บทที่ 5

ผลทดสอบและผลวิเคราะห์วงจรหาขอบภาพตามแนวทางวิธีของ Canny

วงจรหาขอบภาพที่ได้ออกแบบไว้ในบทที่ 4 ทั้ง 4 วงจรคือ

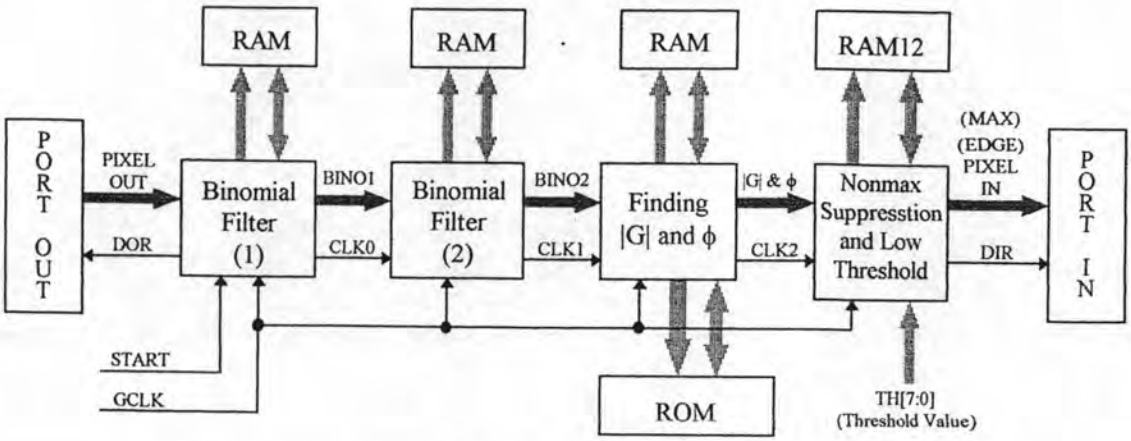
1. วงจร Binomial Filter (1)
2. วงจร Binomial Filter (2)
3. วงจร Finding |G| and ϕ
4. วงจร Nonmax Suppression and Low Threshold

จะต้องได้รับการยืนยันว่าทำงานถูกต้อง โดยทำการทดสอบด้วยการจำลองบนเครื่องคอมพิวเตอร์ และทำการเปรียบเทียบผลลัพธ์ที่ได้กับผลลัพธ์ที่ถูกต้อง ซึ่งจะใช้ซอฟต์แวร์เป็นตัวคำนวณหาผลลัพธ์ที่ถูกต้อง

เมื่อวงจรที่ออกแบบสามารถทำงานได้ถูกต้อง ก็นำไปทำการต่อเป็นฮาร์ดแวร์ โดยใช้ FPGA เป็นอุปกรณ์หลักในการต่อวงจร ร่วมกับหน่วยความจำ และต้องบรรจุข้อมูลที่ได้แปลงจากแฟ้มข้อมูลของวงจรทั้ง 4 ลงไปใน FPGA แต่ละตัวเพื่อโปรแกรมให้ FPGA ทำงานตามหน้าที่ของวงจร โดย FPGA หนึ่งตัวจะทำงานของวงจรหนึ่งวงจร ยกเว้นวงจร Finding |G| and ϕ ที่ต้องใช้ FPGA 2 ตัวในการทำงานของวงจรนี้ แล้วก็ทำการทดสอบและเปรียบเทียบผลของฮาร์ดแวร์ที่ได้ว่าเป็นไปตามผลที่ได้จากการจำลองหรือไม่

วิธีทดสอบวงจร

หลังจากออกแบบวงจร ก็ทำการจำลอง(simulation)เพื่อทดสอบวงจร โดยใช้โปรแกรม Workview PLUS ของบริษัท Viewlogic Systems ในการเขียนวงจรและทดสอบวงจรทั้ง 4 (เกตทางตรรกใช้ไลบรารีของ Xilinx ทั้งหมด เพื่อให้สามารถแปลงลง FPGA ได้ในภายหลัง) ซึ่งการต่อวงจรเป็นดังรูปที่ 5.1 ซึ่งเป็นการทดสอบวงจรรวม



รูปที่ 5.1 แสดงการต่อวงจร เพื่อให้สามารถทำการจำลองบนคอมพิวเตอร์ได้

ซึ่งจำเป็นต้องมีอุปกรณ์ต่อเพิ่มเติมภายนอก ดังแสดงในรูป เขียนแทนด้วยภาษา VHDL เพื่อให้สามารถทำการจำลองได้ โดยมีอุปกรณ์เพิ่มเติมดังนี้

1. *RAM ขนาด 1024 ไบต์* สำหรับเก็บข้อมูลรูปภาพจำนวน 3 แถว ซึ่งวงจร Binomial Filter (1), วงจร Binomial Filter (2) และวงจร Finding |G| and ϕ ใช้ RAM นี้วงจรละหนึ่งตัว
2. *RAM12 ขนาด 1024 ค่า ค่าละ 12 บิต* สำหรับเก็บข้อมูลรูปภาพจำนวน 3 แถว ซึ่งวงจร Nonmax Suppression ต้องใช้เพื่อเก็บค่า |G| ขนาด 9 บิต และค่า ϕ ขนาด 3 บิต ที่ได้จากวงจร Finding |G| and ϕ
3. *ROM ขนาด 65536 ค่า ค่าละ 11 บิต* ใช้เป็นตารางเปิดดู โดยเก็บค่า |G| ขนาด 9 บิต และค่า ϕ ขนาด 2 บิต ซึ่งเป็นไปได้ทั้งหมด 65536 ค่า เพราะค่า Gx และ Gy มีค่าได้ตั้งแต่ 0 ถึง 255 และค่า ϕ มีขนาด 2 บิต เนื่องจากใช้ค่าสัมบูรณ์ของ Gx และ Gy มาอ่านค่า ϕ จาก ROM คือเก็บเฉพาะค่าในควอดแรนต์ที่ 1 แล้วค่อยนำไปแปลงควอดแรนต์ตามค่าบวกลบของ Gx และ Gy
4. *PORT_OUT* เป็นพอร์ตขนาด 8 บิต สำหรับส่งข้อมูลให้วงจร อ่านข้อมูลตัวเลขแอสกีจากแฟ้มข้อมูลชื่อ "FILE_OUT.DAT" โดยหนึ่งบรรทัดแทนหนึ่งข้อมูล แล้วแปลงเป็นรหัสฐานสอง ส่งออกเอาต์พุต เมื่อได้รับสัญญาณขอบขาขึ้นจากขา DOR ของวงจร Binomial Filter (1) เหตุที่กำหนดอุปกรณ์นี้เป็นเอาต์พุต เพราะมองว่าเป็นพอร์ตเอาต์พุตของแหล่งกำเนิดข้อมูล เช่น พอร์ตคอมพิวเตอร์, พอร์ตของกล้องดิจิตอล, ฯลฯ เป็นต้น
5. *PORT_IN* เป็นพอร์ตขนาด 8 บิต สำหรับรับข้อมูลจากวงจร จะอ่านข้อมูลรหัสฐานสองจากอินพุต แล้วแปลงเป็นตัวเลขแอสกีเขียนลงในแฟ้มข้อมูลชื่อ "FILE_IN.DAT"

