สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

อุปกรณ์แปลงข้อมูลแบบดิสครีตโคซายน์ทรานสฟอร์ม 2 มิติ โดยอุปกรณ์ FPGA

Implementation of 2D Discrete Cosine Transform Using FPGA

โดย

นายกฤช เรืองฤทธิ์

นายกฤติกานต์ ยมภักดี

นายจักร แซ่เอว

ได้ผ่านการตรวจแก้ไขแล้ว

อ.ดร. [illegible]

เลขหมู่.....

เลขทะเบียน 61917

วัน,เดือน,ปี 24 ก.ค. 2549

ป.ธ.....

ส.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

ภาควิชา
วิศวกรรมโทรคมนาคม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์แปลงข้อมูลแบบดิสครีตโคซายน์ทรานสฟอร์ม 2 มิติ โดยอุปกรณ์ FPGA

Implementation of 2D Discrete Cosine Transform Using FPGA

โดย

นายกฤช	เรืองฤทธิ์	44010008
นายกฤติกานต์	ยมภักดี	44010010
นายจักร	แซ่เอา	44010058

อาจารย์ที่ปรึกษา

ผศ. อัครพล ตรีรัตน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2547

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **อุปกรณ์แปลงข้อมูลแบบดิสครีตโคซายน์ทรานสฟอร์ม 2 มิติ โดยอุปกรณ์ FPGA**

Implementation of 2D Discrete Cosine Transform Using FPGA

ผู้จัดทำ

1. นายกฤช	**เรืองฤทธิ์**	**44010008**
2. นายกฤติกานต์	ยมภักดี	44010010
3. นายจักร	แซ่เอา	44010058

อาจารย์ที่ปรึกษา

(ผศ. อัครพล ตรีรัตน์)

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์แปลงข้อมูลแบบดิสครีตโคซายน์ทรานสฟอร์ม 2 มิติ โดยอุปกรณ์ FPGA

Implementation of 2D Discrete Cosine Transform Using FPGA

โดย นายกฤช เรืองฤทธิ์ 44010008

นายกฤติกานต์ ยมภักดี 44010010

นายจักร แซ่เอา 44010058

อาจารย์ที่ปรึกษา ผศ. อัครพล ตรีรัตน์

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการศึกษาและการออกแบบตัวแปลงข้อมูลแบบดิสครีตโคซายน์ทรานสฟอร์ม 2 มิติ (2D-Discrete Cosine Transform) โดยแยกเป็นการแปลงดิสครีตโคซายน์ทรานสฟอร์ม 1 มิติ 2 ครั้ง ซึ่งใช้อัลกอริทึมของคอร์ดิก (CORDIC Algorithm) และโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) มาช่วยในการลดจำนวนตัวคูณ โดยทำการจำลองกระบวนการทำงานด้วยโปรแกรมแมทเลบ (MATLAB) และในขั้นตอนการออกแบบจะใช้วิธีอธิบายการทำงานของวงจรด้วยภาษา VHDL (Very high speed integrated circuit Hardware Description Language) ลงบนอุปกรณ์ FPGA (Field Programmable Gate Array) เพื่อเป็นต้นแบบในการใช้งานต่อไป

Abstract

This thesis presents design and implementation of 2D Discrete Cosine Transform Processor. We utilize very useful CORDIC Algorithm and Distributed Arithmetic (DA) in order to reduce computational complexity in proposed algorithm. In this thesis, all the relevant results are first simulated with MATLAB and then we use VHDL as a developing-tool to design and implement in FPGA environment.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของหัวข้อปริญญานิพนธ์	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	2
1.3 ขอบเขตของปริญญานิพนธ์	2
1.4 เนื้อหาของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการ	4
2.1 การเข้ารหัสโดยการทรานสฟอร์ม	4
2.1.1 หลักการพื้นฐานของการทรานสฟอร์ม	4
2.1.2 สมการของการทรานสฟอร์ม	5
2.1.3 การแปลงแบบดิสครีตโคซายน์	7
2.2 ส่วนของการแปลงดิสครีตโคซายน์ 2 มิติ	10
2.3 ส่วนของการทรานสโพส	12
2.4 CORDIC Algorithm	14
2.5 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย	20
2.5.1 ระบบตัวเลข	20
2.5.2 ทฤษฎีเลขคณิตกระจาย	23
2.6 การประยุกต์ใช้งาน DA (Distributed Arithmetic) กับ CORDIC Algorithm	29
2.7 การสื่อสารข้อมูล	32
2.7.1 การสื่อสารข้อมูลแบบขนาน (Parallel Communication)	32
2.7.2 การสื่อสารข้อมูลแบบอนุกรม (Serial Communication)	32
2.7.2.1 การอินเตอร์เฟสตามมาตรฐาน RS-232	33
2.7.2.2 นัน โมเด็ม (Null Modems)	34
2.7.2.3 การเชื่อมต่อระหว่าง DB-9 กับ FPGA	35
2.8 ภาษาวีเอชดีแอล	36
2.8.1 ประวัติความเป็นมาของภาษาวีเอชดีแอล	36
2.8.2 การออกแบบระบบดิจิตอล	36
2.8.3 การออกแบบจากบนลงล่าง (Top-Down Design)	38
2.9 เอฟพีจีเอ	40
2.9.1 การออกแบบวงจรเชิงเลขด้วยชิพอุปกรณ์เอฟพีจีเอ	40
2.9.2 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การคำนวณและการสร้าง 42
3.1 ส่วนของการสร้างสัญญาณนาฬิกาของระบบและสัญญาณนาฬิกาของความถี่บอดเรต 42
3.2 ส่วนของวงจรหารความถี่ระบบ 44
3.3 การแปลงบิทข้อมูลส่งออกพอร์ตอนุกรมไปยังบอร์ด FPGA ด้วยโปรแกรม MATLAB 44
3.4 การรับข้อมูลจากพอร์ตอนุกรม 44
3.5 การออกแบบส่วนของวงจรคงค่าสัญญาณ ส่วนที่ 1 45
3.6 การออกแบบส่วนการแปลงดิสครีตโคซายน์แบบ1มิติ (1D-DCT) ส่วนที่ 1 45
3.6.1 การคำนวณหาสมการและสัมประสิทธิ์ที่ใช้ในการสร้าง 46
3.6.2 โครงสร้างของการแปลงดิสครีตโคซายน์ (Discrete Cosine Transform) 48
3.6.2.1 ส่วนของวงจรบวกและลบ 48
3.6.2.2 ส่วนของวงจรคูณกับค่าสัมประสิทธิ์ในการแปลงดิสครีตโคซายน์ 48
3.6.2.3 โครงสร้างโดยรวมของการแปลงดิสครีตโคซายน์ 49
3.6.3 การออกแบบที่นำหลักการของ CORDIC Algorithm มาประยุกต์ใช้ 49
3.6.3.1 การคำนวณหาสมการและสัมประสิทธิ์ที่ใช้ในการสร้าง 49
3.6.4 โครงสร้างของ CORDIC Algorithm ที่นำไปประยุกต์ใช้ในการแปลง ดิสครีตโคซายน์ 51
3.6.4.1 ส่วนของวงจรที่ใช้หลักการของ CORDIC Algorithm 51
3.6.4.2 โครงสร้างของการแปลงดิสครีตโคซายน์ ที่ใช้หลักการของ CORDIC Algorithm 54
3.7 ส่วนการคำนวณและการสร้างของวงจรทรานสโพส (Transpose) 55
3.7.1 ส่วนของการคำนวณและการสร้างของวงจรมัลติเพล็กซ์ ในวงจรทรานสโพส 55
3.7.2 ส่วนการคำนวณและการสร้างของวงจรทรานสโพสโดยใช้แรม 56
3.7.3 ส่วนของวงจรแรมที่ใช้สร้างจริงบนบอร์ด FPGA 57
3.8 การออกแบบส่วนของวงจรคงค่าสัญญาณ ส่วนที่ 2 (Latch) 58
3.9 การออกแบบส่วนการแปลงดิสครีตโคซายน์แบบ1มิติ (1D-DCT) ส่วนที่ 2 58
3.10 ส่วนของการคำนวณและการสร้างของวงจรมัลติเพล็กซ์ หลังการแปลงดิสครีตโคซายน์ แบบ 1 มิติ ส่วนที่ 2 59
3.11 ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ 59
3.12 ส่วนของการรับบิทข้อมูลจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ผ่านทางพอร์ตอนุกรม 60
3.13 ส่วนของการแปลงดิสครีตโคซายน์กลับแบบ 2 มิติ ด้วยโปรแกรม MATLAB 60
3.14 ส่วนการแสดงโฟว์ลชาร์ทของการสร้างอุปกรณ์การแปลงดิสครีตโคซายน์ 2 มิติ 61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 การทดลองและผลการทดลอง 62
4.1 ส่วนของการรับข้อมูลจากคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม 63
4.1.1 ส่วนของวงจร Digital Clock Managers (DCMs) 63
4.1.2 ส่วนของวงจรหารความถี่ 64
4.1.2.1 ส่วนของวงจรหารความถี่บอดเรต 64
4.1.2.2 ส่วนของวงจรหารความถี่ของระบบ 65
4.1.3 ส่วนของการรับบิทข้อมูลจากพอร์ตอนุกรม (SERIAL_RX) 65
4.1.4 ส่วนของวงจรวันพัลส์ (Onepulse) 66
4.1.5 ส่วนของวงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม (CON_RX) 67
4.2 ส่วนของวงจรคงค่าสัญญาณชั่วคราวและการแปลงดิสครีตโคซายน์แบบ 1 มิติส่วนที่ 1 67
4.2.1 ส่วนของวงจรคงค่าสัญญาณชั่วคราว 8 บิท (Latch) 67
4.2.2 ส่วนของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ 68
4.2.2.1 ส่วนของวงจรบวกและลบขนาด 8 บิท 71
4.2.2.2 ส่วนของวงจรบวกและลบขนาด 9 บิท 71
4.2.2.3 ส่วนของวงจรบวกขนาด 9 บิท 72
4.2.2.4 ส่วนของวงจรลบขนาด 9 บิท 72
4.2.2.5 ส่วนของวงจรแอนด์(AND) สัญญาณ 4 อินพุท 73
4.2.2.6 ส่วนของวงจรแอนด์(AND) สัญญาณ 2 อินพุท 73
4.2.2.7 ส่วนของวงจร DA_CORDIC 1 74
4.2.2.8 ส่วนของวงจร DA_CORDIC 2 74
4.2.2.9 ส่วนของวงจร DA_CORDIC 3 75
4.2.2.10 ส่วนของวงจร DA_CORDIC 4 76
4.2.2.11 ส่วนของวงจร DA_CORDIC 5 76
4.2.2.12 ส่วนของวงจร DA_CORDIC 6 77
4.2.2.13 ส่วนของวงจรเลื่อนบิท 9 บิท 78
4.2.2.14 ส่วนของวงจรเลื่อนบิท 10 บิท 79
4.2.2.15 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา 79
4.2.2.16 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา 80
4.2.2.17 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 45 องศา 81
4.2.2.18 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา 81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.19 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา 82
4.2.2.20 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา 83
4.2.2.21 ส่วนของวงจรบวกสะสม 9 บิท 83
4.2.2.22 ส่วนของวงจรบวกสะสม10 บิท 84
4.2.2.23 ส่วนของวงจรควบคุมภายในวงจรDA_CORDIC 1 - DA_CORDIC 4 85
4.2.2.24 ส่วนของวงจรควบคุมภายในวงจรDA_CORDIC 5 - DA_CORDIC 6 85
4.2.2.25 ส่วนของวงจรรีจีสเตอร์ขนาด 10 บิท (Register) 86
4.2.2.26 ส่วนของวงจรควบคุมการแปลงดิสครีตโคซายน์ ส่วนที่ 1 87
4.3 ส่วนของวงจรทรานสโพส (Transpose) 87
4.3.1 ส่วนของวงจรมัลติเพล็กซ์ 89
4.3.2 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล (RAM) 90
4.3.3 ส่วนของการลดจำนวนบิทข้อมูล 91
4.4 ส่วนของวงจรคงค่าสัญญาณชั่วคราวและการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 หลังทำการทรานสโพส 91
4.4.1 ส่วนของวงจรคงค่าสัญญาณชั่วคราว 10 บิท (Latch) 91
4.4.2 ส่วนของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 92
4.4.2.1 ส่วนของวงจรบวกและลบขนาด 10 บิท 95
4.4.2.2 ส่วนของวงจรบวกและลบขนาด 11 บิท 96
4.4.2.3 ส่วนของวงจรบวกขนาด 11 บิท 96
4.4.2.4 ส่วนของวงจรลบขนาด 11 บิท 97
4.4.2.5 ส่วนของวงจร DA_CORDIC 1_1 97
4.4.2.6 ส่วนของวงจร DA_CORDIC 2_1 98
4.4.2.7 ส่วนของวงจร DA_CORDIC 3_1 98
4.4.2.8 ส่วนของวงจร DA_CORDIC 4_1 99
4.4.2.9 ส่วนของวงจร DA_CORDIC 5_1 99
4.4.2.10 ส่วนของวงจร DA_CORDIC 6_1 100
4.4.2.11 ส่วนของวงจรเลื่อนบิท 11 บิท 100
4.4.2.12 ส่วนของวงจรเลื่อนบิท 12 บิท 101
4.4.2.13 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา 102
4.4.2.14 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา 102

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2.15 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 45 องศา 103
4.4.2.16 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา 104
4.4.2.17 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา 104
4.4.2.18 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา 105
4.4.2.19 ส่วนของวงจรบวกสะสม 11 บิท 106
4.4.2.20 ส่วนของวงจรบวกสะสม 12 บิท 106
4.4.2.21 ส่วนของวงจรควบคุมภายในวงจร DA_CORDIC 1_1 - DA_CORDIC 4_1 107
4.4.2.22 ส่วนของวงจรควบคุมภายในวงจร DA_CORDIC 5_1 - DA_CORDIC 6_1 108
4.4.2.23 ส่วนของวงจรรีจีสเตอร์ขนาด 12 บิท (Register) 108
4.4.2.24 ส่วนของวงจรควบคุมการแปลงดิสครีตโคซายน์ ส่วนที่ 2 หลังการทรานสโพส 109
4.5 ส่วนของการส่งข้อมูลออกจากบอร์ด FPGA ผ่านทางพอร์ตอนุกรม 110
4.5.1 ส่วนของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม (SERIAL_TX) 110
4.6 ส่วนของวงจรควบคุมการทำงานของระบบ 111
4.7 ส่วนประกอบภายในและการเชื่อมต่อของวงจรทั้งหมด 112
4.8 ส่วนของการประมวลผลการทำงานจริง 114
4.8.1 ขั้นตอนการเก็บผลการทดสอบ Hard ware ด้วยชุดข้อมูลขนาด 8×8 114
4.8.2 ขั้นตอนการเก็บผลการทดสอบ Hard ware ด้วยชุดข้อมูลภาพ 120
4.9 ภาพอุปกรณ์ที่ใช้ในการทดสอบการแปลงดิสครีตโคซายน์ 2 มิติ 126
บทที่ 5 บทวิจารณ์และบทสรุป
กิตติกรรมประกาศ
หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 แสดงลักษณะของการแปลงดิสครีตโคซายน์ของบล็อกข้อมูลขนาด 8 X 8 พิกเซล จากข้อมูลทั้งหมด 10

รูปที่ 2.2 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์ 2 มิติ 11

รูปที่ 2.3 แสดงบล็อกไดอะแกรมการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวนอน 11

รูปที่ 2.4 แสดงบล็อกไดอะแกรมการทรานสโพสบล็อกข้อมูล 11

รูปที่ 2.5 แสดงบล็อกไดอะแกรมการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวตั้ง 12

รูปที่ 2.6 แสดงการทรานสโพสโดยอาศัยคุณสมบัติของรีจิสเตอร์ 13

รูปที่ 2.7 การหมุนของเวคเตอร์ในพิกัดฉาก 14

รูปที่ 2.8 ผลการกระทำของค่า σ_i ที่เกิดกับค่า θ 17

รูปที่ 2.9 แสดงการจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิทเครื่องหมาย บิทจำนวนเต็มและบิทเศษส่วน 20

รูปที่ 2.10 แสดงการจัดรูปแบบจำนวนโดยตรงที่มีบิทเครื่องหมายและบิทเศษส่วน 21

รูปที่ 2.11 แสดงการจัดรูปแบบจำนวนอิงดรรชนี 22

รูปที่ 2.12 แสดงการคูณแบบเลขส่วนเติมเต็มสองโดยใช้เลขคณิตกระจาย 26

รูปที่ 2.13 แสดงโครงสร้าง IDCT แบบหนึ่งมิติที่นำ DA ไปใช้ 29

รูปที่ 2.14 แสดงโครงสร้างที่นำ DA ไปใช้กับ CORDIC Algorithm 31

รูปที่ 2.15 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารข้อมูลแบบขนาน 32

รูปที่ 2.16 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารแบบอนุกรม 32

รูปที่ 2.17 แสดงลักษณะของคอนเน็คเตอร์แบบ DB-9 33

รูปที่ 2.18 แสดงบล็อกไดอะแกรมการเชื่อมต่อแบบนันโมเด็ม 34

รูปที่ 2.19 แสดงการเชื่อมต่อระหว่างคอนเน็คเตอร์แบบ DB-9 และ บอร์ด FPGA โดยผ่าน MAX3232 35

รูปที่ 2.20 แสดงขั้นตอนการออกแบบระบบดิจิตอล 37

รูปที่ 2.21 การออกแบบระบบเส้นทางข้อมูล 37

รูปที่ 2.22 แสดงขั้นตอนการออกแบบจากบนลงล่าง 38

รูปที่ 2.23 แสดงผังการแบ่งกลุ่มของวงจรรวมเอซิก 40

รูปที่ 3.1 แสดงการแปลงดิสครีตโคซายน์แบบ 2 มิติ โดยแยกเป็นการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง 42

รูปที่ 3.2 แสดงบล็อกไดอะแกรมของฟังก์ชันต่างๆ ของ DCMs 43

รูปที่ 3.3 แสดงบล็อกไดอะแกรมของฟังก์ชัน DFS ของ DCMs ที่สร้างสัญญาณเอาท์พุท ความถี่ 24 MHz จากสัญญาณอินพุทความถี่ 25 MHz 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.4 แสดงบล็อกไดอะแกรมของวงจรหารความถี่บอดเรต 43
รูปที่ 3.5 แสดงบล็อกไดอะแกรมของวงจรหารความถี่ระบบ 44
รูปที่ 3.6 แสดงบล็อกไดอะแกรมการแปลงบิทข้อมูลส่งออกพอร์ตอนุกรมไปยังบอร์ด FPGA 44
รูปที่ 3.7แสดงบล็อกไดอะแกรมการเปลี่ยนข้อมูลอินพุทจากข้อมูลแบบอนุกรมเป็นข้อมูลแบบขนาน 45
รูปที่ 3.8 แสดงบล็อกไดอะแกรมของวงจรคงค่าสัญญาณ ส่วนที่ 1 45
รูปที่ 3.9 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1 45
รูปที่ 3.10 แสดงบล็อกไดอะแกรมที่ใช้ในการบวกและลบ 48
รูปที่ 3.11 แสดงบล็อกไดอะแกรมของการคูณค่าสัมประสิทธิ์ในการแปลงดิสครีตโคซายน์ 48
รูปที่ 3.12 โครงสร้างโดยรวมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ 49
รูปที่ 3.13 (a) บล็อกไดอะแกรมที่ยังไม่ใช้หลักการของ CORDIC Algorithm 51
(b) บล็อกไดอะแกรมที่ใช้หลักการของ CORDIC Algorithm 51
รูปที่ 3.14 โครงสร้างของ CORDIC Algorithm โดยใช้ทฤษฎีเลขคณิตกระจาย 52
รูปที่ 3.15 แสดงโครงสร้างการแปลงดิสครีตโคซายน์แบบ 1 มิติ ที่ใช้หลักการของ
CORDIC Algorithm 54
รูปที่ 3.16 แสดงบล็อกไดอะแกรมของวงจรรีจิสเตอร์ 54
รูปที่ 3.17 แสดงบล็อกไดอะแกรมการทรานสโพส 55
รูปที่ 3.18 แสดงบล็อกไดอะแกรมของการมัลติเพล็กซ์ข้อมูลจากส่วนของ
การแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1 55
รูปที่ 3.19 แสดงบล็อกไดอะแกรมการทรานสโพสโดยใช้แรม 56
รูปที่ 3.20 แสดงการไหลของข้อมูลในระหว่างกระบวนการเขียนข้อมูล 57
รูปที่ 3.21 แสดงลักษณะการทำงานของ RAM ในโหมด Write first 57
รูปที่ 3.22 แสดงบล็อกไดอะแกรมของวงจรคงค่าสัญญาณ ส่วนที่ 2 58
รูปที่ 3.23 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 58
รูปที่ 3.24 แสดงบล็อกไดอะแกรมของวงจรเก็บค่าข้อมูลเอาท์พุท 59
รูปที่ 3.25 แสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรม
จากบอร์ด FPGAไปยังคอมพิวเตอร์ 59
รูปที่ 3.26 แสดงบล็อกไดอะแกรมการรับบิทข้อมูลจากบอร์ด FPGA ผ่านพอร์ตอนุกรม 60
รูปที่ 3.27 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์กลับแบบ 2 มิติ 60
รูปที่ 3.28 แสดงโฟว์ลชาร์ทของการสร้างอุปกรณ์การแปลงดิสครีตโคซายน์ 2 มิติ 61
รูปที่ 4.1 แสดงการทำงานของการแปลงดิสครีตโคซายน์ 2 มิติ 62
รูปที่ 4.2 แสดงส่วนประกอบของวงจรรับข้อมูลจากคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม 63
รูปที่ 4.3 แสดงสัญลักษณ์ของวงจร DCMs 63
รูปที่ 4.4 แสดงผลการจำลองการทำงานของวงจร DCMs 64
รูปที่ 4.5 แสดงสัญลักษณ์ของวงจรหารความถี่บอดเรต 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.6 แสดงผลการจำลองการทำงานของวงจรหารความถี่บอดเรต 64
รูปที่ 4.7 แสดงสัญลักษณ์ของวงจรหารความถี่ของระบบ 65
รูปที่ 4.8 แสดงผลการจำลองการทำงานของวงจรหารความถี่ของระบบ 65
รูปที่ 4.9 แสดงสัญลักษณ์ของวงจรการรับบิทข้อมูลจากพอร์ตอนุกรม 65
รูปที่ 4.10 แสดงผลการจำลองการทำงานของวงจรการรับบิทข้อมูลจากพอร์ตอนุกรม 66
รูปที่ 4.11 แสดงสัญลักษณ์ของวงจรวันพัลส์(Onepulse) 66
รูปที่ 4.12 แสดงผลการจำลองการทำงานของวงจรวันพัลส์(Onepulse) 66
รูปที่ 4.13 แสดงสัญลักษณ์ของวงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม 67
รูปที่ 4.14 แสดงผลการจำลองการทำงานของวงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม 67
รูปที่ 4.15 แสดงสัญลักษณ์ของวงจรคงค่าสัญญาณชั่วคราว 8 บิท 67
รูปที่ 4.16 แสดงผลการจำลองการทำงานของวงจรคงค่าสัญญาณชั่วคราว 8 บิท 68
รูปที่ 4.17 แสดงสัญลักษณ์ของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ 68
รูปที่ 4.18 แสดงผลการจำลองการทำงานของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ 69
รูปที่ 4.19 แสดงส่วนประกอบภายในของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ 70
รูปที่ 4.20 แสดงสัญลักษณ์ของวงจรบวกและลบขนาด 8 บิท 71
รูปที่ 4.21 แสดงผลการจำลองการทำงานของวงจรบวกและลบขนาด 8 บิท 71
รูปที่ 4.22 แสดงสัญลักษณ์ของวงจรบวกและลบขนาด 9 บิท 71
รูปที่ 4.23 แสดงผลการจำลองการทำงานของวงจรบวกและลบขนาด 9 บิท 72
รูปที่ 4.24 แสดงสัญลักษณ์ของวงจรบวกขนาด 9 บิท 72
รูปที่ 4.25 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 9 บิท 72
รูปที่ 4.26 แสดงสัญลักษณ์ของวงจรลบขนาด 9 บิท 72
รูปที่ 4.27 แสดงผลการจำลองการทำงานของวงจรลบขนาด 9 บิท 73
รูปที่ 4.28 แสดงสัญลักษณ์ของวงจรแอนด์สัญญาณ 4 อินพุท 73
รูปที่ 4.29 แสดงผลการจำลองการทำงานของวงจรแอนด์สัญญาณ 4 อินพุท 73
รูปที่ 4.30 แสดงสัญลักษณ์ของวงจรแอนด์สัญญาณ 2 อินพุท 73
รูปที่ 4.31 แสดงผลการจำลองการทำงานของวงจรแอนด์สัญญาณ 2 อินพุท 74
รูปที่ 4.32 แสดงสัญลักษณ์ของวงจร DA_CORDIC 1 74
รูปที่ 4.33 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 1 74
รูปที่ 4.34 แสดงสัญลักษณ์ของวงจร DA_CORDIC 2 74
รูปที่ 4.35 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 2 75
รูปที่ 4.36 แสดงสัญลักษณ์ของวงจร DA_CORDIC 3 75
รูปที่ 4.37 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 3 75
รูปที่ 4.38 แสดงสัญลักษณ์ของวงจร DA_CORDIC 4 76
รูปที่ 4.39 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 4 76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.40 แสดงสัญลักษณ์ของวงจร DA_CORDIC 5 76
รูปที่ 4.41 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 5 77
รูปที่ 4.42 แสดงสัญลักษณ์ของวงจร DA_CORDIC 6 77
รูปที่ 4.43 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 6 77
รูปที่ 4.44 แสดงองค์ประกอบของวงจร DA_CORDIC 78
รูปที่ 4.45 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 9 บิท 78
รูปที่ 4.46 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 9 บิท 78
รูปที่ 4.47 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 10 บิท 79
รูปที่ 4.48 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 10 บิท 79
รูปที่ 4.49 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา 79
รูปที่ 4.50 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา 80
รูปที่ 4.51 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา 80
รูปที่ 4.52 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา 80
รูปที่ 4.53 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 45 องศา 81
รูปที่ 4.54 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 45 องศา 81
รูปที่ 4.55 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา 81
รูปที่ 4.56 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา 82
รูปที่ 4.57 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา 82
รูปที่ 4.58 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา 82
รูปที่ 4.59 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา 83
รูปที่ 4.60 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา 83
รูปที่ 4.61 แสดงสัญลักษณ์ของวงจรบวกสะสม 9 บิท 83

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.62 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 9 บิท 84
รูปที่ 4.63 แสดงสัญลักษณ์ของวงจรบวกสะสม 10 บิท 84
รูปที่ 4.64 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 10 บิท 84
รูปที่ 4.65 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร
DA_CORDIC 1 – DA_CORDIC 4 85
รูปที่ 4.66 แสดงการจำลองการทำงานของวงจรควบคุมภายในวงจร
DA_CORDIC 1 – DA_CORDIC 4 85
รูปที่ 4.67 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร
DA_CORDIC 5 – DA_CORDIC 6 85
รูปที่ 4.68 แสดงการจำลองการทำงานของวงจรควบคุมภายในวงจร
DA_CORDIC 5 – DA_CORDIC 6 86
รูปที่ 4.69 แสดงสัญลักษณ์ของวงจรรีจิสเตอร์ขนาด 10 บิท 86
รูปที่ 4.70 แสดงผลการจำลองการทำงานของวงจรรีจิสเตอร์ขนาด 10 บิท 86
รูปที่ 4.71 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงดิสครีตโคซายน์ 87
รูปที่ 4.72 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงดิสครีตโคซายน์ 87
รูปที่ 4.73 แสดงสัญลักษณ์ของวงจรทรานสโพส 87
รูปที่ 4.74 แสดงการจำลองการทำงานของวงจรทรานสโพสในการเขียนข้อมูลลงแรม 88
รูปที่ 4.75 แสดงการจำลองการทำงานของวงจรทรานสโพสในการอ่านข้อมูลออกจากแรม 88
รูปที่ 4.76 แสดงส่วนประกอบภายในของวงจรทรานสโพส 89
รูปที่ 4.77 แสดงสัญลักษณ์ของวงจรมัลติเพล็กซ์ 89
รูปที่ 4.78 แสดงผลการจำลองการทำงานของวงจรมัลติเพล็กซ์ 90
รูปที่ 4.79 แสดงสัญลักษณ์ของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 90
รูปที่ 4.80 แสดงผลการจำลองการทำงานของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล 90
รูปที่ 4.81 แสดงสัญลักษณ์ของวงจรลดจำนวนบิทข้อมูล 91
รูปที่ 4.82 แสดงผลการจำลองการทำงานของวงจรลดจำนวนบิทข้อมูล 91
รูปที่ 4.83 แสดงสัญลักษณ์ของวงจรคงค่าสัญญาณชั่วคราว 10 บิท 91
รูปที่ 4.84 แสดงผลการจำลองการทำงานของวงจรคงค่าสัญญาณชั่วคราว 92
รูปที่ 4.85 แสดงสัญลักษณ์ของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 92
รูปที่ 4.86 แสดงผลการจำลองการทำงานของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 93
รูปที่ 4.87 แสดงวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 94
รูปที่ 4.88 แสดงสัญลักษณ์ของวงจรการบวกและลบขนาด 10 บิท 95
รูปที่ 4.89 แสดงผลการจำลองการทำงานของวงจรการบวกและลบขนาด 10 บิท 95
รูปที่ 4.90 แสดงสัญลักษณ์ของวงจรบวกและลบขนาด 11 บิท 96
รูปที่ 4.91 แสดงผลการจำลองการทำงานของวงจรบวกและลบขนาด 11 บิท 96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.92 แสดงสัญลักษณ์ของวงจรบวกขนาด 11 บิท 96
รูปที่ 4.93 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 11 บิท 96
รูปที่ 4.94 แสดงสัญลักษณ์ของวงจรลบขนาด 11 บิท 97
รูปที่ 4.95 แสดงผลการจำลองการทำงานของวงจรลบขนาด 11 บิท 97
รูปที่ 4.96 แสดงสัญลักษณ์ของวงจร DA_CORDIC 1_1 97
รูปที่ 4.97 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 1_1 97
รูปที่ 4.98 แสดงสัญลักษณ์ของวงจร DA_CORDIC 2_1 98
รูปที่ 4.99 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 2_1 98
รูปที่ 4.100 แสดงสัญลักษณ์ของวงจร DA_CORDIC 3_1 98
รูปที่ 4.101 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 3_1 98
รูปที่ 4.102 แสดงสัญลักษณ์ของวงจร DA_CORDIC 4_1 99
รูปที่ 4.103 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 4_1 99
รูปที่ 4.104 แสดงสัญลักษณ์ของวงจร DA_CORDIC 5_1 99
รูปที่ 4.105 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 5_1 99
รูปที่ 4.106 แสดงสัญลักษณ์ของวงจร DA_CORDIC 6_1 100
รูปที่ 4.107 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 6_1 100
รูปที่ 4.108 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 11 บิท 100
รูปที่ 4.109 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 11 บิท 101
รูปที่ 4.110 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 12 บิท 101
รูปที่ 4.111 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 12 บิท 101
รูปที่ 4.112 แสดงสัญลักษณ์ของวางจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา 102
รูปที่ 4.113 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา 102
รูปที่ 4.114 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา 102
รูปที่ 4.115 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา 103
รูปที่ 4.116 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 45 องศา 103
รูปที่ 4.117 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 45 องศา 103
รูปที่ 4.118 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา 104

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.119 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา 104
รูปที่ 4.120 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา 104
รูปที่ 4.121 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา 105
รูปที่ 4.122 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา 105
รูปที่ 4.123 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา 105
รูปที่ 4.124 แสดงสัญลักษณ์ของวงจรบวกสะสม 11 บิท 106
รูปที่ 4.125 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 11 บิท 106
รูปที่ 4.126 แสดงสัญลักษณ์ของวงจรบวกสะสม 12 บิท 106
รูปที่ 4.127 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 12 บิท 107
รูปที่ 4.128 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร
DA_CORDIC 1_1 – DA_CORDIC 4_1 107
รูปที่ 4.129 แสดงการจำลองการทำงานของวงจรควบคุมภายในวงจร
DA_CORDIC 1_1 – DA_CORDIC 4_1 107
รูปที่ 4.130 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร
DA_CORDIC 5_1 – DA_CORDIC 6_1 108
รูปที่ 4.131 แสดงการจำลองการทำงานของวงจรควบคุมภายในวงจร
DA_CORDIC 5_1 – DA_CORDIC 6_1 108
รูปที่ 4.132 แสดงสัญลักษณ์ของวงจรรีจิสเตอร์ขนาด 12 บิท 108
รูปที่ 4.133 แสดงผลการจำลองการทำงานของวงจรรีจิสเตอร์ขนาด 12 บิท 109
รูปที่ 4.134 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงดิสครีตโคซายน์ 109
รูปที่ 4.135 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงดิสครีตโคซายน์ 109
รูปที่ 4.136 แสดงสัญลักษณ์ของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม 110
รูปที่ 4.137 แสดงผลการจำลองการทำงานของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม 110
รูปที่ 4.138 แสดงสัญลักษณ์ของวงจรควบคุม 111
รูปที่ 4.139 แสดงผลการจำลองการทำงานของวงจรควบคุม 111
รูปที่ 4.140 แสดงส่วนประกอบภายในและการเชื่อมต่อของวงจรทั้งหมด 112
รูปที่ 4.141 แสดงค่าอุปกรณ์ที่ใช้ในการประมวลผลการทำงานจริง 113
รูปที่ 4.142 แสดงค่าเวลาที่ใช้ในการประมวลผลการทำงานจริง 113
รูปที่ 4.143 สัญญาณภาพที่นำมาทดสอบเป็นบล็อกขนาด 8X8 114

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.144 แสดงค่า PSNR กับค่าความผิดพลาดที่เกิดขึ้นในการทำงานจริง
ของบล็อกข้อมูลขนาด 8X8 115
รูปที่ 4.145 แสดงการเปรียบเทียบสัญญาณต้นแบบกับสัญญาณที่สร้างกลับ
ของบล็อกข้อมูลขนาด 8X8 116
รูปที่ 4.146 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของบล็อกข้อมูล
ขนาด 8X8เป็นเลขฐาน 10 117
รูปที่ 4.147 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของบล็อกข้อมูล
ขนาด 8X8เป็นเลขฐาน 2 117
รูปที่ 4.148 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของบล็อกข้อมูล
ขนาด 8X8เป็นเลขฐาน 16 118
รูปที่ 4.149 แสดงค่าข้อมูลที่ได้จากโปรแกรม MATLAB เปรียบเทียบกับ
การทำงานจริงบนบอร์ด FPGA 118
รูปที่ 4.150 ฮิสโตแกรมเปรียบเทียบข้อมูลอินพุท ข้อมูลที่ได้จากการแปลง
และข้อมูลในการสร้างกลับของบล็อกข้อมูลขนาด 8X8 119
รูปที่ 4.151 สัญญาณภาพที่นำมาทดสอบ 120
รูปที่ 4.152 แสดงค่า PSNR กับค่าความผิดพลาดที่เกิดขึ้นในการทำงานจริงของสัญญาณภาพ 121
รูปที่ 4.153 แสดงการเปรียบเทียบสัญญาณต้นแบบกับสัญญาณที่สร้างกลับของสัญญาณภาพ 122
รูปที่ 4.154 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของสัญญาณภาพเป็นเลขฐาน 10 123
รูปที่ 4.155 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของสัญญาณภาพเป็นเลขฐาน 2 123
รูปที่ 4.156 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของสัญญาณภาพเป็นเลขฐาน 16 124
รูปที่ 4.157 แสดงค่าข้อมูลที่ได้จากโปรแกรม MATLAB เปรียบเทียบกับ
การทำงานจริงบนบอร์ด FPGA 124
รูปที่ 4.158 ฮิสโตแกรมเปรียบเทียบข้อมูลอินพุท ข้อมูลที่ได้จากการแปลง
และข้อมูลในการสร้างกลับ ของสัญญาณภาพ 125
รูปที่ 4.159 แสดงอุปกรณ์ที่ใช้ในการทดสอบการแปลงดิสครีตโคซายน์แบบ 2 มิติ
ที่ใช้หลักการการแยกการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง 126
รูปที่ 4.160 แสดงบอร์ด FPGA ที่ทำการเชื่อมต่อกับพอร์ตอนุกรม 126

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงค่าสัมประสิทธิ์ของการในรูปแบบของฟังก์ชันตรีโกณมิติ	8
ตารางที่ 2.2 แสดงการคำนวณหาค่าของ sine และ cosine	20
ตารางที่ 2.3 แสดงคุณสมบัติที่สำคัญของรูปแบบจำนวนโดยตรง	22
ตารางที่ 2.4 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง	26
ตารางที่ 2.5 แสดงค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุท	29
ตารางที่ 2.6 แสดงขาสัญญาณของพอร์ตสื่อสารอนุกรมแบบ DB-9	35
ตารางที่ 3.1 แสดงค่าสัมประสิทธิ์ในรูปแบบของฟังก์ชันตรีโกณมิติ	51
ตารางที่ 3.2 แสดงการชี้ค่าภายใน ROM ด้วยอินพุท 2 ค่า	54
ตารางที่ 3.3 แสดงค่าสัมประสิทธิ์ที่ใช้ในการแปลง ที่ค่ามุมต่าง ๆ	55

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของหัวข้อปริญญานิพนธ์

ปัจจุบันการสื่อสารต่าง ๆ นั้นมีความสำคัญต่อการดำเนินชีวิตของคนมากขึ้น ทำให้รูปแบบข้อมูลต่าง ๆ ที่จะทำการส่งผ่านจากต้นทางไปยังปลายทางมีมากขึ้น ตัวอย่างเช่น ข้อมูลเสียง ภาพ และวิดีโอ ถ้าหากว่าข้อมูลมีขนาดใหญ่มากจำนวนบิทที่ใช้ในการส่งก็จะมีมากเช่นกัน ซึ่งจะทำให้เกิดปัญหาในการส่งข้อมูล จนอาจจะมีผลทำให้การส่งข้อมูลนั้นล่าช้า จึงได้มีการนำเทคนิคการบีบอัดข้อมูล (Data Compression) มาใช้ในการส่งข้อมูลเพื่อแก้ปัญหาดังกล่าว

การบีบอัดข้อมูลนั้นเป็นเทคนิคอย่างหนึ่ง ที่นำมาใช้เพื่อเพิ่มประสิทธิภาพในกระบวนการส่งและเก็บข้อมูลด้วยการลดจำนวนบิทให้น้อยลง ซึ่งเทคนิคการบีบอัดข้อมูลนั้นมีหลายเทคนิคโดยในปริญญานิพนธ์ฉบับนี้ได้ศึกษาวิธีการทำงานของการแปลงแบบดิสครีตโคซายน์(Discrete Cosine Transform) 2 มิติ ซึ่งอาศัยหลักการของการแยกการแปลงข้อมูลออกเป็นการแปลงแบบดิสครีตโคซายน์ 1 มิติ จำนวน 2 ครั้ง และได้นำเอา CORDIC Algorithm (Coordinate Rotation Digital Computer) มาประยุกต์ใช้ร่วมกับการแปลงแบบดิสครีตโคซายน์ด้วย ในการออกแบบและสร้างเป็นฮาร์ดแวร์จะใช้การบรรยายพฤติกรรมการทำงานด้วยภาษา VHDL (Very high speed integrated circuit Hardware Description Language) และทำการประมวลผลผ่านทางฮาร์ดแวร์ นำผลการทดลองที่ได้มาเปรียบเทียบกับการประมวลผลจากโปรแกรมแมทแลบ (MATLAB)

แนวความคิดของ CORDIC Algorithm นั้นมีลักษณะพิเศษคือ สามารถคำนวณหาค่าในฟังก์ชันของตรีโกณมิติต่าง ๆ ได้ โดยใช้วิธีการเลื่อนและบวก (Shift and Add) เป็นหลัก และยังมีผลลัพธ์ของสมการอยู่ในรูปผลบวกของผลคูณ (Sum of Product) จึงเป็นทางออก สำหรับการสร้างฮาร์ดแวร์ที่ไม่ต้องการใช้การคูณ (Multiplication) โดยตรง เพื่อหวังว่าจะสามารถลดจำนวนอุปกรณ์ที่ต้องใช้ก็สามารถนำ CORDIC Algorithm ไปประยุกต์ใช้แทนได้

ในการประมวลผลสัญญาณหรือการสื่อสารในปัจจุบัน มีการพัฒนาในด้านต่าง ๆ ให้มีประสิทธิภาพและตรงตามความต้องการของผู้ใช้ เช่น ความถูกต้อง ความเร็วในการประมวลผลและความเร็วในการส่ง ดังนั้นส่วนใหญ่สัญญาณเหล่านี้จึงต้องทำให้อยู่ในรูปของข้อมูลดิจิตอล (Digital) ซึ่งจะทำให้มีคุณสมบัติต่าง ๆ ที่ดีกว่าข้อมูลอนาล็อก (Analog) ที่เคยใช้กันในอดีต และการแก้ปัญหาด้านความเร็วอีกวิธีหนึ่งก็คือการลดขนาดของข้อมูลที่มี ก็จะสามารถลดเวลาในการประมวลผลหรือการส่งข้อมูลได้ ซึ่งวิธีการแปลงแบบดิสครีตโคซายน์ ก็เป็นอีกวิธีการหนึ่งที่ถูกนำมาใช้กันอย่างกว้างขวางในการลดขนาดของข้อมูล ข้อดีอย่างหนึ่งของวิธีการแปลงแบบดิสครีตโคซายน์ก็คือ ผลที่ได้จากการแปลงซึ่งเป็นค่าสัมประสิทธิ์ในโดเมนความถี่ (Frequency Domain) จะเป็นค่าจำนวนจริงเท่านั้น ทำให้เป็นการง่ายต่อการพิจารณา อีกทั้งยังสามารถรักษาข้อมูลที่มีความสำคัญไว้ได้อย่างดีอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1.2.1 เพื่อศึกษาหลักการบีบอัดข้อมูลภาพโดยวิธีการแปลงแบบดิสครีตโคซายน์ (Discrete Cosine Transform)

1.2.2 เพื่อศึกษาและนำหลักการการแยกการแปลงดิสครีตโคซายน์ 1 มิติ จำนวน 2 ครั้ง ไปประยุกต์ใช้กับวิธีการแปลงดิสครีตโคซายน์ 2 มิติ

1.2.3 เพื่อศึกษาและนำหลักการของ CORDIC Algorithm ไปประยุกต์ใช้กับวิธีการแปลงแบบ-ดิสครีตโคซายน์ 2 มิติ

1.2.4 เพื่อศึกษาและนำหลักการของโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) ไปประยุกต์ใช้กับวิธีการแปลงแบบดิสครีตโคซายน์ 2 มิติ

1.2.5 ทำการจำลองการทำงานของวิธีการแปลงแบบดิสครีตโคซายน์ด้วยภาษา VHDL และเก็บผลการทดลองจากโปรแกรม Modelsim XE II 5.7c

1.2.6 นำผลการทดลองที่ได้จากโปรแกรม Modelsim XE II 5.7c มาเปรียบเทียบกับผลการทดลองจากโปรแกรมแมทแลบ

1.2.7 นำผลการทดลองที่ได้จากการทำงานจริงมาเปรียบเทียบกับสัญญาณต้นแบบ

1.3 ขอบเขตของปริญญานิพนธ์

จากวัตถุประสงค์ที่กล่าวมาข้างต้นจะเห็นได้ว่า ในปริญญานิพนธ์ฉบับนี้จะเป็นการศึกษาหลักการบีบอัดข้อมูลภาพขาวดำ โดยวิธีการแปลงแบบดิสครีตโคซายน์ 2 มิติ ซึ่งใช้กระบวนการการแยกการแปลงดิสครีตโคซายน์ 1 มิติ จำนวน 2 ครั้ง โดยใช้หลักการของ CORDIC Algorithm และโครงสร้างเลขคณิตกระจายมาประยุกต์ใช้ เพื่อลดตัวอุปกรณ์คูณ ซึ่งกระบวนการทำงานทั้งหมดจะถูกบรรยายพฤติกรรมการทำงานโดยใช้การเขียนด้วยภาษา VHDL ในการออกแบบ พร้อมทั้งนำผลที่ได้จากการทำงานจริงของการแปลงแบบดิสครีตโคซายน์ 2 มิติ มาเปรียบเทียบกับผลที่ได้จากการจำลองการทำงานโดยใช้โปรแกรม Xilinx – Project Navigator และผลที่ได้จากโปรแกรมแมทแลบ

1.4 เนื้อหาของปริญญานิพนธ์

ในบทที่ 2 จะกล่าวถึงทฤษฎีการทรานสฟอร์ม สมการพื้นฐานของการทรานสฟอร์ม ทฤษฎีของ CORDIC Algorithm สมการพื้นฐานของ CORDIC Algorithm การประยุกต์สมการเพื่อนำไปหาค่าในฟังก์ชันของตรีโกณมิติ (Trigonometric) ต่าง ๆ รวมถึงวิธีการคำนวณค่าวิธีการของการแปลงดิสครีตโคซายน์แบบ 2 มิติ ซึ่งเป็นการเปลี่ยนข้อมูลให้อยู่ในโดเมนความถี่ (Frequency Domain) วิธีการทรานสโพส ทฤษฎีของภาษา VHDL การนำ RAM ภายในชิพมาใช้ในส่วนของทรานสโพส ทฤษฎีของ DA (Distributed Arithmetic) นำมาใช้ร่วมเพื่อให้ง่ายต่อการประมวลผลและลดจำนวนอุปกรณ์ในการนำไปใช้งาน รวมทั้งส่วนของการสื่อสารกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม (Serial Port) เพื่อทำการรับ – ส่งข้อมูล และการนำ DCMs (Digital Clock Managers) มาทำการปรับความถี่ของสัญญาณนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบทที่ 3 กล่าวถึงวิธีการคำนวณและออกแบบการสร้างการแปลงแบบดิสครีตโคซายน์ 2 มิติ โดยใช้หลักการการแปลงดิสครีตโคซายน์ 1 มิติ จำนวน 2 ครั้ง และวิธีการนำหลักการของ CORDIC Algorithm มาประยุกต์ใช้ กับการแปลงดิสครีตโคซายน์ 2 มิติ รวมทั้งวิธีการออกแบบการส่ง – รับ ข้อมูลผ่านทางพอร์ตอนุกรม และการปรับความถี่สัญญาณนาฬิกาใหม่จากความถี่เดิมโดยใช้ DCMs

ในบทที่ 4 กล่าวถึงการทดลองและผลการทดลองที่ได้จากการคำนวณและออกแบบการสร้างการแปลงดิสครีตโคซายน์ 2 มิติ ในแต่ละส่วน ผลการทดลองที่ได้จากการทำงานจริงบนอุปกรณ์ FPGA

ในบทที่ 5 กล่าวถึงบทวิจารณ์และบทสรุป

กิตติกรรมประกาศ

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

การนำเทคนิคการบีบอัดข้อมูล (Data compression) มาใช้เพื่อเพิ่มประสิทธิภาพในกระบวนการส่งและเก็บข้อมูล ด้วยการลดจำนวนบิทให้น้อยลง โดยที่เราจะพูดถึงการบีบอัดข้อมูลที่เป็นภาพเท่านั้น (Image compression) โดยการบีบอัดข้อมูลภาพนี้สามารถแบ่งออกได้เป็น 2 แบบคือ

การบีบอัดข้อมูลแบบที่ไม่มีการสูญเสีย (Loss-less compression) วิธีการแบบนี้ข้อมูลต้นแบบก่อนที่จะทำการบีบอัดข้อมูลเมื่อเทียบกับข้อมูลที่นำกลับคืนมาที่ทางภาครับจะมีความเหมือนกันทุกประการ วิธีการนี้ใช้กับข้อมูลที่มีความสำคัญมาก ๆ โดยจะนำเอาประโยชน์ของส่วนเกินของข้อมูล (Redundancy) มาใช้

การบีบอัดข้อมูลที่ยอมให้เกิดการสูญเสียได้บ้าง (Lossy compression) วิธีนี้สามารถยอมให้เกิดความคลาดเคลื่อนหรือความผิดพลาดของข้อมูลในระดับที่ยอมรับได้ วิธีการนี้มักจะนิยมใช้กับข้อมูลภาพหรือเสียง เนื่องจากประสาทสัมผัสของมนุษย์มีขีดจำกัด เช่น ตาหรือหูของคนเราไม่สามารถที่จะแยกแยะความผิดพลาดของข้อมูลได้ละเอียดมากนัก วิธีนี้จึงอาจเรียกได้อีกอย่างหนึ่งว่า visually loss - less compression

2.1 การเข้ารหัสโดยการทรานสฟอร์ม (Transform Coding)

2.1.1 หลักการพื้นฐานของการทรานสฟอร์ม

การลดข้อมูลด้วยวิธีการทรานสฟอร์มนั้น จะทำการแปลงข้อมูลอินพุทจากโดเมนหนึ่งให้อยู่ในรูปของโดเมนใหม่ โดยใช้วิธีการแปลงแบบต่าง ๆ เช่น การแปลงฟูริเยร์ (Fourier Transform) ในการแปลงข้อมูลให้อยู่ในรูปสัมประสิทธิ์ของความถี่ จะทำให้ที่ความถี่ต่ำ ๆ มีพลังงานสูง และที่ความถี่สูง ๆ พลังงานจะลดลงไป สำหรับข้อมูลภาพโดยทั่วไปนั้นจะประกอบด้วยส่วนที่เป็นความถี่สูง คือส่วนที่เป็นรายละเอียดหรือขอบภายในภาพ และส่วนที่เป็นความถี่ต่ำ คือข้อมูลส่วนที่เป็นพื้นของภาพ ดังนั้นเมื่อต้องการลดข้อมูลให้ได้มาก ค่าของพลังงานความถี่สูงจะถูกตัดทิ้งไป ทำให้รายละเอียดส่วนที่เป็นขอบในภาพขาดหายไป ภาพที่ได้จะขาดความคมชัด

การแปลงที่ใช้ในการลดข้อมูลภาพมีอยู่หลายวิธี เช่น การแปลงฟาสฟูริเยร์(Fast Fourier) การแปลงฮาร์ (Haar Transform) การแปลงคาร์ฮูเนนเลิฟ (Karhunen Loeve Transform) การแปลงวอล์ช – ฮาดามาร์ด (Walsh - Hadamard Transform) การแปลงดิสครีตโคซายน์ (Discrete Cosine Transform) การแปลงดิสครีตซายน์ (Discrete Sine Transform) เป็นต้น ซึ่งแต่ละที่มีข้อดีข้อเสียต่างกันไป แต่ที่นิยมใช้ในการลดขนาดของข้อมูลมากที่สุด คือการแปลงดิสครีตโคซายน์ ตัวอย่างการใช้งานของการแปลงดิส-ครีตโคซายน์ คือ การลดขนาดของข้อมูลแบบ JPEG MPEG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้ารหัสการทรานสฟอร์ม ประกอบด้วย 3 ขั้นตอน คือ

1. นำชุดข้อมูลมาแบ่งออกเป็นบล็อก ๆ ขนาด $N \times N$ ซึ่งแต่ละบล็อกจะถูกนำไปทำการทรานสฟอร์มข้อมูลจากโดเมนหนึ่งไปยังโดเมนหนึ่ง ซึ่งผลที่จากการทรานสฟอร์มจะทำให้พลังงานหรือรายละเอียดของข้อมูลที่กระจายอยู่ทั่วไป ถูกบีบหรือทำให้กระชับขึ้นแล้วแทนค่าด้วยค่าค่าหนึ่ง

2. นำข้อมูลที่ได้จากการทรานสฟอร์ม มาทำการผ่านส่วนของควอนไทซ์เซชัน(Quantization) ซึ่งค่าการควอนไทซ์เซชันจะมีค่าเป็นเท่าไร จะขึ้นอยู่กับ 3 ส่วน คือ บิทเรต (bit rate) ค่าความแปรปรวนของชุดข้อมูลและข้อผิดพลาดที่เกิดขึ้นที่สามารถยอมรับได้ จากการอินเวอร์สทรานสฟอร์มเพื่อทำการสร้างกลับของข้อมูล

3. การนำค่าที่ได้จากการควอนไทซ์เซชันมาทำการเข้ารหัส โดยใช้การเข้ารหัสเลขฐาน 2 (Binary Code) เช่น การเข้ารหัสแบบจำกัดความยาว (Fixed – length code) หรือ การเข้ารหัสแบบรัน – เลนจ์ (Run – length code) และการเข้ารหัสแบบฮัฟแมน (Huffman coding)

2.1.2 สมการของการทรานสฟอร์ม

การทรานสฟอร์มทั้งหมดที่กล่าวถึงนี้จะกล่าวถึงการทรานสฟอร์มที่เป็นเชิงเส้นซึ่งสามารถเขียนสมการของการทรานสฟอร์มแบบ 1 มิติ (1D Forward Transform) ได้ดังนี้

$$\theta_n = \sum_{i=0}^{N-1} X_i a_{n,i} \tag{2.1}$$

เมื่อ θ_n เป็นลำดับการทรานสฟอร์ม (Transform Sequence)

X_i เป็นลำดับของอินพุท (Original Sequence)

$a_{n,i}$ เป็นสัมประสิทธิ์การทรานสฟอร์ม

สามารถเขียนสมการของการอินเวอร์สทรานสฟอร์มแบบ 1 มิติ (1D Inverse Transform) ได้ดังนี้

$$X_n = \sum_{i=0}^{N-1} \theta_i b_{n,i} \tag{2.2}$$

เมื่อพิจารณาสมการที่ 2.1 สามารถเขียนสมการการทรานสฟอร์ม 1 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$\theta = CX \tag{2.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพิจารณาสมการที่ 2.2 สามารถเขียนสมการการอินเวอร์สทรานสฟอร์ม 1 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$X = D\theta \tag{2.4}$$

โดย เมตริกซ์ θ, X มีขนาด $N \times 1$

เมตริกซ์ C, D มีขนาด $N \times N$

เมื่อพิจารณาข้อมูลแบบ 2 มิติ สามารถเขียนสมการของการทรานสฟอร์มแบบ 2 มิติ (2D Forward Transform) ได้ดังนี้

$$\Theta_{k,l} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X_{i,j} a_{k,i} a_{k,j} \tag{2.5}$$

เมื่อพิจารณาสมการที่ 2.5 สามารถเขียนสมการการทรานสฟอร์ม 2 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$\Theta = CXC^T \tag{2.6}$$

สามารถเขียนสมการการอินเวอร์สทรานสฟอร์ม 2 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$X = D\Theta D^T \tag{2.7}$$

โดย $D = C^{-1} = C^T$

หรือสามารถเขียนสมการการอินเวอร์สทรานสฟอร์ม 2 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$X = C^T \Theta C \tag{2.8}$$

โดย Θ เป็นเอาท์พุทของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด $N \times N$

C, D เป็นสัมประสิทธิ์ของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด $N \times N$

X เป็นอินพุทของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด $N \times N$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 การแปลงแบบดิสครีตโคซายน์ (Discrete Cosine Transform)

