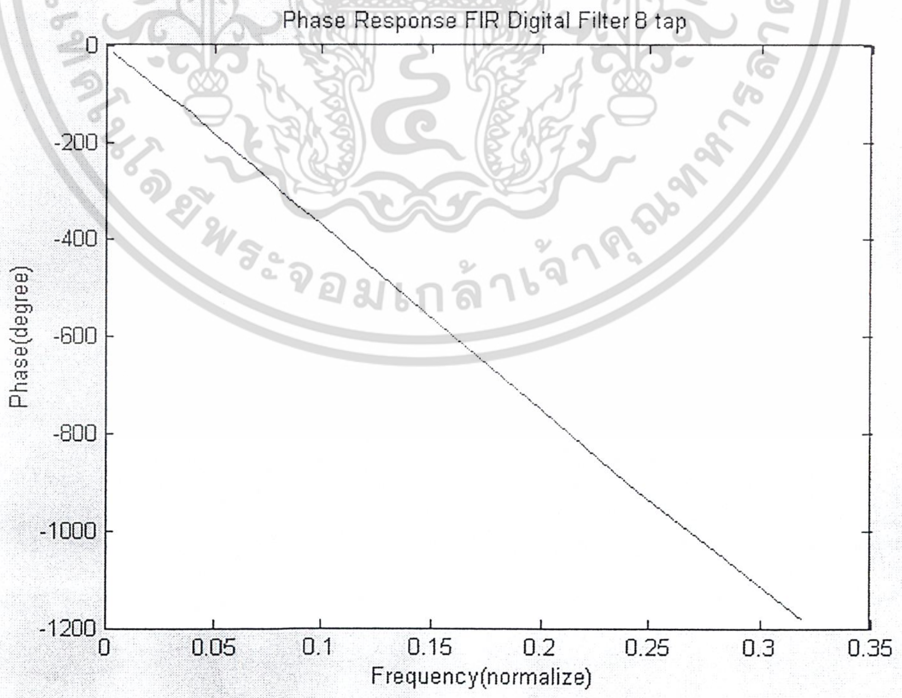
รูปที่ 4.8 กราฟแสดงผลตอบสนองทางความถี่จากผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบกราฟแสดงผลตอบสนองทางเฟสระหว่างผลที่ได้จาก MATLAB กับผลจากการทดลอง