โดยหนึ่งบรรทัดแทนหนึ่งข้อมูล เมื่อได้รับสัญญาณขอบขาขึ้นจากขา DIR ของวงจร Nonmax Suppression and Low Threshold เหตุที่กำหนดอุปกรณ์นี้เป็นอินพุตเพราะมองว่าเป็นพอร์ตอินพุตของปลายทาง

ก่อนเริ่มการจำลอง ต้องมีการแปลงแฟ้มข้อมูลข้อมูลรูปภาพอินพุตให้เป็นแฟ้มข้อมูลแอสกีชื่อ "FILE_OUT.DAT" ก่อน แล้วจึงเริ่มการจำลอง โดยตั้งค่าเริ่มต้นให้กับวงจร กำหนดค่าขีดเริ่มเปลี่ยน แล้วให้สัญญาณ '0' ที่ขา START และให้สัญญาณนาฬิกาเข้าที่ขา GCLK อย่างน้อยหนึ่งสัญญาณนาฬิกาเพื่อให้วงจรทำการตั้งใหม่ตัวเอง จากนั้นก็สั่งให้วงจรเริ่มทำงาน โดยเปลี่ยนสัญญาณที่ขา START เป็น '1' และส่งสัญญาณนาฬิกาเข้าไปให้วงจรอีกจนกว่าวงจรจะทำงานเสร็จ ซึ่งต้องใช้จำนวนสัญญาณนาฬิกาเป็นจำนวนไม่ต่ำกว่า 1,073,374 สัญญาณนาฬิกา ค่านี้ได้จากจำนวนข้อมูลที่วงจรที่ 3 ต้องคำนวณทั้งหมด(258 แถว แถวละ 257 คอลัมน์ คอลัมน์ละ 16 สัญญาณนาฬิกา) บวกกับค่าล้าหลังที่วงจรที่ 3 วงจร Nonmax Suppression and Low Threshold ต้องเริ่มหลังวงจรที่ 0 วงจร Binomial Filter (1) (เป็นจำนวน 12,461 สัญญาณนาฬิกา) บวกด้วยจำนวนสัญญาณนาฬิกาที่วงจรที่ 3 ใช้เริ่มทำงานจนได้เอาต์พุตออกมา(เป็นจำนวน 17 สัญญาณนาฬิกา)

เมื่อวงจรทำงานเสร็จสิ้น และออกจากการจำลองแล้ว ก็จะได้แฟ้มข้อมูลแอสกีชื่อ "FILE_IN.DAT" เก็บผลลัพธ์ที่ได้จากการจำลอง แล้วจึงแปลงแฟ้มข้อมูลนี้ไปเป็นแฟ้มข้อมูลรหัสฐานสองต่อไป แล้วนำไปเปรียบเทียบกับซอฟต์แวร์ที่เขียนให้มีวิธีการคำนวณหาขอบภาพที่เหมือนกับวงจรที่ออกแบบ เพื่อใช้ผลลัพธ์ที่ได้จากซอฟต์แวร์เป็นตัวเปรียบเทียบว่าวงจรที่ออกแบบทำงานได้ถูกต้องหรือไม่

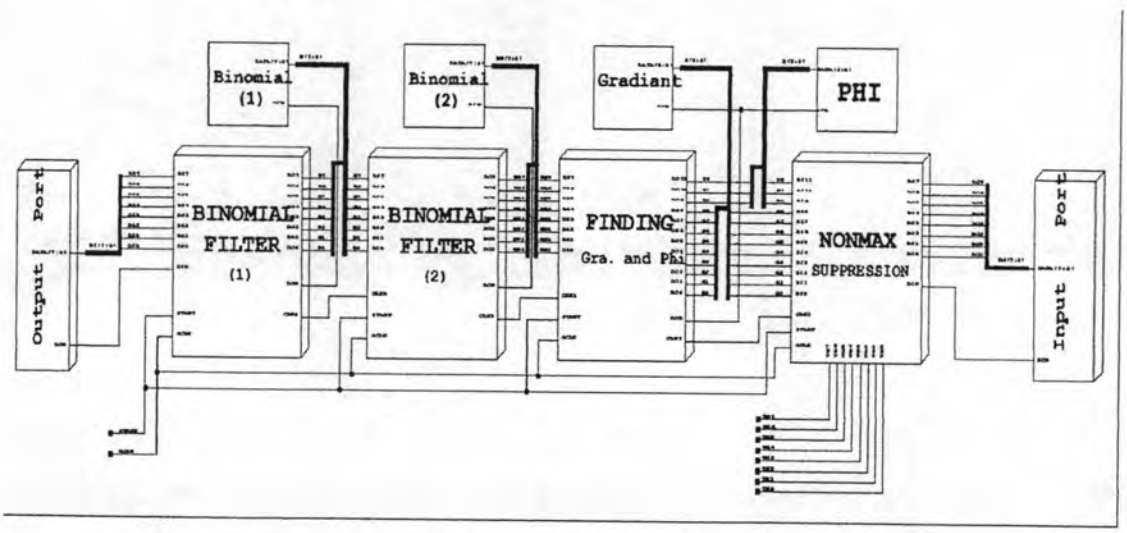
และเพื่อเป็นการยืนยันว่าแต่ละวงจรทำงานได้ถูกต้อง จึงทำการทดสอบทีละวงจร ซึ่งก็จะมีวิธีการทดสอบโดยใช้การจำลองเช่นเดียวกับวิธีการทดสอบวงจรรวม

การทดสอบทีละวงจร

ทำการต่อวงจรเพื่อทดสอบทีละวงจร ซึ่งถ้านำมารวมกันจะได้เป็นดังรูปที่ 5.2 ข้างล่าง โดย แรม และ รอม จะรวมอยู่ในบล็อกของแต่ละวงจรแล้ว ซึ่งเป็นรูปจริงที่ใช้ในการจำลองบนโปรแกรม Workview PLUS โดยมีส่วนประกอบที่เพิ่มเติมมาคือ

1. *BINO1* เป็นพอร์ตขนาด 8 บิต ทำหน้าที่เหมือน PORT_IN แต่ต่อไว้สำหรับรับข้อมูลจากวงจร Binomial Filter (1) แล้วเก็บข้อมูลลงในแฟ้มข้อมูลชื่อ "BINO1.DAT" เมื่อได้รับสัญญาณขอบขาขึ้นจากขา DIR ของวงจร

2. *BINO2* เป็นพอร์ตขนาด 8 บิต ทำหน้าที่เหมือน PORT_IN แต่ต่อไว้สำหรับรับข้อมูลจากวงจร Binomial Filter (2) แล้วเก็บข้อมูลลงในแฟ้มข้อมูลชื่อ "BINO2.DAT" เมื่อได้รับสัญญาณขอบขาขึ้นจากขา DIR ของวงจร
3. *GRA* เป็นพอร์ตขนาด 9 บิต ทำหน้าที่เหมือน PORT_IN แต่ต่อไว้สำหรับรับข้อมูลจากวงจร Finding |G| and ϕ แล้วเก็บข้อมูลลงในแฟ้มข้อมูลชื่อ "GRA.DAT" เมื่อได้รับสัญญาณขอบขาขึ้นจากขา DIR ของวงจร
4. *PHI* เป็นพอร์ตขนาด 3 บิต ทำหน้าที่เหมือน PORT_IN แต่ต่อไว้สำหรับรับข้อมูลจากวงจร Finding |G| and ϕ แล้วเก็บข้อมูลลงในแฟ้มข้อมูลชื่อ "PHI.DAT" เมื่อได้รับสัญญาณขอบขาขึ้นจากขา DIR ของวงจร