ดิสครีตโคซายน์ทรานสฟอร์มคือการแปลงข้อมูลที่อยู่ในสแปเชียล (Spatial Domain) ให้อยู่ในรูปแบบของโดเมนความถี่ เพื่อทำให้ค่าที่ได้ออกมาเป็นอิสระต่อกัน และลดขนาดของข้อมูลลงให้ค่าพลังงานส่วนใหญ่ของข้อมูลอยู่ในช่วงความถี่ต่ำ โดยดิสครีตโคซายน์ทรานสฟอร์ม จะเป็นการแปลงแบบเชิงเส้นที่มีค่าสัมประสิทธิ์ในการแปลงคงที่ และค่าที่ได้ในโดเมนความถี่จะมีเฉพาะจำนวนจริงเท่านั้น ซึ่งข้อมูลส่วนใหญ่ที่จะใช้การแปลงแบบดิสครีตโคซายน์จะเป็นข้อมูลจากภาพนิ่ง หรือเป็นภาพเคลื่อนไหว อินพุทที่เข้ามาจะถูกแยกเป็นบล็อกเล็กๆ โดยส่วนมากแล้วขนาดของบล็อกจะมีค่าเป็นเลขยกกำลังสอง เช่น 4, 8, 16

ในกรณีของการแปลงดิสครีตโคซายน์ ค่าสัมประสิทธิ์ของการทรานสฟอร์มสามารถหาได้ดังนี้

$$[C]_{i,j}=\begin{cases}\sqrt{\frac{1}{N}}\cos\frac{(2j+1)i\pi}{2N} & i=0,\quad j=0,1,\ldots,N-1\\ \sqrt{\frac{2}{N}}\cos\frac{(2j+1)i\pi}{2N} & i=1,2,\ldots,N-1,\quad j=0,1,..,N-1\end{cases} \tag{2.9}$$

เมื่อ N คือ ขนาดของบล็อก $\left(N=2^n\ ,n=2,3,4,\ldots\right)$

จากสมการที่ 2.9 สามารถเขียนค่าสัมประสิทธิ์ของการแปลงดิสครีตโคซายน์ให้อยู่ในรูปของเมตริกซ์ ขนาด $N\times N$ เมื่อ $N=8$ ได้ดังนี้

$$C=\begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \cdots & \frac{1}{\sqrt{2}}\\ \cos\left(\frac{\pi}{16}\right) & \cos\left(\frac{3\pi}{16}\right) & \cdots & \cos\left(\pi-\frac{\pi}{16}\right)\\ \cdots & \cdots & \cdots & \cdots\\ \cos\left(\frac{7\pi}{16}\right) & \cos\left(\pi+\frac{5\pi}{16}\right) & \cdots & \cos\left(7\pi-\frac{7\pi}{16}\right)\end{bmatrix} \tag{2.10}$$

จากค่าสัมประสิทธิ์ของการทรานสฟอร์มในสมการที่ 2.10 สามารถเขียนแทนค่าฟังก์ชันตรีโกณมิติดังกล่าวด้วยตัวแปรได้ดังตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 แสดงค่าสัมประสิทธิ์ของการทรานสฟอร์มในรูปแบบของฟังก์ชันตรีโกณมิติ

ค่าสัมประสิทธิ์	ฟังก์ชันโคไซน์ (cosine)
C_1	$\cos\left(\frac{\pi}{16}\right)$
C_2	$\cos\left(\frac{2\pi}{16}\right)$
C_3	$\cos\left(\frac{3\pi}{16}\right)$
C_4	$\cos\left(\frac{4\pi}{16}\right)$
C_5	$\cos\left(\frac{5\pi}{16}\right)$
C_6	$\cos\left(\frac{6\pi}{16}\right)$
C_7	$\cos\left(\frac{7\pi}{16}\right)$

จากสมการที่ 2.3 เมื่อแทนค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ จากสมการที่ 2.10 แล้ว สามารถเขียนแสดงในรูปเมตริกซ์ได้ดังนี้

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \end{bmatrix} = \begin{bmatrix} c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 \\ c_1 & c_3 & c_5 & c_7 & -c_7 & -c_5 & -c_3 & -c_1 \\ c_2 & c_6 & -c_6 & -c_2 & -c_2 & -c_6 & c_6 & c_2 \\ c_3 & -c_7 & -c_1 & -c_5 & c_5 & c_1 & c_7 & -c_3 \\ c_4 & -c_4 & -c_4 & c_4 & c_4 & -c_4 & -c_4 & c_4 \\ c_5 & -c_1 & c_7 & c_3 & -c_3 & -c_7 & c_1 & -c_5 \\ c_6 & -c_2 & c_2 & -c_6 & -c_6 & c_2 & -c_2 & c_6 \\ c_7 & -c_5 & c_3 & -c_1 & c_1 & -c_3 & c_5 & -c_7 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

สมการที่ใช้สำหรับการแปลงดิสครีตโคซายน์ (Forward Discrete Cosine Transform) ยังสามารถเขียนสมการให้อยู่ในรูปของผลบวกของผลคูณของอินพุทกับค่าสัมประสิทธิ์ได้ดังนี้

หนึ่งมิติ

$$y(n) = e(n)\sum_{k=0}^{N-1} x(k)\cos\left[\frac{(2k+1)n\pi}{2N}\right] \quad ; \; n = 0,1,...,N-1 \qquad (2.11)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สองมิติ

$$y(m,n)=\frac{4}{N^2}e(m)e(n)\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}x(k,l)\cos\left[\frac{(2k+1)n\pi}{2N}\right]\cos\left[\frac{(2l+1)n\pi}{2N}\right] \quad (2.12)$$

$m = 0,1,...,N-1$

$n = 0,1,...,N-1$

$$e(n)=\begin{cases}\frac{1}{\sqrt{2}} & ;\ n=0\\ 1 & ;\ etc\end{cases}$$

$$e(m)=\begin{cases}\frac{1}{\sqrt{2}} & ;\ m=0\\ 1 & ;\ etc\end{cases}$$

ส่วนสมการการแปลงกลับของดิสครีตโคซายน์ (Inverse Discrete Cosine Transform) สามารถแสดงได้ดังนี้

หนึ่งมิติ

$$x(k)=\frac{2}{N}\sum_{n=0}^{N-1}e(n)y(n)\cos\left\{\frac{(2k-1)n\pi}{2N}\right\};\ k=0,1,...,N-1 \quad (2.13)$$

สองมิติ

$$x(k,l)=\frac{2}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}e(m)e(n)y(m,n)\cos\left\{\frac{(2k-1)m\pi}{2N}\right\}\cos\left\{\frac{(2l-1)n\pi}{2N}\right\} \quad (2.14)$$

$k = 0,1,...,N-1$

$l = 0,1,...,N-1$

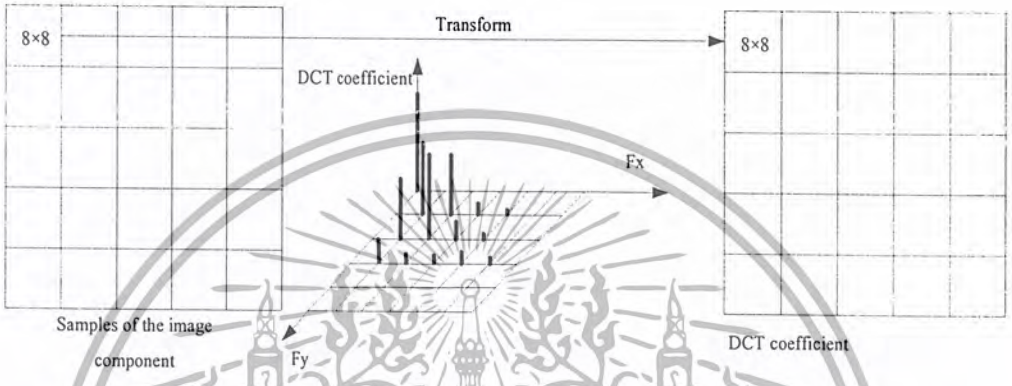
$$e(n)=\begin{cases}\frac{1}{\sqrt{2}} & ;\ n=0\\ 1 & ;\ etc\end{cases}$$

$$e(m)=\begin{cases}\frac{1}{\sqrt{2}} & ;\ m=0\\ 1 & ;\ etc\end{cases}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ส่วนของการแปลงดิสครีตโคซายน์ 2 มิติ (2 Dimensions Discrete Cosine Transform)

ปริญญานิพนธ์ฉบับนี้จะทำการศึกษาข้อมูลภาพโดยที่ยกตัวอย่างของข้อมูลภาพขาวดำ ซึ่งจะถูกแบ่งออกเป็นบล็อกย่อยๆ ขนาด 8×8 พิกเซล เพื่อที่จะทำการแปลงดิสครีตโคซายน์แบบ 2 มิติ (2D Discrete Cosine Transform) ทีละชุดจนครบทุกบล็อกทั้งหมดในข้อมูลภาพ ผลของการทรานสฟอร์มแต่ละบล็อกจะได้ข้อมูลที่มีความสำคัญสูง (เทอมความถี่ต่ำ) อยู่ในด้านมุมซ้ายบนโดยที่ตำแหน่ง (0,0) เป็นองค์ประกอบของ DC ส่วนที่เหลือเป็นข้อมูลที่มีความสำคัญต่ำ (เทอมความถี่สูง) เป็นองค์ประกอบของ AC กระจายออกไปตามแกน x และ y ดังที่แสดงในรูปที่ 2.1



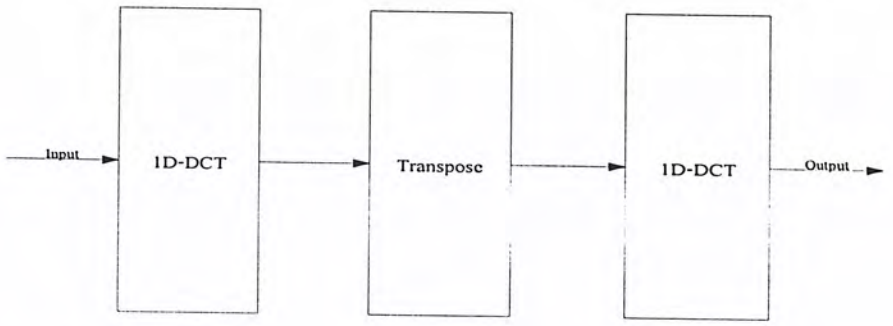
รูปที่ 2.1 แสดงลักษณะของการแปลงดิสครีตโคซายน์ของบล็อกข้อมูลขนาด 8×8 พิกเซล จากข้อมูลทั้งหมด

ซึ่งจากผลการแปลงดิสครีตโคซายน์ ของข้อมูลภาพแต่ละบล็อกที่แสดงในรูปที่ 2.1 จะเห็นว่าผลที่ได้ข้อมูลที่มีความสำคัญสูง (มีค่ามาก) อยู่ในเทอมความถี่ต่ำ ส่วนเทอมความถี่สูง (มุมล่างด้านขวาของเมตริกซ์) จะมีค่าของข้อมูลน้อยมากเมื่อเทียบกับเทอมความถี่ต่ำที่มีความสำคัญมากกว่า แต่ก็ยังคงมีความสำคัญอยู่บ้าง ถ้านำข้อมูลที่ได้จากการแปลงดิสครีตโคซายน์ ไปผ่านการควอนไทซ์ ทำให้ข้อมูลบริเวณความถี่สูงมีค่าเข้าใกล้ศูนย์ ช่วยให้สามารถลดขนาดของข้อมูลลงได้อีก

เมื่อมองในเชิงของฮาร์ดแวร์ การทำเป็นตัวอุปกรณ์ของการแปลงดิสครีตโคซายน์แบบ 2 มิติ ที่มีขนาด $N \times N$ จะต้องใช้ตัวคูณและตัวบวกรวมกันจำนวนมาก แต่สามารถทำการลดจำนวนตัวคูณและตัวบวกลงได้ด้วยการแยกจากการแปลงดิสครีตโคซายน์แบบ 2 มิติ ให้มาเป็นการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง โดยอาศัยคุณสมบัติที่สำคัญของการแปลงดิสครีตโคซายน์ก็คือ คุณสมบัติ Separable Transform ซึ่งจะประกอบด้วยการแปลงแนวนอนของเมตริกซ์ แล้วตามด้วยการแปลงแนวตั้งของเมตริกซ์ ตามลำดับ

การแยกการแปลงดิสครีตโคซายน์แบบ 2 มิติ ให้มาเป็นการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง อธิบายโดยใช้บล็อกไดอะแกรมดังแสดงในรูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

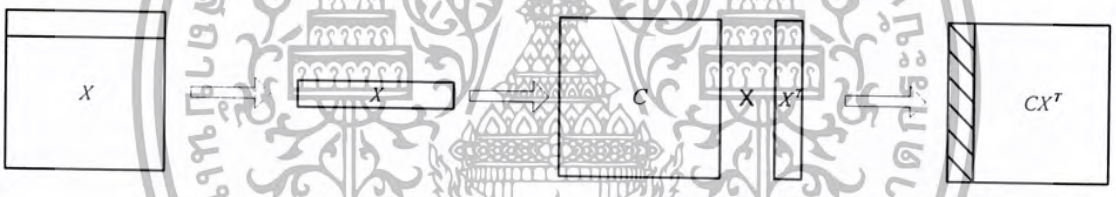


รูปที่ 2.2 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 2 มิติ

การแยกการแปลงดิสครีตโคซายน์แบบ 2 มิติ ให้มาเป็นการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง สามารถอธิบายการทำงาน ได้ดังนี้

1. การแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวนอนของเมตริกซ์

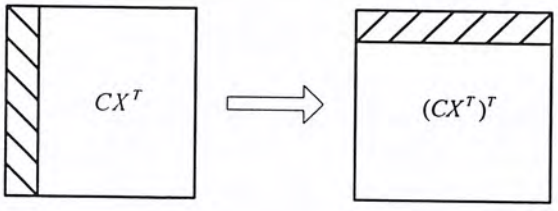
โดยนำข้อมูลตามแนวนอน 1 มิติ ขนาด $1 \times N$ จากข้อมูลเริ่มต้น 2 มิติ ขนาด $N \times N$ มาทำการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทีละแถว จนครบทั้งหมด N แถว สามารถแสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวนอนในรูปของเมตริกซ์ได้ดังรูปที่ 2.3



รูปที่ 2.3 แสดงบล็อกไดอะแกรมการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวนอน

2. การทำการทรานสโพสบล็อกข้อมูลที่ได้จากการแปลงดิสครีตโคซายน์แบบ 1 มิติ

โดยนำข้อมูลตามแนวนอนที่ได้จากการแปลงดิสครีตโคซายน์แบบ 1 มิติทั้ง 64 ค่า มาทำการทรานสโพส เพื่อนำเข้าไปเป็นอินพุทของการแปลงดิสครีตโคซายน์แบบ 1 มิติตามแนวตั้ง สามารถแสดงบล็อกไดอะแกรมของการทรานสโพส ในรูปของเมตริกซ์ได้ดังรูปที่ 2.4

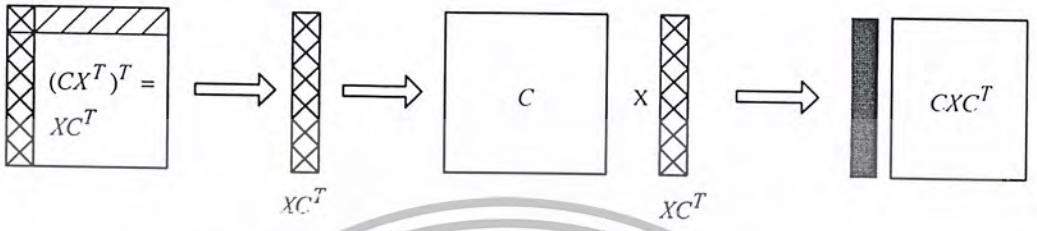


รูปที่ 2.4 แสดงบล็อกไดอะแกรมการทรานสโพสบล็อกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวตั้งของเมตริกซ์

โดยนำข้อมูลตามแนวตั้ง 1 มิติ ขนาด $1 \times N$ จากข้อมูลที่ได้จากการแปลงดิสครีตโคซายน์แบบ 1 มิติ ในส่วนที่ 1 โดยเริ่มต้นที่ 2 มิติ ขนาด $N \times N$ มาทำการแปลงดิสครีตโคซายน์ 1 มิติทีละหลัก จนครบทั้งหมด N หลัก สามารถแสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวตั้งในรูปของเมตริกซ์ได้ดังรูปที่ 2.5



รูปที่ 2.5 แสดงบล็อกไดอะแกรมการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวตั้ง

ผลลัพธ์ที่ได้จากการแปลงดิสครีตโคซายน์แบบ 2 มิติ โดยใช้การแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง คือ

$$\begin{aligned} Y &= C\left(CX^T\right)^T \\ Y &= C\left(X^T\right)^T C^T \\ Y &= CXC^T \end{aligned} \tag{2.15}$$

จะเห็นได้ว่าผลลัพธ์ที่ได้จากการแปลงดิสครีตโคซายน์แบบ 2 มิติ โดยใช้การแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง ดังแสดงในสมการที่ 2.15 ได้ผลลัพธ์เท่ากับการแปลงดิสครีตโคซายน์แบบ 2 มิติ ในสมการที่ 2.6

2.3 ส่วนของการทรานสโพส(Transpose)

การทรานสโพส เป็นฟังก์ชันทางคณิตศาสตร์ที่เป็นตัวกระทำทางเมตริกซ์ชนิดหนึ่ง การทรานสโพส เป็นการสลับตัวบ่งชี้ (Index) ของเมตริกซ์ ระหว่างตัวบ่งชี้ของแถว (Row) กับตัวบ่งชี้ของหลัก (Column) สามารถแสดงได้ดังนี้

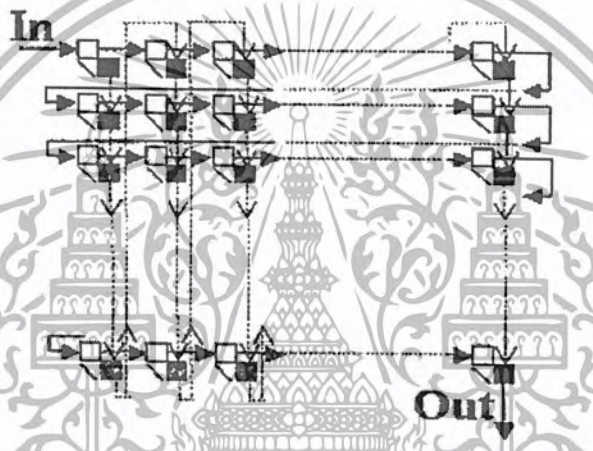
$$A_{(i,j)} = B^T{}_{(j,i)} \tag{2.16}$$

ส่วนทรานสโพส ทำหน้าที่เปลี่ยนตำแหน่งข้อมูลหลังผ่านการแปลงดิสครีตโคซายน์แบบ 1 มิติ ในส่วนแรกทั้ง 64 ค่า ก่อนที่จะนำไปเป็นอินพุทของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ในส่วนถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการประยุกต์ฟังก์ชันการทรานสโพสลงบนฮาร์ดแวร์นั้นเป็นเรื่องที่มีความยุ่งยากมากและไม่สามารถทำการทรานสโพสได้โดยตรงเหมือนกับการทรานสโพสบนซอฟแวร์ จึงมีแนวคิดมากมายที่จะพัฒนาเทคนิคต่างๆ เพื่อประยุกต์ฟังก์ชันการทรานสโพสลงบนฮาร์ดแวร์ โดยอาศัยคุณลักษณะของอุปกรณ์ต่างๆ เข้าช่วย เช่น แรม (RAM) รีจิสเตอร์ (Register) ตัวอย่างของการประยุกต์ฟังก์ชัน การทราน-สโพสลงบนฮาร์ดแวร์สามารถแสดงได้ดังนี้

ทรานสโพสบัฟเฟอร์ (Transpose Buffer) ในตัวอย่างนี้ทำงานโดยอาศัยคุณสมบัติของรีจิสเตอร์ (Register) ทรานสโพสบัฟเฟอร์นี้จะประกอบด้วย บัฟเฟอร์แถว (Row Buffer) และบัฟเฟอร์หลัก (Column Buffer) โดยที่แต่ละบัฟเฟอร์ประกอบด้วย 64 รีจิสเตอร์ ที่ไม่มีความสัมพันธ์ต่อกันและกัน(Independent) สามารถแสดงรูปของทรานสโพสบัฟเฟอร์ได้ดังรูปที่ 2.6



รูปที่ 2.6 แสดงการทรานสโพสโดยอาศัยคุณสมบัติของรีจิสเตอร์

โดยค่าอินพุทของทรานสโพสบัฟเฟอร์ (Transpose Buffer) จะรับเข้ามาเก็บในส่วนของบัฟเฟอร์หลัก (Column Buffer) และจะนำค่าเอาท์พุทออกจากทรานสโพสบัฟเฟอร์ในส่วนของบัฟเฟอร์แถว (Row Buffer) หลักการทำงานของทรานสโพสบัฟเฟอร์ คือ เมื่อข้อมูลอินพุทที่เก็บอยู่ในบัฟเฟอร์หลักจนเต็มทั้ง 64 รีจิสเตอร์ ก็จะถ่ายข้อมูล (Transfer) ไปเก็บในบัฟเฟอร์แถวทั้ง 64 รีจิสเตอร์ หลังจากนั้นค่อยส่งข้อมูลเอาท์พุทออกจากทรานสโพสบัฟเฟอร์ ในส่วนของบัฟเฟอร์แถว ซึ่งทำให้ในขณะที่ส่งข้อมูลเอาท์พุทออกนั้นสามารถรับข้อมูลอินพุทใหม่ได้ทันทีในส่วนของบัฟเฟอร์หลัก โดยไม่ต้องรอให้ส่งข้อมูลเอาท์พุทออกจนหมดก่อน จึงใช้เวลาสำหรับการทรานสโพส 1 บล็อก เท่ากับ 64 รอบของสัญญาณนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 CORDIC Algorithm

เมื่อพิจารณาสมการที่ 2.9 ซึ่งเป็นสมการการหาค่าสัมประสิทธิ์ของการแปลงดิสครีตโคซายน์ เห็นได้ว่าสมการดังกล่าวเป็นฟังก์ชันของตรีโกณมิติ จึงเกิดความสนใจที่จะหาอัลกอริทึม (Algorithm) บางอย่างที่สามารถจะคำนวณหาค่าของฟังก์ชันตรีโกณมิติได้ ซึ่งในปริญญานิพนธ์ฉบับนี้ได้ใช้อัลกอริทึมของคอร์ดิก (CORDIC Algorithm)

CORDIC (Coordinate Rotation Digital Computer) เป็นทฤษฎีของการหาค่าในฟังก์ชันในตรีโกณมิติเช่น sine cosine tangent arcsine arccosine และ arctangent เป็นต้น ซึ่งจะทำการคำนวณค่าโดยวิธีการหมุนเวคเตอร์ในระบบของพิกัดฉาก ซึ่งให้ค่าผลลัพธ์ของขนาดหรือมุมของของเวคเตอร์นั้นๆ ออกมา การคำนวณจำเป็นต้องมีการกำหนดค่าเริ่มต้นที่ค่า ๆ หนึ่งแล้วจึงค่อย ๆ ทำการลดค่าหรือวนรอบในการเปลี่ยนแปลงค่าไปเรื่อยๆจนได้ค่าที่ต้องการ โดยจะพิจารณา Vector Diagram ดังที่แสดงในรูปที่ 2.7

รูปที่ 2.7 การหมุนของเวคเตอร์ในพิกัดฉาก

จากรูปที่ 2.7 กำหนดให้ R คือขนาดของเวคเตอร์ θ คือมุมของเวคเตอร์ x และ y คือตำแหน่งของเวคเตอร์ในพิกัดฉาก ซึ่งจะได้ความสัมพันธ์คือ

$$\tan\theta = \frac{y_i}{x_i}$$

$$\tan(\theta + \Delta\theta_i) = \frac{y_{i+1}}{x_{i+1}} \qquad (2.17)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากเอกลักษณ์ของตรีโกณมิติ

$$\tan(x \pm y) = \frac{\tan x \pm \tan y}{1 \mp \tan x \tan y}$$

$$\tan(\theta + \Delta\theta_i) = \frac{\tan\theta \pm \tan\Delta\theta_i}{1 \mp \tan\theta \tan\Delta\theta_i}$$

เขียนสมการที่ 2.17 ใหม่ได้เป็น

$$\frac{y_{i+1}}{x_{i+1}} = \frac{\dfrac{y_i}{x_i} + \tan\Delta\theta_i}{1 - \left(\dfrac{y_i}{x_i}\right)\tan\Delta\theta_i}$$

นำ $\frac{x_i}{y_i}$ คูณเศษและส่วนด้านขวาของสมการตลอดแล้วทำการจัดรูปใหม่

$$\frac{y_{i+1}}{x_{i+1}} = \frac{y_i + x_i \tan\Delta\theta_i}{x_i - y_i \tan\Delta\theta_i} \tag{2.18}$$

จากสมการที่ 2.18 สามารถเขียนใหม่ได้

$$\left[\frac{y_{i+1}}{x_{i+1}}\right]^2 = \left[\frac{y_i + x_i \tan\Delta\theta_i}{x_i - y_i \tan\Delta\theta_i}\right]^2$$

$$y_{i+1}^2 = x_{i+1}^2\left[\frac{y_i + x_i \tan\Delta\theta_i}{x_i - y_i \tan\Delta\theta_i}\right]^2 \tag{2.19}$$

เมื่อพิจารณาเฉพาะขนาดเวคเตอร์

$$y_{i+1}^2 + x_{i+1}^2 = y_i^2 + x_i^2 \tag{2.20}$$

จากสมการที่ 2.19 และ 2.20

$$y_i^2 + x_i^2 = x_{i+1}^2 + x_{i+1}^2\left[\frac{y_i + x_i \tan\Delta\theta_i}{x_i - y_i \tan\Delta\theta_i}\right]$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คูณตลอดด้วย $(x_i - y_i \tan\Delta\theta_i)^2$

$$\begin{aligned}(x_i - y_i \tan\Delta\theta_i)^2\left(y_i^2 + x_i^2\right) &= (x_i - y_i \tan\Delta\theta_i)^2 x_{i+1}^2 + (y_i + x_i \tan\Delta\theta_i)^2 x_{i+1}^2 \\ &= x_{i+1}^2\left(x_i^2 + x_i^2 \tan^2\Delta\theta_i + y_i^2 \tan^2\Delta\theta_i + y_i^2\right) \\ &= x_{i+1}^2\left(x_i^2 + y_i^2\right)\left(1 + \tan^2\Delta\theta_i\right) \qquad (2.21)\end{aligned}$$

$$\therefore \quad x_{i+1} = \frac{x_i - y_i \tan\Delta\theta_i}{\sqrt{1 + \tan^2\Delta\theta_i}}$$

จากสมการที่ 2.19 $$y_{i+1} = x_{i+1}\left[\frac{y_i + x_i \tan\Delta\theta_i}{x_i - y_i \tan\Delta\theta_i}\right]$$

แทน x_{i+1} จากสมการที่ 2.21 จะได้

$$\begin{aligned}y_{i+1} &= \frac{x_i - y_i \tan\Delta\theta_i}{\sqrt{1 + \tan^2\Delta\theta_i}}\left[\frac{y_i + x_i \tan\Delta\theta_i}{x_i - y_i \tan\Delta\theta_i}\right] \\ &= \frac{y_i - x_i \tan\Delta\theta_i}{\sqrt{1 + \tan^2\Delta\theta_i}} \qquad (2.22)\end{aligned}$$

นำเทอมส่วนของของสมการที่ 2.22 มาพิจารณา

$$\begin{aligned}\sqrt{1 + \tan^2\Delta\theta_i} &= \sqrt{1 + \frac{\sin^2\Delta\theta_i}{\cos^2\Delta\theta_i}} \\ &= \left(\frac{\cos^2\Delta\theta_i + \sin^2\Delta\theta_i}{\cos^2\Delta\theta_i}\right)^{1/2} \qquad (2.23) \\ &= \left(\frac{1}{\cos\Delta\theta_i}\right)\end{aligned}$$

จากสมการที่ 2.23 เขียนสมการที่ 2.21 และ 2.22 ใหม่ได้เป็น

$$\left.\begin{aligned}x_{i+1} &= x_i \cos\Delta\theta_i - y_i \sin\Delta\theta_i \\ y_{i+1} &= y_i \cos\Delta\theta_i + x_i \sin\Delta\theta_i\end{aligned}\right\} \qquad (2.24)$$

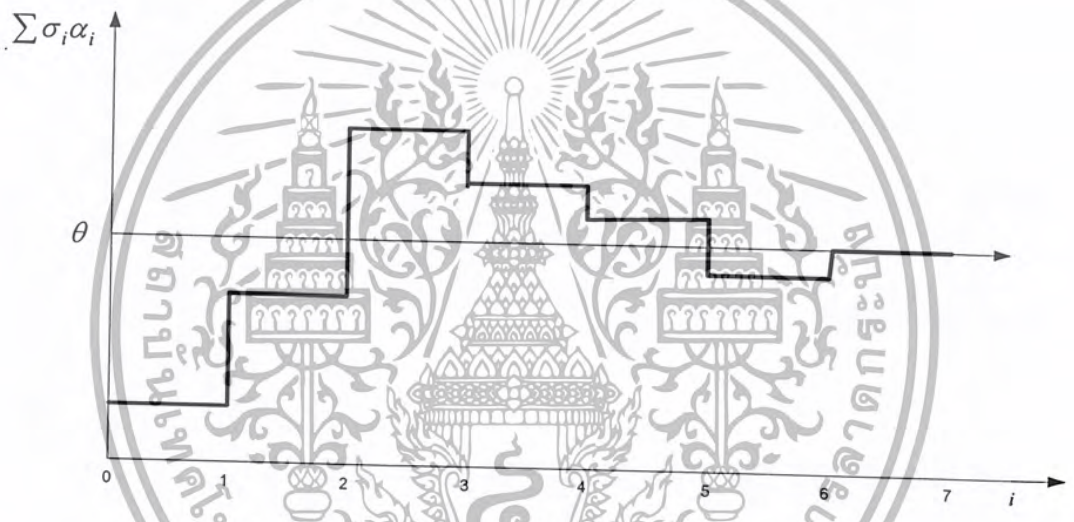
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เห็นได้ว่าสามารถหาตำแหน่งของเวคเตอร์อีกตำแหน่งหนึ่งได้โดยผ่านความสัมพันธ์จากสมการนี้ ซึ่งวิธีการของ CORDIC จะถูกกำหนดค่าการหมุนโดย $\Delta\theta_i$ และถ้าหากให้ลำดับในการหมุนมีค่าเท่ากับ α_i โดยมีการหมุนทั้งหมด n ครั้งและแต่ละครั้งถูกกำหนดเครื่องหมายโดย σ_i ดังนั้นจะทำให้ได้ค่าของ $\Delta\theta_i$ ตามสมการข้างล่าง

$$\Delta\theta_i = \sum_{i=0}^{n-1} \sigma_i \alpha_i \quad ;\sigma_i \in \{-1,1\} \tag{2.25}$$

โดยที่ σ_i จะมีการเปลี่ยนแปลงตามค่ามุมของแต่ละครั้งที่มีการหมุนเป็นตำแหน่ง α_i ซึ่งจะขึ้นอยู่กับผลของค่าผลรวมก่อนหน้านั้นเพื่อพิจารณาค่าของ $\Delta\theta_i$ ไปในทิศทางเดียวโดยเปรียบเทียบกับค่าของ $\Delta\theta_i$ ดังรูปที่ 2.8



รูปที่ 2.8 ผลการกระทำของค่า σ_i ที่เกิดกับค่า θ

จากรูปที่ 2.8 จะเห็นว่าค่าผลรวมของ $\alpha_i \left(\sum \sigma_i \alpha_i\right)$ จะมีค่าที่เป็นไปได้ทั้งช่วงที่มากกว่าและน้อยกว่าค่าของมุม θ และเมื่อจำนวนค่า i สูงขึ้นเรื่อย ๆ นั่นคือเกิดการหมุนของตำแหน่งเวคเตอร์จำนวนหลายครั้งขึ้นก็จะได้ผลรวมของ α_i จนเท่ากับค่าของมุม θ ซึ่งจะเห็นได้ว่า σ_i แต่ละครั้งเป็นตัวแปรสำคัญที่จะกำหนดให้ค่าผลรวมมีทิศทางเข้าหาค่าของ θ และจากวิธีการนี้เองจะได้สมการที่มาช่วยในการพิจารณาเครื่องหมาย(ค่าของ σ_i) คือ

$$z_{i+1} = z_i - \sum_{i=0}^{n-1} \sigma_i \alpha_i \tag{2.26}$$

โดย
$$\sigma_i = \begin{cases} +1 & z_i \geq 0 \\ -1 & z_i < 0 \end{cases}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
61917

พิจารณาจากสมการที่ 2.24 จะเห็นว่าค่าของ $\Delta\theta_i$ ก็คือ α_i ที่ตำแหน่ง i ค่าต่างๆนั่นเอง ซึ่งสามารถเขียนสมการใหม่ได้เป็น

$$\left.\begin{aligned} x_{i+1} &= x_i \cos\alpha_i - \sigma_i y_i \sin\alpha_i \\ y_{i+1} &= y_i \cos\alpha_i + \sigma_i x_i \sin\alpha_i \end{aligned}\right\} \tag{2.27}$$

เพื่อให้ง่ายต่อการคำนวณจะใช้การประมาณค่าของ $\tan\alpha_i$ เทียบเคียงกับอนุกรมกำลังของ 2 ดังสมการที่ 2.28

$$\tan\alpha_i = 2^{-i} \quad ; \; i = 0,1,2,..,n-1 \tag{2.28}$$

จากสมการที่ 2.27 และ 2.28 จะได้ว่า

$$x_{i+1} = \cos\alpha_i \left[x_i - \sigma_i y_i \tan\alpha_i \right]$$
$$y_{i+1} = \cos\alpha_i \left[y_i + \sigma_i x_i \tan\alpha_i \right]$$

ดังนั้น

$$\left.\begin{aligned} x_{i+1} &= k_i \left[x_i - \sigma_i 2^{-i} y_i \right] \\ y_{i+1} &= k_i \left[y_i + \sigma_i 2^{-i} x_i \right] \end{aligned}\right\} \tag{2.29}$$

และเมื่อ $k_i = \cos\alpha_i$ ทำการจัดอยู่ในรูปของเทอม 2^{-i} ได้ดังนี้

$$\begin{aligned} k_i &= \cos\alpha_i \\ &= \frac{1}{\sqrt{\frac{1}{\cos^2\alpha_i}}} \\ &= \frac{1}{\sqrt{\frac{\cos^2\alpha_i + \sin^2\alpha_i}{\cos^2\alpha_i}}} \\ &= \frac{1}{\sqrt{1+\tan^2\alpha_i}} \\ &= \frac{1}{\sqrt{1+2^{-i}}} \end{aligned} \tag{2.30}$$

ซึ่งเมื่อกระทำการวนซ้ำหรือหมุนเวคเตอร์ไปแต่ละค่าเรื่อยๆนำค่า k_i มาแยกพิจารณาก็จะได้สัมประสิทธิ์ใหม่เป็น

$$\begin{aligned} k &= \prod_{i=0}^{n-1} \frac{1}{\sqrt{1+2^{-i}}} \\ &\cong 0.607253 \end{aligned} \tag{2.31}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น จะพิจารณาเฉพาะเทอมของตัวแปรที่จะนำมาใช้ในการวนซ้ำจากสมการที่ 2.26 และ 2.29 จะเขียนความสัมพันธ์ใหม่ได้เป็น

$$\begin{aligned} x_{i+1} &= x_i - \sigma_i 2^{-i} y_i \\ y_{i+1} &= y_i + \sigma_i 2^{-i} x_i \\ z_{i+1} &= z_i - \sigma_i \tan^{-1}\left(2^{-i}\right) \end{aligned} \tag{2.32}$$

ซึ่งจากสมการที่ 2.32 นี้สามารถคำนวณหาค่าผลลัพธ์ที่ต้องการโดยการป้อน x_0, y_0 และ z_0 เป็นค่าเริ่มต้นให้ระบบโดยที่ค่าของ z_0 จะต้องอยู่ในช่วงของ $\pm\frac{\pi}{2}$ เท่านั้นซึ่งถ้าค่าของ z_0 นอกเหนือไปจากนี้อาจจะนำคุณสมบัติของตรีโกณมิติมาช่วยดังที่แสดงในสมการข้างล่างนี้

$$\begin{aligned} \cos\left(z \pm 2\pi\right) &= \cos z \quad , \sin\left(z \pm 2\pi\right) = \sin z \\ \cos\left(z - \pi\right) &= -\cos z \, , \sin\left(z - \pi\right) = -\sin z \end{aligned} \tag{2.33}$$

เมื่อทำการกำหนดหรือป้อนค่าเริ่มต้นแล้วทำการคำนวณค่าออกมาก็จะได้ผลดังตัวอย่างในตารางที่ 2.2 ซึ่งกำหนดให้ $z_0 = 30°$, $x_0 = k$ และ $y_0 = 0$

ตารางที่ 2.2 แสดงการคำนวณหาค่าของ sine และ cosine

i	σ_i	$x_{i+1} \to \cos z_0$	$y_{i+1} \to \sin z_0$	$z_{i+1} \to 0$
		k =0.607253	0.000000	30.0000000
0	1	0.607253	0.607253	-15.00000
1	-1	0.910880	0.303627	11.565051
2	1	0.834973	0.531347	-2.471192
3	-1	0.901391	0.426975	4.653824
4	1	0.874705	0.483312	1.077490
5	1	0.859602	0.510647	-0.712420
6	-1	0.867581	0.497216	0.182754
7	1	0.863697	0.503994	-0.264860
8	-1	0.865666	0.500620	-0.041049

โดยที่ค่าจริงของ $\cos 30° = 0.866025$ และ $\sin 30° = 0.500000$ นอกจากนี้ยังสามารถที่จะหาค่าอื่นๆได้จากการพิจารณาจากผลลัพธ์ของเวคเตอร์ที่ได้จากการหมุนโดยต้องมีตัวใดตัวหนึ่งที่จะถูกนำมาเปรียบเทียบในการพิจารณาเครื่องหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 หลักการเบื้องต้นของโครงสร้างเลขคณิตกระจาย

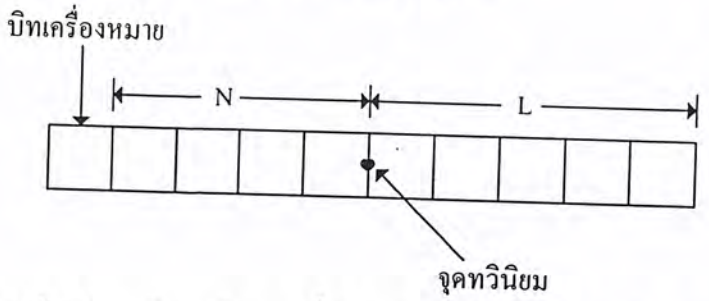
โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) หรือเรียกอย่างย่อๆว่า "DA" เป็นการจัดรูปแบบทางคณิตศาสตร์ของสมการที่อยู่ในรูปของผลบวกของผลคูณ (Sum of Product) ของเลขฐานสิบให้กระจายออกเป็นบิทหรือในรูปแบบของเลขฐานสอง เพื่อให้การคำนวณทางคณิตศาสตร์สามารถแปลงเป็นวงจรดิจิตอลอิเล็กทรอนิกส์ได้ โดยจะปรากฏอยู่ในงานทางด้านการประมวลผลสัญญาณเชิงเลข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

2.5.1 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

1. รูปแบบจำนวนโดยตรง

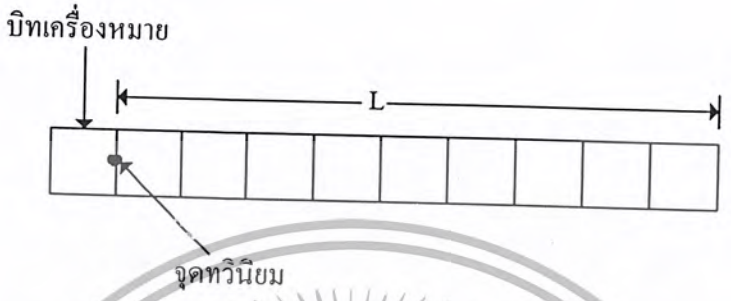
รูปแบบจำนวนโดยตรงปกติจะประกอบไปด้วย 3 ส่วน คือ บิทเครื่องหมาย (Sign bit) 1 บิท บิทจำนวนเต็ม (Integer bit) N บิท และบิทเศษส่วน (Fractional bit) L บิท โดยจะมีจุดทวินิยม (Binary point) อยู่ระหว่างบิทจำนวนเต็มและบิทเศษส่วนดังแสดงในรูปที่ 2.9



รูปที่ 2.9 แสดงการจัดรูปแบบจำนวนโดยตรงที่ประกอบด้วยบิทเครื่องหมาย บิทจำนวนเต็ม และบิทเศษส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนบิท N เป็นตัวกำหนดย่านพลวัตรที่ต้องการ โดยถ้าเลือกให้มีจำนวนน้อยอาจทำให้เกิดการล้น (Overflow) จากการคำนวณได้ แต่ถ้าเลือกให้มีจำนวนมากความเที่ยงตรงก็จะน้อยลง ซึ่งในการสร้างวงจรกรองสัญญาณเชิงเลขโดยการแทนด้วยรูปแบบจำนวนโดยตรงนั้น นิยมที่จะทำมาตราส่วน (Scaling) เพื่อให้ขนาดของสัญญาณมีค่าอยู่ระหว่าง $-1 \le x < 1$ คือมีบิทเครื่องหมาย 1 บิท และบิทเศษส่วน L บิท ดังแสดงในรูปที่ 2.10



รูปที่ 2.10 แสดงการจัดรูปแบบจำนวนโดยตรงที่มีบิทเครื่องหมายและบิทเศษส่วน

โดยทั่วไปเลขฐานสองแบบจำนวนโดยตรงแบ่งออกได้เป็น 3 รูปแบบด้วยกัน คือ

1. แบบขนาดและเครื่องหมาย (Sign magnitude)
2. แบบส่วนเติมเต็มหนึ่ง (1's complement)
3. แบบส่วนเติมเต็มสอง (2's complement)

โดยคุณลักษณะที่สำคัญบางประการของการแทนตัวเลขด้วยเลขฐานสองแบบจำนวนโดยตรงทั้ง 3 รูปแบบสามารถสรุปได้ดังตารางที่ 2.3

ตารางที่ 2.3 แสดงคุณสมบัติที่สำคัญของรูปแบบจำนวนโดยตรง

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

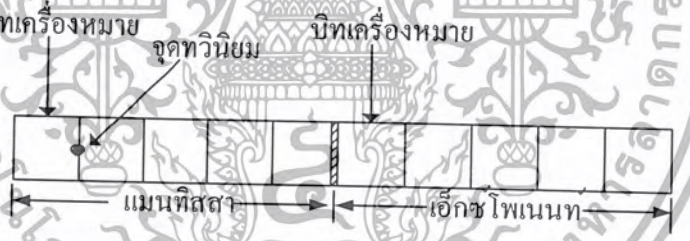
2. รูปแบบจำนวนอิงดรรชนี

รูปแบบจำนวนโดยตรงมีข้อเสียที่สำคัญ 2 ประการ คือ 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

$$X = M \times 2^{e} \tag{2.34}$$

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

M และ e เรียกว่า แมนทิสสา (Mantissa) และเอ็กซ์โพเนนท์ (Exponent) ตามลำดับ ตัวอย่างเช่น จำนวน 0.00110101 และ 01001.11 สามารถแทนได้โดย 0.110101×2^{-2} และ 0.100111×2^{4} ตามลำดับ ส่วนจำนวนที่มีค่าเป็นลบก็ทำในลักษณะเดียวกัน รูปแบบจำนวนอิงดรรชนีสามารถแสดงได้ดังรูปที่ 2.11 โดยแบ่งเป็น 2 ส่วน คือส่วนหนึ่งสำหรับแมนทิสสา และอีกส่วนสำหรับเอ็กซ์โพเนนท์



รูปที่ 2.11 แสดงการจัดรูปแบบจำนวนอิงดรรชนี

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 ทฤษฎีเลขคณิตกระจาย

จากที่ได้กล่าวมาแล้วว่า โครงสร้างเลขคณิตกระจายมีพื้นฐานอยู่บนการคูณแบบเลขส่วนเติมเต็มสอง ดังนั้นในหัวข้อนี้จะได้อธิบายถึงหลักการของการคูณเลขส่วนเติมเต็มสอง

ให้เลขส่วนเติมเต็มสองของ X ซึ่งแทนด้วย $\overline{X}$ และนิยามโดย

$$\overline{X} = \begin{cases} X & ; \ X \geq 0 \\ 2 - |X| & ; \ X < 0 \end{cases} \tag{2.35}$$

โดย X เป็นเลขที่เป็นเศษส่วน (Fractional number)

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

$$\overline{X} = X_0 \cdot X_1 \cdot X_2 \cdot \ldots \cdot X_L \tag{2.36}$$

ค่าของ $\overline{X}$ ในรูปของเลขฐานสิบสามารถหาได้ดังนี้

$$\overline{X} = -X_0 + \sum_{i=1}^{L} X_i 2^{-i} \tag{2.37}$$

จากนั้นพิจารณาผลคูณต่อไปนี้

$$Y = Xm \tag{2.38}$$

ให้ $\overline{Y}, \overline{X}$ และ $\overline{m}$ เป็นเลขส่วนเติมเต็มสองของ Y, X และ m ตามลำดับ จากนั้นพิจารณาจากสมการที่ 2.37 และ สมการที่ 2.38 จะได้

$$\begin{aligned} \overline{Y} &= -Y_0 + \sum_{i=1}^{L} Y_i 2^{-i} \\ &= -X_0 m + \sum_{i=1}^{L} X_i m 2^{-i} \end{aligned} \tag{2.39}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น

$$\overline{Y} = \text{ส่วนเติมเต็มสองของ}\left(-X_0 + 2^{-1}x_1 m + 2^{-2}X_2 m + 2^{-3}X_3 m + \ldots + 2^{-L}X_L m\right)$$
$$= \text{ส่วนเติมเต็มสองของ}\left(-X_0 m + 2^{-1}\left(X_1 m + \ldots + 2^{-1}\left(X_{L-1} m + 2^{-1}\left(X_L m\right)\right)\right)\right) \quad (2.40)$$

ต่อไปพิจารณาส่วนเติมเต็มสองของ $2^{-1}U$ โดย

$$\overline{U} = U_0 \cdot U_1 \cdot U_2 \cdot \ldots \cdot U_M$$

สำหรับ $U \geq 0$ (หรือ $U_0 = 0$) จะได้ส่วนเติมเต็มสองของ $\left(2^{-1}U\right) = 2^{-1}\overline{U}$

สำหรับ $U < 0$ (หรือ $U_0 = 1$)

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

ดังนั้นสรุปได้ว่า

$$\text{ส่วนเติมเต็มสองของ } \left(2^{-1}U\right) = \begin{cases} 2^{-1}\overline{U} & ;\ U_0 = 0 \\ 1 + 2^{-1}\overline{U} & ;\ U0 = 1 \end{cases} \quad (2.41)$$

สมการที่ 2.41 นี้ แสดงให้เห็นได้ว่า ส่วนเติมเต็มสองของ ($2^{-1}U$) เป็นการเลื่อนข้อมูลของ $\overline{U}$ ไปทางขวา 1 บิท

$$\therefore \text{ส่วนเติมเต็มสองของ } \left(2^{-1}U\right) = 2^{-1}\overline{U} \quad (2.42)$$

โดย $2^{-1}U$ 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) ทุกครั้งที่มีการเลื่อนบิทข้อมูล

จากนั้นพิจารณาสมการที่ 2.40 และสมการที่ 2.41 จะได้ว่า

$$\overline{Y} = \left(-X_0\overline{m} + 2^{-1}x_1\overline{m} + 2^{-2}X_2\overline{m} + 2^{-3}X_3\overline{m} + \ldots + 2^{-L}X_L\overline{m}\right)$$
$$= \left(-X_0\overline{m} + 2^{-1}\left(X_1\overline{m} + \ldots + 2^{-1}\left(X_{L-1}\overline{m} + 2^{-1}\left(X_L\overline{m}\right)\right)\right)\right) \quad (2.43)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจากสมการที่ 2.43 จะเห็นได้ว่าผลคูณจากสมการที่ 2.38 สามารถหาได้โดยการใช้หลักการเลื่อนและบวก (Shift and Add) โดยผลลัพธ์ที่ได้จากการคูณแบบเลขส่วนเติมเต็มสอง สามารถหาได้ตามขั้นตอนดังนี้

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

ตัวอย่างการทำงานตามอัลกอริทึมนี้

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

$m = -0.390625$

$$\overline{m} = 2 - |m|$$
$$= 2 - 0.390625$$
$$= 1.609375$$
$$\therefore \overline{m} = 1.100111$$

$X = 0.8125 = \overline{X}$

$\therefore \overline{X} = 0.1101 = X_0 \cdot X_1 \cdot X_2 \cdot X_3 \cdot X_4$

โดยมีขั้นตอนการทำงาน ดังตารางต่อไปนี้

ตารางที่ 2.4 แสดงขั้นตอนการคูณเลขส่วนเติมเต็มสอง

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

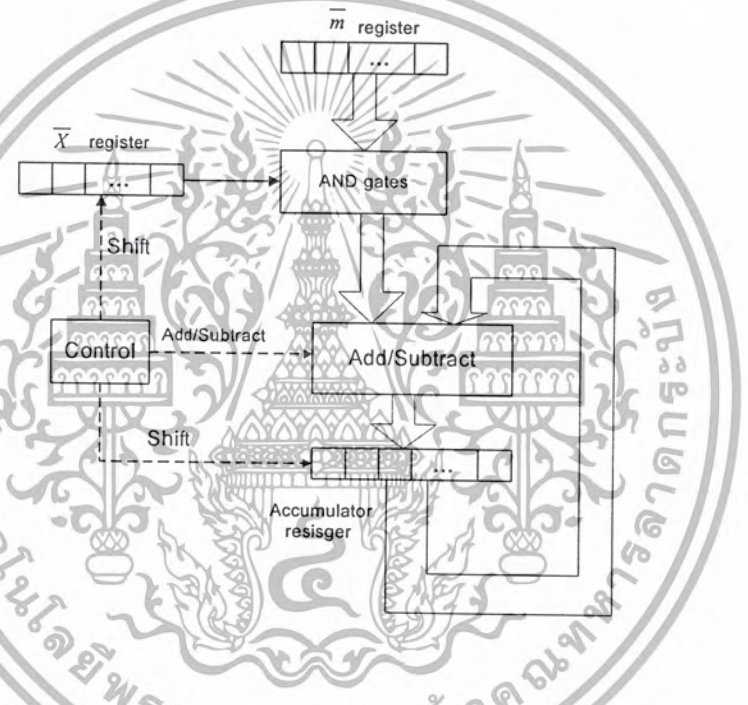
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\therefore \overline{Y} = 1.101\ 0111\ 0110\ = Y_0 \cdot Y_1 \cdot Y_2 \cdot \ldots \cdot Y_{11}$$

จะได้

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

จากอัลกอริทึมดังกล่าวสามารถออกแบบการทำงานและสร้างวงจรแสดงได้ดังรูปที่ 2.12



รูปที่ 2.12 แสดงการคูณแบบเลขส่วนเติมเต็มสองโดยใช้เลขคณิตกระจาย

ที่ผ่านมาเป็นหลักการคูณแบบเลขส่วนเติมเต็มสอง ส่วนทฤษฎีเลขคณิตกระจายก็อาศัยหลักการดังกล่าวมาใช้ โดยทำการกระจายสมการที่อยู่ในรูปผลบวกของผลคูณให้แตกออกมาอยู่ในระดับบิท (Bit level) พิจารณาผลบวกของผลคูณต่อไปนี้

$$Y = \sum_{i=0}^{N} X_i m_i \qquad (2.44)$$

โดย m_i เป็นค่าสัมประสิทธิ์ซึ่งมีค่าคงที่

X_i เป็นข้อมูลอินพุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

$$X_i = -X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \tag{2.45}$$

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

X_{i0} = บิทแสดงเครื่องหมาย

X_{iL} = บิทที่มีนัยสำคัญต่ำสุด (LSB)

$L+1$ = จำนวนบิทที่แทนข้อมูลอินพุท

แทนค่า X_i ในสมการที่ 2.45 ลงในสมการที่ 2.44 จะได้

$$Y = \sum_{i=0}^{N} m_i \left[-X_{i0} + \sum_{j=1}^{L} X_{ij} 2^{-j} \right] \tag{2.46}$$

เมื่อจัดเทอมของผลบวกใหม่จะได้

$$\begin{aligned} Y &= -X_{i0} \sum_{i=0}^{N} m_i + \sum_{j=1}^{L} X_{ij} 2^{-j} \sum_{i=0}^{N} m_i \\ &= -\sum_{i=0}^{N} X_{i0} m_i + \sum_{i=0}^{N} \sum_{j=1}^{L} X_{ij} m_i 2^{-j} \end{aligned} \tag{2.47}$$

จากนั้นทำการกระจายออกให้เป็นระดับบิท ได้ดังนี้

$$\begin{aligned} Y = &-(X_{00}m_0 + X_{10}m_1 + X_{20}m_2 + \ldots + X_{N0}m_N) \\ &+2^{-1}(X_{01}m_0 + X_{11}m_1 + X_{21}m_2 + \ldots + X_{N1}m_N) \\ &+2^{-2}(X_{02}m_0 + X_{12}m_1 + X_{22}m_2 + \ldots + X_{N2}m_N) \\ &+\ldots + 2^{-L}(X_{0L}m_0 + X_{1L}m_1 + X_{2L}m_2 + \ldots + X_{NL}m_N) \end{aligned} \tag{2.48}$$

สมการที่ 2.48 นี้ถูกกระจายออกให้อยู่ในรูปผลบวกของผลคูณระหว่างสัมประสิทธิ์กับข้อมูลอินพุทในระดับบิท ซึ่งเป็นนิยามของการคำนวณแบบเลขคณิตกระจาย และเมื่อเปรียบเทียบสมการที่ 2.48 กับสมการที่ 2.43 จะเห็นว่าการคำนวณหาค่า Y ก็ใช้เลขคณิตกระจายนั่นเอง เพียงแต่นำค่าผลคูณย่อย (Partial product) ที่คำนวณไว้ล่วงหน้าแล้วสำหรับแต่ละค่าที่สอดคล้องกับแต่ละบิทของข้อมูลอินพุทไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เก็บไว้ในตารางเปิดดู ซึ่งเป็นหน่วยความจำ EPROM และใช้ข้อมูลอินพุทเป็นแอดเดรสของหน่วยความจำ เพื่อนำค่าในตารางเปิดดูมาผ่านขั้นตอนการคำนวณตามบูทอัลกอริธึม ซึ่งค่าในตารางเปิดดู สามารถแสดงได้ดังนี้

