สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การแปลงเวฟเลตโดยใช้บอร์ด FPGA

Wavelet Transform Using FPGA Board

โดย

นาย พิทยา ภาษี

นาย ไพศาล เตชศรีอาภา

นาย มงคล แซ่จัง

เลขหมู่..............................

เลขทะเบียน....62428....

วัน,เดือน,ปี...18 ส.ค. 2549...

[redacted] 11624851

ม..............................

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

ผ่านการตรวจชิ้นงานแล้ว

(ลงชื่อ)..............................ผู้ตรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผ่านการตรวจรูปเล่มแล้ว

(ลงชื่อ)..............................ผู้ตรวจ

การแปลงเวฟเลตโดยใช้อุปกรณ์ FPGA

Wavelet Transform using FPGA

โดย

นาย พิทยา	ภาษี	45010534
นาย ไพศาล	เตชศรีอาภา	45010565
นาย มงคล	แซ่จัง	45010599

อาจารย์ที่ปรึกษา

ผศ. อัครพล ตรีรัตน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **การแปลงเวฟเลตโดยใช้อุปกรณ์ FPGA**

Wavelet Transform using FPGA

ผู้จัดทำ

นาย พิทยา ภามี	**45010534**
นาย ไพศาล เตชศรีอาภา	**45010565**
นาย มงคล แซ่จัง	**45010599**

อาจารย์ที่ปรึกษา

(ผศ. อัครพล ตรีรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแปลงเวฟเลตโดยใช้อุปกรณ์ FPGA

Wavelet Transform using FPGA

โดย นาย พิทยา ภาษี 45010534
นาย ไพศาล เตชศรีอาภา 45010565
นาย มงคล แซ่จัง 45010599

อาจารย์ที่ปรึกษา ผศ.อัครพล ตรีรัตน์

บทคัดย่อ

โครงงานนี้เป็นการศึกษาทฤษฎีและเรียนรู้หลักการการแปลงเวฟเลต โดยใช้โครงสร้างของฟิลเตอร์แบงค์ ซึ่งทำการศึกษาโดยการจำลองการทำงานผ่านโปรแกรมแมทแลบ (MATLAB) พร้อมทั้งศึกษาหลักการของลิฟท์ติ้ง (Lifting scheme) เพื่อนำไปออกแบบวงจรลงบน FPGA (Field Programmable Gate Array) โดยใช้ภาษา VHDL (Very high speed integrate circuit Hardware Description Language)

ABSTRACT

This project studies the principle of wavelet transform using filter bank structure which is an efficient tool for signal analysis. We will use MATLAB programming to obtain all the relevant results and compare to the theory. We also study the principle of Lifting Scheme for implementing to FPGA. (Field Programmable Gate Array) using VHDL(Very high speed integrate circuit Hardware Description Language)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานฉบับนี้สำเร็จลุล่วงได้ด้วยดีโดยได้รับความช่วยเหลือและคำชี้แนะจากหลายท่าน ผู้จัดทำขอขอบพระคุณอาจารย์ที่ปรึกษา ผศ. อัครพล ตรีรัตน์ ที่ให้คำปรึกษาและให้ความช่วยเหลือด้านข้อมูล, อุปกรณ์ในการทำปริญญานิพนธ์เป็นอย่างดี ขอขอบพระคุณ Prof. Soontorn Oraintara , University of Texas at Arlington , USA ที่ได้มาบรรยายเรื่อง Wavelet Transform และขอขอบคุณนายคงฤทธิ์ เตชะวชิรศิริ ที่ให้คำปรึกษาและให้ความช่วยเหลือในด้านการใช้งานเอฟพีจีเอ ผู้เขียนระลึกอยู่เสมอว่าหากไม่ได้รับความช่วยเหลือจากบุคคลที่กล่าวมานั้น รายงานฉบับนี้ก็ไม่สามารถที่จะสำเร็จลุล่วงไปด้วยดี จึงขอขอบพระคุณมา ณ ที่นี้

นายพิทยา ภาษี
นายไพศาล เตชศรีอาภา
นายมงคล แซ่จัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 บทนำ 1

1.1 ความเป็นมาของปริญญานิพนธ์ 1

1.2 วัตถุประสงค์ของปริญญานิพนธ์ 1

1.3 ขอบเขตของปริญญานิพนธ์ 2

1.4 เนื้อหาของปริญญานิพนธ์ 2

บทที่ 2 ทฤษฎีและหลักการ

2.1 ความรู้เบื้องต้นของเวฟเลต 3

2.1.1คุณสมบัติโดยทั่วไปของระบบของเวฟเลต 3

2.1.2 คุณสมบัติพิเศษของเวฟเลต 4

2.1.3 การกระจายเวฟเลตและการแปลงเวฟเลต 4

2.2 การแปลงเวฟเลตแบบไม่ต่อเนื่อง 5

2.3 ทฤษฎีมัลติรีโซลูชัน 6

2.3.1 สเกลลิ่งฟังก์ชัน 6

2.3.2 เวฟเลตฟังก์ชัน 7

2.4 ฟิลเตอร์แบงค์และการแปลงเวฟเลต 8

2.4.1 หลักการวิเคราะห์ 8

2.4.1.1 การกรองสัญญาณและการสุ่มค่าตัวอย่าง 9

2.4.2 หลักการสังเคราะห์ 13

2.4.2.1การกรองสัญญาณและการเพิ่มค่าตัวอย่าง 14

2.5 ฟิลเตอร์แบงค์ 15

2.5.1 การกำจัดการเกิดการซ้อนทับกันและผลที่ได้จากวงจรกรองความถี่ 18

2.5.2 โครงสร้างของวงจรกรองความถี่ทั้งรูปแบบโดยตรง 19

2.5.2.1 วงจรกรองความถี่ที่มีโครงสร้างแบบโดยตรง 19

2.5.2.2 วงจรกรองความถี่ที่มีโครงสร้างแบบโพลีเฟส 21

2.6 โครงสร้างการแปลงเวฟเลตโดยใช้วิธีการ Lifting scheme 25

2.7 ภาษาวีเอชดีแอล 28

2.7.1 ประวัติความเป็นมาของภาษาวีเอชดีแอล 28

2.7.2 การออกแบบระบบดิจิตอล 29

2.7.3 การออกแบบจากบนลงล่าง 30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

2.8 เอฟพีจีเอ 32

2.8.1 การออกแบบวงจรเชิงเลขด้วยชิพอุปกรณ์เอฟพีจีเอ 33

2.8.2 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว 33

2.9 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย 34

2.9.1 ระบบตัวเลข 34

2.9.1.1 รูปแบบจำนวนโดยตรง 34

2.9.1.2 รูปแบบจำนวนอิงดรรชนี 36

2.9.2 ทฤษฎีเลขคณิตกระจาย 37

2.10 วงจรแปลงอะนาลอกเป็นดิจิตอล (Analog to Digital Converter: A/D) 42

2.10.1 แบบใช้วงจรเปรียบเทียบขนาน หรือแบบแฟลซ 42

2.10.2 วงจรเอทูดีที่ใช้การอินทีเกรต 43

2.10.3 วงจรเปลี่ยนสัญญาณเอทูดีที่ใช้วงจรนับและวงจรดีทูเอประกอบกัน 45

2.10.3.1 แบบบวงจรนับเดี่ยว (Single Counter) 45

2.10.4 วงจรเปลี่ยนสัญญาณเอทูดีแบบใช้การประมาณค่า 45

2.10.5 การสุ่มและการคงค่า (Sample and Holds) 45

2.11 พอร์ตอนุกรม 46

2.11.1 การสื่อสารข้อมูล 46

2.11.1.1 การสื่อสารข้อมูลแบบขนาน (Parallel Communication) 46

2.11.1.2 การสื่อสารข้อมูลแบบอนุกรม (Serial Communication) 46

2.11.2 การอินเตอร์เฟสตามมาตรฐาน RS-232 47

2.11.3 การเชื่อมต่อระหว่าง DB-9 กับ FPGA 48

บทที่ 3 การคำนวณและการสร้าง 50

3.1 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL ในการออกแบบ 50

3.1.1 สัมประสิทธิ์ที่นำมาใช้ในการแปลงเวฟเลต โดยเลือกใช้สัมประสิทธิ์ของตัวกรองสัญญาณแบบ 4 tap orthonormal filter with two vanishing moments (D4) 50

3.1.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ 3 ระดับ 51

3.1.3 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ 3 ระดับ 52

3.1.4 ส่วนประกอบภายในของการแปลงเวฟเลตโดยใช้วิธี Lifting scheme 53

3.1.4.1 วงจรหน่วงเวลา 53

3.1.4.2 วงจรสุ่มค่าตัวอย่าง 53

3.1.4.3 วงจรเพิ่มค่าตัวอย่าง 53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4.4 วงจรบวก 53
3.1.4.5 วงจรคูณ 54
3.1.5 การประยุกต์ใช้งาน DA (Distributed Arithmetic) กับการแปลงเวฟเลต ที่มีโครงสร้างแบบ lifting scheme 55
3.1.5.1 วงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม 56
3.1.5.2 หน่วยความจำ 56
3.1.5.3 วงจรบวกหรือวงจรลบ 57
3.1.5.4 ส่วนเก็บค่าผลการคำนวณ 57
3.1.6 ส่วนของการออกแบบวงจรหารความถี่สัญญาณนาฬิกา 57
3.1.7 ส่วนของการเก็บสัมประสิทธิ์ scaling และ wavelet ที่ได้ จากการวิเคราะห์ ในแต่ละระดับ 58
3.1.7.1 วงจรควบคุมการทำงานของ RAM 58
3.1.7.2 Random Access Memory (RAM) 59
3.2 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA 59
3.2.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter) 59
3.2.2 วงจรมัลติเพล็กซ์ (Multiplex) 60
3.2.3 ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยัง คอมพิวเตอร์ 60
3.2.4 ส่วนของการรับบิตข้อมูลจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ผ่านทางพอร์ตอนุกรม 61

บทที่ 4 การทดลองและผลการทดลอง 63
4.1 ส่วนของการ Simulate โดยใช้โปรแกรม MATLAB 63
4.1.1 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ 63
4.1.2 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ 63
4.1.3 ผลการทดลอง 64
4.1.3.1 ส่วนของ interface 64
4.1.3.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ เมื่อสัญญาณอินพุทเป็นข้อมูล 69
4.1.3.3 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์เมื่ออินพุทเป็นข้อมูล 70
4.1.4 ผลการทดลองเมื่อใส่สัญญาณอินพุทเป็นสัญญาณเสียง 72
4.2 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL 76
4.2.1. ส่วนของการแปลงเวฟเลตในส่วนวิเคราะห์ 76
4.2.1.1 ส่วนของวงจรสุ่มค่าตัวอย่าง 76
4.2.1.2 วงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม 77
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.3 หน่วยความจำ 78
4.2.1.4 ส่วนของวงจรหารความถี่สอง 79
4.2.1.5 ส่วนของวงจรบวกสะสม 14 79
4.2.1.6 ส่วนของวงจรบวกขนาด 14 บิต 80
4.2.1.7 ส่วนของวงจรควบคุม State 81
4.2.1.8 ส่วนของวงจรควบคุมและวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 82
4.2.1.9 ส่วนของวงจรควบคุมและวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 83
4.2.1.10 ส่วนของวงจรควบคุมและวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 84
4.2.1.11 ส่วนของวงจรแทรกบิต 85
4.2.1.12 ส่วนของวงจรควบคุมหน่วยความจำ 85
4.2.1.13 ส่วนของวงจรมัลติเพล็กซ์ 86
4.2.1.14 ส่วนของวงจรตัดบิต 87
4.2.1.15 ส่วนของวงจรชิฟบิต 88
4.2.2 ส่วนของวงจรส่งบิตข้อมูลจากบอร์ด FPGA ออกทางพอร์ตอนุกรม 88
4.2.2.1 ส่วนของวงจรหารความถี่ 88
4.2.2.2 ส่วนของวงจรส่งบิตข้อมูลออกทางพอร์ตอนุกรม 89
4.3 ส่วนของการแสดงผลผ่านจอคอมพิวเตอร์ 90
4.4 ภาพอุปกรณ์ที่ใช้ในการแปลงเวฟเลตโดยใช้บอร์ด FPGA 92

บทที่ 5 บทวิจารณ์และบทสรุป 93
กิตติกรรมประกาศ
หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 แสดงลักษณะของคลื่นเปรียบเทียบกับเวฟเลต 3

รูปที่ 2.2 การเลื่อนตำแหน่งและการปรับระดับของเวฟเลต 5

รูปที่ 2.3 การแบ่งพื้นที่ออกเป็นส่วนย่อยโดยการใช้สเกลลิ่งฟังก์ชัน 6

รูปที่ 2.4 สเกลลิ่งฟังก์ชันและเวฟเลต 7

รูปที่ 2.5 ตัวสุ่มค่าตัวอย่าง (Down Sampler or Decimator) 9

รูปที่ 2.6 สเปกตรัมของสัญญาณ $X(\omega)$ เมื่อสุ่มสัญญาณด้วยความถี่ f_s 9

รูปที่ 2.7 สเปกตรัมของสัญญาณ $X(\omega)$ เมื่อสุ่มสัญญาณด้วยความถี่ $2f_s$ 10

รูปที่ 2.8 สเปกตรัมของสัญญาณอินพุตก่อนการ downsampling 11

รูปที่ 2.9 สเปกตรัมของสัญญาณเอาท์พุตหลังจากผ่านการ downsampling 11

รูปที่ 2.10 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาคส่ง 12

รูปที่ 2.11 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสองขั้นของภาคส่ง 12

รูปที่ 2.12 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสามขั้นของภาคส่ง 12

รูปที่ 2.13 แถบความถี่ของภาคส่ง 13

รูปที่ 2.14 สเปกตรัมของสัญญาณอินพุตก่อนการ upsampling 14

รูปที่ 2.15 สเปกตรัมของสัญญาณเอาท์พุตเมื่อผ่านการ upsampling 14

รูปที่ 2.16 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาครับ 15

รูปที่ 2.17 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสองขั้นของภาครับ 15

รูปที่ 2.18 ฟิลเตอร์แบงค์ 15

รูปที่ 2.19 ผลตอบสนองทางความถี่ของวงจรกรองความถี่ต่ำและสูง 16

รูปที่ 2.20 แสดงการสลับเครื่องหมายในการหาค่าสัมประสิทธิ์ 18

รูปที่ 2.21 แสดงความสัมพันธ์ระหว่างสัมประสิทธิ์ทั้งภาครับและภาคส่ง 19

รูปที่ 2.22 ฟิลเตอร์แบงค์เชิงตั้งฉากที่มีสัมประสิทธิ์ 4 ค่า 19

รูปที่ 2.23 บล็อกไดอะแกรมของโครงสร้างโดยตรงของวงจรกรองความถี่ 20

รูปที่ 2.24 บล็อกไดอะแกรม การแสดงให้เห็นจริงของโพลีเฟส ของฟังก์ชันการส่งผ่าน 22

รูปที่ 2.25 แสดงการแยกโครงสร้างออกเป็นพจน์คี่และพจน์คู่ 22

รูปที่ 2.26 บล็อคไดอะแกรมก่อนการเปลี่ยนโครงสร้าง 23

รูปที่ 2.27 บล็อคไดอะแกรมหลังการเปลี่ยนโครงสร้าง 23

รูปที่ 2.28 บล็อคไดอะแกรมของโครงสร้างแบบโพลีเฟส 23

รูปที่ 2.29 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนวิเคราะห์ 24

รูปที่ 2.30 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนสังเคราะห์ 24

รูปที่ 2.31 แสดงรูปแบบส่วนวิเคราะห์และส่วนสังเคราะห์ ของฟิลเตอร์แบงค์ 25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.32 บล็อกไดอะแกรมแทนเมตริกซ์ α 26
รูปที่ 2.33 บล็อกไดอะแกรมแทนเมตริกซ์ β 27
รูปที่ 2.34 บล็อกไดอะแกรมแทนเมตริกซ์ γ 27
รูปที่ 2.35 โครงสร้างลิฟท์ติ้งที่แปลงมาจากโพลีเฟสเมตริกซ์ 28
รูปที่ 2.36 แสดงโครงสร้างทั่วไปของ Lifting scheme ทางภาควิเคราะห์ 28
รูปที่ 2.37 แสดงขั้นตอนการออกแบบระบบดิจิตอล 29
รูปที่ 2.38 แสดงการออกแบบระบบเส้นทางข้อมูล 30
รูปที่ 2.39 แสดงขั้นตอนการออกแบบจากบนลงล่าง 31
รูปที่ 2.40 แสดงผังการแบ่งกลุ่มของวงจรรวมเอซิก 32
รูปที่ 2.41 แสดงการจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิตจำนวนเต็มและบิตเศษส่วน 34
รูปที่ 2.42 แสดงการจัดรูปแบบจำนวนโดยตรงที่มีแต่บิตเศษส่วน 35
รูปที่ 2.43 แสดงการจัดรูปแบบจำนวนอิงดรรชนี 36
รูปที่ 2.44 แสดงการคูณแบบเลขส่วนเติมเต็มสองโดยใช้เลขคณิตกระจาย 40
รูปที่ 2.45 แสดงการต่อวงจร Parallel Comparator A/D Converter 43
รูปที่ 2.46 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารข้อมูลแบบขนาน 46
รูปที่ 2.47 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารแบบอนุกรม 46
รูปที่ 2.48 แสดงลักษณะของคอนเน็กเตอร์แบบ DB-9 47
รูปที่ 2.49 แสดงการเชื่อมต่อระหว่างคอนเน็คเตอร์แบบ DB-9
และ บอร์ด FPGA โดยผ่าน MAX3232 49
รูปที่ 3.1 บล็อกไดอะแกรมในส่วนของการ Simulate 50
รูปที่ 3.2 บล็อกไดอะแกรมในส่วนของการ ส่วนของการ Implement ลงบนอุปกรณ์ FPGA 50
รูปที่ 3.3 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านวิเคราะห์ 3 ระดับ 51
รูปที่ 3.4 โครงสร้างภายในของการแปลงเวฟเลตโดยใช้ lifting schemeด้านวิเคราะห์ ทางภาคส่ง 51
รูปที่ 3.5 บล็อกไดอะแกรมของการแปลงเวฟเลตด้านสังเคราะห์ 3 ระดับ 52
รูปที่ 3.6 โครงสร้างภายในของการแปลงเวฟเลตโดยใช้ lifting schemeทางด้านสังเคราะห์ ทางภาครับ 52
รูปที่ 3.7 บล็อกไดอะแกรมของวงจรหน่วงเวลา 53
รูปที่ 3.8 บล็อกไดอะแกรมของวงจรสุ่มค่าตัวอย่าง 53
รูปที่ 3.9 บล็อกไดอะแกรมของวงจรเพิ่มค่าตัวอย่าง 53
รูปที่ 3.10 บล็อกไดอะแกรมของวงจรบวก 53
รูปที่ 3.11 บล็อกไดอะแกรมของวงจรคูณ 54
รูปที่ 3.12 การแปลงเวฟเลตด้านวิเคราะห์โดยนำหลักการของ DA มาใช้ 55
รูปที่ 3.13 การแปลงเวฟเลตด้านสังเคราะห์โดยนำหลักการของ DA มาใช้ 55
รูปที่ 3.14 บล็อกไดอะแกรมของวงจร Parallel in Serial out Shift Register 56
รูปที่ 3.14 บล็อกไดอะแกรมของวงจรบวกหรือวงจรลบ 57
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.16 บล็อกไดอะแกรมของ Accumulator 57
รูปที่ 3.17 แสดงสัญลักษณ์วงจรหารความถี่สองเท่า 58
รูปที่ 3.18 สัญลักษณ์ของวงจรควบคุมการทำงานของ RAM 58
รูปที่ 3.19 แสดงการไหลของข้อมูลในระหว่างกระบวนการ Write 59
รูปที่ 3.20 แสดงเวฟฟอร์มของ Write first mode 59
รูปที่ 3.21 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter) 60
รูปที่ 3.22 บล็อกไดอะแกรมของวงจรมัลติเพล็กซ์ 60
รูปที่ 3.23 บล็อกไดอะแกรมของวงจรส่งข้อมูลแบบอนุกรม 61
รูปที่ 3.24 บล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ 61
รูปที่ 3.25 แสดงบล็อกไดอะแกรมการรับบิตข้อมูลจากบอร์ด FPGA ผ่านพอร์ตอนุกรม 61
รูปที่ 3.26 โฟลว์ชาร์ตแสดงการทำงานของการแปลงเวฟเลต ในส่วนของการ Simulate โดยใช้โปรแกรม Xilinx และแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ 62
รูปที่ 4.1 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านวิเคราะห์ 63
รูปที่ 4.2 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านสังเคราะห์ 63
รูปที่ 4.3 หน้าต่างของแมทแลบ ส่วน interface ของโปรแกรมWavelet Transform 64
รูปที่ 4.4 แสดงการเลือกชนิดของ mother wavelet 65
รูปที่ 4.5 แสดงการเลือก order ของ mother wavelet 66
รูปที่ 4.6 แสดงการเลือกสัญญาณอินพุตที่ใช้ในการแปลงเวฟเลต 67
รูปที่ 4.7 แสดงการเปรียบเทียบระหว่างสัญญาณอินพุตกับสัญญาณที่กู้กลับ เมื่อกดปุ่มAnalysis 68
รูปที่ 4.8 แสดงผลในด้านวิเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ scaling function1 และส่วนของ wavelet function1 69
รูปที่ 4.9 แสดงผลในด้านวิเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ scaling function2 และส่วนของ wavelet function2 69
รูปที่ 4.10 แสดงผลในด้านวิเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ scaling function2 และส่วนของ wavelet function2 70
รูปที่ 4.11 แสดงผลในด้านสังเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step1 70
รูปที่ 4.12 แสดงผลในด้านสังเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step2 71
รูปที่ 4.13 แสดงผลในด้านสังเคราะห์ขั้นที่ 3 โดยจะมีส่วนของ input signal และส่วนของ reconstruct signal 71
รูปที่ 4.14 เมื่อใส่สัญญาณอินพุทเป็นสัญญาณเสียง 72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.15 แสดงผลของการเปรียบเทียบระหว่างสัญญาณอินพุตกับสัญญาณที่กู้กลับ เมื่อกดปุ่มAnalysis 72
รูปที่ 4.16 แสดงผลในด้านวิเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ scaling function1 และส่วนของ wavelet function1 73
รูปที่ 4.17 แสดงผลในด้านวิเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ scaling function1 และส่วนของ wavelet function2 73
รูปที่ 4.18 แสดงผลในด้านวิเคราะห์ขั้นที่3 โดยจะมีส่วนของ scaling function1 และส่วนของ wavelet function3 74
รูปที่ 4.19 แสดงผลในด้านสังเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step1 74
รูปที่ 4.20 แสดงผลในด้านสังเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step2 75
รูปที่ 4.21 แสดงผลในด้านสังเคราะห์ขั้นที่ 3 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step3 75
รูปที่ 4.22 แสดงค่า MSE ที่คำนวณได้เมื่อกดปุ่ม Compare 76
รูปที่ 4.23 แสดงปุ่มที่ใช้ในการฟังเสียงของสัญญาณเสียงอินพุทและสัญญาณเสียงที่กู้กลับมาได้ 76
รูปที่ 4.24 แสดงสัญลักษณ์ของวงจรสุ่มค่าตัวอย่าง 76
รูปที่ 4.25 แสดงผลการจำลองการทำงานของวงจรสุ่มค่าตัวอย่าง 77
รูปที่ 4.26 แสดงสัญลักษณ์ของวงจรดีเลย์ 77
รูปที่ 4.27 แสดงผลการจำลองการทำงานของวงจรดีเลย์ 77
รูปที่ 4.28 แสดงสัญลักษณ์ของหน่วยความจำเฟสคู่ 78
รูปที่ 4.29 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM1 78
รูปที่ 4.30 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM2 78
รูปที่ 4.31 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM3 78
รูปที่ 4.32 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM4 79
รูปที่ 4.33 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM5 79
รูปที่ 4.34 แสดงสัญลักษณ์ของวงจรหารความถี่สอง 79
รูปที่ 4.35 แสดงผลการจำลองการทำงานของวงจรหารความถี่สอง 79
รูปที่ 4.36 แสดงสัญลักษณ์ของวงจรบวกสะสม 14 บิต 80
รูปที่ 4.37 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 14 บิต 80
รูปที่ 4.38 แสดงสัญลักษณ์ของวงจรบวกขนาด 14 บิต 80
รูปที่ 4.39 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 14 บิต 80
รูปที่ 4.40 แสดงสัญลักษณ์ของวงจรวงจรควบคุม State 81
รูปที่ 4.41 แสดงผลการจำลองการทำงานของวงจรวงจรวงจรควบคุม State 81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.42 แสดงสัญลักษณ์ของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 82
รูปที่ 4.43 แสดงผลการจำลองการทำงานของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 82
รูปที่ 4.44 แสดงสัญลักษณ์ของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 83
รูปที่ 4.45 แสดงผลการจำลองการทำงานของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 83
รูปที่ 4.46 แสดงสัญลักษณ์ของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 84
รูปที่ 4.47 แสดงผลการจำลองการทำงานของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 84
รูปที่ 4.48 แสดงสัญลักษณ์ของวงจรแทรกบิต 85
รูปที่ 4.49 แสดงผลการจำลองการทำงานของวงจรแทรกบิต 85
รูปที่ 4.50 แสดงสัญลักษณ์ของวงจรควบคุมหน่วยความจำ RAM 85
รูปที่ 4.51 แสดงผลการจำลองการทำงานของวงจรควบคุมหน่วยความจำ RAM 86
รูปที่ 4.52 แสดงสัญลักษณ์ของวงจรมัลติเพล็กซ์ 86
รูปที่ 4.53 แสดงผลการจำลองการทำงานของวงจรมัลติเพล็กซ์ 87
รูปที่ 4.54 แสดงสัญลักษณ์ของวงจรตัดบิต 87
รูปที่ 4.55 แสดงผลการจำลองการทำงานของวงจรตัดบิต 87
รูปที่ 4.56 แสดงสัญลักษณ์ของวงจรชิฟบิต 88
รูปที่ 4.57 แสดงผลการจำลองการทำงานของวงจรชิฟบิต 88
รูปที่ 4.58 แสดงสัญลักษณ์ของวงจรหารความถี่ 88
รูปที่ 4.59 แสดงผลการจำลองการทำงานของวงจรหารความถี่ 89
รูปที่ 4.60 แสดงสัญลักษณ์ของวงจรส่งบิตข้อมูล 89
รูปที่ 4.61 แสดงผลการจำลองการทำงานของวงจรส่งบิตข้อมูล 89
รูปที่ 4.62 แสดงสัญญาณอินพุตรูปสี่เหลี่ยม 90
รูปที่ 4.63 แสดงสัญญาณเอาท์พุตรูปสี่เหลี่ยมที่กู้กลับมาได้ 90
รูปที่ 4.64 แสดงสัญญาณอินพุตรูปซายน์ 91
รูปที่ 4.65 แสดงสัญญาณเอาท์พุตรูปซายน์ที่กู้กลับมาได้ 91
รูปที่ 4.66 แสดงบอร์ดเอฟพีจีเอที่ทำการเชื่อมต่อกับพอร์ทอนุกรม 92

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของหัวข้อปริญญานิพนธ์

เนื่องจากในปัจจุบัน เทคโนโลยีการสื่อสารข้อมูลได้รับความสนใจในการพัฒนาเป็นอย่างมาก ได้มีการคิดค้นเทคนิคต่างๆที่ใช้ในการส่งข้อมูลให้มีประสิทธิภาพมากที่สุด กล่าวคือ สามารถส่งข้อมูลไปถึงปลายทางได้อย่างถูกต้อง และใช้ทรัพยากรในการส่งน้อยที่สุด จึงมีการใช้เทคนิคการบีบอัดข้อมูล (Data Compression) เพื่อบีบอัดข้อมูลที่จะส่งให้มีขนาดเล็กลง เพื่อให้สามารถส่งข้อมูลไปยังผ่านสายส่งไปยังปลายทางได้มากขึ้น โดยที่ปลายทางต้องมีกระบวนการแปลงข้อมูลที่ถูกบีบอัดให้กลับมาเป็นอย่างเดิม

ในปัจจุบันเทคนิคในการบีบอัดข้อมูลได้รับความสนใจอย่างมาก จากนักวิจัยทั่วโลก เพื่อคิดค้นวิธีการบีบอัดข้อมูลที่มีประสิทธิภาพมากที่สุด โดยในช่วง 4-5 ปีที่ผ่านมา แนวคิดของเวฟเลต(Wavelet) ได้รับความสนใจอย่างกว้างขวาง เนื่องจากให้ประสิทธิภาพในการบีบอัดข้อมูลที่ดี และมีการผิดเพี้ยนของข้อมูลน้อย จึงถูกนำมาประยุกต์ใช้งานทางด้านต่างๆมากมาย ทั้งทางด้านการสื่อสารข้อมูล ทางด้านการแพทย์ ทางด้านอวกาศ เป็นต้น ยกตัวอย่างเช่น เป็นเทคนิคการบีบอัดสัญญาณที่ถูกใน JPEG2000 และเป็นเทคนิคในการบีบอัดสัญญาณคลื่นไฟฟ้าหัวใจ(Electro Cardio Graph) รวมถึงเป็นเทคนิคที่ใช้บีบอัดข้อมูลจากยานอวกาศมายังพื้นโลก อีกด้วย

จากที่ได้กล่าวมาข้างต้น จึงเป็นที่มาให้เกิดปริญญานิพนธ์เล่มนี้ขึ้นมา เพื่อศึกษาแนวคิดของเวฟเลต ว่ามีที่มาและมีขั้นตอนในการประมวลผลสัญญาณอย่างไร โดยจะมีการจำลองการทำงานโดยใช้โปรแกรมแมทแลบ จากนั้นจึงนำไปออกแบบวงจรดิจิตอล โดยใช้ภาษา VHDL และนำไป implement ลงบอร์ด FPGA โดยในส่วนของตัวกรองความถี่ได้เลือกใช้โครงสร้างแบบลิฟท์ติ้ง (lifting scheme) และในส่วนของการคูณจะใช้วิธีการของเลขคณิตกระจาย(Distributed Arithmetric) เพื่อให้ลดจำนวนของการประมวลผลและลดเกต (Gate) ที่นำมาใช้งาน

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1.2.1 เพื่อศึกษาหลักการบีบอัดข้อมูลเสียงโดยวิธีการแปลงเวฟเลต (Wavelet Transform) และจำลองการทำงานโดยใช้โปรแกรมแมทแลบ

1.2.2 ศึกษาหลักการของวงจรกรองความถี่ที่มีโครงสร้างแบบลิฟท์ติ้ง (lifting scheme) และโครงสร้างเลขคณิตกระจาย เพื่อนำมาประยุกต์ใช้กับส่วนของการออกแบบวงจรดิจิตอล

1.2.3 ทำการจำลองการทำงานของวิธีการแปลงเวฟเลต ด้วยการออกแบบวงจรดิจิตอลภาษา VHDL และเก็บผลการทดลองจากโปรแกรม ModelSim XE II 5.7c

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของปริญญานิพนธ์

ปริญญานิพนธ์นี้เป็นการนำวงจรกรองความถี่ที่มีโครงสร้างแบบลิฟท์ติ้งและนำโครงสร้างเลขคณิตกระจายมาประยุกต์ใช้กับวิธีการบีบอัดข้อมูล โดยวิธีการแปลงเวฟเลต และได้ทำการจำลองการทำงานผ่านโปรแกรม Xilinx-Project Navigator ซึ่งกระบวนการทำงานทั้งหมดจะถูกบรรยายพฤติกรรมการทำงานโดยใช้การเขียนด้วยภาษา VHDL ในการออกแบบและทำการจำลองผลการทดลองออกมาและจะนำไปเปรียบเทียบกับผลที่ได้จากโปรแกรมแมทแลบ

1.4 เนื้อหาของปริญญานิพนธ์

ในบทที่ 2 เป็นส่วนของทฤษฎีพื้นฐานการแปลงเวฟเลต, ทฤษฎีพื้นฐานของวงจรกรองความถี่และความสัมพันธ์ของทฤษฎีทั้งสอง, โครงสร้างของวงจรกรองความถี่แบบโพลีเฟส, โครงสร้างการแปลงเวฟเลตโดยใช้หลักการของลิฟท์ติ้ง (Lifting scheme), การประยุกต์ใช้ของวิธีการแปลงเวฟเลตโดยใช้โครงสร้างแบบ lifting scheme, ทฤษฎีของภาษา VHDL, ทฤษฎีของอุปกรณ์ FPGA, ทฤษฎีของ DA (Distributed Arithmetic)

ในบทที่ 3 เป็นส่วนของการคำนวณและการสร้าง โดยจะแบ่งเป็นส่วนของการ Simulate โดยใช้โปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL ในการออกแบบ

ในบทที่ 4 เป็นส่วนของผลการทดลอง ซึ่งประกอบด้วยส่วนของการ Simulate โดยใช้โปรแกรม MATLAB, ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL

ในบทที่ 5 เป็นส่วนของบทวิจารณ์และบทสรุป

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

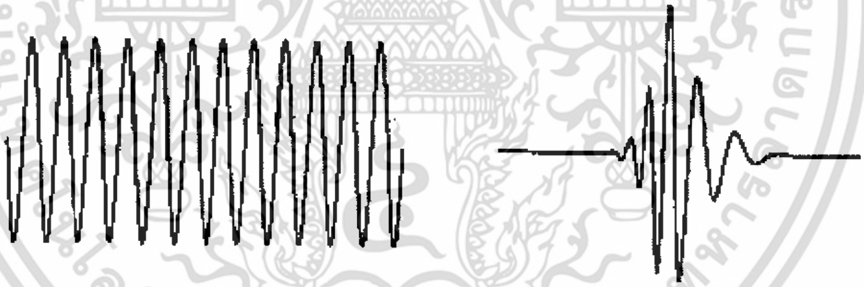
บทที่ 2
ทฤษฎีและหลักการ

2.1 ความรู้เบื้องต้นของเวฟเลต

เมื่อกล่าวถึงคลื่น (wave) จะมีลักษณะเป็นสัญญาณที่มีการเปลี่ยนแปลงอยู่ตลอดเวลาหรือที่เรียกว่าเกิดการออสซิเลต (oscillate) เช่น สัญญาณรูปไซน์ (sinusoid) ซึ่งจะใช้การวิเคราะห์โดยการแปลงฟูเรียร์(Fourier Transform) คลื่นโดยทั่วไปจะมีการกระจายพลังงานอย่างไม่มีที่สิ้นสุดเนื่องจากสัญญาณเปลี่ยนแปลงอยู่ตลอดเวลา ถ้าเทียบกับเวฟเลต (wavelet) เปรียบเสมือนคลื่นเล็กๆ ที่เกิดขึ้นชั่วขณะหนึ่ง มีการเปลี่ยนแปลงแบบไม่คงที่ และมีการแปรผันตามเวลา และมีค่าพลังงานรวมกันอยู่ในช่วงใดช่วงหนึ่ง และสามารถวิเคราะห์ได้ทั้งในเทอมของเวลาและความถี่ในเวลาเดียวกัน โดยที่การกระจายเวฟเลต (wavelet expansion) สามารถเขียนให้อยู่ในรูปสมการที่ 2.1 ดังนี้

$$f(t) = \sum_k \sum_j a_{j,k} \psi_{j,k}(t) \tag{2.1}$$

โดยที่ $a_{j,k}$ จะแทนสัมประสิทธิ์การกระจาย และ $\psi_{j,k}(t)$ จะแทนฟังก์ชันการกระจายของเวฟเลต โดยที่เราจะเรียกเซตของสัมประสิทธิ์การกระจายนี้ว่า การแปลงเวฟเลตแบบไม่ต่อเนื่อง (Discrete Wavelet Transform) และสามารถแปลงกลับได้ตามสมการข้างต้น



รูปที่ 2.1 แสดงลักษณะของคลื่นเปรียบเทียบกับเวฟเลต

2.1.1 ระบบของเวฟเลต (Wavelet System) จะมีคุณสมบัติโดยทั่วไป ดังนี้

1. ระบบของเวฟเลตจะมีลักษณะเป็นบล็อกที่สร้างขึ้น (building blocks)หรือสามารถแสดงเป็นสัญญาณหรือฟังก์ชันได้ และสามารถกระจายให้อยู่ในสองมิติได้
2. การกระจายของเวฟเลตทำให้สัญญาณมีการจำกัด (localization)อยู่ในช่วงเวลาและความถี่หนึ่ง ทำให้พลังงานของสัญญาณส่วนใหญ่สามารถแทนด้วยสัมประสิทธิ์การกระจายเพียงไม่กี่ตัว
3. การคำนวณสัมประสิทธิ์จากสัญญาณสามารถทำได้อย่างมีประสิทธิภาพ(efficiently)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกระจายแบบเวฟเลตจะทำการแปลงให้อยู่ในสองมิติ ทำให้มีการจำกัดสัญญาณให้อยู่ในเทอมของเวลาและความถี่ ดังนั้นเวฟเลตจึงสามารถแสดงตำแหน่งของสัญญาณให้อยู่ในรูปของเวลาและความถี่ได้พร้อมๆกัน เช่นระดับเสียงดนตรี คือ โน้ตตัวหนึ่งสามารถบอกได้ทั้งระดับของเสียง (Tones) และความถี่ของเสียง (frequencies)

2.1.2 คุณสมบัติพิเศษของเวฟเลต (Specific Characteristics of Wavelet Systems)

1. เวฟเลตทุกประเภทสามารถสร้างได้จากสเกลลิ่งฟังก์ชันหรือเวฟเลตเพียงตัวเดียว โดยการแบ่งระดับ (scaling) และการเลื่อนตำแหน่ง (translation)โดยที่เราจะเรียกเวฟเลตต้นกำเนิดนี้ว่า "เวฟเลตแม่" (Mother wavelet) หรือ $\psi(t)$ โดยที่ j จะแทนระดับต่างของสัญญาณ และ k จะแทนการเลื่อนตำแหน่ง ดังสมการที่ 2.2

$$\psi_{j,k}(t) = 2^{j/2}\psi(2^{j}t - k) \tag{2.2}$$

2. เวฟเลตเกือบทุกแบบสามารถใช้เงื่อนไขมัลติรีโซลูชัน (multiresolution) หมายความว่าถ้าสัญญาณใดสัญญาณหนึ่งสามารถแสดงให้อยู่ในเทอมของฟังก์ชันสเกลลิ่ง (scaling function) $\varphi(t-k)$ ได้ สัญญาณนี้ก็สามารถแสดงให้อยู่ในรูป $\varphi(2t-k)$ ได้เช่นกัน หรืออีกนัยหนึ่งคือการกระจายของเวฟเลตจะแบ่งสัญญาณเดิมออกเป็นส่วนย่อยๆได้
3. สัมประสิทธิ์การกระจายในส่วนย่อยๆสามารถคำนวณได้จากสัมประสิทธิ์ของส่วนที่ใหญ่กว่าตามวิธีการแตกกิ่งก้านสาขา หรือ ฟิลเตอร์แบงค์ (filter bank) ทำให้การคำนวณสัมประสิทธิ์การกระจายมีประสิทธิภาพมากยิ่งขึ้น

การวิเคราะห์เวฟเลตจะเหมาะสมกับสัญญาณที่เกิดขึ้นชั่วขณะหนึ่ง ซึ่งจากลักษณะจำเพาะอยู่เฉพาะส่วนของเวฟเลต ทำให้สัมประสิทธิ์การกระจายมีค่าน้อย ซึ่งจะเป็นประโยชน์กับการนำไปประยุกต์ใช้งาน

2.1.3 การกระจายเวฟเลตและการแปลงเวฟเลต (Wavelet expansions and wavelet transform)

สามารถพิสูจน์ได้ว่ามีประสิทธิภาพและสามารถนำไปใช้งานได้จริงในทุกสัญญาณ ตามเหตุผลดังต่อไปนี้

