สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตัวแปลงเวฟเลตโดยอุปกรณ์ FPGA

Implementation of Wavelet Transform using FPGA

โดย

นาย คงฤทธิ์ เตชะวชิรศิริ

นางสาว จันทร์จรัส รุ่งสิริสุวรรณ

นางสาว จิตตินี รุ่งจตุรงค์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 61814
วัน,เดือน,ปี 21 ก.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

ภาควิชา
วิศวกรรมโทรคมนาคม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวแปลงเวฟเลตโดยอุปกรณ์ FPGA

Implementation of Wavelet Transform using FPGA

โดย

นาย คงฤทธิ์	เตชะวชิรศิริ	44010048
นางสาว จันทร์จรัส	รุ่งสิริสุวรรณ	44010065
นางสาว จิตตินี	รุ่งจตุรงค์	44010070

อาจารย์ที่ปรึกษา

ผศ. อัครพล ตรีรัตน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2547

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **ตัวแปลงเวฟเลตโดยอุปกรณ์ FPGA**

Implementation of Wavelet Transform using FPGA

ผู้จัดทำ

นาย คงฤทธิ์ เตชะวชิรศิริ 44010048

นางสาว จันทร์จรัส รุ่งสิริสุวรรณ 44010065

นางสาว จิตตินี รุ่งจตุรงค์ 44010070

อาจารย์ที่ปรึกษา

(ผศ. อัครพล ตรีรัตน์)

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวแปลงเวฟเลตโดยอุปกรณ์ FPGA

Implementation of Wavelet Transform using FPGA

โดย นาย คงฤทธิ์ เตชะวชิรศิริ 44010048

นางสาว จันทร์จรัส รุ่งสิริสุวรรณ 44010065

นางสาว จิตตินี รุ่งจตุรงค์ 44010070

อาจารย์ที่ปรึกษา ผศ. อัครพล ตรีรัตน์

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการศึกษาการแปลงเวฟเลต (Wavelet Transform) โดยอาศัยโครงสร้างของฟิลเตอร์แบงค์ (Filter Bank) ที่มีโครงสร้างแบบโพลีเฟส (Polyphase Structure) โดยเลือกใช้สัมประสิทธิ์ดาร์บีชี (Daubechies) และอาศัยการจัดรูปแบบแบบเลขคณิตกระจาย (Distributed Arithmetic) เพื่อให้ง่ายต่อการประมวลผลและลดจำนวนอุปกรณ์ในการนำไปใช้งาน โดยในปริญญานิพนธ์นี้จะทำการจำลองการทำงานโดยใช้โปรแกรมแมทแลบ (MATLAB) พร้อมทั้งทำการออกแบบวงจรลงบน FPGA (Field Programmable Gate Array) เพื่อทดสอบการใช้งานจริงต่อไป

ABSTRACT

This thesis presents a design and implementation of Wavelet transform processor by using filter bank structure with Daubechies Coefficients. Generally, Filter banks can be used to extract wavelet coefficients. In this thesis we also use the concept of Distributed Arithmetic in proposed structure. In this thesis, the simulation of the processor will be done using MATLAB. Finally, the wavelet transform processor will be implemented using Field Programmable Gate Array.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของปริญญานิพนธ์	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	1
1.3 ขอบเขตของปริญญานิพนธ์	1
1.4 เนื้อหาของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 ความรู้เบื้องต้นของเวฟเลต	3
2.1.1 ระบบของเวฟเลต	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
2.4.2 หลักการสังเคราะห์	11
2.5 ฟิลเตอร์แบงค์	12
2.5.1 การกำจัดการเกิดการซ้อนทับกันและผลที่ได้จากวงจรกรองความถี่	15
2.5.2 โครงสร้างของวงจรกรองความถี่ทั้งรูปแบบโดยตรงและรูปแบบโพลีเฟส	16
2.5.2.1 วงจรกรองความถี่ที่มีโครงสร้างแบบโดยตรง	16
2.5.2.2 วงจรกรองความถี่ที่มีโครงสร้างแบบโพลีเฟส	17
2.6 ภาษาวีเอชดีแอล	20
2.7.1 ประวัติความเป็นมาของภาษาวีเอชดีแอล	20
2.7.2 การออกแบบระบบดิจิตอล	20
2.7.3 การออกแบบจากบนลงล่าง	22
2.7 เอฟพีจีเอ	24
2.7.1 การออกแบบวงจรเชิงเลขด้วยชิพอุปกรณ์เอฟพีจีเอ	25
2.7.2 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว	25
2.8 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย	26
2.8.1 ระบบตัวเลข	26
2.8.1.1 รูปแบบจำนวนโดยตรง	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.1.2 รูปแบบจำนวนอิงดรรชนี 28
2.8.2 ทฤษฎีเลขคณิตกระจาย 29
2.9 วงจรแปลงอนาลอกเป็นดิจิตอล 34
2.9.1. แบบใช้วงจรเปรียบเทียบขนาน หรือแบบแฟลช 34
2.9.2 วงจรเอทูดีที่ใช้การอินทีเกรต 35
2.9.3 วงจรเปลี่ยนสัญญาณเอทูดีที่ใช้วงจรนับและวงจรดีทูเอประกอบกัน 36
2.9.4 วงจรเปลี่ยนสัญญาณเอทูดีแบบใช้การประมาณค่า 37
2.9.5 การสุ่มและการคงค่า 37
2.10 พอร์ตอนุกรม 38
2.10.1 การสื่อสารข้อมูล 38
2.10.1.1 การสื่อสารข้อมูลแบบขนาน 38
2.10.1.2 การสื่อสารข้อมูลแบบอนุกรม 38
2.10.2 การอินเตอร์เฟสตามมาตรฐาน RS-232 39
2.10.3 นันโมเด็ม 40
2.10.4 การเชื่อมต่อระหว่าง DB-9 กับ FPGA 41
บทที่ 3 การคำนวณและการสร้าง 42
3.1 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL 42
3.1.1 สัมประสิทธิ์ที่ใช้ในการแปลงเวฟเลตโดยเลือกใช้สัมประสิทธิ์ดาร์บีชี 4 ค่า 42
3.1.2 ส่วนของการแปลงเวฟเลต 5 ระดับ 43
3.1.3 ส่วนของการแปลงเวฟเลตกลับ 5 ระดับ 44
3.1.4 ส่วนประกอบภายในของฟิลเตอร์แบบโพลีเฟส 45
3.1.4.1 วงจรหน่วงเวลา 45
3.1.4.2 วงจรสุ่มค่าตัวอย่าง 45
3.1.4.3 วงจรเพิ่มค่าตัวอย่าง 46
3.1.4.4 วงจรบวก 46
3.1.4.5 วงจรคูณ 47
3.1.5 การประยุกต์ใช้งาน DA กับการแปลงเวฟเลต โดยใช้ฟิลเตอร์ที่มีโครงสร้างแบบโพลีเฟส 47
3.1.5.1 วงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม 48
3.1.5.2 หน่วยความจำ 49
3.1.5.3 วงจรบวกหรือวงจรลบ 50
3.1.5.4 ส่วนเก็บค่าผลการคำนวณ 50
3.1.6 Random Access Memory 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA 53
3.2.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล 53
3.2.2 วงจรมัลติเพล็กซ์ 53
3.2.3 ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ 53
3.2.4 ส่วนของการรับบิทข้อมูลจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ผ่านทางพอร์ตอนุกรม 54
บทที่ 4 การทดลองและผลการทดลอง 56
4.1 ส่วนของการ Simulate โดยใช้โปรแกรม MATLAB 56
4.1.1 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ 56
4.1.1.1 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 1 56
4.1.1.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 2 57
4.1.1.3 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 3 58
4.1.1.4 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 4 58
4.1.1.5 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 5 59
4.1.2 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ 60
4.1.2.1 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 1 60
4.1.2.2 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 2 61
4.1.2.3 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 3 61
4.1.2.4 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 4 62
4.1.2.5 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 5 63
4.2 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL 64
4.2.1 ส่วนของการแปลงเวฟเลตในส่วนวิเคราะห์ 64
4.2.1.1 ส่วนของวงจรสุ่มค่าตัวอย่าง 64
4.2.1.2 วงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม 64
4.2.1.3 หน่วยความจำ 65
4.2.1.4 ส่วนของวงจรบวกสะสม 14 บิท 66
4.2.1.5 ส่วนของวงจรบวกขนาด 14 บิท 66
4.2.1.6 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 67
4.2.1.7 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 69
4.2.1.8 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 71
4.2.1.9 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4 73
4.2.1.10 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5 74
4.2.1.11 ส่วนของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนวิเคราะห์ 75
4.2.2 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.1 ส่วนของวงจรแทรกบิทจำนวน 2 บิท 75
4.2.2.2 วงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 16 บิท 76
4.2.2.3 ส่วนของวงจรควบคุมตัวเก็บค่าสัมประสิทธิ์ในแต่ละขั้น 76
4.2.2.4 ส่วนของวงจรตัดบิท 2 บิท 79
4.2.3 ส่วนของการแปลงเวฟเลตในส่วนสังเคราะห์ 79
4.2.3.1 หน่วยความจำ 80
4.2.3.2 ส่วนของวงจรเพิ่มค่าตัวอย่าง 81
4.2.3.3 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1 81
4.2.3.4 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2 82
4.2.3.5 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3 83
4.2.3.6 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4 85
4.2.3.7 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5 87
4.2.3.8 ส่วนของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนสังเคราะห์ 89
4.3 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA 90
4.3.1 ส่วนของวงจรควบคุมการสุ่มค่าตัวอย่าง 90
4.3.2 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 8 บิท 91
4.3.3 ส่วนของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ 91
4.3.4 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 16 บิท 91
4.3.5 ส่วนของวงจรมัลติเพล็กซ์ 92
4.3.6 ส่วนของการส่งข้อมูลออกจากบอร์ด FPGA ผ่านทางพอร์ตอนุกรม 93
4.3.6.1 ส่วนของวงจรหารความถี่ 93
4.3.6.2 ส่วนของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม 93
4.3.6.3 ส่วนของวงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม 94
4.4 ส่วนของการแสดงผลผ่านจอคอมพิวเตอร์ 94
4.5 ภาพอุปกรณ์ที่ใช้ในการทดสอบการแปลงเวฟเลต 100
บทที่ 5 บทวิจารณ์และบทสรุป 101
กิตติกรรมประกาศ
หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาคส่ง 9
รูปที่ 2.7 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสองขั้นของภาคส่ง 10
รูปที่ 2.8 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสามขั้นของภาคส่ง 10
รูปที่ 2.9 แถบความถี่ของภาคส่ง 10
รูปที่ 2.10 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาครับ 12
รูปที่ 2.11 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสองขั้นของภาครับ 12
รูปที่ 2.12 ฟิลเตอร์แบงค์ 12
รูปที่ 2.13 ผลตอบสนองทางความถี่ของวงจรกรองความถี่ต่ำและสูง 13
รูปที่ 2.14 แสดงการสลับเครื่องหมายในการหาค่าสัมประสิทธิ์ 15
รูปที่ 2.15 แสดงความสัมพันธ์ระหว่างสัมประสิทธิ์ทั้งภาครับและภาคส่ง 16
รูปที่ 2.16 ฟิลเตอร์แบงค์เชิงตั้งฉากที่มีสัมประสิทธิ์ 4 ค่า 16
รูปที่ 2.17 บล็อกไดอะแกรมของโครงสร้างโดยตรงของวงจรกรองความถี่ 16
รูปที่ 2.18 บล็อกไดอะแกรมการแสดงให้เห็นจริงของโพลีเฟส ของฟังก์ชันการส่งผ่าน 18
รูปที่ 2.19 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนวิเคราะห์ 19
รูปที่ 2.20 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนสังเคราะห์ 19
รูปที่ 2.21 แสดงรูปแบบส่วนวิเคราะห์และส่วนสังเคราะห์ ของฟิลเตอร์แบงค์ 20
รูปที่ 2.22 แสดงขั้นตอนการออกแบบระบบดิจิตอล 21
รูปที่ 2.23 แสดงการออกแบบระบบเส้นทางข้อมูล 22
รูปที่ 2.24 แสดงขั้นตอนการออกแบบจากบนลงล่าง 23
รูปที่ 2.25 แสดงผังการแบ่งกลุ่มของวงจรรวมเอซิก 24
รูปที่ 2.26 แสดงการจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิทจำนวนเต็มและบิทเศษส่วน 26
รูปที่ 2.27 แสดงการจัดรูปแบบจำนวนโดยตรงที่มีแต่บิทเศษส่วน 27
รูปที่ 2.28 แสดงการจัดรูปแบบจำนวนอิงดรรชนี 28
รูปที่ 2.29 แสดงการคูณแบบเลขส่วนเติมเต็มสองโดยใช้ทฤษฎีเลขคณิตกระจาย 32
รูปที่ 2.30 แสดงการต่อวงจร Parallel Comparator A/D Converter 34
รูปที่ 2.31 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารข้อมูลแบบขนาน 38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.32 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารแบบอนุกรม 38
รูปที่ 2.33 แสดงลักษณะของคอนเน็กเตอร์แบบ DB-9 39
รูปที่ 2.34 แสดงบล็อกไดอะแกรมการเชื่อมต่อแบบนันโมเด็ม 40
รูปที่ 2.35 แสดงการเชื่อมต่อระหว่างคอนเน็กเตอร์แบบ DB-9 และ บอร์ด FPGA
โดยผ่าน MAX3232 41
รูปที่ 3.1 บล็อกไดอะแกรมในส่วนของการ Simulate 42
รูปที่ 3.2 บล็อกไดอะแกรมในส่วนของการ ส่วนของการ Implement ลงบนอุปกรณ์ FPGA 42
รูปที่ 3.3 บล็อกไดอะแกรมของการแปลงเวฟเลต 5 ระดับ 43
รูปที่ 3.4 โครงสร้างภายในของการแปลงเวฟเลต ทางภาคส่ง 44
รูปที่ 3.5 บล็อกไดอะแกรมของการแปลงเวฟเลตกลับ 5 ระดับ 44
รูปที่ 3.6 โครงสร้างภายในของการแปลงเวฟเลตกลับ ทางภาครับ 45
รูปที่ 3.7 บล็อกไดอะแกรมของวงจรหน่วงเวลา 45
รูปที่ 3.8 บล็อกไดอะแกรมของวงจรสุ่มค่าตัวอย่าง 45
รูปที่ 3.9 บล็อกไดอะแกรมของวงจรเพิ่มค่าตัวอย่าง 46
รูปที่ 3.10 บล็อกไดอะแกรมของวงจรบวก 46
รูปที่ 3.11 บล็อกไดอะแกรมของวงจรคูณ 47
รูปที่ 3.12 การแปลงเวฟเลต โดยนำหลักการของ DA มาใช้ 47
รูปที่ 3.13 การแปลงเวฟเลตกลับ โดยนำหลักการของ DA มาใช้ 48
รูปที่ 3.14 บล็อกไดอะแกรมของ วงจร Parallel in Serial out Shift Register 48
รูปที่ 3.15 บล็อกไดอะแกรมของวงจรบวกหรือวงจรลบ 50
รูปที่ 3.16 บล็อกไดอะแกรมของ Accumulator 50
รูปที่ 3.17 แสดงการไหลของข้อมูลในระหว่างกระบวนการ Write 51
รูปที่ 3.18 แสดงเวฟฟอร์มของ Write first mode 51
รูปที่ 3.19 โฟลว์ชาร์ตแสดงการทำงานของการแปลงเวฟเลตในส่วนของการ Simulate
โดยใช้โปรแกรม Xilinx 52
รูปที่ 3.20 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter) 53
รูปที่ 3.21 บล็อกไดอะแกรมของวงจรมัลติเพล็กซ์ 53
รูปที่ 3.22 แสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรม
จากบอร์ด FPGA ไปยังคอมพิวเตอร์ 54
รูปที่ 3.23 แสดงบล็อกไดอะแกรมการรับบิทข้อมูลจากบอร์ด FPGA ผ่านพอร์ตอนุกรม 54
รูปที่ 3.24 โฟลว์ชาร์ตแสดงการทำงานของการแปลงเวฟเลตในส่วนของการ Implement
ลงบนอุปกรณ์ FPGA 55
รูปที่ 4.1 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านวิเคราะห์ 56
รูปที่ 4.2 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 1 57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 2 57
รูปที่ 4.4 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 3 58
รูปที่ 4.5 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 4 59
รูปที่ 4.6 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 5 59
รูปที่ 4.7 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านสังเคราะห์ 60
รูปที่ 4.8 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 1 60
รูปที่ 4.9 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 2 61
รูปที่ 4.10 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 3 62
รูปที่ 4.11 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 4 62
รูปที่ 4.12 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 5 63
รูปที่ 4.13 รูปบล็อกไดอะแกรมในส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator 64
รูปที่ 4.14 แสดงสัญลักษณ์ของวงจรสุ่มค่าตัวอย่าง 64
รูปที่ 4.15 แสดงผลการจำลองการทำงานของวงจรสุ่มค่าตัวอย่าง 64
รูปที่ 4.16 แสดงสัญลักษณ์ของวงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม 64
รูปที่ 4.17 แสดงผลการจำลองการทำงานของวงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม 65
รูปที่ 4.18 แสดงสัญลักษณ์ของหน่วยความจำเฟสคู่ 65
รูปที่ 4.19 แสดงผลการจำลองการทำงานของหน่วยความจำเฟสคู่ 65
รูปที่ 4.20 แสดงสัญลักษณ์ของหน่วยความจำเฟสคี่ 65
รูปที่ 4.21 แสดงผลการจำลองการทำงานของหน่วยความจำเฟสคี่ 66
รูปที่ 4.22 แสดงสัญลักษณ์ของวงจรบวกสะสม 14 บิท 66
รูปที่ 4.23 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 14 บิท 66
รูปที่ 4.24 แสดงสัญลักษณ์ของวงจรบวกขนาด 14 บิท 66
รูปที่ 4.25 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 14 บิท 67
รูปที่ 4.26 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 67
รูปที่ 4.27 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 67
รูปที่ 4.28 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 67
รูปที่ 4.29 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1 68
รูปที่ 4.30 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 69
รูปที่ 4.31 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 69
รูปที่ 4.32 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 69
รูปที่ 4.33 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2 70
รูปที่ 4.34 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 71
รูปที่ 4.35 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 71
รูปที่ 4.36 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.37 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3 72
รูปที่ 4.38 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4 73
รูปที่ 4.39 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4 73
รูปที่ 4.40 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4 73
รูปที่ 4.41 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4 73
รูปที่ 4.42 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5 74
รูปที่ 4.43 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5 74
รูปที่ 4.44 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5 74
รูปที่ 4.45 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5 74
รูปที่ 4.46 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนวิเคราะห์ 75
รูปที่ 4.47 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนวิเคราะห์ 75
รูปที่ 4.48 แสดงสัญลักษณ์ของวงจรแทรกบิท 2 บิท 75
รูปที่ 4.49 แสดงผลการจำลองการทำงานของวงจรแทรกบิท 2 บิท 75
รูปที่ 4.50 แสดงสัญลักษณ์ของวงจรRAM 16 bit 76
รูปที่ 4.51 แสดงผลการจำลองการทำงานของวงจร RAM 76
รูปที่ 4.52 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 1 76
รูปที่ 4.53 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 1 (ช่วงที่ 1) 76
รูปที่ 4.54 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 1 (ช่วงที่ 2) 77
รูปที่ 4.55 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 2 77
รูปที่ 4.56 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 2 (ช่วงที่ 1) 77
รูปที่ 4.57 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 2 (ช่วงที่2) 77
รูปที่ 4.58 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 3 78
รูปที่ 4.59 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 3 78
รูปที่ 4.60 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 4 78
รูปที่ 4.61 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 4 78
รูปที่ 4.62 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 5 79
รูปที่ 4.63 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 5 79
รูปที่ 4.64 แสดงสัญลักษณ์ของวงจรตัดบิท 2 บิท 79
รูปที่ 4.65 แสดงผลการจำลองการทำงานของวงจรตัดบิท 2 บิท 79
รูปที่ 4.66 แสดงสัญลักษณ์ของวงจรROM_LOW 80
รูปที่ 4.67 แสดงผลการจำลองการทำงานของวงจรROM_LOW 80
รูปที่ 4.68 แสดงสัญลักษณ์ของวงจรROM_HIGH 80
รูปที่ 4.69 แสดงผลการจำลองการทำงานของวงจรROM_HIGH 80
รูปที่ 4.70 แสดงสัญลักษณ์ของวงจรเพิ่มค่าตัวอย่าง 81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.71 แสดงผลการจำลองการทำงานของวงจรเพิ่มค่าตัวอย่าง 81
รูปที่ 4.72 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1 81
รูปที่ 4.73 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1 81
รูปที่ 4.74 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1 82
รูปที่ 4.75 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1 82
รูปที่ 4.76 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2 82
รูปที่ 4.77 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2 82
รูปที่ 4.78 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2 83
รูปที่ 4.79 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2 83
รูปที่ 4.80 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3 83
รูปที่ 4.81 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3 84
รูปที่ 4.82 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3 85
รูปที่ 4.83 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3 85
รูปที่ 4.84 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4 85
รูปที่ 4.85 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4 86
รูปที่ 4.86 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4 87
รูปที่ 4.87 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4 87
รูปที่ 4.88 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5 87
รูปที่ 4.89 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5 88
รูปที่ 4.90 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5 89
รูปที่ 4.91 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5 89
รูปที่ 4.92 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนสังเคราะห์ 89
รูปที่ 4.93 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนสังเคราะห์ 89
รูปที่ 4.94 รูปบล็อกไดอะแกรมในส่วนของการ Implement ลงบนอุปกรณ์ FPGA 90
รูปที่ 4.95 แสดงสัญลักษณ์ของวงจรควบคุมการสุ่มค่าตัวอย่าง 90
รูปที่ 4.96 แสดงผลการจำลองการทำงานของวงจรควบคุมการสุ่มค่าตัวอย่าง 90
รูปที่ 4.97 แสดงสัญลักษณ์ของวงจรRAM 8 bit 91
รูปที่ 4.98 แสดงผลการจำลองการทำงานของวงจรRAM 8 bit 91
รูปที่ 4.99 แสดงสัญลักษณ์ของวงจรRAM 16 bit 91
รูปที่ 4.100 แสดงผลการจำลองการทำงานของวงจรRAM 16 bit 92
รูปที่ 4.101 แสดงสัญลักษณ์ของวงจรมัลติเพล็กซ์ 92
รูปที่ 4.102 แสดงผลการจำลองการทำงานของวงจรมัลติเพล็กซ์ 92
รูปที่ 4.103 แสดงสัญลักษณ์ของวงจรหารความถี่ 93
รูปที่ 4.104 แสดงผลการจำลองการทำงานของวงจรหารความถี่ 93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.105 แสดงสัญลักษณ์ของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม 93
รูปที่ 4.106 แสดงผลการจำลองการทำงานของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม 93
รูปที่ 4.107 แสดงสัญลักษณ์ของวงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม 94
รูปที่ 4.108 แสดงผลการจำลองการทำงานของวงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม 94
รูปที่ 4.109 แสดงบล็อกไดอะแกรมของการแปลงเวฟเลตทั้งทางด้านวิเคราะห์และด้านสังเคราะห์ 95
รูปที่ 4.110 แสดงสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตในแต่ละระดับที่ได้
จากการแปลงเวฟเลตบนอุปกรณ์ FPGA 95
รูปที่ 4.111 แสดงสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตในแต่ละระดับที่ได้
จากการแปลงเวฟเลตโดยใช้โปรแกรมMATLAB 96
รูปที่ 4.112 แสดงค่าความผิดพลาดของสัมประสิทธิ์สเกลลิ่ง3 และสัมประสิทธิ์เวฟเลต3
ที่ได้จากการแปลงเวฟเลตบนอุปกรณ์FPGAและใช้โปรแกรมMATLAB 96
รูปที่ 4.113 แสดงค่าความผิดพลาดของสัมประสิทธิ์เวฟเลต2 และสัมประสิทธิ์เวฟเลต1
ที่ได้จากการแปลงเวฟเลตบนอุปกรณ์FPGAและใช้โปรแกรมMATLAB 97
รูปที่ 4.114 แสดงสัมประสิทธิ์สเกลลิ่ง 3 และสัมประสิทธิ์เวฟเลต3ที่ได้จากการแปลงเวฟเลต
ด้านวิเคราะห์บนอุปกรณ์FPGA 97
รูปที่ 4.115 แสดงสัมประสิทธิ์สเกลลิ่ง 2 ที่ได้จากการแปลงเวฟเลตด้านสังเคราะห์ในขั้นที่1
และสัมประสิทธิ์เวฟเลต2 ที่ได้จากการแปลงเวฟเลตด้านวิเคราะห์บนอุปกรณ์FPGA 98
รูปที่ 4.116 แสดงสัมประสิทธิ์สเกลลิ่ง 1 ที่ได้จากการแปลงเวฟเลตด้านสังเคราะห์ในขั้นที่2
และสัมประสิทธิ์เวฟเลต1 ที่ได้จากการแปลงเวฟเลตด้านวิเคราะห์บนอุปกรณ์FPGA 98
รูปที่ 4.117 แสดงข้อมูลสุดท้ายที่ทำการกู้กลับมาได้ 99
รูปที่ 4.118 แสดงการเปรียบเทียบข้อมูลต้นแบบกับข้อมูลที่ทำการกู้กลับมาได้ พร้อมทั้งแสดงค่าMSE 99
รูปที่ 4.119 แสดงบอร์ดเอฟพีจีเอที่ทำการเชื่อมต่อกับพอร์ตอนุกรม 100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงคุณสมบัติที่สำคัญของรูปแบบจำนวนโดยตรง 27

ตารางที่ 2.2 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง 31

ตารางที่ 2.3 แสดงค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุต 33

ตารางที่ 2.4 การเชื่อมต่อของพอร์ตสื่อสารสำหรับคอนเน็คเตอร์แบบ DB-9 40

ตารางที่ 3.1 แสดงตารางค่าสัมประสิทธิ์ของดาร์บีชี 4 ค่า ในส่วนของภาคส่ง 42

ตารางที่ 3.2 แสดงตารางค่าสัมประสิทธิ์ของดาร์บีชี 4 ค่า ในส่วนของภาครับ 43

ตารางที่ 3.3 ตารางค่าความจริงของการออกแบบวงจรบวก 46

ตารางที่ 3.4 แอดเดรสและค่าที่เก็บใน ROM phase EVEN 49

ตารางที่ 3.5 แอดเดรสและค่าที่เก็บใน ROM phase ODD 49

ตารางที่ 3.6 แอดเดรสและค่าที่เก็บใน ROM Low pass 49

ตารางที่ 3.7 แอดเดรสและค่าที่เก็บใน ROM High pass 50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของหัวข้อปริญญานิพนธ์

ปัจจุบันนี้เครือข่ายและการสื่อสารได้มีการขยายการเชื่อมโยงไปทั่วโลกตามกระแสโลกาภิวัฒน์ ปริมาณข่าวสารที่ใช้ในการสื่อสารทั้งถูกส่งไปและรับมา ในเครือข่ายต่างก็เพิ่มขึ้นอย่างรวดเร็ว โดยเฉพาะข้อมูลข่าวสารที่เป็นเสียงพูด ได้มีการใช้งานกันเพิ่มขึ้นอย่างรวดเร็ว

ที่ผ่านมานั้น การส่งผ่านข่าวสารเสียงพูดผ่านเครือข่ายดิจิตอลที่ส่งไปโดยไม่มีการบีบอัด เสียงพูดจะถูกสุ่มด้วยอัตรา 8000 ตัวอย่างต่อวินาที โดยแต่ละตัวอย่างมีขนาด 8 บิท (Bit) นั่นคือบิทเรต (Bit Rate) ที่ต้องส่งไปในเครือข่ายเป็น 64000 bps ซึ่งในความเป็นจริงแล้วถ้าเราบีบอัดข้อมูลเสียงลงไปได้ เราจะสามารถส่งข่าวสารผ่านเครือข่ายได้มากขึ้นในเวลาเท่ากัน ทั้งๆที่ความจุของเครือข่ายการสื่อสารมีเท่าเดิม สำหรับในระบบสื่อสารเช่น โทรศัพท์ ซึ่งระยะหลังนี้ ได้มีการพัฒนาระบบดิจิตอลโดยเพิ่มการบีบอัดข้อมูลส่งไปด้วย ดังนั้นจึงได้มีผู้พัฒนาการบีบอัดข้อมูลเสียงด้วยวิธีต่างๆกัน และในช่วงเวลา 4-5 ปีที่ผ่านมาได้มีการสนใจแนวคิดของเวฟเลต (Wavelet) กันอย่างกว้างขวาง เนื่องจากแนวคิดนี้สามารถนำไปใช้งานได้ในหลายๆสาขา ทั้งด้านคณิตศาสตร์ ฟิสิกส์ และงานวิศวกรรม ซึ่งก็ได้มีการนำมาใช้ในการประยุกต์ใช้ในการประมวลผลสัญญาณเชิงดิจิตอลเป็นจำนวนมาก ทั้งเกี่ยวกับ ข้อมูลเสียง, ข้อมูลภาพและสัญญาณต่างๆ

สำหรับการบีบอัดข้อมูลเสียง ได้มีผู้เสนอการใช้เวฟเลต โดยมีวิธี และหลักการที่แตกต่างกันออกไป สำหรับปริญญานิพนธ์เล่มนี้ได้ศึกษาถึง การบีบอัดข้อมูลเสียงโดยวิธีการแปลงเวฟเลต โดยการใช้วงจรกรองความถี่ที่มีโครงสร้างแบบโพลีเฟส (Poly-phase Structure Filter) และโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) มาประยุกต์ใช้กับวิธีการแปลงเวฟเลต เพื่อให้ง่ายต่อการประมวลผลและลดจำนวนอุปกรณ์ในการนำไปใช้งาน

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1.2.1 เพื่อศึกษาหลักการบีบอัดข้อมูลเสียงโดยวิธีการแปลงเวฟเลต (Wavelet Transform)

1.2.2 นำหลักการของวงจรกรองความถี่ที่มีโครงสร้างแบบโพลีเฟส (Poly-phase Structure Filter) และโครงสร้างเลขคณิตกระจาย มาประยุกต์ใช้กับวิธีการแปลงเวฟเลต

1.2.3 ทำการจำลองการทำงานของวิธีการแปลงเวฟเลต ด้วยการออกแบบวงจรดิจิตอลภาษา VHDL และเก็บผลการทดลองจากโปรแกรม ModelSim XE II 5.7c

1.2.4 นำผลการทดลองที่ได้จากโปรแกรม ModelSim XE II 5.7c มาเปรียบเทียบกับผลการทดลองจากโปรแกรมแมทแลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของปริญญานิพนธ์

ปริญญานิพนธ์นี้เป็นการนำวงจรกรองความถี่ที่มีโครงสร้างแบบโพลีเฟส และโครงสร้างเลขคณิตกระจาย มาประยุกต์ใช้กับวิธีการบีบอัดข้อมูลเสียงโดยวิธีการแปลงเวฟเลต และได้ทำการจำลองการทำงานผ่านโปรแกรม Xilinx-Project Navigator ซึ่งกระบวนการทำงานทั้งหมดจะถูกบรรยายพฤติกรรมการทำงานโดยใช้การเขียนด้วยภาษา VHDL ในการออกแบบและทำการจำลองผลการทดลองออกมา และจะนำไปเปรียบเทียบกับผลที่ได้จากโปรแกรมแมทแลบ

1.4 เนื้อหาของปริญญานิพนธ์

ในบทที่ 2 เป็นส่วนของทฤษฎีพื้นฐานการแปลงเวฟเลต, ทฤษฎีพื้นฐานของวงจรกรองความถี่และความสัมพันธ์ของทฤษฎีทั้งสอง, โครงสร้างของวงจรกรองความถี่แบบโพลีเฟส, การประยุกต์ใช้ของวิธีการแปลงเวฟเลตควบคู่ไปกับโครงสร้างของวงจรกรองความถี่แบบโพลีเฟส, ทฤษฎีของภาษา VHDL, ทฤษฎีของอุปกรณ์ FPGA, ทฤษฎีของ DA (Distributed Arithmetic) รวมทั้งส่วนของการสื่อสารกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม (Serial Port) เพื่อทำการรับ – ส่งข้อมูล

ในบทที่ 3 เป็นส่วนของการคำนวณและการสร้าง โดยจะแบ่งเป็นส่วนของการ Simulate โดยใช้โปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL ในการออกแบบ และส่วนของการ Implement ลงบนอุปกรณ์ FPGA รวมทั้งวิธีการออกแบบการรับ-ส่งข้อมูลผ่านทางพอร์ตอนุกรม

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

ในบทที่ 5 เป็นของ บทวิจารณ์และบทสรุป

กิตติกรรมประกาศ

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

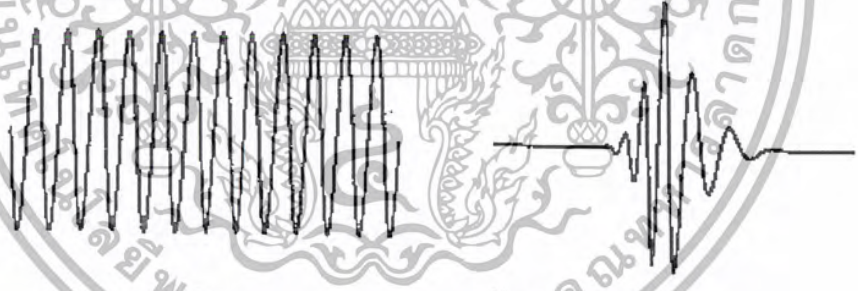
บทที่ 2
ทฤษฎีและหลักการ

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 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เวฟเลตสามารถปรับตัวให้เข้ากับระบบหรือสัญญาณได้หลายรูปแบบ จึงสามารถนำไปประยุกต์ใช้งานได้หลายอย่าง
4. การคำนวณและการแปลงเวฟเลตเหมาะแก่การใช้งานในระบบดิจิตอลและคอมพิวเตอร์ เนื่องจากมีเพียงการคูณและการบวกเท่านั้น

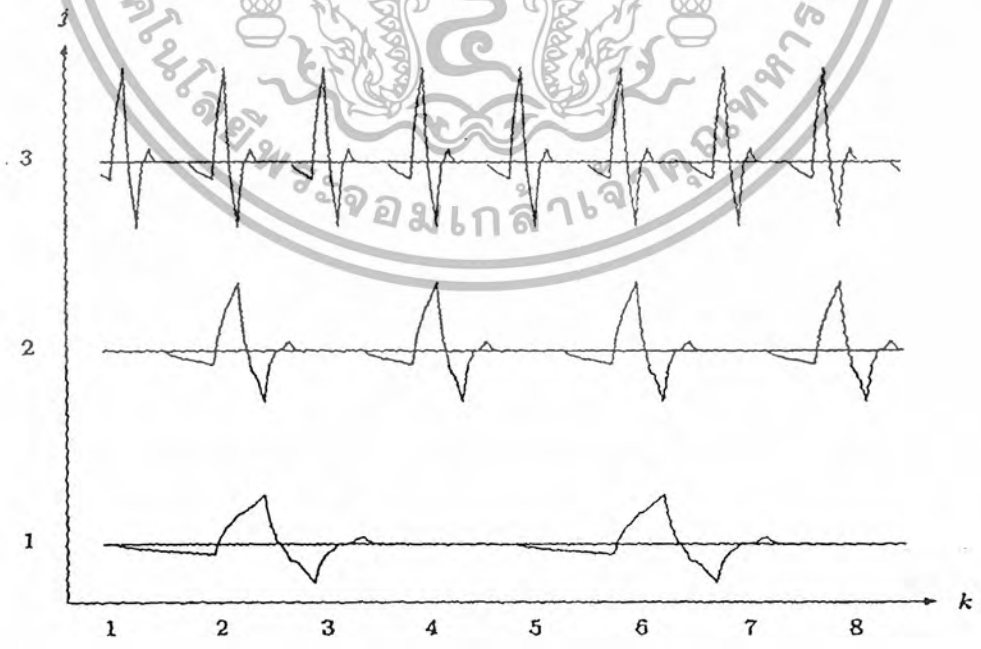
2.2 การแปลงเวฟเลตแบบไม่ต่อเนื่อง (The Discrete Wavelet Transform)

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 และการลดสัญญาณรบกวน



รูปที่ 2.2 การเลื่อนตำแหน่งและการปรับระดับของเวฟเลต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ทฤษฎีมัลติรีโซลูชัน (Multiresolution)

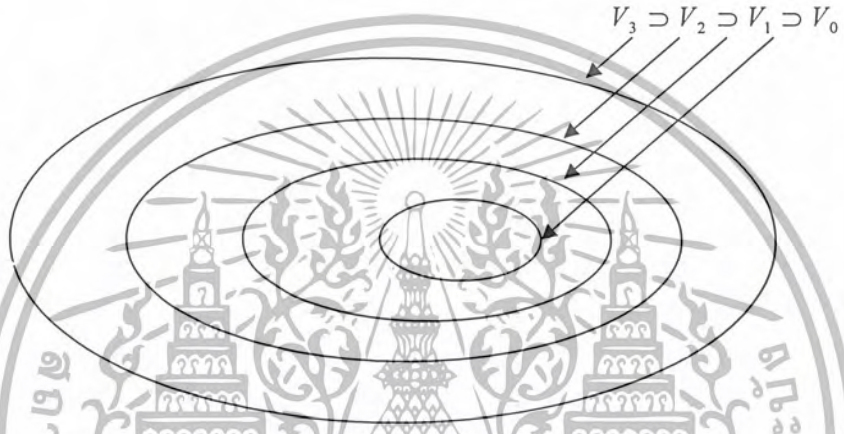
การวิเคราะห์โดยการแบ่งย่อยออกเป็นหลายส่วน สามารถอธิบายให้อยู่ในรูปของพื้นที่ย่อยๆที่เป็นสับเซตของพื้นที่ใหญ่ได้ดังนี้ โดยที่ L^2 จะแทนพื้นที่ทั้งหมดของสัญญาณ