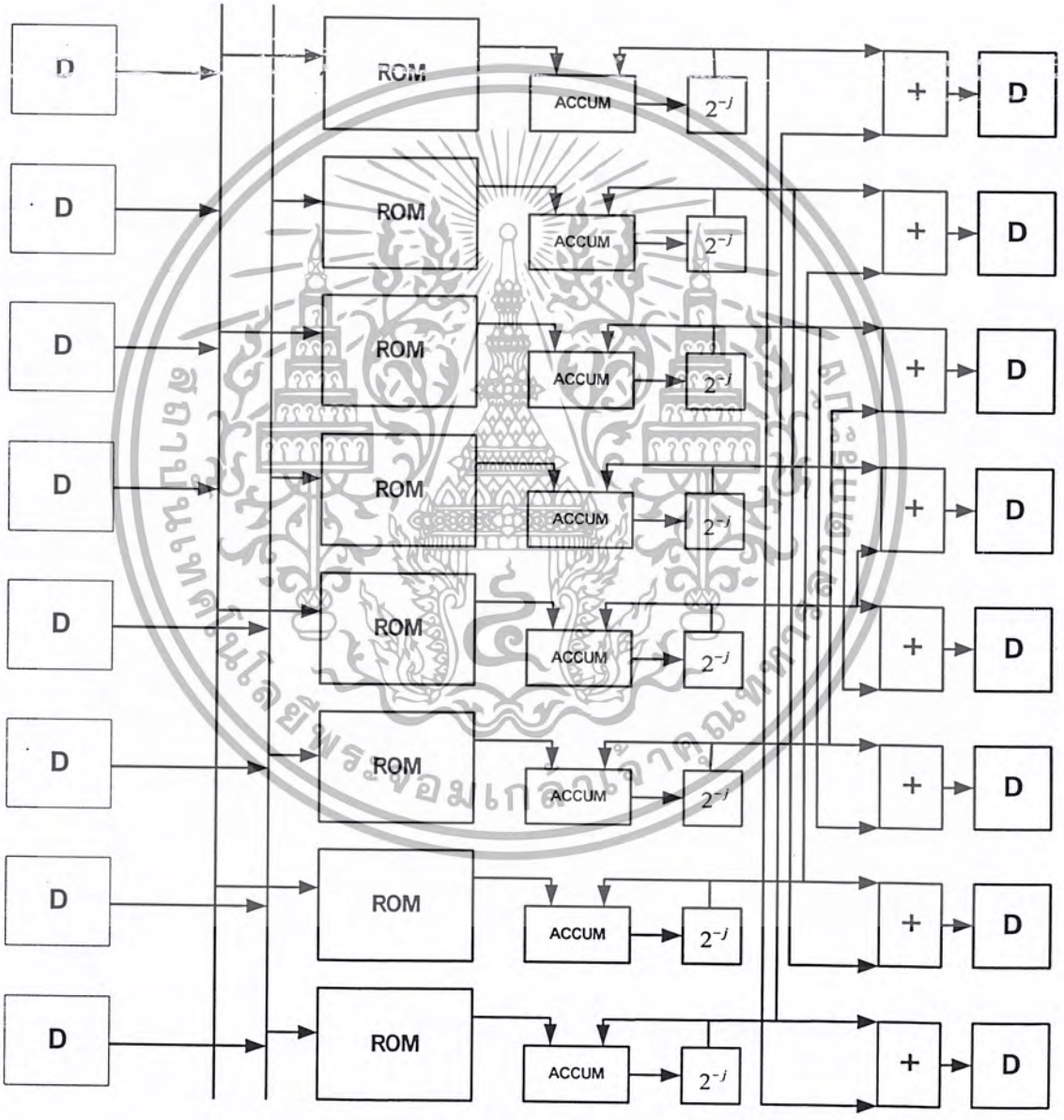
ตารางที่ 2.5 แสดงค่าผลคูณย่อยที่เก็บไว้ในตารางเปิดดูที่กำหนดโดยข้อมูลอินพุท

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 การประยุกต์ใช้งาน DA (Distributed Arithmetic) กับ CORDIC Algorithm

เนื่องจากที่การแปลงแบบ DCT เป็นเทคนิคที่นิยมใช้กันมากในปัจจุบันทั้งยังถูกนำมาใช้เป็นมาตรฐานในการบีบอัดสัญญาณภาพและวีดีโอแบบดิจิตอลอย่างหลากหลาย ในโครงสร้างหลายๆแบบของ DCT โครงสร้างอีกแบบหนึ่งที่นิยมใช้กันก็คือการใช้งานบนโครงสร้างของ DA ซึ่งคุณสมบัติของ DA นี้จะทำงานอยู่ในรูปแบบของผลบวกของผลคูณ (Sum of Products) โดยไม่ใช้การคูณโดยตรงในการประมวลผลของข้อมูล ซึ่งแนวความคิดนี้เหมาะสมกับรูปแบบการคำนวณของ DCT เช่นดังตัวอย่างของการนำโครงสร้าง DA ไปใช้กับ IDCT แบบหนึ่งมิติดังรูปที่ 2.13



รูปที่ 2.13 แสดงโครงสร้าง IDCT แบบหนึ่งมิติที่นำ DA ไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.13 แสดงในข้างต้นจะมี 3 องค์ประกอบที่จะเป็นตัวจำกัดความเร็วในการประมวลผลของ IDCT ในกรณีที่ใช้โครงสร้างของ DA คือเวลาของการเข้าถึงค่าใน ROM ในกรณีที่ ROM มีขนาดใหญ่ความเร็วของ Accumulator และความเร็วของตัว Adder นอกจากนี้เนื่องจาก DA จำเป็นที่จะต้องคำนวณค่าในรูปแบบของบิทอนุกรมและต้องทำการตรวจสอบค่าของบิทเครื่องหมาย (Sign Bit) ด้วย ดังนั้น Accumulator จะได้รับค่าอินพุทไปคำนวณหลังจากที่ชุดของ DA ทำงานเสร็จซึ่งจะเป็นเวลา w ของรอบสัญญาณนาฬิกา เมื่อ w คือความกว้างของจำนวนบิทของค่าอินพุตในแต่ละ sample

ในปริญญานิพนธ์นี้ได้นำเอาการทำงานของ DA มาประยุกต์ใช้กับการคำนวณโดยทฤษฎีของ CORDIC Algorithm ซึ่งจะถูกนำไปพิจารณาร่วมกับสมการพื้นฐานของ CORDIC และกำหนดค่าของ ROM ที่ใช้เป็นขนาด 4 word ซึ่งค่าที่กำหนดนี้จะไม่มีความเกี่ยวข้องกับขนาดความกว้างของค่าอินพุทที่ป้อนเข้ามา

ถึงแม้ว่าการนำไปสร้างโดยใช้ CORDIC Algorithm โดยตรงในการคำนวณค่าของ DCT จะดีกว่าเมื่อสร้างแบบใช้ตัวคูณแต่การนำไปสร้างจริงในรูปแบบโครงสร้างของการคำนวณแล้วก็ยังไม่เหนือไปกว่าการคำนวณโดยใช้ DA ดังนั้นหลักสำคัญก็คือการนำรูปแบบโครงสร้างของ DA มาใช้กับการคำนวณแบบหมุนของ CORDIC Algorithm ซึ่งจะทำให้ได้รูปแบบการคำนวณผลของ CORDIC Algorithm ในโครงสร้างแบบใหม่ขึ้นมาซึ่งสามารถที่จะพิจารณาได้ดังนี้

จากสมการพื้นฐานของ CORDIC Algorithm คือ

$$\begin{aligned} x &= x_i \cos\theta_i - y_i \sin\theta_i \\ y &= y_i \cos\theta_i + x_i \sin\theta_i \end{aligned} \tag{2.49}$$

โดยที่ x และ y เป็นตำแหน่งที่แทนค่าของแต่ละพิกเซลเมื่อแต่ละพิกเซลถูกแทนด้วยจำนวน B บิท ค่าเวคเตอร์ของแต่ละตำแหน่งจะสามารถแสดงเป็นสมการได้ดังนี้

$$\begin{aligned} x &= -x(0) + \sum_{j=1}^{B-1} x(j)2^{-j} \\ y &= -y(0) + \sum_{j=1}^{B-1} y(j)2^{-j} \end{aligned} \tag{2.50}$$

เมื่อ $x(j)$ และ $y(j)$ แทนค่าที่ตำแหน่งบิทต่างๆของ x และ y ทำการนำสมการที่ 2.50 ไปแทนในสมการที่ 2.49 และทำการจัดรูปใหม่ดังนี้

$$x = \left(-x(0) + \sum_{j=1}^{B-1} x(j)2^{-j} \right)\cos\theta - \left(-y(0) + \sum_{j=1}^{B-1} y(j)2^{-j} \right)\sin\theta$$

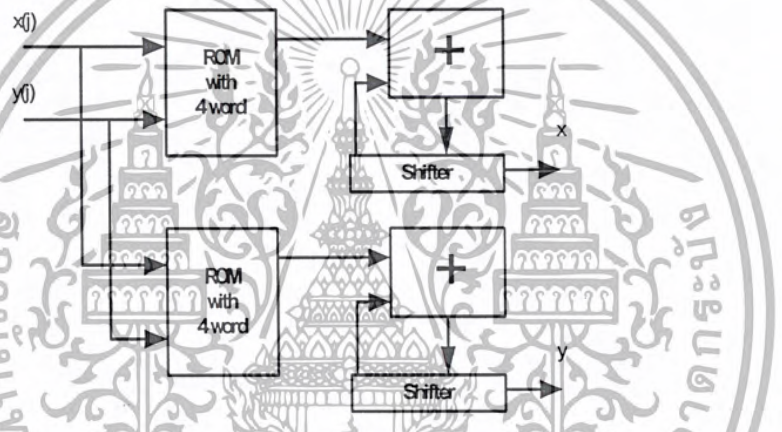
$$y = \left(-y(0) + \sum_{j=1}^{B-1} y(j)2^{-j} \right)\cos\theta + \left(-y(0) + \sum_{j=1}^{B-1} x(j)2^{-j} \right)\sin\theta$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขียนใหม่ได้เป็น

$$x = -(x(0)\cos\theta - y(0)\sin\theta) + \sum_{j=1}^{B-1}(x(j)\cos\theta - y(j)\sin\theta).2^{-j}$$
$$y = -(y(0)\cos\theta + x(0)\sin\theta) + \sum_{j=1}^{B-1}(y(j)\cos\theta + x(j)\sin\theta).2^{-j} \quad (2.51)$$

โดยที่ค่าของ $x(j)$ และ $y(j)$ จะเป็นค่าของ 0 หรือ 1 จากสมการที่ 2.51 จะทำให้มีผลของการหาค่าได้ 4 รูปแบบ ซึ่งค่าเหล่านี้เราสามารถที่จะทำการคำนวณและเก็บไว้ใน Look- up Table (หรือ ROM) และสมการที่ 2.51 ซึ่งจะแสดงให้เห็นว่าสามารถสร้างได้จากการคำนวณแบบผลบวกของผลคูณนั่นก็คือการนำเอา DA มาใช้ได้นั่นเองซึ่งโครงสร้างจะเป็นดังรูปที่ 2.14



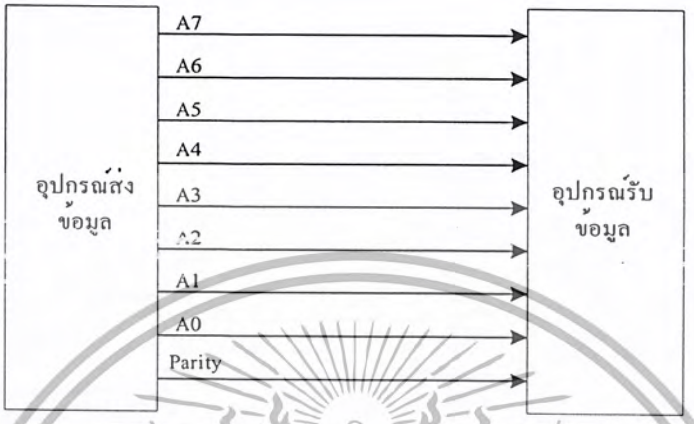
รูปที่ 2.14 แสดงโครงสร้างที่นำ DA ไปใช้กับ CORDIC Algorithm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การสื่อสารข้อมูล

2.7.1 การสื่อสารข้อมูลแบบขนาน (Parallel Communication)

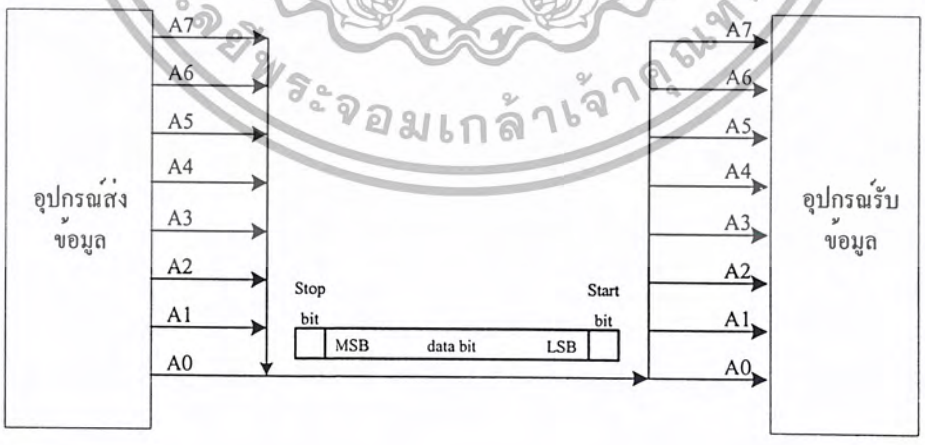
การสื่อสารข้อมูลแบบขนาน คือ การสื่อสารแบบที่ส่งข้อมูลพร้อม ๆ กัน n บิท ผ่านสายสัญญาณ n เส้น สามารถแสดงรูปแบบการสื่อสารข้อมูลแบบขนานได้ดังรูปที่ 2.15



รูปที่ 2.15 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารข้อมูลแบบขนาน

2.7.2 การสื่อสารข้อมูลแบบอนุกรม (Serial Communication)

การสื่อสารข้อมูลแบบอนุกรม คือ การสื่อสารแบบที่ส่งข้อมูลที่ละบิท ผ่านสายสัญญาณเส้นเดียวจนครบจำนวนข้อมูลที่ต้องการ โดยเฟรมของการสื่อสารข้อมูลแบบอนุกรมประกอบด้วย สตาร์ทบิท(Start bit) สตอบบิท (Stop bit) ข้อมูล (Data bit) สามารถแสดงรูปแบบการสื่อสารข้อมูลแบบอนุกรมได้ดังรูปที่ 2.16



รูปที่ 2.16 แสดงบล็อกไดอะแกรมรูปแบบการสื่อสารแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

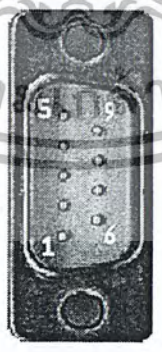
2.7.2.1 การอินเตอร์เฟสตามมาตรฐาน RS-232

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

มาตรฐาน RS-232 จะใช้สัญญาณเส้นเดียวในการส่งสัญญาณ โดยสัญญาณจะส่งไปในทิศทางเดียวกัน 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 2.17 และการเชื่อมต่อของพอร์ตสื่อสารสำหรับคอนเน็คเตอร์แบบ DB-9 สามารถแสดงได้ดังตารางที่ 2.6



รูปที่ 2.17 แสดงลักษณะของคอนเน็กเตอร์แบบ DB-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 แสดงขาสัญญาณของพอร์ตสื่อสารอนุกรมแบบ DB-9

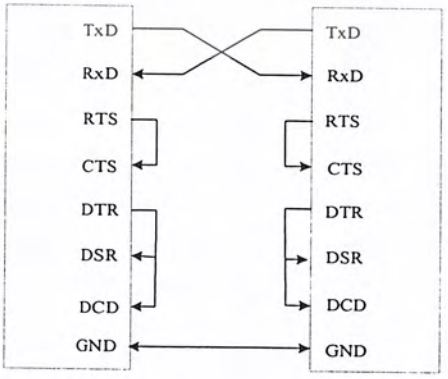
ตำแหน่งขาของ DB-9	สัญญาณ
1	Data Carrier Detect : DCD
2	Received Data : RxD
3	Transmitted Data : TxD
4	Data Terminal Ready : DTR
5	Signal Ground : GND
6	Data Set Ready : DSR
7	Request To Send : RST
8	Clear To Send : CTS
9	Ring Indicator : RI

เปรียบเทียบข้อดีข้อเสียของการสื่อสารข้อมูลแบบอนุกรมและแบบขนาน

- การสื่อสารข้อมูลแบบอนุกรมสามารถสื่อสารได้ระยะทางที่ไกลกว่า
- การสื่อสารข้อมูลแบบอนุกรมใช้สายสัญญาณที่ประหยัดกว่า
- การสื่อสารข้อมูลแบบขนานสามารถสื่อสารข้อมูลได้ทีละมากๆ และรวดเร็วกว่า

2.7.2.2 นันโมเด็ม (Null Modems)

นันโมเด็ม(Null Modems) เป็นการเชื่อมต่อระหว่าง DTE (Data Terminal Equipment) เข้าด้วยกัน ซึ่งสามารถแสดงการเชื่อมต่อแบบนันโมเด็มได้ดังรูปที่ 2.18

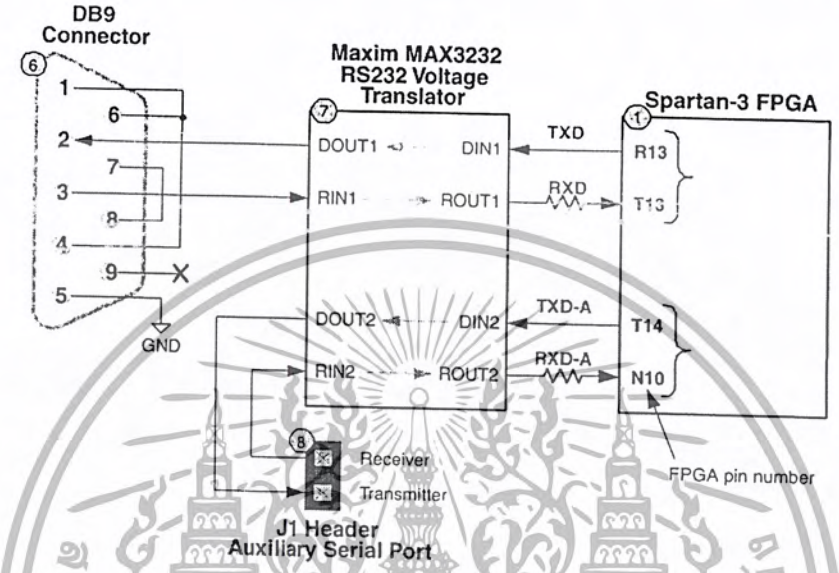


รูปที่ 2.18 แสดงบล็อกไดอะแกรมการเชื่อมต่อแบบนันโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2.3 การเชื่อมต่อระหว่าง DB-9 กับ FPGA

ปริญญานิพนธ์ฉบับนี้ได้ใช้บอร์ด FPGA สปาร์ตัน 3 สตาร์เตอร์ คิท (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 2.19



รูปที่ 2.19 แสดงการเชื่อมต่อระหว่างคอนเน็คเตอร์แบบ DB-9 และ บอร์ด FPGA โดยผ่าน MAX3232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 ภาษาวีเอชดีแอล

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

2.8.1 ประวัติความเป็นมาของภาษาวีเอชดีแอล

วีเอชดีแอล ย่อมาจากคำว่า VHSIC Hardware Description Language (VHSIC: Very High Speed Integrated Circuit) เป็นภาษาโปรแกรมระดับสูง (High Level Language) ที่ใช้สำหรับการออกแบบฮาร์ดแวร์ในระบบดิจิตอล ตัวของภาษาสามารถบรรยายพฤติกรรมการทำงานในรูปของลำดับชั้น (Hierarchy) และสามารถเขียนได้หลายรูปแบบด้วยเหตุผลนี้จึงทำให้ภาษาวีเอชดีแอล เป็นเครื่องมือที่ใช้ออกแบบตั้งแต่ขั้นตอนบนสุด คือ แนวความคิดที่จะแก้ปัญหา ลงไปทีละขั้นจนถึงขั้นตอนของการสร้างวงจรจริง และตัวภาษาก็เปิดโอกาสให้วิศวกรได้พัฒนาและจำลองการทำงานของรูปแบบฟังก์ชันการทำงานของวงจรอย่างสังเขป โดยไม่คำนึงถึงรายละเอียดเกี่ยวกับโครงสร้างวงจรจริงนอกจากนี้ วีเอชดีแอล ยังเป็นภาษาที่สนับสนุนลักษณะต่างๆ ของระบบดิจิตอลที่มีความซับซ้อนได้ทั้งหมด ดังนั้น วีเอชดีแอล จึงเป็นภาษาที่น่าสนใจในการศึกษาและนำไปใช้งานเป็นอย่างยิ่ง สำหรับมาตรฐานของภาษาที่ใช้บรรยายพฤติกรรมวงจร หรือฮาร์ดแวร์ สามารถสรุปได้ดังนี้

- ต้องเป็นภาษาที่นำไปเขียนรูปแบบระบบดิจิตอล และมีคุณสมบัติที่สามารถเข้าใจได้ทั้ง มนุษย์และเครื่องคอมพิวเตอร์โดยไม่ต้องมีการแปลหรือเปลี่ยนแปลงอีก
- สามารถนำไปใช้เป็นเอกสารประกอบโครงการได้
- ต้องเป็นภาษาที่เขียนขึ้นสำหรับใช้จำลองการทำงานของวงจร

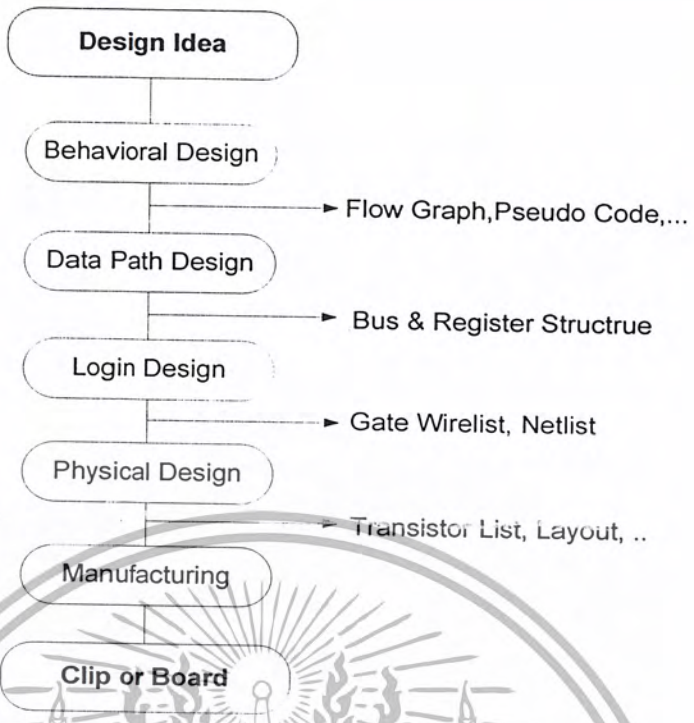
ฉะนั้นภาษาดังกล่าวนี้จึงจัดเป็นภาษาโปรแกรมระดับสูง เช่นเดียวกับภาษาปาสคาล หรือภาษาซี ซึ่งในทางวิศวกรรมภาษาที่ใช้ในการออกแบบฮาร์ดแวร์นี้เรียกว่า ภาษาโปรแกรมระดับสูง

2.8.2 การออกแบบระบบดิจิตอล

ในการออกแบบระบบดิจิตอล เริ่มตั้งแต่การกำเนิดแนวความคิดเบื้องต้นจนกระทั่งได้ออกมาเป็นอุปกรณ์ฮาร์ดแวร์ที่ใช้งานได้จะต้องผ่านขั้นตอนต่าง ๆ มากมาย และในแต่ละขั้นตอนผู้ออกแบบจะต้องตรวจสอบผลลัพธ์ในแต่ละขั้นก่อนเข้าสู่กระบวนการออกแบบในขั้นต่อไป

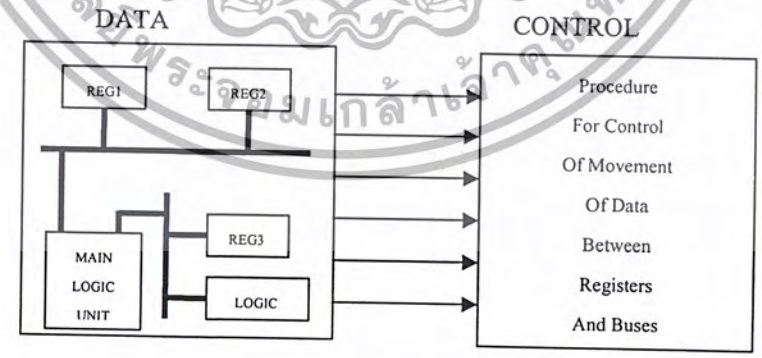
รูปที่ 2.20 แสดงขั้นตอนปกติที่ใช้ในการออกแบบระบบดิจิตอลทั่วไป ขั้นแรกผู้ออกแบบจะกำหนดแนวความคิดในการออกแบบ แล้วทำการพัฒนาให้สามารถนำไปใช้ได้อย่างสมบูรณ์ ซึ่งภายในขั้นตอนนี้ผู้ออกแบบจำเป็นต้องสร้างรูปแบบระบบเชิงพฤติกรรมขึ้นมาตรวจสอบซึ่งอาจจะเป็นผังงาน แสดงแบบหรือรหัสคำสั่งเทียม (Pseudo code) ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 แสดงขั้นตอนการออกแบบระบบดิจิตอล

ขั้นตอนต่อไปเป็นการออกแบบระบบเส้นทางของข้อมูล ผู้ออกแบบจะกำหนดส่วนประกอบของรีจิสเตอร์และวงจรลอจิกที่จำเป็นทั้งหมด เพื่อนำมาประกอบเป็นระบบที่สมบูรณ์ โดยแต่ละองค์ประกอบสามารถเชื่อมต่อกันด้วยบัสหนึ่งหรือสองทิศทาง (Unidirectional or Bidirectional Bus) กระบวนการในการควบคุมการเคลื่อนย้ายข้อมูลระหว่างรีจิสเตอร์และวงจรลอจิกจะขึ้นอยู่กับพฤติกรรมของระบบที่กำหนดไว้ดังรูปที่ 2.21



รูปที่ 2.21 การออกแบบระบบเส้นทางข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

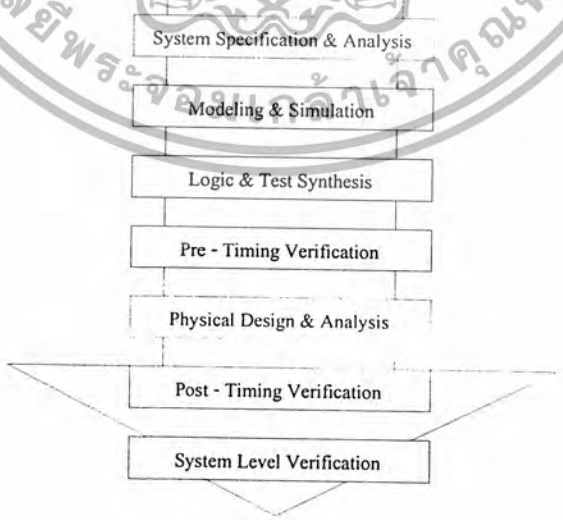
ขั้นตอนถัดมาเป็นการออกแบบวงจรลอจิกซึ่งจะเกี่ยวข้องกับการนำเกทดิจิตอลพื้นฐาน และฟลิปฟลอป (Flip – Flop) มาประกอบเป็นอุปกรณ์ย่อยต่าง ๆ เช่นรีจิสเตอร์เก็บข้อมูล บัสวงจรลอจิก และส่วนควบคุมฮาร์ดแวร์ ซึ่งผลลัพธ์ที่ได้ในขั้นตอนนี้จะเป็นเครือข่ายของการโยงใยระหว่างเกทและฟลิปฟลอปนั่นเอง

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

และในขั้นตอนสุดท้ายจะเป็นการส่งระบบที่ออกแบบไว้ไปทำการเจือที่โรงงานเพื่อผลิตออกมาเป็นวงจรรวมในที่สุด

2.8.3 การออกแบบจากบนลงล่าง (Top-Down Design)

ในการพัฒนาวงจรรวมดิจิตอลขนาดใหญ่ที่มีความซับซ้อน วิศวกรหรือผู้ออกแบบมักจะมองการออกแบบให้อยู่ในรูปของ บล็อกไดอะแกรมก่อนที่ทำวิเคราะห์ให้ลึกถึงรายละเอียดต่อไป ซึ่งภาษาวีเอชดีแอลนั้นอนุญาตให้อธิบายและวิเคราะห์การทำงานของแต่ละบล็อก รวมถึงการปรับปรุงการทำงานจากผลที่วิเคราะห์เพื่อให้ได้การทำงานตามต้องการนอกจากนี้ยัง สามารถเพิ่มเติมในรายละเอียดในแต่ละขั้นตอนได้ ซึ่งหลักการนี้สอดคล้องกับหลักการออกแบบจากบนลงล่างนั่นเอง ถ้าทดลองเปรียบเทียบกับการออกแบบจากล่างขึ้นบน (Bottom-Up Design) จะเห็นได้ว่า การออกแบบจากล่างขึ้นบนจะใช้เวลาการออกแบบมากกว่า 90% เนื่องจากเป็นการวางวงจรด้วยอุปกรณ์ต่างๆ (Schematic Capture) ที่ประกอบกันเข้าเป็นวงจรที่ต้องการออกแบบก่อน แล้วจึงทำการจำลองการทำงาน และตรวจสอบความถูกต้อง วีเอชดีแอล กับหลักการออกแบบจากบนลงล่างจึงเป็นทางออกให้กับวิศวกรให้สามารถ ออกแบบและพัฒนาวงจรที่มีความซับซ้อนได้มากขึ้น ทั้งยังช่วยลดเวลาและค่าใช้จ่ายในการออกแบบด้วย



รูปที่ 2.22 แสดงขั้นตอนการออกแบบจากบนลงล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.22 แสดงถึงขั้นตอนการออกแบบจากบนลงล่าง ทั้งนี้ในทางปฏิบัติอาจมีข้อแตกต่างไปจากนี้บ้างเล็กน้อย เนื่องจากขั้นตอนของการผลิต (Implementation) สามารถกระทำได้หลายเทคโนโลยี สำหรับรายละเอียดของขั้นตอน การออกแบบจากบนลงล่างในแต่ละขั้นตอนมีดังนี้

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

2) รูปแบบและการจำลองการทำงาน คือ การเขียนรูปแบบของระบบที่ต้องการออกแบบโดยใช้ภาษา วีเอชดีแอล หรือ ภาษา เอชดีแอล อื่นๆ สำหรับใช้ในการบรรยายพฤติกรรมการทำงาน พร้อมทั้งทำการจำลองการทำงาน เพื่อเปรียบเทียบและตรวจสอบความถูกต้องกับข้อกำหนด

3) ลอจิกและการทดสอบการสังเคราะห์ คือ หลังจากที่ได้หลักการขั้นต้นพร้อมแนวความคิดที่ผ่านการตรวจสอบแล้วหลักการนี้จะถูกเพิ่มเติมรายละเอียดลงมาเป็นลำดับขั้นที่สอง จนกระทั่งอยู่ในระดับที่จะนำไปผลิตวงจรจริง หรือทำการสังเคราะห์ในขั้นตอนนี้เองเทคโนโลยีที่จะมารองรับวงจรออกแบบจะถูกกำหนดขึ้น และระบบช่วยการออกแบบจะทำการสังเคราะห์วงจรที่ได้จากรูปแบบที่จะเขียนขึ้นให้อยู่ในรูปของวงจรที่ประกอบด้วยอุปกรณ์อิเล็กทรอนิกส์ หรือวงจรในระดับเกท และการเชื่อมต่อระหว่างกันของอุปกรณ์เหล่านั้นหรือไม่ก็อยู่ในรูปของโครงข่ายการเชื่อมต่อ ที่สามารถนำไปผลิตอุปกรณ์อื่นได้

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

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

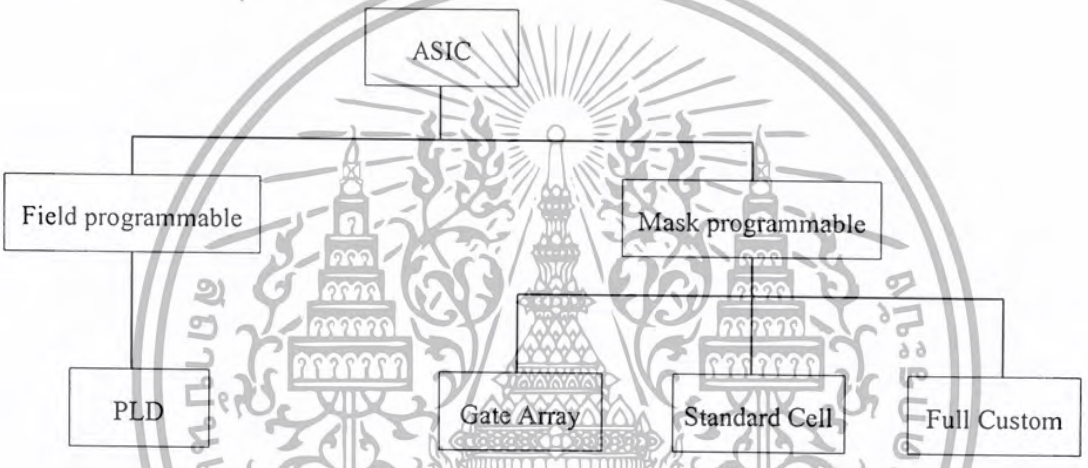
6) การตรวจสอบเวลาหลังการออกแบบ คือ การทำการตรวจสอบการทำงานด้วยตัวแปรทางด้านเวลาทั้งหมด เพื่อความถูกต้องของวงจรเป็นครั้งสุดท้ายก่อนนำไปรวมเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบดิจิตอล เนื่องจากในขั้นตอนนี้ วงจรที่ออกแบบ จะประกอบด้วยจุดต่อทางอินพุทและเอาท์พุท ซึ่งเป็นจุดต่อสำหรับการรับและส่งสัญญาณกับภายนอก

7) การตรวจสอบระบบ คือ การนำวงจรที่ออกแบบไว้ประกอบเข้ากับอุปกรณ์อื่นๆ ให้เป็นระบบที่สมบูรณ์ แล้วทำการทดสอบการทำงานทั้งระบบรวมกับอุปกรณ์อื่นๆ อีกครั้งเพื่อควบคุมคุณภาพของผลิตภัณฑ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 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 2.23



รูปที่ 2.23 แสดงผังการแบ่งกลุ่มของวงจรรวมเอซิก

2.9.1 การออกแบบวงจรเชิงเลขด้วยชิพอุปกรณ์เอฟพีจีเอ

ชิพอุปกรณ์เอฟพีจีเอ เป็นอุปกรณ์ที่ใช้ในการโปรแกรมวงจรที่ได้ออกแบบลงไปเพื่อให้อุปกรณ์ เอฟพีจีเอ มีฟังก์ชันการทำงานตามที่ออกแบบไว้ ในการทำชิพอุปกรณ์ ซึ่งเป็นวิธีการออกแบบไอซี (IC : Integrated Circuit) แบบ เซมิคัสตัม (Semi custom) อีกวิธีหนึ่ง เมื่อเทียบกับการทำเอซิกแล้วนั้นก็มีทั้งข้อดีและข้อเสีย คือ การทำชิพอุปกรณ์เอฟพีจีเอ จะมีข้อจำกัดในด้านขนาดของวงจรเพราะภายในชิพอุปกรณ์เอฟพีจีเอ จะมีจำนวนเกท (gate) ให้ใช้จำนวนจำกัด และการทำชิพอุปกรณ์เอฟพีจีเอ ก็เหมาะสำหรับการทำผลิตภัณฑ์ต้นแบบหรือเพื่อผลิตในปริมาณต่ำ ส่วนข้อดีของการทำชิพอุปกรณ์ก็คือระยะเวลาที่ใช้ในการทำตั้งแต่เขียนรหัส (code) อธิบายฮาร์ดแวร์จนกระทั่งดาวน์โหลด (download) นั้นน้อยกว่าการทำเอซิก มากและการตรวจสอบหรือแก้ไขการออกแบบที่ทำได้สะดวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำชิพอุปกรณ์เอฟพีจีเอ ในปัจจุบันมีประสิทธิภาพและความสะดวกมากขึ้น ทั้งนี้ก็เนื่องจากทางบริษัทผู้ผลิตชิพอุปกรณ์เอฟพีจีเอ ได้เพิ่มความสามารถของชิพอุปกรณ์เอฟพีจีเอ โดยเพิ่มจำนวนองค์ประกอบภายใน หรือปรับปรุงโครงสร้างสถาปัตยกรรมภายในและยังได้เพิ่มประสิทธิภาพของซอฟต์แวร์ที่ใช้ทำ พีพีอาร์ (PPR: Partitioning Placement and Routing) สำหรับอุปกรณ์นั้น ๆ ด้วย

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

2.9.2 ปัจจัยที่ทำให้การออกแบบชิพอุปกรณ์เอฟพีจีเอ ทำได้ง่ายและสะดวกรวดเร็ว

1. ผู้ออกแบบไม่จำเป็นต้องทราบถึงโครงสร้างภายในของตัวชิพเพียงแต่มีความรู้เกี่ยวกับขั้นตอนการออกแบบลอจิกก็เพียงพอแล้ว ต่างกับการใช้ไมโครโปรเซสเซอร์ซึ่งจำเป็นต้องศึกษาโครงสร้างภายในรวมถึงภาษาแอสแซมบลี้ (Assembly) ของไมโครโปรเซสเซอร์ตัวนั้นด้วย

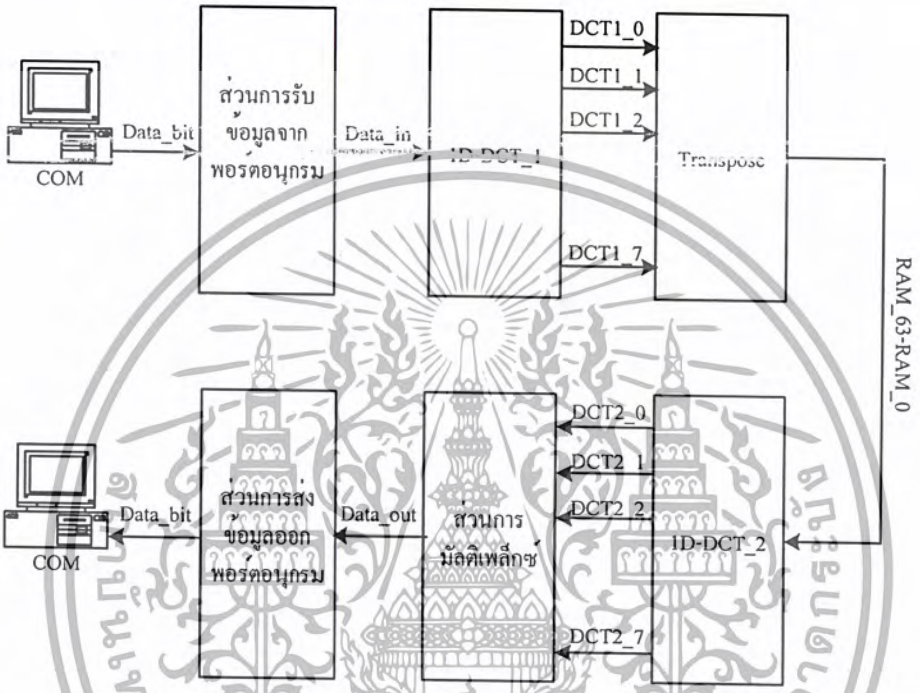
2. มีการออกแบบโดยใช้ภาษาในการอธิบายการทำงานของวงจรหรือเอชดีแอล เป็นเครื่องมือในการออกแบบ ซึ่งเป็นวิธีการที่มีความยืดหยุ่นสูง ทำได้รวดเร็ว และไม่จำเป็นต้องทราบถึงลักษณะของวงจรที่ต้องการว่าจะเชื่อมต่อกันอย่างไร เพียงแต่กำหนดลักษณะการทำงานให้มันจากนั้นตัวซอฟต์แวร์จะทำการสังเคราะห์ให้ทั้งหมด นอกจากนี้ภาษาที่ใช้ยังเป็นมาตรฐานเดียวกันสามารถใช้ได้กับชิพทุกตัวและทุกบริษัท

3. การโปรแกรมสามารถทำได้เองและใช้เวลาไม่นาน เพียงแค่ส่งข้อมูลผ่านสายดาวน์โหลดทางพอร์ตของคอมพิวเตอร์ก็สามารถโปรแกรมตัวชิพขณะที่อยู่ในระบบได้โดยไม่จำเป็นต้องถอดมาโปรแกรมข้างนอก และที่สำคัญสามารถโปรแกรมได้หลายครั้ง จึงทำให้ง่ายในการแก้ไขและพัฒนาโดยไม่ต้องเสียค่าใช้จ่ายเพิ่มแต่อย่างใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

ปริญญานิพนธ์นี้จะทำการลดขนาดของข้อมูลรูปภาพโดยใช้การแปลงดิสครีตโคซายน์แบบ 2 มิติ ด้วยวิธีการแยกเป็นการแปลงดิสครีตโคซายน์ 1 มิติ จำนวน 2 ครั้ง สามารถแสดงขั้นตอนของการลดขนาดข้อมูลภาพด้วยการแปลงดิสครีตโคซายน์เป็นบล็อกไดอะแกรมได้ดังรูปที่ 3.1



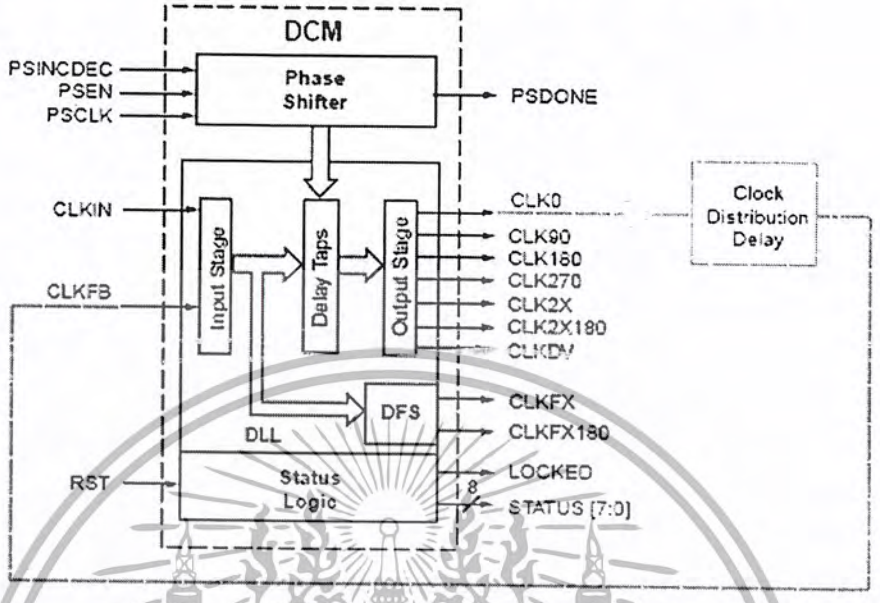
รูปที่ 3.1 แสดงการแปลงดิสครีตโคซายน์แบบ 2 มิติ โดยแยกเป็นการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง

3.1 ส่วนของการสร้างสัญญาณนาฬิกาของระบบและสัญญาณนาฬิกาของความถี่บอดเรต (Baud Rate)

สัญญาณนาฬิกาสามารถแบ่งได้เป็น 2 ส่วน คือ สัญญาณนาฬิกาของระบบ และสัญญาณนาฬิกาของความถี่บอดเรต (Baud Rate) สัญญาณนาฬิกาของระบบผลิตมาจากออสซิลเลเตอร์ (Oscillator) บนบอร์ด FPGA ความถี่ 25 MHz แต่สัญญาณนาฬิกาที่ใช้สำหรับการสื่อสารข้อมูลทางพอร์ตอนุกรมนั้นต้องใช้สัญญาณนาฬิกาตามความถี่บอดเรต เช่น 4800 9600 19200 แซมเปิลต่อวินาที เป็นต้น ซึ่งสามารถสร้างมาจากการนำสัญญาณนาฬิกาของระบบมาผ่านวงจรหารด้วยค่าที่เหมาะสม ดังนั้นเพื่อความสะดวกและความถูกต้องของความถี่บอดเรต จึงต้องหาสัญญาณนาฬิกาของระบบที่สามารถหารได้ลงตัวด้วยจำนวนเต็มบวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในบอร์ด FPGA อุปกรณ์ที่เรียกว่า Digital Clock Managers (DCMs) ทำหน้าที่เป็นตัวคูณและตัวหารสัญญาณนาฬิกาอินพุท หรือเลื่อนเฟส (shift phase) ของสัญญาณเพื่อให้ได้สัญญาณเอาท์พุทตัวใหม่ตามต้องการ สามารถแสดงบล็อกไดอะแกรมของฟังก์ชันต่างๆ ของ DCM ได้ดังรูปที่ 3.2

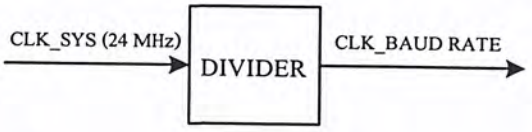


รูปที่ 3.2 แสดงบล็อกไดอะแกรมของฟังก์ชันต่างๆ ของ DCMs

ในที่นี้ใช้เฉพาะฟังก์ชันของตัวสังเคราะห์ความถี่ดิจิตอล (Digital Frequency Synthesizer : DFS) ที่ทำหน้าที่ หารและคูณความถี่สัญญาณอินพุทเพื่อให้ได้ความถี่ของสัญญาณเอาท์พุทตามต้องการ ในปริญญานิพนธ์นี้ต้องการความถี่สัญญาณอ้างอิง 24 MHz จึงต้องนำสัญญาณอินพุทจาก ออสซิลเลเตอร์ 25MHz มาผ่านวงจร DCM เพื่อให้ผลิตความถี่ตามต้องการ สามารถแสดงได้ดังรูปที่ 3.3

รูปที่ 3.3 แสดงบล็อกไดอะแกรมของฟังก์ชัน DFS ของ DCMs
ที่สร้างสัญญาณเอาท์พุทความถี่ 24 MHz จากสัญญาณอินพุทความถี่ 25 MHz

เมื่อได้ค่าสัญญาณนาฬิกาของระบบที่ต้องการทางเอาท์พุทของวงจร DCMs แล้วนำสัญญาณดังกล่าวไปผ่านวงจรหารความถี่เพื่อให้ได้สัญญาณนาฬิกาของความถี่บอดเรตที่ต้องการเพื่อใช้สำหรับการสื่อสารข้อมูลทางพอร์ตอนุกรม สามารถแสดงบล็อกไดอะแกรมวงจรหารความถี่บอดเรตได้ดังรูปที่ 3.4

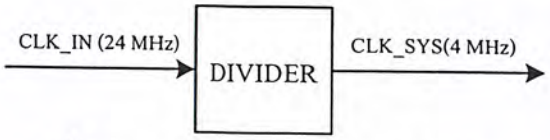


รูปที่ 3.4 แสดงบล็อกไดอะแกรมของวงจรหารความถี่บอดเรต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนของวงจรหารความถี่ระบบ

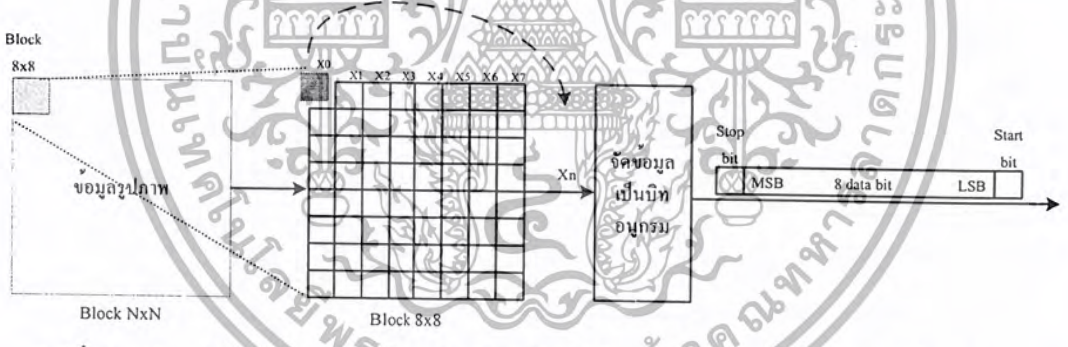
เนื่องจากสัญญาณความถี่ที่มาจาก DCMs มีค่าเท่ากับ 24 MHz เพื่อให้กระบวนการทำงานในระบบมีความถูกต้อง แม่นยำและไม่ผิดพลาดจึงต้องหารความถี่ให้ความถี่ของระบบมีค่าเท่ากับ 4 MHz สามารถแสดงบล็อกไดอะแกรมของวงจรหารความถี่ระบบได้ดังรูปที่ 3.5



รูปที่ 3.5 แสดงบล็อกไดอะแกรมของวงจรหารความถี่ระบบ

3.3 การแปลงบิทข้อมูลส่งออกพอร์ตอนุกรมไปยังบอร์ด FPGA ด้วยโปรแกรม MATLAB

เมื่อแบ่งข้อมูลรูปภาพขนาด $N \times N$ ออกเป็นบล็อกเล็ก ๆ ขนาด 8×8 แล้วนำค่าข้อมูลตามแถวแต่ละพิกเซลของบล็อก 8×8 มาทำการแปลงให้เป็นข้อมูลไบนารีแบบอนุกรม โดยกำหนดเฟรมข้อมูลที่ประกอบด้วย สตาร์ทบิท (Start bit) 1 บิท สตอบบิท (Stop bit) 2 บิท ข้อมูล (Data bit) 8 บิท แล้วส่งเฟรมข้อมูลดังกล่าวออกพอร์ตอนุกรม (Serial port) ไปยังบอร์ด FPGA ด้วยความถี่บอดเรต (Baud Rate) ที่กำหนดไว้ โดยทำการส่งข้อมูลดังกล่าวจนครบทั้ง 8×8 พิกเซล สามารถแสดงบล็อกไดอะแกรมการแปลงบิทข้อมูลส่งออกพอร์ตอนุกรมไปยังบอร์ด FPGA ได้ดังรูปที่ 3.6

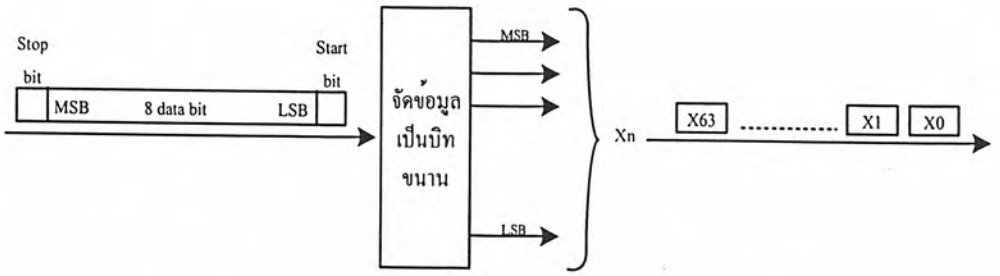


รูปที่ 3.6 แสดงบล็อกไดอะแกรมการแปลงบิทข้อมูลส่งออกพอร์ตอนุกรมไปยังบอร์ด FPGA

3.4 การรับข้อมูลจากพอร์ตอนุกรม

ส่วนของการรับข้อมูลจากพอร์ตอนุกรมนั้น ทำหน้าที่ รับเฟรมบิทของข้อมูลจากพอร์ตอนุกรมตามความถี่บอดเรต (Baud Rate) ที่กำหนดไว้ มาทำการตัดสตาร์ทบิท (Start bit) และสตอบบิท (Stop bit) แล้วทำการแปลงข้อมูลจากบิทอนุกรมเป็นบิทขนาน เมื่อรับข้อมูลครบ 8 ค่า ก็จะส่งต่อไปให้ส่วนของวงจรเก็บค่าข้อมูลอินพุทที่ทำงานตามสัญญาณนาฬิกาของระบบ สามารถแสดงบล็อกไดอะแกรมการรับข้อมูลจากพอร์ตอนุกรมแล้วแปลงข้อมูลเป็นบิทขนานได้ดังรูปที่ 3.7

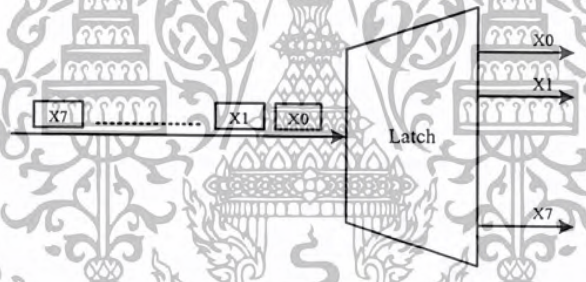
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แสดงบล็อกไดอะแกรมการเปลี่ยนข้อมูลอินพุทจาก
ข้อมูลแบบอนุกรมเป็นข้อมูลแบบขนาน

3.5 การออกแบบส่วนของวงจรคงค่าสัญญาณ ส่วนที่ 1 (Latch)

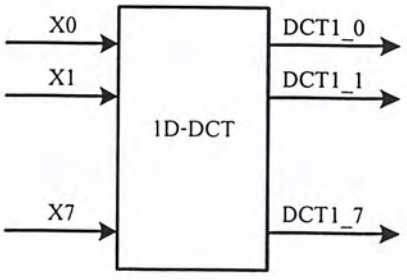
ข้อมูลตามแนวนอนที่จะนำไปทำการแปลงดิสครีตโคซายน์เป็นข้อมูลแบบอนุกรม ดังนั้นต้องทำการจัดรูปแบบข้อมูลใหม่ให้อยู่ในรูปขนาน โดยผ่านส่วนของวงจรคงค่าสัญญาณ(Latch) ที่ทำหน้าที่รอข้อมูลจนครบ 8 ค่า แล้วจึงปล่อยข้อมูลทั้ง 8 ค่า ออกไปพร้อมกันในลักษณะขนาน สามารถแสดงบล็อกไดอะแกรมของวงจรคงค่าสัญญาณได้ดังรูปที่ 3.8



รูปที่ 3.8 แสดงบล็อกไดอะแกรมของวงจรคงค่าสัญญาณ ส่วนที่ 1

3.6 การออกแบบส่วนการแปลงดิสครีตโคซายน์แบบ1มิติ (1D-DCT) ส่วนที่ 1

ส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ เป็นการนำข้อมูลอินพุท 8 ค่า มาผ่านการแปลงดิสครีตโคซายน์ได้เอาท์พุท 8 ค่า ตามหลักการของการแปลงดิสครีตโคซายน์สามารถแสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1 ได้ดังรูปที่ 3.9



รูปที่ 3.9 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.1 การคำนวณหาสมการและสัมประสิทธิ์ที่ใช้ในการสร้าง

สมการการแปลงดิสครีตโคซายน์แบบ 1 มิติ และค่าสัมประสิทธิ์ของการแปลงดิสครีตโค-ซายน์ สามารถแสดงได้ดังนี้

$$Y = CX \tag{3.1}$$

$$[C]_{i,j} = \begin{cases} 2\sqrt{\frac{1}{N}}\cos\frac{(2j+1)i\pi}{2N} & i = 0, j = 0,1,\ldots,N-1 \\ 2\sqrt{\frac{2}{N}}\cos\frac{(2j+1)i\pi}{2N} & i = 1,2,\ldots,N-1, j = 0,1,\ldots,N-1 \end{cases} \tag{3.2}$$

จากสมการที่ 3.1 และ 3.2 สามารถแสดงสมการให้อยู่ในรูปของเมตริกซ์ได้ดังนี้

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \end{bmatrix} = \begin{bmatrix} c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 \\ c_1 & c_3 & c_5 & c_7 & -c_7 & -c_5 & -c_3 & -c_1 \\ c_2 & c_6 & -c_6 & -c_2 & -c_2 & -c_6 & c_6 & c_2 \\ c_3 & -c_7 & -c_1 & -c_5 & c_5 & c_1 & c_7 & -c_3 \\ c_4 & -c_4 & -c_4 & c_4 & c_4 & -c_4 & -c_4 & c_4 \\ c_5 & -c_1 & c_7 & c_3 & -c_3 & -c_7 & c_1 & -c_5 \\ c_6 & -c_2 & c_2 & -c_6 & -c_6 & c_2 & -c_2 & c_6 \\ c_7 & -c_5 & c_3 & -c_1 & c_1 & -c_3 & c_5 & -c_7 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix} \tag{3.3}$$