1. ขนาดของสัมประสิทธิ์การกระจายของเวฟเลต $a_{j,k}$ มีการลดลงอย่างรวดเร็วตามระดับของสัญญาณที่เพิ่มขึ้นตามพจน์ j และ k ดังนั้นจึงเหมาะแก่การใช้งานในด้านการบีบอัดข้อมูลภาพ (image compression) การลดสัญญาณรบกวน (denoising) และการตรวจจับสัญญาณ (detection)
2. การกระจายเวฟเลตทำให้ได้รูปร่างลักษณะของสัญญาณที่มีความถูกต้องและเฉพาะมากกว่าเนื่องจากจะทำให้ได้สัญญาณที่มีความจำเพาะมากกว่าและง่ายต่อการอธิบาย นอกจากนี้การแยกสัญญาณของเวฟเลตจะสามารถแยกได้ทั้งในเทอมของเวลาและความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เวฟเลตสามารถปรับตัวให้เข้ากับระบบหรือสัญญาณได้หลายรูปแบบ จึงสามารถนำไปประยุกต์ใช้งานได้หลายอย่าง
4. การคำนวณและการแปลงเวฟเลตเหมาะแก่การใช้งานในระบบดิจิตอลและคอมพิวเตอร์ เนื่องจากมีเพียงการคูณและการบวกเท่านั้น

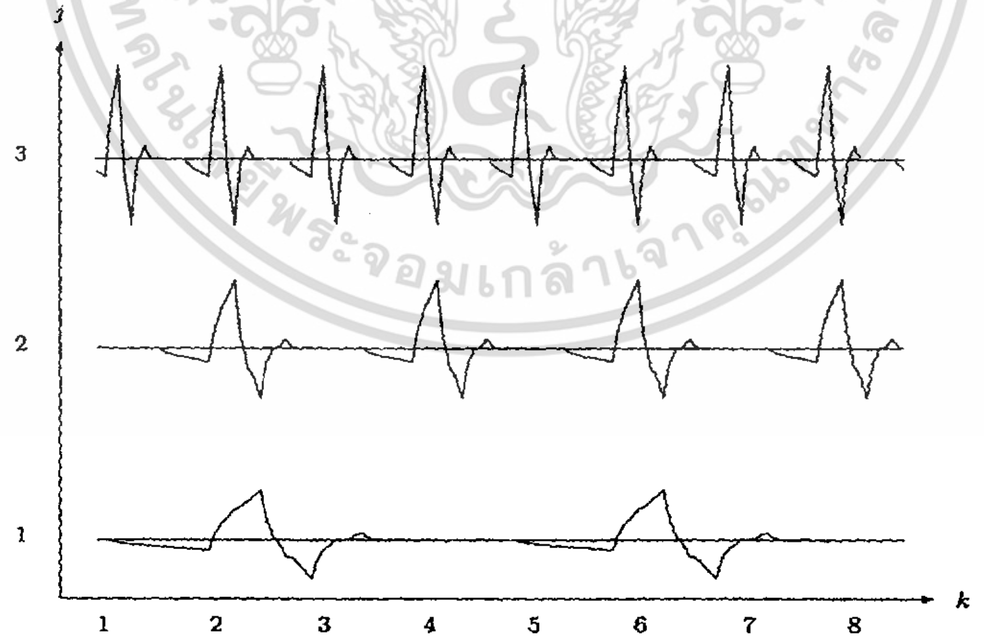
2.2 การแปลงเวฟเลตแบบไม่ต่อเนื่อง (The Discrete Wavelet Transform)

จุดประสงค์ของการแปลงเวฟเลต คือ การกระจายสัญญาณออกเป็นเซตของฟังก์ชันๆหนึ่ง โดยที่สามารถกระจายให้อยู่ในเทอมของเวลาและความถี่ ดังสมการที่ 2.3

$$f(t) = \sum_{j,k} a_{j,k} 2^{j/2} \psi\left(2^j t - k\right) \tag{2.3}$$

โดยที่ $a_{j,k}$ เป็นเซตของสัมประสิทธิ์การกระจายที่มีสองมิติ เรียกว่า การแปลงเวฟเลตแบบไม่ต่อเนื่อง หรือ *Discrete wavelet transforms (DWT)* จะเห็นได้ว่าการกระจายแบบนี้มีตัวระบุตำแหน่งอยู่สองตัวคือ j และ k โดยที่ตัวที่ใช้ในการเลื่อนตำแหน่งหรือ translation คือ k และตัวที่ใช้ในการแบ่งระดับหรือ scaling คือ j

เป้าหมายของการกระจายสัญญาณหรือฟังก์ชันทั่วไป คือ การใช้ประโยชน์จากสัมประสิทธิ์การกระจายให้คุ้มค่ามากที่สุด หรือพยายามให้ตัวสัมประสิทธิ์มีข้อมูลของสัญญาณเดิมให้มากที่สุด และพยายามให้จำนวนของสัมประสิทธิ์มีน้อยที่สุด หรือเป็นศูนย์มากที่สุด ซึ่งเหมาะแก่การใช้งานด้านการบีบอัดข้อมูล การลดสัญญาณรบกวน



รูปที่ 2.2 การเลื่อนตำแหน่งและการปรับระดับของเวฟเลต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ทฤษฎีมัลติรีโซลูชัน (Multiresolution)

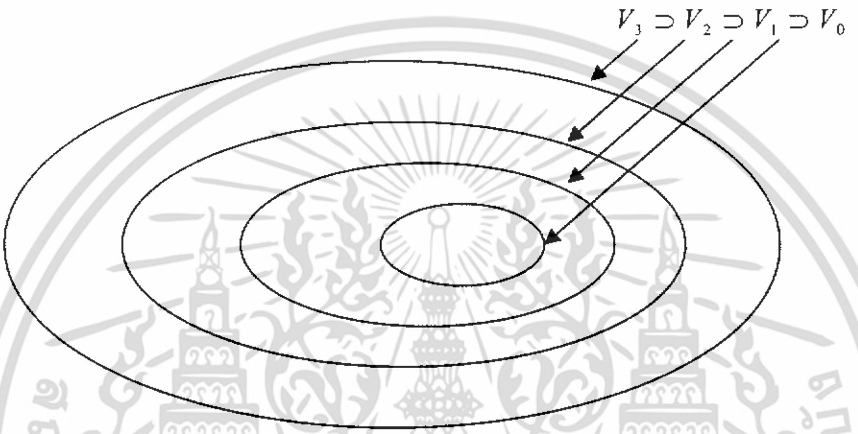
การวิเคราะห์โดยการแบ่งย่อยออกเป็นหลายส่วน สามารถอธิบายให้อยู่ในรูปของพื้นที่ย่อยๆที่เป็นสับเซตของพื้นที่ใหญ่ได้ดังนี้ โดยที่ L^2 จะแทนพื้นที่ทั้งหมดของสัญญาณ

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2 \qquad (2.4)$$

หรือ $$V_j \subset V_{j+1} \qquad (2.5)$$

โดยที่ $$V_{-\infty} = \{0\} \quad \text{และ} \quad V_{\infty} = L^2 \qquad (2.6)$$

และ $$f(t) \in V_j \Leftrightarrow f(2t) \in V_{j+1} \qquad (2.7)$$



รูปที่ 2.3 การแบ่งพื้นที่ออกเป็นส่วนย่อยโดยการใช้สเกลลิ่งฟังก์ชัน

จากที่กล่าวมาข้างต้นเราสามารถหาค่าของสเกลลิ่งฟังก์ชันได้จาก

$$\varphi(t) = \sum_n h(n)\sqrt{2}\varphi(2t-n) \qquad (2.8)$$

โดยที่ $h(n)$ จะหมายถึงสัมประสิทธิ์ของฟังก์ชันสเกลลิ่งส่วนค่า $\sqrt{2}$ จะทำการรักษาระดับมาตรฐานของฟังก์ชันไว้

2.3.1 สเกลลิ่งฟังก์ชัน (The Scaling Function)

เป็นวิธีการหนึ่งในวิธีมัลติรีโซลูชัน โดยการแบ่งสัญญาณออกเป็นส่วนย่อย หรือเป็นระดับแล้วจึงอธิบายเวฟเลตในเทอมของสเกลลิ่งฟังก์ชัน ดังสมการที่ 2.4

$$f(t) = \sum_k a_k \varphi_k(t) \qquad \text{โดยที่} \quad f(t) \in V_0 \qquad (2.9)$$

เราสามารถเพิ่มขนาดของส่วนย่อยนี้โดยการเปลี่ยนค่าระดับในสเกลลิ่งฟังก์ชัน โดยสามารถแสดงให้อยู่ในสองมิติ คือ การสเกลลิ่งและการทรานสเลชัน ได้ดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\varphi_{j,k}(t) = 2^{j/2}\varphi(2^j t - k) \tag{2.10}$$

โดยที่ ถ้า $j \rangle 0$ จะทำให้มีการแบ่งมากขึ้น ทำให้ $\varphi_{j,k}(t)$ มีขนาดแคบหรือเล็กลง และมีการเลื่อนตำแหน่งที่น้อยลง ดังนั้นจะได้รายละเอียดดีขึ้น ในทางตรงกันข้ามถ้า $j \langle 0$ ทำให้ $\varphi_{j,k}(t)$ จะขนาดกว้างขึ้นและเลื่อนตำแหน่งมากขึ้น สรุปได้ว่าในส่วนสเกลลิ่งฟังก์ชันจะทำให้ได้ข้อมูลแบบหยาบๆ

2.3.2 เวฟเลตฟังก์ชัน (The Wavelet Function)

เป็นอีกขั้นตอนหนึ่งในวิธีมัลติรีโซลูชัน โดยการกำหนดค่าความแตกต่างระหว่างสเกลลิ่งฟังก์ชันในแต่ละระดับ ดังนั้นจึงจะอธิบายให้อยู่ในเทอมของสเกลลิ่งฟังก์ชัน ข้อได้เปรียบของการทำสเกลลิ่งฟังก์ชันและเวฟเลตคือ การเป็นฟังก์ชันเชิงตั้งฉากซึ่งกันและกัน ทำให้ง่ายต่อการหาสัมประสิทธิ์ของแต่ละชุด โดยจะแทนเวฟเลตนี้ด้วยสัญญลักษณ์ W_j สามารถอธิบายได้ดังนี้

$$V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2 \tag{2.11}$$

ดังนั้น

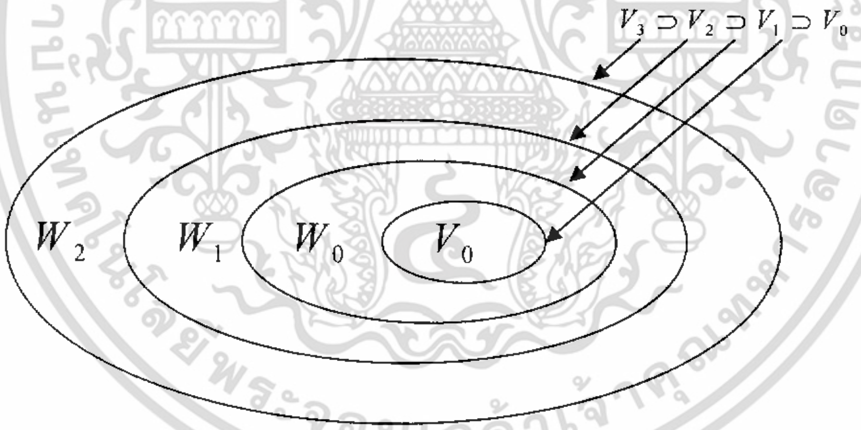
$$V_1 = V_0 \oplus W_0 \tag{2.12}$$

ส่วนต่อไปคือ

$$V_2 = V_0 \oplus W_0 \oplus W_1 \tag{2.13}$$

สามารถอธิบายให้อยู่ในรูปทั่วไปได้ดังนี้

$$L^2 = V_0 \oplus W_0 \oplus W_1 \oplus W_2 \oplus \cdots \tag{2.14}$$



รูปที่ 2.4 สเกลลิ่งฟังก์ชันและเวฟเลต

จากที่อธิบายมาข้างต้น สามารถหาค่า เวฟเลตฟังก์ชันได้ดังนี้

$$\psi(t) = \sum_n h_1(n)\sqrt{2}\varphi(2t - n) \tag{2.15}$$

โดยที่ $h_1(n)$ จะหมายถึงสัมประสิทธิ์ของฟังก์ชันเวฟเลต ส่วนค่า $\sqrt{2}$ จะทำการรักษาระดับมาตรฐานของฟังก์ชันไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ฟิลเตอร์แบงค์และการแปลงเวฟเลต (Filter Banks and the Discrete Wavelet Transform)

ในหัวข้อนี้จะอธิบายถึงการแปลงเวฟเลต (Wavelet transform) ที่สามารถคำนวณได้จากฟิลเตอร์แบงค์ (filter bank) โดยประเด็นหลักจะอยู่ที่ความสัมพันธ์กันระหว่างการแปลงเวฟเลตและฟิลเตอร์แบงค์

2.4.1 หลักการวิเคราะห์ (Analysis-From Fine Scale to Coarse Scale)

ในการที่หาสัมประสิทธิ์ของการแปลงเวฟเลต จะหาได้จากความสัมพันธ์ของสัมประสิทธิ์การกระจายระหว่างระดับต่ำและระดับสูง หรืออาจกล่าวได้ว่าจากระดับที่มีความละเอียดไปสู่ระดับหยาบ โดยสามารถคำนวณได้จากสมการดังต่อไปนี้

$$\varphi(t) = \sum_n h(n)\sqrt{2}\varphi(2t-n) \tag{2.16}$$

ทำการเปลี่ยนระดับและการเลื่อนตำแหน่ง โดยการแทนค่า $t = 2^j t - k$

$$\varphi(2^j t - k) = \sum_n h(n)\sqrt{2}\varphi(2(2^j t - k) - n) = \sum_n h(n)\sqrt{2}\varphi(2^{j+1}t - 2k - n) \tag{2.17}$$

แทนค่า $m = 2k + n$

$$\varphi(2^j t - k) = \sum_m h(m-2k)\sqrt{2}\varphi(2^{j+1}t - m) \tag{2.18}$$

ถ้าเราแทนพจน์ที่เกิดจากการกระจายด้วย V_j

$$V_j = Span\left\{2^{j/2}\varphi(2^j t - k)\right\} \tag{2.19}$$

ดังนั้น จะได้ว่า

$$f(t) \in V_{j+1} \Rightarrow f(t) = \sum_k c_{j+1}(k) 2^{(j+1)/2}\varphi(2^{j+1}t - k) \tag{2.20}$$

จาก $V_{j+1} = V_j \oplus W_j$

$$f(t) = \sum_k c_j(k) 2^{j/2}\varphi(2^j t - k) + \sum_k d_j(k) 2^{j/2}\psi(2^j t - k) \tag{2.21}$$

สามารถหาค่าสัมประสิทธิ์ของสเกลลิ่งฟังก์ชันได้ดังนี้

$$c_j(k) = \sum_m h(m-2k) c_{j+1}(m) \tag{2.22}$$

และจากความสัมพันธ์สามารถหาค่าสัมประสิทธิ์ของเวฟเลตได้ดังนี้

$$d_j(k) = \sum_m h_1(m-2k) c_{j+1}(m) \tag{2.23}$$

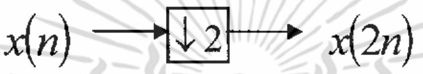
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.1 การกรองสัญญาณและการสุ่มค่าตัวอย่าง (Filtering and Down-Sampling or Decimating)

จากหลักการของการประมวลผลสัญญาณดิจิตอล การกรองสัญญาณสามารถทำได้โดยการทำคอนโวลูชั่นระหว่างสัญญาณอินพุตกับสัมประสิทธิ์ของวงจรกรองความถี่ (Filter coefficients or impulse response) โดยถ้าสมมติให้อินพุตคือ $x(n)$ และสัมประสิทธิ์ของวงจรกรองความถี่คือ $h(n)$ จะได้เอาต์พุตคือ $y(n)$ ดังสมการ

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \tag{2.24}$$

การสุ่มค่าตัวอย่างจะเป็นการนำอินพุต $x(n)$ สร้างออกมาเป็นเอาต์พุต $y(n) = x(2n)$ ดังรูป

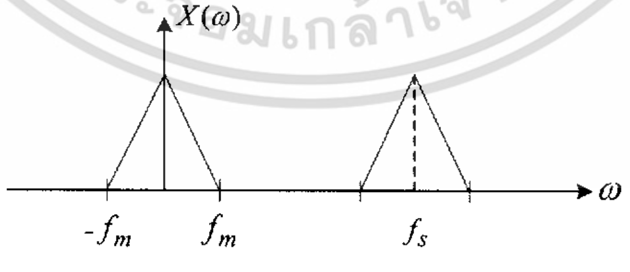


รูปที่ 2.5 ตัวสุ่มค่าตัวอย่าง (Down Sampler or Decimator)

จาก $$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}$$

$$y(n) = x(2n)\ ; \qquad Y(\omega) = \sum_{n=-\infty}^{\infty} x(2n)e^{-j\omega n} \tag{2.25}$$

ถ้าความถี่สูงสุดของสัญญาณเป็น f_m และสุ่มสัญญาณด้วยความถี่ f_s จะไม่ให้เกิด aliasing $f_s \geq 2f_m$ sampling ด้วย f_s จะได้สเปกตรัมของสัญญาณเป็น

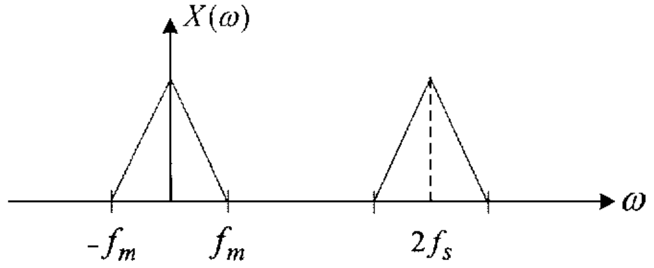


รูปที่ 2.6 สเปกตรัมของสัญญาณ $X(\omega)$ เมื่อสุ่มสัญญาณด้วยความถี่ f_s

ถ้า normalized $f_s = 2\pi \quad \therefore \quad f_m = \dfrac{2\pi f_m}{f_s}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และถ้า sampling ด้วย $2f_s$ จะได้สเปกตรัมของสัญญาณเป็น



รูปที่ 2.7 สเปกตรัมของสัญญาณ $X(\omega)$ เมื่อสุ่มสัญญาณด้วยความถี่ $2f_s$

ถ้า normalized $2f_s = 2\pi \quad \therefore \quad f_m = \dfrac{\pi f_m}{f_s}$

จะเห็นว่าสเปกตรัมของการ sampling ด้วย $2f_s$ จะได้ $f_m = \dfrac{2\pi f_m}{f_s}$ และถ้า sampling ด้วย f_s จะได้ $f_m = \dfrac{\pi f_m}{f_s}$

$\therefore$ การ downsampling จาก $2f_s$ ไป f_s จะทำให้ f_m เพิ่มเป็น 2 เท่า (เกิดการขยายทางความถี่)

จาก $$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}$$

$$= \sum_{n=-\infty}^{\infty} x(2n)e^{-j\omega(2n)} - \sum_{n=-\infty}^{\infty} x(2n-1)e^{-j\omega(2n-1)} \qquad (2.26)$$

และ $$X(\omega-\pi) = \sum_{n=-\infty}^{\infty} x(2n)e^{-j(\omega-\pi)2n} - \sum_{n=-\infty}^{\infty} x(2n-1)e^{-j(\omega-\pi)(2n-1)}$$

$$= \sum_{n=-\infty}^{\infty} x(2n)e^{-j\omega(2n)} + \sum_{n=-\infty}^{\infty} x(2n-1)e^{-j\omega(2n-1)} \qquad (2.27)$$

นำสมการ $\dfrac{(2.26)+(2.27)}{2}$: $\dfrac{X(\omega)+X(\omega-\pi)}{2} = \displaystyle\sum_{n=-\infty}^{\infty} x(2n)e^{-j\omega(2n)}$

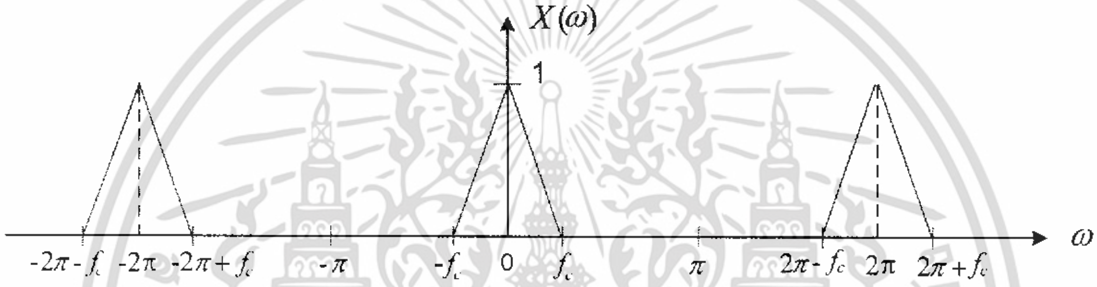
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า y(n) = x(2n)

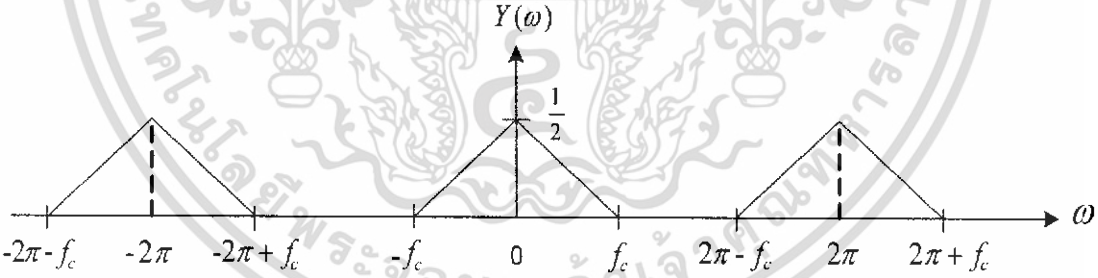
$$Y(\omega) = \sum_{n=-\infty}^{\infty} y(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty} x(2n)e^{-j\frac{\omega}{2}(2n)}$$

$$= \frac{X(\frac{\omega}{2}) + X(\frac{\omega}{2} - \pi)}{2} \qquad (2.28)$$

เมื่อนำไปพล็อตกราฟจะได้



รูปที่ 2.8 สเปกตรัมของสัญญาณอินพุตก่อนการ downsampling

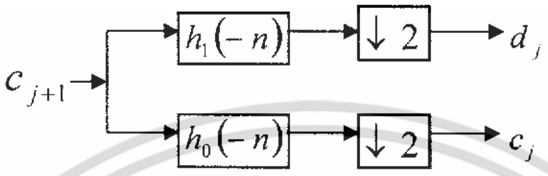


รูปที่ 2.9 สเปกตรัมของสัญญาณเอาท์พุตหลังจากผ่านการ downsampling

จากกราฟเราจะพบว่าการ downsampling จะทำให้สเปกตรัมของสัญญาณอินพุตถูกขยายออกเป็น 2 เท่าและทำให้ขนาดลดลงครึ่งหนึ่ง ถ้าต้องการไม่ให้เกิดการ aliasing จึงต้องมีฟิลเตอร์กรองก่อนที่จะ downsampling (เป็น LPF ที่มี cut-off ที่ $\frac{\pi}{2}$) เรียกว่า decimation filter หรือ anti aliasing filter เพื่อป้องกันไม่ให้เกิดการซ้อนทับเนื่องจากการขยายของความถี่ขึ้น 2 เท่า $\left(f_c < \frac{\pi}{2} \right)$

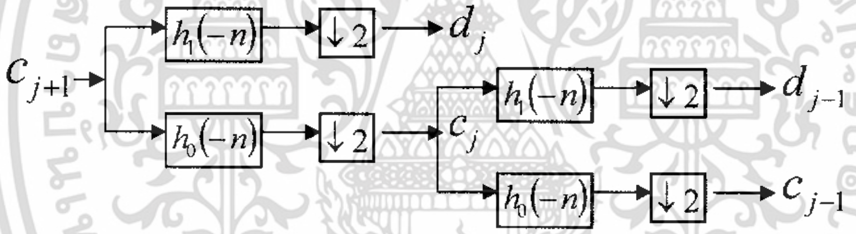
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ (2.22) และ (2.23) จะเห็นได้ว่าสัมประสิทธิ์ของสเกลลิ่งและเวฟเลตในระดับที่ต่างกันเกิดจากการทำคอนโวลูชั่นระหว่างสัมประสิทธิ์การกระจายในระดับ j กับสัมประสิทธิ์ของวงจรกรองความถี่ $h_0(-n)$ และ $h_1(-n)$ จากนั้นจึงนำมาสุ่มค่าตัวอย่าง จึงจะได้สัมประสิทธิ์การกระจายในลำดับถัดไป คือ $j-1$ หรืออาจกล่าวได้ว่า สัมประสิทธิ์ในระดับ j ถูกกรองโดยวงจรกรองความถี่สองตัวซึ่งมีสัมประสิทธิ์คือ $h_0(-n)$ และ $h_1(-n)$ และหลังจากผ่านการสุ่มค่าตัวอย่างก็จะได้สัมประสิทธิ์สเกลลิ่งและเวฟเลตในระดับที่ละเอียดน้อยกว่า ดังรูป



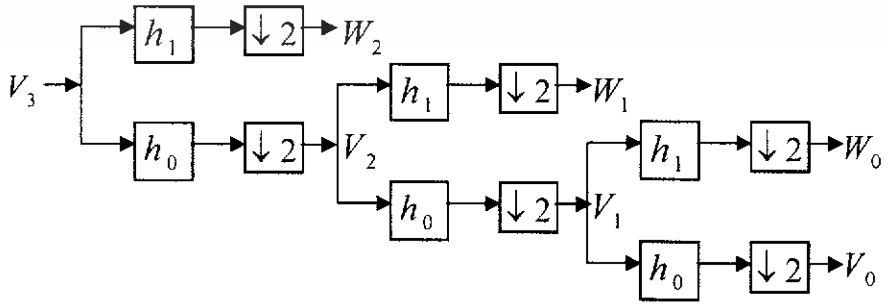
รูปที่ 2.10 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาคส่ง

การแตกออก การกรองสัญญาณและการสุ่มค่าตัวอย่างจะทำซ้ำในส่วนของสัมประสิทธิ์สเกลลิ่งเพื่อให้ได้โครงสร้างตามรูปที่ 2.12 ซึ่งเราเรียกว่า Iterating the filter bank



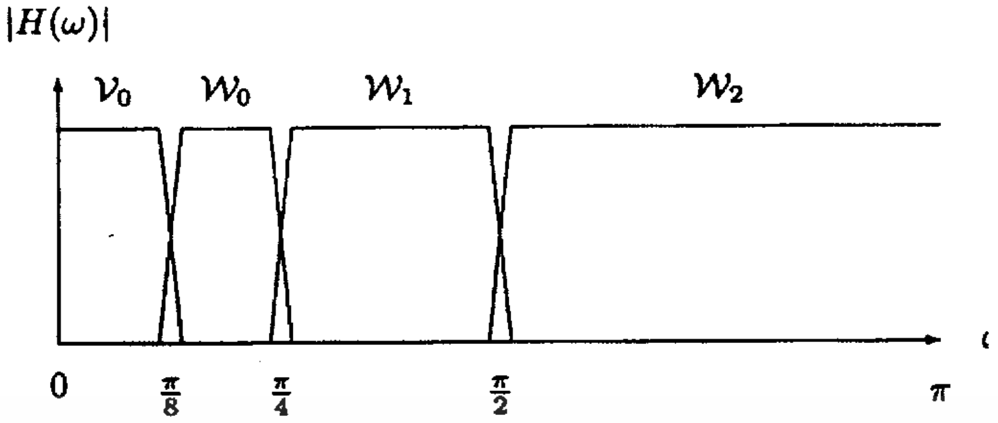
รูปที่ 2.11 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสองขั้นของภาคส่ง

ในขั้นแรกจะแบ่งส่วนของสเกลลิ่งฟังก์ชันออกเป็นส่วนของวงจรกรองความถี่ต่ำ (Lowpass Band) และส่วนของวงจรกรองความถี่สูง (Highpass Band) ซึ่งจะทำให้ได้สัมประสิทธิ์ของสเกลลิ่งและสัมประสิทธิ์ของเวฟเลตในระดับต่ำกว่า ส่วนในขั้นที่สองจะทำการแบ่งส่วนของวงจรกรองความถี่ต่ำเดิมออกเป็นส่วนของวงจรกรองความถี่ต่ำและส่วนของวงจรกรองความถี่สูงอีกครั้ง ดังรูป



รูปที่ 2.12 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสามขั้นของภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แถบความถี่ของภาคส่ง

2.4.2 หลักการสังเคราะห์ (Synthesis-From Coarse Scale to Fine Scale)

ในการที่จะนำสัญญาณเดิมกลับคืนมาสามารถทำได้โดยการรวมกันของสัมประสิทธิ์สเกลลิ่งฟังก์ชันและเวฟเลตในระดับที่มีความละเอียดน้อยกว่า โดยสามารถเขียนเป็นสมการที่อยู่ในรูปของสเกลลิ่งฟังก์ชันในระดับ $j+1$ ได้ดังนี้

$$f(t) = \sum_k c_{j+1}(k) 2^{(j+1)/2} \varphi\left(2^{j+1} t - k\right) \tag{2.29}$$

หรืออาจจะเขียนในรูปของระดับถัดไป ซึ่งจะมีส่วนของเวฟเลตด้วยดังนี้

$$f(t) = \sum_k c_j(k) 2^{j/2} \varphi\left(2^j t - k\right) + \sum_k d_j(k) 2^{j/2} \psi\left(2^j t - k\right) \tag{2.30}$$

เมื่อแทนค่าสมการที่ (2.17) และ (2.15) ลงในสมการที่ (2.30) จะได้

$$f(t) = \sum_k c_j(k) \sum_n h(n) 2^{(j+1)/2} \varphi\left(2^{j+1} t - 2k - n\right) + \sum_k d_j(k) \sum_n h_1(n) 2^{(j+1)/2} \varphi\left(2^{j+1} t - 2k - n\right) \tag{2.31}$$

เนื่องจากฟังก์ชันทั้งหมดนี้เป็นฟังก์ชันเชิงตั้งฉาก จึงทำการคูณสมการที่ (2.29) และ (2.31) ด้วย $\varphi\left(2^{j+1} t - k\right)$ และทำการอินทิเกรตหาค่าสัมประสิทธิ์ได้ดังนี้

$$c_{j+1}(k) = \sum_m c_j(m) h(k - 2m) + \sum_m d_j(m) h_1(k - 2m) \tag{2.32}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.1การกรองสัญญาณและการเพิ่มค่าตัวอย่าง (Filtering and Up-Sampling or Stretching)

ในส่วนของฟิลเตอร์แบงค์ด้านสังเคราะห์จะประกอบด้วยการเพิ่มค่าตัวอย่างและการกรองสัญญาณ ซึ่งหมายความว่าอินพุตของฟิลเตอร์จะมีศูนย์แทรกอยู่ระหว่างสัญญาณเดิม ดังสมการ

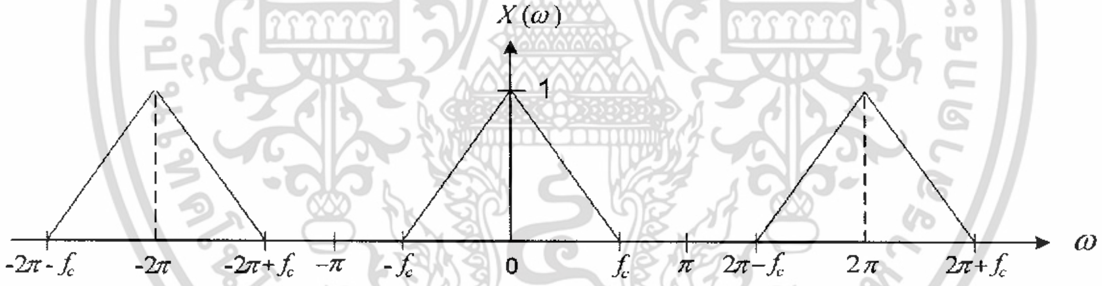
การ Upsampling

$$y(n) = \begin{cases} x(\frac{n}{2}) & n : \text{even} \\ 0 & n : \text{odd} \end{cases}$$

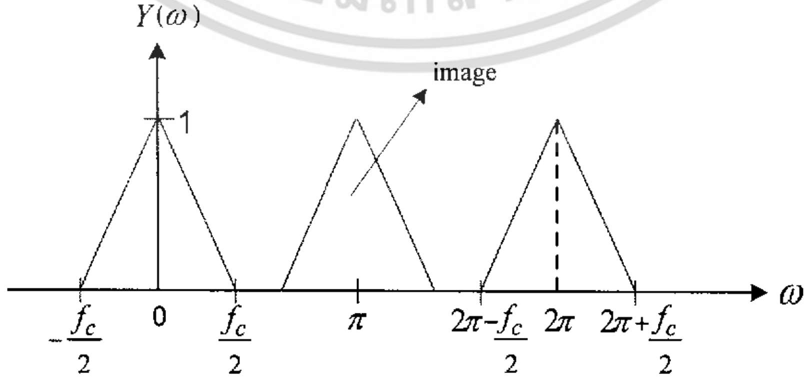
$$Y(\omega) = \sum_{n=-\infty}^{\infty} y(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty} y(2n)e^{-j\omega(2n)} \tag{2.33}$$

จาก $y(n) = x(n/2)$; $$Y(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j2\omega n} = X(2\omega) \tag{2.34}$$

เมื่อนำไปพล็อตกราฟจะได้



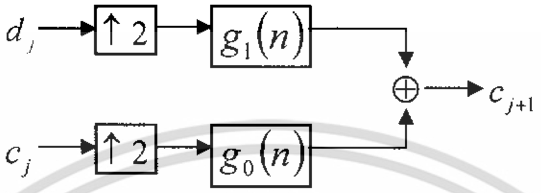
รูปที่ 2.14 สเปกตรัมของสัญญาณอินพุตก่อนการ upsampling



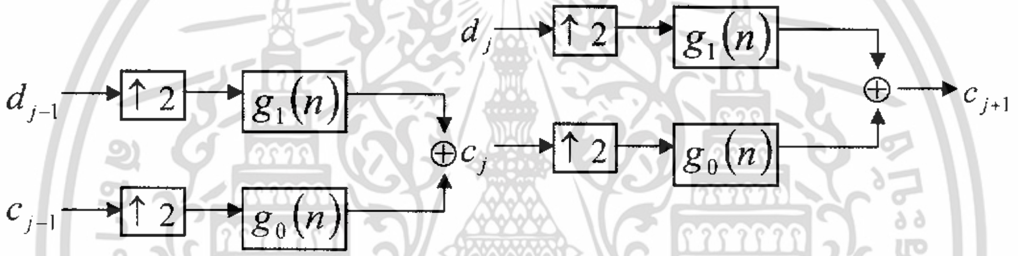
รูปที่ 2.15 สเปกตรัมของสัญญาณเอาท์พุตเมื่อผ่านการ upsampling

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากกราฟเราจะพบว่าการ upsampling จะทำให้ความกว้างสเปกตรัมของสัญญาณลดลงเหลือครึ่งหนึ่งของสัญญาณเดิม และจะเกิดสเปกตรัมของ image ขึ้น(เกิดจาการเติมศูนย์ในตำแหน่งคี่) ซึ่งเราต้องกรองสัญญาณ image ออกโดยใช้วงจรกรองความถี่ต่ำ เพื่อทำให้สเปกตรัมเหมือนเดิม เรียกตัวกรองความถี่ต่ำนี้ว่า interpolation filter และจากสมการที่ (2.32) ซึ่งเกิดจากการเพิ่มค่าตัวอย่างในระดับ j จากนั้นจึงทำการคอนโวลูชั่นกับสัมประสิทธิ์ของวงจรกรองความถี่ $g(n)$ ทั้งในส่วนของสเกลลิ่งฟังก์ชันและเวฟเลต แล้วนำมาบวกกันได้เป็นสัมประสิทธิ์ในระดับ $j+1$ ดังรูป



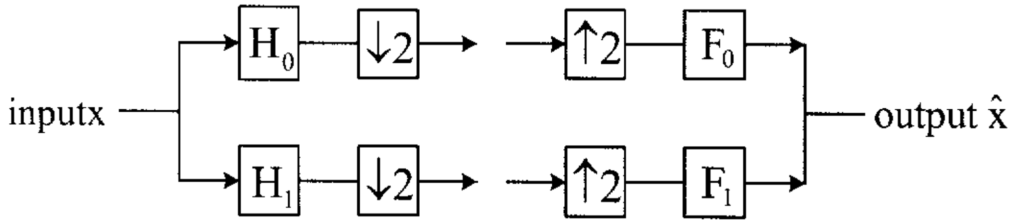
รูปที่ 2.16 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาคครับ



รูปที่ 2.17 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสองขั้นของภาครับ

2.5 ฟิลเตอร์แบงค์ (Filter Bank)

ฟิลเตอร์แบงค์ คือ เซตของวงจรกรองความถี่ที่เชื่อมต่อกับตัวสุ่มค่าสัญญาณ หรือบางครั้งอาจเป็นตัวหน่วงสัญญาณ(Delay) โดยที่ตัวสุ่มค่าสัญญาณจะเป็นตัวที่ทำการลดขนาดของอินพุต (decimator) และตัวเพิ่มค่าสัญญาณจะเป็นตัวที่ทำการขยายขนาด (expander) ฟิลเตอร์แบงค์ทั้งภาคส่งและภาครับ ภาคส่งหรือวงจรกรองความถี่ด้านวิเคราะห์ (analysis filter) จะประกอบด้วยวงจรกรองความถี่ต่ำและวงจรกรองความถี่สูง โดยจะแทนด้วย H_0 และ H_1 ตามลำดับ ดังรูป

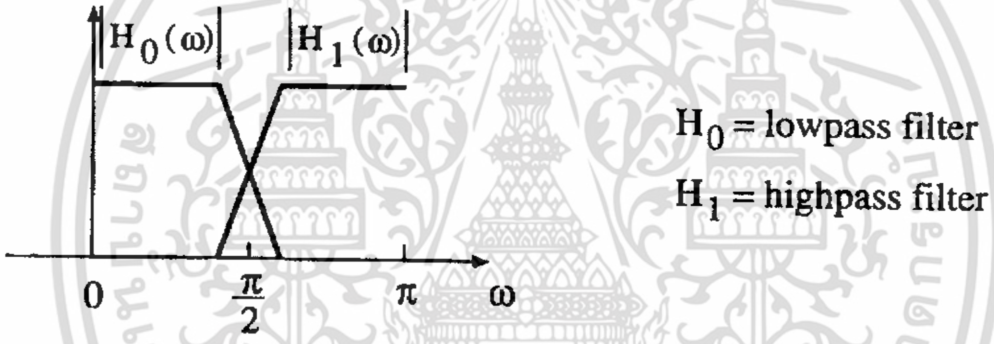


รูปที่ 2.18 ฟิลเตอร์แบงค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยต่อไปเราจะอธิบายถึงการเลือก H_0, H_1, F_0, F_1 เพื่อให้ได้ตรงตามหลักการของการกู้กลับอย่างสมบูรณ์(Perfect reconstruction) ส่วนตรงที่เป็นช่องว่างในรูปจะหมายถึงการที่สัญญาณที่ถูกสุ่มค่าแล้วถูกนำไปเข้ารหัสเพื่อการส่งสัญญาณต่อไป โดยในจุดนี้สัญญาณจะถูกบีบอัดและอาจมีการสูญเสียของข้อมูลไปบ้าง แต่ในเรื่องการกู้กลับอย่างสมบูรณ์จะไม่เกี่ยวข้องกับการบีบอัดข้อมูลดังนั้นช่องว่างตรงนี้จึงไม่มี

ในการแสดงให้เห็นว่า H_0 คือวงจรกรองความถี่ต่ำ และ H_1 คือวงจรกรองความถี่สูง จะแสดงให้เห็นเป็นผลตอบสนองทางความถี่ ดังรูป ซึ่งจะเห็นได้ว่าทั้งสองวงจรไม่เป็นวงจรกรองความถี่ในอุดมคติ เนื่องจากผลตอบสนองมีการซ้อนทับกัน(Overlap) จึงทำให้เกิดการซ้อนทับกันของสเปกตรัม (aliasing) ในแต่ละช่องสัญญาณ รวมถึงยังเกิดการลดทอนสัญญาณในเชิงขนาดและเฟส แต่ในที่นี้เราจะกล่าวถึงเฉพาะการลดทอนสัญญาณในทางขนาดเท่านั้น (amplitude distortion) ดังนั้นในภาครับ วงจรกรองความถี่ด้านสังเคราะห์ (synthesis filter) F_0, F_1 จึงจำเป็นต้องปรับให้เข้ากับภาคส่งด้วย เพื่อที่จะกำจัดค่าผิดพลาดที่จะเกิดขึ้นในวงจรกรองความถี่ด้านวิเคราะห์