รูปที่ 4.9 กราฟแสดงผลตอบสนองทางเฟสจากผล MATLAB

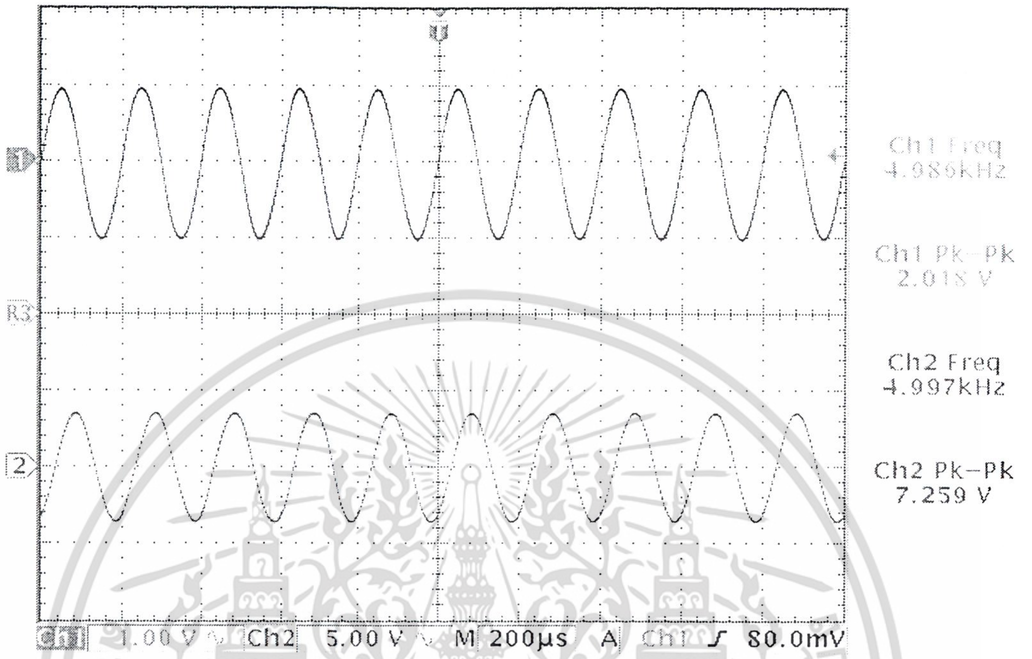


รูปที่ 4.10 กราฟแสดงผลตอบสนองทางเฟสจากผลการทดลอง

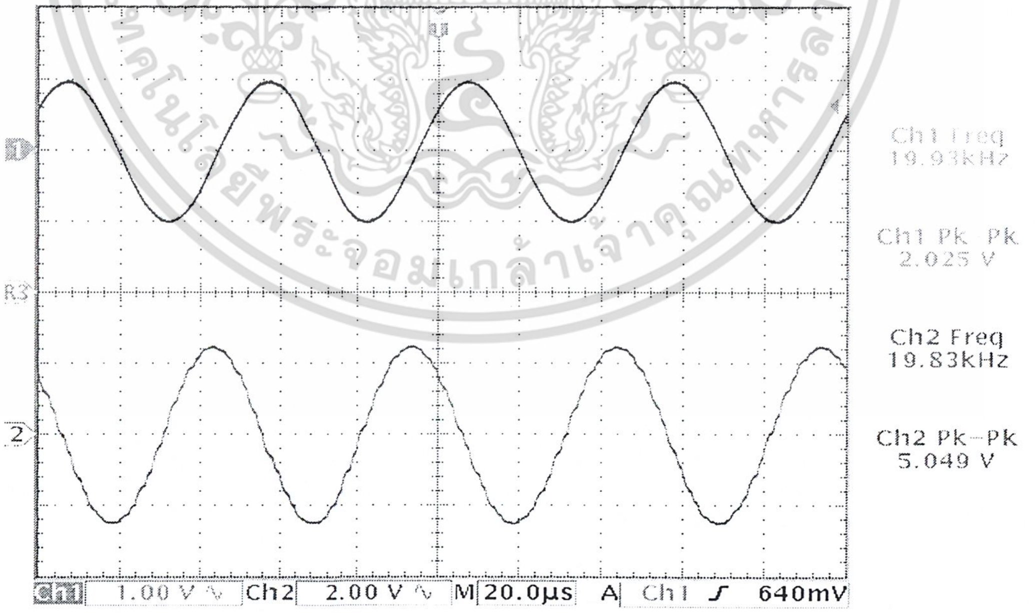
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2.2 วงจรกรองความถี่ต่ำผ่านอันดับที่ 12

- ความถี่คัทออฟ (ω_c) เท่ากับ 20 กิโลเฮิร์ต
- ความถี่ในการสุ่ม (ω_s) เท่ากับ 250 กิโลเฮิร์ต

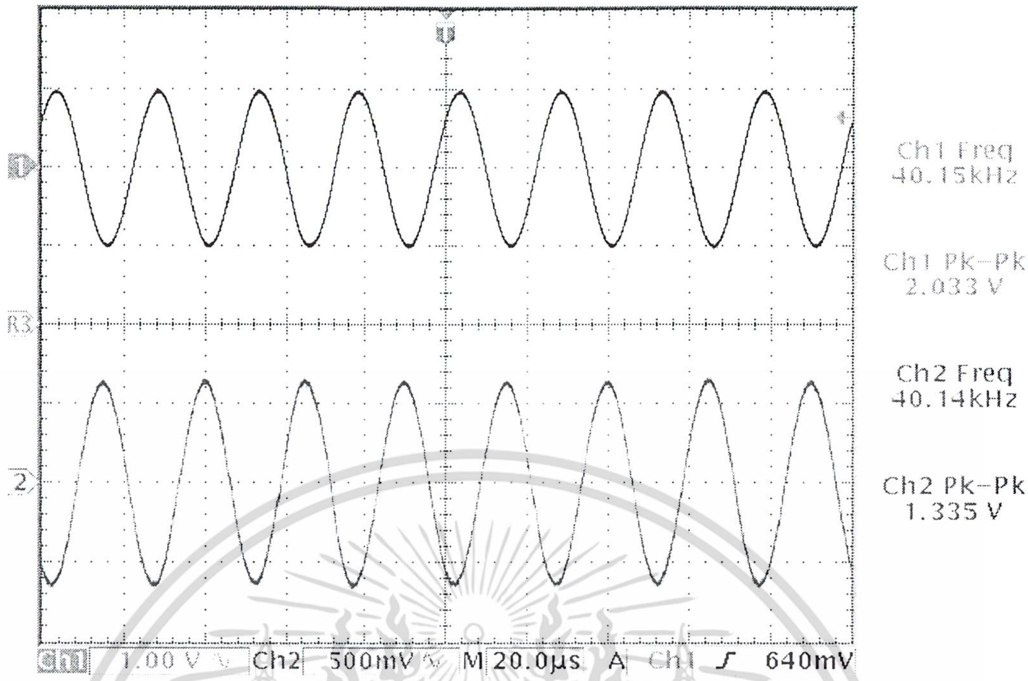


รูปที่ 4.11 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 5 กิโลเฮิร์ต

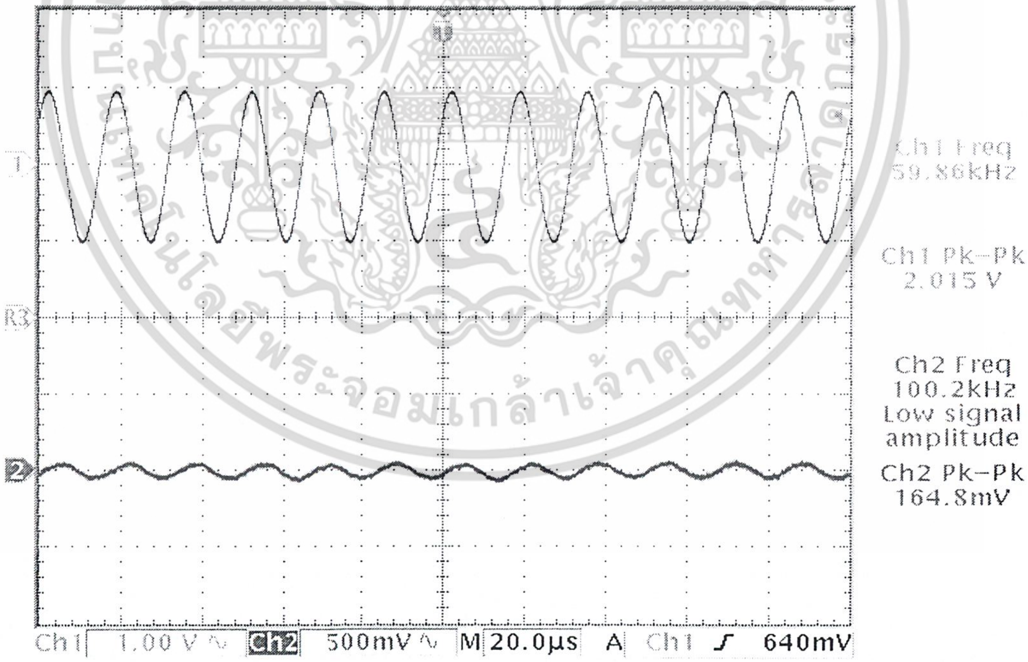


รูปที่ 4.12 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 20 กิโลเฮิร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 40 กิโลเฮิร์ต