จากสมการที่ 3.3 สามารถเขียนให้อยู่ในรูปของผลคูณย่อยแต่ละตัวได้ดังนี้

$$\begin{aligned} y(0) &= c_4x(0) + c_4x(1) + c_4x(2) + c_4x(3) + c_4x(4) + c_4x(5) + c_4x(6) + c_4x(7) \\ y(1) &= c_1x(0) + c_3x(1) + c_5x(2) + c_7x(3) - c_7x(4) - c_5x(5) - c_3x(6) - c_1x(7) \\ y(2) &= c_2x(0) + c_6x(1) - c_6x(2) - c_2x(3) - c_2x(4) - c_6x(5) + c_6x(6) + c_2x(7) \\ y(3) &= c_3x(0) - c_7x(1) - c_1x(2) - c_5x(3) + c_5x(4) + c_1x(5) + c_7x(6) - c_3x(7) \\ y(4) &= c_4x(0) - c_4x(1) - c_4x(2) + c_4x(3) + c_4x(4) - c_4x(5) - c_4x(6) + c_4x(7) \\ y(5) &= c_5x(0) - c_1x(1) + c_7x(2) + c_3x(3) - c_3x(4) - c_7x(5) + c_1x(6) - c_5x(7) \\ y(6) &= c_6x(0) - c_2x(1) + c_2x(2) - c_6x(3) - c_6x(4) + c_2x(5) - c_2x(6) + c_6x(7) \\ y(7) &= c_7x(0) - c_5x(1) + c_3x(2) - c_1x(3) + c_1x(4) - c_3x(5) + c_5x(6) - c_7x(7) \end{aligned} \tag{3.4}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการ 3.4 สามารถทำการจัดรูปใหม่ได้ดังนี้

$$
\begin{aligned}
y(0) &= c_4\left[(x(0)+x(7))+(x(3)+x(4))\right]+c_4\left[(x(1)+x(6))+(x(2)+x(5))\right]\\
y(4) &= c_4\left[(x(0)+x(7))+(x(3)+x(4))\right]-c_4\left[(x(1)+x(6))+(x(2)+x(5))\right]\\
y(2) &= c_2\left[(x(0)+x(7))-(x(3)+x(4))\right]+c_6\left[(x(1)+x(6))-(x(2)+x(5))\right]\\
y(6) &= c_6\left[(x(0)+x(7))-(x(3)+x(4))\right]-c_2\left[(x(1)+x(6))-(x(2)+x(5))\right]\\
y(1) &= \left[c_1(x(0)-x(7))+c_7(x(3)-x(4))\right]+\left[c_3(x(1)-x(6))+c_5(x(2)-x(5))\right]\\
y(7) &= \left[c_7(x(0)-x(7))-c_1(x(3)-x(4))\right]-\left[c_5(x(1)-x(6))-c_3(x(2)-x(5))\right]\\
y(3) &= \left[c_3(x(0)-x(7))-c_5(x(3)-x(4))\right]-\left[c_1(x(2)-x(5))+c_7(x(1)-x(6))\right]\\
y(5) &= \left[c_5(x(0)-x(7))+c_3(x(3)-x(4))\right]+\left[c_1(x(2)-x(5))-c_5(x(1)-x(6))\right]
\end{aligned}
\tag{3.5}
$$

โดยกำหนดให้

$$
\begin{aligned}
P_0 &= x(0)+x(7) & M_0 &= x(0)-x(7)\\
P_1 &= x(1)+x(6) & M_1 &= x(1)-x(6)\\
P_2 &= x(2)+x(5) & M_2 &= x(2)-x(5)\\
P_3 &= x(3)+x(4) & M_3 &= x(3)-x(4)
\end{aligned}
\quad ;
$$

จากสมการที่ 3.5 สามารถทำการจัดรูปใหม่ได้ดังนี้

$$
\begin{aligned}
y(0) &= c_4\left[P_0+P_3\right]+c_4\left[P_1+P_2\right]\\
y(4) &= c_4\left[P_0+P_3\right]-c_4\left[P_1+P_2\right]\\
y(2) &= c_2\left[P_0-P_3\right]+c_6\left[P_1-P_2\right]\\
y(6) &= c_6\left[P_0-P_3\right]-c_2\left[P_1-P_2\right]\\
y(1) &= \left[c_1M_0+c_7M_3\right]+\left[c_3M_1+c_5M_2\right]\\
y(7) &= \left[c_7M_0-c_1M_3\right]-\left[c_5M_1-c_3M_2\right]\\
y(3) &= \left[c_3M_0-c_5M_3\right]-\left[c_1M_2+c_7M_1\right]\\
y(5) &= \left[c_5M_0+c_3M_3\right]+\left[c_7M_2-c_1M_1\right]
\end{aligned}
\tag{3.6}
$$

จากสมการที่ 3.4 ต้องใช้ตัวคูณทั้งหมด 64 ตัว
ต้องใช้ตัวบวกทั้งหมด 56 ตัว
จากสมการที่ 3.6 ต้องใช้ตัวคูณทั้งหมด 24 ตัว
ต้องใช้ตัวบวกทั้งหมด 24 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

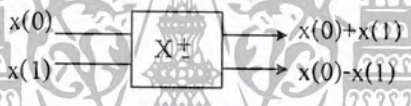
จะเห็นได้ว่าจากสมการที่ 3.4 มาเป็นสมการที่ 3.6 เป็นวิธีที่สามารถลดจำนวนตัวคูณและจำนวนตัวบวกได้ ซึ่งจะส่งผลดีต่อวงจรฮาร์ดแวร์ที่สังเคราะห์ได้อย่างมาก เนื่องจากวงจรฮาร์ดแวร์ของตัวคูณเป็นที่มีขนาดใหญ่ และใช้เกทจำนวนมาก และยังส่งผลต่อการหน่วงเวลาของวงจรที่จะมีค่ามากขึ้นด้วย ทำให้การทำงานโดยรวมของการแปลงดิสครีตโคซายน์ใช้เวลานานขึ้นด้วย

3.6.2 โครงสร้างของการแปลงดิสครีตโคซายน์ (Discrete Cosine Transform)

การสร้างอุปกรณ์การแปลงดิสครีตโคซายน์จะใช้สมการ 3.6 ซึ่งได้ผ่านการจัดรูปแบบมาแล้ว ทำให้มีการใช้จำนวนของตัวคูณน้อยกว่าสมการ 3.4 ที่เป็นสมการพื้นฐานถึง 40 ตัว

3.6.2.1 ส่วนของวงจรบวกและลบ

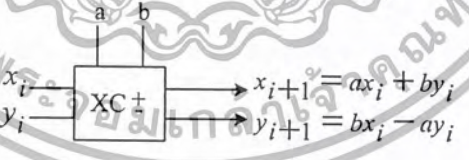
เป็นบล็อกไดอะแกรมที่แสดงให้เห็นถึงการนำสัญญาณอินพุทที่เข้ามา 2 ทาง มาทำการบวกและลบกันได้เป็นเอาท์พุทออกมา 2 ทาง ซึ่งทางหนึ่งจะเป็นการบวกกัน ส่วนอีกทางหนึ่งจะเป็นการลบกันของสัญญาณอินพุท โดยที่สัญญาณอินพุททั้ง 2 ทาง จะต้องมีขนาดของบิทข้อมูลที่เท่ากัน และได้เอาท์พุทที่มีขนาดของบิทข้อมูลมากกว่าอินพุทอยู่ 1 บิท สามารถแสดงบล็อกไดอะแกรมของการบวกและลบได้ดังรูปที่ 3.10



รูปที่ 3.10 แสดงบล็อกไดอะแกรมที่ใช้ในการบวกและลบ

3.6.2.2 ส่วนของวงจรคูณกับค่าสัมประสิทธิ์ในการแปลงดิสครีตโคซายน์

เป็นบล็อกไดอะแกรมที่แสดงถึงการนำสัญญาณอินพุทมาคูณกับค่าสัมประสิทธิ์ที่ทำการลดจำนวนตัวคูณลงแล้ว สามารถแสดงบล็อกไดอะแกรมของการคูณค่าสัมประสิทธิ์ได้ดังรูปที่ 3.11



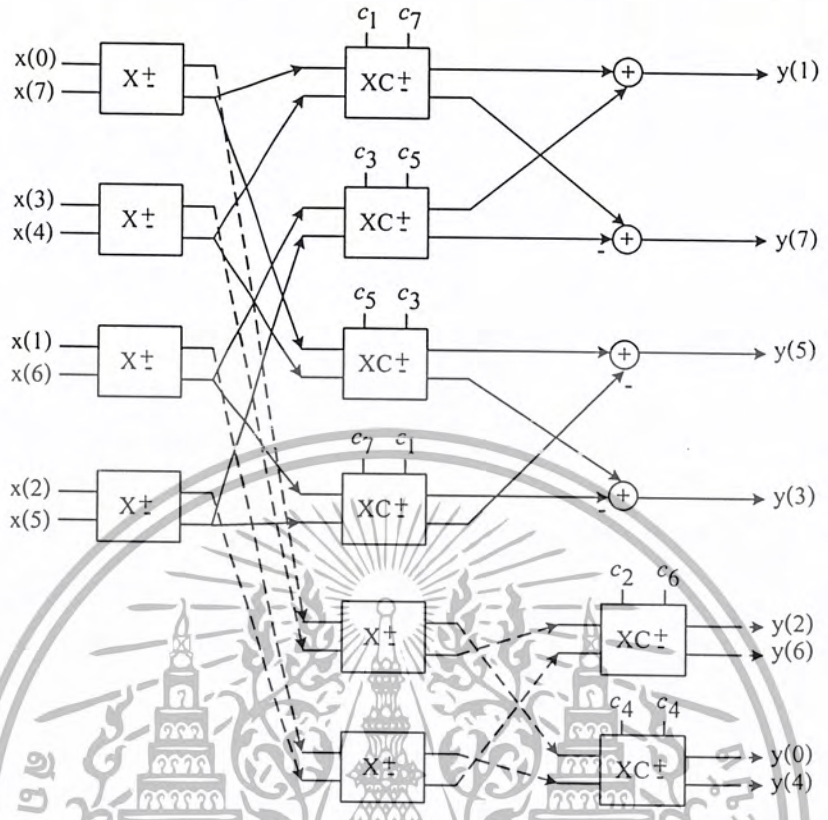
รูปที่ 3.11 แสดงบล็อกไดอะแกรมของการคูณค่าสัมประสิทธิ์
ในการแปลงดิสครีตโคซายน์

ส่วนของวงจรคูณค่าอินพุทกับค่าสัมประสิทธิ์ในการแปลงดิสครีตโคซายน์ยังสามารถเขียนให้อยู่ในรูปของผลคูณของเมตริกซ์ได้ดังนี้

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} a & b \\ b & -a \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{3.7}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2.3 โครงสร้างโดยรวมของการแปลงดิสครีตโคซายน์ (Discrete Cosine Transform)



รูปที่ 3.12 โครงสร้างโดยรวมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ

3.6.3 การออกแบบที่นำหลักการของ CORDIC Algorithm มาประยุกต์ใช้กับการแปลงดิสครีต-โคซายน์

3.6.3.1 การคำนวณหาสมการและสัมประสิทธิ์ที่ใช้ในการสร้าง

จากหลักการของ CORDIC Algorithm ซึ่งจะแสดงรูปแบบของสมการพื้นฐานได้ดังนี้

$$\begin{aligned} x_{i+1} &= x_i \cos\theta_i - y_i \sin\theta_i \\ y_{i+1} &= x_i \sin\theta_i + y_i \cos\theta_i \end{aligned} \tag{3.8}$$

สามารถแสดงอยู่ในรูปแบบของเมตริกซ์ได้ดังนี้

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{3.9}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าสัมประสิทธิ์สามารถแสดงให้อยู่ในรูปแบบของฟังก์ชันตรีโกณมิติสามารถแสดงได้ดังตารางที่ 3.1

ตารางที่ 3.1 แสดงค่าสัมประสิทธิ์ในรูปแบบของฟังก์ชันตรีโกณมิติ

ค่าสัมประสิทธิ์	ฟังก์ชันโคไซน์ (cosine)	ฟังก์ชันไซน์ (sine)
c_1	$\cos\left(\frac{\pi}{16}\right)$	$\sin\left(\frac{7\pi}{16}\right)$
c_2	$\cos\left(\frac{2\pi}{16}\right)$	$\sin\left(\frac{6\pi}{16}\right)$
c_3	$\cos\left(\frac{3\pi}{16}\right)$	$\sin\left(\frac{5\pi}{16}\right)$
c_4	$\cos\left(\frac{4\pi}{16}\right)$	$\sin\left(\frac{4\pi}{16}\right)$
c_5	$\cos\left(\frac{5\pi}{16}\right)$	$\sin\left(\frac{3\pi}{16}\right)$
c_6	$\cos\left(\frac{6\pi}{16}\right)$	$\sin\left(\frac{2\pi}{16}\right)$
c_7	$\cos\left(\frac{7\pi}{16}\right)$	$\sin\left(\frac{\pi}{16}\right)$

จากสมการ 3.6 ทำการแทนค่าของสัมประสิทธิ์ด้วยค่าในตารางที่ 3.1 แล้วจัดให้อยู่ในรูปแบบที่เหมือนกับสมการ 3.8 ที่เป็นสมการพื้นฐานของ CORDIC Algorithm ได้ดังนี้

$$
\begin{aligned}
y(0) &= (P_0 + P_3)\sin\left(\frac{4\pi}{16}\right) + (P_1 + P_2)\cos\left(\frac{4\pi}{16}\right) \\
y(4) &= (P_0 + P_3)\cos\left(\frac{4\pi}{16}\right) - (P_1 + P_2)\sin\left(\frac{4\pi}{16}\right) \\
y(2) &= (P_0 - P_3)\sin\left(\frac{6\pi}{16}\right) + (P_1 - P_2)\cos\left(\frac{6\pi}{16}\right) \\
y(6) &= (P_0 - P_3)\cos\left(\frac{6\pi}{16}\right) - (P_1 - P_2)\sin\left(\frac{4\pi}{16}\right)
\end{aligned}
\qquad (3.10)
$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(1)=\left[M_0\sin\left(\frac{7\pi}{16}\right)+M_3\cos\left(\frac{7\pi}{16}\right)\right]+\left[M_1\sin\left(\frac{5\pi}{16}\right)+M_2\cos\left(\frac{5\pi}{16}\right)\right]$$

$$y(7)=\left[M_0\cos\left(\frac{7\pi}{16}\right)-M_3\sin\left(\frac{7\pi}{16}\right)\right]-\left[M_1\cos\left(\frac{5\pi}{16}\right)-M_2\sin\left(\frac{5\pi}{16}\right)\right]$$

$$y(3)=\left[M_0\cos\left(\frac{3\pi}{16}\right)-M_3\sin\left(\frac{3\pi}{16}\right)\right]-\left[M_1\sin\left(\frac{\pi}{16}\right)+M_2\cos\left(\frac{\pi}{16}\right)\right]$$

$$y(5)=\left[M_0\sin\left(\frac{3\pi}{16}\right)+M_3\cos\left(\frac{3\pi}{16}\right)\right]-\left[M_1\cos\left(\frac{\pi}{16}\right)-M_2\cos\left(\frac{\pi}{16}\right)\right]$$

3.6.4 โครงสร้างของ CORDIC Algorithm ที่นำไปประยุกต์ใช้ในการแปลงดิสครีตโคซายน์

3.6.4.1 ส่วนของวงจรที่ใช้หลักการของ CORDIC Algorithm

รูปที่ 3.13 (a) บล็อกไดอะแกรมที่ยังไม่ใช้หลักการของ CORDIC Algorithm
(b) บล็อกไดอะแกรมที่ใช้หลักการของ CORDIC Algorithm

จากรูปที่ 3.13 จะเห็นว่าได้นำเอาหลักการของ CORDIC Algorithm มาใช้แทนวงจรคูณใน การแปลงดิสครีตโคซายน์ โดยจะใช้หลักการหมุนจากค่าอินพุทที่เข้ามาด้วยมุม θ (rotθ) ซึ่งมุม θ นี้ได้มาจากค่ามุมของสัมประสิทธิ์ในการแปลงดิสครีตโคซายน์ (c_k) และในการสร้างจะใช้ทฤษฎีเลขคณิตกระจาย (DA Algorithm) จะได้ดังนี้

$$x=-x(0)+\sum_{j=1}^{B-1}x(j)2^{-j}$$

$$y=-y(0)+\sum_{j=1}^{B-1}y(j)2^{-j} \qquad (3.11)$$

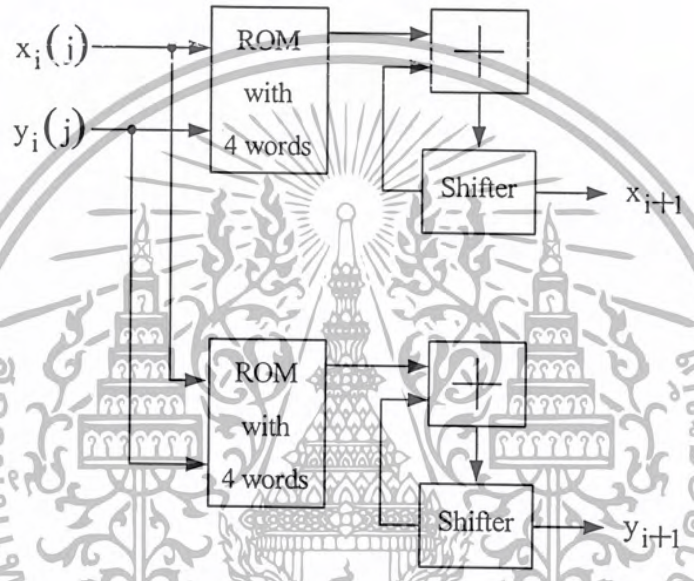
โดยที่ B คือ จำนวนบิทของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำสมการ 3.11 แทนลงไปในสมการ 3.8

$$x_{i+1} = \sum_{j=1}^{B-1} (x_i(j)\cos\theta_i - y_i(j)\sin\theta_i)2^{-j} - (x_i(0)\cos\theta_i - y_i(0)\sin\theta_i)$$
$$y_{i+1} = \sum_{j=1}^{B-1} (y_i(j)\cos\theta_i + x_i(j)\sin\theta_i)2^{-j} - (y_i(0)\cos\theta_i + x_i(0)\sin\theta_i) \quad (3.12)$$

จากสมการ 3.12 สามารถเขียนเป็นโครงสร้างได้ดังรูป 3.14



รูปที่ 3.14 โครงสร้างของ CORDIC Algorithm โดยใช้ทฤษฎีเลขคณิตกระจาย

ค่าใน ROM เป็นไปได้ 4 กรณี เก็บใน ROM 4 แอดเดรส เนื่องจากมีค่าอินพุท 2 ตัว คือ $x_i(j)$ กับ $y_i(j)$ และค่าของอินพุท แต่ละตัวสามารถเป็นไปได้แค่ 0 กับ 1 เท่านั้น สามารถแสดงค่าของอินพุท 2 ค่า ที่เป็นตัวชี้ค่าภายใน ROM ได้ดังตารางที่ 3.2

ตารางที่ 3.2 แสดงการชี้ค่าภายใน ROM ด้วยอินพุท 2 ค่า

$x_i(j)$	$y_i(j)$	ค่าภายใน ROM
0	0	A
0	1	B
1	0	C
1	1	D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ 3.12 สามารถหาค่าของภายใน ROM (A, B, C, D) ได้ดังนี้

- การคำนวณค่าภายใน ROM ของการหาค่า X_{i+1} จากสมการที่ 3.12 สามารถหาได้ดังนี้

$$A_x = 0$$
$$B_x = -\sin\theta_i$$
$$C_x = \cos\theta_i$$
$$D_x = \cos\theta_i - \sin\theta_i$$

- การคำนวณค่าภายใน ROM ของการหาค่า Y_{i+1} จากสมการที่ 3.12 สามารถหาได้ดังนี้

$$A_y = 0$$
$$B_y = \cos\theta_i$$
$$C_y = \sin\theta_i$$
$$D_y = \cos\theta_i + \sin\theta_i$$

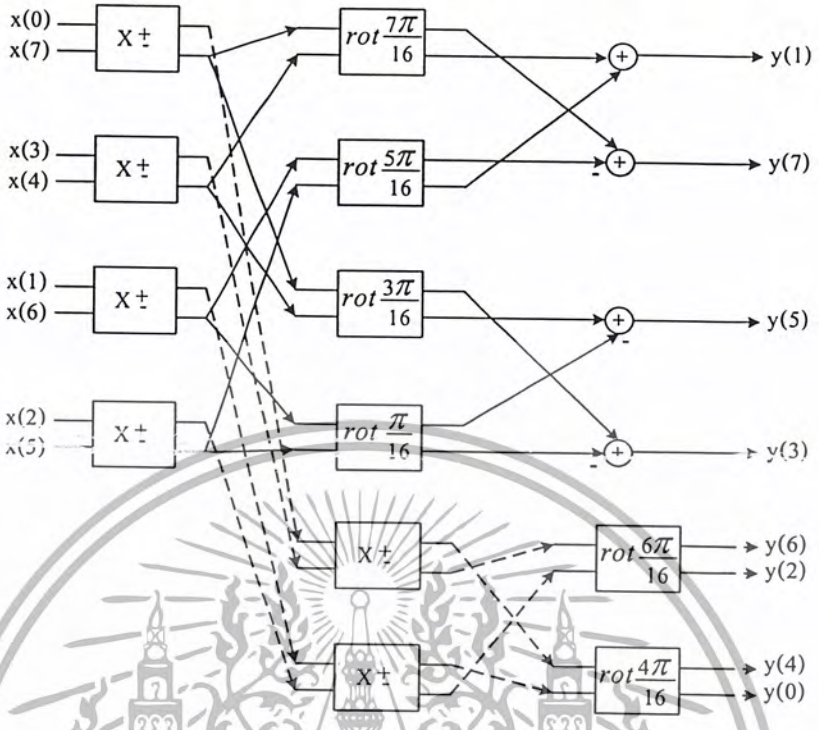
ซึ่งจะสามารถแสดงค่าสัมประสิทธิ์ที่ได้จากการคำนวณค่าภายใน ROM ที่ใช้ในการแปลง เป็นเลขฐาน 2 โดยใช้ค่ามุมต่าง ๆ ดังตารางที่ 3.3

ตารางที่ 3.3 แสดงค่าสัมประสิทธิ์ที่ใช้ในการแปลง ที่ค่ามุมต่าง ๆ

ค่ามุม(องศา)		ค่าสัมประสิทธิ์ที่ใช้ในการแปลงเป็นเลขฐาน 2			
		A	B	C	D
11.25	x	00000000000	11111001110	00011111011	00011001001
	y	00000000000	00011111011	00000110001	00100101101
33.75	x	00000000000	11101110010	00011010100	00001000110
	y	00000000000	00011010100	00010001110	00101100011
45	x	000000000000	111010010110	000101101010	000000000000
	y	000000000000	000101101010	000101101010	001011010100
56.25	x	00000000000	11100101011	00010001110	11110111001
	y	00000000000	00010001110	00011010100	00101100011
67.5	x	000000000000	111000100111	000011000011	111011101011
	y	000000000000	000011000011	000111011001	001010011100
78.75	x	00000000000	11100000101	00000110011	11100110111
	y	00000000000	00000110011	00011111011	00100101101

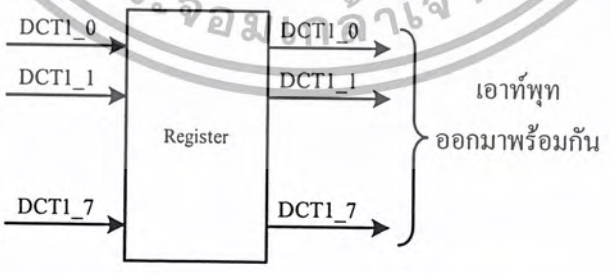
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.4.2 โครงสร้างของการแปลงดิสครีตโคซายน์ ที่ใช้หลักการของ CORDIC Algorithm



รูปที่ 3.15 แสดงโครงสร้างการแปลงดิสครีตโคซายน์แบบ 1 มิติ ที่ใช้หลักการของ CORDIC Algorithm

เนื่องจากผลลัพธ์ที่ออกมาจากการแปลงดิสครีต โคซายน์แบบ 1 มิติ แต่ละตัวจะผ่านการคำนวณที่ต่างกันจึงทำให้ผลลัพธ์ออกมาไม่พร้อมกัน ดังนั้นต้องนำผลลัพธ์ที่ได้ไปผ่านวงจรรีจิสเตอร์เพื่อให้ได้เอาท์พุทแต่ละตัวออกมาพร้อมๆ กัน สามารถแสดงบล็อกไดอะแกรมของวงจรรีจิสเตอร์ได้ดังรูปที่ 3.16

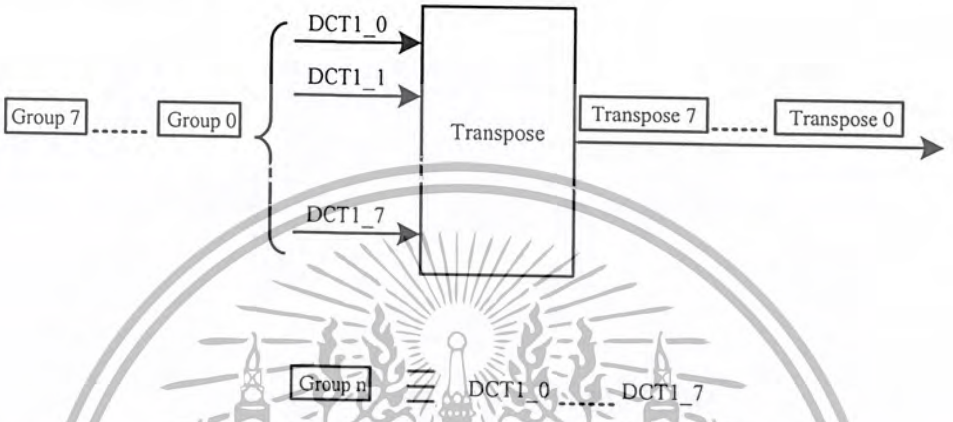


รูปที่ 3.16 แสดงบล็อกไดอะแกรมของวงจรรีจิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ส่วนการคำนวณและการสร้างของวงจรทรานสโพส (Transpose)

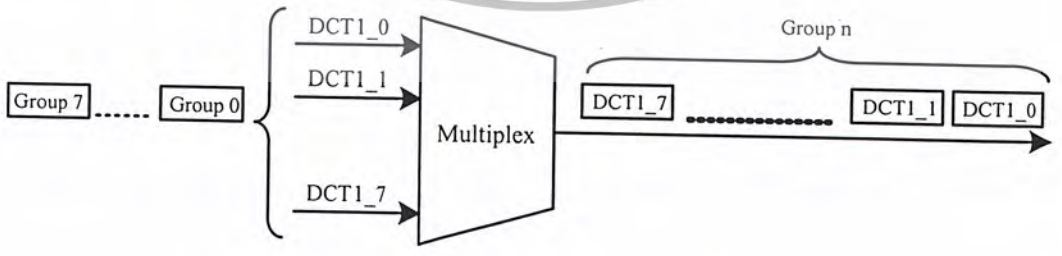
บล็อกทรานสโพสในการแปลงดิสครีตโคซายน์แบบ 2 มิติ ทำหน้าที่นำเอาเอาท์พุทของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ที่ออกมาครั้งละ 8 ค่า (แบบขนาน) มาเก็บค่าในเมตริกซ์จนเต็มบล็อกขนาด 8×8 แล้วค่อยนำค่าในบล็อกไปทำการแปลงดิสครีตโคซายน์แบบ 1 มิติ อีกครั้ง สามารถแสดงบล็อกไดอะแกรมของการทรานสโพส ได้ดังรูปที่ 3.17 ซึ่งวงจรทรานสโพส ประกอบด้วย 2 ส่วน คือ ส่วนของวงจรมัลติเพล็กซ์ (Multiplex) และส่วนของแรม (RAM)



รูปที่ 3.17 แสดงบล็อกไดอะแกรมการทรานสโพส

3.7.1 ส่วนของการคำนวณและการสร้างของวงจรมัลติเพล็กซ์ (Multiplex) ในวงจรทรานสโพส

เมื่อได้รับค่าเอาท์พุทจากการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทั้ง 8 ตัวแล้วจึงนำข้อมูลมาเรียงใหม่คือ นำค่าข้อมูลที่ออกมาจากการแปลงดิสครีตโคซายน์แบบ 1 มิติ แต่ละตัวตั้งแต่ (DCT1_0 – DCT1_7) มาทำการมัลติเพล็กซ์ (Multiplex) เพื่อให้ได้เอาท์พุทเพียงช่องสัญญาณเดียวผลลัพธ์ของการมัลติเพล็กซ์จะเรียงลำดับของอินพุทตั้งแต่ข้อมูลจาก DCT1_0, DCT1_1,..., DCT1_7 สามารถแสดงบล็อกไดอะแกรมของการมัลติเพล็กซ์ข้อมูลจากจากส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติส่วนที่1 ดังแสดงในรูปที่ 3.18

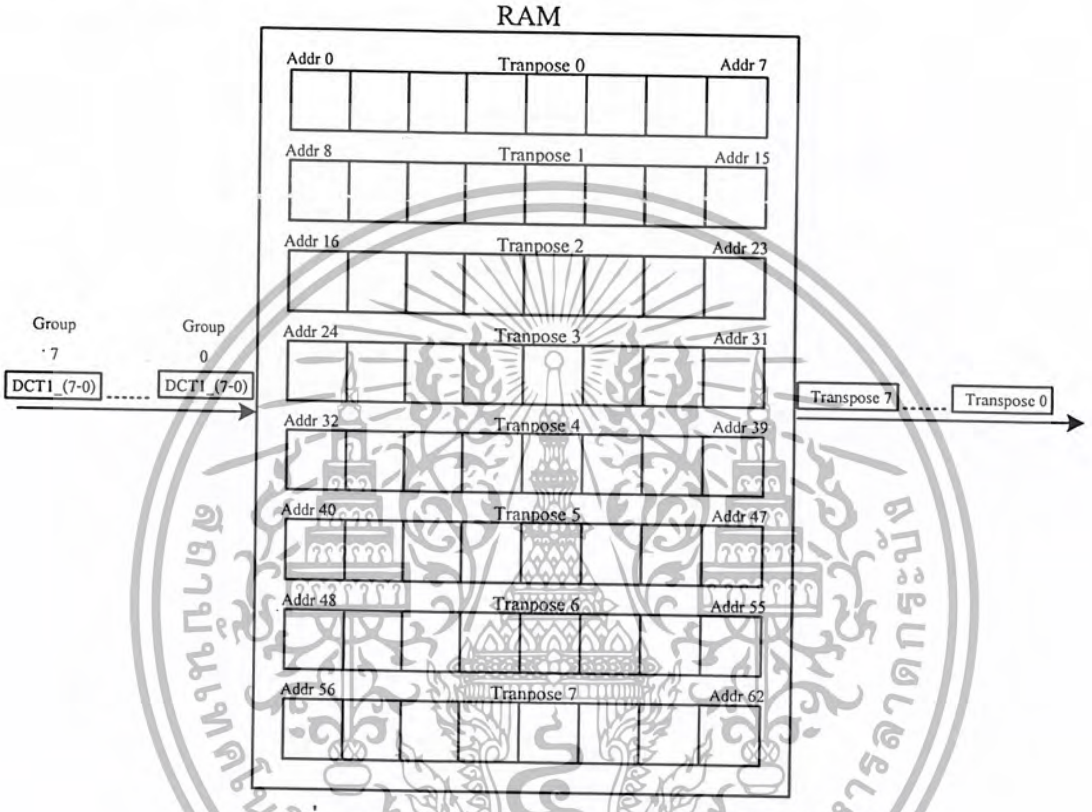


รูปที่ 3.18 แสดงบล็อกไดอะแกรมของการมัลติเพล็กซ์ข้อมูลจากส่วนของ
การแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.2 ส่วนการคำนวณและการสร้างของวงจรทรานสโพสโดยใช้แรม

ในที่นี้นำคุณลักษณะของแรม(RAM) มาใช้งานในเก็บข้อมูลแทนเมตริกซ์ ซึ่งจะนำเอาเอาท์พุทจากวงจรมัลติเพล็กซ์ (Multiplex) มาเก็บค่าในแรมจนเต็มทั้ง 64 แอดเดรส แต่ละแอดเดรสเก็บข้อมูล 10 บิทโดยเก็บข้อมูลตามการอ้างอิงแอดเดรสตามที่ต้องการ เพื่อให้ได้ค่าข้อมูลเอาท์พุทที่ได้เป็นค่าทรานสโพสของบล็อกข้อมูลอินพุทที่รับเข้ามา สามารถแสดงบล็อกไดอะแกรมการทรานสโพสโดยใช้แรมได้ดังรูปที่ 3.19



รูปที่ 3.19 แสดงบล็อกไดอะแกรมการทรานสโพสโดยใช้แรม

การรับข้อมูลของการทรานสโพสมีขั้นตอนการทำงาน ดังนี้

อินพุทชุด 0 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n เมื่อ n = 0, 1, 2, ..., 7

อินพุทชุด 1 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n + 1 เมื่อ n = 0, 1, 2, ..., 7

อินพุทชุด 2 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n + 2 เมื่อ n = 0, 1, 2, ..., 7

อินพุทชุด 3 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n + 3 เมื่อ n = 0, 1, 2, ..., 7

อินพุทชุด 4 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n + 4 เมื่อ n = 0, 1, 2, ..., 7

อินพุทชุด 5 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n + 5 เมื่อ n = 0, 1, 2, ..., 7

อินพุทชุด 6 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n + 6 เมื่อ n = 0, 1, 2, ..., 7

อินพุทชุด 7 ที่รับเข้ามาตั้งแต่ DCT1_0 – DCT1_7 จะเก็บที่แอดเดรส 8n + 7 เมื่อ n = 0, 1, 2, ..., 7

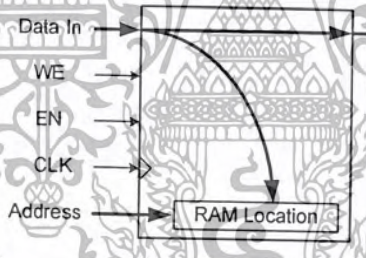
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเก็บข้อมูลจนครบทั้ง 64 ค่า แล้วก็จะนำค่าเอาท์พุทของการทรานสโพส ออกจากแรม โดยเรียงจากแอดเดรสที่ 0 ถึงแอดเดรสที่ 63 แต่จะนำเอาท์พุทออกทีละ 8 ค่า เพื่อนำไปเข้าวงจรคงค่าสัญญาณและการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 ต่อไป ซึ่งลักษณะการส่งข้อมูลของการทรานโพสออกจากแรมแสดงได้ดังต่อไปนี้

เอาท์พุทชุด 0 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 0 - 7 (Transpose 0)
เอาท์พุทชุด 1 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 8 - 15 (Transpose 1)
เอาท์พุทชุด 2 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 16 - 23 (Transpose 2)
เอาท์พุทชุด 3 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 24 - 31 (Transpose 3)
เอาท์พุทชุด 4 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 32 - 39 (Transpose 4)
เอาท์พุทชุด 5 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 40 - 47 (Transpose 5)
เอาท์พุทชุด 6 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 48 - 55 (Transpose 6)
เอาท์พุทชุด 7 คือข้อมูลอนุกรม ตั้งแต่แอดเดรสที่ 56 - 63 (Transpose 7)

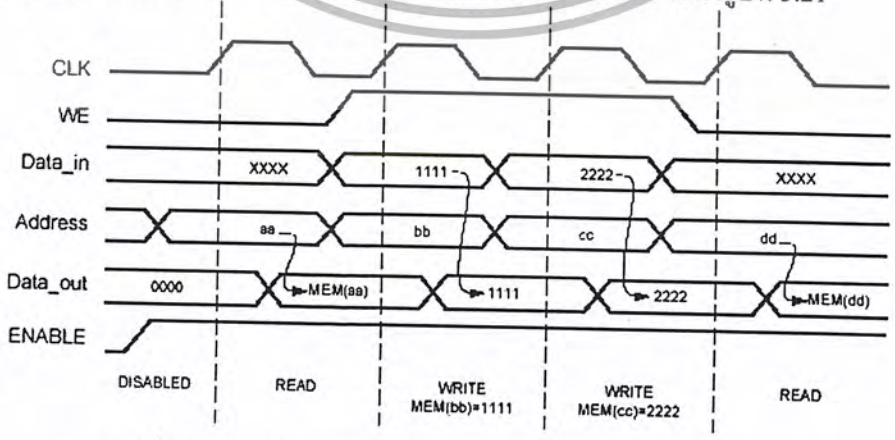
3.7.3 ส่วนของวงจรแรมที่ใช้สร้างจริงบนบอร์ด FPGA

ในส่วนนี้จะเลือกใช้ RAM ที่มี write mode เป็นแบบ write first mode โดยจะมีลักษณะการทำงาน ดังรูปที่ 3.20



รูปที่ 3.20 แสดงการไหลของข้อมูลในระหว่างกระบวนการเขียนข้อมูล

สามารถแสดงลักษณะการทำงานของแรมในโหมด Write-first ได้ดังรูปที่ 3.21



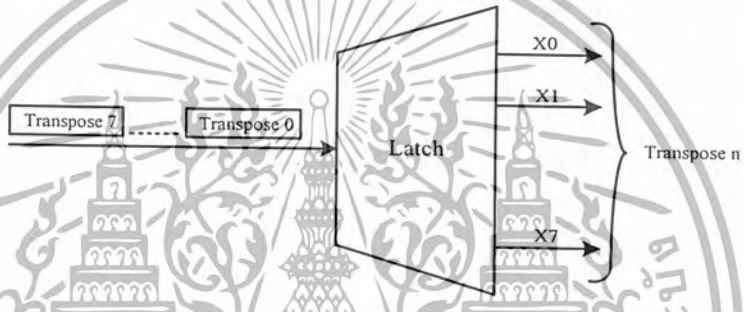
รูปที่ 3.21 แสดงลักษณะการทำงานของ RAM ในโหมด Write first

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปสามารถอธิบายกระบวนการเขียนและอ่านข้อมูลได้ดังนี้ RAM จะทำงานเมื่อมีสัญญาณ Enable อินพุท เป็น 1 และจะมีแอดเดรสเป็นตัวกำหนดตำแหน่งจัดการเก็บข้อมูล ซึ่งเมื่อ WE มีค่าเป็น 1 RAM จะทำหน้าที่เขียนข้อมูล อินพุทลงในแอดเดรสของ RAM ที่ปรากฏ พร้อมกับส่งข้อมูลออกเมื่อ WE มีค่าเป็น 0 จะทำหน้าที่อ่านข้อมูลจากแอดเดรสของ RAM ที่กำหนดไว้

3.8 การออกแบบส่วนของวงจรคงค่าสัญญาณ ส่วนที่ 2 (Latch)

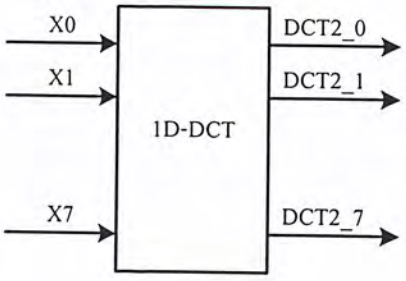
นำข้อมูลที่ได้จากวงจรทรานสโพสที่ข้อมูลมีลักษณะเป็นแบบอนุกรมในช่องสัญญาณเดียวมาทำการจัดรูปแบบข้อมูลใหม่ให้อยู่ในรูปแบบขนาน โดยผ่านส่วนของวงจรคงค่าสัญญาณ (Latch) ที่ทำหน้าที่รอข้อมูลจนครบ 8 ค่า (เท่ากับข้อมูลใน Transpose n) แล้วจึงปล่อยข้อมูลทั้ง 8 ค่า ออกไปพร้อมกันในลักษณะแบบขนาน สามารถแสดงบล็อกไดอะแกรมของวงจรคงค่าสัญญาณส่วนที่ 2 ได้ดังแสดงรูปที่ 3.22



รูปที่ 3.22 แสดงบล็อกไดอะแกรมของวงจรคงค่าสัญญาณ ส่วนที่ 2

3.9 การออกแบบส่วนการแปลงดิสครีตโคซายน์แบบ1มิติ (1D-DCT) ส่วนที่ 2

การแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 เป็นการนำข้อมูลเอาท์พุทจากการทรานสโพสและผ่านวงจรคงค่าสัญญาณแล้วเข้ามาเป็นข้อมูลอินพุทของการแปลงดิสครีตโคซายน์แบบ 1 มิติ โดยรับอินพุทมาครั้งละ 8 ค่า และให้เอาท์พุทมาครั้งละ 8 ค่า จนครบทั้ง 64 ค่า (1 บล็อกข้อมูล) เหมือนกับการแปลงในส่วนแรกที่ได้กล่าวมาแล้ว ซึ่งจะได้ผลลัพธ์ของการแปลงดิสครีตโคซายน์แบบ 2 มิติ ของข้อมูล 1 บล็อกขนาด 8×8 พิกเซล ตามต้องการ สามารถแสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 ได้ดังรูปที่ 3.23

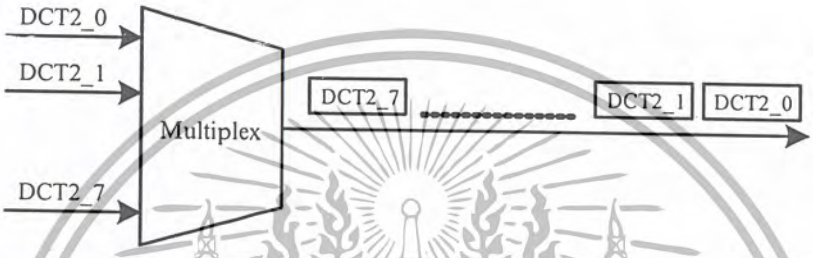


รูปที่ 3.23 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10 ส่วนของการคำนวณและการสร้างของวงจรมัลติเพล็กซ์ (Multiplex) หลังการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2

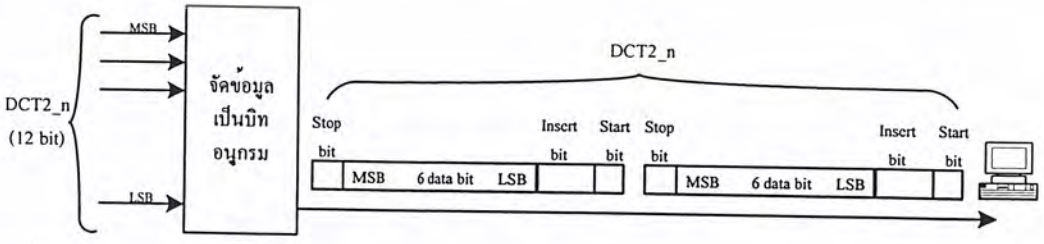
เมื่อได้รับค่าเอาท์พุทจากการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทั้ง 8 ตัวแล้วจึงนำข้อมูลมาเรียงใหม่คือ นำค่าข้อมูลที่ออกมาจากการแปลงดิสครีตโคซายน์แบบ 1 มิติ แต่ละตัวตั้งแต่ (DCT2_0 – DCT2_7) มาทำการมัลติเพล็กซ์ (Multiplex) เพื่อให้ได้เอาท์พุทเพียงช่องสัญญาณเดียวผลลัพธ์ของการมัลติเพล็กซ์จะเรียงลำดับของอินพุทตั้งแต่ข้อมูลจาก DCT2_0, DCT2_1,..., DCT2_7 สามารถแสดงบล็อกไดอะแกรมของการมัลติเพล็กซ์ข้อมูลจากจากส่วนของการแปลงดิสครีตโคซายน์แบบ 2 มิติส่วนที่2 ดังแสดงในรูปที่ 3.24



รูปที่ 3.24 แสดงบล็อกไดอะแกรมของการมัลติเพล็กซ์ข้อมูลจากส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2

3.11 ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์

ส่วนของการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ ทำหน้าที่แปลงบิทข้อมูลแบบขนานขนาด 12 บิท ที่ได้จากแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 ให้เป็นบิทข้อมูลแบบอนุกรมโดยแบ่งเป็น 2 เฟรม โดยแต่ละเฟรมข้อมูลประกอบด้วย สตาร์ทบิท (Start bit) 1 บิท สตอบบิท (Stop bit) 2 บิท บิทข้อมูล (Data bit) 6 บิท และแทรกบิท (Insert bit) 2 บิท เพื่อแก้ไขกรณีการส่งบิทข้อมูลที่เป็น 0 ทั้งหมด จะทำการแทรก "01" เข้าไปใน MSB หลังจากนั้นส่งเฟรมข้อมูลดังกล่าวออกพอร์ตอนุกรม (serial port) ไปยังคอมพิวเตอร์ตามความถี่บอดเรต (Baud Rate) สามารถแสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์ดังรูปที่ 3.25

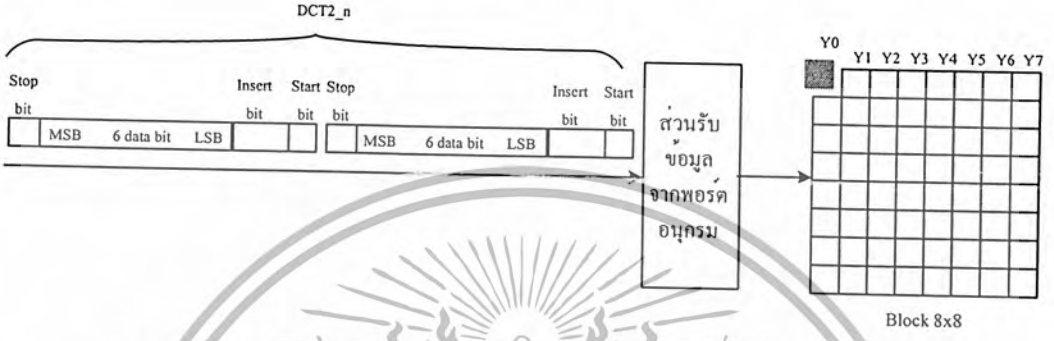


รูปที่ 3.25 แสดงบล็อกไดอะแกรมการแปลงข้อมูลส่งออกพอร์ตอนุกรมจากบอร์ด FPGA ไปยังคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.12 ส่วนของการรับบิทข้อมูลจากบอร์ด FPGA ด้วยโปรแกรม MATLAB ผ่านทางพอร์ตอนุกรม

ส่วนของการรับข้อมูลแบบอนุกรมจากบอร์ดFPGA ด้วยโปรแกรม MATLAB ทำหน้าที่ รับเฟรมบิตของข้อมูลจากพอร์ตอนุกรมตามความถี่บอดเรต (Baud Rate) มาทำการตัดสตาร์ทบิท (Start bit) และสตอบบิท (Stop bit) และบิทแทรก (Insert bit) พร้อมทั้งรวมเฟรมบิทข้อมูลที่แยกกันอยู่เพื่อให้ได้ค่าของข้อมูล 1 พิกเซล แล้วนำค่าได้เก็บในบล็อกข้อมูลขนาด 8×8 จนครบ สามารถแสดงบล็อกไดอะแกรมการรับบิทข้อมูลผ่านทางพอร์ตอนุกรมได้ดังรูปที่ 3.26



รูปที่ 3.26 แสดงบล็อกไดอะแกรมการรับบิทข้อมูลจากบอร์ด FPGA ผ่านพอร์ตอนุกรม

3.13 ส่วนของการแปลงดิสครีตโคซายน์กลับแบบ 2 มิติ ด้วยโปรแกรม MATLAB

นำบล็อกข้อมูลขนาด 8×8 ที่ได้มาทำการแปลงดิสครีตโคซายน์กลับแบบ 2 มิติ ด้วยสมการดังนี้

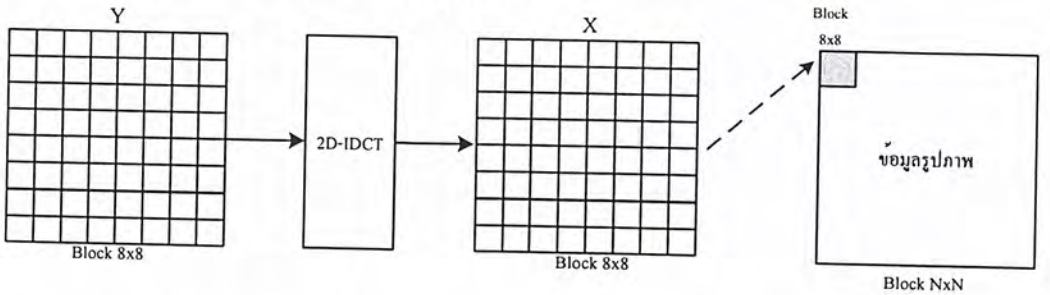
$$X = C^T Y C \tag{3.13}$$

โดย Y เป็นเอาท์พุทของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด 8×8

C เป็นสัมประสิทธิ์ของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด 8×8

X เป็นอินพุทของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด 8×8

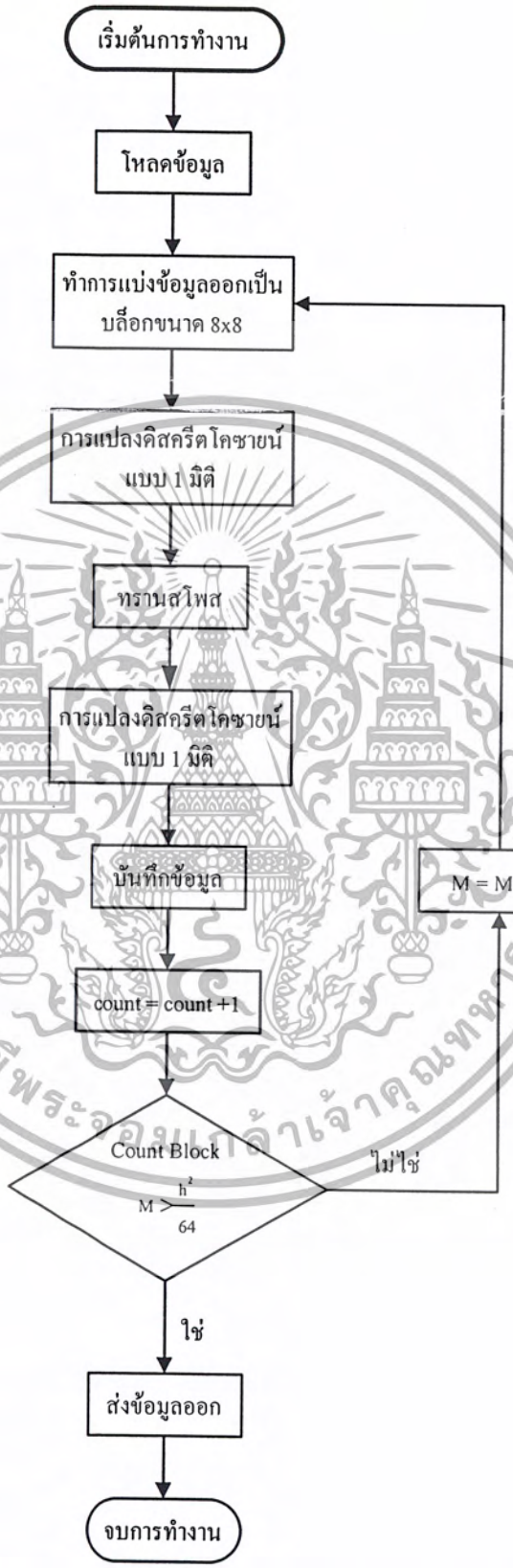
หลังจากได้บล็อกข้อมูลเดิมขนาด 8×8 จากการแปลงดิสครีตโคซายน์กลับแบบ 2 มิติ นำบล็อกข้อมูลที่ได้มาทำเก็บจนได้ข้อมูลรูปภาพเดิมครบขนาด $N \times N$ สามารถแสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์กลับแบบ 2 มิติ ได้ดังรูปที่ 3.27



รูปที่ 3.27 แสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์กลับแบบ 2 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.14 ส่วนการแสดงโฟว์ลชาร์ทของการสร้างอุปกรณ์การแปลงดิสครีตโคซายน์ 2 มิติ

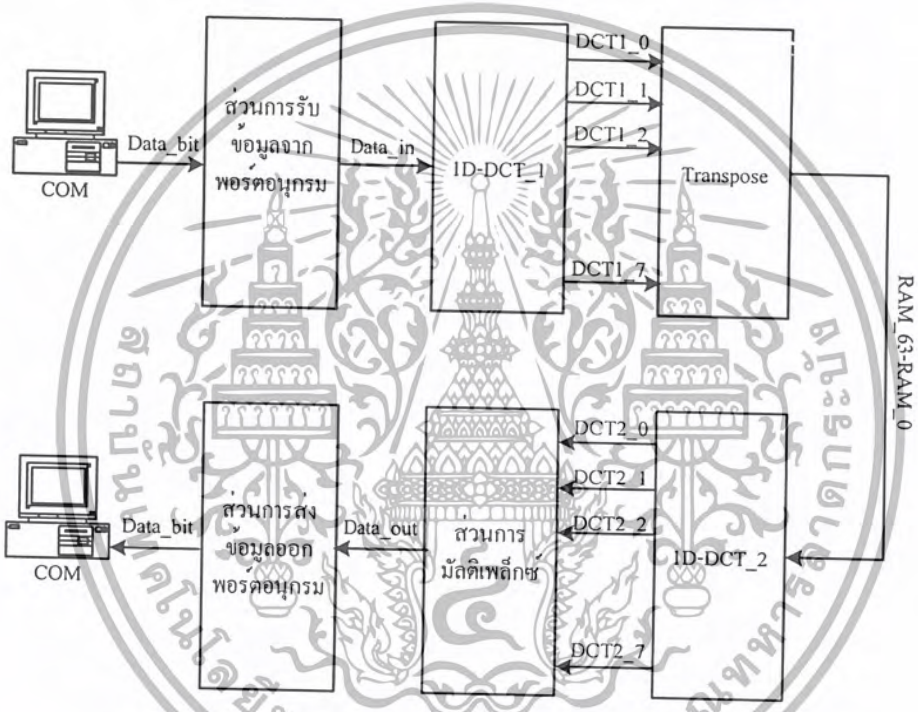


รูปที่ 3.28 แสดงโฟว์ลชาร์ทของการสร้างอุปกรณ์การแปลงดิสครีตโคซายน์ 2 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

การออกแบบส่วนต่างๆ ของการแปลงดิสครีตโคซายน์ 2 มิติ โดยใช้หลักการของคอร์ดิกอัลกอริทึม (CORDIC Algorithm) และหลักการการแยกการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง สามารถทำการเขียนโปรแกรมการทำงานโดยให้แต่ละส่วนทำงานตามที่ได้ออกแบบโดยใช้ภาษา VHDL ทำการคอมไพล์ (Compile) แล้วเขียนโปรแกรมการทำงานจริงลงบนบอร์ด FPGA ส่วนของการสื่อสารข้อมูลทางพอร์ตอนุกรมอาศัยการทำงานและออกแบบโดยใช้โปรแกรม MATLAB การออกแบบและการทดลองได้แบ่งออกเป็นส่วน ๆ ดังรูปที่ 4.1



รูปที่ 4.1 แสดงการทำงานของการแปลงดิสครีตโคซายน์แบบ 2 มิติ

4.1 ส่วนของการรับข้อมูลจากคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม

4.2 ส่วนของวงจรคงค่าสัญญาณชั่วคราวและการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1