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

หรือ
$$V_j \subset V_{j+1} \tag{2.5}$$

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

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



รูปที่ 2.3 การแบ่งพื้นที่ออกเป็นส่วนย่อยโดยการใช้สเกลลิ่งฟังก์ชัน

จากที่กล่าวมาข้างต้นเราสามารถหาค่าของสเกลลิ่งฟังก์ชันได้จาก

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

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

2.3.1 สเกลลิ่งฟังก์ชัน (The Scaling Function)

เป็นวิธีการหนึ่งในวิธีมัลติรีโซลูชัน โดยการแบ่งสัญญาณออกเป็นส่วนย่อย หรือเป็นระดับแล้วจึงอธิบายเวฟเลตในเทอมของสเกลลิ่งฟังก์ชัน ดังสมการที่ 2.9

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

เราสามารถเพิ่มขนาดของส่วนย่อยนี้โดยการเปลี่ยนค่าระดับในสเกลลิ่งฟังก์ชัน โดยสามารถแสดงให้อยู่ในสองมิติ คือ การสเกลลิ่งและการทรานสเลชัน ได้ดังสมการ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

โดยที่ ถ้า $j\rangle 0$ จะทำให้มีการแบ่งมากขึ้น ทำให้ $\varphi_{j,k}(t)$ มีขนาดแคบหรือเล็กลง และมีการเลื่อนตำแหน่งที่น้อยลง ดังนั้นจะได้รายละเอียดดีขึ้น ในทางตรงกันข้ามถ้า $j\langle 0$ ทำให้ $\varphi_{j,k}(t)$ จะขนาดกว้างขึ้น และเลื่อนตำแหน่งมากขึ้น 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 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ฟิลเตอร์แบงก์และการแปลงเวฟเลต (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 $t=2^j t-k$

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

แทนค่า $m=2k+n$

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

ถ้าเราแทนพจน์ที่เกิดจากการกระจายด้วย V_j

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

ดังนั้น จะได้ว่า

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

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจากความสัมพันธ์สามารถหาค่าสัมประสิทธิ์ของเวฟเลตได้ดังนี้

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

2.4.1.1 การกรองสัญญาณและการสุ่มค่าตัวอย่าง (Filtering and Down-Sampling or Decimating)

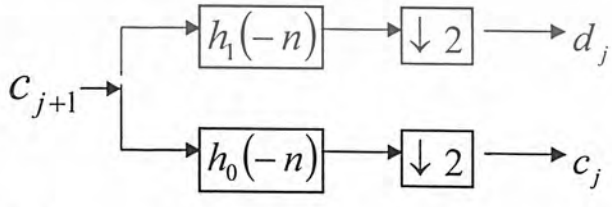
จากหลักการของการประมวลผลสัญญาณดิจิตอล การกรองสัญญาณสามารถทำได้โดยการทำคอนโวลูชันระหว่างสัญญาณอินพุตกับสัมประสิทธิ์ของวงจรกรองความถี่ (Filter coefficients or Impulse response) โดยถ้าสมมติให้อินพุต คือ $x(n)$ และสัมประสิทธิ์ของวงจรกรองความถี่คือ $h(n)$ จะได้เอาต์พุตคือ $y(n)$ ดังสมการ

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

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

รูปที่ 2.5 ตัวสุ่มค่าตัวอย่าง (Down Sampler or Decimator)

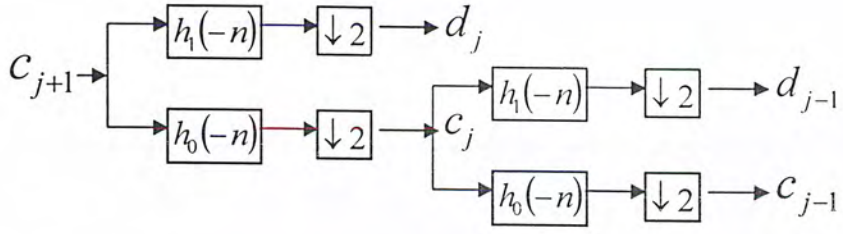
จากสมการที่ (2.22) และ (2.23) จะเห็นได้ว่าสัมประสิทธิ์ของสเกลลิ่งและเวฟเลตในระดับที่ต่างกันเกิดจากการทำคอนโวลูชันระหว่างสัมประสิทธิ์การกระจายในระดับ j กับสัมประสิทธิ์ของวงจรกรองความถี่ $h_0(-n)$ และ $h_1(-n)$ จากนั้นจึงนำมาสุ่มค่าตัวอย่าง จึงจะได้สัมประสิทธิ์การกระจายในลำดับถัดไป คือ $j-1$ หรืออาจกล่าวได้ว่า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



รูปที่ 2.6 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาคส่ง

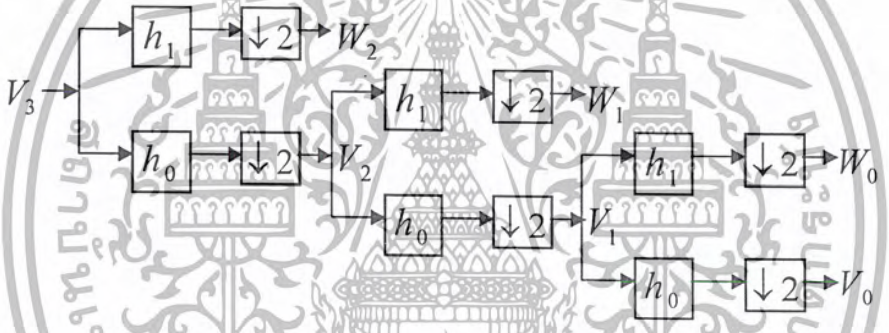
การแตกออก การกรองสัญญาณและการสุ่มค่าตัวอย่างจะทำซ้ำในส่วนของสัมประสิทธิ์สเกลลิ่ง เพื่อให้ได้โครงสร้างตามรูปที่ 2.7 ซึ่งเราเรียกกว่า Iterating the filter bank

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

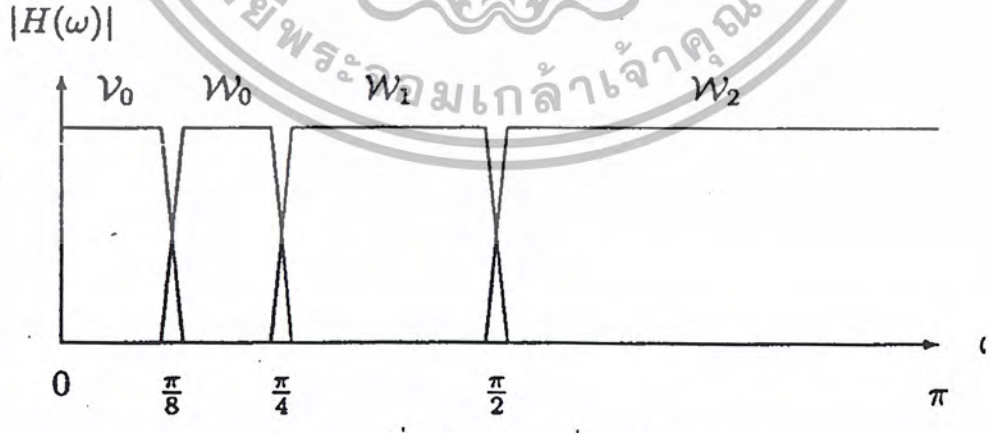


รูปที่ 2.7 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสองขั้นของภาคส่ง

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



รูปที่ 2.8 แสดงการแตกกิ่งก้านสาขาแบบสองแถบสามขั้นของภาคส่ง



รูปที่ 2.9 แถบความถี่ของภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ในการที่จะนำสัญญาณเดิมกลับคืนมาสามารถทำได้โดยการรวมกันของสัมประสิทธิ์สเกลลิ่งฟังก์ชันและเวฟเลตในระดับที่มีความละเอียดน้อยกว่า โดยสามารถเขียนเป็นสมการที่อยู่ในรูปของสเกลลิ่งฟังก์ชันในระดับ $j+1$ ได้ดังนี้

$$f(t)=\sum_k c_{j+1}(k)2^{(j+1)/2}\varphi\left(2^{j+1}t-k\right) \tag{2.25}$$

หรืออาจจะเขียนในรูปของระดับถัดไป ซึ่งจะมีส่วนของเวฟเลตด้วยดังนี้

$$f(t)=\sum_k c_j(k)2^{j/2}\varphi\left(2^j t-k\right)+\sum_k d_j(k)2^{j/2}\psi\left(2^j t-k\right) \tag{2.26}$$

เมื่อแทนค่าสมการที่ (2.17) และ (2.15) ลงในสมการที่ (2.26) จะได้

$$f(t)=\sum_k c_j(k)\sum_n h(n)2^{(j+1)/2}\varphi\left(2^{j+1}t-2k-n\right)+\sum_k d_j(k)\sum_n h_1(n)2^{(j+1)/2}\varphi\left(2^{j+1}t-2k-n\right) \tag{2.27}$$

เนื่องจากฟังก์ชันทั้งหมดนี้เป็นฟังก์ชันเชิงตั้งฉาก จึงทำการคูณสมการที่ (2.25) และ (2.27) ด้วย $\varphi\left(2^{j+1}t-k\right)$ และทำการอินทิเกรตหาค่าสัมประสิทธิ์ได้ดังนี้

$$c_{j+1}(k)=\sum_m c_j(m)h(k-2m)+\sum_m d_j(m)h_1(k-2m) \tag{2.28}$$

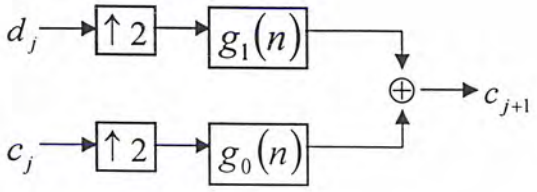
2.4.2.1การกรองสัญญาณและการเพิ่มค่าตัวอย่าง (Filtering and Up-Sampling or Stretching)

ในส่วนของฟิลเตอร์แบงค์ด้านสังเคราะห์จะประกอบด้วยการเพิ่มค่าตัวอย่างและการกรองสัญญาณ ซึ่งหมายความว่าอินพุตของฟิลเตอร์จะมีศูนย์แทรกอยู่ระหว่างสัญญาณเดิม ดังสมการ

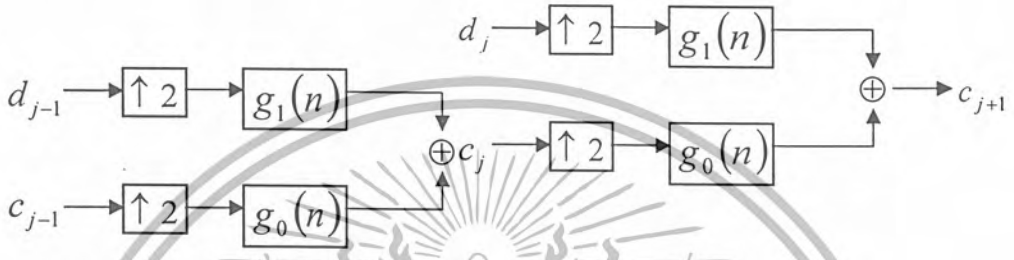
$$y(2n)=x(n) \qquad y(2n+1)=0 \tag{2.29}$$

ทำให้สัญญาณอินพุตถูกขยายออกเป็นสองเท่าของสัญญาณเดิม และค่าศูนย์ก็ถูกแทรกเข้ามา และจากสมการที่ (2.28) ซึ่งเกิดจากการเพิ่มค่าตัวอย่างในระดับ j จากนั้นจึงทำการคอนโวลูชันกับสัมประสิทธิ์ของวงจรกรองความถี่ $g(n)$ ทั้งในส่วนของสเกลลิ่งฟังก์ชันและเวฟเลต แล้วนำมาบวกกันได้เป็นสัมประสิทธิ์ในระดับ $j+1$ ดังรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



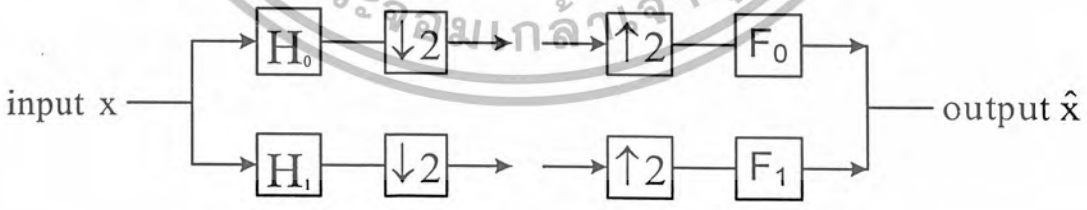
รูปที่ 2.10 แสดงการแตกกิ่งก้านสาขาแบบสองแถบของภาครับ



รูปที่ 2.11 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

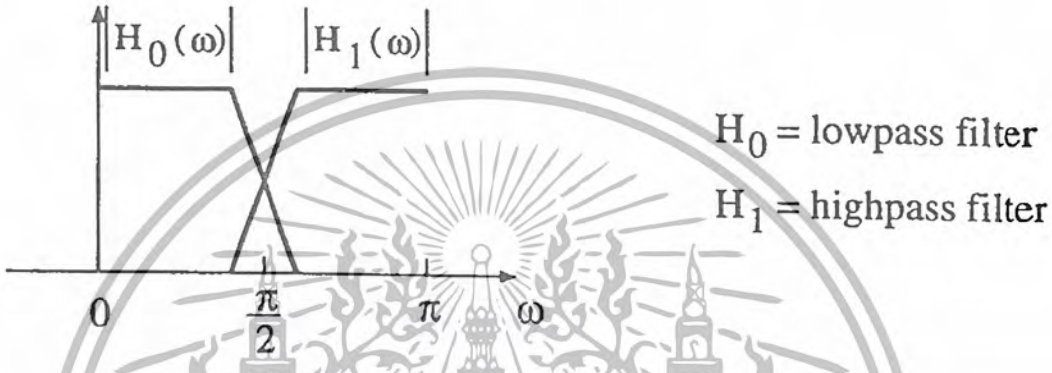


รูปที่ 2.12 ฟิลเตอร์แบงค์

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 2.13 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

$$F_0(z)H_0(z) + F_1(z)H_1(z) = z^{-1} \tag{2.30}$$

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

$$(\downarrow 2)(\uparrow 2)H_0x = \frac{1}{2}[H_0(z)X(z) + H_0(-z)X(-z)] \tag{2.31}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

$$F_0(z)H_0(z)+F_1(-z)H_1(-z)=0 \tag{2.32}$$

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 ซึ่งจะเป็นการกำจัดการลดทอน (No distortion)

$$F_0(z)H_0(z)+F_1(z)H_1(z)=2z^{-l} \tag{2.33}$$

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_0ถ้าอธิบายในเชิง Z-domain เมื่อผ่านการสุ่มค่าตัวอย่างและเพิ่มค่าตัวอย่างจะได้

$$(\downarrow 2) \qquad \frac{1}{2}\left[H_0\left(z^{\frac{1}{2}}\right)X\left(z^{\frac{1}{2}}\right)+H_0\left(-z^{\frac{1}{2}}\right)X\left(-z^{\frac{1}{2}}\right)\right] \tag{2.34}$$

$$(\uparrow 2) \qquad \frac{1}{2}[H_0(z)X(z)+H_0(-z)X(-z)] \tag{2.35}$$

เมื่อผ่านวงจรกรองความถี่จะทำการคูณด้วย $F_0(z)$ซึ่งจะทำให้ผลที่ออกจากวงจรกรองความถี่ต่ำและสูง ดังนี้

วงจรกรองความถี่ต่ำ $$\frac{1}{2}F_0(z)[H_0(z)X(z)+H_0(-z)X(-z)] \tag{2.36}$$

วงจรกรองความถี่สูง $$\frac{1}{2}F_1(z)[H_1(z)X(z)+H_1(-z)X(-z)] \tag{2.37}$$

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

$$\frac{1}{2}[F_0(z)H_0(z)+F_1(z)H_1(z)]X(z)+\frac{1}{2}[F_0(-z)H_0(-z)+F_1(-z)H_1(-z)]X(-z) \tag{2.38}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

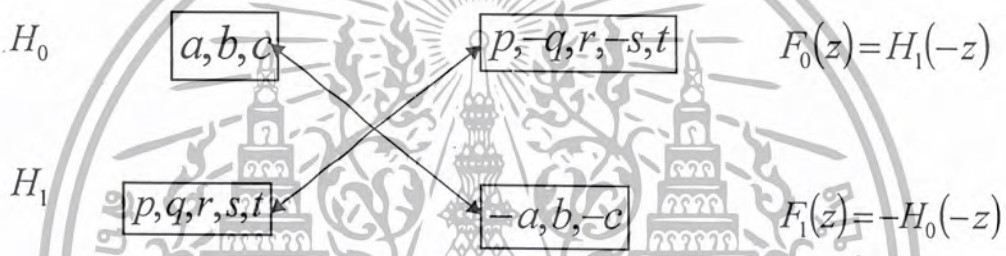
ในการกู้กลับอย่างสมบูรณ์ที่มีการหน่วงเวลา l ครั้ง $\hat{X}(z)$ จะต้องเปลี่ยนเป็น $z^{-l}X(z)$ ดังนั้น เทอมของการลดทอนจะต้องเป็น z^{-l} และเทอมของการซ้อนทับกันของสเปกตรัมจะต้องมีค่าเท่ากับศูนย์

2.5.1 การกำจัดการเกิดการซ้อนทับกันและผลที่ได้จากวงจรกรองความถี่ $P_0 = F_0 H_0$

ในตอนนี้เราจะต้องออกแบบวงจรกรองความถี่สี่ตัว H_0, H_1, F_0, F_1 โดยอาศัยเงื่อนไขที่มีความเกี่ยวข้องกันระหว่างวงจรกรองความถี่ฝั่งส่งและฝั่งรับ ดังนี้

$$F_0(z) = H_1(-z) \text{ และ } F_1(z) = -H_0(-z) \tag{2.39}$$

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



รูปที่ 2.14 แสดงการสลับเครื่องหมายในการหาค่าสัมประสิทธิ์

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

$$P_1(z) = -H_0(-z)H_1(z) = -H_0(-z)F_0(-z) = -P_0(-z) \tag{2.40}$$

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

$$F_0(z)H_0(z) - F_0(-z)H_0(-z) = P_0(z) - P_0(-z) = 2z^{-l} \tag{2.41}$$

หลักการออกแบบวงจรกรองความถี่ทั้งภาคส่งและภาครับสามารถลดได้เหลือ 2 ขั้นตอน ดังนี้

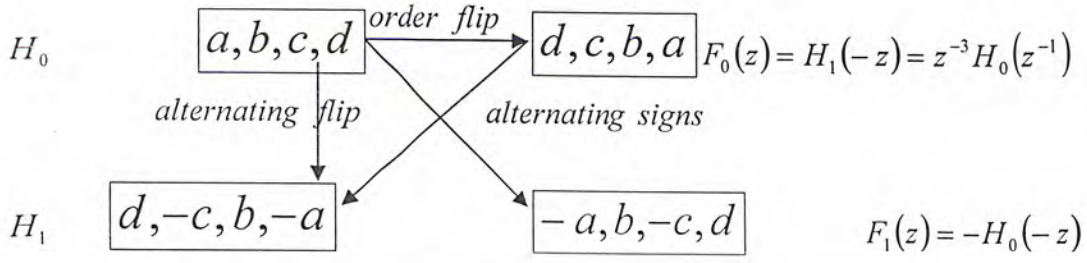
ขั้นแรก : ออกแบบวงจรกรองความถี่ต่ำให้ได้ผลตามสมการ (2.41)

ขั้นสอง : แสดงให้เห็นว่า P_0 เกิดจากผลคูณของ F_0H_0

จากนั้นใช้สมการ (2.39) เพื่อหาค่า F_1 และ H_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบฟิลเตอร์แบงค์ทั้งภาครับและภาคส่งให้ตรงตามหลักการการกู้กลับอย่างสมบูรณ์ จะอยู่ภายใต้พื้นฐานของฟังก์ชันเชิงตั้งฉาก ดังนี้



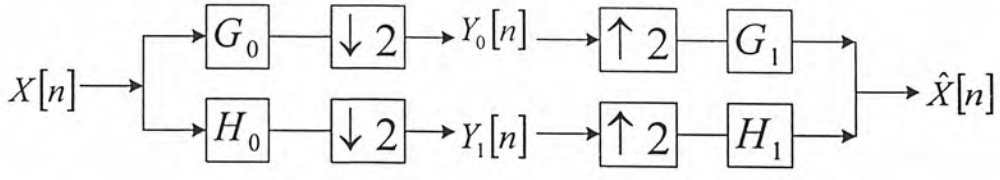
รูปที่ 2.15 แสดงความสัมพันธ์ระหว่างสัมประสิทธิ์ทั้งภาครับและภาคส่ง

รูปที่ 2.16 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 ตามด้วยวงจรกรองความถี่ต่ำผ่านและวงจรกรองความถี่สูงผ่าน โดยสามารถเขียนโครงสร้างโดยตรง ดังบล็อกไดอะแกรมดังรูปที่ 2.17



รูปที่ 2.17 บล็อกไดอะแกรมของโครงสร้างโดยตรงของวงจรกรองความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

จากรูปบล็อกไดอะแกรม ในส่วนวิเคราะห์ $X(n)$ คือสัญญาณอินพุต, G_0 และ H_0 เป็นวงจรกรองความถี่ต่ำผ่าน และความถี่สูงผ่านตามลำดับ $\downarrow 2$ หมายถึงการสุ่มค่าตัวอย่างลดลงครึ่งหนึ่งและ $\uparrow 2$ หมายถึง การเพิ่มค่าตัวอย่างเป็น 2 เท่า เช่นกัน อินพุตนั้น จะถูกกรองด้วยวงจรกรองทั้งสองแบนด์ตามด้วย การสุ่มค่าตัวอย่างลดลงครึ่งหนึ่ง หลังจากนั้นจะได้ เอาต์พุตเป็น Y_0 และ Y_1 ซึ่งแสดงให้เห็นดังสมการที่ 2.42 และ 2.43

$$Y_0[k] = \sum_n X[n] G_0[2k-n] \tag{2.42}$$

$$Y_1[k] = \sum_n X[n] H_0[2k-n] \tag{2.43}$$

เอาต์พุตของวงจรนั้นจะสามารถกู้กลับมาโดยในส่วนสังเคราะห์ของวงจรกรอง โดยที่ Y_0 และ Y_1 จะถูก เพิ่มค่าตัวอย่างเป็น 2 เท่า แล้วผ่านวงจรกรองจะทำให้ได้สัญญาณเดิมกลับคืนมา

2.5.2.2 วงจรกรองความถี่ที่มีโครงสร้างแบบโพลีเฟส (Polyphase Structure Filters)

ตัวกรองความถี่แบบเอฟไออาร์ (FIR: Finite Impulse Response) ที่แสดงโดยการกระจายฟังก์ชันการส่งผ่าน โพลีเฟส และมีผลลัพธ์เป็น โครงสร้างแบบขนาน ซึ่งเมื่อพิจารณา วิธีการนี้จากฟังก์ชัน การส่งผ่าน H_z ที่มี 9 พจน์ดังสมการที่ 2.44

$$\begin{aligned} H_z = {} & h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[4]z^{-4} \\ & + h[5]z^{-5} + h[6]z^{-6} + h[7]z^{-7} + h[8]z^{-8} \end{aligned} \tag{2.44}$$

จากสมการที่ 1 สามารถแบ่งเป็น 2 เทอม คือ เทอมที่มีสัมประสิทธิ์เป็นพจน์กำลังคู่ และ เทอมที่มีสัมประสิทธิ์เป็นพจน์กำลังคี่ดังสมการที่ 2.45 และ 2.46

$$\begin{aligned} H(z) = {} & \left(h[0] + h[2]z^{-2} + h[4]z^{-4} + h[6]z^{-6} + h[8]z^{-8}\right) + \\ & \left(h[1]z^{-1} + h[3]z^{-3} + h[5]z^{-5} + h[7]z^{-7}\right) \end{aligned} \tag{2.45}$$

$$\begin{aligned} H(z) = {} & \left(h[0] + h[2]z^{-2} + h[4]z^{-4} + h[6]z^{-6} + h[8]z^{-8}\right) + \\ & z^{-1}\left(h[1] + h[3]z^{-2} + h[5]z^{-4} + h[7]z^{-6}\right) \end{aligned} \tag{2.46}$$

เมื่อกำหนดให้

$$E_0(z) = h[0] + h[2]z^{-1} + h[4]z^{-2} + h[6]z^{-3} + h[8]z^{-4} \quad \text{และ}$$

$$E_1(z) = h[1] + h[3]z^{-1} + h[5]z^{-2} + h[7]z^{-3} \tag{2.47}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
61814

จากสมการที่ 2.47 ทำให้จัดรูปแบบสมการที่ 2.46 ใหม่ ดังสมการที่ 2.48

$$H(z) = E_0(z^2) + z^{-1}E_1(z^2) \tag{2.48}$$

จากการกระจายของฟังก์ชัน สมการที่ 2.48 ซึ่งเป็นที่รู้กันโดยทั่วไปว่า เป็นการกระจายโพลีเฟส (Polyphase Decomposition) และสำหรับการกระจายโพลีเฟสที่มี L สาขาและมีฟังก์ชันการส่งผ่าน N อันดับ มีรูปแบบดังสมการที่ 2.49

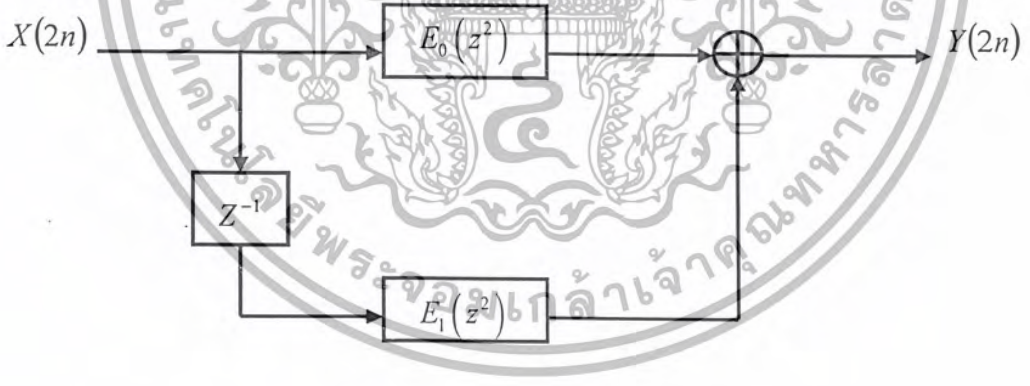
$$H(z) = \sum_{m=0}^{L-1} z^{-m} E_m(z^L) \tag{2.49}$$

โดยที่

$$E_m(z) = \sum_{n=0}^{[(N+1)/L]} h[Ln+m]z^{-n}, \quad 0 \le m \le L-1,$$

$$h[n] = 0 \quad for \quad n > N \tag{2.50}$$

จากวิธีการกระจาย $H(z)$ ดังสมการที่ 2.49 เรียกว่า การแสดงให้จริงของโพลีเฟส (Polyphase Realization) และจากสมการสามารถเขียนอยู่ในรูปของบล็อกไดอะแกรมดังรูปที่ 2.18



รูปที่ 2.18 บล็อกไดอะแกรม การแสดงให้เห็นจริงของโพลีเฟส ของฟังก์ชันการส่งผ่าน

โครงสร้างรูปแบบโพลีเฟส จะมีข้อได้เปรียบ จากรูปแบบโดยตรง สังเกตได้จากในส่วนวิเคราะห์ของฟิลเตอร์แบงค์ของรูปแบบโดยตรง เมื่ออินพุทผ่านการกรองจะประกอบไปด้วย N แซมเปิลเมื่อทำการสุ่มค่าตัวอย่างลดลงครึ่งหนึ่ง แล้ว เราจะใช้เพียง N/2 เท่านั้น ส่วนที่เหลืออีก N/2 เป็นส่วนที่ไม่มีประโยชน์ และจะเห็นได้ว่า เอาต์พุตที่ผ่านวงจรกรองความถี่ต่ำผ่าน และผ่านการสุ่มค่าตัวอย่างแล้วจะประกอบด้วย จำนวนแซมเปิลของ เฟสคู่ของอินพุทเวคเตอร์(X_{even})ทำการคอนโวลูท กับ สัมประสิทธิ์เฟสคู่ของวงจรกรองความถี่ต่ำผ่าน(G_{even}) และ จำนวนแซมเปิลของ เฟสคี่ของอินพุทเวคเตอร์(X_{odd})ทำการ คอน

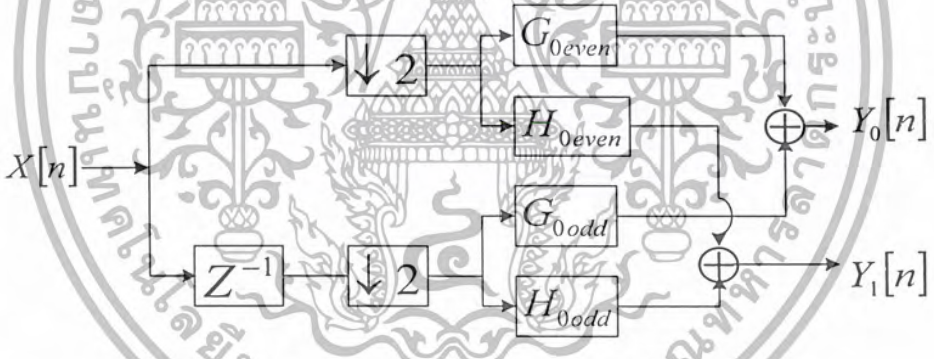
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โวลูท กับ สัมประสิทธิ์เฟสคี่ของวงจรกรองความถี่ต่ำผ่าน(G_{0odd}) สำหรับรูปแบบโพลีเฟส สัญญาณอินพุทจะถูกแบ่งเป็น จำนวนแซมเปิล คู่และคี่ โดยอัตโนมัติเมื่อทำการสุ่มค่าตัวอย่างลดลงครึ่งหนึ่งในทำนองเดียวกัน สัมประสิทธิ์ของวงจรกรองความถี่ ก็จะถูกแบ่งเป็นส่วน คู่และคี่ เช่นกัน ดังนั้น X_{even} ทำการคอนโวลูทกับ G_{0even} และ X_{odd} ก็จะทำการคอนโวลูทกับ G_{0odd} ซึ่งเมื่อนำ 2 เฟสมาบวกกัน ทำให้ได้เอาต์พุตของความถี่ต่ำ ในทางเดียวกัน สำหรับวงจรกรองความถี่สูงผ่านโดย วงจรกรองความถี่สูงผ่าน จะแบ่งเป็นเฟส คู่และคี่, สัมประสิทธิ์ของวงจรกรองความถี่ H_{0even} และ H_{0odd}

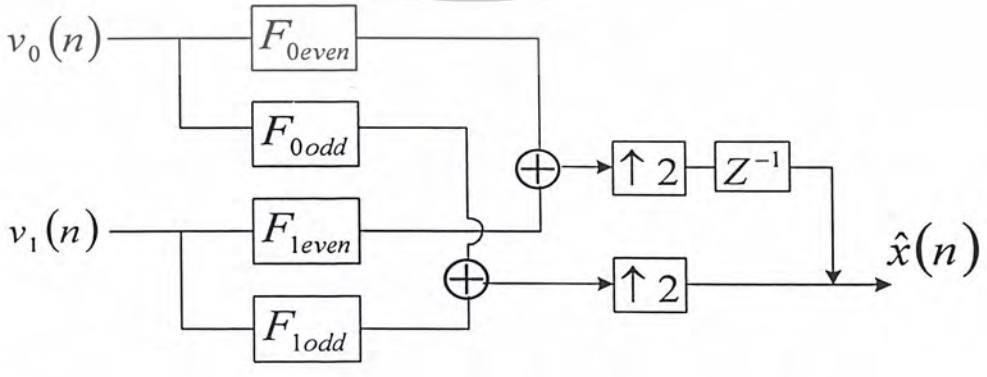
สมการเมตริกซ์ในส่วนวิเคราะห์ของโพลีเฟสเป็นไปตามสมการที่ 2.51

$$\begin{bmatrix} G_{0even} G_{0odd} \\ H_{0even} H_{0odd} \end{bmatrix} \times \begin{bmatrix} X_{even} \\ z^{-1} X_{odd} \end{bmatrix} = H_p \begin{bmatrix} X_{even} \\ z^{-1} X_{odd} \end{bmatrix} = \begin{bmatrix} Y_0 \\ Y_1 \end{bmatrix} \tag{2.51}$$

เนื่องจากว่า เทอมคู่และเทอมคี่ ถูกแบ่งให้เท่ากันแล้วนำไปกรองด้วย สัมประสิทธิ์ของวงจรกรองความถี่ ที่เป็นคู่และคี่ อย่างละครึ่งเช่นกัน ทำให้วงจรกรองความถี่ สามารถทำการปรับปรุงประสิทธิภาพของวงจรในทางขนาน ซึ่งแสดงให้เห็นดังรูปที่ 2.19 และ 2.20



รูปที่ 2.19 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนวิเคราะห์



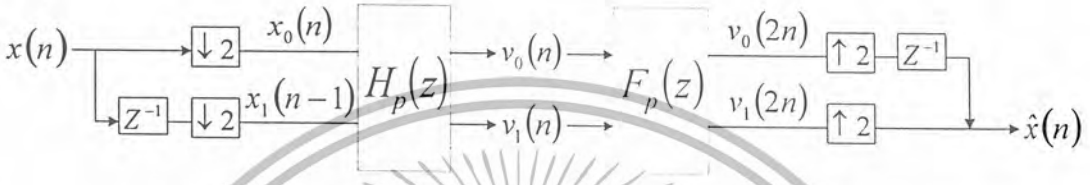
รูปที่ 2.20 โครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนสังเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่2.20 ซึ่งเป็นโครงสร้างของฟิลเตอร์แบบโพลีเฟสในส่วนสังเคราะห์ สามารถเขียนให้อยู่ในรูปของสมการ ดังนี้

$$F(z) = F_1(z^2) + z^{-1}F_0(z^2) \tag{2.52}$$

$$\hat{X}(z) = \begin{bmatrix} z^{-1} & 1 \end{bmatrix} \begin{bmatrix} F_{0,0}(z^2) & F_{1,0}(z^2) \\ F_{0,1}(z^2) & F_{1,1}(z^2) \end{bmatrix} \begin{bmatrix} V_0(z^2) \\ V_1(z^2) \end{bmatrix} \tag{2.53}$$



รูปที่ 2.21 แสดงรูปแบบส่วนวิเคราะห์และส่วนสังเคราะห์ ของฟิลเตอร์แบงค์

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

2.6 ภาษาวีเอชดีแอล

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

2.6.1 ประวัติความเป็นมาของภาษาวีเอชดีแอล

วีเอชดีแอล ย่อมาจากคำว่า VHSIC Hardware Description Language (VHSIC: Very High Speed Integrated Circuit) เป็นภาษาโปรแกรมระดับสูง (High Level Language) ที่ใช้สำหรับการออกแบบฮาร์ดแวร์ในระบบดิจิตอล ตัวของภาษาสามารถบรรยายพฤติกรรมการทำงานในรูปของลำดับชั้น (Hierarchy) และสามารถเขียนได้หลายรูปแบบด้วยเหตุผลนี้จึงทำให้ภาษาวีเอชดีแอล เป็นเครื่องมือที่ใช้ออกแบบตั้งแต่ขั้นตอนบนสุด คือ แนวความคิดที่จะแก้ปัญหา ลงไปทีละขั้นจนถึงขั้นตอนของการสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรจริง และตัวภาษาก็เปิดโอกาสให้วิศวกรได้พัฒนาและจำลองการทำงานของรูปแบบฟังก์ชั่นการทำงานของวงจรอย่างสังเขป โดยไม่คำนึงถึงรายละเอียดเกี่ยวกับโครงสร้างวงจรจริง นอกจากนี้วีเอชดีแอลยังเป็นภาษาที่สนับสนุนลักษณะต่างๆ ของระบบดิจิตอลที่มีความซับซ้อนได้ทั้งหมด ดังนั้น วีเอชดีแอลจึงเป็นภาษาที่น่าสนใจในการศึกษาและนำไปใช้งานเป็นอย่างยิ่ง สำหรับมาตรฐานของภาษาที่ใช้บรรยายพฤติกรรมวงจร หรือฮาร์ดแวร์ สามารถสรุปได้ดังนี้

- ต้องเป็นภาษาที่นำไปเขียนรูปแบบระบบดิจิตอลและมีคุณสมบัติที่สามารถเข้าใจได้ทั้งมนุษย์และเครื่องคอมพิวเตอร์โดยไม่ต้องมีการแปลหรือเปลี่ยนแปลงอีก
- สามารถนำไปใช้เป็นเอกสารประกอบโครงการได้
- ต้องเป็นภาษาที่เขียนขึ้นสำหรับใช้จำลองการทำงานของวงจร

ฉะนั้นภาษาดังกล่าวนี้จึงจัดเป็นภาษาโปรแกรมระดับสูง เช่น เดียวกับภาษาปาสคาล หรือภาษาซี ซึ่งในทางวิศวกรรมภาษาที่ใช้ในการออกแบบฮาร์ดแวร์นี้เรียกว่า ภาษาโปรแกรมระดับสูง