รูปที่ 2.19 ผลตอบสนองทางความถี่ของวงจรกรองความถี่ต่ำและสูง

เป้าหมายของเราคือการหาเงื่อนไขในการกู้กลับอย่างสมบูรณ์ ซึ่งจะหมายความว่าฟิลเตอร์แบงค์จะต้องเป็นฟังก์ชันเชิงตั้งฉากคู่(biorthogonal) คือ ทางด้านภาครับตั้งแต่ $F_0, F_1, \uparrow 2$ จะต้องเป็นส่วนกลับของภาคส่ง

หลักการของการกู้กลับอย่างสมบูรณ์ถือเป็นคุณสมบัติที่สำคัญมาก ซึ่งจะประกอบด้วยสองเงื่อนไขคือ การกำจัดการเกิดซ้อนทับกันของสเปกตรัม และการป้องกันไม่ให้เกิดการลดทอนของสัญญาณ ถ้าตัวดำเนินการสุ่มค่าตัวอย่างทั้ง $(\downarrow 2)$และ $(\uparrow 2)$ไม่ถูกแสดง การกู้กลับที่ปราศจากการหน่วงจะหมายถึง $F_0H_0 + F_1H_1 = I$ ซึ่งสามารถเขียนให้อยู่ในรูป Z-domain ได้ดังนี้

$$F_0(z)H_0(z) + F_1(z)H_1(z) = z^{-1} \tag{2.35}$$

ในที่นี้เราสามารถแทนค่า $-z$ ด้วย z เช่นเดียวกับการแทน $\omega + \pi$ ด้วย ω การรวมกันของ $(\downarrow 2)$และ $(\uparrow 2)$จะเป็นการกำจัดค่าศูนย์ออกจากพจน์คี่ จึงคงเหลือไว้เฉพาะพจน์คู่ ดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

$$(\downarrow 2)(\uparrow 2)H_0 x = \frac{1}{2}\left[H_0(z)X(z) + H_0(-z)X(-z)\right] \tag{2.36}$$

ในส่วนของภาครับเราจะทำการคูณเทอมของ $H_0(-z)X(-z)$ด้วย F_0และคูณเทอม $H_1(-z)X(-z)$ด้วย F_1จะทำให้ได้เงื่อนไขของการกำจัดการเกิดซ้อนทับกันของสเปกตรัม(Alias cancellation) ดังสมการ

$$F_0(z)H_0(z) + F_1(-z)H_1(-z) = 0 \tag{2.37}$$

ถ้าเราสมมติให้ $H_0(z) = 1, H_1(z) = z^{-1}$ และ $F_0(z) = z^{-1}, F_1(z) = 1$ ในสมการ จะทำให้ได้ค่าเท่ากับ $2z^{-1}$ ดังสมการ ซึ่งจะเป็นการกำจัดความผิดเพี้ยนของสัญญาณ (No distortion)

$$F_0(z)H_0(z) + F_1(z)H_1(z) = 2z^{-1} \tag{2.38}$$

จากรูปผลตอบสนองทางความถี่จะเห็นได้ว่า ในส่วนของวงจรกรองความถี่สูง $H_1 = 0$ ที่ $z = 1$ หรือที่ $\omega = 0$ ถ้าแทนค่าในสมการ จะได้ $F_0(1)H_0(1) = 2$ และถ้าทำการปรับค่า (Normalize) โดยการคูณด้วย $\sqrt{2}$ และแทนค่าสัมประสิทธิ์ของวงจรกรองความถี่ต่ำและสูงด้วย $C = \sqrt{2}H_0$และ $D = \sqrt{2}H_1$ ผลรวมของสัมประสิทธิ์วงจรกรองความถี่ต่ำ $c(n) = \sqrt{2}h(n)$ จะเท่ากับ $\sqrt{2}$

ถ้าให้สัญญาณอินพุต คือ $x(n)$และวงจรกรองความถี่ฝั่งส่ง คือ H_0 ถ้าอธิบายในเชิง Z-domain เมื่อผ่านการสุ่มค่าตัวอย่างและเพิ่มค่าตัวอย่างจะได้

$$(\downarrow 2) \qquad \frac{1}{2}\left[H_0\left(z^{\frac{1}{2}}\right)X\left(z^{\frac{1}{2}}\right) + H_0\left(-z^{\frac{1}{2}}\right)X\left(-z^{\frac{1}{2}}\right)\right] \tag{2.39}$$

$$(\uparrow 2) \qquad \frac{1}{2}\left[H_0(z)X(z) + H_0(-z)X(-z)\right] \tag{2.40}$$

เมื่อผ่านวงจรกรองความถี่จะทำการคูณด้วย $F_0(z)$ซึ่งจะทำให้ผลที่ออกจากวงจรกรองความถี่ต่ำและวงจรความถี่สูง ดังนี้

วงจรกรองความถี่ต่ำ
$$\frac{1}{2}F_0(z)\left[H_0(z)X(z) + H_0(-z)X(-z)\right] \tag{2.41}$$

วงจรกรองความถี่สูง
$$\frac{1}{2}F_1(z)\left[H_1(z)X(z) + H_1(-z)X(-z)\right] \tag{2.42}$$

หลังจากนั้นนำเอาต์พุตที่ได้จากทั้งสองวงจรมารวมเข้าด้วยกันเพื่อหา $\hat{x}(n)$ หรือในทาง Z-domain คือ $\hat{X}(z)$ดังสมการ

$$\frac{1}{2}\left[F_0(z)H_0(z) + F_1(z)H_1(z)\right]X(z) + \frac{1}{2}\left[F_0(-z)H_0(-z) + F_1(-z)H_1(-z)\right]X(-z) \tag{2.43}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

62428

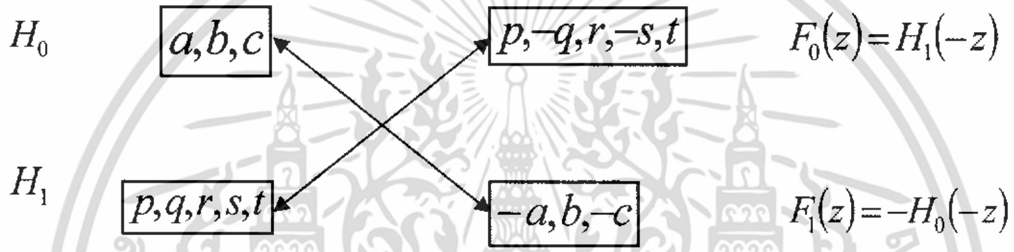
ในการกู้กลับอย่างสมบูรณ์ที่มีการหน่วงเวลา l ครั้ง $\hat{X}(z)$ จะต้องเปลี่ยนเป็น $z^{-l}X(z)$ ดังนั้นเทอมของการลดทอนจะต้องเป็น z^{-l} และเทอมของการซ้อนทับกันของสเปกตรัมจะต้องมีค่าเท่ากับศูนย์

2.5.1 การกำจัดการเกิดการซ้อนทับกันและผลที่ได้จากวงจรกรองความถี่ $P_0 = F_0H_0$

ในตอนนี้เราจะต้องออกแบบวงจรกรองความถี่สี่ตัว H_0, H_1, F_0, F_1 โดยอาศัยเงื่อนไขที่มีความเกี่ยวข้องกันระหว่างวงจรกรองความถี่ฝั่งส่งและฝั่งรับ ดังนี้

$$F_0(z) = H_1(-z) \text{ และ } F_1(z) = -H_0(-z) \tag{2.44}$$

โดยสามารถอธิบายถึงความสัมพันธ์ระหว่าง F_0 กับ H_1 และระหว่าง F_1 กับ H_0 ได้โดยการใช้หลักการใส่เครื่องหมายสลับกัน(Alternating signs) ระหว่างสองฝั่ง ดังนี้



รูปที่ 2.20 แสดงการสลับเครื่องหมายในการหาค่าสัมประสิทธิ์

ผลที่ได้จากวงจรกรองความถี่ต่ำจะเท่ากับ $P_0(z) = F_0(z)H_0(z)$ ส่วนผลที่ได้จากวงจรกรองความถี่สูง คือ $P_1(z) = F_1(z)H_1(z)$ และจากเงื่อนไขความสัมพันธ์ที่กล่าวข้างต้น จะได้ว่า

$$P_1(z) = -H_0(-z)H_1(z) = -H_0(-z)F_0(-z) = -P_0(-z) \tag{2.45}$$

จากเงื่อนไขการกำลัดการลดทอนสัญญาณ $F_0(z)H_0(z) + F_1(z)H_1(z) = 2z^{-l}$ เขียนให้อยู่ในรูปของผลที่ได้จากวงจรกรองความถี่ ดังนี้

$$F_0(z)H_0(z) - F_0(-z)H_0(-z) = P_0(z) - P_0(-z) = 2z^{-l} \tag{2.46}$$

หลักการออกแบบวงจรกรองความถี่ทั้งภาคส่งและภาครับสามารถลดได้เหลือ 2 ขั้นตอน ดังนี้

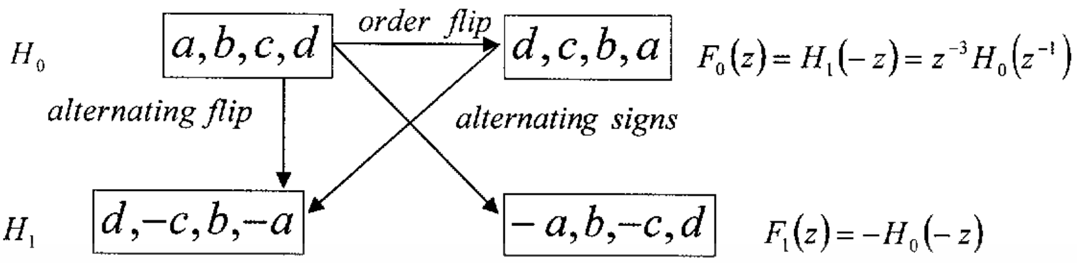
ขั้นแรก : ออกแบบวงจรกรองความถี่ต่ำให้ได้ผลตามสมการ (2.46)

ขั้นสอง : แสดงให้เห็นว่า P_0 เกิดจากผลคูณของ F_0H_0

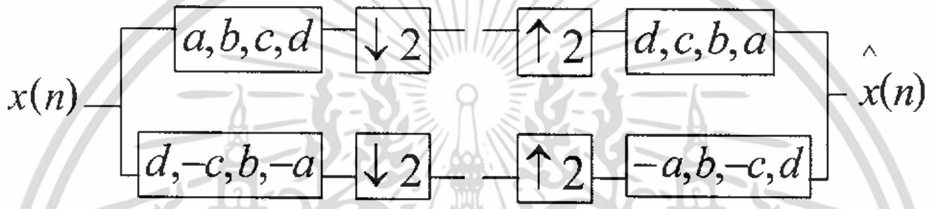
จากนั้นใช้สมการ (2.44) เพื่อหาค่า F_1 และ H_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบฟิลเตอร์แบงค์ทั้งภาครับและภาคส่งให้ตรงตามหลักการการกู้กลับอย่างสมบูรณ์ จะอยู่ภายใต้พื้นฐานของฟังก์ชันเชิงตั้งฉาก ดังนี้



รูปที่ 2.21 แสดงความสัมพันธ์ระหว่างสัมประสิทธิ์ทั้งภาครับและภาคส่ง

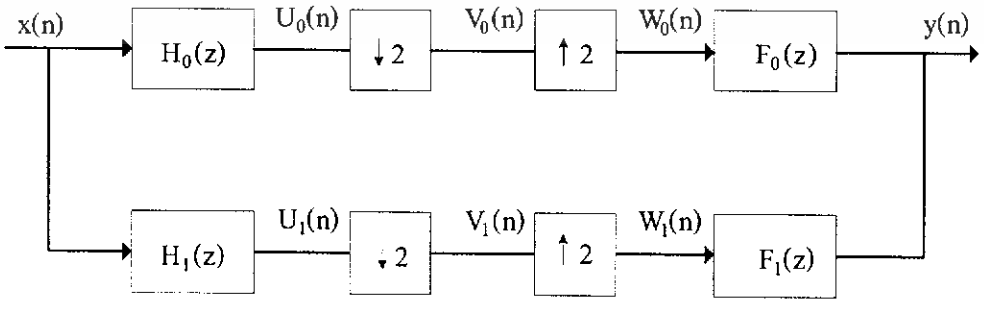


รูปที่ 2.22 ฟิลเตอร์แบงค์เชิงตั้งฉากที่มีสัมประสิทธิ์ 4 ค่า

2.5.2 โครงสร้างของวงจรกรองความถี่ทั้งรูปแบบโดยตรง (Direct Form) และรูปแบบโพลีเฟส (Polyphase Form)

2.5.2.1 วงจรกรองความถี่ที่มีโครงสร้างแบบโดยตรง (Direct Form Filters)

วงจรกรองความถี่รูปแบบโดยตรงในส่วนวิเคราะห์ประกอบไปด้วย เซตของวงจรกรองความถี่ต่ำผ่านและวงจรกรองความถี่สูงผ่านแล้วตามด้วย การสุ่มค่าตัวอย่าง สำหรับวงจรกรองความถี่ในส่วนสังเคราะห์ประกอบไปด้วย การเพิ่มค่าตัวอย่าง ตามด้วย วงจรกรองความถี่ต่ำผ่านและวงจรกรองความถี่สูงผ่าน โดยสามารถเขียนโครงสร้างโดยตรง ดังบล็อกไดอะแกรมดังรูปที่ 2.23



รูปที่ 2.23 บล็อกไดอะแกรมของโครงสร้างโดยตรงของวงจรกรองความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปบล็อกไดอะแกรม ในส่วนวิเคราะห์ $x(n)$ คือสัญญาณอินพุต, H_0 และ F_0 เป็นวงจรกรองความถี่ต่ำผ่าน และความถี่สูงผ่านตามลำดับ $\downarrow 2$ หมายถึงการสุ่มค่าตัวอย่างลดลงครึ่งหนึ่งและ $\uparrow 2$ หมายถึง การเพิ่มค่าตัวอย่างเป็น 2 เท่า เช่นกัน อินพุตนั้น จะถูกกรองด้วยวงจรกรองความถี่ทั้งสองแบนด์ ตามด้วย การสุ่มค่าตัวอย่างลดลงครึ่งหนึ่ง หลังจากนั้นจะได้ เอาต์พุตเป็น $y(n)$ ซึ่งอธิบายในเชิง Z-domain ให้เห็นได้ดังสมการ

$$U_0(z) = H_0(z)X(z) \tag{2.47}$$

$$U_1(z) = H_1(z)X(z) \tag{2.48}$$

$$V_0(z) = \frac{1}{2}\left(U_0(z^{\frac{1}{2}}) + U_1(-z^{\frac{1}{2}})\right)$$

$$= \frac{1}{2}\left(H_0(z^{\frac{1}{2}})X(z^{\frac{1}{2}}) + H_0(-z^{\frac{1}{2}})X(-z^{\frac{1}{2}})\right) \tag{2.49}$$

$$V_1(z) = \frac{1}{2}\left(H_1(z^{\frac{1}{2}})X(z^{\frac{1}{2}}) + H_1(-z^{\frac{1}{2}})X(-z^{\frac{1}{2}})\right) \tag{2.50}$$

$$W_0(z) = V_0(z^2) = \frac{1}{2}\left(H_0(z)X(z) + H_0(-z)X(-z)\right) \tag{2.51}$$

$$W_1(z) = V_1(z^2) = \frac{1}{2}\left(H_1(z)X(z) + H_1(-z)X(-z)\right) \tag{2.52}$$

$$Y(z) = F_0(z)W_0(z) + F_1(z)W_1(z) \tag{2.53}$$

$$Y(z) = \frac{1}{2}F_0(z)\left(H_0(z)X(z) + H_0(-z)X(-z)\right) + \frac{1}{2}F_1(z)\left(H_1(z)X(z) + H_1(-z)X(-z)\right) \tag{2.54}$$

จะเป็นการกู้กลับอย่างสมบูรณ์ ก็ต่อเมื่อ $Y(z) = cz^{-n_0}X(z)$

จัดรูป

$$Y(z) = \frac{1}{2}\left(F_0(z)H_0(z) + F_1(z)H_1(z)\right)X(z) + \frac{1}{2}\left(F_0(z)H_0(-z) + F_1(z)H_1(-z)\right)X(-z) \tag{2.55}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งเงื่อนไข คือ $\frac{1}{2}\Big(F_0(z)H_0(-z)+F_1(z)H_1(-z)\Big)=0$ (Aliasing cancellation)

และ $\frac{1}{2}\Big(F_0(z)H_0(z)+F_1(z)H_1(z)\Big)=cz^{-n_0}$ (No distortion condition)

เอาต์พุตของวงจรนั้นจะสามารถกู้กลับมาโดยในส่วนสังเคราะห์ของวงจร และเพิ่มค่าตัวอย่างเป็น 2 เท่า แล้วผ่านวงจรกรองความถี่จะทำให้ได้สัญญาณเดิมกลับคืนมา

2.5.2.2 วงจรกรองความถี่ที่มีโครงสร้างแบบโพลีเฟส (Polyphase Structure Filters)

ตัวกรองความถี่แบบเอฟไออาร์ (FIR: Finite Impulse Response) ที่แสดงโดยการกระจายฟังก์ชันการส่งผ่านโพลีเฟส และมีผลลัพธ์เป็นโครงสร้างแบบขนาน ซึ่งเมื่อพิจารณา วิธีการนี้จากฟังก์ชันการส่งผ่าน $H(z)$ ที่มี 9 พจน์ดังสมการที่ 2.56

$$H(z)=h[0]+h[1]z^{-1}+h[2]z^{-2}+h[3]z^{-3}+h[4]z^{-4}$$
$$+h[5]z^{-5}+h[6]z^{-6}+h[7]z^{-7}+h[8]z^{-8} \quad (2.56)$$

จากสมการที่ 1 สามารถแบ่งเป็น 2 เทอม คือ เทอมที่มีสัมประสิทธิ์เป็นพจน์กำลังคู่ และ เทอมที่มีสัมประสิทธิ์เป็นพจน์กำลังคี่ดังสมการที่ 2.57 และ 2.58

$$H(z)=(h[0]+h[2]z^{-2}+h[4]z^{-4}+h[6]z^{-6}+h[8]z^{-8})$$
$$+(h[1]z^{-1}+h[3]z^{-3}+h[5]z^{-5}+h[7]z^{-7}) \quad (2.57)$$

$$H(z)=(h[0]+h[2]z^{-2}+h[4]z^{-4}+h[6]z^{-6}+h[8]z^{-8})$$
$$+z^{-1}(h[1]+h[3]z^{-2}+h[5]z^{-4}+h[7]z^{-6}) \quad (2.58)$$

เมื่อกำหนดให้

$$H_e(z)=h[0]+h[2]z^{-2}+h[4]z^{-4}+h[6]z^{-6}+h[8]z^{-8}$$

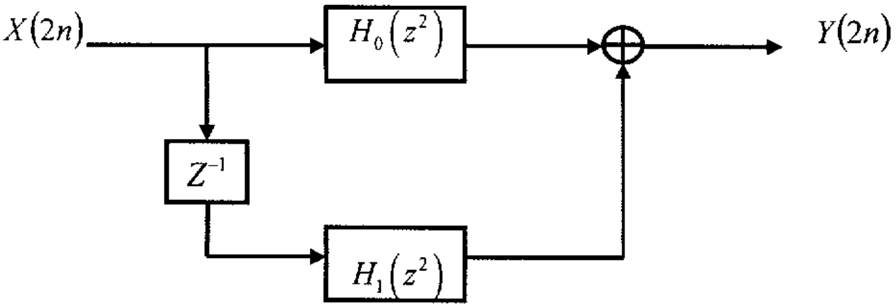
และ $$H_o(z)=h[1]+h[3]z^{-1}+h[5]z^{-2}+h[7]z^{-3} \quad (2.59)$$

จากสมการที่ 2.59 ทำให้จัดรูปแบบสมการที่ 2.58 ใหม่ ดังสมการที่ 2.60

$$H(z)=H_e(z^2)+z^{-1}H_o(z^2) \quad (2.60)$$

จากวิธีการกระจาย $H(z)$ ดังสมการที่ 2.60 เรียกว่า การทำให้เป็นจริงของโพลีเฟส (Polyphase Realization) และจากสมการสามารถเขียนอยู่ในรูปของบล็อกไดอะแกรมดังรูปที่ 2.24

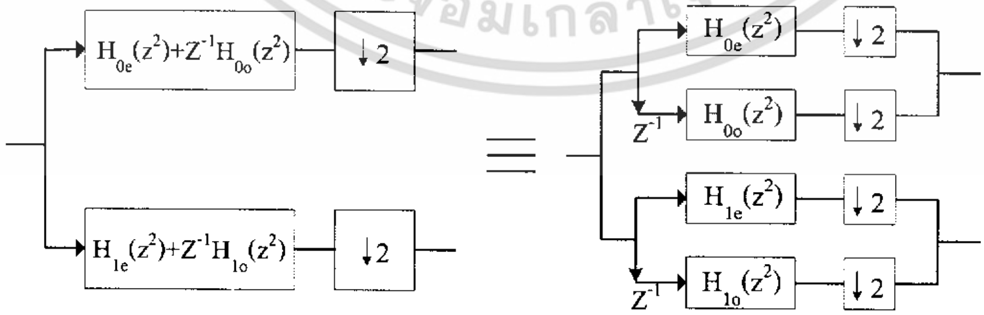
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 บล็อกไดอะแกรม การแสดงให้เห็นจริงของโพลีเฟส ของฟังก์ชันการส่งผ่าน

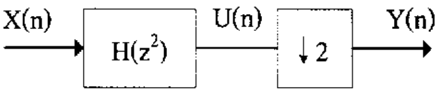
โครงสร้างรูปแบบโพลีเฟส จะมีข้อได้เปรียบ จากรูปแบบโดยตรง สังเกตได้จากในส่วนวิเคราะห์ของฟิลเตอร์แบงค์ของรูปแบบโดยตรง เมื่ออินพุตผ่านการกรองจะประกอบไปด้วย N แซมเปิลเมื่อทำการสุ่มค่าตัวอย่างลดลงครึ่งหนึ่ง แล้ว เราจะใช้เพียง N/2 เท่านั้น ส่วนที่เหลืออีก N/2 เป็นส่วนที่ไม่มีประโยชน์ และจะเห็นได้ว่า เอาต์พุตที่ผ่านวงจรกรองความถี่ต่ำผ่าน และผ่านการสุ่มค่าตัวอย่างแล้วจะประกอบด้วยจำนวนแซมเปิลของ เฟสคู่ของอินพุตเวกเตอร์(X_{even})ทำการคอนโวลูท กับ สัมประสิทธิ์เฟสคู่ของวงจรกรองความถี่ต่ำผ่าน(H_{0even}) และ จำนวนแซมเปิลของ เฟสคี่ของอินพุตเวกเตอร์(X_{odd})ทำการคอนโวลูท กับ สัมประสิทธิ์เฟสคี่ของวงจรกรองความถี่ต่ำผ่าน(H_{0odd}) สำหรับรูปแบบโพลีเฟส สัญญาณอินพุตจะถูกแบ่งเป็น จำนวนแซมเปิล คู่และคี่ โดยอัตโนมัติเมื่อทำการสุ่มค่าตัวอย่างลดลงครึ่งหนึ่ง ในทำนองเดียวกัน สัมประสิทธิ์ของวงจรกรองความถี่ ก็จะถูกแบ่งเป็นส่วน คู่และคี่เช่นกัน ดังนั้น X_{even} ทำการคอนโวลูทกับ H_{0even} และ X_{odd} ก็จะทำการคอนโวลูทกับ H_{0odd} ซึ่งเมื่อนำ 2 เฟสมาบวกกัน ทำให้ได้เอาต์พุตของความถี่ต่ำ ในทำนองเดียวกัน สำหรับวงจรกรองความถี่สูงผ่าน โดยวงจรกรองความถี่สูงผ่าน จะแบ่งเป็นเฟสคู่และคี่, สัมประสิทธิ์ของวงจรกรองความถี่สูงผ่านก็คือ H_{1even} และ H_{1odd}

จากรูปที่ 2.24 เราสามารถแยกพจน์ของ $H_0\left(z^2\right)$ และพจน์ของ $H_1\left(z^2\right)$ ออกเป็นพจน์คี่และพจน์คู่ได้



รูปที่ 2.25 แสดงการแยกโครงสร้างออกเป็นพจน์คี่และพจน์คู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 บล็อคไดอะแกรมก่อนการเปลี่ยนโครงสร้าง

จากรูปที่

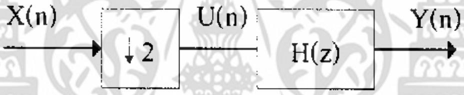
$$U(z) = H(z^2)X(z) \tag{2.61}$$

$$Y(z) = \frac{1}{2}\left(U(z^{\frac{1}{2}}) + U(-z^{\frac{1}{2}})\right)$$

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

$$= \frac{1}{2}H(z)\left(X(z^{\frac{1}{2}}) + X(-z^{\frac{1}{2}})\right) \tag{2.62}$$

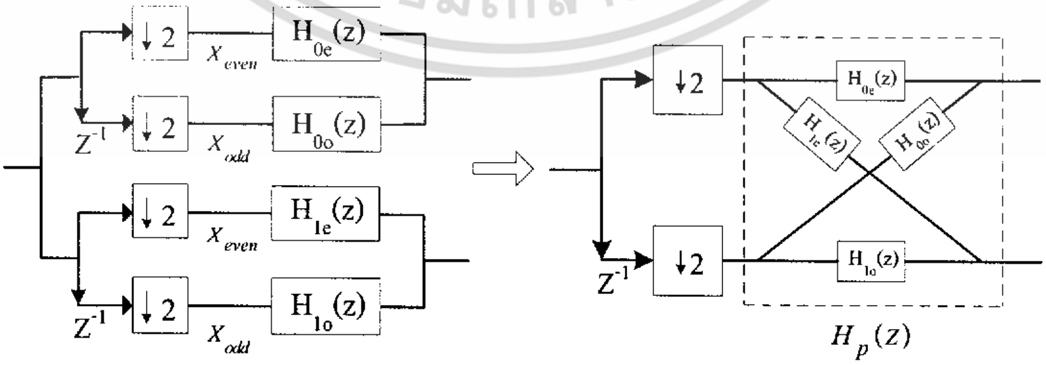
จากสมการข้างต้น เราสามารถเปลี่ยนจากบล็อกไดอะแกรมข้างต้นมาเป็นบล็อกไดอะแกรมข้างล่างได้เป็นบล็อกไดอะแกรมดังรูป



รูปที่ 2.27 บล็อคไดอะแกรมหลังการเปลี่ยนโครงสร้าง

ระบบข้างล่างนี้เหมาะสมมากกว่าระบบข้างบนที่จะนำไป implement ลงบอร์ด FPGA เพราะอินพุตที่เข้ามาคอนโวลูชั่นจะเหลือจำนวนเพียงครึ่งเดียวเมื่อเทียบกับแบบแรก ทำให้เวลาในการประมวลผลจะลดลงและยังช่วยลดการใช้หน่วยความจำในการประมวลผล

ดังนั้นเราสามารถเปลี่ยนโครงสร้างของโพลีเฟสให้เป็นดังรูปใหม่ได้



รูปที่ 2.28 บล็อคไดอะแกรมของโครงสร้างแบบโพลีเฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ $H_p(z)$ คือ โพลีเฟสเมตริกซ์

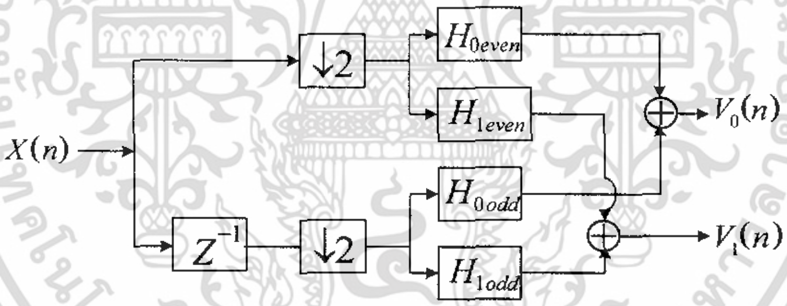
$$H_p(z) = \begin{bmatrix} H_{0e}(z) & H_{0o}(z) \\ H_{1e}(z) & H_{1o}(z) \end{bmatrix}$$

$$\begin{bmatrix} Y_0(z) \\ Y_1(z) \end{bmatrix} = \begin{bmatrix} H_{0e}(z) & H_{0o}(z) \\ H_{1e}(z) & H_{1o}(z) \end{bmatrix} \begin{bmatrix} X_{even} \\ X_{odd} \end{bmatrix}$$

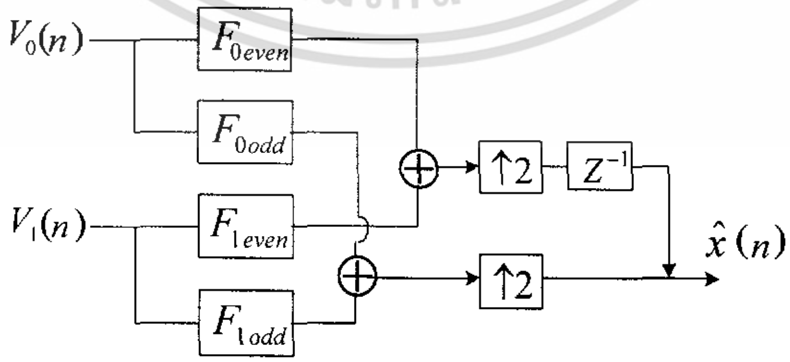
สมการเมตริกซ์ในส่วนวิเคราะห์ของโพลีเฟสจะเป็นไปตามสมการที่ 2.63

$$\begin{bmatrix} H_{0even} H_{0odd} \\ H_{1even} H_{1odd} \end{bmatrix} \times \begin{bmatrix} X_{even} \\ z^{-1} X_{odd} \end{bmatrix} = H_p \begin{bmatrix} X_{even} \\ z^{-1} X_{odd} \end{bmatrix} = \begin{bmatrix} Y_0 \\ Y_1 \end{bmatrix} \tag{2.63}$$

เนื่องจากว่า เทอมคู่และเทอมคี่ ถูกแบ่งให้เท่ากันแล้วนำไปกรองด้วย สัมประสิทธิ์ของวงจรกรองความถี่ ที่เป็นคู่และคี่ อย่างละครึ่งเช่นกัน ทำให้วงจรกรองความถี่ สามารถทำการปรับปรุงประสิทธิภาพของวงจรในทางขนาน ซึ่งแสดงให้เห็นดังรูปที่ 2.28 และ 2.29



รูปที่ 2.29 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนวิเคราะห์



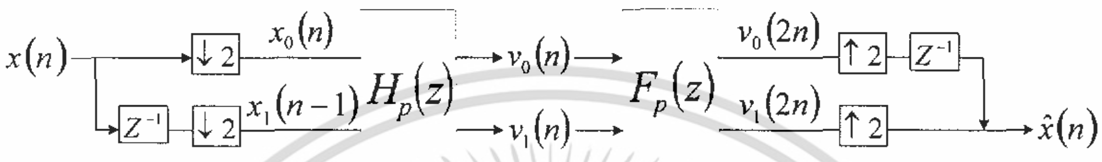
รูปที่ 2.30 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนสังเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.30 ซึ่งเป็นโครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนสังเคราะห์ สามารถเขียนให้อยู่ในรูปของสมการ ดังนี้

$$F(z) = F_1(z^2) + z^{-1}F_0(z^2) \tag{2.64}$$

$$\hat{X}(z) = \begin{bmatrix} z^{-1} & 1 \end{bmatrix} \begin{bmatrix} F_{0,0}(z^2) & F_{1,0}(z^2) \\ F_{0,1}(z^2) & F_{1,1}(z^2) \end{bmatrix} \begin{bmatrix} V_0(z^2) \\ V_1(z^2) \end{bmatrix} \tag{2.65}$$



รูปที่ 2.31 แสดงรูปแบบส่วนวิเคราะห์และส่วนสังเคราะห์ ของฟิลเตอร์แบงค์

สำหรับรูปแบบโดยตรง ส่วนสังเคราะห์ของฟิลเตอร์แบงค์ อินพุตจะทำการเพิ่มค่าตัวอย่าง ด้วยการบวก 0 แล้วจึงนำไปกรอง แต่ในรูปแบบโพลีเฟสนั้น อินพุตจะทำการกรองก่อน แล้วจึงทำการเพิ่มค่าตัวอย่าง เป็นผลทำให้ลดจำนวนของการคำนวณในการกรองความถี่ ลงไปถึงครึ่งหนึ่ง ซึ่งลดลงครึ่งหนึ่งทั้งใน ส่วนวิเคราะห์และส่วนสังเคราะห์ ของฟิลเตอร์แบงค์ ทำให้ประสิทธิภาพทั้งหมดเพิ่มขึ้น 50% ดังนั้นรูปแบบโพลีเฟส จึงแสดงให้เห็นประสิทธิภาพของส่วนอุปกรณ์ด้วย

2.6 โครงสร้างการแปลงเวฟเลตโดยใช้วิธีการ Lifting scheme

สมมติเราสร้างสัญญาณ $f(t)$ โดยสุ่มที่ค่าแซมปลิ้งค่าหนึ่ง เราจะให้ค่าสัญญาณตั้งต้นมีค่าเป็น $\lambda_0, k = f(k)$ โดย $k \in Z$ แทนสัญญาณที่เริ่มทดลอง เราต้องการทราบว่ามันเป็นไปได้หรือไม่ที่จะแทนสัญญาณตัวนี้ด้วยค่าสัมประสิทธิ์เพียงไม่กี่ตัว พูดอีกอย่างคือสัมประสิทธิ์ที่มีค่าของแซมปลิ้งใหญ่กว่าเดิม การแทนสัญญาณแบบนี้เป็นสิ่งจำเป็นที่เอาไปใช้กับการบีบอัดข้อมูล ซึ่งในบางครั้งมันอาจจะเป็นไปไม่ได้ที่จะได้ค่าสัมประสิทธิ์ค่าน้อยๆที่มาแทนสัญญาณตัวหนึ่ง แต่ก็ดีกว่าการประมาณสัญญาณโดยไม่สนใจค่าความผิดพลาดเลย ดังนั้นเราจึงต้องการการควบคุมที่แม่นยำในการแทนข้อมูลที่ผิดพลาดด้วยสัมประสิทธิ์จำนวนน้อยๆ สังเกตได้ว่าเราต้องการให้ค่าความแตกต่างระหว่างสัญญาณตั้งต้นกับสัญญาณที่ประมาณค่าออกมามีความแตกต่างกันน้อยที่สุด

เราสามารถลดจำนวนสัมประสิทธิ์สัญญาณโดยสุ่มเอาแต่จำนวนคู่และสามารถเขียนเป็นสมการใหม่เป็น

$$\lambda_{-1,k} = \lambda_{0,2k} \quad \text{โดย} \quad k \in Z \tag{2.66}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราต้องการจะรู้ว่ามีจำนวนข้อมูลเท่าไหร่ที่สูญหาย พูดอีกอย่างก็คือเราต้องการข้อมูลพิเศษที่จะนำมาใช้กู้สัญญาณตั้งต้น $\{\lambda_{0,k}\}$ จาก $\{\lambda_{-1,k}\}$ โดยเราจะใช้สัมประสิทธิ์ $\{\gamma_{-1,k}\}$ มาเข้ารหัส ส่วนข้อมูลที่ต่างจากสัญญาณตั้งต้นกับสัญญาณที่ประมาณขึ้นมาโดยให้เป็นสัมประสิทธิ์เวฟเลต อันที่จริงแล้วมีหลายวิธีที่ทำได้ซึ่งขึ้นอยู่กับข้อมูลทางสถิติของสัญญาณ โดยวิธีที่เหมาะสมนั้นจะได้สัมประสิทธิ์เวฟเลตที่มีขนาดเล็กกว่า

โดยจะกล่าวว่าข้อมูลที่หายไปนั้นก็อยู่ในสัมประสิทธิ์จำนวนคี่นั่นเอง $\gamma_{-1,k} = \lambda_{0,2k+1}$ โดย $k \in Z$ ซึ่งตัวเลือกนี้ก็จะสอดคล้องกับวิธีการ Lazy wavelet จริงๆแล้ววิธีการนี้ไม่ยุ่งยากอะไรเพียงแค่แซมปลิ้งสัญญาณให้อยู่ในรูปแซมเปิลคู่และคี่ สังเกตได้ว่าเราไม่ได้ทำอะไรยุ่งยาก สัมประสิทธิ์เวฟเลตจะมีขนาดเล็กในกรณีที่แซมเปิลคี่มีขนาดเล็ก และจะต้องใช้วิธีการ Lazy wavelet มาร่วมด้วย

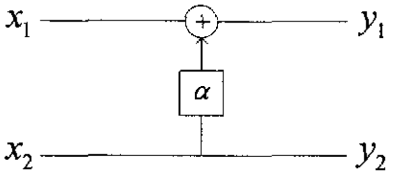
เราจะใช้วิธีการที่ซับซ้อนขึ้นในการกู้สัญญาณตั้งต้น $\{\lambda_{0,k}\}$ จากแซมเปิลย่อย $\{\lambda_{-1,k}\}$ จากแซมเปิลคู่ $\{\lambda_{0,2k}\}$ จากสมการข้างต้นเราจะพบว่า $\lambda_{0,2k} = \lambda_{1,k}$ แต่เราสามารถทำนายได้ว่าแซมเปิลคี่นั้นมาจาก $\{\lambda_{-1,k}\}$ เช่นกัน สมมติให้แซมเปิลที่อยู่ติดกันมีความสัมพันธ์กัน โดยเราจะให้แซมเปิลคี่ $\lambda_{0,2k+1}$ เป็นค่าเฉลี่ยระหว่างสองแซมเปิลที่ติดกัน $\lambda_{-1,k}$ กับ $\lambda_{-1,k+1}$ เราไม่ได้ต้องการให้ถูกที่สุดเพราะสัมประสิทธิ์เวฟเลตจะเข้ารหัสค่าความแตกต่างของสัญญาณตั้งต้นและสัญญาณที่เราประมาณค่า ตามสมการ

$$\gamma_{-1,k} = \lambda_{0,2k+1} - \frac{1}{2}\left(\lambda_{-1,k} + \lambda_{-1,k+1}\right) \tag{2.67}$$

ถ้าสัญญาณนั้นสัมพันธ์กันด้วยแบบไหนก็ตาม ส่วนใหญ่แล้วค่าสัมประสิทธิ์เวฟเลตที่แทนตัวสัญญาณที่ต่างกันจะมีค่าน้อย ค่าสัมประสิทธิ์ตัวใดที่มีค่าต่ำกว่าค่าความผิดพลาดที่จุดเริ่มต้นเราจะไม่สนใจ โดยวิธีนี้เราจะได้สัมประสิทธิ์ที่แทนสัญญาณที่ดีกว่า ถ้าสัญญาณตั้งต้นกับสัญญาณที่แปลงค่าแล้วเหมือนกัน สัมประสิทธิ์เวฟเลตทั้งหมดจะมีค่าเป็นศูนย์

Lifting scheme เป็นการแยกแฟคเตอร์โพลีเฟสเมตริก
จากที่ได้กล่าวมาข้างต้นเราสามารถเขียนสมการโพลีเฟสในรูปเมตริกซ์ได้
กำหนดให้

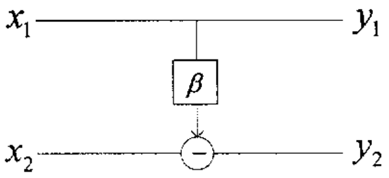
เมตริกซ์ $\begin{bmatrix} 1 & \alpha \\ 0 & 1 \end{bmatrix}$ แทนด้วยสัญลักษณ์



รูปที่ 2.32 บล็อกไดอะแกรมแทนเมตริกซ์ α

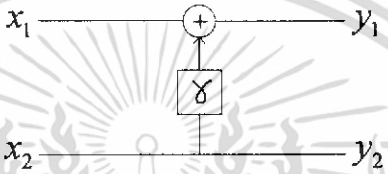
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมตริกซ์ $\begin{bmatrix} 1 & 0 \\ \beta & 1 \end{bmatrix}$ แทนด้วยสัญลักษณ์



รูปที่ 2.33 บล็อกไดอะแกรมแทนเมตริกซ์ β

เมตริกซ์ $\begin{bmatrix} 1 & \gamma \\ 0 & 1 \end{bmatrix}$ แทนด้วยสัญลักษณ์



รูปที่ 2.34 บล็อกไดอะแกรมแทนเมตริกซ์ γ

จากสมการที่ สมการโพลีเฟส สามารถเขียนอยู่ในรูปเมตริก ให้เป็น $\begin{bmatrix} c_0 & -s_0 \\ s_0 & c_0 \end{bmatrix}$

