สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การออกแบบ Matched Filter โดยใช้อุปกรณ์ FPGA

Design Matched Filter By Using FPGA Device

โดย

นายจตุพล กันทอง

นางสาวพัลลภา เลิศรุ่งชัยสกุล

นายวิชัย ขาวสวัสดิ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่.....
เลขทะเบียน 54930
วัน,เดือน,ปี 1 ... 2548

b.....
i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบ Matched Filter โดยใช้อุปกรณ์ FPGA

Design Matched Filter By Using FPGA Device

โดย

นายจตุพล กันทอง 44015002

นางสาวพัลลภา เลิศรุ่งชัยสกุล 44015019

นายวิชัย ขาวสวัสดิ์ 44015025

อาจารย์ที่ปรึกษา

อาจารย์อัครพล ตรีรัตน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2546

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **การออกแบบ Matched Filter โดยใช้อุปกรณ์ FPGA**

Design Matched Filter By Using FPGA Device

ผู้จัดทำ

1. นายจตุพล กันทอง 44015002
2. นางสาวพัลลภา เลิศรุ่งชัยสกุล 44015019
3. นายวิชัย ขาวสวัสดิ์ 44015025

..... อาจารย์ที่ปรึกษา

(อาจารย์ อัครพล ตรีรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2546

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **การออกแบบ Matched Filter โดยใช้อุปกรณ์ FPGA**

Design Matched Filter By Using FPGA Device

ผู้จัดทำ

1. นายจตุพล กันทอง 44015002
2. นางสาวพัลลภา เลิศรุ่งชัยสกุล 44015019
3. นายวิชัย ขาวสวัสดิ์ 44015025

..... อาจารย์ที่ปรึกษา

(อาจารย์ อัครพล ตรีรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2546

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **การออกแบบ Matched Filter โดยใช้อุปกรณ์ FPGA**

Design Matched Filter By Using FPGA Device

ผู้จัดทำ

1. นายจตุพล กันทอง 44015002

2. นางสาวพัลลภา เลิศรุ่งชัยสกุล 44015019

3. นายวิชัย ขาวสวัสดิ์ 44015025

..... อาจารย์ที่ปรึกษา

(อาจารย์ อัครพล ตรีรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2546

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **การออกแบบ Matched Filter โดยใช้อุปกรณ์ FPGA**

Design Matched Filter By Using FPGA Device

ผู้จัดทำ

1. นายจตุพล กันทอง 44015002

2. นางสาวพัลลภา เลิศรุ่งชัยสกุล 44015019

3. นายวิชัย ขาวสวัสดิ์ 44015025

.....อาจารย์ที่ปรึกษา

(อาจารย์ อัครพล ตรีรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบ Matched Filter โดยใช้อุปกรณ์ FPGA

Design Matched Filter By Using FPGA Device

โดย นายจตุพล กันทอง 44015002

นางสาวพัลลภา เลิศรุ่งชัยสกุล 44015019

นายวิชัย ขาวสวัสดิ์ 44015025

อาจารย์ที่ปรึกษา อาจารย์อัครพล ตรีรัตน์

บทคัดย่อ

โครงงานนี้เป็นการศึกษาหลักการทำงานของแมทช์ฟิลเตอร์ (Matched Filter) ในระบบ DS-CDMA Detector โดยทำการรับสัญญาณชนิดรูทเรสต์โคไซน์(Root-Raised Cosine) เพื่อแก้ปัญหาการสอดแทรกระหว่างสัญลักษณ์ (ISI:Intersymbol Interference) และอาศัยหลักการเลขคณิตกระจาย (Distributed Arithmetic:DA) ในการสร้าง Matched Filter ในโครงงานนี้ใช้การออกแบบและพัฒนาโดยภาษา VHDL เพื่อสังเคราะห์วงจรลงบนอุปกรณ์ FPGA

Abstract

This project is to studying of Matched Filter design which is used in DS-CDMA Detector. The Matched Filter receives root-raise-cosine signal for avoiding ISI. In this project, we use DA-based structure and use VHDL to design and implement via FPGA environment. Finally, the performance analysis is performed and investigated for a designed Matched Filter.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	
1.1 ความเป็นมาของหัวข้อปริญญานิพนธ์	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	2
1.3 ขอบเขตของปริญญานิพนธ์	2
1.4 เนื้อหาของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการ	
2.1 การใช้ช่องสัญญาณร่วมแบบแบ่งรหัสชนิดไดเรกต์ซีเควนซ์	3
2.2 ภาคส่งของไดเรกต์ซีเควนซ์ซีดีเอ็มเอ	5
2.2.1 ส่วนมอดูเลเตอร์(Modulator)	5
2.2.2 ส่วนวอลช์สเปรด	5
2.3 ภาครับของระบบไดเรกต์ซีเควนซ์ซีดีเอ็มเอ	10
2.4 แมทช์ฟิลเตอร์(Matched Filter)	10
2.4.1 รูปพัลส์ในทางอุดมคติของแมทช์ฟิลเตอร์	19
2.4.2 รูปพัลส์ในทางปฏิบัติของแมทช์ฟิลเตอร์	19
2.5 การสอดแทรกระหว่างสัญลักษณ์(Intersymbol Interference)	19
2.6 พัลส์เรสต์โค ไซน์(Raise Cosine Pulse)	23
2.7 พัลส์รูทเรสต์โค ไซน์(Root raise cosine pulse)	27
2.8 ทฤษฎีการชักค่าตัวอย่าง(Sampling Theorem)	29
2.9 ทฤษฎีการตัดสินใจ(Decision Theorem)	33
2.9.1 กฎของเบย์(Bayes' Rule)	33
2.10 ไดอะแกรมรูปตา(Eye Diagram)	41
2.11 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย	44
2.11.1 ระบบตัวเลข	44
2.11.2 ทฤษฎีเลขคณิตกระจาย	47
บทที่ 3 การคำนวณและการสร้าง	53
3.1 การออกแบบวงจรกรองสัญญาณดิจิตอลแบบ FIR	53
3.1.1โครงสร้างสำหรับวงจรกรองสัญญาณดิจิตอลแบบ FIR	55
3.1.2 ข้อกำหนดของตัวกรอง	56
3.2 การหาค่าของตารางเปิดดู	57
3.3 การออกแบบอุปกรณ์	60
3.3.1 อุปกรณ์ควบคุม (Control)	60
3.3.2 อุปกรณ์ PIPO	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 อุปกรณ์ SHIFT 60
3.3.4 อุปกรณ์ ROM 61
3.3.5 อุปกรณ์ ADD 61
3.3.6 อุปกรณ์หารความถี่ 62
3.4 ภาษาวีเอชดีแอล 62
3.4.1 การออกแบบระบบดิจิตอล 62
3.4.2 ประวัติความเป็นมาของภาษาวีเอชดีแอล 64
3.4.3 องค์ประกอบพื้นฐานต่างๆ ของภาษาวีเอชดีแอล 64
3.4.4 การบรรยายเชิงพฤติกรรม 69
3.4.5 การโปรเซส 69
3.4.6 การกำหนดตัวดำเนินการภายในโปรเซส 70
3.4.7 การกำหนดการกระทำภายในโปรเซส 70
3.4.8 การกระตุ้นและยับยั้งการกระทำของโปรเซส 70
3.4.9 การออกแบบจากบนลงล่าง (Top-Down Design) 70
3.5 เอฟพีจีเอ 72
3.5.1 การออกแบบวงจรเชิงเลขด้วย ชิพอุปกรณ์เอฟพีจีเอ 73
3.5.2 สถาปัตยกรรมภายในของชิพอุปกรณ์เอฟพีจีเอ 74
3.5.2.1 แอลอี (LE : Logic Element) 74
3.5.2.2 แอลเอบี (LAB: Logic Array Block) 76
3.5.2.3 อีเอบี (EAB: Embedded Array Block) 77
3.5.2.4 ไอโออี (IOE: Input Output Element) 78
3.6 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว 79
3.7 ขั้นตอนในการออกแบบและทดสอบการทำงานของวงจรภายใน 80
บทที่ 4 การทดลองและผลการทดลอง 84
4.1 ขั้นตอนในการซิมมูเลตด้วยโปรแกรมแมทแลบ 84
4.1.1 ป้อนสัญญาณ Root-raised cosine ให้กับตัวกรอง 84
4.1.2 ป้อนสัญญาณ Root-raised cosine ที่มีสัญญาณรบกวน แบบ AWGN ให้กับตัวกรอง 91
4.1.3 ป้อนสัญญาณรูปคลื่นไซน์ให้กับตัวกรอง 104
4.1.4 ป้อนสัญญาณรูปคลื่นสี่เหลี่ยมให้กับตัวกรอง 104
4.2 ขั้นตอนในการซิมมูเลตด้วยโปรแกรมวีเอชดีแอล 105
4.2.1 อุปกรณ์ DIVIDER 105
4.2.2 อุปกรณ์ CONTROL 105
4.2.3 อุปกรณ์ PIPO 106

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 อุปกรณ์ SHIFT 106
4.2.5 อุปกรณ์ ROM 107
4.2.6 อุปกรณ์ ADD 107
4.2.7 ทำการรวมอุปกรณ์ต่างๆ 108
4.3 ขั้นตอนในการทดลองจริง 110
4.3.1 ป้อนสัญญาณ Root-raised cosine ให้กับตัวกรอง 110
4.3.2 ป้อนสัญญาณรูปคลื่นไซน์ให้กับตัวกรอง 115
4.3.3 ป้อนสัญญาณรูปคลื่นสี่เหลี่ยมให้กับตัวกรอง 116
บทที่ 5 บทวิจารณ์และบทสรุป 117
ภาคผนวก
กิตติกรรมประกาศ
เอกสารอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 แสดงบล็อกไดอะแกรมภาคส่งของ DS-CDMA 5

รูปที่ 2.2 โครงสร้างของวงจร PN Generator 8

รูปที่ 2.3 บล็อกไดอะแกรมภาครับของระบบไดเรกต์ซีเควนซ์ซีดีเอ็มเอ 10

รูปที่ 2.4 แผนภาพแสดงความสัมพันธ์ของสัญญาณที่ภาคส่งและภาครับ เมื่อส่งผ่านช่องสัญญาณ AWGN 10

รูปที่ 2.5 แผนภาพแสดงโครงสร้างของวงจรภาครับสัญญาณ 11

รูปที่ 2.6 ขั้นตอน Demodulation/Detector ของสัญญาณดิจิตอล 12

รูปที่ 2.7 (ก) characteristic ของแมทช์ฟิลเตอร์ 14

(ข) เอาท์พุทของแมทช์ฟิลเตอร์ 14

รูปที่ 2.8 บล็อกไดอะแกรมของระบบ baseband 15

รูปที่ 2.9 เครื่องรับแบบ correlation 17

รูปที่ 2.10 (ก) รูปคลื่นของสัญญาณ square wave 18

(ข) impulse response ของแมทช์ฟิลเตอร์ 18

(ค) output ของแมทช์ฟิลเตอร์ที่มีอินพุทเป็น square wave 18

รูปที่ 2.11 ภาพอธิบายประกอบเรื่องการส่งสัญญาณผ่านช่องสัญญาณที่มีแบนด์วิดท์จำกัด 20

รูปที่ 2.12 การส่งสัญญาณพัลส์ด้วยอัตราไนควิสต์ 21

รูปที่ 2.13 ผลตอบสนองอิมพัลส์ของรูปคลื่นสัญญาณที่เป็นแบบเรสต์โคไซน์ 24

รูปที่ 2.14 บล็อกไดอะแกรมของวงจร pulse generation 25

รูปที่ 2.15 ผลตอบสนองทางความถี่ของเรสต์โคไซน์ 27

รูปที่ 2.16 ผลตอบสนองอิมพัลส์ของรูปคลื่นสัญญาณที่เป็นแบบรูทเรสต์โคไซน์ 28

รูปที่ 2.17 ผลตอบสนองทางความถี่ของรูทเรสต์โคไซน์ 29

รูปที่ 2.18 สเปกตรัมของสัญญาณที่ถูกชักค่าตัวอย่าง 30

รูปที่ 2.19 สเปกตรัมของสัญญาณที่เกี่ยวเนื่องกับรูปที่ 2.18 เมื่อใช้ช่วงเวลาในการสุ่มค่าสัญญาณต่างๆ 32

รูปที่ 2.20 แผนภาพซิกแนลสเปซสำหรับการตัดสินใจบิทของระบบ BPSK 37

รูปที่ 2.21 ความน่าจะเป็นแบบมีเงื่อนไขของการส่งสัญญาณดิจิตอล 0 และ 1 40

รูปที่ 2.22 ความสัมพันธ์ระหว่างความน่าจะเป็นของความผิดพลาด P_e กับขนาดของ $\frac{E_b}{N_o}$ สำหรับระบบBPSK 41

รูปที่ 2.23 ภาพ eye diagram 42

รูปที่ 2.24 ไดอะแกรมรูปตาที่ได้จากไนควิสต์ฟิลเตอร์ 42

รูปที่ 2.25 ไดอะแกรมรูปตาที่ได้จากเรสต์โคไซน์ฟิลเตอร์ 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.26 ไดอะแกรมรูปตาที่ได้จากรูทเรสต์โคไซน์ฟิลเตอร์ทางด้านส่ง 43
รูปที่ 2.27 ไดอะแกรมรูปตาที่ได้จากรูทเรสต์โคไซน์ฟิลเตอร์ทางด้านรับ 43
รูปที่ 2.28 การจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิทจำนวนเต็มและบิทเศษส่วน 44
รูปที่ 2.29 การจัดรูปแบบจำนวนโดยตรงที่มีแต่บิทเศษส่วน 45
รูปที่ 2.30 การจัดรูปแบบจำนวนอิงดรรชนี 47
รูปที่ 2.31 การคูณแบบเลขส่วนเติมเต็มสองโดยใช้เลขคณิตกระจาย 50
รูปที่ 3.1 ผลตอบสนองแอมปลิจูดของวงจรกรองสัญญาณความถี่ต่ำผ่านในอุดมคติ 54
รูปที่ 3.2 การจัดโครงสร้างของวงจรกรองสัญญาณดิจิตอลแบบ FIR 55
รูปที่ 3.3 ตัวกรองแบบ FIR ที่ถูกลดรูปลงตามคุณสมบัติของตัวกรอง Root-Raised Cosine 56
รูปที่ 3.4 สัมประสิทธิ์ของตัวกรอง Root-Raised cosine แบบ FIR อันดับที่ 31
ที่มีค่าโรลออฟแฟคเตอร์เท่ากับ 0.3 56
รูปที่ 3.5 ขนาดของตัวกรอง Root-Raised cosine แบบ FIR อันดับที่ 31
ที่มีค่าโรลออฟแฟคเตอร์เท่ากับ 0.3 มีความถี่คัทออฟที่ 30kH$_z$ 57
รูปที่ 3.6 อุปกรณ์ควบคุม 60
รูปที่ 3.7 อุปกรณ์ PIPO 60
รูปที่ 3.8 อุปกรณ์ SHIFT 61
รูปที่ 3.9 อุปกรณ์ ROM 61
รูปที่ 3.10 อุปกรณ์ ADD 61
รูปที่ 3.11 อุปกรณ์หารความถี่ 62
รูปที่ 3.12 ขั้นตอนการออกแบบระบบดิจิตอล 63
รูปที่ 3.13 การออกแบบระบบเส้นทางข้อมูล 63
รูปที่ 3.14 โครงสร้างโดยทั่วไปของหน่วยการออกแบบเอนทิตี้ 65
รูปที่ 3.15 โครงสร้างโดยทั่วไปของหน่วยการออกแบบสถาปัตยกรรม 65
รูปที่ 3.16 โครงสร้างโดยทั่วไปของส่วนการประกาศแพ็กเก็ต 67
รูปที่ 3.17 โครงสร้างของบอดี้แพ็กเก็ต 67
รูปที่ 3.18 โครงสร้างโดยทั่วไปของหน่วยการออกแบบโครงแบบ 68
รูปที่ 3.19 ตัวดำเนินการในวีเอชดีแอล 68
รูปที่ 3.20 รูปแบบของการบรรยายแบบโปรเซส 70
รูปที่ 3.21 ขั้นตอนการออกแบบจากบนลงล่าง 71
รูปที่ 3.22 ผังแสดงการแบ่งกลุ่มของวงจรเอซิก 72
รูปที่ 3.23 ลักษณะของตัว FPGA และการนำไปใช้งาน 73
รูปที่ 3.24 โครงสร้างของชิพอุปกรณ์เอฟพีจีเอ ตระกูล FLEX 10K 74
รูปที่ 3.25 โครงสร้างภายในของแอลอี 75
รูปที่ 3.26 โครงสร้างของ Package body 75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.27 โครงข่ายของการเชื่อมต่อ 76
รูปที่ 3.28 โครงสร้างภายในของแอลเอบี 77
รูปที่ 3.29 โครงสร้างภายในอีเอบี 79
รูปที่ 3.30 โครงสร้างภายในของไอโออี 79
รูปที่ 3.31 การโปรแกรมลงในชิพอุปกรณ์เอฟพีจีเอ 80
รูปที่ 3.32 โครงสร้างของวงจรกรองสัญญาณเชิงเลขอันดับที่ 2 จากการแทนด้วยปริภูมิสเตท 80
รูปที่ 3.33 ไทม์มิ่งไดอะแกรมของสัญญาณควบคุม 81
รูปที่ 3.34 สเตทไดอะแกรมของ Control Unit 83
รูปที่ 4.1 สัญญาณอินพุทและเอาท์พุทของตัวกรองที่มีค่าโรลออฟ = 0.1 84
รูปที่ 4.2 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีค่าโรลออฟ = 0.1 84
รูปที่ 4.3 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทที่รับสัญญาณอินพุทมีค่าโรลออฟ = 0.1 85
รูปที่ 4.4 สัญญาณอินพุทและเอาท์พุทของตัวกรองที่มีค่าโรลออฟ = 0.2 85
รูปที่ 4.5 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีค่าโรลออฟ = 0.2 86
รูปที่ 4.6 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทโรลออฟ = 0.2 86
รูปที่ 4.7 สัญญาณอินพุทและเอาท์พุทของตัวกรองที่มีค่าโรลออฟ = 0.3 87
รูปที่ 4.8 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.3 87
รูปที่ 4.9 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทโรลออฟ = 0.3 88
รูปที่ 4.10 สัญญาณอินพุทและเอาท์พุทของตัวกรองที่มีค่าโรลออฟ = 0.4 88
รูปที่ 4.11 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีค่าโรลออฟ = 0.4 89
รูปที่ 4.12 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.4 89
รูปที่ 4.13 สัญญาณอินพุทและเอาท์พุทของตัวกรองที่มีค่าโรลออฟ = 0.5 90
รูปที่ 4.14 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีค่าโรลออฟ = 0.5 90
รูปที่ 4.15 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.5 91
รูปที่ 4.16 สัญญาณอินพุทและเอาท์พุทของตัวกรองที่มีค่าโรลออฟ = 0.7 91
รูปที่ 4.17 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.7 92
รูปที่ 4.18 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทที่รับสัญญาณอินพุทมีค่าโรลออฟ = 0.7 92
รูปที่ 4.19 สัญญาณอินพุทและเอาท์พุทของตัวกรองที่มีค่าโรลออฟ = 0.9 93
รูปที่ 4.20 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.9 93
รูปที่ 4.21 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทที่รับสัญญาณอินพุทมีค่าโรลออฟ = 0.9 94
รูปที่ 4.22 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN
และเอาท์พุทของตัวกรองมี SNR = 0 95
รูปที่ 4.23 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 0 95
รูปที่ 4.24 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทมี SNR = 0 96
รูปที่ 4.25 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรองมี SNR = 5 96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.26 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGNมี SNR = 5 97
รูปที่ 4.27 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทมี SNR = 5 97
รูปที่ 4.28 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรองมี SNR = 10 98
รูปที่ 4.29 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGNมี SNR = 10 98
รูปที่ 4.30 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทมี SNR = 10 99
รูปที่ 4.31 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรองมี SNR = 15 99
รูปที่ 4.32 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGNมี SNR = 15 100
รูปที่ 4.33 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทมี SNR = 15 100
รูปที่ 4.34 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรองมี SNR = 20 101
รูปที่ 4.35 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 20 101
รูปที่ 4.36 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทมี SNR = 20 102
รูปที่ 4.37 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรองมี SNR = 25 102
รูปที่ 4.38 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGNมี SNR = 25 103
รูปที่ 4.39 ไดอะแกรมรูปตาของสัญญาณเอาท์พุทมี SNR = 25 103
รูปที่ 4.40 สัญญาณอินพุท และเอาท์พุทที่เป็นสัญญาณรูปไซน์ของตัวกรอง 104
รูปที่ 4.41 สัญญาณอินพุท และเอาท์พุทที่เป็นสัญญาณรูปสี่เหลี่ยมของตัวกรอง 104
รูปที่ 4.42 การซิมมูเลตอุปกรณ์ DIV 105
รูปที่ 4.43 การซิมมูเลตอุปกรณ์ CONTROL 105
รูปที่ 4.44 การซิมมูเลตอุปกรณ์ PIPO 106
รูปที่ 4.45 การซิมมูเลตอุปกรณ์ SHIFT 106
รูปที่ 4.46 การซิมมูเลตอุปกรณ์ ROM 107
รูปที่ 4.47 การซิมมูเลตอุปกรณ์ ADD 107
รูปที่ 4.48 วงจรรวมของตัวกรองเอฟไออาร์ 108
รูปที่ 4.49 วงจรรวมของตัวกรองเอฟไออาร์ 108
รูปที่ 4.50 ความเร็วสูงสุดในการทำงานของวงจร 109
รูปที่ 4.51 การซิมมูเลตวงจรรวมของตัวกรองเอฟไออาร์ 109
รูปที่ 4.52 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 110
รูปที่ 4.53 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 111
รูปที่ 4.54 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 111
รูปที่ 4.55 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 112
รูปที่ 4.56 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 112

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.57 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 113
รูปที่ 4.58 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 113
รูปที่ 4.59 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 114
รูปที่ 4.60 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch2) ของตัวกรอง 114
รูปที่ 4.61 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch1) ที่เป็นสัญญาณ 115
รูปไซน์ของตัวกรอง(ความถี่อินพุท 8.5 kHz)
รูปที่ 4.62 สัญญาณอินพุท(Ch 1) และเอาท์พุท(Ch 2) ที่เป็นสัญญาณรูปไซน์ 115
ของตัวกรอง (ความถี่อินพุท 25 kHz)
รูปที่ 4.63 สัญญาณอินพุท(Ch 1) และเอาท์พุท(Ch 2) ที่เป็นสัญญาณรูปไซน์ 116
ของตัวกรอง (ความถี่อินพุท 35 kHz)
รูปที่ 4.64 สัญญาณอินพุท(Ch 1) และเอาท์พุท(Ch 2) ที่เป็นสัญญาณรูปสี่เหลี่ยมของตัวกรอง 116

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงค่าสัมประสิทธิ์ตามขนาดของชิพตรีจิสเตอร์	9
ตารางที่ 2.2 แสดงคุณสมบัติที่สำคัญของรูปแบบจำนวนโดยตรง	45
ตารางที่ 2.3 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง	49
ตารางที่ 2.4 แสดงค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุท	52
ตารางที่ 3.1 ตารางเปิดดูที่มีขนาด 4 บิท ตัวที่ 1	58
ตารางที่ 3.2 ตารางเปิดดูที่มีแอดเดรสขนาด 4 บิท ตัวที่ 2	58
ตารางที่ 3.3 ตารางเปิดดูที่มีแอดเดรสขนาด 4 บิท ตัวที่ 3	59
ตารางที่ 3.4 ตารางเปิดดูที่มีแอดเดรสขนาด 4 บิท ตัวที่ 4	59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของหัวข้อปริญญานิพนธ์

ปัจจุบันความต้องการในการสื่อสารข้อมูลเพิ่มขึ้นมากจากในอดีต โดยเพิ่มทั้งในแง่ปริมาณข้อมูลในการรับ – ส่งที่มากขึ้น และความหลากหลายของรูปแบบข้อมูลที่รับ – ส่ง อีกทั้งได้มีการขยายตัวเพื่อรองรับการใช้งานของสาธารณะชนมากขึ้น จากเดิมที่การสื่อสารข้อมูลจำกัดวงอยู่เฉพาะตามองค์กรใหญ่หรือสถาบันการศึกษาเนื่องจากมีค่าใช้จ่ายที่ค่อนข้างสูงมากก็ขยายตัวออกไปตามบ้านเรือนประชาชน คนทั่วไปสามารถเชื่อมต่อกับโลกภายนอกได้โดยอาศัยคู่สายโทรศัพท์เพียงเส้นเดียวโดยมีค่าใช้จ่ายไม่สูงนัก นอกจากระบบดังกล่าวแล้ว ยังมีความต้องการรูปแบบของการบริการอีกหลากหลายทั้งในรูปแบบการใช้สายและแบบไร้สายผ่านทางโทรศัพท์เคลื่อนที่มีการบริการในรูปแบบของพูดคุยสนทนา การส่งข้อความสั้น (Short Massage) เข้าเครือข่ายอินเตอร์เน็ตโดยใช้เทคโนโลยี WAP (Wire Less Application Protocal) หรือการส่งข้อมูลระหว่างเครื่องใช้ไฟฟ้าด้วยกันโดยใช้เทคโนโลยีบรูทูธ (Bluetooth) ความสามารถในการรองรับอัตราการส่งข้อมูลและจำนวนผู้ใช้ที่เพิ่มขึ้น ดังกล่าวเกิดขึ้นได้จากการใช้ช่องสัญญาณความถี่ที่มีอยู่อย่างจำกัดอย่างเต็มประสิทธิภาพ ทั้งเทคโนโลยีการบีบอัดข้อมูลการส่งและเทคโนโลยีเข้าถึงแบบหลายทางทำให้สามารถส่งข้อมูลผ่านช่องสัญญาณเดิมเพิ่มขึ้นจากในอดีตเป็นอย่างมาก

แต่ความต้องการไม่ได้จำกัดอยู่เพียงเท่านี้ โดยเฉพาะอย่างยิ่งในระบบการสื่อสารไร้สายซึ่งจะเป็นที่แพร่หลายมากในระบบการสื่อสารยุคปัจจุบัน เนื่องจากสามารถอำนวยความสะดวกให้กับผู้ใช้เป็นอย่างมากเพราะสามารถให้บริการได้ทุกเวลาและทุกสถานที่ รูปแบบและประสิทธิภาพในการให้บริการจะเปลี่ยนแปลงไปในทิศทางที่ดีขึ้นจากเดิมที่โทรศัพท์เคลื่อนที่มีความสามารถเพียงรับส่งข้อมูลเสียงหรือข้อความสั้น ต่อมามีการพัฒนาระบบ WAP เพื่อให้สามารถเชื่อมต่อระบบอินเตอร์เน็ตโดยใช้ตัวโทรศัพท์เอง แต่ก็ยังมีข้อจำกัดในด้านการประมวลผลข้อมูลและแสดงผลของตัวโทรศัพท์ที่มีประสิทธิภาพไม่สูงนัก หรือถ้าจะอาศัยโทรศัพท์เป็นช่องทางเชื่อมต่อคอมพิวเตอร์เข้ากับระบบอินเตอร์เน็ต ก็จะมีอัตราการรับ – ส่งข้อมูลไม่สูงนักของจำกัดเหล่านี้กำลังถูกแก้ไขปรับปรุงเพื่อให้ระบบมีประสิทธิภาพสูงขึ้น

การเข้าใช้ช่องสัญญาณร่วมกันโดยการแบ่งรหัส Code Division Multiplex Access :(CDMA) ถูกเลือกใช้กับระบบสื่อสารข้อมูลในปัจจุบัน ซึ่งชนิดของข้อมูลจะมีความหลากหลายมากขึ้นไม่จำกัดเฉพาะข้อมูลเสียงเท่านั้น เนื่องจากคุณสมบัติที่ดีกว่าระบบการใช้ช่องสัญญาณร่วมแบบอื่น ๆ ในการรองรับอัตราการส่งที่สูงและผู้ใช้จำนวนมาก โดยให้ประสิทธิภาพและความถูกต้องสูงเหมาะสำหรับการสื่อสารในปัจจุบันและอนาคตซึ่งข้อมูลเป็นแบบดิจิตอลมาก และคาดว่าจะเป็นระบบที่จะใช้กันอย่างแพร่หลายในอนาคตไม่ได้จำกัดเฉพาะในระบบโทรศัพท์เคลื่อนที่เท่านั้น จึงเหมาะสมอย่างมากที่ได้ศึกษาและออกแบบสร้าง DS – CDMA โดยใช้หลักการของแมทช์ฟิลเตอร์ซึ่งเป็นส่วนหนึ่งของระบบ CDMA นี้เพื่อเป็นแนวทางในการพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1.2.1 เพื่อศึกษาทฤษฎีและหลักการทำงานของแมทช์ฟิลเตอร์

1.2.2 ทำการจำลองการทำงานของแมทช์ฟิลเตอร์โดยใช้โปรแกรมแมทแลบ (Matlab)

1.2.3 ทำการจำลองการทำงานของแมทช์ฟิลเตอร์ด้วยภาษา วีเอชดีแอล และเก็บผลด้วยโปรแกรม max+plus2

1.2.4 ทำการออกแบบสร้างเป็นฮาร์ดแวร์บนชิพของอุปกรณ์ เอฟพีจีเอ ด้วยภาษา วีเอชดีแอล

1.3 ขอบเขตของปริญญานิพนธ์

ในระบบสื่อสารข้อมูลที่สมบูรณ์จะประกอบด้วยทั้งภาคส่งและภาครับ ซึ่งในโครงการนี้จะเป็นการศึกษาภาครับในส่วน DS-CDMA โดยอาศัยหลักการของแมทช์ฟิลเตอร์เท่านั้น และทำการจำลองการทำงานของแมทช์ฟิลเตอร์โดยใช้โปรแกรมแมทแลบ และทดลองให้เห็นถึงการทำงานจริง ซึ่งกระบวนการทำงานทั้งหมดจะถูกบรรยายพฤติกรรมการทำงาน เพื่อสร้างเป็นฮาร์ดแวร์โดยใช้การเขียนด้วยภาษาวีเอชดีแอล ในการออกแบบวงจรทางลอจิกภายในอุปกรณ์ FPGA

1.4 เนื้อหาของปริญญานิพนธ์

ในบทที่ 2 จะกล่าวถึงทฤษฎีของแมทช์ฟิลเตอร์ โดยแสดงให้เห็นข้อดีต่าง ๆ ของแมทช์ฟิลเตอร์ รวมถึงหลักการของเลขคณิตกระจาย

ในบทที่ 3 จะกล่าวถึงการคำนวณและการออกแบบการสร้างแมทช์ฟิลเตอร์ พร้อมทั้งตัวอย่างการออกแบบในส่วนของการออกแบบโครงสร้างเอฟพีจีเอและทฤษฎีของภาษาวีเอชดีแอล รวมถึงหลักการออกแบบเพื่อโปรแกรมลงบน ชิพเอฟพีจีเอ และโครงสร้างในส่วนต่าง ๆ ของบอร์ดเอฟพีจีเอ

ในบทที่ 4 กล่าวถึงการทดลองและผลการทดลองที่ได้จากการคำนวณและการออกแบบการสร้างแมทช์ฟิลเตอร์

ในบทที่ 5 กล่าวถึงบทวิจารณ์และบทสรุป

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 การใช้ช่องสัญญาณร่วมแบบแบ่งรหัสชนิดไดเรกต์ซีเควนซ์

การใช้ช่องสัญญาณร่วมแบบแบ่งรหัส(Code Division Multiple Access หรือ CDMA) พัฒนามาจากการทำเปรดสเปกตรัม ซึ่งในยุคแรกๆของการทำเสปรดสเปกตรัมนั้นมีจุดมุ่งหมายเพื่อนำมาใช้งานในวงการทหารสำหรับป้องกันการรบกวนของข้าศึกต่อการรับส่งสัญญาณ(Enemy Jamming)และเป็นการใช้งานแบบผู้ใช้รายเดียว โดยในปัจจุบันได้นำวิธีการสเปดสเปกตรัมมาประยุกต์ในระบบสื่อสารข้อมูลแบบผู้ใช้หลายราย โดยอาศัยประโยชน์ที่ได้จากการเพิ่มอัตราการส่งข้อมูลให้สูงขึ้น ซึ่งในทางทฤษฎีแล้วมีความเป็นไปได้ที่สามารถเพิ่มอัตราการส่งข้อมูลให้สูงกว่าระบบที่มีการใช้งานอยู่อย่างแพร่หลายในขณะนี้ คือระบบการใช้ช่องสัญญาณร่วมแบบแบ่งความถี่(Frequency Division Multiple Access:FDMA) และระบบการใช้ช่องสัญญาณร่วมแบบแบ่งเวลา(Time Division Multiple Access:TDMA) นอกจากนั้นยังให้ความทนทานต่อการรบกวนระหว่างช่องสัญญาณ เป็นระบบที่มีประสิทธิภาพสูงสุดในสภาวะที่มีผู้ใช้หลายราย คุณสมบัติเช่นนี้ทำให้เทคนิคการใช้ช่องสัญญาณร่วมแบบแบ่งรหัสเป็นที่สนใจของผู้ออกแบบระบบการสื่อสารไร้สายเป็นอย่างยิ่ง

เทคนิค CDMA แบ่งได้เป็นสองชนิดใหญ่ๆ คือ CDMA แบบไดเรกต์ซีเควนซ์(Direct Sequence-CDMA หรือ DS-CDMA) และ CDMA แบบฟรีเควนซีฮอบ(Frequency Hopped CDMA หรือ FH-CDMA) โดยเทคนิค DS-CDMA เป็นการคูณรหัสคงที่ชุดหนึ่งที่มีอัตราข้อมูลสูงกว่าเข้ากับข้อมูลของแต่ละผู้ใช้ ข้อมูลที่ได้จากการคูณจะมีอัตราข้อมูลสูงกว่าข้อมูลเดิม ข้อมูลตรงนี้จากหลายผู้ใช้จะถูกรวมกันเป็นข้อมูลไม่ต่อเนื่องแบบหลายระดับ จากนั้นจึงส่งเข้าช่องสัญญาณ ในการแยกข้อมูลทางด้านฝั่งรับจะใช้รหัสคงที่ที่ใช้คูณเป็นตัวในด้านส่งของผู้ใช้แต่ละรายคูณเข้าไปกับข้อมูลที่ส่งมา ผลการคูณเมื่อนำมาเฉลี่ยให้เป็นอัตราข้อมูลของผู้ใช้จะได้เฉพาะข้อมูลของผู้ใช้รายนั้นๆออกมา

ส่วนเทคนิค FH-CDMA เป็นระบบการใช้ช่องสัญญาณร่วมที่ความถี่พาห์ของผู้ใช้แต่ละรายจะเปลี่ยนไปในรูปแบบของการสุ่มภายในความถี่ช่องสัญญาณ ข้อมูลดิจิตอลจะถูกกระจายไปด้วยรูปแบบที่แน่นอนและถูกส่งไปกับความถี่พาห์ที่ต่างกัน โดยที่ความถี่พาห์หนึ่งๆจะเล็กกว่าแถบความถี่ทั้งหมดที่กระจายแล้ว

ระบบไดเรกต์ซีเควนซ์ซีดีเอ็มเอ(Direct Sequence-Code Division Multiple Access : DS-CDMA)

การวิจัยและพัฒนาระบบโทรศัพท์เคลื่อนที่แบบ CDMA เริ่มขึ้นอย่างจริงจังเมื่อประมาณต้นปี ค.ศ. 1989 และไม่กี่ปีต่อมา คือ ในปลายปี ค.ศ. 1992 ผลงานวิจัยและพัฒนาเหล่านี้ก็ได้รับการเสนอและยอมรับได้เป็นมาตรฐานที่เรียกว่า IS-95 ในปัจจุบันโทรศัพท์เคลื่อนที่แบบ CDMA นี้ได้เริ่มมีการนำมาใช้งานจริงจังกันในหลายประเทศทั่วโลก

หลักการเบื้องต้นของระบบ CDMA ได้จัดแบ่งความถี่ที่มีอยู่ออกเป็นช่วง โดยแต่ละช่วงมีความกว้างเท่ากับ 1.25 MHz ด้วยขนาดแถบความถี่ที่กว้างขนาดนี้ CDMA อนุญาติให้ผู้ใช้โทรศัพท์จำนวนมากจำนวนหนึ่งส่งสัญญาณเสียงลงบนความถี่เดียวกันได้ ซึ่งหมายความว่าผู้ใช้ทุกๆคนที่ใช้คลื่นพาห์เดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะรบกวนกันเชิงความถี่ตลอดเวลาการใช้งาน หากแต่ว่าโทรศัพท์แต่ละเครื่องสามารถแยกแยะว่าสัญญาณส่วนใดที่เป็นของตนเองได้โดยอาศัยชุดรหัสที่แตกต่างกันในการรับและส่งข้อมูล จึงเป็นที่มาของชื่อระบบ CDMA ชุดรหัสที่ใช้ในการทำสเปรดสเปกตรัมเหล่านี้มีชื่อเรียกว่า PN(Pseudorandom Noise)ซึ่งรหัสแต่ละชุดที่ใช้ต้องมีคุณสมบัติที่ตั้งฉาก(Orthogonal)กับรหัสอื่นๆทั้งหมด

ขั้นตอนการสร้างสัญญาณสำหรับการส่งออกเริ่มด้วยการนำสัญญาณเสียงที่อยู่ในรูปของสัญญาณดิจิตอลซึ่งมีอัตราของข้อมูลค่อนข้างต่ำ(เช่น 9600 bps)มาคูณกับสัญญาณ PN ซึ่งมีอัตราข้อมูลที่สูงกว่ามาก(1.228 Mbps) ผลลัพธ์ที่ได้ก็คือ ข้อมูลที่มีอัตราการส่งสูงกว่าสัญญาณเสียงที่ต้องการส่งจะส่งมาก ด้วยเหตุนี้จึงเรียกว่ามีการสเปรดสเปกตรัม(Spead Spectrum)เกิดขึ้นและแถบความถี่ของช่องสัญญาณที่ใช้ในการส่งนี้ต้องมีขนาดใหญ่ขึ้นด้วย คือ 1.25 MHz ส่วนของภาครับสัญญาณที่ปลายทางก็จะทำการดีสเปรด (Despread) สัญญาณที่รับได้เพื่อดึงสัญญาณเสียงเดิมกลับคืนมาในการดีสเปรดนั้นจะอาศัยวงจรคอรีเลเตอร์(Correlator)ซึ่งทำหน้าที่คูณสัญญาณที่ได้รับกับรหัส PN ชุดเดิมซึ่งเหมือนกับการคูณสัญญาณที่ได้รับกับรหัส PN ชุดเดิมซึ่งเหมือนกับการคูณที่ภาคส่งและจากนั้นก็จะหาค่าเฉลี่ยของสัญญาณทุกๆหนึ่งคาบของอัตราบิทของสัญญาณเสียง จะได้สัญญาณเสียงของสำหรับผู้รับแต่ละคนออกมา(หมายเหตุ: สัญญาณข้อมูลที่ต้องการส่งออกไม่จำเป็นต้องเป็นสัญญาณเสียงเท่านั้น อาจเป็นสัญญาณข้อมูลแบบอื่นก็ได้)

การที่เครื่องรับโทรศัพท์เคลื่อนที่แต่ละเครื่องที่ส่งผ่านสัญญาณเสียง หรือ ข้อมูลในรูปของสัญญาณดิจิตอลโดยใช้รหัส PN ที่แตกต่างกันและมีคุณสมบัติที่ตั้งฉากซึ่งกันและกัน ทำให้การรับและส่งสัญญาณของโทรศัพท์เครื่องๆหนึ่ง จะไม่ไปดีสเปรดสัญญาณที่ถูกสเปรดด้วยรหัส PN ชุดอื่นๆออกมา ด้วยเหตุนี้เครื่องโทรศัพท์แต่ละเครื่องจึงสามารถแยกแยะสัญญาณที่เป็นของตนเองได้ ค่าพารามิเตอร์ค่าหนึ่งที่สำคัญ คือ อัตราการประมวลผล (Processing gain) ซึ่งได้กำหนดให้มีค่าเท่ากับอัตราส่วนระหว่างอัตราบิทข้อมูลของรหัส PN ต่ออัตราการส่งของข้อมูลจริง ในกรณีที่ส่งสัญญาณข้อมูลด้วยอัตรา 9600 bps จะได้ค่าอัตราการประมวลผลเท่ากับ 1.228 Mbps/9600 bps = 128 หรือเท่ากับ 21 dB

คุณลักษณะของระบบไดเรกต์ซีเควนซ์ซีดีเอ็มเอเป็นดังนี้

- โดยธรรมชาติแล้วในระหว่างการสนทนาของมนุษย์จะมีเพียงประมาณ 35% ของเวลาทั้งหมดเท่านั้นที่มีสัญญาณเสียงของผู้พูด ส่วนเวลาที่เหลือจะเป็นการฟัง ในระบบ CDMA นั้นเนื่องจากผู้ใช้บริการทุกคนจะถูกจัดให้ช่องสัญญาณช่องเดียวกันหมด ดังนั้นถ้ามีผู้ใช้บางส่วนที่ไม่มีการพูดก็หมายถึงว่าปริมาณสัญญาณรบกวนระหว่างกันของผู้ใช้ ณ เวลานั้นๆก็ลดลงด้วย ซึ่งโดยหลักการแล้วอาจจะลดได้ถึง 65 % นั่นหมายถึงเราสามารถเพิ่มขนาดความจุของระบบขึ้นได้ประมาณ 3 เท่า

- ในการส่งข้อมูลที่อัตราการส่งสูงๆเช่น มากกว่า 10 Kbps มากๆนั้น ในระบบ FDMA และระบบ TDMA จำเป็นต้องอาศัยวงจรอีควอไลเซอร์ในการลดทอนสัญญาณที่เรียกว่า การสอดแทรกทางสัญลักษณ์ซึ่งเกิดจากการแผ่ของสัญญาณเนื่องจากการสเปรดของการหน่วงเวลา ในขณะที่วงจรภาครับของระบบ CDMA ไม่มีความจำเป็นต้องใช้วงจรอีควอไลเซอร์สำหรับการปรับแต่งรูปสัญญาณ แต่ต้องอาศัยวงจรคอร์รีเลเตอร์ เพื่อใช้ในการดีสเปรดสัญญาณกลับคืนมา ซึ่งวงจรคอร์รีเลเตอร์นั้นมีความซับซ้อนน้อยกว่าวงจรอีควอไลเซอร์มาก

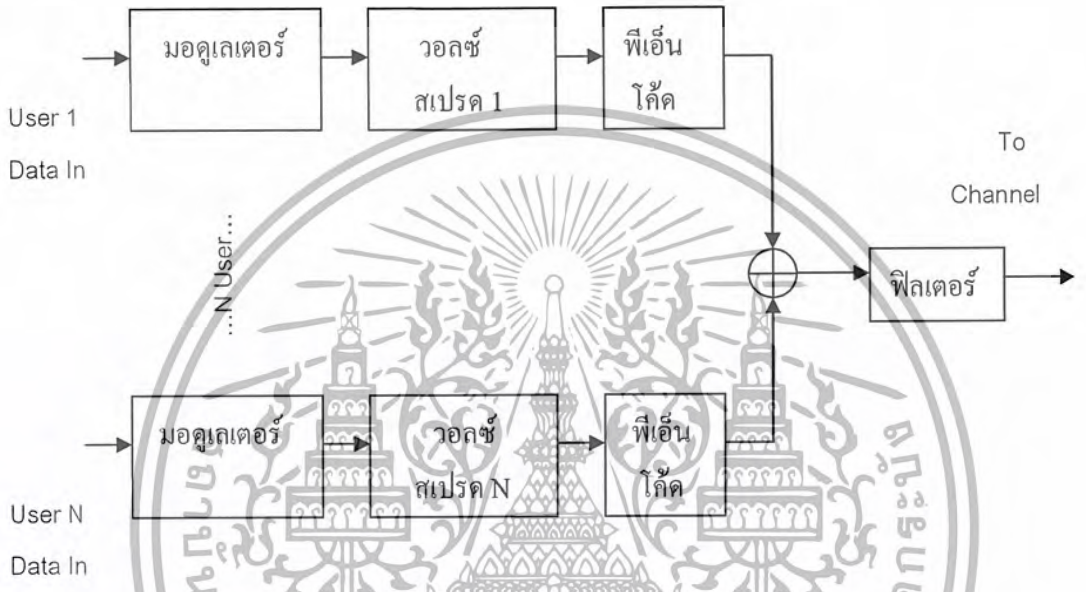
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบ CDMA ใช้วงจรรับส่งวิทยุเพียงชุดเดียวสำหรับครอบคลุมบริเวณหนึ่งๆ ดังนั้นจึงง่ายและสะดวกในการติดตั้งอีกทั้งประหยัดเนื้อที่ในการจัดเก็บอุปกรณ์ด้วย

- เนื่องจากในแต่ละเซลล์ใช้คลื่นวิทยุความถี่เดียวกันจึงไม่มีความจำเป็นต้องมีการแฮนด์ออฟจากความถี่หนึ่งไปยังอีกค่าความถี่หนึ่งในกรณีที่ตัวเครื่องโทรศัพท์เคลื่อนที่จากเซลล์หนึ่ง

2.2 ภาคส่งของไดเรกต์ซีเควนซ์ซีดีเอ็มเอ

บล็อกไดอะแกรมภาคส่งของไดเรกต์ซีเควนซ์ซีดีเอ็มเอ



รูปที่ 2.1 บล็อกไดอะแกรมภาคส่งของ DS-CDMA

2.2.1 ส่วนมอดูเลเตอร์(Modulator)

เป็นการจัดระดับสัญญาณจากผู้ใช้ให้เหมาะสมกับการทำงาน ซึ่งในขั้นตอนถัดไปเป็นการประมวลผลด้านสัญญาณโดยการคูณและบวกกันของสัญญาณ การที่ข้อมูลจากผู้ใช้มีระดับสัญญาณเป็น '0' และ '1' ไม่เหมาะสมที่จะใช้ในการประมวลผลแบบดังกล่าว ต้องมีการแปลงระดับสัญญาณให้เหมาะสมก่อน โดยการมอดูเลตแบบ Binary Phase Shift Keying(BPSK) โดยขั้นตอนนี้เหมือนการเทียบระดับสัญญาณใหม่เพื่อให้ระดับสัญญาณเปลี่ยนเป็น '-1' และ '1'

2.2.2 ส่วนวอลช์สเปรด

เป็นการขั้นตอนการเข้ารหัสของระบบ โดยการคูณข้อมูลผู้ใช้ที่ผ่านการมอดูเลตแบบ BPSK แล้วด้วยรหัสชุดหนึ่งที่มีอัตราส่งสูงกว่าอัตราข้อมูล ในที่นี้เลือกใช้รหัสวอลช์

รหัสวอลช์ที่คูณเข้ากับข้อมูลของผู้ใช้แต่ละรายจะแตกต่างกันตรงนี้มีความสำคัญและเป็นหัวใจของระบบ เพราะรหัสวอลช์จะเป็นตัวแยกข้อมูลของผู้ใช้แต่ละรายจากข้อมูลที่เข้ารหัสเสร็จ ซึ่งจะใช้ในขั้นตอนถอดรหัสในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัสวอลช์ที่ใช้จะมีขนาดกว้างเท่ากับจำนวนผู้ใช้ มีความยาวเป็นค่าๆหนึ่ง และมีอัตราข้อมูลมากกว่าเป็นจำนวนเท่าที่เท่ากับความยาวของรหัส เพื่อให้รหัสวอลช์ทั้งชุดได้คูณกับข้อมูลในทุกลำดับ กล่าวคือในระบบที่มีจำนวนผู้ใช้ m ราย และอัตราการส่งข้อมูลเป็น Fs จะต้องใช้วอลช์ที่มีขนาด $n \times m$ และความถี่ $Fs \times m$

เราขอยกตัวอย่างประเภทของรหัส PN ที่ใช้ในระบบโทรศัพท์เคลื่อนที่ระบบ CDMA ตามมาตรฐาน IS-95 ซึ่งมีรหัส PN อยู่ด้วยกัน 3 ประเภท คือ รหัสวอลช์,รหัส PN ยาว และรหัส PN สั้นโดยรายละเอียดของรหัสแต่ละประเภทมีดังนี้

- รหัสวอลช์ (Walsh Code) มีชื่อเรียกอีกอย่างหนึ่งว่ารหัสฮาดามาร์ด (Hadamard Code) ประกอบด้วยรหัสที่มีคุณสมบัติที่ตั้งฉากระหว่างกันทั้งหมด 64 ชุด โดยที่รหัสแต่ละชุดมีความยาวเท่ากับ 64 บิต รหัสประเภทนี้ออกแบบมาเพื่อใช้สำหรับการสเปรดช่องสัญญาณที่มีอัตราการส่ง 1.228 Mbps

หมายเหตุ : รหัส 2 ชุด จะมีคุณสมบัติที่ตั้งฉากระหว่างกันก็ต่อเมื่อค่าสหสัมพันธ์ไขว้ (Cross Correlation) ระหว่างรหัสคู่นี้มีค่าเป็นศูนย์ การคำนวณหาค่าสหสัมพันธ์ไขว้สามารถหาได้จาก

โดยที่ X_i และ Y_i คือค่าคือค่าของบิทที่ i ของรหัสชุดแรกและชุดที่สองในรูปของไบโพลาร์ นั่นคือว่า $X_i, Yi = -1$ ถ้าบิทที่ i นั้นมีค่าเป็นศูนย์ และ $X_i, Yi = 1$ ถ้าบิตที่ i นั้นมีค่าเป็นหนึ่ง

รหัสวอลช์สามารถสร้างได้จากเมตริกซ์ฮาดามาร์ด (Hadamard Matrices) โดยที่ m คือค่า 2 ยกกำลังเลขจำนวนเต็ม (m = 1,2,3 ....) ค่าเริ่มต้นของเมตริกซ์ฮาดามาร์ดที่ m = 1 กำหนดให้เป็น

$$H_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{2.1}$$

และค่า H_{2m} อื่นๆสร้างจากความสัมพันธ์ต่อไปนี้

$$H_{2m} = \begin{bmatrix} H_m & H_m \\ H_m & \overline{H_m} \end{bmatrix} \tag{2.2}$$

ตัวอย่างกรณี m = 2 ได้

$$H_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \tag{2.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และกรณี m = 4 ได้

$$H_8 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

2.2.3 รหัส PN

- รหัส PN ยาว รหัสประเภทนี้สร้างจากชิฟต์รีจิสเตอร์ขนาด 42 บิท โดยการต่อเชื่อมกันให้มีโครงสร้างเป็น Maximum Length LFSRs และใช้โพลิโนเมียลต่อไปนี้

$$p(x) = x^{42} + x^{35} + x^{33} + x^{31} + x^{27} + x^{26} + x^{25} + x^{22} + x^{21} + x^{19} + x^{18} + x^{17} + x^{16} + x^{10} + x^{7} + x^{6} + x^{5} + x^{3} + x^{2} + x + 1 \tag{2.5}$$

ผลที่ได้คือ รหัสที่มีความยาวถึง $2^{42} - 1 = 4.398 \times 10^{12}$ เนื่องจากอัตราการสร้างรหัสนี้มีค่าเท่ากับ 1.228 Mbps ดังนั้นรหัสนี้จะมีการซ้ำทุกๆ 41.425 วัน

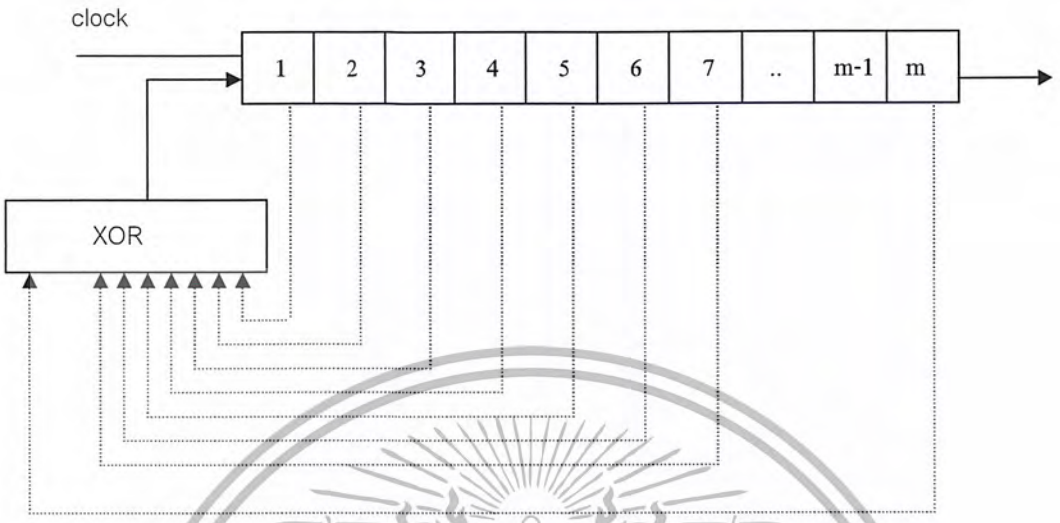
- รหัส PN สั้น รหัสนี้มีอยู่ 2 ชุด คือ I และ Q โดยแต่ละชุดสร้างจากการใช้ชิฟต์รีจิสเตอร์ขนาด 15 บิท โดยต่อเชื่อมกันให้มีโครงสร้างแบบ Maximum Lenrth LFSRs และใช้โพลิโนเมียล 2 ชุด ต่อไปนี้

$$I(x) = x^{15} + x^{13} + x^{9} + x^{8} + x^{7} + x^{5} + x^{25} + 1 \tag{2.6}$$

$$Q(x) = x^{15} + x^{12} + x^{11} + x^{10} + x^{6} + x^{5} + x^{4} + x^{3} + 1 \tag{2.7}$$

ลักษณะของรหัส PN จะมีการกระจายข้อมูลคล้ายๆแบบสุ่ม แต่ไม่ได้เป็นการสุ่มจริง คือ ข้อมูลจะมีรูปแบบที่แน่นอนสามารถทำนายได้ หากทราบกระบวนการสร้างรหัสและในที่สุดก็จะกลับมาซ้ำค่าเดิมเสมอ เพื่อทำให้สามารถคืนค่าเดิมของข้อมูลกลับมาได้ถูกต้องในส่วนของภาครับโดยปกติจะใช้รหัส PN ที่มีความยาวมากเพื่อให้มีการกระจายของข้อมูลสูง การใช้งานรหัส PN แม้ว่าจะเป็นข้อมูลที่ทราบค่าล่วงหน้าแล้วแต่เนื่องจากรหัส PN ซึ่งใช้เพียงชุดเดียวแต่ก็มีขนาดที่ยาวมาก จึงนิยมสร้างจากวงจรแทนการเก็บเป็นค่าคงที่ซึ่งเปลืองทรัพยากรของระบบมากกว่า เรียกวงจรที่ใช้สร้างรหัส PN ว่า วงจรแทนการเก็บเป็นค่าคงที่ซึ่งเปลืองทรัพยากรของระบบมากกว่า เรียก วงจรที่ใช้สร้างรหัส PN ว่า วงจร PN Generator โดยมีโครงสร้างของวงจรเป็นดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 โครงสร้างของวงจร PN Generator

หากพิจารณาโครงสร้างของวงจรจะพบว่าประกอบด้วยชิฟต์รีจิสเตอร์(Shift Register)และการกระทำทางตรรกะแบบ XOR ของสัญญาณจากบิทต่างๆของชิฟต์รีจิสเตอร์ ได้เอาท์พุทของบิทท้ายสุดเป็นเอาท์พุทของวงจร มีสิ่งที่ต้องระวัง คือ ในจังหวะเริ่มต้นการทำงานของวงจร ข้อมูลในชิฟต์รีจิสเตอร์ต้องมีอย่างน้อยที่สุดบิทหนึ่งที่เป็น '1' ไม่เช่นนั้นแล้ววงจร PN Generator จะไม่ให้เอาท์พุทใดๆออกมาเลย

สำหรับความยาวของรหัสที่สร้างจากวงจร PN Generator ก่อนที่จะกลับมาซ้ำค่าเดิมจะขึ้นอยู่กับขนาดของชิฟต์รีจิสเตอร์ที่ใช้ตามสมการ

$$\text{ความยาวของรหัส} = 2^m - 1$$

โดยที่ m คือ ขนาดของชิฟต์รีจิสเตอร์ที่ใช้

สำหรับการที่จะเลือกว่าจะใช้สัญญาณบิทใดของชิฟต์รีจิสเตอร์ในการทำ XOR สามารถใช้ได้ตามตารางที่ 2.1 ซึ่งจะแบ่งตามขนาดของชิฟต์รีจิสเตอร์ที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

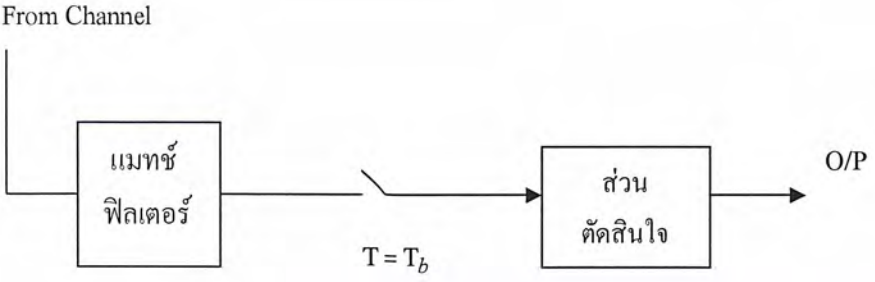
ขนาดของชิพต์ รีจีสเตอร์(m)	ค่าสัมประสิทธ์ที่สามารถใช้ได้
2	[1,2]
3	[1,3]
4	[1,4]
5	[2,5][2,3,4][1,2,4,5]
6	[1,6][1,2,5,6][2,3,5,6]
7	[3,7][1,2,3,7][1,2,3,4,5,6,7][2,3,4,7] [1,2,3,4,5,7][2,4,6,7][1,7][1,3,6,7] [2,5,6,7]
8	[2,3,4,8][3,5,6,8][1,2,5,6,8] [1,3,5,8][2,5,6,8][1,5,6,8] [1,2,3,4,6,8][1,6,7,8]
9	[4,9][3,4,6,9][4,5,8,9] [1,4,8,9][2,3,5,9][1,2,5,6,9] [5,6,8,9][1,3,4,6,7,9][2,7,8,9]
10	[3,10,][2,3,8,10][3,4,5,6,7,8,9,10] [1,2,3,5,6,10][2,3,6,8,9,10][1,3,4,5,6,7,8,10]
11	[2,11][2,5,8,11][2,3,7,11] [2,3,5,11][1,3,8,9,10,11]
12	[1,4,6,12][1,2,5,7,8,9,11,12] [1,3,4,6,8,10,11,12][1,2,5,10,11,12] [2,3,9,12][1,2,4,6,11,12]
13	[1,3,4,13][4,5,7,9,10,13][1,4,7,8,11,13] [1,2,3,6,8,9,10,13][5,6,7,8,12,12][1,5,7,8,9,13]
14	[1,6,10,14][3,4,6,7,9,10,14] [4,5,6,7,8,9,12,14][1,6,8,14] [5,6,9,10,11,12,13,14][1,2,3,4,5,7,8,10,13,14]
15	[1,15][1,5,10,15][1,3,12,15] [1,2,4,5,10,15][1,2,6,7,11,15][1,2,3,6,7,15]
16	[1,3,12,16][1,3,6,7,11,12,13,16] [1,2,4,6,8,9,10,11,15,16][1,6,8,14]

ตารางที่ 2.1 แสดงค่าสัมประสิทธิ์ตามขนาดชิพต์รีจิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ภาครับของระบบไดเรกต์ซีเควนซ์ซีดีเอ็มเอ

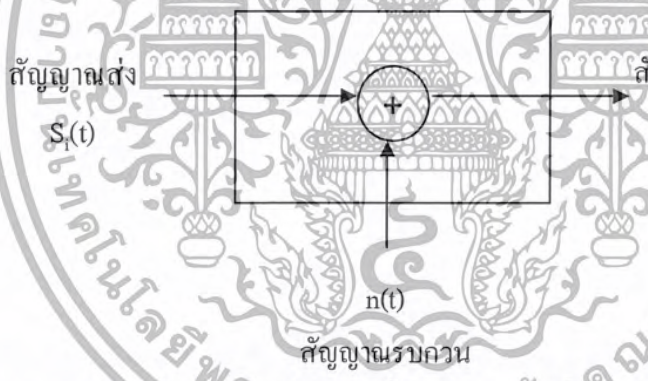
บล็อกไดอะแกรมของภาครับระบบไดเรกต์ซีเควนซ์ซีดีเอ็มเอ