2.6.2 การออกแบบระบบดิจิตอล

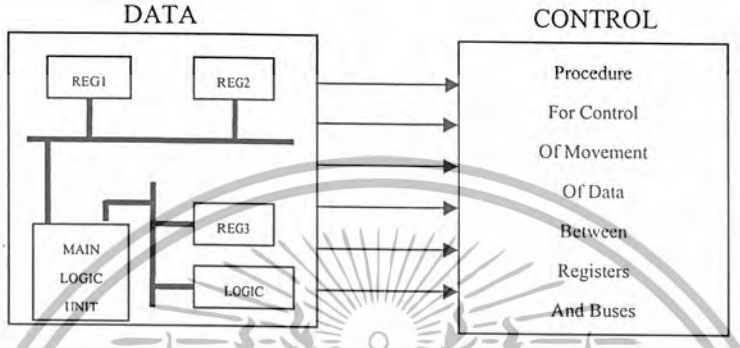
ในการออกแบบระบบดิจิตอล เริ่มตั้งแต่การกำเนิดแนวความคิดเบื้องต้นจนกระทั่งได้ออกมาเป็นอุปกรณ์ฮาร์ดแวร์ที่ใช้งาน ได้จะต้องผ่านขั้นตอนต่างๆ มากมาย และในแต่ละขั้นตอนผู้ออกแบบจะต้องตรวจสอบผลลัพธ์ในแต่ละขั้นก่อนเข้าสู่กระบวนการออกแบบในขั้นต่อไป

รูปที่ 2.22 แสดงขั้นตอนปกติที่ใช้ในการออกแบบ แล้วทำการพัฒนาให้สามารถนำไปใช้ได้อย่างสมบูรณ์ ซึ่งภายในจะทำการสร้างรูปแบบเชิงพฤติกรรมขึ้นมาตรวจสอบ ซึ่งอาจจะเป็นผังงาน ขั้นตอนนี้ผู้ออกแบบจำเป็นต้องสร้างรูปแบบระบบแสดงแบบ หรือรหัสคำสั่งเทียม (Pseudo code) ก็ได้

รูปที่ 2.22 แสดงขั้นตอนการออกแบบระบบดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนต่อไปเป็นการออกแบบระบบเส้นทางของข้อมูล ผู้ออกแบบจะกำหนดส่วนประกอบของรีจิสเตอร์และวงจรลอจิกที่จำเป็นทั้งหมด เพื่อนำมาประกอบเป็นระบบที่สมบูรณ์ โดยแต่ละองค์ประกอบสามารถเชื่อมต่อกันด้วยบัสหนึ่งหรือสองทิศทาง (Unidirectional or Bidirectional Bus) กระบวนการในการควบคุมการเคลื่อนย้ายข้อมูลระหว่างรีจิสเตอร์และวงจรลอจิกจะขึ้นอยู่กับพฤติกรรมของระบบที่กำหนดไว้ดังรูปที่ 2.23



รูปที่ 2.23 แสดงการออกแบบระบบเส้นทางข้อมูล

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

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

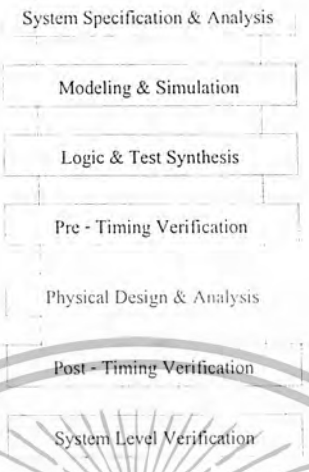
และในขั้นตอนสุดท้ายจะเป็นการส่งระบบที่ออกแบบไว้ไปทำการเจือที่โรงงานเพื่อผลิตออกมาเป็นวงจรรวมในที่สุด

2.6.3 การออกแบบจากบนลงล่าง (Top-Down Design)

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วีเอชดีแอลกับหลักการออกแบบจากบนลงล่างจึงเป็นทางออกให้กับวิศวกรให้สามารถ ออกแบบและพัฒนาวงจรที่มีความซับซ้อนได้มากขึ้น ทั้งยังช่วยลดเวลาและค่าใช้จ่ายในการออกแบบด้วย



รูปที่ 2.24 แสดงขั้นตอนการออกแบบจากบนลงล่าง

จากรูปที่ 2.24 แ สดงถึงขั้นตอนการออกแบบจากบนลงล่าง 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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Delay Time) เสมอ ถึงแม้ว่าจะเป็น เวลาที่น้อยมากในระดับนาโนวินาทีก็ตาม แต่ถ้าภายในวงจรหนึ่งประกอบด้วยเกทของฟังก์ชันต่างๆ จำนวน 10,000 เกท ขึ้นไป เวลาดังกล่าวนี้จะสะสมกันมากขึ้น จนอาจทำให้การทำงานของวงจรรวมทั้งหมดผิดพลาดไป หรือไม่สามารถที่จะทำงานในย่านความถี่สัญญาณนาฬิกาที่สูงได้

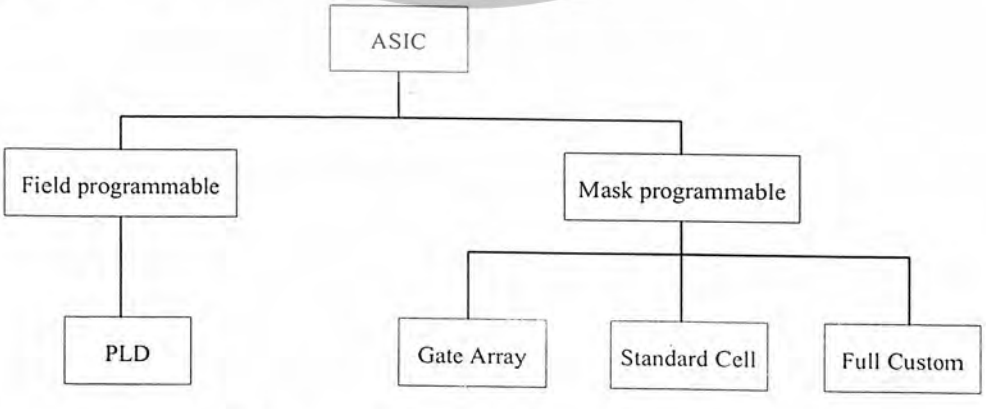
5) การออกแบบทางกายภาพและการวิเคราะห์ คือ ขั้นตอนในการผลิตเป็นวงจรจริง (Technology and device mapping) โดยจะนำข้อมูลที่ได้จากการสังเคราะห์ มาใช้ในการผลิตเป็นวงจรรวม ซึ่งอาจจะอยู่ในรูปของแผงวงจรไฟฟ้า ที่ประกอบด้วยอุปกรณ์หลายๆ ชิ้นหรืออยู่ในรูปของวงจรรวมเอซิก (ASIC)

6) การตรวจสอบเวลาหลังการออกแบบ คือ การทำการตรวจสอบการทำงานด้วยตัวแปรทางด้านเวลาทั้งหมด เพื่อความถูกต้องของวงจรเป็นครั้งสุดท้ายก่อนนำไปรวมเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบดิจิตอล เนื่องจากในขั้นตอนนี้ วงจรที่ออกแบบ จะประกอบด้วยจุดต่อทางอินพุทและเอาท์พุท ซึ่งเป็นจุดต่อสำหรับการรับและส่งสัญญาณกับภายนอก

7) การตรวจสอบระบบ คือ การนำวงจรที่ออกแบบไว้ประกอบเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบที่สมบูรณ์ แล้วทำการทดสอบการทำงานทั้งระบบรวมกับอุปกรณ์อื่นๆ อีกครั้งเพื่อควบคุมคุณภาพของผลิตภัณฑ์

2.7 เอฟพีจีเอ

เทคโนโลยีความก้าวหน้าของอุตสาหกรรมอิเล็กทรอนิกส์ในปัจจุบัน ทำให้เกิดการพัฒนาความสามารถของอุปกรณ์ต่างๆ ซึ่งทำให้ลดค่าใช้จ่ายต่างๆได้มาก ในขณะเดียวกันก็มีการเพิ่มประสิทธิภาพและระดับความเชื่อถือได้ของวงจรรวมที่สูงขึ้นเห็นได้ชัดจากเทคโนโลยีไมโครโปรเซสเซอร์ และหน่วยความจำปัจจุบัน ทุกๆครั้งที่มีการพัฒนาขึ้นทำให้เกิดช่องว่างระหว่างวงจรรวมและไอซีมาตรฐานมากขึ้น นักออกแบบอุปกรณ์ทางด้านดิจิตอล ได้พิจารณาถึงการผลิตให้มีขนาดมากขึ้นและการผลิตวงจรรวมเอซิก (ASIC: Application Specific Integrated Circuit) ซึ่งวงจรรวมจะแบ่งตามสร้างออกเป็น 2 กลุ่ม คือ ฟิลด์โปรแกรมเมเบิล (Field programmable) และ แมสโปรแกรมเมเบิล (Mask programmable) ดังแสดงในรูปที่ 2.25



รูปที่ 2.25 แสดงผังการแบ่งกลุ่มของวงจรรวมเอซิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1 การออกแบบวงจรเชิงเลขด้วยชิพอุปกรณ์เอฟพีจีเอ

ชิพอุปกรณ์เอฟพีจีเอ เป็นอุปกรณ์ที่ใช้ในการโปรแกรมวงจรที่ได้ออกแบบลงไปเพื่อให้อุปกรณ์เอฟพีจีเอ มีฟังก์ชันการทำงานตามที่ออกแบบไว้ ในการทำชิพอุปกรณ์ ซึ่งเป็นวิธีการออกแบบ ไอซี (IC : Integrated Circuit) แบบ เซมิคัสตัม (Semi custom) อีกวิธีหนึ่ง เมื่อเทียบกับการทำ เอซิก แล้วนั้นก็มีทั้งข้อดีและข้อเสีย คือ การทำชิพอุปกรณ์เอฟพีจีเอ จะมีข้อจำกัดในด้านขนาดของวงจรเพราะภายในชิพอุปกรณ์เอฟพีจีเอจะมีจำนวนเกท (gate) ให้ใช้จำนวนจำกัด และการทำชิพอุปกรณ์เอฟพีจีเอ ก็เหมาะสำหรับการทำผลิตภัณฑ์ต้นแบบหรือเพื่อผลิตในปริมาณต่ำ ส่วนข้อดีของการทำชิพอุปกรณ์ก็คือ ระยะเวลาที่ใช้ในการทำตั้งแต่เขียนรหัส (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

2.7.2 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว

1. ผู้ออกแบบไม่จำเป็นต้องทราบถึงโครงสร้างภายในของตัวชิพเพียงแต่มีความรู้เกี่ยวกับขั้นตอนการออกแบบลอจิกก็เพียงพอแล้ว ต่างกับการใช้ไมโครโปรเซสเซอร์ซึ่งจำเป็นต้องศึกษาโครงสร้างภายในรวมถึงภาษาแอสแซมบลี (Assembly) ของไมโครโปรเซสเซอร์ตัวนั้นด้วย

2. มีการออกแบบโดยใช้ภาษาในการอธิบายการทำงานของวงจรหรือเอชดีแอล เป็นเครื่องมือในการออกแบบ ซึ่งเป็นวิธีการที่มีความยืดหยุ่นสูง ทำได้รวดเร็ว และไม่จำเป็นต้องทราบถึงลักษณะของวงจรที่ต้องการว่าจะเชื่อมต่อกันอย่างไร เพียงแต่กำหนดลักษณะการทำงานให้มันจากนั้นตัวซอฟต์แวร์จะทำการสังเคราะห์ ให้ทั้งหมด นอกจากนี้ภาษาที่ใช้ยังเป็นมาตรฐานเดียวกันสามารถใช้ได้กับชิพทุกตัวและทุกบริษัท

3. การโปรแกรมสามารถทำได้เองและใช้เวลาไม่นาน เพียงแค่ส่งข้อมูลผ่านสายดาวน์โหลดทางพอร์ตของคอมพิวเตอร์ก็สามารถโปรแกรมตัวชิพขณะที่อยู่ในระบบได้โดยไม่จำเป็นต้องถอดมาโปรแกรมข้างนอก และที่สำคัญสามารถโปรแกรมได้หลายครั้ง จึงทำให้ง่ายในการแก้ไขและพัฒนาโดยไม่ต้องเสียค่าใช้จ่ายเพิ่มแต่อย่างใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย

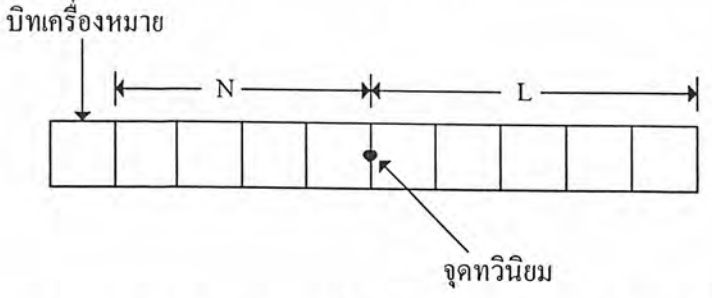
โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) หรือเรียกอย่างย่อๆว่า "DA" เป็นการจัดรูปแบบทางคณิตศาสตร์ของสมการที่อยู่ในรูปของผลบวกของผลคูณ (Sum of Product) ของเลขฐานสิบให้กระจายออกเป็นบิทหรือในรูปแบบของเลขฐานสอง เพื่อให้การคำนวณทางคณิตศาสตร์สามารถแปลงเป็นวงจรดิจิตอลอิเล็กทรอนิกส์ได้ โดยจะปรากฏอยู่ในงานทางด้านการประมวลผลสัญญาณเชิงเลข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

2.8.1 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

2.8.1.1 รูปแบบจำนวนโดยตรง

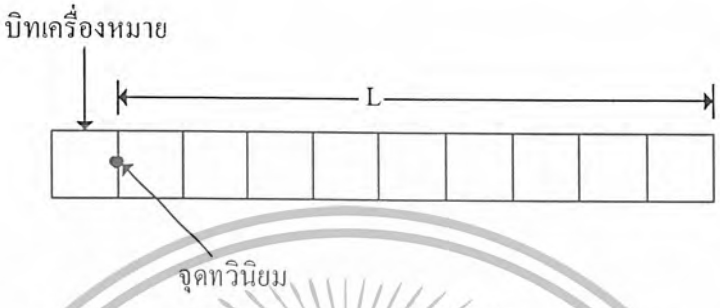
รูปแบบจำนวนโดยตรงปกติจะประกอบไปด้วย 3 ส่วน คือ บิทเครื่องหมาย (Sign bit) 1 บิท, บิทจำนวนเต็ม (Integer bit) N บิท และบิทเศษส่วน (Fractional bit) L บิท โดยจะมีจุดทวินิยม (Binary point) อยู่ระหว่างบิทจำนวนเต็มและบิทเศษส่วนดังแสดงในรูปที่ 2.26



รูปที่ 2.26 แสดงการจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิทจำนวนเต็มและบิทเศษส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนบิท N เป็นตัวกำหนดย่านพลวัตรที่ต้องการ โดยถ้าเลือกให้มีจำนวนน้อยอาจทำให้เกิดการล้น (Overflow) จากการคำนวณได้ แต่ถ้าเลือกให้มีจำนวนมากความเที่ยงตรงก็จะน้อยลง ซึ่งในการสร้างวงจรกรองสัญญาณเชิงเลขโดยการแทนด้วยรูปแบบจำนวนโดยตรงนั้น นิยมที่จะทำมาตราส่วน (Scaling) เพื่อให้ขนาดของสัญญาณมีค่าอยู่ระหว่าง $-1 \le x < 1$ คือมีบิทเครื่องหมาย 1 บิท และบิทเศษส่วน L บิท ดังแสดงในรูปที่ 2.27



รูปที่ 2.27 แสดงการจัดรูปแบบจำนวนโดยตรงที่มีแต่บิทเศษส่วน

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 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

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

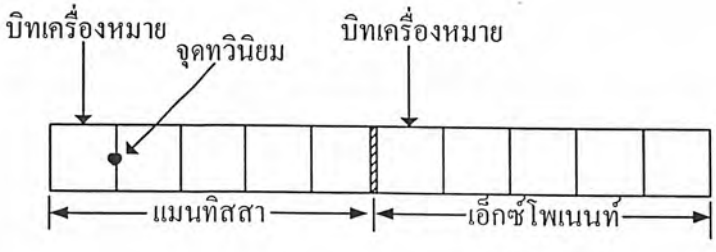
2.8.1.2 รูปแบบจำนวนอิงดรรชนี

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 ถูกตัดให้จำนวนบิทเศษส่วนเหลือเพียง 4 บิท เปอร์เซ็นต์ความผิดพลาดจะเป็น 4.59 % และ 39.6 % ตามลำดับ โดยข้อเสียนี้สามารถแก้ไขได้โดยการใช้รูปแบบจำนวนอิงดรรชนี ซึ่งตัวเลข X แสดงได้โดย

$$X = M \times 2^{e} \tag{2.54}$$

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 2.28 โดยแบ่งเป็น 2 ส่วน คือส่วนหนึ่งสำหรับแมนทิสสา และอีกส่วนสำหรับเอ็กซ์โพเนนท์



รูปที่ 2.28 แสดงการจัดรูปแบบจำนวนอิงดรรชนี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

2.8.2 ทฤษฎีเลขคณิตกระจาย

จากที่ได้กล่าวมาแล้วว่า โครงสร้างเลขคณิตกระจายมีพื้นฐานอยู่บนการคูณแบบเลขส่วนเติมเต็มสอง ดังนั้นในหัวข้อนี้จะได้อธิบายถึงหลักการของการคูณเลขส่วนเติมเต็มสอง

ให้เลขส่วนเติมเต็มสองของ X ซึ่งแทนด้วย $\overline{X}$ และนิยามโดย

$$\bar{x} = \begin{cases} x & \text{ถ้า } X \geq 0 \\ 2 - |x| & \text{ถ้า } X < 0 \end{cases} \tag{2.55}$$

โดย X เป็นเลขที่เป็นเศษส่วน (Fractional number)

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

$$\overline{X} = X_0.X_1X_2...X_L \tag{2.56}$$

ค่าของ $\overline{X}$ ในรูปของเลขฐานสิบสามารถหาได้ดังนี้

$$X = -X_0 + \sum_{i=1}^{L} X_i 2^{-i} \tag{2.57}$$

จากนั้นพิจารณาผลคูณต่อไปนี้

$$Y = Xm \tag{2.58}$$

ให้ $\overline{Y}$, $\overline{X}$ และ $\overline{m}$ เป็นเลขส่วนเติมเต็มสองของ Y, X และ m ตามลำดับ จากนั้นพิจารณาจากสมการที่ (2.57) และ สมการที่ (2.58) จะได้

$$\begin{aligned} Y &= -Y_0 + \sum_{i=1}^{L} Y_i 2^{-i} \\ &= -X_0 m + \sum_{i=1}^{L} X_i m 2^{-i} \end{aligned} \tag{2.59}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น

$$\begin{aligned}\overline{Y} &= \text{ส่วนเติมเต็มสองของ}\left(-X_0 m + 2^{-1}X_1 m + 2^{-2}X_2 m + 2^{-3}X_3 m + \ldots + 2^{-L}X_L m\right) \\ &= \text{ส่วนเติมเต็มสองของ}\left(-X_0 m + 2^{-1}(X_1 m + \ldots + 2^{-1}(X_{L-1} m + 2^{-1}(X_L m)))\right)\end{aligned} \quad (2.60)$$

ต่อไปพิจารณาส่วนเติมเต็มสองของ $2^{-1}U$ โดย

$\overline{U} = U_0.U_1U_2\ldots U_M$; สำหรับ $U \geq 0$ (หรือ $U_0 = 0$)

ส่วนเติมเต็มสองของ $(2^{-1}U) = 2^{-1}\overline{U}$; และสำหรับ $U < 0$ (หรือ $U_0 = 1$)

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

ดังนั้นสรุปได้ว่า

$$\text{ส่วนเติมเต็มสองของ}(2^{-1}U) = \begin{cases} 2^{-1}\overline{U} & ; U_0 = 0 \\ 1 + 2^{-1}\overline{U} & ; U_0 = 1 \end{cases} \quad (2.61)$$

สมการที่ (2.61) นี้แสดงให้เห็นได้ว่า ส่วนเติมเต็มสองของ $(2^{-1}U)$ เป็นการเลื่อนข้อมูลของ $\overline{U}$ ไปทางขวา 1 บิท

$$\therefore \text{ส่วนเติมเต็มสองของ}(2^{-1}U) = 2_2^{-1}\overline{U} \quad (2.62)$$

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

จากนั้นพิจารณาสมการที่ (2.61) และสมการที่ (2.62) จะได้ว่า

$$\begin{aligned}\overline{Y} &= -X_0\overline{m} + 2^{-1}X_1\overline{m} + 2^{-2}X_2\overline{m} + 2^{-3}X_3\overline{m} + \ldots + 2^{-L}X_L\overline{m} \\ &= -X_0\overline{m} + 2^{-1}(X_1\overline{m} + \ldots + 2^{-1}(X_{L-1}\overline{m} + 2^{-1}(X_L\overline{m})))\end{aligned} \quad (2.63)$$

ซึ่งจากสมการที่ (2.63) จะเห็นได้ว่าผลคูณจากสมการที่ (2.58) สามารถหาได้โดยการใช้หลักการเลื่อนและบวก (Shift and add) โดยผลลัพธ์ที่ได้จากการคูณแบบเลขส่วนเติมเต็มสอง สามารถหาได้ตามขั้นตอนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ตัวอย่างการทำงานตามอัลกอริธึมนี้

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

$m = -0.390625$

$\overline{m} = 2 - |m|$ m เป็นเลขลบ

$= 2 - 0.390625$

$= 1.609375$

$\therefore \overline{m} = 1.100111$

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

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

โดยมีขั้นตอนการทำงาน ดังตารางต่อไปนี้

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

ตารางที่ 2.2 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง

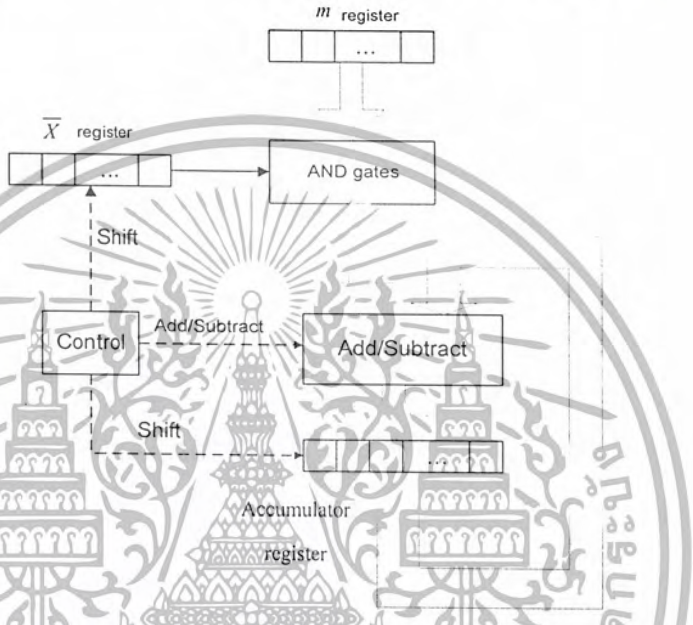
$\therefore \overline{Y} = 1.101\ \ 0111\ \ 0110 = Y_0.Y_1Y_2...Y_{11}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 2.29



รูปที่ 2.29 แสดงการคูณแบบเลขส่วนเติมเต็มสอง โดยใช้เลขคณิตกระจาย

ที่ผ่านมาเป็นหลักการคูณแบบเลขส่วนเติมเต็มสอง ส่วนทฤษฎีเลขคณิตกระจายก็อาศัยหลักการดังกล่าวมาใช้ โดยทำการกระจายสมการที่อยู่ในรูปผลบวกของผลคูณให้แตกออกมาอยู่ในระดับบิท (Bit level) พิจารณาผลบวกของผลคูณต่อไปนี้

$$Y = \sum_{i=0}^{N} m_i\, X_i \qquad (2.64)$$

โดย m_i เป็นค่าสัมประสิทธิ์ซึ่งที่ค่าคงที่

X_i เป็นข้อมูลอินพุท

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

$$X_i = -X_{i0} + \sum_{j=1}^{L} X_{ij}\, 2^{-j} \qquad (2.65)$$

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

X_{i0} = บิทแสดงเครื่องหมาย

X_{iL} = บิทที่มีนัยสำคัญต่ำสุด (LSB)

$L+1$ = จำนวนบิทที่แทนข้อมูลอินพุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แทนค่า X_i ในสมการที่ (2.65) ลงในสมการที่ (2.64) จะได้

$$Y = \sum_{i=0}^{N} m_i \left[-X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \right] \tag{2.66}$$

เมื่อจัดเทอมของผลบวกใหม่จะได้

$$\begin{aligned} Y &= -X_{i0} \sum_{i=0}^{N} m_i + \sum_{j=1}^{L} X_{ij} 2^{-j} \sum_{i=0}^{N} m_i \\ &= -\sum_{i=0}^{N} X_{i0} m_i + \sum_{i=0}^{N} \sum_{j=1}^{L} X_{ij} m_i 2^{-j} \end{aligned} \tag{2.67}$$

จากนั้นทำการกระจายออกให้เป็นระดับบิท ได้ดังนี้

$$\begin{aligned} Y = &-(X_{00}m_0 + X_{10}m_1 + X_{20}m_2 + ... + X_{N0}m_N) \\ &+ 2^{-1}(X_{01}m_0 + X_{11}m_1 + X_{21}m_2 + ... + X_{N1}m_N) \\ &+ 2^{-2}(X_{02}m_0 + X_{12}m_1 + X_{22}m_2 + ... + X_{N2}m_N) \\ &+ ... + 2^{-L}(X_{0L}m_0 + X_{1L}m_1 + X_{2L}m_2 + ... + X_{NL}m_N) \end{aligned} \tag{2.68}$$

สมการที่ (2.68) นี้ถูกกระจายออกให้อยู่ในรูปผลบวกของผลคูณระหว่างสัมประสิทธิ์กับข้อมูลอินพุทในระดับบิท ซึ่งเป็นนิยามของการคำนวณแบบเลขคณิตกระจาย และเมื่อเปรียบเทียบสมการที่ (2.68) กับสมการที่ (2.63) จะเห็นว่าการคำนวณหาค่า Y ก็ใช้เลขคณิตกระจายนั่นเอง เพียงแต่นำค่าผลคูณย่อย (Partial product) ที่คำนวณไว้ล่วงหน้าแล้วสำหรับแต่ละค่าที่สอดคล้องกับแต่ละบิทของข้อมูลอินพุทไปเก็บไว้ในตารางเปิดดู ซึ่งเป็นหน่วยความจำ EPROM และใช้ข้อมูลอินพุทเป็นแอดเดรสของหน่วยความจำ เพื่อนำค่าในตารางเปิดดูมาผ่านขั้นตอนการคำนวณตามบูทอัลกอริธึม ซึ่งค่าในตารางเปิดดูสามารถแสดงได้ดังนี้

Bit pattern ของข้อมูลอินพุท	ผลคูณย่อยที่เก็บไว้ในตารางเปิดดู
X_{Nj} ..... X_{2j} X_{1j} X_{0j}	
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	$m_2 + m_1$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bit pattern ของข้อมูลอินพุท	ผลคูณย่อยที่เก็บไว้ในตารางเปิดดู
X_{Nj} ..... X_{2j} X_{1j} X_{0j}	
0 ..... 1 1 1	$m_2 + m_1 + m_0$
⋮	⋮
1 ..... 1 1 1	$m_N + m_{N-1} + ... + m_2 + m_1 + m_0$

ตารางที่ 2.3 แสดงค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุท

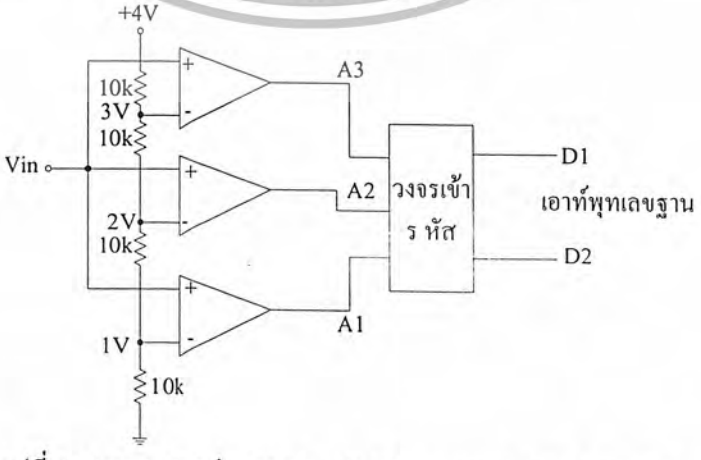
2.9 วงจรแปลงอะนาลอกเป็นดิจิตอล (Analog to Digital Converter: A/D)

วงจรเปลี่ยนสัญญาณอะนาลอกเป็นดิจิตอล ที่ใช่อยู่ทั่วๆไปมีหลายแบบ คือ

2.9.1. แบบใช้วงจรเปรียบเทียบขนาน หรือแบบแฟลซ (Parallel Comparator Simultaneous "Flash" A/D Converter) วงจรเอทูดีแบบนี้ใช้หลักการง่ายๆ อีกทั้งยังเป็นวิธีที่รวดเร็วที่สุด คือ ใช้วงจรเปรียบเทียบที่ต่อขนานกันดังรูปที่ 2.30 ประกอบด้วยออปแอมป์ที่ต่อเป็นวงจรเปรียบเทียบ และตัวต้านทานต่อไว้เพื่อแบ่งแรงดันที่ขาอินพุตแบบกลับ(inverting) ให้มีขนาดต่างๆกัน จากหลักการของวงจรเปรียบเทียบทั่วไปเมื่อแรงดันที่ขาอินพุตแบบไม่กลับ(non-inverting) มีค่าสูงกว่าที่ขาอินพุตแบบกลับเอาท์พุตจะได้แรงดันค่าต่างๆมีผลต่อเอาท์พุต ของวงจรเปรียบเทียบแต่ละตัวอย่างไร ซึ่งเอาต์พุตที่ได้จากวงจรเปรียบเทียบนี้จะนำไปเข้ารหัสให้เป็นเลขฐานสองต่อไป

จำนวนของวงจรเปรียบเทียบที่ต้องใช้ในวงจรขึ้นอยู่กับขนาดของสัญญาณอะนาลอกที่อินพุตจากวงจรรูปที่ 2.30 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



รูปที่ 2.30 แสดงการต่อวงจร Parallel Comparator A/D Converter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ข้อเสียอีกประการหนึ่งก็คือ เอาท์พุตที่ได้ไม่เป็นเลขฐานสอง ต้องมีวงจรเพิ่มเติมไปทำงานเข้ารหัส "แฟลช" (Flash Type A/D Converter) วงจรเอทูดีชนิดนี้ ใช้เวลาในการแปลงได้เร็วในระดับนาโนวินาทีทีเดียว

2.9.2 วงจรเอทูดีที่ใช้การอินทีเกรต วงจรเปลี่ยนสัญญาณเอทูดีที่ใช้ เทคนิคการอินทีเกรตสัญญาณมี 4 แบบ คือ

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

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

สัญญาณแรมป์มีแรงดันเป็นบวกขึ้นเรื่อยๆ จนมากกว่าระดับแรงดันอินพุตและเอาท์พุทของ วงจรเปรียบเทียบก็ตกลงมาเป็นระดับ "Low" ปิดแอนด์เกต ไม่มีสัญญาณผ่านไปให้วงจรนับ

วงจรนับจะหยุดนับและเก็บค่าไว้ที่วงจรแลทช์ จากนั้นจึงทำการรีเซต วงจรนับและวงจรกำเนิดสัญญาณแรมป์

วงจรแบบนี้เป็นหลักการเบื้องต้นของดิจิตอลโวลต์มิเตอร์ซึ่งถ้าใช้วงจรนับเลขฐานสองแทนแบบ BCD เอาต์พุตก็จะอ่านค่าได้เลขฐานสองโดยตรง

วงจรลักษณะนี้ มักนำไปใช้งานในการเปลี่ยนเวลาเป็นขนาดของสัญญาณ (Time to Amplitude Conversion) หรืออาจใช้ในดิจิตอลโวลท์มิเตอร์ แต่ไม่ใช้กับงานที่ต้องการความถูกต้องสูง เนื่องจากการเปลี่ยนแปลงในแหล่งกำเนิดสัญญาณแรมป์ขึ้นกับอุณหภูมิ และผลตอบสนองต่อสัญญาณอินพุต ทำให้ไม่มีความคงที่ ดังนั้นจึงมีการปรับปรุงให้ดีขึ้นกลายเป็นแบบสโลบคู่ (Dual Slope)

- แบบสโลบคู่ (Dual – Slope A/D Converters)

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

วงจรเปรียบเทียบก็จะได้แรงดันลบจากวงจรอินทีเกรเตอร์ แล้วให้เอาต์พุตเป็นบวกทำการเปิดสัญญาณแอนด์เกตให้สัญญาณนาฬิกาผ่านเข้าสู่วงจรนับ วงจรนับจะทำการนับไปยังค่าที่กำหนดไว้คงที่แล้วทำการสับสวิตช์ต่อเข้ากันกับแรงดันอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 แต่แทนที่จะให้อินพุตสวิตช์ 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

ถ้าให้อินพุตเป็นกระแสก็ไม่ต้องคำนึงถึงแรงดันออฟเซตของวงจรอินทีเกรเตอร์ แต่ถ้าหากใช้กับอินพุตที่เป็นแรงดันต้องปรับออฟเซตของออปแอมป์เสียก่อน

การใช้อินพุตเป็นกระแสทำให้ย่านการใช้งานทางไฟสลับกว้าง

2.9.3 วงจรเปลี่ยนสัญญาณเอทูดีที่ใช้วงจรนับและวงจรดีทูเอประกอบกัน

2.9.3.1 แบบบวงจรนับเดี่ยว (Single Counter)

แท้ที่จริงแล้วสัญญาณแรมป์เชิงเส้น (Linear ramp) อาจประกอบขึ้นด้วยสัญญาณขั้นบันไดเล็กๆ จำนวนมากที่เกิดจากการต่อเอาต์พุทของวงจรนับเข้ากับวงจรแปลงดีทูเอ โดยขนาดของขั้นบันไดแต่ละขั้นขึ้นอยู่กับจำนวนบิตหรือความละเอียดของวงจรดีทูเอนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำเนิดสัญญาณแรมป์เดี่ยวด้วยวงจรนับ และวงจรดีทูเอ เมื่อเริ่มแปลงสัญญาณวงจรนับจะถูกรีเซต เอาต์พุทของวงจรดีทูเอ มีระดับ 0 เมื่อแรงดันถูกป้อนเข้าไปยังอินพุทของวงจรเปรียบเทียบ เอาต์พุทก็จะขึ้นสู่ระดับ "High" และเปิดสัญญาณนาฬิกาไปสู่วงจรนับ แต่ละพัลส์ของสัญญาณนาฬิกา ทำให้เกิดการนับและเพิ่มแรงดันขึ้น 1 ขั้น

เมื่อเอาท์พุตของดีทูเอมีค่ามากกว่าอินพุต Vin เอาท์พุตของวงจรเปรียบเทียบก็จะกลายเป็น "Low" ทำให้สัญญาณนาฬิกาไม่อาจผ่านไปยังวงจรนับได้ ดังนั้นวงจรควบคุมจะทำการแลทช์เอาต์พุตของวงจรนับและรีเซตวงจรนับให้เริ่มต้นรอบใหม่อีกครั้งหนึ่ง

การทำงานจะคล้ายกับแบบใช้วงจรนับเดี่ยว แต่การนับจะไม่ได้เริ่มจากศูนย์ แต่จะทำการนับขึ้นหรือ นับลงจากค่าล่าสุดไปยังค่าใหม่ แล้วแต่ว่าแรงดันอินพุตในรอบใหม่มีค่าสูงกว่าหรือต่ำกว่าค่าที่แล้ว ข้อดีของดีทูเอแบบแทร็กกิ้ง คือ ทำงานได้เร็วขึ้น

2.9.4 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

2.9.5 การสุ่มและการคงค่า (Sample and Holds)

วงจรเอทูดีต้องการเวลาในการนับในการแปลงสัญญาณหรือ เรียกว่า Conversion Time เพื่อเปลี่ยนสัญญาณอะนาล็อกเป็นสัญญาณดิจิตอลที่เหมาะสม ถ้าสัญญาณอะนาล็อกมีการเปลี่ยนแปลงในระหว่างช่วงเวลาการแปลง เอาท์พุทของวงจรเปลี่ยนสัญญาณอาจเกิดความผิดพลาดจึงต้องมีการป้องกันด้วยวงจรสุ่มและคงค่าสัญญาณ (Sample and Holds) เพื่อใช้จับสัญญาณอะนาล็อกที่จุดเริ่มของการเปลี่ยนสัญญาณแล้วเก็บไว้ในตัวเก็บประจุระหว่างช่วงการแปลง หลังจากที่มีการเปลี่ยนสัญญาณเสร็จสิ้น จึงจับสัญญาณอะนาล็อกค่าใหม่มาเก็บประจุระหว่างช่วงเวลาการแปลง หลังจากที่การแปลงสัญญาณเสร็จสิ้น จึงสัญญาณอะนาล็อกค่าใหม่มาเก็บไว้อีกครั้งเป็นเช่นต่อๆ ไป เราจึงมักได้พบเห็นวงจรสุ่มและคงค่าสัญญาณปรากฏคู่กับวงจรเอทูดีอยู่เสมอ

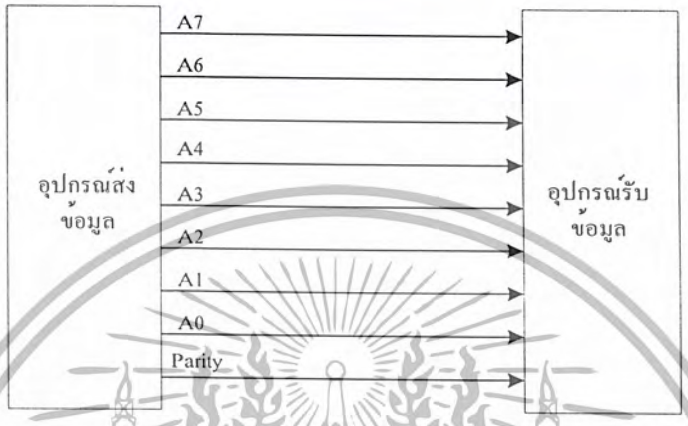
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 พอร์ตอนุกรม