นำเมตริกซ์ $\begin{bmatrix} c_0 & -s_0 \\ s_0 & c_0 \end{bmatrix}$ มาแยกแฟคเตอร์ให้อยู่ในรูปเมตริกซ์ย่อย α , β , γ

$$\begin{bmatrix} c_0 & -s_0 \\ s_0 & c_0 \end{bmatrix} = \begin{bmatrix} 1+\alpha\beta & \alpha+\gamma+\alpha\beta\gamma \\ \beta & 1+\beta\gamma \end{bmatrix} \tag{2.68}$$

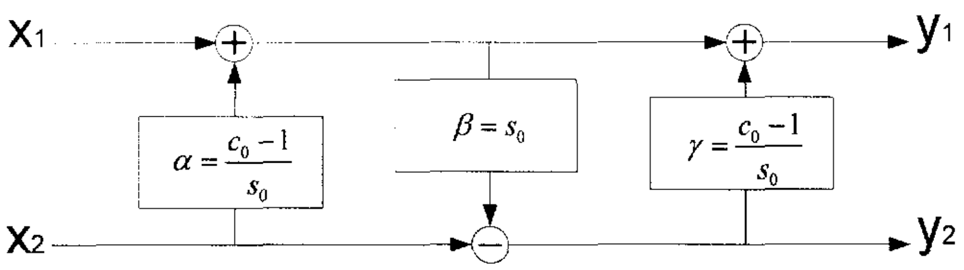
$$\begin{bmatrix} c_0 & -s_0 \\ s_0 & c_0 \end{bmatrix} = \begin{bmatrix} 1+\alpha\beta & \alpha+\gamma+\alpha\beta\gamma \\ \beta & 1+\beta\gamma \end{bmatrix} \tag{2.69}$$

สามาค่า α , β , γ ดังนี้

$$\alpha = \frac{c_0 - 1}{s_0} \quad , \quad \beta = s_0 \quad \text{และ} \quad \gamma = \frac{c_0 - 1}{s_0}$$

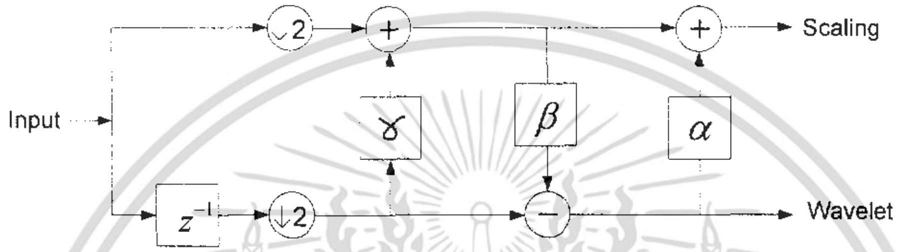
นำมาเขียนโครงสร้างใหม่ได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.35 แสดงโครงสร้างลิฟท์ติ้งที่แปลงมาจากโพลีเฟสเมตริกซ์

นำมาเขียนโครงสร้างของ Lifting scheme ทั่วไปได้เป็น



รูปที่ 2.36 แสดงโครงสร้างทั่วไปของ Lifting scheme ทางภาควิเคราะห์

2.7 ภาษาวีเอชดีแอล

ความซับซ้อนและขนาดของระบบดิจิตอลในปัจจุบันได้เพิ่มมากขึ้นทุกขณะ ส่งผลให้มีการนำคอมพิวเตอร์เพื่อช่วยในการออกแบบ มาใช้ในกระบวนการออกแบบฮาร์ดแวร์เพิ่มขึ้น อีกทั้งอุปกรณ์และวิธีการออกแบบใหม่ๆ ก็ถูกพัฒนาขึ้นมาเพื่อช่วยอำนวยความสะดวกให้กับนักออกแบบมากขึ้นด้วย สำหรับภาษาบรรยายอุปกรณ์ฮาร์ดแวร์ เอชดีแอล (HDL: Hardware Description Language) ก็เป็นเครื่องมืออย่างหนึ่งที่ได้รับการพัฒนามาอย่างต่อเนื่องเพื่อช่วยในการปรับปรุงขบวนการออกแบบระบบดิจิตอลให้เป็นไปอย่างมีประสิทธิภาพ

2.7.1 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 นอกจากนี้วีเอชดีแอลยังเป็นภาษาที่สนับสนุนลักษณะต่างๆ ของระบบดิจิตอลที่มีความซับซ้อนได้ทั้งหมด ดังนั้น วีเอชดีแอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงเป็นภาษาที่น่าสนใจในการศึกษาและนำไปใช้งานเป็นอย่างยิ่ง สำหรับมาตรฐานของภาษาที่ใช้บรรยายพฤติกรรมวงจรหรือฮาร์ดแวร์ สามารถสรุปได้ดังนี้

-ต้องเป็นภาษาที่นำไปเขียนรูปแบบระบบดิจิตอลและมีคุณสมบัติที่สามารถเข้าใจได้ทั้งมนุษย์และเครื่องคอมพิวเตอร์โดยไม่ต้องมีการแปลหรือเปลี่ยนแปลงอีก

- สามารถนำไปใช้เป็นเอกสารประกอบโครงการได้

- ต้องเป็นภาษาที่เขียนขึ้นสำหรับใช้จำลองการทำงานของวงจร

ฉะนั้นภาษาดังกล่าวนี้จึงจัดเป็นภาษาโปรแกรมระดับสูง เช่น เดียวกับภาษาปาสคาล หรือภาษาซี ซึ่งในทางวิศวกรรมภาษาที่ใช้ในการออกแบบฮาร์ดแวร์นี้เรียกว่า ภาษาโปรแกรมระดับสูง

2.7.2 การออกแบบระบบดิจิตอล

ในการออกแบบระบบดิจิตอล เริ่มตั้งแต่การกำเนิดแนวความคิดเบื้องต้นจนกระทั่งได้ออกมาเป็นอุปกรณ์ฮาร์ดแวร์ที่ใช้งานได้จะต้องผ่านขั้นตอนต่างๆ มากมาย และในแต่ละขั้นตอนผู้ออกแบบจะต้องตรวจสอบผลลัพธ์ในแต่ละขั้นก่อนเข้าสู่กระบวนการออกแบบในขั้นต่อไป

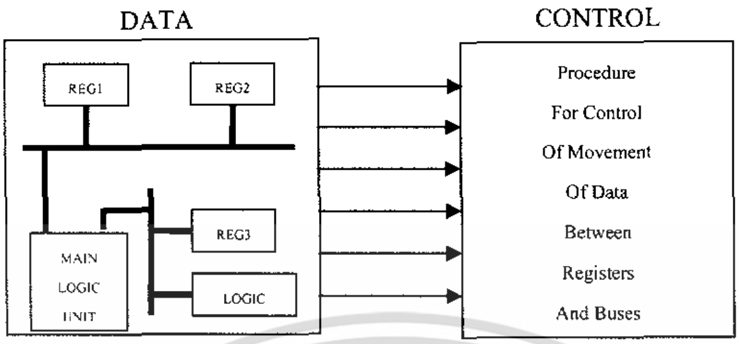
รูปที่ 2.37 แสดงขั้นตอนปกติที่ใช้ในการออกแบบ แล้วทำการพัฒนาให้สามารถนำไปใช้ได้อย่างสมบูรณ์ ซึ่งภายในจะทำการสร้างรูปแบบเชิงพฤติกรรมขึ้นมาตรวจสอบ ซึ่งอาจจะเป็นผังงาน ขั้นตอนนี้ผู้ออกแบบจำเป็นต้องสร้างรูปแบบระบบแสดงผล หรือรหัสคำสั่งเทียม (Psedo code) ก็ได้

รูปที่ 2.37 แสดงขั้นตอนการออกแบบระบบดิจิตอล

ขั้นตอนต่อไปเป็นการออกแบบระบบเส้นทางของข้อมูล ผู้ออกแบบจะกำหนดส่วนประกอบของรีจิสเตอร์และวงจรลอจิกที่จำเป็นทั้งหมด เพื่อนำมาประกอบเป็นระบบที่สมบูรณ์ โดยแต่ละองค์ประกอบสามารถเชื่อมต่อกันด้วยบัสหนึ่งหรือสองทิศทาง (Unidirectional or Bidirectional Bus) กระบวนการใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมการเคลื่อนย้ายข้อมูลระหว่างรีจิสเตอร์และวงจรลอจิกจะขึ้นอยู่กับพฤติกรรมของระบบที่กำหนดไว้ดังรูปที่ 2.38



รูปที่ 2.38 แสดงการออกแบบระบบเส้นทางข้อมูล

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

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

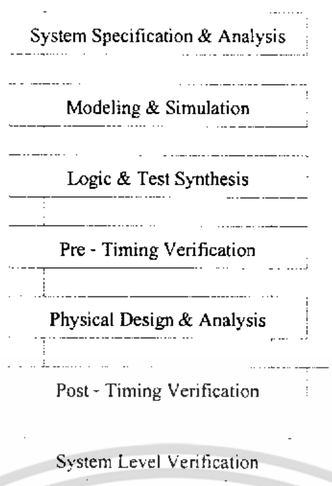
และในขั้นตอนสุดท้ายจะเป็นการส่งระบบที่ออกแบบไว้ไปทำการเจือที่โรงงานเพื่อผลิตออกมาเป็นวงจรรวมในที่สุด

2.7.3 การออกแบบจากบนลงล่าง (Top-Down Design)

ในการพัฒนาวงจรรวมดิจิตอลขนาดใหญ่ที่มีความซับซ้อน วิศวกรหรือผู้ออกแบบมักจะมองการออกแบบให้อยู่ในรูปของบล็อกไดอะแกรมก่อนที่ทำวิเคราะห์ให้ลึกถึงรายละเอียดต่อไป ซึ่งภาษาวีเอชดีแอลนั้นอนุญาตให้อธิบายและวิเคราะห์การทำงานของแต่ละบล็อก รวมถึงการปรับปรุงการทำงานจากผลที่วิเคราะห์เพื่อให้ได้การทำงานตามต้องการนอกจากนี้ยังสามารถเพิ่มเติมในรายละเอียดในแต่ละขั้นตอนได้ ซึ่งหลักการนี้สอดคล้องกับหลักการออกแบบจากบนลงล่างนั่นเอง ถ้าทดลองเปรียบเทียบกับการออกแบบจากล่างขึ้นบ (Bottom-Up Design) จะเห็นได้ว่า การออกแบบจากล่างขึ้นบนจะใช้เวลาการออกแบบมากกว่า 90% เนื่องจากเป็นการวางวงจรด้วยอุปกรณ์ต่างๆ (Schematic Capture) ที่ประกอบกันเข้าเป็นวงจรที่ต้องการออกแบบก่อน แล้วจึงทำการจำลองการทำงาน และตรวจสอบความถูกต้อง

วีเอชดีแอลกับหลักการออกแบบจากบนลงล่างจึงเป็นทางออกให้กับวิศวกรให้สามารถ ออกแบบและพัฒนาวงจรที่มีความซับซ้อนได้มากขึ้น ทั้งยังช่วยลดเวลาและค่าใช้จ่ายในการออกแบบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.39 แสดงขั้นตอนการออกแบบจากบนลงล่าง

จากรูปที่ 2.39 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 สำหรับใช้ในการบรรยายพฤติกรรมการทำงาน พร้อมทั้งทำการจำลองการทำงาน เพื่อเปรียบเทียบและตรวจสอบความถูกต้องกับข้อกำหนด

3) ลอจิกและการทดสอบการสังเคราะห์ คือ หลังจากที่ได้หลักการขั้นต้นพร้อมแนวความคิดที่ผ่านการตรวจสอบแล้วหลักการนี้จะถูกเพิ่มเติมรายละเอียดลงมาเป็นลำดับขั้นที่สอง จนกระทั่งอยู่ในระดับที่จะนำไปผลิตวงจรจริง หรือทำการสังเคราะห์ในขั้นตอนนี้เองเทคโนโลยีที่จะมารองรับวงจรออกแบบจะถูกกำหนดขึ้น และระบบช่วยการออกแบบจะทำการสังเคราะห์วงจรที่ได้จากรูปแบบที่จะเขียนขึ้นให้อยู่ในรูปของวงจรที่ประกอบด้วยอุปกรณ์อิเล็กทรอนิกส์ หรือวงจรในระดับเกท และการเชื่อมต่อระหว่างกันของอุปกรณ์เหล่านั้นหรือไม่ก็อยู่ในรูปของโครงข่ายการเชื่อมต่อ ที่สามารถนำไปผลิตอุปกรณ์อื่นได้

4) การตรวจสอบเวลาก่อนการออกแบบ คือ หลังจากได้ทำการสังเคราะห์วงจรให้อยู่ในระดับเกท หรือ โครงข่ายการเชื่อมต่อแล้ว ข้อมูลนี้จะถูกนำไปใช้สำหรับการจำลองการทำงานในเรื่องความถูกต้องของฟังก์ชันพร้อมกับนำข้อมูลที่เกี่ยวข้องกับเวลาเข้ามาใช้ในการประกอบในการพิจารณาด้วย ซึ่งตามปกติแล้วอุปกรณ์ ทางอิเล็กทรอนิกส์ทุกชิ้นจะต้องมีเวลาหน่วงของการแพร่กระจาย (Propagation Delay Time) เสมอ ถึงแม้ว่าจะเป็น เวลาที่น้อยมากในระดับนาโนวินาทีก็ตาม แต่ถ้าภายในวงจรหนึ่งประกอบด้วยเกทของฟังก์ชันต่างๆ จำนวน 10,000 เกท ขึ้นไป เวลาดังกล่าวนี้จะสะสมกันมากขึ้น จนอาจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้การทำงานของวงจรรวมทั้งหมดผิดพลาดไป หรือไม่สามารถที่จะทำงานในย่านความถี่สัญญาณนาฬิกาที่สูงได้

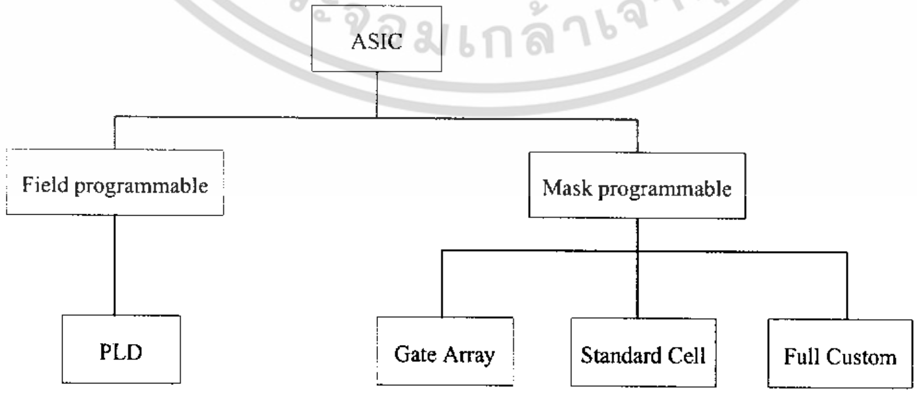
5) การออกแบบทางกายภาพและการวิเคราะห์ คือ ขั้นตอนในการผลิตเป็นวงจรจริง (Technology and device mapping) โดยจะนำข้อมูลที่ได้จากการสังเคราะห์ มาใช้ในการผลิตเป็นวงจรรวม ซึ่งอาจจะอยู่ในรูปของแผงวงจรไฟฟ้า ที่ประกอบด้วยอุปกรณ์หลายๆ ชิ้นหรืออยู่ในรูปของวงจรรวมเอซิก (ASIC)

6) การตรวจสอบเวลาหลังการออกแบบ คือ การทำการตรวจสอบการทำงานด้วยตัวแปรทางด้านเวลาทั้งหมด เพื่อความถูกต้องของวงจรเป็นครั้งสุดท้ายก่อนนำไปรวมเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบดิจิตอล เนื่องจากในขั้นตอนนี้ วงจรที่ออกแบบ จะประกอบด้วยจุดต่อทางอินพุตและเอาท์พุท ซึ่งเป็นจุดต่อสำหรับการรับและส่งสัญญาณกับภายนอก

7) การตรวจสอบระบบ คือ การนำวงจรที่ออกแบบไว้ประกอบเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบที่สมบูรณ์ แล้วทำการทดสอบการทำงานทั้งระบบรวมกับอุปกรณ์อื่นๆ อีกครั้งเพื่อควบคุมคุณภาพของผลิตภัณฑ์

2.8 เอฟพีจีเอ

เทคโนโลยีความก้าวหน้าของอุตสาหกรรมอิเล็กทรอนิกส์ในปัจจุบัน ทำให้เกิดการพัฒนาความสามารถของอุปกรณ์ต่างๆ ซึ่งทำให้ลดค่าใช้จ่ายต่างๆ ได้มาก ในขณะเดียวกันก็มีการเพิ่มประสิทธิภาพและระดับความเชื่อถือได้ของวงจรรวมที่สูงขึ้นเห็นได้ชัดจากเทคโนโลยีไมโครโปรเซสเซอร์ และหน่วยความจำปัจจุบัน ทุกๆครั้งที่มีการพัฒนาขึ้นทำให้เกิดช่องว่างระหว่างวงจรรวมและไอซีมาตรฐานมากขึ้น นักออกแบบอุปกรณ์ทางด้านดิจิตอล ได้พิจารณาถึงการผลิตให้มีขนาดมากขึ้นและการผลิตวงจรรวมเอซิก (ASIC: Application Specific Integrated Circuit) ซึ่งวงจรรวมจะแบ่งตามโครงสร้างออกเป็น 2 กลุ่ม คือ ฟิลด์โปรแกรมเมเบิล (Field programmable) และ แมสโปรแกรมเมเบิล (Mask programmable) ดังแสดงในรูปที่ 2.40



รูปที่ 2.40 แสดงผังการแบ่งกลุ่มของวงจรรวมเอซิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.1 การออกแบบวงจรเชิงเลขด้วยชิพอุปกรณ์เอฟพีจีเอ

ชิพอุปกรณ์เอฟพีจีเอ เป็นอุปกรณ์ที่ใช้ในการโปรแกรมวงจรที่ได้ออกแบบลงไปเพื่อให้อุปกรณ์เอฟพีจีเอ มีฟังก์ชั่นการทำงานตามที่ออกแบบไว้ ในการทำชิพอุปกรณ์ ซึ่งเป็นวิธีการออกแบบ ไอซี (IC : Integrated Circuit) แบบ เซมิคัสตัม (Semi custom) อีกวิธีหนึ่ง เมื่อเทียบกับการทำเอซิก แล้วนั้นก็มีทั้งข้อดีและข้อเสีย คือ การทำชิพอุปกรณ์เอฟพีจีเอ จะมีข้อจำกัดในด้านขนาดของวงจรเพราะภายในชิพอุปกรณ์เอฟพีจีเอจะมีจำนวนเกท (gate) ให้ใช้จำนวนจำกัด และการทำชิพอุปกรณ์เอฟพีจีเอ ก็เหมาะสำหรับการทำผลิตภัณฑ์ต้นแบบหรือเพื่อผลิตในปริมาณต่ำ ส่วนข้อดีของการทำชิพอุปกรณ์ก็คือระยะเวลาที่ใช้ในการทำตั้งแต่เขียนรหัส (code) อธิบายฮาร์ดแวร์จนกระทั่งดาวน์โหลด (download) นั้นน้อยกว่าการทำเอซิก มากและการตรวจสอบหรือแก้ไขการออกแบบที่ทำได้สะดวก

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

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

2.8.2 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว

1. ผู้ออกแบบไม่จำเป็นต้องทราบถึงโครงสร้างภายในของตัวชิพเพียงแต่มีความรู้เกี่ยวกับขั้นตอนการออกแบบลอจิกก็เพียงพอแล้ว ต่างกับการใช้ไมโครโปรเซสเซอร์ซึ่งจำเป็นต้องศึกษาโครงสร้างภายในรวมถึงภาษาแอสแซมบลี (Assembly) ของไมโครโปรเซสเซอร์ตัวนั้นด้วย

2. มีการออกแบบโดยใช้ภาษาในการอธิบายการทำงานของวงจรหรือเอชดีแอล เป็นเครื่องมือในการออกแบบ ซึ่งเป็นวิธีการที่มีความยืดหยุ่นสูง ทำได้รวดเร็ว และ ไม่จำเป็นต้องทราบถึงลักษณะของวงจรที่ต้องการว่าจะเชื่อมต่อกันอย่างไร เพียงแต่กำหนดลักษณะการทำงานให้มันจากนั้นตัวซอฟต์แวร์จะทำการสังเคราะห์ ให้ทั้งหมด นอกจากนี้ภาษาที่ใช้ยังเป็นมาตรฐานเดียวกันสามารถใช้ได้กับชิพทุกตัวและทุกบริษัท

3. การโปรแกรมสามารถทำได้เองและใช้เวลาไม่นาน เพียงแค่ส่งข้อมูลผ่านสายดาวน์โหลดทางพอร์ตของคอมพิวเตอร์ก็สามารถโปรแกรมตัวชิพขณะที่อยู่ในระบบได้โดยไม่จำเป็นต้องถอดมโปรแกรมข้างนอก และที่สำคัญสามารถโปรแกรมได้หลายครั้ง จึงทำให้ง่ายในการแก้ไขและพัฒนาโดยไม่ต้องเสียค่าใช้จ่ายเพิ่มแต่อย่างใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย

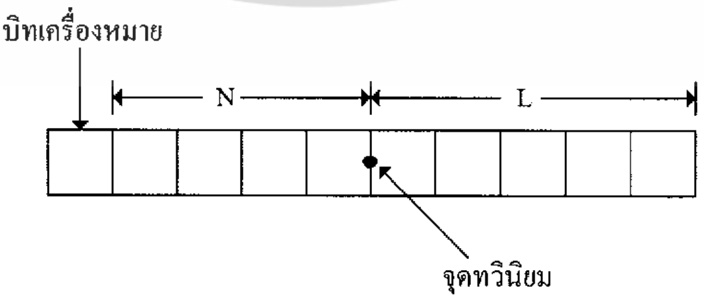
โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) หรือเรียกอย่างย่อๆว่า "DA" เป็นการจัดรูปแบบทางคณิตศาสตร์ของสมการที่อยู่ในรูปของผลบวกของผลคูณ (Sum of Product) ของเลขฐานสิบให้กระจายออกเป็นบิตหรือในรูปแบบของเลขฐานสอง เพื่อให้การคำนวณทางคณิตศาสตร์สามารถแปลงเป็นวงจรดิจิตอลอิเล็กทรอนิกส์ได้ โดยจะปรากฏอยู่ในงานทางด้านการประมวลผลสัญญาณเชิงเลข หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจายที่นำมาใช้งานด้านการประมวลผลสัญญาณเชิงเลข คือการแปลงฟังก์ชันถ่ายโอน (Transfer Function) ซึ่งเป็นสมการที่อยู่ในรูปผลบวกของผลคูณระหว่างสัญญาณอินพุตกับค่าสัมประสิทธิ์ของฟังก์ชั่นถ่ายโอนของระบบ โดยในการคูณกันระหว่างค่าสัมประสิทธิ์ของฟังก์ชั่นถ่ายโอนกับสัญญาณอินพุต จะใช้การคูณเลขฐานสองแบบส่วนเติมเต็มสอง (2's complement) และการคูณจะใช้แบบเปิดตาราง (Look-up table) โดยค่าผลบวกของผลคูณระหว่างสัมประสิทธิ์และสัญญาณอินพุตจะถูกเก็บไว้ในหน่วยความจำ EPROM และจะใช้สัญญาณอินพุตเป็นแอดเดรสของ EPROMโดยตรง ทั้งค่าสัมประสิทธิ์ของวงจรกรองและสัญญาณอินพุตจะถูกแทนด้วยเลขส่วนเติมเต็มสอง ดังนั้นโครงสร้างเลขคณิตกระจายจึงมีทฤษฎีพื้นฐานอยู่บนการคูณแบบเลขส่วนเติมเต็มสอง (2's complement Multiplication)

2.9.1 ระบบตัวเลข

สำหรับระบบเชิงเลข ตัวเลขต่างๆจะถูกแทนด้วยเลขฐานสอง ซึ่งโดยทั่วไปมีรูปแบบที่นิยมใช้กันอยู่ 2 รูปแบบ คือ รูปแบบจำนวนโดยตรง (Fixed point format) และ รูปแบบจำนวนอิงดรรชนี (Floating point format) ซึ่งรูปแบบจำนวนโดยตรงจะมีวงจรฮาร์ดแวร์ที่ใช้ในการคำนวณที่ง่ายกว่า แต่ให้ค่าจากการคูณค่อนข้างจำกัด ส่วนรูปแบบจำนวนอิงดรรชนีจะสามารถแทนค่าของสัญญาณ คือให้ย่านพลวัตร (Dynamic range) ได้มากกว่า แต่ต้องใช้วงจรฮาร์ดแวร์ที่สลับซับซ้อน แพงกว่า และให้ความเร็วในการประมวลผลที่ลดลง

2.9.1.1 รูปแบบจำนวนโดยตรง

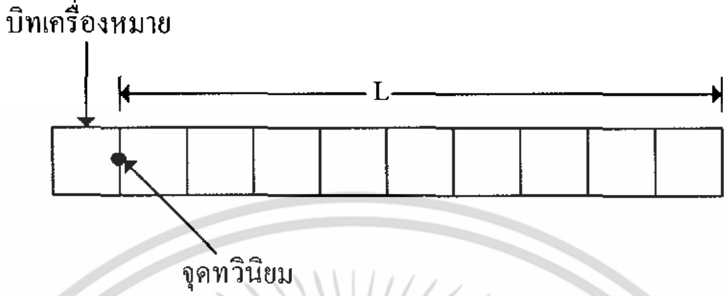
รูปแบบจำนวนโดยตรงปกติจะประกอบไปด้วย 3 ส่วน คือ บิตเครื่องหมาย (Sign bit) 1 บิต, บิตจำนวนเต็ม (Integer bit) N บิต และบิตเศษส่วน (Fractional bit) L บิต โดยจะมีจุดทวินิยม (Binary point) อยู่ระหว่างบิตจำนวนเต็มและบิตเศษส่วนดังแสดงในรูปที่ 2.41



รูปที่ 2.41 แสดงการจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิตจำนวนเต็มและบิตเศษส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนบิต N เป็นตัวกำหนดย่านพลวัตรที่ต้องการ โดยถ้าเลือกให้มีจำนวนน้อยอาจทำให้เกิดการล้น (Overflow) จากการคำนวณได้ แต่ถ้าเลือกให้มีจำนวนมากความเที่ยงตรงก็จะน้อยลง ซึ่งในการสร้างวงจรกรองสัญญาณเชิงเลขโดยการแทนด้วยรูปแบบจำนวนโดยตรงนั้น นิยมที่จะทำมาตราส่วน (Scaling) เพื่อให้ขนาดของสัญญาณมีค่าอยู่ระหว่าง $-1 \le x < 1$ คือมีบิตเครื่องหมาย 1 บิต และบิตเศษส่วน L บิต ดังแสดงในรูปที่ 2.42



รูปที่ 2.42 แสดงการจัดรูปแบบจำนวนโดยตรงที่มีแต่บิตเศษส่วน

โดยทั่วไปเลขฐานสองแบบจำนวนโดยตรงแบ่งออกได้เป็น 3 รูปแบบด้วยกัน คือ (1) แบบขนาดและเครื่องหมาย (Sign magnitude), (2) แบบส่วนเติมเต็มหนึ่ง (1's complement) และ (3) แบบส่วนเติมเต็มสอง (2's complement) โดยคุณลักษณะที่สำคัญบางประการของการแทนตัวเลขด้วยเลขฐานสองแบบจำนวนโดยตรงทั้ง 3 รูปแบบสามารถสรุปได้ดังตารางที่ 2.1

Features	Sign and magnitude	2' complement	1'complement
Range	$-(1-2^{-L}) \le x \le (1-2^{-L})$	$-1 \le x \le (1-2^{-L})$	$-(1-2^{-L}) \le x \le (1-2^{-L})$
Representation of zero	0.000 and 1.000	0.000	0.000 and 1.111
Arithmetic rules	Simple must be kept track of, separately	Simple; negative numbers elegantly handled	Simple, but "end around carry" should be carefully handled
Suitability for serial arithmetic	Not so good	Excellent	Good

ตารางที่ 2.1 แสดงคุณสมบัติที่สำคัญของรูปแบบจำนวนโดยตรง

ใน 3 รูปแบบนี้ตัวเลขแบบส่วนเติมเต็มสองเป็นที่นิยมใช้กันมากในระบบการประมวลผลสัญญาณเชิงเลข ทั้งนี้เนื่องมาจาก

1. มีการแทนค่าเลขศูนย์ได้เพียงค่าเดียว

2. การสร้างวงจรฮาร์ดแวร์สำหรับการบวก ลบ และคูณ ของเลขส่วนเติมเต็มสองทำได้ง่ายโดยในการคูณสามารถใช้หลักการเลื่อนและบวก (Shift and add)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ในระหว่างผลการบวกย่อย (Partial sum) ของการบวกเลขส่วนเติมเต็มสอง สามหรือสี่จำนวน ถึงแม้อาจจะเกิดการล้น (ตัวทดจากผลการบวกล้นข้ามไปทับบิตเครื่องหมาย) 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

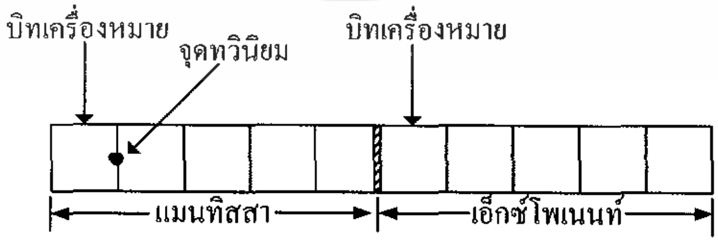
2.9.1.2 รูปแบบจำนวนอิงดรรชนี

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 ถูกตัดให้จำนวนบิตเศษส่วนเหลือเพียง 4 บิต เปอร์เซ็นต์ความผิดพลาดจะเป็น 4.59 % และ 39.6 % ตามลำดับ โดยข้อเสียนี้สามารถแก้ไขได้โดยการใช้รูปแบบจำนวนอิงดรรชนี ซึ่งตัวเลข X แสดงได้โดย

$$X = M \times 2^{e} \tag{2.70}$$

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 2.43 โดยแบ่งเป็น 2 ส่วน คือส่วนหนึ่งสำหรับแมนทิสสา และอีกส่วนสำหรับเอ็กซ์โพเนนท์



รูปที่ 2.43 แสดงการจัดรูปแบบจำนวนอิงดรรชนี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

2.9.2 ทฤษฎีเลขคณิตกระจาย

จากที่ได้กล่าวมาแล้วว่า โครงสร้างเลขคณิตกระจายมีพื้นฐานอยู่บนการคูณแบบเลขส่วนเติมเต็มสอง ดังนั้นในหัวข้อนี้จะได้อธิบายถึงหลักการของการคูณเลขส่วนเติมเต็มสอง

ให้เลขส่วนเติมเต็มสองของ X ซึ่งแทนด้วย $\overline{X}$ และนิยามโดย

$$\overline{x} = \begin{cases} x & \text{ถ้า} X \geq 0 \\ 2 - |x| & \text{ถ้า} X < 0 \end{cases} \tag{2.71}$$

โดย X เป็นเลขที่เป็นเศษส่วน (Fractional number)

ในระบบเลขส่วนเติมเต็มสองจะใช้บิตที่มีนัยสำคัญสูงสุด (MSB) เป็นบิตแสดงเครื่องหมาย โดยถ้าเป็นบวกแทนด้วย “0” และถ้าเป็นลบแทนด้วย “1” ถ้าให้ X แทนด้วยเลขฐานสองขนาด L+1 บิต ดังนั้นรูปแบบของเลขส่วนเติมเต็มสองจะเขียนได้ดังนี้

$$\overline{X} = X_0.X_1.X_2...X_L \tag{2.72}$$

ค่าของ $\overline{X}$ ในรูปของเลขฐานสิบสามารถหาได้ดังนี้

$$X = -X_0 + \sum_{i=1}^{L} X_i 2^{-i} \tag{2.73}$$

จากนั้นพิจารณาผลคูณต่อไปนี้

$$Y = Xm \tag{2.74}$$

ให้ $\overline{Y}$, $\overline{X}$ และ $\overline{m}$ เป็นเลขส่วนเติมเต็มสองของ Y , X และ m ตามลำดับ จากนั้นพิจารณาจากสมการที่ (2.73) และ สมการที่ (2.74) จะได้

$$\begin{aligned} Y &= -Y_0 + \sum_{i=1}^{L} Y_i 2^{-i} \\ &= -X_0 m + \sum_{i=1}^{L} X_i m 2^{-i} \end{aligned} \tag{2.75}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น

$$\overline{Y} = \text{ส่วนเติมเต็มสองของ} \left(-X_0 m + 2^{-1}X_1 m + 2^{-2}X_2 m + 2^{-3}X_3 m + \ldots + 2^{-L}X_L m\right)$$
$$= \text{ส่วนเติมเต็มสองของ} \left(-X_0 m + 2^{-1}(X_1 m + \ldots + 2^{-1}(X_{L-1} m + 2^{-1}(X_L m)))\right) \quad (2.76)$$

ต่อไปพิจารณาส่วนเติมเต็มสองของ $2^{-1}U$ โดย

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

$$\text{ส่วนเติมเต็มสองของ} \quad (2^{-1}U) = \begin{cases} 2^{-1}\overline{U} & ; U_0 = 0 \\ 1 + 2^{-1}\overline{U} & ; U_0 = 1 \end{cases} \quad (2.77)$$

สมการที่ (2.77) นี้แสดงให้เห็นได้ว่า ส่วนเติมเต็มสองของ ($2^{-1}U$) เป็นการเลื่อนข้อมูลของ $\overline{U}$ ไปทางขวา 1 บิต

$$\therefore \text{ส่วนเติมเต็มสองของ} \quad (2^{-1}U) = 2_2^{-1}\overline{U} \quad (2.78)$$

โดย $2_2^{-1}\overline{U}$ แสดงถึงการเลื่อนข้อมูลของ $\overline{U}$ ไปทางขวา 1 บิต แบบเลขส่วนเติมเต็มสอง ซึ่งสัญลักษณ์ 2_2^{-1} (ซึ่งโดยทั่วไปนิยมเขียนเป็น 2^{-1}) เป็นการแสดงว่าในกรณีที่ $\overline{U}$ เป็นเลขบวก ซึ่งบิตเครื่องหมายจะเป็นเลขศูนย์ โดยหลังจากเลื่อนข้อมูลไปทางขวา 1 บิตแล้ว บิตที่มาแทนบิตเครื่องหมายก็ยังคงเป็นเลขศูนย์ ส่วนในกรณีที่ $\overline{U}$ เป็นเลขลบ หลังจากเลื่อนข้อมูลไปทางขวา 1 บิตแล้ว บิตที่มาแทนบิตเครื่องหมายจะเป็นเลขหนึ่ง (จาก $1 + 2^{-1}\overline{U}$) ซึ่งในการสร้างเพื่อใช้งานจริงจำเป็นจะต้องมีวงจรที่ใช้ในการตรวจสอบบิตเครื่องหมาย (Sign digit) ทุกครั้งที่มีการเลื่อนข้อมูล

จากนั้นพิจารณาสมการที่ (2.77) และสมการที่ (2.78) จะได้ว่า

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

$$= -X_0\overline{m} + 2^{-1}(X_1\overline{m} + \ldots + 2^{-1}(X_{L-1}\overline{m} + 2^{-1}(X_L\overline{m}))) \quad (2.79)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจากสมการที่ (2.80) จะเห็นได้ว่าผลคูณจากสมการที่ (2.81) สามารถหาได้โดยการใช้หลักการเลื่อนและบวก (Shift and add) โดยผลลัพธ์ที่ได้จากการคูณแบบเลขส่วนเติมเต็มสอง สามารถหาได้ตามขั้นตอนดังนี้

1. เคลียร์ค่าข้อมูลในแอคคิวมูเลเตอร์รีจิสเตอร์
2. บวก $X_L\overline{m}$ กับค่าที่อยู่ในแอคคิวมูเลเตอร์รีจิสเตอร์
3. เลื่อนค่าที่อยู่ในแอคคิวมูเลเตอร์รีจิสเตอร์ไปทางขวา 1 บิต
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 บิต

$m = -0.390625$

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

$= 2 - 0.390625$

$= 1.609375$

$\therefore \overline{m} = 1.100111$

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

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

โดยมีขั้นตอนการทำงาน ดังตารางต่อไปนี้

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

ตารางที่ 2.2 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง

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

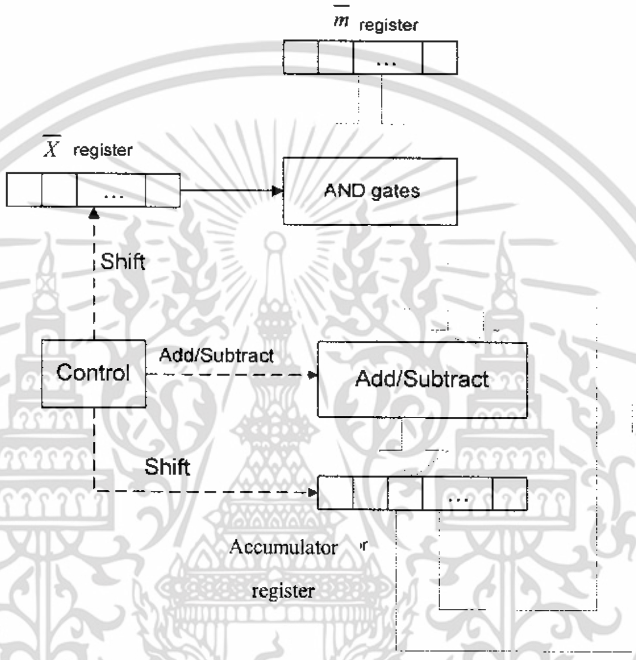
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 2.41



รูปที่ 2.41 แสดงการคูณแบบเลขส่วนเติมเต็มสองโดยใช้เลขคณิตกระจาย

ที่ผ่านมาเป็นหลักการคูณแบบเลขส่วนเติมเต็มสอง ส่วนทฤษฎีเลขคณิตกระจายก็อาศัยหลักการดังกล่าวมาใช้ โดยทำการกระจายสมการที่อยู่ในรูปผลบวกของผลคูณให้แตกออกมาอยู่ในระดับบิต (Bit level) พิจารณาผลบวกของผลคูณต่อไปนี้

$$Y = \sum_{i=0}^{N} m_i X_i \tag{2.80}$$

โดย m_i เป็นค่าสัมประสิทธิ์ซึ่งที่ค่าคงที่

X_i เป็นข้อมูลอินพุต

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X_i = -X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \tag{2.81}$$

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

X_{i0} = บิตแสดงเครื่องหมาย

X_{iL} = บิตที่มีนัยสำคัญต่ำสุด (LSB)

$L+1$ = จำนวนบิตที่แทนข้อมูลอินพุต

แทนค่า X_i ในสมการที่ (2.81) ลงในสมการที่ (2.80) จะได้

$$Y = \sum_{i=0}^{N} m_i \left[-X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \right] \tag{2.82}$$

เมื่อจัดเทอมของผลบวกใหม่จะได้

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

$$Y = -\sum_{i=0}^{N} X_{i0} m_i + \sum_{i=0}^{N} \sum_{j=1}^{L} X_{ij} m_i 2^{-j} \tag{2.83}$$

จากนั้นทำการกระจายออกให้เป็นระดับบิต ได้ดังนี้

$$\begin{aligned} Y = &-(X_{00}m_0 + X_{10}m_1 + X_{20}m_2 + \ldots + X_{N0}m_N) \\ &+ 2^{-1}(X_{01}m_0 + X_{11}m_1 + X_{21}m_2 + \ldots + X_{N1}m_N) \\ &+ 2^{-2}(X_{02}m_0 + X_{12}m_1 + X_{22}m_2 + \ldots + X_{N2}m_N) \\ &+ \ldots + 2^{-L}(X_{0L}m_0 + X_{1L}m_1 + X_{2L}m_2 + \ldots + X_{NL}m_N) \end{aligned} \tag{2.84}$$