4.3 ส่วนของวงจรทรานสโพส

4.4 ส่วนของวงจรคงค่าสัญญาณชั่วคราวและการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 หลังทำการทรานสโพส

4.5 ส่วนของการส่งข้อมูลออกจากบอร์ด FPGA ผ่านทางพอร์ตอนุกรม

4.6 ส่วนของวงจรควบคุมการทำงานของระบบ

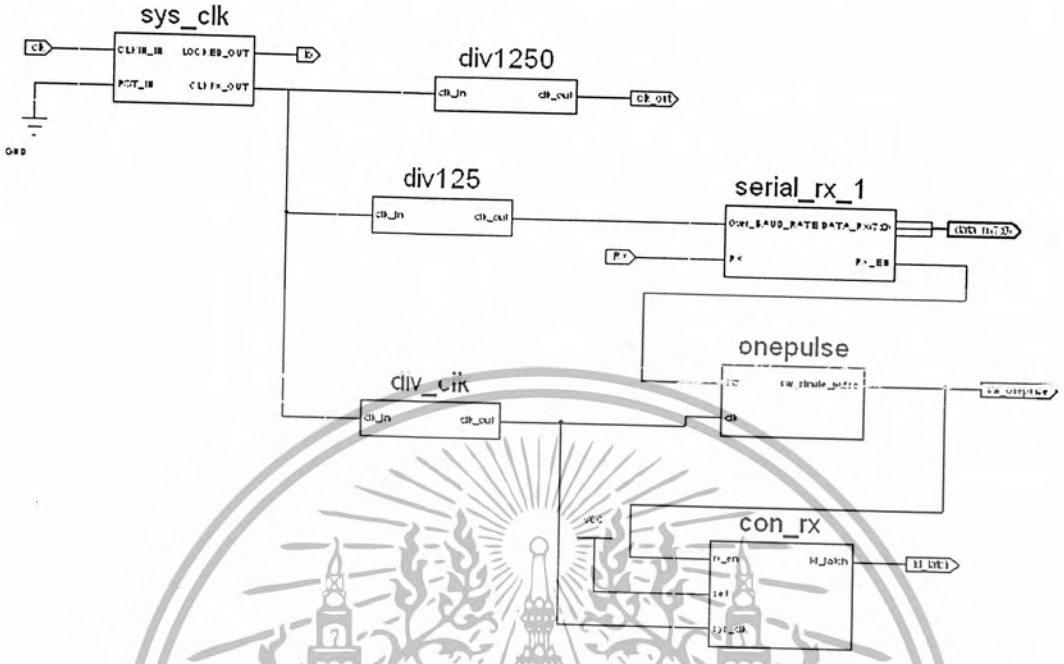
4.7 ส่วนประกอบภายในและการเชื่อมต่อของวงจรทั้งหมด

4.8 ส่วนของการประมวลผลการทำงานจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 ส่วนของการรับข้อมูลจากคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม

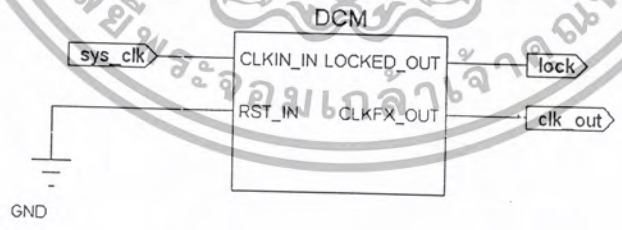
ส่วนประกอบของวงจรรับข้อมูลจากคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม



รูปที่ 4.2 แสดงส่วนประกอบของวงจรรับข้อมูลจากคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม

4.1.1 ส่วนของวงจร Digital Clock Managers (DCMs)

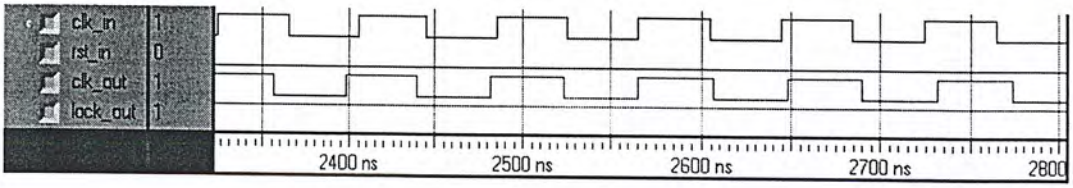
ส่วนของวงจร Digital Clock Managers (DCMs) ทำหน้าที่สร้างสัญญาณนาฬิกาเอาท์พุทให้มีความถี่ตามที่ต้องการ เพื่อให้สัญญาณนาฬิกาเอาท์พุทที่ได้เป็นสัญญาณนาฬิกาอ้างอิงของระบบ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจร DCMs ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.3



รูปที่ 4.3 แสดงสัญลักษณ์ของวงจร DCMs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.4

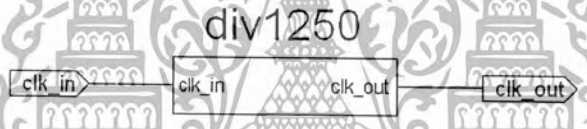


รูปที่ 4.4 แสดงผลการจำลองการทำงานของวงจร DCMs

4.1.2 ส่วนของวงจรหารความถี่

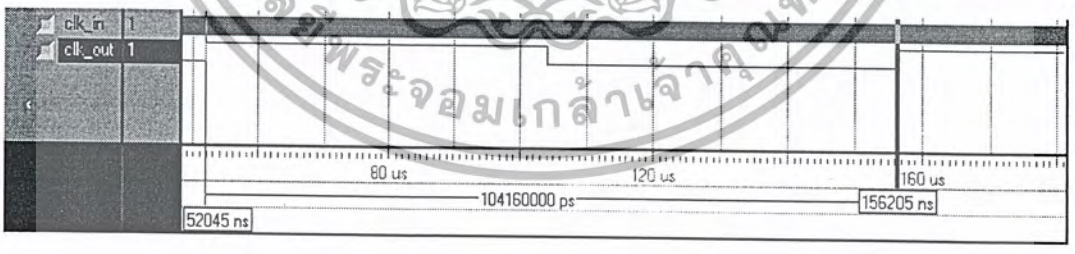
4.1.2.1 ส่วนของวงจรหารความถี่บอดเรต

ส่วนของวงจรหารความถี่บอดเรต ทำหน้าที่หารความถี่สัญญาณนาฬิกาจาก DCMs ให้ได้เอาท์พุทเป็นความถี่บอดเรต (Baud Rate) ที่ใช้ในการรับ - ส่งบิทข้อมูลผ่านทางพอร์ตอนุกรม ปริญญานิพนธ์นี้รับสัญญาณอินพุทจาก DCMs ความถี่ 24 MHz หาร 1250 ให้ได้ความถี่ 9600 Hz สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรหารความถี่ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.5



รูปที่ 4.5 แสดงสัญลักษณ์ของวงจรหารความถี่บอดเรต

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.6

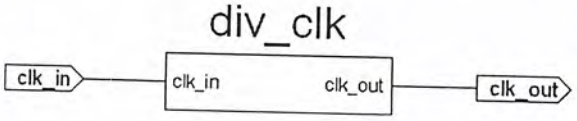


รูปที่ 4.6 แสดงผลการจำลองการทำงานของวงจรหารความถี่บอดเรต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

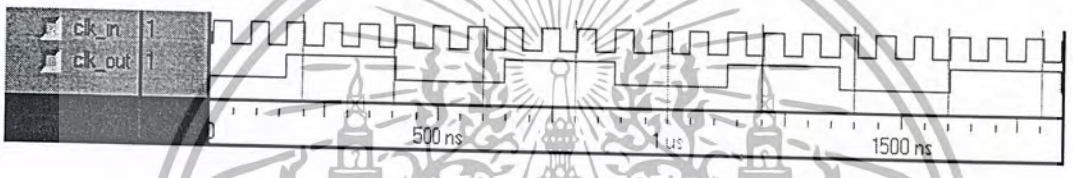
4.1.2.2 ส่วนของวงจรหารความถี่ของระบบ

ส่วนของวงจรหารความถี่ของระบบ ทำหน้าที่หารความถี่สัญญาณนาฬิกาจาก DCMs ให้ได้เอาท์พุทเป็นความถี่ 4 MHz เพื่อให้อุปกรณ์บนบอร์ด FPGA ไม่ทำงานเร็วเกินไป สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรหารความถี่ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.7



รูปที่ 4.7 แสดงสัญลักษณ์ของวงจรหารความถี่ของระบบ

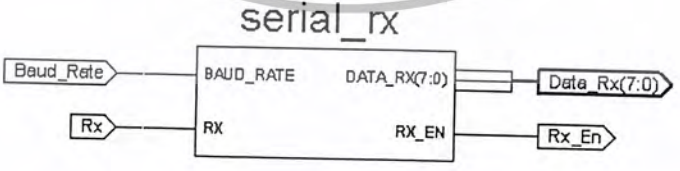
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.8



รูปที่ 4.8 แสดงผลการจำลองการทำงานของวงจรหารความถี่ของระบบ

4.1.3 ส่วนของการรับบิทข้อมูลจากพอร์ตอนุกรม (SERIAL_RX)

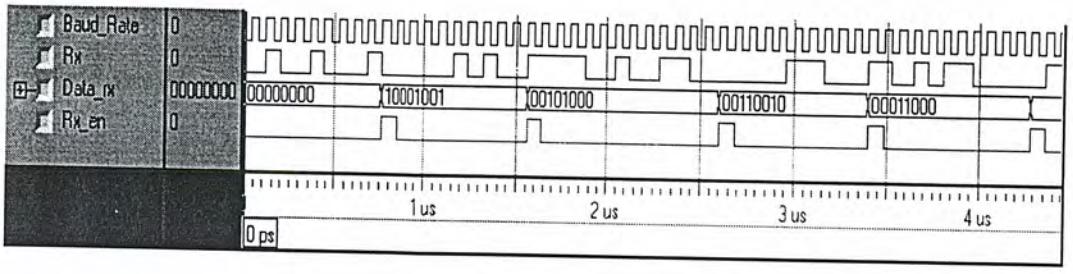
ส่วนของการรับข้อมูลจากพอร์ตอนุกรม ทำหน้าที่รับเฟรมบิทของข้อมูลจากพอร์ตอนุกรมมาทำการแปลงข้อมูลจากบิทอนุกรมเป็นบิทขนาน โดยจะทำการแซมปลิ้ง (Sampling) ค่าอินพุทที่รับเข้ามาจากพอร์ตอนุกรมโดยบิทหนึ่งทำการแซมปลิ้ง 10 ค่า จากนั้นนำค่าที่ได้จากการแซมปลิ้งมาเก็บไว้เพื่อหาค่าความเป็นไปได้ของแต่ละบิทอินพุท เพื่อป้องกันการรับข้อมูลจากพอร์ตอนุกรมผิดพลาด เมื่อรับข้อมูลครบ 8 ค่า ก็จะส่งต่อไปให้ส่วนของวงจรคงค่าสัญญาณ (Latch) ต่อไป สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรการรับบิทข้อมูลจากพอร์ตอนุกรมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.9



รูปที่ 4.9 แสดงสัญลักษณ์ของวงจรการรับบิทข้อมูลจากพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

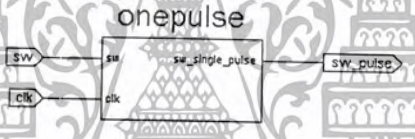
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.10



รูปที่ 4.10 แสดงผลการจำลองการทำงานของวงจรการรับบิทข้อมูลจากพอร์ตอนุกรม

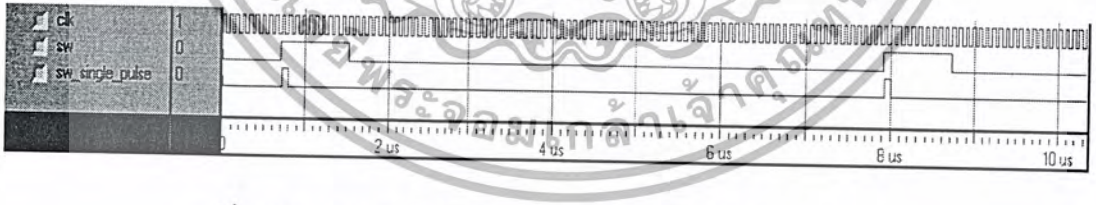
4.1.4 ส่วนของวงจรวันพัลส์ (Onepulse)

ส่วนของวงจรวันพัลส์ ทำหน้าที่เปลี่ยนขนาดความกว้างของพัลส์ (Pulse) อินพุทที่เป็นไปตามความถี่บอดเรต (Baud Rate) เพื่อให้ความกว้างของพัลส์เอาท์พุท เป็นไปตามความกว้างของสัญญาณนาฬิการะบบ แล้วส่งสัญญาณต่อให้วงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม (CON_RX) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรวันพัลส์ (Onepulse) ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.11



รูปที่ 4.11 แสดงสัญลักษณ์ของวงจรวันพัลส์(Onepulse)

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.12

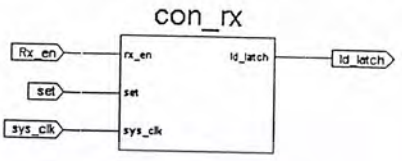


รูปที่ 4.12 แสดงผลการจำลองการทำงานของวงจรวันพัลส์ (Onepulse)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

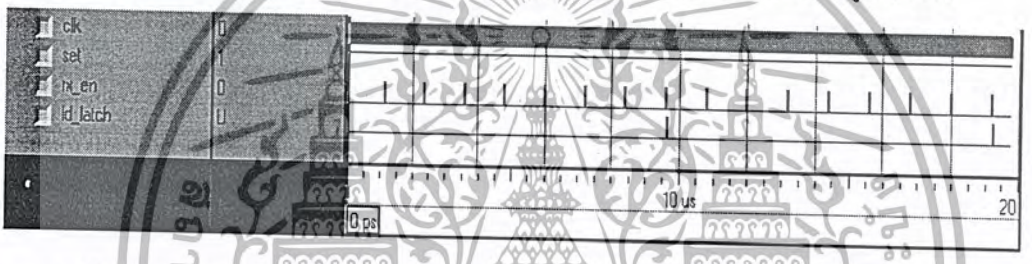
4.1.5 ส่วนของวงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม (CON_RX)

วงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม ทำหน้าที่สร้างสัญญาณควบคุมเพื่อให้การทำงานของวงจรรับบิทอนุกรมมีความถูกต้องเป็นไปตามลำดับขั้น โดยจะส่งสัญญาณทริกไปยังวงจรคงค่าสัญญาณและวงจรควบคุมรวม เมื่อรับค่าข้อมูลได้ครบ 8 ค่า สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมบิทข้อมูลจากพอร์ตอนุกรม ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.13



รูปที่ 4.13 แสดงสัญลักษณ์ของวงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.14

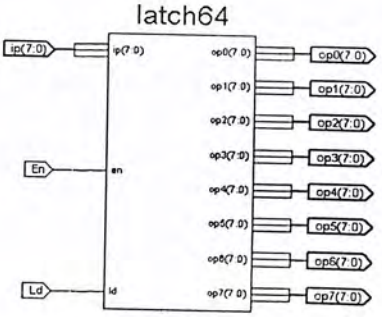


รูปที่ 4.14 แสดงผลการจำลองการทำงานของวงจรควบคุมการรับบิทข้อมูลจากพอร์ตอนุกรม

4.2 ส่วนของวงจรคงค่าสัญญาณชั่วคราวและการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1

4.2.1 ส่วนของวงจรคงค่าสัญญาณชั่วคราว 8 บิท (Latch)

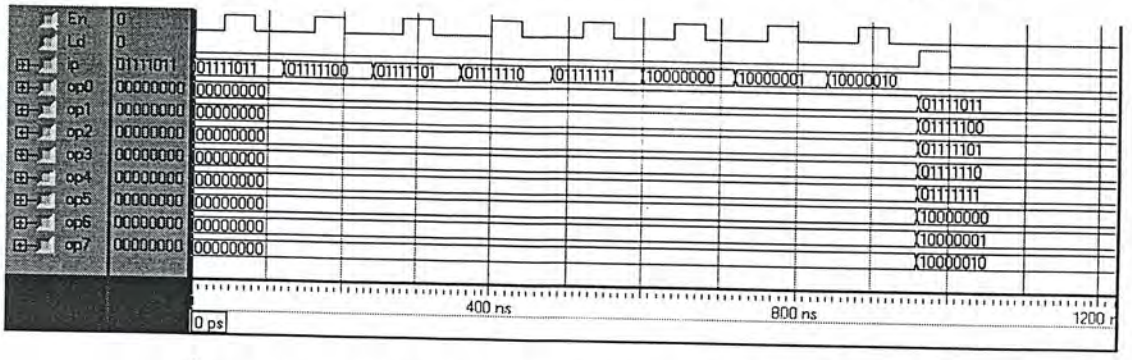
เป็นส่วนของวงจรที่ ทำหน้าที่คงค่าข้อมูลอินพุทไว้ชั่วคราว โดยจะทำการรับข้อมูลอินพุทที่ป้อนเข้ามานำมาเก็บไว้ชั่วคราวจนกว่าจะครบ 8 ค่า แต่ละค่า ๆ ละ 8 บิท แล้วทำการส่งค่าทั้ง 8 ค่าที่รับมาได้ออกมาพร้อม ๆ กัน เพื่อส่งค่าไปเป็นอินพุทของส่วนการแปลงดิสครีตโคซายน์ต่อไป แล้วทำการรับค่าข้อมูลชุดใหม่เข้ามาให้ครบทั้ง 8 ค่าในรอบถัดไป จะทำงานซ้ำแบบนี้จนกระทั่งสัญญาณอินพุทครบ 64 ค่า สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรคงค่าสัญญาณชั่วคราวขนาด 8 บิทได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.15



รูปที่ 4.15 แสดงสัญลักษณ์ของวงจรคงค่าสัญญาณชั่วคราว 8 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

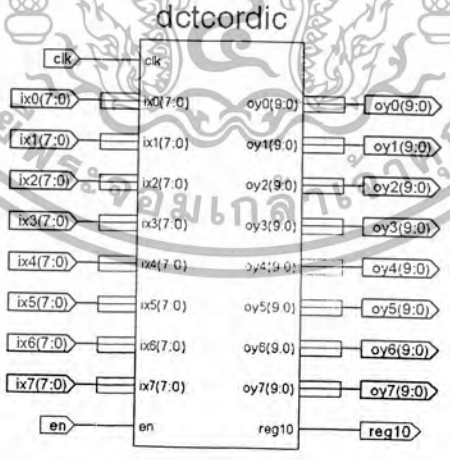
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.16



รูปที่ 4.16 แสดงผลการจำลองการทำงานของวงจรคงค่าสัญญาณชั่วคราว 8 บิท

4.2.2 ส่วนของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ

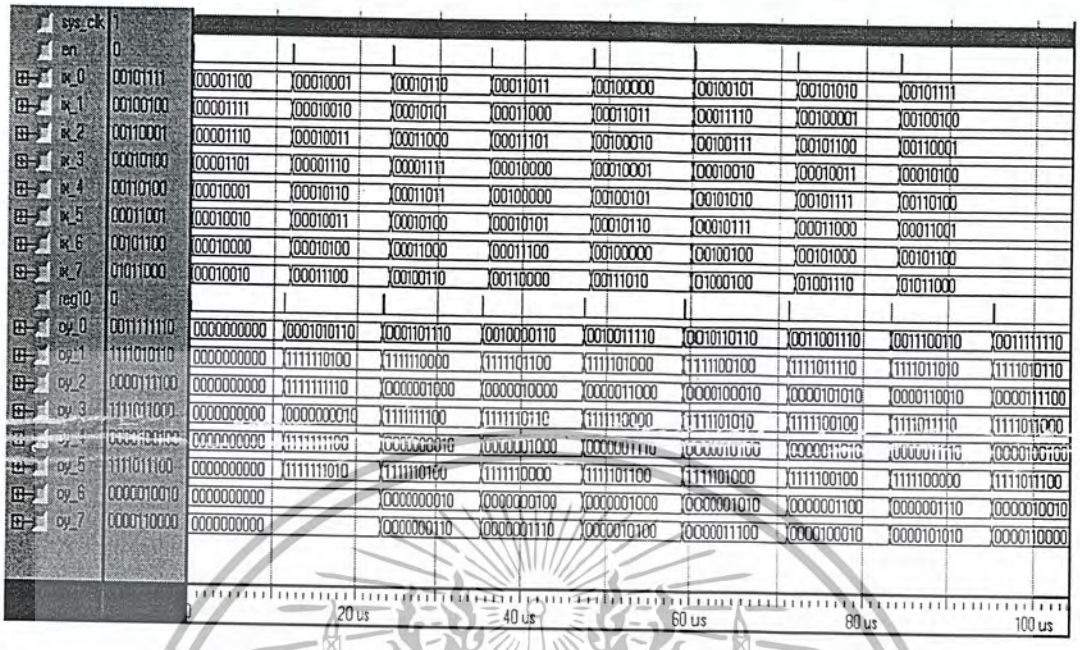
เป็นส่วนของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่ในการคำนวณค่าการแปลงดิสครีตโคซายน์แบบ 1 มิติ ของข้อมูลที่รับข้อมูลอินพุทมาจากส่วนของวงจรคงค่าสัญญาณชั่วคราว ขนาด 8 บิท 8 ค่า ซึ่งภายในส่วนของวงจรการแปลงดิสครีตโคซายน์ประกอบไปด้วยส่วนของวงจรบวกและลบขนาด 8 บิท ส่วนของวงจรบวกและลบ 9 บิท ส่วนของวงจรบวก 9 บิท ส่วนของวงจรลบ 9 บิท ส่วนของวงจรคอร์ดิก ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณทำหน้าที่ในการคูณข้อมูลกับค่าคงที่สัมประสิทธิ์ ส่วนของวงจรรีจิสเตอร์ขนาด 10 บิท และ ส่วนของวงจรคอนโทรล (controller) ซึ่งส่วนของวงจรคอนโทรลนี้เป็นส่วนที่สำคัญมาก สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.17



รูปที่ 4.17 แสดงสัญลักษณ์ของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

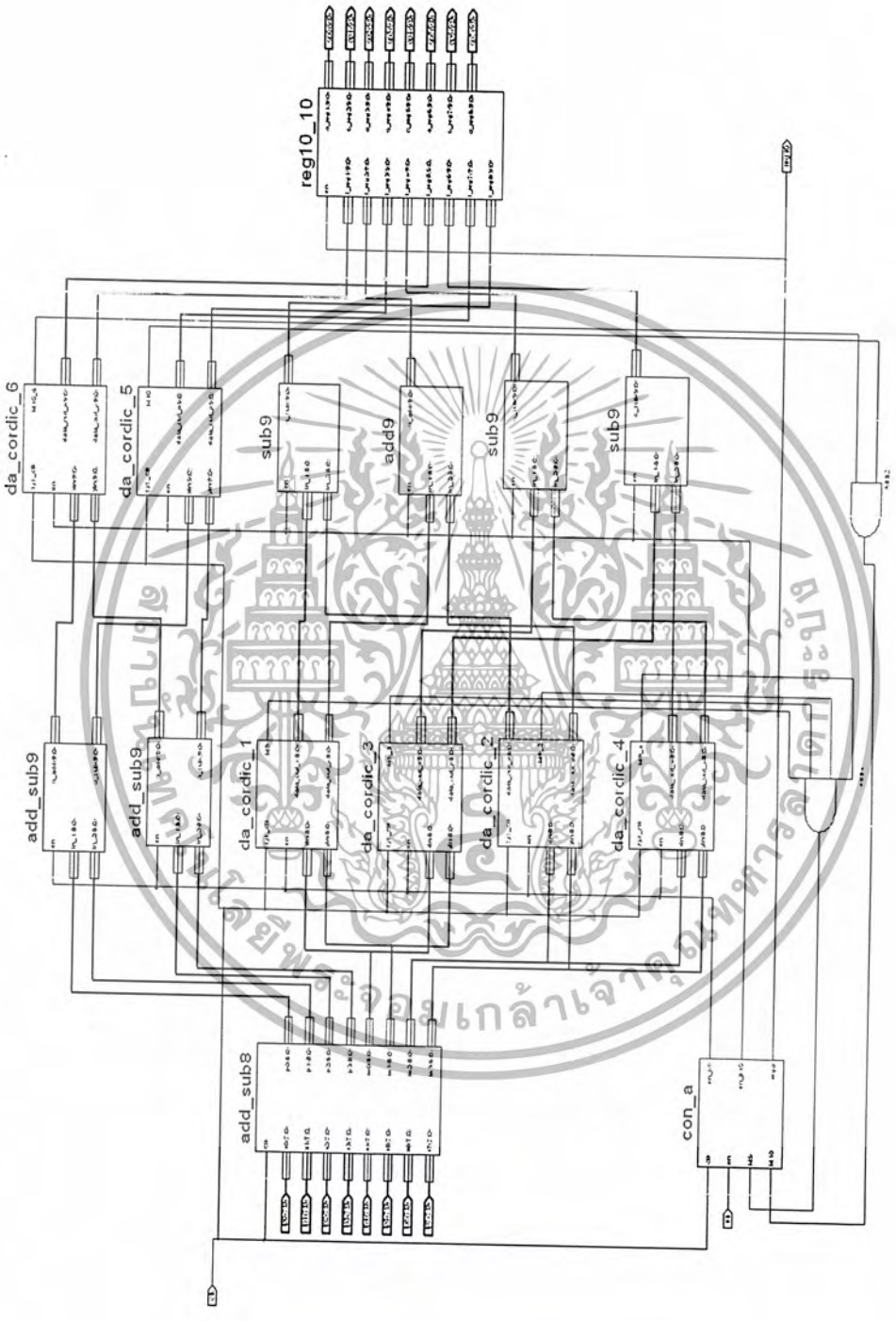
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.18



รูปที่ 4.18 แสดงผลการจำลองการทำงานของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 1 มีดังนี้

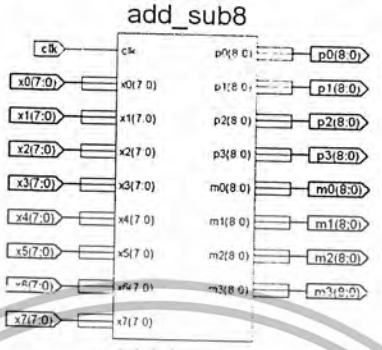


รูปที่ 4.19 แสดงส่วนประกอบภายในของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

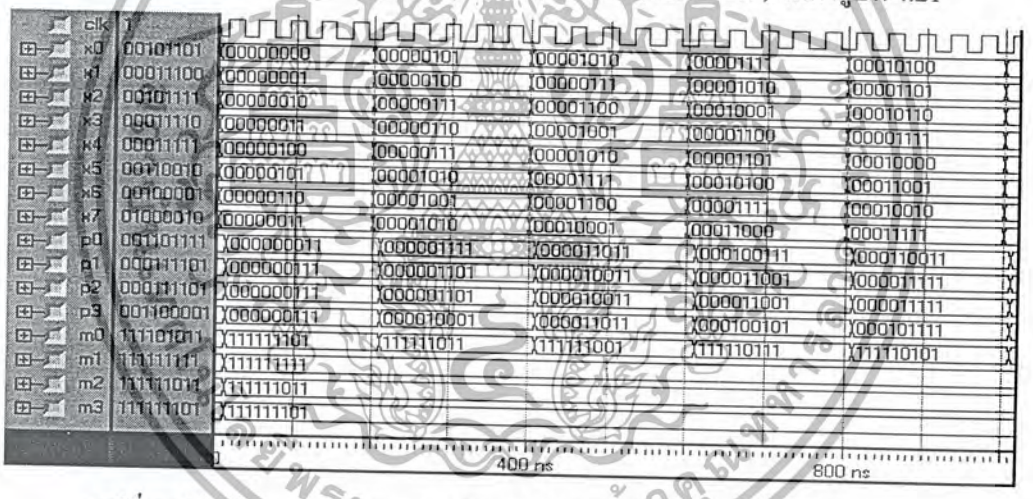
4.2.2.1 ส่วนของวงจรบวกและลบขนาด 8 บิท

เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่บวกและลบข้อมูลขนาด 8 บิท ซึ่งผลจากการบวกและลบข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 9 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.20



รูปที่ 4.20 แสดงสัญลักษณ์ของวงจรบวกและลบขนาด 8 บิท

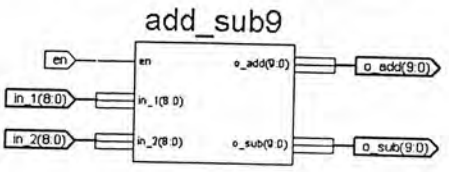
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.21



รูปที่ 4.21 แสดงผลการจำลองการทำงานของวงจรบวกและลบขนาด 8 บิท

4.2.2.2 ส่วนของวงจรบวกและลบขนาด 9 บิท

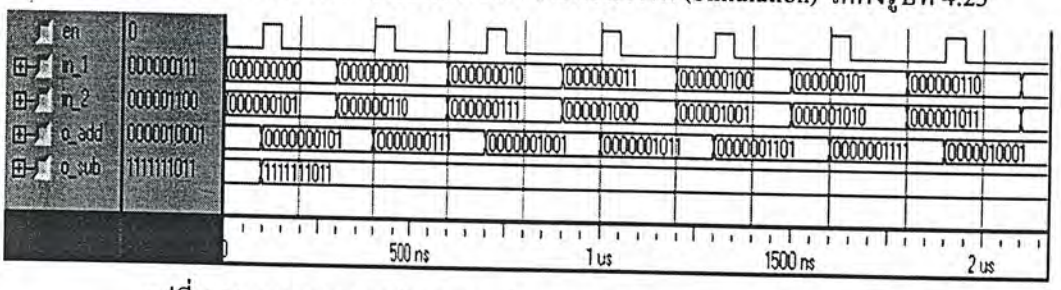
เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่บวกและลบข้อมูลขนาด 9 บิท ซึ่งผลจากการบวกและลบข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 10 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.22



รูปที่ 4.22 แสดงสัญลักษณ์ของวงจรบวกและลบขนาด 9 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.23



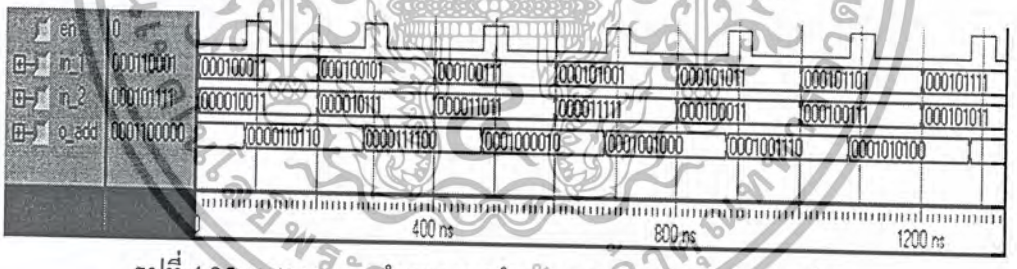
รูปที่ 4.23 แสดงผลการจำลองการทำงานของวงจรบวกและลบขนาด 9 บิท

4.2.2.3 ส่วนของวงจรบวกขนาด 9 บิท

เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่บวกข้อมูลขนาด 9 บิท ซึ่งผลจากการบวกข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 10 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.24

รูปที่ 4.24 แสดงสัญลักษณ์ของวงจรบวกขนาด 9 บิท

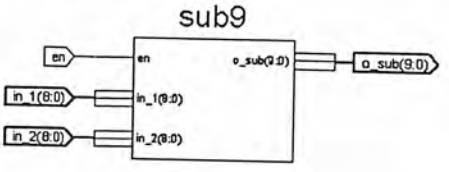
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.25



รูปที่ 4.25 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 9 บิท

4.2.2.4 ส่วนของวงจรลบขนาด 9 บิท

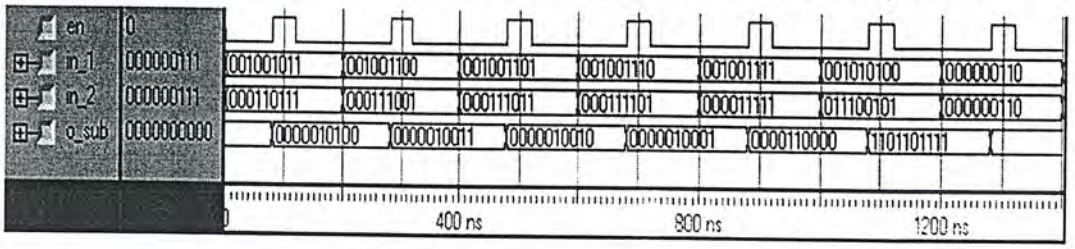
เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่ลบข้อมูลขนาด 9 บิท ซึ่งผลจากการลบข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 10 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.26



รูปที่ 4.26 แสดงสัญลักษณ์ของวงจรลบขนาด 9 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

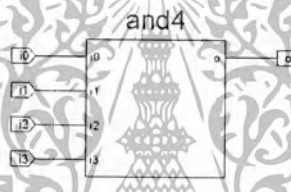
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.27



รูปที่ 4.27 แสดงผลการจำลองการทำงานของวงจรลบขนาด 9 บิท

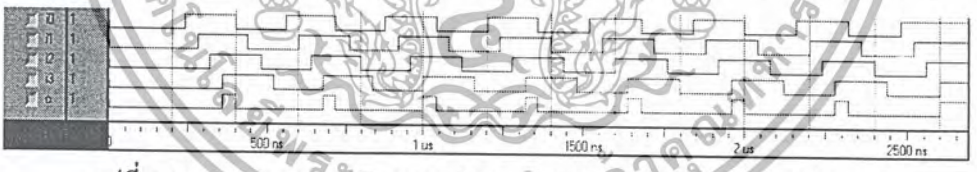
4.2.2.5 ส่วนของวงจรแอนด์ (AND) สัญญาณ 4 อินพุท

เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่เปรียบเทียบสัญญาณโหลดที่ออกมาจากวงจร DA_CORDIC_1 - DA_CORDIC_4 แต่ละตัว เพื่อส่งสัญญาณกลับไปให้ส่วนของวงจรควบคุมเมื่อวงจรDA_CORDIC ทำงานเสร็จพร้อมกันทั้งหมดแล้ว สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.28



รูปที่ 4.28 แสดงสัญลักษณ์ของวงจรแอนด์สัญญาณ 4 อินพุท

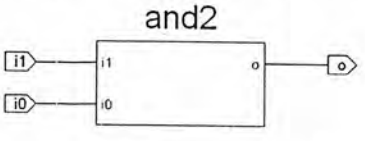
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.29



รูปที่ 4.29 แสดงผลการจำลองการทำงานของวงจรแอนด์สัญญาณ 4 อินพุท

4.2.2.6 ส่วนของวงจรแอนด์ (AND) สัญญาณ 2 อินพุท

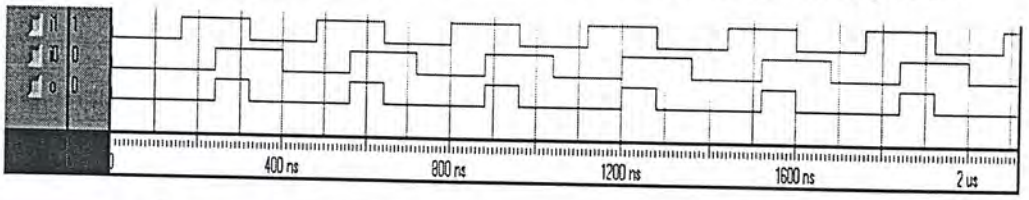
เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่เปรียบเทียบสัญญาณโหลดที่ออกมาจากวงจร DA_CORDIC_5- DA_CORDIC_6 แต่ละตัว เพื่อส่งสัญญาณกลับไปให้ส่วนของวงจรควบคุมเมื่อวงจร DA_CORDIC ทำงานเสร็จพร้อมกันทั้งหมดแล้ว สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.30



รูปที่ 4.30 แสดงสัญลักษณ์ของวงจรแอนด์สัญญาณ 2 อินพุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

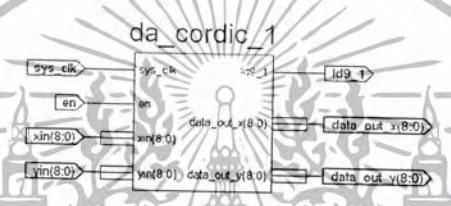
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.31



รูปที่ 4.31 แสดงผลการจำลองการทำงานของวงจรแอนด์สัญญาณ 2 อินพุท

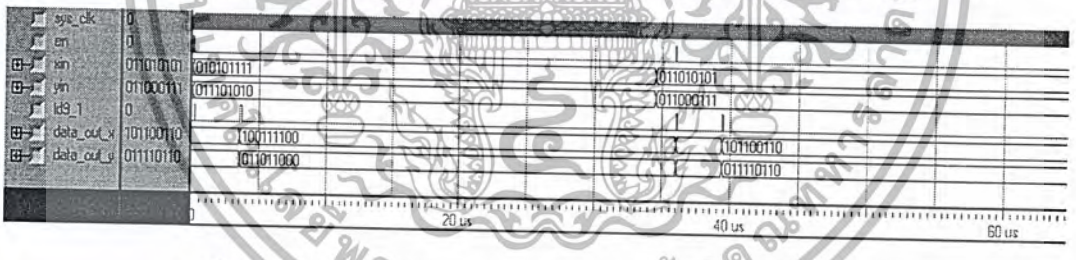
4.2.2.7 ส่วนของวงจร DA_CORDIC 1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 9 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 78.75 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.32



รูปที่ 4.32 แสดงสัญลักษณ์ของวงจร DA_CORDIC 1

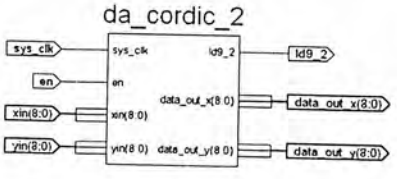
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.33



รูปที่ 4.33 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 1

4.2.2.8 ส่วนของวงจร DA_CORDIC 2

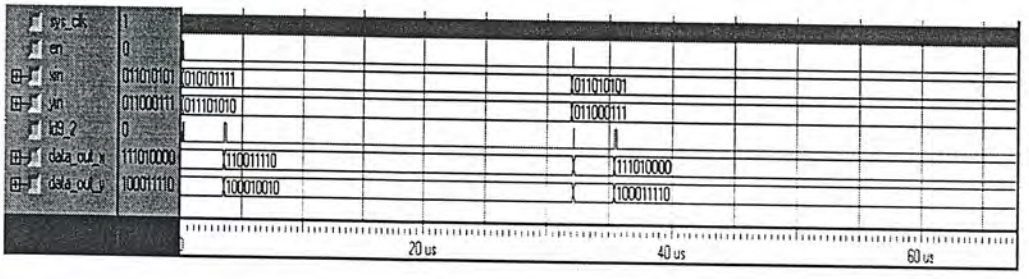
เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 9 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 56.25 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.34



รูปที่ 4.34 แสดงสัญลักษณ์ของวงจร DA_CORDIC 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

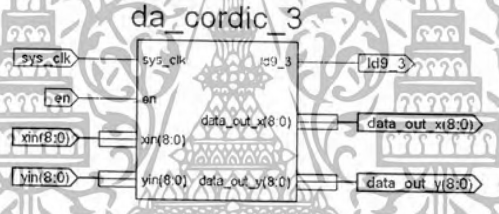
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.35



รูปที่ 4.35 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 2

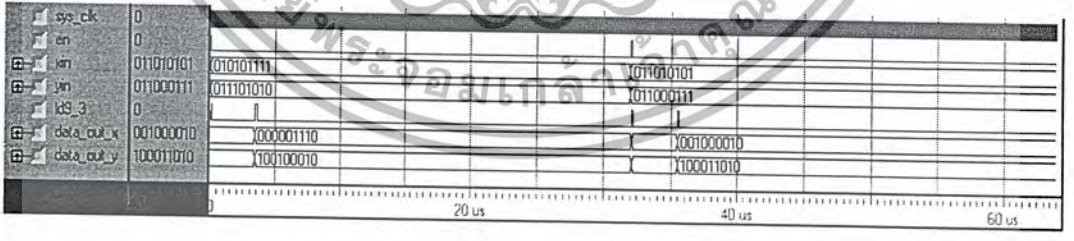
4.2.2.9 ส่วนของวงจร DA_CORDIC 3

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 9 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 33.75 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.36



รูปที่ 4.36 แสดงสัญลักษณ์ของวงจร DA_CORDIC 3

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.37

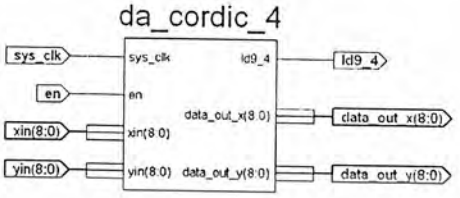


รูปที่ 4.37 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

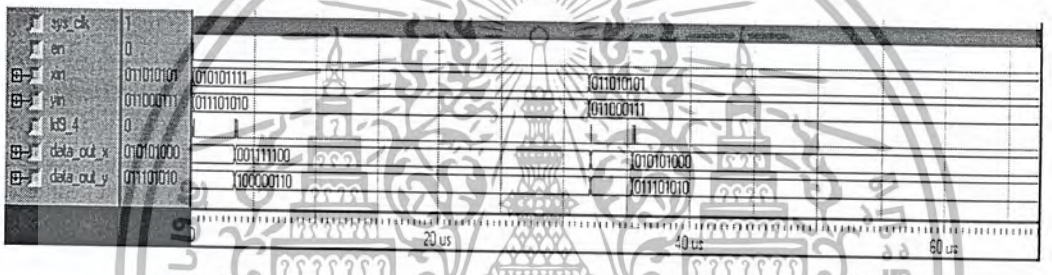
4.2.2.10 ส่วนของวงจร DA_CORDIC 4

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 9 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 11.25 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.38



รูปที่ 4.38 แสดงสัญลักษณ์ของวงจร DA_CORDIC 4

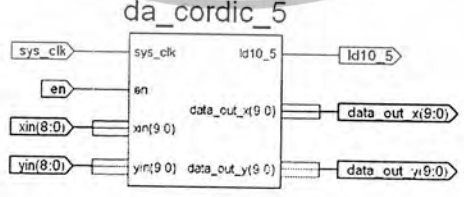
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.39



รูปที่ 4.39 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 4

4.2.2.11 ส่วนของวงจร DA_CORDIC 5

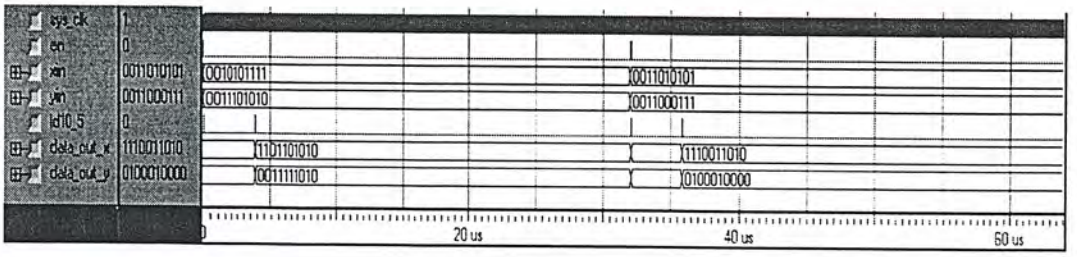
เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 10 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 67.5 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.40



รูปที่ 4.40 แสดงสัญลักษณ์ของวงจร DA_CORDIC 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

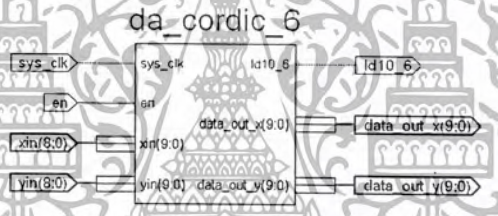
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.41



รูปที่ 4.41 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 5

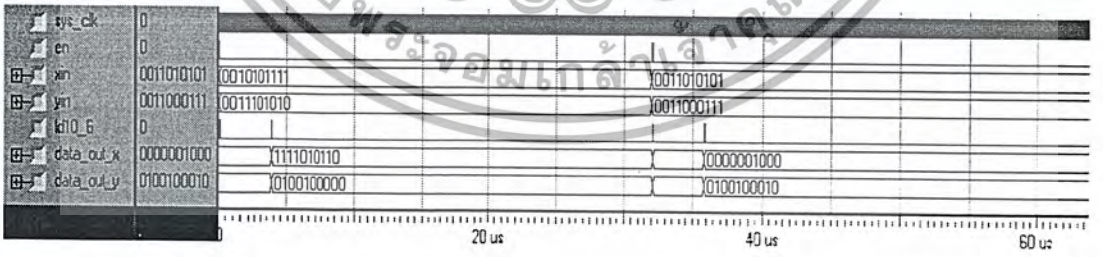
4.2.2.12 ส่วนของวงจร DA_CORDIC 6

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 10 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 45 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.42



รูปที่ 4.42 แสดงสัญลักษณ์ของวงจร DA_CORDIC 6

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.43

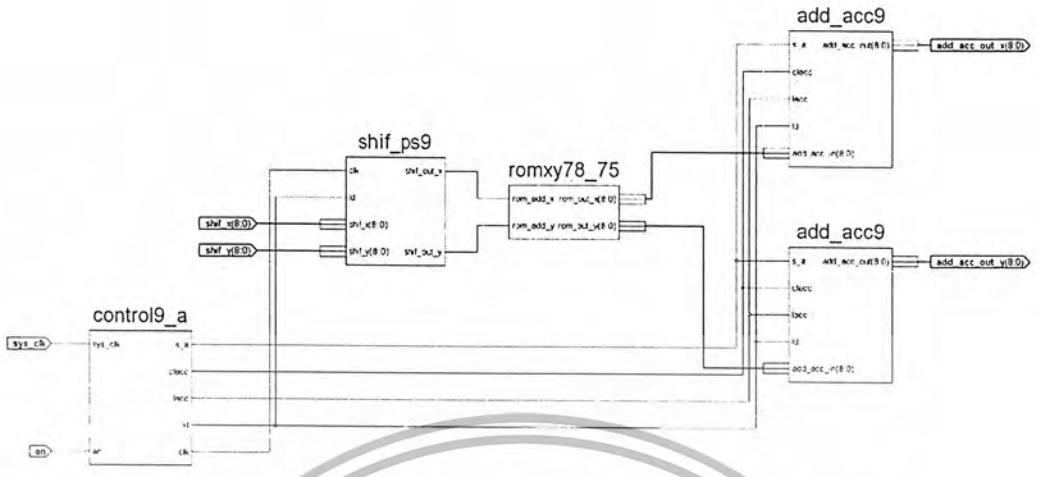


รูปที่ 4.43 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 6

ซึ่งภายในส่วนของวงจร DA_CORDIC จะประกอบด้วย ส่วนของวงจรเลื่อนบิท ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุมต่างๆ ส่วนของวงจรบวกสะสม และส่วนของวงจรควบคุมการทำงานของวงจร DA_CORDIC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของวงจร DA_CORDIC เป็นดังนี้



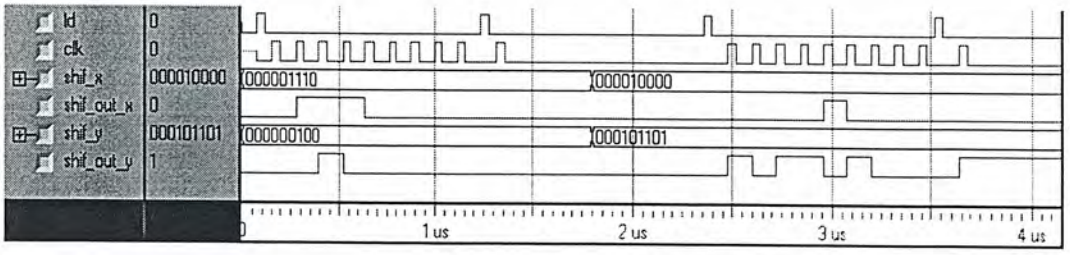
รูปที่ 4.44 แสดงองค์ประกอบของวงจร DA_CORDIC

4.2.2.13 ส่วนของวงจรเลื่อนบิท 9 บิท

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC 1 – DA_CORDIC 4 ทำหน้าที่เลื่อนบิทข้อมูลอินพุทออกมาทีละ 1 บิท เพื่อนำไปชี้ตำแหน่งค่าข้อมูลในรอม (ROM) ที่ถูกเก็บไว้เพื่อส่งข้อมูลออกจากรอม สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเลื่อนบิท 9 บิท ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.45

รูปที่ 4.45 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 9 บิท

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.46

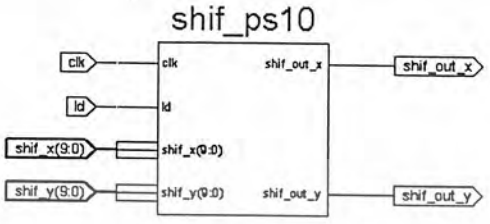


รูปที่ 4.46 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 9 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

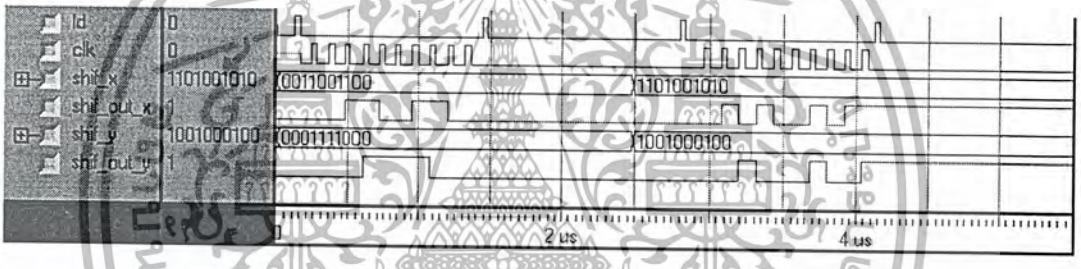
4.2.2.14 ส่วนของวงจรเลื่อนบิท 10 บิท

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC 5 – DA_CORDIC 6 ทำหน้าที่เลื่อนบิทข้อมูลอินพุทออกมาทีละ 1 บิท เพื่อนำไปชี้ตำแหน่งค่าข้อมูลในรอม (ROM) ที่ถูกเก็บไว้เพื่อส่งข้อมูลออกจากรอม สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเลื่อนบิท 10 บิท ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.47



รูปที่ 4.47 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 10 บิท

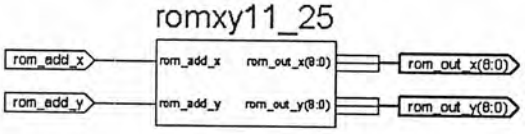
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.48



รูปที่ 4.48 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 10 บิท

4.2.2.15 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา

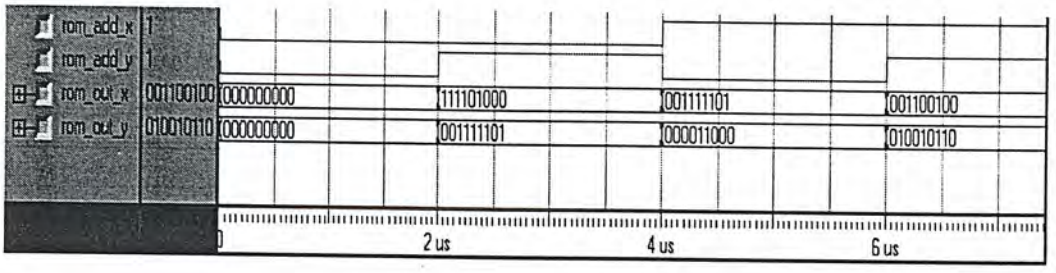
เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 11.25 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ได้สัญลักษณ์ (Symbol) ดังรูปที่4.49



รูปที่ 4.49 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

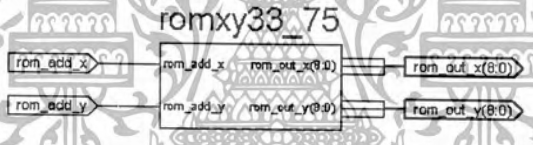
จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.50



รูปที่ 4.50 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา

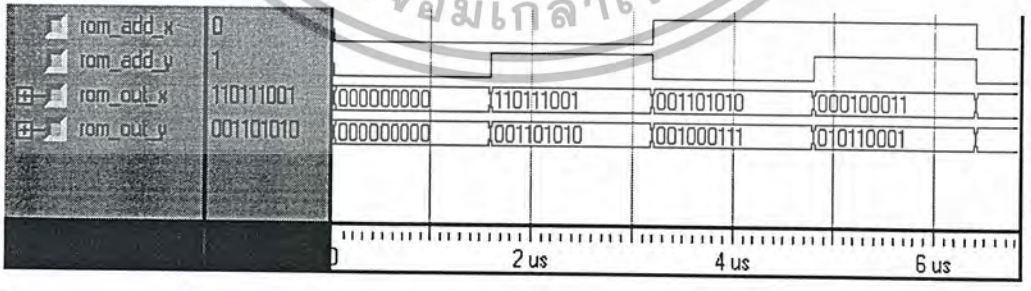
4.2.2.16 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 33.75 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.51



รูปที่ 4.51 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา

จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.52

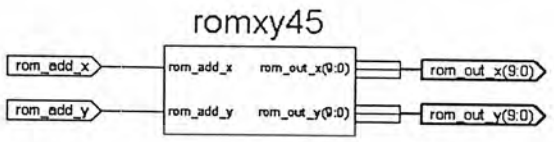


รูปที่ 4.52 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ ของมุม 33.75 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

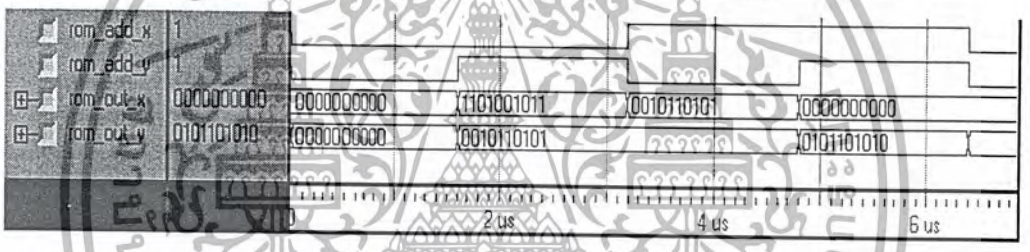
4.2.2.17 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ของมุม 45 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 45 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.53



รูปที่ 4.53 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 45 องศา

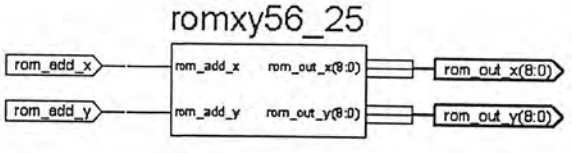
จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.54



รูปที่ 4.54 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ ของมุม 45 องศา

4.2.2.18 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ของมุม 56.25 องศา

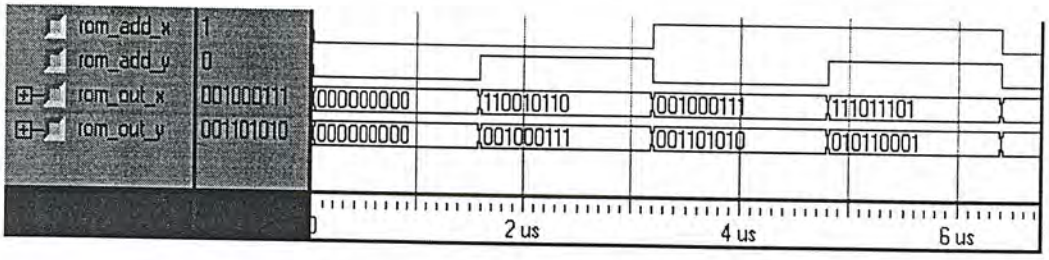
เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 56.25 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.55