2.10.1 การสื่อสารข้อมูล

2.10.1.1 การสื่อสารข้อมูลแบบขนาน (Parallel Communication)

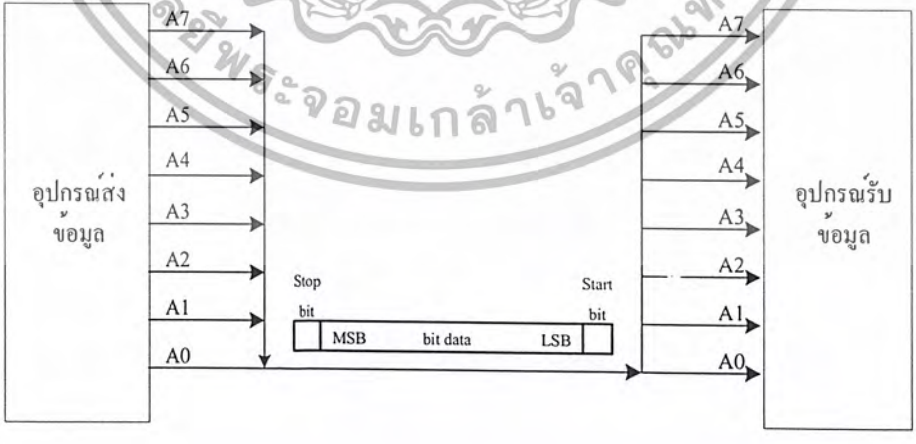
การสื่อสารข้อมูลแบบขนาน คือ การสื่อสารแบบที่ส่งข้อมูลพร้อมๆกัน n บิท ผ่านสายสัญญาณ n เส้น สามารถแสดงรูปแบบการสื่อสารข้อมูลแบบขนานได้ดังรูปที่ 2.31



รูปที่ 2.31 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารข้อมูลแบบขนาน

2.10.1.2 การสื่อสารข้อมูลแบบอนุกรม (Serial Communication)

การสื่อสารข้อมูลแบบอนุกรม คือ การสื่อสารแบบที่ส่งข้อมูลที่ละบิต ผ่านสายสัญญาณเส้นเดียวจนครบจำนวนข้อมูลที่ต้องการ โดยเฟรมของการสื่อสารข้อมูลแบบอนุกรมประกอบด้วย สตาร์ทบิท(start bit) , สตอบบิท (stop bit) , ข้อมูล (bit data) สามารถแสดงรูปแบบการสื่อสารข้อมูลแบบอนุกรมได้ดังรูปที่ 2.32



รูปที่ 2.32 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราบอดเรด (Baud Rate) ในการสื่อสาร คือ ความเร็วในการรับ-ส่งข้อมูลแบบอนุกรมมีหน่วยเป็นบิทต่อวินาที (bit/sec) ซึ่งจะบอกถึงจำนวนบิทที่รับ-ส่ง ในเวลา 1 วินาที เช่น ส่งข้อมูลด้วยอัตรา 9600 บิทต่อวินาที หมายถึง เวลา 1 วินาที รับ-ส่งข้อมูลได้ 9600 บิท รวมทั้งบิทข้อมูล (Data bit) สตาร์ทบิท (Start bit) สตอบบิท (Stop bit) ด้วย

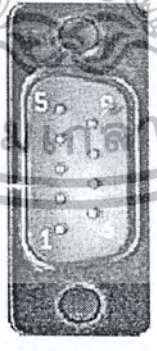
2.10.2 การอินเตอร์เฟสตามมาตรฐาน RS-232

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 จะใช้สัญญาณเส้นเดียวในการส่งสัญญาณโดยจะสัญญาณจะส่งไปในทิศทางเดียวกัน 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 2.33 และการเชื่อมต่อของพอร์ตสื่อสารสำหรับคอนเน็คเตอร์แบบ DB-9 สามารถแสดงได้ดังตารางที่ 2.4



รูปที่ 2.33 แสดงลักษณะของคอนเน็กเตอร์แบบ DB-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

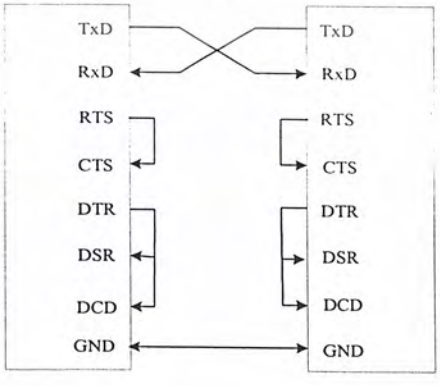
ตารางที่ 2.4 การเชื่อมต่อของพอร์ตสื่อสารสำหรับคอนเน็คเตอร์แบบ DB-9

เปรียบเทียบข้อดีข้อเสียของการสื่อสารข้อมูลแบบอนุกรมและขนาน

- การสื่อสารข้อมูลแบบอนุกรมสามารถสื่อสารได้ระยะทางที่ไกลกว่า
- การสื่อสารข้อมูลแบบอนุกรมใช้สายสัญญาณที่ประหยัดกว่า
- การสื่อสารข้อมูลแบบขนานสามารถสื่อสารข้อมูลได้ทีละมากๆ และรวดเร็วกว่า

2.10.3 นันโมเด็ม (Null Modems)

นันโมเด็ม (Null Modems) เป็นการเชื่อมต่อระหว่าง DTE (Data Terminal Equipment) เข้าด้วยกัน ซึ่งสามารถแสดงการเชื่อมต่อแบบนันโมเด็มได้ดังรูปที่ 2.34

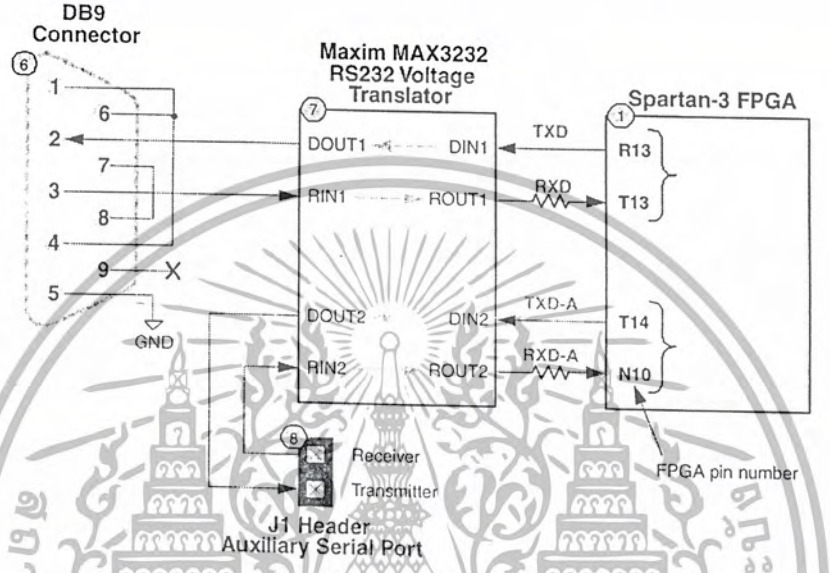


รูปที่ 2.34 แสดงบล็อกไดอะแกรมการเชื่อมต่อแบบนันโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.4 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 2.35



รูปที่ 2.35 แสดงการเชื่อมต่อระหว่างคอนเน็คเตอร์แบบ DB-9 และ บอร์ด FPGA โดยผ่าน MAX3232

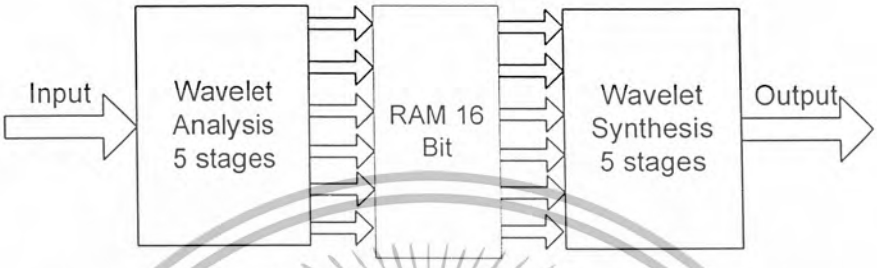
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การคำนวณและการสร้าง

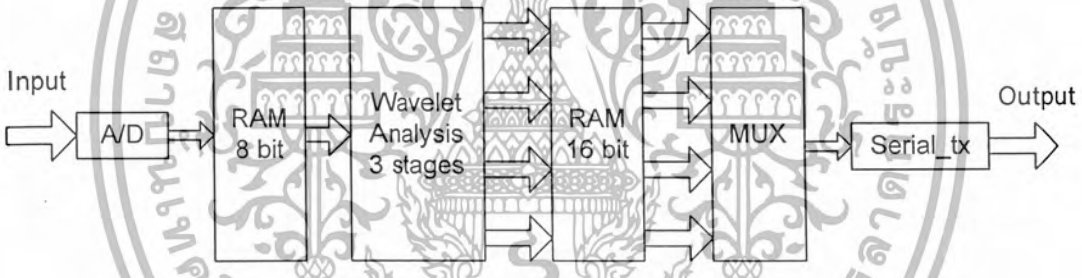
การแปลงเวฟเลตแบบ 1 มิติ ในบทนี้จะแบ่งออกเป็น 2 ส่วนใหญ่ๆดังนี้

3.1 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL ในการออกแบบมีบล็อกไดอะแกรม ดังรูปที่ 3.1



รูปที่ 3.1 บล็อกไดอะแกรมในส่วนของการ Simulate

3.2 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA โดยมีบล็อกไดอะแกรมดังรูปที่ 3.2



รูปที่ 3.2 บล็อกไดอะแกรมในส่วนของการ ส่วนของการ Implement ลงบนอุปกรณ์ FPGA

<u>การออกแบบ</u>

3.1 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL

3.1.1 สัมประสิทธิ์ที่นำมาใช้ในการแปลงเวฟเลต โดยเลือกใช้สัมประสิทธิ์ดาร์บีชี 4 ค่า

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

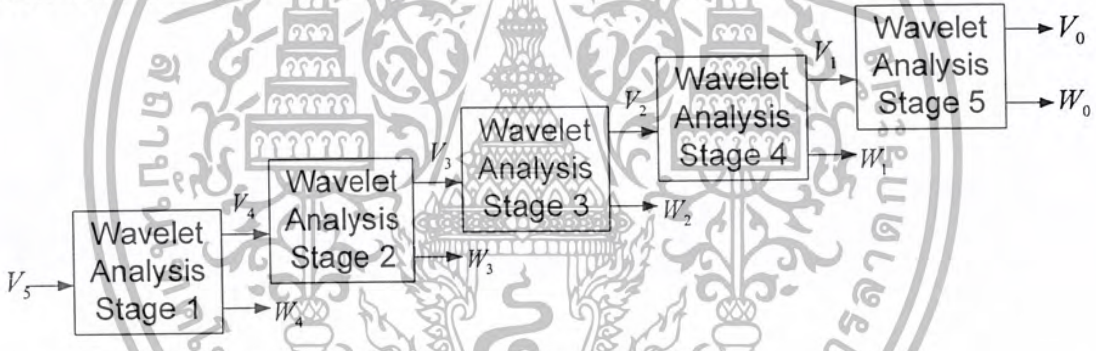
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเพื่อให้ได้ผลตรงตามหลักการการกู้กลับอย่างสมบูรณ์ (Perfect Reconstruct) จะทำให้ได้สัมประสิทธิ์ในการแปลงเวฟเลตทางด้านสังเคราะห์ในภาครับ โดยสามารถหาได้จากสมการที่ 2.39 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

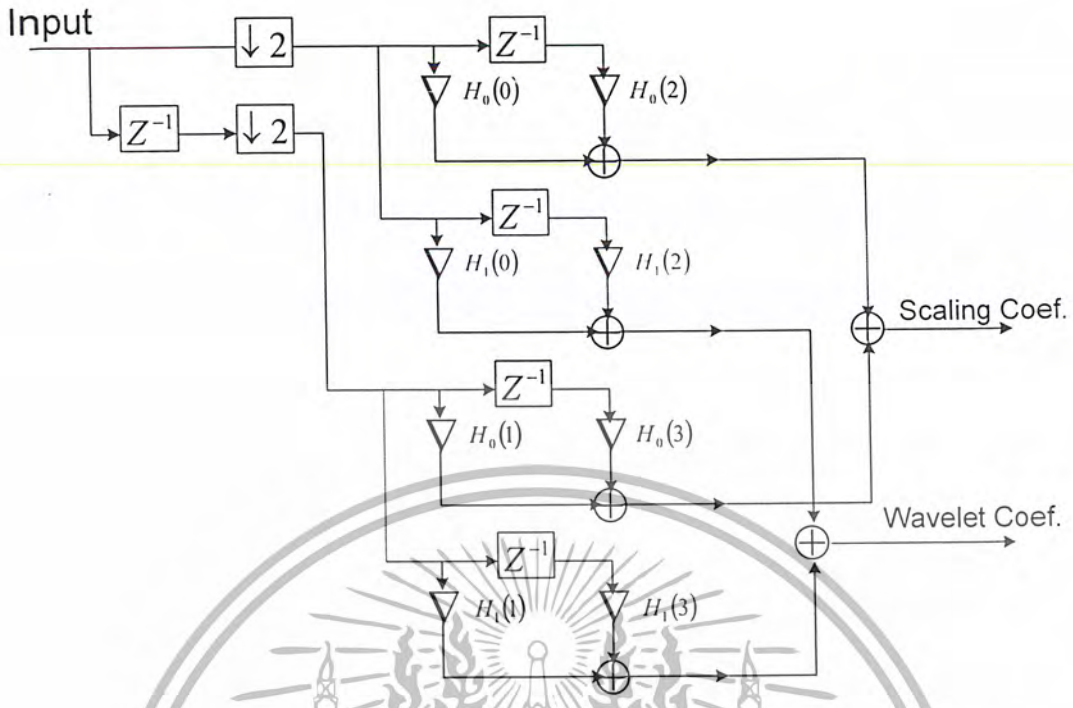
3.1.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ 5 ระดับ จะประกอบด้วยการแปลงเวฟเลตทางด้านวิเคราะห์พื้นฐาน 5 ระดับ ดังรูปที่ 3.3



รูปที่ 3.3 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านวิเคราะห์ 5 ระดับ

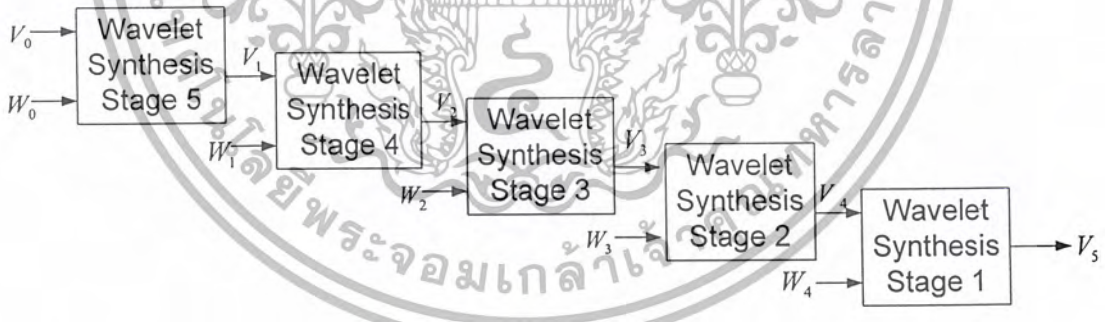
โดยที่ในแต่ละระดับจะมีโครงสร้างภายในเป็นโครงสร้างของฟิลเตอร์แบบโพลีเฟส เหมือนกันในทุกๆระดับ พิจารณาโครงสร้างในระดับที่ 1 มีส่วนประกอบภายใน ดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 โครงสร้างภายในของการแปลงเวฟเลตด้านวิเคราะห์ ทางภาคส่ง

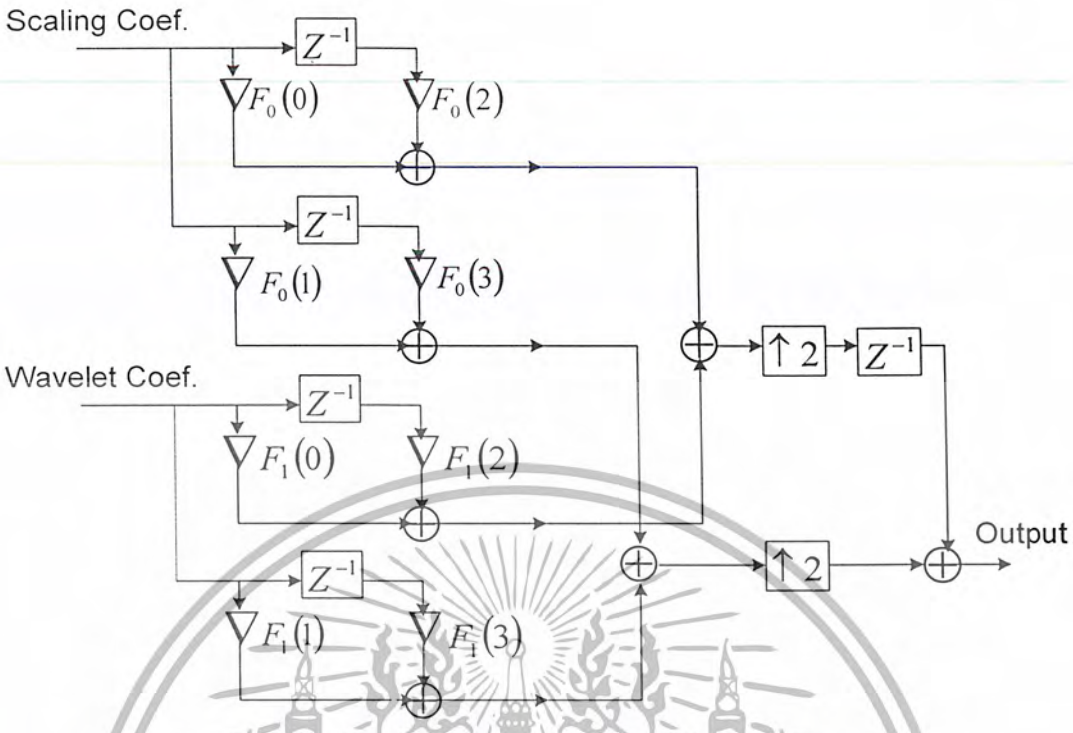
3.1.3 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ 5 ระดับ จะประกอบด้วยการแปลงเวฟเลตด้านสังเคราะห์ในขั้นพื้นฐาน 5 ระดับ ดังรูปที่ 3.5



รูปที่ 3.5 บล็อกไดอะแกรมของการแปลงเวฟเลตด้านสังเคราะห์ 5 ระดับ

โดยที่ในแต่ละระดับจะมีโครงสร้างภายในเป็นโครงสร้างของฟิลเตอร์แบบโพลีเฟส เหมือนกันในทุกๆระดับ พิจารณาโครงสร้างในระดับที่ 1 มีส่วนประกอบภายใน ดังรูปที่ 3.6

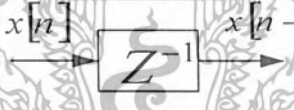
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 โครงสร้างภายในของการแปลงเวฟเลตทางด้านสังเคราะห์ ทางภาครับ

3.1.4 ส่วนประกอบภายในของฟิลเตอร์แบบโพลีเฟส

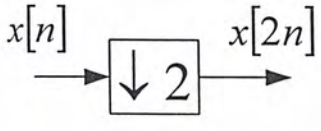
3.1.4.1 วงจรหน่วงเวลา (Delay)



รูปที่ 3.7 บล็อกไดอะแกรมของวงจรหน่วงเวลา

วงจรหน่วงเวลามีหลักการทำงาน คือ จะทำการเลื่อนเวลาของอินพุตที่เข้ามาให้มีเวลาที่ช้ากว่าเดิม อยู่ 1 หน่วย ดังรูปที่ 3.7

3.1.4.2 วงจรสุ่มค่าตัวอย่าง (Down-Sampling)



รูปที่ 3.8 บล็อกไดอะแกรมของวงจรสุ่มค่าตัวอย่าง

จากรูปที่ 3.8 จะเห็นได้ว่าวงจรนี้จะทำการลดขนาดของอินพุตลงเหลือครึ่งหนึ่งเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4.3 วงจรเพิ่มค่าตัวอย่าง (Up-Sampling)

$$x(n) \longrightarrow \boxed{\uparrow 2} \longrightarrow x\left(\frac{n}{2}\right)$$

รูปที่ 3.9 บล็อกไดอะแกรมของวงจรเพิ่มค่าตัวอย่าง

จากรูปที่ 3.9 จะเห็นได้ว่าเมื่อมีอินพุตเข้ามา 1 ค่า เอาต์พุตจะมีค่าของอินพุตตัวเดิมพร้อมด้วยการแทรกค่าศูนย์เข้าไประหว่างค่าเดิม ทำให้ขนาดของเอาต์พุตที่ได้มีขนาดเป็น 2 เท่าเมื่อเทียบกับขนาดของอินพุต

3.1.4.4 วงจรบวก (Adder)

รูปที่ 3.10 บล็อกไดอะแกรมของวงจรบวก

ในการบวกข้อมูล 14 บิท 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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4.5 วงจรคูณ (Multiplier)

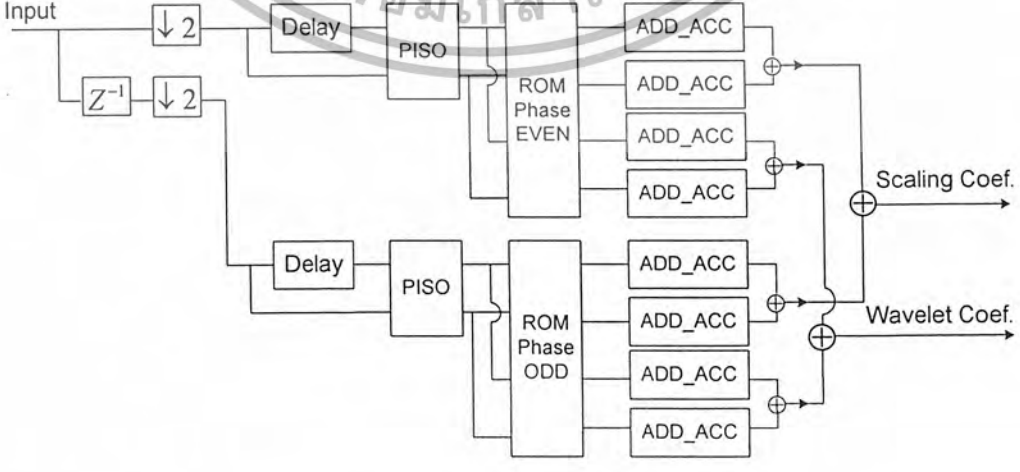
รูปที่ 3.11 บล็อกไดอะแกรมของวงจรคูณ

ในส่วนของวงจรคูณนั้น อินพุตที่เข้ามาจะทำการคูณกับค่าคงที่ที่มีอยู่แล้ว ซึ่งจะได้ผลลัพธ์เป็นผลคูณระหว่างอินพุตกับค่าคงที่ ดังรูปที่ 3.11

ในส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ และการแปลงเวฟเลตทางด้านสังเคราะห์ จะพบว่าภายใน 1 ระดับของการแปลงเวฟเลตด้านวิเคราะห์และการแปลงเวฟเลตทางด้านสังเคราะห์ จะใช้วงจรคูณระดับละ 8 วงจรสำหรับการคูณกับสัมประสิทธิ์ 4 ค่า และเนื่องจากเราทำทั้งหมด 5 ระดับ จึงจะต้องใช้วงจรคูณจำนวนทั้งหมด 80 วงจร ในการ Simulate วงจรคูณจะใช้ทรัพยากรเกตจำนวนมาก ดังนั้นจึงพัฒนามาใช้หลักการเลขคณิตกระจายมาประยุกต์ใช้ ซึ่งหลักการนี้จะใช้จำนวนทรัพยากรเกตน้อยกว่าวงจรคูณในรูปแบบปกติ

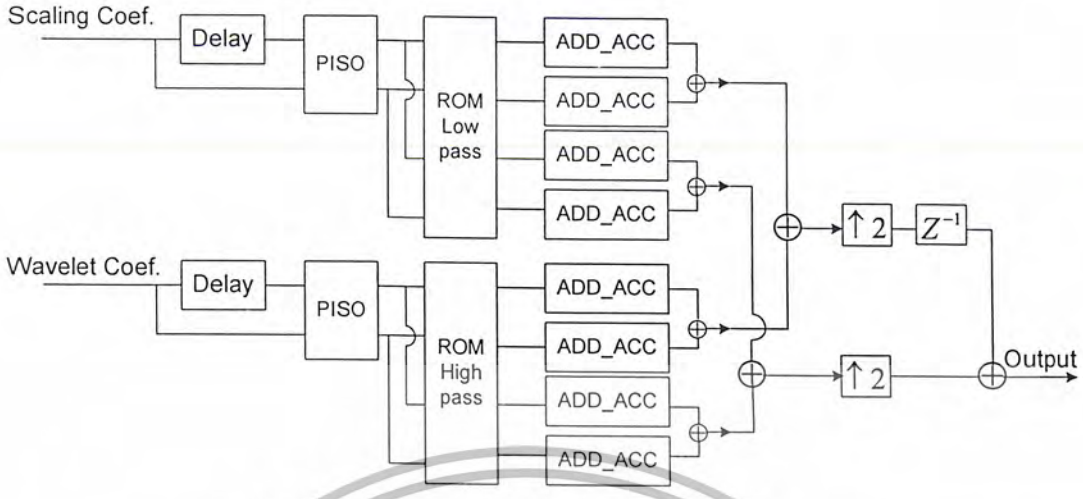
3.1.5 การประยุกต์ใช้งาน DA (Distributed Arithmetic) กับการแปลงเวฟเลตโดยใช้ฟิลเตอร์ที่มีโครงสร้างแบบโพลีเฟส

การแปลงแบบเวฟเลต โดยใช้โครงสร้างของฟิลเตอร์แบบโพลีเฟสมีจำนวนของวงจรคูณเป็นจำนวนมาก ทำให้มีการใช้ทรัพยากรเกตสิ้นเปลือง ดังรูปที่ 3.4 และ 3.6จึงมีการประยุกต์ใช้งานบนโครงสร้างของ DA ซึ่งคุณสมบัติของ DA นี้จะทำงานอยู่ในรูปแบบของผลบวกของผลคูณ (Sum of Products) โดยไม่ใช้การคูณโดยตรงในการประมวลผลของข้อมูล ซึ่งแนวความคิดนี้เหมาะสมกับรูปแบบการคำนวณของการแปลงเวฟเลต โดยสามารถเปลี่ยนให้อยู่ในรูปโครงสร้าง ดังรูปที่ 3.12



รูปที่ 3.12 การแปลงเวฟเลตด้านวิเคราะห์ โดยนำหลักการของ DA มาใช้

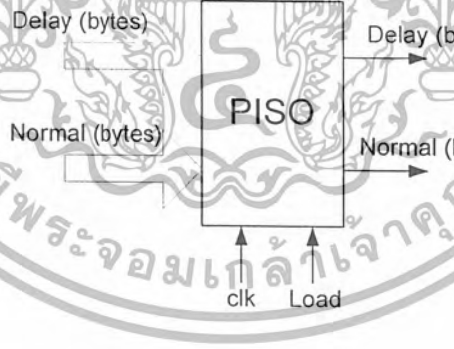
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 การแปลงเวฟเลตด้านสังเคราะห์ โดยนำหลักการของ DA มาใช้

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 บล็อกไดอะแกรมของ วงจร Parallel in Serial out Shift Register

การทำงานของวงจร เมื่อมีสัญญาณอินพุต (Input) เข้ามา และสัญญาณที่ขา $\overline{Load}$ มีค่า '1' จะทำการ *Load* ข้อมูลเข้ามาเลขเก็บในชิพรีจีสเตอร์ (Shift Register) เมื่อขา $\overline{Load}$ มีค่า '0' เมื่อป้อนสัญญาณนาฬิกา (Clock) ก็จะทำให้รีจีสเตอร์ (Register) ทำการเลื่อน ข้อมูลออกเอาท์พุต (Output) ทีละบิต เริ่มจากบิทที่มีนัยสำคัญต่ำสุด (LSB) ตามสัญญาณนาฬิกา (Clock)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5.2 หน่วยความจำ (ROM)

ในการแปลงเลฟเลตและการแปลงเวฟเลตกลับนั้น จะเก็บค่าสัมประสิทธิ์ค่าต่างๆไว้ในรอม ซึ่งมีค่า 14 บิท ในการแปลงเวฟเลตจะใช้รอม 2 ตัวในการเก็บค่าสัมประสิทธิ์ 8 ค่า ซึ่งจะแบ่งออกเป็น ROM phase even และ ROM phase odd ส่วนในการแปลงเวฟเลตกลับจะใช้รอม 2 ตัวเช่นกัน เพื่อใช้ในการเก็บค่าสัมประสิทธิ์ 8 ค่า ซึ่งจะแบ่งออกเป็น ROM Low-pass และ ROM High-pass เป็นเลขทศนิยมฐาน 2 แบบคิดเครื่องหมาย ดังตารางที่ 3.4-3.7

ROM phase EVEN					
Normal bit	Delay bit	$Y_0(0)$	$Y_0(2)$	$Y_1(0)$	$Y_1(2)$
0	0	00000000000000	00000000000000	00000000000000	00000000000000
0	1	00000000000000	00011100101100	00000000000000	01101011000100
1	0	00111101110100	00000000000000	11101111011100	00000000000000
1	1	00111101110100	00011100101100	11101111011100	01101011000100

ตารางที่ 3.4 แอดเดรสและค่าที่เก็บใน ROM phase EVEN

ROM phase ODD					
Normal bit	Delay bit	$Y_0(1)$	$Y_0(3)$	$Y_1(1)$	$Y_1(3)$
0	0	00000000000000	00000000000000	00000000000000	00000000000000
0	1	00000000000000	11101111011100	00000000000000	11000010001100
1	0	01101011000100	00000000000000	11100011010100	00000000000000
1	1	01101011000100	11101111011100	11100011010100	11000010001100

ตารางที่ 3.5 แอดเดรสและค่าที่เก็บใน ROM phase ODD

ROM Low pass					
Normal bit	Delay bit	$Y_0(0)$	$Y_0(2)$	$Y_0(1)$	$Y_0(3)$
0	0	00000000000000	00000000000000	00000000000000	00000000000000
0	1	00000000000000	01101011000100	00000000000000	00111101110100
1	0	11101111011100	00000000000000	00011100101100	00000000000000
1	1	11101111011100	01101011000100	00011100101100	00111101110100

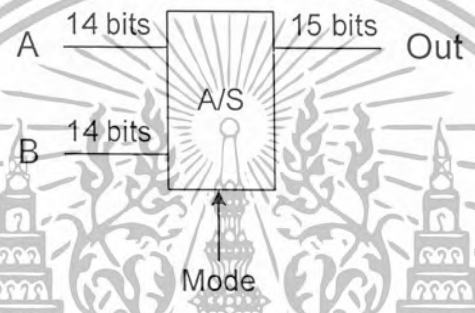
ตารางที่ 3.6 แอดเดรสและค่าที่เก็บใน ROM Low pass

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ROM High pass					
Normal bit	Delay bit	$Y_1(0)$	$Y_1(2)$	$Y_1(1)$	$Y_1(3)$
0	0	00000000000000	00000000000000	00000000000000	00000000000000
0	1	00000000000000	11100011010100	00000000000000	11101111011100
1	0	11000010001100	00000000000000	01101011000100	00000000000000
1	1	11000010001100	11100011010100	01101011000100	11101111011100

ตารางที่ 3.7 แอดเดรสและค่าที่เก็บใน ROM High pass

3.1.5.3 วงจรบวกหรือวงจรลบ (Adder/Subtractor)

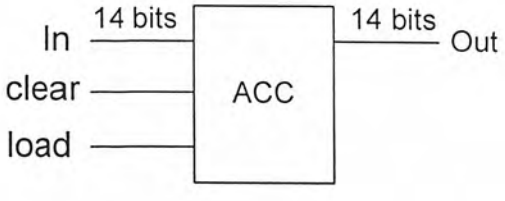


รูปที่ 3.15 บล็อกไดอะแกรมของวงจรบวกหรือวงจรลบ

ในส่วนของวงจรบวกหรือลบ (Adder & Subtractor) คือ วงจรบวกและวงจรลบที่จะรับค่าที่อ่านจาก ROM เข้ามาทำการบวกหรือลบกับค่าใน Accumulator ซึ่งสามารถเลือกการบวกหรือลบได้จากขา mode ผลลัพธ์ (output) จะมี 15 บิท บิทที่มีนัยสำคัญสูงสุด MSB หรือบิทที่ 15 จะเป็น บิทเช็คเครื่องหมายเพื่อนำไปเลื่อนแทนที่ในการเลื่อนขวาของข้อมูลในวงจรเก็บค่า (Accumulator)

3.1.5.4 ส่วนเก็บค่าผลการคำนวณ (Accumulator)

มีหน้าที่ ทำการ Load ค่า output ที่ได้จากการบวกหรือลบ (A/S) เข้ามาเก็บแล้วทำการเลื่อนขวา 1 บิท แล้วส่งกลับไปบวกหรือลบกับค่าต่อไปในส่วนของ A/S ตามรูปที่ 3.16

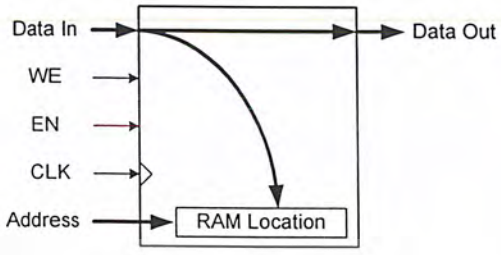


รูปที่ 3.16 บล็อกไดอะแกรมของ Accumulator

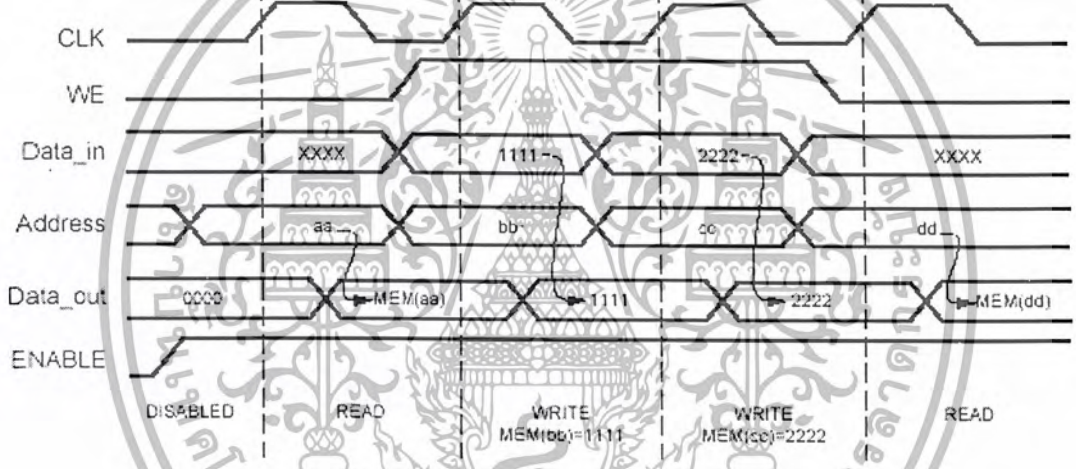
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.6 Random Access Memory (RAM)

ในส่วนนี้จะเลือกใช้ RAM ที่มี write mode เป็นแบบ write first mode โดยจะมีบล็อกไดอะแกรม ดังรูปที่ 3.17



รูปที่ 3.17 แสดงการไหลของข้อมูลในระหว่างกระบวนการ Write

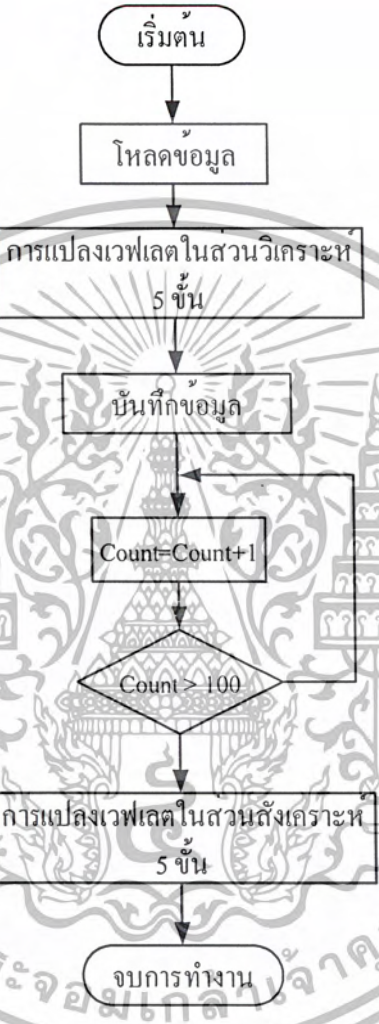


รูปที่ 3.18 แสดงเวฟฟอร์มของ Write first mode

จากรูปที่ 3.18 สามารถอธิบายกระบวนการ Read และ write data ได้ดังนี้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



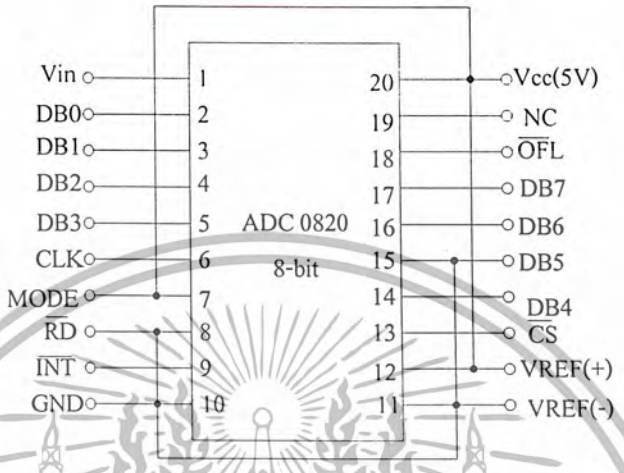
รูปที่ 3.19 โฟลว์ชาร์ตแสดงการทำงานของการแปลงเวฟเลต
ในส่วนของการ Simulateโดยใช้โปรแกรม Xilinx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA

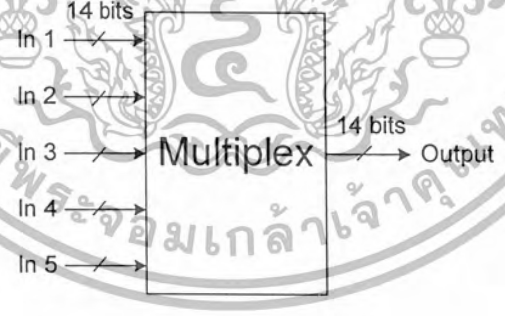
3.2.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter)

จะใช้ ไอซีเบอร์ ADC 0820 ซึ่งจะทำการแปลงระดับสัญญาณ ขนาด 8 บิท ระดับสัญญาณสามารถแปลงได้ ทั้ง สัญญาณบวก (สูงสุด +5 โวลท์) และสัญญาณลบ (ต่ำสุด -5 โวลท์) ดังวงจรข้างล่างนี้



รูปที่ 3.20 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter)

3.2.2 วงจรมัลติเพล็กซ์ (Multiplex) เป็นการมัลติเพล็กซ์สัมประสิทธิ์ที่ได้จากการแปลงเวฟเลต 3 ขั้นและอินพุตที่ได้จากการแซมปลิ้งออกไป



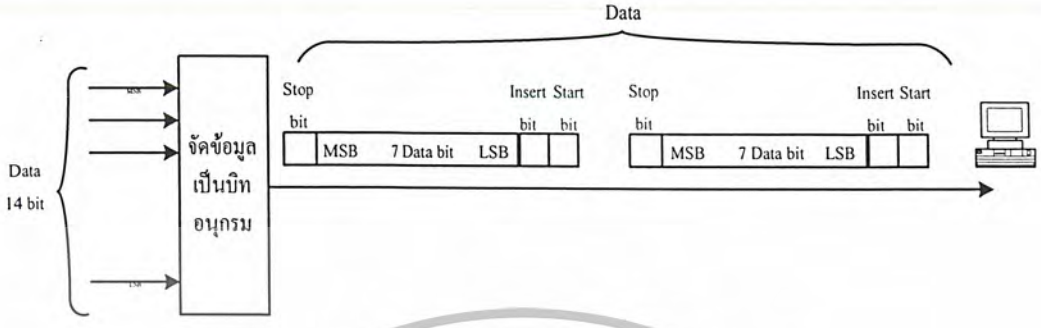
รูปที่ 3.21 บล็อกไดอะแกรมของวงจรมัลติเพล็กซ์

3.2.3 ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGAไปยังคอมพิวเตอร์

ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ ทำหน้าที่แปลงบิทข้อมูลแบบขนานขนาด 14 บิท ที่ได้จากแปลงเวฟเลตทรานสฟอร์ม ให้เป็นบิทข้อมูลแบบอนุกรม โดยแบ่งเป็น 2 เฟรม โดยแต่ละเฟรมข้อมูลประกอบด้วย สตาร์ทบิท(Start bit) 1 บิท , สตอบบิท(Stop bit) 1 บิท , บิทข้อมูล(Data bit) 7 บิท และแทรกบิท(Insert bit) 1 บิท เพื่อแก้ไขกรณีการส่งบิทบิทข้อมูลที่เป็น 0 ทั้งหมด หลังจากนั้นส่งเฟรมข้อมูลดังกล่าวออกพอร์ตอนุกรม(serial port) ไปยังคอมพิวเตอร์ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่บอดเรด(Baud Rate) สามารถแสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGAไปยังคอมพิวเตอร์ดังรูปที่ 3.22



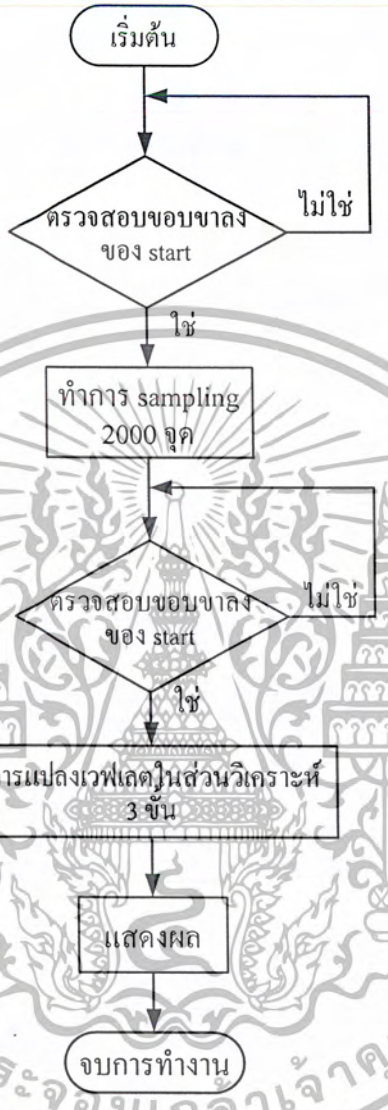
รูปที่ 3.22 แสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรม
จากบอร์ด FPGAไปยังคอมพิวเตอร์

3.2.4 ส่วนของการรับบิทข้อมูลจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ผ่านทางพอร์ตอนุกรม

ส่วนของการรับข้อมูลแบบอนุกรมจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ทำหน้าที่รับเฟรมบิทของข้อมมูลจากพอร์ตอนุกรมตามความถี่บอดเรด (Baud Rate) มาทำการตัดสตาร์ทบิท (Start bit) และสตอบบิท (Stop bit) และบิทแทรก (Insert bit) ทิ้ง พร้อมทั้งรวมเฟรมบิทข้อมูลที่แยกกันอยู่เพื่อให้ได้ค่าของข้อมูล 1 ค่า แล้วนำค่าที่ได้เก็บให้ครบทุกค่า สามารถแสดงบล็อกไดอะแกรมการรับบิทข้อมูลผ่านทางพอร์ตอนุกรมได้ดังรูปที่ 3.23

รูปที่ 3.23 แสดงบล็อกไดอะแกรมการรับบิทข้อมูลจากบอร์ด FPGA ผ่านพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 โฟลว์ชาร์ตแสดงการทำงานของการแปลงเวฟเลต
ในส่วนของการ Implement ลงบนอุปกรณ์ FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

การออกแบบส่วนต่างๆ ของการแปลงเวฟเลต โดยใช้โครงสร้างของฟิลเตอร์แบบโพลีเฟส (Polyphase Structure) สามารถแบ่งออกเป็น 4 ส่วน คือ

4.1 ส่วนของการ Simulate โดยใช้โปรแกรม MATLAB

4.2 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL

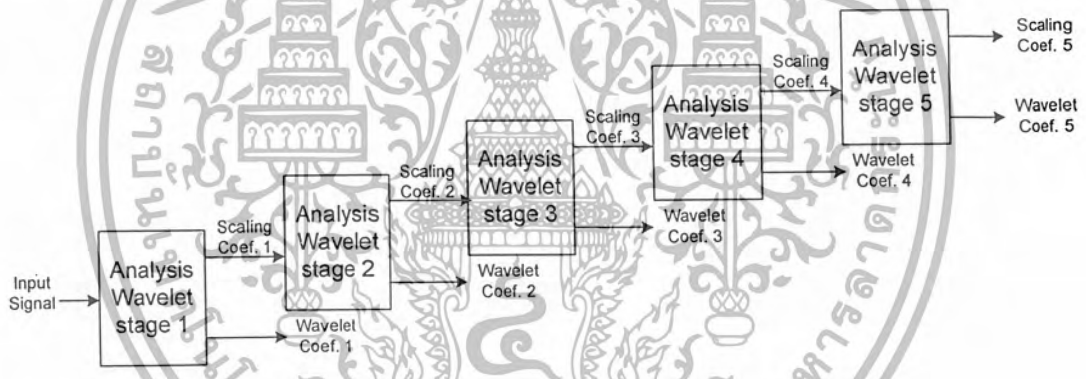
4.3 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA

4.4 ส่วนของการแสดงผลผ่านจอคอมพิวเตอร์

4.5 ภาพอุปกรณ์ที่ใช้ในการทดสอบการแปลงเวฟเลต

4.1 ส่วนของการ Simulate โดยใช้โปรแกรม MATLAB จะแบ่งออกเป็น 2 ส่วน คือ

4.1.1 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ (Analysis Wavelet Transform) โดยจะทำการแปลงเวฟเลตทั้งหมด 5 ขั้น ดังรูป

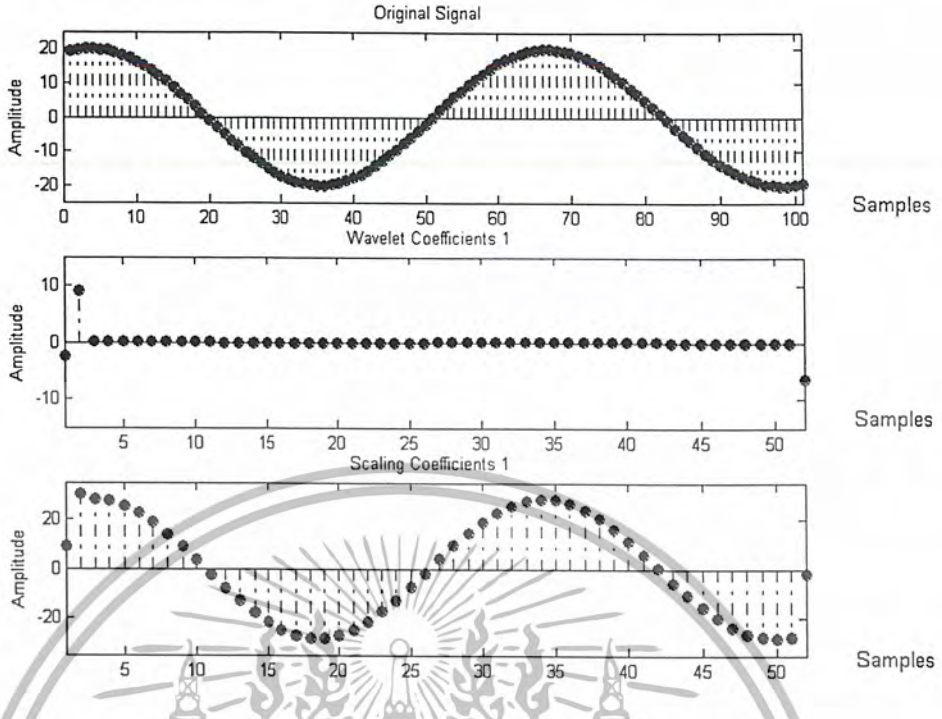


รูปที่ 4.1 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านวิเคราะห์

4.1.1.1 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 1

ในขั้นแรกทำการป้อนข้อมูลเป็นสัญญาณดิสครีตซายน์ (Discrete Sine Signal) ขนาด 100 จุดในการแปลงเวฟเลตโดยเลือกใช้สัมประสิทธิ์เวฟเลตดาร์บีชี 4 ค่า เก็บผลการทดลองค่าสัมประสิทธิ์สเกลลิ่ง 1 และสัมประสิทธิ์เวฟเลต 1 ที่ได้ ดังรูปที่ 4.2

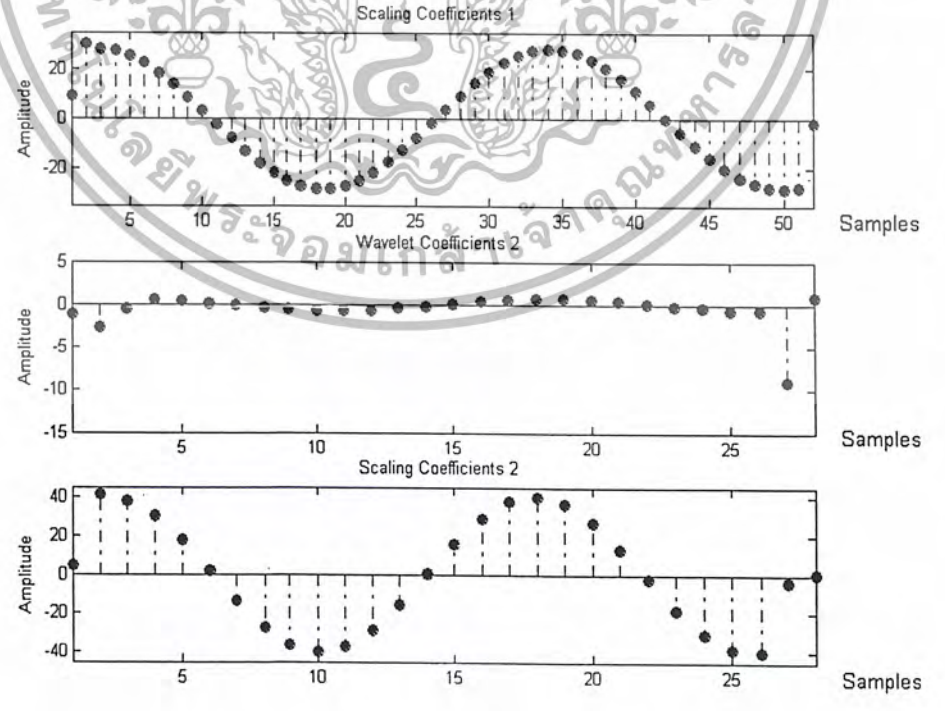
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 1

4.1.1.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 2

ส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 1 ที่ได้จากการแปลงเวฟเลตในขั้นแรก มาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 2 และสัมประสิทธิ์เวฟเลต 2 ดังรูปที่ 4.3

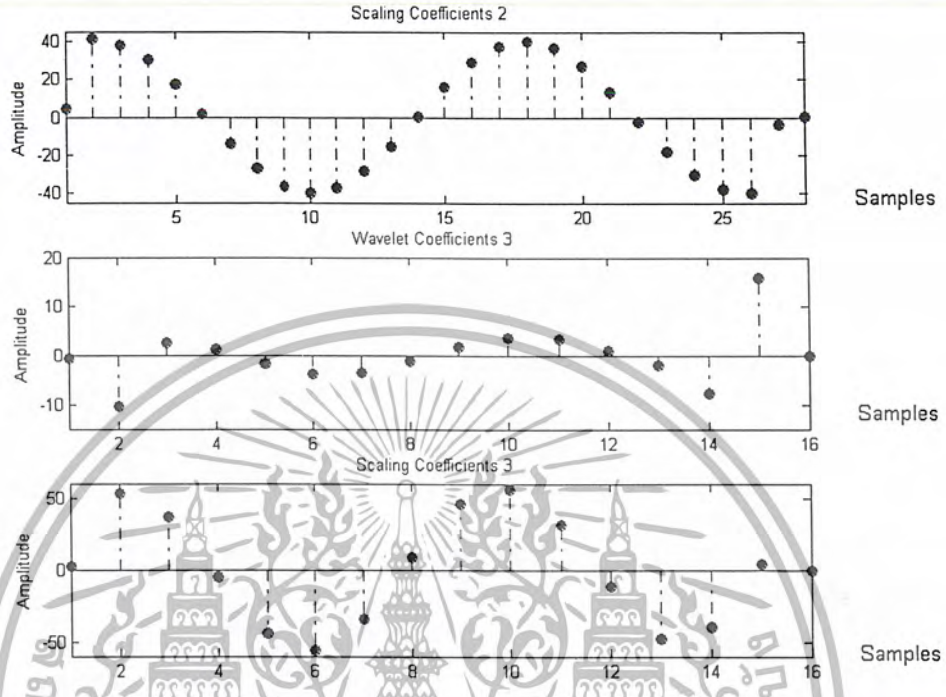


รูปที่ 4.3 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1.3 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 3

ส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 2 ที่ได้จากการแปลงเวฟเลตในขั้นที่สอง มาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 3 และสัมประสิทธิ์เวฟเลต 3 ดังรูปที่ 4.4

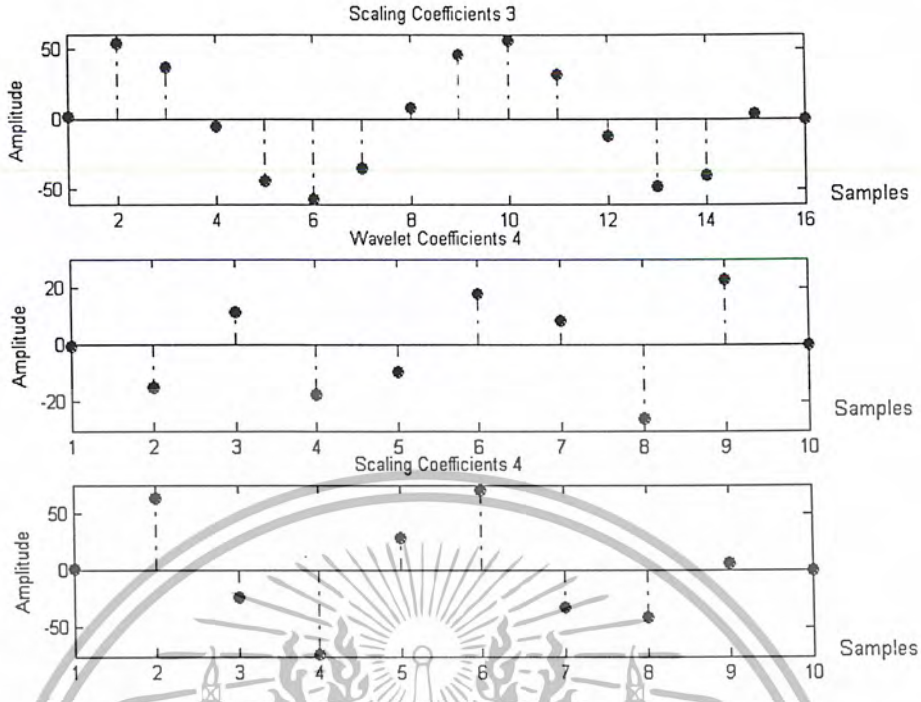


รูปที่ 4.4 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 3

4.1.1.4 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 4

ส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 3 ที่ได้จากการแปลงเวฟเลตในขั้นที่สาม มาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 4 และสัมประสิทธิ์เวฟเลต 4 ดังรูปที่ 4.5

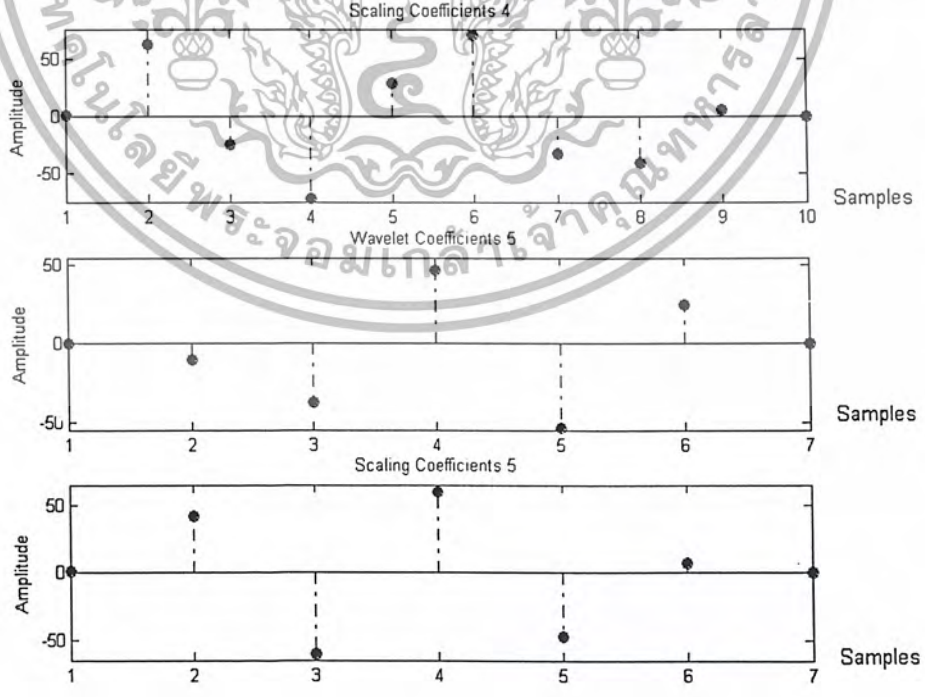
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 4

4.1.1.5 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 5

ส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 4 ที่ได้จากการแปลงเวฟเลตในขั้นที่สี่ มาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 5 และสัมประสิทธิ์เวฟเลต 5 ดังรูปที่ 4.6

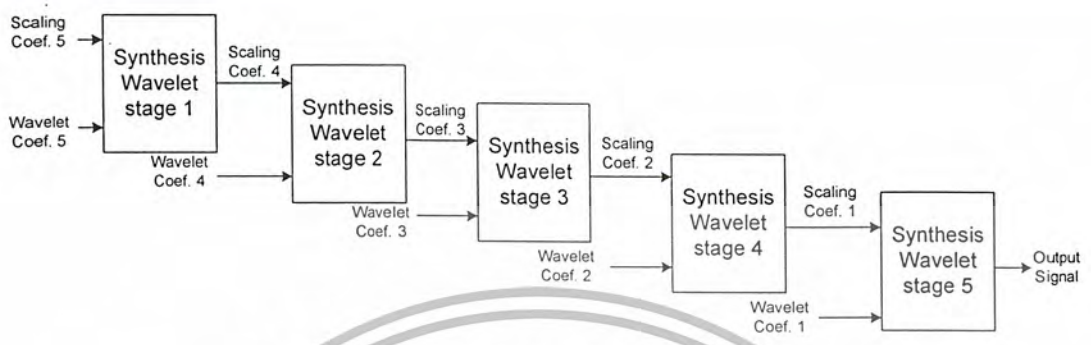


รูปที่ 4.6 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ (Synthesis Wavelet Transform)

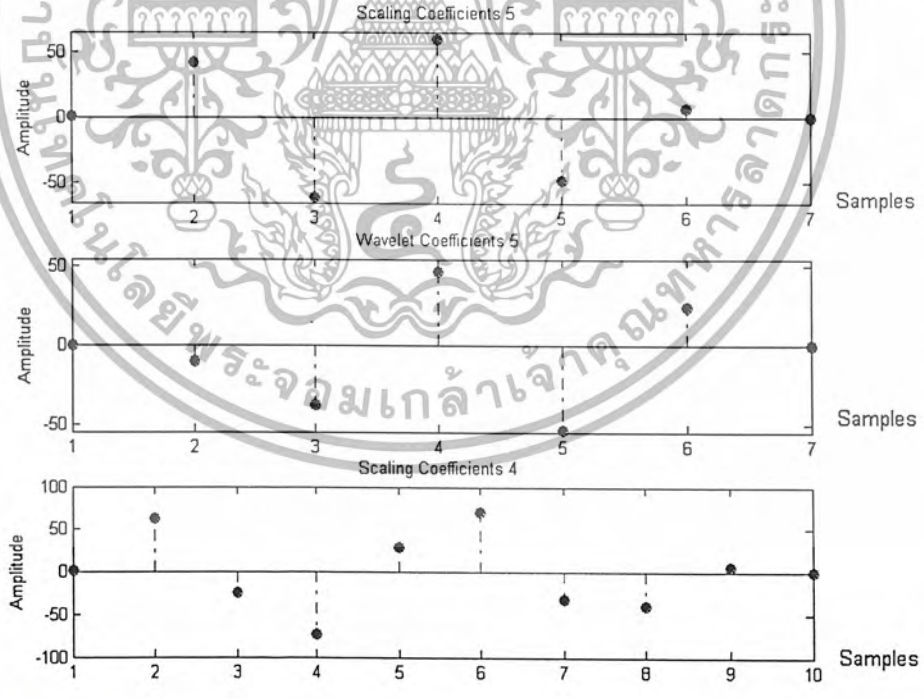
ในส่วนนี้จะประกอบด้วยการแปลงเวฟเลตกลับทั้งหมด 5 ขั้นตอน เช่นเดียวกับทางด้านวิเคราะห์ ดังรูปที่ 4.7



รูปที่ 4.7 บล็อกไดอะแกรมของการแปลงเวฟเลตทางด้านสังเคราะห์

4.1.2.1 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 1

ในส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 5 และสัมประสิทธิ์เวฟเลต 5 ที่ได้จากการแปลงเวฟเลตในส่วนวิเคราะห์ มาทำการแปลงกลับให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 4 ดังรูปที่ 4.8

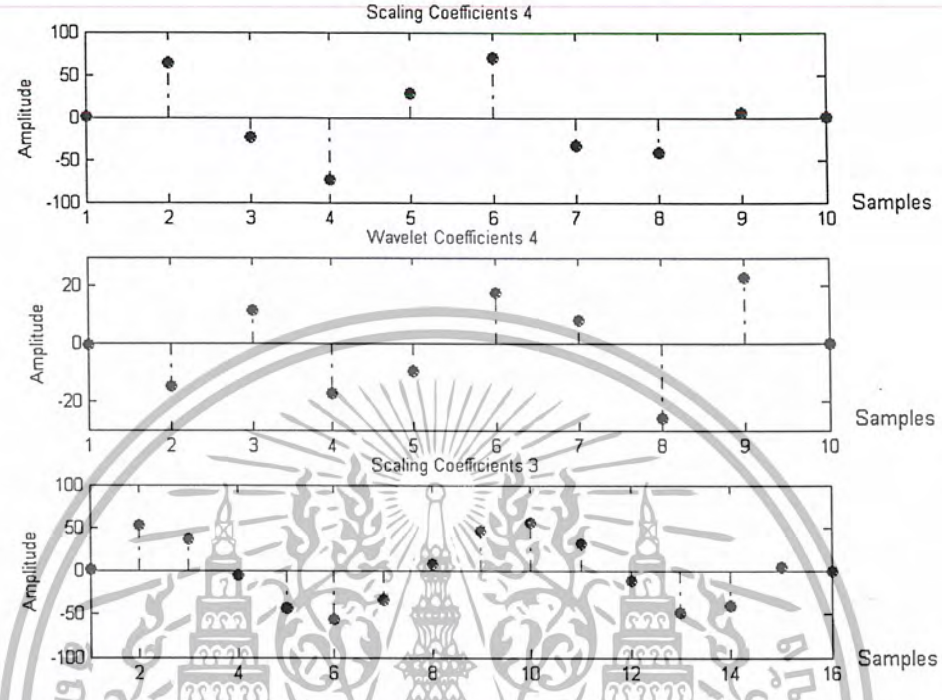


รูปที่ 4.8 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2.2 ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ขั้นที่ 2

ในส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 4 และสัมประสิทธิ์เวฟเลต 4 ที่ได้จากการแปลงเวฟเลตในส่วนวิเคราะห์ มาทำการแปลงกลับให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 3 ดังรูปที่ 4.9

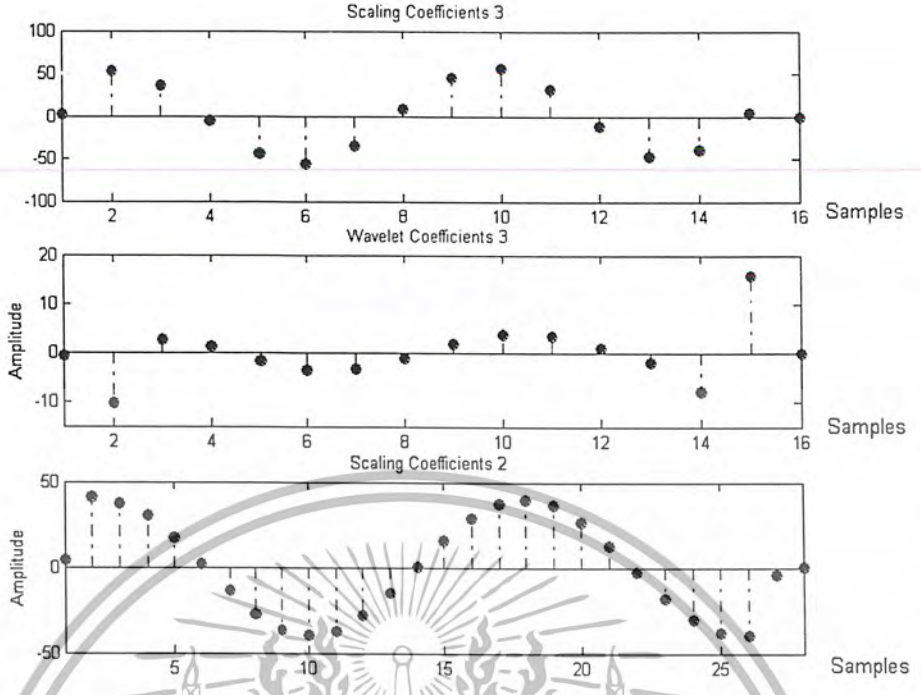


รูปที่ 4.9 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 2

4.1.2.3 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 3

ในส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 3 และสัมประสิทธิ์เวฟเลต 3 ที่ได้จากการแปลงเวฟเลตในส่วนวิเคราะห์ มาทำการแปลงกลับให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 2 ดังรูปที่ 4.10

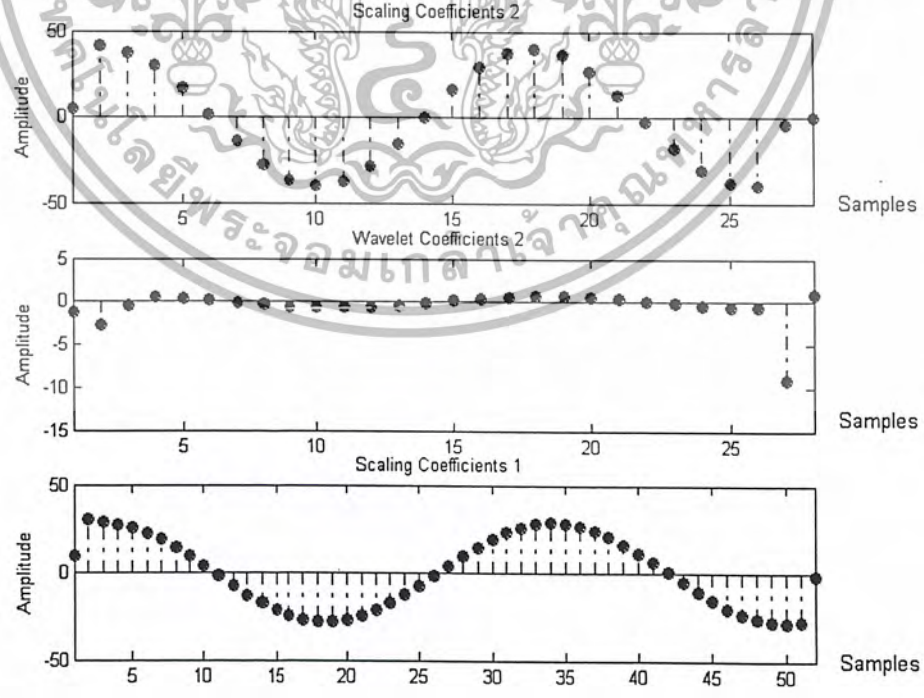
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 3

4.1.2.4 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 4

ในส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 2 และสัมประสิทธิ์เวฟเลต 2 ที่ได้จากการแปลงเวฟเลตในส่วนวิเคราะห์ มาทำการแปลงกลับให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง 1 ดังรูปที่ 4.11

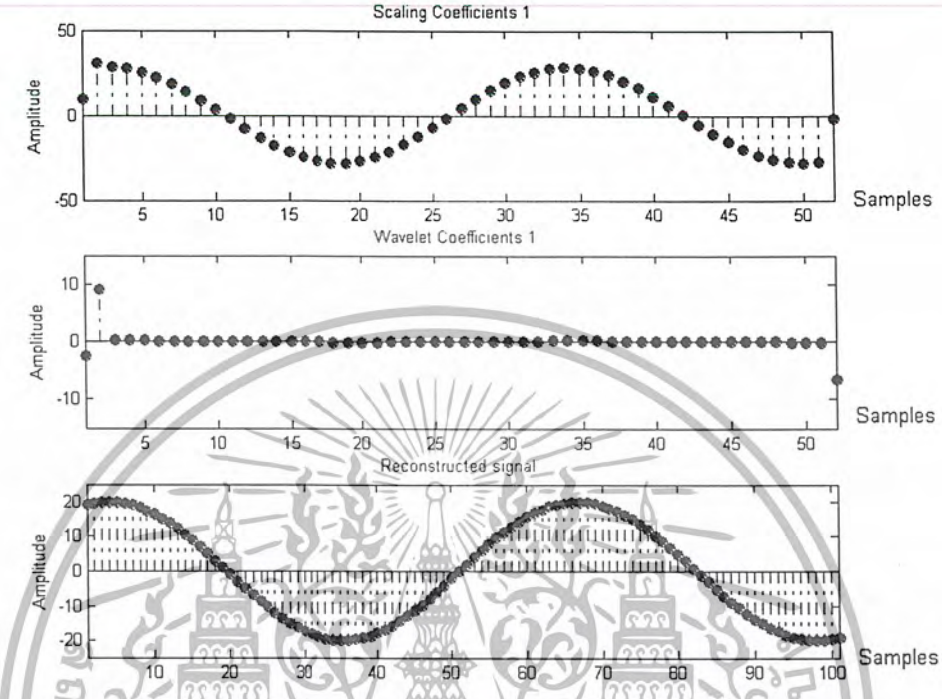


รูปที่ 4.11 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2.5 ส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 5

ในส่วนนี้จะนำค่าสัมประสิทธิ์สเกลลิ่ง 1 และสัมประสิทธิ์เวฟเลต 1 ที่ได้จากการแปลงเวฟเลตในส่วนวิเคราะห์ มาทำการแปลงกลับให้อยู่ในรูปของสัญญาณกู้กลับ ดังรูปที่ 4.12



รูปที่ 4.12 แสดงผลการจำลองการทำงานของการแปลงเวฟเลตทางด้านสังเคราะห์ขั้นที่ 5

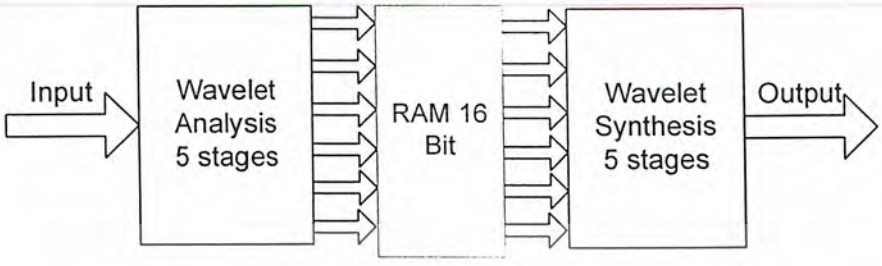
จากการเปรียบเทียบสัญญาณต้นแบบกับสัญญาณกู้กลับ นำผลลัพธ์ที่ได้จากการแปลงเวฟเลตทางด้านสังเคราะห์ไปหาค่าความผิดพลาดโดยเทียบกับข้อมูลต้นแบบ ได้ค่าความผิดพลาดจากการแปลงเวฟเลต (Mean Square Error) ดังนี้

$$MSE = 1.3914 \times 10^{-30}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL

ในส่วนนี้จะแบ่งออกเป็น 3 ส่วน คือ ส่วนของการแปลงเวฟเลตทางด้านวิเคราะห์ ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล และส่วนของการแปลงเวฟเลตทางด้านสังเคราะห์ ดังรูปที่ 4.13



รูปที่ 4.13 รูปบล็อกไดอะแกรมในส่วนของการ Simulate ด้วยโปรแกรม Xilinx-Project Navigator

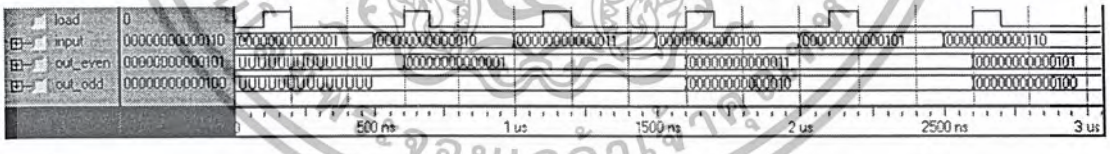
4.2.1 ส่วนของการแปลงเวฟเลตในส่วนวิเคราะห์ (Analysis Wavelet Transform)

ในส่วนนี้จะใช้โครงสร้างของฟิลเตอร์แบบโพลีเฟส โดยมีส่วนประกอบภายในดังนี้

4.2.1.1 ส่วนของวงจรสุ่มค่าตัวอย่าง (Down-Sampling)

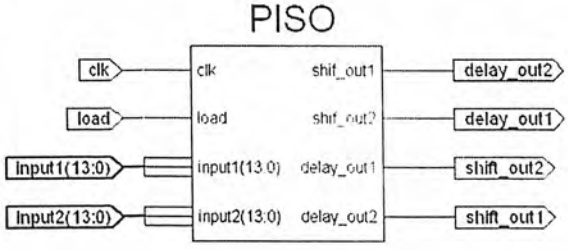
รูปที่ 4.14 แสดงสัญลักษณ์ของวงจรสุ่มค่าตัวอย่าง

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.15



รูปที่ 4.15 แสดงผลการจำลองการทำงานของวงจรสุ่มค่าตัวอย่าง

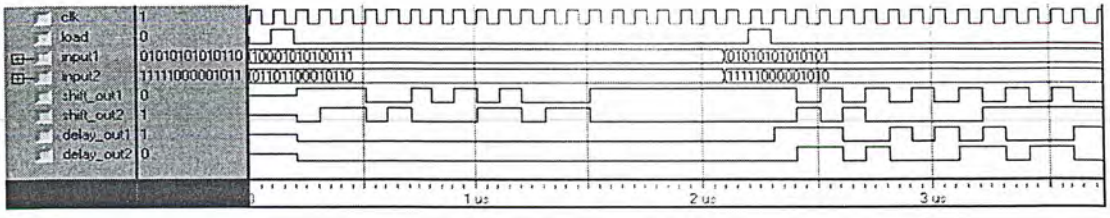
4.2.1.2 วงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม (Parallel in Serial out (PISO))