รูปที่ 2.3 บล็อกไดอะแกรมภาครับของระบบไดเรกต์ซีเควนซ์ซีดีเอ็มเอ

2.4 แมทช์ฟิลเตอร์(Matched Filter)

ขั้นตอนการคีมอดูเลตสัญญาณซึ่งเป็นกระบวนการที่เกิดขึ้นที่ภาครับ โดยเราจะพิจารณาเฉพาะกรณีที่ช่องสัญญาณได้รับผลกระทบจากสัญญาณรบกวนเกาส์สีขาวแบบบวก(Additive White Gaussian Noise:AWGN)จากข้อกำหนดที่กล่าวมาเราสามารถวาดเป็นแบบจำลองช่องสัญญาณได้ดังรูปที่2.4



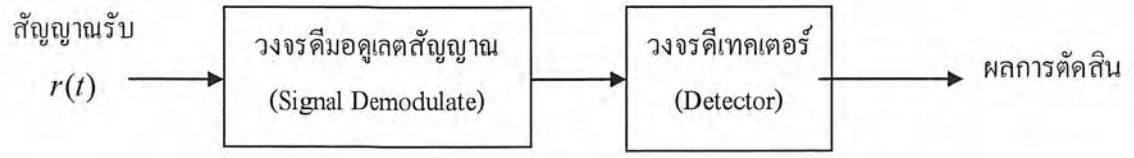
รูปที่2.4 แผนภาพแสดงความสัมพันธ์ของสัญญาณที่ภาคส่งและภาครับเมื่อส่งผ่านช่องสัญญาณ AWGN

จากรูปเราจะได้ความสัมพันธ์ระหว่างสัญญาณที่ส่งออก ณ ภาคส่งกับสัญญาณที่รับได้ ณ ภาครับดังนี้

$$r(t) = S_i(t) + n(t) \qquad , 0 \le t \le T \tag{2.8}$$

โดย $S_i(t)$ คือ สัญญาณที่ภาคส่งใช้ในการแทนข้อมูลดิจิตอล ทั้งนี้ i = 1,2,….,M ระยะเวลาที่ใช้ในการส่งสัญญาณดิจิตอลแต่ละครั้งจะมีความยาวเท่ากับ Tวินาที และ $n(t)$ คือ สัญญาณรบกวนสำหรับช่วงเวลาการส่งข้อมูลดิจิตอลแต่ละครั้ง $n(t)$ เป็นฟังก์ชันแซมเปิลของกระบวนการแรนดอมเกาส์สีขาวแบบบวก (AWGN random process)ที่มีความหนาแน่นสเปกตรัมกำลัง(power spectral density)เท่ากับ $\frac{No}{2}$ W/Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



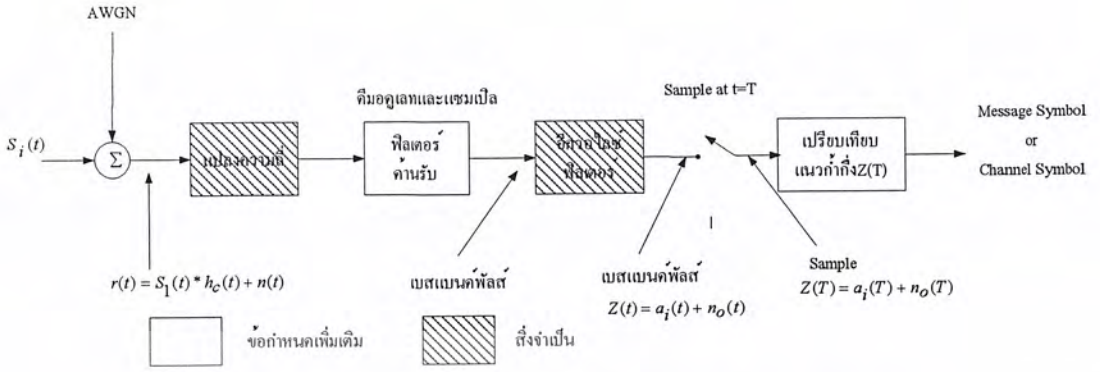
รูปที่ 2.5 แผนภาพแสดงโครงสร้างของวงจรภาครับสัญญาณ

สัญญาณ $r(t)$ ที่รับได้จะถูกป้อนเข้าวงจรภาครับซึ่งประกอบด้วยองค์ประกอบย่อย 2 ส่วน คือ วงจรดีมอดูเลตสัญญาณ(signal demodulator) และวงจรดีเทคเตอร์(detector) ดังแสดงในรูปที่ 2.5 สำหรับองค์ประกอบส่วนแรกคือ วงจรดีมอดูเลตสัญญาณ โดยทั่วไปวงจรดีมอดูเลตมีได้หลายรูปแบบ สำหรับวงจรที่เราจะอธิบายในรายละเอียดมีชื่อเรียกว่า วงจรดีมอดูเลตแบบแมทช์ฟิลเตอร์ (matched filter demodulator) ส่วนองค์ประกอบในส่วนที่สองคือ วงจรดีเทคเตอร์ ซึ่งหน้าที่หลักของวงจรดีเทคเตอร์คือการนำชุดตัวแปรแรนดอม(random) $r_1, r_2, \ldots, r_N$ ที่ได้จากด้านออกของวงจรดีมอดูเลเตอร์มาประมวลเพื่อใช้ในการตัดสินใจว่าสัญญาณที่ได้รับนั้นน่าจะเป็นสัญญาณดิจิตอลชุดใดที่ต้นทางส่งออก กรรมวิธีที่ใช้ในการตัดสินใจมีพื้นฐานการคำนวณโดยอาศัยทฤษฎีความน่าจะเป็นเป็นหลัก

แมทช์ฟิลเตอร์เป็นฟิลเตอร์แบบเชิงเส้น (linear filter)ที่ออกแบบให้มีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน (signal-to-noise ratio) สูงๆ และเป็นฟิลเตอร์ที่ดีที่สุดสำหรับใช้รับข้อมูล(data)ตัวเดียวด้วยการที่ผลตอบสนองอิมพัลส์(impulse response)แมทช์(matches)กับรูปคลื่น(waveform)ของบิทข้อมูล(data bit)ที่รับได้ เพียงเพราะว่าผลตอบสนองอิมพัลส์เป็นการรีเวอร์ส(reverse)รูปคลื่นของข้อมูลในทางเวลา ดังนั้นแมทช์ฟิลเตอร์ก็จะมีผลตอบสนองความถี่(frequency response)ที่เหมือนกับสเปกตรัมความถี่(frequency spectrum)ของสัญญาณที่รับได้ด้วย เราต้องการฟิลเตอร์ที่มีการส่งผ่าน(transmission)ขนาดใหญ่และรูปคลื่นที่รับได้มีกำลังงาน(energy)สูงๆ แต่ในส่วนของความถี่เราต้องการให้รูปคลื่นที่รับได้มีกำลังงานน้อยๆเพราะฟิลเตอร์จะทำให้เกิดการลดทอน(attenuation) เพื่อป้องกันสัญญาณรบกวนที่จะเข้าไปในดีเทคเตอร์ โดยสิ่งที่เราต้องการนี้สามารถทำได้โดยใช้เงื่อนไขของแมทช์ฟิลเตอร์ แต่ข้อกำหนดของแมทช์ฟิลเตอร์นั้นจะอยู่บนพื้นฐานของระบบที่ช่องสัญญาณได้รับผลกระทบจากสัญญาณรบกวนเกาส์สีขาวแบบบวกเท่านั้น

สมมติให้มีสัญญาณ $S(t)$ บวกกับสัญญาณรบกวนเกาส์สีขาวแบบบวก $n(t)$ เป็นอินพุทของฟิลเตอร์แบบเชิงเส้นที่แสดงในรูปที่ 2.6 ที่เวลา t = T เอาท์พุทแซมเปิล Z(T) จะประกอบด้วยส่วนประกอบของสัญญาณ a_i และส่วนประกอบของสัญญาณรบกวน n_o ค่าความแปรปรวน(variance)ของสัญญาณรบกวน(กำลังเฉลี่ยของสัญญาณรบกวน)จะแทนด้วยสัญลักษณ์ σ_o^2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ขั้นตอน demodulation/detector ของสัญญาณดิจิตอล

ดังนั้นอัตราส่วนระหว่างกำลังสัญญาณที่เกิดขึ้นในช่วงเวลาสั้นๆต่อกำลังเฉลี่ยของสัญญาณรบกวน $\left(\frac{S}{N}\right)$ ที่เวลา $t = \mathrm{T}$ เอาท์พุทแซมเปิลในขั้นตอน(step)ที่ 1 คือ

$$\left(\frac{S}{N}\right)_{\mathrm{T}} = \frac{a_i^{\,2}}{\sigma_o^{\,2}} \tag{2.9}$$

จะพบว่าฟังก์ชันถ่ายโอน(transfer function) $H_o(f)$ ที่มีค่ามากที่สุดตามสมการที่ (2.5) นั้น สามารถแสดงได้โดยให้สัญญาณเอาท์พุทของฟิลเตอร์ (a_i) อยู่ในเทอมฟังก์ชันถ่ายโอน $H(f)$ และสัญญาณอินพุทเป็นการแปลงฟูเรียร์(Fourier transform)ของสัญญาณอินพุท $S(t)$

$$a_i(t) = \int_{-\infty}^{\infty} H(f)S(f)e^{j2\pi ft}df \tag{2.10}$$

โดยที่ $S(f)$ คือ การแปลงฟูเรียร์ของสัญญาณอินพุท$[S(t)]$

ถ้าเป็นความหนาแน่นกำลังเชิงสเปกตรัมแบบสเปกตรัมสองข้าง (two - side power spectral density)ของอินพุทของสัญญาณรบกวน คือ $\frac{No}{2}$ (W/Hz) ดังนั้นเราสามารถที่จะแสดงกำลังเอาท์พุทของสัญญาณรบกวน(output noise power)

$$\sigma_o^{\,2} = \frac{No}{2}\int_{-\infty}^{\infty}|H(f)|^2 df \tag{2.11}$$

รวมสมการที่ (2.11) ถึง (2.13) จะได้เป็น

$$\left(\frac{S}{N}\right)_{\mathrm{T}} = \frac{\left|\int_{-\infty}^{\infty} H(f)S(f)e^{j2\pi fT}df\right|^2}{\frac{No}{2}\int_{-\infty}^{\infty}|H(f)|^2 df} \tag{2.12}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะพบว่าค่าของ $H(f) = H_o(f)$ ซึ่งจะทำให้ได้ค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน $\left(\frac{S}{N}\right)_T$ สูงๆ โดยใช้ความไม่เท่ากันของชวาร์ท(Schwarz's inequality)แบบแรกของความไม่เท่ากันมีรูปแบบเป็น

$$\left|\int_{-\infty}^{\infty} f_1(x) f_2(x) dx\right|^2 \leq \int_{-\infty}^{\infty}\left|f_1(x)\right|^2 dx \int_{-\infty}^{\infty}\left|f_2(x)\right|^2 dx \tag{2.13}$$

ถ้า $f_1(x) = k f_2{}^*(x)$ โดยที่ k คือ ค่าคงที่ , * เป็นคอมเพล็กซ์คอนจูเกต(complex conjugate) ถ้าแทน $f_1(x)$ และ $f_2(x)$ ด้วย $H(f)$ และ $S(f)$ ตามลำดับ จะได้

$$\left|\int_{-\infty}^{\infty} H(f)S(f)e^{j2\pi fT} df\right|^2 \leq \int_{-\infty}^{\infty}\left|H(f)\right|^2 df \int_{-\infty}^{\infty}\left|S(f)\right|^2 df \tag{2.14}$$

แทนในสมการที่ 2.7

$$\left(\frac{S}{N}\right)_T \leq \frac{2}{N_0}\int_{-\infty}^{\infty}\left|S(f)\right|^2 df \tag{2.15}$$

หรือ

$$\max\left(\frac{S}{N}\right)_T = \frac{2E}{No} \tag{2.16}$$

$$\mathrm{E} = \int_{-\infty}^{\infty}\left|S(f)\right|^2 df \tag{2.17}$$

โดยที่ E คือ กำลังงานของสัญญาณอินพุท $[S(t)]$

ค่าสูงสุดของเอาท์พุทของอัตราส่วนสัญญาณต่อสัญญาณรบกวนจะขึ้นอยู่กับกำลังงานของสัญญาณอินพุทและความหนาแน่นสเปกตรัมกำลัง โดยจะไม่ขึ้นอยู่กับรูปร่างลักษณะเฉพาะของรูปคลื่นที่ใช้ ในสมการที่ (2.11) จะมีค่าเท่ากับสมการที่ (2.13) เมื่อใช้ฟังก์ชันถ่ายโอนของฟิลเตอร์ที่ดีที่สุด(optimum filter)

$$H(f) = H_0(f) = kS^*(f)e^{-j2\pi fT} \tag{2.18}$$

หรือ

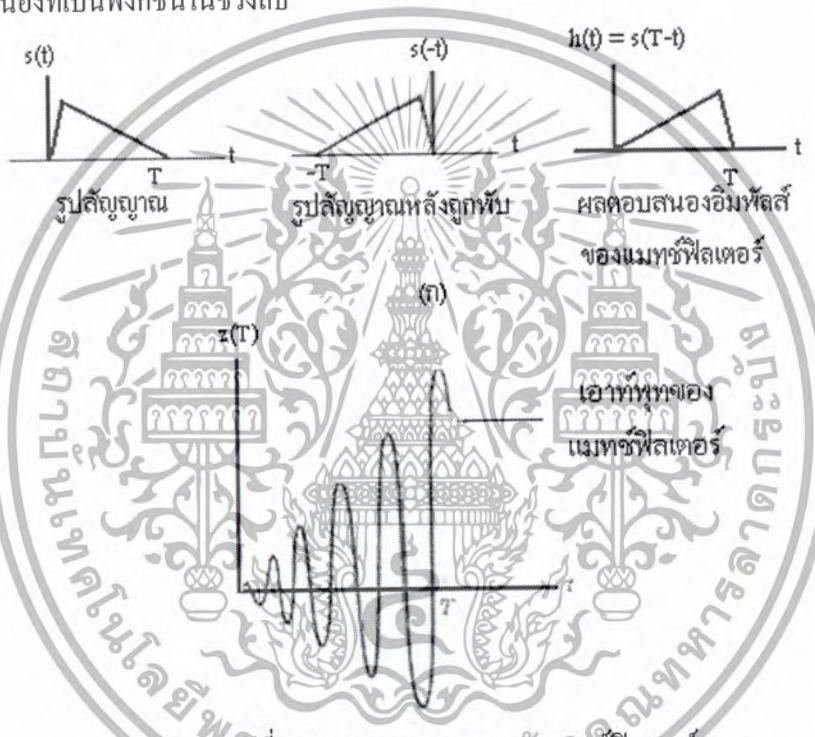
$$h(f) = \mathcal{F}^{-1}\left\{kS^*(f)e^{-j2\pi fT}\right\} \tag{2.19}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก $S(t)$ เป็นค่าจำนวนจริง(real) สามารถเขียนได้เป็น

$$h(t) = \begin{cases} ks(T-t).....0 \le t \le T \\ 0.....\text{อื่นๆ} \end{cases} \quad (2.20)$$

ดังนั้นผลตอบสนองอิมพัลส์ของฟิลเตอร์ที่ให้ค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนสูงๆ จะเป็นเหมือนกระจกเงาสะท้อนสัญญาณ $S(t)$ สัญญาณ $S(t)$ จะถูกหน่วงเวลา(delay) ไปช่วงเวลา T โดยหน่วงเวลา T วินาที ตามสมการ (2.20) การหน่วงเวลา T วินาที ที่ทำเป็นฟังก์ชัน $h(t)$ อยู่ในช่วงเวลา $0 \le t \le$ T ที่เป็นค่าบวก จะไม่มีการหน่วงเวลา T ที่เป็นตอบสนอง $S(-t)$ เพราะว่าเป็นการบอกลักษณะของผลตอบสนองที่เป็นฟังก์ชันในช่วงลบ



รูปที่ 2.7(ก) characteristic ของแมทช์ฟิลเตอร์
(ข) เอาท์พุทของแมทช์ฟิลเตอร์

จากสมการที่ (2.20) และรูปที่ 2.7 (ก) เป็นคุณสมบัติพื้นฐานของแมทช์ฟิลเตอร์และผลตอบสนองอิมพัลส์ของฟิลเตอร์จะได้จากการหน่วงเวลาสัญญาณ $S(t)$ โดยทำการพลิกเฟสให้ได้เป็นสัญญาณ $S(-t)$ และทำการหน่วงเวลาไป T วินาที ได้เป็น $S(t-\mathrm{T})$

เอาท์พุท Z(T) ของฟิลเตอร์ สามารถอธิบายได้ในโดเมนของเวลา(time domain)คือ เป็นการคอนโวลูชั่น(convolution)ระหว่างรูปคลื่นของสัญญาณอินพุท r(t) กับผลตอบสนองอิมพัลส์ของฟิลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

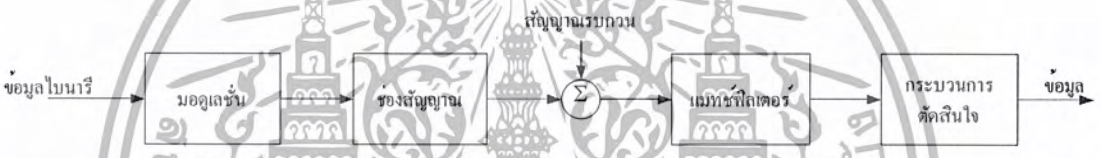
$$Z(t) = r(t) * h(t)$$
$$= \int_0^t r(\tau)h(T-\tau)d\tau \quad (2.21)$$

แทน $h(t)$ ของสมการ (2.20) ใน $h(t-\tau)$ ของสมการ(2.21)

$$Z(t) = \int_0^t r(\tau)S[T-(t-\tau)]d\tau$$
$$= \int_0^t r(\tau)S[T-t+\tau]d\tau \quad (2.22)$$

เมื่อ $t = T$ จะได้

$$Z(T) = \int_0^T r(\tau)S(\tau)d\tau \quad (2.23)$$



รูปที่ 2.8 บล็อกไดอะแกรมของระบบ baseband

พิจารณา ระบบแบนด์มูลฐาน (baseband) ดังแสดงในรูป 2.8 รูปคลื่น ที่รับได้ คือ $S_1(t)$ หรือ $S_2(t)$ ซึ่งจะตัดสินจากบิทข้อมูลที่ส่งมา สามารถแสดงได้โดยถ้าไม่มีการรบกวนระหว่างสัญลักษณ์ (symbol) ผลลัพธ์ที่ได้จะมีอัตราความผิดพลาด (error rate) ต่ำที่สุดถ้าแมทช์ฟิลเตอร์ออกแบบให้มีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน(γ) สูงที่สุด

$$\gamma = \frac{\left[S_2(T_b) - S_2(T_b)\right]^2}{N} \quad (2.24)$$

เมื่อ T_b = เป็นเวลาแซมปลิ้ง(sampling) ที่บิทสุดท้ายของช่วง

N = พลังงานรวมของสัญญาณรบกวนที่วงจรตัดสินใจ

โดยที่ฟังก์ชันนี้จะมีค่าสูงสุดถ้าผลตอบสนองความถี่ของแมทช์ฟิลเตอร์ มีค่าดังสมการ (2.25)

$$H(f) = \frac{KP^*(f)e^{-j2\pi fT_b}}{G_n(f)} \quad (2.25)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$P^*(f)$ = คอมเพล็กซ์คอนจูเกตของสเปกตรัมความถี่ของรูปคลื่น $P(t) = s_2(t) - s_1(t)$

$G_n(f)$ = ความหนาแน่นกำลังเชิงสเปกตรัมแบบสเปกตรัมสองข้างของสัญญาณรบกวน สำหรับสัญญาณรบกวนเกาส์สีขาว, $G_n(f)$ จะมีค่าคงที่ = $\frac{No}{2}$

K = ตัวแปรอิสระภายใต้เงื่อนไขนี้ ค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน จะมีค่าตามสมการ (2.26)

$$\gamma_{MAX} = \int_{-\infty}^{\infty} \frac{|P(f)|^2}{|G_n(f)|} df \tag{2.26}$$

และค่าความน่าจะเป็นของความผิดพลาด(probability of error) สำหรับการตัดสินใจเลือกจุดกึ่งกลางระหว่างค่าที่เป็นไปได้ที่รับได้ 2 ค่า ตามสมการที่ (2.27)

$$P_e = Q\left[\frac{\sqrt{\gamma_{MAX}}}{2}\right] \tag{2.27}$$

โดยที่ Q(x) เป็นสัญลักษณ์แสดงถึงฟังก์ชัน Q ที่มีค่าตามสมการ (2.28)

$$Q(x) = \int_{x}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz \tag{2.28}$$

ผลตอบสนองอิมพัลส์ของฟิลเตอร์เป็นการแปลงฟูเรียร์ผกผัน(Inverse Fourier Transform) ของผลตอบสนองความถี่ สำหรับกรณีของสัญญาณรบกวนสีขาว ในสมการ (2.25) ถ้าเราให้ K = $\frac{No}{2}$ จะได้ค่าตามสมการ (2.29)

$$H(f) = P^*(f)e^{-j2\pi T_b} \tag{2.29}$$

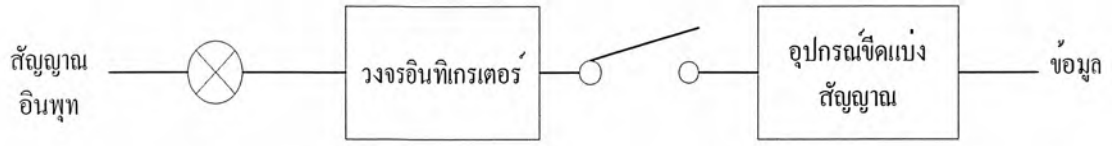
เราได้แปลง $P^*(f)$ จาก $P(-t)$ เพราะฉะนั้นผลตอบสนองอิมพัลส์ของแมทช์ฟิลเตอร์จะได้

$$h(t) = p(T_b - t) = S_2(T_b - t) - S_2(T_b - t) \tag{2.30}$$

เหมือนกับสัญญาณแบนด์มูลฐาน ที่ใช้ในระบบความถี่วิทยุ(Radio Frequency: RF) การใช้เครื่องรับ คอร์รีเลชั่น(correlation) เช่นแสดงในรูป 2.9 ซึ่งเครื่องรับชนิดนี้จะแสดงภายหลังสามารถอิมพลีเมนท์(implement) สำหรับเฟสชิฟต์คีย์อิงดีมอดูเลเตอร์(PSK demodulator) ซึ่งเป็นการคอร์รีเลชั่นสัญญาณ $S_2(t) - S_1(t)$ ซึ่งถูกผลิตโดยเฟสล็อกลูป(phase locked loop)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 2.9 เครื่องรับแบบ correlation

ใช้สมการ (2.26) และ (2.27) จะแสดงในรูปของการคำนวณอัตราความผิดพลาดของบิท (Bit Error Rate:BER)สำหรับเลขฐานสองของระบบเฟสชิพต์คีย์อิง ซึ่งทั้งสองสัญญาณ ที่จะถูกส่งเป็น

$$S_1(t) = -\mathrm{A}\sin\omega_c t$$

หรือ

$$S_2(t) = +\mathrm{A}\sin\omega_c(t)$$

ดังนั้น $P(t) = 2\mathrm{A}\sin\omega_c(t)$ ขณะนี้เราพิจารณาสมการ (2.26) ซึ่งนำมาใช้กับกรณีของแมทช์ฟิลเตอร์ความหนาแน่นสเปกตรัมกำลังของสัญญาณรบกวนเกาส์สีขาว(White Gaussian noise power spectral density) ถูกแสดงโดย $G_n(t) = \frac{No}{2}$ และ

$$\gamma_{\mathrm{MAX}} = \frac{\int_{-\infty}^{\infty}|P(f)|^2}{\frac{N_0}{2}}df \qquad (2.31)$$

เราสามารถประยุกต์ทฤษฎีของพาร์เซวาล (Parseval's theorem) ตามสมการ (2.31) ทฤษฎีนี้พลังงานของสัญญาณสามารถหา โดยการอินทิเกรทความหนาแน่นสเปกตรัมกำลังตลอดทั้งความถี่หรืออินทิเกรทฟังก์ชันทางเวลาของสี่เหลี่ยม(squared time function) ตลอดทั้งช่วงเวลา

$$\int_{-\infty}^{\infty}|P(f)|^2 df = \int_{-\infty}^{\infty}p^2(t)dt \qquad (2.32)$$

ในกรณีนี้ $P(t) = 4\mathrm{A}^2\sin^2\omega_c(t)$ และมันจะมีค่าอยู่ในช่วง $0 \le t \le \mathrm{T_b}$ และแทนสมการ (2.32) ในสมการ (2.31) จะได้

$$\gamma_{MAX} = \frac{\int_0^{T_b}4A^2\sin(\omega_c)dt}{\frac{N_0}{2}} \qquad (2.33)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
54930

การอินทิเกรทจะได้ค่าโดยการขยายเทอม $\sin^2 \omega t = \frac{1}{2} - \frac{1}{2}\cos \omega t$ และโดยการสังเกตในเทอมของ โค-ซายน์(cosine) อินทิเกรทแล้วจะได้ค่าเท่ากับศูนย์ ถ้าความถี่คลื่นพาห์เป็นผลคูณของอัตราข้อมูล ดังนั้นจะได้

$$\gamma_{\max} = \frac{4A^{2T_b}}{N_0} \tag{2.34}$$

แทนสมการนี้ลงในสมการ (2.27) จะได้

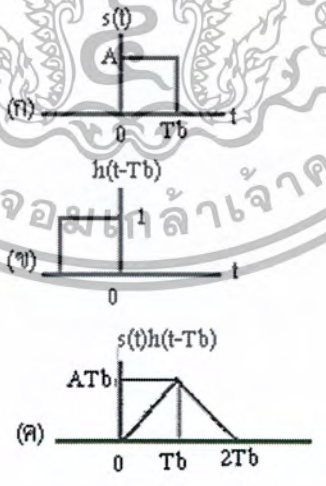
$$P_e = Q\left[\sqrt{\frac{A^2 T_b}{N_0}}\right] \tag{2.35}$$

P_e ที่แสดงให้เห็นพบบ่อยในเทอมของกำลังงานต่อบิท (E_b) หารด้วยความหนาแน่นของสัญญาณรบกวนแบบข้างเดียว (one –sided noise density : N_o) กำลังงานต่อบิทในกรณีนี้ให้

$$E_b = \mathrm{A}^2 \frac{T_b}{2}$$

ดังนั้นจะได้

$$P_e = Q\left[\sqrt{\frac{2E_b}{N_0}}\right] \tag{2.36}$$



รูปที่ 2.10 (ก) รูปคลื่นของสัญญาณ square wave

(ข) impulse response ของแมทช์ฟิลเตอร์

(ค) output ของแมทช์ฟิลเตอร์ที่มีอินพุทเป็น square wave

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำความสัมพันธ์ตามสมการ (2.36) มาพล็อตจะได้กราฟในรูป 2.10 ในอนาล็อกความน่าจะเป็นของความผิดพลาดมีการมอดูเลทหลายชนิด $\frac{E_b}{N_o}$ เป็นเหมือนกับอัตราส่วนสัญญาณต่อสัญญาณรบกวนในแมทช์ฟิลเตอร์ พลังงานของสัญญาณกำหนดเป็น $S = \frac{A^2}{2}$ แบนด์วิดท์(bandwidth)ของแมทช์ฟิลเตอร์ เป็น 1.5เท่าของอัตราข้อมูลที่สัญญาณมูลฐาน เหตุนี้มันจะเท่ากับอัตราข้อมูล $\frac{1}{T_b}$ ที่ความถี่คลื่นพาห์ ดังนั้นพลังงานทั้งหมดของสัญญาณรบกวน $N = \frac{N_o}{T_b}$ แทนค่าในสมการ (2.35) จะได้

$$P_e = Q\left[\sqrt{\frac{2S}{N}}\right] \tag{2.37}$$

ในที่นี้ $\frac{S}{N}$ คือ อัตราส่วนสัญญาณต่อสัญญาณรบกวน ในแบนด์วิดท์จะเท่ากับอัตราข้อมูล

2.4.1รูปพัลส์ในทางอุดมคติของแมทช์ฟิลเตอร์

ถึงแม้ว่าแมทช์ฟิลเตอร์จะให้ผลลัพธ์ที่ดีที่สุดในกรณีที่มีสัญญาณรบกวนเกาส์สีขาวแบบบวกรวมอยู่ด้วยแต่ก็ยังคงเกิดปัญหาขึ้นเมื่อใช้พัลส์รูปสี่เหลี่ยม เมื่อพิจารณาถึงทฤษฎีของฟูเรียร์ทรานฟอร์มแล้วพัลส์รูปสี่เหลี่ยมในทางโดเมนเวลา(Time Domain)จะเหมือนกับพัลส์ซิงค์(sinc pulse) ในโดเมนความถี่ที่ส่วนปลายของพัลส์ซิงค์จะขยายไปถึงอนันต์(infinity)ซึ่งจะเป็นรูปพัลส์ที่ในระบบที่มีแบนด์วิดท์จำกัดต้องการ

รูปพัลส์ในทางอุดมคติควรจะมี 2 คุณลักษณะ คือ จะมีแบนด์วิดท์จำกัดที่ในระบบการส่งแบบแบนด์ลิมิต(band limited)ยอมรับได้และมีค่าการสอดแทรกระหว่างสัญลักษณ์(Intersymbol Interference)เป็นศูนย์ถ้าสุ่มสัญญาณในช่วงเวลาที่เหมาะสม

2.4.2รูปพัลส์ในทางปฏิบัติของแมทช์ฟิลเตอร์

ถึงแม้ว่าซิงค์พัลส์จะมีลักษณะเหมือนกับพัลส์ในทางอุดมคติแต่จะไม่สามารถอิมพลีเมนท์(implement)ได้จริงในทางปฏิบัติเพราะว่าพัลส์ขยายไปในช่วงเวลาอนันต์ ซึ่งจะทำให้ในโดเมนความถี่เกิดสัญญาณที่ไม่ต่อเนื่องขึ้น อย่างไรก็ตามรูปสัญญาณพัลส์ในทางปฏิบัติสามารถปรับเปลี่ยนได้โดยโรลออฟ(roll-off)ของสเปกตรัม

รูปพัลส์ชนิดหนึ่งที่มีคุณลักษณะเหมือนกับซิงค์พัลส์แต่จะไม่มีสัญญาณที่ไม่ต่อเนื่องในโดเมนความถี่ซึ่งก็คือ พัลส์เรสต์โคไซน์(Raise Cosine Pulse)

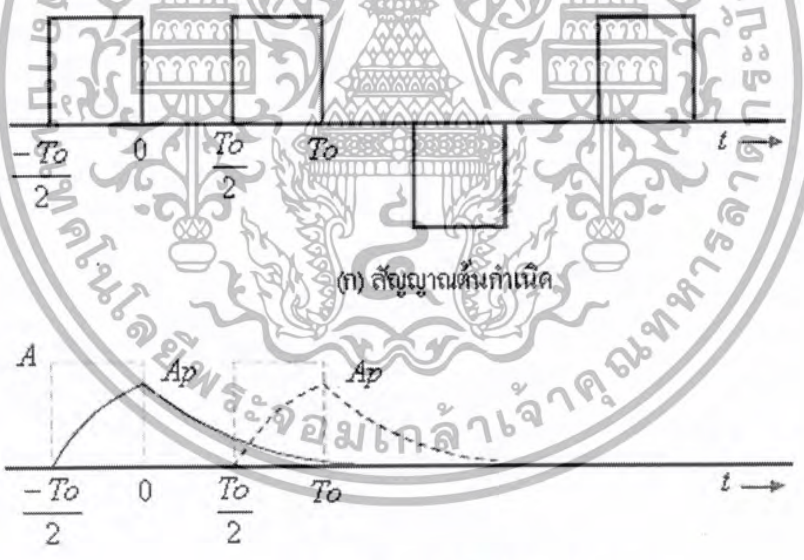
2.5 การสอดแทรกระหว่างสัญลักษณ์(Intersymbol Interference)

ในการสื่อสารระบบดิจิตอล ข้อมูลถูกส่งโดยการใช้พัลส์และตัวข้อมูลไม่ได้อาศัยอยู่กับรูปลักษณะของพัลส์แต่จะอาศัยอยู่ในขนาดหรือ ระดับเปรียบเทียบระหว่างค่าของพัลส์เหล่านั้น ดังนั้นวัตถุประสงค์ของการรับข้อมูลในระบบดิจิตอลจึงไม่ขึ้นอยู่กับการรับให้ได้พัลส์ที่มีรูปร่างเหมือนกับพัลส์ที่ส่งมาทุกประการ หลักสำคัญนั้นขึ้นกับว่า เมื่อรับพัลส์สัญญาณแล้วสามารถทำการตัดสินแยกค่าขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบระหว่างพัลส์ได้อย่างถูกต้อง เครื่องรับนั้นจะตัดสินค่าระดับของพัลส์ โดยการชักค่าตัวอย่างขนาดของแต่ละพัลส์ที่ตำแหน่งที่พัลส์ควรจะมีขนาดมากที่สุดและจะตัดสินใจโดยการเปรียบเทียบค่าตัวอย่างขนาดของพัลส์ที่ส่งมานั้นว่าควรจะมีค่าดิจิตอลเป็นเท่าใด

ในทางปฏิบัติพัลส์จะถูกส่งผ่านช่องสัญญาณที่มีแบนด์วิดท์จำกัด ดังนั้นส่วนประกอบของพัลส์ที่มีความถี่สูงจึงอาจไม่สามารถผ่านช่องสัญญาณไปยังเครื่องรับได้อย่างสมบูรณ์ เป็นเหตุทำให้เกิดการขยายแผ่ฐานของพัลส์กว้างกว่าเดิม ที่เรียกว่าเกิด ดิสเปอร์ชัน(dispersion)ของพัลส์ขึ้น รูปที่ 2.11แสดงตัวอย่างประกอบเรื่องนี้ โดยรูป 2.11(ก) แสดงตัวอย่างลักษณะของพัลส์ที่มีส่วนประกอบที่เป็นความถี่สูงอยู่แต่เมื่อพัลส์นี้ถูกส่งผ่านช่องสัญญาณที่มีแบนด์วิดท์จำกัดที่ทำให้ส่วนประกอบของความถี่สูงในตัวพัลส์ ตามรูป 2.11(ก) ถูกขจัดออกไปจึงทำให้ความคมบริเวณขอบพัลส์หายไป คือ จะทำให้การเพิ่มขึ้นของขอบหน้าและการลดลงของขอบหลังของพัลส์เกิดการช้าลง ดังแสดงในรูป 2.11(ข) การเกิดการขยายกว้างของฐานพัลส์หรือการเกิดดิสเปอร์ชันนี้จะมีค่ามากยิ่งขึ้นเมื่อแบนด์วิดท์ของช่องสัญญาณมีค่าแคบมากยิ่งขึ้น การเกิดดิสเปอร์ชันนี้จะทำให้เกิดการสอดแทรกของขอบฐานพัลส์ไปในพัลส์อื่นที่อยู่ข้างเคียง ดังแสดงในรูป 2.11(ข) ซึ่งการสอดแทรกนี้อาจจะก่อให้เกิดความผิดพลาดในการตัดสินค่าระดับขนาดของพัลส์ขึ้นได้ ซึ่งจะเป็นผลให้เกิดการแปรข้อมูลผิดพลาดตามขึ้นมาด้วย การเกิดการสอดแทรกระหว่างพัลส์เช่นนี้มีชื่อเรียกว่า การสอดแทรกระหว่างสัญลักษณ์ ซึ่งนิยมเรียกย่อว่า ไอเอสไอ(ISI)



(ก) สัญญาณต้นกำเนิด

(ข) สัญญาณที่เกิดดิสเปอร์ชัน

รูปที่ 2.11 ภาพอธิบายประกอบเรื่องการส่งสัญญาณผ่านช่องสัญญาณที่มีแบนด์วิดท์จำกัด

สมการที่ได้มาของแมทช์ฟิลเตอร์ต้องการสมมติให้ไม่มีการสอดแทรกของสัญญาณที่เดินทางมาในทิศทางที่แตกต่างกัน ในความหมายนี้เอาท์พุทของแมทช์ฟิลเตอร์ที่เวลาแซมปลิ้ง T_b บิทปัจจุบันจะต้องไม่ได้รับผลกระทบจากบิทก่อนหน้า การสอดแทรกของสัญญาณเป็นปัญหาที่สำคัญมากและเราต้องการให้การสอดแทรกของสัญญาณมีค่าน้อยๆ ผลกระทบของสัญญาณรบกวนทำให้เกิดการลดทอนของตัว-

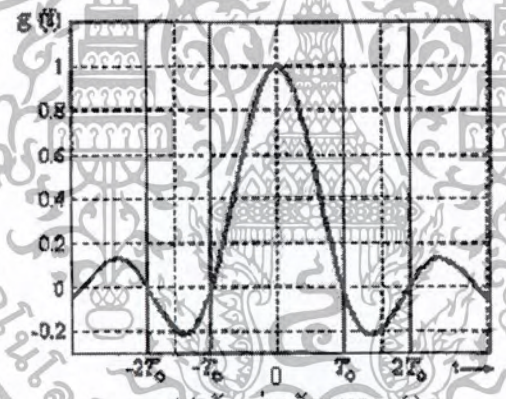
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลางในการส่ง ในกรณีที่เราพิจารณา ตัวกลางของสัญญาณที่ส่งไม่สนใจการลดทอน มันเป็นสิ่งจำเป็นที่จะเลือกรูปร่างของสัญญาณนั้นไม่มีสัญลักษณ์ของการลดทอน

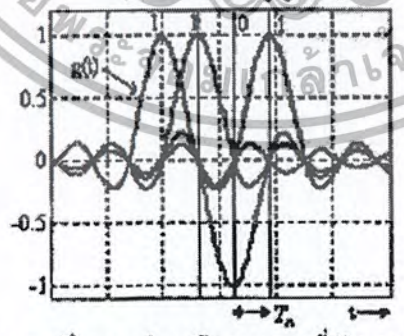
รูปคลื่นของสัญญาณเป็นรูปคลื่นสี่เหลี่ยม (square wave) ความยาวของบิท T_b แมทช์ฟิลเตอร์มีผลตอบสนองอิมพัลส์เป็นรูปสี่เหลี่ยม รูปคลื่นของสัญญาณที่แสดงในรูป 2.11 เอาท์พุทของแมทช์ฟิลเตอร์เป็นการ คอนโวลูชั่น(convolution) ระหว่างสัญญาณ 2 สัญญาณผลตอบสนองเอาท์พุทของแมทช์ฟิลเตอร์ (matched filter output response) มีค่าสูงสุดที่การชักตัวอย่างขณะนั้นจากการสังเกตุผลตอบสนอง เป็นศูนย์ที่ $2T_b$ ซึ่งจะเป็นการชักตัวอย่างถัดไปดังนั้นเงื่อนไขที่ต้องการของ แมทช์ฟิลเตอร์ที่เกี่ยวกับการสอดแทรกระหว่างสัญลักษณ์ แต่สเปกตรัมความถี่ของพัลส์สี่เหลี่ยม(square pulse) มีขนาดกว้างทำให้ใช้สเปกตรัมความถี่อย่างฟุ่มเฟือย สเปกตรัมความถี่ของพัลส์จะได้

$$S(f) = \mathrm{A}T_b\, e^{-j2\pi fT_b}\, \frac{\sin(\pi fT_b)}{\pi fT_b} \tag{2.38}$$

ผลลัพธ์ของพัลส์รูปคลื่นสี่เหลี่ยม โดยปกติอัตราข้อมูลจะมีค่าต่ำที่ความถี่ RF ในทางปฏิบัติสัญญาณของพัลส์ RF ที่ใช้ส่งมีคุณสมบัติเปิด (turn on) และปิด (turn off) สเปกตรัมความถี่สามารถหาได้จากสมการ (2.39)



(ก) ตัวอย่างสัญญาณ $g(t)$



(ข) แสดงถึงภาวะว่าจะมีขณะเวลาที่สัญญาณไม่รบกวนกัน

รูปที่ 2.12 การส่งสัญญาณพัลส์ด้วยอัตราไนควิสต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นที่รู้กันว่าถ้ารูปร่างของสัญญาณหรือพัลส์มีอยู่ในช่วงเวลาที่จำกัดอย่างสมบูรณ์แล้วแบนด์-วิดท์ของพัลส์หรือสัญญาณนั้นก็จะมีความกว้างอย่างไม่สิ้นสุด และในทางกลับกันเรารู้ว่าถ้าสัญญาณใดมีแบนด์ที่จำกัดแล้วสัญญาณนั้นจะต้องเกิดอยู่ในช่วงเวลาที่ยาวนานไม่สิ้นสุดเช่นกัน อย่างไรก็ตามถ้าเราสามารถเลือกรูปพัลส์ที่มีความหนาแน่นสเปกตรัมเชิงกำลังอยู่ในช่วงความถี่จำกัด (แบนด์วิดท์จำกัด) และสามารถเลือกทำการชักค่าตัวอย่างสัญญาณนั้นที่ขณะเวลาที่ทำให้ การสอดแทรกระหว่างสัญลักษณ์มีค่าเป็นศูนย์ หรือมีค่าต่ำมากๆได้ ก็จะมีผลให้เราสามารถที่จะสื่อสารกันได้ด้วยระบบดิจิตอลอย่างดี ด้วยเหตุที่ว่าการตัดสินสัญญาณในระบบดิจิตอลนั้นจะถูกหรือผิดก็ขึ้นอยู่กับค่าขนาดของตัวอย่างสัญญาณที่ถูกชักออกมาใช้ในการตัดสินใจ ยกตัวอย่างเช่น เมื่อเราเลือกพัลส์เป็น

$$g(t) = \quad Sa\left[\frac{\pi t}{T_o}\right] \tag{2.39}$$

ซึ่งมีลักษณะดังแสดงในรูป 2.12(ก) ดังนั้นถ้าเราเลือกทำการส่งพัลส์ดังกล่าวทุกช่วงเวลาที่ห่างกัน T_o แล้วก็จะไม่ทำให้เกิดการสอดแทรกระหว่างสัญลักษณ์ที่เวลา $\pm nT_o$ เลย(n คือเลขจำนวนจริงใดๆ) ซึ่งเห็นได้ชัดในรูป 2.12(ข) และเมื่อพิจารณาคุณสมบัติของการแปลงฟูเรียร์จะพบว่าแบนด์วิดท์ของการส่งพัลส์ดังกล่าวจะใช้แบนด์วิดท์เพียง $\frac{1}{2T_o}$ หรือ $\frac{f_o}{2}$ เท่านั้น $\left[f_o = \frac{1}{T_o}\right]$ ทั้งนี้เพราะ

$$Sa\left[\frac{\pi t}{T_o}\right] \longleftrightarrow T_o p\left[\frac{T_o \omega}{2\pi}\right] \tag{2.40}$$

จากสมการ(2.40)แสดงว่าพัลส์ $g(t) = Sa\left[\frac{\pi t}{T_o}\right]$ นั้นคือ ผลตอบสนองอิมพัลส์หนึ่งหน่วยของวงจรกรองความถี่ที่มี $H(\omega) = T_o p\left[\frac{T_o \omega}{2\pi}\right]$ ซึ่งก็คือ วงจรกรองความถี่ต่ำผ่านตามอุดมคติที่มีอัตราขยาย T_o และความถี่คัทออฟที่ $\omega_c = \frac{\pi}{T_o}$ หรือ $f_c = \frac{f_o}{2}$ นั่นเอง ดังนั้นเราจึงรู้ว่าถ้าเราทำการส่งพัลส์ $g(t) = Sa\left[\frac{\pi t}{T_o}\right]$ ผ่านวงจรกรองความถี่ต่ำผ่านที่มีแบนด์วิดท์ตามอุดมคติกว้างกว่า $\frac{f_o}{2}$ เฮิรตซ์ แล้วก็ย่อมจะไม่มีผลทำให้เกิดดิสเปอร์ชันกับพัลส์ $g(t)$ นั้นอย่างแน่นอน

ตามรูปที่ 2.12(ข) เราสมมติว่าได้ส่งพัลส์ $g(t)$ ตามสมการที่(2.39)แทนรหัสไบนารี 1101 โดยแต่ละพัลส์ห่างกัน T_o วินาที จะเห็นว่าตรงขณะเวลาที่แต่ละพัลส์มีค่าสูงสุดนั้น ค่าของพัลส์อื่นจะมีค่าเป็นศูนย์พอดี ดังนั้นถ้าเราเลือกชักค่าตัวอย่างสัญญาณที่เวลานี้ก็จะไม่มีการรบกวนจากอิทธิพลของการสอดแทรกระหว่างสัญลักษณ์จากพัลส์ข้างเคียงเลย

อัตราในการส่งพัลส์ $g(t)$ ที่มีลักษณะตามสมการ(2.39) จำนวน f_o พัลส์ต่อวินาที ผ่านช่องการสื่อสารที่มีแบนด์วิดท์ $\frac{f_o}{2}$ เฮิรตซ์นี้ถือว่าเป็นอัตราการส่ง อัตราไนควิสต์ แต่อย่างไรก็ตามในทางปฏิบัตินั้นเราไม่สามารถที่จะสร้างวงจรกรองความถี่ต่ำผ่านในอุดมคติขึ้นได้ ประกอบกับทั้งการชักค่าตัวอย่างออกจากพัลส์ $g(t)$ ที่เวลา nT_o ให้ถูกต้องจริงๆนั้นมีความวิกฤตมาก เพราะฐานของพัลส์ $g(t)$ ที่มีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สะบัดขึ้นลงตามเวลานั้นมีขนาดที่ลดลงช้ามาก จึงทำให้เกิดการสอดแทรกระหว่างสัญลักษณ์ได้ง่ายในเวลาที่ใกล้กับเวลา nT_o

เพื่อขยายผลให้ใช้ในทางปฏิบัติได้เกี่ยวกับเรื่องนี้ ไนควิสต์ได้ให้ข้อเสนอแนะไว้ว่า วงจรกรองความถี่ต่ำผ่านที่มีค่าฟังก์ชันถ่ายโอนแบบสมมาตรชนิดคี่(odd symmetry)กับแกนตั้งที่ความถี่จุดตัดของวงจรกรองความถี่ตามอุดมคตินั้นจะมีค่าเป็นศูนย์ที่ทุกขณะเวลาที่ห่างกัน T_o วินาที รูปพัลส์ในลักษณะของผลตอบสนองอิมพัลส์ดังกล่าวนิยมเรียกกันว่า รูปพัลส์แบบไนควิสต์(Nyquist pulse shape) รูปพัลส์ที่ได้มาจากผลตอบสนองอิมพัลส์ของระบบที่มีฟังก์ชันถ่ายโอนนี้เมื่ออยู่ในกลุ่มที่มีแกนสมมาตรเดียวกัน ก็จะถือว่าเป็นรูปพัลส์ที่มีความเท่าเทียม(equivalent)กัน เพราะว่าค่าฟังก์ชันพัลส์ในลักษณะดังกล่าวจะต้องมีลำดับในการชักค่าตัวอย่างสัญญาณที่เวลาห่างกัน T_o วินาทีเท่ากัน แม้ว่าฟังก์ชันถ่ายโอนเหล่านี้จะมีแบนด์วิดท์สัมบูรณ์ที่ต่างกันอยู่ คือ จาก $\frac{f_o}{2}$ ถึง f_o แต่เพราะมีผลตอบสนองอิมพัลส์ที่เท่าเทียมกัน ดังนั้นเพื่อง่ายกับการอ้างอิงถึง จึงได้มีการกำหนดเรียกชื่อแบนด์วิดท์ $\frac{f_o}{2}$ เฮิรตซ์ นี้ว่า แบนด์วิดท์ไนควิสต์ (Nyquist bandwidth)

2.6 พัลส์เรสต์โคไซน์(Raised Cosine Pulse)

เป็นพัลส์ที่มีสเปกตรัมความถี่เป็นของฟังก์ชันโคไซน์สแควร์(cosine square function) ซึ่งผลตอบสนองทางความถี่ของเรสต์โคไซน์แสดงดังรูป 2.15 และหาค่าได้จากสมการ

$$Y(f) = T_b \qquad 0 < |f| < \left(\frac{1}{2T_b} - \beta\right)$$

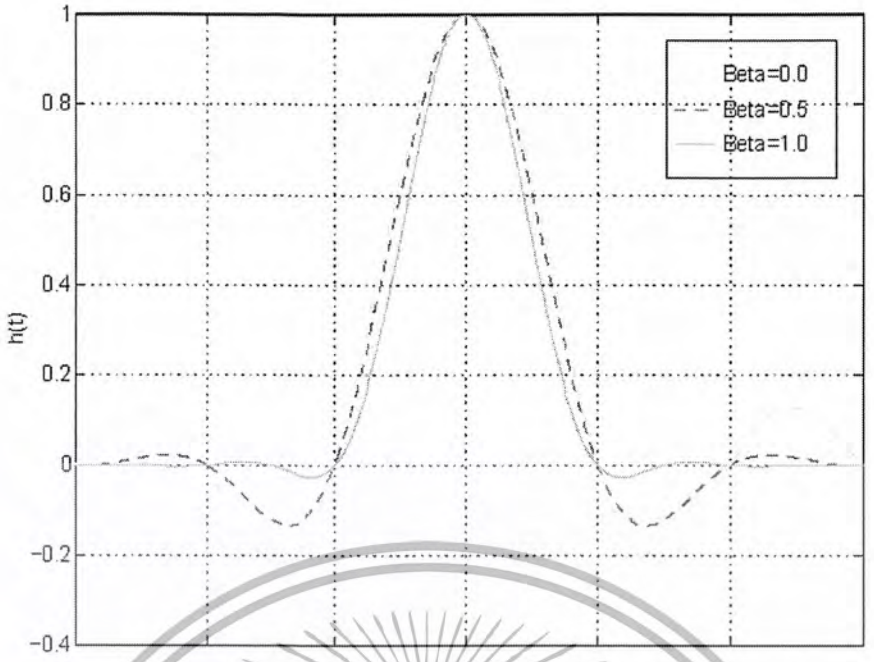
$$Y(f) = Tb\left[\frac{1}{2} + \frac{1}{2}\cos\frac{\pi}{2\beta}\left(f + \beta - \frac{1}{2Tb}\right)\right] \qquad \left(\frac{1}{2Tb} - \beta\right) < |f| < \left(\frac{1}{2T_b} + \beta\right)$$

$$Y(f) = 0 \qquad |f| > \left(\frac{1}{2T_b} + \beta\right) \qquad (2.41)$$

จากสมการ(2.41) เราสามารถเปลี่ยนรูปได้เป็น

$$Y(f) = T_b\left[\cos\frac{\pi}{4\beta}\left(f + \beta - \frac{1}{2T_b}\right)\right]^2 \qquad (2.42)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 ผลตอบสนองอิมพัลส์ของรูปคลื่นสัญญาณที่เป็นแบบเรสต์โคไซน์

ดังนั้นสเปกตรัมของเรสต์โคไซน์ ซึ่งบางครั้งจะอ้างถึงว่ามีค่าเท่ากับฟังก์ชันโคไซน์สแควร์ ซึ่งค่าของ β สามารถเปลี่ยนค่าจาก 0 - $\frac{1}{2T_b}$ โดยเป็นการกำหนดแบนด์วิดท์ ค่าของ β ที่ถูกกำหนดให้อยู่ในช่วงนี้จะทำให้ไม่เกิดการสอดแทรกระหว่างสัญลักษณ์ อย่างไรก็ตามค่าของ β ที่มีค่าน้อยๆจะต้องมีบิทไทม์มิ่ง(bit timing)ที่มีความถูกต้องแม่นยำสูงๆ และรูปร่างของผลตอบสนองฟิลเตอร์(filter response shape)ต้องมีอิมพลีเมนท์ที่มีความถูกต้องแม่นยำสูงๆเช่นกัน

เราจะพิจารณาผลตอบสนองทางเวลา(time response)ของพัลส์เรสต์โคไซน์ ซึ่งให้มีค่าเป็นศูนย์ในแต่ละช่วงเวลาแซมปลิ้ง(T_b) ผลตอบสนองทางเวลาหาได้จากการแปลงฟูเรียร์ผกผันของสเปกตรัมความถี่

$$Y(t) = \int_{-\infty}^{\infty} Y(f) e^{j2\pi ft}\, df \tag{2.43}$$

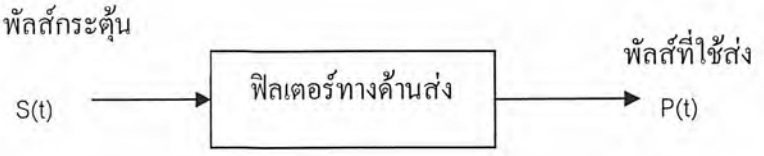
แทนสมการ $Y(f)$ จากสมการ2.41 จะได้ผลตอบสนองทางเวลาดังนี้

$$Y(t) = \left[\frac{1}{1-(4\beta t)^2}\right]\left[\frac{\sin\left(\pi \frac{t}{T_b}\right)}{\pi \frac{t}{T_b}}\right]\cos(2\pi\beta t) \tag{2.44}$$

ฟังก์ชัน $\frac{\sin(X)}{X}$ จะมีค่าเท่ากับศูนย์ที่ทุกๆค่าของ t สำหรับ $\frac{\pi t}{T_b} = N\pi$ ยกเว้นที่ $t = 0$ ดังนั้นเราจะได้ค่าเป็นศูนย์ที่ $t = N_{Tb}$ โดยที่ N เป็นเลขจำนวนจริงที่มีค่าไม่เท่ากับศูนย์ สำหรับค่า $\beta > 0$ ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทอมแรกของสมการจะเป็นการลดค่าของ $Y(t)$ ที่อยู่ใกล้กับช่วงเวลาในการแซมปลิ้งที่ไม่มีค่าเป็น 1 ในการลดค่าของ Y(t)นี้จะต้องมีค่าบิทไทม์มิ่งที่ถูกต้องแม่นยำ



รูปที่ 2.14 บล็อกไดอะแกรมของวงจร pulse generation

การวิเคราะห์สมการ (2.44) จะสัมพันธ์กับเอาท์พุทของแมทช์ฟิลเตอร์ในระบบที่เป็นสัญญาณแบนด์มูลฐาน ซึ่งจะใช้ในการทดสอบฟิลเตอร์ในระบบแบนด์มูลฐานและระบบความถี่วิทยุได้ทั้งคู่

อันดับแรก เราควรที่จะเลือกผลตอบสนองความถี่ที่ต้องการตามสมการ (2.42) ซึ่งต้องเป็นค่าจริง โดยลิเนียร์เฟสดีเลย์(linear phase delay)จะสัมพันธ์กับไทม์ดีเลย์(time delay)ที่ยอมรับได้ และลิเนียร์เฟสดีเลย์นี้จะทำได้ยากในระบบที่เป็นอนาล็อกแต่จะใช้ได้ดีในดิจิตอลฟิลเตอร์ชนิดไอไออาร์ (IIR) ส่วนนอนลิเนียร์เฟสดีเลย์ (nonlinear phase delay) จะเป็นแฟคเตอร์ที่สำคัญในการเพิ่มค่าอัตราความผิดพลาด (error rate) สำหรับการส่งข้อมูล ในอีกแง่หนึ่งดิจิตอลฟิลเตอร์ชนิดเอฟไออาร์(FIR)จะมีสัมประสิทธิ์ที่สมมาตรกันซึ่งมีลิเนียร์เฟสดีเลย์ในทางอุดมคตินี่เป็นเหตุผลหลักในการเลือกใช้ดิจิตอลฟิลเตอร์

รูปคลื่นที่ใช้ในการส่งข้อมูลจะมีสเปกตรัมความถี่เป็นแบบฟังก์ชันโคไซน์สแควร์ ตามสมการที่ (2.42) และสเปกตรัมความถี่นี้ก็จะเป็นเอาท์พุทของแมทช์ฟิลเตอร์ด้วย สามารถสังเกตได้จากสมการที่ (2.25) สำหรับรูปคลื่นที่มีสเปกตรัมเป็นค่าจริงและสำหรับสัญญาณรบกวนสีขาวที่มีสเปกตรัม $G_n(f)$ เป็นค่าคงที่ แมทช์ฟิลเตอร์จะมีผลตอบสนองความถี่เหมือนกับรูปคลื่นที่ส่งมาตามสมการต่อไปนี้

$$P(f)H(f) = T_b \qquad 0 < |f| < \left(\frac{1}{2T_b} - \beta\right)$$

$$P(f)H(f) = T_b\left[\cos\frac{\pi}{4\beta}\left(f + \beta - \frac{1}{2T_b}\right)\right]^2 \qquad \left(\frac{1}{2Tb} - \beta\right) < |f| < \left(\frac{1}{2Tb} + \beta\right)$$

$$P(f)H(f) = 0 \qquad \text{อื่นๆ} \qquad (2.45)$$

โดยที่ $P(f)$ คือ สเปกตรัมความถี่ของรูปคลื่นที่ส่ง

ผลตอบสนองทั้งหมดของ $P(f)H(f)$ สามารถหาได้หลายวิธี อย่างไรก็ตามเราต้องการเครื่องรับแมทช์ฟิลเตอร์ตามสมการ(2.32) สำหรับค่าของอัตราความผิดพลาดของบิทมีค่าต่ำที่สุด สามารถทำได้โดยให้รูปสัญญาณที่ภาครับและภาคส่งมีค่าเท่ากัน โดยให้ค่า $H(f) = P(f)$ ถ้าแทนค่าดังนี้จะได้ค่าตามสมการ(2.46)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$H(f) = P(f) = \sqrt{T_b} \qquad 0 < |f| < \left(\frac{1}{2Tb} - \beta\right)$$

$$H(f) = P(f) = \sqrt{T_b}\ \cos\frac{\pi}{4\beta}\left(f + \beta - \frac{1}{2T_b}\right) \qquad \left(\frac{1}{2Tb} - \beta\right) < |f| < \left(\frac{1}{2Tb} + \beta\right) \quad (2.46)$$

รูปสัญญาณที่ส่งมาและแมทช์ฟิลเตอร์ในเครื่องรับจะมีโคไซน์(ไม่ใช่เรสต์โคไซน์)โรลออฟ (rolloff)ได้จากผลลัพธ์สุดท้ายของเรสต์โคไซน์โรลออฟ

พิจารณาการกำเนิดของรูปคลื่นสัญญาณที่ทำการส่ง โดยการนำพัลส์สี่เหลี่ยมที่มีความยาวเป็น T_b ไปผ่านฟิลเตอร์เพื่อเปลี่ยนรูปร่างตามรูป 2.14

จากสมการ(2.45) สามารถหาผลตอบสนองความถี่ของฟิลเตอร์ที่เครื่องส่ง $F_x(f)$ ได้จาก

$$S(f)Fx(f) = P(f) \quad (2.47)$$

หรือ

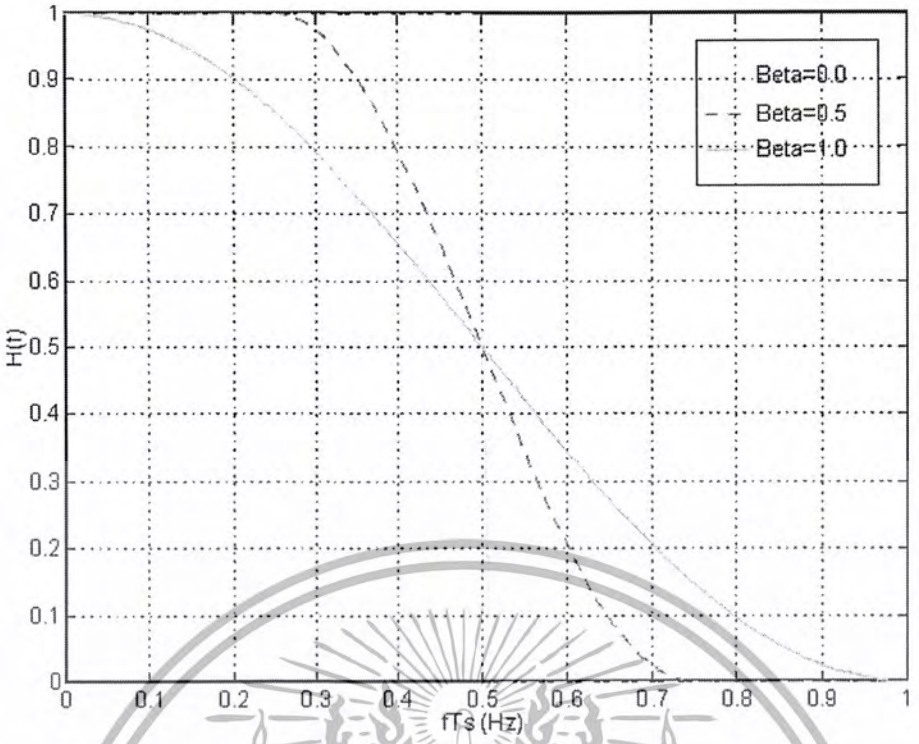
$$Fx(f) = \frac{P(f)}{S(f)} \quad (2.48)$$