รูปที่ 4.55 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.56



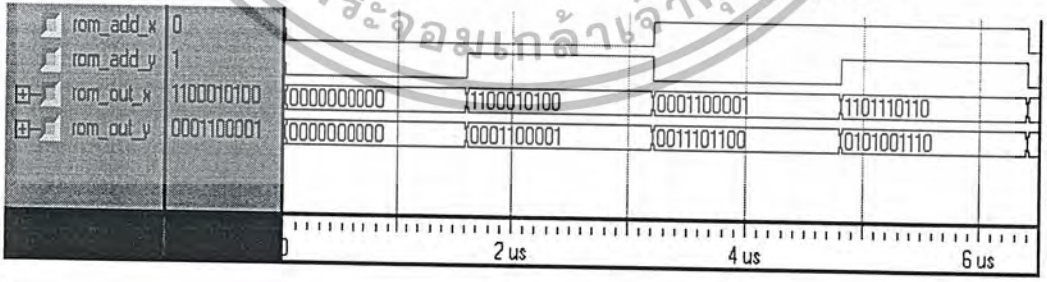
รูปที่ 4.56 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ ของมุม 56.25 องศา

4.2.2.19 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ของมุม 67.5 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 67.5 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.57

รูปที่ 4.57 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา

จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.58

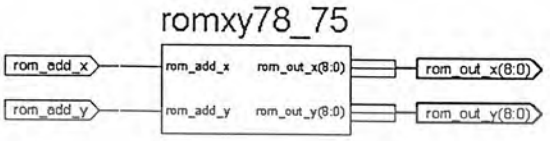


รูปที่ 4.58 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ ของมุม 67.5 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

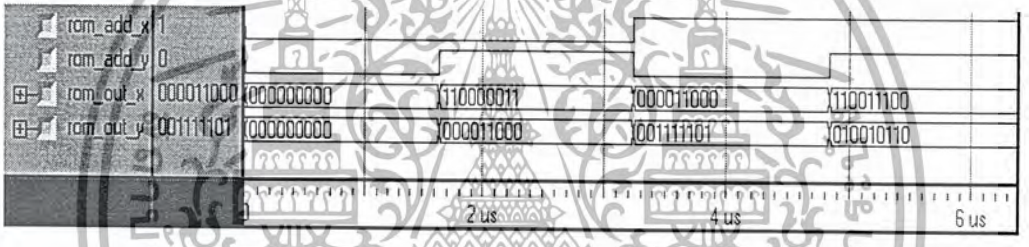
4.2.2.20 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ของมุม 78.75 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 78.75 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.59



รูปที่ 4.59 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา

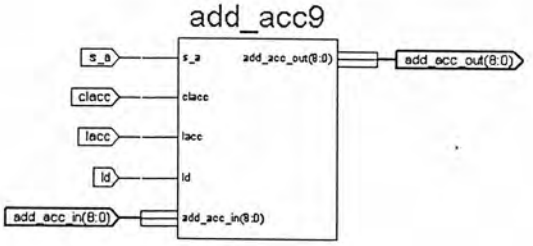
จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.60



รูปที่ 4.60 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ ของมุม 78.75 องศา

4.2.2.21 ส่วนของวงจรบวกสะสม 9 บิท

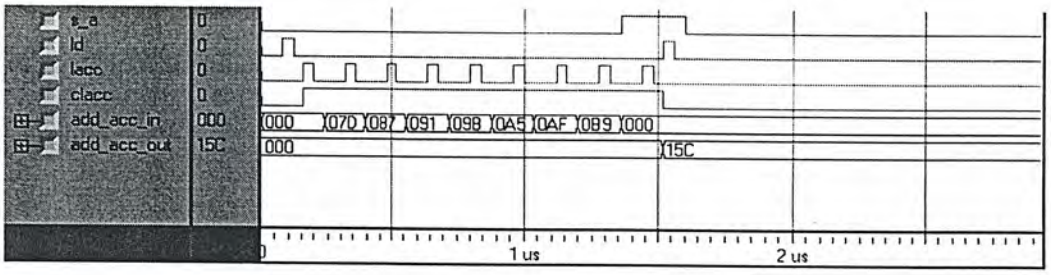
เป็นส่วนที่อยู่ในวงจร DA_CORDIC 1 – DA_CORDIC 4 โดยรับอินพุทมาจากส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ มาทำการบวกสะสมตามหลักการของโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรบวกสะสมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.61



รูปที่ 4.61 แสดงสัญลักษณ์ของวงจรบวกสะสม 9 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

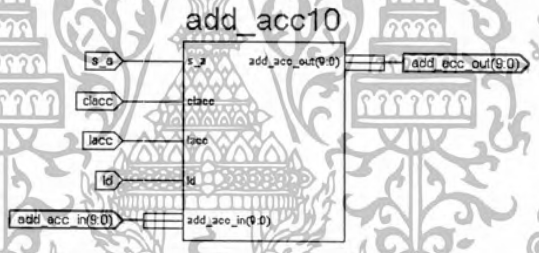
จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.62



รูปที่ 4.62 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 9 บิท

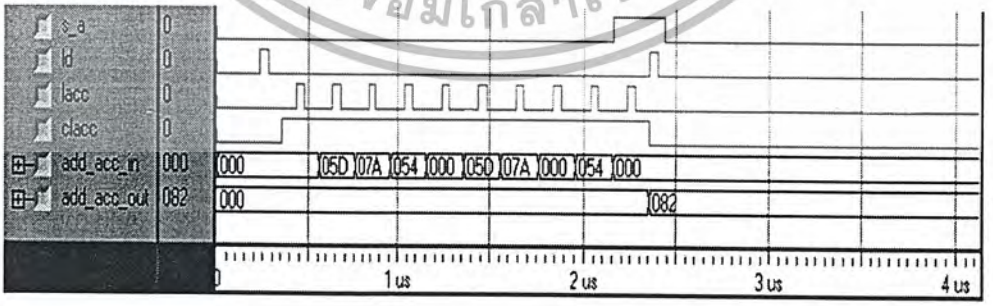
4.2.2.22 ส่วนของวงจรบวกสะสม10 บิท

เป็นส่วนที่อยู่ในวงจร DA_CORDIC 5 - DA_CORDIC 6 โดยรับอินพุทมาจากส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีต โคซายน์ มาทำการบวกสะสมตามหลักการของโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรบวกสะสมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.63



รูปที่ 4.63 แสดงสัญลักษณ์ของวงจรบวกสะสม 10 บิท

จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.64

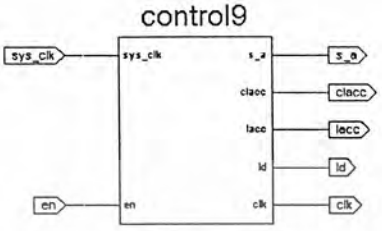


รูปที่ 4.64 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 10 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

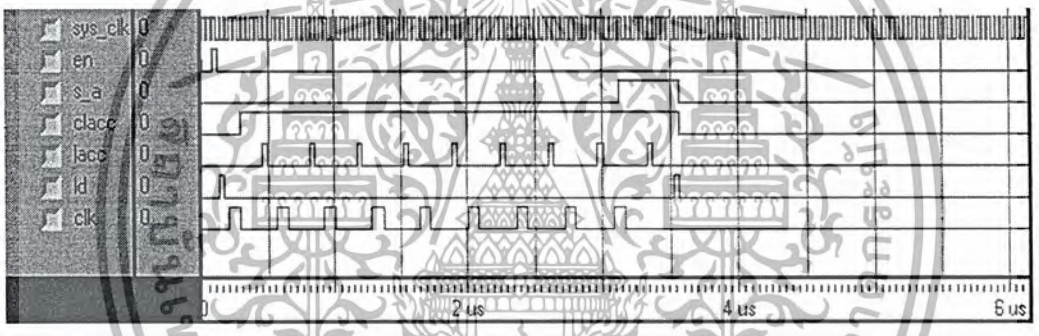
4.2.2.23 ส่วนของวงจรควบคุมภายในวงจร DA_CORDIC 1 - DA_CORDIC 4

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำหน้าที่สร้างสัญญาณควบคุมการทำงานของแต่ละส่วนภายในวงจร DA_CORDIC 1-DA_CORDIC 4 ให้มีความถูกต้อง สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมภายในวงจร DA_CORDIC 1 - DA_CORDIC 4 ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.65



รูปที่ 4.65 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร DA_CORDIC 1 – DA_CORDIC 4

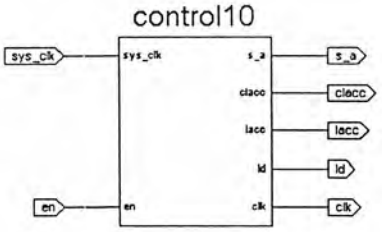
จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.66



รูปที่ 4.66 แสดงการจำลองการทำงานของวงจรควบคุมภายในวงจร DA_CORDIC 1 – DA_CORDIC 4

4.2.2.24 ส่วนของวงจรควบคุมภายในวงจร DA_CORDIC 5 - DA_CORDIC 6

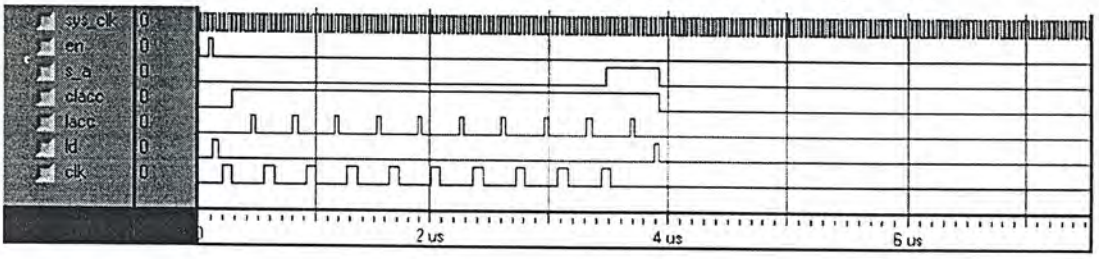
เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำหน้าที่สร้างสัญญาณควบคุมการทำงานของแต่ละส่วนภายในวงจร DA_CORDIC 5-DA_CORDIC 6 ให้มีความถูกต้อง สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมภายในวงจร DA_CORDIC 5-DA_CORDIC 6 ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.67



รูปที่ 4.67 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร DA_CORDIC 5 – DA_CORDIC 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

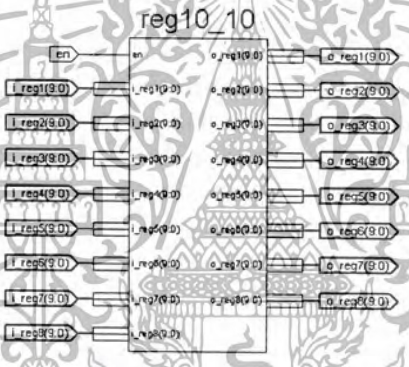
จากโปรแกรมสามารถแสดงผลการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.68



รูปที่ 4.68 แสดงการจำลองการทำงานของวงจรควบคุมภายในวงจร DA_CORDIC 5 – DA_CORDIC 6

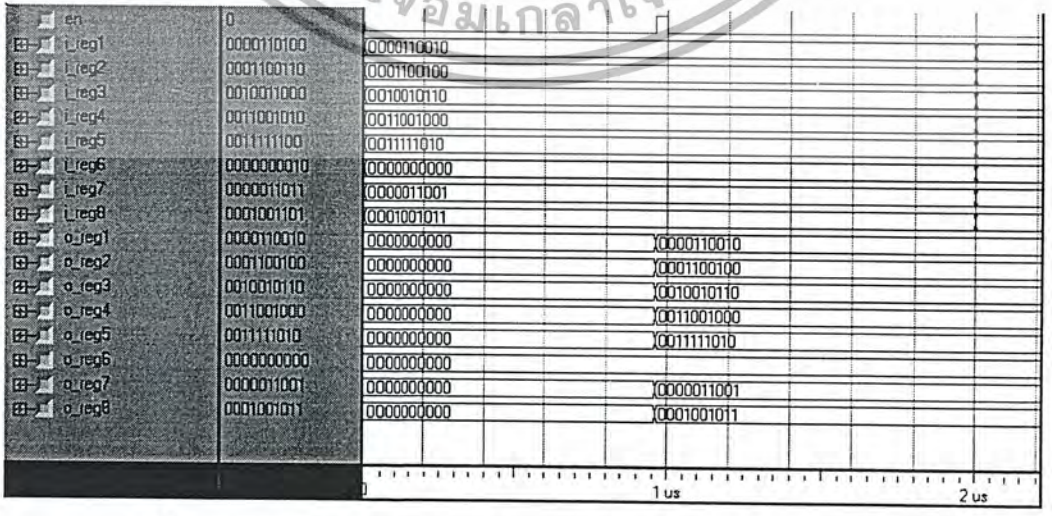
4.2.2.25 ส่วนของวงจรรีจีสเตอร์ขนาด 10 บิท (Register)

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่เก็บข้อมูลขนาด 10 บิท ที่ได้จากการแปลงดิสครีตโคซายน์ที่ส่งเอาท์พุทออกมาไม่พร้อมกัน โดยจะทำการเก็บข้อมูลขนาด 10 บิท ให้ครบ 8 ค่า แล้วทำการส่งข้อมูลเอาท์พุทออกไปพร้อม ๆ กันเพื่อเป็นอินพุทของวงจรส่วนต่อไป สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรรีจิสเตอร์ขนาด 10 บิท ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.69



รูปที่ 4.69 แสดงสัญลักษณ์ของวงจรรีจิสเตอร์ขนาด 10 บิท

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.70

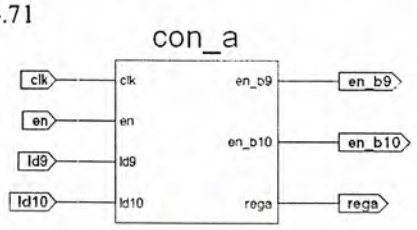


รูปที่ 4.70 แสดงผลการจำลองการทำงานของวงจรรีจิสเตอร์ขนาด 10 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

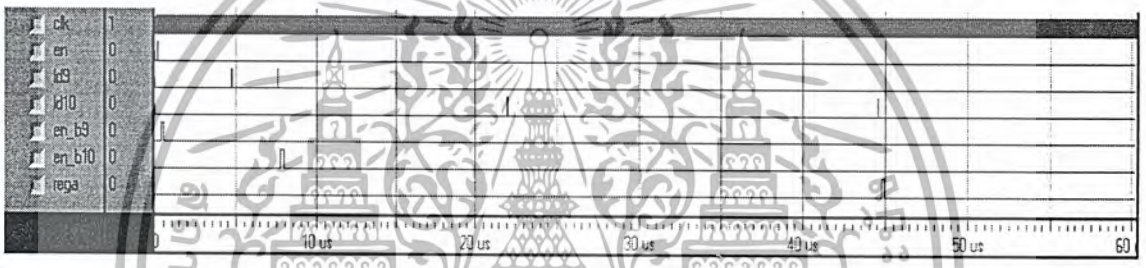
4.2.2.26 ส่วนของวงจรควบคุมการแปลงดิสครีตโคซายน์ ส่วนที่ 1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่สร้างสัญญาณควบคุมการทำงานของแต่ละวงจร ในส่วนการแปลงดิสครีตโคซายน์เพื่อให้สัญญาณเอาท์พุทที่ได้จากการแปลงดิสครีตโคซายน์มีความถูกต้อง สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.71



รูปที่ 4.71 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงดิสครีตโคซายน์

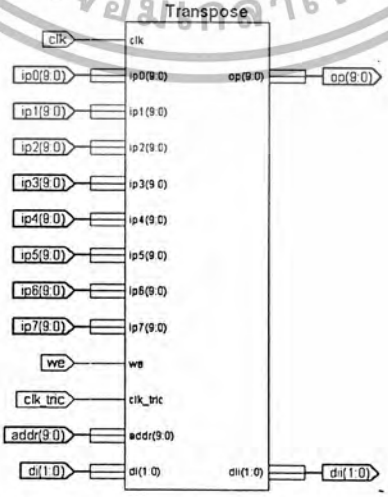
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.72



รูปที่ 4.72 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงดิสครีตโคซายน์

4.3 ส่วนของวงจรทรานสโพส (Transpose)

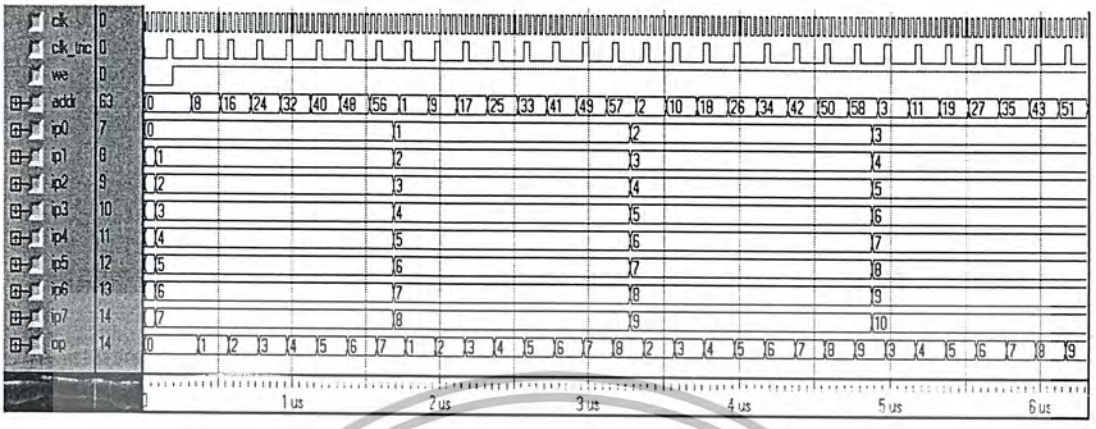
เป็นส่วนของวงจรที่ทำหน้าที่เปลี่ยนตำแหน่งของข้อมูลขนาด 10 บิท 64 ค่า ที่รับมาจากการแปลงดิสครีตโคซายน์แบบ 1 มิติ เพื่อนำไปเป็นอินพุทของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ในส่วนถัดไป โดยสามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรทรานสโพสได้สัญลักษณ์ ดังรูปที่ 4.73



รูปที่ 4.73 แสดงสัญลักษณ์ของวงจรทรานสโพส

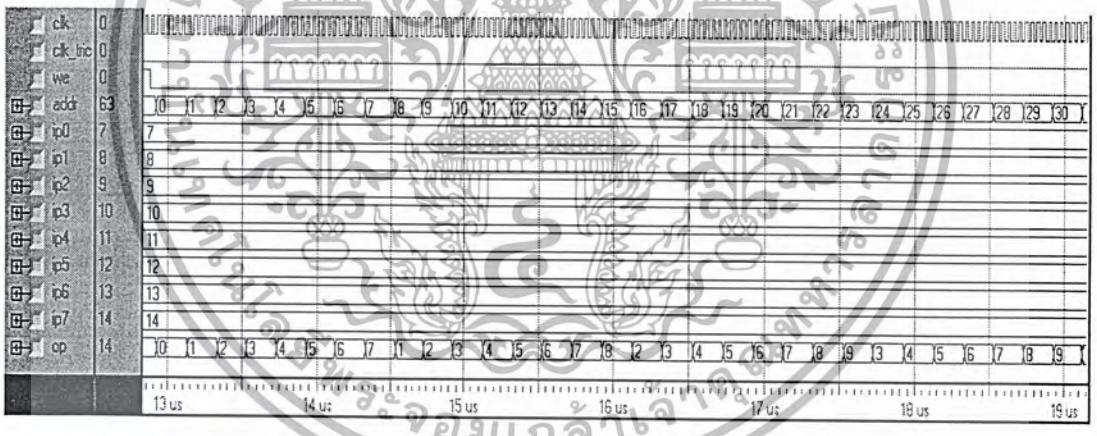
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมสามารถจำลองการทำงานของการเขียนข้อมูลลงในแรมเพื่อทำการทรานสโพสได้ ดังรูปที่ 4.74



รูปที่ 4.74 แสดงการจำลองการทำงานของวงจรทรานสโพสในการเขียนข้อมูลลงแรม

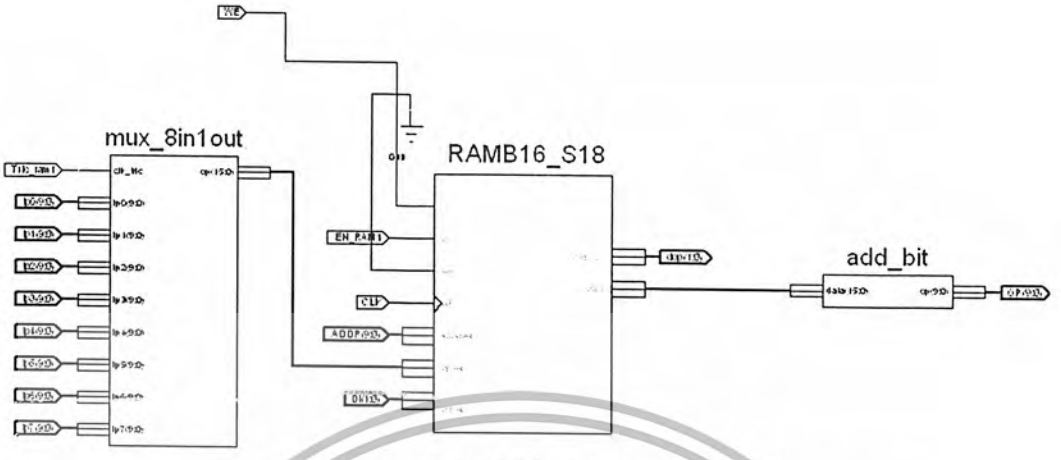
จากโปรแกรมสามารถจำลองการทำงานของการอ่านข้อมูลลงในแรมเพื่อทำการทรานสโพสได้ ดังรูปที่ 4.75



รูปที่ 4.75 แสดงการจำลองการทำงานของวงจรทรานสโพสในการอ่านข้อมูลออกจากแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

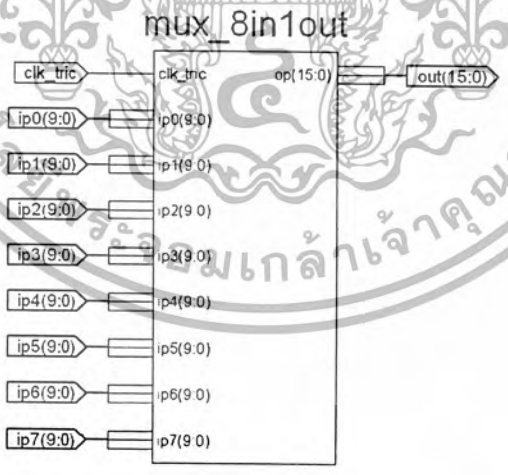
ส่วนประกอบภายในของวงจรทรานสโพส



รูปที่ 4.76 แสดงส่วนประกอบภายในของวงจรทรานสโพส

4.3.1 ส่วนของวงจรมัลติเพล็กซ์

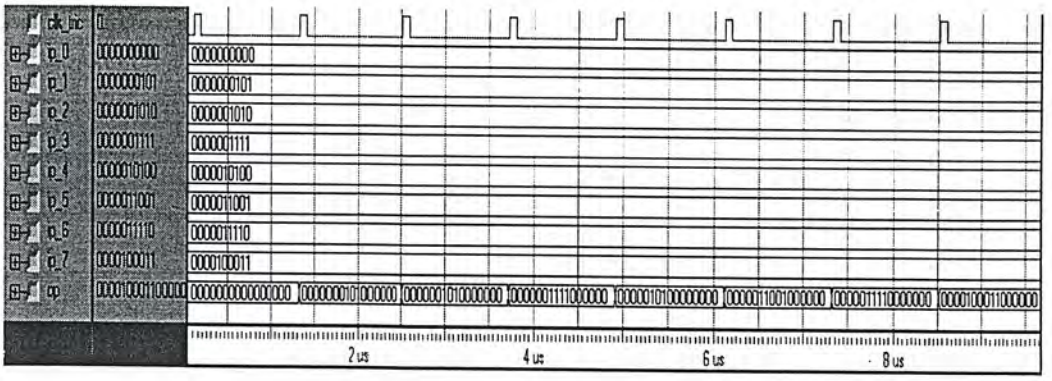
เป็นส่วนที่ทำหน้าที่ในการจัดสรรข้อมูลอินพุททั้ง 8 ค่า เพื่อให้ส่งออกเอาท์พุทเพียงช่องสัญญาณเดียวโดยที่ทำการแทรกบิทข้อมูลที่ LSB จำนวน 6 บิท เพื่อจะเก็บค่าที่ได้ในแรม สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรมัลติเพล็กซ์ ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.77



รูปที่ 4.77 แสดงสัญลักษณ์ของวงจรมัลติเพล็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

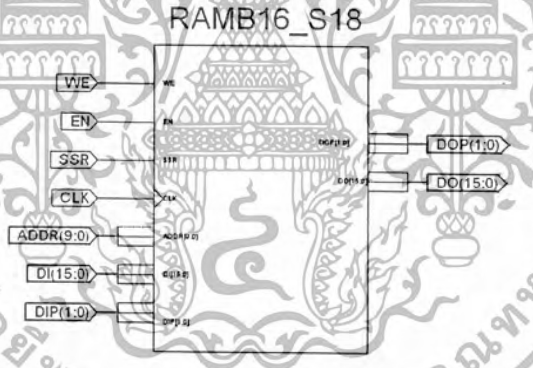
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.78



รูปที่ 4.78 แสดงผลการจำลองการทำงานของวงจรมัลติเพล็กซ์

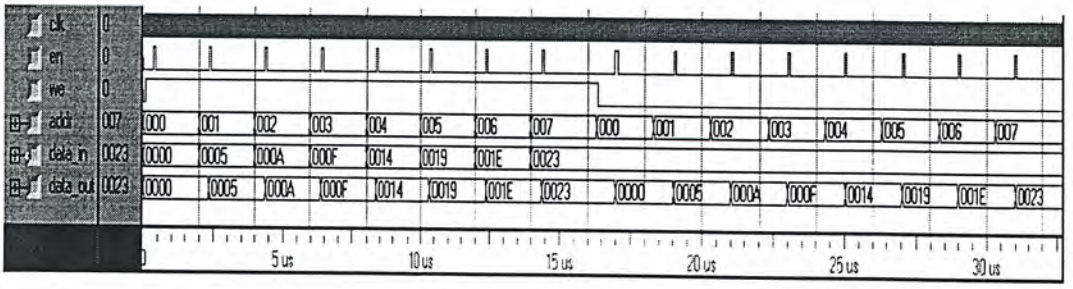
4.3.2 ส่วนของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล (RAM)

เป็นส่วนที่ทำหน้าที่ในการเก็บข้อมูลอินพุททั้ง 64 ค่า ที่รับมาจากวงจรมัลติเพล็กซ์มาเก็บในแรมแล้วจะทำการเรียกข้อมูลที่เป็นค่าทรานสโพสออกไปยังส่วนถัดไป โดยอาศัยการควบคุมการทำงานจากวงจรควบคุม โดยสามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูลได้สัญลักษณ์ ดังรูปที่ 4.79



รูปที่ 4.79 แสดงสัญลักษณ์ของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล

จากวงจร RAM ที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.80

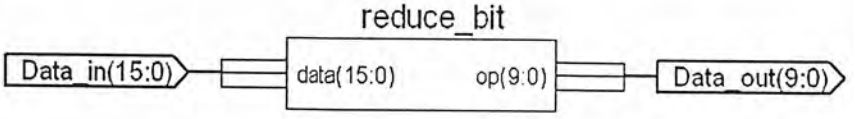


รูปที่ 4.80 แสดงผลการจำลองการทำงานของวงจรเก็บค่าข้อมูลและเรียกค่าข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 ส่วนของการลดจำนวนบิทข้อมูล

ทำหน้าที่ในการลดจำนวนบิทข้อมูลที่แทรกมาจาก 16 บิท ให้เหลือบิทข้อมูล 10 บิท เพื่อนำข้อมูลที่ได้ไปทำการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 ต่อไป โดยสามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรลดจำนวนบิทข้อมูล ดังรูปที่ 4.81



รูปที่ 4.81 แสดงสัญลักษณ์ของวงจรลดจำนวนบิทข้อมูล

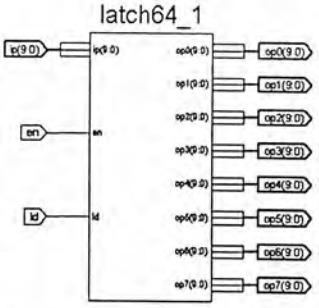
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.82

รูปที่ 4.82 แสดงผลการจำลองการทำงานของวงจรลดจำนวนบิทข้อมูล

4.4 ส่วนของวงจรคงค่าสัญญาณชั่วคราวและการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 หลังทำการทรานสโพส

4.4.1 ส่วนของวงจรคงค่าสัญญาณชั่วคราว 10 บิท (Latch)

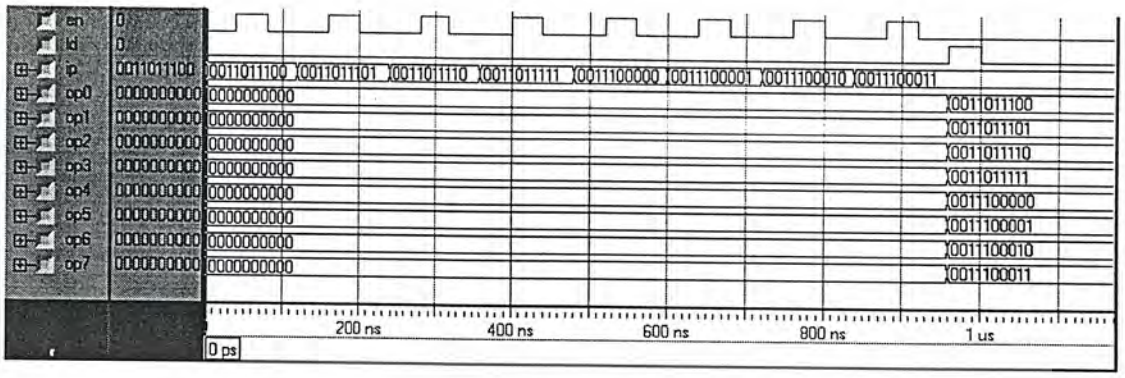
เป็นส่วนของวงจรที่เก็บค่าข้อมูลอินพุทไว้ชั่วคราว โดยจะทำการรับข้อมูลอินพุทที่ป้อนเข้ามานำมาเก็บไว้ชั่วคราวจนกว่าจะครบ 8 ค่า แต่ละค่า ๆ ละ 10 บิท แล้วทำการส่งค่าทั้ง 8 ค่าที่รับมาได้ส่งออกมาพร้อม ๆ กัน เพื่อส่งค่าไปเป็นอินพุทของส่วนการแปลงดิสครีตโคซายน์ต่อไป แล้วทำการรับค่าข้อมูลชุดใหม่เข้ามาให้ครบทั้ง 8 ค่าในรอบถัดไป จะทำงานซ้ำแบบนี้จนกระทั่งสัญญาณอินพุทครบทั้ง 64 ค่า สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรคงค่าสัญญาณชั่วคราวขนาด 10 บิทได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.83



รูปที่ 4.83 แสดงสัญลักษณ์ของวงจรคงค่าสัญญาณชั่วคราว 10 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

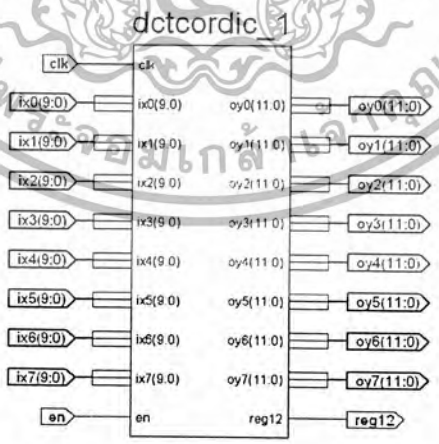
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.84



รูปที่ 4.84 แสดงผลการจำลองการทำงานของวงจรคงค่าสัญญาณชั่วคราว

4.4.2 ส่วนของวงจรการแปลงดิสครีตโคซายน์ทรานสฟอร์มแบบ 1 มิติ ส่วนที่ 2

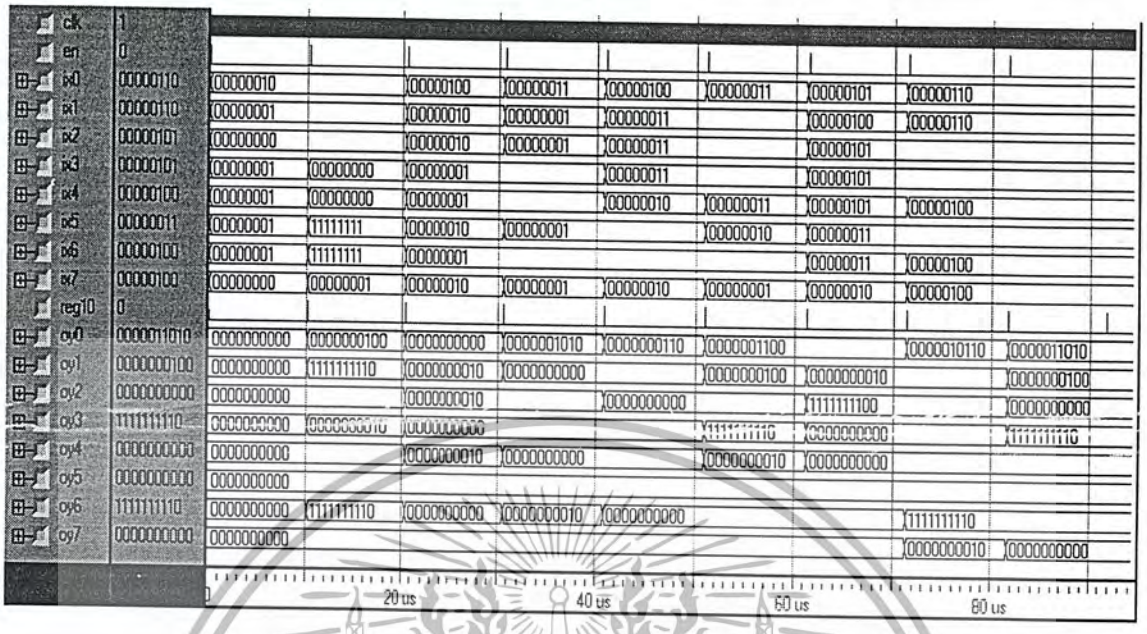
เป็นส่วนของวงจรการแปลงดิสครีตโคซายน์ ทำหน้าที่ในการคำนวณค่าการแปลงดิสครีต-โคซายน์แบบ 1 มิติ ของข้อมูลที่รับข้อมูลอินพุทมาจากวงจรส่วนของวงจรคงค่าสัญญาณชั่วคราว ขนาด 10 บิท 8 ค่า ซึ่งภายในส่วนของวงจรการแปลงดิสครีตโคซายน์ประกอบไปด้วยส่วนของวงจรบวกและลบ 10 บิท ส่วนของวงจรบวกและลบ 11 บิท ส่วนของวงจรบวก 11 บิท ส่วนของวงจรลบ 11 บิท ส่วนของวงจรคอร์ดิก ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ส่วนของวงจรรีจีสเตอร์ขนาด 12 บิท และ ส่วนของวงจรคอนโทรล (controller) ซึ่งส่วนของวงจรคอนโทรลนี้เป็นส่วนที่สำคัญมาก สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.85



รูปที่ 4.85 แสดงสัญลักษณ์ของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

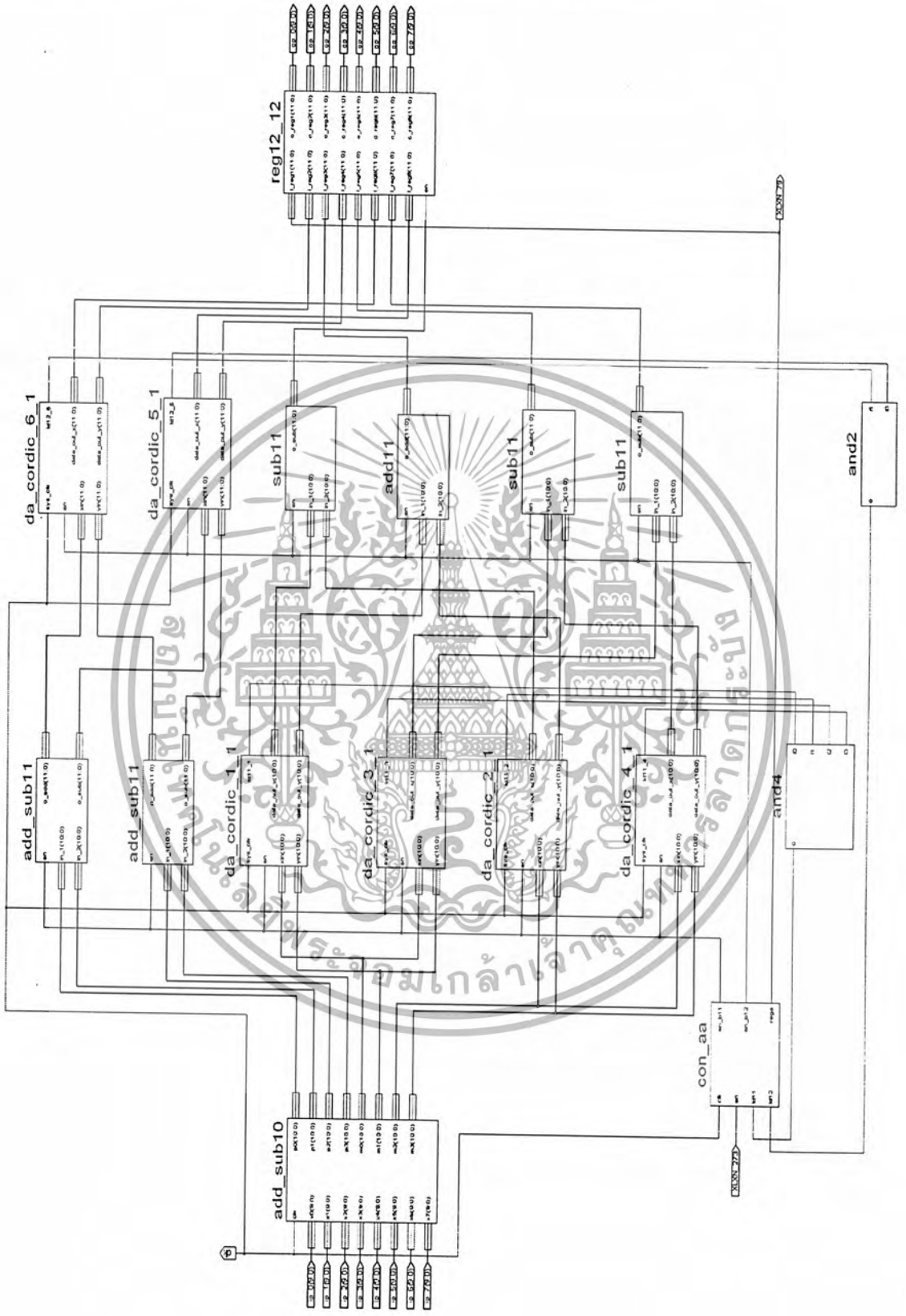
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.86



รูปที่ 4.86 แสดงผลการจำลองการทำงานของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2 ดังนี้

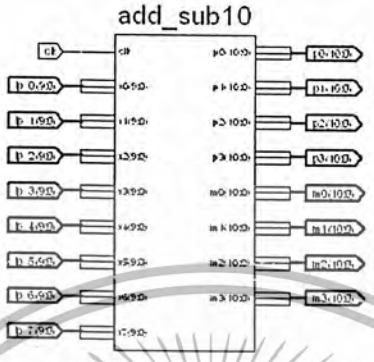


รูปที่ 4.87 แสดงวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่วนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

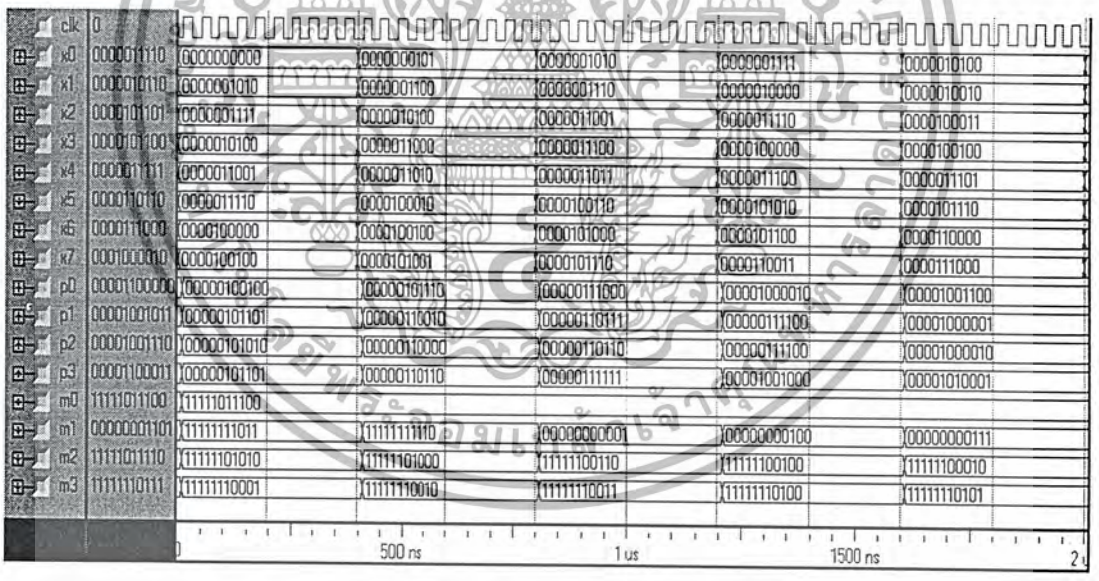
4.4.2.1 ส่วนของวงจรบวกและลบขนาด 10 บิท

เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่บวกและลบข้อมูลขนาด 10 บิท ซึ่งผลจากการบวกและลบข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 11 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.88



รูปที่ 4.88 แสดงสัญลักษณ์ของวงจรบวกและลบขนาด 10 บิท

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.89

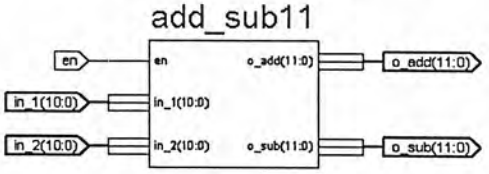


รูปที่ 4.89 แสดงผลการจำลองการทำงานของวงจรบวกและลบขนาด 10 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

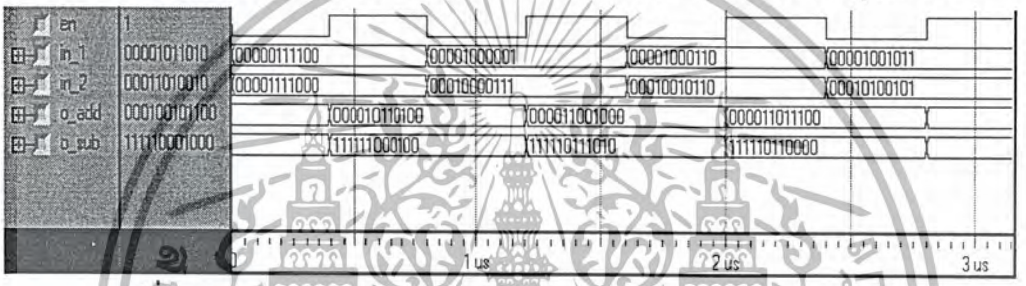
4.4.2.2 ส่วนของวงจรบวกและลบขนาด 11 บิท

เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่บวกและลบข้อมูลขนาด 11 บิท ซึ่งผลจากการบวกและลบข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 12 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.90



รูปที่ 4.90 แสดงสัญลักษณ์ของวงจรบวกและลบขนาด 11 บิท

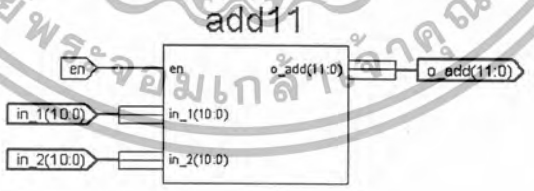
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.91



รูปที่ 4.91 แสดงผลการจำลองการทำงานของวงจรบวกและลบขนาด 11 บิท

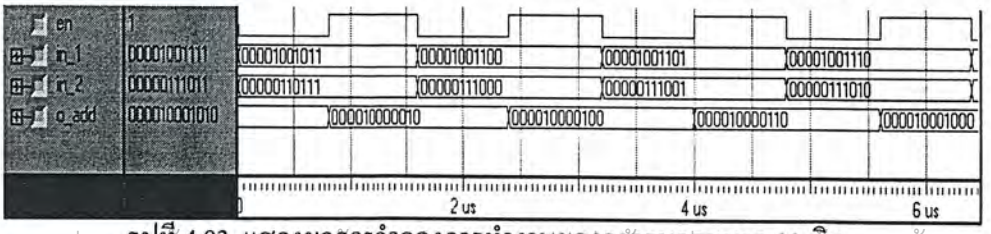
4.4.2.3 ส่วนของวงจรบวกขนาด 11 บิท

เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่บวกข้อมูลขนาด 11 บิท ซึ่งผลจากการบวกข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 12 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.92



รูปที่ 4.92 แสดงสัญลักษณ์ของวงจรบวกขนาด 11 บิท

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.93

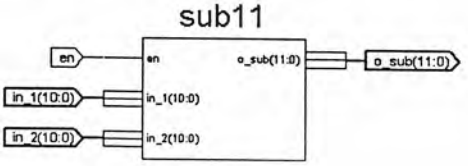


รูปที่ 4.93 แสดงผลการจำลองการทำงานของวงจรบวกขนาด 11 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

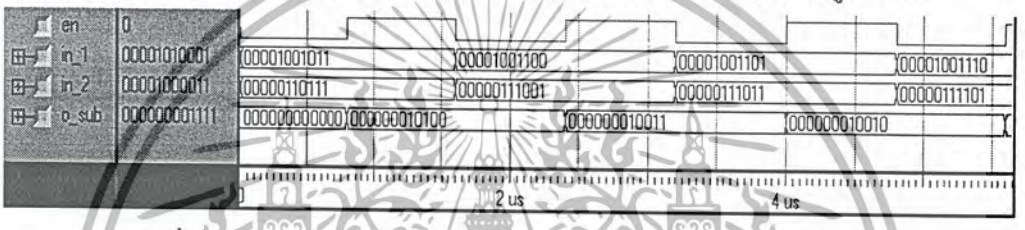
4.4.2.4 ส่วนของวงจรลบขนาด 11 บิท

เป็นส่วนที่อยู่ภายในวงจรการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทำหน้าที่ลบข้อมูลขนาด 11 บิท ซึ่งผลจากการลบข้อมูลนี้จะได้ผลลัพธ์เป็นข้อมูลขนาด 12 บิท สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.94



รูปที่ 4.94 แสดงสัญลักษณ์ของวงจรลบขนาด 11 บิท

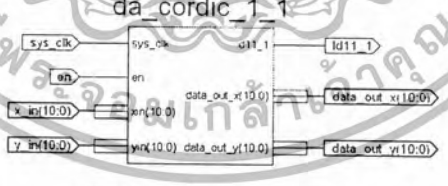
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.95



รูปที่ 4.95 แสดงผลการจำลองการทำงานของวงจรลบขนาด 11 บิท

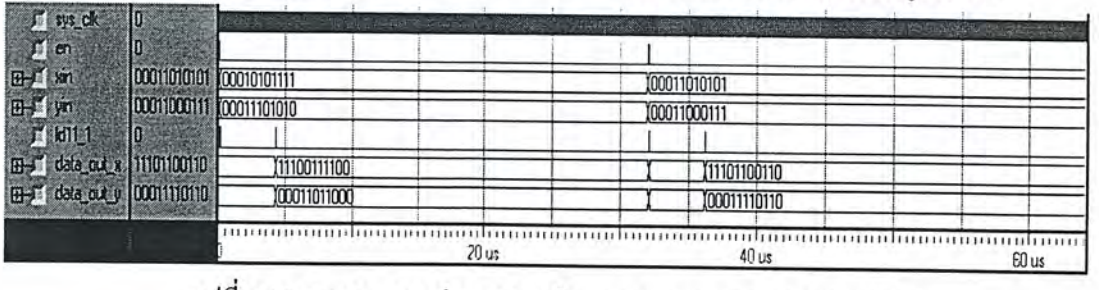
4.4.2.5 ส่วนของวงจร DA_CORDIC 1_1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 11 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 78.75 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.96



รูปที่ 4.96 แสดงสัญลักษณ์ของวงจร DA_CORDIC 1_1

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.97

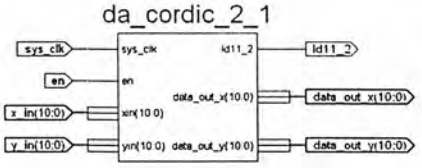


รูปที่ 4.97 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 1_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

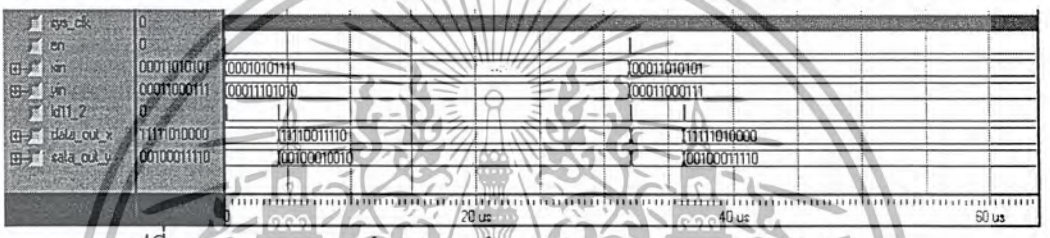
4.4.2.6 ส่วนของวงจร DA_CORDIC 2_1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 11 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 56.25 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.98



รูปที่ 4.98 แสดงสัญลักษณ์ของวงจร DA_CORDIC 2_1

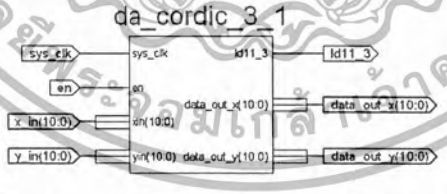
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.99



รูปที่ 4.99 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 2_1

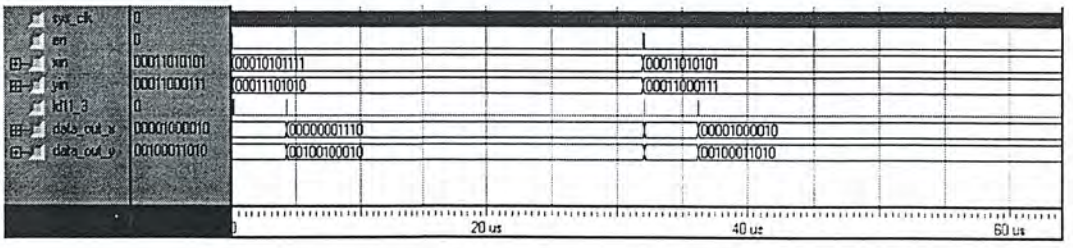
4.4.2.7 ส่วนของวงจร DA_CORDIC 3_1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 11 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 33.75 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.100



รูปที่ 4.100 แสดงสัญลักษณ์ของวงจร DA_CORDIC 3_1

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.101

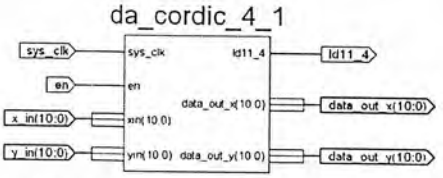


รูปที่ 4.101 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 3_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

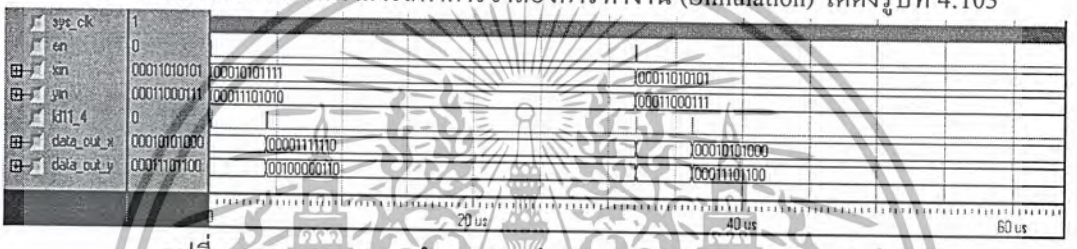
4.4.2.8 ส่วนของวงจร DA_CORDIC 4_1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 11 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 11.25 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.102



รูปที่ 4.102 แสดงสัญลักษณ์ของวงจร DA_CORDIC 4_1

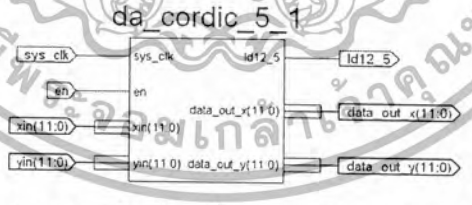
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.103



รูปที่ 4.103 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 4_1

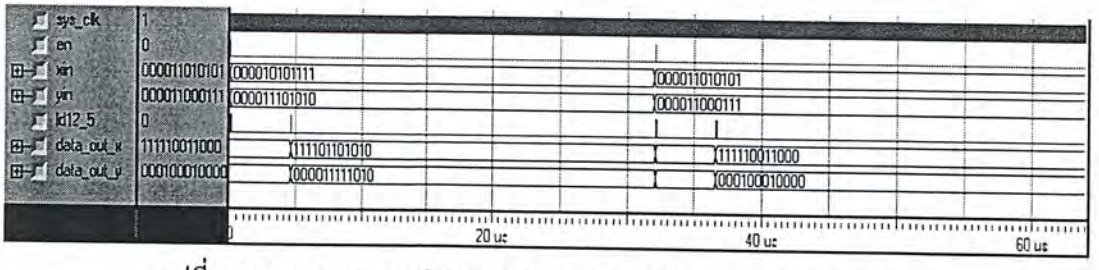
4.4.2.9 ส่วนของวงจร DA_CORDIC 5_1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 12 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 67.5 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.104



รูปที่ 4.104 แสดงสัญลักษณ์ของวงจร DA_CORDIC 5_1

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.105

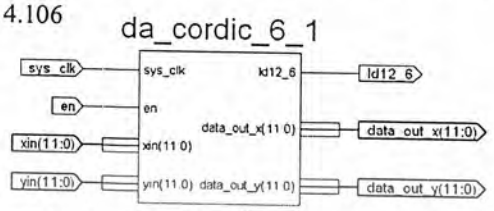


รูปที่ 4.105 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 5_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