รูปที่ 4.14 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 60 กิโลเฮิร์ต

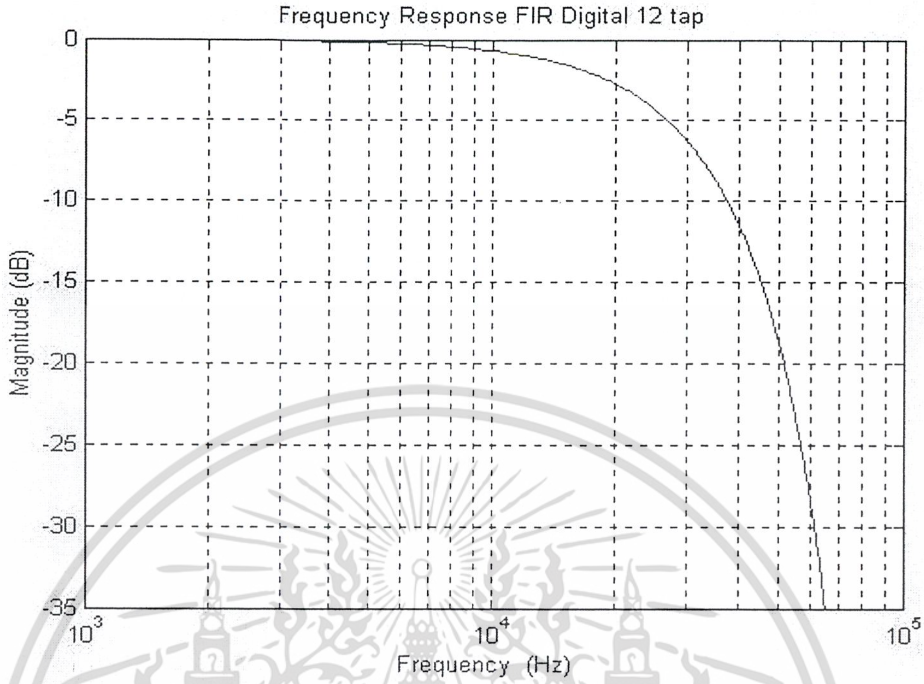
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่ (kHz)	ค่าแรงดันเอาท์พุต (Vp-p)	ความต่างเฟส(องศา)
1	7.689	-15.5
5	7.259	-78.9
10	6.737	-152.2
15	5.963	-229.3
17	5.644	-260.0
19	5.277	-288.8
20	5.049	-310.1
25	3.979	-376.5
30	2.978	-444.3
35	2.069	-525.5
40	1.335	-604.7
45	806.8m	-677.3
50	437.8m	-755.6
60	164.8m	-902.0

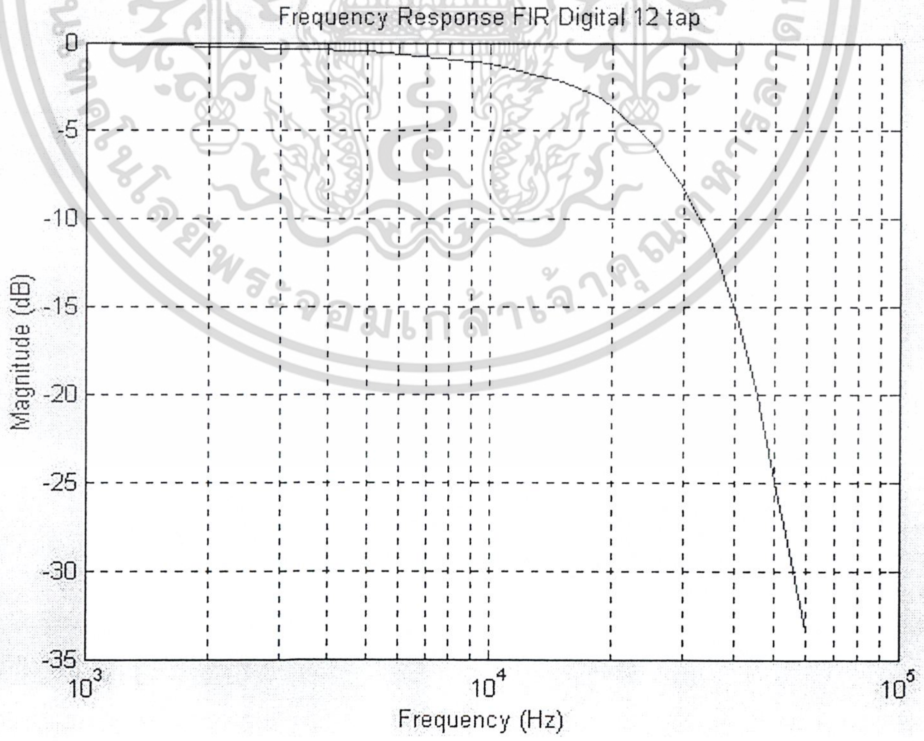
ตารางที่ 4.4 ผลการทดลองของวงจรกรองความถี่ต่ำผ่านอันดับที่ 12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบกราฟแสดงผลตอบสนองทางความถี่ระหว่างผลที่ได้จาก MATLAB กับผลจากการทดลอง