แทนค่าในสมการ (2.47) ได้สมการดังต่อไปนี้

$$Fx(f) = \frac{1}{\sqrt{T_b}\left[\dfrac{\sin(\pi T_b)}{\pi T_b f}\right]} \qquad 0 < f < \left(\frac{1}{2Tb} - \beta\right) \quad (2.49)$$

$$Fx(f) = \frac{\cos\left[\dfrac{\pi}{4\beta}\left(f + \beta - \dfrac{1}{2T_b}\right)\right]}{\sqrt{T_b}\left[\dfrac{\sin(\pi T_b f)}{\pi T_b f}\right]} \qquad \left(\frac{1}{2Tb} - \beta\right) < f \leq \left(\frac{1}{2Tb} + \beta\right) \quad (2.50)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 ผลตอบสนองทางความถี่ของเรสต์โคไซน์

2.7 พัลส์รูทเรสต์โคไซน์(Root raised cosine pulse)

จะมีผลตอบสนองทางความถี่ของรูทเรสต์โคไซน์แสดงดังรูป 2.17 และหาค่าได้จากสมการ

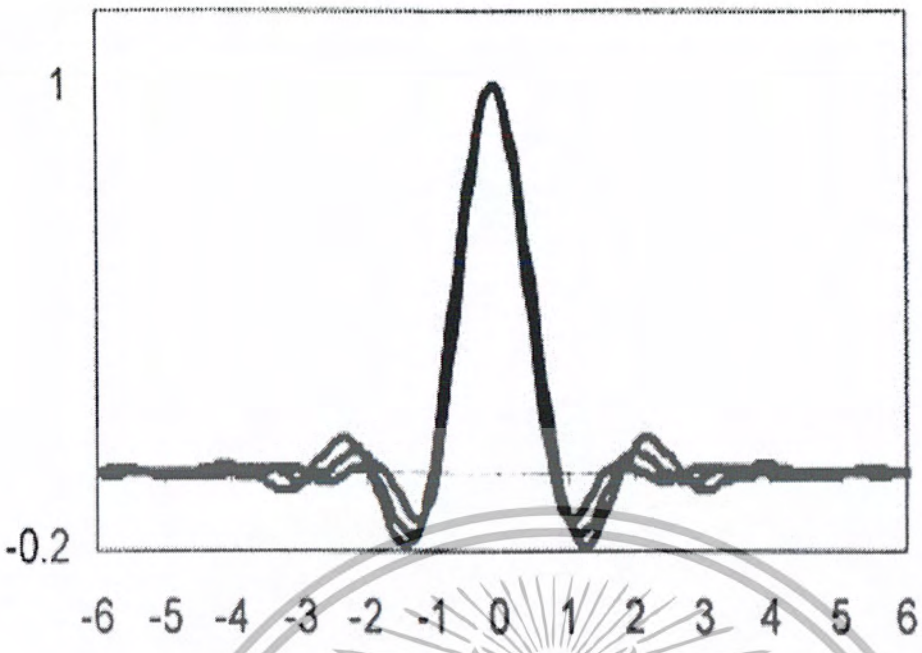
$$H(f) = T_b \qquad 0 \le |f| \le \frac{(1-\beta)}{2T_b}$$

$$H(f) = T_b \sqrt{\frac{1}{2}\left\{1+\cos\left[\pi\frac{T_b}{\beta}\left(|f|-\frac{1}{2T_b}\right)\right]\right\}} \qquad \frac{(1-\beta)}{2} \le |f| \le \frac{(1+\beta)}{2T_b}$$

$$H(f) = 0 \qquad |f| > \frac{(1+\beta)}{2T_b} \tag{2.51}$$

เมื่อลองเปรียบเทียบสมการผลตอบสนองทางความถี่ของพัลส์รูทเรสต์โคไซน์กับพัลส์เรสต์โค-ไซน์จะเห็นว่าผลตอบสนองทางความถี่ของพัลส์รูทเรสต์โคไซน์ มีค่าเท่ากับค่าสแควร์รูท (Square root) ผลตอบสนองทางความถี่ของพัลส์เรสต์โคไซน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



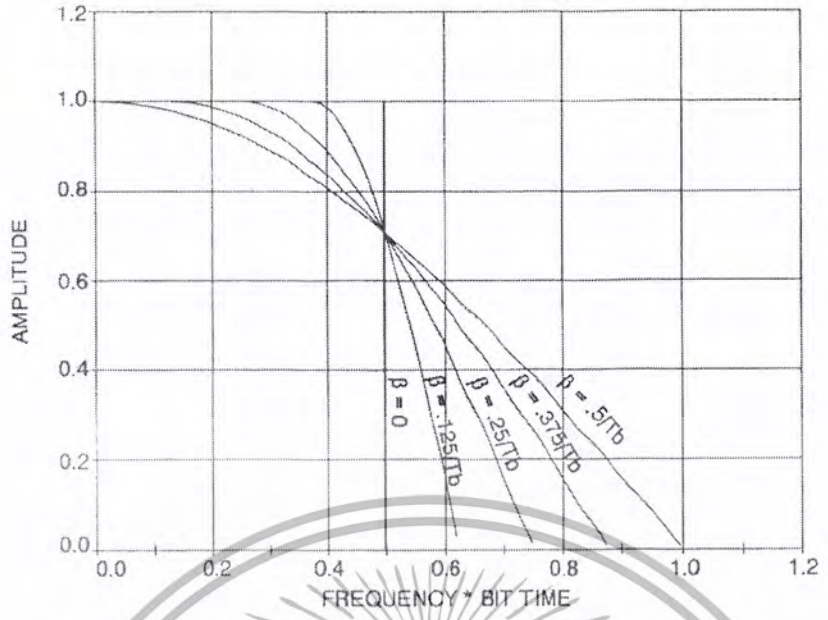
รูปที่ 2.16 ผลตอบสนองอิมพัลส์ของรูปคลื่นสัญญาณที่เป็นแบบ รูทเรสต์โคไซน์

$$h(t) = 1-\beta+4\frac{\beta}{\pi} \qquad t=0$$

$$h(t) = \frac{\beta}{\sqrt{2}}\left[\left(1+\frac{2}{\pi}\right)\sin\left(\frac{\pi}{4\beta}\right)+\left(1-\frac{2}{\pi}\right)\cos\left(\frac{\pi}{4\beta}\right)\right] \qquad t=\pm\frac{T}{4\beta}$$

$$h(t) = \frac{\sin\left[\pi(1-\beta)\frac{t}{T_b}\right]+4\beta\frac{t}{T}\cos\left[\pi(1+\beta)\frac{t}{T_b}\right]}{\pi\frac{t}{T_b}\left[1-\left(4\beta\frac{t}{T_b}\right)^2\right]} \qquad \text{สำหรับ } t \text{ อื่นๆ} \qquad (2.52)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ผลตอบสนองทางความถี่ของรูทเรสต์โคไซน์

เมื่อเปรียบเทียบผลตอบสนองอิมพัลส์ของพัลส์รูทเรสต์โคไซน์กับพัลส์เรสต์โคไซน์จะไม่เห็นถึงความแตกต่างมากนัก

2.8ทฤษฎีการชักค่าตัวอย่าง(Sampling Theorem)

สัญญาณที่เกิดอย่างต่อเนื่องตลอดเวลาที่เรียกกันว่า สัญญาณอนาล็อก(analog signal)นั้น ถ้าค่าความหนาแน่นสเปกตรัมของมันเกิดอยู่ในแบนด์วิดท์ที่จำกัดแล้ว เราจะพบว่าในโดเมนเวลานั้น สัญญาณอนาล็อกที่มีคุณสมบัติดังกล่าวจะมีข้อมูลที่มีความซ้ำซ้อนเกินความจำเป็นรวมอยู่ด้วยมากในระหว่างเวลาส่งสัญญาณนั้น ดังนั้นถ้าเราเลือกชักค่าตัวอย่าง(sampling)ของสัญญาณนั้นที่ขณะเวลาต่างๆที่อยู่ห่างกันอย่างเหมาะสมแล้ว เพียงจากค่าตัวอย่างของสัญญาณที่ชักออกมาได้นี้ก็สามารถเก็บข้อมูลของสัญญาณทั้งหมดได้อย่างสมบูรณ์ ซึ่งกล่าวได้ว่าการเลือกชักค่าตัวอย่างสัญญาณที่เวลาที่เหมาะสมจะทำให้เราสามารถลดความซ้ำซ้อนหรือความฟุ่มเฟือยของการใช้ค่าสัญญาณลงได้ เพราะฉะนั้นเราจึงสามารถนำเอาช่วงเวลาที่ไม่ตรงกับขณะเวลาที่มีการชักค่าสัญญาณไปใช้ในการทำประโยชน์อย่างอื่นได้และเราสามารถใช้เพียงข้อมูลเท่าที่เราชักค่ามาได้ซึ่งขึ้นอยู่กับค่าตัวอย่างของสัญญาณอนาล็อกที่ชั่วขณะเวลาที่ทำการชักค่าสัญญาณนั้นไปจัดการประมวลผลตามต้องการได้สะดวก เนื่องจากค่าตัวอย่างที่ชักออกมานั้นเกิดที่เวลาห่างกันเป็นห้วงๆไม่ติดต่อกัน เราจึงเรียกค่าสัญญาณที่เกิดจากการชักค่าตัวอย่างสัญญาณนี้ว่าสัญญาณดิสครีตทางเวลา หรือ นิยมเรียกสั้นๆว่า สัญญาณดิสครีต (discrete signal) ทฤษฎีสำคัญที่บอกความสัมพันธ์ในการกำหนดคาบเวลาสำหรับการชักค่าตัวอย่างสัญญาณอนาล็อกเพื่อสร้างสัญญาณดิสครีตที่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลสำหรับสัญญาณอนาล็อกเดิมอยู่ครบถ้วนคือ ทฤษฎีการชักตัวอย่าง(sampling theorem)ซึ่งมีใจความต่อไปนี้

ถ้าสัญญาณใดๆที่มีค่าฟังก์ชันมีความหนาแน่นสเปกตรัมอยู่ในช่วงความถี่ที่มีแบนด์วิดท์ไม่เกิน B เฮิรตซ์แล้ว การชักค่าตัวอย่างของสัญญาณนั้นที่ขณะเวลาที่ห่างกันอย่างสม่ำเสมอเป็นคาบไม่เกิน $\frac{1}{2B}$ วินาที จะทำให้สามารถกำหนดค่าสัญญาณอนาล็อกเดิมกลับคืนมาได้จากค่าตัวอย่างสัญญาณเหล่านี้ เมื่อลองพิจารณาเกี่ยวกับทฤษฎีการชักตัวอย่างนี้ให้ละเอียดขึ้น โดยการสมมติว่าสัญญาณอนาล็อก $f(t)$ เป็นสัญญาณที่มีความหนาแน่นเชิงสเปกตรัมอยู่ไม่เกินความถี่ B เฮิรตซ์ ดังแสดงในรูป 2.18 ถ้าเราทำการชักค่าตัวอย่างของ $f(t)$ ที่ทุกขณะเวลาที่ห่างกัน T วินาที โดยกำหนดให้สัญญาณที่เกิดจากการชักค่าตัวอย่างของ $f(t)$ นี้คือ $f_s(t)$ แล้วจะทำการอธิบายในเชิงคณิตศาสตร์ได้ว่า

$$f_s(t) = f(t)\,\delta_T(t) \tag{2.53}$$

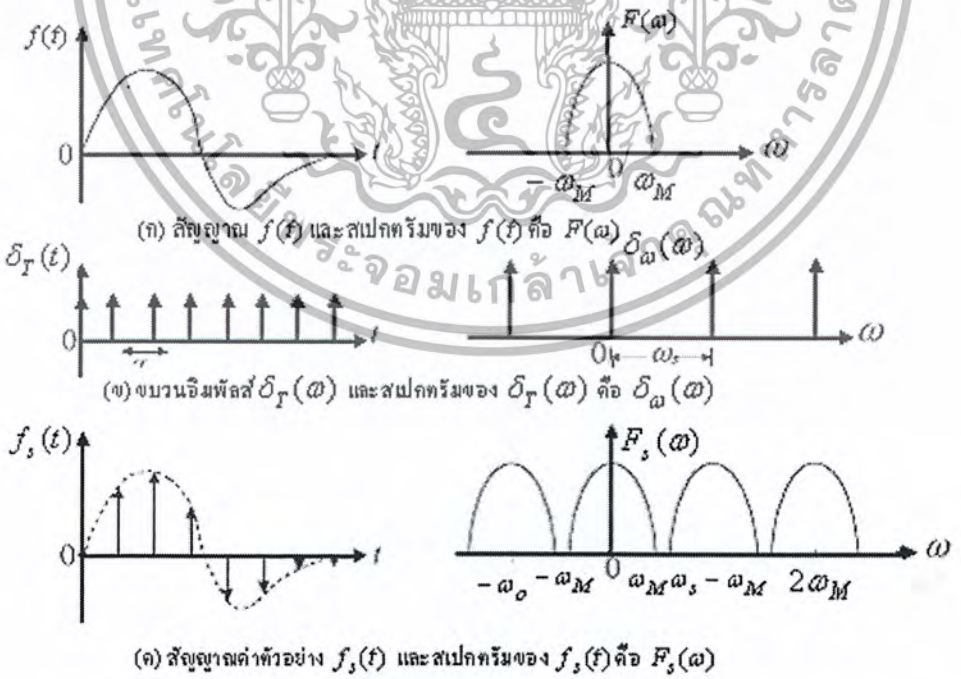
โดยในที่นี้ $\delta_T(t)$ คือ ขบวนอิมพัลส์(impulse train) ที่มีการนิยามว่า

$$\delta_T(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT) \tag{2.54}$$

โดยอาศัยคุณสมบัติของฟังก์ชันอิมพัลส์(2.53)ทำให้สามารถเขียน(2.54)ได้เป็น

$$f_s(t) = \sum_{n=-\infty}^{\infty} f(nT)\delta(t-nT) \tag{2.55}$$

สมการ(2.55)แสดงให้เห็นแน่นอนว่า $f_s(t)$ คือ สัญญาณ $f(t)$ ที่มีค่าเพียงแต่ที่ขณะเวลา $t = nT$, (n = 1,2,...)เท่านั้น



รูปที่ 2.18 สเปกตรัมของสัญญาณที่ถูกชักค่าตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่การแปลงฟูเรียร์เบื้องต้นของขบวนอิมพัลส์มีรูปดังต่อไปนี้ คือ

$$\delta_{\mathrm{T}}(t) \quad \leftrightarrow \delta\omega_s(\omega)\omega_s \tag{2.56}$$

โดยที่ $\omega_S = \frac{2\pi}{\mathrm{T}}$

โดยอาศัยคุณสมบัติตามสมการ(2.55)ถ้ากำหนดให้ $f_S(\mathrm{t}) \leftrightarrow F_s(\omega)$ และ $f(t) \leftrightarrow F(\omega)$ จะได้

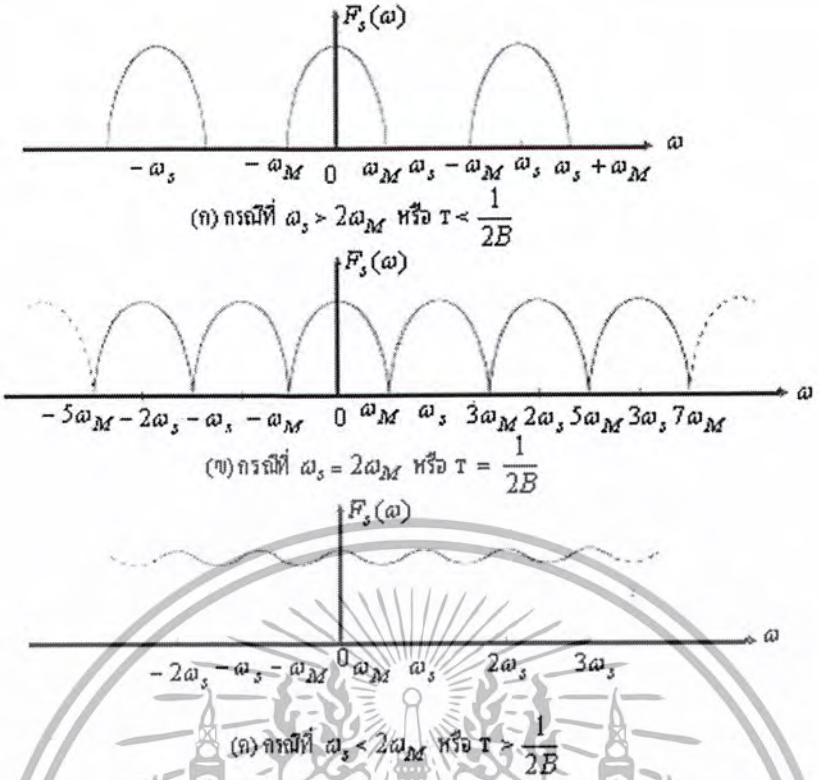
$$F_s(\omega) = \frac{1}{2\pi} F(\omega) \otimes \delta\omega_s(\omega)\omega_s$$

$$= \frac{\omega_s}{2\pi} \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} F(z)\delta(\omega - n\omega_s - z)dz$$

$$= \frac{1}{\mathrm{T}} \sum_{n=-\infty}^{\infty} F(\omega - n\omega_s) \tag{2.57}$$

จาก(2.57)จะทำให้รู้ว่า $f_s(t)$ หรือสัญญาณดิสครีตที่เกิดจากการชักตัวอย่างของ $f(t)$ จะมีฟังก์ชันความหนาแน่นเชิงสเปกตรัม ที่มีรูปร่างเหมือนฟังก์ชันความหนาแน่นเชิงสเปกตรัมของสัญญาณ $f(t)$ แต่จะเกิดอยู่ซ้ำกันทุกคาบความถี่ ω_s ดังแสดงในรูป (2.18) เมื่อ $f(t)$ มีองค์ประกอบของความถี่สูงสุด คือ B เฮิรตซ์ ค่าความถี่เชิงมุมที่ตรงกันกับค่าความถี่สูงสุดนี้จะมีค่า $\omega_M = 2\pi$B เรเดียนต่อวินาที จะพบว่าถ้า $\omega_s \rangle\ 2\omega_M$ หรือ $\mathrm{T} \langle\ \frac{1}{2\mathrm{B}}$ แล้ว $F_S(\omega$ จะมีลักษณะเหมือนกับการเอา $F(\omega)$ มาเรียงกันห่างๆบนแกนความถี่ ω ดังแสดงในรูป(2.19)ค หรือรูป ก แต่เมื่อ $\omega_S = 2\omega_M$ หรือ $\mathrm{T} = \frac{1}{2\mathrm{B}}$ จะพบว่า $F_s(\omega)$ มีลักษณะเหมือนกับการเอา $F(\omega)$ มาเรียงชิดติดๆกันพอดีบนแกน ω ดังแสดงในรูป ข และถ้า $\omega_s \langle\ 2\omega_M$ หรือ $\mathrm{T} \rangle\ \frac{1}{2\mathrm{B}}$ แล้วจะทำให้ $F_s(\omega)$ มีลักษณะที่เกิดจากผลรวมของ $F(\omega)$ ที่มาเรียงเหลื่อมทับกันดังแสดงในรูป ค การเกิดเหลื่อมทับกันของ $F(\omega)$ ตามรูป ค ทำให้เกิดผลเทียบเท่ากับการพับกลับ(fold over)ขององค์ประกอบของสัญญาณที่มีความถี่สูงเกิน $\frac{\omega_S}{2}$ ซึ่งจะกลับมาทับกับองค์ประกอบของสัญญาณ $F_s(\omega)$ ที่มีความถี่ต่ำและรวมตัวกัน จึงทำให้มีสเปกตรัมผิดเพี้ยนไปจากเดิม ปรากฏการณ์ที่เกิดการเหลื่อมทับ หรือ การพับกลับของส่วนประกอบของสเปกตรัมดังกล่าวนี้มีคำศัพท์เรียกเฉพาะว่า การเกิดการซ้อนทับของสเปกตรัม(aliasing)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 สเปกตรัมของสัญญาณที่เกี่ยวเนื่องกับรูปที่ 2.18 เมื่อใช้ช่วงเวลาในการสุ่มค่าสัญญาณต่างๆ

จากรูป 2.19 จะเห็นว่าหากคาบเวลาของการชักค่าตัวอย่างสัญญาณ T น้อยกว่า $\frac{1}{2B}$ วินาทีแล้ว เราจะสามารถใช้วงจรกรองความถี่มาแยกเอาส่วนของสเปกตรัมเฉพาะส่วนที่มีความถี่ต่ำของ $F_s(\omega)$ ซึ่งเหมือนกับ $F(\omega)$ ออกมาได้โดยสะดวก เมื่อทำเช่นนี้ได้ก็เท่ากับว่าเราสามารถแยกหรือตรวจจับ(detect) เอา $f(t)$ จาก $f_s(t)$ ได้โดยใช้วงจรกรองความถี่เป็นอุปกรณ์ที่ใช้ช่วยแยกสัญญาณได้ แต่อย่างไรก็ตาม ในกรณีที่ T มากกว่า $\frac{1}{2B}$ แล้ว จะทำให้เกิดมีการซ้อนทับกันของส่วนประกอบความถี่ดังรูป ค ทำให้เราไม่สามารถใช้วงจรกรองความถี่แยกเอา $F(\omega)$ จาก $F_s(\omega)$ ได้เลย การใช้รูปแบบทางคณิตศาสตร์มาช่วยวิเคราะห์ผลดังกล่าว ทำให้เราสามารถมองลึกลงไปได้ว่า ทำไมทฤษฎีการชักตัวอย่าง จึงกล่าวว่าคาบเวลาของการชักค่าตัวอย่างสัญญาณจึงต้องมีค่าไม่เกิน $\frac{1}{2B}$ วินาที และจะเห็นว่าในกรณีที่ $T = \frac{1}{2B}$ พอดี ฟังก์ชันความหนาแน่นเชิงสเปกตรัม $F_s(\omega)$ จะเกิดจาก $F(\omega)$ ที่มาเรียงชิดกันดังแสดงในรูป ข ทำให้เรารู้ว่าในกรณีนี้มีแต่วงจรกรองความถี่ในอุดมคติเท่านั้นที่จะสามารถแยกเอาแต่สเปกตรัมในช่วงความถี่ที่ต้องการ คือ $F(\omega)$ ออกมาได้ ปกติในทางปฏิบัติวงจรกรองความถี่หาได้มีคุณสมบัติที่จะแยกส่วนของความถี่ได้อย่างเฉียบพลันไม่ กล่าวคือ จะต้องมีการโรลออฟ(roll off)หรือการค่อยๆลดลงของค่าฟังก์ชันถ่ายโอนอยู่บริเวณใกล้จุดตัด(cutoff)ความถี่ ดังนั้นในทางปฏิบัติจึงมักจะเลือกใช้คาบเวลาการชักตัวอย่าง T ให้น้อยกว่า $\frac{1}{2B}$ เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า T = $\frac{1}{2B}$ นี้มีความสำคัญในทางทฤษฎี เพราะมันเป็นค่าวิกฤตที่จะบอกว่าการชักค่าตัวอย่างสัญญาณนั้นจะมีผลกระทบต่อการแยกตรวจจับสัญญาณในภายหลังหรือไม่ ค่าคาบเวลา T = $\frac{1}{2B}$ นี้มีชื่อเฉพาะเรียกว่า ช่วงเวลาการชักตัวอย่างไนควิสต์ หรือบางครั้งก็นิยมเรียกสั้นๆว่า ช่วงเวลาไนควิตซ์ (Nyquist interval)และค่าความถี่ f = 2B นี้มีชื่อเรียกว่า ความถี่การชักตัวอย่างไนควิสต์(Nyquist sampling frequency) หรือบางครั้งนิยมเรียกสั้นๆว่า ความถี่ไนควิสต์ หรือ อัตราไนควิสต์(Nyquist rate) ของการชักค่าตัวอย่างสัญญาณ

2.9 ทฤษฎีการตัดสินใจ(Decision Theorem)

หน้าที่หลักของวงจรดีเทกเตอร์ก็คือ การนำชุดตัวแปรแรนดอม $r_1, r_2, ..., r_N$ ที่ได้จากด้านออกของวงจรดีมอดูเลเตอร์มาประมวลเพื่อใช้ในการตัดสินใจว่าสัญญาณที่ได้รับนั้นน่าจะเป็นสัญญาณดิจิตอลชุดใดที่ต้นทางส่งออก กรรมวิธีที่ใช้ในการตัดสินใจมีพื้นฐานการคำนวณโดยอาศัยทฤษฎีความน่าจะเป็นเป็นหลัก

2.9.1กฎของเบย์(Bayes'Rule)

ในการทดลองบางอย่างจะให้ผลลัพธ์ที่ไม่มีความแน่นอน คือ มีความเป็นไปได้หลายอย่าง เช่น ในการทอดลูกเต๋าครั้งหนึ่งผลลัพธ์ที่ได้ออกมาอาจจะเป็น ได้ถึง 6 ค่า คือ ค่าจาก 1 ถึง 6 ถ้ากำหนดให้ A แทนค่าผลลัพธ์ที่อาจจะเกิดขึ้น ในการทดลองเช่นนั้นค่าหนึ่ง เช่น ในกรณีของการทอดลูกเต๋า A นี้อาจเป็นค่าเลขบนหน้าลูกเต๋าหน้าใดหน้าหนึ่งที่ปรากฏขึ้นมาภายหลังจากการทอดลูกเต๋าเป็นต้น ในการทอดลูกเต๋าทั้งหมด N ครั้ง สมมติว่าได้ผลออกมาเป็น A จำนวน N_A ครั้ง อัตราส่วน $\frac{N_A}{N}$ นี้ ถูกนิยามว่า คือ ความถี่สัมพัทธ์(relative frequency)ของเหตุการณ์นี้ ซึ่งค่าอัตราส่วนนี้คงจะใช้บอกอะไรไม่ได้มาก ถ้าหากว่าจำนวนการทดลอง N นั้นมีค่าจำนวนน้อย แต่ถ้าหากว่าจำนวนการทดลอง N นั้นมีค่าจำนวนน้อย แต่ถ้าหากจำนวนการทดลอง N นั้นมาก ค่าความถี่สัมพัทธ์ของเหตุการณ์นั้นจะมีค่าลิมิต(limit)เข้าสู่ค่าๆหนึ่ง ค่าลิมิตนี้ถูกนิยามว่า คือ ค่าความน่าจะเป็น(probability)หรือโอกาสการเกิดของผลลัพธ์นั้น โอกาสการเกิดของผลลัพธ์ A เขียนเป็นนิยามในรูปแบบเชิงคณิตศาสตร์ได้ คือ

$$P(A) = \lim_{N\to\infty} \frac{N_A}{N} \qquad (2.58)$$

ควรสังเกตจากสมการ(2.58) ว่าค่าของ P(A) นั้นมีได้สูงสุด คือ เท่ากับ 0 $\leq$ P(A) $\leq$ 1 การทดลองนี้ให้ผลลัพธ์ (result) คือ A ออกมานั้น ในบางครั้งในวิชาสถิติอาจจะกล่าวว่าในการทดลองนั้นมีเหตุการณ์(event) A เกิดขึ้น ซึ่งก็ควรทำความเข้าใจว่า เหตุการณ์ต่างๆก็คือ ผลลัพธ์ต่างๆนั่นเอง

เหตุการณ์ 2 เหตุการณ์นั้น ถูกนิยามว่า ไม่มีส่วนร่วม (disjoint) หรือไม่เกิดร่วมกันและกัน (mutually exclusive)ถ้าเหตุการณ์ทั้งสองนั้นไม่มีโอกาสเกิดขึ้นพร้อมๆกัน กล่าวคือ ในกรณีที่ถ้าเหตุการณ์ A เกิดขึ้นแล้วเหตุการณ์ B ก็จะไม่เกิดขึ้นหรือในทางกลับกัน ก็คือ ถ้าเหตุการณ์ B เกิดขึ้นแล้วเหตุการณ์ A ก็จะไม่เกิดขึ้น เราจะกล่าวว่าเหตุการณ์ A และB นั้นไม่มีส่วนร่วมกัน อธิบายได้ด้วยสัญลักษณ์ทางคณิตศาสตร์ดังนี้คือ

$$P(A \cap B) = P(AB) = 0 \qquad (2.59)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีของเหตุการณ์ที่ไม่มีส่วนร่วมกันเช่นนี้ เราจะพบว่าโอกาสการเกิดของเหตุการณ์ A หรือ เหตุการณ์ B ซึ่งใช้สัญลักษณ์ $P(\mathrm{A} \cup \mathrm{B})$ มีค่าดังนี้คือ

$$P(\mathrm{A} \cup \mathrm{B}) = \lim_{N \to \infty} \frac{N_A + N_B}{N}$$

$$= P(\mathrm{A}) + P(\mathrm{B}) \tag{2.60}$$

โดยในที่นี้ N_A และ N_B คือ จำนวนผลลัพธ์จากการทดลองที่เป็นเหตุการณ์ A และ เหตุการณ์ B ตามลำดับ

เหตุการณ์ 2 เหตุการณ์ คือ A และ B ที่อาจเกิดขึ้นได้พร้อมกันในบางครั้งนั้น ถ้ามีบางส่วนของเหตุการณ์ A เกิดขึ้นพร้อมกับเหตุการณ์ B โอกาสของการเกิดเหตุการณ์ A และ B ขึ้นพร้อมกันนี้เราเรียกว่า โอกาสเกิดร่วม(joint probability) ของ A และ B ในการทดลอง N ครั้ง ถ้าให้โอกาสในการเกิด A และ B ร่วมกันคือ N_{AB} แล้ว โดยใช้หลักการของความถี่สัมพัทธ์จะได้

$$P(\mathrm{AB}) = \lim_{N \to \infty} \frac{N_{\mathrm{AB}}}{N} \tag{2.61}$$

โดยทั่วไปแล้ว $N_{AB} \leq N_A$ และ $N_{AB} \leq N_B$ ทั้งนี้เนื่องจากเหตุการณ์ A และ เหตุการณ์ B ไม่จำเป็นที่จะต้องเกิดพร้อมกันเสมอ

ในการทดลองบางอย่างการเกิดขึ้นของเหตุการณ์ B อาจเกี่ยวพันกับเหตุการณ์ A โอกาสที่เหตุการณ์ B เกิดขึ้น เมื่อเหตุการณ์ A ได้เกิดแล้ว มีชื่อเรียกว่า โอกาสใต้เงื่อนไข (condition probability) ของเหตุการณ์ B เมื่อกำหนดเหตุการณ์ A มีสัญลักษณ์เป็น $\mathrm{P}(\mathrm{B}/\mathrm{A})$ ซึ่งถ้านิยามตามหลักการของความถี่สัมพัทธ์จะได้

$$\mathrm{P}(\mathrm{B}/\mathrm{A}) = \lim_{N \to \infty} \frac{N_{\mathrm{AB}}}{N_{\mathrm{A}}} = \lim_{N \to \infty} \frac{\frac{\mathrm{N_{AB}}}{\mathrm{N}}}{\frac{\mathrm{N_A}}{\mathrm{N}}}$$

$$= \frac{\mathrm{P}(\mathrm{AB})}{\mathrm{P}(\mathrm{A})} \quad , \text{เมื่อ } \mathrm{P}(\mathrm{A}) \neq 0 \tag{2.62}$$

ในทำนองเดียวกันจะได้ว่า

$$\mathrm{P}(\mathrm{A}/\mathrm{B}) = \frac{\mathrm{P}(\mathrm{AB})}{\mathrm{P}(\mathrm{B})} \quad , \text{เมื่อ } \mathrm{P}(\mathrm{B}) \neq 0 \tag{2.63}$$

จาก (2.62) และ(2.63) จะได้

$$P(\mathrm{AB}) = \mathrm{P}(\mathrm{B}/\mathrm{A})\,\mathrm{P}(\mathrm{B}) = \mathrm{P}(\mathrm{A}/\mathrm{B})\,\mathrm{P}(\mathrm{A}) \tag{2.64}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือ

$$P(B/A) = \frac{P(B)P(A/B)}{P(A)} \tag{2.65}$$

ความสัมพันธ์ตามสมการ (2.64) รู้จักกันทั่วไปว่าเป็น กฎของเบส์(Bayes'rule)

ถ้าโอกาสใต้เงื่อนไขของเหตุการณ์ B เมื่อกำหนดเหตุการณ์ A มีค่าเท่ากับโอกาสการเกิดของเหตุการณ์ B โดยปกติ กล่าวคือ

$$P(B/A) = P(B) \tag{2.66}$$

จะกล่าวได้ว่าเหตุการณ์ B เป็นอิสระทางสถิติกับเหตุการณ์ A จงสังเกตว่าการเป็นอิสระของเหตุการณ์ทางสถิตินั้นไม่ได้หมายความว่าเหตุการณ์ A เกิดแล้ว เหตุการณ์ B จะไม่เกิดขึ้น แต่จะบอกว่าเมื่อเหตุการณ์ A เกิดขึ้นแล้ว โอกาสที่เหตุการณ์นั้นจะเป็นเหตุการณ์ B ด้วย จะมีค่าเท่ากับโอกาสของเหตุการณ์ B ที่จะเกิดขึ้นโดยลำพังเมื่อเทียบกับเหตุการณ์ในการทดลองทั้งหมด หรือถ้าใช้หลักของความถี่สัมพัทธ์ก็จะได้ว่าในกรณีที่ B เป็นอิสระทางสถิติกับเหตุการณ์ A จะได้

$$\lim_{x\to\infty} \frac{N_{AB}}{N_A} = \lim_{x\to\infty} \frac{N_B}{N} \tag{2.67}$$

จาก(2.62)และ(2.66) ทำให้เราได้เงื่อนไขของเหตุการณ์ A และเหตุการณ์ B ที่เป็นอิสระกันทางสถิติ คือ

$$P(AB) = P(A)\,P(B) \tag{2.68}$$

การสังเกตว่าคุณสมบัติความเป็นอิสระกันทางสถิติของเหตุการณ์ 2 เหตุการณ์นั้น แตกต่างจากคุณสมบัติความไม่มีส่วนร่วมหรือไม่เกิดร่วมกันของเหตุการณ์ 2 เหตุการณ์ ซึ่งแสดงว่าเหตุการณ์ A เกิดขึ้นแล้วเหตุการณ์ B จะไม่เกิดขึ้นอย่างแน่นอน คือ $P(AB) = 0$

เราจะอธิบายความหมายของความน่าจะเป็นต่อไปนี้และการใช้งานของมัน

$P(s_i/r)$ = P (ต้นทางส่งสัญญาณ s_i / r) โดย i = 1 , 2 ,....M

พจน์ $P(s_i/r)$ คือ ค่าความน่าจะเป็นที่ต้นทางส่งสัญญาณ s_i เมื่อเราได้รับเวกเตอร์

$$r = \begin{bmatrix} r_1 \\ r_2 \\ \cdot \\ \cdot \\ r_N \end{bmatrix} \tag{2.69}$$

เมื่อเราคำนวณค่า $P(s_i/r)$ สำหรับสัญญาณที่แสดงในรูปของเวคเตอร์ s_i แต่ละตัวจนครบทั้งหมด M รูปแบบ วงจรดีเทกเตอร์ก็จะตัดสินใจเลือกสัญญาณที่ให้ค่าความน่าจะเป็นดังกล่าวสูงสุด

ในลำดับถัดมาเราจะแสดงค่าความน่าจะเป็น $P(s_i/r)$ ในรูปแบบที่ต่างไปโดยอาศัยกฎเกณฑ์ของเบส์ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$P(s_i / r) = \frac{P(r/s_i)}{P(r)} \tag{2.70}$$

$P(r/s_i)$ คือ ค่าความน่าจะเป็นที่ได้รับเวกเตอร์ r เมื่อต้นทางได้ส่งสัญญาณ s_i ออก ค่านี้จึงเป็นค่าบอกถึงคุณสมบัติเชิงสถิติของแหล่งกำเนิดสัญญาณดิจิตอลล้วนๆ ไม่เกี่ยวข้องหรือสัมพันธ์กับช่องสัญญาณเลยมีชื่อเรียกเฉพาะว่า ความน่าจะเป็นไพออริ (priori probability) ส่วนพจน์ท้ายสุด $P(r)$ สามารถเขียนใหม่ในรูปต่อไปนี้โดยอาศัยความน่าจะเป็นแบบมาจินอล(marginal probability)

$$P(s_i / r) = \sum_{i=1}^{M} P(r/s_i)P(s_i) \tag{2.71}$$

สังเกตว่า $P(r)$ นั้นเป็นค่าที่ได้จากการพิจารณาความน่าจะเป็นของสัญญาณ s_i ทุกรูปแบบ ค่าที่คำนวณได้จึงไม่ขึ้นกับว่าทางต้นทางได้ส่งสัญญาณ s_i รูปแบบใดออก

เมื่อมองภาพรวมของความน่าจะเป็น $P(s_i / r)$ ตามสมการ(2.70)จะเห็นว่าค่านี้ขึ้นอยู่กับความน่าจะเป็น 2 ส่วน คือ $P(s_i)$ และ $P(r/s_i)$ ส่วนแรกนั้นสามารถคำนวณได้ง่ายถ้าภาคส่งให้กำเนิดสัญญาณแต่ละแบบด้วยความน่าจะเป็นเท่าๆกัน นั่นคือ $P(s_i) = 1/\mathrm{M}$ โดย $i = 1, 2, ..., \mathrm{M}$ เมื่อ $P(s_i)$ เป็นค่าคงที่ตายตัวเท่ากันหมด การหาค่าสูงสุดของ $P(s_i / r)$ จึงขึ้นอยู่กับ $P(r/s_i)$ เท่านั้น ฉะนั้นเกณฑ์ในการตัดสินใจข้อมูลบิทดิจิตอลสามารถเปลี่ยนจากการหาค่าสูงสุดของ $P(s_i / r)$ ไปเป็นการหาค่าสูงสุดของ $P(r/s_i)$ แทนได้

ค่าของ $P(r/s_i)$ ได้มีการคำนวณไว้ก่อนหน้านี้แล้วในรูปของฟังก์ชันความน่าจะเป็น สำหรับช่องสัญญาณรบกวนเกาส์สีขาวแบบบวก ซึ่งมีความสัมพันธ์ดังนี้

$$P(r/s_i) = \frac{1}{(\pi N_o)^{N/2}} \exp\left[-\sum_{j=1}^{N} \frac{(r_j - s_{ij})^2}{N_o} \right] \tag{2.72}$$

โดยทั่วไปการคำนวณจะง่ายขึ้นถ้าเราเลือกใช้ค่า $\log_e$ ของ $P(r/s_i)$ แทนค่า $P(r/s_i)$ ตรงๆ และผลที่ได้คือ

$$\ln P(r/s_i) = -\frac{N}{2}\ln(\pi N_o) - \frac{1}{N_o}\sum_{j=1}^{N}(r_j - s_{ij})^2 \tag{2.73}$$

สังเกตว่าการใส่ค่า ln เข้าไปไม่ส่งผลต่อกระบวนการตัดสินใจบิทแต่อย่างใด เพราะฟังก์ชัน ln จัดเป็นฟังก์ชันประเภทโมโนโทนิค(monotonic) จากความสัมพันธ์ที่ได้นี้จะเห็นว่าสมการจะให้ค่าสูงสุดได้เมื่อพจน์ที่ 2 ทางขวาของสมการมีค่าต่ำสุด ซึ่งสังเกตต่อจะเห็นว่าพจน์ดังกล่าวเทียบได้กับการหาค่าระยะยูซิดีน(Euclidean)ระหว่างเวคเตอร์ r กับเวคเตอร์ s_i นั่นเอง

$$D(r, s_i) = \sum_{j=1}^{N}(r_j - s_{ij})^2 \tag{2.74}$$

การวิเคราะห์ที่ผ่านมาชี้ให้เห็นว่าขั้นตอนการตัดสินใจบิทที่ซับซ้อนแท้จริงแล้วสามารถกระทำได้ด้วยกระบวนการที่ง่าย คือ เป็นเพียงการหาระยะยูซิดีนระหว่างเวคเตอร์ r กับเวคเตอร์ s_i ที่สั้นที่สุดเท่านั้น

หากเรากระจายค่า $D(r, s_i)$ ตามสมการ(2.73)ต่อจะได้ผลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$D(r,s_i) = \sum_{j=1}^{N} r_j^2 - 2\sum_{j=1}^{N} r_j s_{ij} + \sum_{j=1}^{N} s_{ij}^2$$

$$= \|r\|^2 - 2r * s_i + \|s_i\|^2 \qquad (2.75)$$

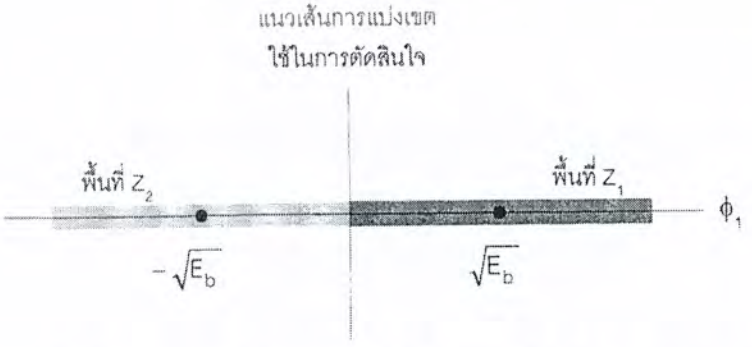
โดย $i = 1, 2, \ldots, M$ เนื่องจากพจน์ $\|r\|^2$ เป็นค่าคงที่ ฉะนั้นจึงไม่ส่งผลต่อการตัดสินใจสำหรับพจน์ถัดมา $r * s_i$ คือ การโปรเจกต์เวกเตอร์สัญญาณที่รับได้ลงบนเวกเตอร์ของสัญญาณที่เป็นไปได้แต่ละตัว การมอดูเลทสัญญาณโดยทั่วไปมักจะกำหนดให้พลังงานของสัญญาณแต่ละตัวมีค่าเท่ากัน เช่น การมอดูเลทแบบเฟสชิพท์คีย์อิงค์(Phase shift keying) เป็นต้น อย่างไรก็ดีการมอดูเลทบางรูปแบบสัญญาณที่ใช้จะมีพลังงานไม่เท่ากัน

จากที่ได้อธิบายมาทั้งหมดจะขอนิยามฟังก์ชันต้นทุนตัวใหม่ $C(r,s_i)$ ที่มีเครื่องหมายสลับกับ $D(r,s_i)$ ดังนี้คือ

$$C(r,s_i) = 2\int_0^T r(t)s_i(t)dt - E_i \qquad (2.76)$$

โดย $i = 1, 2, \ldots, M$ สังเกตว่าฟังก์ชันต้นทุนใหม่นี้ได้ตัดพจน์ $\|r\|^2$ ออกเพราะเป็นค่าคงที่และมิได้ส่งผลกระทบต่อการตัดสินใจ ส่วนการโปรเจกต์เวกเตอร์ $r * s_i$ นั้นได้เขียนแสดงใหม่ในรูปของการอินทริเกรทแทน

ในระบบสื่อสารดิจิตอลการวัดสมรรถนะของระบบ จะพิจารณาจากความน่าจะเป็นของความผิดพลาด(probability of error)เป็นเรื่องหลักยกตัวอย่างเช่น วงจรภาครับไบนารีเฟสชิพท์คีย์อิงค์(Binary Phase shift keying) สมมติให้สัญญาณที่รับได้ $r(t)$ ถูกนำไปคูณกับฟังก์ชันเบซิส $\phi_1(t)$ ก่อนที่จะเข้าสู่วงจรอินทิเกรทจนครบคาบของเวลาการส่งหนึ่งบิท ตั้งแต่เวลา 0 ถึง T_b ค่าที่ได้จึงนำไปป้อนเข้าสู่วงจรตัดสินใจ(decision device)เพื่อตัดสินใจว่าสัญญาณที่รับได้นั้นน่าจะเกิดจากการส่งข้อมูลดิจิตอล 0 หรือ 1 เกณฑ์ที่ใช้ในการตัดสินใจบิทว่าน่าจะเป็น 0 หรือ 1 จัดว่าเป็นประเด็นสำคัญที่เราต้องพิจารณาอย่างรอบคอบในรายละเอียด ก่อนอื่นให้พิจารณาแผนภาพซิกแนลสเปซอยู่ใกล้จุดที่ $\phi_1 = \sqrt{E_b}$ มากกว่าก็จัดให้อยู่ในพื้นที่ Z_1 และถ้าจุดใดบนซิกแนลสเปซอยู่ใกล้จุดที่ $\phi_1 = -\sqrt{E_b}$ มากกว่าก็จัดให้อยู่ในพื้นที่ Z_2



รูปที่2.20 แผนภาพซิกแนลสเปซสำหรับการตัดสินใจบิทของระบบ BPSK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อแบ่งเขตเรียบร้อยแล้วเราก็สามารถนำรูปดังกล่าวมาใช้ช่วยในการตัดสินบิทนั่นคือถ้าสัญญาณที่รับได้ตกอยู่ในพื้นที่ Z_1 ก็ตัดสินใจว่าเป็นสัญญาณ $s_1(t)$ ที่ถูกส่งออกจากต้นทาง(ข้อมูลดิจิตอลเป็น 1) และถ้าสัญญาณที่รับได้ตกอยู่ในพื้นที่ Z_2 ก็ตัดสินใจว่าเป็นสัญญาณ $s_2(t)$ ที่ถูกส่งออกจากต้นทาง(ข้อมูลดิจิตอลเป็น 0)

การตัดสินใจโดยอาศัยกฎเกณฑ์ดังกล่าวนี้สามารถก็ให้เกิดความผิดพลาดได้และมีความผิดพลาดเกิดขึ้นได้ 2 รูปแบบ สำหรับความผิดพลาดแบบที่หนึ่ง ต้นทางตั้งใจส่งข้อมูลดิจิตอลเท่ากับ 0 ออก จึงได้ให้กำเนิดสัญญาณ $s_2(t)$ และส่งออก หากแต่ระดับของสัญญาณรบกวนเกาส์มีขนาดสูงมากจนทำให้สัญญาณที่รับได้กลับไปตกอยู่ในพื้นที่ Z_1 กรณีเช่นนี้ทำให้วงจรภาครับมีการตัดสินใจที่ผิดพลาด สำหรับความผิดพลาดแบบที่ 2 ต้นทางตั้งใจส่งข้อมูลดิจิตอลเท่ากับ 1 ออก จึงได้ให้กำเนิดสัญญาณ $s_1(t)$ และส่งออก หากแต่ระดับของสัญญาณรบกวนเกาส์มีขนาดสูงมากจนทำให้สัญญาณที่รับได้กลับไปตกอยู่ในพื้นที่ Z_2 ผลที่เกิดขึ้นคือ วงจรภาครับมีการตัดสินใจผิดพลาดเช่นเดียวกัน

ในการคำนวณหาค่าความน่าจะเป็นของความผิดพลาดของทั้ง 2 รูปแบบจะอาศัยความสัมพันธ์ที่ได้อธิบายไว้ในสมการ แต่ก่อนอื่นต้องนำสัญญาณที่รับได้ $r(t)$ ไปผ่านวงจรอินทิเกรทตามขั้นตอนการทำงานของวงจรภาครับ โดยผลลัพธ์ที่ได้มีค่าเท่ากับ

$$r_1 = \int_0^{T_b} r(t)\phi_1(t)dt \tag{2.77}$$

สำหรับกรณีความผิดพลาดแบบที่ 1 ต้นทางส่งสัญญาณ $s_2(t)$ ออก เพื่อแทนข้อมูลดิจิตอล 0 ดังนั้น $s_{21} = -\sqrt{E_b}$ เมื่อคำนวณฟังก์ชันความน่าจะเป็นจริงตามสมการที่ จะได้ว่า

$$\begin{aligned} P(r_1/0) &= \frac{1}{\sqrt{\pi N_o}} \exp\left[-\frac{(r_1 - s_{21})^2}{N_o}\right] \\ &= \frac{1}{\sqrt{\pi N_o}} \exp\left[-\frac{(r_1 + \sqrt{E_b})^2}{N_o}\right] \end{aligned} \tag{2.78}$$

ค่าความน่าจะเป็นของความผิดพลาดแบบที่ 1 ซึ่งมีค่าเท่ากับความน่าจะเป็นที่ต้นทางส่งข้อมูลดิจิตอลเป็น 0 แต่วงจรภาครับตัดสินใจว่าเป็นข้อมูลดิจิตอล 1 จึงมีค่าเท่ากับ

$$\begin{aligned} P_{e0} &= \int_0^{\infty} P(r_1/0)dr_1 \\ &= \int_{-\infty}^{0} \frac{1}{\sqrt{\pi N_o}} \exp\left[-\frac{(r_1 - \sqrt{E_b})^2}{N_o}\right] dr_1 \end{aligned} \tag{2.79}$$

ทั้งนี้ระดับแรงดันเทรชโฮลด์(Threshold Voltage) ที่ใช้ในการตัดสินใจมีค่าเท่ากับ 0 จากนั้นเราจะอาศัยคุณสมบัติพื้นฐานของการอินทิเกรทคือ เปลี่ยนตัวแปรจาก r_1 เป็น z โดยที่

$$z = \frac{1}{\sqrt{N_o}}(r_1 + \sqrt{E_b}) \tag{2.80}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งทำให้เราสามารถแสดงค่า P_{e0} ได้กระชับขึ้นดังนี้

$$P_{e0} = \frac{1}{\sqrt{\pi}} \int_{\sqrt{E_b/N_o}}^{\infty} \exp(-z^2)dz$$

$$= \frac{1}{2} erfc\left[\sqrt{\frac{E_b}{N_o}}\right] \tag{2.81}$$

โดยที่ $erfc(*)$ คือ ฟังก์ชันมาตรฐานที่ใช้กันทั่วไป

สำหรับความผิดพลาดแบบที่ 2 ซึ่งเกิดขึ้นเมื่อต้นทางได้ส่งข้อมูลดิจิตอลเป็น 1 แต่ทางวงจรภาครับตัดสินบิทดังกล่าวมีค่าเป็น 0 การคำนวณค่าความผิดพลาดนี้ไม่แตกต่างไปจากการคำนวณความผิดพลาดแบบที่1 แต่อย่างใด ในกรณีนี้ต้นทางส่งสัญญาณ $s_1(t)$ ออก เพื่อแทนข้อมูลดิจิตอล 1 ดังนั้น $s_{11} = \sqrt{E_b}$ เมื่อคำนวณฟังก์ชันความน่าจะเป็นจริงตามสมการที่(2.81) จะได้ว่า

$$P(r_1/1) = \frac{1}{\sqrt{\pi N_o}} \exp\left[-\frac{(r_1 - s_{11})^2}{N_o}\right]$$

$$= \frac{1}{\sqrt{\pi N_o}} \exp\left[-\frac{(r_1 - \sqrt{E_b})^2}{N_o}\right] \tag{2.82}$$

ค่าความน่าจะเป็นของความผิดพลาดแบบที่ 2 ซึ่งมีค่าเท่ากับความน่าจะเป็นที่ต้นทางส่งข้อมูลดิจิตอลเป็น 1 แต่วงจรภาครับตัดสินใจว่าเป็นข้อมูลดิจิตอล 0 จึงมีค่าเท่ากับ

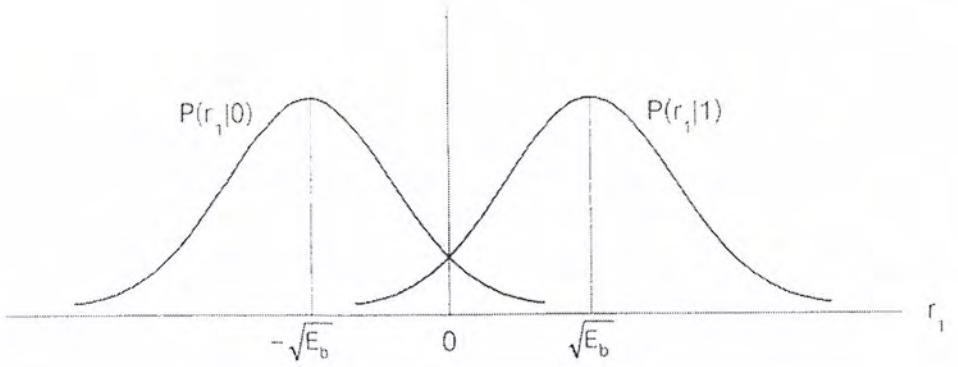
$$P_{e1} = \int_{-\infty}^{0} P(r_1/1)dr_1$$

$$= \int_{-\infty}^{0} \frac{1}{\sqrt{\pi N_o}} \exp\left[-\frac{(r_1 - \sqrt{E_b})^2}{N_o}\right] dr_1 \tag{2.83}$$

สังเกตว่าค่าความน่าจะเป็น $P(r_1/0)$ และ $P(r_1/1)$ เมื่อนำมาแสดงบนซิกแนลสเปซพบว่ามีความสมมาตรกันเมื่อเทียบกับ 0 ดูรูปที่ 2.21 ประกอบ อาศัยคุณสมบัติการอินทิเกรทในลักษณะเดียวกับการคำนวณของความผิดพลาดแบบแรก จะได้ว่าค่าความผิดพลาดแบบที่ 2 P_{e1} มีค่าเท่ากับความผิดพลาดแบบที่ 1 P_{e0}

$$P_{e1} = \frac{1}{2} erfc\left[\sqrt{\frac{E_b}{N_o}}\right] \tag{2.84}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 ความน่าจะเป็นแบบมีเงื่อนไขของการส่งสัญญาณดิจิตอล 0 และ 1

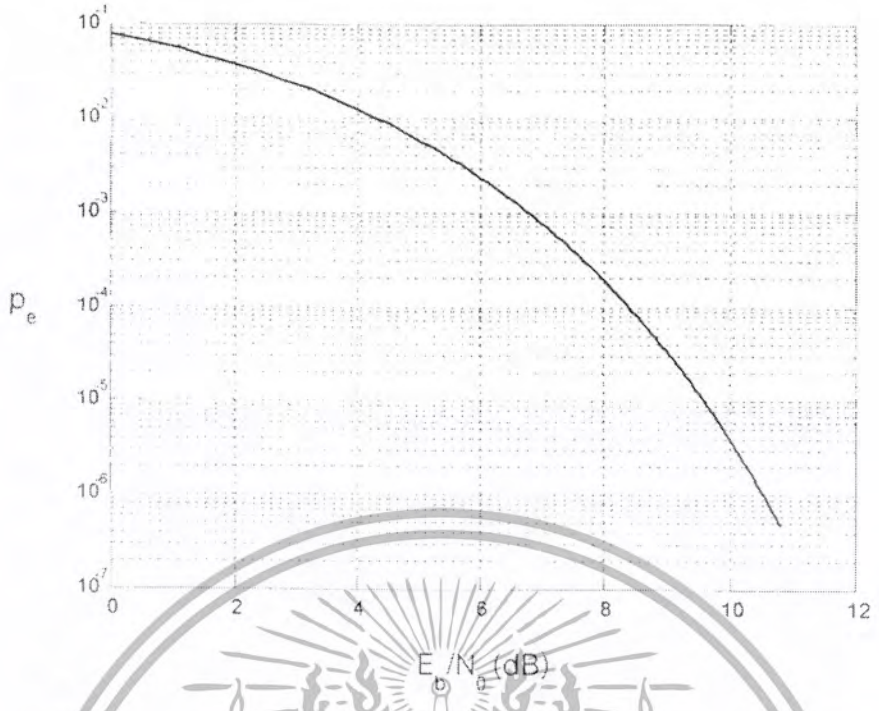
สมมติว่าต้นทางส่งข้อมูลดิจิตอล 0 และ 1 ด้วยความน่าจะเป็นที่เท่ากัน ฉะนั้นถ้านำความผิดพลาดทั้ง 2 แบบมาพิจารณารวมกันเพื่อหาค่าเฉลี่ยของความผิดพลาดโดยรวมจะได้ว่ากรณีการมอดูเลทแบบไบนารีเฟสชิฟท์คีย์อิงค์ จะมีค่าคว่มน่าจะเป็นของความผิดพลาดเท่ากับ

$$P_e = \frac{1}{2}P_{e0} + \frac{1}{2}P_{e1}$$

$$= \frac{1}{2}erfc\left[\sqrt{\frac{E_b}{N_o}}\right] \qquad (2.85)$$

จากสมการความสัมพันธ์ถ้าเราเพิ่มพลังงานต่อบิท E_b ของสัญญาณขึ้น โดยที่ระดับของสัญญาณรบกวนไม่เปลี่ยนแปลง จุดของสัญญาณทั้งคู่บนซิกแนลสเปซก็จะห่างกันมากขึ้นและช่วยทำให้ความน่าจะเป็นของความผิดพลาด P_e มีค่าลดลงตามไปด้วย ดูรายละเอียดความสัมพันธ์ระหว่างค่าความน่าจะเป็นของความผิดพลาด P_e กับขนาดของ $\frac{E_b}{N_o}$ สำหรับระบบไบนารีเฟสชิพท์คีย์อิงค์ได้ในรูปที่ 2.22 จากรูปจะเห็นว่าถ้าขนาดของ $\frac{E_b}{N_o}$ มีค่าเกิน 10 dB ความน่าจะเป็นของความผิดพลาดจะต่ำในระดับ 10^{-6} กล่าวคือ ในการส่งข้อมูลจำนวนล้านบิทจะมีโอกาสผิดพลาดโดยเฉลี่ยเพียงบิทเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

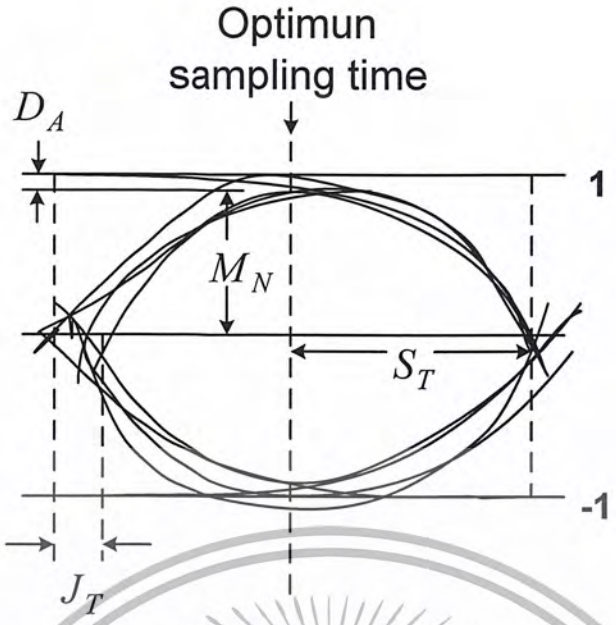


รูปที่ 2.22 ความสัมพันธ์ระหว่างความน่าจะเป็นของความผิดพลาด P_e กับขนาดของ $\frac{E_b}{N_o}$ สำหรับระบบBPSK

2.10 ไดอะแกรมรูปตา(Eye Diagram)

ไดอะแกรมรูปตานั้นจะแสดงผลลัพธ์ที่ได้จากการวัดผลตอบสนองของระบบสัญญาณเบสแบนด์ โดยได้จากการต่อเพลทแนวตั้ง(plate vertical)ของออสซิลโลสโคปเข้ากับผลตอบสนองของผู้รับซึ่งเป็นแรนดอมพัลส์(random pulse)และต่อรูปคลื่นฟันเลื่อย(sawtooth wave)ที่ความถี่เดียวกับสัญญาณเข้ากับเพลทแนวนอน(plate vertical) โดยที่เวลาในแกนแนวนอนของออสซิลโลสโคปจะถูกเซ็ทให้มีค่าเท่ากับคาบเวลาของสัญลักษณ์(symbol)หรือพัลส์ ซึ่งจะทำให้เกิดการซ้อนทับกันของรูปคลื่นในแต่ละช่วงเวลา จนเป็นเหมือนกลุ่มเส้นทางเดินใน 1 ช่วงเวลา(0,T) จากรูปที่ 2.23 แสดงให้เห็นถึงไดอะแกรมรูปตาที่ได้จากผลลัพธ์ของสัญญาณพัลส์ไบโพลาร์(bipolar pulse) เพราะว่าสัญลักษณ์ที่เกิดจากการสุ่มนั้นบางเวลามีค่าเป็นบวกและบางเวลามีค่าเป็นลบ ทำให้ผลลัพธ์ที่ได้มีรูปเหมือนตา โดยที่เวลาการสุ่มสัญญาณที่ดีที่สุดจะทำให้รูปตามีช่วงเปิดสูงที่สุดแสดงว่าสามารถหลีกเลี่ยงสัญญาณรบกวนได้มาก ถ้าในระบบที่ไม่มีฟิลเตอร์และมีแบนด์วิดท์ที่ใช้ในการส่งพัลส์ข้อมูลมีค่าไม่จำกัด ผลตอบสนองของระบบควรจะเป็นรูปพัลส์สี่เหลี่ยม ในกรณีนี้ไดอะแกรมควรที่จะมองเหมือนกล่องมากกว่ารูปตา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