รูปที่ 4.16 แสดงสัญลักษณ์ของวงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.17



รูปที่ 4.17 แสดงผลการจำลองการทำงานของวงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม

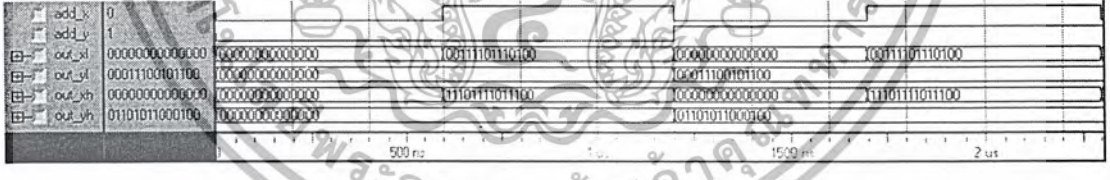
4.2.1.3 หน่วยความจำ (ROM)

ในส่วนของการแปลงเวฟเลตทางฝั่งวิเคราะห์จะประกอบด้วยหน่วยความจำ 2 ตัว คือ ROM phase EVEN และ ROM phase ODD ดังนี้

4.2.1.3.1 หน่วยความจำเฟสคู่

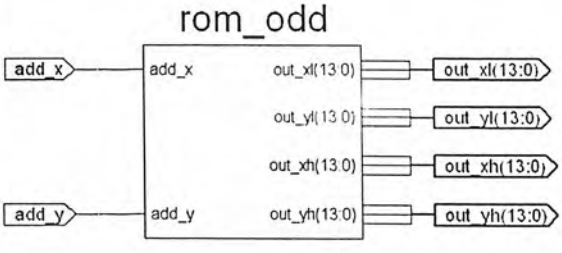
รูปที่ 4.18 แสดงสัญลักษณ์ของหน่วยความจำเฟสคู่

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.19



รูปที่ 4.19 แสดงผลการจำลองการทำงานของหน่วยความจำเฟสคู่

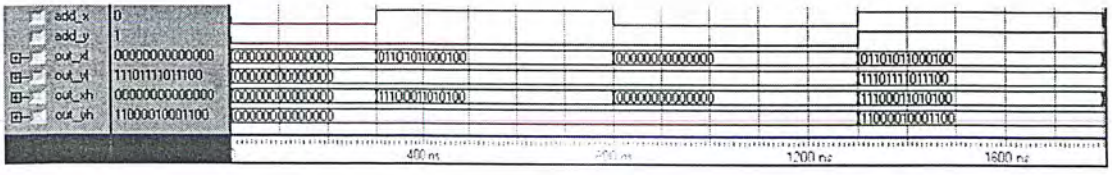
4.2.1.3.2 หน่วยความจำเฟสคี่



รูปที่ 4.20 แสดงสัญลักษณ์ของหน่วยความจำเฟสคี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

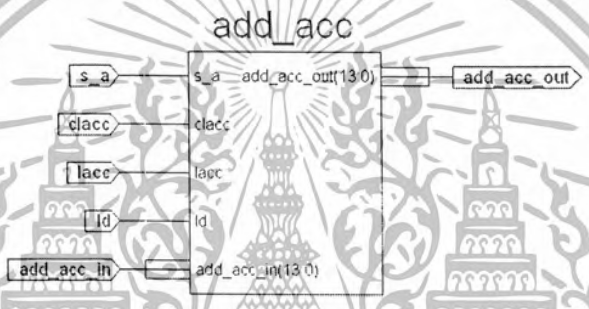
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.21



รูปที่ 4.21 แสดงผลการจำลองการทำงานของหน่วยความจำเฟสคี่

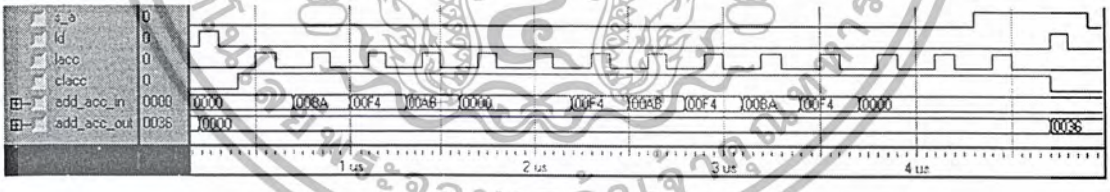
4.2.1.4 ส่วนของวงจรบวกสะสม 14 บิท

เป็นส่วนที่อยู่ในวงจรแปลงเวฟเลต โดยรับอินพุทมาจากส่วนของวงจรเก็บค่าสัมประสิทธิ์ มาทำการบวกสะสมตามหลักการของโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรบวกสะสมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.22



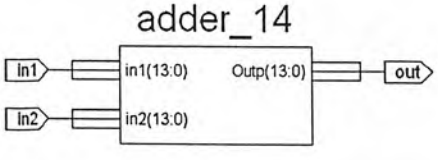
รูปที่ 4.22 แสดงสัญลักษณ์ของวงจรบวกสะสม 14 บิท

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.23



รูปที่ 4.23 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 14 บิท

4.2.1.5 ส่วนของวงจรบวกขนาด 14 บิท



รูปที่ 4.24 แสดงสัญลักษณ์ของวงจรบวกขนาด 14 บิท

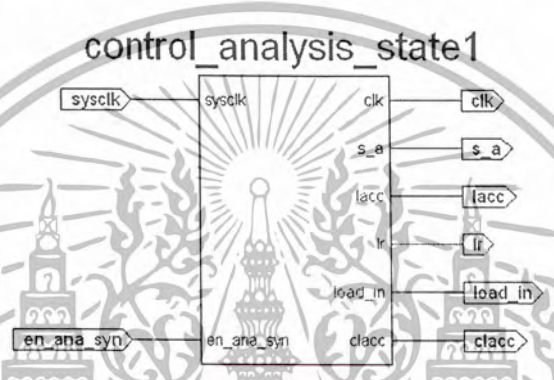
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.25

รูปที่ 4.25 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 14 บิท

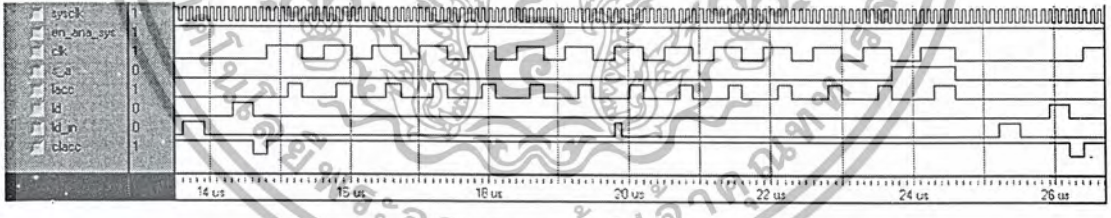
4.2.1.6 ส่วนของวงจรควบคุมและวงจรแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1

ในส่วนนี้มีหน้าที่ในการนำข้อมูลอินพุตมาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตอันดับที่ 1 โดยมีสัญลักษณ์ ดังรูปที่ 4.26

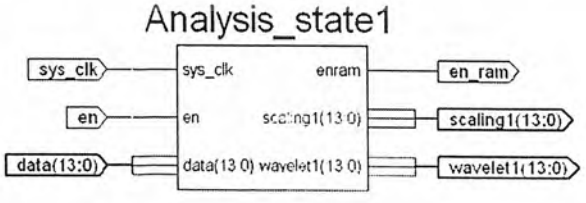


รูปที่ 4.26 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.27



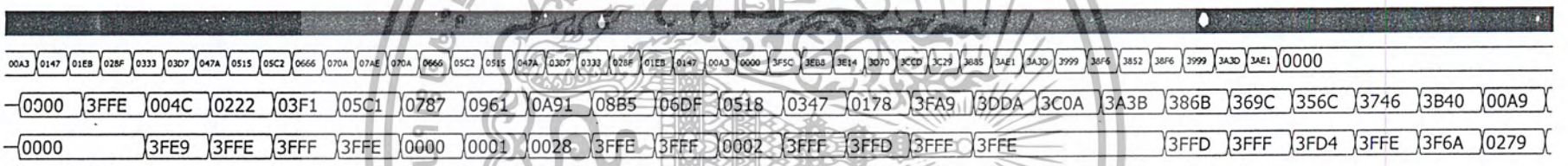
รูปที่ 4.27 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1



รูปที่ 4.28 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

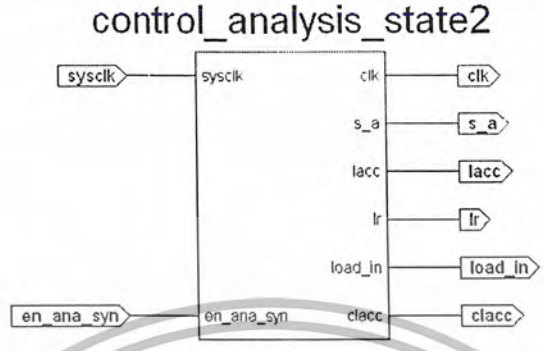


รูปที่ 4.29 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 1

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

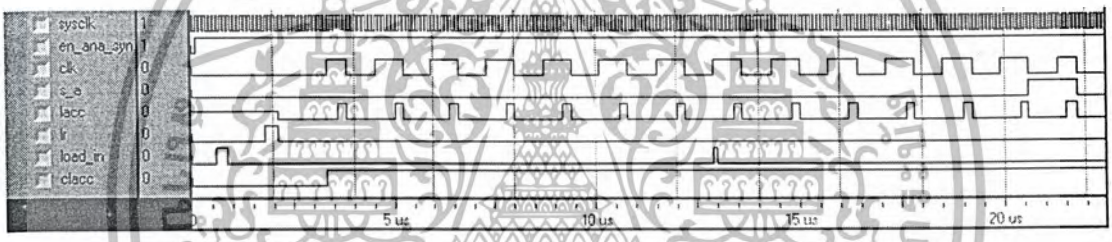
4.2.1.7 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2

ส่วนนี้จะทำหน้าที่ในการนำสัมประสิทธิ์สเกลลิ่งอันดับที่ 1 ที่ได้จากการแปลงเวฟเลตในขั้นแรก มาแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตอันดับที่ 2

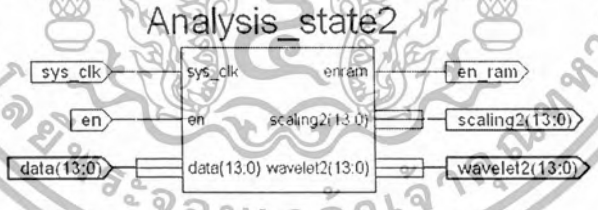


รูปที่ 4.30 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.31



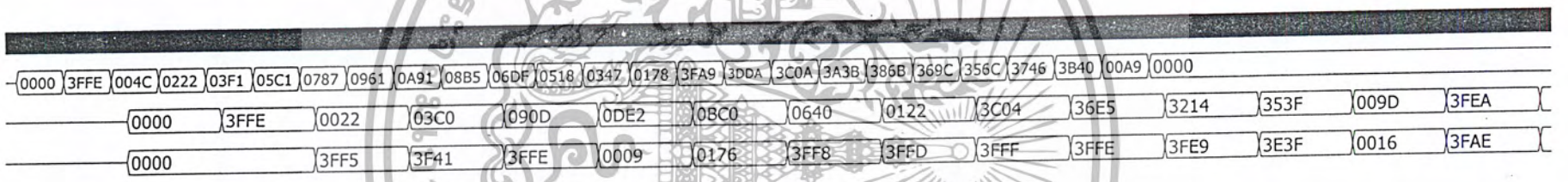
รูปที่ 4.31 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2



รูปที่ 4.32 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

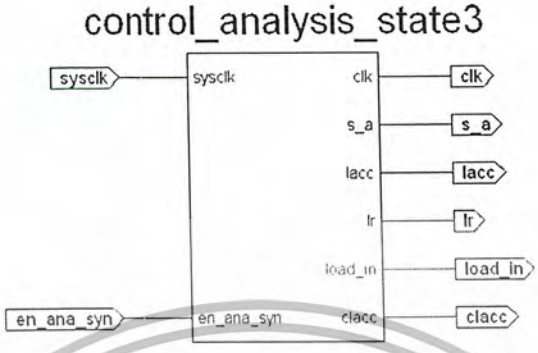


รูปที่ 4.33 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 2

ลิขสิทธิ์ของวิทยาลัยการทัพบก ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น การนำไปใช้ประโยชน์ด้านการศึกษาต้องได้รับอนุญาตเป็นลายลักษณ์อักษรจากวิทยาลัยการทัพบกก่อนเท่านั้น

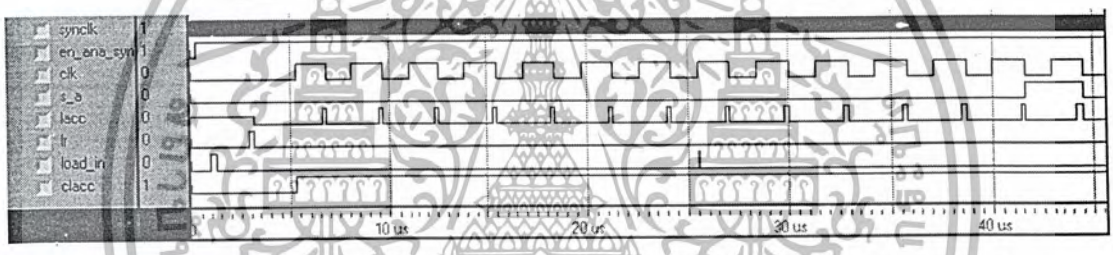
4.2.1.8 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3

ส่วนนี้จะนำสัมประสิทธิ์สเกลลิ่งอันดับ 2 ที่ได้จากการแปลงเวฟเลตในขั้นที่ 2 มาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตอันดับที่ 3

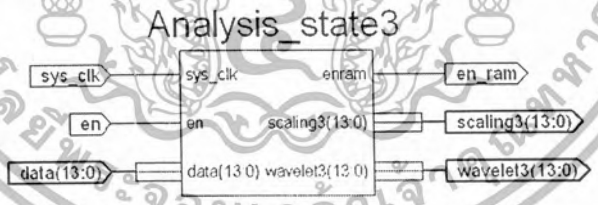


รูปที่ 4.34 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.35



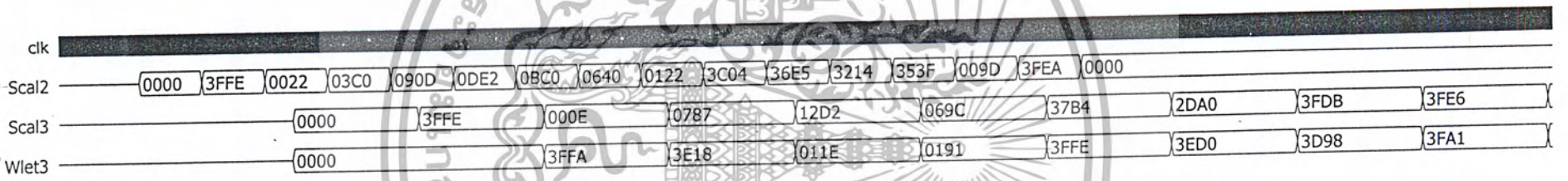
รูปที่ 4.35 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3



รูปที่ 4.36 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

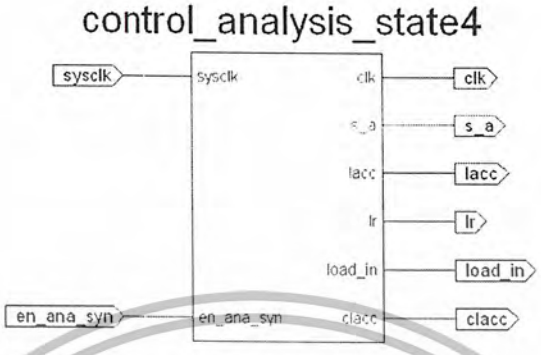


รูปที่ 4.37 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 3

ลิขสิทธิ์ของวิทยาลัยการทัพบก ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของลิขสิทธิ์ทุกครั้งที่มีการนำไปใช้

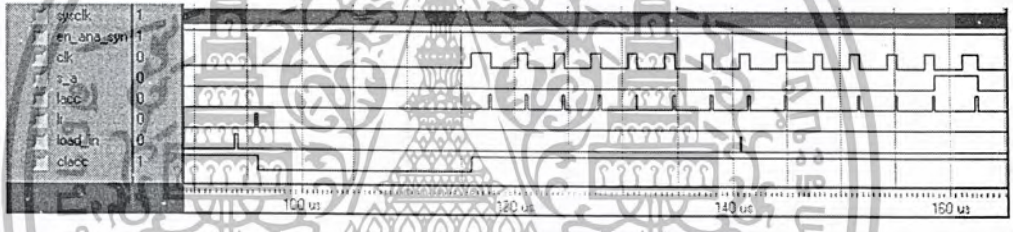
4.2.1.9 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4

ส่วนนี้จะนำสัมประสิทธิ์สเกลลิ่งอันดับ 3 ที่ได้จากการแปลงเวฟเลตในขั้นที่ 3 มาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตอันดับที่ 4

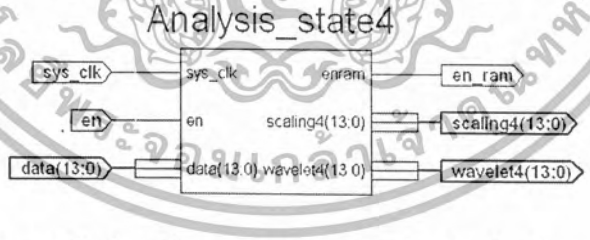


รูปที่ 4.38 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.39

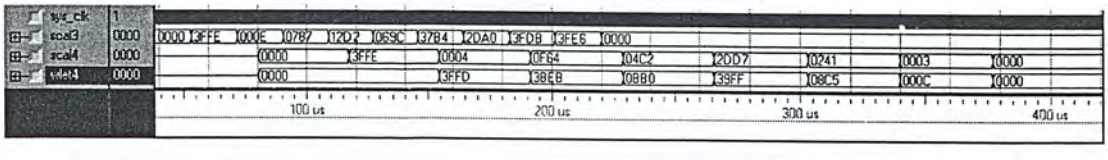


รูปที่ 4.39 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4



รูปที่ 4.40 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.41

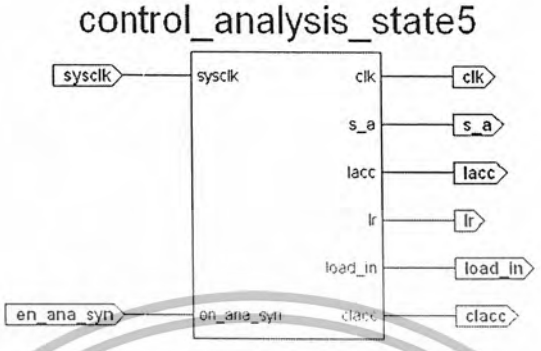


รูปที่ 4.41 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

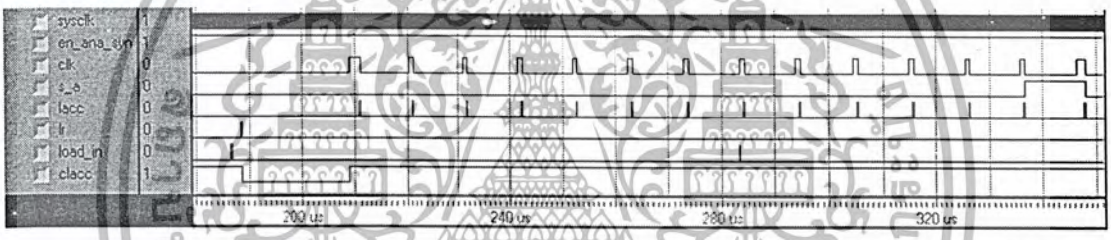
4.2.1.10 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5

ส่วนนี้จะนำสัมประสิทธิ์สเกลลิ่งอันดับ 4 ที่ได้จากการแปลงเวฟเลตในขั้นที่ 4 มาทำการแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตอันดับที่ 5

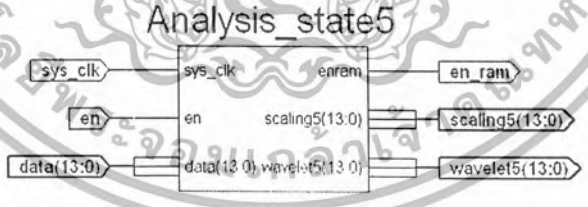


รูปที่ 4.42 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.43

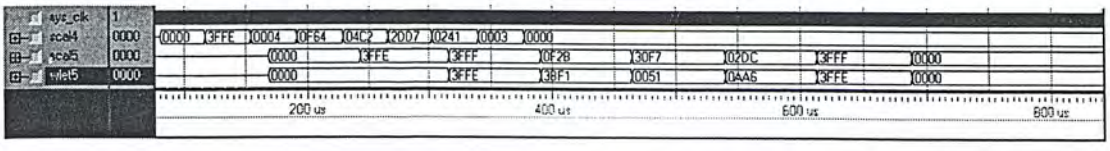


รูปที่ 4.43 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5



รูปที่ 4.44 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5

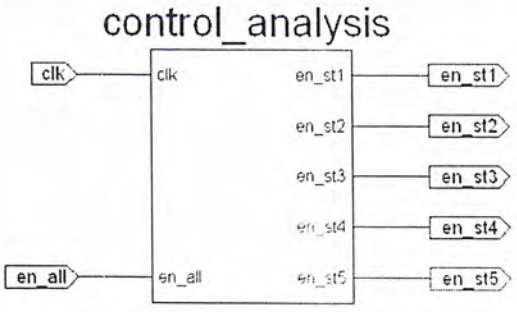
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.45



รูปที่ 4.45 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนวิเคราะห์ขั้นที่ 5

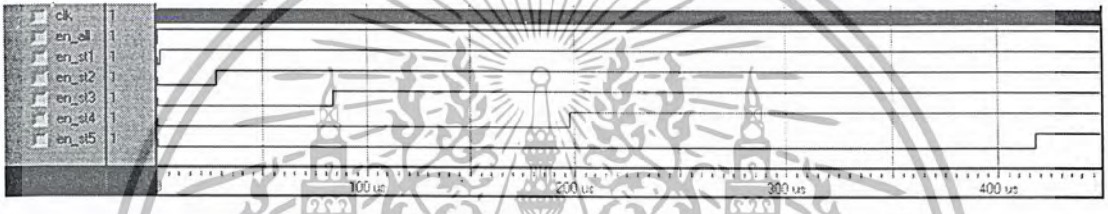
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.11 ส่วนของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนวิเคราะห์



รูปที่ 4.46 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนวิเคราะห์

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.47



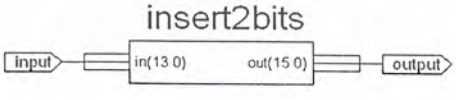
รูปที่ 4.47 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนวิเคราะห์

4.2.2 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล (RAM)

ในส่วนนี้จะประกอบด้วยส่วนของวงจรแทรกบิท วงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล และส่วนของวงจรควบคุมการเก็บค่าและเรียกค่าข้อมูลในแต่ละขั้นตอน รวมทั้งหมด 5 ขั้น ดังนี้

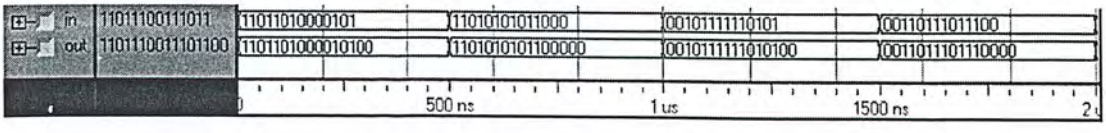
4.2.2.1 ส่วนของวงจรแทรกบิทจำนวน 2 บิท

เนื่องจากในการแปลงเวฟเลต 1 ค่า จะแทนด้วยไบนารีจำนวน 14 บิท แต่ข้อมูลที่จะทำการบันทึกลงใน RAM จะต้องมีค่า 16 บิท จึงต้องทำการแทรกบิทจำนวน 2 บิทต่อไปส่วนท้ายของข้อมูล 14 บิท ก่อนป้อนข้อมูลเข้าสู่ RAM



รูปที่ 4.48 แสดงสัญลักษณ์ของวงจรแทรกบิท 2 บิท

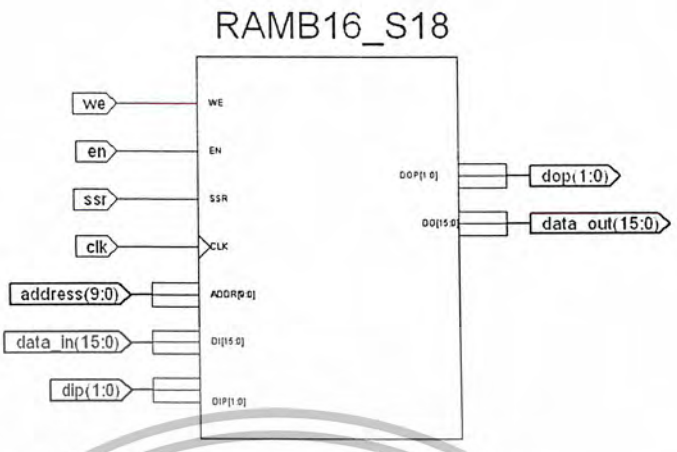
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.49



รูปที่ 4.49 แสดงผลการจำลองการทำงานของวงจรแทรกบิท 2 บิท

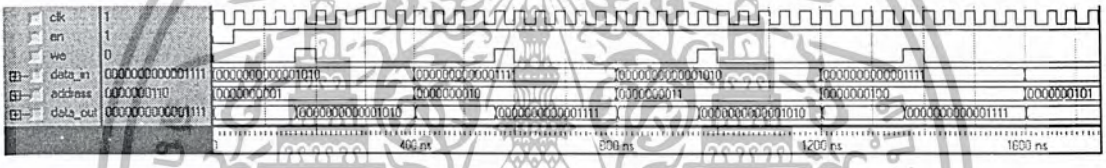
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.2 วงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 16 บิท (RAM 16 bit)



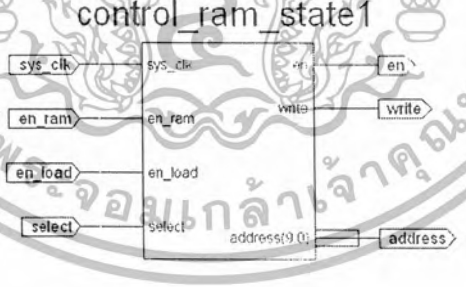
รูปที่ 4.50 แสดงสัญลักษณ์ของวงจรRAM 16 bit

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.51



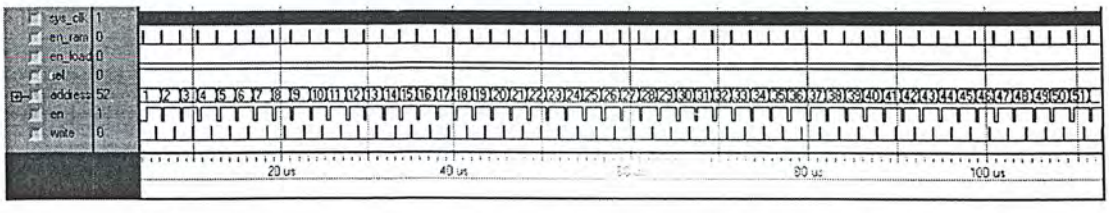
รูปที่ 4.51 แสดงผลการจำลองการทำงานของวงจร RAM

4.2.2.3 ส่วนของวงจรควบคุมตัวเก็บค่าสัมประสิทธิ์ในแต่ละขั้น



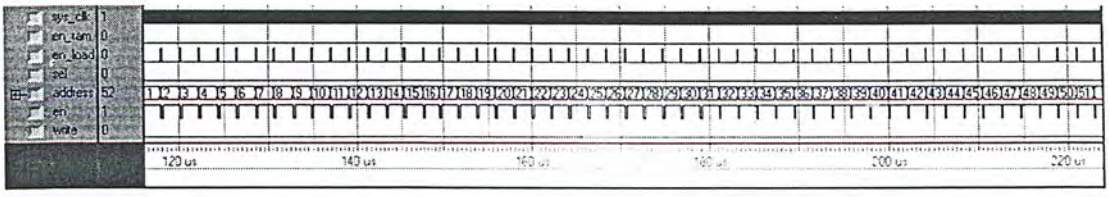
รูปที่ 4.52 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 1

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.53 และ 4.54

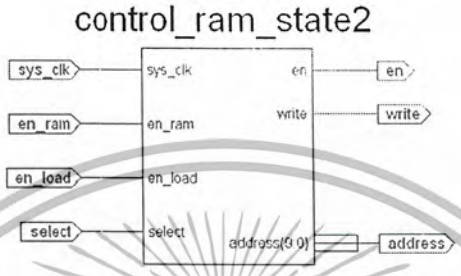


รูปที่ 4.53 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 1 (ช่วงที่ 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

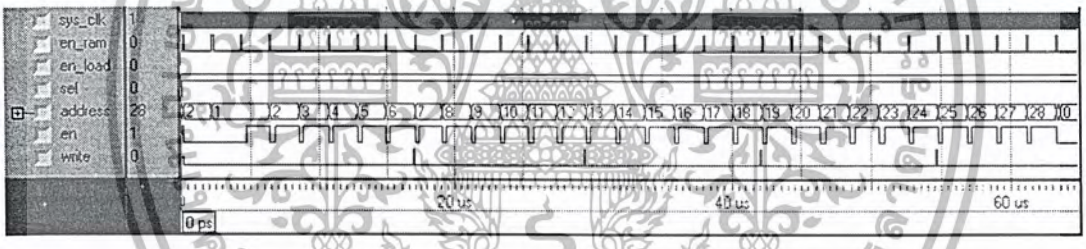


รูปที่ 4.54 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 1 (ช่วงที่ 2)

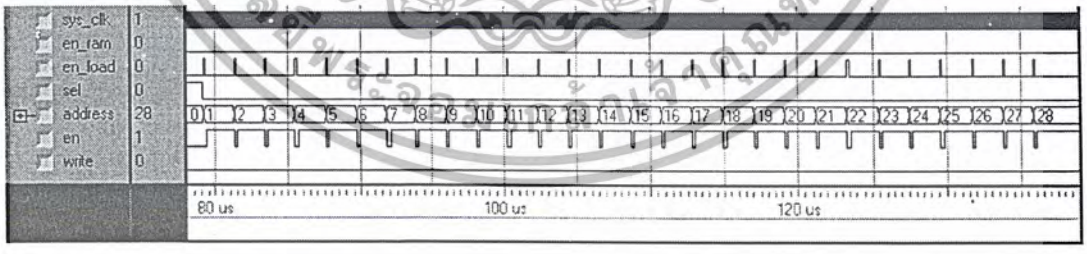


รูปที่ 4.55 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 2

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.56 และ 4.57

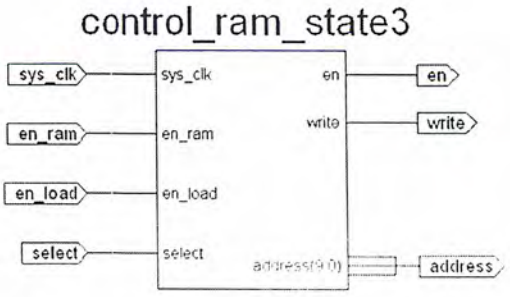


รูปที่ 4.56 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 2 (ช่วงที่ 1)



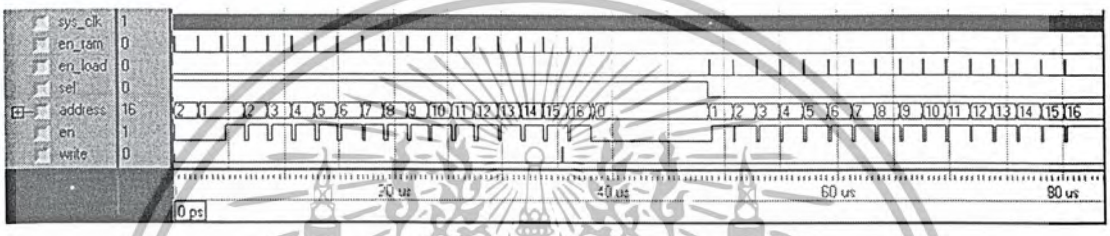
รูปที่ 4.57 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 2 (ช่วงที่2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

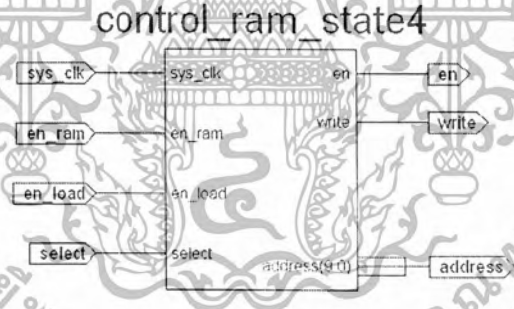


รูปที่ 4.58 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 3

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.59

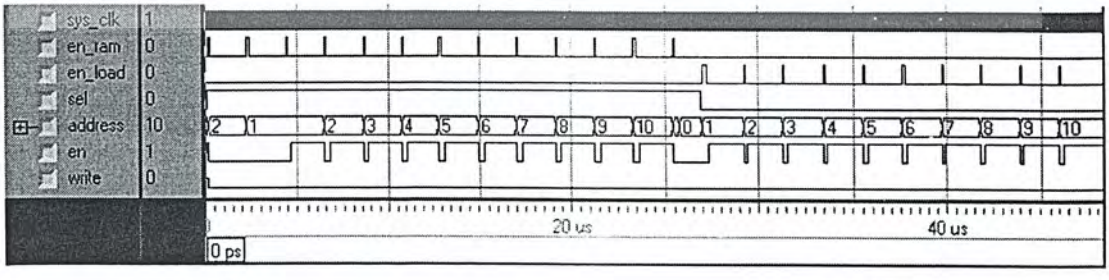


รูปที่ 4.59 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 3



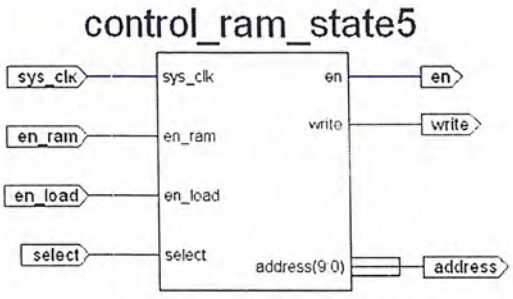
รูปที่ 4.60 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 4

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.61



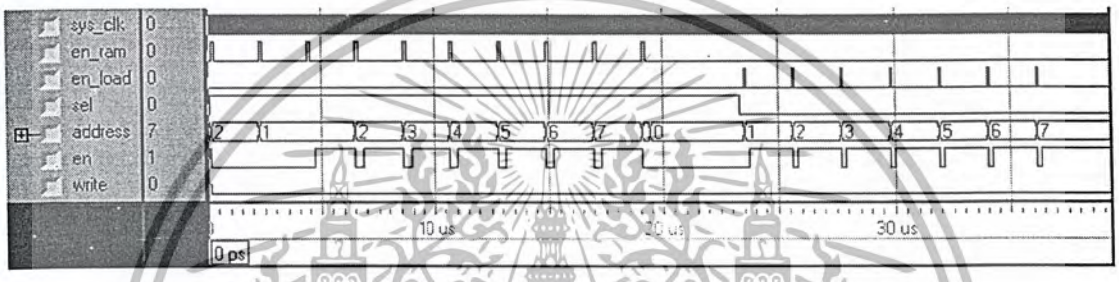
รูปที่ 4.61 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



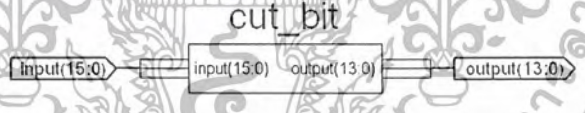
รูปที่ 4.62 แสดงสัญลักษณ์ของวงจรควบคุม RAM ขั้นที่ 5

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.63



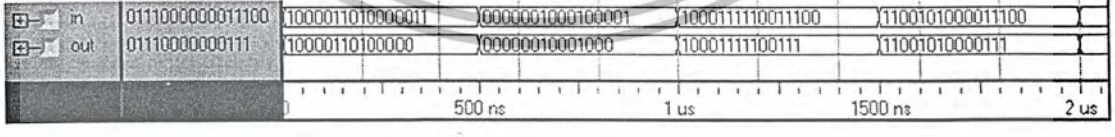
รูปที่ 4.63 แสดงผลการจำลองการทำงานของวงจรควบคุม RAM ขั้นที่ 5

4.2.2.4 ส่วนของวงจรตัดบิท 2 บิท



รูปที่ 4.64 แสดงสัญลักษณ์ของวงจรตัดบิท 2 บิท

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.65



รูปที่ 4.65 แสดงผลการจำลองการทำงานของวงจรตัดบิท 2 บิท

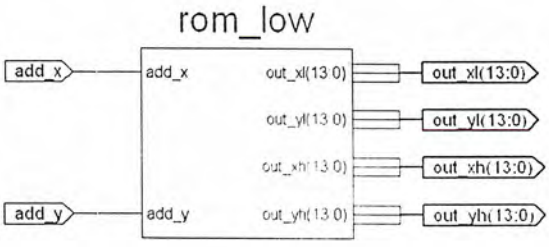
4.2.3 ส่วนของการแปลงเวฟเลตในส่วนสังเคราะห์ (Synthesis Wavelet Transform)