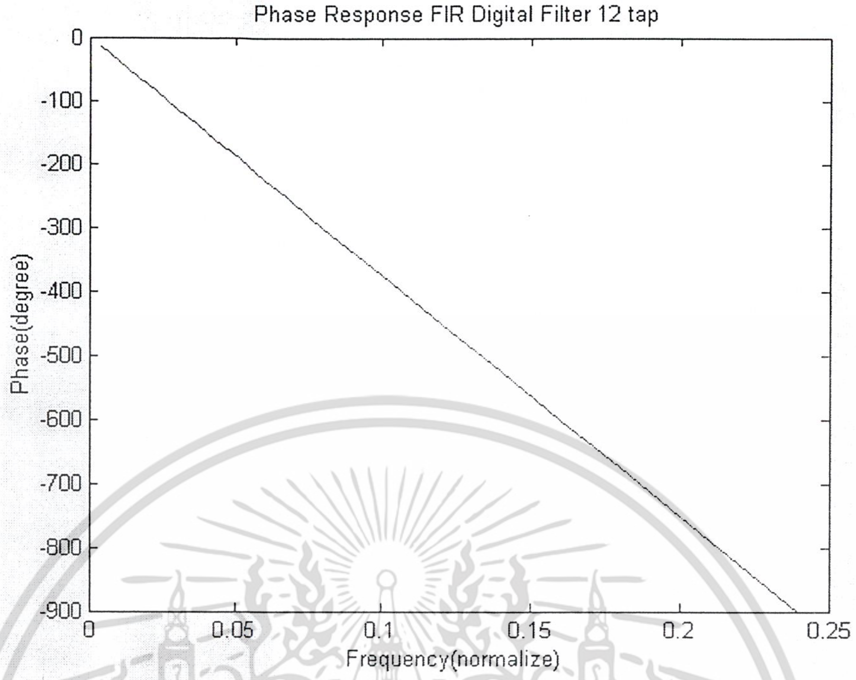
รูปที่ 4.15 กราฟแสดงผลตอบสนองทางความถี่จากผล MATLAB



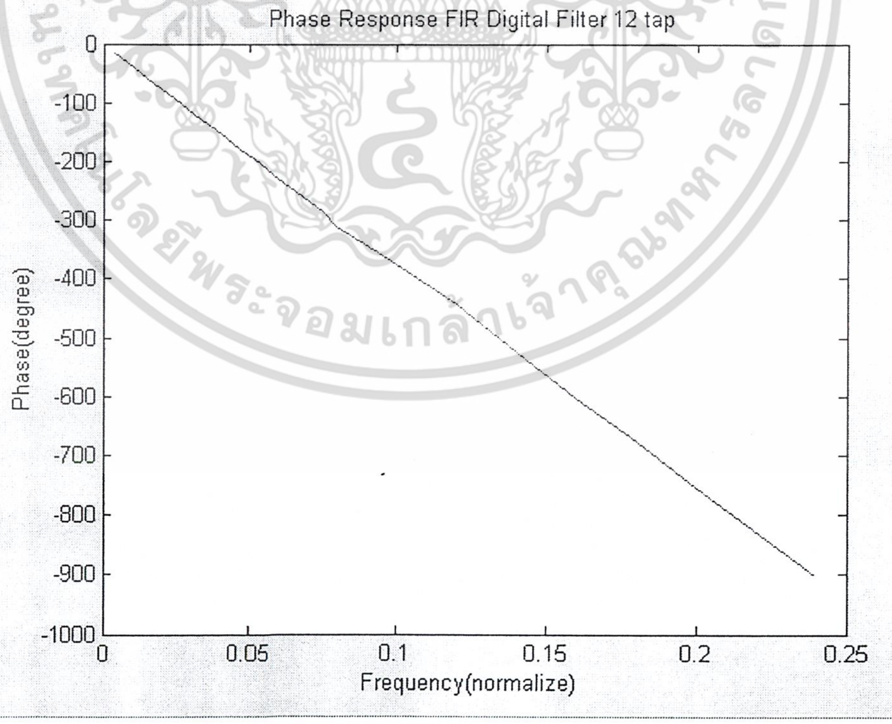
รูปที่ 4.16กราฟแสดงผลตอบสนองทางความถี่จากผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบกราฟแสดงผลตอบสนองทางเฟสระหว่างผลที่ได้จาก MATLAB กับผลจากการทดลอง