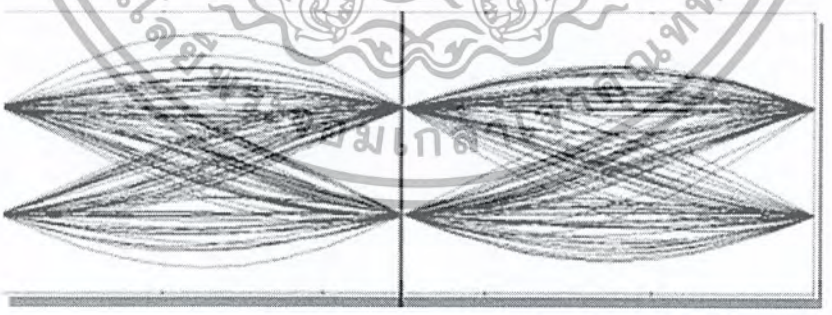


รูปที่ 2.23 ภาพ eye diagram

จากรูป ค่า D_A คือ ค่าที่ได้จากการวัดความผิดเพี้ยนที่เกิดจากการแทรกสอดระหว่างสัญลักษณ์ J_T คือ ค่าทีได้จากการวัดช่วงเวลาจิตเตอร์ , M_N คือ ค่าที่ได้จากการวัดค่าสัญญาณรบกวนและ S_T คือ ความไวของช่วงเวลาการเกิดความผิดพลาด ดังแสดงให้ดูในรูป ดังนั้นไดอะแกรมรูปตาจะใช้สำหรับวัดค่าของการสอดแทรกระหว่างสัญลักษณ์ ซึ่งเท่ากับว่าเมื่อรูปตาปิดจะหมายความว่าค่าการสอดแทรกระหว่างสัญลักษณ์มีค่าเพิ่มขึ้น ถ้ารูปตาเปิดจะหมายความว่าค่าการสอดแทรกระหว่างสัญลักษณ์มีค่าลดลง

ตัวอย่างของไดอะแกรมรูปตาที่ได้จากฟิลเตอร์แบบต่างๆ

- ไนควิสต์ฟิลเตอร์(Nyquist filter)

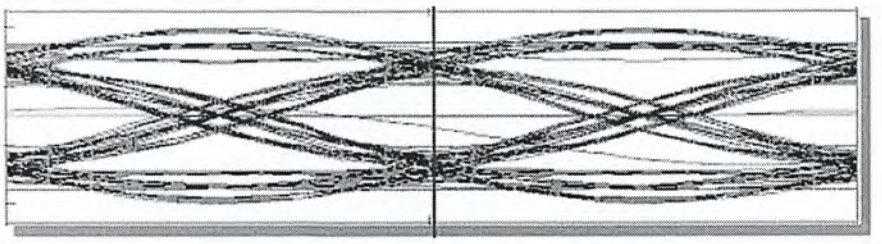


รูปที่ 2.24 ไดอะแกรมรูปตาที่ได้จากไนควิสต์ฟิลเตอร์

จากรูปจะเห็นว่าไดอะแกรมรูปตานั้นมีลักษณะช่วงเปิดที่กว้างมาก แสดงว่ามีการเกิดการสอดแทรกระหว่างสัญลักษณ์น้อย จึงทำให้ไนควิสต์ฟิลเตอร์นั้นไม่มีการสอดแทรกระหว่างสัญลักษณ์แต่ไม่สามารถสร้างได้จริงในทางปฏิบัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เรสต์โคไซน์ฟิลเตอร์(Raised cosine filter)

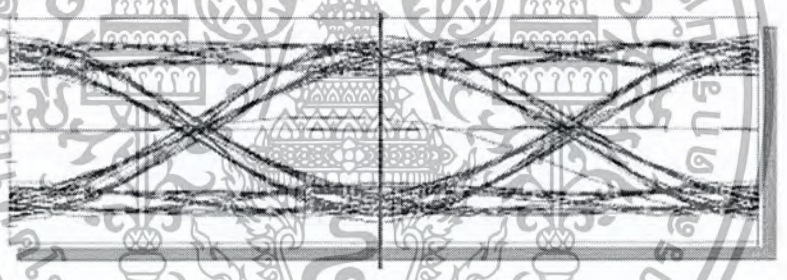


รูปที่ 2.25 ไดอะแกรมรูปตาที่ได้จากเรสต์โคไซน์ฟิลเตอร์

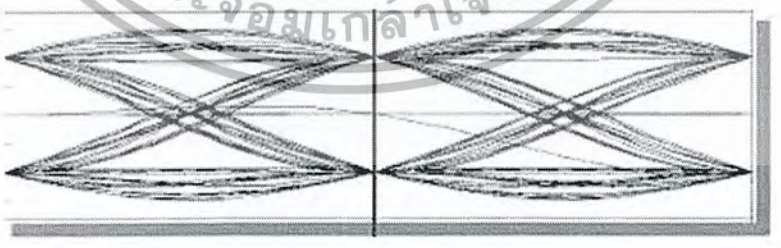
จากรูปไดอะแกรมรูปตาจะเห็นว่าเรสต์โคไซน์ฟิลเตอร์จะมีลักษณะช่วงเปิดของรูปตาที่แคบกว่าไนควิสต์ฟิลเตอร์ แสดงว่ามีการเกิดการสอดแทรกระหว่างสัญลักษณ์มากกว่า แต่สามารถใช้งานจริงในทางปฏิบัติ

ส่วนไดอะแกรมรูปตาที่ได้จากเอาท์พุทของแมทช์ฟิลเตอร์ก็จะมีลักษณะเหมือนกันและมีการเกิดการสอดแทรกระหว่างสัญลักษณ์ด้วย

- รูทเรสต์โคไซน์ฟิลเตอร์(Root raised cosine filter)



รูปที่ 2.26 ไดอะแกรมรูปตาที่ได้จากรูทเรสต์โคไซน์ฟิลเตอร์ทางด้านส่ง



รูปที่ 2.27 ไดอะแกรมรูปตาที่ได้จากรูทเรสต์โคไซน์ฟิลเตอร์ทางด้านรับ

เราจะใช้รูทเรสต์โคไซน์ฟิลเตอร์ทั้งทางด้านส่งและทางด้านรับ โดยที่ทางด้านส่งจะมีการส่งสัญญาณที่มีการสอดแทรกทางสัญลักษณ์ไปด้วย แต่การสอดแทรกทางสัญลักษณ์นี้จะถูกกำจัดออกไปที่ทางด้านรับ เมื่อพิจารณาจากรูปไดอะแกรมรูปตาที่ทางด้านส่งจะมีช่องเปิดที่น้อยกว่าไดอะแกรมรูปตาที่ทางด้านรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย

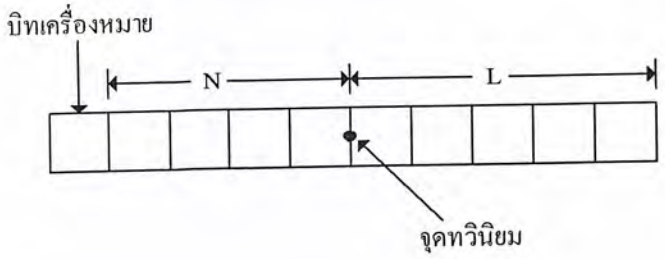
โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) หรือเรียกอย่างย่อๆว่า "DA" เป็นการจัดรูปแบบทางคณิตศาสตร์ของสมการที่อยู่ในรูปของผลบวกของผลคูณ (Sum of Product) ของเลขฐานสิบให้กระจายออกเป็นบิทหรือในรูปแบบของเลขฐานสอง เพื่อให้การคำนวณทางคณิตศาสตร์สามารถแปลงเป็นวงจรดิจิตอลอิเล็กทรอนิกส์ได้ โดยจะปรากฏอยู่ในงานทางด้านการประมวลผลสัญญาณเชิงเลข หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจายที่นำมาใช้งานด้านการประมวลผลสัญญาณเชิงเลข คือการแปลงฟังก์ชันถ่ายโอน (Transfer Function) ซึ่งเป็นสมการที่อยู่ในรูปผลบวกของผลคูณระหว่างสัญญาณอินพุทกับค่าสัมประสิทธิ์ของวงจรกรอง โดยในการคูณกันระหว่างค่าสัมประสิทธิ์ของวงจรกรองกับสัญญาณอินพุท จะใช้การคูณเลขฐานสองแบบส่วนเติมเต็มสอง(2's complement) และการคูณจะใช้แบบเปิดตาราง (Look-up table) โดยค่าผลบวกของผลคูณระหว่างสัมประสิทธิ์และสัญญาณอินพุทจะถูกเก็บไว้ในหน่วยความจำ EPROM และจะใช้สัญญาณอินพุทเป็นแอดเดรสของ EPROM โดยตรง ทั้งค่าสัมประสิทธิ์ของวงจรกรองและสัญญาณอินพุทจะถูกแทนด้วยเลขส่วนเติมเต็มสอง ดังนั้นโครงสร้างเลขคณิตกระจายจึงมีทฤษฎีพื้นฐานอยู่บนการคูณแบบเลขส่วนเติมเต็มสอง (2's complement Multiplication)

2.11.1 ระบบตัวเลข

สำหรับระบบเชิงเลข ตัวเลขต่างๆจะถูกแทนด้วยเลขฐานสอง ซึ่งโดยทั่วไปมีรูปแบบที่นิยมใช้กันอยู่ 2 รูปแบบ คือ รูปแบบจำนวนโดยตรง (Fixed point format) และ รูปแบบจำนวนอิงครรชนี (Floating point format) ซึ่งรูปแบบจำนวนโดยตรงจะมีวงจรฮาร์ดแวร์ที่ใช้ในการคำนวณที่ง่ายกว่า แต่ให้ค่าจากการคูณค่อนข้างจำกัด ส่วนรูปแบบจำนวนอิงครรชนีจะสามารถแทนค่าของสัญญาณ คือให้ย่านพลวัตร (Dynmic range) ได้มากกว่า แต่ต้องใช้วงจรฮาร์ดแวร์ที่สลับซับซ้อน แพงกว่า และให้ความเร็วในการประมวลผลที่ลดลง

1. รูปแบบจำนวนโดยตรง

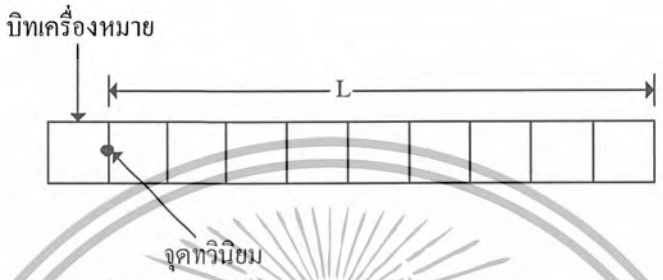
รูปแบบจำนวนโดยตรงปกติจะประกอบไปด้วย 3 ส่วน คือ บิทเครื่องหมาย (Sign bit) 1 บิท บิทจำนวนเต็ม (Integer bit) N บิท และบิทเศษส่วน (Fractional bit) L บิท โดยจะมีจุดทวินิยม (Binary point) อยู่ระหว่างบิทจำนวนเต็มและบิทเครื่องหมายดังแสดงในรูปที่ 2.28



รูปที่ 2.28 การจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิทจำนวนเต็มและบิทเศษส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนบิท N เป็นตัวกำหนดย่านพลวัตรที่ต้องการ โดยถ้าเลือกให้มีจำนวนน้อยอาจทำให้เกิดการล้น (Overflow) จากการคำนวณได้ แต่ถ้าเลือกให้มีจำนวนมากความเที่ยงตรงก็จะน้อยลง ซึ่งในการสร้างวงจรกรองสัญญาณเชิงเลขโดยการแทนด้วยรูปแบบจำนวนโดยตรงนั้น นิยมที่จะทำมาตราส่วน (Scaling) เพื่อให้ขนาดของสัญญาณมีค่าอยู่ระหว่าง $-1 \le x < 1$ คือมีบิทเครื่องหมาย 1 บิท และบิทเศษส่วน L บิท ดังแสดงในรูปที่ 2.29



รูปที่ 2.29 การจัดรูปแบบจำนวนโดยตรงที่มีแต่บิทเศษส่วน

โดยทั่วไปเลขฐานสองแบบจำนวนโดยตรงแบ่งออกได้เป็น 3 รูปแบบด้วยกัน คือ (1) แบบขนาดและเครื่องหมาย (Sign magnitude) (2) แบบส่วนเติมเต็มหนึ่ง (1's complement) (3) แบบส่วนเติมเต็มสอง (2's complement) โดยคุณลักษณะที่สำคัญบางประการของการแทนตัวเลขด้วยเลขฐานสองแบบจำนวนโดยตรงทั้ง 3 รูปแบบสามารถสรุปได้ดังตารางที่ 2.2

ตารางที่ 2.2 แสดงคุณสมบัติที่สำคัญของรูปแบบจำนวนโดยตรง

ลักษณะทั่วไป	ขนาดและเครื่องหมาย	แบบส่วนเติมเต็ม 2	แบบส่วนเติมเต็ม 1
ช่วงความยาว	$-(1-2^{-L}) \le x \le (1-2^{-L})$	$-1 \le x \le (1-2^{-L})$	$-(1-2^{-L}) \le x \le (1-2^{-L})$
แสดงค่าเป็นศูนย์	0.000 และ 1.000	0.000	0.000 และ 1.111
กฎของเลขคณิต	ง่ายแต่ต้องให้มี 2 ทาง	ง่ายแต่เหมาะกับเลขลบ	ง่ายแต่ต้องระวังในการใช้
เหมาะสำหรับใช้กับเลขคณิตตามลำดับ	ไม่ค่อยดี	ดีมาก	ดี

ใน 3 รูปแบบนี้ตัวเลขแบบส่วนเติมเต็มสองเป็นที่นิยมใช้กันมากในระบบการประมวลผลสัญญาณเชิงเลข ทั้งนี้เนื่องมาจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. มีการแทนค่าเลขศูนย์ได้เพียงค่าเดียว

2. การสร้างวงจรฮาร์ดแวร์สำหรับการบวก ลบ และคูณ ของเลขส่วนเติมเต็มสองทำได้ง่ายโดยในการคูณสามารถใช้หลักการเลื่อนและบวก (Shift and add)

3. ในระหว่างผลการบวกย่อย (Partial sum) ของการบวกเลขส่วนเติมเต็มสอง สามหรือสี่จำนวนถึงแม้อาจจะเกิดการล้น (ตัวทดจากผลการบวกล้นข้ามไปทับบิทเครื่องหมาย) แต่ผลลัพธ์สุดท้ายมักให้ค่าถูกต้องเสมอ ถ้าผลบวกอยู่ในช่วง –1 ถึง $1-2^{-L}$ ดังตัวอย่าง

7/8	0.111	
+4/8	0.100	
11/8	1.011	ผลบวกย่อยที่ผิดเนื่องจากเกิดการล้น
6/8	1.010	
5/8	0.101	ผลบวกที่ถูกต้อง

2 รูปแบบจำนวนอิงดรรชนี

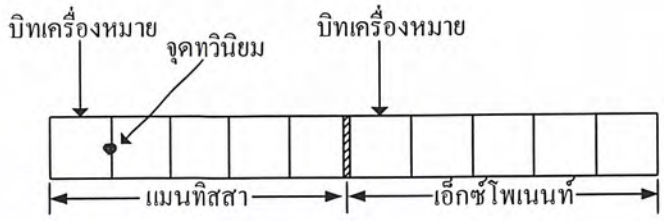
รูปแบบจำนวนโดยตรงมีข้อเสียที่สำคัญ 2 ประการ คือ (1) ย่านพลวัตรของตัวเลขมีค่าน้อย เช่น การแทนด้วยเลขส่วนเติมเต็มสอง ค่าที่น้อยที่สุดคือ –1 และค่าที่มากที่สุด คือ $1-2^{-L}$ เปอร์เซ็นต์ความผิดพลาดที่เกิดจากการตัด (Truncation) หรือการปัด (Rounding) จะเพิ่มมากขึ้นเมื่อขนาดของตัวเลขมีค่าลดลง ตัวอย่างเช่น ถ้าจำนวน 0.11011010 และ 0.000110101 ถูกตัดให้จำนวนบิทเศษส่วนเหลือเพียง 4 บิท เปอร์เซ็นต์ความผิดพลาดจะเป็น 4.59 % และ 39.6 % ตามลำดับ โดยข้อเสียนี้สามารถแก้ไขได้โดยการใช้รูปแบบจำนวนอิงดรรชนี ซึ่งตัวเลข X แสดงได้โดย

$$X = M \times 2^{e} \tag{2.86}$$

โดย e เป็นจำนวนเต็ม และ $\frac{1}{2} \le |M| < 1$

M และ e เรียกว่า แมนทิสสา (Mantissa) และ เอ็กซ์โพเนนท์ (Exponent) ตามลำดับ ตัวอย่างเช่น จำนวน 0.00110101 และ 01001.11 สามารถแทนได้โดย 0.110101×2^{-2} และ 0.100111×2^{4} ตามลำดับ ส่วนจำนวนที่มีค่าเป็นลบก็ทำในลักษณะเดียวกัน รูปแบบจำนวนอิงดรรชนีสามารถแสดงได้ดังรูปที่ 2.30 โดยแบ่งเป็น 2 ส่วน คือส่วนหนึ่งสำหรับแมนทิสสา และอีกส่วนสำหรับเอ็กซ์โพเนนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.30 การจัดรูปแบบจำนวนอิงดรรชนี

ข้อดีของการใช้จำนวนอิงดรรชนี คือแทนค่าของสัญญาณได้ละเอียดกว่า และแม่นยำกว่าแบบจำนวนโดยตรง แต่การบวก ลบ หรือคูณจะยุ่งยากกว่ามาก วงจรจึงซับซ้อนและแพงกว่าแบบจำนวนโดยตรงมาก นอกจากนี้ความเร็วในการประมวลผลยังช้ากว่าด้วย ดังนั้นสำหรับการประมวลผลแบบเวลาจริง (Real time) จึงนิยมใช้ระบบตัวเลขแบบจำนวนโดยตรง

2.11.2 ทฤษฎีเลขคณิตกระจาย

จากที่ได้กล่าวมาแล้วว่า โครงสร้างเลขคณิตกระจายมีพื้นฐานอยู่บนการคูณแบบเลขส่วนเติมเต็มสอง ดังนั้นในหัวข้อนี้จะได้อธิบายถึงหลักการของการคูณเลขส่วนเติมเต็มสอง

ให้เลขส่วนเติมเต็มสองของ X ซึ่งแทนด้วย $\overline{X}$ และนิยามโดย

$$\overline{x} = \begin{cases} x5 & \text{ถ้า } X \geq 0 \\ 2-|x| & \text{ถ้า } X < 0 \end{cases} \tag{2.87}$$

โดย X เป็นเลขที่เป็นเศษส่วน (Fractional number)

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

$$\overline{X} = X_0.X_1.X_2...X_L \tag{2.88}$$

ค่าของ $\overline{X}$ ในรูปของเลขฐานสิบสามารถหาได้ดังนี้

$$X = -X_0 + \sum_{i=1}^{L} X_i 2^{-i} \tag{2.89}$$

จากนั้นพิจารณาผลคูณต่อไปนี้

$$Y = Xm \tag{2.90}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ $\overline{Y}$, $\overline{X}$ และ $\overline{m}$ เป็นเลขส่วนเติมเต็มสองของ Y, X และ m ตามลำดับ จากนั้นพิจารณาจากสมการที่ (2.88) และ สมการที่ (2.89) จะได้

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

$$= -X_0 m + \sum_{i=1}^{L} X_i m 2^{-i} \tag{2.91}$$

ดังนั้น

$$\overline{Y} = \text{ส่วนเติมเต็มสองของ}\left(-X_0 m + 2^{-1}X_1 m + 2^{-2}X_2 m + 2^{-3}X_3 m + \ldots + 2^{-L}X_L m\right)$$

$$= \text{ส่วนเติมเต็มสองของ}\left(-X_0 m + 2^{-1}(X_1 m + \ldots + 2^{-1}(X_{L-1} m + 2^{-1}(X_L m)))\right) \tag{2.92}$$

ต่อไปพิจารณาส่วนเติมเต็มสองของ $2^{-1}U$ โดย $\overline{U} = U_0.U_1U_2\ldots U_M$ สำหรับ $U \geq 0$ (หรือ $U_0 = 0$) ส่วนเติมเต็มสองของ $(2^{-1}U) = 2^{-1}\overline{U}$ และสำหรับ $U < 0$ (หรือ $U_0 = 1$)ส่วนเติมเต็มสองของ $(2^{-1}U) = 2 - \left|2^{-1}U\right| = 1 + 2^{-1}\left(2 - |U|\right) = 1 + 2^{-1}\overline{U}$ ดังนั้นสรุปได้ว่า

$$\text{ส่วนเติมเต็มสองของ } (2^{-1}U) = \begin{cases} 2^{-1}\overline{U} & \text{ถ้า } U_0 = 0 \\ 1 + 2^{-1}\overline{U} & \text{ถ้า } U_0 = 1 \end{cases} \tag{2.93}$$

สมการที่ (2.93) นี้แสดงให้เห็นได้ว่าส่วนเติมเต็มสองของ $(2^{-1}U)$ เป็นการเลื่อนข้อมูลของ $\overline{U}$ ไปทางขวา 1 บิท

$$\therefore \text{ส่วนเติมเต็มสองของ } (2^{-1}U) = 2_2^{-1}\overline{U} \tag{2.94}$$

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

จากนั้นพิจารณาสมการที่ (2.92) และสมการที่ (2.93) จะได้ว่า

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

$$= -X_0\overline{m} + 2^{-1}(X_1\overline{m} + \ldots + 2^{-1}(X_{L-1}\overline{m} + 2^{-1}(X_L\overline{m}))) \tag{2.95}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจากสมการที่ (2.94) จะเห็นได้ว่าผลคูณจากสมการที่ (2.91) สามารถหาได้โดยการใช้หลักการเลื่อนและบวก (Shift and add) โดยผลลัพธ์ที่ได้จากการคูณแบบเลขส่วนเติมเต็มสอง สามารถหาได้ตามขั้นตอนดังนี้

1. เคลียร์ค่าข้อมูลในแอคคิวมูเลเตอร์รีจิสเตอร์
2. บวก $X_L \overline{m}$ กับค่าที่อยู่ในแอคคิวมูเลเตอร์รีจิสเตอร์
3. เลื่อนค่าที่อยู่ในแอคคิวมูเลเตอร์รีจิสเตอร์ไปทางขวา 1 บิท
4. ทำซ้ำข้อ 2 และ 3 สำหรับค่า $X_{L-1},...,X_1$
5. ลบค่า $X_0 \overline{m}$ ออกจากค่าที่อยู่ในแอคคิวมูเลเตอร์รีจิสเตอร์ (ลบแบบเลขส่วนเติมเต็มสอง)

ตัวอย่างการทำงานตามอัลกอริธึมนี้

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

$m = -0.390625$

$\overline{m} = 2 - |m|$ $\because$ m เป็นเลขลบ

$= 2 - 0.390625$

$= 1.609375$

$\therefore \overline{m} = 1.100111$

$X = 0.8125 = \overline{X}$ $\because$ X เป็นเลขบวก

$\therefore \overline{X} = 0.1101 = X_0.X_1X_2X_3X_4$

โดยมีขั้นตอนการทำงาน ดังตารางต่อไปนี้

ตารางที่ 2.3 ขั้นตอนการคูณเลขส่วนเติมเต็มสอง

การดำเนินการ	ข้อมูลในแอคคิวมูเลเตอร์รีจิสเตอร์
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
เลื่อนข้อมูลใน ACC ไปทางขวา 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

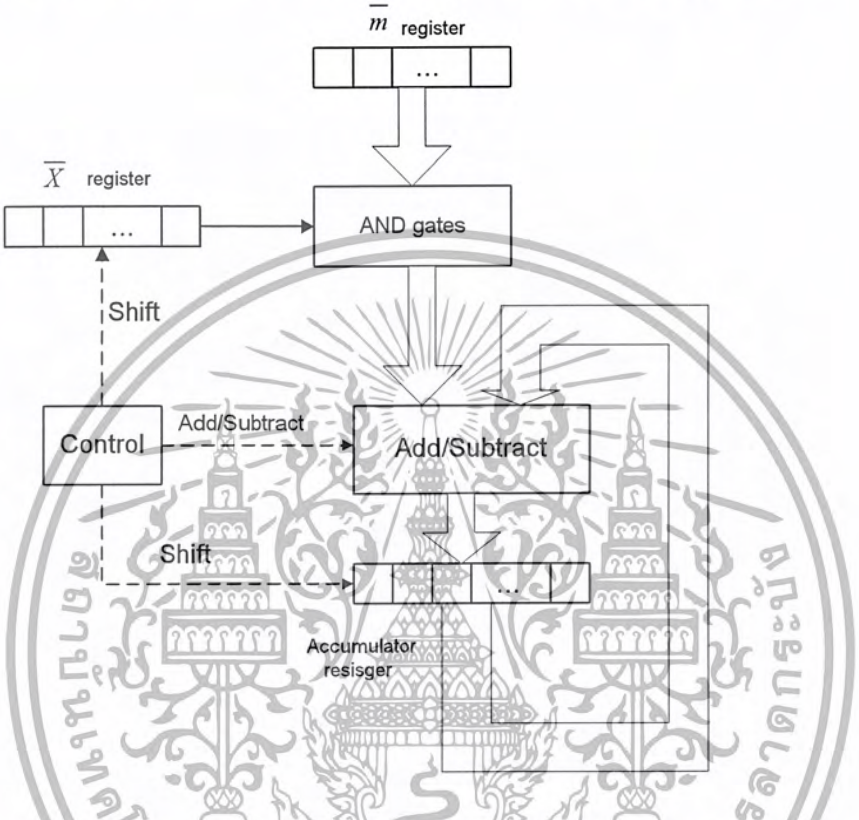
$\therefore \overline{Y} = 1.101\ \ 0111\ \ 0110 = Y_0.Y_1Y_2...Y_{11}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะได้

$$Y = -Y_0 + \sum_{i=1}^{11} Y_i 2^{-i}$$
$$= -1 + \left(2^{-1} + 2^{-3} + 2^{-5} + 2^{-6} + 2^{-7} + 2^{-9} + 2^{-10}\right)$$
$$= -0.3173828125$$

จากอัลกอริธึมดังกล่าวสามารถออกแบบการทำงานและสร้างวงจรแสดงได้ดังรูปที่ 2.31



รูปที่ 2.31 การคูณแบบเลขส่วนเติมเต็มสองโดยใช้เลขคณิตกระจาย

ที่ผ่านมาเป็นหลักการคูณแบบเลขส่วนเติมเต็มสอง ส่วนทฤษฎีเลขคณิตกระจายก็อาศัยหลักการดังกล่าวมาใช้ โดยทำการกระจายสมการที่อยู่ในรูปผลบวกของผลคูณให้แตกออกมาอยู่ในระดับบิท (Bit level)

พิจารณาผลบวกของผลคูณต่อไปนี้

$$Y = \sum_{i=0}^{N} m_i X_i \qquad (2.96)$$

โดย m_i เป็นค่าสัมประสิทธิ์ซึ่งทีค่าคงที่

X_i เป็นข้อมูลอินพุท

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X_i = -X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \tag{2.97}$$

โดย X_{ij} = บิทต่างๆของข้อมูล X_i มีค่าเป็น 0 หรือ 1

X_{i0} = บิทแสดงเครื่องหมาย

X_{iL} = บิทที่มีนัยสำคัญต่ำสุด (LSB)

$L+1$ = จำนวนบิทที่แทนข้อมูลอินพุท

แทนค่า X_i ในสมการที่ (2.97) ลงในสมการที่ (2.96)จะได้

$$Y = \sum_{i=0}^{N} m_i \left[-X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \right] \tag{2.98}$$

จัดเทอมของผลบวกใหม่จะได้

$$\begin{aligned} Y &= -X_{i0} \sum_{i=0}^{N} m_i + \sum_{j=1}^{L} X_{ij} 2^{-j} \sum_{i=0}^{N} m_i \\ &= -\sum_{i=0}^{N} X_{i0} m_i + \sum_{i=0}^{N} \sum_{j=1}^{L} X_{ij} m_i 2^{-j} \end{aligned} \tag{2.99}$$

จากนั้นทำการกระจายออกให้เป็นระดับบิท ได้ดังนี้

$$\begin{aligned} Y = &-(X_{00} m_0 + X_{10} m_1 + X_{20} m_2 + \ldots + X_{N0} m_N) \\ &+ 2^{-1} (X_{01} m_0 + X_{11} m_1 + X_{21} m_2 + \ldots + X_{N1} m_N) \\ &+ 2^{-2} (X_{02} m_0 + X_{12} m_1 + X_{22} m_2 + \ldots + X_{N2} m_N) \\ &+ \ldots + 2^{-L} (X_{0L} m_0 + X_{1L} m_1 + X_{2L} m_2 + \ldots + X_{NL} m_N) \end{aligned} \tag{2.100}$$

สมการที่ (2.100) นี้ถูกกระจายออกให้อยู่ในรูปผลบวกของผลคูณระหว่างสัมประสิทธิ์กับข้อมูลอินพุทในระดับบิท ซึ่งเป็นนิยามของการคำนวณแบบเลขคณิตกระจาย และเมื่อเปรียบเทียบสมการที่ (2.100) กับสมการที่ (2.95) จะเห็นว่าการคำนวณหาค่า Y ก็ใช้เลขคณิตกระจายนั่นเอง เพียงแต่นำค่าผลคูณย่อย (Partial product) ที่คำนวณไว้ล่วงหน้าแล้วสำหรับแต่ละค่าที่สอดคล้องกับแต่ละบิทของข้อมูลอินพุทไปเก็บไว้ในตารางเปิดดู ซึ่งเป็นหน่วยความจำ EPROM และใช้ข้อมูลอินพุทเป็นแอดเดรสของหน่วยความจำ เพื่อนำค่าในตารางเปิดดูมาผ่านขั้นตอนการคำนวณตามบูทอัลกอริธึม ซึ่งค่าในตารางเปิดดูสามารถแสดงได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 แสดงค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุท

Bit pattern ของข้อมูลอินพุท X_{Nj} ..... 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	$m_N + m_{N-1} + ... + m_2 + m_1 + m_0$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การคำนวณและการสร้าง

ในบทนี้เราจะนำทฤษฎีและหลักการออกแบบมาสร้างใช้งานจริง โดยใช้โปรแกรม MATLAB ช่วยในการออกแบบและจำลองการทำงาน ส่วนในการสร้างจะใช้บอร์ดทดลอง FPGA สำหรับระบบมีลักษณะดังนี้

- DS-CDMA
- สัญญาณที่รับคือ Root-Raised Cosine
- สัญญาณเมื่อผ่านตัวกรองคือ Raised Cosine
- กำหนดตัวกรองมีค่าโรลออฟแฟคเตอร์เท่ากับ 0.3
- เลือกใช้ตัวกรองแบบเอฟไอาร์(FIR) อันดับที่ 31
- ความละเอียดของสัญญาณเอาท์พุทอยู่ในระดับ 16 บิท

3.1 การออกแบบวงจรกรองสัญญาณดิจิตอลแบบ FIR

ในการออกแบบวงจรกรองสัญญาณเชิงเลขชนิดผลตอบสนองอิมพัลส์จำกัด(FIR filter)โดยทั่วไปจะทำการออกแบบจากอนุกรมฟูเรียร์ (Fourier Series) โดยผลตอบสนองที่ได้นี้จะเป็นผลตอบสนองในทางอุดมคติซึ่งในทางปฏิบัติค่าสัมประสิทธิ์ หรือผลตอบสนองอิมพัลส์ที่จะนำมาใช้งานนั้นจะต้องมีจำนวนที่จำกัด ซึ่งผลจากการตัดค่าผลตอบสนองอิมพัลส์ให้มีจำนวนที่จำกัดนี้ จะทำให้เกิดความพริ้ว (Ripples) ขึ้นที่ย่านผ่านและหยุดสัญญาณ รวมทั้งเกิดโอเวอร์ชูท (Overshoot) ขึ้นที่ขอบของย่านผ่านสัญญาณ ซึ่งผลที่เกิดขึ้นนี้รวมเรียกว่าปรากฏการณ์กิบส์ (Gibss's phenomenon) การลดผลที่เกิดขึ้นนี้ทำได้โดยการถ่วงน้ำหนัก(Weighted) ด้วยฟังก์ชั่นหน้าต่าง(Windows function) ชนิดต่าง

ผลตอบสนองทางความถี่ของวงจรกรองสัญญาณที่มีผลตอบสนองอิมพัลส์จำกัดจำนวน N ตัว สามารถคำนวณได้จากการแปลงฟูเรียร์เวลาต่อเนื่อง (Discrete-time Fourier Transform: DTFT) ของผลตอบสนองอิมพัลส์ได้ดังนี้

$$H(\omega) = \sum_{n=0}^{N-1} h(n) e^{-j\omega n} \tag{3.1}$$

สมมุติให้เฟสเป็นเชิงเส้น (linear phase) จะสามารถเขียนได้เป็น

$$H(\omega) = A(\omega) e^{jM\omega} \tag{3.2}$$

โดย

$A(\omega)$ คือ ผลตอบสนองแอมปลิจูด

$M = \dfrac{(N-1)}{2}$ คือ ค่าคงที่ความหน่วงกลุ่ม (Constant Group Delay) ของวงจร

สำหรับ N เป็นจำนวนคี่ จะได้โดย

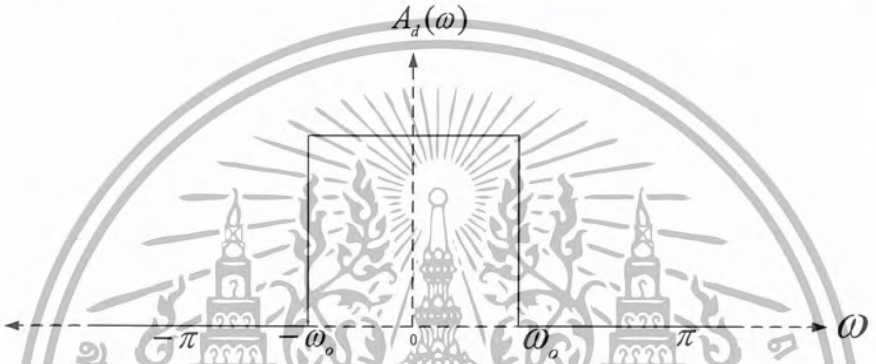
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$A(\omega) = \sum_{n=-M}^{M} \hat{h}(n)\cos(\omega_n) \tag{3.3}$$

$\hat{h}(n) = h(n+M)$ ซึ่งคือค่า $h(n)$ ที่ถูกเลื่อนไปทางซ้ายเพื่อทำให้เกิดความสมมาตร (Symmetric) ที่ n=0 ผลการแปลงกับฟูเรียร์เวลาเต็มหน่วย (Inverse DTFT) ของ $A(\omega)$ จะทำให้ผลตอบสนองอิมพัลส์ดังนี้

$$\hat{h}(n) = \frac{1}{\pi}\int_{0}^{\pi} A(\omega)\cos(\omega_n)d\omega \tag{3.4}$$

สำหรับวงจรกรองสัญญาณความถี่ต่ำผ่านในอุดมคติ ผลตอบสนองทางแอมปลิจูดสามารถแสดงได้ดังรูป



รูปที่ 3.1 ผลตอบสนองแอมปลิจูดของวงจรกรองสัญญาณความถี่ต่ำผ่านในอุดมคติ

ผลตอบสนองในทางอุดมคติจะมีค่าเป็น 1 ถ้าความถี่อยู่ในย่านผ่านสัญญาณคือ ระหว่าง 0 ถึง ω_o และมีค่าเป็น 0 ถ้าความถี่อยู่ในย่านหยุดสัญญาณ คือ ระหว่าง ω_o ถึง π (พิจารณาค่า normalized frequency คือ $\omega_s = 2\pi$) ซึ่งเขียนเป็นสมการได้โดย

$$A_d = \begin{cases} 1 \ldots\ldots\ldots\ldots\ldots\ldots 0 < \omega < \omega_o \\ 0 \ldots\ldots\ldots\ldots\ldots\ldots \omega_o < \omega < \pi \end{cases} \tag{3.5}$$

อาศัยสมการที่ (4)และสมการที่ (5) จะทำให้ได้ผลตอบสนองอิมพัลส์ในอุดมคติดังนี้

$$\hat{h}_d(n) = \frac{1}{\pi}\int_{0}^{\pi} A_d(\omega)\cos(\omega_n)d\omega$$

$$= \frac{1}{\pi}\int_{0}^{0} \cos(\omega_n)d\omega = \frac{\sin(\omega_o n)}{\pi n} \tag{3.6}$$

ซึ่งค่า $\hat{h}_d(n)$ ที่เป็นผลของสมการที่ (3.6) โดยทั่วไปจะมีจำนวนเป็นอนันต์เพื่อที่จะให้ได้ผลตอบสนองในทางอุดมคติ แต่ในทางปฏิบัติจะต้องทำการตัดให้เหลือจำนวนเพียง N ตัว และเลื่อนทางขวา M ตำแหน่งเพื่อที่จะทำให้ระบบคอซอล(Causal) ซึ่งจะทำให้ได้ผลตอบสนองอิมพัลส์เป็น

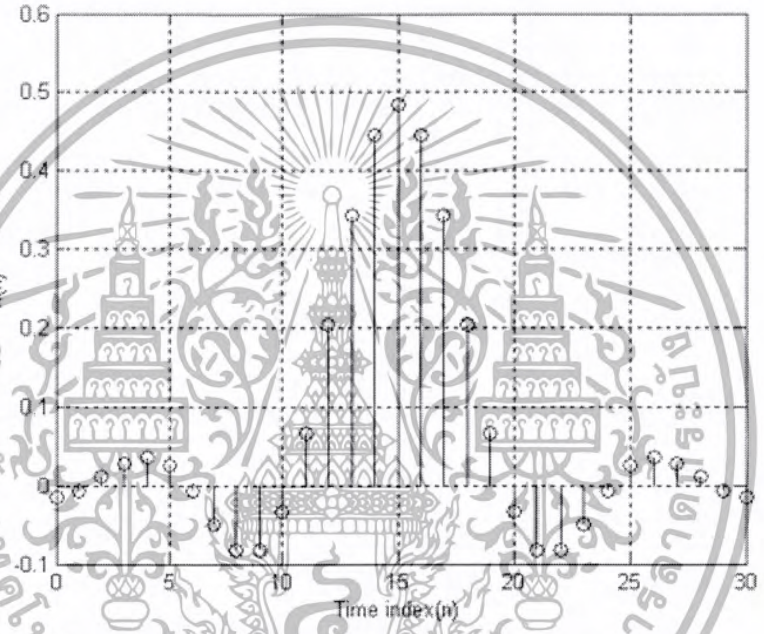
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$h(n) = \begin{cases} \dfrac{\sin(\omega_O(n-M)}{\pi(n-M)} & \text{.....สำหรับ } 0 \le n \le N-1 \\ 0 & \text{.....อื่นๆ} \end{cases} \quad (3.7)$$

ในทำนองเดียวกันเราสามารถหา $\widehat{h}_d(n)$ สำหรับวงจรกรองชนิดอื่น ๆ ได้โดยใช้วิธีการข้างต้นกระทำกับผลตอบสนองทางความถี่ของวงจรกรองสัญญาณนั้น ๆ

3.1.1 ข้อกำหนดของตัวกรอง

หาค่าสัมประสิทธิ์ของตัวกรองที่เวลา $t = nT, n = 0,1,...,30$ โดยใช้ฟังก์ชั่นใน MATLAB ได้ดังแสดงในรูปที่ 3.2

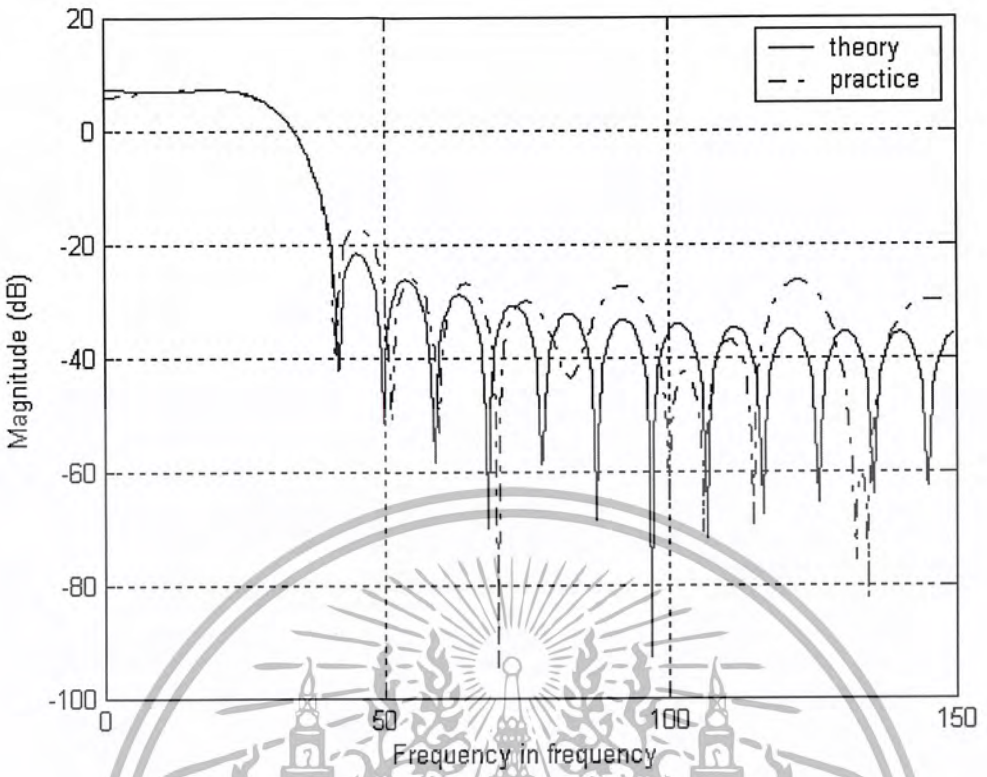


รูปที่ 3.2 สัมประสิทธิ์ของตัวกรอง Root-Raised cosine แบบ FIR อันดับที่ 31 ที่มีค่าโรลออฟแฟคเตอร์เท่ากับ 0.3

ซึ่ง $h(n)$ ที่ได้จะมีลักษณะสมมาตรปกติ (Symmetric) และเป็นไปตามเงื่อนไขของ $h(n)$ ดังนี้

$$h(n) = h(N-1-n), n = 0,1,2,...,N-1 \quad (3.8)$$

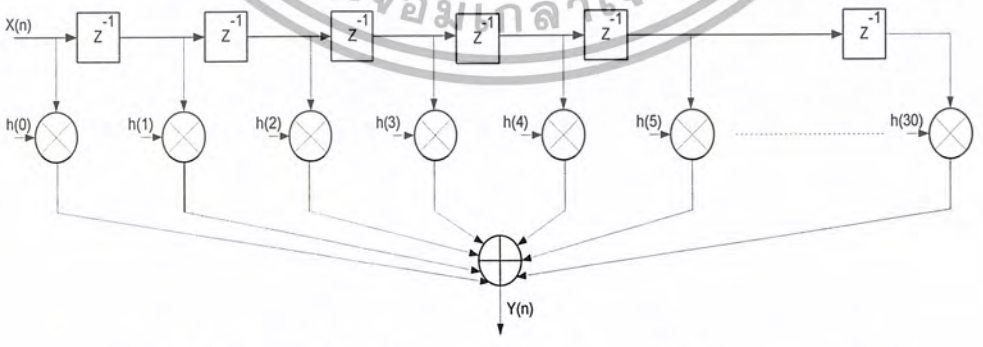
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ขนาดของตัวกรอง Root-Raised cosine แบบ FIR อันดับที่ 31 ที่มีค่าโรลออฟแฟคเตอร์เท่ากับ 0.3 มีความถี่คัทออฟที่ 30kH_z

3.1.2โครงสร้างสำหรับวงจรกรองสัญญาณดิจิตอลแบบ FIR

สำหรับวงจรกรองดิจิตอลแบบ FIR จะใช้ผลตอบสนองอิมพัลส์ หรือ $h(n)$ เพื่อสร้างวงจรซึ่งกระบวนการทำงานของวงจรกรองในที่นี้ก็คือการทำคอนโวลูชั่น ระหว่าง $h(n)$ กับสัญญาณอินพุท $x(n)$ โดยสามารถจัดโครงสร้างได้ดังรูป



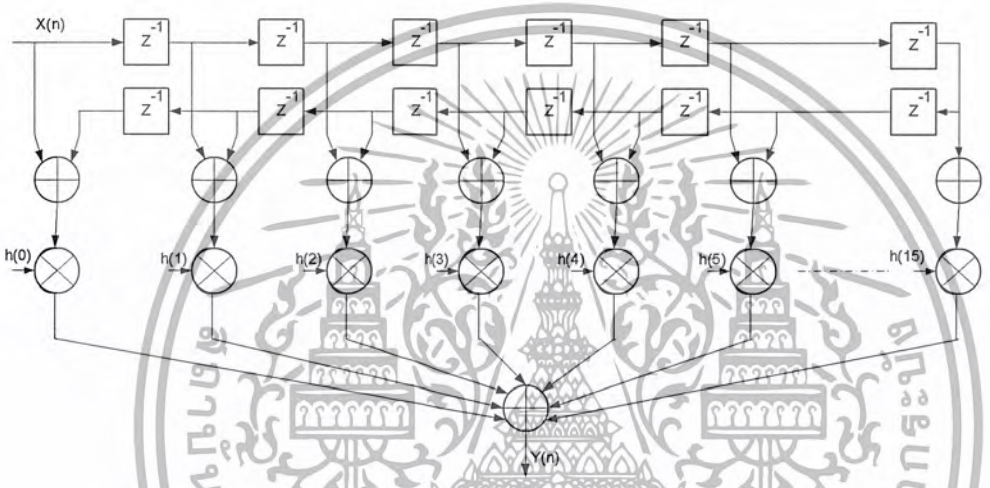
รูปที่3.4 การจัดโครงสร้างของวงจรกรองสัญญาณดิจิตอลแบบ FIR

การทำคอนโวลูชั่นระหว่าง $h(n)$)กับสัญญาณอินพุท $x(n)$ แสดงได้ดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(n) = x(n) * h(n) = \sum_{i=0}^{N-1} h(i)x(n-i) \tag{3.9}$$

ซึ่งสามารถแสดงเป็นสมการผลต่างสืบเนื่อง (Difference Equation) ได้ดังนี้ $y(n) = h(0)x(n) + h(1)x(n-1) + h(2)x(n-2) + ... + h(N-1)x(n-N+1)$ สำหรับในกรณีที่อันดับของวงจรกรองหรือ จำนวนผลตอบสนองอิมพัลส์ซึ่งก็คือ ตัวแปร N เป็นจำนวนคี่จะทำให้ผลตอบสนองอิมพัลส์มีลักษณะสมมาตรคี่ คือ $h(n) = h(N-1-n)$ เช่น กรณี N = 5 จะได้ว่า $h(0) = h(4), h(1) = h(3)$ โดยที่ 2 ไม่มีคู่สมมาตรซึ่งทำให้การจัดโครงสร้างสามารถลดจำนวนคูณลงไปได้ทำให้ขนาดของวงจรกรองเล็กลงซึ่งปกติในทางดิจิตอลตัวคูณจะมีขนาดของวงจรที่ใหญ่และใช้เวลาในการทำงานมากเมื่อเทียบกับส่วนอื่น ๆ โดยวงจรกรองในอันดับที่ 31 ที่ได้จัดโครงสร้างใหม่แสดงดังรูป



รูปที่ 3.6 ตัวกรองแบบ FIR ที่ถูกลดรูปลงตามคุณสมบัติของตัวกรอง Root-Raised Cosine

3.2 การหาค่าของตารางเปิดดู

ค่าสัมประสิทธิ์ของตัวกรองจะถูกแทนด้วยระบบตัวเลขแบบ 2'complement ขนาด 16 บิท โดยที่บิทสูงสุดเป็นบิตแสดงเครื่องหมายและอีก 15 บิท ใช้แทนค่าสัมประสิทธิ์ โดยให้อยู่หน้าจุดทศนิยม 1 บิท กับอยู่หลังจุดทศนิยม 14 บิท

จากรูปที่ 3.4 จะมีค่า $h(n)$ ที่ใช้มีจำนวน 15 ค่า ถ้าทำการใช้เพียงตารางเดียวจะต้องใช้แอดเดรสขนาด 15 บิท ซึ่งจะได้ค่าในตารางเปิดดูทั้งหมด $2^{15} = 32768$ ค่า เราสามารถทำการลดขนาดของตารางเปิดดูลงได้อีก โดยแบ่งเป็นตารางที่มีแอดเดรสขนาด 4 บิท จำนวน 4 ตัว จะได้ตารางเปิดดูของวงจรกรองความถี่ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 ตารางเปิดดูที่มีขนาด 4 บิท ตัวที่ 1

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(2)+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)

ตารางที่3.1

ตารางที่ 3.2 ตารางเปิดดูที่มีแอดเดรสขนาด 4 บิท ตัวที่ 2

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 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 ตารางเปิดดูที่มีแอดเดรสขนาด 4 บิท ตัวที่3

Address	Data	Address	Data
0000	0	1000	h(11)
0001	h(8)	1001	h(11)+h(8)
0010	h(9)	1010	h(11)+h(9)
0011	h(9)+h(8)	1011	h(11)+h(9)+h(8)
0100	h(10)	1100	h(11)+h(10)
0101	h(10)+h(8)	1101	h(11)+h(10)+h(8)
0110	h(10)+h(9)	1110	h(11)+h(10)+h(9)
0111	h(10)+h(9)+h(8)	1111	h(11)+h(10)+h(9)+h(8)

ตารางที่ 3.3

ตารางที่ 3.4 ตารางเปิดดูที่มีแอดเดรสขนาด 4 บิท ตัวที่ 4

Address	Data	Address	Data
0000	0	1000	h(15)
0001	h(12)	1001	h(15)+h(12)
0010	h(13)	1010	h(15)+h(13)
0011	h(13)+h(12)	1011	h(15)+h(13)+h(12)
0100	h(14)	1100	h(15)+h(14)
0101	h(14)+h(12)	1101	h(15)+h(14)+h(12)
0110	h(14)+h(13)	1110	h(15)+h(14)+h(13)
0111	h(14)+h(13)+h(12)	1111	h(15)+h(14)+h(13)+h(12)

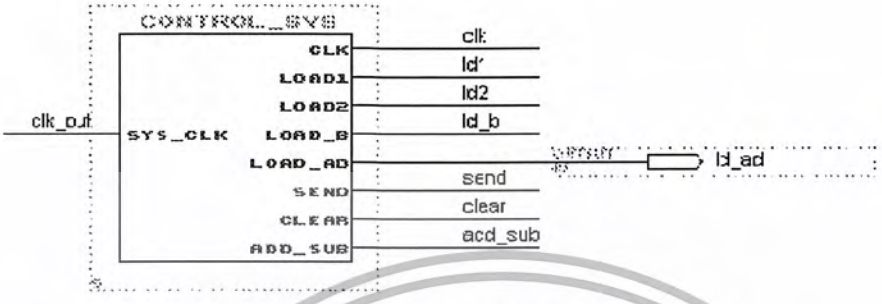
ตารางที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบอุปกรณ์

3.3.1 อุปกรณ์ควบคุม (Control)

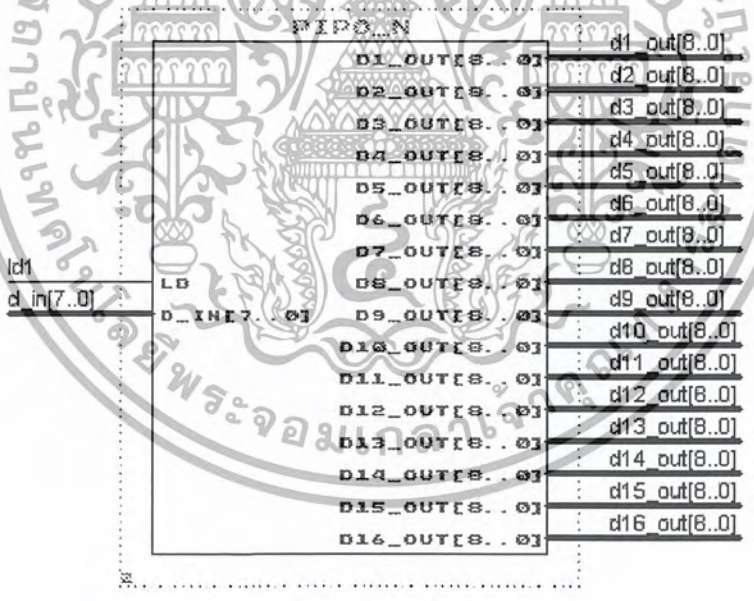
มีหน้าที่ ในการควบคุมอุปกรณ์ต่างๆภายในระบบ เช่น การป้อนคล็อกให้ระบบ, การโหลดข้อมูลเข้าระบบ, เคลียร์ข้อมูลภายในระบบ และ ส่งข้อมูลออกเอาท์พุท เป็นต้น



รูปที่ 3.6 อุปกรณ์ควบคุม

3.3.2 อุปกรณ์ PIPO

ทำหน้าที่ รับข้อมูลเข้ามาจำนวน 8 บิท และทำการบวกข้อมูลที่เข้ามาก่อน เป็นส่วนของการพับโครงสร้าง FIR ให้มีขนาดลดลง มีเอาท์พุทจำนวน 9 บิท เพื่อทำการส่งให้กับอุปกรณ์ต่อไป

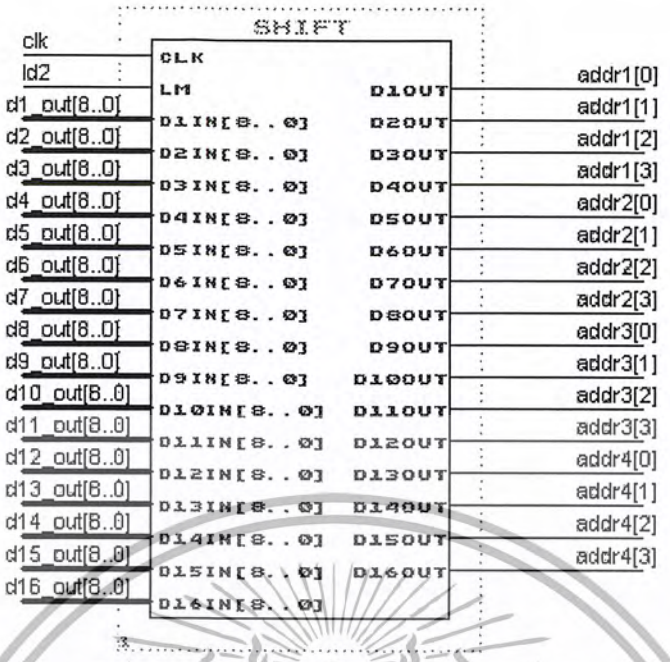


รูปที่ 3.7 อุปกรณ์ PIPO

3.3.3 อุปกรณ์ SHIFT

ทำหน้าที่รับข้อมูลเข้ามาขนาด 9 บิท นำบิทแรกของข้อมูลแต่ละตัวไปเป็นแอดเดรสเพื่อชี้ค่าสัมประสิทธิ์ออกมา โดยเอาท์พุทมีขนาด 1 บิท แบ่งออกเป็น 4 ชุดไปชี้ตารางเปิดดูทีละชุด และทำการชิพข้อมูลไปเรื่อยๆจนหมด

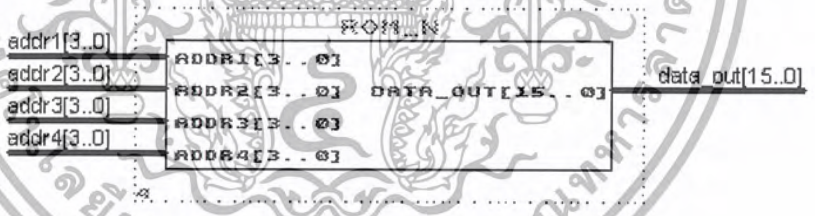
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 อุปกรณ์ SHIFT

3.3.4 อุปกรณ์ ROM

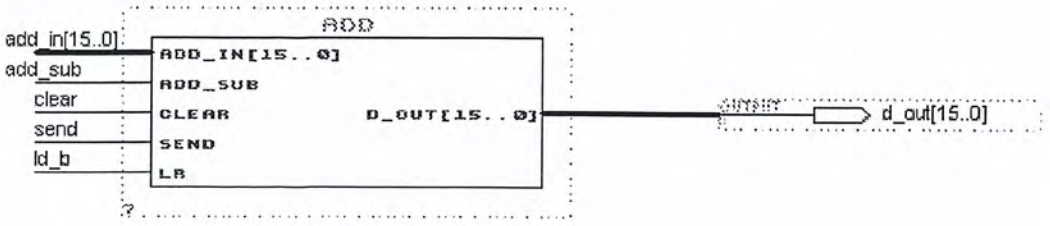
ทำหน้าที่เก็บค่าสัมประสิทธิ์ ซึ่งมีตารางเปิดดูอยู่ 4 ชุด เมื่ออินพุทขนาด 4 บิท เข้ามา ก็จะไปเป็นแอดเดรสให้กับตารางเปิดดู แล้วนำค่าสัมประสิทธิ์ตัวนั้นออกมา



รูปที่ 3.9 อุปกรณ์ ROM

3.3.5 อุปกรณ์ ADD

ทำหน้าที่ บวกค่าสัมประสิทธิ์ ที่แทนด้วยระบบตัวเลขแบบ 2'complement ขนาด 16 บิท โดยนำค่าที่ได้ก่อนมาบวกกับค่าตัวใหม่ ซึ่งจะทำการชิพค่าตัวเก่าไปทางซ้าย 1 บิทก่อน เอาท์พุทมีความละเอียดขนาด 16 บิท

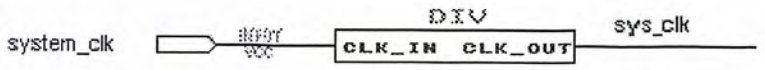


รูปที่ 3.10 อุปกรณ์ ADD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.6 อุปกรณ์หารความถี่

ทำหน้าที่ในการหารความถี่จากออสซิลเลเตอร์โมดูลบนบอร์ด FPGA ให้มีความสัมพันธ์กับความถี่ของวงจรแปลงอนาล็อกเป็นดิจิตอล



รูปที่ 3.11 อุปกรณ์หารความถี่

3.4 ภาษาวีเอชดีแอล

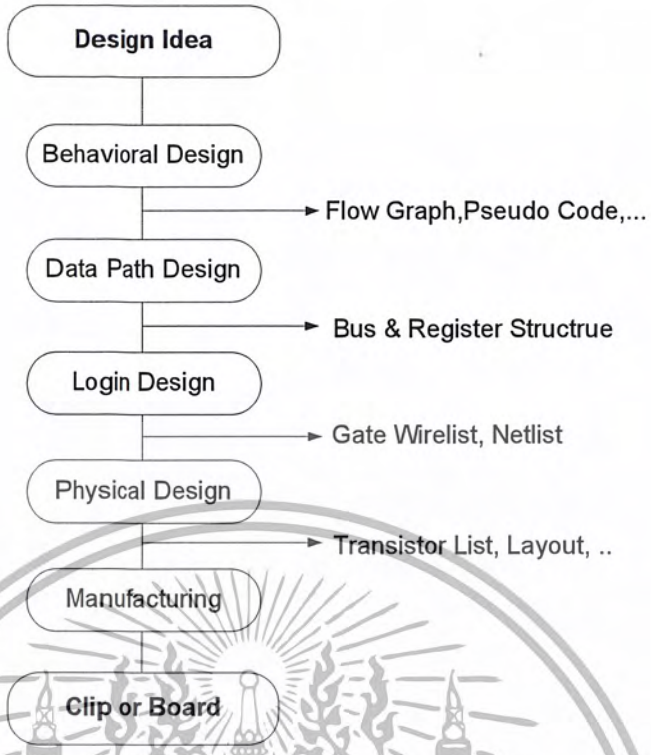
ความซับซ้อนและขนาดของระบบดิจิตอลในปัจจุบันได้เพิ่มมากขึ้นทุกขณะ ส่งผลให้มีการนำคอมพิวเตอร์เพื่อช่วยในการออกแบบ มาใช้ในกระบวนการออกแบบฮาร์ดแวร์เพิ่มขึ้นเช่นกัน อีกทั้งอุปกรณ์และวิธีการออกแบบใหม่ๆ ก็ถูกพัฒนาขึ้นมาเพื่อช่วยอำนวยความสะดวกให้กับนักออกแบบมากขึ้นด้วย สำหรับภาษาบรรยายอุปกรณ์ฮาร์ดแวร์ เอชดีแอล (HDL: Hardware Description Language) ก็เป็นเครื่องมืออย่างหนึ่งที่ได้รับการพัฒนามาอย่างต่อเนื่องเพื่อช่วยในการปรับปรุงขบวนการออกแบบระบบดิจิตอลเป็นไปอย่างมีประสิทธิภาพ