รูปที่ 5.2 แสดงการทดสอบวงจร โดยการจำลองบนโปรแกรม Workview PLUS

ผลทดสอบวงจร Binomial Filter (1) และ (2)

เอาต์พุตที่ได้จากวงจรจะเก็บไว้ในแฟ้มข้อมูล "BINO1.DAT" เมื่อนำมาแปลงกลับเป็นแฟ้มข้อมูลรูปภาพจะได้ดังรูปที่ 5.3 ด้านขวา โดยแสดงเปรียบเทียบกับแฟ้มข้อมูลอินพุตในรูปที่ 5.3 ด้านซ้าย

รูปที่ 5.3 แสดงภาพอินพุต* กับภาพเอาต์พุตที่ได้จากการจำลองวงจร Binomial Filter (1)

จากรูปจะเห็นได้ว่าวงจร Binomial Filter (1) ที่ออกแบบสามารถทำการเกลี่ยภาพได้ โดยภาพเอาต์พุตนี้ เมื่อเปรียบเทียบกับภาพเอาต์พุตที่ได้จากการคำนวณด้วยซอฟต์แวร์ จะได้ผลลัพธ์ตรงกัน

ส่วนวงจร Binomial Filter (2) จะทำหน้าที่เกลี่ยภาพเหมือนวงจร Binomial Filter (1) ดังนั้นการทำงานจึงเหมือนกัน เอาต์พุตที่ได้จากวงจรจะเก็บไว้ในแฟ้มข้อมูล "BINO2.DAT" เมื่อนำมาแปลงกลับเป็นแฟ้มข้อมูลรูปภาพจะได้ดังรูปข้างล่างด้านขวา โดยแสดงเปรียบเทียบกับแฟ้มข้อมูล "BINO1.DAT" ในรูปด้านซ้าย

รูปที่ 5.4 แสดงภาพอินพุตกับภาพเอาต์พุตที่ได้จากการจำลองวงจร Binomial Filter (1) และ (2)

---

* ภาพอินพุต มีที่มาจากไฟล์รูปภาพที่มากับหนังสือของ Sid-Ahmed (1995)

จากรูปจะเห็นว่าเมื่อวงจร Binomial Filter (1) และ (2) ทำงานต่อเรียงกัน จะทำให้เกลี่ยภาพลงได้มากขึ้น(ภาพมัวขึ้น) โดยภาพเอาต์พุตนี้ เมื่อเปรียบเทียบกับภาพเอาต์พุตที่ได้จากการคำนวณด้วยซอฟต์แวร์ จะได้ผลลัพธ์ตรงกัน

ผลทดสอบวงจร Finding |G| and ϕ

อินพุตของวงจรเป็นแฟ้มข้อมูลรูปภาพที่ผ่านวงจร Binomial Filter (1) และ (2) มาแล้ว และเอาต์พุตที่ได้จากวงจรจะมีสองค่าคือค่า |G| และ ϕ แยกกันเก็บไว้ในแฟ้มข้อมูล "GRA.DAT" และ "PHI.DAT" ตามลำดับ เมื่อแฟ้มข้อมูล "GRA.DAT" มาแปลงกลับเป็นแฟ้มข้อมูลรูปภาพ, ทำการผกผัน(invert)รูปภาพคือกลับขาวกับดำ เพื่อพิมพ์ลงบนกระดาษได้สะดวก และทำการปรับเท่า(equalize) เพื่อให้เห็นลายเส้นของภาพได้ชัดเจน จะได้ดังรูปข้างล่างด้านซ้าย และแฟ้มข้อมูล "PHI.DAT" เมื่อแปลงเป็นรูปภาพแล้วทำการปรับเท่า เพื่อแปลงค่าทิศทางจาก 0 ถึง 7 ขยายไปเป็น 0 ถึง 255 เพื่อเน้นให้เห็นค่าแทนทิศทางทั้ง 8 ค่า จะได้ดังรูปด้านขวา

รูปที่ 5.5 แสดงภาพอินพุตกับภาพเอาต์พุตที่ได้จากการจำลองวงจร Finding |G| and ϕ

ในรูปภาพที่แสดงค่า |G| จะมองเห็นขอบภาพได้ชัดเจน ส่วนรูปภาพที่แสดงค่า ϕ ซึ่งเป็นทิศทางของ |G| จะมองไม่ออก เพราะเป็นการแทนทิศทางด้วยค่าที่กำหนดขึ้น ซึ่งมีทั้งหมด 8 ค่าด้วยกัน โดยภาพเอาต์พุตทั้งสองนี้ เมื่อเปรียบเทียบกับภาพเอาต์พุตที่ได้จากการคำนวณด้วยซอฟต์แวร์ จะได้ผลลัพธ์ที่ตรงกัน

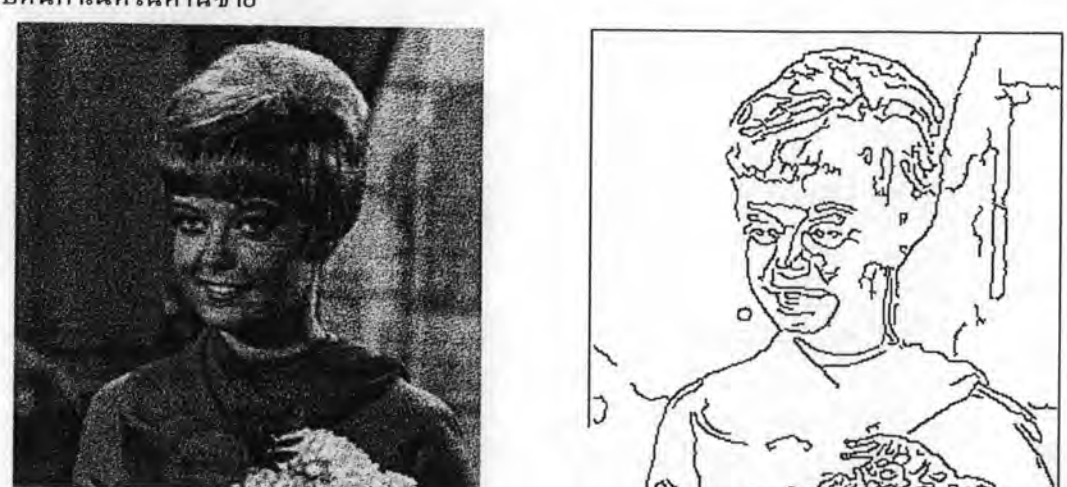
ผลทดสอบวงจร Nonmax Suppression and Low Threshold

ต้องมีการตั้งค่าขีดเริ่มเปลี่ยนค่าต่ำจากแหล่งกำเนิดภายนอก เอาต์พุตที่ได้จากวงจรจะเก็บไว้ในแฟ้มข้อมูล "FILE_IN.DAT" เมื่อนำมาแปลงกลับเป็นแฟ้มข้อมูลรูปภาพจะได้ดังรูปที่ 5.6 ด้านขวา โดยแสดงเปรียบเทียบกับแฟ้มข้อมูล "GRA.DAT" ในรูปที่ 5.6 ด้านซ้าย