รูปที่ 4.17 กราฟแสดงผลตอบสนองทางเฟสจากผล MATLAB

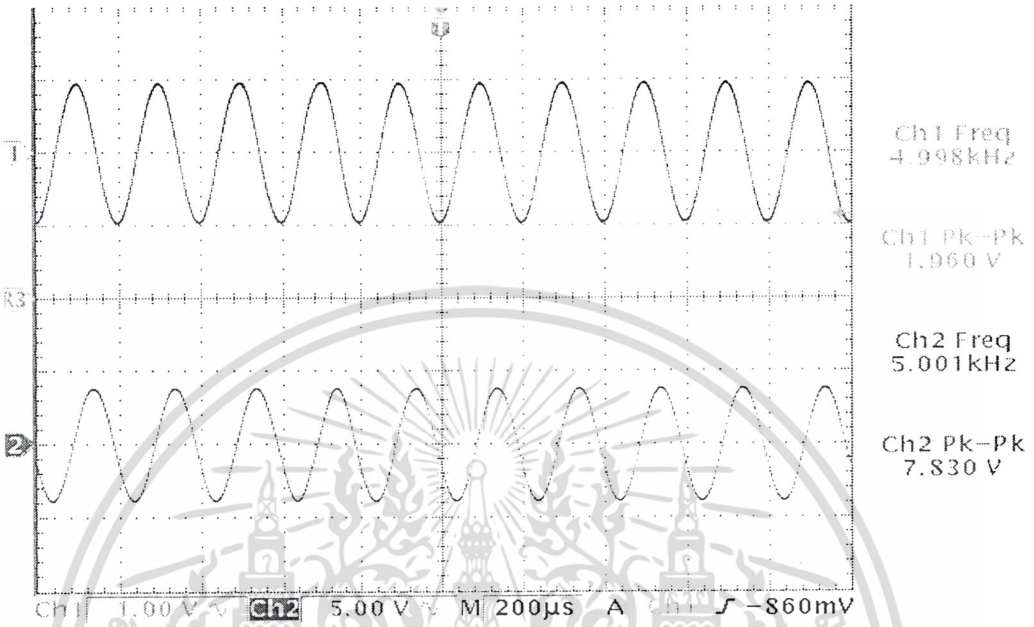


รูปที่ 4.18 กราฟแสดงผลตอบสนองทางเฟสจากผลการทดลอง

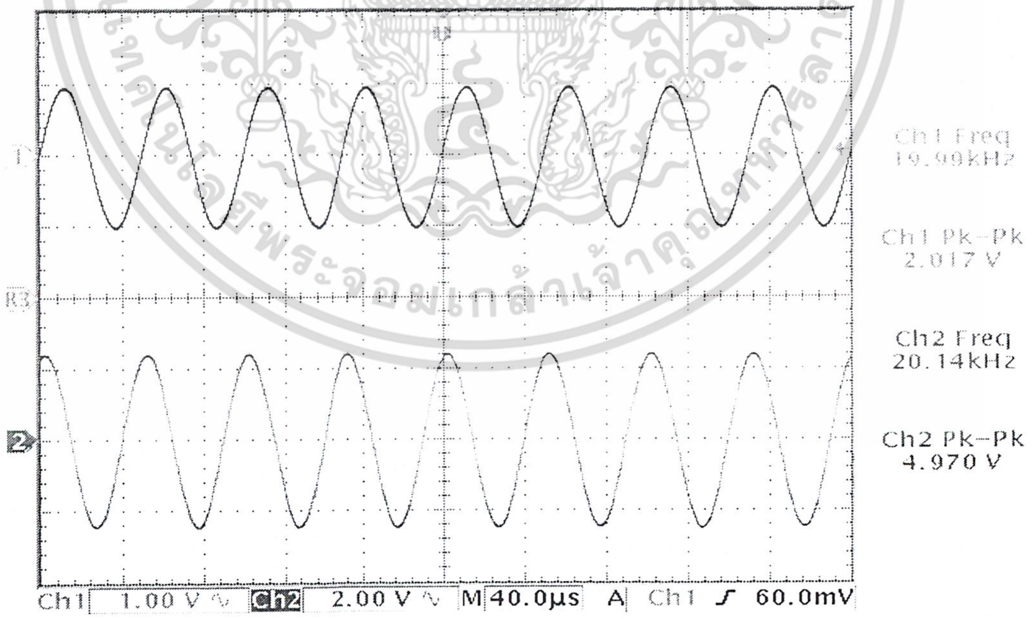
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2.3 วงจรกรองความถี่ต่ำผ่านอันดับที่ 16

- ความถี่คัทออฟ (ω_c) เท่ากับ 20 กิโลเฮิร์ต
- ความถี่ในการสุ่ม (ω_s) เท่ากับ 250 กิโลเฮิร์ต