3.4.1 การออกแบบระบบดิจิตอล

ในการออกแบบระบบดิจิตอล เริ่มตั้งแต่การกำเนิดแนวความคิดเบื้องต้นจนกระทั่งได้ออกมาเป็นอุปกรณ์ฮาร์ดแวร์ที่ใช้งานได้จะต้องผ่านขั้นตอนต่างๆ มากมาย และในแต่ละขั้นตอนผู้ออกแบบจะต้องตรวจสอบผลลัพธ์ในแต่ละขั้นก่อนเข้าสู่กระบวนการออกแบบในขั้นต่อไป

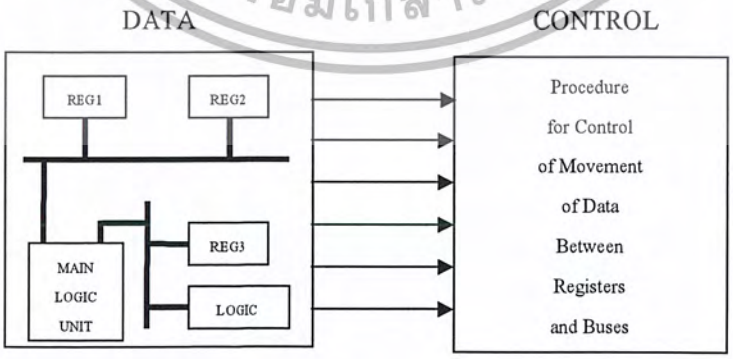
รูปที่ 3.12 แสดงขั้นตอนปกติที่ใช้ในการออกแบบระบบดิจิตอลทั่วไป ขั้นแรกผู้ออกแบบจะกำหนดแนวความคิดในการออกแบบ แล้วทำการพัฒนาให้สามารถนำไปใช้ได้อย่างสมบูรณ์ ซึ่งภายในขั้นตอนนี้ผู้ออกแบบจำเป็นต้องสร้างรูปแบบระบบเชิงพฤติกรรมขึ้นมาตรวจสอบซึ่งอาจจะเป็นผังงานแสดงแบบหรือรหัสคำสั่งเทียม (Pseudo code) ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 ขั้นตอนการออกแบบระบบดิจิตอล

ขั้นตอนต่อไปเป็นการออกแบบระบบเส้นทางของข้อมูล ผู้ออกแบบจะกำหนดส่วนประกอบของรีจิสเตอร์และวงจรลอจิกที่จำเป็นทั้งหมด เพื่อนำมาประกอบเป็นระบบที่สมบูรณ์ โดยแต่ละองค์ประกอบสามารถเชื่อมต่อกันด้วยบัสหนึ่งหรือสองทิศทาง (Unidirectional or Bidirectional Bus) กระบวนการในการควบคุมการเคลื่อนย้ายข้อมูลระหว่างรีจิสเตอร์และวงจรลอจิกจะขึ้นอยู่กับพฤติกรรมของระบบที่กำหนดไว้ดังรูปที่ 3.13



รูปที่ 3.13 การออกแบบระบบเส้นทางข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนถัดมาเป็นการออกแบบวงจรลอจิกซึ่งจะเกี่ยวข้องกับการนำเกทดิจิตอลพื้นฐาน และฟลิปฟลอป (Flip – Flop) มาประกอบเป็นอุปกรณ์ย่อยต่าง ๆ เช่นรีจิสเตอร์เก็บข้อมูล บัสวงจรลอจิก และส่วนควบคุมฮาร์ดแวร์ ซึ่งผลลัพธ์ที่ได้ในขั้นตอนนี้จะเป็นเครือข่ายของการโยงใยระหว่างเกทและฟลิปฟลอปนั่นเอง

การออกแบบในขั้นตอนถัดไปเป็นการเปลี่ยนเครือข่ายการโยงใยที่ได้จากขั้นตอนที่แล้ว ให้เป็นลำดับของทรานซิสเตอร์ (Transistor List) และ โครงงาน (Layout) ซึ่งขั้นตอนนี้จะเกี่ยวข้องกันโดยตรงกับการจัดวางทรานซิสเตอร์หรือไลบรารีเซลล์เพื่อแทนเกทและฟลิปฟลอปต่าง ๆ

และในขั้นตอนสุดท้ายจะเป็นการส่งระบบที่ออกแบบไว้ไปทำการเจือที่โรงงานเพื่อผลิตออกมาเป็นวงจรรวมในที่สุด

3.4.2 ประวัติความเป็นมาของภาษาวีเอชดีแอล

วีเอชดีแอล ย่อมาจากคำว่า VHSIC Hardware Description Language (VHSIC: Very High Speed Integrated Circuit) เป็นภาษาโปรแกรมระดับสูง (High Level Language) ที่ใช้สำหรับการออกแบบฮาร์ดแวร์ในระบบดิจิตอล ตัวของภาษาสามารถบรรยายพฤติกรรมการทำงานในรูปของลำดับชั้น (Hierarchy) และสามารถเขียนได้หลายรูปแบบด้วยเหตุผลนี้จึงทำให้ภาษาวีเอชดีแอล เป็นเครื่องมือที่ใช้ออกแบบตั้งแต่ขั้นตอนบนสุด คือ แนวความคิดที่จะแก้ปัญหา ลงไปทีละขั้นจนถึงขั้นตอนของการสร้างวงจรจริง และตัวภาษาก็เปิดโอกาสให้วิศวกรได้พัฒนาและจำลองการทำงานของรูปแบบฟังก์ชั่นการทำงานของวงจรอย่างสังเขป โดยไม่คำนึงถึงรายละเอียดเกี่ยวกับโครงสร้างวงจรจริง นอกจากนี้วีเอชดีแอล ยังเป็นภาษาที่สนับสนุนลักษณะต่างๆ ของระบบดิจิตอลที่มีความซับซ้อนได้ทั้งหมด ดังนั้นวีเอชดีแอล จึงเป็นภาษาที่น่าสนใจในการศึกษาและนำไปใช้งานเป็นอย่างยิ่ง สำหรับมาตรฐานของภาษาที่ใช้บรรยายพฤติกรรมวงจร หรือฮาร์ดแวร์ของระบบสำหรับโครงการ ไอซีวีเอชเอส ที่ ดีโอดี ได้ให้ไว้สามารถสรุปได้ดังนี้

- ต้องเป็นภาษาที่นำไปเขียนรูปแบบระบบดิจิตอล และมีคุณสมบัติที่สามารถเข้าใจได้ทั้งมนุษย์และเครื่องคอมพิวเตอร์โดยไม่ต้องมีการแปลหรือเปลี่ยนแปลงอีก
- สามารถนำไปใช้เป็นเอกสารประกอบโครงการได้
- ต้องเป็นภาษาที่เขียนขึ้นสำหรับใช้จำลองการทำงานของวงจร

ฉะนั้นภาษาดังกล่าวนี้จึงจัดเป็นภาษาโปรแกรมระดับสูง เช่น เดียวกับภาษาปาสคาล หรือภาษาซี ซึ่งในทางวิศวกรรมภาษาที่ใช้ในการออกแบบฮาร์ดแวร์นี้เรียกว่า ภาษาโปรแกรมระดับสูง

3.4.3 องค์ประกอบพื้นฐานต่างๆ ของภาษาวีเอชดีแอล

ในการเขียนรูปแบบภาษาบรรยายระบบดิจิตอล ในมุมมองของการออกแบบลักษณะบนลงล่างจะต้องทำความเข้าใจในเรื่องของโครงสร้างและส่วนประกอบต่างๆ ของรูปแบบภาษาวีเอชดีแอลเสียก่อน ซึ่งส่วนประกอบที่สำคัญและเป็นพื้นฐานของการเขียน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) หน่วยการออกแบบเอนทิตี้ (Entity Design Unit)

หน่วยการออกแบบนี้ เป็นส่วนที่ใช้สำหรับติดต่อระหว่างโลกภายนอกกับรูปแบบที่เขียนขึ้น ที่เรียกว่า หน่วยการออกแบบเอนทิตี้ ในส่วนนี้ใช้กำหนดจุดเชื่อมต่อ ของรูปแบบ กำหนดทิศทางการไหลของสัญญาณ และประเภทของค่าที่สามารถกำหนดให้กับสัญญาณตามจุดต่างๆ ของข้อมูลที่ไหลผ่านจุดต่อเหล่านั้น รูปที่ 3.14 แสดงให้เห็นโครงสร้างอย่างง่ายๆ ของ หน่วยการออกแบบเอนทิตี้

```
ENTITY component_name IS
        Input and output ports
        Physical and other parameter
END [component_name];
```

รูปที่ 3.14 โครงสร้างโดยทั่วไปของหน่วยการออกแบบเอนทิตี้

ส่วนนี้จะขึ้นต้นด้วยคำว่า ENTITY และ IS ระหว่างคำทั้งสองเป็นส่วนสำหรับชื่อของรูปแบบที่ต้องการจะเขียน (component_name) สำหรับการตั้งชื่อนั้นต้องเป็นไปตามกฎเกณฑ์ของภาษาหลังจากนั้นจะตามด้วยส่วนที่ใช้กำหนดช่องทางเข้าและออกข้อมูล (input – output) รวมทั้งพารามิเตอร์อื่นๆ ส่วนนี้เรียกว่าส่วนหัว (entity header) และที่สำคัญคือ หน่วยการออกแบบเอนทิตี้จะต้องปิดท้ายด้วยคำว่า END และเครื่องหมายอัฒภาคเสมอ (;)

2) หน่วยการออกแบบสถาปัตยกรรม (Architecture Design Unit)

คือส่วนที่ใช้เขียนบรรยายพฤติกรรมของรูปแบบ ในมุมมองของการจำลองการทำงานพฤติกรรมต่างๆ ที่บรรยายในส่วนนี้ขึ้นอยู่กับข้อมูลที่ผ่านเข้าและออก ตรงช่องทางตลอดจนพารามิเตอร์ต่างๆ ที่กำหนดในหน่วยการออกแบบเอนทิตี้ รูปที่ 3.15 แสดงให้เห็นถึงโครงสร้างอย่างง่ายๆของหน่วยการออกแบบสถาปัตยกรรม

```
ARCHITECTURE identifier OF component_name IS
[declaration]
BEGIN
        specification of the functionality of the
        component in terms of its input lines and as
        influenced by physical and other parameters
END [identifier];
```

รูปที่ 3.15 โครงสร้างโดยทั่วไปของหน่วยการออกแบบสถาปัตยกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของหน่วยการออกแบบสถาปัตยกรรม เริ่มต้นด้วยคำว่า ARCHITECTURE และตามด้วยชื่อ (identifier) สิ่งที่ต้องกำหนดลงไปได้แก่ สิ่งที่แสดงให้เห็นว่า ARCHITECTURE นั้นใช้บรรยายหน่วยการออกแบบเอนทิตี้ใดๆ (OF < entity design unit > IS) ส่วนที่อยู่ระหว่าง ARCHITECTURE และ BEGIN เป็นพื้นที่ส่วนประกาศหน่วยของสถาปัตยกรรมกำหนด (architecture declarative area) ที่เป็นเพียงส่วนเผื่อเลือก (option) ในบริเวณนี้สามารถใช้เขียนประกาศกำหนดค่าต่างๆ ที่จะนำไปใช้ภายในสถาปัตยกรรมนั้นได้ อาทิเช่นประเภท (type) ต่างๆ (ตัวอย่างเช่น bit และ bit_vector) สัญญาณ (signal) ตัวคงที่ (constant) โปรแกรมย่อย (ได้แก่ function และ procedure) และอุปกรณ์(component) ส่วนที่ใช้บรรยายความสัมพันธ์ระหว่างข้อมูลที่ไหลเข้า และไหลออกของรูปแบบ (สัญญาณที่กำหนดในชุดคำสั่ง PORT) นั้นจะถูกบรรยายในบริเวณเนื้อที่ระหว่างคำว่า BEGIN กับ END ของหน่วยการออกแบบสถาปัตยกรรม และนอกจากนั้นชุดคำสั่งทุกคำสั่งที่อยู่ภายในบริเวณนี้จะเป็นชุดคำสั่งแบบแข่งขนาน (concurrent statement) เท่านั้น หน่วยการออกแบบสถาปัตยกรรมจะต้องปิดท้ายด้วยคำสั่ง END และชื่อของสถาปัตยกรรมนั้นๆ ที่เป็นส่วนเผื่อเลือกโดยทั่วไปการเขียนรูปแบบระบบดิจิตอลด้วยภาษาวีเอชดีแอลสามารถเขียนได้ในลักษณะต่างๆ ดังนี้

- ประเภทการไหลของข้อมูล (Dataflow description)
- ประเภทพฤติกรรม (Behavioral description)
- ประเภทโครงสร้าง (Structure description)
- ประเภทผสม (Mixed model description)

3) หน่วยการออกแบบแพ็กเก็ต (Package Design Unit)

ข้อมูลต่างๆ ตลอดจนโปรแกรมย่อย (subprogram) ที่เป็นประโยชน์ต่อการเขียนรูปแบบบรรยายระบบดิจิตอล สามารถเก็บไว้ในส่วนที่เรียกว่า package ได้ และข้อมูลเหล่านี้สามารถถูกเรียกไปใช้ได้โดย entity design unit architecture design unit หรือจาก package design unit อื่นๆ ด้วยชุดคำสั่ง USE statement นอกจากนั้นสิ่งที่นิยมทำกันมากคือรูปแบบ (model) มาตรฐานต่างๆ อาทิเช่น standard components (model ของ IC ตระกูล 74xx) จะถูกเก็บไว้ในแพ็กเก็ต ที่ทุกคนสามารถเข้าถึงและนำไปใช้ได้

โดยปกติแล้ว แพ็กเก็ตจะแบ่งออกเป็น 2 ส่วนคือ การประกาศแพ็กเก็ต (Package declaration) และส่วนของบอดี้แพ็กเก็ต (Package body) เนื่องจาก แพ็กเก็ตถูกสร้างขึ้นเป็นส่วนแยกต่างหากออกจากรูปแบบที่กำลังเขียนอยู่ ฉะนั้นการที่นำแพ็กเก็ตไปใช้นั้นจะต้องมีการเชื่อมโยงหรืออ้างอิงเสียก่อน ซึ่งในภาษาวีเอชดีแอลสามารถกระทำได้ด้วยชุดคำสั่ง USE

- การประกาศแพ็กเก็ต (Package Declaration)

ส่วนที่มีความสำคัญที่สุดของแพ็กเก็ต (มองในแง่ของการนำไปใช้จากภายนอก) ได้แก่ การประกาศแพ็กเก็ต เพราะจะเป็นส่วนที่กำหนดชื่อ (identifier) ของสิ่งที่ประกาศอยู่ภายในแพ็กเก็ตสำหรับนำไปใช้ภายนอกตัวของแพ็กเก็ตเอง ถ้าสิ่งใดๆ ถูกประกาศในส่วนของบอดี้แพ็กเก็ต แต่ไม่ถูกประกาศในการประกาศแพ็กเก็ตจะไม่สามารถถูกนำค่า และพฤติกรรมไปใช้จากส่วนนอกได้ ซึ่งสามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบได้กับสิ่งที่ประกาศไว้ในส่วนของการประกาศเอนทิตี้ คือ จุดเชื่อมต่อ ที่มีหน้าที่ติดต่อกับโลกภายนอก ฉะนั้นโดยทั่วไปแล้วแพ็กเก็ตสามารถสร้างขึ้นได้โดยไม่จำเป็นต้องมีส่วนบอดี้และยังสามารถถูกนำไปใช้จากรูปแบบ (model) ภายนอกได้เช่นใช้สำหรับประกาศ ชนิด (TYPE) เช่นเดียวกันกับบอดี้แพ็กเก็ตที่ไม่จำเป็นต้องมีการประกาศแพ็กเก็ต แต่แพ็กเก็ตนั้นจะไม่สามารถนำไปใช้ ในรูปแบบ (model) อื่นได้ การเขียนการประกาศแพ็กเก็ต มีกฎเกณฑ์ตามที่แสดงในรูปที่ 3.16

```
PACKAGE package_name IS
        package_declarative_part
END package_name ;
```

รูปที่ 3.16 โครงสร้างโดยทั่วไปของส่วนการประกาศแพ็กเก็ต

- โครงสร้างของแพ็กเก็ต (Package Body)

โครงสร้างที่ประกอบด้วยคำสั่งต่างๆ ในรูปของคำสั่งลำดับ ที่ใช้บรรยายฟังก์ชันการทำงานของโปรแกรมย่อย (Subprogram) ทั้งหลายที่ชื่อของโปรแกรมย่อยนั้นๆ ที่ถูกประกาศไปในส่วนของการประกาศแพ็กเก็ต แล้วจะถูกเก็บไว้ในส่วนบอดี้แพ็กเก็ต แต่ถูกกำหนดค่าคงที่ต่างๆ อันได้แก่ตัวค่าคงที่ที่ถูกประกาศชื่อก่อนในส่วนของการประกาศแพ็กเก็ต แต่ถูกกำหนดค่าในส่วนของบอดี้แพ็กเก็ต ฉะนั้นส่วนบอดี้แพ็กเก็ต จึงไม่จำเป็นต้องมี ถ้าในส่วนของการประกาศแพ็กเก็ต ไม่มีการประกาศชื่อ ที่เป็นโปรแกรมย่อย หรือ ค่าคงที่ การเขียนบอดี้แพ็กเก็ตนั้นเป็นไปตามกฎเกณฑ์ที่แสดงในรูปที่ 3.17

```
PACKAGE BODY package_name IS
        declarative part
END package_name ;
```

รูปที่ 3.17 โครงสร้างของบอดี้แพ็กเก็ต

4) หน่วยการออกแบบโครงแบบ (Configuration Design Unit)

รูปแบบหนึ่งของระบบดิจิตอลไม่ว่าจะเป็นอะไร จะมีหน่วยการออกแบบเอนทิตี้ได้ เพียงหนึ่งเดียวเท่านั้น แต่ในขณะที่หน่วยการออกแบบเอนทิตี้ หนึ่งหน่วยนี้อาจจะมีสถาปัตยกรรม ที่เป็นหน่วยรองได้หลายหน่วย ดังนั้นจะต้องมี หน่วยการออกแบบโครงแบบมาเพื่อกำหนดการใช้โครงแบบ (Configuration) ประกอบเอนทิตี้กับหน่วยการออกแบบสถาปัตยกรรมหน่วยไหนเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CONFIGURATION identifier OF entity_name IS
        configuration_declarative_part
END;
```

รูปที่ 3.18 โครงสร้างโดยทั่วไปของหน่วยการออกแบบโครงแบบ

5) โปรแกรมย่อย

การใช้ฟังก์ชันและโพรซีเจอร์ใน วีเอชดีแอล เปรียบได้กับการใช้โปรแกรมย่อยในการเขียนโปรแกรมภาษาขั้นสูงทั่วๆ ไปค่าที่ถูกส่งกลับหรือถูกเปลี่ยนแปลงโดยโปรแกรมย่อยอาจจะมีหรือไม่มีผลต่อฮาร์ดแวร์โดยตรงก็ได้ เช่นถ้าใช้ฟังก์ชันแทนการกระทำในสมการบูลีนก็จะมีผลต่อวงจรลอจิกจริงๆ ในขณะที่ถ้าใช้โปรแกรมย่อยในการเปลี่ยนชนิดของข้อมูลหรือในการคำนวณค่าการหน่วงเวลาแล้ว ก็จะไม่มีผลต่อโครงสร้างของฮาร์ดแวร์

6) โอเปอร์เรเตอร์

การบรรยายเชิงพฤติกรรมในภาษา วีเอชดีแอล มีตัวดำเนินการหรือโอเปอร์เรเตอร์ทางลอจิกและคณิตศาสตร์เช่นเดียวกับภาษาซอฟต์แวร์ทั่วไปดังรูปที่ 3.19