รูปที่ 5.6 แสดงภาพอินพุตกับภาพเอาต์พุตที่ได้จากการจำลองวงจร Nonmax Suppression and Low Threshold

ผลที่ได้จากวงจรนี้จะเป็นรูปขอบภาพ ที่ผ่านการตรวจสอบขีดเริ่มเปลี่ยนค่าต่ำแล้ว ซึ่งในรูปนี้ตั้งค่าขีดเริ่มเปลี่ยนไว้ที่ '0' โดยภาพเอาต์พุตนี้ เมื่อเปรียบเทียบกับภาพเอาต์พุตที่ได้จากการคำนวณด้วยซอฟต์แวร์ จะได้ผลลัพธ์ตรงกัน

หลังจากนั้นก็นำรูปภาพนี้ไปผ่านซอฟต์แวร์ตรวจสอบขีดเริ่มเปลี่ยนค่าสูงแบบฮิสเตอร์เรซิส เพื่อเปลี่ยนให้เป็นรูปขอบภาพ ดังตัวอย่างรูปที่ 5.7 ด้านขวาเป็นรูปที่ผ่านการตรวจสอบขีดเริ่มเปลี่ยนค่าต่ำและสูงแล้ว (ขีดเริ่มเปลี่ยนค่าต่ำ '10 ' และค่าสูง '20 ') โดยเปรียบเทียบกับรูปต้นกำเนิดในด้านซ้าย

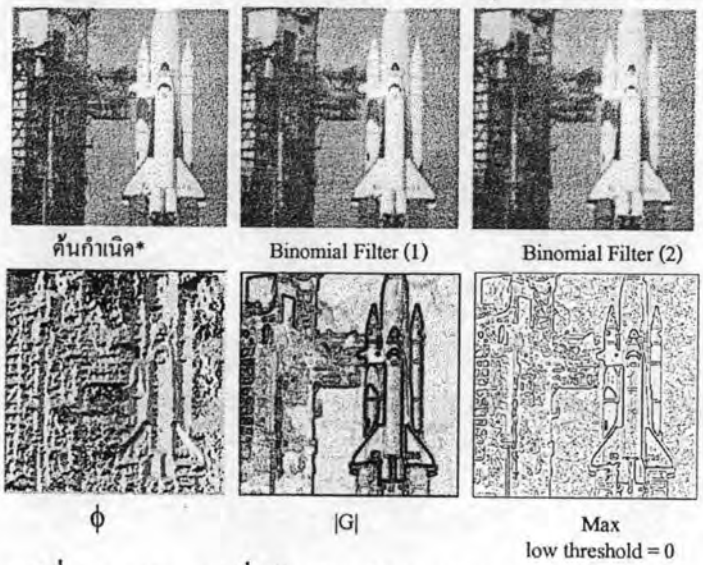


รูปที่ 5.7 แสดงรูปภาพที่ต้องการหาขอบภาพกับรูปขอบภาพที่ได้

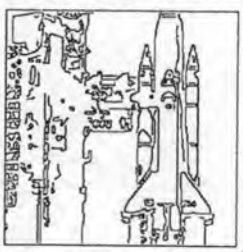
ผลการจำลองวงจรรวม

การจำลองวงจรรวม ต้องใช้สัญญาณนาฬิกาในการทำงานทั้งหมดไม่น้อยกว่า 1,073,374 + 1 สัญญาณนาฬิกา(เพื่อการตั้งใหม่ก่อนทำงาน) ต่อการทำงานเพื่อหาขอบภาพจากรูปภาพ(ขนาด 256 x 256 จุดภาพ)หนึ่งรูป ซึ่งการจำลองความเร็วของวงจรจะขึ้นอยู่กับความเร็วของสัญญาณนาฬิกา เช่น ถ้าเราต้องการให้วงจรสามารถทำงานได้ 30 รูปภาพต่อวินาที ก็ต้องใช้สัญญาณนาฬิกาที่ความเร็ว 30 x 1,073,375 เฮิรตซ์ หรือประมาณ 32.20 เมกะเฮิรตซ์ เป็นต้น

ส่วนเวลาที่ใช้ในการจำลองการทำงานของวงจรบนเครื่องเพนเทียมความเร็วซีพียู 133 เมกะเฮิรตซ์ หน่วยความจำ 16 เมกะไบต์ ใช้เวลาในการจำลองประมาณ 8 ชั่วโมง ดังนั้นในงานวิจัยนี้จึงทำการจำลองเพียง 2 รูป เพราะผลที่ได้เหมือนกับซอฟต์แวร์ จึงใช้ผลซอฟต์แวร์แทนผล การจำลองได้ ซึ่งรูปแรกคือรูปที่ใช้ประกอบคำอธิบายในหัวข้อ *"การทดสอบทีละวงจร"* และรูปที่สองแสดงในรูปที่ 5.8 และเมื่อผ่านการตรวจสอบขีดเริ่มเปลี่ยนค่าสูงด้วยซอฟต์แวร์จะได้ดังรูปที่ 5.9



<u>รูปที่ 5.8</u> แสดงภาพที่ได้จากการจำลองบน Workview PLUS



<u>รูปที่ 5.9</u> ภาพที่ได้หลังผ่านซอฟต์แวร์ตรวจสอบขีดเริ่มเปลี่ยนค่าสูง โดยตั้งไว้ที่ '20'

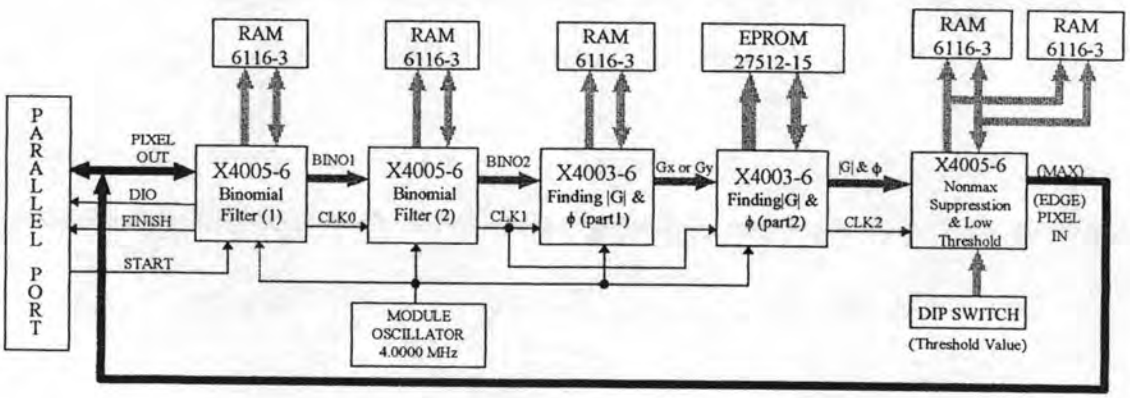
---

* ภาพต้นกำเนิดมีที่มาจากไฟล์รูปภาพที่มากับหนังสือของ Baxes (1994) นำมาแปลงเป็นขนาด 256 x 256 จุดภาพ