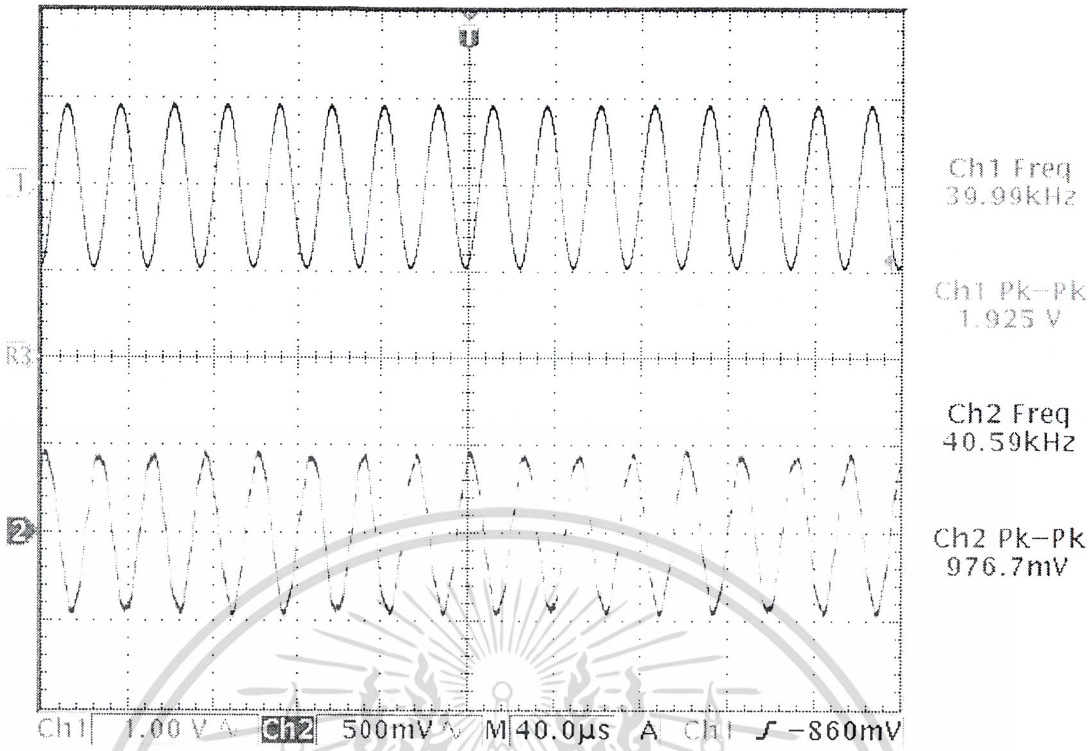


รูปที่ 4.19 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 5 กิโลเฮิร์ต

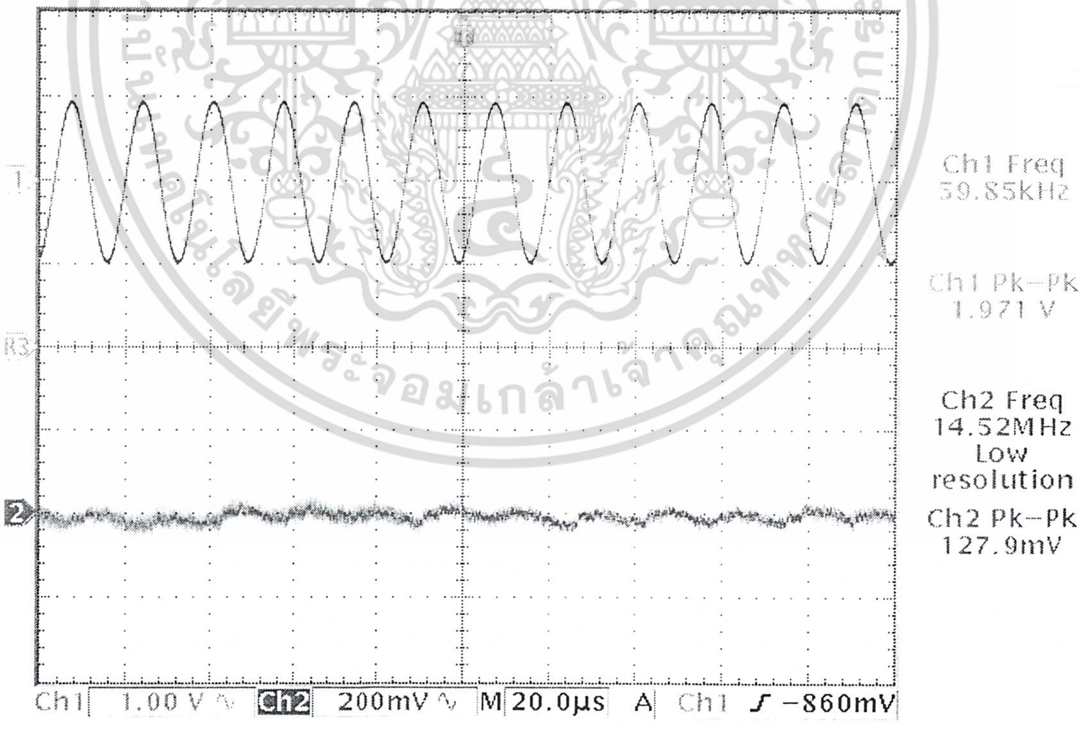


รูปที่ 4.20 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 20 กิโลเฮิร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 40 กิโลเฮิร์ต



รูปที่ 4.22 แสดงสัญญาณเอาท์พุตเมื่อป้อนสัญญาณอินพุตที่แรงดัน 2 Vp-p และความถี่ 60 กิโลเฮิร์ต

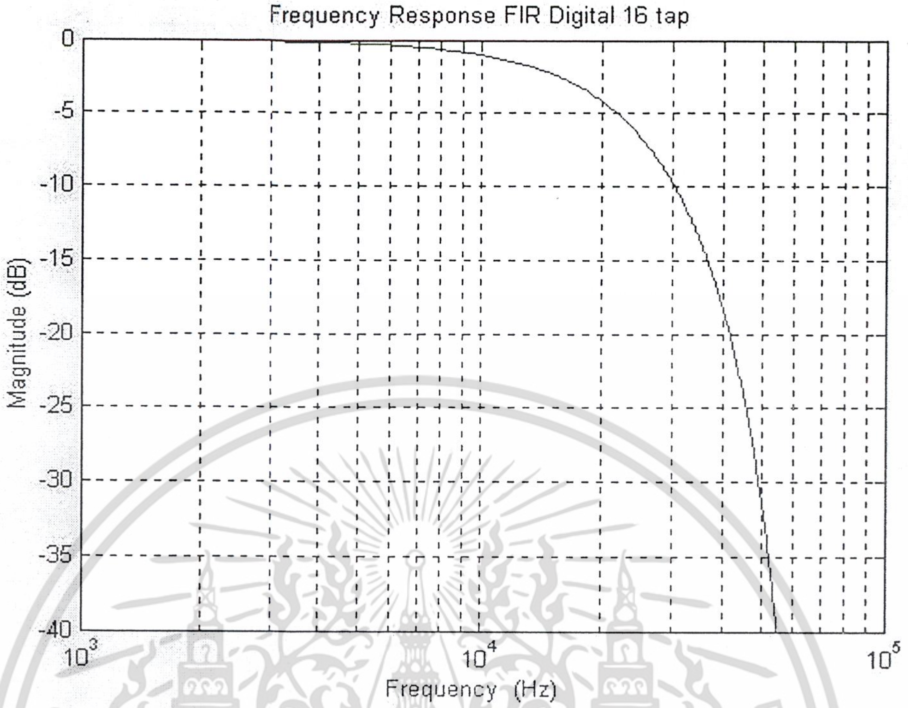
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่ (kHz)	ค่าแรงดันเอาท์พุต (Vp-p)	ความต่างเฟส(องศา)
1	8.005	-14.4
5	7.830	-62.0
10	6.966	-144.0
15	6.164	-215.9
16	5.732	-246.9
17	5.549	-257.7
18	5.347	-273.6
19	5.031	-280.8
20	4.970	-300.9
22	4.397	-326.0
25	3.678	-360.8
30	2.552	-444.1
35	1.644	-514.3
40	976.7m	-596.0
45	516.5m	-664.2
50	263.8m	-720.0
55	146.5m	-819.2
60	127.9m	-898.3