สมการที่ (2.84) นี้ถูกกระจายออกให้อยู่ในรูปผลบวกของผลคูณระหว่างสัมประสิทธิ์กับข้อมูลอินพุตในระดับบิต ซึ่งเป็นนิยามของการคำนวณแบบเลขคณิตกระจาย และเมื่อเปรียบเทียบสมการที่ (2.84) กับสมการที่ (2.79) จะเห็นว่าการคำนวณหาค่า Y ก็ใช้เลขคณิตกระจายนั่นเอง เพียงแต่นำค่าผลคูณย่อย (Partial product) ที่คำนวณไว้ล่วงหน้าแล้วสำหรับแต่ละค่าที่สอดคล้องกับแต่ละบิตของข้อมูลอินพุตไปเก็บไว้ในตารางเปิดดู ซึ่งเป็นหน่วยความจำ EPROM และใช้ข้อมูลอินพุตเป็นแอดเดรสของหน่วยความจำ เพื่อนำค่าในตารางเปิดดูมาผ่านขั้นตอนการคำนวณตามบูทอัลกอริธึม ซึ่งค่าในตารางเปิดดูสามารถแสดงได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bit pattern ของข้อมูลอินพุต 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	$m_N + m_{N-1} + \ldots + m_2 + m_1 + m_0$

ตารางที่ 2.3 แสดงค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุต

2.10 วงจรแปลงอะนาลอกเป็นดิจิตอล (Analog to Digital Converter: A/D)

วงจรเปลี่ยนสัญญาณอะนาลอกเป็นดิจิตอล ที่ใช่อยู่ทั่วๆไปมีหลายแบบ คือ

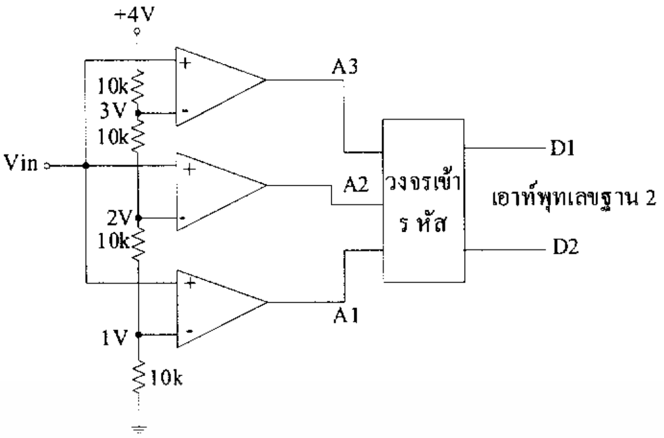
2.10.1 แบบใช้วงจรเปรียบเทียบขนาน หรือแบบแฟลช (Parallel Comparator Simultaneous "Flash" A/D Converter)

วงจรเอทูดีแบบนี้ใช้หลักการง่ายๆ อีกทั้งยังเป็นวิธีที่รวดเร็วที่สุด คือ ใช้วงจรเปรียบเทียบที่ต่อขนานกันดังรูปที่ 2.41 ประกอบด้วยออปแ อมป์ที่ต่อเป็นวงจรเปรียบเทียบ และตัวต้านทานต่อไว้เพื่อแบ่งแรงดันที่ขาอินพุตแบบกลับ (inverting) ให้มีขนาดต่างๆกัน จากหลักการของวงจรเปรียบเทียบทั่วไปเมื่อแรงดันที่ขาอินพุตแบบไม่กลับ(non-inverting) มีค่าสูงกว่าที่ขาอินพุตแบบกลับ เอาท์พุตจะได้แรงดันค่าต่างๆมีผลต่อเอาท์พุตของวงจรเปรียบเทียบแต่ละตัวอย่างไร ซึ่งเอาต์พุตที่ได้จากวงจรเปรียบเทียบนี้จะนำไปเข้ารหัสให้เป็นเลขฐานสองต่อไป

จำนวนของวงจรเปรียบเทียบที่ต้องใช้ในวงจรขึ้นอยู่กับขนาดของสัญญาณอะนาลอกที่อินพุตจากวงจรรูปที่ 2.41 ถ้าแรงดันอินพุตมีค่า 1 โวลท์ ไม่เพียงพอที่จะทำให้วงจรเปรียบเทียบแปลงให้ค่าเอาท์พุตเป็น "High" ที่แรงดันระหว่าง 1 ถึง 2 โวลท์ วงจรเปรียบเทียบที่มีระดับแรงดันเทรชโฮลด์ (threshold) ต่ำสุดให้เอาท์พุตเป็น "High"

แรงดัน 2-3 โวลท์ วงจรเปรียบเทียบทั้ง A1 และ A2 ให้เอาท์พุตเป็น "High" ถ้าแรงดันอินพุตมากกว่า 3 โวลท์ วงจรเปรียบเทียบก็จะเป็น "High" ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.42 แสดงการต่อวงจร Parallel Comparator A/D Converter

เมื่อต้องการวงจรที่มีความละเอียดสูงขึ้นจำเป็นต้องใช้วงจรเปรียบเทียบเพิ่มขึ้น โดยจำนวนวงจรเปรียบเทียบหาได้จาก 2^N-1 เมื่อ N แทนจำนวนบิตหรือความละเอียดที่ต้องการ จะเห็นได้ว่าที่ความละเอียด 8 บิตต้องใช้วงจรเปรียบเทียบมากถึง 255 ตัวซึ่งเป็นข้อเสียของ วงจรดีทูเอแบบนี้

ข้อเสียอีกประการหนึ่งก็คือเอาท์พุตที่ได้ไม่เป็นเลขฐานสองต้องมีวงจรเพิ่มเติมไปทำงานเข้ารหัส "แฟลช" (Flash Type A/D Converter) วงจรเอทูดีชนิดนี้ ใช้เวลาในการแปลงได้เร็วในระดับนาโนวินาทีทีเดียว

2.10.2 วงจรเอทูดีที่ใช้การอินทีเกรต วงจรเปลี่ยนสัญญาณเอทูดีที่ใช้เทคนิคการอินทีเกรตสัญญามี 4 แบบ คือ

- แบบสโลปเดี่ยวหรือแบบแรมป์ (Single Ramp หรือ Single Slope A/D Converter)

เมื่อเริ่มทำการเปลี่ยนสัญญาณ สัญญาณแรมป์และวงจรนับจะถูกรีเซตเป็น "0" แรงดันอะนาลอกถูกป้อนไปยังวงจรเปรียบเทียบทางขาอินพุตแบบไม่กลับ เมื่อแรงดันอินพุตที่ขานี้เป็นบวกมากกว่าที่ขาอินพุตแบบกลับ วงจรเปรียบเทียบก็จะให้เอาท์พุตเป็นระดับ "High" ทำให้แอนด์เกตปล่อยสัญญาณนาฬิกาผ่านไปยังวงจรนับได้ และทำให้เริ่มเกิดสัญญาณแรมป์

สัญญาณแรมป์มีแรงดันเป็นบวกขึ้นเรื่อยๆ จนมากกว่าระดับแรงดันอินพุตและเอาท์พุทของ วงจรเปรียบเทียบก็ตกลงมาเป็นระดับ "Low" ปิดแอนด์เกต ไม่มีสัญญาณผ่านไปให้วงจรนับ

วงจรนับจะหยุดนับและเก็บค่าไว้ที่วงจรแลทซ์ จากนั้นจึงทำการรีเซต วงจรนับและวงจรกำเนิดสัญญาณแรมป์

วงจรแบบนี้เป็นหลักการเบื้องต้นของดิจิตอลโวลต์มิเตอร์ซึ่งถ้าใช้วงจรนับเลขฐานสองแทนแบบ BCD เอาต์พุตก็จะอ่านค่าได้เลขฐานสองโดยตรง

วงจรลักษณะนี้ มักนำไปใช้งานในการเปลี่ยนเวลาเป็นขนาดของสัญญาณ (Time to Amplitude Conversion) หรืออาจใช้ในดิจิตอลโวลท์มิเตอร์ แต่ไม่ใช้กับงานที่ต้องการความถูกต้องสูง เนื่องจากการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนแปลงในแหล่งกำเนิดสัญญาณแรมป์ขึ้นกับอุณหภูมิ และผลตอบสนองต่อสัญญาณอินพุต ทำให้ไม่มีความคงที่ ดังนั้นจึงมีการปรับปรุงให้ดีขึ้นกลายเป็นแบบสโลบคู่ (Dual Slope)

- แบบสโลบคู่ (Dual – Slope A/D Converters)

ส่วนแรก คือวงจรกำเนิดสัญญาณแรมป์หรือวงจรอินทีเกรเตอร์ ที่อินพุตแบบกลับของออปแอมป์มีสภาพเป็นกราวด์เทียม (Virtual ground) ถ้ามีแรงดันอินพุต 2 โวลท์ จะได้กระแสไหลผ่านตัวต้านทาน 10 กิโลโอห์ม เท่ากับ 0.2 มิลลิแอมป์ไปยังจุดร่วม (summing point) เนื่องจากค่าความต้านทานอินพุตของออปแอมป์นั้นสูงมาก กระแสที่ไหลจึงเกิดขึ้นผ่านตัวเก็บประจุ ได้เอาต์พุตออกมาเป็นแรมป์ทางลบ

วงจรเปรียบเทียบก็จะได้แรงดันลบจากวงจรอินทีเกรเตอร์ แล้วให้เอาต์พุตเป็นบวกทำการเปิดสัญญาณแอนด์เกตให้สัญญาณนาฬิกาผ่านเข้าสู่วงจรนับ วงจรนับจะทำการนับไปยังค่าที่กำหนดไว้คงที่ แล้วทำการสับสวิตช์ต่อเข้ากันกับแรงดันอ้างอิง

สรุปได้ว่า แรงดันอินพุต ที่ไม่รู้ค่าถูกป้อนเข้าไปในวงจรอินทีเกรเตอร์เมื่อครบช่วงเวลา t_1 วงจรนับจึงถูกรีเซตให้เป็น 0 อินพุตของวงจรอินทีเกรเตอร์ก็จะถูกสวิตช์ต่อกลับมาที่แรงดันอ้างอิง (ที่มีแรงดันคงที่) ให้ความชันของสัญญาณแรมป์คงที่เพิ่มค่าขึ้นไปจนถึงระดับ 0 ช่วงเวลา t_2 นี้เป็นสัดส่วนโดยตรงกับสัญญาณอินพุต

ข้อดีของวงจรเปลี่ยนสัญญาณแบบสโลบคู่นี้คือ ความถูกต้องสูง , ราคาถูก , เสถียรภาพทางด้านอุณหภูมิดี ข้อเสียคือความเร็วต่ำ ในการเปลี่ยนสัญญาณ 1 ครั้งอาจจะใช้เวลาถึง 100 มิลลิวินาที (ในขณะที่แบบแฟลชใช้เวลาประมาณ 30 มิลลิวินาที)

- แบบชาร์จบาลานซ์ (Charge Balance A/D Converters)

วงจรเปลี่ยนสัญญาณเอทูดีแบบชาร์จบาลานซ์ ใช้วงจรสำคัญคล้ายๆแบบสโลบคู่นั่นเอง แต่แทนที่จะให้อินพุตสวิตซ์ ไปมาระหว่างแรงดันที่ไม่รู้ค่ากับแรงดันอ้างอิง ก็ทำการแทรกพัลส์ของกระแสอ้างอิงโดยตรงที่จุดรวม (Summing point) ของวงจรอินทีเกรเตอร์ในช่วงเวลาที่คงที่ โดยที่จำนวนของพัลส์จะเป็นสัดส่วนโดยตรงกับแรงดันอินพุตที่ไม่รู้ค่า

ประโยชน์ของเทคนิคนี้คือ แรงดันตกคร่อมตัวเก็บประจุของวงจรอินทีเกรเตอร์ จะมีค่าใกล้เคียง 0 โวลท์ ดังนั้นจึงไม่เกิดความผิดพลาดจากผลของกระแสรั่วไหล เอทูดีชนิดนี้จึงมีความถูกต้องสูงกว่าแบบสโลบคู่

- แบบเดลต้า-ซิกม่า (Delta – Sigma A/D Converters)

เมื่อมีแรงดันอินพุตป้อนเข้าไปที่วงจรอินทีเกรเตอร์จะให้เอาต์พุตเข้าไปในวงจรเปรียบเทียบ เปรียบเทียบกับแรงดันคงที่ พัลส์ของกระแสที่ได้จะขึ้นอยู่กับเอาต์พุตของวงจรเปรียบเทียบ โดยสวิตช์ที่ทำขึ้นจากเฟสจะควบคุมกระแสให้เข้าไปยังที่จุดรวมหรือลงกราวด์ไป ส่วนวงจรจะนับวงจรพัลส์ด้วยหลักการที่คล้ายกัน

ข้อสรุปของเอทูดีแบบอินทีเกรตสัญญาณ

จุดสำคัญของอินทีเกรติ้งเทคนิคนี้ คืออินพุตที่ให้กับวงจรอินทีเกรเตอร์ต้องเป็นกระแสไอซีคอนเวอร์เตอร์บางตัวอาจมีอินพุตให้สองขา แต่จะมีขาหนึ่งต่อตรงกับจุด Summing point ใช้กับอุปกรณ์ที่เป็นแหล่งจ่ายกระแสโดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าให้อินพุตเป็นกระแสก็ไม่ต้องคำนึงถึงแรงดันออฟเซตของวงจรอินทีเกรเตอร์ แต่ถ้าหากใช้กับอินพุตที่เป็นแรงดันต้องปรับออฟเซตของออปแอมป์เสียก่อน

การใช้อินพุตเป็นกระแสทำให้ย่านการใช้งานทางไฟสลับกว้าง

2.10.3 วงจรเปลี่ยนสัญญาณเอทูดีที่ใช้วงจรนับและวงจรดีทูเอประกอบกัน

2.10.3.1 แบบวงจรนับเดี่ยว (Single Counter)

แท้ที่จริงแล้วสัญญาณแรมป์เชิงเส้น (Linear ramp)อาจประกอบขึ้นด้วยสัญญาขั้นบันไดเล็กๆ จำนวนมากที่เกิดจากการต่อเอาต์พุทของวงจรนับเข้ากับวงจรแปลงดีทูเอ โดยขนาดของขั้นบันไดแต่ละขั้นขึ้นอยู่กับจำนวนบิตหรือความละเอียดของวงจรดีทูเอนั้นๆ

การกำเนิดสัญญาณแรมป์เดี่ยวด้วยวงจรนับ และวงจรดีทูเอ เมื่อเริ่มแปลงสัญญาณวงจรนับจะถูกรีเซต เอาต์พุทของวงจรดีทูเอ มีระดับ 0 เมื่อแรงดันถูกป้อนเข้าไปยังอินพุตของวงจรเปรียบเทียบ เอาต์พุทก็จะขึ้นสู่ระดับ “High” และเปิดสัญญาณนาฬิกาไปสู่วงจรนับ แต่ละพัลส์ของสัญญาณนาฬิกา ทำให้เกิดการนับและเพิ่มแรงดันขึ้น 1 ขั้น

เมื่อเอาท์พุตของดีทูเอมีค่ามากกว่าอินพุต Vin เอาท์พุตของวงจรเปรียบเทียบก็จะกลายเป็น “Low” ทำให้สัญญาณนาฬิกาไม่อาจผ่านไปยังวงจรนับได้ ดังนั้นวงจรควบคุมจะทำการแลทซ์เอาต์พุตของวงจรนับและรีเซตวงจรนับให้เริ่มต้นรอบใหม่อีกครั้งหนึ่ง

การทำงานจะคล้ายกับแบบใช้วงจรนับเดี่ยว แต่การนับจะไม่ได้เริ่มจากศูนย์ แต่จะทำการนับขึ้นหรือนับลงจากค่าล่าสุดไปยังค่าใหม่ แล้วแต่ว่าแรงดันอินพุตในรอบใหม่มีค่าสูงกว่าหรือต่ำกว่าค่าที่แล้ว ข้อดีของดีทูเอแบบแทร็กกิ้งคือทำงานได้เร็วขึ้น

2.10.4 วงจรเปลี่ยนสัญญาณเอทูดีแบบใช้การประมาณค่า (Successive Approximation A/D Converter)

วงจรเอทูดีแบบนี้มีข้อได้เปรียบทางด้านความละเอียด เพราะความละเอียด n บิต สามารถกำหนดได้จากสัญญาณนาฬิกา n ลูก ตัวอย่างเช่น วงจรแปลงขนาด 8 บิต ต้องการพลัซ์ของสัญญาณนาฬิกาเพียง 8 ลูก ในขณะที่แบบใช้วงจรนับต้องใช้พลัซ์ถึง 256 ลูก วงจรแปลงเอทูดีชนิดนี้มีความเร็วสูง และความละเอียดสูง จึงเป็นวงจรที่นำมาใช้อย่างแพร่หลาย

2.10.5 การสุ่มและการคงค่า (Sample and Holds)

วงจรเอทูดีต้องการเวลาในการนับในการแปลงสัญญาณหรือเรียกว่า Conversion Time เพื่อเปลี่ยนสัญญาณอะนาล็อกเป็นสัญญาณดิจิตอลที่เหมาะสมถ้าสัญญาณอะนาล็อกมีการเปลี่ยนแปลงในระหว่างช่วงเวลาการแปลงเอาท์พุทของวงจรเปลี่ยนสัญญาณอาจเกิดความผิดพลาดจึงต้องมีการป้องกันด้วยวงจรสุ่มและคงค่าสัญญาณ (Sample and Holds) เพื่อใช้จับสัญญาณอะนาล็อกที่จุดเริ่มของการเปลี่ยนสัญญาณแล้วเก็บไว้ในตัวเก็บประจุระหว่างช่วงการแปลง หลังจากที่มีการเปลี่ยนสัญญาณเสร็จสิ้น จึงจับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

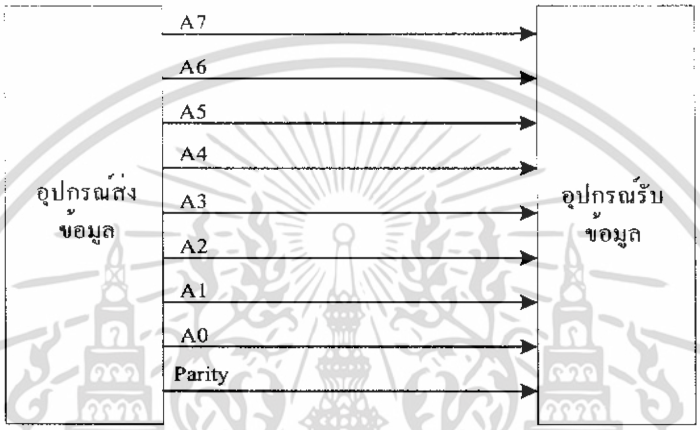
สัญญาณอะนาล็อกค่าใหม่มาเก็บไว้อีกครั้งเป็นเช่นต่อๆไป เราจึงมักได้พบเห็นวงจรสุ่มและคงค่าสัญญาณปรากฏคู่กับวงจรเอทูดีอยู่เสมอ

2.11 พอร์ตอนุกรม

2.11.1 การสื่อสารข้อมูล

2.11.1.1 การสื่อสารข้อมูลแบบขนาน (Parallel Communication)

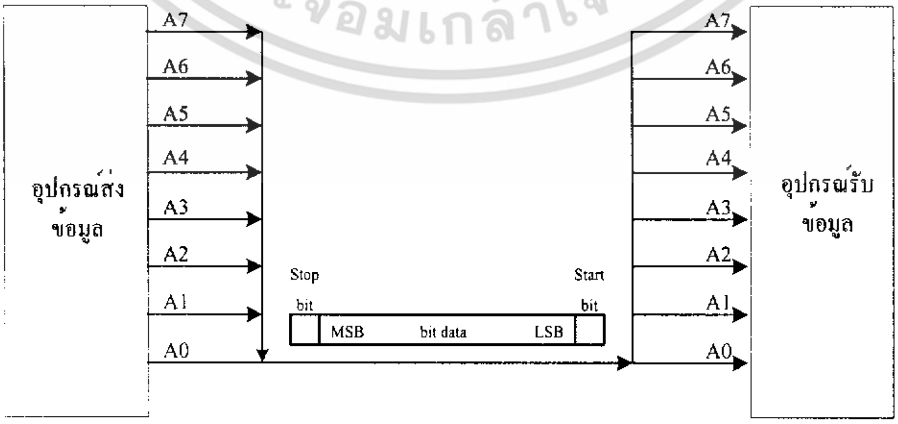
การสื่อสารข้อมูลแบบขนานคือการสื่อสารแบบที่ส่งข้อมูลพร้อมๆกัน n บิตผ่านสายสัญญาณ n เส้น สามารถแสดงรูปแบบการสื่อสารข้อมูลแบบขนานได้ดังรูปที่ 2.43



รูปที่ 2.43 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารข้อมูลแบบขนาน

2.11.1.2 การสื่อสารข้อมูลแบบอนุกรม (Serial Communication)

การสื่อสารข้อมูลแบบอนุกรม คือ การสื่อสารแบบที่ส่งข้อมูลทีละบิต ผ่านสายสัญญาณเส้นเดียวจนครบจำนวนข้อมูลที่ต้องการ โดยเฟรมของการสื่อสารข้อมูลแบบอนุกรมประกอบด้วย สตาร์ทบิต(start bit), สตอบบิต (stop bit), ข้อมูล (data bit) สามารถแสดงรูปแบบการสื่อสารข้อมูลแบบอนุกรมได้ดังรูปที่ 2.44



รูปที่ 2.44 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราบอดเรด (Baud Rate) ในการสื่อสาร คือ ความเร็วในการรับ-ส่งข้อมูลแบบอนุกรมมีหน่วยเป็นบิตต่อวินาที (bit/sec) ซึ่งจะบอกถึงจำนวนบิตที่รับ-ส่ง ในเวลา 1 วินาที เช่น ส่งข้อมูลด้วยอัตรา 9600 บิตต่อวินาที หมายถึง เวลา 1 วินาที รับ-ส่งข้อมูลได้ 9600 บิต รวมทั้งบิตข้อมูล (Data bit) สตาร์ทบิต (Start bit) สตอบบิต (Stop bit) ด้วย

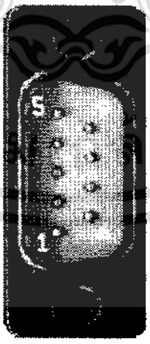
2.11.2 การอินเตอร์เฟสตามมาตรฐาน RS-232

มาตรฐาน RS-232 เป็นมาตรฐานที่ได้รับการพัฒนามานานและถูกใช้งานอย่างแพร่หลาย เราใช้ RS-232 เชื่อมต่อ DTE (Data Terminal Equipment) เช่น คอมพิวเตอร์หรือเทอร์มินัล (Terminal) เป็นต้น เข้ากับ DCE (Data Communication Equipment) เช่น โมเด็ม (Modem) ทีเอ อะเด็บเตอร์ (TA adapter) พล็อตเตอร์ (Plotter) เป็นต้น ตัวอย่างการเชื่อมต่อเช่น การต่อเทอร์มินัลเข้ากับโมเด็ม

มาตรฐาน RS-232 จะใช้สัญญาณเส้นเดียวในการส่งสัญญาณ โดยจะสัญญาณจะส่งไปในทิศทางเดียวกัน สำหรับการแทนค่าแรงดันในการส่งสัญญาณเป็นดังนี้

- สัญญาณของลอจิก “1” แทนด้วยระดับแรงดันไฟฟ้าระหว่าง -3 ถึง -25 โวลต์
- สัญญาณของลอจิก “0” แทนด้วยระดับแรงดันไฟฟ้า ระหว่าง 3 ถึง 25 โวลต์
- ส่วนแรงดันไฟฟ้าระหว่าง 3 ถึง -3 โวลต์ ไม่มีการนิยาม

การเชื่อมต่อกับพอร์ตสื่อสารของคอมพิวเตอร์ส่วนบุคคลจะเลือกใช้พอร์ตสื่อสารแบบอนุกรม 9 ขา (DB-9) ซึ่งสามารถทำการส่งสัญญาณข้อมูลได้ตามมาตรฐาน RS-232 โดยลักษณะของคอนเน็คเตอร์แบบ DB-9 สามารถแสดงได้ดังรูปที่ 2.45 และการเชื่อมต่อของพอร์ตสื่อสารสำหรับคอนเน็คเตอร์แบบ DB-9 สามารถแสดงได้ดังตารางที่ 2.4



รูปที่ 2.45 แสดงลักษณะของคอนเน็กเตอร์แบบ DB-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 การเชื่อมต่อของพอร์ตสื่อสารสำหรับคอนเน็คเตอร์แบบ DB-9

ตำแหน่งขาของ DB-9	**สัญญาณ**
1	Data Carrier Detect : DCD
2	Received Data : RxD
3	Transmitted Data : TxD
4	Data Terminal Ready : DTR
5	Signal Ground : GND
6	Data Set Ready : DSR
7	Request To Send : RST
8	Clear To Send : CTS
9	Ring Indicator : RI

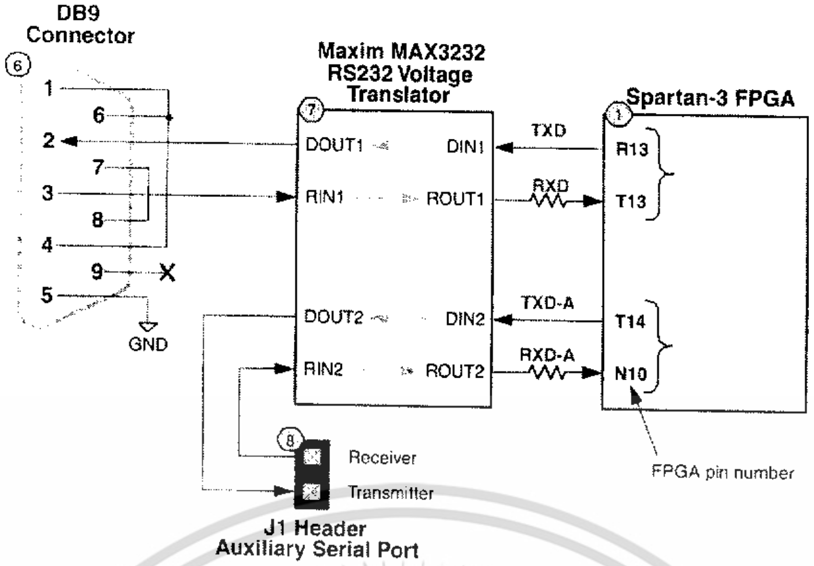
เปรียบเทียบข้อดีข้อเสียของการสื่อสารข้อมูลแบบอนุกรมและขนาน

- การสื่อสารข้อมูลแบบอนุกรมสามารถสื่อสารได้ระยะทางที่ไกลกว่า
- การสื่อสารข้อมูลแบบอนุกรมใช้สายสัญญาณที่ประหยัดกว่า
- การสื่อสารข้อมูลแบบขนานสามารถสื่อสารข้อมูลได้ทีละมากๆ และรวดเร็วกว่า

2.11.3 การเชื่อมต่อระหว่าง DB-9 กับ FPGA

โครงงานนี้ได้ใช้บอร์ด FPGA สปาร์ตัน 3 สตาร์เตอร์คิท (Spartan-3 Starter Kit Board) การเชื่อมต่อระหว่างคอนเน็คเตอร์แบบ DB-9 และ บอร์ด FPGA จะต้องผ่าน MAX3232 ก่อนเพื่อปรับแรงดันที่รับมาได้จากพอร์ต RS232 ให้เหมาะสมกับระดับแรงดันภายในบอร์ด ซึ่งสามารถแสดงการเชื่อมต่อระหว่างคอนเน็คเตอร์แบบ DB-9 และ บอร์ด FPGA โดยผ่าน MAX3232 ได้ดังรูปที่ 2.46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.46 แสดงการเชื่อมต่อระหว่างคอนเน็คเตอร์แบบ DB-9 และ บอร์ด FPGA โดยผ่าน MAX3232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

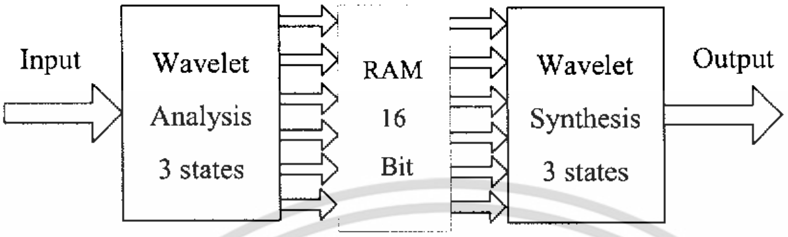
บทที่ 3

การคำนวณและการสร้าง

การแปลงเวฟเลตแบบ 1 มิติ ในบทนี้จะแบ่งออกเป็น 2 ส่วนใหญ่ๆดังนี้

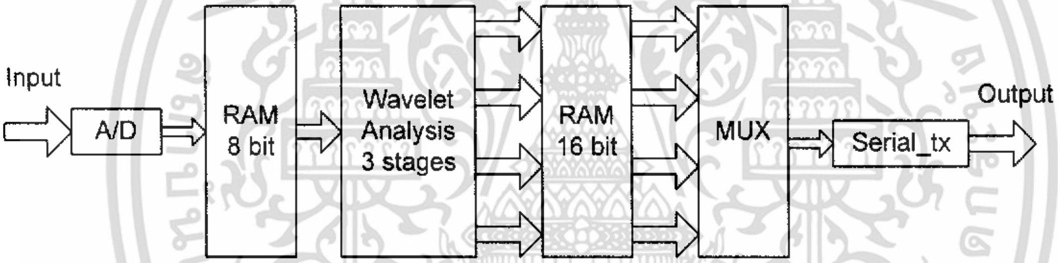
3.1 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL ในการออกแบบ

มีบล็อกไดอะแกรม ดังรูปที่ 3.1



รูปที่ 3.1 บล็อกไดอะแกรมในส่วนของการ Simulate

3.2 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA โดยมีบล็อกไดอะแกรมดังรูปที่ 3.2



รูปที่ 3.2 บล็อกไดอะแกรมในส่วนของการ Implement ลงบนอุปกรณ์ FPGA

การออกแบบ

3.1 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL

3.1.1 สัมประสิทธิ์ที่นำมาใช้ในการแปลงเวฟเลต โดยเลือกใช้สัมประสิทธิ์ของตัวกรองสัญญาณแบบ 4 tap orthonormal filter with two vanishing moments (D4)

n	$H_0(n)$	$H_1(n)$
0	0.48296291314453	0.12940952255126
1	0.83651630373781	0.22414386804201
2	0.22414386804201	-0.83651630373781
3	-0.12940952255126	0.48296291314453

ตารางที่ 3.1 แสดงตารางค่าสัมประสิทธิ์ของดาร์บีชี 4 ค่า ในส่วนของภาคส่ง

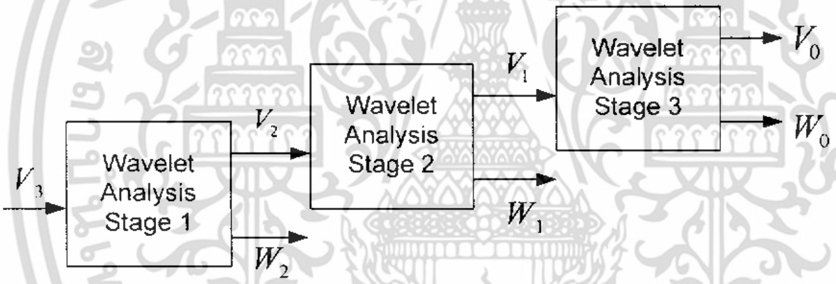
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเพื่อให้ได้ผลตรงตามหลักการการกู้กลับอย่างสมบูรณ์ (Perfect Reconstruct) จะทำให้ได้สัมประสิทธิ์ในการแปลงเวฟเลตทางด้านสังเคราะห์ในภาครับ โดยสามารถหาได้จากสมการที่ 2.44 ดังตารางที่ 3.2

n	$F_0(n)$	$F_1(n)$
0	-0.12940952255126	0.48296291314453
1	0.22414386804201	-0.83651630373781
2	0.83651630373781	0.22414386804201
3	0.48296291314453	0.12940952255126

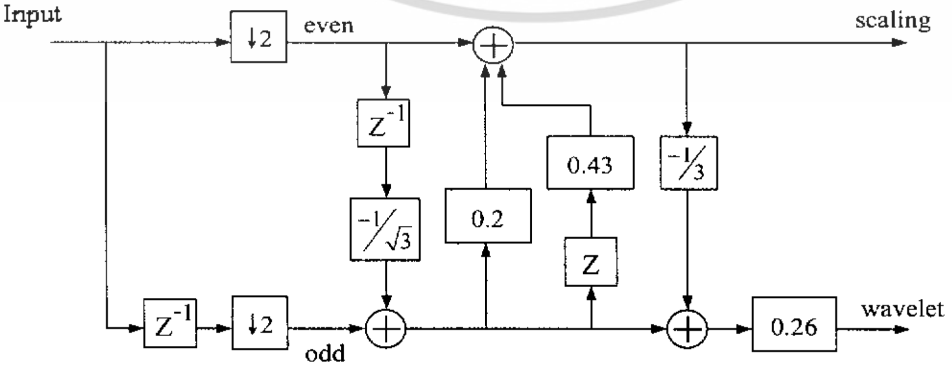
ตารางที่ 3.2 แสดงตารางค่าสัมประสิทธิ์ของดาร์บีชี 4 ค่า ในส่วนของภาครับ

3.1.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ 3 ระดับ จะประกอบด้วยการแปลงเวฟเลตทางด้านวิเคราะห์พื้นฐาน 3 ระดับ ดังรูปที่ 3.3



รูปที่ 3.3 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านวิเคราะห์ 3 ระดับ

โดยที่ในแต่ละระดับจะมีโครงสร้างภายในเป็นโครงสร้างของของการวิเคราะห์แบบ lifting scheme เหมือนกันในทุกๆระดับ ซึ่ง มีส่วนประกอบภายใน ดังรูปที่ 3.4

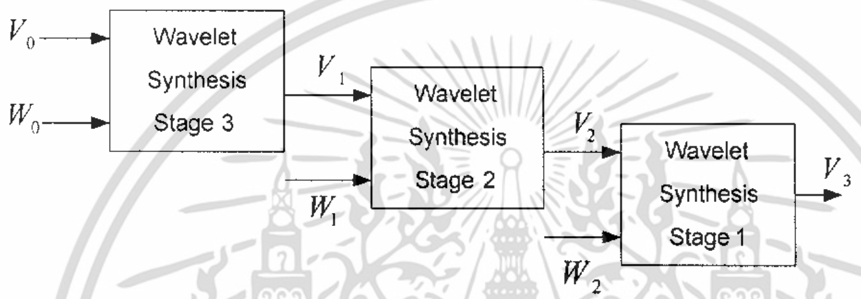


รูปที่ 3.4 โครงสร้างภายในของการแปลงเวฟเลตโดยใช้ lifting scheme ด้านวิเคราะห์ ทางภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

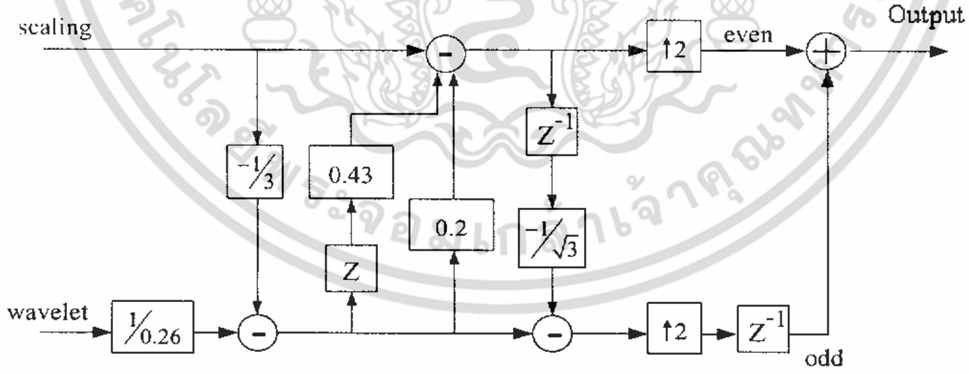
ในการวิเคราะห์เวฟเลตในระดับสอง และ ระดับสาม จะใช้โครงสร้างภายในเหมือนในระดับที่หนึ่ง แต่ต่างกันตรงที่ input ที่เข้ามาใน ระดับที่สาม จะมี timing เป็นสองเท่าของระดับที่สอง และ input ที่เข้ามาในระดับที่สองจะมี timing เป็นสองเท่าของ input ที่เข้ามาในระดับที่หนึ่ง ซึ่งจุดต่างตรงนี้สามารถทำให้ไม่ต้องออกแบบวงจรในระดับที่สองและระดับที่สาม เพียงแต่เพิ่ม timing ของสัญญาณนาฬิกาที่ควบคุม ระดับที่สอง ให้เป็นสองเท่าของระดับที่หนึ่ง และเพิ่ม timing ของสัญญาณนาฬิกาในระดับที่สามให้เป็นสองเท่าของระดับที่สอง โดยวงจรเพิ่ม timing นั้นจะกล่าวต่อไป

3.1.3 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ 3 ระดับ จะประกอบด้วยการแปลงเวฟเลตด้านสังเคราะห์ในขั้นพื้นฐาน 3 ระดับ ดังรูปที่ 3.5



รูปที่ 3.5 บล็อกไดอะแกรมของการแปลงเวฟเลตด้านสังเคราะห์ 3 ระดับ

โดยที่ในแต่ละระดับจะมีโครงสร้างภายในเป็นโครงสร้างของฟิลเตอร์แบบโพลีเฟส เหมือนกันในทุกๆระดับ พิจารณาโครงสร้างในระดับที่ 1 มีส่วนประกอบภายใน ดังรูปที่ 3.6

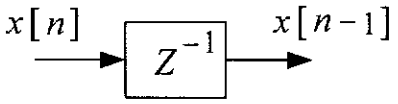


รูปที่ 3.6 โครงสร้างภายในของการแปลงเวฟเลตโดยใช้ lifting schemeทางด้านสังเคราะห์ ทางภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 ส่วนประกอบภายในของการแปลงเวฟเลตโดยใช้ Lifting scheme

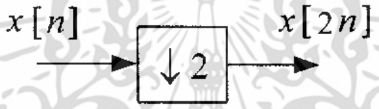
3.1.4.1 วงจรหน่วงเวลา (Delay)



รูปที่ 3.7 บล็อกไดอะแกรมของวงจรหน่วงเวลา

วงจรหน่วงเวลามีหลักการทำงาน คือ จะทำการเลื่อนเวลาของอินพุตที่เข้ามาให้มีเวลาที่ช้ากว่าเดิมอยู่ 1 หน่วย ดังรูปที่ 3.7

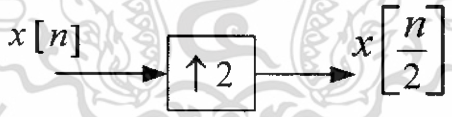
3.1.4.2 วงจรสุ่มค่าตัวอย่าง (Down-Sampling)



รูปที่ 3.8 บล็อกไดอะแกรมของวงจรสุ่มค่าตัวอย่าง

จากรูปที่ 3.8 จะเห็นได้ว่าวงจรนี้จะทำการลดขนาดของอินพุตลงเหลือครึ่งหนึ่งเท่านั้น

3.1.4.3 วงจรเพิ่มค่าตัวอย่าง (Up-Sampling)



รูปที่ 3.9 บล็อกไดอะแกรมของวงจรเพิ่มค่าตัวอย่าง

จากรูปที่ 3.9 จะเห็นได้ว่าเมื่อมีอินพุตเข้ามา 1 ค่า เอาต์พุตจะมีค่าของอินพุตตัวเดิมพร้อมด้วยการแทรกค่าศูนย์เข้าไประหว่างค่าเดิม ทำให้ขนาดของเอาต์พุตที่ได้มีขนาดเป็น 2 เท่าเมื่อเทียบกับขนาดของอินพุต

3.1.4.4 วงจรบวก (Adder)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

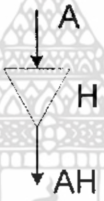
รูปที่ 3.10 บล็อกไดอะแกรมของวงจรบวก

ในการบวกข้อมูล 14 bit เราจะได้ Truth table ดังตารางที่ 3.3

In1	In2	Carry in	out	Carry out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

ตารางที่ 3.3 ตารางค่าความจริงของการออกแบบวงจรบวก

3.1.4.5 วงจรคูณ (Multiplier)