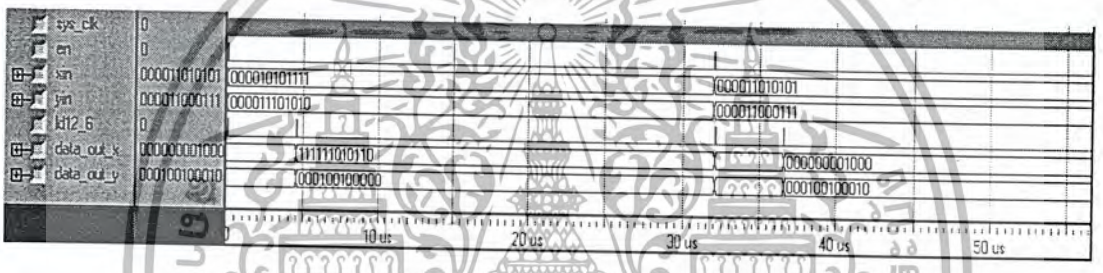
4.4.2.10 ส่วนของวงจร DA_CORDIC 6_1

เป็นวงจรที่อยู่ในส่วนของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ซึ่งใช้โครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) เข้ามาช่วยในการคำนวณ ทำหน้าที่ในการคูณข้อมูลขนาด 12 บิท กับค่าคงที่สัมประสิทธิ์ของมุม 45 องศา สามารถเขียนโปรแกรมที่สังเคราะห์เป็นอุปกรณ์ที่มีสัญลักษณ์ (Symbol) ดังรูปที่ 4.106



รูปที่ 4.106 แสดงสัญลักษณ์ของวงจร DA_CORDIC 6_1

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.107

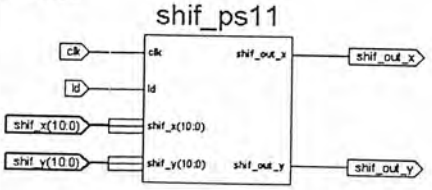


รูปที่ 4.107 แสดงผลการจำลองการทำงานของวงจร DA_CORDIC 6_1

ซึ่งภายในส่วนของวงจร DA_CORDIC จะประกอบด้วย ส่วนของวงจรเลื่อนบิท ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุมต่าง ๆ ส่วนของวงจรบวกสะสม และส่วนของวงจรควบคุมการทำงานของวงจร DA_CORDIC

4.4.2.11 ส่วนของวงจรเลื่อนบิท 11 บิท

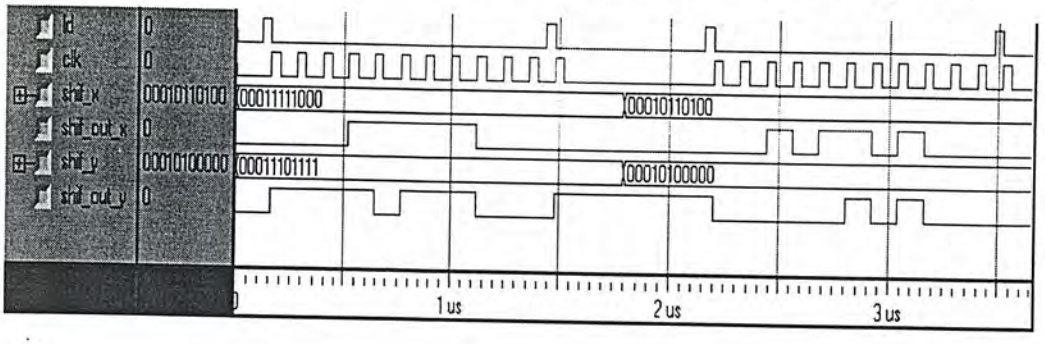
เป็นวงจรที่อยู่ภายในส่วนของวงจร DA_CORDIC 1_1 – DA_CORDIC 4_1 ทำหน้าที่เลื่อนบิทข้อมูลอินพุทออกมาทีละ 1 บิท เพื่อนำไปชี้ตำแหน่งค่าข้อมูลในรอม (ROM) ที่ถูกเก็บไว้เพื่อส่งข้อมูลออกจากรอม (ROM) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเลื่อนบิท 11 บิท ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.108



รูปที่ 4.108 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 11 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

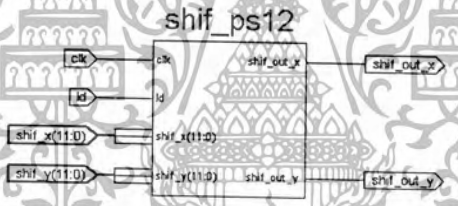
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.109



รูปที่ 4.109 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 11 บิท

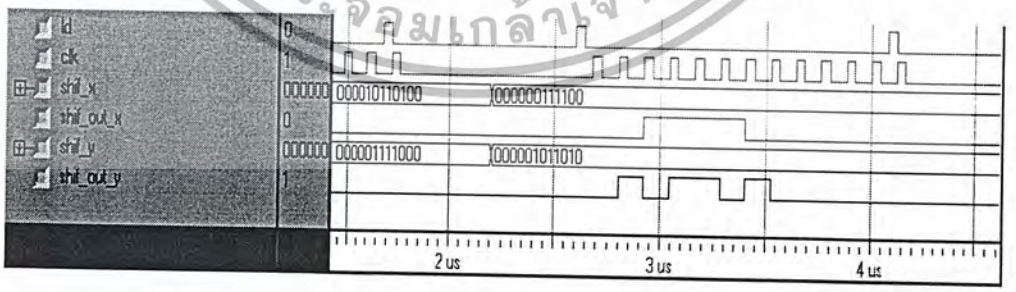
4.4.2.12 ส่วนของวงจรเลื่อนบิท 12 บิท

เป็นวงจรที่อยู่ภายในส่วนของวงจร DA_CORDIC 5_1 – DA_CORDIC 6_1 ทำหน้าที่เลื่อนบิทข้อมูลอินพุทออกมาทีละ 1 บิท เพื่อนำไปชี้ตำแหน่งค่าข้อมูลในรอม (ROM) ที่ถูกเก็บไว้เพื่อส่งข้อมูลออกจากรอม สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเลื่อนบิท 12 บิท ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.110



รูปที่ 4.110 แสดงสัญลักษณ์ของวงจรเลื่อนบิท 12 บิท

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.111

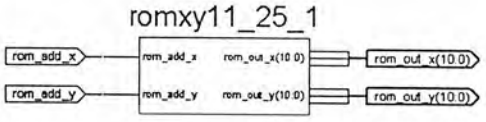


รูปที่ 4.111 แสดงผลการจำลองการทำงานของวงจรเลื่อนบิท 12 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

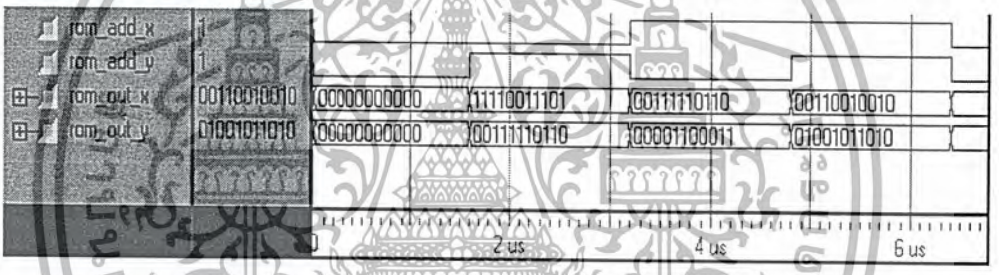
4.4.2.13 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 11.25 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.112



รูปที่ 4.112 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา

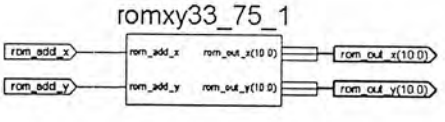
จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.113



รูปที่ 4.113 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 11.25 องศา

4.4.2.14 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา

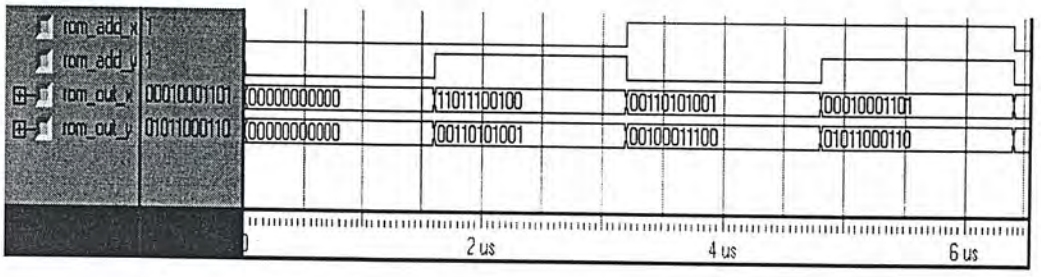
เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 33.75 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.114



รูปที่ 4.114 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

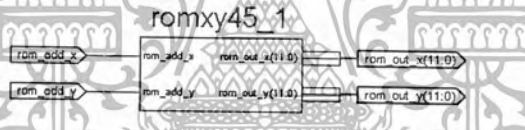
จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.115



รูปที่ 4.115 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 33.75 องศา

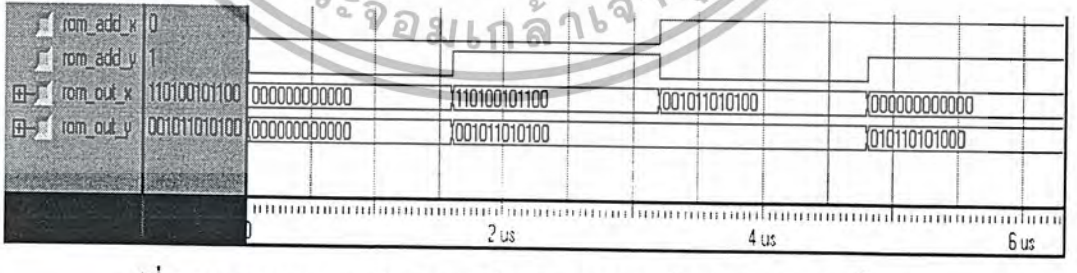
4.4.2.15 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 45 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 45 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.116



รูปที่ 4.116 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 45 องศา

จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.117

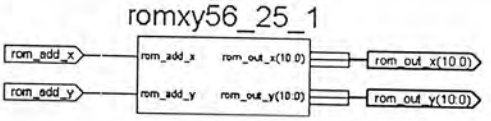


รูปที่ 4.117 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 45 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

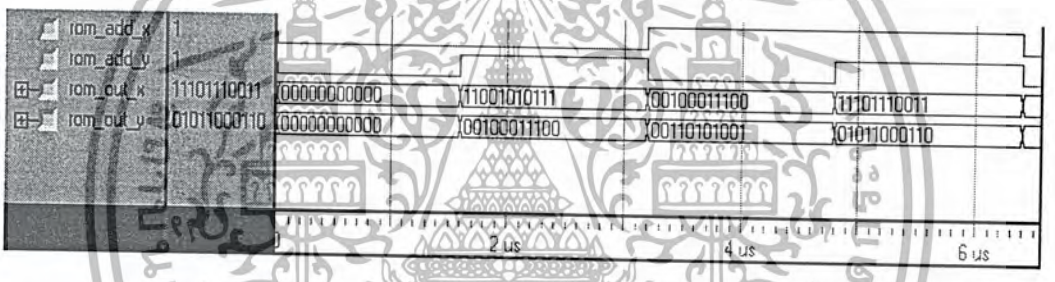
4.4.2.16 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 56.25 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.118



รูปที่ 4.118 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา

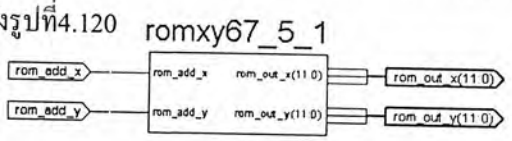
จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.119



รูปที่ 4.119 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 56.25 องศา

4.4.2.17 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา

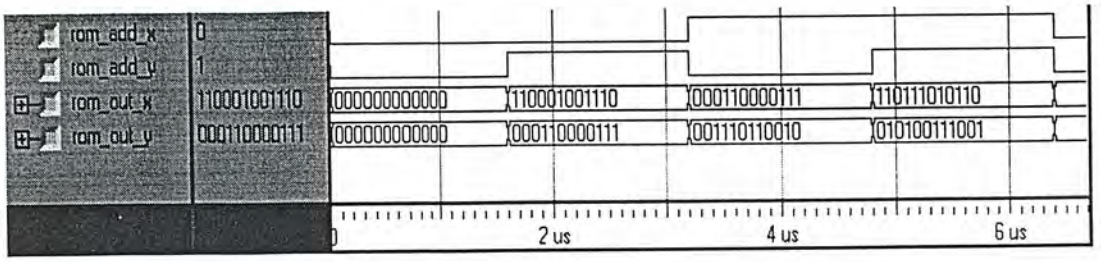
เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 67.5 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ได้สัญลักษณ์ (Symbol) ดังรูปที่4.120



รูปที่ 4.120 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.121



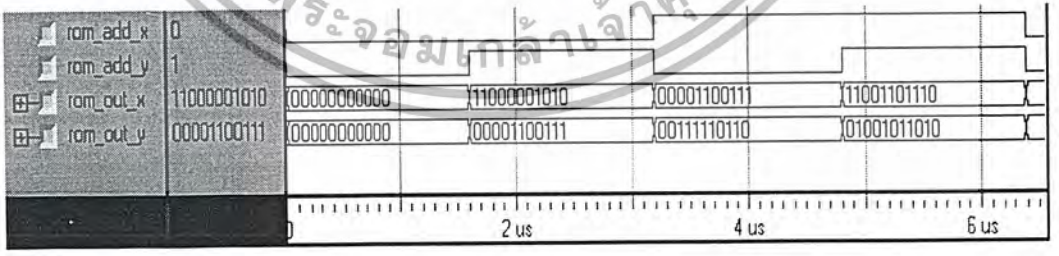
รูปที่ 4.121 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 67.5 องศา

4.4.2.18 ส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา

เป็นวงจรที่อยู่ในส่วนของวงจร DA_CORDIC ทำการเก็บค่าสัมประสิทธิ์ที่หาได้จากหลักการของคอร์ดิกอัลกอริทึมของมุม 78.75 องศาและส่งข้อมูลออกตามอินพุทที่ชี้ตำแหน่งข้อมูลที่ต้องการ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.122

รูปที่ 4.122 แสดงสัญลักษณ์ของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา

จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.123

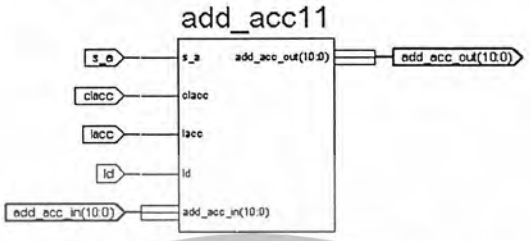


รูปที่ 4.123 แสดงผลการจำลองการทำงานของวงจรเก็บค่าสัมประสิทธิ์
การแปลงดิสครีตโคซายน์ของมุม 78.75 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

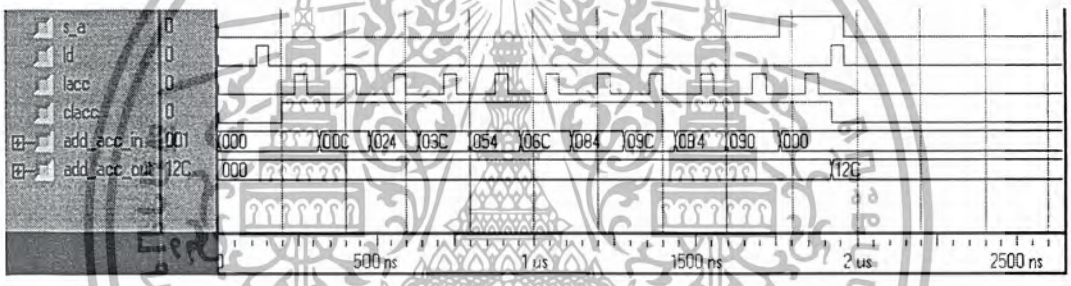
4.4.2.19 ส่วนของวงจรบวกสะสม 11 บิท

เป็นส่วนที่อยู่ใน DA_CORDIC 1_1 - DA_CORDIC 4_1 โดยรับอินพุทมาจากส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ มาทำการบวกสะสมตามหลักการของโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรบวกสะสม ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.124



รูปที่ 4.124 แสดงสัญลักษณ์ของวงจรบวกสะสม 11 บิท

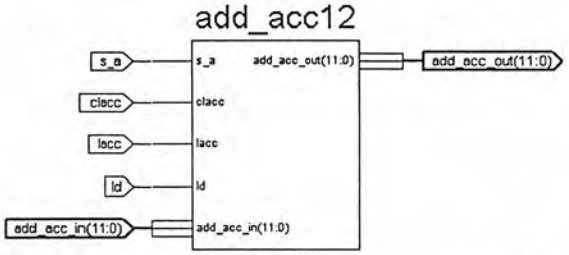
จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.125



รูปที่ 4.125 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 11 บิท

4.4.2.20 ส่วนของวงจรบวกสะสม 12 บิท

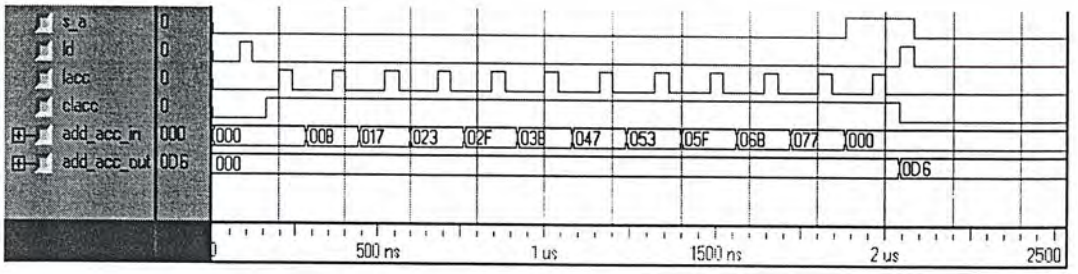
เป็นส่วนที่อยู่ในวงจร DA_CORDIC 5_1 - DA_CORDIC 6_1 โดยรับอินพุทมาจากส่วนของวงจรเก็บค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ มาทำการบวกสะสมตามหลักการของโครงสร้างเลขคณิตกระจาย (Distributed Arithmetic) สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรบวกสะสม ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.126



รูปที่ 4.126 แสดงสัญลักษณ์ของวงจรบวกสะสม 12 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

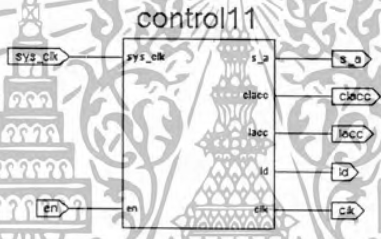
จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.127



รูปที่ 4.127 แสดงผลการจำลองการทำงานของวงจรบวกสะสม 12 บิท

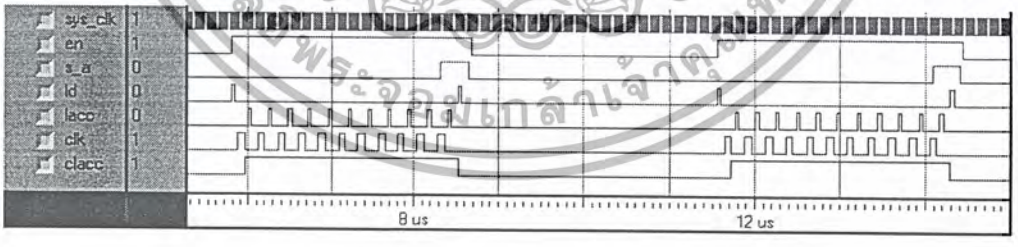
4.4.2.21 ส่วนของวงจรควบคุมภายในวงจร DA_CORDIC 1_1 - DA_CORDIC 4_1

ทำหน้าที่สร้างสัญญาณควบคุมการทำงานของแต่ละส่วนภายในวงจร DA_CORDIC 1_1 - DA_CORDIC 4_1 ให้มีความถูกต้องได้ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมภายในวงจรDA_CORDIC1_1–DA_CORDIC4_1 ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.128



รูปที่ 4.128 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร DA_CORDIC 1_1 – DA_CORDIC 4_1

จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.129

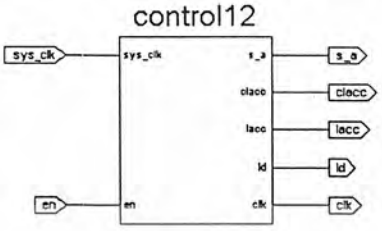


รูปที่ 4.129 แสดงการจำลองการทำงานของวงจรควบคุมภายใน
วงจร DA_CORDIC 1_1 – DA_CORDIC 4_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

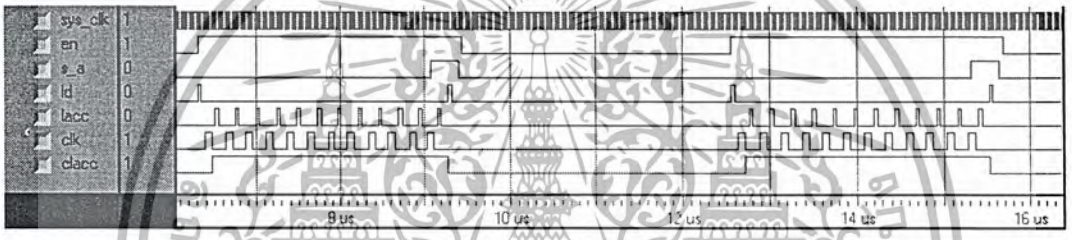
4.4.2.22 ส่วนของวงจรควบคุมภายในวงจร DA_CORDIC 5_1 - DA_CORDIC 6_1

ทำหน้าที่สร้างสัญญาณควบคุมการทำงานของแต่ละส่วนภายในวงจร DA_CORDIC 5_1 - DA_CORDIC 6_1 ให้มีความถูกต้อง สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมภายในวงจร DA_CORDIC 5_1 – DA_CORDIC 6_1 ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.130



รูปที่ 4.130 แสดงสัญลักษณ์ของวงจรควบคุมภายในวงจร DA_CORDIC 5_1 – DA_CORDIC 6_1

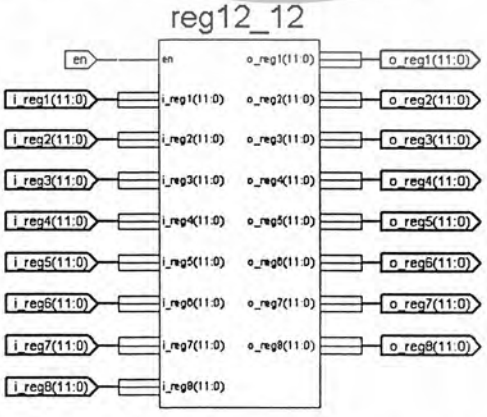
จากโปรแกรมสามารถแสดงผลการจำลองการทำงานได้ดังรูปที่ 4.131



รูปที่ 4.131 แสดงการจำลองการทำงานของวงจรควบคุมภายในวงจร DA_CORDIC 5_1 – DA_CORDIC 6_1

4.4.2.23 ส่วนของวงจรรีจิสเตอร์ขนาด 12 บิท (Register)

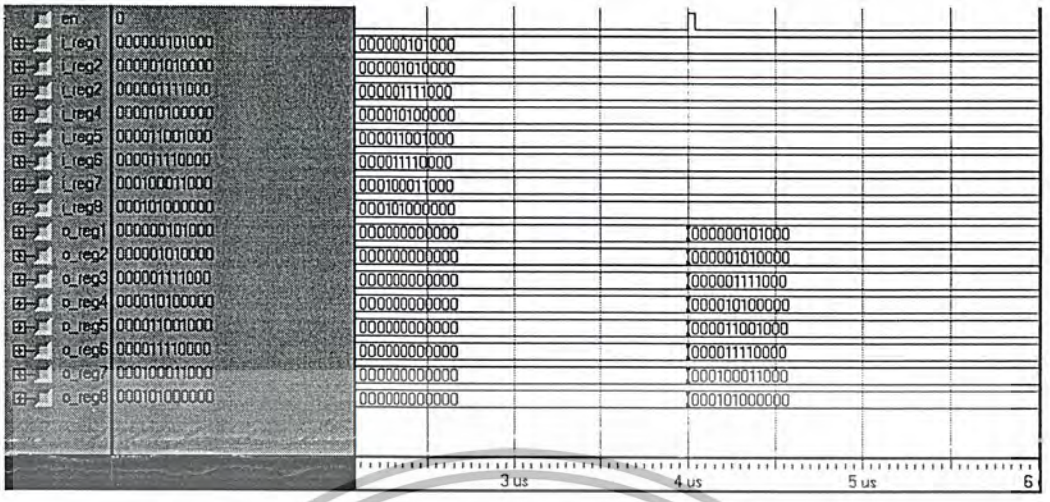
ทำหน้าที่เก็บข้อมูลขนาด 12 บิท ที่ได้จากการแปลงดิสครีตโคซายน์ที่ส่งเอาท์พุทออกมาไม่พร้อมกัน โดยจะทำการเก็บข้อมูลขนาด 12 บิท ให้ครบ 8 ค่า แล้วทำการส่งข้อมูลเอาท์พุทออกไปพร้อม ๆ กันเพื่อเป็นอินพุทของวงจรส่วนต่อไป สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรรีจิสเตอร์ขนาด 12 บิท ได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.132



รูปที่ 4.132 แสดงสัญลักษณ์ของวงจรรีจิสเตอร์ขนาด 12 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

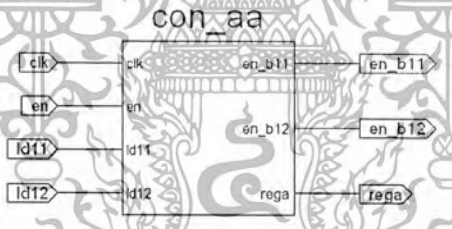
จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.133



รูปที่ 4.133 แสดงผลการจำลองการทำงานของวงจรรีจิสเตอร์ขนาด 12 บิท

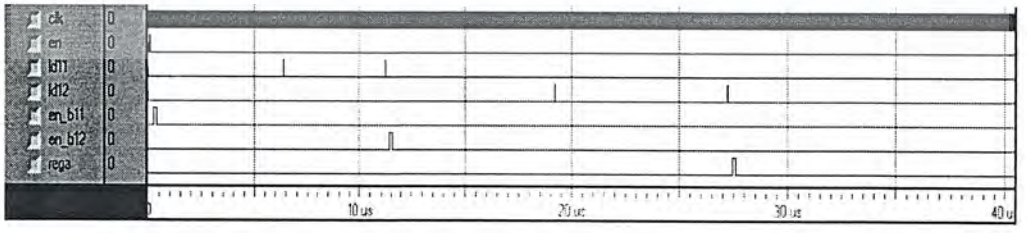
4.4.2.24 ส่วนของวงจรควบคุมการแปลงดิสครีตโคซายน์ส่วนที่ 2 หลังการทรานสโพส

ทำหน้าที่สร้างสัญญาณควบคุมการทำงานของแต่ละวงจรในส่วนการแปลงดิสครีต-โคซายน์ เพื่อให้สัญญาณเอาท์พุทที่ได้จากการแปลงดิสครีตโคซายน์มีความถูกต้อง สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรควบคุมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.134



รูปที่ 4.134 แสดงสัญลักษณ์ของวงจรควบคุมการแปลงดิสครีตโคซายน์

จากโปรแกรมที่เขียนขึ้นสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.135



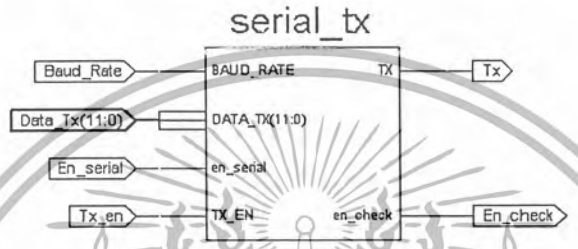
รูปที่ 4.135 แสดงผลการจำลองการทำงานของวงจรควบคุมการแปลงดิสครีตโคซายน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ส่วนของการส่งข้อมูลออกจากบอร์ด FPGA ผ่านทางพอร์ตอนุกรม

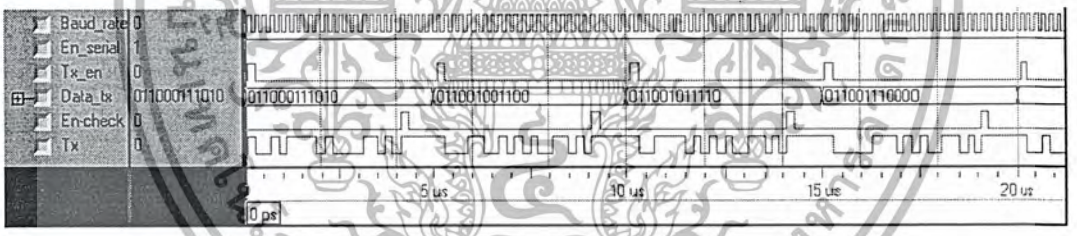
4.5.1 ส่วนของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม (SERIAL_TX)

ส่วนของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม ทำหน้าที่ส่งเฟรมบิทของข้อมูลออกพอร์ตอนุกรมโดยการแปลงข้อมูลบิทขนานที่รับมาจากวงจรมัลติเพล็กซ์(Multiplex) เป็นบิทอนุกรมแล้วส่งข้อมูลออกพอร์ตอนุกรม ตามความถี่บอดเรด โดยแบ่งเฟรมข้อมูลออกเป็น 2 เฟรม บิทข้อมูลแต่ละเฟรมประกอบด้วย 6 บิท ของข้อมูลที่ต้องการส่งจริง และ 2 บิท ของข้อมูลที่แทรกเพิ่มเข้าไปเพื่อป้องกันการส่งข้อมูลที่เป็นบิท 0 ทั้งหมด สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.136



รูปที่ 4.136 แสดงสัญลักษณ์ของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม

จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.137

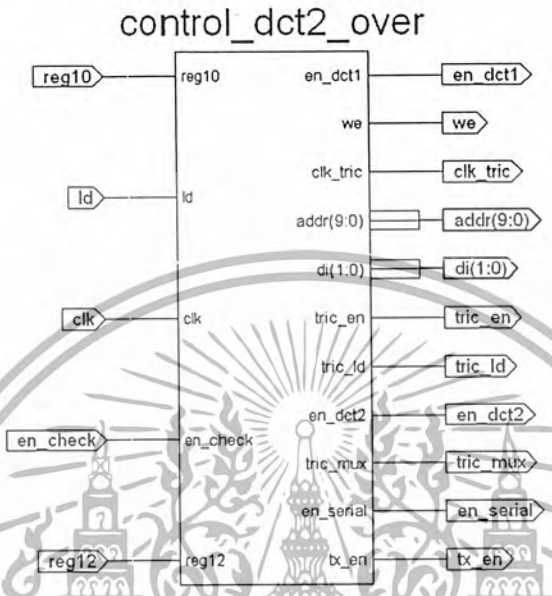


รูปที่ 4.137 แสดงผลการจำลองการทำงานของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

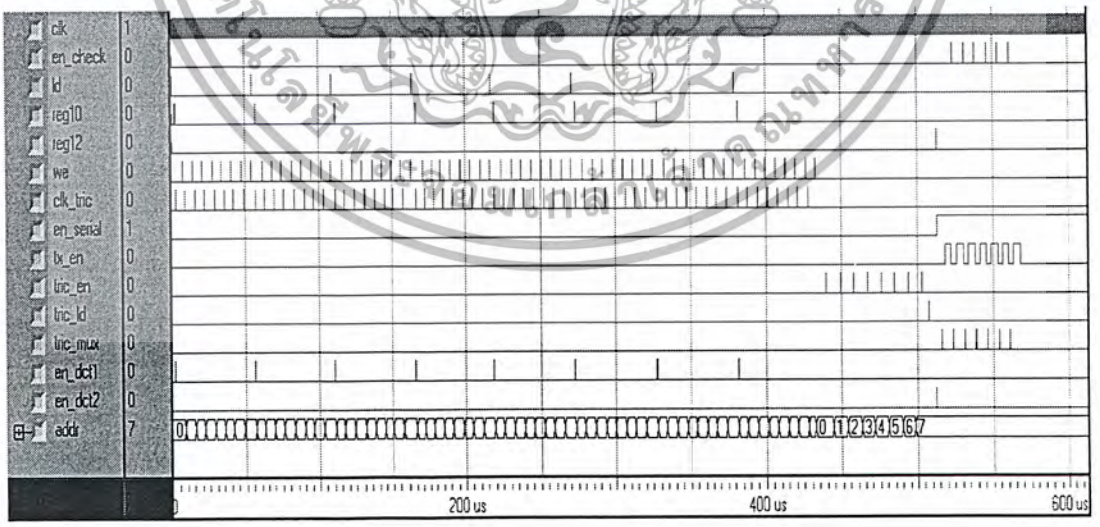
4.6 ส่วนของวงจรควบคุมการทำงานของระบบ

ส่วนของวงจรควบคุมการทำงานของระบบ (Control) ทำหน้าที่ในการควบคุมและจัดลำดับการทำงานของแต่ละวงจรให้ถูกต้องตามกระบวนการที่ได้กำหนดไว้ สามารถสังเคราะห์อุปกรณ์จากโปรแกรมของวงจรส่งบิทข้อมูลออกพอร์ตอนุกรมได้สัญลักษณ์ (Symbol) ดังรูปที่ 4.138



รูปที่ 4.138 แสดงสัญลักษณ์ของวงจรควบคุม

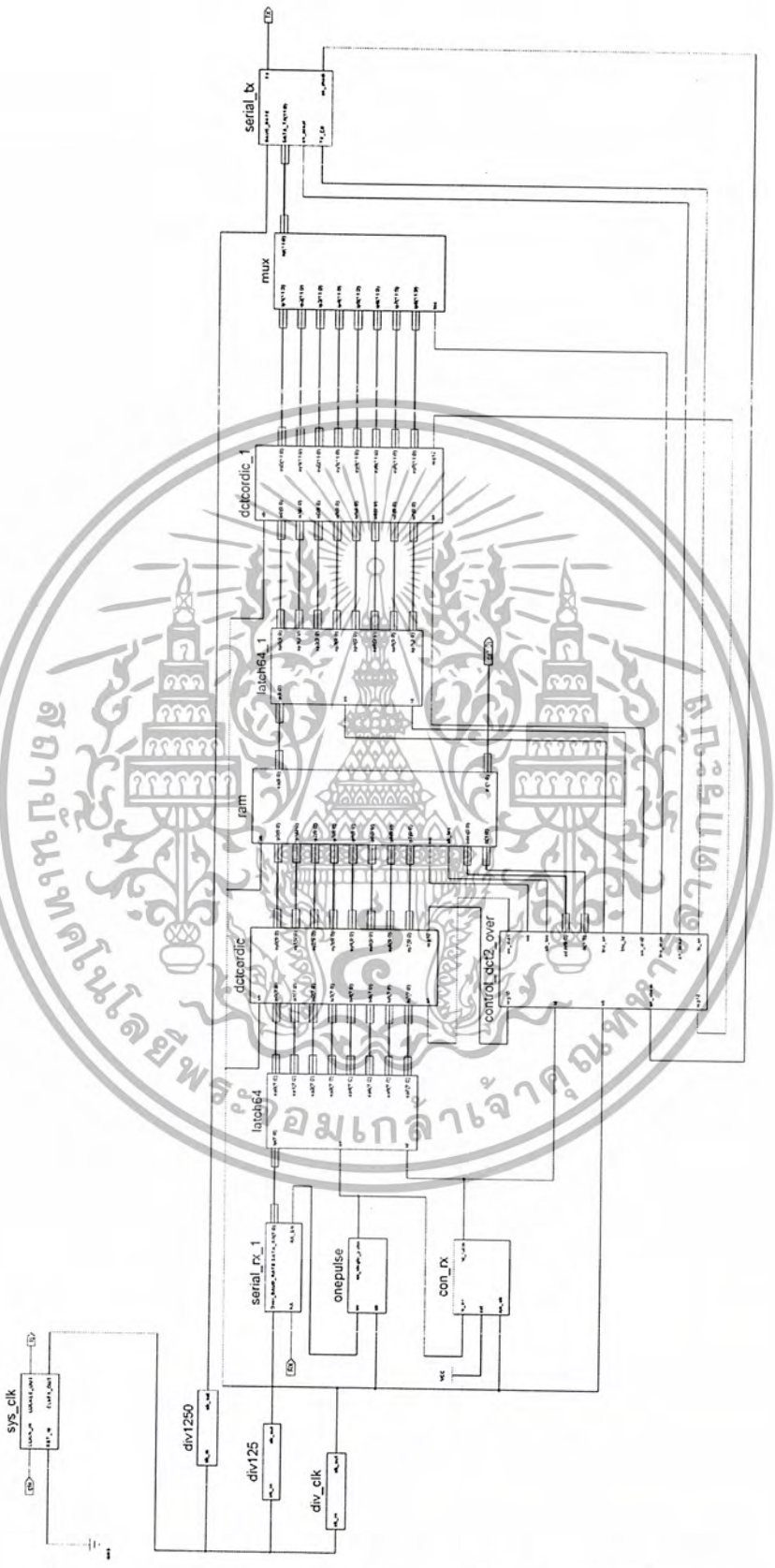
จากโปรแกรมที่เขียนสามารถทำการจำลองการทำงาน (Simulation) ได้ดังรูปที่ 4.139



รูปที่ 4.139 แสดงผลการจำลองการทำงานของวงจรควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

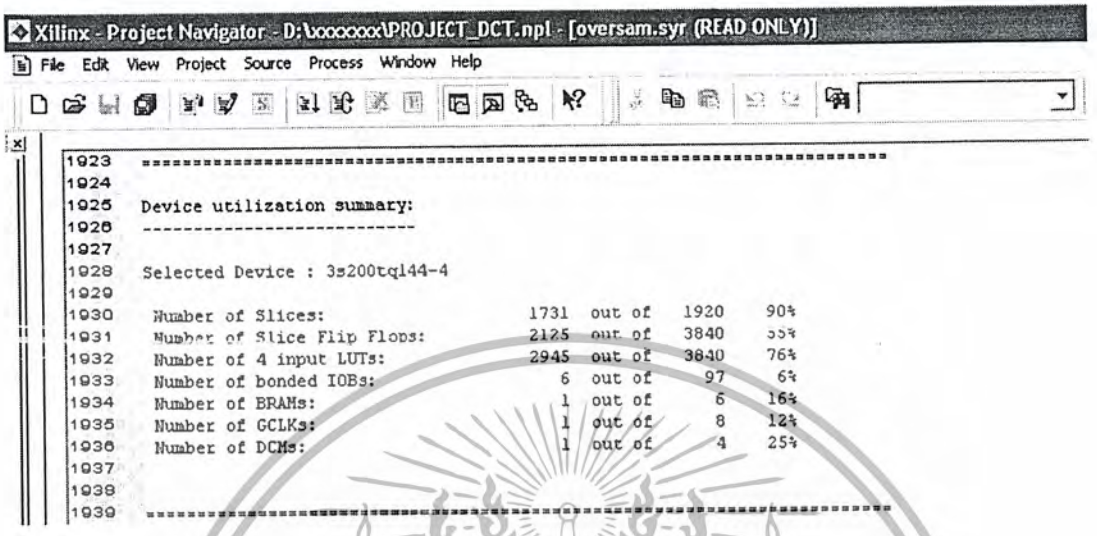
4.7 ส่วนประกอบภายในและการเชื่อมต่อของวงจรทั้งหมด



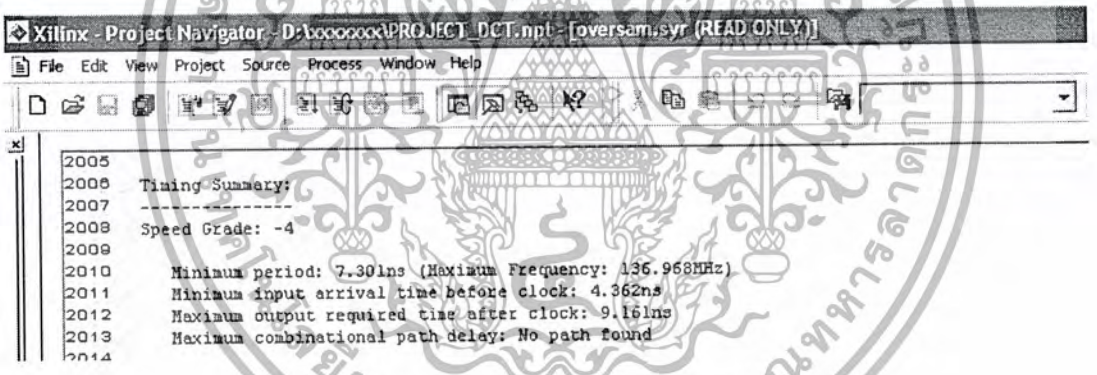
รูปที่ 4.140 แสดงส่วนประกอบภายในและการเชื่อมต่อของวงจรทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากส่วนประกอบภายในและการเชื่อมต่อของวงจรทั้งหมด ข้างต้นสามารถแสดงการคอมไพล์วงจรที่ออกแบบขึ้นมาโดยใช้ซอร์ฟแวร์ในการสังเคราะห์สามารถแสดงค่าอุปกรณ์ที่สามารถรองรับการทำงานและรายละเอียดต่าง ๆ ของเกท ที่ใช้งานโดยสามารถแสดงรายละเอียดได้ ดังรูปที่ 4.141



รูปที่ 4.141 แสดงค่าอุปกรณ์ที่ใช้ในการประมวลผลการทำงานจริง



รูปที่ 4.142 แสดงค่าเวลาที่ใช้ในการประมวลผลการทำงานจริง

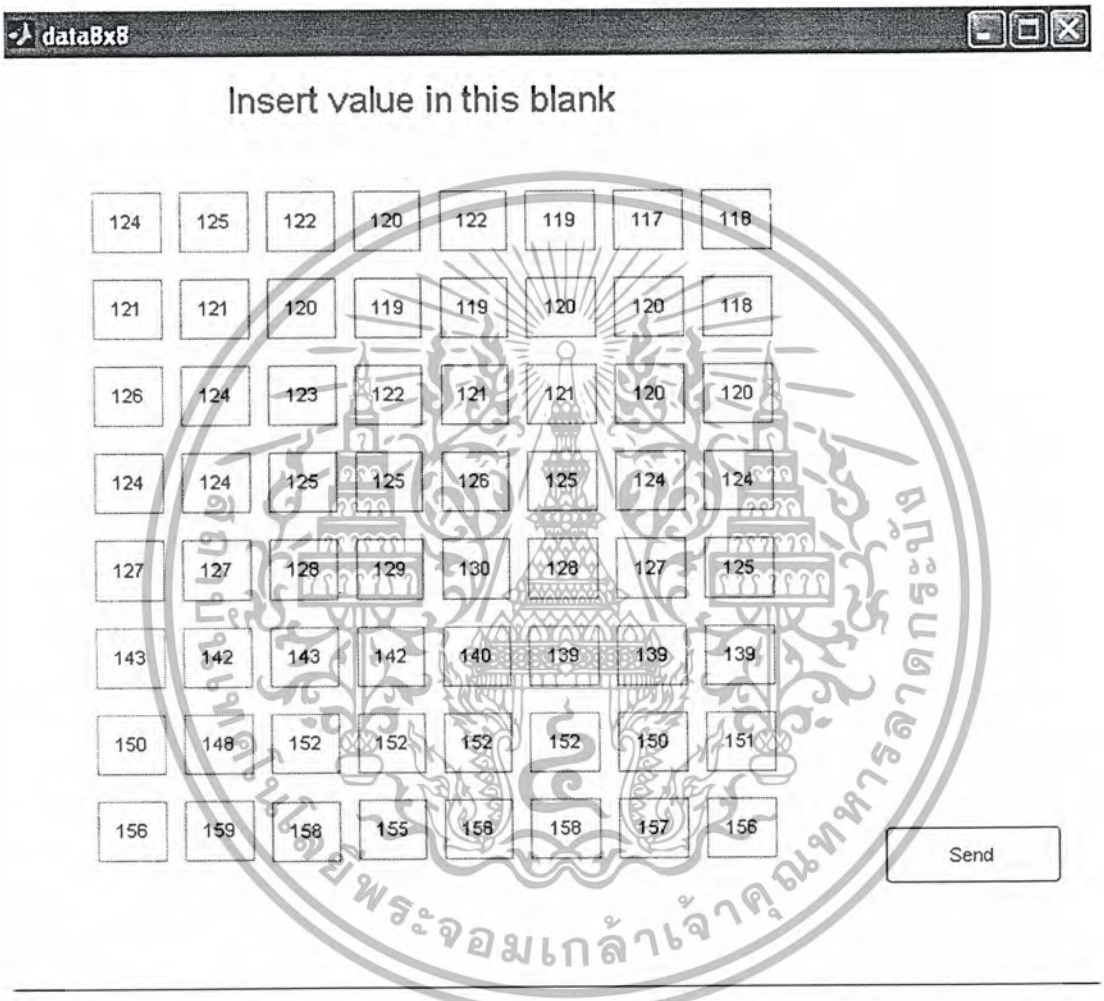
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.8 ส่วนของการประมวลผลการทำงานจริง

4.8.1 ขั้นตอนการเก็บผลการทดสอบ Hard ware ด้วยชุดข้อมูลขนาด 8X8

ในขั้นตอนการประมวลผลการทำงานจริงจะทำการส่งข้อมูลผ่านทางพอร์ตอนุกรมโดยใช้โปรแกรม MATLAB เป็นตัวรับข้อมูลจากผู้ใช้งาน (Graphic User Interface) ซึ่งขั้นตอนและผลการทดลองได้ดังต่อไปนี้

1. ข้อมูลภาพที่นำมาทดสอบเป็นบล็อกขนาด 8X8 ซึ่งมีค่าอินพุท ดังรูปที่ 4.143



รูปที่ 4.143 สัญญาณภาพที่นำมาทดสอบเป็นบล็อกขนาด 8X8

2. ทำการป้อนค่าข้อมูลภาพที่ผู้ใช้ต้องการทำการแปลงโดยผ่านโปรแกรมที่เขียนขึ้นมาเพื่อทดสอบการทำงาน Hard ware ที่เขียนด้วย MATLAB ซึ่งค่าสัญญาณอินพุทเป็นเกรย์สเกลที่อยู่ในช่วง 0 ถึง 255 ต้องทำการถ่วงน้ำหนักให้อยู่ในช่วง -127 ถึง 128 ซึ่งการถ่วงน้ำหนักจะทำการประมวลผลด้วยโปรแกรม MATLAB และข้อมูลที่ได้กลับมาจาก Hard ware จะเป็นค่าข้อมูลที่ได้ทำการแปลงดิสครีตโค-ซายน์แบบ 2 มิติ แล้วจะอยู่ในรูปของข้อมูลที่เป็นไบนารี ดังนั้นเราจึงต้องนำข้อมูลที่รับมาได้มาทำการประมวลผลและทำการแปลงกลับด้วยโปรแกรม MATLAB แล้วจึงนำค่าที่ได้มาเปรียบเทียบกับข้อมูลอินพุทที่เป็นสัญญาณภาพต้นแบบมีขั้นตอนคือ

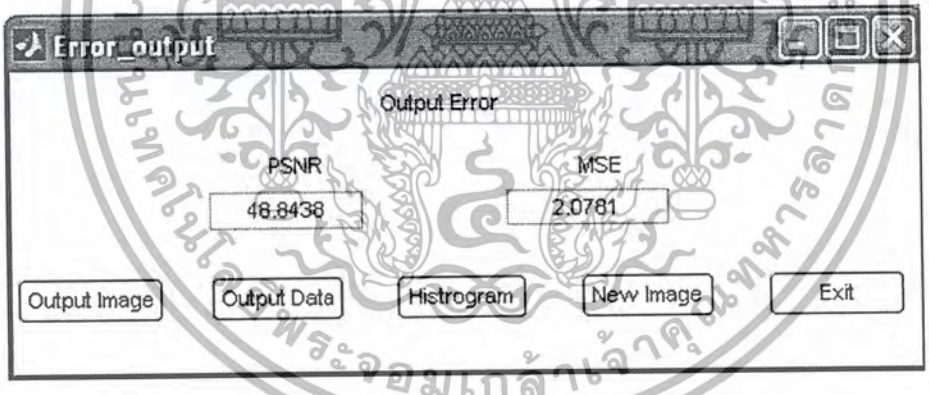
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 ทำการถ่วงค่าน้ำหนักให้ค่าอยู่ในช่วง -127 ถึง 128 โดยนำข้อมูลอินพุทมาลบกับ 128

2.2 ทำการแปลงข้อมูลที่เป็นเลขฐานสิบในเป็นเลขฐานสองแล้วส่งออกพอร์ตอนุกรม

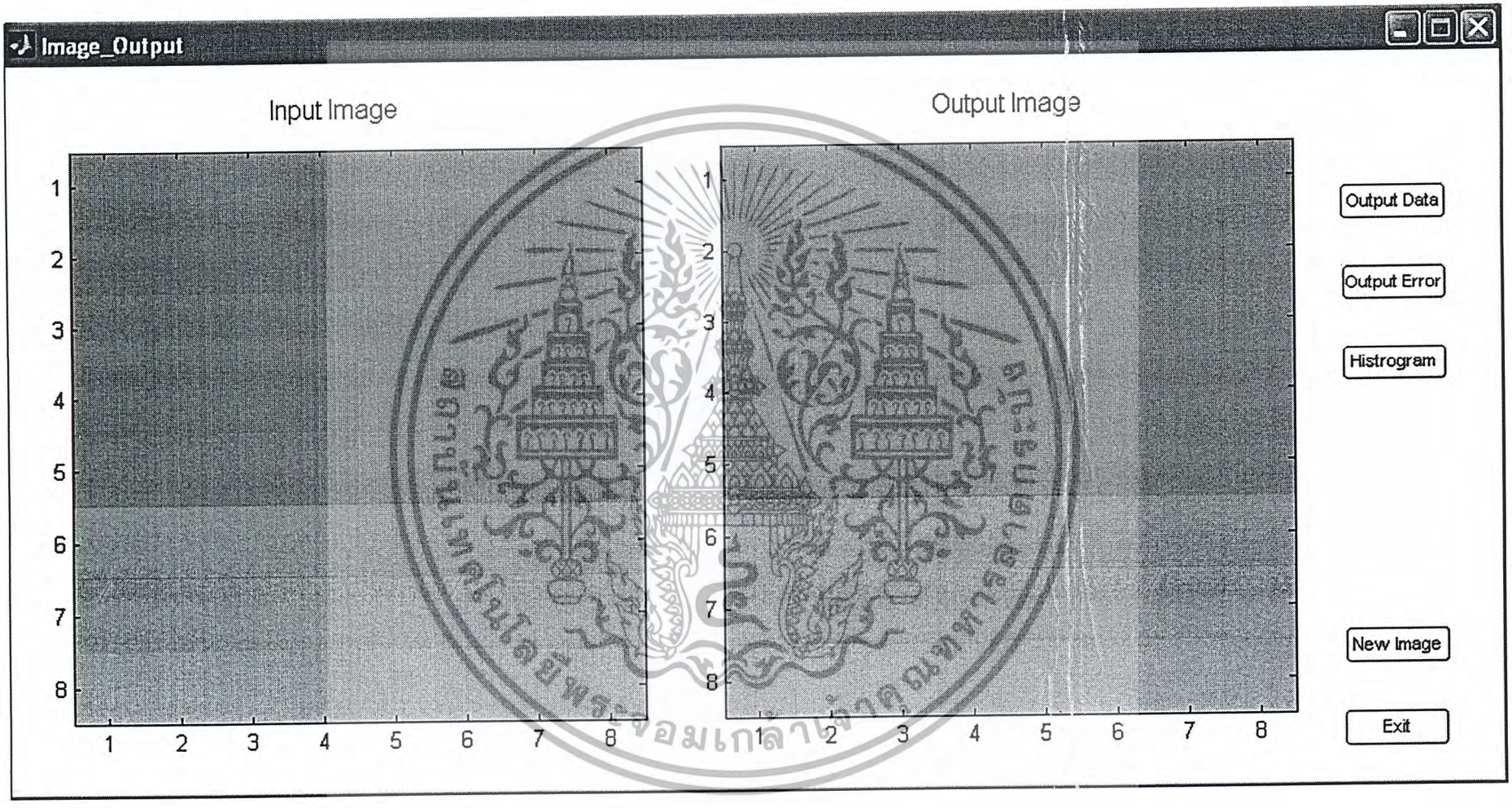
2.3 ค่าที่ได้จากการแปลงดิสครีตโคซายน์ 2 มิติ ค่าหนึ่งจะเป็นข้อมูลจำนวน 12 บิท เราจะทำการแบ่งข้อมูลออกเป็น 2 เฟรม เฟรมละ 6 บิท แล้วทำการแทรก '01' ที่ MSB ทั้งนี้เพื่อลดอัตราเกิดความผิดพลาดในการรับส่งในกรณีที่ข้อมูลเป็นศูนย์หมด

3. ซึ่งจากการทำงานของ Hard ware จะส่งค่ากลับมายังโปรแกรมที่เขียนขึ้นมาเพื่อทดสอบการทำงานของ Hard ware จำนวน 128 ไบท์ เนื่องจากค่าที่ได้ผ่านการแปลงดิสครีตโคซายน์แล้ว จะมีค่า ๆ ละ 12 บิท แทรกบิท 4 บิท จำนวน 64 ชุด (64 X 16 = 1024 บิท) และค่าที่ส่งกลับมาจาก Hard ware จะเป็นเลขฐาน 2 ชุดละ 1 ไบท์ จำนวน 128 ชุด ดังนั้นจึงต้องทำการตัดบิทที่แทรก 2 บิท ออกแล้วนำเฟรมแรกที่รับมาได้มาต่อกับเฟรมที่สอง จะได้ข้อมูลจำนวน 12 บิท จากนั้นเปลี่ยนข้อมูลที่ได้ทำการตัดบิทแปลงให้เป็นเลขฐาน 10 แล้วทำการแปลงกลับดิสครีตโคซายน์แบบ 2 มิติ ด้วยโปรแกรม MATLAB ซึ่งจากสัญญาณเสียงที่นำมาทดสอบ (man1.jpg) มีขนาด 128X128 พิกเซล ซึ่งการทำงานของโปรแกรมที่เขียนขึ้นมาเพื่อทดสอบต้องแบ่งข้อมูลออกเป็นบล็อกขนาด 8X8 จะต้องทำการส่งค่าทั้งหมดจำนวน 256 ครั้ง ซึ่งเมื่อทำการประมวลผลเสร็จเรียบร้อยแล้วเราสามารถหาค่าความผิดพลาดที่เกิดจากการแปลงดิสครีตโคซายน์แบบ 2 มิติ