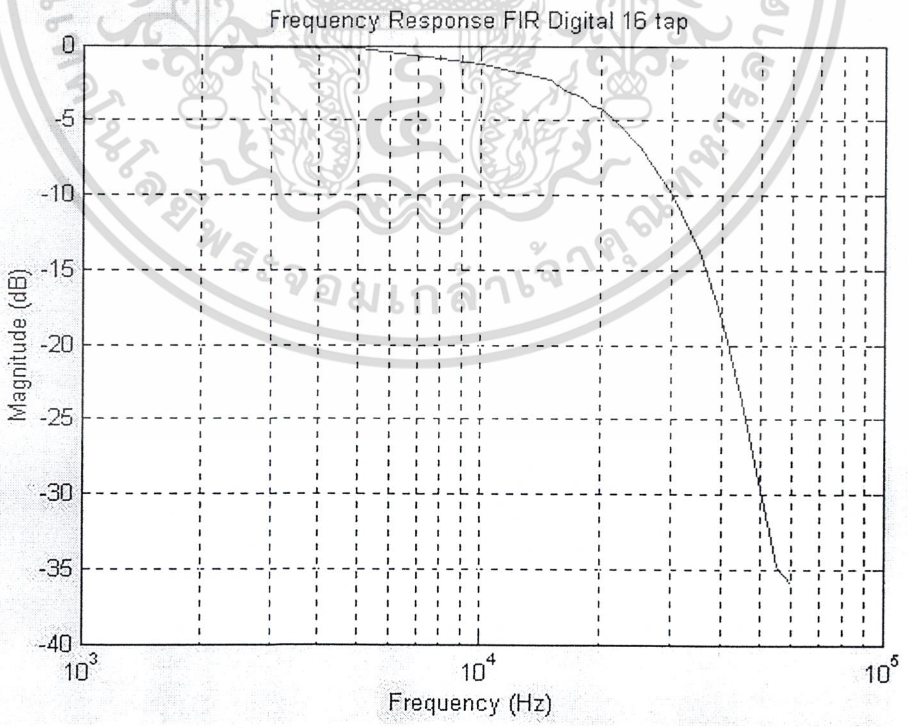
ตารางที่ 4.5 ผลการทดลองของวงจรกรองความถี่ต่ำผ่านอันดับที่ 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบกราฟแสดงผลตอบสนองทางความถี่ระหว่างผลที่ได้จาก MATLAB กับผลจากการทดลอง



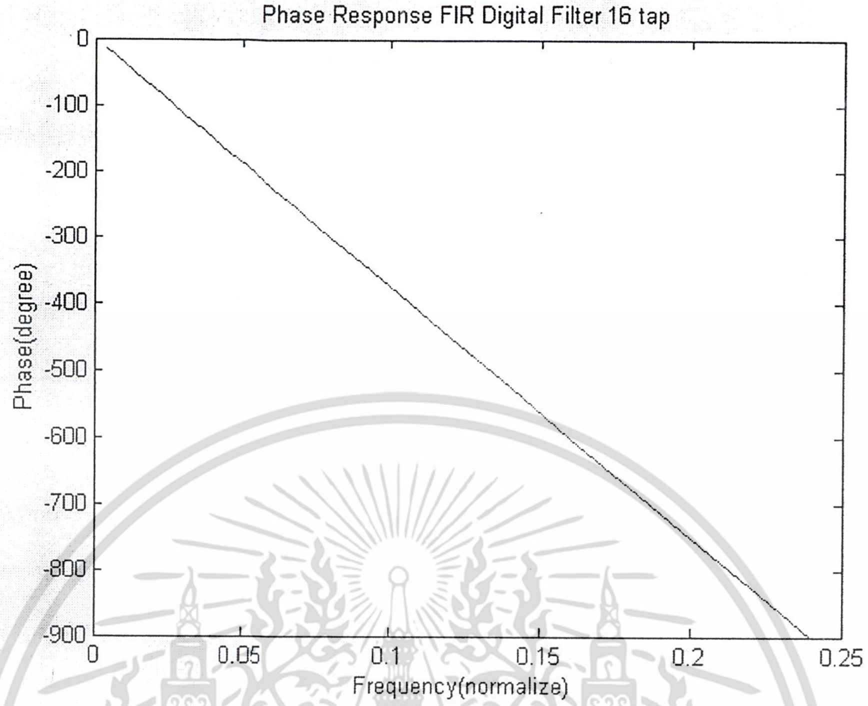
รูปที่ 4.23 กราฟแสดงผลตอบสนองทางความถี่จากผล MATLAB