รูปที่ 3.11 บล็อกไดอะแกรมของวงจรคูณ

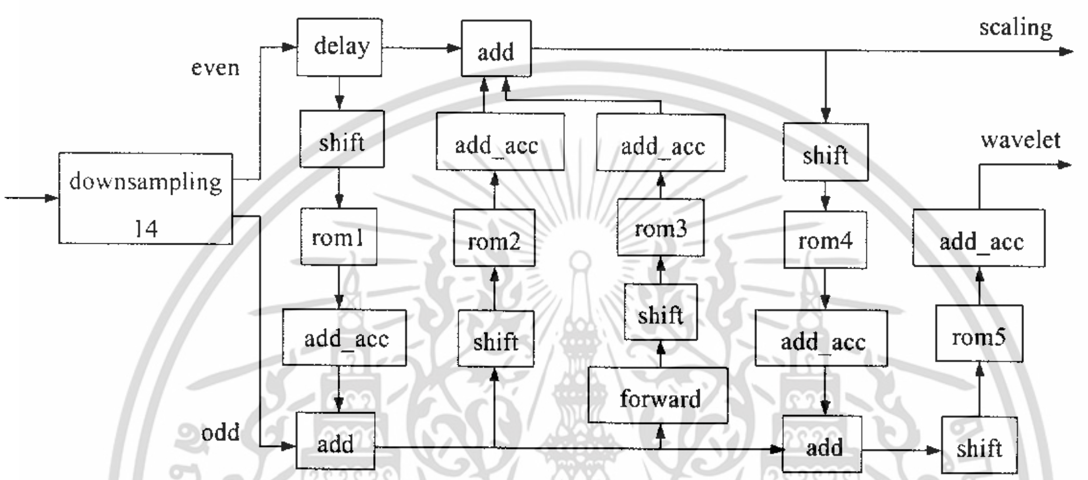
ในส่วนของวงจรคูณนั้น อินพุตที่เข้ามาจะทำการคูณกับค่าคงที่ที่มีอยู่แล้ว ซึ่งจะได้ผลลัพธ์เป็นผลคูณระหว่างอินพุตกับค่าคงที่ ดังรูปที่ 3.11

ในส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ และการแปลงเวฟเลตทางด้านสังเคราะห์ จะพบว่าภายใน 1 ระดับของการแปลงเวฟเลตด้านวิเคราะห์และการแปลงเวฟเลตทางด้านสังเคราะห์ จะใช้วงจรคูณระดับละ 5 วงจรสำหรับการคูณกับสัมประสิทธิ์ 5 ค่า และเนื่องจากเราทำทั้งหมด 3 ระดับ จึงจะต้องใช้วงจรคูณจำนวนทั้งหมด 15 วงจร ในการ Stimulate วงจรคูณจะใช้ทรัพยากรเกตจำนวนมาก ดังนั้นจึงพัฒนามาใช้หลักการเลขคณิตกระจายมาประยุกต์ใช้ ซึ่งหลักการนี้จะใช้จำนวนทรัพยากรเกตน้อยกว่าวงจรคูณในรูปแบบปกติ

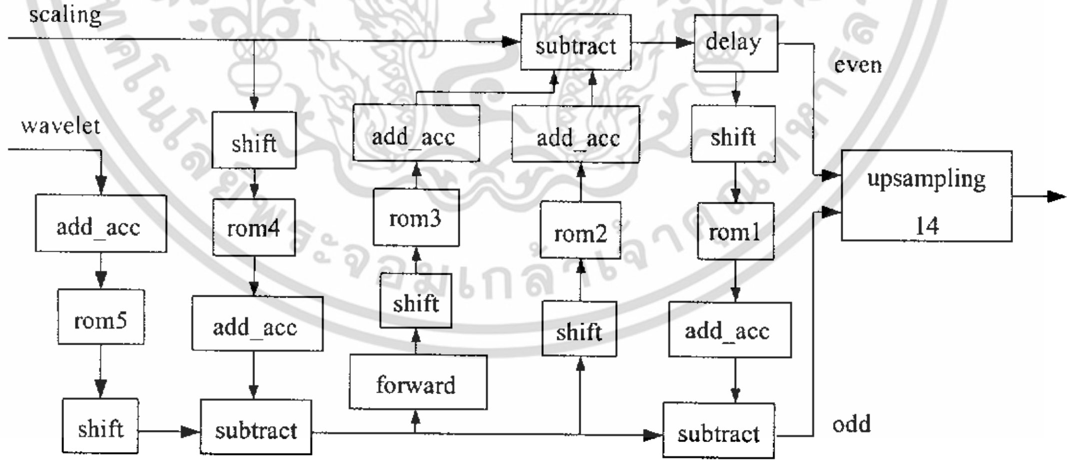
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5 การประยุกต์ใช้งาน DA (Distributed Arithmetic) กับการแปลงเวฟเลต ที่มีโครงสร้างแบบ lifting scheme

การแปลงแบบเวฟเลต โดยใช้โครงสร้างของฟิลเตอร์แบบ lifting scheme มีจำนวนของวงจรคูณเป็นจำนวนมาก ทำให้มีการใช้ทรัพยากรเกตสิ้นเปลือง ดังรูปที่ 3.4 และ 3.6 จึงมีการประยุกต์ใช้งานบนโครงสร้างของ DA ซึ่งคุณสมบัติของ DA นี้จะทำงานอยู่ในรูปแบบของผลบวกของผลคูณ (Sum of Products) โดยไม่ใช้การคูณโดยตรงในการประมวลผลของข้อมูล ซึ่งแนวความคิดนี้เหมาะสมกับรูปแบบการคำนวณของการแปลงเวฟเลต โดยสามารถเปลี่ยนให้อยู่ในรูปโครงสร้าง ดังรูปที่ 3.12



รูปที่ 3.12 การแปลงเวฟเลตด้านวิเคราะห์โดยนำหลักการของ DA

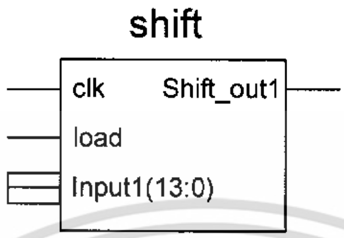


รูปที่ 3.13 การแปลงเวฟเลตด้านสังเคราะห์โดยนำหลักการของ DA มาใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของการแปลงเวฟเลตด้านวิเคราะห์ และการแปลงเวฟเลตทางด้านสังเคราะห์โดยใช้หลักการ DA ดังรูปที่ 3.12 และ 3.13 จะประกอบด้วยวงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม, หน่วยความจำ, ส่วนเก็บค่าผลการคำนวณ และวงจรบวกหรือวงจรลบ ดังต่อไปนี้

3.1.5.1 วงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม (Parallel in Serial out (PISO))



รูปที่ 3.14 บล็อกไดอะแกรมของวงจร Parallel in Serial out Shift Register

การทำงานของวงจร เมื่อมีสัญญาณอินพุท (Input) เข้ามา และสัญญาณที่ขา load มีค่า '1' จะทำการ load ข้อมูลเข้ามาเลขเก็บในชิพรีจีสเตอร์ (Shift Register) เมื่อขา load มีค่า '0' เมื่อป้อนสัญญาณนาฬิกา (clk) ก็จะทำให้รีจีสเตอร์ (Register) ทำการเลื่อน ข้อมูลออกเอาท์พุท (Shift_out1) ทีละบิต เริ่มจากบิตที่มีนัยสำคัญต่ำสุด (LSB) ถึงข้อมูลที่สูงสุด (MSB) ตามสัญญาณนาฬิกา (Clock)

3.1.5.2 หน่วยความจำ (ROM)

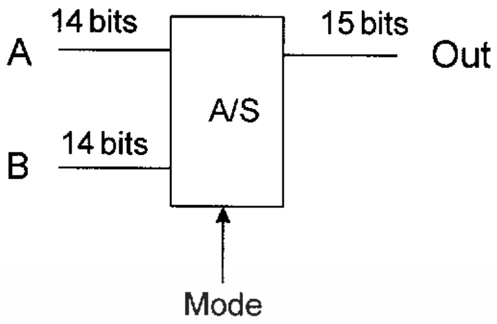
ในการแปลงเวฟเลตนั้น จะเก็บค่าที่ใช้ในการคูณไว้ในROM ซึ่งมีค่า 14 บิต ในการแปลงเวฟเลตแบบ lifting scheme นั้น จะมีการคูณ 5 ครั้ง ในการวิเคราะห์แต่ละ state ดังนั้นในการออกแบบวงจรจะออกแบบ Rom 5 ตัว เพื่อเก็บค่าในการคูณทั้ง 5 ครั้ง เพื่อทำการคูณแบบ DA ดังที่ได้กล่าวมาแล้ว ซึ่งสามารถออกแบบหน่วยความจำ ROM ได้ดังตารางที่ 3.4

Input Bit	"0"	"1"
Rom1	0	10110110000110
Rom2	0	00011001101110
Rom3	0	00110111011011
Rom4	0	11010101010101
Rom5	0	00100010010011

ตารางที่ 3.4 ตารางแสดงผลลัพธ์ของวงจร ROM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5.3 วงจรบวกหรือวงจรลบ (Adder/Subtractor)

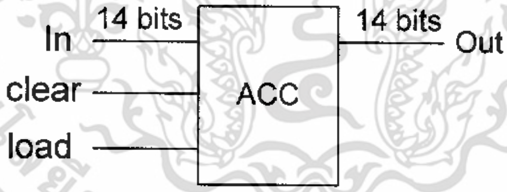


รูปที่ 3.15 บล็อกไดอะแกรมของวงจรบวกหรือวงจรลบ

ในส่วนของวงจรบวกหรือลบ (Adder & Subtractor) คือ วงจรบวกและวงจรลบที่จะรับค่าที่อ่านจาก ROM เข้ามาทำการบวกหรือลบกับค่าใน Accumulator ซึ่งสามารถเลือกการบวกหรือลบได้จากขา mode ผลลัพธ์ (output) จะมี 15 บิต บิตที่มีนัยสำคัญสูงสุด MSB หรือบิตที่ 15 จะเป็นบิตเช็คเครื่องหมายเพื่อนำไปเลื่อนแทนที่ในการเลื่อนขวาของข้อมูลในวงจรเก็บค่า (Accumulator)

3.1.5.4 ส่วนเก็บค่าผลการคำนวณ (Accumulator)

มีหน้าที่ ทำการ Load ค่า output ที่ได้จากการบวกหรือลบ (A/S) เข้ามาเก็บแล้วทำการเลื่อนขวา 1 บิต แล้วส่งกลับไปบวกหรือลบกับค่าต่อไปในส่วนของ A/S ตามรูป Block diagram

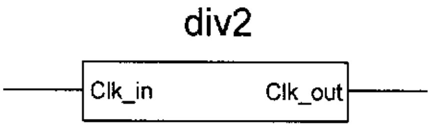


รูปที่ 3.16 บล็อกไดอะแกรมของ Accumulator

3.1.6 ส่วนของการออกแบบวงจรหารความถี่สัญญาณนาฬิกา

ดังที่ได้กล่าวมาในหัวข้อ 3.1.2 ว่า ในการวิเคราะห์เวฟเลต โดยใช้โครงสร้างแบบ lifting scheme ในแต่ระดับนั้นสามารถใช้วงจรเดียวกันได้ แต่ในการวิเคราะห์ระดับสองนั้นต้องใช้สัญญาณนาฬิกาที่มีความถี่น้อยกว่าในระดับที่หนึ่ง สองเท่า และในการวิเคราะห์ระดับที่สามนั้นต้องใช้สัญญาณนาฬิกาที่มีความถี่น้อยกว่าในระดับที่สอง สองเท่า จึงต้องทำการออกแบบวงจรหารสัญญาณนาฬิกาลงสองเท่า ซึ่งมีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

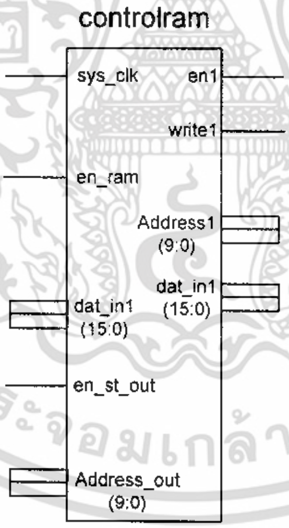


รูปที่ 3.17 แสดงสัญลักษณ์วงจรหารความถี่สองเท่า
วงจรนี้จะรับสัญญาณนาฬิกาเข้ามาแล้วเพิ่มค่าเวลาของคาบให้เป็นสองเท่าแล้วส่งออกไป

3.1.7ส่วนของการเก็บสัมประสิทธิ์ scaling และ wavelet ที่ได้จากการวิเคราะห์ในแต่ละระดับ

เมื่อข้อมูล input ผ่านการวิเคราะห์ในแต่ละระดับ จะมีการเก็บสัมประสิทธ์ wavelet ระดับที่หนึ่ง , สัมประสิทธิ์ wavelet ระดับที่สอง , สัมประสิทธิ์ scaling ระดับที่สาม และ สัมประสิทธิ์ wavelet ระดับที่สาม ลงในหน่วยความจำRAM เพื่อเตรียมส่งออกไปยังพอร์ตอนุกรมอย่างเป็นลำดับ ซึ่งส่วนนี้ประกอบด้วย วงจรที่ใช้ควบคุม RAM เพื่อให้สามารถเก็บข้อมูลเข้า RAM ได้อย่างถูกต้อง และ วงจร RAM เพื่อใช้ในการเก็บสัมประสิทธิ์ scaling และ wavelet ที่ได้จากการวิเคราะห์ในแต่ละระดับ

3.1.7.1 วงจรควบคุมการทำงานของ RAM



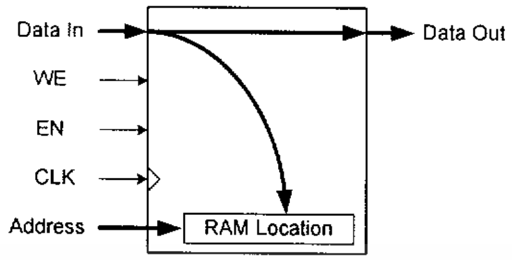
รูปที่ 3.18 สัญลักษณ์ของวงจรควบคุมการทำงานของ RAM

มีหลักการทำงานคือ เมื่อมีสัญญาณ en_ram เข้ามา มันจะส่งสัญญาณออกไป enable ตัวของ RAM โดยในการออกแบบนั้นต้องควบคุมเวลา ของการนำไปเก็บให้ตรงกับ timing ของสัญญาณ input นอกจากนั้นยังมีขา en_st_out และ ขา address_out เพื่อกระตุ้นให้ RAM ปล่อยข้อมูลที่เก็บออกมาเพื่อเตรียม multiplex กับสัมประสิทธิ์ส่วนอื่นๆ และส่งออกทางพอร์ตอนุกรมต่อไป

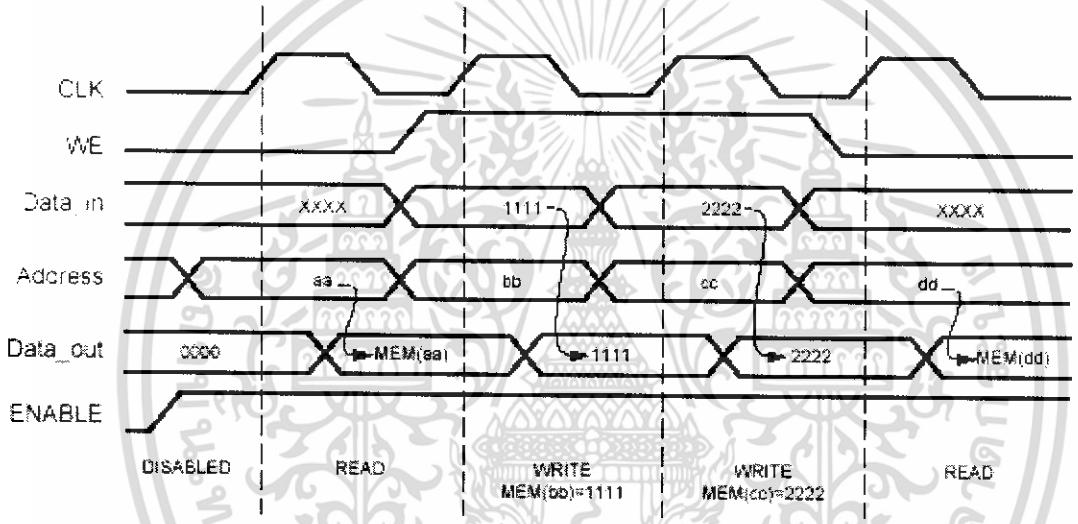
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.7.2 Random Access Memory (RAM)

ในส่วนนี้จะเลือกใช้ RAM ที่มี write mode เป็นแบบ write first mode โดยจะมีบล็อกไดอะแกรม ดังรูปที่ 3.19



รูปที่ 3.19 แสดงการไหลของข้อมูลในระหว่างกระบวนการ Write



รูปที่ 3.20 แสดงเวฟฟอร์มของ Write first mode

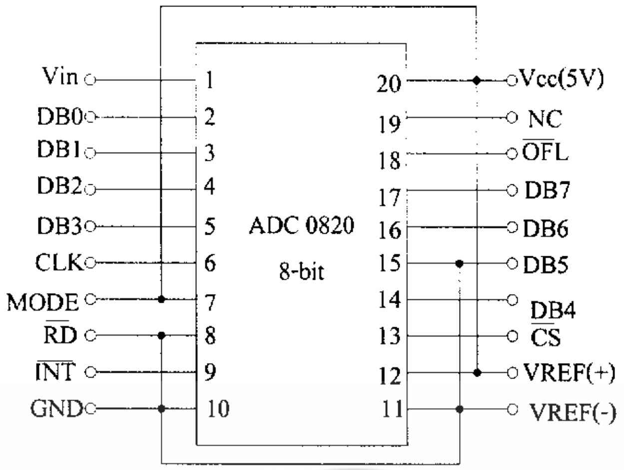
จากรูปที่ 3.20 สามารถอธิบายกระบวนการ Read และ write data ได้คือ เมื่อ ENABLE มีค่าเป็น 1 RAM จะเริ่มต้นการทำงานพร้อมขอบขาขึ้นของ CLK โดยจะทำงาน read หรือ write จะขึ้นกับค่าของ WE ถ้า WE มีค่าเป็น 0 จะทำหน้าที่ read ข้อมูลจาก Address ของ RAM ส่งออกไปยัง Data Out และเมื่อ WE มีค่าเป็น 1 จะหน้าที่ write ข้อมูล Data In ลงใน Address ของ RAM พร้อมกับส่งข้อมูลออกสู่ Data Out

3.2 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA

3.2.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter)

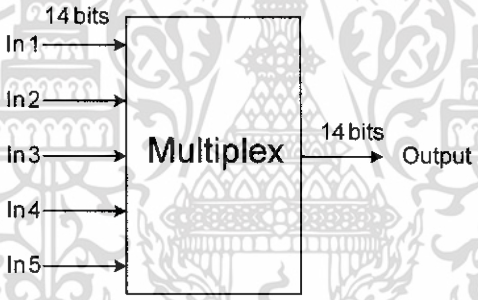
จะใช้ไอซีเบอร์ ADC 0820 ซึ่งจะทำการแปลงระดับสัญญาณขนาด 8 บิต ระดับสัญญาณสามารถแปลงได้ทั้งสัญญาณบวก (สูงสุด +5 โวลท์) และสัญญาณลบ (ต่ำสุด -5 โวลท์) ดังวงจรข้างล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter)

3.2.2 วงจรมัลติเพล็กซ์ (Multiplex) เป็นการมัลติเพล็กซ์สัมประสิทธิ์ที่ได้จากการแปลงเวฟเลต 3 ขั้นและอินพุทที่ได้จากการแซมปลิ้งออกไป

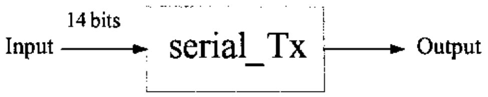


รูปที่ 3.22 บล็อกไดอะแกรมของวงจรมัลติเพล็กซ์

3.2.3 ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGAไปยังคอมพิวเตอร์

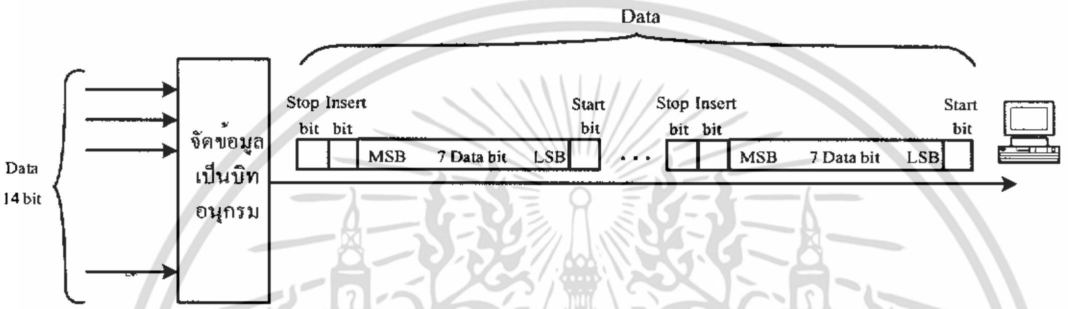
ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ ทำหน้าที่แปลงบิตข้อมูลแบบขนานขนาด 14 บิต ที่ได้จากแปลงเวฟเลตทรานสฟอร์ม ให้เป็นบิตข้อมูลแบบอนุกรม โดยแบ่งเป็น 2 เฟรม โดยแต่ละเฟรมข้อมูลประกอบด้วย สตาร์ทบิต(Start bit) 1 บิต , สตอบบิต(Stop bit) 1 บิต , บิตข้อมูล(Data bit) 7 บิต และแทรกบิต(Insert bit) 1 บิต เพื่อแก้ไขกรณีการส่งบิตบิตข้อมูลที่เป็น 0 ทั้งหมด หลังจากนั้นส่งเฟรมข้อมูลดังกล่าวออกพอร์ตอนุกรม(serial port)ไปยังคอมพิวเตอร์ตามความถี่บอดเรด(BaudRate)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.23 บล็อกไดอะแกรมของวงจรส่งข้อมูลแบบอนุกรม

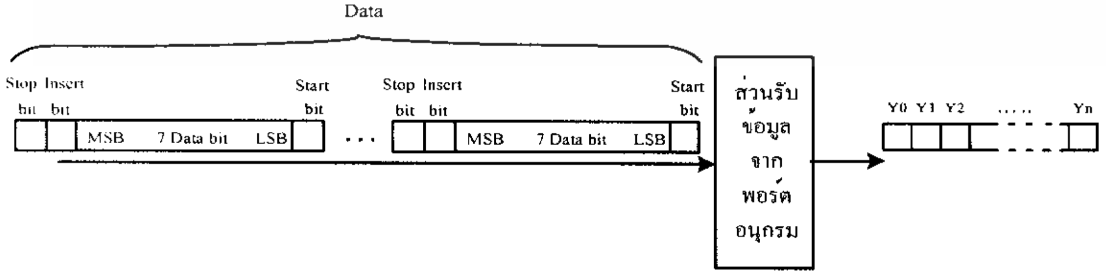
สามารถแสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ ดังรูปที่ 3.24



รูปที่ 3.24 แสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรม
จากบอร์ด FPGA ไปยังคอมพิวเตอร์

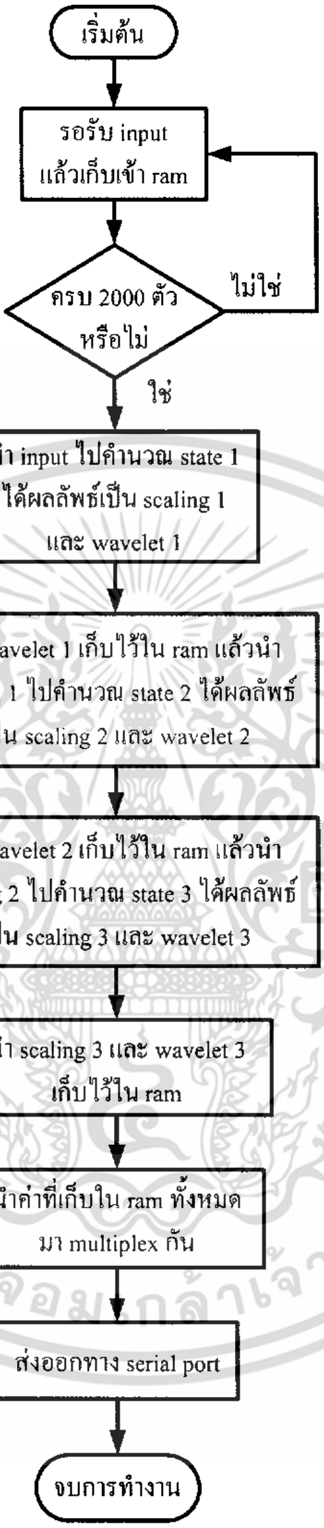
3.2.4 ส่วนของการรับบิตข้อมูลจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ผ่านทางพอร์ตอนุกรม

ส่วนของการรับข้อมูลแบบอนุกรมจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ทำหน้าที่รับเฟรมบิตของข้อมมูลจากพอร์ตอนุกรมตามความถี่บอดเรด (Baud Rate) มาทำการตัดสตาร์ทบิต (Start bit) และสตอบบิต (Stop bit) และบิตแทรก (Insert bit) ทิ้ง พร้อมทั้งรวมเฟรมบิตข้อมูลที่แยกกันอยู่เพื่อให้ได้ค่าของข้อมูล 1 ค่า แล้วนำค่าที่ได้เก็บให้ครบทุกค่า สามารถแสดงบล็อกไดอะแกรมการรับบิตข้อมูลผ่านทางพอร์ตอนุกรมได้ดังรูปที่ 3.25



รูปที่ 3.25 แสดงบล็อกไดอะแกรมการรับบิตข้อมูลจากบอร์ด FPGA ผ่านพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.26 โฟลว์ชาร์ตแสดงการทำงานของการแปลงเวฟเลต ในส่วนของการ Simulate โดยใช้โปรแกรม Xilinx และแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

การออกแบบส่วนต่างๆ ของการแปลงเวฟเลต โดยใช้โครงสร้างของฟิลเตอร์แบบโพลีเฟส (Polyphase Structure) สามารถแบ่งออกเป็น 2 ส่วน คือ

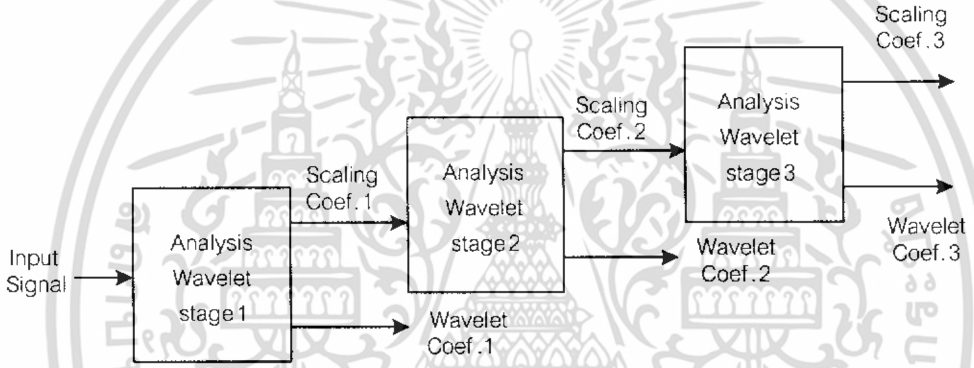
4.1 ส่วนของการ Simulate โดยใช้โปรแกรม MATLAB

4.2 ส่วนของการ Simulate โดยใช้โปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL

4.1 ส่วนของการ Simulate โดยใช้โปรแกรม MATLAB จะแบ่งออกเป็น 2 ส่วน คือ

4.1.1 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ (Analysis Wavelet Transform)

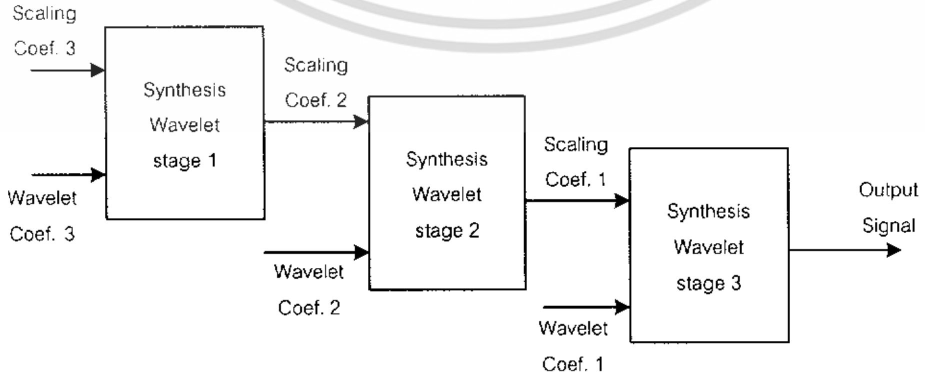
โดยจะทำการแปลงเวฟเลตทั้งหมด 3 ขั้น ดังรูปที่ 4.1



รูปที่ 4.1 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านวิเคราะห์

4.1.2 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ (Synthesis Wavelet Transform)

ในส่วนนี้จะประกอบด้วยการแปลงเวฟเลตกลับทั้งหมด 3 ขั้นตอน เช่นเดียวกับทางด้านวิเคราะห์ ดังรูปที่ 4.2



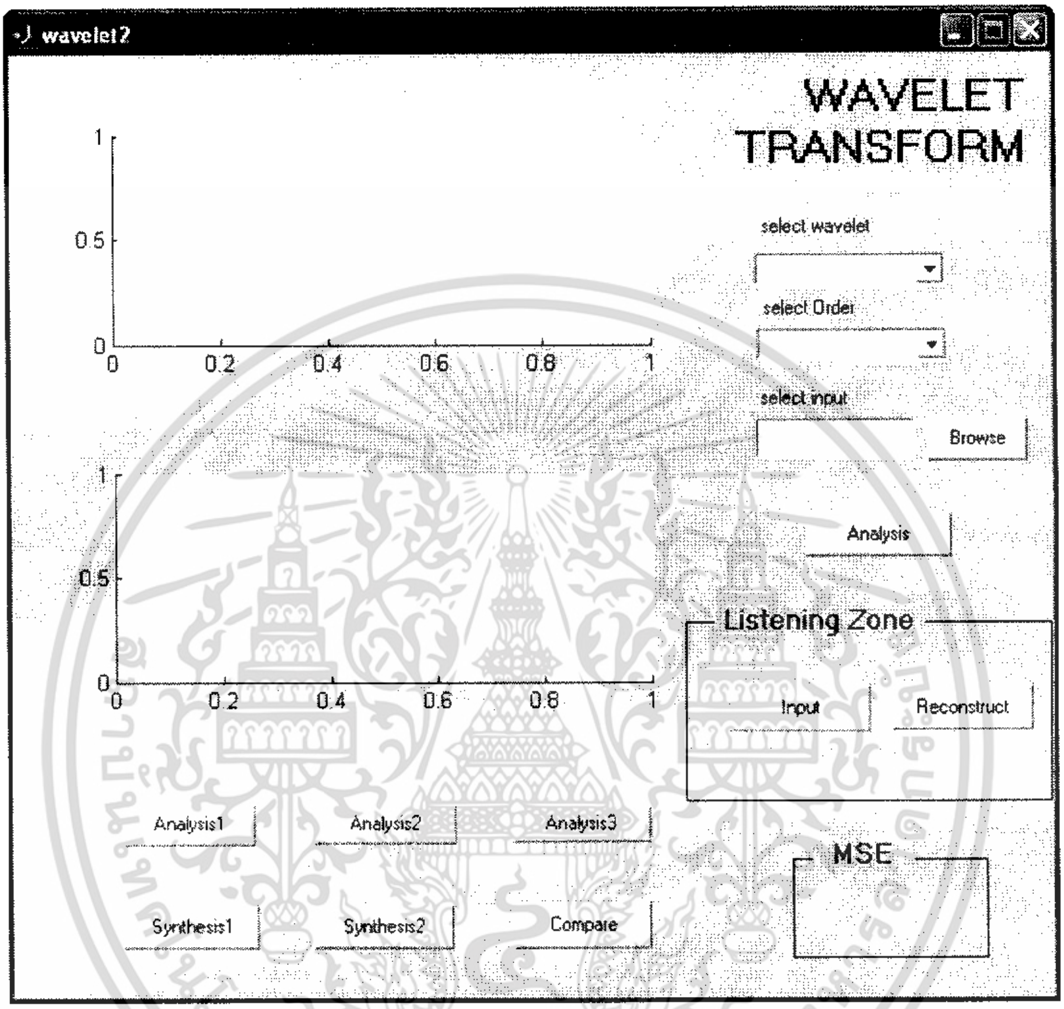
รูปที่ 4.2 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านสังเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 ผลการทดลอง

ทดลองการแปลงเวฟเลตโดยใช้โปรแกรม MATLAB

4.1.3.1 ส่วนของ interface



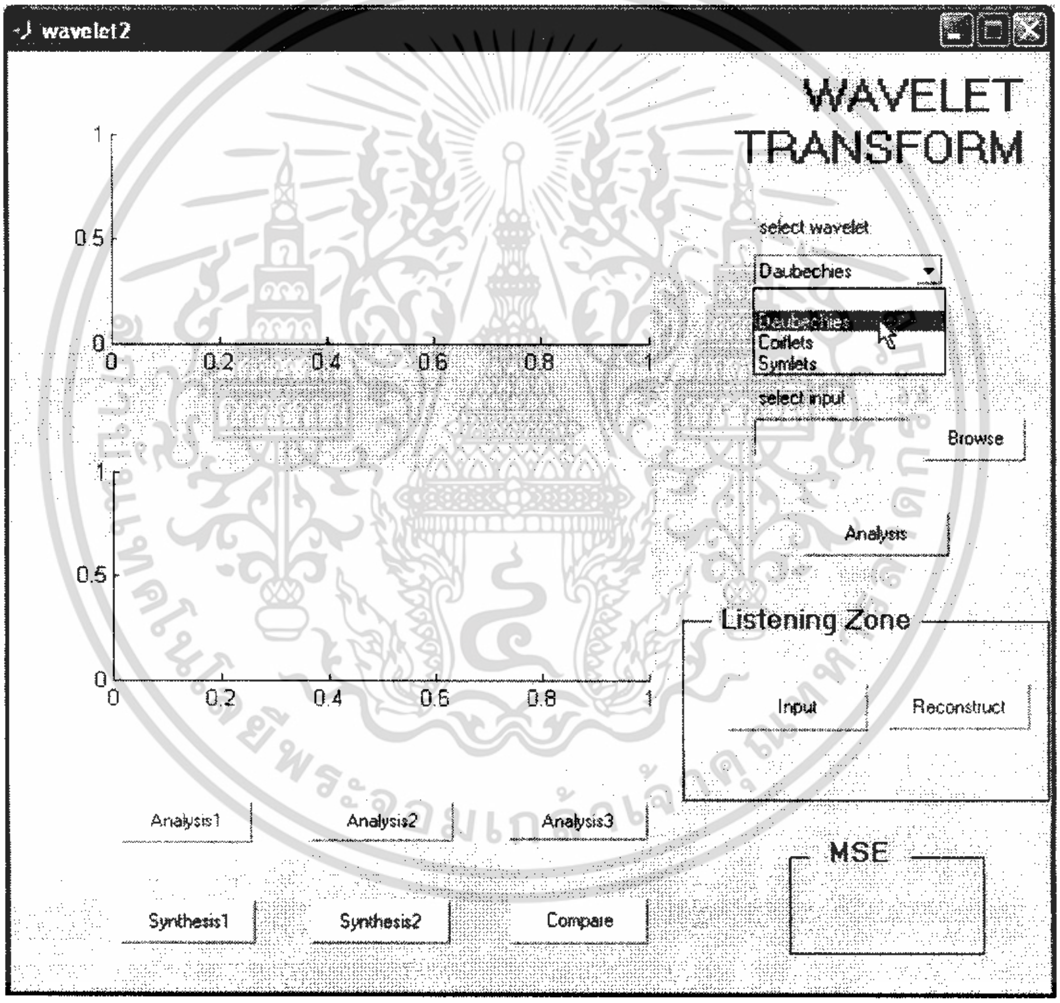
รูปที่ 4.3 หน้าต่างของแมทแลบ ส่วน interface ของโปรแกรมWavelet Transform

จากรูป 4.3 แสดง Interface ของโปรแกรมคำนวณและแสดงผลการแปลงเวฟเลต โดยตัวโปรแกรมประกอบด้วยส่วนแสดงผลกราฟ 2 กราฟ ซึ่งใช้แสดงผลการคำนวณที่ได้ในการแปลงเวฟเลต โดยใช้ Filter Banks ในแต่ละขั้น ส่วนของการเลือกชนิดของเวฟเลต และ อันดับ และส่วนของการเลือกสัญญาณที่จะใช้ในการแปลงเวฟเลต นอกจากนี้ยังมีส่วนที่ใช้แสดงค่า MSE เพื่อแสดงค่าความผิดพลาดระหว่างสัญญาณอินพุต กับสัญญาณที่กู้กลับคืนมาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงานของโปรแกรม

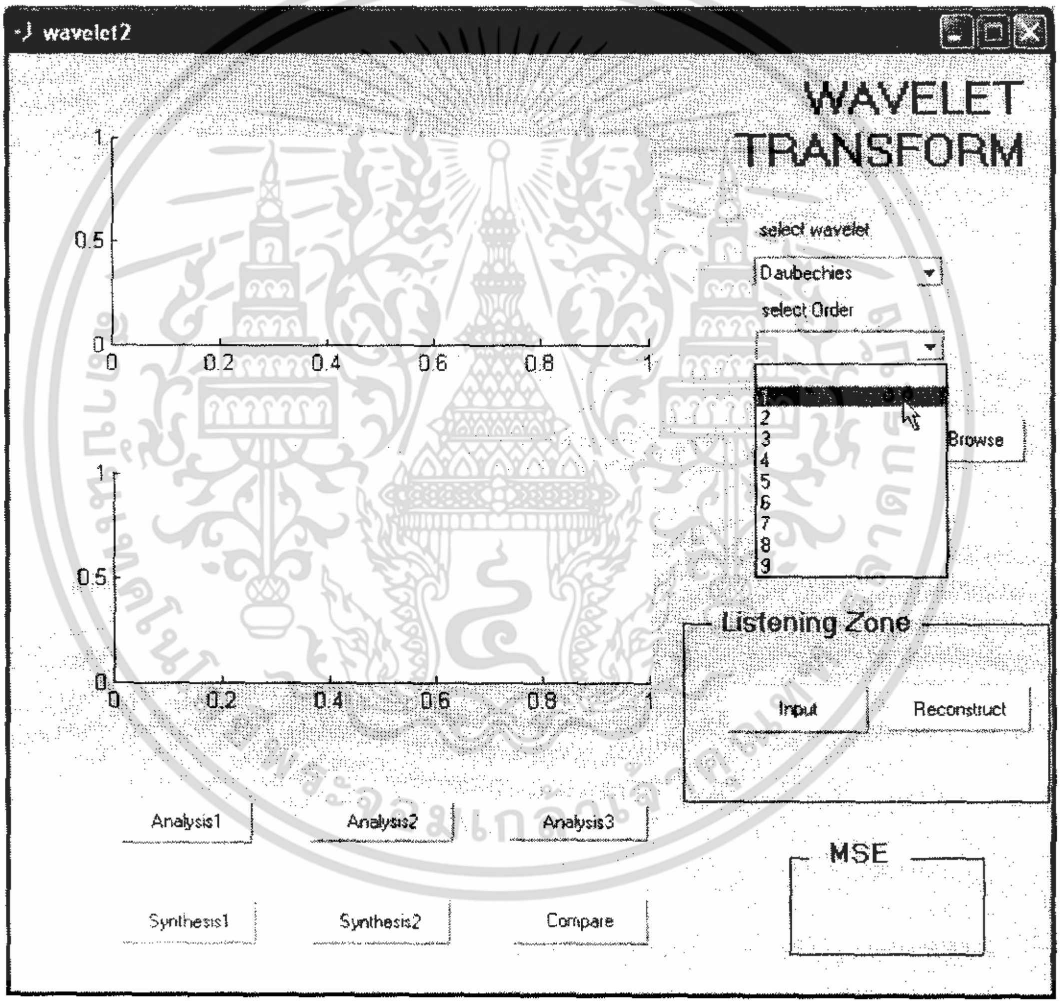
1. ทำการเลือกชนิดของ mother wavelet ที่จะใช้ในการแปลงเวฟเลต ดังรูป 4.4 โดยจะมีให้เลือก 3 ชนิด คือ Daubechies , Coiflets และ Symlets
2. เลือก Order ของ mother wavelet โดยมีให้เลือกตั้งแต่ 1- 9 ดังรูปที่ 4.5
3. เลือกสัญญาณอินพุตที่จะนำมาแปลงเวฟเลต โดยการกดปุ่ม Browse แล้วจะขึ้นหน้าต่างชื่อ File Selector ขึ้นมา จากนั้น เลือกไฟล์สัญญาณอินพุต ซึ่งสามารถรับสัญญาณได้สองแบบ คือสัญญาณ 1 มิติที่สร้างจากโปรแกรมแมทแล็บ (ไฟล์นามสกุล . mat) และสัญญาณเสียง (ไฟล์นามสกุล .wav) ดังรูปที่ 4.6