```
PREDEFINE OPERATORS
LOGICAL OPERATORS : NOT AND OR NAND NORXOR
        OPERAND TYPE : BIT BOOLEAN
        RESULT TYPE : BIT BOOLEAN
RELATIONAL OPERATORS : = / = < <= > >=
        OPERAND TYPE : any type
        RESULT TYPE : Boolen
ARITHMETIC OPERATORS : + - * / ** MOD REM ABS
        OPERAND TYPE : INTEGER REAL Physical
        RESULT TYPE : INTEGER REAL Physical
        CONCANTENATION OPERATOR : &
        OPERAND TYPE : ARRAY of any type
        RESULT TYPE : array of any type
        RESULT TYPE : array of any type
```

รูปที่ 3.19 ตัวดำเนินการใน วีเอชดีแอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) เวลาและความพร้อมเพรียง

ในวงจรอิเล็กทรอนิกส์อุปกรณ์ทุกๆ ตัวจะอยู่ในสภาพเตรียมพร้อมเสมอ (Always Active) และจะมีเรื่องของเวลา เข้ามาเกี่ยวข้องในทุกๆ เหตุการณ์ที่เกิดขึ้นเสมอ วีเอชดีแอล เป็นภาษาที่ได้รับการออกแบบมาเพื่อให้สามารถบรรยายรูปแบบและการพ้องกันของเวลา สำหรับการทำงานของอุปกรณ์ได้อย่างถูกต้อง การบรรยายการทำงานที่อยู่ภายในส่วน ของการบรรยายสถาปัตยกรรม จะมีการทำงานที่พร้อมเพรียงกันเสมอ หรือแม้แต่โปรเซสซึ่งมีการทำงานภายในเป็น แบบลำดับคำสั่งก็ตาม ซึ่งหากมีหลายๆ โปรเซสอยู่ภายในโครงสร้างเดียวกัน ทุกๆ โปรเซสก็จะทำงานไปพร้อมๆ กันด้วย

8) สัญญาณและตัวแปร

สัญญาณมีลักษณะเป็นเสมือนตัวกลางฮาร์ดแวร์ ที่ใช้ในการผ่านข้อมูลและมีเรื่องของเวลามาเกี่ยวข้องด้วยการกำหนดค่าให้สัญญาณจะใช้สัญลักษณ์ <= ในการส่งค่าและสามารถใช้คำสั่ง AFTER เพื่อกำหนดช่วงเวลาในการส่งผ่านค่าของสัญญาณ เช่น <= a AFTER 12 NS หมายถึงการกำหนดค่าสัญญาณ a ให้กับ w หลังจากเวลา ผ่านไป 12 นาโนวินาที ในทางตรงข้ามตัวแปรมีลักษณะเป็นเสมือนตัวกลางที่ใช้ในการส่งผ่านข้อมูลและไม่มีเรื่องของเวลาเข้ามาเกี่ยวข้องด้วย ซึ่งตัวแปรจะถูกใช้ในส่วนที่มีการทำงานเป็นแบบลำดับคำสั่งเช่นในฟังก์ชั่น โพรซีเจอร์และโปรเซสสำหรับกำหนดค่าให้กับตัวแปรจะใช้สัญลักษณ์ : =

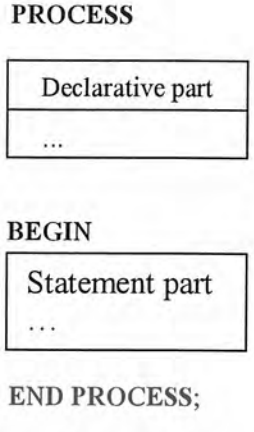
3.4.4 การบรรยายเชิงพฤติกรรม

การบรรยายลักษณะการทำงานของอุปกรณ์ฮาร์ดแวร์ในเชิงพฤติกรรม เป็นการบรรยายลักษณะการเปลี่ยนแปลงของข้อมูลในรูปแบบของอัลกอริธึม สำหรับการคำนวณผลลัพธ์ที่เกิดขึ้น ซึ่งสืบเนื่องมาจากการเปลี่ยนแปลงสภาวะของข้อมูลที่เข้ามา โดยไม่คำนึงถึงลักษณะโครงสร้าง หรือความสัมพันธ์ของอุปกรณ์ที่อยู่ภายในว่าจะเป็นอย่างไร ในหัวข้อนี้จะแสดงถึงการบรรยายเชิงพฤติกรรมแทนการใช้โมดูลฮาร์ดแวร์รวมถึงข้อกำหนดต่างๆ ที่ควรรู้

3.4.5 การโปรเซส

โปรเซสเป็นรูปแบบพื้นฐานอย่างหนึ่งที่ใช้ในการกำหนดให้กับสัญญาณ โปรเซสจะอยู่ในสถานะที่เตรียมพร้อมอยู่เสมอและจะปฏิบัติคำสั่งพร้อมๆ กันกับโปรเซสอื่นๆ ที่อยู่ในสถาปัตยกรรมบรรยายเดียวกัน โดยโปรเซสจะปฏิบัติงานตามคำสั่งทันทีที่มีเหตุการณ์เกิดขึ้นกับสัญญาณที่อยู่ทางด้านขวามือของสัญลักษณ์กำหนดค่าให้กับสัญญาณ (<=) การบรรยาย โปรเซสจะเริ่มต้นด้วยคำสั่ง PROCESS และ END PROCESS ในรูปที่ 3.20 เป็นการแสดงส่วนประกอบของการบรรยายแบบโปรเซส ซึ่งประกอบด้วยส่วนของการประกาศตัวแปรที่ต้องใช้และส่วนการปฏิบัติ คำสั่งเพื่อให้ได้ผลลัพธ์ที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 รูปแบบของการบรรยายแบบโปรเซส

3.4.6 การกำหนดตัวดำเนินการภายในโปรเซส

ตัวดำเนินการภายในโปรเซสมี 3 ชนิดคือ ตัวแปร (Variable) ไฟล์ (File) และตัวคงที่ (Constant) ซึ่งตัวดำเนิน การทั้ง 3 ชนิดนี้หากมีการประกาศไว้ในโปรเซสใดก็จะใช้ได้เฉพาะภายในโปรเซสนั้นเท่านั้นสำหรับการติดต่อภายนอก หรือระหว่างโปรเซสสามารถทำได้โดยใช้สัญญาณ (Signal) หรือตัวคงที่ที่ได้ประกาศไว้ในส่วนของ ARCHITECTURE

3.4.7 การกำหนดการกระทำภายในโปรเซส

การกระทำใดๆ ภายในโปรเซสจะเป็นการปฏิบัติแบบลำดับ (Sequential) เสมอ ซึ่งภายในโปรเซสสามารถใช้ประโยคเงื่อนไขหรือการทำซ้ำได้เช่น IF-ELSE CASE-WHEN FOR LOOP และ WHILE LOOP

3.4.8 การกระตุ้นและยับยั้งการกระทำของโปรเซส

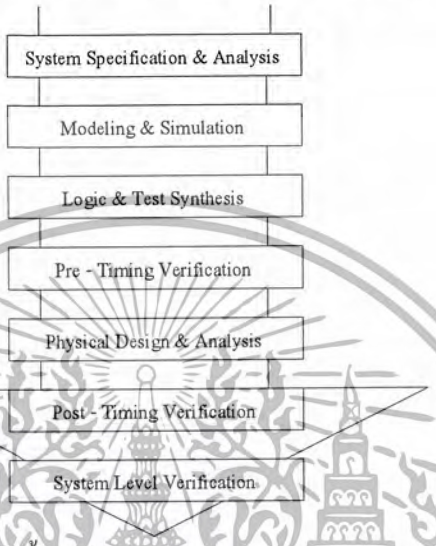
การกระทำภายในโปรเซสจะอยู่ในสภาพเตรียมพร้อม และมีการปฏิบัติงานอยู่ตลอดเวลาที่มีการเปลี่ยนแปลงของเหตุการณ์ เกิดขึ้น อย่างไรก็ดีเราสามารถกระตุ้นหรือยับยั้งการกระทำภายในโปรเซสได้โดยการกำหนดรายการของสัญญาณที่ต้องการให้โปรเซสปฏิบัติงานเมื่อมีเหตุการณ์เกิดขึ้นกับสัญญาณที่กำหนดไว้เท่านั้น ส่วนเหตุการณ์ใดๆ ที่เกิดขึ้นกับสัญญาณ ที่ไม่ได้กำหนดไว้ในรายการก็จะไม่ส่งผลให้มีการกระทำภายในโปรเซส ซึ่งรายการของสัญญาณนี้เรียกว่า Sensitivity List และกำหนดไว้ภายในวงเล็บหลังคำสั่ง PROCESS

3.4.9 การออกแบบจากบนลงล่าง (Top-Down Design)

ในการพัฒนาวงจรรวมดิจิตอลขนาดใหญ่ที่มีความซับซ้อน วิศวกรหรือผู้ออกแบบมักจะมองการออกแบบให้อยู่ในรูปของ บล็อกไดอะแกรมก่อนที่ทำวิเคราะห์ให้ลึกถึงรายละเอียดต่อไป ซึ่งภาษาวีเอชดีแอลนั้นอนุญาตให้อธิบายและวิเคราะห์การณ์ ทำงานของแต่ละบล็อก รวมถึงการปรับปรุงการทำงานจากผลที่วิเคราะห์เพื่อให้ได้การทำงานตามต้องการนอกจากนี้ยัง สามารถเพิ่มเติมในรายละเอียดใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ละขั้นตอนได้ ซึ่งหลักการนี้สอดคล้องกับหลักการออกแบบจากบนลงล่างนั่นเอง ถ้าทดลองเปรียบเทียบกับการออกแบบจากล่างขึ้นบน (Bottom-Up Design) จะเห็นได้ว่า การออกแบบจากล่างขึ้นบนจะใช้เวลาการออกแบบมากกว่า 90% เนื่องจากเป็นการวางวงจรด้วยอุปกรณ์ต่างๆ (Schematic capture) ที่ประกอบกันเข้าเป็นวงจรที่ต้องการออกแบบ ก่อนแล้วจึงทำการจำลองการทำงาน และตรวจสอบความถูกต้อง วีเอชดีแอล กับหลักการออกแบบจากบนลงล่างจึงเป็นทางออกให้กับวิศวกรให้สามารถ ออกแบบและพัฒนาวงจรที่มีความซับซ้อนได้มากขึ้น ทั้งยังช่วยลดเวลาและค่าใช้จ่ายในการออกแบบด้วย



รูปที่ 3.21 ขั้นตอนการออกแบบจากบนลงล่าง

จากรูปที่ 3.21 แสดงถึงขั้นตอนการออกแบบจากบนลงล่าง ทั้งนี้ในทางปฏิบัติอาจมีข้อแตกต่างไปจากนี้บ้าง เล็กน้อยเนื่องจากขั้นตอนของการผลิต (Implementation) สามารถกระทำได้หลายเทคโนโลยี สำหรับรายละเอียดของขั้นตอน การออกแบบจากบนลงล่างในแต่ละขั้นตอนมีดังนี้

1) ความต้องการของระบบและการวิเคราะห์ คือ การสร้างข้อกำหนดของความต้องการ และวิเคราะห์ระบบ เพื่อหาแนวความคิดและหลักการ (Idea and Concept) ในการแก้ปัญหา

2) รูปแบบและการจำลองการทำงาน คือ การเขียนรูปแบบของระบบที่ต้องการออกแบบโดยใช้ภาษา วีเอชดีแอล หรือ ภาษา เอชดีแอล อื่นๆ สำหรับบรรยายพฤติกรรมการทำงาน พร้อมทั้งจำลองการทำงาน เพื่อเปรียบเทียบและตรวจสอบความถูกต้องกับข้อกำหนด

3) ลอจิกและการทดสอบการสังเคราะห์ คือ หลังจากที่ได้หลักการขั้นต้นพร้อมแนวความคิดที่ผ่านการตรวจสอบแล้วหลักการนี้จะถูกเพิ่มเติมรายละเอียดลงมาเป็นลำดับขั้นที่สอง จนกระทั่งอยู่ในระดับที่จะนำไปผลิตวงจรจริง หรือสังเคราะห์ในขั้นตอนนี้เองเทคโนโลยีที่จะมารองรับวงจรออกแบบจะถูกกำหนดขึ้น และระบบช่วยการออกแบบจะสังเคราะห์วงจรที่ได้จากรูปแบบที่จะเขียนขึ้นให้อยู่ในรูปของวงจรที่ประกอบด้วยอุปกรณ์อิเล็กทรอนิกส์ หรือวงจรในระดับเกท และการเชื่อมต่อระหว่างกันของอุปกรณ์เหล่านั้นหรือไม่ก็อยู่ในรูปของโครงข่ายการเชื่อมต่อ ที่สามารถนำไปผลิตอุปกรณ์อื่นได้

4) การตรวจสอบเวลาก่อนการออกแบบ คือ หลังจากการสังเคราะห์วงจรให้อยู่ในระดับเกทหรือโครงข่ายการเชื่อมต่อแล้ว ข้อมูลนี้จะถูกใช้สำหรับจำลองการทำงานในเรื่องความถูกต้องของฟังก์ชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พร้อมกับนำข้อมูลที่เกี่ยวข้องกับเวลาเข้ามาประกอบการพิจารณาด้วย ซึ่งตามปกติแล้วอุปกรณ์ ทางอิเล็กทรอนิกส์ทุกชิ้นจะมีเวลาหน่วงของการแพร่กระจาย (Propagation Delay Time) เสมอ ถึงแม้ว่าจะเป็นเวลาที่น้อยมากในระดับนาโนวินาทีก็ตาม แต่ถ้าภายในวงจรหนึ่งประกอบด้วยเกทของฟังก์ชันต่างๆ จำนวน 10,000 เกท ขึ้นไป เวลาดังกล่าวนี้จะสะสมกันมากขึ้น จนอาจทำให้การทำงานของวงจรรวมทั้งหมดผิดพลาดไป หรือไม่สามารถทำ งานในย่านความถี่สัญญาณนาฬิกาที่สูงได้

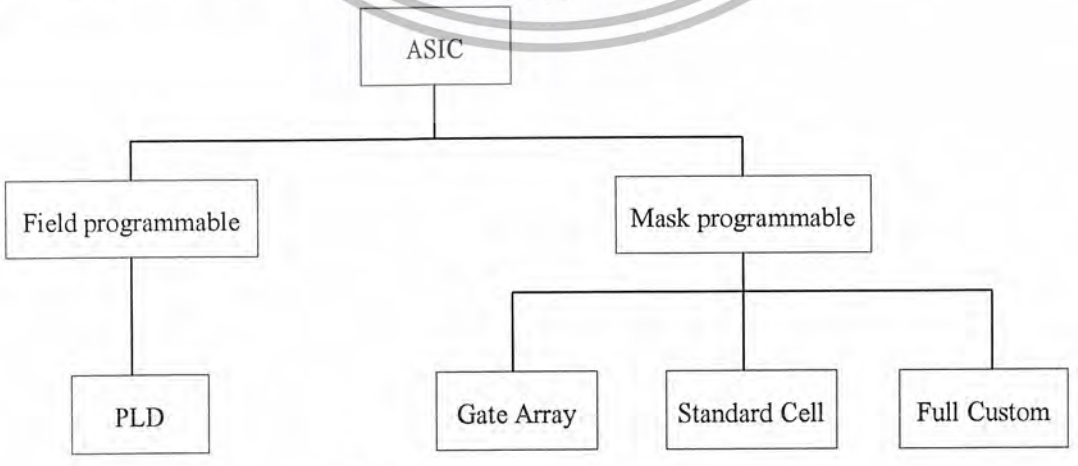
5) การออกแบบทางกายภาพและการวิเคราะห์ คือ ขั้นตอนการผลิตเป็นวงจรจริง (Technology and device mapping) โดยนำข้อมูลที่ได้จากการสังเคราะห์มาผลิต ซึ่งอาจ จะอยู่ในรูปของแผงวงจรไฟฟ้าที่ประกอบด้วยอุปกรณ์หลายๆ ชิ้นหรืออยู่ในรูปของวงจรรวมเอซิก (ASIC)

6) การตรวจสอบเวลาหลังการออกแบบ คือ การทำการตรวจสอบการทำงานด้วยตัวแปรทางด้านเวลาทั้งหมด เพื่อความถูกต้องของวงจรเป็นครั้งสุดท้ายก่อนนำไปรวมเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบดิจิตอล เนื่องจากในขั้นตอนนี้ วงจรที่ออกแบบ จะประกอบด้วยจุดต่อทางอินพุทและเอาท์พุท ซึ่งเป็นจุดต่อสำหรับการรับและส่งสัญญาณกับภายนอก

7) การตรวจสอบระบบ คือ การนำวงจรที่ออกแบบไว้ประกอบเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบที่สมบูรณ์ แล้วทำการทดสอบการทำงานทั้งระบบรวมกับอุปกรณ์อื่นๆ อีกครั้งเพื่อควบคุมคุณภาพของผลิตภัณฑ์

3.5 เอฟพีจีเอ

เทคโนโลยีความก้าวหน้าของอุตสาหกรรมอิเล็กทรอนิกส์ในปัจจุบัน ทำให้เกิดการพัฒนาความสามารถของอุปกรณ์ต่างๆ ซึ่งทำให้ลดค่าใช้จ่ายต่างๆ ได้มาก ในขณะเดียวกันก็มีการเพิ่มประสิทธิภาพและระดับความเชื่อถือได้ของวงจรรวมที่สูงขึ้นเห็นได้ชัดจากเทคโนโลยีไมโครโปรเซสเซอร์ และหน่วยความจำปัจจุบัน ทุกๆครั้งที่มีการพัฒนาขึ้นทำให้เกิดช่องว่างระหว่างวงจรรวมและไอซีมาตรฐานมากขึ้น นักออกแบบอุปกรณ์ทางด้านดิจิตอล ได้พิจารณาถึงการผลิตให้มีขนาดมากขึ้นและการผลิตวงจรรวมเอซิก (ASIC: Application Specific Integrated Circuit) ซึ่งวงจรรวมจะแบ่งตามสร้างออกเป็น 2 กลุ่ม คือ ฟิลด์โปรแกรมเมเบิล (Field programmable) และ แมสโปรแกรมเมเบิล (Mask programmable) ดังแสดงในรูปที่ 3.22



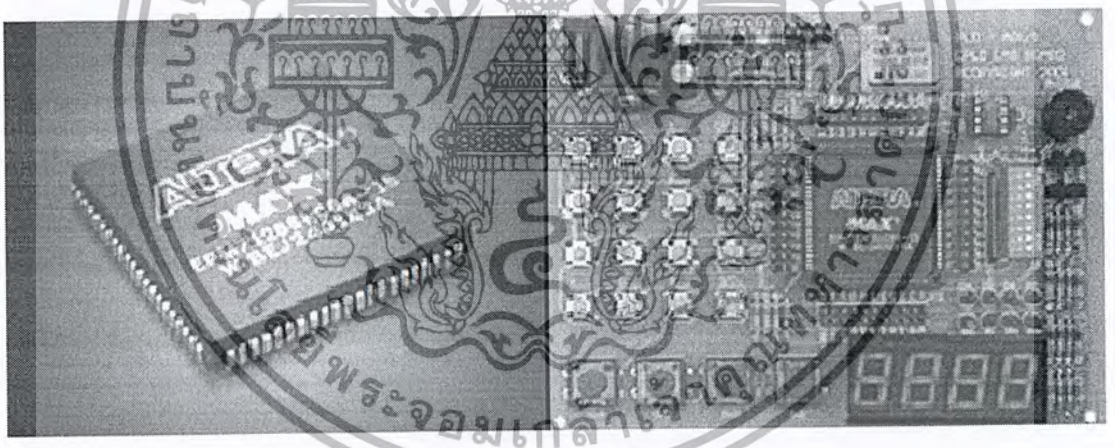
รูปที่ 3.22 ผังแสดงการแบ่งกลุ่มของวงจรรวมเอซิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.1 การออกแบบวงจรเชิงเลขด้วย ชิพอุปกรณ์เอฟพีจีเอ

ชิพอุปกรณ์เอฟพีจีเอ เป็นอุปกรณ์ที่ใช้ในการโปรแกรมวงจรที่ได้ออกแบบลงไปเพื่อให้อุปกรณ์เอฟพีจีเอ มีฟังก์ชันการทำงานตามที่ออกแบบไว้ ในการทำชิพอุปกรณ์ ซึ่งเป็นวิธีการออกแบบไอซี(IC : Integrated Circuit) แบบ เซมิคัสตัม (Semi custom) อีกวิธีหนึ่ง เมื่อเทียบกับการทำเอซิก แล้วนั้นก็มีทั้งข้อดีและข้อเสีย คือ การทำชิพอุปกรณ์เอฟพีจีเอ จะมีข้อจำกัดในด้านขนาดของวงจรเพราะภายในชิพอุปกรณ์เอฟพีจีเอ จะมีจำนวนเกท (g a t e) ให้ใช้จำนวนจำกัด และการทำชิพอุปกรณ์เอฟพีจีเอก็เหมาะสำหรับการทำผลิตภัณฑ์ต้นแบบหรือเพื่อผลิตในปริมาณต่ำ ส่วนข้อดีของการทำชิพอุปกรณ์ก็คือระยะเวลาที่ใช้ในการทำตั้งแต่เขียนรหัส (code) อธิบายฮาร์ดแวร์จนกระทั่งดาวน์โหลด (download) นั้นน้อยกว่าการทำเอซิกมากและการตรวจสอบหรือแก้ไขการออกแบบที่ทำได้สะดวก

การทำชิพอุปกรณ์เอฟพีจีเอ ในปัจจุบันมีประสิทธิภาพและความสะดวกมากขึ้น ทั้งนี้ก็เนื่องจากทางบริษัทผู้ผลิตชิพอุปกรณ์เอฟพีจีเอ ได้เพิ่มความสามารถของชิพอุปกรณ์เอฟพีจีเอ โดยเพิ่มจำนวนองค์ประกอบภายใน หรือปรับปรุงโครงสร้างสถาปัตยกรรมภายในและยังได้เพิ่มประสิทธิภาพของซอฟต์แวร์ที่ใช้ทำ พีพีอาร์ (PPR: Partitioning Placement and Routing) สำหรับอุปกรณ์นั้นๆด้วย ลักษณะของตัว ชิพอุปกรณ์เอฟพีจีเอ และการนำไปใช้งานแสดงดังในรูปที่ 3.23



รูปที่ 3.23 ลักษณะของตัว FPGA และการนำไปใช้งาน

สำหรับตัวชิพอุปกรณ์เอฟพีจีเอนั้นมีโครงสร้างพื้นฐาน เทคโนโลยีที่ใช้สร้าง ตลอดจนเทคนิควิธีการโปรกรมที่แตกต่างกันสำหรับผู้ผลิตแต่ละราย นอกจากนั้นชิพอุปกรณ์เอฟพีจีเอ ของแต่ละผู้ผลิตก็มีโครงสร้างและความสามารถที่แตกต่างกันบางส่วน ในการใช้งานนั้นชิพอุปกรณ์เอฟพีจีเอ สามารถนำไปประยุกต์ใช้งานได้ เช่น การประมวลผลสัญญาณเชิงเลข (DSP: Digital Signal Processing) การออกแบบไมโครคอนโทรลเลอร์ เป็นต้น

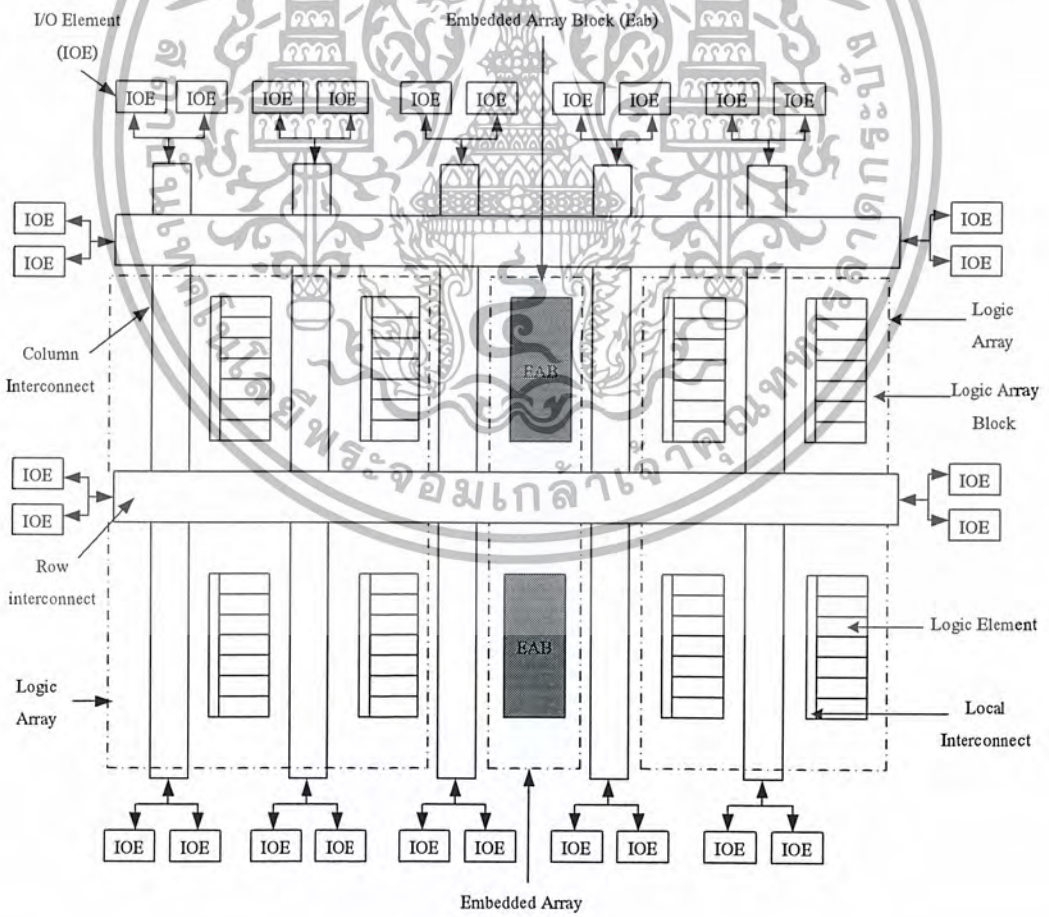
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 สถาปัตยกรรมภายในของชิพอุปกรณ์เอฟพีจีเอ

ชิพอุปกรณ์เอฟพีจีเอของบริษัท Altera ตระกูล FLEX 10 K เป็นอุปกรณ์ที่มีความหนาแน่นเกท ประมาณตั้งแต่ 25,000 – 100,000 เกท โดยการจัดโครงสร้าง (Configuration) จะใช้วิธีโหลดโครงสร้างเข้าไปใน SRAM ภายใน ซึ่งหมายความว่าไม่ได้มีการจ่ายไฟเลี้ยงให้ โครงสร้างที่จัดเอาไว้ก็จะหายไป ชิพอุปกรณ์เอฟพีจีเอ ประเภทนี้จะสามารถโปรแกรมซ้ำได้ไม่จำกัดจำนวนครั้ง และการทำงานของลอจิกฟังก์ชั่นจะใช้วิธีการเปิดตารางความจริงดู (Look Up table : LUT) โดยโครงสร้างของชิพอุปกรณ์เอฟพีจีเอตระกูล FLEX 10 K แสดงดังรูปที่ 3.24 โดยในโครงสร้างของชิพอุปกรณ์เอฟพีจีเอตระกูล FLEX 10 K สามารถที่จะแบ่งเป็นส่วนต่างๆ ได้ดังนี้

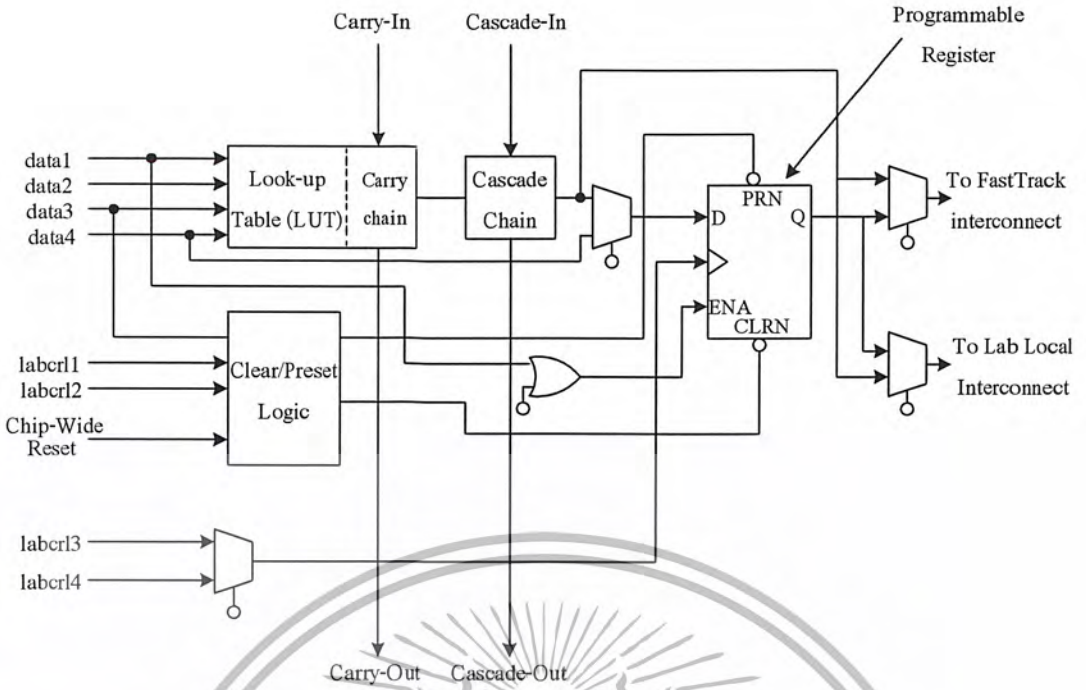
3.5.2.1 แอลอี (LE : Logic Element)

ในรูปที่ 3.25 แสดงโครงสร้างภายในของแอลอี โดยการกระทำทางบูลีนของลอจิกเกทจะสร้างด้วยวิธีการ LUT โดย LUT คือ 1x16 SRAM ซึ่ง LUT เพียงตัวเดียวสามารถนำมาทำโครงข่ายของลอจิกเกทที่มี 4 อินพุท และ 1 เอาท์พุท โดยโครงข่ายของลอจิกเกทจะถูกแปลงไปเป็นตารางค่าความจริง (Truth Table) ดังแสดงในรูปที่ 3.26

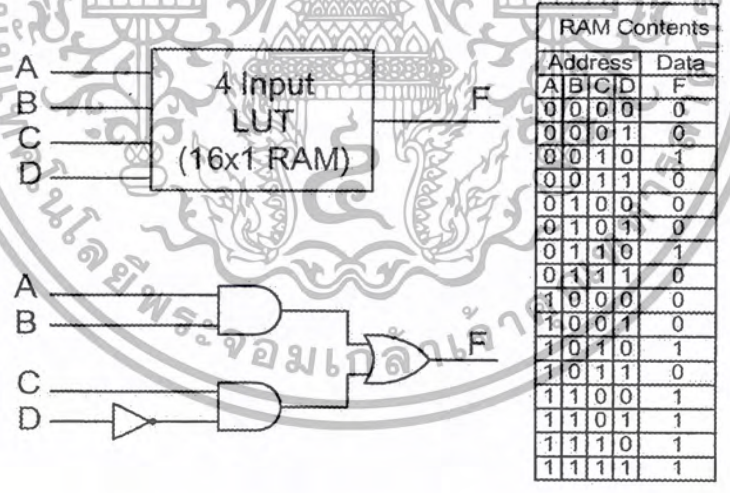


รูปที่ 3.24 โครงสร้างของชิพอุปกรณ์เอฟพีจีเอ ตระกูล FLEX 10 K

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



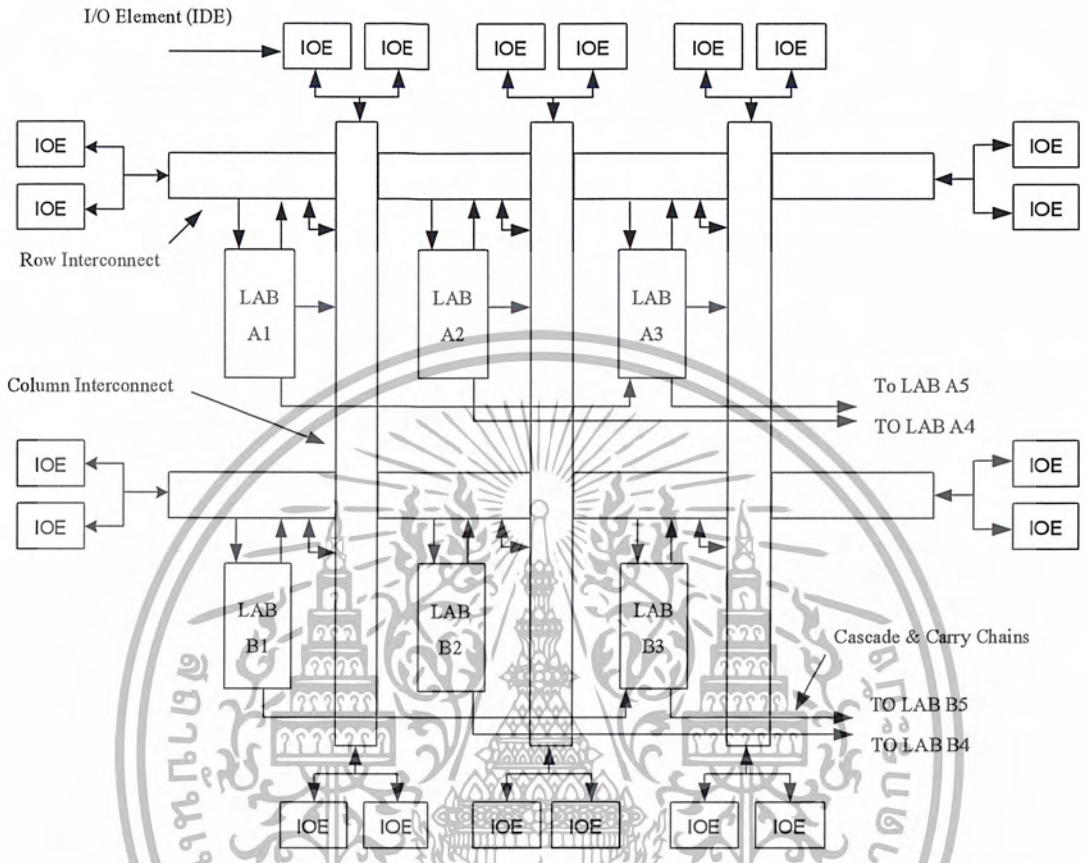
รูปที่ 3.25 โครงสร้างภายในของแอลอี



รูปที่ 3.26 การใช้งาน LUT เป็นโครงข่ายของลอจิก

ถ้าโครงข่ายของลอจิกเกทมีความซับซ้อนขึ้นจะต้องใช้ LUT ของแต่ละ LE เป็นจำนวนหลายตัว โดยเอาท์พุทของ LUT จะส่งต่อไปยังฟลิปฟลอปและต่อไปยังโครงข่ายการเชื่อมต่อ (Interconnection Network) ดังแสดงในรูปที่ 3.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

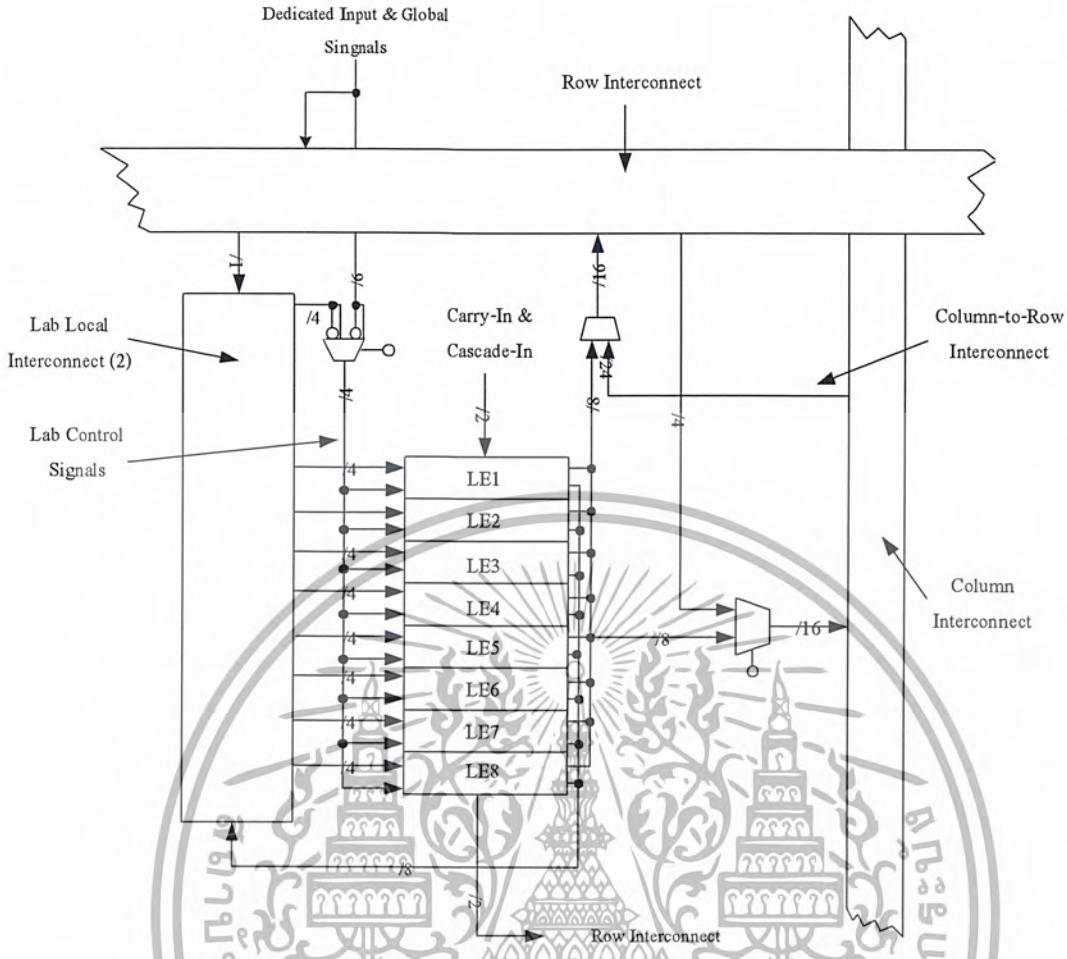


รูปที่ 3.27 โครงข่ายของการเชื่อมต่อ

3.5.2.2 แอลเอบี (LAB: Logic Array Block)

แอลเอบี 1 ตัว จะประกอบไปด้วย 8 แอลอี ดังแสดงในรูปที่ 3.28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

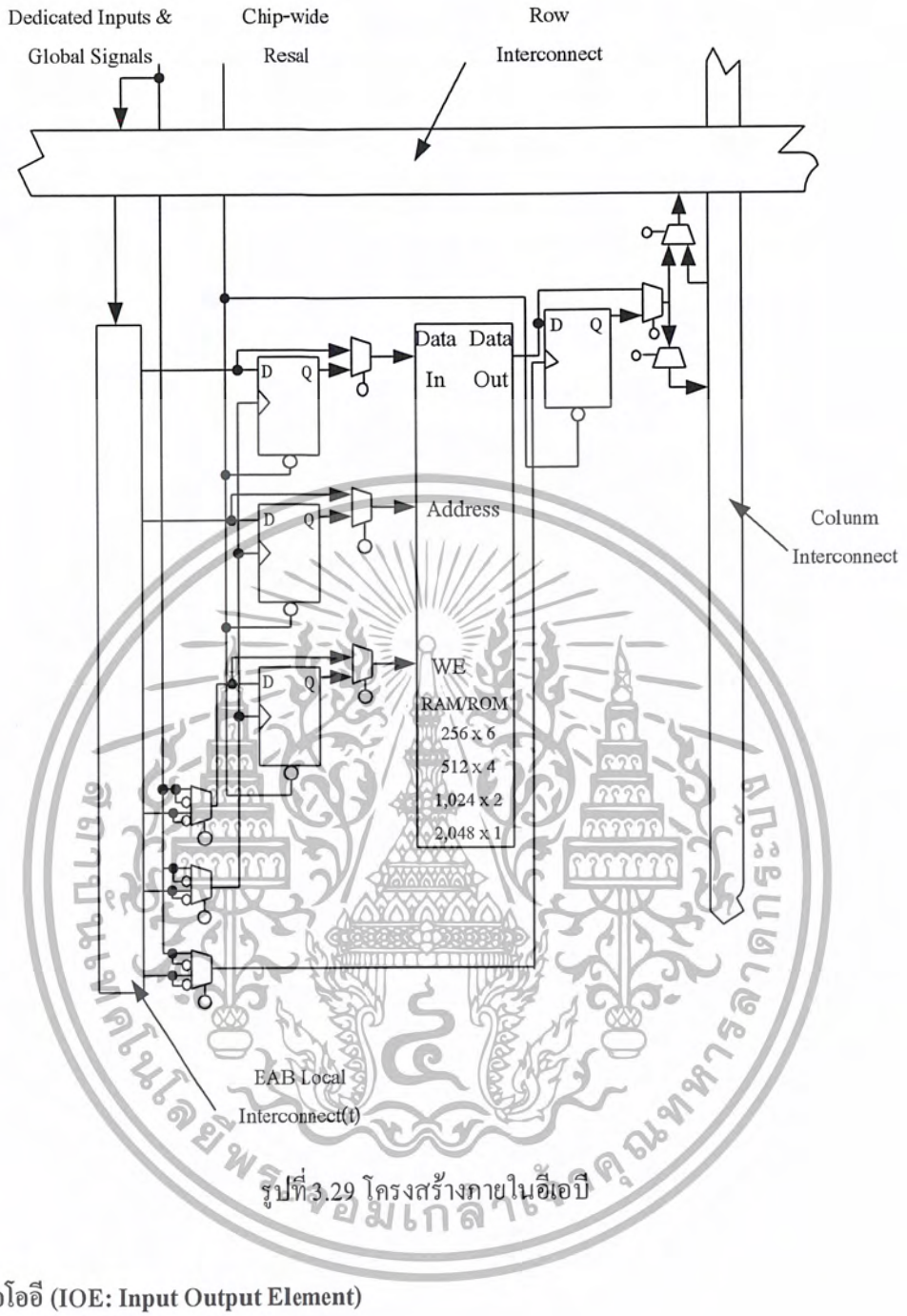


รูปที่ 3.28 โครงสร้างภายในของ แอลเอบี

3.5.2.3 อีเอบี (EAB: Embedded Array Block)

สถาปัตยกรรมโดยทั่วไปของ FLEX 10 K จะมีลักษณะของแอลเอบี ที่มีการจัดเรียงแบบเมตริกซ์ และ อีเอบี ซึ่งมีการเชื่อมต่อผ่านทางแถวและคอลัมน์ โดยในแต่ละแถวจะมี 1 อีเอบี จะมีขนาด 2048 บิท และสามารถกำหนดความกว้าง (Width) ความลึก (Depth) ของ อีเอบี ได้โดยไม่ส่งผลต่อความเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

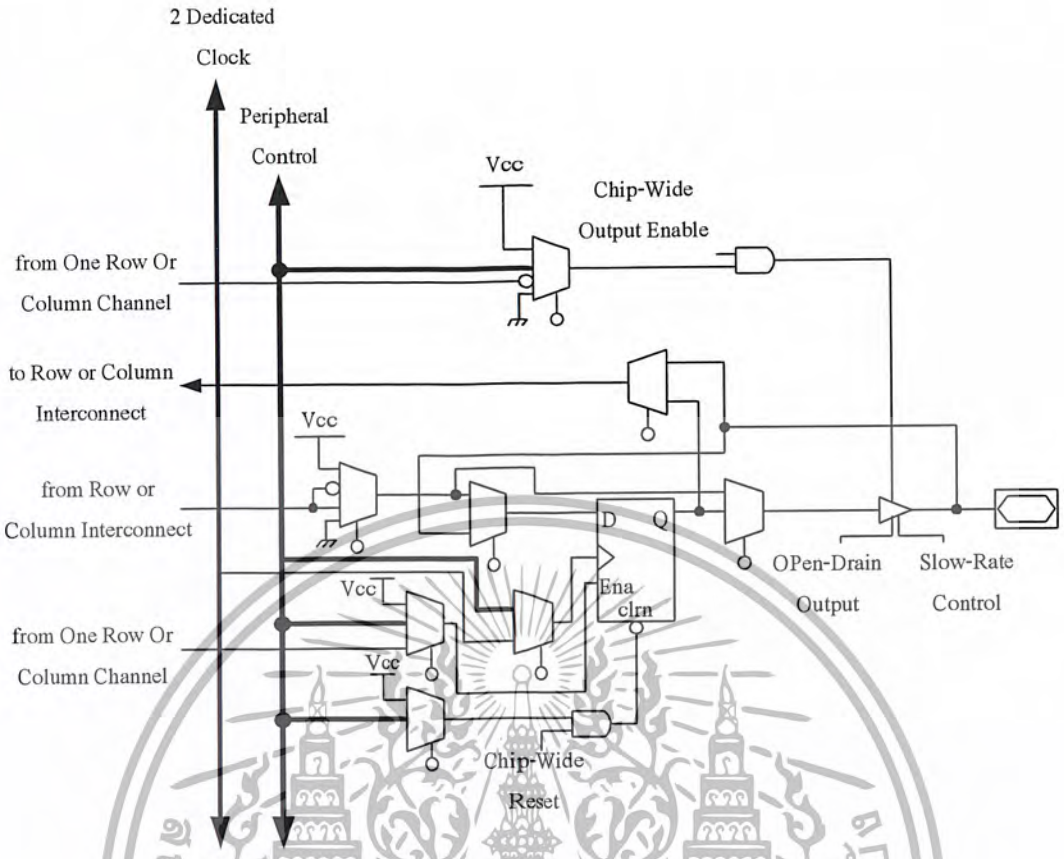


รูปที่ 3.29 โครงสร้างภายในอีเอบี

3.5.2.4 ไอโออี (IOE: Input Output Element)

ไอโออี จะถูกต่ออยู่กับขา I/O โดยจะประกอบด้วยส่วนของวงจรที่เป็น Tri State และส่วนที่เป็นฟลิปฟลอป ซึ่งเป็น option ดังแสดงในรูปที่ 3.30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.30 โครงสร้างภายในของไอโออี

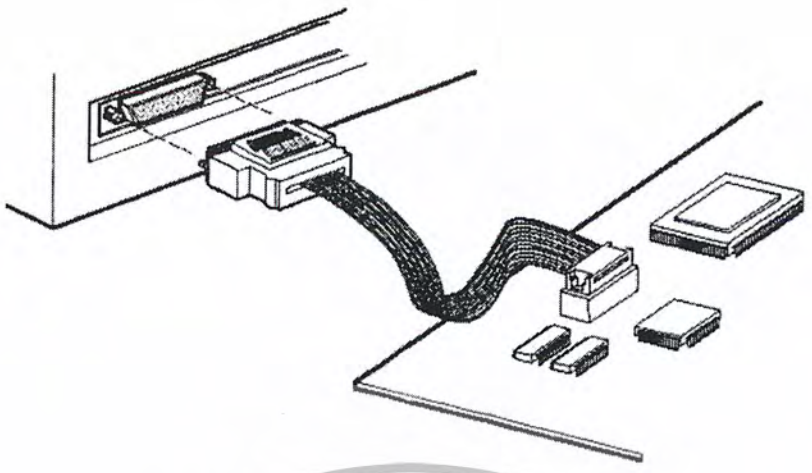
3.6 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว

1. ผู้ออกแบบไม่จำเป็นต้องทราบถึงโครงสร้างภายในของตัวชิพเพียงแต่มีความรู้เกี่ยวกับขั้นตอนการออกแบบลอจิกก็เพียงพอแล้ว ต่างกับการใช้ไมโครโปรเซสเซอร์ซึ่งจำเป็นต้องศึกษาโครงสร้างภายในรวมถึงภาษาแอสแซมบี้ (Assembly) ของไมโครโปรเซสเซอร์ตัวนั้นด้วย

2. มีการออกแบบโดยใช้ภาษาในการอธิบายการทำงานของวงจรหรือเอชดีแอล เป็นเครื่องมือในการออกแบบ ซึ่งเป็นวิธีการที่มีความยืดหยุ่นสูง ทำได้รวดเร็ว และไม่จำเป็นต้องทราบถึงลักษณะของวงจรที่ต้องการว่าจะเชื่อมต่อกันอย่างไร เพียงแต่กำหนดลักษณะการทำงานให้มันจากนั้นตัวซอฟต์แวร์จะทำการสังเคราะห์ ให้ทั้งหมด นอกจากนี้ภาษาที่ใช้ยังเป็นมาตรฐานเดียวกันสามารถใช้ได้กับชิพทุกตัวและทุกบริษัท

3. การโปรแกรมสามารถทำได้เองและใช้เวลาไม่นาน เพียงแค่ส่งข้อมูลผ่านสายดาวน์โหลดทางพอร์ตของคอมพิวเตอร์ก็สามารถโปรแกรมตัวชิพขณะที่อยู่ในระบบได้โดยไม่จำเป็นต้องถอดมาโปรแกรมข้างนอก ดังรูปที่ 3.36 และที่สำคัญสามารถโปรแกรมได้หลายครั้ง จึงทำให้ง่ายในการแก้ไขและพัฒนาโดยไม่ต้องเสียค่าใช้จ่ายเพิ่มแต่อย่างใด

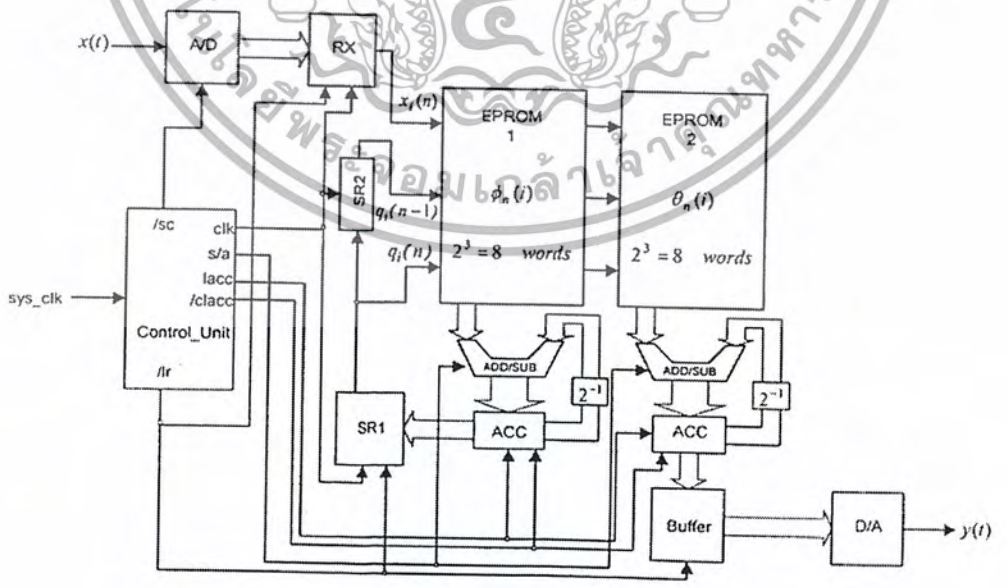
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.31 การโปรแกรมลงในชิพอุปกรณ์เอฟพีจีเอ

3.7 ขั้นตอนในการออกแบบและทดสอบการทำงานของวงจรภายใน

ในการออกแบบโครงสร้างของวงจรกรองสัญญาณเชิงเลข ที่ใช้โครงสร้างเลขคณิตกระจาย โดยการแทนด้วยปริภูมิสเตทที่นำเสนอนั้น จะทำการพิจารณาจากวงจรกรองในอันดับที่ 2 เนื่องจากในวงจรที่มีอันดับสูงขึ้นก็จะมีส่วนประกอบที่ใช้งานที่เหมือนกัน ซึ่งโครงสร้างของวงจรกรองสัญญาณเชิงเลขอันดับที่ 2 จากการแทนด้วยปริภูมิสเตท แสดงดังรูปที่ 3.32



รูปที่ 3.32 โครงสร้างของวงจรกรองสัญญาณเชิงเลขอันดับที่ 2 จากการแทนด้วยปริภูมิสเตท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.32 จะแยกอธิบายขั้นตอนของการทำงานเป็น 4 ขั้นตอน ดังนี้

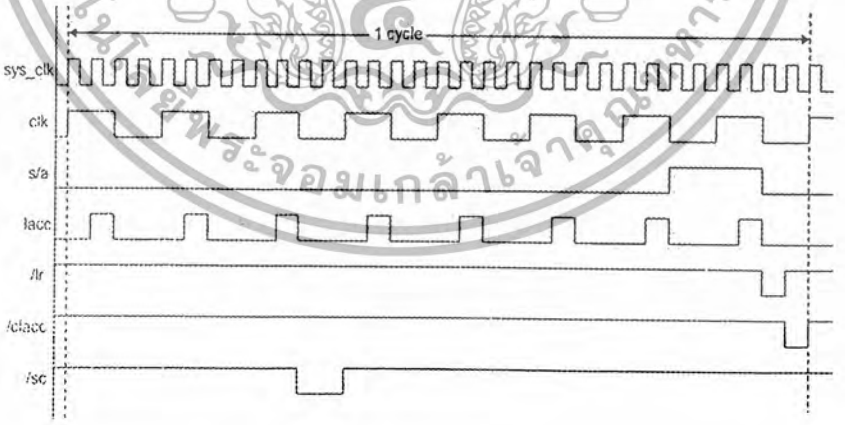