ในส่วนนี้จะประกอบด้วยส่วนของวงจรเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม ส่วนของหน่วยความจำ ส่วนของวงจรบวกสะสม 14 บิท ส่วนของวงจรบวกขนาด 14 บิท ส่วนของวงจรเพิ่มค่าตัวอย่าง และส่วนของวงจรควบคุม (controller) ซึ่งจะคล้ายคลึงกับส่วนของการแปลงเวฟเลตในส่วนวิเคราะห์ แต่จะมีความแตกต่างกันในส่วนของหน่วยความจำ และส่วนของการเพิ่มค่าตัวอย่าง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3.1 หน่วยความจำ

4.2.3.1.1 หน่วยความจำของวงจรกรองความถี่ต่ำ

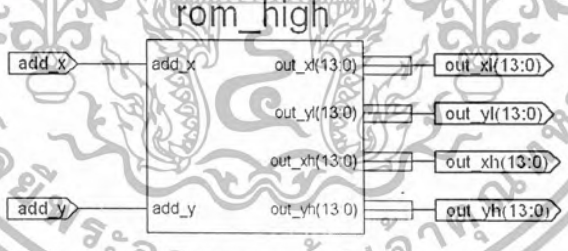


รูปที่ 4.66 แสดงสัญลักษณ์ของวงจรROM_LOW

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.67

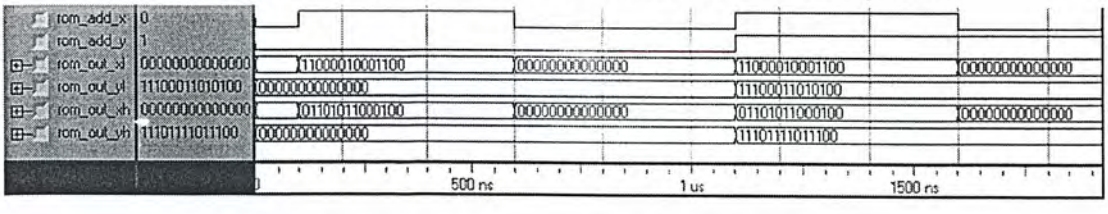
รูปที่ 4.67 แสดงผลการจำลองการทำงานของวงจรROM_LOW

4.2.3.1.2 หน่วยความจำของวงจรกรองความถี่สูง



รูปที่ 4.68 แสดงสัญลักษณ์ของวงจรROM_HIGH

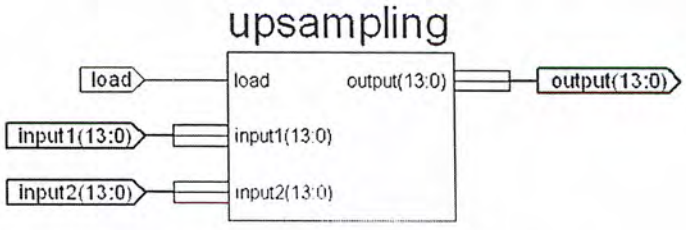
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.69



รูปที่ 4.69 แสดงผลการจำลองการทำงานของวงจรROM_HIGH

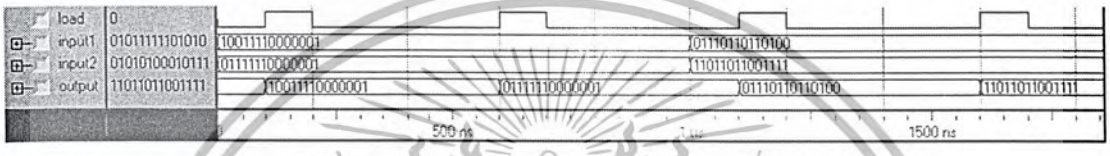
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3.2 ส่วนของวงจรเพิ่มค่าตัวอย่าง (Up-Sampling)



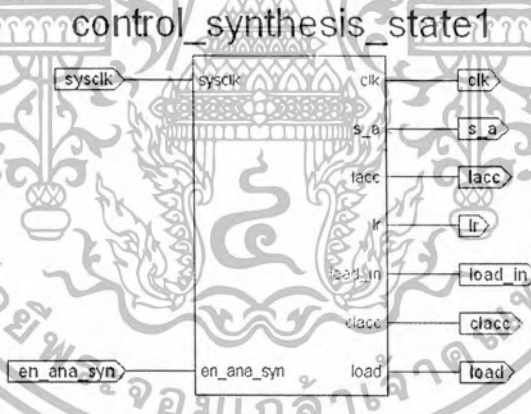
รูปที่ 4.70 แสดงสัญลักษณ์ของวงจรเพิ่มค่าตัวอย่าง

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.71



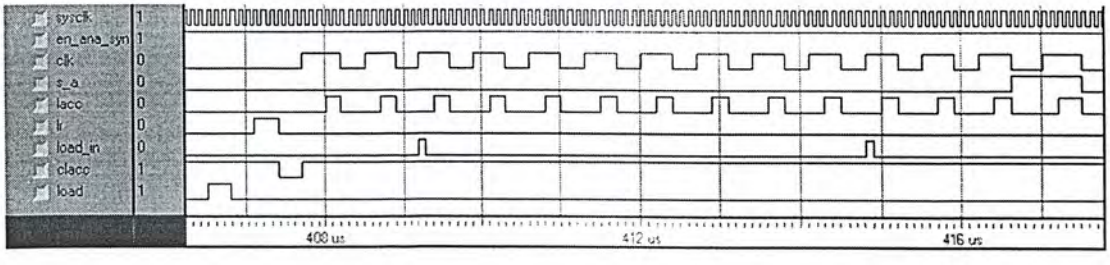
รูปที่ 4.71 แสดงผลการจำลองการทำงานของวงจรเพิ่มค่าตัวอย่าง

4.2.3.3 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1



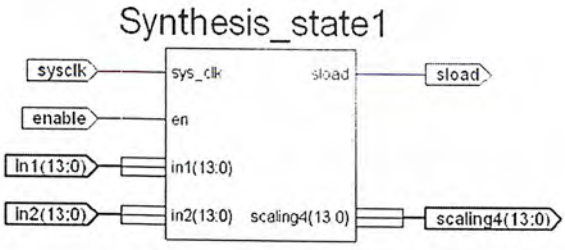
รูปที่ 4.72 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.73



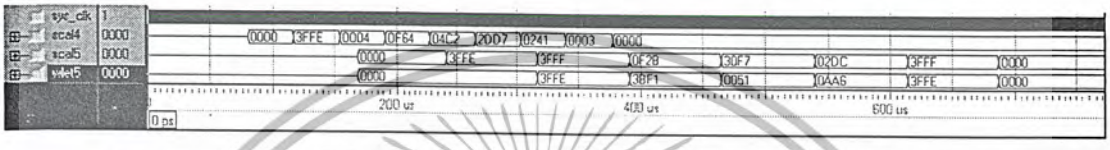
รูปที่ 4.73 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



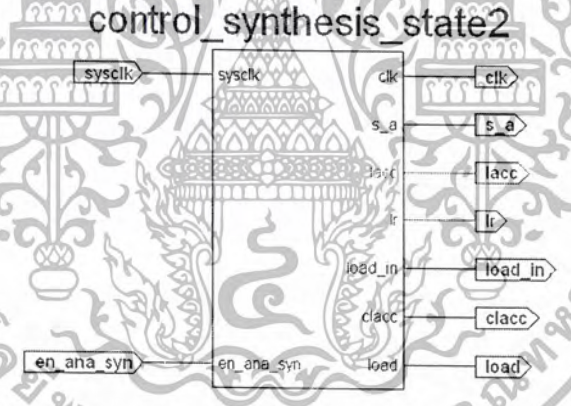
รูปที่ 4.74 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.75



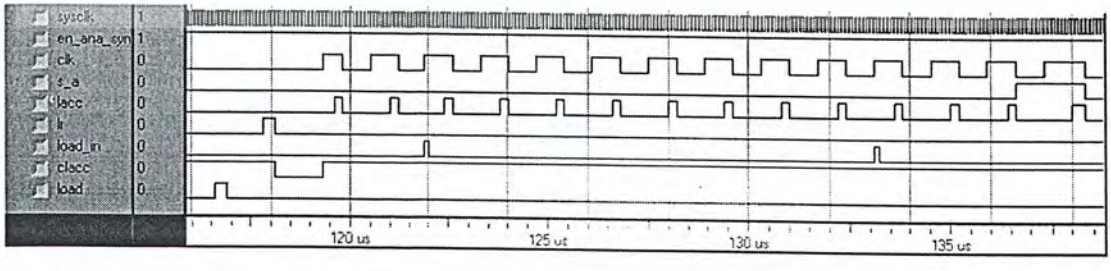
รูปที่ 4.75 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 1

4.2.3.4 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2



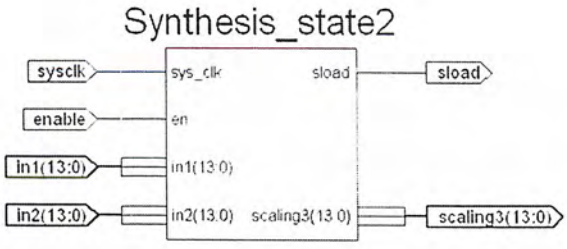
รูปที่ 4.76 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.77



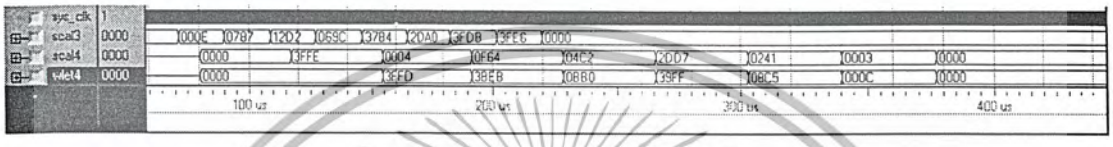
รูปที่ 4.77 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



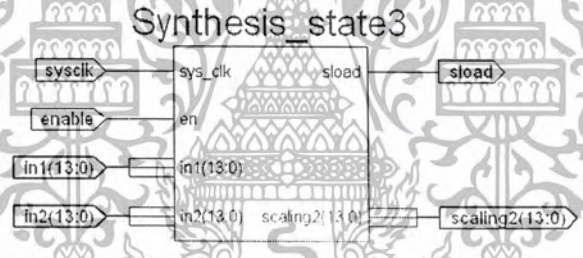
รูปที่ 4.78 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.79



รูปที่ 4.79 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 2

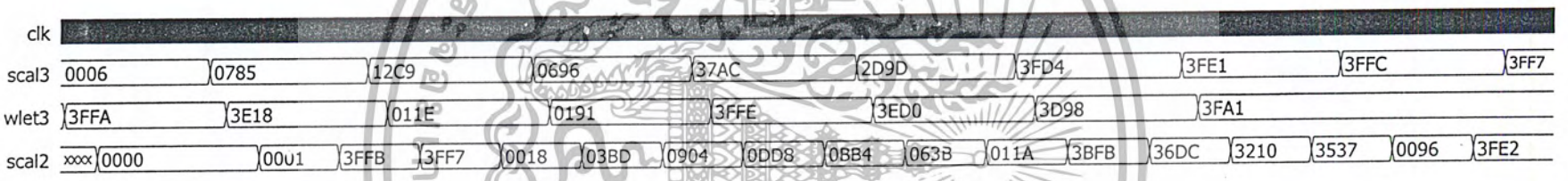
4.2.3.5 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3



รูปที่ 4.80 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3

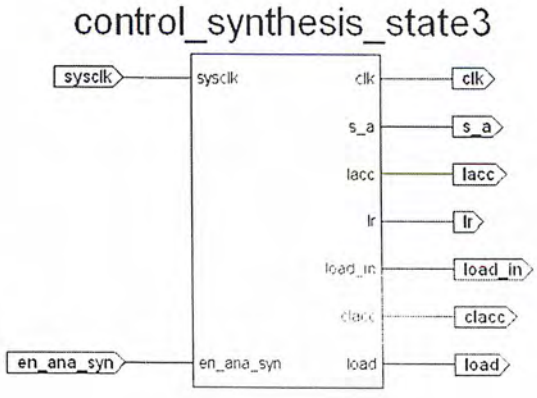
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



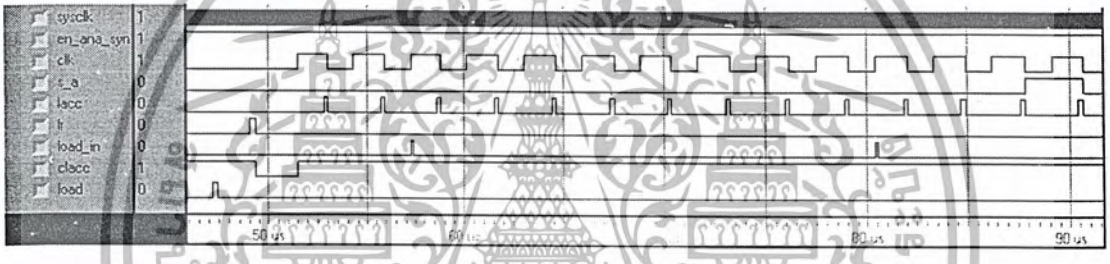
รูปที่ 4.81 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3

ลิขสิทธิ์ของวิทยาลัยนิวัฒนทหารลาดกระบัง



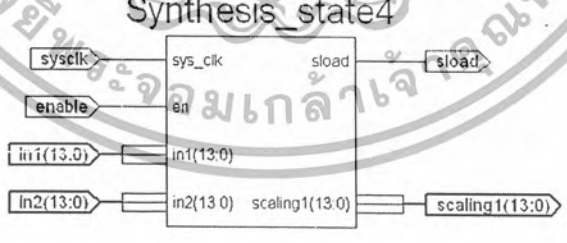
รูปที่ 4.82 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.83



รูปที่ 4.83 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 3

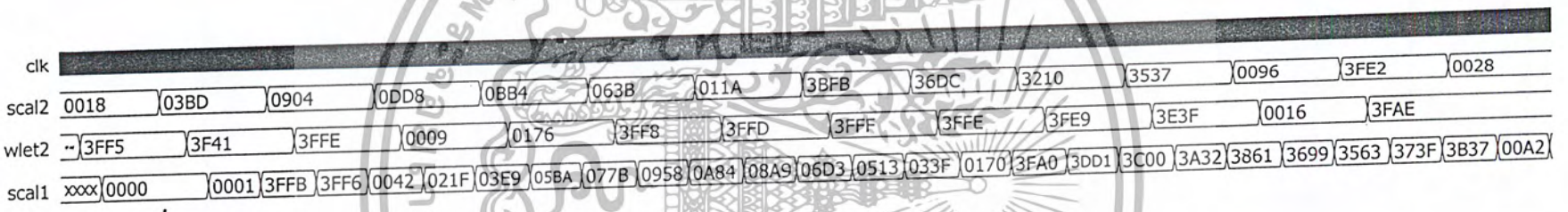
4.2.3.6 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4



รูปที่ 4.84 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4

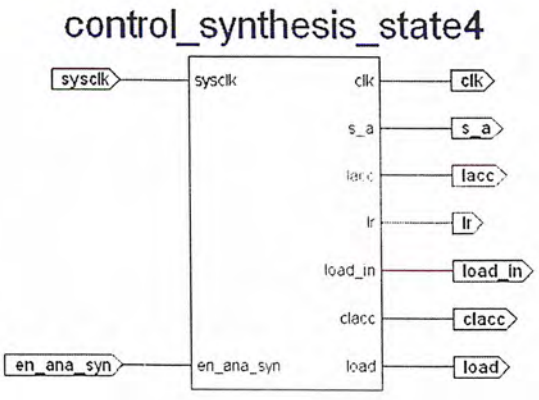
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.85

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



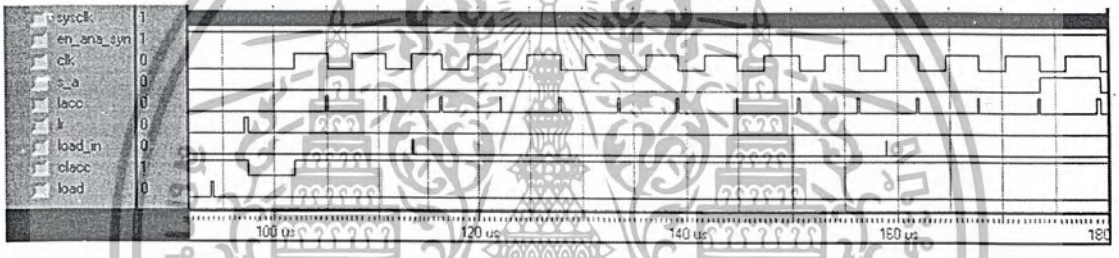
รูปที่ 4.85 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4

ลิขสิทธิ์ของวิทยาลัยการทัพบก ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น



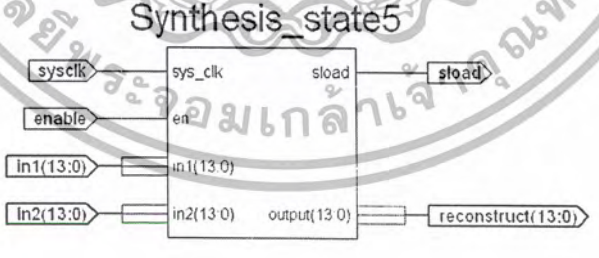
รูปที่ 4.86 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.87



รูปที่ 4.87 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 4

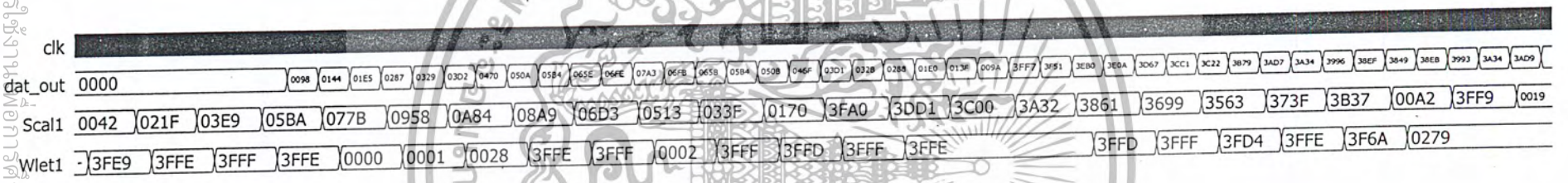
4.2.3.7 ส่วนของวงจรควบคุมและวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5



รูปที่ 4.88 แสดงสัญลักษณ์ของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5

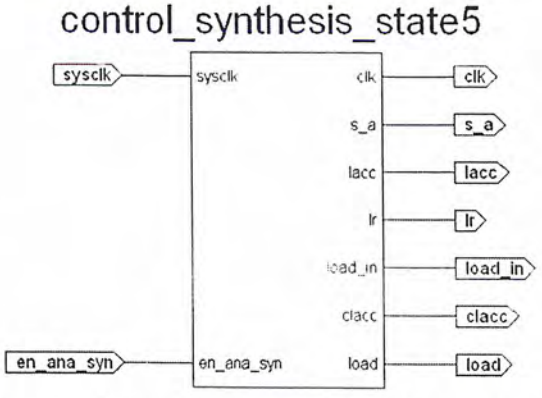
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



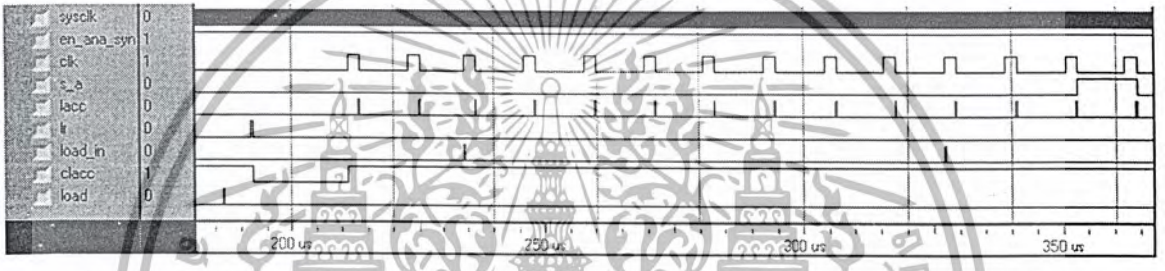
รูปที่ 4.89 แสดงผลการจำลองการทำงานของวงจรการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5

ลิขสิทธิ์ของสถาบันวิทยาลัยการทัพบก ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น



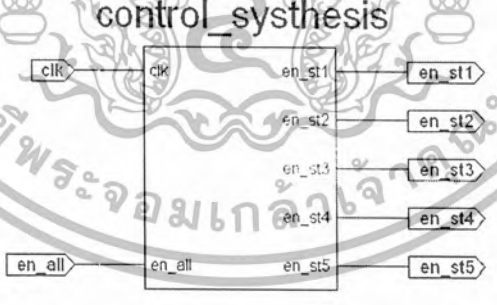
รูปที่ 4.90 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.91



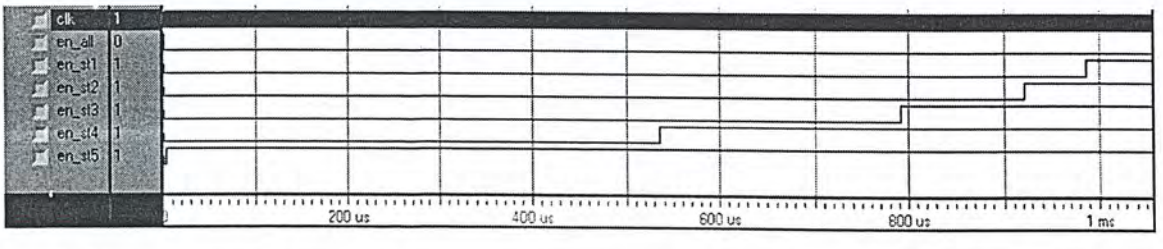
รูปที่ 4.91 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตในส่วนสังเคราะห์ขั้นที่ 5

4.2.3.8 ส่วนของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนสังเคราะห์



รูปที่ 4.92 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนสังเคราะห์

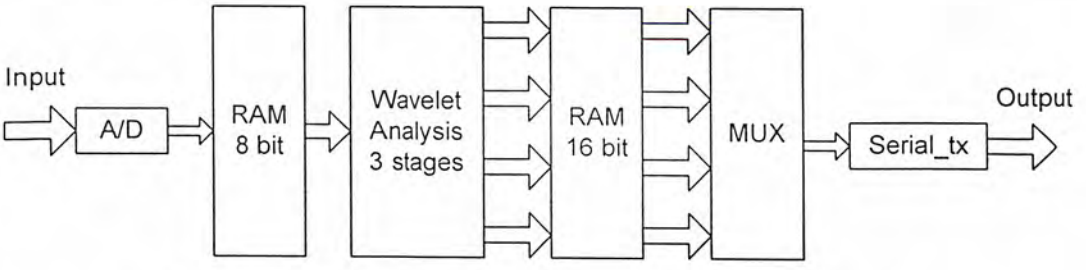
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.93



รูปที่ 4.93 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงเวฟเลตรวมในส่วนสังเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ส่วนของการ Implement ลงบนอุปกรณ์ FPGA

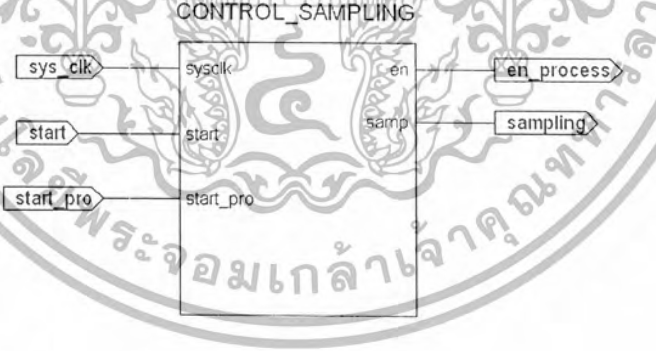


รูปที่ 4.94 รูปบล็อกไดอะแกรมในส่วนของการ Implement ลงบนอุปกรณ์ FPGA

ในส่วนนี้จะทำการเก็บผลการทดลอง โดยแบ่งเป็น 6 ส่วน คือ ส่วนของวงจรควบคุมการสุ่มค่าตัวอย่าง ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 8 บิท, ส่วนของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ 3 ขั้น, ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 16 บิท, ส่วนของวงจรมัลติเพล็กซ์ และส่วนของการส่งข้อมูลออกจากบอร์ด FPGA ผ่านทางพอร์ตอนุกรม ดังนี้

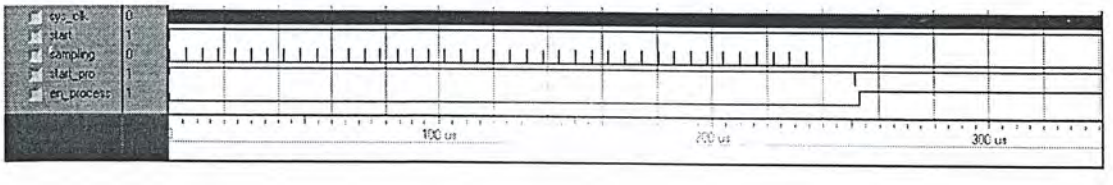
4.3.1 ส่วนของวงจรควบคุมการสุ่มค่าตัวอย่าง (Control Sampling)

ในการรับข้อมูลอินพุตจากตัวกำเนิดสัญญาณ สัญญาณที่ป้อนเข้าเป็นสัญญาณอนาล็อก จึงต้องทำการแปลงให้อยู่ในรูปของสัญญาณดิจิตอล โดยการผ่านวงจร A/D ซึ่งจะทำหน้าที่ในการสุ่มค่าตัวอย่างเพื่อให้ได้เป็นสัญญาณดิจิตอล ซึ่งการสุ่มค่าตัวอย่างนี้จะถูกควบคุมโดยวงจรควบคุมการสุ่มค่าตัวอย่าง ดังรูปที่ 4.95



รูปที่ 4.95 แสดงสัญลักษณ์ของวงจรควบคุมการสุ่มค่าตัวอย่าง

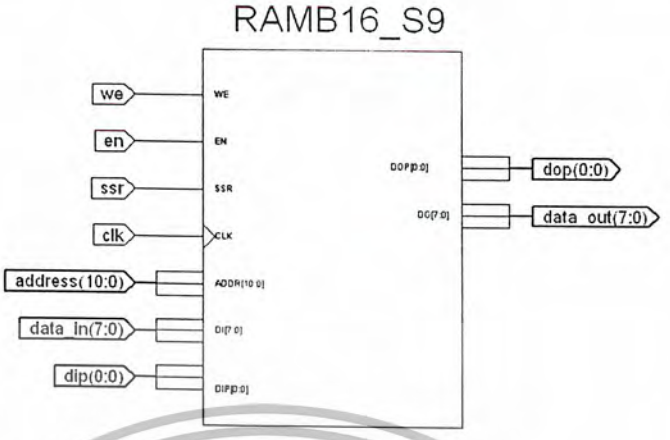
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.96



รูปที่ 4.96 แสดงผลการจำลองการทำงานของวงจรควบคุมการสุ่มค่าตัวอย่าง

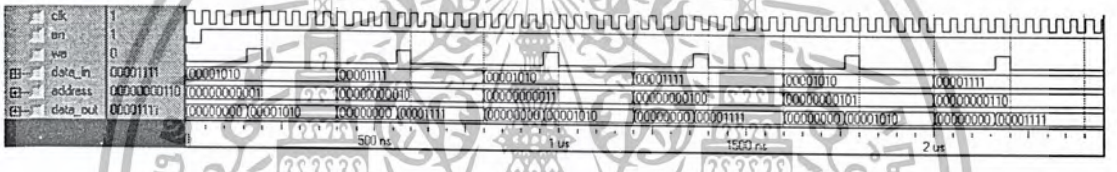
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 8 บิท (RAM 8 bit)



รูปที่ 4.97 แสดงสัญลักษณ์ของวงจรRAM 8 บิท

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.98

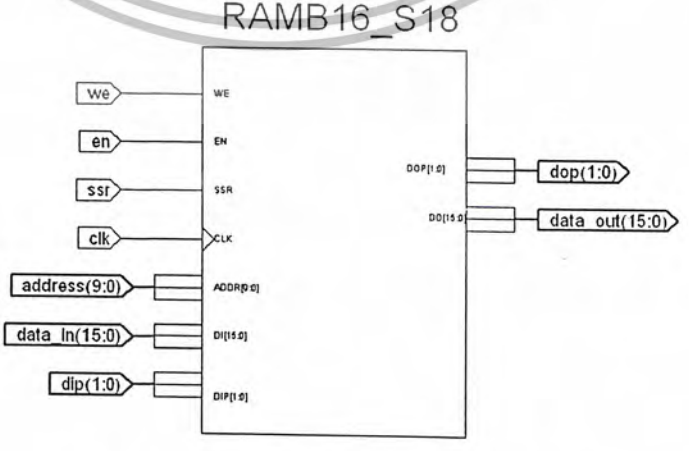


รูปที่ 4.98 แสดงผลการจำลองการทำงานของวงจรRAM 8 บิท

4.3.3 ส่วนของวงจรแปลงเวฟเลตในส่วนวิเคราะห์ (Analysis Wavelet Transform)

ในส่วนนี้จะประกอบไปด้วยการแปลงเวฟเลตทั้งหมด 3 ขั้นตอน ซึ่งผลการทดลองจะเหมือนกับในส่วนของการ Simulation โดยใช้โปรแกรม Xilinx แต่จะแตกต่างกันตรงที่ในส่วนนี้จะทำการแปลงเพียง 3 ขั้นตอน เท่านั้น

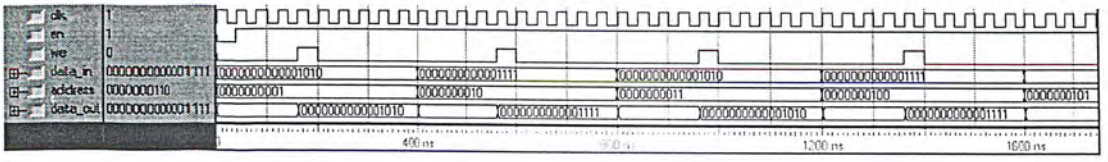
4.3.4 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 16 บิท (RAM 16 bit)



รูปที่ 4.99 แสดงสัญลักษณ์ของวงจรRAM 16 บิท

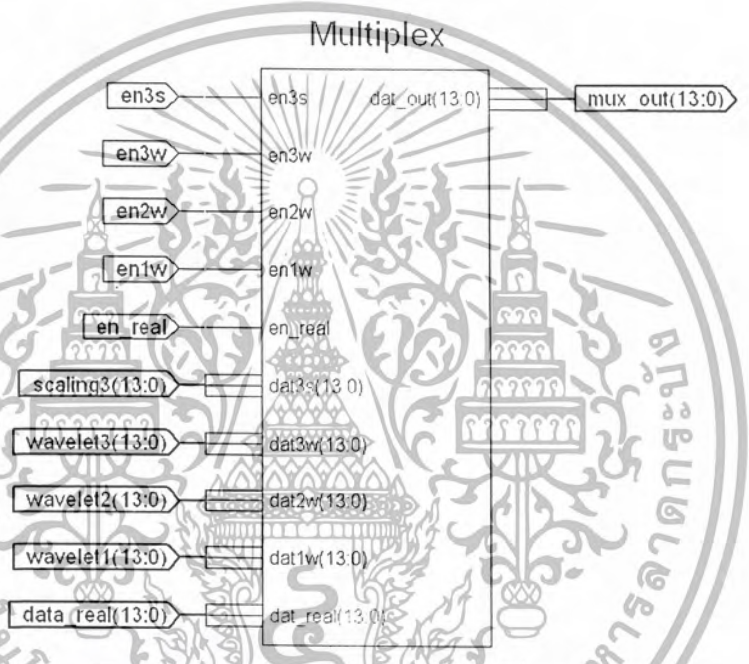
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.100



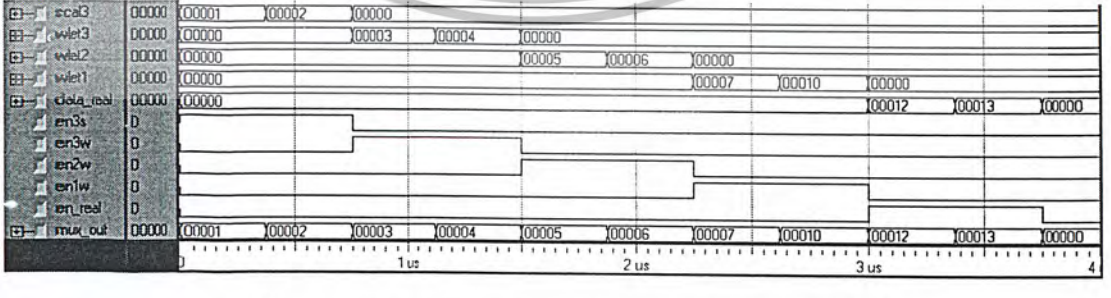
รูปที่ 4.100 แสดงผลการจำลองการทำงานของวงจรRAM 16 บิท

4.3.5 ส่วนของวงจรมัลติเพล็กซ์ (Multiplex)



รูปที่ 4.101 แสดงสัญลักษณ์ของวงจรมัลติเพล็กซ์

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.102



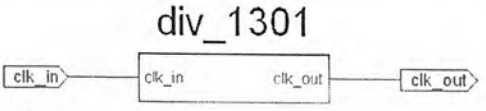
รูปที่ 4.102 แสดงผลการจำลองการทำงานของวงจรมัลติเพล็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.6 ส่วนของการส่งข้อมูลออกจากบอร์ด FPGA ผ่านทางพอร์ตอนุกรม แบ่งเป็น 3 ส่วน คือ

4.3.6.1 ส่วนของวงจรหารความถี่

ส่วนของวงจรหารความถี่ ทำหน้าที่หารความถี่ที่ของระบบให้ได้เอาท์พุตเป็นความถี่บอดเรด(Baud Rate) ที่ใช้ในการรับ-ส่งบิทข้อมูลทางพอร์ตอนุกรม โครงงานนี้ใช้แหล่งกำเนิดความถี่ 25 MHz สำหรับใช้เป็นความถี่ของระบบ แต่เราต้องหารความถี่เพื่อให้ได้เท่ากับความถี่บอดเรด 9600 บิทต่อวินาทีสามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรหารความถี่ได้สัญลักษณ์ ดังรูปที่ 4.103



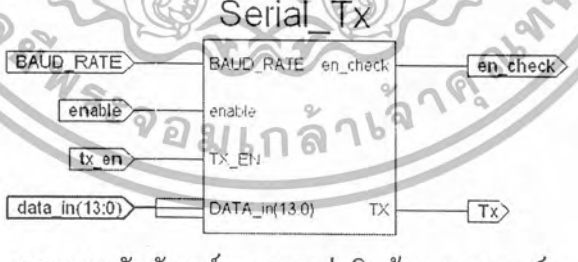
รูปที่ 4.103 แสดงสัญลักษณ์ของวงจรหารความถี่

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.104

รูปที่ 4.104 แสดงผลการจำลองการทำงานของวงจรหารความถี่

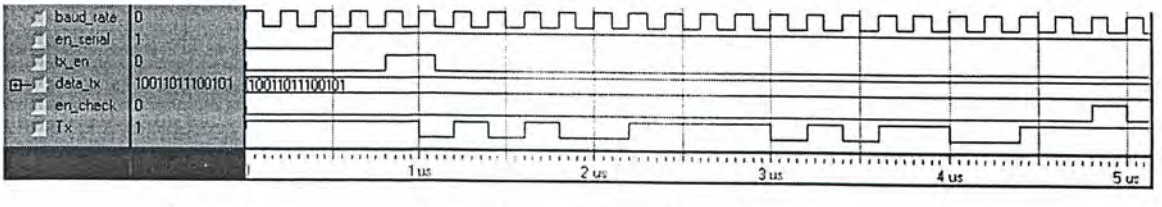
4.3.6.2 ส่วนของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม (SERIAL_TX)

ส่วนของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม ทำหน้าที่ส่งเฟรมบิตของข้อมูลออกพอร์ตอนุกรมโดยการแปลงข้อมูลบิทขนานที่รับมาจากวงจรมัลติเพล็กซ์ เป็นบิทอนุกรมแล้วส่งข้อมูลออกพอร์ตอนุกรมตามความถี่บอดเรด สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรมได้สัญลักษณ์ ดังรูปที่ 4.105



รูปที่ 4.105 แสดงสัญลักษณ์ของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.106

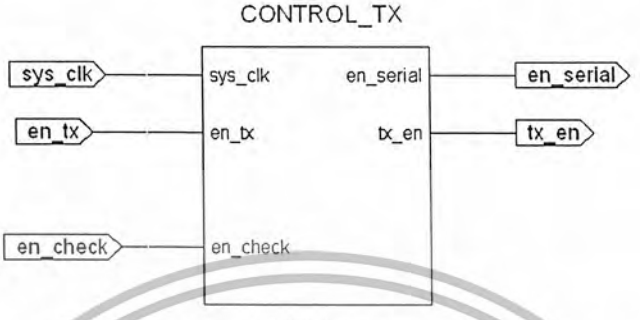


รูปที่ 4.106 แสดงผลการจำลองการทำงานของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

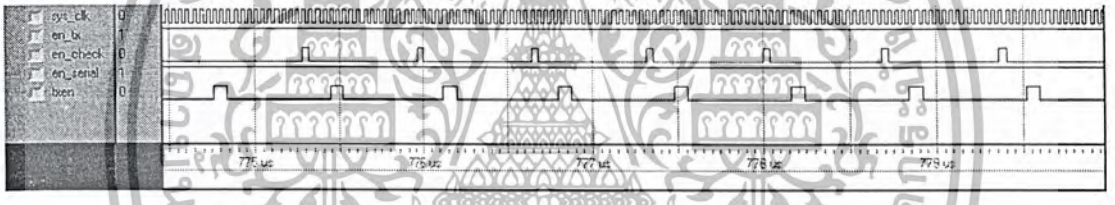
4.3.6.3 ส่วนของวงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม (CON_TX)

วงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม ทำหน้าที่สร้างสัญญาณควบคุมเพื่อให้การทำงานของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรมมีความถูกต้องเป็นไปตามลำดับขั้น สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม ดังรูปที่ 4.107



รูปที่ 4.107 แสดงสัญลักษณ์ของวงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน(Simulation) ได้ดังรูปที่ 4.108



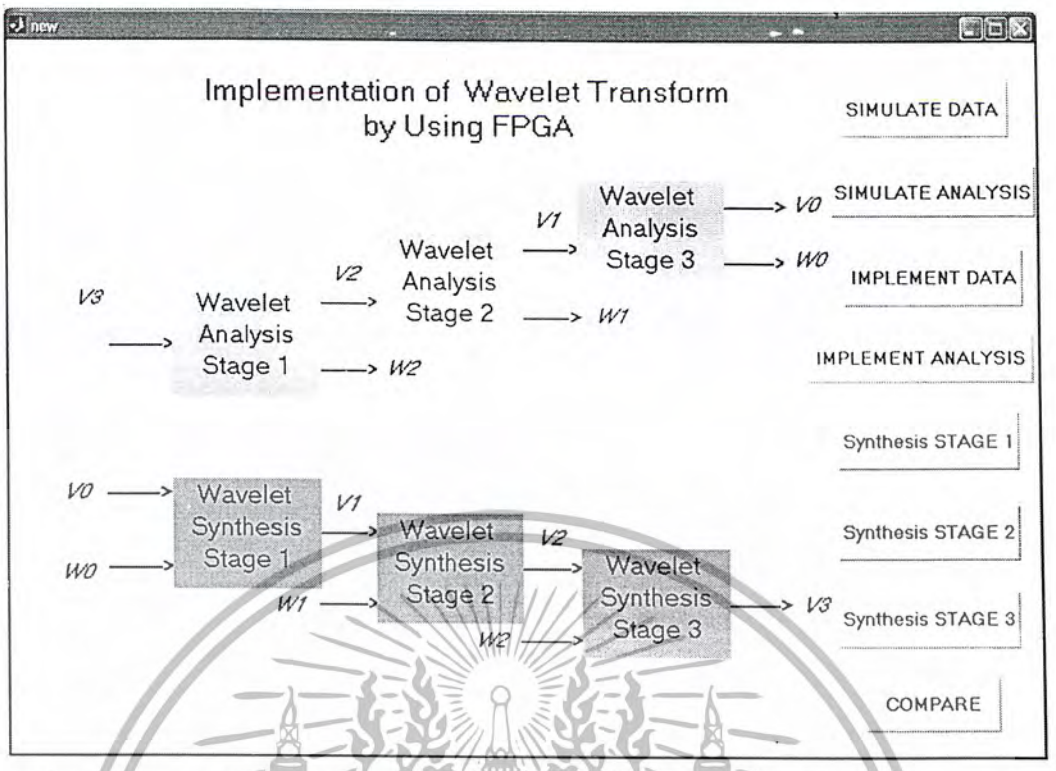
รูปที่ 4.108 แสดงผลการจำลองการทำงานของวงจรควบคุมการส่งบิทข้อมูลออกพอร์ตอนุกรม

4.4 ส่วนของการแสดงผลผ่านจอคอมพิวเตอร์

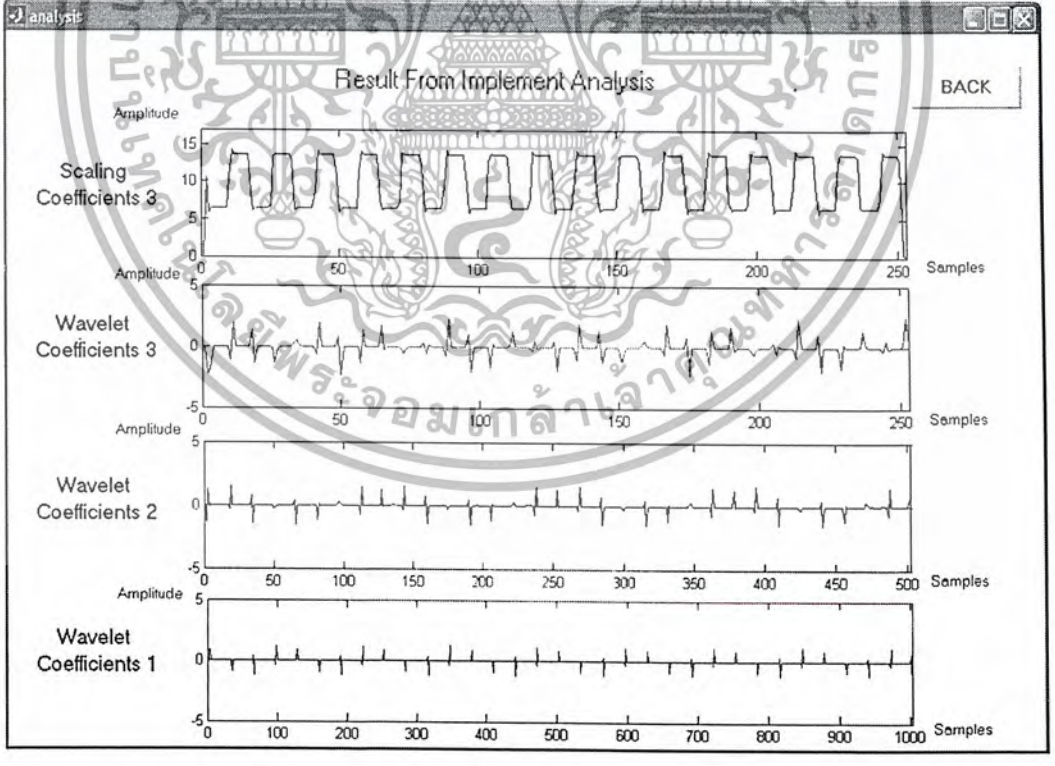
จากการนำสัญญาณอินพุตจากเครื่องกำเนิดสัญญาณส่งไปยังโปรแกรมที่ใช้เขียนขึ้นลงบน FPGA แล้วทำการเก็บผลการทดลอง โดยการส่งข้อมูลจาก FPGA ไปยังพอร์ตอนุกรมของคอมพิวเตอร์ แล้วแสดงผลทั้งหมดผ่านทาง GUIs (Graphical User Interfaces) โดยโปรแกรม MATLAB

ในการเก็บผลการทดลอง จะทำการป้อนสัญญาณอินพุตเป็นสัญญาณรูปสี่เหลี่ยม (Square wave) แล้วทำการเก็บค่าสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตในแต่ละระดับและทำการนำอินพุตที่ได้มาทำการแปลงเวฟเลตส่วนวิเคราะห์โดยใช้โปรแกรมMATLAB 3 ขั้นและหาความผิดพลาดในแต่ละสัมประสิทธิ์ที่หามาได้ พร้อมทั้งเปรียบเทียบรูปสัญญาณอินพุต กับสัญญาณเอาท์พุตที่ทำการกู้กลับมาได้ โดยจะมีหน้าต่างแสดงผล ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

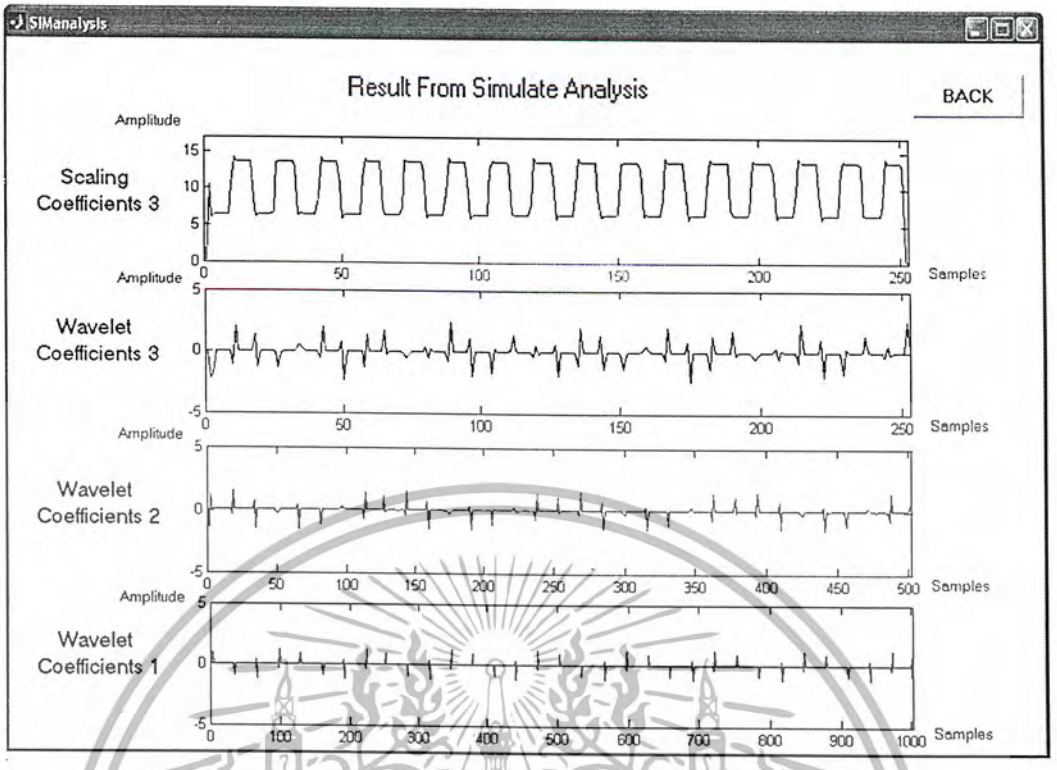


รูปที่ 4.109 แสดงบล็อกไดอะแกรมของการแปลงเวฟเลตทั้งทางด้านวิเคราะห์และด้านสังเคราะห์

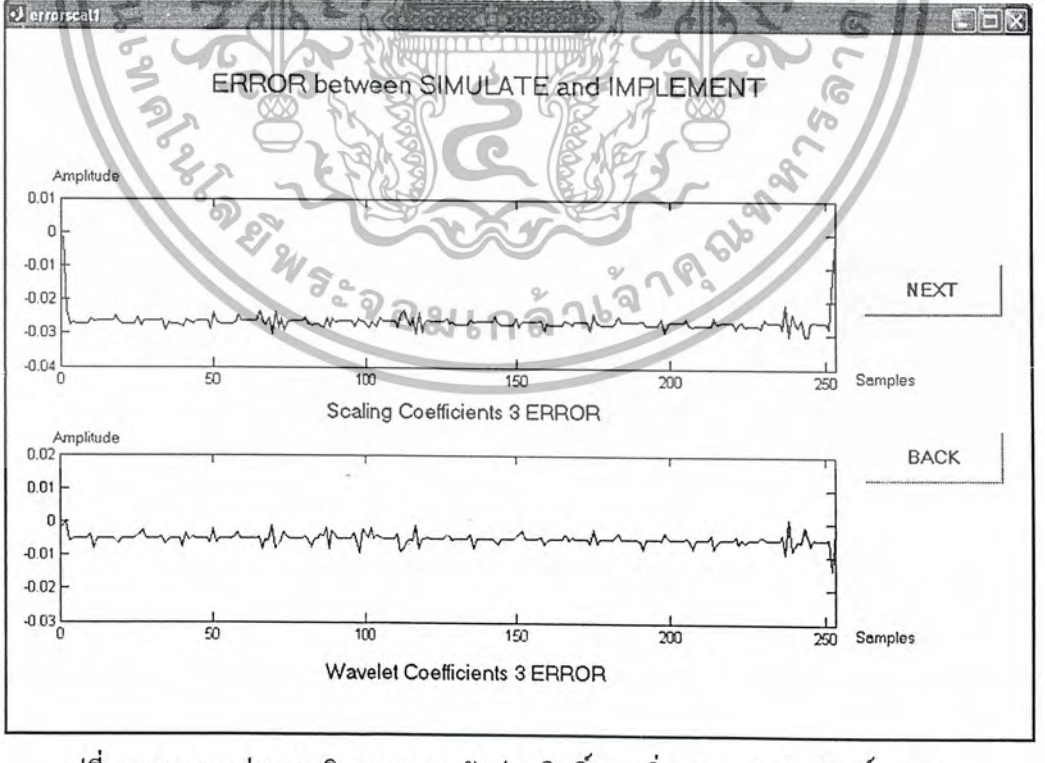


รูปที่ 4.110 แสดงสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตในแต่ละระดับ
ที่ได้จากการแปลงเวฟเลตบนอุปกรณ์ FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

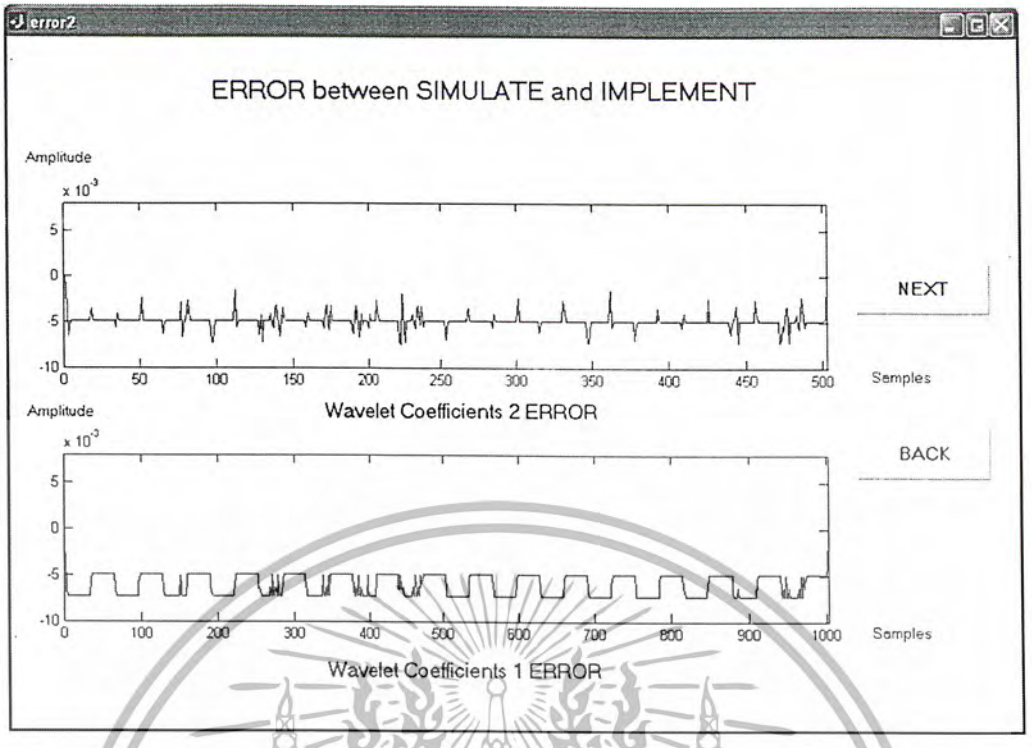


รูปที่ 4.111 แสดงสัมประสิทธิ์สเกลลิ่ง และสัมประสิทธิ์เวฟเลตในแต่ละระดับ
ที่ได้จากการแปลงเวฟเลตโดยใช้โปรแกรมMATLAB

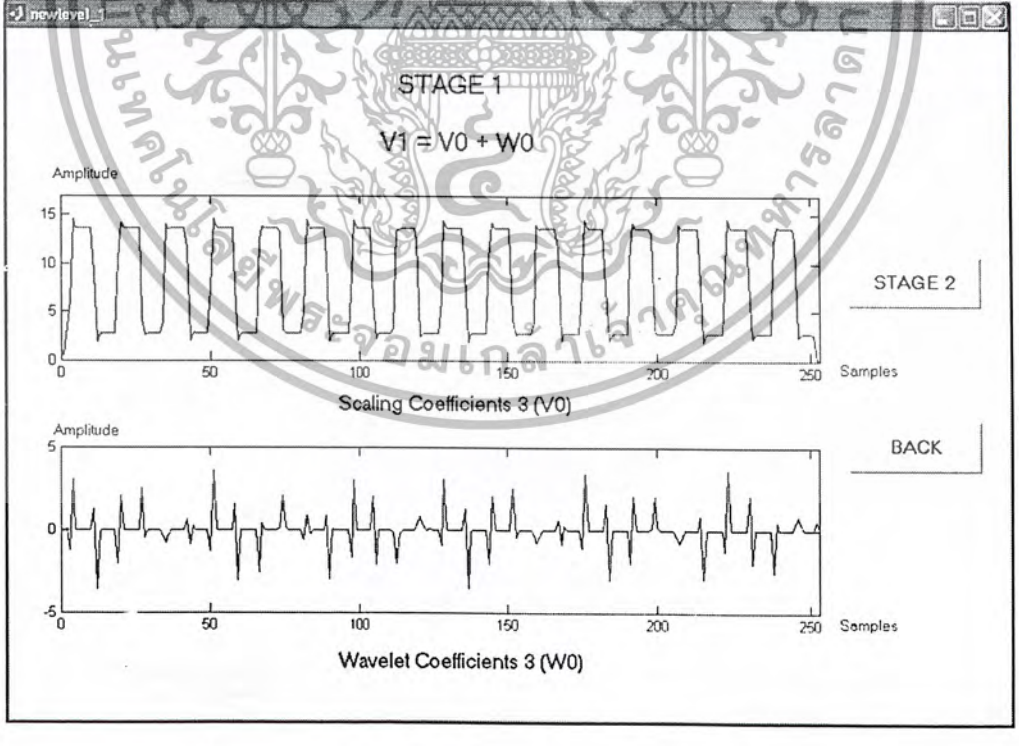


รูปที่ 4.112 แสดงค่าความผิดพลาดของสัมประสิทธิ์สเกลลิ่ง3 และสัมประสิทธิ์เวฟเลต3
ที่ได้จากการแปลงเวฟเลตบนอุปกรณ์FPGAและใช้โปรแกรมMATLAB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

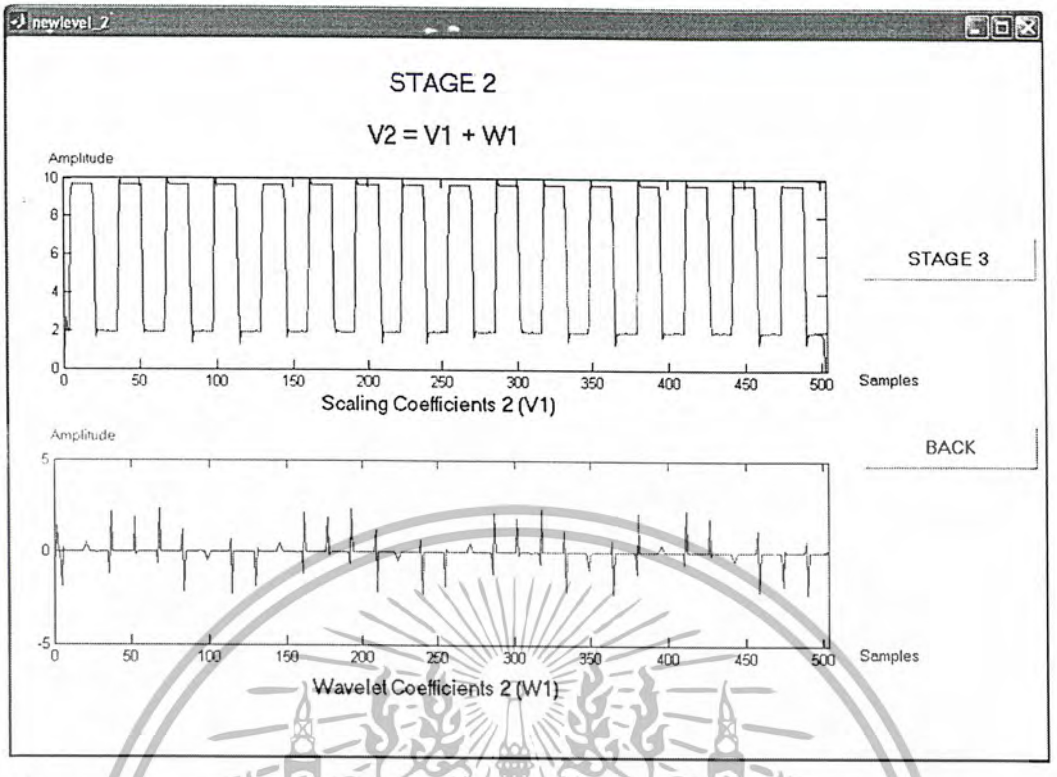


รูปที่ 4.113 แสดงค่าความผิดพลาดของสัมประสิทธิ์เวฟเลต2 และสัมประสิทธิ์เวฟเลต1
ที่ได้จากการแปลงเวฟเลตบนอุปกรณ์FPGAและใช้โปรแกรมMATLAB

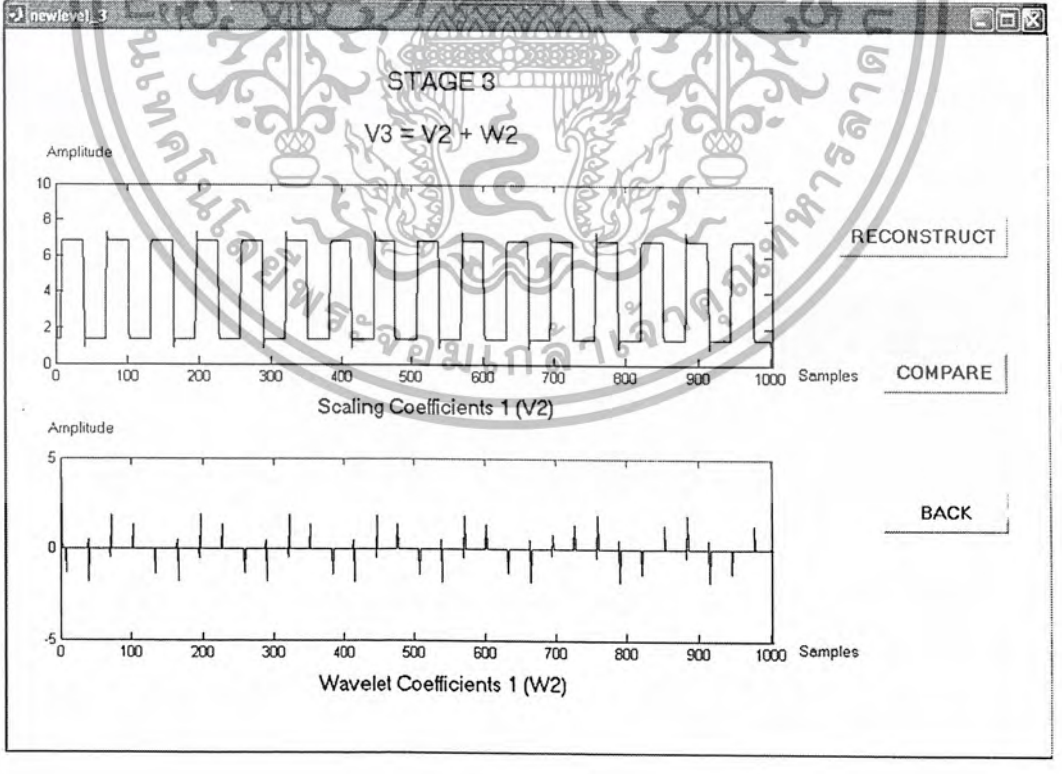


รูปที่ 4.114 แสดงสัมประสิทธิ์สเกลลิ่ง 3 และสัมประสิทธิ์เวฟเลต3
ที่ได้จากการแปลงเวฟเลตด้านวิเคราะห์บนอุปกรณ์FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

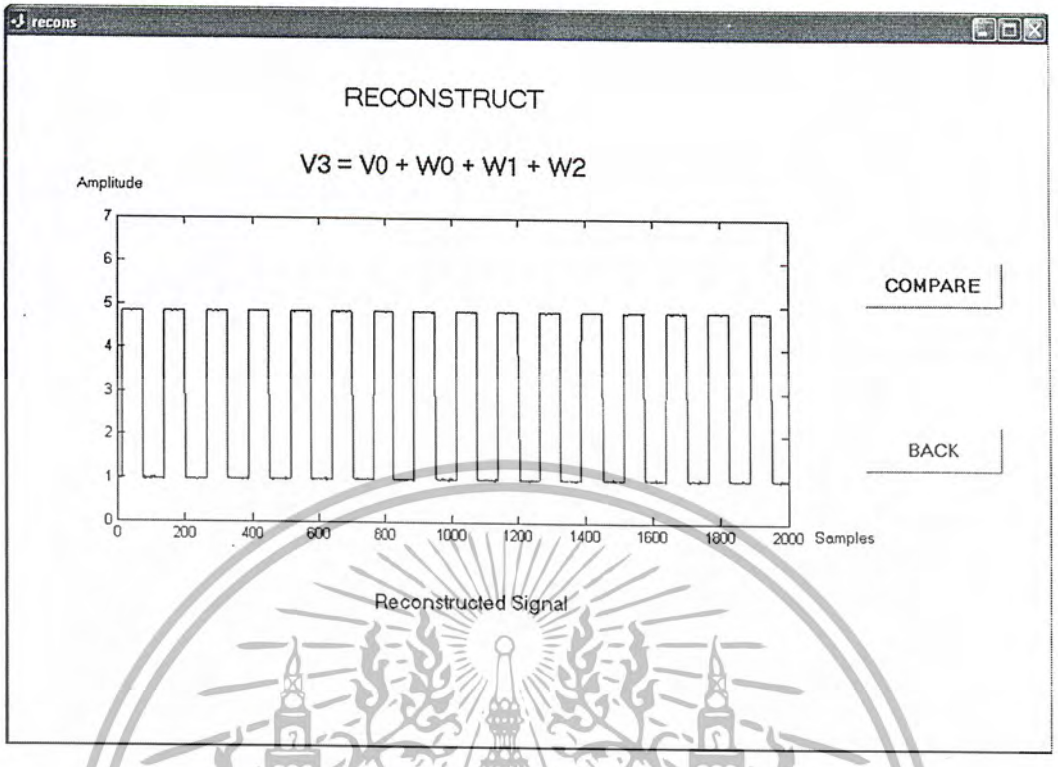


รูปที่ 4.115 แสดงสัมประสิทธิ์สเกลลิ่ง 2 ที่ได้จากการแปลงเวฟเลตด้านสังเคราะห์ในขั้นที่1 และสัมประสิทธิ์เวฟเลต2 ที่ได้จากการแปลงเวฟเลตด้านวิเคราะห์บนอุปกรณ์FPGA

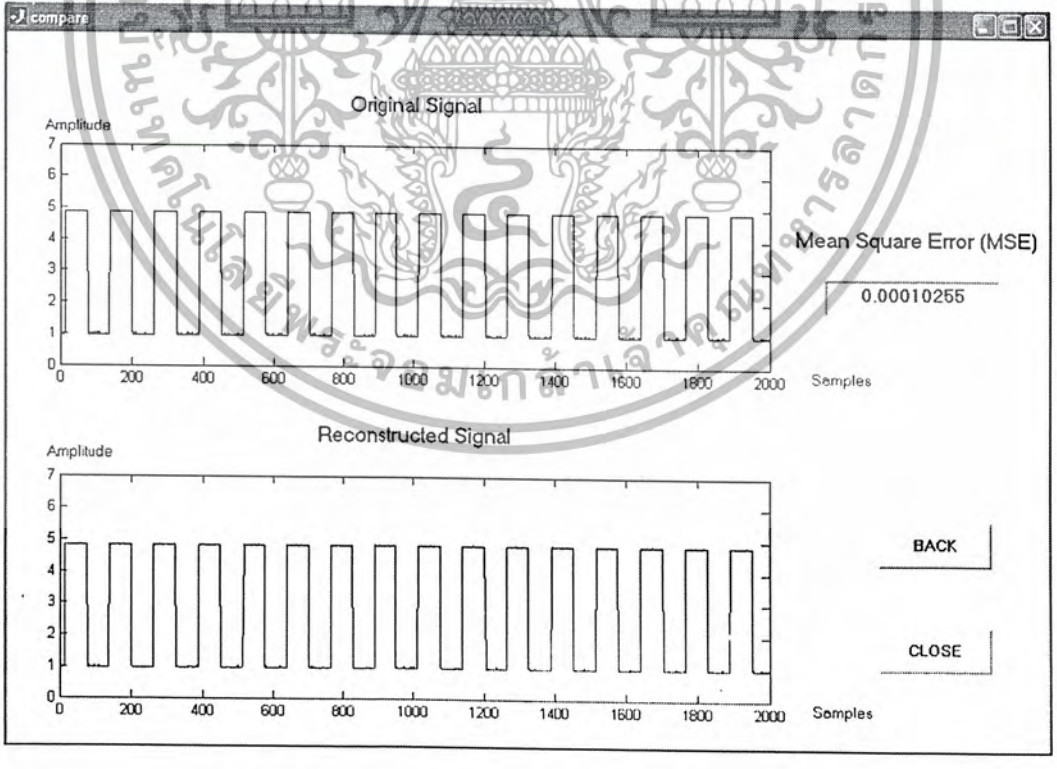


รูปที่ 4.116 แสดงสัมประสิทธิ์สเกลลิ่ง 1 ที่ได้จากการแปลงเวฟเลตด้านสังเคราะห์ในขั้นที่2 และสัมประสิทธิ์เวฟเลต1 ที่ได้จากการแปลงเวฟเลตด้านวิเคราะห์บนอุปกรณ์FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



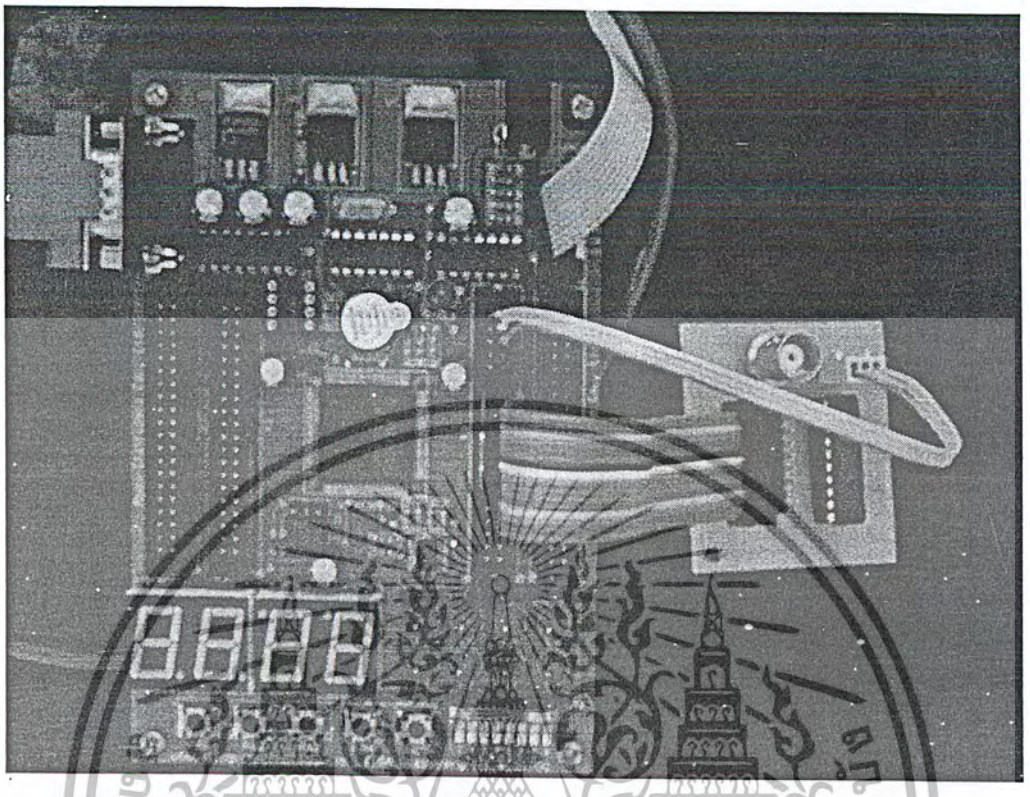
รูปที่ 4.117 แสดงข้อมูลสุดท้ายที่ทำการกู้กลับมาได้



รูปที่ 4.118 แสดงการเปรียบเทียบข้อมูลต้นแบบ กับข้อมูลที่ทำการกู้กลับมาได้ พร้อมทั้งแสดงค่า MSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ภาพอุปกรณ์ที่ใช้ในการทดสอบการแปลงเวฟเลต



รูปที่ 4.119 แสดงบอร์ดเอฟพีจีเอที่ทำการเชื่อมต่อกับพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

จากการศึกษา การแปลงเวฟเลตทางด้านวิเคราะห์ และการแปลงเวฟเลตทางด้านสังเคราะห์นั้น จะไม่มีการสูญเสีย (Lossless) ในทางทฤษฎี แต่จากการจำลองการทำงานการแปลงเวฟเลต โดยใช้โปรแกรมแมทแลบพบว่า ได้เกิดค่าความผิดพลาด เนื่องจากโปรแกรมแมทแลบมีขอบเขตของทศนิยมที่จำกัด ทำให้ค่าของสัมประสิทธิ์ดาร์บีชีที่เลือกใช้นั้นถูกปัดค่า ทำให้มีค่าคลาดเคลื่อนไป จากการทดลองโดยใช้ข้อมูลต้นแบบเป็นสัญญาณดีสครีตซายน์ขนาด 100 จุด และเลือกใช้สัมประสิทธิ์ดาร์บีชี 4 ค่า นำมาทำการแปลงเวฟเลตทั้งทางด้านวิเคราะห์ ซึ่งจะแปลงให้อยู่ในรูปของสัมประสิทธิ์สเกลลิ่งและสัมประสิทธิ์เวฟเลต และทำการแปลงเวฟเลตด้านสังเคราะห์ ซึ่งจะได้สัญญาณกู้กลับ เมื่อนำมาเปรียบเทียบกับสัญญาณต้นแบบ สามารถหาค่าความผิดพลาดได้ ซึ่งมีค่าน้อยมากประมาณศูนย์ ซึ่งตรงตามทฤษฎี

ส่วนของการออกแบบ ด้วยโปรแกรม Xilinx-Project Navigator โดยใช้ภาษา VHDL จะใช้ระบบการแทนข้อมูลตัวเลข Fixed Point 14 บิท เขียนโปรแกรมให้ทำการแปลงเวฟเลตทั้งทางด้านวิเคราะห์ และด้านสังเคราะห์ทั้งหมด 5 ระดับ เมื่อนำสัมประสิทธิ์ที่ได้จากส่วนของการออกแบบโดยใช้ภาษา VHDL เปรียบเทียบกับผลที่ได้จากโปรแกรมแมทแลบ สามารถพิสูจน์ได้ว่า โปรแกรมที่เขียนขึ้นนั้น สามารถทำงานได้จริง แต่มีค่าความผิดพลาดเกิดขึ้น เนื่องจากระบบ Fixed Point 14 บิท ไม่สามารถแทนค่าทศนิยมของสัมประสิทธิ์ได้ครบ จึงเกิดการปัดค่าขึ้น

การ Implement ลงบนอุปกรณ์ FPGA ทำการแปลงเวฟเลตในส่วนของการวิเคราะห์ 3 ระดับ แล้วทำการแสดงผลของค่าสัมประสิทธิ์ที่ประมวลผลได้ พร้อมทั้งนำสัมประสิทธิ์นั้นไปทำการแปลงเวฟเลตด้านสังเคราะห์ 3 ระดับและแสดงผลค่าสัมประสิทธิ์ที่ได้จากการแปลงกลับด้วย นอกจากนั้นจะแสดงผลค่าความผิดพลาดที่เกิดขึ้นระหว่างสัญญาณต้นแบบและสัญญาณกู้กลับเป็นค่า Mean Square Error (MSE) จะเห็นว่าโปรแกรมที่ Implement ลงบนอุปกรณ์ FPGA สามารถทำงานได้จริง

ปัญหาที่เกิดขึ้น เนื่องจากอุปกรณ์ FPGA มีจำนวนทรัพยากรเป็นจำนวนจำกัด แต่ในการออกแบบการแปลงเวฟเลต 5 ระดับ มีการใช้ทรัพยากรบนอุปกรณ์ FPGA เป็นจำนวนมากเกินกว่าที่จะสามารถ Implement ลงบนอุปกรณ์ FPGA ที่มีอยู่ได้ จึงทำให้ในส่วนของการ Implement มีเฉพาะการแปลงเวฟเลตทางด้านวิเคราะห์เพียง 3 ระดับเท่านั้น วิธีการแก้ปัญหาคือ ออกแบบโปรแกรมให้มีการใช้ทรัพยากรน้อยลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานฉบับนี้สำเร็จลุล่วงได้ด้วยดี โดยได้รับความช่วยเหลือ และคำชี้แนะจากหลายท่าน ผู้จัดทำขอขอบพระคุณอาจารย์ที่ปรึกษา ผศ.อัครพล ตรีรัตน์ ที่ให้คำปรึกษาและความช่วยเหลือด้านข้อมูลและอุปกรณ์ในการทำปริญญานิพนธ์ ขอขอบคุณอาจารย์ศรวัฒน์ ชิวปรีชา ที่ให้คำปรึกษาและให้ความช่วยเหลือในด้านการใช้งานเอฟพีจีเอ ผู้เขียนพึงระลึกอยู่เสมอว่ารายงานฉบับนี้จะไม่สามารถสำเร็จลงได้เลย หากขาดความช่วยเหลือจากทุกท่านจึงขอขอบพระคุณมาอย่างสูง

นายคงฤทธิ์ เตชะวชิรศิริ
นางสาวจันทร์จรัส รุ่งสิริสุวรรณ
นางสาวจิตตินี รุ่งจตุรงค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

[1] Khalid Sayood, "Introduction to Data Compression", Morgan Kaufmann Publishers, San Francisco, California, 1996.

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

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

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

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

[6] ชำนาญ ปัญญาโส, วัชรากร หนูทอง, "ภาษา VHDL สำหรับการออกแบบวงจรดิจิตอล", ซีเอ็ด ยูเคชั่น, กรุงเทพฯ, 2547.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้