รูปที่ 4.4 แสดงการเลือกชนิดของ mother wavelet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

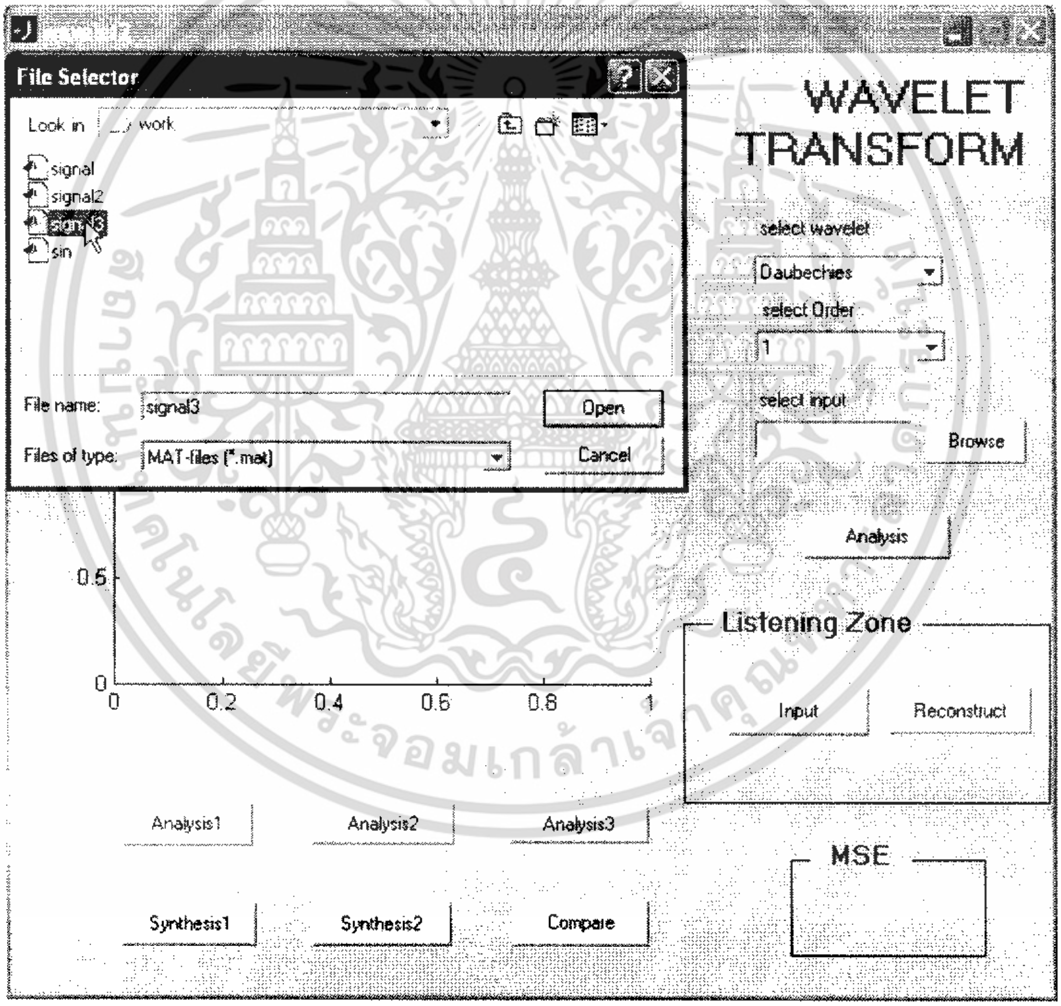
4. จากนั้นกดปุ่ม Analysis เพื่อทำการวิเคราะห์และคำนวณการแปลงเวฟเลต ทั้งด้าน Analysis และ Synthesis โดยใช้ Filter Bank ซึ่งจะใช้สัมประสิทธิ์ของตัวกรองสัญญาณตามชนิดและ order ของ mother wavelet ที่เลือกไว้ เมื่อประมวลผลเสร็จจะแสดงผลเป็นกราฟเปรียบเทียบกันระหว่าง สัญญาณอินพุตและสัญญาณที่ถูกกู้กลับคืนมาได้ดังรูปที่ 4.7
5. กดปุ่ม Analysis1 เพื่อแสดงผลระหว่าง Scaling Function1 ที่ออกมาจาก ตัวกรองความถี่ต่ำผ่านในการ analysis ขั้นที่1 และ Wavelet Function1 ที่ออกมาจาก ตัวกรองความถี่สูงผ่านในการ analysis ขั้นที่1 ดังรูปที่ 4.8 ซึ่งตัดมาจากตัวโปรแกรม



รูปที่ 4.5 แสดงการเลือก order ของ mother wavelet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

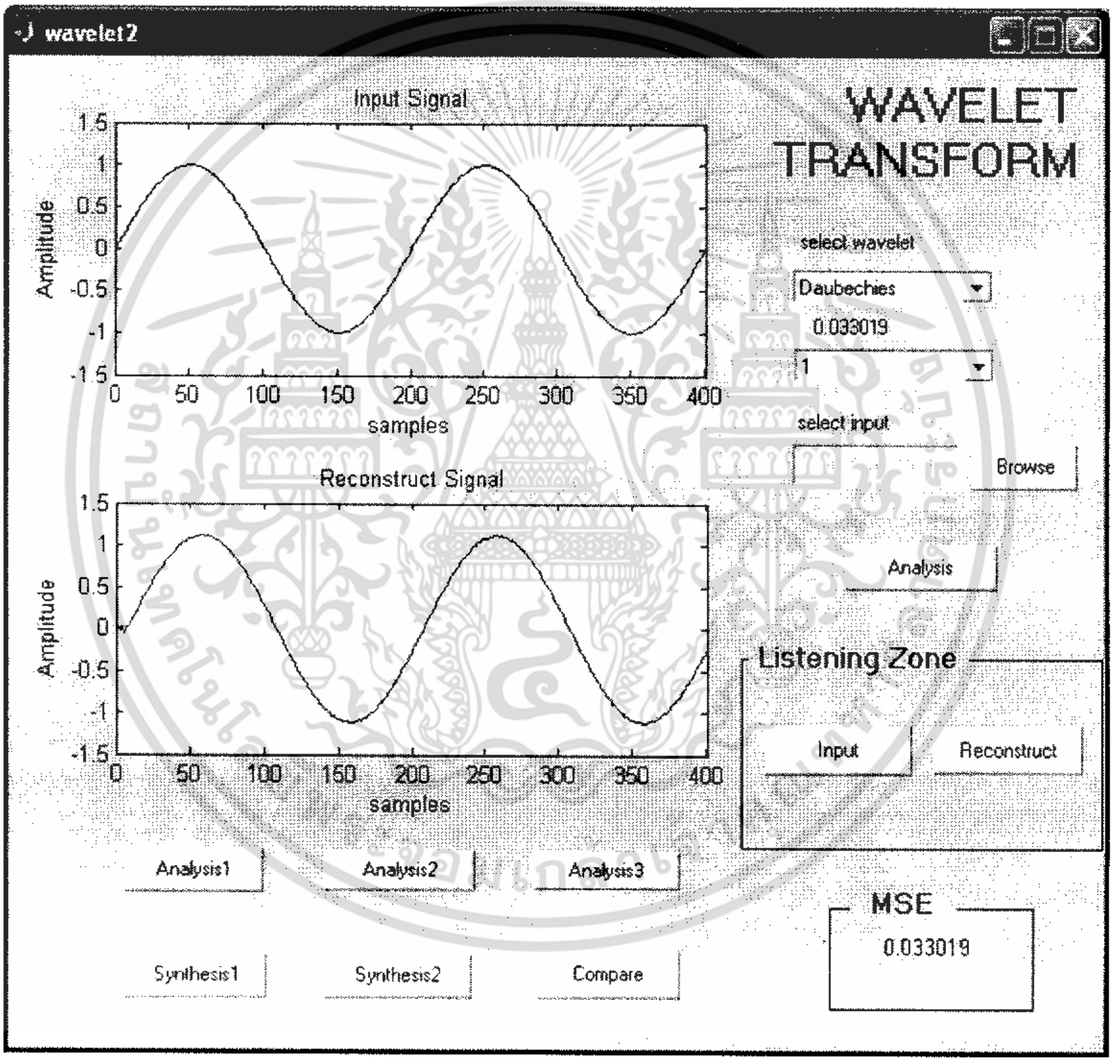
6. กดปุ่ม Analysis2 เพื่อแสดงผลระหว่าง Scaling Function2 ที่ออกมาจาก ตัวกรองความถี่ต่ำผ่านในการ analysis ขั้นที่2 และ Wavelet Function2 ที่ออกมาจาก ตัวกรองความถี่สูงผ่านในการ analysis ขั้นที่2 ดังรูปที่ 4.9 ซึ่งตัดมาจากตัวโปรแกรม
7. กดปุ่ม Analysis3 เพื่อแสดงผลระหว่าง Scaling Function3 ที่ออกมาจาก ตัวกรองความถี่ต่ำผ่านในการ analysis ขั้นที่3 และ Wavelet Function2 ที่ออกมาจาก ตัวกรองความถี่สูงผ่านในการ analysis ขั้นที่3 ดังรูปที่ 4.10 ซึ่งตัดมาจากตัวโปรแกรม



รูปที่ 4.6 แสดงการเลือกสัญญาณอินพุตที่ใช้ในการแปลงเวฟเลต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

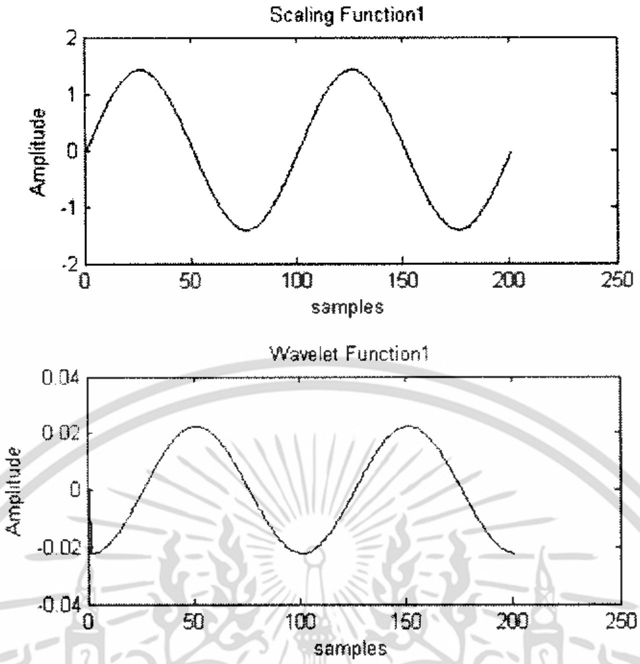
8. กดปุ่ม Synthesis1 เพื่อแสดงผลระหว่าง สัญญาณอินพุต กับสัญญาณที่กู้กลับมาได้เมื่อผ่านการ Synthesis ขั้นที่1 ดังรูปที่ 4.11 ซึ่งตัดมาจากตัวโปรแกรม
9. กดปุ่ม Synthesis2 เพื่อแสดงผลระหว่าง สัญญาณอินพุต กับสัญญาณที่กู้กลับมาได้เมื่อผ่านการ Synthesis ขั้นที่2 ดังรูปที่ 4.12 ซึ่งตัดมาจากตัวโปรแกรม
10. กดปุ่ม Synthesis3 เพื่อแสดงผลระหว่าง สัญญาณอินพุต กับสัญญาณที่กู้กลับมาได้เมื่อผ่านการ Synthesis ขั้นที่3 ดังรูปที่ 4.13



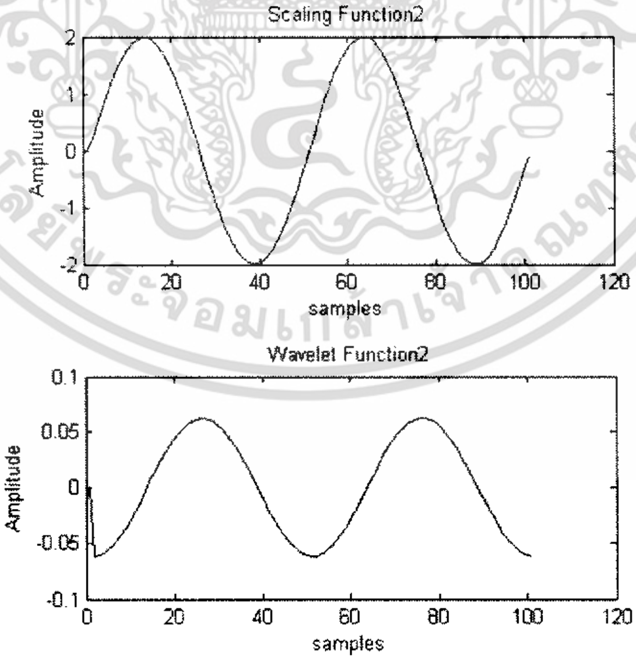
รูปที่ 4.7 แสดงการเปรียบเทียบระหว่างสัญญาณอินพุตกับสัญญาณที่กู้กลับ เมื่อกดปุ่ม Analysis แล้วกดปุ่ม compare ก็จะแสดงค่า MSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ เมื่อสัญญาณอินพุตเป็นข้อมูล

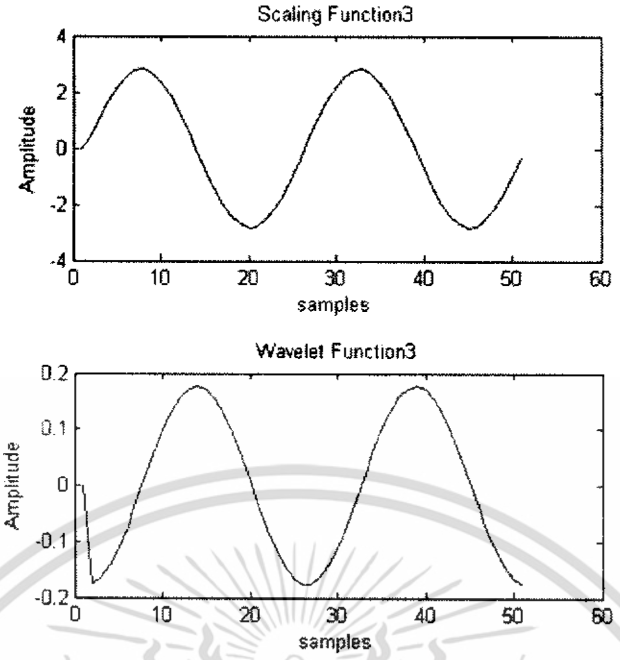


รูปที่ 4.8 แสดงผลในด้านวิเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ scaling function1 และ ส่วนของ wavelet function1



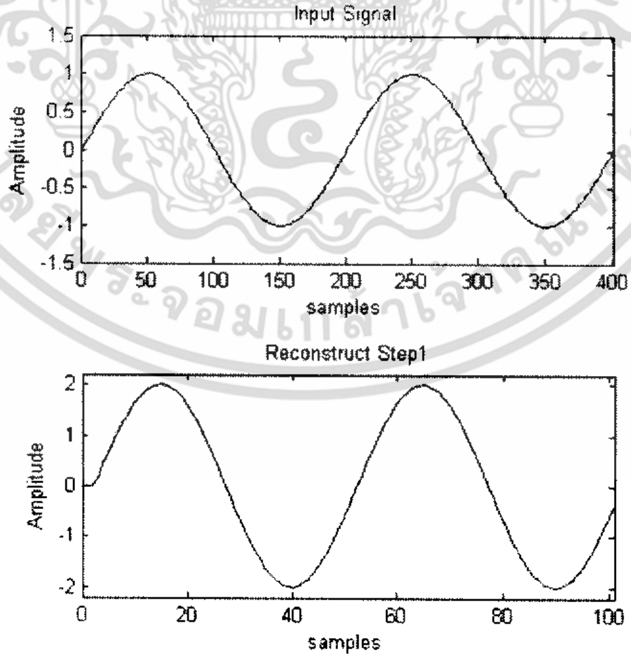
รูปที่ 4.9 แสดงผลในด้านวิเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ scaling function2 และ ส่วนของ wavelet function2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



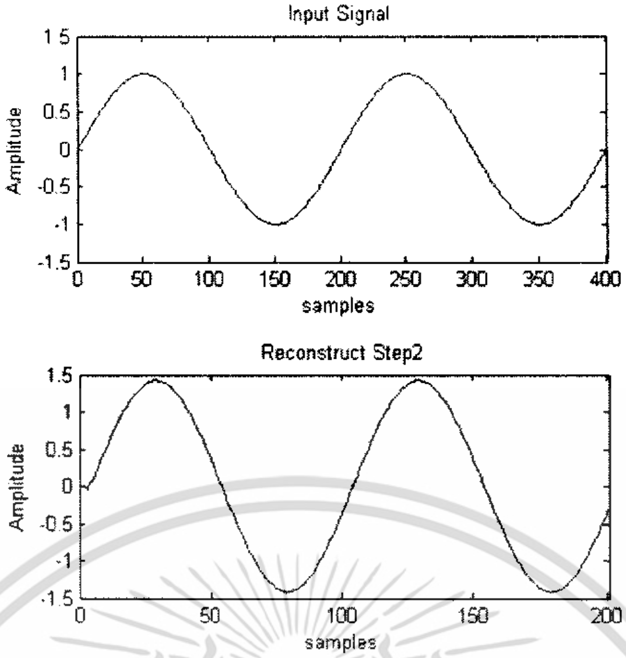
รูปที่ 4.10 แสดงผลในด้านวิเคราะห์ขั้นที่ 3 โดยจะมีส่วนของ scaling function3 และส่วนของ wavelet function3

4.1.3.3 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์เมื่ออินพุตเป็นข้อมูล

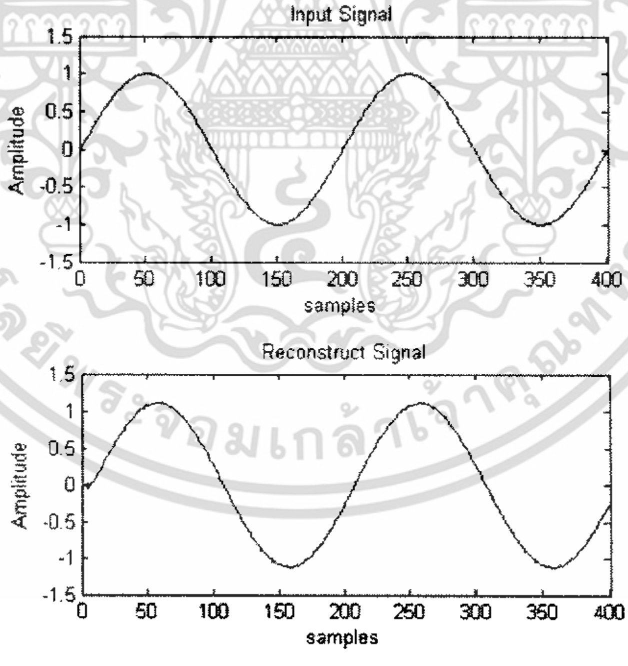


รูปที่ 4.11 แสดงผลในด้านสังเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



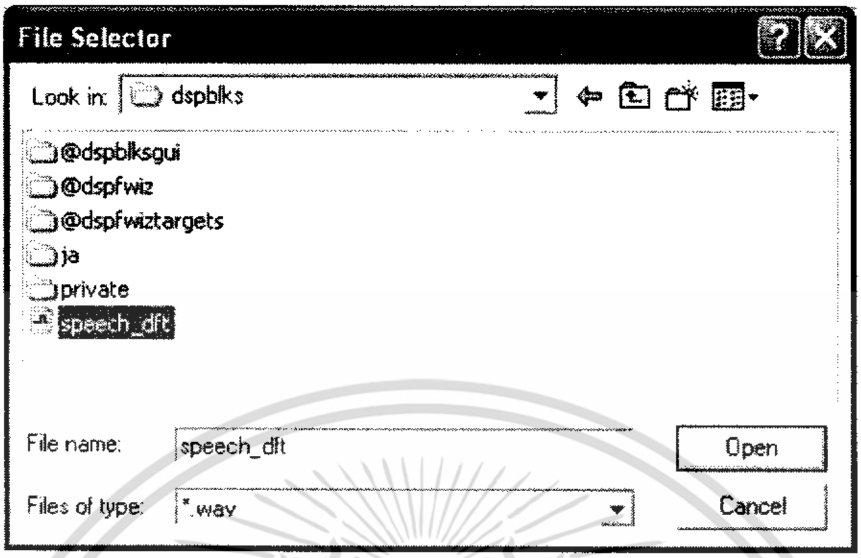
รูปที่ 4.12 แสดงผลในด้านสังเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step2



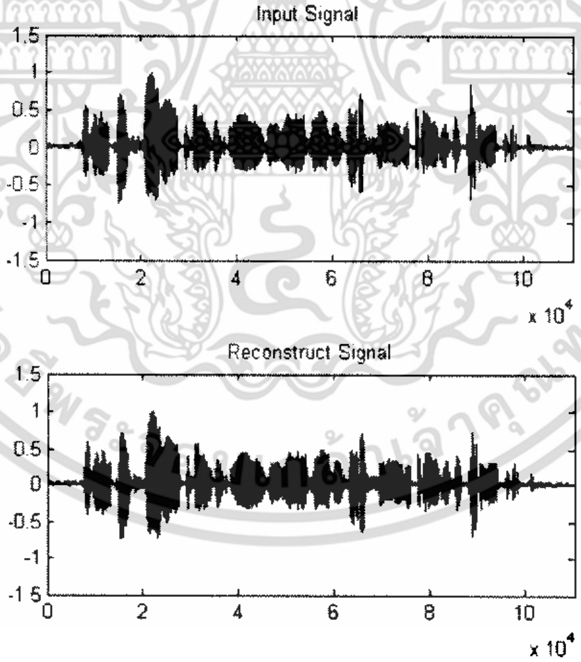
รูปที่ 4.13 แสดงผลในด้านสังเคราะห์ขั้นที่ 3 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 ผลการทดลองเมื่อใส่สัญญาณอินพุตเป็นสัญญาณเสียง

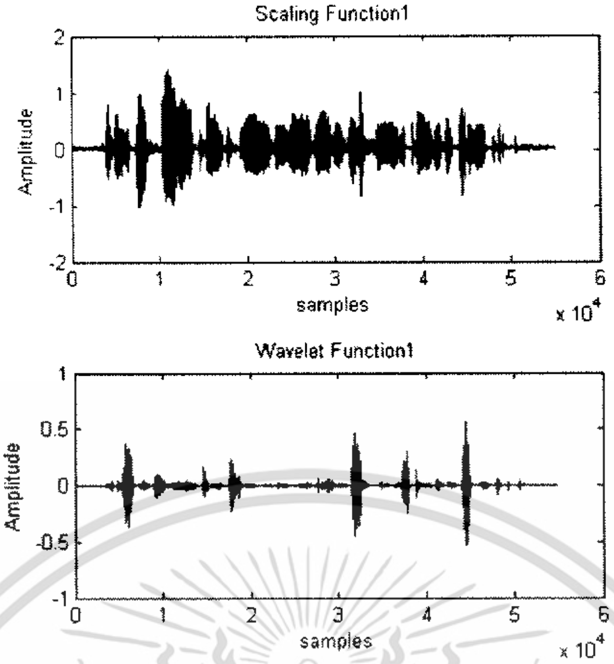


รูปที่ 4.14 เมื่อใส่สัญญาณอินพุตเป็นสัญญาณเสียง

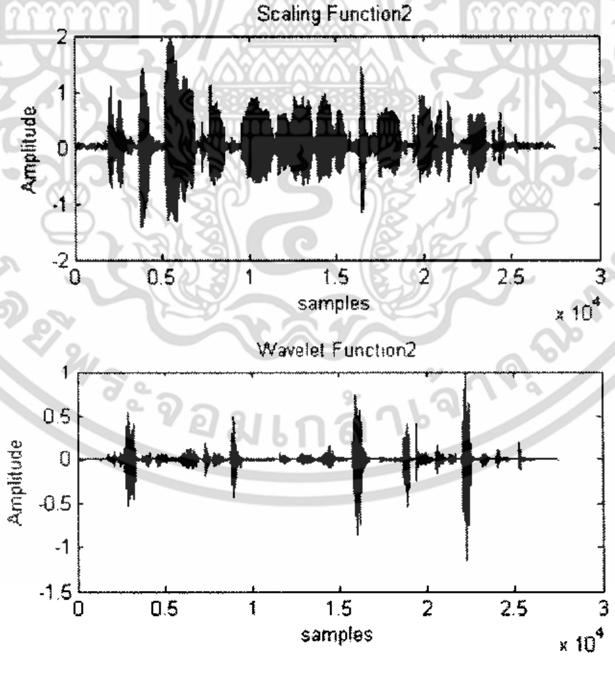


รูปที่ 4.15 แสดงผลของการเปรียบเทียบระหว่างสัญญาณอินพุตกับสัญญาณที่กู้กลับ เมื่อกดปุ่มAnalysis

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

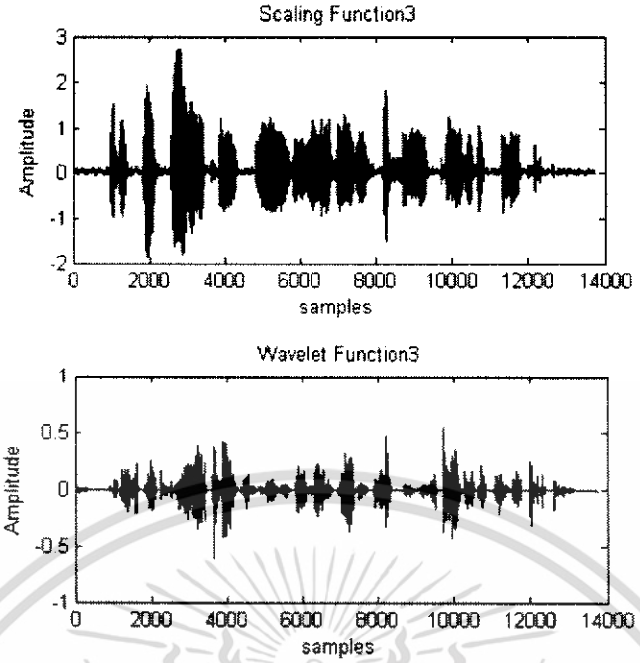


รูปที่ 4.16 แสดงผลในด้านวิเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ scaling function1 และส่วนของ wavelet function1

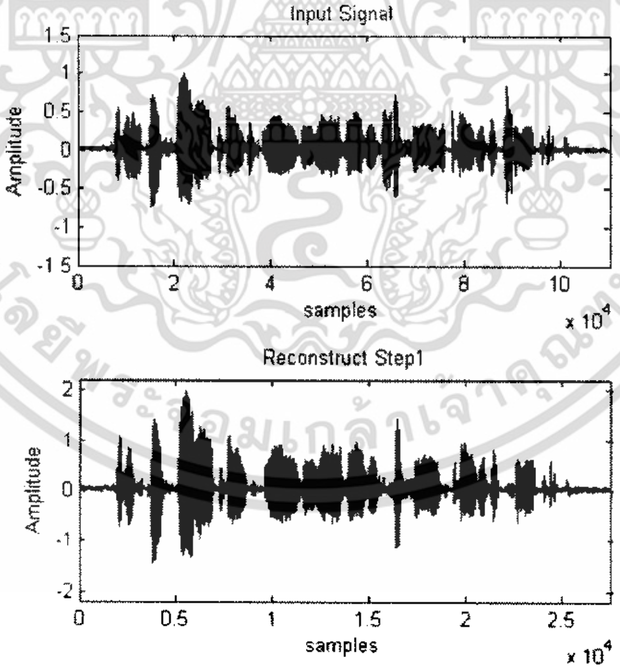


รูปที่ 4.17 แสดงผลในด้านวิเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ scaling function2 **และส่วนของ** wavelet function2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

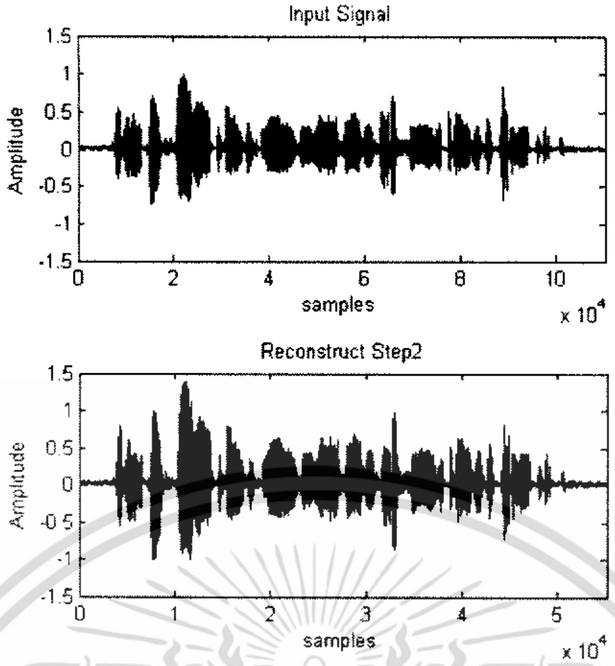


รูปที่ 4.18 แสดงผลในด้านวิเคราะห์ขั้นที่ 3 โดยจะมีส่วนของ scaling function3 และส่วนของ wavelet function3

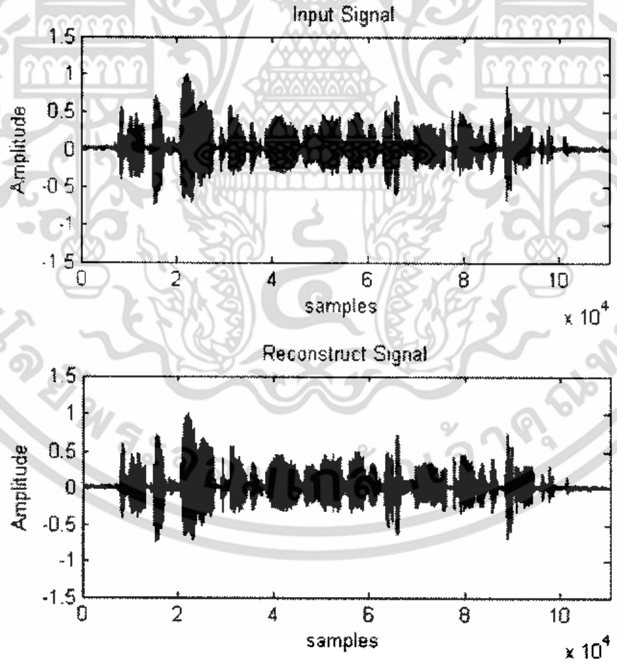


รูปที่ 4.19 แสดงผลในด้านสังเคราะห์ขั้นที่ 1 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

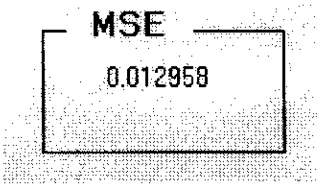


รูปที่ 4.20 แสดงผลในด้านสังเคราะห์ขั้นที่ 2 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step2

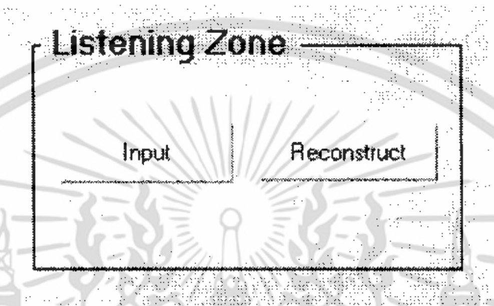


รูปที่ 4.21 แสดงผลในด้านสังเคราะห์ขั้นที่ 3 โดยจะมีส่วนของ input signal และส่วนของ reconstruct step3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 แสดงค่า MSE ที่คำนวณได้เมื่อกดปุ่ม Compare



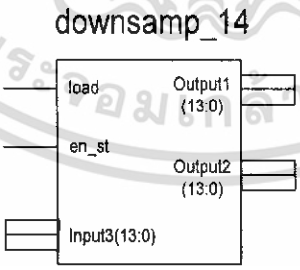
รูปที่ 4.23 แสดงปุ่มที่ใช้ในการฟังเสียงของสัญญาณเสียงอินพุตและสัญญาณเสียงที่กู้กลับมาได้

4.2 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL

4.2.1 ส่วนของการแปลงเวฟเลตในส่วนวิเคราะห์ (Analysis Wavelet Transform)

ในส่วนนี้จะใช้โครงสร้างแบบ Lifting scheme โดยมีส่วนประกอบภายในดังนี้

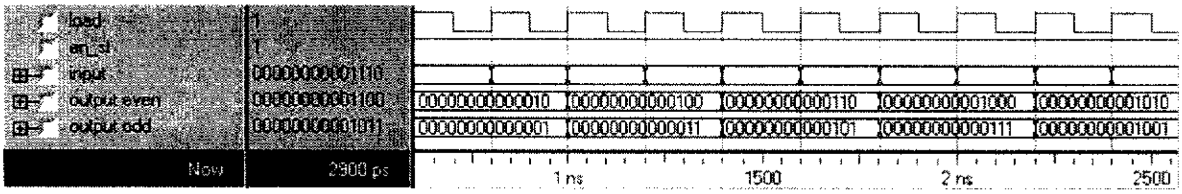
4.2.1.1 ส่วนของวงจรสุ่มค่าตัวอย่าง (DownSampling)



รูปที่ 4.24 แสดงสัญลักษณ์ของวงจรสุ่มค่าตัวอย่าง

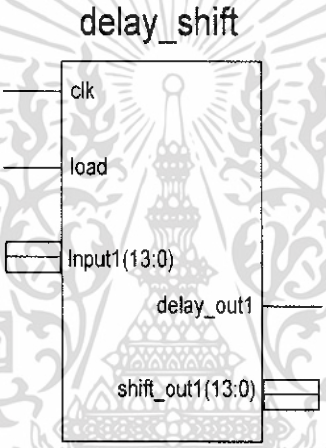
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



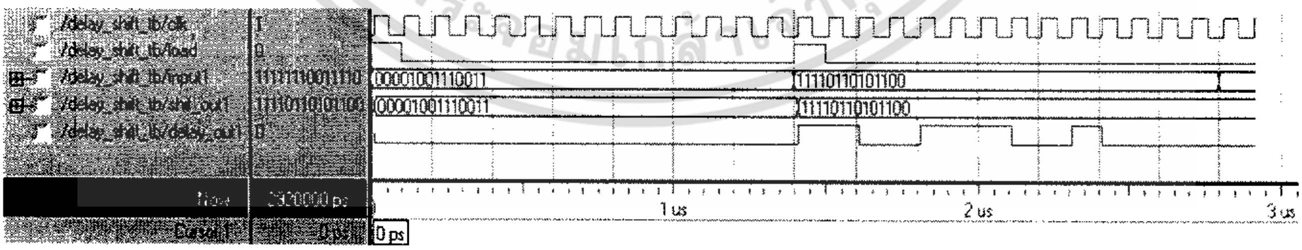
รูปที่ 4.25 แสดงผลการจำลองการทำงานของวงจรสุ่มค่าตัวอย่าง

4.2.1.2 ส่วนของวงจรดีเลย์ (Delay)



รูปที่ 4.26 แสดงสัญลักษณ์ของวงจรดีเลย์

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป

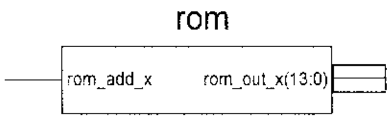


รูปที่ 4.27 แสดงผลการจำลองการทำงานของวงจรดีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

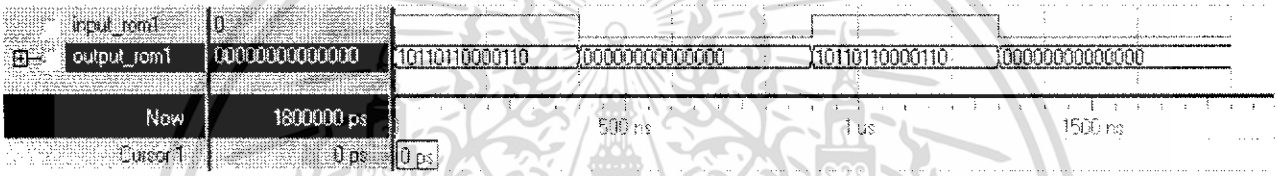
4.2.1.3 ส่วนของวงจรหน่วยความจำ (Rom)

ภายใน State 1 ประกอบด้วย 5 หน่วยความจำ ซึ่งออกแบบโดยใช้วงจร ROM1, ROM2, ROM3, ROM4 และ ROM5



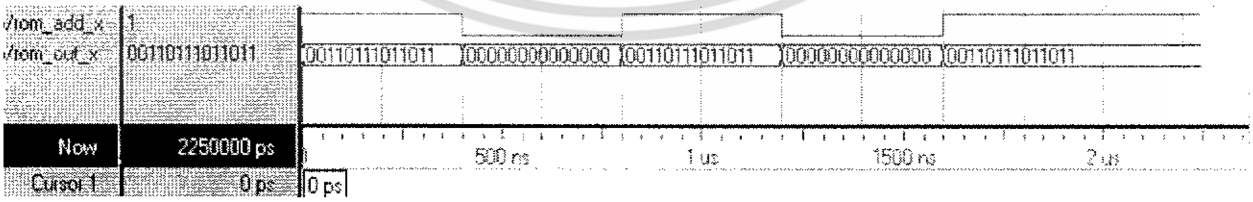
รูปที่ 4.28 แสดงสัญลักษณ์ของวงจรหน่วยความจำ

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



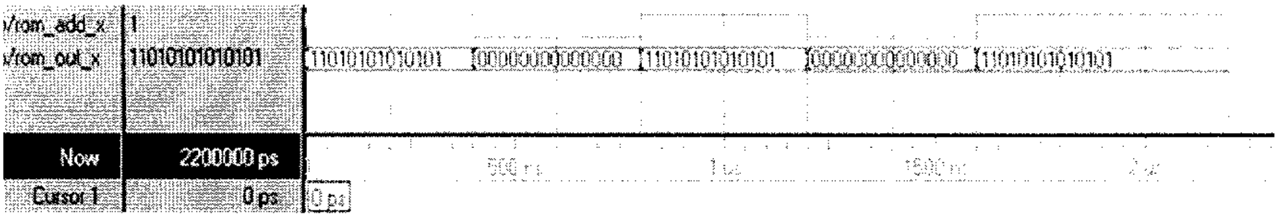
รูปที่ 4.29 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM1

รูปที่ 4.30 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM2

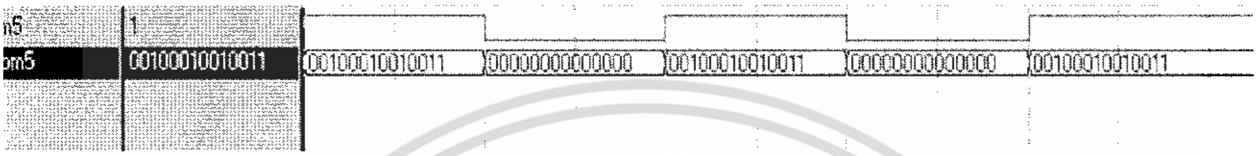


รูปที่ 4.31 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

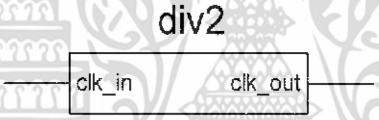


รูปที่ 4.32 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM4



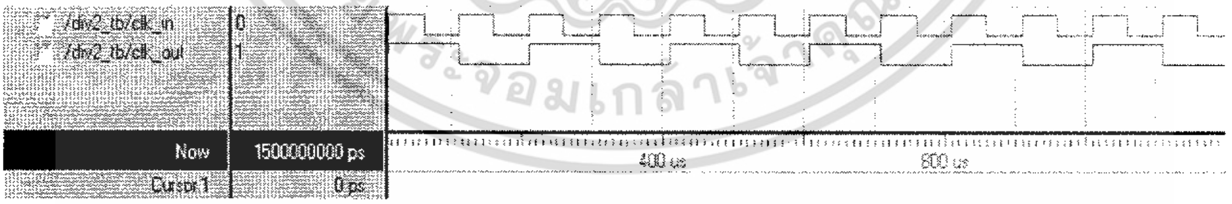
รูปที่ 4.33 แสดงผลการจำลองการทำงานของวงจรหน่วยความจำ ROM5