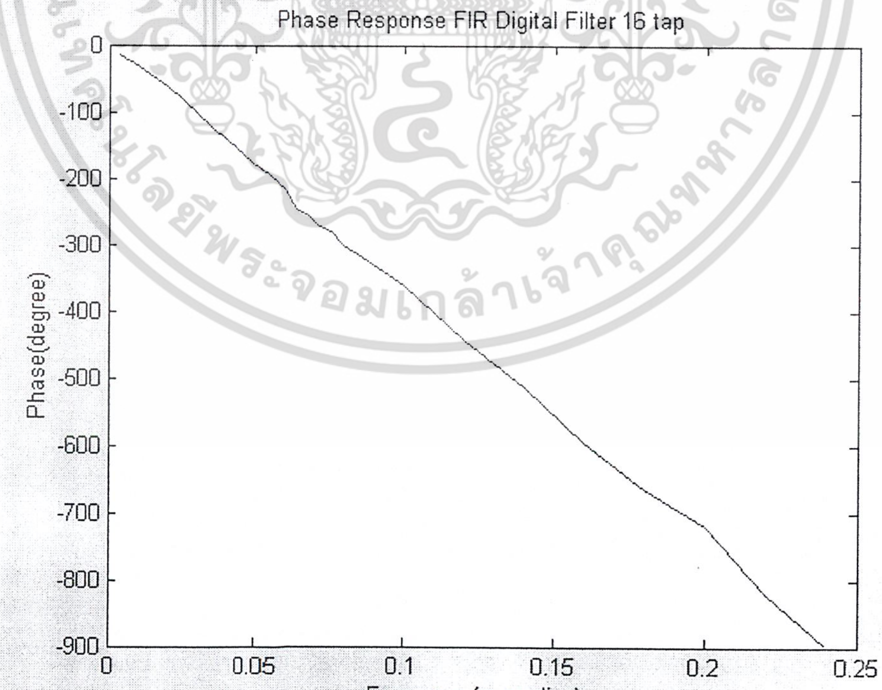
รูปที่ 4.24 กราฟแสดงผลตอบสนองทางความถี่จากผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบกราฟแสดงผลตอบสนองทางเฟสระหว่างผลที่ได้จาก MATLAB กับผลจากการทดลอง



รูปที่ 4.25 กราฟแสดงผลตอบสนองทางเฟสจากผล MATLAB



รูปที่ 4.26 กราฟแสดงผลตอบสนองทางเฟสจากผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

โครงงานนี้เป็นการศึกษาระบบการประมวลผลสัญญาณดิจิตอล เพื่อออกแบบตัวกรองความถี่ดิจิตอลแบบ FIR หรือตัวกรองไม่ป้อนกลับเชิงเลข โดยใช้โปรแกรม MATLAB ช่วยในการออกแบบและใช้ ภาษา VHDL ในการสร้างวงจรกรองเชิงเลขอันดับที่ 8, 12 และ 16 ผลที่ได้จากการทดลองและผลที่ไดจากโปรแกรม MATLAB มีค่าใกล้เคียงกันและพบว่าการทำงานของวงจรที่ได้ออกแบบไว้ทำงานได้ถูกต้องตามที่ได้ออกแบบไว้ที่อันดับ 16 ส่วนการทำงานของวงจรที่อันดับ 8 และ 12 การทำงานของวงจรที่ได้ออกแบบไว้ไม่ตรงไปตามที่ได้ออกแบบไว้ทั้งผลการทดลองและผลจากโปรแกรม MATLAB ซึ่งเป็นไปตามทฤษฎีของวงจรกรองเชิงเลขแบบจำกัดที่มีข้อเสียที่จะใช้วงจรกรองที่มีอันดับสูงถึงแม้จะต้องการให้มีลักษณะทางความถี่ที่ง่ายๆก็ตาม

โดยขั้นตอนในการสร้างโดยใช้ภาษา VHDL จะสะดวกกว่าการนำเอาอุปกรณ์มาเชื่อมต่อให้เป็นวงจรอย่างมาก เนื่องจากการใช้ภาษา VHDL นั้น การแก้ไขการทำงานของวงจรทำได้โดยการแก้ไขโค้ดของวงจร และไม่ต้องมีการโยงสายเชื่อมต่อระหว่างอุปกรณ์ต่างๆ ให้วุ่นวาย การตรวจสอบแก้ไขข้อผิดพลาดจึงทำได้ง่าย ดังนั้นการใช้ภาษา VHDL จึงมีความเหมาะสมเป็นอย่างมากที่จะนำมาใช้ในการสร้างวงจรกรองสัญญาณเชิงเลขในอันดับที่สูงๆ ต่อไป แต่เนื่องจากข้อจำกัดทางความจุของบอร์ด FPGA ที่ใช้ในโครงงานนี้จึงไม่สามารถสร้างวงจรกรองเชิงเลขอันดับที่สูงไปกว่าอันดับที่ 16ได้ ซึ่งเป็นขีดจำกัดในการทำโครงงานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงไปได้ ก็จากความร่วมมือ ความช่วยเหลือ ให้คำแนะนำและคำปรึกษาจาก รศ.ดร.กอบชัย เดชหาญ อ.เชื้อ นกอยู่และคุณศววัตน์ ชิวปรีชา ทางคณะผู้จัดทำจึงขอขอบคุณไว้ ณ ที่นี้ด้วย

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

-พรชัย ภววงษ์ศักดิ์ , การประมวลผลสัญญาณดิจิตอลเบื้องต้น. กรุงเทพมหานคร : มหาวิทยาลัยมหานคร , 2543.

-อรรถสิทธิ์ หล่าสกุล , Digital Signal Processing (DSP) Fundamentals. กรุงเทพมหานคร : สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2543

-วัลลภ สุระกำพลธร , การประมวลผลสัญญาณเชิงเลข. กรุงเทพมหานคร : สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2533

-John Wiley & Sons, Inc , Digital Filter Design. U.S.A. : Texas Instruments Incorporated ,1987

Andres antoniou , Digital filters analysis and design. Newyork : McGraw-Hill ,1987

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้