การประกอบฮาร์ดแวร์

การต่อฮาร์ดแวร์จริง จำเป็นต้องแบ่งวงจรรวมที่ประกอบไปด้วยวงจรทั้ง 4 ออก เพื่อให้สามารถแปลงลง FPGA ได้ ซึ่ง FPGA ที่ใช้เป็นของบริษัท Xilinx โดยมีเบอร์ให้ใช้คือ เบอร์ XC4003-6 และ XC4005-6 (ขีด 6 ใช้บอกความเร็ว(speed grade) ตัวเลขยิ่งน้อยยิ่งเร็ว) ซึ่งจากการใช้โปรแกรม XDM ของ Xilinx เพื่อทำการแปลงแฟ้มข้อมูลของวงจรที่ได้จากโปรแกรม Workview PLUS ให้เป็นแฟ้มข้อมูลของ Xilinx พบว่าวงจร Binomial Filter (1), (2) และวงจร Nonmax Suppression and Low Threshold สามารถใส่ลงไปในเบอร์ XC4005 ได้(วงจรละตัว) แต่วงจร Finding |G| and ϕ ไม่สามารถใส่ลงไปได้ เนื่องจากใช้จำนวนขามากเกินไป ทำให้ต้องทำการแบ่งวงจรนี้ออกเป็น 2 วงจร กำหนดเป็นวงจรที่ 2_1 และ 2_2 โดยแบ่งที่ตำแหน่งดังที่แสดงในรูปที่ 5.11 ในหน้าที่ 67 ทำให้เมื่อแปลงแฟ้มข้อมูลแล้ววงจรที่ 2_1 และ 2_2 สามารถใส่ลงไปในเบอร์ XC4003 ได้(วงจรละตัว)

ดังนั้นการต่อฮาร์ดแวร์จึงต้องใช้ FPGA ทั้งหมด 5 ตัว ร่วมกับหน่วยความจำทั้ง แรม และอีพรอม และอุปกรณ์อื่นๆ ซึ่งแสดงการต่อฮาร์ดแวร์ด้วยแผนภาพบล็อกได้ดังรูปที่ 5.10



รูปที่ 5.10 แสดงแผนภาพบล็อกของฮาร์ดแวร์จริง

ซึ่งข้อมูลเกี่ยวกับจำนวนเกตทางตรรกและเวลาการประวิง(delay time) ที่ได้จากการรายงานผลของโปรแกรม XDM สามารถสรุปได้ดังตารางที่ 5.1, 5.2 และ 5.3 ในหน้าถัดไป โดยค่าในตารางที่ 3 จะเป็นค่าที่บอกให้ทราบว่าความเร็วสัญญาณนาฬิกาสูงสุดที่สามารถป้อนให้ฮาร์ดแวร์มีค่าประมาณ 3.7 เมกะเฮิรตซ์

ตารางที่ 5.1 ตารางแสดงวงจรที่ใช้ FPGA เบอร์ XC4005

	XC4005 หนึ่งตัวมี	วงจร Binomial Filter (1) ใช้	วงจร Binomial Filter (2) ใช้	วงจร Nonmax Suppresion and Low Threshold ใช้
เกตทางตรรก	5,000	1,923	2,077	2,776
CLB	196	148	150	186
ขา I/O	61	42	41	56

ตารางที่ 5.2 ตารางแสดงวงจรที่ใช้ FPGA เบอร์ XC4003

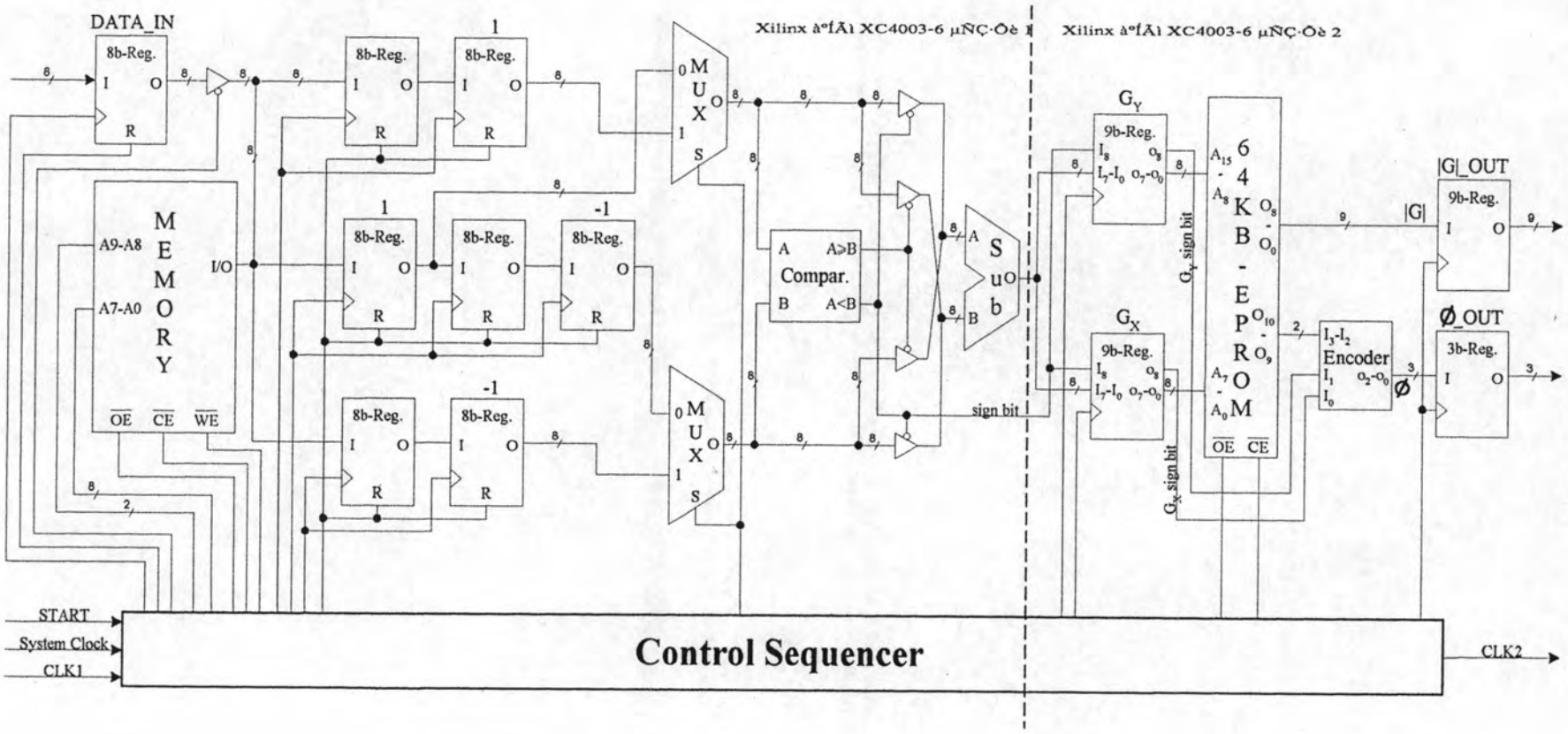
	XC4003 หนึ่งตัวมี	วงจร Finding $\|G\|$ and ϕ วงจรที่ 2_1	วงจร Finding $\|G\|$ and ϕ วงจรที่ 2_2
เกตทางตรรก	3,000	1,650	828
CLB	100	98	56
ขา I/O	61	42	53