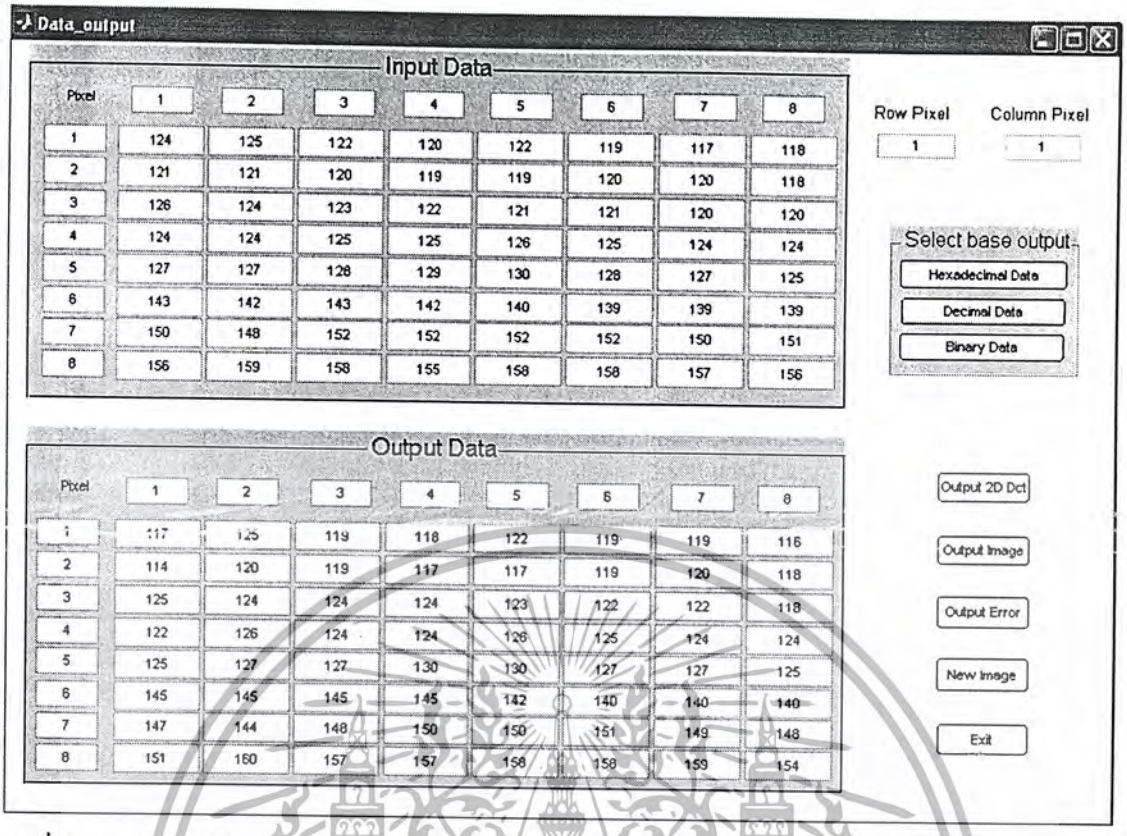


รูปที่ 4.144 แสดงค่า PSNR กับค่าความผิดพลาดที่เกิดขึ้นในการทำงานจริงของบล็อกข้อมูลขนาด 8X8

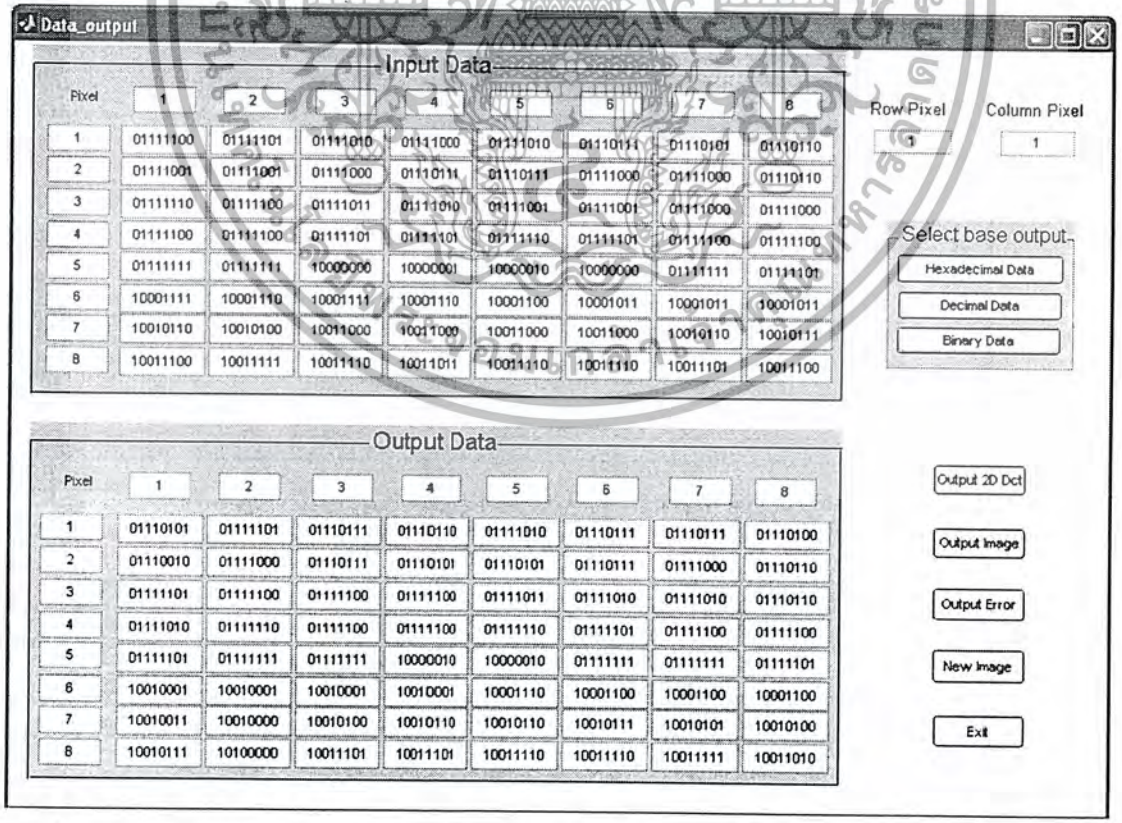
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.145 แสดงการเปรียบเทียบสัญญาณต้นแบบกับสัญญาณที่สร้างกลับของบล็อกข้อมูลขนาด 8X8

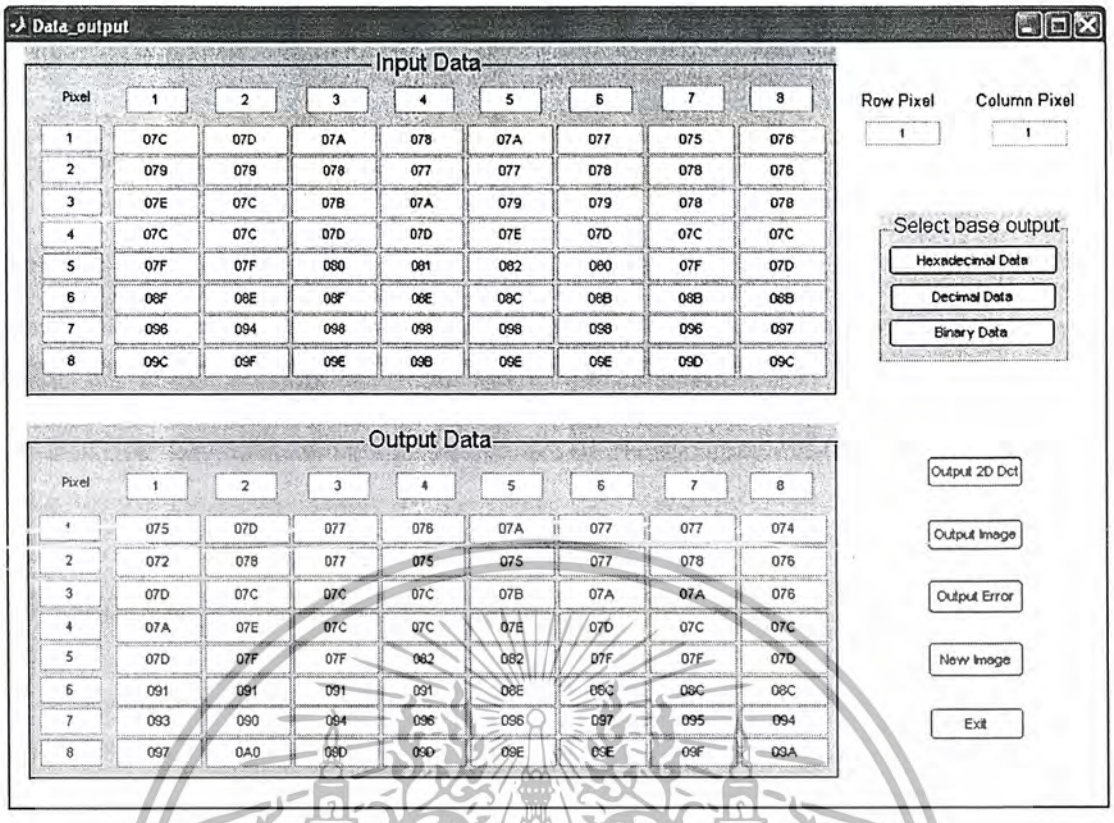


รูปที่ 4.146 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของบล็อกข้อมูลขนาด 8X8เป็นเลขฐาน 10

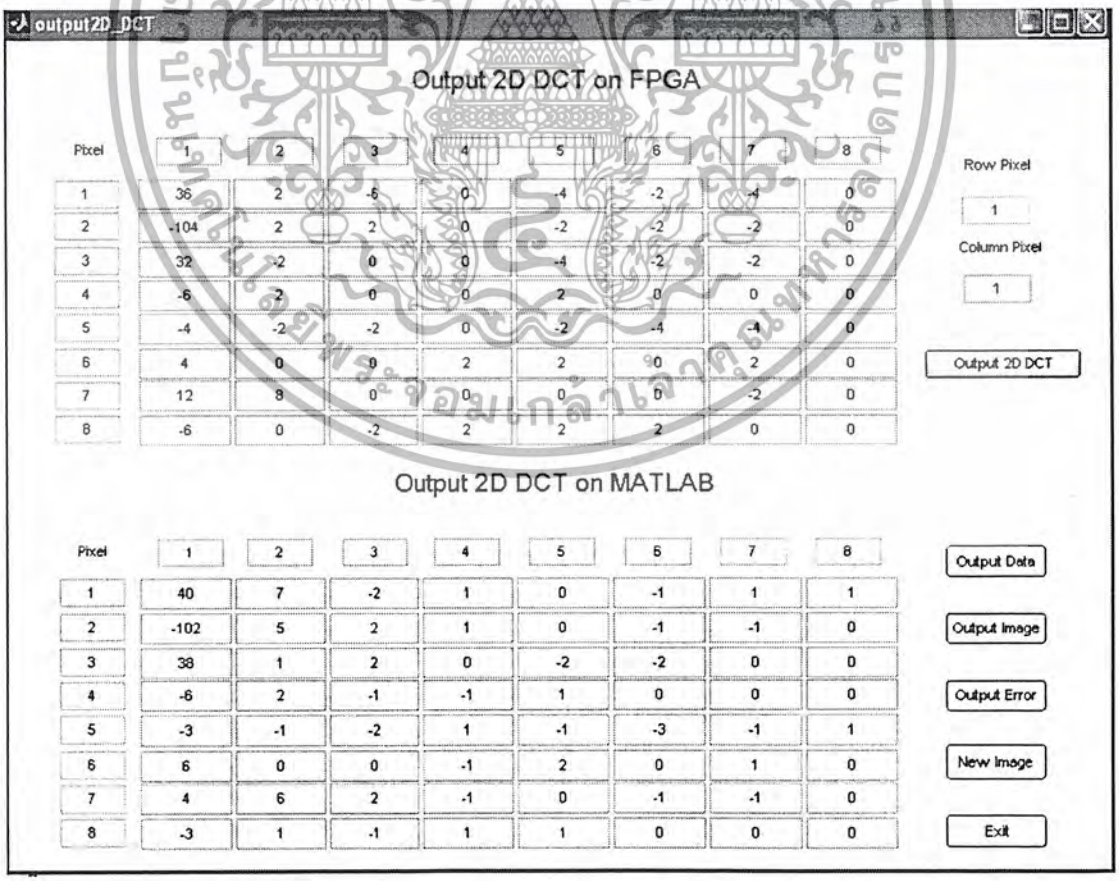


รูปที่ 4.147 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของบล็อกข้อมูลขนาด 8X8เป็นเลขฐาน 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

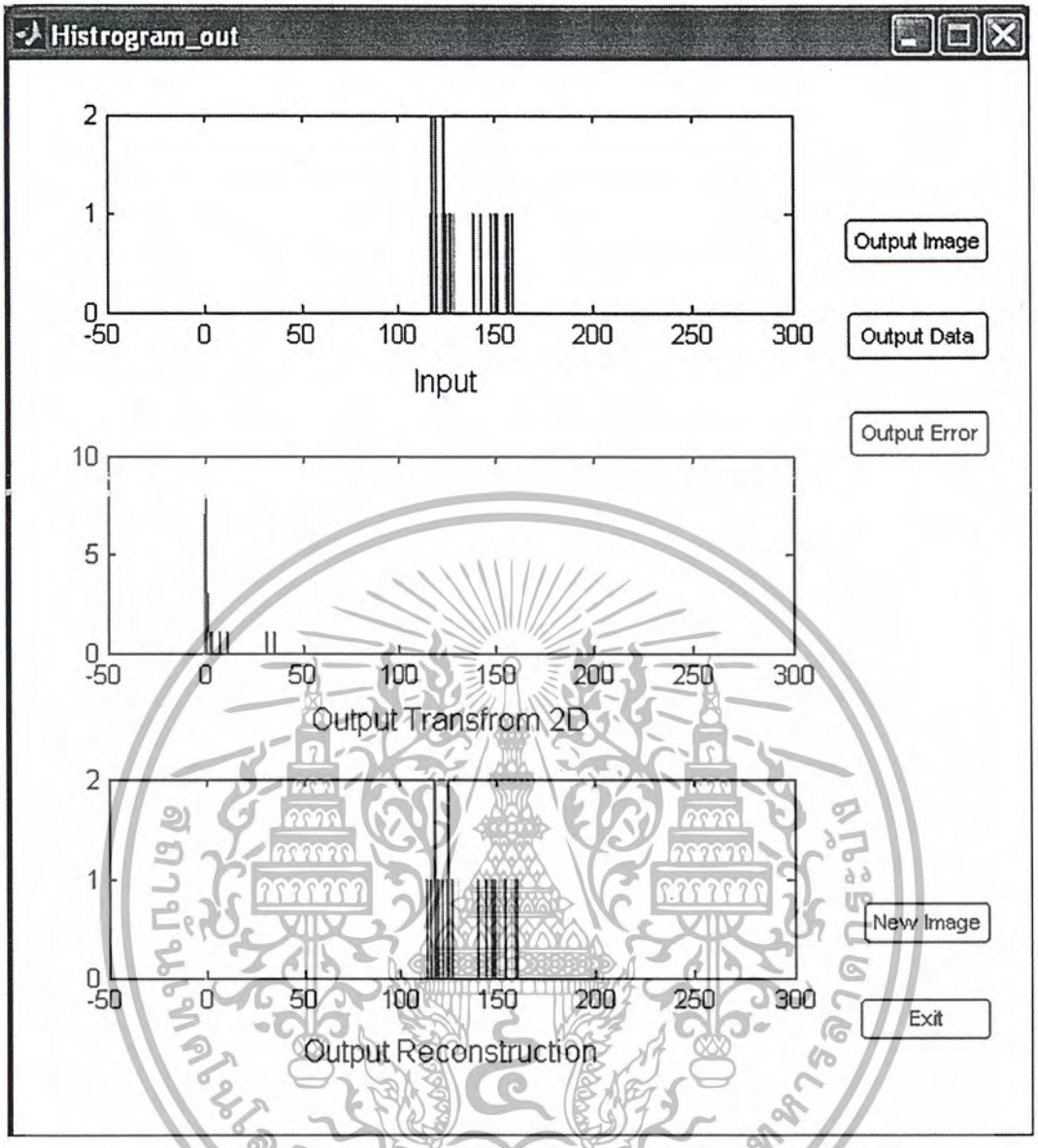


รูปที่ 4.148 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของบล็อกข้อมูลขนาด 8X8เป็นเลขฐาน 16



รูปที่ 4.149 แสดงค่าข้อมูลที่ได้จากโปรแกรม MATLAB เปรียบเทียบกับการทำงานจริงบนบอร์ด FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



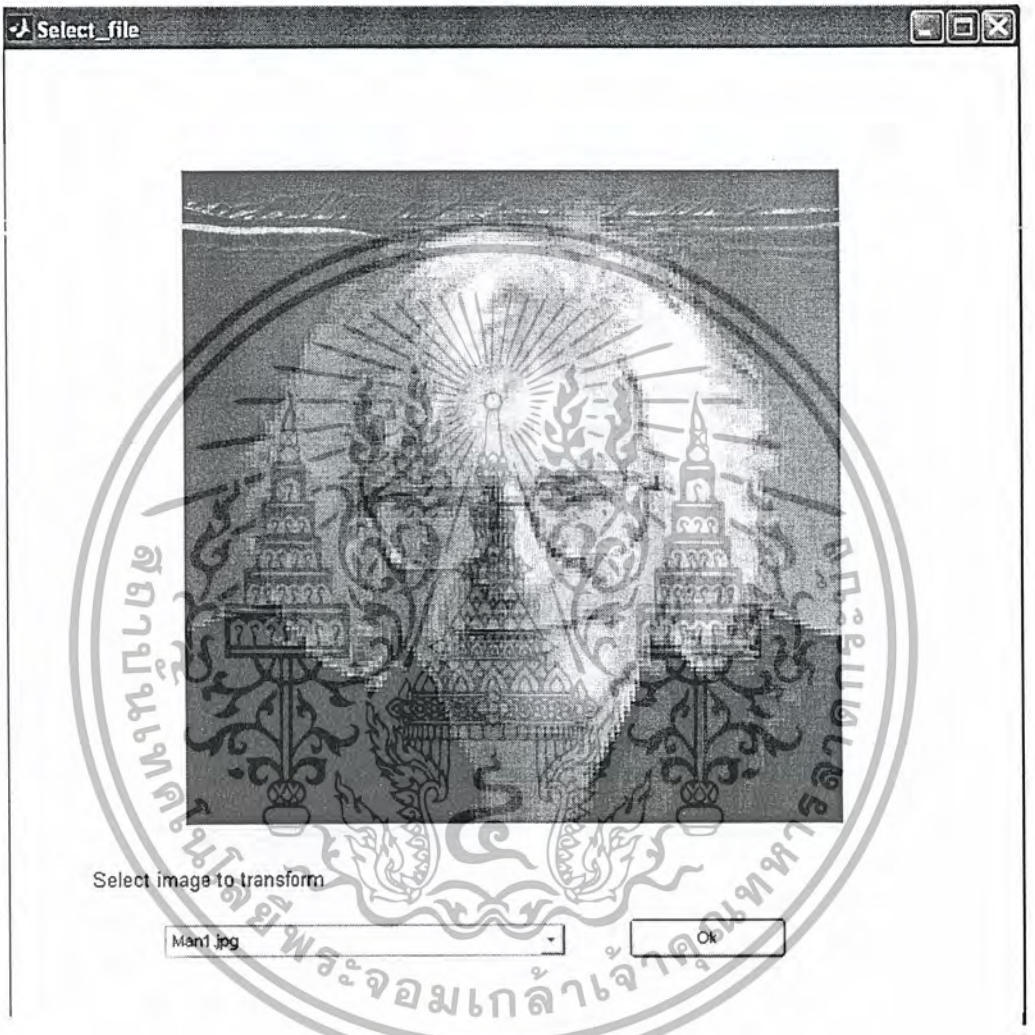
รูปที่ 4.150 ฮิสโตแกรมเปรียบเทียบข้อมูลอินพุท ข้อมูลที่ได้จากการแปลง และข้อมูลในการสร้างกลับของบล็อกข้อมูลขนาด 8X8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.8.2 ขั้นตอนการเก็บผลการทดสอบ Hard ware ด้วยชุดข้อมูลภาพ

ในขั้นตอนการประมวลผลการทำงานจริงจะทำการส่งข้อมูลผ่านทางพอร์ตอนุกรมโดยใช้โปรแกรม MATLAB เป็นตัวรับข้อมูลจากผู้ใช้งาน (Graphic User Interface) ซึ่งขั้นตอนและผลการทดลองได้ดังต่อไปนี้

1. ข้อมูลภาพที่นำมาทดสอบมีชื่อว่า man1.jpg ซึ่งมีลักษณะสัญญาณดังรูปที่ 4.151



รูปที่ 4.151 สัญญาณภาพที่นำมาทดสอบ

2. ทำการป้อนค่าข้อมูลภาพที่ผู้ใช้ต้องการทำการแปลงโดยผ่านโปรแกรมที่เขียนขึ้นมาเพื่อทดสอบการทำงาน Hard ware ที่เขียนด้วย MATLAB ซึ่งค่าสัญญาณอินพุทเป็นเกรย์สเกลที่อยู่ในช่วง 0 ถึง 255 ต้องทำการถ่วงน้ำหนักให้อยู่ในช่วง -127 ถึง 128 ซึ่งการถ่วงน้ำหนักจะทำการประมวลผลด้วยโปรแกรม MATLAB และข้อมูลที่ได้กลับมาจาก Hard ware จะเป็นค่าข้อมูลที่ได้ทำการแปลงดิสครีตโค-ซายน์ 2 มิติ แล้วจะอยู่ในรูปของข้อมูลที่เป็นไบนารี ดังนั้นเราจึงต้องนำข้อมูลที่รับมาได้มาทำการประมวลผลและทำการแปลงกลับด้วยโปรแกรม MATLAB แล้วจึงนำค่าที่ได้มาเปรียบเทียบกับข้อมูลอินพุทที่เป็นสัญญาณภาพต้นแบบมีขั้นตอนคือ

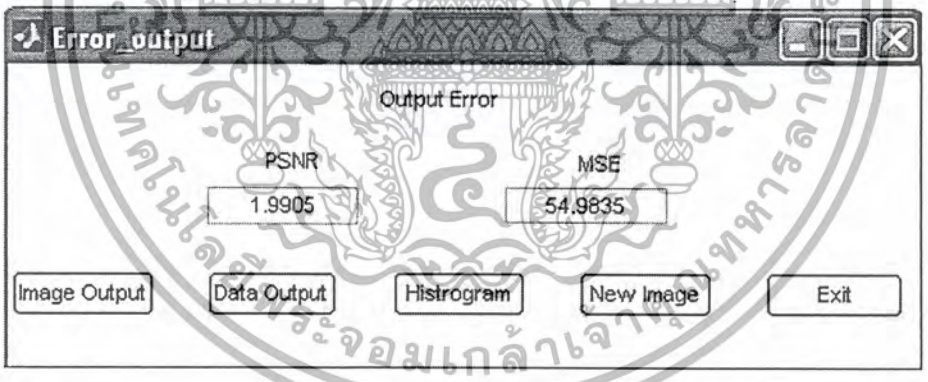
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 ทำการถ่วงค่าน้ำหนักให้ค่าอยู่ในช่วง -127 ถึง 128 โดยนำข้อมูลอินพุทมาลบกับ 128

2.2 ทำการแปลงข้อมูลที่เป็นเลขฐานสิบในเป็นเลขฐานสองแล้วส่งออกพอร์ตอนุกรม

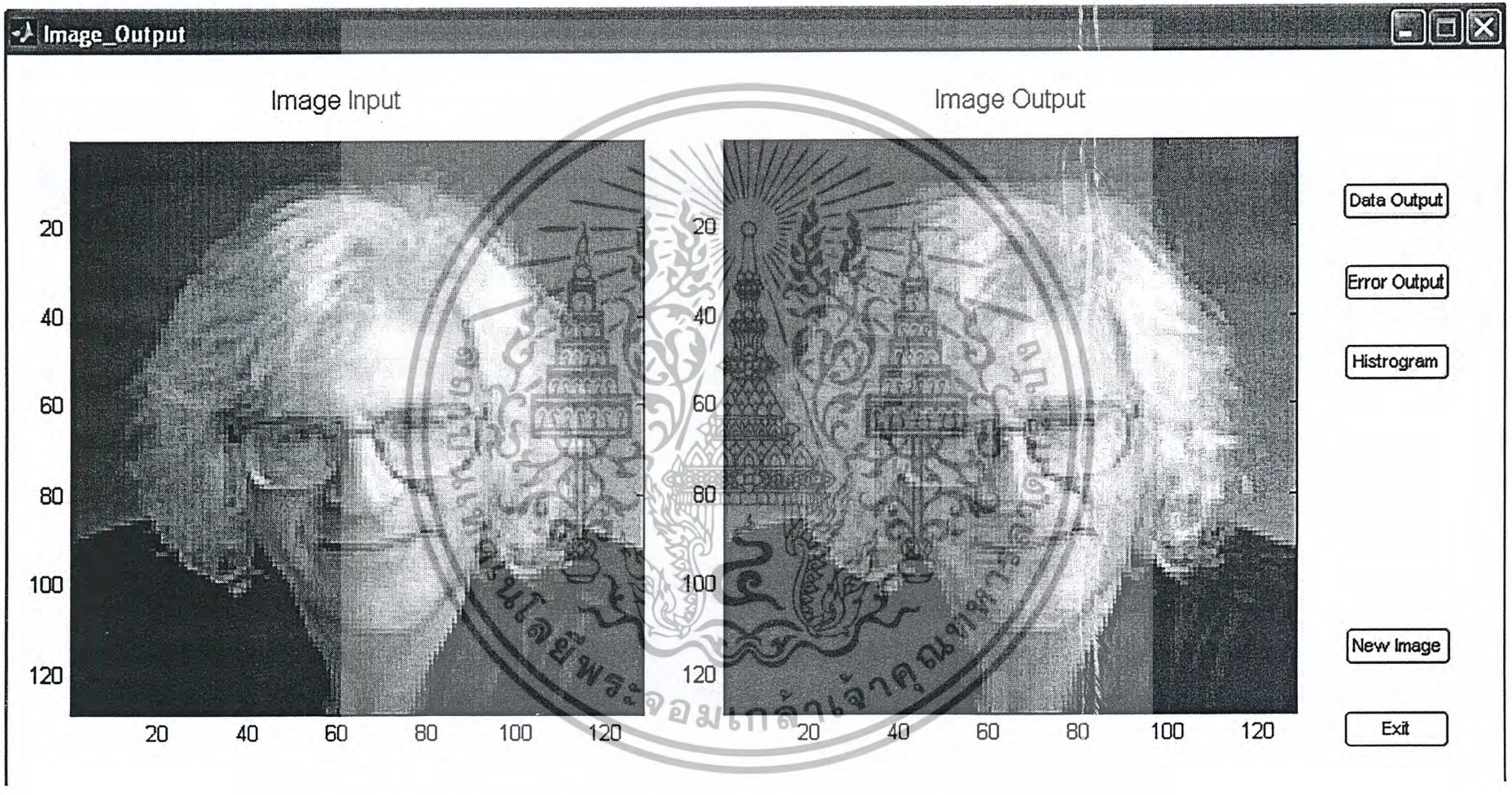
2.3 ค่าที่ได้จากการแปลงดิสครีตโคซายน์ 2 มิติ ค่าหนึ่งจะเป็นข้อมูลจำนวน 12 บิท เราจะทำการแบ่งข้อมูลออกเป็น 2 เฟรม เฟรมละ 6 บิท แล้วทำการแทรก '01' ที่ MSB ทั้งนี้เพื่อลดอัตราเกิดความผิดพลาดในการรับส่งในกรณีที่ข้อมูลเป็นศูนย์หมด

3 ซึ่งจากการทำงานของ Hard ware จะส่งค่ากลับมายังโปรแกรมที่เขียนขึ้นมาเพื่อทดสอบการทำงานของ Hard ware จำนวน 128 ไบท์ เนื่องจากค่าที่ได้ผ่านการแปลงดิสครีตโคซายน์แล้ว จะมีค่า ๆ ละ 12 บิท แทรกบิท 4 บิท จำนวน 64 ชุด (64 X 16 = 1024 บิท) และค่าที่ส่งกลับมาจาก Hard ware จะเป็นเลขฐาน 2 ชุดละ 1 ไบท์ จำนวน 128 ชุด ดังนั้นจึงต้องทำการตัดบิทที่แทรก 2 บิท ออกแล้วนำเฟรมแรกที่รับมาได้มาต่อกับเฟรมที่สอง จะได้ข้อมูลจำนวน 12 บิท จากนั้นเปลี่ยนข้อมูลที่ได้ทำการตัดบิทแปลงให้เป็นเลขฐาน 10 แล้วทำการแปลงกลับดิสครีตโคซายน์แบบ 2 มิติ ด้วยโปรแกรม MATLAB ซึ่งจากสัญญาณเสียงที่นำมาทดสอบ (man1.jpg) มีขนาด 128X128 พิกเซล ซึ่งการทำงานของโปรแกรมที่เขียนขึ้นมาเพื่อทดสอบต้องแบ่งข้อมูลออกเป็นบล็อกขนาด 8X8 จะต้องทำการส่งค่าทั้งหมดจำนวน 256 ครั้ง ซึ่งเมื่อทำการประมวลผลเสร็จเรียบร้อยแล้วเราสามารถหาค่าความผิดพลาดที่เกิดจากการแปลงดิสครีตโคซายน์แบบ 2 มิติ

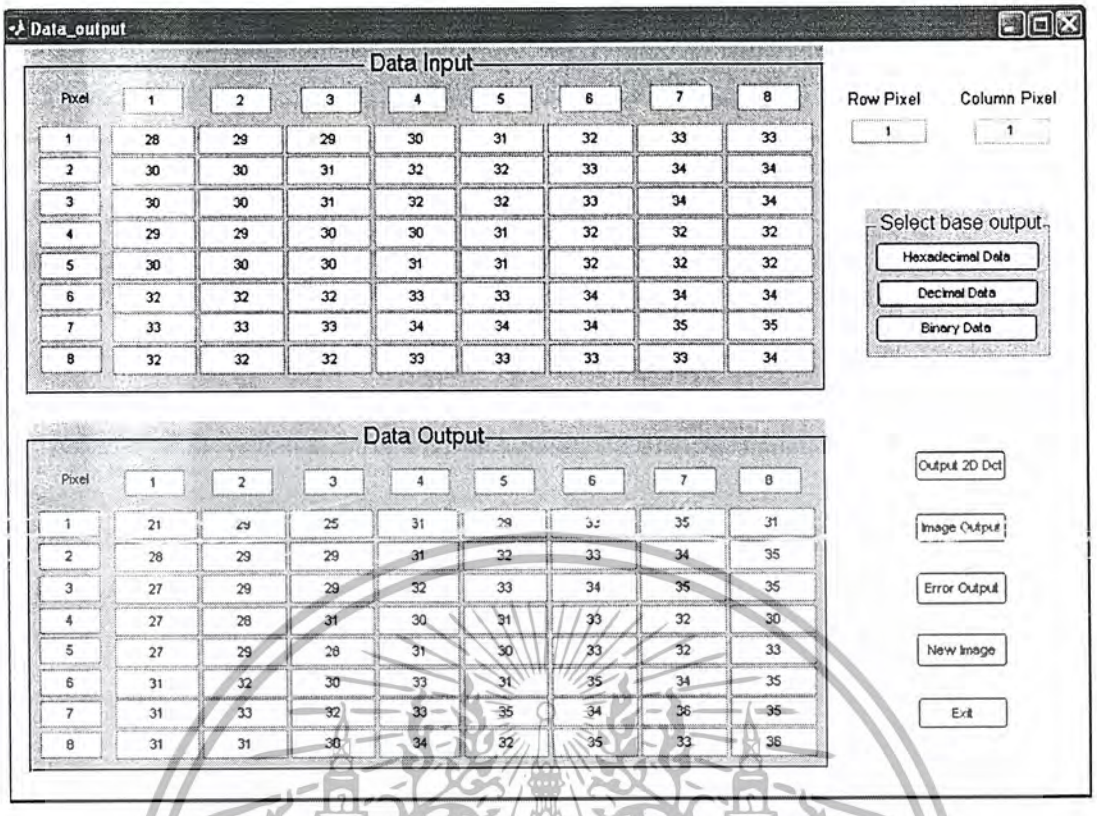


รูปที่ 4.152 แสดงค่า PSNR กับค่าความผิดพลาดที่เกิดขึ้นในการทำงานจริงของสัญญาณภาพ

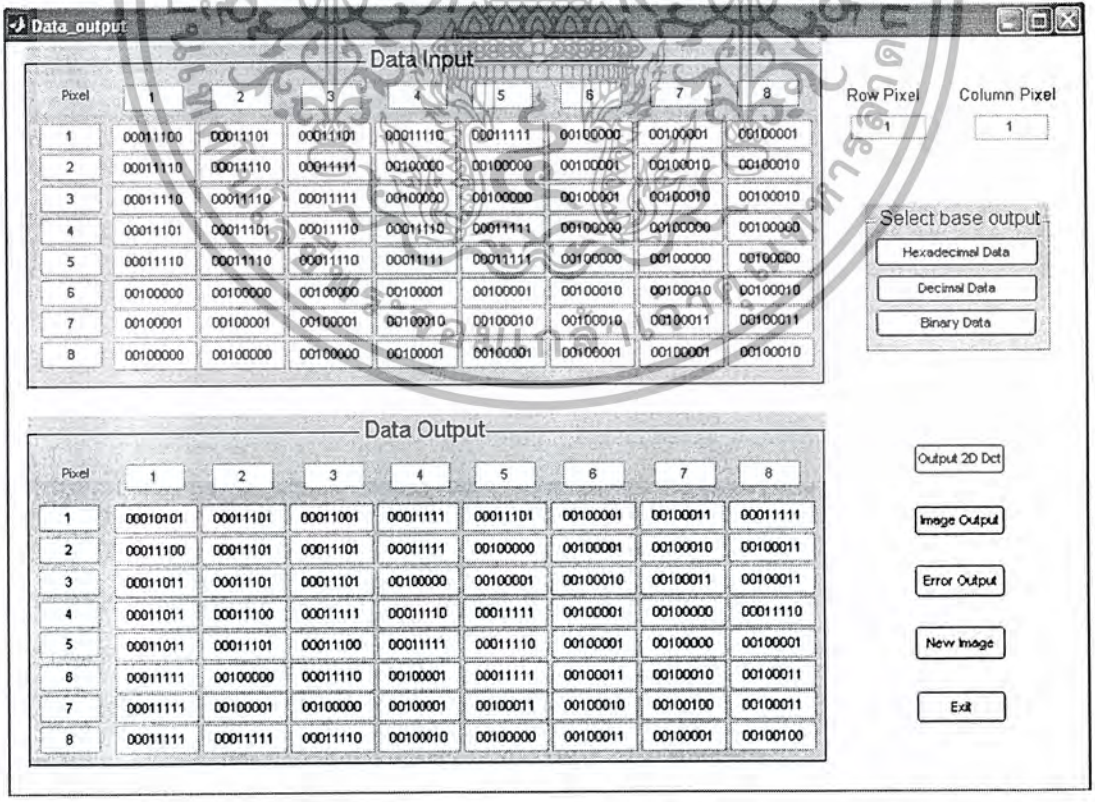
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.153 แสดงการเปรียบเทียบสัญญาณต้นแบบกับสัญญาณที่สร้างกลับของสัญญาณภาพ

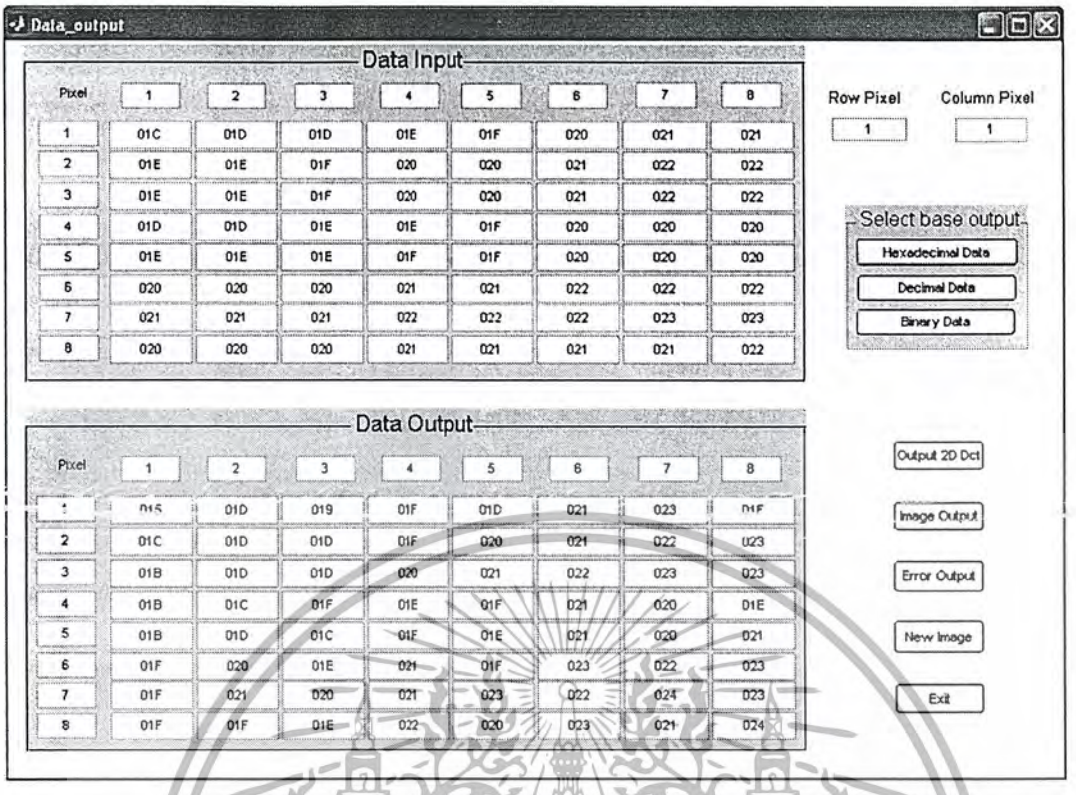


รูปที่ 4.154 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของสัญญาณภาพเป็นเลขฐาน 10

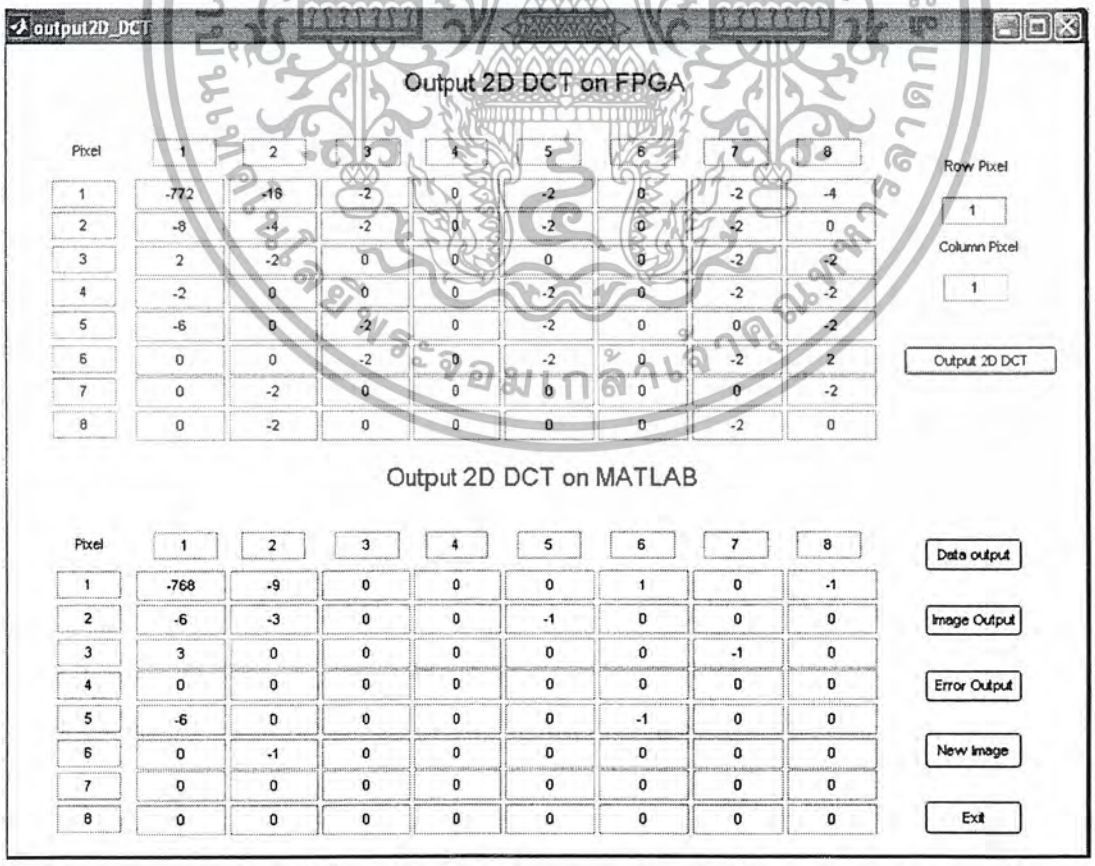


รูปที่ 4.155 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของสัญญาณภาพเป็นเลขฐาน 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

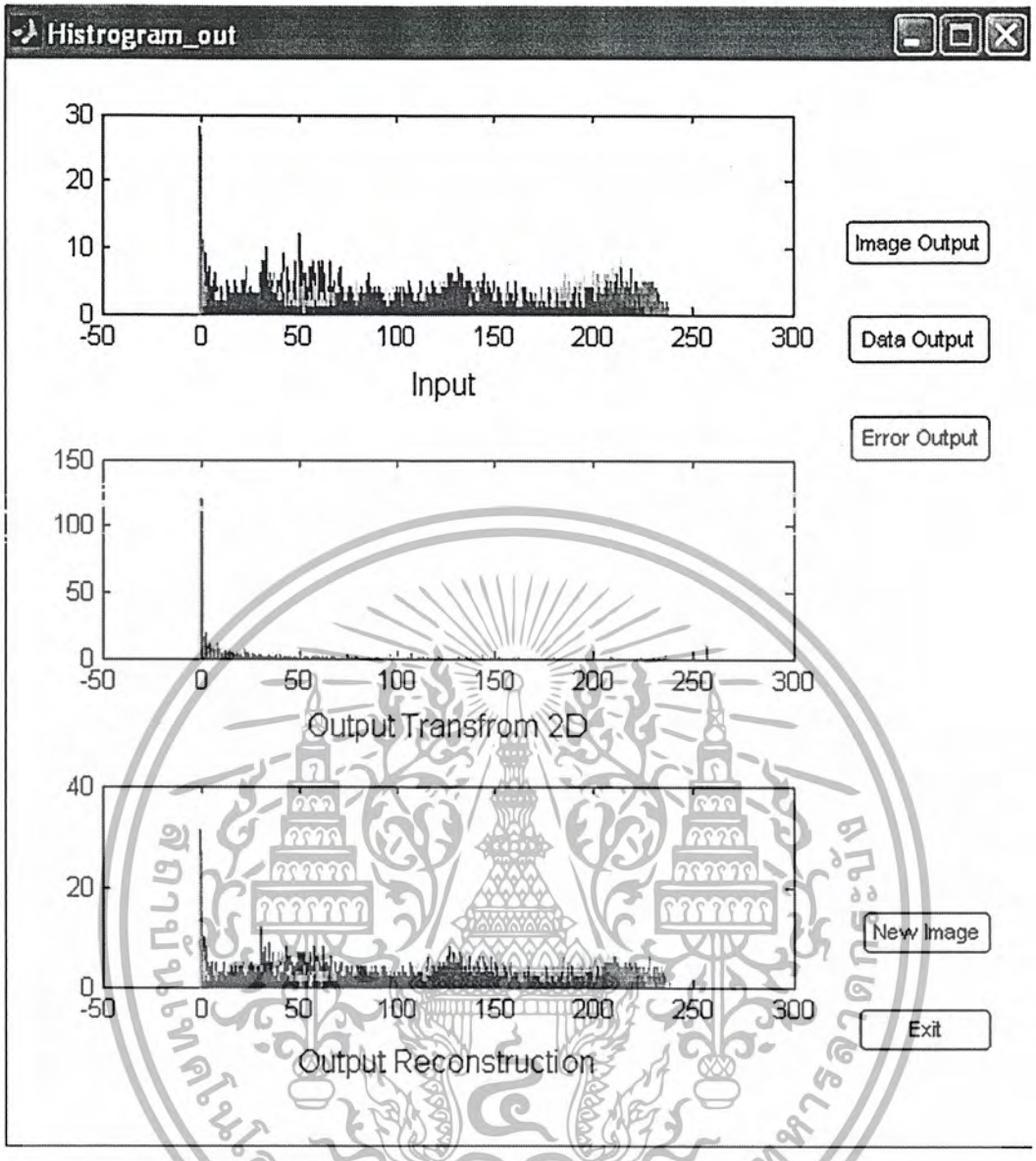


รูปที่ 4.156 แสดงค่าข้อมูลอินพุทเปรียบเทียบกับข้อมูลเอาท์พุทของสัญญาณภาพเป็นเลขฐาน 16



รูปที่ 4.157 แสดงค่าข้อมูลที่ได้จากโปรแกรม MATLAB เปรียบเทียบกับการทำงานจริงบนบอร์ด FPGA

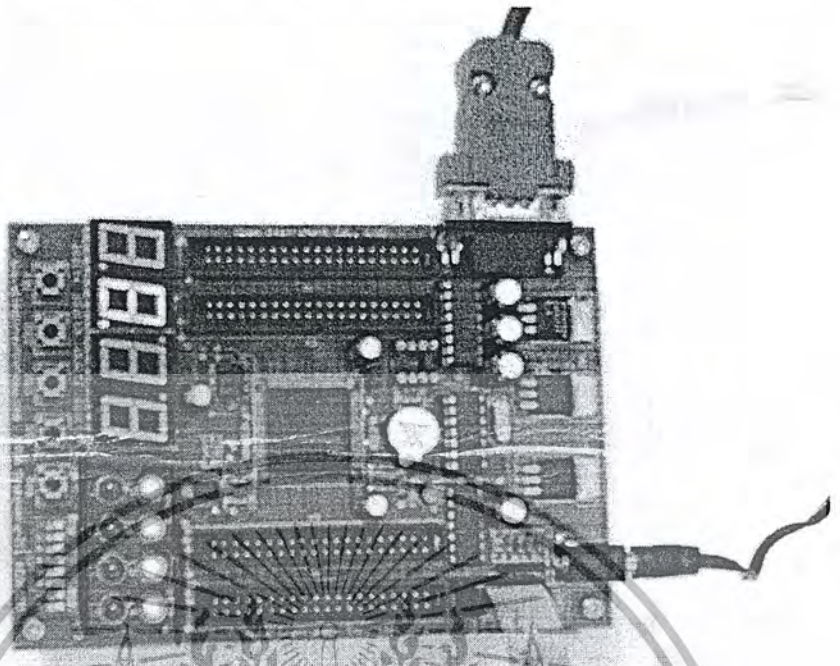
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



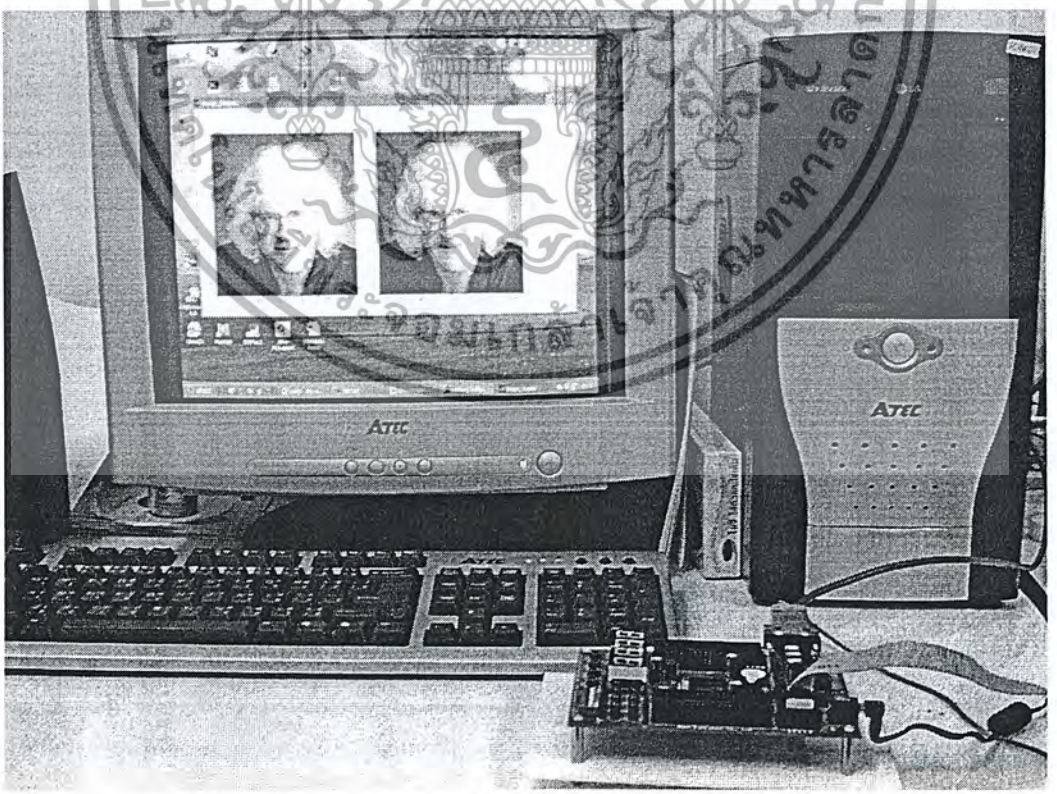
รูปที่ 4.158 ฮิสโตแกรมเปรียบเทียบข้อมูลอินพุท ข้อมูลที่ได้จากการแปลง และข้อมูลในการสร้างกลับของสัญญาณภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.9 ภาพอุปกรณ์ที่ใช้ในการทดสอบการแปลงดิสครีตโคซายน์ 2 มิติ



รูปที่ 4.159 แสดงอุปกรณ์ที่ใช้ในการทดสอบการแปลงดิสครีตโคซายน์แบบ 2 มิติ
ที่ใช้หลักการการแยกการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง



รูปที่ 4.160 แสดงบอร์ด FPGA ที่ทำการเชื่อมต่อกับพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

จากการศึกษาในทางทฤษฎีและทำการทดลองตามขอบเขตของปริญญานิพนธ์นี้สามารถที่สรุปความสำคัญของเนื้อหาแต่ละส่วนได้เป็นดังนี้

ปริญญานิพนธ์ฉบับนี้ได้ทำการศึกษาหลักการบีบอัดข้อมูลภาพขาวดำ โดยใช้กระบวนการแปลงแบบดิสครีตโคซายน์แบบ 2 มิติ โดยอาศัยหลักการของการแยกการแปลงแบบดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง โดยเขียนบรรยายพฤติกรรมด้วยภาษา วีเอชดีแอล ในการออกแบบ พร้อมทั้งทำการประมวลผลการทำงาน Hard ware

ในส่วนของการแปลงดิสครีตโคซายน์ 2 มิติ ใช้หลักการของการแยกการแปลงดิสครีตโคซายน์โดยใช้หลักการการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง ซึ่งมีหลักการการทำงานดังนี้ จะนำข้อมูลทั้ง 8 ค่า ตามแนวนอนเข้าทำการแปลงดิสครีตโคซายน์แบบ 1 มิติ ในส่วนที่ 1 จนข้อมูลครบทั้งบล็อก (64 ค่า) จากนั้นนำผลที่ได้จากการแปลงดิสครีตโคซายน์แบบ 1 มิติ ส่งเข้าไปยังส่วนการทรานสโพส เพื่อทำการเปลี่ยนตำแหน่งของข้อมูล จากแถวเป็นหลัก และจากหลักเป็นแถว เมื่อทำการทรานสโพสเรียบร้อยแล้วก็จะส่งข้อมูลที่ได้ตามแนวตั้งไปยังการแปลงดิสครีตโคซายน์แบบ 1 มิติ ในส่วนที่ 2 จนครบทั้ง 64 ค่าก็จะได้ข้อมูลที่ได้จากการแปลงดิสครีตโคซายน์ 2 มิติครบ 1 บล็อก

ในส่วนของการแปลงดิสครีสโคซายน์จะใช้หลักการของ CORDIC Algorithm และ DA มาทำการสร้างส่วนของฮาร์ดแวร์เพื่อช่วยลดจำนวนอุปกรณ์ในการสร้างลง อีกทั้งในส่วนของการทรานสโพสจะใช้แรมเข้ามาช่วยเก็บค่าของข้อมูลและเรียกค่าข้อมูลตามที่ต้องการ

ในส่วนของการจำลองการทำงาน (Simulation) ด้วยโปรแกรม MATLAB ซึ่งผลที่ได้เป็นไปตามทฤษฎีโดยที่ความถูกต้องแม่นยำจะขึ้นอยู่กับการเก็บค่าความละเอียดจากการคำนวณแต่ละรอบและจำนวนรอบที่ใช้ในการคำนวณ (การหมุนมุม) ส่วนของการแปลงดิสครีตโคซายน์ที่สร้างโดยใช้ภาษาวีเอชดีแอล เมื่อทำการจำลองการทำงานดูผลที่ได้จากโปรแกรม Xilinx-Project Navigator ผลการทดลองที่ได้มีค่าที่ใกล้เคียงกับค่าที่คำนวณได้ทางทฤษฎี อีกทั้งความผิดพลาด (Error) ที่เกิดขึ้นอันเนื่องมาจากชุดของข้อมูล และค่าสัมประสิทธิ์ที่ใช้ในการคำนวณที่ถูกเก็บค่าไว้ในรอม (ROM) จะถูกแปลงเป็นเลขไบนารี (Binary) ที่มีค่าจำนวนบิทจำกัด ทำให้เกิดค่าผิดพลาดขึ้น

ในส่วนของการประมวลผลการทำงานจริงจะทำการส่งข้อมูลภาพ จากคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมไปยัง Hard ware ซึ่งจะทำการแปลงข้อมูลด้วยหลักการการแปลงดิสครีตโคซายน์แบบ 2 มิติ จากนั้นส่งข้อมูลที่ทำการประมวลผลได้ มาทำการแปลงดิสครีตโคซายน์กลับผ่านทางโปรแกรมแมทแลบ พร้อมทั้งแสดงผลการทดลองผ่านทาง GUI (Graphic User Interface) โดยทำการแสดงข้อมูลภาพต้นแบบ ข้อมูลภาพที่สร้างคืนมาได้ ผลการแปลงดิสครีตโคซายน์ 2 มิติ ค่าความผิดพลาดที่เกิดขึ้นจากการทำงานจริง และแสดงฮิสโตแกรมของข้อมูลภาพต้นแบบ ข้อมูลที่ผ่านการแปลง และข้อมูลภาพที่สร้างกลับ ผ่านทางโปรแกรมแมทแลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานฉบับนี้สำเร็จลุล่วงได้ด้วยดีโดยได้รับความช่วยเหลือ และชี้แนะจากหลายท่าน ผู้จัดทำขอขอบพระคุณอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ อัครพล ตรีรัตน์ ที่ให้คำปรึกษาและความช่วยเหลือด้านข้อมูล และอุปกรณ์ในการทำปริญญานิพนธ์ ขอขอบคุณอาจารย์ศรวัฒน์ ชิวปรีชา ที่ให้คำปรึกษาและให้ความช่วยเหลือในด้านการใช้งานเอฟพีจีเอ และขอขอบคุณนายนริศร รณรงค์ฤทธิ์ ที่ให้คำปรึกษาและช่วยเหลือด้านข้อมูลมาโดยตลอด ผู้เขียนพึงระลึกอยู่เสมอว่าปริญญานิพนธ์ฉบับนี้จะไม่สามารถสำเร็จลงได้เลย หากขาดความช่วยเหลือจากทุกท่านจึงขอขอบพระคุณมาอย่างสูง

นายกฤช เรืองฤทธิ์

นายกฤติกานต์ ยมภักดี

นายจักร แซ่เอา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

[1] ชำนาญ ปัญญาใส, วัชรากร หนูทอง "ภาษา VHDL สำหรับการออกแบบวงจรดิจิตอล", ซีเอ็ดยูเคชั่น, กรุงเทพฯ 2547

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

[3] K. R. Roa and J. J. Hwang , "Techniques and Standards for Image , Video and Audio Coding", Prentice Hall PRT, Upper Saddle River , New Jersey 07458, 1996

[4] Z.Mohd—Yusof, I.Suleiman, Z.Asper, "Implementation of Two Dimensional Forward DCT And Inverse DCT Using FPGA", IEEE. 2001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้