4.2.1.4 ส่วนของวงจรหารความถี่สอง



รูปที่ 4.34 แสดงสัญลักษณ์ของวงจรหารความถี่สอง

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป

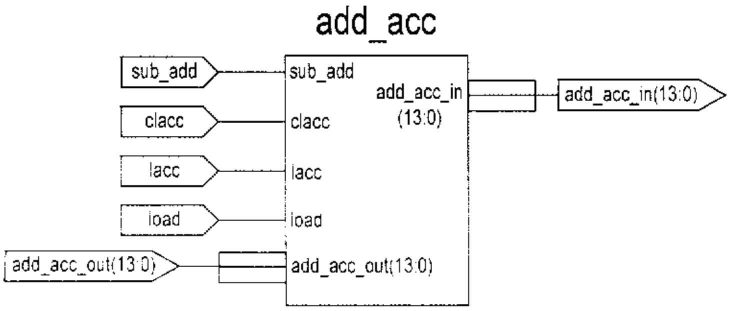


รูปที่ 4.35 แสดงผลการจำลองการทำงานของวงจรหารความถี่สอง

4.2.1.5 ส่วนของวงจรบวกสะสม 14 บิต

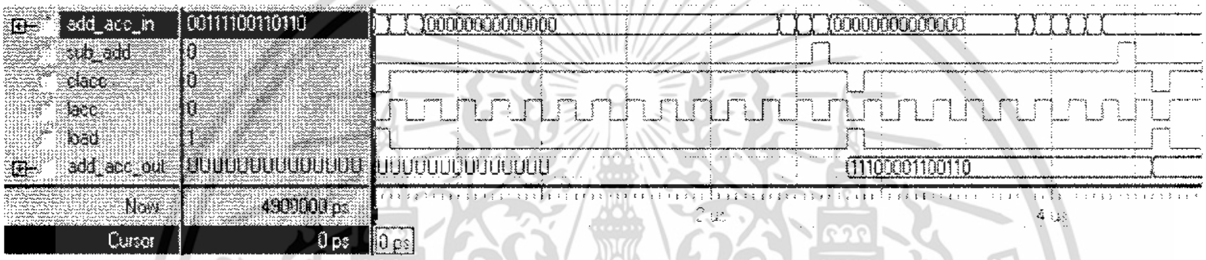
เป็นส่วนที่อยู่ในวงจรแปลงเวฟเลต โดยรับอินพุตมาจากส่วนของวงจรเก็บค่าสัมประสิทธิ์มาทำการบวกสะสมตามหลักการของโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรบวกสะสมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



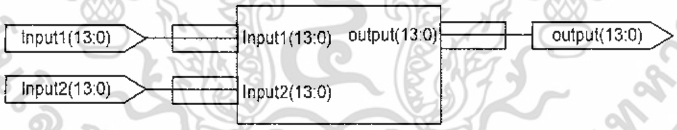
รูปที่ 4.36 แสดงสัญลักษณ์ของวงจรบวกสะสม 14 บิต

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.37



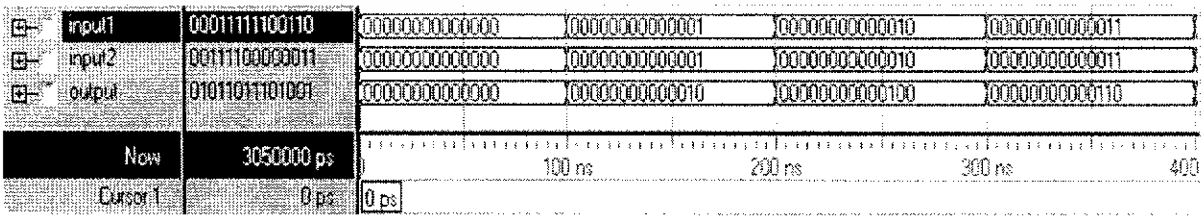
รูปที่ 4.37 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 14 บิต

4.2.1.6 ส่วนของวงจรบวกขนาด 14 บิต



รูปที่ 4.38 แสดงสัญลักษณ์ของวงจรบวกขนาด 14 บิต

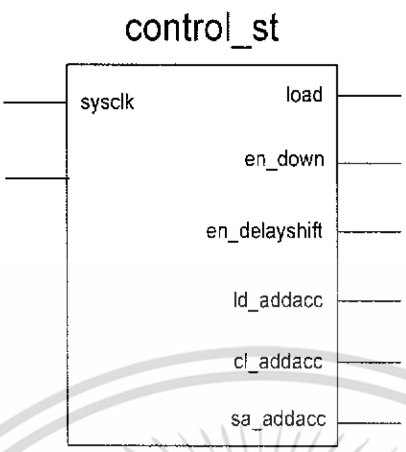
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.39



รูปที่ 4.39 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 14 บิต

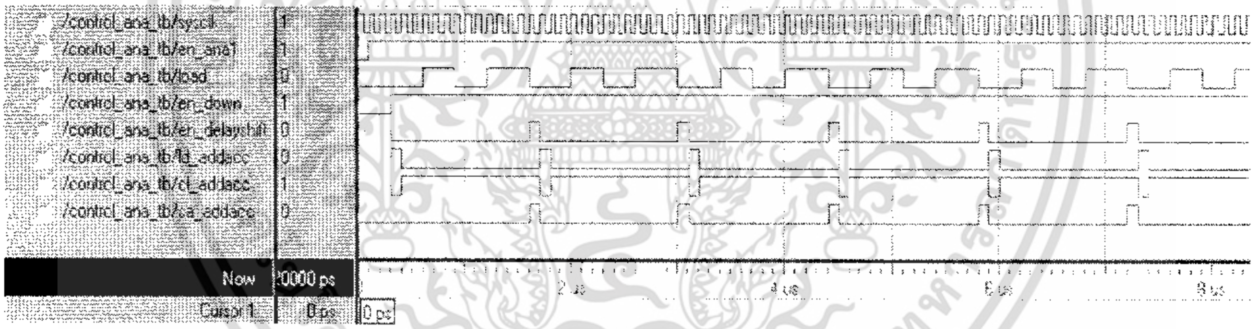
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.7 ส่วนของวงจรสร้างสัญญาณควบคุม state (Control_st)



รูปที่ 4.40 แสดงสัญลักษณ์ของวงจรสร้างสัญญาณควบคุม State

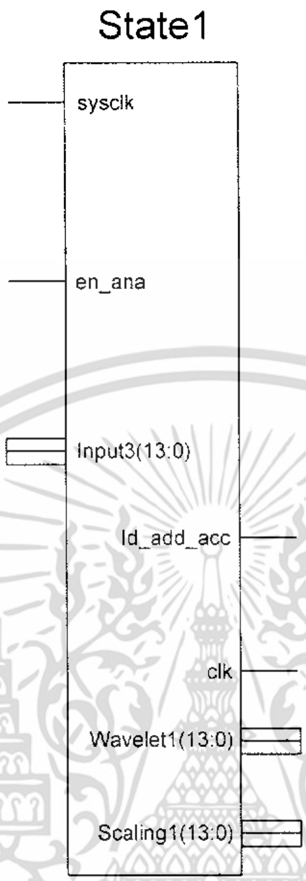
ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.41 แสดงผลการจำลองการทำงานของวงจรสร้างสัญญาณควบคุม State

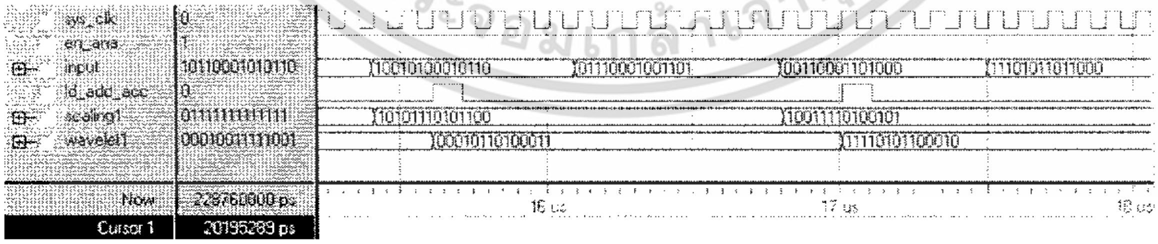
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.8 ส่วนของวงจรควบคุมและวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1



รูปที่ 4.42 แสดงสัญลักษณ์ของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1

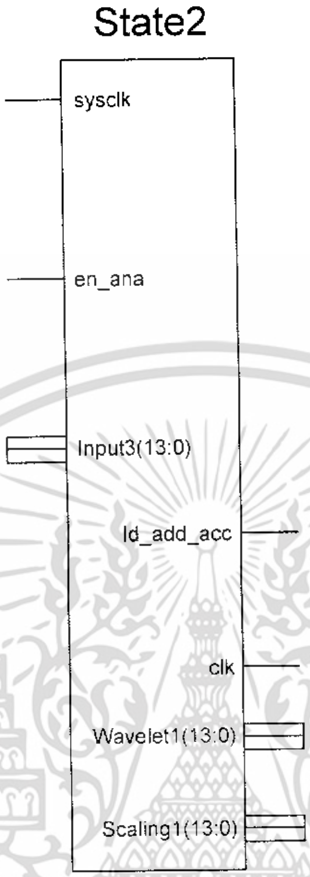
ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7c แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.43 แสดงผลการจำลองการทำงานของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1

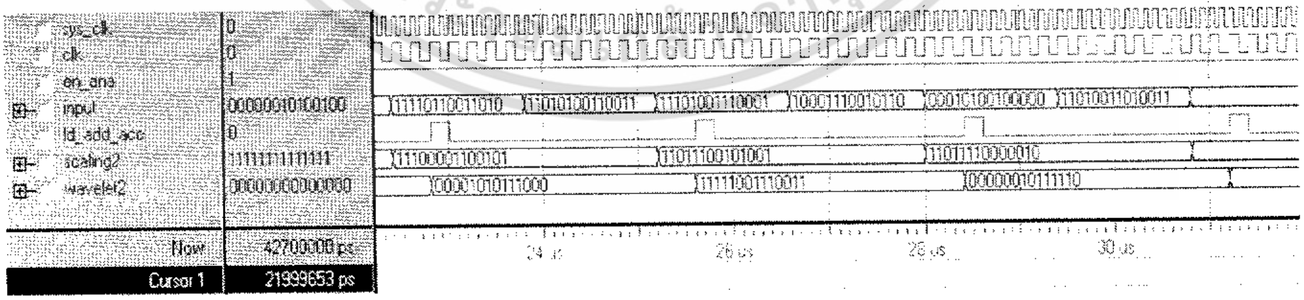
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.9 ส่วนของวงจรควบคุมและวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2



รูปที่ 4.44 แสดงสัญลักษณ์ของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2

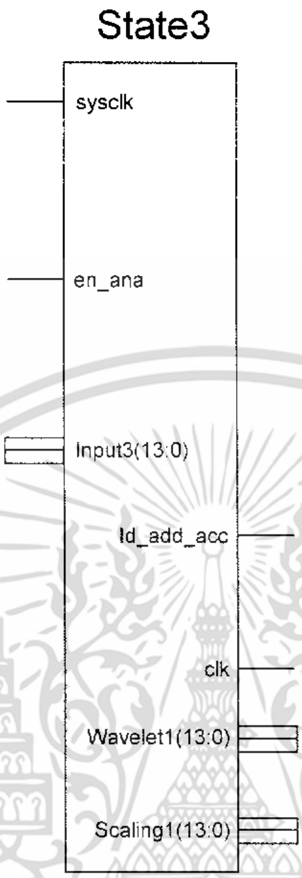
ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.45 แสดงผลการจำลองการทำงานของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2

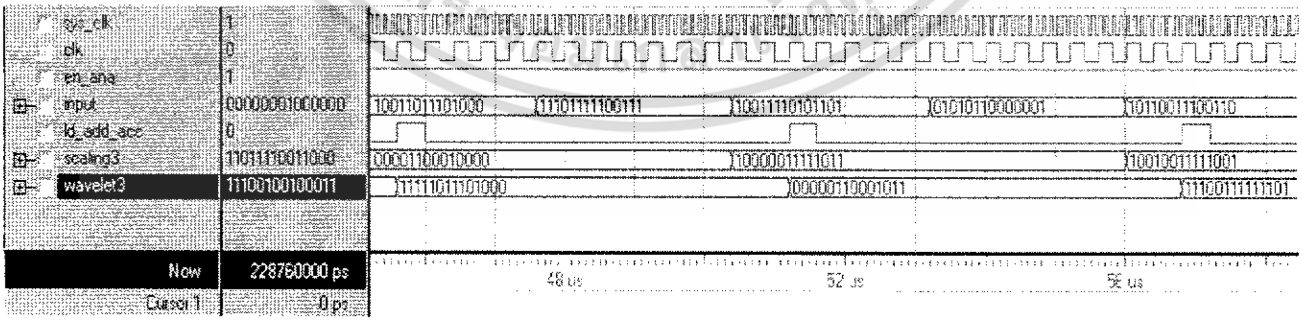
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.10 ส่วนของวงจรควบคุมและวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3



รูปที่ 4.46 แสดงสัญลักษณ์ของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป

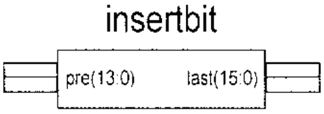


รูปที่ 4.47 แสดงผลการจำลองการทำงานของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

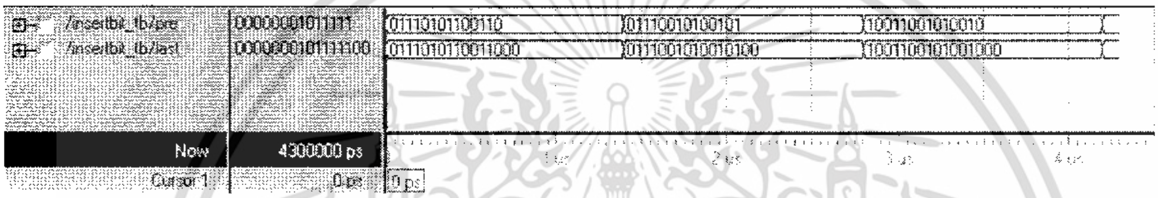
4.2.1.11 ส่วนของวงจรแทรกบิต

ทำหน้าที่แทรกบิตท้ายสองบิตลงในสัญญาณก่อนที่จะนำไปเก็บเข้า RAM มีสัญลักษณ์ดังรูป



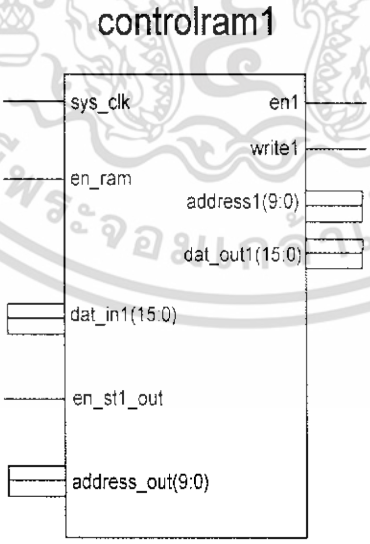
รูปที่ 4.48 แสดงสัญลักษณ์ของวงจรแทรกบิต

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.49 แสดงผลการจำลองการทำงานของวงจรแทรกบิต

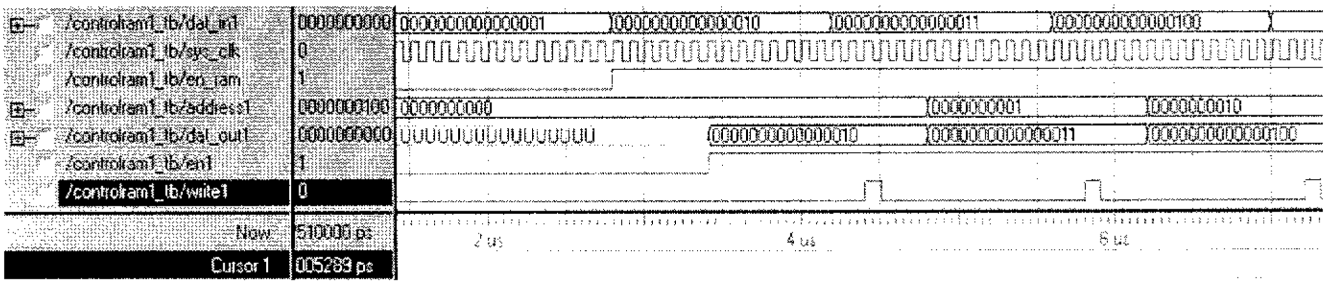
4.2.1.12 ส่วนของวงจรควบคุมหน่วยความจำ(controlram1)



รูปที่ 4.50 แสดงสัญลักษณ์ของวงจรควบคุมหน่วยความจำ RAM

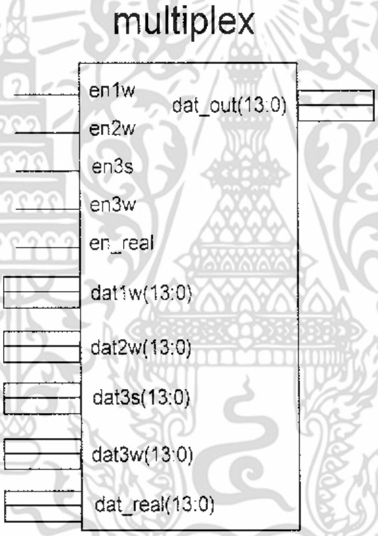
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.51 แสดงผลการจำลองการทำงานของวงจรควบคุมวงจรหน่วยความจำ(controlram1)

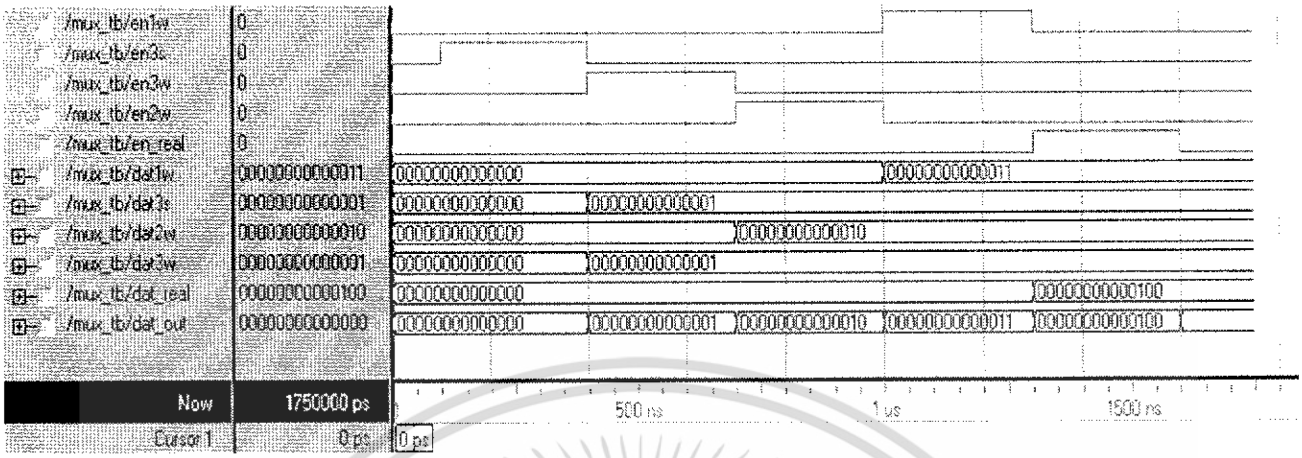
4.2.1.13 ส่วนของวงจรมัลติเพล็กซ์



รูปที่ 4.52 แสดงสัญลักษณ์ของวงจรส่วนของวงจรมัลติเพล็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

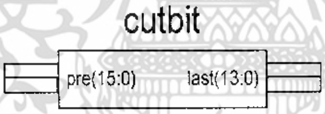
ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.53 แสดงผลการจำลองการทำงานของวงจรควบคุมวงจรวงจรมัลติเพล็กซ์

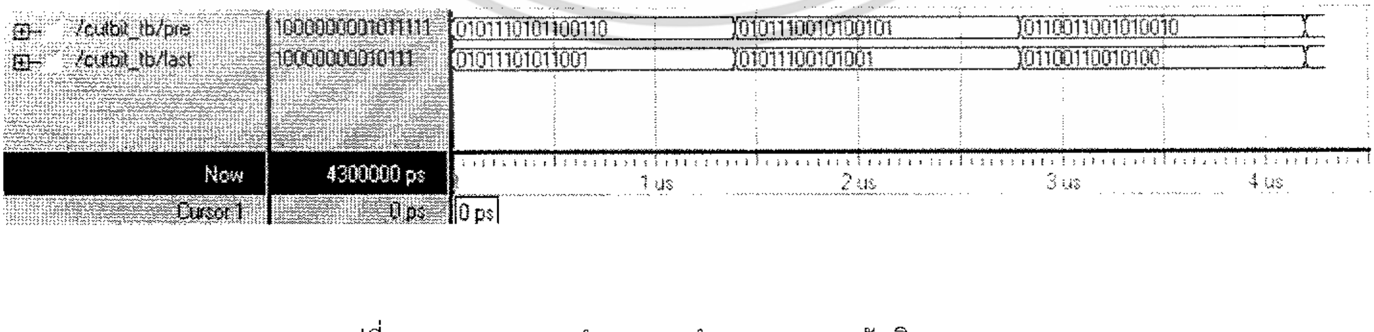
4.2.1.14 ส่วนของวงจรตัดบิต (Cut bit)

เป็นส่วนที่ใช้ในการตัดบิตท้ายของข้อมูลที่ออกจาก RAM ออก 2 บิต ให้เหลือ 14 บิต เพื่อนำไปทำการคำนวณต่อไป



รูปที่ 4.54 แสดงสัญลักษณ์ของวงจรตัดบิต

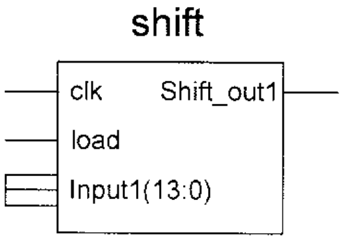
ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.55 แสดงผลการจำลองการทำงานของวงจรตัดบิต

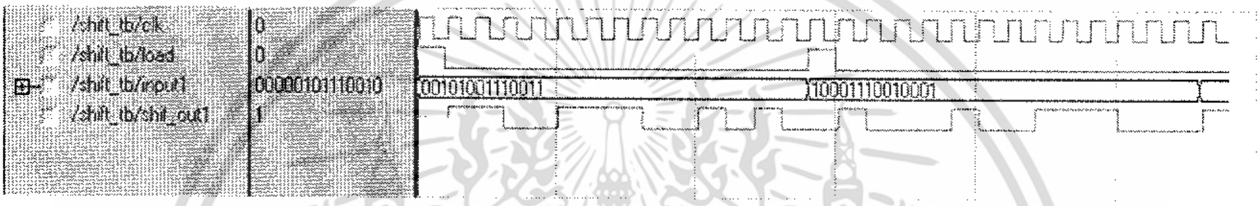
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.15 ส่วนของวงจรชิฟบิต



รูปที่ 4.56 แสดงสัญลักษณ์ของวงจรชิฟบิต

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7e แล้วจะได้ผลการทดลอง ดังรูป



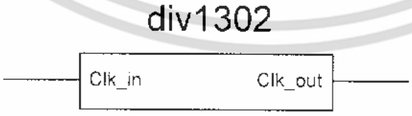
รูปที่ 4.57 แสดงผลการจำลองการทำงานของวงจรชิฟบิต

4.2.2 ส่วนของวงจรส่งบิตข้อมูลจากบอร์ด FPGA ออกทางพอร์ตอนุกรม

แบ่งออกเป็น 2 ส่วนคือ

4.2.2.1 ส่วนของวงจรหารความถี่

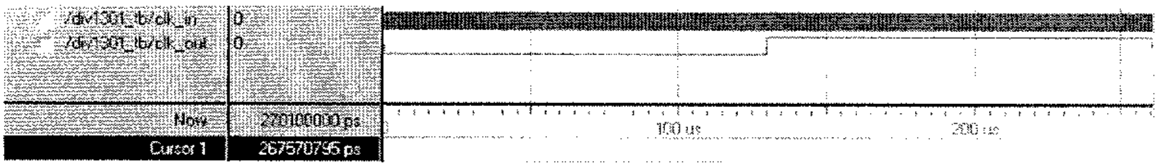
ส่วนของวงจรหารความถี่ ทำหน้าที่หารความถี่ที่ของระบบให้ได้เอาท์พุทเป็นความถี่บอดเรต(Baud Rate) ที่ใช้ในการรับ-ส่งบิตข้อมูลทางพอร์ตอนุกรม โครงงานนี้ใช้แหล่งกำเนิดความถี่ 25 MHz สำหรับใช้เป็นความถี่ของระบบ แต่เราต้องหารความถี่เพื่อให้ได้เท่ากับความถี่บอดเรต 9600 บิตต่อวินาที สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรหารความถี่ได้ สัญลักษณ์ ดังรูป



รูปที่ 4.58 แสดงสัญลักษณ์ของวงจรหารความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

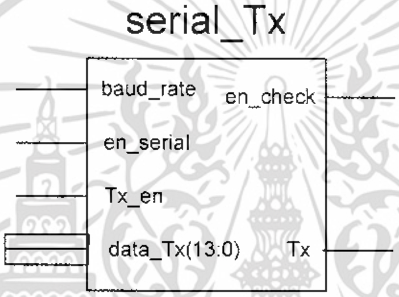
ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7c แล้วจะได้ผลการทดลอง ดังรูป



รูปที่ 4.59 แสดงผลการจำลองการทำงานของวงจรหารความถี่

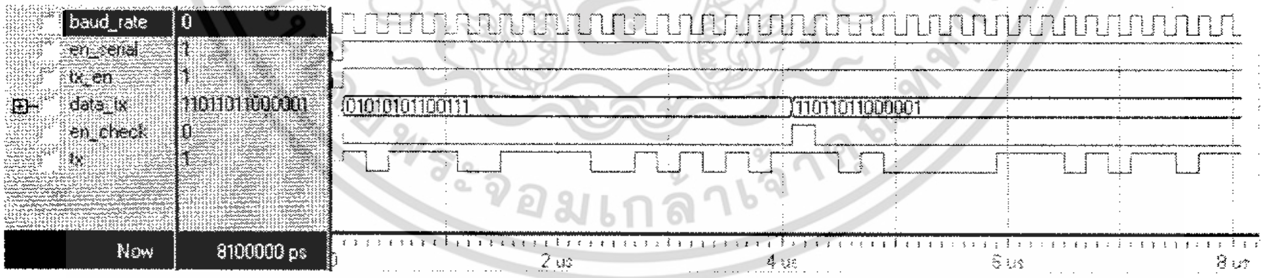
4.2.2.2 ส่วนของวงจรส่งบิตข้อมูลออกทางพอร์ตอนุกรม

โดยจะอินพุตบิตข้อมูลขนาด 14 บิต แล้วส่งข้อมูลออกมาทีละบิตผ่านทางพอร์ท RS-232



รูปที่ 4.60 แสดงสัญลักษณ์ของวงจรส่งบิตข้อมูล

ทำการจำลองการทำงานด้วยโปรแกรม ModelSim XE II 5.7c แล้วจะได้ผลการทดลอง ดังรูป

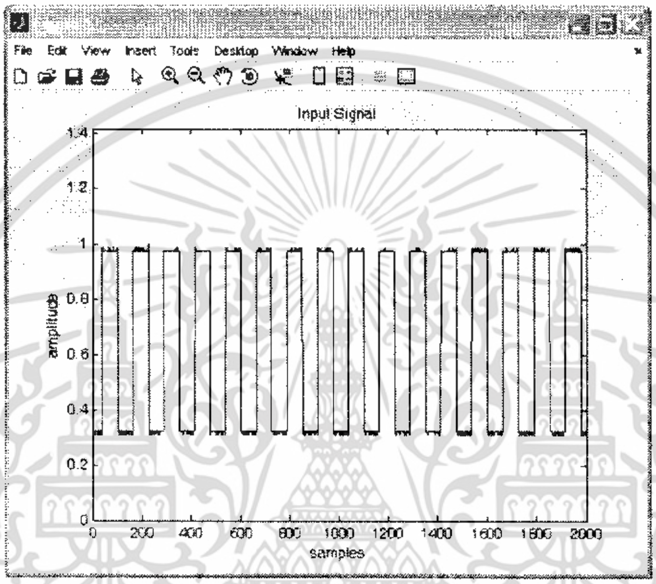


รูปที่ 4.61 แสดงผลการจำลองการทำงานของวงจรส่งบิตข้อมูล

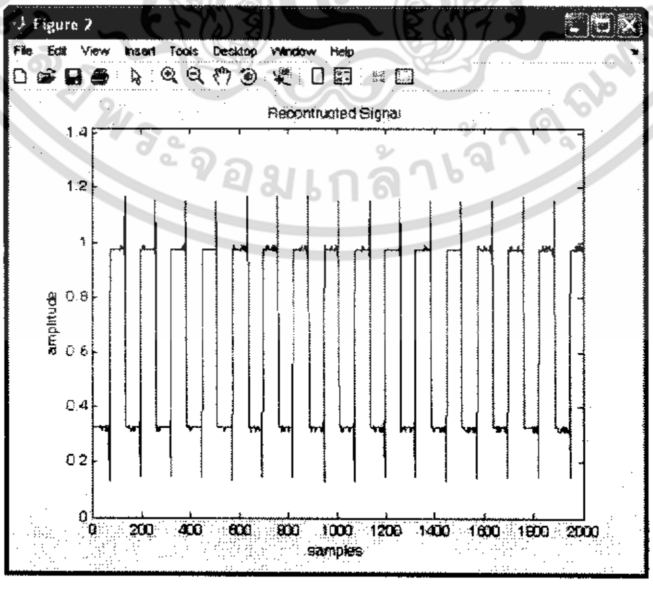
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ส่วนของการแสดงผลผ่านจอคอมพิวเตอร์

จากการนำสัญญาณอินพุตจากเครื่องกำเนิดสัญญาณส่งไปยังโปรแกรมที่เขียนขึ้นลงบนบอร์ด FPGA แล้วทำการเก็บผลการทดลองโดยการส่งข้อมูลจากบอร์ด FPGA ไปยังพอร์ทอนุกรมของคอมพิวเตอร์ แล้วแสดงผลทั้งหมดผ่านทางโปรแกรม MATLAB ในการเก็บผลการทดลอง จะทำการป้อนสัญญาณอินพุตเป็นสัญญาณรูปสี่เหลี่ยม(Square wave) และสัญญาณรูปซายน์ (Sine wave) แล้วนำเอาท์พุตที่ได้มาทำการแปลงเวฟเลตส่วนสังเคราะห์โดยใช้โปรแกรมแมทแลบ (MATLAB) 2 ขั้น พร้อมทั้งแสดงสัญญาณอินพุตกับสัญญาณเอาท์พุตที่ทำการกู้กลับมาได้ โดยจะมีหน้าต่างแสดงผลดังนี้

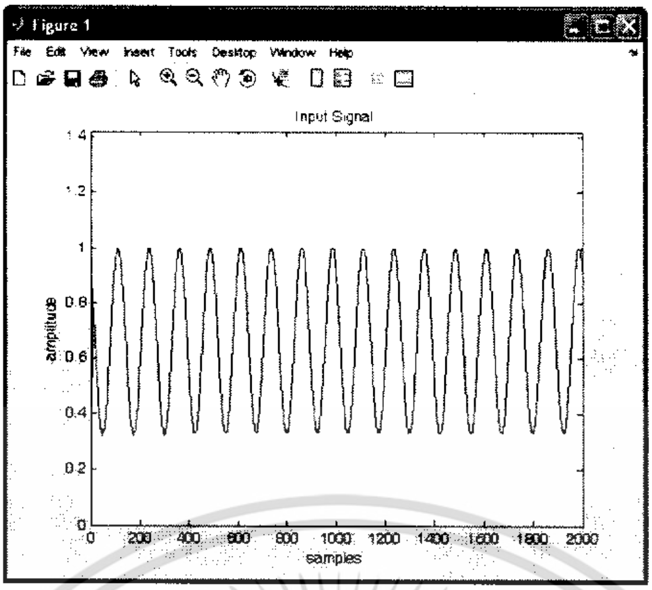


รูปที่ 4.62 แสดงสัญญาณอินพุตเป็นสัญญาณสี่เหลี่ยม

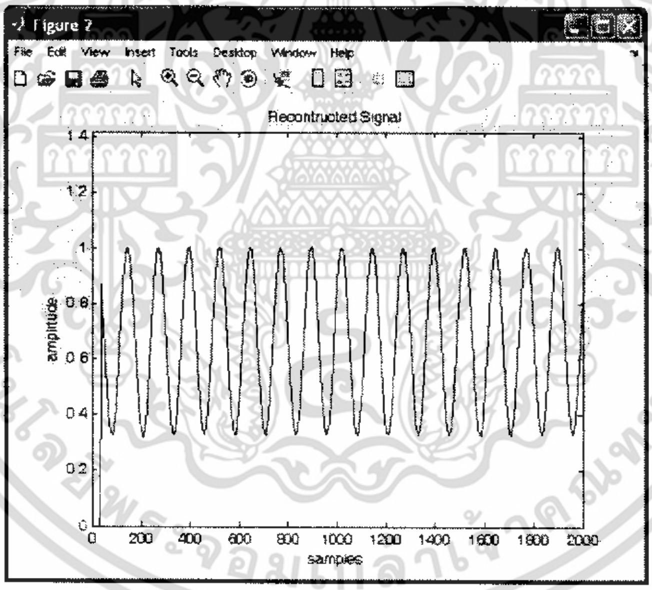


รูปที่ 4.63 แสดงสัญญาณเอาท์พุตสัญญาณสี่เหลี่ยมที่กู้กลับมาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



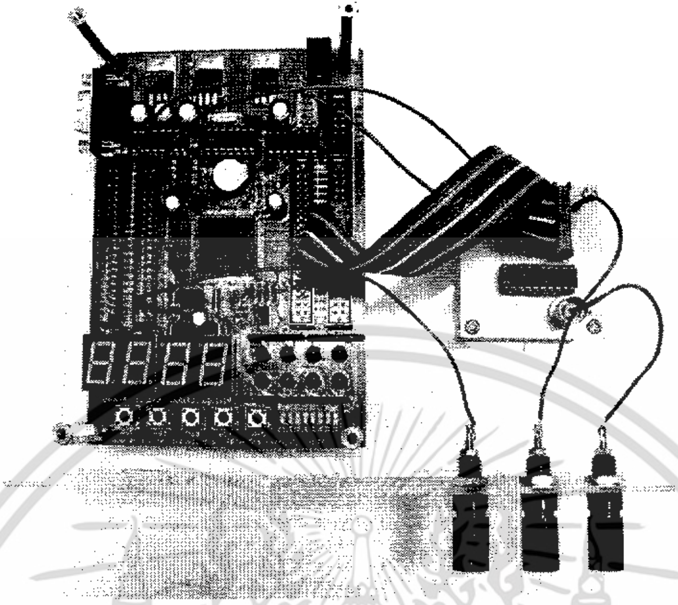
รูปที่ 4.62 แสดงสัญญาณอินพุตรูปซายน์



รูปที่ 4.65 แสดงสัญญาณเอาท์พุตรูปซายน์ที่กู้กลับมาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ภาพอุปกรณ์ที่ใช้ในการทดสอบการแปลงเวฟเลตโดยใช้บอร์ด FPGA



รูปที่ 4.66 แสดงบอร์ดเอฟพีจีเอที่ทำการเชื่อมต่อกับพอร์ทอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

การแปลงเวฟเลตในปัจจุบันเป็นวิธีที่นิยมใช้ในการบีบอัดสัญญาณ เนื่องจากไม่มีการสูญเสียของข้อมูล(lossless) นั่นคือสามารถกู้สัญญาณกลับคืนได้อย่างสมบูรณ์ นอกจากนี้ยังสามารถนำวิธีการของเวฟเลตไปประยุกต์ใช้ในด้านต่างๆได้ เช่น การลดสัญญาณรบกวนของข้อมูลและการตรวจจับกรอบของสัญญาณ เป็นต้น ดังนั้นจึงมีการนำเวฟเลตไปใช้อย่างแพร่หลายทั้งในด้านการแพทย์ ซึ่งต้องการส่งสัญญาณแบบไม่มีการสูญเสีย เพื่อให้นำไปวินิจฉัยโรคได้อย่างถูกต้อง

การแปลงเวฟเลตนั้น เป็นการแปลงแบบไม่มีการสูญเสีย (lossless) แต่ผลการทดลองในส่วนของการจำลองการทำงานด้วยโปรแกรมแมทแลบ พบว่า ค่าความผิดพลาดซึ่งคำนวณมาจากค่า MSE เกิดขึ้นสาเหตุเกิดจากค่าจุดทศนิยมในโปรแกรมแมทแลบนั้นมีขอบเขตที่จำกัด ทำให้สัมประสิทธิ์ของตัวกรองสัญญาณถูกปัดค่า ทำให้เกิดความคลาดเคลื่อนขึ้น และนอกจากนี้ความผิดพลาดยังเกิดจากสัญญาณที่กู้กลับคืนมาได้มีการเลื่อนเฟส ทำให้สัญญาณเกิดการหน่วงขึ้น เนื่องจากค่าความผิดพลาดได้คำนวณมาจาก MSE ซึ่งการหน่วงเฟสทำให้ MSE มีค่ามาก

ในส่วนของการออกแบบวงจรดิจิตอลโดยใช้ภาษา VHDL จะใช้ระบบการแทนข้อมูลตัวเลข Fixed Point 14 บิต ซึ่งในวงจรคูณนั้นผลการคูณที่ได้จะมีค่าเกิน 14 บิต ทำให้ต้องมีการตัดทศนิยมตำแหน่งท้ายทิ้ง ซึ่งทำให้เกิดความผิดพลาดขึ้น

นอกจากนี้ในการออกแบบวงจรพบว่ามีปัญหาเกิดขึ้น คือ เมื่อนำวงจรที่ทำการจำลองการทำงานแล้วได้ผลลัพธ์ถูกต้อง 2 วงจรมาเชื่อมต่อกัน กลับให้ผลลัพธ์ที่ผิดพลาด ซึ่งในส่วนนี้คาดว่าเกิดจากมีการเหลื่อมล้ำทางเวลาระหว่าง 2 วงจร ซึ่งทางแก้ไขคือ ควรมีวงจรหน่วงเวลาที่เหมาะสมคั่นระหว่างทั้งสองวงจร จะทำให้ได้ผลลัพธ์ที่ถูกต้อง

ปัญหาอีกอันหนึ่งที่พบคือ ปัญหาที่เกิดจากทรัพยากรภายในบอร์ด FPGA มีไม่พอ ทำให้ไม่สามารถสังเคราะห์วงจรลงบอร์ดได้ ซึ่งทางแก้ไขคือ ก่อนการทำงานต้องออกแบบวงจรให้รัดกุม ไม่ใช้ทรัพยากรฟุ่มเฟือยเกินไปหรือถ้าไม่สามารถลดการใช้ทรัพยากรลงได้ ก็อาจซื้อบอร์ด FPGA ที่มีจำนวนทรัพยากรมากกว่านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

[1] Martin Vetterli, "Wavelets and Subband Coding ", Prentice Hall PTR, Englewood Cliffs, New Jersey 07632,1995.

[2] Gilbert Strang , Truong Nguyen, " Wavelets and Filter Banks", Wellesley - Cambridge Press,1996.

[3] C. Sidney Burrus , Ramesh A. Gopinath , and Haitao Guo, " Introduction to Wavelets and Wavelet Transforms", Prentice Hall,Upper Saddle River,New Hersey 07458,1998.

[4] Sanjit K. Mitra , " Digital Signal Processing ", Mc Graw Hill, 2002.

[5] Randy K. Young, "Wavelet Theory and its application", Kluwer Academics Publishers.

[6] Michel Misiti , Yves Misiti , Georges Oppenheim , Jean-Michel Poggi, " Wavelets Toolbox",The Math Works,Inc.,24 Prime Park Way,Natick,Mass. 01760-1500

[7] ชำนาญ ปัญญาใส, วัชรากร หนูทอง, "ภาษา VHDL สำหรับการออกแบบวงจรดิจิตอล", ซีเอ็ด ยูเคชั่น, กรุงเทพฯ,2547.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้