ตารางที่ 5.3* ตารางแสดงเวลาการประวิงของแต่ละวงจร

วงจร	Binomial Filter (1)	Binomial Filter (2)	Finding $\|G\|$ and ϕ วงจรที่ 2_1	Finding $\|G\|$ and ϕ วงจรที่ 2_2	Nonmax Suppresion and Low Threshold
คาบเวลาสัญญาณนาฬิกาต่ำสุด	196.4 นาโนวินาที	199.8 นาโนวินาที	228.0 นาโนวินาที	74.4 นาโนวินาที	269.5 นาโนวินาที
ความเร็วสัญญาณนาฬิกาสูงสุด	5.1 เมกะเฮิรตซ์	5.0 เมกะเฮิรตซ์	4.4 เมกะเฮิรตซ์	13.4 เมกะเฮิรตซ์	3.7 เมกะเฮิรตซ์

---

* ค่าของตาราง 5.3 ได้มาจากการใช้ xdelay (เป็นทางเลือกในโปรแกรม XDM ของ Xilinx) มาคำนวณหาเวลาการประวิง



รูปที่ 5.11 แสดงตำแหน่งที่ใช้แบ่งวงจร Finding |G| and ϕ ออกเป็น 2 ส่วน
เพื่อให้สามารถแปลงใส่ลงไปใน Xilinx เบอร์ XC4003-6 ได้

การทดสอบฮาร์ดแวร์

การทดสอบจะใช้พอร์ตขนานของคอมพิวเตอร์หนึ่งพอร์ต เพื่อใช้รับและส่งข้อมูลให้ฮาร์ดแวร์ โดยคอมพิวเตอร์จะเป็นทั้งต้นกำเนิดและปลายทางของข้อมูล โปรแกรมมีการทำงานตามรหัสเทียม(pseudo code)ดังต่อไปนี้

```
เตรียมทรัพยากร และ จองหน่วยความจำเพื่อเก็บข้อมูลรูปภาพ
เปิดแฟ้มข้อมูลรูปภาพต้นกำเนิดและสร้างแฟ้มข้อมูลปลายทาง
โหลดข้อมูลรูปภาพไว้ในหน่วยความจำและปิดแฟ้มข้อมูลต้นกำเนิด
ห้ามทำงานการขัดจังหวะ(disable interrupt)
ส่งสัญญาณ START เป็น '0' (เพื่อทำการตั้งใหม่ฮาร์ดแวร์)
ส่งจุดภาพ(ข้อมูล 1 ไบต์)แรกออกสู่พอร์ต
สั่งสัญญาณ START ให้เป็น '1' เพื่อสั่งให้ฮาร์ดแวร์เริ่มทำงาน
วนซ้ำ:
    อ่านสัญญาณ DIO และ FINISH
    ถ้าสัญญาณ DIO เป็น '1'
        อ่านจุดภาพ(เอาต์พุตฮาร์ดแวร์)จากพอร์ตเข้าไปเก็บไว้ในหน่วยความจำ
        ส่งจุดภาพตัวต่อมาออกสู่พอร์ต
    ถ้าสัญญาณ FINISH เป็น '1'
        ส่งสัญญาณ START เป็น '0'
        ให้ทำงานการขัดจังหวะ (enable interrupt)
        เขียนจุดภาพขอบภาพจากหน่วยความจำลงในแฟ้มข้อมูลปลายทางและปิดแฟ้ม
        จบการทำงานของโปรแกรม
    กลับไปวนซ้ำ
```

รูปที่ 5.12 แสดงรหัสเทียมของโปรแกรมที่ใช้ในการติดต่อกับฮาร์ดแวร์

การจองหน่วยความจำจะต้องจองให้มากกว่าข้อมูลของรูปภาพเพื่อเป็นข้อมูลเปล่าในการป้อนให้ฮาร์ดแวร์เพราะทุกการคำนวณหนึ่งจุดภาพ(16 สัญญาณนาฬิกา)ของฮาร์ดแวร์จะให้ค่า DIO หนึ่งครั้ง และฮาร์ดแวร์ต้องทำงานเป็นจำนวน 257 จุดภาพจึงจะเริ่มต้องการค่าจุดภาพของภาพต้นกำเนิดเป็นอินพุต และฮาร์ดแวร์ต้องทำงานอีกเป็นจำนวน 780 จุดภาพจึงจะให้ค่าจุดภาพขอบภาพที่ฮาร์ดแวร์คำนวณได้ออกมา

ผลการทดสอบฮาร์ดแวร์

ฮาร์ดแวร์ต้องใช้สัญญาณนาฬิกาในการทำงานเป็นจำนวน 1,073,375 สัญญาณนาฬิกา บวกอีก 7 สัญญาณนาฬิกา(เพื่อให้สัญญาณ DIO เป็น '1') หรือเท่ากับประมาณ 67,087 จุดภาพ (1 จุดภาพใช้ 16 สัญญาณนาฬิกา) หมายความว่าในการสั่งให้ฮาร์ดแวร์ทำการหาขอบภาพนั้น คอมพิวเตอร์และฮาร์ดแวร์จะต้องมีการรับข้อมูลและส่งข้อมูลเป็นจำนวนอย่างละ 67,087 ไบต์ คือ รับหนึ่งไบต์ส่งหนึ่งไบต์สลับกันไป เท่ากับมีข้อมูลวิ่งเข้าออกจากพอร์ตของคอมพิวเตอร์เป็นจำนวน 134,714 ไบต์

เนื่องจากบอร์ดที่ใช้ประกอบฮาร์ดแวร์ได้ต่อขึ้นเอง โดย FPGA เบอร์ XC4003-6 และ XC4005-6 ที่ใช้ต่างก็เป็นไอซีตัวถังแบบ QFP ขนาด 84 ขา จึงทำให้มีปัญหาเรื่องสัญญาณรบกวน ทำให้ผลลัพธ์ที่ได้จากฮาร์ดแวร์มีค่าผิดไปจากผลของซอฟต์แวร์ โดยความถี่สูงสุดของสัญญาณนาฬิกาที่ใช้ทำการทดสอบคือ 500 กิโลเฮิรตซ์ ดังนั้นจึงได้ลดความถี่ของสัญญาณนาฬิกาลงเพื่อลดความผิดพลาดที่เกิดจากสัญญาณรบกวนนี้ลง โดยตัวกำเนิดสัญญาณนาฬิกาที่ใช้คือ มอดูลออสซิลเลเตอร์ ความเร็ว 4 เมกะเฮิรตซ์ แล้วใช้เกตทางตรรกภายใน FPGA ลดทอนค่าความถี่นี้ลงให้เหลือ 125 กิโลเฮิรตซ์ ซึ่งฮาร์ดแวร์ให้ผลลัพธ์ที่ยอมรับได้ โดยฮาร์ดแวร์ใช้สัญญาณนาฬิกาในการทำงานเท่ากับ 1,073,382 สัญญาณนาฬิกา ดังนั้นความเร็วของฮาร์ดแวร์เมื่อคิดเป็นเวลาคือประมาณ 8.59 วินาที ซึ่งก็คือประมาณ 7,812 ไบต์/วินาที (นับรวมทั้งจุดภาพอินพุต, เอาต์พุตและข้อมูลเปล่า)หรือเท่ากับประมาณ 7,631 จุดภาพ/วินาที(เอาเฉพาะจำนวนจุดภาพในหนึ่งรูปภาพมาคิด)