วงจร A/D (Analog to Digital Converter) ซึ่งถูกควบคุมด้วยสัญญาณ /sc จะทำการแปลงสัญญาณเชิงอุปมาน(analog) $x(t)$ ให้เป็นลำดับสัญญาณเชิงเลข(digital) $x(n)$ ขนาด 8 บิท

1. สัญญาณควบคุม/lr ทำหน้าที่โหลดลำดับสัญญาณหรือข้อมูล $x(n)$ เข้าไปกับไว้ในรีจิสเตอร์ RX

2. สัญญาณนาฬิกา clk จะทำการเลื่อนข้อมูลภายในรีจิสเตอร์ RX , SR1 และ SR2 ไปครั้งละ 1 บิทเอาท์พุทของรีจิสเตอร์ RX , SR1 และ SR2 ที่ถูกเลื่อนแต่ละครั้งจะเป็นแอดเดรสของ EPROM ทั้ง 2 ตัว เอาท์พุทของ EPROM ทั้ง 2 ตัว จะถูกส่งไปบวกกับค่าที่อยู่ใน ACC ด้วยวงจรADD/SUB (ซึ่งถูกควบคุมด้วยสัญญาณ s/a) ผลลัพธ์ที่ได้จะถูกโหลดเข้าเก็บไว้ใน ACC ด้วยสัญญาณ lacc (การคูณค่าที่อยู่ใน ACC ด้วย 2^{-1} หรือเป็นการเลื่อนข้อมูลไปทางขวา 1 บิท ก่อนที่จะนำไปบวกกับค่าจาก EPROM ถูกออกแบบในลักษณะฮาร์ดแวร์สเกลลิ่ง (Hardware scaling) จึงไม่จำเป็นที่จะต้องมีวงจรเลื่อนข้อมูลไปทางขวา 1 บิท)

3. clk จะเลื่อนข้อมูลในแต่ละรีจิสเตอร์ไปอีก 1 บิท แล้วทำซ้ำข้อ 3 จนกระทั่ง clk เลื่อนข้อมูลไปถึงบิทที่ 8 จึงนำค่าที่ได้จากเอาท์พุทของ EPROM ทั้ง 2 ตัว ไปลบออกจากค่าที่อยู่ใน ACCผลลัพธ์ที่ได้จากการคำนวณของ EPROM1 จะถูกโหลดเข้าไปเก็บไว้ในรีจิสเตอร์ SR1 ส่วนผลลัพธ์ที่ได้จากการคำนวณของ EPROM2 จะถูกโหลดเข้าไปเก็บไว้ใน buffer (ด้วยสัญญาณ /lr) เพื่อทำการแปลงสัญญาณเชิงเลขให้เป็นสัญญาณเชิงอุปมาน ด้วยวงจร D/A จากนั้นทำการลบข้อมูลภายใน ACC ด้วยสัญญาณ/clacc และจะวนกลับไปทำงานซ้ำในขั้นตอนที่ 1 , 2 , 3 , 4 ตามลำดับ

ส่วนสัญญาณที่ใช้ในหารควบคุมการทำงานสามารถแสดงไทม์มิ่งไดอะแกรม(Timing diagram) ได้ดังนี้



รูปที่ 3.33 ไทม์มิ่งไดอะแกรมของสัญญาณควบคุม

โดยมีขั้นตอนในการออกแบบส่วนต่าง ๆ ดังนี้

1. รีจิสเตอร์ RX, SRI ซึ่งเป็น PISO (Parallel in Serial out shift register) และรีจิสเตอร์ SR2 ซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยมีขั้นตอนในการออกแบบส่วนต่าง ๆ ดังนี้

1. รีจิสเตอร์ RX, SRI ซึ่งเป็น PISO (Parallel in Serial out shift register) และรีจิสเตอร์ SR2 ซึ่งเป็น SISO (Serial in Serial out shift register) โดยในส่วนของรีจิสเตอร์นี้ถูกออกแบบไว้สำหรับเลื่อนข้อมูลหรือสัญญาณในรีจิสเตอร์ขนาด 8 บิท ซึ่งผลที่ได้จากการเลื่อนข้อมูลแต่ละบิทจะเป็นตัวกำหนดแอดเดรสของ EPROM และรีจิสเตอร์ Buffer ซึ่งใช้ในการเก็บผลลัพธ์ที่ได้จากการคำนวณไว้เพื่อรอการแปลงเป็นสัญญาณเชิงอุปมาน

2. หน่วยความจำ EPROM1 และ EPROM2 สำหรับเก็บค่าผลคูณย่อยไว้ในตารางเปิดดู ซึ่งเมื่อเข้าใจหลักการทำงานของโครงสร้างเลขคณิตกระจาย ก็จะทราบว่าจะต้องใช้ขนาดของหน่วยความจำเท่าใดกรณีของวงจรกรองในอันดับที่ 2 แต่ละหน่วยความจำจะมีขนาดเท่ากับ 8 x 8 = 64 บิท (กรณีใช้ความยาวคำของค่าในหน่วยความจำขนาด 8 บิท) โดย EPROM1 จะเก็บค่าที่ใช้คำนวณสมการสเตท และ EPROM2 เก็บค่าที่ได้ในการคำนวณสมการเอาท์พุท ซึ่งทำการเขียนเป็นหน่วยการออกแบบแพ็กเก็ต (Package Desing unit) เพื่อเก็บค่าในตารางเปิดดู $\phi_n(i)$ และ $\theta_n(i)$ ไว้ในหน่วยการออกแบบนี้ ค่าในตารางเปิดดูนี้สามารถเรียกไปใช้ได้โดยหน่วยการออกแบบเอนทิตี้ (Entity Dising unit) และหน่วยการออกแบบสถาปัตยกรรม (Architecture Design unit)

3. วงจรสเกลลิ่งแอคคิวมูเลเตอร์ (Scaling Accumulator) โดยอาศัยคุณสมบัติการบวกลบแบบเลขส่วนเติมเต็มสองแทนการคูณโดยตรง ซึ่งประกอบด้วยวงจร บวก/ลบ (ADD/SUB) และรีจิสเตอร์แอคคิวมูเลเตอร์ เอาท์พุทของรีจิสเตอร์แอคคิวมูเลเตอร์ถูกออกแบบให้เป็นในลักษณะฮาร์ดแวร์สเกลลิ่ง (Hardware Scaling) ด้วย 2^{-1} ก่อนที่จะป้อนกลับไปบวกกับค่าที่ออกมาจากหน่วยความจำตัวต่อไป และเพื่อเป็นการหลีกเลี่ยงการเลื่อนบิทผิดพลาดไปยังตำแหน่งของบิทเครื่องหมายในกรณีที่ตัวบวกเกิดการล้น (Overflow) ดังนั้นจึงต้องมีวงจรเพิ่มเติมสำหรับการตรวจสอบบิทเครื่องหมาย (Sing Digit) โดยใช้เอ็กคลูซีฟออร์เกทนั้น คือ จะเอาบิทเครื่องหมายของข้อมูลที่เข้าวงจรบวก (a^0 และ b^0) และ บทตัวทด (c^0) มาทำเอ็กคลูซีฟออร์กัน โดยจากวงจรบวก ถ้า ดังนั้นบิทเครื่องหมายของผลลัพธ์จากการเลื่อนข้อมูล (d) ไปทางขวา 1 บิท (2^{-1} d) หาได้จากสมการ $d^0 = a^0 \oplus b^0 \oplus c^0$

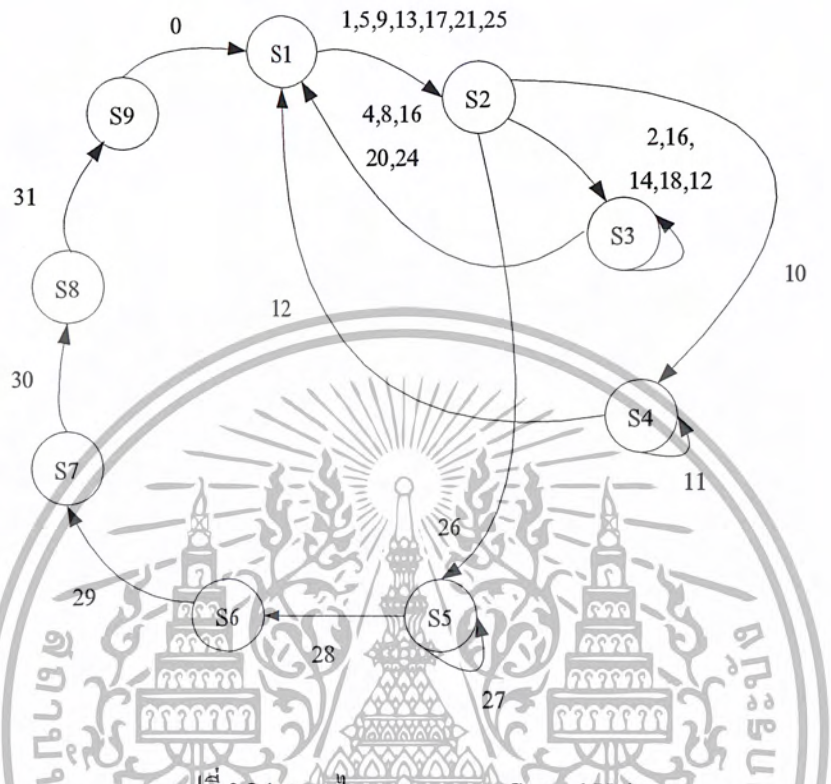
วงจรสเกลลิ่งแอคคิวมูเลเตอร์ จะมีวงจร SD ใช้เพื่อป้องกันกรณีที่เกิดการล้นของสัญญาณไปทับบิทเครื่องหมาย และแสดงถึงลักษณะการเชื่อมโยงของสัญญาณในลักษณะของฮาร์ดแวร์สเกลลิ่งด้วย 2^{-1} หรือเลื่อนข้อมูลไปทางขวา 1 บิท ซึ่งวงจร SD เสมือนเป็นวงจรที่ใช้เติมบิทเครื่องหมายหลังจากการเลื่อนข้อมูลไปทางขวา โดยแสดงเป็นตัวอย่างการทำงานของวงจรเมื่อเปรียบเทียบกับการคำนวณโดยเลขฐานสิบ

ในกรณีค่าที่จะเก็บไว้ในตารางเปิดดูมีค่าเกิน 1 จะต้องทำการหารให้มีค่าไม่เกินหนึ่ง แต่ก่อนที่จะโหลดเอาท์พุทออกจะต้องมีการคูณกลับก่อน เช่น กรณีที่ค่าในตารางเปิดดูถูกหารด้วย 2 ดังนั้นเวลาโหลดเอาท์พุทออกจะต้องคูณ 2 กลับคืน ซึ่งสามารถแสดงวงจรสเกลลิ่งแอคคิวมูเลเตอร์ได้ดังนี้

วงจรจรควบคุม Control Unit มีไว้เพื่อควบคุมส่วนต่าง ๆ ภายในวงจรกรองสัญญาณเชิงเลขให้ทำงานสอดคล้องกันเพื่อที่จะสร้างเอาท์พุทที่ถูกต้องตามต้องการวงจรควบคุม โดยทั่วไปจะผลิตสัญญาณนาฬิกาให้สอดคล้องกับการทำงานและคงสภาพของระดับสัญญาณนั้น ๆ ไว้ตามช่วงเวลาที่เหมาะสมกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไดอะแกรมที่กำหนดไว้เพื่อผลิตสัญญาณควบคุม clk , /lr , /sc , lacc/clacc , s/a โดยมีสเตทไดอะแกรม (State diagram) ดังนี้



รูปที่ 3.34 สเตทไดอะแกรมของ Control Unit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

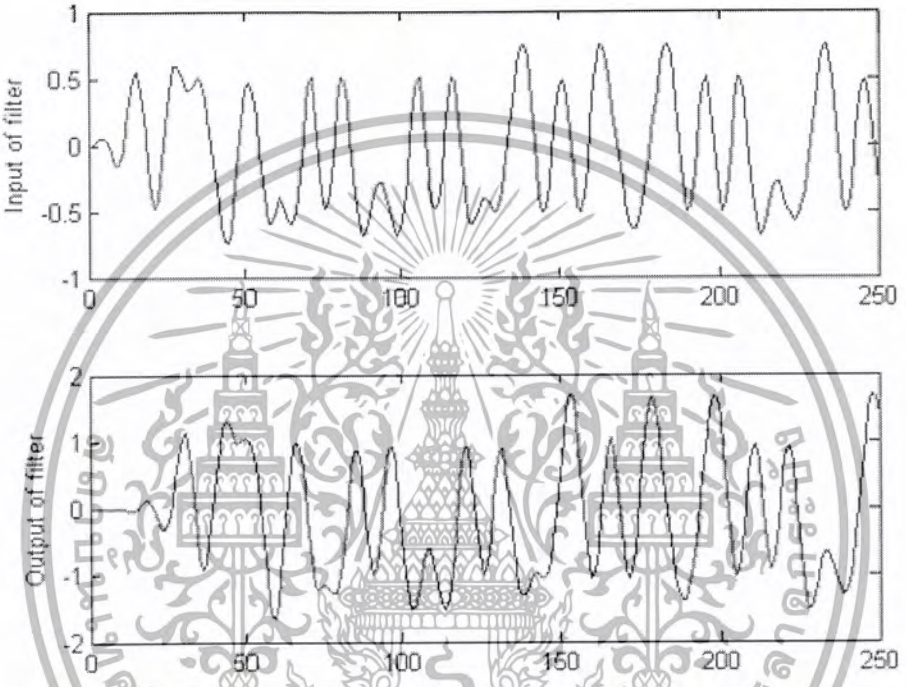
การทดลองและผลการทดลอง

4.1 ขั้นตอนในการซิมมูเลตด้วยโปรแกรมแมทแลบ

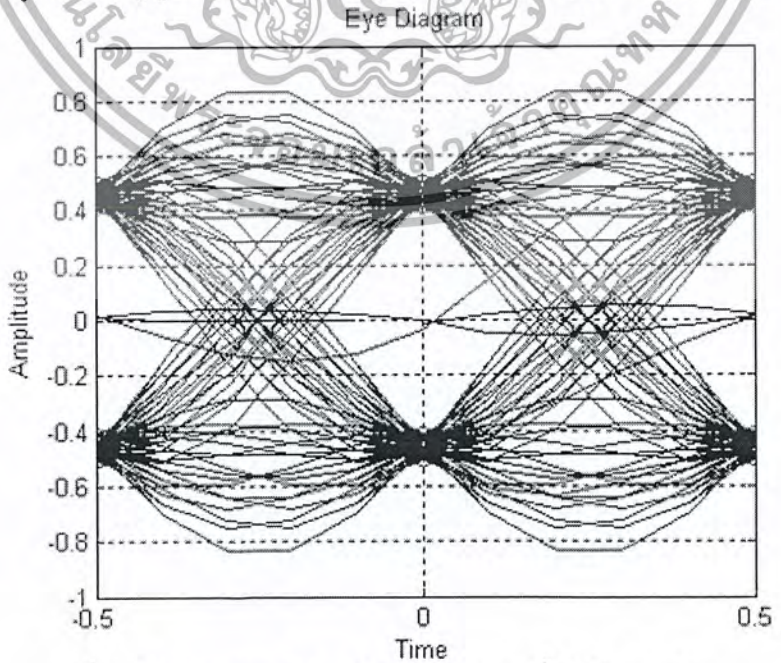
4.1.1 ป้อนสัญญาณ Root-raised cosine ให้กับตัวกรอง

กำหนดให้ตัวกรองมีค่าโรลออฟแฟคเตอร์เท่ากับ 0.3 และทำการเปลี่ยนแปลงค่าโรลออฟแฟคเตอร์ของสัญญาณอินพุท เพื่อสังเกตเอาท์พุทจากตัวกรอง

- อินพุทมี โรลออฟ = 0.1

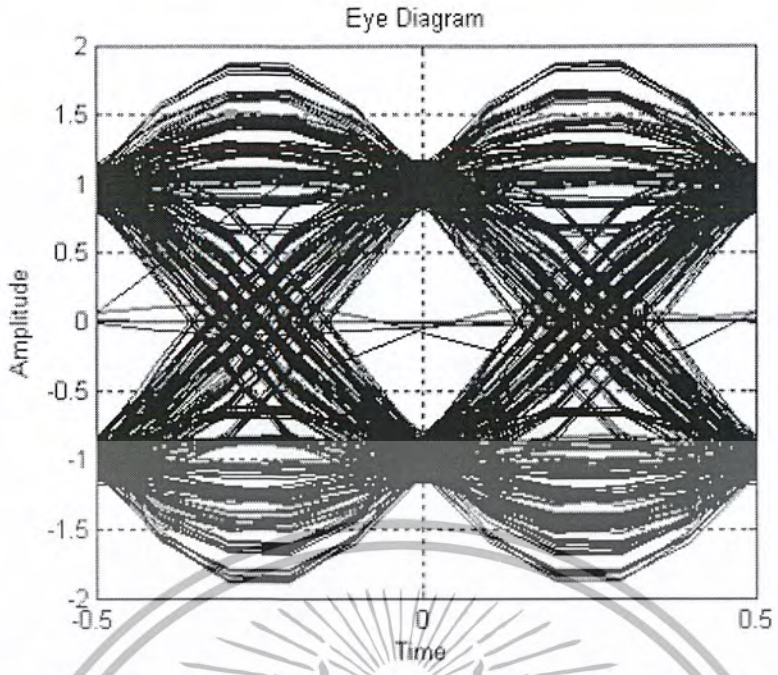


รูปที่ 4.1 สัญญาณอินพุทและเอาท์พุทของตัวกรอง ที่มีค่าโรลออฟ = 0.1



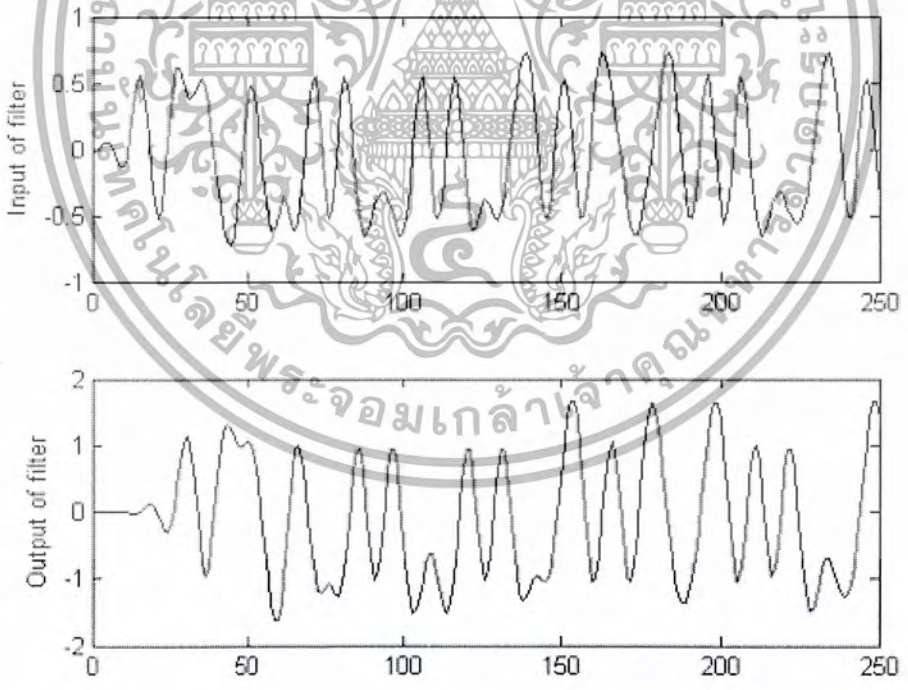
รูปที่ 4.2 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



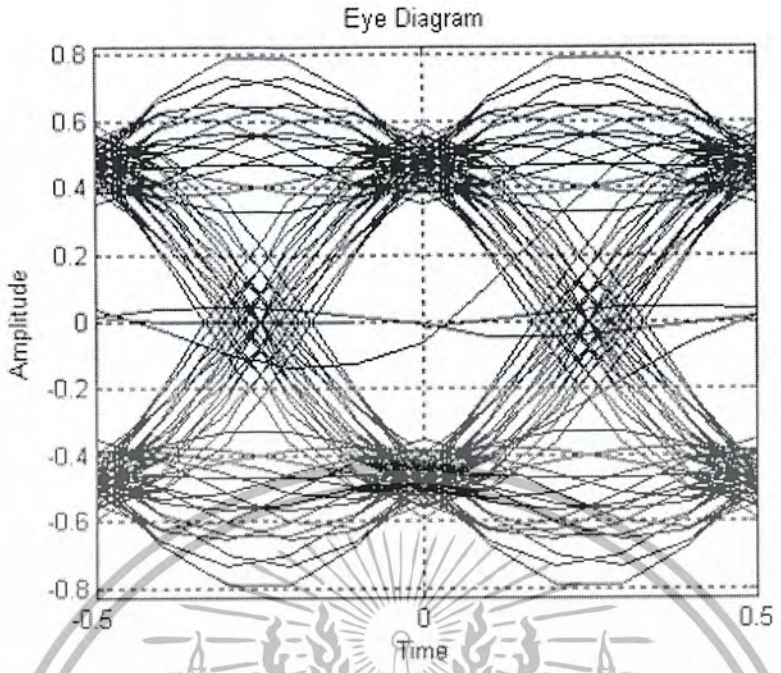
รูปที่ 4.3 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท ที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.1

- อินพุทมี โรลออฟ = 0.2

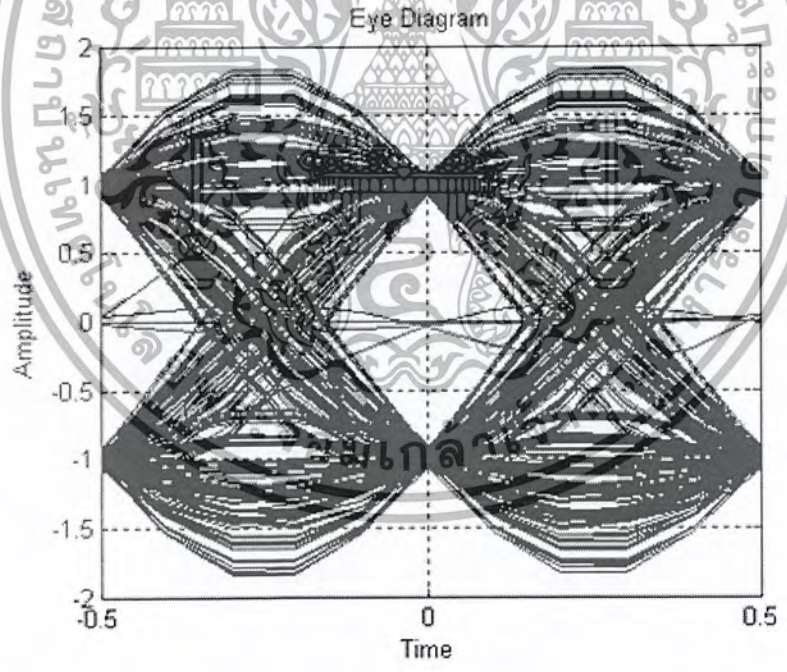


รูปที่ 4.4 สัญญาณอินพุทและเอาท์พุทของตัวกรอง ที่มีค่าโรลออฟ = 0.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



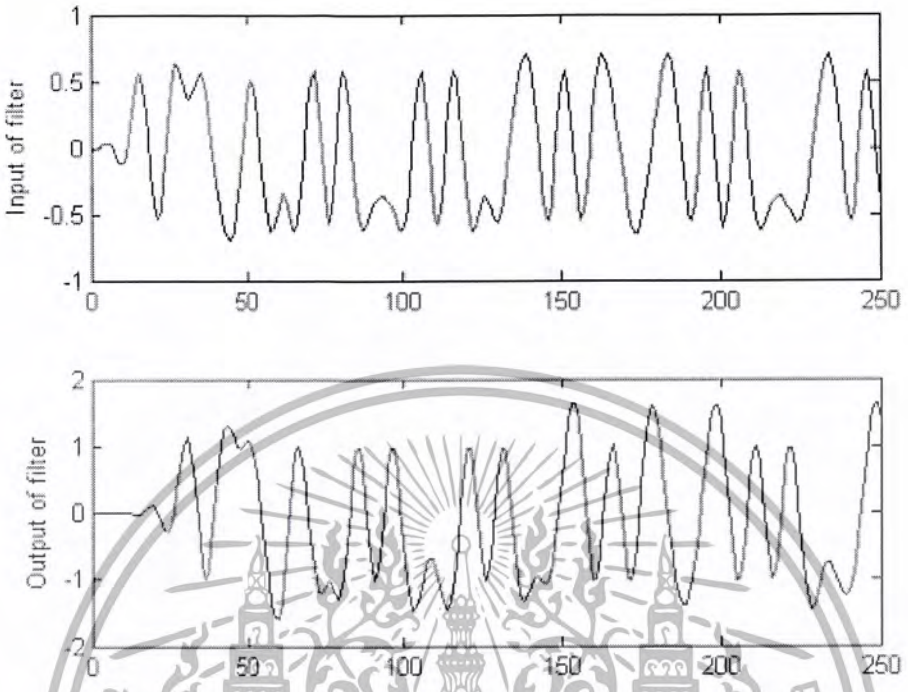
รูปที่ 4.5 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.2



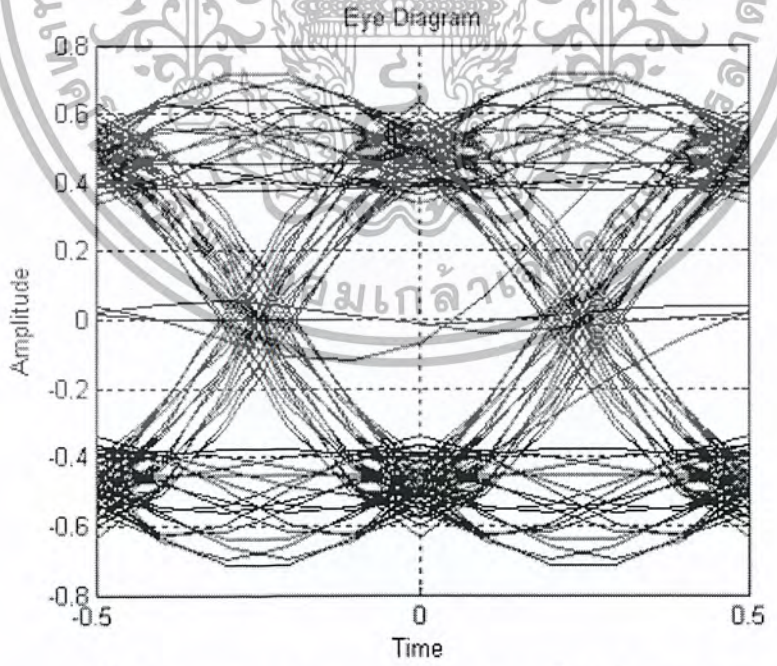
รูปที่ 4.6 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท ที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อินพุทมี โรลออฟ = 0.3

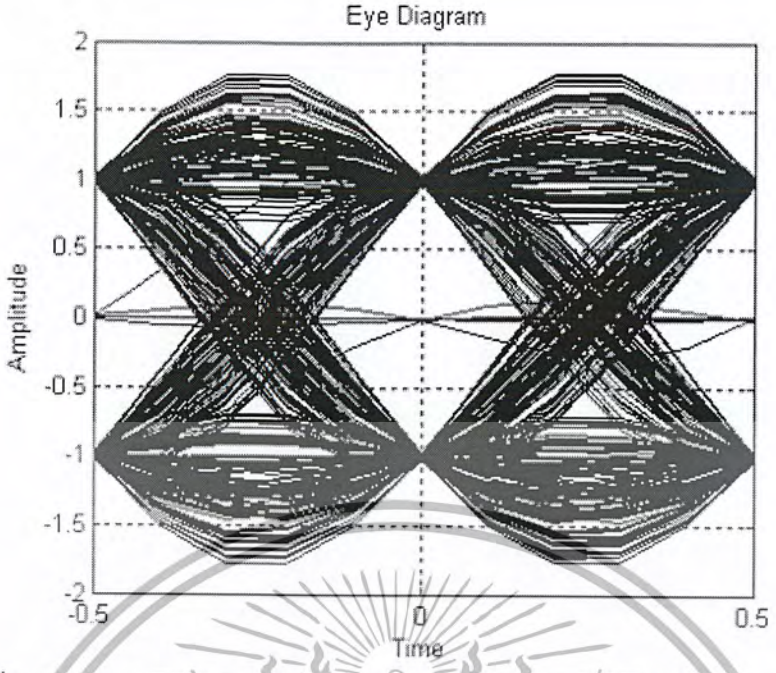


รูปที่ 4.7 สัญญาณอินพุทและเอาท์พุทของตัวกรอง ที่มีค่าโรลออฟ = 0.3



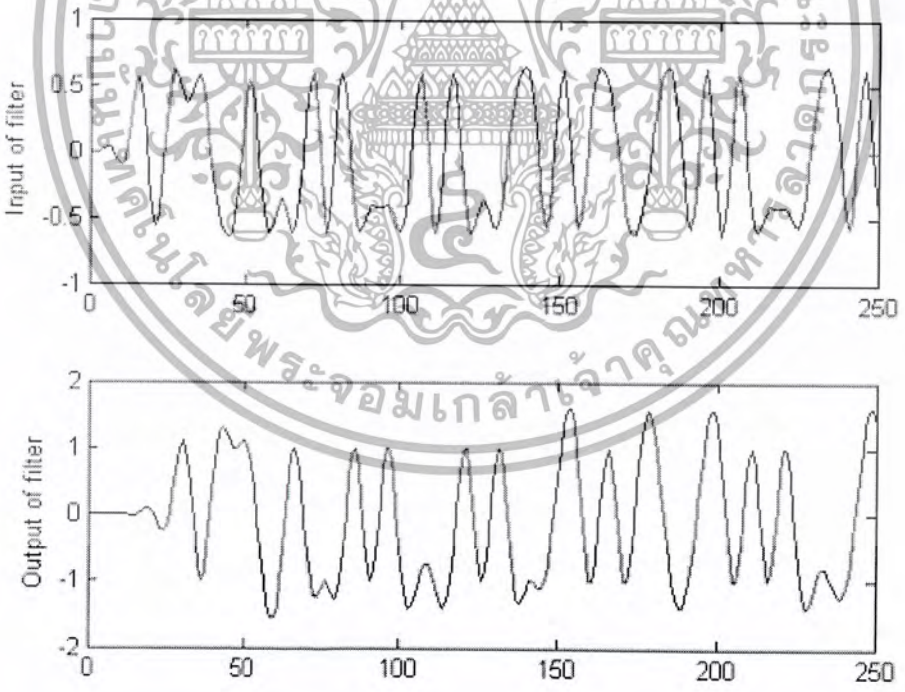
รูปที่ 4.8 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



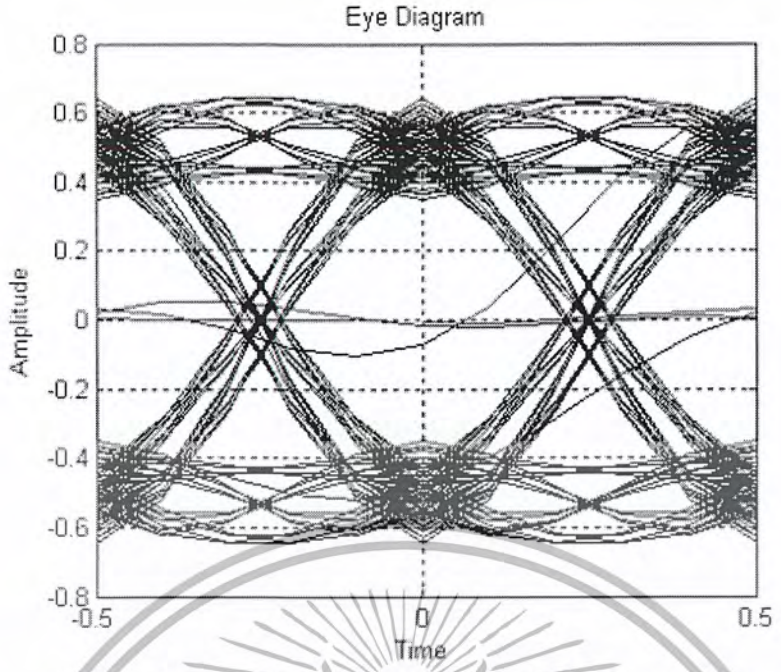
รูปที่ 4.9 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท ที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.3

- อินพุทมี โรลออฟ = 0.4

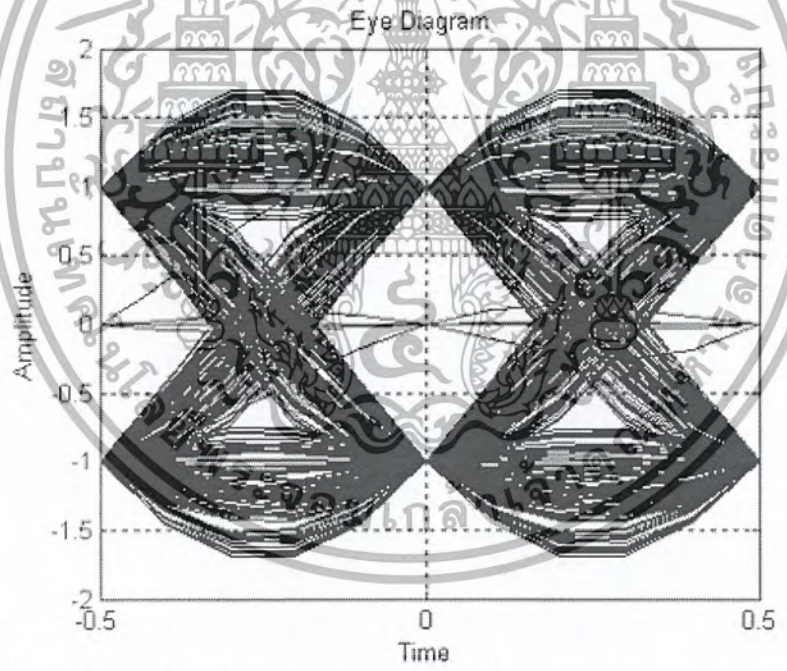


รูปที่ 4.10 สัญญาณอินพุทและเอาท์พุทของตัวกรอง ที่มีค่าโรลออฟ = 0.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



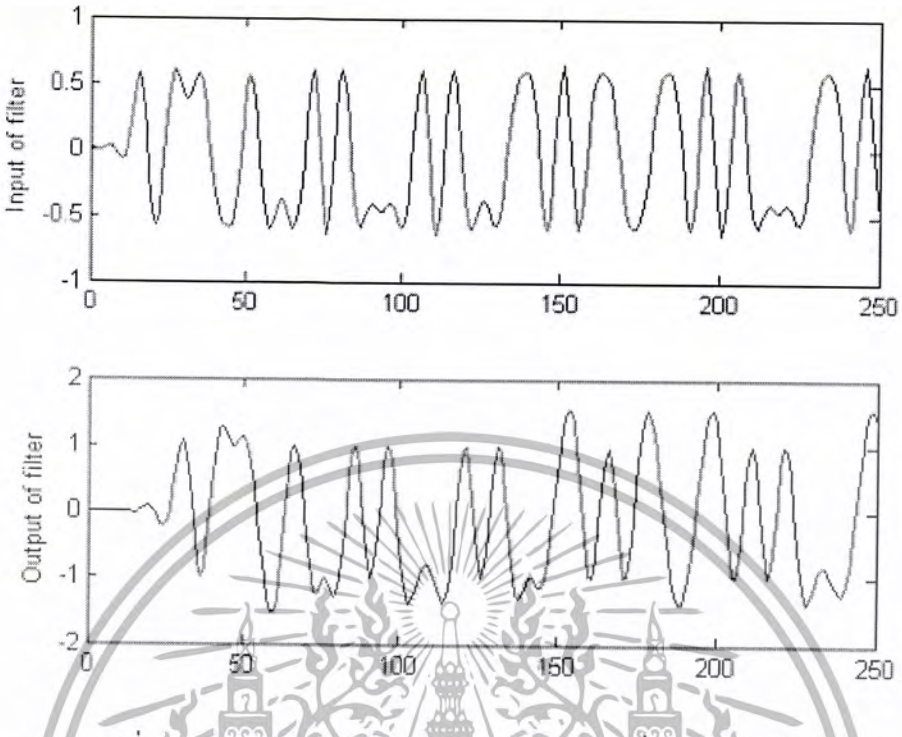
รูปที่ 4.11 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.4



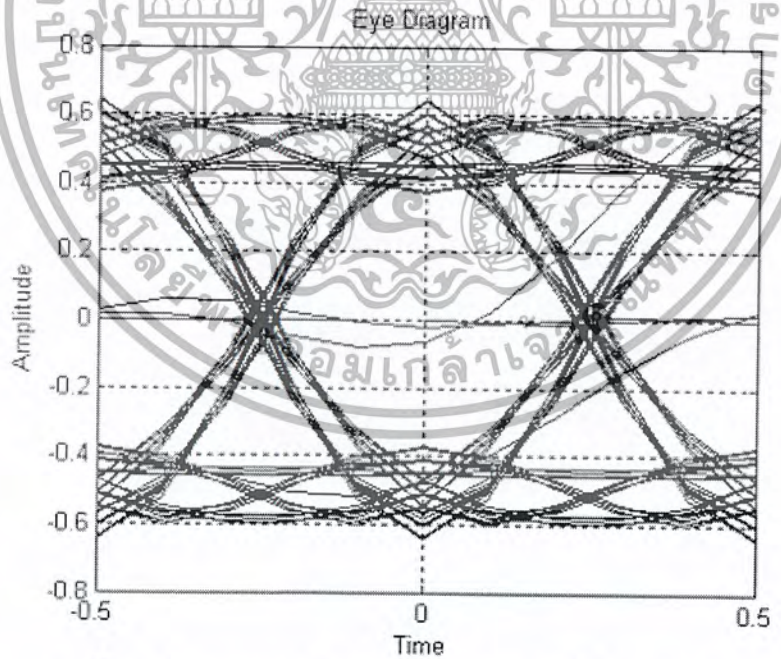
รูปที่ 4.12 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท ที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อินพุทมี โรลออฟ = 0.5

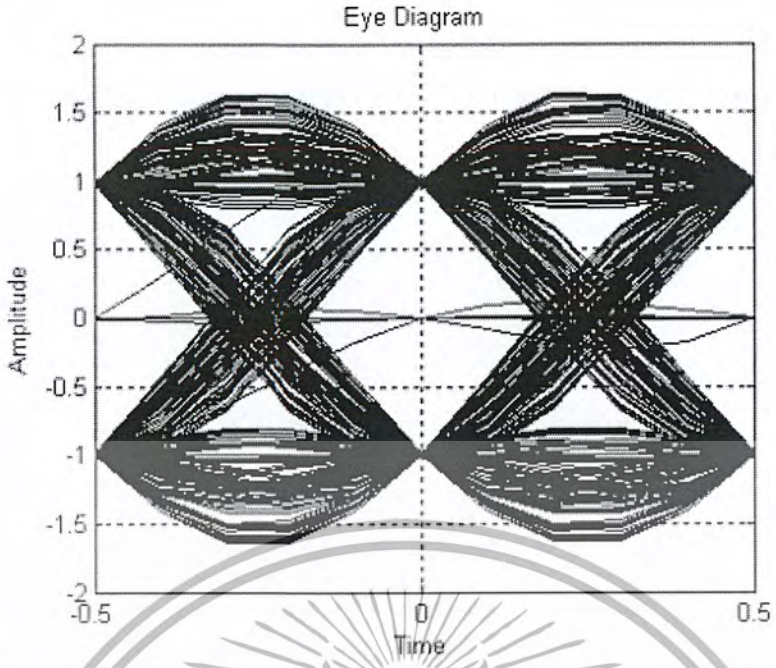


รูปที่ 4.13 สัญญาณอินพุทและเอาท์พุทของตัวกรอง ที่มีค่าโรลออฟ = 0.5



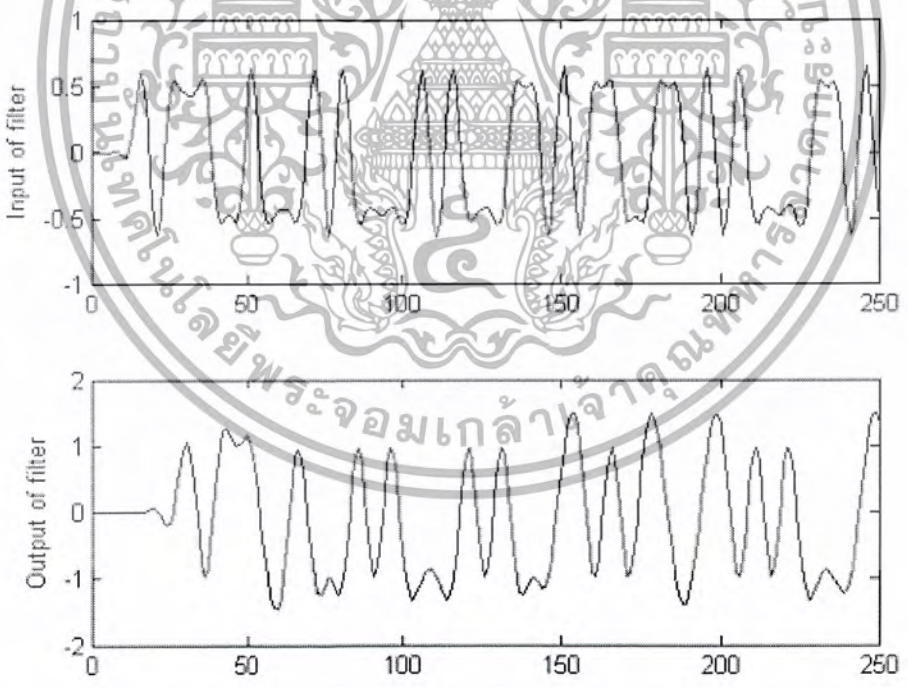
รูปที่ 4.14 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



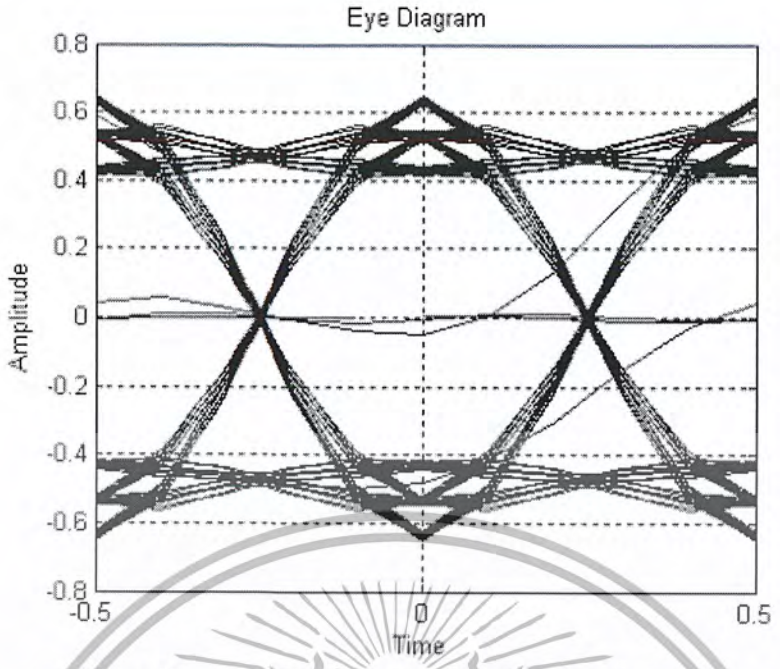
รูปที่ 4.15 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท ที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.5

- อินพุทมี โรลออฟ = 0.7

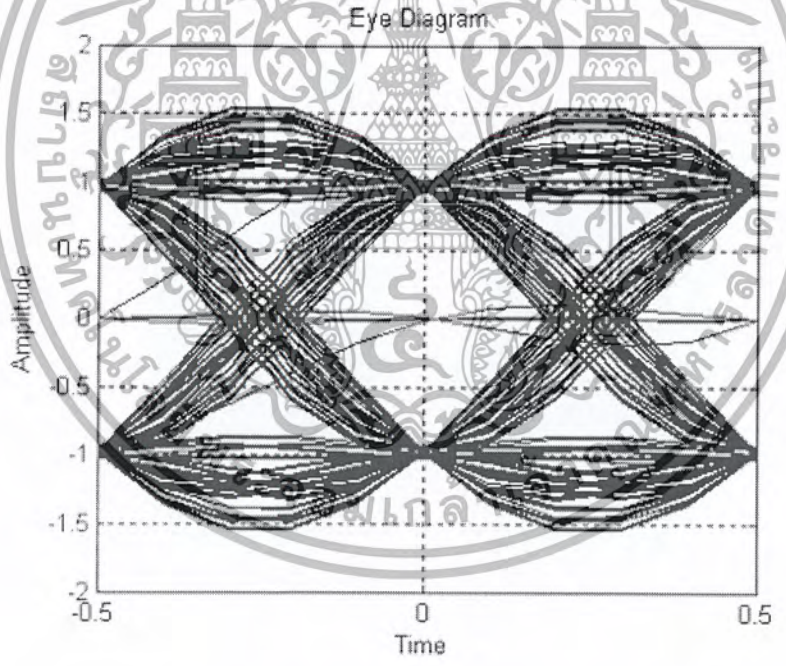


รูปที่ 4.16 สัญญาณอินพุทและเอาท์พุทของตัวกรอง ที่มีค่าโรลออฟ = 0.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



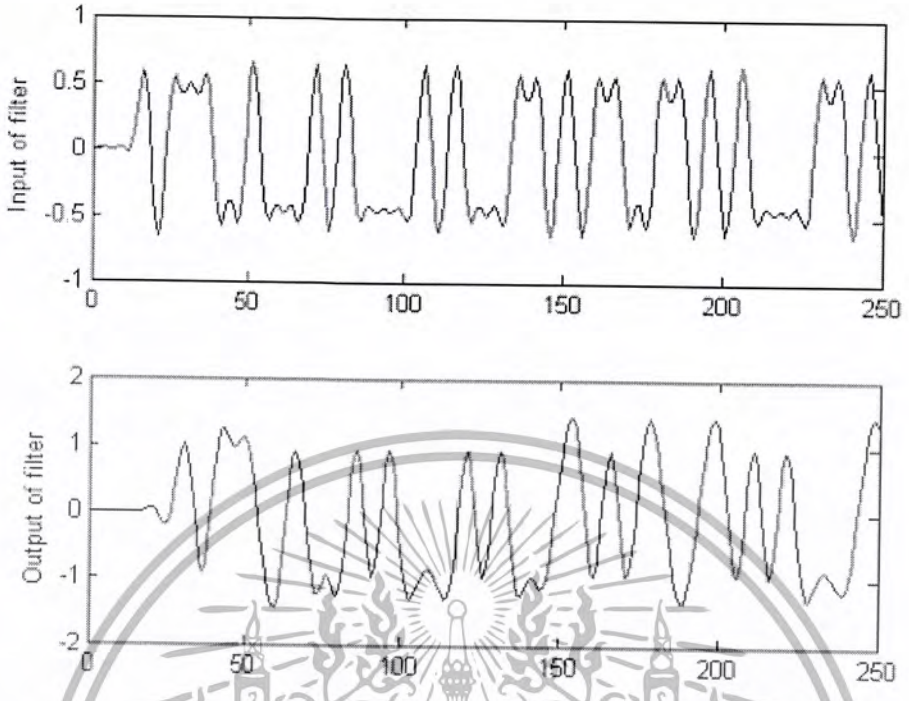
รูปที่ 4.17 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.7



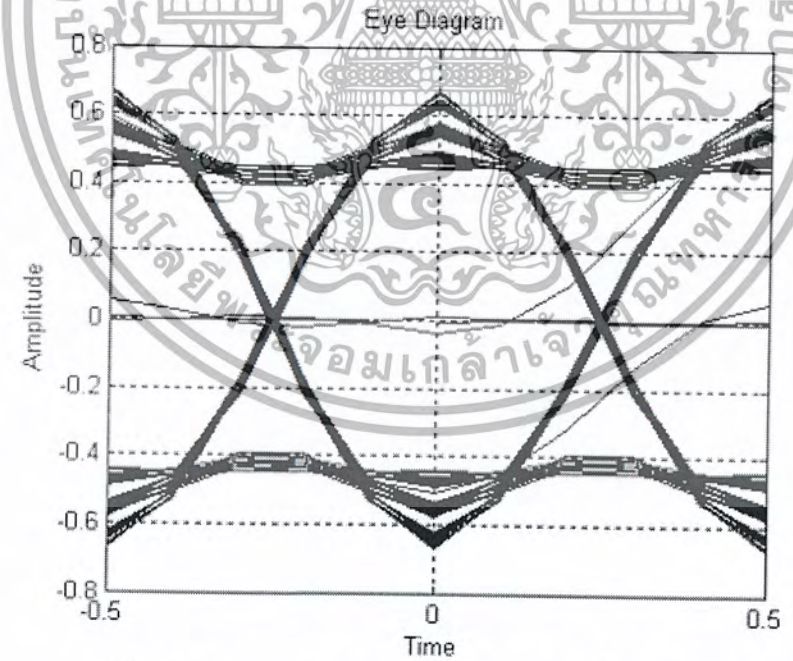
รูปที่ 4.18 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท ที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-อินพุทมี โรลออฟ = 0.9

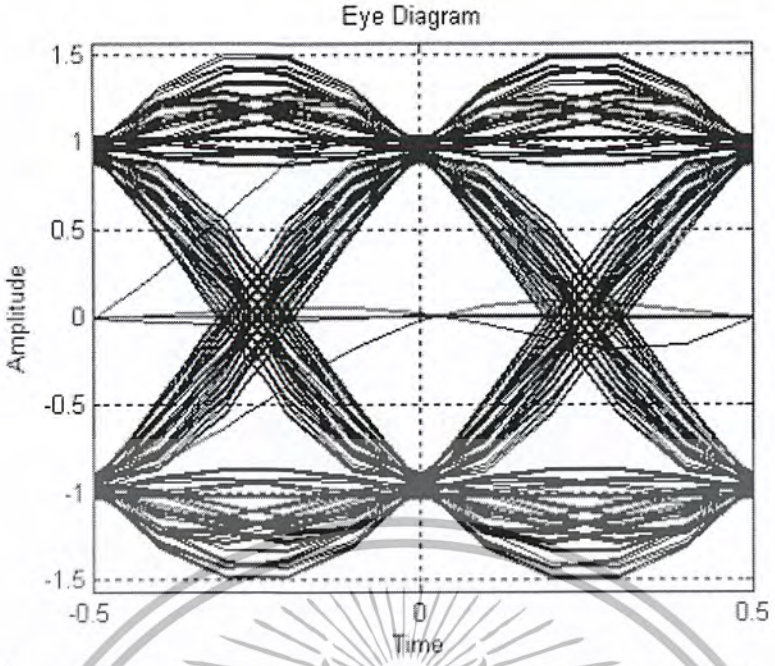


รูปที่ 4.19 สัญญาณอินพุทและเอาท์พุทของตัวกรอง ที่มีค่าโรลออฟ = 0.9



รูปที่ 4.20 ไดอะแกรมรูปตาของสัญญาณอินพุท ที่มีค่าโรลออฟ = 0.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท ที่รับสัญญาณอินพุทมีค่า โรลออฟ = 0.9

จากการซิมมูเลตจะเห็นได้ว่า เมื่อทำการกำหนดค่าโรลออฟของตัวกรองมีค่าเท่ากับ 0.3 และนำสัญญาณอินพุทที่มีค่าโรลออฟน้อยกว่า 0.3 มาผ่านตัวกรอง เอาท์พุทที่ได้จะมีไดอะแกรมรูปตาในช่วงที่เวลาเป็นศูนย์กว้าง มีช่วงเปิดที่แคบ ซึ่งแสดงให้เห็นว่าเกิดการแทรกสอดทางด้านเอาท์พุทขึ้น นั่นคือสัญญาณอินพุทกับตัวกรองไม่แมทช์กัน

แต่เมื่อนำสัญญาณอินพุทที่มีค่าโรลออฟตั้งแต่ 0.3 ถึง 0.4 มาผ่านตัวกรอง เอาท์พุทที่ได้จะมีไดอะแกรมรูปตาในช่วงเวลาที่เป็นศูนย์แคบ คือมีช่วงเปิดที่กว้าง ซึ่งแสดงให้เห็นว่าไม่เกิดการแทรกสอดทางด้านเอาท์พุท นั่นคือสัญญาณอินพุทกับตัวกรองเกิดการแมทช์กัน

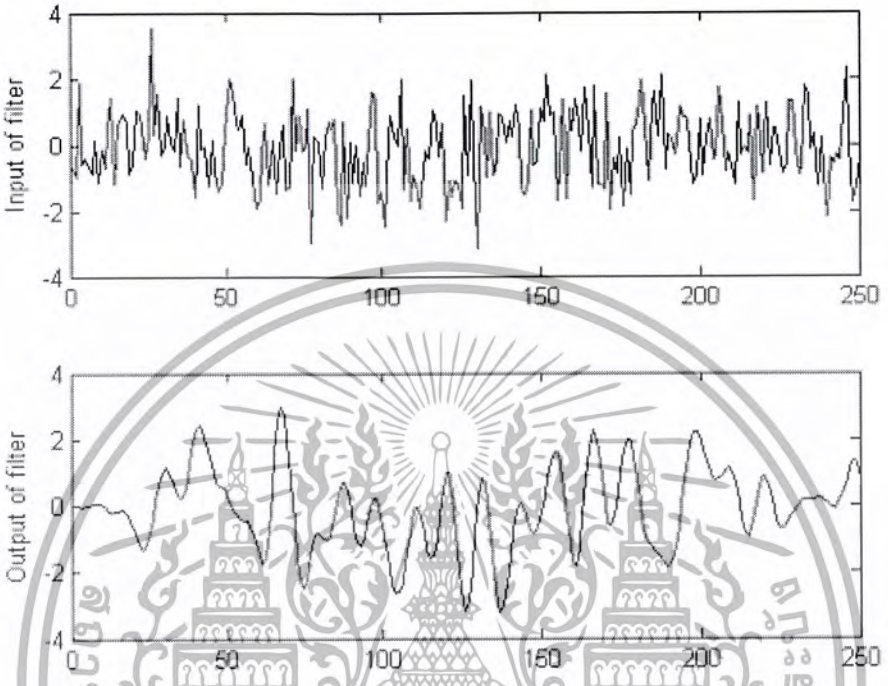
และเมื่อนำสัญญาณอินพุทที่มีค่าโรลออฟตั้งแต่ 0.5 ขึ้นไป มาผ่านตัวกรอง เอาท์พุทที่ได้จะมีไดอะแกรมรูปตาในช่วงเวลาที่เป็นศูนย์กว้างขึ้นเรื่อยๆ ทำให้มีช่วงเปิดที่แคบลงเรื่อยๆ ซึ่งจะเกิดการสอดแทรกขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

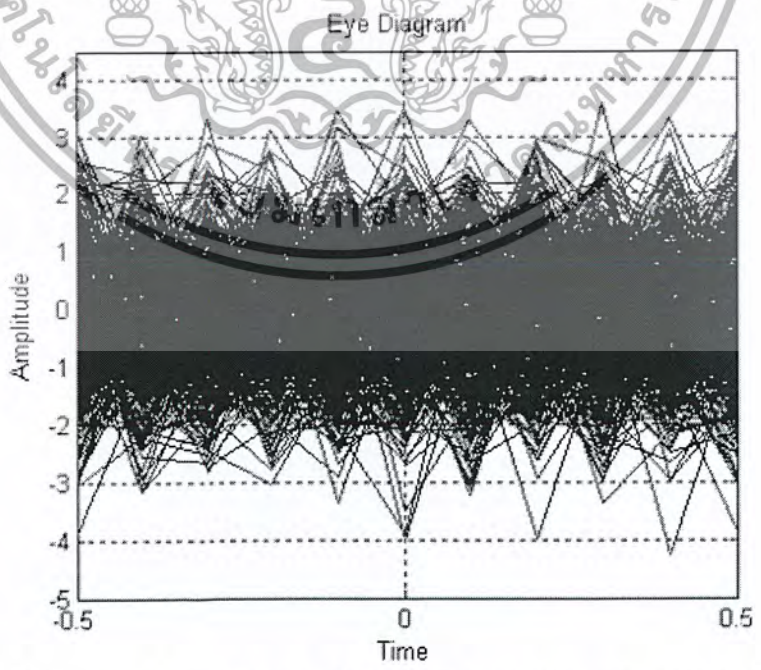
4.1.2 ป้อนสัญญาณ Root-raised cosine ที่มีสัญญาณรบกวนแบบ AWGN ให้กับตัวกรอง

โดยกำหนดให้ตัวกรองและสัญญาณอินพุทมีค่าโรลออฟเท่ากับ 0.3 เท่ากัน และทำการเปลี่ยนแปลงค่า SNR(Signal to Noise Ratio) เพื่อสังเกตเอาท์พุทจากตัวกรอง

- SNR = 0

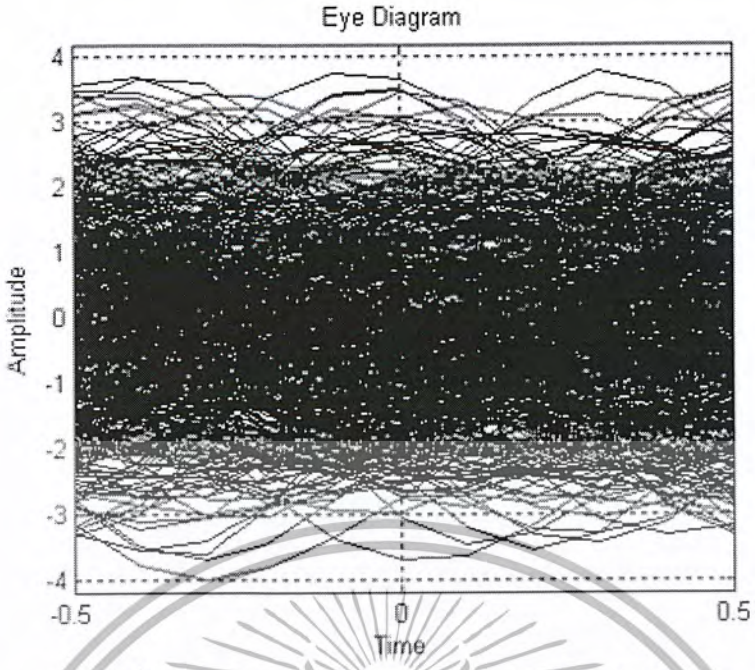


รูปที่ 4.22 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรอง มี SNR = 0

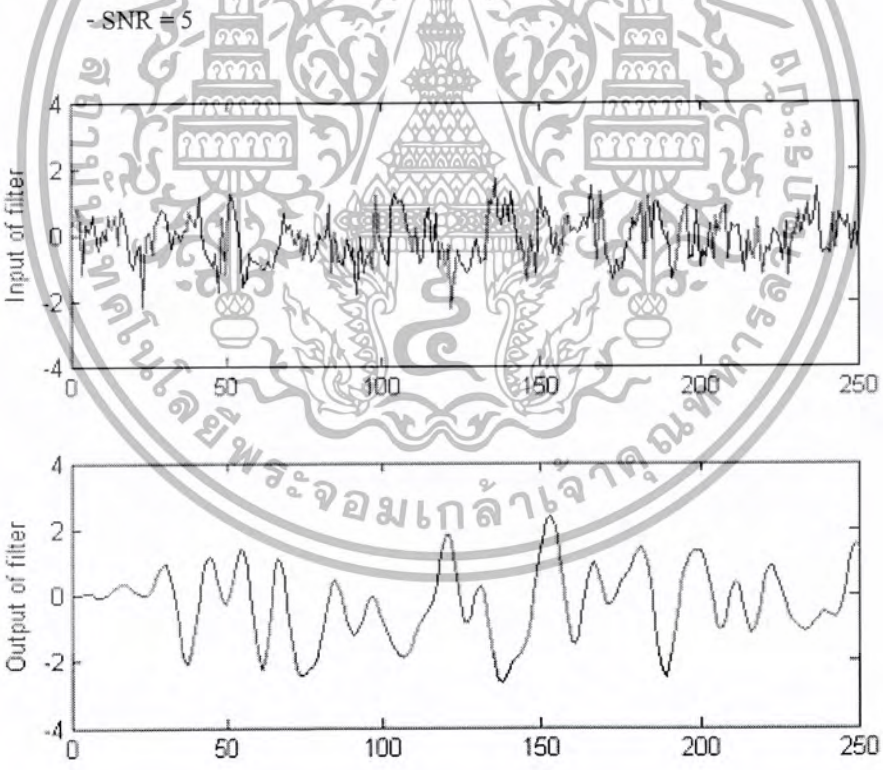


รูปที่ 4.23 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

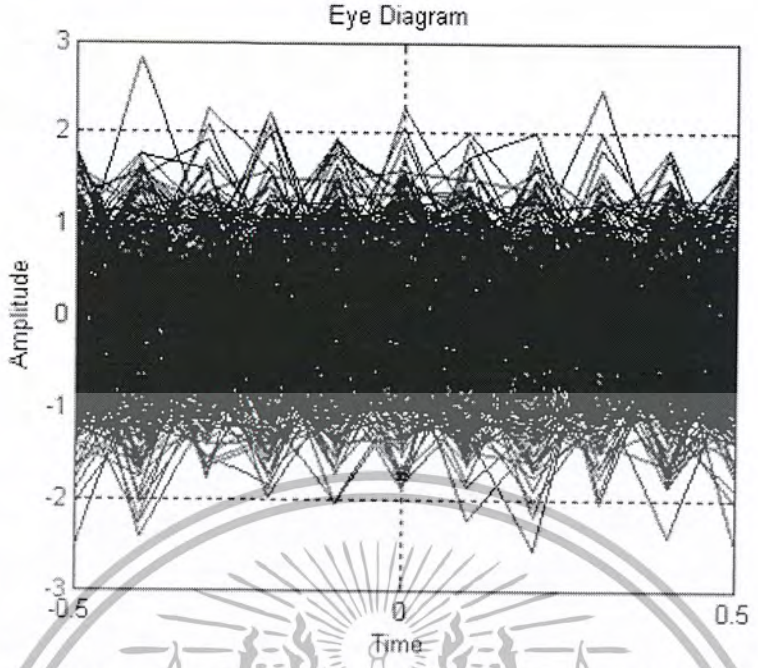


รูปที่ 4.24 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท มี SNR = 0

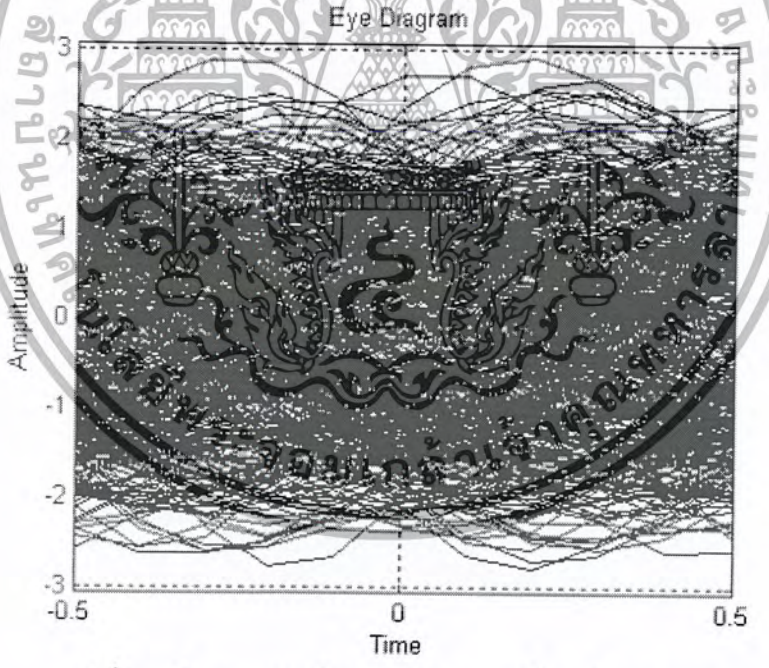


รูปที่ 4.25 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรอง มี SNR = 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



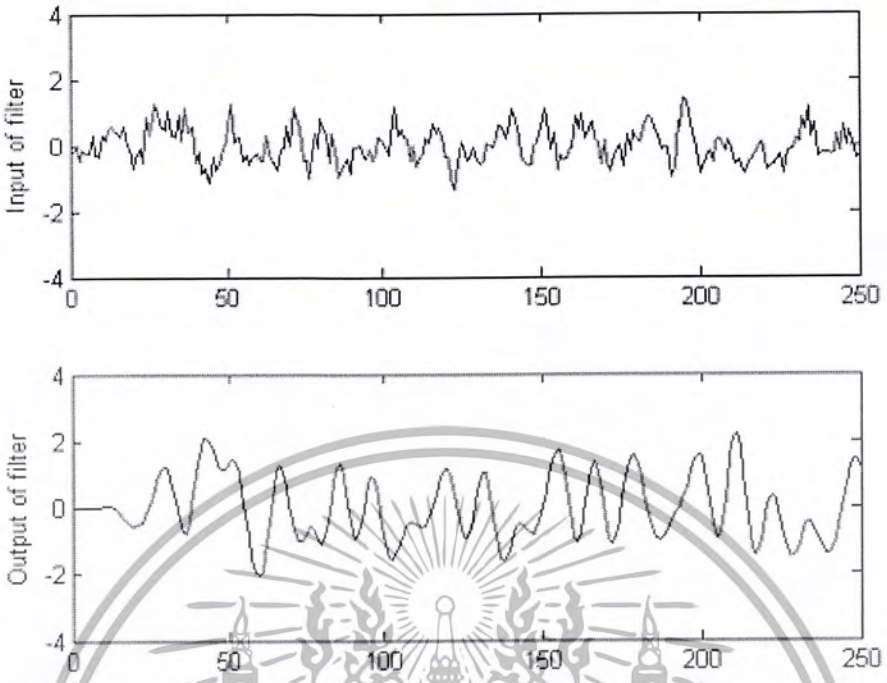
รูปที่ 4.26 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 5



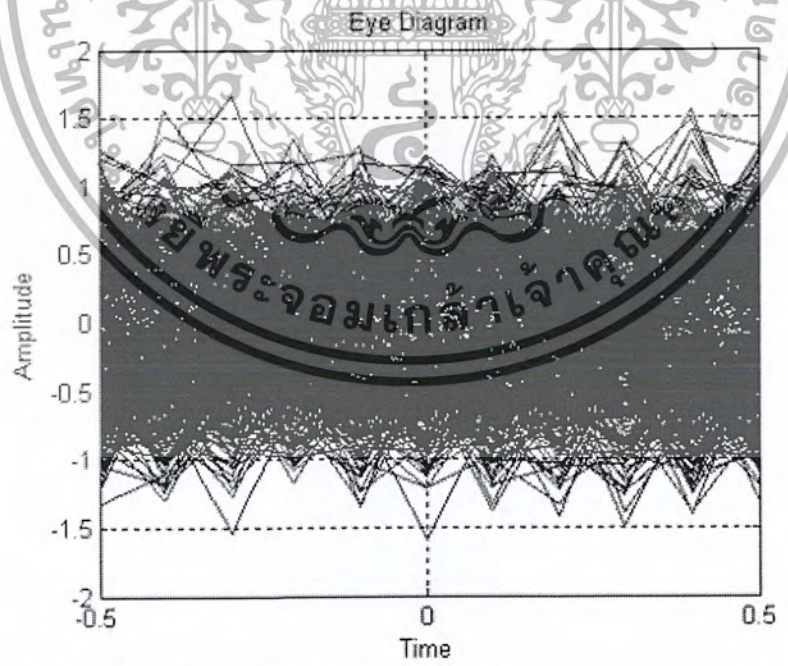
รูปที่ 4.27 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท มี SNR = 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- SNR = 10

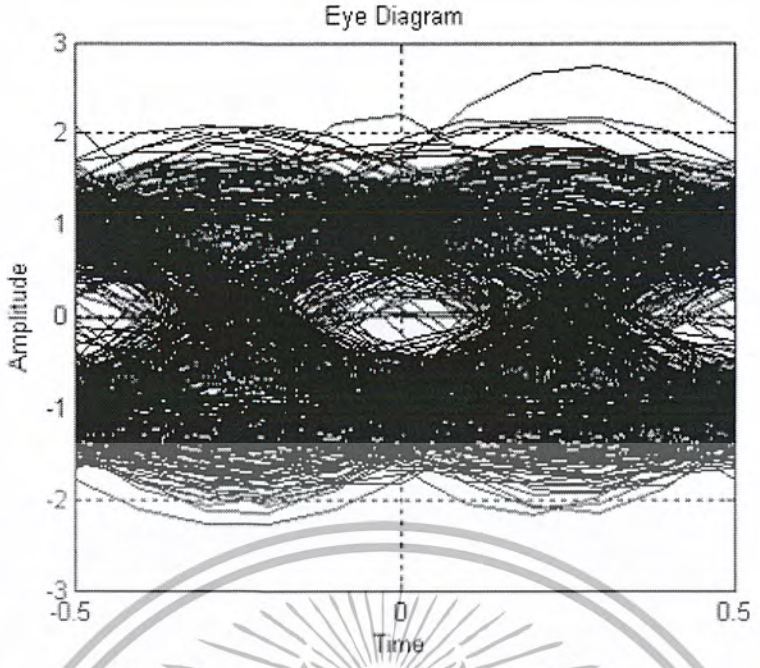


รูปที่ 4.28 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรอง มี SNR = 10

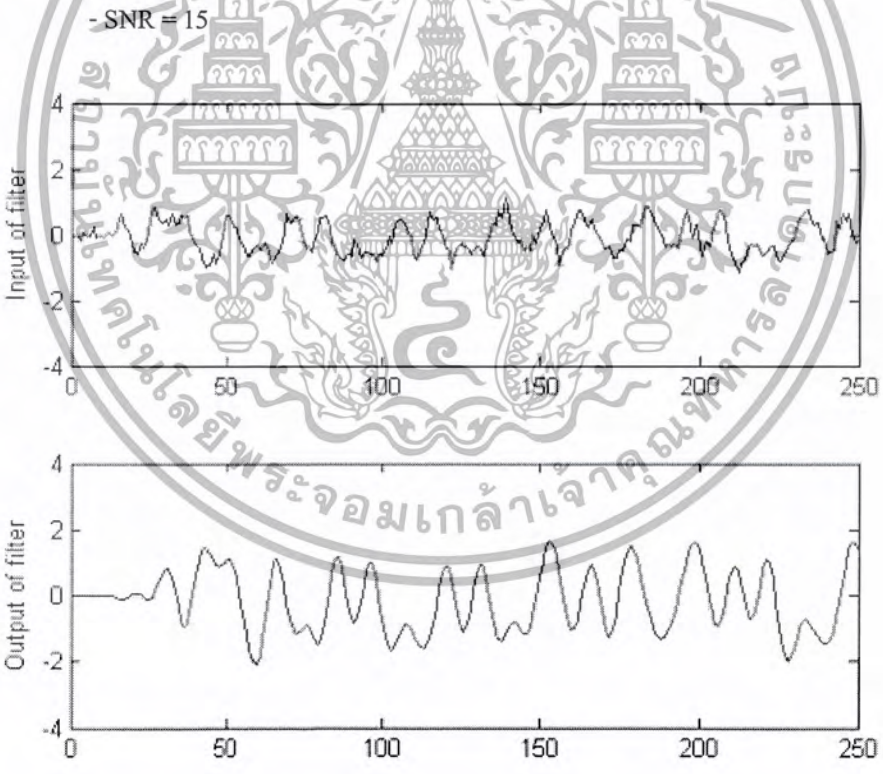


รูปที่ 4.29 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

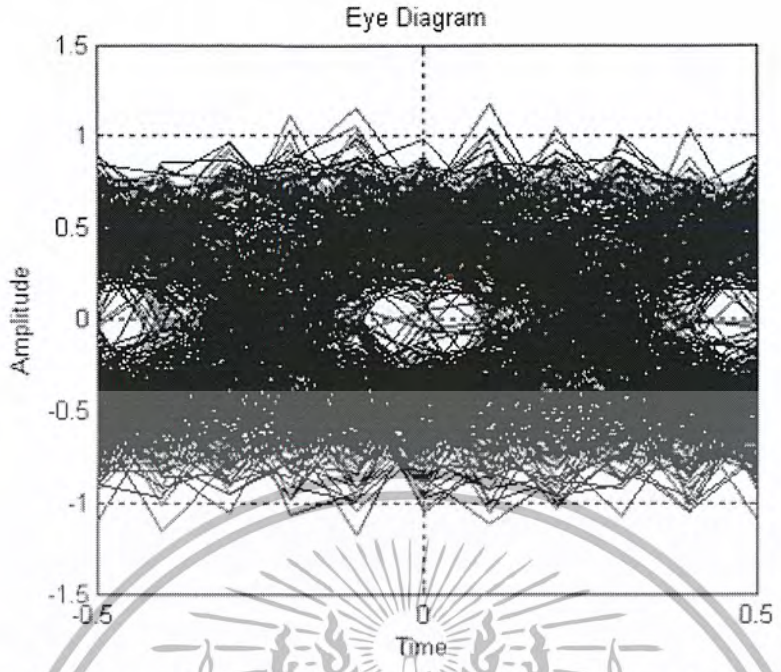


รูปที่ 4.30 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท มี SNR = 10

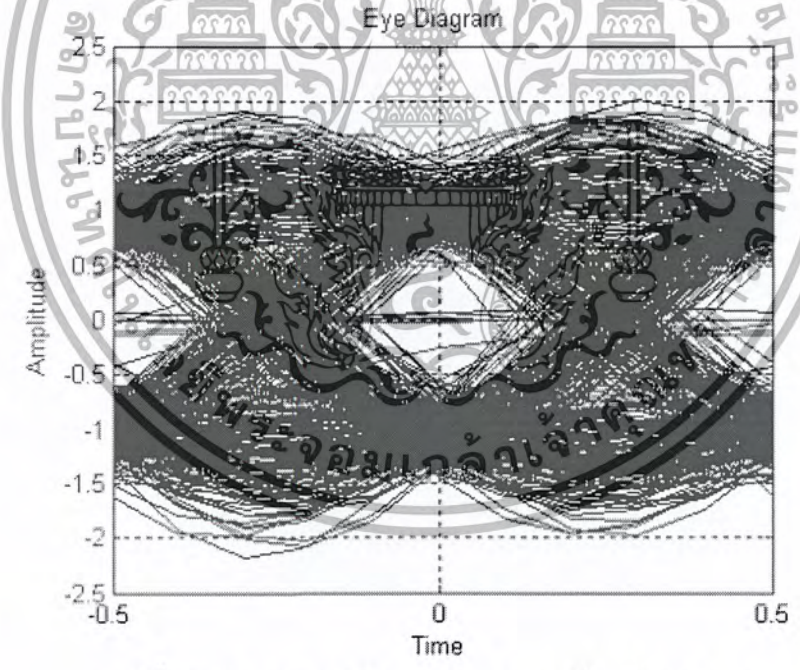


รูปที่ 4.31 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรอง มี SNR = 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



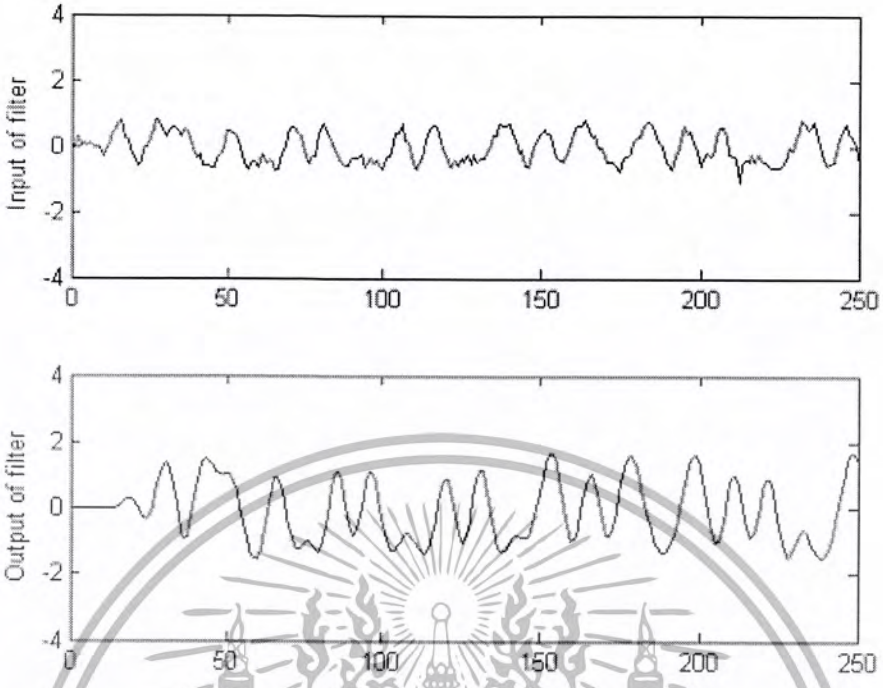
รูปที่ 4.32 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 15



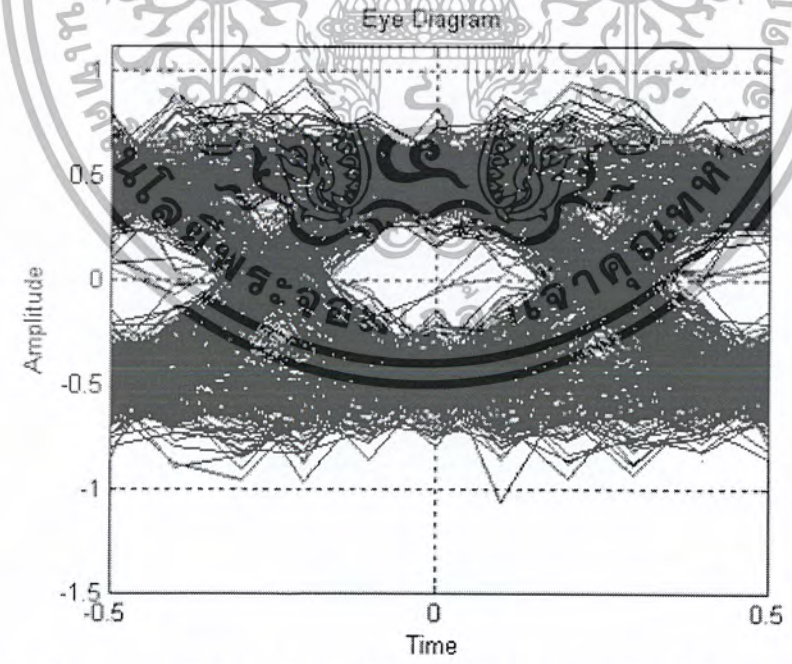
รูปที่ 4.33 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท มี SNR = 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- SNR = 20

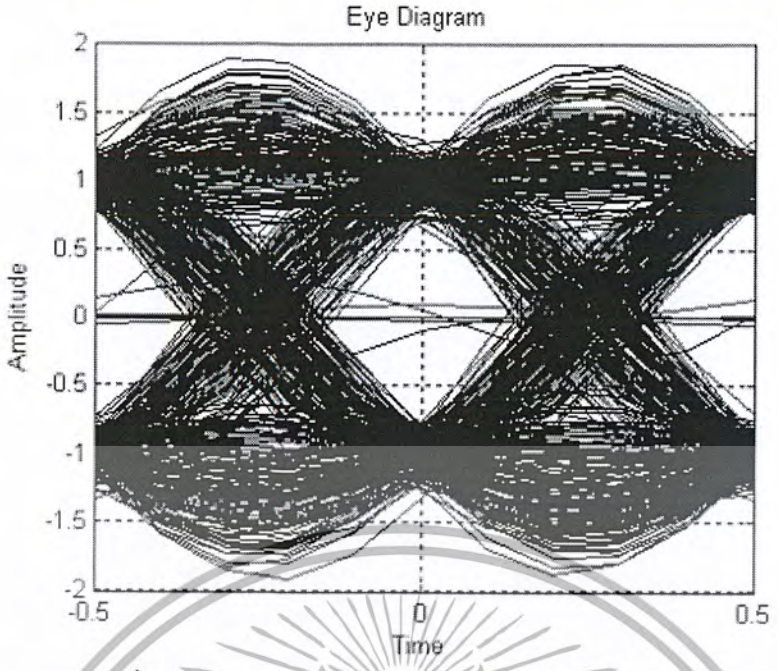


รูปที่ 4.34 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรอง มี SNR = 20

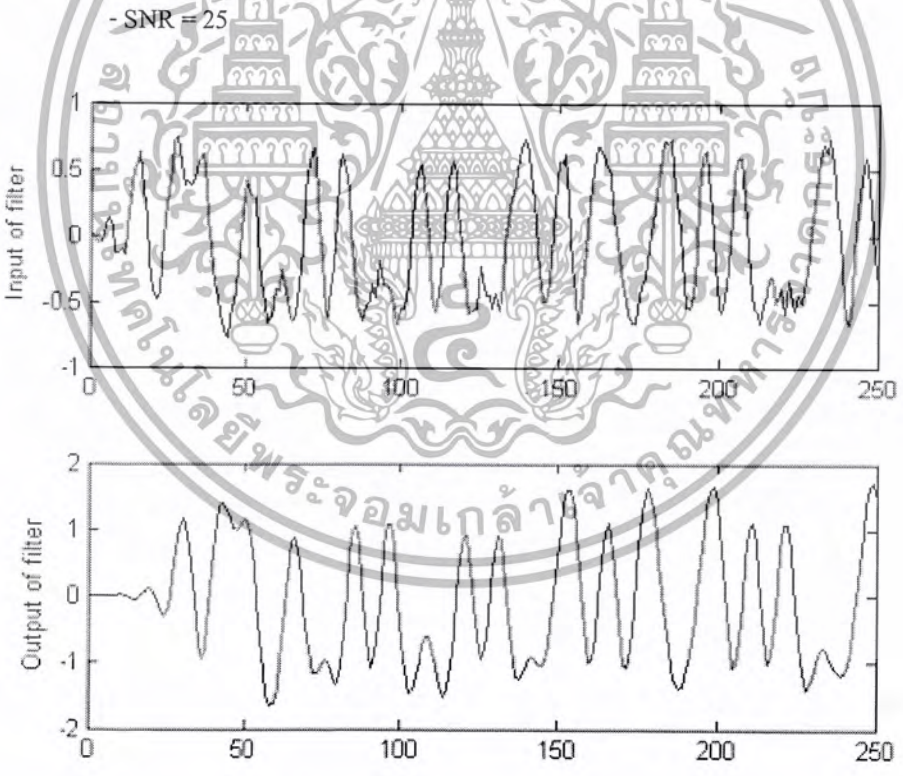


รูปที่ 4.35 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

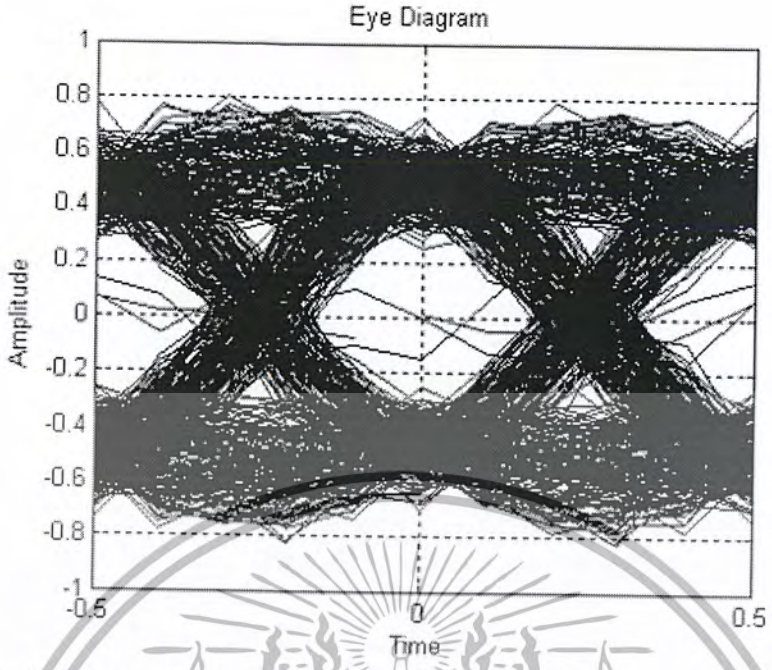


รูปที่ 4.36 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท มี SNR = 20

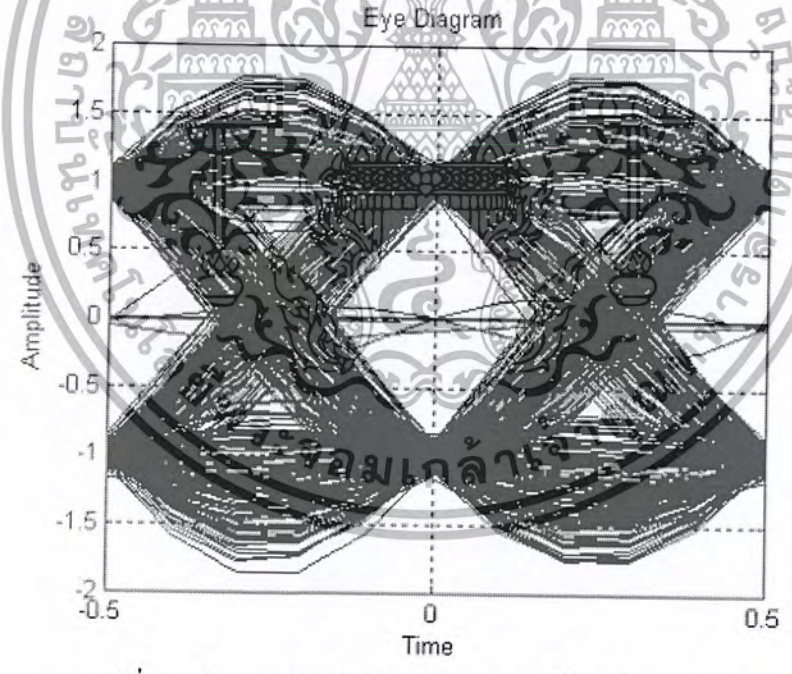


รูปที่ 4.37 สัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN และเอาท์พุทของตัวกรอง มี SNR = 25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.38 ไดอะแกรมรูปตาของสัญญาณอินพุทที่มีสัญญาณรบกวนแบบ AWGN มี SNR = 25

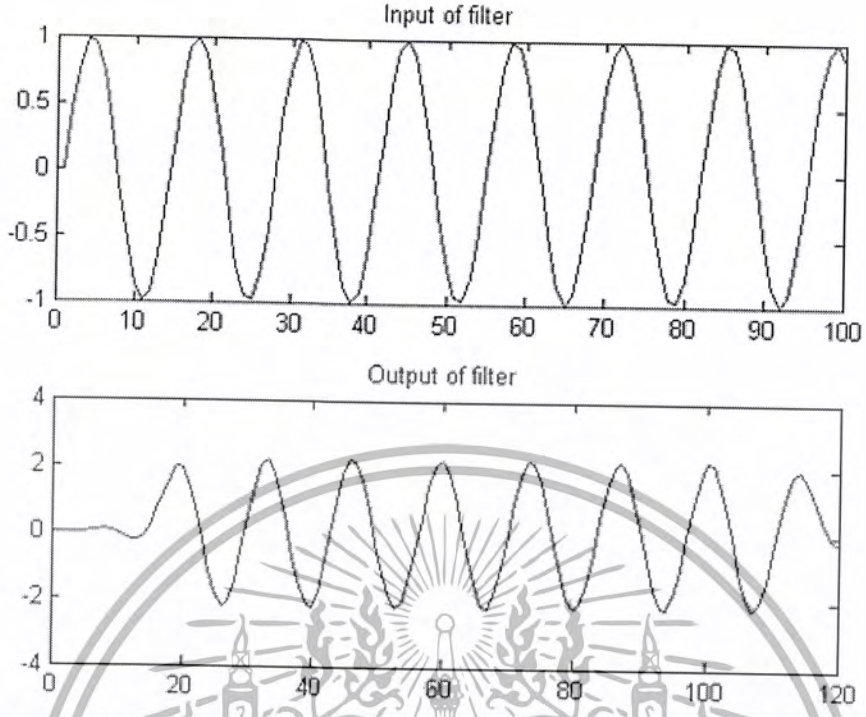


รูปที่ 4.39 ไดอะแกรมรูปตาของสัญญาณเอาท์พุท มี SNR = 25

จากการซิมมูเลตจะเห็นได้ว่า เมื่อทำการเพิ่มค่า SNR ให้มีค่าสูงขึ้นเรื่อยๆ จะสามารถกำจัดสัญญาณรบกวนได้มากขึ้นเช่นกัน และจะได้ไดอะแกรมรูปตาตรงช่วงเวลาที่เป็นศูนย์แคบลงเรื่อยๆ เช่นกัน ทำให้ไดอะแกรมรูปตามีช่วงเปิดที่กว้างขึ้น มีการสอดแทรกที่น้อยลง

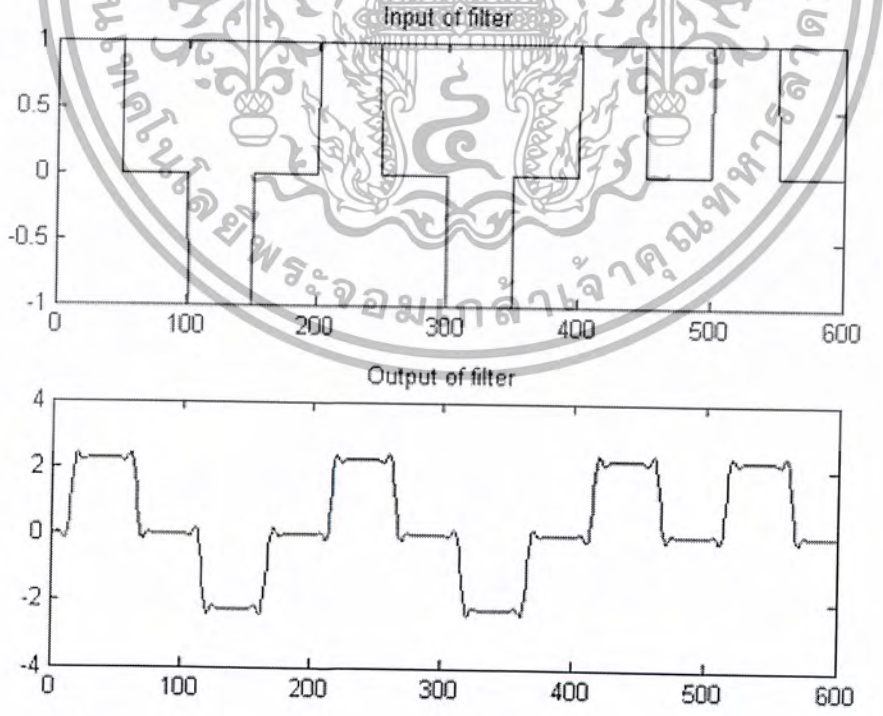
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 ป้อนสัญญาณรูปคลื่นไซน์ให้กับตัวกรอง



รูปที่ 4.40 สัญญาณอินพุทและเอาท์พุทที่เป็นสัญญาณรูปไซน์ของตัวกรอง

4.1.4 ป้อนสัญญาณรูปคลื่นสี่เหลี่ยมให้กับตัวกรอง

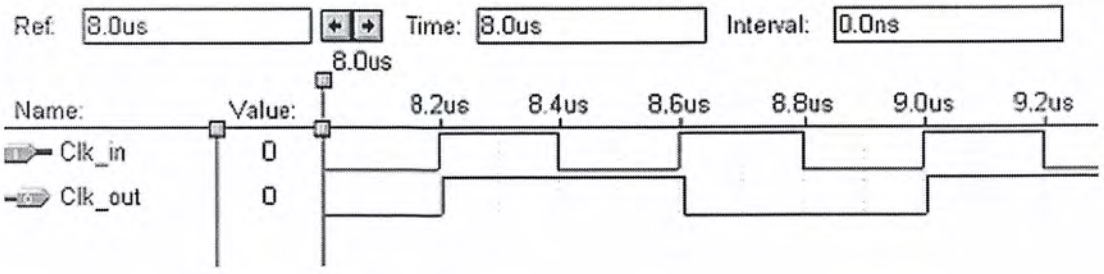


รูปที่ 4.41 สัญญาณอินพุท และเอาท์พุทที่เป็นสัญญาณรูปสี่เหลี่ยมของตัวกรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ขั้นตอนในการซิมมูเลตด้วยโปรแกรมวีเอชดีแอล

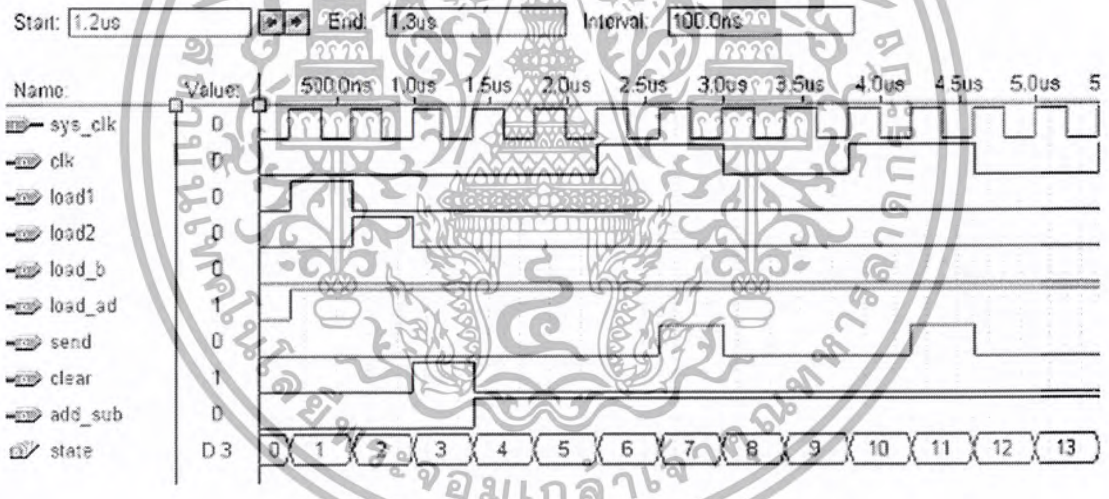
4.2.1 อุปกรณ์ DIVIDER



รูปที่ 4.42 การซิมมูเลตอุปกรณ์ DIV

ทำหน้าที่ ในการหารความถี่จากคริสตอลในบอร์ดเอฟพีจีเอ โดยขา Clk _in ทำการรับคล็อคเข้ามาเพื่อหารความถี่ จากรูปเป็นการหารความถี่ด้วย 2 และนำเอาท์พุทออกขา Clk_out

4.2.2 อุปกรณ์ CONTROL



รูปที่ 4.43 การซิมมูเลตอุปกรณ์ CONTROL

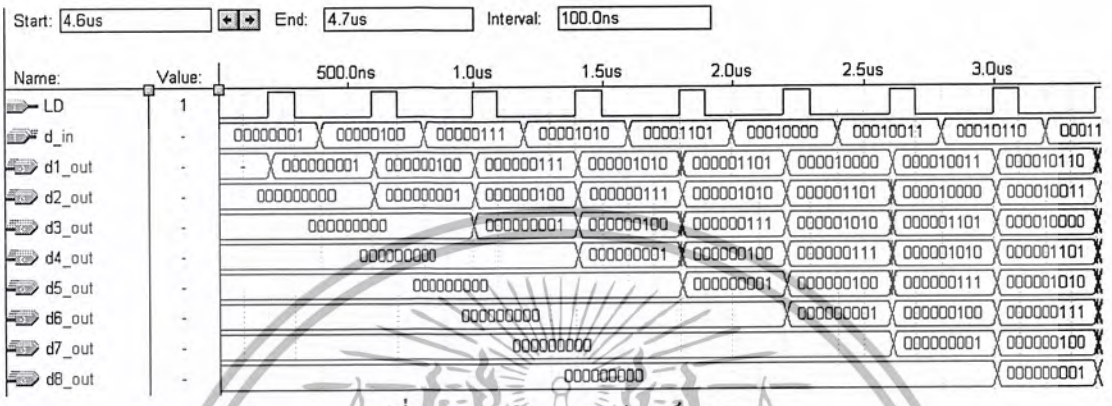
ทำหน้าที่ ควบคุมอุปกรณ์ต่างๆภายในระบบ แต่ละขามีหน้าที่ดังนี้

- sys_clk ทำหน้าที่ รับคล็อคที่มาจากอุปกรณ์ DIV
- load1 ทำหน้าที่ ควบคุมอุปกรณ์ PIPO ให้รับค่ามาจาก ADC
- load2 ทำหน้าที่ ควบคุมอุปกรณ์ SHIFT ให้รับค่าที่ออกมาจาก PIPO
- load_b ทำหน้าที่ ควบคุมอุปกรณ์ ADD ให้ส่งค่าออกทางเอาท์พุท
- load_ad ทำหน้าที่ ป้อนคล็อคให้กับอุปกรณ์ ADC เพื่อให้ ADC ป้อนอินพุทให้กับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- send ทำหน้าที่ ควบคุมอุปกรณ์ ADD ให้ป้อนค่ากลับมาบวกใหม่ภายใน ADD
- clear ทำหน้าที่ เคลียร์ข้อมูลภายในอุปกรณ์ ADD
- add_sub ทำหน้าที่ ควบคุมให้อุปกรณ์ ADD ทำการบวกหรือลบ โดยถ้าเป็น 1 จะทำการบวก แต่ถ้าเป็น 0 จะทำการลบ

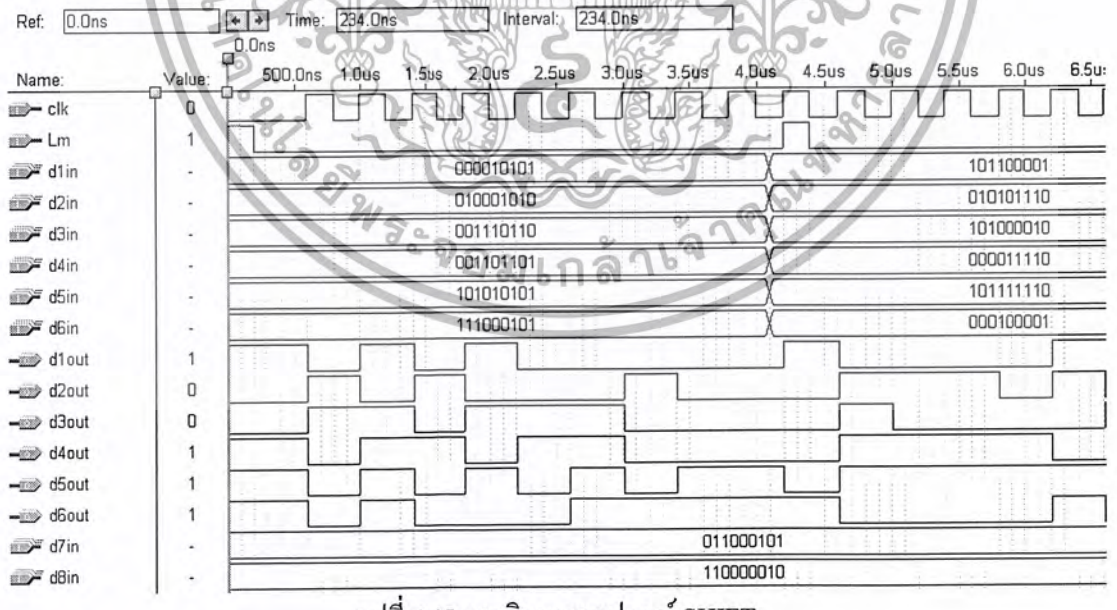
4.2.3 อุปกรณ์ PIPO



รูปที่ 4.44 การซิมมูเลตอุปกรณ์ PIPO

ทำหน้าที่ รับข้อมูลเข้ามาและเลื่อนไปให้กับขาต่อไป ข้อมูลจะถูกเลื่อนไปทั้งหมด โดยที่ขา LD จะทำการโหลดข้อมูลเข้ามาภายในตัว PIPO

4.2.4 อุปกรณ์ SHIFT

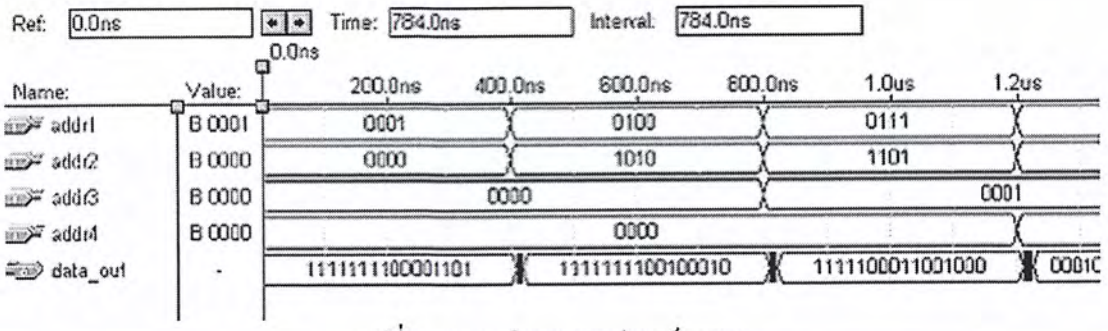


รูปที่ 4.45 การซิมมูเลตอุปกรณ์ SHIFT

ทำหน้าที่ เลื่อนข้อมูลทีละบิท และนำบิท LSB ไปเป็นแอดเดรสให้กับ ROM โดยข้อมูลจะถูกเลื่อนด้วยคล็อค และโหลดข้อมูลด้วยขา Lm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

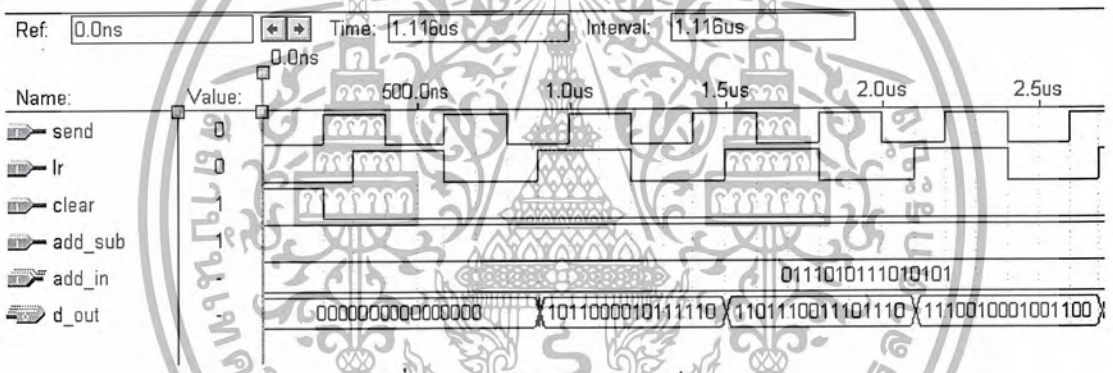
4.2.5 อุปกรณ์ ROM



รูปที่ 4.46 การซิมมูเลตอุปกรณ์ ROM

ทำหน้าที่ เก็บค่าสัมประสิทธิ์ของตัวกรองเอาไว้ โดยแบ่งเป็น 4 ชุด มี adddr1 , addr2, addr3 , addr4 เป็นแอดเดรสชุดที่ 1, 2 , 3 และ 4 ตามลำดับ

4.2.6 อุปกรณ์ ADD

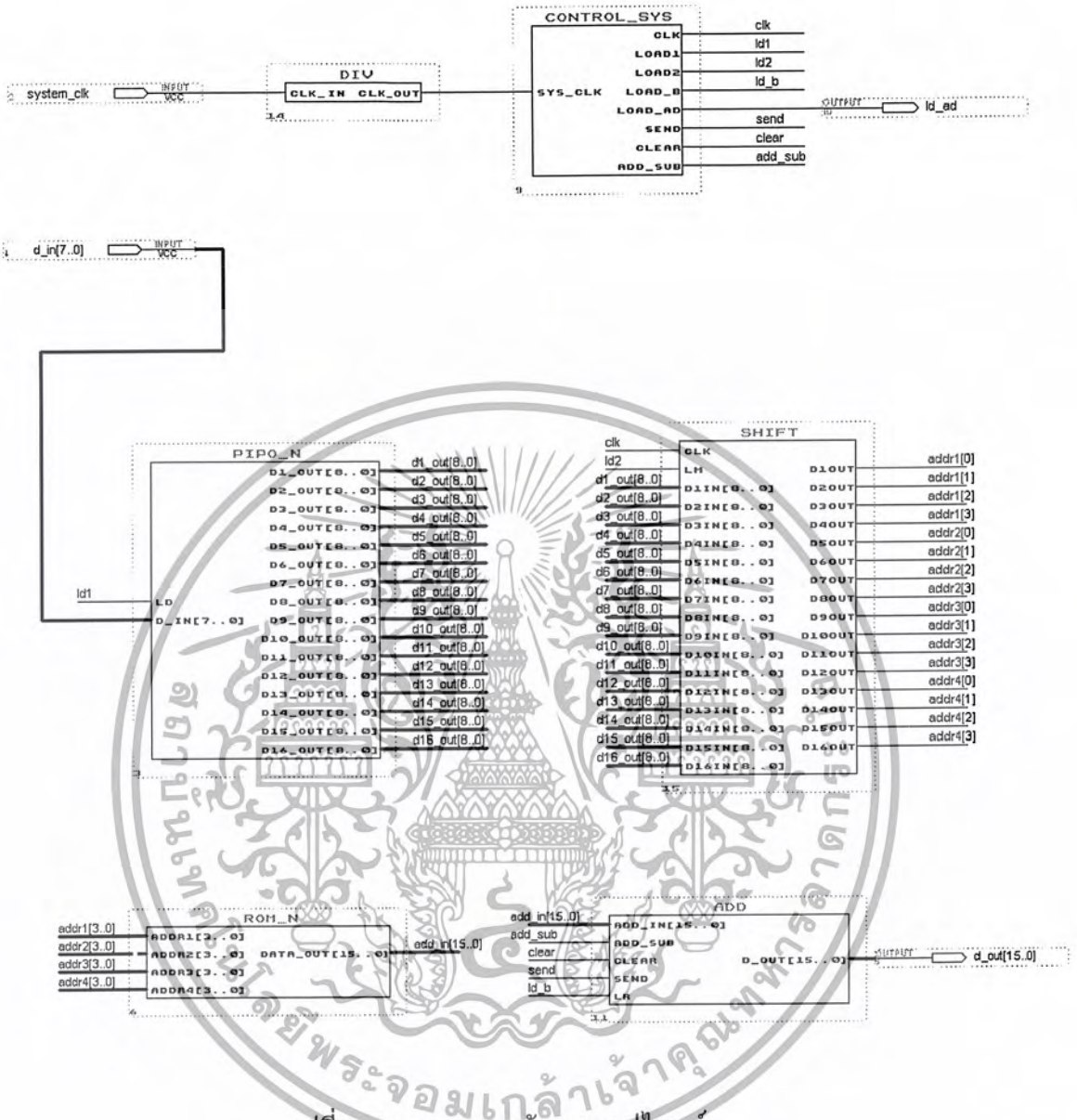


รูปที่ 4.47 การซิมมูเลตอุปกรณ์ ADD

ทำหน้าที่ บวกและลบค่าสัมประสิทธิ์ที่ออกมาจาก ROM และส่งออกเอาท์พุท โดยมีขา add_in เป็นขาอินพุทรับค่ามาจาก ROM และมีขา d_out เป็นเอาท์พุทของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.7 ทำการรวมอุปกรณ์ต่างๆ



รูปที่ 4.48 วงจรรวมของตัวกรองเอฟไออาร์

- จำนวนโลจิคเซล และทรัพยากรที่ใช้

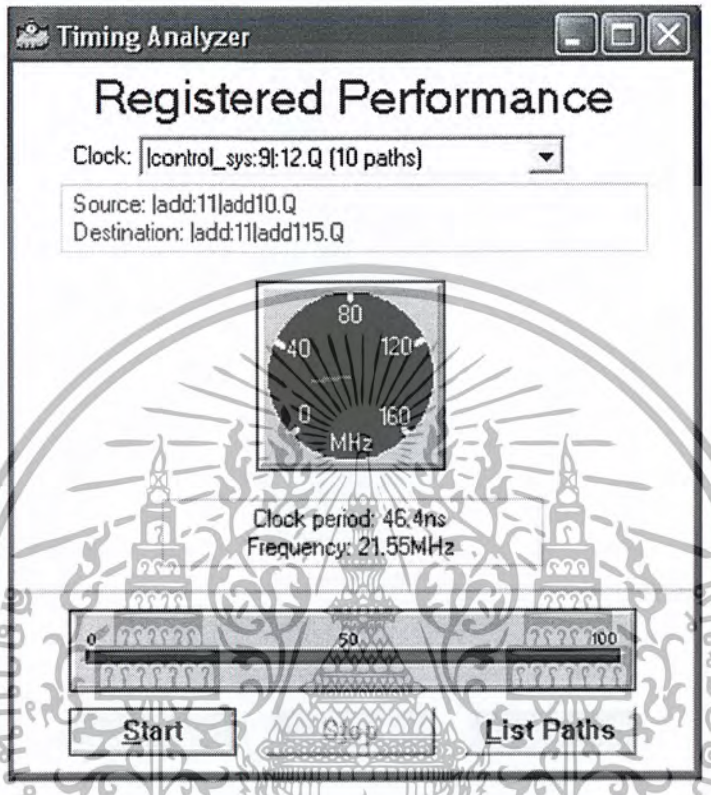
```
** DEVICE SUMMARY **