เนื่องจากผลลัพธ์ที่ได้จากวงจร Binomial (1) และ (2) มีค่าผิดไปจากผลที่ได้จากซอฟต์แวร์ ถึงแม้มองด้วยตาเปล่า ผลลัพธ์(รูป)ทั้งสองจะดูไม่แตกต่างกัน แต่ค่าที่ผิดนี้ทำให้ผลลัพธ์ปลายทางของฮาร์ดแวร์ที่ประกอบขึ้นให้ภาพลายเส้นขอบที่ไม่ต่อเนื่อง และมีสัญญาณรบกวนมากกว่าผลของซอฟต์แวร์ แต่ด้วยตาเปล่ายังคงมองออกว่าภาพลายเส้นขอบเป็นรูปอะไร และคล้ายคลึงกับรูปที่ได้จากซอฟต์แวร์ ดูรูปเปรียบเทียบระหว่างรูปขอบภาพที่ได้จากฮาร์ดแวร์หลังผ่านซอฟต์แวร์ตรวจสอบขีดเริ่มเปลี่ยนค่าสูงแบบฮิสเตอร์เรซิส กับรูปที่ได้จากซอฟต์แวร์(รูปที่ถูกต้อง)ในรูปที่ 5.13* ถึง 5.15 ในหน้าที่ 71 ถึง 73

---

* ภาพต้นกำเนิดในรูปที่ 5.12 ถึง 5.13 มีที่มาจากไฟล์รูปภาพที่มากับหนังสือของ Sid-Ahmed (1995) และของ Baxes (1994) ถ้าเป็นของ Baxes ต้องนำมาแปลงเป็นขนาด 256 x 256 จุดภาพก่อน

สรุปท้ายบท

โปรแกรมที่ใช้ในการเขียนและออกแบบวงจรคือโปรแกรม Workview PLUS ของบริษัท Viewlogic Systems โดยสามารถทำการจำลองและมีไลบราลี FPGA ของบริษัท Xilinx ให้ใช้ช่วยในการออกแบบ ซึ่งการออกแบบก็ได้ใช้ไลบราลีนี้ในการออกแบบทั้งหมด

หลังจากออกแบบเสร็จก็ทำการทดสอบวงจรที่ออกแบบ โดยการจำลอง ซึ่งผลลัพธ์ที่ได้จากวงจรจะนำไปเปรียบเทียบกับผลลัพธ์ที่ได้จากซอฟต์แวร์(ที่เขียนให้มีการคำนวณเหมือนวงจรที่ออกแบบทุกประการ เพื่อใช้เป็นผลลัพธ์ต้นแบบ) ซึ่งปรากฏว่าวงจรทั้ง 4 วงจรสามารถทำงานได้ถูกต้องทั้งหมด

แล้วก็นำแฟ้มข้อมูลที่ออกแบบได้จาก Workview PLUS มาเป็นอินพุตให้กับโปรแกรม XDM ของ Xilinx ทำการเปลี่ยนแฟ้มข้อมูลเป็นแฟ้มข้อมูลบรรจุลงใน FPGA เพื่อโปรแกรมให้ FPGA ทำงาน ซึ่ง ณ จุดนี้ วงจร Binomial Filter (1) และ (2) และวงจร Nonmax Suppression and Low Threshold สามารถใช้กับ FPGA ของ Xilinx เบอร์ XC4005-6 ได้ ส่วนวงจร Finding $|G|$ and ϕ จะต้องทำการแบ่งออกเป็น 2 วงจรย่อย จึงจะสามารถใช้กับเบอร์ XC4003-6 ได้

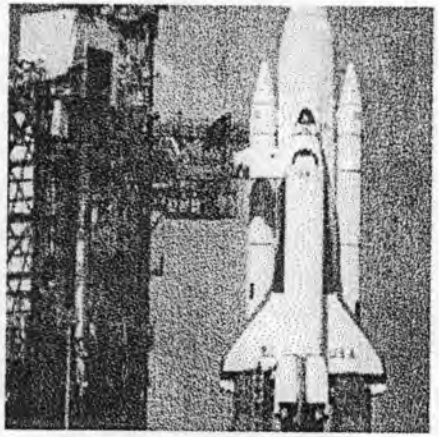
เมื่อต่อฮาร์ดแวร์จริงแล้ว ผลลัพธ์ที่ได้จากฮาร์ดแวร์ ได้ผลที่ไม่ถูกต้อง โดยมีผลผิดพลาดไปจากผลของวงจรที่จำลองได้ แต่ถ้าใช้สายตาในการตรวจสอบ จะพบว่ารูปขอบภาพที่ได้จากการทำงานของฮาร์ดแวร์ ยังคงให้เส้นขอบภาพที่สำคัญออกมา

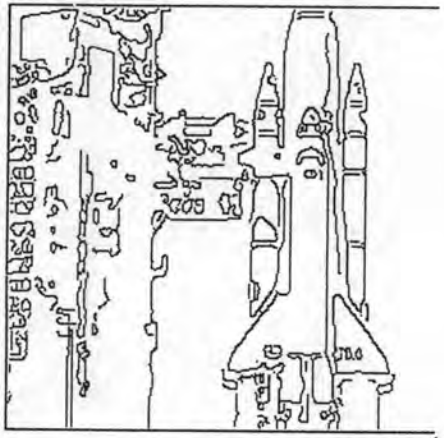
ภาพต้นกำเนิด

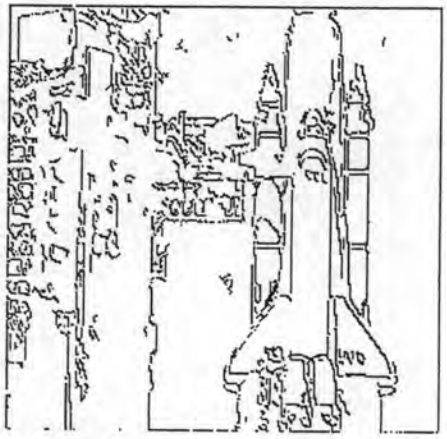
ผลจากซอฟต์แวร์

ผลจากฮาร์ดแวร์

เทรคโชลด์ค่าต่ำ = 10 และค่าสูง = 20



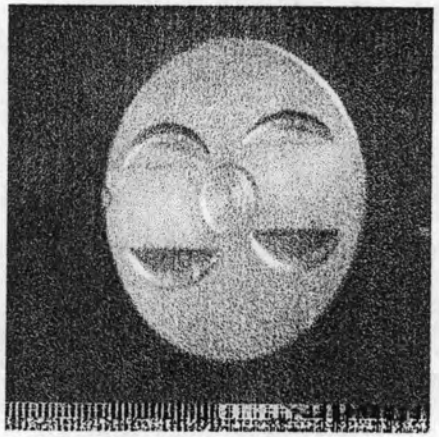




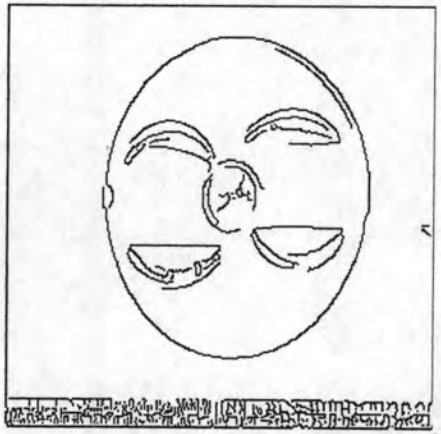
เทรคโชลด์ค่าต่ำ = 10 และค่าสูง = 20

<u>รูปที่ 5.13</u> แสดงรูปต้นกำเนิด, รูปขอบภาพที่ได้จากซอฟต์แวร์ และรูปขอบภาพที่ได้จากฮาร์ดแวร์ (ชุดที่ 1)

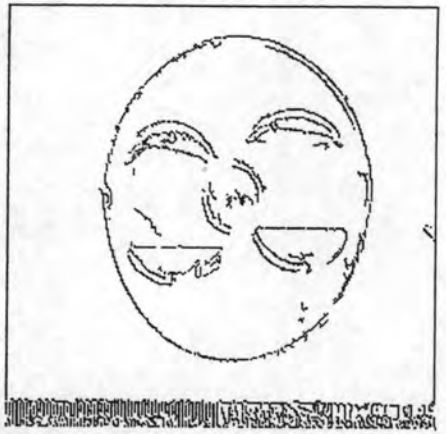
ภาพต้นกำเนิด



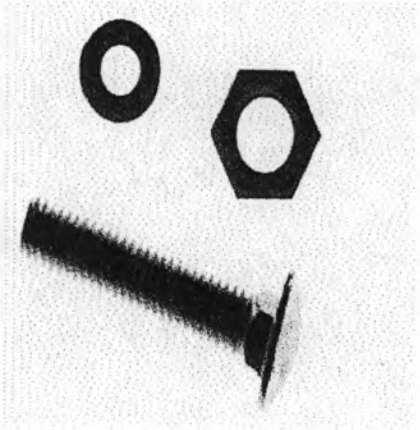
ผลจากซอฟต์แวร์

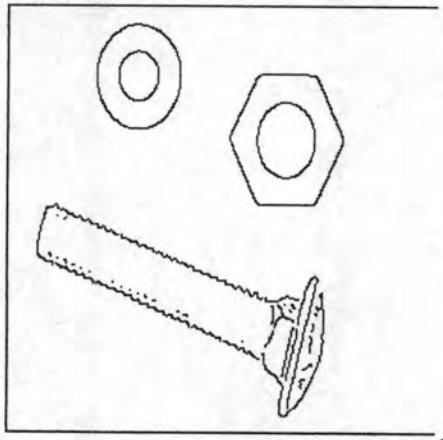


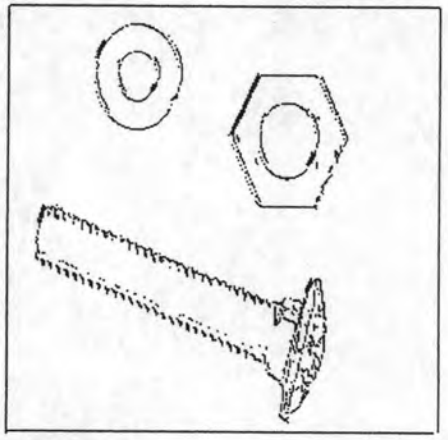
ผลจากฮาร์ดแวร์



เทรดโชลด์ค่าต่ำ = 10 และค่าสูง = 20



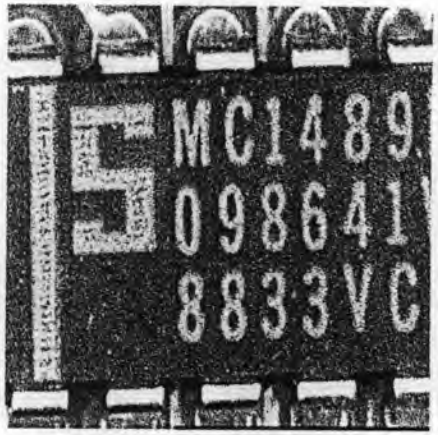




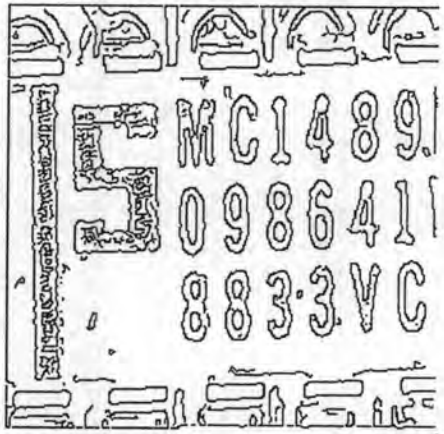
เทรดโชลด์ค่าต่ำ = 5 และค่าสูง = 10

<u>รูปที่ 5.14</u> แสดงรูปต้นกำเนิด, รูปขอบภาพที่ได้จากซอฟต์แวร์ และรูปขอบภาพที่ได้จากฮาร์ดแวร์ (ชุดที่ 2)

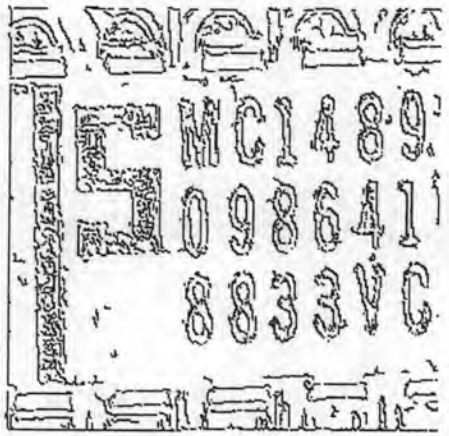
ภาพต้นกำเนิด



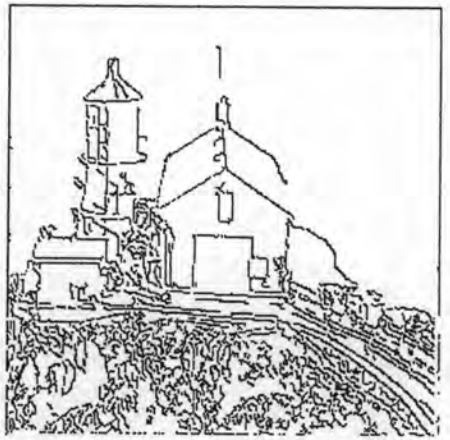
ผลจากซอฟต์แวร์



ผลจากฮาร์ดแวร์



เทรดโชลด์ค่าต่ำ = 10 และค่าสูง = 30



เทรดโชลด์ค่าต่ำ = 10 และค่าสูง = 30

<u>รูปที่ 5.15</u> แสดงรูปต้นกำเนิด, รูปขอบภาพที่ได้จากซอฟต์แวร์ และรูปขอบภาพที่ได้จากฮาร์ดแวร์ (ชุดที่ 3)