Chip/                                Input  Output  Bidir  Memory  Memory              LCs
POF       Device                     Pins   Pins    Pins   Bits %  Utilized  LCs  % Utilized

fir_n     EP1K50TC144-3              9      17      0      0          0 %     1108     38 %

User Pins:                           9      17      0
```

รูปที่ 4.49 จำนวนโลจิคเซล และทรัพยากรที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

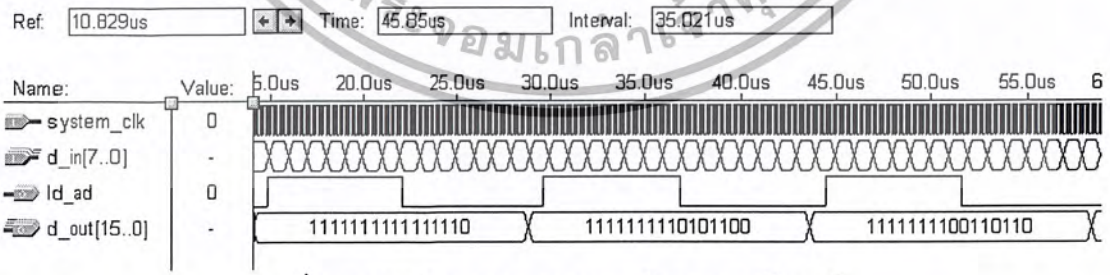
จากรูป บอร์ดที่ใช้คือ EPF1K50TC144-3 ตระกูล ACEX1K ซึ่งมีจำนวนเกตทั้งหมด 50,000 เกต จากการออกแบบวงจรกรองเอฟไออาร์จะใช้ขาอินพุททั้งหมด 9 ขา และเอาท์พุททั้งหมด 17 ขา ทรัพยากรที่ใช้ 38%

- ความเร็วสูงสุดในการทำงานของวงจร



รูปที่ 4.50 ความเร็วสูงสุดในการทำงานของวงจร

- ผลการซิมมูเลต



รูปที่ 4.51 การซิมมูเลตวงจรรวมของตัวกรองเอฟไออาร์

จากรูปเอาท์พุทของระบบจะมีจำนวน 16 บิท เพื่อให้ได้เอาท์พุทออกมามีความละเอียดมากที่สุด แต่การแสดงผลทางสโคปจะต้องผ่านวงจร DAC ที่มีการทำงานเพียง 8 บิท ดังนั้นจึงต้องนำ 8 บิท สูงสุดออกเอาท์พุท DAC แทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

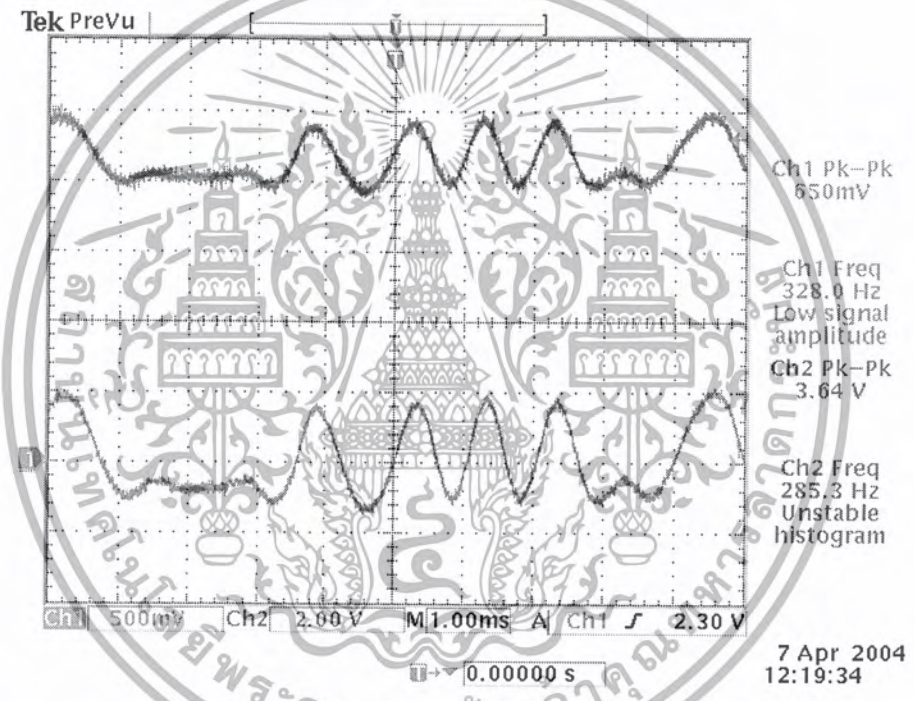
4.3 ขั้นตอนในการทดลองจริง

4.3.1 ป้อนสัญญาณ Root-raised cosine ให้กับตัวกรอง

โดยสัญญาณที่ป้อนให้กับวงจรจะทำการป้อนผ่านทางซาวน์การ์ดของคอมพิวเตอร์ ซึ่งจะป้อนไปให้กับวงจร ADC เพื่อแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิตอล จากนั้นทำการส่งผ่านไปให้กับบอร์ดเอฟพีจีเอที่ได้ออกแบบวงจรกรองความถี่ชนิดรูทเรสโคไซน์ไว้ ทางด้านเอาท์พุทจะผ่านวงจร DAC เพื่อแปลงสัญญาณดิจิตอลให้เป็นสัญญาณอนาล็อกอีกครั้ง

กำหนดให้ตัวกรองมีค่าโรลออฟแฟคเตอร์เท่ากับ 0.3 และทำการเปลี่ยนแปลงค่าโรล-ออฟแฟคเตอร์ของสัญญาณอินพุท เพื่อสังเกตเอาท์พุทจากตัวกรอง

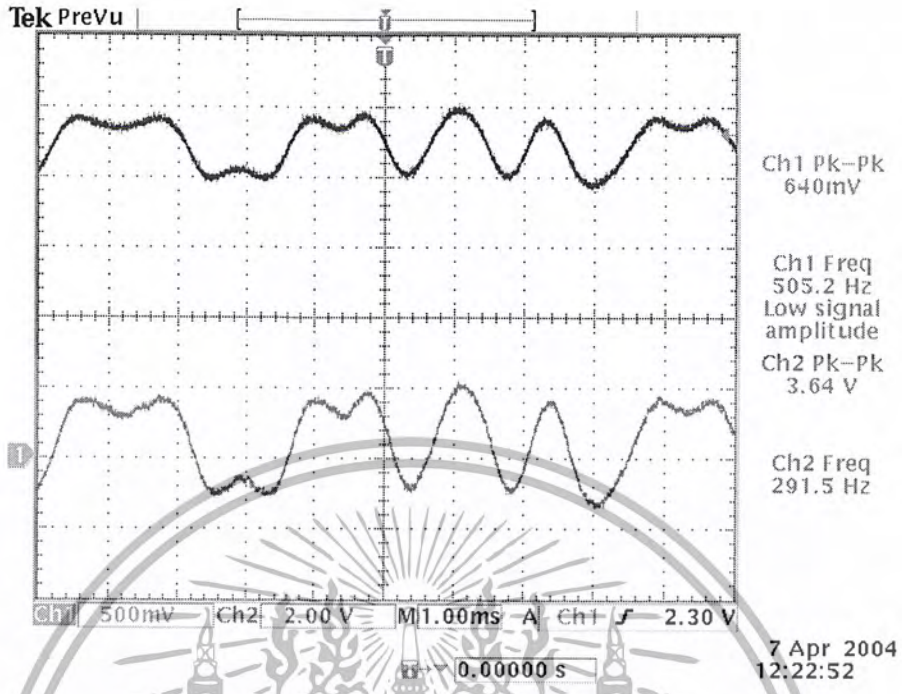
- อินพุทมี โรลออฟ = 0.1



รูปที่ 4.52 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.1 และเอาท์พุท(Ch2) ของตัวกรอง

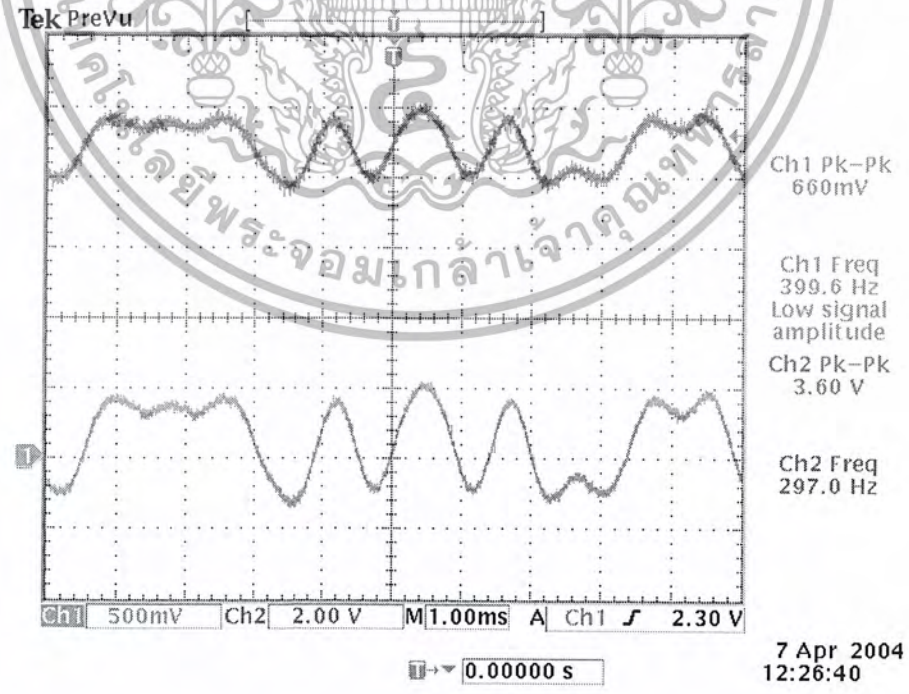
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อินพุทมี โรลออฟ = 0.2



รูปที่ 4.53 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.2 และเอาท์พุท(Ch2) ของตัวกรอง

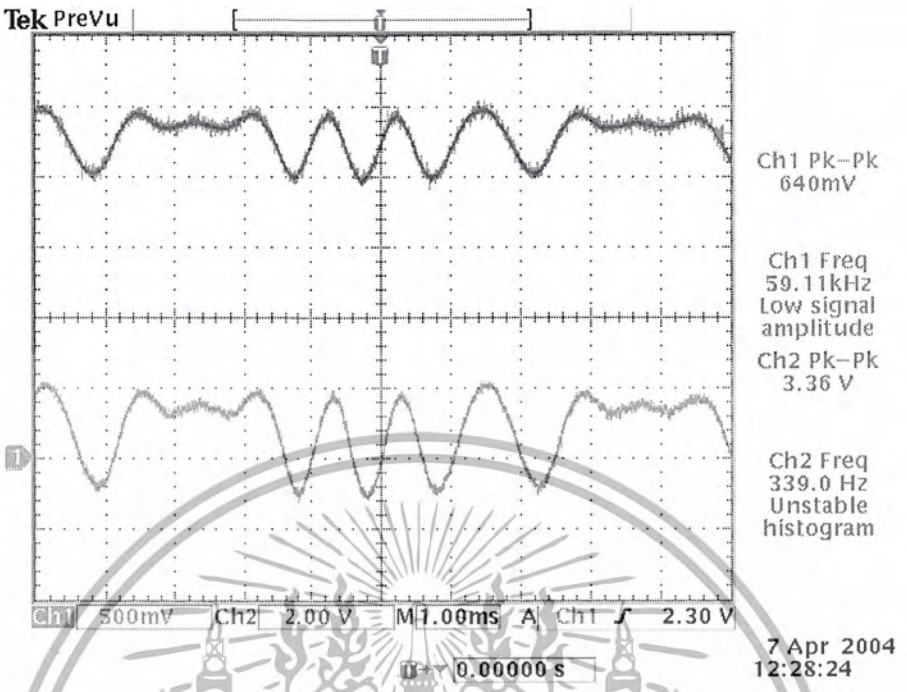
- อินพุทมี โรลออฟ = 0.3



รูปที่ 4.54 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.3 และเอาท์พุท(Ch2) ของตัวกรอง

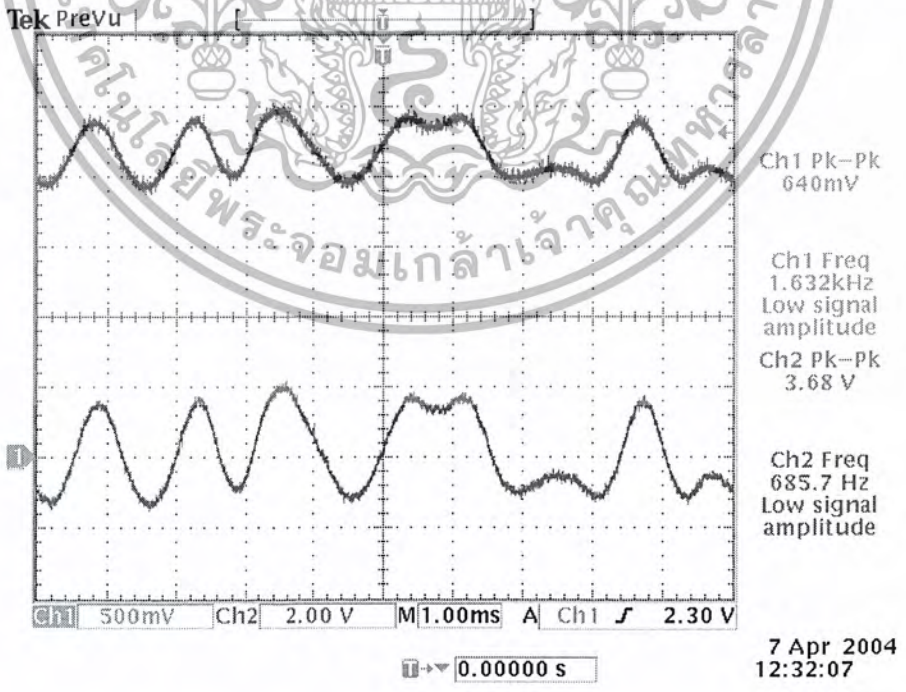
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อินพุทมี โรลออฟ = 0.4



รูปที่ 4.55 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.4 และเอาท์พุท(Ch2) ของตัวกรอง

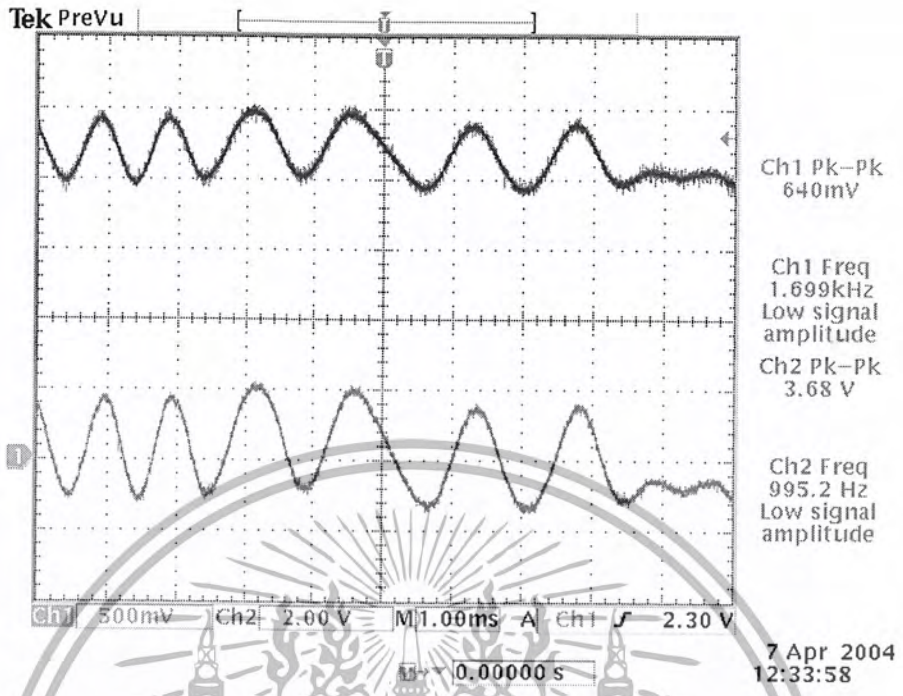
- อินพุทมี โรลออฟ = 0.5



รูปที่ 4.56 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.5 และเอาท์พุท(Ch2) ของตัวกรอง

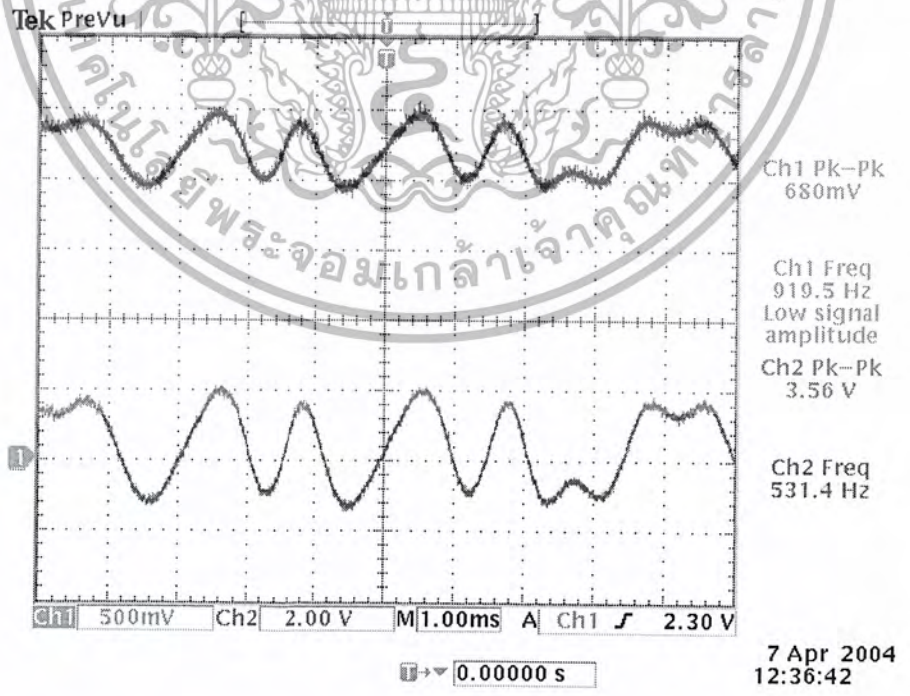
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อินพุทมี โรลออฟ = 0.6



รูปที่ 4.57 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.6 และเอาท์พุท(Ch2) ของตัวกรอง

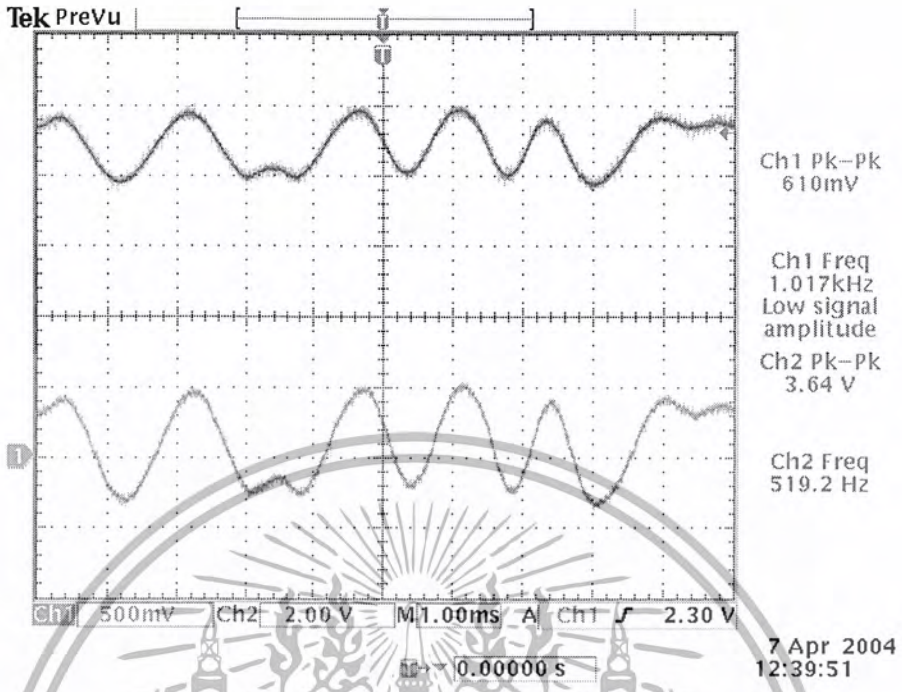
- อินพุทมี โรลออฟ = 0.7



รูปที่ 4.58 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.7 และเอาท์พุท(Ch2) ของตัวกรอง

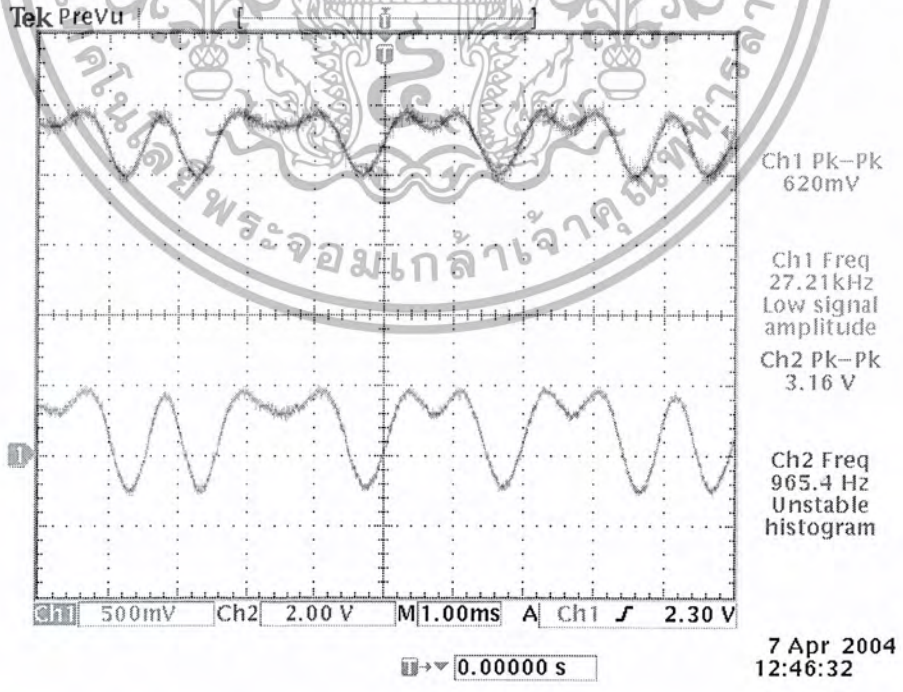
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อินพุทมี โรลออฟ = 0.8



รูปที่ 4.59 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.8 และเอาท์พุท(Ch2) ของตัวกรอง

- อินพุทมี โรลออฟ = 0.9

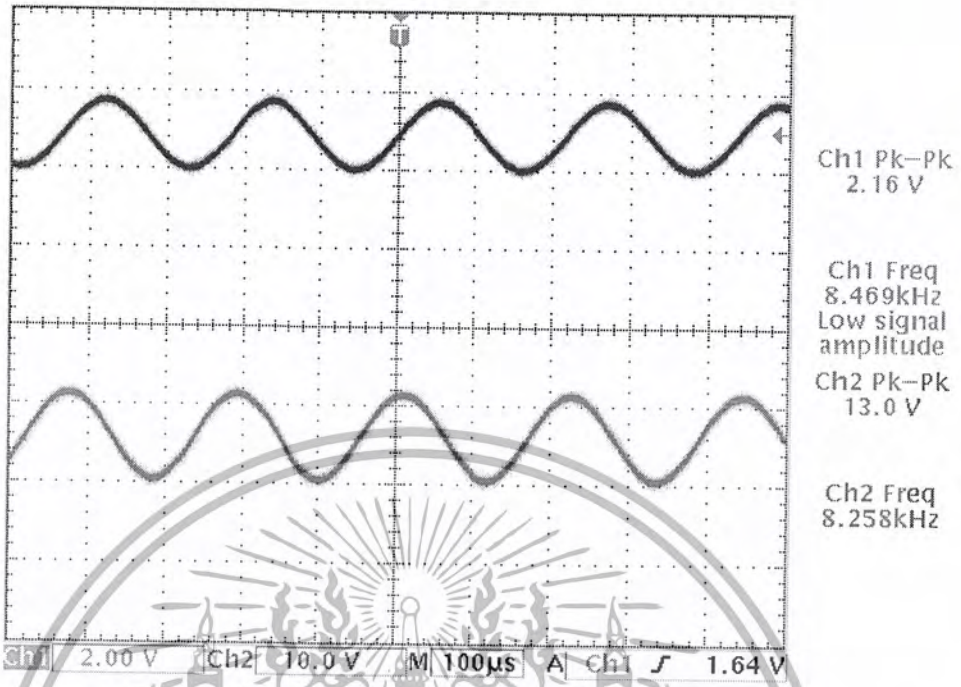


รูปที่ 4.60 สัญญาณอินพุท(Ch1) มีค่าโรลออฟ = 0.9 และเอาท์พุท(Ch2) ของตัวกรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

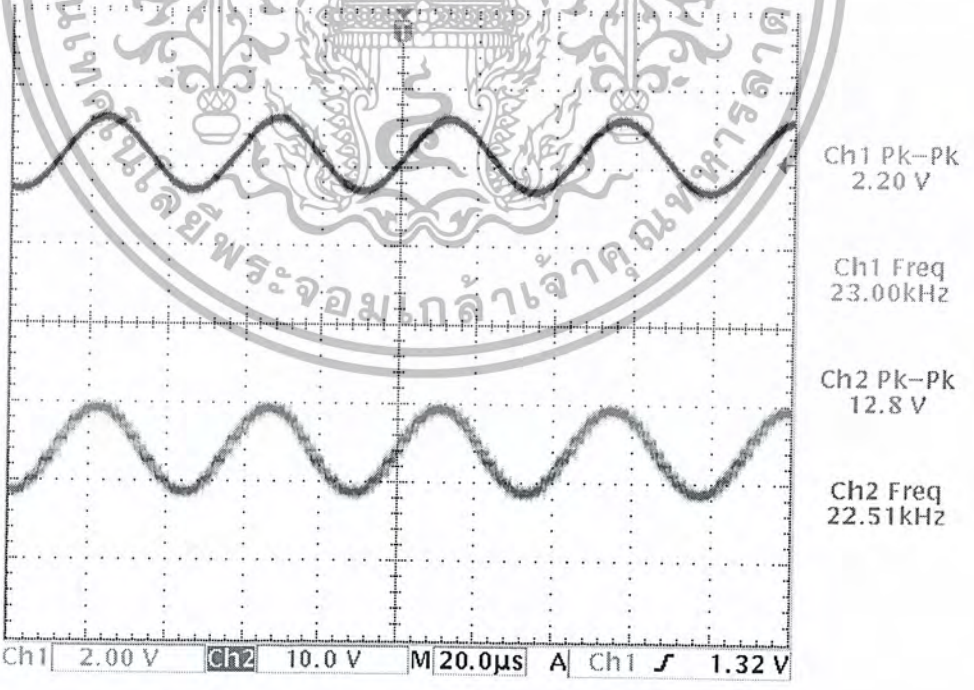
4.3.2 ป้อนสัญญาณรูปคลื่นไซน์ให้กับตัวกรอง

- ป้อนสัญญาณอินพุทที่ความถี่ประมาณ 8.5 kHz



รูปที่ 4.61 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch1) ที่เป็นสัญญาณรูปไซน์ความถี่ 8.5 kHzของตัวกรอง

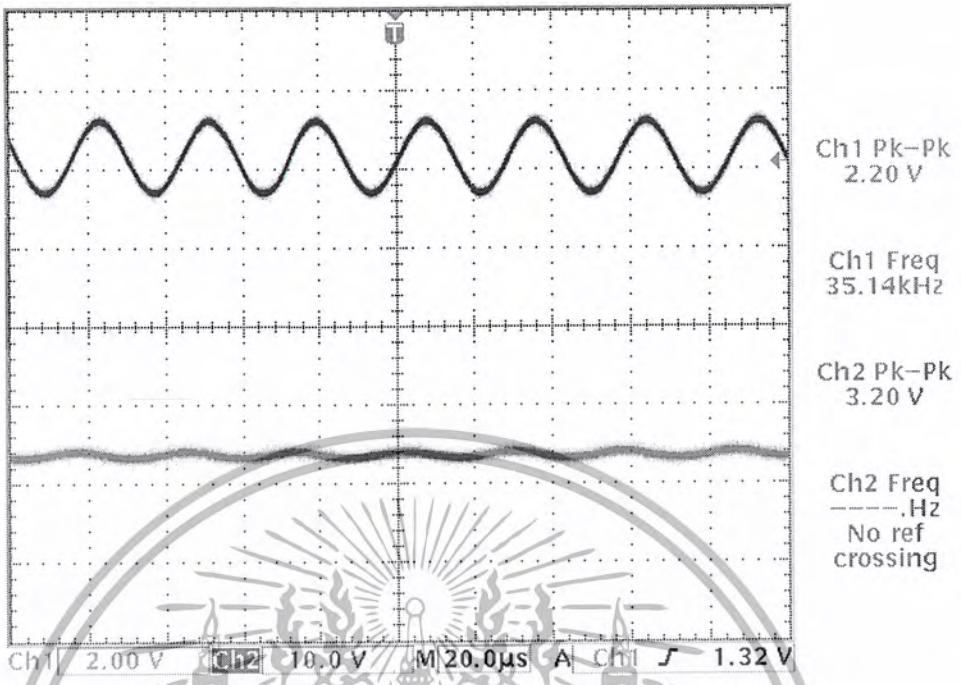
- ป้อนสัญญาณอินพุทที่ความถี่ประมาณ 23 kHz



รูปที่ 4.62 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch1) ที่เป็นสัญญาณรูปไซน์ความถี่ 23 kHzของตัวกรอง

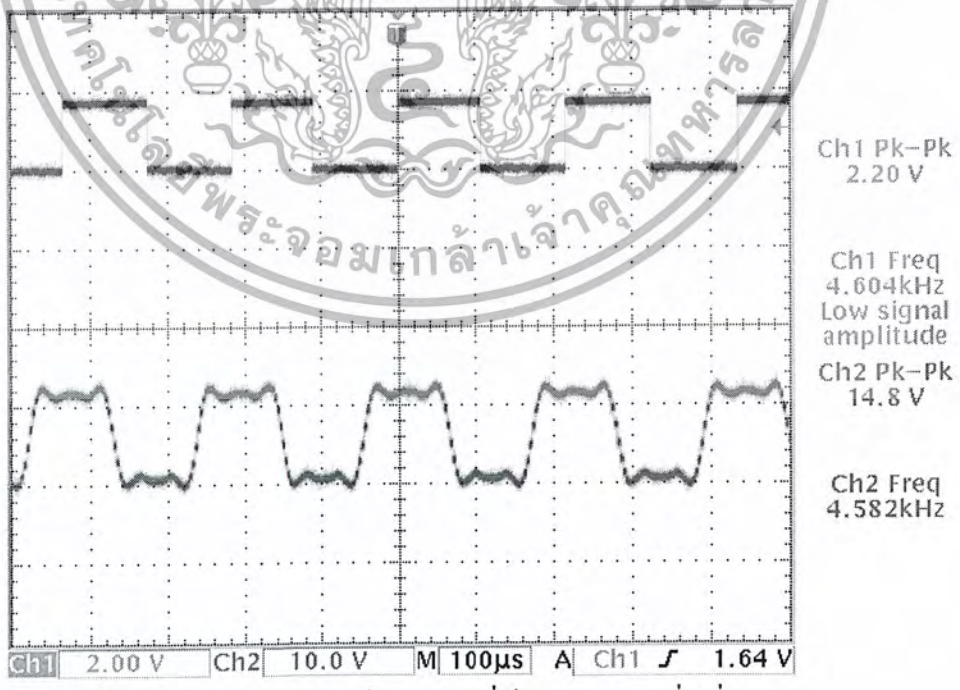
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ป้อนสัญญาณอินพุทที่ความถี่ประมาณ 35 kHz



รูปที่ 4.63 สัญญาณอินพุท(Ch1) และเอาท์พุท(Ch1) ที่เป็นสัญญาณรูปไซน์ความถี่ 35 kHzของตัวกรอง

4.3.3 ป้อนสัญญาณรูปคลื่นสี่เหลี่ยมให้กับตัวกรอง



รูปที่ 4.64 สัญญาณอินพุท(Ch 1) และเอาท์พุท(Ch 2) ที่เป็นสัญญาณรูปสี่เหลี่ยมของตัวกรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

สรุปผลการทดลอง

จากการทดลองและออกแบบวงจรแมทช์ฟิลเตอร์ โดยออกแบบเป็นชนิดรูทเรสต์โคไซน์ฟิลเตอร์ ทำให้มีความเข้าใจในการออกแบบฟิลเตอร์โดยใช้ทฤษฎีเลขคณิตกระจายและหลักการแก้ปัญหาการแทรกสอดทางสัญลักษณ์ การออกแบบแบ่งออกเป็น 2 ส่วนคือ

- ทางด้านฮาร์ดแวร์ ได้ใช้อุปกรณ์ FPGA ในการสร้างวงจรฟิลเตอร์ การทดสอบได้ใช้อุปกรณ์ ADC (Analog to Digital Circuit) ทำการแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิตอลป้อนให้กับอุปกรณ์ FPGA ทางด้านเอาท์พุทของอุปกรณ์ FPGA จะส่งผ่านไปยัง DAC (Digital to Analog Circuit) เพื่อทำการแปลงสัญญาณดิจิตอลออกเป็นสัญญาณอานาล็อก ในการดูเอาท์พุท

- ทางด้านซอฟแวร์ ได้อธิบายการทำงานของวงจรด้วยภาษา VHDL โดยออกแบบโดยใช้โปรแกรม MAX Plus II

ผลการทดลองที่ได้จากอุปกรณ์ FPGA เมื่อเปรียบเทียบกับการใช้โปรแกรม MATLAB จะแตกต่างกัน เมื่อป้อนสัญญาณเป็นแบบรูทเรสต์โคไซน์ เนื่องจากสัญญาณรูทเรสต์โคไซน์ที่ป้อนให้กับอุปกรณ์ FPGA ป้อนผ่านทางซาวน์การ์ด ทำให้มีการปรับออฟเซทให้กับวงจร ADC ได้ไม่ดีที่ควรสามารถแก้ปัญหาได้โดยสร้างวงจรเพื่อทำการปรับออฟเซทภายนอกหลังส่งออกจากซาวด์การ์ดแล้ว หรือสร้างรอมเก็บค่าของสัญญาณรูทเรสต์โคไซน์และป้อนให้กับอุปกรณ์ FPGA โดยตรง ไม่ต้องผ่าน ADC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม MATLAB

- โปรแกรมหาค่าสัมประสิทธิ์ของ Matched filter ชนิด Root-Raised cosine filter

```
clc;clear all;
fs = 270;
rolloff = 0.3;
n_T = [-3 3];
rate = 5;
T = 1;
rrcos = rcosfir(rolloff,n_T,rate,T ,'sqrt');
n = 0:1:length(rrcos)-1;
figure(1);stem(n,rrcos);grid;title('Coefficient 0f Root-Raised
cosine ');
% plot frequency response
[amp,q]=freqz(rrcos,1,512,fs);
figure(2);
plot(q,20*log10(abs(amp)));grid;
ylabel('Magnitude (dB)');
xlabel('Frequency in kHz');
 % plot phase response
figure(3);
p = unwrap(angle(amp));
plot(q,p*180/pi);
ylabel('Phase (degrees)');
xlabel('Frequency in kHz');
% plot pole/zero
figure(4);
[b,a] = eqtflength(rrcos,1);
[z,p,k] = tf2zp(b,a);
zplane(z,p);grid;
```

- โปรแกรมแปลงค่าสัมประสิทธิ์ให้เป็น 2'Complement เพื่อเก็บไว้ในตารางเปิดดู

```
clc;clear all;
fs = 270;
rolloff = 0.3;
n_T = [-3 3];
rate = 5;
T = 1;
rrcos = rcosfir(rolloff,n_T,rate,T ,'sqrt');
h = rrcos(1,1:16);

bit = [0 1];
c = 0;
n = 0;
for a1 = 1:1:2
for a2 = 1:1:2
for a3 = 1:1:2
for a4 = 1:1:2
    n = n+1;
    c = c+1;
    result(n) =
(h(4)*bit(a1))+(h(3)*bit(a2))+(h(2)*bit(a3))+(h(1)*bit(a4));
    f = 14;
    m = 16;
    if result(n)<0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        a(n) = round(result(n)*(2^f));
        aa(n) =(2^m)+a(n);
    else
        aa(n) = round(result(n)*(2^f));
    end
    bi = dec2bin(aa(n),m);
    fprintf(' "%s" \n',bi);
end;
end;
end;
end;
reply = input('Start new loop yes or no:','s');
if reply == 'y'
clc;
n = 0;
for a5 = 1:1:2
for a6 = 1:1:2
for a7 = 1:1:2
for a8 = 1:1:2
    n = n+1;
    c = c+1;
    result(n) =
(h(8)*bit(a5))+(h(7)*bit(a6))+(h(6)*bit(a7))+(h(5)*bit(a8));
    f = 14;
    m = 16;
    if result(n)<0
        a(n) = round(result(n)*(2^f));
        aa(n) =(2^m)+a(n);
    else
        aa(n) = round(result(n)*(2^f));
    end
    bi = dec2bin(aa(n),16);
    fprintf(' "%s" \n',bi);
end;
end;
end;
end;

reply1 = input('Start new loop yes or no:','s');
if reply1 == 'y'
clc;
n = 0;
for a9 = 1:1:2
for a10 = 1:1:2
for a11 = 1:1:2
for a12 = 1:1:2
    n = n+1;
    c = c+1;
    result(n) =
(h(12)*bit(a9))+(h(11)*bit(a10))+(h(10)*bit(a11))
+(h(9)*bit(a12));
    f = 14;
    m = 16;
    if result(n)<0
        a(n) = round(result(n)*(2^f));
        aa(n) =(2^m)+a(n);
    else
        aa(n) = round(result(n)*(2^f));
    end
    bi = dec2bin(aa(n),16);
    fprintf(' "%s" \n',bi);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
end;
end;
end;

reply2 = input('Start new loop yes or no:','s');
if reply2 == 'y'
clc;
n = 0;
for a13 = 1:1:2
for a14 = 1:1:2
for a15 = 1:1:2
for a16 = 1:1:2
    c = c+1;
    n = n+1;
    result(n) =
(h(16)*bit(a13))+(h(15)*bit(a14))+(h(14)*bit(a15))+(h(13)*bit(a
16));
    f = 14;
    m = 16;
    if result(n)<0
       a(n) = round(result(n)*(2^f));
       aa(n) =(2^m)+a(n);
    else
       aa(n) = round(result(n)*(2^f));
    end
    bi = dec2bin(aa(n),16);
    fprintf(' "%s" \n',bi);
end;
end;
end;
end;
else
end
else
end
else
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โปรแกรมป้อนสัญญาณ Root-Raised cosine ให้กับ Matched filter ชนิด Root-Raised cosine filter

```
clc;clear all;
fs = 270;
rolloff1 = 0.3;
n_T = [-3 3];rate = 5;T = 1;
rrcos = rcosfir(rolloff1,n_T,rate,T ,'sqrt');
%RRC signal
data = 1000;
x = randsrc(data,1,[],1245);
rolloff2 = 0.3;
fd = 1;fs = 5;
[ys,ts] = rcosine(fd,fs,'fir/sqrt',rolloff2);
[y1,t1] = rcosflt(x,fd,fs,'filter',ys);
snr = 20;
r = awgn(y1,snr,var(y1)); % add AWGN
%At Receiver
z1 = conv(y1,rrcos);
z2 = conv(r,rrcos);
figure(1);subplot(211);plot(y1);title('Root-Raised cosine
signal');grid;
subplot(212);plot(z1);title('Output of Root-Raised cosine
filter');grid;
N = 10;
e0 = eyediagram(y1,N);
set(e0,'Name','Eye Diagram Input of  Filter');
e1 =  eyediagram(z1,N);
set(e1,'Name','Eye Diagram Output of Filter');
figure(4);subplot(211);plot(r);title('Root-Raised cosine
signal');grid;
subplot(212);plot(z2);title('Output of Root-Raised cosine
filter');grid;
N = 10;
e2 = eyediagram(y1,N);
set(e2,'Name','Eye Diagram Input of  Filter');
e3 =  eyediagram(z1,N);
set(e3,'Name','Eye Diagram Output of  Filter');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โปรแกรมป้อนสัญญาณ ไซน์ให้กับ Matched filter ชนิด Root-Raised cosine filter

```
clc;clear all;
fs = 270;
rolloff1 = 0.3;
n_T = [-3 3];
rate = 5;
T = 1;
rrcos = rcosfir(rolloff1,n_T,rate,T ,'sqrt');
f1 = 20;
f2 = 50;
N_point = 100;
tx1 = [0:N_point]/(fs*1000);
x1 =  sin(2*pi*f1*1000*tx1);
x2 = 0.5*sin(2*pi*f2*1000*tx1)+x1;
y = conv(x2,rrcos);
figure(1);subplot(211);plot(x2);title('Input of filter');
subplot(212);plot(y);title('Output of filter')
% plot  Spectrum
spec_x=fft(x2,1024);
spec_y=fft(y,1024);
spectrum_x=abs(spec_x);
spectrum_y=abs(spec_y);
fre=[1:512];
frequency=fre*fs/1024;
figure(2);
subplot(2,1,1);
plot(frequency,spectrum_x(fre));grid;
xlabel('Frequency in kHz');
ylabel('Input Spectrum');
subplot(2,1,2);
plot(frequency,spectrum_y(fre));grid;
xlabel('Frequency in kHz');
ylabel('Output Spectrum');
```

- โปรแกรมป้อนสัญญาณ สี่เหลี่ยม ให้กับ Matched filter ชนิด Root-Raised cosine filter

```
clc;clear all;
fs = 270;
rolloff1 = 0.3;
n_T = [-3 3];
rate = 5;
T = 1;
rrcos = rcosfir(rolloff1,n_T,rate,T ,'sqrt');
T = 100;
bits = ['1', '0', '1', '0', '1', '1'];
signal1 = [ones(1,T/2),zeros(1,T/2)];
signal0 = -1*signal1;
x1 = [];
for n=1:length(bits)
      x1=[x1  eval(strcat('signal',bits(n)))];
end
y = conv(x1,rrcos);
figure(1);subplot(211);plot(x1);title('Input of filter');
subplot(212);plot(y);title('Output of filter')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โปรแกรมส่งสัญญาณออกทางซาวน์การ์ด

```
clc;clear all;
fs = 5;
fd = 1;
rolloff = 0.3;
data = 1000;
x = randsrc(data,1,[],1245);
[ys,ts] = rcosine(fd,fs,'fir/sqrt',0.3);
[yc,tc] = rcosflt(x,fd,fs,'filter',ys);
snr = 20;
r = awgn(yc,snr,var(yc)); % add AWGN
y = yc+max(yc);
rm = r+max(r);
soundsc(y,4*fs)
```

โปรแกรม VHDL

- อุปกรณ์ DIVIDER

```
library ieee;
use ieee.std_logic_1164.all;

entity div is
port(Clk_in : in std_logic;
     Clk_out : out std_logic);
end ;

architecture rtl of div is
begin
     process(Clk_in)
     variable Clk_temp : std_logic := '0';
     variable count : integer := 0 ;
     begin
       if Clk_in'Event and Clk_in='1'then
          if count<0 then
          count := count+1;
          Clk_temp := Clk_temp;
          Else
          count := 0;
          Clk_temp := not(Clk_temp);
          end if;
          Clk_out <= Clk_temp;
       end if;
    end process;
end rtl;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อุปกรณ์ CONTROL

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity control_sys is
port(sys_clk : in std_logic;
     clk,load1,load2,load_b,load_ad,send,clear,add_sub : out
std_logic);
end;

architecture rtl of control_sys is
signal state : integer range 0 to 37;
begin
 process(sys_clk)
  begin
   if sys_clk'event and sys_clk = '1' then
     if state /= 37 then
        state <= state+1;
     else state <= 0;
     end if;
     if (state=0) then
        clk <= '0';
        load1 <= '1';
        load2 <= '0';
        load_b <= '0';
        load_ad <= '1';
        send <= '0';
        clear <= '0';
        add_sub <= '0';
     elsif (state=1) then
        clk <= '0';
        load1 <= '0';
        load2 <= '1';
        load_b <= '0';
        load_ad <= '1';
        send <= '0';
        clear <= '0';
        add_sub <= '0';
     elsif (state=2) then
        clk <= '0';
        load1 <= '0';
        load2 <= '0';
        load_b <= '0';
        load_ad <= '1';
        send <= '0';
        clear <= '1';
        add_sub <= '0';
     elsif
(state=3)or(state=7)or(state=8)or(state=11)or(state=12)or(state
=15)or(state=16)then
        clk <= '0';
        load1 <= '0';
        load2 <= '0';
        load_b <= '0';
        load_ad <= '1';
        send <= '0';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            clear <= '0';
            add_sub <= '1';
        elsif (state=4) then
            clk <= '0';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '1';
            send <= '1';
            clear <= '0';
            add_sub <= '1';
        elsif (state=5)or(state=9)or(state=13)or(state=17) then
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '1';
            send <= '0';
            clear <= '0';
            add_sub <= '1';
        elsif (state=6)or(state=10)or(state=14) then
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '1';
            send <= '1';
            clear <= '0';
            add_sub <= '1';
        elsif (state=18)or(state=22)or(state=26)or(state=30) then
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '0';
            send <= '1';
            clear <= '0';
            add_sub <= '1';
        elsif
(state=19)or(state=20)or(state=23)or(state=24)or(state=27)or(st
ate=28) then
            clk <= '0';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '0';
            send <= '0';
            clear <= '0';
            add_sub <= '1';
        elsif (state=21)or(state=25)or(state=29) then
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            send <= '0';
            clear <= '0';
            add_sub <= '1';
        elsif (state=22)or(state=26)or(state=30) then
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            load_b <= '0';
            load_ad <= '0';
            send <= '0';
            clear <= '0';
            add_sub <= '1';
        elsif (state=31)or(state=32) then
            clk <= '0';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '0';
            send <= '0';
            clear <= '0';
            add_sub <= '0';
        elsif (state=33) then
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '0';
            send <= '0';
            clear <= '0';
            add_sub <= '0';
        elsif (state=34) then
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '0';
            send <= '1';
            clear <= '0';
            add_sub <= '0';
        elsif (state=35) then
            clk <= '0';
            load1 <= '0';
            load2 <= '0';
            load_b <= '1';
            load_ad <= '0';
            send <= '0';
            clear <= '0';
            add_sub <= '0';
        else
            state <= 0;
            clk <= '1';
            load1 <= '0';
            load2 <= '0';
            load_b <= '0';
            load_ad <= '0';
            send <= '0';
            clear <= '0';
            add_sub <= '0';
        end if;
    end if;
  end process;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อุปกรณ์ PIPO

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity pipo_n is
port(LD : in std_logic;
d_in : in std_logic_vector(7 downto 0);
d1_out,d2_out,d3_out,d4_out,d5_out,d6_out,d7_out,d8_out : out
std_logic_vector(8 downto 0);
d9_out,d10_out,d11_out,d12_out,d13_out,d14_out,d15_out : out
std_logic_vector(8 downto 0);                d16_out : out
std_logic_vector(8 downto 0));
end;

architecture rtl of pipo_n is
signal p1h,p2h,p3h,p4h,p5h,p6h,p7h,p8h,p9h: std_logic_vector(7
downto 0);
signal p10h,p11h,p12h,p13h,p14h,p15h,p16h : std_logic_vector(7
downto 0);
signal p17h,p18h,p19h,p20h,p21h,p22h,p23h: std_logic_vector(7
downto 0);
signal p24h,p25h,p26h,p27h,p28h,p29h,p30h,p31h :
std_logic_vector(7 downto 0);
begin
 process(LD)
  begin
   if LD = '1' then
      p31h <= p30h; p30h <= p29h; p29h <= p28h;
      p28h <= p27h; p27h <= p26h; p26h <= p25h;
      p25h <= p24h; p24h <= p23h; p23h <= p22h;
      p22h <= p21h; p21h <= p20h; p20h <= p19h;
      p19h <= p18h; p18h <= p17h; p17h <= p16h;
      p16h <= p15h; p15h <= p14h; p14h <= p13h;
      p13h <= p12h; p12h <= p11h; p11h <= p10h;
      p10h <= p9h;  p9h <= p8h;   p8h <= p7h;
      p7h <= p6h;   p6h <= p5h;   p5h <= p4h;
      p4h <= p3h;   p3h <= p2h;   p2h <= p1h;
      p1h <= d_in;
   end if;
 end process;
   d1_out <= ('0' & p1h)+p31h;   d2_out <= ('0' & p2h)+p30h;
d3_out <= ('0' & p3h)+p29h;
   d4_out <= ('0' & p4h)+p28h;   d5_out <= ('0' & p5h)+p27h;
d6_out <= ('0' & p6h)+p26h;
   d7_out <= ('0' & p7h)+p25h;   d8_out <= ('0' & p8h)+p24h;
d9_out <= ('0' & p9h)+p23h;
   d10_out <= ('0' & p10h)+p22h; d11_out <= ('0' & p11h)+p21h;
   d12_out <= ('0' & p12h)+p20h; d13_out <= ('0' & p13h)+p19h;
   d14_out <= ('0' & p14h)+p18h;  d15_out <= ('0' & p15h)+p17h;
   d16_out <= ('0' & p16h);
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อุปกรณ์ SHIFT

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity shift is
port(clk :in std_logic;
     Lm : in std_logic;
     d1in,d2in,d3in,d4in,d5in,d6in,d7in,d8in,d9in : in
std_logic_vector(8 downto 0);
     d10in,d11in,d12in,d13in,d14in,d15in,d16in : in
std_logic_vector(8 downto 0);
     d1out,d2out,d3out,d4out,d5out,d6out,d7out,d8out,d9out :
out std_logic;
     d10out,d11out,d12out,d13out,d14out,d15out,d16out : out
std_logic);
end;

architecture rtl of shift is
signal s1h,s2h,s3h,s4h,s5h,s6h,s7h,s8h,s9h : std_logic_vector(8
downto 0);
signal s10h,s11h,s12h,s13h,s14h,s15h,s16h : std_logic_vector(8
downto 0);
begin
 process(clk,Lm)
  begin
   if Lm = '1' then
    s1h <= d1in;   s2h <= d2in;   s3h <= d3in;   s4h <= d4in;
    s5h <= d5in;   s6h <= d6in;   s7h <= d7in;   s8h <= d8in;
    s9h <= d9in;   s10h <= d10in; s11h <= d11in; s12h <= d12in;
    s13h <= d13in; s14h <= d14in; s15h <= d15in; s16h <= d16in;
   elsif clk'event and clk = '1' then
    s1h(7 downto 0) <= s1h(8 downto 1);   s2h(7 downto 0) <=
s2h(8 downto 1);
    s3h(7 downto 0) <= s3h(8 downto 1);   s4h(7 downto 0) <=
s4h(8 downto 1);
    s5h(7 downto 0) <= s5h(8 downto 1);   s6h(7 downto 0) <=
s6h(8 downto 1);
    s7h(7 downto 0) <= s7h(8 downto 1);   s8h(7 downto 0) <=
s8h(8 downto 1);
    s9h(7 downto 0) <= s9h(8 downto 1);   s10h(7 downto 0) <=
s10h(8 downto 1);
    s11h(7 downto 0) <= s11h(8 downto 1); s12h(7 downto 0) <=
s12h(8 downto 1);
    s13h(7 downto 0) <= s13h(8 downto 1); s14h(7 downto 0) <=
s14h(8 downto 1);
    s15h(7 downto 0) <= s15h(8 downto 1); s16h(7 downto 0) <=
s16h(8 downto 1);
   end if;
end process;
    d1out <= s1h(0);   d2out <= s2h(0);   d3out <= s3h(0);
d4out <= s4h(0);
    d5out <= s5h(0);   d6out <= s6h(0);   d7out <= s7h(0);
d8out <= s8h(0);
    d9out <= s9h(0);   d10out <= s10h(0); d11out <= s11h(0);
d12out <= s12h(0);
    d13out <= s13h(0); d14out <= s14h(0); d15out <= s15h(0);
d16out <= s16h(0);
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อุปกรณ์ ROM

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity rom_n is
   port(addr1,addr2,addr3,addr4 :in std_logic_vector(3 downto
0);
        data_out                :out std_logic_vector(15 downto
0));
end;

architecture rtl of rom_n is
   signal d1,d2,d3,d4 : std_logic_vector(15 downto 0);
begin
   d1 <= "0000000000000000" when addr1 = "0000" else
 "1111111100001101" when addr1 = "0001" else
 "1111111110010110" when addr1 = "0010" else
 "1111111010100100" when addr1 = "0011" else
 "0000000010101101" when addr1 = "0100" else
 "1111111110111010" when addr1 = "0101" else
 "0000000001000011" when addr1 = "0110" else
 "1111111101010001" when addr1 = "0111" else
 "0000000111010101" when addr1 = "1000" else
 "0000000011100011" when addr1 = "1001" else
 "0000000101101100" when addr1 = "1010" else
 "0000000001111001" when addr1 = "1011" else
 "0000001010000010" when addr1 = "1100" else
 "0000000110010000" when addr1 = "1101" else
 "0000001000011001" when addr1 = "1110" else
 "0000000100100110" when addr1 = "1111" ;

d2 <= "0000000000000000" when addr2 = "0000" else
 "0000001001011000" when addr2 = "0001" else
 "0000000110011111" when addr2 = "0010" else
 "0000001111110111" when addr2 = "0011" else
 "1111111110010010" when addr2 = "0100" else
 "0000000111101010" when addr2 = "0101" else
 "0000000100110001" when addr2 = "0110" else
 "0000001110001001" when addr2 = "0111" else
 "1111100011010111" when addr2 = "1000" else
 "1111111001011111" when addr2 = "1001" else
 "1111111001110101" when addr2 = "1010" else
 "0000000011001110" when addr2 = "1011" else
 "1111110001101001" when addr2 = "1100" else
 "1111111011000001" when addr2 = "1101" else
 "1111111000000111" when addr2 = "1110" else
 "0000000001100000" when addr2 = "1111" ;

d3 <= "0000000000000000" when addr3 = "0000" else
 "1111101010110110" when addr3 = "0001" else
 "1111101010101111" when addr3 = "0010" else
 "1111010101100101" when addr3 = "0011" else
 "1111110111011010" when addr3 = "0100" else
 "1111100010010000" when addr3 = "0101" else
 "1111100010001010" when addr3 = "0110" else
 "1111001101000000" when addr3 = "0111" else
 "0000010001010110" when addr3 = "1000" else
 "1111111100001100" when addr3 = "1001" else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   "1111111100000110" when addr3 = "1010" else
   "1111100110111100" when addr3 = "1011" else
   "0000001000110001" when addr3 = "1100" else
   "1111110011100111" when addr3 = "1101" else
   "1111110011100000" when addr3 = "1110" else
   "1111011110010110" when addr3 = "1111" ;

 d4 <= "0000000000000000" when addr4 = "0000" else
   "0000110100001100" when addr4 = "0001" else
   "0001010111101001" when addr4 = "0010" else
   "0010001011110101" when addr4 = "0011" else
   "0001110010000110" when addr4 = "0100" else
   "0010100110010010" when addr4 = "0101" else
   "0011001001101111" when addr4 = "0110" else
   "0011111101111011" when addr4 = "0111" else
   "0001111011111000" when addr4 = "1000" else
   "0010110000000100" when addr4 = "1001" else
   "0011010011100001" when addr4 = "1010" else
   "0100000111101101" when addr4 = "1011" else
   "0011101101111101" when addr4 = "1100" else
   "0100100010001010" when addr4 = "1101" else
   "0101000101100111" when addr4 = "1110" else
   "0101111001110011" when addr4 = "1111" ;

 data_out <= d1+d2+d3+d4;
 end;
```

- อุปกรณ์ ADD

```
 library ieee;
 use ieee.std_logic_1164.all;
 use ieee.std_logic_unsigned.all;

 entity add is
    port(add_in  : in std_logic_vector(15 downto 0);
         add_sub,clear,send,ir : in std_logic;
         d_out : out std_logic_vector(15 downto 0));
 end;

 architecture rtl of add is
    signal s : std_logic_vector(16 downto 0);
    signal sum,add1,do : std_logic_vector(15 downto 0);
 begin

 a_dd : process(add_sub,add_in)
       variable s_sub : std_logic_vector(15 downto 0);
       begin
       if add_sub = '1' then
          s <= ('0' & add_in)+add1;
          sum(15) <= s(16) xor add_in(15) xor add1(15);
          sum(14 downto 0) <= s(15 downto 1);
       else
          s_sub := (add_in xnor "0000000000000000");
          s_sub := s_sub+1;
          sum(15 downto 0) <= add1(15 downto 0)+s_sub(15 downto
 0);
       end if;
       end process a_dd;

 a_cc : process(clear,send)
       begin
       if clear = '1' then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            add1 <= (others => '0');
        elsif send'event and send = '1' then
            do <= sum;
            add1 <= sum;
        end if;
        end process a_cc;

buff : process(lr)
        begin
        if lr = '1' then
             d_out(0)<= '0';
           d_out(15 downto 1) <= do(14 downto 0);
        end if;
        end process buff;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลุล่วงได้ด้วยดีด้วยความกรุณาของ อาจารย์อัครพล ตรีรัตน์ อาจารย์ที่ปรึกษา อาจารย์ศรวัฒน์ ชิวปรีชา ผู้คอยเอาใจใส่ดูแลและเมตตาแก่ศิษย์ด้วยดีเสมอมา ขอขอบคุณสำหรับกำลังใจและให้คำแนะนำเกี่ยวกับปัญหาต่างๆระหว่างการทำงานนี้ และขอขอบพระคุณอาจารย์ทุกท่านๆที่ประสิทธิ์ประสาทวิชาความรู้ต่างๆให้แก่ศิษย์ทั้งโดยทางตรงและทางอ้อม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

[1] วิวัฒน์ กิรานนท์ "วิศวกรรมการสื่อสาร"สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง พิมพ์ครั้งที่ 2 พ.ศ.2542

[2] สัญฉกร วุฒิสิทธิกุลกิจ "หลักการระบบโทรศัพท์เคลื่อนที่" สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย พ.ศ.2546

[3]รองศาสตราจารย์ สมศักดิ์ มิตะถา"การออกแบบวงจรดิจิตอลและวงจรตรรก"สถาบันเทคโนโลยีเจ้าคุณทหารลาดกระบัง พ.ศ.2543

[4] John G.Proakis , "Digital Communications,"Mc Graw – Hill, Inc , 1995

[5] Marvin E.Frerking , "Digital Signal Processing in Communication System,"Van Noistrand Reinhold,1994

[6] Bernard Sklar, "Digital Communication,"Prentice Hall Inc,2001

[7] ALTERA Data Book,1998

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้